PUESTA EN OPERACIÓN DEL SISTEMA DE CONTROL DE MOVIMIENTO DE LA ARQUITECTURA DE AUTOMATIZACIÓN ROCKWELL AUTOMATION

JUAN JOSE ARIAS JARAMILLO

UNIVERSIDAD PONTIFICIA BOLIVARIANA
FACULTAD DE INGENIERA ELECTRONICA
ESCUELA DE INGENIERIA
SECCIONAL BUCARAMANGA
2011

PUESTA EN OPERACIÓN DEL SISTEMA DE CONTROL DE MOVIMIENTO DE LA ARQUITECTURA DE AUTOMATIZACIÓN ROCKWELL AUTOMATION

JUAN JOSE ARIAS JARAMILLO

MONOGRAFIA

JUAN CARLOS MANTILLA SAAVEDRA
DIRECTOR DEL PROYECTO

UNIVERSIDAD PONTIFICIA BOLIVARIANA
FACULTAD DE INGENIERA ELECTRONICA
ESCUELA DE INGENIERIA
SECCIONAL BUCARAMANGA
2011

Nota de Aceptación

____
____
____
____
____
____

____
Firma de Presidente del Jurado

____
Firma del Jurado

____
Firma del Jurado

Bucaramanga, 19 Febrero de 2011

AGRADECIMIENTOS

A Dios, por todas las oportunidades que me ha brindado en la vida, sobre todo el estar con vida hoy, disfrutando de la grata compañía de mi familia y de las personas que estimo.

A mis padres, por todo su esfuerzo y entrega, por ser mi apoyo incondicional y ser la vos de mi conciencia.

A mi hermana por el ánimo constante, por el apoyo incondicional y por los consejos oportunos.

Al ingeniero Juan Carlos Mantilla Saavedra, por su paciencia, su apoyo y su continua entrega.

Al ingeniero Marco Antonio Villamizar Araque, por su apoyo y disposición.

A la ingeniera Leidy Johanna Olarte Silva por su disposición y colaboración en el proceso y desarrollo del proyecto.

Mil gracias a todos por su grandioso e incalculable aporte.

Juan José Arias Jaramillo.

OBJETIVOS

General

Poner en operación el sistema de control de movimiento de la arquitectura de automatización *Rockwell Automation*.

Específicos

- Recopilar y analizar la información técnica de la arquitectura de control Rockwell Automation basada en el PLC ControlLogix de Allen Bradley.
- Instalar y cablear los dispositivos del sistema de control de movimiento de la arquitectura Rockwell Automation.
- Configurar y programar el módulo SERCOS y los Servodrives para la operación del sistema Motor-Encoder.
- Poner en funcionamiento el sistema integral de control de movimiento.
- Desarrollar la documentación del proyecto.

CONTENIDO

LISTA DE TABLAS

LISTA DE FIGURAS

LISTA DE ANEXOS

RESUMEN GENERAL DE MONOGRAFIA

TITULO: **PUESTA EN OPERACIÓN DEL SISTEMA DE CONTROL DE MOVIMIENTO DE LA ARQUITECTURA DE AUTOMATIZACIÓN ROCKWELL AUTOMATION**

AUTOR(ES): **JUAN JOSE ARIAS JARAMILLO**

FACULTAD: **FACULTAD DE INGENIERIA ELECTRONICA**

DIRECTOR(A): **JUAN CARLOS MANTILLA SAAVEDRA**

RESUMEN

El presente proyecto se realizó en base al sistema integrado de control de movimiento de la arquitectura Rockwell Automation. En su desarrollo se realizó un estudio teórico de las características generales del control de movimiento, de modo que se organizaron los conceptos de acuerdo al alcance y los objetivos de la aplicación. A su vez se efectuó un reconocimiento de la arquitectura integrada de Rockwell Automation implementada en el Laboratorio de Automatización, para identificar la estructura y prestaciones que dicha arquitectura ofrece, haciendo especial énfasis en la arquitectura integrada de control de movimiento mediante la utilización de la plataforma Logix. Finalmente se llevó a cabo la puesta en operación del sistema de control de movimiento, utilizando los conocimientos obtenidos en el estudio y recopilación teórica del sistema integrado de control de movimiento y de cada uno de los dispositivos con los que se contaban en el laboratorio de Automatización de Procesos. La puesta en operación del sistema comprendió el montaje y cableado de los dispositivos, la configuración y la programación de los equipos y la ejecución de las pruebas de funcionamiento para garantizar el correcto funcionamiento del sistema. Adicionalmente a los objetivos planteados en el proyecto, se desarrolló una interfaz gráfica de usuario utilizando el software FactoryTalk Studio de Rockwell Automation, para brindar una interacción más clara entre el usuario y el sistema.

PALABRAS CLAVE: Controlador, Servo-motor, Servo-Drive, Módulo SERCOS, Fibra óptica.

V° B° DIRECTOR DE MONOGRAFIA

ABSTRACT OF THESIS PROJECT

TITLE: **STARTING SYSTEM**
OPERATION CONTROL OF MOVEMENT OF ROCKWELL AUTOMATION ARCHITECTURE

AUTHOR(S): **JUAN JOSE ARIAS JARAMILLO**

DEPARTAMENT: **ELECTRONIC ENGINEERING**

DIRECTOR: **JUAN CARLOS MANTILLA SAAVEDRA**

ABSTRACT

This project was made using control movement integrate system of architecture Rockwell Automation. During the investigation was performed a theoretical study of general control movement characteristics, allowing an organization of the concepts agree with the size and objectives of its application. At the same time was recognized the integrate architecture of Rockwell Automation used in Automation Laboratory, like a way to identify structure and benefits that this could offer, making important control movement of integrate architecture by used of Logix stage. Finally, the system control movement was placed in operation, using knowledge of study and theoretical compilation of this system and each one device available in Process Automation Laboratory. The operation of this system was integrated by device assembly and cabling, equipment configuration and programming and execution of work proofs to ensure correct operation of system. Also was developed a user graphical interface using FactoryTalk Studio software of Rockwell Automation, giving interaction more clear between user and system.

KEYWORDS: Controller, Servomotor, Servo drive, SERCOS Module, Fiber optics.

V° B° THESIS DIRECTOR

INTRODUCCIÓN

La automatización es el proceso mediante el cual se trasfieren las tareas de producción, realizadas habitualmente por operadores humanos a un conjunto de elementos tecnológicos capaces de aumentar la productividad y la calidad, disminuyendo los tiempos y costos de producción, a la vez que permite tener una menor tasa de accidentalidad laboral. [1]

Las arquitecturas basadas en PLC (Controlador Lógico Programable), como centros de mando o cerebros del sistema de control, son muy utilizadas, aunque actualmente estos dispositivos están siendo remplazados por equipos más robustos y más escalables, a los cuales se les conoce como PAC´s (Controladores de Automatización Programables).

El PAC, posee las mismas características de ejecución y programación lógica que posee un PLC, pero adicionalmente cuenta con la versatilidad de monitorización y el poder de cálculo de un PC; lo que lo ha convertido en una herramienta eficaz para el desarrollo tecnológico de muchas empresas industrializadas.

Existen gran cantidad de aplicaciones industriales que requieren un especial cuidado y atención de variables como posición y velocidad, para el desarrollo de sus productos, lo que demanda un control capaz de satisfacer estas demandas de una manera óptima y precisa. Este tipo de control es conocido como control de movimiento y está diseñado para proporcionar el control adecuado dependiendo del tipo de aplicación que se esté trabajando, basado en una estructura predeterminada de control, adecuada a las características de los equipos que se utilizan para dicha aplicación.

En el mercado se pueden encontrar varias empresas dedicadas al desarrollo de sistemas integrales de automatización y de aplicaciones, algunas de las que se pueden encontrar a nivel regional son: SIEMENS, Schneider Electric y Rockwell Automation.

El presente proyecto se realizó en base al sistema integrado de control de movimiento de la arquitectura Rockwell Automation disponible en el laboratorio de automatización de procesos de la universidad, permitiendo realizar una apropiación teórica y practica que facilite a futuro los trabajos sobre este sistema.

1. MARCO TEÓRICO

1.1. FUNDAMENTOS DEL CONTROL DE MOVIMIENTO

Un sistema de control de movimiento tiene como propósito controlar alguna o la combinación de las siguientes variables:

- Posición
- Velocidad
- Aceleración
- Torque

Este sistema generalmente se encuentra conformado por seis componentes básicos que son: Software de aplicación, Controlador de movimiento, Amplificador o drive, Motor, Elementos mecánicos y Dispositivo de retroalimentación. [2]

1.1.1. Software de Aplicación.

Este permite generar las posiciones deseadas y perfiles de control de movimiento.

1.1.2. Controlador de Movimiento.

Este actúa como cerebro del sistema, toma los perfiles de las posiciones y movimientos deseados y crea las trayectorias que deberán seguir los motores, y posteriormente entrega una señal de voltaje, a los servomotores; o una señal de pulsos de paso y dirección, a motores de pasos.

El controlador de movimiento usa generalmente lazos de control PID, dado que este tipo de control demanda un alto nivel determinismo y es vital para una operación consistente. [3]

- **Control de Velocidad.** En la figura 1 se muestra un control típico de velocidad PID comercial, en ella se observa el diagrama a bloques de un controlador "PID", donde la entrada de referencia es un valor de velocidad "*Velocity Request*" que se calcula cada n milésimas de segundo, recibe el valor real de velocidad a través del encoder "*Velocity Feddback*", y tomando en cuenta la acción del PID, así como de los otros lazos de Feed Forward a su salida genera el valor de referencia para el servo-accionamiento.

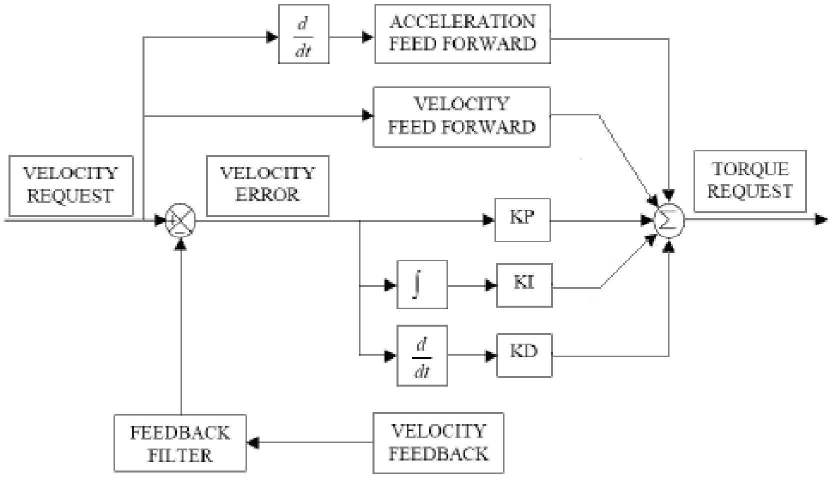


Figura 1. Control típico de Velocidad PID Comercial. [3]

- **Control de Posición.** La figura 2 presenta el diagrama de bloques de un control típico de Posición PID comercial, en el cual la entrada de referencia es un valor de Posición "*Position Request*" que se calcula cada n milésimas de segundo, recibe el valor real de Posición a través del encoder "*Position Feddback*" y tomando en cuenta la acción del PID, así como de los otros lazos de Feed Forward a su salida genera el valor de referencia para el servo-accionamiento. [3]

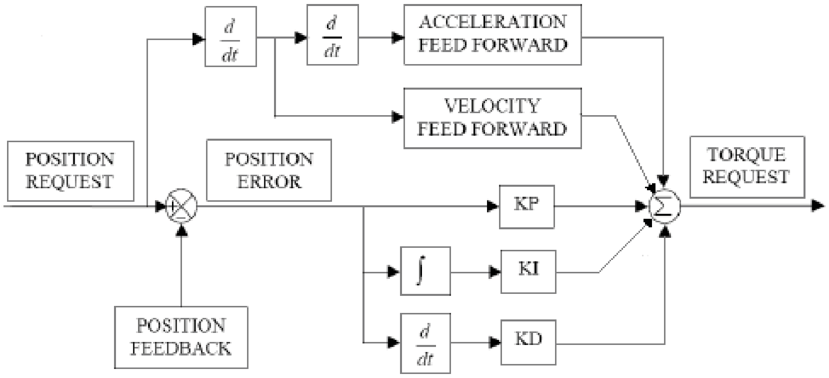


Figura 2. Control Típico de Posición PID Comercial. [3]

- **Perfil de Movimiento.** El movimiento de los objetos móviles se puede especificar por varios parámetros, que define en conjunto el perfil de movimiento. Estos parámetros son:

  - Velocidad (Speed). El índice del cambio de la posición
  - Aceleración (Acceleration). El índice del cambio de la velocidad durante un aumento de la velocidad.

- Desaceleración (Deceleration). El índice del cambio de la velocidad durante una disminución de la velocidad.
- Tirón (Jerk). El índice del cambio de aceleración o desaceleración.

Existen muchos perfiles de movimiento, como: lineal, rampa, trapezoidal, Sinusoidal, Cubica, Curva-S, etc., sin embargo los más utilizados por los controladores de movimiento son el trapezoidal y la curva S. [3]

- **Perfil Trapezoidal.** Un perfil trapezoidal es típicamente compuesto de tres secciones, una fase de aceleración, una fase de velocidad constante y una fase de desaceleración. Como se muestra en la figura 3, en la primera parte la velocidad aumenta definida por el índice de aceleración hasta alcanzar la velocidad objetivo, la velocidad entonces permanecerá constante hasta el punto de reducción de la velocidad, donde la velocidad disminuirá entonces definida por el índice de desaceleración hasta parar. [3]

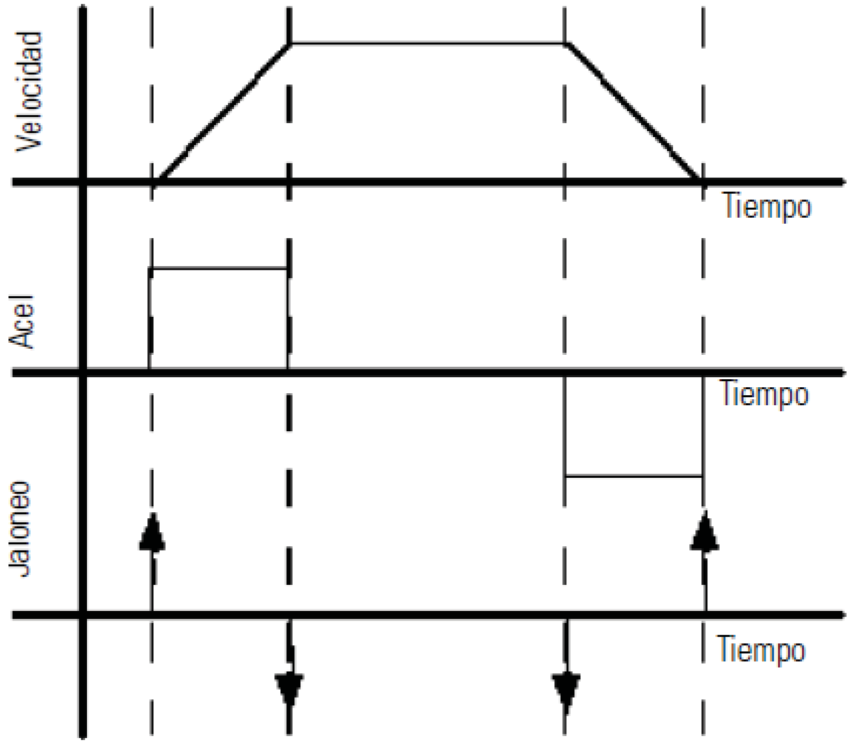


Figura 3. Perfil Trapezoidal. [4]

- **Perfil Curva-S.** Un perfil de curva-s puede ser considerado como un agregado del perfil trapezoidal. Este típicamente está compuesto de siete secciones, cuatro fases de Jerk (tirón/jaloneo), una fase de aceleración, una fase de velocidad constante y una fase de desaceleración. [3]

En la figura 4 se observan las 7 fases, las áreas sombreadas representan un Jerk, es decir que ante un cambio de fase del perfil trapezoidal este cambio es suavizado con una curva, las fases que no están sombreadas trabajan idénticamente al perfil trapezoidal, es decir las fase de aceleración hasta la velocidad objetivo, la fase de velocidad constante y la fase de desaceleración hasta cero. [3]

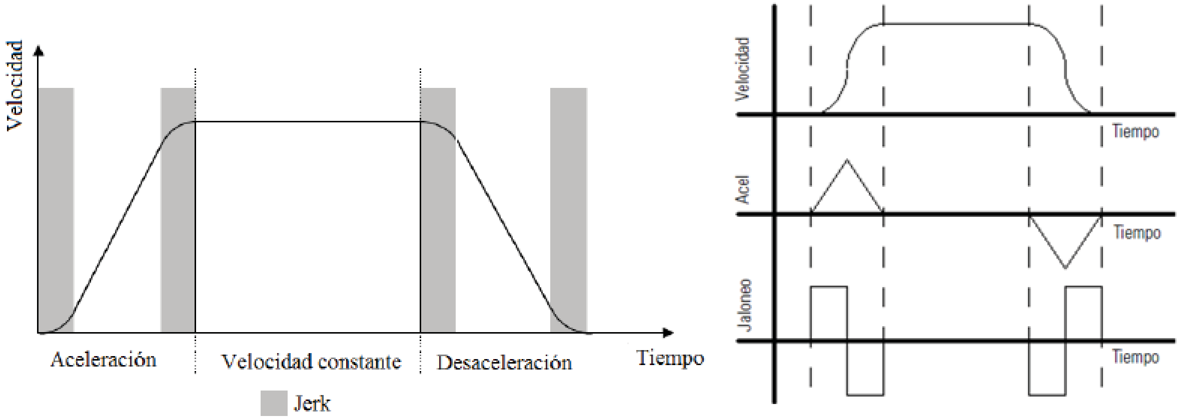


Figura 4. Perfil Curva-S. [3] [4]

1.1.3.Amplificador o Drive.

Toma los comandos del controlador y proporciona la corriente necesaria para dirigir o girar el motor. [2]

1.1.4.Motor.

Este convierte la energía eléctrica en energía mecánica y produce el torque requerido para moverse a la posición deseada. [2]

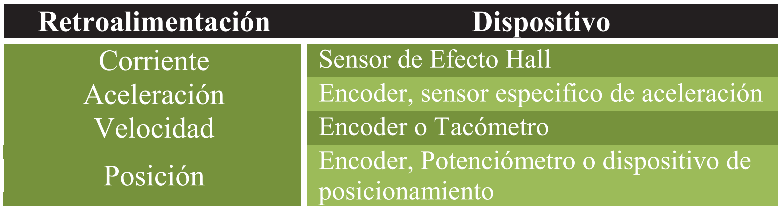
1.1.5.Elementos Mecánicos.

Los motores están diseñados para proporcionar torque a algunos dispositivos mecánicos. Éstos incluyen deslizadores lineales, brazos robóticos y actuadores especiales. [2]

1.1.6.Dispositivo de Retroalimentación.

Algunas aplicaciones de control de movimiento (por ejemplo, control de motores de pasos) no requieren ningún dispositivo de retroalimentación de posición, pero para los servomotores sí es vital. El dispositivo de retroalimentación, generalmente un codificador de cuadratura, detecta la posición del motor y reporta el resultado al controlador, y de esa manera cierra el lazo con el controlador de movimiento. [2]

En la tabla 1 se presentan algunos dispositivos que se pueden utilizar para medir las variables fundamentales del control de movimiento.



Retroalimentación	Dispositivo
Corriente	Sensor de Efecto Hall
Aceleración	Encoder, sensor especifico de aceleración
Velocidad	Encoder o Tacómetro
Posición	Encoder, Potenciómetro o dispositivo de posicionamiento

Tabla 1. Dispositivos de Realimentación. [3]

1.2. ARQUITECTURA INTEGRADA DE ROCKWELL AUTOMATION

La Arquitectura Integrada de Rockwell Automation es una infraestructura de automatización industrial que proporciona soluciones escalables para todo el rango de las disciplinas de automatización, entre ellas, control secuencial, de movimiento, de procesos, control de variadores, seguridad e información.

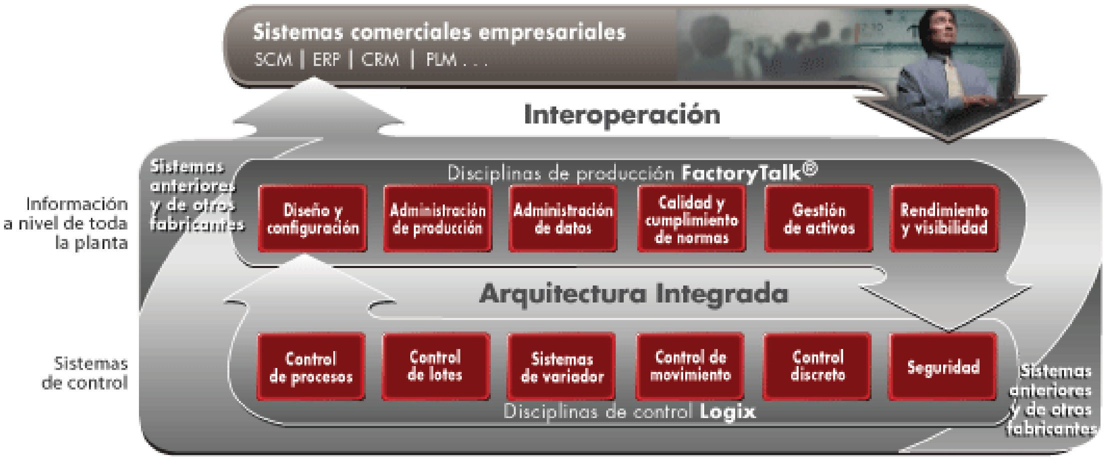


Figura 5. Estructura General de la Arquitectura Integrada [5]

A diferencia de las arquitecturas de control tradicionales, la Arquitectura Integrada reduce el costo total de adquisición que paga el cliente, al usar una sola infraestructura de control tanto grande como pequeña. Esto permite a los usuarios de la Arquitectura Integrada reutilizar los diseños y prácticas de ingeniería para reducir el tiempo y costo de desarrollo, responder más rápidamente a las demandas de los clientes y el mercado, reducir los costos de mantenimiento y el tiempo improductivo, y tener fácil acceso a los datos de la planta y de producción desde sistemas empresariales para una mejor toma de decisiones administrativas.

La Arquitectura Integrada es posible a través de una combinación única de tecnologías de habilitación que incluyen la plataforma de controlLogix, la arquitectura de red abierta NetLinx, la plataforma de visualización View y los servicios de información y datos FactoryTalk. [6]

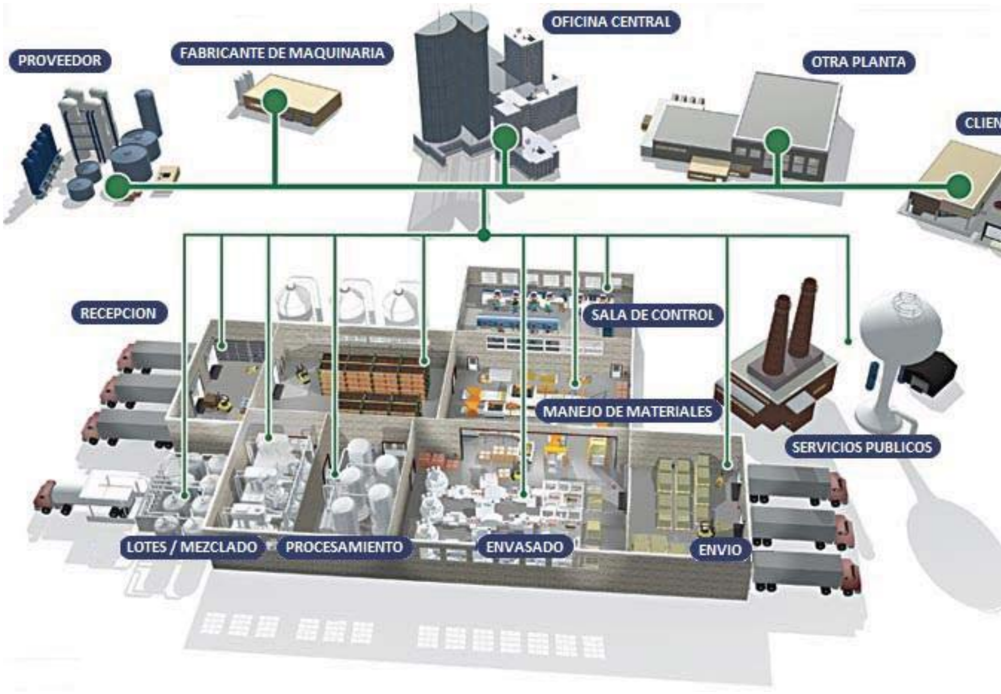


Figura 6. Estructura Global Arquitectura Integrada [7]

- La plataforma de control Logix proporciona una variedad de funciones y tamaños de controlador, y permite que el usuario elija el controlador más económico para la aplicación. Todos los controladores Logix se programan a través de un solo paquete de software que permite al cliente elegir entre los lenguajes de texto estructurado, diagrama de escalera, diagrama de bloques de funciones y diagrama de función secuencial con los cuales programar. La programación se realiza en el controlador Logix y no en controladores separados, lo cual reduce considerablemente los costos de diseño, hardware y programación.

- Las redes abiertas NetLinx proporcionan una infraestructura de red para flujo de información. Las redes NetLinx comparten un protocolo común que permite el flujo transparente de información desde el dispositivo de planta más pequeño hasta el sistema administrativo de la empresa, sin programación adicional, lo cual ayuda a los usuarios a aumentar la flexibilidad, reducir el costo de integración e instalación, y aumentar la productividad. [6]

- Las plataformas de visualización View proporcionan un conjunto escalable y unificado de soluciones de interface de operador y monitoreo de nivel de supervisión que comparten el mismo software de desarrollo, ofreciendo un desarrollo e implementación más rápido, mayor flexibilidad y menor costo de adquisición.

- FactoryTalk es un conjunto de servicios incorporado en los componentes de la Arquitectura Integrada que simplifica considerablemente la producción, transferencia y uso de datos. Los servicios FactoryTalk proporcionan al usuario la capacidad de crear un tag una sola vez y reutilizarlo en todo el sistema de control. En un sistema habilitado para FactoryTalk, los componentes de visualización comparten una base de datos de tag común con la plataforma de control, lo cual reduce considerablemente el tiempo de ingeniería y el costo dedicado a crear tags individuales para uso en los niveles de controlador, visualización e información, y a establecer las correspondencias entre ello. [6]

1.2.1. Plataforma de ControlLogix

La plataforma de Control Logix abarca las tecnologías de control, de conexión en red y de visualización que juegan un papel decisivo para conseguir la integración horizontal.

- **Control.** Debido a la variedad de aplicaciones que pueden encontrarse en una sola planta, generalmente se hace uso de múltiples controladores: un controlador lógico programable para aplicaciones discretas, un controlador de movimiento para aplicaciones de movimiento, y un sistema de control distribuido (DCS) para aplicaciones de control de procesos, entre otros. Los controladores programables de automatización (PAC) eliminan este obstáculo al proporcionar soluciones escalables totalmente integradas para toda la gama de disciplinas de automatización, incluidas aplicaciones discretas, de movimiento, de procesos, de lotes, de variadores y de seguridad. [8]

  Estos controladores modernos son más rápidos, fiables y precisos. Otro gran adelanto de los controladores programables de automatización es su Backplane incorporado en red, que permite mezclar varios procesadores, redes y E/S sin ninguna restricción en un mismo chasis. Esta arquitectura flexible responde a las crecientes necesidades de comunicación y permite distribuir fácilmente el control de un chasis a otro, a medida que aumentan las necesidades del sistema. Además, con la transición a redes digitales abiertas como DeviceNet, ControlNet y EtherNet/IP, los usuarios pueden conectar en red dispositivos de varios suministradores en un controlador único. Igual que el Backplane ha dejado de ser el antiguo rack, el controlador evoluciona a una máquina más eficiente y expandible. Los controladores programables de automatización que existen en la actualidad ofrecen una capacidad de escalado que permite a los usuarios implementar un sistema que responda a las necesidades de aplicación actuales, pero que puede expandirse fácilmente de acuerdo a las necesidades futuras.

Figura 7. Plataforma ControlLogix [8]

La familia de controladores PAC Logix de Rockwell Automation incluye Allen-Bradley ControlLogix, FlexLogix, CompactLogix, SoftLogix, GuardLogix y DriveLogix. Todos los controladores de la familia Logix aprovechan la misma máquina de control y el mismo software de programación y configuración RSLogix 5000. RSLogix 5000 cumple la norma IEC-61131-3 y ofrece al usuario una serie de lenguajes de diagrama de lógica de escalera, de diagrama de bloque de funciones, de diagrama de función secuencial y de texto estructurado para obtener mayores eficiencias en la programación de aplicaciones multidisciplinarias de control. En el caso de aplicaciones de lote, se incorpora un configurador lógico de estado de fase que cumple con la norma ISA S88 en todos los controladores de la familia Logix, lo que proporciona un marco estándar para la configuración y el secuenciamiento de la lógica de estado de fase. La utilización de PhaseManager en toda la plataforma proporciona capacidad de escalado sin precedentes, lo que disminuye el coste a los usuarios. (Ver Anexo A) [8]

- **Comunicación.** El impulso para integrar la empresa de fabricación desde el dispositivo más simple hasta el sistema de información de más alto nivel supone una tarea que ninguna red puede gestionar sola. Las diferencias en los requisitos de aplicación, como el volumen y el determinismo, definen los requisitos de la red. En una instalación de producción, usted puede emplear redes a nivel de los dispositivos con requisitos de volumen relativamente bajos, así como redes de información y de control con requisitos de volumen mucho más elevados. [8]

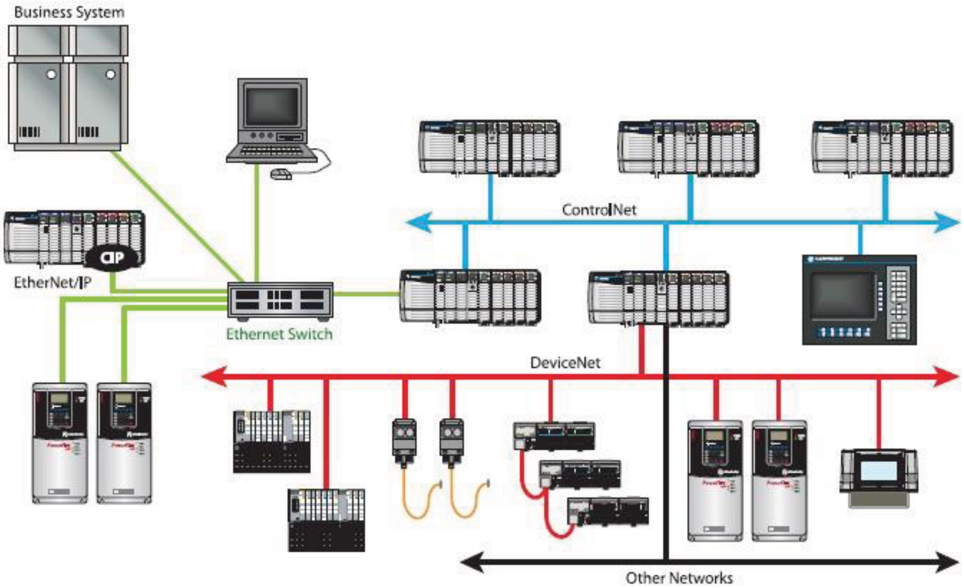


Figura 8. Redes de Comunicación Utilizadas por Rockwell [8]

Este concepto parece bastante claro, pero la integración de valiosa información sobre la producción puede ser bastante complicada si estas redes no utilizan el mismo protocolo.

Si conecta tres dispositivos “compatibles con Ethernet” de tres suministradores distintos, hay muchas posibilidades de que éstos no se comuniquen, ya que pueden estar utilizando protocolos diferentes. Como consecuencia, será necesario contar con hardware caro y con tiempo de programación para traducir los mensajes de manera que puedan entenderse en todas las redes. Esta “traducción” no sólo es costosa para el usuario sino que también introduce tiempos de espera, lo que afecta al rendimiento de la red. [8]

Las comunicaciones transparentes exitosas en toda la planta requieren redes que utilicen un protocolo común. Rockwell Automation ha adoptado el uso de redes de protocolo industrial común (CIP) para su Arquitectura Integrada. EtherNet/IP, ControlNet y DeviceNet utilizan el protocolo CIP, tal como establece la ODVA, la organización para la gestión de redes CIP independiente. [8]

Al utilizar redes CIP abiertas, los clientes reciben comunicaciones transparentes desde los dispositivos de la planta hasta el sistema de TI sin necesidad de enrutamiento, conexión en puente, asignación de datos, proxies ni programación. Estas redes funcionan colectivamente con cientos de productos interoperables de más de 400 suministradores. Rockwell Automation también proporciona interoperabilidad con otras redes industriales populares, como Foundation Fieldbus y HART, que suelen utilizarse en las aplicaciones de procesamiento. (Ver Anexo B) [8]

- **Visualización:** Con los avances en control y conexiones en red, los usuarios buscan soluciones de interface operador máquina (HMI) que puedan crear una estrategia de visualización común a toda la empresa.

  En la actualidad muchos clientes tienen una plataforma de desarrollo distinta para cada nivel (dispositivos de empresa, ordenadores industriales individuales o dispositivos incorporados de gama inferior) en sus instalaciones. En ocasiones existen hasta tres o cuatro plataformas de desarrollo de HMI distintas, por lo que los usuarios se ven obligados a mantener licencias de software y conocimientos de distintas empresas o distintos paquetes de software, de acuerdo con el tamaño de la aplicación. Cerrar la brecha entre las plataformas puede ser costoso para los clientes porque esto supone aumentar los costes de formación, reducir la productividad, disminuir la conectividad y reducir la capacidad de personalizar. En resumen, aumenta los costes totales de propiedad y, en consecuencia, disminuye la competitividad. [8]

Figura 9. Pantallas de Visualización [8]

La solución es una arquitectura de HMI escalable que emplee la tecnología que los desarrolladores puedan aprovechar para construir aplicaciones HMI que abarquen desde el nivel de la máquina, con terminales incorporados de Windows CE y ordenadores de mano, hasta el nivel de supervisión de plataformas de gama superior autónomas y distribuidas. Las aplicaciones también se pueden ampliar y personalizar. Estas nuevas tecnologías ofrecen a los suministradores la oportunidad de crear un entorno de HMI más expandible y abierto.

Rockwell Automation proporciona una suite escalable y unificada de soluciones de visualización y de control para prácticamente cualquier nivel de la empresa de fabricación. Todos los productos de visualización son independientes de la plataforma y se construyen sobre un entorno de configuración común. Esto deja paso a una oportunidad significativa para reducir costes y mejorar el tiempo de desarrollo, a la vez que ofrece la habilidad para volver a utilizar aplicaciones en distintas plataformas, a fin de hacer frente a nuevos requisitos.

Asimismo, los usuarios pueden elegir una interface que corresponda a su aplicación, se integra fácilmente con otros productos y tiene características que se pueden actualizar fácilmente en el futuro. [8]

- **Control de Movimiento Kinetix.** La solución de control de movimiento integrado Kinetix ofrece una integración transparente de los controladores Logix de Allen-Bradley, SERCOS interface, módulos digitales de control de movimiento, servo-variadores Allen-Bradley, motores y accionadores, lo que convierte el movimiento integrado en el nuevo estándar para el control de máquinas. [9]

  Los variadores Kinetix están optimizados para módulos de movimiento digital ControlLogix, CompactLogix, GuardLogix y SoftLogix de SERCOS interface. El sistema de comunicaciones serial en tiempo real (SERCOS, Serial Real-Time Communications System) es una interface de controladores/variadores que emplea cables de fibra óptica inmunes al ruido. El resultado es un control de movimiento dotado de la técnica más moderna con las siguientes ventajas:

  - Diagnósticos avanzados y generación de informes de proceso mediante SERCOS interface. [8]
  - Amplia variedad de opciones de módulos de movimiento.

- Un anillo de fibra óptica sirve como única interface entre la aplicación de control y el variador. Sustituye a costosos cableados de comando y realimentación, lo que reduce tanto los costes de tiempo de instalación como de cableado.
- Hasta 16 ejes de movimiento que se pueden controlar desde un módulo de movimiento ControlLogix o SoftLogix Un módulo de movimiento CompactLogix puede controlar hasta 4 ejes.
- Los sistemas ControlLogix y SoftLogix son totalmente expandibles, con hasta 32 ejes admisibles por controlador. Si se necesitan más ejes, se pueden utilizar varios controladores. [8] (Ver Anexo C)

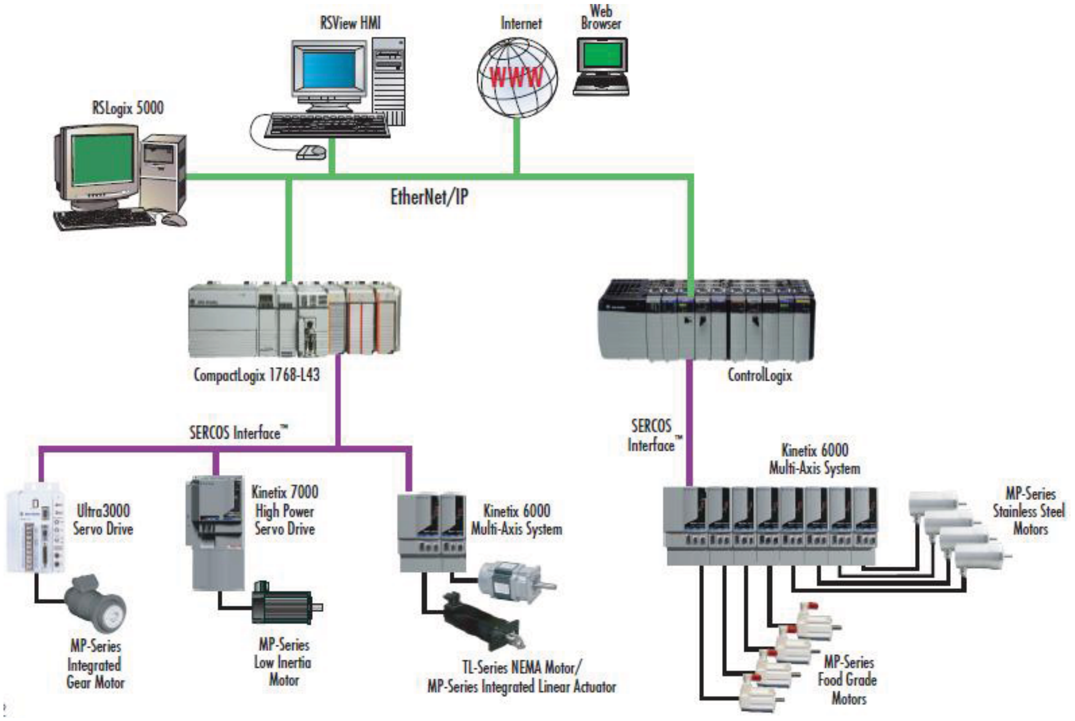


Figura 10. Arquitectura Kinetix [9]

2. DESCRIPCION DE HARDWARE Y SOFTWARE

2.1. DESCRIPCION DEL HARDWARE

2.1.1. Procesador Logix 5562 (1756-L62)

El ControlLogix 5562 es un controlador con una capacidad de 4 MB para procesar datos y lógica de memoria no volátil. Puede comunicarse por las diferentes opciones de comunicación del NetLinx y también tiene la opción de comunicarse por un puerto serial RS-232. A su vez, guarda información de programas a través de una memoria CompactFlash industrial, permitiendo tener un Backup de proyectos que estén descargados en el controlador. El controlador se puede instalar en cualquier ranura del chasis; en este caso en particular el procesador se encuentra en la primera ranura. [10]

Figura 11. ControlLogix 5562. [10]

El procesador cuenta adicionalmente en su panel frontal con unos led´s indicadores, que permiten informar al operario el estado del procesador. (Ver Figura 12)

A su vez la tabla 2 presenta la descripción de cada una de estas indicaciones y en la tabla 3 se muestran las características técnicas de este controlador especifico.

INDICADOR	ESTADO	DESCRIPCION
RUN	Apagado	El controlador está en modo Programación o prueba.
	Verde Fijo	El controlador está en modo Marcha
I/O	Apagado	No hay dispositivos en la configuración de E/S del controlador o el controlador no contiene un proyecto
	Verde Fijo	El controlador se está comunicando con todos los dispositivos es su configuración de E/S.
	Verde Intermitente	Uno o más dispositivos en la configuración de E/S del controlador no responden.
	Rojo Intermitente	El chasis está defectuoso. Reemplazar chasis.
FORCE	Apagado	No hay *tags* que contengan valores forzados.
	Ámbar fijo	Los forzados de E/S están activos. (habilitados)
	Ámbar Intermitente	Una o más direcciones de entrada o salida han sido forzadas al estado activado o desactivado.
RS232	apagado	No hay actividad.
	Verde Fijo	Recibiendo o transmitiendo datos.

Tabla 2. Indicadores de estado del Controlador [10]

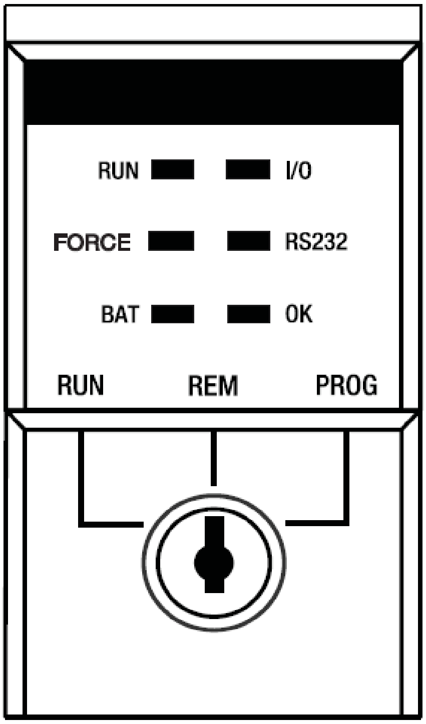


Figura 12. Panel Frontal del Controlador. [10]

Características del Controladores Logix 5562	
Tareas del controlador: ● Continua ● Periódica ● Evento	- 32 tareas (sólo 1 continua) - tareas de eventos: acepta todos los activadores de evento.
Memoria de usuario	4 Mbytes
Memoria E/S	478 Kbytes
Memoria de usuario no volátil	CompactFlash
E/S Digitales, máx.	128000
E/S Análogas, máx.	4000
Puertos de comunicación incorporados	1 puerto RS- 232 en serie (DF1 ó ASCII)
Opciones de comunicación (estas opciones tienen productos y perfiles específicos para su plataforma - otras opciones están disponibles a través de productos de otros fabricantes y perfiles genéricos)	- EtherNet/IP - ControlNet - DeviceNet - Data Highway Plus - E/S remotas universales en serie - Modbus mediante rutina de lógica de escalera - DH-485 - SynchLink
Redundancia del controlador	soporte total de redundancia
E/S nativas	E/S 1756 ControlLogix
Movimiento simple	motores paso a paso servo mediante DeviceNet variador de CA analógico
Movimiento integrado	SERCOS Interface interface analógica interface hidráulica interface SSI
Opciones de montaje y/o instalación	Chasis 1756
Lenguajes de programación	- lógica de escalera de relés - texto estructurado - bloques de funciones - diagrama de función secuencial
Disipación de energía, máx.	3.5 W
Disipación térmica, máx.	11.9 BTU/hr
Corriente del backplane (mA) a 5 V	1.20 A
Corriente del backplane (mA) a 24 V	0.014 A

Tabla 3. Características del Controladores Logix 5562 [11]

2.1.2. Modulo interface de 3 ejes SERCOS (1756-M03SE)

Los servomódulos de SERCOS interface actúan como un vínculo entre la plataforma ControlLogix y servovariadores inteligentes. SERCOS representa el protocolo IEC 61491 Serial Real-time Communication System (sistema de comunicación serial en tiempo real) a través de un medio de fibra óptica. La interface SERCOS, es una interface abierta de controlador a variador digital diseñada para la comunicación en serie, en tiempo real y de alta velocidad que permite enviar información detallada de estado del variador desde el variador al controlador y viceversa. [11]

Figura 13. Módulo SERCOS. [11]

Especificaciones Técnicas SERCOS Interface	
Atributos	1756-M03SE
Número Max. de Ejes, por Módulo	3
Número Máx. de Ejes, por Controlador	32
Velocidad de Datos	4 Mbps 8 Mbps
Tiempo de Ciclo SERCOS 4 Mbps	0.5 ms (solo con drives Kinetix 6000), hasta 2 drives 1 ms, hasta 4 drives 2 ms, hasta 8 drives
Tiempo de Ciclo SERCOS 8 Mbps	0.5 ms (solo con drives Kinetix 6000), hasta 4 drives 1 ms, hasta 8 drives 2 ms, hasta 16 drives
Modos de Control	Posición, Velocidad y Torque
Corriente del Backplane (mA) a 5 V	760 mA
Corriente del Backplane (mA) a 24 V	2.5 mA
Disipación de Energía	5.0 W
Localización del Modulo	Basada en chasis, cualquier slot
Chasis	1756-A4, 1756-A7, 1756-A10, 1756-A13, 1756-A17
Cables de Fibra Óptica	- 2090-SCEPxx-0 sin chaqueta, polietileno clorado - 2090-SCVPxx-0 chaqueta estándar, cloruro de polivinilo - 2090-SCNPxx-0 chaqueta de nylon

Tabla 4. Especificaciones Técnicas Modulo SERCOS [12]

Los módulos son compatibles con el conjunto de instrucciones de movimiento RSLogix 5000 (38 instrucciones de movimiento) y las utilidades de configuración de ejes. Las instrucciones de movimiento proporcionan una amplia gama de posibilidades de movimiento, incluido el posicionamiento punto-punto, sincronismo digital, levas de posicionamiento y sincronización, y el movimiento circular y lineal multieje. [11]

Indicadores del Módulo SERCOS. El panel frontal del módulo SERCOS posee unos indicadores LED, que muestran al operador el estado del módulo (comunicación y conexión). La tabla 5 muestra los estados que presenta el indicador *CP* (indicador de estado de fase de comunicación) y su respectiva descripción para su correcto entendimiento; de igual manera la tabla 6 y 7 presenta el estado, descripción y acción recomendada del indicador de estado de los anillos SERCOS y del indicador *OK* (indicador de estado de comunicación y funcionamiento del módulo) respectivamente.

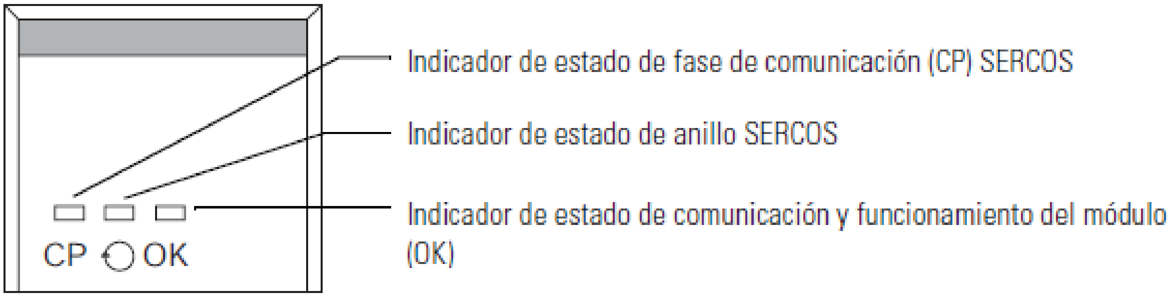


Figura 14. Panel Frontal del Módulo SERCOS. [13]

Los módulos SERCOS interface pueden trabajar con los siguientes Servodrive:

- 2093 Kinetix 2000 Multi-Axis Servo Drive
- 2094 Kinetix 6000 Multi-Axis Servo Drive
- 2099 Kinetix 7000 High-Power Servo Drive
- 2098 Ultra3000 SERCOS Servo Drive. [12]

Estado	Descripción
Anaranjado fijo	Fase 1: Se está efectuando una detección automática de baudios.
Apagado	Fase 0: Se está buscando un anillo cerrado.
Rojo parpadeante	Fase 1: Se están buscando nodos activos.
Rojo/verde alternante	Fase 2: Se están configurando nodos para comunicación.
Verde parpadeante	Fase 3: Se están configurando parámetros específicos del dispositivo
Verde fijo	Fase 4: Sistema configurado y activo.

Tabla 5. Indicador de Estado de Fase de Comunicación (CP) SERCOS [13]

Estado	Descripción	Acción recomendada
Verde fijo	El anillo, el variador y los ejes están configurados y comunicándose activamente con los nodos en el anillo	Ninguna
Rojo parpadeante	El módulo ha detectado un fallo de instalación o configuración en el anillo.	Compruebe la instalación y configuración del sistema tal y como se indica a continuación:. •Asegúrese de que las direcciones del variador y de los ejes son correctas. •Retire los ejes sobrantes del anillo. •Asegúrese de que el programa de aplicación haya seleccionado el período de ciclo del anillo y la velocidad en baudios correctos.
Rojo fijo	El módulo ha detectado un fallo de hardware o de instalación en el anillo	Compruebe la instalación y el hardware del sistema tal y como se indica a continuación: •Asegúrese de que todos los cables estén instalados correctamente. •Asegúrese de que el cable es del tipo y longitud correctos. •Asegúrese de que el programa de aplicación haya configurado el nivel de transmisión del anillo del módulo como alto al usar los cables especificados. •Asegúrese de que los niveles de transmisión del variador estén configurados correctamente. •Inspeccione los cables para determinar si presentan degradación. •Inspeccione los variadores en busca de fallos y, de encontrarlos, corríjalos
Apagado	El módulo no ha detectado datos de anillo en su receptor o bien no ha conseguido completar debidamente la fase 2	Compruebe el sistema y la instalación tal y como se indica a continuación: •Asegúrese de que todos los cables estén instalados correctamente. •Inspeccione el cable para determinar ni presenta degradación o rotura. •Compruebe si hay fallos en los variadores
Verde parpadeante	El anillo, el variador o los ejes no están configurados, pero por lo menos uno ha sido identificado	No supone un problema, si no se ha configurado el sistema. Si tiene problemas para configurar el anillo, el variador y los ejes, asegúrese de que el proyecto RSLogix 5000 esté correctamente configurado para el equipo en uso

Tabla 6. Indicador de Estado de Anillo SERCOS [13]

Estado	Descripción	Acción recomendada
Apagado	El módulo no funciona.	•Conecte la alimentación eléctrica al chasis. •Compruebe que el módulo esté completamente insertado en el chasis y el Backplane.
Verde parpadeante	El módulo ha pasado los diagnósticos internos, pero no ha establecido comunicaciones activas.	Ninguna, si no ha configurado el módulo.
Verde fijo	•Se están intercambiando datos. •El módulo está en el estado operativo normal.	Ninguna. El módulo está listo para entrar en acción.
Rojo parpadeante	•Se ha producido un fallo mayor recuperable. •Se está efectuando una actualización NVS.	Si se está efectuando una actualización NVS, conclúyala. Si no se está efectuando una actualización NVS: Reinicialice el módulo
Rojo fijo	Es posible que se haya producido un fallo irrecuperable.	•Reinicialice el módulo. •Si se mantiene la luz roja fija, reemplace el módulo.

Tabla 7. Indicador Estado de Comunicación y Funcionamiento Módulo (OK). [13]

2.1.3. Servo Drive Digital ULTRA 3000 (2098-DSD-005-SE)

Figura 15. Servo Drive Ultra 3000 [14]

El Ultra3000 es un servovariador digital de alto rendimiento que ofrece flexibilidad en aplicaciones de uno o varios ejes. El Ultra3000 es compatible con una gran variedad de motores Allen-Bradley que ofrece par continuo desde menos de un Nm hasta 700 Nm, para aplicaciones con requisitos de potencia de 5 kW - 22 kW. [15]

Ventajas del servovariador digital Ultra3000:

- El Ultra3000 proporciona una integración simple a las arquitecturas de control de máquinas de Allen-Bradley. Acepta fuentes analógicas de comando de seguimiento maestro y paso/dirección de 0 – 10 V, además de conectividad SERCOS y DeviceNet en toda la familia Ultra3000.

- Hay combinaciones optimizadas de variador/motor para cada aplicación. El Ultra3000 puede hacer funcionar a una amplia variedad de servomotores sin escobillas, incluyendo los motores de la serie Y-, LD-, N-, H-, F-, W- y MP- de Allen-Bradley, además de motores lineales y de otros fabricantes.

- Servicio técnico incorporado para encoders absolutos de múltiples vueltas para eliminar costosos y arduos ciclos a la posición inicialde la máquina, y adicionalmente ofrece la opción de suministrar alimentación de lógica externa para mantener la posición durante pérdidas de energía.

- Igual que los demás servovariadores de movimiento integrado Kinetix, el Ultra3000 emplea software de programación RSLogix 5000 para la integración completa de aplicaciones de control discretas y de movimiento, lo que elimina la necesidad de contar con múltiples herramientas de programación. RSLogix 5000 también proporciona potentes herramientas de puesta en servicio y de diagnóstico para facilitar la instalación, la operación y el mantenimiento de la máquina.

- Cuenta adicionalmente con el software Ultraware, el cual es una poderosa herramienta de puesta en marcha y diagnóstico, diseñada para aumentar la productividad y permitir optimizar el rendimiento del sistema rápida y fácilmente.

- Emplea tecnología Smart Motor para proporcionar identificación automática de la correcta conectividad motor-variador, lo que reduce el tiempo de puesta en servicio. [6][14]

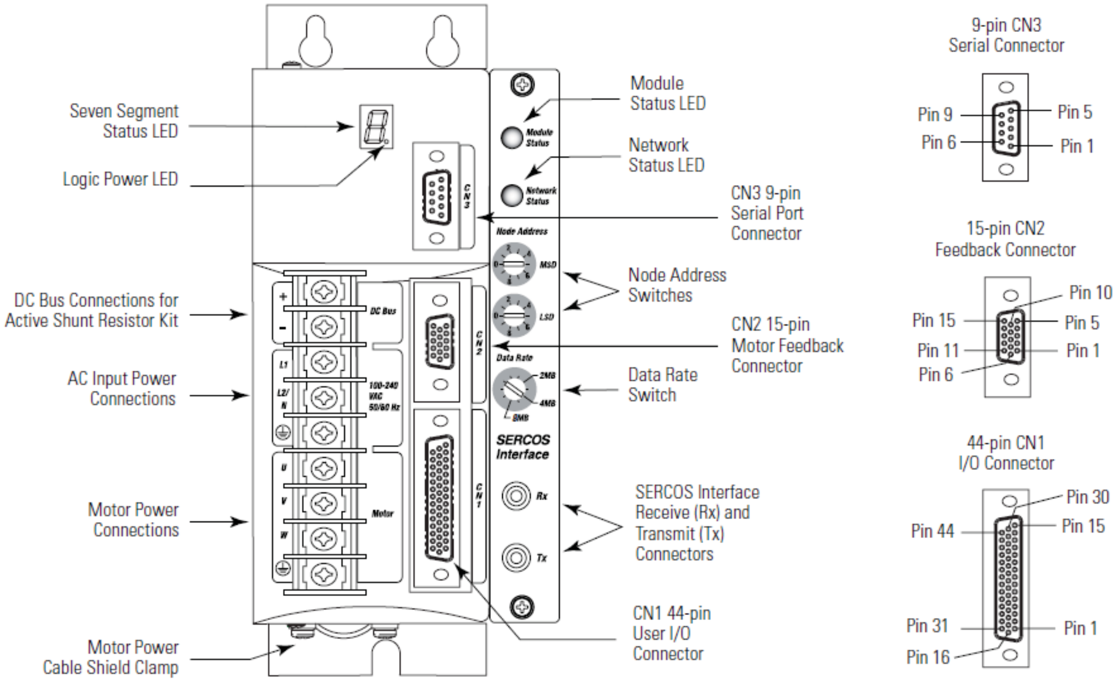


Figura 16. Descripcion de Conexiones Panel Frontal del 2098-DSD-005-SE. [16]

El servovariador Ultra 3000 2098-DSD-005-SE es un variador de 0.5kW con interface de comunicación SERCOS y alimentación a 230V; las características técnicas de este servo drive se recopilan en el Anexo D. La descripción de las señales de los puertos CN1 (User I/O), CN2 (Motor Feedback) y CN3 (Serial Port) se presentan claramente en el Anexo E páginas 38 y 39.

La conexión entre el Ultra 3000 SE y el módulo SERCOS se realiza a través de fibra óptica y utilizando una topología de anillo (ver figura 17). Para la correcta puesta en marcha de una red en anillo se debe configurar la dirección de nodo (posición) de cada servo drive dentro de la red, esto se realiza manualmente a través de los *Node Address Switches* ubicados en el panel frontal de cada Ultra 3000; se debe configurar también en cada servo drive la velocidad de transmisión de manera manual en el panel frontal del dispositivo a través del *Data Rate Switch,* la dirección de cada servo drive debe ser la misma. Para comprender mejor esta configuración remitirse al Anexo F páginas 70 y 71.

Con el fin de identificar es estado del módulo, la conexión y detectar errores del servo drive, revisar las páginas 117 a 122 del Anexo E para identificar a que corresponde el código de error mostrado en el display siete segmentos y su posible solución y revisar la página 123 del mismo Anexo para comprender el SERCOS Module Status LED y el SERCOS Network Status LED.

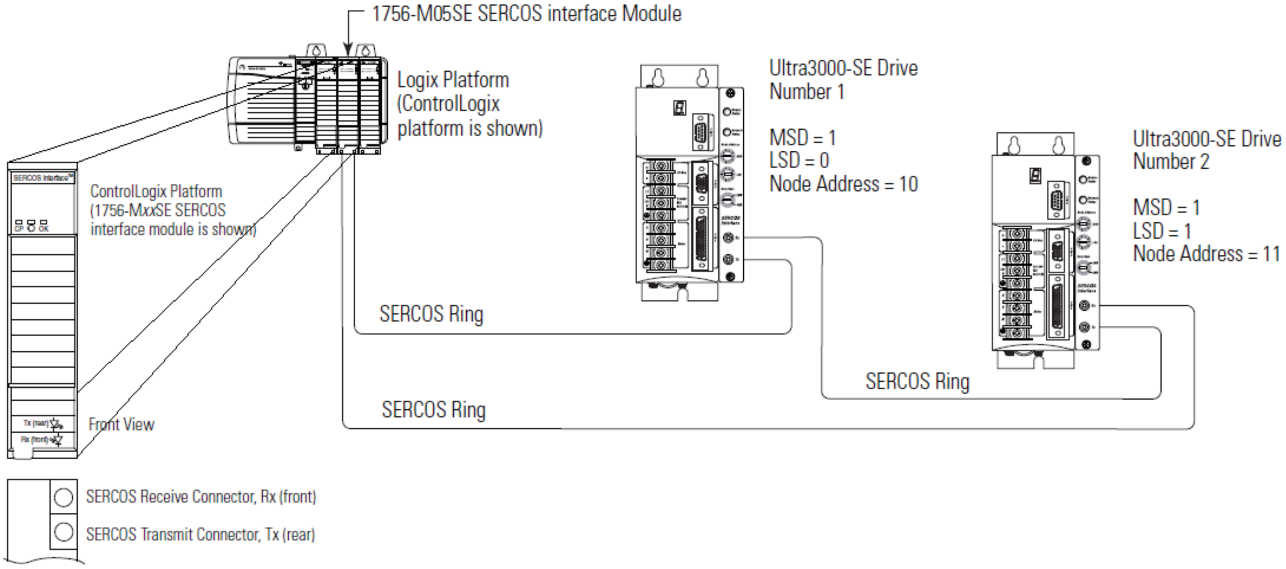


Figura 17. Conexión y Direccionamiento de los Servodrives. [17]

La figura 18 presenta la conexión entre el Software RSLogix 5000, el Controlador ControlLogix, los Servo drives Ultra 3000 y los motores.

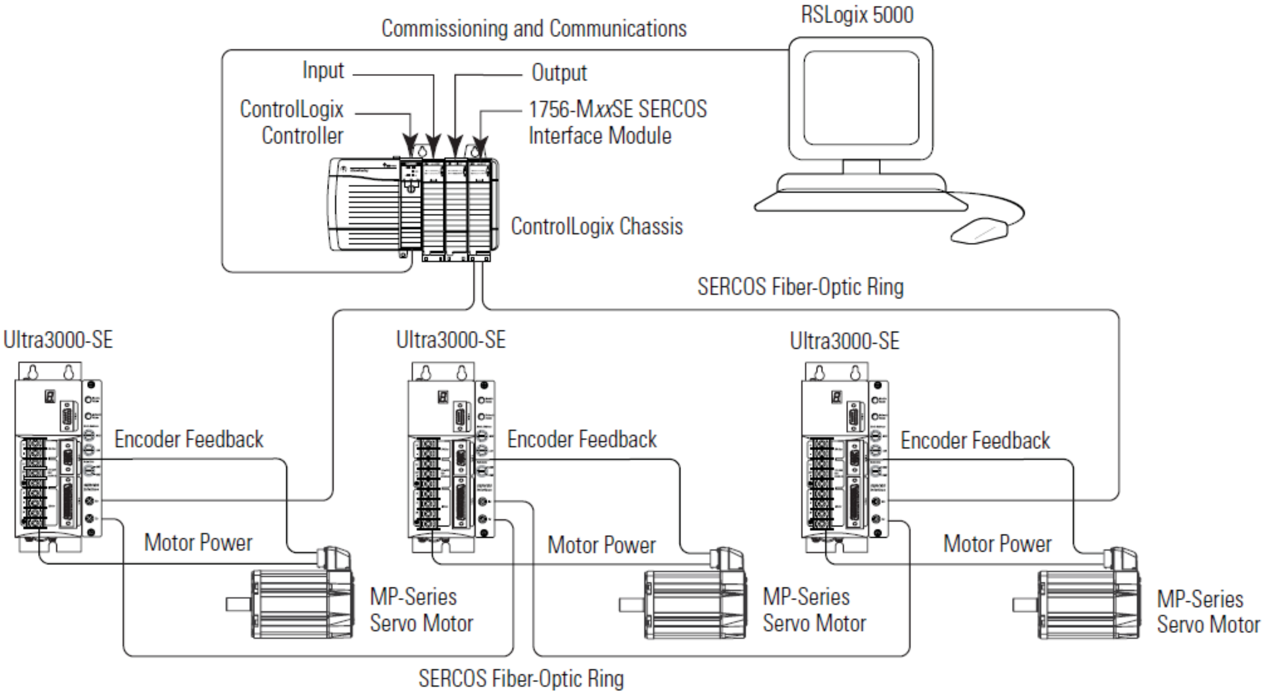


Figura 18. Estructura de Conexión Ultra 3000 SE [16]

2.1.4. Servomotor (MPL-A1510V-EJ42AA)

Figura 19. Servomotores de la Serie MP de Baja Inercia.

La serie MPL ofrece un nuevo diseño que reduce el tamaño del motor y proporciona un par significativamente más alto. Un diseño de estator de núcleo segmentado, administración térmica mejorada así como múltiples configuraciones electrónicas disponibles crean un motor que ofrece óptimas funciones de rendimiento. Esta serie de motores es típicamente usada con las familias de Servo Drives Kinetix 2000, Kinetix 6000, Kinetix 7000 y Ultra 3000 de Allen-Bradley. Esta serie tiene disponibles nueve tamaños de estructura y en crecimiento, los motores de la serie MPL ofrecen capacidad de par continuo de 0.26 a 163 Nm (2.3 a 1440 lb-pulg.) y un torque pico 0.77 a 278 Nm (6.8 a 2460 lb-pulg.). Las opciones de retroalimentación del motor de serie MPL incluyen un encoder de 2,000 líneas de alto rendimiento que produce 8,000 conteos por revolución para lograr una retroalimentación de posición precisa. [14][18]

Max Speed – rpm	8000
Continuous Stall Torque - Nm (lb-in.)	0.26 (2.3)
Peak Stall Torque - Nm (lb-in.)	0.77 (6.8)
Motor Rated Output - kW	0.16
Continuous Stall Current Amperes (0-peak)	1.05
Peak Stall Current Amperes (0-peak)	3.4
Rotor Inertia - kg-m2 (lb-in-s2)	0.000006 (0.000053)
Motor Weight, Approx - kg (lb)	1.0 (2.2)
Voltage Rating – V	230

Tabla 8. Especificaciones Servomotor MPL-A1510V [14][18]

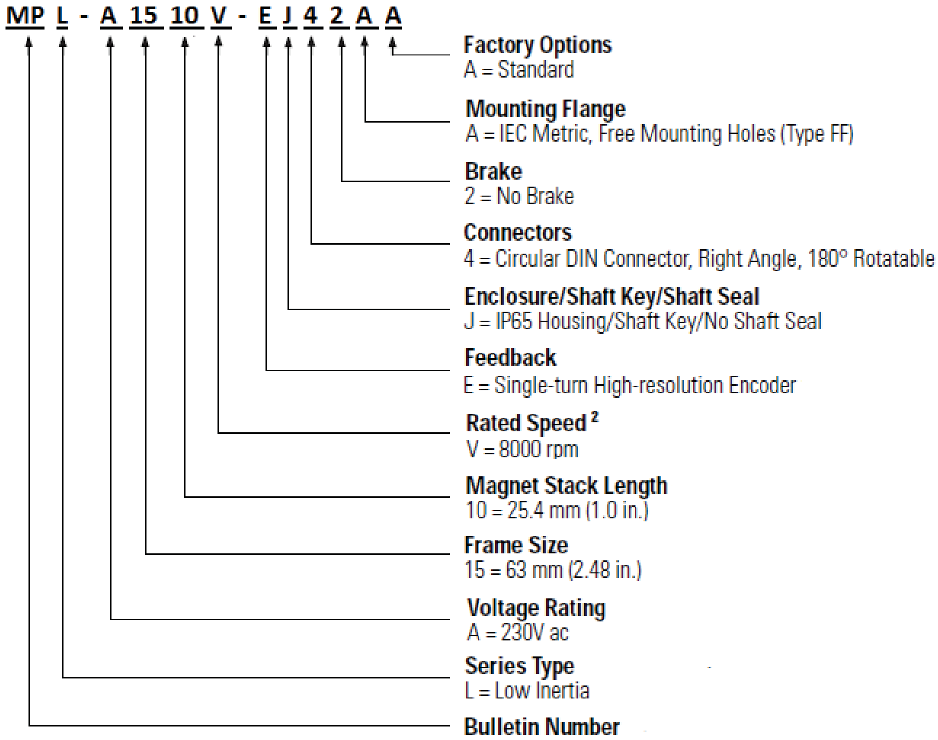


Figura 20. Descripción del Número de Catálogo del Servomotor [18]

Las dimensiones del servomotor se pueden ver el Anexo G páginas 18 y 19, a su vez las referencias de las cargas radiales y axiales en las páginas 27-29.

2.1.5. Filtro RFI para Servo drives ULTRA (2090-UXLF-106)

Figura 21. Filtro RFI [19]

Los dispositivos de referencia 2090-UXLF-106 son filtros monofásicos AC que se utilizan con los servo drives para eliminar cualquier ruido que pueda presentarse en la red eléctrica, y dar confiabilidad en el funcionamiento de los servomotores y de todo el sistema integrado de movimiento. [19]

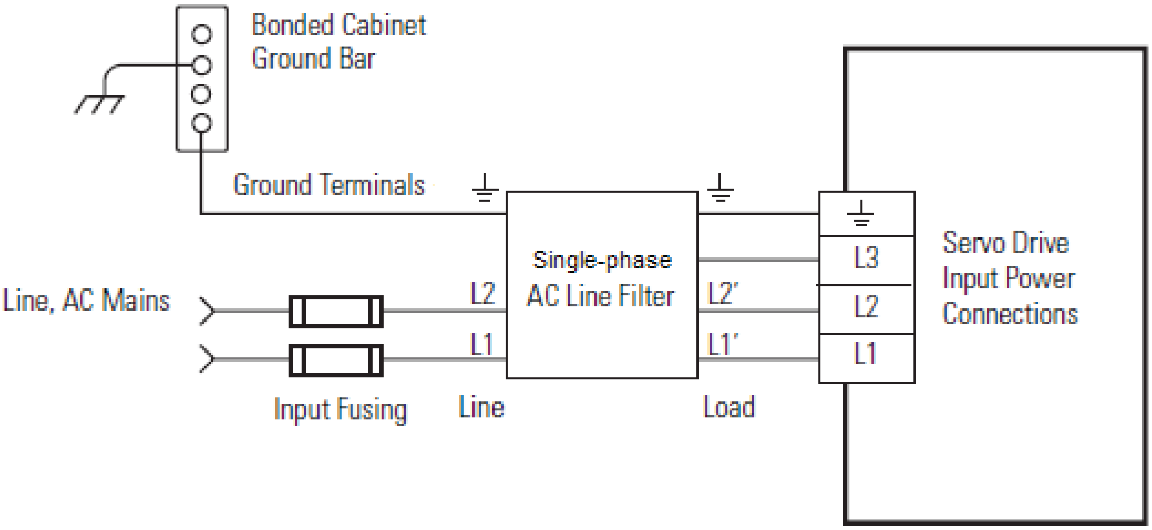


Figura 22. Conexión del filtro RFI. [19]

La figura 22 presenta como hacer la conexión del filtro AC. Para una mayor información técnica de este filtro. (Ver Anexo H)

2.1.6. Cable de Poder para el Servomotor (2090-XXNPMF-16S03)

Cable de poder estándar para la familia de servomotores MP, en la tabla 9 se presentan las especificaciones del cable. (Ver Anexo I)

Tipo de cable / Color de la Chaqueta	Descripción	Tamaño del cable AWG	Peso aproximado kg/m (lb/ft)	Longitudes estándar de cables
Cable estándar industrial TPE, Negro	Cuatro conductores, 600V, cable blindado para tres fases de poder con otros cuatro conductores, 18 AWG, blindado, para el motor frenos y piezas de repuesto.	16	0.276 (0.186)	1 (3.2) 2 (6.5) 3 (9.8) 4 (13.1) 5 (16.4) 7 (22.9) 9 (29.5) 12 (39.4) 15 (49.2) 20 (65.6) 25 (82.0) 30 (98.4) 40 (131.2) 60 (196.8) 90 (295.3)

Tabla 9. Especificaciones del Cable de Poder. [25]

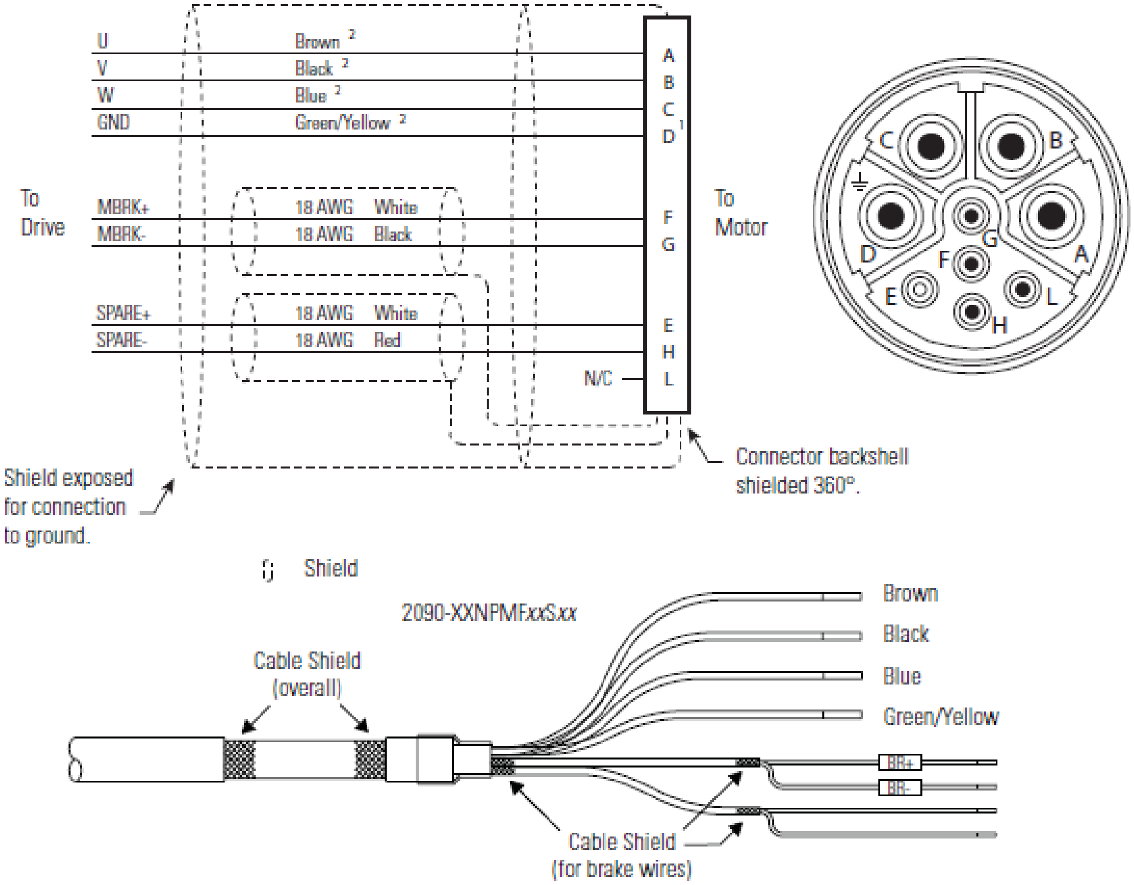


Figura 23. Identificación de los terminales del Cable de Poder [16][20]

2.1.7. Cable Feedback para el Servomotor (2090-XXNFMF-S03)

Cable Feedback estándar para la familia de servomotores MP, RDD, HPK, LDC y LDL, en la tabla 10 se presentan las especificaciones del cable. (Ver Anexo I)

Tipo de cable / Color de la chaqueta	Descripción	Tamaño del cable AWG	Peso aproximado kg/m (lb/ft)	Longitudes estándar de cables
Cable estándar industrial TPE, Negro	Conector DIN roscado (para el motor) a conductores libres (para drive), 30 V	28 de Feedback 16 de energía, 5V 22 de energía, 9V	0.120 (1.35)	1 (3.2) 2 (6.5) 3 (9.8) 4 (13.1) 5 (16.4) 7 (22.9) 9 (29.5) 12 (39.4) 15 (49.2) 20 (65.6) 25 (82.0) 30 (98.4) 40 (131.2) 60 (196.8) 90 (295.3)

Tabla 10. Especificaciones del cable Feedback. [25]

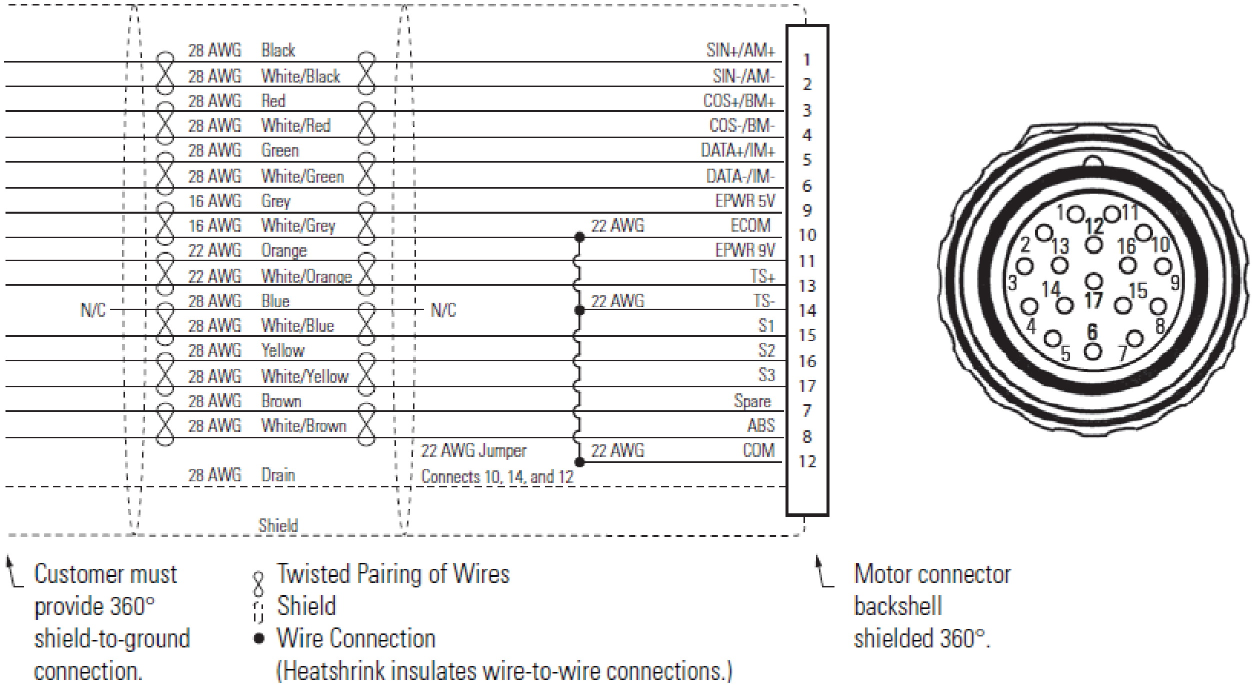


Figura 24. Identificación de los Terminales del Cable Feedback [21]

2.1.8. Breakout Board CN2 (2090-UXBB-DM15)

Adaptador para conexión del cable Feedback del servomotor con el conector CN2 del servo drive Ultra 3000 y 5000.

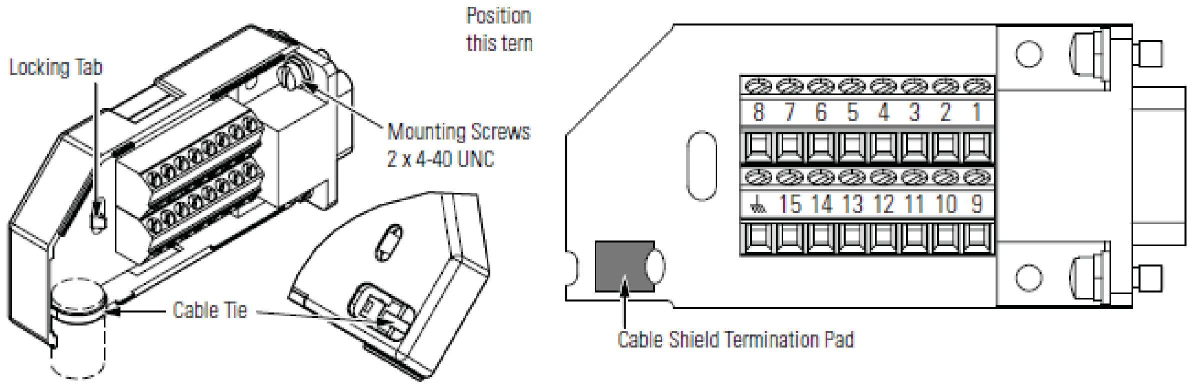


Figura 25. Breakout Board CN2. [22]

En la siguiente figura se especifica la conexión que se debe realizar entre el cable y el adaptador. (Ver Figura 26)

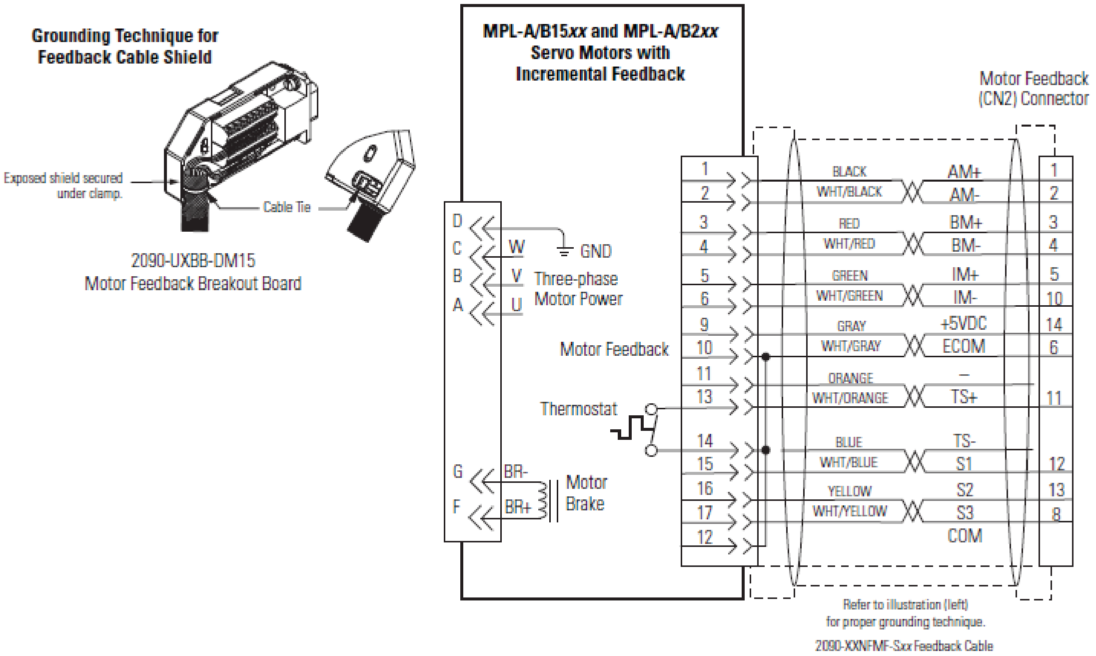
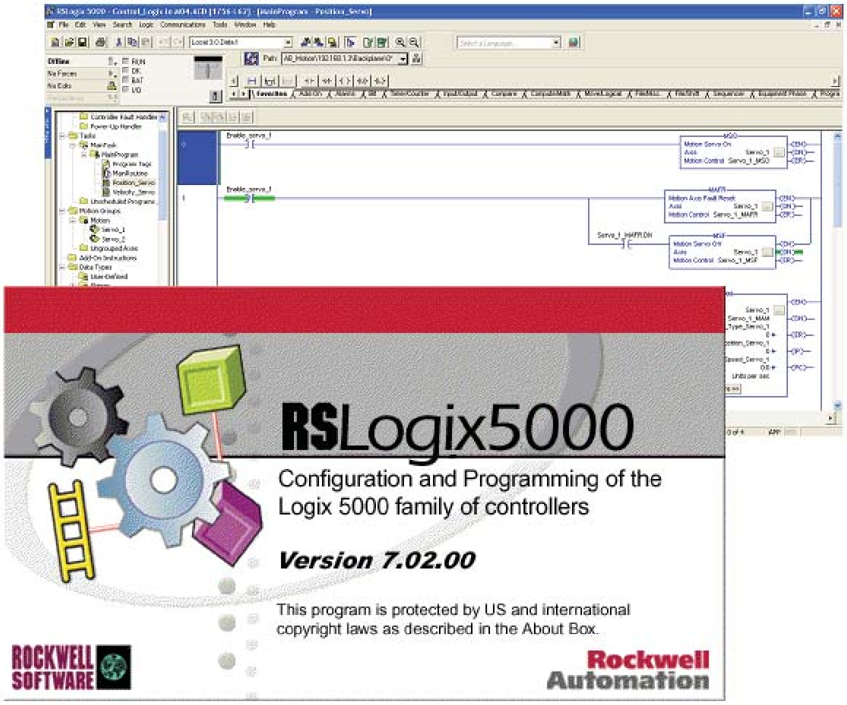


Figura 26. Conexión del Cable Feedback al conector CN2 [17]

2.2. DESCRIPCION DEL SOFTWARE

2.2.1. RSLogix 5000



RSLogix5000

Configuration and Programming of the Logix 5000 family of controllers

Version 7.02.00

This program is protected by US and international copyright laws as described in the About Box.

Figura 27. Software de Programación RSLogix 5000

RSLogix 5000 es una plataforma de software que permite configurar, programar y supervisar el funcionamiento de los PAC’s Logix como el ControlLogix y el CompactLogix. Proporciona la lógica de escalera, texto estructurado, diagrama de bloques de función secuencial y editores de diagrama de funciones para el desarrollo del programa, así como el apoyo para el modelo de equipos, estado de fase de lotes y control de maquinaria. [26]

Este software presenta las siguientes características: [8]

- Puede utilizarse para aplicaciones de base discreta, de proceso, de lote, de movimiento, de seguridad y de variadores.
- Es compatible con la familia escalable de controladores programables de automatización (PAC) Logix.
- Permite fragmentar la aplicación en programas más pequeños que pueden volver a utilizarse, rutinas e instrucciones que pueden crearse al utilizar distintos lenguajes de programación: diagrama de lógica de escalera, diagrama de bloque de funciones, texto estructurado y diagrama de funciones secuenciales.
- Incluye un conjunto extenso de instrucciones incorporadas que usted puede aumentar al crear sus propias instrucciones add-on definidas por el usuario.
- Permite escribir la aplicación sin tener que preocuparse de la configuración de la memoria.
- Proporciona la capacidad de crear tipos de datos definidos por el usuario para representar fácilmente componentes específicos de la aplicación en una estructura.
- Incorpora datos y los comparte con otros productos de software de Rockwell Automation para reducir drásticamente el tiempo de entrada de datos, proporcionar auditorías y facilitar el manejo de códigos y su uso repetido.
- La codificación de documentación es más simple y rápida.
- Permite importar y exportar la totalidad del proyecto, o componentes y secciones de código, para editarlos con herramientas de otros fabricantes y para poder compartirlos fácilmente.
- Permite distintos niveles de seguridad y protección de propiedad intelectual.
- Permite realizar fácilmente la depuración y el mantenimiento de la aplicación con forzado de E/S, edición de tiempo de ejecución, adición de tiempo de ejecución de módulos E/S seleccionados y tendencias.
- Simplifica el mantenimiento porque siempre se puede obtener la fuente original en el lenguaje de programación en el que fue escrita desde el controlador.

- La configuración de datos y el acceso a ellos en los módulos E/S es simple gracias a cuadros de diálogos fáciles de usar y estructuras de datos predefinidas.
- Permite realizar actualizaciones manuales y automáticas firmware de módulos.

2.2.2. Instrucciones de Control de Movimiento de Controladores Logix5000

El software RSLogix 5000 proporciona la capacidad de control de movimiento incorporada y altamente integrada. Una sola plataforma de hardware y software satisface sus necesidades combinadas de controlador programable y control de movimiento.

RSLogix 5000 posee un amplio número de instrucciones de control de movimiento incorporadas, fáciles de usar, incluye también interpolación circular y lineal multi-eje y transformaciones robóticas Kinematics, que hace posible el fácil manejo de un gran espectro de aplicaciones de movimiento. Puede crear y monitorear rápidamente perfiles de movimiento y CAMs dinámicamente complejos con editor gráfico de perfil de movimiento incorporado a RSLogix 5000. [9]

Las instrucciones de control de movimiento utilizan tres tipos de secuencias de temporización que son:

- **Inmediata.** Esta instrucción se completa en un scan.
- **Mensaje.** Esta instrucción se completa con varios scanes porque la instrucción envía mensajes al servo módulo.
- **Proceso.** Completar la instrucción puede llevar una cantidad indefinida de tiempo.

A continuación se explican de forma más detallada:

- **Secuencia de Tipo Inmediato.** Las instrucciones de tipo inmediato de control de movimiento se ejecutan para la finalización en un scan. Si el controlador detecta un error durante la ejecución de estas instrucciones, se establece un bit de estado de error y finaliza la operación.

  Los ejemplos de instrucciones de tipo inmediato incluyen:

  - Instrucción Motion Change Dynamics (MCD).
  - Instrucción Motion Group Strobe Position (MGSP).

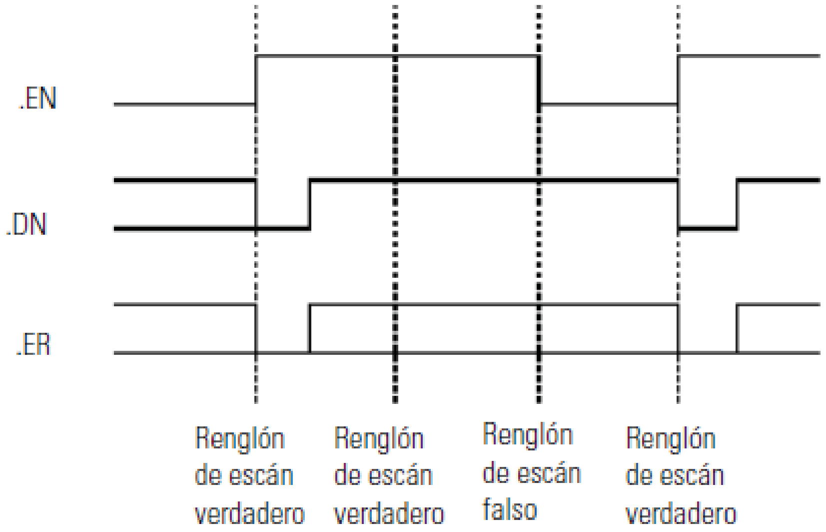


Figura 28. Condiciones del Renglón Instrucción de Tipo Inmediato. [23]

Las instrucciones inmediatas funcionan de la siguiente manera:

1. Cuando el renglón que contiene la instrucción de control de movimiento se vuelve verdadero, el controlador:

- Establece el bit Habilitar (.EN).
- Borra el bit Listo (.DN).
- Borra el bit Error (.ER).

El controlador ejecuta la instrucción por completo.

2.

Si el controlador	Entonces
No detecta un error cuando se ejecuta la instrucción	El controlador establece el bit .DN.
Detecta un error cuando se ejecuta la instrucción	El controlador establece el bit .ER y almacena un código de error en la estructura de control.

3. La próxima vez que el renglón se vuelva falso una vez establecido el bit .DN o el .ER, el controlador borra el bit .EN.

4. El controlador puede volver a ejecutar la instrucción cuando el renglón se vuelve verdadero.

- **Secuencia de Tipo Mensaje.** Las instrucciones de control de movimiento de tipo mensaje envían uno o más mensajes al servo módulo.

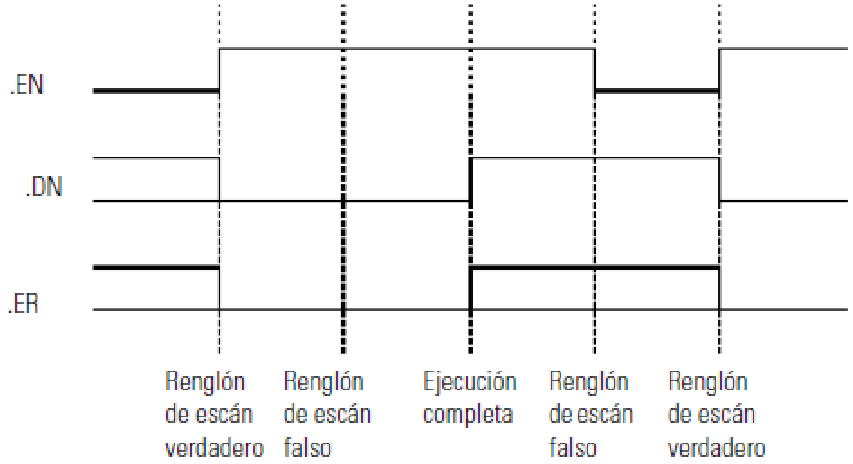


Figura 29. Condiciones del renglón instrucción de tipo mensaje. [23]

Los ejemplos de instrucciones de tipo mensaje incluyen la:

- Instrucción Motion Direct Drive On (MDO).
- Instrucción Motion Redefine Position (MRP).

Las instrucciones de tipo mensaje funcionan de la siguiente manera:

1. Cuando el renglón que contiene la instrucción de control de movimiento se vuelve verdadero, el controlador:

- Establece el bit Habilitar (.EN).
- Borra el bit Listo (.DN).
- Borra el bit Error (.ER).

2. El controlador comienza a ejecutar la instrucción configurando una solicitud de mensaje al servo módulo.

El resto de la instrucción se ejecuta en paralelo con el scan del programa.

3. El controlador verifica si el servo módulo está listo para recibir un nuevo mensaje.

4. El controlador coloca los resultados de la verificación en la palabra de estado de mensaje de la estructura de control.

5. Cuando el módulo está listo, el controlador construye y transmite el mensaje al módulo.

Este proceso se puede repetir varias veces si la instrucción requiere de múltiples mensajes.

6.

Si el controlador	Entonces
No detecta un error cuando se ejecuta la instrucción	El controlador establece el bit .DN si se han enviado todos los mensajes al módulo.
Detecta un error cuando se ejecuta la instrucción	El controlador establece el bit .ER y almacena un código de error en la estructura de control.

7. La próxima vez que el renglón se vuelve falso, una vez establecido el bit .DN o .ER, el controlador borra el bit .EN.

8. El controlador puede ejecutar la instrucción nuevamente cuando el renglón se vuelve verdadero.

- **Secuencia de Tipo Proceso.** Las instrucciones de control de movimiento de tipo proceso inician procesos de control de movimiento que pueden tardar un tiempo indefinido para completarse.

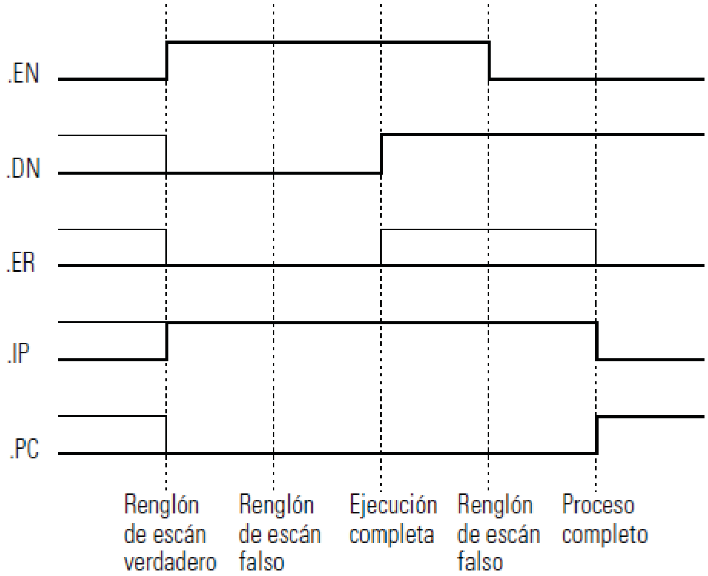


Figura 30. Condiciones del renglón instrucción de tipo proceso. [23]

Los ejemplos de instrucciones de tipo proceso incluyen la:

- Instrucción Motion Arm Watch Position (MAW).
- Instrucción Motion Axis Move (MAM).

Las instrucciones de tipo proceso funcionan de la siguiente manera:

1. Cuando el renglón que contiene la instrucción de control de movimiento se vuelve verdadero, el controlador:

- Establece el bit Habilitar (.EN).
- Borra el bit Listo (.DN).
- Borra el bit Error (.ER).
- Borra el bit Proceso Completo (.PC).
- Establece el bit En Proceso (.IP).

2. El controlador inicia el proceso de movimiento.

3.

Si	Entonces el controlador
El controlador no detecta un error cuando se ejecuta la instrucción	· Establece el bit .DN. · Establece el bit En Proceso (.IP).
El controlador detecta un error cuando se ejecuta la instrucción	· Establece el bit .ER. · Almacena un código de error en la estructura de control.
El controlador detecta otra ocurrencia de la instrucción de control de movimiento	Borra el bit .IP para esa ocurrencia.
El proceso de control de movimiento llega al punto donde la instrucción puede volver a ejecutarse	Establece el bit .DN. Para algunas instrucciones de tipo proceso, tales como MAM, esto ocurrirá durante el primer escán. Para otras instrucciones, tales como MAH, el bit .DN no se establecerá hasta que se complete todo el proceso de vuelta a la posición inicial.
Ocurre uno de los eventos siguientes durante el proceso de control de movimiento: • Se completa el proceso de movimiento • Se ejecuta otra ocurrencia de la instrucción • Otra instrucción detiene el proceso de movimiento • Un fallo de movimiento detiene el proceso de movimiento	Borra el bit .IP.

4. Luego del inicio del proceso de movimiento, el scan del programa puede continuar.

El resto de la instrucción y el proceso de control continúan en paralelo con el scan del programa.

5. La próxima vez que el renglón se vuelve falso, una vez establecido el bit .DN o .ER, el controlador borra el bit .EN.

6. La instrucción puede ejecutarse nuevamente cuando el renglón se vuelve verdadero.

Existen 6 tipos de instrucciones para el control de movimiento en el RSLogix 5000:

- ✓ Instrucciones de estado de movimiento
- ✓ Instrucciones que producen movimiento
- ✓ Instrucciones de grupo de movimiento
- ✓ Instrucciones de evento de movimiento
- ✓ Instrucciones de configuración de movimiento
- ✓ Instrucciones de movimiento coordinado

Antes de describir cada tipo de instrucción de movimiento, se presentan los cinco estados de operación de un eje, esto con el fin de entender mejor la descripción de cada una de las instrucciones que se describen más adelante. Estos cinco estados son:

Estado de operación	***Descripción***
Eje listo	Éste es el estado normal de encendido del eje. En este estado: • La salida de habilitación del variador del servomódulo está inactiva. • La acción del servo está inhabilitada. • No hay fallos del servo.
Control directo del variador	Este estado de funcionamiento permite al servomódulo DAC controlar directamente un variador externo. En este estado: • La salida de habilitación del variador del servomódulo está activa. • La acción del servo de posición está inhabilitada.
Control del servo	Este estado de funcionamiento permite al servomódulo realizar movimiento de lazo cerrado. En este estado: • La salida de habilitación del variador del servomódulo está activa. • La acción del servo está habilitada. • El eje está forzado a mantener la posición de servo comandada.
Eje en fallo	En este estado de funcionamiento, hay un fallo en el servo y el estado de la salida de habilitación del variador, la acción del servo y la condición del contacto OK dependen de los fallos y las acciones de fallo presentes.

Interrupción	Este estado de funcionamiento permite a los contactos de relé OK abrir un conjunto de contactos en la cadena de paro de emergencia de la fuente de alimentación del variador. En este estado: • la salida de habilitación del variador del servomódulo está inactiva. • la acción del servo está inhabilitada. • el contacto en buen estado está abierto.

Tabla11. Estados de Operación de un Eje. [24]

- **Instrucciones de Estado de Movimiento.** Las instrucciones de control de estado de movimiento controlan o cambian directamente los estados de operación de un eje. Las instrucciones de estado de movimiento son:

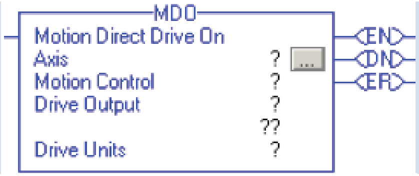
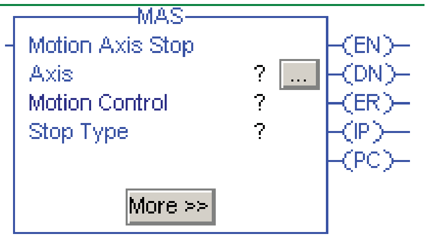
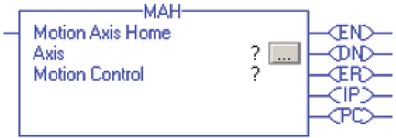
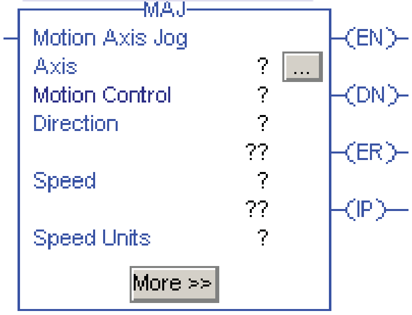
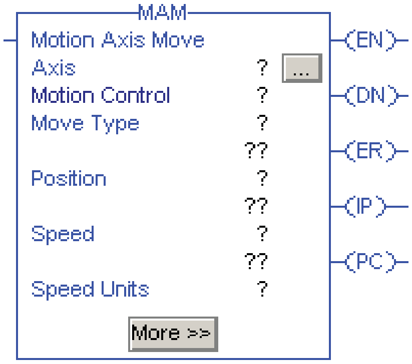
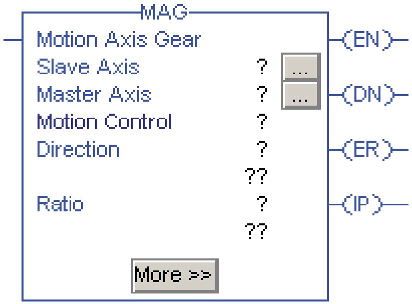
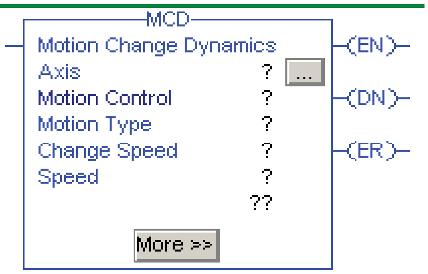
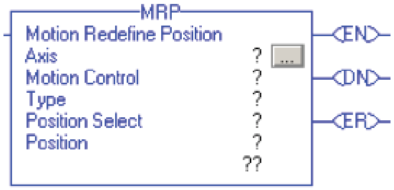
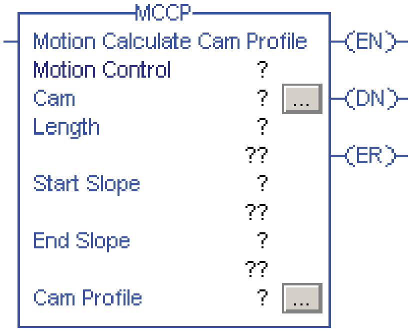
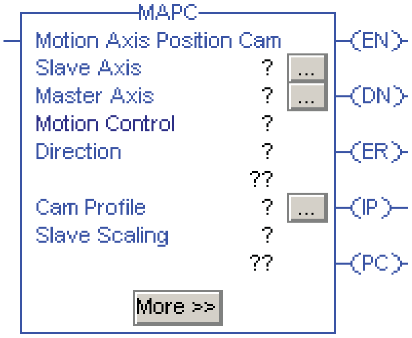
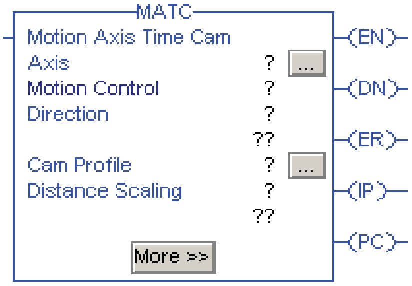
INSTRUCCIONES	DESCRIPCION	SIMBOLO
MSO (Motion Servo On)	Habilita el servo-drive y activar el lazo del servo-eje.	
MSF (Motion Servo Off)	Inhabilita el servo-drive y desactivar el lazo del servo-eje.	
MASD (Motion Axis Shutdown)	Forza un eje al estado de interrupción de operación. Una vez que el eje está en el estado de interrupción de operación, el controlador bloqueará cualquier instrucción que inicie movimiento del eje.	
MASR (Motion Axis Shutdown Reset)	Cambiar un eje de un estado de desactivación existente a una tensión de eje listo. Si se eliminan del estado de desactivación todos los ejes de un servomódulo como resultado de esta instrucción, se cerrarán los contactos de relé OK para el módulo.	
MDO (Motion Direct Drive On)	Habilitar el servodrive y establecer el voltaje de salida del servo de un eje.	
MDF (Motion Direct Drive Off)	Desactivar el servodrive y establecer el voltaje de salida del servo al voltaje de offset de salida.	
MAFR (Motion Axis Fault Reset)	Borrar todos los fallos de movimiento para un eje.	

Tabla 12. Instrucciones de Estado de Movimiento [23]

- **Instrucciones que Producen Movimiento.** Las instrucciones que producen movimiento controlan todos los aspectos de posición del eje. Las instrucciones que producen movimiento son:

INSTRUCCIONES	DESCRIPCION	SIMBOLO
MAS (Motion Axis Stop)	Detener algún proceso de movimiento en un eje	
MAH (Motion Axis Home)	Volver el eje a la posición inicial	
MAJ (Motion Axis Jog)	Impulsar un eje	
MAM (Motion Axis Move)	Mover un eje a una posición específica	
MAG (Motion Axis Gear)	Iniciar un engranaje electrónico entre 2 ejes	

MCD (Motion Change Dynamics)	Cambiar la velocidad, aceleración o desaceleración de un movimiento o impulso en progreso	
MRP (Motion Redefine Position)	Cambiar el comando o la posición real de un eje	
MCCP (Motion Calculate Cam Profile)	Calcular un perfil de levas basado en una matriz de puntos de levas	
MAPC (Motion Axis Position Cam)	Iniciar una operación electrónica de levas entre 2 ejes	
MATC (Motion Axis Time Cam)	Iniciar una operación electrónica de levas en función al tiempo	

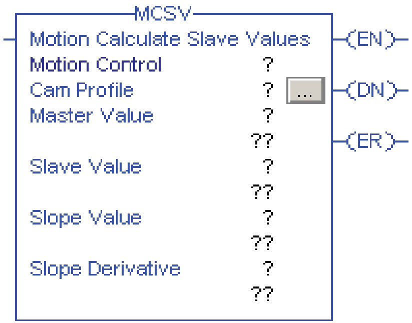
MCSV (Motion Calculate Slave Values)	Calcular el valor del esclavo, la pendiente y la derivada de la pendiente para un perfil de levas y un valor de maestro	 MCSV Motion Calculate Slave Values Motion Control ? Cam Profile ? ... Master Value ? ?? Slave Value ? ?? Slope Value ? ?? Slope Derivative ? ?? EN DN ER

Tabla 13. Instrucciones que Producen Movimiento. [23]

- **Instrucciones de Grupo de Movimiento.** Las instrucciones de control de grupo incluyen todas las instrucciones de control de movimiento que operan en todos los ejes el en grupo especificado. Las instrucciones que pueden aplicarse a grupos incluyen estroboscopio, control de interrupción e instrucciones de paro.

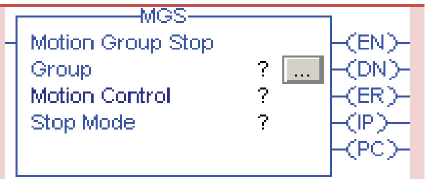
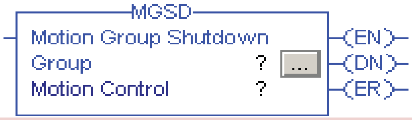
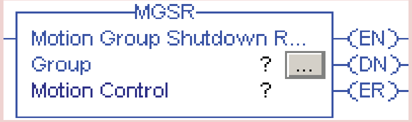
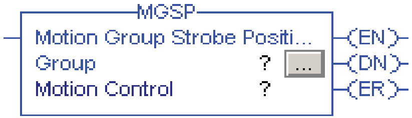
INSTRUCCIONES	DESCRIPCION	SIMBOLO
MGS (Motion Group Stop)	Iniciar un paro de movimiento en un grupo de ejes.	 MGS Motion Group Stop Group ? ... Motion Control ? Stop Mode ? EN DN ER IP PC
MGSD (Motion Group Shutdown)	Forzar todos los ejes de un grupo en el estado de interrupción de operación.	 MGSD Motion Group Shutdown Group ? ... Motion Control ? EN DN ER
MGSR (Motion Group Shutdown Reset)	Realizar una transición de un grupo de ejes del estado operativo de interrupción al estado de eje listo.	 MGSR Motion Group Shutdown R... Group ? ... Motion Control ? EN DN ER
MGSP (Motion Group Strobe Position)	Enclavar la posición de comando actual y real de todos los ejes en un grupo.	 MGSP Motion Group Strobe Positi... Group ? ... Motion Control ? EN DN ER

Tabla 14. Instrucciones de Grupo de Movimiento. [23]

- **Instrucciones de Evento de Movimiento.** Las instrucciones de evento de movimiento controlan la activación y la desactivación de las funciones de verificación de eventos especiales, tales como la posición de registro y control. Las instrucciones de evento de movimiento son:

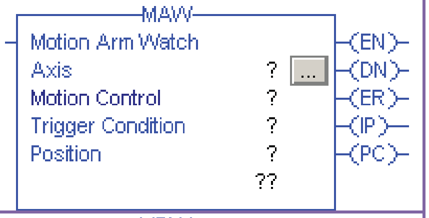
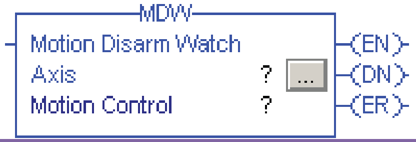
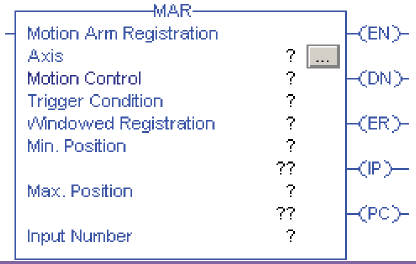
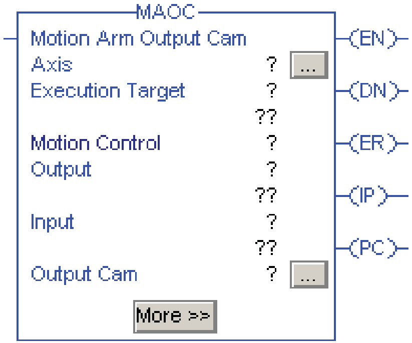
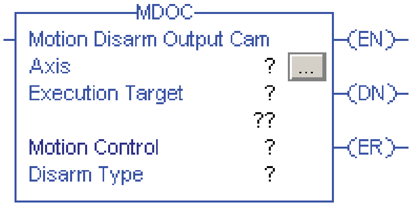
INSTRUCCIONES	DESCRIPCION	SIMBOLO
MAW (Motion Arm Watch)	Activar la verificación de un evento de posición de control para un eje.	
MDW (Motion Disarm Watch)	Desactivar la verificación de un evento de posición de control para un eje.	
MAR (Motion Arm Registration)	Activar la verificación de evento de registro del servomódulo para un eje.	
MDR (Motion Disarm Registration)	Desactivar la verificación de evento de registro del servomódulo para un eje.	
MAOC (Motion Arm Output Cam)	Activar una leva de salida	
MDOC (Motion Disarm Output Cam)	Desactivar una leva de salida	

Tabla 15. Instrucciones de Evento de Movimiento. [23]

- **Instrucciones de Configuración de Movimiento.** Las instrucciones de configuración incluyen todas las instrucciones de control de movimiento que se utilizan para establecer y aplicar parámetros de servo configuración a un eje.

Este grupo de instrucciones incluye instrucciones de diagnóstico de prueba de conexión e instrucciones de ajuste, entre estas se encuentran:

- Una prueba de conexión de encoder de motor.
- Una prueba de conexión de encoder.
- Una prueba de impulso de cero.

Las instrucciones de configuración de ejes son:

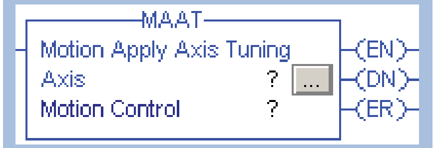
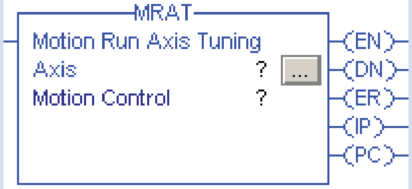
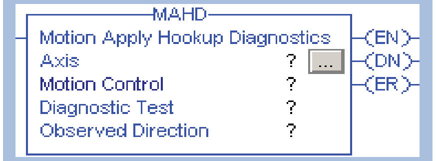
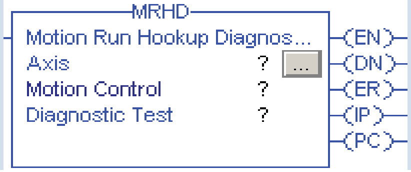
INSTRUCCIONES	DESCRIPCION	SIMBOLO
MAAT (Motion Apply Axis Tuning)	Calcular un conjunto completo de ganancias del servo y límites dinámicos basados en una instrucción MRAT previamente ejecutada. La instrucción MAAT también actualiza el servomódulo con los nuevos parámetros de ganancia.	 MAAT Motion Apply Axis Tuning Axis ? ... Motion Control ? EN DN ER
MRAT (Motion Run Axis Tuning)	Ordenar al servomódulo que realice un perfil de control de movimiento de ajuste para un eje.	 MRAT Motion Run Axis Tuning Axis ? ... Motion Control ? EN DN ER IP PC
MAHD (Motion Apply Hookup Diagnostics)	Aplicar los resultados de una instrucción MRDH previamente ejecutada. La instrucción MAHD genera un nuevo conjunto de polaridades de encoder y servo basadas en la dirección observada de movimiento durante la instrucción MRHD.	 MAHD Motion Apply Hookup Diagnostics Axis ? ... Motion Control ? Diagnostic Test ? Observed Direction ? EN DN ER
MRHD (Motion Run Hookup Diagnostics)	Ordenar el servomódulo que realice una de tres pruebas de diagnóstico en un eje.	 MRHD Motion Run Hookup Diagnos... Axis ? ... Motion Control ? Diagnostic Test ? EN DN ER IP PC

Tabla 16. Instrucciones de Configuración de Movimiento. [23]

- **Instrucciones de Movimiento Coordinado.** Las instrucciones de movimiento coordinado permiten mover los tres ejes en un sistema de coordenadas. Las instrucciones de movimiento coordinado son:

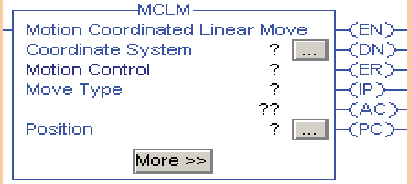
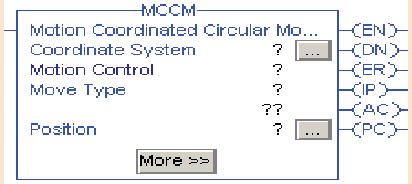
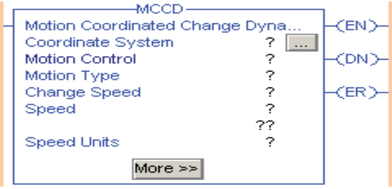
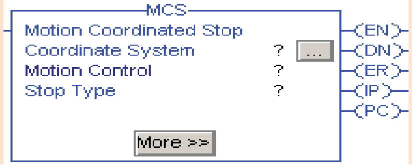
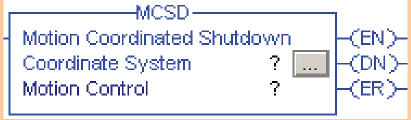
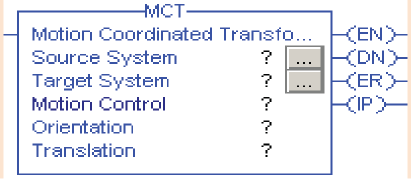
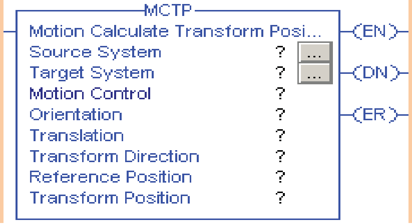
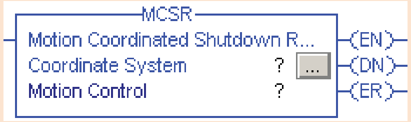
INSTRUCCIONES	DESCRIPCION	SIMBOLO
MCLM (Motion Coordinated Linear Move)	Iniciar un movimiento coordinado lineal sencillo o multidimensional para los ejes específicos dentro del Sistema de coordenadas cartesianas.	
MCCM (Motion Coordinated Circular Move)	Iniciar un movimiento coordinado circular bidimensional o tridimensional para los ejes específicos dentro del Sistema de coordenadas cartesianas.	
MCCD (Motion Coordinated Change Dynamics)	Iniciar un cambio en la dinámica de la ruta para el movimiento coordinado activo en un sistema de coordenadas específico.	
MCS (Motion Coordinated Stop)	Detener los ejes de un sistema de coordenadas o cancelar una transformación.	
MCSD (Motion Coordinated Shutdown)	Iniciar una interrupción controlada de todos los ejes del sistema de coordenadas específico.	
MCT (Motion Coordinated Transform)	Iniciar una transformación que vincula dos sistemas coordinados juntos.	
MCTP (Motion Calculate Transform Position)	Calcular la posición de un sistema de coordenadas con respecto a otro sistema de coordenadas.	
MCSR (Motion Coordinated Shutdown Reset)	Restablecer todos los ejes de un sistema de coordenadas específico desde el estado de interrupción al estado de eje preparado y borrar los fallos del eje.	

Tabla 17. Instrucciones de Movimiento Coordinado [23]

3. IMPLEMENTACIÓN DEL SISTEMA DE CONTROL DE MOVIMIENTO.

3.1. Montaje y Cableado

El montaje y cableado de potencia del PAC y los Servo Drives que componen el sistema de control de movimiento se realizó como parte del proyecto: *IMPLEMENTACIÓN DEL SISTEMA DE CONTROL TECNOLOGÍA ROCKWELL AUTOMATION EN EL LABORATORIO DE AUTOMATIZACIÓN INDUSTRIAL DE LA UNIVERSIDAD PONTIFICIA BOLIVARIANA.* El sistema de control de movimiento, como ya se mencionó con anterioridad se constituye por un software de programación y configuración (RSLogix 5000), un controlador (ControlLogix 5562), el módulo SERCOS (1756-M03SE), los servo drives (2098-DSD-005-SE) y los servo motores (MPL-A1510V-EJ42AA). El controlador, el módulo SERCOS y el modulo Ethernet que permite la comunicación del controlador con el software RSLogix 5000 se encuentran integrados en el chasis 1756-A13 ubicado en el gabinete principal (Segmento denominado Etapa de Control). (Ver Figura 31)

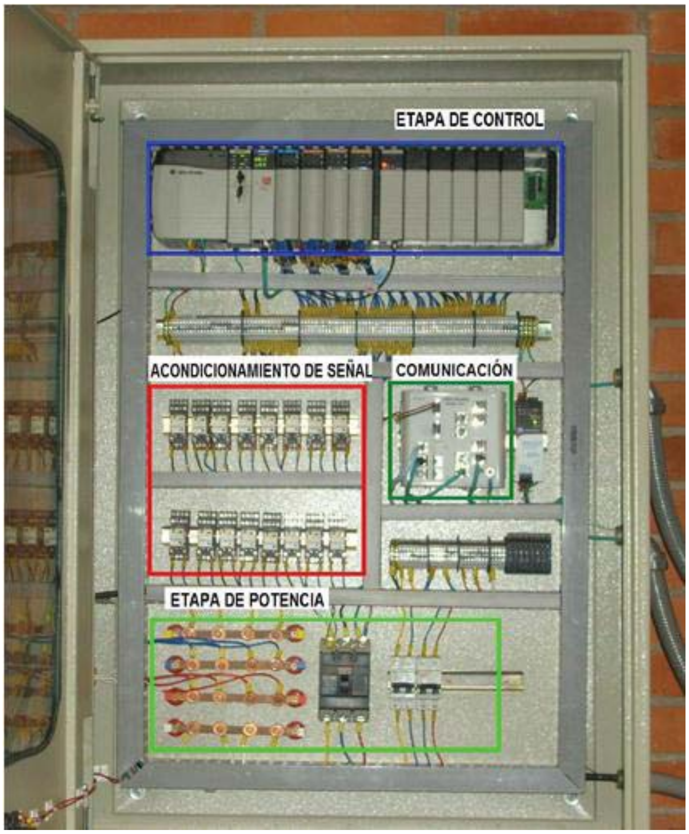


Figura 31. Distribución del gabinete principal. [19]

Figura 32. Identificación del Controlador y Módulos Ethernet y SERCOS. [28]

Los servo drives y sus respectivos filtros se encuentran ubicados en el gabinete de control de movimiento tal y como se muestra en la figura 33.

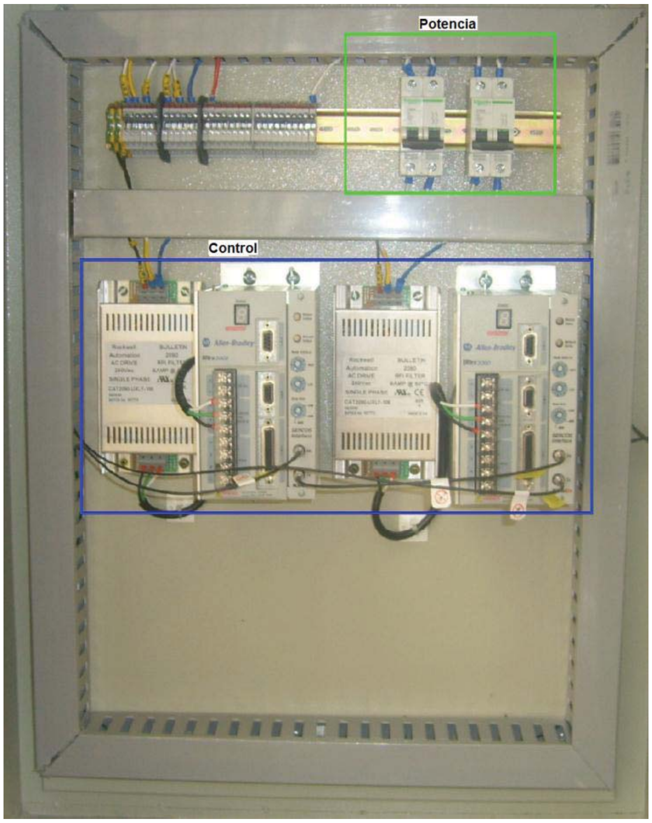


Figura 33. Distribución del Gabinete de Control. [28]

Figura 34. Identificación de Filtro y el Drive. [28]

Todo el montaje y conexión de los equipos obedece a las sugerencias y especificaciones que brinda el fabricante en sus documentos y publicaciones. La figura 17 presenta la forma de realizar la conexión de los servo drives y el módulo SERCOS a través de fibra óptica; la figura 22 muestra cómo se debe realizar la conexión entre los filtros RFI y los Servo Drives; la figura 23 muestra los terminales del cable de poder, a su vez de los terminales allí especificados se conectan solo los identificados como Brown al terminal U, Black al terminar V, el terminal Blue al terminal W y Green/Yellow al terminal de tierra del drive, tal como se ve en el recuadro amarillo en la figura 35.

Figura 35. Conexión del Cable de Poder del Servo al Drive [28]

La figura 25 y 26 presentan la distribución del conector CN2 y la respectiva conexión entre el cable de Feedback y el conector respectivamente, en la figura 35, recuadro naranja se puede apreciar la conexión del conector CN2 con el Drive.

La estructura general del sistema de control implementado se puede apreciar en la figura 35, allí se muestra cada uno de los dispositivos que conforma el sistema y la conexión entre ellos. (Ver Figura 36)

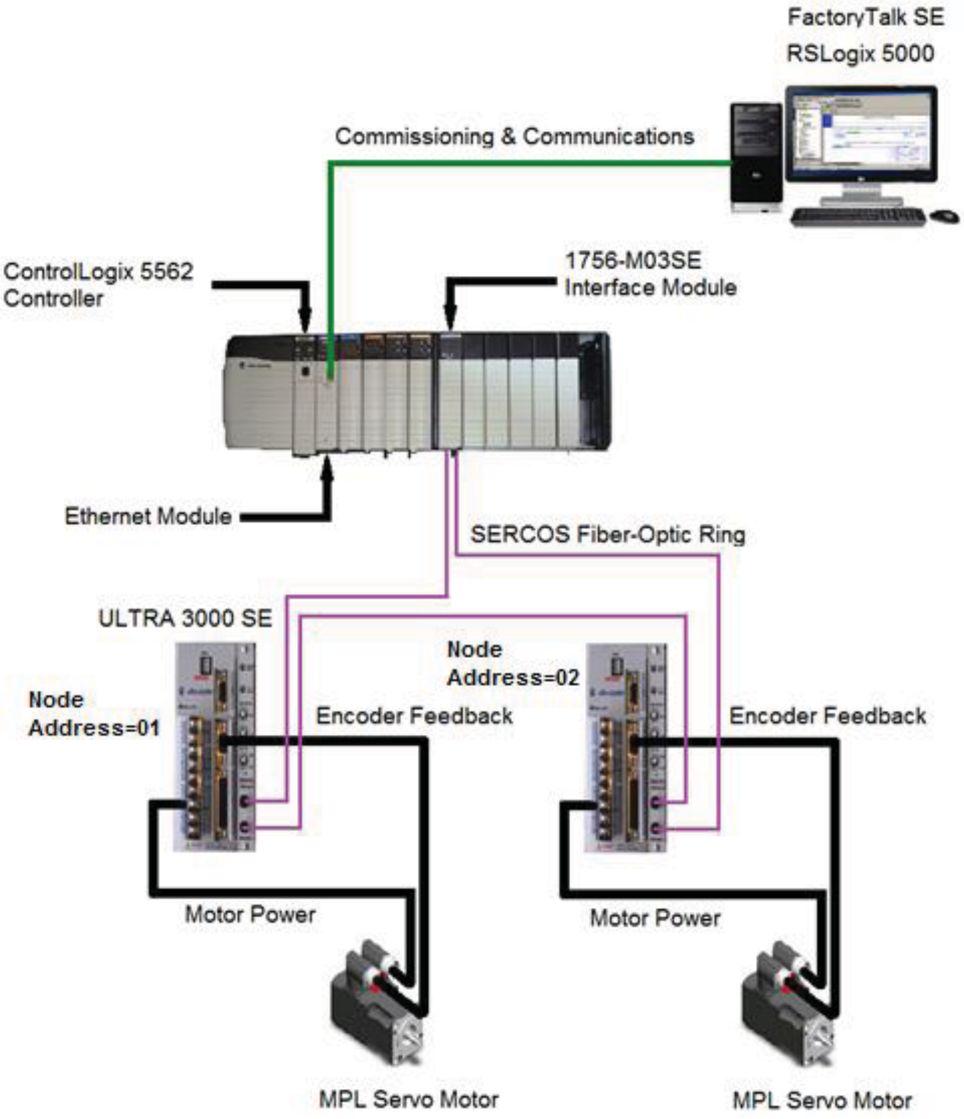


Figura 36. Estructura General del Sistema de Control de Movimiento Implementado. [28]

La figura 37 presenta el gabinete de control de movimiento, con la conexión de cada uno de los equipos que lo integra.

Figura 37. Gabinete del Sistema de Control de Movimiento Implementado. [28]

3.2. Descripción de las Instrucciones Utilizadas en la Programación del PAC

Este ítem especifica las instrucciones de movimiento, de entrada y salida, de cálculo/matemáticas y de control de programa, utilizadas en el controlador 1756-L62 para la aplicación realizada.

Inicialmente se explicarán las instrucciones de movimiento:

- **Motion Servo On (MSO).** La instrucción MSO es una instrucción de salida que utiliza la ejecución de tipo mensaje. Use la instrucción MSO para habilitar el servo drive y activar el lazo del servo eje.

  Un uso común para esta instrucción es la activación de un lazo del servo eje como preparación para ordenar el movimiento. Para usar la instrucción MSO, configure el eje como servo eje. [24]

Operandos:

Operando	Tipo	Formato	Descripción
AXIS	AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje en el cual se realizará la operación.
Motion Control	MOTION_INSTRUCTION	tag	Estructura usada para obtener acceso a los parámetros de estado de instrucción.

Tabla 18. Operandos de la Instrucción MSO [23]

Estructura MOTION_INSTRUCTION

Mnemónico:	Descripción:
.EN (Habilitar) Bit 31	Se establece cuando el renglón hace una transición de falso a verdadero y permanece establecido hasta que la transacción de mensaje del servo se completa y el renglón pasa a falso.
Bit .DN (Listo) 29	Se establece cuando la acción del servo del eje ha sido habilitada correctamente y los bits Drive Enable y Servo Active Status se han establecido.
Bit .ER (Error) 28	Se establece para indicar que la instrucción ha detectado un error, como si se hubiese especificado un eje no configurado.

Tabla 19. Estructura MOTION_INSTRUCTION [23]

A su vez para observar los posibles errores que pueden surgir al usar esta instrucción remitirse a la página 395 del Anexo J.

- **Motion Servo Off (MSF).** La instrucción MSF es una instrucción de salida que utiliza la ejecución de tipo mensaje.

  Use la instrucción MSF para inhabilitar el servo drive y desactivar el lazo del servo eje. Aunque la acción del servo está inhabilitada, el controlador sigue monitorizando la posición real del eje.

  Cuando la acción del servo se habilita usando la instrucción de activar servo de movimiento (MSO), el servo mantiene la posición. Para usar la instrucción MSF, configure el eje como servo eje. [24]

Operandos:

Operando	Tipo	Formato	Descripción
AXIS	**AXIS_GENERIC** **AXIS_SERVO** **AXIS_SERVO_DRIVE**	**tag**	**Nombre del eje en el cual se realizará la acción.**
Motion Control	**MOTION_INSTRUCTION**	**tag**	**Estructura usada para obtener acceso a los parámetros de estado de instrucción.**

Tabla 20. Operandos de la Instrucción MSF [23]

Estructura MOTION_INSTRUCTION

Mnemónico:	Descripción:
.EN (Habilitar) Bit 31	**Se establece cuando el renglón hace una transición de falso a verdadero y permanece establecido hasta que la transacción de mensaje del servo se completa y el renglón pasa a falso.**
Bit .DN (Listo) 29	**Se establece cuando la acción del servo del eje ha sido habilitada correctamente y los bits Drive Enable y Servo Active Status se han establecido.**
Bit .ER (Error) 28	**Se establece para indicar que la instrucción ha detectado un error, como si se hubiese especificado un eje no configurado.**

Tabla 21. Estructura MOTION_INSTRUCTION [23]

A su vez para observar los posibles errores que pueden surgir al usar esta instrucción remitirse a la página 395 del Anexo J.

- **Motion Axis Fault Reset (MAFR).** La instrucción MAFR es una instrucción de salida que utiliza la ejecución de tipo mensaje. Use la instrucción MAFR para borrar todos los fallos de movimiento para un eje. Éste es el único método para borrar los fallos de movimiento del eje.

La instrucción MAFR generalmente se usa como parte de un programa de administración de fallos. Un programa de administración de fallos proporciona una acción específica como respuesta a fallos posibles. Una vez eliminada la condición de fallo, el bloque MAFR restablece todos los bits de estado de fallo activos. Para usar la instrucción MAFR, configure el eje como servo eje o eje de sólo posición. [24]

Operandos:

Operando	*Tipo*	*Formato*	*Descripción*
AXIS	AXIS_FEEDBACK AXIS_VIRTUAL AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje en el cual se realizará la acción.
Motion Control	MOTION_INSTRUCTION	tag	Estructura usada para obtener acceso a los parámetros de estado de instrucción.

Tabla 22. Operandos de la instrucción MAFR [23]

Estructura MOTION_INSTRUCTION

Mnemónico:	*Descripción:*
.EN (Habilitar) Bit 31	Se establece cuando el renglón hace una transición de falso a verdadero y permanece establecido hasta que la transacción de mensaje del servo se completa y el renglón pasa a falso.
Bit .DN (Listo) 29	Se establece cuando se han borrado correctamente los fallos del eje.
Bit .ER (Error) 28	Se establece para indicar que la instrucción ha detectado un error, como si se hubiese especificado un eje no configurado.

Tabla 23. Estructura MOTION_INSTRUCTION [23]

A su vez para observar los posibles errores que pueden surgir al usar esta instrucción remitirse a la página 395 del Anexo J.

- **Motion Axis Jog (MAJ).** La instrucción MAJ es una instrucción de salida que utiliza la ejecución de tipos inmediato y proceso. Use la instrucción MAJ para iniciar un perfil de control de movimiento por impulsos para un eje.

  Para interrumpir un impulso de eje, use la instrucción de impulsos de eje de movimiento (MAJ) con una velocidad de cero o la instrucción de paro de eje de movimiento (MAS).

  Para usar la instrucción MAJ:

  - Configure el eje como servo eje o como eje virtual.
  - Asegúrese de que la tensión del eje tiene el servo activado si el eje es un Servo eje. [24]

Operandos:

Operando	Tipo	Formato	Descripción
AXIS	AXIS_VIRTUAL AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	Tag	Nombre del eje por impulsar
Motion Control	MOTION_ INSTRUCTION	Tag	Tag de control para la instrucción
Direction	DINT	Tag Inmediato	**Para esta dirección de impulso** – **Introduzca** Avance – 0 Retroceso – 1
Speed	REAL	Tag Inmediato	Velocidad para mover el eje en unidades de velocidad.
Speed Units	DINT	Inmediato	¿Qué unidades desea utilizar para la velocidad? - Unidades por seg (0) - % del Máximo (1)
Accel Rate	REAL	Tag Inmediato	Velocidad de aceleración del eje en unidades de aceleración
Accel Units	DINT	Inmediato	¿Qué unidades desea utilizar para la velocidad de aceleración? - Unidades por seg2 (0) - % del Máximo (1)
Decel Rate	REAL	Tag Inmediato	Velocidad de desaceleración del eje en unidades de desaceleración.
Decel Units	DINT	Inmediato	¿Qué unidades desea utilizar para la velocidad de desaceleración? - Unidades por seg2 (0) - % del Máximo (1)
Profile	DINT	Inmediato	Seleccione el perfil de velocidad para ejecutar el impulso: - Trapezoidal (0) - S-curve (1)
Accel Jerk	REAL	Inmediato o tag	Siempre **debe** introducir valores para los operandos Accel y Decel Jerk. Esta instrucción sólo utiliza los valores si el perfil está configurado como curva en S. - Accel Jerk es el régimen de jaloneo de aceleración del eje. - Decel Jerk es el régimen de jaloneo de desaceleración del eje. Utilice estos valores para comenzar. • Accel Jerk = 100 (% de Tiempo) • Decel Jerk = 100 (% de Tiempo) • Unidades de jaloneo = 2 Introduzca los regímenes de jaloneo en estas unidades de jaloneo. 0 = Unidades por seg3 1 = % del Máximo
Decel Jerk	REAL	Inmediato o tag	
Jerk Units	DINT	Inmediato	

			2 = % de Tiempo (utilice este valor para comenzar)
Merge	DINT	Inmediato	¿Desea convertir todos los movimientos actuales del eje en un impulso puro regido por esta instrucción independientemente de las instrucciones de control de movimiento actualmente en proceso? - NO – Seleccione Inhabilitado (0) - YES – Seleccione Habilitado (1)
Merge Speed	DINT	Inmediato	Si Merge está habilitado, ¿A qué velocidad desea desplazarse por impulsos? - Velocidad de esta instrucción – Seleccione Programmed (0) - Velocidad actual del eje – Seleccione Current (1)

Tabla 24. Operandos de la instrucción MAJ [23]

Estructura MOTION_INSTRUCTION

Mnemónico:	Descripción:
.EN	El bit de habilitación indica cuándo la instrucción está habilitada. Permanece establecido hasta que se hace falsa la condición de entrada de renglón.
.DN	El bit de efectuado indica cuándo la instrucción inicia impulso del eje.
.ER	El bit de error indica cuándo la instrucción detecta un error, por ejemplo si el eje no está configurado.
.IP	• El bit en proceso se establece cuando se inicia con éxito el proceso por impulsos. • Se restablece cuando ocurre uno de los eventos siguientes: • Otra ocurrencia MAH reemplaza la instrucción actual. • Se cancela la operación por impulsos. • Un fallo del servo termina la instrucción MAJ.
.ACCEL	El bit .ACCEL indica que la velocidad ha aumentado para la instrucción individual a la que se encuentra asociado, como impulso, movimiento o sincronismo digital.
.DECEL	El bit .DECEL indica que la velocidad ha disminuido para la instrucción individual a la que se encuentra asociado, como impulso, movimiento o sincronismo digital.

Tabla 25. Estructura MOTION_INSTRUCTION [24]

A su vez para observar los posibles errores que pueden surgir al usar esta instrucción remitirse a la página 395 del Anexo J.

- **Motion Axis Move (MAM).** La instrucción MAM es una instrucción de salida que utiliza la ejecución de tipos inmediato y proceso. Use la instrucción MAM para iniciar un perfil de control de movimiento para un eje.

Para usar la instrucción MAM:

- Configure el eje como servo eje o como eje virtual.
- Asegúrese de que la tensión del eje tiene el servo activado si el eje es un servo eje. [24]

Operandos:

<table>
<tr><th>Operando</th><th>Tipo</th><th>Formato</th><th colspan="3">Descripción</th></tr>
<tr><td>AXIS</td><td>AXIS_VIRTUAL
AXIS_GENERIC
AXIS_SERVO
AXIS_SERVO_DRIVE</td><td>Tag</td><td colspan="3">Nombre del eje
Para un movimiento absoluto u offset de maestro incremental, introduzca el eje esclavo.</td></tr>
<tr><td>Motion Control</td><td>MOTION_ INSTRUCTION</td><td>Tag</td><td colspan="3">Tag de control para la instrucción</td></tr>
<tr><td rowspan="7">Move Type</td><td rowspan="7">DINT</td><td rowspan="7">Tag
Inmediato</td><td>Para</td><td>Use este tipo de movimiento</td><td>E introduzca</td></tr>
<tr><td>Mover un eje a una posición absoluta</td><td>Absoluto</td><td>0</td></tr>
<tr><td>Mover un eje una distancia específica desde donde se encuentra ahora</td><td>Incremental</td><td>1</td></tr>
<tr><td>Mover un eje giratorio a una posición absoluta en la dirección más corta independientemente de su posición actual</td><td>Giratorio de ruta más corta</td><td>2</td></tr>
<tr><td>Mover un eje giratorio a una posición absoluta en la dirección positiva independientemente de su posición actual</td><td>Giratorio positivo</td><td>3</td></tr>
<tr><td>Mover un eje giratorio a una posición absoluta en la dirección negativa independientemente de su posición actual</td><td>Giratorio negativo</td><td>4</td></tr>
<tr><td>Desvía el valor de</td><td>Offset de</td><td>5</td></tr>
</table>

			maestro de una leva de posición a una posición absoluta	maestro absoluto	
			Desvía el valor de maestro de una leva de posición mediante una distancia incremental	Offset de maestro incremental	6
Position	REAL	Inmediato Tag	Posición absoluta o distancia incremental para el movimiento		
			Para este tipo de movimiento	**Introduzca este valor de posición**	
			Absoluto	Posición para mover a	
			Incremental	Distancia para mover	
			Giratorio de ruta más corta	Posición para mover a. Introduzca un valor positivo menor que el valor de posición de desbobinado.	
			Giratorio positivo		
			Giratorio de negativo		
			Offset de maestro absoluto	Posición offset absoluta	
			Offset de maestro incremental	Distancia offset incremental	
Speed	REAL	Tag Inmediato	Velocidad para mover el eje en unidades de velocidad.		
Speed Units	DINT	Inmediato	¿Qué unidades desea utilizar para la velocidad? - Unidades por seg (0) - % del Máximo (1)		
Accel Rate	REAL	Tag Inmediato	Velocidad de aceleración del eje en unidades de aceleración		
Accel Units	DINT	**Inmediato**	¿Qué unidades desea utilizar para la velocidad de aceleración? - Unidades por seg2 (0) - % del Máximo (1)		
Decel Rate	REAL	Tag Inmediato	Velocidad de desaceleración del eje en unidades de desaceleración.		
Decel Units	DINT	Inmediato	¿Qué unidades desea utilizar para la velocidad de desaceleración? • Unidades por seg2 (0) • % del Máximo (1)		
Profile	DINT	Inmediato	Seleccione el perfil de velocidad para ejecutar el movimiento: • Trapezoidal (0) • S-curve (1)		
Accel Jerk	REAL	Tag Inmediato	La instrucción sólo utiliza los operandos de jaloneo si el perfil es curva en S.		
Decel Jerk	REAL	Tag Inmediato	No obstante, siempre debe completarlos. - Accel Jerk es el régimen de jaloneo de aceleración		

Jerk Units	DINT	Inmediato	del eje. - Decel Jerk es el régimen de jaloneo de desaceleración del eje. Utilice estos valores para comenzar. • Accel Jerk = 100 • Decel Jerk = 100 • Unidades de jaloneo = 2 (% de Tiempo) Puede introducir también los regímenes de jaloneo en estas unidades de jaloneo. • Unidades por seg3 (0) • % del Máximo (1)
Merge	DINT	Inmediato	¿Desea convertir todos los movimientos actuales del eje en un movimiento puro regido por esta instrucción independientemente de las instrucciones de control de movimiento actualmente en proceso? • NO – Seleccione Inhabilitado (0) • YES – Seleccione Habilitado (1)
Merge Speed	DINT	Inmediato	Si Merge está habilitado, ¿A qué velocidad desea moverse? • Velocidad de esta instrucción – Seleccione Programmed (0) • Velocidad actual del eje – Seleccione Current (1)

Tabla 26. Operandos de la instrucción MAM [23]

Estructura MOTION_INSTRUCTION

Mnemónico:	Descripción:
.EN	El bit de habilitación indica cuándo la instrucción está habilitada. Permanece establecido hasta que se hace falsa la condición de entrada de renglón.
.DN	El bit de efectuado indica cuándo la instrucción inicia un movimiento del eje.
.ER	El bit de error indica cuándo la instrucción detecta un error, por ejemplo si el eje no está configurado.
.IP	• El bit en proceso se establece cuando se inicia con éxito el proceso de movimiento. • Se restablece cuando ocurre uno de los eventos siguientes: • La instrucción MAM se completa. • Se cancela la operación de movimiento. • Un fallo del servo interrumpe la instrucción MAM.
.PC	El bit de proceso concluido se establece cuando la instrucción completa una operación de movimiento.
.ACCEL	El bit .ACCEL indica que la velocidad ha aumentado para la instrucción individual a la que se encuentra asociado, como impulso, movimiento o sincronismo digital.
.DECEL	El bit .DECEL indica que la velocidad ha disminuido para la instrucción individual a la que se encuentra asociado, como impulso, movimiento o sincronismo digital.

Tabla 27. Estructura MOTION_INSTRUCTION [24]

A su vez para observar los posibles errores que pueden surgir al usar esta instrucción remitirse a la página 395 del Anexo J.

Posteriormente se explicaran las instrucciones de entrada/salida:

- **Get System Value (GSV)**

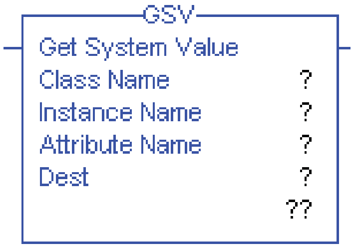


Figura 38. Instrucción Get System Value. [28]

Las instrucciones GSV reciben datos de sistema del controlador que se almacenan en objetos. El controlador almacena datos del sistema en objetos. No hay un archivo de estado, a diferencia del procesador PLC-5.

Cuando se habilita, la instrucción GSV recupera la información especificada y la coloca en un destino definido.

Cuando se introduce una instrucción GSV, el software de programación muestra las clases de objetos válidas, nombres de objetos y nombres de atributos para cada instrucción.

Operandos:

Operando	Tipo	Formato	Descripción
Class name		Nombre	nombre del objeto
Instance name		Nombre	nombre de objeto específico cuando el objeto requiere un nombre
Attribute Name		Nombre	atributo del objeto el tipo de datos depende del atributo que usted selecciona
Destination	SINT INT DINT REAL estructura	Tag	destino de los datos del atributo

Tabla 28. Operandos de la Instrucción GSV. [27]

Condiciones de fallo:

Ocurrirá un fallo menor si	Tipo de fallo	Código de fallo
dirección de objeto no válida	4	5
se especificó un objeto que no es compatible con GSV	4	6
atributo no válido	4	6
el destino GSV no es suficientemente grande **para** retener los datos solicitados	4	7

Tabla 29. Condiciones de Fallo Instrucción GSV. [27]

Ejecución:

Condición	Acción de lógica de escalera de relés
preescán	La condición de salida de renglón se establece como falsa.
condición de entrada de renglón es falsa	La condición de salida de renglón se establece como falsa.
condición de entrada de renglón es verdadera	La instrucción se ejecuta. La condición de salida de renglón se establece como verdadera.
EnableIn se establece	n. a.
la instrucción se ejecuta	Obtenga o establezca el valor especificado.
post-escán	La condición de salida de renglón se establece como falsa.

Tabla 30. Ejecución de la instrucción GSV. [27]

Objetos GSV. Cuando introduce una instrucción GSV, usted especifica el objeto y su atributo al cual desea tener acceso. En ciertos casos, existirá más de una instancia del mismo tipo de objeto, por lo que también puede ser necesario especificar el nombre del objeto. Por ejemplo, puede haber varias tareas en su aplicación. Cada tarea tiene su propio objeto TASK al cual usted accede mediante el nombre de la tarea.

Para obtener información acerca de los objetos a los cuales se pueden tener acceso ver Anexo K pagina 185, allí especifica en que documentos se puede encontrar la información de cada objeto.

De igual manera a continuación se explicarán las instrucciones de cálculo/matemáticas:

- **Compute (CPT).** La instrucción CPT realiza las operaciones aritméticas que usted define en la expresión. Cuando se habilita, la instrucción CPT evalúa la expresión y coloca el resultado en Destination.

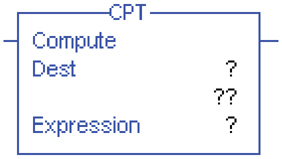
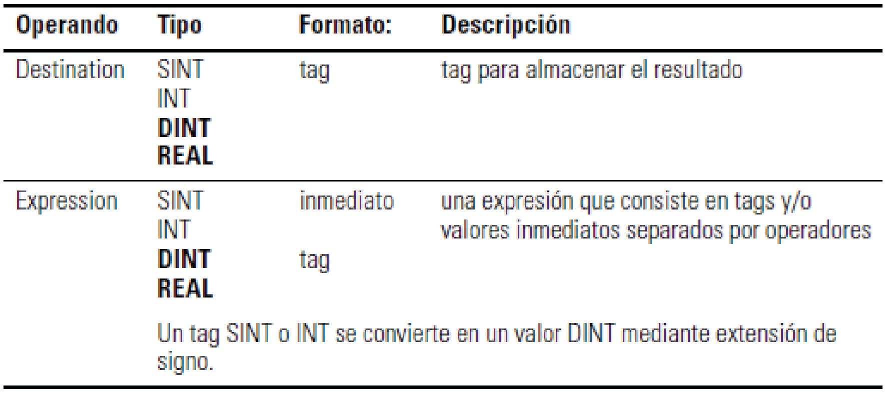


Figura 39. Instrucción Compute. [28]

La ejecución de una instrucción CPT es un poco más lenta y usa más memoria que la ejecución de las otras instrucciones de cálculo/matemáticas. La ventaja de la instrucción CPT es que le permite introducir expresiones complejas en una sola instrucción sin importar su longitud.

Operandos:



Operando	Tipo	Formato:	Descripción
Destination	SINT INT **DINT** **REAL**	tag	tag para almacenar el resultado
Expression	SINT INT **DINT** **REAL**	inmediato tag	una expresión que consiste en tags y/o valores inmediatos separados por operadores
	Un tag SINT o INT se convierte en un valor DINT mediante extensión de signo.		

Tabla 31. Operandos de la instrucción CPT. [27]

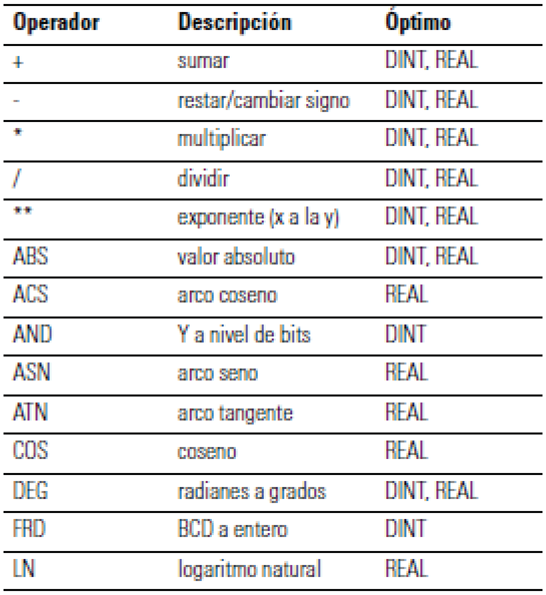
Condiciones de fallo: Esta instrucción no presenta ninguna condición de fallo.

Ejecución:

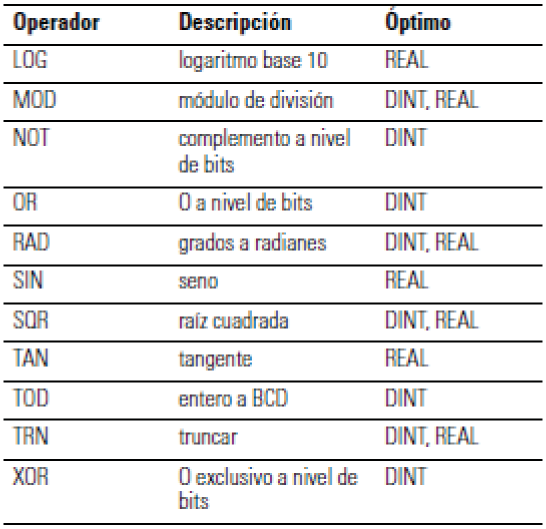
Condición	Acción de lógica de escalera de relés
Preescán	La condición de salida de renglón se establece como falsa.
condición de entrada de renglón es falsa	La condición de salida de renglón se establece como falsa.
condición de entrada de renglón es verdadera	La instrucción evalúa la expresión y coloca el resultado en Destination. La condición de salida de renglón se establece como verdadera.
post-escán	La condición de salida de renglón se establece como falsa.

Tabla 32. Ejecución de la instrucción CPT. [27]

Operadores válidos:



Operador	Descripción	Óptimo
+	sumar	DINT, REAL
-	restar/cambiar signo	DINT, REAL
*	multiplicar	DINT, REAL
/	dividir	DINT, REAL
**	exponente (x a la y)	DINT, REAL
ABS	valor absoluto	DINT, REAL
ACS	arco coseno	REAL
AND	Y a nivel de bits	DINT
ASN	arco seno	REAL
ATN	arco tangente	REAL
COS	coseno	REAL
DEG	radianes a grados	DINT, REAL
FRD	BCD a entero	DINT
LN	logaritmo natural	REAL



Operador	Descripción	Óptimo
LOG	logaritmo base 10	REAL
MOD	módulo de división	DINT, REAL
NOT	complemento a nivel de bits	DINT
OR	O a nivel de bits	DINT
RAD	grados a radianes	DINT, REAL
SIN	seno	REAL
SQR	raíz cuadrada	DINT, REAL
TAN	tangente	REAL
TOD	entero a BCD	DINT
TRN	truncar	DINT, REAL
XOR	O exclusivo a nivel de bits	DINT

Expresiones de formato:

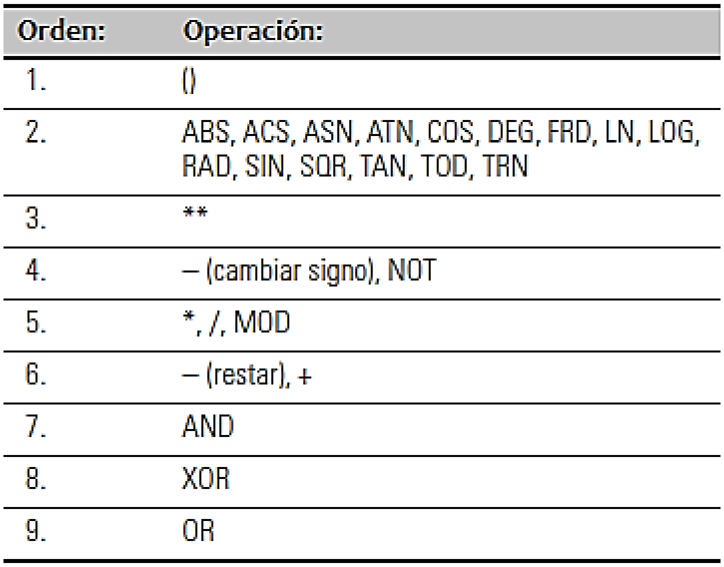
Por cada operador que use en una expresión, usted tiene que proporcionar uno o dos operandos (tags o valores inmediatos). Use la siguiente tabla para formatear operadores y operandos dentro de una expresión:

Para operadores que operan en:	Use este formato:	Ejemplos:
un operando	operador(operando)	ABS(*tag_a*)
dos operandos	operando_a operador operando_b	• *tag_b* + 5 • *tag_c* AND *tag_d* • (*tag_e* ** 2) MOD (*tag_f* / *tag_g*)

Orden de ejecución de las operaciones:

Las operaciones que usted escribe en la expresión son realizadas por la instrucción en un orden prescrito, que no es necesariamente el orden en que usted las escribe.

Usted puede anular el orden de operación agrupando términos dentro de paréntesis, forzando a la instrucción para que realice una operación dentro del paréntesis antes de otras operaciones. Las operaciones de igual orden se realizan de izquierda a derecha.



Orden:	Operación:
1.	()
2.	ABS, ACS, ASN, ATN, COS, DEG, FRD, LN, LOG, RAD, SIN, SQR, TAN, TOD, TRN
3.	**
4.	– (cambiar signo), NOT
5.	*, /, MOD
6.	– (restar), +
7.	AND
8.	XOR
9.	OR

Finalmente se explicarán las instrucciones de control de programa:

- **Jump to Subroutine (JSR).** La instrucción JSR inicia la ejecución de la rutina especificada, la cual se conoce como subrutina:
    - La subrutina se ejecuta una vez.
    - Después de que se ejecuta la subrutina, la ejecución de la lógica regresa a la rutina que contiene la instrucción JSR.

Operandos:

Operando	Tipo	Formato	Descripción
Nombre de la rutina	ROUTINE	nombre	rutina a ejecutar (es decir, subrutina)
	BOOL SINT INT DINT REAL estructura	inmediato tag tag de matriz	datos de esta rutina que desea copiar a un tag en la subrutina • Los parámetros de entrada son opcionales. • Introduzca múltiples parámetros de entrada, si es necesario.
Parámetro de retorno	BOOL SINT INT DINT REAL estructura	tag tag de matriz	el tag en esta rutina al cual desea copiar un resultado de la subrutina • Los parámetros de retorno son opcionales. • Introduzca múltiples parámetros de retorno, si es necesario.

Tabla 33. Operandos de la instrucción JSR. [27]

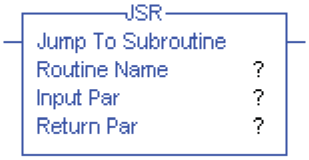


Figura 40. Instrucción Jump to Subroutine. [28]

Condiciones de fallo:

Ocurrirá un fallo mayor si	Tipo de fallo	Código de fallo
La instrucción JSR tiene menos parámetros de entrada que la instrucción SBR	4	31
La instrucción JSR salta a una rutina de fallo	4 o proporcionado por el usuario	0 o proporcionado por el usuario
La instrucción RET tiene menos parámetros de retorno que la instrucción JSR	4	31
La rutina principal contiene una instrucción RET	4	31

Tabla 34. Condiciones de Fallo Instrucción JSR. [27]

Ejecución:

Condición	Acción de lógica de escalera de relés	Acción de texto estructurado
preescán	El controlador ejecuta todas las subrutinas independientemente de la condición del renglón. Para asegurar que todos los renglones en la subrutina estén previamente escaneados, el controlador ignora las instrucciones RET. (es decir, las instrucciones RET no salen de la subrutina). – En las versiones 6.x y anteriores, se pasan los parámetros de entrada y retorno. – En las versiones 7.x y posteriores, no se pasan los parámetros de entrada y retorno. Si existen llamadas recursivas a la misma subrutina, la subrutina es preescaneada sólo la primera vez. Si existen múltiples llamadas (no recursivas) a la misma subrutina, la subrutina es preescaneada cada vez. La condición de salida de renglón se establece en falso (lógica de escalera de relés solamente).	
La condición de entrada de renglón es falsa para la instrucción JSR	La subrutina *no* se ejecuta. Las salidas en la subrutina permanecen en su último estado. La condición de salida de renglón se establece como	n. a.

	falsa.	
condición de entrada de renglón es verdadera	La instrucción se ejecuta. La condición de salida de renglón se establece como verdadera.	n. a.
EnableIn se establece	n. a.	EnableIn siempre se establece. La instrucción se ejecuta.

Tabla 35. Ejecución de la instrucción JSR. [27]

3.3. Programación y Configuración del Proyecto de Control de Movimiento

Como ya se ha mencionado, toda la programación y configuración de un sistema de control de movimiento se hace a través del software RSLogix 5000. Para programar y configurar el control de movimiento de la aplicación se siguieron los procedimientos que se describen a continuación:

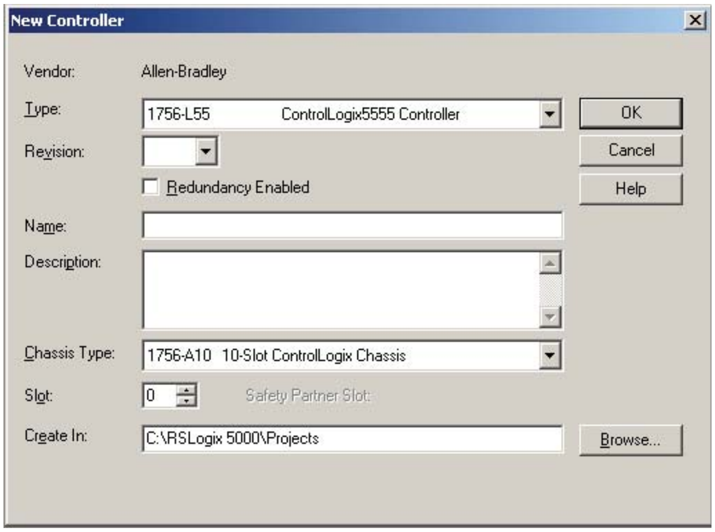
- Crear un nuevo proyecto de Controlador
- Seleccionar un dispositivo maestro de CST.
- Añadir el módulo de movimiento SERCOS.
- Añadir los servo drives.
- Crear un grupo de control de movimiento.
- Añadir un eje.
- Configurar un eje.
- Ejecutar las pruebas de conexión.
- Sintonice un eje SERCOS.
- Pruebe un eje con Motion Direct Commands.
- Desarrollar la lógica para el control de movimiento.

A continuación se describe el procedimiento en detalle de la programación y configuración:

3.3.1. Crear un nuevo proyecto de Controlador.

I. Abrir el Software RSLogix 5000.

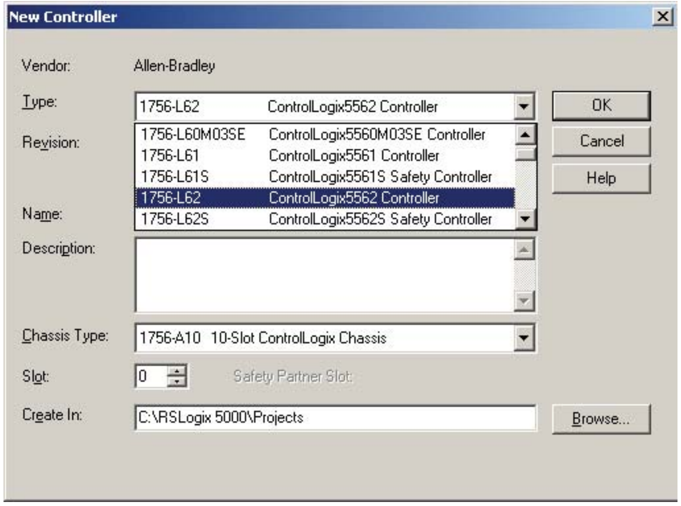
II. En el menú Archivo (***File***), seleccionar Nuevo ***(New)***, a continuación aparecerá la ventana ***New Controller .*** (Ver Figura 41)



Vendor: Allen-Bradley
Type: 1756-L55 ControlLogix5555 Controller
Revision:
☐ Redundancy Enabled
Name:
Description:
Chassis Type: 1756-A10 10-Slot ControlLogix Chassis
Slot: 0
Safety Partner Slot:
Create In: C:\RSLogix 5000\Projects

Figura 41. Ventana *New Controller.* [28]

III. Seleccionar el tipo de controlador que se va a implementar en la casilla ***Type***, en este caso se utiliza el controlador *ControlLogix 5562 (1756-L62).*



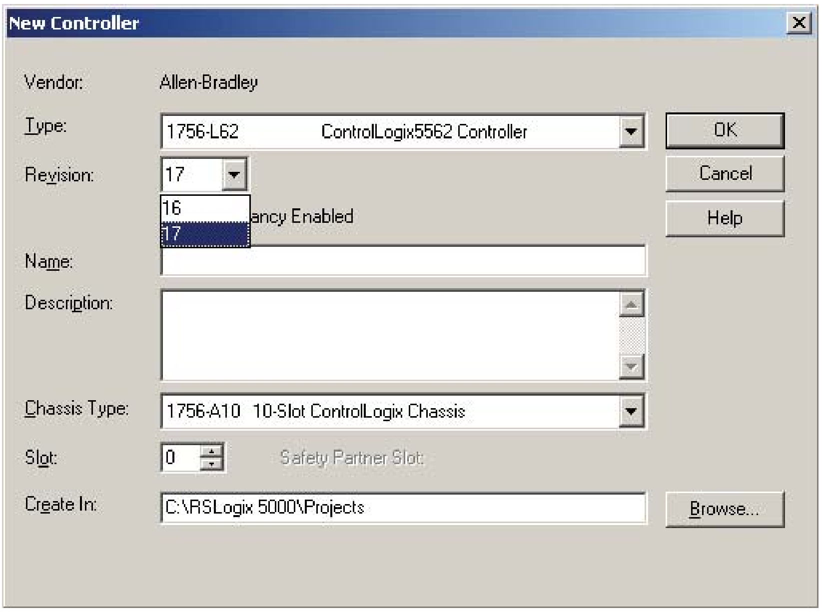
Vendor: Allen-Bradley
Type: 1756-L62 ControlLogix5562 Controller

1756-L60M03SE	ControlLogix5560M03SE Controller
1756-L61	ControlLogix5561 Controller
1756-L61S	ControlLogix5561S Safety Controller
1756-L62	ControlLogix5562 Controller
1756-L62S	ControlLogix5562S Safety Controller

Revision:
Name:
Description:
Chassis Type: 1756-A10 10-Slot ControlLogix Chassis
Slot: 0
Safety Partner Slot:
Create In: C:\RSLogix 5000\Projects

Figura 42. Selección del Controlador del Sistema. [28]

IV. Ingresar la revisión del software, para esto, seleccionar 17 en la casilla ***Revision.*** (Ver Figura 43)



New Controller

Vendor: Allen-Bradley

Type: 1756-L62 ControlLogix5562 Controller

Revision: 17

Name:

Description:

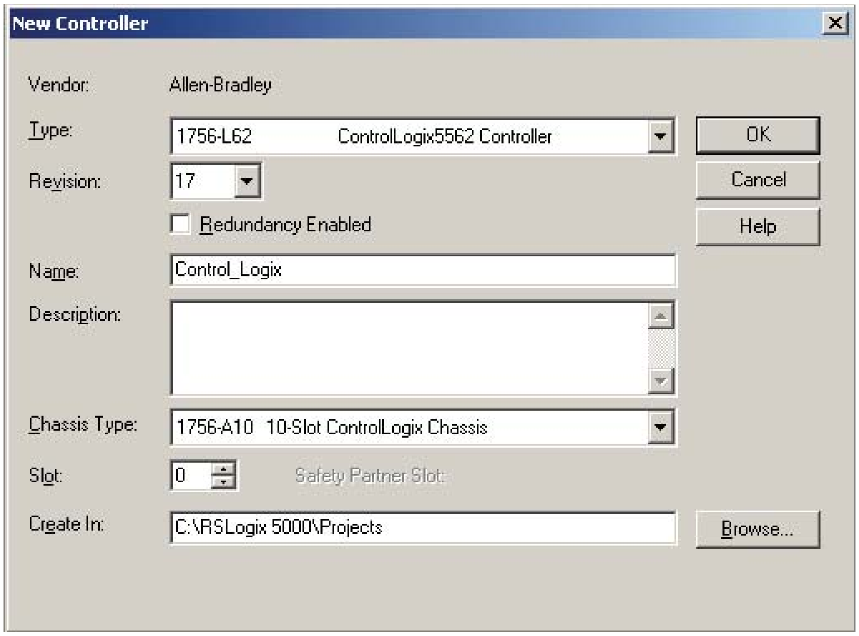
Chassis Type: 1756-A10 10-Slot ControlLogix Chassis

Slot: 0

Create In: C:\RSLogix 5000\Projects

Figura 43. Selección de la Revisión del Software. [28]

V. Darle un nombre al Controlador del sistema, puede darle cualquier nombre, solo ingréselo en la casilla ***Name.***



New Controller

Vendor: Allen-Bradley

Type: 1756-L62 ControlLogix5562 Controller

Revision: 17

☐ Redundancy Enabled

Name: Control_Logix

Description:

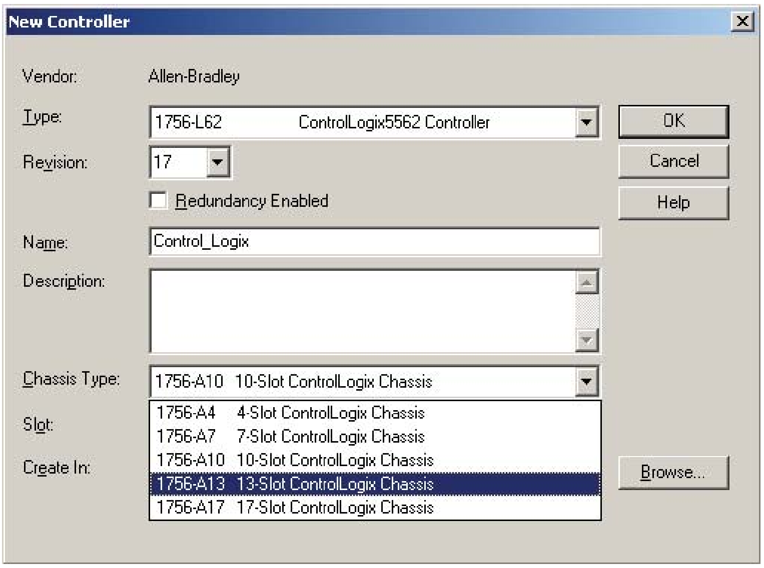
Chassis Type: 1756-A10 10-Slot ControlLogix Chassis

Slot: 0

Create In: C:\RSLogix 5000\Projects

Figura 44. Asignar Nombre al Controlador. [28]

VI. Elegir el tipo de chasis que se está utilizando, en este caso se tiene el chasis 1756-A13, el cual posee 13 slots. En la casilla ***Chassis Type***, buscar y seleccionar la referencia dada con anterioridad.


New Controller

Vendor: Allen-Bradley

Type: 1756-L62 ControlLogix5562 Controller

Revision: 17

☐ Redundancy Enabled

Name: Control_Logix

Description:

Chassis Type: 1756-A10 10-Slot ControlLogix Chassis

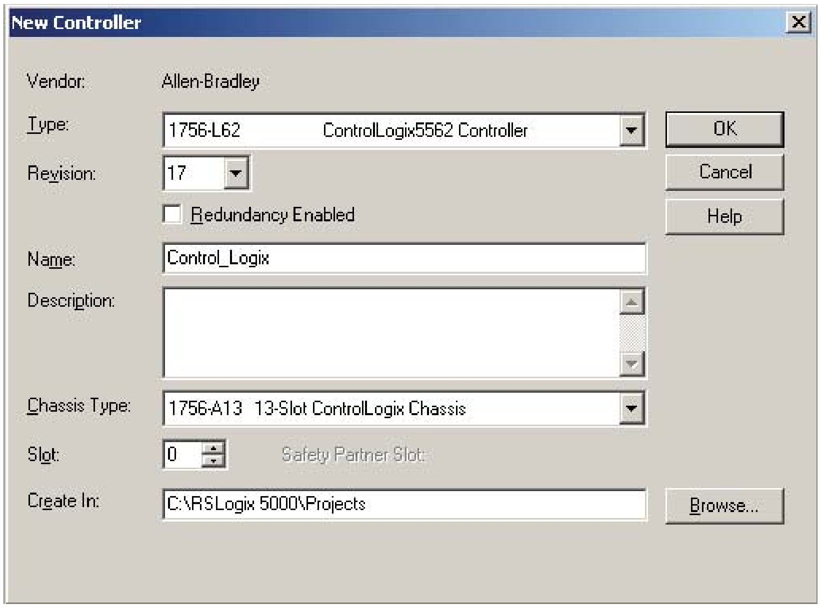
1756-A4 4-Slot ControlLogix Chassis
1756-A7 7-Slot ControlLogix Chassis
1756-A10 10-Slot ControlLogix Chassis
1756-A13 13-Slot ControlLogix Chassis
1756-A17 17-Slot ControlLogix Chassis

Slot:

Create In:

Figura 45. Seleccionar el Tipo de Chasis. [28]

VII. Seleccionar el Slot en el cual se encuentra ubicado el controlador, en este caso el controlador se encuentra en el primer Slot (posición 0 en el chasis), por lo que en la casilla ***Slot*** se dejara 0.


New Controller

Vendor: Allen-Bradley

Type: 1756-L62 ControlLogix5562 Controller

Revision: 17

☐ Redundancy Enabled

Name: Control_Logix

Description:

Chassis Type: 1756-A13 13-Slot ControlLogix Chassis

Slot: 0

Safety Partner Slot:

Create In: C:\RSLogix 5000\Projects

Figura 46. Especificación de la Posición del Controlador en el Chasis. [28]

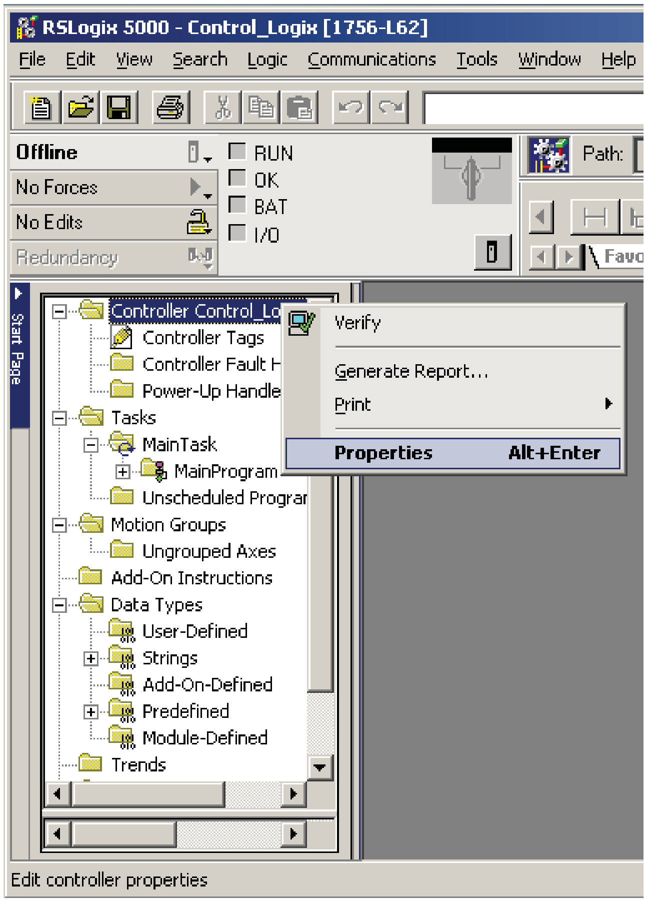
VIII. Especificar la ubicación donde se guardara el proyecto, generalmente se deja la opción por defecto, pero si se desea guardar el proyecto en otra ubicación, especifíquela en el campo ***Create In*.**

3.3.2. Seleccionar un Dispositivo Maestro de CST.

Si el controlador usa un eje de movimiento, hay que definir un dispositivo en el chasis como maestro del sistema coordinado de tiempo (CST). Esto sincroniza todos los módulos servo y controladores en el chasis según el mismo reloj.

Solo debe existir un dispositivo en el chasis identificado como sistema coordinado de tiempo.

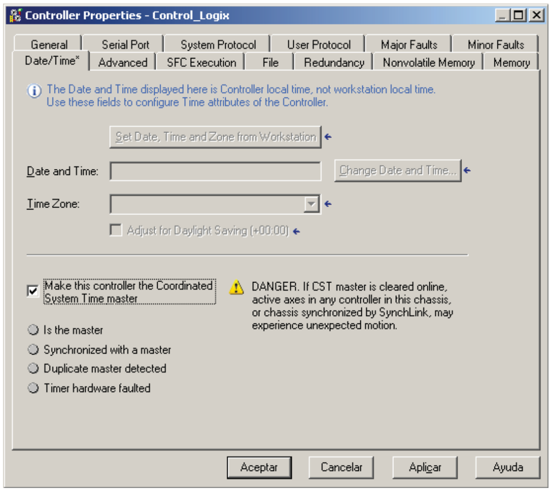
I. En el organizador del controlador, haga clic con el botón derecho del mouse en la carpeta ***Controller*** y seleccione ***Properties***.



Edit controller properties

Figura 47. Ver Propiedades del Controlador. [28]

II. Haga clic en la ficha Date/Time y seleccione la casilla de la opción ***Make this controller the Coordinated System Time master*** y aceptar los cambios.



The Date and Time displayed here is Controller local time, not workstation local time. Use these fields to configure Time attributes of the Controller.

☑ Make this controller the Coordinated System Time master

DANGER. If CST master is cleared online, active axes in any controller in this chassis, or chassis synchronized by SynchLink, may experience unexpected motion.

Figura 48. Seleccionar la Opción CST para el Controlador. [28]

3.3.3. Añadir el Módulo de Control de Movimiento.

I. Para añadir un módulo, en el organizador del controlador, haga clic con el botón derecho del mouse en la carpeta ***I/O Configuration*** y seleccione ***New Module.***

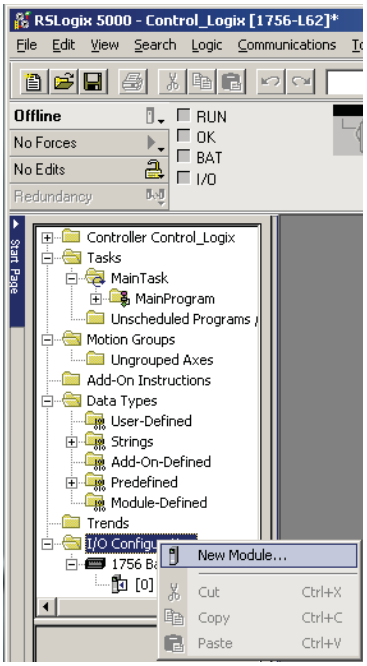
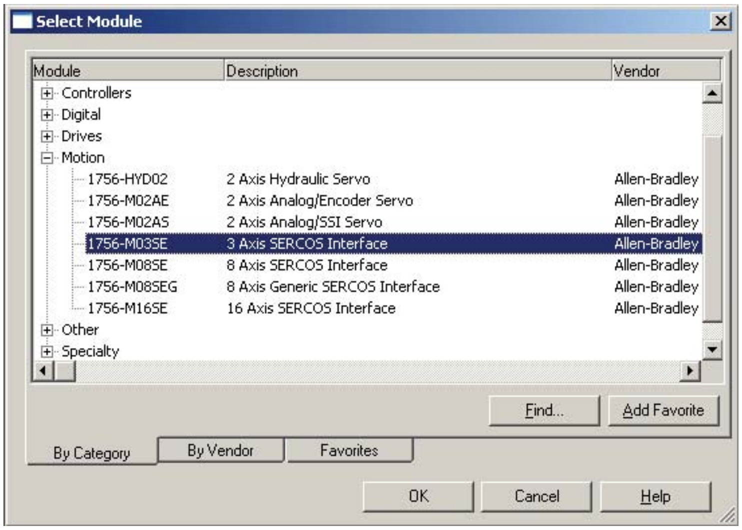


Figura 49. Añadir Nuevo Módulo al Proyecto. [28]

II. Seleccione el modulo a insertar en la ventana ***Select Module***, en este caso se tiene una interface SERCOS de 3 ejes referencia 1756-M03SEy luego de clic en ***OK***.

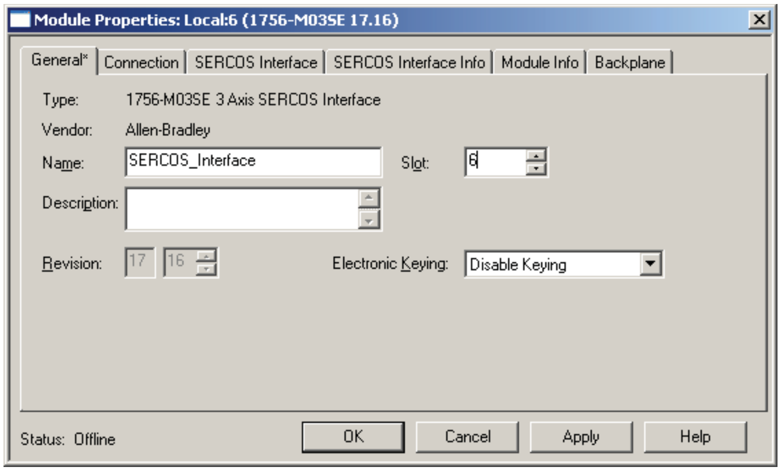


Module	Description	Vendor
Controllers		
Digital		
Drives		
Motion		
1756-HYD02	2 Axis Hydraulic Servo	Allen-Bradley
1756-M02AE	2 Axis Analog/Encoder Servo	Allen-Bradley
1756-M02AS	2 Axis Analog/SSI Servo	Allen-Bradley
1756-M03SE	3 Axis SERCOS Interface	Allen-Bradley
1756-M08SE	8 Axis SERCOS Interface	Allen-Bradley
1756-M08SEG	8 Axis Generic SERCOS Interface	Allen-Bradley
1756-M16SE	16 Axis SERCOS Interface	Allen-Bradley
Other		
Specialty		

Figura 50. Insertar Interface SERCOS. [28]

III. Dele un nombre al módulo.

IV. Asigne la posición (slot) donde se encuentra ubicado el módulo SERCOS, en este caso, el modulo está en la posición 6.



Type: 1756-M03SE 3 Axis SERCOS Interface
Vendor: Allen-Bradley
Name: SERCOS_Interface
Slot: 6
Description:
Revision: 17 16
Electronic Keying: Disable Keying
Status: Offline

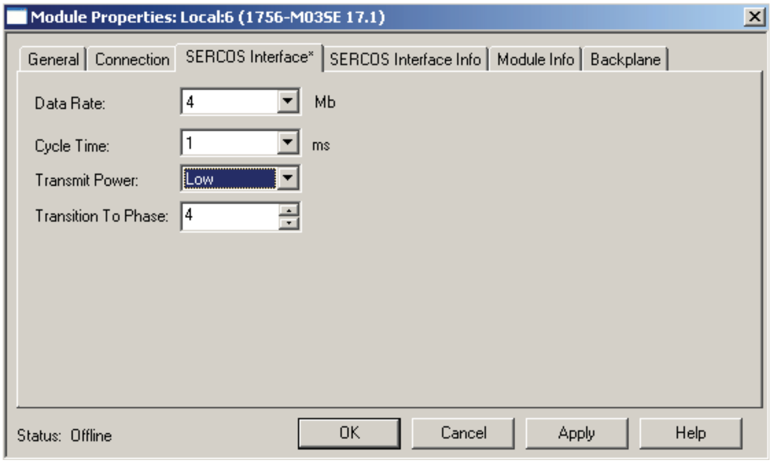
Figura 51. Asignación de Nombre y Posición del Módulo en Chasis. [28]

V. Si lo desea, ingrese una descripción del módulo.

VI. Elegir el tipo de ***Electronic Keying*** que desea utilizar, en este caso se utiliza la opción ***Disable Keying.***

Para comprender las propiedades de la opción ***Electronic Keying***, ver Anexo L de la página 23 a la 29.

VII. Diríjase a la pestaña ***SERCOS Interface***, seleccione 4 como ***Data Rate***, en ***Cycle Time*** seleccione 1, dado que este valor es sugerido por el fabricante cuando se trabaja a 4 Mb hasta con 4 drives (ver tabla 4) y en ***Transmit Power*** seleccione ***Low***, puesto que la distancia de transmisión es corta.



Data Rate: 4 Mb

Cycle Time: 1 ms

Transmit Power: Low

Transition To Phase: 4

Status: Offline

Figura 52. Configuración de la Interface SERCOS del Módulo. [28]

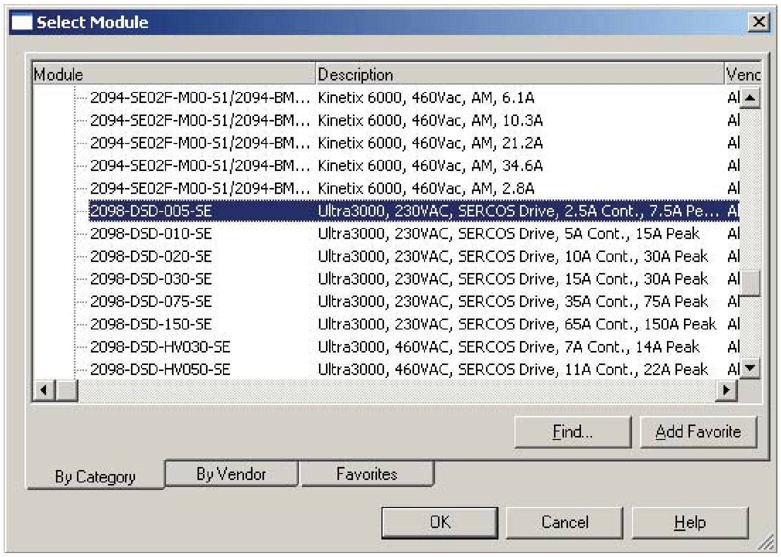
VIII. Seleccione ***OK*** para aplicar los cambios en la configuración del módulo SERCOS.

3.3.4. Añadir los Servo Drives

I. Para añadir un módulo, haga clic en la Red SERCOS (***Network SERCOS***) y seleccione Nuevo Módulo (***New Module***).

Figura 53. Insertar Servo Drive 1. [28]

II. Seleccione el módulo que desea agregar (2098-DSD-005-SE) y haga clic en ***OK***.

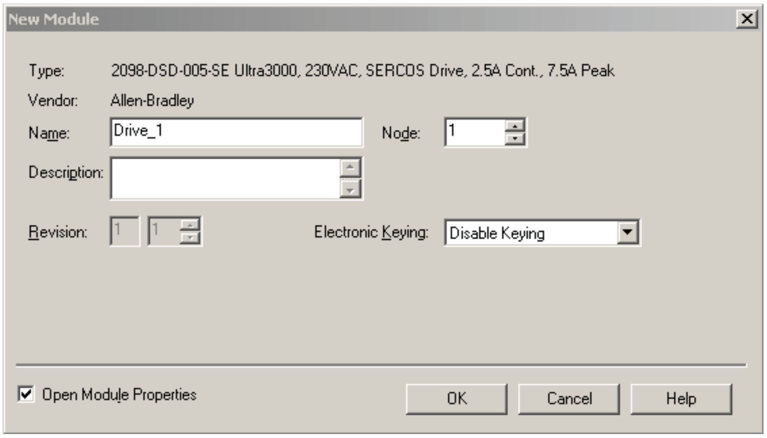


Module	Description	Venc
2094-SE02F-M00-S1/2094-BM...	Kinetix 6000, 460Vac, AM, 6.1A	Al
2094-SE02F-M00-S1/2094-BM...	Kinetix 6000, 460Vac, AM, 10.3A	Al
2094-SE02F-M00-S1/2094-BM...	Kinetix 6000, 460Vac, AM, 21.2A	Al
2094-SE02F-M00-S1/2094-BM...	Kinetix 6000, 460Vac, AM, 34.6A	Al
2094-SE02F-M00-S1/2094-BM...	Kinetix 6000, 460Vac, AM, 2.8A	Al
2098-DSD-005-SE	Ultra3000, 230VAC, SERCOS Drive, 2.5A Cont., 7.5A Pe...	Al
2098-DSD-010-SE	Ultra3000, 230VAC, SERCOS Drive, 5A Cont., 15A Peak	Al
2098-DSD-020-SE	Ultra3000, 230VAC, SERCOS Drive, 10A Cont., 30A Peak	Al
2098-DSD-030-SE	Ultra3000, 230VAC, SERCOS Drive, 15A Cont., 30A Peak	Al
2098-DSD-075-SE	Ultra3000, 230VAC, SERCOS Drive, 35A Cont., 75A Peak	Al
2098-DSD-150-SE	Ultra3000, 230VAC, SERCOS Drive, 65A Cont., 150A Peak	Al
2098-DSD-HV030-SE	Ultra3000, 460VAC, SERCOS Drive, 7A Cont., 14A Peak	Al
2098-DSD-HV050-SE	Ultra3000, 460VAC, SERCOS Drive, 11A Cont., 22A Peak	Al

Figura 54. Seleccionar el Servo Drive Utilizado. [28]

III. Darle un nombre al módulo, asignar el número de nodo del servo drive. El número de nodo debe coincidir con la dirección asignada de manera manual al drive; en este caso es 1.

IV. Seleccionar ***Disable Keying*** en la casilla ***Electronic Keying,*** e ingresar alguna descripcion en ***Description*** si es necesario, dar clic en ***OK*** para terminar la configuración del módulo. Para añadir el segundo servo drive se siguen los mismos pasos anteriores, pero el nombre del módulo será Drive_2 y el nodo en este caso será 2.



Type: 2098-DSD-005-SE Ultra3000, 230VAC, SERCOS Drive, 2.5A Cont., 7.5A Peak
Vendor: Allen-Bradley
Name: Drive_1
Node: 1
Description:
Revision: 1 1
Electronic Keying: Disable Keying
☑ Open Module Properties

Figura 55. Configuración de Servo Drive 1. [28]

3.3.5. Crear un Grupo de Control de Movimiento

I. Haga clic con el botón derecho del mouse en la carpeta ***Motion Groups*** y seleccione ***New Group***.

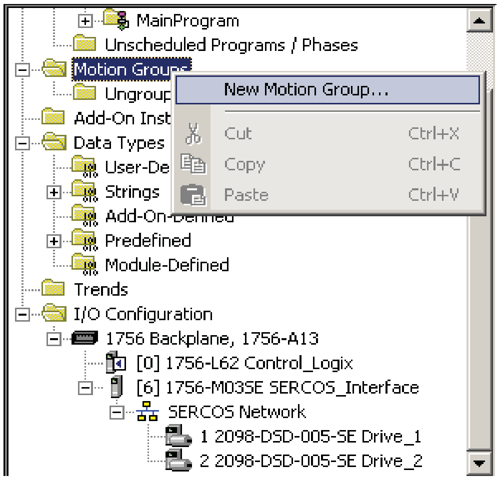
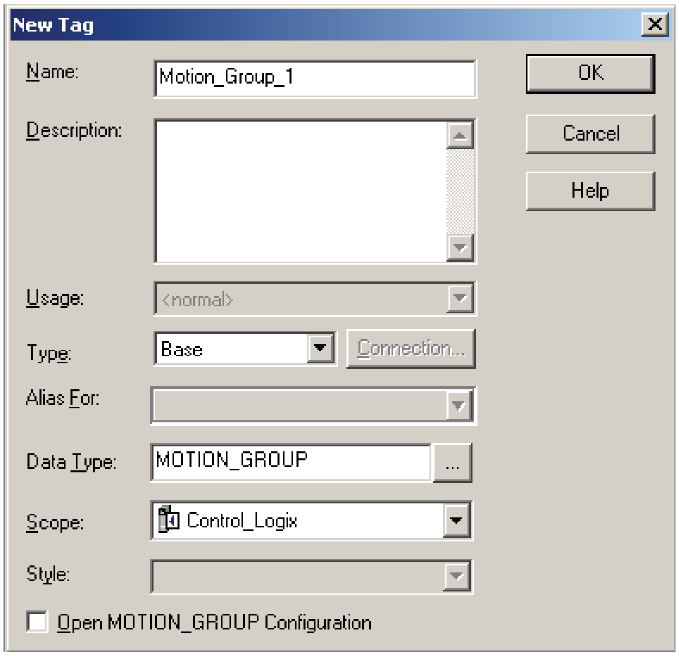


Figura 56. Crear un Nuevo Grupo de Movimiento. [28]

II. Darle un nombre al grupo y una descripción si se desea, de clic en ***OK*** para guardar los cabios.



Name: Motion_Group_1
Description:
Usage: <normal>
Type: Base
Alias For:
Data Type: MOTION_GROUP
Scope: Control_Logix
Style:
☐ Open MOTION_GROUP Configuration

Figura 57. Configuración del Grupo de Control de Movimiento. [28]

3.3.6. Añadir un Eje

I. Haga clic con el botón derecho del mouse en la carpeta ***Motion_Group_1*** (grupo de movimiento creado con anterioridad) y seleccione ***New Axis***.

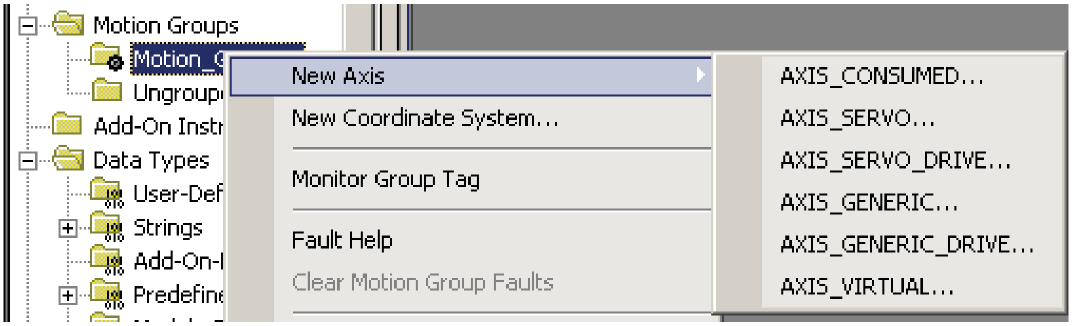


Figura 58. Añadir un nuevo eje (***Axis***). [28]

Si se usa alguno de los siguientes módulos de movimiento, entonces utilice el tipo de dato ***AXIS_SERVO_DRIVE*** para el eje.

- 1756-M03SE
- 1756-M08SE
- 1756-M16SE
- 1768-M04SE

II. Escoja la opción ***AXIS_SERVO_DRIVE*** pues el módulo de movimiento usado es el *1756-M03SE.*

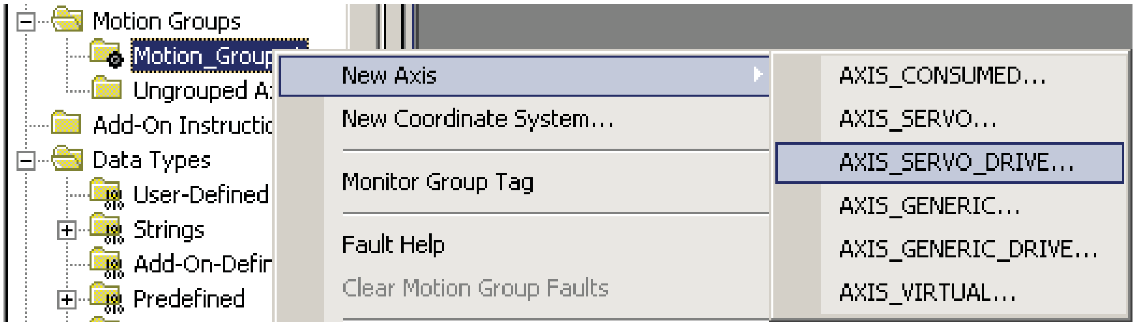
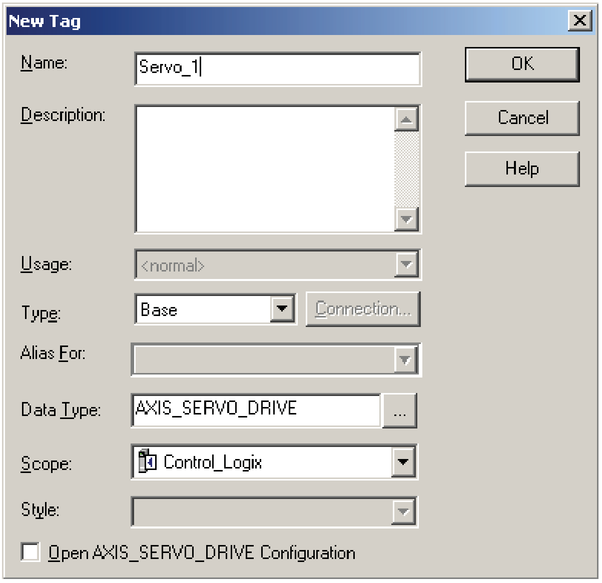


Figura 59. Insertar un ***AXIS_SERVO_DRIVE.*** [28]

III. Asígnele un nombre al eje (Servo_1 en este caso) en la ventana ***New Tag*** y seleccione ***OK*** para guardar los cambios. Añada un segundo axis de la misma forma que el anterior dele un nombre (Servo_2).


New Tag

Name: Servo_1

Description:

Usage: <normal>

Type: Base

Alias For:

Data Type: AXIS_SERVO_DRIVE

Scope: Control_Logix

Style:

☐ Open AXIS_SERVO_DRIVE Configuration

Figura 60. Asignación de Nombre al Nuevo Eje. [28]

3.3.7. Configurar un Eje

I. Haga clic con el botón derecho del mouse en la carpeta del eje 1 (Servo_1) y seleccione la opción ***Properties.***

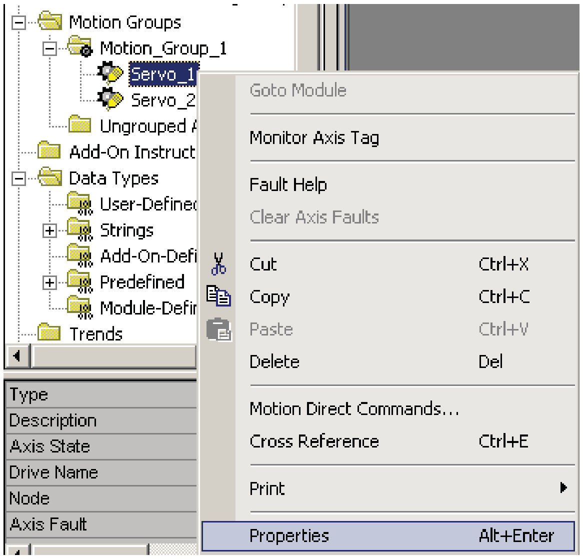
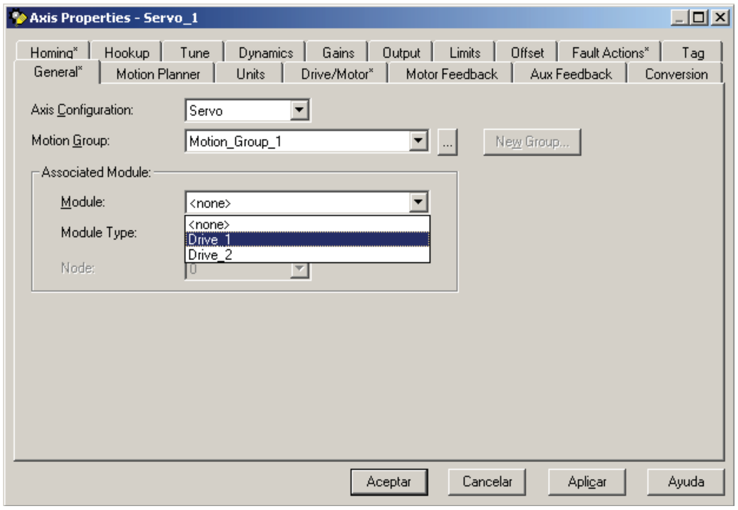


Figura 61. Abrir el Menú de Propiedades del Eje 1. [28]

II. Hacer clic en la pestaña ***General.***

Seleccionar el modulo asociado con el eje. Para el eje 2 (Servo_2) asociar el módulo de movimiento 2 (Servo 2) en este punto.



Axis Properties - Servo_1

Axis Configuration: Servo

Motion Group: Motion_Group_1

Associated Module:

Module: <none>

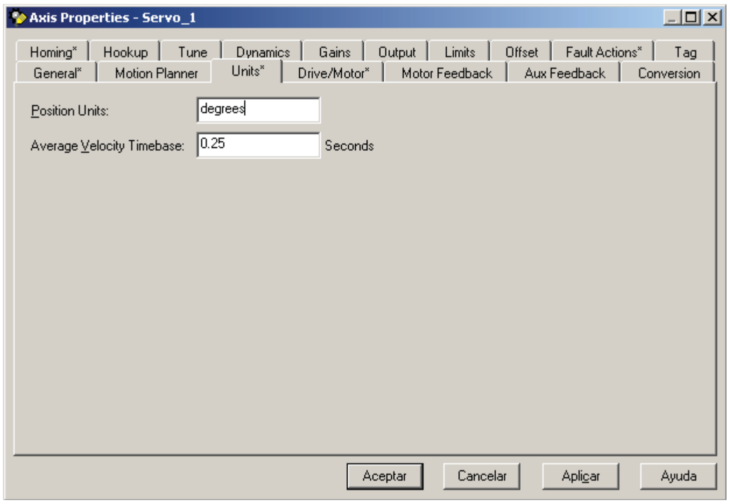
<none>
Drive_1
Drive_2

Module Type:

Node: 0

Figura 62. Relacionar el Eje 1 con el Módulo de Movimiento 1. [28]

III. Remítase a la pestaña ***Units*** e ingrese el tipo de unidades y la base de tiempo promedio de velocidad que desea utilizar.



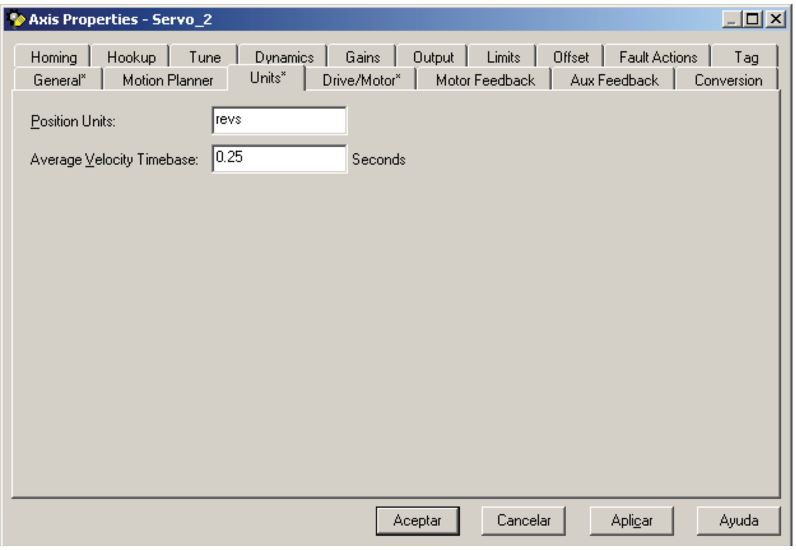
Axis Properties - Servo_1

Position Units: degrees

Average Velocity Timebase: 0.25 Seconds

Figura 63. Definición de Unidades y Base de Tiempo Promedio Eje 1. [28]

Con el eje 1 se trabaja posición, por tal motivo las unidades se trabajan en grados (degrees) y la base de tiempo promedio de velocidad (***Average Velocity Timebase***) se dejara en el valor por defecto (0.25).



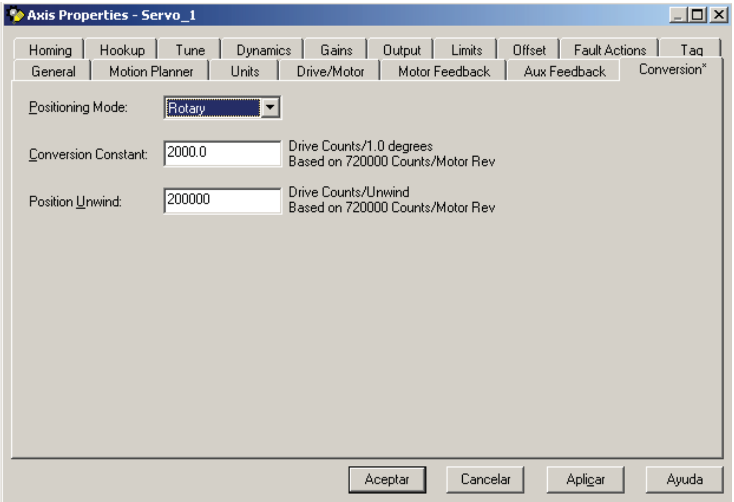
Position Units: revs

Average Velocity Timebase: 0.25 Seconds

Figura 64. Definición de Unidades y Base de Tiempo Promedio Eje 2. [28]

Con el eje 2 (Servo_2) se trabaja velocidad, por tanto, las unidades se traba en revoluciones (revs) y ***Average Velocity Timebase*** se trabaja con su valor por defecto.

IV. Haga clic sobre la pestaña ***Conversion*** seleccione el tipo de posicionamiento que desea usar. En este caso se está trabajando con motores rotativos seleccionar la opción ***Rotary*** y luego dar clic en ***Aplicar*** para guardar los cambios.



Positioning Mode: Rotary

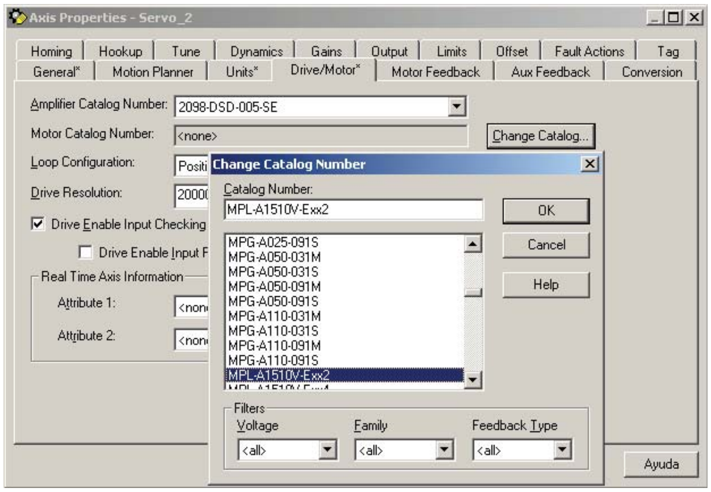
Conversion Constant: 2000.0 Drive Counts/1.0 degrees Based on 720000 Counts/Motor Rev

Position Unwind: 200000 Drive Counts/Unwind Based on 720000 Counts/Motor Rev

Figura 65. Selección del Modo de Posicionamiento a Usar. [28]

Esta configuración se hace de la misma forma para ambos ejes.

V. Ir a la pestaña Drive/Motor, seleccionar el número de catálogo del amplificador y el motor relacionados al eje.



Amplifier Catalog Number: 2098-DSD-005-SE
Motor Catalog Number: <none>
Loop Configuration: Positi
Drive Resolution: 20000
☑ Drive Enable Input Checking
☐ Drive Enable Input F
Attribute 1: <non
Attribute 2: <non

Catalog Number: MPL-A1510V-Exx2

MPG-A025-091S
MPG-A050-031M
MPG-A050-031S
MPG-A050-091M
MPG-A050-091S
MPG-A110-031M
MPG-A110-031S
MPG-A110-091M
MPG-A110-091S
MPL-A1510V-Exx2

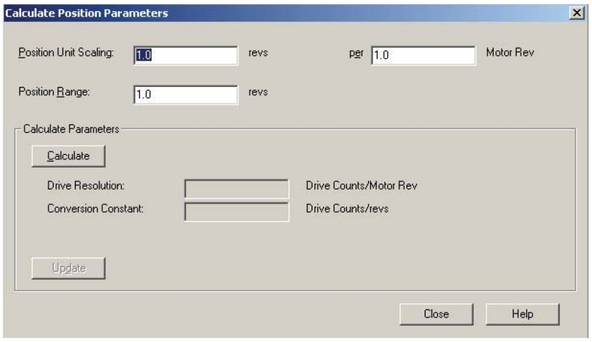
Voltage: <all>
Family: <all>
Feedback Type: <all>

Figura 66. Seleccionar Amplificador y Motor relacionados al eje 1. [28]

En este numeral la configuración del eje 1 y 2 es la misma, dado que ambos ejes se relacionan con un amplificador 2098-DSD-005-SE y un motor MPL-1510V-EJ42AA.

VI. En la opcion ***Loop Configuration*** se puede seleccionar la opción ***Position Servo*** o la opción **Dual Commad Servo**, dado a que estas dos permiten realizar el contro de posición y velocidad utilizando únicamente el dispositivo Feedback integrado en el motor. Para ver el diagrama de bloques de cada uno de estas opciones de lazo, ver Anexo L páginas 365 y 368.

VII. Para calcular la resolución que el drive debe manejar, se debe dar clic en ***Calculate,*** se abrirá una ventana llamada ***Calculate Position Parameters.***

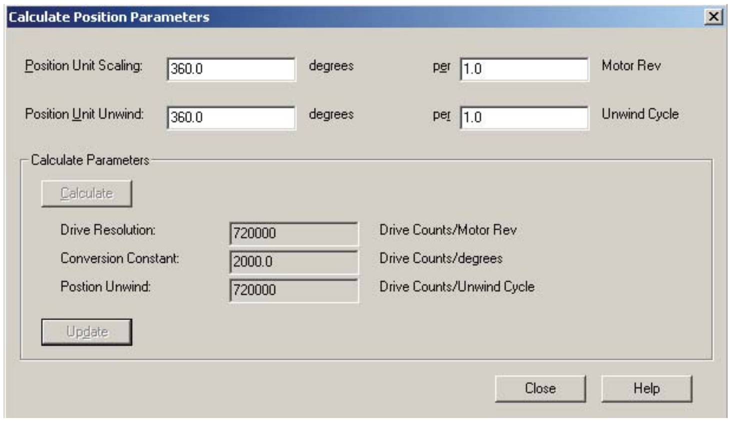


Position Unit Scaling: 1.0 revs per 1.0 Motor Rev
Position Range: 1.0 revs
Drive Resolution: Drive Counts/Motor Rev
Conversion Constant: Drive Counts/revs

Figura 67. Ventana Calculate Position Parameters. [28]

VIII. Como el eje 1 se trabaja como un eje de posición, con unidades de grados, se deberán poner los siguientes datos para hacer el cálculo correcto:
Position Unit Scaling: 360 degrees per 1.0 Motor Rev
Position Unit Unwind: 360 degrees per 1.0 Unwind Cycle

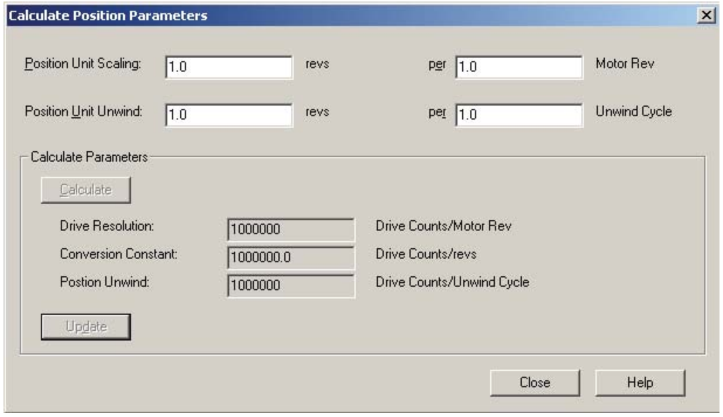
Una vez ingrese estos valores dar clic en calcular (***Calcuate***), para que el sistema calcule los valores de ***Drive Resolution*** y ***Conversion Constant***. Para conservar los valores calculados dar clic en ***Update*** y luego en ***Close***.



Position Unit Scaling: 360.0 degrees per 1.0 Motor Rev
Position Unit Unwind: 360.0 degrees per 1.0 Unwind Cycle
Drive Resolution: 720000 Drive Counts/Motor Rev
Conversion Constant: 2000.0 Drive Counts/degrees
Postion Unwind: 720000 Drive Counts/Unwind Cycle

Figura 68. Calculo de Parámetros de Posición para el Eje 1. [28]

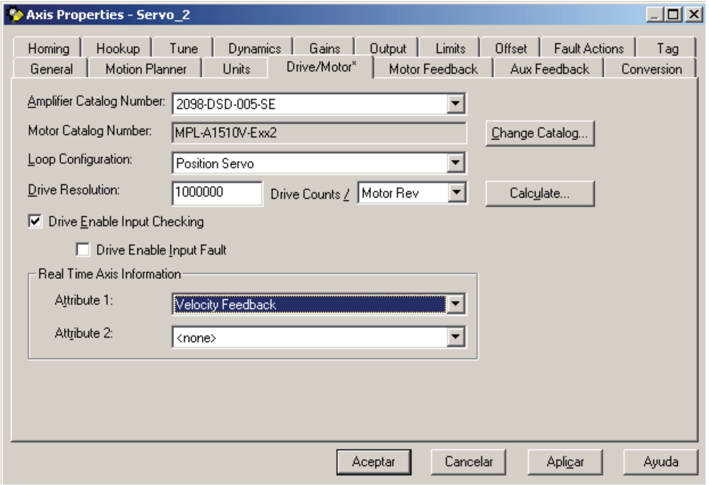
Para el eje 2 la unidad de escalado de posición será 1 por 1 revolución del motor y el rango de posición será 1 también, esto debido a que se trabaja como un eje de velocidad.



Position Unit Scaling: 1.0 revs per 1.0 Motor Rev
Position Unit Unwind: 1.0 revs per 1.0 Unwind Cycle
Drive Resolution: 1000000 Drive Counts/Motor Rev
Conversion Constant: 1000000.0 Drive Counts/revs
Postion Unwind: 1000000 Drive Counts/Unwind Cycle

Figura 69. Calculo de Parámetros de Posición para el Eje 2. [28]

IX. Ahora especificar que atributo se desea tener en cuenta para la información del eje en tiempo real, para el eje 1 el ***Attribute 1*** será ***Position Feedback*** y para el eje 2 será ***Velocity Feedback***, el ***Atribute 2*** se dejara en blanco, dado a que con la instrucción GSV se puede acceder a cualquier información del drive o el motor.



Amplifier Catalog Number: 2098-DSD-005-SE

Motor Catalog Number: MPL-A1510V-Exx2

Loop Configuration: Position Servo

Drive Resolution: 1000000 Drive Counts / Motor Rev

☑ Drive Enable Input Checking

☐ Drive Enable Input Fault

Real Time Axis Information

Attribute 1: Velocity Feedback

Attribute 2: <none>

Figura 70. Real Time Axis Information Attribute Eje 2. [28]

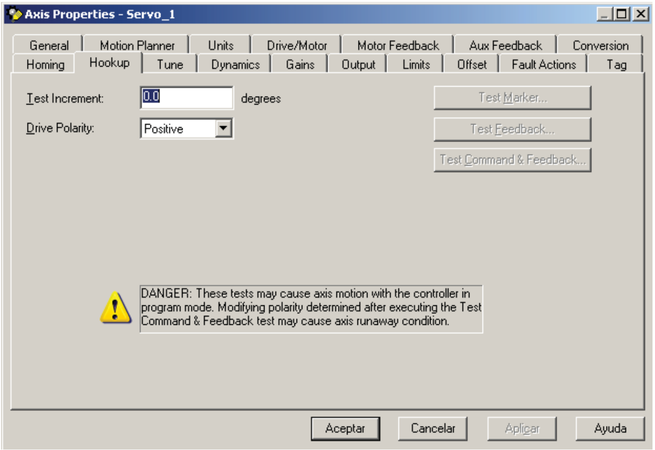
3.3.8. Ejecutar las Pruebas de Conexión

I. Descargar el proyecto al controlador.

II. Coloque el controlador en ***Remote Mode***.

III. Haga doble clic sobre el eje que desea testear para abrir sus propiedades.

IV. Remítase a la pestaña ***Hookup***.



Test Increment: 0.0 degrees

Drive Polarity: Positive

DANGER: These tests may cause axis motion with the controller in program mode. Modifying polarity determined after executing the Test Command & Feedback test may cause axis runaway condition.

Figura 71. Ventana para Test de Conexión del Eje. [28]

V. Ingrese el valor de incremento para el test (***Test Increment***).

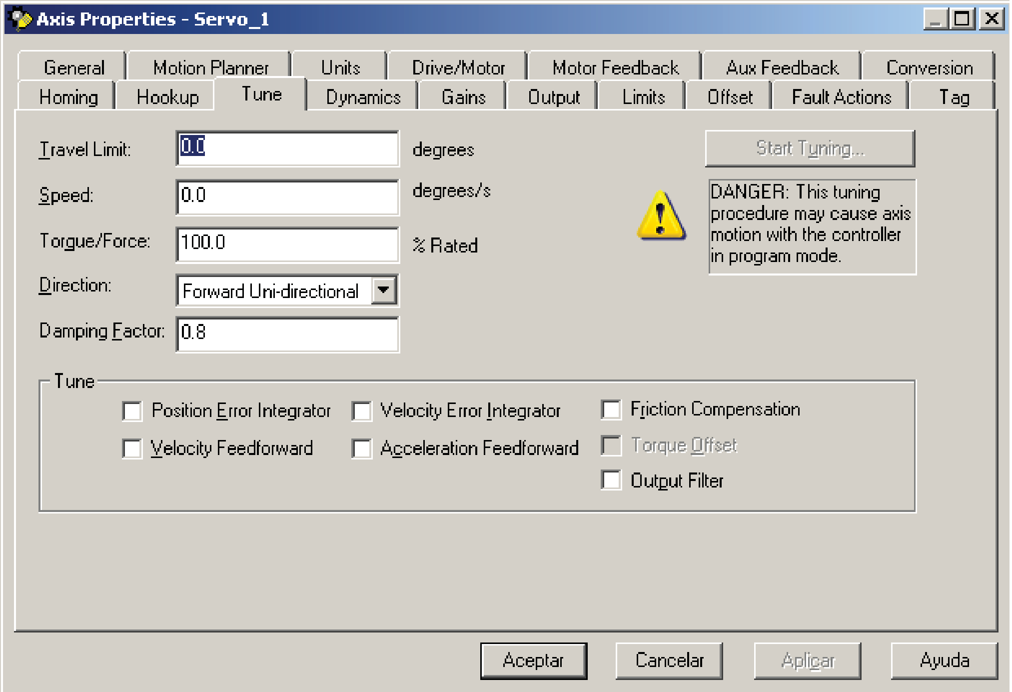
VI. Haga clic sobre el botón ***Test Marker*** y siga las instrucciones que se muestran.

VII. Haga clic sobre el botón ***Test Feedback*** y siga las instrucciones que se muestran.

VIII. Haga clic sobre el botón ***Test Command & Feedback*** y siga las instrucciones que se muestran.

3.3.9. Sintonice un Eje SERCOS

I. Haga clic en la pestaña ***Tune.***



Travel Limit: 0.0 degrees

Speed: 0.0 degrees/s

Torque/Force: 100.0 % Rated

Direction: Forward Uni-directional

Damping Factor: 0.8

DANGER: This tuning procedure may cause axis motion with the controller in program mode.

Tune

☐ Position Error Integrator ☐ Velocity Error Integrator ☐ Friction Compensation

☐ Velocity Feedforward ☐ Acceleration Feedforward ☐ Torque Offset

☐ Output Filter

Figura 72. Ventana de Sintonización del Eje. [28]

II. Ajustar el ***Trabel Limit*** (este es el límite de movimiento del eje durante el procedimiento de ajuste).

III. Ajustar la velocidad (esta debe ser la velocidad máxima con la que se trabajara el equipo).

IV. Hacer clic en el botón ***Start Tuning***.

V. Aceptar los cambios establecidos para que se guarden en la configuración del eje.

3.3.10. Pruebe un Eje con Motion Direct Commands

Los ***Motion Direct Commands*** permiten emitir comandos de movimiento mientras está en línea con el controlador, sin tener que escribir o ejecutar un programa de aplicación. [29]

Los ***Motion Direct Commands*** son particularmente útiles cuando se está poniendo en marcha o resolviendo problemas de una aplicación de movimiento. Durante la puesta en marcha, puede configurar un eje y monitorear el comportamiento mediante las tendencias incluidas en el ***Controller Organizer***. Mediante ***Motion Direct Commands*** puede ajustar de manera precisa el sistema con o sin carga para optimizar su rendimiento. En el estado de prueba o de resolución de problemas, puede emitir ***Motion Direct Commands*** para establecer o restablecer condiciones tales como Home. Generalmente, durante el desarrollo inicial o durante la ejecución de mejoras a aplicaciones existentes, usted necesita probar el sistema por áreas pequeñas manejables. Estas tareas incluyen lo siguiente: [29]

- Inicio para establecer condiciones iniciales.
- Movimiento incremental a una posición física.
- Monitorear dinámica del sistema bajo condiciones específicas.

Para obtener acceso a ***Motion Direct Commands*** para el grupo de movimiento o eje, haga clic con el botón derecho del mouse en el grupo o eje en el ***Controller Organizer.*** [29]

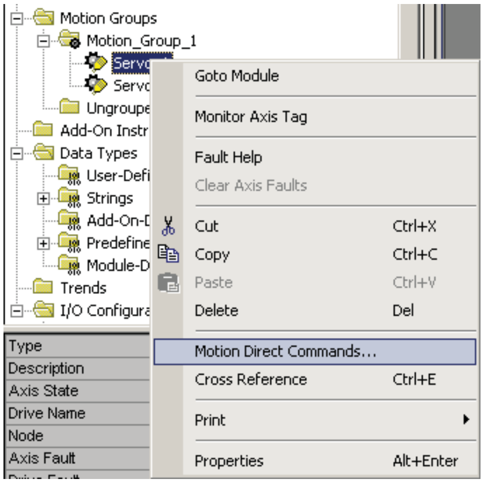
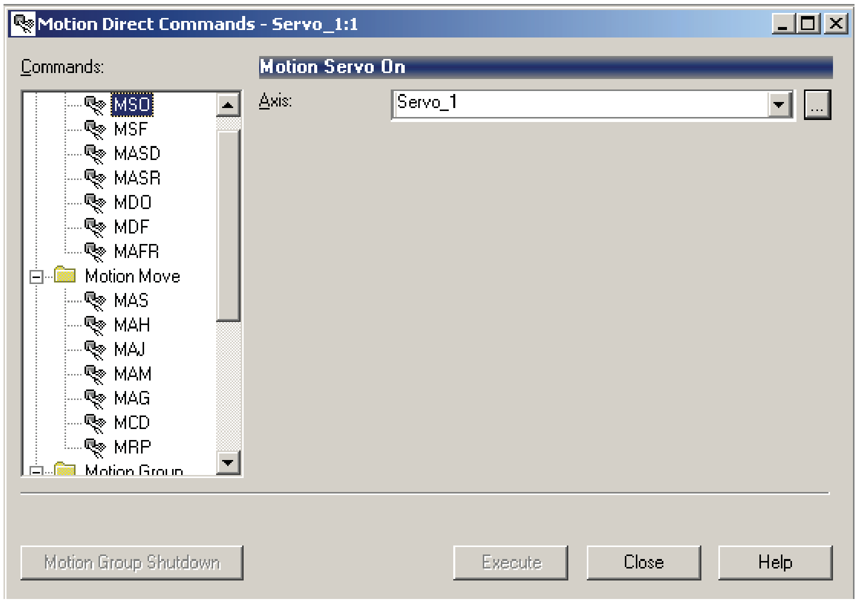


Figura 73. Acceso a los Motion Direct Commands de un eje.

La ventana ***Motion Direct Command*** variará según el comando que usted haya seleccionado. En la lista Command, usted puede escribir el mnemónico y la lista avanza a la igualdad más cercana o puede desplazarse hacia abajo por la lista para seleccionar un comando. Haga clic en el comando deseado y aparecerá su cuadro de diálogo. [29]



Motion Servo On

Axis: Servo_1

Figura 74. Ventana de Motion Direct Commands.

3.3.11. Desarrollar la Lógica para el Control de Movimiento.

Para elaborar un programa de control de movimiento se debe hacer uso de las instrucciones de control de movimiento, insertándolas directamente en el programa de lógica de escalera (LD), texto estructurado (ST) o diagrama de función secuencial (SFC). [30] [31]

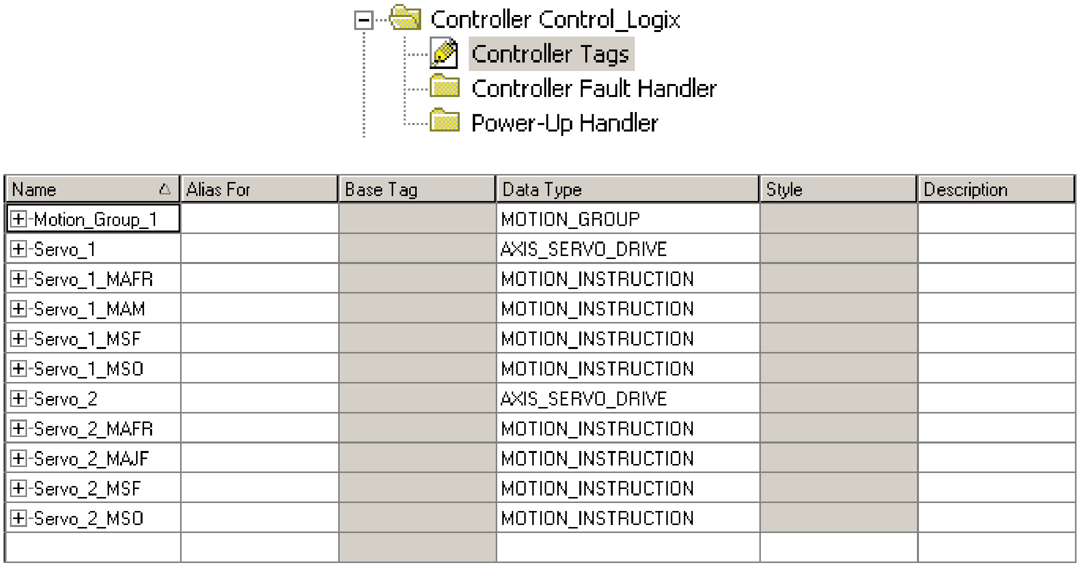
Las instrucciones de control de movimiento efectúan operaciones en uno o más ejes. Antes de usar estas instrucciones se deben haber configurado cada uno de los ejes del sistema tal y como se ha hecho en los numerales anteriores. [xz]

Cada instrucción de movimiento posee un ***Tag*** de control de movimiento, este ***Tag*** usa un tipo de dato ***MOTION_INSTRUCTION*** para almacenar toda la información de estado durante la ejecución de la instrucción. [30]

A continuación se muestra la lógica implementada para el desarrollo de la aplicación.

Para la implementación del sistema se realizó una interface gráfica que le permite al usuario interactuar directamente con los servomotores a través de instrucciones simples.

En primera instancia se crearon los Tags tipo ***Motion_ Instruction*** para cada una de las instrucciones que se utilizaron (MAFR, MAJ, MAM, MSF, MSO). Estos Tags se crean en los Tags del controlador (***Controller Tags***), donde se encuentran los Tags tipo ***Motion_Group*** y ***Axis_Servo_Drive*** los cuales se crearon automáticamente en el transcurso de la configuración de los ejes y el grupo de movimiento. Los tags del controlador son variables de tipo global, disponibles para todos los programas que posea controlador.

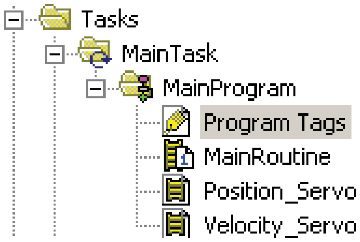


Controller Control_Logix
Controller Tags
Controller Fault Handler
Power-Up Handler

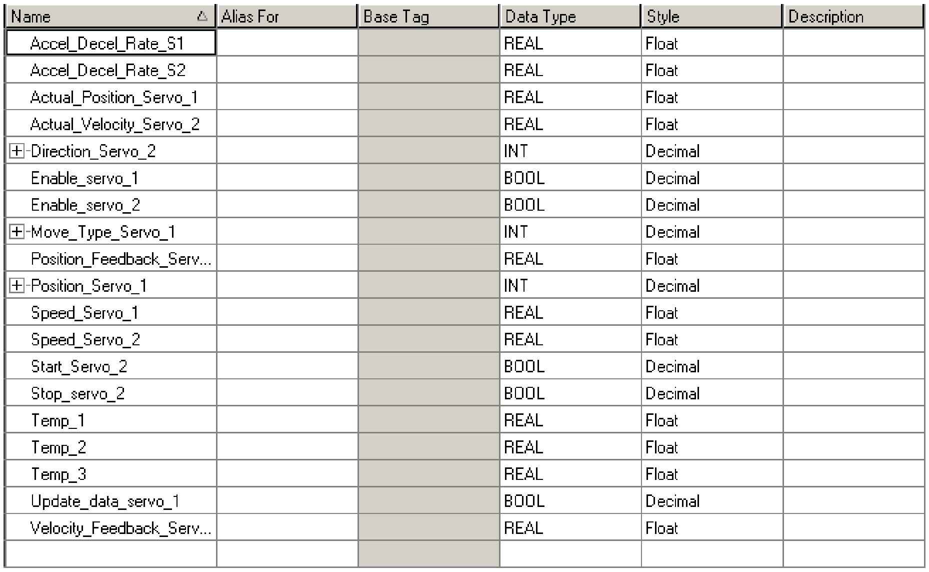
Name	Alias For	Base Tag	Data Type	Style	Description
Motion_Group_1			MOTION_GROUP		
Servo_1			AXIS_SERVO_DRIVE		
Servo_1_MAFR			MOTION_INSTRUCTION		
Servo_1_MAM			MOTION_INSTRUCTION		
Servo_1_MSF			MOTION_INSTRUCTION		
Servo_1_MSO			MOTION_INSTRUCTION		
Servo_2			AXIS_SERVO_DRIVE		
Servo_2_MAFR			MOTION_INSTRUCTION		
Servo_2_MAJF			MOTION_INSTRUCTION		
Servo_2_MSF			MOTION_INSTRUCTION		
Servo_2_MSO			MOTION_INSTRUCTION		

Figura 75. *Controller Tags* del proyecto de movimiento.

En segunda instancia se crean todas las Tags necesarias para la interacción de la HMI con el controlador, la ejecución de cálculos y la correcta operación de los comandos de movimiento. Estas Tags se crean en las tags de programa (***Program Tags***) y son variables locales que solo están disponibles dentro del programa.



Tasks
MainTask
MainProgram
Program Tags
MainRoutine
Position_Servo
Velocity_Servo

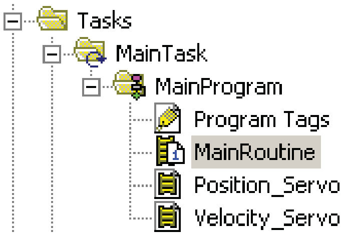


Name	Alias For	Base Tag	Data Type	Style	Description
Accel_Decel_Rate_S1			REAL	Float	
Accel_Decel_Rate_S2			REAL	Float	
Actual_Position_Servo_1			REAL	Float	
Actual_Velocity_Servo_2			REAL	Float	
⊞Direction_Servo_2			INT	Decimal	
Enable_servo_1			BOOL	Decimal	
Enable_servo_2			BOOL	Decimal	
⊞Move_Type_Servo_1			INT	Decimal	
Position_Feedback_Serv...			REAL	Float	
⊞Position_Servo_1			INT	Decimal	
Speed_Servo_1			REAL	Float	
Speed_Servo_2			REAL	Float	
Start_Servo_2			BOOL	Decimal	
Stop_servo_2			BOOL	Decimal	
Temp_1			REAL	Float	
Temp_2			REAL	Float	
Temp_3			REAL	Float	
Update_data_servo_1			BOOL	Decimal	
Velocity_Feedback_Serv...			REAL	Float	

Figura 76. *Program Tags* del proyecto de movimiento.

El programa se realizó utilizando lógica en escalera; posee una rutina principal y dos subrutinas, que se describirán a continuación:

- **Rutina Principal (*MainRoutine*)**



La rutina principal es la rutina que se ejecuta cuando el controlador esta en modo RUN, la lógica que está programada allí se ejecutara secuencialmente hasta que llega a la línea final y se repite continuamente, mientras el controlador no se ponga en modo STOP.

En el proyecto esta rutina principal tiene 3 líneas:

Línea 0: Esta linea posee una istruccion JSR, la cual permite ejecutar un salto a la subrutina *Position_Servo* y ejecutar la logica que alli recide.

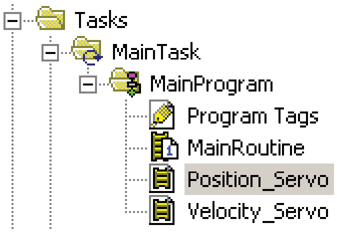
Línea 1: Esta linea posee una istruccion JSR, la cual permite ejecutar un salto a la subrutina *Velocity_Servo* y ejecutar la logica que alli recide.


JSR
Jump To Subroutine
Routine Name Velocity_Servo

Línea 2: Esta línea marca el final del programa secuencial (END).

A continuación se explican las subrutinas (*Subrutines*):

- **Position Servo**

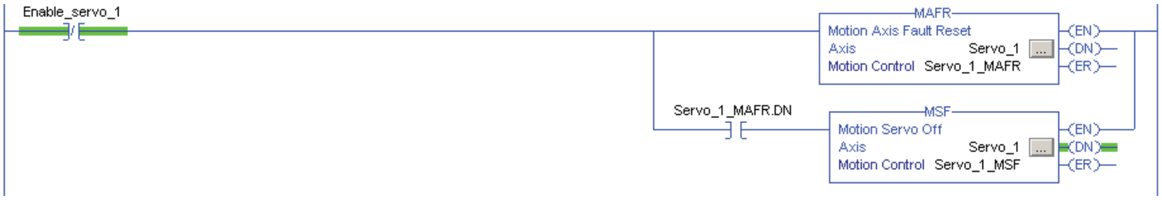

- Tasks
  - MainTask
    - MainProgram
      - Program Tags
      - MainRoutine
      - Position_Servo
      - Velocity_Servo

Esta rutina permite hacer el control sobre el eje 1 (Servo_1), que en este caso se ha destinado para mover el motor a una determinada posición en grados. Esta subrutina cuenta con 5 líneas:

Línea 0: Esta línea permite habilitar el servomotor y lo deja listo para ejecutar cualquier orden de movimiento. La instrucción MSO se ejecuta cuando el contacto Enable_servo_1 se habilita. Este contacto corresponde a la llave selectora de dos posiciones para la habilitación del servo drive ubicada en la ventana de control de posición de la HMI.


Enable_servo_1
MSO
Motion Servo On
Axis Servo_1
Motion Control Servo_1_MSO

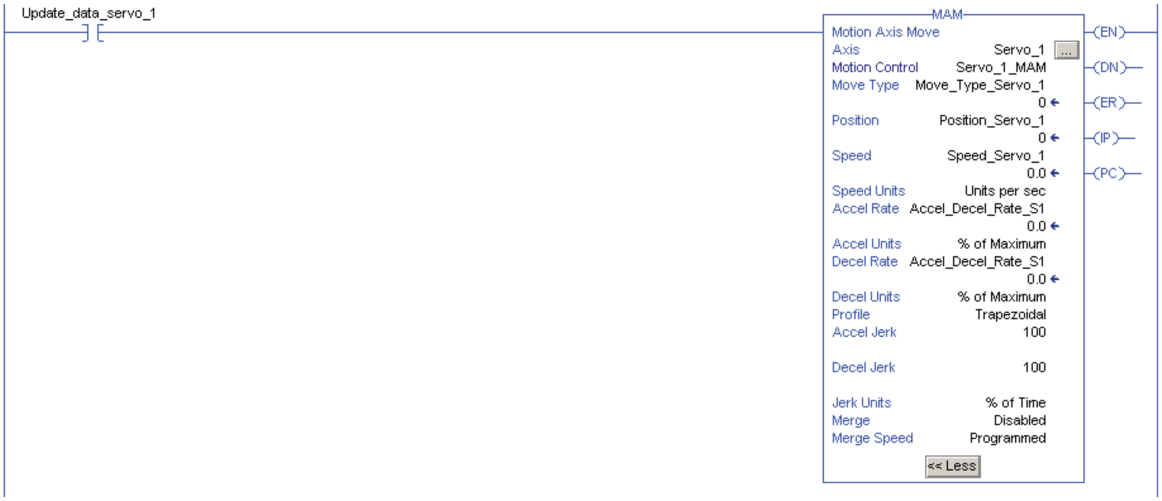
Línea 1: Esta línea permite hacer un reseteo de las fallas de movimiento presentes durante la operación del sistema mediante la instrucción MAFR y también permite deshabilitar el servomotor mediante la instrucción MSF, una vez que la instrucción MAFR se halla ejecutado. Estas dos instrucciones se ejecutan cuando el contacto Enable_servo_1 tiene un valor lógico 0, dado a que el contacto es un contacto normalmente cerrado (lógica negativa).



MAFR
Motion Axis Fault Reset
Axis Servo_1
Motion Control Servo_1_MAFR

MSF
Motion Servo Off
Axis Servo_1
Motion Control Servo_1_MSF

Línea 2: Esta línea permite mover el eje del servomotor 1 a una posición específica en grados (Operando Position) con una velocidad (Operando Speed) y una aceleración/desaceleración (Operando Accel/Decel Rate) definida por el usuario en la HMI. Existen cuatro tipos de movimiento (Operando Move Type), que son: Absoluto, Incremental, Horario y Antihorario. El tipo de movimiento deseado se elige en la HMI a través de una llave selectora de cuatro posiciones. La instrucción MAM se ejecuta cuando el contacto Update_data_servo_1 se habilite, este contacto corresponde al pulsador Ejecutar ubicado en la pantalla de control de posición en la HMI.

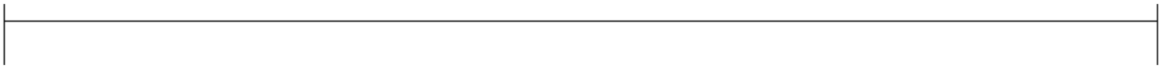


MAM
Motion Axis Move
Axis Servo_1
Motion Control Servo_1_MAM
Move Type Move_Type_Servo_1
0
Position Position_Servo_1
0
Speed Speed_Servo_1
0.0
Speed Units Units per sec
Accel Rate Accel_Decel_Rate_S1
0.0
Accel Units % of Maximum
Decel Rate Accel_Decel_Rate_S1
0.0
Decel Units % of Maximum
Profile Trapezoidal
Accel Jerk 100
Decel Jerk 100
Jerk Units % of Time
Merge Disabled
Merge Speed Programmed

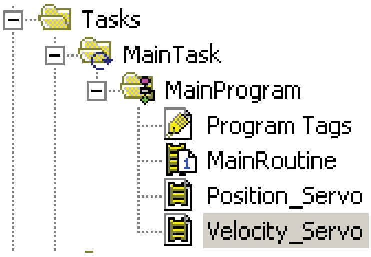
Línea 3: Esta línea ejecuta constantemente la instrucción GSV, la cual permiten obtener la posición actual del servo para poder presentarlos en la HMI.

GSV
Get System Value
Class Name Axis
Instance Name Servo_1
Attribute Name ActualPosition
Dest Actual_Position_Servo_1
0.0

Línea 4: Esta línea marca el final de la subrutina (END).



Velocity Servo

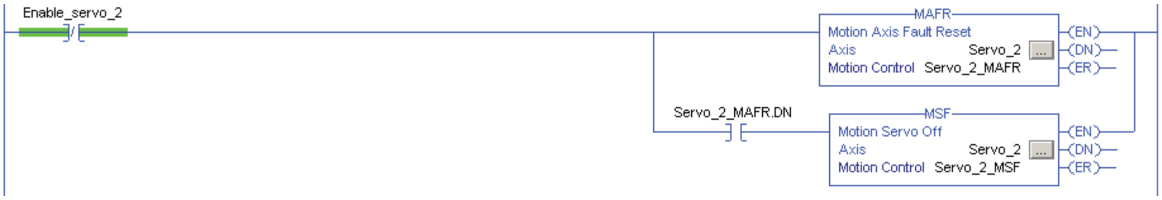


Esta rutina permite hacer el control sobre el eje 2 (Servo_2), que en este caso se ha destinado para mover el motor a una determinada velocidad. Esta subrutina cuenta con 7 líneas:

Línea 0: Esta línea permite habilitar el servomotor y lo deja listo para ejecutar cualquier orden de movimiento. La instrucción MSO se ejecuta cuando el contacto Enable_servo_2 se habilita. Este contacto corresponde a la llave selectora de dos posiciones para la habilitación del servo drive ubicada en la ventana de control de velocidad de la HMI.

Motion Servo On
Axis Servo_2
Motion Control Servo_2_MSO

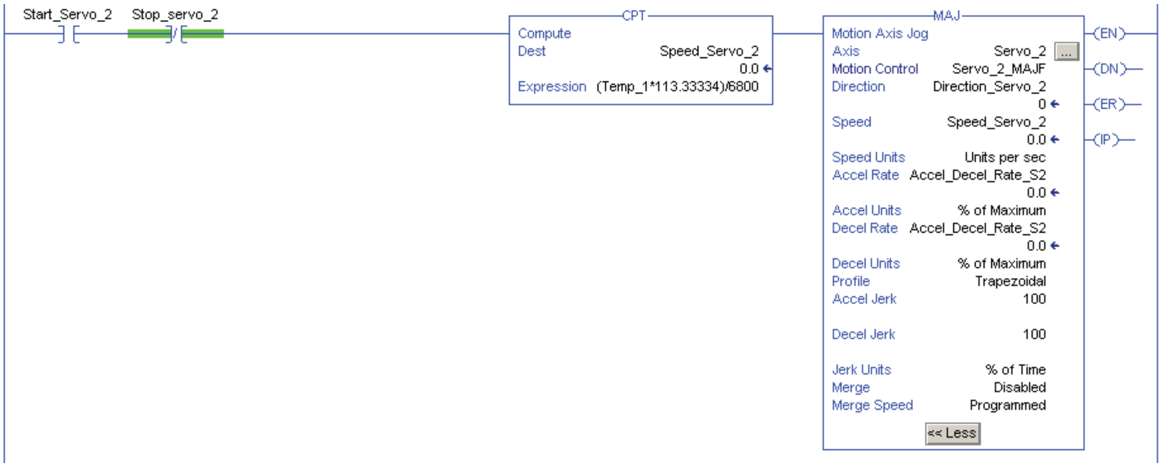
Línea 1: Esta línea permite hacer un reseteo de las fallas de movimiento presentes durante la operación del sistema mediante la instrucción MAFR y también permite deshabilitar el servomotor mediante la instrucción MSF, una vez que la instrucción MAFR se halla ejecutado. Estas dos instrucciones se ejecutan cuando el contacto Enable_servo_2 tiene un valor lógico 0, dado a que el contacto es un contacto normalmente cerrado (lógica negativa).



Motion Axis Fault Reset
Axis Servo_2
Motion Control Servo_2_MAFR

Motion Servo Off
Axis Servo_2
Motion Control Servo_2_MSF

Línea 2: Esta línea permite mover el servomotor 2 a una velocidad específica (Operando Speed), con un sentido de giro (Operando Direction) y una aceleración definida (Operando Accel/Decel Rate) por el usuario en la HMI. La instrucción MAJ se ejecuta siempre y cuando el contacto Start_Servo_2 este activo (1 lógico) y el Stop_Servo_2 no este activo (0 lógico). La instrucción CPT permite hacer el ajuste del dato de velocidad ingresado a través de la HMI, pues el usuario ingresa datos los datos de velocidad en RPM, mientras que el operando de velocidad de la instrucción MAJ recibe datos en revoluciones por segundo.


Start_Servo_2
Stop_servo_2

CPT
Compute
Dest Speed_Servo_2
0.0
Expression (Temp_1*113.33334)/6800

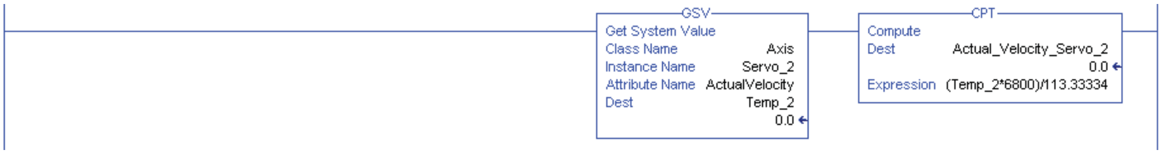
MAJ
Motion Axis Jog
Axis Servo_2
Motion Control Servo_2_MAJF
Direction Direction_Servo_2
0
Speed Speed_Servo_2
0.0
Speed Units Units per sec
Accel Rate Accel_Decel_Rate_S2
0.0
Accel Units % of Maximum
Decel Rate Accel_Decel_Rate_S2
0.0
Decel Units % of Maximum
Profile Trapezoidal
Accel Jerk 100
Decel Jerk 100
Jerk Units % of Time
Merge Disabled
Merge Speed Programmed

Línea 3: Esta línea permite detener el eje del motor mediante la instrucción MAS. Cada una de las instrucciones utilizadas se explican detalladamente en el numeral 3.2.


Stop_servo_2
Start_Servo_2

MAS
Motion Axis Stop
Axis Servo_2
Motion Control Servo_2_MSF
Stop Type Jog
Change Decel No
Decel Rate Accel_Decel_Rate_S2
0.0
Decel Units % of Maximum
Change Decel Jerk Yes
Decel Jerk 100
Jerk Units % of Time

Línea 4: Esta línea en primera instancia obtiene el valor de la velocidad actual del servo mediante la instrucción GSV, y en segunda instancia ajusta este valor a las unidades de RPM.


GSV
Get System Value
Class Name Axis
Instance Name Servo_2
Attribute Name ActualVelocity
Dest Temp_2
0.0

CPT
Compute
Dest Actual_Velocity_Servo_2
0.0
Expression (Temp_2*6800)/113.33334

Línea 5: Esta línea marca el final de la subrutina (END).

3.4. Descripción de la Interface Gráfica Desarrollada para la Aplicación

La operación del sistema de control de movimiento se ejecuta a través de la interfaz gráfica implementada con el software FactoryTalk View.

El software FactoryTalk View brinda un conjunto escalable y unificado de soluciones de monitorización y control diseñadas para ampliar las aplicaciones a nivel de máquina autónoma a aplicaciones HMI de supervisión por una red. Incluye FactoryTalk View SE y FactoryTalk View ME, los cuales son software diseñados con un aspecto y navegación comunes para ayudar a acelerar el desarrollo de la aplicación de HMI. [8]

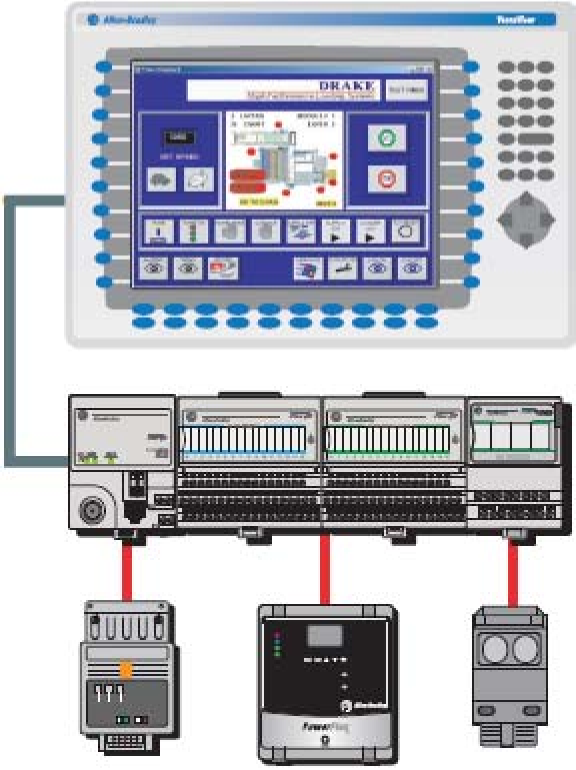


Figura 77. Aplicación con el FactoryTalk View Studio. [8]

- **FactoryTalk View Machine Edition (ME).** Es una unidad de software compacto, que está orientado al control de una máquina o un proceso pequeño. Compatible con soluciones de interface de operador abiertas e incorporadas para monitoreo y control de procesos pequeños y de máquinas individuales. A su vez, puede ser instalado en un hardware dedicado denominado PanelView Plus o Plus CE, o en un PC. [8] (Ver Figura 78)

Figura 78. Aplicación con FactoryTalk View ME, SE (Local) y PanelView Plus. [32]

- **FactoryTalk View SE.** Es una arquitectura escalable que puede ser instalada como una aplicación en un solo PC (Usuario, Servidor o Stand-alone), o como aplicación distribuida, en varios servidores con respaldo o varios clientes. (Ver Figuras 72 y 73)

  A su vez FactoryTalk View:

  - Ofrece monitorización a nivel supervisor y control de aplicaciones que requieren una estructura escalable y distribuida.
  - Es compatible con aplicaciones de servidor distribuido/multiusuario con lo que se consigue el máximo control de la información en cualquier ubicación.
  - La arquitectura escalable se puede aplicar a aplicaciones autónomas/de un servidor, así como a aplicaciones a gran escala con múltiples usuarios o múltiples servidores. [8]

Para la aplicación realizada se utilizo el FactoryTalk View SE, ya que las dimensiones de la PanelView Plus 600 utilizada en el Laboratorio de Automatización no permiten una visión clara del sistema de control de movimiento, por lo que a través del FT View SE se realizo la configuración de las dimensiones de la interfaz. A su vez esta aplicación se ejecuta desde el computador, que esta integrado a la red de la arquitectura Rockwell Automation. (Ver Figura 79)

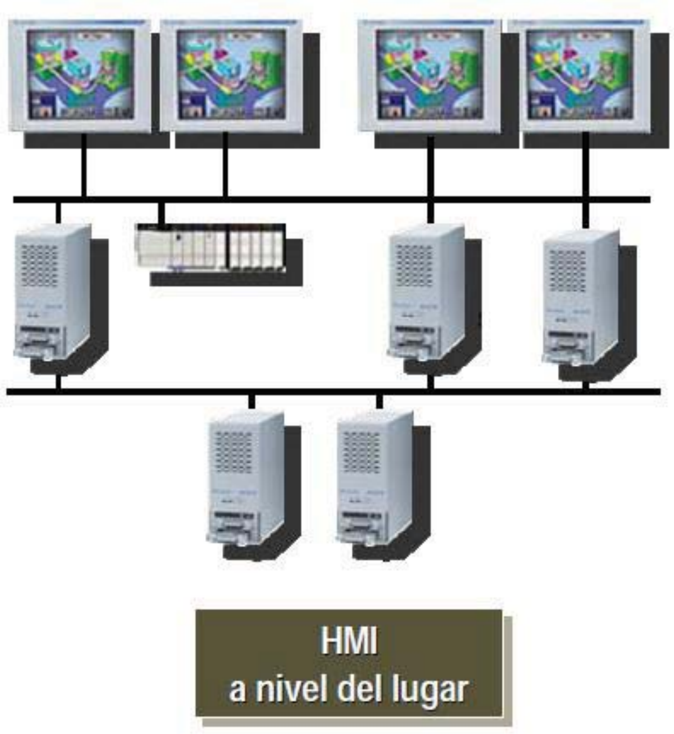
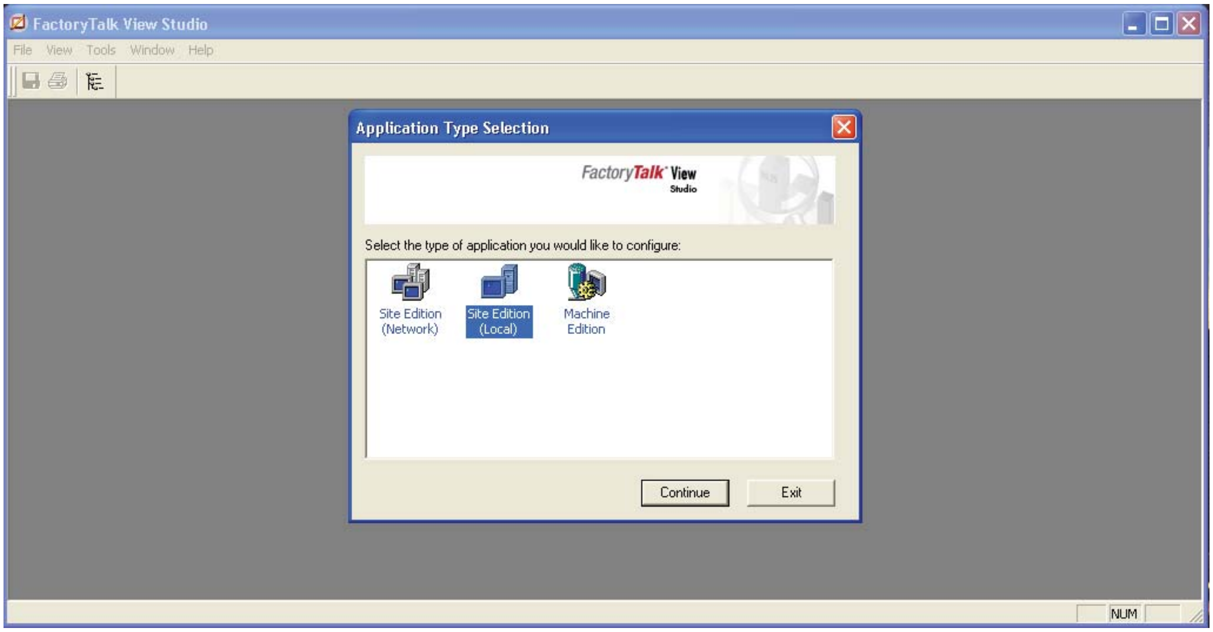


Figura 79. Aplicación con el FactoryTalk View SE (Network). [32]

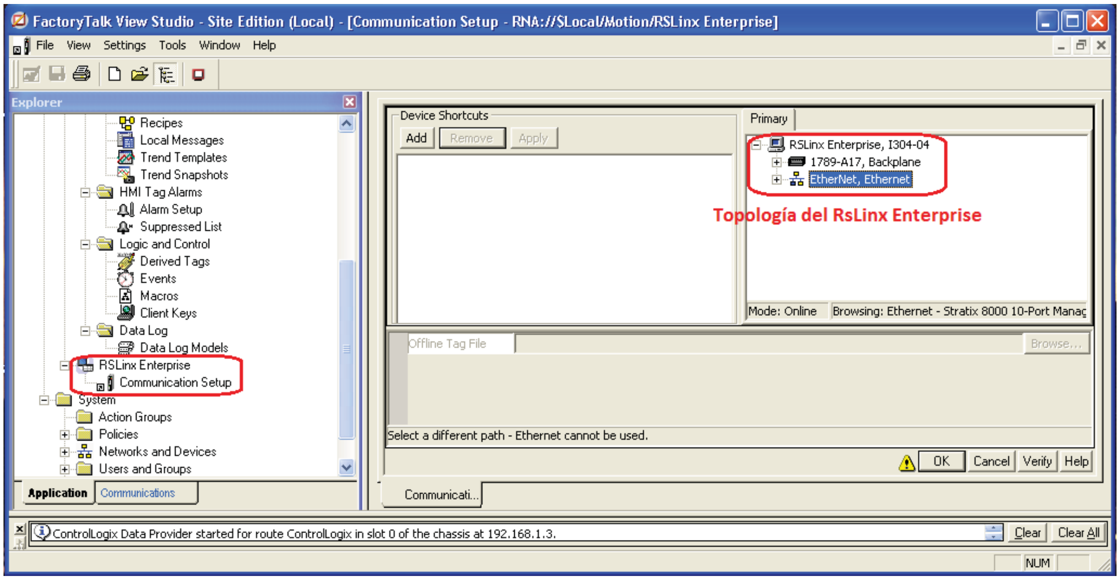


Select the type of application you would like to configure:

Figura 80. Ejecución del FactoryTalk View SE (Local). [28]

En el FactoryTalk View SE (Local) inicialmente se realizó la configuración de la comunicación a través del RsLinx Enterprise accediendo a *Communication Setup*.

El RsLinx Enterprise se utiliza cuando la comunicación es desde un Panel View Plus o del FT View Studio a un controlador Allen Bradley. (Ver Figura 81)

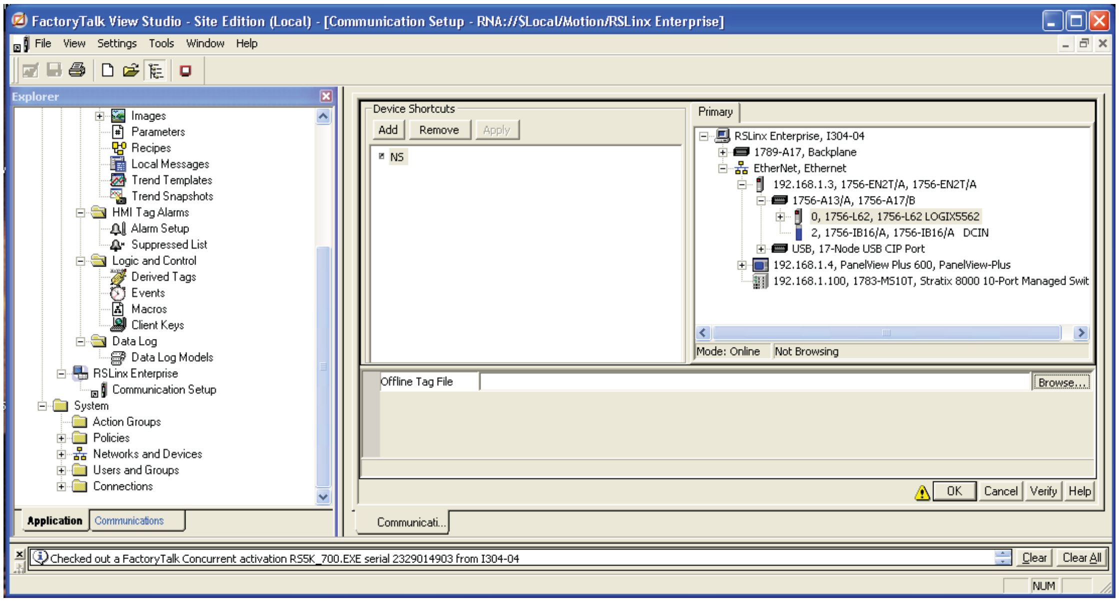


Select a different path - Ethernet cannot be used.

ControlLogix Data Provider started for route ControlLogix in slot 0 of the chassis at 192.168.1.3.

Figura 81. Ventana para Configuración de la Comunicación. [28]

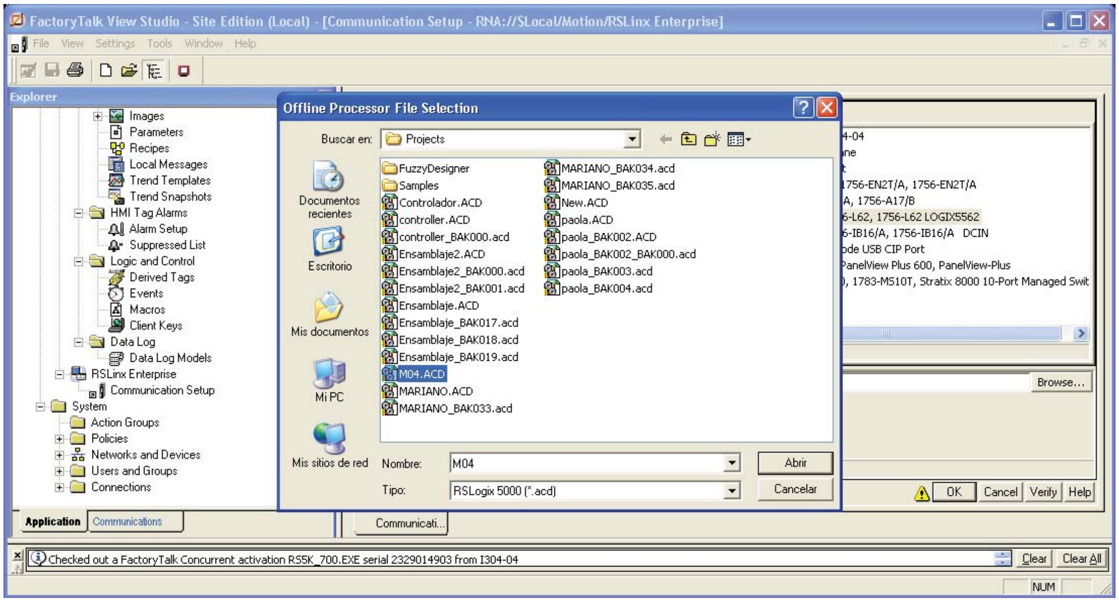
En la configuración de la comunicación se seleccionó el controlador 1756-L62 y se creó el acceso al dispositivo, de modo que el FT View Studio tenga la ruta para poder realizar la comunicación con el controlador. (Ver Figura 82)



Checked out a FactoryTalk Concurrent activation RSSK_700.EXE serial 2329014903 from I304-04

Figura 82. Configuración de Communication Settup en FTView Studio. [28]

A su vez en la configuración de la comunicación se especificó el programa realizado para el sistema de control de movimiento en el RsLogix 5000, de modo que se cargan los Tag's Ofline asociados a este programa, para relacionarlos con la interfaz desarrollada. (Ver Figura 83)



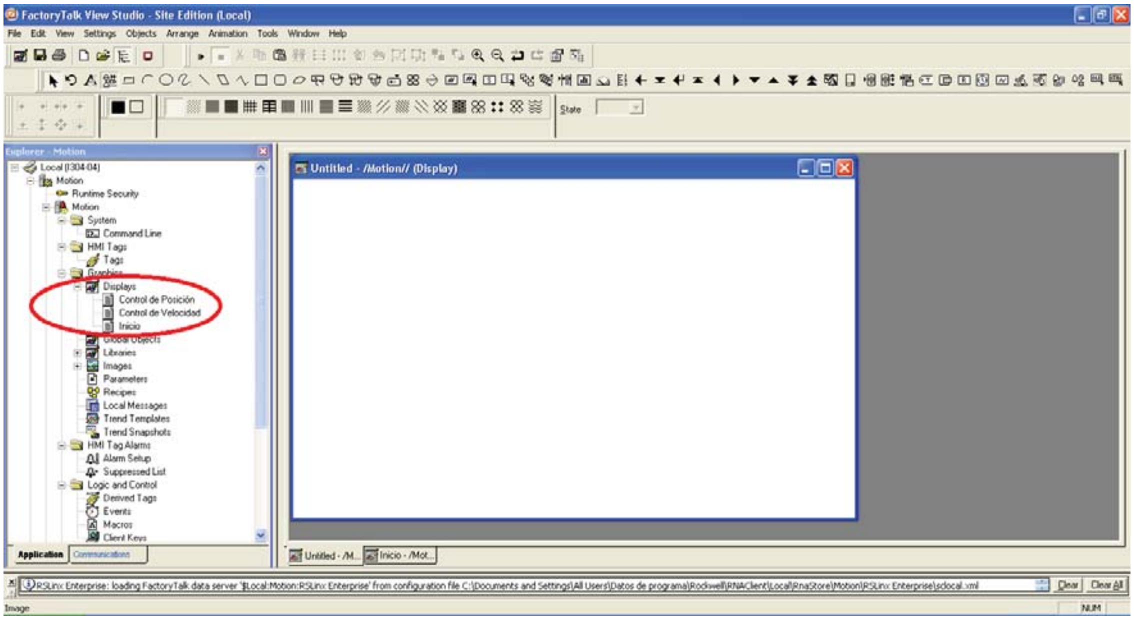
Nombre: M04

Tipo: RSLogix 5000 (*.acd)

Checked out a FactoryTalk Concurrent activation RS5K_700.EXE serial 2329014903 from I304-04

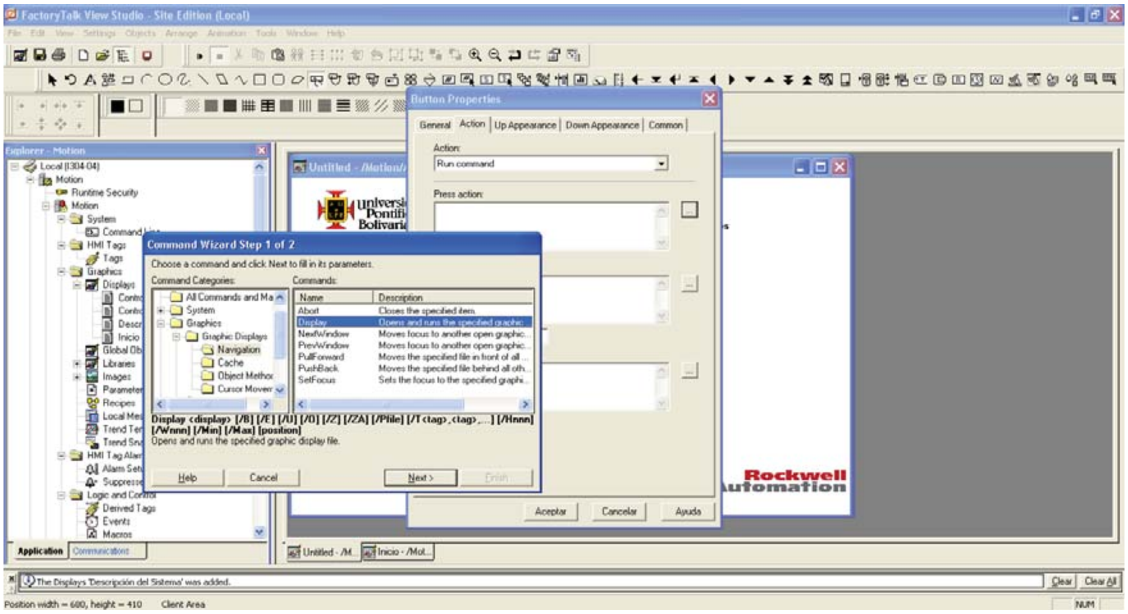
Figura 83. Configuración de Communication Settup en FTView Studio. [28]

Posteriormente se realizó la configuración de las tres pantallas ó *Displays* asociadas al sistema de control de movimiento. (Ver Figuras 84, 85 y 86)



RSLinx Enterprise: loading FactoryTalk data server '$Local:Motion:RSLinx Enterprise' from configuration file C:\Documents and Settings\All Users\Datos de programa\Rockwell\RNAClient\Local\RnaStore\Motion\RSLinx Enterprise\sclocal.xml

Figura 84. Configuración de Display's en FT View Studio. [28]



Action: Run command

Press action:

Choose a command and click Next to fill in its parameters.

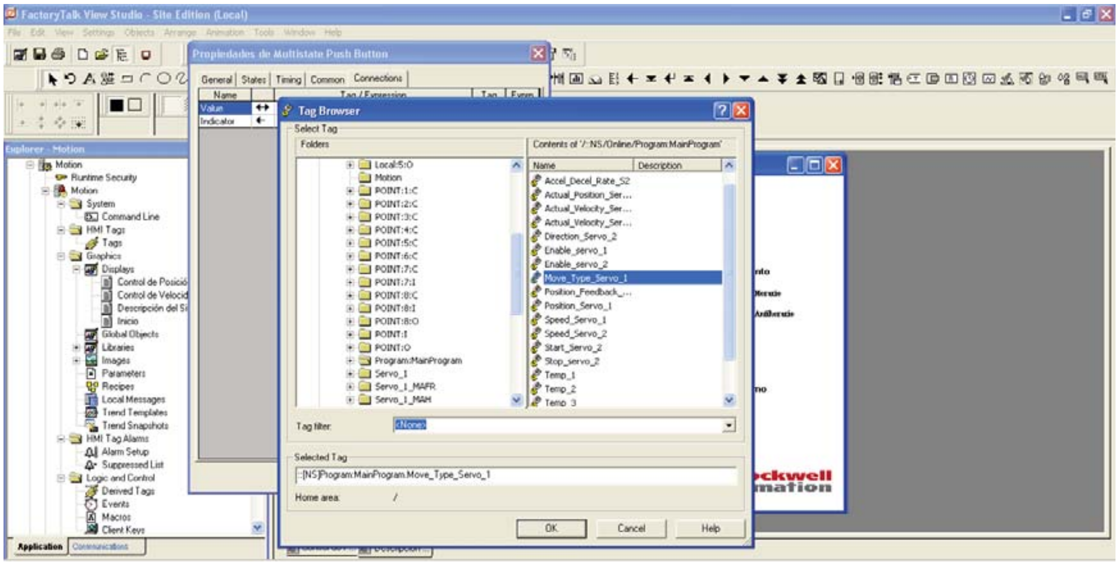
Name	Description
Abort	Closes the specified item.
Display	Opens and runs the specified graphic
NextWindow	Moves focus to another open graphic...
PrevWindow	Moves focus to another open graphic...
PullForward	Moves the specified file in front of all ...
PushBack	Moves the specified file behind all oth...
SetFocus	Sets the focus to the specified graphi...

Display <display> [/B] [/E] [/U] [/O] [/Z] [/ZA] [/Plile] [/T<tag>,<tag>,...] [/Hnnn] [/Wnnn] [/Min] [/Max] [position]
Opens and runs the specified graphic display file.

The Displays 'Descripción del Sistema' was added.

Position width = 600, height = 410 Client Area

Figura 85. Configuración de los Elementos de la Interfaz en FT View. [28]



Contents of '/::NS/Online/Program:MainProgram'

Name	Description
Accel_Decel_Rate_S2	
Actual_Position_Ser...	
Actual_Velocity_Ser...	
Actual_Velocity_Ser...	
Direction_Servo_2	
Enable_servo_1	
Enable_servo_2	
Move_Type_Servo_1	
Position_Feedback_...	
Position_Servo_1	
Speed_Servo_1	
Speed_Servo_2	
Start_Servo_2	
Stop_servo_2	
Temp_1	
Temp_2	
Temp_3	

Tag filter: <None>

Selected Tag: ::[NS]Program:MainProgram.Move_Type_Servo_1

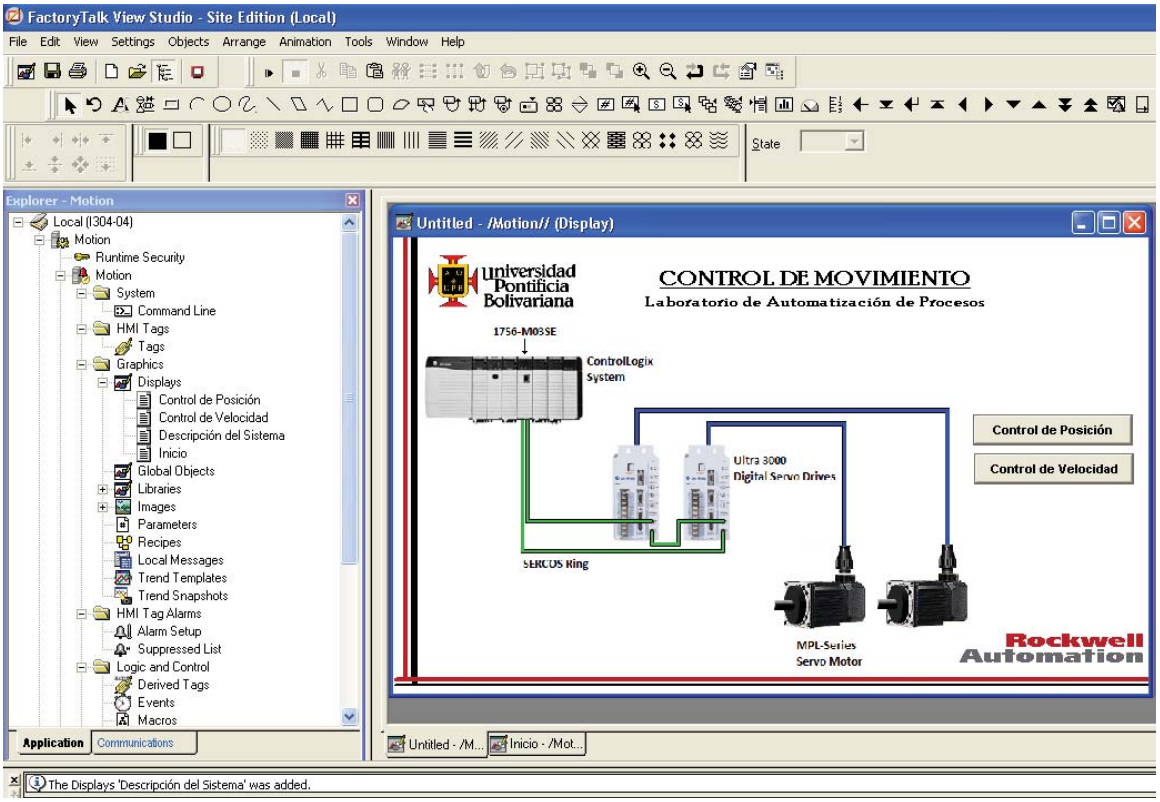
Home area: /

Figura 86. Configuración de Tag's con las Pantallas de la Interfaz. [28]

En la interfaz del sistema de control de movimiento se desarrollaron cuatro pantallas denominadas: Inicio, Control de Posición y Control de Velocidad que se explicaran a continuación:

- **Inicio.** Es la pantalla principal que le indica al operador la aplicación realizada para el Sistema de Control de Movimiento, ya que muestra la conexión realizada y el hardware utilizado: Rack, Módulo Sercos 1756-M03SE, Ultra

3000 y los Servomotores. A su vez, desde esta pantalla se da acceso, al control que se desea realizar sobre el sistema, a través de dos botones denominados: Control de Posición y Control de Movimiento, los cuales a su vez dan acceso a la pantalla correspondiente. (Ver Figura 87)



The Displays 'Descripción del Sistema' was added.

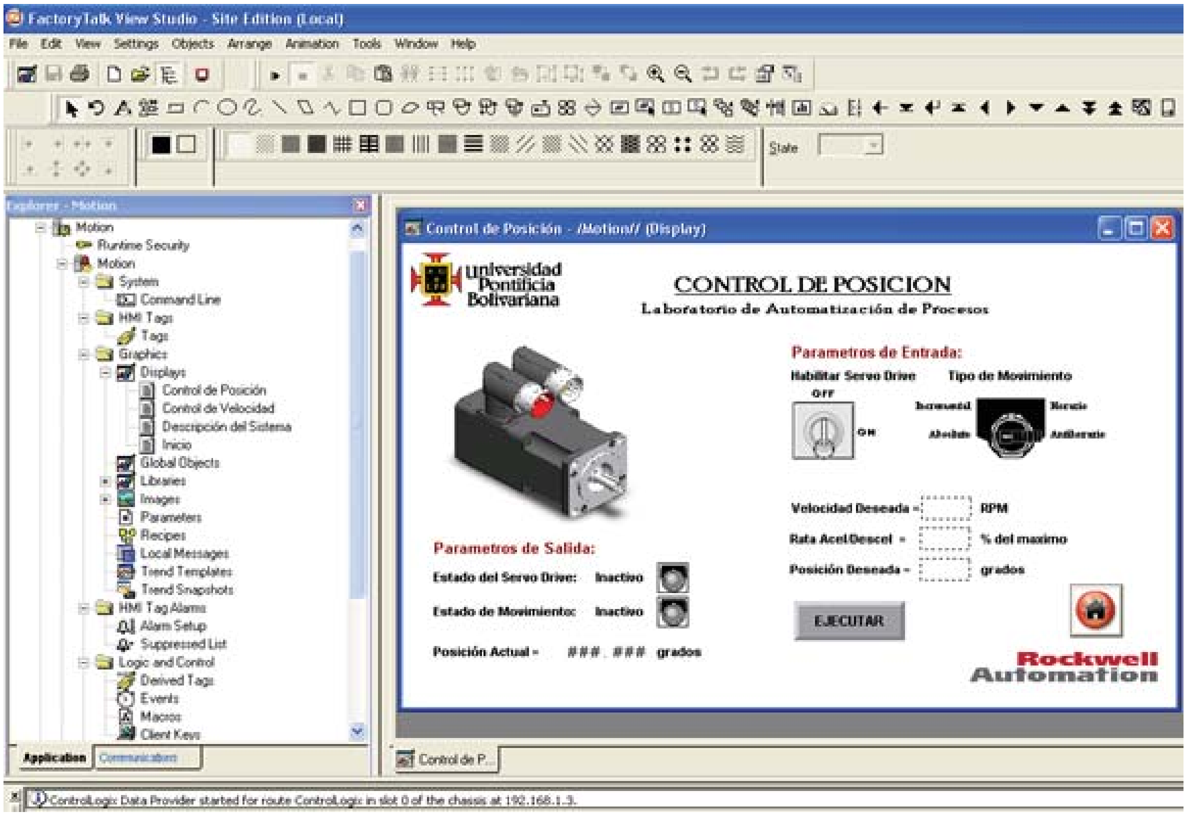
Figura 87. Pantalla Inicio del Sistema de Control de Movimiento. [28]

- **Control de Posición.** En esta pantalla se ejecuta el control de posición del Servo drive configurado, a través de la configuración realizada en los *Parametros de Entrada* detallados a continuación:

  - **Habilitar Servo Drive.** Llave selectora, a través de la cual se activa o desactiva el Servodrive.
  - **Tipo de Movimiento.** Llave selectora con la que se elige el moviento que se desea para el Servo Drive: Absoluto, Incremental, Horario o Antihorario.
  - **Velocidad Deseada (RPM), Rata Acel/Descel y Posición Deseada**. Campos de entrada de número, en la que se ingresa la velocidad, aceleración y posición en grados del ServoDrive. Posterior, a la especificación de estos parametros, se da clic sobre el botón Ejecutar, para realizar el control deseado sobre el Servo Drive. (Ver Figura 82)

A su vez, de acuerdo a la operación del Servo Drive se observa en esta pantalla los Parametros de Salida, que corresponden:

- **Estado del Servo Drive.** Indica el estado del Servo Drive.
- **Estado del Movimiento.** Indica si se esta ejecutando un movimiento.
- **Posición Actual (Grados).** Indica el valor de la posición obtenida del encoder del Servo Drive. (Ver Figura 88)



Estado del Servo Drive: Inactivo

Estado de Movimiento: Inactivo

Posición Actual = ###.### grados

ControlLogix Data Provider started for route ControlLogix in slot 0 of the chassis at 192.168.1.3.

Figura 88. Pantalla para Control de Posición. [28]

- **Control de Velocidad.** En esta pantalla se ejecuta el control de velocidad del Servo drive configurado, a través de la configuración realizada en los *Parametros de Entrada* detallados a continuación:

  - **Habilitar Servo Drive.** Llave selectora, a través de la cual se activa o desactiva el Servodrive.
  - **Sentido de Giro.** Llave selectora con la que se elige el sentido de giro deseado para el Servo Drive: Horario o Antihorario.

- **Velocidad Deseada (RPM) y Rata Acel/Descel**. Campos de entrada de número, en la que se ingresa la velocidad y la aceleración del ServoDrive para el control. Posterior, a la especificación de estos parametros, se da clic sobre el botón Start para iniciar el control de velocidad con estos parámetros. A su vez si se desea detener el Servo Drive y configurar de nuevo estos parámetros se pulsa el botón Stop. (Ver Figura 89)

Por otra parte, de acuerdo a la operación del Servo Drive se observa en esta pantalla los Parametros de Salida, que corresponden:

- **Estado del Servo Drive.** Indica el estado del Servo Drive.
- **Estado del Movimiento.** Indica si se esta ejecutando un movimiento.
- **Velocidad Actual (RPM).** Indica el valor de la velocidad obtenida del encoder del Servo Drive. (Ver Figura 89)

Estado del Servo Drive: Inactivo

Estado de Movimiento: Inactivo

Velocidad Actual = ###.### RPM

Figura 89. Pantalla para Control de Velocidad. [28]

A su vez la ejecución de la aplicación se realiza con Factory Talk View Site Edition Client tal como se muestra en las figura 90, 91 y 92.

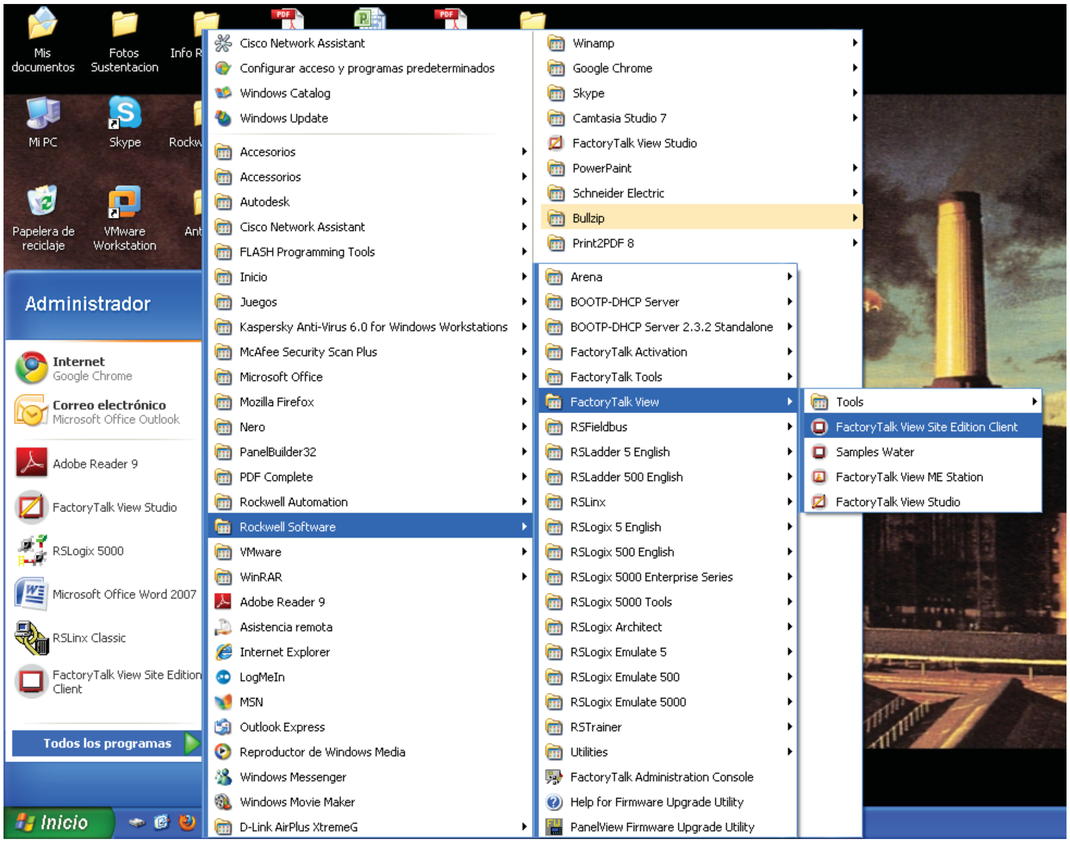


Figura 90. Ejecución de la Aplicación. [28]

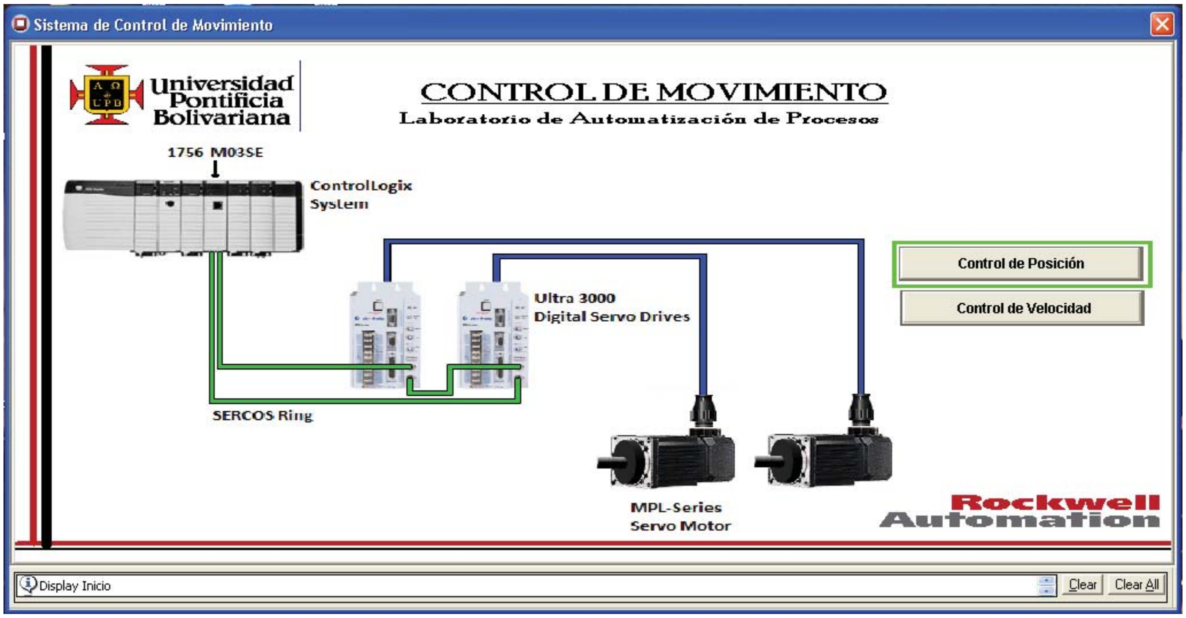


Figura 91. Ejecución de Interfaz para el Sistema de Control de Movimiento. [28]

Parametros de Entrada:

Habilitar Servo Drive

Tipo de Movimiento

Velocidad Deseada = 5000 RPM

Rata Acel/Descel = 100 % del maximo

Posición Deseada = 100 grados

Parametros de Salida:

Estado del Servo Drive: Activo

Estado de Movimiento: Inactivo

Posición Actual = 0.101 grados

Figura 92. Ejecución del Control de Posición. [28]

CONCLUSIONES

La Arquitectura Integrada de Rockwell Automation es una infraestructura de automatización industrial que proporciona soluciones escalables para todo el rango de las disciplinas de automatización, ya que combina la plataforma de Control Logix, la arquitectura de red abierta NetLinx, la plataforma de visualización View y los servicios de información y datos FactoryTalk bajo una misma estructura, permitiendo reducir el tiempo y costo de desarrollo, reducir los costos de mantenimiento y el tiempo improductivo, y adicionalmente tener fácil acceso a los datos de la planta y de producción desde sistemas empresariales para una mejor toma de decisiones administrativas.

El sistema de control de movimiento integrado de Rockwell Automation ofrece las funcionalidades típicas de un sistema de control, integrando aplicaciones discretas, de movimiento, de potencia y mecánicas en una misma plataforma, para una solución simplificada y de alto rendimiento que controla la posición y la velocidad de las máquinas en forma precisa. Este tipo es muy utilizado en para aplicaciones de envasado, conversión, manejo de materiales, ensamblado electrónico, automotriz, robótica, equipo médico o de laboratorio, entre otras.

Se implementó una interfaz gráfica (HMI) para la operación del Sistema de Control de Movimiento, utilizando el Software Factory Talk View Studio, con el objetivo de garantizar que la interacción entre el operario y el sistema sea clara y transparente, y a su vez aprovechar las herramientas brindadas por la arquitectura integrada de Rockwell Automation.

La implementación del sistema de control de movimiento brinda el soporte teórico para establecer el alcance de las aplicaciones que se podrían desarrollar en un futuro en proyectos de grado, y a su vez permite realizar prácticas de funcionamiento, programación y configuración del sistema.

BIBLIOGRAFIA

[1] Automatización de Procesos Industriales, Ebook, Msc .Juan Carlos Villamizar. Universidad Pontifica Bolivariana.

[2] NATIONAL INSTRUMENTS. Fundamentos de Control de Movimiento. [Página de internet]. En: http://sine.ni.com/np/app/culdesac/p/ap/motion/lang/es/pg/1/sn/n17:motion/docid/tut-8480 [Consulta: 2010-15-07].

[3] Rodríguez G, Luis. Estación de Trabajo Ultra3000. Ingeniera en Control y Automatización. Instituto Politécnico Nacional. Escuela de Ingenieria Mecánica y Eléctrica. México. Octubre/2009.

[4] ROCKWELL AUTOMATION, Instrucciones de control de movimiento de controladores Logix5000. Publicación 1756-RM007K-ES-P – Julio 2008 [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2010-9-10].

[5] ROCKWELL AUTOMATION. Arquitectura Integrada Escalable, Multidisciplinario y Habilitada por Información. [Página de internet]. http://www.rockwellautomation.com/es/solutions/integratedarchitecture/index.html [Consulta: 2010-15-07]

[6] ROCKWELL AUTOMATION. Arquitectura integrada. Publicación IAEMEA-BR002B-ES-P – Marzo 2006 [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2010-12-05].

[7] ROCKWELL AUTOMATION, Arquitectura Integrada Herramientas de simplificación. Publicación MR1769-BR002A-EN-P. Julio de 2007 [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2011-1-06].

[8] ROCKWELL AUTOMATION. Arquitectura integrada. Versión IA-BR004A-ES-P – Enero 2007 [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2010-11-15].

[9] ROCKWELL AUTOMATION. Movimiento Integrado Kinetix. Publicación MOTION-BR001E-ES-P – Febrero de 2006 [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2010-11-10].

[10]ROCKWELL AUTOMATION. Controladores Logix5000. Publicación 1756-QR107C-ES-P Junio 2005 [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2010-08-18].

[11]ROCKWELL AUTOMATION. Guía de selección de ControlLogix. Publicación 1756-SG001G-ES-P — Marzo de 2004 [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2010-10-20].

[12]ROCKWELL AUTOMATION. 1756 ControlLogix Integrated Motion Modules Specifications. Publication 1756-TD004B-EN-E - Mayo 2010 [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2010-9-06].

[13]ROCKWELL AUTOMATION. Módulo ControlLogix SERCOS Interface Instrucciones de Instalación. Publicación 1756-IN572G-ES-P – Mayo 2005 [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2010-10-08].

[14]ROCKWELL AUTOMATION. Servoariadores de la familia Ultra. Publicación 2098-BROO1B-ES-P — Junio de 2001 [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2010-10-08].

[**15**] ROCKWELL AUTOMATION. Kinetix Integrated Motion Brochure. Publicación MOTION-BR001E-ES-P – Febrero de 2006 [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2010-10-20].

[16]ROCKWELL AUTOMATION. Ultra3000 Installation Manual. Publication 2098-IN003E-EN-P — Abril 2004 [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2010-09-24].

[17]ROCKWELL AUTOMATION. Ultra3000 Servo Drives Integration Manual. Publication 2098-IN005C-EN-P — March 2008 [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2010-09-15].

[18]ROCKWELL AUTOMATION. MP-Series Low Inertia Motors Product Profile Publication MP-PP001N-EN-P – December 2008 [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2010-10-25].

[19]RODRIGUEZ D. JAIME A., SANGREGORIO S. VIYILS. Implementación del sistema de control tecnología *Rockwell Automation* en el laboratorio de automatización industrial de la universidad pontificia bolivariana. Universidad Pontificia Bolivariana, Agosto de 2010.

[20]ROCKWELL AUTOMATION. Universal Power Cables Packing. Data Publication 2090-PC001B-EN-P - Junio 2008 [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2010-8-11].

[21]ROCKWELL AUTOMATION. Universal Feedback Cables Packing. Data Publication 2090-PC002B-EN-P - Junio 2008. [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2010-8-11].

[22]ROCKWELL AUTOMATION. CN2 Motor Feedback Breakout Board. Publication 2090-IN006B-EN-P - Agosto 2010 [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2010-8-16].

[23]ROCKWELL AUTOMATION. Instrucciones de Control de Movimiento de Controladores Logix5000. Publicación 1756-RM007K-ES-P – Julio 2008 [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2010-10-05].

[24]ROCKWELL AUTOMATION. Manual de Referencia del Conjunto de Instrucciones de Control de Movimiento. Publicación 1756-6.4.3ES - Octubre 1999. [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2011-01-06].

[25]ROCKWELL AUTOMATION. Kinetix Motion Control Selection Guide. Publication GMC-SG001P-EN-P - January 2011 [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2011-02-20].

[26]ROCKWELL AUTOMATION. RSLogix 5000 [Página de Internet]. En: http://www.rockwellautomation.com/rockwellsoftware/design/rslogix5000/. [Consulta: 2010-12-27].

[27]ROCKWELL AUTOMATION. Logix Controllers General Instructions Reference Manual. Publication 1756-RM003K-ES-P – Julio 2008 [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2010-10-22].

[28]Arias J, Juan José. Puesta en Operación del Sistema de Control de Movimiento de la Arquitectura de Automatización Rockwell Automation. Especialización en Control e Instrumentación Industrial. Tercera Cohorte. Facultad de Ingeniería Electrónica. UPB - Bucaramanga. Enero / 2011.

[29]ROCKWELL AUTOMATION, Configuración y Puesta en Marcha del Sistema CIP Motion. Publicación MOTION-UM003A-ES-P – Febrero 2010 [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2010-10-22].

[30]ROCKWELL AUTOMATION, ControlLogix System User Manual Publicación 1756-UM001E-ES-P - Agosto 2002 [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2011-01-09].

[31]ROCKWELL AUTOMATION, SERCOS and Analog Motion Configuration and Startup. Publication MOTION-UM001A-EN-P - January 2010 [Documento electrónico]. En: http://literature.rockwellautomation.com [Consulta: 2010-11-25].

[32]ROCKWELL AUTOMATION. Soluciones para Máquinas a Nivel HMI – Escalable y Perfectamente Integradas. Rendimiento y Visibilidad. Año 2008.

ANEXOS

ANEXO A. Especificaciones de los Controladores Logix.

CARACTERISTICAS DE LOS CONTROLADORES LOGIX

	ControlLogix	CompactLogix L4x	CompactLogix L3x	GuardLogix
				
Disciplinas de control				
Discretas	■ Sí	■ Sí	■ Sí	■ Sí
De movimiento	■ 32 ejes totalmente integrados con SERCOS interfax	■ 4 ejes totalmente integrados con SERCOS interface		■ 32 ejes totalmente integrados con SERCOS interface
Variador	■ Sí	■ Sí	■ Sí	■ Sí
Proceso	■ Sí	■ Sí	■ Sí	■ Sí
Lote	■ Sí	■ Sí	■ Sí	■ Sí
Seguridad	■ SIL 2			■ SIL 3
Redundancia (varía según el sistema)	■ Sí	■ Sí	■ Sí	
Lenguajes de programación				
Lógica de escalera de relés	■ Sí	■ Sí	■ Sí	■ Sí
Diagrama de bloque de funciones	■ Sí	■ Sí	■ Sí	■ Sí
Diagrama de función secuencial	■ Sí	■ Sí	■ Sí	■ Sí
Texto estructurado	■ Sí	■ Sí	■ Sí	■ Sí
C/C++				
Capacidades				
Memoria (máx.)	■ 16 Mbyte	■ 2 Mbyte	■ 2 Mbyte	■ 4 Mbyte
Memoria no volátil	■ Extraíble	■ Extraíble	■ Extraíble	■ Extraíble
E/S locales - #nº de módulos (tipo)	■ 16 (ControlLogix I/O)	■ 16 (ControlLogix I/O)	■ 30 (Compact I/O)	■ 16 (ControlLogix I/O)
E/S distribuidas mediante redes CIP	■ Sí	■ Sí	■ Sí	■ Sí
Redes				
En serie (ASCII, DF-1, DH-485, Modbus)	■ 1	■ 1	■ 1	■ 1
EtherNet/IP	■ múltiple	■ hasta 2	■ hasta 1	■ múltiple
ControlNet	■ múltiple		■ hasta 1	■ múltiple
DeviceNet	■ múltiple	■ múltiple	■ múltiple	■ múltiple
Data Highway Plus	■ múltiple			■ múltiple
E/S remotas	■ múltiple			■ múltiple
Otros fabricantes	■ muchos	■ muchos	■ muchos	■ muchos
Software				
Software de configuración	■ RSLogix5000	■ RSLogix5000	■ RSLogix5000	■ RSLogix5000

	FlexLogix	**DriveLogix**	**SoftLogix**
			
Disciplinas de control			
Discretas	■ Sí	■ Sí	■ Sí
De movimiento			■ 16 ejes totalmente integrados con SERCOS interface
Variador	■ Sí	■ Sí	■ Sí
Proceso	■ Sí	■ Sí	■ Sí
Lote	■ Sí	■ Sí	■ Sí
Seguridad			
Redundancia (varía según el sistema)	■ Sí		
Lenguajes de programación			
Lógica de escalera de relés	■ Sí	■ Sí	■ Sí
Diagrama de bloque de funciones	■ Sí	■ Sí	■ Sí
Diagrama de función secuencial	■ Sí	■ Sí	■ Sí
Texto estructurado	■ Sí	■ Sí	■ Sí
C/C++			■ Sì (mediante rutinas externas)
Capacidades			
Memoria (máx.)	■ 512 kbyte	■ 1,5 Mbyte	■ 64 Mbyte
Memoria no volátil	■ Fijo	■ Extraíble	■ Rutina
E/S locales - #nº de módulos (tipo)	■ 16 (FLEX I/O)	■ 16 (Compact I/O)	■ ninguna
E/S distribuidas mediante redes CIP	■ Sí	■ Sí	■ Sí
Redes			
En serie (ASCII, DF-1, DH-485, Modbus)	■ 1	■ 1	■ múltiple
EtherNet/IP	■ hasta 2	■ 1	■ hasta 1
ControlNet	■ hasta 2	■ 1	■ hasta 1
DeviceNet	■ múltiple	■ 1	■ múltiple
Data Highway Plus			
E/S remotas			
Otros fabricantes	■ muchos	■ muchos	
Software			
Software de configuración	■ RSLogix5000	■ RSLogix5000	■ RSLogix5000

ANEXO B. Redes Soportadas por Rockwell Automation.

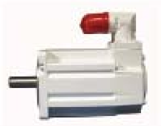
REDES INDUSTRIALES UTILIZADAS POR ROCKWELL AUTOMATION

	EtherNet/IP	ControlNet	DeviceNet
Capacidades de red			
Configuración de dispositivos	■ Sí	■ Sí	■ Sí
Recolección de datos	■ Sí	■ Sí	■ Sí
Enclavamiento entre dispositivos similares	■ Sí	■ Sí	■ Sí
Control en tiempo real	■ Sí	■ Sí	■ Sí
Conexión en red de seguridad	■ Sí	■ Sí	■ Sí
Control de movimiento	■ Sí		
Integración con email, web, voz, vídeo	■ Sí		
Conectividad de dispositivos			
Ordenadores	■ Sí	■ Sí	■ Sí
HMI	■ Sí	■ Sí	■ Sí
Controladores, robots	■ Sí	■ Sí	■ Sí
E/S	■ Sí	■ Sí	■ Sí
Variadores	■ Sí	■ Sí	■ Sí
Arrancadores de motor, sobrecargas			■ Sí
Válvulas	■ Sí	■ Sí	■ Sí
Código de barras, lectores RFID	■ Sí	■ Sí	■ Sí
Sensores foto-eléctricos, proxis			■ Sí
Instrumentación de proceso	■ Sí		
Topología/ Rendimiento/ Capacidad			
Topologías	■ Árbol, estrella, anillo, línea, alta disponibles	■ Línea principal con línea de derivación 1 m, anillo	■ Línea principal con líneas deriv. nodo múltiple 20’
Medios físicos	■ Cualquier medio físico Ethernet estándar – cobre, fibra	■ Fibra, coaxial RG6U (CATV)	■ Medios físicos gruesos, delgados, planos, 4 ó 5 conductores
Opciones de medios físicos IP-67	■ Sí	■ Sí	■ Sí
Capacidad (nodos/red)	■ Miles	■ 99	■ 63
Rendimiento configurable	■ Sí	■ Sí	■ Sí

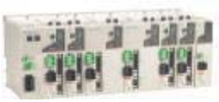
	Hart	Foundation fieldbus	Profibus PA
Capacidades de red			
Configuración de dispositivos	■ Sí	■ Sí	■ Sí
Recolección de datos	■ Sí	■ Sí	■ Sí
Enclavamiento entre dispositivos similares			
Control en tiempo real	■ Sí	■ Sí	■ Sí
Módulo de comunicación	■ Analoga y Digital	■ Digital	■ Digital
Tipo de comunicación	■ Maestro/Esclavo	■ Token Passing	■ Maestro/Esclavo
Conectividad de dispositivos			
E/S	■ Sí		
Válvulas	■ Sí	■ Sí	■ Sí
Instrumentación de proceso	■ Sí	■ Sí	■ Sí
Topología/ Rendimiento/ Capacidad			
Topologías	■ Punto a punto o multipunto	■ Cualquier Bus	■ Cualquier Bus
Medios físicos	■ Tradicional 2, 3 o 4 cables	■ 2 cables	■ 2 cables
Velocidad de comunicación digital	■ 1.2 Kbps	■ 31.25 Kbps	■ 31.25 Kbps
Transmisión o cableado	■ Alimentacion y Comunicación	■ Alimentacion y Comunicación	■ Alimentacion y Comunicación
Disposiciones sobre lugares peligrosos	■ Intrínsecamente seguro	■ FINCO/FNICO	■ FINCO/FNICO
Servidor OPC	■ Sí	■ Sí	

ANEXO C. Especificaciones Equipos Kinetix.

CARACTERISTICAS DE ACCIONADORES, SERVOMOTORES Y SERVOVARIADORES KINETIX

	Motores de baja inercia de la serie MP	Motores aptos para la industria de alimentos de la serie MP	Motores de acero inoxidable de la serie MP	Motores de la serie TL
ACCIONADORES Y SERVOMOTORES KINETIX				
Voltaje y velocidad del motor	■ Bobinados de 230 y 460 voltios ■ Hasta 8000 rpm	■ Bobinados de 230 y 460 voltios ■ Hasta 5000 rpm	■ Bobinados de 230 y 460 voltios ■ Hasta 5000 rpm	■ Bobinado de 230 voltios ■ Hasta 5000 rpm
Opciones de realimentación	■ Realimentación de alta resolución absoluta con opción de múltiples vueltas ■ Encoder incremental de 2000 líneas (dimensiones de estructura ■ Dispositivo de resolución	■ Realimentación de alta resolución absoluta con opción de múltiples vueltas	■ Realimentación de alta resolución absoluta con opción de múltiples vueltas	■ Realimentación en serie de alta resolución con capacidad de múltiples vueltas (requiere batería de respaldo) ■ Encoder incremental de 2000 líneas
Par de parada continuo	■ 0,26 –163 Nm (2,3 – 1440 libras-pulgada)	■ 1,6 a 19,4 Nm (14 – 172)	■ 3,6 – 21,5 Nm (32 – 190 libras-pulgada)	■ 0,086 – 5,42 Nm (0,76 – 48 libras-pulgada)
Principales características y aplicaciones	■ Baja inercia Características y opciones de primera; motor para uso general de rendimiento dinámico	■ Motores de baja inercia para proyecciones de agua de servicio ligero Para carne, aves y productos lácteos, se recomiendan motores de acero inoxidable de la serie MP.	■ Fn Para condiciones de alta presión, muy cáusticas. Sus aplicaciones incluyen manipulación de carne y aves, alimentos sin procesar, ciencias biológicas y productos de consumo.	■ Familia de productos de baja inercia compactos optimizada para aplicaciones en las que el coste es fundamental ■ Opciones de montaje JIS o NEMA

	Motores de la serie HPK	Servomotores 1326AB	Motores de la serie F	Accionadores lineales integrados de la serie MP	Accionadores rotatorios y motores con engranaje integrado de la serie MP
					
Voltaje y velocidad del motor	■ Bobinado de 460 voltios ■ Opciones de velocidad base de 1500 y 3000 rpm	■ Bobinado de 460 voltios ■ Hasta 7250 rpm	■ Bobinado de 230 voltios ■ Hasta 4000 rpm	■ Bobinados de 230 y 460 voltios ■ Hasta 583 mm/seg Velocidad lineal (23 pulgadas/seg)	■ Bobinados de 230 y 460 voltios ■ Hasta 194 rpm
Opciones de realimentación	■ Realimentación de alta resolución absoluta con opción de múltiples vueltas	■ Realimentación de alta resolución absoluta con opción de múltiples vueltas ■ Dispositivo de resolución	■ Encoder incremental de 2000 líneas	■ Realimentación de alta resolución absoluta con opción de múltiples vueltas	■ Realimentación de alta resolución absoluta con opción de múltiples vueltas
Par de parada continuo	■ 96 a 482 Nm (849–4266 libras-pulgada)	■ 2,7–53 Nm (24-469 libras-pulgada)	■ 3,5-24,2 Nm (31- 214 libras-pulgada)	■ Fuerza continua de 1906–8863 N (428-1992 libras)	■ Par de parada continuo de 15-700 Nm (133 – 6195 libras-pulgada)
Principales características y aplicaciones	■ La precisión de un servomotor combinada con la gran potencia y efectividad de coste de un motor de inducción	■ Inercia media para rendimiento uniforme según la inercia	■ Inercia media para rendimiento uniforme según la inercia	■ Accionador integrado que elimina acopladores, adaptadores, correas, embragues y montaje. Accionador lineal basado en tornillo esférico de precisión de larga durabilidad para mejorar la eficiencia, el posicionamiento preciso y de tamaño compacto	■ Salida de par alto en un diseño compacto. Contragolpe rotativo < 3 arcos minuto

	Kinetix 2000	Kinetix 6000	Kinetix 7000	Ultra3000 con SERCOS interface
SERVOVARIADORES KINETIX				
Plataformas de control Logix	■ CompactLogix, ControlLogix, GuardLogix y SoftLogix con software RSLogix 5000	■ CompactLogix, ControlLogix, GuardLogix y SoftLogix con software RSLogix 5000	■ CompactLogix, ControlLogix, GuardLogix y SoftLogix con software RSLogix 5000	■ CompactLogix, ControlLogix, GuardLogix y SoftLogix con software RSLogix 5000
Corriente de salida continua (A rms)	■ 1 - 9,5	■ 3,7–34	■ 40–180	■ 2,5–65 A (230 V)7–47 A (460 V)
Corriente de salida continua (A rms)	■ 0,3 kW–3 kW	■ 1,2–22 kW	■ 22–112 kW	■ 0,25.1,5 kW (115 V), 0,5.3 kW (230 V) monofasico ■ 1,5.11 kW (230 V), 3.22 kW (460 V) 3-trifasico
Voltios de entrada	■ 230 VCC unipolar y trifásico	■ 195 – 265 VCA trifásico ■ 324 – 528 VCA trifásico	■ 380 . 480 VCA trifasico ■ Configuraciones de bus comun de 450-750 VCC con potencia regenerativa	■ 115–240 VCA monofásico ■ 230–480 V trifásico
Realimentación	■ Encoders (múltiples vueltas) absolutos inteligentes de alta resolución ■ Encoders (una vuelta) de alta resolución inteligentes ■ Encoder incremental ■ Encoders de seno y coseno ■ Capacidades de realimentación absolutas en serie de 17 bits	■ Encoders (multiples vueltas) absolutos inteligentes de alta resolucion ■ Encoders (una vuelta) inteligentes de alta resolucion ■ Encoder incremental ■ Encoders de seno y coseno ■ Dispositivo de resolucion	■ Encoders (multiples vueltas) absolutos inteligentes de alta resolucion ■ Encoders (una vuelta) inteligentes de alta resolucion ■ Encoder incremental ■ Encoders de seno y coseno	■ Encoders (multiples vueltas) absolutos inteligentes de alta resolucion ■ Encoder (una vuelta) inteligentes de alta resolucion ■ Encoder incremental ■ Encoders de seno y coseno
E/S	■ Habilitar (1), Carrera de seguimiento (2), Inicio (1), Entradas de registro (2)	■ Habilitar (1), Carrera de seguimiento (2), Inicio (1), Entradas de registro (2)	■ Habilitar (1), Carrera de seguimiento (2), Inicio (1), Entradas de registro (2)	■ Habilitar (1), Carrera de seguimiento (2), Inicio (1), Entradas de registro (2)
Almacenamiento de firmware en memoria flash programable de campo para facilitar las actualizaciones	■ Sí	■ Sí	■ Sí	■ Sí
Capacidades de diagnóstico de variador remoto avanzadas	■ Sí	■ Sí	■ Sí	■ Sí
Servicio técnico de función de seguridad (incorporado)		■ EN-954-1 categoría 3, IEC61508 SIL3 desactivación segura y prevención de reinicio accidental	■ EN-954-1 categoría 3, IEC61508 SIL3 desactivación segura y prevención de reinicio accidental	
Compatibilidad de motor	■ Baja inercia serie MP, aptos para la industria de alimentos serie MP, acero inoxidable serie MP, con engranaje integrado serie MP, serie TL	■ Baja inercia serie MP, aptos para la industria de alimentos serie MP, acero inoxidable serie MP, con engranaje integrado serie MP, 1326AB, serie F	■ Motores asincronos y de iman permanente sincronos incluidos los de la serie HPK, de baja inercia de la serie MP y servomotores 1326AB	■ Baja inercia serie MP, aptos para la industria de alimentos serie MP, acero inoxidable serie MP, con engranaje integrado serie MP, 1326AB, serie F
Capacidad de reconocimiento de motor automática con dispositivos de realimentación inteligentes	■ Sí	■ Sí	■ Sí	■ Sí
Compatible con accionadores lineales integrados de la serie MP	■ Sí	■ Sí	■ Sí	■ Sí

ANEXO D. Especificaciones de los Servo Drives Digitales Ultra 3000.

ESPECIFICACIONES TECNICAS DEL SERVO-VARIADOR DIGITAL ULTRA 3000 (2098-DSD-005-SE)

CARACTERISTICAS ELECTRICAS	
Corriente de salida pico (Amps)	7.5
Corriente de salida continua (Amps)	2.5
Potencia de salida continua (kW)	0.5
Potencia de derivación continua externa (kW)	300 Watts
Potencia de derivación pico externa	4 kW

ENTRADA	Columna1
Corriente de entrada continua (Amps RMS)	5
Voltaje de entrada	100 – 240 Volts CA monofásico (trifásico para –075 y –150) 12 – 24 VCC requeridos para E/S digitales Alimentación de lógica externa de 5 VCC opcional
Frecuencia de entrada	47 – 63 Hz

MODOS DE OPERACIÓN Y FUENTES DE COMANDOS	
Modo analógico de velocidad/corriente	Entrada de +/–10 Volts
Velocidad, corriente y relaciones de seguidor preseleccionados	8 valores preseleccionados, selección binaria mediante entradas digitales o comandos en serie, sincronismo digital electrónico
Paso y dirección, paso hacia arriba/paso hacia abajo	Frecuencia máxima de 2.5 MHz, entrada diferencial o unipolar
Seguimiento de encoder maestro	Frecuencia de línea máxima de 2.5 MHz, entrada diferencial o unipolar
Comandos digitales en serie	Mediante puerto en serie y protocolo ASCII de 7 bits

ENTRADAS/SALIDAS	
Entradas digitales para uso general	8 ópticamente aisladas de 12 – 24 Volts, entradas altas activas - asignables a una o más selecciones
Selección de entradas para uso general	Habilitación de variador, inhabilitación de comunicaciones en serie, índice de pausa, índice de paro, posición inicial de pausa, posición inicial de paro, selección predefinida, posición preseleccionada establecida, inhibición de integrador, habilitación de seguidor, habilitación de avance, habilitación de retroceso, anulación de modo de operación, estroboscopio de posición, detector de posición inicial, índice de inicio, definición de posición inicial, detector de registro, eliminación de offsets de comando, vuelta a la posición inicial, restablecimiento de fallo
Salidas digitales para uso general	4 salidas ópticamente aisladas de 12 – 24 volts, 50 Miliamperes máximo
Selecciones de salida para uso general	En posición, dentro de ventana de posición, velocidad cero, dentro de ventana de velocidad, variador a velocidad nominal habilitado, bus de CC cargado, preparado, en movimiento, en espera, seguimiento, fin de secuencia, límite de corriente, registrado, en posición inicial, eje en posición inicial, conmutación de inicio efectuada, inhabilitación por fallo de freno, desaceleración/inhabilitación por fallo, ignorar fallo, indicar fallo, sobrecarrera excedida
Salida de relé para uso general	1 relé normalmente abierto, voltaje máximo de 30 volts CC, corriente máxima de 1 Ampere
Respuesta a captura de entrada de registro	<100 µseg
Entrada de comando analógico	1 convertidor analógico a digital de 14 bits (+/–10 v, diferencial)
Salida analógica para uso general	1 convertidor digital a analógico de 8 bits (+/–10 v, +/–2 ma, unipolar)

COMUNICACIONES	
En serie	1 puerto con RS-232/RS-422/RS-485 a 1,200 – 57,600 baudios
Conexión en red	DeviceNet, SERCOS

LAZOS DE CONTROL	
Modos	Control de corriente, velocidad, posición
Tipos	Todos los lazos son digitales
PWM	Modulación de vector de espacio de 8 kHz
Ancho de banda de lazo de velocidad	300 Hz

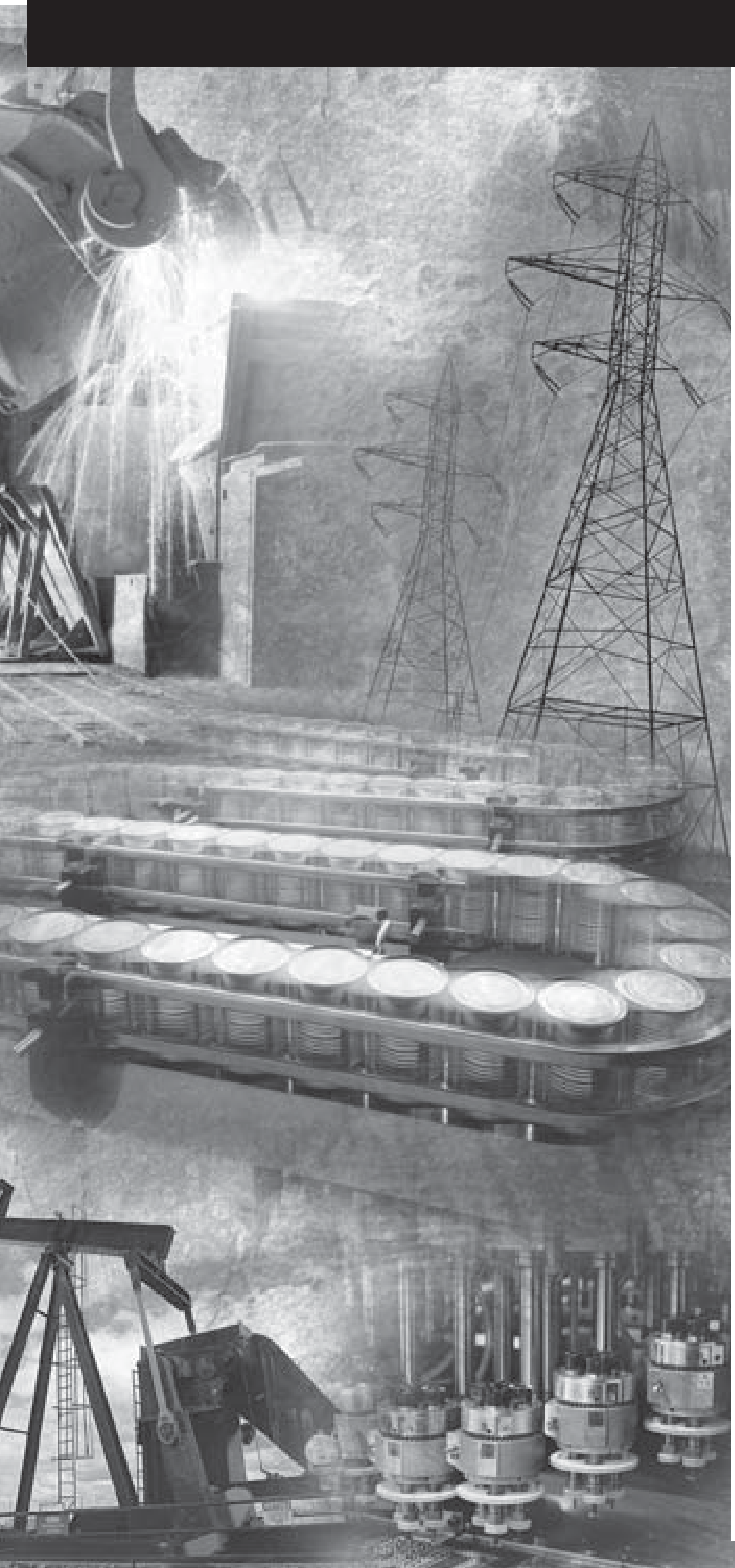
RETROALIMENTACIÓN DE MOTOR	
Modos de entrada	Incremental con índice, seno/coseno alta resolución absoluta (una revolución y múltiples revoluciones)
Frecuencia de entrada máxima	2.5 MHz (líneas de encoder), más de 1 millón de conteos/rev. (alta resolución)
Inicio de conmutación	Detector Hall

RETROALIMENTACIÓN AUXILIAR	
Operación	Entrada de retroalimentación auxiliar de lazo de posición
Modos de entrada	A cuadratura B
Tipo de entrada	Receptor de línea
Frecuencia de entrada máx.	2.5 MHz (líneas de encoder)

CONECTORES	
Conector de control CN1	Conector D hembra de alta densidad de 44 posiciones
Conector de retroalimentación de motor CN2	Conector D hembra de alta densidad de 15 posiciones
Conector de puerto en serie CN3	Conector D hembra de 9 posiciones
Conector principal de motor de CA y bus de CC	Bloque de terminales de tornillo de 9 posiciones

AMBIENTALES	
Temperatura de almacenamiento	–40 °C a 70 °C (–40 °F a 158 °F)
Temperatura de operación	0 °C a 55 °C (32 °F a 131 °F)
Humedad	5 % a 90 % sin condensación
Altitud	1,500 m/5,000 pies (reducción 3 % por 300 m arriba de 1,500 m)
Vibración	10 hasta 2,000 Hz a 2 G
Choque	15 G 11 mseg medio senoidal
Peso	3.7 lbs (1.68 kg)

ANEXO E. Ultra 3000 Digital Servo Drives Installation Manual.



Allen-Bradley

Ultra3000
Digital Servo Drives

(Catalog Numbers
2098-DSD-005, -010, and -020
2098-DSD-*xxx*X
2098-DSD-*xxx*-SE
2098-DSD-*xxx*-DN
2098-DSD-*xxx*X-DN

2098-DSD-030, -075, and -150
2098-DSD-*xxx*X
2098-DSD-*xxx*-SE
2098-DSD-*xxx*-DN
2098-DSD-*xxx*X-DN

2098-DSD-HV030, -HV050, -HV100, -HV150, and -HV220
2098-DSD-HV*xxx*X
2098-DSD-HV*xxx*-SE
2098-DSD-HV*xxx*-DN
2098-DSD-HV*xxx*X-DN)

Installation Manual

Rockwell
Automation

Important User Information

Because of the variety of uses for the products described in this publication, those responsible for the application and use of this control equipment must satisfy themselves that all necessary steps have been taken to assure that each application and use meets all performance and safety requirements, including any applicable laws, regulations, codes and standards.

The illustrations, charts, sample programs and layout examples shown in this guide are intended solely for purposes of example. Since there are many variables and requirements associated with any particular installation, Allen-Bradley® does not assume responsibility or liability (to include intellectual property liability) for actual use based upon the examples shown in this publication.

Allen-Bradley publication SGI-1.1, *Safety Guidelines for the Application, Installation and Maintenance of Solid-State Control* (available from your local Allen-Bradley office), describes some important differences between solid-state equipment and electromechanical devices that should be taken into consideration when applying products such as those described in this publication.

Throughout this manual we use notes to make you aware of safety considerations:

ATTENTION ⚠ Identifies information about practices or circumstances that can lead to personal injury or death, property damage or economic loss.

Attention statements help you to:

- identify a hazard
- avoid a hazard
- recognize the consequences

IMPORTANT Identifies information that is critical for successful application and understanding of the product.

Table of Contents

Preface

Chapter 1

Installing Your Ultra3000

Chapter 2

Ultra3000 Connector Data

Chapter 3

Connecting Your Ultra3000

Chapter 4

Troubleshooting Status Indicators

Appendix A

Specifications and Dimensions

Appendix B

Interconnect Diagrams

Appendix C

Catalog Numbers and Accessories

Preface

Introduction

Read this preface to familiarize yourself with the rest of the manual. This preface contains the following topics:

- Who Should Use this Manual
- Purpose of this Manual
- Contents of this Manual
- Product Receiving and Storage Responsibility
- Related Documentation
- Conventions Used in this Manual
- Allen-Bradley Support

Who Should Use this Manual

Use this manual for designing, installing, and wiring your Ultra™3000 Digital Servo Drive (DSD). The manual is intended for engineers or technicians directly involved in the installation and wiring of the Ultra3000.

If you do not have a basic understanding of the Ultra3000, contact your local Allen-Bradley representative for information on available training courses before using this product.

Purpose of this Manual

This manual provides the mounting, wiring, and connecting procedures for the Ultra3000 and standard Rockwell Automation/ Allen-Bradley motors recommended for use with the Ultra3000.

For power up procedures, troubleshooting tables, and system integration with Ultraware or the ControlLogix® and SoftLogix™ modules/PCI cards (see table below) refer to the *Ultra3000 Digital Servo Drives Integration Manual* (publication 2098-IN005*x*-EN-P). Manuals are available electronically (as a .pdf) or in hardcopy from www.theautomationbookstore.com.

Interface	ControlLogix Motion Module	SoftLogix PCI Card
SERCOS interface™	1756-M*xx*SE	1784-PM16SE
Analog interface	1756-M02AE	1784-PM02AE

Contents of this Manual

Refer to the following listing for the descriptive contents of this installation manual.

Chapter	Title	Contents
	Preface	Describes the purpose, background, and scope of this manual. Also specifies the audience for whom this manual is intended.
1	*Installing Your Ultra3000*	Provides mounting information for the Ultra3000.
2	*Ultra3000 Connector Data*	Provides I/O, encoder, and serial interface connector locations and signal descriptions.
3	*Connecting Your Ultra3000*	Provides connection and wiring information for the Ultra3000.
4	*Troubleshooting Status Indicators*	Provides troubleshooting tables that define the Ultra3000 status LED error codes.
Appendix A	*Specifications and Dimensions*	Provides physical, electrical, environmental, and functional specifications for the Ultra3000.
Appendix B	*Interconnect Diagrams*	Provides interconnect diagrams for the Ultra3000.
Appendix C	*Catalog Numbers and Accessories*	Provides catalog numbers and descriptions of the Ultra3000 and related products.

Product Receiving and Storage Responsibility

You, the customer, are responsible for thoroughly inspecting the equipment before accepting the shipment from the freight company. Check the item(s) you receive against your purchase order. If any items are obviously damaged, it is your responsibility to refuse delivery until the freight agent has noted the damage on the freight bill. Should you discover any concealed damage during unpacking, you are responsible for notifying the freight agent. Leave the shipping container intact and request that the freight agent make a visual inspection of the equipment.

Store the product in its shipping container prior to installation. If you are not going to use the equipment for a period of time, store using the following guidelines.

- Use a clean, dry location
- Maintain an ambient temperature range of -40 to 70° C (-40 to 158° F)
- Maintain a relative humidity range of 5% to 95%, non-condensing
- Store it where it cannot be exposed to a corrosive atmosphere
- Store it in a non-construction area

Related Documentation

The following documents contain additional information concerning related Allen-Bradley products. To obtain a copy, contact your local Allen-Bradley office, distributor, or download them from www.theautomationbookstore.com

For:	Read This Document:	Catalog Number:
Information on configuring and troubleshooting your Ultra3000	*Ultra3000 Digital Servo Drives Integration Manual*	2098-IN005*x*-EN-P
Ultraware Installation Instructions	*Ultraware CD Installation Instructions*	2098-IN002*x*-EN-P
Information on configuring your Ultra3000 using Ultraware	*Ultraware User Manual*	2098-UM001*x*-EN-P
Information on communicating with the Ultra3000 using DeviceNet™	*Ultra3000 DeviceNet Reference Manual*	2098-RM001*x*-EN-P
Information on attaching Ultra3000 drives to a DeviceNet network	*DeviceNet Cable System Planning and Installation Manual*	DN-6.7.2
A description and specifications for the Ultra Family including motors and motor accessories	*Motion Control Selection Guide*	GMC-SG001*x*-EN-P
Application sizing and configuration information	*Motion Book Servo Sizing CD* (v4.0 service pack 4 or above)	Motion Book-*mmmyy*
More detailed information on the use of ControlLogix motion features and application examples	*ControlLogix Motion Module Programming Manual*	1756-RM086*x*-EN-P
ControlLogix SERCOS interface module installation instructions	*3, 8, or 16 Axis SERCOS interface Module Installation Instructions*	1756-IN572*x*-EN-P
ControlLogix Analog Encoder Servo module installation instructions	*Analog Encoder (AE) Servo Module Installation Instructions*	1756-IN047*x*-EN-P
SoftLogix SERCOS interface PCI card installation instructions	*16 Axis PCI SERCOS interface Card Installation Instructions*	1784-IN041*x*-EN-P
SoftLogix Analog Encoder PCI card installation instructions	*PCI 2 Axis Servo Card Installation Instructions*	1784-IN005*x*-EN-P
The instructions needed to program a motion application	*Logix Controller Motion Instruction Set Reference Manual*	1756-RM007*x*-EN-P
Information on configuring and troubleshooting your ControlLogix motion module	*ControlLogix Motion Module Setup and Configuration Manual*	1756-UM006*x*-EN-P
Information on configuring and troubleshooting your SoftLogix PCI card	*SoftLogix Motion Card Setup and Configuration Manual*	1784-UM003*x*-EN-P
Information on proper handling, installing, testing, and troubleshooting fiber-optic cables	*Fiber-Optic Cable Installation and Handling Instructions*	2090-IN010*x*-EN-P
Information, examples, and techniques designed to minimize system failures caused by electrical noise	*System Design for Control of Electrical Noise Reference Manual*	GMC-RM001*x*-EN-P
For declarations of conformity (DoC) currently available from Rockwell Automation	Rockwell Automation Product Certification website	www.ab.com/ certification/ce/docs
An article on wire sizes and types for grounding electrical equipment	*National Electrical Code*	Published by the National Fire Protection Association of Boston, MA.
A glossary of industrial automation terms and abbreviations	*Allen-Bradley Industrial Automation Glossary*	AG-7.1

Conventions Used in this Manual

The following conventions are used throughout this manual.

- Bulleted lists such as this one provide information, not procedural steps
- Numbered lists provide sequential steps or hierarchical information
- Words that you type or select appear in bold
- When we refer you to another location, the section or chapter name appears in italics
- Abbreviations for the Ultra3000 drives, shown in the table below, are used throughout this manual

Ultra3000 Drive	Abbreviation
Ultra3000 with SERCOS interface	Ultra3000-SE
Ultra3000 with DeviceNet interface	Ultra3000-DN

Allen-Bradley Support

Allen-Bradley offers support services worldwide, with over 75 Sales/Support Offices, 512 authorized Distributors and 260 authorized Systems Integrators located throughout the United States alone, plus Allen-Bradley representatives in every major country in the world.

Local Product Support

Contact your local Allen-Bradley representative for:

- Sales and order support
- Product technical training
- Warranty support
- Support service agreements

Technical Product Assistance

If you need technical assistance, contact your local Allen-Bradley representative or Rockwell Automation Technical Support at (440) 646-5800 / www.ab.com/support. Please have the catalog numbers of your products available when you call.

Comments Regarding this Manual

To offer comments regarding the contents of this manual, go to www.ab.com/manuals/gmc and download the Motion Control Problem Report form. Mail or fax your comments to the address/fax number given on the form.

Chapter 1

Installing Your Ultra3000

Chapter Objectives

This chapter provides system installation guidelines and procedures for mounting your Ultra3000. This chapter covers the following topics:

- Complying with European Union Directives
- Ultra3000 System Component Overview
- Before Mounting Your System
- HF Bonding Your System
- Planning Your Panel Layout
- Mounting Your Ultra3000 Drive

ATTENTION

The following information is a guideline for proper installation. The National Electrical Code and any other governing regional or local codes overrule this information. The Allen-Bradley Company cannot assume responsibility for the compliance or the noncompliance with any code, national, local or otherwise, for the proper installation of this system or associated equipment. If you ignore codes during installation, hazard of personal injury and/or equipment damage exists.

Complying with European Union Directives

If this product is installed within the European Union or EEC regions and has the CE mark, the following regulations apply.

Note: Declarations of Conformity (DOCs) to European Union Directives are available on-line at www.ab.com/certification/ce/docs. The web site is the authoritative source for verifying compliance and suitability for use of this and other Rockwell Automation/Allen-Bradley products.

EMC Directive

This unit is tested to meet Council Directive 89/336/EEC Electromagnetic Compatibility (EMC) using a technical construction file and the following standards, in whole or in part:

- EN 50081-2 EMC - Emission Standard, Part 2 - Industrial Environment
- EN 50082-2 EMC - Immunity Standard, Part 2 - Industrial Environment
- EN 61800-3 - Adjustable Speed Electrical Power Drive Systems, Part 3 - EMC Product Standard including specific test methods

The product described in this manual is intended for use in an industrial environment.

Meeting CE Requirements

To meet CE requirements the following components are required:

- Install an AC line filter (2090-UXLF-*xxx* or -HV*xxx*) between the AC power source and the drive input, and as close to the drive as possible (refer to *Appendix C* for available AC line filters). The supply must be grounded for the filter to operate properly.
- Connect auxiliary input power (if required) from the load side of the AC line filter to the drive.
- Use 2090 series motor power and feedback cables and terminate the motor power cable shields to the chassis clamp provided (refer to *Chapter 3* for wiring instructions).
- When installing the Ultra3000 system inside an enclosure, run input power wiring (grounded to the enclosure) in conduit outside of the enclosure.
- Separate signal and power cables as shown in *Planning Your Panel Layout* of this chapter.

Low Voltage Directive

These units are tested to meet Council Directive 73/23/EEC Low Voltage Directive. The EN 60204-1 Safety of Machinery-Electrical Equipment of Machines, *Part 1-Specification for General Requirements* standard applies in whole or in part. Additionally, the standard EN 50178 *Electronic Equipment for use in Power Installations* applies in whole or in part.

Refer to *Appendix B* for interconnect information.

Ultra3000 System Component Overview

This section provides an overview of the Ultra3000 system components and a typical installation.

Ultra3000 System Component	Catalog Numbers	Description
Ultra3000 Drives	2098-DSD-*xxx* and -*xxx*X	Ultra3000 and Ultra3000 with indexing available with 500W, 1, 2, 3, 7.5 and 15 kW continuous output and 230V input power.
	2098-DSD-HV*xxx*, and -HV*xxx*X	Ultra3000 and Ultra3000 with indexing available with 3, 5, 10, 15, and 22 kW continuous output and 460V input power.
Ultra3000-SE SERCOS interface Drives	2098-DSD-*xxx*-SE	Ultra3000 with SERCOS interface available with 500W, 1, 2, 3, 7.5 and 15 kW continuous output and 230V input power.
	2098-DSD-HV*xxx*-SE	Ultra3000 with SERCOS interface available with 3, 5, 10, 15, and 22 kW continuous output and 460V input power.
Ultra3000-DN DeviceNet Drives	2098-DSD-*xxx*-DN and -*xxx*X-DN	Ultra3000 with DeviceNet and Ultra3000 with indexing DeviceNet available with 500W, 1, 2, 3, 7.5 and 15 kW continuous output with 230V input power.
	2098-DSD-HV*xxx*-DN and -HV*xxx*X-DN	Ultra3000 with DeviceNet and Ultra3000 with indexing DeviceNet available with 3, 5, 10, 15, and 22 kW continuous output with 460V input power.
ControlLogix/ SoftLogix Platforms	1756-M*xx*SE module 1784-PM16SE PCI card	The SERCOS interface module/PCI card serves as a link between the ControlLogix/SoftLogix platform and Ultra3000 system. The communication link uses the IEC 61491 SErial Real-time COmmunication System (SERCOS) protocol over a fiber-optic cable.
RSLogix™ 5000 software	9324-RLD300ENE	RSLogix 5000 provides support for programming, commissioning, and maintaining the Logix family of controllers.
Ultraware Software	2098-UWCPRG	The Ultra3000 Analog and DeviceNet drives are configured using Ultraware software.
Servo Motors	MP-Series, 1326AB, F-, H-, N-, and Y-Series	The MP-Series (Low Inertia, Integrated Gear, and Food Grade) 230 and 460V, 1326AB (M2L/S2L) 460V, and F-, H-, N-, and Y-Series 230V motors are available for use with the Ultra3000 drives.
Cables	Motor Power, Feedback, and Brake cables	Motor power, feedback, and brake cables include integral molded, bayonet style, quick connect/ quick-release connectors at the motor. Power and brake cables have flying leads on the drive end and straight connectors that connect to servo motors. Standard feedback cables have angled connectors (45°) on the drive end and straight connectors that connect to servo motors.
	Fiber-Optic cables	SERCOS fiber-optic cables are available in enclosure only, PVC, nylon, and glass with connectors at both ends.
AC Line Filters	2090-UXLF-*xxx*	AC line filters with 6, 10, 23, 32, 36, and 50A are available for Ultra3000 (230V) drive systems.
	2090-UXLF-HV*xxx*	AC line filters with 23, 30, and 50A are available for Ultra3000 (460V) drive systems.
External Shunt Modules	2090-UCSR-*xxxx*, 9101-1183, and 2090-SR*xxx*-*xx*	External shunt modules are available when the Ultra3000 internal shunt capability is exceeded.

Note: Refer to *Appendix C* for a complete list of catalog numbers for the Ultra3000 system components listed above.

The typical Ultra3000 system installation includes the following, as shown in the figures below.

Figure 1.1
Ultra3000-SE (SERCOS) Digital Servo Drive System Overview

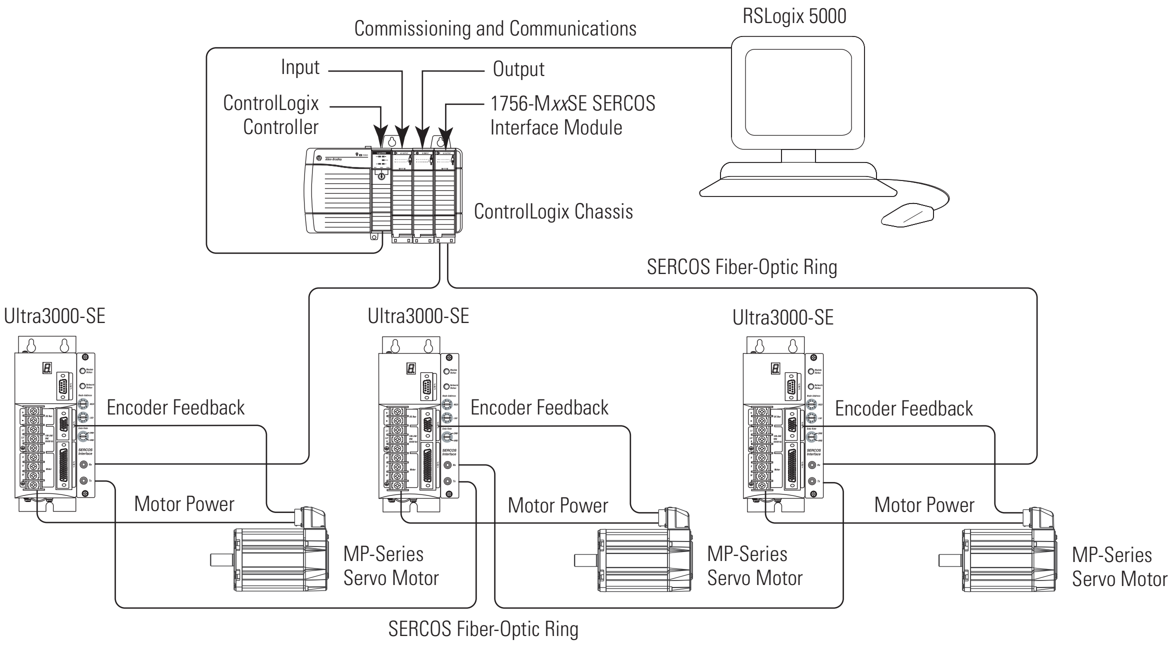


Figure 1.2
Ultra3000 Digital Servo Drive System Overview

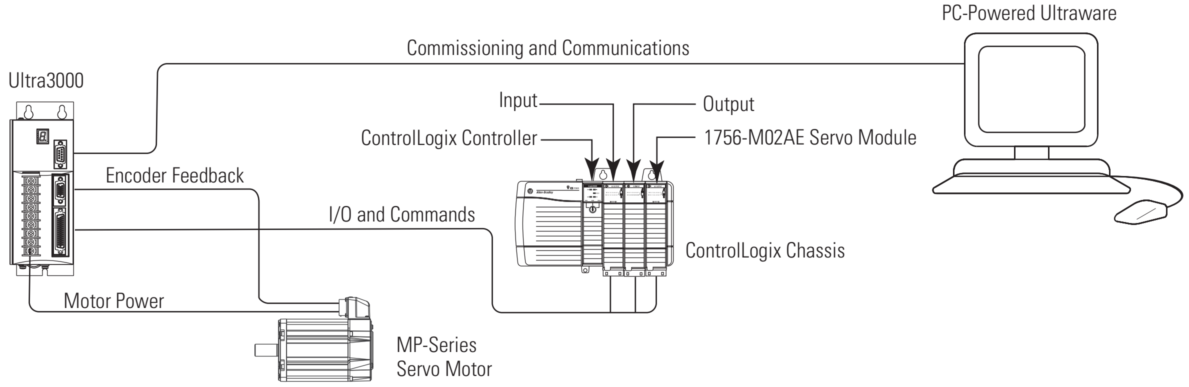
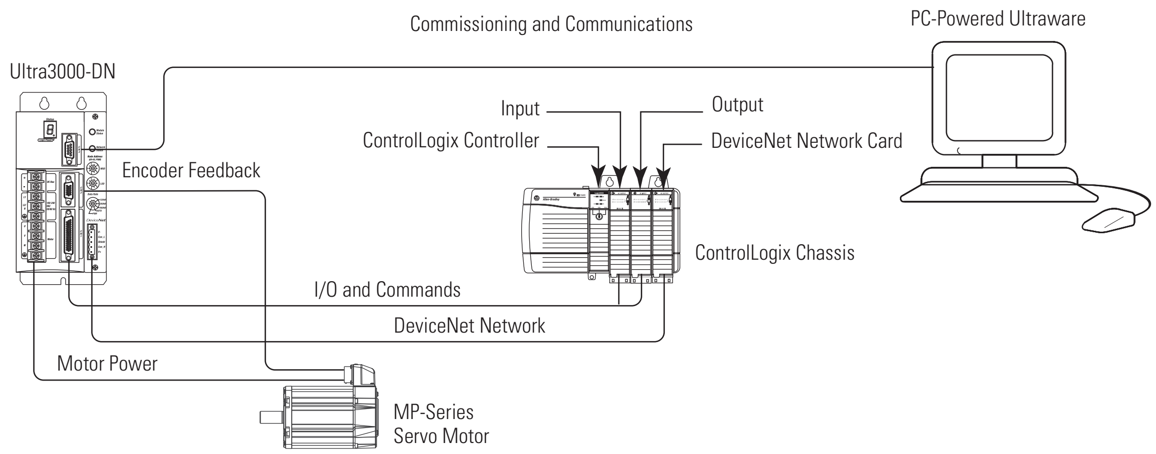


Figure 1.3
Ultra3000-DN (DeviceNet) Digital Servo Drive System Overview



Before Mounting Your System

Before you mount your Ultra3000 system make sure you understand the following:

- how to unpack the system
- the minimum mounting requirements

Unpacking Modules

Each Ultra3000 ships with the following:

- One Ultra3000 drive
- One installation manual (publication 2098-IN003*x*-EN-P)

Remove all packing material, wedges, and braces from within and around the components. After unpacking, check the item(s) name plate catalog number against the purchase order.

System Mounting Requirements

There are several things that you need to take into account when preparing to mount the Ultra3000:

- The Ultra3000 must be enclosed in a grounded conductive enclosure offering protection as defined in standard EN 60529 (IEC 529) to IP22 such that they are not accessible to an operator or unskilled person, in order to comply with UL® and CE requirements. A NEMA 4X enclosure exceeds these requirements providing protection to IP66.
- The ambient temperature of the location in which you will install the Ultra3000 must not exceed 55° C (131° F).
- You must install the Ultra3000 vertically on the panel (refer to Figure 1.4 for mounting orientation).
- You must install the panel on a flat, rigid, vertical surface that won't be subjected to shock, vibration, moisture, oil mist, dust, or corrosive vapors.
- You need to maintain minimum clearances (refer to Figure 1.4) for proper airflow, easy module access, and proper cable bend radius.
- The Ultra3000 can operate at elevations to 1000 m (3280 ft) without derating, however, the continuous current rating must be de-rated by 3% for each additional 300 m (984 ft) up to 3000 m (9842 ft). Consult your local Allen-Bradley representative prior to operating above 3000 m (9842 ft).

ATTENTION

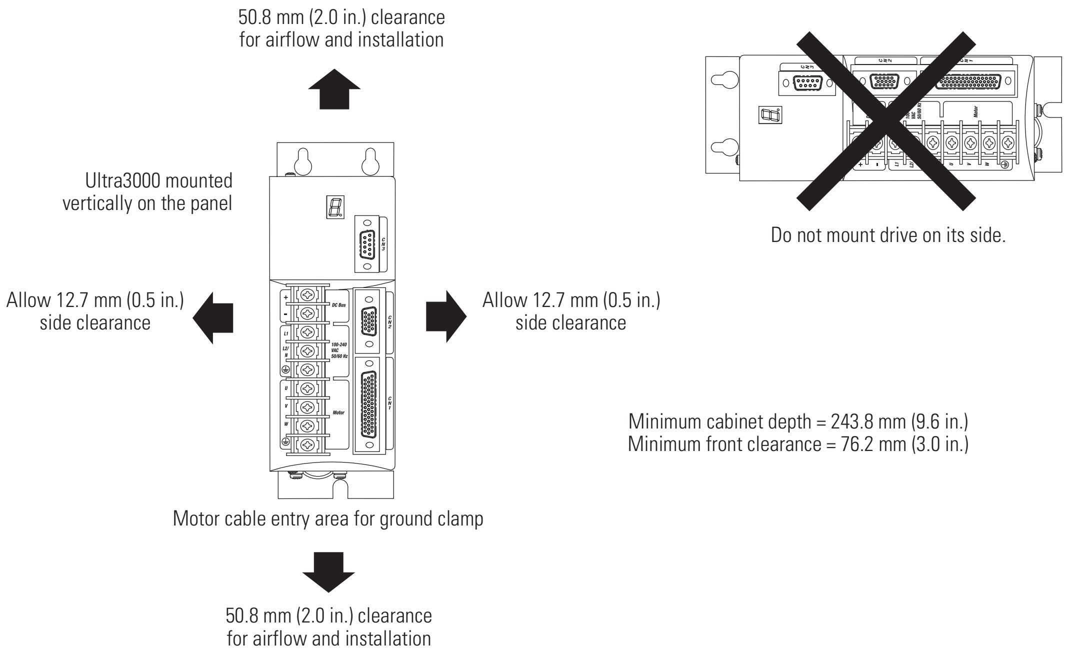
Plan the installation of your system so that you can perform all cutting, drilling, tapping, and welding with the system removed from the enclosure. Because the system is of the open type construction, be careful to keep any metal debris from falling into it. Metal debris or other foreign matter can become lodged in the circuitry, which can result in damage to components.

Refer to *Appendix A* for mounting dimensions, power dissipation, and environmental specifications for the Ultra3000.

Ventilation Requirements

This section provides information to assist you in sizing your cabinet and locating your Ultra3000 drive(s) inside the cabinet.

Figure 1.4
Minimum Clearance Requirements



50.8 mm (2.0 in.) clearance
for airflow and installation

Ultra3000 mounted
vertically on the panel

Allow 12.7 mm (0.5 in.)
side clearance

Allow 12.7 mm (0.5 in.)
side clearance

Motor cable entry area for ground clamp

50.8 mm (2.0 in.) clearance
for airflow and installation

Do not mount drive on its side.

Minimum cabinet depth = 243.8 mm (9.6 in.)
Minimum front clearance = 76.2 mm (3.0 in.)

IMPORTANT If the cabinet is ventilated, use filtered or conditioned air to prevent the accumulation of dust and dirt on electronic components. The air should be free of oil, corrosives, or electrically conductive contaminates.

Refer to *Appendix A* for Ultra3000 power dissipation specifications.

Sizing an Enclosure

As an additional aid in sizing an enclosure, with no active method of heat dissipation, either of the following approximate equations can be used:

Metric	**Standard English**
$A = \frac{0.38Q}{1.8T - 1.1}$	$A = \frac{4.08Q}{T - 1.1}$
Where T is temperature difference between inside air and outside ambient (°C), Q is heat generated in enclosure (Watts), and A is enclosure surface area (m^2). The exterior surface of all six sides of an enclosure is calculated as	Where T is temperature difference between inside air and outside ambient (°F), Q is heat generated in enclosure (Watts), and A is enclosure surface area (ft^2). The exterior surface of all six sides of an enclosure is calculated as
A = 2dw + 2dh + 2wh	A = (2dw + 2dh + 2wh) / 144
Where d (depth), w (width), and h (height) are in meters.	Where d (depth), w (width), and h (height) are in inches.

Transformer Sizing

The Ultra3000 does not require isolation transformers. However, a transformer may be required to match the voltage requirements of the controller to the available service. To size a transformer for the main AC power inputs, the power output (KVA) of each axis must be known. This can be derived by calculating the horsepower for each axis and converting that horsepower into units of watts. If you are supplying power to more than one motor and an Ultra3000, simply add the kW ratings together from each calculation to get a system kW total.

IMPORTANT If using an autotransformer, ensure that the phase to neutral/ground voltages do not exceed the input voltage ratings of the drive.

Definitions:

kW = power or real power
KVA = apparent power

Transformer KVA rating = (Sum of average output power of each axis) x 2.0.

IMPORTANT If you are using the Rockwell Automation/ Allen-Bradley system sizing program, the average speed and average torque data has already been calculated and can be used in the above equation. If you are not sure of the exact speed and torque in your application, another approach is to look at the speed/torque curve for your Ultra3000/motor combination and use the values for the worst case continuous speed and torque.

IMPORTANT Calculations are multiplied by a factor to compensate for the power and loss elements within a power system. A factor of 2.0 is used with a single phase system and a factor of 1.5 is used with a three phase system. This factor should minimize the effects of the secondary line voltage sagging in the transformer during peak current periods.

Example: sizing a transformer to the voltage requirements of an 2098-DSD-020 and MPL-A320P motor:

$$KVA = \frac{Speed(RPM) x Torque(lb - in)}{63,025} x \frac{746 Watts}{HP} x \frac{KVA}{1000 Watts} x 2.0$$

$$KVA = \frac{(5,000(RPM)) X 17.7(lb - in)}{42,250}$$

$$Transformer\ Size = 2.1\ KVA$$

The speed/torque curve information for 230V motors is based upon an Ultra3000 input voltage of 230V ac. For a 115V ac input voltage, the maximum speed can be reduced up to one half.

Fuse Sizing

In the United States, the National Electric Code (NEC) specifies that fuses must be selected based on the motor full load amperage (FLA). The typical fuse size should be 300% of the motor FLA for non-time delay fuses (and time-delay class CC fuses) or 175% of motor FLA for time delay fuses. If these ratings are not high enough for starting currents, the NEC allows non-time delay fuses (and time-delay class CC fuses) to be sized up to 400% of the motor FLA and time-delay fuses to be sized up to 225% of the motor FLA.

In most cases, fuses selected to match the drive input current rating will meet the NEC requirements and provide the full drive capabilities. Dual element, time delay (slow acting) fuses should be used to avoid nuisance trips during the inrush current of power initialization. Refer to the section *Ultra3000 Power Specifications* in *Appendix A* for input current and inrush current specifications.

The Ultra3000 utilizes solid state motor short circuit protection rated as shown in the table below.

Drive Models:	**Input Power Type**	**Short Circuit Current Rating with No Fuse Restrictions:**	**Short Circuit Current Rating with Fuse Restrictions:**
2098-DSD-*xxx-xx* or *xxx*X-*xx*	Input Power and Auxiliary Input Power	Suitable for use on a circuit capable of delivering not more than 5000 rms symmetrical amperes, 240V maximum.	Suitable for use on a circuit capable of delivering not more than 200,000 rms symmetrical amperes, 240V maximum, when protected by high interrupting capacity, current limiting fuses meeting UL 198C (Class CC, G, J, L, R, T).
2098-DSD-HV*xxx-xx* or HV*xxx*X-*xx*		Suitable for use on a circuit capable of delivering not more than 5000 rms symmetrical amperes, 480V maximum.	Suitable for use on a circuit capable of delivering not more than 200,000 rms symmetrical amperes, 480V maximum, when protected by high interrupting capacity, current limiting fuses meeting UL 198C (Class CC, G, J, L, R, T).

Wiring to the auxiliary power terminals (L1 AUX and L2/N AUX) of the drive should be 2.5 mm^2 (14 AWG) minimum and fusing for the auxiliary power should be selected to properly protect the wire. For example, if 60° C (140° F) wire is used, the fuse should not exceed 8A. If 75° C (167° F) wire is used, the fuse should not exceed 13A. Refer to *Fuse Specifications* in *Appendix A* for fuse examples.

HF Bonding Your System

Bonding is the practice of connecting metal chassis, assemblies, frames, shields and enclosures to reduce the effects of electromagnetic interference (EMI). For more information on the concept of high-frequency (HF) bonding, the ground plane principle, and electrical noise reduction, refer to the *System Design for Control of Electrical Noise Reference Manual* (publication GMC-RM001*x*-EN-P).

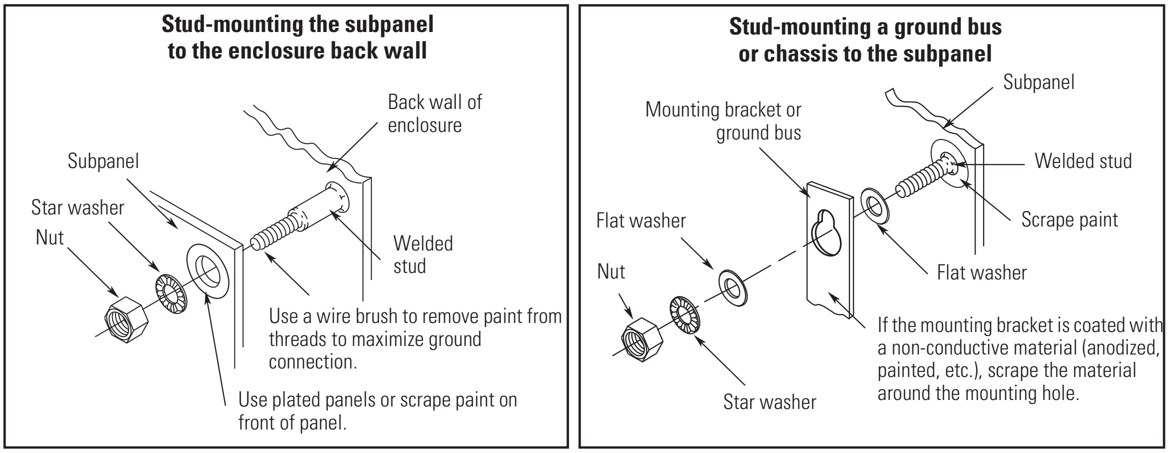
Bonding Modules

Unless specified, most paints are not conductive and they act as insulators. To achieve a good bond between modules and the subpanel, surfaces need to be paint-free or plated. Bonding metal surfaces creates a low-impedance exit path for high-frequency energy.

IMPORTANT To improve the bond between the drive and subpanel, construct your subpanel out of zinc plated (paint-free) steel.

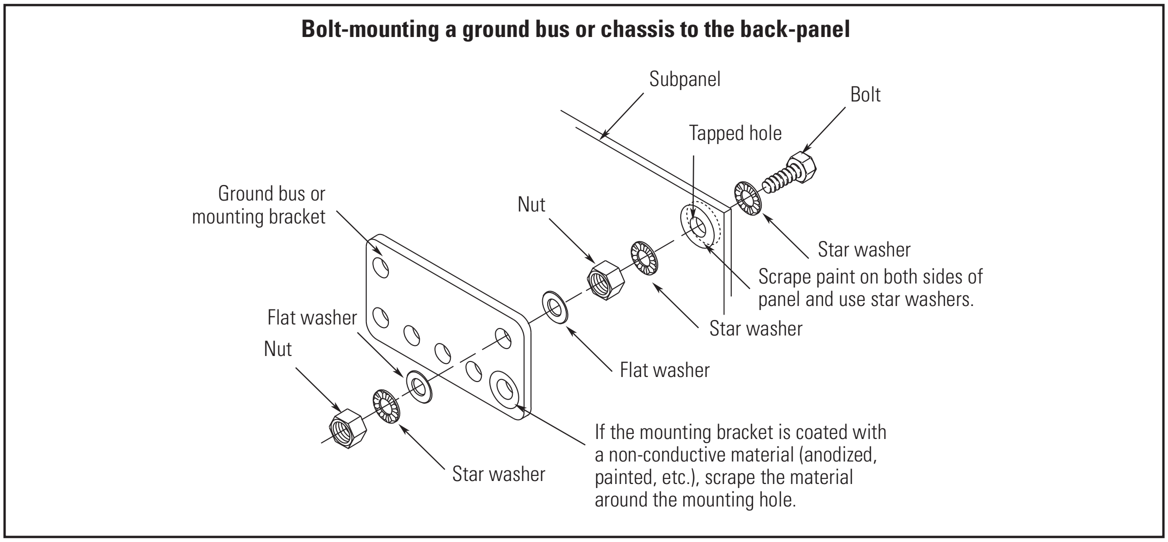
Improper bonding blocks that direct exit path and allows high-frequency energy to travel elsewhere in the cabinet. Excessive high-frequency energy can effect the operation of other microprocessor controlled equipment. The illustrations that follow (refer to Figure 1.5) show details of recommended bonding practices for painted panels, enclosures, and mounting brackets.

Figure 1.5
Recommended Bonding Practices



Stud-mounting the subpanel to the enclosure back wall

Stud-mounting a ground bus or chassis to the subpanel

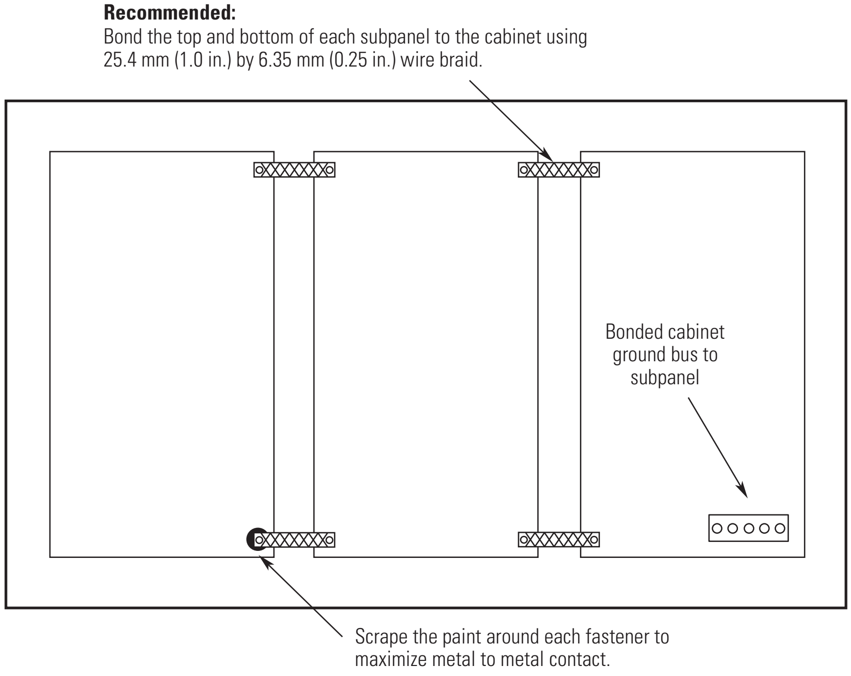


Bolt-mounting a ground bus or chassis to the back-panel

Bonding Multiple Subpanels

Bonding multiple subpanels creates a common low impedance exit path for the high frequency energy inside the cabinet. Subpanels that are not bonded together may not share a common low impedance path. This difference in impedance may affect networks and other devices that span multiple panels. Refer to the figure below for recommended bonding practices.

Figure 1.6
Multiple Subpanels and Cabinet



Planning Your Panel Layout

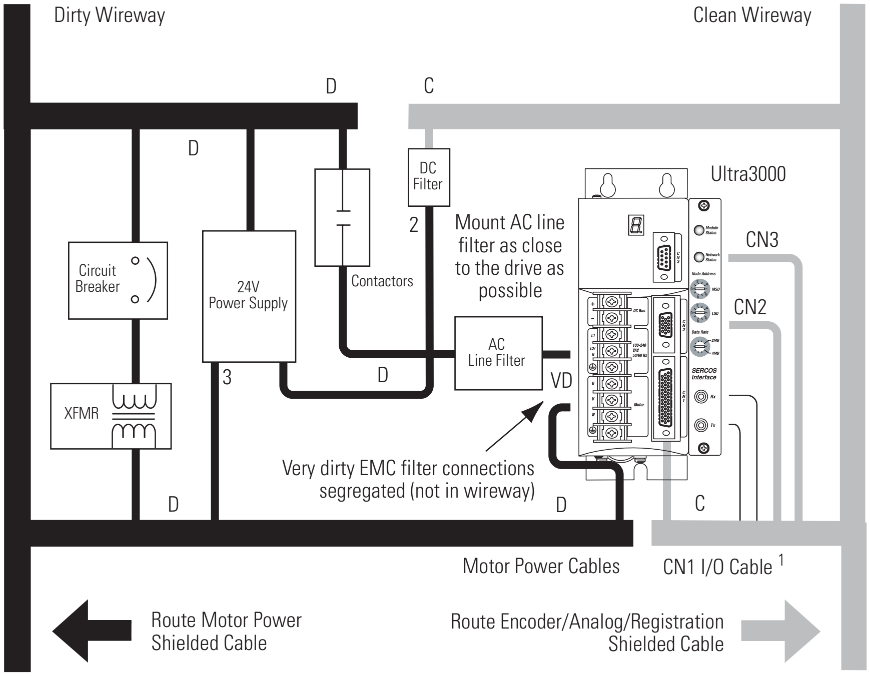
This section outlines the practices which minimize the possibility of noise-related failures as they apply specifically to Ultra3000 installations. For more information on the concept of electrical noise reduction, refer to *System Design for Control of Electrical Noise* (publication GMC-RM001*x*-EN-P).

Establishing Noise Zones

Observe the following guidelines when laying out your panel (refer to Figure 1.7 for zone locations).

- The clean zone (C) is above and beneath the Ultra3000 and includes CN1, CN2, CN3, and the DC filter (grey wireways).
- The dirty zone (D) is left of the Ultra3000 (black wireways) and includes the circuit breakers, transformer, AC line filter, contactors, 24V dc power supply, and motor power cables.
- The very dirty zone (VD) is limited to where the AC line (EMC) filter AC output jumpers over to the Ultra3000. Shielded cable is required only if the very dirty cables enter a wireway.
- The SERCOS fiber-optic cables are immune to electrical noise.

Figure 1.7
Establishing Noise Zones



[1] If I/O cable contains (dirty) relay wires, route cable with motor power wires in dirty wireway.

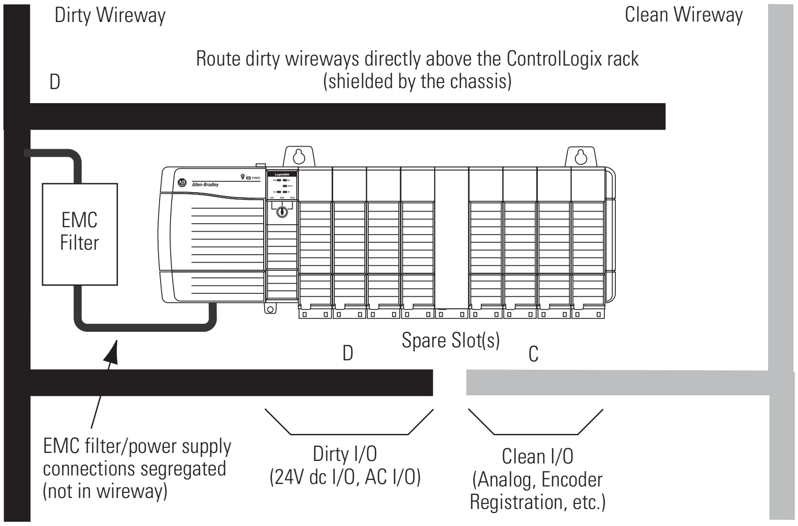
[2] This is a clean 24V dc available for CN1 I/O power supply. The 24V enters the clean wireway and exits to the right.

[3] This is a dirty 24V dc available for motor brakes and contactors. The 24V enters the dirty wireway and exits to the left.

Observe the following guidelines when installing your 1756-M*xx*SE SERCOS interface module (refer to Figure 1.8 for zone locations).

- The clean zone (C) is beneath the less noisy modules (I/O, analog, encoder, registration, etc. (grey wireway).
- The dirty zone (D) is above and below the power supply and noisy modules (black wireway).
- The SERCOS fiber-optic cables are immune to electrical noise.

Figure 1.8
Establishing Noise Zones (ControlLogix)



Cable Categories for the Ultra3000

The table below indicates the zoning requirements of cables connecting to the Ultra3000.

Wire/Cable	Connector	Zone			Method	
		Very Dirty	Dirty	Clean	Ferrite Sleeve	Shielded Cable
DC-/DC+	TB1	X				
L1, L2, L3 (shielded cable)			X			X
L1, L2, L3 (unshielded cable)		X				
U, V, W (motor power)			X			X
Registration Wiring	CN1			X		X
Other 24V Wiring	CN1		X			
Motor Feedback	CN2			X		X
Serial Communications	CN3			X		X
Fiber-Optic	Rx and Tx	No Restrictions				

The table below indicates the zoning requirements of cables connecting to the External Shunt Resistor Kit.

Wire/Cable	Connector	Zone			Method	
		Very Dirty	Dirty	Clean	Ferrite Sleeve	Shielded Cable
Shunt Connections (shielded option)	TB2		X			X
Shunt Connections (unshielded option)		X				
Fan (if present)	N/A		X			

Mounting Guidelines to Reduce Electrical Noise

When mounting an AC line (EMC) filter or external shunt resistor refer to the sections below for guidelines designed to reduce system failures caused by excessive electrical noise.

ATTENTION High voltage exists in AC line filters. The filter must be grounded properly before applying power. Filter capacitors retain high voltages after power removal. Before handling the equipment, voltages should be measured to determine safe levels. Failure to observe this precaution could result in personal injury.

AC Line Filters

Observe the following guidelines when mounting your AC line (EMC) filter (refer to Figure 1.7 for an example).

- Mount the AC line filter and bonded cabinet ground bus on the same panel as the Ultra3000, and as close to the Ultra3000 as possible.
- Good HF bonding to the panel is critical. For painted panels, refer to Figure 1.5.
- Segregate input and output wiring as far as possible.

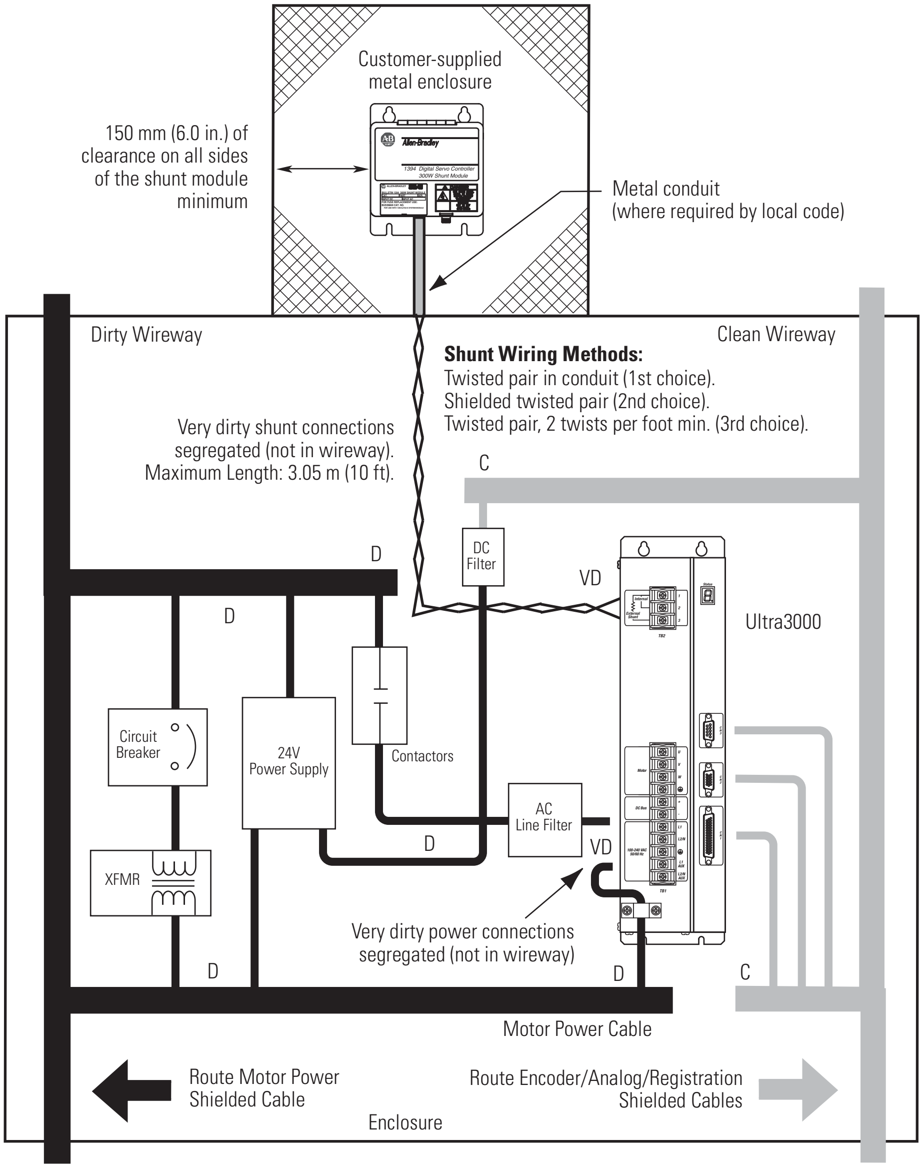
IMPORTANT CE test certification applies only to AC line filter and single drive. Multiple drive loads may perform satisfactorily, but the user takes legal responsibility.

External Shunt Resistor

Observe the following guidelines when mounting your external shunt resistor (refer to Figure 1.9 and for an example).

- Mount circuit components and wiring in the very dirty zone or in an external shielded enclosure. Run shunt power and fan wiring inside metal conduit to minimize the effects of EMI and RFI.
- Mount resistors (other than metal-clad) in a shielded and ventilated enclosure outside the cabinet.
- Keep unshielded wiring as short as possible. Keep shunt wiring as flat to the cabinet as possible.

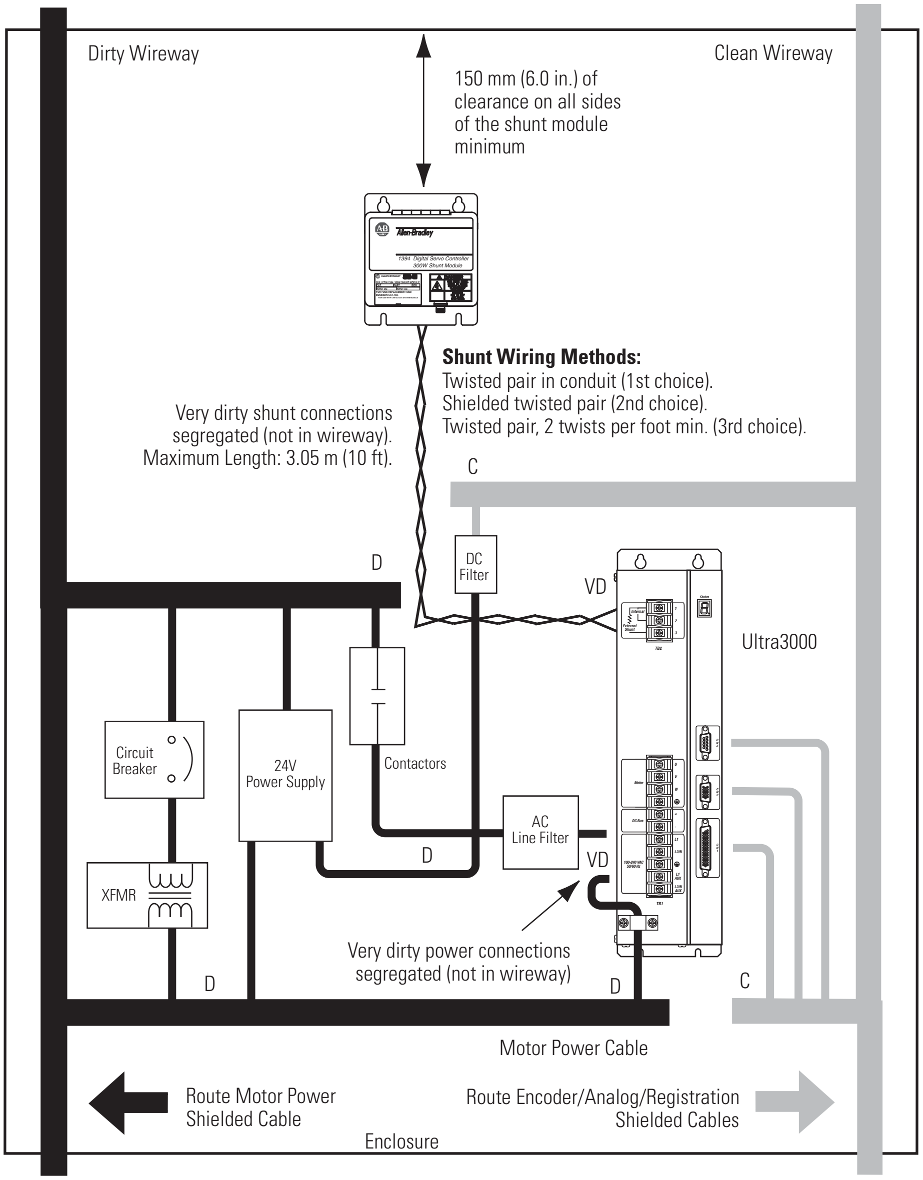
Figure 1.9
External Shunt Resistor Outside the Enclosure



When mounting your shunt module inside the enclosure, follow these additional guidelines (refer to Figure 1.10 and for an example).

- Metal-clad modules can be mounted anywhere in the dirty zone, but as close to the Ultra3000 as possible.
- Shunt power wires can be run with motor power cables.
- Keep unshielded wiring as short as possible. Keep shunt wiring as flat to the cabinet as possible.
- Separate shunt power cables from other sensitive, low voltage signal cables.

Figure 1.10
External Shunt Resistor Inside the Enclosure



Mounting Your Ultra3000 Drive

The procedures in this section assume you have prepared your panel and understand how to bond your system. For installation instructions regarding other equipment and accessories, refer to the instructions that came with each of the accessories for their specific requirements.

ATTENTION

This drive contains ESD (Electrostatic Discharge) sensitive parts and assemblies. You are required to follow static control precautions when you install, test, service, or repair this assembly. If you do not follow ESD control procedures, components can be damaged. If you are not familiar with static control procedures, refer to Allen-Bradley publication 8000-4.5.2, *Guarding Against Electrostatic Damage* or any other applicable ESD Protection Handbook.

To mount your Ultra3000 drive:

1. Layout the position for the Ultra3000 and accessories in the enclosure (refer to *Establishing Noise Zones* for panel layout recommendations). Mounting hole dimensions for the Ultra3000 are shown in *Appendix A*.

2. Attach the Ultra3000 to the cabinet, first using the upper mounting slots of the drive and then the lower. The recommended mounting hardware is M5 metric (1/4-20) or #10 MS bolts. Observe bonding techniques as described in *HF Bonding Your System*.

   IMPORTANT

   To improve the bond between the Ultra3000 and subpanel, construct your subpanel out of zinc plated (paint-free) steel.

3. Tighten all mounting fasteners.

Ultra3000 Connector Data

Chapter Objectives

This chapter provides I/O, encoder, and serial interface connector locations and signal descriptions for your Ultra3000. This chapter includes:

- Understanding Ultra3000 Connectors
- Understanding Ultra3000 I/O Specifications
- Understanding Motor Encoder Feedback Specifications
- Understanding Auxiliary Encoder Feedback Specifications
- Understanding the Serial Interface

Switch and LED locations are shown, however for switch and LED configuration, refer to the *Ultra3000 Digital Servo Drives Integration Manual* (publication 2098-IN005*x*-EN-P).

Understanding Ultra3000 Connectors

The following table provides a brief description of the Ultra3000 front panel connectors and describes the connector type.

Designator	**Description**	**Connector**
CN1	User Input/Output	44-pin high-density D-shell
CN2	Motor Feedback	15-pin high-density D-shell
CN3	Serial Port	9-pin standard D-shell
TB	DC bus, Motor and AC power	9-position screw style barrier terminal strip (2098-DSD-005*x-xx*, -010*x-xx*, and -020*x-xx*)
TB1	DC bus, Motor, AC power, and auxiliary AC power	11- or 12-position screw style barrier terminal strip (2098-DSD-030*x-xx*, -075*x-xx*, -150*x-xx,* HV*xxx-xx*, and HV*xxx*X-*xx*)
TB2	Shunt	3-position screw style barrier terminal strip (2098-DSD-030*x-xx*, -075*x-xx*, -150*x-xx,* HV*xxx-xx*, and HV*xxx*X-*xx*)

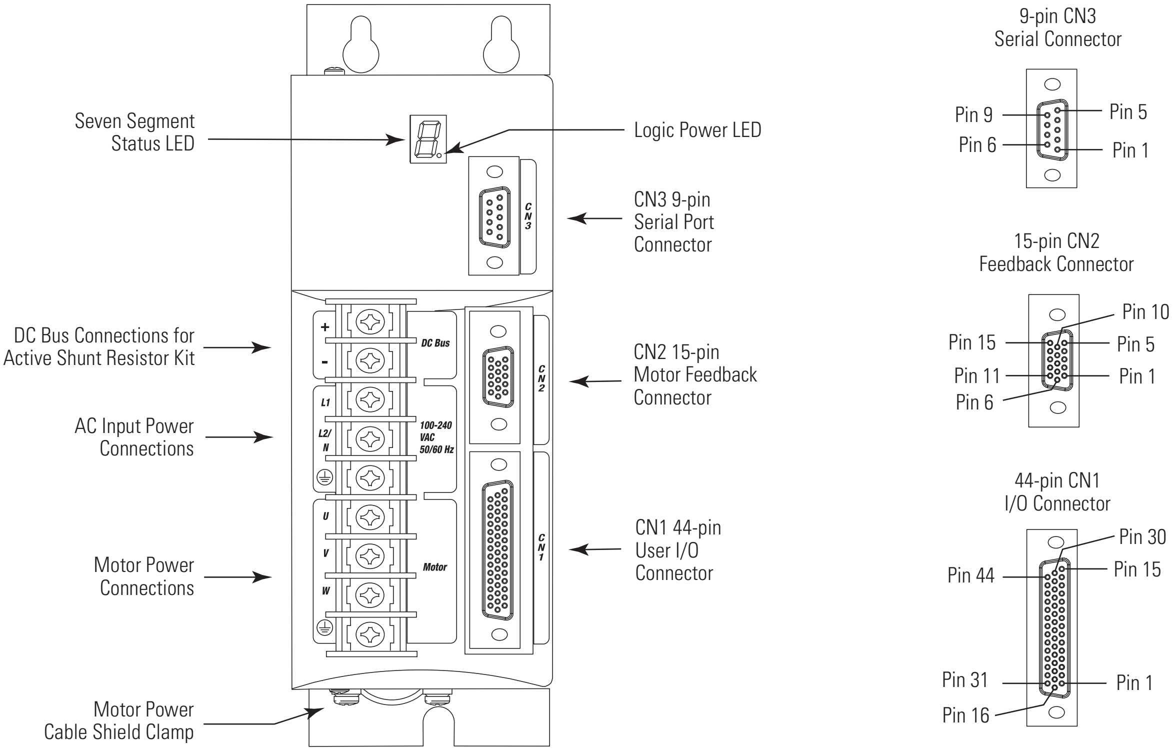
All signal connections on the Ultra3000 use commonly available D-shell type connectors.

For connector pin-outs and the location of connectors, switches, and status LEDs on:	**Refer to:**
2098-DSD-*xxx* and -HV*xxx* Ultra3000 drives	Figures 2.1-2.4 and the tables that follow on pages 2-2 through 2-9.
2098-DSD-*xxx* and -HV*xxx* Ultra3000 drives with SERCOS interface	Figures 2.5-2.8 and the tables that follow on pages 2-10 through 2-17.
2098-DSD-*xxx* and -HV*xxx* Ultra3000 drives with DeviceNet interface	Figures 2.9-2.12 and the tables that follow on pages 2-18 through 2-25.

Ultra3000 Front Panel Connections

Use the figure below to locate the front panel connections on the Ultra3000 230V drives (500W, 1 kW, and 2 kW).

Figure 2.1
Ultra3000 Front Panel Connections
for 2098-DSD-005, -005X, -010, -010X, -020, and -020X



Serial Port Connector

The following table provides the signal descriptions and pin-outs for the CN3 serial port (9-pin) connector.

CN3 Pin	Description	Signal
1	RS-422/RS-485 Input+	RCV+
2	RS-232 Input	RCV
3	RS-232 Output	XMT
4	RS-422/RS-485 Output+	XMT+
5	Common	COM
6	Reserved	–
7	RS-422/RS-485 Input-	RCV-
8	RS-422/RS-485 Output-	XMT-
9	Reserved	–

I/O Connector

The following table provides the signal descriptions and pin-outs for the CN1 I/O (44-pin) connector.

CN1 Pin	Description	Signal
1	Auxiliary Encoder Power Out (+5V)	EPWR
2	Common	ECOM
3	Auxiliary Logic Power In (+5V)	AUXPWR
4	Auxiliary A+/Step+/CW+	AX+
5	Auxiliary A-/Step-/CW-	AX-
6	Auxiliary B+/Dir+/CCW+	BX+
7	Auxiliary B-/Dir-/CCW-	BX-
8	Auxiliary Encoder Ch I+	IX+
9	Auxiliary Encoder Ch I-	IX-
10	Unbuffered Motor Encoder Ch A+	AM+
11	Unbuffered Motor Encoder Ch A-	AM-
12	Unbuffered Motor Encoder Ch B+	BM+
13	Unbuffered Motor Encoder Ch B-	BM-
14	Unbuffered Motor Encoder Ch I+	IM+
15	Unbuffered Motor Encoder Ch I-	IM-
16	Buffered Motor Encoder Ch A+	AMOUT+
17	Buffered Motor Encoder Ch A-	AMOUT-
18	Buffered Motor Encoder Ch B+	BMOUT+
19	Buffered Motor Encoder Ch B-	BMOUT-
20	Buffered Motor Encoder Ch I+	IMOUT+
21	Buffered Motor Encoder Ch I-	IMOUT-
22	Common	ACOM

CN1 Pin	Description	Signal
23	Programmable Analog Output	AOUT
24	Analog Current Limit Input	ILIMIT
25	Command +	COMMAND+
26	Command -	COMMAND-
27	I/O Common	IOCOM
28	I/O Common	IOCOM
29	I/O Power	IOPWR
30	I/O Power	IOPWR
31	Digital Input 1	INPUT1
32	Digital Input 2	INPUT2
33	Digital Input 3	INPUT3
34	Digital Input 4	INPUT4
35	Digital Input 5	INPUT5
36	Digital Input 6	INPUT6
37	Digital Input 7	INPUT7
38	Digital Input 8	INPUT8
39	Digital Output 1	OUTPUT1
40	Digital Output 2	OUTPUT2
41	Digital Output 3	OUTPUT3
42	Digital Output 4	OUTPUT4
43	Normally Open Relay Output+	RELAY+
44	Normally Open Relay Output-	RELAY-

Motor Encoder Connector

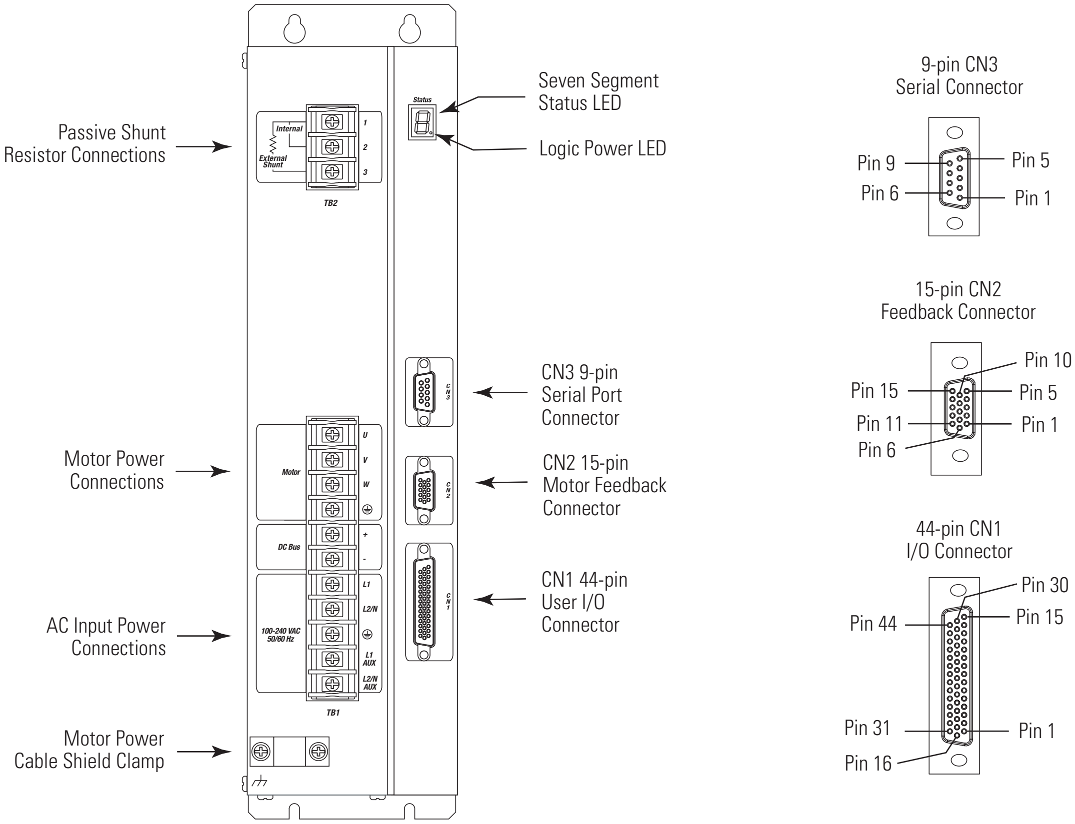
The following table provides the signal descriptions and pin-outs for the CN2 motor encoder (15-pin) connector.

CN2 Pin	Description	Signal
1	Channel A+/Sine Differential Input+	AM+
2	Channel A-/Sine Differential Input-	AM-
3	Channel B+/Cosine Differential Input+	BM+
4	Channel B-/Cosine Differential Input-	BM-
5	Channel I+/Index Pulse+	IM+
6	Common	ECOM
7	Reserved	–
8	Commutation Channel S3	S3

CN2 Pin	Description	Signal
9	Positive Overtravel Limit	+LIMIT
10	Channel I-/Index Pulse-	IM-
11	Thermostat	TS
12	Commutation Channel S1	S1
13	Commutation Channel S2	S2
14	Encoder Power (+5V)	EPWR_5V
15	Negative Overtravel Limit	-LIMIT

Use the figure below to locate the front panel connections on the Ultra3000 230V drives (3 kW).

Figure 2.2
Ultra3000 Front Panel Connections for 2098-DSD-030 and -030X



Serial Port Connector

The following table provides the signal descriptions and pin-outs for the CN3 serial port (9-pin) connector.

CN3 Pin	Description	Signal
1	RS-422/RS-485 Input+	RCV+
2	RS-232 Input	RCV
3	RS-232 Output	XMT
4	RS-422/RS-485 Output+	XMT+
5	Common	COM
6	Reserved	–
7	RS-422/RS-485 Input-	RCV-
8	RS-422/RS-485 Output-	XMT-
9	Reserved	–

I/O Connector

The following table provides the signal descriptions and pin-outs for the CN1 I/O (44-pin) connector.

CN1 Pin	Description	Signal
1	Auxiliary Encoder Power Out (+5V)	EPWR
2	Common	ECOM
3	Reserved	–
4	Auxiliary A+/Step+/CW+	AX+
5	Auxiliary A-/Step-/CW-	AX-
6	Auxiliary B+/Dir+/CCW+	BX+
7	Auxiliary B-/Dir-/CCW-	BX-
8	Auxiliary Encoder Ch I+	IX+
9	Auxiliary Encoder Ch I-	IX-
10	Unbuffered Motor Encoder Ch A+	AM+
11	Unbuffered Motor Encoder Ch A-	AM-
12	Unbuffered Motor Encoder Ch B+	BM+
13	Unbuffered Motor Encoder Ch B-	BM-
14	Unbuffered Motor Encoder Ch I+	IM+
15	Unbuffered Motor Encoder Ch I-	IM-
16	Buffered Motor Encoder Ch A+	AMOUT+
17	Buffered Motor Encoder Ch A-	AMOUT-
18	Buffered Motor Encoder Ch B+	BMOUT+
19	Buffered Motor Encoder Ch B-	BMOUT-
20	Buffered Motor Encoder Ch I+	IMOUT+
21	Buffered Motor Encoder Ch I-	IMOUT-
22	Common	ACOM

CN1 Pin	Description	Signal
23	Programmable Analog Output	AOUT
24	Analog Current Limit Input	ILIMIT
25	Command +	COMMAND+
26	Command -	COMMAND-
27	I/O Common	IOCOM
28	I/O Common	IOCOM
29	I/O Power	IOPWR
30	I/O Power	IOPWR
31	Digital Input 1	INPUT1
32	Digital Input 2	INPUT2
33	Digital Input 3	INPUT3
34	Digital Input 4	INPUT4
35	Digital Input 5	INPUT5
36	Digital Input 6	INPUT6
37	Digital Input 7	INPUT7
38	Digital Input 8	INPUT8
39	Digital Output 1	OUTPUT1
40	Digital Output 2	OUTPUT2
41	Digital Output 3	OUTPUT3
42	Digital Output 4	OUTPUT4
43	Normally Open Relay Output+	RELAY+
44	Normally Open Relay Output-	RELAY-

Motor Encoder Connector

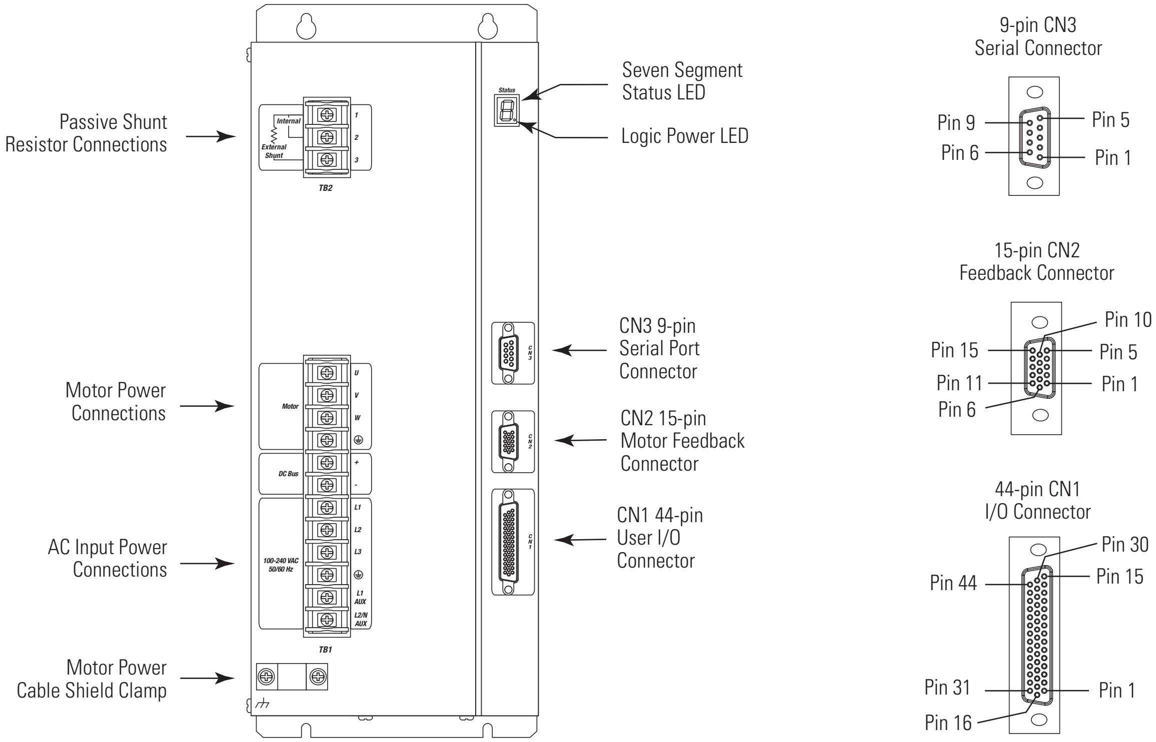
The following table provides the signal descriptions and pin-outs for the CN2 motor encoder (15-pin) connector.

CN2 Pin	Description	Signal
1	Channel A+/Sine Differential Input+	AM+
2	Channel A-/Sine Differential Input-	AM-
3	Channel B+/Cosine Differential Input+	BM+
4	Channel B-/Cosine Differential Input-	BM-
5	Channel I+/Index Pulse+	IM+
6	Common	ECOM
7	Encoder Power (+9V)	EPWR_9V
8	Commutation Channel S3	S3

CN2 Pin	Description	Signal
9	Positive Overtravel Limit	+LIMIT
10	Channel I-/Index Pulse-	IM-
11	Thermostat	TS
12	Commutation Channel S1	S1
13	Commutation Channel S2	S2
14	Encoder Power (+5V)	EPWR_5V
15	Negative Overtravel Limit	-LIMIT

Use the figure below to locate the front panel connections on the Ultra3000 230V (7.5 and 15 kW).

Figure 2.3
Ultra3000 Front Panel Connections for 2098-DSD-075, -075X, -150, and -150X



Serial Port Connector

The following table provides the signal descriptions and pin-outs for the CN3 serial port (9-pin) connector.

CN3 Pin	Description	Signal
1	RS-422/RS-485 Input+	RCV+
2	RS-232 Input	RCV
3	RS-232 Output	XMT
4	RS-422/RS-485 Output+	XMT+
5	Common	COM
6	Reserved	–
7	RS-422/RS-485 Input-	RCV-
8	RS-422/RS-485 Output-	XMT-
9	Reserved	–

I/O Connector

The following table provides the signal descriptions and pin-outs for the CN1 I/O (44-pin) connector.

CN1 Pin	Description	Signal
1	Auxiliary Encoder Power Out (+5V)	EPWR
2	Common	ECOM
3	Reserved	–
4	Auxiliary A+/Step+/CW+	AX+
5	Auxiliary A-/Step-/CW-	AX-
6	Auxiliary B+/Dir+/CCW+	BX+
7	Auxiliary B-/Dir-/CCW-	BX-
8	Auxiliary Encoder Ch I+	IX+
9	Auxiliary Encoder Ch I-	IX-
10	Unbuffered Motor Encoder Ch A+	AM+
11	Unbuffered Motor Encoder Ch A-	AM-
12	Unbuffered Motor Encoder Ch B+	BM+
13	Unbuffered Motor Encoder Ch B-	BM-
14	Unbuffered Motor Encoder Ch I+	IM+
15	Unbuffered Motor Encoder Ch I-	IM-
16	Buffered Motor Encoder Ch A+	AMOUT+
17	Buffered Motor Encoder Ch A-	AMOUT-
18	Buffered Motor Encoder Ch B+	BMOUT+
19	Buffered Motor Encoder Ch B-	BMOUT-
20	Buffered Motor Encoder Ch I+	IMOUT+
21	Buffered Motor Encoder Ch I-	IMOUT-
22	Common	ACOM

CN1 Pin	Description	Signal
23	Programmable Analog Output	AOUT
24	Analog Current Limit Input	ILIMIT
25	Command +	COMMAND+
26	Command -	COMMAND-
27	I/O Common	IOCOM
28	I/O Common	IOCOM
29	I/O Power	IOPWR
30	I/O Power	IOPWR
31	Digital Input 1	INPUT1
32	Digital Input 2	INPUT2
33	Digital Input 3	INPUT3
34	Digital Input 4	INPUT4
35	Digital Input 5	INPUT5
36	Digital Input 6	INPUT6
37	Digital Input 7	INPUT7
38	Digital Input 8	INPUT8
39	Digital Output 1	OUTPUT1
40	Digital Output 2	OUTPUT2
41	Digital Output 3	OUTPUT3
42	Digital Output 4	OUTPUT4
43	Normally Open Relay Output+	RELAY+
44	Normally Open Relay Output-	RELAY-

Motor Encoder Connector

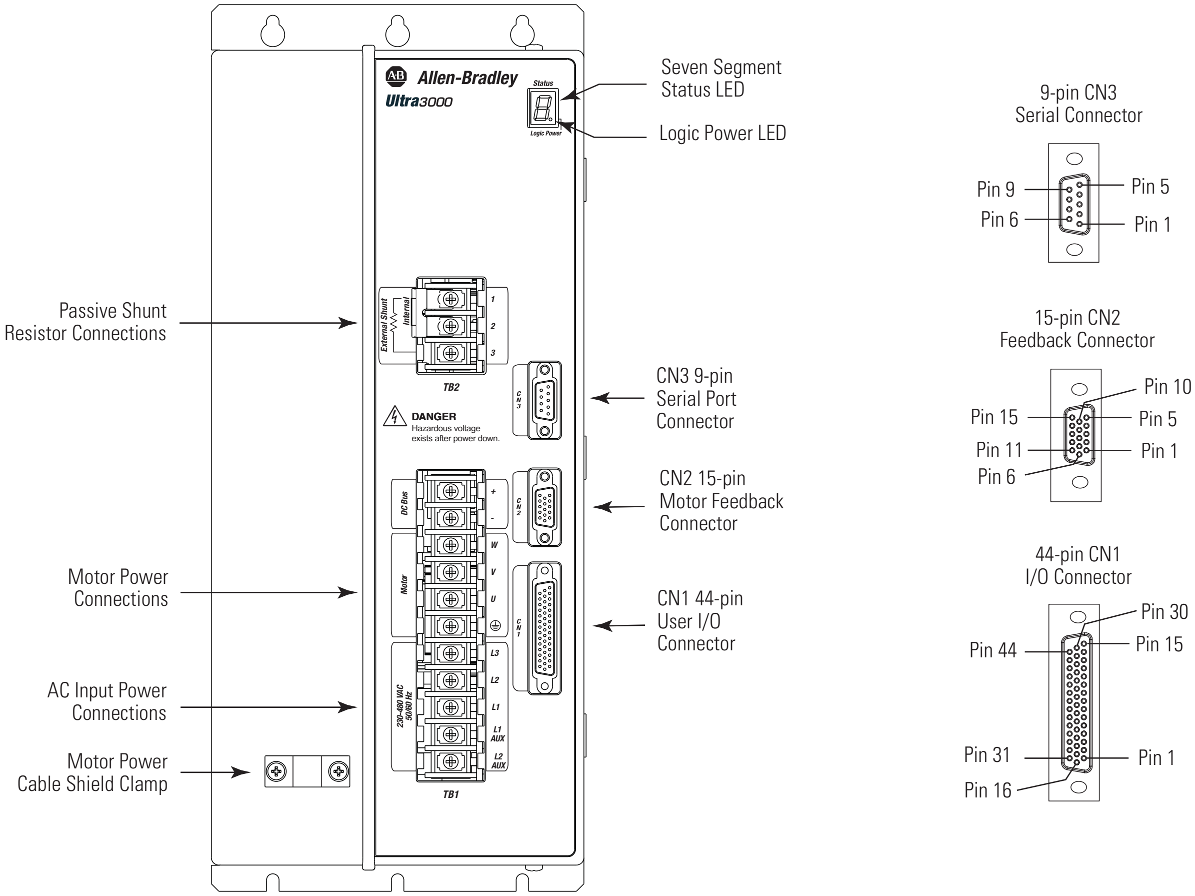
The following table provides the signal descriptions and pin-outs for the CN2 motor encoder (15-pin) connector.

CN2 Pin	Description	Signal
1	Channel A+/Sine Differential Input+	AM+
2	Channel A-/Sine Differential Input-	AM-
3	Channel B+/Cosine Differential Input+	BM+
4	Channel B-/Cosine Differential Input-	BM-
5	Channel I+/Index Pulse+	IM+
6	Common	ECOM
7	Encoder Power (+9V)	EPWR_9V
8	Commutation Channel S3	S3

CN2 Pin	Description	Signal
9	Positive Overtravel Limit	+LIMIT
10	Channel I-/Index Pulse-	IM-
11	Thermostat	TS
12	Commutation Channel S1	S1
13	Commutation Channel S2	S2
14	Encoder Power (+5V)	EPWR_5V
15	Negative Overtravel Limit	-LIMIT

Use the figure below to locate the front panel connections on the Ultra3000 460V drives (3W, 5 kW, 10 kW, 15 kW, and 22 kW).

Figure 2.4
Ultra3000 Front Panel Connections for 2098-DSD-HV*xxx* and HV*xxx*X



Serial Port Connector

The following table provides the signal descriptions and pin-outs for the CN3 serial port (9-pin) connector.

CN3 Pin	Description	Signal
1	RS-422/RS-485 Input+	RCV+
2	RS-232 Input	RCV
3	RS-232 Output	XMT
4	RS-422/RS-485 Output+	XMT+
5	Common	COM
6	Reserved	–
7	RS-422/RS-485 Input-	RCV-
8	RS-422/RS-485 Output-	XMT-
9	Reserved	–

I/O Connector

The following table provides the signal descriptions and pin-outs for the CN1 I/O (44-pin) connector.

CN1 Pin	Description	Signal
1	Auxiliary Encoder Power Out (+5V)	EPWR
2	Common	ECOM
3	Reserved	–
4	Auxiliary A+/Step+/CW+	AX+
5	Auxiliary A-/Step-/CW-	AX-
6	Auxiliary B+/Dir+/CCW+	BX+
7	Auxiliary B-/Dir-/CCW-	BX-
8	Auxiliary Encoder Ch I+	IX+
9	Auxiliary Encoder Ch I-	IX-
10	Unbuffered Motor Encoder Ch A+	AM+
11	Unbuffered Motor Encoder Ch A-	AM-
12	Unbuffered Motor Encoder Ch B+	BM+
13	Unbuffered Motor Encoder Ch B-	BM-
14	Unbuffered Motor Encoder Ch I+	IM+
15	Unbuffered Motor Encoder Ch I-	IM-
16	Buffered Motor Encoder Ch A+	AMOUT+
17	Buffered Motor Encoder Ch A-	AMOUT-
18	Buffered Motor Encoder Ch B+	BMOUT+
19	Buffered Motor Encoder Ch B-	BMOUT-
20	Buffered Motor Encoder Ch I+	IMOUT+
21	Buffered Motor Encoder Ch I-	IMOUT-
22	Common	ACOM

CN1 Pin	Description	Signal
23	Programmable Analog Output	AOUT
24	Analog Current Limit Input	ILIMIT
25	Command +	COMMAND+
26	Command -	COMMAND-
27	I/O Common	IOCOM
28	I/O Common	IOCOM
29	I/O Power	IOPWR
30	I/O Power	IOPWR
31	Digital Input 1	INPUT1
32	Digital Input 2	INPUT2
33	Digital Input 3	INPUT3
34	Digital Input 4	INPUT4
35	Digital Input 5	INPUT5
36	Digital Input 6	INPUT6
37	Digital Input 7	INPUT7
38	Digital Input 8	INPUT8
39	Digital Output 1	OUTPUT1
40	Digital Output 2	OUTPUT2
41	Digital Output 3	OUTPUT3
42	Digital Output 4	OUTPUT4
43	Normally Open Relay Output+	RELAY+
44	Normally Open Relay Output-	RELAY-

Motor Encoder Connector

The following table provides the signal descriptions and pin-outs for the CN2 motor encoder (15-pin) connector.

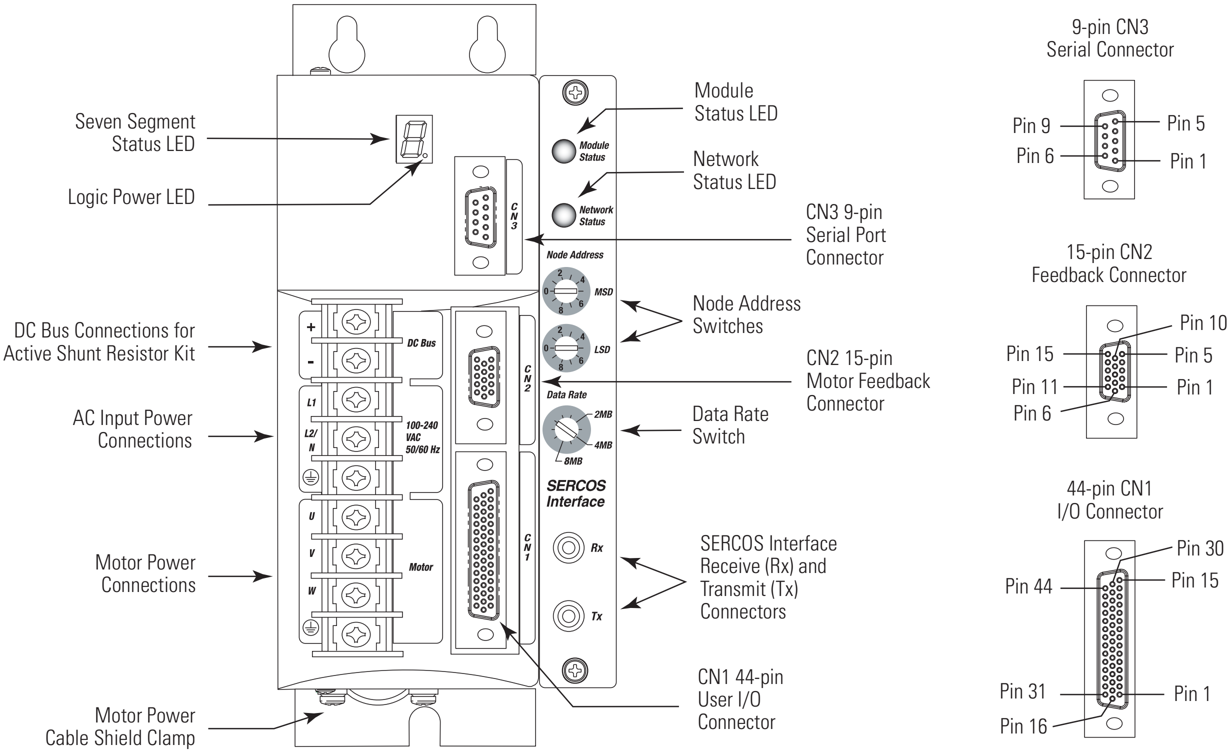
CN2 Pin	Description	Signal
1	Channel A+/Sine Differential Input+	AM+
2	Channel A-/Sine Differential Input-	AM-
3	Channel B+/Cosine Differential Input+	BM+
4	Channel B-/Cosine Differential Input-	BM-
5	Channel I+/Index Pulse+	IM+
6	Common	ECOM
7	Encoder Power (+9V)	EPWR_9V
8	Commutation Channel S3	S3

CN2 Pin	Description	Signal
9	Positive Overtravel Limit	+LIMIT
10	Channel I-/Index Pulse-	IM-
11	Thermostat	TS
12	Commutation Channel S1	S1
13	Commutation Channel S2	S2
14	Encoder Power (+5V)	EPWR_5V
15	Negative Overtravel Limit	-LIMIT

Ultra3000 (with SERCOS) Front Panel Connections

Use the figure below to locate the front panel connections on the Ultra3000 with SERCOS interface 230V drives (500W, 1 kW, and 2 kW).

Figure 2.5
Ultra3000 Front Panel Connections for 2098-DSD-005-SE, -010-SE, and -020-SE



Serial Port Connector

The following table provides the signal descriptions and pin-outs for the CN3 serial port (9-pin) connector.

CN3 Pin	Description	Signal
1	RS-422/RS-485 Input+	RCV+
2	RS-232 Input	RCV
3	RS-232 Output	XMT
4	RS-422/RS-485 Output+	XMT+
5	Common	COM
6	Reserved	–
7	RS-422/RS-485 Input-	RCV-
8	RS-422/RS-485 Output-	XMT-
9	Reserved	–

I/O Connector

The following table provides the signal descriptions and pin-outs for the CN1 I/O (44-pin) connector.

CN1 Pin	Description	Signal
1	Auxiliary Encoder Power Out (+5V)	EPWR
2	Common	ECOM
3	Auxiliary Logic Power In (+5V)	AUXPWR
4	Auxiliary Encoder Ch A+	AX+
5	Auxiliary Encoder Ch A-	AX-
6	Auxiliary Encoder Ch B+	BX+
7	Auxiliary Encoder Ch B-	BX-
8	Auxiliary Encoder Ch I+	IX+
9	Auxiliary Encoder Ch I-	IX-
10	Unbuffered Motor Encoder Ch A+	AM+
11	Unbuffered Motor Encoder Ch A-	AM-
12	Unbuffered Motor Encoder Ch B+	BM+
13	Unbuffered Motor Encoder Ch B-	BM-
14	Unbuffered Motor Encoder Ch I+	IM+
15	Unbuffered Motor Encoder Ch I-	IM-
16	Buffered Motor Encoder Ch A+	AMOUT+
17	Buffered Motor Encoder Ch A-	AMOUT-
18	Buffered Motor Encoder Ch B+	BMOUT+
19	Buffered Motor Encoder Ch B-	BMOUT-
20	Buffered Motor Encoder Ch I+	IMOUT+
21	Buffered Motor Encoder Ch I-	IMOUT-
22	Common	ACOM

CN1 Pin	Description	Signal
23	Reserved	–
24	Analog Current Limit Input	ILIMIT
25	Reserved	–
26	Reserved	–
27	I/O Common	IOCOM
28	I/O Common	IOCOM
29	I/O Power	IOPWR
30	I/O Power	IOPWR
31	Drive Enable Input	ENABLE
32	Home Sensor Input	HOME
33	Registration Sensor 1 Input	REG1
34	Registration Sensor 2 Input	REG2
35	Reserved	–
36	Reserved	–
37	Positive Overtravel Input	OT_POS
38	Negative Overtravel Input	OT_NEG
39	Drive Ready	READY [1]
40	Reserved	–
41	Reserved	–
42	Reserved	–
43	Brake Relay Output+	BRAKE+
44	Brake Relay Output-	BRAKE-

[1] READY signal only available with firmware version 1.29 (or above). Requires use of drive-mounted breakout board (2090-U3CBB-DM44).

Motor Encoder Connector

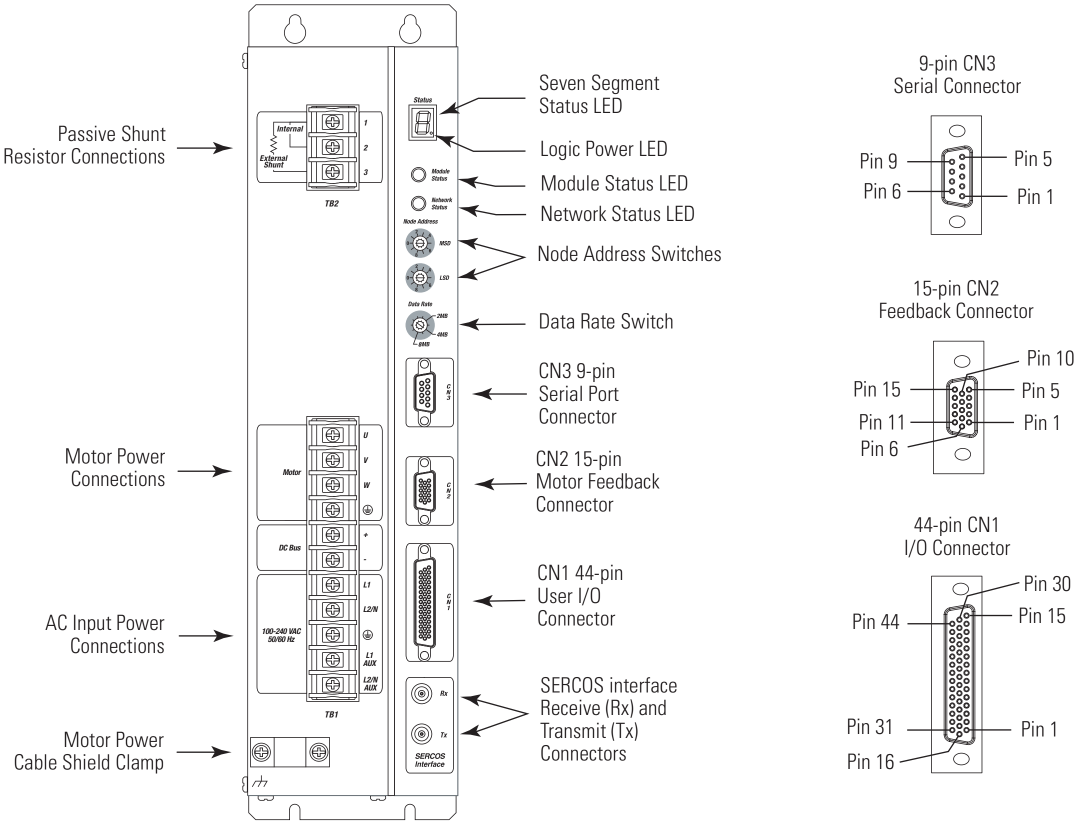
The following table provides the signal descriptions and pin-outs for the CN2 motor encoder (15-pin) connector.

CN2 Pin	Description	Signal
1	Channel A+/Sine Differential Input+	AM+
2	Channel A-/Sine Differential Input-	AM-
3	Channel B+/Cosine Differential Input+	BM+
4	Channel B-/Cosine Differential Input-	BM-
5	Channel I+/Index Pulse+	IM+
6	Common	ECOM
7	Reserved	–
8	Commutation Channel S3	S3

CN2 Pin	Description	Signal
9	Positive Overtravel Limit	+LIMIT
10	Channel I-/Index Pulse-	IM-
11	Thermostat	TS
12	Commutation Channel S1	S1
13	Commutation Channel S2	S2
14	Encoder Power (+5V)	EPWR_5V
15	Negative Overtravel Limit	-LIMIT

Use the figure below to locate the front panel connections on the Ultra3000 with SERCOS interface 230V drive (3 kW).

Figure 2.6
Ultra3000 Front Panel Connections for 2098-DSD-030-SE



Serial Port Connector

The following table provides the signal descriptions and pin-outs for the CN3 serial port (9-pin) connector.

CN3 Pin	Description	Signal
1	RS-422/RS-485 Input+	RCV+
2	RS-232 Input	RCV
3	RS-232 Output	XMT
4	RS-422/RS-485 Output+	XMT+
5	Common	COM
6	Reserved	–
7	RS-422/RS-485 Input-	RCV-
8	RS-422/RS-485 Output-	XMT-
9	Reserved	–

I/O Connector

The following table provides the signal descriptions and pin-outs for the CN1 I/O (44-pin) connector.

CN1 Pin	Description	Signal
1	Auxiliary Encoder Power Out (+5V)	EPWR
2	Common	ECOM
3	Reserved	–
4	Auxiliary Encoder Ch A+	AX+
5	Auxiliary Encoder Ch A-	AX-
6	Auxiliary Encoder Ch B+	BX+
7	Auxiliary Encoder Ch B-	BX-
8	Auxiliary Encoder Ch I+	IX+
9	Auxiliary Encoder Ch I-	IX-
10	Unbuffered Motor Encoder Ch A+	AM+
11	Unbuffered Motor Encoder Ch A-	AM-
12	Unbuffered Motor Encoder Ch B+	BM+
13	Unbuffered Motor Encoder Ch B-	BM-
14	Unbuffered Motor Encoder Ch I+	IM+
15	Unbuffered Motor Encoder Ch I-	IM-
16	Buffered Motor Encoder Ch A+	AMOUT+
17	Buffered Motor Encoder Ch A-	AMOUT-
18	Buffered Motor Encoder Ch B+	BMOUT+
19	Buffered Motor Encoder Ch B-	BMOUT-
20	Buffered Motor Encoder Ch I+	IMOUT+
21	Buffered Motor Encoder Ch I-	IMOUT-
22	Common	ACOM

CN1 Pin	Description	Signal
23	Reserved	–
24	Analog Current Limit Input	ILIMIT
25	Reserved	–
26	Reserved	–
27	I/O Common	IOCOM
28	I/O Common	IOCOM
29	I/O Power	IOPWR
30	I/O Power	IOPWR
31	Drive Enable Input	ENABLE
32	Home Sensor Input	HOME
33	Registration Sensor 1 Input	REG1
34	Registration Sensor 2 Input	REG2
35	Reserved	–
36	Reserved	–
37	Positive Overtravel Input	OT_POS
38	Negative Overtravel Input	OT_NEG
39	Drive Ready	READY [1]
40	Reserved	–
41	Reserved	–
42	Reserved	–
43	Brake Relay Output+	BRAKE+
44	Brake Relay Output-	BRAKE-

[1] READY signal only available with firmware version 1.29 (or above). Requires use of drive-mounted breakout board (2090-U3BB2-DM44).

Motor Encoder Connector

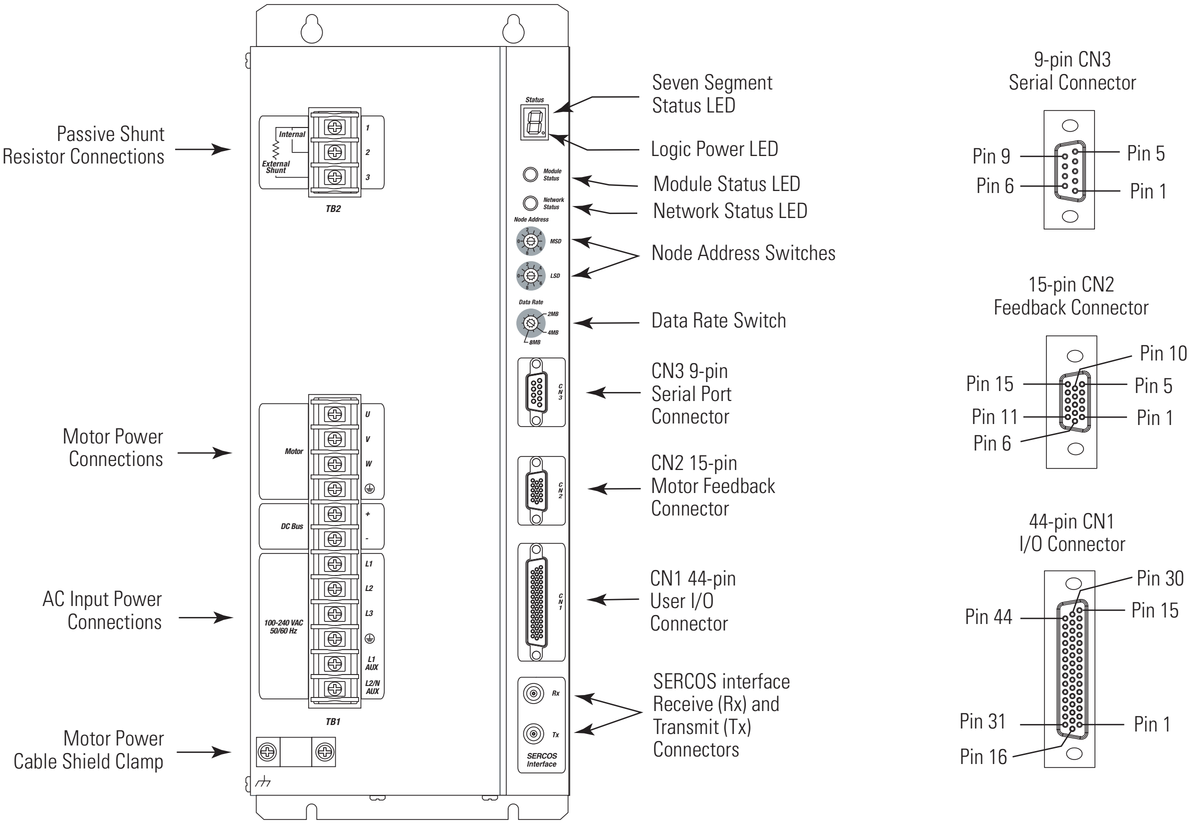
The following table provides the signal descriptions and pin-outs for the CN2 motor encoder (15-pin) connector.

CN2 Pin	Description	Signal
1	Channel A+/Sine Differential Input+	AM+
2	Channel A-/Sine Differential Input-	AM-
3	Channel B+/Cosine Differential Input+	BM+
4	Channel B-/Cosine Differential Input-	BM-
5	Channel I+/Index Pulse+	IM+
6	Common	ECOM
7	Encoder Power (+9V)	EPWR_9V
8	Commutation Channel S3	S3

CN2 Pin	Description	Signal
9	Positive Overtravel Limit	+LIMIT
10	Channel I-/Index Pulse-	IM-
11	Thermostat	TS
12	Commutation Channel S1	S1
13	Commutation Channel S2	S2
14	Encoder Power (+5V)	EPWR_5V
15	Negative Overtravel Limit	-LIMIT

Use the figure below to locate the front panel connections on the Ultra3000 with SERCOS interface 230V drives (7.5 and 15 kW).

Figure 2.7
Ultra3000 Front Panel Connections for 2098-DSD-075-SE and -150-SE



Serial Port Connector

The following table provides the signal descriptions and pin-outs for the CN3 serial port (9-pin) connector.

CN3 Pin	Description	Signal
1	RS-422/RS-485 Input+	RCV+
2	RS-232 Input	RCV
3	RS-232 Output	XMT
4	RS-422/RS-485 Output+	XMT+
5	Common	COM
6	Reserved	–
7	RS-422/RS-485 Input-	RCV-
8	RS-422/RS-485 Output-	XMT-
9	Reserved	–

I/O Connector

The following table provides the signal descriptions and pin-outs for the CN1 I/O (44-pin) connector.

CN1 Pin	Description	Signal
1	Auxiliary Encoder Power Out (+5V)	EPWR
2	Common	ECOM
3	Reserved	–
4	Auxiliary Encoder Ch A+	AX+
5	Auxiliary Encoder Ch A-	AX-
6	Auxiliary Encoder Ch B+	BX+
7	Auxiliary Encoder Ch B-	BX-
8	Auxiliary Encoder Ch I+	IX+
9	Auxiliary Encoder Ch I-	IX-
10	Unbuffered Motor Encoder Ch A+	AM+
11	Unbuffered Motor Encoder Ch A-	AM-
12	Unbuffered Motor Encoder Ch B+	BM+
13	Unbuffered Motor Encoder Ch B-	BM-
14	Unbuffered Motor Encoder Ch I+	IM+
15	Unbuffered Motor Encoder Ch I-	IM-
16	Buffered Motor Encoder Ch A+	AMOUT+
17	Buffered Motor Encoder Ch A-	AMOUT-
18	Buffered Motor Encoder Ch B+	BMOUT+
19	Buffered Motor Encoder Ch B-	BMOUT-
20	Buffered Motor Encoder Ch I+	IMOUT+
21	Buffered Motor Encoder Ch I-	IMOUT-
22	Common	ACOM

CN1 Pin	Description	Signal
23	Reserved	–
24	Analog Current Limit Input	ILIMIT
25	Reserved	–
26	Reserved	–
27	I/O Common	IOCOM
28	I/O Common	IOCOM
29	I/O Power	IOPWR
30	I/O Power	IOPWR
31	Drive Enable Input	ENABLE
32	Home Sensor Input	HOME
33	Registration Sensor 1 Input	REG1
34	Registration Sensor 2 Input	REG2
35	Reserved	–
36	Reserved	–
37	Positive Overtravel Input	OT_POS
38	Negative Overtravel Input	OT_NEG
39	Drive Ready	READY [1]
40	Reserved	–
41	Reserved	–
42	Reserved	–
43	Brake Relay Output+	BRAKE+
44	Brake Relay Output-	BRAKE-

[1] READY signal only available with firmware version 1.29 (or above). Requires use of drive-mounted breakout board (2090-U3BB2-DM44).

Motor Encoder Connector

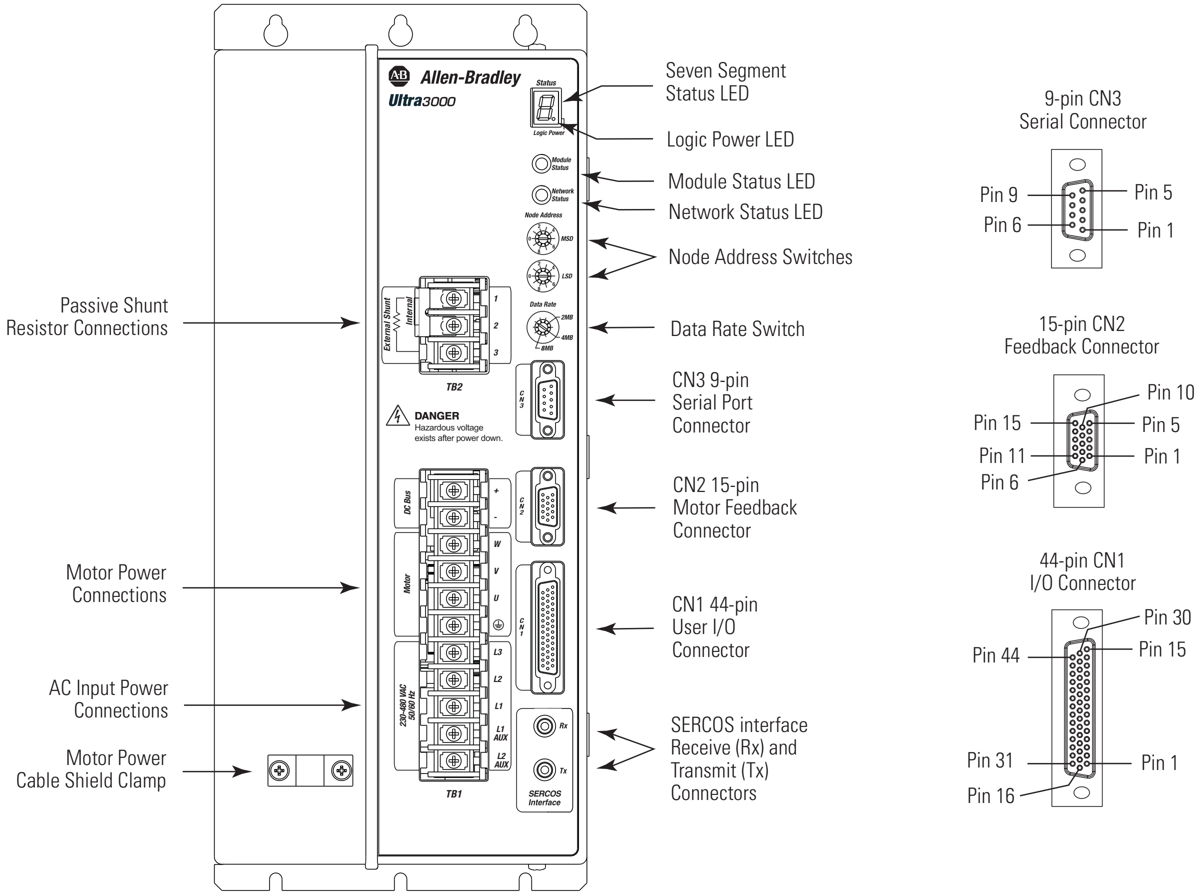
The following table provides the signal descriptions and pin-outs for the CN2 motor encoder (15-pin) connector.

CN2 Pin	Description	Signal
1	Channel A+/Sine Differential Input+	AM+
2	Channel A-/Sine Differential Input-	AM-
3	Channel B+/Cosine Differential Input+	BM+
4	Channel B-/Cosine Differential Input-	BM-
5	Channel I+/Index Pulse+	IM+
6	Common	ECOM
7	Encoder Power (+9V)	EPWR_9V
8	Commutation Channel S3	S3

CN2 Pin	Description	Signal
9	Positive Overtravel Limit	+LIMIT
10	Channel I-/Index Pulse-	IM-
11	Thermostat	TS
12	Commutation Channel S1	S1
13	Commutation Channel S2	S2
14	Encoder Power (+5V)	EPWR_5V
15	Negative Overtravel Limit	-LIMIT

Use the figure below to locate the front panel connections on the Ultra3000 with SERCOS interface 460V drives (3 kW, 5 kW, 10 kW, 15 kW, and 22 kW).

Figure 2.8
Ultra3000 Front Panel Connections for 2098-DSD-HV*xxx*-SE



Serial Port Connector

The following table provides the signal descriptions and pin-outs for the CN3 serial port (9-pin) connector.

CN3 Pin	Description	Signal
1	RS-422/RS-485 Input+	RCV+
2	RS-232 Input	RCV
3	RS-232 Output	XMT
4	RS-422/RS-485 Output+	XMT+
5	Common	COM
6	Reserved	–
7	RS-422/RS-485 Input-	RCV-
8	RS-422/RS-485 Output-	XMT-
9	Reserved	–

I/O Connector

The following table provides the signal descriptions and pin-outs for the CN1 I/O (44-pin) connector.

CN1 Pin	Description	Signal
1	Auxiliary Encoder Power Out (+5V)	EPWR
2	Common	ECOM
3	Reserved	–
4	Auxiliary Encoder Ch A+	AX+
5	Auxiliary Encoder Ch A-	AX-
6	Auxiliary Encoder Ch B+	BX+
7	Auxiliary Encoder Ch B-	BX-
8	Auxiliary Encoder Ch I+	IX+
9	Auxiliary Encoder Ch I-	IX-
10	Unbuffered Motor Encoder Ch A+	AM+
11	Unbuffered Motor Encoder Ch A-	AM-
12	Unbuffered Motor Encoder Ch B+	BM+
13	Unbuffered Motor Encoder Ch B-	BM-
14	Unbuffered Motor Encoder Ch I+	IM+
15	Unbuffered Motor Encoder Ch I-	IM-
16	Buffered Motor Encoder Ch A+	AMOUT+
17	Buffered Motor Encoder Ch A-	AMOUT-
18	Buffered Motor Encoder Ch B+	BMOUT+
19	Buffered Motor Encoder Ch B-	BMOUT-
20	Buffered Motor Encoder Ch I+	IMOUT+
21	Buffered Motor Encoder Ch I-	IMOUT-
22	Common	ACOM

CN1 Pin	Description	Signal
23	Reserved	–
24	Analog Current Limit Input	ILIMIT
25	Reserved	–
26	Reserved	–
27	I/O Common	IOCOM
28	I/O Common	IOCOM
29	I/O Power	IOPWR
30	I/O Power	IOPWR
31	Drive Enable Input	ENABLE
32	Home Sensor Input	HOME
33	Registration Sensor 1 Input	REG1
34	Registration Sensor 2 Input	REG2
35	Reserved	–
36	Reserved	–
37	Positive Overtravel Input	OT_POS
38	Negative Overtravel Input	OT_NEG
39	Drive Ready	READY [1]
40	Reserved	–
41	Reserved	–
42	Reserved	–
43	Brake Relay Output+	BRAKE+
44	Brake Relay Output-	BRAKE-

[1] READY signal only available with firmware version 1.29 (or above). Requires use of drive-mounted breakout board (2090-U3BB2-DM44).

Motor Encoder Connector

The following table provides the signal descriptions and pin-outs for the CN2 motor encoder (15-pin) connector.

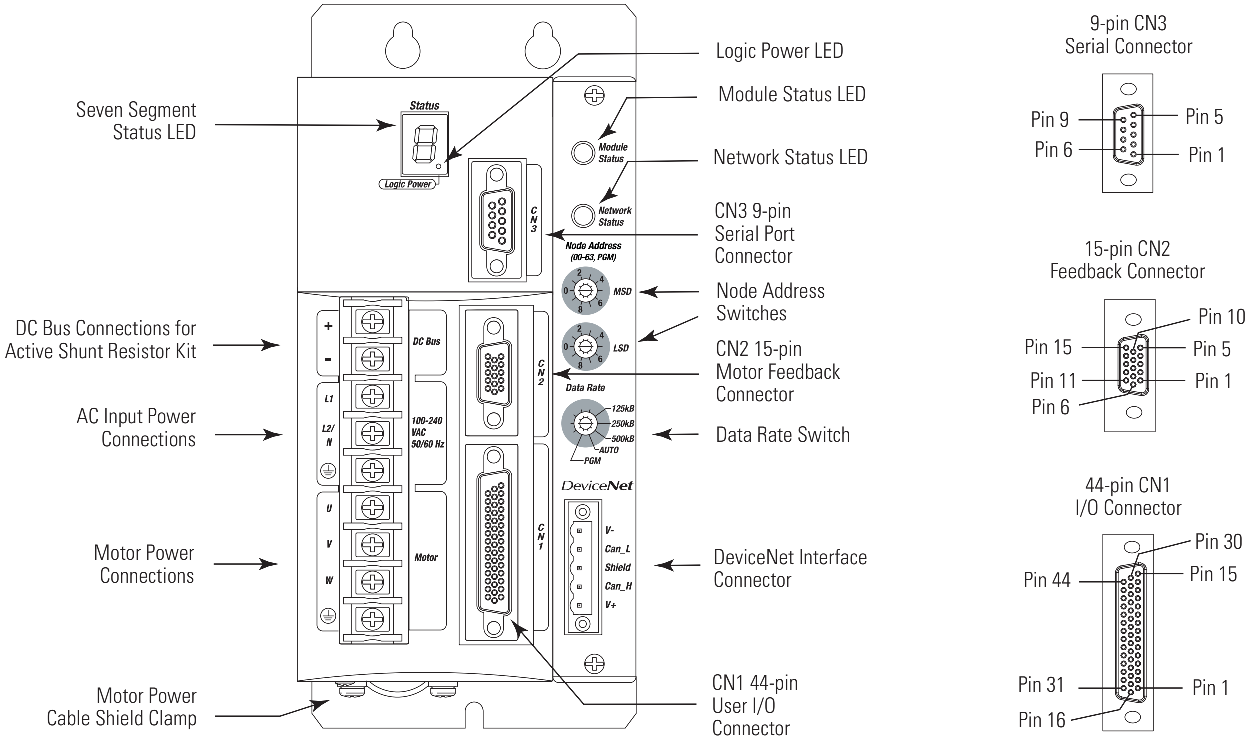
CN2 Pin	Description	Signal
1	Channel A+/Sine Differential Input+	AM+
2	Channel A-/Sine Differential Input-	AM-
3	Channel B+/Cosine Differential Input+	BM+
4	Channel B-/Cosine Differential Input-	BM-
5	Channel I+/Index Pulse+	IM+
6	Common	ECOM
7	Encoder Power (+9V)	EPWR_9V
8	Commutation Channel S3	S3

CN2 Pin	Description	Signal
9	Positive Overtravel Limit	+LIMIT
10	Channel I-/Index Pulse-	IM-
11	Thermostat	TS
12	Commutation Channel S1	S1
13	Commutation Channel S2	S2
14	Encoder Power (+5V)	EPWR_5V
15	Negative Overtravel Limit	-LIMIT

Ultra3000 (with DeviceNet) Front Panel Connections

Use the figure below to locate the front panel connections on the Ultra3000 with DeviceNet Interface 230V drives (500W, 1 kW, and 2 kW).

Figure 2.9
Ultra3000 Front Panel Connections
for 2098-DSD-005-DN, -005X-DN, -010-DN, -010X-DN, -020-DN, and -020X-DN



Serial Port Connector

The following table provides the signal descriptions and pin-outs for the CN3 serial port (9-pin) connector.

CN3 Pin	Description	Signal
1	RS-422/RS-485 Input+	RCV+
2	RS-232 Input	RCV
3	RS-232 Output	XMT
4	RS-422/RS-485 Output+	XMT+
5	Common	COM
6	Reserved	–
7	RS-422/RS-485 Input-	RCV-
8	RS-422/RS-485 Output-	XMT-
9	Reserved	–

I/O Connector

The following table provides the signal descriptions and pin-outs for the CN1 I/O (44-pin) connector.

CN1 Pin	Description	Signal
1	Auxiliary Encoder Power Out (+5V)	EPWR
2	Common	ECOM
3	Auxiliary Logic Power In (+5V)	AUXPWR
4	Auxiliary A+/Step+/CW+	AX+
5	Auxiliary A-/Step-/CW-	AX-
6	Auxiliary B+/Dir+/CCW+	BX+
7	Auxiliary B-/Dir-/CCW-	BX-
8	Auxiliary Encoder Ch I+	IX+
9	Auxiliary Encoder Ch I-	IX-
10	Unbuffered Motor Encoder Ch A+	AM+
11	Unbuffered Motor Encoder Ch A-	AM-
12	Unbuffered Motor Encoder Ch B+	BM+
13	Unbuffered Motor Encoder Ch B-	BM-
14	Unbuffered Motor Encoder Ch I+	IM+
15	Unbuffered Motor Encoder Ch I-	IM-
16	Buffered Motor Encoder Ch A+	AMOUT+
17	Buffered Motor Encoder Ch A-	AMOUT-
18	Buffered Motor Encoder Ch B+	BMOUT+
19	Buffered Motor Encoder Ch B-	BMOUT-
20	Buffered Motor Encoder Ch I+	IMOUT+
21	Buffered Motor Encoder Ch I-	IMOUT-
22	Common	ACOM

CN1 Pin	Description	Signal
23	Programmable Analog Output	AOUT
24	Analog Current Limit Input	ILIMIT
25	Command +	COMMAND+
26	Command -	COMMAND-
27	I/O Common	IOCOM
28	I/O Common	IOCOM
29	I/O Power	IOPWR
30	I/O Power	IOPWR
31	Digital Input 1	INPUT1
32	Digital Input 2	INPUT2
33	Digital Input 3	INPUT3
34	Digital Input 4	INPUT4
35	Digital Input 5	INPUT5
36	Digital Input 6	INPUT6
37	Digital Input 7	INPUT7
38	Digital Input 8	INPUT8
39	Digital Output 1	OUTPUT1
40	Digital Output 2	OUTPUT2
41	Digital Output 3	OUTPUT3
42	Digital Output 4	OUTPUT4
43	Normally Open Relay Output+	RELAY+
44	Normally Open Relay Output-	RELAY-

Motor Encoder Connector

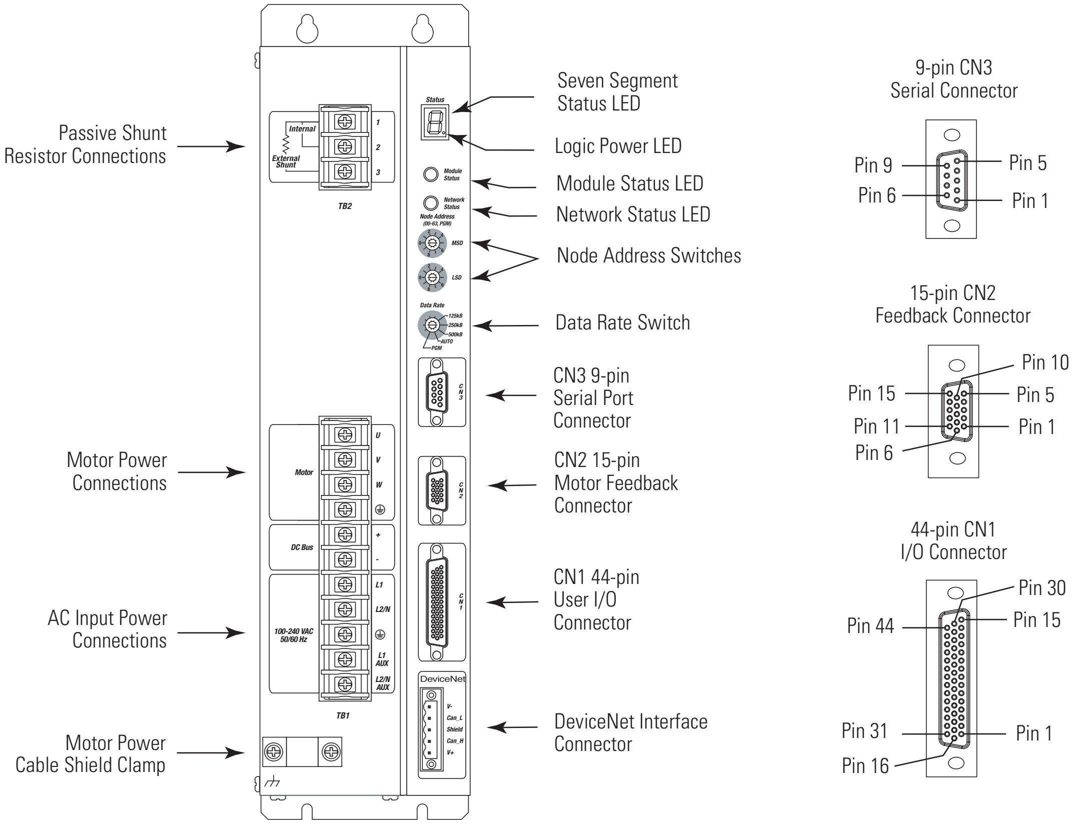
The following table provides the signal descriptions and pin-outs for the CN2 motor encoder (15-pin) connector.

CN2 Pin	Description	Signal
1	Channel A+/Sine Differential Input+	AM+
2	Channel A-/Sine Differential Input-	AM-
3	Channel B+/Cosine Differential Input+	BM+
4	Channel B-/Cosine Differential Input-	BM-
5	Channel I+/Index Pulse+	IM+
6	Common	ECOM
7	Reserved	–
8	Commutation Channel S3	S3

CN2 Pin	Description	Signal
9	Positive Overtravel Limit	+LIMIT
10	Channel I-/Index Pulse-	IM-
11	Thermostat	TS
12	Commutation Channel S1	S1
13	Commutation Channel S2	S2
14	Encoder Power (+5V)	EPWR_5V
15	Negative Overtravel Limit	-LIMIT

Use the figure below to locate the front panel connections on the Ultra3000 with DeviceNet Interface 230V drives (3 kW).

Figure 2.10
Ultra3000 Front Panel Connections for 2098-DSD-030-DN and -030X-DN



Serial Port Connector

The following table provides the signal descriptions and pin-outs for the CN3 serial port (9-pin) connector.

CN3 Pin	Description	Signal
1	RS-422/RS-485 Input+	RCV+
2	RS-232 Input	RCV
3	RS-232 Output	XMT
4	RS-422/RS-485 Output+	XMT+
5	Common	COM
6	Reserved	–
7	RS-422/RS-485 Input-	RCV-
8	RS-422/RS-485 Output-	XMT-
9	Reserved	–

I/O Connector

The following table provides the signal descriptions and pin-outs for the CN1 I/O (44-pin) connector.

CN1 Pin	Description	Signal
1	Auxiliary Encoder Power Out (+5V)	EPWR
2	Common	ECOM
3	Reserved	–
4	Auxiliary A+/Step+/CW+	AX+
5	Auxiliary A-/Step-/CW-	AX-
6	Auxiliary B+/Dir+/CCW+	BX+
7	Auxiliary B-/Dir-/CCW-	BX-
8	Auxiliary Encoder Ch I+	IX+
9	Auxiliary Encoder Ch I-	IX-
10	Unbuffered Motor Encoder Ch A+	AM+
11	Unbuffered Motor Encoder Ch A-	AM-
12	Unbuffered Motor Encoder Ch B+	BM+
13	Unbuffered Motor Encoder Ch B-	BM-
14	Unbuffered Motor Encoder Ch I+	IM+
15	Unbuffered Motor Encoder Ch I-	IM-
16	Buffered Motor Encoder Ch A+	AMOUT+
17	Buffered Motor Encoder Ch A-	AMOUT-
18	Buffered Motor Encoder Ch B+	BMOUT+
19	Buffered Motor Encoder Ch B-	BMOUT-
20	Buffered Motor Encoder Ch I+	IMOUT+
21	Buffered Motor Encoder Ch I-	IMOUT-
22	Common	ACOM

CN1 Pin	Description	Signal
23	Programmable Analog Output	AOUT
24	Analog Current Limit Input	ILIMIT
25	Command +	COMMAND+
26	Command -	COMMAND-
27	I/O Common	IOCOM
28	I/O Common	IOCOM
29	I/O Power	IOPWR
30	I/O Power	IOPWR
31	Digital Input 1	INPUT1
32	Digital Input 2	INPUT2
33	Digital Input 3	INPUT3
34	Digital Input 4	INPUT4
35	Digital Input 5	INPUT5
36	Digital Input 6	INPUT6
37	Digital Input 7	INPUT7
38	Digital Input 8	INPUT8
39	Digital Output 1	OUTPUT1
40	Digital Output 2	OUTPUT2
41	Digital Output 3	OUTPUT3
42	Digital Output 4	OUTPUT4
43	Normally Open Relay Output+	RELAY+
44	Normally Open Relay Output-	RELAY-

Motor Encoder Connector

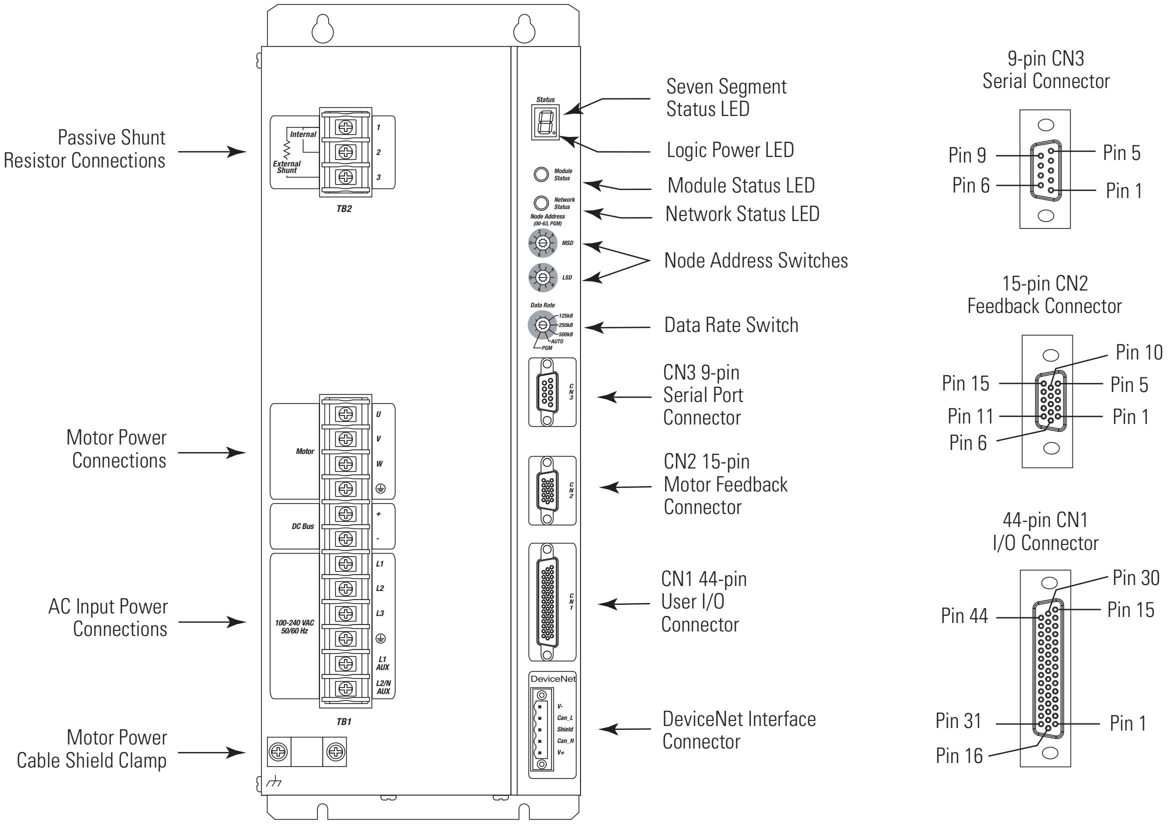
The following table provides the signal descriptions and pin-outs for the CN2 motor encoder (15-pin) connector.

CN2 Pin	Description	Signal
1	Channel A+/Sine Differential Input+	AM+
2	Channel A-/Sine Differential Input-	AM-
3	Channel B+/Cosine Differential Input+	BM+
4	Channel B-/Cosine Differential Input-	BM-
5	Channel I+/Index Pulse+	IM+
6	Common	ECOM
7	Encoder Power (+9V)	EPWR_9V
8	Commutation Channel S3	S3

CN2 Pin	Description	Signal
9	Positive Overtravel Limit	+LIMIT
10	Channel I-/Index Pulse-	IM-
11	Thermostat	TS
12	Commutation Channel S1	S1
13	Commutation Channel S2	S2
14	Encoder Power (+5V)	EPWR_5V
15	Negative Overtravel Limit	-LIMIT

Use the figure below to locate the front panel connections on the Ultra3000 with DeviceNet Interface 230V drives (7.5 and 15 kW).

Figure 2.11
Ultra3000 Front Panel Connections
for 2098-DSD-075-DN, -075X-DN, -150-DN, and -150X-DN



Serial Port Connector

The following table provides the signal descriptions and pin-outs for the CN3 serial port (9-pin) connector.

CN3 Pin	Description	Signal
1	RS-422/RS-485 Input+	RCV+
2	RS-232 Input	RCV
3	RS-232 Output	XMT
4	RS-422/RS-485 Output+	XMT+
5	Common	COM
6	Reserved	–
7	RS-422/RS-485 Input-	RCV-
8	RS-422/RS-485 Output-	XMT-
9	Reserved	–

I/O Connector

The following table provides the signal descriptions and pin-outs for the CN1 I/O (44-pin) connector.

CN1 Pin	Description	Signal
1	Auxiliary Encoder Power Out (+5V)	EPWR
2	Common	ECOM
3	Reserved	–
4	Auxiliary A+/Step+/CW+	AX+
5	Auxiliary A-/Step-/CW-	AX-
6	Auxiliary B+/Dir+/CCW+	BX+
7	Auxiliary B-/Dir-/CCW-	BX-
8	Auxiliary Encoder Ch I+	IX+
9	Auxiliary Encoder Ch I-	IX-
10	Unbuffered Motor Encoder Ch A+	AM+
11	Unbuffered Motor Encoder Ch A-	AM-
12	Unbuffered Motor Encoder Ch B+	BM+
13	Unbuffered Motor Encoder Ch B-	BM-
14	Unbuffered Motor Encoder Ch I+	IM+
15	Unbuffered Motor Encoder Ch I-	IM-
16	Buffered Motor Encoder Ch A+	AMOUT+
17	Buffered Motor Encoder Ch A-	AMOUT-
18	Buffered Motor Encoder Ch B+	BMOUT+
19	Buffered Motor Encoder Ch B-	BMOUT-
20	Buffered Motor Encoder Ch I+	IMOUT+
21	Buffered Motor Encoder Ch I-	IMOUT-
22	Common	ACOM

CN1 Pin	Description	Signal
23	Programmable Analog Output	AOUT
24	Analog Current Limit Input	ILIMIT
25	Command +	COMMAND+
26	Command -	COMMAND-
27	I/O Common	IOCOM
28	I/O Common	IOCOM
29	I/O Power	IOPWR
30	I/O Power	IOPWR
31	Digital Input 1	INPUT1
32	Digital Input 2	INPUT2
33	Digital Input 3	INPUT3
34	Digital Input 4	INPUT4
35	Digital Input 5	INPUT5
36	Digital Input 6	INPUT6
37	Digital Input 7	INPUT7
38	Digital Input 8	INPUT8
39	Digital Output 1	OUTPUT1
40	Digital Output 2	OUTPUT2
41	Digital Output 3	OUTPUT3
42	Digital Output 4	OUTPUT4
43	Normally Open Relay Output+	RELAY+
44	Normally Open Relay Output-	RELAY-

Motor Encoder Connector

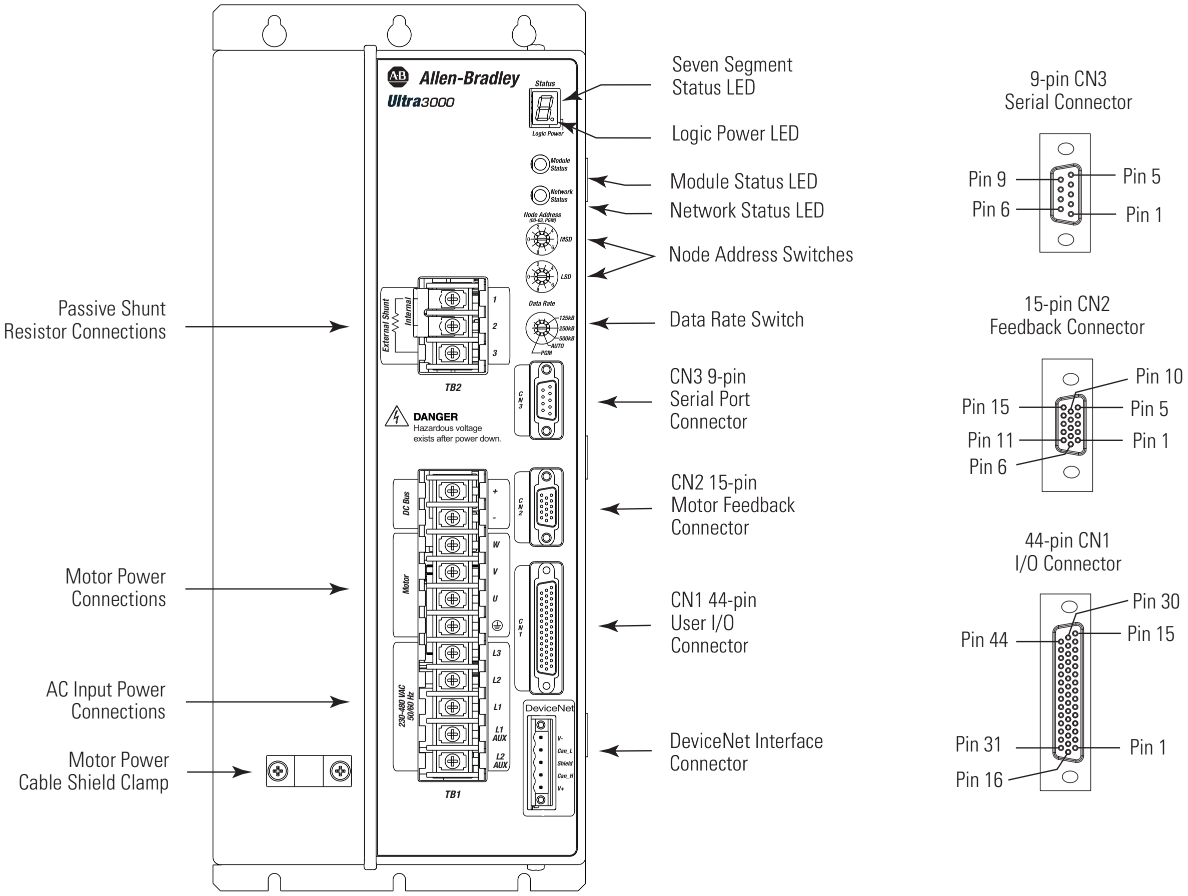
The following table provides the signal descriptions and pin-outs for the CN2 motor encoder (15-pin) connector.

CN2 Pin	Description	Signal
1	Channel A+/Sine Differential Input+	AM+
2	Channel A-/Sine Differential Input-	AM-
3	Channel B+/Cosine Differential Input+	BM+
4	Channel B-/Cosine Differential Input-	BM-
5	Channel I+/Index Pulse+	IM+
6	Common	ECOM
7	Encoder Power (+9V)	EPWR_9V
8	Commutation Channel S3	S3

CN2 Pin	Description	Signal
9	Positive Overtravel Limit	+LIMIT
10	Channel I-/Index Pulse-	IM-
11	Thermostat	TS
12	Commutation Channel S1	S1
13	Commutation Channel S2	S2
14	Encoder Power (+5V)	EPWR_5V
15	Negative Overtravel Limit	-LIMIT

Use the figure below to locate the front panel connections on the Ultra3000 with DeviceNet Interface 460V drives (3 kW, 5 kW, 10 kW, 15 kW, and 22 kW).

Figure 2.12
Ultra3000 Front Panel Connections for 2098-DSD-HV*xxx*-DN and HV*xxx*X-DN



Serial Port Connector

The following table provides the signal descriptions and pin-outs for the CN3 serial port (9-pin) connector.

CN3 Pin	Description	Signal
1	RS-422/RS-485 Input+	RCV+
2	RS-232 Input	RCV
3	RS-232 Output	XMT
4	RS-422/RS-485 Output+	XMT+
5	Common	COM
6	Reserved	–
7	RS-422/RS-485 Input-	RCV-
8	RS-422/RS-485 Output-	XMT-
9	Reserved	–

I/O Connector

The following table provides the signal descriptions and pin-outs for the CN1 I/O (44-pin) connector.

CN1 Pin	Description	Signal
1	Auxiliary Encoder Power Out (+5V)	EPWR
2	Common	ECOM
3	Reserved	–
4	Auxiliary A+/Step+/CW+	AX+
5	Auxiliary A-/Step-/CW-	AX-
6	Auxiliary B+/Dir+/CCW+	BX+
7	Auxiliary B-/Dir-/CCW-	BX-
8	Auxiliary Encoder Ch I+	IX+
9	Auxiliary Encoder Ch I-	IX-
10	Unbuffered Motor Encoder Ch A+	AM+
11	Unbuffered Motor Encoder Ch A-	AM-
12	Unbuffered Motor Encoder Ch B+	BM+
13	Unbuffered Motor Encoder Ch B-	BM-
14	Unbuffered Motor Encoder Ch I+	IM+
15	Unbuffered Motor Encoder Ch I-	IM-
16	Buffered Motor Encoder Ch A+	AMOUT+
17	Buffered Motor Encoder Ch A-	AMOUT-
18	Buffered Motor Encoder Ch B+	BMOUT+
19	Buffered Motor Encoder Ch B-	BMOUT-
20	Buffered Motor Encoder Ch I+	IMOUT+
21	Buffered Motor Encoder Ch I-	IMOUT-
22	Common	ACOM
23	Programmable Analog Output	AOUT
24	Analog Current Limit Input	ILIMIT
25	Command +	COMMAND+
26	Command -	COMMAND-
27	I/O Common	IOCOM
28	I/O Common	IOCOM
29	I/O Power	IOPWR
30	I/O Power	IOPWR
31	Digital Input 1	INPUT1
32	Digital Input 2	INPUT2
33	Digital Input 3	INPUT3
34	Digital Input 4	INPUT4
35	Digital Input 5	INPUT5
36	Digital Input 6	INPUT6
37	Digital Input 7	INPUT7
38	Digital Input 8	INPUT8
39	Digital Output 1	OUTPUT1
40	Digital Output 2	OUTPUT2
41	Digital Output 3	OUTPUT3
42	Digital Output 4	OUTPUT4
43	Normally Open Relay Output+	RELAY+
44	Normally Open Relay Output-	RELAY-

Motor Encoder Connector

The following table provides the signal descriptions and pin-outs for the CN2 motor encoder (15-pin) connector.

CN2 Pin	Description	Signal
1	Channel A+/Sine Differential Input+	AM+
2	Channel A-/Sine Differential Input-	AM-
3	Channel B+/Cosine Differential Input+	BM+
4	Channel B-/Cosine Differential Input-	BM-
5	Channel I+/Index Pulse+	IM+
6	Common	ECOM
7	Encoder Power (+9V)	EPWR_9V
8	Commutation Channel S3	S3
9	Positive Overtravel Limit	+LIMIT
10	Channel I-/Index Pulse-	IM-
11	Thermostat	TS
12	Commutation Channel S1	S1
13	Commutation Channel S2	S2
14	Encoder Power (+5V)	EPWR_5V
15	Negative Overtravel Limit	-LIMIT

Understanding Ultra3000 I/O Specifications

A description of the Ultra3000 digital I/O power requirements and I/O signal specifications is provided on the following pages. Also included are I/O circuitry examples.

Digital I/O Power Supply

All Ultra3000 drives require an isolated external 12-24V power supply for proper operation of the digital I/O.

IMPORTANT Do not tie the 24V digital I/O common (CN1-27 and -28) to the auxiliary encoder +5V common (CN1-2).

The following table provides a description of the digital I/O power supply (CN1-29 and -30).

Parameter	Description	Minimum	Maximum
I/O Power Supply Voltage	Voltage range of the external power supply for proper operation of the digital I/O.	10.8V	26.4V
I/O Power Supply Current	Current draw from the external power supply for the digital I/O, not including the relay output usage.	—	300 mA

Auxiliary 5V Logic Supply

The Ultra3000 drives (2098-DSD-005, -010, and -020) require an external +5V power supply in applications in which it is necessary to maintain logic power when the AC line voltage is removed. The +24V I/O supply (IOPWR) allows use of the drive-mounted breakout boards with 24V to 5V dc converter (2090-U3CBB-DM12 and -DM44). The following table provides a description of the +24V (IOPWR) power supply requirements when used to maintain logic power.

Parameter	Description	Minimum	Maximum
Input Voltage Range	Input voltage range of the external power supply for drive-mounted breakout boards with 24V to 5V converter.	18V	30V
Input Current	Input current draw from the external power supply for the drive-mounted breakout boards with 24V to 5V converter.	—	400 mA

IMPORTANT A single 24V power supply can be used to power the digital I/O and supply 24V to the drive-mounted breakout boards (2090-U3CBB-DM*xx*) provided the cumulative minimum current requirements are met.

Two versions of the drive-mounted breakout board with 24V to 5V auxiliary power converter exist:

- 12-pin CN1 connector designed for use with SERCOS interface applications (catalog number 2090-U3CBB-DM12)
- 44-pin CN1 connector (catalog number 2090-U3CBB-DM44)

If an auxiliary +5V dc logic supply is used, the SERCOS ring remains active and motor position can be monitored by the drive even when the AC input power is removed. Since the drive is able to monitor the motor position, additional homing sequences can be avoided when the AC input power is re-applied.

IMPORTANT Only the 2098-DSD-005, -010, and -020 models support an auxiliary +5V logic supply since an auxiliary AC input is not available. Refer to *Chapter 3* for more information on the auxiliary AC input.

IMPORTANT Whenever the auxiliary +5V dc logic supply is used and the AC input supply is disconnected, the drive must be disabled. When the AC input supply is reconnected, the drive should not be re-enabled for at least 1.0 second, to allow the power stage circuitry to fully charge.

IMPORTANT Once the AC input supply is applied, the auxiliary +5V dc logic supply must not be interrupted. Removing the +5V dc logic supply with the AC input voltage applied will cause the drive to reboot and loss of control will occur.

Using an External +5V Logic Supply

When using an external +5V dc power supply with your Ultra3000 (2098-DSD-005, -010, and -020), the +5V dc must not be grounded inside the supply, since it will be referenced to the drive common. External +5V dc power supply connections should be made to CN1-2 and CN1-3.

IMPORTANT Using the drive-mounted breakout board with 24V to 5V auxiliary power converter is preferred to using an external +5V dc power supply.

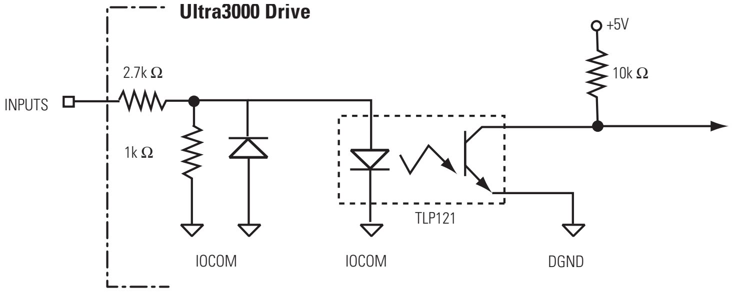
The following table provides a description of the requirements for an external +5V dc power supply used to power the logic.

Parameter	Description	Minimum	Maximum
Voltage	Voltage tolerance of the external logic supply.	5.1V	5.25V
Current	Current output capability of the external +5V dc power supply.	1.5A	—

Digital Inputs

There are eight opto-isolated digital inputs. All digital inputs (SERCOS and non-SERCOS) have the same configuration, as shown in the figure below.

Figure 2.13
Digital Input Circuit



The following table provides a description of the digital input specifications.

Parameter	Description	Minimum	Maximum
ON State Voltage	Voltage applied to the input, with respect to IOCOM, to guarantee an ON state.	10.8.V	26.4V
ON State Current	Current flow to guarantee an ON State	3.0 mA	12.0 mA
OFF State Voltage	Voltage applied to the input, with respect to IOCOM, to guarantee an OFF state.	-1.0V	2.0V
Propagation Delay	Signal propagation delay from the digital input to the firmware-accessible registers.	—	100 µS

On SERCOS drives, the following inputs have dedicated functionality.

Pin	Signal	Description
CN1-31	ENABLE	Drive Enable Input, an active state enables the power electronics to control the motor.
CN1-32	HOME	Home Sensor, an active state indicates to a homing sequence that the sensor has been seen.
CN1-33 CN1-34	REG1 REG2	Registration Sensor, a transition is used to record position values.
CN1-37 CN1-38	OT_POS OT_NEG	Overtravel Input, an inactive state indicates that a position limit has been exceeded. An active state occurs when 24V is removed from the input.

IMPORTANT Overtravel limit input devices must be normally closed.

On non-SERCOS drives, digital inputs can be configured for a variety of functions using Ultraware. Refer to Ultraware Help for digital input functionality.

Assigned Preselect Inputs

Active or inactive states select one of 64 presets shown in the following binary table. Active state indicates current flow through the input optocoupler. Inactive state indicates no current flow.

Preset Selects	Binary Code						Selected Preset or Index
	5	4	3	2	1	0	
Select up to 64 locations via preselect inputs 5 through 0 using BCD format. (codes for preset selects 1 and 0 are shown)	0	0	0	0	0	0	Preset 0 or Index 0 is selected.
	0	0	0	0	0	1	Preset 1 or Index 1 is selected.
	0	0	0	0	1	0	Preset 2 or Index 2 is selected.
	0	0	0	0	1	1	Preset 3 or Index 3 is selected.
			↓				↓
	1	1	1	1	1	1	Preset 64 or Index 64 is selected.

Input Interface Examples for Active High Inputs

Figure 2.14
Drive Input Connected to Switch/Relay Contact

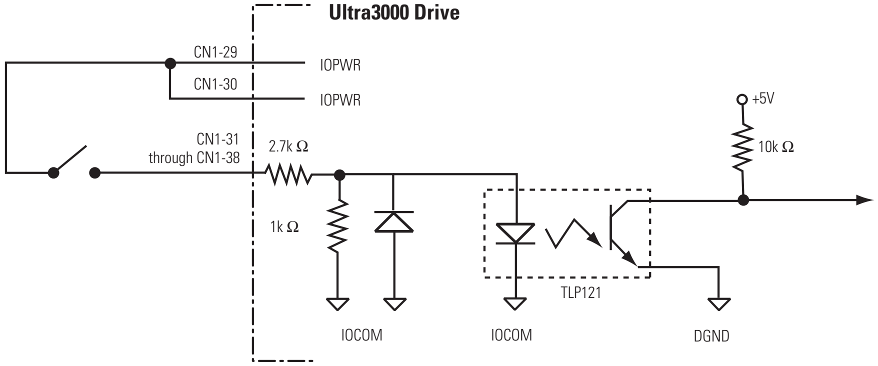


Figure 2.15
Drive Input Connected to Opto-Isolator

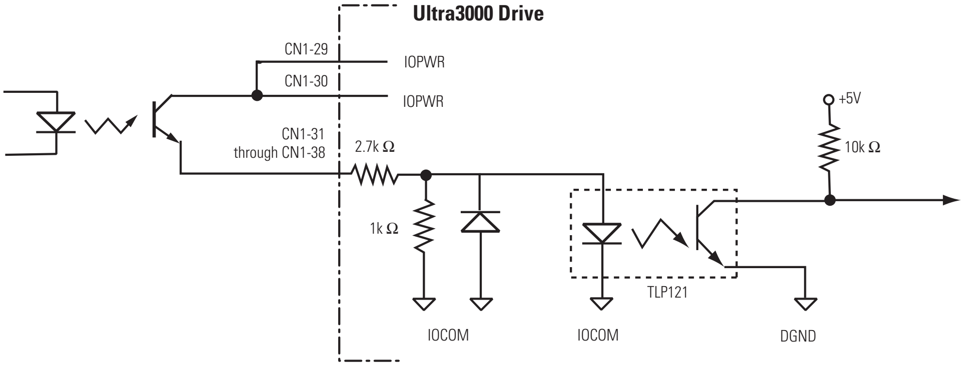


Figure 2.16
Drive Input Connected to NPN Transistor

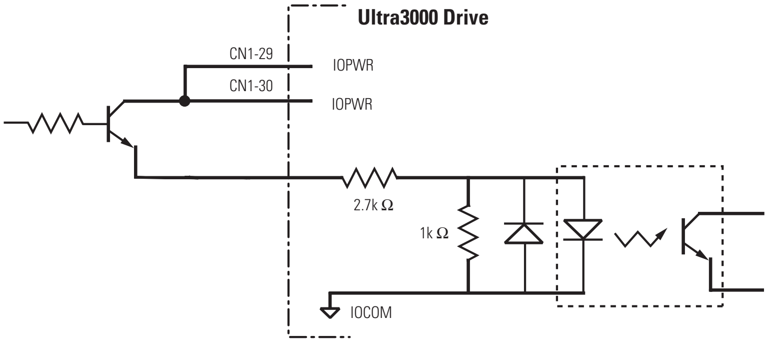


Figure 2.17
Drive Input Connected to NPN Transistor using Switch/Relay

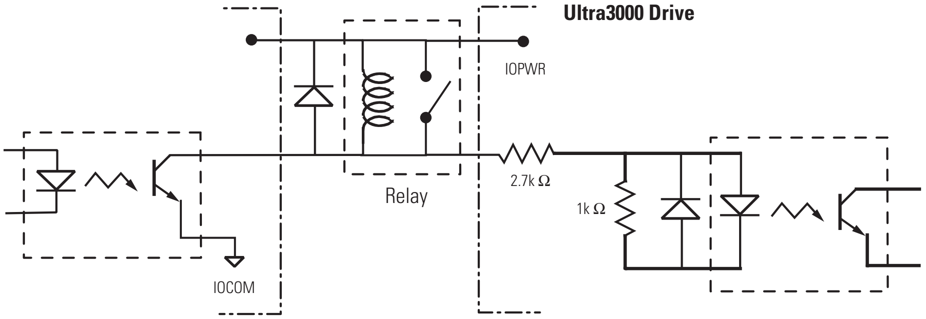


Figure 2.18
Drive Input Connected to NPN Transistor using Opto-Isolator

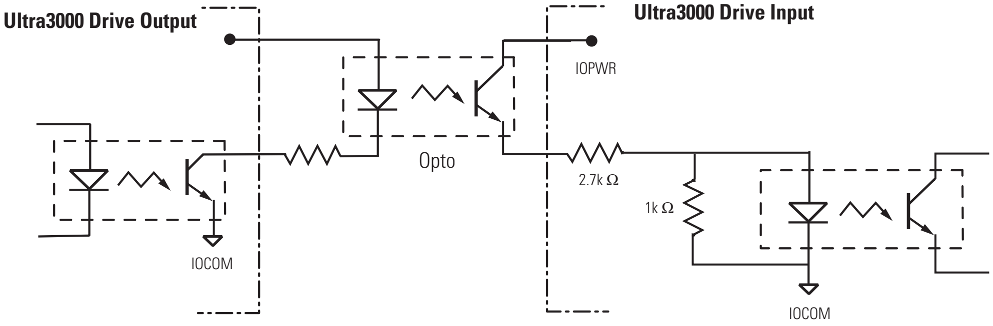


Figure 2.19
Drive Input Connected to another Ultra3000 Output

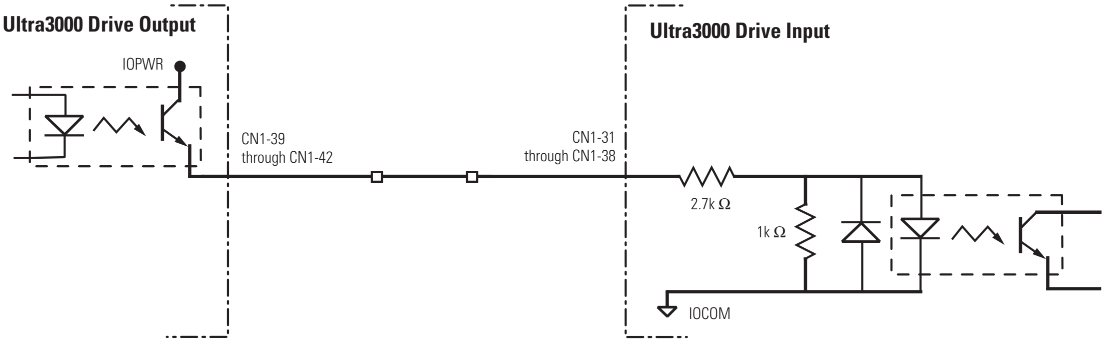
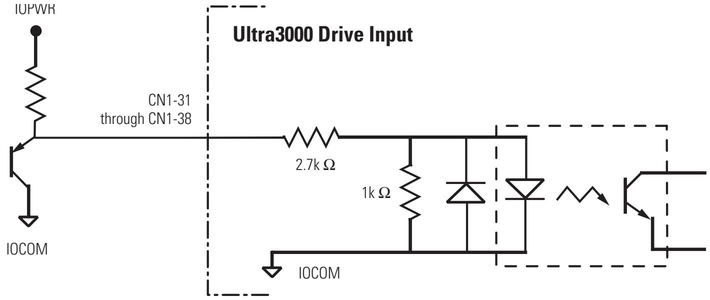


Figure 2.20
Drive Input Connected to PNP Transistor



Input Interface Examples for Active Low Inputs

Figure 2.21
Drive Input Connected to Normally Closed Switch

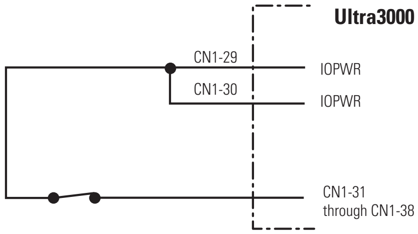


Figure 2.22
Drive Input Connected to Opto-Isolator

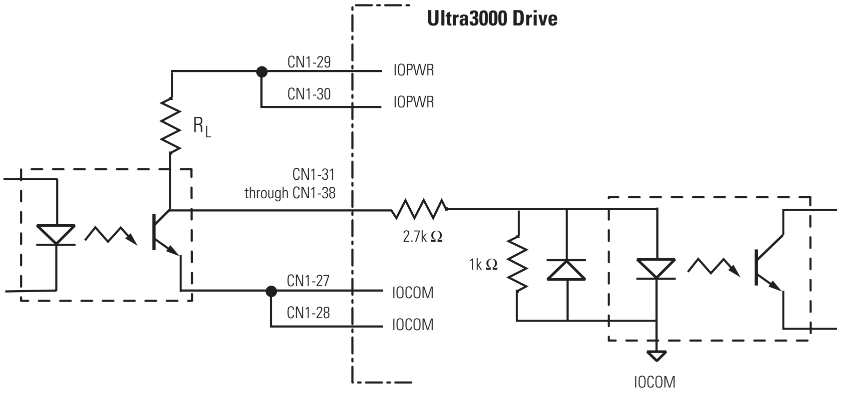


Figure 2.23
Drive Input Connected to NPN Transistor

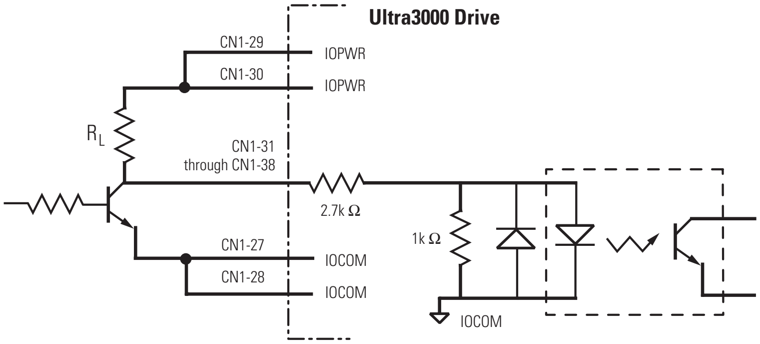
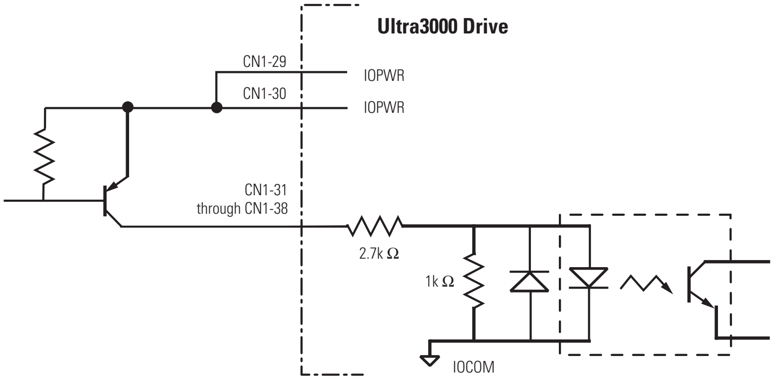


Figure 2.24
Drive Input Connected to PNP Transistor



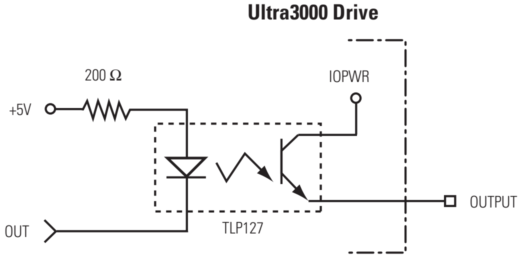
Digital Outputs

There are four opto-isolated transistor outputs that can be configured for a variety of functions through software. Additionally, the drive has a relay output with normally open contacts. On SERCOS drives, the relay output is dedicated as a Brake output, where closed contacts release a motor brake.

The configuration of the transistor outputs is shown in Figure 2.25, and the configuration of the relay output is shown in Figure 2.26.

IMPORTANT There is no overload protection on the transistor outputs.

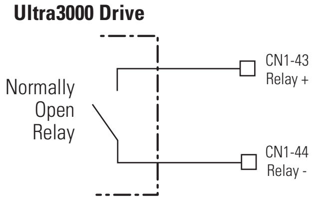
Figure 2.25
Transistor Output Hardware Configuration



The following table provides a description of the digital output specifications.

Parameter	Description	Minimum	Maximum
ON State Current	Current flow when the output transistor is ON	—	50 mA
OFF State Current	Current flow when the output transistor is OFF	—	0.1 mA
ON State Voltage	Voltage across the output transistor when ON	—	1.5V
OFF State Voltage	Voltage across the output transistor when OFF	—	50V

Figure 2.26
Relay Output Hardware Configuration



The following table provides a description of the relay output specifications.

Parameter	Description	Minimum	Maximum
ON State Current	Current flow when the relay is closed	—	1A
ON State Resistance	Contact resistance when the relay is closed	—	1Ω
OFF State Voltage	Voltage across the contacts when the relay is open	—	30V

Drive Output Interface Examples

Figure 2.27
Drive Output Connected to an Opto-Isolator

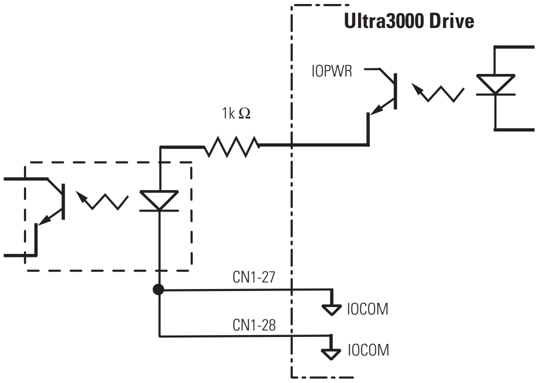


Figure 2.28
Drive Output Connected to an LED Indicator

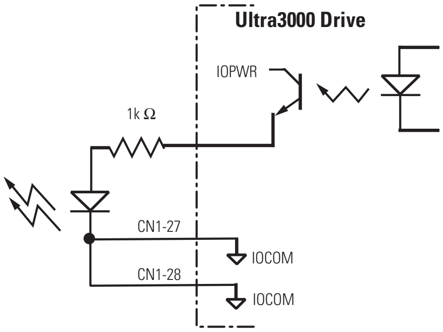


Figure 2.29
Drive Output Connected to a Resistive Load

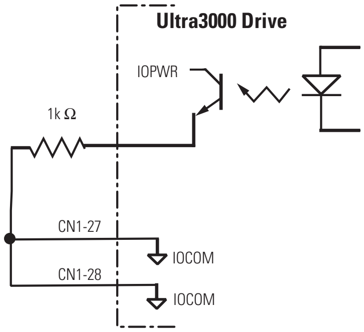


Figure 2.30
Drive Output Connected to a Switch/Relay

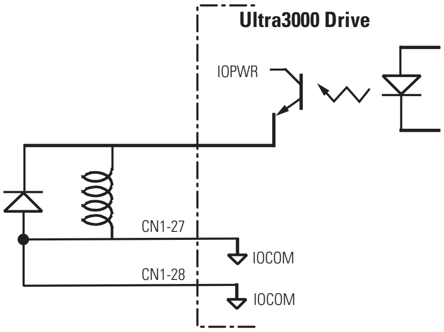


Figure 2.31
Drive Output Connected to an Active Low Input using a Switch/Relay

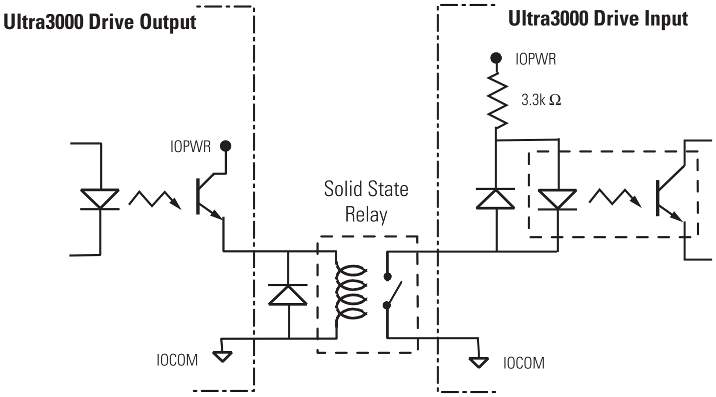


Figure 2.32
Drive Output Connected to an Active Low Input using an Opto-Isolator

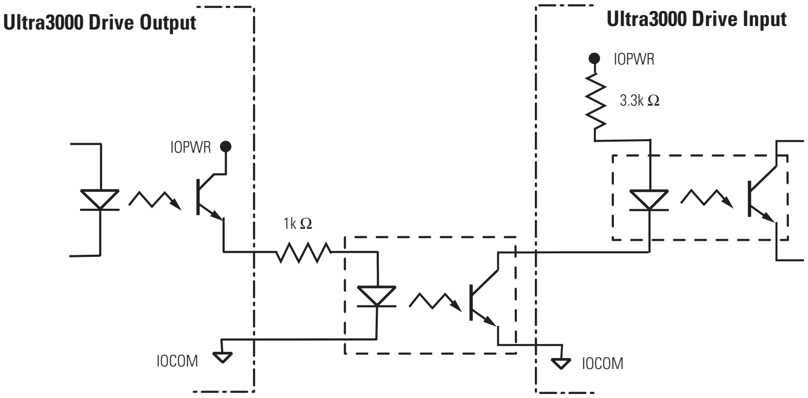
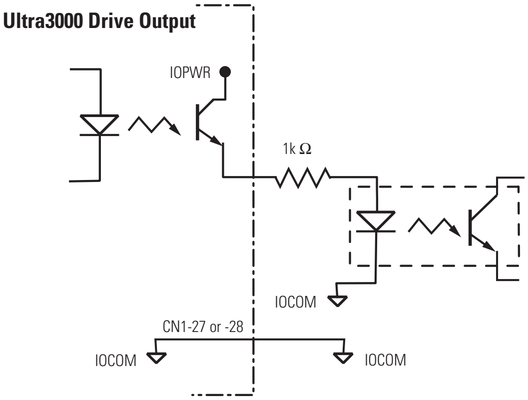


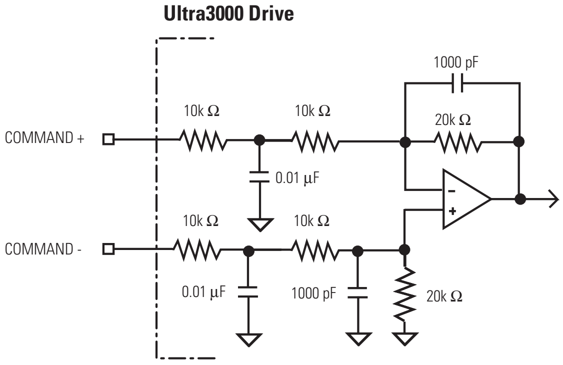
Figure 2.33
Drive Output Connected to an Active High (sinking) Input



Analog COMMAND Input

The COMMAND input to the drive can provide a position, velocity, or current command signal. A 14 bit A/D converter digitizes the signal. The configuration of the input is shown in Figure 2.34.

Figure 2.34
Analog COMMAND Input Configuration



The following table provides a description of the analog COMMAND input specifications.

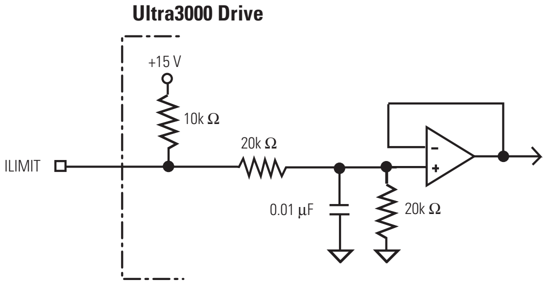
Parameter	Description	Minimum	Maximum
Resolution	Number of states that the input signal is divided into which is $2^{(\text{to the number of bits})}$.	14 bits	—
Input Impedance	Open circuit impedance measured between the + and - inputs.	20 kΩ	—
Input Signal Range	Voltage applied to the input	-10V	+10V
Offset Error	Deviation from the correct value expected from analog-to-digital conversion when 0V is applied to the input.	—	50 mV
Gain Error	Deviation of the transfer function from unity gain, expressed in a percent of full scale.	—	1%
Propagation Delay	Delay from the input to the firmware-accessible registers.	—	100 µS

Analog ILIMIT Input

The ILIMIT input specifies to the drive if the drive output current should be limited. If the ILIMIT input is not connected, current is not limited. A 10 bit A/D converter digitizes the signal. The configuration of the ILIMIT input is shown in Figure 2.35.

The input range is 0 to 10V, and the drive current is limited inversely proportional to the input voltage. A +10V input corresponds to no current limiting, and a 0V input prevents any drive current.

Figure 2.35
Analog ILIMIT Input Configuration



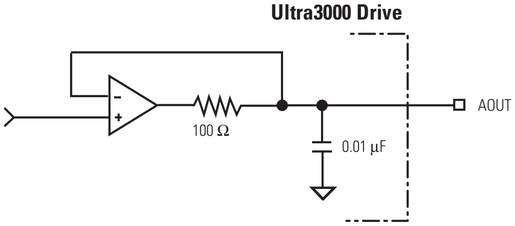
The following table provides a description of the analog ILIMIT input specifications.

Parameter	Description	Minimum	Maximum
Resolution	Number of states that the input signal is divided into which is $2^{(\text{to the number of bits})}$.	10 bits	—
Input Impedance	Open circuit impedance measured between the input and analog common.	10 kΩ	—
Input Signal Range	Voltage applied to the input	0V	+10V
Offset Error	Deviation from the correct value expected from analog-to-digital conversion when 0V is applied to the input.	—	50 mV
Gain Error	Deviation of the transfer function from unity gain, expressed in a percent of full scale.	—	1%
Propagation Delay	Delay from the input to the firmware-accessible registers.	—	100 µS

Analog Output

The Ultra3000 includes a single analog output (not supported on the SERCOS models) that can be configured through software to represent drive variables. Figure 2.36 shows the configuration of the analog output.

Figure 2.36
Analog Output Configuration



IMPORTANT Output values can vary during power-up until the specified power supply voltage is reached.

The following table provides a description of the analog output specifications.

Parameter	Description	Minimum	Maximum
Resolution	Number of states that the output signal is divided into, which is $2^{\text{(to the number of bits)}}$.	8 Bits	—
Output Current	Current capability of the output.	-2 mA	+2 mA
Output Signal Range	Range of the output voltage.	-10V	+10V
Offset Error	Deviation when the output should be at 0V.	—	500 mV[1]
Gain Error	Deviation of the transfer function from unity gain, expressed in a percent of full scale.	—	10%[1]
Bandwidth	Frequency response of the analog output	50 Hz	—

[1] The offset and gain errors of the analog output can be corrected for an application using Ultraware scale and offset settings.

Understanding Motor Encoder Feedback Specifications

The Ultra3000 can accept motor encoder signals from the following types of encoders:

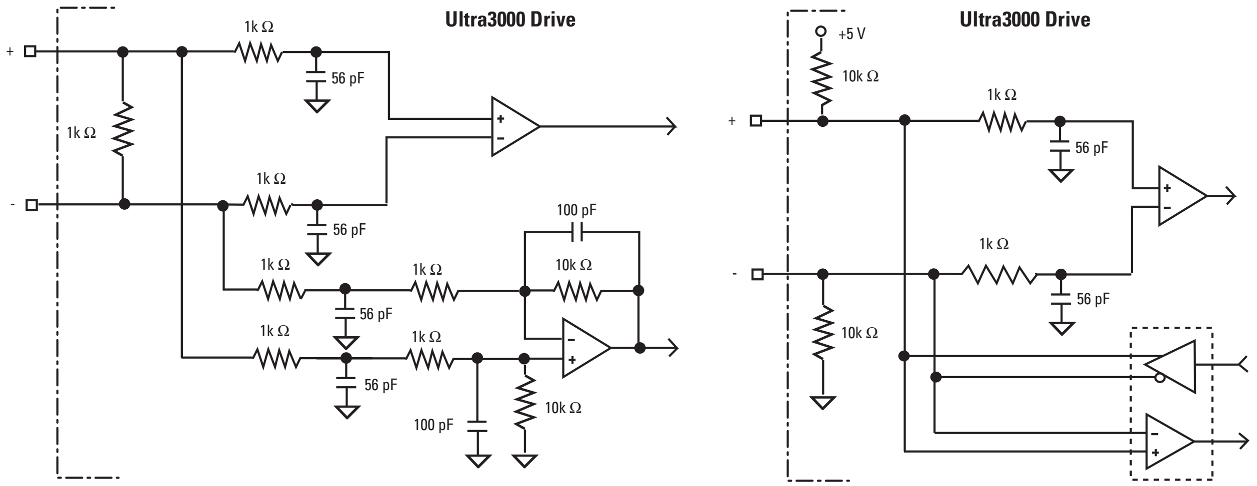
- Incremental encoders with TTL outputs, with or without Hall signals
- Sine/Cosine encoders, with or without Hall signals
- Intelligent absolute encoders
- Intelligent high-resolution encoders
- Intelligent incremental encoders

Note: The intelligent absolute, high-resolution, and incremental encoders are available only in Allen-Bradley motors.

AM, BM, and IM Inputs

AM, BM, and IM Input encoder signals are filtered using analog and digital filtering. The inputs also include illegal state change detection. Refer to Figure 2.37 for a schematic of the AM, BM, and IM inputs.

Figure 2.37
Schematic of the Motor Encoder Inputs



AM and BM Channel Inputs

IM Channel Input

The Ultra3000 supports both TTL and Sine/Cosine encoders. The following table provides a description of the AM, BM, and IM inputs for TTL encoders.

Parameter	Description	Minimum	Maximum
AM, BM, and IM ON State Input Voltage	Input voltage difference between the + input and the - input that is detected as an ON state.	+1.0V	+7.0V
AM, BM, and IM OFF State Input Voltage	Input voltage difference between the + input and the - input that is detected as an OFF state.	-1.0V	-7.0V
Common Mode Input Voltage	Potential difference between any encoder signal and logic ground.	-7.0V	+12.0V
DC Current Draw	Current draw into the + or - input.	-30 mA	30 mA
AM, BM Input Signal Frequency	Frequency of the AM or BM signal inputs. The count frequency is 4 times this frequency, since the circuitry counts all four transitions.	—	2.5 MHz
IM Pulse Width	Pulse width of the index input signal. Since the index is active for a percentage of a revolution, the speed will determine the pulse width.	125 nS	—
AM / BM Phase Error, 2.5 MHz Line Frequency	Amount that the phase relationship between the AM and BM inputs can deviate from the nominal 90°.	-22.5°	+22.5°
AM / BM Phase Error, 1 MHz Line Frequency	Amount that the phase relationship between the AM and BM inputs can deviate from the nominal 90°.	-45°	+45°

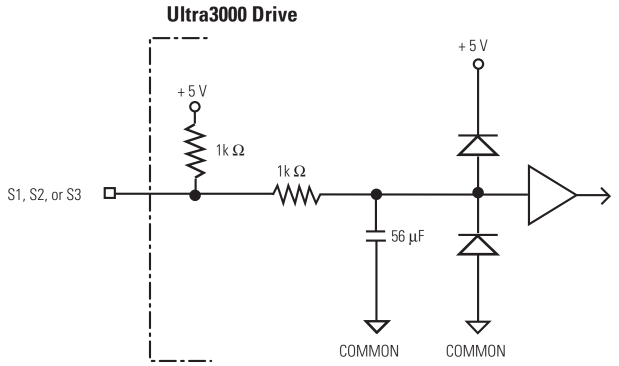
The following table provides a description of the AM and BM inputs for Sine/Cosine encoders.

Parameter	Description	Minimum	Maximum
Sine and Cos Input Signal Frequency	Frequency of the Sine or Cos signal inputs.	—	100 kHz
Sine and Cos Input Voltage	Peak-to-peak input voltages of the Sine and Cos inputs	0.5V (p-p)	2.0V (p-p)

Hall Inputs

The Ultra3000 can use Hall signals to initialize the commutation angle for sinusoidal commutation. Hall signals must be single-ended and can be either open collector type or TTL type. Figure 2.38 shows the configuration of the Hall inputs. If the motor does not have Hall signals, the drive can be configured through software to ignore the signals.

Figure 2.38
Hall Input Configuration



Thermostat Input

The Ultra3000 can monitor a thermostat signal from a motor and will generate a fault if the motor overheats. Figure 2.39 shows the configuration of the thermostat input. Figure 2.40 on page 2-43 shows a typical connection to a motor with a normally closed thermostat. The logic is designed so that an open condition will generate a fault. If the motor does not have a thermostat signal, the drive can be configured through software to ignore the signal.

Figure 2.39
Thermostat Input Configuration

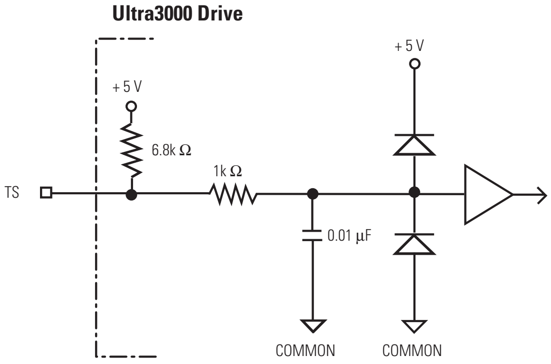
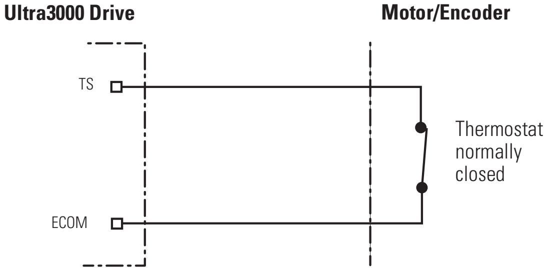


Figure 2.40
Typical Thermostat Connection



+ Limit and - Limit Inputs

The Ultra3000 drive includes integral overtravel limit inputs on the motor encoder connector (CN2). The logic is designed so that an open condition will halt motion in the corresponding direction. The integral limits are configured by the actual motor file and not software programmable. Although typically not for use on standard servo motors, they may be activated for linear motors or other unique applications. Figure 2.41 shows the configuration of the +Limit and -Limit inputs. Figure 2.42 shows a typical connection to a motor with integral limit switches.

Figure 2.41
+ Limit and - Limit Input Configuration

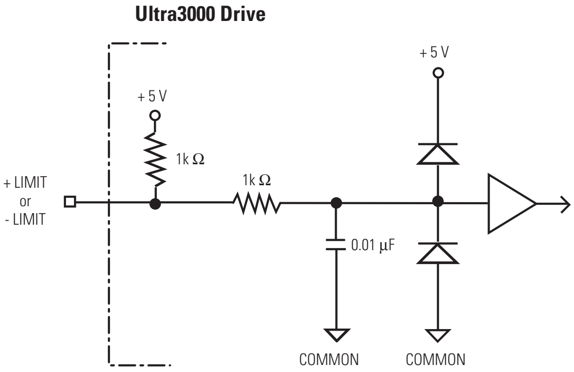
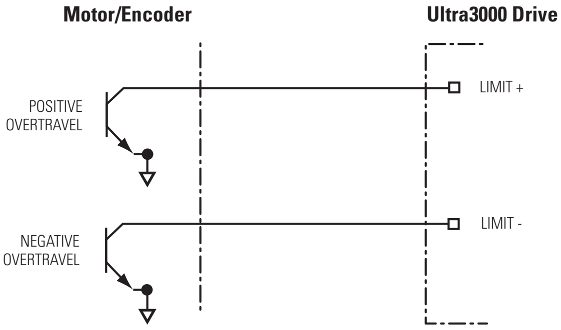


Figure 2.42
Typical + Limit and - Limit Connection



Encoder Phasing

For proper motor commutation and control, it is important that the motor feedback signals are phased properly. The drive has been designed so that a positive current applied to a motor will produce a positive velocity and increasing position readings, as interpreted by the drive. Additionally, if Hall signals are used to initialize the commutation angle, the Hall signals must sequence properly and the phase relationship to the motor back-EMF signals must be understood. Figure 2.43 shows the proper sequencing of the Hall signals when positive current is applied to the motor. If the Hall signals are out of phase with the back-EMF signals, the drive can be configured through software to compensate for the phase offset, as long as the sequencing of the Hall signals is correct. Figure 2.44 shows an example where the Hall signals have an offset of 60 degrees.

Figure 2.43
Sequencing and Phasing of the Hall Signals

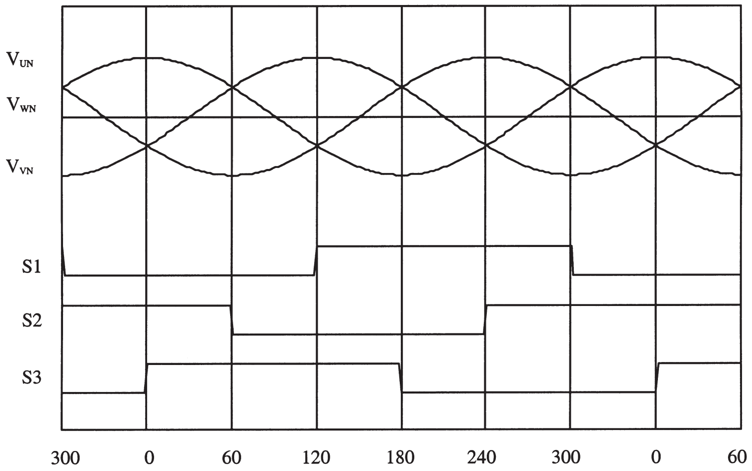


Figure 2.44
Sequencing and Phasing of the Hall Signals (60° Hall Offset Example)

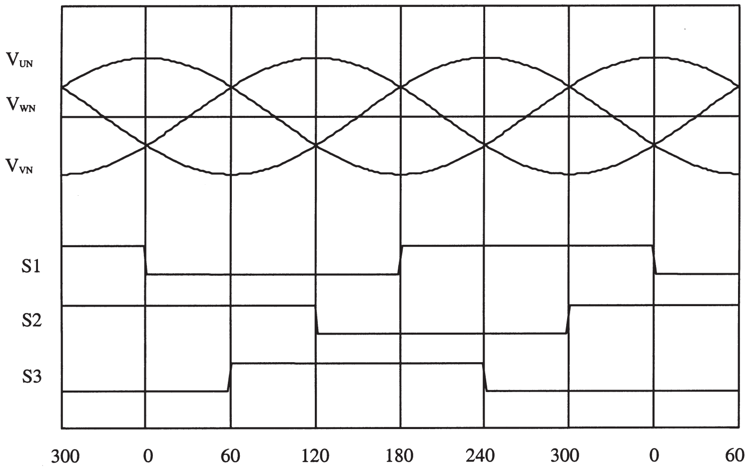


Figure 2.45 shows the proper phasing of TTL A/B encoder signals when positive current is applied.

Figure 2.45
Phasing of TTL A/B Encoder Signals

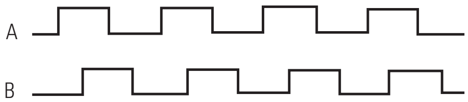
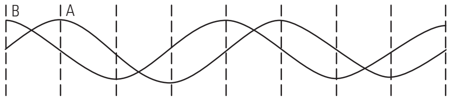


Figure 2.46 shows the proper phasing of Sine/Cosine encoder signals when positive current is applied.

IMPORTANT Notice that the Sine/Cosine encoder signal phasing is different than the phasing of the TTL encoders.

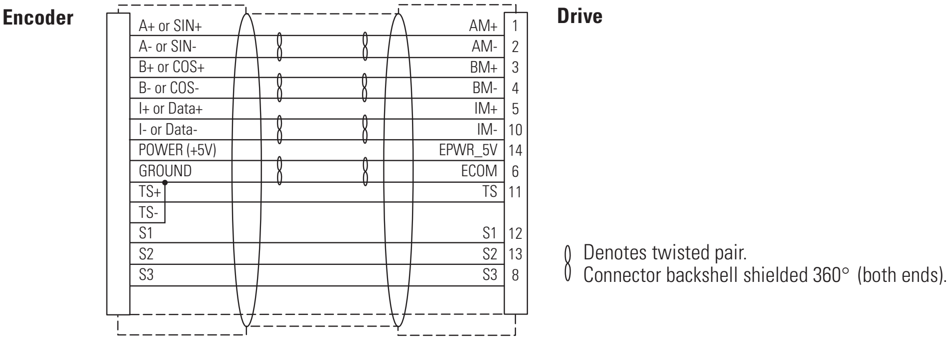
Figure 2.46
Phasing of Sine/Cosine Encoder Signals



Motor Encoder Connection Diagram

Figure 2.47 shows a typical wiring diagram of a motor feedback cable. If the thermostat, limit, or Hall signals are not available, no connections are required, but the drive must be configured through software to ignore these signals. Refer to *Appendix B* for specific Ultra3000 drive/motor interconnect diagrams.

Figure 2.47
Drive/Motor Wiring Diagram



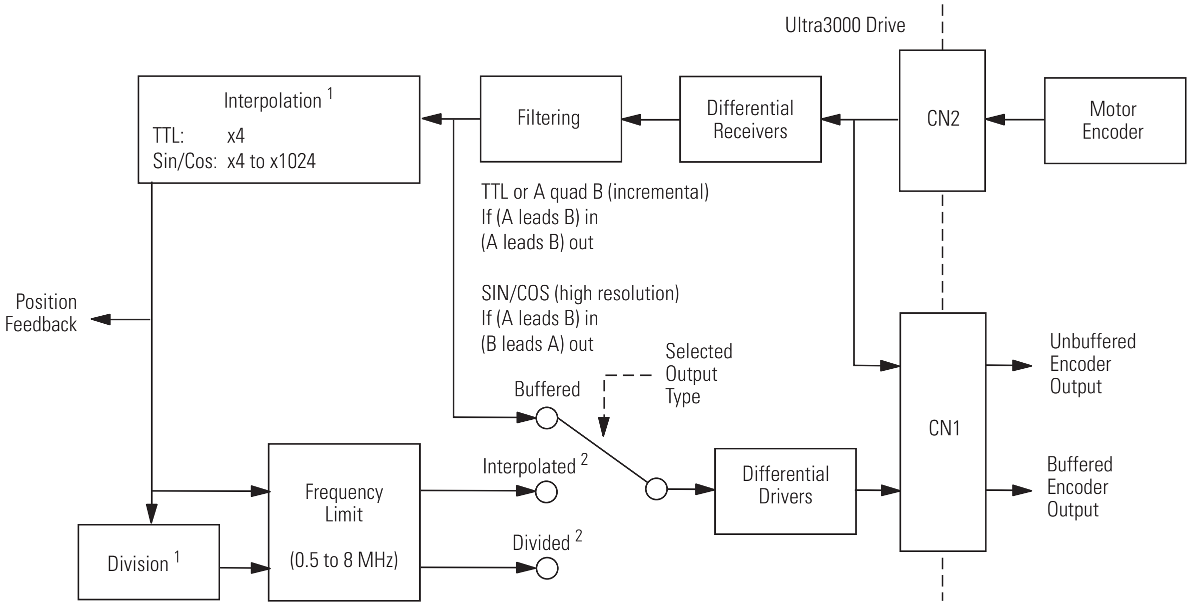
IMPORTANT Total resistance of the wiring for +5V encoder power and ground connections between the drive and motor must be less than 1.4 ohms.

Understanding Motor Feedback Signals and Outputs

The Ultra3000 is compatible with motors equipped with both incremental A quad B or high resolution (Stegmann Hiperface®) SIN/COS encoders.

The buffered motor encoder outputs use RS-485 differential drivers and have a maximum signal frequency of 2.5 MHz. The drivers can drive a 2V differential voltage into a 100 ohm load. Use the block diagram below to follow the motor encoder input through CN2 to the buffered and unbuffered outputs on CN1.

Figure 2.48
Motor Encoder Outputs



[1] Interpolation and division operations are performed in firmware and the resulting output frequency is updated at 250 µs intervals.

[2] Interpolated and divided output not available on SERCOS drives.

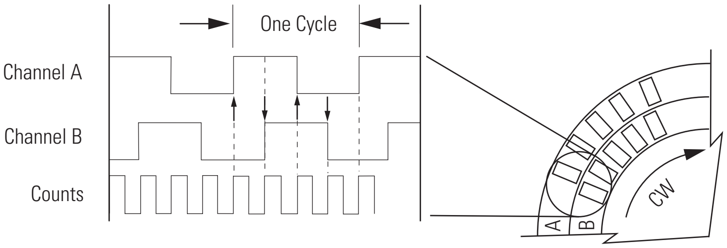
Unbuffered Encoder Outputs

The unbuffered outputs available from the drive (CN1-10 through -15) are tied directly to the incoming (incremental or high resolution) encoder signals (CN2-1 through -6). The unbuffered outputs are not filtered or conditioned.

Incremental Encoder Output

Incremental encoder counts are generated in the drive by counting the (high to low and low to high) transitions of the incoming A and B encoder signals. In Figure 2.49 the channel A signal has two transitions, as does the channel B signal, which results in x4 interpolation (4 transitions/line equals 4 counts/line). So, for example, typical 2000 line/rev encoder output becomes 8000 counts/rev in the drive. Counts are not directly available at the encoder outputs, only the A quad B representation.

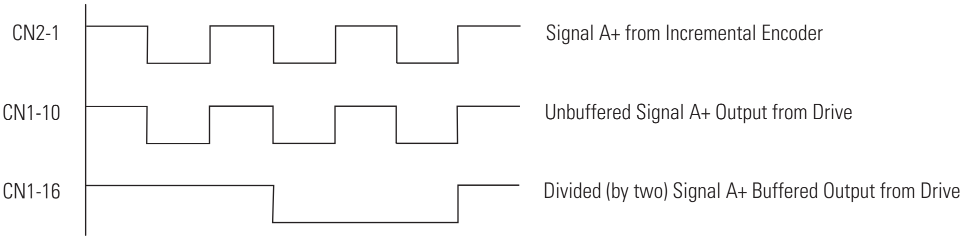
Figure 2.49
Incremental Encoder Counts



The incremental buffered outputs available from the drive (CN1-16 through -21) are software selectable as follows:

- **Buffered Outputs** are a filtered representation of the original incoming encoder (CN2) signals. Buffered outputs have the same number of cycles/rev as found on CN2.
- **Interpolated Outputs** are the same as buffered outputs when using an incremental encoder. The only interpolation performed on an A quad B signal is the drive's internal counting of transitions (4 counts/line). Because counts are not available outside the drive, selecting this in software is the same as selecting buffered (as described above).
- **Divided Outputs** are the same as buffered outputs, except when divided is selected in the software, the lines/rev are then reduced by the value of the divisor chosen in the software (as shown in the figure below).

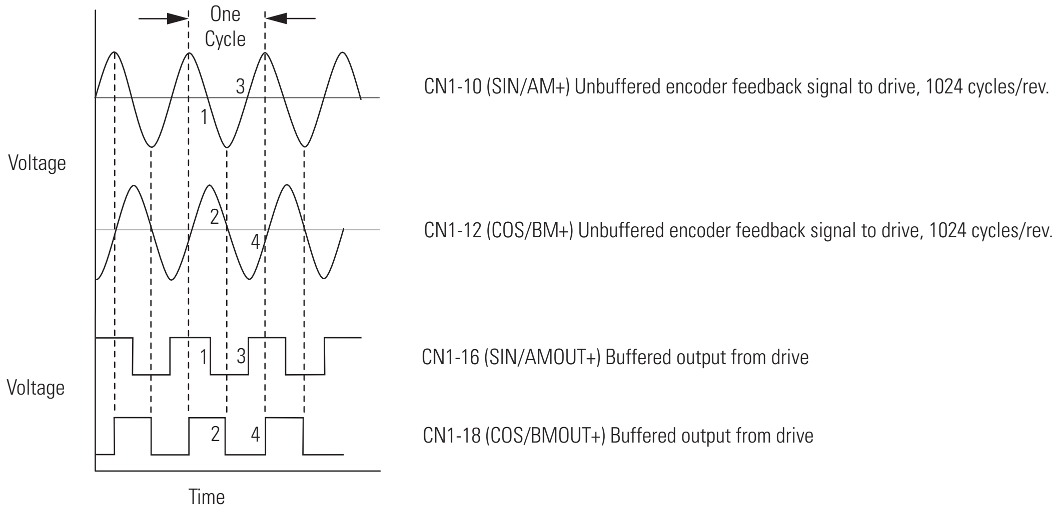
Figure 2.50
Incremental Encoder Divided



High Resolution Encoder Output

When the incoming encoder feedback on CN2 is a high resolution (SIN/COS) signal, the drive is capable of generating more than just 4 counts/cycle (as with incremental encoders). The Ultra3000 drive is capable of breaking the SIN/COS encoder signals into as many as 1024 counts/cycle. So, for example, a 1024 cycle/rev SIN/COS encoder can result in 1024 x 1024 (high resolution) counts/rev.

Figure 2.51
Absolute High Resolution Encoder Signals

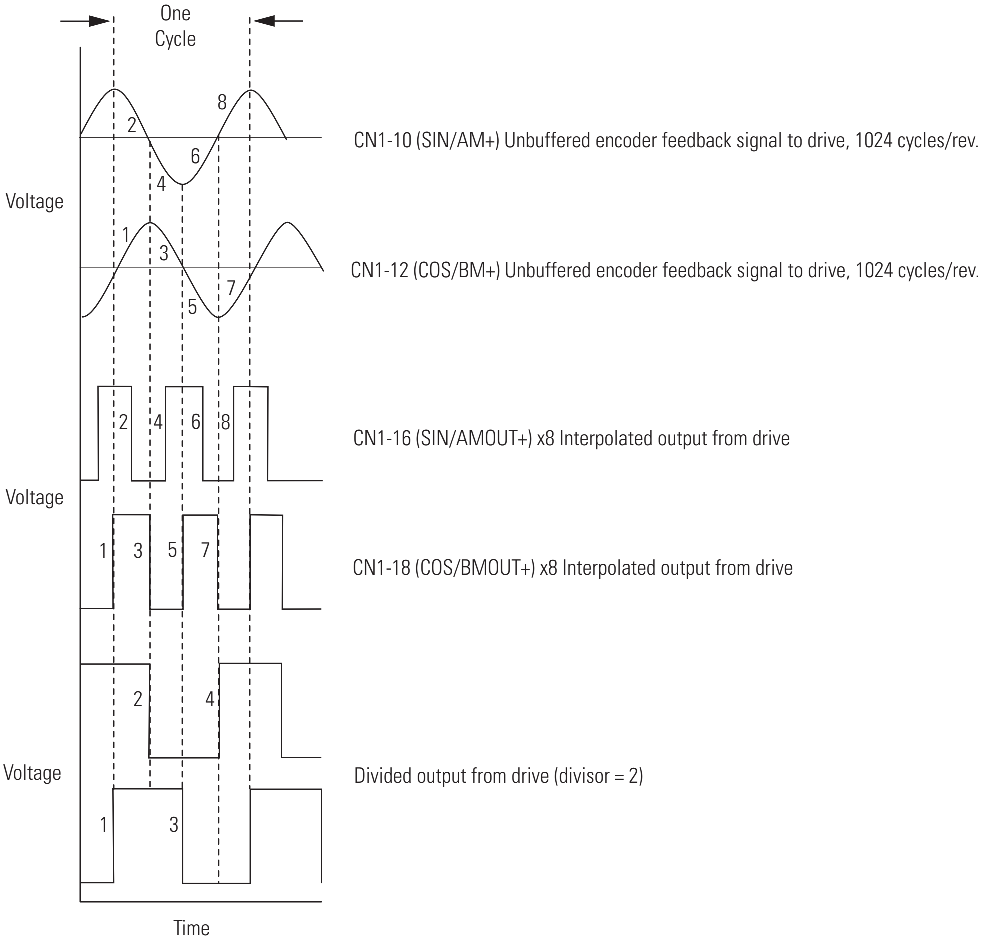


The high resolution buffered outputs available from the drive (CN1-16 through -21) are software selectable as follows:

- **Buffered Outputs** are conditioned SIN/COS signals resulting in a square wave (A quad B) signal (refer to Figure 2.51). This signal will have the same number of cycles/rev as the incoming SIN/COS encoder signals found on CN2.
- **Interpolated Outputs** are square wave (A quad B) signals reflecting the interpolation value chosen in software. The minimum interpolation value allowed is x4, which gives the same output as selecting buffered (as described above).
- **Divided Outputs** are the result of a divisor (selected in software) and an interpolation value (also selected in software). For example, with an interpolation value of x8 and a divisor of 2, the CN1 buffered output will be the (x4) square wave representation of the original incoming SIN/COS signal from CN2.

IMPORTANT The interpolation value selected in software is what the drive uses internally to close the feedback loops regardless of any divisor value chosen to condition the signals present on CN1.

Figure 2.52
Interpolated and Divided Absolute High Resolution Encoder Counts



Understanding Auxiliary Encoder Feedback Specifications

The Ultra3000 can accept an auxiliary encoder signal of the following types.

Figure 2.53
Auxiliary Encoder Input Signal Types

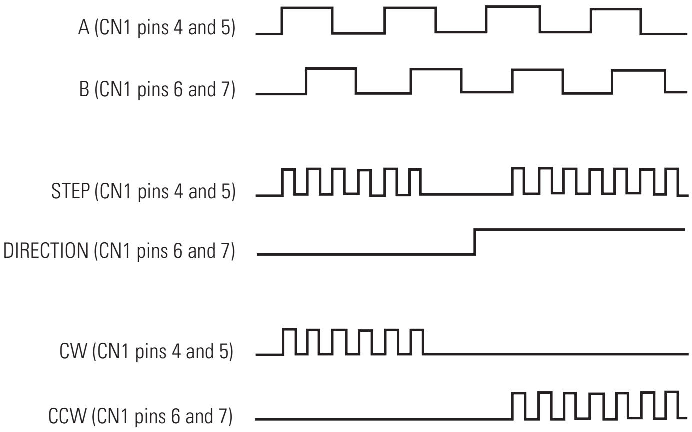
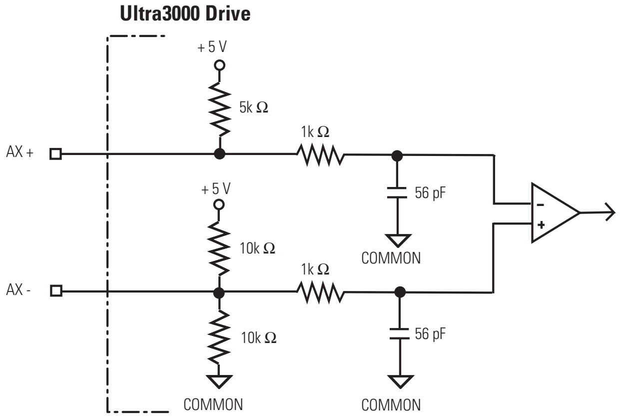


Figure 2.54 shows the configuration of the AX Auxiliary Encoder Input channel. The BX and IX channels have the same configuration.

Note: CW pulses are only counted when the CCW input is low, and CCW pulses are only counted when the CW input is low.

Figure 2.54
Auxiliary Encoder Input Configuration



Note: For single-ended connections, the negative terminals (CN1 pins 5 and 7) should be left disconnected, and the signal connections should be made to CN1 pins 4 and 6.

The following table provides a description of the auxiliary encoder interface.

Parameter	Description	Minimum	Maximum
ON State Input Voltage	Input voltage difference between the + input and the - input that is detected as an ON state.	+1.0V	+7.0V
OFF State Input Voltage	Input voltage difference between the + input and the - input that is detected as an OFF state.	-1.0V	-7.0V
Common Mode Input Voltage	Voltage between an input and logic ground.	-7.0V	+12.0V
Signal Frequency	Frequency of the AX or BX signal inputs. Count frequency is 4 times this frequency for A/B type inputs, and equal to this frequency for Step/Dir and CW/CCW type inputs.	—	2.5 MHz
Pulse Width	Time interval that a Step/Dir type input or CW/CCW type input must remain in a single state for detection.	200 nS	—
Setup Time	Time interval that the Direction, CW, or CCW must be stable before the corresponding Step, CCW, or CW signal changes state.	200 nS	—

5V Auxiliary Encoder Power Supply

All Ultra3000 drives supply 5V dc for the operation of an auxiliary encoder. The following table provides a description of the auxiliary encoder power supply.

Parameter	Description	Minimum	Maximum
Output Voltage	Voltage range of the external power supply for proper operation of an auxiliary encoder.	4.75V	5.25V
Output Current	Current draw from the external power supply for the auxiliary encoder.	—	250 mA

Pin	Signal	Description
CN1-1	EPWR	Auxiliary Encoder Power Out (+5V)
CN1-2	ECOM	Common

IMPORTANT The internal 5V dc power supply has a resettable fuse that opens at 3 amps and automatically resets itself when the current falls below 3 amps. There are no internal fuses requiring replacement.

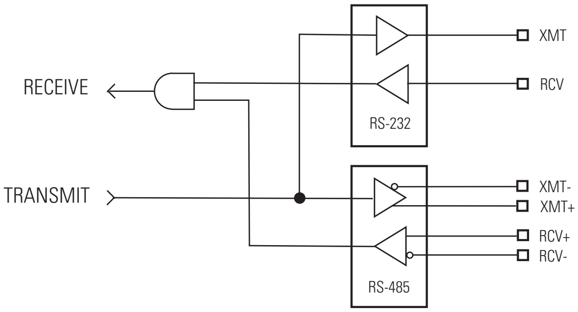
Understanding the Serial Interface

The Ultra3000 includes one serial port that implements the standard NRZ asynchronous serial format, and supports RS-232, RS-422, and RS-485 communication standards.

Standard baud rates include 1,200, 2,400, 4,800, 9,600, 19,200, and 38,400 baud. Data lengths of 7 and 8 bits are supported. Parity settings include odd, even, and none.

The connector pinout dedicates separate pins for the RS-232 and RS-422/ RS-485 signals, so that the communication standard can be changed by just using a different cable. Refer to Figure 2.55 for the serial interface configuration.

Figure 2.55
Serial Interface Configuration

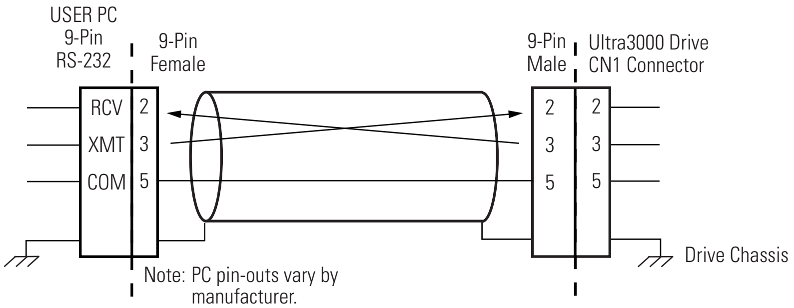


Default Serial Interface Settings

The default setting of the Ultra3000 serial interface is as follows.

Parameter	Default Setting
Baud Rate	38,400
Frame Format	8 Data, No Parity, One Stop
Drive Address	0

Figure 2.56
RS-232 Connection Diagram

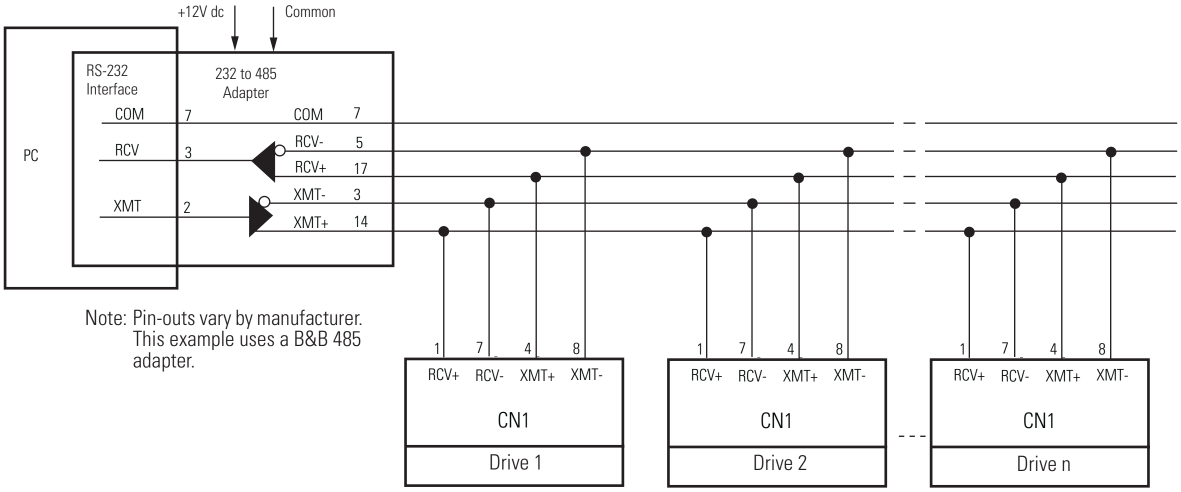


Note: PC pin-outs vary by manufacturer.

Multiple Axes RS-232 Communications

You can control multiple axes systems with a computer equipped with an RS-232 serial port. An RS-232 serial communication port may be converted to four wire RS-485 communication by attaching an RS-232 to four wire RS-485 converter. The figure below depicts the use of such a device.

Figure 2.57
RS-232 to RS-485 Connection Diagram



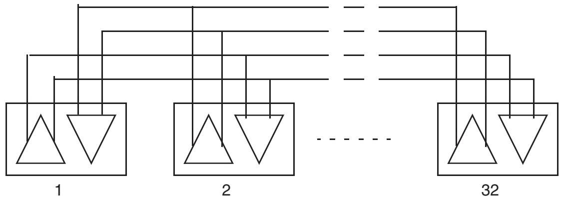
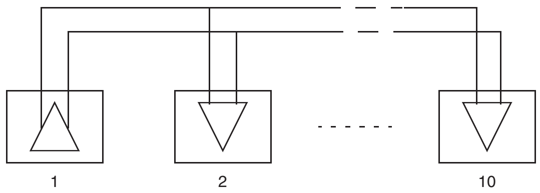
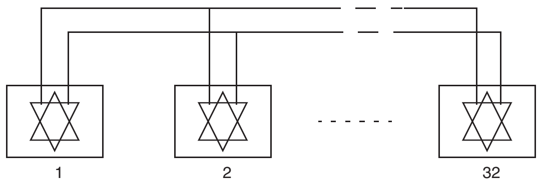
Note: Pin-outs vary by manufacturer. This example uses a B&B 485 adapter.

Four-Wire RS-485 Connections

The Ultra3000 uses a variation of the RS-485 standard, known as four-wire RS-485. Four-wire RS-485 uses one differential signal for host to drive transmissions, and another differential signal for drive to host transmissions. The RS-485 standard specifies a single differential signal for transmissions in both directions.

The four-wire RS-485 configuration also allows the host to use a RS-422 interface type. Because the host is driving multiple receivers and receiving from multiple transmitters, RS-422 is limited to multiple axes connections with 10 or less drives. The figure below summarizes the four-wire RS-485, RS-422, and RS-485 standards.

Figure 2.58
RS-485/RS-422 Communication Comparison

Four-Wire RS-485 Differential 4 Wires 2 Signal Pairs 1 to 32 Transmitters 1 to 32 Receivers	
RS-422 Differential 4 Wires 2 Signal Pairs 1 Transmitter 1 to 10 Receivers	
RS-485 Standard Differential 2 Wires 1 Signal Pair 1 to 32 Transmitters 1 to 32 Receivers Note: Not applicable to Ultra3000 drives	

Restoring Drive Communications

The Ultra3000 includes a mechanism for restoring serial communications, in case the drive has unknown serial interface settings or communications cannot be established.

For the first 3 seconds after reset or power-up, the drive listens for messages with the following serial interface settings.

Parameter	Default Setting
Baud Rate	9,600
Frame Format	8 Data, No Parity, One Stop
Drive Address	254

If a message is received during this time, the drive will respond and these settings will be retained until the next reset or power-down, allowing the normal serial interface settings to be determined. If no messages are received during this time, the normal serial interface settings are used.

IMPORTANT Only one drive should be connected if this mechanism is used, since multiple drives would all respond and the response would be garbled.

Connecting Your Ultra3000

Chapter Objectives

This chapter provides procedures for wiring your Ultra3000 and making cable connections. This chapter includes:

- Understanding Basic Wiring Requirements
- Determining Your Type of Input Power
- Grounding Your Ultra3000
- Power Wiring Requirements
- Connecting Input Power
- Connecting Motor Power and Brakes
- Understanding Shunt Connections
- Understanding Feedback and I/O Cable Connections
- Connecting Your SERCOS Fiber-Optic Cables
- Connecting to a DeviceNet Network

Understanding Basic Wiring Requirements

This section contains basic wiring information for the Ultra3000.

ATTENTION

Plan the installation of your system so that you can perform all cutting, drilling, tapping, and welding with the system removed from the enclosure. Because the system is of the open type construction, be careful to keep any metal debris from falling into it. Metal debris or other foreign matter can become lodged in the circuitry, which can result in damage to components.

IMPORTANT

This section contains common PWM servo system wiring configurations, size, and practices that can be used in a majority of applications. National Electrical Code, local electrical codes, special operating temperatures, duty cycles, or system configurations take precedence over the values and methods provided.

Building Your Own Cables

IMPORTANT Factory made cables are designed to minimize EMI and are recommended over hand-built cables to ensure system performance.

When building your own cables, follow the guidelines listed below.

- Connect the cable shield to the connector shells on both ends of the cable for a complete 360° connection.
- Use a twisted pair cable whenever possible, twisting differential signals with each other, and single-ended signals with the appropriate ground return.

Refer to *Appendix C* for drive connector kit catalog numbers.

Routing Power and Signal Wiring

Be aware that when you route power and signal wiring on a machine or system, radiated noise from nearby relays, transformers, and other electronic drives, can be induced into motor or encoder feedback, communications, or other sensitive low voltage signals. This can cause system faults and communication problems.

Refer to *Chapter 1* for examples of routing high and low voltage cables in wireways. Refer to *System Design for Control of Electrical Noise* (publication GMC-RM001*x*-EN-P) for more information.

Determining Your Type of Input Power

On the following pages are examples of typical single-phase and three-phase facility input power wired to single-phase and three-phase Ultra3000 drives.

IMPORTANT The Ultra3000 (2098-DSD-HV*xxx*) 460V drives are designed to operate from grounded or ungrounded power configurations. For systems requiring CE or for Ultra3000 (2098-DSD-*xxx*) 230V drives, the supply must be grounded.

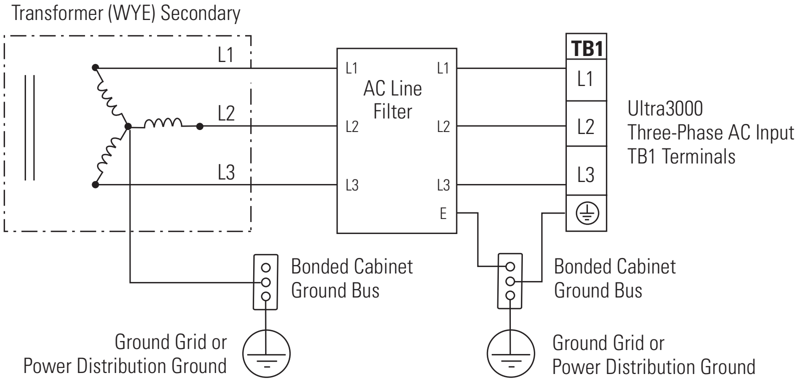
The grounded power configuration allows you to ground your single-phase or three-phase power at a neutral point. Match your secondary to one of the examples and be certain to include the grounded neutral connection.

IMPORTANT Grounded power (WYE secondary) is the preferred configuration. Examples with delta secondaries (though not preferred) are also shown.

Three-Phase Power Wired to Three-Phase Drives

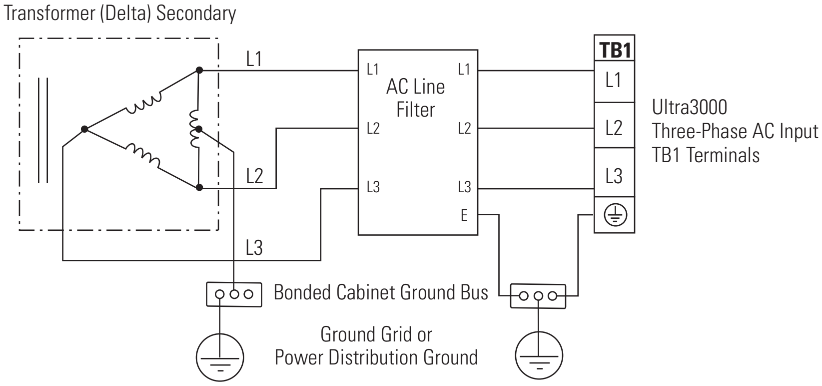
The following examples illustrate grounded three-phase power wired to three-phase Ultra3000 drives when phase-to-phase voltage is within drive specifications.

Figure 3.1
Three-Phase Power Configuration (WYE Secondary)



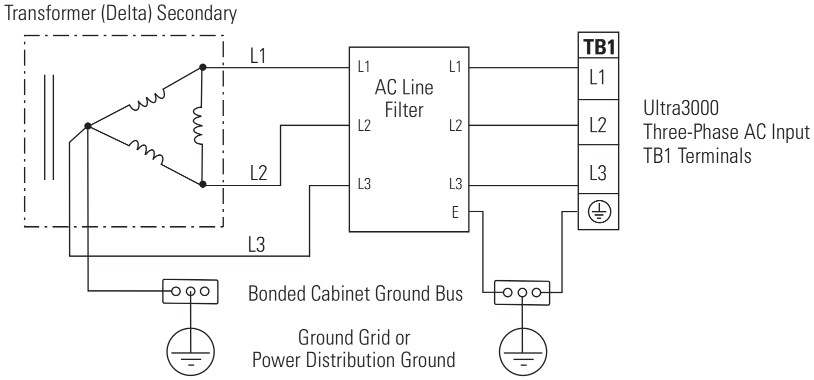
Note: Feeder and branch short circuit protection is not illustrated.

Figure 3.2
Three-Phase Power Configuration (Preferred Delta Secondary)



Note: Feeder and branch short circuit protection is not illustrated.

Figure 3.3
Three-Phase Power Configuration (Tolerated Delta Secondary)

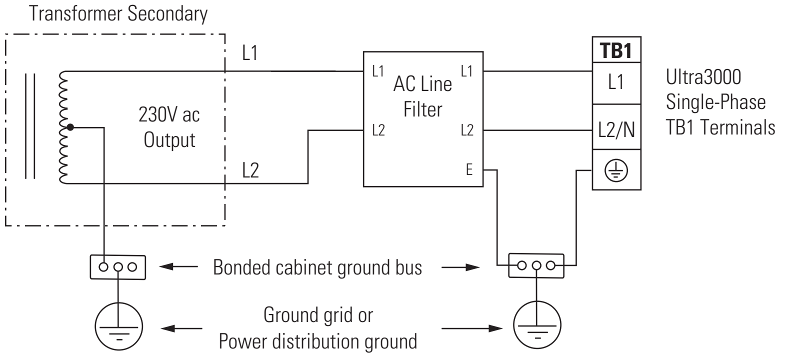


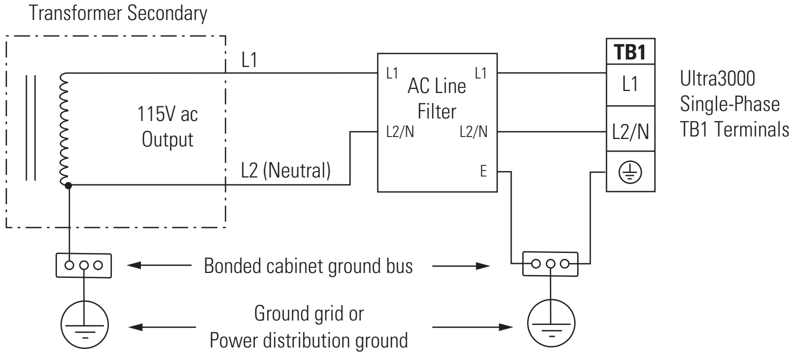
Note: Feeder and branch short circuit protection is not illustrated.

Single-Phase Power Wired to Single-Phase Drives

The following examples illustrate grounded single-phase power wired to single-phase Ultra3000 drives when phase-to-phase voltage is within drive specifications.

Figure 3.4
Single-Phase Grounded Power Configurations





Note: Reducing transformer output will reduce motor speed.
Note: Feeder and branch short circuit protection is not illustrated.

Isolation Transformer in Grounded Power Configurations

When using an isolation transformer, attach a chassis ground wire to the neutral connection. This accomplishes the following:

- Prevents the system from floating and thereby avoids any high voltages that might otherwise occur (e.g., through static).
- Provides a solid earth path for fault conditions.

ATTENTION

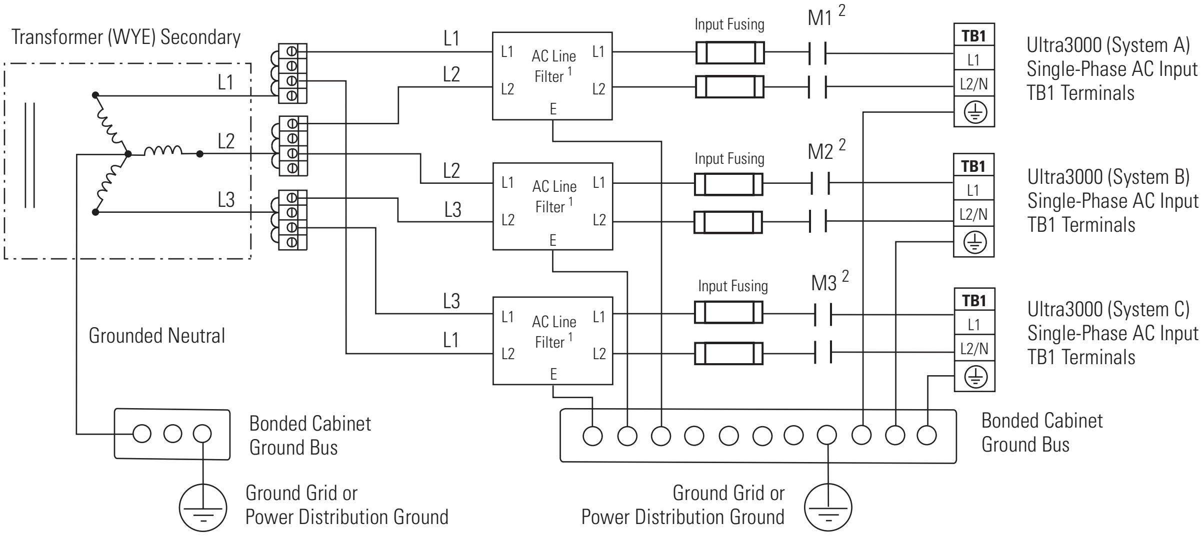
If the supply transformer is an auto transformer (not recommended), a chassis earth ground should not be added.

A chassis earth ground should already be included elsewhere in the system, and adding another would create a short.

Three-Phase Power Wired to Single-Phase Drives

The following examples (figures 3.5 and 3.6) illustrate grounded three-phase power wired to single-phase Ultra3000 drives when phase-to-phase voltage is *within* drive specifications.

Figure 3.5
Single-Phase Amplifiers on Three-Phase Power (WYE)

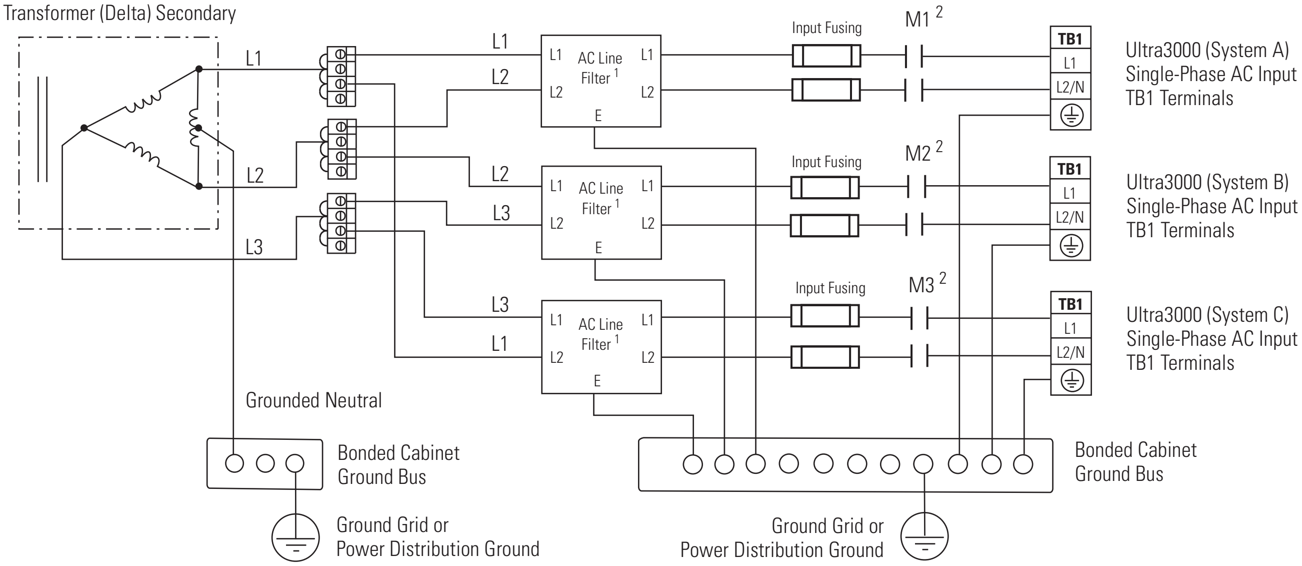


[1] AC line filter is optional, but is required for CE compliance.

[2] Contactors (MI, M2, and M3) may be optional. For more information refer to *Understanding the Machinery Directive* (publication SHB-900).

Note: Feeder short circuit protection is not illustrated.

Figure 3.6
Single-Phase Amplifiers on Three-Phase Power (Delta)



[1] AC line filter is optional, but is required for CE compliance.

[2] Contactors (MI, M2, and M3) may be optional. For more information refer to *Understanding the Machinery Directive* (publication SHB-900).

Note: Feeder short circuit protection is not illustrated.

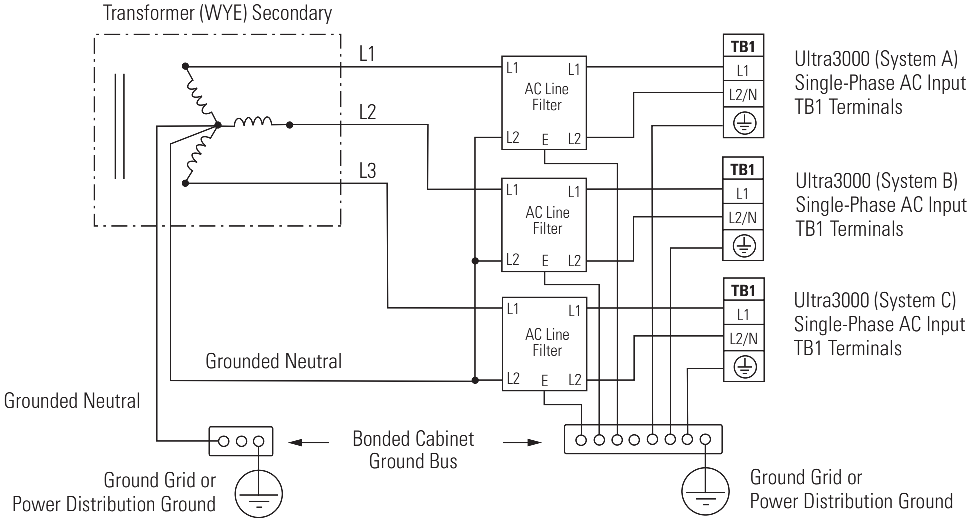
The following examples (figures 3.7 and 3.8) illustrate grounded three-phase power wired to single-phase Ultra3000 drives when phase-to-phase voltage *exceeds* drive specifications.

A neutral must be connected when single-phase drives are attached to a three-phase isolating transformer secondary. It is not necessary that all three-phases be loaded with drives, but each drive must have its power return via the neutral connection.

ATTENTION

Failure to connect the neutral can result in supply voltage swings at the individual loads (i.e., drives). This occurs when the neutral point moves vectorially as a result of load variations normally experienced by the individual drives. The supply voltage swing may cause undervoltage and overvoltage trips on the drives, and the drive can be damaged if the overvoltage limit is exceeded.

Figure 3.7
Single-Phase Amplifiers (One EMC Filter/Drive)



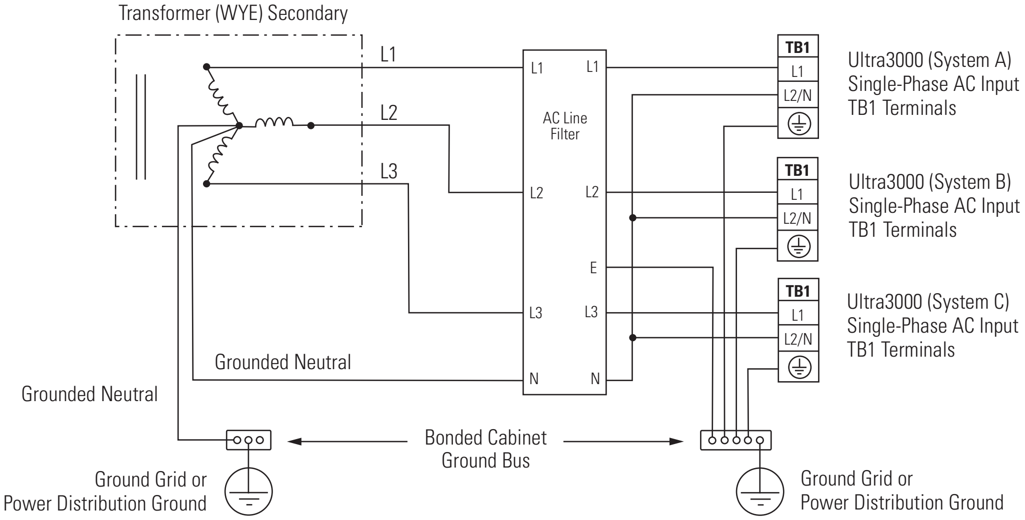
Note: Feeder and branch short circuit protection is not illustrated.

IMPORTANT

Providing an EMC line filter for each drive is the preferred configuration, and required for CE compliance.

If a three-phase line filter is used to feed multiple single-phase drives (not recommended), it is important that the filter include a neutral connection as shown in Figure 3.8. This applies if three-phase is brought directly into the filter (i.e., no isolating transformer present).

Figure 3.8
Single-Phase Amplifiers (One EMC Filter/Multiple Drives)



Note: Feeder and branch short circuit protection is not illustrated.

Voiding of CE Compliance

The three-phase and neutral in-line filter applications described above may not be adequate from an EMC aspect for CE compliance. Therefore, EMC validity and CE marking by Rockwell Automation is voided when three-phase and neutral in line filters are used.

ATTENTION

The three-phase isolation transformer and neutral in-line filter applications described in this document have not been tested for EMC by Rockwell Automation, and products used in such installations are not considered CE marked by Rockwell Automation.

If this three-phase isolation transformer and neutral in-line filter application is used, the responsibility for EMC validation lies with the user and CE marking of the system becomes the user's responsibility.

If CE compliance is a customer requirement, single-phase line filters which have been tested by Rockwell and specified for the product should be used. Refer to *AC Line Filters* on page C-3 for catalog numbers.

Grounding Your Ultra3000

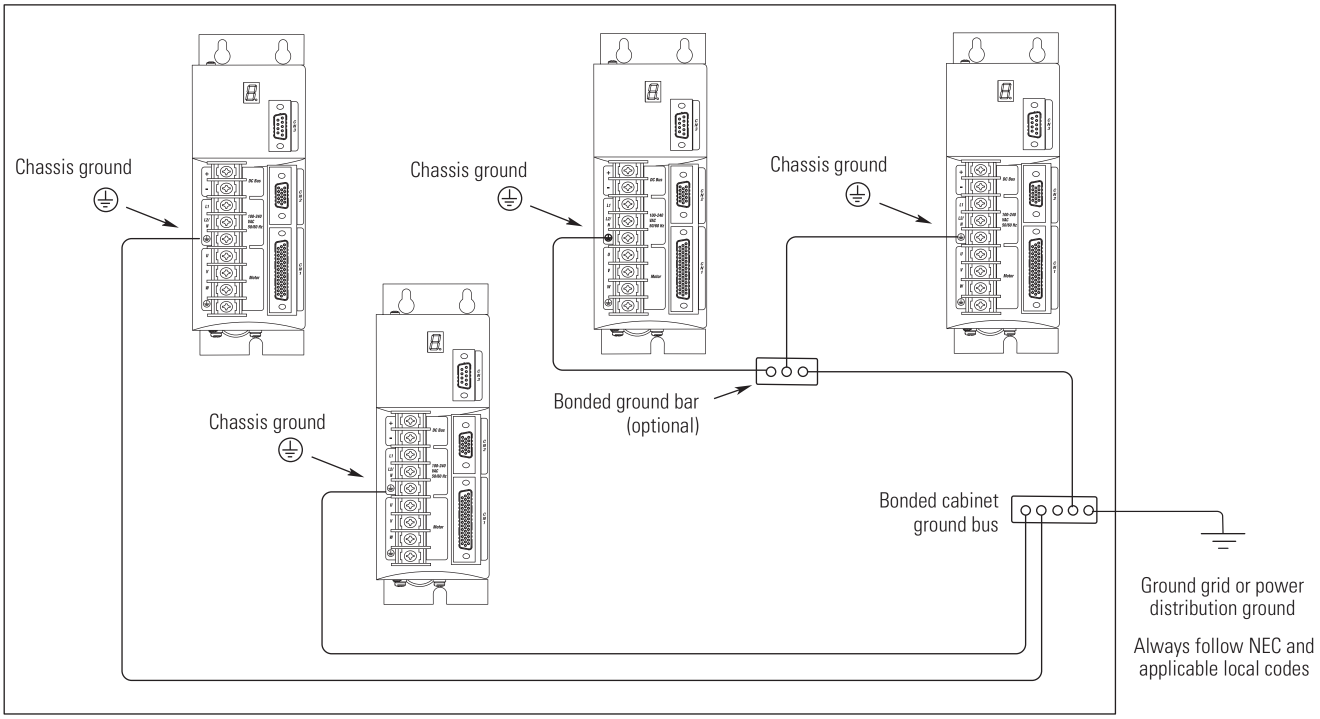
All equipment and components of a machine or process system should have a common earth ground point connected to their chassis. A grounded system provides a safety ground path for short circuit protection. Grounding your modules and panels minimize shock hazard to personnel and damage to equipment caused by short circuits, transient overvoltages, and accidental connection of energized conductors to the equipment chassis. For CE grounding requirements, refer to *Meeting CE Requirements* in *Chapter 1*.

IMPORTANT To improve the bond between the Ultra3000 and subpanel, construct your subpanel out of zinc plated (paint-free) steel.

Grounding Your System to the Subpanel

ATTENTION The National Electrical Code contains grounding requirements, conventions, and definitions. Follow all applicable local codes and regulations to safely ground your system. Refer to the illustration below for details on grounding your Ultra3000. Refer to *Appendix B* for the power wiring diagram for your Ultra3000 drive.

Figure 3.9
Chassis Ground Configuration (Multiple Ultra3000 Systems on One Panel)

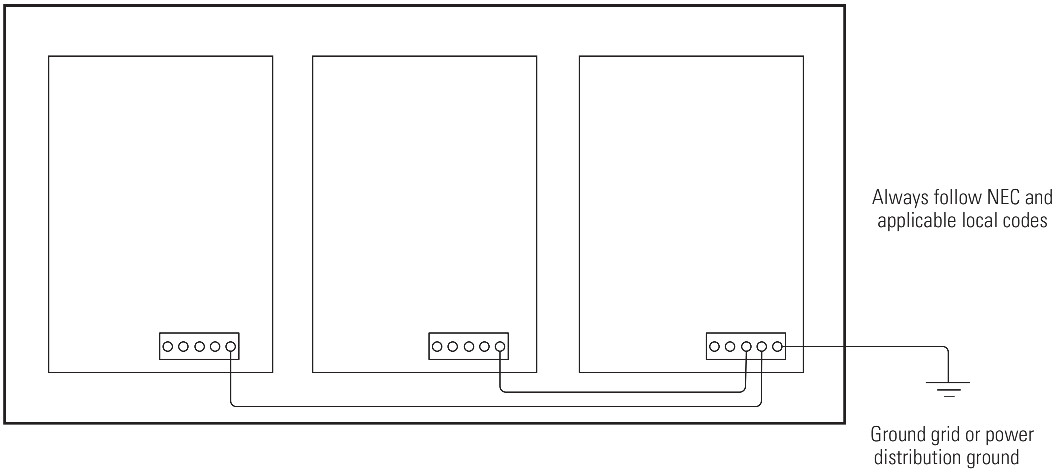


Grounding Multiple Subpanels

To ground multiple subpanels, refer to the figure below.

Note: HF bonding is not illustrated. For HF bonding information, refer to *Bonding Multiple Subpanels* on page 1-12.

Figure 3.10
Subpanels Connected to a Single Ground Point



Motor Power Cable Shield Termination

Factory supplied motor power cables for MP-Series, 1326AB, and F-, H-, N-, and Y-Series motors are shielded, and the power cable is designed to be terminated at the drive during installation. A small portion of the cable jacket is removed which exposes the shield braid. The exposed area must be clamped to the bottom of the drive chassis (refer to Figure 3.11) or the front of the drive chassis (refer to Figure 3.12) using the clamp provided.

ATTENTION

To avoid hazard of electrical shock, ensure shielded power cables are grounded at a minimum of one point for safety.

Connecting Cable Shields at the Drive

All motor power cable shields require attachment to the clamp as shown in the figures below.

Figure 3.11
Motor Power Cable Shield Connection (bottom of drive)

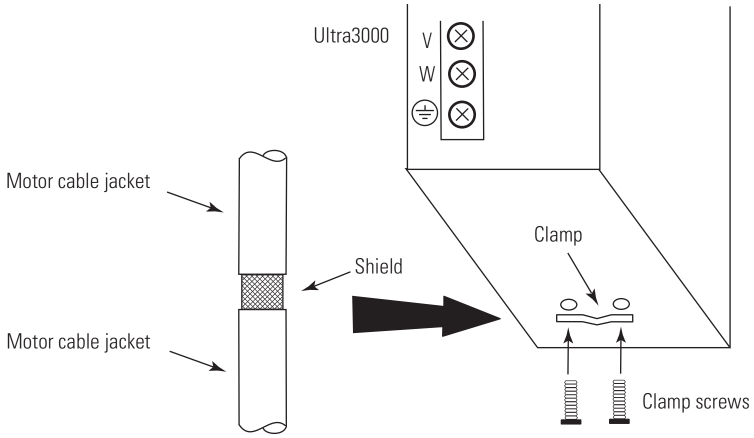
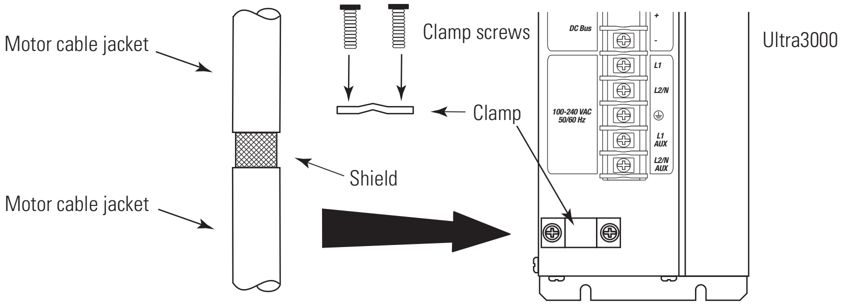


Figure 3.12
Motor Power Cable Shield Connection (front of drive)

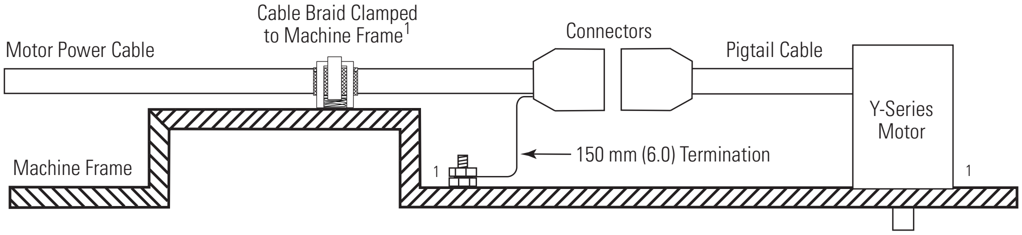


IMPORTANT The MP-Series Food Grade motors have shielded brake wires included in the motor power cable (2090-XXNPMF-*xx*S*xx*). Fold the brake wire shields back under the drive clamp before making connections to the brake circuit.

Connecting the Y-Series Cable Shield at the Motor

Y-Series motors have a short pigtail cable which connects to the motor, but is not shielded. The preferred method for grounding the Y-Series motor power cable on the motor side is to expose a section of the cable shield and clamp it directly to the machine frame. The motor power cable also has a 150 mm (6.0 in.) shield termination wire with a ring lug that connects to the closest earth ground. Use this method in addition to the cable clamp. The termination wire may be extended to the full length of the motor pigtail if necessary, but it is best to connect the supplied wire directly to ground without lengthening. Refer to Figure 3.13 for an illustration and Figure B.15 in *Appendix B* for the interconnect diagram.

Figure 3.13
Y-Series Motor Power Cable Connection



[1] Remove paint from machine frame to ensure proper HF-bond between machine frame and motor case, shield clamp, ground stud, etc.

Power Wiring Requirements

Power wiring requirements are given in the tables below. Wire should be copper with 75° C (167° F) minimum rating, unless otherwise noted. Phasing of main AC power is arbitrary and earth ground connection is required for safe and proper operation.

IMPORTANT The National Electrical Code and local electrical codes take precedence over the values and methods provided.

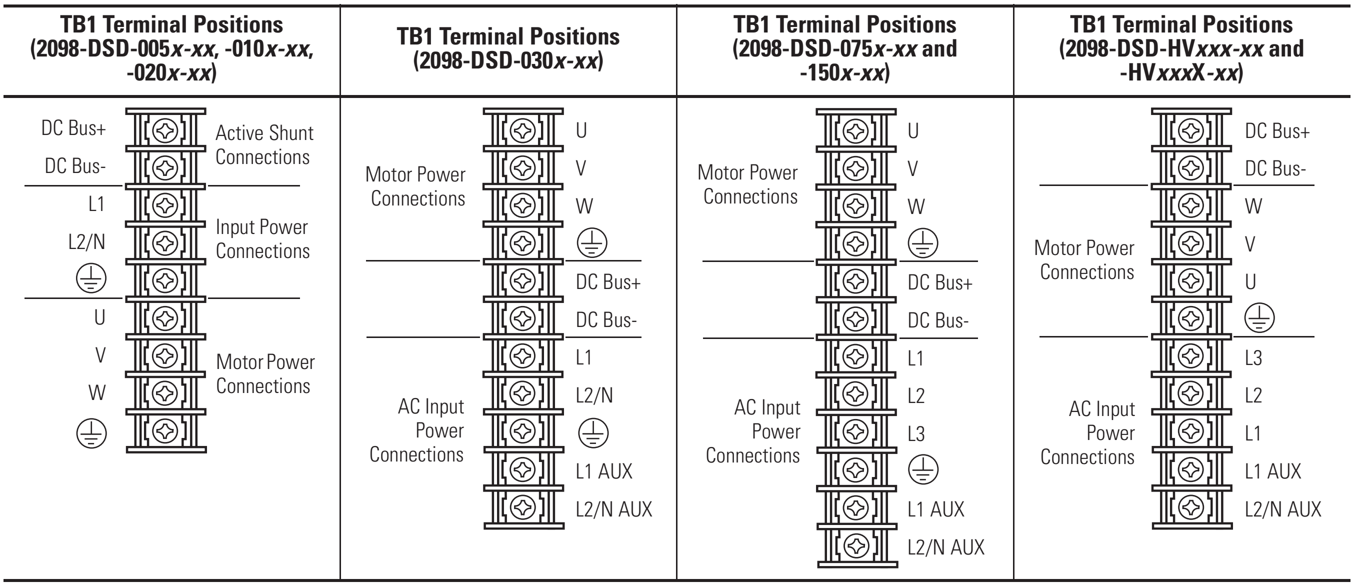
Ultra3000 Power Wiring Requirements

Ultra3000 Drives	Description	Connects to Terminals TB1	Recommended Wire Size mm^2 (AWG)	Terminal Block Torque Values Nm (lb-in.)
2098-DSD-005*x-xx*	Input Power [1] 100-240V ac single-phase	L1, L2/N, and ⏚	1.5 (16)	1.25 (11)
2098-DSD-010*x-xx*			2.5 (14)	
2098-DSD-020*x-xx*			3 (12)	
2098-DSD-030*x-xx*	Input Power [1] 100-240V ac single-phase	L1, L2/N, ⏚ L1 AUX, and L2/N AUX	6 (10)	
2098-DSD-075*x-xx*	Input Power [1] 100-240V ac three-phase	L1, L2, L3, ⏚ L1 AUX, and L2/N AUX		
2098-DSD-150*x-xx*			10 (8)	4.0 (35)
2098-DSD-HV030*x-xx*	Input Power [1] 230-480V ac three-phase	⏚ L3, L2, L1 L1 AUX, and L2/N AUX	2.5 (14)	1.25 (11)
2098-DSD-HV050*x-xx*				
2098-DSD-HV100*x-xx*			4 (12)	
2098-DSD-HV150*x-xx*			6 (10)	
2098-DSD-HV220*x-xx*			10 (8)	
All Ultra3000 drives (230V and 460V)	Motor Power	U, V, W, and ⏚	Motor power cable depends on motor/ drive combination.	

[1] The input power may be optionally isolated through a transformer.

Note: Refer to Figure 3.14 for TB1 terminal positions.

Figure 3.14
TB1 Terminal Positions



Shunt Module Power Wiring Requirements

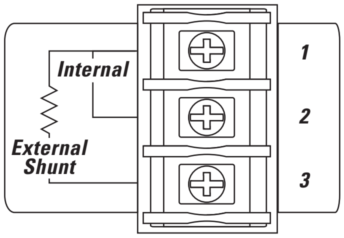
Ultra3000 Drives	Shunt Module Catalog Number	Connects to Terminals	Recommended Wire Size mm² (AWG)	Terminal Block Torque Values Nm (lb-in.)
2098-DSD-005*x-xx*, -010*x-xx*, and -020*x-xx*	2090-UCSR-A300	TB1-DC Bus+ TB1-DC Bus- TB1-L1 TB1-L2/N ⏚	4.0 (12)[1]	1.25 (11)
2098-DSD-030*x-xx*	9101-1183	TB2-1 TB2-2 TB2-3	2.5 (14)[1]	
2098-DSD-075*x-xx* and 2098-DSD-150*x-xx*	2090-UCSR-P900		10 (8)[1]	
2098-DSD-HV030-*xx*, -HV030X-*xx* -DSD-HV050-*xx*, and -HV050X-*xx*	2090-SR120-09		6 (10)[2]	
2098-DSD-HV100-*xx* and -HV100X-*xx*	2090-SR040-09 or 2090-SR040-18			
2098-DSD-HV150-*xx* and -HV150X-*xx*	2090-SR025-09 or 2090-SR025-18			
2098-DSD-HV220-*xx* and -HV220X-*xx*	2090-SR020-36			

[1] 75° C (167° F), 600V copper wire. Maximum length 3.05 m (10 ft).

[2] 105° C (221° F), 600V copper wire (preferred) 90° C (194° F) (minimum). Maximum length 3.05 m (10 ft).

Note: Refer to Figure 3.15 for TB2 terminal positions.

Figure 3.15
TB2 Terminal Positions



Refer to *Shunt Module Interconnect Diagrams* in *Appendix B* for an example with your Ultra3000 drive.

The Ultra3000 utilizes solid state motor overload protection which operates in accordance with UL 508C.

Motor overload protection trips:	At:
Eventually	100% overload.
Within 8 minutes	200% overload.
Within 20 seconds	600% overload.

IMPORTANT Do not connect an external I/O power supply to the DC bus. The DC+ and DC- terminals connect directly to the power bus of the drive.

ATTENTION This drive contains ESD (Electrostatic Discharge) sensitive parts and assemblies. You are required to follow static control precautions when you install, test, service, or repair this assembly. If you do not follow ESD control procedures, components can be damaged. If you are not familiar with static control procedures, refer to Allen-Bradley publication 8000-4.5.2, *Guarding Against Electrostatic Damage* or any other applicable ESD Protection Handbook.

ATTENTION

To avoid personal injury and/or equipment damage, ensure installation complies with specifications regarding wire types, conductor sizes, branch circuit protection, and disconnect devices. The National Electrical Code (NEC) and local codes outline provisions for safely installing electrical equipment.

To avoid personal injury and/or equipment damage, ensure motor power connectors are used for connection purposes only. Do not use them to turn the unit on and off.

To avoid personal injury and/or equipment damage, ensure shielded power cables are grounded to prevent potentially high voltages on the shield.

Connecting Input Power

This procedure assumes you have mounted your Ultra3000 drive and are ready to wire your AC input power.

IMPORTANT When tightening screws to secure the wires, refer to the table on page 3-13 for torque values.

IMPORTANT To ensure system performance, run wires and cables in the wireways as established in *Chapter 1*.

Refer to *Appendix B* for the power wiring diagram for your Ultra3000.

To wire your input power:

1. Prepare your wires by stripping approximately 12 mm (0.50 in.) of insulation from the end.

   IMPORTANT Use caution not to nick, cut, or otherwise damage strands as you remove the insulation.

2. Route the power cable to your Ultra3000 drive.

3. Locate the TB1 terminal block and remove the plastic cover. Refer to *Understanding Ultra3000 Connectors* in *Chapter 2* for the front panel connections of your Ultra3000 drive.

IMPORTANT The auxiliary AC power inputs require dual element time delay (slow acting) fuses to accommodate inrush current. Refer to the section *Ultra3000 (230V) Power Specifications* in *Appendix A* for the inrush current on the auxiliary AC power input.

4. Using a screw driver, loosen the screw for each of the terminal locations and attach wires as shown in the table below.

If you have this drive:	Insert this wire from the power supply:	Into this terminal on TB1:
2098-DSD-005*x-xx*, -010*x-xx*, or -020*x-xx*	L1	L1
	L2	L2/N
	Ground	⏚
2098-DSD-030*x-xx*	L1	L1
	L2	L2/N
	Ground	⏚
	L1 (auxiliary AC)	L1 AUX
	L2 (auxiliary AC)	L2/N AUX
2098-DSD-075*x-xx,* -150*x-xx* or 2098-DSD-HV*xxx-xx,* -HV*xxx*X-*xx*	L1	L1
	L2	L2
	L3	L3
	Ground	⏚
	L1 (auxiliary AC)	L1 AUX
	L2 (auxiliary AC)	L2/N AUX

IMPORTANT The DC bus connections should not be used for connecting multiple drives together. Contact your Allen-Bradley representative for further assistance if the application may require DC power connections.

5. Tighten each terminal screw. Refer to the table on page 3-13 for torque value.

6. Gently pull on each wire to make sure it does not come out of its terminal. Re-insert and tighten any loose wires.

7. Re-attach the plastic cover to the terminal block.

Connecting Motor Power and Brakes

This procedure assumes you have wired your input power and are ready to wire the motor power and brake connections.

IMPORTANT When tightening screws to secure the wires, refer to the table on page 3-13 for torque values.

IMPORTANT To ensure system performance, run wires and cables in the wireways as established in *Chapter 1*.

Refer to *Appendix B* for the power wiring diagram for your Ultra3000.

Applying the Motor Cable Shield Clamp

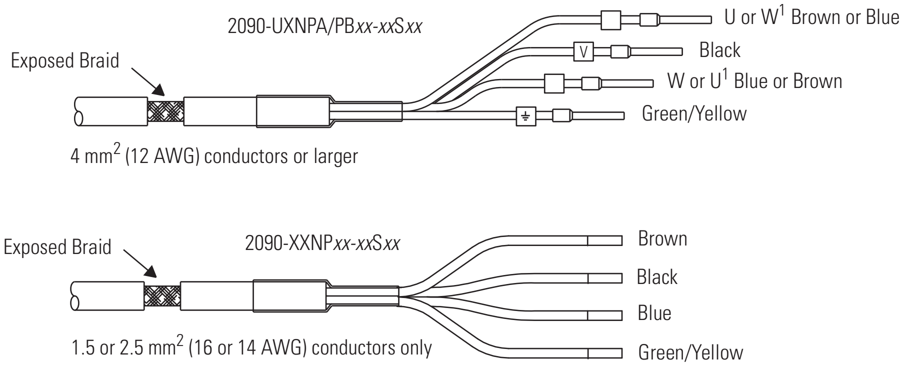
To apply the motor power cable shield clamp:

1. Loosen the two screws securing the cable clamp on the Ultra3000 drive (refer to figures 3.11 or 3.12 for the cable clamp location on your Ultra3000 drive).
2. Feed the cable into the clamp and position the exposed portion of the cable braid directly in line with the clamp.
3. Extend the cable along side the drive leaving an appropriate length of cable for terminating the motor power wires.
4. Check that the exposed cable braid is still in line with the clamp (adjust alignment if necessary).
5. Tighten the screws with a torque of 0.9-1.1 Nm (8.0-10.0 lb-in.).

Wiring Motor Power

When using MP-Series (low inertia and integrated gear), 1326AB, and F-, H-, or N-Series motors refer to Figure 3.16 for your motor power cable configuration. Refer to *Appendix B* for the motor/drive interconnect diagrams.

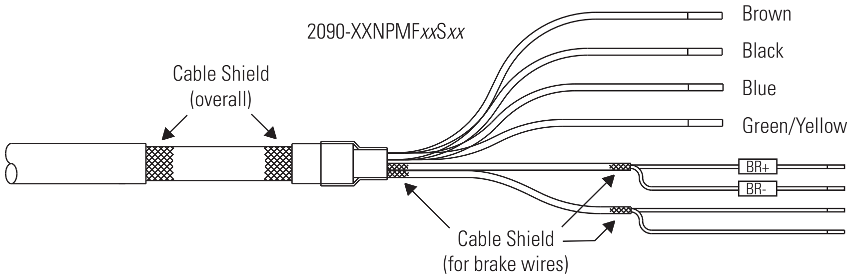
Figure 3.16
Motor Power Cable (MPL-A/B, MPG-A/B, 1326AB, and F-, H-, or N-Series Motors)



[1] Motor cable leads (shortest to longest) are labeled differently, depending on the drive input voltage (230V/460V).

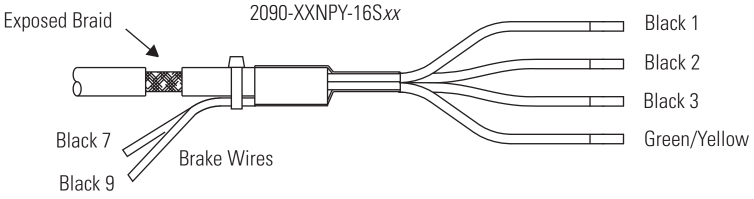
When using MP-Series (food grade) motors, the brake wires are included in the motor power cable (2090-XXNPMF-*xx*S*xx*), as shown in Figure 3.17. Refer to *Appendix B* for the motor/drive interconnect diagram.

Figure 3.17
Motor Power Cable (MPF-A/B Food Grade Motors)



When using Y-Series motors, the brake wires are included in the motor power cable (2090-XXNPY-16S*xx*), as shown in Figure 3.18. Refer to *Appendix B* for the motor/drive interconnect diagram.

Figure 3.18
Motor Power Cable (Y-Series Motors)



Refer to the table below for the catalog number of the motor power cable for your Ultra3000 drive.

For this motor:	This Universal Cable is available:	Use this Motor Power Cable when Universal Cables are not available:
MP-Series (Low Inertia and Integrated Gear) (230V or 460V systems)	2090-XXNPMP-16S*xx* 2090-XXNPMP-14S*xx* 2090-XXNPMP-10S*xx* 2090-XXNPMP-8S*xx*	N/A
1326AB-Series (M2L/S2L)		
MP-Series (Food Grade)	2090-XXNPMF-16S*xx* 2090-XXNPMF-14S*xx* 2090-XXNPMF-10S*xx*	
F-Series	2090-XXNPHF-16S*xx* 2090-XXNPHF-14S*xx*	2090-UXNPAHF-10S*xx* 2090-UXNPAHF-8S*xx*
H-Series	2090-XXNPH-16S*xx* 2090-XXNPHF-14S*xx*	2090-UXNPAHF-10S*xx* 2090-UXNPAHF-8S*xx* 2090-UXNPAH-6S*xx*
N-Series	2090-XXNPN-16S*xx*	N/A
Y-Series	2090-XXNPY-16S*xx*	N/A

To wire your motor power:

1. Route the motor power cable to your Ultra3000 drive.

IMPORTANT To ensure system performance, run wires and cables in the wireways as established in *Chapter 1*.

2. Using a screw driver, loosen the screw for each of the terminal locations and attach wires as shown in the table below.

Insert the motor power wires from this servo motor:		**Into this terminal on TB1:**
1326AB- (M2L/S2L), F-, H-, N-, and MP-Series	**Y-Series**	
U / Brown	1 / Black	U
V / Black	2 / Black	V
W / Blue	3 / Black	W
⏚ Green/Yellow	⏚ Green/Yellow	⏚

IMPORTANT Ensure motor power is wired with proper phasing relative to the motor terminals. On some motors, the motor leads may be labeled R, S, and T which correspond to U, V, and W respectively.

3. Tighten each terminal screw. Refer to the table on page 3-13 for torque values.

4. Gently pull on each wire to make sure it does not come out of its terminal. Re-insert and tighten any loose wires.

5.

If your motor is:	Then:
Y-Series	1. Connect the 152.4 mm (6.0 in.) termination wire at the motor end of the cable to the closest earth ground (refer to Figure 3.13 for an illustration). 2. Go to *Understanding Motor Brake Connections*.
Not Y-Series	Go to *Understanding Motor Brake Connections*.

Understanding Motor Brake Connections

The procedure for wiring your motor brake varies slightly, depending on the motor series you are using. Refer to the table below to determine where the brake wires for your servo motor are located and for the appropriate brake cable or connector kit catalog number.

For this Motor Series:	The Brake Wires are:	Cable Catalog Number:
MP-Series (low inertia and integrated gear)	In a separate brake cable (the motor has a brake connector)	2090-UXNBMP-18S*xx* brake cable
1326AB-Series (M2L/S2L)		
F-Series		Cable connector kit 9101-0330
H-Series		
N-Series		Cable connector kit 9101-1698
MP-Series (food grade)	Included in the power cable (the motor does not have a brake connector)	2090-XXNPMF-*xx*S*xx* power cable
Y-Series		2090-XXNPY-16S*xx* power cable

To wire your motor brakes refer to *Ultra3000/Motor Interconnect Diagrams* in *Appendix B* for the interconnect diagram showing the brake connections for your Ultra3000 drive and servo motor.

Understanding Shunt Connections

Follow these guidelines when installing and wiring your active or passive shunt module/resistor.

IMPORTANT When tightening screws to secure the wires, refer to the table on page 3-13 for torque values.

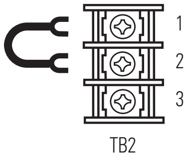
IMPORTANT To ensure system performance, run wires and cables in the wireways as established in *Chapter 1*.

Refer to *Appendix B* for the Ultra3000 interconnect diagrams.

If your application requires an:	And you are wiring to this Ultra3000 drive:	Then refer to:
External Shunt	2098-DSD-005*x*-*xx*, 2098-DSD-010*x*-*xx*, or 2098-DSD-020*x*-*xx*	• *Planning Your Panel Layout* in *Chapter 1*. • Figure B.5 on page B-7 of *Appendix B*. • The installation instructions provided with your shunt (publication 2090-IN002*x*-EN-P).
	2098-DSD-030*x*-*xx*	• *Planning Your Panel Layout* in *Chapter 1*. • Figure B.7 on page B-8 of *Appendix B*. • The installation instructions provided with your shunt (publication 2090-IN003*x*-EN-P).
	2098-DSD-075*x*-*xx*, or 2098-DSD-150*x*-*xx*	• *Planning Your Panel Layout* in *Chapter 1*. • Figures B.7 and B.8 on page B-8 of *Appendix B*. • The installation instructions provided with your shunt (publication 2090-IN001*x*-EN-P).
	2098-DSD-HV030-*xx*, -HV050-*xx*, or -HV100-*xx*	• *Planning Your Panel Layout* in *Chapter 1*. • Figure B.7 on page B-8 of *Appendix B*. • The installation instructions provided with your shunt (publication 2090-IN004*x*-EN-P).
	2098-DSD-HV150-*xx*, or -HV220-*xx*	• *Planning Your Panel Layout* in *Chapter 1*. • Figure B.9 on page B-9 of *Appendix B*. • The installation instructions provided with your shunt.
Internal Shunt	2098-DSD-030*x*-*xx*, 2098-DSD-075*x*-*xx*, 2098-DSD-150*x*-*xx*, 2098-DSD-HV*xxx*-*xx*, or 2098-DSD-HV*xxx*X-*xx*	Verify the TB2 internal shunt jumper is in place between TB2-1 and TB2-2, as shown in Figure 3.19 below.

Figure 3.19
Connecting Your Shunt Resistor

Connecting the Internal Shunt Resistor[1]



---

[1] This is the factory default jumper setting for TB2.

Understanding Feedback and I/O Cable Connections

Factory made cables with premolded connectors are designed to minimize EMI and are recommended over hand-built cables to improve system performance. However, other options are available for building your own feedback and I/O cables. Refer to the table below for the available options.

Drive Connector	**Connector Option**		**Option Catalog Number**	**Reference**
CN1 I/O Connector	44-pin drive-mounted breakout board with 24V to 5V auxiliary power converter		2090-U3CBB-DM44	• *CN1 Control Interface Breakout Boards with 24V to 5V Auxiliary Power Converter* (publication 2090-IN008*x*-EN-P). • *Understanding Ultra3000 I/O Specifications* beginning on page 2-26.
	12-pin drive-mounted breakout board with 24V to 5V auxiliary power converter for SERCOS interface applications		2090-U3CBB-DM12	
	44-pin panel-mounted breakout board kit		2090-U3BK-D44*xx*	*Understanding Ultra3000 I/O Specifications* beginning on page 2-26.
	44-pin, drive-mounted breakout board.[1]		2090-U3BB2-DM44	
	44-pin (high-density D-shell) drive connector kit		2090-U3CK-D44	
	Single-axis flying lead to 1756-M02AE module or 1784-PM02AE PCI card		2090-U3CC-D44*xx*	
	Two-axis pre-wired to 1756-M02AE module		2090-U3AE-D44*xx*	• *Understanding Ultra3000 I/O Specifications* on page B-20. • *Understanding Ultra3000 I/O Specifications* beginning on page 2-26.
CN2 Feedback Connector	Premolded cable at drive and motor end		2090-UXNFB*xx*-S*xx*	• The table below for your motor feedback cable catalog number. • *Motor Feedback Connector Pin-outs* on page 3-24. • *Understanding Motor Encoder Feedback Specifications* beginning on page 2-40.
	Flying lead cable at drive end (2090-XXNF*xx*-S*xx*)	15-pin drive-mounted breakout board	2090-UXBB-DM15	
		15-pin panel-mounted breakout board kit	2090-UXBK-DM15*xx*	
		15-pin (high-density D-shell) drive connector kit	2090-UXCK-D15	
CN3 Serial Connector	PC serial connector to premolded drive connector		2090-UXPC-DM09	*Understanding the Serial Interface* beginning on page 2-51.
	9-pin drive-mounted breakout board		2090-UXBB-DM09	
	9-pin (high-density D-shell) drive connector kit		2090-UXCK-D09	

---

[1] This breakout board accepts 1 - 0.14 mm^2 (16 - 26 AWG) wire. For applications that require a 44-pin drive-mounted breakout board that accepts 4 - 0.5 mm^2 (12 - 22 AWG) wire, contact your local Allen-Bradley representative.

Refer to the table below for motor feedback cable catalog numbers available for specific motor/feedback combinations.

For this Motor Series:	Using this Type of Motor Feedback:	Use this Feedback Cable Premolded:	Use this Feedback Cable Flying Lead:
MPL-A*xxxx* or MPG-A*xxx-xxx*	High-resolution encoder	2090-UXNFBMP-S*xx*	2090-XXNFMP-S*xx* [1]
MPL-A*xxxx*	Incremental encoder		
MPL-B*xxxx* or MPG-B*xxx-xxx*	High-resolution encoder		
1326AB	M2L/S2L High-resolution encoder		
MPF-A*xxxx* or MPF-B*xxxx*	High-resolution encoder	N/A	2090-XXNFMF-S*xx* [1]
F-Series	Incremental encoder	2090-UXNFBHF-S*xx*	2090-XXNFHF-S*xx* [1]
H-Series			
N-Series		2090-UXNFBN-S*xx*	2090-XXNFN-S*xx* [1]
Y-Series		2090-UXNFBY-S*xx*	2090-XXNFY-S*xx* [1]

[1] Requires 2090-UXBB-DM15 drive-mounted breakout board, 2090-UXBK-D15xx breakout board kit, or 2090-UXCK-D15 mating connector kit.

Note: Refer to *Maximum Feedback Cable Lengths* on page A-13 to determine the maximum length of your motor feedback cable.

Motor Feedback Connector Pin-outs

The following tables provide the signal descriptions and pin-outs for the motor feedback (CN2) 15-pin connector to MP-Series (low inertia and integrated gear), 1326AB, and N-Series motors.

Motor Connector Pin	High Resolution Feedback Signals for: MPL-B*xxx*-M/-S MPL-A5*xx*-M/-S and 1326AB-B*xxx*-M2L/-S2L Motors	High Resolution Feedback Signals for: MPL-A3*xx*-M/-S MPL-A4*xx*-M/-S MPL-A45*xx*-M/-S MPG-A/B*xxx*-M/-S Motors	Incremental Encoder Feedback Signals for: MPL-A*xxx*-H MPL-B*xxx*-H and N-Series Motors	Drive (CN2) Connector Pin
A	Sine+	Sine+	AM+	1
B	Sine-	Sine-	AM-	2
C	Cos+	Cos+	BM+	3
D	Cos-	Cos-	BM-	4
E	Data+	Data+	IM+	5
F	Data-	Data-	IM-	10
K	Reserved	EPWR_5V	EPWR_5V	14
L	Reserved	ECOM	ECOM	6
N	EPWR_9V	Reserved	Reserved	7
P	ECOM	Reserved	Reserved	6
R	TS+	TS+	TS+	11
S	TS-	TS-	TS-	6
T	Reserved	Reserved	S1/Hall A	12
U	Reserved	Reserved	S2/Hall B	13
V	Reserved	Reserved	S3/Hall C	8

The following tables provide the signal descriptions and pin-outs for the motor feedback (CN2) 15-pin connector to MP-Series food grade motors.

Motor Connector Pin	High Resolution Feedback Signals for:		Drive (CN2) Connector Pin
	MPF-B*xxx*-M/-S and MPF-A5*xx*-M/-S Motors	MPF-A3*xx*-M/-S MPF-A4*xx*-M/-S MPF-A45*xx*-M/-S Motors	
1	Sine+	Sine+	1
2	Sine-	Sine-	2
3	Cos+	Cos+	3
4	Cos-	Cos-	4
5	Data+	Data+	5
6	Data-	Data-	10
9	Reserved	EPWR_5V	14
10	Reserved	ECOM	6
11	EPWR_9V	Reserved	7
12	ECOM	Reserved	6
13	TS+	TS+	11
14	TS-	TS-	6

The following tables provide the signal descriptions and pin-outs for the motor feedback (CN2) 15-pin connector to F-, H-, and Y-Series (230V) motors.

Motor Connector Pin	Incremental Encoder Feedback Signals to F- and H-Series Motors	Drive Connector Pin
A	AM+	1
B	AM-	2
C	BM+	3
D	BM-	4
E	IM+	5
F	IM-	10
G	Reserved	–
H	Reserved	–
J	EPWR_5V	14
K	EPWR_5V	14
L	ECOM	6
M	ECOM	6
N	S2/Hall B	13
P	S3/Hall C	8
R	TS+	11
S	TS-	6
T	S1/Hall A	12

Motor Connector Pin	Incremental Encoder Feedback Signals to Y-Series Motors	Drive (CN2) Connector Pin
9	AM+	1
10	AM-	2
11	BM+	3
12	BM-	4
13	IM+	5
14	IM-	10
15	S1/Hall A	12
17	S2/Hall B	13
19	S3/Hall C	8
22	EPWR_5V	14
23	ECOM	6
24	Drain	Connector Housing
Reserved	Reserved	7
Reserved	Reserved	9
Reserved	Reserved	11
Reserved	Reserved	15

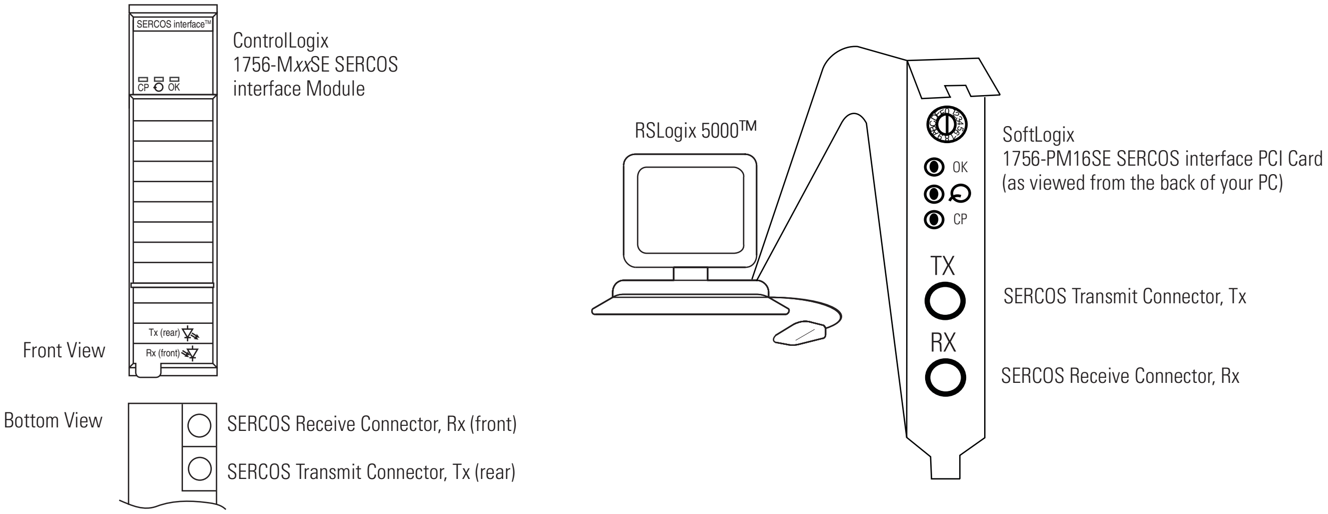
Connecting Your SERCOS Fiber-Optic Cables

This procedure assumes you have your ControlLogix chassis with 1756-M*xx*SE interface module or personal computer with 1784-PM16SE PCI card and Ultra3000 SERCOS interface system(s) mounted and are ready to connect the fiber-optic cables.

The SERCOS fiber-optic ring is connected using the SERCOS Receive and Transmit connectors. Refer to *Chapter 2* for the location of the connectors on your Ultra3000-SE drive(s) and Figure 3.20 to locate the connectors on your SERCOS interface module or PCI card.

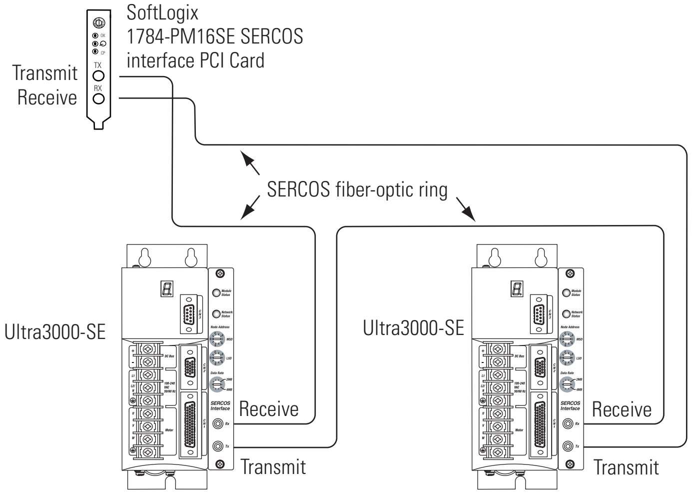
Note: Plastic cable is available in lengths up to 32 m (105.0 ft). Glass cable is available in lengths up to 200 m (656.7 ft).

Figure 3.20
ControlLogix and SoftLogix SERCOS Connector Locations



Refer to Figure 3.21 for an example of fiber-optic ring connections between the Ultra3000-SE drive(s) and the SoftLogix PCI card.

Figure 3.21
Fiber-Optic Ring Connection (Example 1)



Refer to figures 3.22, 3.23, and 3.24 for examples of fiber-optic ring connections between the Ultra3000-SE drive(s) and the ControlLogix SERCOS interface module.

Figure 3.22
Fiber-Optic Ring Connection (Example 2)

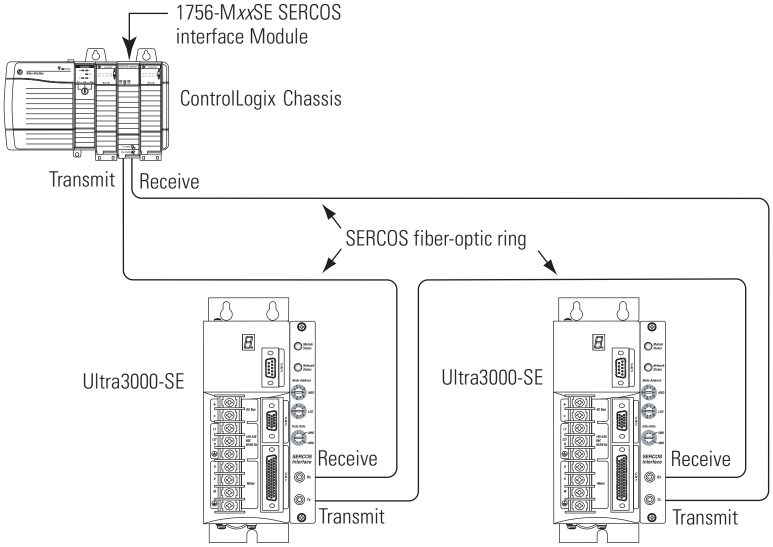
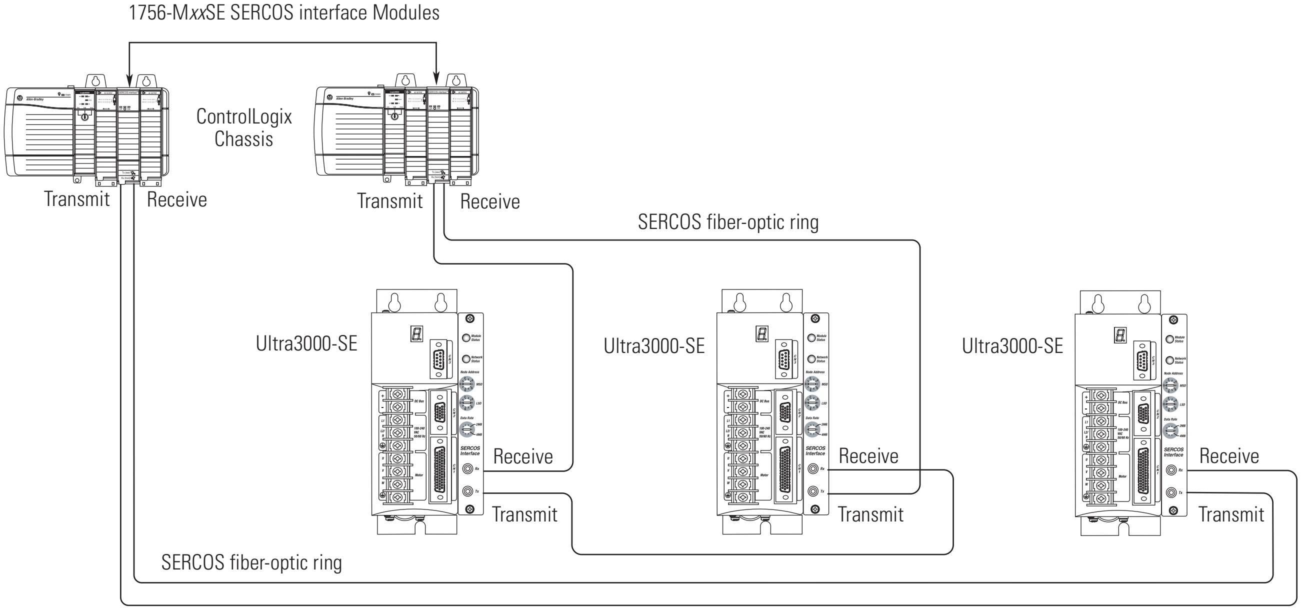


Figure 3.23
Fiber-Optic Ring Connection (Example 3)



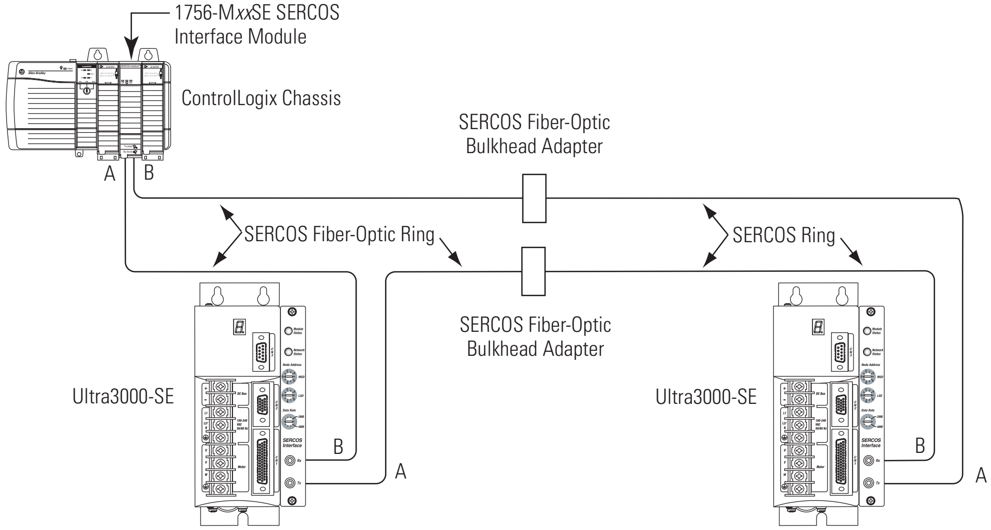
Note: You can mount the two 1756-M*xx*SE SERCOS interface modules in two separate ControlLogix chassis (as shown above) or you can mount them in the same chassis.

Cable lengths of 32 m (105 ft) for plastic cable and 200 m (656.7 ft) for glass cable are possible for each transmission section (point A to B). In Figure 3.24, the second Ultra3000-SE system is located in a separate cabinet and connected with bulkhead adapters.

IMPORTANT To avoid signal loss, do not mix glass and plastic cables when connecting to a bulkhead adapter. Use glass-to-glass or plastic-to-plastic cable on both sides of the adapter.

IMPORTANT Clean the fiber-optic cable connectors prior to installation. Dust in the connectors can reduce signal strength. For more information, refer to *Fiber Optic Cable Installation and Handling Instructions* (publication 2090-IN010*x*-EN-P).

Figure 3.24
Fiber-Optic Ring Connection (Example 4)



To connect the SERCOS fiber-optic cables:

1. Insert one end of a fiber-optic cable into the Receive SERCOS connector on the Ultra3000-SE and thread the connector on finger tight.

2. Insert the other end of the cable (from step 1) into the Transmit SERCOS connector on the ControlLogix module/SoftLogix PCI Card and thread the connector on finger tight.

3. Insert one end of another fiber-optic cable into the Transmit SERCOS connector on the last Ultra3000-SE drive in the ring and thread the connector on finger tight.

4. Insert the other end of the cable (from step 3) into the Receive SERCOS connector on the ControlLogix module/SoftLogix PCI Card and thread the connector on finger tight.

5. Complete the ring by connecting the Transmit and Receive connectors from one drive to the next until all are connected (refer to the examples above).

Refer to *Appendix C* for SERCOS fiber-optic cable and bulkhead adapter catalog numbers.

Connecting to a DeviceNet Network

A DeviceNet network is an arrangement of electrical power and device distribution. A DeviceNet network is planned and adjusted for optimal communications.

Before proceeding to add devices, you need to record the following:

- Network data rate
- Network cable system map (topology) to which you are connecting
- Distances between cable system components
- Device current draw and voltage drop for each device on the network
- Limitation of the trunk and drop cables

Refer to the table below for recommended trunk and drop lengths.

Data Rates	125 Kbps	250 Kbps	500 Kbps
Thick Trunk Line	500 m (1,640 ft)	250 m (820 ft)	100 m (328 ft)
Thin Trunk Lengths	100 m (328 ft)	100 m (328 ft)	100 m (328 ft)
Maximum Drop Length	6 m (20 ft)	6 m (20 ft)	6 m (20 ft)
Cumulative Drop Budget	156 m (512 ft)	78 m (256 ft)	39 m (128 ft)

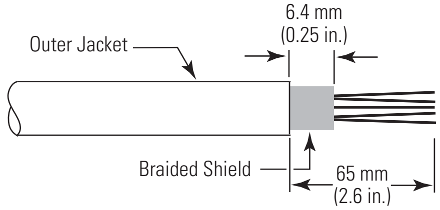
Refer to the *DeviceNet Cable System Planning and Installation Manual* (publication DN-6.7.2) for specific guidance in calculating and attaching the Ultra3000-DN to a network.

Connecting Your DeviceNet Cable

To wire the DeviceNet connector:

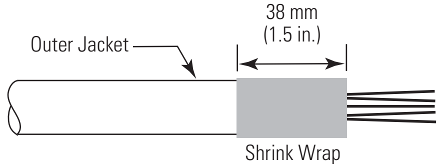
1. Strip 65 mm (2.6 in.) to 75 mm (2.96 in.) of the outer jacket from the end of the cable, leaving no more than 6.4 mm (0.25 in.) of the braided shield exposed.

Figure 3.25
Exposing the Braided Shield



2. Wrap the end of the cable with 38 mm (1.5 in.) of shrink wrap, covering part of the exposed wires and part of the outer jacket.

Figure 3.26
Adding Shrink Wrap

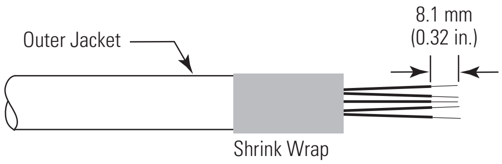


3. Strip 8.1 mm (0.32 in.) of the insulation from the end of each of the insulated wire.

IMPORTANT Use caution not to nick, cut, or otherwise damage strands as you remove the insulation.

Trim the last 6.4 mm (0.25 in.) of the bare wires so that the outside dimension does not exceed 0.17 mm (0.045 in.).

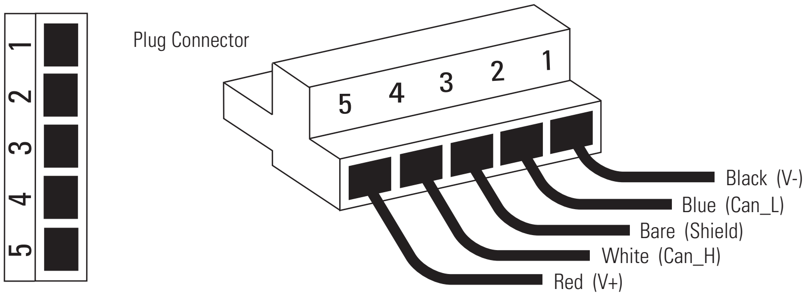
Figure 3.27
Exposing Wire Strands



4. Using a screwdriver, loosen the screw for each of the terminal locations (refer to Figure 3.28) and attach wires as shown in the table below.

Insert this wire:		Into this terminal on the DeviceNet connector:
Color	**Designation**	
Black	V-	1
Blue	Can_L	2
Bare	Shield	3
White	Can_H	4
Red	V+	5

Figure 3.28
Wiring the DeviceNet Connector



5. Tighten each terminal screw using a 1/8 inch flat blade screwdriver.

6. Gently pull on each wire to make sure it does not come out of its terminal. Re-insert and tighten any loose wires.

7. Plug the DeviceNet cable into the DeviceNet interface connector. Refer to *Ultra3000 (with DeviceNet) Front Panel Connections* beginning on page 2-18 for the connector location.

8. Attach the Ultra3000 with DeviceNet to the DeviceNet network.

Troubleshooting Status Indicators

Chapter Objectives

This chapter provides a description of maintenance and troubleshooting activities for the Ultra3000. This chapter includes these sections:

- Safety Precautions
- General Troubleshooting
- Troubleshooting for SERCOS Drives
- Troubleshooting for DeviceNet Drives

For power-up procedures and system integration with Ultraware or ControlLogix and SoftLogix modules/PCI cards (see table below) refer to the *Ultra3000 Digital Servo Drives Integration Manual* (publication 2098-IN005*x*-EN-P). Manuals are available electronically (as a .pdf) or in hardcopy from www.theautomationbookstore.com.

Interface	ControlLogix Motion Module	SoftLogix PCI Card
SERCOS interface	1756-M*xx*SE	1784-PM16SE
Analog interface	1756-M02AE	1784-PM02AE

Safety Precautions

Observe the following safety precautions when troubleshooting your Ultra3000 drive.

ATTENTION

DC bus capacitors may retain hazardous voltages after input power has been removed. Before working on the drive, measure the DC bus voltage to verify it has reached a safe level or wait the full time interval listed on the drive warning label. Failure to observe this precaution could result in severe bodily injury or loss of life.

Do not attempt to defeat or override the drive fault circuits. You must determine the cause of a fault and correct it before you attempt to operate the system. If you do not correct a drive or system malfunction, it could result in personal injury and/or damage to the equipment as a result of uncontrolled machine system operation.

Test equipment (such as an oscilloscope or chart recorder) must be properly grounded. Failure to include an earth ground connection could result in a potentially fatal voltage on the oscilloscope chassis.

General Troubleshooting

Refer to the *Error Codes* section below to identify problems, potential causes, and appropriate actions to resolve the problems. If problems persist after attempting to troubleshoot the system, please contact your Allen-Bradley representative for further assistance. To determine if your Ultra3000 drive has an error, refer to the table below.

If the Logic Power LED is ON and the Status LED display on your:	Is:	Then:
2098-DSD-*xxx*, -*xxx*X, -HV*xxx*, or -HV*xxx*X drive	Actively cycling segments in a full circle	Your Ultra3000 drive is ready.
2098-DSD-*xxx*-DN, -*xxx*X-DN, -HV*xxx*-DN, -HV*xxx*X-DN drive		Your Ultra3000 drive is ready.
2098-DSD-*xxx*-SE or -HV*xxx*-SE drive	Displaying a fixed 4	Your Ultra3000 drive is ready.
All drives	Flashing E followed by two numbers	Your Ultra3000 drive has an error. Proceed to the section *Error Codes* below.
	Flashing L	Your Ultra3000 drive is in an Overtravel condition and motion restrictions are in effect.

Error Codes

The following list of problematic symptoms (no error code shown) and problems with assigned error codes is designed to help you resolve problems.

When a fault is detected, the seven-segment LED will display an E followed by the flashing of the two-digit error code, one digit at a time. This is repeated until the problem is cleared.

Error Code	Problem or Symptom	Possible Cause(s)	Action/Solution
	Power (PWR) indicator not ON	No AC power or auxiliary logic power.	Verify AC power or auxiliary +5V logic power is applied to the Ultra3000.
		Internal power supply malfunction.	Call your Allen-Bradley representative.
	Power (PWR) indicator is ON, but seven segment Status LED display is OFF. Note: This applies to 2098-DSD-005, -010, and -020 Ultra3000 models only.	Externally applied +5V auxiliary power supply voltage is too low.	Verify that the external +5V auxiliary power supply (as measured at the drive terminals) reads between 5.10V and 5.25V.
	Motor jumps when first enabled	Motor wiring error.	Check motor wiring.
		Incorrect motor chosen.	Verify the proper motor is selected.
	Digital I/O not working correctly	I/O power supply disconnected.	Verify connections and I/O power source.
E01	Non-Volatile Memory Endurance Exceeded	Range of motion and number of home position definitions during the product life exceeds the maximum allowed (applies only to systems with absolute feedback).	This is an unrecoverable fault, the drive must be sent back to the factory.
E02	Velocity Exceeds Position Rollover /2	The velocity command or feedback exceeds half the machine cycle length per millisecond (applies only when the machine cycle position rollover is enabled).	Increase machine cycle size or reduce velocity profile. This error only applies to firmware versions prior to 1.10.

Error Code	Problem or Symptom	Possible Cause(s)	Action/Solution
E03	Absolute Feedback Range Exceeded	The motor position exceeds +/- 2047 revolutions from the home position (applies only to systems with absolute feedback).	• Decrease application range of motion. • Upgrade firmware.
E04	Motor Overtemperature	Motor thermostat trips due to: • High motor ambient temperature and/or • Excessive current	• Operate within (not above) the continuous torque rating for the ambient temperature (40°C maximum). • Lower ambient temperature, increase motor cooling.
		Motor wiring error.	Check motor wiring.
		Incorrect motor selection.	Verify the proper motor has been selected.
E05	IPMFault	Motor cables shorted.	Verify continuity of motor power cable and connector.
		Motor winding shorted internally.	Disconnect motor power cables from the motor. If the motor is difficult to turn by hand, it may need to be replaced.
		Ultra3000 temperature too high.	• Check for clogged vents or defective fan. • Ensure cooling is not restricted by insufficient space around the unit.
		Operation above continuous power rating.	• Verify ambient temperature is not too high. • Operate within the continuous power rating. • Reduce acceleration rates.
		Ultra3000 has a bad IPM output, short circuit, or overcurrent.	Remove all power and motor connections, and preform a continuity check from the DC bus to the U, V, and W motor outputs. If a continuity exists, check for wire fibers between terminals, or send drive in for repair.
		An attempt was made to enable the drive without waiting at least 1.0 second after applying the main AC power. Note: This applies to 2098-DSD-005, -010, and -020 Ultra3000 models only (when using an external +5V auxiliary power supply).	Wait at least 1.0 second after the main AC is applied before enabling the drive.
E06	Hardware Overtravel (SERCOS only)	Dedicated overtravel input is inactive.	• Check wiring. • Verify motion profile.
E07	RESERVED		Call your local Allen-Bradley representative.
E08	RESERVED		
E09	Bus Undervoltage	Low AC line/AC power input.	• Verify voltage level of the incoming AC power. • Check AC power source for glitches or line drop. • Install an uninterruptible power supply (UPS) on your AC input.

Error Code	Problem or Symptom	Possible Cause(s)	Action/Solution
E10	Bus Overvoltage	Excessive regeneration of power. When the motor is driven by an external mechanical power source, it may regenerate too much peak energy through the Ultra3000's power supply. The system faults to save itself from an overload.	• Change the deceleration or motion profile. • Use a larger system (motor and Ultra3000). • Use a resistive shunt. • If a shunt is connected, verify the wiring is correct and shunt fuse is not blown.
		Excessive AC input voltage.	Verify input is within specifications.
E11	IllegalHall State	Incorrect phasing.	Check the Hall phasing.
		Bad connections.	• Verify the Hall wiring. • Verify 5V power supply to the encoder.
E12	Home Search Failed	Home sensor and/or marker is outside the overtravel limits.	• Check wiring. • Reposition the overtravel limits or sensor.
E13	Home Position In Limit	Home sensor, marker, or final home position exceeds a hardware overtravel limit.	• Reposition the overtravel limits or home sensor. • Adjust the final home position.
E14	SERCOS Hardware Fault (SERCOS drives only)	A fault was detected with the operation of the drive's internal SERCOS hardware.	Contact your local Allen-Bradley representative.
	DeviceNet Communications Network problem (DeviceNet drives only)	DeviceNet communications network is broken	Troubleshoot DeviceNet communications.
E15	Excessive Electrical Cycle Length	Electrical cycle length exceeds maximum lines per electrical cycle	Replace the linear motor/encoder.
E16	Software Overtravel (SERCOS only)	Programmed overtravel limit has been exceeded.	• Verify motion profile. • Verify overtravel settings are appropriate.
E17	User-Specified Current Fault	User-Specified average current level has been exceeded.	Increase to a less restrictive setting.
E18	Overspeed Fault	Motor speed has exceeded 125% of maximum rated speed.	• Check cables for noise. • Check tuning.
E19	Excess Position Error	Position error limit was exceeded.	• Increase the feedforward gain. • Increase following error limit or time. • Check position loop tuning.

Error Code	Problem or Symptom	Possible Cause(s)	Action/Solution
E20	Motor Encoder State Error	The motor encoder encountered an illegal transition.	• Replace the motor/encoder. • Use shielded cables with twisted pair wires. • Route the feedback away from potential noise sources. • Check the system grounds. • Verify that the unbuffered encoder signals are not subjected to EMI in the CN1 cable. Remove these signals from the CN1 cable if they are not being used. • Verify that the motor has a high-frequency bond to the drive's enclosure panel. • Verify that any stage connected to the motor shaft (for example using a ball screw) has a high-frequency bond to the machine frame and the drive's enclosure panel.
		Bad encoder.	Replace motor/encoder.
E21	Auxiliary Encoder state error	The auxiliary encoder encountered an illegal transition.	• Use shielded cables with twisted pair wires. • Route the encoder cable away from potential noise sources. • Bad encoder - replace encoder. • Check the ground connections.
		Setup time violation for Step/Direction or CW/CCW input.	Check timing of Step/Direction or CW/CCW inputs to determine if setup time requirements are being met.
E22	Motor Thermal Protection Fault	The internal filter protecting the motor from overheating has tripped.	• Reduce acceleration rates. • Reduce duty cycle (ON/OFF) of commanded motion. • Increase time permitted for motion. • Use larger Ultra3000 and motor. • Check tuning.
E23	IPM Thermal Protection Fault	The internal filter protecting the drive from over heating has tripped.	• Reduce acceleration rates. • Reduce duty cycle (ON/OFF) of commanded motion. • Increase time permitted for motion. • Use larger Ultra3000 and motor. • Check tuning.
E24	Excess Velocity Error	Velocity error limit was exceeded.	• Increase time or size of allowable error. • Reduce acceleration. • Check tuning.
E25	Sensor Not Assigned	Homing or registration motion was attempted without a sensor assigned.	Assign a sensor to a digital input.
E26	User-Specified Velocity Fault	User specified velocity level was exceeded.	Increase to a less restrictive setting.
E27	Axis Not Homed	Absolute positioning was attempted without homing.	Verify homing sequence.

Error Code	Problem or Symptom	Possible Cause(s)	Action/Solution
E28	Motor Parameter Error	Parameter loaded from smart encoder or received from SERCOS controller is incompatible with the drive.	• Select a different motor through the SERCOS controller. • Select a different motor.
E29	Encoder Output Frequency Exceeded	Encoder output frequency exceeds the maximum user specified value. This only applies when the encoder output is synthesized by the drive.	• Increase the encoder output maximum frequency parameter. • Decrease the encoder interpolation parameter. • Increase the encoder output divider parameter.
E30	Encoder Communication Fault	Communication was not established with an intelligent encoder.	• Verify motor selection. • Verify the motor supports automatic identification. • Verify motor encoder wiring.
E31	Encoder Data	Encoder data is corrupted.	Replace the motor/encoder.
E32	Sine/Cosine Encoder Frequency Limit Exceeded	Maximum frequency of the sine/cosine circuitry has been exceeded.	• Decrease velocity. • Use encoder with lower resolution (before interpolation).
E33	Absolute Position Exceeds Position Rollover	Motion is commanded to a position outside the position rollover range. • An absolute index is initiated that specifies a position outside the position rollover range. • A homing cycle is initiated with the home position outside the position rollover range. • A define home is initiated with the home position outside the position rollover range. • A preset position is initiated that specifies a position outside the position rollover range.	Set motion command to a position within the position rollover range.
E34	Ground Fault	Wiring error.	Check motor power wiring.
		Motor internal ground short.	Replace motor.
		Internal malfunction.	Disconnect motor power cable from drive and enable drive with current limit set to 0. If fault remains, call your A-B representative. If fault clears, then a wiring error or motor internal problem exists.
E35	Precharge Fault	Low AC input voltage.	Check input AC voltage on all phases.
		Internal malfunction.	Call your A-B representative.
E36	Power Circuitry Overtemperature	Excessive heat exists in the power circuitry.	• Reduce acceleration rates. • Reduce duty cycle (ON/OFF) of commanded motion. • Increase time permitted for motion. • Use larger Ultra3000 and motor. • Check tuning.
E37	AC Line Loss	One or more phases of the input AC power is missing.	Check input AC voltage on all phases.
E38	RESERVED		Call your local Allen-Bradley representative.

Error Code	Problem or Symptom	Possible Cause(s)	Action/Solution
E39	Self-sensing Commutation Startup Error	Motion required for self-sensing startup commutation was obstructed.	• Verify that there are no impediments to motion at startup, such as hard limits. • Increase self-sensing current if high friction or load conditions exist. • Check motor or encoder wiring using wiring diagnostics.
E40	230V Shunt Protection Fault	Ineffective shunt resistor	• Verify that the shunt resistor (internal or external) is connected. • If an external shunt resistor is connected, verify that the shunt fuse is not blown. • If a non Allen-Bradley external shunt resistor is used, verify that the resistance value is within specifications. • Verify that the motor is not being driven mechanically, causing the motor to behave as a generator.
		Excessive regeneration	
E41	460V Shunt Protection Fault	Ineffective shunt resistor	
		Excessive regeneration	
E42	Motor Keying Error (SERCOS drives only)	The motor physically connected to the drive differs from the motor specified in the user program.	Select the correct motor in the user program.
E43	Drive Enable Input (SERCOS drives only)	• An attempt was made to enable the axis through software while the Drive Enable hardware input was inactive. • The Drive Enable input transitioned from active to inactive while the axis was enabled.	• Disable the Drive Enable Input fault. • Verify that Drive Enable hardware input is active whenever the drive is enabled through software.
E50	Duplicate Node Fault (SERCOS drives only)	Duplicate node address detected on SERCOS ring.	Verify that each SERCOS drive is assigned a unique node address.
All others	RESERVED		Call your local Allen-Bradley representative.

Troubleshooting for SERCOS Drives

SERCOS Module Status LED

Use the table below for troubleshooting the SERCOS Module Status LED on your Ultra3000 (2098-DSD-*xxx*-SE or -HV*xxx*-SE).

If the SERCOS Module Status LED is:	Status is:	Potential Cause is:	Possible Resolution is:
Steady Green	Normal	Drive is enabled.	Normal operation when drive is enabled.
Flashing Green	Standby	Drive is not enabled.	Normal operation when drive is disabled.
Flashing Red-Green	DC Bus Undervoltage	The DC bus voltage is low.	• Normal operation when using auxiliary power (main AC power is not applied). • When using main AC power, refer to the section *Error Codes* to continue troubleshooting.
Flashing Red	Minor fault	Drive is faulted, but the fault can be cleared.	Refer to the section *Error Codes* to continue troubleshooting.
Steady Red	Unrecoverable fault	Drive is faulted, and the fault cannot be cleared.	Contact your local Allen-Bradley representative.

SERCOS Network Status LED

Use the table below for troubleshooting the SERCOS Network Status LED on your Ultra3000 (2098-DSD-*xxx*-SE or -HV*xxx*-SE).

If the SERCOS Network Status LED is:	Status is:	Potential Cause is:	Possible Resolution is:
Steady Green	Communication ready	No faults or failures.	N/A
Flashing Green	Establishing communication	System is still in the process of establishing SERCOS communication.	Wait for steady green LED status.
		Node address setting on the drive module does not match SERCOS controller configuration.	Verify proper node switch setting.
Flashing Red	No communication [1]	Loose fiber optic connection.	Verify proper fiber optic cable connections.
		Broken fiber optic cable.	Replace fiber optic cable.
		Receive fiber optic cable connected to SERCOS transmit connector and vice versa.	Check proper SERCOS fiber optic cable connections.

[1] Refer to *Fiber Optic Cable Installation and Handling Instructions* (publication 2090-IN010*x*-EN-P) for more information.

Troubleshooting for DeviceNet Drives

DeviceNet Module Status LED

Use the table below for troubleshooting the DeviceNet Module Status LED on your Ultra3000 (2098-DSD-*xxx*-DN, -*xxx*X-DN, -HV*xxx*-DN, or -HV*xxx*X-DN).

If the Module Status LED is:	Status is:	Potential Cause is:	Possible Resolution is:
Off	Not powered	No power	There is no power going to the device.
Steady-Green	Operational	Normal operation	Normal operation - no action needed.
Flashing-Green	Device is in stand-by	Processing or waiting for input	Normal operation - no action needed.
Flashing-Red	Recoverable fault	Not operational	Power cycle or reset the drive.
Steady-Red	Unrecoverable fault	Drive problem	1. Check drive for power-up error. 2. Replace drive.
Flashing-Red/ Green	Self testing	Self-test in progress	The device is in self test, wait.

DeviceNet Network Status LED

Use the table below for troubleshooting the DeviceNet Network Status LED on your Ultra3000 (2098-DSD-*xxx*-DN, -*xxx*X-DN, -HV*xxx*-DN, or -HV*xxx*X-DN).

If the Network Status LED is:	Status is:	Potential Cause is:	Possible Resolution is:
Off	• Not powered • Not on-line	• No power going to the device • Failed Duplicate MAC ID check	1. Check the Module Status LED to verify that the drive is powered. 2. Check that one or more nodes are communicating on the network. 3. Check that at least one other node on the network is operational and the data rate is the same as the drive.
Flashing-green	• On-line • Not connected	• Passed Duplicate MAC ID check • No connection established	No action is needed. The LED is flashing to signify that there are no open communication connections between the drive and any other device. Any connection (I/O or explicit message) made to the drive over DeviceNet will cause the LED to stop flashing and remain Steady-ON for the duration of any open connection.
Steady-green	• On-line • Connected	One or more connections established	No action needed. This condition is normal.
Flashing-red	• On-line • Time-out	I/O connection timed out	1. Re-initiate I/O messaging by the master controller. 2. Reduce traffic or errors on the network so that messages can get through within the necessary time frame.
Steady-red	Network Failure	• Failed Duplicate MAC ID check • Bus-off	1. Ensure that all nodes have unique addresses. 2. If all node addresses are unique, examine network for correct media installation. 3. Ensure that all nodes have the same Data Rate.

Specifications and Dimensions

Chapter Objectives

This appendix covers the following topics:

- Certifications
- Ultra3000 Power Specifications
- Ultra3000 General Specifications
- Dimensions

Certifications

The Ultra3000 is certified for the following when the product or package is marked.

- UL listed to U.S. and Canadian safety standards (UL 508 C File E145959)
- CE marked for all applicable directives

Note: Refer to www.ab.com/certification/ce/docs for more information.

Ultra3000 Power Specifications

The following sections provide power specifications for the Ultra3000.

Ultra3000 (230V) Power Specifications

The table below lists general power specifications and requirements for the Ultra3000 230V drives (2098-DSD-005*x-xx*, -010*x-xx*, and -020*x-xx*).

Specification	**Description**		
	2098-DSD-005	**2098-DSD-010**	**2098-DSD-020**
AC Input Voltage [1]	100-240V_{rms} Single Phase		
AC Input Frequency	47 - 63 Hz		
AC Input Current [2, 3] Nominal Maximum inrush (230V ac input)	 $5A_{rms}$ 100A (0-peak)	 $9A_{rms}$ 100A (0-peak)	 $18A_{rms}$ 100A (0-peak)
Continuous Output Current	2.5A (0-peak)	5A (0-peak)	10A (0-peak)
Intermittent Output Current	7.5A (0-peak)	15A (0-peak)	30A (0-peak)
Bus Capacitance	1410 µF	1880 µF	1880 µF
Internal Shunt Resistance	N/A	N/A	N/A
Shunt On	N/A	N/A	N/A
Shunt Off	N/A	N/A	N/A
Bus Overvoltage	400V dc	400V dc	400V dc
Energy Absorption Capability 115V ac input 230V ac input	 93 Joules 38 Joules	 125 Joules 51 Joules	
Continuous Power Output 115V ac input 230V ac input	 0.25 kW 0.5 kW	 0.5 kW 1.0 kW	 1.0 kW 2.0 kW

[1] Specification is for nominal voltage. The absolute limits are ±10%, or 88-265V_{rms}.

[2] The 2098-005*x-xx*, -010*x-xx*, and -020*x-xx* (230V) drives are limited to one contactor cycles per two minutes.

[3] Power initialization requires a short period of inrush current. Dual element time delay (slow blow) fuses are recommended (refer to *Fuse Specifications* on page A-5).

IMPORTANT Only the 2098-DSD-005, -010, and -020 models support an auxiliary +5V logic supply since an auxiliary AC input is not available. Refer to *Auxiliary 5V Logic Supply* on page 2-26 for more information.

The table below lists general power specifications and requirements for the Ultra3000 230V drives (2098-DSD-030*x-xx*, -075*x-xx*, and -150*x-xx*).

Specification	**Description**		
	2098-DSD-030	**2098-DSD-075**	**2098-DSD-150**
AC Input and Auxiliary Input Voltage[1]	100-240V_{rms} Single-Phase	100-240V_{rms} Three-Phase	
AC Input Frequency	47 - 63 Hz		
Main AC Input Current [2, 4] Nominal, Maximum inrush, 230V ac input	 28A_{rms} 50A (0-peak)	 30A_{rms} 50A (0-peak)	 46A_{rms} 68A (0-peak)
Auxiliary AC Input Current Nominal, 115V ac input Nominal, 230V ac input Maximum inrush, 115V ac input [3] Maximum inrush, 230V ac input [3]	 1.0A_{rms} 0.5A_{rms} 47A (0-peak) 95A (0-peak)	 1.0A_{rms} 0.5A_{rms} 47A (0-peak) 95A (0-peak)	 1.0A_{rms} 0.5A_{rms} 47A (0-peak) 95A (0-peak)
Continuous Output Current	15A (0-peak)	35A (0-peak)	65A (0-peak)
Intermittent Output Current	30A (0-peak)	75A (0-peak)	150A (0-peak)
Bus Capacitance	2370 μF	4290 μF	7520 μF
Internal Shunt Resistance	35 Ohms	16.5 Ohms	9.1 Ohms
Shunt On	420V dc	420V dc	420V dc
Shunt Off	402V dc	402V dc	402V dc
Bus Overvoltage	452V dc	452V dc	452V dc
Internal Shunt Continuous power Peak power	 50W 4.5 kW	 50W 10 kW	 180W 18 kW
External Shunt Resistance Continuous power Peak power	 30 Ohms (-0/+5%) 2.4 kW 6 kW	 16.5 Ohms (-0/+5%) 4 kW 10 kW	 9 Ohms (-0/+5%) 8 kW 19 kW
Energy Absorption Capability 115V ac input 230V ac input	 211 Joules 117 Joules	 381 Joules 211 Joules	 669 Joules 370 Joules
Continuous Power Output 115V ac input 230V ac input	 1.5 kW 3 kW	 3.75 kW 7.5 kW	 7.5 kW 15 kW

[1] Specification is for nominal voltage. The absolute limits are ±10%, or 88-265V_{rms}.

[2] The 2098-DSD-030*x-xx*, -075*x-xx*, and -150*x-xx* (230V) drives are limited to one contactor cycles per two minutes.

[3] 400 μs half wave sine.

[4] Power initialization requires a short period of inrush current. Dual element time delay (slow blow) fuses are recommended (refer to *Fuse Specifications* on page A-5).

Ultra3000 (460V) Power Specifications

The table below lists general power specifications and requirements for the Ultra3000 460V drives (2098-DSD-HV030*x-xx*, -HV050*x-xx*, -HV100*x-xx*, -HV150*x-xx*, and -HV220*x-xx*).

Specification	**Description**				
	2098-DSD-HV030	**2098-DSD-HV050**	**2098-DSD-HV100**	**2098-DSD-HV150**	**2098-DSD-HV220**
AC Input and Auxiliary Input Voltage [1, 2]	230-480V_{rms} Three Phase				
AC Input Frequency	47 - 63 Hz				
Main AC Input Current [3, 5] Nominal, 460V ac input Maximum inrush, 460V ac input	4A_{rms} 6A_{rms}	7A_{rms} 6A_{rms}	14A_{rms} 6A_{rms}	20A_{rms} 6A_{rms}	28A_{rms} 6A_{rms}
Auxiliary AC Input Current Nominal, 230V ac input Nominal, 360V ac input Nominal, 480V ac input Maximum inrush, 230V ac input [4] Maximum inrush, 480V ac input [4]	0.55A_{rms} 0.35A_{rms} 0.25A_{rms} 47A (0-peak) 68A (0-peak)	0.55A_{rms} 0.35A_{rms} 0.25A_{rms} 47A (0-peak) 68A (0-peak)	0.55A_{rms} 0.35A_{rms} 0.25A_{rms} 47A (0-peak) 68A (0-peak)	0.55A_{rms} 0.35A_{rms} 0.25A_{rms} 47A (0-peak) 68A (0-peak)	0.55A_{rms} 0.35A_{rms} 0.25A_{rms} 47A (0-peak) 68A (0-peak)
Continuous Output Current	7A (0-peak)	11A (0-peak)	23A (0-peak)	34A (0-peak)	47A (0-peak)
Intermittent Output Current	14A (0-peak)	22A (0-peak)	46A (0-peak)	68A (0-peak)	94A (0-peak)
Bus Capacitance	470 µF	470 µF	705 µF	940 µF	1880 µF
Internal Shunt Resistance	120 Ohms	120 Ohms	40 Ohms	25 Ohms	20 Ohms
Shunt On	800V dc	800V dc	800V dc	800V dc	800V dc
Shunt Off	750V dc	750V dc	750V dc	750V dc	750V dc
Bus Overvoltage	810V dc	810V dc	810V dc	810V dc	810V dc
Internal Shunt Continuous power Peak power	100W 5.3 kW	100W 5.3 kW	200W 16 kW	200W 25.6 kW	400W 32 kW
External Shunt Resistance (-0/+5%) Continuous power Peak power	120 Ohms 3 kW 5.3 kW	120 Ohms 5 kW 5.3 kW	40 Ohms 10 kW 16 kW	25 Ohms 15 kW 25.6 kW	20 Ohms 22 kW 32 kW
Energy Absorption Capability 230V ac input with 230V motor 230V ac input with 460V motor 460V ac input	15 Joules 129 Joules 55 Joules	15 Joules 129 Joules 55 Joules	22 Joules 194 Joules 82 Joules	29 Joules 259 Joules 109 Joules	59 Joules 517 Joules 219 Joules
Continuous Power Output 230V ac input 460V ac input	1.5 kW 3.0 kW	2.5 kW 5.0 kW	5.0 kW 10 kW	7.5 kW 15 kW	11 kW 22 kW

[1] Specification is for nominal voltage. The absolute limits are ±10%, or 207-528V_{rms}.

[2] The 2098-DSD-HV*xxx-xx* drives can be powered with 230-240V_{rms} in order to be used in conjunction with motors designed for 230V operation. In such cases, the voltage levels used for shunting and DC bus overvoltage limits are adjusted to be compatible with the voltage limit of the motor.

[3] The 2098-DSD-HV*xxx -xx* (460V) drives are limited to three contactor cycles per minute.

[4] 400 µs half wave sine.

[5] Power initialization requires a short period of inrush current (processor controlled via soft start circuitry). Dual element time delay (slow blow) fuses are recommended (refer to *Fuse Specifications* on page A-5).

Fuse Specifications

Use class CC, G, J, L, R, or T class fuses, with current ratings as indicated in the table below. The table below lists fuse examples recommended for use with the Ultra3000 (230V and 460V) drives.

Refer to *Power Wiring Requirements* in *Chapter 3* for input wire size.

Catalog Number	Input Voltage	Voltage Type	Recommended Fuse	
			Class CC [1]	Class J [1]
2098-DSD-005*x-xx*	230V	Input Power	FNQ-R-6	LPJ-6SP
2098-DSD-010*x-xx*			FNQ-R-10	LPJ-10SP
2098-DSD-020*x-xx*			FNQ-R-20	LPJ-20SP
2098-DSD-030*x-xx*			FNQ-R-30	LPJ-30SP
2098-DSD-075*x-xx*			FNQ-R-30	LPJ-30SP
2098-DSD-150*x-xx*			N/A	LPJ-60SP
2098-DSD-*xxxx-xx*		Auxiliary Input Power	FNQ-R-10	LPJ-10SP
2098-DSD-HV030*x-xx*	460V	Input Power	KTK-R-5	LPJ-5SP
2098-DSD-HV050*x-xx*			KTK-R-8	LPJ-8SP
2098-DSD-HV100*x-xx*			KTK-R-20	LPJ-17-1/2SP
2098-DSD-HV150*x-xx*			KTK-R-30	LPJ-30SP
2098-DSD-HV220*x-xx*			N/A	LPJ-35SP
2098-DSD-HV*xxxx-xx*		Auxiliary Input Power	FNQ-R-10	LPJ-10SP

[1] Bussmann® Fuse

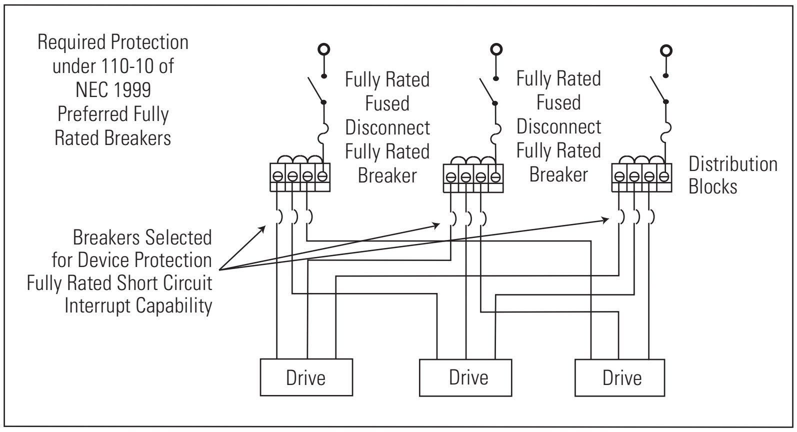
IMPORTANT Follow the wire size information as shown in the *Power Wiring Requirements* table on page 3-13 and select the fuse for the appropriate temperature rating of the wire. Auxiliary input fuses must not exceed 13A.

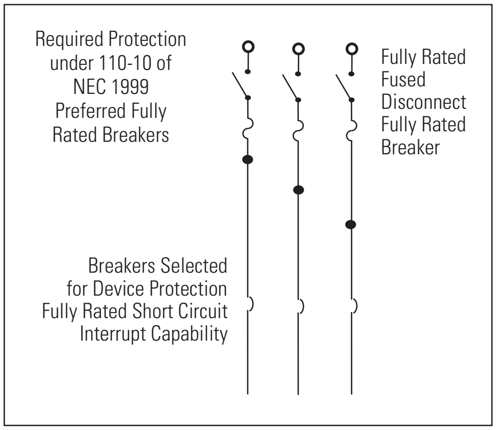
Circuit Breaker Specifications

While circuit breakers offer some convenience, there are limitations for their use. Circuit breakers do not handle high current inrush as well as fuses. The Ultra3000 needs to be protected by a device having a short circuit interrupt current rating of the service capacity provided or a maximum of 100,000A.

The wiring interconnection in Figure A.1 provide examples of the needed protection and follows UL and NEC codes. Full compliance is dependent on final wiring design and installation.

Figure A.1
Circuit Protection under NEC 1999 110-10 (preferred fully rated devices)





The table below lists circuit breaker examples recommended for use with the Ultra3000 (460V) drives.

Catalog Number	Input Voltage	Circuit Breakers
2098-DSD-HV030*x-xx*	460V	140M-F8E-C16
2098-DSD-HV050*x-xx*		140M-F8E-C20
2098-DSD-HV100*x-xx*		140M-F8E-C32
2098-DSD-HV150*x-xx*		140M-F8E-C45
2098-DSD-HV220*x-xx*		N/A

ATTENTION

Bulletin 140M circuit breakers should not be used on the output of an ac drive as an isolating disconnect switch or motor overload device. These devices are designed to operate on sine wave voltage and the drive's PWM waveform does not allow it to operate properly. As a result, damage to the device will occur.

Contactor Ratings

The table below lists contactor examples recommended for use with the Ultra3000 (460V) drives.

Catalog Number	**Input Voltage**	**Contactor**
2098-DSD-HV030*x*-*xx*	460V	100-C23*x*10 (AC Coil) 100-C23Z*x*10 (DC Coil)
2098-DSD-HV050*x*-*xx*		100-C30*x*10 (AC Coil) 100-C30Z*x*10 (DC Coil)
2098-DSD-HV100*x*-*xx*		100-C37*x*10 (AC Coil) 100-C37Z*x*10 (DC Coil)
2098-DSD-HV150*x*-*xx*		100-C43*x*10 (AC Coil) 100-C43Z*x*10 (DC Coil)
2098-DSD-HV220*x*-*xx*		100-C60*x*10 (AC Coil) 100-C60Z*x*10 (DC Coil)

Power Dissipation Specifications

Use the following table to size an enclosure and calculate required ventilation for the Ultra3000. Typical heat losses run approximately one-half maximum power losses. The maximum power losses are shown below.

Catalog Number	**Maximum Loss** Watts
2098-DSD-005*x*-*xx*	48
2098-DSD-010*x*-*xx*	48
2098-DSD-020*x*-*xx*	50
2098-DSD-030*x*-*xx*	150 + dissipative shunt
2098-DSD-075*x*-*xx*	300 + dissipative shunt
2098-DSD-150*x*-*xx*	500 + dissipative shunt

Catalog Number	**Maximum Loss** Watts
2098-DSD-HV030*x*-*xx*	175 + dissipative shunt
2098-DSD-HV050*x*-*xx*	175 + dissipative shunt
2098-DSD-HV100*x*-*xx*	350 + dissipative shunt
2098-DSD-HV150*x*-*xx*	350 + dissipative shunt
2098-DSD-HV220*x*-*xx*	600 + dissipative shunt

Ultra3000 General Specifications

The following sections provide physical, environmental, control, I/O, communication, feedback, connector, and AC line filter specifications for the Ultra3000 drives.

Physical and Environmental Specifications

Specification	Description		Specification	Description	
Weight			Weight		
2098-DSD-005*x-xx*	1.8 kg	(4.1 lbs)	2098-DSD-HV030*x-xx*	8.55 kg	(18.8 lbs)
2098-DSD-010*x-xx*	2.1 kg	(4.6 lbs)	2098-DSD-HV050*x-xx*	8.55 kg	(18.8 lbs)
2098-DSD-020*x-xx*	2.1 kg	(4.6 lbs)	2098-DSD-HV100*x-xx*	10.44 kg	(22.96 lbs)
2098-DSD-030*x-xx*	6.2 kg	(13.6 lbs)	2098-DSD-HV150*x-xx*	10.44 kg	(22.96 lbs)
2098-DSD-075*x-xx*	9.3 kg	(20.6 lbs)	2098-DSD-HV220*x-xx*	14.1 kg	(31.0 lbs)
2098-DSD-150*x-xx*	14.1 kg	(31.0 lbs)			
Ambient Temperature	Storage: -40° C to 70° C (-40° F to 158° F) Operation: 0° C to 55° C (32° F to 131° F)				
Relative Humidity	5-95% non-condensing				
Altitude	1500 m (4921.5 ft) - Derate 3% per 300 m (984.3 ft) above 1500 m (4,921.5 ft)				
Vibration	5-2000 Hz @ 2.5g peak, 0.0006 mm (0.015 in.) maximum displacement				
Shock	15g, 11 ms half-sine				

Control Specifications

Specification	Description
Commutation	3-Phase Sinusoidal, Space Vector Modulated (SVM)
Current Regulator	Digital PI - 125 µs update rate
Velocity Regulator	Digital PID - 250 µs update rate
Position Regulator	Digital PID with feed-forward - 1 ms update rate
PWM	4 or 8 kHz, space vector modulation
Velocity Loop Bandwidth (maximum)	300 Hz

Inputs and Outputs Specifications

Specification	Description
Digital Inputs	8 Optically Isolated 12-24V Inputs, Active High, Current Sinking
Digital Outputs	4 Optically Isolated 12-24V Outputs, Active High, Current Sourcing
Relay Output	1 Normally Open Relay - 30V dc Maximum Voltage, 1A Maximum Current
I/O Response	100 µsec
Digital I/O Firmware Scan Period	1 mS
Analog Inputs	
COMMAND	14 bit A/D, ±10V
ILIMIT	10 bit A/D, 0 to 10V
Analog Output	±10V, 8 bits, 2 mA maximum

Communication Specifications

Specification	Description
SERCOS (option)	
Baud Rates	4 and 8 Mbit
Node Addresses	01-99
DeviceNet (option)	
Power Consumption from Network	60 mA
Data Rates	125, 250, and 500 kps, and auto-baud
Node Addresses	00-63
Messaging Capabilities	Explicit, Polled I/O, Change of State, and Cyclic Messaging
Serial	
Ports	One RS-232/RS-422/RS-485
Baud Rates	1200, 2400, 4800, 9600, 19200, and 38400 baud

Motor Feedback Specifications

Specification	Description
Encoder Types	Incremental, Sine/Cosine, Intelligent, and Absolute
Maximum Input Frequency	100 kHz (Sine/Cosine Input)
	2.5 MHz (TTL Input) per channel
Commutation Startup	Hall Sensor or None

Auxiliary Feedback Specifications

Specification	Description
Input Modes	A quad B, Step/Direction, CW/CCW
Maximum Signal Frequency	2.5 MHz
Input Types	Incremental only, (differential, single-ended, open collector [1])

[1] Differential input types are recommended.

Connector Specifications

The table below lists connector specifications. Refer to *Appendix C* for a list of drive connectors available from other suppliers.

Connector	Description	Specification
CN1	User Input/Output	44-pin High Density Female D-Sub Connector
CN2	Motor Feedback Connector	15-pin High Density Female D-Sub Connector
CN3	Serial Port Connector	9-pin Female D-Sub Connector
TB1 and TB2	Main and Auxiliary AC, DC Bus, Motor Power, and Shunt Connectors	Screw Terminal Block

AC Line Filter Specifications

The following AC line filters are compatible with the Ultra3000 drives.

AC Line Filter Catalog Number	Specifications							
	Voltage	**Phase**	**Current**	**Power Loss**	**Weight** kg (lb)	**Humidity**	**Vibration**	**Operating Temperature**
2090-UXLF-106	250V ac 50/60 Hz	Single	6A @ 50° C (122° F)	3.5W	0.3 (0.66)	90% RH	10-200 Hz @ 1.8 g	-25 to 85° C (-13 to 185 ° F)
2090-UXLF-110			10A @ 50° C (122° F)	2.7W	0.95 (2.0)			
2090-UXLF-123			23A @ 50° C (122° F)	10W	1.6 (3.5)			
2090-UXLF-132			32A @ 50° C (122° F)	20W				
2090-UXLF-136			36A @ 50° C (122° F)	–	1.75 (3.9)			
2090-UXLF-336	250V ac 50/60 Hz	Three		–	2.7 (5.9)			
2090-UXLF-350			50A @ 50° C (122° F)	–				
2090-UXLF-HV323	520V ac 50/60 Hz		23A @ 50° C (122° F)	20W	1.6 (3.5)			
2090-UXLF-HV330			30A @ 50° C (122° F)	51W	1.8 (4.0)			
2090-UXLF-HV350			50A @ 50° C (122° F)	25W	4.8 (10.6)			

Use the table below to determine which AC line filter is best suited for your Ultra3000 drive (based on the length of the motor cables).

Ultra3000 Drives	AC Line Filter Catalog Number	
	Motor Cables < 30 m	Motor Cables > 30 m
2098-DSD-005*x-xx*	2090-UXLF-106	2090-UXLF-110
2098-DSD-010*x-xx*	2090-UXLF-110	2090-UXLF-110
2098-DSD-020*x-xx*	2090-UXLF-123	2090-UXLF-123
2098-DSD-030*x-xx*	2090-UXLF-136	2090-UXLF-132
2098-DSD-075*x-xx*	2090-UXLF-336	2090-UXLF-HV330
2098-DSD-150*x-xx*	2090-UXLF-350	2090-UXLF-HV350
2098-DSD-HV030*x-xx* 2098-DSD-HV050*x-xx* 2098-DSD-HV100*x-xx* 2098-DSD-HV150*x-xx*	2090-UXLF-HV323	2090-UXLF-HV323
2098-DSD-HV220*x-xx*	2090-UXLF-HV330	2090-UXLF-HV330

Ultra Family External Shunt Module Specifications

The following external shunt modules are compatible with the Ultra3000 drives with regenerative loads that exceed the capacity of the internal shunt resistor.

Ultra3000 Drives	Shunt Module Catalog Number	Specifications						Fuse Replacement
		Drive Voltage VAC	**Resistance** Ohms	**Peak Power** kW	**Peak Current** Amps	**Continuous Power** Watts	**Shipping Weight** kg (lbs)	
2098-DSD-005, 010, and 020	2090-UCSR-A300	230	36	4.0	10.5	300	1.51 (3.3)	—
2098-DSD-030	9101-1183		30	5.9	14.0	200	—	CCMR-4-½[2]
2098-DSD-075 and 150	2090-UCSR-P900		18	10	23.3	900	4.08 (9.0)	FWP-10A14F[1]
2098-DSD-HV030 and -HV050	2090-SR120-09	460	120	5.3	6.7		3.63 (8.0)	FWP-2.5A14F[1]
2098-DSD-HV100	2090-SR040-09		40	16	20.0			FWP-5A14F[1]
	2090-SR040-18		40		20.0	1800	8.6 (19.0)	FWP-10A14F[1]
2098-*xxx*-HV150	M3575R-H27B0,C [3]		25-31	25.6	31.0	900	11.3 (25.0)	N/A
	M3575R-H27BF,C [3]					1800		N/A
2098-*xxx*-HV220	M3575R-H33BF,C [3]		20-25	32.0	36.0	3600	12.7 (28.0)	N/A

[1] Bussmann part number.

[2] LittelFuse® part number.

[3] Bonitron part number. For more information contact:
Bonitron, Inc.
521 Fairground Court,
Nashville, TN 37211
Tel: (615) 244-2825
www.BONITRON.com

Maximum Feedback Cable Lengths

Although motor feedback cables are available in standard lengths up to 90 m (295.3 ft), the drive/motor/feedback combination may limit the maximum cable length, as shown in the tables below. These tables assume the use of cables recommended in the *Motion Control Selection Guide* (publication GMC-SG001*x*-EN-P).

The maximum cable lengths for Ultra3000 drives with MP-Series (low inertia and integrated gear) motors are given in the table below.

Drive Family	**MPL-A (230V) Motors**		**MPL-B (460V) Motors**		**MPG-A (230V) Motors**	**MPG-B (460V) Motors**
	Absolute High-Res [1] m (ft)	**Incremental** [2] m (ft)	**Absolute High-Res** [1] m (ft)	**Incremental** [2] m (ft)	**Absolute High-Res** [3] m (ft)	**Absolute High-Res** [3] m (ft)
Ultra3000	90 (295.3)	45 (147.6)	90 (295.3)	45 (147.6)	90 (295.3)	60 (196.8)

[1] Refers to MPL-A/B*xxxx*S/M (single-turn or multi-turn) low inertia motors with absolute high-resolution feedback.

[2] Refers to MPL-A/B*xxxx*H low inertia motors with 2000-line incremental feedback.

[3] Refers to MPG-A/B*xxxx*S/M (single-turn or multi-turn) integrated gear motors with absolute high-resolution feedback.

The maximum cable lengths for Ultra3000 drives with MP-Series food grade motors are given in the table below.

Drive Family	**MPF-A (230V) Motors**	**MPF-B (460V) Motors**
	Absolute High-Resolution [1] m (ft)	**Absolute High-Resolution** [1] m (ft)
Ultra3000	90 (295.3)	90 (295.3)

[1] Refers to MPF-A/B*xxxx*S/M (single-turn or multi-turn) food grade motors with absolute high-resolution feedback.

The maximum cable lengths for Ultra3000 drives with 1326AB (M2L/S2L) and F-, H-, N-, and Y-Series motors are given in the table below.

Drive Family	**1326AB (M2L/S2L) (460V) Motors**	**F-, H-, N-, or Y-Series (230V) Motors**
	Absolute High-Resolution [1] m (ft)	**Incremental** [2] m (ft)
Ultra3000	90 (295.3)	30 (98.4)

[1] Refers to 1326AB-B*xxxx*-M2L/S2L (single-turn or multi-turn) motors with absolute high-resolution feedback.

[2] Refers to F-, H-, N-, or Y-Series motors with incremental (optical encoder) feedback.

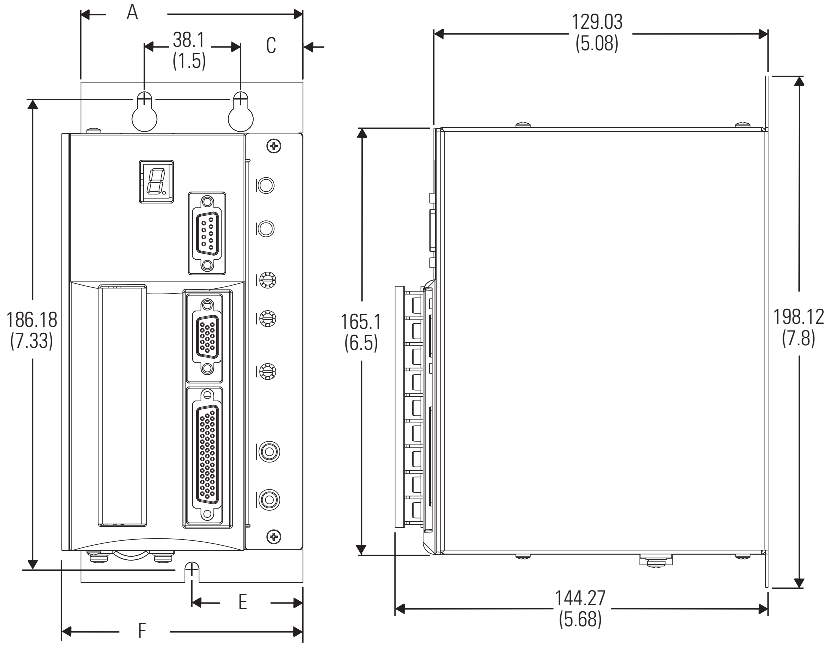
Dimensions

The following diagrams show the dimensions and mounting hole locations for the Ultra3000 drives.

Ultra3000 (230V) Dimensions

In the figure below, -*xxx* is replaced by -005, -010, or -020 to represent the Ultra3000 500W, 1 kW, and 2 kW drives respectively.

Figure A.2
Ultra3000 (230V) Dimensions (2098-DSD-*xxx*, -*xxx*X, -*xxx*-SE, -*xxx*-DN, -*xxx*X-DN)



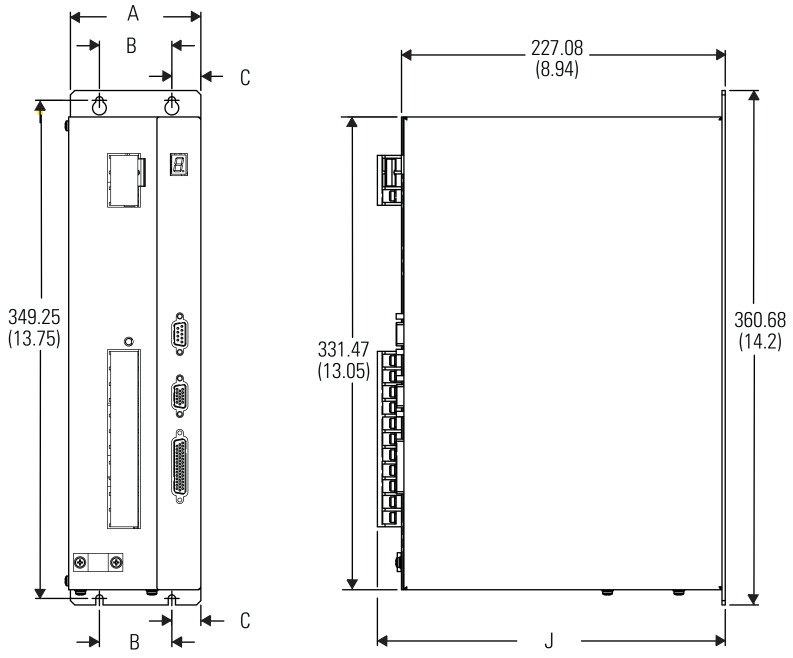
Dimensions are in millimeters (inches)

Unit shown is the 2098-DSD-005-SE

Ultra3000 Drive 2098-DSD	**A**	**C**	**E**	**F**
-005, -005X	65.02 (2.56)	13.26 (0.52)	32.77 (1.29)	72.64 (2.86)
-010, -010X, -020, -020X				98.1 (3.89)
-005-SE, 005-DN, and -005X-DN	87.88 (3.46)	24.64 (0.97)	43.94 (1.73)	95.5 (3.76)
-010-SE, -010-DN, -010X-DN, -020-SE, -020-DN, -020X-DN				121.54 (4.79)

In the figure below, -*xxx* is replaced by -030, -075, or -150 to represent the Ultra3000 3, 7.5, and 15 kW drives respectively.

Figure A.3
Ultra3000 (230V) Dimensions (2098-DSD-*xxx*, -*xxx*X, -*xxx*-SE, -*xxx*-DN, -*xxx*X-DN)



Dimensions are in millimeters (inches)

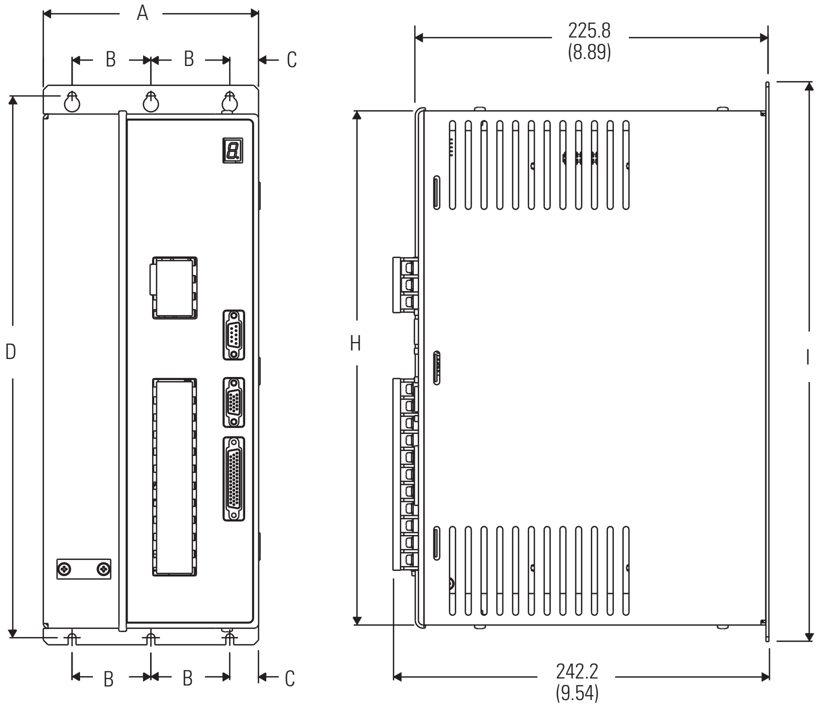
Unit shown is the 2098-DSD-030

Ultra3000 Drive 2098-DSD	**A**	**B**	**C**	**J**
-030, -030X, -030-SE, -030-DN, -030X-DN	91.44 (3.6)	50.8 (2.0)	20.32 (0.8)	243.84 (9.6)
-075, -075X, -075-SE, -075-DN, -075X-DN	138.68 (5.41)	88.9 (3.5)	24.89 (0.96)	247.14 (9.73)
-150, -150X, -150-SE, -150-DN, -150X-DN	188.97 (7.44)	139.7 (5.5)	24.6 (0.97)	241.05 (9.49)

Ultra3000 (460V) Dimensions

In the figure below, *xxx* is replaced by 030, 050, 100, 150, or 220 to represent the Ultra3000 3, 5, 10, 15, and 22 kW drives respectively.

Figure A.4
Ultra3000 (460V) Dimensions (2098-DSD-HV*xxx*, -HV*xxx*X, -HV*xxx*-SE, -HV*xxx*-DN, -HV*xxx*X-DN)



Dimensions are in millimeters (inches)

Unit shown is the 2098-DSD-HV030

Ultra3000 Drive[1] 2098-DSD-HV	**A**	**C**	**B**	**D**	**H**	**I**
030*x*, 030-*xx*, 050*x*, 050-*xx*	138.7 5.46)	18.5 (0.73)	50.8 (2.0)	349.3 (13.75)	331.5 (13.05)	360.7 (14.2)
100*x*, 100-*xx*, 150*x*, 150-*xx*	151.6 (5.97)	25 (0.99)				
220*x*, 220-*xx*	203.2 (8.0)	25.4 (1.0)	76.2 (3.0)	380.4 (14.98)	362.6 14.26)	391.8 (15.43)

[1] The *x* represents the indexing (X) option. The -*xx* represents the SERCOS interface (SE) or DeviceNet (DN) option. SERCOS interface is not available with the DeviceNet option.

Interconnect Diagrams

Chapter Objectives

This appendix contains the following interconnect diagrams:

- Power Interconnect Diagrams
- Shunt Module Interconnect Diagrams
- Ultra3000/Motor Interconnect Diagrams
- Control String Examples (120V ac)
- Controlling a Brake Example
- Ultra3000 to Logix Cable and Interconnect Diagrams
- Ultra3000 to IMC-S Compact Cable and Interconnect Diagram

Ultra3000 Interconnect Diagram Notes

The notes in the table below apply to the power, drive/motor, shunt, and 120V ac control string interconnect diagrams.

ATTENTION

The National Electrical Code and local electrical codes take precedence over the values and methods provided. Implementation of these codes are the responsibility of the machine builder.

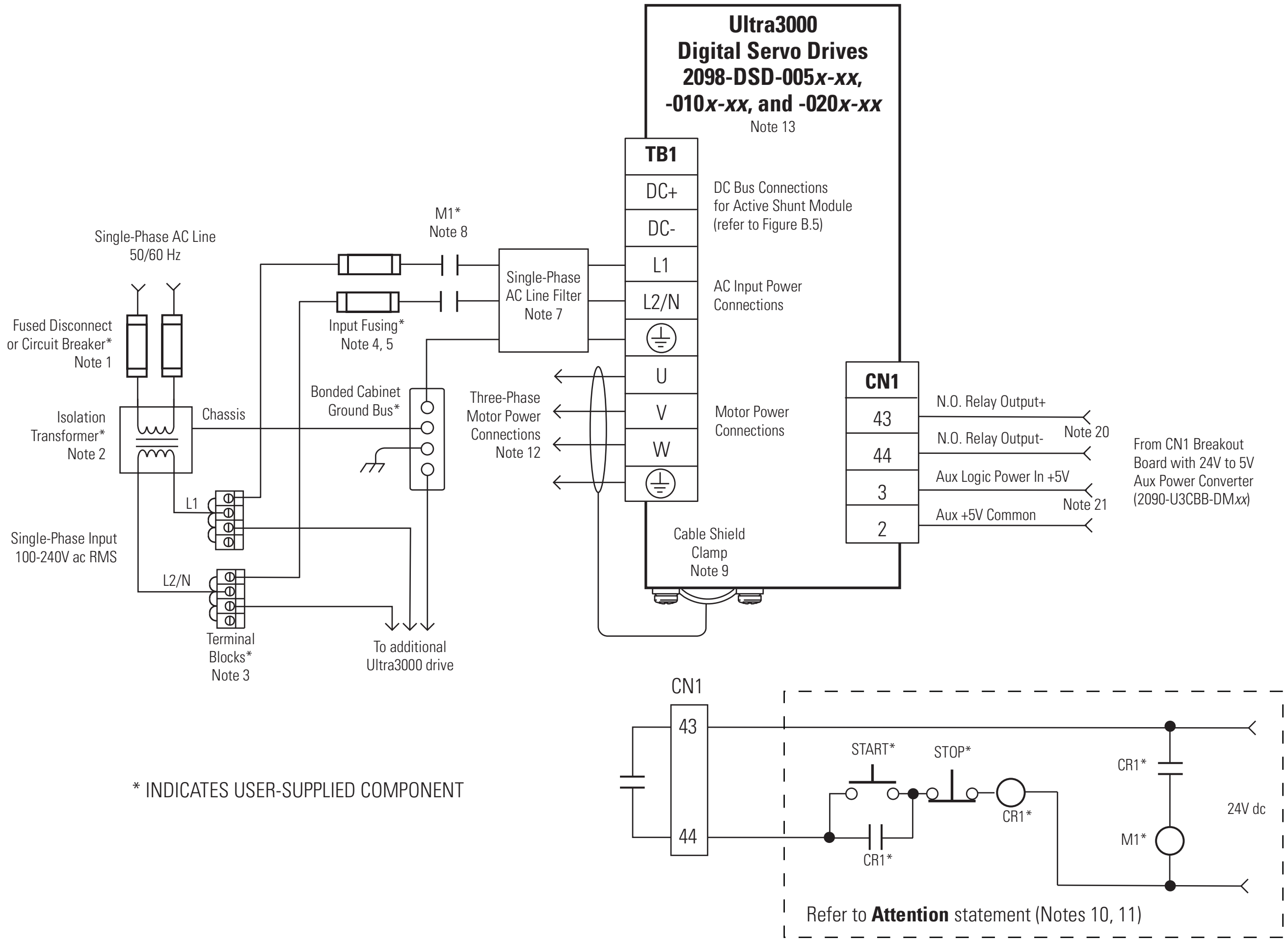
Note:	Information:
1	A disconnecting device is required for maintenance and safety. If a grounded neutral is used instead of L2, only L1 may be switched or fused.
2	An isolation transformer is optional. If the transformer secondary has a neutral connection, neutral must be bonded to ground. Multiple drive modules may be powered from one transformer or other AC supply source.
3	Do not daisy chain drive module power connections. Make separate connections directly to the AC supply.
4	For power wiring specifications, refer to *Power Wiring Requirements* in *Chapter 3*.
5	For input fuse sizes, refer to *Fuse Specifications* in *Appendix A*.
6	May be used to maintain power to logic section of drive and status LEDs when main AC input power is removed. A separate AC line source may be used if voltage is between 88-265V ac$_{RMS}$ on 2098-DSD-*xxx* (230V drives) or 207-528V ac$_{RMS}$ on 2098-DSD-HV*xxx* (460V drives).
7	For AC line filter specifications, refer to *AC Line Filter Specifications* in *Appendix A*.
8	Drive Enable input must be opened when main power is removed and auxiliary power is present, or a drive fault will occur. A delay of at least 1.0 second must be observed before attempting to enable the drive after main power is restored.
9	Cable shield clamp must be used in order to meet CE requirements. No external connection to chassis ground required.
10	**ATTENTION** Implementation of safety circuits and risk assessment is the responsibility of the machine builder. Please reference international standards EN1050 and EN954 estimation and safety performance categories. For more information refer to *Understanding the Machinery Directive* (publication SHB-900).
11	The recommended minimum wire size for wiring the safety circuit to the contactor enable connector is 1.5 mm^2 (16 AWG).
12	For motor cable specifications and drive/motor cable combinations, refer to the *Motion Control Selection Guide* (publication GMC-SG001*x*-EN-P).
13	The Ultra3000 referenced is either a 2098-DSD-*xxx* or -*xxx*X (Ultra3000 with indexing), -*xxx*-SE (SERCOS interface), -*xxx*-DN (DeviceNet interface), -*xxx*X-DN (DeviceNet with indexing) 230V drive.
14	The Ultra3000 referenced is either a 2098-DSD-HV*xxx* or -HV*xxx*X (Ultra3000 with indexing), -HV*xxx*-SE (SERCOS interface), -HV*xxx*-DN (DeviceNet interface), -HV*xxx*X-DN (DeviceNet with indexing) 460V drive.
15	Wire colors are for flying lead cable (2090-XXNF*xx*-S*xx*) and may vary from the premolded connector cable (2090-UXNFB*xx*-S*xx*). Wires without terminations at the drive are not shown for clarity.
16	If flying lead feedback cable has a drain wire, it must be folded back and clamped with the cable shield (CN2 breakout board 2090-UXBB-DM15).
17	Only the MPG-B*xxx* encoder uses the +5V dc supply. MPL-B*xxx* and 1326AB (M2L/S2L) encoders use the +9V dc supply.
18	Only the MPL-A5*xx* and MPF-A5*xx* encoders use the +9V dc supply. MPG-A*xxx* encoders use the +5V dc supply.
19	Use a flyback diode for noise suppression of the motor brake coil. For more information, refer to *System Design for Control of Electrical Noise Reference Manual* (publication GMC-RM001*x*-EN-P).
20	Relay Output (CN1, pins 43 and 44) must be configured as Ready in Ultraware software.
21	The preferred method for supplying the auxiliary power is by using the 12- or 44-pin drive-mounted breakout board with 24V to 5V auxiliary power converter (catalog number 2090-U3CBB-DM12 or -DM44). Auxiliary +5V power is required to maintain encoder position when the main AC power is disconnected.

Power Interconnect Diagrams

The Ultra3000 (2098-DSD-005*x-xx*, -010*x-xx*, and -020*x-xx*) power wiring with 24V dc control string (non-SERCOS drives only) is shown in the figure below. To avoid a separate 5V dc auxiliary logic power supply, the 24V to 5V converter breakout board (2090-U3CBB-DM*xx*) is used to wire the control interface (CN1) connector. For the control string diagram with 120V ac input refer to Figure B.16.

For SERCOS drives, input line contactor is part of the PLC program and output control.

Figure B.1
Typical Power Wiring of Ultra3000 System
(2098-DSD-005*x-xx*, -010*x-xx*, and -020*x-xx*)

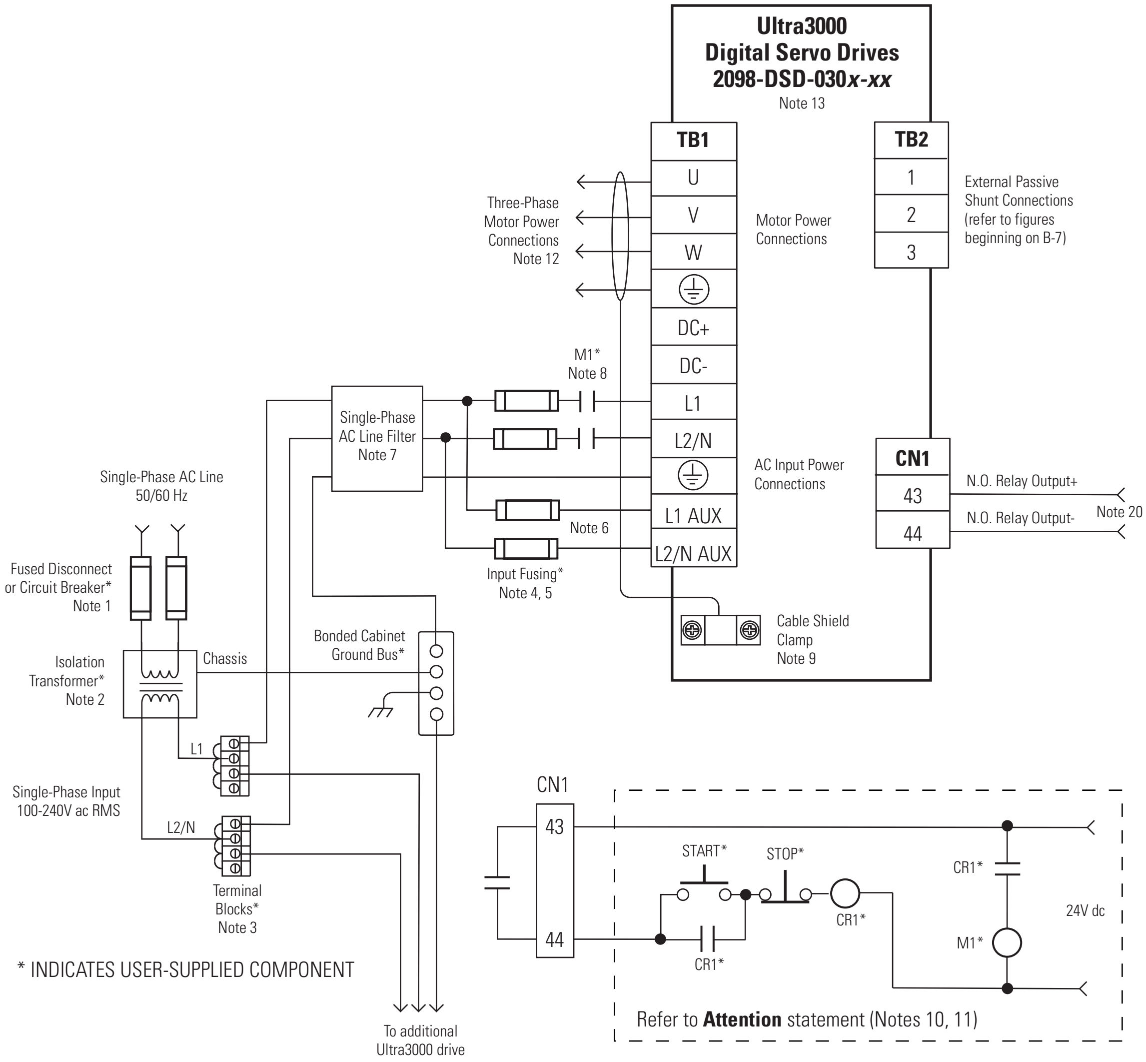


Note: Refer to Figure 2.26 in the chapter *Ultra3000 Connector Data* for more information on the relay output.

The Ultra3000 (2098-DSD-030*x-xx*) power wiring with 24V dc control string (non-SERCOS drives only) is shown in the figure below. For the control string diagram with 120V ac input refer to Figure B.17.

For SERCOS drives, input line contactor is part of the PLC program and output control.

Figure B.2
Typical Power Wiring of Ultra3000 System
(2098-DSD-030*x-xx*)

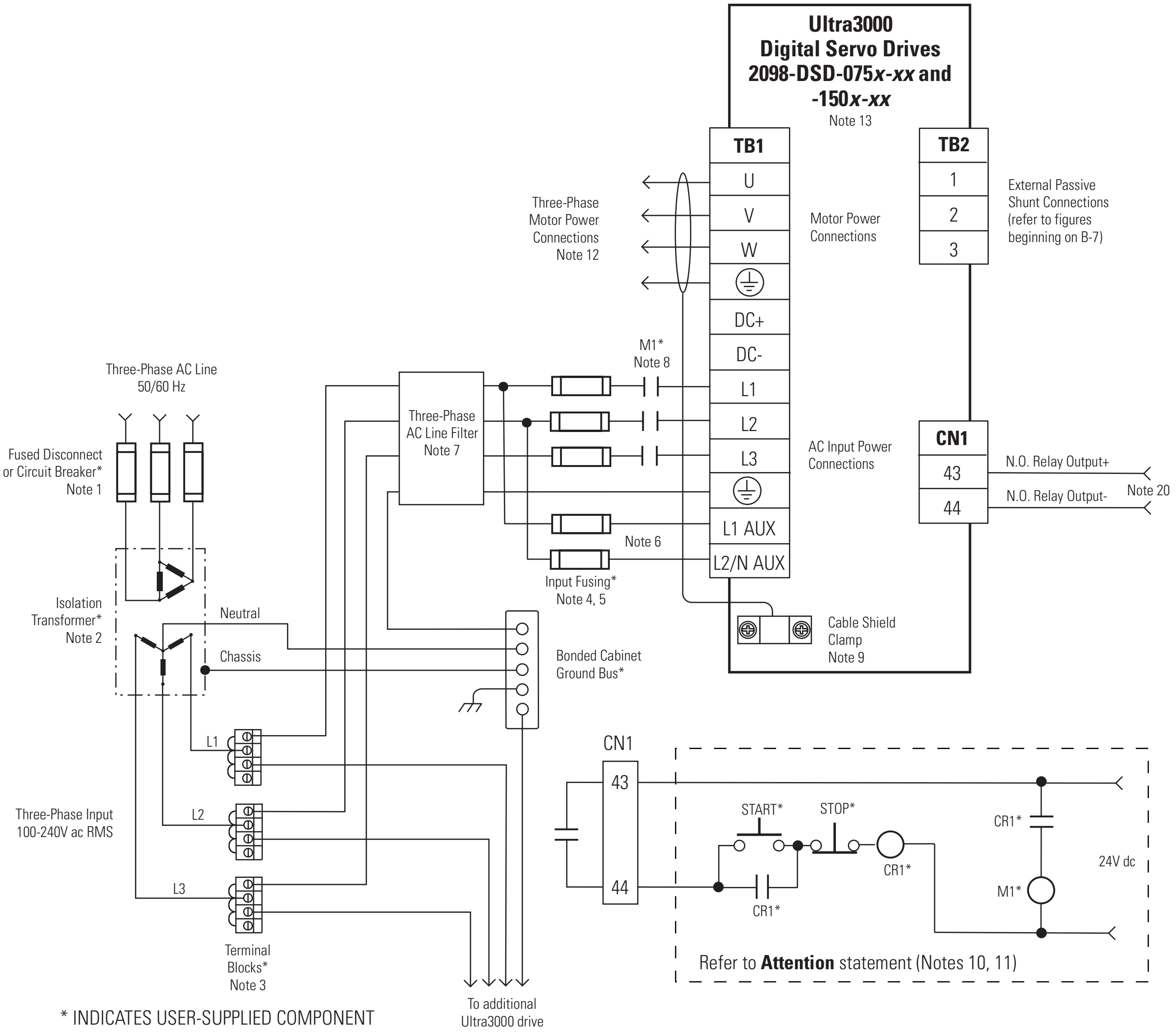


Note: Refer to Figure 2.26 in the chapter *Ultra3000 Connector Data* for more information on the relay output.

The Ultra3000 (2098-DSD-075*x-xx* and -150*x-xx*) power wiring with 24V dc control string (non-SERCOS drives only) is shown in the figure below. For the control string diagram with 120V ac input refer to Figure B.18.

For SERCOS drives, input line contactor is part of the PLC program and output control.

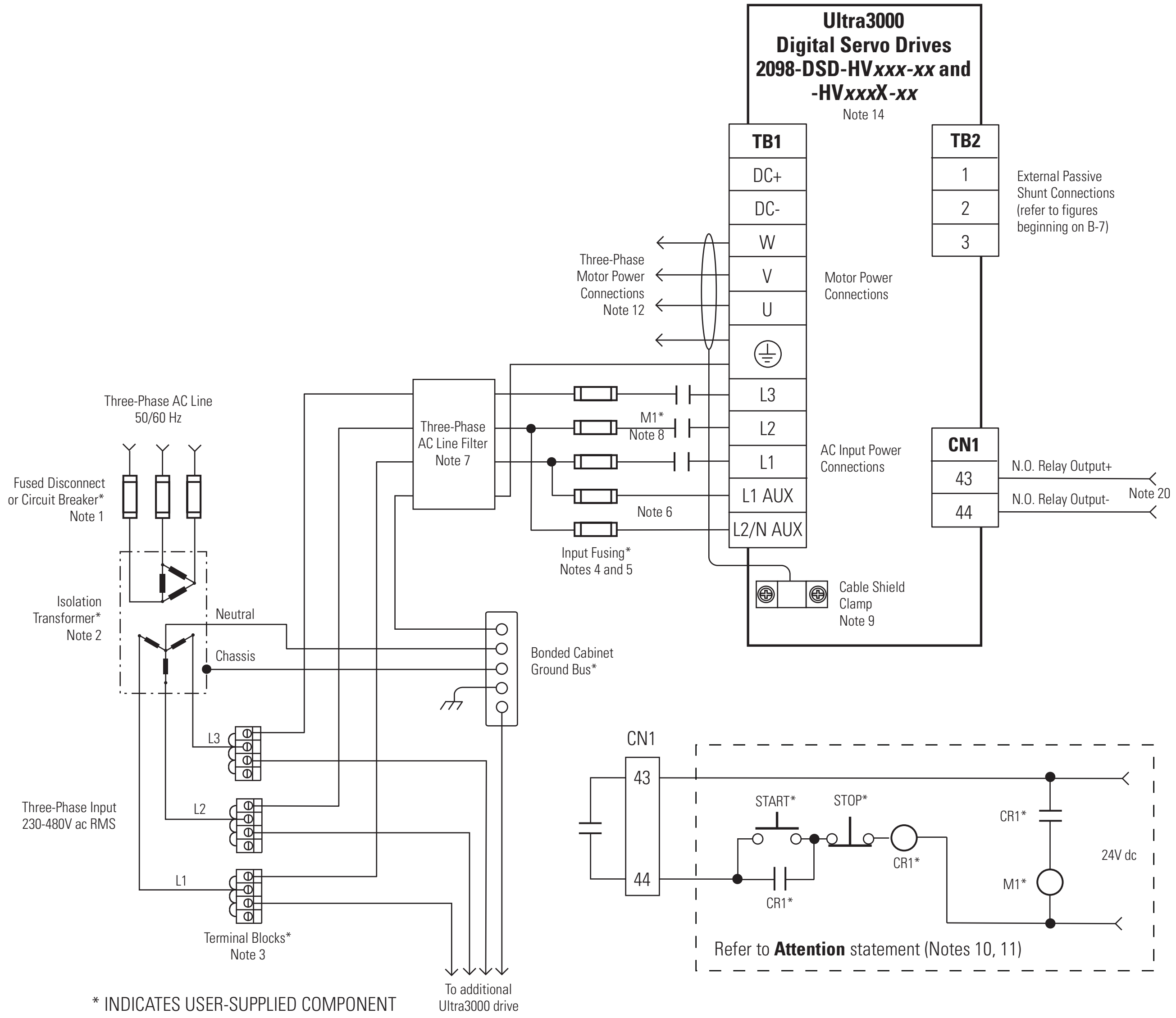
Figure B.3
Typical Power Wiring of Ultra3000 System (2098-DSD-075*x-xx* and -150*x-xx*)



Note: Refer to Figure 2.26 in the chapter *Ultra3000 Connector Data* for more information on the relay output.

The Ultra3000 (2098-DSD-HV*xxx-xx* and -HV*xxx*X-*xx*) power wiring with 24V dc control string is shown in the figure below. For the control string diagram with 120V ac input refer to Figure B.18.

Figure B.4
Typical Power Wiring of Ultra3000 System
(2098-DSD-HV*xxx-xx* and -HV*xxx*X-*xx*)



Note: Refer to Figure 2.26 in the chapter *Ultra3000 Connector Data* for more information on the relay output.

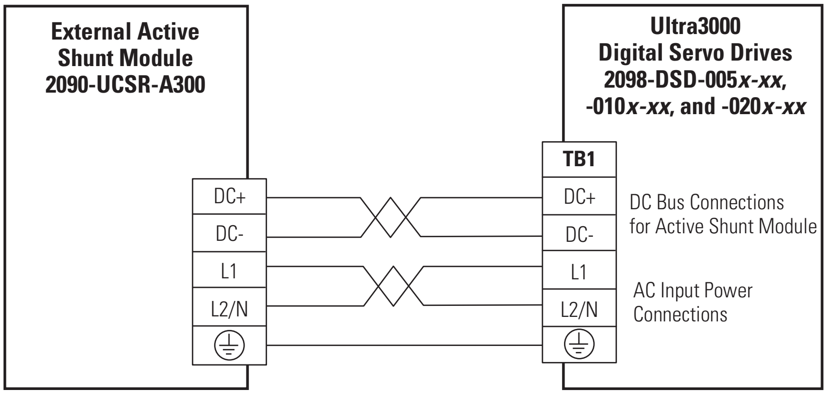
Shunt Module Interconnect Diagrams

This section contains the interconnect diagrams connecting the Ultra3000 drives with active and passive shunt modules. Refer to *External Shunt Kits* in *Appendix C* for Ultra3000/shunt combinations.

Active Shunt Module Diagrams

In the figure below, the Ultra3000 (2098-DSD-005*x-xx*, -010*x-xx*, or -020*x-xx*) is shown wired with the 2090-UCSR-A300 active shunt module.

Figure B.5
External Active Shunt Module Interconnect Diagram

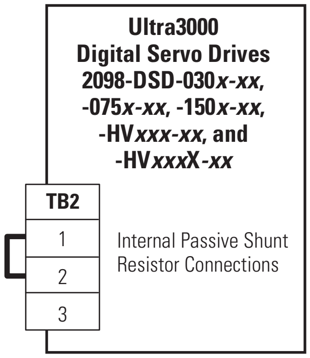


Passive Shunt Module Diagrams

In the Figure B.6, the Ultra3000 is shown wired for internal shunt operation. This is the factory default jumper setting.

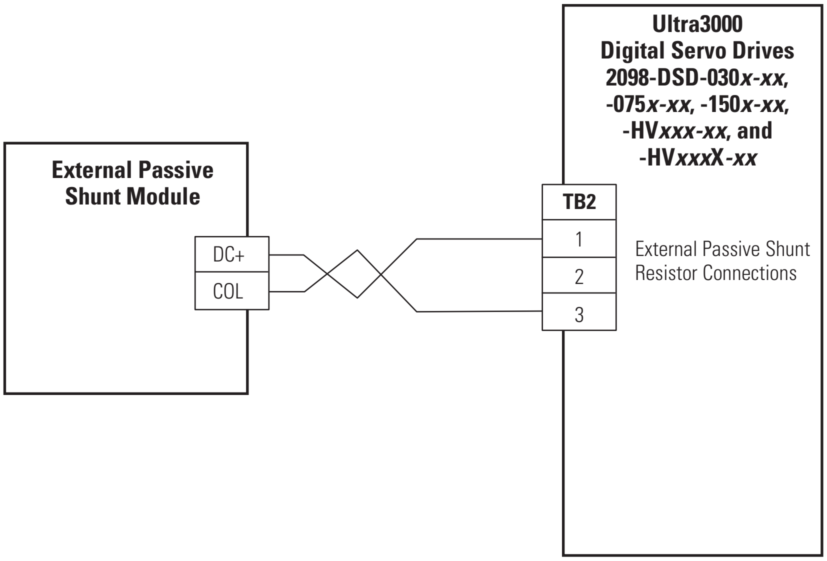
IMPORTANT Internal shunt operation is only present on the drives listed in the figure below.

Figure B.6
Internal Shunt Interconnect Diagram



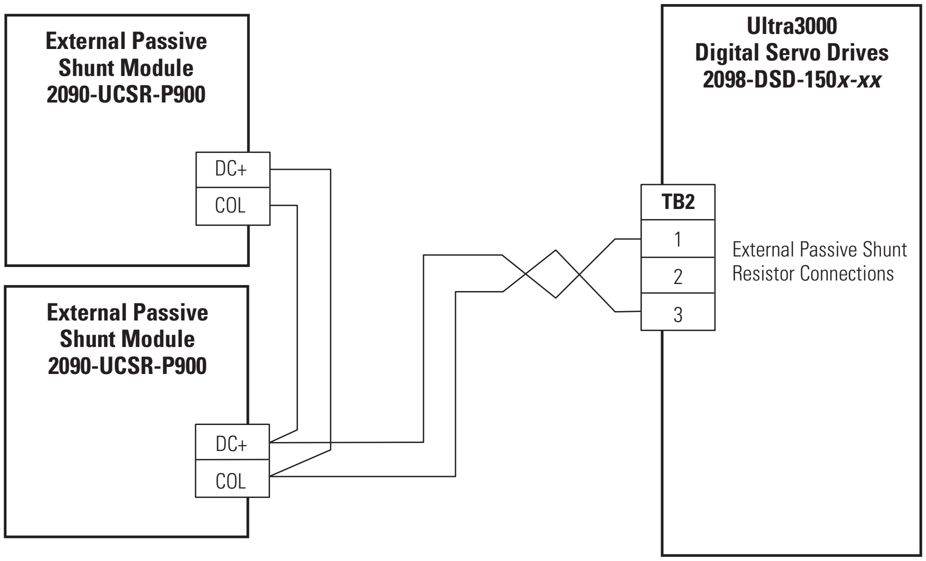
In the figure below, the Ultra3000 is shown wired with an external passive shunt resistor. Refer to *External Shunt Kits* in *Appendix C* for Ultra3000/shunt combinations.

Figure B.7
External Passive Shunt Module Interconnect Diagram



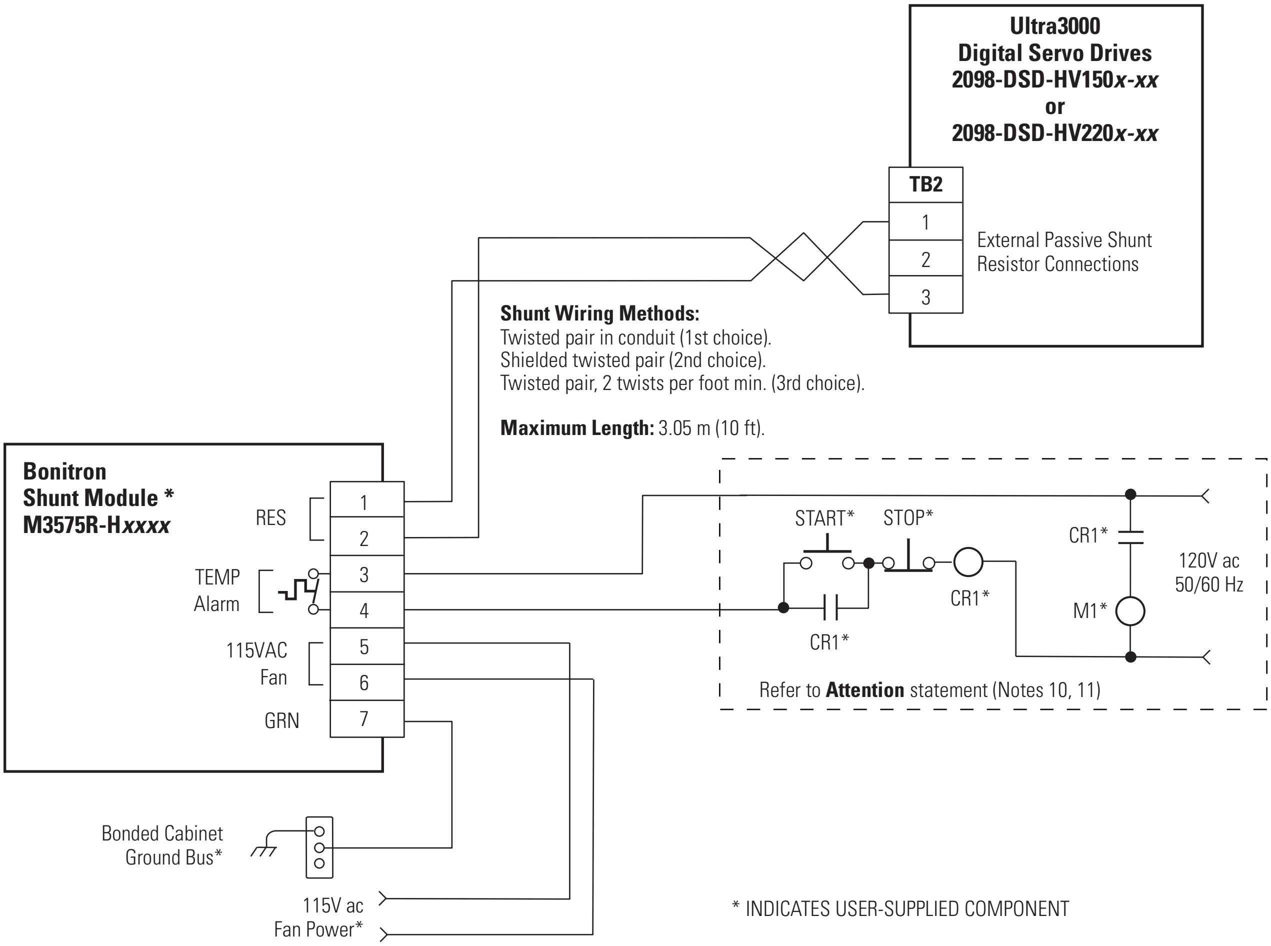
In the figure below, the Ultra3000 (2098-DSD-150*x-xx*) is shown wired with two external passive shunt resistors. When two 900W shunt modules are connected in parallel, the shunt capacity is doubled for a total of 1800W of continuous power dissipation.

Figure B.8
External Passive Shunt Module Interconnect Diagram



In the figure below, the Ultra3000 (2098-DSD-HV150*x-xx* or -HV220*x-xx*) is shown wired to a Bonitron shunt module.

Figure B.9
External Passive Shunt Module Interconnect Diagram

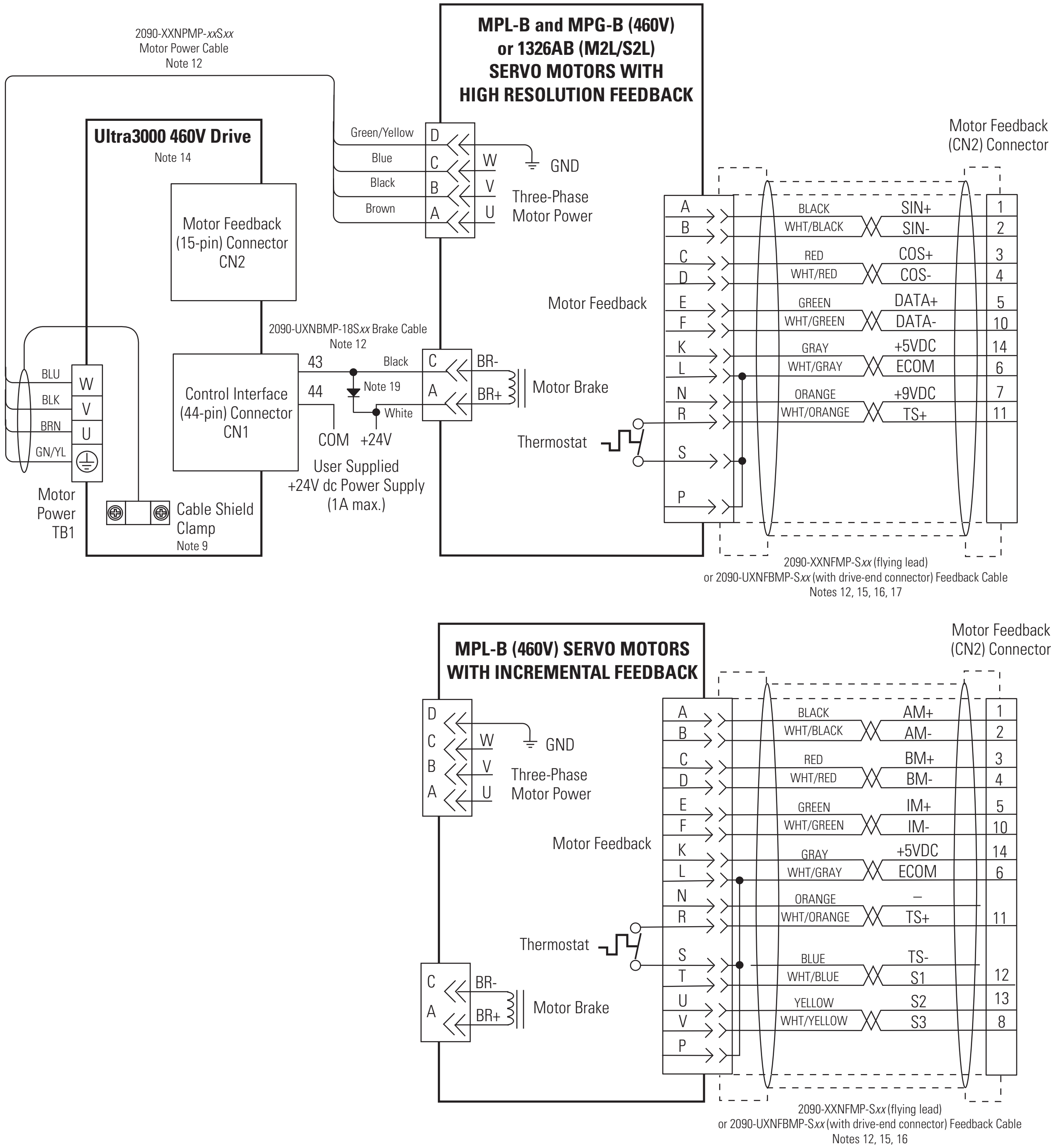


Ultra3000/Motor Interconnect Diagrams

This section contains the motor power, brake, and feedback signal interconnect diagrams between the Ultra3000 and MP-Series, 1326AB-(M2L/S2L), F-, H-, N-, and Y-Series motors.

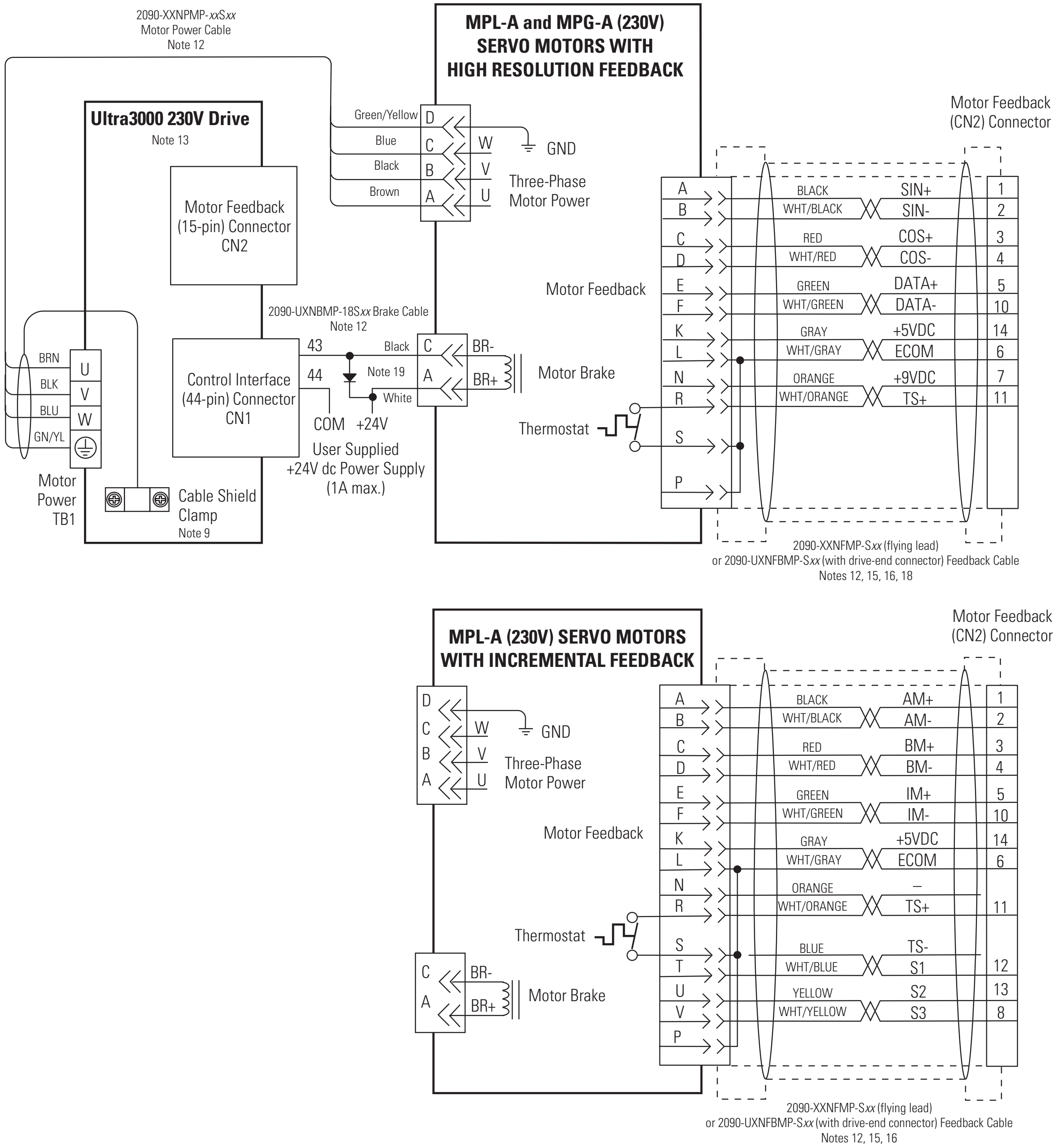
In the figure below, the Ultra3000 (460V) is shown connected to MP-Series or 1326AB (M2L/S2L) servo motors.

Figure B.10
Ultra3000 to MP-Series or 1326AB (M2L/S2L) Motor Configuration



In the figure below, the Ultra3000 (230V) is shown connected to MP-Series (low inertia and integrated gear) 230V servo motors.

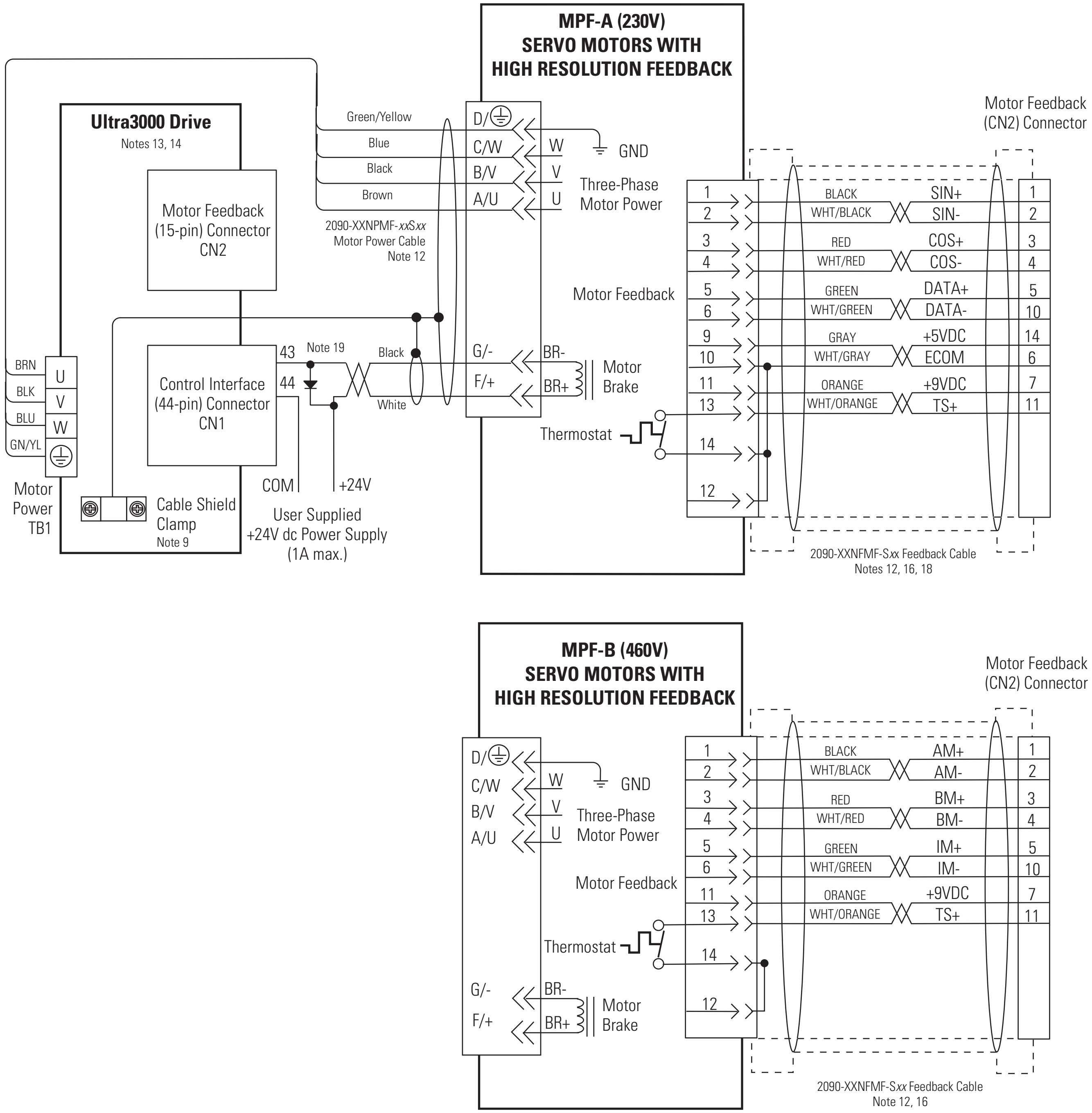
Figure B.11
Ultra3000 to MP-Series (230V) Motor Configuration



IMPORTANT MPL-A5*xxx* motors are not compatible with Ultra3000 (2098-DSD-005, -010, -020) drives.

In the figure below, the Ultra3000 (230V) is shown connected to MP-Series food grade servo motors.

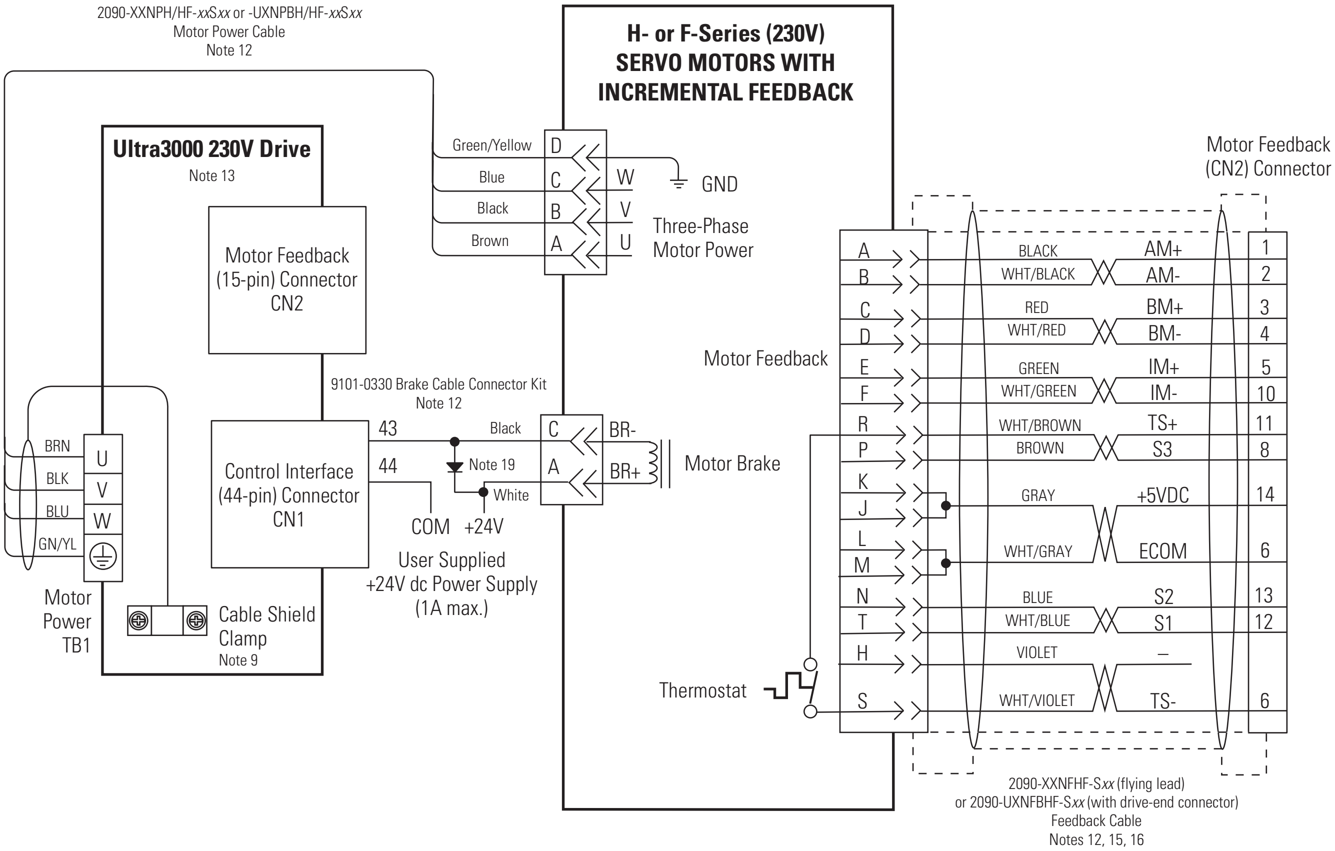
Figure B.12
Ultra3000 to MP-Series Food Grade Motor Configuration



IMPORTANT MPF-A5*xxx* motors are not compatible with Ultra3000 (2098-DSD-005, -010, -020) drives.

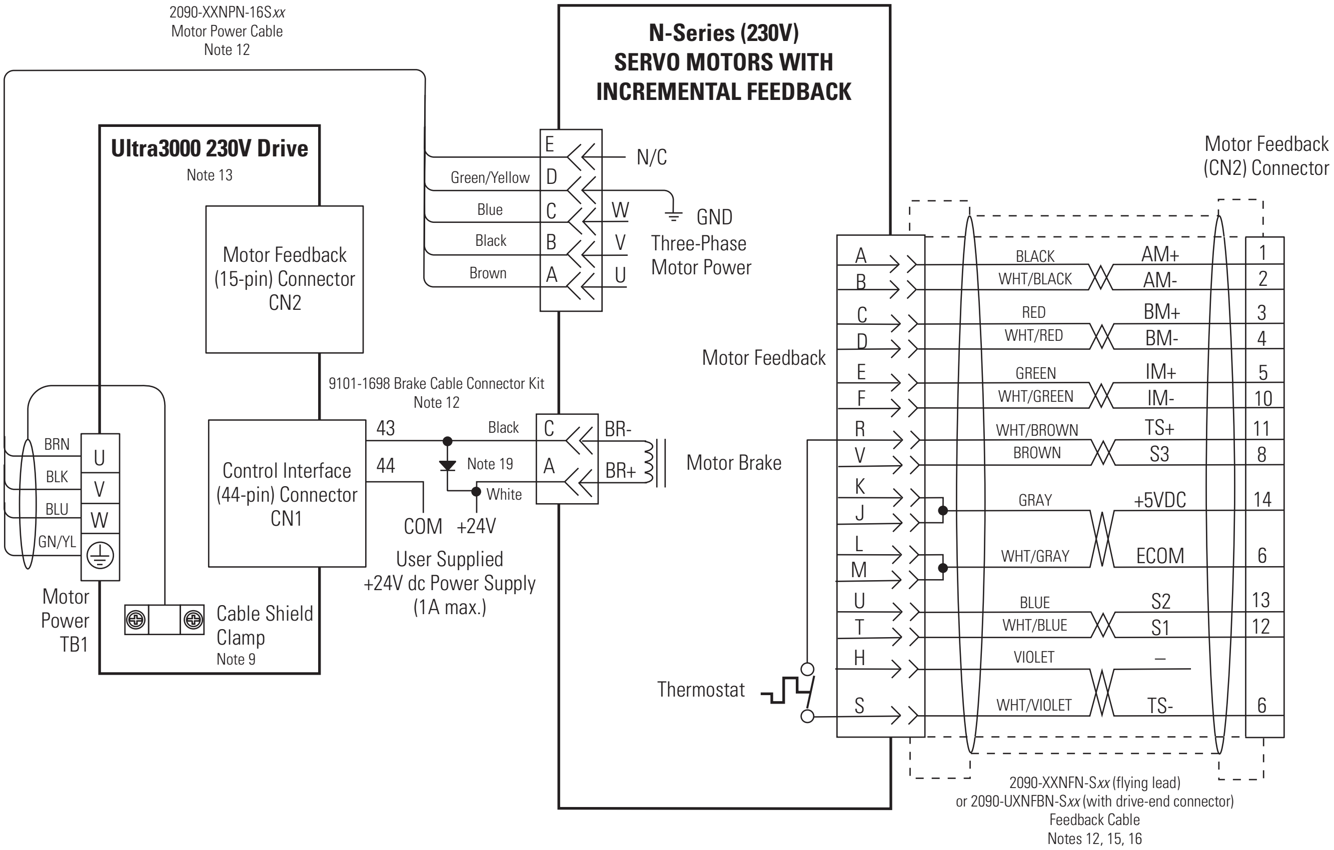
In the figure below, the Ultra3000 (230V) is shown connected to H- and F-Series (230V) servo motors.

Figure B.13
Ultra3000 to H- and F-Series (230V) Motor Configuration



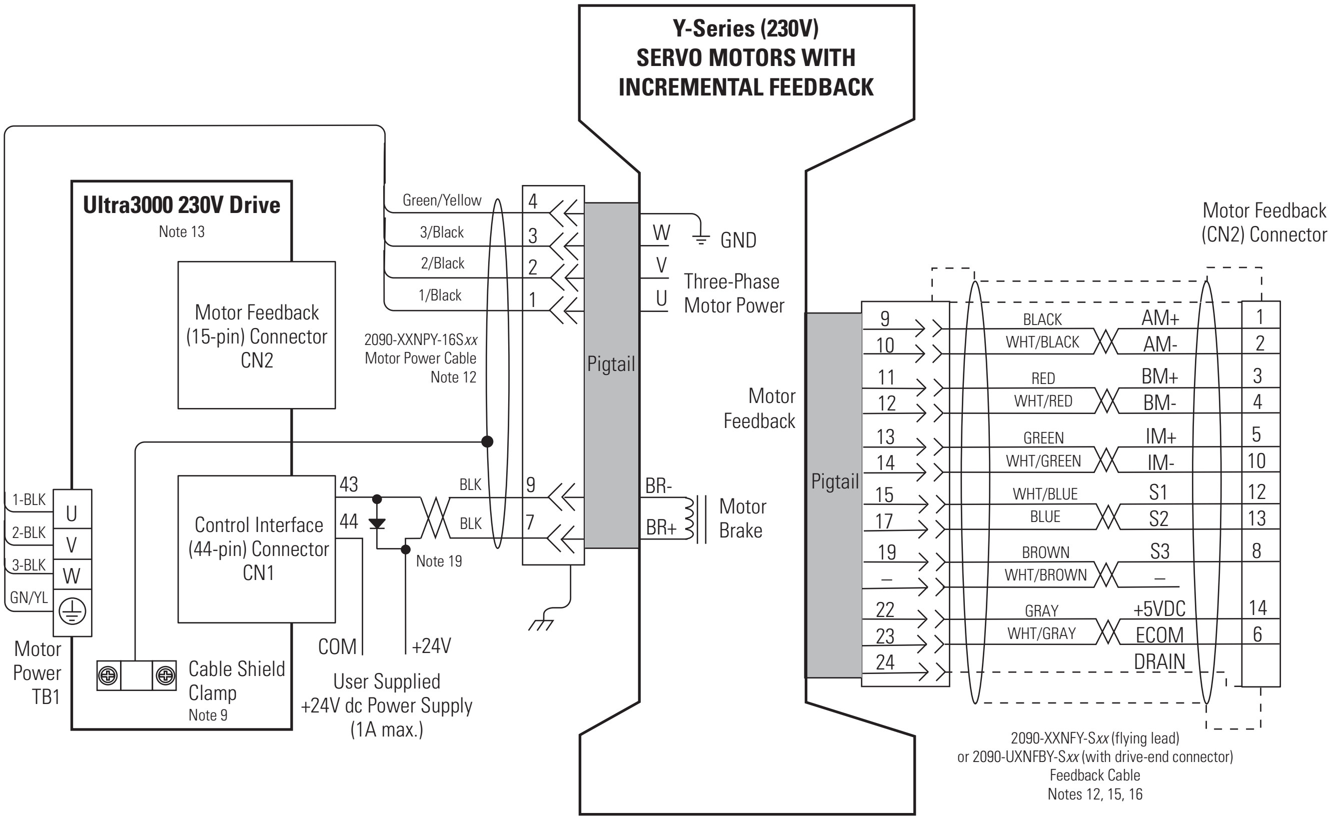
In the figure below, the Ultra3000 (230V) is shown connected to N-Series (230V) servo motors.

Figure B.14
Ultra3000 to N-Series (230V) Motor Configuration



In the figure below, the Ultra3000 (230V) is shown connected to Y-Series (230V) servo motors.

Figure B.15
Ultra3000 to Y-Series (230V) Motor Configuration



Control String Examples (120V ac)

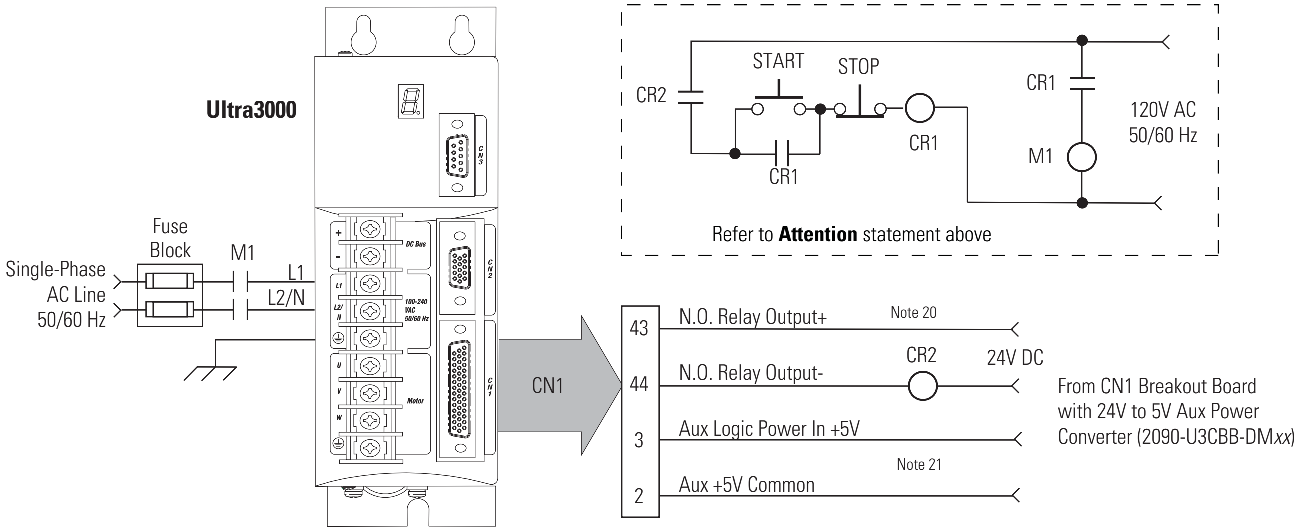
This section provides information to assist you in using the configurable Drive Ready output in a control string with 120V ac input voltage. Refer to Figure 2.26 in the chapter *Ultra3000 Connector Data* for more information on the digital relay output.

The 120V ac control string wired to the Ultra3000 (2098-DSD-005*x-xx*, -010*x-xx*, or -020*x-xx*) drives is shown in the figure below.

ATTENTION

Implementation of safety circuits and risk assessment is the responsibility of the machine builder. Please reference international standards EN1050 and EN954 estimation and safety performance categories. For more information refer to *Understanding the Machinery Directive* (publication SHB-900).

Figure B.16
120V ac Single-Phase Control String Example

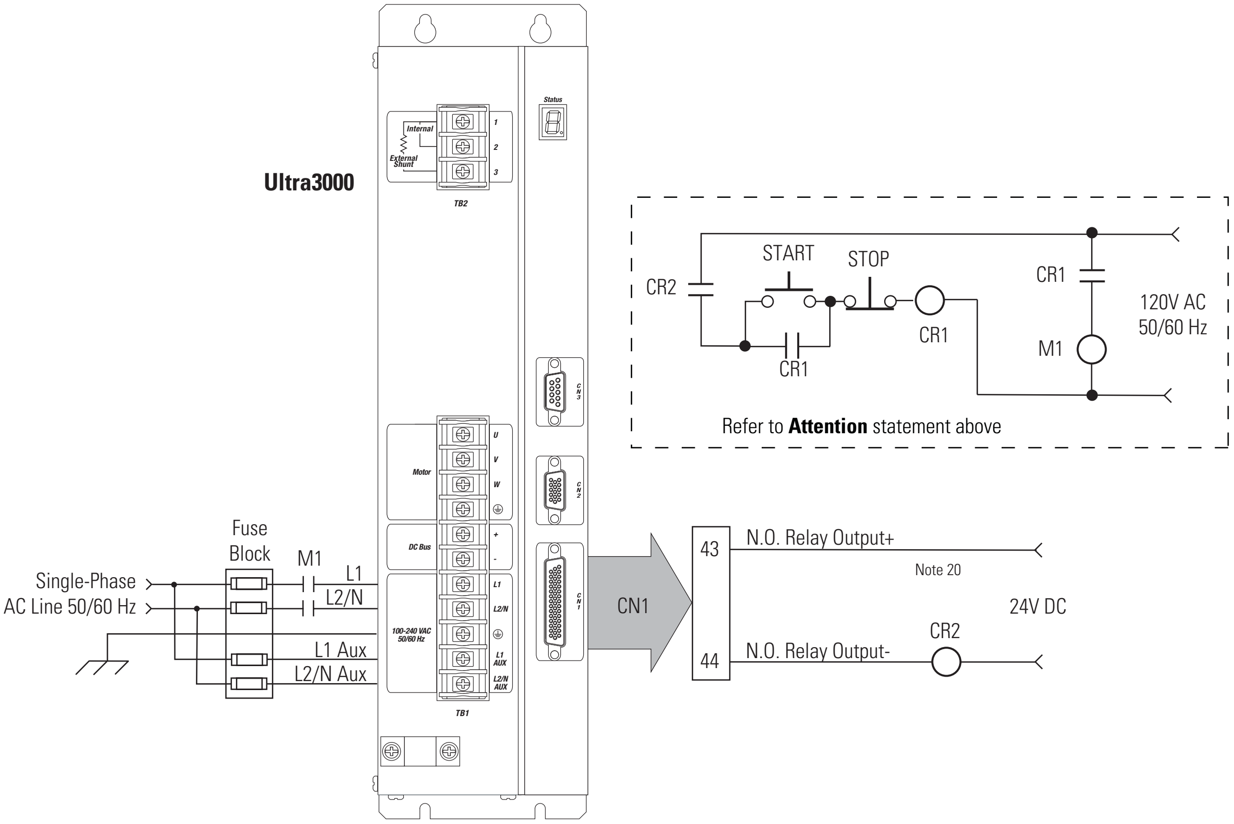


The 120V ac control string wired to the Ultra3000 (2098-DSD-030*x*-*xx*) drives is shown in the figure below.

ATTENTION

Implementation of safety circuits and risk assessment is the responsibility of the machine builder. Please reference international standards EN1050 and EN954 estimation and safety performance categories. For more information refer to *Understanding the Machinery Directive* (publication SHB-900).

Figure B.17
120V ac Single-Phase Control String Example

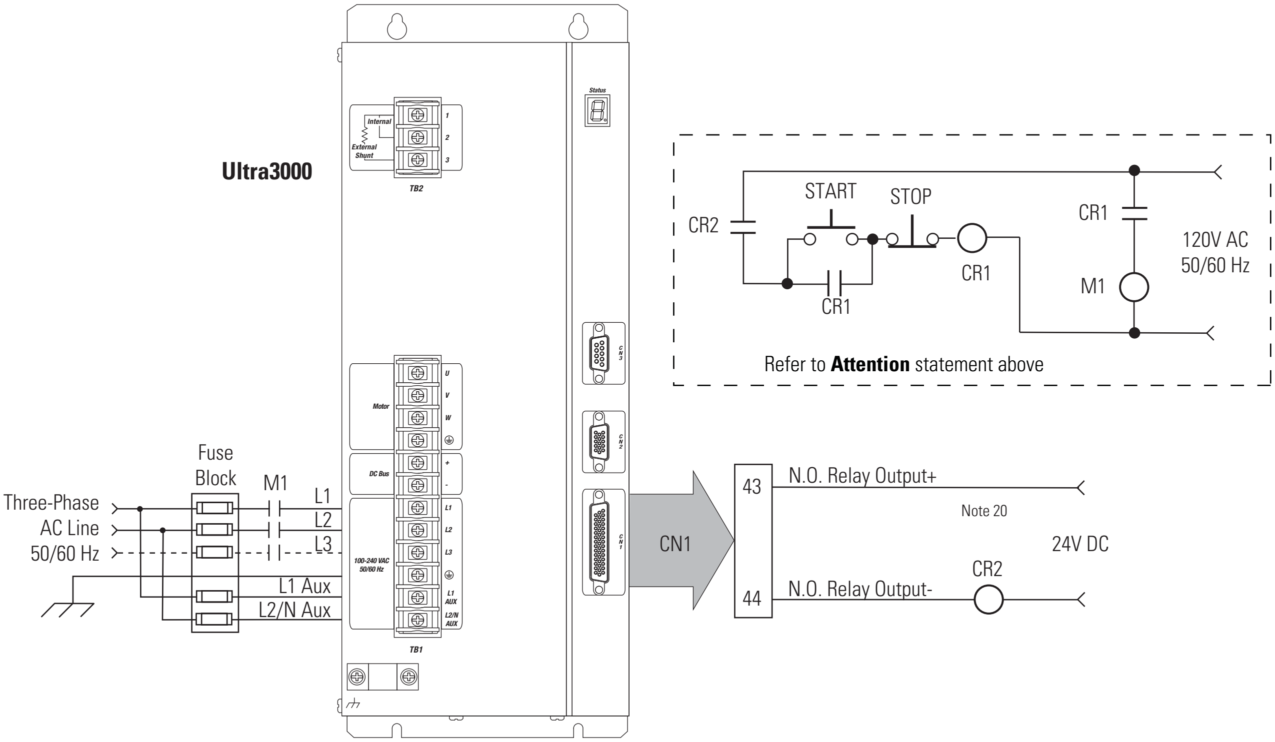


The 120V ac control string wired to the Ultra3000 (2098-DSD-075*x-xx*, -150*x-xx*, -HV*xxx-xx*, and -HV*xxx*X-*xx*) drives is shown in the figure below.

ATTENTION

Implementation of safety circuits and risk assessment is the responsibility of the machine builder. Please reference international standards EN1050 and EN954 estimation and safety performance categories. For more information refer to *Understanding the Machinery Directive* (publication SHB-900).

Figure B.18
120V ac Three-Phase Control String Example



Controlling a Brake Example

The relay output of the Ultra3000 is suitable for directly controlling a motor brake, subject to the relay voltage limit of 30V dc, and the relay current limit of 1A dc. For brake requirements outside of these limits, an external relay must be used. If a transistor output is used, a control relay is also required.

The following table lists Allen-Bradley motors that are compatible with the internal relay output (CN1, pins 43 and 44), when used for controlling a brake. All other motors require an external control relay.

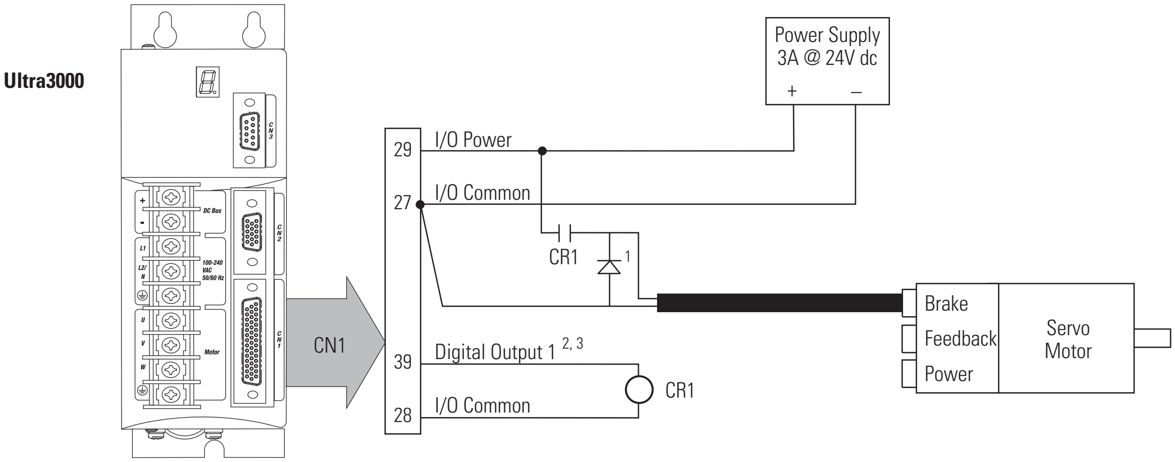
Compatible Brake Motors	Brake Current
F-4030, -4050, and -4075	0.88A
H-3007 and -3016	0.60A
H-4030, -4050, and -4075	0.69A
N-2302, and -2304	0.28A
N-3406, -3412, -4214, and -4220	0.36A
N-5630, -5637, and -5647	0.71A
Y-1002 and -1003	0.26A
Y-2006 and -2012	0.31A
Y-3023	0.37A

Compatible Brake Motors	Brake Current
MPL/MPF-*x*310, -*x*320, -*x*330 [1]	0.50A
MPL-*x*420, -*x*430, -*x*4520, -*x*4530, -*x*4540 [1]	0.64A
MPF-*x*430, -*x*4530, -*x*4540 [1]	
MPG-*x*004 [1]	0.33A
MPG-*x*010 [1]	0.45A
MPG-*x*025 [1]	
MPG-*x*050 [1]	0.50A
MPG-*x*110 [1]	1.0A
1326AB-B4*xxx*	0.88A

[1] Applies to 230V and 460V motors.

Figure B.19 shows an example configuration using Digital Output 1 and an external control relay to control a motor brake which exceeds internal relay ratings.

Figure B.19
Example Configuration Controlling a Motor Brake



[1] Flyback diode (1N4004 rated 1.0A @ 400V dc) suppresses collapsing field of brake coil.

[2] Digital Output 1 (pin 39) configured as Brake in Ultraware software.

[3] For Digital Output 1 specifications, refer to Figure 2.25 in *Chapter 2*.

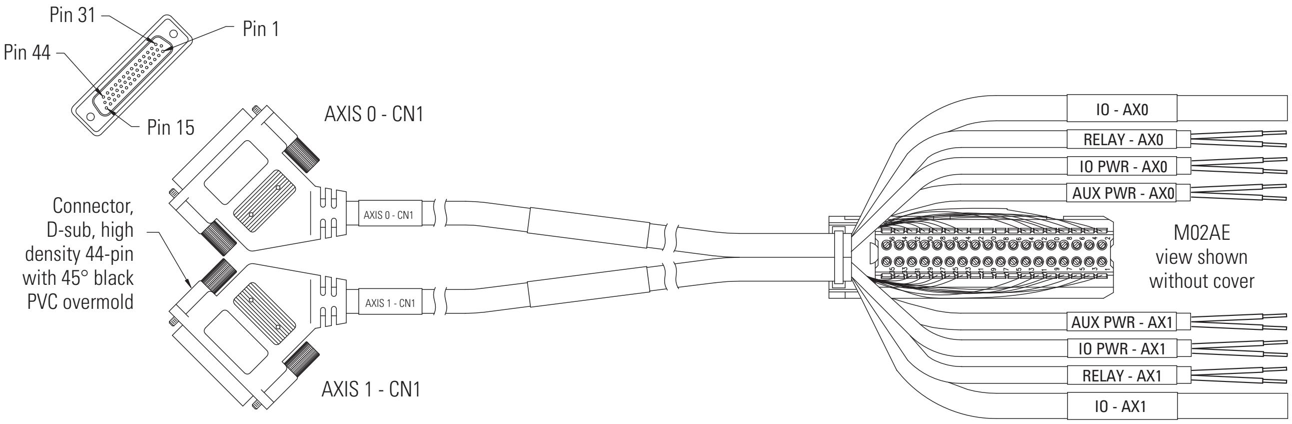
IMPORTANT Flyback diodes must be used when controlling a brake coil with the relay or digital output.

Ultra3000 to Logix Cable and Interconnect Diagrams

This section provides information to assist you in wiring the Ultra3000 CN1 (44-pin) cable connector with either the ControlLogix 1756-M02AE servo module or SoftLogix™ 1784-PM02AE motion card.

Use the 2090-U3AE-D44*xx* control interface cable (shown below) when connecting two Ultra3000 drives to the 1756-M02AE servo module. This cable includes the 1756-TBCH pre-wired terminal block. Refer to Figure B.22 for the interconnect diagram.

Figure B.20
2090-U3AE-D44*xx* Two Axis Cable



Use the 2090-U3CC-D44*xx* control interface cable (shown below) when connecting a single Ultra3000 drive to either the 1756-M02AE (ControlLogix) servo module or 1784-PM02AE (SoftLogix) PCI card. The 1756-TBCH removable terminal block is required when wiring to the ControlLogix module. The 1784-PM02AE-TP*xx* termination panel is required when wiring to the SoftLogix PCI Card. Refer to Figure B.23 for the interconnect diagram.

Figure B.21
Control Interface Cable and Terminations

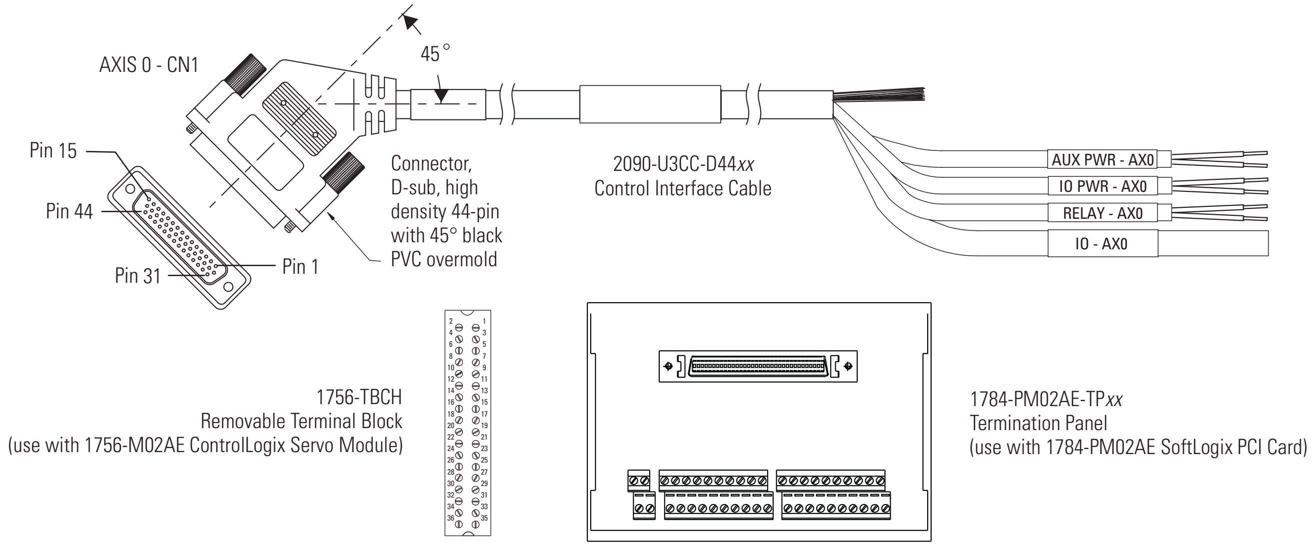
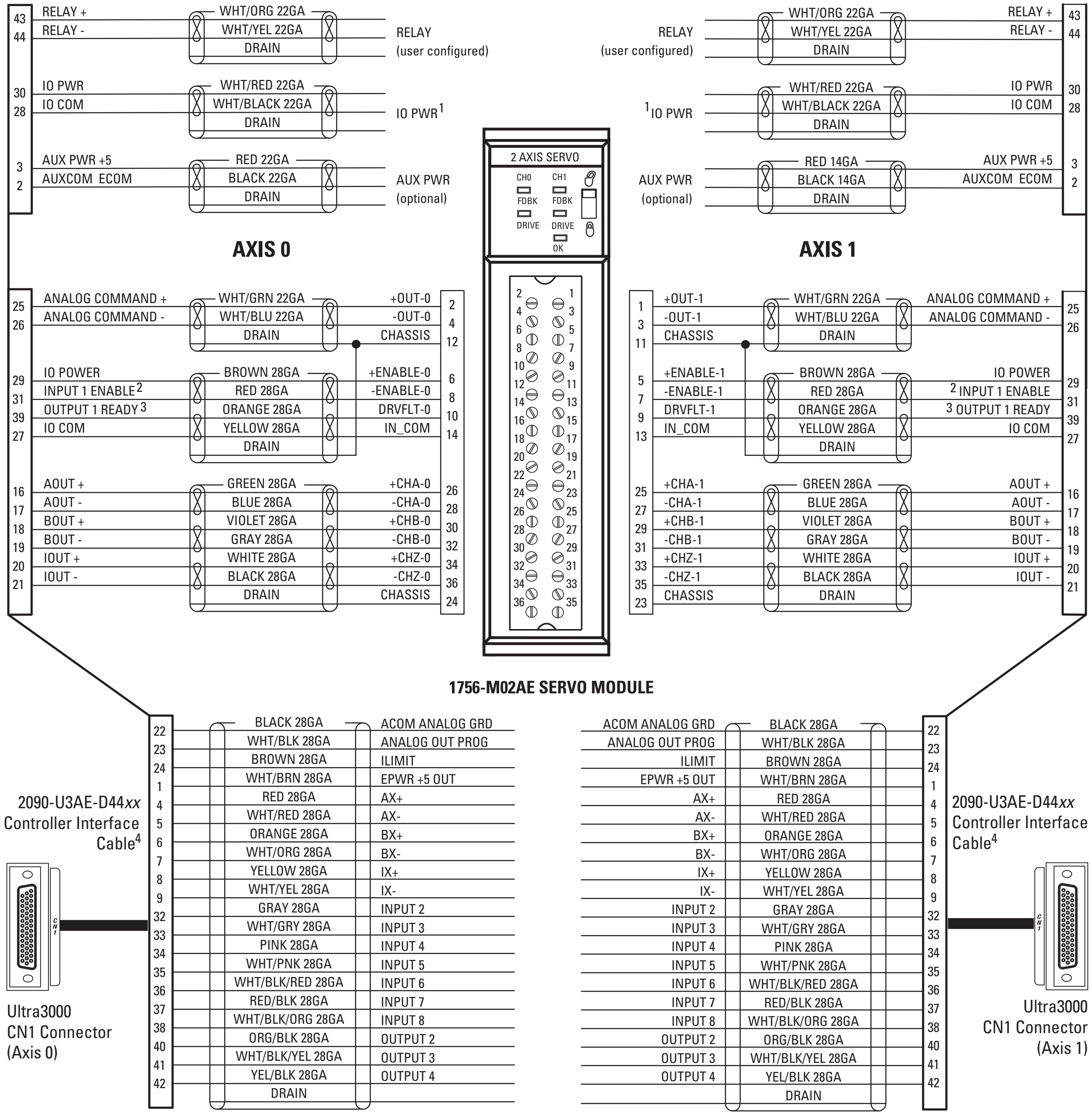


Figure B.22
Ultra3000 to ControlLogix Servo Module Interconnect Diagram



---

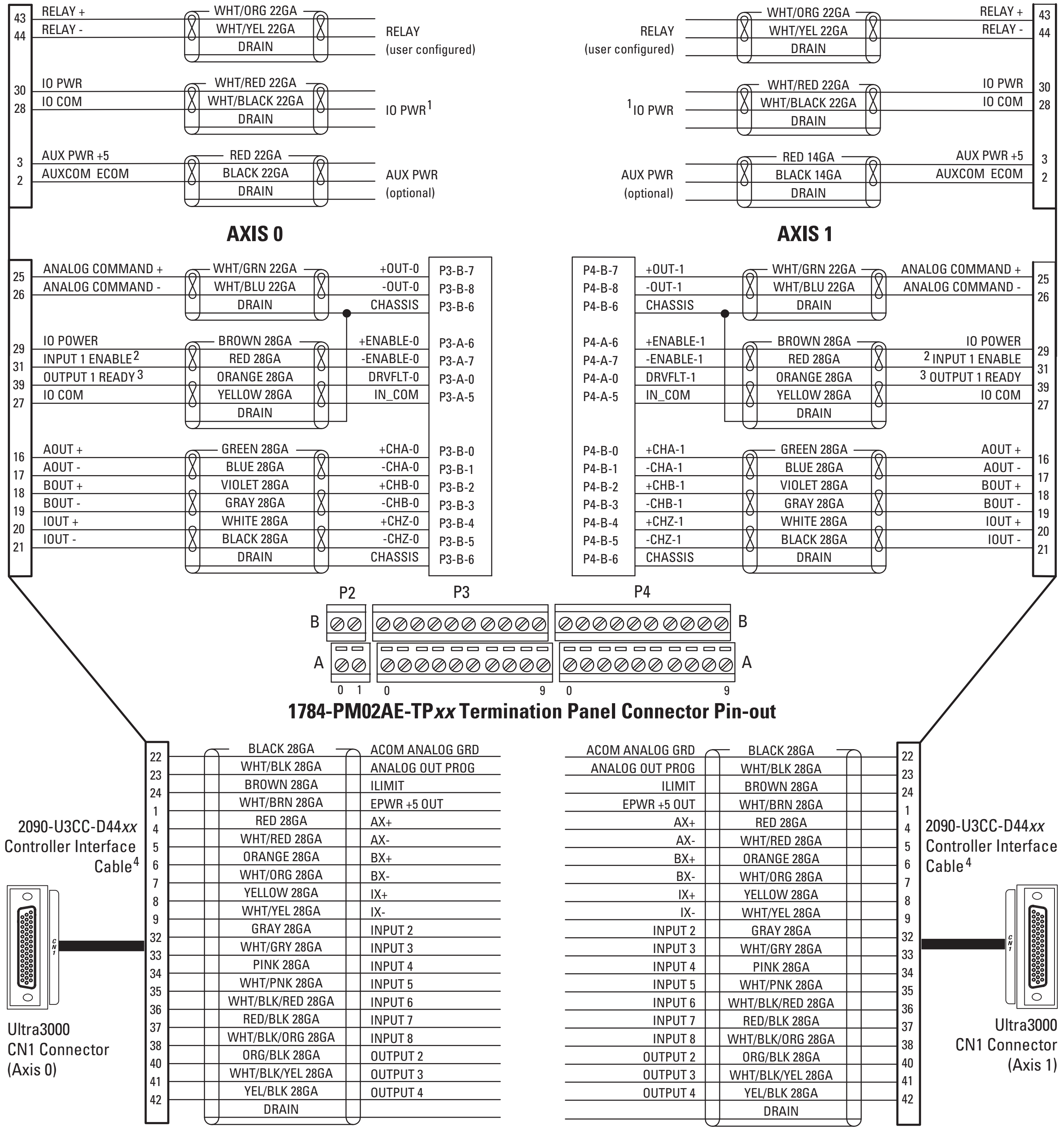
[1] I/O power (pins 28 and 30) must be connected to user-supplied 12-24V dc.

[2] Input 1 (pin 31) must configured as Drive Enable using Ultraware software.

[3] Output 1 (pin 39) must be configured as Ready using Ultraware software.

[4] This cable does not carry the unbuffered motor encoder signals (CN1 pins 10-15). Contact your Allen-Bradley sales representative if these signals are required for your application.

Figure B.23
Ultra3000 to SoftLogix PCI Card Interconnect Diagram



---

[1] I/O power (pins 28 and 30) must be connected to user-supplied 12-24V dc.

[2] Input 1 (pin 31) must configured as Drive Enable using Ultraware software.

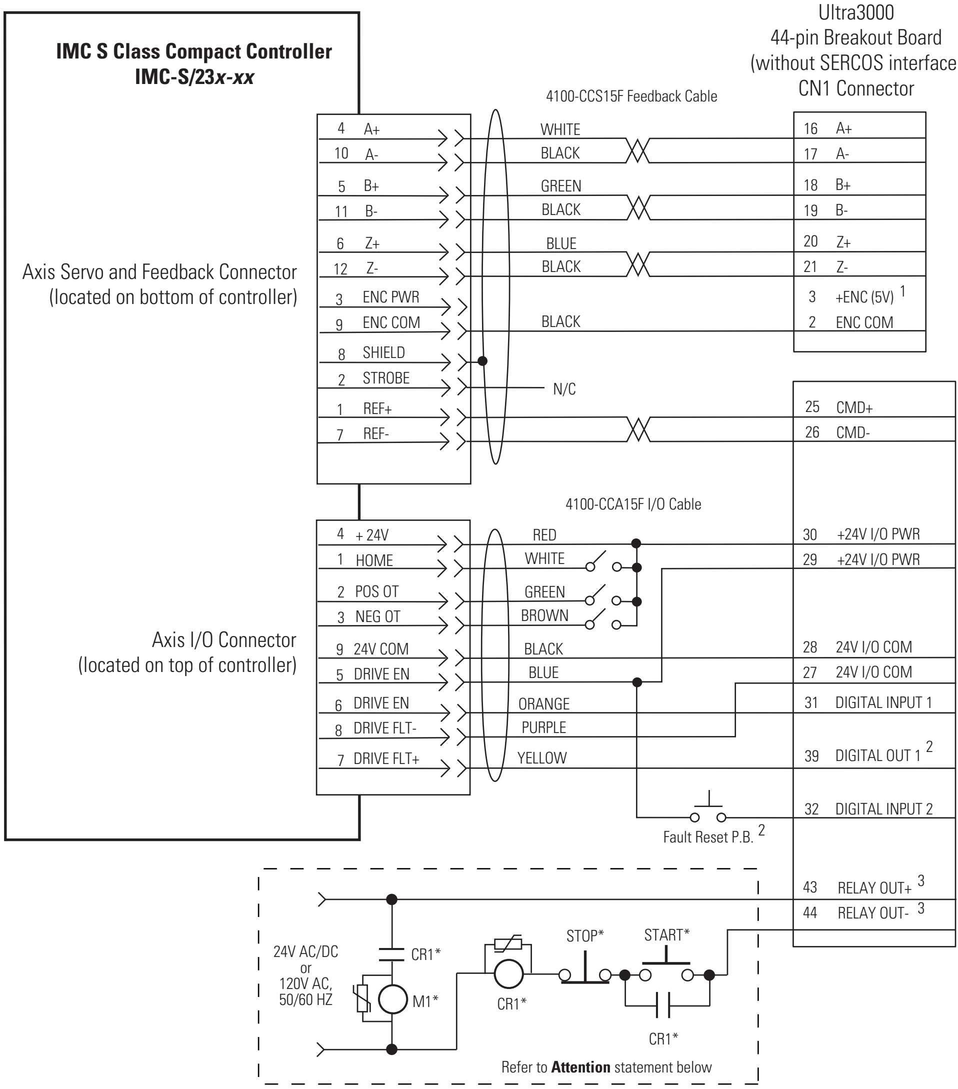
[3] Output 1 (pin 39) must be configured as Ready using Ultraware software.

[4] This cable does not carry the unbuffered motor encoder signals (CN1 pins 10-15). Contact your Allen-Bradley sales representative if these signals are required for your application.

Ultra3000 to IMC-S Compact Cable and Interconnect Diagram

This section provides information to assist you in wiring the IMC-S/23*x-xx* Compact Controller when connecting the 4100-CCS15F feedback cable and 4100-CCA15F I/O cable to your Ultra3000.

Figure B.24
Ultra3000 to IMC-S/23*x-xx* Compact Controller Configuration



[1] The preferred method for supplying the auxiliary +5V is by using the 12- or 44-pin drive mounted breakout board with 24V to 5V auxiliary power converter (catalog number 2090-U3CBB-DM12 or -DM44). Auxiliary +5V power is required to maintain encoder position with an external position controller during a controlled stop condition.

[2] Drive Enable and Fault Reset are configured in Ultraware software.

[3] Relay Output (CN1, pins 43 and 44) must be configured as Ready in Ultraware software.

ATTENTION

Implementation of safety circuits and risk assessment is the responsibility of the machine builder. Please reference international standards EN1050 and EN954 estimation and safety performance categories. For more information refer to *Understanding the Machinery Directive* (publication SHB-900).

Catalog Numbers and Accessories

Chapter Objectives

This appendix lists the Ultra3000 drives and accessory items in tables by catalog number providing detailed descriptions of each component. This appendix describes catalog numbers for:

- Ultra3000 Drives
- Software
- AC Line Filters
- External Shunt Kits
- Motor Power Cables
- Motor Feedback Cables
- MP-Series Motor Brake Cable
- Ultra3000 Interface Cables
- SERCOS Interface Fiber-Optic Cables
- Drive End Connector Kits
- Motor End Connector Kits
- Breakout Boards, Cables, and Kits

Contact your local Allen-Bradley sales office for additional information. Refer to the *Motion Control Selection Guide* (publication GMC-SG001*x*-EN-P) for details on products.

Ultra3000 Drives

Use the following table to identify Ultra3000 240V drives with ratings of 500W, 1 kW, and 2 kW where -*xxx* equals 005, 010, or 020.

Description	Catalog Number
Ultra3000 Digital Servo Drive	2098-DSD-*xxx*
Ultra3000i Digital Servo Drive (with Indexing)	2098-DSD-*xxx*X
Ultra3000 Digital Servo Drive (with SERCOS)	2098-DSD-*xxx*-SE
Ultra3000 Digital Servo Drive (with DeviceNet)	2098-DSD-*xxx*-DN
Ultra3000 Digital Servo Drive (with DeviceNet and Indexing)	2098-DSD-*xxx*X-DN

Use the following table to identify Ultra3000 240V drives with ratings of 3 kW, 7.5 kW, and 15 kW where -*xxx* equals 030, 075, or 150.

Description	Catalog Number
Ultra3000 Digital Servo Drive	2098-DSD-*xxx*
Ultra3000i Digital Servo Drive (with Indexing)	2098-DSD-*xxx*X
Ultra3000 Digital Servo Drive (with SERCOS)	2098-DSD-*xxx*-SE
Ultra3000 Digital Servo Drive (with DeviceNet)	2098-DSD-*xxx*-DN
Ultra3000 Digital Servo Drive (with DeviceNet and Indexing)	2098-DSD-*xxx*X-DN

Use the following table to identify Ultra3000 460V drives with ratings of 3 kW, 5 kW, 10 kW, 15 kW, and 22 kW where -*xxx* equals 030, 050, 100, 150, or 220.

Description	Catalog Number
Ultra3000 Digital Servo Drive	2098-DSD-HV*xxx*
Ultra3000i Digital Servo Drive (with Indexing)	2098-DSD-HV*xxx*X
Ultra3000 Digital Servo Drive (with SERCOS)	2098-DSD-HV*xxx*-SE
Ultra3000 Digital Servo Drive (with DeviceNet)	2098-DSD-HV*xxx*-DN
Ultra3000 Digital Servo Drive (with DeviceNet and Indexing)	2098-DSD-HV*xxx*X-DN

Software

The Ultra3000-SE drives are configured using RSLogix 5000. Ultra3000, Ultra3000 with indexing, Ultra3000-DN, and Ultra3000-DN with indexing drives are configured using Ultraware.

Both are Windows® based applications that allow drive configuration to be done off-line and saved to disk.

Description	Catalog Number
RSLogix 5000 Software (version 11.0 or above)	9324-RLD300ENE
Ultraware Software	2098-UWCPRG

AC Line Filters

Use the following table to identify the AC Line Filter for your application.

AC Line Filter Description	AC Line Filter Fuse Block	Catalog Number
AC Line Filter (single-phase, 240V)	6 Amp	2090-UXLF-106
	10 Amp	2090-UXLF-110
	23 Amp	2090-UXLF-123
	32 Amp	2090-UXLF-132
	36 Amp	2090-UXLF-136
	50 Amp	2090-UXLF-150
AC Line Filter (three-phase, 240V)	36 Amp	2090-UXLF-336
	50 Amp	2090-UXLF-350
	70 Amp	2090-UXLF-370
AC Line Filter (three-phase, 480V)	23 Amp	2090-UXLF-HV323
	30 Amp	2090-UXLF-HV330
	50 Amp	2090-UXLF-HV350

External Shunt Kits

Use the following table to identify the external shunt kit for your application.

Shunt Module Catalog Number:	Description:	Ultra3000 Drive Compatibility:
2090-UCSR-A300	Active Shunt Module (300W)	2098-DSD-005, -010, -020
9101-1183	Passive Shunt Module (200W)	2098-DSD-030
2090-UCSR-P900	Passive Shunt Module (900W)	2098-DSD-075, -150
2090-SR120-09		2098-DSD-HV030 and -HV050
2090-SR040-09		2098-DSD-HV100
2090-SR040-18	Passive Shunt Module (1800W)	
M3575R-H27B0,C [1]	Passive Shunt Module (900W)	2098-DSD-HV150
M3575R-H27BF,C [1]	Passive Shunt Module (1800W)	
M3575R-H33BF,C [1]	Passive Shunt Module (3600W)	2098-DSD-HV220

[1] Bonitron part number. For more information contact Bonitron, Inc. 521 Fairground Court, Nashville, TN 37211. Tel: (615) 244-2825. www.BONITRON.com.

Cables

Use the following tables to identify motor power, feedback, interface, and brake cables for your Ultra3000 drive. Length of cable *xx* is in meters. Refer to your Allen-Bradley representative for available cable lengths.

Motor Power Cables

Description	**Catalog Number**
H-Series, 230V, non-flex, 16 AWG, straight	2090-XXNPH-16S*xx*
H-Series, 230V, non-flex, 16 AWG, right angle	2090-UXNPAH-16R*xx*
H-Series, 230V, non-flex, 6 AWG, straight	2090-UXNPAH-6S*xx*
H-Series, 230V, non-flex, 6 AWG, right angle	2090-UXNPAH-6R*xx*
H and F-Series, 230V, non-flex, 14 AWG, straight	2090-XXNPHF-14S*xx*
H and F-Series, 230V, non-flex, 14 AWG, right angle	2090-UXNPAHF-14R*xx*
H and F-Series, 230V, non-flex, 10 AWG, straight	2090-UXNPAHF-10S*xx*
H and F-Series, 230V, non-flex, 10 AWG, right angle	2090-UXNPAHF-10R*xx*
H and F-Series, 230V, non-flex, 8 AWG, straight	2090-UXNPAHF-8S*xx*
H and F-Series, 230V, non-flex, 8 AWG, right angle	2090-UXNPAHF-8R*xx*
MP-Series (MPL/MPG), 230/460V, non-flex, 16 AWG, straight	2090-XXNPMP-16S*xx*
1326AB (M2L/S2L), 460V, non-flex, 16 AWG, straight	
MP-Series (MPL/MPG), 230/460V, non-flex, 14 AWG, straight	2090-XXNPMP-14S*xx*
MP-Series (MPL/MPG), 230/460V, non-flex, 10 AWG, straight	2090-XXNPMP-10S*xx*
1326AB (M2L/S2L), 460V, non-flex, 10 AWG, straight	
MP-Series (MPL/MPG), 230/460V, non-flex, 8 AWG, straight	2090-XXNPMP-8S*xx*
MP-Series (MPF), 230/460V, non-flex, 16 AWG, straight	2090-XXNPMF-16S*xx*
MP-Series (MPF), 230/460V, non-flex, 14 AWG, straight	2090-XXNPMF-14S*xx*
MP-Series (MPF), 230/460V, non-flex, 10 AWG, straight	2090-XXNPMF-10S*xx*
N-Series, 230V, non-flex, 16 AWG, straight	2090-XXNPN-16S*xx*
N-Series, 230V, non-flex, 16 AWG, right angle	2090-UXNPAN-16R*xx*
Y-Series, 230V, non-flex, 16 AWG, straight	2090-XXNPY-16S*xx*

Motor Feedback Cables

Description	Catalog Number
H-Series, non-flex, connector at both ends, right angle	2090-UXNFBH-R*xx*
H-Series, non-flex, motor connector to flying leads, right angle	2090-UXNFDH-R*xx*
H and F-Series, non-flex, connector at both ends, straight	2090-UXNFBHF-S*xx*
H and F-Series, non-flex, motor connector to flying leads, straight	2090-XXNFHF-S*xx*
H and F-Series, non-flex, connector at both ends, right angle	2090-UXNFBHF-R*xx*
H and F-Series, non-flex, connector at both ends, right angle, skewed	2090-UXNFBHF-K*xx*
MP-Series (MPL/MPG), non-flex, connector at both ends, straight	2090-UXNFBMP-S*xx*
1326AB-Series (M2L/S2L), non-flex, connector at both ends, straight	
MP-Series (MPL/MPG), non-flex, motor connector to flying leads, straight	2090-XXNFMP-S*xx*
1326AB-Series (M2L/S2L), non-flex, motor connector to flying leads, straight	
MP-Series (MPF), non-flex, motor connector to flying leads, straight	2090-XXNFMF-S*xx*
N-Series, non-flex, connector at both ends, straight	2090-UXNFBN-S*xx*
N-Series, non-flex, motor feedback connector to flying leads, straight	2090-XXNFN-S*xx*
N-Series, non-flex, connector at both ends, right angle	2090-UXNFBN-R*xx*
N-Series, non-flex, connector at both ends, right angle, skewed	2090-UXNFBN-K*xx*
N-Series, non-flex, connector at both ends, right angle, skewed	2090-UXNFBN23-K*xx*
N-Series, non-flex, motor connector to flying leads, right angle, skewed	2090-UXNFDN23-K*xx*
N-Series, non-flex, motor connector to flying leads, right angle	2090-UXNFDN-R*xx*
Y-Series, non-flex, connector at both ends, straight	2090-UXNFBY-S*xx*
Y-Series, non-flex, motor feedback connector to flying leads, straight	2090-XXNFY-S*xx*
Drive Feedback Cable, non-flex, CN2 connector to flying leads	2090-UXNFM-S*xx*

MP-Series Motor Brake Cable

Description	Catalog Number
MP-Series motor brake cable, 0.75 mm^2 (18 AWG)	2090-UXNBMP-18S*xx*

Ultra3000 Interface Cables

Description	Catalog Number
Serial Interface Cable, 9-pin D-shell, CN3 to personal computer.	2090-UXPC-D09*xx*
Controller Interface Cable, 44-pin D-shell, CN1 to 1756-M02AE motion module (two-axis to premolded connector).	2090-U3AE-D44*xx*
Controller Interface Cable, 44-pin D-shell, CN1 to 1756-M02AE motion module (single-axis to no connector).	2090-U3CC-D44*xx*

SERCOS Interface Fiber-Optic Cables

Use the following table to identify the SERCOS interface fiber-optic plastic cables for your Ultra3000 (connectors at both ends).

Description	Catalog Number
SERCOS fiber-optic plastic cable (for use inside enclosure only)	2090-SCEP*x-x*
SERCOS fiber-optic plastic (PVC) cable (for use outside enclosure)	2090-SCVP*x-x*
SERCOS fiber-optic plastic (nylon) cable (for use outside enclosure in harsh environments)	2090-SCNP*x-x*
SERCOS fiber-optic glass (PVC) cable	2090-SCVG*x-x*

Note: Cable length (*x-x*) is in meters. Plastic cable is available in lengths up to 32 m (105.0 ft). Glass cable is available in lengths up to 200 m (656.7 ft).

Drive End Connector Kits

Use the following table to identify the drive-end connector kit for your serial, feedback, and I/O cable. Refer to the chapter *Ultra3000 Connector Data* for pin signal, and wiring information.

Description	Catalog Number
Drive Connector Kit, 9-pin D-shell, serial cable to drive (CN3)	2090-UXCK-D09
Drive Connector Kit, 15-pin high density D-shell, feedback cable to drive (CN2)	2090-UXCK-D15
Drive Connector Kit, 44-pin high density D-shell, I/O cable to drive (CN1)	2090-U3CK-D44

The following table lists Amp drive cable connectors that are not available from Rockwell Automation. Please contact Amp at 1-800-522-6752 or a distributor for additional information.

Connector Components	CN1 Controller 44-Pin High-Density D-Shell	CN2 Motor Feedback 15-Pin High-Density D-Shell	CN3 Serial 9-Pin Standard-Density D-Shell
Drive Connector	748366-1	748364-1	205204-4
Crimp Pin Contacts	748333-4 [1]	748333-4 [1]	5-66506-7 [2]
Unshielded Backshell Kit	748678-3	748678-1	748678-1
Shielded Backshell Kit	745173-3	745171-5	745171-5
Ferrules [3]	745508-1	745508-6	745508-6

[1] Accepts 22-28 AWG wire.

[2] Accepts 20-24 AWG wire.

[3] Ferrules are only required for use with shielded backshell kits.

Motor End Connector Kits

Use the following table to identify the motor-end connector kit for your motor power, feedback, and brake cable.

Motor Series	Motor Connector Kit	Description
F-4000	9101-0326	Straight Power Connector Kit
	9101-0399	Right-Angle Power Connector Kit
F-6000	9101-0327	Straight Power Connector Kit
	9101-0400	Right-Angle Power Connector Kit
F-Series	9101-0329	Straight Feedback Connector Kit
	9101-0330	Straight Brake Connector Kit
	9101-0402	Right-Angle Feedback Connector Kit
	9101-0403	Right-Angle Brake Connector Kit
H-2000	9101-0325	Straight Power Connector Kit
	9101-0398	Right-Angle Power Connector Kit
H-3000	9101-0325	Straight Power Connector Kit
	9101-0398	Right-Angle Power Connector Kit
H-4000	9101-0326	Straight Power Connector Kit
	9101-0399	Right-Angle Power Connector Kit
H-6000	9101-0327	Straight Power Connector Kit
	9101-0400	Right-Angle Power Connector Kit
H-8000	9101-0328	Straight Power Connector Kit
	9101-0401	Right-Angle Power Connector Kit
H-Series	9101-0329	Straight Feedback Connector Kit
	9101-0330	Straight Brake Connector Kit
	9101-0402	Right-Angle Feedback Connector Kit
	9101-0403	Right-Angle Brake Connector Kit
MP-Series and 1326AB (M2L/S2L)	2090-MPPC-S	Straight Power Connector Kit
	2090-MPPC-08S	Straight Power Connector Kit, 10 mm^2 (8 AWG)
	2090-MPFC-S	Straight Feedback Connector Kit
	2090-MPBC-S	Straight Brake Connector Kit
N-Series	9101-1557	Straight Power Connector Kit
	9101-1558	Straight Feedback Connector Kit
	9101-1698	Brake Connector Kit
Y-Series	9106-0066	Straight Power and Feedback Connector Kit

Breakout Board Kits

Breakout board kits include the 2090-U*x*BB-D*xx* DIN rail terminal block, 2090-U*x*BC-D*xxxx* cable, and mounting hardware.

Catalog Number	Description
2090-UXBK-D15*xx*	15-pin, high density D-shell for Ultra3000 CN2 interface connector
2090-U3BK-D44*xx*	44-pin, high density D-shell for Ultra3000 CN1 control interface connector

Breakout Boards

Catalog Number	Description
2090-U3BB-DM12 [1]	12-pin, drive-mounted breakout board for Ultra3000 CN1 connector recommended for use with SERCOS interface applications
2090-U3BB2-DM44 [1, 2]	44-pin, drive-mounted breakout board for Ultra3000 CN1 control interface connector
2090-U3CBB-DM12 [3]	12-pin, drive-mounted breakout board for Ultra3000 CN1 connector recommended for use with SERCOS interface applications with 24V to 5V auxiliary power converter
2090-U3CBB-DM44 [3]	44-pin, drive-mounted breakout board for Ultra3000 CN1 connector with 24V to 5V auxiliary power converter
2090-U3BB-D44	44-pin, high density D-shell for Ultra3000 CN1 control interface connector
2090-UXBB-DM15 [4]	15-pin, drive-mounted breakout board for Ultra3000/5000 CN2 interface connector
2090-UXBB-D15	15-pin, high-density D-shell for Ultra3000/5000 CN2 interface connector
2090-UXBB-DM09	9-pin, drive-mounted breakout board for Ultra3000/5000 CN3 serial interface

[1] For specifications, refer to the *CN1 Control Interface Breakout Boards Installation Instructions* (publication 2090-IN007*x*-EN-P).

[2] This breakout board accepts 1 - 0.14 mm^2 (16 - 26 AWG) wire. For applications that require a 44-pin drive-mounted breakout board that accepts 4 - 0.5 mm^2 (12 - 22 AWG) wire, contact your local Allen-Bradley representative.

[3] Only for use with the Ultra3000 (2098-DSD-005*x-xx*, -010*x-xx*, -020*x-xx*) drives. Requires an external +24V dc power supply (refer to *Auxiliary 5V Logic Supply* on page 2-26). For specifications, refer to the *CN1 Control Interface Breakout Boards with Integral 24V to 5V Auxiliary Power Converter Installation Instructions* (publication 2090-IN008*x*-EN-P).

[4] For specifications, refer to the *CN2 Motor Feedback Breakout Board Installation Instructions* (publication 2090-IN006*x*-EN-P).

Breakout Cables

Catalog Number	Description
2090-UXBC-D15*xx*	15-pin, high-density D-shell for Ultra3000 CN2 interface connector
2090-U3BC-D44*xx*	44-pin, high density D-shell for Ultra3000 CN1 control interface connector

For more information refer to our web site: **www.ab.com/motion**
For Allen-Bradley Technical Support information refer to: www.ab.com/support or Tel: (1) 440.646.5800

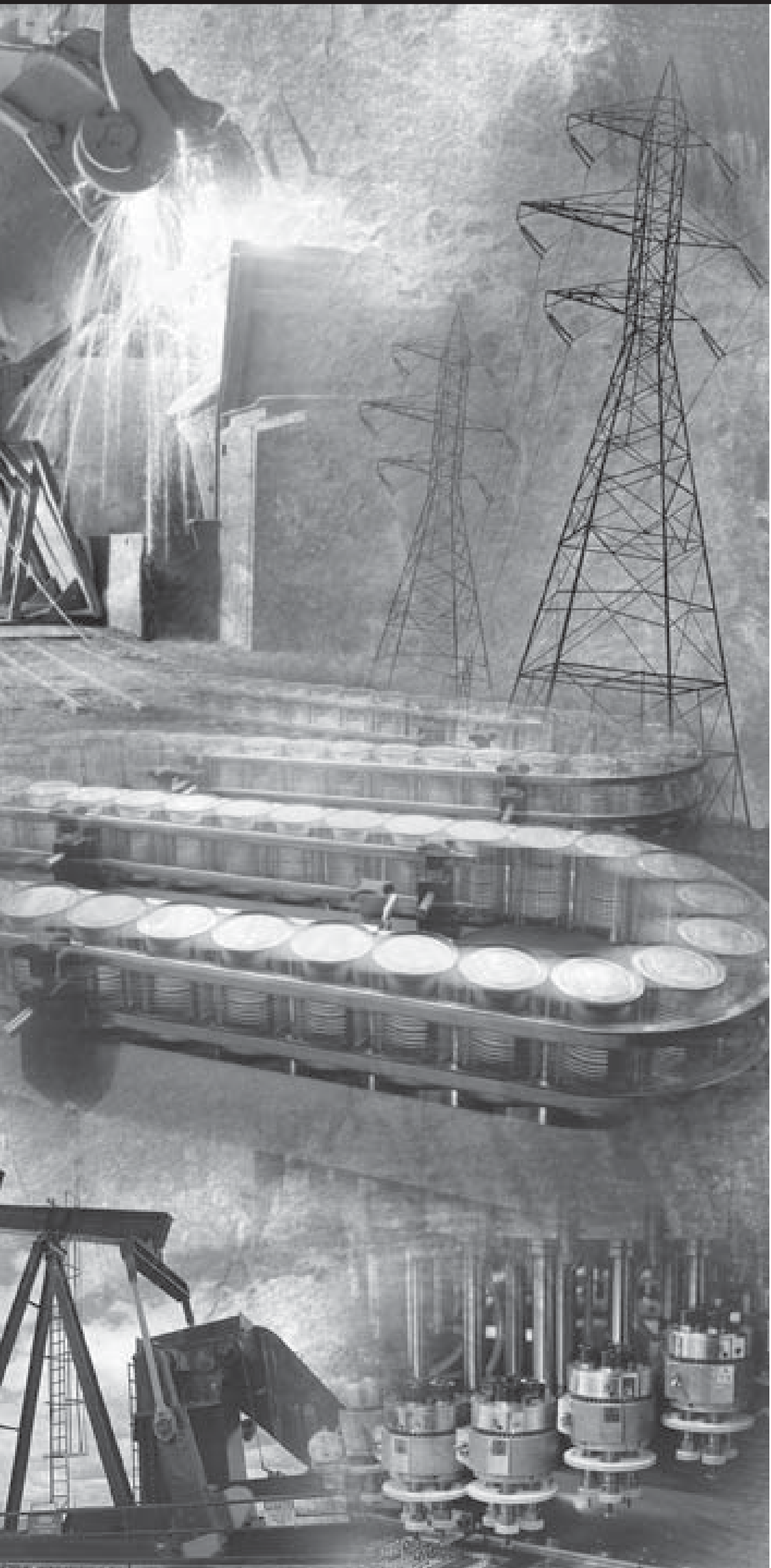
www.rockwellautomation.com

Power, Control and Information Solutions Headquarters

Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
Europe/Middle East/Africa: Rockwell Automation, Vorstlaan/Boulevard du Souverain 36, 1170 Brussels, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia Pacific: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Publication 2098-IN003E-EN-P — April 2004
0013-1083-005-01
Supersedes Publication 2098-IN003D-EN-P — May 2003

ANEXO F. Ultra3000 Digital Servo Drives Integration Manual.



Allen-Bradley

Ultra3000
Digital Servo Drives

Catalog Numbers
2098-DSD-005, -010, and -020
2098-DSD-*xxx*X
2098-DSD-*xxx*-SE
2098-DSD-*xxx*-DN
2098-DSD-*xxx*X-DN

2098-DSD-030, -075, and -150
2098-DSD-*xxx*X
2098-DSD-*xxx*-SE
2098-DSD-*xxx*-DN
2098-DSD-*xxx*X-DN

2098-DSD-HV030, -HV050, -HV100, -HV150, and -HV220
2098-DSD-HV*xxx*X
2098-DSD-HV*xxx*-SE
2098-DSD-HV*xxx*-DN
2098-DSD-HV*xxx*X-DN

Integration Manual

Rockwell Automation

ALLEN-BRADLEY • ROCKWELL SOFTWARE

Important User Information

Solid state equipment has operational characteristics differing from those of electromechanical equipment. Safety Guidelines for the Application, Installation and Maintenance of Solid State Controls (publication SGI-1.1 available from your local Rockwell Automation sales office or online at http://literature.rockwellautomation.com) describes some important differences between solid state equipment and hard-wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

In no event will Rockwell Automation, Inc. be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.

The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, Rockwell Automation, Inc. cannot assume responsibility or liability for actual use based on the examples and diagrams.

No patent liability is assumed by Rockwell Automation, Inc. with respect to use of information, circuits, equipment, or software described in this manual.

Reproduction of the contents of this manual, in whole or in part, without written permission of Rockwell Automation, Inc., is prohibited.

Throughout this manual, when necessary, we use notes to make you aware of safety considerations.

WARNING	Identifies information about practices or circumstances that can cause an explosion in a hazardous environment, which may lead to personal injury or death, property damage, or economic loss.
IMPORTANT	Identifies information that is critical for successful application and understanding of the product.
ATTENTION	Identifies information about practices or circumstances that can lead to personal injury or death, property damage, or economic loss. Attentions help you identify a hazard, avoid a hazard, and recognize the consequence.
SHOCK HAZARD	Labels may be on or inside the equipment, for example, a drive or motor, to alert people that dangerous voltage may be present.
BURN HAZARD	Labels may be on or inside the equipment, for example, a drive or motor, to alert people that surfaces may reach dangerous temperatures.

Preface

Chapter 1

Commissioning Your Ultra3000 Drive

Chapter 2

Troubleshooting Your Ultra3000 Servo Drive

Appendix A

Interconnect Diagrams

Appendix B

Understanding Motor Feedback Signals and Outputs

Appendix C

Minimizing the Effects of Feedback Signal Loss

Appendix D

Exporting and Importing Drive Setup Files

Preface

About This Publication

This manual provides power-up procedures, system integration, and troubleshooting tables for the Ultra3000 Digital Servo Drives. The purpose of this manual is to assist you in the integration of your Ultra3000 servo drive as a standalone drive by using Ultraware software or with a Logix controller by using RSLogix 5000 software.

System Integration Architecture

Drive Type	Catalog Numbers	Command Interface	Software
SERCOS interface drive	2098-DSD-*xxx*-SE and 2098-DSD-HV*xxx*-SE	Fiber-optic SERCOS interface	RSLogix 5000
Analog drive	2098-DSD-*xxx* and 2098-DSD-HV*xxx*	Analog command interface	Ultraware or RSLogix 5000 (1)
Digital drive with DeviceNet interface	2098-DSD-*xxx*-DN and 2098-DSD-HV*xxx*-DN	DeviceNet communication interface	Ultraware and RSNetWorx
Indexing DeviceNet drives	2098-DSD-*xxx*X-DN and 2098-DSD-HV*xxx*X-DN	Standalone control	
Indexing drive	2098-DSD-*xxx*X and 2098-DSD-HV*xxx*X		Ultraware

(1) Use RSLogix 5000 software when the 1756-M02AE analog module controls the Ulta3000 drive.

Who Should Use This Manual

This manual is intended for engineers or programmers directly involved in the operation, field maintenance, and integration of the Ultra3000 servo drives.

If you do not have a basic understanding of the Ultra3000 drives, contact your local Rockwell Automation sales representative before using this product for information on available training courses.

Conventions Used in This Manual

The following conventions are used throughout this manual:

- Bulleted lists such as this one provide information, not procedural steps
- Numbered lists provide sequential steps or hierarchical information
- Abbreviations for the Ultra3000 drives are used throughout this manual

Ultra3000 Drive	Abbreviation
Ultra3000 drive with SERCOS interface	Ultra3000-SE
Ultra3000 drive with DeviceNet interface	Ultra3000-DN
Ultra3000 drive with Indexing	Ultra3000X
Ultra3000 analog	Ultra3000

Additional Resources

The following documents contain additional information concerning related Rockwell Automation products.

Resource	Description
Ultra3000 Digital Servo Drives Installation Manual, publication 2098-IN003	The instructions needed for the installation and wiring of the Ultra3000 drives.
Ultraware CD Installation Instructions, publication 2098-IN002	Ultraware software installation instructions.
Ultraware User Manual, publication 2098-UM001	Information on configuring your Ultra3000 drive by using Ultraware software.
Ultra3000 DeviceNet Reference Manual, publication 2098-RM001	Information on communicating with Ultra3000 drives by using the DeviceNet network.
DeviceNet Media Design and Installation Guide, publication DNET-UM072	Information on how to design and install a DeviceNet network cable system.
Kinetix Motion Control Selection Guide, publication GMC-SG001	Specifications, motor/servo-drive system combinations, and accessories for Kinetix motion control products.
Motion Analyzer CD, publication PST-SG003	Drive and motor sizing with application analysis software.
Resistive Brake Module Installation Instructions, publication 2090-IN009	Information on installing and wiring Bulletin 2090 resistive brake modules.
MP-Series Integrated Linear Stages User Manual, publication MP-UM001	Information on installing and wiring MP-Series integrated linear stages.
ControlLogix Motion Module Programming Manual, publication 1756-RM086	Detailed information on the use of ControlLogix motion features and application examples.
ControlLogix SERCOS interface Module Installation Instructions, publication 1756-IN572	ControlLogix SERCOS interface module installation instructions.
Synchronous Serial Interface (SSI) Servo Module Installation Instructions, publication 1756-IN595	Information on mounting and wiring the 1756-M02AS servo module.
Analog Encoder (AE) Servo Module Installation Instructions, publication 1756-IN047	Information on mounting and wiring the 1756-M02AE servo module.
ControlLogix Controllers User Manual, publication 1756-UM001	Information on installing, configuring, programming, and operating a ControlLogix system.
CompactLogix SERCOS interface Module Installation Instructions, publication 1768-IN005	CompactLogix SERCOS interface module installation instructions.
CompactLogix Controllers User Manual, publication 1768-UM001	Information on installing, configuring, programming, and operating a CompactLogix system.
Logix5000 Controllers Motion Instructions Reference Manual, publication 1756-RM007	Instructions needed to program a motion application.
Motion Modules in Logix5000 Control Systems User Manual, publication LOGIX-UM002	Information on configuring and troubleshooting your ControlLogix and CompactLogix SERCOS interface modules.
Fiber Optic Cable Installation and Handling Instructions, publication 2090-IN010	Information on proper handling, installing, testing, and troubleshooting fiber-optic cables.
System Design for Control of Electrical Noise Reference Manual, publication GMC-RM001	Information, examples, and techniques designed to minimize system failures caused by electrical noise.
EMC Noise Management DVD, publication GMC-SP004	
Rockwell Automation Configuration and Selection Tools, website http://ab.com/e-tools	Online product selection and system configuration tools, including AutoCAD (DXF) drawings.
Rockwell Automation Product Certification link, website http://ab.com	Declarations of conformity currently available from Rockwell Automation.
National Electrical Code, published by the National Fire Protection Association of Boston, MA	Article on wire sizes and types for grounding electrical equipment.
Rockwell Automation Industrial Automation Glossary, publication AG-7.1	Glossary of industrial automation terms and abbreviations.

You can view or download publications at http://literature.rockwellautomation.com. To order paper copies of technical documentation, contact your local Rockwell Automation distributor or sales representative.

Commissioning Your Ultra3000 Drive

This chapter provides you with information to apply power and configure your Ultra3000 servo drive.

Introduction

These procedures assume you have completed mounting, wiring, and connecting your Ultra3000 drive as described in the Ultra3000 Digital Servo Drives Installation Manual, publication 2098-IN003.

For installation information regarding equipment and accessories not included here, refer to Additional Resources on page 8 for the information available for those products.

General Startup Precautions

These precautions apply to all of the procedures in this chapter. Be sure to read and thoroughly understand them before proceeding.

ATTENTION

⚠ This product contains stored energy devices. To avoid hazard of electrical shock, verify that all voltages on the system bus network have been discharged before attempting to service, repair, or remove this unit. Only qualified personnel familiar with solid state control equipment and safety procedures in publication NFPA 70E or applicable local codes should attempt this procedure.

ATTENTION

⚠ This drive contains ESD (electrostatic discharge) sensitive parts and assemblies. You are required to follow static control precautions when you install, test, service, or repair this assembly. If you do not follow ESD control procedures, components can be damaged.

If you are not familiar with static control procedures, refer to Guarding Against Electrostatic Damage Service Bulletin, publication 8000-4.5.2, or any other applicable ESD awareness handbook.

Understanding the Serial Connection

If your personal computer has a serial port, use a 2090-UXPC-D09*xx* serial cable or similar null modem cable with wiring as described in the Ultra3000 Digital Servo Drives Installation Manual, publication 2098-IN003.

If your personal computer has USB ports, use a USB to serial adapter (catalog number 9300-USBS) to convert your RS-232 port to USB. The 2090-UXPC-D09*xx* serial cable is still needed between the 9300-USBS converter and the Ultra3000 drive.

USB Communication Rate Compatibility

USB Converter	Communication Rate
USB 1.0	1200, 2400, 4800, 9600, or 19,200, bps
USB 2.0	1200, 2400, 4800, 9600, 19,200, or 38,400 bps

IMPORTANT

The USB converter must be setup as COM1, COM2, COM3, or COM4 and must match the serial port configuration in Ultraware software for the converter and Ultra3000 drive to communicate.

If RSLinx software is running on your personal computer, shutdown the program from the system tray to avoid conflicts between applications.

Configuring Your Ultra3000 Drive and Ultra3000 Drive with Indexing

The procedures in this section are listed in this table and apply to Ultra3000 drives and Ultra3000 drives with indexing.

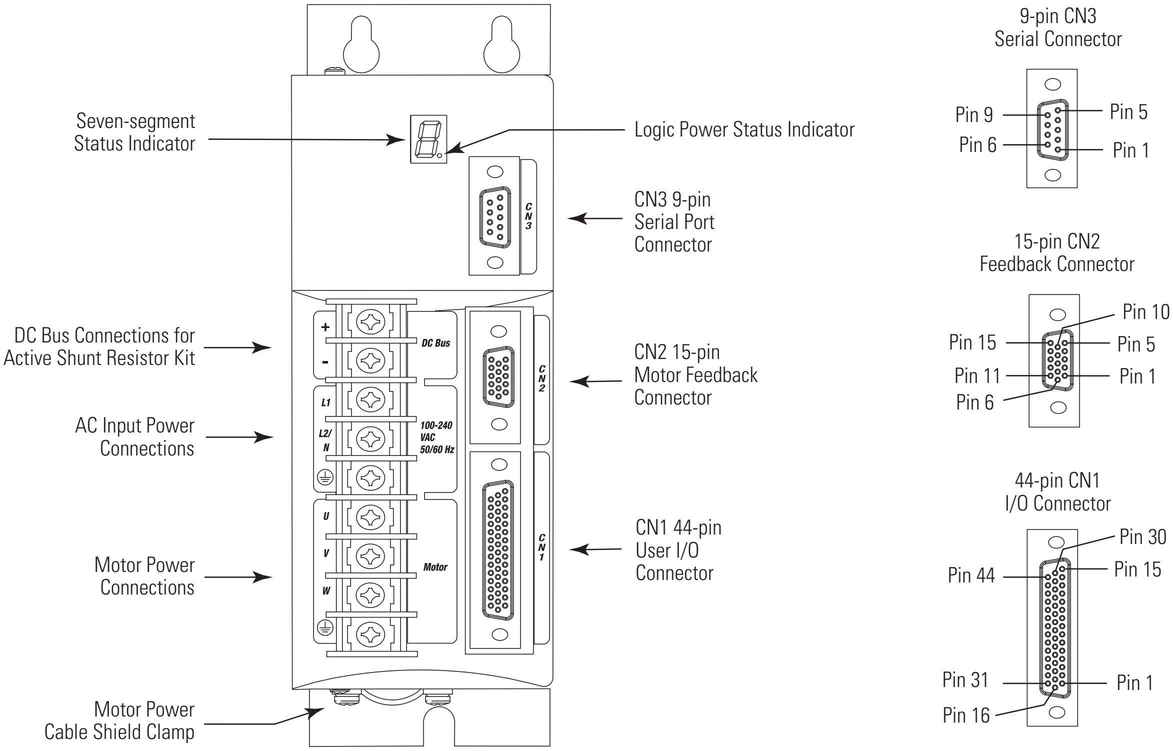
Ultra3000 Drive Configuration Procedures

Procedure	Page
Apply Power To Your Ultra3000 Drive	15
Detect Your Ultra3000 Drive	16
Understanding the Workspace and Drive Branches	17
Select a Motor	21
Tune Your Motor	22
Configure Displayed Units	23
Test Your Motor (non-indexing move)	25
Test Your Motor (indexing move)	26
Indexing and Non-indexing Move Examples	29

Front Panel Connections

Use this figure to locate the front panel connections on the Ultra3000 230V drives (500 W, 1 kW, and 2 kW).

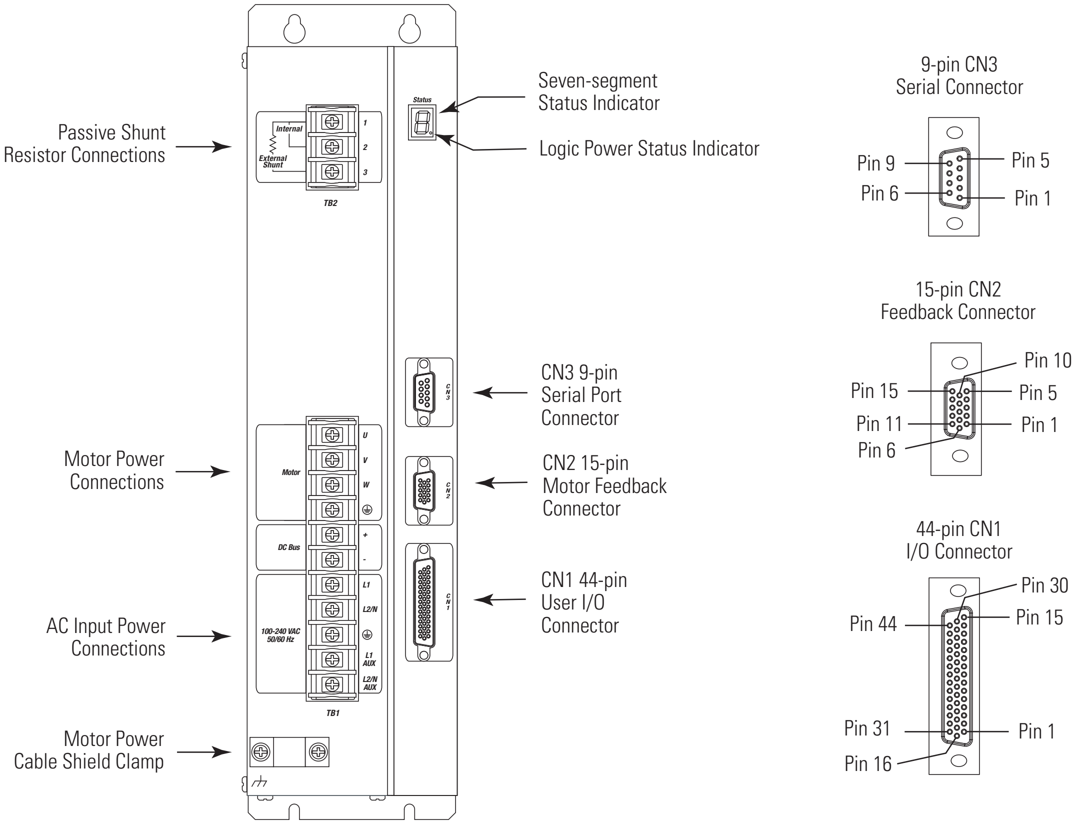
Front Panel Connections for 2098-DSD-005, 2098-DSD-005X, 2098-DSD-010, 2098-DSD-010X, 2098-DSD-020, and 2098-DSD-020X Drives



For CN1, CN2, and CN3 connector pin-out information, refer to the Ultra3000 Digital Servo Drives Installation Manual, publication 2098-IN003.

Use this figure to locate the front panel connections on the Ultra3000 230V drives (3 kW).

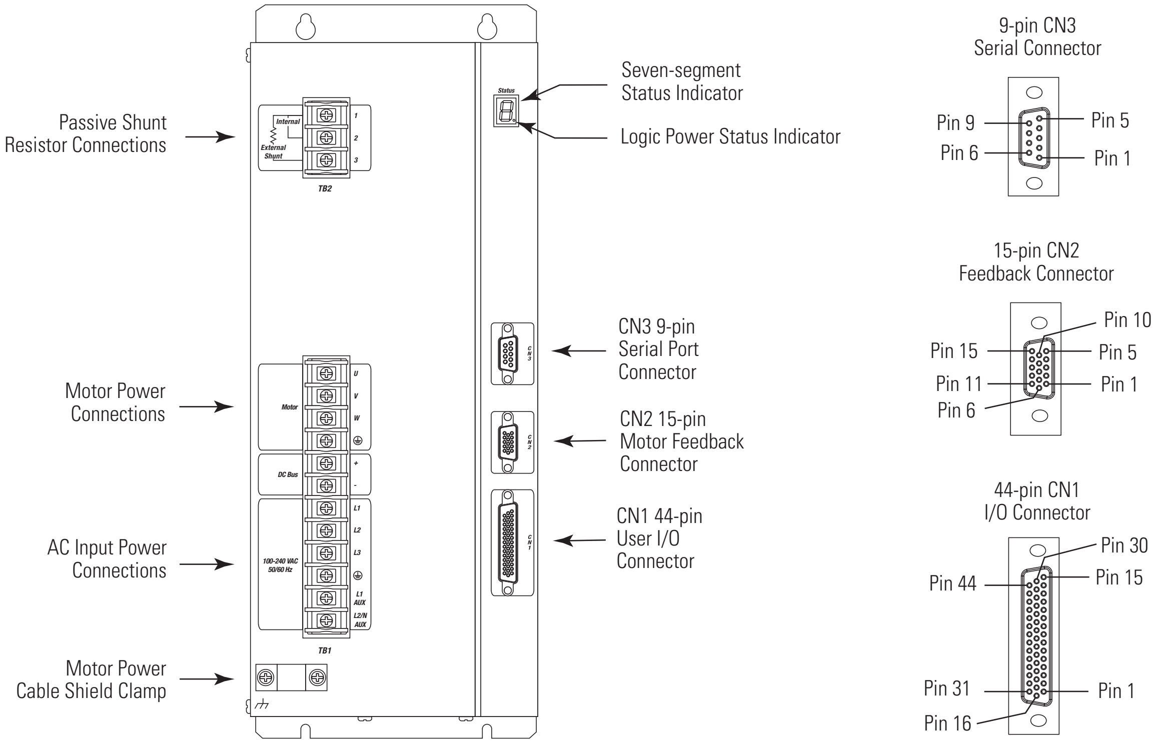
Front Panel Connections for 2098-DSD-030 and 2098-DSD-030X Drives



For CN1, CN2, and CN3 connector pin-out information, refer to the Ultra3000 Digital Servo Drives Installation Manual, publication 2098-IN003.

Use this figure to locate the front panel connections on the Ultra3000 230V drives (7.5 and 15 kW).

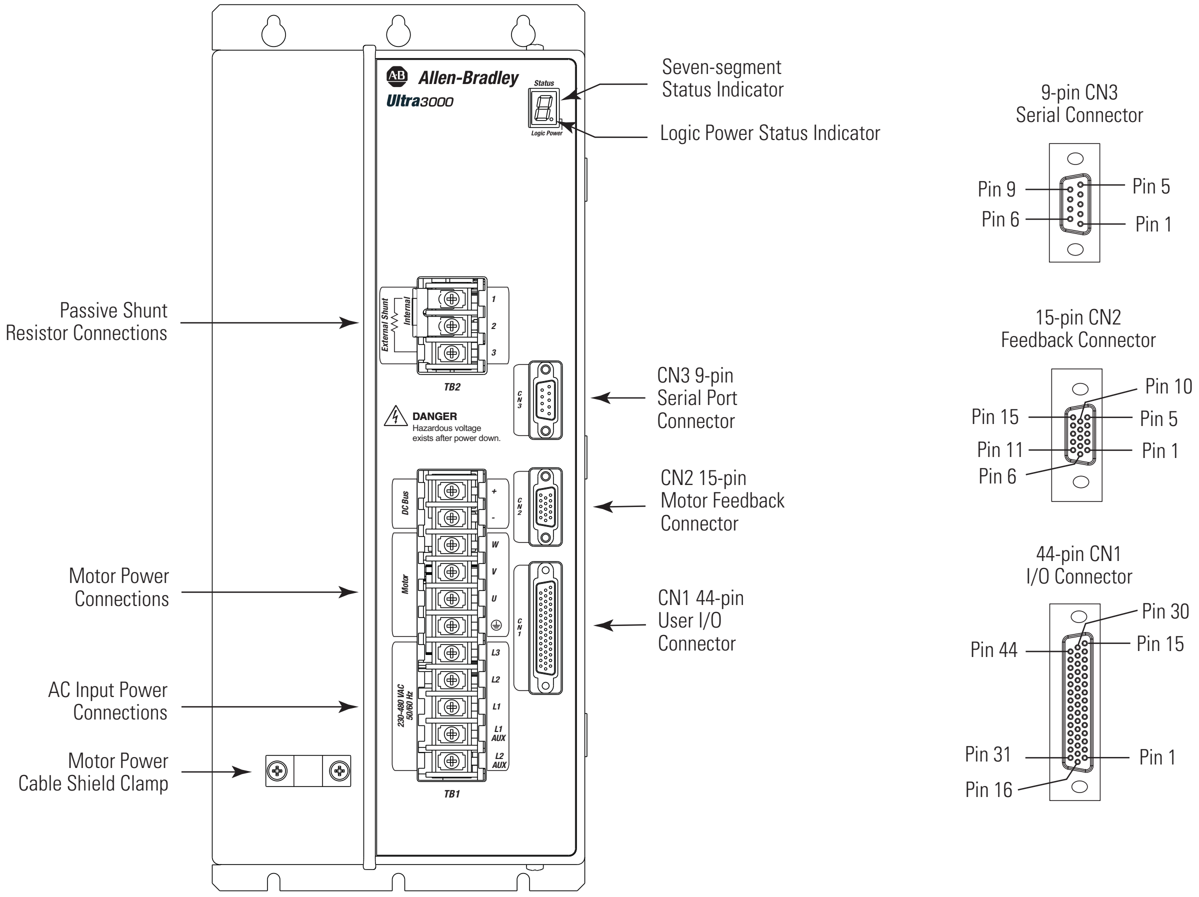
Front Panel Connections for 2098-DSD-075, 2098-DSD-075X, 2098-DSD-150, and 2098-DSD-150X Drives



For CN1, CN2, and CN3 connector pin-out information, refer to the Ultra3000 Digital Servo Drives Installation Manual, publication 2098-IN003.

Use this figure to locate the front panel connections on the Ultra3000 460V drives (3 W, 5 kW, 10 kW, 15 kW, and 22 kW).

Front Panel Connections for 2098-DSD-HV*xxx* and 2098-DSD-HV*xxx*X Drives



For CN1, CN2, and CN3 connector pin-out information, refer to the Ultra3000 Digital Servo Drives Installation Manual, publication 2098-IN003.

Apply Power To Your Ultra3000 Drive

This procedure assumes you have wired your Ultra3000 system, verified the wiring, and are ready to begin using your Ultraware software.

ATTENTION ⚠

High voltage exists in ac line filters. The filter must be grounded properly before applying power. Filter capacitors retain high voltages after power removal. Before handling the equipment, voltages should be measured to determine safe levels. Failure to observe this precaution could result in personal injury.

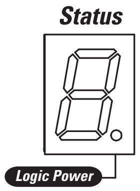
Follow these steps to apply power to your Ultra3000 drive.

1. Disconnect any load to the motor, making sure the motor is free of all linkages when initially applying power to the system.

   ATTENTION ⚠

   To avoid damage to the drive due to improper sequencing of input power and the Drive Enable (Input 1) signal, do not apply the Drive Enable signal without first applying input power.

2. Apply input power to the Ultra3000 drive and observe the front panel Logic Power status indicator.



If the Logic Power status indicator is	Then
ON	Go to step 3.
Not ON	1. Check your input power connections. 2. Repeat step 2.

3. Observe the front panel seven-segment status indicator on your Ultra3000 drive.

Seven-segment Status Indicator	Status	Do This
Actively cycling segments in a full circle	The drive is ready.	Go to Detect Your Ultra3000 Drive on page 16.
Flashing E followed by two numbers	The drive is faulted.	Go to Error Codes on page 98.

Detect Your Ultra3000 Drive

This procedure assumes you have successfully applied power to your drive. These steps are designed to make sure that your Ultra3000 drive is communicating with your Ultraware software.

Follow these steps to detect your Ultra3000 drive.

1. Start your Ultraware software.

   Refer to the Ultraware User Manual, publication 2098-UM001, for more information on starting the Ultraware software.

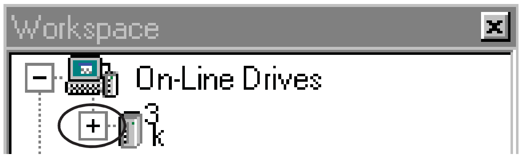
2. Create a new file.

   The software will scan for online drives.

3. Click Stop Scanning when your drive is detected or wait for the scanning to time out.

4. Look for the Ultra3000 icon under the On-Line Drives tree.

   The Ultra3000 icon indicates that your drive is detected.

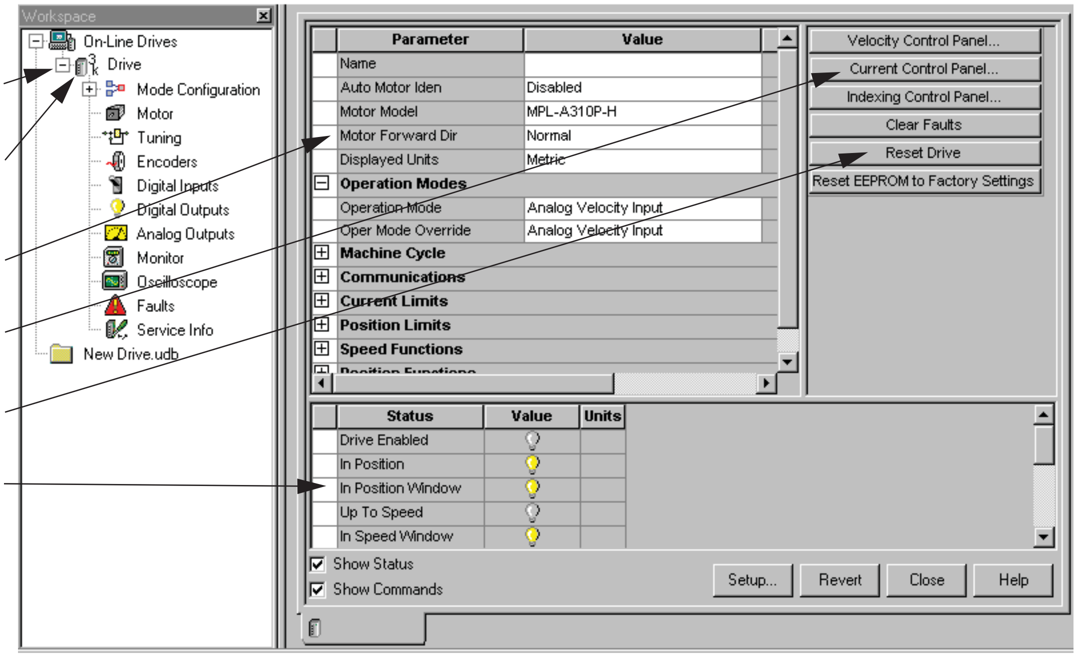


5. Click the [+] next to the Ultra3k icon to expand the branch menu.

If your Ultra3000 drive	Then
Is detected and listed under the On-Line Drives tree	1. The software and hardware are communicating and the system is ready. 2. Go to Select a Motor on page 21.
Is not detected	1. Check your serial cable connections. 2. Use Recover Communications (in Ultraware) to establish a connection. 3. Go to main step 1 of this section.

Understanding the Workspace and Drive Branches

This section provides a description of the Ultraware workspace and various drive branches.



Parameter	Value
Name	
Auto Motor Iden	Disabled
Motor Model	MPL-A310P-H
Motor Forward Dir	Normal
Displayed Units	Metric
Operation Modes	
Operation Mode	Analog Velocity Input
Oper Mode Override	Analog Velocity Input
Machine Cycle	
Communications	
Current Limits	
Position Limits	
Speed Functions	
Position Functions	

Status	Value	Units
Drive Enabled		
In Position		
In Position Window		
Up To Speed		
In Speed Window		

☑ Show Status
☑ Show Commands

Click the [+] next to 3k Drive to expand the parameter group.

Double-click the 3k Drive icon in the Ultraware workspace to display the various drive branches.

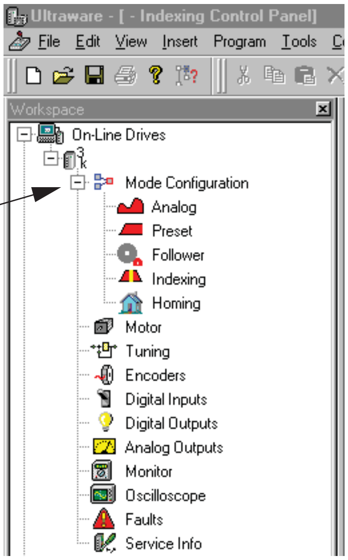
Configure drive parameters for an off-line drive.

Open the Control Panel dialogs to issue motion commands.

Execute commands to clear faults, reset the drive, or reset the EEPROM.

Monitor the status of an online drive.

Mode Configuration Branch



Click the [+] next to Mode Configuration to select the drive's command source.

Motor Branch

Use the Motor Branch to:

- select a motor for the associated online or offline Ultra3000 drive. Once you select a motor, the status values associated with the selected motor appear in the Status pane of this dialog.
- monitor the status as related to the selected motor.
- perform diagnostics on the motor.

Diagnostic commands are not available for SERCOS drives.

Tuning Branch

Use the Tuning Branch to:

- configure Velocity and Position Regulator Gains that are used in tuning.
- monitor Velocity, Position, and Current loop status.
- open dialogs where you can execute commands for autotuning, manual position tuning, and manual velocity tuning.

Encoders Branch

Use the Encoders Branch to:

- define the motor and auxiliary encoders.
- configure the motor encoder and optional auxiliary encoder.

Digital Inputs Branch

Use the Digital Inputs Branch to:

- assign functionality to digital inputs.
- monitor the status of digital inputs.

Digital Outputs Branch

Use the Digital Outputs Branch to:

- assign functionality to digital outputs.
- set both active and inactive brake delays.
- monitor the status of digital outputs and the digital relay.
- open other dialogs where you can override the state of digital outputs and the relay.

Analog Outputs Branch

Use the Analog Outputs Branch to:

- assign drive signals to analog outputs.
- monitor the status of analog outputs.
- open a dialog where you can monitor and override the analog output value.

Monitor Branch

Use the Monitor Branch to:

- view a collection of statuses.
- open the Monitor Setup dialog where you can select the collection of statuses to display in this dialog.
- load a monitor previously saved.
- save a monitor for later use.

Oscilloscope Branch

Use the Oscilloscope Branch to trace one of four drive signals by:

- configuring the oscilloscope by selecting the drive signal to trace.
- executing commands that run the oscilloscope's tracing function continuously or in response to the configured trigger.
- monitoring the oscilloscope as it traces the selected drive signal.

Faults Branch

Use the Faults Branch to:

- set fault limits.
- monitor fault status.
- execute the Clear Faults command.
- open a dialog where you can review the drive's fault history.
- enable or disable faults.

TIP For more information on setting fault limits, refer to Appendix C, Minimizing the Effects of Feedback Signal Loss on page 141.

Service Information Branch

Use the Service Information Branch to:

- modify the size of an off-line drive file before transferring the configuration to an online drive.
- display and monitor service information about the drive.
- display the firmware version of the drive.

Select a Motor

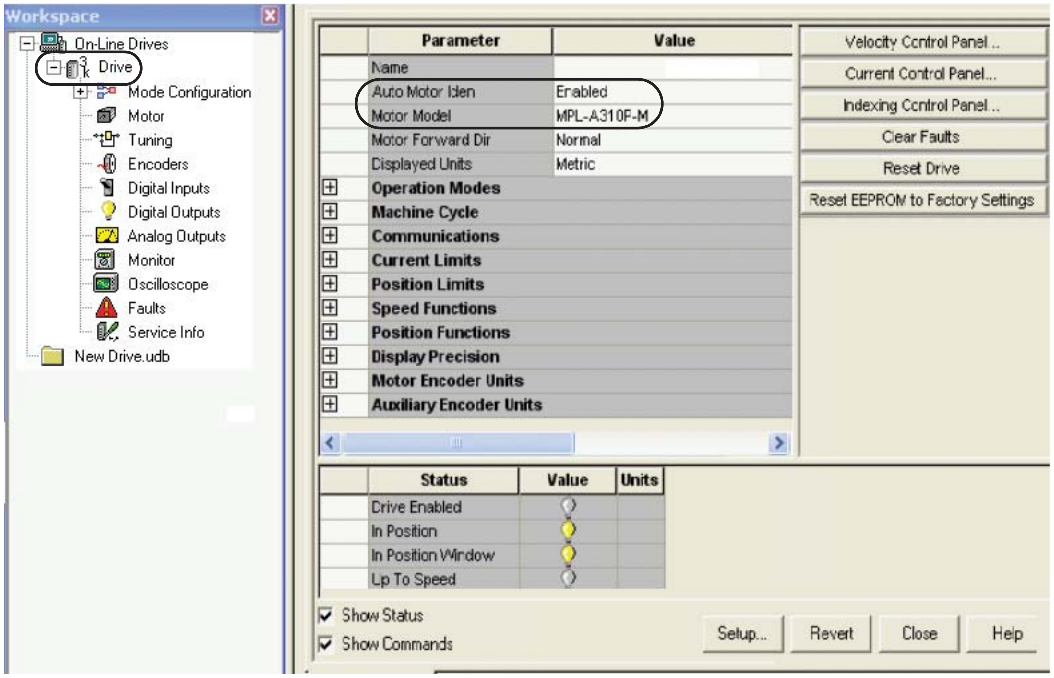
This procedure assumes you have power applied to your drive and the drive is detected by the Ultraware software.

Refer to the Ultraware User Manual, publication 2098-UM001, for more information on selecting a motor.

Follow these steps to select a motor.

1. Double-click the Ultra3000 icon (Ultra3k) under the On-Line Drives tree.

   The Ultra3000 Drive properties dialog opens.



Parameter	Value
Name	
Auto Motor Iden	Enabled
Motor Model	MPL-A310F-M
Motor Forward Dir	Normal
Displayed Units	Metric

Status	Value	Units
Drive Enabled	○	
In Position	●	
In Position Window	●	
Up To Speed	○	

☑ Show Status
☑ Show Commands

   Actual values depend on your application. Auto Motor Iden default value is Enabled and remains Enabled if motor with intelligent encoder is detected or selected. Value changes to Disabled if motor without intelligent encoder is selected.

2. Check the Motor Model parameter value.

If motor is	Value (motor cat. no.)	Go To
An Allen-Bradley motor with intelligent encoder	Is recognized by the Ultraware software	Assign Digital Inputs on page 22.
	Is not recognized by the Ultraware software	Go to Error Codes in Chapter 2 and refer to troubleshooting for E30.
Not an Allen-Bradley motor with intelligent encoder		Step 3.

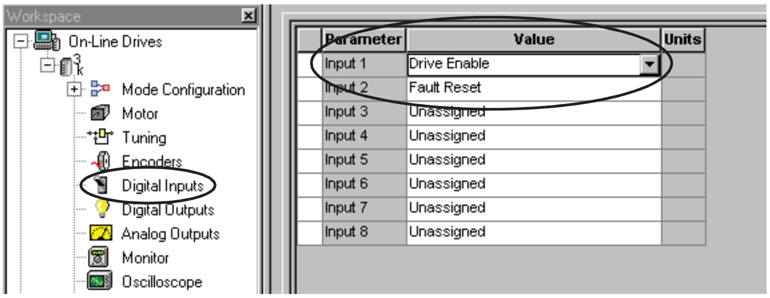
3. From the Motor Model pull-down menu, choose your motor.

Assign Digital Inputs

Follow these steps to assign Digital Inputs 1 and 2.

1. Double-click the Digital Inputs branch.

   The Digital Inputs properties dialog opens.



Parameter	Value	Units
Input 1	Drive Enable	
Input 2	Fault Reset	
Input 3	Unassigned	
Input 4	Unassigned	
Input 5	Unassigned	
Input 6	Unassigned	
Input 7	Unassigned	
Input 8	Unassigned	

2. Verify that Input 1 value is set to Drive Enable (this is default).

ATTENTION

To avoid fault action or damage to the drive due to improper sequencing of input power and the Drive Enable signal, you must assign one of the eight inputs as Drive Enable (Input 1 is the default setting).

3. Configure remaining digital inputs as required by your application.

4. Close the Digital Inputs properties dialog.

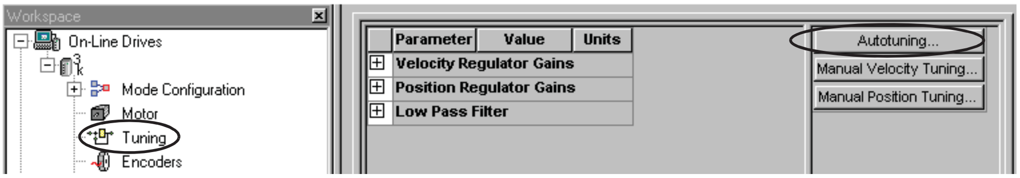
Tune Your Motor

This procedure assumes your drive is detected and you have selected a motor. In this procedure you will autotune your motor.

Follow these steps to autotune your motor.

1. Double-click the Tuning branch.

   The Tuning properties dialog opens.



Parameter	Value	Units
Velocity Regulator Gains		
Position Regulator Gains		
Low Pass Filter		

2. Click Autotuning.

   The Autotuning dialog opens.

3. Apply 12…24V to input 1.

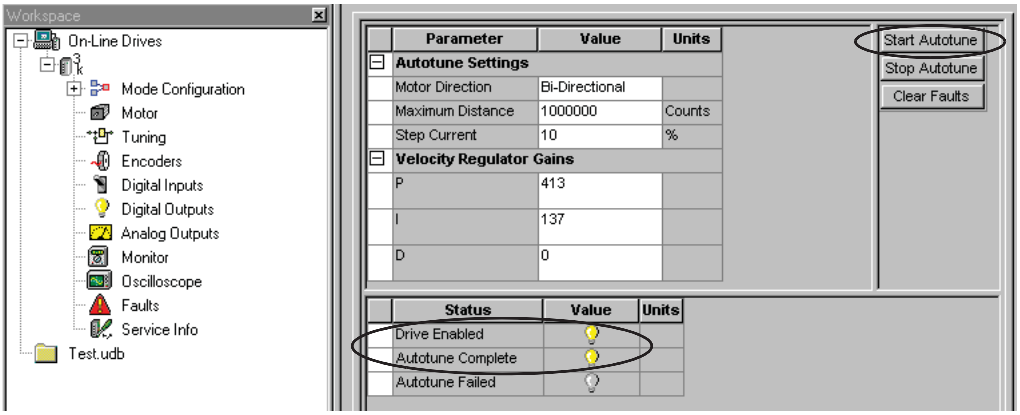
   Input 1 was configured as Drive Enable in a previous step (Drive Enabled light turns yellow).

ATTENTION ⚠ To avoid damage to the drive due to improper sequencing of input power and the Drive Enable signal, do not apply Drive Enable signal without first applying input power.

4. Make the appropriate autotune settings for your application.

5. Click Start Autotune.

   The motor responds and the tuning process is complete (Autotune Complete light turns yellow). Actual values depend on your application.



Parameter	Value	Units
Autotune Settings		
Motor Direction	Bi-Directional	
Maximum Distance	1000000	Counts
Step Current	10	%
Velocity Regulator Gains		
P	413	
I	137	
D	0	

Status	Value	Units
Drive Enabled	●	
Autotune Complete	●	
Autotune Failed	○	

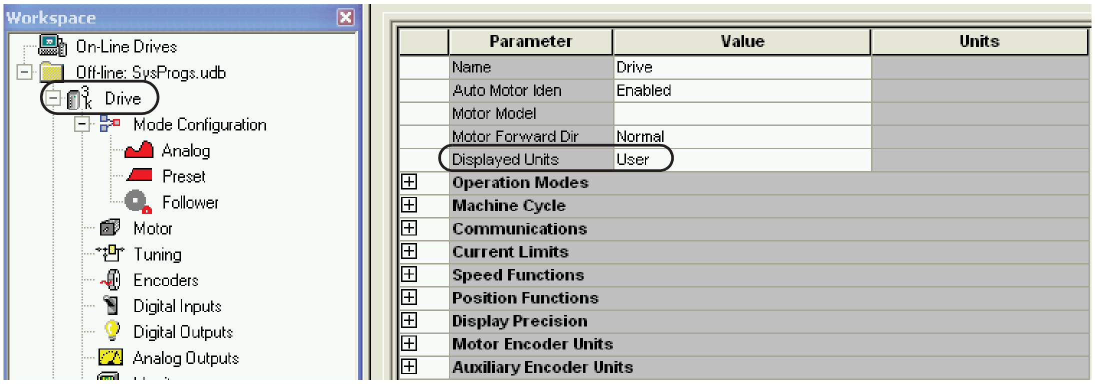
6. Close the Tuning properties dialog.

Configure Displayed Units

The default value setting for Displayed Units is metric. English units are also an option. For values of your own choosing, select User. User units is similar to setting up an application conversion constant. This is useful when the application requires the use of a transmission or other equipment. For example, if motor encoder activity is being measured in counts and the number of revolutions (rpm) is more meaningful, you can change counts to rpm. You can make similar settings for auxiliary encoder units.

1. Double-click the Ultra3000 icon (Ultra3k) under the On-Line Drives tree.

The Ultra3000 Drive properties dialog opens.

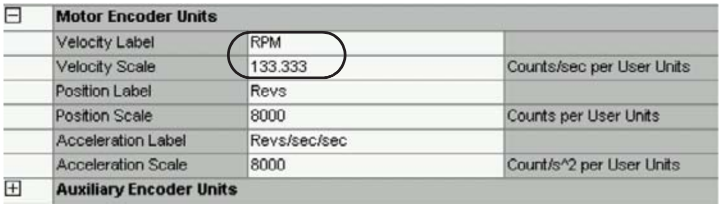


	Parameter	Value	Units
	Name	Drive	
	Auto Motor Iden	Enabled	
	Motor Model		
	Motor Forward Dir	Normal	
	Displayed Units	User	
⊞	**Operation Modes**		
⊞	**Machine Cycle**		
⊞	**Communications**		
⊞	**Current Limits**		
⊞	**Speed Functions**		
⊞	**Position Functions**		
⊞	**Display Precision**		
⊞	**Motor Encoder Units**		
⊞	**Auxiliary Encoder Units**		

2. Click the Value field next to Display Units and choose User.

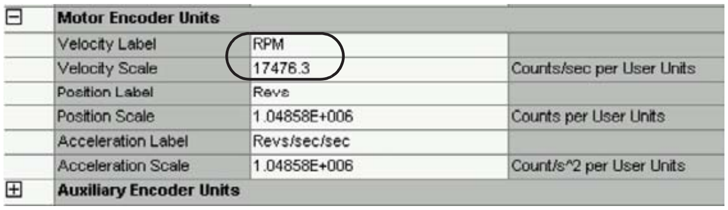
3. Click the [+] next to Motor Encoder Units.

Use these parameter settings for an incremental encoder. To display velocity in rpm divide 8000 counts/rev by 60 seconds/minute or 133.333. For position and acceleration use 8000.



⊟	**Motor Encoder Units**		
	Velocity Label	RPM	
	Velocity Scale	133.333	Counts/sec per User Units
	Position Label	Revs	
	Position Scale	8000	Counts per User Units
	Acceleration Label	Revs/sec/sec	
	Acceleration Scale	8000	Count/s^2 per User Units
⊞	**Auxiliary Encoder Units**		

Use these parameter settings for a Stegmann encoder. To display velocity in rpm divide 1,048,576 counts/rev by 60 seconds/minute or 17476.267. For position and acceleration use 1048576. Ultraware software may truncate or convert the number into scientific notation.



⊟	**Motor Encoder Units**		
	Velocity Label	RPM	
	Velocity Scale	17476.3	Counts/sec per User Units
	Position Label	Revs	
	Position Scale	1.04858E+006	Counts per User Units
	Acceleration Label	Revs/sec/sec	
	Acceleration Scale	1.04858E+006	Count/s^2 per User Units
⊞	**Auxiliary Encoder Units**		

The Indexing parameters now list the position as revs and acceleration/deceleration as revs/sec/sec as defined above. These examples are for rotary motors directly coupled to the machine.

Test Your Motor (non-indexing move)

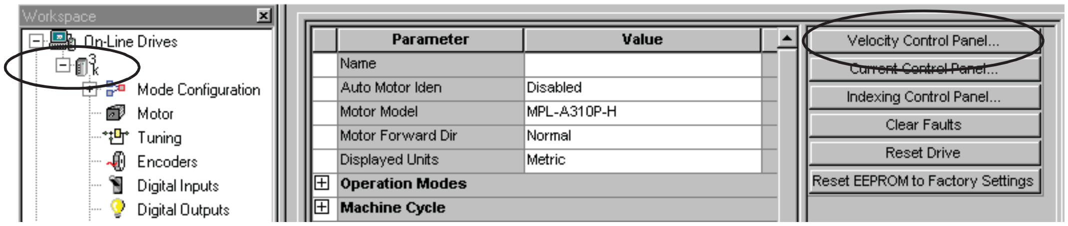
This procedure assumes you have applied power to your drive, the Ultraware software is running, the drive is detected, and you have selected a motor. In this procedure you will enable the drive and set the motor velocity to test the motor.

Refer to the Ultraware User Manual, publication 2098-UM001, for more information on using the velocity control panel.

Follow these steps to jog the motor at a constant speed.

1. Double-click the U3k icon.

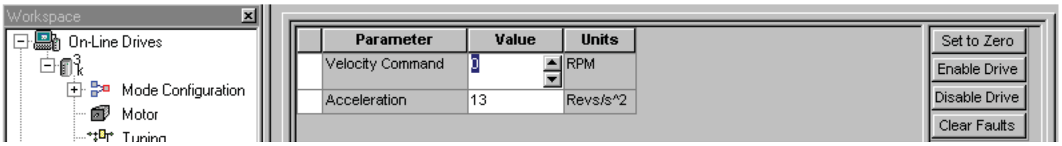
   The drive properties dialog opens.



Parameter	Value
Name	
Auto Motor Iden	Disabled
Motor Model	MPL-A310P-H
Motor Forward Dir	Normal
Displayed Units	Metric
Operation Modes	
Machine Cycle	

2. Click Velocity Control Panel.

   The velocity control panel dialog opens.

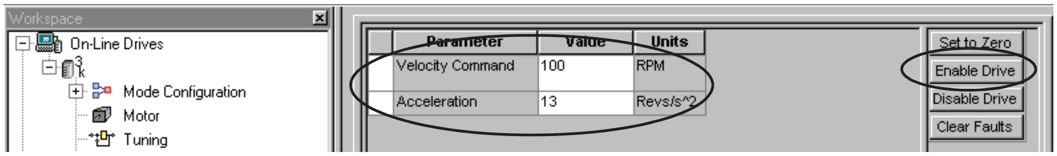


Parameter	Value	Units
Velocity Command	0	RPM
Acceleration	13	Revs/s^2

3. Apply 12…24V dc to input 1.

   Input 1 was configured as Drive Enable in a previous step.

4. Click Enable Drive.



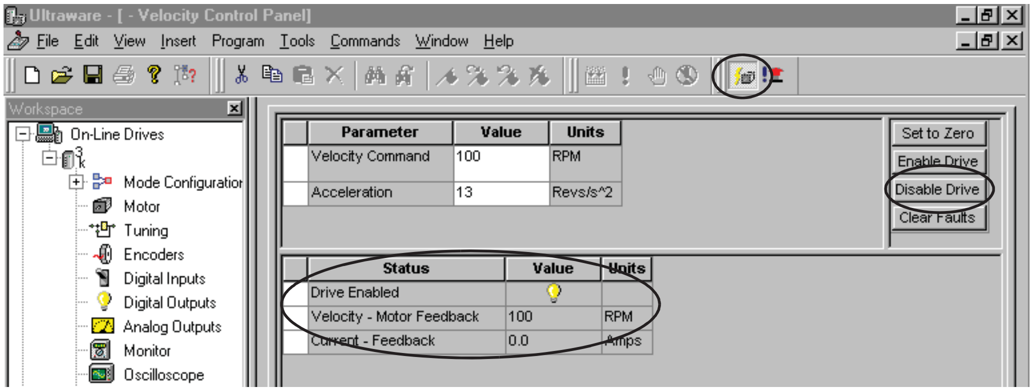
Parameter	Value	Units
Velocity Command	100	RPM
Acceleration	13	Revs/s^2

5. In the Velocity Command box, enter an appropriate low speed.

6. Press Enter.

   The motor should be turning at the velocity you entered in step 5.

7. Observe the Status table.



Parameter	Value	Units
Velocity Command	100	RPM
Acceleration	13	Revs/s^2

Status	Value	Units
Drive Enabled	●	
Velocity - Motor Feedback	100	RPM
Current - Feedback	0.0	Amps

- Drive Enable status = lamp is on (yellow)
- Velocity - Motor Feedback status = the value you entered in step 5

8. Click Disable Drive.

   The motor stops.

9. Close the velocity control panel.

   The drive is software disabled and the enable icon in the toolbar is no longer illuminated.

Test Your Motor (indexing move)

This procedure assumes you have applied power to your drive, the Ultraware software is running, the drive is detected, and you have selected a motor. In this procedure you will enable the drive and make an incremental move to test the motor.

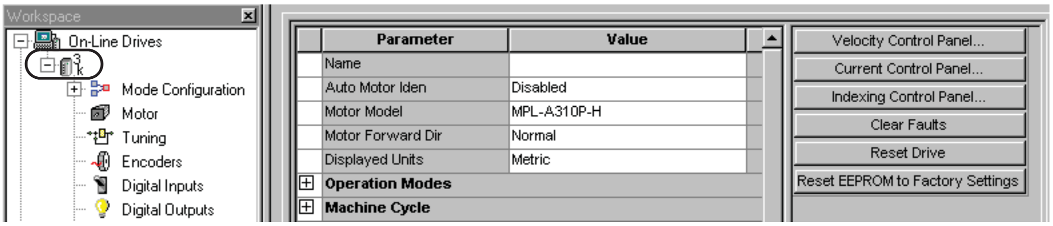
IMPORTANT If you do not have an Ultra3000X indexing drive, you cannot access the indexing functions.

Refer to the Ultraware User Manual, publication 2098-UM001, for more information on using the indexing control panel.

Follow these steps to test your motor.

1. Double-click the U3k icon.

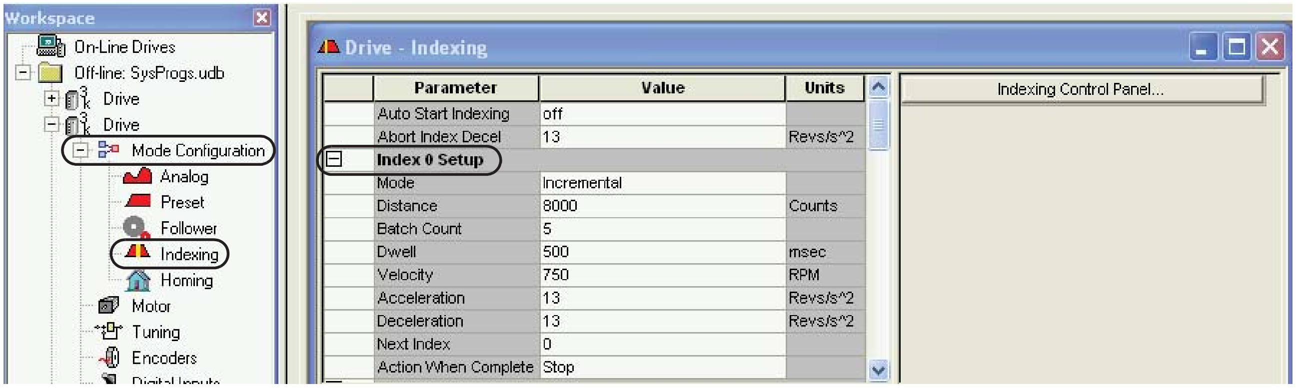
The drive properties dialog opens.



Parameter	Value
Name	
Auto Motor Iden	Disabled
Motor Model	MPL-A310P-H
Motor Forward Dir	Normal
Displayed Units	Metric
⊞ **Operation Modes**	
⊞ **Machine Cycle**	

2. Expand the Mode Configuration branch and double-click Indexing.

   The Indexing Setup dialog opens.



Parameter	Value	Units
Auto Start Indexing	off	
Abort Index Decel	13	Revs/s^2
⊟ **Index 0 Setup**		
Mode	Incremental	
Distance	8000	Counts
Batch Count	5	
Dwell	500	msec
Velocity	750	RPM
Acceleration	13	Revs/s^2
Deceleration	13	Revs/s^2
Next Index	0	
Action When Complete	Stop	

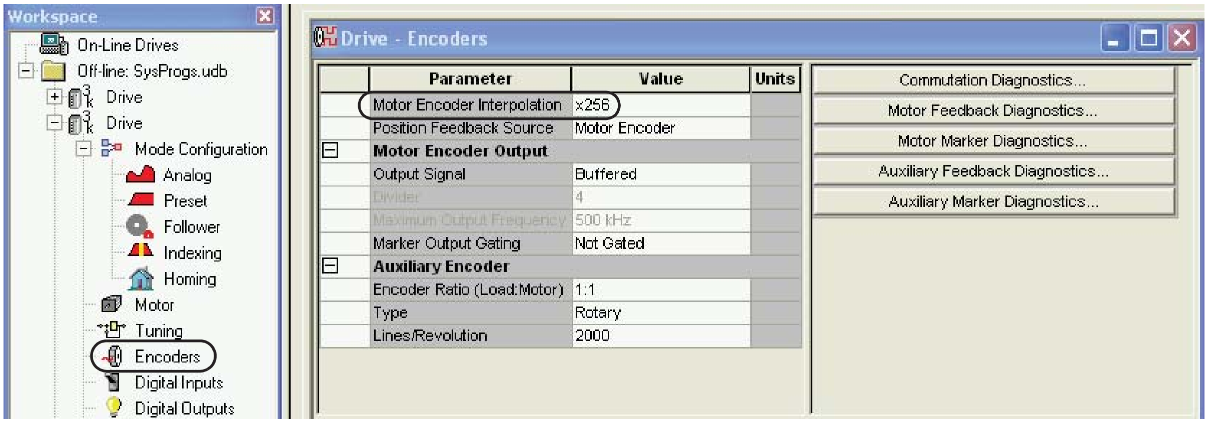
3. Expand Index 0 Setup.

4. Configure your incremental move with the following values for Index 0.

   - Mode = Incremental
   - Distance = 8000 counts
   - Batch count = 5
   - Dwell = 500 ms
   - Velocity = 750 rpm
   - Acceleration = 13 Rev/s^2
   - Deceleration = 13 Rev/s^2
   - Next Index = 0
   - Action When Complete = Stop

   In this example, the Bulletin MPL motor uses an incremental, 2000 ppr (pulse per revolution) feedback device. Therefore, the Ultra3000 drive uses quadrature or 2000 ppr x 4 to equal 8000 counts per revolution.

   If a Bulletin MPL motor with high-resolution feedback is used (catalog number MPL-A310P-M, for example), the feedback device is 1024 ppr (pulses per revolution). However, the interpolation factor, as set in the Encoders tab of the Workspace, determines the counts per revolution. Default interpolation is x256 which totals 1024 x 256 or 262,144 counts per revolution.

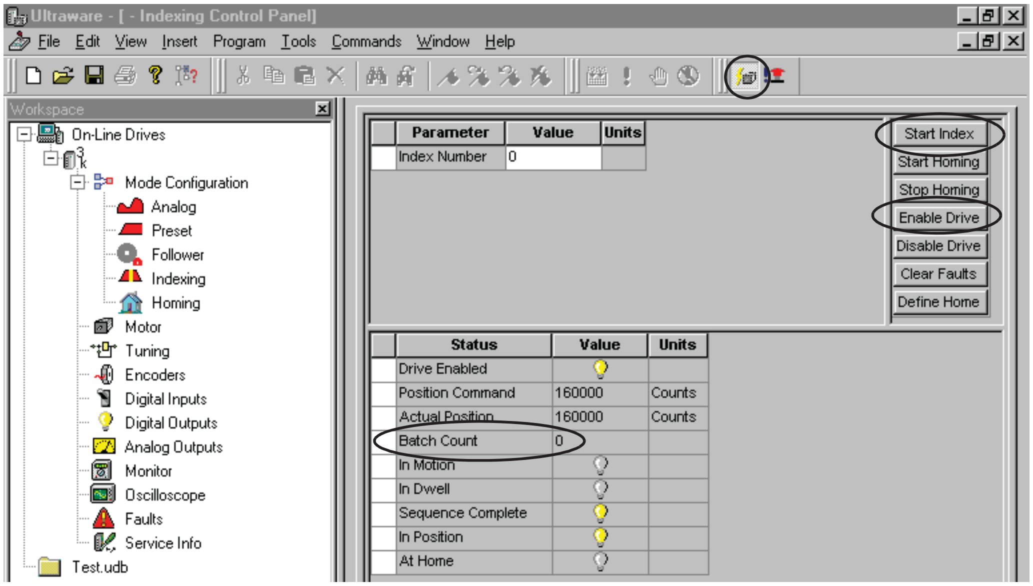


	Parameter	Value	Units
	Motor Encoder Interpolation	x256	
	Position Feedback Source	Motor Encoder	
⊟	**Motor Encoder Output**		
	Output Signal	Buffered	
	Divider	4	
	Maximum Output Frequency	500 kHz	
	Marker Output Gating	Not Gated	
⊟	**Auxiliary Encoder**		
	Encoder Ratio (Load:Motor)	1:1	
	Type	Rotary	
	Lines/Revolution	2000	

These settings may not be appropriate for your application.

5. Click Indexing Control Panel in the drive properties dialog.

   The indexing control panel dialog opens and the software enable icon should be on.



Parameter	Value	Units
Index Number	0	

Status	Value	Units
Drive Enabled	●	
Position Command	160000	Counts
Actual Position	160000	Counts
Batch Count	0	
In Motion	○	
In Dwell	○	
Sequence Complete	●	
In Position	●	
At Home	○	

6. Click Enable Drive.

   The Drive Enabled lamp is on (yellow).

7. Click Start Index.

   Your incremental move begins. Observe the Batch Count value count down from 5…0 while your move is running. Also, observe the Position Command and Actual Position values following the incremental index 0 count setup for each move.

8. Close the indexing control panel dialog.

   The drive is software disabled and the toolbar Enable icon is no longer on.

9. Close the Indexing mode dialog.

Indexing and Non-indexing Move Examples

This section provides examples of indexing and non-indexing moves you can make with your Ultra3000 drive by using Ultraware software.

Ultra3000 Drive Configuration Procedures

Procedure	Page
Analog Velocity Mode (non-indexing)	29
Analog Position Mode (non-indexing)	32
Preset Velocity Control (non-indexing)	34
Preset Position Control (indexing move)	37
Master Follower and Preset Gear Ratios (non-indexing move)	40
Incremental Indexing (indexing move)	44
Absolute Indexing (indexing move)	48

Analog Velocity Mode (non-indexing)

This procedure assumes you have applied power to your drive, the Ultraware software is running, the drive is detected, and you have tested a motor. In this procedure you will run the drive in Analog Velocity mode.

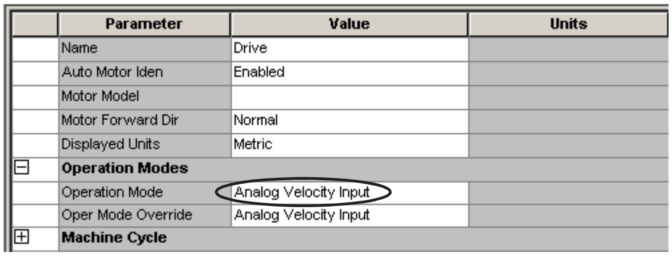
Refer to the Ultraware User Manual, publication 2098-UM001, for more information on Analog Velocity mode.

Follow these steps to run your drive in Analog Velocity mode.

1. Double-click the U3k icon.

   The drive properties dialog opens.

2. Expand the Operation Modes parameter and verify the Operation Mode is Analog Velocity Input.

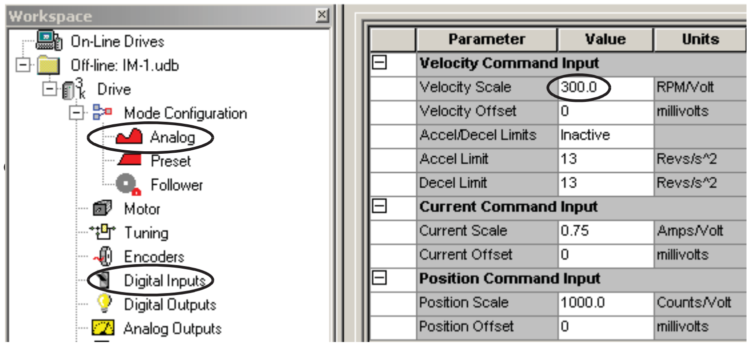


	Parameter	Value	Units
	Name	Drive	
	Auto Motor Iden	Enabled	
	Motor Model		
	Motor Forward Dir	Normal	
	Displayed Units	Metric	
⊟	**Operation Modes**		
	Operation Mode	Analog Velocity Input	
	Oper Mode Override	Analog Velocity Input	
⊞	**Machine Cycle**		

3. Close the Drive Branch dialog.

4. Expand the Mode Configuration branch and double-click Analog.

   The Analog Setup dialog opens.



	Parameter	Value	Units
⊟	**Velocity Command Input**		
	Velocity Scale	300.0	RPM/Volt
	Velocity Offset	0	millivolts
	Accel/Decel Limits	Inactive	
	Accel Limit	13	Revs/s^2
	Decel Limit	13	Revs/s^2
⊟	**Current Command Input**		
	Current Scale	0.75	Amps/Volt
	Current Offset	0	millivolts
⊟	**Position Command Input**		
	Position Scale	1000.0	Counts/Volt
	Position Offset	0	millivolts

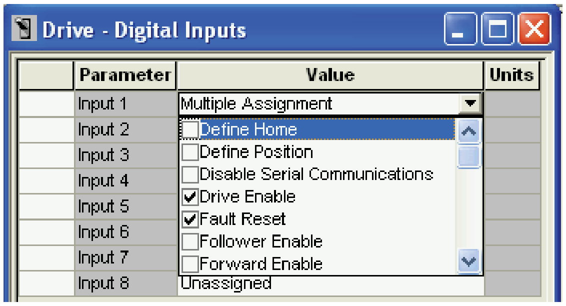
5. In the Velocity Scale box, enter 300.0 and verify Velocity Offset is set to 0.

6. Double-click the Digital Inputs branch.

   a. Verify that Input 1 is configured as Drive Enable input (factory default).

   b. Verify that Input 2 is configured as the Fault Reset input.

   If more digital inputs are required for an application than are available in Ultraware software, you can combine inputs for multiple assignments. In this example both Drive Enable and Fault Reset are combined in Input 1. To reset a fault, toggle 12…24V dc to Input 1 or CN1-31. Then re-energize 12…24V dc to Input 1 or CN1-31 to keep the drive enabled.

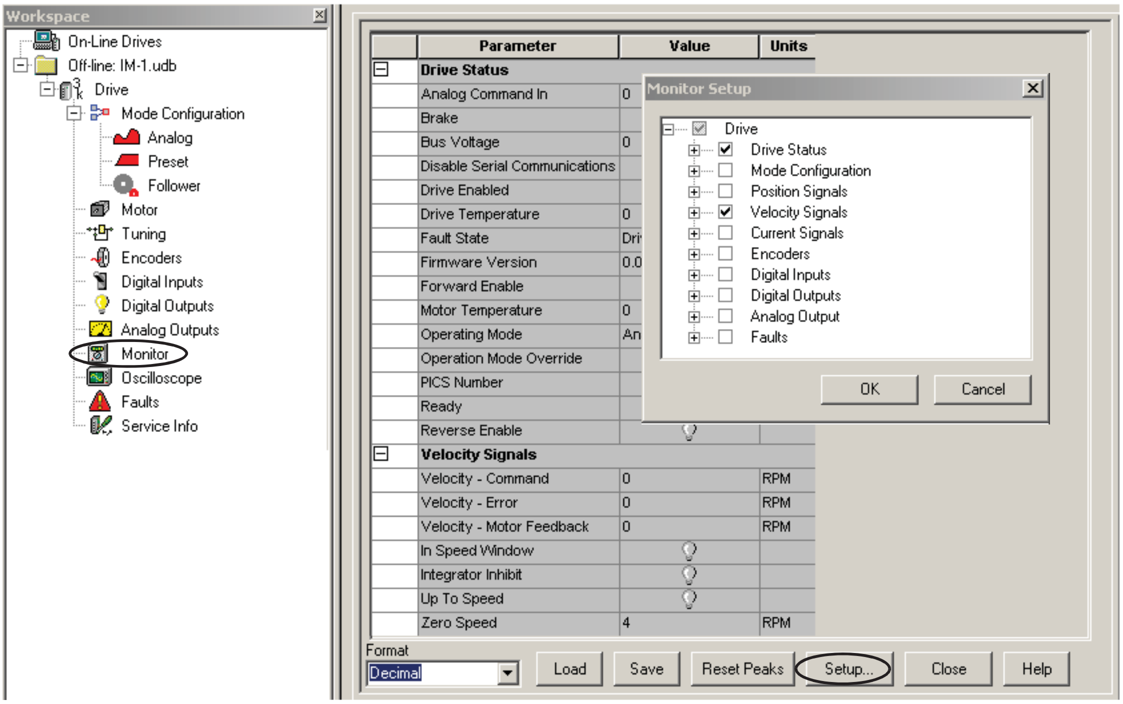


Parameter	Value	Units
Input 1	Multiple Assignment	
Input 2		
Input 3		
Input 4		
Input 5		
Input 6		
Input 7		
Input 8	Unassigned	

☐ Define Home
☐ Define Position
☐ Disable Serial Communications
☑ Drive Enable
☑ Fault Reset
☐ Follower Enable
☐ Forward Enable

7. Double-click the Monitor branch.

The (default) Drive Status parameters display.



	Parameter	Value	Units
⊟	**Drive Status**		
	Analog Command In	0	
	Brake		
	Bus Voltage	0	
	Disable Serial Communications		
	Drive Enabled		
	Drive Temperature	0	
	Fault State	Dri	
	Firmware Version	0.0	
	Forward Enable		
	Motor Temperature	0	
	Operating Mode	An	
	Operation Mode Override		
	PICS Number		
	Ready		
	Reverse Enable	○	
⊟	**Velocity Signals**		
	Velocity - Command	0	RPM
	Velocity - Error	0	RPM
	Velocity - Motor Feedback	0	RPM
	In Speed Window	○	
	Integrator Inhibit	○	
	Up To Speed	○	
	Zero Speed	4	RPM

Monitor Setup

- ☑ Drive
  - ☑ Drive Status
  - ☐ Mode Configuration
  - ☐ Position Signals
  - ☑ Velocity Signals
  - ☐ Current Signals
  - ☐ Encoders
  - ☐ Digital Inputs
  - ☐ Digital Outputs
  - ☐ Analog Output
  - ☐ Faults

8. Click Setup.

The Monitor Setup dialog opens.

9. In the Monitor Setup dialog, check Velocity Signals.

10. Click OK.

The Monitor Status dialog closes.

11. Apply 12…24V dc to input 1.

Input 1 was configured as Drive Enable in a previous step. Make sure the Enable icon in the toolbar is active. This means the drive can enable.

12. Observe the drive responding to a 0…±10V dc analog signal applied to CN1-25 and CN1-26 (1V dc = 300 rpm, per the setup).

- Analog Command voltage
- Velocity Command rpm (300 rpm/analog input voltage)
- Velocity - Motor Feedback

13. Remove the 12…24V dc (Drive Enable) from input 1.

14. Close the Monitor and Digital Inputs branch dialogs and the Analog mode configuration dialog.

Analog Position Mode (non-indexing)

This procedure assumes you have applied power to your drive, the Ultraware software is running, the drive is detected, and you have tested a motor. In this procedure you will run the drive in Analog Position mode.

Refer to the Ultraware User Manual, publication 2098-UM001, for more information on Analog Position mode.

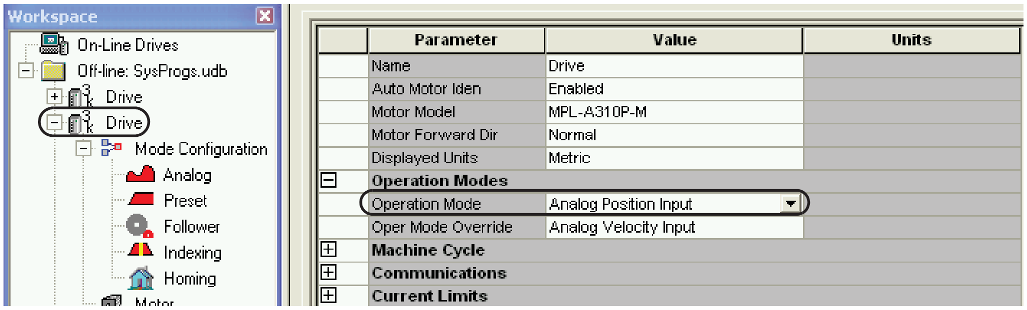
Follow these steps to run your drive in Analog Position mode.

1. Double-click the U3k icon.

The drive properties dialog opens.

2. Expand the Operation Modes parameter.

Verify the Operation Mode is Analog Position Input.



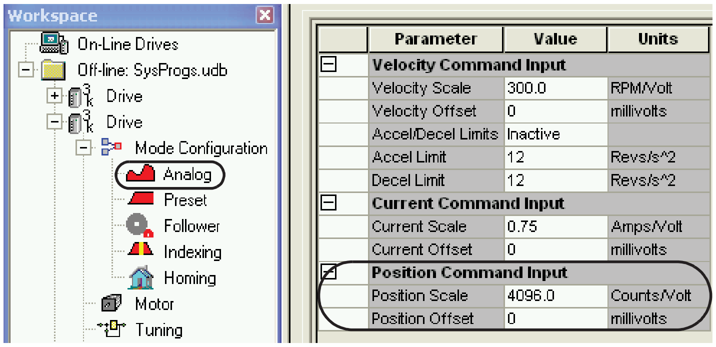
Workspace
On-Line Drives
Off-line: SysProgs.udb
Drive
Drive
Mode Configuration
Analog
Preset
Follower
Indexing
Homing
Motor

	Parameter	Value	Units
	Name	Drive	
	Auto Motor Iden	Enabled	
	Motor Model	MPL-A310P-M	
	Motor Forward Dir	Normal	
	Displayed Units	Metric	
⊟	**Operation Modes**		
	Operation Mode	Analog Position Input	
	Oper Mode Override	Analog Velocity Input	
⊞	**Machine Cycle**		
⊞	**Communications**		
⊞	**Current Limits**		

3. Close the Drive branch dialog.

4. Expand the Mode Configuration branch and double-click Analog.

The Analog Setup dialog opens.



Workspace
On-Line Drives
Off-line: SysProgs.udb
Drive
Drive
Mode Configuration
Analog
Preset
Follower
Indexing
Homing
Motor
Tuning

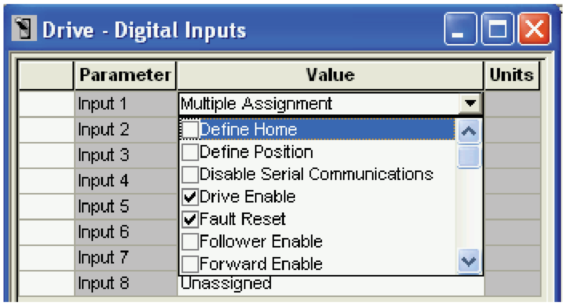
	Parameter	Value	Units
⊟	**Velocity Command Input**		
	Velocity Scale	300.0	RPM/Volt
	Velocity Offset	0	millivolts
	Accel/Decel Limits	Inactive	
	Accel Limit	12	Revs/s^2
	Decel Limit	12	Revs/s^2
⊟	**Current Command Input**		
	Current Scale	0.75	Amps/Volt
	Current Offset	0	millivolts
⊟	**Position Command Input**		
	Position Scale	4096.0	Counts/Volt
	Position Offset	0	millivolts

5. Enter the Position Scale value appropriate for your application.

In this example, the motor is catalog number MPL-A310P-M with 1024 ppr multiplied by an interpolation factor of 8, or 8192 counts per motor revolution. With a Position Scale value of 4096 counts per volt the motor will turn one revolution for every 2V dc.

6. Double-click the Digital Inputs branch.
   a. Verify that Input 1 is configured as Drive Enable input (factory default).
   b. Verify that Input 2 is configured as the Fault Reset input.

   If more digital inputs are required for an application than are available in Ultraware software, you can combine inputs for multiple assignments. In this example both Drive Enable and Fault Reset are combined in Input 1. To reset a fault, toggle 12…24V dc to Input 1 or CN1-31. Then re-energize 12…24V dc to Input 1 or CN1-31 to keep the drive enabled.

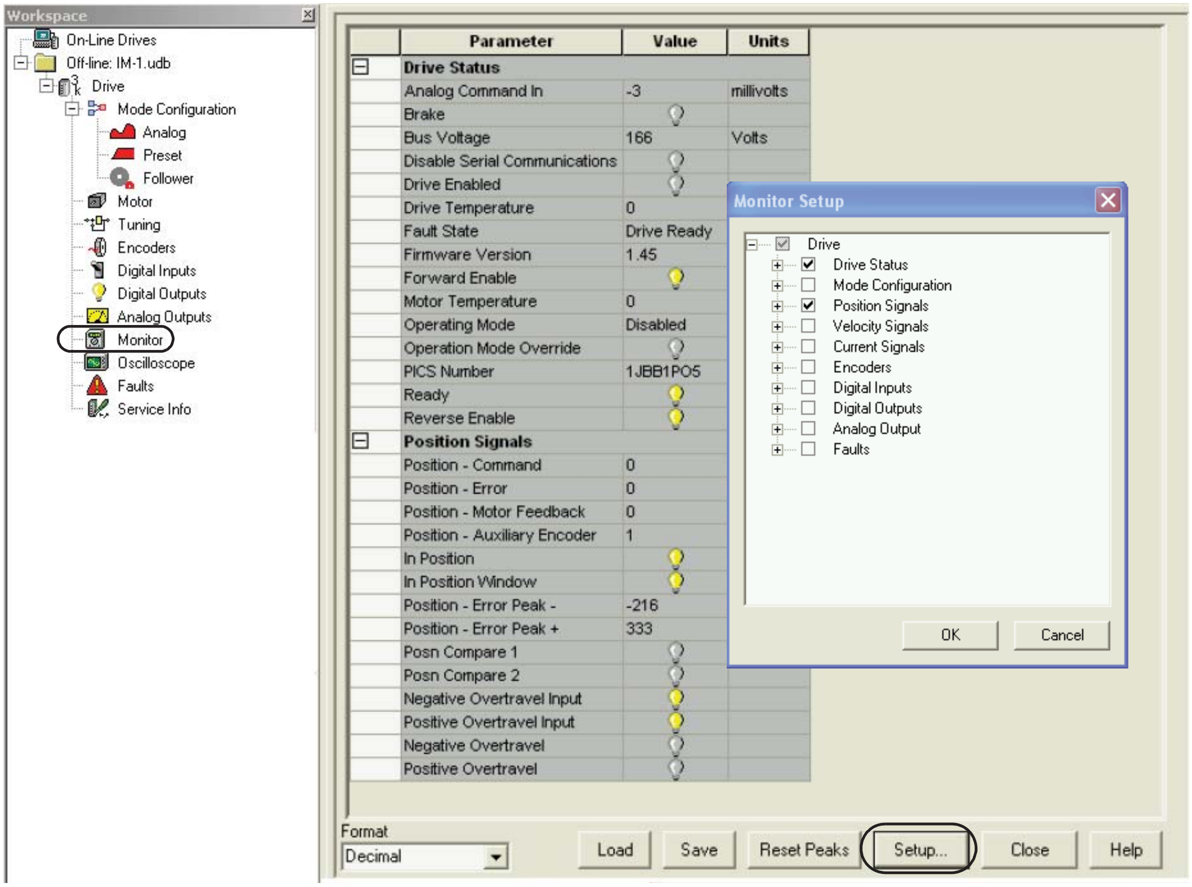


Drive - Digital Inputs

Parameter	Value	Units
Input 1	Multiple Assignment	
Input 2	☐ Define Home	
Input 3	☐ Define Position	
Input 4	☐ Disable Serial Communications	
Input 5	☑ Drive Enable	
Input 6	☑ Fault Reset	
Input 7	☐ Follower Enable ☐ Forward Enable	
Input 8	Unassigned	

7. Double-click the Monitor branch.

   The (default) Drive Status parameters display.



Parameter	Value	Units
Drive Status		
Analog Command In	-3	millivolts
Brake	○	
Bus Voltage	166	Volts
Disable Serial Communications	○	
Drive Enabled	○	
Drive Temperature	0	
Fault State	Drive Ready	
Firmware Version	1.45	
Forward Enable	●	
Motor Temperature	0	
Operating Mode	Disabled	
Operation Mode Override	○	
PICS Number	1JBB1PO5	
Ready	●	
Reverse Enable	●	
Position Signals		
Position - Command	0	
Position - Error	0	
Position - Motor Feedback	0	
Position - Auxiliary Encoder	1	
In Position	●	
In Position Window	●	
Position - Error Peak -	-216	
Position - Error Peak +	333	
Posn Compare 1	○	
Posn Compare 2	○	
Negative Overtravel Input	●	
Positive Overtravel Input	●	
Negative Overtravel	○	
Positive Overtravel	○	

Format: Decimal

Monitor Setup

- ☑ Drive
  - ☑ Drive Status
  - ☐ Mode Configuration
  - ☑ Position Signals
  - ☐ Velocity Signals
  - ☐ Current Signals
  - ☐ Encoders
  - ☐ Digital Inputs
  - ☐ Digital Outputs
  - ☐ Analog Output
  - ☐ Faults

8. Click Setup.

The Monitor Setup dialog opens.

9. In the Monitor Setup dialog, check Position Signals.

10. Click OK.

The Monitor Status dialog closes.

11. Apply 12…24V dc to input 1.

Input 1 was configured as Drive Enable in a previous step. Make sure the Enable icon in the toolbar is active. This means the drive can enable.

12. In the Monitor dialog, observe the Drive Status and Position Signals parameters.

- Drive Enabled lamp is ON (yellow)
- 1V dc = 4096 counts or 1/2 motor revolution
- Supply 0…±10V dc to CN1-25 and CN1-26 and observe Position Command and Position-Motor Feedback

13. Remove the 12…24V dc (Drive Enable) from input 1.

14. Close the Monitor dialog and the Analog mode configuration dialog.

Preset Velocity Control (non-indexing)

This procedure assumes you have applied power to your drive, the Ultraware software is running, the drive is detected, and you have tested a motor. In this procedure you will run the drive by using preset velocity control.

Refer to the Ultraware User Manual, publication 2098-UM001, for more information on preset velocity control.

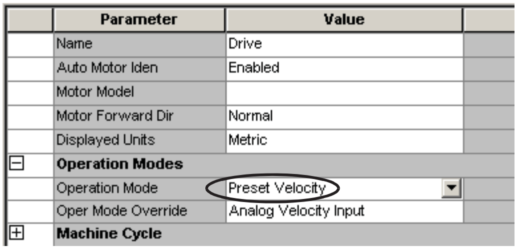
Follow these steps to use preset velocity control.

1. Double-click the U3k icon.

The drive properties dialog opens.

2. Expand the Operation Modes parameter.

3. Click the current setting and use the pull-down menu to change the Operation Mode to Preset Velocity.

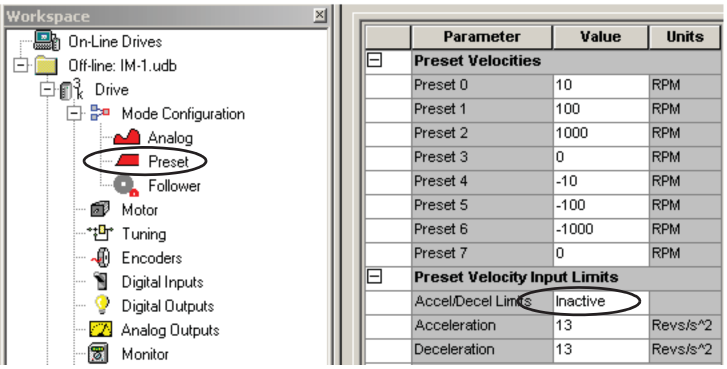


	Parameter	Value
	Name	Drive
	Auto Motor Iden	Enabled
	Motor Model	
	Motor Forward Dir	Normal
	Displayed Units	Metric
⊟	**Operation Modes**	
	Operation Mode	Preset Velocity
	Oper Mode Override	Analog Velocity Input
⊞	**Machine Cycle**	

4. Close the Drive Branch dialog.

5. Expand the Mode Configuration branch and double-click Preset.

   The Preset setup dialog opens.



	Parameter	Value	Units
⊟	**Preset Velocities**		
	Preset 0	10	RPM
	Preset 1	100	RPM
	Preset 2	1000	RPM
	Preset 3	0	RPM
	Preset 4	-10	RPM
	Preset 5	-100	RPM
	Preset 6	-1000	RPM
	Preset 7	0	RPM
⊟	**Preset Velocity Input Limits**		
	Accel/Decel Limits	Inactive	
	Acceleration	13	Revs/s^2
	Deceleration	13	Revs/s^2

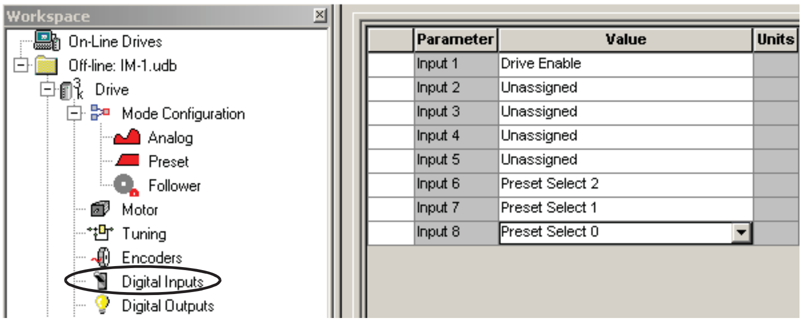
6. Enter the Preset Velocity values as shown in the table above or otherwise appropriate to your application.

7. Set the Preset Velocity Input Limits value to Inactive.

8. Close the Preset dialogs.

9. Double-click the Digital Inputs branch.

10. Click the Value fields and use the pull-down menus to change the input values as described below.



Parameter	Value	Units
Input 1	Drive Enable	
Input 2	Unassigned	
Input 3	Unassigned	
Input 4	Unassigned	
Input 5	Unassigned	
Input 6	Preset Select 2	
Input 7	Preset Select 1	
Input 8	Preset Select 0	

11. Using this table, determine the sequence of these three inputs that correspond to the preset velocity entered.

Preset Selects	Binary Code						Selected Preset or Index
	5	4	3	2	1	0	
Select up to 64 locations via preselect inputs 5…0 by using BCD format. (codes for preset selects 1 and 0 are shown)	0	0	0	0	0	0	Preset 0 or Index 0 is selected.
	0	0	0	0	0	1	Preset 1 or Index 1 is selected.
	0	0	0	0	1	0	Preset 2 or Index 2 is selected.
	0	0	0	0	1	1	Preset 3 or Index 3 is selected.
	↓						↓
	1	1	1	1	1	1	Preset 64 or Index 64 is selected.

12. Apply 12…24V dc to input 1.

Input 1 was configured as Drive Enable in a previous step.

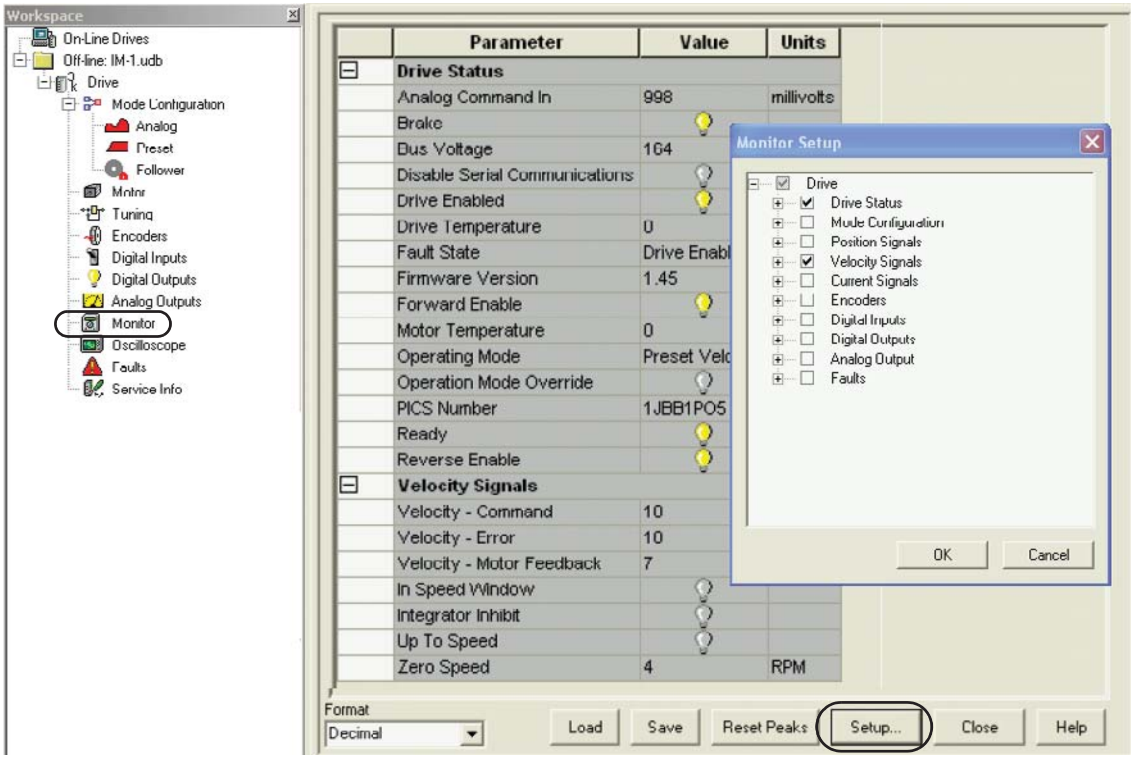
a. Verify the toolbar Enable icon is active, indicating the drive is enabled.

b. Verify the Drive Enabled lamp is ON (yellow)

c. If none of the Preset Selects are ON, observe the motor running at the selected speed (rpm) for Preset 0 (10 rpm in this example).

13. Double-click the Monitor branch.

The (default) Drive Status parameters display.



	Parameter	Value	Units
⊟	**Drive Status**		
	Analog Command In	998	millivolts
	Brake	●	
	Bus Voltage	164	
	Disable Serial Communications	○	
	Drive Enabled	●	
	Drive Temperature	0	
	Fault State	Drive Enabl	
	Firmware Version	1.45	
	Forward Enable	●	
	Motor Temperature	0	
	Operating Mode	Preset Velc	
	Operation Mode Override	○	
	PICS Number	1JBB1PO5	
	Ready	●	
	Reverse Enable	●	
⊟	**Velocity Signals**		
	Velocity - Command	10	
	Velocity - Error	10	
	Velocity - Motor Feedback	7	
	In Speed Window	○	
	Integrator Inhibit	○	
	Up To Speed	○	
	Zero Speed	4	RPM

Monitor Setup

- ☑ Drive
  - ☑ Drive Status
  - ☐ Mode Configuration
  - ☐ Position Signals
  - ☑ Velocity Signals
  - ☐ Current Signals
  - ☐ Encoders
  - ☐ Digital Inputs
  - ☐ Digital Outputs
  - ☐ Analog Output
  - ☐ Faults

14. Click Setup.

The Monitor Setup dialog opens.

15. In the Monitor Setup dialog, check Velocity Signals.

16. Click OK.

The Monitor Status dialog closes and the setup changes take affect.

17. Observe that Velocity - Command matches what was entered in Preset Velocity 0.

18. Observe the Velocity - Motor Feedback continually updating to maintain the commanded velocity.

19. Apply 12…24V dc to Preset Select 0 configured as Digital Input 8 (CN1-38).

The Velocity - Command now matches Preset 1 (100 rpm in this example).

20. Remove the 12…24V dc (Drive Enable) from input 1.

21. Close the Monitor Status dialog and Digital Inputs dialog.

Preset Position Control (indexing move)

This procedure assumes you have applied power to your drive, the Ultraware software is running, the drive is detected, and you have tested a motor. In this procedure you will run the drive by using preset position control.

Refer to the Ultraware User Manual, publication 2098-UM001, for more information on preset position control.

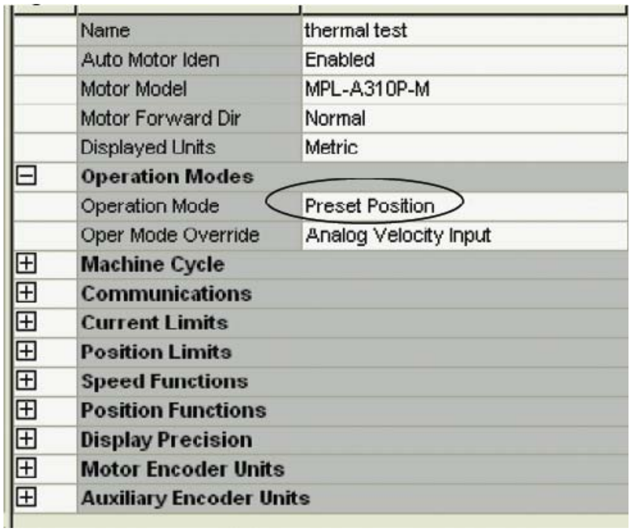
Follow these steps to use preset position control.

1. Double-click the U3k icon.

The drive properties dialog opens.

2. Expand the Operation Modes parameter.

3. Click the current setting and use the pull-down menu to change the Operation Mode to Preset Position.



	Name	thermal test
	Auto Motor Iden	Enabled
	Motor Model	MPL-A310P-M
	Motor Forward Dir	Normal
	Displayed Units	Metric
⊟	**Operation Modes**	
	Operation Mode	Preset Position
	Oper Mode Override	Analog Velocity Input
⊞	**Machine Cycle**	
⊞	**Communications**	
⊞	**Current Limits**	
⊞	**Position Limits**	
⊞	**Speed Functions**	
⊞	**Position Functions**	
⊞	**Display Precision**	
⊞	**Motor Encoder Units**	
⊞	**Auxiliary Encoder Units**	

4. Close the Drive Branch dialog.

5. Expand the Mode Configuration branch and double-click Preset.

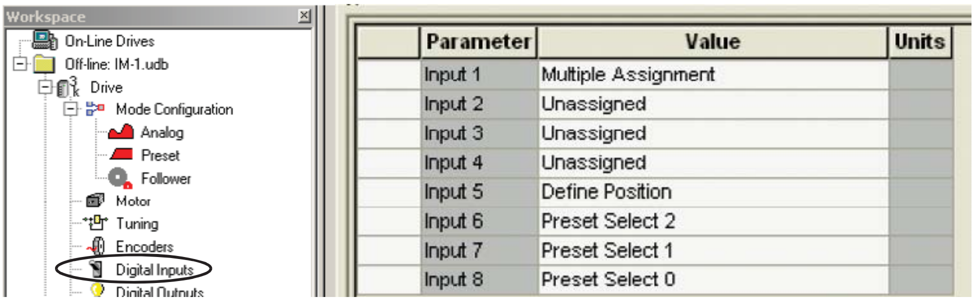
The Preset setup dialog opens.

⊟	**Preset Position 0 Setup**		
	Position	8192	Counts
	Velocity	750	RPM
	Acceleration	12	Revs/s^2
	Deceleration	12	Revs/s^2
⊟	**Preset Position 1 Setup**		
	Position	4096	Counts
	Velocity	750	RPM
	Acceleration	12	Revs/s^2
	Deceleration	12	Revs/s^2

6. Enter the Preset Velocity values as shown in the table above or otherwise appropriate to your application.

7. Double-click the Digital Inputs branch.

8. Click the Value fields and use the pull-down menus to change the input values as described below.



Parameter	Value	Units
Input 1	Multiple Assignment	
Input 2	Unassigned	
Input 3	Unassigned	
Input 4	Unassigned	
Input 5	Define Position	
Input 6	Preset Select 2	
Input 7	Preset Select 1	
Input 8	Preset Select 0	

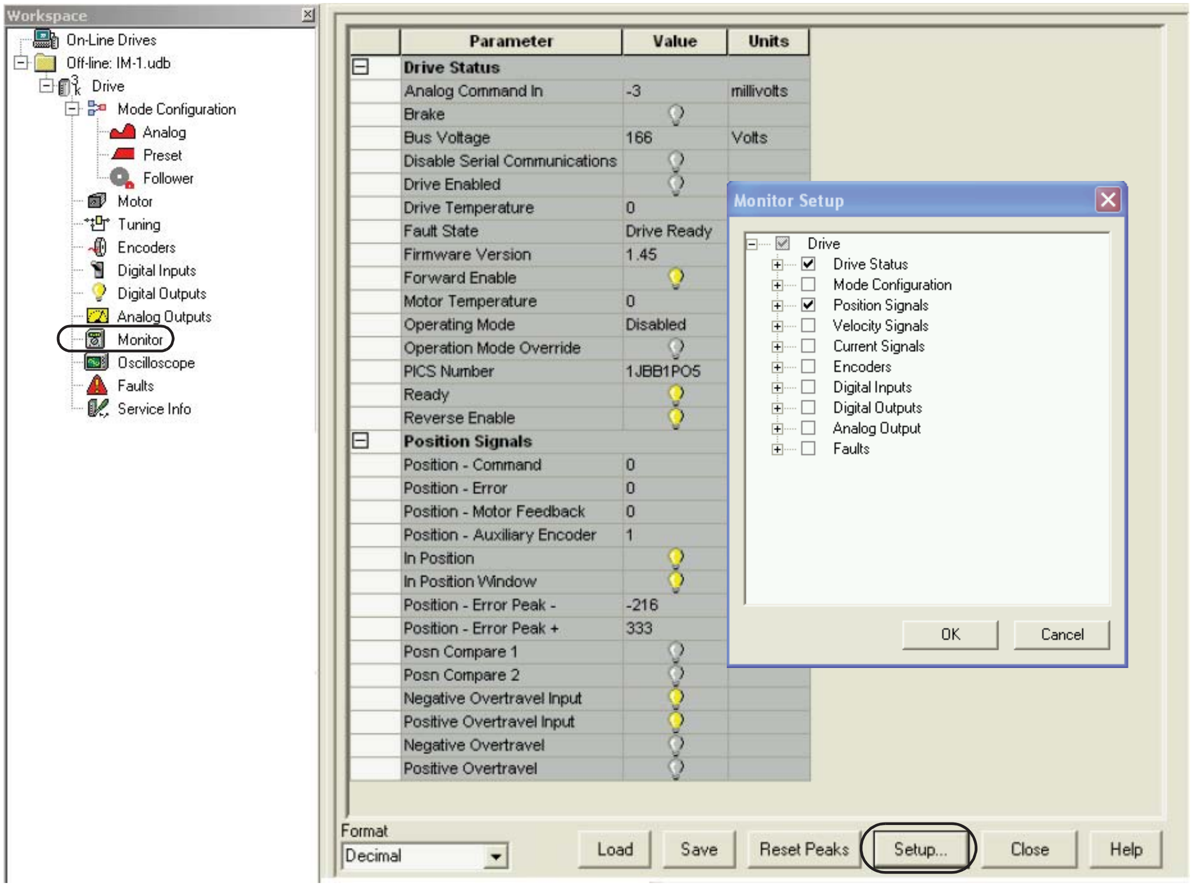
9. Using this table, determine the sequence of these three inputs that correspond to the preset positions entered.

Preset Selects	**Binary Code**						**Selected Preset or Index**
	5	4	3	2	1	0	
Select up to 64 locations via preselect inputs 5…0 by using BCD format. (codes for preset selects 1 and 0 are shown)	0	0	0	0	0	0	Preset 0 or Index 0 is selected.
	0	0	0	0	0	1	Preset 1 or Index 1 is selected.
	0	0	0	0	1	0	Preset 2 or Index 2 is selected.
	0	0	0	0	1	1	Preset 3 or Index 3 is selected.
			↓				↓
	1	1	1	1	1	1	Preset 64 or Index 64 is selected.

In this example, Preset Position 0 and 1 are configured so Preset Select 0 is either off (Preset Position 0) or on (Preset Position 1).

10. Double-click the Monitor branch.

The (default) Drive Status parameters display.



	Parameter	Value	Units
⊟	**Drive Status**		
	Analog Command In	-3	millivolts
	Brake	○	
	Bus Voltage	166	Volts
	Disable Serial Communications	○	
	Drive Enabled	○	
	Drive Temperature	0	
	Fault State	Drive Ready	
	Firmware Version	1.45	
	Forward Enable	●	
	Motor Temperature	0	
	Operating Mode	Disabled	
	Operation Mode Override	○	
	PICS Number	1JBB1PO5	
	Ready	●	
	Reverse Enable	●	
⊟	**Position Signals**		
	Position - Command	0	
	Position - Error	0	
	Position - Motor Feedback	0	
	Position - Auxiliary Encoder	1	
	In Position	●	
	In Position Window	●	
	Position - Error Peak -	-216	
	Position - Error Peak +	333	
	Posn Compare 1	○	
	Posn Compare 2	○	
	Negative Overtravel Input	●	
	Positive Overtravel Input	●	
	Negative Overtravel	○	
	Positive Overtravel	○	

Format: Decimal

Monitor Setup

- ☑ Drive
  - ☑ Drive Status
  - ☐ Mode Configuration
  - ☑ Position Signals
  - ☐ Velocity Signals
  - ☐ Current Signals
  - ☐ Encoders
  - ☐ Digital Inputs
  - ☐ Digital Outputs
  - ☐ Analog Output
  - ☐ Faults

11. Click Setup.

The Monitor Setup dialog opens.

12. In the Monitor Setup dialog, check Position Signals.

13. Click OK.

The Monitor Status dialog closes and the setup changes take affect.

14. Apply 12…24V dc to input 1.

Input 1 was configured as Drive Enable in a previous step.

a. Verify the toolbar Enable icon is active, indicating the drive is enabled.

b. Verify the Drive Enabled lamp is ON (yellow)

c. If none of the Preset Selects are ON, observe the motor move to Preset Position 0.

In this example, 1 revolution (8192 counts).

15. Apply 12…24V dc to Preset Select 0 configured as Digital Input 8 (CN1-38).

The motor moves to Preset Position 1 (4096 counts).

TIP Preset Positions are absolute and not incremental position.

16. Close the Preset, Monitor Branch, and Digital Inputs dialog.

Master Follower and Preset Gear Ratios (non-indexing move)

This procedure assumes you have applied power to your drive, the Ultraware software is running, the drive is detected, and you have tested a motor.

An external auxiliary encoder is powered by the Ultra3000 drive through pins CN1-1 and CN1-2. The encoder signals are wired to pins CN1-4…CN1-9. In this procedure, you will run the drive in Follower-Auxiliary Encoder mode.

Refer to the Ultraware User Manual, publication 2098-UM001, for more information on Position Follower mode.

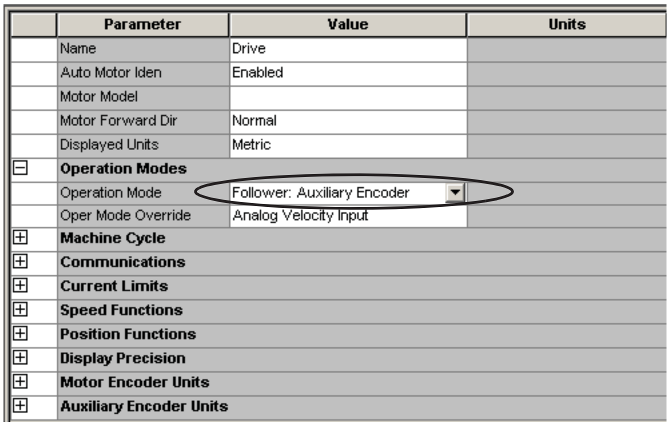
Follow these steps to run the drive in Follower-Auxiliary Encoder mode.

1. Double-click the U3k icon.

The drive properties dialog opens.

2. Expand the Operation Modes parameter.

3. Click the current setting and use the pull-down menu to change the Operation Mode to Follower: Auxiliary Encoder.



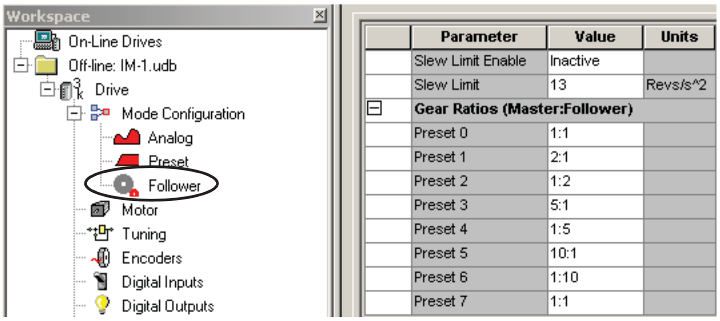
	Parameter	Value	Units
	Name	Drive	
	Auto Motor Iden	Enabled	
	Motor Model		
	Motor Forward Dir	Normal	
	Displayed Units	Metric	
⊟	**Operation Modes**		
	Operation Mode	Follower: Auxiliary Encoder	
	Oper Mode Override	Analog Velocity Input	
⊞	**Machine Cycle**		
⊞	**Communications**		
⊞	**Current Limits**		
⊞	**Speed Functions**		
⊞	**Position Functions**		
⊞	**Display Precision**		
⊞	**Motor Encoder Units**		
⊞	**Auxiliary Encoder Units**		

4. Close the Drive Branch dialog.

5. Expand the Mode Configuration branch.

6. Double-click Follower.

7. Enter the Gear Ratio preset values as shown in the table below or according to your specific application.

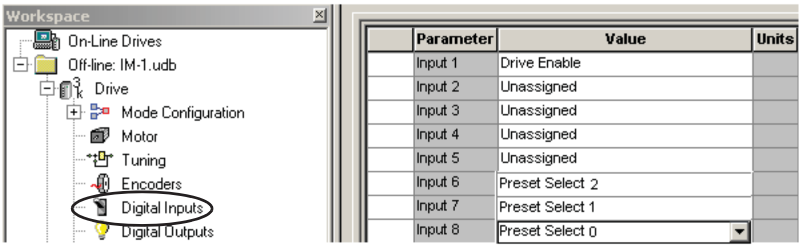


	Parameter	Value	Units
	Slew Limit Enable	Inactive	
	Slew Limit	13	Revs/s^2
⊟	**Gear Ratios (Master:Follower)**		
	Preset 0	1:1	
	Preset 1	2:1	
	Preset 2	1:2	
	Preset 3	5:1	
	Preset 4	1:5	
	Preset 5	10:1	
	Preset 6	1:10	
	Preset 7	1:1	

8. Close the Mode Configuration dialog.

9. Double-click the Digital Inputs branch.

10. Use the pull-down menu to change the input values.



	Parameter	Value	Units
	Input 1	Drive Enable	
	Input 2	Unassigned	
	Input 3	Unassigned	
	Input 4	Unassigned	
	Input 5	Unassigned	
	Input 6	Preset Select 2	
	Input 7	Preset Select 1	
	Input 8	Preset Select 0	

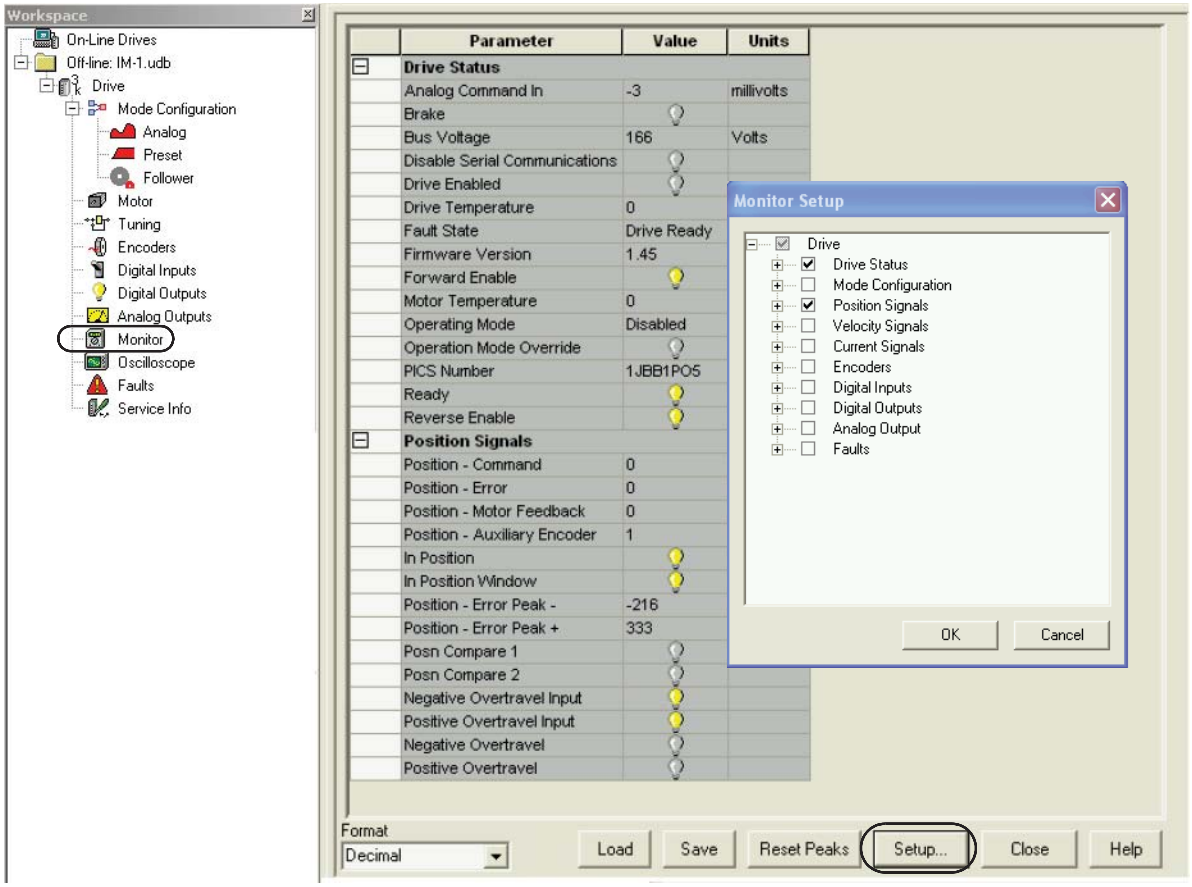
11. Using this table, determine the sequence of these three inputs that correspond to the preset gear ratios entered.

Preset Selects	Binary Code						Selected Preset or Index
	5	4	3	2	1	0	
Select up to 64 locations via preselect inputs 5…0 by using BCD format. (codes for preset selects 1 and 0 are shown)	0	0	0	0	0	0	Preset 0 or Index 0 is selected.
	0	0	0	0	0	1	Preset 1 or Index 1 is selected.
	0	0	0	0	1	0	Preset 2 or Index 2 is selected.
	0	0	0	0	1	1	Preset 3 or Index 3 is selected.
			↓				↓
	1	1	1	1	1	1	Preset 64 or Index 64 is selected.

In this example, preset gear ratio 0 and 1 are configured.

12. Double-click the Monitor branch.

The (default) Drive Status parameters display.



Parameter	Value	Units
Drive Status		
Analog Command In	-3	millivolts
Brake	○	
Bus Voltage	166	Volts
Disable Serial Communications	○	
Drive Enabled	○	
Drive Temperature	0	
Fault State	Drive Ready	
Firmware Version	1.45	
Forward Enable	●	
Motor Temperature	0	
Operating Mode	Disabled	
Operation Mode Override	○	
PICS Number	1JBB1PO5	
Ready	●	
Reverse Enable	●	
Position Signals		
Position - Command	0	
Position - Error	0	
Position - Motor Feedback	0	
Position - Auxiliary Encoder	1	
In Position	●	
In Position Window	●	
Position - Error Peak -	-216	
Position - Error Peak +	333	
Posn Compare 1	○	
Posn Compare 2	○	
Negative Overtravel Input	●	
Positive Overtravel Input	●	
Negative Overtravel	○	
Positive Overtravel	○	

Monitor Setup

- ☑ Drive
  - ☑ Drive Status
  - ☐ Mode Configuration
  - ☑ Position Signals
  - ☐ Velocity Signals
  - ☐ Current Signals
  - ☐ Encoders
  - ☐ Digital Inputs
  - ☐ Digital Outputs
  - ☐ Analog Output
  - ☐ Faults

13. Click Setup.

The Monitor Setup dialog opens.

14. In the Monitor Setup dialog, check Position Signals.

15. Click OK.

The Monitor Status dialog closes and the setup changes take affect.

16. Apply 12…24V dc to input 1.

Input 1 was configured as Drive Enable in a previous step.

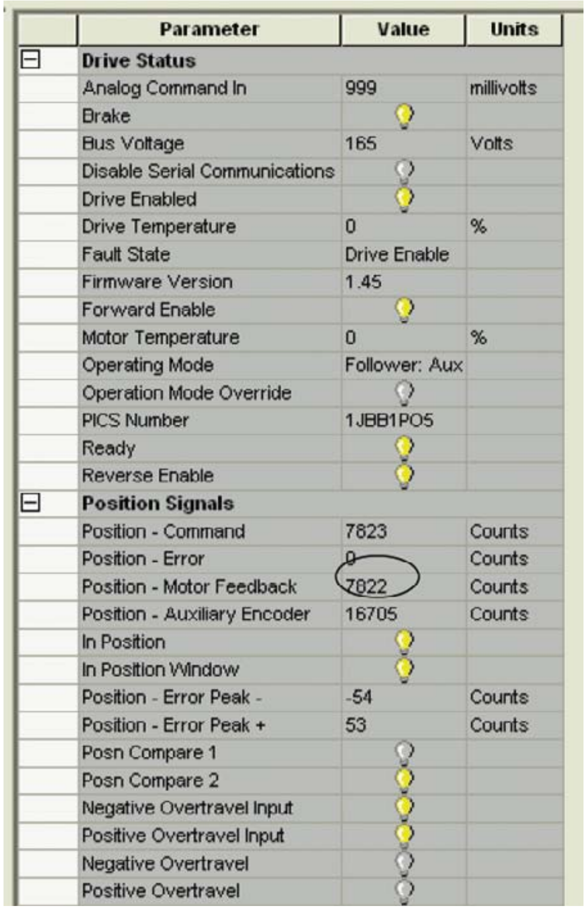
a. Verify the toolbar Enable icon is active, indicating the drive is enabled.

b. Verify the Drive Enabled lamp is ON (yellow)

c. If none of the Presets are ON, move the auxiliary encoder and observe the motor rotate at Preset 0 Gear Ratio or 1:1.

17. Apply 12…24V dc to Preset Select 0 which is configured for Digital Input 8 or pin CN1-38.

Notice that the auxiliary encoder uses Preset 1 as the Gear Ratio or 2:1. This means for every two revolutions of the auxiliary encoder, the motor rotates 1 revolution.



	Parameter	Value	Units
⊟	**Drive Status**		
	Analog Command In	999	millivolts
	Brake	●	
	Bus Voltage	165	Volts
	Disable Serial Communications	○	
	Drive Enabled	●	
	Drive Temperature	0	%
	Fault State	Drive Enable	
	Firmware Version	1.45	
	Forward Enable	●	
	Motor Temperature	0	%
	Operating Mode	Follower: Aux	
	Operation Mode Override	○	
	PICS Number	1JBB1PO5	
	Ready	●	
	Reverse Enable	●	
⊟	**Position Signals**		
	Position - Command	7823	Counts
	Position - Error	0	Counts
	Position - Motor Feedback	7822	Counts
	Position - Auxiliary Encoder	16705	Counts
	In Position	●	
	In Position Window	●	
	Position - Error Peak -	-54	Counts
	Position - Error Peak +	53	Counts
	Posn Compare 1	○	
	Posn Compare 2	●	
	Negative Overtravel Input	●	
	Positive Overtravel Input	●	
	Negative Overtravel	○	
	Positive Overtravel	○	

18. Remove the 12…24V dc (Drive Enable) from input 1.

19. Close the Monitor Branch and Digital Inputs dialog.

Incremental Indexing (indexing move)

This procedure assumes you have applied power to your indexing drive, the Ultraware software is running, the drive is detected, and you have tested a motor. In this procedure you will run the drive in Incremental Indexing mode.

Refer to the Ultraware User Manual, publication 2098-UM001, for more information on incremental indexing moves.

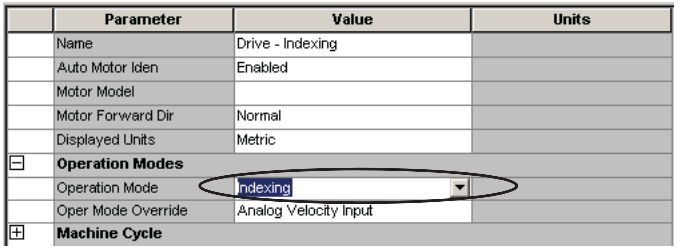
Follow these steps to set parameters for an incremental indexing move.

1. Double-click the U3k icon.

   The drive properties dialog opens.

2. Expand the Operation Modes parameter.

3. Click the current setting and use the pull-down menu to change the Operation Mode to Indexing.



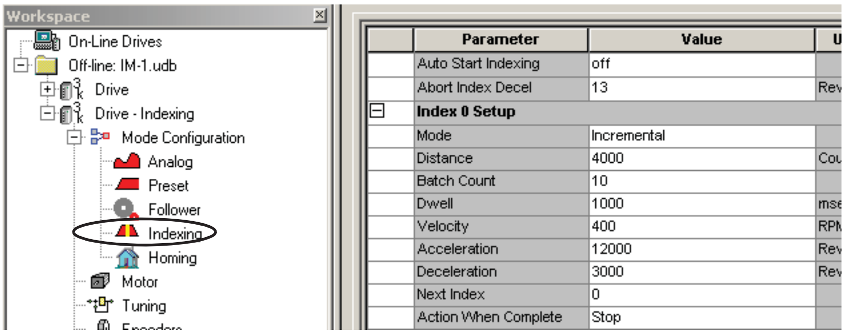
	Parameter	Value	Units
	Name	Drive - Indexing	
	Auto Motor Iden	Enabled	
	Motor Model		
	Motor Forward Dir	Normal	
	Displayed Units	Metric	
⊟	**Operation Modes**		
	Operation Mode	Indexing	
	Oper Mode Override	Analog Velocity Input	
⊞	**Machine Cycle**		

4. Close the Drive Branch dialog.

5. Expand the Mode Configuration branch.

6. Double-click Indexing.

7. Enter the Index 0 parameter values.

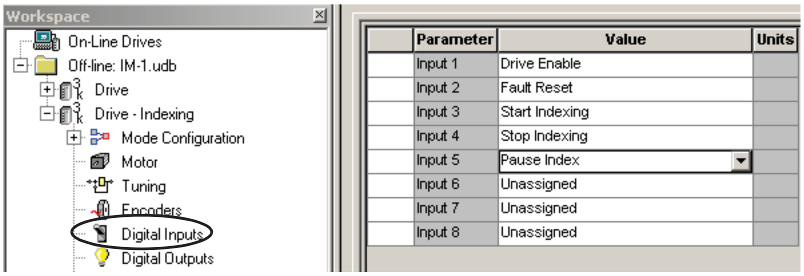


	Parameter	Value	U
	Auto Start Indexing	off	
	Abort Index Decel	13	Rev
⊟	**Index 0 Setup**		
	Mode	Incremental	
	Distance	4000	Cou
	Batch Count	10	
	Dwell	1000	mse
	Velocity	400	RP
	Acceleration	12000	Rev
	Deceleration	3000	Rev
	Next Index	0	
	Action When Complete	Stop	

8. Close the Indexing Parameters dialog.

9. Double-click the Digital Inputs branch.

10. Use the pull-down menu to change the input values.

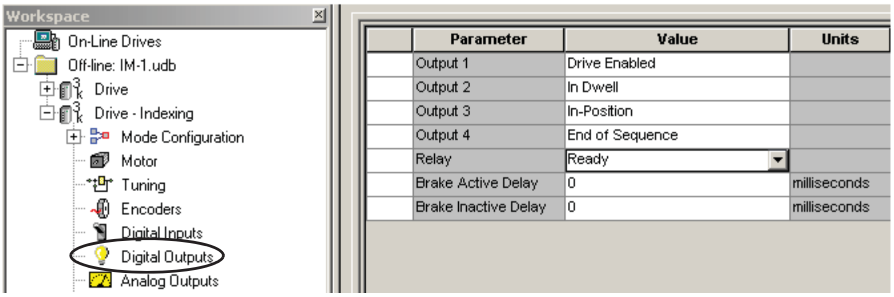


Parameter	Value	Units
Input 1	Drive Enable	
Input 2	Fault Reset	
Input 3	Start Indexing	
Input 4	Stop Indexing	
Input 5	Pause Index	
Input 6	Unassigned	
Input 7	Unassigned	
Input 8	Unassigned	

11. Close the Digital Inputs dialog.

12. Double-click the Digital Outputs branch.

13. Use the pull-down menu to change the output values.



Parameter	Value	Units
Output 1	Drive Enabled	
Output 2	In Dwell	
Output 3	In-Position	
Output 4	End of Sequence	
Relay	Ready	
Brake Active Delay	0	milliseconds
Brake Inactive Delay	0	milliseconds

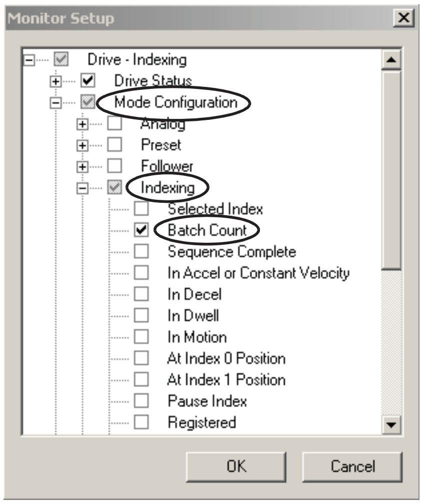
14. Close the Digital Outputs dialog.

Follow these steps to verify the number of indexing moves by using drive signals.

1. Double-click the Monitor branch.

2. Click Setup.

3. Expand the Mode Configuration branch/the Indexing branch/and check Batch Count.



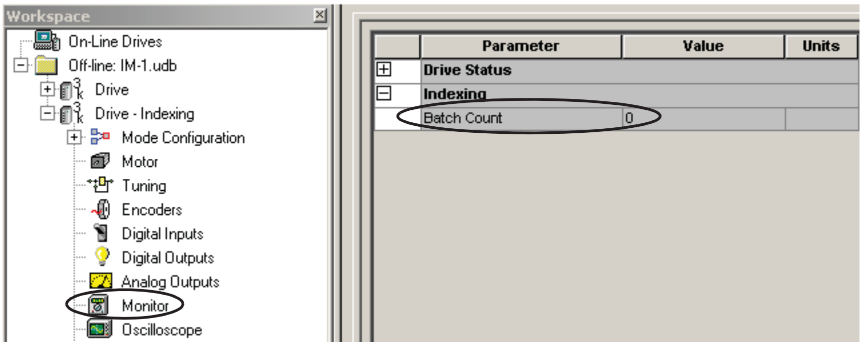
- ☑ Drive - Indexing
  - ☑ Drive Status
  - ☑ Mode Configuration
    - ☐ Analog
    - ☐ Preset
    - ☐ Follower
    - ☑ Indexing
      - ☐ Selected Index
      - ☑ Batch Count
      - ☐ Sequence Complete
      - ☐ In Accel or Constant Velocity
      - ☐ In Decel
      - ☐ In Dwell
      - ☐ In Motion
      - ☐ At Index 0 Position
      - ☐ At Index 1 Position
      - ☐ Pause Index
      - ☐ Registered

4. Click OK.

5. Apply 12…24V dc to input 1.

   Input 1 was configured as Drive Enable in a previous step.

6. Apply 12…24V dc to input 3 to the indexing move.



	Parameter	Value	Units
⊞	**Drive Status**		
⊟	**Indexing**		
	Batch Count	0	

7. Double-click the Monitor branch and watch Batch Count count down from 10 to 0.

8. Observe Outputs 2 and 3 for axis in dwell and in position.

9. Observe Output 4 when the Indexing move is complete.

10. Remove the 12…24V dc (Drive Enable) from input 1.

Follow these steps to use the stop indexing feature.

1. Apply 12…24V dc to input 1.

   Input 1 was configured as Drive Enable in a previous step.

2. Apply 12…24V dc to input 3 to the indexing move.

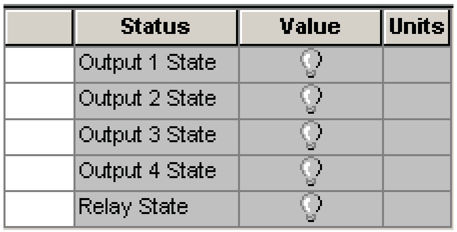
3. Apply 12…24V dc to input 4 and verify that the indexing move has stopped.

4. Apply 12…24V dc to input 3 (again) and verify the original indexing move is re-initiated.

5. Apply 12…24V dc to input 5 and verify the index move is paused.

6. Double-click the Digital Outputs branch.

7. Observe that Output 4 is not illuminated, indicating end of sequence has not been reached.



	Status	Value	Units
	Output 1 State	💡	
	Output 2 State	💡	
	Output 3 State	💡	
	Output 4 State	💡	
	Relay State	💡	

8. Close the Digital Outputs dialog.

9. Observe the Monitor branch to see that the Batch Count value is held at the remaining value.

10. Remove the 12…24V dc from Input 5 and verify the indexing move continues.

11. Close the dialogs.

12. Remove the 12…24V dc (Drive Enable) from input 1.

Absolute Indexing (indexing move)

This procedure assumes you have applied power to your indexing drive, the Ultraware software is running, the drive is detected, and you have tested a motor. In this procedure you will run the drive in Absolute Indexing mode.

Refer to the Ultraware User Manual, publication 2098-UM001, for more information on absolute indexing moves.

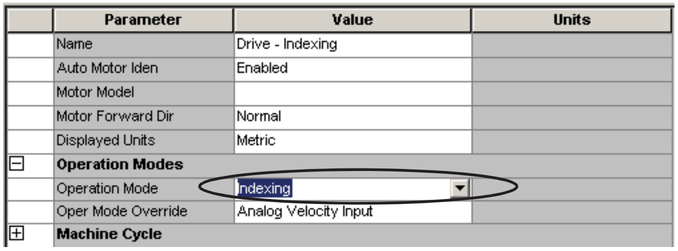
Follow these steps to set parameters for an absolute indexing move.

1. Double-click the U3k icon.

The drive properties dialog opens.

2. Expand the Operation Modes parameter.

3. Click the current setting and use the pull-down menu to change the Operation Mode to Indexing.



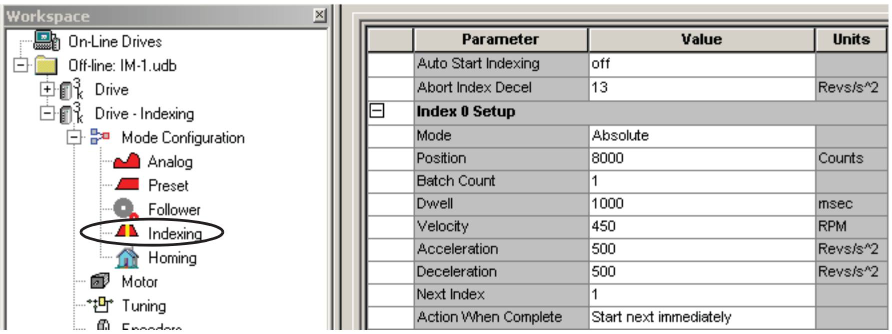
	Parameter	Value	Units
	Name	Drive - Indexing	
	Auto Motor Iden	Enabled	
	Motor Model		
	Motor Forward Dir	Normal	
	Displayed Units	Metric	
⊟	**Operation Modes**		
	Operation Mode	Indexing	
	Oper Mode Override	Analog Velocity Input	
⊞	**Machine Cycle**		

4. Close the Drive Branch dialog.

5. Expand the Mode Configuration branch.

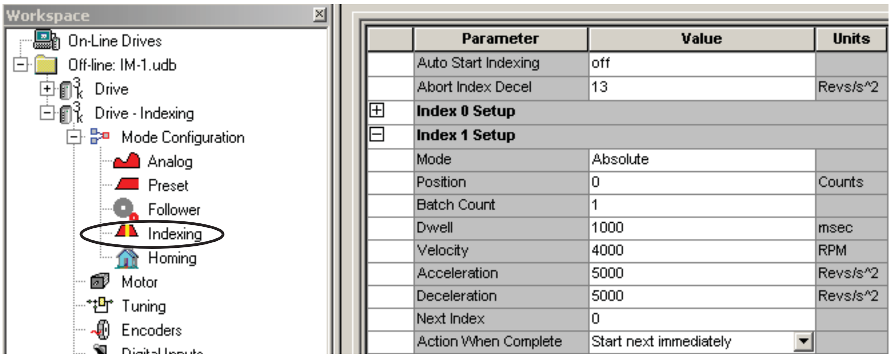
6. Double-click Indexing.

7. Enter the Index 0 parameter values as shown in the table below.



	Parameter	Value	Units
	Auto Start Indexing	off	
	Abort Index Decel	13	Revs/s^2
⊟	**Index 0 Setup**		
	Mode	Absolute	
	Position	8000	Counts
	Batch Count	1	
	Dwell	1000	msec
	Velocity	450	RPM
	Acceleration	500	Revs/s^2
	Deceleration	500	Revs/s^2
	Next Index	1	
	Action When Complete	Start next immediately	

8. Enter the Index 1 parameter values as shown in the table below.



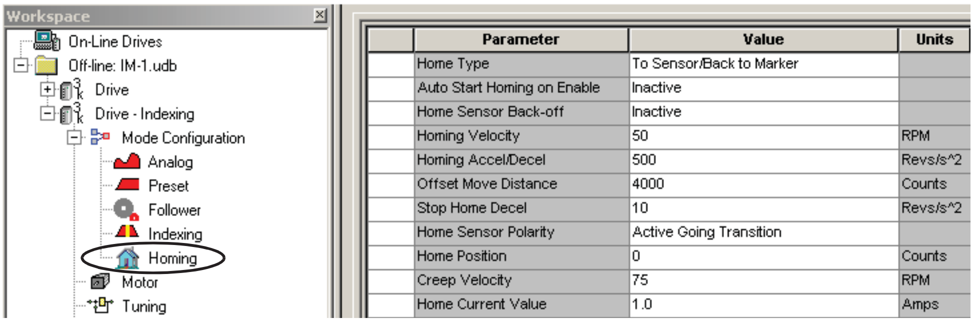
	Parameter	Value	Units
	Auto Start Indexing	off	
	Abort Index Decel	13	Revs/s^2
⊞	**Index 0 Setup**		
⊟	**Index 1 Setup**		
	Mode	Absolute	
	Position	0	Counts
	Batch Count	1	
	Dwell	1000	msec
	Velocity	4000	RPM
	Acceleration	5000	Revs/s^2
	Deceleration	5000	Revs/s^2
	Next Index	0	
	Action When Complete	Start next immediately	

9. Close the Indexing Parameters dialog.

10. Expand the Mode Configuration branch.

11. Double-click Homing.

12. Enter the Homing parameter values as shown in the table below.



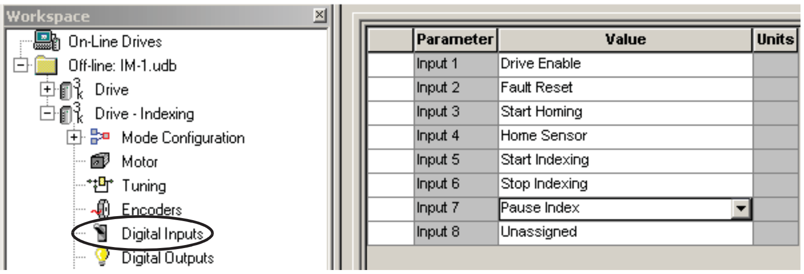
Parameter	Value	Units
Home Type	To Sensor/Back to Marker	
Auto Start Homing on Enable	Inactive	
Home Sensor Back-off	Inactive	
Homing Velocity	50	RPM
Homing Accel/Decel	500	Revs/s^2
Offset Move Distance	4000	Counts
Stop Home Decel	10	Revs/s^2
Home Sensor Polarity	Active Going Transition	
Home Position	0	Counts
Creep Velocity	75	RPM
Home Current Value	1.0	Amps

13. Close the Homing Parameters dialog.

14. Close the Mode Configuration dialog.

15. Double-click the Digital Inputs branch.

16. Use the pull-down menu to change the input values.

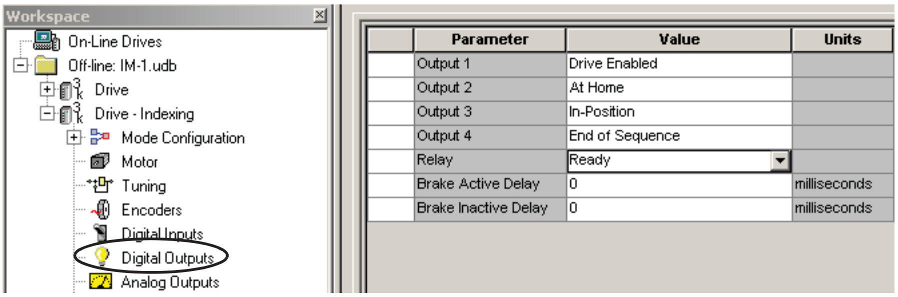


Parameter	Value	Units
Input 1	Drive Enable	
Input 2	Fault Reset	
Input 3	Start Homing	
Input 4	Home Sensor	
Input 5	Start Indexing	
Input 6	Stop Indexing	
Input 7	Pause Index	
Input 8	Unassigned	

17. Close the Digital Inputs dialog.

Follow these steps to use digital outputs to indicate an event has occurred.

1. Double-click the Digital Outputs branch.
2. Use the pull-down menu to change the output values.

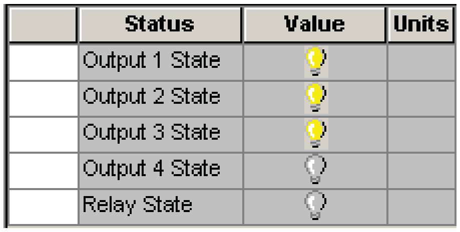


	Parameter	Value	Units
	Output 1	Drive Enabled	
	Output 2	At Home	
	Output 3	In-Position	
	Output 4	End of Sequence	
	Relay	Ready	
	Brake Active Delay	0	milliseconds
	Brake Inactive Delay	0	milliseconds

3. Close the Digital Outputs dialog.
4. Apply 12…24V dc to input 1.

   Input 1 was configured as Drive Enable in a previous step.
5. Apply 12…24V dc to input 3 (momentarily) to start the homing routine.
6. Apply 12…24V dc to input 4 (momentarily) to simulate a homing sensor.

   The drive goes into reverse to find the marker and completes the homing routine.
7. Double-click the Digital Outputs branch.
8. Observe Digital Outputs status.



	Status	Value	Units
	Output 1 State	●	
	Output 2 State	●	
	Output 3 State	●	
	Output 4 State	○	
	Relay State	○	

   - Output 1 is on because the drive is enabled.
   - Output 2 is on because the drive has been homed.
   - Output 3 is on because the motor is in position.

9. Apply 12…24V dc to input 5 and observe Digital Outputs 2 and 3 change states.
10. Apply 12…24V dc to input 6 (momentarily) to stop the indexing move.

11. Turn off input 5.
12. Apply 12…24V dc to input 4 (momentarily again) to restart the indexing move.
13. Turn off input 4.
14. Apply 12…24V dc to input 7 to pause the indexing move.
15. Remove the 12…24V dc and observe the index move continue.
16. Close the dialogs.
17. Remove the 12…24V dc (Drive Enable) from input 1.

Configuring Your Ultra3000 Drive with RSLogix 5000 Software

In this section you will configure your Ultra3000 drive by using Ultraware software, configure the Logix analog motion module by using RSLogix 5000 software, and test/tune your axis.

Configure Your Ultra3000 Drive

Follow these steps to configure your Ultra3000 drive.

1. Apply power to your Ultra3000 drive

   Refer to the section Apply Power To Your Ultra3000 Drive.

2. Start your Ultraware software and make sure your Ultra3000 drive is detected.

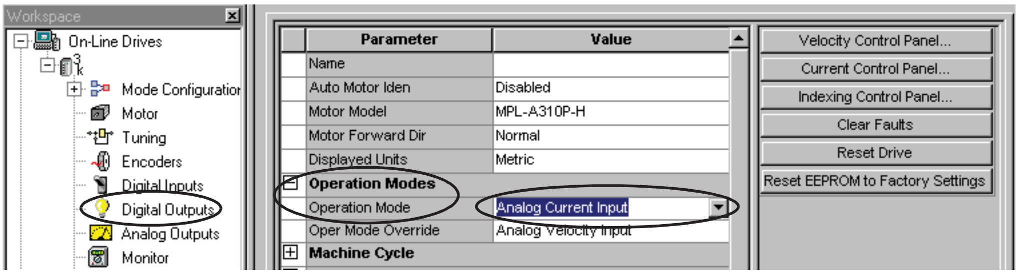
   Refer to the section Detect Your Ultra3000 Drive.

3. Select a motor.

   Refer to the section Select a Motor.

4. Expand Operation Modes in the Drive properties dialog.

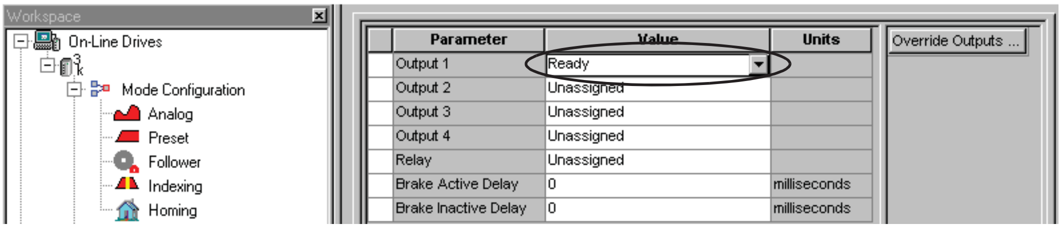
5. From the Operation Mode pull-down menu, choose Analog Current Input.



Parameter	Value
Name	
Auto Motor Iden	Disabled
Motor Model	MPL-A310P-H
Motor Forward Dir	Normal
Displayed Units	Metric
⊟ **Operation Modes**	
Operation Mode	Analog Current Input
Oper Mode Override	Analog Velocity Input
⊞ **Machine Cycle**	

6. Double-click Digital Outputs.

   The Digital Output properties dialog opens.



Parameter	Value	Units
Output 1	Ready	
Output 2	Unassigned	
Output 3	Unassigned	
Output 4	Unassigned	
Relay	Unassigned	
Brake Active Delay	0	milliseconds
Brake Inactive Delay	0	milliseconds

7. From the Output 1 pull-down menu, choose Ready.

Configuring Your Logix Analog Motion Module

This procedure assumes that you have finished configuring your Ultra3000 drive.

For help using RSLogix 5000 software as it applies to configuring the Logix analog modules, refer to Additional Resources on page 8.

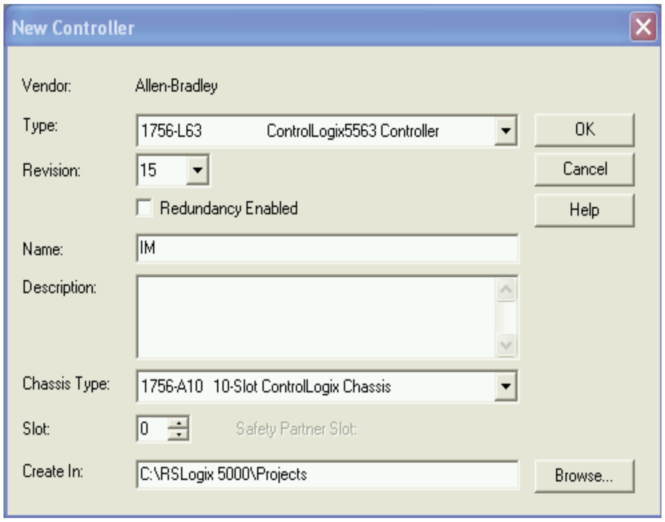
Configure Your Logix Controller

Follow these steps to configure your Logix controller.

1. Apply power to your Logix chassis containing the analog motion module and open your RSLogix 5000 software.

2. From the File menu, choose New.

The New Controller dialog opens. The ControlLogix 1756-L63 controller was used in this example.



Vendor: Allen-Bradley
Type: 1756-L63 ControlLogix5563 Controller
Revision: 15
☐ Redundancy Enabled
Name: IM
Description:
Chassis Type: 1756-A10 10-Slot ControlLogix Chassis
Slot: 0 Safety Partner Slot:
Create In: C:\RSLogix 5000\Projects

3. Configure the new controller.

a. From the Type pull-down menu, choose your controller.

b. From the Revision pull-down menu, choose your RSLogix 5000 software version.

c. In the Name box, name your file.

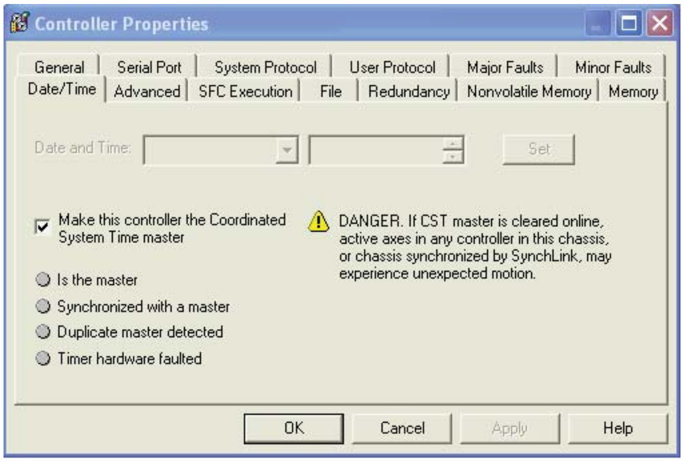
d. From the Chassis Type pull-down menu, choose your Logix chassis.

e. Enter the Logix processor slot.

4. Click OK.

5. From the Edit menu, choose Controller Properties.

The Controller Properties dialog opens.



☑ Make this controller the Coordinated System Time master

DANGER. If CST master is cleared online, active axes in any controller in this chassis, or chassis synchronized by SynchLink, may experience unexpected motion.

○ Is the master
○ Synchronized with a master
○ Duplicate master detected
○ Timer hardware faulted

6. Click the Date/Time tab.

7. Check the Make this controller the Coordinated System Time master checkbox.

IMPORTANT Only one Logix processor can be assigned as the Coordinated System Time master.

8. Click OK.

Configure Your Logix Module

Follow these steps to configure your Logix module.

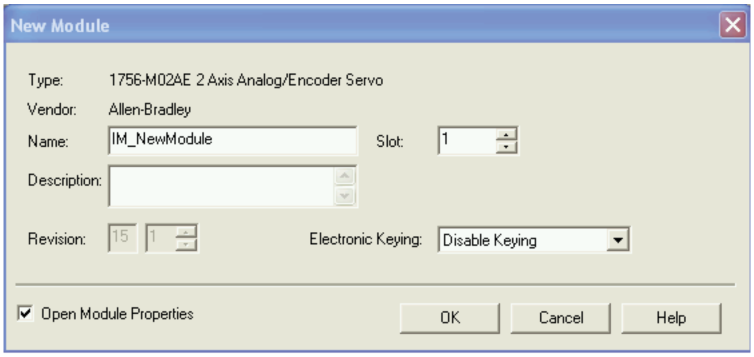
1. In the Explorer dialog, right-click I/O Configuration and choose New Module.

   The Select Module dialog opens.

2. Expand the Motion category and select 1756-M02AE, 1756-HYD02, 1756-M02AS, or 1784-PM02AE as appropriate for your actual hardware configuration.

3. Click OK.

   Your new module appears under the I/O Configuration folder in the Explorer dialog and the New Module dialog opens.



Type: 1756-M02AE 2 Axis Analog/Encoder Servo
Vendor: Allen-Bradley
Name: IM_NewModule
Slot: 1
Description:
Revision: 15 1
Electronic Keying: Disable Keying
☑ Open Module Properties

4. Configure the new module.

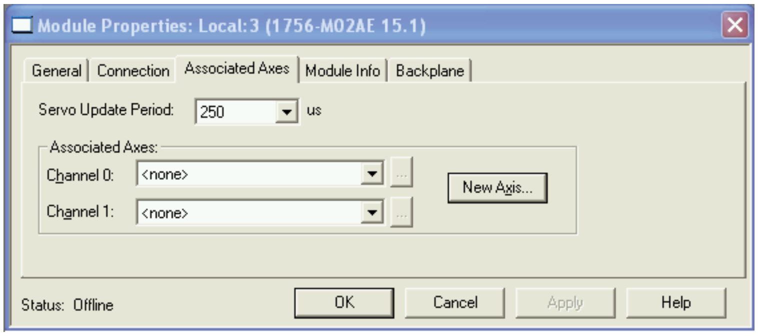
   a. In the Name box, enter your module name

   b. Enter the module slot.

   c. From the Electronic Keying pull-down menu, choose your keying option

5. Click OK.

   Your new module appears under the I/O Configuration folder in the Explorer dialog and the Module Properties dialog opens.

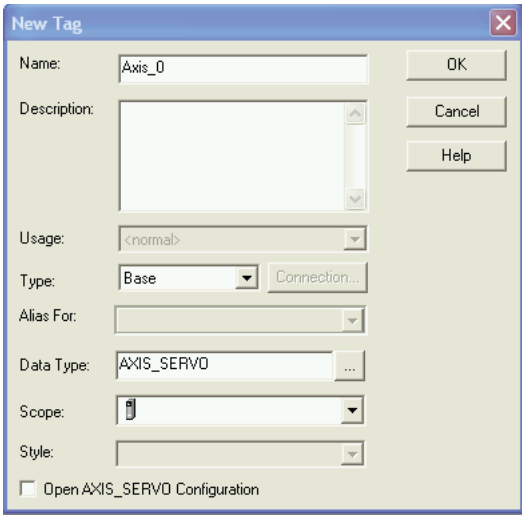


Servo Update Period: 250 us
Associated Axes:
Channel 0: <none>
Channel 1: <none>
Status: Offline

6. Click the Associated Axes tab.

7. Click New Axis.

The New Tag dialog opens.



New Tag

Name: Axis_0

Description:

Usage: <normal>

Type: Base

Alias For:

Data Type: AXIS_SERVO

Scope:

Style:

☐ Open AXIS_SERVO Configuration

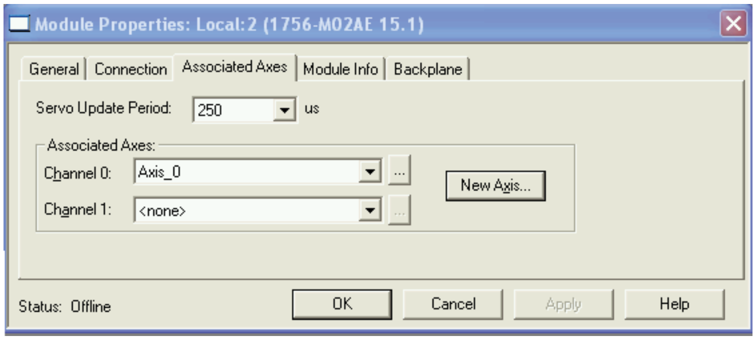
8. Configure the new tag.

a. In the Name box, enter your axis name.

b. From the Data Type pull-down menu, choose AXIS_SERVO.

9. Click OK.

10. From the Channel 0 pull-down menu, choose your axis.



Module Properties: Local:2 (1756-M02AE 15.1)

Servo Update Period: 250 us

Associated Axes:

Channel 0: Axis_0

Channel 1: <none>

Status: Offline

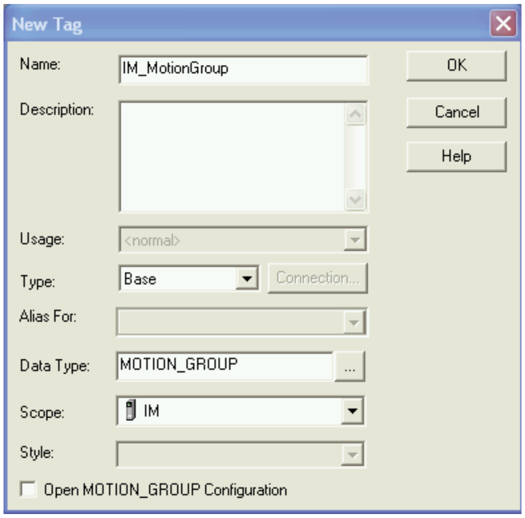
11. Click OK.

Configure the Motion Group

Follow these steps to configure the motion group.

1. In the Explorer dialog, right-click Motion Groups and choose New Motion Group.

   The New Tag dialog opens.



Name: IM_MotionGroup
Description:
Usage: <normal>
Type: Base
Alias For:
Data Type: MOTION_GROUP
Scope: IM
Style:
☐ Open MOTION_GROUP Configuration

2. In the Name box, enter your motion group name.
3. Click OK.

   The new group appears under the Motion Group folder.

4. Right-click the new motion group and choose Properties.

   The Motion Group Properties dialog opens.

Unassigned:
Assigned: Axis_0

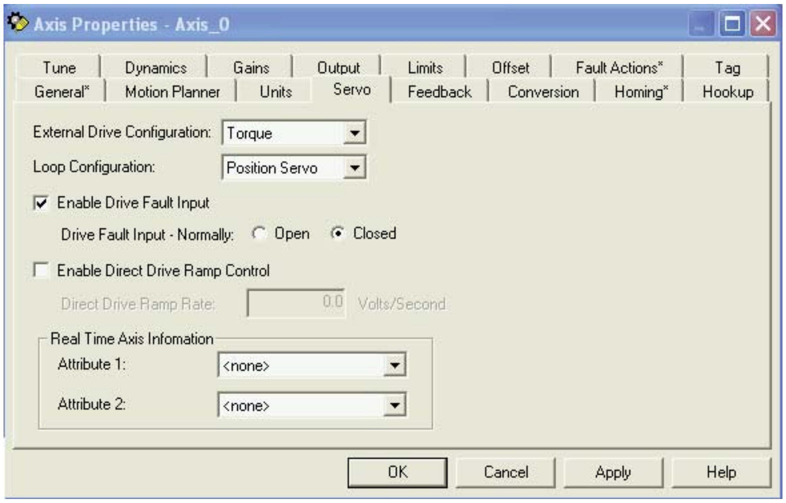
5. Click the Axis Assignment tab and move your axis (created earlier) from Unassigned to Assigned.
6. Click the Attribute tab and edit the default values as appropriate for your application.
7. Click OK.

Configure Axis Properties

Follow these steps to configure axis properties.

1. In the Explorer dialog, right-click an axis and choose Properties.

   The Axis Properties dialog opens.



Axis Properties - Axis_0

External Drive Configuration: Torque

Loop Configuration: Position Servo

☑ Enable Drive Fault Input

Drive Fault Input - Normally: ○ Open ● Closed

☐ Enable Direct Drive Ramp Control

Direct Drive Ramp Rate: 0.0 Volts/Second

Real Time Axis Infomation

Attribute 1: <none>

Attribute 2: <none>

2. Click the Servo tab.
   a. From the External Drive Configuration pull-down menu choose Torque.

      In Torque mode, both position and velocity loops are closed in the Logix controller. In Velocity mode, only the position loop is closed in the Logix controller.

   b. Check the Enable Drive Fault Input checkbox.
   c. Select Drive Fault Input - Normally Closed.

3. Click the Units tab and edit the default values as appropriate for your application.

4. Click the Conversion tab and edit the default values as appropriate for your application.

5. Click OK.

6. Verify your Logix program and save the file.

Download Your Program

After completing the Logix configuration, you must download your program to the Logix processor.

Testing and Tuning Your Axis

This procedure assumes that you have configured your Ultra3000 drive and the analog motion module.

IMPORTANT Before proceeding with testing and tuning your axis, verify that the seven-segment status indicator is actively cycling in a full circle.

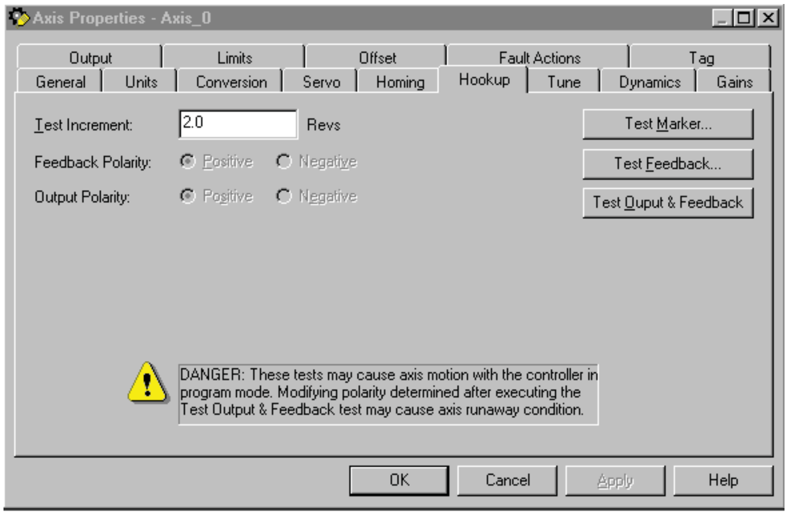
For help using RSLogix 5000 software as it applies to the analog Logix modules, refer to Additional Resources on page 8.

Test Your Axis

Follow these steps to test your axis.

1. Remove the load from your axis.
2. In your Motion Group folder, right-click the axis and choose Properties.

   The Axis Properties dialog opens.



Test Increment: 2.0 Revs

Feedback Polarity: ● Positive ○ Negative

Output Polarity: ● Positive ○ Negative

DANGER: These tests may cause axis motion with the controller in program mode. Modifying polarity determined after executing the Test Output & Feedback test may cause axis runaway condition.

3. Click the Hookup tab.

4. In the Test Increment box, enter 2.0 as the number of revolutions for the test (or another number more appropriate for your application).

Test	Description
Test Marker	Verifies marker detection capability as you rotate the motor shaft.
Test Feedback	Verifies feedback connections are wired correctly as you rotate the motor shaft.
Test Command & Feedback	Verifies motor power and feedback connections are wired correctly as you command the motor to rotate. Also, lets you define polarity.

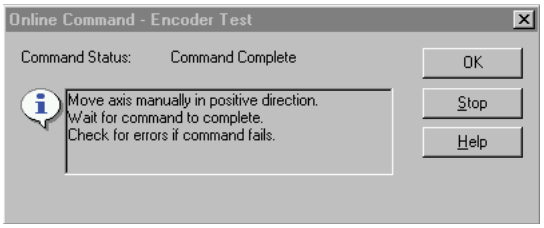
5. Apply Drive Enable (Input 1) signal (CN1-31) for the axis you are testing.

ATTENTION

To avoid personal injury or damage to equipment, apply 24V Drive Enable signal (CN1-31) only to the axis you are testing.

6. Click the desired test (Marker/Feedback/Command & Feedback) to verify connections.

The Online Command dialog opens. Follow the test instructions. When the test completes, the Command Status changes from Executing to Command Complete

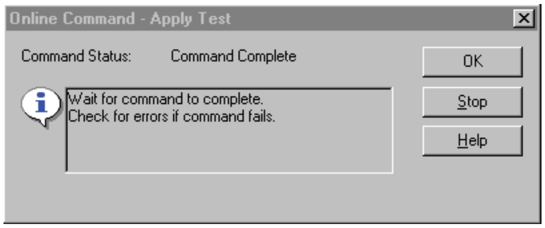


Command Status: Command Complete

Move axis manually in positive direction.
Wait for command to complete.
Check for errors if command fails.

7. Click OK.

The Online Command - Apply Test dialog opens (Feedback and Command & Feedback tests only). When the test completes, the Command Status changes from Executing to Command Complete.

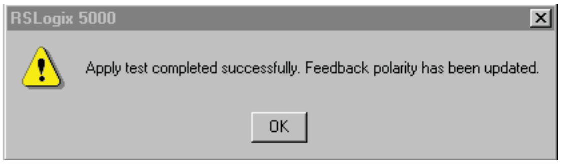
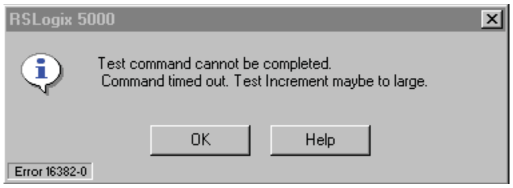


Command Status: Command Complete

Wait for command to complete.
Check for errors if command fails.

8. Click OK.

9. Determine if your test completed successfully.

If	Then
Your test completes successfully, this dialog appears.  Apply test completed successfully. Feedback polarity has been updated. 	1. Click OK. 2. Remove Drive Enable signal (CN1-31). 3. Go to Tune Your Axis.
Your test failed, this dialog appears.  Test command cannot be completed. Command timed out. Test Increment maybe to large. Error 16382-0	1. Click OK. 2. Verify that the main three-phase bus power is up. 3. Verify that the Drive Enable signal (CN1-31) is applied to the axis you are testing. 4. Verify conversion constant entered in the Conversion tab. 5. Return to step 6 and run the test again.

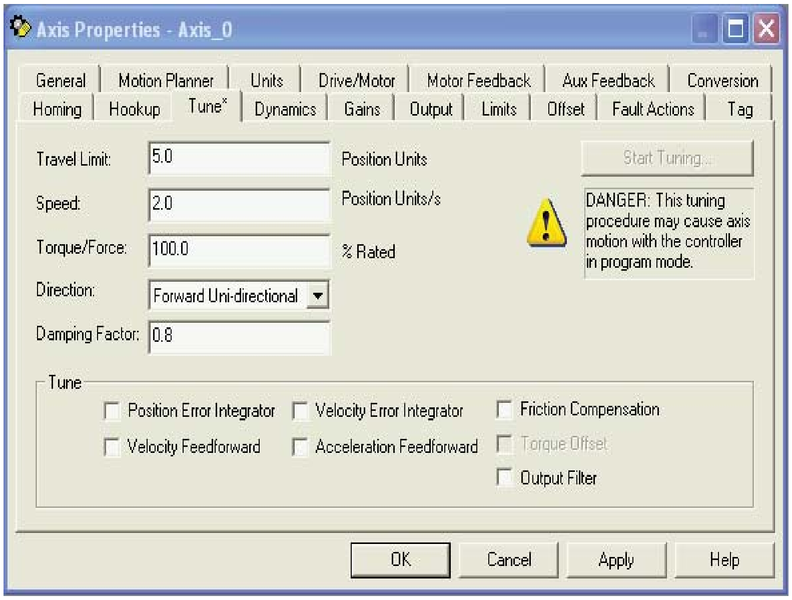
Tune Your Axis

Follow these steps to tune your axis.

1. Verify that the load is still removed from the axis being tuned.

ATTENTION To reduce the possibility of unpredictable motor response, tune your motor with the load removed first, then reattach the load and perform the tuning procedure again to provide an accurate operational response.

2. Click the Tune tab.



Travel Limit: 5.0 Position Units

Speed: 2.0 Position Units/s

Torque/Force: 100.0 % Rated

Direction: Forward Uni-directional

Damping Factor: 0.8

DANGER: This tuning procedure may cause axis motion with the controller in program mode.

Tune

☐ Position Error Integrator ☐ Velocity Error Integrator ☐ Friction Compensation

☐ Velocity Feedforward ☐ Acceleration Feedforward ☐ Torque Offset

☐ Output Filter

3. In the Travel Limit and Speed boxes, enter values.

In this example, Travel Limit = 5 and Speed = 2. Actual value of programmed units depends on your application.

4. From the Direction pull-down menu, choose your direction (Forward Uni-directional is default).

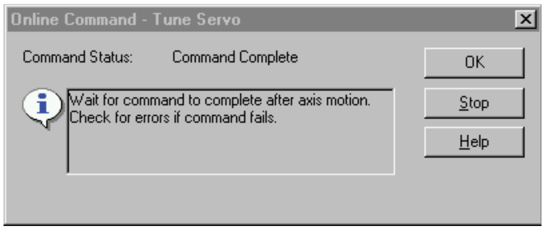
5. Check the Tune boxes appropriate for your application.

6. Apply Drive Enable (Input 1) signal (CN1-31) for the axis you are tuning.

ATTENTION To avoid personal injury or damage to equipment, apply 24V Drive Enable signal (CN1-31) only to the axis you are tuning.

7. Click Start Tuning to auto-tune your axis.

The Online Command - Tune Servo dialog opens. When the test completes, the Command Status changes from Executing to Command Complete.

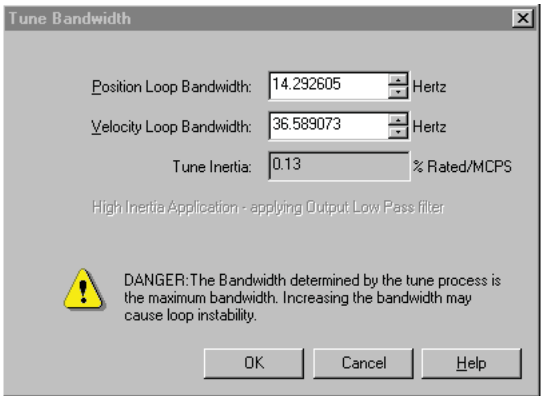


Command Status: Command Complete

Wait for command to complete after axis motion.
Check for errors if command fails.

8. Click OK.

The Tune Bandwidth dialog opens.



Position Loop Bandwidth: 14.292605 Hertz
Velocity Loop Bandwidth: 36.589073 Hertz
Tune Inertia: 0.13 % Rated/MCPS

High Inertia Application - applying Output Low Pass filter

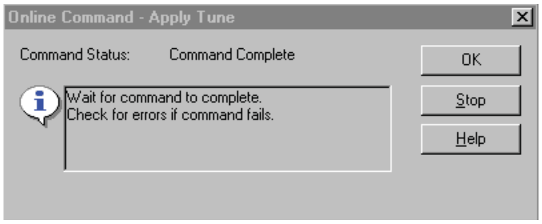
DANGER:The Bandwidth determined by the tune process is the maximum bandwidth. Increasing the bandwidth may cause loop instability.

Actual bandwidth values (Hz) depend on your application and may require adjustment once motor and load are connected.

Record your bandwidth data for future reference.

9. Click OK.

The Online Command - Apply Tune dialog opens. When the test completes, the Command Status changes from Executing to Command Complete.

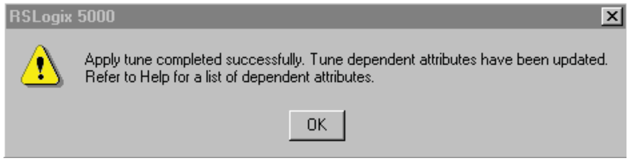
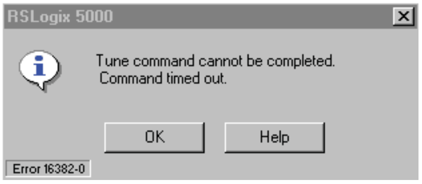


Command Status: Command Complete

Wait for command to complete.
Check for errors if command fails.

10. Click OK.

11. Determine if your test completed successfully.

If	Then
Your test completes successfully, this dialog appears.  RSLogix 5000 Apply tune completed successfully. Tune dependent attributes have been updated. Refer to Help for a list of dependent attributes. 	1. Click OK. 2. Remove Drive Enable (Input 1) signal (CN1-31) applied earlier. 3. You are finished tuning your Ultra3000 drive.
Your test failed, this dialog appears.  RSLogix 5000 Tune command cannot be completed. Command timed out. Error 16382-0	1. Click OK. 2. Make an adjustment to motor velocity. 3. Refer to the appropriate Logix motion module setup and configuration manual for more information. 4. Return to step 7 and run the test again.

Configuring Your Ultra3000 Drive with SERCOS

The procedures in this section are listed in this table and apply to Ultra3000-SE drives with SERCOS interface.

Ultra3000 Drive Configuration Procedures

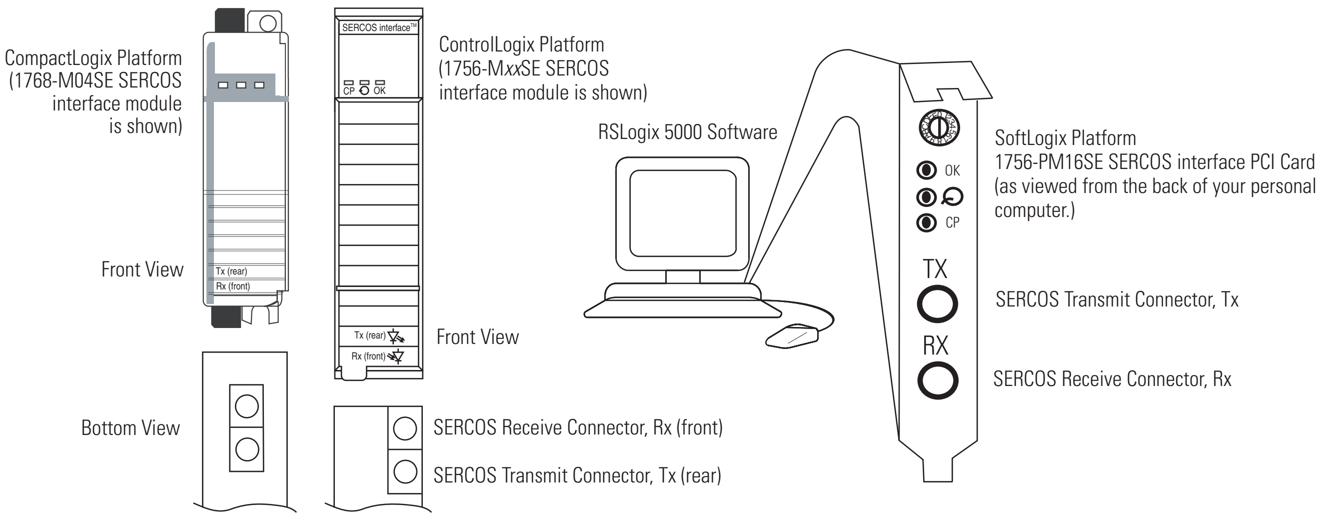
Procedure	Page
Configure Your Ultra3000-SE Drive	70
Configuring Your Logix SERCOS interface Module	72
Download Your Program	80
Apply Power to Your Ultra3000 Drive with SERCOS	81
Testing and Tuning Your Axis	83

These procedures assume you have connected the fiber-optic cables between your Ultra3000-SE drive and the SERCOS interface module.

Front Panel Connections

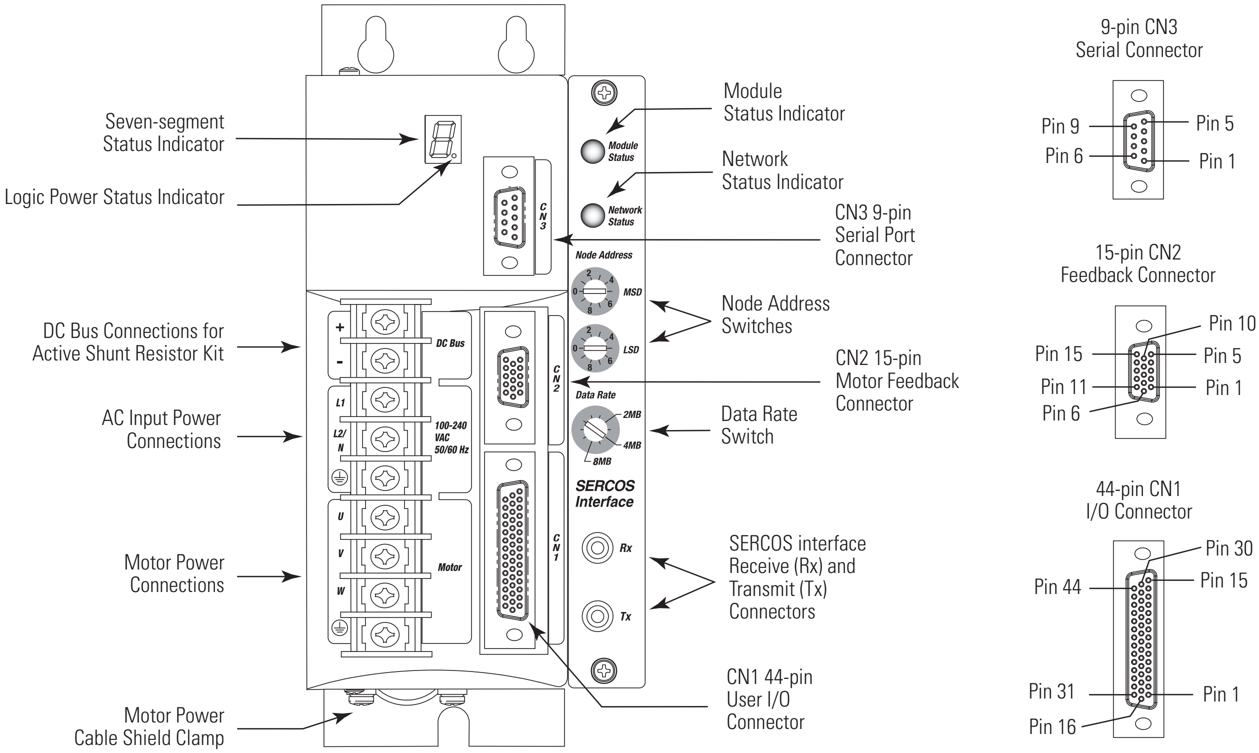
Use this figure to locate the SERCOS ring status indicators and fiber-optic cable connections on your SERCOS interface module or PCI card.

CompactLogix, ControlLogix, and SoftLogix SERCOS Connector Locations



Use this figure to locate the front panel connections on the Ultra3000-SE 230V drives (500W, 1 kW, and 2 kW).

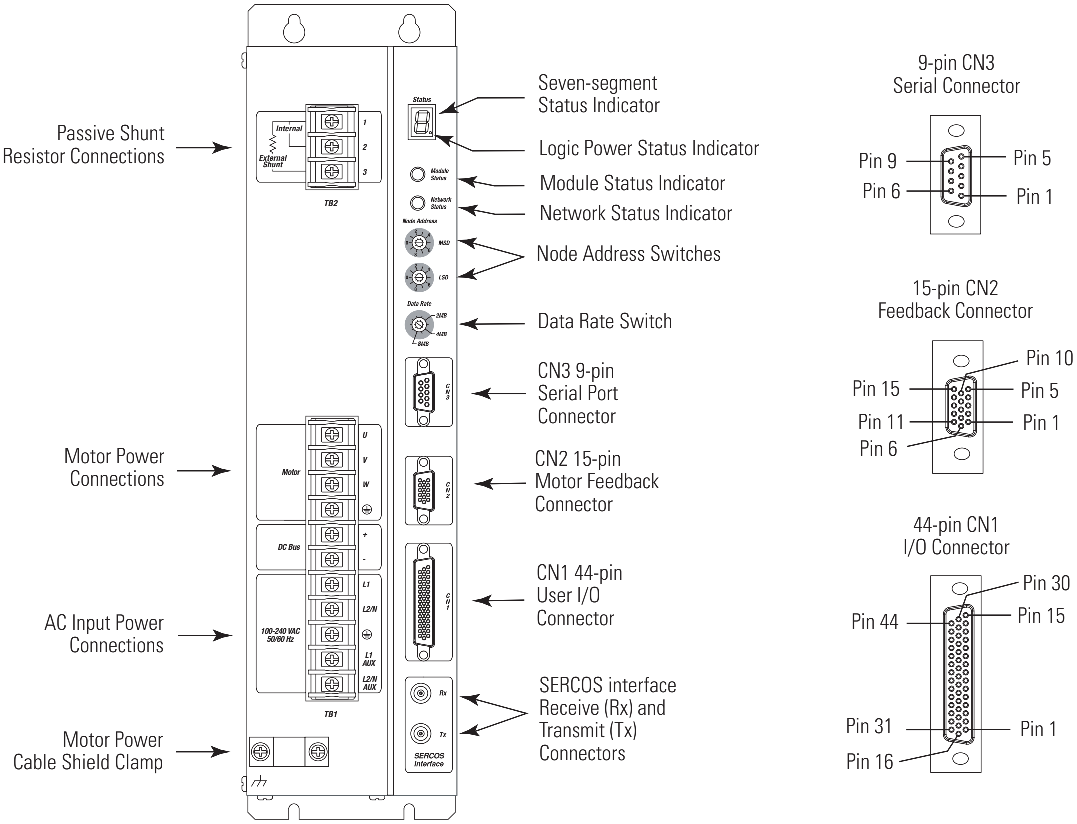
Front Panel Connections for 2098-DSD-005-SE, 2098-DSD-010-SE, and 2098-DSD-020-SE Drives



For CN1, CN2, and CN3 connector pin-out information, refer to the Ultra3000 Digital Servo Drives Installation Manual, publication 2098-IN003.

Use this figure to locate the front panel connections on the Ultra3000-SE 230V drive (3 kW).

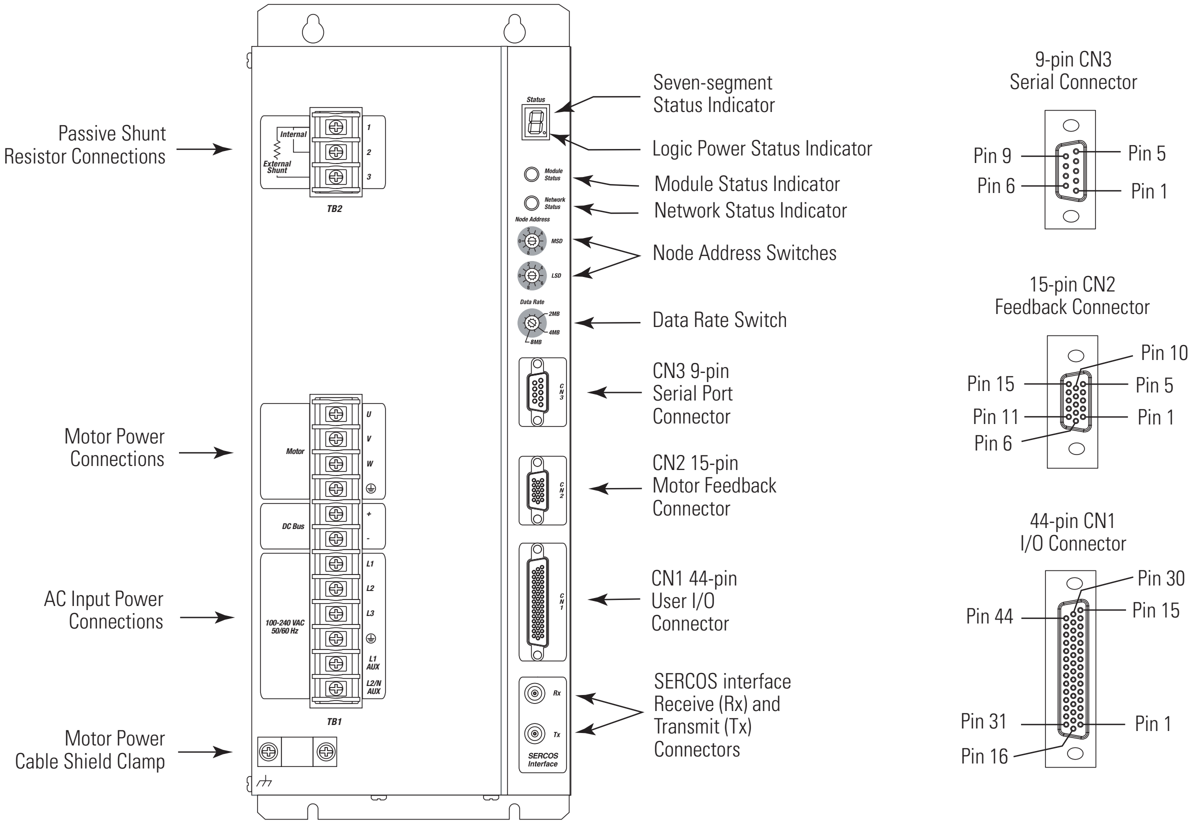
Front Panel Connections for 2098-DSD-030-SE Drive



For CN1, CN2, and CN3 connector pin-out information, refer to the Ultra3000 Digital Servo Drives Installation Manual, publication 2098-IN003.

Use this figure to locate the front panel connections on the Ultra3000-SE 230V drives (7.5 and 15 kW).

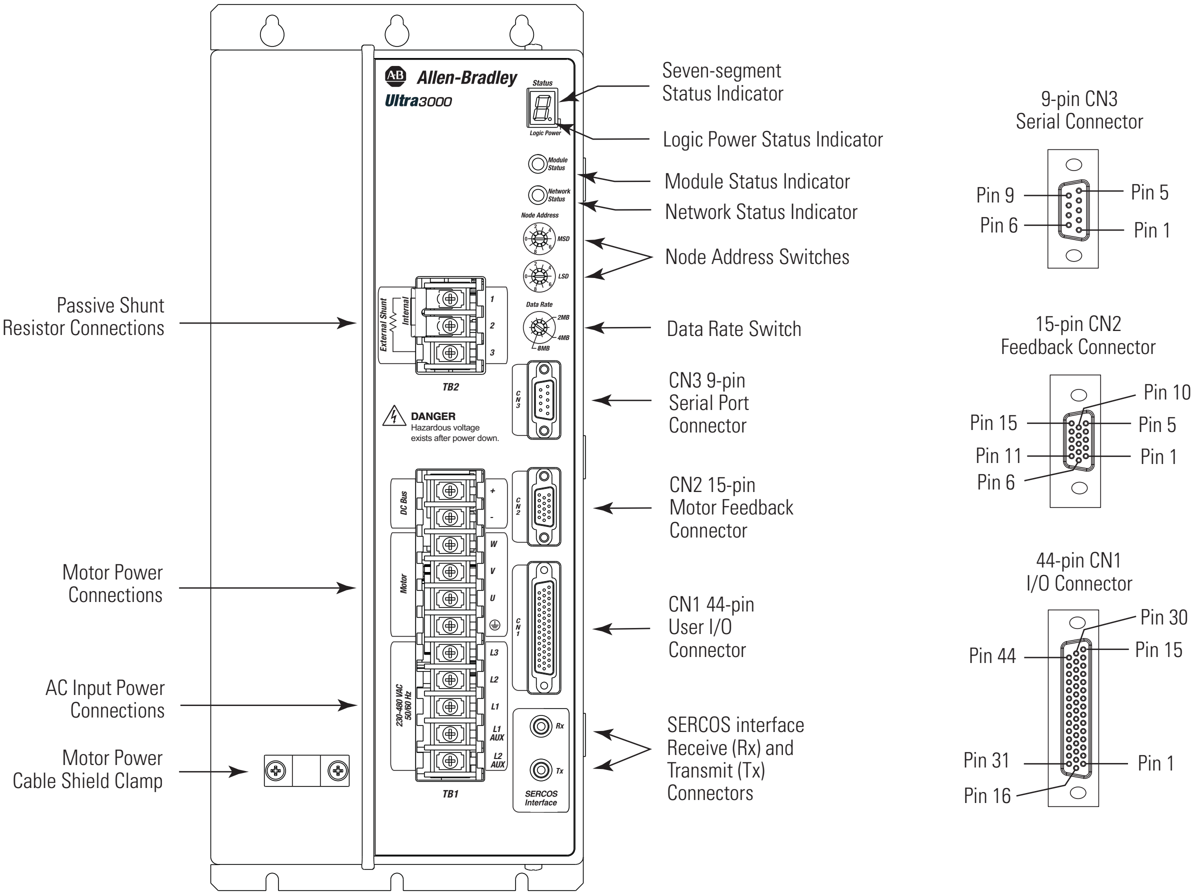
Front Panel Connections for 2098-DSD-075-SE and 2098-DSD-150-SE Drives



For CN1, CN2, and CN3 connector pin-out information, refer to the Ultra3000 Digital Servo Drives Installation Manual, publication 2098-IN003.

Use this figure to locate the front panel connections on the Ultra3000-SE 460V drives (3 kW, 5 kW, 10 kW, 15 kW, and 22 kW).

Front Panel Connections for 2098-DSD-HV*xxx*-SE Drives



For CN1, CN2, and CN3 connector pin-out information, refer to the Ultra3000 Digital Servo Drives Installation Manual, publication 2098-IN003.

Configure Your Ultra3000-SE Drive

Follow these steps to configure your Ultra3000-SE drive.

1. Verify that there is no power applied to the drive and that the SERCOS fiber-optic cables are correctly plugged into the Tx and Rx connectors.

   To verify your fiber-optic cable connections, refer to Fiber-optic Ring Connections on page 71.

2. Set the node address for each drive in your system.

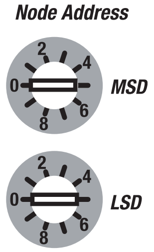
   Valid node addresses are 01…99. The MSD rotary switch sets the most significant digit and the LSD rotary switch sets the least significant digit.

   Refer to the figures on pages 66…69 for switch locations. Refer to this table for examples.

Node Address	MSD Switch	LSD Switch
10	1	0
11	1	1
12	1	2

Use the MSD and LSD rotary switches on the SERCOS panel of the drive to set node addresses.

For an example, refer to Ultra3000-SE Drive Node Addressing on page 71.



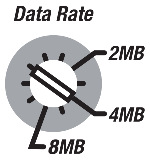
IMPORTANT Whenever the node address setting is changed, you must cycle power for the change to register in the drive.

3. Set the data rate.

   Valid data rates are 4 and 8 MB. The 2 MB setting does not apply.

   Refer to the figures on pages 66…69 for switch locations.

   Use the Data Rate rotary switch on the SERCOS panel of the drive to set the data rate.

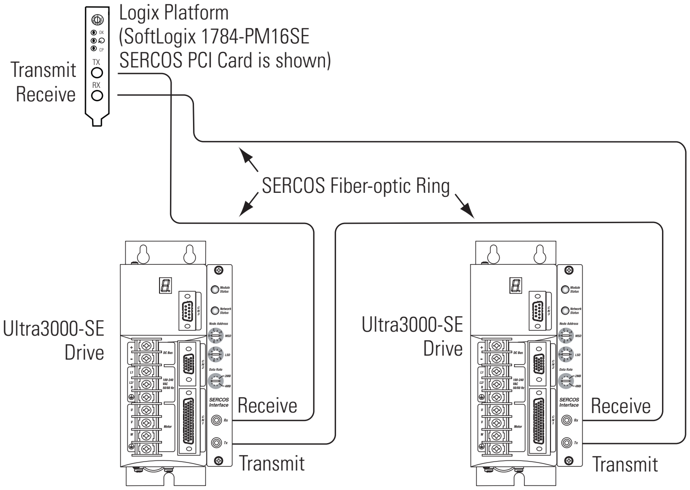


4. Verify Input 1 (CN1-31) is configured as Drive Enable and tied to 12…24V dc.

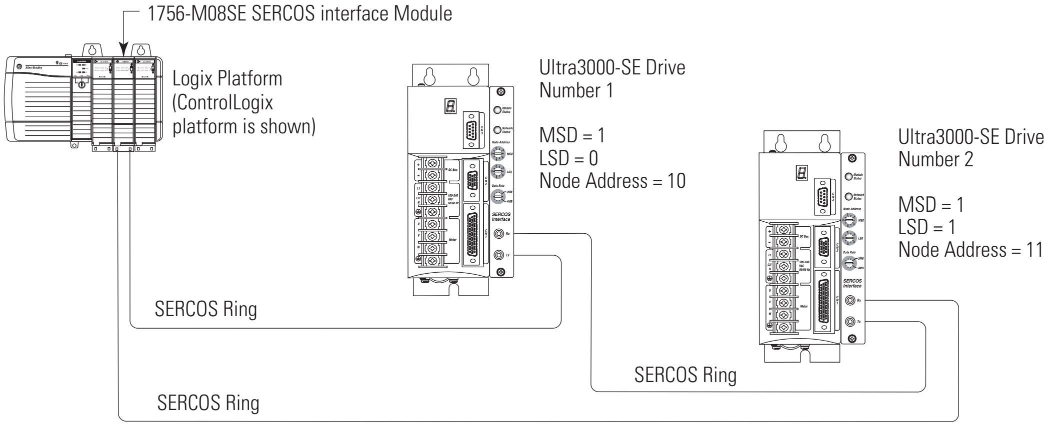
5. If using Overtravel inputs, verify that 12…24V dc is tied to CN1-37 and CN1-38.

IMPORTANT	Without CN1-37 and CN1-38 inputs applied, the drive/system will fault.

Fiber-optic Ring Connections



Ultra3000-SE Drive Node Addressing



Configuring Your Logix SERCOS interface Module

This procedure assumes that you have configured the Ultra3000-SE communication rate.

IMPORTANT In order for the Ultra3000 drive to communicate with the SERCOS interface module (indicated by the three status indicators on the module going solid green), your RSLogix 5000 software must be version 11.0 or later.

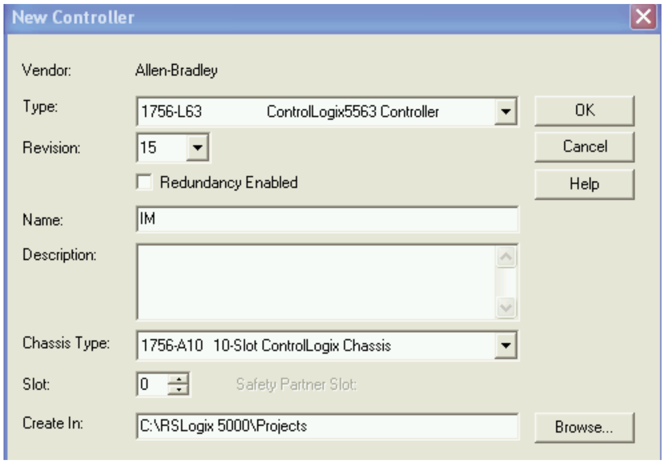
For help using RSLogix 5000 software as it applies to configuring the ControlLogix, CompactLogix, or SoftLogix SERCOS modules, refer to Additional Resources on page 8.

Configure Your Logix Controller

Follow these steps to configure your Logix controller.

1. Apply power to your Logix chassis containing the SERCOS interface module and open your RSLogix 5000 software.
2. From the File menu, choose New.

   The New Controller dialog opens.



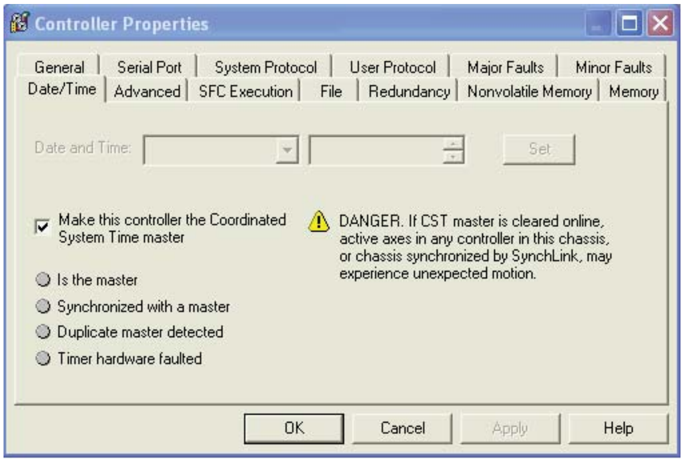
Vendor: Allen-Bradley
Type: 1756-L63 ControlLogix5563 Controller
Revision: 15
☐ Redundancy Enabled
Name: IM
Description:
Chassis Type: 1756-A10 10-Slot ControlLogix Chassis
Slot: 0 Safety Partner Slot:
Create In: C:\RSLogix 5000\Projects

3. Configure the new controller.
   a. From the Type pull-down menu, choose your controller.
   b. From the Revision pull-down menu, choose your RSLogix 5000 software version.
   c. In the Name box, name your file.
   d. From the Chassis Type pull-down menu, choose your Logix chassis.
   e. Enter the Logix processor slot.

4. Click OK.

5. From the Edit menu, choose Controller Properties.

   The Controller Properties dialog opens.



☑ Make this controller the Coordinated System Time master

○ Is the master
○ Synchronized with a master
○ Duplicate master detected
○ Timer hardware faulted

⚠ DANGER. If CST master is cleared online, active axes in any controller in this chassis, or chassis synchronized by SynchLink, may experience unexpected motion.

6. Click the Date/Time tab.

7. Check the Make this controller the Coordinated System Time master checkbox.

IMPORTANT	Only one Logix processor can be assigned as the Coordinated System Time master.

8. Click OK.

Configure Your Logix Module

Follow these steps to configure your Logix module.

1. In the Explorer dialog, right-click I/O Configuration and choose New Module.

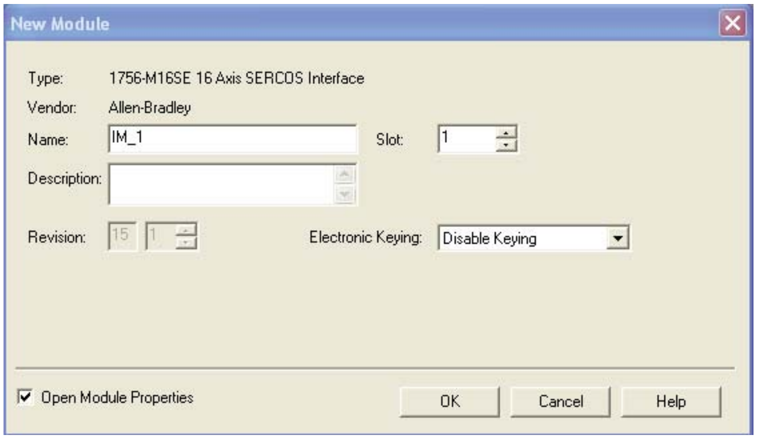
   The Select Module dialog opens.

2. Expand the Motion category and select 1756-M*xx*SE 1756-L60M03SE, 1768-M04SE, or 1784-PM16SE as appropriate for your actual hardware configuration.

3. Click OK.

   Your new module appears under the I/O Configuration folder in the Explorer dialog.

The New Module dialog opens.



Type: 1756-M16SE 16 Axis SERCOS Interface
Vendor: Allen-Bradley
Name: IM_1
Slot: 1
Description:
Revision: 15 1
Electronic Keying: Disable Keying
☑ Open Module Properties

4. Configure the new module.

a. In the Name box, name your file.

b. In the Slot box, enter the slot where your module resides.

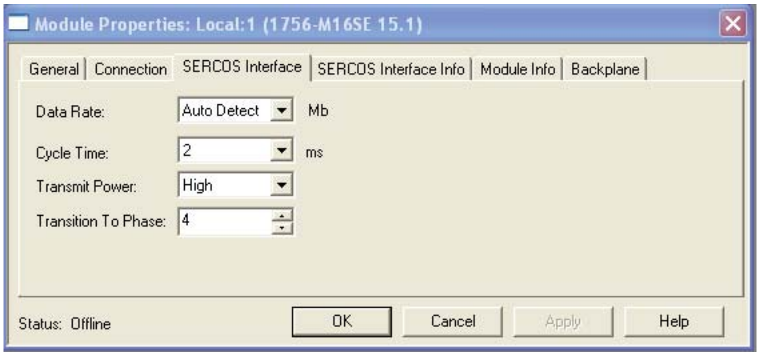
c. From the Electronic Keying pull-down menu, choose an electronic keying option (choose Disable Keying if unsure).

d. Check the Open Module Properties checkbox.

5. Click OK.

Your new module appears under the I/O Configuration folder in the Explorer dialog and the Module Properties dialog opens.

6. Click the SERCOS Interface tab and reference the table below.



Data Rate: Auto Detect Mb
Cycle Time: 2 ms
Transmit Power: High
Transition To Phase: 4
Status: Offline

Logix SERCOS Module	Number of Axes	Data Rate
1756-M03SE or 1756-L60M03SE	Up to 3	4 or 8 Mbps
1756-M08SE	Up to 8	
1756-M16SE or 1784-PM16SE	Up to 16	
1768-M04SE	Up to 4	

7. Verify that the Data Rate setting matches the Data Rate (communication rate) switch setting on the Ultra3000-SE drive.

8. Set the Cycle Time according to this table.

Data Rate	Number of Axes	Cycle Time
4 Mbps	Up to 4	1 ms
	Up to 8	2 ms
	No support for axes 9…16	
8 Mbps	Up to 8	1 ms
	Up to 16	2 ms

   The number of axes/module is limited to the number of axes as shown in step 6.

9. Verify that Transmit Power is set to High.

10. Set Transition to Phase.

    Transition to Phase default setting is 4 (phase 4). The Transition to Phase setting will stop the ring in the phase specified.

11. Click OK.

12. Repeat steps 1…11 for each Logix module.

Configure Your Ultra3000-SE Drive

Follow these steps to configure your Ultra3000-SE drive.

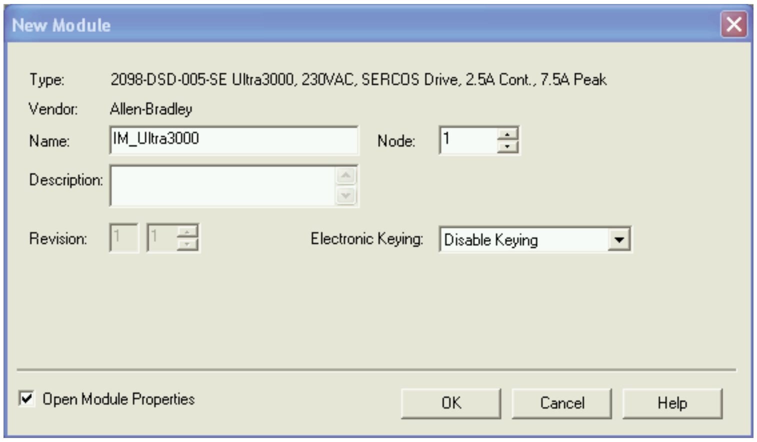
1. Right-click your new module and choose New Module.

   The Select Module dialog opens.

2. Expand the Drives category and select 2098-DSD-*xxx*-SE or 2098-DSD-HV*xxx*-SE drive as appropriate for your actual hardware configuration.

3. Click OK.

The New Module dialog opens.



Type: 2098-DSD-005-SE Ultra3000, 230VAC, SERCOS Drive, 2.5A Cont., 7.5A Peak
Vendor: Allen-Bradley
Name: IM_Ultra3000
Node: 1
Description:
Revision: 1 1
Electronic Keying: Disable Keying
☑ Open Module Properties

4. Configure the new module.

a. In the Name box, enter your module name.

b. In the Node box, enter the node address.

Set the node address in the software to match the node address setting on the drive.

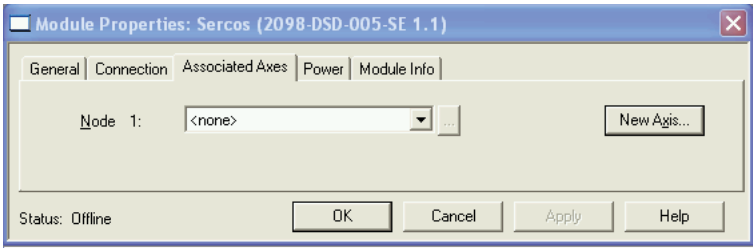
Refer to Configure Your Ultra3000-SE Drive, step 2, on page 70.

c. From the Electronic Keying pull-down menu, choose an electronic keying option (choose Disable Keying if unsure).

d. Check the Open Module Properties checkbox.

5. Click OK.

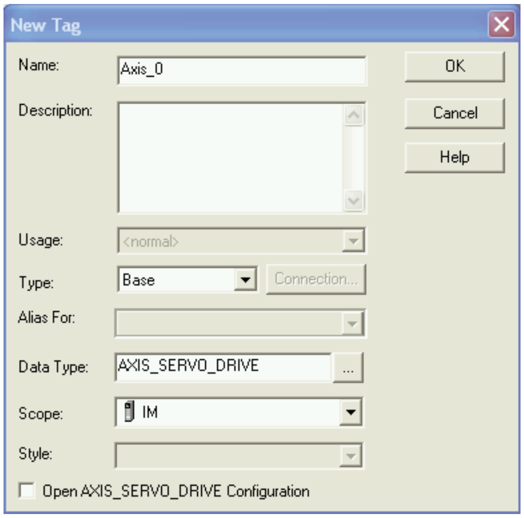
6. Click the Associated Axes tab.



Node 1: <none>
Status: Offline

7. Click New Axis.

The New Tag dialog opens.


New Tag

Name: Axis_0

Description:

Usage: <normal>

Type: Base

Alias For:

Data Type: AXIS_SERVO_DRIVE

Scope: IM

Style:

☐ Open AXIS_SERVO_DRIVE Configuration

8. Configure the new tag.

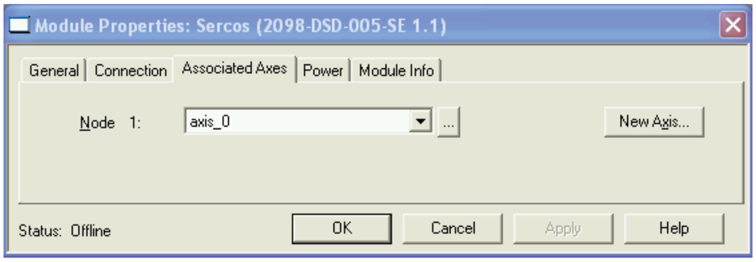
a. In the Name box, enter your module name.

b. In the Data Type pull-down menu, choose AXIS_SERVO_DRIVE.

9. Click OK.

The axis appears under the Ungrouped Axes folder in the Explorer dialog.

10. From the Node 1 pull-down menu, choose the node address for your axis.


Module Properties: Sercos (2098-DSD-005-SE 1.1)

Node 1: axis_0

Status: Offline

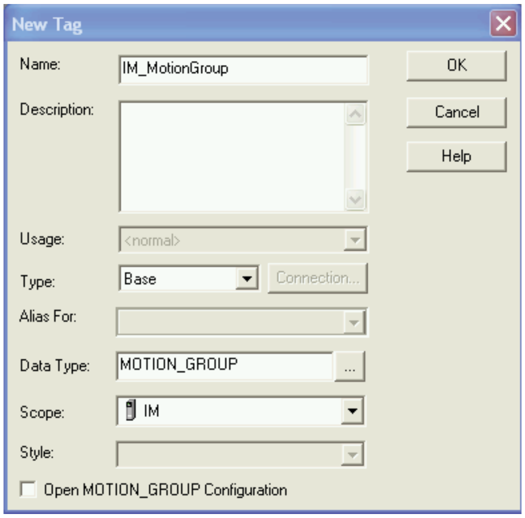
11. Click OK.

Configure the Motion Group

Follow these steps to configure the motion group.

1. In the Explorer dialog, right-click Motion Groups and choose New Motion Group.

   The New Tag dialog opens.



New Tag

Name: IM_MotionGroup

Description:

Usage: <normal>

Type: Base

Alias For:

Data Type: MOTION_GROUP

Scope: IM

Style:

☐ Open MOTION_GROUP Configuration

2. In the Name box, enter the new motion group name.

3. Click OK.

   The new group appears under Motion Group folder.

4. Right-click the new motion group and choose Properties.

   The Motion Group Properties dialog opens.

Motion Group Properties - IM_MotionGroup

Unassigned:

Assigned: Axis_0

5. Click the Axis Assignment tab and move your axis (created earlier) from Unassigned to Assigned.

6. Click the Attribute tab and edit the default values as appropriate for your application.

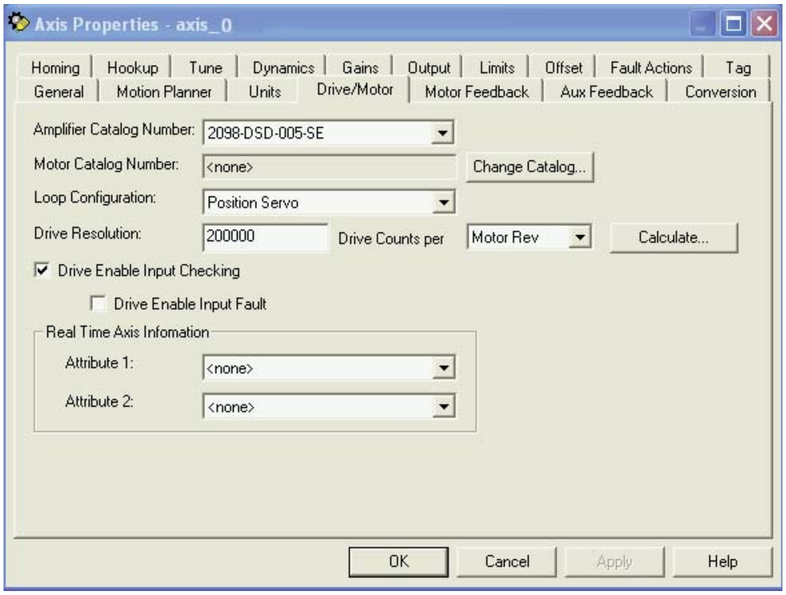
7. Click OK.

Configure Axis Properties

Follow these steps to configure axis properties.

1. In the Explorer dialog, right-click an axis and choose Properties.

The Axis Properties dialog opens.



Amplifier Catalog Number: 2098-DSD-005-SE

Motor Catalog Number: <none>

Loop Configuration: Position Servo

Drive Resolution: 200000 Drive Counts per Motor Rev

☑ Drive Enable Input Checking

☐ Drive Enable Input Fault

Real Time Axis Infomation

Attribute 1: <none>

Attribute 2: <none>

2. Click the Drive/Motor tab.

a. From the Amplifier Catalog Number pull-down menu, choose the Ultra3000 amplifier (2098-DSD-*xxx*-SE or 2098-DSD-HV*xxx*-SE).

b. Click Change Catalog to set the motor catalog number.

To verify the amplifier and motor catalog numbers, refer to the amplifier and motor name plates.

c. Check the Drive Enable Input Checking checkbox.

When checked (default), means a hard drive-enable input signal is required. Uncheck to remove that requirement.

3. Click the Motor Feedback tab and verify that the Feedback Type shown is appropriate for your actual hardware configuration.

4. Click the Units tab and edit default values as appropriate for your application.

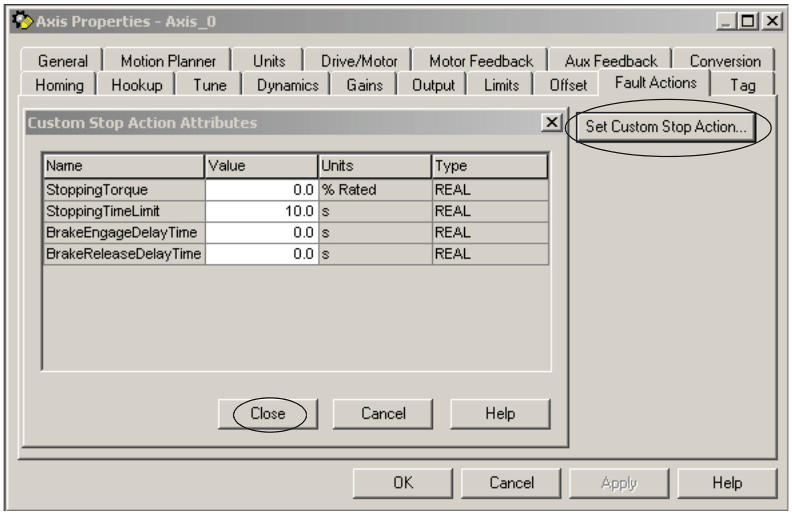
5. Click the Conversion tab and edit default values as appropriate for your application.

6. Click the Fault Actions tab.

TIP For more information on setting fault limits, refer to Appendix C, Minimizing the Effects of Feedback Signal Loss on page 141.

7. Click the Set Custom Stop Action.

The Custom Stop Action Attributes dialog opens.



Name	Value	Units	Type
StoppingTorque	0.0	% Rated	REAL
StoppingTimeLimit	10.0	s	REAL
BrakeEngageDelayTime	0.0	s	REAL
BrakeReleaseDelayTime	0.0	s	REAL

8. Set the appropriate values.

a. Set the Brake Engage Delay Time.

Use this setting to hold motor torque on load until the delay time has expired (brake is engaged).

b. Set the Brake Release Delay Time.

Use this setting to make sure no motor movement occurs until the delay time has expired (brake is released).

c. Click Close.

9. Click OK.

10. Verify your Logix program and save the file.

Download Your Program

After completing the Logix configuration you must download your program to the Logix processor.

Apply Power to Your Ultra3000 Drive with SERCOS

This procedure assumes that you have configured your Ultra3000-SE drive and your SERCOS interface module.

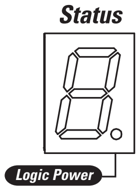
ATTENTION ⚠ High voltage exists in ac line filters. The filter must be grounded properly before applying power. Filter capacitors retain high voltages after power removal. Before handling the equipment, voltages should be measured to determine safe levels. Failure to observe this precaution could result in personal injury.

Follow these steps to apply power to your Ultra3000-SE drive.

1. Disconnect any load to the motor, making sure that the motor is free of all linkages when initially applying power to the system.

   ATTENTION ⚠ To avoid damage to the drive due to improper sequencing of input power and the Drive Enable signal, do not issue the Drive Enable command from RSLogix 5000 software without first applying input power.

2. Apply input power to the Ultra3000-SE drive and observe the front panel Logic Power status indicator.



If the Logic Power Status Indicator is	Then
ON	Go to step 3.
Not ON	1. Check your input power connections. 2. Repeat step 2.

3. Observe the front panel seven-segment status indicator on your Ultra3000-SE drive.

The status indicator cycles through SERCOS phases until final configuration (phase 4) is reached.

Seven-segment Status Indicator	**Status**	**Do This**
Actively cycling (phase 0)	The drive is looking for a closed SERCOS ring. Wait for phase 1 or take corrective action until you reach phase 1.	Check fiber-optic connections.
Displaying a fixed 1 (phase 1)	The drive is looking for active nodes. Wait for phase 2 or take corrective action until you reach phase 2.	Check node addressing.
Displaying a fixed 2 (phase 2)	The drive is configuring nodes for communication. Wait for phase 3 or take corrective action until you reach phase 3.	Check program motor and drive configuration against installed hardware.
Displaying a fixed 3 (phase 3)	The drive is configuring device specific parameters. Wait for phase 4 or take corrective action until you reach phase 4.	Check motor catalog number against selection. (1)
Displaying a fixed 4 (phase 4)	The drive is configured and active.	Go to step 4.
Flashing E followed by two numbers	Drive is faulted.	Go to Error Codes on page 98.

(1) You can get diagnostic information from the module by highlighting the module name in RSLogix 5000 software. A Pseudo Key Failure often indicates that the motor selection does not match the motor installed.

4. Observe the module status indicator.

Module Status Indicator	**Status**	**Do This**
Steady green	The drive is enabled.	Go to step 5.
Flashing green	The drive is disabled.	Go to step 5.
Not steady green/ not flashing green	The drive is faulted.	Go to SERCOS Module Status Indicator troubleshooting on page 104.

5. Observe the network status indicator.

Network Status Indicator	**Status**	**Do This**
Flashes green	Establishing communication with network.	Wait for steady green.
Steady green	Communication is ready.	Go to step 6.
Not steady green/ not flashing green	The drive is faulted.	Go to SERCOS Network Status Indicator troubleshooting on page 104.

6. Observe the three SERCOS indicators on the SERCOS module.

Three SERCOS Indicators	**Status**	**Do This**
Flashing green and red	Establishing communication.	Wait for steady green on all three indicators.
Steady green	Communication ready.	Go to Testing and Tuning Your Axis on page 83.
Not flashing green and red/ not steady green	SERCOS module is faulted.	Go to the appropriate Logix manual for specific instructions and troubleshooting.

Testing and Tuning Your Axis

This procedure assumes that you have configured your Ultra3000-SE drive, your SERCOS interface module, and applied power to the system.

IMPORTANT Before proceeding with testing and tuning your axis, verify that the Ultra3000-SE status indicators are as described in this table.

Status Indicator	Indication	Status
Seven-segment	Displaying a fixed 4 (phase 4)	The drive is ready.
Module	Flashing green	The drive is disabled.
Network	Steady green	SERCOS communication is ready.

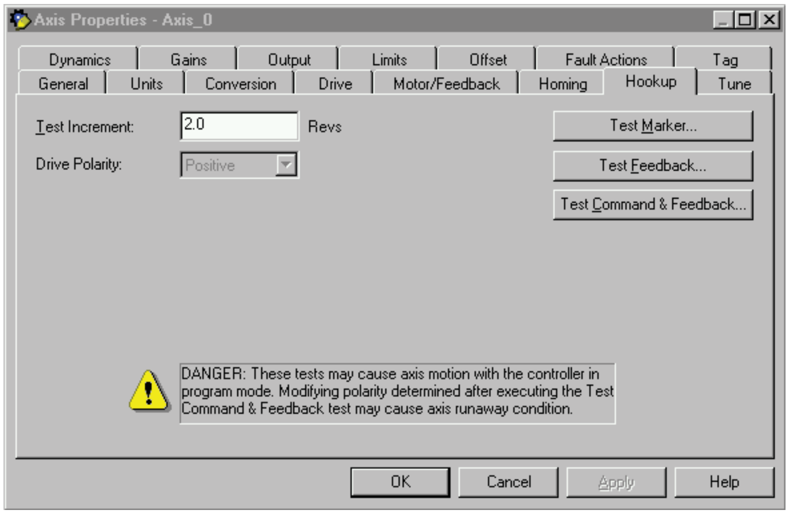
For help using RSLogix 5000 software as it applies to testing and tuning your axes with the ControlLogix, CompactLogix, or SoftLogix SERCOS modules, refer to Additional Resources on page 8.

Test Your Axis

Follow these steps to test your axis.

1. Verify that the load was removed from your motor.
2. Right-click the axis in your Motion Group folder and choose Properties.

   The Axis Properties dialog opens.



Test Increment: 2.0 Revs

Drive Polarity: Positive

DANGER: These tests may cause axis motion with the controller in program mode. Modifying polarity determined after executing the Test Command & Feedback test may cause axis runaway condition.

3. Click the Hookup tab.

4. In the Test Increment box, enter 2.0 as the number of revolutions for the test (or another number more appropriate for your application).

Test	Description
Test Marker	Verifies marker detection capability as you rotate the motor shaft.
Test Feedback	Verifies feedback connections are wired correctly as you rotate the motor shaft.
Test Command & Feedback	Verifies motor power and feedback connections are wired correctly as you command the motor to rotate. Also, lets you define polarity.

5. Apply Drive Enable (Input 1) signal (CN1-31) for the axis you are testing.

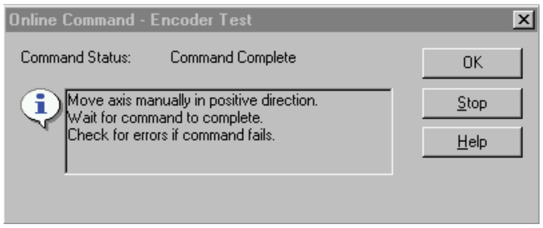
   This step is required only if you checked the box for Drive Enable Input Checking, in the Drive/Motor tab, Axis Properties dialog.

ATTENTION ⚠

To avoid personal injury or damage to equipment, apply 24V Drive Enable signal (CN1-31) only to the axis you are testing.

6. Click the desired test (Marker/Feedback/Command & Feedback) to verify connections.

   The Online Command dialog opens. Follow the test instructions. When the test completes, the Command Status changes from Executing to Command Complete.

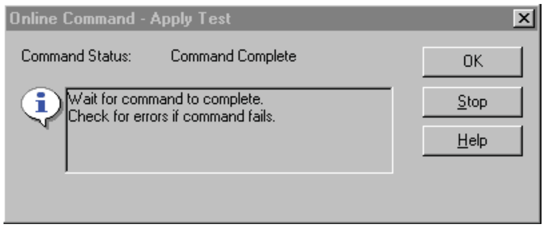

Online Command - Encoder Test

Command Status: Command Complete

Move axis manually in positive direction.
Wait for command to complete.
Check for errors if command fails.

7. Click OK.

   The Online Command - Apply Test dialog opens (Feedback and Command & Feedback tests only). When the test completes, the Command Status changes from Executing to Command Complete.

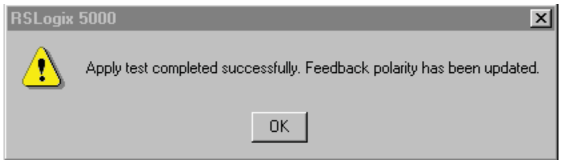
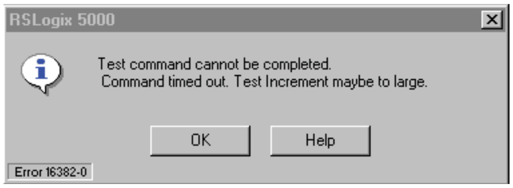

Online Command - Apply Test

Command Status: Command Complete

Wait for command to complete.
Check for errors if command fails.

8. Click OK.

9. Determine if your test completed successfully.

If	Then
Your test completes successfully, this dialog appears.  Apply test completed successfully. Feedback polarity has been updated. 	1. Click OK. 2. Remove Drive Enable signal (CN1-31). 3. Go to Tune Your Axis on page 86.
Your test failed, this dialog appears.  Test command cannot be completed. Command timed out. Test Increment maybe to large. Error 16382-0 	1. Click OK. 2. Verify that the main three-phase bus power is up. 3. Verify that the Drive Enable signal (CN1-31) is applied to the axis you are testing. 4. Verify conversion constant entered in the Conversion tab. 5. Return to main step 6 and run the test again.

Tune Your Axis

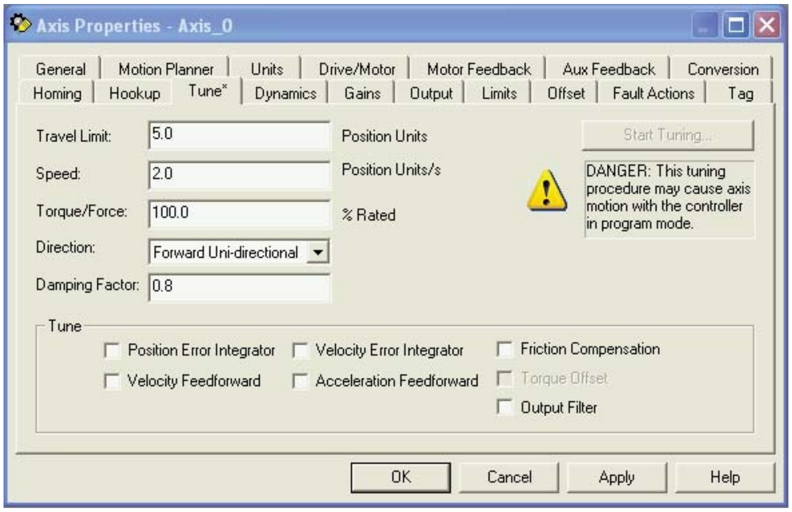
Follow these steps to tune your axis.

1. Verify that the load is still removed from the axis being tuned.

ATTENTION

To reduce the possibility of unpredictable motor response, tune your motor with the load removed first, then reattach the load and perform the tuning procedure again to provide an accurate operational response.

2. Click the Tune tab.



Travel Limit: 5.0 Position Units

Speed: 2.0 Position Units/s

Torque/Force: 100.0 % Rated

Direction: Forward Uni-directional

Damping Factor: 0.8

DANGER: This tuning procedure may cause axis motion with the controller in program mode.

Tune

☐ Position Error Integrator ☐ Velocity Error Integrator ☐ Friction Compensation

☐ Velocity Feedforward ☐ Acceleration Feedforward ☐ Torque Offset

☐ Output Filter

3. In the Travel Limit and Speed boxes, enter appropriate values.

In this example, Travel Limit = 5 and Speed = 2. The actual value of programmed units depend on your application.

4. From the Direction pull-down menu, choose the appropriate direction (Forward Uni-directional is default).

5. Check the Tune boxes appropriate for your application.

6. Apply Drive Enable (Input 1) signal (CN1-31) for the axis you are tuning.

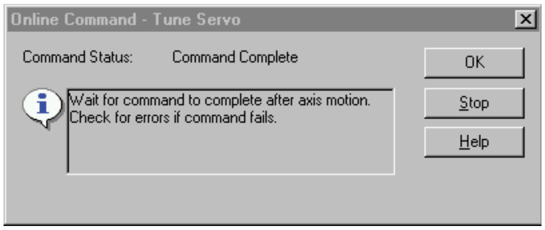
This step is required only if you checked the box for Drive Enable Input Checking, in the Drive/Motor tab, Axis Properties dialog.

ATTENTION

To avoid personal injury or damage to equipment, apply 24V Drive Enable signal (CN1-31) only to the axis you are tuning.

7. Click Start Tuning to auto-tune your axis.

The Online Command - Tune Servo dialog opens. When the test completes, the Command Status changes from Executing to Command Complete.



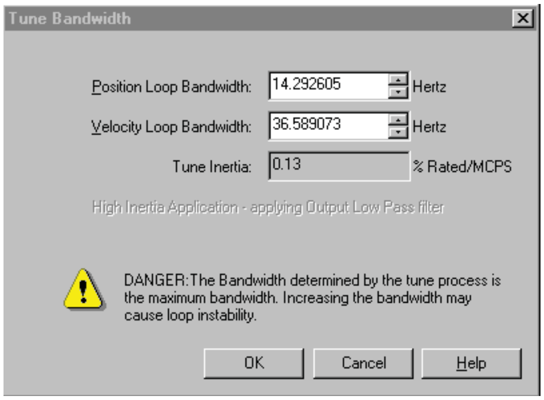
Online Command - Tune Servo

Command Status: Command Complete

Wait for command to complete after axis motion.
Check for errors if command fails.

8. Click OK.

The Tune Bandwidth dialog opens.



Tune Bandwidth

Position Loop Bandwidth: 14.292605 Hertz

Velocity Loop Bandwidth: 36.589073 Hertz

Tune Inertia: 0.13 % Rated/MCPS

High Inertia Application - applying Output Low Pass filter

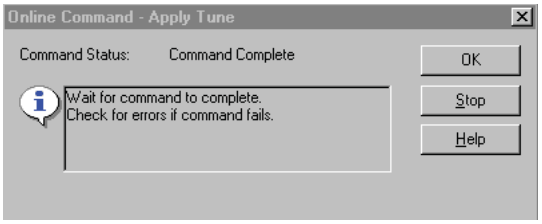
DANGER:The Bandwidth determined by the tune process is the maximum bandwidth. Increasing the bandwidth may cause loop instability.

Actual bandwidth values (Hz) depend on your application and may require adjustment once motor and load are connected.

9. Record your bandwidth data for future reference.

10. Click OK.

The Online Command - Apply Tune dialog opens. When the test completes, the Command Status changes from Executing to Command Complete.



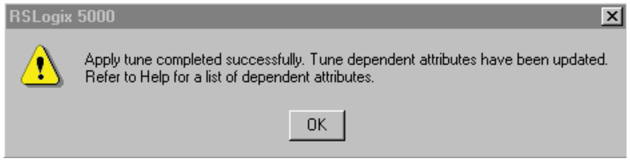
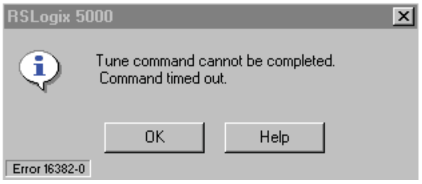
Online Command - Apply Tune

Command Status: Command Complete

Wait for command to complete.
Check for errors if command fails.

11. Click OK.

12. Determine if your test completed successfully.

If	Then
Your test completes successfully, this dialog appears.  RSLogix 5000 Apply tune completed successfully. Tune dependent attributes have been updated. Refer to Help for a list of dependent attributes. 	1. Click OK. 2. Remove Drive Enable (Input 1) signal (CN1-31) applied earlier. 3. You are finished tuning your axis.
Your test failed, this dialog appears.  RSLogix 5000 Tune command cannot be completed. Command timed out. Error 16382-0	1. Click OK. 2. Make an adjustment to motor velocity. 3. Refer to appropriate Logix motion module setup and configuration manual for more information. 4. Return to step 7 and run the test again.

Configuring Your Ultra3000 Drive with DeviceNet

The procedures in this section are listed in this table and apply to Ultra3000-DN drives with indexing.

Ultra3000 Drive Configuration Procedures

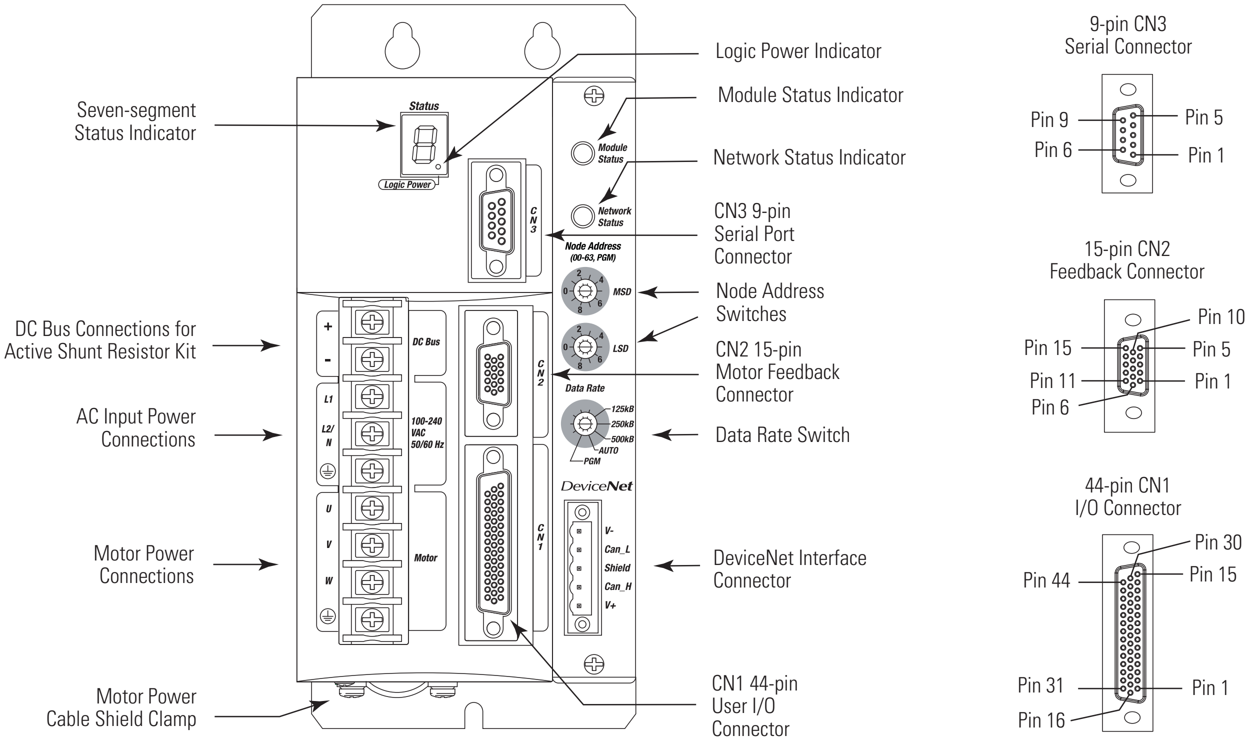
Procedure	Page
Configure Your Ultra3000 Drive with DeviceNet	93
Apply Power to Your Ultra3000 Drive with DeviceNet	94

These procedures assume you have completed wiring the DeviceNet interface connector on your Ultra3000-DN drive.

Front Panel Connections

Use this figure to locate the front panel connections on the Ultra3000-DN 230V drives (500W, 1 kW, and 2 kW).

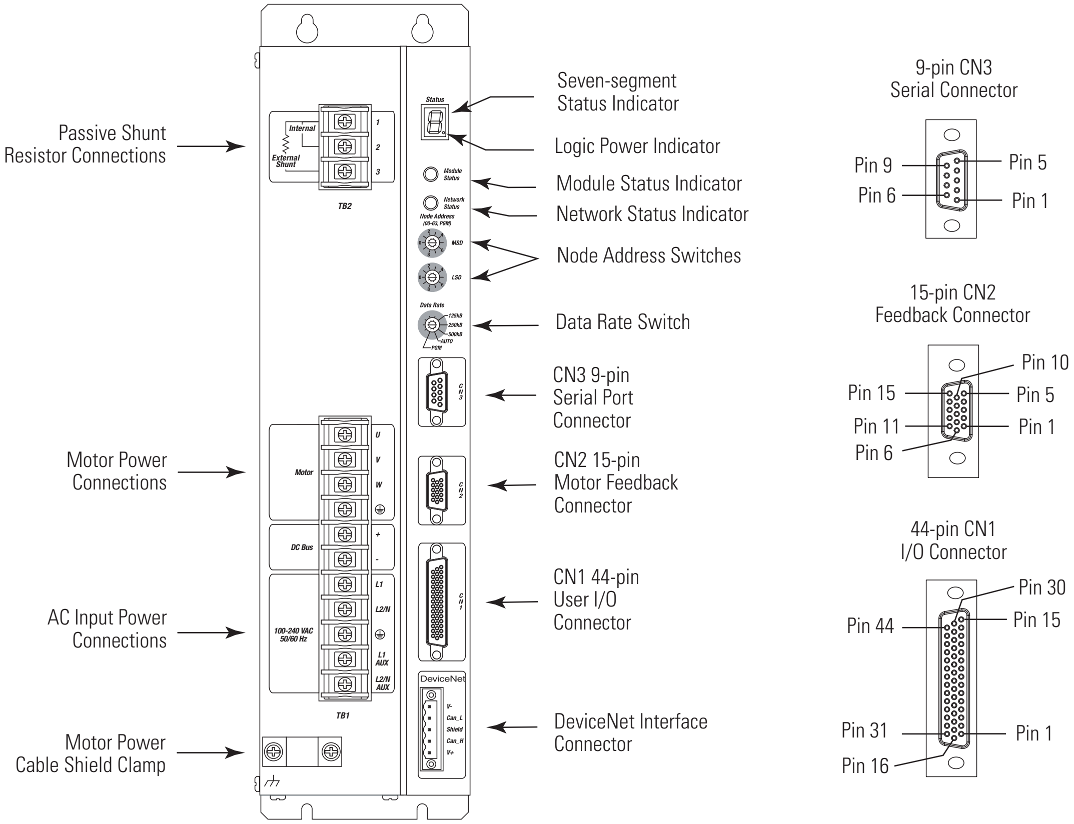
Front Panel Connections for 2098-DSD-005-DN, 2098-DSD-005X-DN, 2098-DSD-010-DN, 2098-DSD-010X-DN, 2098-DSD-020-DN, and 2098-DSD-020X-DN Drives



For CN1, CN2, and CN3 connector pin-out information, refer to the Ultra3000 Digital Servo Drives Installation Manual, publication 2098-IN003.

Use this figure to locate the front panel connections on the Ultra3000-DN 230V drives (3 kW).

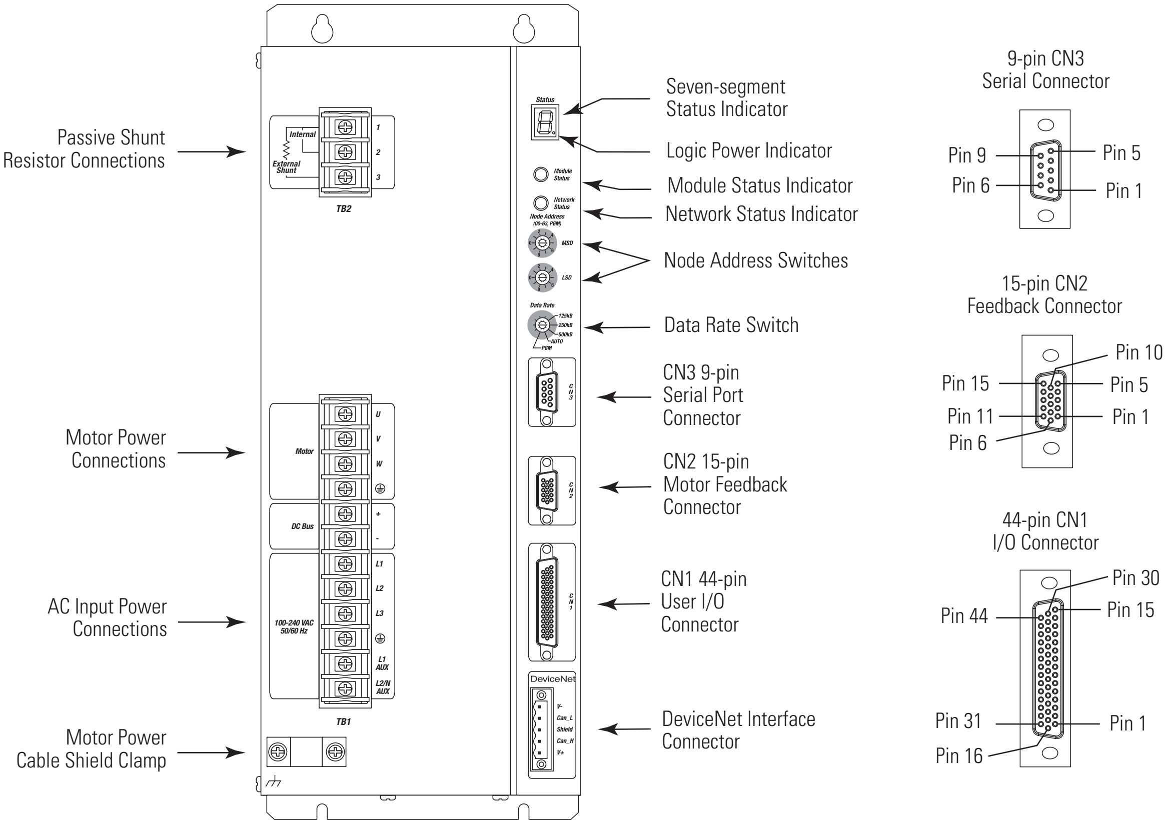
Front Panel Connections for 2098-DSD-030-DN and 2098-DSD-030X-DN Drives



For CN1, CN2, and CN3 connector pin-out information, refer to the Ultra3000 Digital Servo Drives Installation Manual, publication 2098-IN003.

Use this figure to locate the front panel connections on the Ultra3000-DN 230V drives (7.5 and 15 kW).

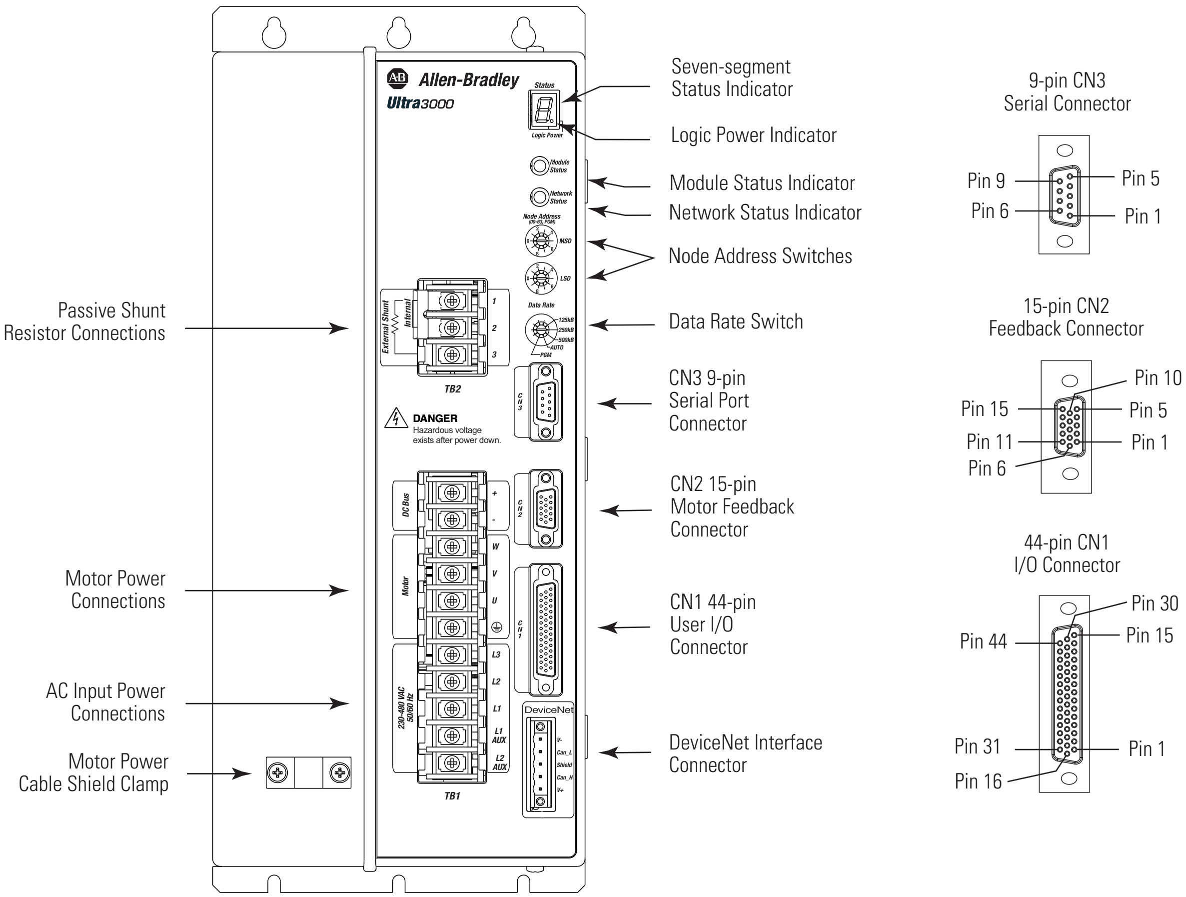
Front Panel Connections for 2098-DSD-075-DN, 2098-DSD-075X-DN, 2098-DSD-150-DN, and 2098-DSD-150X-DN Drives



For CN1, CN2, and CN3 connector pin-out information, refer to the Ultra3000 Digital Servo Drives Installation Manual, publication 2098-IN003.

Use this figure to locate the front panel connections on the Ultra3000-DN 460V drives (3 kW, 5 kW, 10 kW, 15 kW, and 22 kW).

Front Panel Connections for 2098-DSD-HV*xxx*-DN and 2098-DSD-HV*xxx*X-DN Drives



For CN1, CN2, and CN3 connector pin-out information, refer to the Ultra3000 Digital Servo Drives Installation Manual, publication 2098-IN003.

Configure Your Ultra3000 Drive with DeviceNet

Follow these steps to configure your Ultra3000-DN drive.

1. Verify that there is no power applied to the drive and that the DeviceNet cable is connected.

   Refer to the figures on pages 89…92 for switch locations.

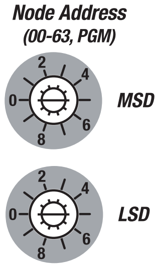
2. Set the node address for each drive in your system.

   Valid node addresses are 00…63 and PGM. The MSD rotary switch sets the most significant digit and the LSD rotary switch sets the least significant digit.

   Refer to the figures on pages 89…92 for the switch locations. Refer to this table for examples.

Node Address	MSD Switch	LSD Switch
10	1	0
11	1	1
12	1	2

Use the MSD and LSD rotary switches on the DeviceNet drive panel to set node addresses.



Selecting an invalid node address (> 63) sets the node address according to a non-volatile parameter stored in the drive.

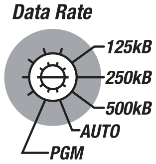
3. Set the data rate.

   Valid data rates are 125, 250, and 500 Kbps, AUTO, and PGM.

   Refer to the figures on pages 89…92 for the switch location.

   Selecting AUTO automatically matches the device data rate to the rate of the network. Selecting PGM sets the data rate according to a non-volatile parameter stored in the drive.

Use the Data Rate rotary switch on the DeviceNet drive panel to set the data rate.



Apply Power to Your Ultra3000 Drive with DeviceNet

This procedure assumes you have wired your Ultra3000-DN system, verified the wiring, and are ready to begin using Ultraware software.

ATTENTION ⚠

High voltage exists in ac line filters. The filter must be grounded properly before applying power. Filter capacitors retain high voltages after power removal. Before handling the equipment, voltages should be measured to determine safe levels. Failure to observe this precaution could result in personal injury.

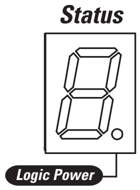
Follow these steps to apply power to your Ultra3000-DN drive.

1. Disconnect any load to the motor, making sure the motor is free of all linkages when initially applying power to the system.

   ATTENTION ⚠

   To avoid damage to the drive due to improper sequencing of input power and the Drive Enable signal, you must assign one of the eight inputs as Drive Enable (Input 1 is the default setting).

2. Apply input power to the Ultra3000-DN drive and observe the front panel Logic Power status indicator.



If the Logic Power status indicator is	Then
ON	Go to step 3.
Not ON	1. Check your input power connections. 2. Repeat step 2.

3. Observe the front panel seven-segment status indicator on your Ultra3000-DN drive.

Seven-segment Status Indicator	Status	Do This
Actively cycling segments in a full circle	The drive is ready.	Go to step 4.
Flashing E followed by two numbers	The drive is faulted.	Go to Error Codes on page 98.

4. Observe the module status indicator.

Module Status Indicator	Status	Do This
Steady green	The drive is ready.	Go to step 5.
Not steady green	The drive is faulted.	Go to DeviceNet Module Status Indicator on page 109.

5. Observe the network status indicator.

Network Status Indicator	Status	Do This
Off	Establishing communication with network.	Wait for flashing or steady green.
Flashing or steady green	Communication is ready.	Go to step .
Not flashing or steady green	The drive is faulted.	Go to DeviceNet Network Status Indicator on page 109.

For further commissioning procedures, refer to the following table for procedures.

Ultra3000 Drive Configuration Procedures

Procedure	Page
Detect Your Ultra3000 Drive	16
Understanding the Workspace and Drive Branches	17
Select a Motor	21
Tune Your Motor	22
Configure Displayed Units	23
Test Your Motor (non-indexing move)	25
Test Your Motor (indexing move)	26
Indexing and Non-indexing Move Examples	29

Refer to the Ultra3000 with DeviceNet Reference Manual, publication 2098-RM001, for information on communicating with the Ultra3000 drives using DeviceNet.

Chapter 2

Troubleshooting Your Ultra3000 Servo Drive

Introduction

This chapter provides troubleshooting tables for your Ultra3000 servo drive.

Safety Precautions

Observe the following safety precautions when troubleshooting your Ultra3000 servo drive.

ATTENTION

DC bus capacitors may retain hazardous voltages after input power has been removed. Before working on the drive, measure the dc bus voltage to verify it has reached a safe level or wait the full time interval listed on the drive warning label. Failure to observe this precaution could result in severe bodily injury or loss of life.

Do not attempt to defeat or override the drive fault circuits. You must determine the cause of a fault and correct it before you attempt to operate the system. If you do not correct a drive or system malfunction, it could result in personal injury and/or damage to the equipment as a result of uncontrolled machine system operation.

Test equipment (such as an oscilloscope or chart recorder) must be properly grounded. Failure to include an earth ground connection could result in a potentially fatal voltage on the oscilloscope chassis.

General Troubleshooting

Refer to the Error Codes section below to identify problems, potential causes, and appropriate actions to resolve the problems. If problems persist after attempting to troubleshoot the system, please contact your Rockwell Automation sales representative for further assistance.

Determine Ultra3000 Drive Status

Ultra3000 Drives with the Logic Power Status Indicator ON	Status Indicator	Then
2098-DSD-*xxx*, 2098-DSD-*xxx*X, 2098-DSD-HV*xxx*, or 2098-DSD-HV*xxx*X	Actively cycling segments in a full circle	Your Ultra3000 drive is ready.
2098-DSD-*xxx*-DN, 2098-DSD-*xxx*X-DN, 2098-DSD-HV*xxx*-DN, 2098-DSD-HV*xxx*X-DN		Your Ultra3000 drive is ready.
2098-DSD-*xxx*-SE or 2098-DSD-HV*xxx*-SE drive	Displaying a fixed 4	Your Ultra3000 drive is ready.
All Ultra3000 drives	Flashing E followed by two numbers	Your Ultra3000 drive has an error. Proceed to the section Error Codes below.
	Flashing L	Your Ultra3000 drive is in an Overtravel condition and motion restrictions are in effect.

Error Codes

The following list of problematic symptoms (no error code shown) and problems with assigned error codes is designed to help you resolve anomalies.

When a fault is detected, the seven-segment status indicator will display E followed by the flashing of the two-digit error code, one digit at a time. This is repeated until the problem is cleared.

Seven-segment Status Indicator Error Codes

Error Code	Problem or Symptom	Possible Cause	Action/Solution
	Power (PWR) indicator not ON	No ac power or auxiliary logic power.	Verify ac power or auxiliary +5V logic power is applied to the Ultra3000.
		Internal power supply malfunction.	Call your Allen-Bradley representative.
	Power (PWR) indicator is ON, but seven-segment status indicator is OFF. Note: This only applies to Ultra3000 models 2098-DSD-005, 2098-DSD-010, and 2098-DSD-020.	Externally applied +5V auxiliary power supply voltage is too low.	Verify that the external +5V auxiliary power supply (as measured at the drive terminals) reads between 5.10V and 5.25V.
	Motor jumps when first enabled	Motor wiring error.	Check motor wiring.
		Incorrect motor chosen.	Verify the proper motor is selected.
	Digital I/O not working correctly	I/O power supply disconnected.	Verify connections and I/O power source.
E01	Non-Volatile Memory Endurance Exceeded	Range of motion and number of home position definitions during the product life exceeds the maximum allowed (applies only to systems with absolute feedback).	This is an unrecoverable fault, the drive must be sent back to the factory.

Error Code	Problem or Symptom	Possible Cause	Action/Solution
E02	Velocity Exceeds Position Rollover /2	The velocity command or feedback exceeds half the machine cycle length per millisecond (applies only when the machine cycle position rollover is enabled).	Increase machine cycle size or reduce velocity profile.
E03	Absolute Feedback Range Exceeded	The motor position exceeds +/- 2047 revolutions from the home position (applies only to systems with absolute feedback).	• Decrease application range of motion. • Upgrade firmware. This error only applies to firmware versions prior to 1.10.
E04	Motor Overtemperature	Motor thermostat trips due to: • High motor ambient temperature and/or • Excessive current	• Operate within (not above) the continuous torque rating for the ambient temperature (40°C maximum). • Lower ambient temperature, increase motor cooling.
		Motor wiring error.	Check motor wiring.
		Incorrect motor selection.	Verify the proper motor has been selected.
E05	IPM Fault	Motor cables shorted.	Verify continuity of motor power cable and connector.
		Motor winding shorted internally.	Disconnect motor power cables from the motor. If the motor is difficult to turn by hand, it may need to be replaced.
		Ultra3000 temperature too high.	• Check for clogged vents or defective fan. • Ensure cooling is not restricted by insufficient space around the unit.
		Operation above continuous power rating.	• Verify ambient temperature is not too high. • Operate within the continuous power rating. • Reduce acceleration rates.
		Ultra3000 has a bad IPM output, short circuit, or overcurrent.	Remove all power and motor connections, and preform a continuity check from the dc bus to the U, V, and W motor outputs. If a continuity exists, check for wire fibers between terminals, or send drive in for repair.
		An attempt was made to enable the drive without waiting at least 1.0 second after applying the main ac power. Note: This only applies to Ultra3000 models 2098-DSD-005, 2098-DSD-010, and 2098-DSD-020 (when using an external +5V auxiliary power supply).	Wait at least 1.0 second after the main ac is applied before enabling the drive.
E06	Hardware Overtravel (SERCOS only)	Dedicated overtravel input is inactive.	• Check wiring. • Verify motion profile.
E07	RESERVED		Call your local Allen-Bradley representative.
E08	RESERVED		
E09	Bus Undervoltage	Low ac line/ac power input.	• Verify voltage level of the incoming ac power. • Check ac power source for glitches or line drop. • Install an uninterruptible power supply (UPS) on your ac input.

Error Code	Problem or Symptom	Possible Cause	Action/Solution
E10	Bus Overvoltage	Excessive regeneration of power. When the motor is driven by an external mechanical power source, it may regenerate too much peak energy through the Ultra3000's power supply. The system faults to save itself from an overload.	• Change the deceleration or motion profile. • Use a larger system (motor and Ultra3000). • Use a resistive shunt. • If a shunt is connected, verify the wiring is correct and shunt fuse is not blown.
		Excessive ac input voltage.	Verify input is within specifications.
E11	Illegal Hall State	Incorrect phasing.	Check the Hall phasing.
		Bad connections.	• Verify the Hall wiring. • Verify 5V power supply to the encoder.
E12	Home Search Failed	Home sensor and/or marker is outside the overtravel limits.	• Check wiring. • Reposition the overtravel limits or sensor.
E13	Home Position In Limit	Home sensor, marker, or final home position exceeds a hardware overtravel limit.	• Reposition the overtravel limits or home sensor. • Adjust the final home position.
E14	SERCOS Hardware Fault (SERCOS drives only)	A fault was detected with the operation of the drive's internal SERCOS hardware.	Contact your local Allen-Bradley representative.
	DeviceNet Communications Network problem (DeviceNet drives only)	DeviceNet communications network is broken	Troubleshoot DeviceNet communications.
E15	Excessive Electrical Cycle Length	Electrical cycle length exceeds maximum lines per electrical cycle	Replace the linear motor/encoder.
E16	Software Overtravel (SERCOS only)	Programmed overtravel limit has been exceeded.	• Verify motion profile. • Verify overtravel settings are appropriate.
E17	User-specified Current Fault	User-specified average current level has been exceeded.	Increase to a less restrictive setting.
E18	Overspeed Fault	Motor speed has exceeded 125% of maximum rated speed.	• Check cables for noise. • Check tuning.
E19	Excess Position Error	Position error limit was exceeded.	• Increase the feedforward gain. • Increase following error limit or time. • Check position loop tuning.

Error Code	Problem or Symptom	Possible Cause	Action/Solution
E20	Motor Encoder State Error	The motor encoder encountered an illegal transition.	• Replace the motor/encoder. • Use shielded cables with twisted pair wires. • Route the feedback away from potential noise sources. • Check the system grounds. • Verify that the unbuffered encoder signals are not subjected to EMI in the CN1 cable. Remove these signals from the CN1 cable if they are not being used. • Verify that the motor has a high-frequency bond to the drive's enclosure panel. • Verify that any stage connected to the motor shaft (for example using a ball screw) has a high-frequency bond to the machine frame and the drive's enclosure panel.
		Bad encoder.	Replace motor/encoder.
E21	Auxiliary Encoder state error	The auxiliary encoder encountered an illegal transition.	• Use shielded cables with twisted pair wires. • Route the encoder cable away from potential noise sources. • Faulty encoder, replace encoder. • Check the ground connections.
		Setup time violation for Step/Direction or CW/CCW input.	Check timing of Step/Direction or CW/CCW inputs to determine if setup time requirements are being met.
E22	Motor Thermal Protection Fault	The internal filter protecting the motor from overheating has tripped.	• Reduce acceleration rates. • Reduce duty cycle (ON/OFF) of commanded motion. • Increase time permitted for motion. • Use larger Ultra3000 and motor. • Check tuning.
E23	IPM Thermal Protection Fault	The internal filter protecting the drive from over heating has tripped.	• Reduce acceleration rates. • Reduce duty cycle (ON/OFF) of commanded motion. • Increase time permitted for motion. • Use larger Ultra3000 and motor. • Check tuning.
E24	Excess Velocity Error	Velocity error limit was exceeded.	• Increase time or size of allowable error. • Reduce acceleration. • Check tuning.
E25	Sensor Not Assigned	Homing or registration motion was attempted without a sensor assigned.	Assign a sensor to a digital input.
E26	User-specified Velocity Fault	User-specified velocity level was exceeded.	Increase to a less restrictive setting.
E27	Axis Not Homed	Absolute positioning was attempted without homing.	Verify homing sequence.

Error Code	Problem or Symptom	Possible Cause	Action/Solution
E28	Motor Parameter Error	Parameter loaded from smart encoder or received from SERCOS controller is incompatible with the drive.	• Select a different motor through the SERCOS controller. • Select a different motor.
E29	Encoder Output Frequency Exceeded	Encoder output frequency exceeds the maximum user-specified value. This only applies when the encoder output is synthesized by the drive.	• Increase the encoder output maximum frequency parameter. • Decrease the encoder interpolation parameter. • Increase the encoder output divider parameter.
E30	Encoder Communication Fault	Communication was not established with an intelligent encoder.	• Verify motor selection. • Verify the motor supports automatic identification. • Verify motor encoder wiring.
E31	Encoder Data	Encoder data is corrupted.	Replace the motor/encoder.
E32	Sine/Cosine Encoder Frequency Limit Exceeded	Maximum frequency of the sine/cosine circuitry has been exceeded.	• Decrease velocity. • Use encoder with lower resolution (before interpolation).
E33	Absolute Position Exceeds Position Rollover	Motion is commanded to a position outside the position rollover range. • An absolute index is initiated that specifies a position outside the position rollover range. • A homing cycle is initiated with the home position outside the position rollover range. • A define home is initiated with the home position outside the position rollover range. • A preset position is initiated that specifies a position outside the position rollover range.	Set motion command to a position within the position rollover range.
E34	Ground Fault	Wiring error.	Check motor power wiring.
		Motor internal ground short.	Replace motor.
		Internal malfunction.	Disconnect motor power cable from drive and enable drive with current limit set to 0. If fault remains, call your Allen-Bradley representative. If fault clears, then a wiring error or motor internal problem exists.
E35	Precharge Fault	Low ac input voltage.	Check input ac voltage on all phases.
		Internal malfunction.	Call your Allen-Bradley representative.
E36	Power Circuitry Overtemperature	Excessive heat exists in the power circuitry.	• Reduce acceleration rates. • Reduce duty cycle (ON/OFF) of commanded motion. • Increase time permitted for motion. • Use larger Ultra3000 and motor. • Check tuning.
E37	AC Line Loss	One or more phases of the input ac power is missing.	Check input ac voltage on all phases.

Error Code	Problem or Symptom	Possible Cause	Action/Solution
E39	Self-sensing Commutation Startup Error	Motion required for self-sensing startup commutation was obstructed.	• Verify that there are no impediments to motion at startup, such as hard limits. • Increase self-sensing current if high friction or load conditions exist. • Check motor or encoder wiring using wiring diagnostics.
E40	230V Shunt Protection Fault	Ineffective shunt resistor	• Verify that the shunt resistor (internal or external) is connected. • If an external shunt resistor is connected, verify that the shunt fuse is not blown. • If a non Allen-Bradley external shunt resistor is used, verify that the resistance value is within specifications. • Verify that the motor is not being driven mechanically, causing the motor to behave as a generator.
		Excessive regeneration	
E41	460V Shunt Protection Fault	Ineffective shunt resistor	
		Excessive regeneration	
E42	Motor Keying Error (SERCOS drives only)	The motor physically connected to the drive differs from the motor specified in the user program.	Select the correct motor in the user program.
E43	Drive Enable Input (SERCOS drives only)	• An attempt was made to enable the axis through software while the Drive Enable hardware input was inactive. • The Drive Enable input transitioned from active to inactive while the axis was enabled.	• Disable the Drive Enable Input fault. • Verify that Drive Enable hardware input is active whenever the drive is enabled through software.
E44	Lost motion fault (only applies to applications with Stegmann feedback devices)	Detection occurs during a fault reset. Absolute position in the drive is incorrect and the motion has been lost due to line loss condition.	• Cycle power. • Cycle power and re-home drive if drive was homed in the same power cycle that the lost motion fault occurred.
E50	Duplicate Node Fault (SERCOS drives only)	Duplicate node address detected on SERCOS ring.	Verify that each SERCOS drive is assigned a unique node address.
All others	RESERVED		Call your local Allen-Bradley representative.

Troubleshooting for SERCOS Drives

These troubleshooting tables apply to Ultra3000-SE drives (2098-DSD-*xxx*-SE and 2098-DSD-HV*xxx*-SE).

SERCOS Module Status Indicator

SERCOS Module Status Indicator	Status	Potential Cause	Possible Resolution
Steady green	Normal	Drive is enabled.	Normal operation when drive is enabled.
Flashing green	Standby	Drive is not enabled.	Normal operation when drive is disabled.
Flashing red-green	DC Bus Undervoltage	The dc bus voltage is low.	• Normal operation when using auxiliary power (main ac power is not applied). • When using main ac power, refer to the section Error Codes to continue troubleshooting.
Flashing red	Minor fault	Drive is faulted, but the fault can be cleared.	Refer to the section Error Codes to continue troubleshooting.
Steady red	Unrecoverable fault	Drive is faulted, and the fault cannot be cleared.	Contact your local Allen-Bradley representative.

SERCOS Network Status Indicator

SERCOS Network Status Indicator	Status	Potential Cause	Possible Resolution
Steady green	Communication ready	No faults or failures.	N/A
Flashing green	Establishing communication	System is still in the process of establishing SERCOS communication.	Wait for steady green status indicator.
		Node address setting on the drive module does not match SERCOS controller configuration.	Verify proper node switch setting.
Flashing red	No communication [1]	Loose fiber optic connection.	Verify proper fiber optic cable connections.
		Broken fiber optic cable.	Replace fiber optic cable.
		Receive fiber optic cable connected to SERCOS transmit connector and vice versa.	Check proper SERCOS fiber optic cable connections.

[1] Refer to Fiber Optic Cable Installation and Handling Instructions, publication 2090-IN010, for more information.

Understanding Drive Fault Behavior

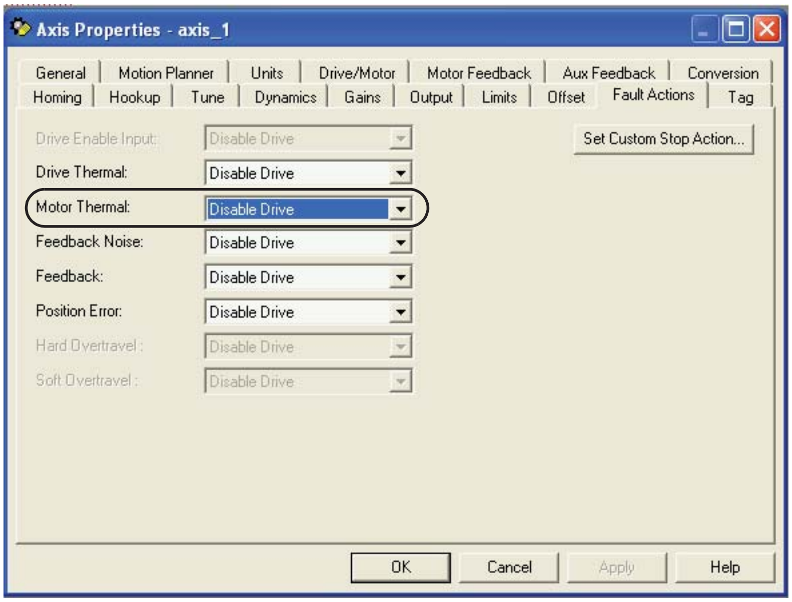
The following RSLogix 5000 fault actions are configurable from the Axis Properties dialog, Fault Actions tab.

RSLogix 5000 Drive Fault Action Definitions

Drive Fault Action	Definition
Shutdown	The drive disables and the contactor enable relay opens. Uncontrolled stop, motor coasts to a stop.
Disable Drive	The drive is disabled. Uncontrolled Stop, motor coasts to a stop.
Stop Motion	N/A. Drive continues to operate. Status is provided by the Module Status Indicator.
Status Only	Drive continues to operate. Status is provided by the Module Status Indicator.

Ultraware software handles all hard faults by disabling the drive. RSLogix 5000 software provides the four fault handling options, however, for the Ultra3000 drives, only Shutdown and Disable Drive are active settings and both result in a coast-to-stop condition. Disable Drive is the default fault action for all attributes and only selected faults are programmable. In the Drive Fault Behavior table on page 106, the controlling attribute is given for programmable fault actions.

RSLogix 5000 Axis Properties - Fault Actions Tab



Drive Enable Input: Disable Drive
Drive Thermal: Disable Drive
Motor Thermal: Disable Drive
Feedback Noise: Disable Drive
Feedback: Disable Drive
Position Error: Disable Drive
Hard Overtravel : Disable Drive
Soft Overtravel : Disable Drive

Default setting for Motor Overtemperature fault (E04).

Refer to Appendix C, beginning on page 141, for information on how to reduce unexpected motion as a result of feedback signal loss and setting the Position Error fault action.

When a fault is detected, the seven-segment status indicator displays E followed by the flashing two-digit error code, one digit at a time. This is repeated until the fault is cleared.

Drive Fault Behavior

Software Fault Message RSLogix 5000 (Ultraware)	Error Code	Description	Drive Fault Action/Attribute	RSLogix 5000 Programmable Fault Action?
DriveHardFault (Non-Volatile Memory Endurance Exceeded)	E01	Range of motion and number of home position definitions during the product life exceeds the maximum allowed (applies only to systems with absolute feedback).	DISABLE	NO
DriveHardFault (Velocity Exceeds Position Rollover /2)	E02	The velocity command or feedback exceeds half the machine cycle length per millisecond (applies only when the machine cycle position rollover is enabled).	DISABLE	NO
DriveHardFault (Absolute Feedback Range Exceeded)	E03	The motor position exceeds +/- 2047 revolutions from the home position (applies only to systems with absolute feedback).	DISABLE	NO
MotorOvertempFault (Motor Overtemperature)	E04	The motor thermal switch was tripped.	DISABLE/ Motor Thermal	YES
DriveHardFault (IPM Fault)	E05	A problem in the drive power structure was detected. Motor cables shorted, motor winding shorted internally, temperature too high, operation above continuous power rating, or has a bad IPM output, short circuit, or overcurrent.	DISABLE	NO
HardOvertravelFault (+/- Hard Overtravel)	E06	Axis moved beyond the physical travel limits in the positive/negative direction. This fault can be configured for status only.	DISABLE/ Hard Overtravel	YES
MotFeedbackFault (Channel BM Line Loss)	E07	The feedback wiring is open, shorted or missing.	DISABLE	NO
MotFeedbackFault (Channel AM Line Loss)	E08	The feedback wiring is open, shorted or missing.	DISABLE	NO
DriveUndervoltageFault (Bus Undervoltage)	E09	With 3-phase present, the dc bus voltage is below limits. The trip point if 275V and 137V dc for 460V/230V drives respectively.	DISABLE	NO
DriveOvervoltageFault (Bus Overvoltage)	E10	The dc bus voltage is above limits. The trip point is 820V and 410V dc for 460V/230V drives respectively.	DISABLE	NO
CommutationFault (Illegal Hall State)	E11	State of Hall inputs is incorrect.	DISABLE	NO
DriveHardFault (Home Search Failed)	E12	Home sensor and/or marker is outside the overtravel limits.	DISABLE	NO
DriveHardFault (Home Position In Limit)	E13	Home sensor, marker, or final home position exceeds a hardware overtravel limit.	DISABLE	NO
SERCOSFault (SERCOS or DeviceNet Communications Network problem)	E14	SERCOS or DeviceNet communications network is broken	DISABLE	NO
DriveHardFault (Excessive Electrical Cycle Length)	E15	Configuration information is not valid.	N/A	NO
SoftOvertravelFault (+/- Software Overtravel)	E16	Programmed positive/negative overtravel limit has been exceeded.	DISABLE Soft Overtravel	YES

Software Fault Message **RSLogix 5000** (Ultraware)	**Error Code**	**Description**	**Drive Fault Action/Attribute**	**RSLogix 5000 Programmable Fault Action?**
DriveHardFault (User-specified Current Fault)	E17	User-specified average current level has been exceeded.	DISABLE	NO
OverspeedFault (Overspeed Fault)	E18	Motor speed has exceeded 125% of maximum rated speed.	DISABLE	NO
PositionErrorFault (Excess Position Error)	E19	Axis position error limit has been exceeded. This fault can be configured for status only.	DISABLE Position Error	YES
MotFeedbackFault (Motor Encoder State Error)	E20	The motor encoder encountered an illegal transition.	DISABLE	NO
AuxFeedbackNoiseFault (Auxiliary Encoder State Error)	E21	The auxiliary encoder encountered an illegal transition.	DISABLE	NO
OverloadFault (Motor Thermal Protection Fault)	E22	The internal filter protecting the motor from overheating has tripped.	DISABLE	NO
DriveOvertempFault (IPM Thermal Protection Fault)	E23	The internal filter protecting the drive from over heating has tripped.	DISABLE Drive Thermal	YES
DriveHardFault (Excess Velocity Error)	E24	Velocity error limit was exceeded.	DISABLE	NO
DriveHardFault (Sensor Not Assigned)	E25	Homing or registration motion was attempted without a sensor assigned.	DISABLE	NO
DriveHardFault (User-specified Velocity Fault)	E26	User-specified velocity level was exceeded.	DISABLE	NO
DriveHardFault (Axis Not Homed)	E27	Absolute positioning was attempted without homing.	DISABLE	NO
DriveHardFault (Motor Parameter Error)	E28	Parameter loaded from smart encoder or received from SERCOS controller is incompatible with the drive.	DISABLE	NO
DriveHardFault (Encoder Output Frequency Exceeded)	E29	Encoder output frequency exceeds the maximum user-specified value. This only applies when the encoder output is synthesized by the drive.	DISABLE	NO
DriveHardFault (Encoder Communication Fault)	E30	Communication was not established with an intelligent (i.e. Stegmann) encoder on the motor feedback port.	DISABLE	NO
DriveHardFault (Encoder Data)	E31	Encoder data is corrupted.	DISABLE	NO
DriveHardFault (Sine/Cosine Encoder Frequency Limit Exceeded)	E32	Maximum frequency of the sine/cosine circuitry has been exceeded.	DISABLE	NO
DriveHardFault (Absolute Position Exceeds Position Rollover)	E33	Absolute position exceeds position rollover.	DISABLE	NO
DriveHardFault (Ground Fault)	E34	Excessive ground current in the converter was detected.	DISABLE	NO

Software Fault Message **RSLogix 5000** (Ultraware)	**Error Code**	**Description**	**Drive Fault Action/Attribute**	**RSLogix 5000 Programmable Fault Action?**
DriveHardFault (Precharge Fault)	E35	The converter pre-charge cycle has failed.	DISABLE	NO
DriveHardFault (Power Circuitry Overtemperature)	E36	Excessive heat exists in the power circuitry.	DISABLE	NO
DriveHardFault (AC Line Loss)	E37	One or more phases of the input ac power is missing.	DISABLE	NO
SERCOSFault (SERCOS Ring Fault)	E38	The SERCOS ring is not active after being active and operational.	DISABLE	NO
DriveHardFault (Self-sensing Commutation Startup Error)	E39	Self-sense commutation fault detected.	DISABLE	NO
DriveHardFault (230V Shunt Protection Fault)	E40	Ineffective shunt resistor.	DISABLE	NO
DriveHardFault (460V Shunt Protection Fault)	E41	Ineffective shunt resistor.	DISABLE	NO
ConfigFault (Motor Keying Error)	E42	The motor physically connected to the drive differs from the motor specified in the user program.	N/A	NO
DriveEnableInputFault (Drive Enable Input)	E43	Generated when Enable input switches off when drive is enabled.	DISABLE Drive Enable Input	YES
DriveHardFault (Unknown Fault)	E44	Lost motion fault (only applies to applications with Stegmann feedback devices)	DISABLE	NO
SERCOSFault (Duplicate Node Fault)	E50	Duplicate node address detected on SERCOS ring.	DISABLE	NO
RESERVED	All Others			

Troubleshooting for DeviceNet Drives

These troubleshooting tables apply to Ultra3000-DN drives (2098-DSD-*xxx*-DN, 2098-DSD-*xxx*X-DN, 2098-DSD-HV*xxx*-DN, or 2098-DSD-HV*xxx*X-DN).

DeviceNet Module Status Indicator

Module Status Indicator	Status	Potential Cause	Possible Resolution
Off	Not powered	No power	There is no power going to the device.
Steady-green	Operational	Normal operation	Normal operation - no action needed.
Flashing-green	Device is in stand-by	Processing or waiting for input	Normal operation - no action needed.
Flashing-red	Recoverable fault	Not operational	Power cycle or reset the drive.
Steady-red	Unrecoverable fault	Drive problem	1. Check drive for power-up error. 2. Replace drive.
Flashing-red/ Flashing-green	Self testing	Self-test in progress	The device is in self test, wait.

DeviceNet Network Status Indicator

Network Status Indicator	Status	Potential Cause	Possible Resolution
Off	• Not powered • Not online	• No power going to the device • Failed Duplicate MAC ID check	1. Check the Module Status indicator to verify that the drive is powered. 2. Check that one or more nodes are communicating on the network. 3. Check that at least one other node on the network is operational and the data rate is the same as the drive.
Flashing-green	• Online • Not connected	• Passed Duplicate MAC ID check • No connection established	No action is needed. The indicator is flashing to signify that there are no open communication connections between the drive and any other device. Any connection (I/O or explicit message) made to the drive over DeviceNet will cause the indicator to stop flashing and remain Steady-ON for the duration of any open connection.
Steady-green	• Online • Connected	One or more connections established	No action needed. This condition is normal.
Flashing-red	• Online • Time-out	I/O connection timed out	1. Re-initiate I/O messaging by the master controller. 2. Reduce traffic or errors on the network so that messages can get through within the necessary time frame.
Steady-red	Network Failure	• Failed Duplicate MAC ID check • Bus-off	1. Ensure that all nodes have unique addresses. 2. If all node addresses are unique, examine network for correct media installation. 3. Ensure that all nodes have the same Data Rate.

Node Problems

Give particular attention to the task of setting initial addresses and data rates. Survey the network to ensure all assignments are known. Some nodes can be logically assigned to a group of devices, but physically located away from those devices. One incorrect node can cause other nodes to appear to be bus-off (steady-red indicator). If a node goes bus-off and the device is reset only to go bus-off again, the problem is likely not with the device, but rather the setting of the address, data rate, or a network-wide problem related to topology, grounding, intermittent power/data connections, or electrical noise. If a scanner goes bus-off, nodes will not reallocate (Flashing-green or red) even if they are functioning correctly.

Device Failure - Indicator Status Check

A steady-red Module Status indicator can mean an error. If the Network Status indicator goes steady-red at power-up, it could mean there is a Duplicate MAC ID. The user response is to test all devices for unique addresses. If a steady-red indicator remains on after the Duplicate MAC ID test shows all devices to have a unique node address, it means a bus-off error.

Follow these steps to resolve the error.

1. Check data rate settings.
2. If symptom persists, replace node address (with another address and correct data rate).
3. If symptom persists, replace tee tap.
4. If symptom persists, check topology.
5. If symptom persists, check power for noise with oscilloscope or power disturbance analyzer.

Scanner Problems

If using a scanner, check the scan list, data rate, and addresses of devices. Verify series and revision of the scanner is the latest. If the scanner is bus-off, recycle the 24V supply and then reset the scanner.

If the scanner goes bus-off again, the problem is some combination of these issues.

- Defective node device
- Incorrect node data rate
- Bad network topology
- Faulty wiring
- Faulty scanner
- Faulty power supply
- Bad grounding
- Electrical noise

Power Supply Problems

If a single power supply is used, add up the current requirements of all devices drawing power from the network. This total should be considered the minimum current rating in selecting the power supply used.

In addition, check these symptoms.

- Length and current level in trunk and drop cables
- Size and length of the cable supplying power to the trunk
- Voltage measured at the middle and ends of the network
- Noise in network power measured with an oscilloscope

Cable Installation and Design Problems

Cable installation and design refers to the physical layout and connections on the network. Walk the network if possible to determine the actual layout and connections. Network management software displays only a logical record of the network.

Make sure you have a diagram of the physical layout and a record of the information in these tables.

Cable Checks
Number of nodes.
Individual drop lengths.
Branched drop length.
Cumulative drop length.
Total trunk length.
Power supply cable length and gauge.
Terminator locations and size.

Power Checks
Break the earth ground of the V- and Shield and verify >1.0 Mohm to frame ground with power supply off.
Use a multi-meter to check for short circuit between CAN_H and CAN_L, or CAN (H or L) to Shield, V- or V+.
Total power load and at its distribution points.
Spot check power for noise.

Adjusting the Physical Network Configuration

Try these methods to improve the efficiency of your physical network configuration.

- Shortening the overall length of the cable system
- Moving the power supply in the direction of an overloaded cable section
- Moving devices from an overloaded cable section to a less loaded section
- Moving higher current loads closer to the power supply
- Adding another power supply to an overloaded network
- Moving the power supply from the end to the middle of the network

Appendix A

Interconnect Diagrams

Introduction

This appendix provides you with interconnect diagrams for your Ultra3000 servo drive.

Wiring Examples

This appendix provides wiring examples to assist you in wiring the Ultra3000 drive system. The notes in this table apply to the power, shunt, motor, actuator, and control string interconnect diagrams.

ATTENTION The National Electrical Code and local electrical codes take precedence over the values and methods provided. Implementation of these codes are the responsibility of the machine builder.

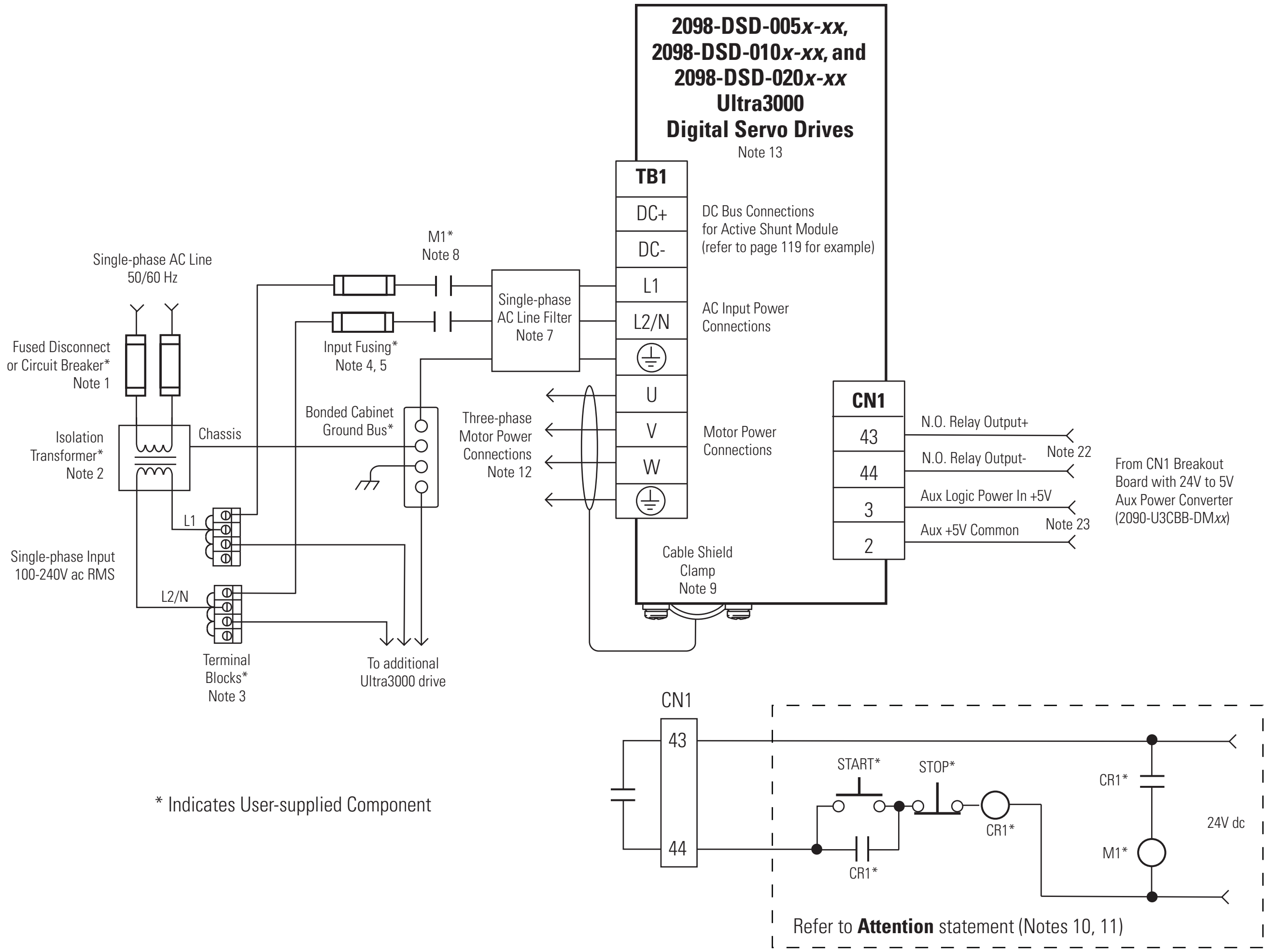
Note	Information
1	A disconnecting device is required for maintenance and safety. If a grounded neutral is used instead of L2, only L1 may be switched or fused.
2	An isolation transformer is optional. If the transformer secondary has a neutral connection, neutral must be bonded to ground. Multiple drive modules may be powered from one transformer or other ac supply source.
3	Do not daisy chain drive module power connections. Make separate connections directly to the ac supply.
4	For power wiring specifications, refer to the Ultra3000 Installation Manual, publication 2098-IN003.
5	For input fuse sizes, refer to the Ultra3000 Installation Manual, publication 2098-IN003.
6	May be used to maintain power to logic section of drive and status indicators when main ac input power is removed. A separate ac line source may be used if voltage is between 88…265V ac (rms) on 2098-DSD-*xxx* (230V drives) or 207…528V ac (rms) on 2098-DSD-HV*xxx* (460V drives). In this configuration, a separate line filter for logic power may be required.
7	Place ac (EMC) line filter as close to the drive as possible and do not route very dirty wires in wireway. If routing in wireway is unavoidable, use shielded cable with shields grounded to the drive chassis and filter case. For ac line filter specifications, refer to the Ultra3000 Installation Manual, publication 2098-IN003.
8	Drive Enable input must be opened before main power is removed and auxiliary power is present, or a drive fault will occur. A delay of at least 1.0 second must be observed before attempting to enable the drive after main power is restored.
9	Cable shield clamp must be used in order to meet CE requirements. No external connection to chassis ground required.
10	**ATTENTION** Implementation of safety circuits and risk assessment is the responsibility of the machine builder. Please reference international standards EN1050 and EN954 estimation and safety performance categories. For more information refer to Understanding the Machinery Directive, publication SHB-900.
11	The recommended minimum wire size for wiring the safety circuit is 1.5 mm^2 (16 AWG).
12	For motor cable specifications and drive/motor cable combinations, refer to the Kinetix Motion Control Selection Guide, publication GMC-SG001.
13	The Ultra3000 drive referenced is either a 2098-DSD-*xxx* or 2098-DSD-*xxx*X (Ultra3000 with indexing), 2098-DSD-*xxx*-SE (SERCOS interface), 2098-DSD-*xxx*-DN (DeviceNet interface), or 2098-DSD-*xxx*X-DN (DeviceNet with indexing) 230V drive.
14	The Ultra3000 drive referenced is either a 2098-DSD-HV*xxx* or 2098-DSD-HV*xxx*X (Ultra3000 with indexing), 2098-DSD-HV*xxx*-SE (SERCOS interface), 2098-DSD-HV*xxx*-DN (DeviceNet interface), or 2098-DSD-HV*xxx*X-DN (DeviceNet with indexing) 460V drive.
15	Wire colors are for flying-lead cable and may vary from premolded cable connectors.
16	Motor power cables (2090-XXNPMF-*xx*S*xx* and 2090-CPBM6DF-16AA*xx*) have a drain wire that must be folded back under the cable shield clamp.
17	Only the MPG-B*xxx* encoder uses the +5V dc supply. MPL-B3*xx*…MPL-B9*xx* and 1326AB (M2L/S2L) encoders use the +9V dc supply.
18	Only the MPL-A5*xx* encoders use the +9V dc supply. MPG-A*xxx* and MPL-A3*xx*, MPL-A4*xx*, MPL-A45*xx* encoders use the +5V dc supply.
19	MPL-A15*xx*, MPL-A2*xx*, MPF-A3*xx*, MPF-A4*xx*, MPF-A45*xx*, MPS-A*xxx*, and MPAS-A*xxx* encoders use the +5V dc supply. MPL-B15*xx*, MPL-B2*xx*, MPF-A5*xx*, MPF-B*xxx*, MPS-B*xxx*, and MPAS-B*xxx* encoders use +9V dc.
20	Brake connector pins for MPF-A/B5*xx* motors are labeled plus (+) and minus (-). All other MP-Series brake connector pins are labeled F and G. Power connector pins on MPF-A/B5*xx* motors are labeled U, V, W, and GND. All other MP-Series power connector pins are labeled A, B, C, and D.
21	Use a flyback diode or MOV for noise suppression of the motor brake coil. Refer to Controlling a Brake Example on page 131.
22	Relay Output (CN1, pins 43 and 44) must be configured as Ready in Ultraware software.
23	The preferred method for supplying auxiliary power is by using the 12-pin or 44-pin drive-mounted breakout board with 24V to 5V auxiliary power converter (2090-U3CBB-DM12 or -DM44). Auxiliary +5V power is required to maintain encoder position when the main ac power is disconnected.

Power Interconnect Diagrams

This is the power wiring diagram with 24V dc control string for 2098-DSD-005*x-xx*, 2098-DSD-010*x-xx*, and 2098-DSD-020*x-xx* Ultra3000 drives (non-SERCOS drives only). To avoid a separate 5V dc auxiliary logic power supply, the 24V to 5V converter breakout board (catalog number 2090-U3CBB-DM*xx*) is used to wire the control interface (CN1) connector. For the control string diagram with 120V ac input refer to the figure on page 128.

For SERCOS drives, input line contactor is part of the PLC program and output control.

Typical Power Wiring of Ultra3000 (230V) System

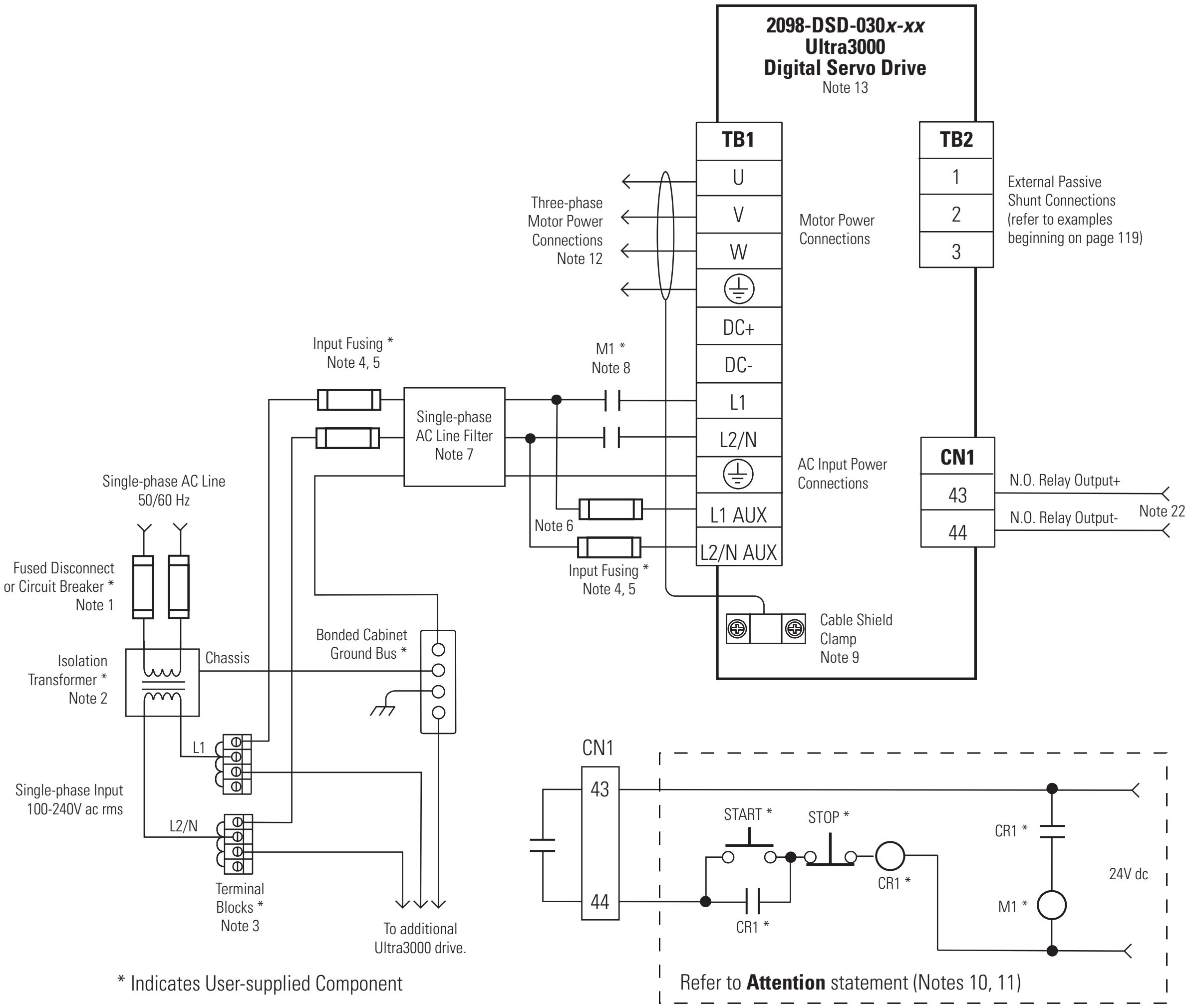


* Indicates User-supplied Component

This is the power wiring diagram with 24V dc control string for the 2098-DSD-030*x-xx* drive (non-SERCOS drives only). For the control string diagram with 120V ac input refer to the figure on page 129.

For SERCOS drives, input line contactor is part of the PLC program and output control.

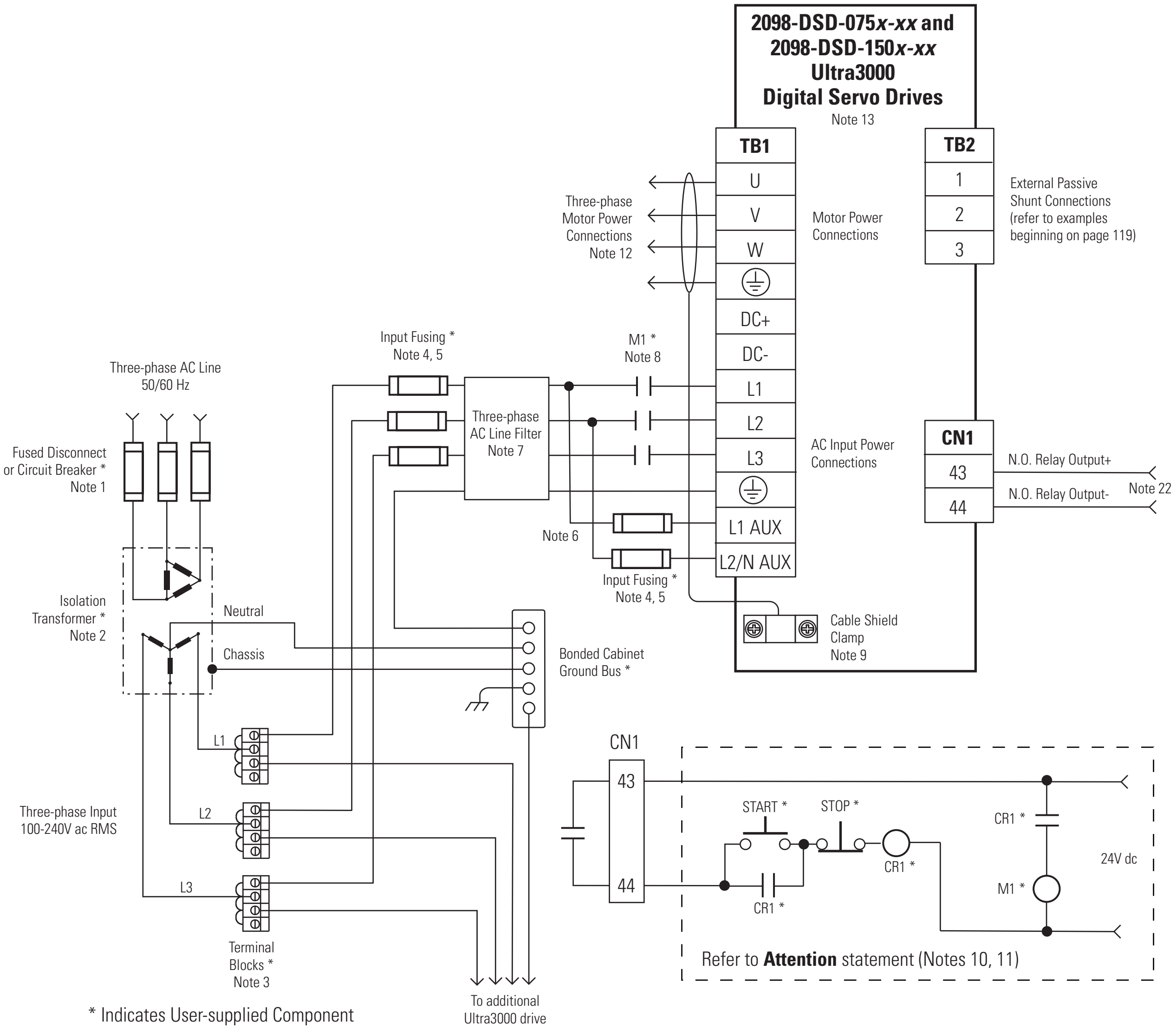
Typical Power Wiring of Ultra3000 (230V) System



This is the power wiring diagram with 24V dc control string for 2098-DSD-075*x-xx* and 2098-DSD-150*x-xx* Ultra3000 drives (non-SERCOS drives only). For the control string diagram with 120V ac input refer to the figure on page 130.

For SERCOS drives, input line contactor is part of the PLC program and output control.

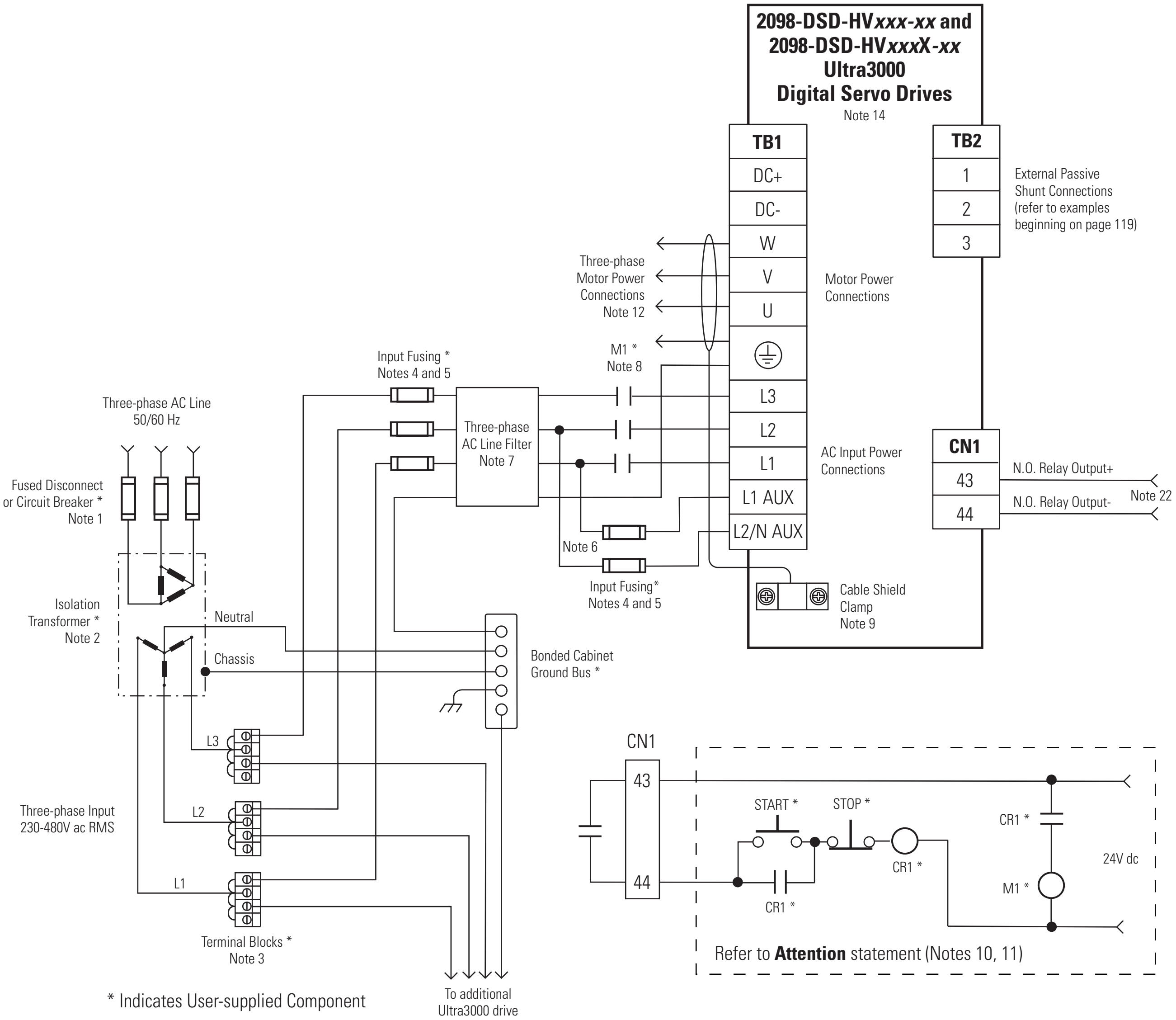
Typical Power Wiring of Ultra3000 (230V) System



* Indicates User-supplied Component

This is the power wiring diagram with 24V dc control string for 2098-DSD-HV*xxx*-*xx* and 2098-DSD-HV*xxx*X-*xx* Ultra3000 drives. For the control string diagram with 120V ac input refer to the figure on page 130.

Typical Power Wiring of Ultra3000 (460V) System

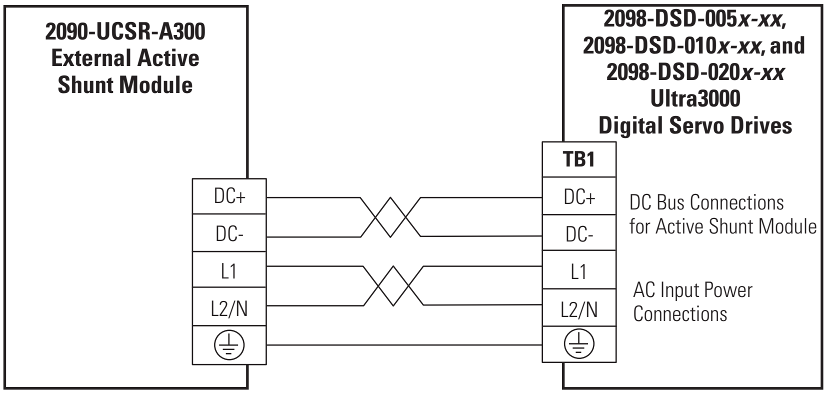


* Indicates User-supplied Component

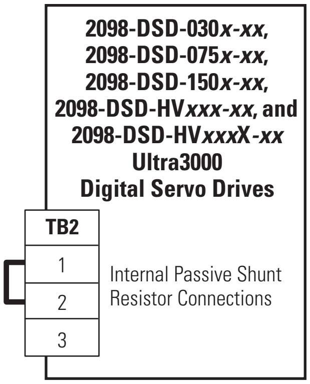
Shunt Module Interconnect Diagrams

This section contains the interconnect diagrams for Ultra3000 drives with active and passive shunt modules.

External Active Shunt Module Interconnect Diagram

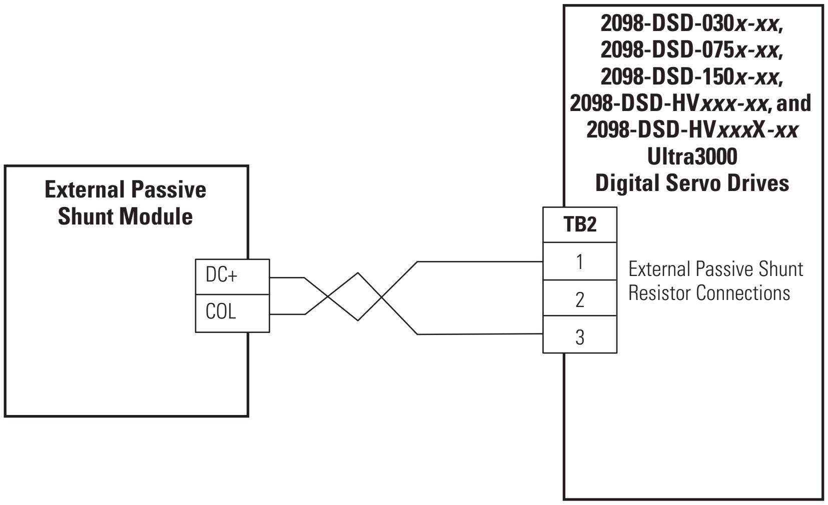


Internal Passive Shunt Interconnect Diagram (default configuration)



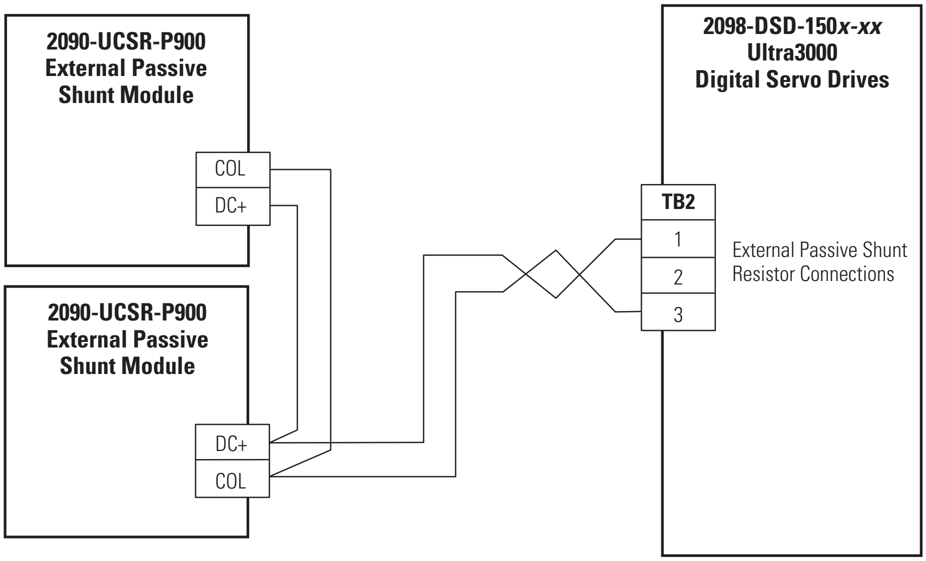
IMPORTANT	Internal shunt operation is only present on the drives listed in the figure above.

External Passive Shunt (single-shunt) Diagram



In this figure, the 2098-DSD-150*x-xx* Ultra3000 drive is wired with two external passive shunt resistors. When two 900 W shunt modules are connected in parallel, the shunt capacity is doubled for a total of 1800 W of continuous power dissipation.

External Passive Shunt (two-shunt) Diagram



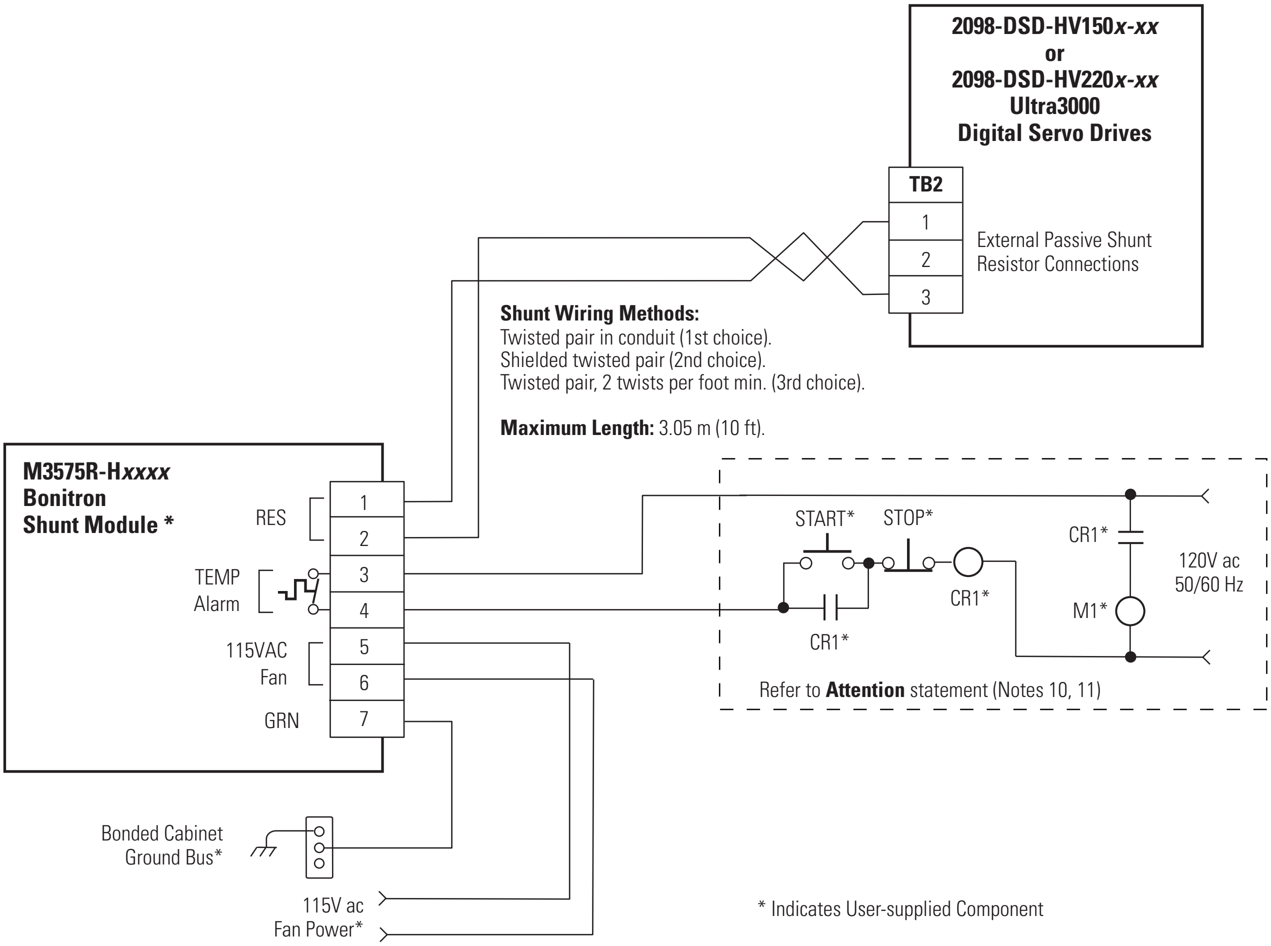
ATTENTION

⚠

To avoid damage to your external shunt module, verify that the proper 230V or 460V fuse is installed prior to applying power. Refer to Passive Shunt Modules Installation Instructions, publication 2090-IN004, for more information.

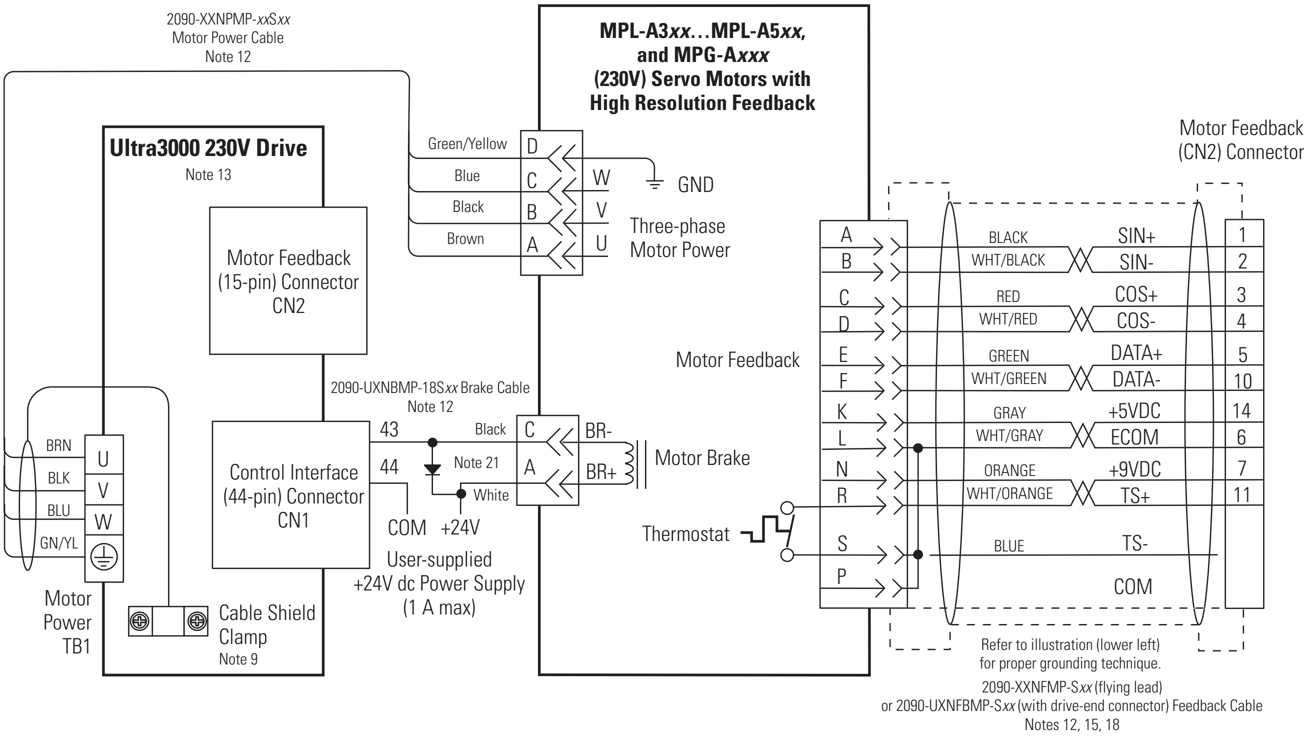
In this figure, the 2098-DSD-HV150*x-xx* or 2098-DSD-HV220*x-xx* Ultra3000 drive is wired to a Bonitron shunt module.

External Passive Shunt (Bonitron shunt) Diagram

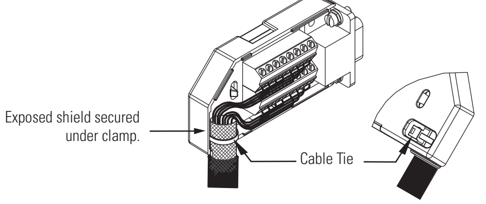


Ultra3000 Drives/Rotary Motors and Actuators Wiring Examples

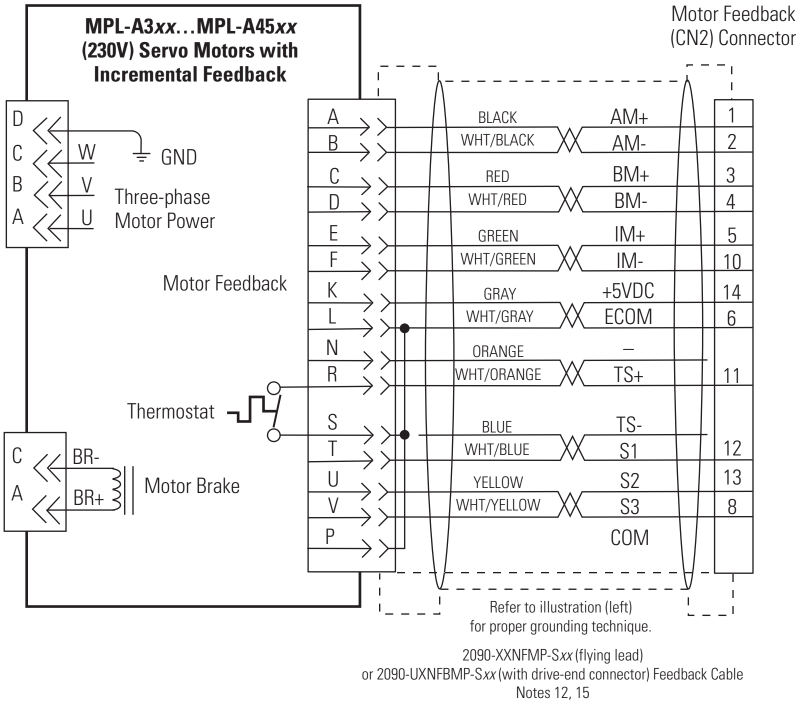
Wiring Examples with MPL-A and MPG-A (MP-Series) Motors/Actuators



Grounding Technique for Feedback Cable Shield

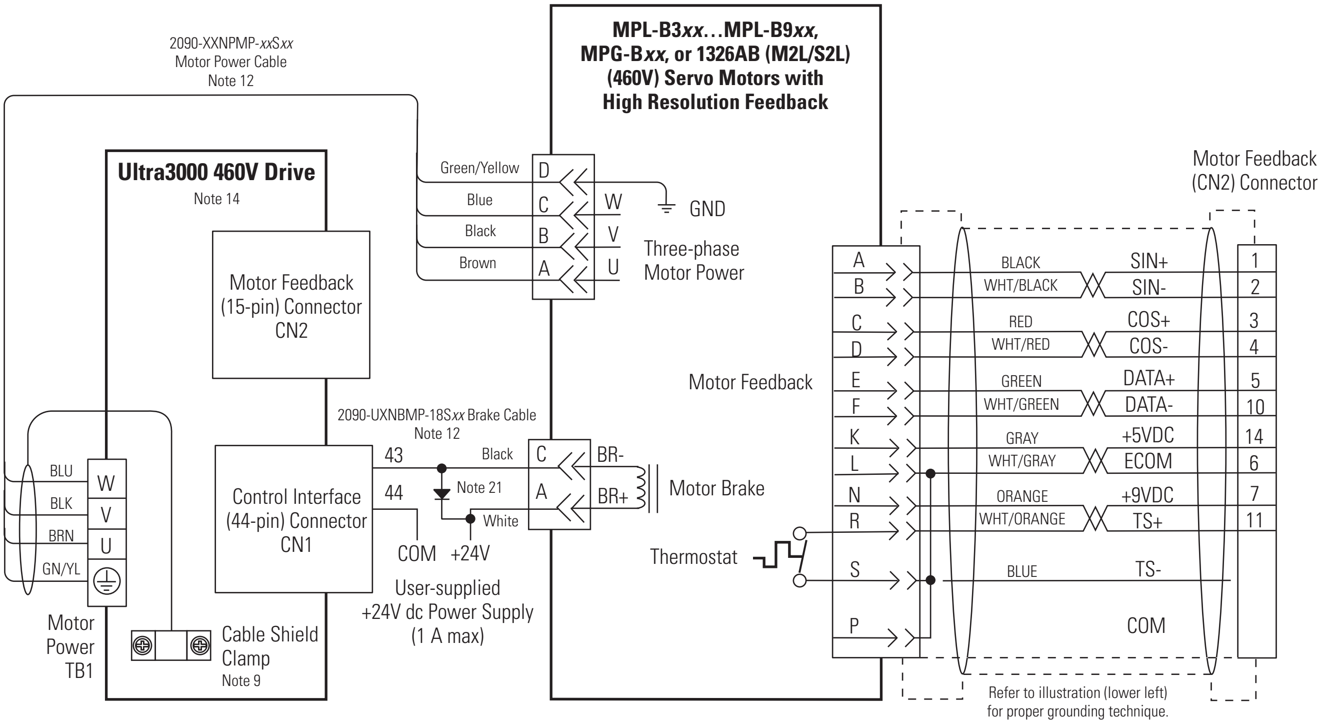


2090-UXBB-DM15
Motor Feedback Breakout Board



IMPORTANT MPF-A5*xxx* motors are not compatible with 2098-DSD-005, 2098-DSD-010, and 2098-DSD-020 Ultra3000 drives.

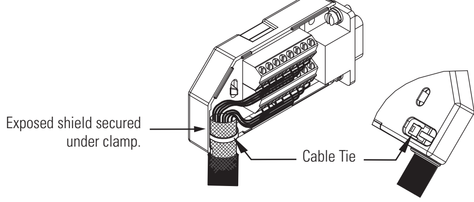
Wiring Examples with MPL-B, MPG-B, and 1326AB Motors/Actuators



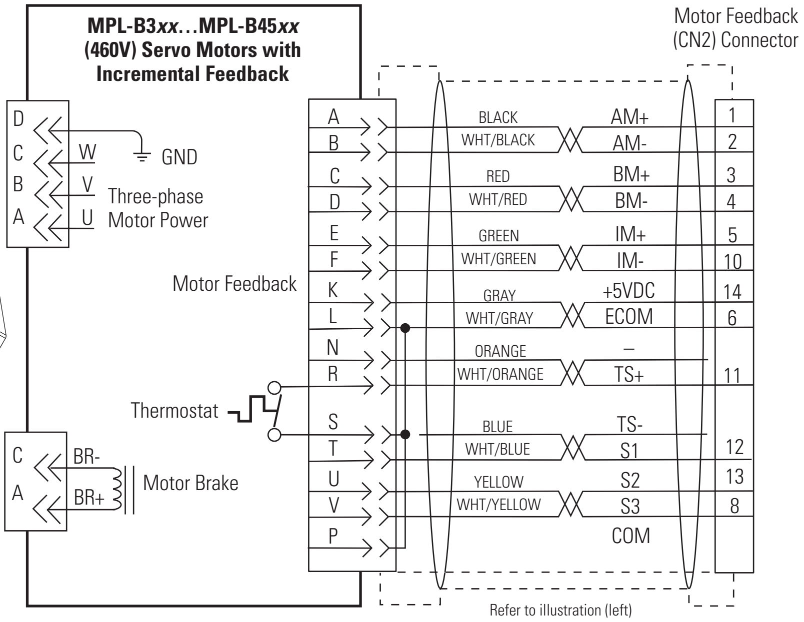
Refer to illustration (lower left) for proper grounding technique.

2090-XXNFMP-S*xx* (flying lead)
or 2090-UXNFBMP-S*xx* (with drive-end connector) Feedback Cable
Notes 12, 15, 17

Grounding Technique for Feedback Cable Shield



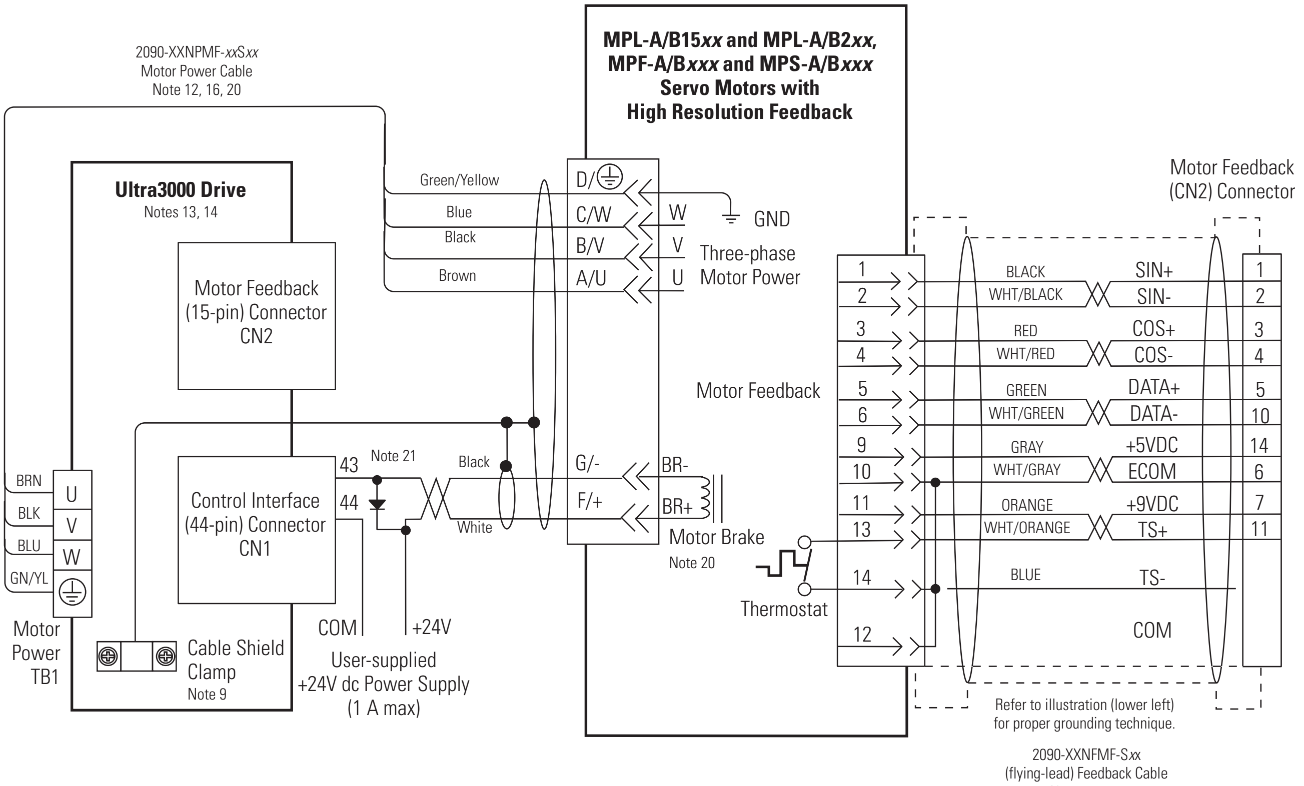
2090-UXBB-DM15
Motor Feedback Breakout Board



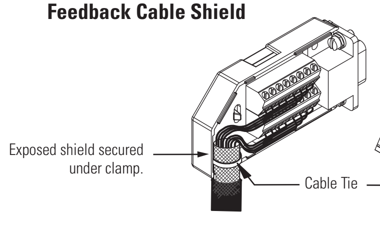
Refer to illustration (left) for proper grounding technique.

2090-XXNFMP-S*xx* (flying lead)
or 2090-UXNFBMP-S*xx* (with drive-end connector) Feedback Cable
Notes 12, 15

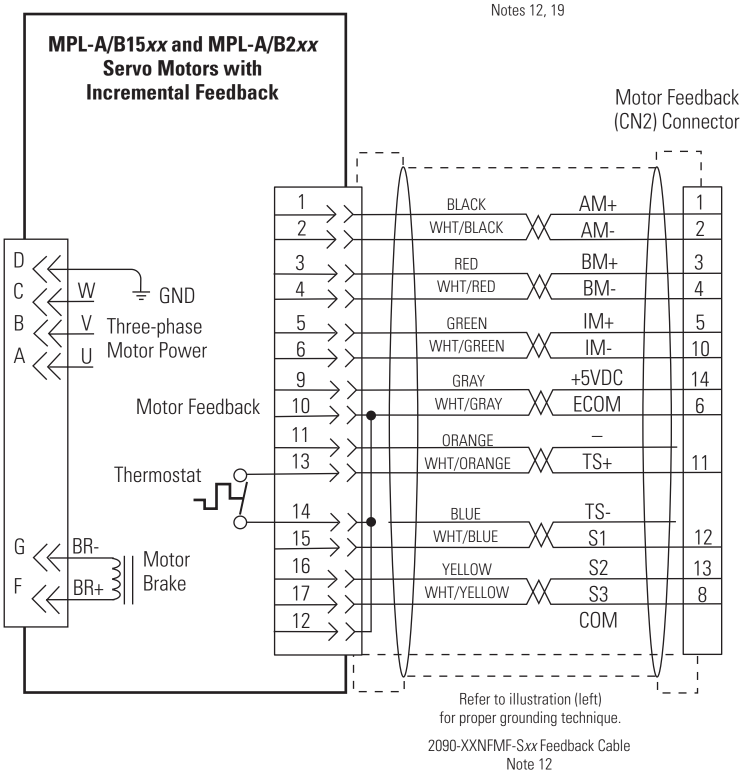
Wiring Examples with MPL-A/B, MPF-A/B, and MPS-A/B (MP-Series) Motors



Grounding Technique for Feedback Cable Shield

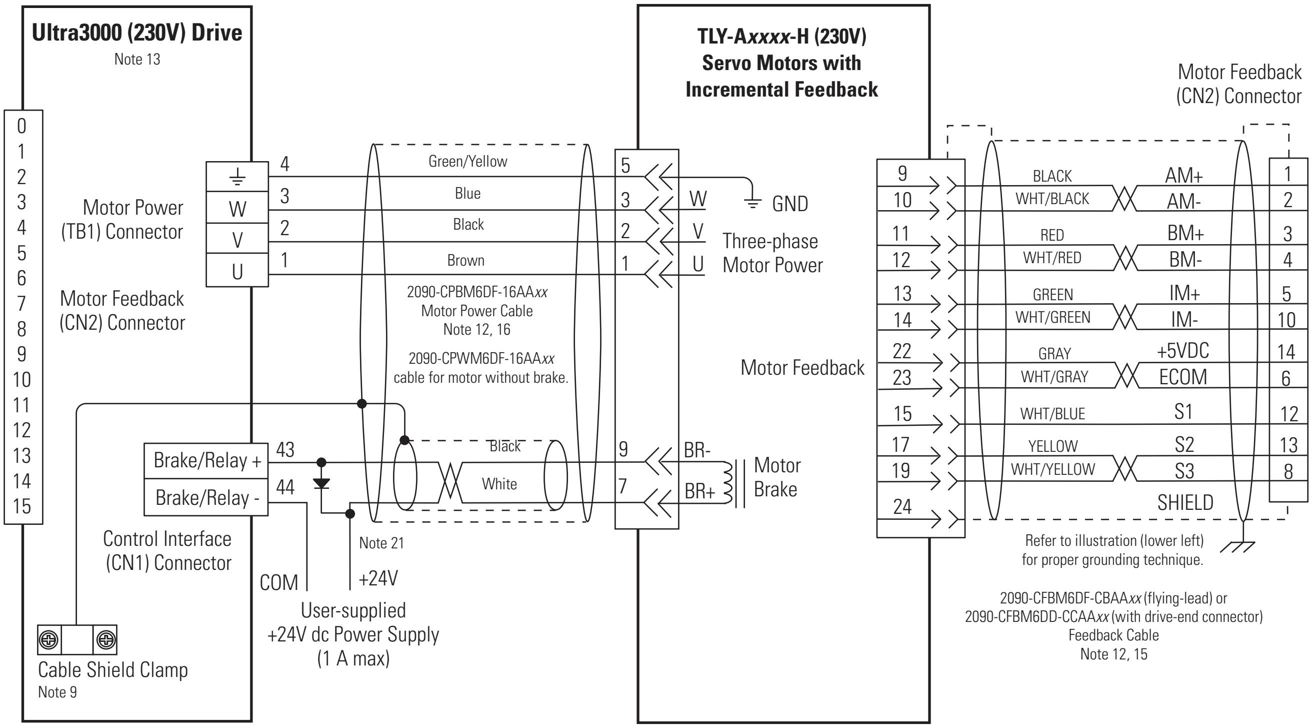


2090-UXBB-DM15
Motor Feedback Breakout Board

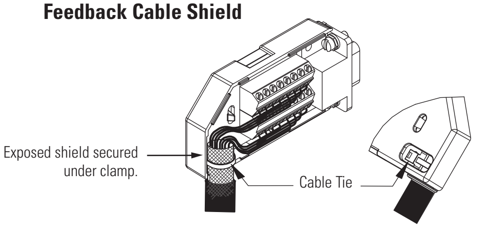


IMPORTANT MPF-A5*xxx* motors are not compatible with 2098-DSD-005, 2098-DSD-010, and 2098-DSD-020 Ultra3000 drives.

Wiring Example with TLY-A (TL-Series) Motors

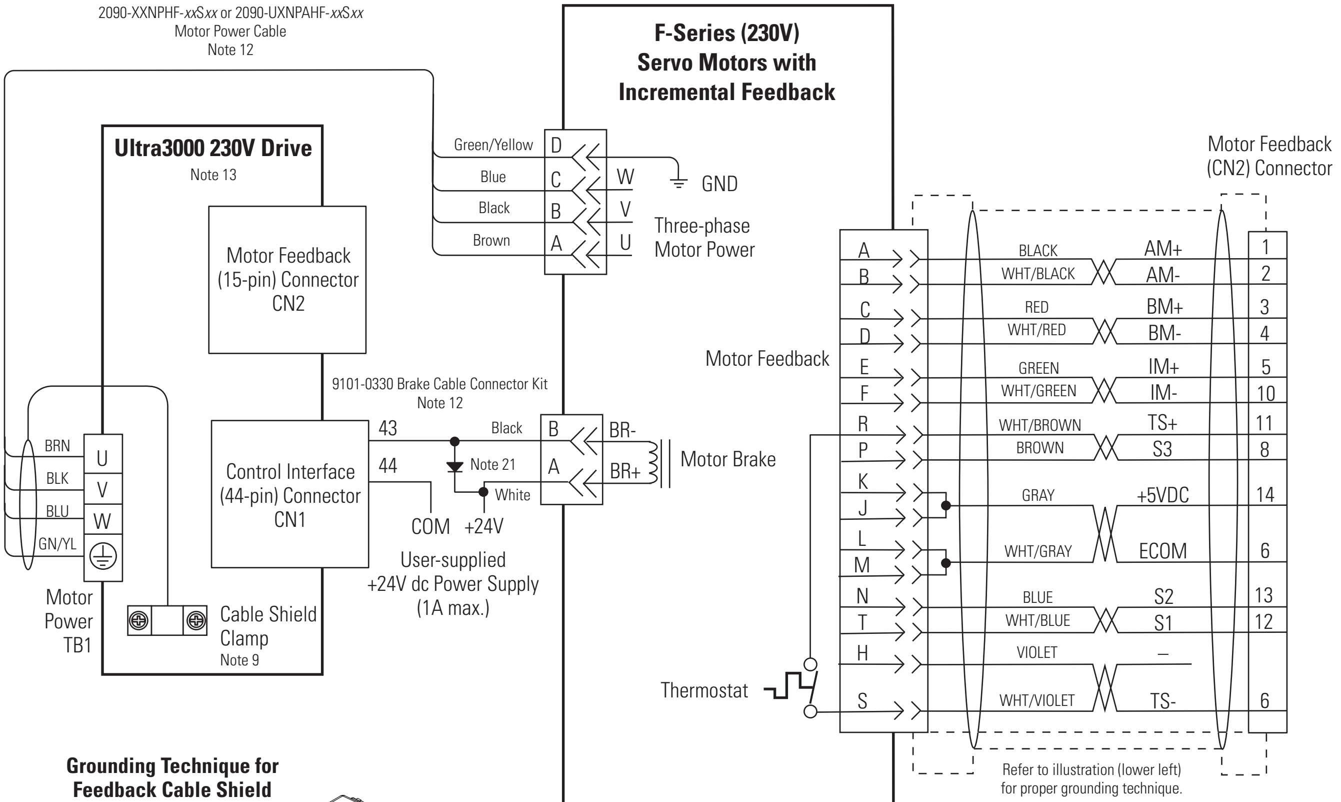


Grounding Technique for Feedback Cable Shield



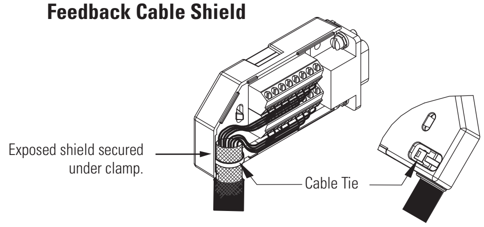
2090-UXBB-DM15
Motor Feedback Breakout Board

Wiring Examples with F-Series (230V) Motors



2090-XXNFHF-S*xx* (flying lead)
or 2090-UXNFBHF-S*xx* (with drive-end connector)
Feedback Cable
Notes 12, 15

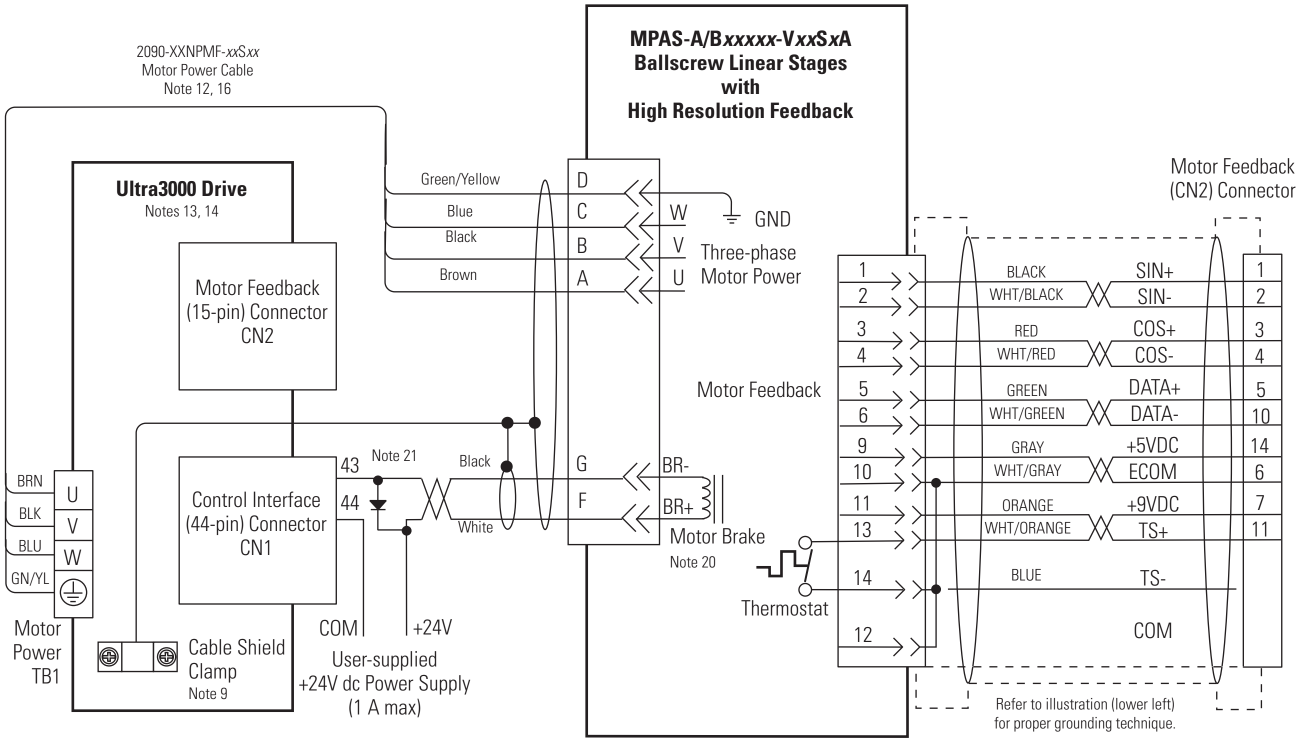
Grounding Technique for Feedback Cable Shield



2090-UXBB-DM15
Motor Feedback Breakout Board

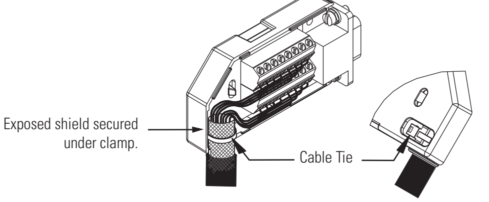
Ultra3000 Drives/Linear Actuators Wiring Examples

Wiring Example with MPAS-A/B (MP-Series) Linear Stages

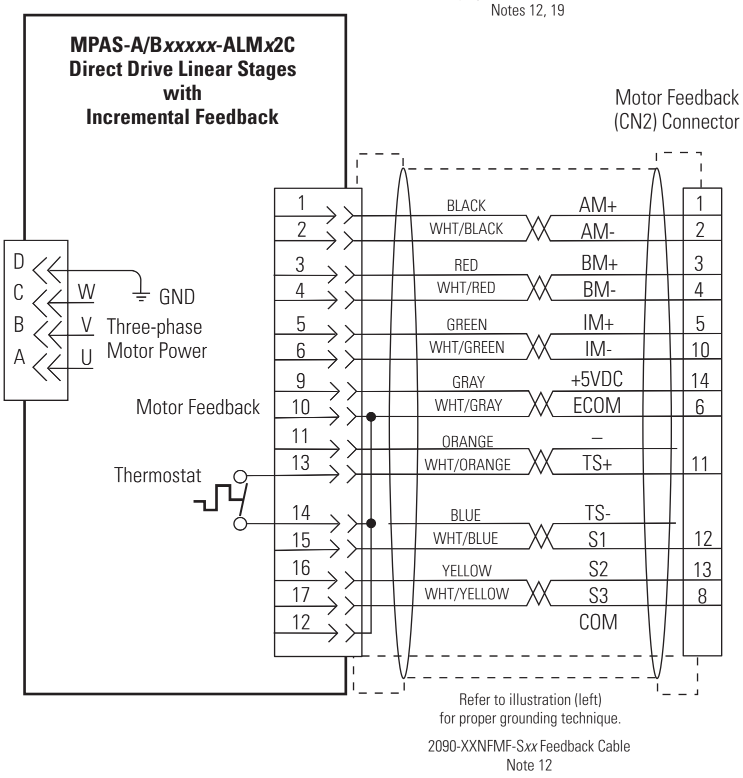


2090-XXNFMF-Sxx
(flying-lead) Feedback Cable
Notes 12, 19

Grounding Technique for Feedback Cable Shield



2090-UXBB-DM15
Motor Feedback Breakout Board



2090-XXNFMF-Sxx Feedback Cable
Note 12

Control String Examples (120V ac)

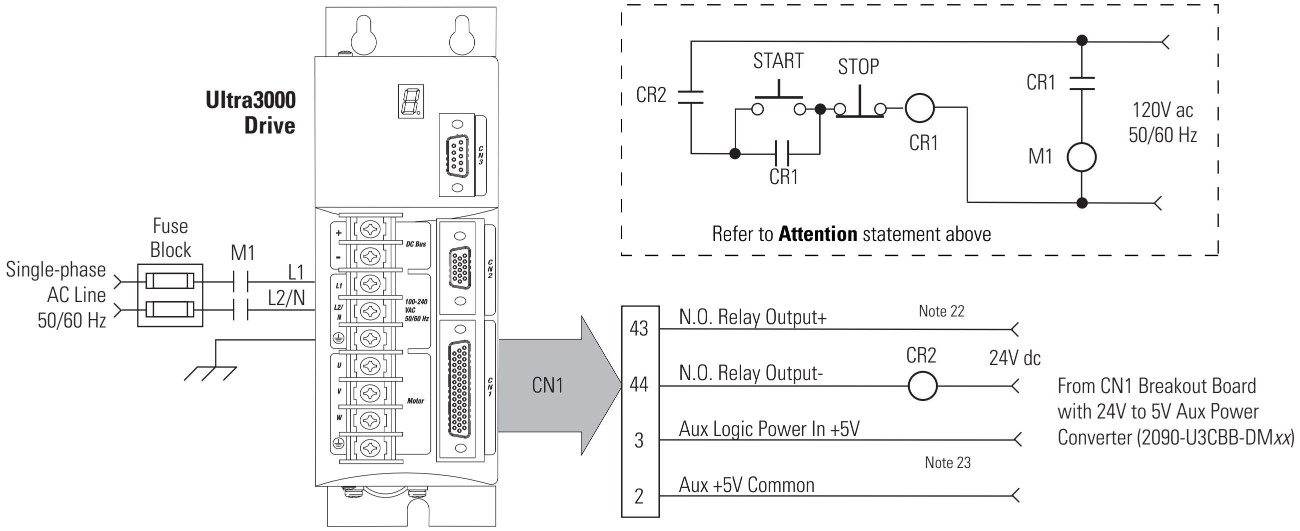
This section provides information to assist you in using the configurable Drive Ready output in a control string with 120V ac input voltage.

In this example, the 2098-DSD-005*x-xx*, 2098-DSD-010*x-xx*, or 2098-DSD-020*x-xx* Ultra3000 drive is wired to the 120V ac control string.

ATTENTION

Implementation of safety circuits and risk assessment is the responsibility of the machine builder. Please reference international standards EN1050 and EN954 estimation and safety performance categories. For more information refer to Understanding the Machinery Directive, publication SHB-900.

120V ac Single-phase Control String Example

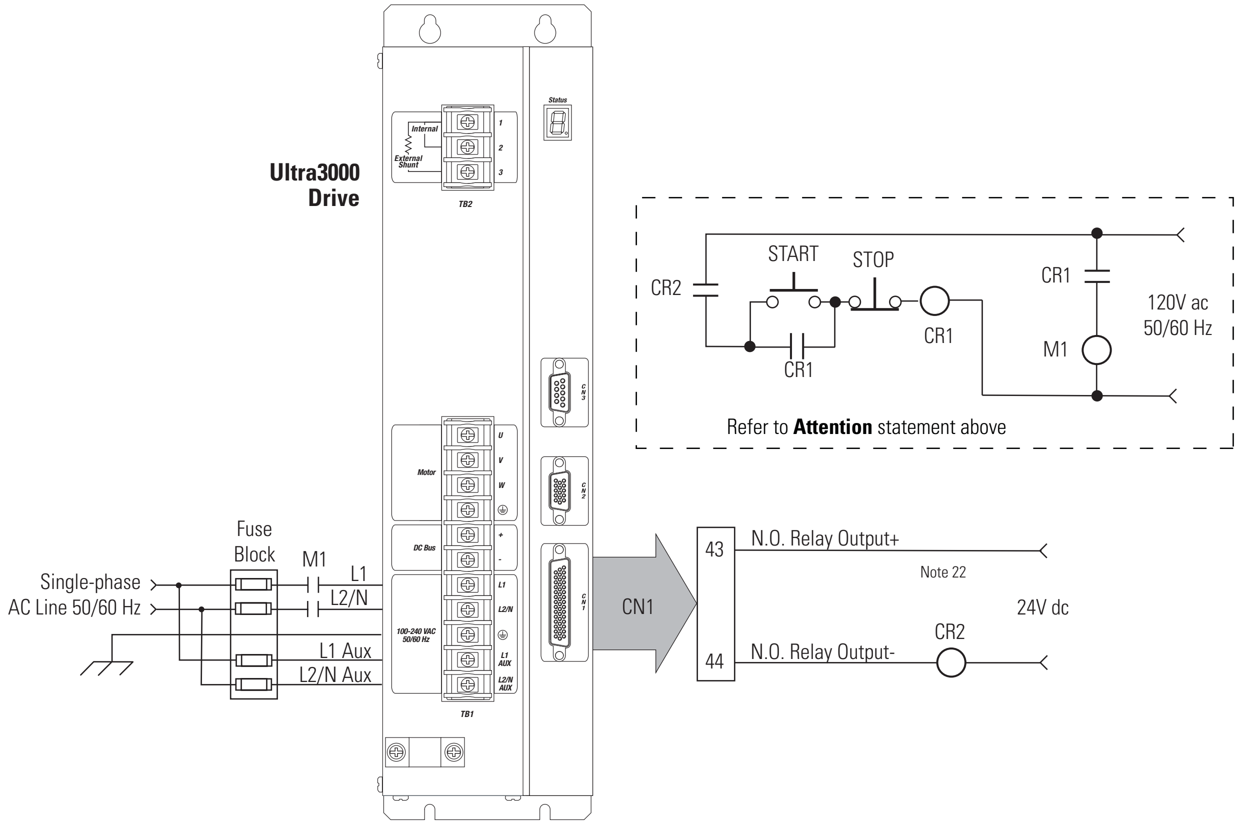


In this example, the 2098-DSD-030*x-xx* Ultra3000 drive is wired to the 120V ac control string.

ATTENTION

Implementation of safety circuits and risk assessment is the responsibility of the machine builder. Please reference international standards EN1050 and EN954 estimation and safety performance categories. For more information refer to Understanding the Machinery Directive, publication SHB-900.

120V ac Single-phase Control String Example

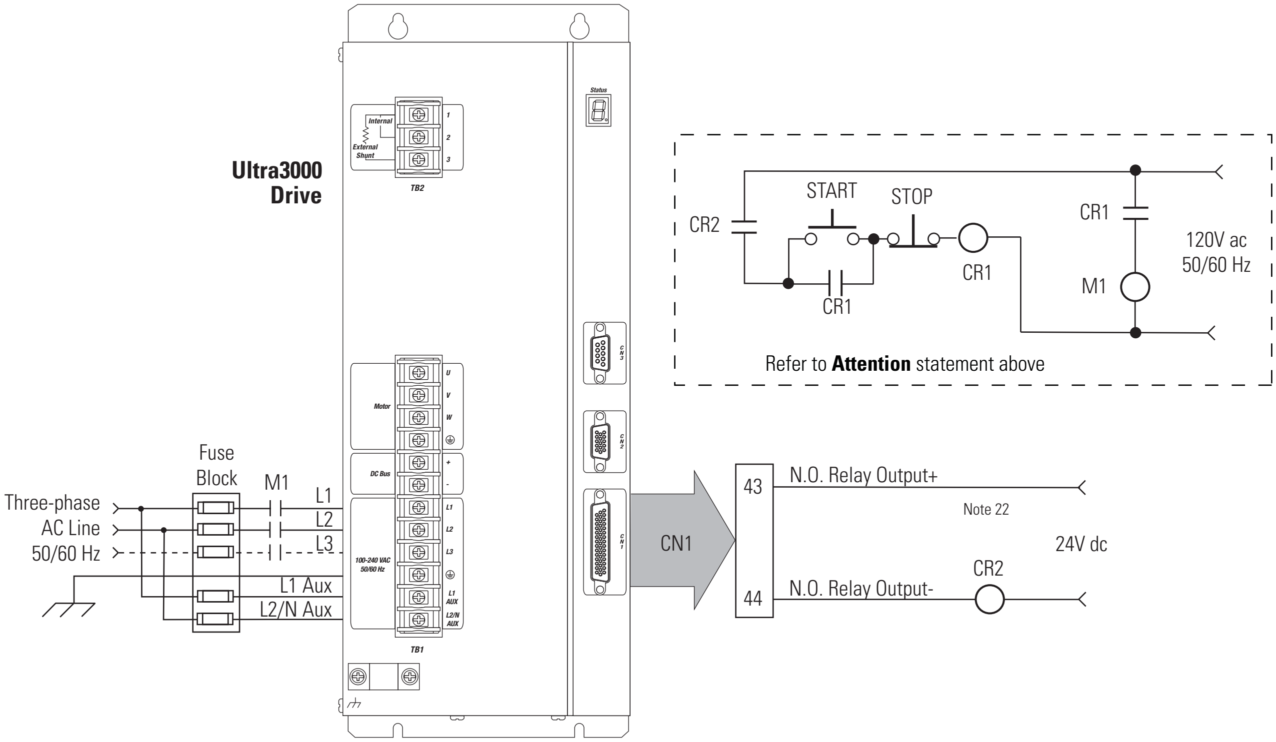


In this example, the 2098-DSD-075*x-xx*, 2098-DSD-150*x-xx*, 2098-DSD-HV*xxx-xx*, and 2098-DSD-HV*xxx*X-*xx* Ultra3000 drive is wired to the 120V ac control string.

ATTENTION

Implementation of safety circuits and risk assessment is the responsibility of the machine builder. Please reference international standards EN1050 and EN954 estimation and safety performance categories. For more information refer to Understanding the Machinery Directive, publication SHB-900.

120V ac Three-phase Control String Example



Refer to **Attention** statement above

Controlling a Brake Example

The relay output of the Ultra3000 drive (pins CN1-43 and CN1-44) is suitable for directly controlling a motor brake, subject to the relay voltage limit of 30V dc, and the relay current limit of 1 A dc. For brake requirements outside of these limits, an external relay must be used. If a transistor output is used, a control relay is also required.

Coil Currents Rated at < 1.0 A

Compatible Brake Motors [1]	Coil Current
MPL-*x*1510, MPL-*x*1520, MPL-*x*1530	0.43…0.53 A
MPL-*x*210, MPL-*x*220, MPL-*x*230	0.46…0.56 A
MPL/MPF-*x*310, MPL/MPF-*x*320, MPL/MPF-*x*330	0.45…0.55 A
MPS-*x*330	
MPL-*x*420, MPL-*x*430, MPL-*x*4520, MPL-*x*4530, MPL-*x*4540, MPL-B4560	0.576…0.704 A
MPF-*x*430, MPF-*x*4530, MPF-*x*4540	
MPS-*x*4540	
1326AB-B4*xxx*	0.88 A
F-4030, F-4050, and F-4075	0.69 A

[1] Use of the variable *x* indicates this specification applies to 230V and 460V motors.

Compatible Brake Motors/Actuators	Coil Current
MPG-*x*004	0.30…0.36 A
MPG-*x*010	0.41…0.49 A
MPG-*x*025	
MPG-*x*050	0.45…0.55 A
MPG-*x*110	0.90…1.10 A
TLY-A110-H, TLY-A120-H, and TLY-A130-H	0.18…0.22 A
TLY-A220-H and TLY-A230-H	0.333…0.407 A
TLY-A2530-H and TLY-A2540-H	0.351…0.429 A
TLY-A410-H	0.648…0.792 A

Coil Currents Rated at >1.0 A and ≤1.3 A

Compatible Brake Motors	Coil Current
MPL-*x*520, MPL-*x*540, MPL-*x*560, MPL-*x*580 [1]	1.05…1.28 A
MPF-*x*540 [1]	
MPS-B560	

[1] Use of the variable *x* indicates this specification applies to 230V and 460V motors.

Compatible Brake Motors	Coil Current
F-6100, F-6200, and F-6300	1.30 A
1326AB-B5*xxx* and 1326AB-B7*xxx*	1.20 A

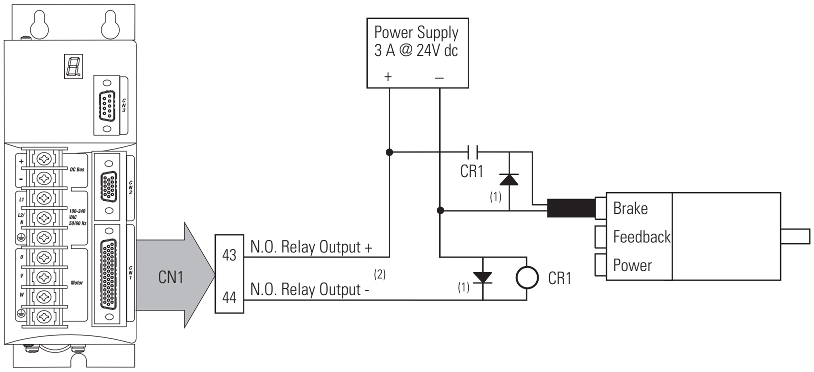
Coil Currents Rated at >1.3 A and ≤3.0 A

Compatible Brake Motors	Coil Current
MPL-B640, MPL-B660, and MPL-B680	1.91…2.19 A
MPL-B860 and MPL-B880	
MPL-B960 and MPL-B980	

A separate power source is required to disengage the brake. Removing power causes the brake to engage, but may also cause electrical arcing to occur at the relay contacts until the brake power dissipates. A customer-supplied diode or metal oxide varistor (MOV) is recommended to prevent arcing. Use of an MOV can also reduce the time to mechanically engage the brake.

IMPORTANT Electrical arcing may occur at the relay contacts until the brake power dissipates. A customer-supplied diode or metal oxide varistor (MOV) is recommended to prevent arcing.

Example Configuration Controlling a Motor Brake



(1) Flyback diode (1N4004 rated 1.0 A @ 400V dc) suppresses collapsing field of brake coil.

(2) For non-SERCOS drive, the relay output (CN1-43 and CN1-44) must be configured as a brake.

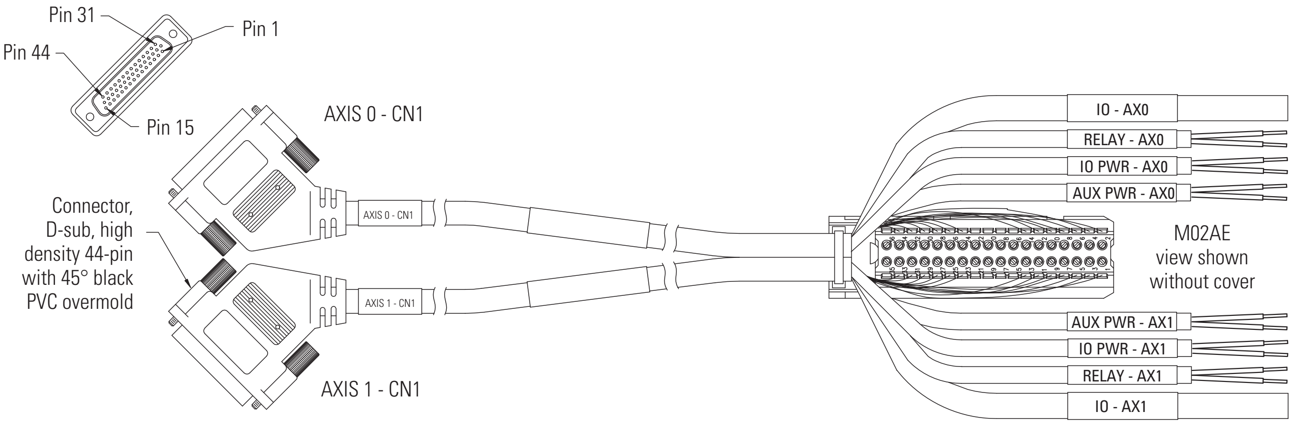
For more information on minimizing electrical noise, refer to the System Design for Control of Electrical Noise Reference Manual, publication GMC-RM001.

Ultra3000 Drive to Logix Analog Module Diagrams

This section provides information to assist you in wiring the Ultra3000 drive CN1 (44-pin) cable connector with either the ControlLogix 1756-M02AE servo module or SoftLogix 1784-PM02AE motion card.

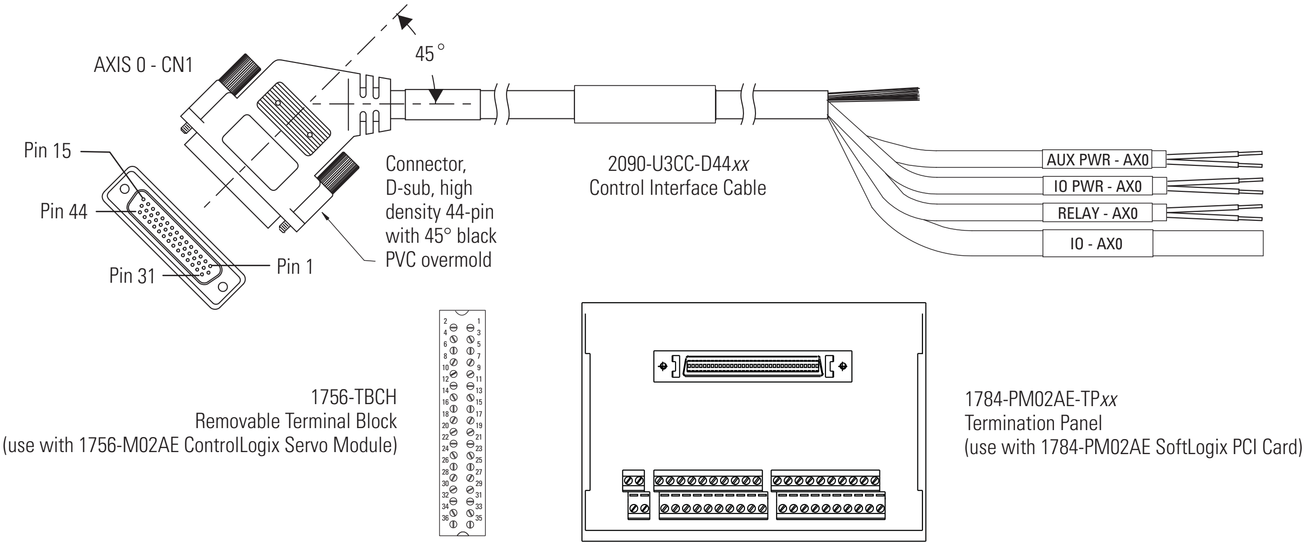
Use the 2090-U3AE-D44*xx* control interface cable (shown below) when connecting two Ultra3000 drives to the 1756-M02AE servo module. This cable includes the 1756-TBCH pre-wired terminal block. Refer to for the interconnect diagram.

2090-U3AE-D44*xx* Two Axis Cable

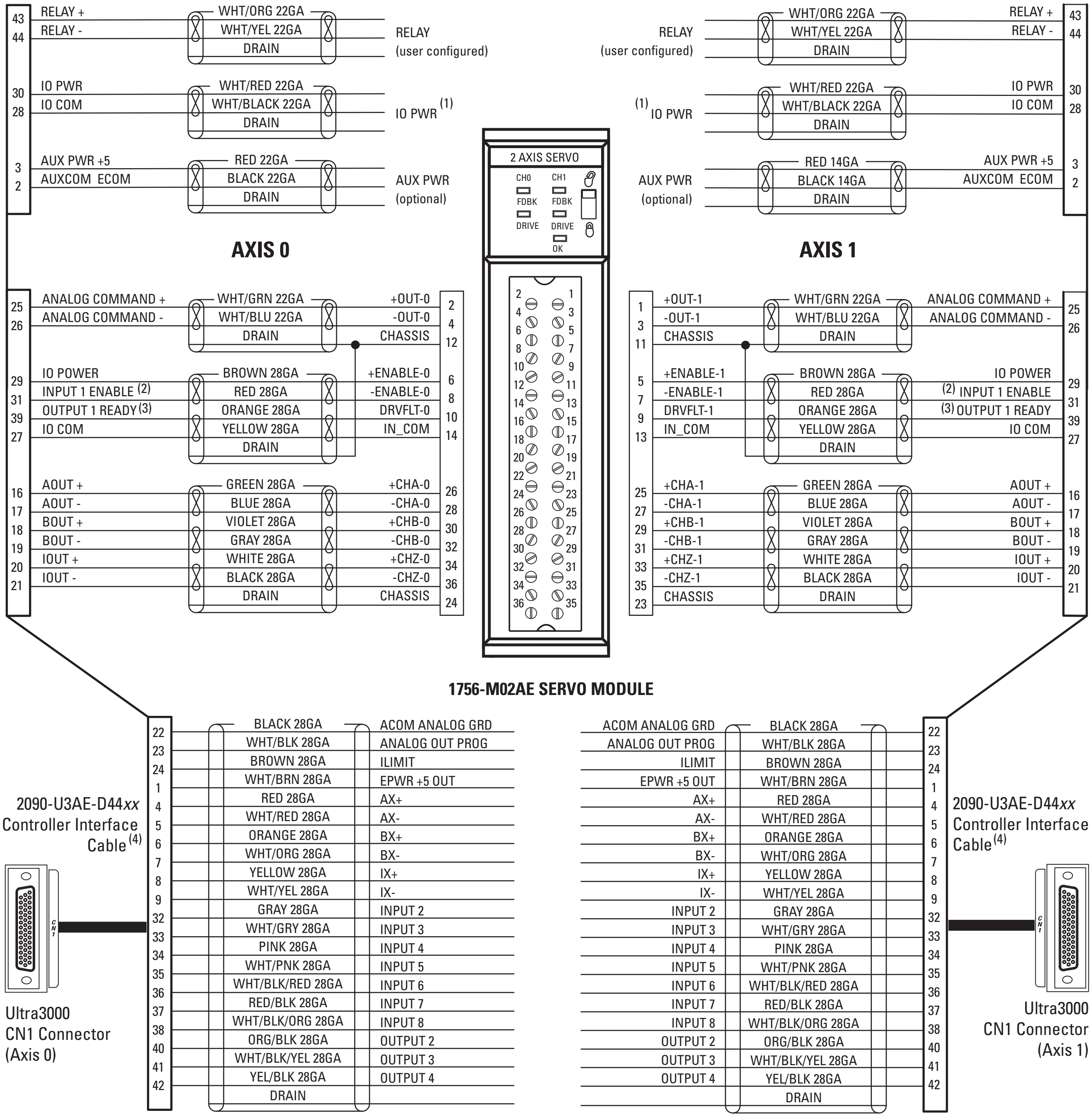


Use the 2090-U3CC-D44*xx* control interface cable (shown below) when connecting a single Ultra3000 drive to either the 1756-M02AE (ControlLogix) servo module or 1784-PM02AE (SoftLogix) PCI card. The 1756-TBCH removable terminal block is required when wiring to the ControlLogix module. The 1784-PM02AE-TP*xx* termination panel is required when wiring to the SoftLogix PCI Card. Refer to for the interconnect diagram.

Control Interface Cable and Terminations



Ultra3000 Drive to ControlLogix Servo Module Interconnect Diagram



---

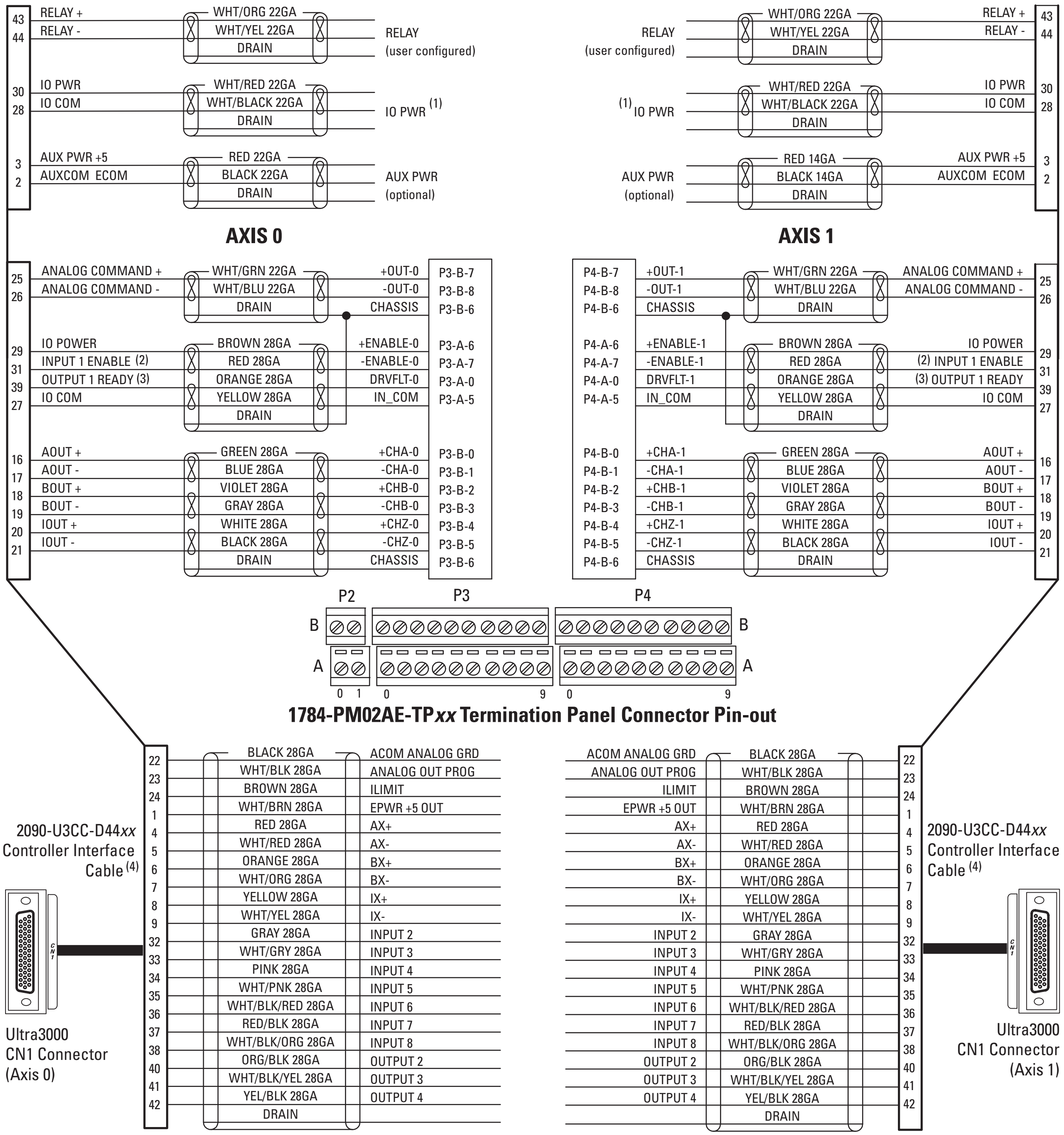
(1) I/O power (pins 28 and 30) must be connected to user-supplied 12…24V dc.

(2) Input 1 (pin 31) must configured as Drive Enable using Ultraware software.

(3) Output 1 (pin 39) must be configured as Ready using Ultraware software.

(4) This cable does not carry the unbuffered motor encoder signals (CN1 pins 10…15). Contact your Allen-Bradley sales representative if these signals are required for your application.

Ultra3000 Drive to SoftLogix PCI Card Interconnect Diagram



---

(1) I/O power (pins 28 and 30) must be connected to user-supplied 12…24V dc.

(2) Input 1 (pin 31) must configured as Drive Enable using Ultraware software.

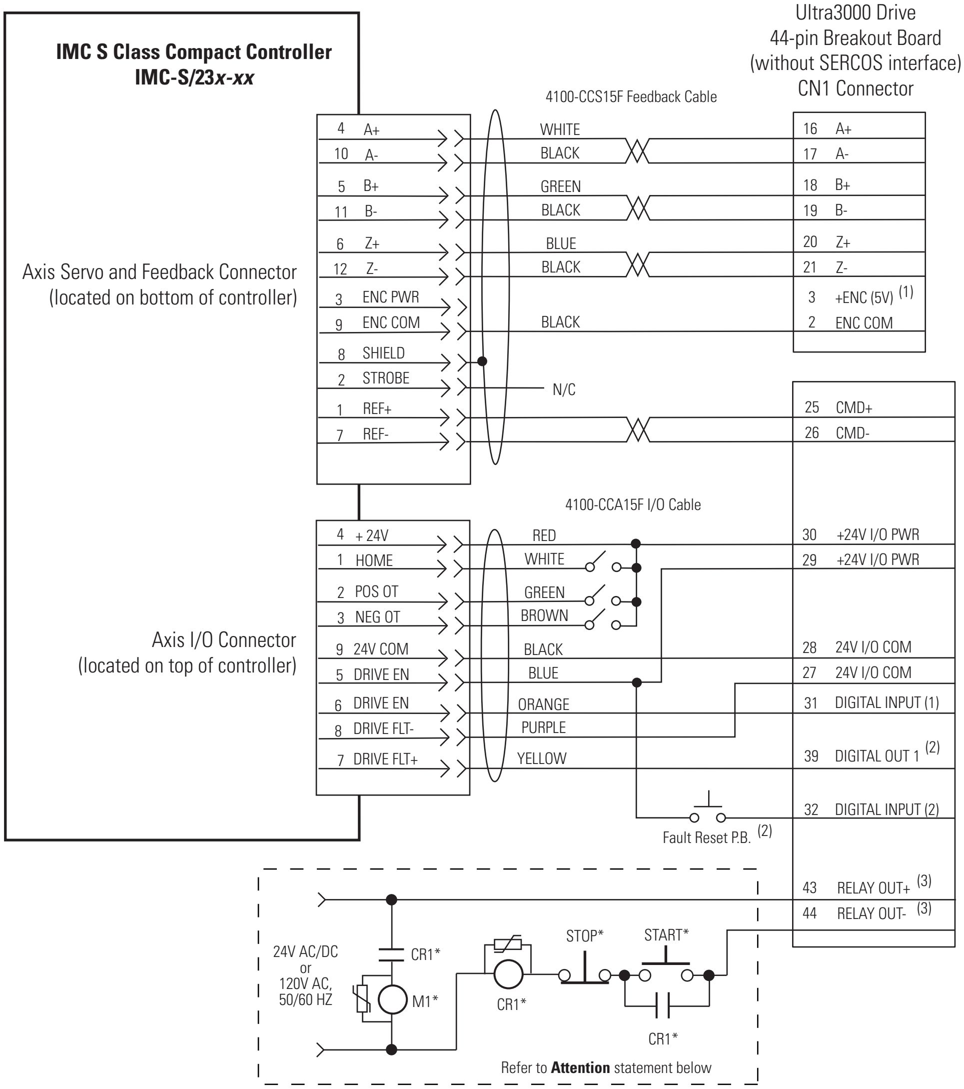
(3) Output 1 (pin 39) must be configured as Ready using Ultraware software.

(4) This cable does not carry the unbuffered motor encoder signals (CN1 pins 10…15). Contact your Allen-Bradley sales representative if these signals are required for your application.

Ultra3000 Drive to IMC-S Compact Controller Diagram

This section provides information to assist you in wiring the IMC-S/23*x*-*xx* Compact Controller when connecting the 4100-CCS15F feedback cable and 4100-CCA15F I/O cable to your Ultra3000 drive.

Ultra3000 Drive to IMC-S/23*x*-*xx* Compact Controller Configuration



---

(1) The preferred method for supplying the auxiliary +5V is by using the 12- or 44-pin drive mounted breakout board with 24V to 5V auxiliary power converter (catalog number 2090-U3CBB-DM12 or -DM44). Auxiliary +5V power is required to maintain encoder position with an external position controller during a controlled stop condition.

(2) Drive Enable and Fault Reset are configured in Ultraware software.

(3) Relay Output (CN1, pins 43 and 44) must be configured as Ready in Ultraware software.

ATTENTION ⚠

Implementation of safety circuits and risk assessment is the responsibility of the machine builder. Please reference international standards EN1050 and EN954 estimation and safety performance categories. For more information refer to Understanding the Machinery Directive, publication SHB-900.

Understanding Motor Feedback Signals and Outputs

Introduction

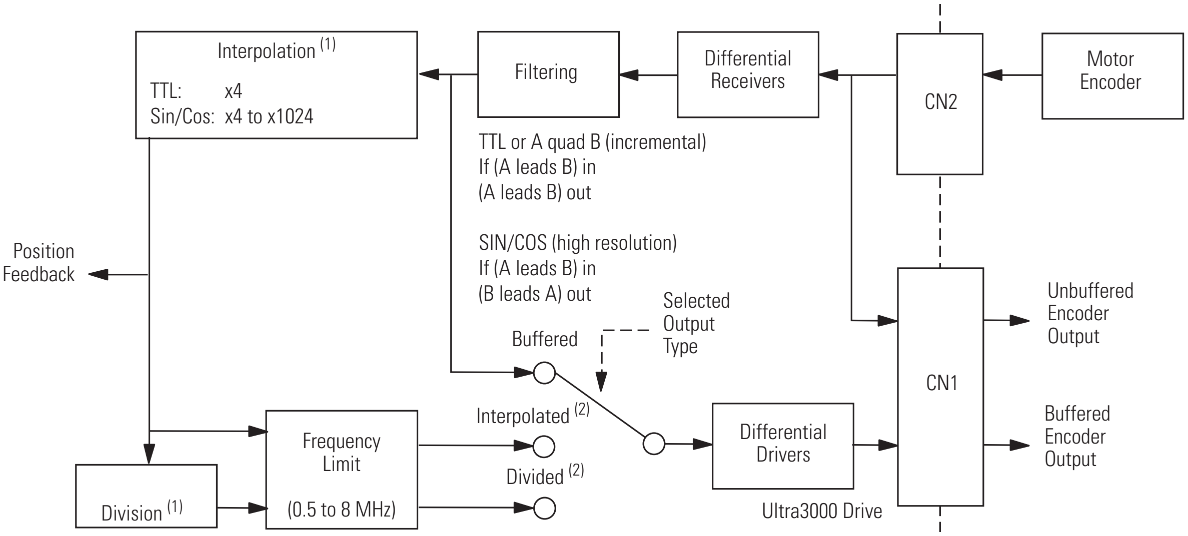
This appendix provides you with motor encoder input signal information and drive encoder output information specific to the Ultra3000 servo drives.

Topic	Page
Introduction	137
Unbuffered Encoder Outputs	138
Incremental Encoder Outputs	138
High Resolution Encoder Outputs	139

The Ultra3000 drive is compatible with motors equipped with both incremental A quad B or high resolution (Stegmann Hiperface) SIN/COS encoders.

The buffered motor encoder outputs use RS-485 differential drivers and have a maximum signal frequency of 2.5 MHz. The drivers can drive a 2V differential voltage into a 100 ohm load. Use the block diagram below to follow the motor encoder input through CN2 to the buffered and unbuffered outputs on CN1.

Motor Encoder Outputs



[1] Interpolation and division operations are performed in firmware and the resulting output frequency is updated at 250 μs intervals.

[2] Interpolated and divided output not available on Ultra3000 SERCOS drives.

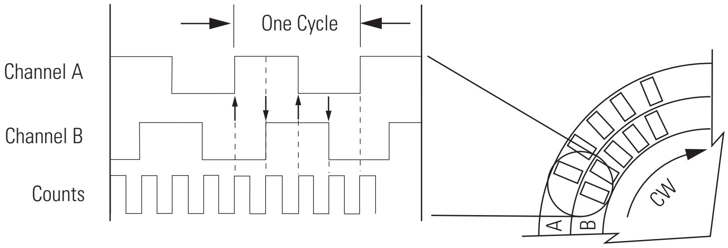
Unbuffered Encoder Outputs

The unbuffered outputs available from the drive (CN1-10 through CN1-15) are tied directly to the incoming (incremental or high resolution) encoder signals (CN2-1 through CN2-6). The unbuffered outputs are not filtered or conditioned.

Incremental Encoder Outputs

Incremental encoder counts are generated in the drive by counting the (high to low and low to high) transitions of the incoming A and B encoder signals. In the channel A signal has two transitions, as does the channel B signal, which results in x4 interpolation (4 transitions/ line equals 4 counts/line). For example, typical 2000 line/rev encoder output becomes 8000 counts/rev in the drive. Counts are not directly available at the encoder outputs, only the A quad B representation.

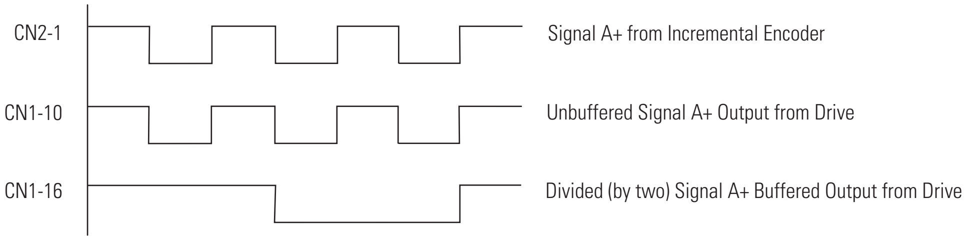
Incremental Encoder Counts



The incremental buffered outputs (listed below) are available from the drive (CN1-16 through CN1-21) and software selectable.

- **Buffered Outputs** are a filtered representation of the original incoming encoder (CN2) signals. Buffered outputs have the same number of cycles/rev as found on CN2.
- **Interpolated Outputs** are the same as buffered outputs when using an incremental encoder. The only interpolation performed on an A quad B signal is the drive's internal counting of transitions (4 counts/line). Because counts are not available outside the drive, selecting this in software is the same as selecting buffered (as described above).
- **Divided Outputs** are the same as buffered outputs, except when divided is selected in the software, the lines/rev are then reduced by the value of the divisor chosen in the software (as shown in the figure below).

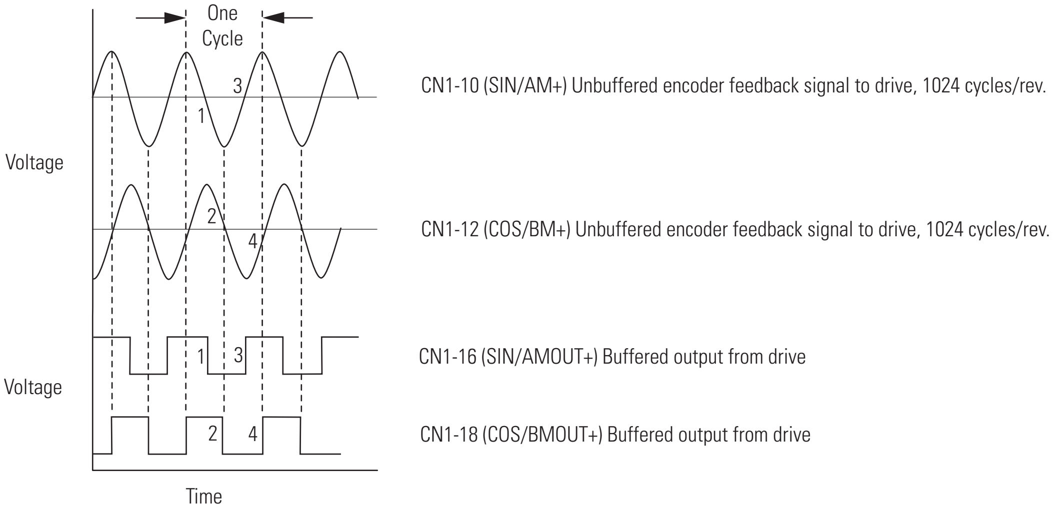
Incremental Encoder Divided



High Resolution Encoder Outputs

When the incoming encoder feedback on CN2 is a high resolution (SIN/COS) signal, the drive is capable of generating more than just 4 counts/cycle (as with incremental encoders). The Ultra3000 drive is capable of breaking the SIN/COS encoder signals into as many as 1024 counts/cycle. For example, a 1024 cycle/rev SIN/COS encoder can result in 1024 x 1024 (high resolution) counts/rev.

Absolute High Resolution Encoder Signals

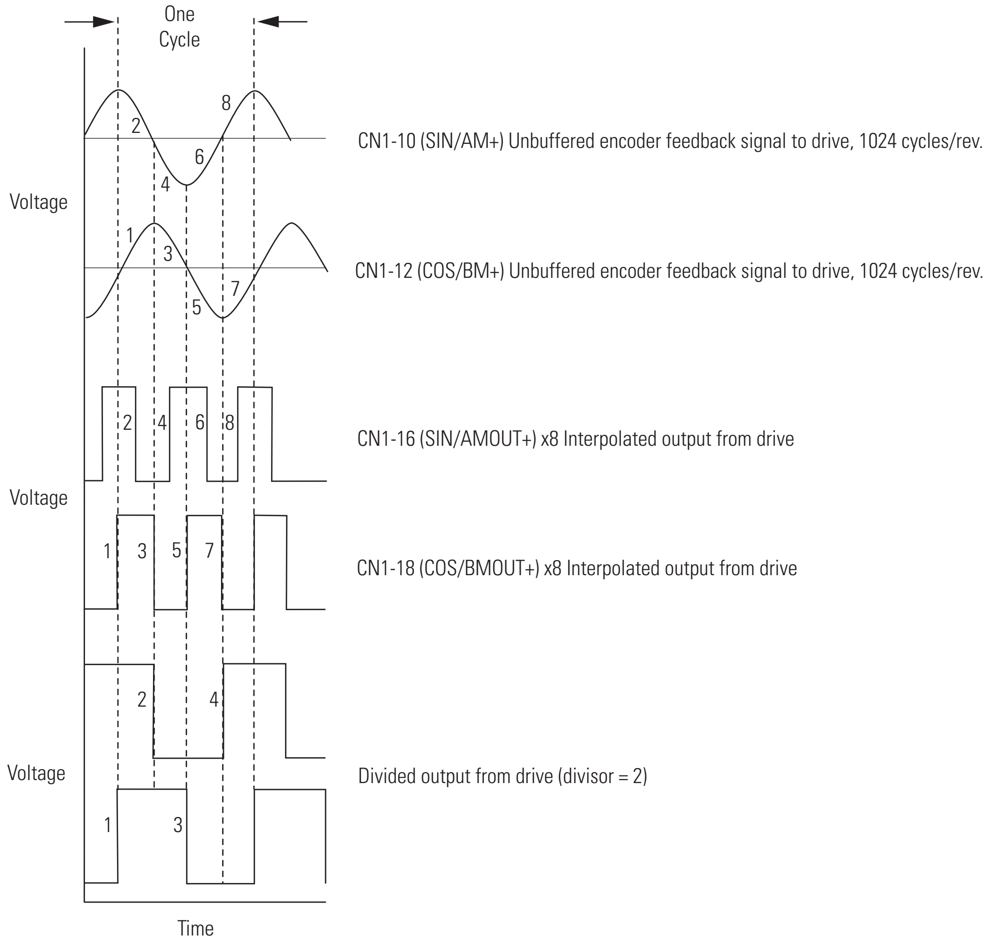


The high resolution buffered outputs (listed below) are available from the drive (CN1-16 through CN1-21) and software selectable.

- **Buffered Outputs** are conditioned SIN/COS signals resulting in a square wave (A quad B) signal (refer to). This signal will have the same number of cycles/rev as the incoming SIN/COS encoder signals found on CN2.
- **Interpolated Outputs** are square wave (A quad B) signals reflecting the interpolation value chosen in software. The minimum interpolation value allowed is x4, which gives the same output as selecting buffered (as described above).
- **Divided Outputs** are the result of a divisor (selected in software) and an interpolation value (also selected in software). For example, with an interpolation value of x8 and a divisor of 2, the CN1 buffered output will be the (x4) square wave representation of the original incoming SIN/COS signal from CN2.

IMPORTANT The interpolation value selected in software is what the drive uses internally to close the feedback loops regardless of any divisor value chosen to condition the signals present on CN1.

Interpolated and Divided Absolute High Resolution Encoder Counts



Appendix C

Minimizing the Effects of Feedback Signal Loss

Introduction

This appendix contains information on how to reduce unexpected motion as a result of feedback signal loss.

ATTENTION ⚠

To avoid personal injury or damage to equipment due to unexpected motion, you must configure the associated motion system parameters and attributes to minimize the effects of feedback signal loss on the system.

In the event of feedback signal loss, unexpected motion may occur. When the feedback signal is lost, the position feedback remains stationary, regardless of the actual axis position and the drive command position. In an effort to compensate for the discrepancy between the position feedback and the drive command position, the drive attempts to maintain axis motion, which quickly results in an increase of position and velocity error. Once the position error reaches or exceeds your specified limit, the drive indicates an Excess Position Error (E19) to alert you of the fault condition. Similarly, an Excess Velocity Error (E24) is indicated once your specified velocity error limit is reached or exceeded.

These system fault indications are intended to alert you of the fault conditions and to aid you in safely bringing the system to a stop. By optimizing the sensitivity threshold and the response time of the drive to position and velocity error fault conditions, you can quickly and safely disable the drive and be alerted by a fault indication to proceed with other safety measures when the feedback signal is lost.

Setting Position Error Limits in Ultraware and RSLogix 5000 Software

Parameters for setting the position and velocity error limits according to the specific needs of the user application exist in both the Ultraware and the RSLogix 5000 software. Adjust these limit settings to be as close to the maximum position and velocity error excursion limit values of the application as possible, but wide enough to avoid nuisance fault tripping. This lets the drive detect abnormal operating conditions and proceed to proper fault handling as quickly as possible. The method and the degree of adjustment will vary depending on the application and on whether Ultraware or RSLogix 5000 software is used.

Minimizing the Position Error Limit Setting

By minimizing the position error limit setting, the amount of motion that can take place before the condition of Excess Position Error (E19) is met can be restricted to a safe value. This safe value is dictated by the application and is defined by you. Adjustment of the position error limit setting is performed by using RSLogix 5000 or Ultraware software, depending on the application.

Setting Position Error Limit Parameters

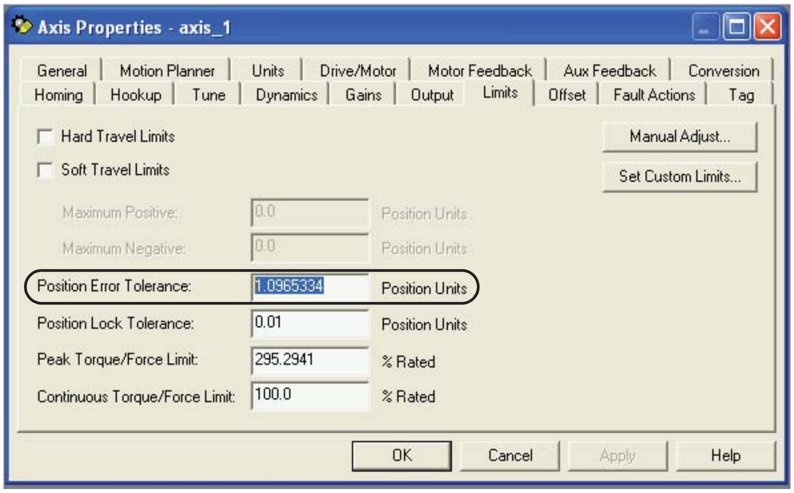
Software	Application	Parameter
RSLogix 5000	SERCOS	Position Error Tolerance, Axis Properties, Limits tab
	Non-SERCOS	
Ultraware	Non-SERCOS	Following Error Limit, Faults branch

Adjust the Position Error Tolerance parameter if using RSLogix 5000 software. This parameter is found in the Limits tab of the Axis Properties dialog. The range for this parameter is 0…2147483647 position units, with a default value determined by the motor or encoder device selected.

You can convert a value, expressed in position units, to a number of axis revolutions by using the raw feedback count, interpolation factor, drive count, and conversion properties of the motor or encoder device and of RSLogix 5000 software Axis Properties settings. In the RSLogix 5000 software, the TTL encoder device interpolation factor is fixed at 4, so 2000 raw feedback counts per revolution results in 8000 interpolated feedback counts per revolution. For sin/cos encoder devices, the interpolation factor is fixed at 2048. The SR encoder devices produce 1024 raw feedback counts per revolution, while the SK encoder devices produce 128 raw feedback counts per revolution. The resulting range of interpolated feedback counts per revolution is 262144…2097152, depending on which sin/cos encoder device is used. Interpolated feedback counts are converted to drive counts by using the Drive Resolution ratio constant found in the Drive/Motor tab of the Axis Properties dialog. Drive counts are converted to position units by using the Conversion ratio constant found in the Conversion tab of the Axis Properties dialog.

The Position Error Tolerance parameter ride-through time setting is fixed at 10 ms and is not adjustable through RSLogix 5000 software. This setting defines the duration of time for which the position error limit setting must be reached or exceeded before an Excess Position Error (E19) is asserted.

RSLogix 5000 Position Error Tolerance Parameter



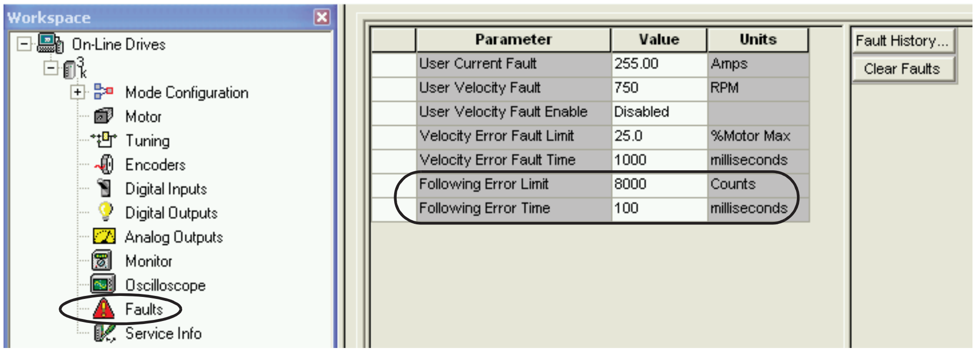
☐ Hard Travel Limits
☐ Soft Travel Limits
Maximum Positive: 0.0 Position Units
Maximum Negative: 0.0 Position Units
Position Error Tolerance: 1.0965334 Position Units
Position Lock Tolerance: 0.01 Position Units
Peak Torque/Force Limit: 295.2941 % Rated
Continuous Torque/Force Limit: 100.0 % Rated

Adjust the Following Error Limit parameter if using Ultraware software. This parameter is found in the Faults branch of the main menu. The range for this parameter is 0…2147483647 interpolated feedback counts, with a default value of 8000 interpolated feedback counts.

You can convert a value expressed in interpolated feedback counts to a number of axis revolutions by using the raw feedback count and interpolation factor (if any) properties of the motor or encoder device and of the Ultraware software Encoder branch. In Ultraware software, the TTL encoder device interpolation factor is fixed at 1, so 2000 raw feedback counts per revolution will result in 2000 interpolated feedback counts per revolution. For the sin/cos encoder devices, the interpolation factor is variable from 4 to 1024. The SR encoder devices produce 1024 raw feedback counts per revolution, while the SK encoder devices produce 128 raw feedback counts per revolution. The resulting range of interpolated feedback counts per revolution is 512…1048576, depending on which sin/cos encoder device is used. Interpolated feedback counts are the units used for the position and velocity error limit parameters in the Ultraware software.

Adjust the Following Error Time parameter to specify a ride-through time for the position error parameter. This setting defines the duration of time for which the position error limit setting must be reached or exceeded before an Excess Position Error (E19) is asserted. The range for the Following Error Time parameter is 0…65,535 ms, with a default value of 100 ms.

Ultraware Following Error Parameters



Parameter	Value	Units
User Current Fault	255.00	Amps
User Velocity Fault	750	RPM
User Velocity Fault Enable	Disabled	
Velocity Error Fault Limit	25.0	%Motor Max
Velocity Error Fault Time	1000	milliseconds
Following Error Limit	8000	Counts
Following Error Time	100	milliseconds

Setting Velocity Error Limits in Ultraware Software

The position error limit and time parameters let you define a tight window of error tolerance for the system. In some instances, however, the nature of the application requires a large tolerance value, and your specified window of error tolerance becomes too wide to be effective in quickly stopping unexpected motion. When this happens, you can implement additional protection in the Ultra3000 drive safety strategy by adjusting the velocity error limit setting.

Minimizing the Velocity Error Limit Setting

By minimizing the velocity error limit setting, you can restrict the amount of velocity error to a safe value before a condition of Excess Velocity Error (E24) is reached. This means that if an unexpected motion event related to feedback loss occurs and is not quickly terminated by the Excess Position Error (E19) response, an Excess Velocity Error (E24) condition will occur and be used to achieve the same rapid termination of the unexpected motion. Adjustment of the velocity error limit setting is readily available in the Ultraware software only. The RSLogix 5000 software velocity error limit setting is fixed at 25% of the maximum motor speed, with a fixed ride-through time setting of 1000 ms.

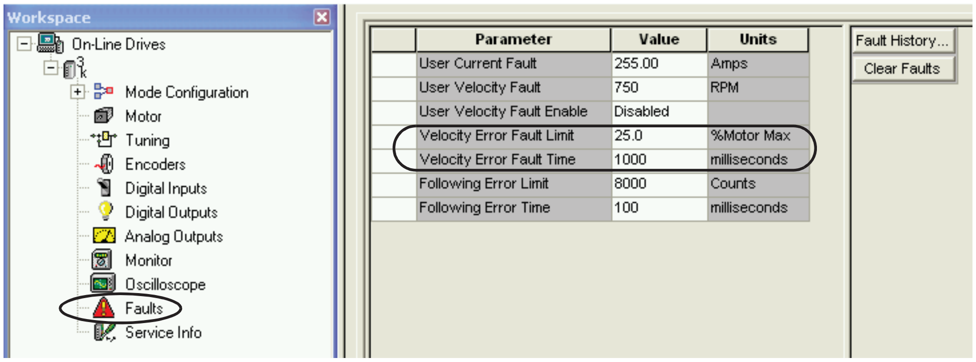
For Ultra3000 drives used with RSLogix 5000 software, the VelocityError attribute is available to you as a tag in the Controller Tags branch of the main workspace window, but the VelocityErrorTime attribute is not. You can incorporate the VelocityError attribute into your RSLogix 5000 application and manually configure fault-handling routines that monitor this attribute. You can also simulate the VelocityErrorTime attribute with an additional code fragment and incorporate it into your fault-handling routines. This technique lets you specify velocity error limit settings in cases where the default settings do not suit the application.

IMPORTANT To make use of the VelocityError attribute, you must select it as one of the two RealTimeAxis Information attributes under the Drive/ Motor tab in the Axis Properties dialog.

For Ultra3000 drives used with Ultraware software, you can configure the Velocity Error Fault Limit parameter found in the Faults branch of the main menu. The range for this parameter is 1…100% of maximum motor speed, with a default value of 25%. Adjust the Velocity Error Fault Time parameter to specify a ride-through time for the Velocity Error Fault Limit parameter. The range for this parameter is 0…65,535 ms, with a default value of 1000 ms.

The Velocity Error Fault Limit parameter is being monitored even when the drive is used in position mode. In this case, the velocity command is the output of the position loop, not the rate of change of the position command. If the position error is large, the output of the position loop, which is also the velocity command, is large. This results in a large velocity error.

Ultraware Velocity Error Fault Parameters



Parameter	Value	Units
User Current Fault	255.00	Amps
User Velocity Fault	750	RPM
User Velocity Fault Enable	Disabled	
Velocity Error Fault Limit	25.0	%Motor Max
Velocity Error Fault Time	1000	milliseconds
Following Error Limit	8000	Counts
Following Error Time	100	milliseconds

Configuring Fault Actions in RSLogix 5000 Software

The sole purpose of adjusting the position and velocity error limit settings to within a tighter range is to detect a position or velocity error fault condition early. This lets you proceed with an appropriate fault action and reduce the distance and duration of unexpected motion. When the position error limit setting is met or exceeded, an Excess Position error (E19) is indicated by the drive. When using Ultraware software, this results in the drive disabling and the axis coasting to a stop. However, when using RSLogix 5000 software, this results in a corresponding fault action that you configure. The response to an Excess Velocity error (E24) in Ultraware software, is the same as that of E19, but is not readily programmable in RSLogix 5000 software. When necessary, you can program fault actions resulting from excess velocity error with additional code.

When configuring the fault actions to minimize unexpected motion, you can select Shutdown or Disable Drive for the Position Error attribute found in the Fault Actions tab of the Axis Properties dialog. In the event of feedback signal loss, either of these settings will result in the drive disabling and the axis coasting to a stop. Selecting the Status Only or Stop Motion fault action causes the drive to simply assert a fault indication, but motion continues when the feedback signal is lost. Axis motion continues until the velocity error fault condition is met, which may take some time. Therefore, selecting the Status Only or Stop Motion fault actions is not recommended as an effective Ultra3000 unexpected motion drive safety strategy.

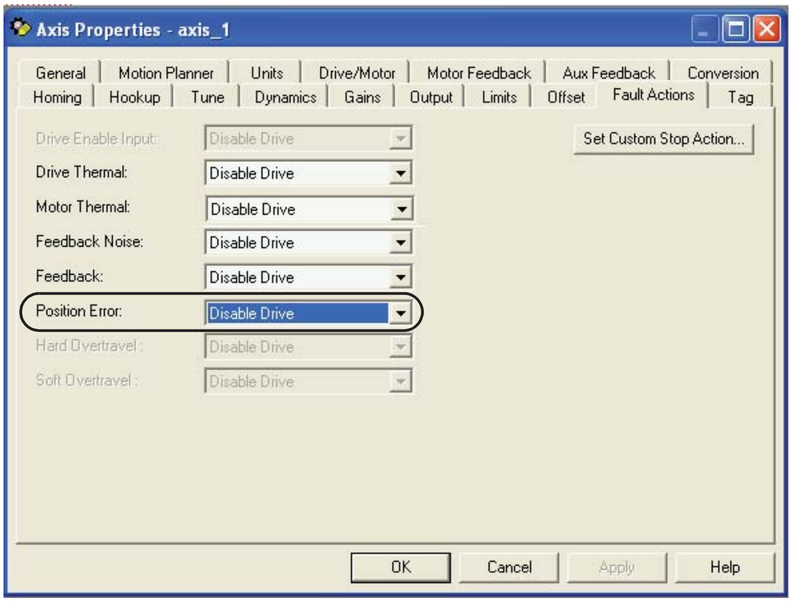
TIP

The Feedback attribute found in the Fault Actions tab of the Axis Properties dialog does not result in any action for the Ultra3000 drive, regardless of the setting. The Feedback Noise attribute does not result in any action due to a position or velocity error fault and should be configured independently, based on the application need.

ATTENTION

To avoid personal injury or damage to equipment due to unexpected motion, the Position Error attribute found in the Fault Actions tab of the Axis Properties dialog must be set to Shutdown or Disable Drive. This results in the drive disabling and the axis coasting to a stop. Further safety measures must also be taken to make sure that, under worst-case conditions, the coasting motion is brought to a safe stop by hardware over-travel limits or other methods.

RSLogix 5000 Axis Properties - Fault Actions Tab



Drive Enable Input: Disable Drive
Drive Thermal: Disable Drive
Motor Thermal: Disable Drive
Feedback Noise: Disable Drive
Feedback: Disable Drive
Position Error: Disable Drive
Hard Overtravel : Disable Drive
Soft Overtravel : Disable Drive

Position and Velocity Error Limit Adjustment Example with Ultraware Software

This example uses Ultraware software and an Ultra3000 indexing drive (catalog number 2098-DSD-030X) with an MP-Series (230V) motor (catalog number MPL-A330P-S) and appropriate power and feedback cables to illustrate position and velocity error limit setting optimization.

A simple motion application is used in this example to perform a repeating clockwise-counterclockwise incremental axis move and observe the default Ultraware position and velocity error limit settings. Then, with the motion application running, you can observe the actual application position and velocity error excursion limit values using the oscilloscope feature of Ultraware software. Based on these excursion limit values, you can adjust the default position and velocity error limit settings to closely match the needs of the application under normal operating conditions. Finally, with the new position and velocity error limit settings in place, you test the system to make sure that it is still functional and that no nuisance faults occur.

Create and Run a Sample Application in Ultraware Software

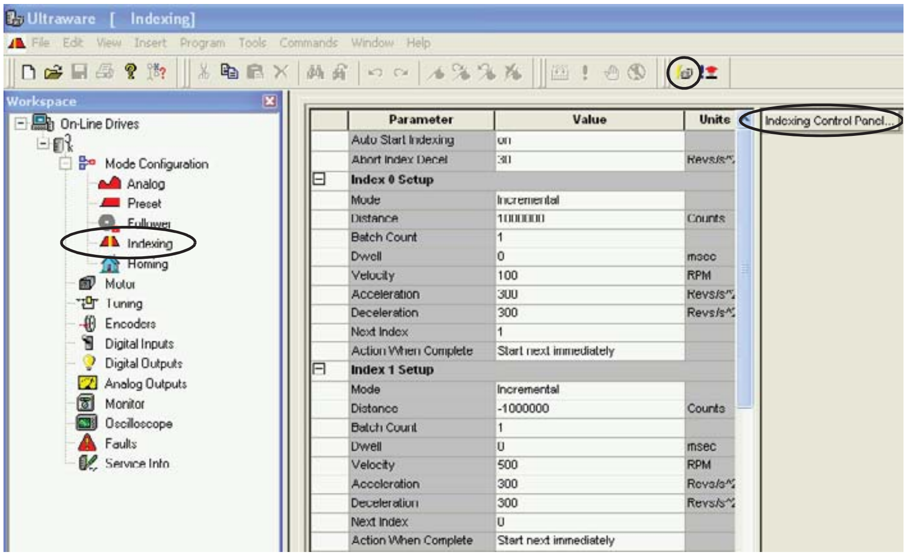
Follow these steps to change the default values in Ultraware software.

1. Open your Ultraware software.

   For help detecting your drive and navigating Ultraware software, refer to Chapter 1.

2. From the workspace, expand Mode Configuration and double-click Indexing.

   The indexing dialog opens with default motion profile values.



Parameter	Value	Units
Auto Start Indexing	on	
Abort Index Decel	30	Revs/s^
Index 0 Setup		
Mode	Incremental	
Distance	1000000	Counts
Batch Count	1	
Dwell	0	msec
Velocity	100	RPM
Acceleration	300	Revs/s^
Deceleration	300	Revs/s^
Next Index	1	
Action When Complete	Start next immediately	
Index 1 Setup		
Mode	Incremental	
Distance	-1000000	Counts
Batch Count	1	
Dwell	0	msec
Velocity	500	RPM
Acceleration	300	Revs/s^
Deceleration	300	Revs/s^
Next Index	0	
Action When Complete	Start next immediately	

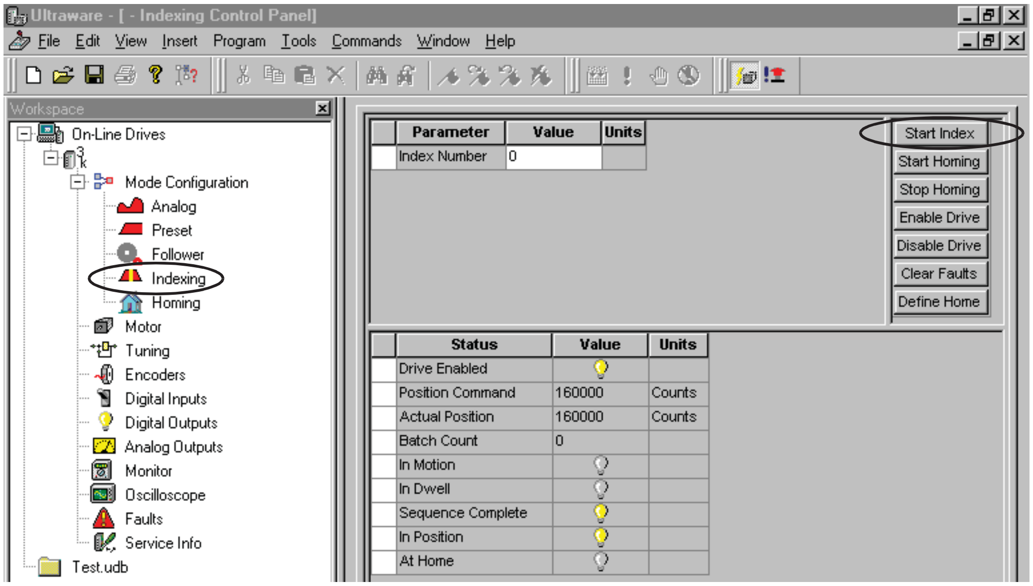
3. Expand Index 0 Setup and Index 1 Setup.

For this example, the default values were replaced with the values shown.

4. Click the Drive Enable icon to enable the Ultra3000 drive.

5. Click Indexing Control Panel.

   The indexing control panel dialog opens.



Parameter	Value	Units
Index Number	0	

Status	Value	Units
Drive Enabled	●	
Position Command	160000	Counts
Actual Position	160000	Counts
Batch Count	0	
In Motion	○	
In Dwell	○	
Sequence Complete	●	
In Position	●	
At Home	○	

ATTENTION

To avoid injury or damage caused by unexpected motion, make sure that all system and user safety measures are taken before running the application.

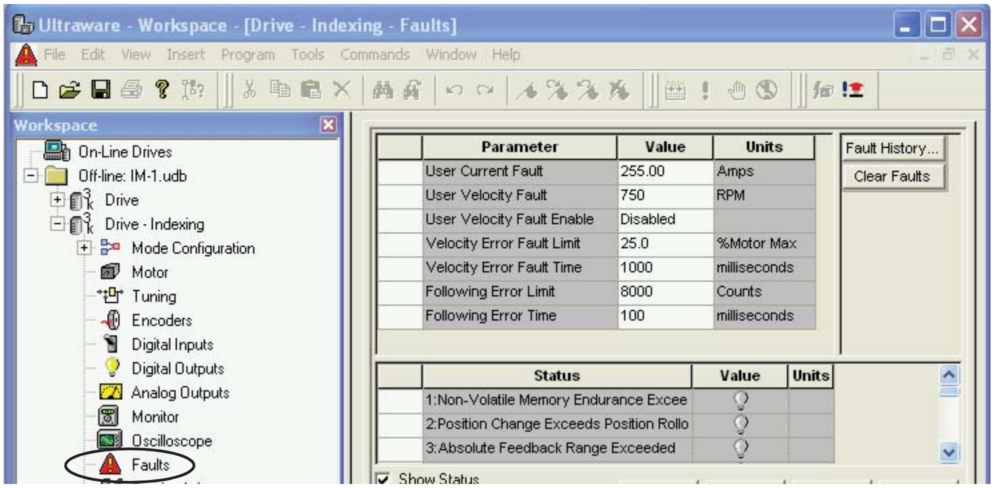
6. Click Start Index.

   The application begins. Keep the application running while observing the default Following Error and Velocity Error Fault limit settings in Ultraware software to understand the significance of their values and units.

Understanding Error Limit Settings in Ultraware Software

With the motion application configured and operational, observe the default Following Error and Velocity Error Fault limit settings in Ultraware software to understand the significance of their values and units.

Ultraware Faults Branch



Parameter	Value	Units
User Current Fault	255.00	Amps
User Velocity Fault	750	RPM
User Velocity Fault Enable	Disabled	
Velocity Error Fault Limit	25.0	%Motor Max
Velocity Error Fault Time	1000	milliseconds
Following Error Limit	8000	Counts
Following Error Time	100	milliseconds

Status	Value	Units
1:Non-Volatile Memory Endurance Excee	○	
2:Position Change Exceeds Position Rollo	○	
3:Absolute Feedback Range Exceeded	○	

☑ Show Status

The Following Error Limit setting is in units of interpolated counts and can be related to actual position error in units of raw feedback counts and axis revolutions by referring to the motor and encoder properties found in the Motor and Encoders branches of Ultraware software.

In this example, the motor encoder provides 1024 raw counts per motor axis revolution with an interpolation constant of 1024. This creates a total of 1048576 interpolated counts per motor axis revolution. Therefore, the default 8000 interpolated-count position-error limit setting allows for maximum position error of 0.76% of a full motor axis revolution before the condition of Excess Position Error (E19) is reached.

The Velocity Error Fault Limit setting is in units of percent of maximum motor speed and can be related to actual velocity by referring to the motor properties under the Motor branch. In this example, the motor is rated for a maximum speed of 5000 rpm. Therefore, the default 25% setting allows for as much as 1250 rpm of velocity error before the condition of Excess Velocity Error (E24) is reached.

Optimal Following Error Limit/Time and Velocity Error Fault Limit/ Time settings are chosen based on your knowledge of the system requirements and the maximum system position and velocity error excursion limit values. The default limit settings may not be optimal for every application. You must understand the application needs and adjust these default limit settings as appropriate.

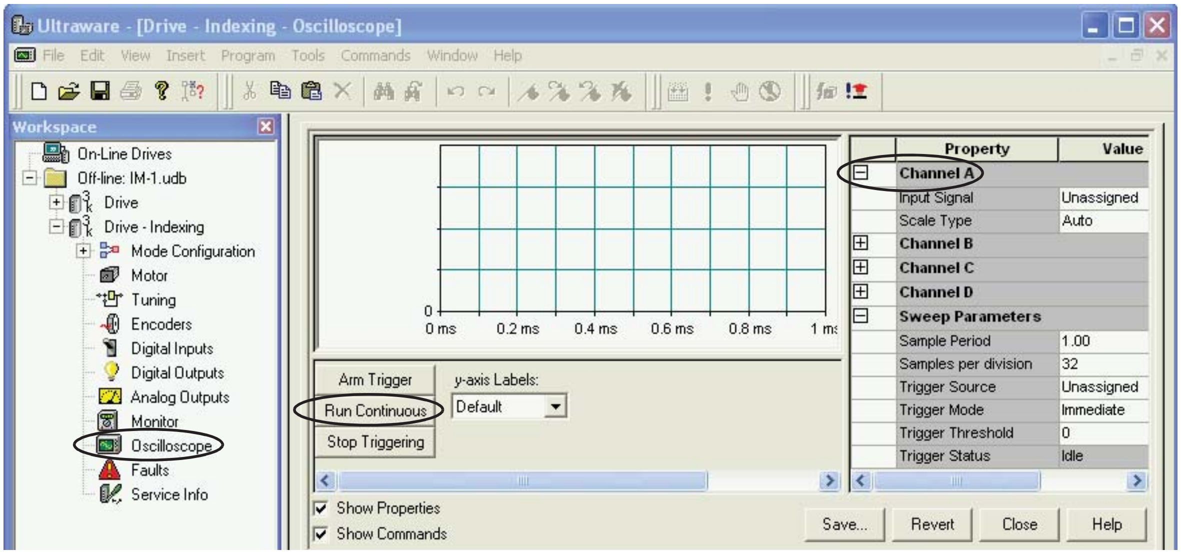
Use the Oscilloscope Feature

Once the system is running, the default position and velocity error limit values can be monitored and optimized via the oscilloscope branch.

IMPORTANT Since RSLogix 5000 software already provides an efficient plotting utility, do not use this feature with a SERCOS drive. Using the Ultraware oscilloscope feature, along with RSLogix 5000 software, creates a substantial drive processor load and may interfere with normal drive operation.

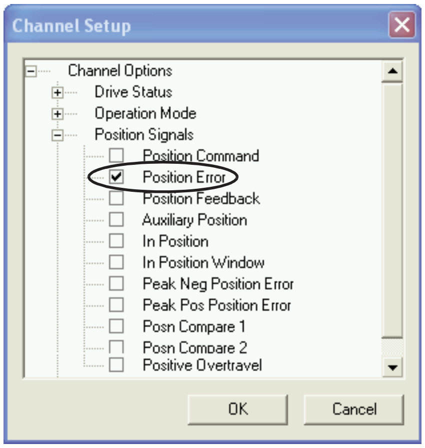
Follow these steps to configure the oscilloscope branch.

1. Double-click the Oscilloscope branch and expand the Channel A tab on the right side of the dialog.



Property	Value
Channel A	
Input Signal	Unassigned
Scale Type	Auto
Channel B	
Channel C	
Channel D	
Sweep Parameters	
Sample Period	1.00
Samples per division	32
Trigger Source	Unassigned
Trigger Mode	Immediate
Trigger Threshold	0
Trigger Status	Idle

2. Click the Input Signal default value and use the pull-down menu to select Position Error from the Channel Setup menu.



- Channel Options
  - Drive Status
  - Operation Mode
  - Position Signals
    - ☐ Position Command
    - ☑ Position Error
    - ☐ Position Feedback
    - ☐ Auxiliary Position
    - ☐ In Position
    - ☐ In Position Window
    - ☐ Peak Neg Position Error
    - ☐ Peak Pos Position Error
    - ☐ Posn Compare 1
    - ☐ Posn Compare 2
    - ☐ Positive Overtravel

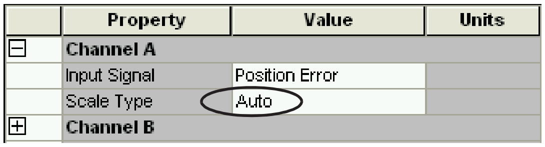
3. Click OK.

4. Click the Scale Type default value and select Auto.

5. Click the Run Continuous trigger setting and observe the main dialog.

   The Position Error signal of the running system is displayed. It is likely that the signal is highly dynamic, and the auto-scaling feature of Ultraware automatically adjusts the dialog scaling to fit the signal for every sampling and trigger instance.

6. Click the Scale Type value and select Auto.



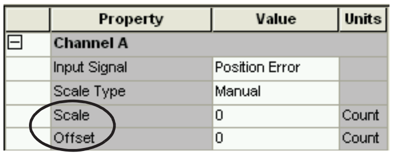
	Property	Value	Units
⊟	**Channel A**		
	Input Signal	Position Error	
	Scale Type	Auto	
⊞	**Channel B**		

7. Click Run Continuous and observe the main oscilloscope window.

   The Position Error signal of the running system is displayed. It is likely that this signal is highly dynamic, and the auto-scaling feature of Ultraware software automatically adjusts the oscilloscope window vertical scale to fit the entire signal waveform for every sampling instance.

8. If a fixed scale is desired, Click the Scale Type value again and select Manual.

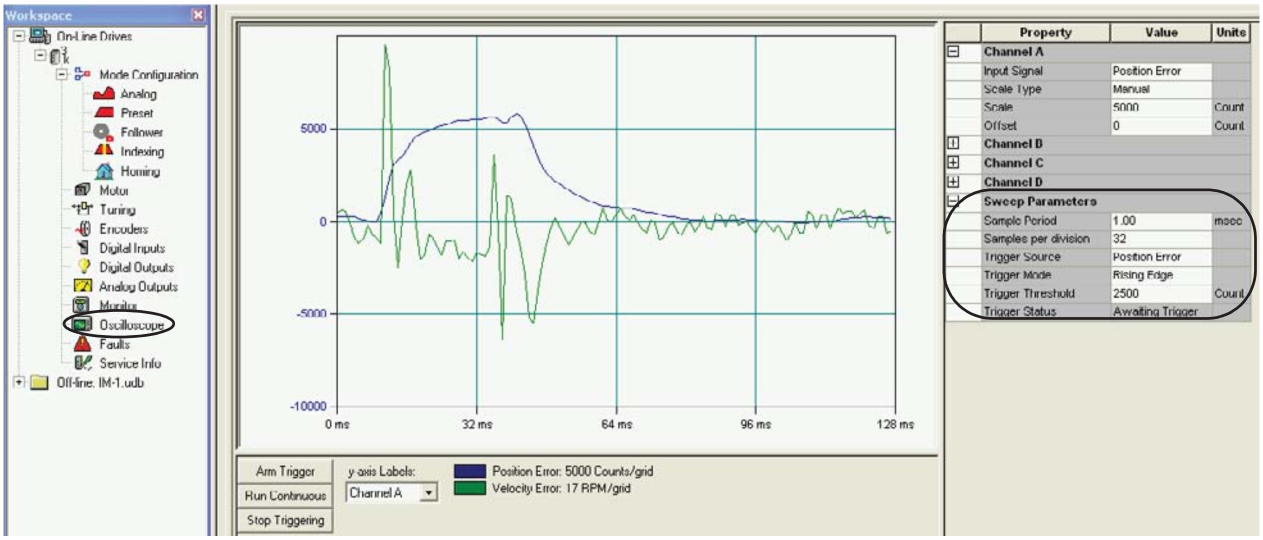
   The Scale and Offset fields appear.



	Property	Value	Units
⊟	**Channel A**		
	Input Signal	Position Error	
	Scale Type	Manual	
	Scale	0	Count
	Offset	0	Count

9. Enter values (in counts) in the Scale and a Offset fields that results in the entire periodic Position Error signal waveform visible in the oscilloscope window.

   Steps 8 and 9 are purely for the purpose of visualization. Fixing the vertical scale of the oscilloscope window lets you observe the amplitude excursion limits of the Position Error signal from a fixed range of reference.



y-axis Labels: Channel A

	Property	Value	Units
⊟	**Channel A**		
	Input Signal	Position Error	
	Scale Type	Manual	
	Scale	5000	Count
	Offset	0	Count
⊞	**Channel B**		
⊞	**Channel C**		
⊞	**Channel D**		
⊟	**Sweep Parameters**		
	Sample Period	1.00	msec
	Samples per division	32	
	Trigger Source	Position Error	
	Trigger Mode	Rising Edge	
	Trigger Threshold	2500	Count
	Trigger Status	Awaiting Trigger	

10. Repeat steps 1…6 for Channel B and view the Velocity Error signal.

    Access the Velocity Error signal in the Velocity Signals group in the same Channel Options pull-down menu. Make sure to use the Auto scale setting for Channel B. Both signals should be visible in the main oscilloscope window. You can toggle between Channel A and Channel B on the y-axis scale with the y-axis Labels field pull-down menu.

Using the Trigger Feature

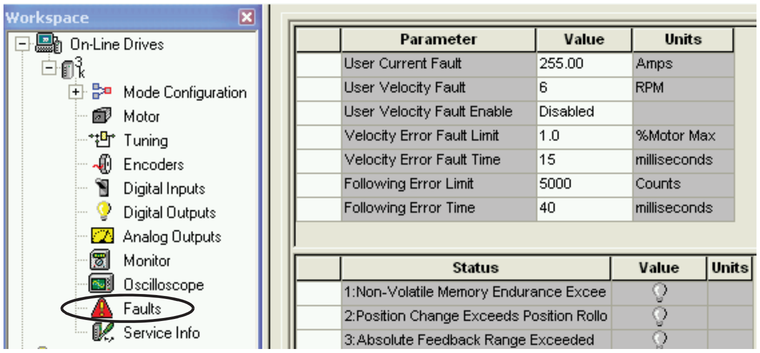
The bottom-right portion of the screen contains trigger level settings for oscilloscope Channel A and B. You can specify the level settings, in interpolated drive counts, to trigger on actual Position Error signal amplitude levels. You can use the trigger feature to capture significant Position Error amplitude events during normal system operation and to get very accurate application excursion limits. Using the trigger feature is an alternative to visually extracting the Position Error excursion limits.

Interpreting the Results

Once the actual position and velocity error excursion limit values are known, you can adjust the default Following Error Limit/Time and Velocity Error Fault Limit/Time settings to be just above these actual application extremes. This makes sure that any feedback loss events causing abnormal position and velocity error amplitude increase are detected and addressed by the drive in a timely manner.

In this example, the worst case Position Error is greater than 5000 counts, for around 32 ms, and the worst case Velocity Error is greater than 22 rpm, for around 8 ms. The position and velocity error limit settings in the Faults branch can be adjusted, according to these observations, to 5000 counts for 40 ms and 50 rpm (or 1% of 5000) for around 15 ms, respectively. The settings take effect as soon as they are entered.

Ultraware Faults Branch



	Parameter	Value	Units
	User Current Fault	255.00	Amps
	User Velocity Fault	6	RPM
	User Velocity Fault Enable	Disabled	
	Velocity Error Fault Limit	1.0	%Motor Max
	Velocity Error Fault Time	15	milliseconds
	Following Error Limit	5000	Counts
	Following Error Time	40	milliseconds

	Status	Value	Units
	1:Non-Volatile Memory Endurance Excee	○	
	2:Position Change Exceeds Position Rollo	○	
	3:Absolute Feedback Range Exceeded	○	

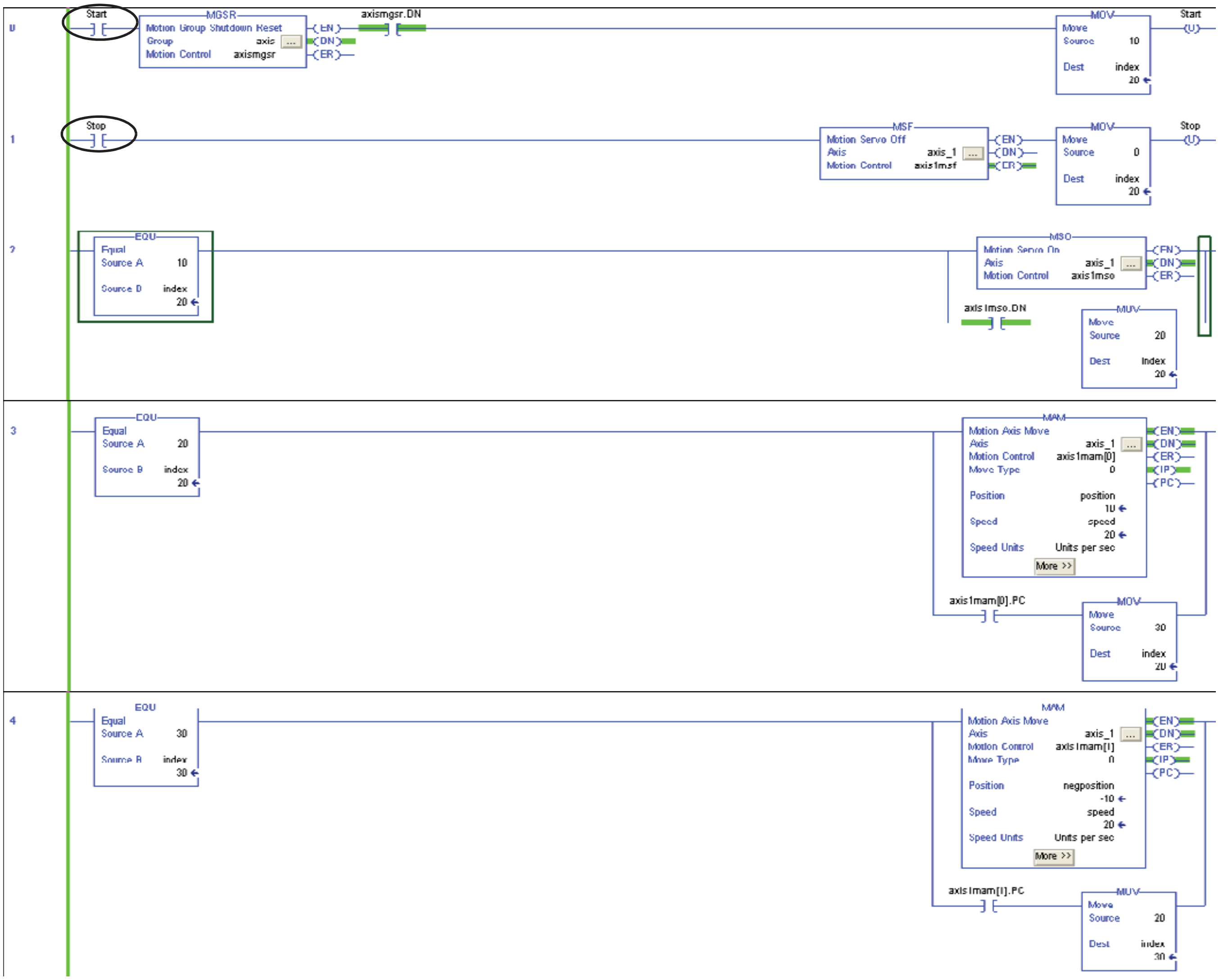
The new position and velocity error limit settings are significantly lower than the default settings. Once the system is operating with new settings in place, no nuisance faults should occur. If faults E19 or E24 occur during normal system operation, the excursion limit values were not accurately extracted and the appropriate steps need to be repeated.

Position Error Limit Adjustment Example with RSLogix 5000 Software

This example uses RSLogix 5000 software, a ControlLogix controller (catalog number 1756-L63), and an Ultra3000i indexing drive (catalog number 2098-DSD-030X) with a 230V MP-Series motor (catalog number MPL-A330P-S) and appropriate power and communication cables to illustrate position error limit setting optimization. A simple motion application is used in this example to perform a repeating clockwise-counterclockwise incremental axis move.

In this example, you will observe the default position error limit setting in RSLogix 5000 software and, with the motion application running, observe the actual application position error excursion limit values. This is done by using the trending feature of the RSLogix 5000 software. Based on these excursion limit values, you can adjust the default position error limit settings to closely match the needs of the application under normal operating conditions. Finally, with the new position error limit setting in place, you can test the system to make sure that it is still functional and that no nuisance faults occur.

Ladder Logic Diagram for Position Error Limit Adjustment



In the ladder logic example diagram, the Start input (when activated) enables the axis for motion. The Stop input disables the axis. A repeating sequence of two MAM commands is performed to move the axis an incremental distance in one direction and then return the axis to its original position.

Run the RSLogix 5000 Software Example Program

Follow these steps to run the example program.

1. Open your RSLogix 5000 software and create an application as shown in the ladder logic example diagram on page 154.
2. Download your program to the Logix controller.
3. Check to make sure the software is online and in Run mode.

ATTENTION Make sure that all system and user safety measures are taken before starting the motion application.

4. Right-click the Start input in the ladder logic diagram and select Toggle Bit.

   The motion application begins to run. While the application is running, the ladder logic diagram displays the state of the motion application by highlighting active ladder logic elements as they are used by the application.

5. With the motion application configured and operational, proceed to Observe the Default Position Error Tolerance Limit Setting on page 156.

Observe the Default Position Error Tolerance Limit Setting

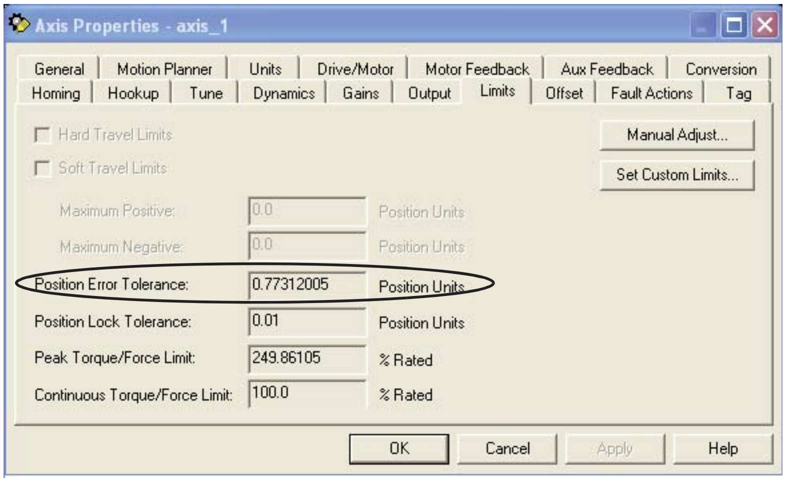
To understand the significance of the default Position Error Tolerance limit setting, observe the default value in RSLogix 5000 software.

TIP The Position Error Tolerance limit setting, units and conversion settings, and real-time axis attribute settings are accessible through the Axis Properties dialog.

Follow these steps to observe the Position Error Tolerance setting.

1. Right-click your axis in the Explorer dialog and choose Properties.

   The Axis Properties dialog opens.

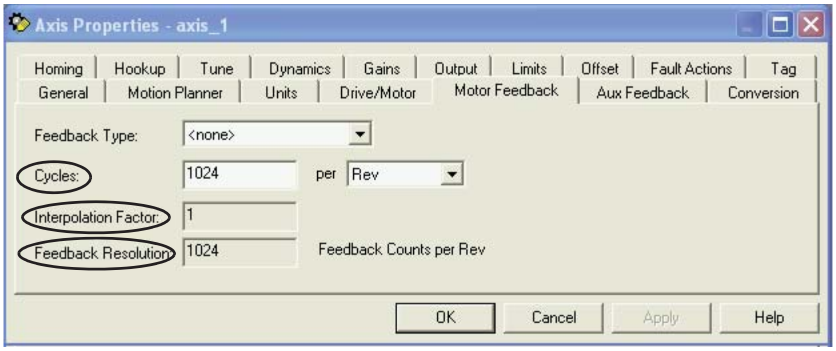


☐ Hard Travel Limits
☐ Soft Travel Limits
Maximum Positive: 0.0 Position Units
Maximum Negative: 0.0 Position Units
Position Error Tolerance: 0.77312005 Position Units
Position Lock Tolerance: 0.01 Position Units
Peak Torque/Force Limit: 249.86105 % Rated
Continuous Torque/Force Limit: 100.0 % Rated

2. Click the Limits tab.

   In this example, the default setting for Position Error Tolerance is 0.77312005 position units. This value can be related to position error in units of raw, interpolated, and drive resolution feedback counts and axis revolutions. Refer to the feedback cycles, interpolation factor, drive resolution, and conversion constant properties found under the Motor Feedback, Drive/Motor, and Conversion tabs of the Axis Properties menu.

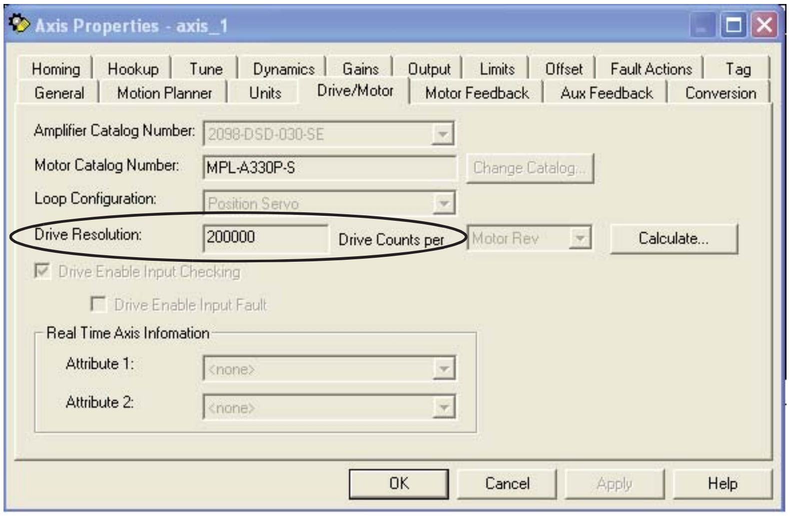
3. Click the Motor Feedback tab.



Feedback Type: <none>
Cycles: 1024 per Rev
Interpolation Factor: 1
Feedback Resolution: 1024 Feedback Counts per Rev

4. Observe the Cycles, Interpolation Factor, and Feedback Resolution values.

5. Click the Drive/Motor tab.

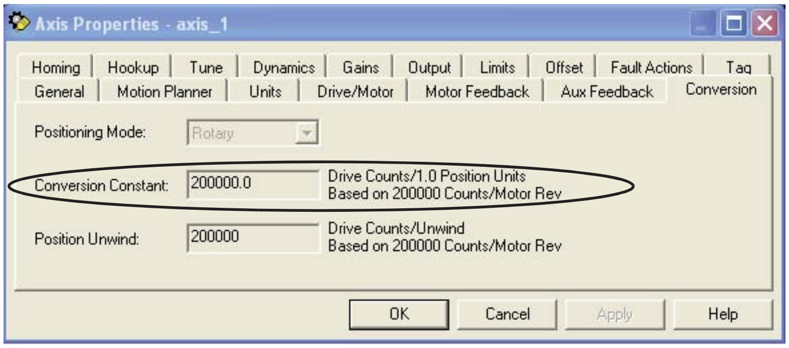


Amplifier Catalog Number: 2098-DSD-030-SE
Motor Catalog Number: MPL-A330P-S
Loop Configuration: Position Servo
Drive Resolution: 200000 Drive Counts per Motor Rev
☑ Drive Enable Input Checking
☐ Drive Enable Input Fault
Real Time Axis Infomation
Attribute 1: <none>
Attribute 2: <none>

6. Observe the Drive Resolution value.

For the MPL-A330P-S motor used in this example, 1024 raw feedback counts per 1 axis revolution are used to generate 2097152 (1024 x 2048) interpolated feedback counts per 1 axis revolution by using an interpolation factor of 2048. These interpolated feedback counts are converted to 200,000 drive resolution counts per 1 axis revolution by setting the Drive Resolution parameter in the Drive/Motor dialog to 200,000.

7. Click the Conversion tab.



Positioning Mode: Rotary

Conversion Constant: 200000.0 Drive Counts/1.0 Position Units Based on 200000 Counts/Motor Rev

Position Unwind: 200000 Drive Counts/Unwind Based on 200000 Counts/Motor Rev

8. Observe the Conversion Constant value.

The Conversion tab shows how the user-defined position units relate to drive resolution counts, and, therefore, to raw and interpolated feedback counts and axis revolutions.

Summary

In this example, the Position Error Tolerance limit setting of 0.77312005 position units at 200,000 drive resolution counts per 1 position unit results in approximately 154,624 drive resolution counts (0.77312005 x 200,000) of allowable position error. Because in this case, 1 position unit is also equal to one axis revolution, the Position Error Tolerance limit setting is equal to approximately 77.3% (0.77312005) of 1 axis revolution.

The optimal Position Error Tolerance limit setting is chosen based on your knowledge of the system requirements and the maximum system position error excursion limit values. The default limit setting shown above may not be optimal for every application. You must understand the application needs and adjust this default limit setting as appropriate.

TIP The Dynamics tab provides additional limit parameters, such as maximum speed, acceleration, and deceleration of the drive and motor system. These limits are specified in the same position units.

Trending Excursion Limits of the Position Error Parameter

With the motion application running, you can trend the position error parameter. The trending feature of RSLogix 5000 software lets you plot drive parameters in real-time and view them in graphical form. Before trending of any drive parameter is possible, the Real Time Axis Attribute field, found under the Axis Properties dialog, Drive/Motor tab, must be configured to specify which of the numerous drive parameters (tags) should be made available from the drive to the RSLogix 5000 software and Logix controller for the purpose of trending.

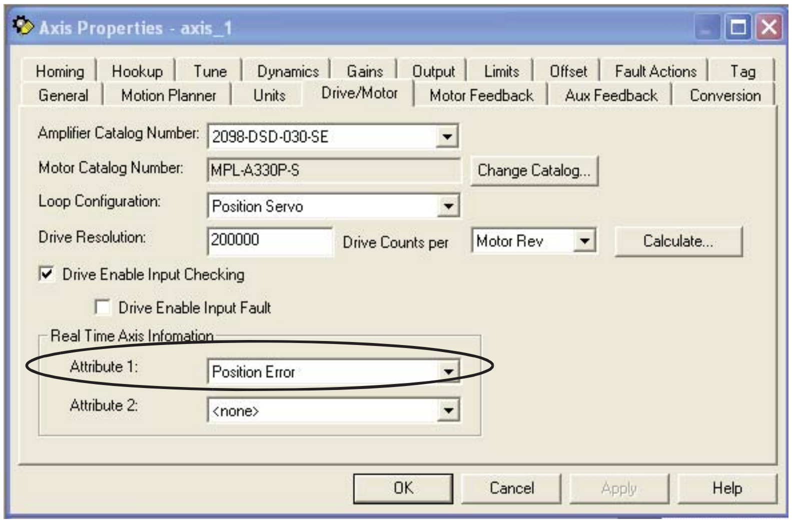
TIP The trending feature is accessible through the Controller Tags branch in the RSLogix 5000 workspace dialog. Right-click the parameter tag that you wish to trend and select Trend (parameter).

Configure the Position Error Parameter Tag for Trending

Follow these steps to configure the position error parameter tag for trending in RSLogix 5000 software.

1. Set the Logix controller to the Offline mode.
2. Right-click your axis of interest in the Explorer dialog and choose Properties.

   The Axis Properties dialog opens.



Amplifier Catalog Number: 2098-DSD-030-SE

Motor Catalog Number: MPL-A330P-S

Loop Configuration: Position Servo

Drive Resolution: 200000 Drive Counts per Motor Rev

☑ Drive Enable Input Checking

☐ Drive Enable Input Fault

Real Time Axis Infomation

Attribute 1: Position Error

Attribute 2: <none>

3. Click the Drive/Motor tab.

4. From the Attribute 1 pull-down menu, choose the Position Error parameter.

   This configures the Logix controller to receive the position error parameter data from the drive.

5. Click OK.

Locate the Position Error Parameter Tag and Configure It for Trending

Once the system is running and the Logix controller is configured to receive position error parameter data from the drive, the position error characteristics of the Ultra3000 drive and motor system can be monitored by using the trending feature of RSLogix 5000 software.

Follow these steps to locate the Position Error parameter tag and configure it for trending in RSLogix 5000 software.

1. Set the Logix controller to the Online mode.

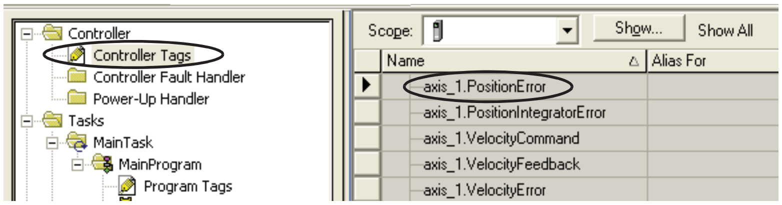
2. Download the program to the controller when prompted.

3. Start your application.

   In this example, the Start input is used (right-click and choose Toggle Bit).

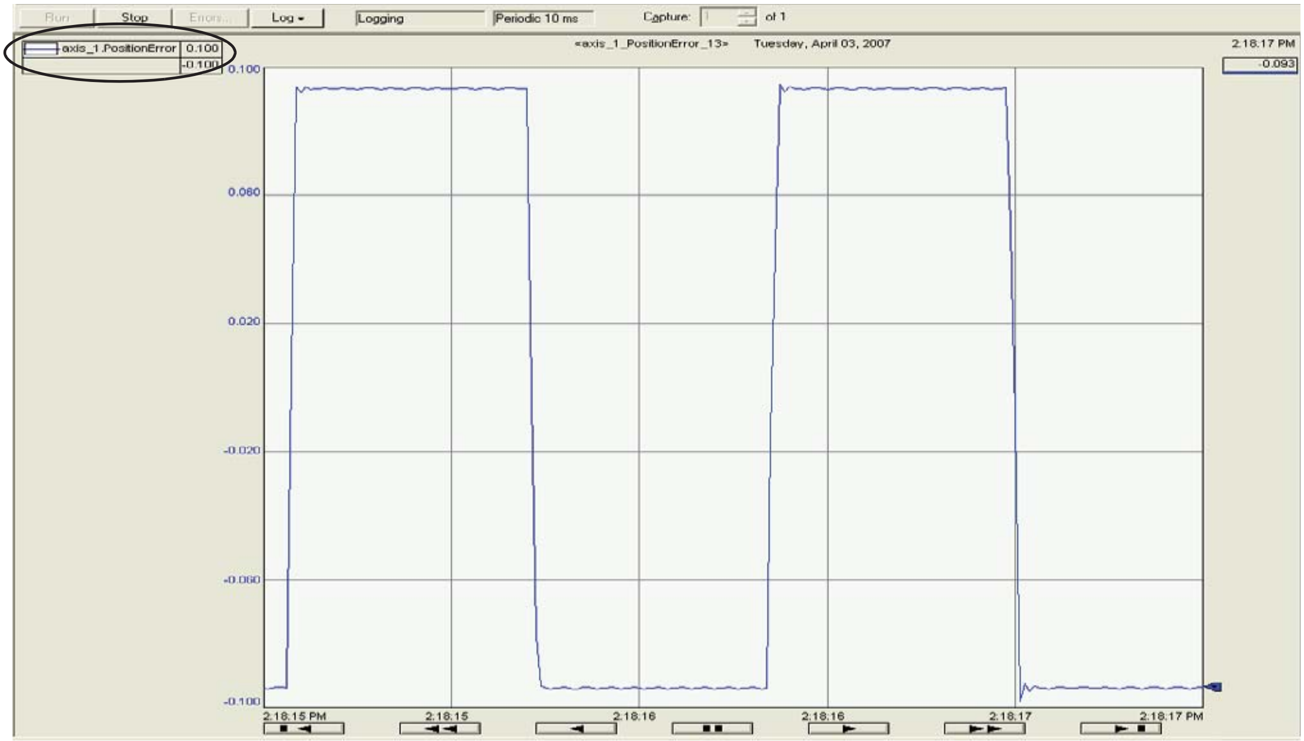
4. Double-click Controller Tags in the Explorer dialog and choose Position Error.

5. Expand the axis_*x* tag list for your axis and scroll down to the Position Error parameter.



6. Right-click the PositionError parameter tag and choose Trend Axis_*x*.PositionError.

   The RSLogix 5000 trending window opens and the position error signal of the running Ultra3000 drive/motor system becomes visible.



The default trending dialog settings are sufficient to display the position-error waveform. However, you may need to adjust the default dialog settings so you can capture and extract the position error excursion limit values of the running application more easily.

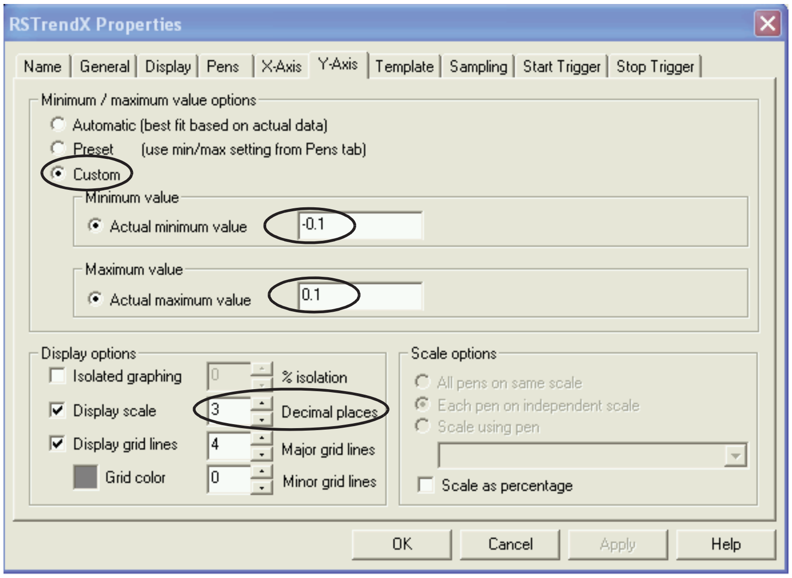
Change the Default Trending Dialog Settings

You can change the X or Y scales of the trending dialog (time base and amplitude, respectively) or the sampling period of the acquisition cycle by using the RSTrendX chart properties control panel.

Follow these steps to change the default trending dialog settings.

1. Right-click the trending dialog and choose Chart Properties.

The RSTrendX Properties dialog opens.



Minimum / maximum value options
○ Automatic (best fit based on actual data)
○ Preset (use min/max setting from Pens tab)
● Custom

Minimum value
● Actual minimum value -0.1

Maximum value
● Actual maximum value 0.1

Display options
☐ Isolated graphing 0 % isolation
☑ Display scale 3 Decimal places
☑ Display grid lines 4 Major grid lines
Grid color 0 Minor grid lines

Scale options
○ All pens on same scale
● Each pen on independent scale
○ Scale using pen
☐ Scale as percentage

2. Click the Y-Axis tab.

3. Under Display options, change the number of decimal places to 3 (or more), depending on the resolution needed for your application.

4. Click Apply.

5. Click OK to close the RSTrendX dialog.

6. Observe the trending dialog waveform again.

Make note of the positive and negative excursion limit values for position error waveform.

7. Right-click the trending window and click the Y-Axis tab.

The Y-Axis tab opens again.

8. Under Minimum/maximum value options, click Custom.

9. Set the Minimum and Maximum values to capture the positive and negative excursion limit range observed in your application.

10. Click OK to accept the new trending dialog settings.

    In this example, the Y-Axis scaling was adjusted to show a range of -0.10 (min) to 0.10 (max) position units, with 3 decimal places of resolution. If the waveform is not symmetrical, the minimum and maximum values may be different from each other.

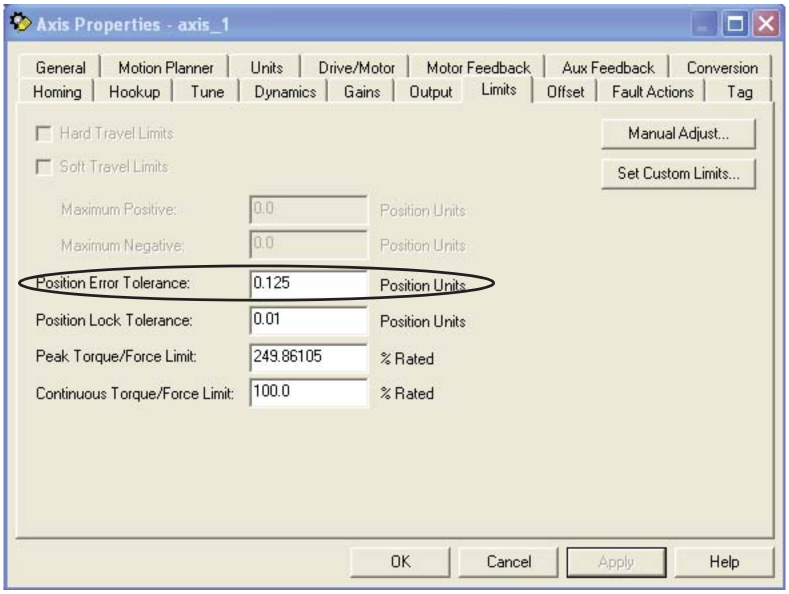
Set the New Position Error Limit

Follow these steps to adjust the default Position Error Tolerance limit setting based on your knowledge of the position error excursion limit values of the application.

1. Set the Logix controller to the Offline mode.

2. Right-click your axis in the Explorer dialog and choose Properties.

   The Axis Properties dialog opens.



☐ Hard Travel Limits
☐ Soft Travel Limits
Maximum Positive: 0.0 Position Units
Maximum Negative: 0.0 Position Units
Position Error Tolerance: 0.125 Position Units
Position Lock Tolerance: 0.01 Position Units
Peak Torque/Force Limit: 249.86105 % Rated
Continuous Torque/Force Limit: 100.0 % Rated

3. Click the Limits tab.

4. Change the Position Error Tolerance limit setting according to your minimum and maximum excursion limit values obtained earlier.

   In this example the new Position Error Tolerance setting is 0.125 position units, just above the 0.10 position units observed in the trending dialog. This position error margin is added to avoid nuisance trips. Values above this new limit setting will cause a position error fault (E19).

5. Click OK.

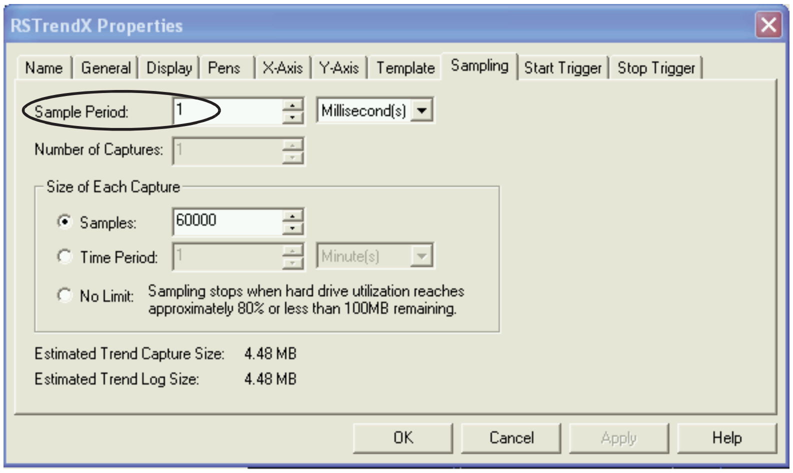
Verify the New Position Error Limit

You can use the RSLogix 5000 trigger feature to detect parameter tag events which meet or exceed your new Position Error Tolerance limit setting. After configuring the trigger feature, you can verify that the new settings are not exceeded by the system under normal operating conditions.

Follow these steps to set up the trigger feature and verify the new Position Error Tolerance limit setting.

1. Right-click the trending dialog and choose Chart Properties.

   The RSTrendX Properties dialog opens.



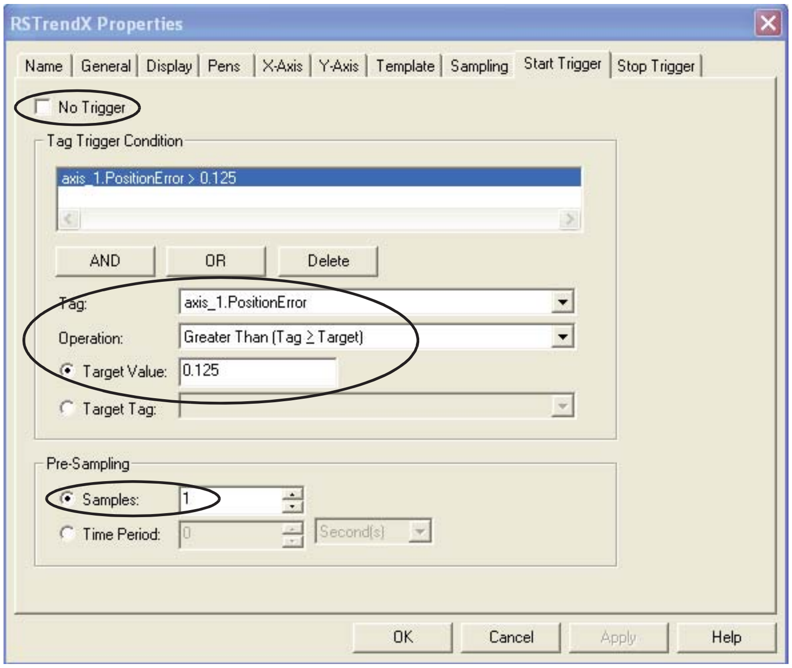
Sample Period: 1 Millisecond(s)
Number of Captures: 1
Size of Each Capture
● Samples: 60000
○ Time Period: 1 Minute(s)
○ No Limit: Sampling stops when hard drive utilization reaches approximately 80% or less than 100MB remaining.
Estimated Trend Capture Size: 4.48 MB
Estimated Trend Log Size: 4.48 MB

2. Click the Sampling tab.

3. Adjust the Sampling Period to provide sufficient resolution to visualize your application.

   A sampling period of 1.0 ms is used in this example.

4. Click the Start Trigger tab.



☐ No Trigger

axis_1.PositionError > 0.125

Tag: axis_1.PositionError

Operation: Greater Than (Tag ≥ Target)

● Target Value: 0.125

○ Target Tag:

● Samples: 1

○ Time Period: 0 Second(s)

5. Uncheck the No Trigger checkbox.

6. From the Tag pull-down menu, choose axis_*x*.PositionError.

7. From the Operation pull-down menu, choose Greater Than (Tag≥Target).

   The trigger is now set to when the parameter exceeds the target value.

8. Click Target Value and enter the new limit.

   In this example, the target value is 0.125 position counts.

9. Click Samples and set the value to 1 (single sample).

   This speeds up the trigger response time.

10. Click OK.

Visualize the New Position Error Limit

Follow these steps to visualize the new position error limit.

1. Set the Logix processor to online operation.
2. Download the program to the controller when prompted.
3. Right-click the Start bit in your program and choose Toggle Bit.
4. Click Run and observe the trending dialog.

   No waveform (no trigger events) indicates the new position error limit threshold is not exceeded.

The new Position Error Tolerance limit settings are verified when no trigger events occur in the application under normal operating conditions. If trigger events occur, it is likely that the excursion limit values are not properly captured and need to be determined again. Repeat the excursion limit extraction process and assure that absolute (global) positive and negative excursion limit values are captured.

Additional Methods

It is good practice to make sure that your application can maintain the physical speed and position limits of the overall system. Use these additional safety measures to guard against unexpected motion.

- Hardware over-travel protection
- Dynamic braking circuit
- Bulletin 2090 resistive brake module (RBM)

Exporting and Importing Drive Setup Files

This appendix provides you with procedures for exporting and importing drive setup files used with Ultra3000, Ultra3000X with indexing, Ultra3000-DN DeviceNet, and Ultra3000X-DN DeviceNet with indexing servo drives.

Introduction

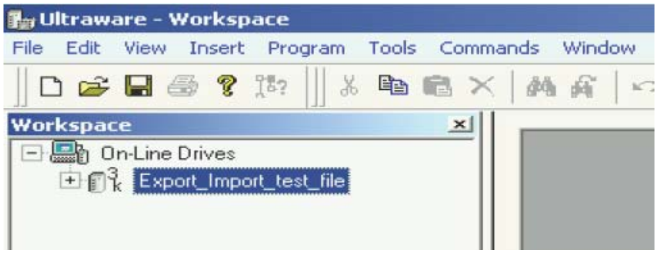
Drive setup files contain all the configuration data used by Ultraware software to run a particular application. In the event you are replacing your Ultra3000 drive you can reuse the original setup file. Follow these procedures to export the setup file to a temporary location and then import that same file and associate it with the replacement drive.

Export a Drive Setup File

Follow these steps to export a drive setup file.

1. Open your Ultraware software and go online with the replacement Ultra3000 drive.

   Refer to Detect Your Ultra3000 Drive on page 16, for more information on starting the Ultraware software and detecting your drive.

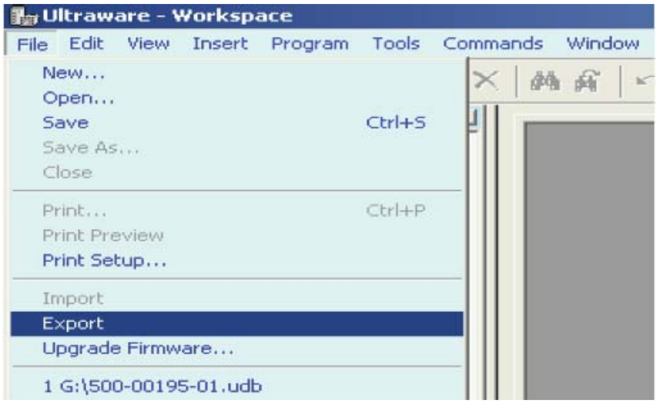


2. Select the drive setup file.

   In this example, Export_Import_test_file is selected.

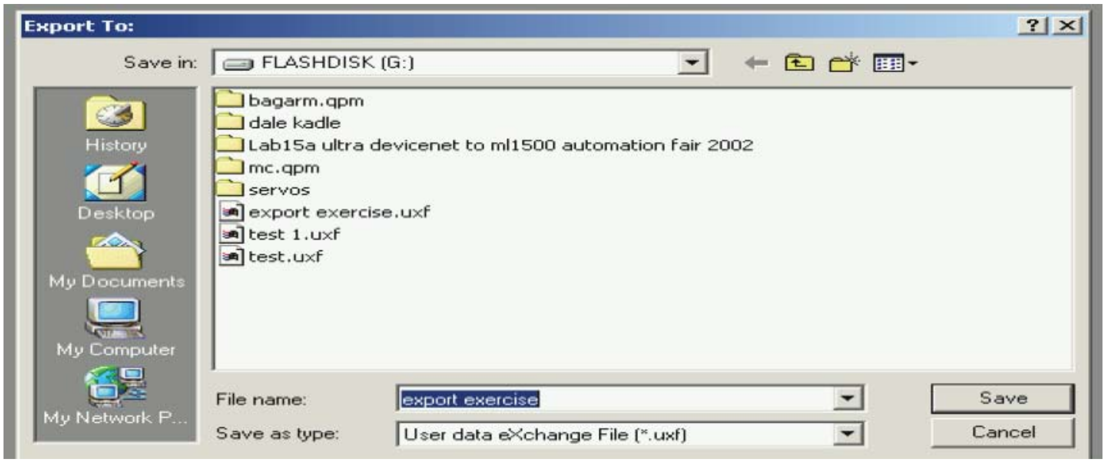
3. Choose Export from the File menu.

   If Export is grey and cannot be chosen, then you haven't selected a file to export.



   The Export To: dialog opens.

4. Browse to a location to temporarily store the drive setup file.



Save in: FLASHDISK (G:)

bagarm.qpm
dale kadle
Lab15a ultra devicenet to ml1500 automation fair 2002
mc.qpm
servos
export exercise.uxf
test 1.uxf
test.uxf

File name: export exercise
Save as type: User data eXchange File (*.uxf)

5. Name the file.

   The file is assigned the .uxf (Ultraware Exchange File) extension.

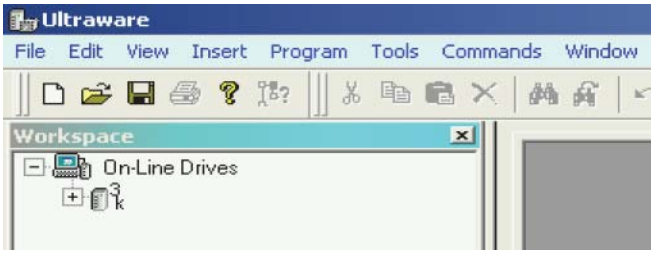
6. Click Save.

Import a Drive Setup File

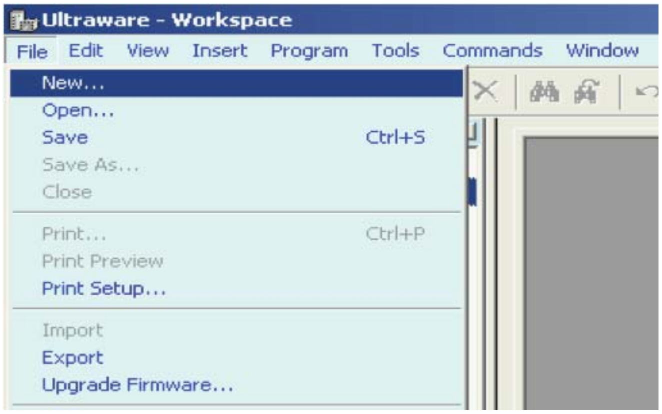
Follow these steps to import a drive setup file.

1. Open your Ultraware software and go online with the replacement Ultra3000 drive.

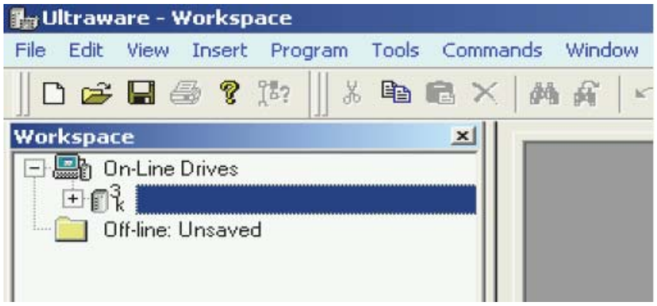
   Refer to Detect Your Ultra3000 Drive on page 16, for more information on starting the Ultraware software and detecting your drive.



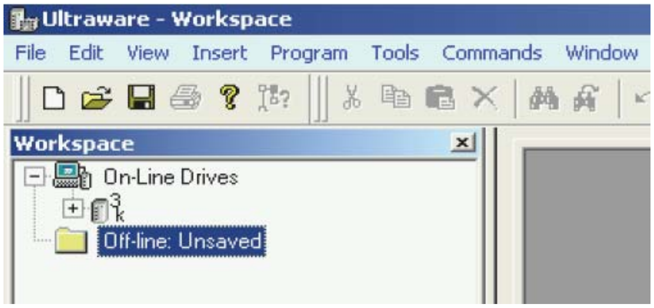
2. From the File menu, choose New to create an offline place to put the drive setup file.



   An Off-line: Unsaved folder is created.

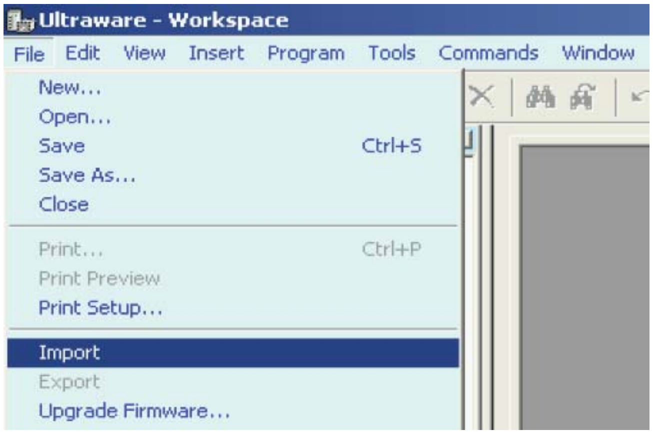


3. Select the Off-line: Unsaved folder.

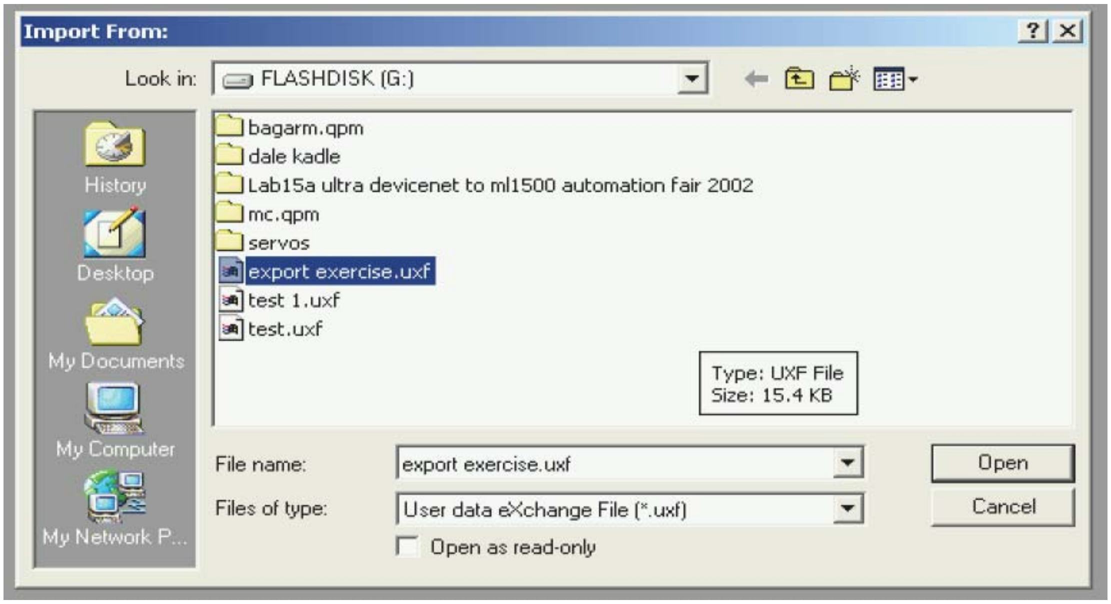


4. From the File menu, choose Import.

   If Import is grey and cannot be chosen, then you haven't selected the Off-line: Unsaved folder.



The Import From: dialog opens.



Look in: FLASHDISK (G:)

File name: export exercise.uxf

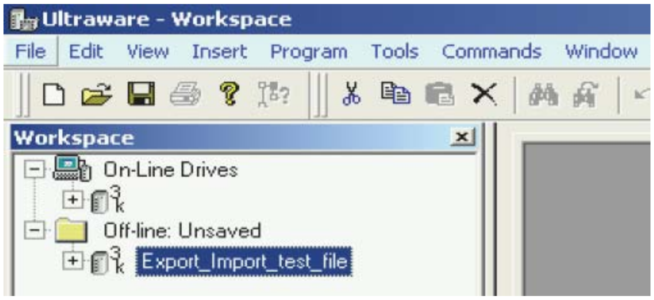
Files of type: User data eXchange File (*.uxf)

☐ Open as read-only

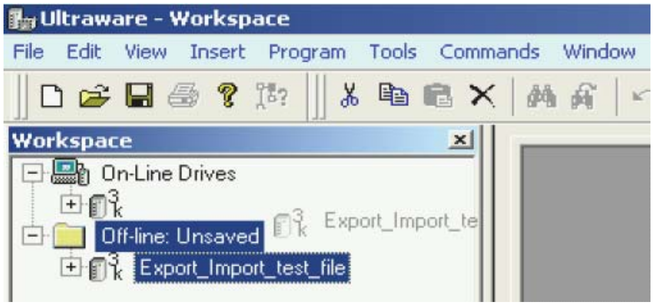
5. Choose the drive setup file that you saved earlier.

   In this example, export exercise.uxf is the file to import.

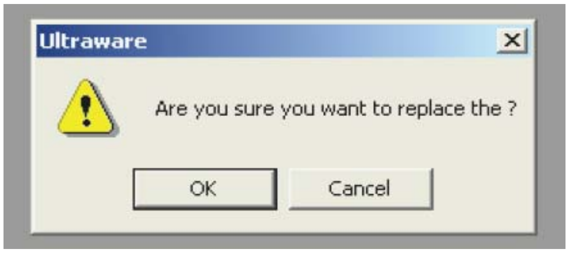
The drive setup file appears under the Off-line: Unsaved folder.



6. Drag and drop the drive setup file onto the online drive.



This Ultraware dialog opens.



Are you sure you want to replace the ?

7. Click OK.

The file loads and you are ready to restart the application.

Rockwell Automation Support

Rockwell Automation provides technical information on the Web to assist you in using its products. At http://support.rockwellautomation.com, you can find technical manuals, a knowledge base of FAQs, technical and application notes, sample code and links to software service packs, and a MySupport feature that you can customize to make the best use of these tools.

For an additional level of technical phone support for installation, configuration, and troubleshooting, we offer TechConnect Support programs. For more information, contact your local distributor or Rockwell Automation representative, or visit http://support.rockwellautomation.com.

Installation Assistance

If you experience a problem with a hardware module within the first 24 hours of installation, please review the information that's contained in this manual. You can also contact a special Customer Support number for initial help in getting your module up and running.

United States	1.440.646.3434 Monday – Friday, 8am – 5pm EST
Outside United States	Please contact your local Rockwell Automation representative for any technical support issues.

New Product Satisfaction Return

Rockwell tests all of its products to ensure that they are fully operational when shipped from the manufacturing facility. However, if your product is not functioning, it may need to be returned.

United States	Contact your distributor. You must provide a Customer Support case number (see phone number above to obtain one) to your distributor in order to complete the return process.
Outside United States	Please contact your local Rockwell Automation representative for return procedure.

www.rockwellautomation.com

Power, Control and Information Solutions Headquarters

Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
Europe/Middle East/Africa: Rockwell Automation, Vorstlaan/Boulevard du Souverain 36, 1170 Brussels, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia Pacific: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Publication 2098-IN005C-EN-P — March 2008
Supersedes Publication 2098-IN005B-EN-P — August 2004

ANEXO G. MP-Series Low-Inertia Brushless Servo Motors Installation Instructions.

Installation Instructions

MP-Series Low-inertia Brushless Servo Motors with 75 mm or Smaller Frame Sizes

Catalog Numbers MPL-A1510, MPL-A1520, MPL-A1530, MPL-A210, MPL-A220, MPL-A230, MPL-B1510, MPL-B1520, MPL-B1530, MPL-B210, MPL-B220, MPL-B230

Important User Information

Solid state equipment has operational characteristics differing from those of electromechanical equipment. Safety Guidelines for the Application, Installation and Maintenance of Solid State Controls, publication SGI-1.1, available from your local Rockwell Automation sales office or online at http://www.literature.rockwellautomation.com, describes some important differences between solid state equipment and hard-wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

In no event will Rockwell Automation, Inc. be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.

The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, Rockwell Automation, Inc. cannot assume responsibility or liability for actual use based on the examples and diagrams.

No patent liability is assumed by Rockwell Automation, Inc. with respect to use of information, circuits, equipment, or software described in this manual.

Reproduction of the contents of this manual, in whole or in part, without written permission of Rockwell Automation, Inc., is prohibited.

Throughout this manual, when necessary, we use notes to make you aware of safety considerations.

WARNING ⚠	Identifies information about practices or circumstances that can cause an explosion in a hazardous environment, which may lead to personal injury or death, property damage, or economic loss.
IMPORTANT	Identifies information that is critical for successful application and understanding of the product.
ATTENTION ⚠	Identifies information about practices or circumstances that can lead to personal injury or death, property damage, or economic loss. Attentions help you to identify a hazard, avoid a hazard, and recognize the consequences.
SHOCK HAZARD ⚠	Labels may be located on or inside the equipment, for example, a drive or motor, to alert people that dangerous voltage may be present.
BURN HAZARD ⚠	Labels may be located on or inside the equipment, for example, a drive or motor, to alert people that surfaces may be at dangerous temperatures.

About This Publication

This publication provides installation instructions for the MP-series low-inertia (MPL) motors with a frame size of 75 mm (2.95 in.) or smaller.

Use this document if you are responsible for installing these Allen-Bradley motor products. Please read all instructions before installing this motor.

Catalog Number Explanation

MP L - B 2 10 V - E J 4 2 A A

FACTORY DESIGNATED OPTIONS
A = Standard
MOUNTING FLANGE
A = IEC Metric
BRAKE
2 = No Brake
4 = 24V dc Brake
CONNECTORS
4 = Circular DIN, Right Angle, 180° Rotatable
ENCLOSURE/SHAFT KEY/SHAFT SEAL
J = IP66 Housing/Shaft Key/No Shaft Seal
FEEDBACK
E = Single-turn High Resolution Encoder
H = 2000 Line Encoder
V = Multi-turn High Resolution Encoder
RATED SPEED
P = 5000 rpm
T = 6000 rpm
U = 7000 rpm
V = 8000 rpm
MAGNET STACK LENGTH DESIGNATOR
10
20
30
FRAME SIZE (IEC 72-1 FLANGE NUMBER)
15 = 63 mm (2.48 in.)
2 = 75 mm (2.95 in.)
VOLTAGE RATING
A = 230V ac
B = 460V ac
SERIES TYPE
L = Low-inertia
SERIES

Before You Begin

Before unpacking the product, inspect the shipping carton for damage. If damage is visible, immediately contact the shipper and request assistance. Otherwise, proceed with unpacking.

Remove the motor carefully from its shipping container, and visually inspect the motor for any damage. Carefully examine the motor frame, front output shaft, and mounting pilot for any defects.

Keep the original packing material in case you need to return the product for repair or transport it to another location. Use both the inner and outer packing cartons to verify adequate protection for a unit returned for service.

ATTENTION ⚠

Do not attempt to open and modify the motor. Modifications that can be performed in the field are described in this manual, other changes should not be attempted.

Only a qualified Allen-Bradley employee can service this type of motor.

Failure to observe these safety procedures could result in personal injury or damage to equipment.

Installation and Maintenance Guidelines

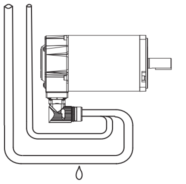
The guidelines in this section provide you with general information about installing servo motors. Instructions specific to MPL servo motor installation follow this section.

To Prolong Motor Life

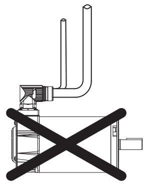
Thoughtful design and proper maintenance can increase the life of a servo motor. The following are guidelines to maximize the life of a servo motor.

- Always install the motor with any cable entry point positioned underneath the motor housing, and provide a drip loop in each cable. A drip loop is a downward bend in the cable that lets water gather and drip off the cable rather than continue to flow along the cable. These two installation practices greatly reduce the potential for moisture related problems, and are depicted in the illustration.

Recommended Connector Orientation for Drip Loop



Cable enters beneath the motor, and drip loop is formed.



Motor is positioned so cable enters from above.

Cable lacks drip loop.

- Avoid installing the motor with the shaft pointing upward, even if shaft seal in installed. This orientation increases the risk of contaminant ingress.
- Bearing contamination will shorten the life of a motor. The probability of this occurring can be significantly reduced by installing a shaft seal. Ingress Protection Codes (IP Ratings) for various mounting orientations are described in the Specifications section.
  - A shaft seal is recommended whenever the motor shaft is exposed to moisture and other fluids, including lubricating oil from a gearbox, or significant amounts of fine dust.
  - A shaft seal may be unnecessary if the motor shaft area is free of liquids or fine dust and a lower IP rating will suffice.
- The brake on these motors are a permanent magnet-type holding brake. The brake releases when voltage is applied. A separate 24V dc power source must be connected with proper polarity to disengage the brake. This power source may be applied by a servo motor controller, in addition to manual operator control.

  If system main power fails, holding brakes can withstand occasional use as stopping brakes. However, this is potentially damaging to the system, increases brake wear, and reduces brake life.

IMPORTANT

Holding brakes are not designed to stop rotation of the motor shaft, nor are they intended to be used as a safety device. They are designed to hold a motor shaft at 0 rpm for up to the rated brake holding torque.

The recommended method of preventing motor shaft rotation is first, command the servo drive to 0 rpm, second, verify the motor is at 0 rpm, third, engage the brake, and finally, disable the drive.

Disabling the drive also removes the potential for brake wear caused by a badly tuned servo system oscillating the shaft.

Mount the Motor

All MPL motors include a mounting pilot for aligning the motor on a machine. Preferred fasteners are stainless steel. The installation must comply with all local regulations and use of equipment and installation practices that promote electromagnetic compatibility and safety.

ATTENTION

Unmounted motors, disconnected mechanical couplings, loose shaft keys, and disconnected cables are dangerous if power is applied.

Disassembled equipment should be appropriately identified (tagged-out) and access to electrical power restricted (locked-out).

Before applying power to the motor, remove the shaft key and other mechanical couplings that could be thrown from the shaft.

Failure to observe these safety procedures could result in personal injury.

The dimensions and dimensional symbols for the different frame sizes and stack lengths in the MPL motors are referenced in the Product Dimensions drawing and tables.

Mechanical Connections

Mechanical connections to the motor shaft, such as couplings and pulleys, require a torsionally rigid coupling or a reinforced timing belt. The high dynamic performance of servo motors can cause couplings, pulleys, or belts to loosen or slip over time. A loose or slipping connection will cause system instability and may damage the motor shaft. All connections between the machine and the motor shaft must be rigid to achieve acceptable system response. Periodically inspect connections to verify their rigidity.

ATTENTION

Do not strike the shaft, key, couplings, or pulleys with tools during installation or removal.

Damage may occur to the motor bearings and the feedback device if sharp impact to the shaft is applied during installation of couplings and pulleys, or a shaft key. Damage to the feedback device also may result by applying leverage from the faceplate to remove devices mounted on the motor shaft.

Apply a constant pressure, with a wheel puller for example, to the user end of the shaft to remove a friction fit or a stuck device.

Failure to observe these safety procedures could result in damage to the motor and its components.

When mounting couplings or pulleys to the shaft, verify that the connections are properly aligned and that axial and radial loads are within the specifications of the motor.

Refer to Load Force Capacities for guidelines on how to achieve 20,000 hours of motor bearing life.

A shaft key provides a rigid mechanical connection with the potential for self-alignment, but the key must be properly installed in the keyway. Refer to Product Dimensions for dimensional information about the key and shaft keyway.

Instructions for removing or installing a shaft key are provided in Shaft Key Removal and Installation.

Connector Orientation

The housings for the feedback and power and brake connectors can be rotated once up to 180 degrees. This allows either connector to face down if the motor is installed in a vertical application, rearward if connector access is restricted in a horizontal application, or to the side of the motor.

ATTENTION

Connectors are designed to be rotated into a fixed position during installation of the motor, and remain in that position without further adjustment. Strictly limit the applied forces and the number of times the connector is rotated to be sure that connectors meet the requirements of IP66.

Failure to observe these safety procedures could result in damage to the motor and its components.

To rotate the connectors:

1. Mount and fully seat a mating cable on the connector.
2. Grasp both connectors by their housings and slowly rotate them to the outside of the motor. If necessary, repeat this step for each connector (feedback or power/brake).

Only apply force to the connectors; do not apply force to the cable. No tools (for example, pliers and vise-grips) should be used to assist with the rotation of the connector.

Interconnect Cables

Knowledgeable cable routing improves system ElectroMagnetic Compatibility (EMC). Refer to the Shield the Power Cable section for suggested grounding techniques.

To install cables properly, observe these guidelines.

- Do not connect feedback cable or power and brake cable while power is applied.

ATTENTION

Do not connect or disconnect the motor feedback cable, or the power and brake cable while power is applied to them.

Inadvertent pin connections may result in unexpected motion or result in irreversible damage to the components.

- Keep wire lengths as short as physically possible.
- Separate cables by 0.3 m (1 ft) minimum for every 9 m (30 ft) of parallel run.
- Route signal cables that transmit encoder, serial, or analog data at low voltages away from the motor and power wiring.
- Ground both ends of the cable shield and twist the signal wire pairs to prevent electromagnetic interference from other equipment.

ATTENTION

High voltage can be present on the shields of a power cable, if the shields are not grounded.

Verify there is a connection to ground for all shields in the power cable.

Failure to observe these safety procedures could result in personal injury or damage to equipment.

Electrical Noise

ElectroMagnetic Interference (EMI), commonly called noise, may adversely impact motor performance by inducing stray signals. Effective techniques to counter EMI include filtering the ac power, shielding and separating signal carrying lines, and practicing good grounding techniques. Effective ac power filtering can be achieved by using isolated ac power transformers or properly installed ac line filters.

To reduce EMI, observe these guidelines.

- Physically separate signal lines from motor cabling and power wiring. Do not route signal wires with motor and power wires, and do not route signal wires over the vent openings of servo drives or other electrical power sources.
- Ground all equipment using a single-point parallel ground system that employs ground bus bars or large straps. If necessary, use additional electrical noise reduction techniques to reduce EMI in noisy environments.

Shield the Power Cable

To shield the power cable:

1. Verify the separate signal wire shield connects to the overall chassis ground by looping back each of the signal wire pairs as shown in the diagram.
2. Clamp all three shields together at the power cable (chassis) ground connection on the drive.

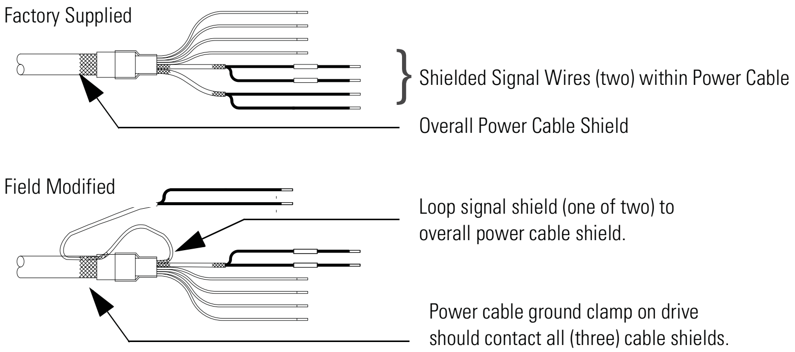
ATTENTION

High voltage can be present on the shields of a power cable, if the shields are not grounded.

Verify there is a connection to ground for all shields in the power cable.

Failure to observe these safety procedures could result in personal injury or damage to equipment.

Power Cable Shielding

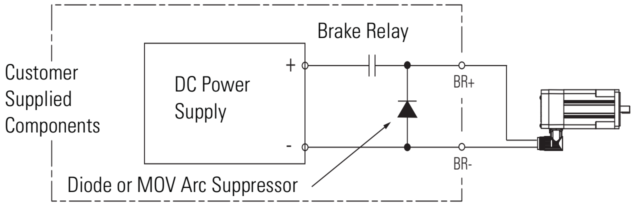


Brake Control and Power Regulation

The dc power source for a permanent magnet brake, such as that on the MP-Series servo motor with 75 mm (2.95 in.) or smaller frame, requires a dc power supply with low ripple voltage.

A motor brake requires relay contacts to open and remove power to the brake coil. Removing power causes the brake to mechanically engage, but it also may cause electrical arcing to occur at the contacts until the residual brake power sufficiently dissipates. A customer supplied diode is recommended to prevent electrical arcing at the brake relay contacts. Substituting a metal oxide varistor (MOV) for the diode can reduce the time to mechanically engage the brake. The following diagram shows typical customer supplied components for brake control, including an arc suppressing diode or MOV.

Arc Suppression in the Motor Brake Circuit



The Kinetix 6000 and Kinetix 7000 drives from Rockwell Automation have a brake control relay that includes a MOV arc suppressor. Customer supplied arc suppression is not required in this case, unless power consumption by the brake requires an external brake relay.

Shaft Seals

An optional shaft seal is installed on the motor shaft to protect the front bearing from fluids or fine dust that could contaminate the motor bearing and reduce its lifetime. An IP66 rating for the motor depends on the usage of shaft seals and environmentally sealed connectors/cables.

- Refer to the Specifications for brief descriptions of IP ratings.
- Refer to the Kinetix Motion Control Selection Guide, publication GMC-SG001, to find the catalog numbers of seal kits available for your motor.

Install the MPL Motors

Observe the following when installing an MPL motor.

ATTENTION

⚠

Do not strike the shaft, couplings, or pulleys with tools during installation or removal.

Damage may occur to the motor bearings and the feedback device if sharp impact to the shaft is applied during installation of couplings and pulleys.

Failure to observe these safety procedures could result in damage to the motor and its components.

1. Allow sufficient clearances in the area of the motor for it to stay within its specified operating temperature range. Refer to the Specifications for the operating temperature range.

   Do not install the motor in an area with restricted airflow. Keep other heat producing devices away from the motor.

   To obtain the specified motor thermal rating, mount the motor so the heat dissipation is at a minimum equivalent to the following surface.

For Motor	Aluminum Heatsink Dimensions, Approx.
MPL-*x*15*xx*	203.2 x 203.2 x 6.35 mm (8 x 8 x 0.25 in.)
MPL-*x*2*xx*	254.0 x 254.0 x 6.35 mm (10 x 10 x 0.25 in.)

ATTENTION

⚠

Outer surfaces of the motor can reach high temperatures, 125 °C (257 °F), during motor operation.

Take precautions to prevent accidental contact with hot surfaces. Consider motor surface temperature when selecting motor mating connections and cables.

Failure to observe these safety procedures could result in personal injury or damage to equipment.

2. Position the motor with the cable connections beneath the motor.

   Refer to the Recommended Connector Orientation for Drip Loop section for a visual reference of correct motor and cable positioning.

3. Properly mount and align the motor.
   - All MPL motors include a mounting pilot for aligning the motor on a machine.
   - The index pulse occurs on a 2000 line encoder when the shaft key is aligned with the connectors.
   - Make sure belt loading is within the motor limits, and all belts and pulleys are properly aligned.
   - Follow manufacturer recommendations for gearboxes, pulleys, or other motor accessories.

4. Form a drip loop in the cables directly before each cable attaches to the motor. Refer to the Recommended Connector Orientation for Drip Loop diagram for a visual example.

ATTENTION

Be sure that cables are installed and restrained to prevent uneven tension or flexing at the cable connectors.

Excessive and uneven lateral force at the cable connectors may result in the connector's environmental seal opening and closing as the cable flexes.

Failure to observe these safety procedures could result in damage to the motor and its components.

---

5. Attach the feedback cable, and the combination power and brake cable to the motor.

ATTENTION

Do not connect or disconnect the motor feedback cable, or the power and brake cable while power is applied to them.

Inadvertent pin connections may result in unexpected motion or result in irreversible damage to the components.

---

a. Carefully align each cable connector with the respective motor connector as shown in the diagram.

b. Do not apply excessive force when mating the cable and motor connectors. If the connectors do not go together with light hand force, realign and try again.

c. Hand tighten the knurled collar five to six turns to fully seat each connector.

ATTENTION

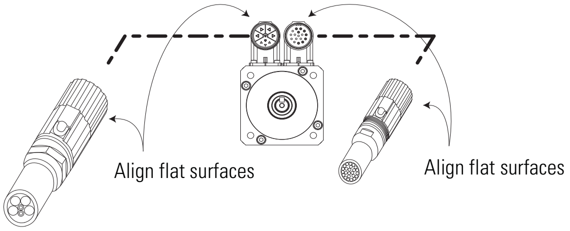
Keyed connectors must be properly aligned and hand-tightened the recommended number of turns.

Improper alignment is indicated by the need for excessive force, such as the use of tools, to fully seat connectors.

Connectors must be fully tightened for connector seals to be effective.

Failure to observe these safety procedures could result in damage to the motor, cables, and connector components.

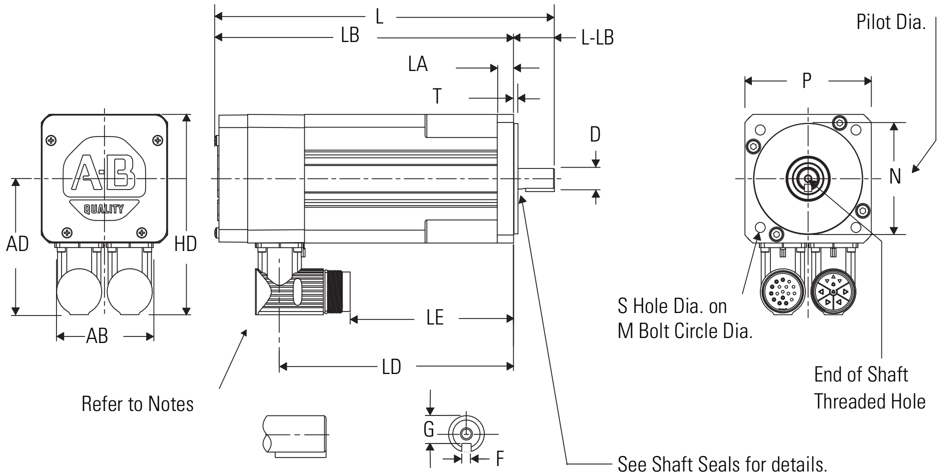
Motor Connector Alignment



Brake and Power Connector

Feedback Connector

Product Dimensions



Notes:
1. Connectors may rotate 180° outward once.
2. LD measures to rotational center of connectors.
3. LE measures to front of connector.

Shaft Key
MPL-*x*15*xx* = 3 x 3 x 14
MPL-*x*2*xx* = 4 x 4 x 16

MPL motors are designed to metric dimensions. Inch dimensions are mathematical conversions from millimeters. Untoleranced dimensions are for reference.

The dimensions in the table are for non-brake motors. Footnotes provide the additional dimensions for the brake option and tolerances for dimensions.

MPL Servo Motor Dimensions, Approx.

MPL-A or MPL-B	**AB** **mm (in.)**	**AD** **mm (in.)**	**D** [1] **mm (in.)**	**HD** **mm (in.)**	**L** [2], [3] **mm (in.)**	**L-LB** [2] **mm (in.)**	**LA** **mm (in.)**	**LB** [3] **mm (in.)**	**LD** [3] **mm (in.)**	**LE** [3] **mm (in.)**
1510	53.9 (2.12)	66.5 (2.62)	9.0 (0.3543)	94.0 (3.70)	133.2 (5.25)	19.7 (0.776)	9.0 (0.35)	113.5 (4.47)	78.3 (3.08)	39.2 (1.54)
1520					146.2 (5.76)			126.5 (4.98)	91.3 (3.60)	52.4 (2.06)
1530					171.2 (6.74)			151.5 (5.96)	116.3 (4.58)	77.2 (3.04)

MPL Servo Motor Dimensions, Approx.

MPL-A or MPL-B	**AB** mm (in.)	**AD** mm (in.)	**D** (1) mm (in.)	**HD** mm (in.)	**L** (2), (3) mm (in.)	**L-LB** (2) mm (in.)	**LA** mm (in.)	**LB** (3) mm (in.)	**LD** (3) mm (in.)	**LE** (3) mm (in.)
210	53.9 (2.12)	74.0 (2.91)	11.0 (0.4331)	109.0 (4.29)	137.3 (5.40)	22.7 (0.894)	9.0 (0.35)	114.6 (4.51)	78.6 (3.09)	39.6 (1.56)
220					162.8 (6.41)			140.1 (5.52)	104.1 (4.10)	65.1 (2.56)
230					188.3 (7.41)			165.6 (6.52)	129.6 (5.10)	90.6 (3.57)

(1) Tolerance for this dimension is: MPL-*x*15*xx* +0.007, -0.002 (+0.0003, -0.0001), and MPL-*x*2*xx* +0.008, -0.003 (+0.0003, -0.0001).

(2) Tolerance for this dimension is ±0.7 (±.0.028).

(3) If ordering an MPL-x15xx motor with brake, add 36.1 mm 1.421 (in.) to L and LB, and 33.4 mm (1.32 in.) to LD and LE.
If ordering an MPL-x2xx motor with brake, add 39.0 mm (1.535 in.) to L and LB, and 24.7 mm (0.97 in.) to LD and LE.

MPL Servo Motor Dimensions, Approx.

MPL-A or MPL-B	**M** mm (in.)	**N** (1) mm (in.)	**P** mm (in.)	**S** (2) mm (in.)	**T** mm (in.)	**F** (3) mm (in.)	**G** (4) mm (in.)	**End of Shaft** (5)
1510	63.0 (2.480)	40.0 (1.575)	55.0 (2.17)	5.8 (0.228)	2.50 (0.098)	3.0 (0.118)	7.2 (0.283)	M3 x 0.5 - 6H x 9.0 (0.35)
1520								
1530								
210	75.0 (2.953)	60.0 (2.362)	70.0 (2.76)		2.5 (0.098)	4.0 (0.157)	8.5 (0.335)	M4 x 0.7 - 6H x 10.0 (0.39)
220								
230								

(1) Tolerance for this dimension is: MPL-*x*15*xx* +0.011, -0.005 (+0.0004, -0.0002); and MPL-*x*2*xx* +0.012, -0.007 (+0.0005, -0.0003).

(2) Tolerance for this dimension is: MPL-*x*15*xx* and MPL-*x*2*xx* +0.3, -0.0 (+0.012, -0.0).

(3) Tolerance for this dimension is: MPL-*x*15*xx* -0.004, -0.029 (-0.00016, -0.00114); and MPL-*x*2*xx* +0.0, -0.03 (+0.0, -0.0012).

(4) Tolerance for this dimension is: +0.0, -0.1 (+0.0, -0.0004).

(5) Metric (M) threading dimensions include major diameter (mm) X thread pitch - tolerance class X thread depth in mm and (in.).

Motor Connectors

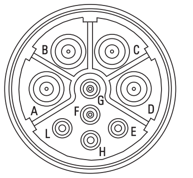
The tables below contain connector pin descriptions for the feedback connector, and the combined power and brake connector.

Feedback

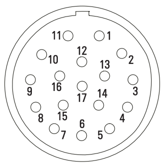
Pin	High Resolution Encoder MPL-A*xxx* (230V)	High Resolution Encoder MPL-B*xxx* (460V)	Incremental Encoder MPL-A/B*xxxx*-H*xxxx*
1	Sin+	Sin+	A+
2	Sin-	Sin-	A-
3	Cos+	Cos+	B+
4	Cos-	Cos-	B-
5	Data+	Data+	I+
6	Data-	Data-	I-
7	Reserved	Reserved	Reserved
8			
9	+5V dc		+5 V dc
10	Common		Common
11	Reserved	+9V dc	Reserved
12		Common	
13	TS+	TS+	TS+
14	TS-	TS-	TS-
15	Reserved	Reserved	S1
16			S2
17			S3

Power and Brake

Pin	MPL-A*xxx* and MPL-B*xxx*
A	Phase U (1)
B	Phase V (1)
C	Phase W (1)
D	Ground
E	Reserved
F	BR+
G	BR-
H	Reserved
L	



Intercontec P/N
BEDC090NN00000005000



Intercontec P/N
AEDC113NN00000012000

---

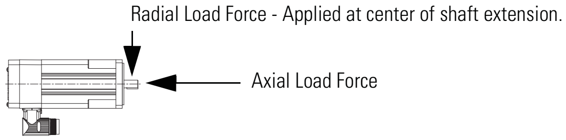
(1) The U, V and W power phases may be labelled as R, S, and T respectively.

Load Force Capacities

Motors are capable of operating with a sustained shaft load. The location of the radial and axial load force is shown in the figure, and maximum values are listed.

Loads are measured in pounds, kilograms are mathematical conversions.

Load Forces on Shaft



The following tables represent 20,000 hour L10 bearing fatigue life at various loads and speeds. This 20,000 hour life does not account for possible application-specific life reduction that may occur due to bearing grease contamination from external sources.

Radial Load Force Ratings

Motor	1000 rpm		2000 rpm		3500 rpm		4000 rpm		at Max Speed		
	kg	(lb)	kg	(lb)	kg	(lb)	kg	(lb)	kg	(lb)	rpm
MPL-A/B1510	24	(52)	19	(41)	—	—	15	(33)	12	(26)	8000
MPL-A/B1520	25	(56)	20	(45)	—	—	16	(36)	14	(30)	7000
MPL-A/B1530	28	(62)	22	(49)	—	—	18	(39)	15	(32)	7000
MPL-A/B210	24	(52)	19	(41)	—	—	15	(32)	12	(26)	8000
MPL-A/B220	27	(59)	21	(47)	18	(39)	—	—	15	(33)	6000
MPL-A/B230	29	(64)	23	(51)	19	(42)	—	—	17	(37)	5000

Axial Load Force Ratings with Maximum Radial Load

Motor	1000 rpm		2000 rpm		3500 rpm		4000 rpm		at Max Speed		
	kg	(lb)	kg	(lb)	kg	(lb)	kg	(lb)	kg	(lb)	rpm
MPL-A/B1510	15	(33)	10	(22)	—	—	7	(15)	5	(10)	8000
MPL-A/B1520	14	(31)	10	(22)	—	—	6	(13)	4	(9)	7000
MPL-A/B1530	13	(29)	9	(20)	—	—	6	(13)	4	(8)	7000
MPL-A/B210	15	(33)	10	(22)	—	—	7	(15)	5	(10)	8000
MPL-A/B220	14	(30)	9	(20)	7	(15)	—	—	5	(10)	6000
MPL-A/B230	13	(28)	9	(19)	6	(13)	—	—	5	(10)	5000

Axial Load Force Ratings with Zero Radial Load

Motor	1000 rpm		2000 rpm		3500 rpm		4000 rpm		at Max Speed		
	kg	(lb)	kg	(lb)	kg	(lb)	kg	(lb)	kg	(lb)	rpm
MPL-A/B1510	24	(53)	17	(37)	—	—	12	(26)	8	(18)	8000
MPL-A/B1520	24	(53)	17	(37)	—	—	12	(26)	9	(19)	7000
MPL-A/B1530	24	(53)	17	(37)	—	—	12	(26)	9	(19)	7000
MPL-A/B210	24	(53)	17	(37)	—	—	12	(26)	8	(18)	8000
MPL-A/B220	24	(53)	17	(37)	13	(28)	—	—	10	(22)	6000
MPL-A/B230	24	(53)	17	(37)	13	(28)	—	—	10	(22)	5000

Troubleshooting and Maintenance

Standard troubleshooting and maintenance for this motor includes the following activities.

ATTENTION

Do not strike the shaft, key, couplings, or pulleys with tools during installation or removal of any device.

Damage may occur to the motor bearings and the feedback device if sharp impact to the shaft is applied during installation of couplings and pulleys, or a shaft key. Damage to the feedback device also may result by applying leverage from the faceplate to remove devices mounted on the motor shaft.

Apply a constant pressure, with a wheel puller, to the user end of the shaft to remove any friction fit or stuck device from the motor shaft.

Failure to observe these safety procedures could result in damage to the motor and its components.

Shaft Key Removal and Installation

Shaft keys are toleranced for an interference fit (slightly larger than the opening) to provide a secure and rigid fit for the mating connection.

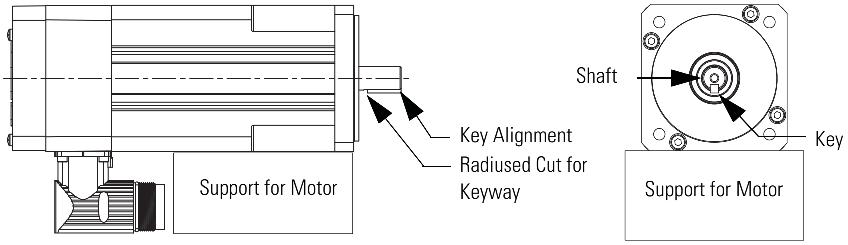
To Remove a Shaft Key

Lift the key by grasping it with a plier or similar tool, or lever the key with a flat-blade screwdriver inserted between the key and the slot.

To Install a Shaft Key

1. Verify the replacement key matches the keyway in the shaft and the mating mechanical connection (for example, a coupling or pulley) before proceeding.
2. Support the motor to prevent movement.
3. Align the front of the key with the front of the motor shaft. This prevents a radiused end-of-cut at the motor end of the keyway from interfering with correct seating of the key.
4. Apply a constant force across the exposed surface of the key with a controlled press device (for example, a screw clamp) to push the key into the shaft.

Shaft Key Installation



Motor Cables and Accessory Kits

Accessories available from the factory include motor cables and shaft seals.

Motor Cables

Factory manufactured feedback and power cables are available in standard cable lengths. They provide the sealing needed to achieve environmental ratings and shield termination.

Refer to the Kinetix Motion Control Selection Guide, publication GMC-SG001, for the catalog numbers of cables appropriate for these motors.

Shaft Seals

Shaft seals provide environmental sealing of MPL motors. Shaft seals provide an additional barrier to moisture and particle intrusion to the motor bearings. Motors are shipped without a shaft seal, but this option is easily installed or replaced in the field.

MPL shaft seals have a Nitrile contact surface. Shaft seals require a lubricant provided in the kit.

Shaft Seal Dimensions, Approx.

Motor	Catalog No.	Inside Diameter	Outside Diameter	Width
MPL-*x*15*xx* or MPL-*x*2*xx*	MPL-SSN-F63/F75	12.0 mm (0.47 in.)	24.0 mm (0.95 in.)	7.0 mm (0.28 in.)

Refer to the Shaft Seal Installation Instructions, publication 2090-IN012, for detailed installation instructions.

Specifications

Always store a motor in a clean and dry location within these environmental conditions. Appropriate mounting, cabling, and a shaft seal may be required to attain a specific IP rating. Exterior surfaces of the MP-Series small frame servo motors are made from these materials.

MP-Series Low-inertia Motors with 75 mm or Smaller Frame Size

Attribute	Value
Temperature, Operating	0…40 °C (32…104 °F)
Temperature, Storage	-30…70 °C (-22…158 °F)
Relative Humidity	5…95% noncondensing
Atmosphere	Noncorrosive
Shock	20g peak, 6 ms duration
Vibration	2.5g, 30 to 2000 Hz
Ingress Protection Codes (1) (2)	
IP50	Motor without a shaft seal, mounted in the shaft up direction
IP51	Motor without a shaft seal, mounted in the shaft horizontal direction
IP53	Motor without a shaft seal, mounted in the shaft down direction
IP66 (3)	Motor with an optional shaft seal
Material by Location	
Housing	Aluminum
Shaft	39NiCrMo3 or 1144 steel (respectively per UNI7845 or ASTM-A311 Class B)
Shaft Key	Carbon steel

(1) IP ratings are determined using Rockwell Automation cables to connect to the motor.

(2) IP rating applies to a motor mounted as described.

(3) This rating is for dust tightness and powerful water jets ejecting water, not cleaning agents.

Additional Resources

For additional information about motors and compatible Rockwell Automation drives, refer to these publications.

For Information About	Read This Publication	Publication Number
MPL-series Brushless Motors	Motor Installation Instructions	MP-IN001, MP-IN002, MP-IN003, MP-IN004, and MP-IN005
Ultra3000 Digital Servo Drives	Ultra3000 Installation Instructions	2098-IN003
Ultra5000 Intelligent Positioning Drives	Ultra5000 Installation Instructions	2098-IN001
Kinetix 6000 Drives	Kinetix 6000 Installation Instructions	2094-IN001
1394 SERCOS Drive	1394 SERCOS Installation Instructions	1394-IN002
Installing and Removing Shaft Seals	Shaft Seal Installation Instructions	2090-IN012
A glossary of industrial automation terms and abbreviations	Rockwell Automation Industrial Automation Glossary	AG-7.1
How to minimize and control system-level noise.	System Design for Control of Electrical Noise Reference Manual	GMC-RM001
An overview of Allen-Bradley motion controls and systems, including information about the this and other motors.	Kinetix Motion Control Selection Guide	GMC-SG001

These publications are available from your local Rockwell Automation office. At the website http://www.literature.rockwellautomation.com you may download an electronic version of these publications.

Rockwell Automation Support

Rockwell Automation provides technical information on the Web to assist you in using its products. At http://support.rockwellautomation.com, you can find technical manuals, a knowledge base of FAQs, technical and application notes, sample code and links to software service packs, and a MySupport feature that you can customize to make the best use of these tools.

For an additional level of technical phone support for installation, configuration, and troubleshooting, we offer TechConnect Support programs. For more information, contact your local distributor or Rockwell Automation representative, or visit http://support.rockwellautomation.com.

Installation Assistance

If you experience a problem with a hardware module within the first 24 hours of installation, please review the information that's contained in this manual. You can also contact a special Customer Support number for initial help in getting your module up and running.

United States	1.440.646.3223 Monday – Friday, 8am – 5pm EST
Outside United States	Please contact your local Rockwell Automation representative for any technical support issues.

New Product Satisfaction Return

Rockwell tests all of its products to ensure that they are fully operational when shipped from the manufacturing facility. However, if your product is not functioning, it may need to be returned.

United States	Contact your distributor. You must provide a Customer Support case number (see phone number above to obtain one) to your distributor in order to complete the return process.
Outside United States	Please contact your local Rockwell Automation representative for return procedure.

www.rockwellautomation.com

Power, Control and Information Solutions Headquarters

Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
Europe/Middle East/Africa: Rockwell Automation, Vorstlaan/Boulevard du Souverain 36, 1170 Brussels, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia Pacific: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Publication MP-IN006B-EN-P - June 2006 PN 0013-2073-02
Supersedes Publication MP-IN006A-EN-P - December 2005

ANEXO H. AC Line Filters-Installation Instructions.

Installation Instructions

AC Line Filters

Description	Cat. No.
Single-phase (250...300V ac)	2090-UXLF-106
	2090-UXLF-110
	2090-UXLF-123
	2090-UXLF-132
	2090-UXLF-136
	2090-XXLF-TC116
Three-phase (250V ac)	2090-UXLF-336
	2090-UXLF-350

Description	Cat. No.	Cat. No.
Three-phase (500...520V ac)	2090-UXLF-HV323	2090-XXLF-TC316
	2090-UXLF-HV330	2090-XXLF-TC350
	2090-UXLF-HV350	2090-XXLF-TC365
	2090-XXLF-X330B	2090-XXLF-TC3100
	2090-XXLF-375	2090-XXLF-TC3150
	2090-XXLF-375B	2090-XXLF-TC3200
	2090-XXLF-3100	2090-XXLF-TC3250

About This Publication

This publication contains installation instructions for ac line filters used with Rockwell Automation servo drive systems. Included are mounting and wiring instructions, and mounting dimensions.

For product specifications, refer to the Kinetix Motion Control Selection Guide, publication GMC-SG001.

Important User Information

Solid state equipment has operational characteristics differing from those of electromechanical equipment. Safety Guidelines for the Application, Installation and Maintenance of Solid State Controls (publication SGI-1.1 available from your local Rockwell Automation sales office or online at http://literature.rockwellautomation.com) describes some important differences between solid state equipment and hard-wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

In no event will Rockwell Automation, Inc. be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.

The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, Rockwell Automation, Inc. cannot assume responsibility or liability for actual use based on the examples and diagrams.

No patent liability is assumed by Rockwell Automation, Inc. with respect to use of information, circuits, equipment, or software described in this manual.

Reproduction of the contents of this manual, in whole or in part, without written permission of Rockwell Automation, Inc., is prohibited.

Throughout this manual, when necessary, we use notes to make you aware of safety considerations.

WARNING ⚠	Identifies information about practices or circumstances that can cause an explosion in a hazardous environment, which may lead to personal injury or death, property damage, or economic loss.
IMPORTANT	Identifies information that is critical for successful application and understanding of the product.
ATTENTION ⚠	Identifies information about practices or circumstances that can lead to personal injury or death, property damage, or economic loss. Attentions help you to identify a hazard, avoid a hazard, and recognize the consequences.
SHOCK HAZARD ⚠	Labels may be on or inside the equipment, for example, a drive or motor, to alert people that dangerous voltage may be present.
BURN HAZARD ⚠	Labels may be on or inside the equipment, for example, a drive or motor, to alert people that surfaces may reach dangerous temperatures.

Before You Begin

For general guidelines when laying out your panel and mounting your ac line filter, refer to the System Design for Control of Electrical Noise Reference Manual, publication GMC-RM001. For guidelines specific to your application, refer to Additional Resources on page 11 for the user manual available for your servo drive.

ATTENTION

To avoid personal injury or damage to equipment due to hazardous voltages, follow these guidelines when installing your ac line filter. NEC and local regulations always take precedence.

- Disconnect mains power prior to installation.
- Verify that the rated voltage is compatible with the local supply voltage.
- Connect the earth ground connection first when making connections.

IMPORTANT

Remove corrosion, oil, or grease from bus bar mountings by scrubbing surfaces with a dry scrubbing pad. Do not touch cleaned surfaces with bare hands to prevent further contamination.

TIP

The Bulletin 2094 mounting brackets, catalog number 2094-XNBRKT-1, are designed to save panel space by letting you mount the Kinetix 6000 power rail or line interface module (LIM) over the ac line filter.

Refer to the 2094 Mounting Brackets Installation Instructions, publication 2094-IN008, for more information.

Installing the AC Line Filter

Follow these steps to install your ac line filter.

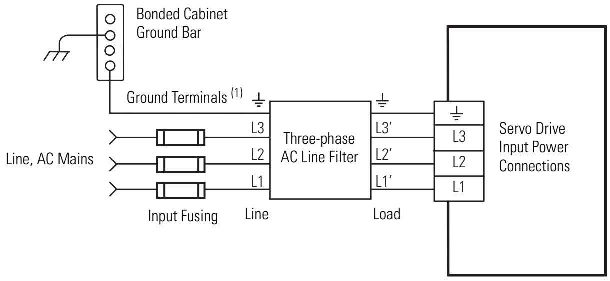
1. Determine the location for your ac line filter on the panel.

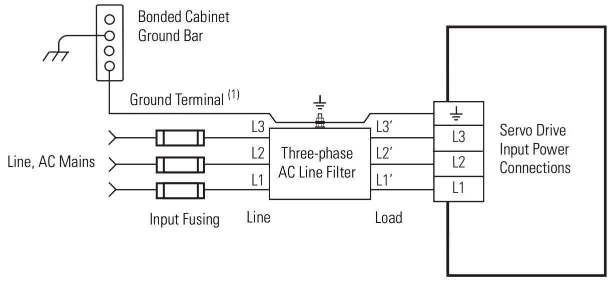
   Refer to Additional Resources on page 11 for the user manual with specific information regarding your servo-drive system layout. Position the ac line filter as close to the drive as possible.

2. Mount the ac line filter on the panel.

   Refer to Product Dimensions on page 5 for mounting dimensions. Use proper high-frequency (HF) bonding techniques to improve overall system performance.

3. Connect the ac line filter as shown in these examples.





[1] Ground terminal designators could also be PE/PE' or E/E'.

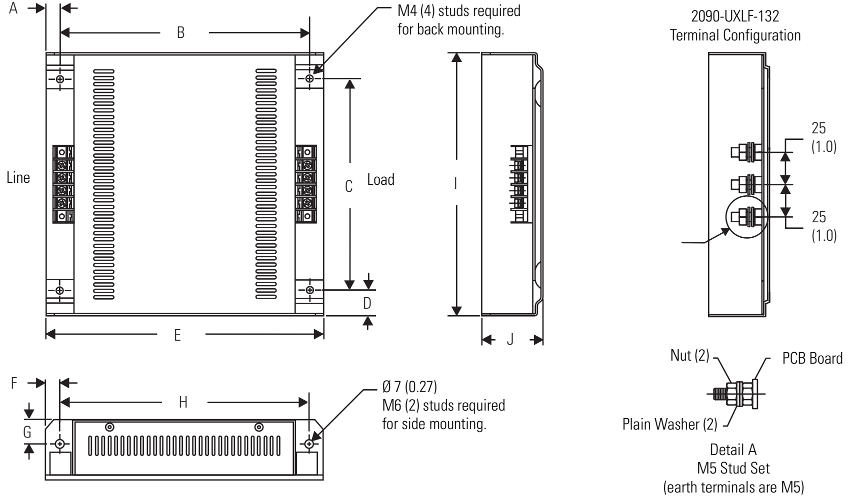
These examples show a three-phase ac line filter. Single-phase ac line filter installation is similar. For additional wiring information, refer to the appropriate user manual for interconnect diagrams specific to your servo drive application.

Product Dimensions

Mounting dimensions for ac line filters are given on the following pages.

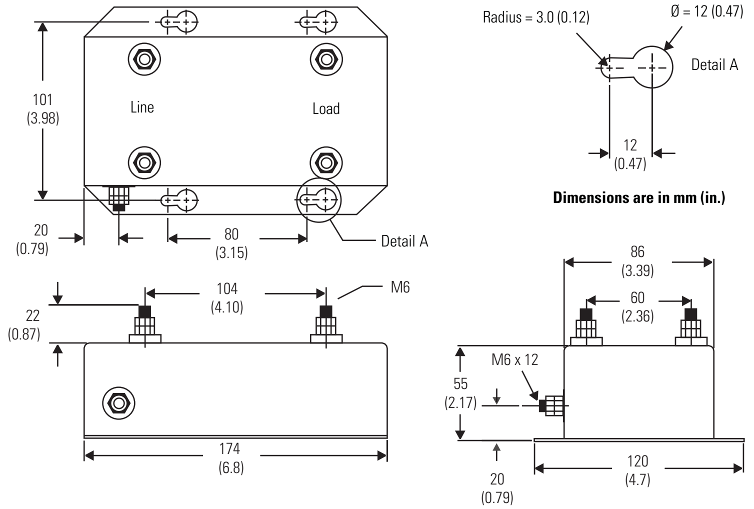
AC Line Filter Dimensions
2090-UXLF-106, 2090-UXLF-110, 2090-UXLF-123, 2090-UXLF-132, 2090-UXLF-HV323

Dimensions are in mm (in.)



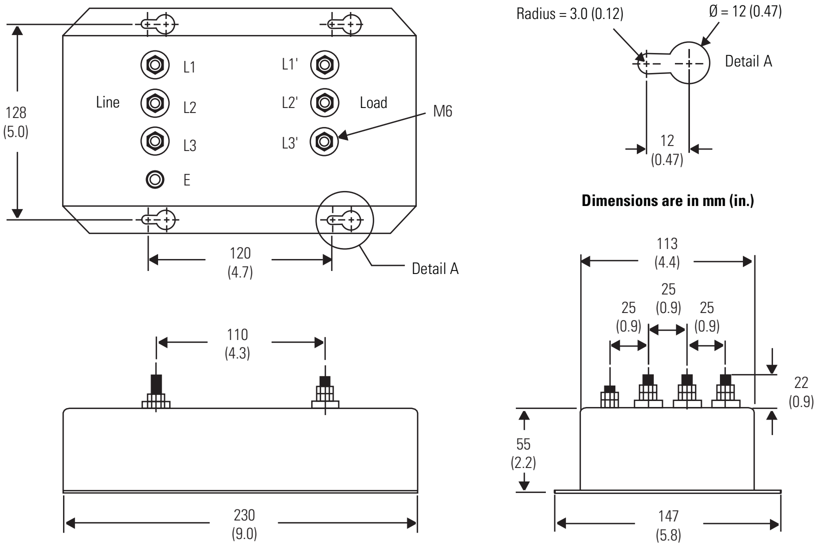
Cat. No.	A	B	C	D	E	F	G	H	I	J
	mm (in.)									
2090-UXLF-106	9.0 (0.35)	152.0 (5.99)	55.0 (2.17)	18.0 (0.71)	170.0 (6.69)	9.0 (0.35)	10.0 (0.39)	152.0 (5.99)	92.0 (3.62)	25.0 (0.98)
2090-UXLF-110	11.0 (0.43)	192.0 (7.56)	104.0 (4.0)	20.0 (0.79)	214.0 (8.42)	11.0 (0.43)	16.0 (0.63)	192.0 (7.56)	145.0 (5.71)	40.0 (1.58)
2090-UXLF-123			164.0 (6.46)				19.0 (0.75)		204 (8.04)	47.0 (1.85)
2090-UXLF-132										
2090-UXLF-HV323										

AC Line Filter Dimensions
2090-UXLF-136



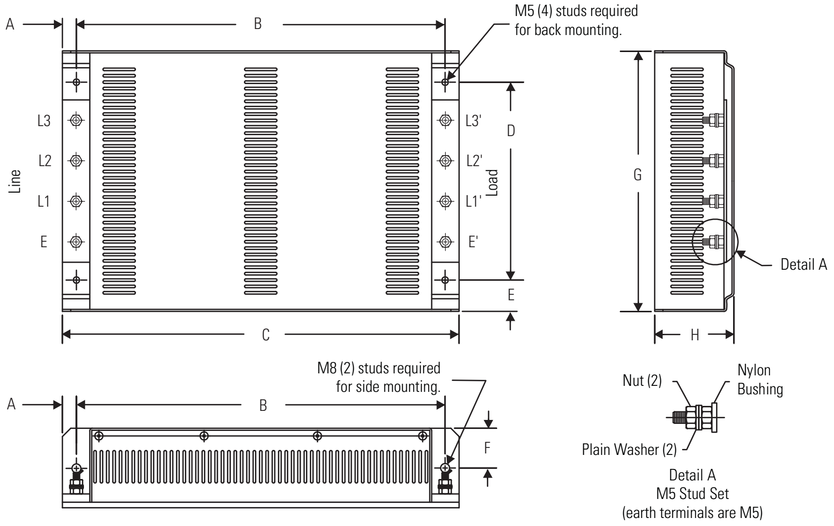
Dimensions are in mm (in.)

AC Line Filter Dimensions
2090-UXLF-336 and 2090-UXLF-350



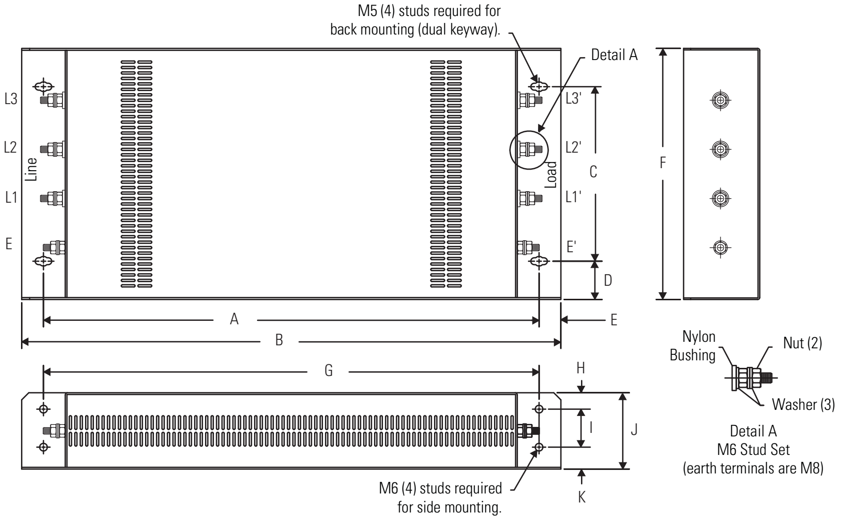
Dimensions are in mm (in.)

AC Line Filters Dimensions
2090-UXLF-HV330 and 2090-XXLF-X330B



Cat. No.	A	B	C	D	E	F	G	H
	mm (in.)							
2090-UXLF-HV330	11.0 (0.4)	338 (13.3)	360 (14.2)	145 (5.7)	29.5 (1.1)	16.0 (0.63)	204 (8.0)	40.0 (1.6)
2090-XXLF-X330B	15.0 (0.6)	330 (13.0)		155 (6.1)	20.0 (0.8)	32.5 (1.3)	195 (7.7)	65.0 (2.5)

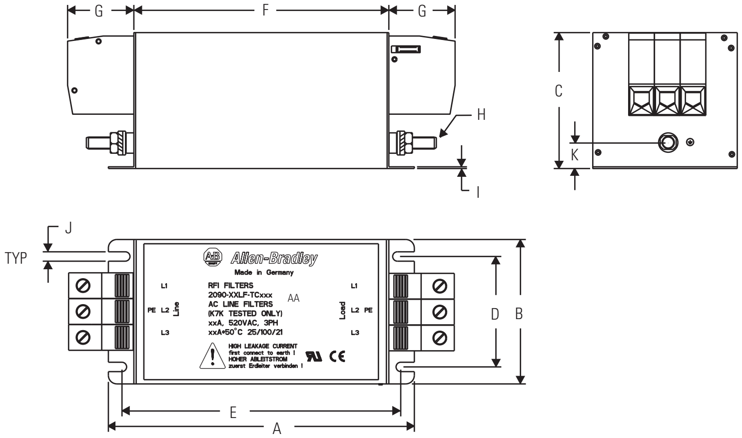
AC Line Filter Dimensions
2090-UXLF-HV350, 2090-XXLF-375, 2090-XXLF-375B, and 2090-XXLF-3100



Cat. No.	A	B	C	D	E	F
	mm (in.)					
2090-UXLF-HV350	578 (22.7)	618 (24.3)	160 (6.3)	35 (1.4)	20 (0.8)	230 (9.0)
2090-XXLF-375 2090-XXLF-375B	646 (25.4)	686 (27.0)	192 (7.5)			262 (10.3)
2090-XXLF-3100	741 (29.2)	785 (30.9)	215 (8.4)	30 (1.2)	21.5 (0.85)	275 (10.8)

Cat. No.	G	H	I	J	K
	mm (in.)				
2090-UXLF-HV350	578 (22.7)	15 (0.6)	35 (1.4)	70 (2.7)	20 (0.8)
2090-XXLF-375 2090-XXLF-375B	646 (25.4)				
2090-XXLF-3100	741 (29.2)		47 (1.8)	80 (3.1)	18 (0.7)

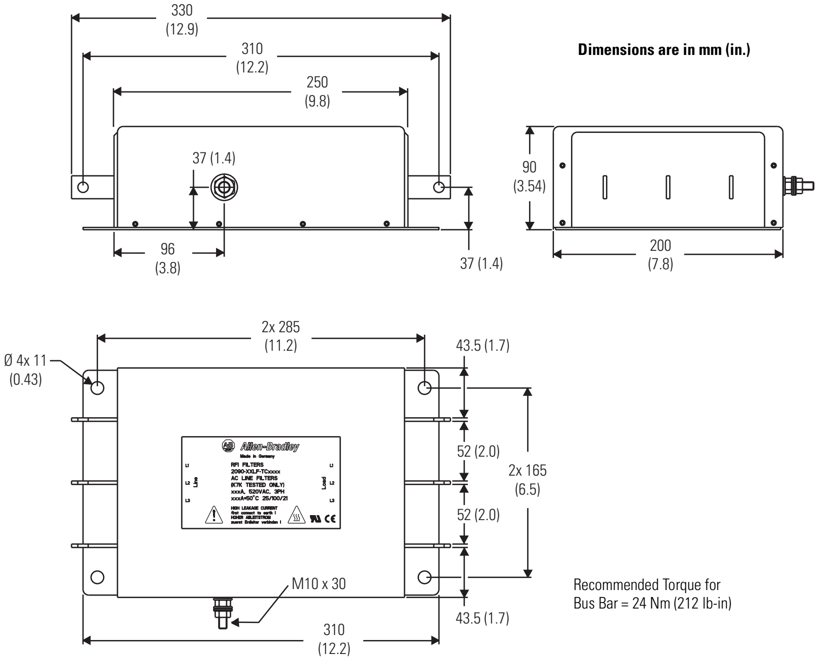
AC Line Filter Dimensions
2090-XXLF-TC316, 2090-XXLF-TC350, 2090-XXLF-TC365, and 2090-XXLF-TC3100



Cat. No.	A	B	C	D	E	F
	mm (in.)					
2090-XXLF-TC316	230 (9.0)	50 (1.96)	80 (3.15)	25 (0.98)	215 (8.46)	200 (7.87)
2090-XXLF-TC350	180 (7.08)	85 (3.35)		65 (2.56)	164 (6.45)	150 (5.90)
2090-XXLF-TC365						
2090-XXLF-TC3100	240 (9.45)	95 (3.74)	90 (3.54)	75 (2.95)	223 (8.78)	210 (8.27)

Cat. No.	G	H	I	J	K
	mm (in.)				
2090-XXLF-TC316	12 (0.47)	M6x25	1.0 (0.04)	5.5 (0.21)	15 (0.59)
2090-XXLF-TC350	39 (1.53)				
2090-XXLF-TC365					
2090-XXLF-TC3100	43 (1.69)	M8x40	1.5 (0.06)	5.5 (0.21)	16 (0.63)

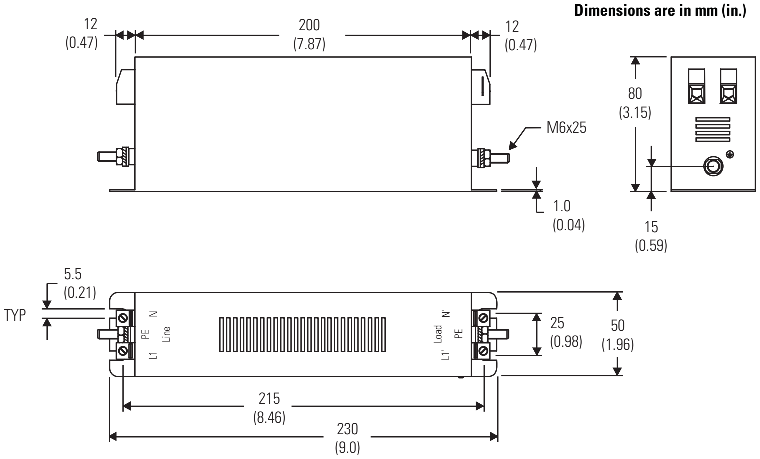
AC Line Filter Dimensions
2090-XXLF-TC3150, 2090-XXLF-TC3200, and 2090-XXLF-TC3250



Dimensions are in mm (in.)

Recommended Torque for
Bus Bar = 24 Nm (212 lb-in)

AC Line Filter Dimensions
2090-XXLF-TC116



Dimensions are in mm (in.)

Additional Resources

The following documents contain additional information concerning related Allen-Bradley products.

Resource	Description
Kinetix 2000 Multi-axis Servo Drive User Manual, publication 2093-UM001	Information on system layout, including the ac line filter for your Kinetix motion control servo drive.
Kinetix 6000 Multi-axis Servo Drive User Manual, publication 2094-UM001	
Kinetix 7000 High Power Servo Drive User Manual, publication 2099-UM001	
Ultra3000 Digital Servo Drive Installation Manual, publication 2098-IN003	
Ultra5000 Intelligent Positioning Drive Installation Manual, publication 2098-IN001	
Ultra1500 Digital Servo Amplifier User Manual, publication 2092-UM001	
Kinetix Motion Control Selection Guide, publication GMC-SG001	Kinetix motion control drive/motor specifications, combinations, and accessories including ac line filters.
System Design for Control of Electrical Noise Reference Manual, publication GMC-RM001	Information, examples, and techniques designed to minimize system failures caused by electrical noise.
EMC Noise Management DVD, GMC-SP004	
Rockwell Automation Configuration and Selection Tools, website http://ab.com/e-tools	Online product selection and system configuration tools, including AutoCAD (DXF) drawings.
Rockwell Automation Product Certification, website http://rockwellautomation.com/products/certification	For declarations of conformity (DoC) currently available from Rockwell Automation.
National Electrical Code, Published by the National Fire Protection Association of Boston, MA.	An article on wire sizes and types for grounding electrical equipment.

You can view or download publications at http://literature.rockwellautomation.com. To order paper copies of technical documentation, contact your local Rockwell Automation distributor or sales representative.

Rockwell Automation Support

Rockwell Automation provides technical information on the Web to assist you in using its products. At http://support.rockwellautomation.com, you can find technical manuals, a knowledge base of FAQs, technical and application notes, sample code and links to software service packs, and a MySupport feature that you can customize to make the best use of these tools.

For an additional level of technical phone support for installation, configuration, and troubleshooting, we offer TechConnect Support programs. For more information, contact your local distributor or Rockwell Automation representative, or visit http://support.rockwellautomation.com.

Installation Assistance

If you experience a problem with a hardware module within the first 24 hours of installation, please review the information that's contained in this manual. You can also contact a special Customer Support number for initial help in getting your module up and running.

United States	1.440.646.3223 Monday – Friday, 8am – 5pm EST
Outside United States	Please contact your local Rockwell Automation representative for any technical support issues.

New Product Satisfaction Return

Rockwell tests all of its products to ensure that they are fully operational when shipped from the manufacturing facility. However, if your product is not functioning, it may need to be returned.

United States	Contact your distributor. You must provide a Customer Support case number (see phone number above to obtain one) to your distributor in order to complete the return process.
Outside United States	Please contact your local Rockwell Automation representative for return procedure.

www.rockwellautomation.com

Power, Control and Information Solutions Headquarters

Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
Europe/Middle East/Africa: Rockwell Automation, Vorstlaan/Boulevard du Souverain 36, 1170 Brussels, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia Pacific: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Publication 2090-IN013B-EN-P — January 2007 PN 334021-P02

Supersedes Publication 2090-IN013A-EN-P — August 2006

ANEXO I. Kinetix Motion Control-Selection Guide.

Selection Guide

Allen-Bradley

Kinetix Motion Control

Rotary Motion
- MP-Series
- TL-Series
- HPK-Series
- RDD-Series

Linear Motion
- MP-Series
- TL-Series
- LDC-Series
- LDL-Series

Logix Motion Modules
- 1756
- 1768

Servo Drives
- 2071
- 2097
- 2093
- 2098
- 2094
- 2099

Motion Accessories
- 2090
- 1394

LISTEN.
THINK.
SOLVE.®

Allen-Bradley · Rockwell Software

Rockwell Automation

Important User Information

This guide has been developed as a quick reference tool for Allen-Bradley motion controls and systems. It is not intended to replace user manuals or technical documentation supplied with our Allen-Bradley equipment, which should be referred to for actual installation, connection, operation, and maintenance of Allen-Bradley equipment.

Because of the variety of uses for the products described in this publication, those responsible for the application and use of this control equipment must satisfy themselves that all necessary steps have been taken to ensure that each application and use meets all performance and safety requirements, including any applicable laws, regulations, codes, and standards.

Reproduction of the contents of this copyrighted publication, in whole or in part, without written permission of Rockwell Automation, is prohibited.

Contents of This Guide

This selection guide provides features, specifications, and dimensions for selecting Kinetix Motion Control servo drives, motors, actuators, and accessory components.

Chapter	Title	Contents	Page
	Preface	An introduction to Kinetix Integrated Motion and tables for comparing features and specifications from one family of motion control products to another.	5
1	Rotary Motion	MP-Series, TL-Series, and HPK-Series rotary servo motors. RDD-Series rotary direct-drive motors.	19
2	Linear Motion	MP-Series integrated linear stages. MP-Series and TL-Series electric cylinders. LDC-Series and LDL-Series linear motors.	91
3	Controller Platforms	EtherNet/IP network modules, SERCOS interface modules, and analog servo modules for ControlLogix and CompactLogix programmable automation controllers. MicroLogix programmable logic controllers for component-level motion control.	237
4	Kinetix 6200 and Kinetix 6500 Modular Multi-axis Servo Drives	Modular, multi-axis, safe-speed servo drive family includes Bulletin 2094 IAM and AM power modules, and Kinetix 6200 and Kinetix 6500 control modules.	249
5	Kinetix 6000 Multi-axis Servo Drives	Multi-axis servo drive family includes IAM (converter) modules and AM (inverter) modules.	275
6	Kinetix 2000 Multi-axis Servo Drives	Multi-axis servo drive family includes IAM (converter) modules, AM (inverter) modules, and Bulletin 2093 shunt module, slot-filler module, and power rails.	301
7	Kinetix 7000 High Power Servo Drives	Single-axis high-power servo drives.	321
8	Kinetix 300 EtherNet/IP Indexing Servo Drives	Single-axis, EtherNet/IP network, indexing drive family and Bulletin 2097 drive accessories.	339
9	Kinetix 3 Component Servo Drives	Single-axis, indexing component drive family and Bulletin 2071 drive accessories.	353
10	Ultra3000 Digital Servo Drives	Single-axis digital servo drives.	361
11	Ultra5000 Intelligent Positioning Drives	Single-axis standalone drives, ANSI C programmable.	379
12	Motion Control Accessories	Motor cable specifications and selection by motor family; interface cable specifications; connector kits; Bulletin 2094 shunt module, slot-filler module, and power rails; and assorted power components for the Kinetix Motion Control drive families.	395
13	Rotary Motion System Combinations	Cable selection, performance specifications, and typical torque/speed curves for the optimum drive/rotary motor combinations (listed by drive family).	511
14	Linear Motion System Combinations	Cable information, performance specifications, and typical force/velocity curves for the optimum drive/linear actuator combinations (listed by drive family).	665

This selection guide contains new products and updated information.

New and Updated Information

This revision contains product specifications for these new Kinetix Motion Control servo drives, motors, actuators, and accessory items.

Section	Changes	Page
Preface	Kinetix 3 drive specifications summary	15
Chapter 1	MPL-A4560F motor performance specifications	23
	MPL-B980E motor performance specifications	24
	Addition of SpeedTec DIN standard (non-flex) cables for compatible motors	Throughout
Chapter 2	Bulletin MPAR electric cylinder parallel mount configurations	134
	Addition of SpeedTec DIN standard (non-flex) cables for compatible motors/actuators	Throughout
Chapter 3	MicroLogix controller platform	241
Chapter 4	Peak current specifications	258
	IAM power module specifications for 2094-BC04-M03-M and 2094-BC07-M05-M drives	Throughout
	AM power module specifications for 2094-BM03-M and 2094-BM05-M drives	
Chapter 5	IAM and AM module series change	276
	Peak enhancement specifications	281
	IAM module specifications for 2094-BC04-M03-S and 2094-BC07-M05-S (series B) drives	Throughout
	AM module specifications for 2094-BM03-S and 2094-BM05-S (series B) drives	
Chapter 6	Footnote with battery requirement for Kinetix 2000 drives and Bulletin TLY motors and Bulletin TLAR electric cylinders	303 304
Chapter 8	Footnote with battery requirement for Kinetix 300 drives and Bulletin TLY motors and Bulletin TLAR electric cylinders	340
	Revised Kinetix 300 drive specifications	Throughout
Chapter 9	Kinetix 3 component servo drives	353
Chapter 12	Standard (non-flex) SpeedTec DIN motor/actuator cables	396
	Motor power and feedback cable connector kits for circular plastic (M6) connectors	425
	Bulletin 2090 serial cables for Kinetix 3 drives	434
	Bulletin 2071 breakout boards for Kinetix 3 drives	446
Chapter 13	Cables, performance specifications, and torque/speed curves for Kinetix 6200/6500 (catalog numbers 2094-BM03-M and 2094-BM05-M) drives with compatible rotary motors	519
	Cables, performance specifications, and torque/speed curves for Kinetix 6000 (catalog numbers 2094-BM03-S and 2094-BM05-S) series B drives with compatible rotary motors	
	Cables, performance specifications, and torque/speed curves for Kinetix 3 drives with compatible rotary motors	619
Chapter 14	Cables, performance specifications, and force/velocity curves for Kinetix 6200/6500 (catalog numbers 2094-BM03-M and 2094-BM05-M) drives with compatible linear motors	685
	Cables, performance specifications, and force/velocity curves for Kinetix 6000 (catalog numbers 2094-BM03-S and 2094-BM05-S) series B drives with compatible linear motors	
	Cables, performance specifications, and force/velocity curves for Kinetix 3 drives with compatible linear motors and actuators	729

Notes:

What Is Kinetix Integrated Motion?

The Kinetix Integrated Motion offerings are part of the Rockwell Automation Integrated Architecture system. The Integrated Architecture system brings together a wide range of high-performance products that are integrated into RSLogix 5000 software for simplified and enhanced machine design, operation, and maintenance.

With Kinetix Integrated Motion, you'll benefit from the seamless integration of Allen-Bradley Logix controllers (ControlLogix and CompactLogix), high-performance networks (EtherNet/IP and SERCOS), and a broad range of Allen-Bradley AC and servo drives, linear and rotary motors, and linear actuator options. RSLogix 5000 software offers an extensive set of advanced motion tools for programming, configuration, commissioning, diagnostics, and maintenance support. Catalog number driven configuration makes motion system commissioning fast and simple, and an extensive library of motion instructions provide the right functionality for any application.

Kinetix Integrated Motion offers a variety of servo drive, motor, and actuator families for single-axis and multi-axis applications. These systems offer:

- servo drive power ranges from 0.5…149 kW.
- a choice of SERCOS interface or EtherNet/IP networks.
- a wide range of rotary motors, rotary direct drive motors, linear motors, and linear actuators/stages. Motors offer continuous torque as low as 0.10 N•m (0.85 lb•in) and up to 955 N•m (8452 lb•in). Linear actuators offer peak forces of up to 14,679 N (3300 lb).
- Smart Motor Technology provides automatic motor identification for fast, easy configuration and commissioning.
- use of a single software package, RSLogix 5000, for complete support of drive configuration, programming, commissioning, diagnostics and maintenance.
- powerful on-line motion tools including real-time data trending, graphical PCAM and TCAM profile editor, auto and manual drive tuning, and advanced drive diagnostics.
- Automatic Device Replacement (ADR) plug-and-run drive/motor/actuator support.
- Motion Analyzer software for comprehensive motion-application sizing and analysis, optimization, selection, and validation of your Kinetix motion control system.

Connected Components Platform

Part of Rockwell Automation's Machine Solutions offering, Connected Components is a preferred control solution for machine builders who provide stand-alone machines at low cost. Connected Components provides just enough control to meet machine and end-user requirements while helping to improve operating efficiencies. Engineering and application tool sets allow easy design and installation with preferred interoperability of the broad range of component-class products.

The Kinetix 3 component servo drive provides a motion control solution for machine builders producing low-cost equipment at high volumes. The component servo drive can apply the appropriate level of control for the application without the added complexity. Systems can include serial commands from MicroLogix controllers or discrete wiring directly to the sensor or controller with TL-Series low-inertia motors.

Motion Analyzer Software

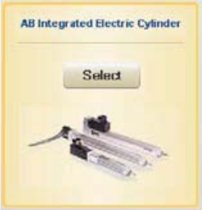
Motion Analyzer software is a comprehensive motion-application sizing tool used for analysis, optimization, selection, and validation of your Kinetix motion control system.

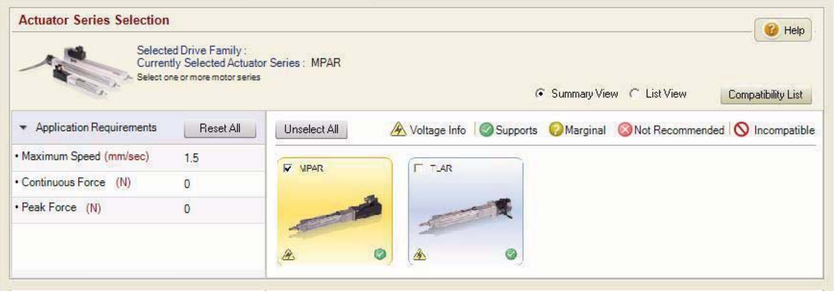
Motion Analyzer software facilitates the machine design and investigation process by making it fast, simple, and accurate. Motion Analyzer software offers a fact-based decision path and design optimization approach that enables machine builders to:

- reduce motion system design risk.
- reduce time from machine design to shipment.
- optimize motion control system cost and size.
- enhance machine performance and reliability.
- create a bill of materials.

Motion Analyzer software includes the full range of Kinetix Motion Control products and features:

MP-Series Electric Cylinders - deliver off-the-shelf linear motion without the need to search through catalogs looking for suitable ballscrews, timing belts, pulleys, and bearings. No mechanical data to enter, just input load information and move profile. Output even includes an L_{10} life estimate.

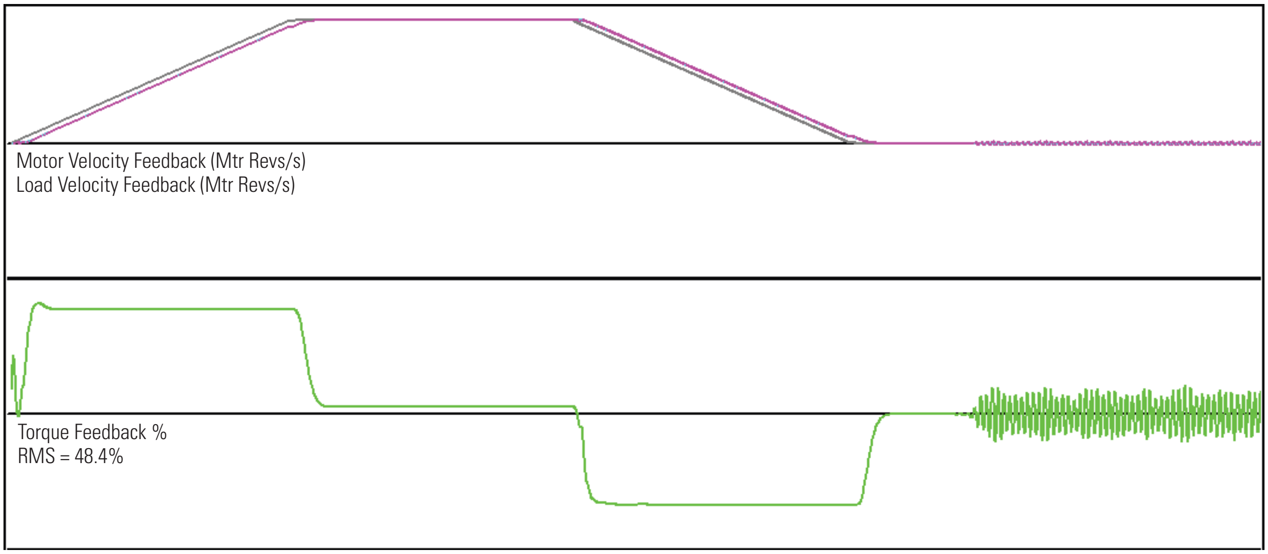




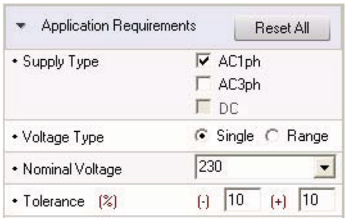
Selected Drive Family :
Currently Selected Actuator Series : MPAR
Select one or more motor series

Application Requirements	
• Maximum Speed (mm/sec)	1.5
• Continuous Force (N)	0
• Peak Force (N)	0

Tuning simulation - is a tool to help predict how your machine will perform under real-world conditions. Emulates tuning an axis in RSLogix 5000 software (including Auto-tune) and then simulates the behavior of the load, motor, and drive. Factors-in the mechanical compliance or backlash to give a realistic simulation.



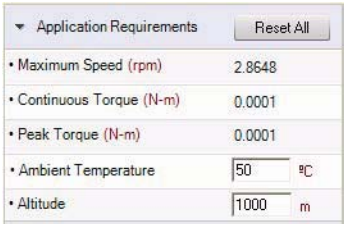
Variable mains supply analysis - is especially useful for machine builders exporting machines overseas.



Application Requirements

- Supply Type ☑ AC1ph ☐ AC3ph ☐ DC
- Voltage Type ● Single ○ Range
- Nominal Voltage 230
- Tolerance (%) (-) 10 (+) 10

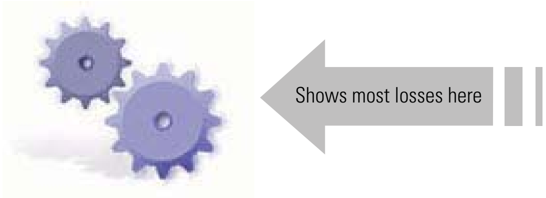
Motor thermal performance prediction - takes into account the motor ambient temperature to verify performance in extreme-heat conditions.

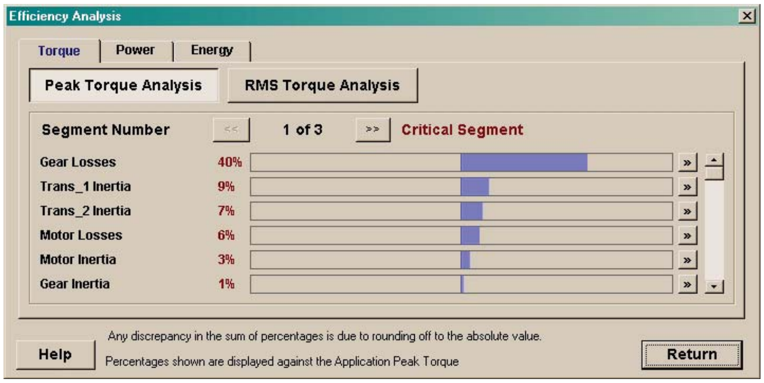


Application Requirements

- Maximum Speed (rpm) 2.8648
- Continuous Torque (N-m) 0.0001
- Peak Torque (N-m) 0.0001
- Ambient Temperature 50 ºC
- Altitude 1000 m

Efficiency analysis - provides an understanding of where the torque produced by the motor is consumed.





Efficiency Analysis

Segment Number 1 of 3 Critical Segment

Gear Losses	40%
Trans_1 Inertia	9%
Trans_2 Inertia	7%
Motor Losses	6%
Motor Inertia	3%
Gear Inertia	1%

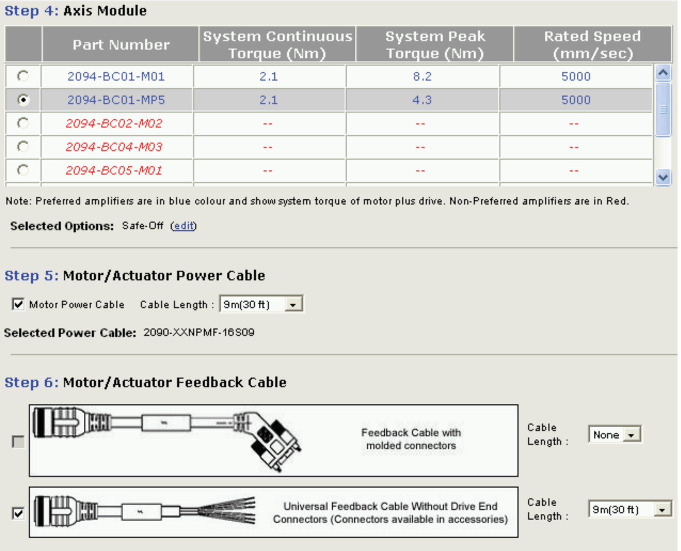
Any discrepancy in the sum of percentages is due to rounding off to the absolute value.

Percentages shown are displayed against the Application Peak Torque

Motion Analyzer software also helps generate a bill of materials (BOM). With its rule-based approach, selecting the right drive, motor, cables, I/O connectors, and other accessory items is fast and error free.

You can have a system specified as a BOM or have selections based on the results of motor/actuator and drive sizing calculated by Motion Analyzer software.

Once finished, you can printout the BOM or export the BOM file to Microsoft Word or Excel application files.



Step 4: Axis Module

	Part Number	System Continuous Torque (Nm)	System Peak Torque (Nm)	Rated Speed (mm/sec)
○	2094-BC01-M01	2.1	8.2	5000
●	2094-BC01-MP5	2.1	4.3	5000
○	*2094-BC02-M02*	--	--	--
○	*2094-BC04-M03*	--	--	--
○	*2094-BC05-M01*	--	--	--

Note: Preferred amplifiers are in blue colour and show system torque of motor plus drive. Non-Preferred amplifiers are in Red.

Selected Options: Safe-Off (edit)

Step 5: Motor/Actuator Power Cable

☑ Motor Power Cable Cable Length : 9m(30 ft)

Selected Power Cable: 2090-XXNPMF-16S09

Step 6: Motor/Actuator Feedback Cable

☐ Feedback Cable with molded connectors Cable Length : None

☑ Universal Feedback Cable Without Drive End Connectors (Connectors available in accessories) Cable Length : 9m(30 ft)

To download Motion Analyzer software, go to http://www.ab.com/motion/software/analyzer.html.

Kinetix Motion Control System Checklist

1 - Start with Motion Analyzer Software

Motion Analyzer software is a comprehensive sizing tool used for analysis, optimization, selection, and validation of your Kinetix Motion Control system. Given any drive and compatible motor/actuator, Motion Analyzer software will provide you with the data to determine the optimum drive and motor/actuator combination for your application.

You can also use Motion Analyzer software to build your bill of materials and receive an itemized list of system components with catalog numbers and product descriptions.

2 - Select System Components

Kinetix Motion Control System Component	Product Family	Chapter	√
Rotary Servo Motors (see features table on page 9)	• MP-Series Low Inertia Motors • MP-Series Medium Inertia Motors • MP-Series Food Grade Motors • MP-Series Stainless Steel Motors • RDD-Series Direct Drive Motors • HPK-Series Motors • TL-Series Motors	Chapter 1	
Linear Servo Motors (see features table on page 11)	• LDC-Series Iron Core Linear Motors • LDL-Series Ironless Linear Motors	Chapter 2	
Linear Actuators (see features table on page 12)	• MP-Series Integrated Linear Stages • MP-Series Integrated Multi-axis Linear Stages • MP-Series and TL-Series Electric Cylinders		
Controller Platforms	• ControlLogix Platform (Bulletin 1756) • CompactLogix Platform (Bulletin 1768) • MicroLogix Platform (Bulletin 1763 and 1766)	Chapter 3	
Servo Drives (see features table on page 14)	Kinetix 6200 and Kinetix 6500 Modular Multi-axis Servo Drives	Chapter 4	
	Kinetix 6000 Multi-axis Servo Drives	Chapter 5	
	Kinetix 2000 Multi-axis Servo Drives	Chapter 6	
	Kinetix 7000 High Power Servo Drives	Chapter 7	
	Kinetix 300 EtherNet/IP Indexing Servo Drives	Chapter 8	
	Kinetix 3 Component Servo Drives	Chapter 9	
	Ultra3000 Digital Servo Drives	Chapter10	
	Ultra5000 Intelligent Positioning Servo Drives	Chapter 11	
Motion Control Accessories	• Motor Cables • Interface Cables • Connector Kits • Power Components	Chapter 12	

3 - Verify System Combinations

Rotary System Combinations, in chapter 13, includes cable combinations, system performance specifications tables, connector kit requirements, and torque/speed curves for the optimum drive/motor or drive/actuator combination.

Linear System Combinations, in chapter 14, includes cable combinations, system performance specifications tables, connector kit requirements, and force/velocity curves for the optimum drive/actuator combination.

Rotary Servo Motors

Rotary motors (except TL-Series) are UL Recognized components to applicable UL and CSA standards. CE marked for all applicable directives. Refer to http://www.ab.com for more information.

MP-Series Servo Motors

Motor Features	MP-Series (Bulletin MPL) Low Inertia Motors	MP-Series (Bulletin MPM) Medium Inertia Motors	MP-Series (Bulletin MPF) Food Grade Motors	MP-Series (Bulletin MPS) Stainless Steel Motors
Main characteristics	• High torque to size ratio • Smart Motor Technology • Low rotor inertia	• High torque to size ratio • Smart Motor Technology • Medium rotor inertia • Easy migration from 1326AB motors	• Combined characteristics of MP-Series low-inertia motors and features specifically designed for food and beverage applications. • Low rotor inertia	• Specifically designed for hygienic environments for use with high pressure, highly caustic washdown applications. • Low rotor inertia
Features	• 230V and 460V windings • High-energy rare-earth magnets • Shaft end threaded hole • DIN connectors, rotates 180° • Standard IEC 72-1 mounting dimensions	• 230V and 460V windings • Multiple winding speed options • High-energy rare-earth magnets • Shaft end threaded hole • SpeedTec-ready DIN connectors, rotates 180° • Standard IEC 72-1 mounting dimensions	• Epoxy coated • 230V and 460V windings • Shaft end threaded hole • SpeedTec-ready DIN connectors, rotates 180° • Standard IEC 72-1 mounting dimensions	• Smooth, passivated 300 series stainless-steel cylindrical exterior. • Certified and listed to NSF/ANSI Standard 169 • 230V and 460V windings • Shaft end threaded hole • Cable extensions, 3 m (9.8 ft) • Standard IEC 72-1 mounting dimensions
Motor type	Brushless AC synchronous servo motors			
Environmental rating	• IP50 minimum, without shaft seal (standard). • IP66 with optional shaft seal and use of environmentally sealed cable connectors.	• IP50 minimum, without shaft seal (standard). • IP67 with optional shaft seal and use of environmentally sealed cable connectors.	• IP66/IP67 with shaft seal (standard) and use of environmentally sealed cable connectors. • Food grade grease on shaft seal	• IP66/IP67 with shaft seal (standard) and use of environmentally sealed cable connectors. • IP69K for 1200 psi washdown
Continuous torque	0.26…163 N•m (2.3…1440 lb•in)	2.18…62.8 N•m (19.3…556 lb•in)	1.6…19.4 N•m (14…172 lb•in)	3.6…21.5 N•m (32…190 lb•in)
Peak torque	0.74…278 N•m (6.6…2460 lb•in)	6.6…154.2 N•m (58…1365 lb•in)	3.61…48.6 N•m (32…430 lb•in)	11.1…98 N•m (67.8…600 lb•in)
Speed	Up to 8000 rpm	Up to 7000 rpm	Up to 5000 rpm	3000 and 5000 rpm
Motor rated output	0.16…18.6 kW	0.75…7.50 kW	0.73…4.1 kW	1.3…3.5 kW
Feedback options	• Multi-turn, high-resolution absolute position • Single-turn, high-resolution absolute position	• Multi-turn, high-resolution absolute position • Single-turn, high-resolution absolute position • Resolver	• Multi-turn, high-resolution absolute position • Single-turn, high-resolution absolute position	
Motor options	• 24V DC brake • Shaft seal kit • Keyless shaft (limited frame sizes)	• 24V DC brake • Shaft seal kit • Positive air pressure kit	• 24V DC brake • Shaft seal kit • Positive air pressure kit	• 24V DC brake • Shaft seal kit with slinger • Positive air pressure kit
Typical applications	• Packaging • Converting • Material handling • Electronic assembly • Automotive • Metal forming	• Printing • Web handling • Converting • Automotive • Metal forming	• Food packaging • Volumetric filling • Form, fill, seal • Food handling • For meat and poultry applications, the MP-Series Stainless Steel motors are recommended.	• Meat and poultry • Food slicing and filling • Raw food handling • Processing • Life science • Consumer products

RDD-Series and HPK-Series Servo Motors

Motor Features	RDD-Series Motors	HPK-Series Motors
Main characteristics	• Smart Motor Technology • Direct coupling to the load • Bearingless housed configuration	• High-power • Large load inertia
Features	• 460V windings • Multiple winding speed options • SpeedTec-ready DIN connectors, rotates 180° • Standard IEC 72-1 mounting dimensions	• 400V and 460V windings • DIN connectors, rotates 180° • Blower cooled • IEC flange or foot mount
Motor type	Direct-drive rotary servo motor	Asynchronous Induction Motors
Environmental rating	IP65 with use of environmentally sealed cable connectors.	IP54
Continuous torque	32.7…426 N•m (289…3770 lb•in)	96…482 N•m (849…4266 lb•in)
Peak torque	86.5…1050 N•m (766…9293 lb•in)	192…964 N•m (1699…8531 lb•in)
Speed	Base speeds between 177…1836 rpm	Base speeds of 1500 and 3000 rpm
Motor rated output	1.97…8.69 kW	17.1…150 kW
Feedback options	• Multi-turn, high-resolution Heidenhain EnDat 2.2 • Single-turn, high-resolution Heidenhain EnDat 2.2	• Multi-turn, high-resolution absolute position • Single-turn, high-resolution absolute position
Motor options	N/A	380…480V AC brake
Typical applications	• Use to replace mechanical gear reduction (gear boxes, belts, pulleys) • Tight space constraints • Axes with high-power and high-performance requirements	• High power packaging • Converting • Wind/unwind/rewind • Sheeters • Flying knife • Material handling

TL-Series Servo Motors

Motor Features	TL-Series (Bulletin TL and TLY) Motors
Main characteristics	• Compact size, high torque density • Metric and NEMA frame sizes • Smart Motor Technology • Low rotor inertia
Features	• 230V windings • High-energy Neodymium-Iron-Boron (NeFeB) magnets • Cable extensions, 1 m (3.2 ft) • 17-bit serial communication
Motor type	Brushless AC Synchronous Servo Motors
Environmental rating	IP65 with optional shaft seal
Continuous torque	0.086…5.42 N•m (0.76…48 lb•in)
Peak torque	0.22…13 N•m (1.94…115 lb•in)
Speed	4500, 5000, and 6000 rpm
Motor rated output	0.037…2.0 kW
Feedback options	• Multi-turn, high-resolution absolute position • Incremental (2000 counts)
Motor options	• 24V DC brake • Shaft seal kit
Typical applications	• Robotics • Material handling • X-Y tables • Specialty machinery • Semiconductor manufacturing • Medical/laboratory equipment • Light packaging machines • Office machinery

Linear Servo Motors

Linear motors are UL Recognized components to applicable UL and CSA standards. CE marked for all applicable directives. Refer to http://www.ab.com for more information.

LDC-Series and LDL-Series Linear Servo Motors

Linear Motor Features	LDC-Series Linear Servo Motors	LDL-Series Linear Servo Motors
Main characteristics	• High thrust force to cost ratio for less costly solutions • Cogging torque < 5% of the continuous force • 230/400 and 460V AC operation	• Non-cogging technology for super smooth motion • No magnetic attraction between the coil and magnet channel allows for the use of smaller, less expensive linear bearings • No external magnetic field to have to shield in magnetic sensitive applications • 230V AC operation
Features	• Speed capabilities to 10 m/s (32.8 ft/s) to increase machine productivity • Direct drive technology for extreme servo responsiveness • No wear parts to increase machine productivity through less maintenance and replacement. • Standard MP-Series motor power and feedback connectors to easily combine with Allen-Bradley extension and flex cables	
Motor type	Iron core coil and magnet track	Ironless coil and magnet channel
Environmental rating	IP65 and RoHS compliant	
Continuous forces	74…2882 N (17…648 lb)	63…596 N (14…134 lb)
Peak forces	188…5246 N (42…1179 lb)	209…1977 N (47…444 lb)
Peak velocity	10 m/s (32.8 ft/s)	10 m/s (32.8 ft/s)
Cogging torque	< 5% of the continuous force	Zero
Field-installable accessories	• Cooling plates • Bulkhead connector kit • Encoder connector kit • Hall sensor for connectorized coil • Hall sensor for flying-lead coil	• Bulkhead connector kit • Encoder connector kit • Hall sensor for connectorized coil • Hall sensor for flying-lead coil
Typical applications	• Form-fill and seal packaging machines • Large format gantries (pick and place, scribing and palletizing) • Material handling (pallet movers and sheet glass) • Plasma, laser and water jet cutting machines • Machine tools • Flying cut off machines • Coordinate measuring machines • Large format routers • Large format printers (step axis)	• Wafer cutting, handling and marking • Computer-to-plate printing machines • Large format printing (print head axis) • Solar and flat panel scribing (scribe head axis) • Axis requiring extremely smooth/constant velocity

Linear Actuators

Actuators are UL Recognized components to applicable UL and CSA standards. CE marked for all applicable directives. Refer to http://www.ab.com for more information.

MP-Series Integrated Linear Stages

Actuator Features	MP-Series (Bulletin MPAS) Integrated Linear Stages	MP-Series (Bulletin MPMA) Integrated Multi-axis Linear Stages
Main characteristics	• Rugged linear stages with integrated direct-drive linear motor or ballscrew with MP-Series servo motor • Available in three frame sizes (base widths) to accommodate a variety of load requirements for general automation • Smart Motor Technology (ballscrew) • Very high linear speeds (direct drive)	• Out of box alignment of 30 arc seconds • Field replaceable quick change cable management for ease of maintenance • Sealed stages having IP30 rating for environmental protection • Caged ball-type linear guides that retain lubrication for longer bearing life and provide lower noise levels • Absolute encoders on ballscrew axis and incremental encoders on direct-drive linear motor axis • MP-Series motor power and feedback connectors for connection to Allen-Bradley extension cables and drives • Access holes for easy lubrication
Features	• 200/230V and 400/460V operation (230V only for 150 mm direct-drive frame size) • High-energy neodymium (NeFeB) magnets • Heavy duty connectors • Operation without limit and home switches • Carriage and base mounting design allows 200 mm and 250 mm frame sizes to be stacked • Standard MP-Series motor power and feedback connectors • Optional air purge kit for added protection against ingress of foreign substances	
Actuator type	• Direct-drive linear stage • Ballscrew-drive linear stage	
Environmental rating	Unique, long life strip seal system provides IP30 environmental rating to prevent debris, larger than 2.5 mm (0.1 in.) diameter, from entering the linear stage.	
Continuous forces	83…521 N (19…117 lb)	
Peak forces	312…1212 N (70…273 lb)	
Peak velocities	200…5000 mm/s (7.9…196.9 in./s)	
Stroke lengths (1)	120…1940 mm (4.7…76.4 in.)	
Feedback options	• Multi-turn, high-resolution absolute position (ballscrew) • 5 micron resolution incremental magnetic linear encoder (direct drive)	
Field-installable accessories	• Cable track module replacement kit • Strip seal replacement kit • Top cover • Side cover • Coupling • T-nut kit (package of 10) • Toe-clip kit (package of 10) • Grease gun kit • Grease replacement cartridge	• Cable track module replacement kit • Strip seal replacement kits • Top cover kits (for Y or Z-axis only) • Side cover kits • Coupling kits (for Y or Z-axis only) • Tee-nut kit (package of 10) • Tee-nut bar kit • Grease gun kit • Grease replacement cartridge • Rotary servo motor (for Y or Z-axis only)
Typical applications	• Electronic assembly • Pick and place • Robots • Inspection • Labeling • Dispensing • Micro-arraying	• Material handling • Pick and place • Dispensing • Scanning • Contouring • Contoning • Flying shape cutting

(1) Applies to Bulletin MPAS linear stages. Not all Bulletin MPAS stroke lengths (travels) are available with Bulletin MPMA multi-axis linear stages.

MP-Series and TL-Series Electric Cylinders

Actuator Features	TL-Series (Bulletin TLAR) Electric Cylinders	MP-Series (Bulletin MPAR) Electric Cylinders	MP-Series (Bulletin MPAI) Heavy Duty Electric Cylinders
Main characteristics	State-of-the-art design features ball-screw construction driven by TL-Series (Bulletin TLY) servo motors.	State-of-the-art design features ball-screw construction driven by MP-Series (Bulletin MPL) servo motors.	• State-of-the-art design features ball-screw and roller-screw construction driven by MP-Series (Bulletin MPL) servo motors. • Front flange-mount and front trunnion-mount cylinders
	• Fully assembled and ready to mount cylinders contribute to reductions in mechanical design engineering, wiring, and commissioning time • Smart Motor Technology • Very high linear speeds		
Features	• 200/230V operation • Absolute, high-resolution feedback options consistent with TL-Series (Bulletin TLY) servo motors • Standard TL-Series motor power and feedback connectors	• 200/230V and 400/460V operation • Absolute, high-resolution feedback options consistent with MP-Series servo motors • Standard MP-Series motor power and feedback connectors	
	• Rated for 100% duty cycle and designed for repeatable, reproducible performance over the actuator's operating life • Absolute feedback allows operation without limit and home switches • No piping, valving, air, or oil supply required		
	ISO 15552 pneumatic-class frame sizes 32, 40, and 63 mm		Frame sizes 83 and 110 mm
Actuator type	Ball-screw driven electric cylinders		Ball-screw and roller-screw electric cylinders
Environmental rating	IP40 (complete unit) includes rod-end seal and breather port	• IP40 (complete unit) includes rod-end seal and breather port • IP66 for electronic components with the use of environmentally sealed (Bulletin 2090) cable connectors	IP67 with the use of environmentally sealed (Bulletin 2090) cable connectors
Continuous feed force	240…2000 N (54…450 lb)		2002…7784 N (450…1750 lb)
Max feed force	300…2500 N (67…562 lb)		4003…8896 N (900…2000 lb)
Peak velocities	0.15…1.0 m/s (5.9…39.4 in./s)		279…559 mm/s (11.0…22.0 in./s)
Stroke lengths (1)	100…800 mm (4.0…32.0 in.)		150, 300, 450 mm (6.0, 12.0, 18.0 in.)
Optional equipment	24V DC holding brakes		24V DC holding brakes
Field-installable accessories	• Foot mounting for axial motor attachment • Flange mounting • Trunnion mounting kit • Trunnion support • Mounting attachments (clevis foot, right-angle clevis foot) • Piston-rod attachments (rod eye, rod clevis, rod coupler, coupling piece) • Guide rod		• Mounting plates • Front flange mount • Rear clevis mount • Rod-end attachments (rod eye, rod clevis) • Anti-rotation guide
Typical applications	• Material handling (loading, unloading, lifts, pick and place, diverters, transfers, gantries) • Volumetric filling and process control (web guides, valve, nozzle, van, and gate positioning) • Fabrication (adjustments for machine backstops and cutting tools, works alignment) • Push, pull, eject, press, or clamp parts • Packaging (consumer products, automotive, medical) • Electronic assembly • Insertion systems • Inspection and test equipment		

(1) Not all stroke lengths (travels) are available with all frame sizes.

Servo Drives

Servo drives meet CE compliance and are UL Listed to U.S. and Canadian safety standards. Refer to http://www.ab.com for more information.

Servo Drives

Drive Features	Kinetix 6500	Kinetix 6200	Kinetix 6000	Kinetix 7000
Main characteristics	• Multi-axis • Common Bus • Modular Design • Integrated Motion Drive		• Multi-axis • Common Bus • Enhanced Peak Performance (1) • Integrated Motion Drive	• High-power applications • Common Bus • Integrated Motion Drive
	Safe Speed Monitoring and Safe Torque-off Control TÜV certified SIL CL3, PLe, category 4		Safe-off Control TÜV certified SIL CL3, PLe, category 3	
Drive configuration	1…8 Axes on Bulletin 2094 Power Rail			Single-axis
Input voltage	324…528V AC, Three-phase (460V systems)		195…265V AC, Three-phase (230V systems)	342…528V AC Three-phase
			324…528V AC, Three-phase (460V systems)	
Common-bus follower input voltage	458…747V DC (460V systems)		275…375V DC (230V systems)	450…750V DC
			458…747V DC (460V systems)	
Continuous output power (inverter)	1.8…6.6 kW (460V systems)		1.2…11 kW (230V systems)	22…149 kW
			1.8…22 kW (460V systems)	
Continuous output current (inverter)	2.8…10.3 A rms (460V systems)		3.7…34.6 A rms (230V systems)	40…248 A rms
			2.8…34.6 A rms (460V systems)	
Drive digital inputs	• Enable, Home, OverTravel ± • High Speed Registration (2/axis)			• Enable, Home, OverTravel ± • High Speed Registration (2)
Drive digital outputs	Motor Brake Relay Output (with suppression)			
DPI connector	N/A		DriveExplorer software or DPI HIM Module	
Programming	RSLogix 5000 software (Ladder Logic, Structured Text, and Sequential Function Charts)			
	Version 18 or later	Version 17 or later	Version 11 or later	Version 15 or later
Logix module compatibility	1756-EN2T, 1756-EN2TR, 1756-EN3TR	• 1756-M03SE, 1756-M08SE, 1756-M16SE • 1768-M04SE		
I/O control	EtherNet/IP	Fiber-optic SERCOS		
Feedback	• High-resolution absolute multi-turn and single-turn encoder • Incremental encoder • EnDat 2.1 and 2.2 encoders (Kinetix 6200/6500 support has not been implemented)		• High-resolution absolute multi-turn and single-turn encoder • Incremental encoder • EnDat 2.1 and 2.2 encoder support with 2090-K6CK-KENDAT feedback module • Resolver	• High-resolution absolute multi-turn and single-turn encoder • Incremental encoder • EnDat 2.1 and 2.2 encoder support with 2090-K7CK-KENDAT feedback module
	Feedback-only Auxiliary Axis			
Rotary motors compatibility	• MP-Series Low Inertia (Bulletin MPL) • MP-Series Medium Inertia (Bulletin MPM) • MP-Series Food Grade (Bulletin MPF) • MP-Series Stainless Steel (Bulletin MPS) • MP-Series RDD-Series Direct Drive (Bulletin RDB)		• MP-Series (Bulletin MPL/MPM/MPF/MPS) • RDD-Series Direct Drive (Bulletin RDB) • TL-Series (Bulletin TLY-A*xxxx*-H)	• HPK-Series • MP-Series (Bulletin MPL and MPM) • RDD-Series Direct Drive (Bulletin RDB)
Linear motors compatibility	LDC-Series Iron Core		• LDC-Series Iron Core • LDL-Series Ironless	N/A
Linear actuator compatibility	• MP-Series Linear Stages (Bulletin MPAS) • MP-Series Multi-axis Linear Stages (Bulletin MPMA) • MP-Series Electric Cylinders (Bulletin MPAR and MPAI)		• MP-Series Linear Stages (Bulletin MPAS) • MP-Series Multi-axis Stages (Bulletin MPMA) • MP-Series Electric Cylinders (Bulletin MPAR and MPAI)	N/A
Accessory compatibility	• 2094 Line Interface Modules • 2090 Resistive Brake Modules • 1394 External Passive Shunt		• 2094 Line Interface Modules • 2090 Resistive Brake Modules • 1394 External Passive Shunt • 1336 External Active Shunt	• 8720MC Regenerative PS • 8720MC Line Reactor • 1336 External Active Shunt • 2094 Line Interface Modules

(1) Refer to Kinetix 6000 IAM/AM Module Series Change on page 276 for more information.

Servo Drives

Drive Features	Kinetix 2000	Kinetix 300	Kinetix 3
Main Characteristics	• Low-power SERCOS interface solution for complex motion applications • Multi-axis • Common Bus • Integrated Motion Drive	• Single-axis solution for low-complexity motion applications • Low cost EtherNet/IP network solution with integrated safe torque-off functionality • Flexible control architecture for simple analog, PTO, or EtherNet/IP Indexing control • 120V models drive 240V motors at full speed • Memory module for Automatic Device Replacement (ADR) Safe Torque-off Control, ISO-13849-1 Certified, PLd, category 3	• Single-axis solution for low-complexity motion applications, with or without a PLC. • Indexing, analog, preset velocity, and pulse-train command modes • Modbus-RTU or I/O control • Performs indexing on up to 64 points
Drive Configuration	1…8 Axes on Bulletin 2093 Power Rail	Single-axis	
Input Voltage	170…264V AC, (230V nom) single-phase or three-phase	• 120V AC, single-phase • 240V AC, single-phase with integrated AC (EMC) line filter • 120V/240V AC, single-phase or three-phase • 480V AC, three-phase	170…264V AC, (230V nom) single-phase or three-phase
Common Bus Follower Input Voltage	240…375V DC (230V systems)	N/A	
Continuous Output Power	2.0 kW (single-phase input)	0.4…1.7 kW (single-phase input)	50 W…1.50 kW
		0.5…3.0 kW (single-phase or three-phase input)	
	3.0 kW (three-phase input)	1.0…3.0 kW (three-phase input)	
Continuous Output Current	1.0…9.5 A rms	2.0…12.0 A rms	0.61…9.90 A rms
Drive Digital Inputs	• Enable, Home, OverTravel ± • High Speed Registration (2/axis)	• Enable and OverTravel ± • High Speed Registration (1) • Eight configurable inputs	• Pulse train and analog inputs • Dedicated E-stop input • Ten configurable inputs
Drive Digital Outputs	Motor Brake Relay Output (with suppression)	• Ready • Four configurable outputs	• Servo alarm • Six configurable outputs
DPI Connector	• DriveExplorer • DPI HIM	N/A	
Programming	RSLogix 5000 Software (Ladder Logic, Structured Text, and Sequential Function Charts)	• Built-in Web server for configuration and diagnostics • RSLogix 5000 Software (Ladder Logic, Structured Text, and Sequential Function Charts)	• Ultraware Software • RSLogix 500 Software if using Modbus-RTU control
Logix Module Compatibility	• 1756-M03SE, 1756-M08SE, 1756-M16SE • 1768-M04SE	• 1769-L23*x* or 1769-L3*x* controller with integrated EtherNet/IP port • 1768-L4*x* controller with 1768-ENBT or 1768-EWEB module • 1766-L32*x* controller with integrated EtherNet/IP port	• 1766-L32*xxx* controller with RS-232 port • 1763-L16*xxx* controller with RS-232 port
I/O Control	Fiber-optic SERCOS	EtherNet/IP	Digital inputs
Feedback	• High-resolution absolute multi-turn and single-turn encoder • Incremental Encoder		
	Feedback-only Auxiliary Axis	Auxiliary axis for master gearing mode	N/A
Rotary Motors Compatibility	• MP-Series (Bulletin MPL/MPM/MPF/MPS) • TL-Series (Bulletin TLY)		TL-Series (Bulletin TL and TLY)
Linear Motors Compatibility	• LDC-Series Iron Core • LDL-Series Ironless	N/A	• LDC-Series Iron Core • LDL-Series Ironless
Linear Actuator Compatibility	• MP-Series Electric Cylinders • MP-Series Heavy Duty Electric Cylinders • TL-Series Electric Cylinders • MP-Series Linear Stages		• TL-Series Electric Cylinders • MP-Series Linear Stages (direct drive)
Accessory Compatibility	• 2094 Line Interface Modules • 1336 External Active Shunt	• 2097 I/O Terminal Expansion Block • 2097 Memory Module Programmer • 2097 AC (EMC) Line Filters • 2097 Shunt Resistors	• 2071 I/O Breakout Board • 2071 Motor Feedback Breakout Board • 2090 Control and Configuration Cables

Servo Drives

Drive Features	Ultra3000 (SERCOS)	Ultra3000 (Non-SERCOS)	Ultra5000
Main characteristics	• Standalone • SERCOS Interface • Integrated Motion Drive	• Standalone • Indexing • DeviceNet with Indexing	• Standalone • Integrated Control • ANSI C Programmable
Drive configuration	Single-axis		Single-axis
Input voltage	100…240V AC, Single-phase (230V systems)		
	240V AC, Three-phase (230V systems)		
	230/480V AC, Three-phase (460V systems)		
Common-bus follower input voltage	N/A		
Continuous output power	0.5…15 kW (230V systems)		
	3…22 kW (460V systems)		
Continuous output current	1.8…45.9 A rms (230V systems)		
	5.0…33.2 A rms (460V systems)		
Drive digital inputs	• Enable, Home, OverTravel ± • High Speed Registration (2)	8 Digital Inputs (sinking)	16 Digital Inputs (sinking or sourcing)
Drive digital outputs	Motor Brake Relay Output	• 4 Digital Outputs • Normally Open Relay Output	• 7 Digital Outputs • Normally Open Relay Output
DPI connector	N/A		
Programming	RSLogix 5000 software (Ladder Logic, Structured Text, and Sequential Function Charts)	Ultraware software	
Logix module compatibility	• 1756-M03SE • 1756-M08SE • 1756-M16SE • 1768-M04SE	• 1756-M02AE • 1756-HYD02 • 1756-M02AS	N/A
I/O control	Fiber-optic SERCOS	• Analog ± 10V • Preset Positions • Stepper/Follower • DeviceNet interface • RS-485	• Preset Positions • Stepper/Follower • DeviceNet interface • RS-485
Feedback	• High-resolution absolute (multi-turn) encoder • High-resolution (single-turn) encoder • Incremental encoder		
Rotary motors compatibility	• MP-Series Low Inertia (Bulletin MPL) • MP-Series Medium Inertia (Bulletin MPM) • MP-Series Food Grade (Bulletin MPF) • MP-Series Stainless Steel (Bulletin MPS) • TL-Series (Bulletin TLY-A*xxxx*-H)		• MP-Series (Bulletin MPL-*x*3*xxx*…MPL-*x*9*xxx*) • MP-Series (Bulletin MPM/MPF/MPS) • TL-Series (Bulletin TLY-A*xxxx*-H)
Linear motors compatibility	• LDC-Series • LDL-Series		N/A
Linear actuator compatibility	• MP-Series Linear Stages (Bulletin MPAS) • MP-Series Multi-axis Linear Stages (Bulletin MPMA) • MP-Series Electric Cylinders (Bulletin MPAR and MPAI)		N/A
Accessory compatibility	• 2090 External Shunt Modules • 2090 Resistive Brake Modules		

Additional Resources

Resource	Description	
Literature Library, website http://www.rockwellautomation.com/literature	Electronic copies of installation instructions, user manuals, and other publications related to your Kinetix Motion Control selection process.	
Hardcopy motion control publications	Your local Rockwell Automation distributor or sales representative.	
Kinetix 6200 and Kinetix 6500 Safe Speed Monitoring Servo Drives Safety Reference Manual, publication 2094-RM001	Information on wiring, configuring, and troubleshooting the safe-speed features of your Kinetix 6200 and Kinetix 6500 drives.	
Kinetix 6200 and Kinetix 6500 Safe Torque-off Servo Drives Safety Reference Manual, publication 2094-RM002	Information on wiring, configuring, and troubleshooting the safe torque-off features of your Kinetix 6200 and Kinetix 6500 drives.	
Kinetix Safe-off Feature Safety Reference Manual, publication GMC-RM002	Information on wiring and troubleshooting your Kinetix 6000 and Kinetix 7000 servo drives with the safe-off feature.	
System Design for Control of Electrical Noise Reference Manual, publication GMC-RM001	Information, examples, and techniques designed to minimize system failures caused by electrical noise.	
EMC Noise Management DVD, publication GMC-SP004		
ControlLogix Selection Guide, publication 1756-SG001	Information to determine which ControlLogix controller fits your application and the product specifications to help design a ControlLogix system and select the appropriate components.	
CompactLogix Selection Guide, publication 1769-SG001	Information to determine which CompactLogix controller fits your application and the product specifications to help design a CompactLogix system and select the appropriate components.	
Integrated Architecture Recommended Literature Reference Manual, publication IASIMP-RM001	This document provides lists of technical publications for Integrated Architecture products. These lists are not all-inclusive, but they do include the most-commonly accessed publications for the related products.	
Industrial Ethernet Media Brochure, publication 1585-BR001	Information to determine which Bulletin 1585 Ethernet cable fits your application and the product specifications to help select the appropriate components.	
Technical Support	Phone	440-646-5800
	Fax	440-646-5801
	Email	RACleAsktheExpert@ra.rockwell.com
Motion Analyzer software download from http://www.ab.com/motion/software/analyzer.html	Comprehensive motion application sizing tool used for analysis, optimization, selection, and validation of your Kinetix Motion Control system.	
Rockwell Automation Configuration and Selection Tools, website http://www.ab.com	Online product selection and system configuration tools, including AutoCad (DXF) drawings.	

Notes:

Chapter 1

Rotary Motion

Use this chapter to become familiar with the Kinetix Motion Control rotary servo motors and select the rotary motion components required for your application. To compare features from one family of rotary motion components to another, refer to Rotary Servo Motors on page 9.

Common Rotary Motor Specifications

These rotary motor specifications are common to all motor families unless otherwise noted.

Environmental Specifications

Attribute	MP-Series	RDD-Series	TL-Series	HPK-Series
Temperature, ambient	0…40 °C (32…104 °F)			
Temperature, storage	-30…70 °C (-22…158 °F)			
Relative humidity (noncondensing)	5…95%			
Shock	20 g peak, 6 ms duration			10 g peak, 6 ms duration
Vibration	2.5 g peak @ 30…2000 Hz			

Environmental Ratings

IP Rating	Dust Protection	Liquid Protection	Shaft Seal	Motor
IP50	Limited protection from dust (no harmful deposit).	No protection from liquids.	Motor without shaft seal.	Bulletin MPL and MPM
IP54		Protected against sprays from all directions.		TL-Series, HPK-Series
IP65	Total protection from dust.	Protected against low-pressure jets of water from all directions.	Motor with shaft seal (cable connectors rated IP54).	TL-Series
			Motor with shaft seal and Bulletin 2090 environmentally sealed cable connectors.	RDD-Series (1)
IP66		Protected against strong jets of water.		Bulletin MPL, MPF, and MPS
IP67		Protected against the effects of temporary liquid immersion.		Bulletin MPF MPS, and MPM
IP69K		Protected against the effects of water/stream jets up to 100 bar (1200 psi) with nozzle temperature at approximately 80 °C (176 °F).		Bulletin MPS

(1) Environmental rating applies from mounting face to rear of motor. Customer is responsible for providing environmental protection to mounting face of motor.

Motor Brake Application Guidelines

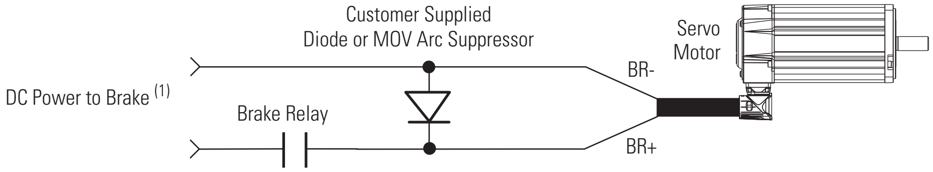
The brakes offered as options on these servo motors are holding brakes designed to hold the motor shaft at 0 rpm up to the rated brake holding torque. The brakes release when voltage is applied to the brake coil. Voltage and polarity supplied to the brake must be as specified to be sure of proper brake performance.

The brakes are not designed for stopping rotation of the motor shaft. Servo drive inputs should be used to stop motor shaft rotation. The recommended method of stopping motor shaft rotation is to command the servo drive to decelerate the motor to 0 rpm, and engage the brake after the servo drive has decelerated the motor to 0 rpm.

If system mains power fails, the brakes can withstand use as stopping brakes. However, use of the brakes as stopping brakes creates rotational mechanical backlash that is potentially damaging to the system, increases brake pad wear, and reduces brake life. The brakes are not designed nor are they intended to be used as a safety device.

A separate power source is required to disengage the brake. This power source can be controlled by the servo motor controls, in addition to manual operator controls. Electrical arcing may occur at the relay contacts until the brake power dissipates. A customer supplied diode or metal oxide varistor (MOV) is recommended to prevent arcing. Use of an MOV can also reduce the time to mechanically engage the brake. For brake response time specifications, refer to the motor brake specification tables.

Suppression Device for Brake Relay Contacts



(1) Kinetix 2000, Kinetix 6000, Kinetix 6200, Kinetix 6500, and Kinetix 7000 servo drives provide motor brake relay outputs and supply an MOV arc suppressor, so customer supplied arc suppressor is not required unless the coil current of motor brake is greater than the maximum brake current rating of the drive relay output.

MP-Series Low Inertia Motors

MP-Series low-inertia (Bulletin MPL), high output brushless servo motors utilize innovative design characteristics to reduce motor size while delivering significantly higher torque. These compact and highly dynamic brushless servo motors are designed by Allen-Bradley to meet the demanding requirements of high performance motion systems.

For drive compatibility, refer to Servo Drives on page 14.

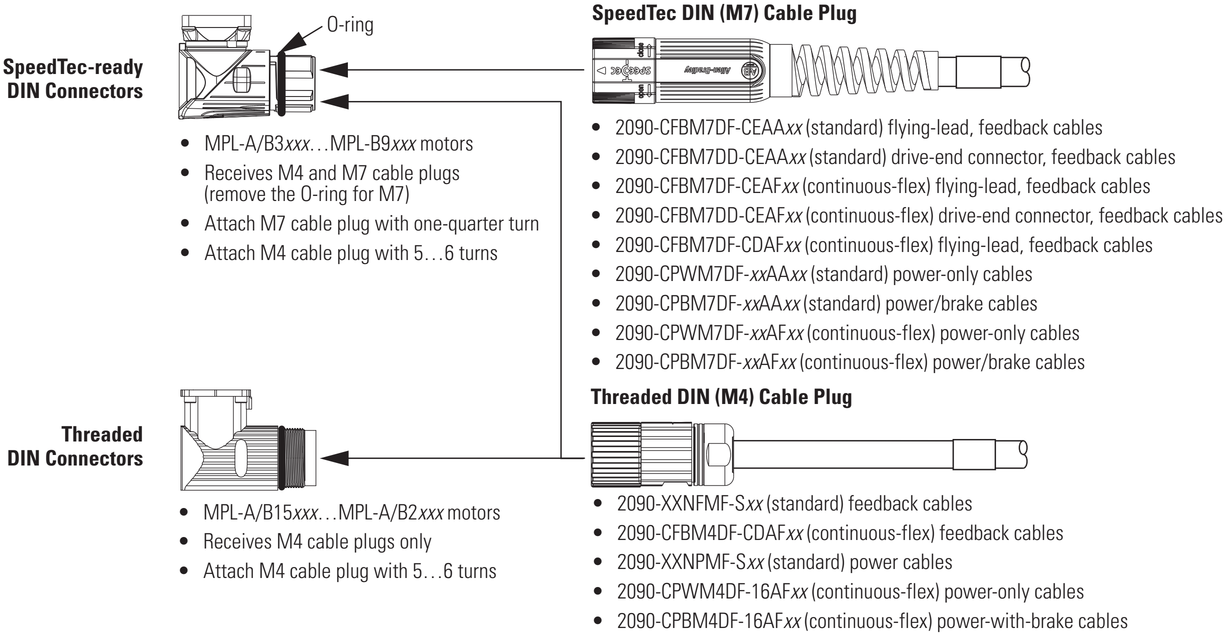
MP-Series Low Inertia Motor High Resolution Encoder Features

MP-Series low-inertia motors are available with high performance encoders with a choice of Single-turn (-E, -S) or Multi-turn (-V, -M) high resolution feedback.

- Up to 2 million counts per revolution (-M and -S) for smooth performance (MPL-A/B3*xx*, MPL-A/B4*xx*, MPL-A/B45*xx*, MPL-A/B5*xx*, MPL-B6*xx*, MPL-B8*xx*, and MPL-B9*xx* motors).
- Up to 260 thousand counts per revolution (-E and -V) for smooth performance (MPL-A/B15*xx* and MPL-A/B2*xx* motors).
- Single-turn encoder provides high-resolution absolute position feedback within one turn.
- Multi-turn encoder provides high-resolution absolute position feedback within 4096 turns. The electromechanical design does not require a battery.

Motor Connector/Cable Compatibility

MP-Series (Bulletin MPL) motors are equipped with threaded and SpeedTec-ready DIN connectors.



SpeedTec-ready DIN Connectors

- MPL-A/B3*xxx*…MPL-B9*xxx* motors
- Receives M4 and M7 cable plugs (remove the O-ring for M7)
- Attach M7 cable plug with one-quarter turn
- Attach M4 cable plug with 5…6 turns

SpeedTec DIN (M7) Cable Plug

- 2090-CFBM7DF-CEAA*xx* (standard) flying-lead, feedback cables
- 2090-CFBM7DD-CEAA*xx* (standard) drive-end connector, feedback cables
- 2090-CFBM7DF-CEAF*xx* (continuous-flex) flying-lead, feedback cables
- 2090-CFBM7DD-CEAF*xx* (continuous-flex) drive-end connector, feedback cables
- 2090-CFBM7DF-CDAF*xx* (continuous-flex) flying-lead, feedback cables
- 2090-CPWM7DF-*xx*AA*xx* (standard) power-only cables
- 2090-CPBM7DF-*xx*AA*xx* (standard) power/brake cables
- 2090-CPWM7DF-*xx*AF*xx* (continuous-flex) power-only cables
- 2090-CPBM7DF-*xx*AF*xx* (continuous-flex) power/brake cables

Threaded DIN Connectors

- MPL-A/B15*xxx*…MPL-A/B2*xxx* motors
- Receives M4 cable plugs only
- Attach M4 cable plug with 5…6 turns

Threaded DIN (M4) Cable Plug

- 2090-XXNFMF-S*xx* (standard) feedback cables
- 2090-CFBM4DF-CDAF*xx* (continuous-flex) feedback cables
- 2090-XXNPMF-S*xx* (standard) power cables
- 2090-CPWM4DF-16AF*xx* (continuous-flex) power-only cables
- 2090-CPBM4DF-16AF*xx* (continuous-flex) power-with-brake cables

For information on transitioning your Bulletin MPL motor installation from bayonet cables to circular DIN cables, refer to 2090-Series Motor Power and Feedback Transition Cables on page 399.

MP-Series Low Inertia Motor Options

MP-Series low-inertia motors are available with these options:

- 24V DC brake.
- Shaft seal kit available for field installation. Shaft seals are made of nitrile. Kits include a lubricant to reduce wear.
- Optional keyless shaft available in limited frame sizes with extended lead times (MPL-A/B3*xx*, MPL-A/B4*xx*, MPL-A/B45*xx*, and MPL-A/B5*xx* motors).

Motor Shaft Seal Kit Combinations and Dimensions

Motor Series	**Shaft Seal Cat. No.**	**Inside Diameter** mm (in.)	**Outside Diameter** mm (in.)	**Width** mm (in.)
MPL-A15*xx* and MPL-B15*xx*	MPL-SSN-F63F75	12 (0.47)	24 (0.95)	7 (0.28)
MPL-A2*xx* and MPL-B2*xx*				
MPL-A3*xx* and MPL-B3*xx*	MPL-SSN-A3B3	17 (0.67)	47 (1.85)	7 (0.28)
MPL-A4*xx* and MPL-B4*xx*	MPL-SSN-A4B4	20 (0.79)	52 (2.05)	7 (0.28)
MPL-A45*xx* and MPL-B45*xx*	MPL-SSN-A5B5	25 (0.98)	62 (2.44)	7 (0.28)
MPL-A520 and MPL-B520 MPL-A540 and MPL-B540 MPL-A560 and MPL-B560	MPL-SSN-F165	30 (1.18)	72 (2.83)	8 (0.31)
MPL-B580	MPL-SSN-F165-32MM	35 (1.38)	72 (2.83)	8 (0.31)
MPL-B6*xx*	MPL-SSN-A6B6	40 (1.57)	90 (3.54)	8 (0.31)
MPL-B8*xx*	MPL-SSN-A8B8	45 (1.77)	75 (2.95)	8 (0.31)
MPL-B9*xx*	MPL-SSN-A9B9	52 (2.05)	72 (2.83)	8 (0.31)

MP-Series Low Inertia Motor Performance Specifications

MP-Series Low Inertia Motor (230V) Performance Specifications

Motor MPL-	**Max Speed** rpm	**Continuous Stall Torque** Nm (lb-in)	**Peak Stall Torque** Nm (lb-in)	**Motor Rated Output** kW	**Speed at Motor Rated Output** rpm	**Rotor Inertia** (1) kg-m^2 (lb-in-s^2)	**Motor Weight, approx.** (1) kg (lb)
A1510V	8000	0.26 (2.3)	0.77 (6.8)	0.16	8000	0.0000074 (0.000065)	1.0 (2.2)
A1520U	7000	0.49 (4.3)	1.58 (14)	0.27	7000	0.000013 (0.00012)	1.2 (2.6)
A1530U	7000	0.90 (8.0)	2.80 (25)	0.39	7000	0.000023 (0.00020)	1.6 (3.4)
A210V	8000	0.55 (4.9)	1.50 (13.5)	0.37	8000	0.000015 (0.00013)	1.4 (3.1)
A220T	6000	1.61 (14.2)	4.74 (42)	0.62	6000	0.000039 (0.00035)	2.0 (4.4)
A230P	5000	2.10 (18.6)	8.20 (73)	0.86	5000	0.000063 (0.00056)	2.6 (5.7)
A310P	5000	1.58 (14)	3.61 (32)	0.73	4750	0.000044 (0.00039)	2.7 (5.8)
A310F	3000	1.58 (14)	3.61 (32)	0.46	3000		
A320P	5000	3.05 (27)	7.91 (70)	1.3	4750	0.000078 (0.00069)	3.7 (8.0)
A320H	3500	3.05 (27)	7.91 (70)	1.0	3350		
A330P	5000	4.18 (37)	11.1 (98)	1.8	5000	0.00012 (0.0010)	4.6 (10)
A420P	5000	4.74 (42)	10.2 (90)	2.0	5000	0.00026 (0.0023)	4.3 (9.4)
A430P	5000	5.99 (53)	19.8 (175)	2.2	5000	0.00038 (0.0033)	5.5 (12)
A430H	3500	6.21 (55)	19.8 (175)	1.8	3500		
A4530K	4000	8.13 (72)	20.3 (180)	2.5	4000	0.00040 (0.0036)	7.3 (16)
A4530F	2800	8.36 (74)	20.3 (180)	1.9	2800		
A4540C	1500	10.2 (90)	27.1 (240)	1.5	1500	0.00052 (0.0046)	8.6 (19)
A4540F	3000	10.2 (90)	27.1 (240)	2.6	3000		
A4560F	3000	14.1 (125)	34.4 (305)	3.0	3000	0.00078 (0.0067)	11.82 (26)
A520K	4000	10.7 (95)	24.3 (215)	3.5	3500	0.000783 (0.0069)	9.8 (21.5)
A540K	4000	19.4 (172)	48.6 (430)	5.5	4000	0.00147 (0.013)	15.0 (33)
A560F	3000	26.8 (237)	61.0 (540)	5.3	3000	0.00213 (0.019)	20.2 (44.5)

(1) Refer to MP-Series Low Inertia Motor Brake Specifications on page 25 for Brake Rotor Inertia and Brake Motor Weight.

System Combinations (230V)

For MP-Series Low Inertia Motors and	**Refer to**
Kinetix 6000 (230V) drives	page 513
Kinetix 2000 (230V) drives	page 559
Kinetix 300 (240V) drives	page 595
Ultra3000/5000 (230V) drives (1)	page 627

(1) MPL-A15*xxx* and MPL-A2*xxx* motors are not compatible with Ultra5000 drives. All other MPL-A*xxxx* motors are compatible with both Ultra3000 and Ultra5000 drives.

MP-Series Low Inertia Motor (460V) Performance Specifications

Motor MPL-	**Max Speed** rpm	**Continuous Stall Torque** Nm (lb-in)	**Peak Stall Torque** Nm (lb-in)	**Motor Rated Output** kW	**Speed at Motor Rated Output** rpm	**Rotor Inertia** (1) kg-m^2 (lb-in-s^2)	**Motor Weight, approx.** (1) kg (lb)
B1510V	8000	0.26 (2.3)	0.77 (6.8)	0.16	8000	0.0000074 (0.000065)	1.0 (2.2)
B1520U	7000	0.49 (4.3)	1.58 (14)	0.27	7000	0.000013 (0.00012)	1.2 (2.6)
B1530U	7000	0.90 (8.0)	2.80 (25)	0.39	7000	0.000023 (0.00020)	1.6 (3.4)
B210V	8000	0.55 (4.9)	1.50 (13.5)	0.37	8000	0.000015 (0.00013)	1.4 (3.1)
B220T	6000	1.61 (14.2)	4.74 (42)	0.62	6000	0.000039 (0.00035)	2.0 (4.4)
B230P	5000	2.10 (18.6)	8.20 (73)	0.86	5000	0.000063 (0.00056)	2.6 (5.7)
B310P	5000	1.58 (14)	3.61 (32)	0.77	5000	0.000044 (0.00039) (2)	2.7 (5.8)
B320P	5000	3.05 (27)	7.91 (70)	1.5	5000	0.000078 (0.00069) (2)	3.7 (8.0)
B330P	5000	4.18 (37)	11.1 (98)	1.8	5000	0.00012 (0.0010) (2)	4.6 (10)
B420P	5000	4.74 (42)	13.5 (120)	1.9	5000	0.00026 (0.0023) (2)	4.3 (9.4)
B430P	5000	6.55 (58)	19.8 (175)	2.2	5000	0.00038 (0.0033) (2)	5.5 (12)
B4530F	3000	8.25 (73)	20.3 (180)	2.1	3000	0.00040 (0.0036) (2)	7.3 (16)
B4530K	4000	8.25 (73)	20.3 (180)	2.6	4000		
B4540F	3000	10.2 (90)	27.1 (240)	2.6	3000	0.00052 (0.0046) (2)	8.6 (19)
B4560F	3000	14.1 (125)	34.4 (305)	3.2	3000	0.00078 (0.0067) (2)	11.82 (26)
B520K	4000	10.7 (95)	23.2 (205)	3.5	3500	0.000783 (0.0069)	9.8 (21.5)
B540D	2000	19.4 (172)	41.0 (362)	3.4	2000	0.00147 (0.013)	15 (33)
B540K	4000	19.4 (172)	48.6 (430)	5.4	4000	0.00147 (0.013)	
B560F	3000	26.8 (237)	67.8 (600)	5.5	3000	0.00213 (0.019)	20.2 (44.5)
B580F	3000	34.0 (301)	87.0 (770)	7.1	3000	0.00289 (0.023)	25.4 (56)
B580J	3800	34.0 (301)	81.0 (716)	7.9	3800		
B640F	3000	36.7 (325)	72.3 (640)	6.11	2000	0.004 (0.0354)	26.8 (59)
B660F	3000	48.0 (425)	101.1 (895)	6.15	2000	0.0058 (0.051)	35.0 (77)
B680D	2000	62.8 (556)	154.2 (1365)	9.3	2000	0.00775 (0.0685)	40.4 (89)
B680F	3000	60.0 (531)	108.5 (960)	7.5	2000		
B860D	2000	83.0 (735)	152.5 (1350)	12.5	2000	0.0169 (0.150)	57.3 (126)
B880C	1500	110.0 (973)	203 (1800)	12.6	1500	0.0224 (0.198)	72.7 (160)
B880D	2000	110.0 (973)	147 (1300)	12.6	2000		
B960B	1200	130.0 (1150)	231 (2050)	12.7	1200	0.0273 (0.242)	76.0 (167)
B960C	1500	124.3 (1100)	226 (2000)	14.8	1500		
B960D	2000	124.3 (1100)	226 (2000)	15.0	2000		
B980B	1000	162.7 (1440)	278 (2460)	15.2	1000	0.0354 (0.313)	94.5 (208)
B980C	1500	158.2 (1400)	271 (2400)	16.8	1500		
B980D	2000	158.2 (1400)	260 (2300)	18.6	2000		
B980E	2750	141.0 (1250)	237 (2100)	13.0	1500		

(1) Refer to MP-Series Low Inertia Motor Brake Specifications on page 25 for Brake Rotor Inertia and Brake Motor Weight.

(2) Rotor inertia may vary slightly depending on feedback.

System Combinations (460V)

For MP-Series Low Inertia Motors and	Refer to
Kinetix 6000 and Kinetix 6200/6500 (460V) drives	page 519
Kinetix 7000 drives	page 581
Kinetix 300 (480V) drives	page 600
Ultra3000/5000 (460V) drives (1)	page 633

(1) MPL-B15*xxx* and MPL-B2*xxx* motors are not compatible with Ultra5000 drives. All other MPL-B*xxxx* motors are compatible with both Ultra3000 and Ultra5000 drives.

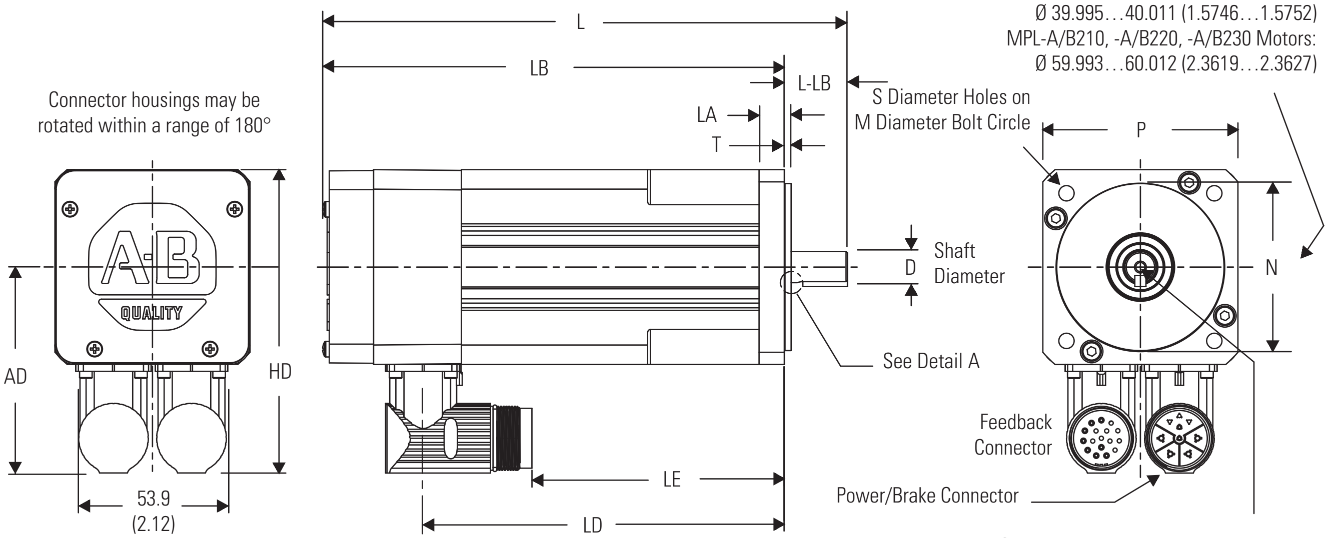
MP-Series Low Inertia Motor Brake Specifications

Motor MPL-	**Max Backlash (brake engaged)** arc minutes	**Holding Torque** Nm (lb-in)	**Coil Current at 24V DC** A	**Brake Response Time**			**Brake Rotor Inertia** kg-m^2 (lb-in-s^2)	**Brake Motor Weight, approx.** kg (lb)
				Release ms	**Engage (using external arc suppression device)**			
					MOV ms	**Diode** ms		
A/B1510V	0	0.9 (8.0)	0.43…0.53	23	9	18	0.0000099 (0.000088)	1.2 (2.6)
A/B1520U							0.000015 (0.00013)	1.4 (3.1)
A/B1530U							0.000026 (0.00023)	1.8 (3.9)
A/B210V		4.5 (40)	0.46…0.56	58	20	42	0.000033 (0.00029)	1.8 (4.0)
A/B220T							0.000057 (0.00050)	2.4 (5.4)
A/B230P							0.000082 (0.00073)	3.0 (6.7)
A/B310	45	4.18 (37)	0.45…0.55	50	20	110	0.000057 (0.00050)	3.7 (8)
A/B320							0.000092 (0.00081)	4.6 (10)
A/B330							0.00013 (0.0011)	5.6 (12.4)
A/B420	37	10.2 (90)	0.576…0.704	110	25	160	0.00030 (0.0027)	6.0 (13.2)
A/B430							0.00042 (0.0038)	7.3 (16)
A/B4530							0.00044 (0.0039)	9.1 (20)
A/B4540							0.00056 (0.0050)	11.0 (24)
A/B4560							0.00084 (0.0072)	15.1 (33.2)
A/B520	25	28.3 (250)	1.05…1.28	70	50	250	0.000897 (0.0079)	12.38 (27.25)
A/B540							0.00157 (0.0139)	17.6 (38.75)
A/B560							0.00227 (0.020)	22.8 (50.1)
B580							0.0030 (0.026)	29.0 (63.8)
B640	25	70.0 (619)	1.91…2.19	200	120	900	0.00438 (0.03863)	37.27 (82.0)
B660							0.00628 (0.0555)	42.95 (94.5)
B680							0.0079 (0.0698)	50.8 (112.0)
B860		106.0 (938)	2.05…2.50	250	200	1000	0.0177 (0.1570)	72.7 (160)
B880							0.0232 (0.205)	87.7 (193)
B960		153.0 (1350)	3.85…4.70	300	200	1200	0.0290 (0.256)	89.5 (197)
B980							0.0378 (0.334)	116.5 (256)

MP-Series Low Inertia Motor Dimensions

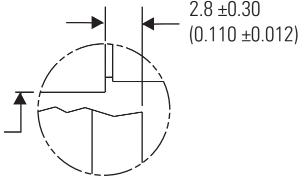
MPL-A/B15*xx* and MPL-A/B2*xx* Motor Dimensions (threaded DIN connectors)

Dimensions are in mm (in.)



Pilot Diameter Tolerances
MPL-A/B1510, -A/B1520, -A/B1530 Motors:
Ø 39.995…40.011 (1.5746…1.5752)
MPL-A/B210, -A/B220, -A/B230 Motors:
Ø 59.993…60.012 (2.3619…2.3627)

Detail A



MPL-A/B1510, -A/B1520, -A/B1530 and
MPL-A/B210, -A/B220, -A/B230 Motors:
Ø 11.89…11.95 (0.468…0.470)

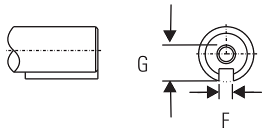
Shaft Diameter Tolerances
MPL-A/B1510, -A/B1520, -A/B1530 Motors:
Ø 8.998…9.007 (0.3543…0.3546)

MPL-A/B210, -A/B220, -A/B230 Motors:
Ø 10,997…11.008 (0.4330…0.4334)

Shaft End Threaded Hole
MPL-A/B1510, -A/B1520, -A/B1530 Motors:
Thread - M3 x 0.5-6H
Thread Depth - 9.0 (0.35)
MPL-A/B210, -A/B220, -A/B230 Motors:
Thread - M4 x 0.7-6H
Thread Depth - 10.0 (0.39)

Optional Shaft Seal
Refer to page 22
for MP-Series
small frame motor
shaft seal kit information.

Shaft Detail with Key



MPL-A/B1510, -A/B1520, -A/B1530 = 3 x 3 x 14 Key
MPL-A/B210, -A/B220, -A/B230 = 4 x 4 x 16 Key

Shaft, Pilot, and Keyway Tolerances	MPL-A/B15*xx*	MPL-A/B2*xx*
Shaft Runout (T.I.R.)	0.030 (0.0012)	0.035 (0.0014)
Pilot Eccentricity (T.I.R.)	0.08 (0.0031)	0.08 (0.0031)
Max Face Runout (T.I.R.)	0.08 (0.0031)	0.08 (0.0031)
Keyway Depth (G)	7.10…7.20 (0.280…0.283)	8.40…8.50 (0.331…0.335)
Keyway Width (F)	2.971…2.996 (0.117…0.118)	3.97…4.00 (0.156…0.157)

MPL-A/B15*xx* and MPL-A/B2*xx* Motor Dimensions (threaded DIN connectors)

Motor Series MPL-A or MPL-B	**AD** mm (in.)	**HD** mm (in.)	**T** mm (in.)	**LA** mm (in.)	**LD** (1) mm (in.)	**LE** (1) mm (in.)	**L** (1) mm (in.)	**LB** (1) mm (in.)	**L-LB** (2) mm (in.)	**D** mm (in.)	**M** mm (in.)	**S** (3) mm (in.)	**N** mm (in.)	**P** mm (in.)	**G** mm (in.)	**F** mm (in.)
1510	66.5 (2.62)	94.0 (3.70)	2.50 (0.098)	9.0 (0.35)	78.3 (3.08)	39.2 (1.54)	133.2 (5.25)	113.5 (4.47)	19.7 (0.776)	9.0 (0.35)	63.0 (2.480)	5.80 (0.228)	40.0 (1.57)	55.0 (2.17)	7.2 (0.283)	3.0 (0.118)
1520					91.3 (3.60)	52.4 (2.06)	146.2 (5.76)	126.5 (4.98)								
1530					116.3 (4.58)	77.2 (3.04)	171.2 (6.74)	151.5 (5.96)								
210	74.0 (2.91)	109 (4.29)	2.50 (0.098)	9.0 (0.35)	78.6 (3.09)	39.6 (1.56)	137.3 (5.40)	114.6 (4.51)	22.7 (0.894)	11.0 (0.43)	75.0 (2.953)	5.80 (0.228)	60.0 (2.36)	70.0 (2.76)	8.5 (0.335)	4.0 (0.157)
220					104.1 (4.10)	65.1 (2.56)	162.8 (6.41)	140.1 (5.52)								
230					129.6 (5.10)	90.6 (3.57)	188.3 (7.41)	165.6 (6.52)								

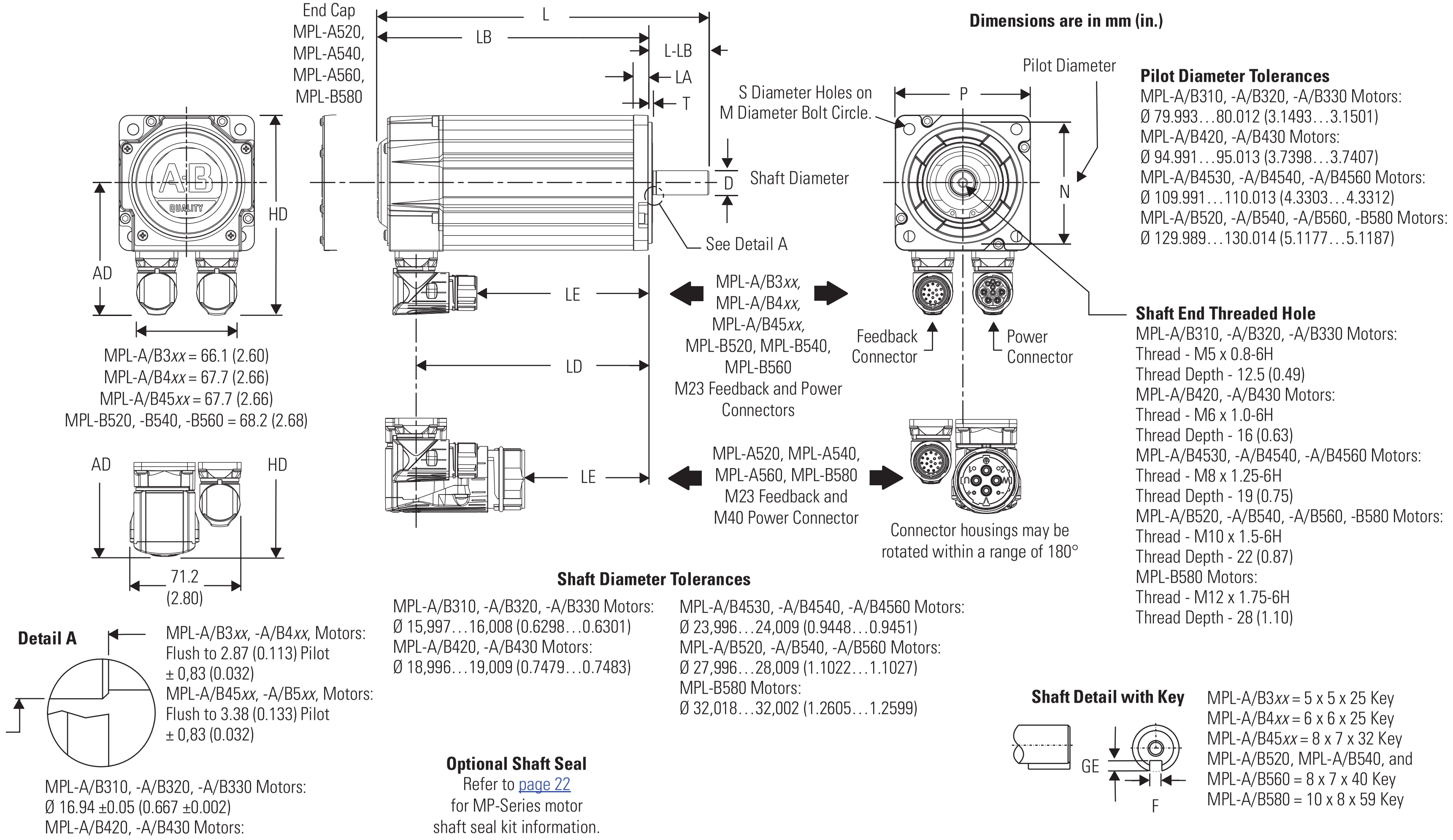
(1) If ordering an MPL-A/B1510, A/B1520, or A/B1530 motor with brake, add 36.1 mm (1.421 in.) to dimension L and LB, and add 33.4 mm (1.32 in.) to LD and LE. If ordering an MPL-A/B210, A/B220, or A/B230 motor with brake, add 39.0 mm (1.535 in.) to dimension L and LB, and add 24.7 mm (0.97 in.) to LD and LE.

(2) Tolerance for this dimension is ±0.7 mm (±0.028 in.).

(3) Tolerance for this dimension is +0.3 mm (+0.012 in.).

Motors are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

MPL-A/B3*xx*, MPL-A/B4*xx*, MPL-A/B45*xx*, MPL-A/B5*xx* Motor Dimensions (SpeedTec DIN connectors)



Pilot Diameter Tolerances
MPL-A/B310, -A/B320, -A/B330 Motors:
Ø 79.993…80.012 (3.1493…3.1501)
MPL-A/B420, -A/B430 Motors:
Ø 94.991…95.013 (3.7398…3.7407)
MPL-A/B4530, -A/B4540, -A/B4560 Motors:
Ø 109.991…110.013 (4.3303…4.3312)
MPL-A/B520, -A/B540, -A/B560, -B580 Motors:
Ø 129.989…130.014 (5.1177…5.1187)

Shaft End Threaded Hole
MPL-A/B310, -A/B320, -A/B330 Motors:
Thread - M5 x 0.8-6H
Thread Depth - 12.5 (0.49)
MPL-A/B420, -A/B430 Motors:
Thread - M6 x 1.0-6H
Thread Depth - 16 (0.63)
MPL-A/B4530, -A/B4540, -A/B4560 Motors:
Thread - M8 x 1.25-6H
Thread Depth - 19 (0.75)
MPL-A/B520, -A/B540, -A/B560, -B580 Motors:
Thread - M10 x 1.5-6H
Thread Depth - 22 (0.87)
MPL-B580 Motors:
Thread - M12 x 1.75-6H
Thread Depth - 28 (1.10)

Detail A
MPL-A/B3*xx*, -A/B4*xx*, Motors:
Flush to 2.87 (0.113) Pilot
± 0,83 (0.032)
MPL-A/B45*xx*, -A/B5*xx*, Motors:
Flush to 3.38 (0.133) Pilot
± 0,83 (0.032)

MPL-A/B310, -A/B320, -A/B330 Motors:
Ø 16.94 ±0.05 (0.667 ±0.002)
MPL-A/B420, -A/B430 Motors:
Ø 19.94 ±0.05 (0.785 ±0.002)
MPL-A/B4530, -A/B4540, -A/B4560 Motors:
Ø 24.94 ±0.05 (0.982 ±0.002)
MPL-A/B520, -A/B540, -A/B560, -B580 Motors:
Ø 29.92 ±0.05 (1.178 ±0.002)

Shaft Diameter Tolerances
MPL-A/B310, -A/B320, -A/B330 Motors:
Ø 15,997…16,008 (0.6298…0.6301)
MPL-A/B420, -A/B430 Motors:
Ø 18,996…19,009 (0.7479…0.7483)
MPL-A/B4530, -A/B4540, -A/B4560 Motors:
Ø 23,996…24,009 (0.9448…0.9451)
MPL-A/B520, -A/B540, -A/B560 Motors:
Ø 27,996…28,009 (1.1022…1.1027)
MPL-B580 Motors:
Ø 32,018…32,002 (1.2605…1.2599)

Optional Shaft Seal
Refer to page 22
for MP-Series motor
shaft seal kit information.

Shaft Detail with Key
MPL-A/B3*xx* = 5 x 5 x 25 Key
MPL-A/B4*xx* = 6 x 6 x 25 Key
MPL-A/B45*xx* = 8 x 7 x 32 Key
MPL-A/B520, MPL-A/B540, and
MPL-A/B560 = 8 x 7 x 40 Key
MPL-A/B580 = 10 x 8 x 59 Key

Shaft, Pilot, and Keyway Tolerances	MPL-A/B3*xx*	MPL-A/B4*xx*	MPL-A/B45*xx*	MPL-A/B5*xx*	MPL-B580
Shaft Runout (T.I.R.)	0.035 (0.0014)	0.04 (0.0016)	0.04 (0.0016)	0.04 (0.0016)	0.05 (0.002)
Pilot Eccentricity (T.I.R.)	0.08 (0.0031)	0.08 (0.0031)	0.10 (0.0039)	0.10 (0.0039)	0.10 (0.0039)
Max Face Runout (T.I.R.)	0.08 (0.0031)	0.08 (0.0031)	0.10 (0.0039)	0.10 (0.0039)	0.10 (0.0039)
Keyway Depth (GE)	3.00…3.10 (0.118…0.122)	3.50…3.60 (0.138…0.142)	4.00…4.20 (0.158…0.165)	4.00…4.20 (0.158…0.165)	5.00…5.20 (0.197…0.205)
Keyway Width (F)	4.97…5.00 (0.196…0.197)	5.97…6.00 (0.235…0.236)	7.96…8.00 (0.314…0.315)	7.964…8.000 (0.3135…0.3150)	9.964…10.000 (0.3923…0.3937)

MPL-A/B3*xx*, MPL-A/B4*xx*, MPL-A/B45*xx*, MPL-A/B5*xx* Motor Dimensions (SpeedTec DIN connectors)

Motor Series MPL-	**AD** mm (in.)	**HD** mm (in.)	**T** mm (in.)	**LA** mm (in.)	**LD**[1] mm (in.)	**LE**[1] mm (in.)	**L**[1] mm (in.)	**LB**[1] mm (in.)	**L-LB**[2] mm (in.)	**D** mm (in.)	**M** mm (in.)	**S**[3] mm (in.)	**N** mm (in.)	**P** mm (in.)	**GE** mm (in.)	**F** mm (in.)
A/B310	87.2 (3.44)	132.0 (5.20)	2.74 (0.108)	9.90 (0.39)	102.0 (4.03)	62.0 (2.45)	168.0 (6.62)	128.0 (5.04)	40.0 (1.58)	16.0 (0.629)	100.0 (3.937)	7.0 (0.283)	80.0 (3.15)	89.4 (3.52)	3.0 (0.118)	5.0 (0.197)
A/B320					128.0 (5.03)	88.0 (3.45)	193.0 (7.62)	153.0 (6.04)								
A/B330					153.0 (6.03)	113.0 (4.45)	219.0 (8.62)	179.0 (7.04)								
A/B420	90.9 (3.58)	140.1 (5.52)	2.74 (0.108)	10.16 (0.40)	124.0 (4.89)	84.0 (3.31)	190.0 (7.48)	150.0 (5.90)	40.0 (1.58)	19.0 (0.748)	115.0 (4.528)	10.0 (0.401)	95.0 (3.74)	98.3 (3.87)	3.5 (0.138)	6.0 (0.236)
A/B430					150.0 (5.89)	110.0 (4.31)	215.0 (8.48)	175.0 (6.90)								
A/B4530	98.6 (3.88)	155.4 (6.12)	2.74 (0.108)	12.19 (0.48)	153.0 (6.02)	113.0 (4.44)	229.0 (9.0)	179.0 (7.03)	50.0 (1.97)	24.0 (0.945)	130.0 (5.118)	10.0 (0.401)	110.0 (4.331)	113.7 (4.48)	4.0 (0.158)	8.0 (0.315)
A/B4540					178.0 (7.02)	138.0 (5.44)	254.0 (10.0)	204.0 (8.03)								
A/B4560					229.0 (9.02)	189.0 (7.44)	305.0 (12.0)	255.0 (10.03)								
A520	136.4 (5.37)	208.1 (8.19)	3.12 (0.123)	14.0 (0.55)	151.0 (5.95)	80.0 (3.15)	236.0 (9.28)	176.0 (6.92)	60.0 (2.362)	28.0 (1.102)	165.0 (6.496)	12.0 (0.481)	130.0 (5.118)	143.5 (5.65)	4.0 (0.158)	8.0 (0.315)
A540					202.0 (7.95)	131.0 (5.15)	287.0 (11.28)	227.0 (8.92)								
A560					253.0 (9.95)	182.0 (7.15)	337.0 (13.28)	277.0 (10.92)								
B520	113.4 (4.47)	185.2 (7.29)	3.12 (0.123)	14.0 (0.55)	149.0 (5.88)	109.0 (4.30)	236.0 (9.28)	176.0 (6.92)	60.0 (2.38)	28.0 (1.102)	165.0 (6.496)	12.0 (0.481)	130.0 (5.118)	143.5 (5.65)	4.0 (0.158)	8.0 (0.315)
B540					200.0 (7.88)	160.0 (6.30)	287.0 (11.28)	227.0 (8.92)								
B560					251.0 (9.88)	211.0 (8.30)	337.0 (13.28)	277.0 (10.92)								
B580	136.4 (5.37)	208.1 (8.19)	3.12 (0.123)	14.0 (0.55)	304.0 (11.95)	232.0 (9.15)	408.0 (16.07)	328.0 (12.92)	80.0 (3.15)	32.0 (1.260)					5.0 (0.198)	10.0 (0.393)

(1) If ordering an MPL-A/B310, MPL-A/B320, or MPL-A/B330 motor with brake, add 35.0 mm (1.38 in.) to dimensions L, LB, LE, and LD.
If ordering an MPL-A/B420, MPL-A/B430, MPL-A/B4530, MPL-A/B4540, or MPL-A/B4560 motor with brake, add 48.0 mm (1.89 in.) to dimensions L, LB, LE, and LD.
If ordering an MPL-A/B520, MPL-A/B540, MPL-A/B560, or MPL-B580 motor with brake, add 52.0 mm (2.03 in.) to dimensions L, LB, LE, and LD.

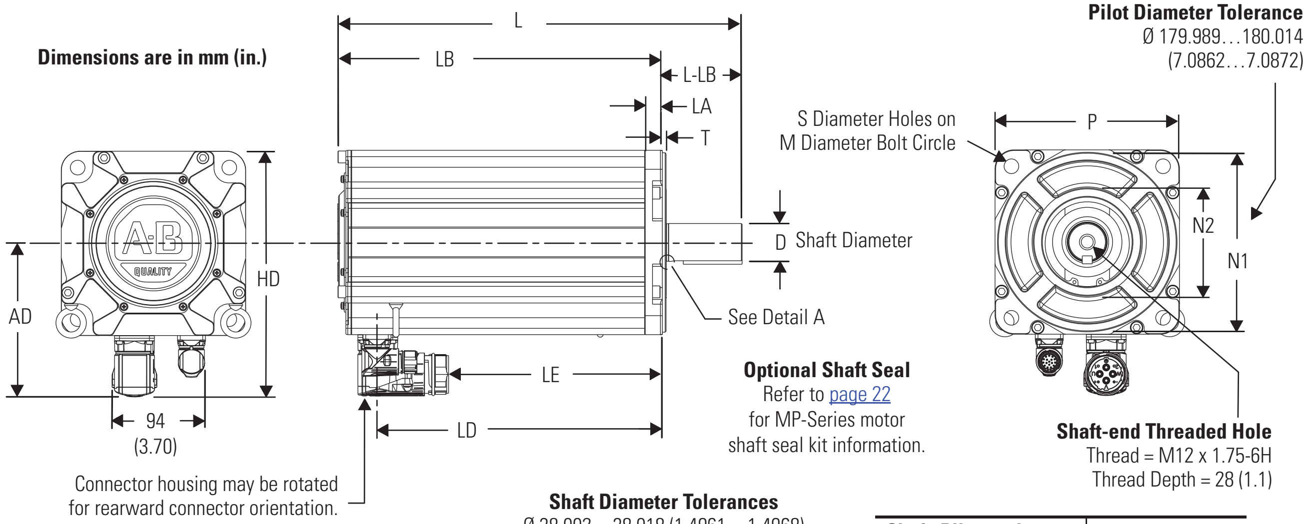
(2) Tolerance for this dimension is ±0.7 mm (±0.028 in.) in.

(3) Tolerance for this dimension is +0.36 mm (±0.007 in.) on MPL-A/B3*xx*, MPL-A/B4*xx*, MPL-A/B45*xx*, and +0.43 mm (±0.008 in.) on MPL-A/B5*xx*.

Motors are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

MPL-B6*xx* Motor Dimensions (SpeedTec DIN connectors)

Dimensions are in mm (in.)



Pilot Diameter Tolerance
Ø 179.989…180.014
(7.0862…7.0872)

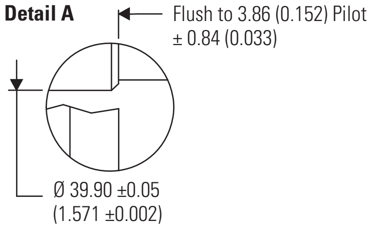
Connector housing may be rotated for rearward connector orientation.

Optional Shaft Seal
Refer to page 22
for MP-Series motor
shaft seal kit information.

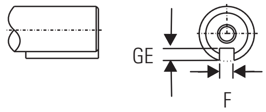
Shaft-end Threaded Hole
Thread = M12 x 1.75-6H
Thread Depth = 28 (1.1)

Shaft Diameter Tolerances
Ø 38.002…38.018 (1.4961…1.4968)

Detail A



Shaft Detail with Key



Key = 10 x 8 x 59

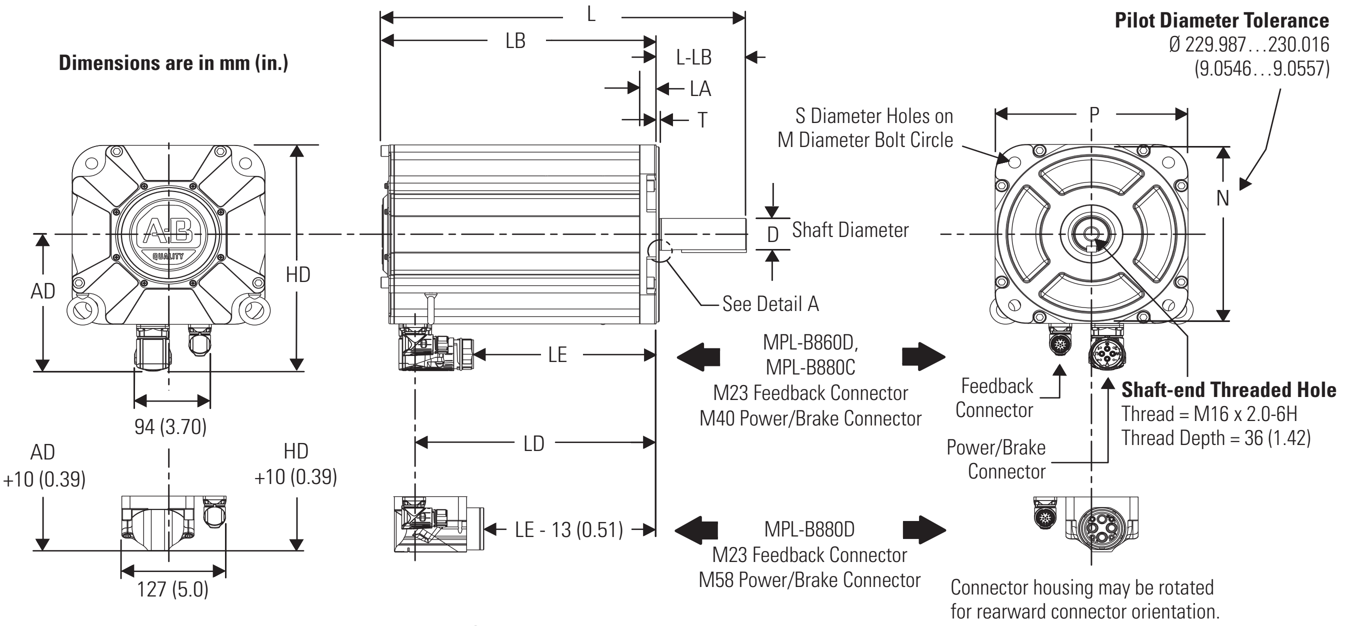
Shaft, Pilot, and Keyway Tolerances	MPL-B6*xx*
Shaft Runout (T.I.R.)	0.05 (0.002)
Pilot Eccentricity (T.I.R.)	0.10 (0.0039)
Max Face Runout (T.I.R.)	0.10 (0.0039)
Keyway Depth (GE)	5.00…5.20 (0.197…0.205)
Keyway Width (F)	9.964…10.000 (0.3923…0.3937)

Motor Series MPL-	AD mm (in.)	HD mm (in.)	T mm (in.)	LA mm (in.)	LD (1) mm (in.)	LE (1) mm (in.)	L (1) mm (in.)	LB (1) mm (in.)	L-LB (2) mm (in.)	D mm (in.)	M mm (in.)	S (3) mm (in.)	N mm (in.)	P mm (in.)	GE mm (in.)	F mm (in.)
B640	154.0 (6.06)	246.5 (9.70)	3.73 (0.147)	17.8 (0.70)	184.0 (7.23)	113.0 (4.43)	304.0 (11.96)	224.0 (8.83)	80.0 (3.15)	38.0 (1.5)	215.0 (8.465)	14.50 (0.579)	180.0 (7.09)	184.9 (7.28)	5.0 (0.197)	10.0 (0.394)
B660					234.0 (9.23)	163.0 (6.43)	355.0 (13.96)	275.0 (10.83)								
B680					285.0 (11.23)	214.0 (8.43)	405.0 (15.96)	325.0 (12.83)								

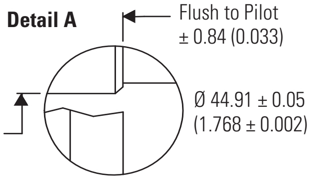
(1) If ordering an MPL-B640, MPL-B660, or MPL-B680 motor with brake, add 89 mm (3.5 in.) to dimensions LD, LE, L, and LB.
(2) Tolerance for this dimension is ±0.7 mm (±0.028 in.).
(3) Tolerance for this dimension is ±0.215 mm (±0.008 in.).

Motors are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

MPL-B8*xx* Motor Dimensions (SpeedTec DIN connectors)



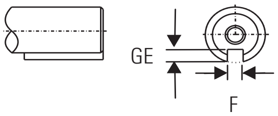
Detail A



Optional Shaft Seal
Refer to page 22
for MP-Series motor
shaft seal kit information.

Shaft Diameter Tolerances
Ø 42.002…42.018 (1.6536…1.6542)

Shaft Detail with Key



Key = 12 x 8 x 79

Shaft, Pilot, and Keyway Tolerances	MPL-B8*xx*
Shaft Runout (T.I.R.)	0.05 (0.0016)
Pilot Eccentricity (T.I.R.)	0.10 (0.0039)
Max Face Runout (T.I.R.)	0.10 (0.0039)
Keyway Depth (GE)	5.00…5.20 (0.197…0.205)
Keyway Width (F)	11.957…12.000 (0.4708…0.4724)

Motor Series	AD mm (in.)	HD mm (in.)	T mm (in.)	LA mm (in.)	LD [1] mm (in.)	LE [1] mm (in.)	L [1] mm (in.)	LB [1] mm (in.)	L-LB [2] mm (in.)	D mm (in.)
MPL-B860	179 (7.05)	297 (11.67)	3.86 (0.152)	20.3 (0.80)	243 (9.55)	171 (6.75)	394 (15.53)	284 (11.20)	110 (4.33)	42.0 (1.654)
MPL-B880					293 (11.55)	222 (8.75)	445 (17.53)	335 (13.20)		

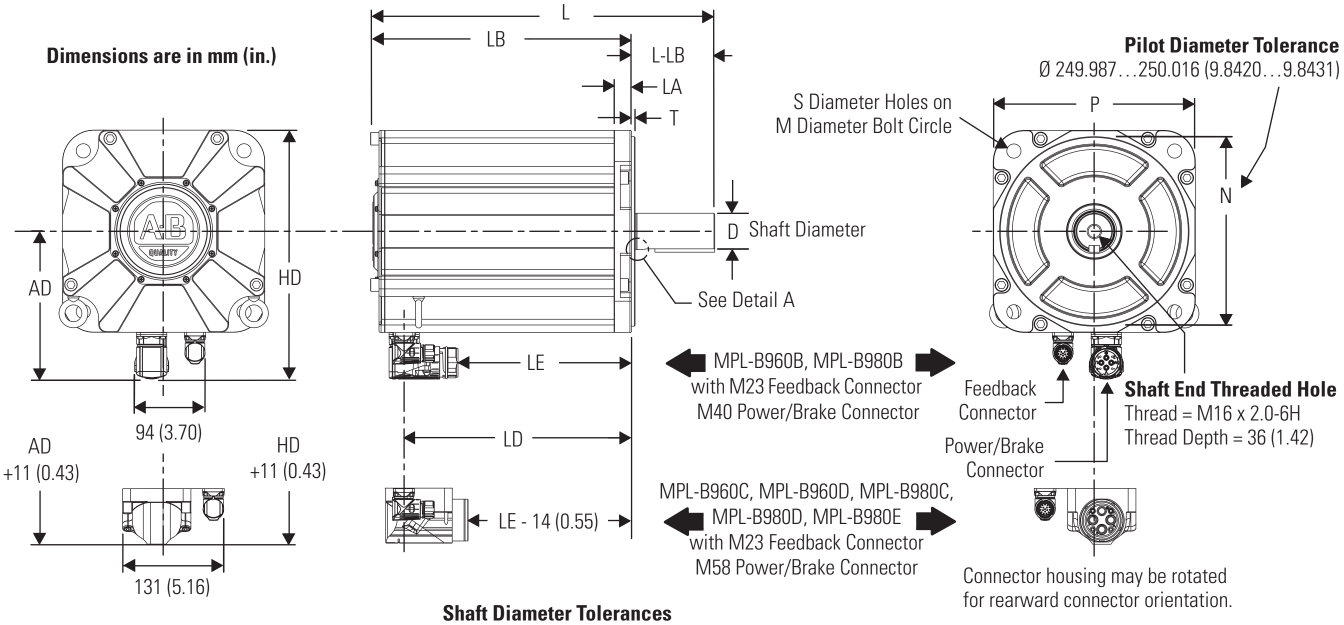
(1) If ordering an MPL-B860 or MPL-B880 motor with brake, add 108 mm (4.24 in.) to dimensions LD, LE, L, and LB.
(2) Tolerance for this dimension is ±0.7 mm (±0.028 in.).

Motor Series	M mm (in.)	S [1] mm (in.)	N mm (in.)	P mm (in.)	GE mm (in.)	F mm (in.)
MPL-B860	265 (10.43)	14.50 (0.579)	230 (9.055)	235 (9.25)	5.0 (0.197)	12.0 (0.4724)
MPL-B880						

(1) Tolerance for this dimension is +0.43 mm (±0.008 in.).

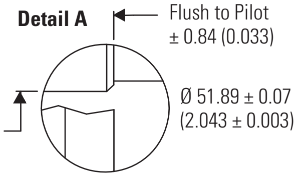
Motors are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

MPL-B9*xx* Motor Dimensions (SpeedTec DIN connectors)



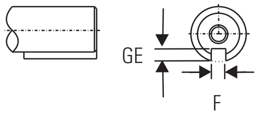
Shaft Diameter Tolerances
Ø 48.002…48.018 (1.8899…1.8905)

Detail A



Optional Shaft Seal
Refer to page 22
for MP-Series motor
shaft seal kit information.

Shaft Detail with Key



Key = 14 x 9 x 79

Shaft, Pilot, and Keyway Tolerances	MPL-B9*xx*
Shaft Runout (T.I.R.)	0.05 (0.002)
Pilot Eccentricity (T.I.R.)	0.125 (0.005)
Max Face Runout (T.I.R.)	0.125 (0.005)
Keyway Depth (GE)	5.50…5.70 (0.217…0.224)
Keyway Width (F)	13.957…14.000 (0.5495…0.5512)

Motor Series	AD mm (in.)	HD mm (in.)	T mm (in.)	LA mm (in.)	LD (1) mm (in.)	LE (1) mm (in.)	L (1) mm (in.)	LB (1) mm (in.)	L-LB (2) mm (in.)	D mm (in.)
MPL-B960	195 (7.68)	328 (12.92)	4.88 (0.192)	22.9 (0.90)	249 (9.80)	178 (7.0)	403 (15.87)	293 (11.55)	110 (4.33)	42.0 (1.6)
MPL-B980					300 (11.80)	229 (9.0)	454 (17.87)	344 (13.55)		

(1) If ordering an MPL-B960 or MPL-B980 motor with brake, add 127 mm (5.0 in.) to dimensions LD, LE, L, and LB.
(2) Tolerance for this dimension is ±0.7 mm (±0.028 in.).

Motor Series	M mm (in.)	S (1) mm (in.)	N mm (in.)	P mm (in.)	GE mm (in.)	F mm (in.)
MPL-B960	300 (11.81)	18.50 (0.738)	250 (9.84)	267 (10.50)	5.50 (0.217)	14.0 (0.5512)
MPL-B980						

(1) Tolerance for this dimension is +0.52 mm (±0.010 in.).

Motors are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

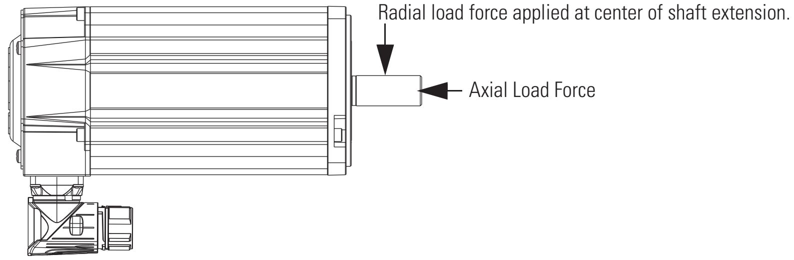
MP-Series Low Inertia Motor Load Force Ratings

Bulletin MPL motors are capable of operating with the maximum radial or maximum axial shaft loads listed in the following tables. Radial loads listed are applied in the middle of the shaft extension. The tables starting below represent an L_{10} bearing fatigue life of 20,000 hours. This 20,000-hour life does not account for possible application-specific life reduction that may occur due to bearing grease contamination from external sources. Maximum operating speed is limited by motor winding.

Radial Load Force Ratings

Motor Cat. No.	**500 rpm** kg (lb)	**1000 rpm** kg (lb)	**2000 rpm** kg (lb)	**3000 rpm** kg (lb)	**3500 rpm** kg (lb)	**4000 rpm** kg (lb)	**5000 rpm** kg (lb)	**6000 rpm** kg (lb)	**7000 rpm** kg (lb)	**8000 rpm** kg (lb)
MPL-A/B1510	–	24 (52)	19 (41)	–	–	15 (33)	–	–	–	12 (26)
MPL-A/B1520	–	25 (56)	20 (45)	–	–	16 (36)	–	–	14 (30)	–
MPL-A/B1530	–	28 (62)	22 (49)	–	–	18 (39)	–	–	15 (33)	–
MPL-A/B210	–	24 (52)	19 (41)	–	–	15 (33)	–	–	–	12 (26)
MPL-A/B220	–	27 (59)	21 (47)	–	18 (39)	–	–	15 (33)	–	–
MPL-A/B230	–	29 (64)	23 (51)	–	19 (42)	–	17 (37)	–	–	–
MPL-A/B310	78 (172)	62 (137)	49 (108)	–	40 (88)	–	36 (79)	–	–	–
MPL-A/B320	87 (192)	69 (152)	55 (121)	–	45 (99)	–	40 (88)	–	–	–
MPL-A/B330	–	74 (163)	59 (130)	–	49 (108)	–	43 (95)	–	–	–
MPL-A/B420	–	78 (172)	62 (137)	–	51 (112)	–	45 (99)	–	–	–
MPL-A/B430	106 (234)	84 (185)	67 (148)	–	55 (121)	–	49 (108)	–	–	–
MPL-A/B4530	133 (293)	105 (232)	84 (185)	73 (161)	–	66 (146)	–	–	–	–
MPL-A4540C	140 ((309)	112 (245)	96 (211)	–	–	–	–	–	–	–
MPL-A/B4540	140 (309)	111 (245)	89 (196)	77 (170)	–	–	–	–	–	–
MPL-A/B4560	–	151 (332)	119 (263)	95 (209)	–	–	–	–	–	–
MPL-A/B520	–	127 (280)	100 (222)	88 (194)	–	80 (176)	–	–	–	–
MPL-A/B540	–	143 (316)	114 (251)	99 (219)	–	90 (199)	–	–	–	–
MPL-A/B560	–	153 (338)	121 (268)	106 (234)	–	–	–	–	–	–
MPL-B580	–	153 (338)	121 (268)	106 (234)	–	–	–	–	–	–
MPL-B640	253 (557)	200 (442)	159 (351)	139 (307)	–	–	–	–	–	–
MPL-B660	275 (607)	219 (482)	173 (382)	151 (334)	–	–	–	–	–	–
MPL-B680	291 (641)	230 (508)	183 (404)	160 (353)	–	–	–	–	–	–
MPL-B860	347 (764)	276 (607)	219 (481)	–	–	–	–	–	–	–
MPL-B880	368 (810)	292 (643)	231 (510)	–	–	–	–	–	–	–
MPL-B960	466 (1028)	370 (816)	323 (713)	–	–	–	–	–	–	–
MPL-B980	494 (1089)	392 (864)	352 (775)	–	–	–	–	–	–	–

MPL-*xxxxx-xxxx*AA Load Forces



Axial Load Force Ratings (maximum radial load)

Motor Cat. No.	500 rpm kg (lb)	1000 rpm kg (lb)	2000 rpm kg (lb)	3000 rpm kg (lb)	3500 rpm kg (lb)	4000 rpm kg (lb)	5000 rpm kg (lb)	6000 rpm kg (lb)	7000 rpm kg (lb)	8000 rpm kg (lb)
MPL-A/B1510	–	15 (33)	10 (22)	–	–	7 (15)	–	–	–	5 (11)
MPL-A/B1520	–	14 (31)	10 (22)	–	–	6 (13)	–	–	4 (9)	–
MPL-A/B1530	–	13 (29)	9 (20)	–	–	6 (13)	–	–	4 (8)	–
MPL-A/B210	–	15 (33)	10 (22)	–	–	7 (15)	–	–	–	5 (11)
MPL-A/B220	–	14 (30)	9 (20)	–	7 (15)	–	–	5 (11)	–	–
MPL-A/B230	–	13 (28)	9 (19)	–	6 (13)	–	5 (11)	–	–	–
MPL-A/B310	30 (66)	23 (51)	16 (35)	–	13 (29)	–	11 (24)	–	–	–
MPL-A/B320	34 (75)	25 (55)	19 (42)	–	15 (33)	–	13 (29)	–	–	–
MPL-A/B330	–	27 (60)	20 (44)	–	16 (35)	–	13 (29)	–	–	–
MPL-A/B420	–	36 (79)	27 (60)	–	21 (46)	–	18 (40)	–	–	–
MPL-A/B430	52 (115)	39 (86)	29 (64)	–	22 (49)	–	19 (42)	–	–	–
MPL-A/B4530	45 (99)	34 (75)	25 (55)	21 (46)	–	19 (42)	–	–	–	–
MPL-A4540C	31 (68)	37 (81)	49 (108)	–	–	–	–	–	–	–
MPL-A/B4540	49 (108)	36 (79)	27 (60)	22 (49)	–	–	–	–	–	–
MPL-A/B4560	–	53 (117)	40 (88)	30 (65)	–	–	–	–	–	–
MPL-A/B520	–	42 (94)	30 (68)	26 (58)	–	22 (50)	–	–	–	–
MPL-A/B540	–	48 (107)	35 (79)	30 (66)	–	26 (58)	–	–	–	–
MPL-A/B560	–	52 (115)	43 (95)	32 (71)	–	–	–	–	–	–
MPL-B580	–	52 (115)	43 (95)	32 (71)	–	–	–	–	–	–
MPL-B640	89 (197)	66 (146)	48 (107)	41 (90)	–	–	–	–	–	–
MPL-B660	98 (217)	72 (159)	53 (118)	45 (99)	–	–	–	–	–	–
MPL-B680	104 (230)	77 (169)	34 (125)	47 (104)	–	–	–	–	–	–
MPL-B860	145 (320)	107 (237)	79 (175)	–	–	–	–	–	–	–
MPL-B880	153 (338)	113 (250)	84 (185)	–	–	–	–	–	–	–
MPL-B960	142 (314)	105 (232)	88 (194)	–	–	–	–	–	–	–
MPL-B980	153 (338)	113 (249)	94 (207)	–	–	–	–	–	–	–

Axial Load Force Ratings (zero radial load)

Motor Cat. No.	**500 rpm** kg (lb)	**1000 rpm** kg (lb)	**2000 rpm** kg (lb)	**3000 rpm** kg (lb)	**3500 rpm** kg (lb)	**4000 rpm** kg (lb)	**5000 rpm** kg (lb)	**6000 rpm** kg (lb)	**7000 rpm** kg (lb)	**8000 rpm** kg (lb)
MPL-A/B1510	–	24 (53)	17 (37)	–	–	12 (26)	–	–	–	8 (18)
MPL-A/B1520	–	24 (53)	17 (37)	–	–	12 (26)	–	–	9 (19)	–
MPL-A/B1530	–	24 (53)	17 (37)	–	–	12 (26)	–	–	9 (19)	–
MPL-A/B210	–	24 (53)	17 (37)	–	–	12 (26)	–	–	–	8 (18)
MPL-A/B220	–	24 (53)	17 (37)	–	13 (28)	–	–	10 (22)	–	–
MPL-A/B230	–	24 (53)	17 (37)	–	13 (28)	–	10 (22)	–	–	–
MPL-A/B310	49 (108)	36 (79)	27 (60)	–	21 (46)	–	18 (40)	–	–	–
MPL-A/B320	49 (108)	36 (79)	27 (60)	–	21 (46)	–	18 (40)	–	–	–
MPL-A/B330	–	36 (79)	27 (60)	–	21 (46)	–	18 (40)	–	–	–
MPL-A/B420	–	51 (112)	38 (84)	–	30 (66)	–	25 (55)	–	–	–
MPL-A/B430	69 (152)	51 (112)	38 (84)	–	30 (66)	–	25 (55)	–	–	–
MPL-A/B4530	69 (152)	51 (112)	38 (84)	31 (68)	–	28 (62)	–	–	–	–
MPL-A4540C	68 (150)	51 (112)	43 (95)	–	–	–	–	–	–	–
MPL-A/B4540	69 (152)	51 (112)	38 (84)	31 (68)	–	–	–	–	–	–
MPL-A/B4560	–	69 (152)	51 (112)	38 (84)	–	–	–	–	–	–
MPL-A/B520	–	67 (149)	49 (109)	41 (92)	–	36 (81)	–	–	–	–
MPL-A/B540	–	67 (149)	49 (109)	41 (92)	–	36 (81)	–	–	–	–
MPL-A/B560	–	67 (149)	49 (109)	41 (92)	–	–	–	–	–	–
MPL-B580	–	67 (149)	49 (109)	41 (92)	–	–	–	–	–	–
MPL-B640	136 (300)	99 (219)	74 (163)	62 (137)	–	–	–	–	–	–
MPL-B660	136 (300)	99 (219)	74 (163)	62 (137)	–	–	–	–	–	–
MPL-B680	136 (300)	99 (219)	74 (163)	62 (137)	–	–	–	–	–	–
MPL-B860	201 (443)	147 (323)	110 (242)	–	–	–	–	–	–	–
MPL-B880	201 (443)	147 (323)	110 (242)	–	–	–	–	–	–	–
MPL-B960	215 (473)	159 (350)	133 (293)	–	–	–	–	–	–	–
MPL-B980	215 (473)	159 (350)	133 (293)	–	–	–	–	–	–	–

MP-Series Low Inertia Motor Catalog Number

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your motor. For questions regarding product availability, contact your Allen-Bradley distributor.

MP L - *x xx xx x* - *x x x x* A A

Factory Options
A = Standard

Mounting Flange
A = IEC metric, free mounting holes (type FF)

Brake
2 = No Brake
4 = 24V DC Brake

Connectors
4 = Circular (threaded) DIN connector, right angle, 180° rotatable (1)
7 = Circular (SpeedTec) DIN connector, right angle, 180° rotatable (2)

Enclosure/Shaft Key/Shaft Seal
J = Shaft key/No shaft seal
K = No shaft key/No shaft seal (3)

Feedback
V = Multi-turn, 128 sin/cos, absolute encoder (Hiperface protocol) (1)
M = Multi-turn, 1024 sin/cos, absolute encoder (Hiperface protocol) (2)
E = Single-turn, 128 sin/cos, absolute encoder (Hiperface protocol) (1)
S = Single-turn, 1024 sin/cos, absolute encoder (Hiperface protocol) (2)
H = 2000 lines/revolution, incremental encoder (4)
R = Resolver (5)

Rated Speed (6)
B = 1000 rpm
C = 1500 rpm
D = 2000 rpm
F = 3000 rpm
H = 3500 rpm
J = 3750 rpm
K = 4000 rpm
P = 5000 rpm
T = 6000 rpm
U = 7000 rpm
V = 8000 rpm

Magnet Stack Length (6)
10 = 25.4 mm (1.0 in.)
20 = 50.8 mm (2.0 in.)
30 = 76.2 mm (3.0 in.)
40 = 101.6 mm (4.0 in.)
50 = 127.0 mm (5.0 in.)
60 = 152.4 mm (6.0 in.)
70 = 177.8 mm (7.0 in.)
80 = 203.2 mm (8.0 in.)

Frame Size
15 = 63 mm
2 = 75 mm
3 = 100 mm
4 = 115 mm
45= 130 mm
5 = 165 mm
6 = 215 mm
8 = 265 mm
9 = 300 mm

Voltage Class
A = 200V
B = 400V

Series Type
L = Low Inertia

Series
MP = Premium permanent magnet rotary servo motor

---

(1) Applies to MPL-A/B15*xx* and MPL-A/B2*xx* motors.

(2) Applies to MPL-A/B3*xx*, MPL-A/B4*xx*, MPL-A/B45*xx*, MPL-A/B5*xx*, MPL-B6*xx*, MPL-B8*xx*, and MPL-B9*xx* motors.

(3) Requires longer lead times. Applies to limited frame sizes only.

(4) Applies to MPL-A/B15*xxx*-H, MPL-A/B2*xxx*-H, MPL-A/B3*xxx*-H, MPL-A/B4*xxx*-H, MPL-A/B45*xxx*-H motors.

(5) Applies to MPL-B3*xxx*-R, MPL-B4*xxx*-R, and MPL-B45*xxx*-R motors.

(6) Not all combinations are available. Only the configurations for rated speed and magnet stack length as listed in MP-Series Low Inertia Motor (230V) Performance Specifications (page 23) and MP-Series Low Inertia Motor (460V) Performance Specifications (page 24) are available.

MP-Series Medium Inertia Motors

The MP-Series (Bulletin MPM) medium-inertia servo motors offer a compact, power dense, feature-rich solution for applications with heavier loads and greater inertia. Leveraging the proven MP-Series motor technology and quality standards, these new servo motors are ideal for users with print, converting, web handling, automotive, and other applications requiring more power in a smaller package.

For drive compatibility, refer to Servo Drives on page 14.

MP-Series Medium Inertia Motor High Resolution Encoder Features

MP-Series medium-inertia motors are available with high performance encoders with a choice of Single-turn (-S) or Multi-turn (-M) high-resolution feedback.

- Up to 2 million counts per revolution (-M and -S) for smooth performance.
- Single-turn encoder provides high-resolution absolute position feedback within one turn.
- Multi-turn encoder provides high-resolution absolute position feedback within 4096 turns. The electromechanical design does not require a battery.

Motor Connector/Cable Compatibility

MP-Series (Bulletin MPM) motors are equipped with SpeedTec-ready DIN connectors.

- MPM-A/B*xxx* motors
- Receives M4 and M7 cable plugs (remove the O-ring for M7)
- Attach M7 cable plug with one-quarter turn

- 2090-CFBM7DF-CEAA*xx* (standard) flying-lead, feedback cables
- 2090-CFBM7DD-CEAA*xx* (standard) drive-end connector, feedback cables
- 2090-CFBM7DF-CEAF*xx* (continuous-flex) flying-lead, feedback cables
- 2090-CFBM7DD-CEAF*xx* (continuous-flex) drive-end connector, feedback cables
- 2090-CFBM7DF-CDAF*xx* (continuous-flex) flying-lead, feedback cables

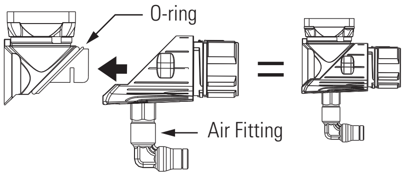
- 2090-CPWM7DF-*xx*AA*xx* (standard) power-only cables
- 2090-CPBM7DF-*xx*AA*xx* (standard) power/brake cables
- 2090-CPWM7DF-*xx*AF*xx* (continuous-flex) power-only cables
- 2090-CPBM7DF-*xx*AF*xx* (continuous-flex) power/brake cables

MP-Series Medium Inertia Motor Options

MP-Series medium-inertia motors are available with these options:

- 24V DC brake.
- Shaft seal kit available for field installation. Shaft seals are made of nitrile. Kits include a lubricant to reduce wear.
- Positive Air Pressure kit (catalog number MPF-7-AIR-PURGE) is mounted on the feedback connector to provide positive air pressure to further reduce the chance of contamination inside the motor.

  Refer to the MP-Series Medium Inertia Servo Motor Installation Instructions, publication MPM-IN001, for more information.



Motor Shaft Seal Kit Combinations and Dimensions

Motor Series	**Shaft Seal Cat. No.**	**Inside Diameter** mm (in.)	**Outside Diameter** mm (in.)	**Width** mm (in.)
MPM-A115*xx* and MPL-B115*xx*	MPL-SSN-A4B4	20 (0.79)	52 (2.05)	7 (0.28)
MPM-A130*xx* and MPL-B130*xx*	MPL-SSN-A5B5	25 (0.98)	62 (2.44)	
MPM-A165*xx* and MPM-B165*xx*	MPL-SSN-F165	30 (1.18)	72 (2.83)	8 (0.31)
MPM-A215*xx* and MPL-B215*xx*	MPL-SSN-A6B6	40 (1.57)	90 (3.54)	

MP-Series Medium Inertia Motor Performance Specifications

MP-Series Medium Inertia Motor (230V) Performance Specifications

Motor MPM-	**Base Speed** rpm	**Max Speed** rpm	**Continuous Stall Torque** Nm (lb-in)	**Peak Stall Torque** Nm (lb-in)	**Motor Rated Output** kW	**Speed at Motor Rated Output** rpm	**Rotor Inertia** [(1)] kg-m^2 (lb-in-s^2)	**Motor Weight,** [(1)] **approx.** kg (lb)
A1151M	4500	6000	2.18 (19.3)	6.60 (58.0)	0.90	5000	0.00065 (0.00575)	3.45 (7.6)
A1152F	3000	5000	4.74 (42.0)	13.5 (119)	1.40	4000	0.00077 (0.00682)	5.20 (11.4)
A1153F	3000	5000	6.55 (58.0)	19.8 (175)	1.45	4000	0.00089 (0.00784)	6.4 (14.0)
A1302F	3000	4500	5.99 (53.0)	13.5 (119)	1.65	4000	0.000983 (0.00870)	6.8 (15.0)
A1304F	3000	4000	9.30 (82.0)	19.3 (171)	2.20	3500	0.001223 (0.01082)	9.6 (21.2)
A1651F	3000	5000	10.7 (95.0)	20.5 (181)	2.50	3000	0.006605 (0.05846)	15.3 (33.8)
A1652F	3000	4000	13.5 (119)	36.0 (319)	4.03	3500	0.007265 (0.06430)	20.6 (45.4)
A1653F	3000	4000	18.6 (165)	42.0 (372)	5.10	3000	0.008025 (0.07103)	25.6 (56.4)
A2152F	3000	4000	27.0 (239)	56.0 (496)	5.20	2000	0.02059 (0.18223)	35.8 (79.0)
A2153F	3000	3600	34.0 (301)	58.0 (513)	5.80	2000	0.02254 (0.19950)	44.6 (98.3)
A2154C	1500	2000	55.0 (487)	106 (938)	6.50	1750	0.02449 (0.21675)	53.6 (118)
A2154E	2250	2650	44.0 (389)	84.0 (743)	7.00	2000		

(1) Refer to MP-Series Medium Inertia Motor Brake Specifications on page 41 for Rotor Inertia and Brake Motor Weight.

System Combinations (230V)

For MP-Series Medium Inertia Motors and	**Refer to**
Kinetix 6000 (230V) drives	page 528
Kinetix 2000 (230V) drives	page 559
Kinetix 300 (240V) drives	page 604
Ultra3000/5000 (230V) drives	page 641

MP-Series Medium Inertia Motor (460V) Performance Specifications

Motor MPM-	**Base Speed** rpm	**Max Speed** rpm	**Continuous Stall Torque** Nm (lb-in)	**Peak Stall Torque** Nm (lb-in)	**Motor Rated Output** kW	**Speed at Motor Rated Output** rpm	**Rotor Inertia** (1) kg-m^2 (lb-in-s^2)	**Motor Weight,** (1) **approx.** kg (lb)
B1151F	3000	5000	2.18 (19.3)	6.6 (58.0)	0.75	4000	0.00065 (0.00575)	3.45 (7.6)
B1151T	6000	7000			0.90	5000		
B1152C	1500	3000	4.74 (42.0)	13.5 (119)	1.20	2500	0.00077 (0.00681)	5.20 (11.4)
B1152F	3000	5200			1.40	4000		
B1152T	6000	7000						
B1153E	2250	3500	6.55 (58.0)	19.8 (175)	1.40	3000	0.00089 (0.00788)	6.40 (14.0)
B1153F	3000	5500			1.45	4000		
B1153T	6000	7000						
B1302F	3000	4500	5.99 (53.0)	13.5 (119)	1.65	4000	0.000983 (0.00870)	6.80 (15.0)
B1302M	4500	6000						
B1302T	6000	7000						
B1304C	1500	2750	10.2 (90.0)	27.1 (240)	2.00	3500	0.001223 (0.01082)	9.60 (21.2)
B1304E	2250	4000			2.20			
B1304M	4500	6000						
B1651C	1500	3500	10.7 (95.0)	23.2 (205)	2.50	3000	0.006605 (0.05846)	15.3 (33.8)
B1651F	3000	5000						
B1651M	4500	5000						
B1652C	1500	2500	16.0 (142)	40.0 (354)	3.80	2500	0.007265 (0.06430)	20.6 (45.4)
B1652E	2250	3500	19.4 (172)	48.0 (425)	4.30	3500		
B1652F	3000	4500						
B1653C	1500	2500	26.8 (237)	67.8 (600)	4.60	2000	0.008025 (0.07103)	25.6 (56.4)
B1653E	2250	3500		62.0 (549)	5.10	3000		
B1653F	3000	4000		56.0 (496)				
B2152C	1500	2500	36.7 (325)	72.3 (640)	5.60	2000	0.02059 (0.18224)	35.8 (79.0)
B2152F	3000	4500	33.0 (292)		5.90	2500		
B2152M	4500	5000	30.0 (266)	50.0 (443)				
B2153B	1300	2000	48.0 (425)	101.1 (895)	6.80	1750	0.02254 (0.19949)	44.6 (98.3)
B2153E	2250	3000			7.20	2000		
B2153F	3000	3800	45.0 (398)	99.0 (876)				
B2154B	1300	2000	62.8 (556)	154.2 (1365)	6.90	1750	0.02449 (0.21675)	53.6 (118.2)
B2154E	2250	3000	56.0 (496)	112.0 (991)	7.50	2000		
B2154F	3000	3300		88.0 (779)				

(1) Refer to MP-Series Medium Inertia Motor Brake Specifications on page 41 for Rotor Inertia and Brake Motor Weight.

System Combinations (460V)

For MP-Series Medium Inertia Motors and	Refer to
Kinetix 6000 and Kinetix 6200/6500 (460V) drives	page 530
Kinetix 7000 drives	page 586
Kinetix 300 (480V) drives	page 605
Ultra3000/5000 (460V) drives	page 644

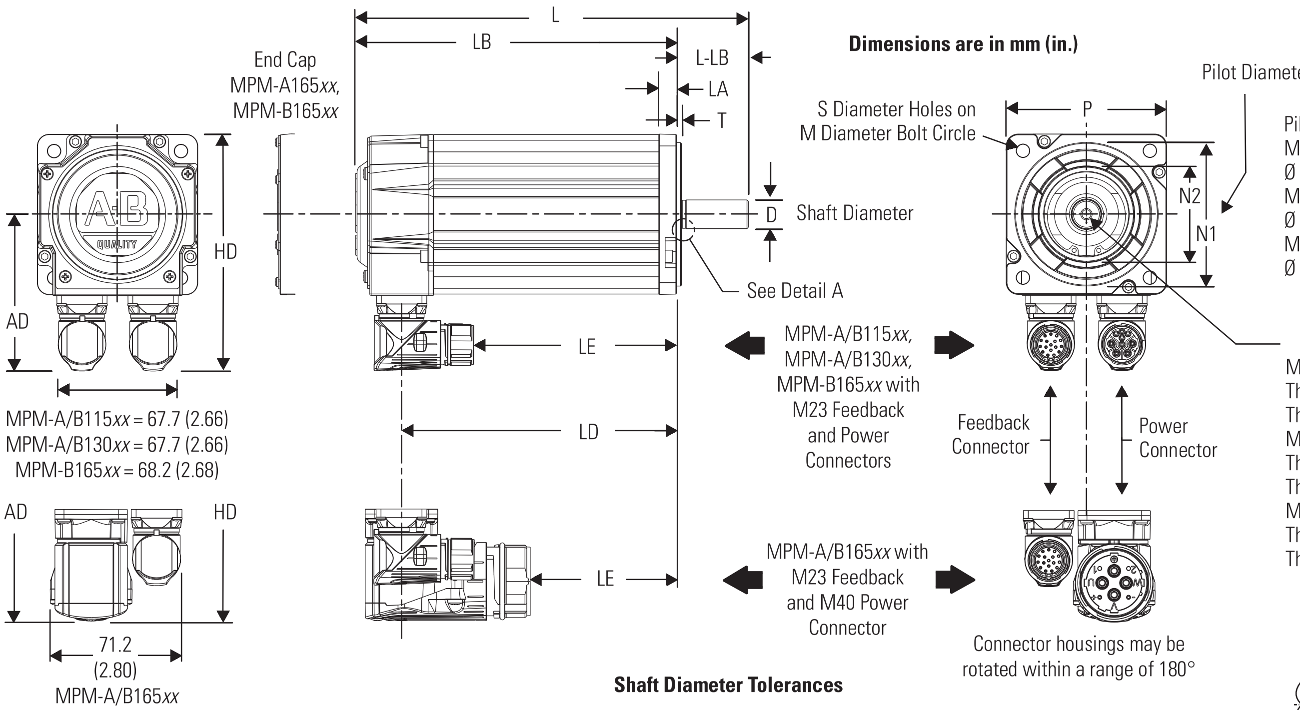
MP-Series Medium Inertia Motor Brake Specifications

Motor MPM-	**Max Backlash (brake engaged)** arc minutes	**Holding Torque** Nm (lb-in)	**Coil Current at 24V DC** A	**Brake Response Time**			**Brake Rotor Inertia** kg-m^2 (lb-in-s^2)	**Brake Motor Weight, approx.** kg (lb)
				Release ms	**Engage (using external arc suppression device)**			
					MOV ms	**Diode** ms		
A/B1151	45	4.18 (37)	0.45…0.55	50	20	110	0.00065 (0.00575)	5.2 (11.4)
A/B1152							0.00077 (0.00681)	6.9 (15.2)
A/B1153							0.00089 (0.00788)	8.1 (17.8)
A/B1302	48	10.2 (90)	0.576…0.704	110	25	160	0.000983 (0.00870)	8.6 (19.0)
A/B1304							0.001223 (0.01082)	11.7 (25.7)
A/B1651	25	28.3 (250)	1.05…1.28	70	50	250	0.006605 (0.05846)	17.9 (39.5)
A/B1652							0.007265 (0.06430)	23.2 (51.1)
A/B1653							0.008025 (0.07103)	28.2 (62.1)
A/B2152	25	70 (619)	1.84…2.25	200	120	900	0.02059 (0.18224)	43.8 (96.5)
A/B2153							0.02254 (0.19949)	53.6 (115.8)
A/B2154							0.02449 (0.21675)	61.6 (135.7)

MP-Series Medium Inertia Motor Dimensions

MPM-A/B115*xx*, MPM-A/B130*xx*, MPM-A/B165*xx* Motor Dimensions

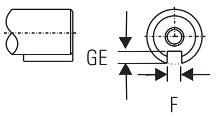
Dimensions are in mm (in.)



Pilot Diameter Tolerances
MPM-A/B115*xx* Motors:
Ø 94.991…95.013 (3.7398…3.7407)
MPM-A/B130*xx* Motors:
Ø 109.991…110.013 (4.3303…4.3312)
MPM-A/B165*xx* Motors:
Ø 129.991…130.014 (5.1178…5.1187)

Shaft End Threaded Hole
MPM-A/B115*xx* Motors:
Thread - M6 x 1.0-6H
Thread Depth - 16 (0.63)
MPM-A/B130*xx* Motors:
Thread - M8 x 1.25-6H
Thread Depth - 19 (0.75)
MPM-A/B165*xx* Motors:
Thread - M10 x 1.5-6H
Thread Depth - 22 (0.87)

Shaft Detail with Key

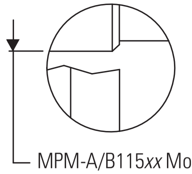


MPM-A/B115*xx* = 6 x 6 x 25 Key
MPM-A/B130*xx* = 8 x 7 x 32 Key
MPM-A/B165*xx* = 8 x 7 x 40 Key

Shaft Diameter Tolerances
MPM-A/B115*xx* Motors:
Ø 18,996…19,009 (0.7479…0.7483)
MPM-A/B130*xx* Motors:
Ø 23,996…24,009 (0.9448…0.9451)
MPM-A/B165*xx* Motors:
Ø 27,996…28,009 (1.1022…1.1027)

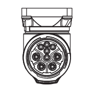
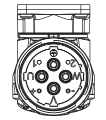
Optional Shaft Seal
Refer to page 38
for MP-Series motor
shaft seal kit information.

Detail A



MPM-A/B115*xx* Motors:
Flush to 2.87 (0.113) Pilot ± 0,83 (0.032)
MPM-A/B130*xx*, MPM-A/B165*xx* Motors:
Flush to 3.38 (0.133) Pilot ± 0,83 (0.032)

MPM-A/B115*xx* Motors:
Ø 19.94 ±0.05 (0.785 ±0.002)
MPM-A/B130*xx* Motors:
Ø 24.94 ±0.05 (0.982 ±0.002)
MPM-A/B165*xx* Motors:
Ø 29.92 ±0.05 (1.178 ±0.002)

Power Connectors on MPM-A/B165*xx* Motors		MPM-A165*xx*	MPM-B165*xx*
M23 Power Connector		N/A	MPM-B1651F, MPM-B1651C, MPM-B1652C, MPM-B1653C
M40 Power Connector		MPM-A165*xx*	MPM-B1651M, MPM-B1652E, MPM-B1652F, MPM-B1653E, MPM-B1653F

Shaft, Pilot, and Keyway Tolerances	MPM-A/B115*xx*	MPM-A/B130*xx*	MPM-A/B165*xx*
Shaft Runout (T.I.R.)	0.04 (0.0016)	0.04 (0.0016)	0.04 (0.0016)
Pilot Eccentricity (T.I.R.)	0.08 (0.0031)	0.10 (0.0039)	0.10 (0.0039)
Max Face Runout (T.I.R.)	0.08 (0.0031)	0.10 (0.0039)	0.10 (0.0039)
Keyway Depth (GE)	3.50…3.60 (0.138…0.142)	4.00…4.20 (0.158…0.165)	4.00…4.20 (0.158…0.165)
Keyway Width (F)	5.97…6.00 (0.235…0.236)	7.96…8.00 (0.314…0.315)	7.964…8.000 (0.3135…0.3150)

MPM-A/B115*x*, MPM-A/B130*x*, MPM-A/B165*x* Motor Dimensions

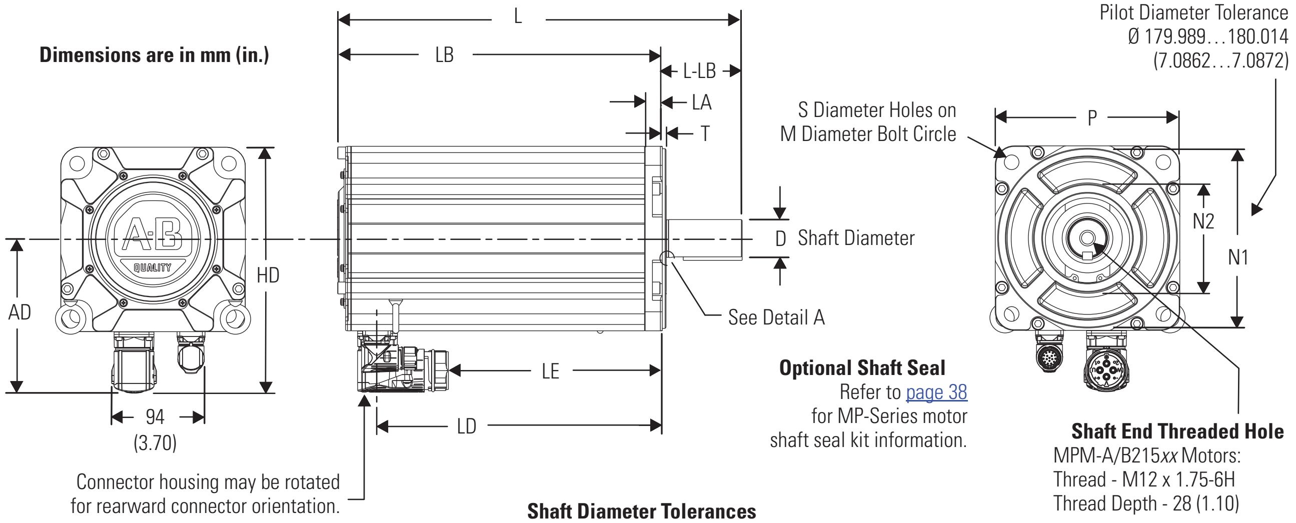
Motor Series MPM-	**AD** (1) mm (in.)	**HD** (1) mm (in.)	**T** mm (in.)	**LA** mm (in.)	**LD** (2) (3) mm (in.)	**LE** (2) (4) mm (in.)	**L** (2) mm (in.)	**LB** (2) mm (in.)	**L-LB** (5) mm (in.)	**D** mm (in.)	**M** mm (in.)	**S** (6) mm (in.)	**N1** mm (in.)	**N2** mm (in.)	**P** mm (in.)	**GE** mm (in.)	**F** mm (in.)
A/B1151	90.9 (3.58)	140.1 (5.52)	2.74 (0.108)	10.16 (0.40)	124 (4.89)	84.1 (3.31)	190 (7.48)	150 (5.90)	40.0 (1.58)	19.0 (0.748)	115.0 (4.528)	10.0 (0.401)	95.0 (3.74)	59.0 (2.32)	98.3 (3.87)	3.5 (0.138)	6.0 (0.236)
A/B1152					150 (5.89)	110 (4.31)	215 (8.48)	175 (6.90)									
A/B1153					175 (6.89)	135 (5.31)	241 (9.48)	201 (7.90)									
A/B1302	98.6 (3.88)	155.4 (6.12)	2.74 (0.108)	12.19 (0.48)	153 (6.02)	113 (4.44)	229 (9.0)	179 (7.03)	50.0 (1.97)	24.0 (0.945)	130.0 (5.118)	10.0 (0.401)	110.0 (4.331)	70.3 (2.77)	113.7 (4.48)	4.0 (0.158)	8.0 (0.315)
A/B1304					204 (8.02)	164 (6.44)	279 (11.0)	229 (9.03)									
A/B1651	113.4 (4.47)	185.2 (7.29)	3.12 (0.123)	14.0 (0.55)	200 (7.88)	160 (6.30)	287 (11.28)	227 (8.92)	60.0 (2.36)	28.0 (1.102)	165.0 (6.496)	12.0 (0.481)	130.0 (5.118)	81.0 (3.19)	143.5 (5.65)	4.0 (0.158)	8.0 (0.315)
A/B1652					251 (9.88)	211 (8.30)	337 (13.28)	277 (10.92)									
A/B1653					302 (11.88)	262 (10.30)	388 (15.28)	328 (12.92)									

(1) This dimension applies to MPM-B165*x* motors with M23 connectors. For MPM-A/B165*x* motors with M40 connectors, add 23 mm (0.91 in.).
(2) If ordering an MPM-A/B115*x* or MPM-A/B130*x* motor with brake, add 48.5 mm (1.91 in.) to dimensions L, LB, LE, and LD.
If ordering an MPM-A/B165*x*, motor with brake, add 51.5 mm (2.03 in.) to dimensions L, LB, LE, and LD.
(3) This dimension applies to MPM-B165*x* motors with M23 connectors. For MPM-A/B165*x* motors with M40 connectors, add 2.0 mm (0.07 in.).
(4) This dimension applies to MPM-B165*x* motors with M23 connectors. For MPM-A/B165*x* motors with M40 connectors, subtract 29.0 mm (1.15 in.).
(5) The tolerance for this dimension is ±0.7 mm (±0.028 in.).
(6) For MPM-A/B115*x* and MPM-A/B130*x* motors, the tolerance for this dimension is +0.36 mm (±0.007 in.).
For MPM-A/B165*x* motors, the tolerance is +0.43 mm (±0.008 in.).

Motors are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

MPM-A/B215*x* Motor Dimensions

Dimensions are in mm (in.)



Pilot Diameter Tolerance
Ø 179.989…180.014
(7.0862…7.0872)

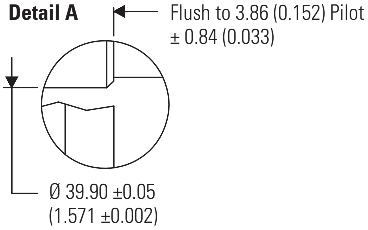
Connector housing may be rotated for rearward connector orientation.

Optional Shaft Seal
Refer to page 38 for MP-Series motor shaft seal kit information.

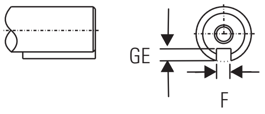
Shaft End Threaded Hole
MPM-A/B215*xx* Motors:
Thread - M12 x 1.75-6H
Thread Depth - 28 (1.10)

Shaft Diameter Tolerances
Ø 38.002…38.018 (1.4961…1.4968)

Detail A



Shaft Detail with Key



Key = 10 x 8 x 59

Shaft, Pilot, and Keyway Tolerances	**MPM-A/B215*x***
Shaft Runout (T.I.R.)	0.05 (0.002)
Pilot Eccentricity (T.I.R.)	0.10 (0.0039)
Max Face Runout (T.I.R.)	0.10 (0.0039)
Keyway Depth (GE)	5.00…5.20 (0.197…0.205)
Keyway Width (F)	9.964…10.000 (0.3923…0.3937)

Motor Series MPM-	**AD** mm (in.)	**HD** mm (in.)	**T** mm (in.)	**LA** mm (in.)	**LD** (1) mm (in.)	**LE** (1) mm (in.)	**L** (1) mm (in.)	**LB** (1) mm (in.)	**L-LB** (2) mm (in.)	**D** mm (in.)	**M** mm (in.)	**S** (3) mm (in.)	**N1** mm (in.)	**N2** mm (in.)	**P** mm (in.)	**GE** mm (in.)	**F** mm (in.)
A/B2152	154 (6.06)	247 (9.70)	3.73 (0.147)	17.8 (0.70)	234 (9.23)	163 (6.43)	355 (13.96)	275 (10.81)	80.0 (3.150)	38.0 (1.50)	215 (8.465)	14.50 (0.579)	180 (7.09)	108 (4.25)	185 (7.28)	5.0 (0.197)	10.0 (0.394)
A/B2153					285 (11.23)	214 (8.43)	405 (15.96)	325 (12.81)									
A/B2154					336 (13.23)	265 (10.43)	456 (17.96)	376 (14.81)									

(1) If ordering an MPM-A/B215*x* motor with brake, add 88.9 mm (3.5 in.) to dimensions LD, LE, L, and LB.
(2) Tolerance for this dimension is ±0.7 mm (±0.028 in.).
(3) Tolerance for this dimension is +0.43 mm (±0.008 in.).

Motors are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

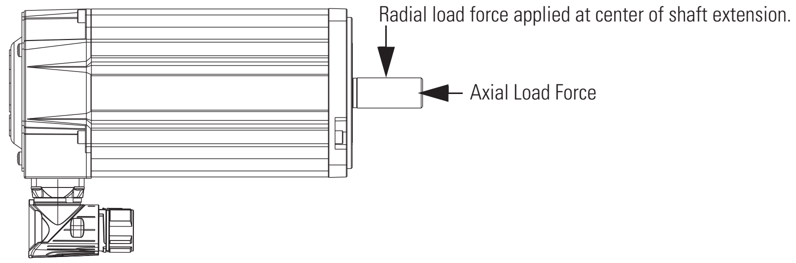
MP-Series Medium Inertia Motor Load Force Ratings

Bulletin MPM motors are capable of operating with the maximum radial or maximum axial shaft loads listed in the following tables. Radial loads listed are applied in the middle of the shaft extension. The tables starting below represent an L_{10} bearing fatigue life of 20,000 hours. This 20,000-hour life does not account for possible application-specific life reduction that may occur due to bearing grease contamination from external sources. Maximum operating speed is limited by motor winding.

Radial Load Force Ratings

Motor Cat. No.	**1000 rpm** kg (lb)	**2000 rpm** kg (lb)	**3000 rpm** kg (lb)	**5000 rpm** kg (lb)	**7000 rpm** kg (lb)
MPM-A/B1151	77 (170)	61 (134)	54 (119)	45 (99)	40 (88)
MPM-A/B1152	84 (185)	66 (145)	58 (128)	49 (108)	43 (95)
MPM-A/B1153	88 (194)	70 (154)	61 (134)	51 (112)	46 (101)
MPM-A/B1302	105 (231)	83 (183)	72 (159)	61 (134)	54 (119)
MPM-A/B1304	115 (253)	91 (200)	80 (176)	67 (148)	–
MPM-A/B1651	141 (311)	112 (247)	97 (214)	82 (181)	–
MPM-A/B1652	151 (333)	119 (262)	104 (229)	–	–
MPM-A/B1653	156 (344)	123 (271)	107 (236)	–	–
MPM-A/B2152	216 (476)	171 (377)	149 (328)	–	–
MPM-A/B2153	228 (502)	180 (396)	156 (344)	–	–
MPM-A/B2154	235 (518)	185 (407)	161 (355)	–	–

MPM-*xxxxx*-*xxxx*AA Load Forces



Axial Load Force Ratings (maximum radial load)

Motor Cat. No.	**1000 rpm** kg (lb)	**2000 rpm** kg (lb)	**3000 rpm** kg (lb)	**5000 rpm** kg (lb)	**7000 rpm** kg (lb)
MPM-A/B1151	29 (64)	22 (48)	18 (40)	14 (31)	12 (26)
MPM-A/B1152	31 (68)	23 (51)	19 (42)	15 (33)	13 (29)
MPM-A/B1153	33 (73)	24 (53)	20 (44)	16 (35)	14 (31)
MPM-A/B1302	26 (57)	19 (42)	16 (35)	13 (29)	11 (24)
MPM-A/B1304	30 (66)	22 (48)	18 (40)	15 (33)	–
MPM-A/B1651	37 (81)	28 (62)	23 (51)	18 (40)	–
MPM-A/B1652	41 (90)	30 (66)	25 (55)	–	–
MPM-A/B1653	43 (95)	32 (70)	27 (59)	–	–
MPM-A/B2152	55 (121)	40 (88)	34 (75)	–	–
MPM-A/B2153	60 (132)	44 (97)	36 (79)	–	–
MPM-A/B2154	63 (139)	46 (101)	38 (84)	–	–

Axial Load Force Ratings (zero radial load)

Motor Cat. No.	**1000 rpm** kg (lb)	**2000 rpm** kg (lb)	**3000 rpm** kg (lb)	**5000 rpm** kg (lb)	**7000 rpm** kg (lb)
MPM-A/B1151	46 (101)	34 (75)	28 (62)	23 (51)	19 (42)
MPM-A/B1152	46 (101)	34 (75)	28 (62)	23 (51)	19 (42)
MPM-A/B1153	46 (101)	34 (75)	28 (62)	23 (51)	19 (42)
MPM-A/B1302	46 (101)	34 (75)	28 (62)	23 (51)	19 (42)
MPM-A/B1304	46 (101)	34 (75)	28 (62)	23 (51)	–
MPM-A/B1651	61 (134)	44 (97)	38 (84)	30 (66)	–
MPM-A/B1652	61 (134)	44 (97)	38 (84)	–	–
MPM-A/B1653	61 (134)	44 (97)	38 (84)	–	–
MPM-A/B2152	90 (198)	65 (143)	54 (119)	–	–
MPM-A/B2153	90 (198)	65 (143)	54 (119)	–	–
MPM-A/B2154	90 (198)	65 (143)	54 (119)	–	–

MP-Series Medium Inertia Motor Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your motor. For questions regarding product availability, contact your Allen-Bradley distributor.

MP M - *x xxx x x* -*x* J 7 *x* A A

Factory Options
A = Standard

Mounting Flange
A = IEC metric, free mounting holes (type FF)

Brake
2 = No brake
4 = 24V DC brake

Connectors
7 = Circular (SpeedTec) DIN connector, right angle, 180° rotatable

Enclosure/Shaft Key/Shaft Seal
J = Shaft key/No shaft seal

Feedback
2 = Resolver, 4 pole (1)
M = Multi-turn, 1024 sin/cos, absolute encoder (Hiperface protocol)
S = Single-turn, 1024 sin/cos, absolute encoder (Hiperface protocol)

Base Speed (2)
B = 2300 rpm
C = 1500 rpm
E = 2250 rpm
F = 3000 rpm
M = 4500 rpm
T = 6000 rpm

Magnet Stacks (2)
1 = 1 stack
2 = 2 stacks
3 = 3 stacks
4 = 4 stacks

Frame Size - Bolt Circle Diameter (BCD)
115 = 115 mm
130 = 130 mm
165 = 165 mm
215 = 215 mm

Voltage Class
A = 200V
B = 400V

Series Type
M = Medium Inertia

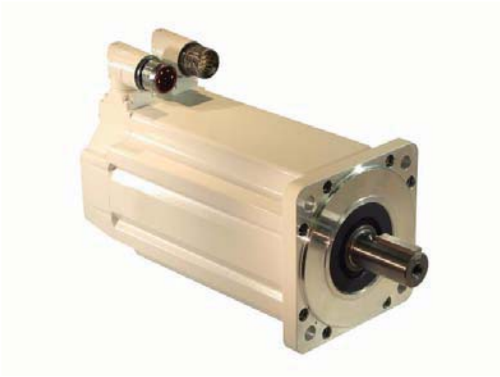
Series
MP = Premium permanent magnet rotary servo motor

(1) Resolver feedback is not available on all models.

(2) Not all combinations are available. Only the configurations for rated speed and magnet stacks as listed in MP-Series Medium Inertia Motor (230V) Performance Specifications (page 39) and MP-Series Medium Inertia Motor (460V) Performance Specifications (page 40) are available.

Notes:

MP-Series Food Grade Motors



MP-Series (Bulletin MPF) food-grade motors combine the characteristics of the MP-Series low-inertia servo motors with features specifically designed to meet the unique needs of many food and beverage applications. These high performance servo motors address the challenges of food environments by incorporating improved sealing techniques and noncorrosive food-grade fasteners and coatings. For meat and poultry applications, and for applications with high pressure wash and caustic chemicals, the MP-Series stainless-steel servo motors are recommended.

For drive compatibility, refer to Servo Drives on page 14.

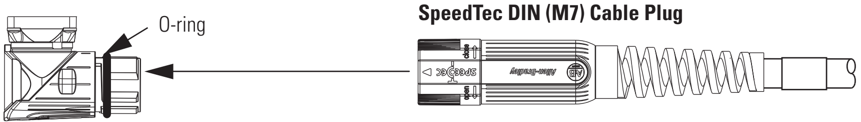
MP-Series Food Grade Motor High Resolution Encoder Features

MP-Series Food Grade motors are available with high performance encoders with a choice of Single-turn (-S) or Multi-turn (-M) high resolution feedback.

- Up to 2 million counts per revolution for smooth performance
- Single-turn encoder provides high-resolution absolute position feedback within one turn.
- Multi-turn encoder provides high-resolution absolute position feedback within 4096 turns.

Motor Connector/Cable Compatibility

MP-Series (Bulletin MPF) motors are equipped with SpeedTec-ready DIN connectors.



SpeedTec-ready DIN Connectors

- MPF-A/B*xxx* motors
- Receives M4 and M7 cable plugs (remove the O-ring for M7)
- Attach M7 cable plug with one-quarter turn

SpeedTec DIN (M7) Cable Plug

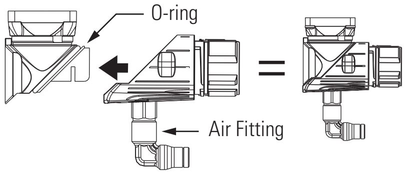
- 2090-CFBM7DF-CEAA*xx* (standard) flying-lead, feedback cables
- 2090-CFBM7DD-CEAA*xx* (standard) drive-end connector, feedback cables
- 2090-CFBM7DF-CEAF*xx* (continuous-flex) flying-lead, feedback cables
- 2090-CFBM7DD-CEAF*xx* (continuous-flex) drive-end connector, feedback cables
- 2090-CFBM7DF-CDAF*xx* (continuous-flex) flying-lead, feedback cables

- 2090-CPWM7DF-*xx*AA*xx* (standard) power-only cables
- 2090-CPBM7DF-*xx*AA*xx* (standard) power/brake cables
- 2090-CPWM7DF-*xx*AF*xx* (continuous-flex) power-only cables
- 2090-CPBM7DF-*xx*AF*xx* (continuous-flex) power/brake cables

MP-Series Food Grade Motor Options

MP-Series Food Grade motors are available with these options:

- 24V DC brake
- Shaft seal kit is available for field replacement. Shaft seals are made of PTFE. Kits include a lubricant to reduce wear.
- Positive Air Pressure kit (catalog number MPF-7-AIR-PURGE) is mounted on the feedback connector to provide positive air pressure to further reduce the chance of contamination inside the motor.
- Refer to the MP-Series Food Grade Servo Motor Installation Instructions, publication MP-IN004, for more information.



Motor Shaft Seal Kit Combinations and Dimensions

Motor Series	Shaft Seal Cat. No.	Inside Diameter mm (in.)	Outside Diameter mm (in.)	Width mm (in.)
MPF-A3*xx* and MPF-B3*xx*	MPF-SST-A3B3	23 (0.90)	47 (1.85)	6 (0.24)
MPF-A4*xx* and MPF-B4*xx*	MPF-SST-A4B4	20 (0.79)	52 (2.05)	7 (0.28)
MPF-A45*xx* and MPF-B45*xx*	MPF-SST-A45B45	31 (1.22)	62 (2.44)	7 (0.28)
MPF-A5*xx* and MPF-B5*xx*	MPF-SST-F165	30 (1.18)	72 (2.84)	8 (0.32)

MP-Series Food Grade Motor Performance Specifications

MP-Series Food Grade (230V) Performance Specifications

Motor MPF-	Max Speed rpm	Continuous Stall Torque Nm (lb-in)	Peak Stall Torque Nm (lb-in)	Motor Rated Output kW	Speed at Motor Rated Output rpm	Rotor Inertia (1) kg-m^2 (lb-in-s^2)	Motor Weight, approx. (1) kg (lb)
A310P	5000	1.58 (14)	3.61 (32)	0.73	4750	0.000044 (0.00039)	2.8 (6.1)
A320P	5000	3.05 (27)	7.91 (70)	1.3	4750	0.000078 (0.00069)	3.8 (8.3)
A320H	3500	3.05 (27)	7.91 (70)	1.0	3350		
A330P	5000	4.18 (37)	11.1 (98)	1.6	5000	0.00012 (0.0010)	4.7 (10)
A430P	5000	5.99 (53)	19.8 (175)	1.9	5000	0.00038 (0.0033)	5.6 (12)
A430H	3500	6.21 (55)	19.8 (175)	1.8	3500		
A4530K	4000	8.13 (72)	20.3 (180)	2.3	4000	0.00040 (0.0036)	7.4 (16)
A4540F	3000	10.2 (90)	27.1 (240)	2.5	3000	0.00052 (0.0046)	8.7 (19)
A540K	4000	19.4 (172)	48.6 (430)	4.1	4000	0.00147 (0.013)	16 (35)

(1) Refer to MP-Series Low Inertia Motor (230V) Performance Specifications on page 23 for Brake Rotor Inertia and Brake Motor Weight.

System Combinations (230V)

For MP-Series Food Grade Motors and	Refer to
Kinetix 6000 (230V) drives	page 539
Kinetix 2000 drives	page 565
Kinetix 300 (240V) drives	page 607
Ultra3000/5000 (230V) drives	page 652

MP-Series Food Grade Motor (460V) Performance Specifications

Motor MPF-	**Max Speed** rpm	**Continuous Stall Torque** Nm (lb-in)	**Peak Stall Torque** Nm (lb-in)	**Motor Rated Output** kW	**Speed at Motor Rated Output** rpm	**Rotor Inertia** (1) kg-m^2 (lb-in-s^2)	**Motor Weight, approx.** (1) kg (lb)
B310P	5000	1.58 (14)	3.61 (32)	0.77	5000	0.000044 (0.00039)	2.8 (6.1)
B320P	5000	3.05 (27)	7.91 (70)	1.5	5000	0.000078 (0.00069)	3.8 (8.3)
B330P	5000	4.18 (37)	11.1 (98)	1.6	5000	0.00012 (0.0010)	4.7 (10)
B430P	5000	6.55 (58)	19.8 (175)	2.0	5000	0.00038 (0.0033)	5.6 (12)
B4530K	4000	8.25 (73)	20.3 (180)	2.4	4000	0.00040 (0.0036)	7.4 (16)
B4540F	3000	10.2 (90)	27.1 (240)	2.5	3000	0.00052 (0.0046)	8.7 (19)
B540K	4000	19.4 (172)	48.6 (430)	4.1	4000	0.00147 (0.013)	16 (35)

(1) Refer to MP-Series Low Inertia Motor (230V) Performance Specifications on page 23 for Brake Rotor Inertia and Brake Motor Weight.

System Combinations (460V)

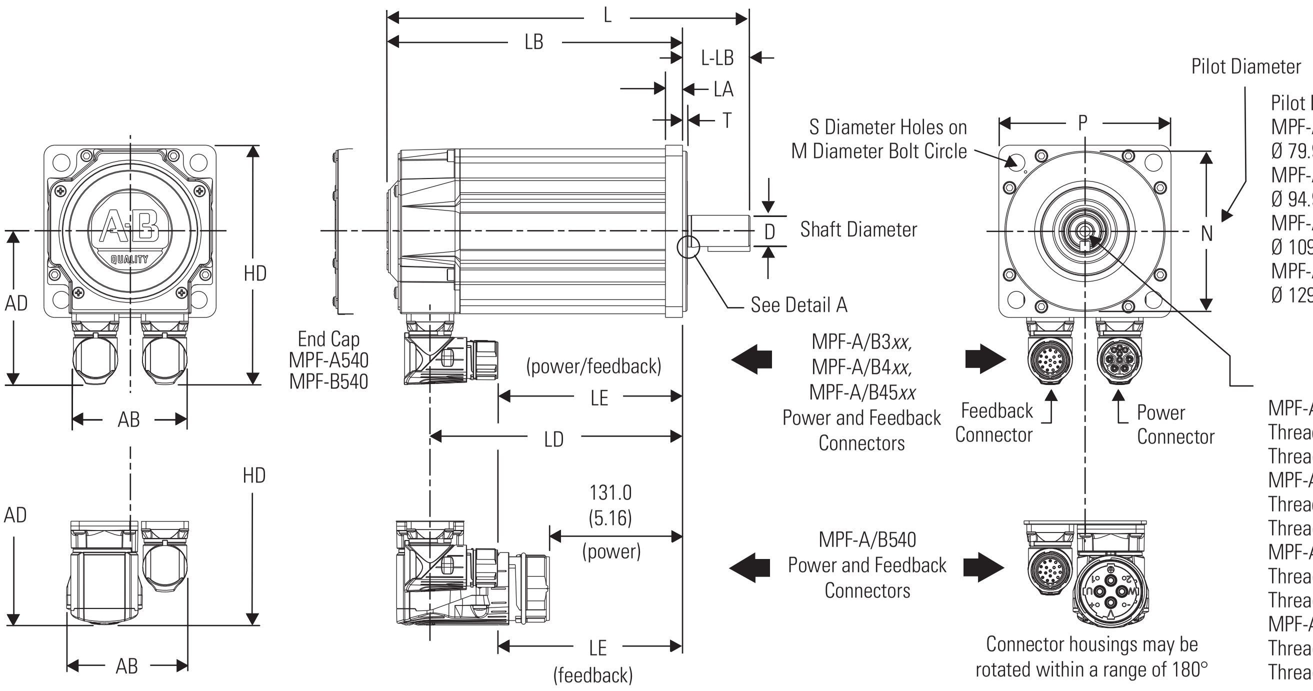
For MP-Series Food Grade Motors and	**Refer to**
Kinetix 6000 and Kinetix 6200/6500 (460V) drives	page 543
Kinetix 300 (480V) drives	page 609
Ultra3000/5000 (460V) drives	page 655

MP-Series Food Grade Motor Brake Specifications

Motor MPF-	**Max Backlash (brake engaged)** arc minutes	**Holding Torque** Nm (lb-in)	**Coil Current at 24V DC** A	**Brake Response Time**			**Brake Rotor Inertia** kg-m^2 (lb-in-s^2)	**Brake Motor Weight, approx.** kg (lb)
				Release ms	**Engage (using external arc suppression device)**			
					MOV ms	**Diode** ms		
A/B310	45	4.18 (37)	0.45…0.55	50	20	110	0.000057 (0.00050)	3.8 (8.3)
A/B320							0.000092 (0.00081)	4.7 (10)
A/B330							0.00013 (0.0011)	5.7 (13)
A/B430	37	10.2 (90)	0.576…0.704	110	25	160	0.00042 (0.0038)	7.4 (16)
A/B4530							0.00044 (0.0039)	9.2 (20)
A/B4540							0.00056 (0.0050)	11 (24)
A/B540	25	28.3 (250)	1.05…1.28	70	50	250	0.00157 (0.0139)	19 (41)

MP-Series Food Grade Motor Dimensions

MPF-A/B3*xx*, MPF-A/B4*xx*, MPF-A/B45*xx*, MPF-A/B5*xx* Motor Dimensions

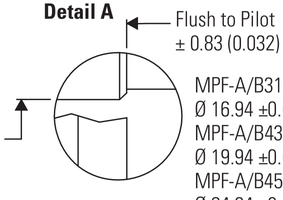


Connector housings may be rotated within a range of 180°

Pilot Diameter Tolerances
MPF-A/B310, -A/B320, -A/B330 Motors:
Ø 79.993…80.012 (3.1493…3.1501)
MPF-A/B430 Motors:
Ø 94.991…95.013 (3.7398…3.7407)
MPF-A/B4530, -A/B4540 Motors:
Ø 109.991…110.013 (4.3303…4.3312)
MPF-A/B540 Motors:
Ø 129.989…130.014 (5.1177…5.1187)

Shaft End Threaded Hole
MPF-A/B310, -A/B320, -A/B330 Motors:
Thread - M5 x 0.8-6H
Thread Depth - 12.5 (0.49)
MPF-A/B430 Motors:
Thread - M6 x 1.0-6H
Thread Depth - 16 (0.63)
MPF-A/B4530, -A/B4540 Motors:
Thread - M8 x 1.25-6H
Thread Depth - 19 (0.75)
MPF-A/B540 Motors:
Thread - M10 x 1.5-6H
Thread Depth - 22 (0.87)

Detail A



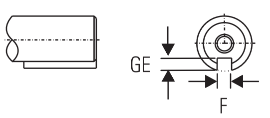
MPF-A/B310, -A/B320, -A/B330 Motors:
Ø 16.94 ±0.05 (0.667 ±0.002)
MPF-A/B430 Motors:
Ø 19.94 ±0.05 (0.785 ±0.002)
MPF-A/B4530, -A/B4540 Motors:
Ø 24.94 ±0.05 (0.982 ±0.002)
MPF-A/B540 Motors:
Ø 28.0 ±0.05 (1.102 ±0.002)

Shaft Diameter Tolerances

MPF-A/B310, -A/B320, -A/B330 Motors:
Ø 15.989…16.013 (0.6295…0.6305)
MPF-A/B430 Motors:
Ø 18.996…19.009 (0.7479…0.7483)
MPF-A/B4530, -A/B4540 Motors:
Ø 23.996…24.009 (0.9448…0.9451)
MPF-A/B540 Motors:
Ø 27.996…28.009 (1.1022…1.1027)

Optional Shaft Seal
Refer to page 50 for MP-Series Food Grade motors shaft seal kit information.

Shaft Detail with Key



MPF-A/B310, -A/B320, -A/B330 = 5 x 5 x 25 Key
MPF-A/B430 = 6 x 6 x 25 Key
MPF-A/B4530, -A/B4540 = 8 x 7 x 32 Key
MPF-A/B540 = 8 x 7 x 40 Key

Shaft, Pilot, and Keyway Tolerances	MPF-A/B3*xx*	MPF-A/B4*xx*	MPF-A/B45*xx*	MPF-A/B540
Shaft Runout (T.I.R.)	0.035 (0.0014)	0.04 (0.0016)	0.04 (0.0016)	0.04 (0.0016)
Pilot Eccentricity (T.I.R.)	0.08 (0.0031)	0.08 (0.0031)	0.10 (0.0039)	0.10 (0.0039)
Max Face Runout (T.I.R.)	0.08 (0.0031)	0.08 (0.0031)	0.10 (0.0039)	0.10 (0.0039)
Keyway Depth (GE)	3.00…3.10 (0.118…0.122)	3.50…3.60 (0.138…0.142)	4.00…4.20 (0.158…0.165)	4.00…4.20 (0.158…0.165)
Keyway Width (F)	4.97…5.00 (0.196…0.197)	5.97…6.00 (0.235…0.236)	7.96…8.00 (0.314…0.315)	7.964…8.000 (0.3135…0.3150)

MPF-A/B3*xx*, MPF-A/B4*xx*, MPF-A/B45*xx*, MPF-A/B5*xx* Motor Dimensions

Motor Series MPF-	AB mm (in.)	AD mm (in.)	HD mm (in.)	T mm (in.)	LA mm (in.)	LD (1) mm (in.)	LE (1) mm (in.)	L (1) mm (in.)	LB (1) mm (in.)	L-LB (2) mm (in.)	D mm (in.)	M mm (in.)	S (3) mm (in.)	N mm (in.)	P mm (in.)	GE mm (in.)	F mm (in.)
A/B310	66.0 (2.60)	87.25 (3.43)	133.4 (5.25)	2.74 (0.11)	9.91 (0.39)	102.0 (4.03)	62.0 (2.45)	168.0 (6.62)	128.0 (5.04)	40.0 (1.57)	16.0 (0.629)	100.0 (3.94)	7.0 (0.283)	80.0 (3.15)	92.39 (3.64)	3.0 (0.118)	5.0 (0.197)
A/B320						128.0 (5.03)	88.0 (3.45)	193.0 (7.62)	153.0 (6.04)								
A/B330						153.0 (6.03)	113.0 (4.45)	219.0 (8.62)	179.0 (7.04)								
A/B430	67.7 (2.66)	90.9 (3.58)	142.0 (5.59)	2.74 (0.11)	10.16 (0.40)	150.0 (5.89)	110.0 (4.31)	215.0 (8.48)	175.0 (6.90)	40.0 (1.57)	19.0 (0.748)	115.0 (4.53)	10.0 (0.401)	95.0 (3.74)	102.1 (4.02)	3.5 (0.138)	6.0 (0.236)
A/B4530	67.7 (2.66)	98.6 (3.88)	157.6 (6.20)	2.74 (0.11)	12.19 (0.48)	153.0 (6.02)	113.0 (4.44)	229.0 (9.0)	179.0 (7.03)	50.0 (1.97)	24.0 (0.945)	130.0 (5.12)	10.0 (0.401)	110.0 (4.33)	118.1 (4.65)	4.0 (0.158)	8.0 (0.315)
A/B4540						178.0 (7.02)	138.0 (5.44)	254.0 (10.0)	204.0 (8.03)								
A/B540	71.2 (2.80)	136.4 (5.37)	209.0 (8.23)	3.12 (0.12)	13.97 (0.55)	202.0 (7.95)	162.0 (6.38)	287.0 (11.30)	227.0 (8.92)	60.0 (2.36)	28.0 (1.102)	165.0 (6.50)	12.0 (0.481)	130.0 (5.12)	145.3 (5.72)	4.0 (0.158)	8.0 (0.315)

(1) If ordering an MPF-A/B310, MPF-A/B320, or MPF-A/B330 motor with brake, add 35 mm (1.38 in.) to dimensions L, LB, LD, and LE.
If ordering an MPF-A/B430 motor with brake, add 48 mm (1.89 in.) to dimensions L, LB, LD, and LE.
If ordering an MPF-A/B4530 or MPF-A/B4540 motor with brake, add 23 mm (0.90 in.) to dimensions L, LB, LD, and LE.
If ordering an MPF-A/B540 motor with brake, add 51 mm (2.0 in.) to dimensions L, LB, LD, and LE.

(2) Tolerance for this dimension is ±0.7 mm (±0.028 in.).

(3) Tolerance for this dimension is: MPF-A/B3*xx*, MPF-A/B4*xx*, or MPF-A/B45*xx* +0.36 mm (±0.007 in.) and MPF-A/B5*xx* +0.43 mm (±0.008 in.).

Motors are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

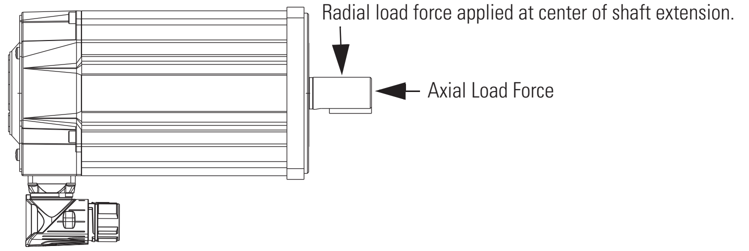
MP-Series Food Grade Motor Load Force Ratings

Bulletin MPF motors are capable of operating with the maximum radial or maximum axial shaft loads listed in the following tables. Radial loads listed are applied in the middle of the shaft extension. The tables below represent an L_{10} bearing fatigue life of 20,000 hours. This 20,000-hour life does not account for possible application-specific life reduction that may occur due to bearing grease contamination from external sources. Maximum operating speed is limited by motor winding.

Radial Load Force Ratings

Motor Cat. No.	**500 rpm** kg (lb)	**1000 rpm** kg (lb)	**2000 rpm** kg (lb)	**3000 rpm** kg (lb)	**3500 rpm** kg (lb)	**4000 rpm** kg (lb)	**5000 rpm** kg (lb)
MPF-A/B310	78 (172)	62 (137)	49 (108)	–	40 (88)	–	36 (79)
MPF-A/B320	87 (192)	69 (152)	55 (121)	–	45 (99)	–	40 (88)
MPF-A/B330	–	74 (163)	59 (130)	–	49 (108)	–	43 (95)
MPF-A/B430	106 (234)	84 (185)	67 (148)	–	55 (121)	–	49 (108)
MPF-A/B4530	133 (293)	105 (232)	84 (185)	73 (161)	–	66 (146)	–
MPF-A/B4540	140 (309)	111 (245)	89 (196)	77 (170)	–	–	–
MPF-A/B540	–	143 (316)	114 (251)	99 (219)	–	90 (199)	–

MPF-*xxxxx*-*x*J7*x*AB Load Forces



Axial Load Force Ratings (maximum radial load)

Motor Cat. No.	**500 rpm** kg (lb)	**1000 rpm** kg (lb)	**2000 rpm** kg (lb)	**3000 rpm** kg (lb)	**3500 rpm** kg (lb)	**4000 rpm** kg (lb)	**5000 rpm** kg (lb)
MPF-A/B310	30 (66)	23 (51)	16 (35)	–	13 (29)	–	11 (24)
MPF-A/B320	34 (75)	25 (55)	19 (42)	–	15 (33)	–	13 (29)
MPF-A/B330	–	27 (60)	20 (44)	–	16 (35)	–	13 (29)
MPF-A/B430	52 (115)	39 (86)	29 (64)	–	22 (49)	–	19 (42)
MPF-A/B4530	45 (99)	34 (75)	25 (55)	21 (46)	–	19 (42)	–
MPF-A/B4540	49 (108)	36 (79)	27 (60)	22 (49)	–	–	–
MPF-A/B540	–	48 (107)	35 (79)	30 (66)	–	26 (58)	–

Axial Load Force Ratings (zero radial load)

Motor Cat. No.	500 rpm kg (lb)	1000 rpm kg (lb)	2000 rpm kg (lb)	3000 rpm kg (lb)	3500 rpm kg (lb)	4000 rpm kg (lb)	5000 rpm kg (lb)
MPF-A/B310	49 (108)	36 (79)	27 (60)	–	21 (46)	–	18 (40)
MPF-A/B320	49 (108)	36 (79)	27 (60)	–	21 (46)	–	18 (40)
MPF-A/B330	–	36 (79)	27 (60)	–	21 (46)	–	18 (40)
MPF-A/B430	69 (152)	51 (112)	38 (84)	–	30 (66)	–	25 (55)
MPF-A/B4530	69 (152)	51 (112)	38 (84)	31 (68)	–	28 (62)	–
MPF-A/B4540	69 (152)	51 (112)	38 (84)	31 (68)	–	–	–
MPF-A/B540	–	67 (149)	49 (109)	41 (92)	–	36 (81)	–

MP-Series Food Grade Motor Catalog Number

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your motor. For questions regarding product availability, contact your Allen-Bradley distributor.

MP F - *x xx xx x -x* J 7 *x* B A

Factory Options
A = Standard

Mounting Flange
B = IEC metric (oversized), free mounting holes (type FF)

Brake
2 = No brake
4 = 24V DC brake

Connectors
7 = Circular (SpeedTec) DIN connector, right angle, 180° rotatable

Enclosure/Shaft Key
J = IP66/IP67 housing/Keyed shaft

Feedback
M = Multi-turn, 1024 sin/cos, absolute encoder (Hiperface protocol)
S = Single-turn, 1024 sin/cos, absolute encoder (Hiperface protocol)

Rated Speed (1)
F = 3000 rpm
H = 3500 rpm
K = 4000 rpm
P = 5000 rpm

Magnet Stack Length (1)
10 = 25.4 mm (1.0 in.)
20 = 50.8 mm (2.0 in.)
30 = 76.2 mm (3.0 in.)
40 = 101.6 mm (4.0 in.)

Frame Size
3 = 100 mm
4 = 115 mm
45 = 130 mm
5 = 165 mm

Voltage Class
A = 200V
B = 400V

Series Type
F = Food grade enclosure (including shaft seal)

Series
MP = Premium permanent magnet rotary servo motor

(1) Not all combinations are available. Only the configurations for rated speed and magnet stack length as listed in MP-Series Food Grade (230V) Performance Specifications (page 50) and MP-Series Food Grade Motor (460V) Performance Specifications (page 51) are available.

MP-Series Stainless Steel Motors

The MP-Series stainless-steel motors are specifically designed to meet the unique needs of hygienic environments. With these servo motors, the benefits of Kinetix Integrated Motion are available to a greater range of applications, allowing the use of servo motors even in high pressure, highly caustic washdown environments. The MP-Series Stainless Steel motors extend the wide range of Allen-Bradley servo motors into new applications on food, beverage, brewing, dairy, pharmaceutical, and health and beauty manufacturing equipment.

For drive compatibility, refer to Servo Drives on page 14.

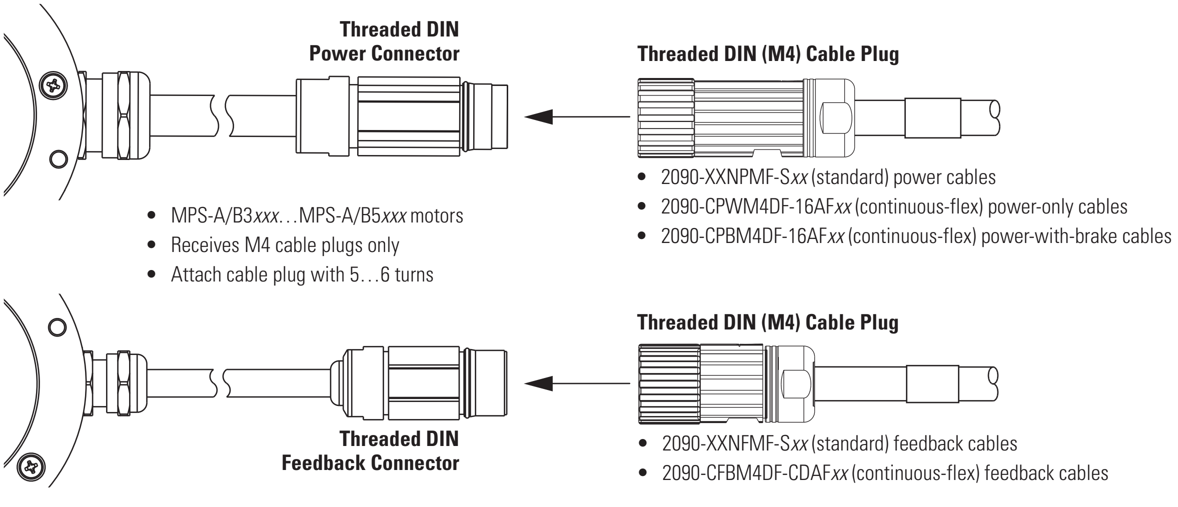
MP-Series Stainless Steel Motor High Resolution Encoder Features

MP-Series Stainless Steel motors are available with high performance encoders with a choice of Single-turn (-S) or Multi-turn (-M) high resolution feedback.

- Up to 2 million counts per revolution for smooth performance
- Single-turn encoder provides high-resolution absolute position feedback within one turn.
- Multi-turn encoder provides high-resolution absolute position feedback within 4096 turns.

Motor Connector/Cable Compatibility

MP-Series (Bulletin MPS) motors are equipped with threaded DIN connectors.



- 2090-XXNPMF-S*xx* (standard) power cables
- 2090-CPWM4DF-16AF*xx* (continuous-flex) power-only cables
- 2090-CPBM4DF-16AF*xx* (continuous-flex) power-with-brake cables

- MPS-A/B3*xxx*…MPS-A/B5*xxx* motors
- Receives M4 cable plugs only
- Attach cable plug with 5…6 turns

- 2090-XXNFMF-S*xx* (standard) feedback cables
- 2090-CFBM4DF-CDAF*xx* (continuous-flex) feedback cables

MP-Series Stainless Steel Motor Options

MP-Series Stainless Steel motors are available with these options:

- 24V DC brake.
- Shaft seal kit (with slinger) is available for field replacement. Shaft seals are made of PTFE. Kits include a lubricant to reduce wear.
- Positive Air Pressure accessory kit (catalog number MPS-AIR-PURGE) is mounted to the rear cover of the motor to provide positive air pressure and further reduce the chance of contamination inside the motor. No special tool is required for installation and removal.

Refer to the MP-Series Stainless Steel Servo Motor Installation Instructions, publication MP-IN005, for more information.

Motor Shaft Seal Kit Combinations and Dimensions

Motor Series	Shaft Seal Cat. No.	Shaft Seal Dimensions			Slinger Dimensions		
		Inside Diameter mm (in.)	**Outside Diameter** mm (in.)	**Width** mm (in.)	**Inside Diameter** mm (in.)	**Outside Diameter** mm (in.)	**Width** mm (in.)
MPS-A3*xx* and MPF-B3*xx*	MPS-SST-A3B3	23.0 (0.90)	47.0 (1.85)	6.0 (0.24)	16.0 (0.63)	50.8 (2.0)	5.1 (0.20)
MPS-A45*xx* and MPF-B45*xx*	MPS-SST-A45B45	31.0 (1.22)	62.0 (2.44)	7.0 (0.27)	24.0 (0.94)	63.5 (2.50)	5.2 (0.20)
MPS-B5*xx*	MPS-SST-F165	36 (1.42)	72 (2.84)	8 (0.32)	28 (1.10)	82.6 (3.25)	5.1 (0.20)

MP-Series Stainless Steel Motor Performance Specifications

MP-Series Stainless Steel Motor (230V) Performance Specifications

Motor MPS-	**Max Speed** rpm	**Continuous Stall Torque** Nm (lb-in)	**Peak Stall Torque** Nm (lb-in)	**Motor Rated Output** kW	**Speed at Motor Rated Output** rpm	**Rotor Inertia** [1] kg-m^2 (lb-in-s^2)	**Motor Weight, approx.** [1] kg (lb)
A330P	5000	3.6 (32)	11.1 (98)	1.3	5000	0.00012 (0.0010)	7.4 (16.2)
A4540F	3000	8.1 (72)	27.1 (240)	1.4	3000	0.00052 (0.0046)	13 (28.5)

(1) Refer to MP-Series Stainless Steel Motor Brake Specifications on page 56 for Brake Rotor Inertia.

System Combinations (230V)

For MP-Series Stainless Steel Motors and	Refer to
Kinetix 6000 (230V) drives	page 546
Kinetix 2000 drives	page 567
Kinetix 300 (240V) drives	page 611
Ultra3000/5000 (230V) drives	page 657

MP-Series Stainless Steel Motor (460V) Performance Specifications

Motor MPS-	**Max Speed** rpm	**Continuous Stall Torque** Nm (lb-in)	**Peak Stall Torque** Nm (lb-in)	**Motor Rated Output** kW	**Speed at Motor Rated Output** rpm	**Rotor Inertia** [1] [2] kg-m^2 (lb-in-s^2)	**Motor Weight, approx.** [1] kg (lb)
B330P	5000	3.6 (32)	11.1 (98)	1.3	5000	0.00012 (0.0010)	7.4 (16.2)
B4540F	3000	8.1 (72)	27.1 (240)	1.4	3000	0.00052 (0.0046)	13 (28.5)
B560F	3000	21.5 (190)	67.8 (600)	3.5	3000	0.00227 (0.0200)	30 (66)

(1) Refer to MP-Series Stainless Steel Motor Brake Specifications on page 58 for Brake Rotor Inertia.

(2) Rotor inertia may vary slightly depending on feedback.

System Combinations (460V)

For MP-Series Stainless Steel Motors and	Refer to
Kinetix 6000 and Kinetix 6200/6500 (460V) drives	page 547
Kinetix 300 (480V) drives	page 612
Ultra3000/5000 (460V) drives	page 658

MP-Series Stainless Steel Motor Brake Specifications

MP-Series Stainless Steel Motor Brake Specifications

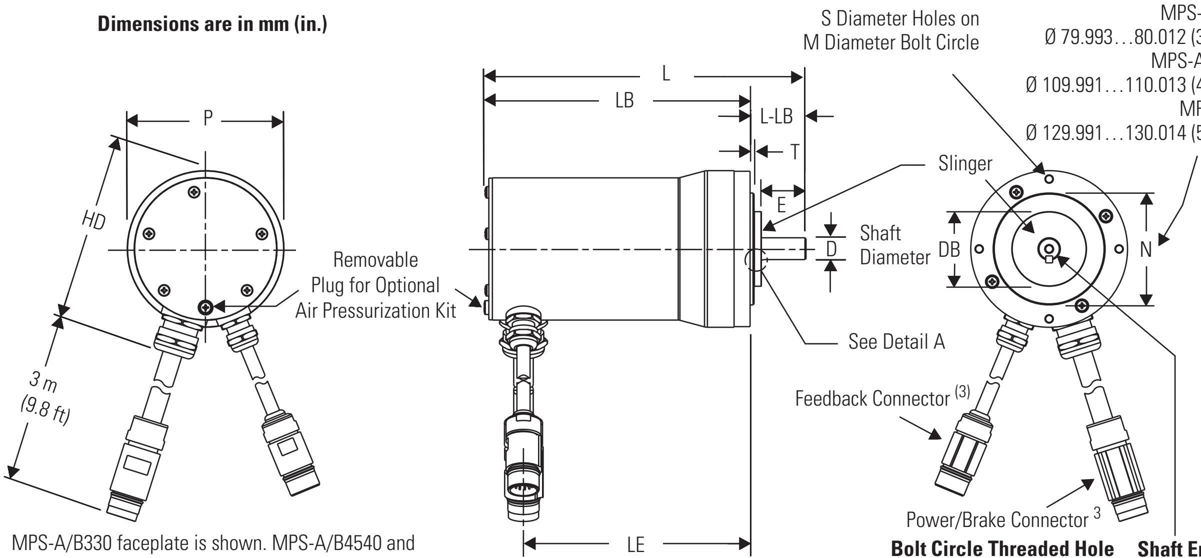
Motor MPS-	**Max Backlash (brake engaged)** arc minutes	**Holding Torque** Nm (lb-in)	**Coil Current at 24V DC** A	**Brake Response Time**			**Brake Rotor Inertia** kg-m^2 (lb-in-s^2)	**Brake Motor Weight, approx.** kg (lb)
				Release ms	**Engage (using external arc suppression device)**			
					MOV ms	**Diode** ms		
A/B330	45	4.18 (37)	0.45…0.55	50	20	110	0.00013 (0.0011)	8.8 (19.3)
A/B4540	48	10.2 (90)	0.576…0.704	110	25	160	0.00052 (0.0046)	15.4 (34.0)
B560	25	28.3 (250)	1.05…1.28	70	50	250	0.00227 (0.0200)	32.2 (70.8)

MP-Series Stainless Steel Motor Dimensions

MPS-A/B330, MPS-A/B4540, MPS-B560 Motor Dimensions

Dimensions are in mm (in.)

Pilot Diameter Tolerances
MPS-A/B330 Motors:
Ø 79.993…80.012 (3.1493…3.1501)
MPS-A/B4540 Motors:
Ø 109.991…110.013 (4.3303…4.3312)
MPS-B560 Motors:
Ø 129.991…130.014 (5.1178…5.1187)



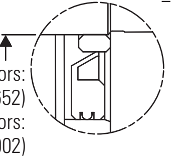
MPS-A/B330 faceplate is shown. MPS-A/B4540 and MPS-B560 faceplate has additional screws evenly spaced.

Bolt Circle Threaded Hole
MPS-A/B330 Motors:
Thread - M6 x 1.0
Thread Depth - 7.0 (0.28)
MPS-A/B4540 Motors:
Thread - M8 x 1.25-6H
Thread Depth - 9.0 (0.35)
MPS-B560 Motors:
Thread - M10 x 1.5-6H
Thread Depth - 11 (0.43)

Shaft End Threaded Hol
MPS-A/B330 Motors:
Thread - M5 x 0.8-6H
Thread Depth - 12.5 (0.49)
MPS-A/B4540 Motors:
Thread - M8 x 1.25-6H
Thread Depth - 19 (0.75)
MPS-B560 Motors:
Thread - M10 x 1.5-6H
Thread Depth - 22 (0.87)

Shaft Diameter Tolerances
MPS-A/B330 Motors:
Ø 15.997…16.008 (0.6298…0.6301)
MPS-A/B4540 Motors:
Ø 23.996…24.009 (0.9448…0.9451)
MPS-B560 Motors:
Ø 27.996…28.009 (1.1022…1.1027)

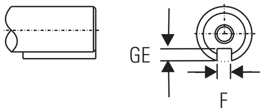
Detail A



Flush to Pilot
± 0.83 (0.032)

MPS-A/B330 Motors:
Ø 16.891…16.896 (0.6650…0.6652)
MPS-A/B4540 Motors:
Ø 24.94 ±0.05 (0.982 ±0.002)
MPS-A/B560 Motors:
Ø 29.92 ±0.05 (1.178 ±0.002)

Shaft Detail with Key



MPS-A/B330 = 5 x 5 x 25 Key
MPS-A/B4540 = 8 x 7 x 32 Key
MPS-B560 = 8 x 7 x 40 Key

Optional Shaft Seal
(with slinger)
Refer to page 57
for MP-Series
Stainless Steel motor
shaft seal kit information.

Shaft, Pilot, and Keyway Tolerances	MPS-A/B330	MPS-A/B4540	MPS-B560
Shaft Runout (T.I.R.)	0.035 (0.0014)	0.04 (0.0016)	0.04 (0.0016)
Pilot Eccentricity (T.I.R.)	0.08 (0.0031)	0.10 (0.0039)	0.10 (0.0039)
Max Face Runout (T.I.R.)	0.08 (0.0031)	0.10 (0.0039)	0.10 (0.0039)
Keyway Depth (GE)	3.00…3.10 (0.118…0.122)	4.00…4.20 (0.158…0.165)	4.00…4.20 (0.158…0.165)
Keyway Width (F)	4.97…5.00 (0.196…0.197)	7.96…8.00 (0.314…0.315)	7.96…8.00 (0.314…0.315)

Motor Series MPS-	**HD** mm (in.)	**T** mm (in.)	**E** mm (in.)	**LE (1)** mm (in.)	**L (1)** mm (in.)	**LB (1)** mm (in.)	**L-LB (2)** mm (in.)	**D** mm (in.)	**DB** mm (in.)	**M** mm (in.)	**S** mm (in.)	**N** mm (in.)	**P** mm (in.)	**GE** mm (in.)	**F** mm (in.)
A/B330	135 (5.31)	2.87 (0.113)	32.1 (1.26)	162 (6.38)	230 (9.05)	190 (7.49)	40.0 (1.57)	16.0 (0.63)	50.8 (2.0)	100 (3.94)	7.0 (0.28)	80.0 (3.15)	112 (4.41)	3.0 (0.118)	5.0 (0.197)
A/B4540	164 (6.46)	3.38 (0.133)	41.4 (1.63)	185 (7.30)	266 (10.45)	216 (8.48)	50.0 (1.97)	24.0 (0.945)	63.5 (2.50)	130 (5.12)	9.0 (0.35)	110 (4.33)	143.2 (5.64)	4.0 (0.158)	8.0 (0.315)
B560	198 (7.79)	3.38 (0.13)	51.52 (2.03)	302 (11.90)	396 (15.60)	336 (13.24)	60.0 (2.36)	28.0 (1.10)	82.6 (3.25)	165 (6.49)	11.0 (0.43)	130 (5.12)	181 (7.13)	4.0 (0.158)	8.0 (0.315)

(1) If ordering an MPS-A/B330 motor with brake, add 35 mm (1.38 in.) to dimensions L and LB, and 34 mm (1.34 in.) to dimension LE. If ordering an MPS-A/B4540 motor with brake add 48.5 mm (1.91 in.) to dimensions L, LB, and LE.

(2) Tolerance for this dimension is ±0.7 mm (±0.028 in.).

(3) Specifications for the 3 m (9.8 ft) power and feedback cable leads are identical to those of the 2090-XXNPMF-*xx*S*xx* (power) and 2090-XXNFMF-S*xx* (feedback) cables. Refer to 2090-Series Motor/Acutator Cables with SpeedTec DIN Connectors Overview beginning on page 396 for more information.

Motors are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

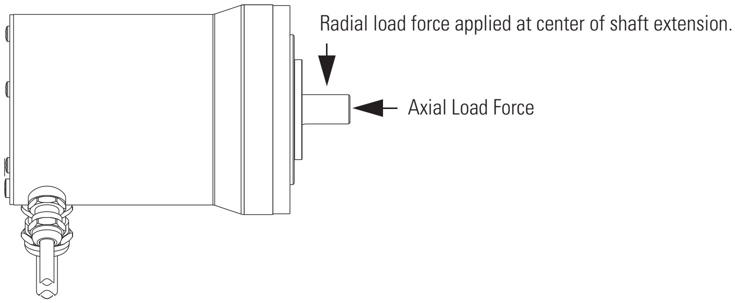
MP-Series Stainless Steel Motor Load Force Ratings

Bulletin MPS motors are capable of operating with the maximum radial or maximum axial shaft loads listed in the following tables. Radial loads listed are applied in the middle of the shaft extension. The tables starting below represent an L_{10} bearing fatigue life of 20,000 hours. This 20,000-hour life does not account for possible application-specific life reduction that may occur due to bearing grease contamination from external sources. Maximum operating speed is limited by motor winding.

Radial Load Force Ratings

Motor Cat. No.	**500 rpm** kg (lb)	**1000 rpm** kg (lb)	**2000 rpm** kg (lb)	**3000 rpm** kg (lb)	**3500 rpm** kg (lb)	**4000 rpm** kg (lb)	**5000 rpm** kg (lb)
MPS-A/B330	–	74 (163)	59 (130)	–	49 (108)	–	43 (95)
MPS-A/B4540	140 (309)	111 (245)	89 (195)	77 (170)	–	–	–
MPS-B560	–	154 (338)	122 (268)	106 (234)	–	–	–

MPS-*xxxxx*-*x*J5*x*DA Load Forces



Axial Load Force Ratings (maximum radial load)

Motor Cat. No.	**500 rpm** kg (lb)	**1000 rpm** kg (lb)	**2000 rpm** kg (lb)	**3000 rpm** kg (lb)	**3500 rpm** kg (lb)	**4000 rpm** kg (lb)	**5000 rpm** kg (lb)
MPS-A/B330	–	27 (59)	20 (44)	–	16 (35)	–	13 (29)
MPS-A/B4540	49 (107)	36 (80)	27 (59)	22 (49)	–	–	–
MPS-B560	–	52 (115)	39 (85)	32 (71)	–	–	–

Axial Load Force Ratings (zero radial load)

Motor Cat. No.	**500 rpm** kg (lb)	**1000 rpm** kg (lb)	**2000 rpm** kg (lb)	**3000 rpm** kg (lb)	**3500 rpm** kg (lb)	**4000 rpm** kg (lb)	**5000 rpm** kg (lb)
MPS-A/B330	–	36 (79)	27 (59)	–	21 (46)	–	18 (40)
MPS-A/B4540	69 (152)	51 (112)	38 (83)	31 (69)	–	–	–
MPS-B560	–	68 (149)	50 (109)	42 (92)	–	–	–

MP-Series Stainless Steel Motor Catalog Number

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your motor. For questions regarding product availability, contact your Allen-Bradley distributor.

MP S - *x xx xx x*-*x* J 5 *x* D A

Factory Options
A = Standard

Mounting Flange
D = IEC metric, tapped mounting holes (type FT)

Brake
2 = No brake
4 = 24V DC brake

Connectors (1)
5 = Cables with circular DIN (threaded) connector

Enclosure/Shaft Key/Shaft Seal
J = IP66/IP67/IP69K housing/Shaft key/Shaft seal

Feedback
M = Multi-turn, 1024 sin/cos, absolute encoder (Hiperface protocol)
S = Single-turn, 1024 sin/cos, absolute encoder (Hiperface protocol)

Rated Speed (2)
F = 3000 rpm
P = 5000 rpm

Magnet Stack Length (2)
30 = 76.2 mm (3.0 in.)
40 = 101.6 mm (4.0 in.)
60 = 152.4 mm (6.0 in.)

Frame Size
3 = 100 mm
45 = 130 mm
5 = 165 mm

Voltage Class
A = 200V
B = 400V

Series Type
S = Stainless Steel

Series
MP = Premium permanent magnet rotary servo motor

(1) The motor has 3 m (9.8 ft) cables with nickel-plated connector extensions.

IMPORTANT The connectors are O-ring sealed, but not designed to withstand direct high-pressure washdown with aggressive cleaning compounds. The 3 m (9.8 ft) cables are provided so the connectors can be positioned in an area away from direct exposure to the cleaning process, such as within washdown-rated conduit or junction boxes.

(2) Not all combinations are available. Only the configurations for rated speed and magnet stack length as listed in MP-Series Stainless Steel Motor Performance Specifications (page 58) and MP-Series Stainless Steel Motor (460V) Performance Specifications (page 58) are available.

RDD-Series Direct Drive Servo Motors

The RDD-Series (Bulletin RDB) direct-drive servo motor design provides direct-coupling to the load, thus improving system performance and efficiency by eliminating the need for inefficient mechanical power transmission devices, such as gearboxes, timing belts and pulleys. The initial RDD-Series offering incorporates a bearingless housed configuration designed for applications where the load is already supported by its own bearings.

For drive compatibility, refer to Servo Drives on page 14.

RDD-Series Direct-drive Motor High-resolution Encoder Features

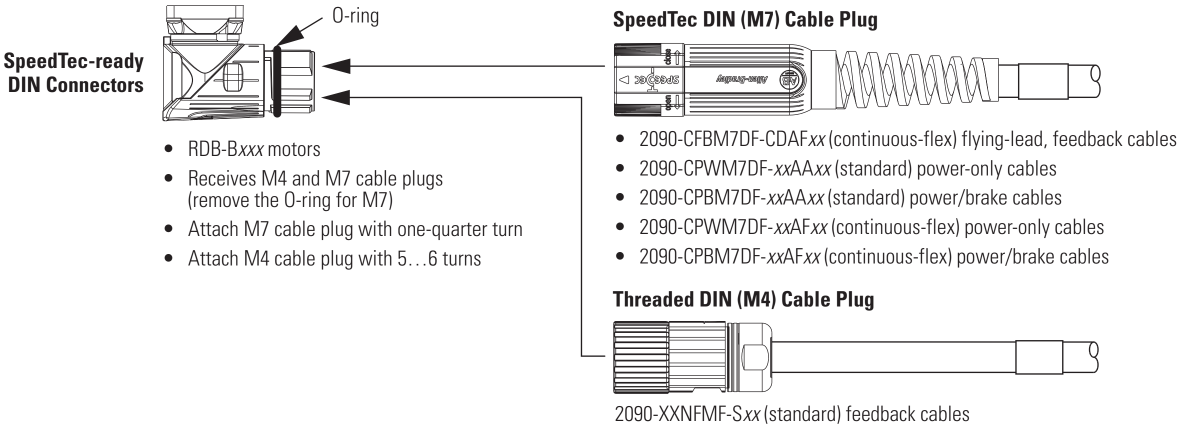
RDD-Series direct drive motors are available with high performance encoders with a choice of Single-turn (-3) or Multi-turn (-7) high-resolution feedback.

- Up to 4 million counts per revolution (-3 and -7) for smooth performance.
- Single-turn encoder provides high-resolution absolute position feedback within one turn.
- Multi-turn encoder provides high-resolution absolute position feedback within 4096 turns. The electromechanical design does not require a battery.

Use the 2090-K6CK-KENDAT feedback module (Kinetix 6000 drives) and 2090-K7CK-KENDAT feedback module (Kinetix 7000 drives) for wiring motor feedback connections. Support for Kinetix 6200 and Kinetix 6500 drives has not been implemented.

Motor Connector/Cable Compatibility

RDD-Series motors are equipped with SpeedTec-ready DIN connectors, however, EnDat encoder feedback requires the additional conductors included in the cables listed.



- RDB-B*xxx* motors
- Receives M4 and M7 cable plugs (remove the O-ring for M7)
- Attach M7 cable plug with one-quarter turn
- Attach M4 cable plug with 5…6 turns

- 2090-CFBM7DF-CDAF*xx* (continuous-flex) flying-lead, feedback cables
- 2090-CPWM7DF-*xx*AA*xx* (standard) power-only cables
- 2090-CPBM7DF-*xx*AA*xx* (standard) power/brake cables
- 2090-CPWM7DF-*xx*AF*xx* (continuous-flex) power-only cables
- 2090-CPBM7DF-*xx*AF*xx* (continuous-flex) power/brake cables

2090-XXNFMF-S*xx* (standard) feedback cables

RDD-Series Direct Drive Motor Performance Specifications

Motor RDB-	**Base Speed** rpm	**Max Speed** rpm	**Continuous Stall Torque** Nm (lb-in)	**Peak Stall Torque** Nm (lb-in)	**Motor Rated Output** kW	**Speed at Motor Rated Output** rpm	**Rotor Inertia** kg-m^2 (lb-in-s^2)	**Motor Weight, approx.** kg (lb)
B21519	750	1235	32.7 (289)	86.5 (766)	3.64	1235	0.0094 (0.083)	19.1 (42)
B2151C	1500	2125			5.23	2125		
B21529	750	1035	45.4 (402)	116 (1027)	4.33	1035	0.0126 (0.112)	24.5 (54)
B2152C	1500	2125			6.41	2125		
B21539	750	1250	53.7 (475)	143 (1266)	5.34	1250	0.0157 (0.139)	29.5 (65)
B2153C	1500	2250			5.87	1772		
B29014	200	450	49.2 (435)	110 (974)	1.97	391	0.028 (0.25)	28.6 (63)
B29016	375	785			3.18	729		
B29019	750	1500			3.63	1128		
B29024	200	435	98.0 (867)	214 (1894)	3.33	413	0.047 (0.42)	42.7 (94)
B29026	375	885			4.05	632		
B29029	750	1200						
B29034	200	500	140 (1239)	318 (2815)	5.16	493	0.066 (0.58)	55.4 (122)
B29036	375	750			5.49	646		
B29039	750	1000	122 (1080)	318 (2815)	4.41	578		
B41014	200	385	183 (1620)	340 (3009)	5.20	360	0.123 (1.09)	67.6 (149)
B41016	375	700			4.83	440		
B41018	625	700						
B41024	200	365	332 (2938)	690 (6107)	7.29	350	0.225 (1.99)	108 (238)
B41026	375	600	308 (2726)					
B41035	250	490	426 (3770)	1050 (9293)	8.69	361	0.302 (2.67)	136 (300)

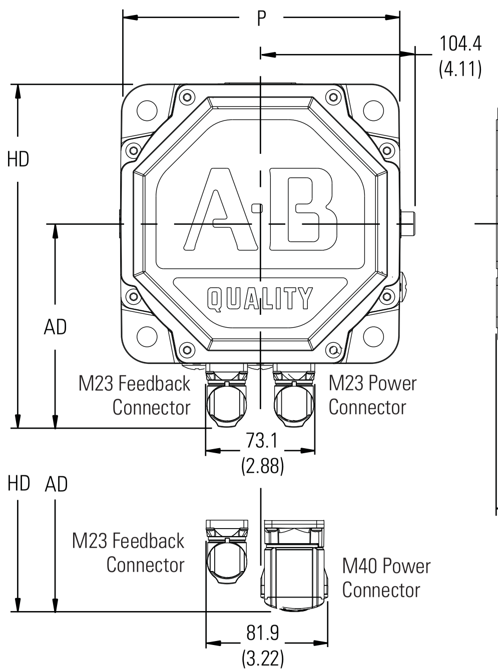
System Combinations (460V)

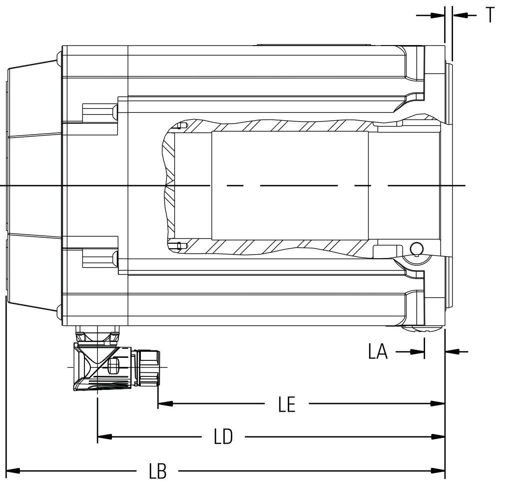
For RDD-Series Direct Drive Motors and	**Refer to**
Kinetix 6000 and Kinetix 6200/6500 (460V) drives	page 549
Kinetix 7000 drives	page 591

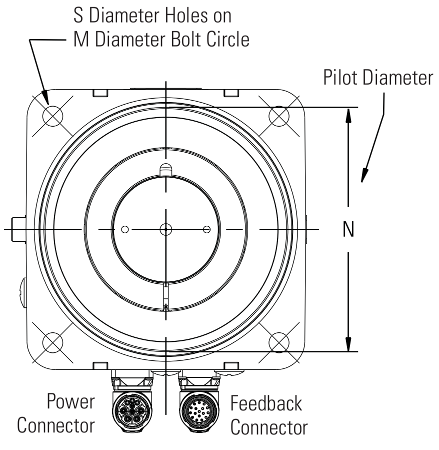
RDD-Series Direct Drive Motor Dimensions

Bulletin RDB-B215*xx* Motor Dimensions

Dimensions are in mm (in.)







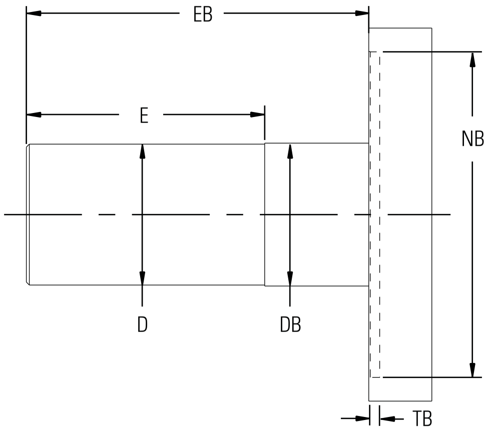
Pilot Diameter (N) Tolerance
Ø 163.989…164.014
(6.4563…6.4572)

Pilot Diameter (NB) Tolerance
Ø 164.040…164.090
(6.4583…6.4602)

Shaft Diameter (D) Tolerance
Ø 70.985…71.000
(2.7947…2.7953)

Shaft Diameter (DB) Tolerance
Ø 71.985…72.000
(2.8340…2.8346)

Machine Mounting Dimensions

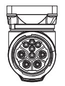
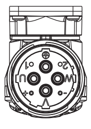


Shaft and Pilot Tolerances	RDB-B215*xx*
Shaft Runout (T.I.R.)	0.13 (0.005)
Pilot Concentricity (T.I.R.)	0.10 (0.004)
Mounting Surface Perpendicularity	0.10 (0.004)

Bulletin RDB-B215*xx* Motor Dimensions

Motor Series RDB-	**AD** (1) mm (in.)	**HD** (1) mm (in.)	**T** mm (in.)	**LA** mm (in.)	**LD** mm (in.)	**LE** mm (in.)	**LB** (2) mm (in.)	**D** mm (in.)	**DB** mm (in.)	**E** (3) mm (in.)	**EB** (4) mm (in.)	**TB** mm (in.)	**M** mm (in.)	**S** (5) mm (in.)	**N** mm (in.)	**NB** mm (in.)	**P** mm (in.)
B2151	136.7 (5.38)	230.9 (9.09)	5.0 (0.197)	14.0 (0.55)	166 (6.52)	124 (4.90)	226 (8.90)	71.0 (2.795)	72.0 (2.834)	49.0 (1.93)	104 (4.09)	5.5 (0.22)	215 (8.465)	13.50 (0.5315)	164 (6.456)	164 (6.459)	189 (7.44)
B2152					200 (7.86)	158 (6.24)	260 (10.24)			83.0 (3.27)	138 (5.43)						
B2153					234 (9.20)	192 (7.58)	294 (11.58)			117 (4.61)	172 (6.77)						

(1) Dimension is for motor with M23 power connector. For motor with M40 power connector, add 18.6 (0.73 in.).
(2) Tolerance for this dimension is ±0.52 mm (±0.02 in.).
(3) Tolerance for this dimension is ±0.40 mm (±0.015 in.).
(4) Tolerance for this dimension is ±1.50 mm (±0.06 in.) static, ±0.13 mm (±0.005 in.) dynamic.
(5) Tolerance for this dimension is +0.430, -0.000 mm (+0.0169, -0.0000 in.).

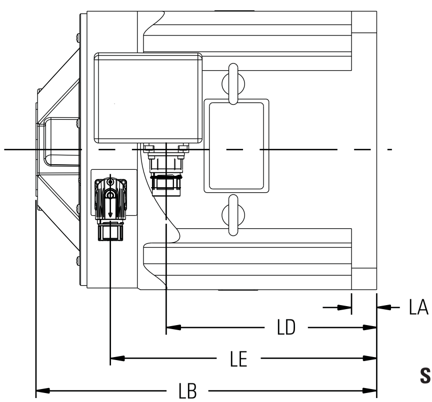
Power Connectors on RDD-Series Motors		**RDB-B215*xx***
M23 Power Connector		RDB-B21519, RDB-B21529, RDB-B2151C, RDB-B21539, RDB-B2152C
M40 Power Connector		RDB-B2153C

Motors are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

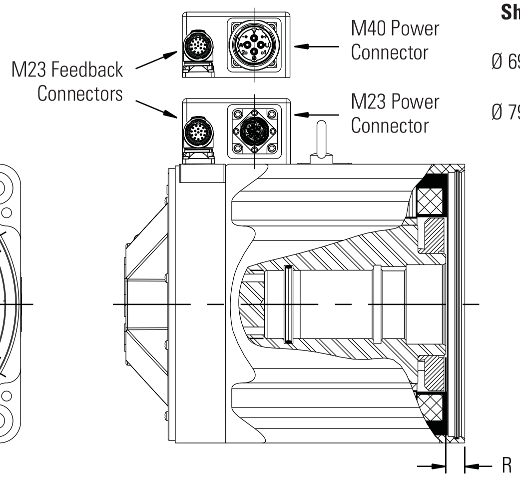
Bulletin RDB-B290*xx* and RDB-B410*xx* Motor Dimensions

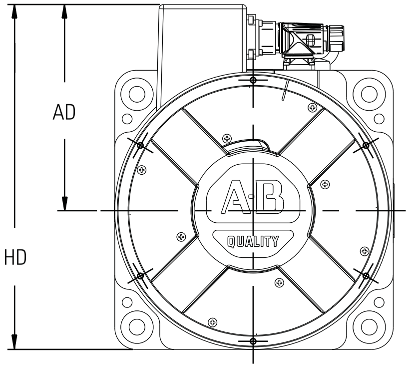
Dimensions are in mm (in.)

Shaft, Pilot, and Keyway Tolerances	RDB-B290*xx*	RDB-B410*xx*
Shaft Runout (T.I.R.)	0.038 (0.0015)	
Pilot Concentricity (T.I.R.)	0.05 (0.002)	
Mounting Surface Perpendicularity	0.05 (0.002)	
Keyway Depth (G)	24.80…24.99 (0.976…0.984)	29.80…29.99 (1.173…1.181)
Keyway Depth (GD)	7.90…8.00 (0.311…0.315)	
Keyway Width (F)	11.957…12.000 (0.4707…0.4724)	

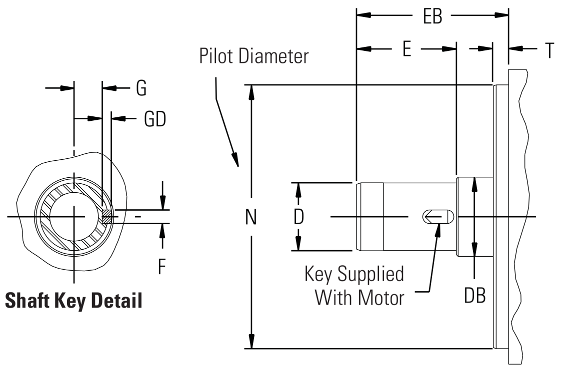


RDB-B290*xx* and RDB-B410*xx* motors have either M23 or M40 power connectors, with no significant difference in dimensions.





Machine Mounting Dimensions

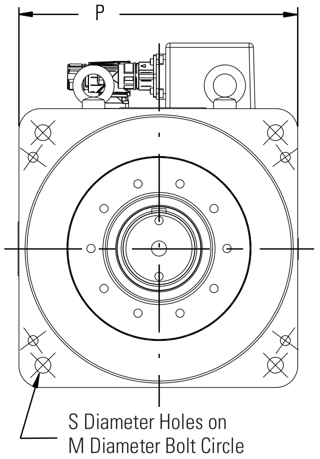


Shaft Key Detail

Shaft Diameter (D) Tolerance
RDB-B290*xx* Motors:
Ø 59.988…59.999 (2.3617…2.3622)
RDB-B410*xx* Motors:
Ø 69.988…69.999 (2.7554…2.7559)

Shaft Diameter (DB) Tolerance
RDB-B290*xx* Motors:
Ø 69.988…69.999 (2.7554…2.7559)
RDB-B410*xx* Motors:
Ø 79.988…79.999 (3.1491…3.1496)

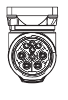
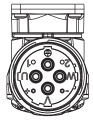
Pilot Diameter Tolerance
RDB-B290*xx* Motors:
Ø 232.92…232.96 (9.170…9.172)
RDB-B410*xx* Motors:
Ø 333.94…333.98 (13.147…13.149)



Bulletin RDB-B290*xx* and RDB-B410*xx* Motor Dimensions

Motor Series RDB-	**AD** mm (in.)	**HD** mm (in.)	**T** mm (in.)	**LA** [1] mm (in.)	**LD** mm (in.)	**LE** mm (in.)	**LB** [2] mm (in.)	**D** mm (in.)	**DB** mm (in.)	**E** [3] mm (in.)	**EB** [4] mm (in.)	**M** mm (in.)	**S** mm (in.)	**N** mm (in.)	**P** [5] mm (in.)	**G** mm (in.)	**GD** mm (in.)	**F** mm (in.)
B2901	182.3 (7.18)	305.9 (12.05)	13.5 (0.53)	22.2 (0.88)	86.5 (3.40)	136 (5.34)	201 (7.92)	59.9 (2.362)	69.9 (2.755)	43.94 (1.730)	88.92 (3.540)	290 (11.417)	14.0 (0.551)	232.9 (9.17)	245.9 (9.68)	24.99 (0.984)	8.00 (0.315)	12.0 (0.472)
B2902					136 (5.36)	185 (7.30)	251 (9.90)			88.14 (3.470)	134.11 (5.280)							
B2903					186 (7.31)	235 (9.25)	301 (11.83)			124.7 (4.910)	170.69 (6.720)							
B4101	256.3 (10.09)	432.1 (17.01)	17.8 (0.70)	25.4 (1.00)	105 (4.14)	164 (6.46)	230 (9.05)	69.9 (2.755)	79.9 (3.149)	40.39 (1.590)	114.05 (4.490)	410 (16.142)	17.5 (0.689)	333.9 (13.14)	350.0 (13.78)	29.99 (1.181)	8.00 (0.315)	12.0 (0.472)
B4102					174 (6.86)	233 (9.18)	300 (11.77)			83.82 (3.300)	167.89 (6.610)							
B4103					243 (9.58)	302 (11.90)	368 (14.49)			118.62 (4.670)	253.49 (9.980)							

(1) Tolerance for this dimension is ±2.15 mm (±0.085 in.).
(2) Tolerance for this dimension is ±2.30 mm (±0.09 in.).
(3) Tolerance for this dimension is ±0.13 mm (±0.005 in.).
(4) Tolerance for this dimension is ±1.50 mm (±0.060 in.) static, ±0.05 mm (±0.002 in.) dynamic.
(5) Tolerance for this dimension is ±1.52 mm (±0.06 in.).

Power Connectors on RDD-Series Motors		**RDB-B290*xx***	**RDB-B410*xx***
M23 Power Connector		RDB-B29014, RDB-B29016, RDB-B29024, RDB-B29019, RDB-B29034, RDB-B29026	N/A
M40 Power Connector		RDB-B29036, RDB-B29029, RDB-B29039	RDB-B410*xx*

Motors are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

RDD-Series Direct Drive Servo Motor Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your motor. For questions regarding product availability, contact your Allen-Bradley distributor.

RD B - B *xxx* *x* *x* - *x* B 7 2 AA

Factory Options
AA = Standard

Brake
2 = No brake

Connectors
7 = Circular (SpeedTec) DIN connector, right angle, 180° rotatable

Enclosure/Shaft
B = IP65 housing/Blind bore

Feedback
7 = Multi-turn, 2048 sin/cos, absolute encoder (EnDat protocol)
3 = Single-turn, 2048 sin/cos, absolute encoder (EnDat protocol)

Base Speed @ 440V (1)
4 = 200 rpm
5 = 250
6 = 375 rpm
8 = 625
9 = 750 rpm
C = 1500 rpm

Magnet Stacks (1)
1 = 1 stack
2 = 2 stacks
3 = 3 stacks
4 = 4 stacks

Frame Size - Bolt Circle Diameter (BCD)
215 = 215 mm
290 = 290 mm
410 = 410 mm

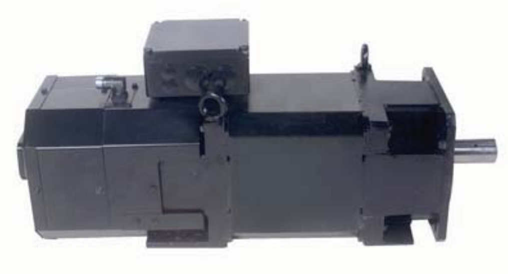
Voltage Class
B = 400V

Housing Type
B = Bearingless housing

Series
RD = Rotary direct drive servo motor

(1) Not all combinations are available. Only the configurations for rated speed and magnet stacks as listed in RDD-Series Direct Drive Motor Performance Specifications on page 63 are available.

HPK-Series Asynchronous Servo Motors



The HPK-Series Asynchronous Servo Motors employ proven induction motor technology optimized for servo system performance. These high horsepower motors offer exceptional performance for rapid acceleration and deceleration. Team these AC motors with the Kinetix 7000 high power servo drives to extend the range of Kinetix Integrated Motion solutions up to 150 kW.

For drive compatibility, refer to Servo Drives on page 14.

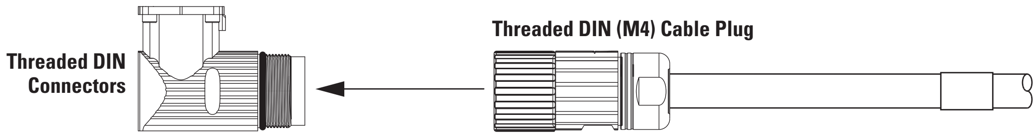
HPK-Series Asynchronous Servo Motor Encoder Features

HPK-Series motors are available with high performance encoders with a choice of Single-turn (-S) or Multi-turn (-M) high-resolution feedback.

- Up to 2 million counts per revolution for smooth performance.
- Single-turn encoder provides high-resolution absolute position feedback within one turn.
- Multi-turn encoder provides high-resolution absolute position feedback within 4096 turns.

Motor Connector/Cable Compatibility

HPK-Series motors are equipped with threaded DIN connectors.

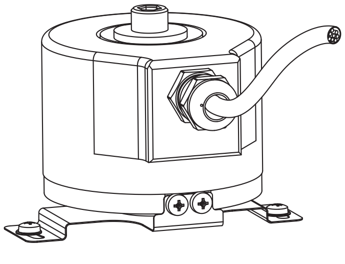


- HPK-B/E*xxxx* motors
- Receives only M4 cable plugs
- Attach cable plug with 5…6 turns

- 2090-XXNFMF-S*xx* (standard) feedback cables
- 2090-CFBM4DF-CDAF*xx* (continuous-flex) feedback cables

HPK-Series Asynchronous Servo Motor Options

- Multiple junction box mounting locations (F1, F2, and F3)
- Holding brake, 380V...460V AC
- 460V and 400V windings
- Encoder kit (catalog number HPK-*xxxxxx*-ENC-*xx*) available for field replacement. Kits are pre-programmed for a specific motor catalog number. Kit includes cable and DIN connector.



HPK-Series Asynchronous Servo Motor Performance Specifications

HPK-Series Asynchronous Servo Motor (460V) Performance Specifications

Motor HPK-	**Base Speed** rpm	**Continuous Stall Torque** Nm (lb-in)	**Peak Stall Torque** Nm (lb-in)	**Motor Rated Output at Base Speed** kW (Hp)	**Rotor Inertia** (1) kg-m^2 (lb-in-s^2)	**Motor Weight, approx.** kg (lb)
B1307C	1500	112 (991)	260 (2301)	17.1 (22.9)	0.081 (0.7168)	135 (297)
B1308C		141 (1247)	262 (2319)	21.6 (28.9)	0.098 (0.8673)	152 (335)
B1310C		155 (1372)	325 (2876)	23.8 (31.9)	0.111 (0.9823)	166 (366)
B1613C		271 (2398)	541.9 (4796)	41.7 (55.9)	0.206 (1.8231)	275 (606)
B1815C		360 (3186)	850 (7523)	55.9 (74.9)	0.468 (4.1418)	474 (1045)
B2010C		482 (4266)	970 (8585)	75.0 (100.5)	0.885 (7.8322)	531 (1170)
B2212C		714 (6319)	1356 (12,000)	112 (150)	1.900 (16.829)	1095 (2414)
B2510C		955 (8452)	1927 (17,054)	150 (200)	3.070 (27.192)	1005 (2216)
B1307E	3000	96 (849)	165 (1460)	29.8 (39.9)	0.081 (0.7168)	135 (297)
B1308E		115 (1018)	230 (2035)	35.7 (47.8)	0.098 (0.8673)	152 (335)
B1609E		156 (1381)	270 (2390)	48.4 (64.8)	0.147 (1.3009)	231 (469)
B1611E		183 (1619)	400 (3540)	57.0 (76.4)	0.177 (1.5664)	244 (538)
B1613E		237 (2097)	459 (4062)	73.7 (98.8)	0.206 (1.8231)	275 (606)
B2010E		295 (2610)	500 (4425)	92.0 (125)	0.885 (7.8322)	351 (1170)

(1) Rotor inertia may vary slightly depending on feedback.

HPK-Series Asynchronous Servo Motor (400V) Performance Specifications

Motor HPK-	**Base Speed** rpm	**Continuous Stall Torque** Nm (lb-in)	**Peak Stall Torque** Nm (lb-in)	**Motor Rated Output at Base Speed** kW (Hp)	**Rotor Inertia** (1) kg-m^2 (lb-in-s^2)	**Motor Weight, approx.** kg (lb)
E1307C	1500	112 (991)	263 (2327)	17.1 (22.9)	0.081 (0.7168)	135 (297)
E1310C		155 (1372)	380 (3363)	23.8 (32.4)	0.111 (0.9823)	166 (366)
E1613C		271 (2398)	625 (5531)	41.7 (55.9)	0.206 (1.8231)	275 (606)
E1815C		360 (3186)	840 (7434)	55.9 (74.9)	0.468 (4.1418)	474 (1045)
E2010C		482 (4266)	870 (7700)	75.0 (100.5)	0.885 (7.8322)	531 (1170)
E1307E	3000	96.0 (849)	202 (1788)	29.8 (39.9)	0.081 (0.7168)	135 (297)
E1308E		107 (947)	200 (1770)	33.2 (45.0)	0.098 (0.8673)	152 (335)
E1609E		156 (1381)	359 (3176)	48.4 (64.9)	0.147 (1.3009)	213 (469)
E1611E		183 (1619)	430 (3805)	57.0 (76.4)	0.177 (1.5664)	244 (538)
E1613E		237 (2097)	430 (3805)	73.7 (98.8)	0.206 (1.8231)	275 (606)

(1) Rotor inertia may vary slightly depending on feedback.

System Combinations

For HPK-Series Motors and	Refer to
Kinetix 7000 drives	page 574

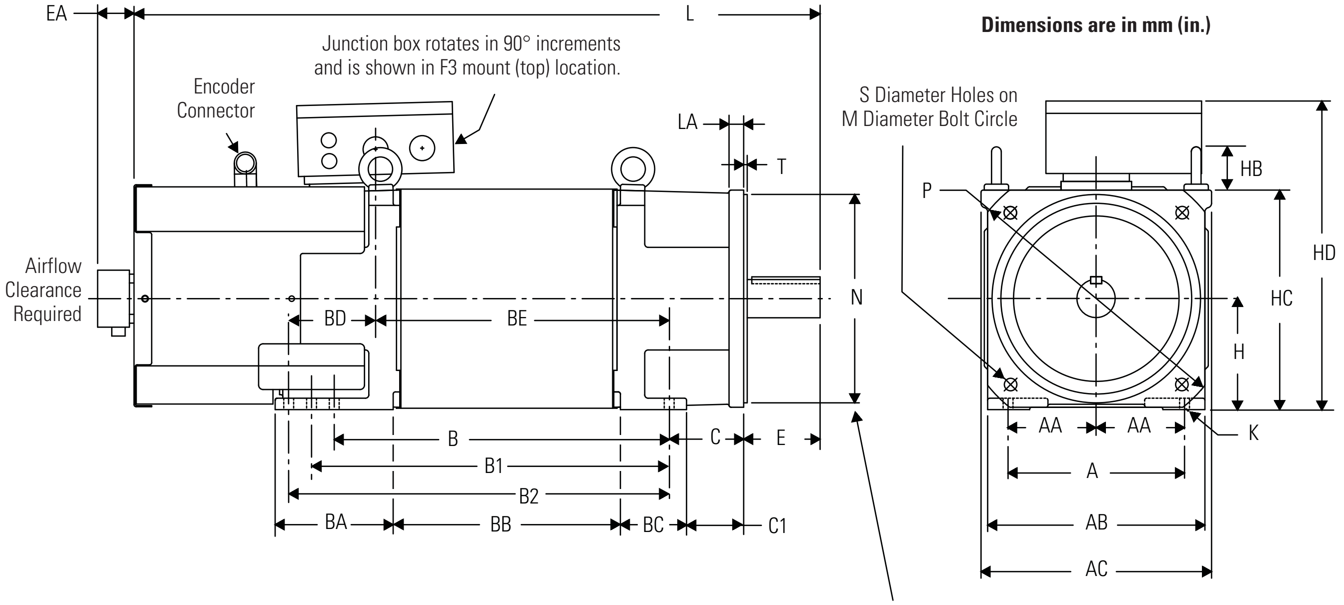
HPK-Series Asynchronous Servo Motor Brake Specifications

Motor Cat. No. HPK-	**Holding Torque** Nm (lb-in)
B/E1307C	20 (177)
B/E1308C	
B/E1310C	
B/E1613C	
B/E1815C	48 (425)
B/E2010C	
B2212C	102 (903)

Motor Cat. No. HPK-	**Holding Torque** Nm (lb-in)
B/E1307E	20 (177)
B/E1308E	
B/E1609E	
B/E1611E	
B/E1613E	
B/E2010E	48 (425)

HPK-Series Asynchronous Servo Motor Dimensions

HPK-B/E13*xx* and HPK-B/E16*xx* Motor (non-brake) Dimensions



Dimensions are in mm (in.)

Junction box rotates in 90° increments and is shown in F3 mount (top) location.

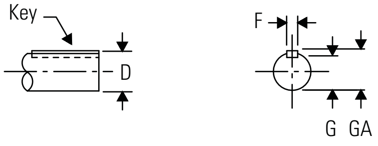
S Diameter Holes on M Diameter Bolt Circle

Airflow Clearance Required

Shaft Diameter Tolerances
HPK-B/E1307, 1308, 1310
Ø 48.003…48.016 (1.8899…1.8904)
HPK-B/E1609, 1611, 1613
Ø 55.011…55.030 (2.1658…2.1665)

Pilot Diameter Tolerances
HPK-B/E1307, 1308, 1310
Ø 249.99…250.02 (9.842…9.844)
HPK-B/E1609, 1611, 1613
Ø 299.99…300.02 (11.809…11.811)

Shaft Detail with Key



HPK-B/E1307, 1308, 1310 = 14 x 9 x 80 Key
HPK-B/E1609, 1611, 1613 = 16 x 10 x 90 Key

Shaft, Pilot, and Keyway Tolerances	**HPK-B/E13*xx***	**HPK-B/E16*xx***
Shaft Runout (T.I.R.)	0.06 (0.002)	0.06 (0.002)
Max Face Runout (T.I.R.)	0.13 (0.005)	0.13 (0.005)
Keyway Depth (GA)	51.30…51.9 (2.02…2.04)	58.67…58.92 (2.31…2.32)
Keyway Depth (G)	42.42…42.92 (1.67…1.69)	48.50…49.00 (1.91…1.93)
Keyway Width (F)	13.94…13.99 (0.549…0.551)	15.95…16.00 (0.628…0.630)

HPK-B/E13*xx* and HPK-B/E16*xx* Motor (non-brake) Dimensions

Motor Series HPK-B/E	**H** mm (in.)	**HC** mm (in.)	**HD** mm (in.)	**HB** mm (in.)	**L** mm (in.)	**EA** mm (in.)	**LA** mm (in.)	**T** mm (in.)	**P** mm (in.)	**M** mm (in.)	**S** (1) mm (in.)	**N** mm (in.)	**D** mm (in.)	**G** mm (in.)	**GA** mm (in.)	**F** mm (in.)
1307	132 (5.20)	262 (10.3)	366 (14.4)	53.0 (2.09)	806 (31.7)	52.0 (2.05)	17.0 (0.67)	5.0 (0.20)	350 (13.7)	300 (11.8)	18.5 (0.73)	250 (9.84)	48.0 (1.89)	42.4 (1.67)	51.3 (2.02)	13.9 (0.55)
1308					845 (33.2)											
1310					876 (34.4)											
1609	160 (6.30)	316 (12.4)	444 (17.4)	62.0 (2.44)	886 (34.8)	52.0 (2.05)	21.0 (0.83)	5.0 (0.20)	400 (15.7)	350 (13.7)	18.5 (0.73)	300 (11.8)	55.0 (2.16)	48.5 (1.91)	58.7 (2.31)	16.0 (0.63)
1611			450 (17.7)		937 (36.8)											
1613					987 (38.8)											

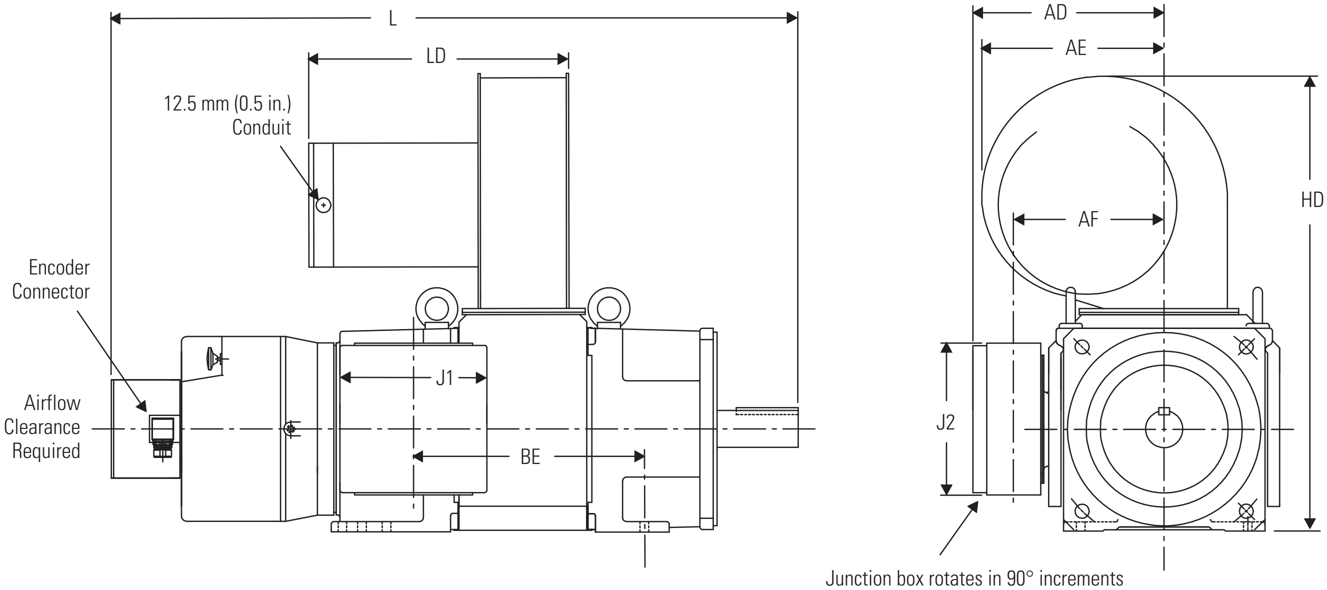
(1) Tolerance for this dimension is +0.52, -0.0 mm (+0.02, -0.0 in.).

Motor Series HPK-B/E	**K** mm (in.)	**A** mm (in.)	**AA** mm (in.)	**AB** mm (in.)	**AC** mm (in.)	**B** mm (in.)	**B1** mm (in.)	**B2** mm (in.)	**BA** mm (in.)	**BB** mm (in.)	**BC** mm (in.)	**C** mm (in.)	**C1** mm (in.)	**E** mm (in.)	**BD** mm (in.)	**BE** mm (in.)
1307	12.0 (0.47)	216 (8.50)	108 (4.25)	260 (10.2)	279 (10.9)	333 (13.1)	365 (14.3)	390 (15.3)	154 (6.06)	187 (7.36)	95.0 (3.74)	89.0 (3.50)	57.0 (2.24)	110 (4.33)	91.0 (3.58)	300 (11.8)
1308						371 (14.6)	403 (15.8)	428 (16.8)		226 (8.90)						338 (13.3)
1310						403 (15.8)	435 (17.1)	462 (18.1)		257 (10.1)						369 (14.5)
1609	14.0 (0.55)	254 (10.0)	127 (5.0)	313 (12.3)	332 (13.0)	N/A	414 (16.3)	452 (17.8)	170 (6.69)	225 (8.86)	95.0 (3.74)	108 (4.25)	82.0 (3.23)	110 (4.33)	130 (5.12)	321 (12.6)
1611						N/A	464 (18.2)	502 (19.7)		276 (10.8)						372 (14.6)
1613						N/A	515 (20.2)	548 (21.5)		327 (12.8)					125 (4.92)	423 (16.6)

Motors are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

HPK-B/E13*xx* and HPK-B/E16*xx* Motor (brake) Dimensions

Dimensions are in mm (in.)



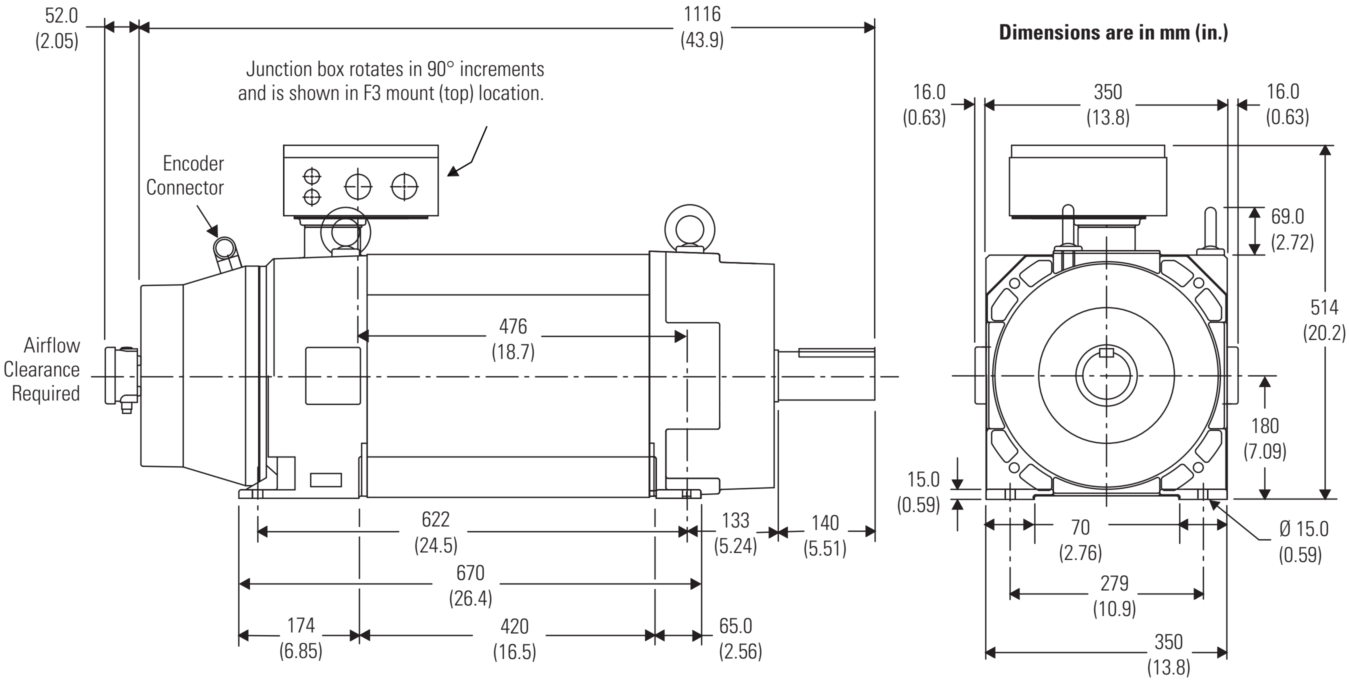
Junction box rotates in 90° increments
and is shown in F1 mount (left side) location,
motor is also available in F2 mount (right side) location.

HPK-B/E13*xx* and HPK-B/E16*xx* Motor (brake) Dimensions

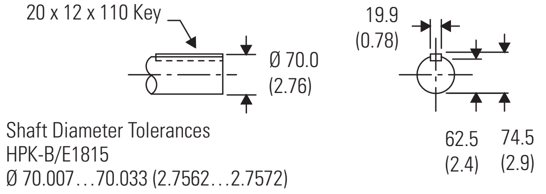
Motor Series HPK-B/E	**L** mm (in.)	**LD** mm (in.)	**BE** mm (in.)	**AD** mm (in.)	**AE** mm (in.)	**AF** mm (in.)	**HD** mm (in.)	**J1** mm (in.)	**J2** mm (in.)
1307	888 (34.9)	336 (13.2)	298 (11.7)	247 (9.72)	236 (9.29)	195 (7.68)	588 (23.1)	190 (7.48)	196 (7.72)
1308	926 (36.4)		336 (13.2)						
1310	957 (37.7)		368 (14.5)						
1609	967 (38.0)	332 (13.0)	328 (12.9)	285 (11.2)	236 (9.29)	225 (8.86)	670 (26.3)	224 (8.82)	228 (8.98)
1611	1018 (40.1)		379 (14.9)						
1613	1069 (42.1)		430 (16.9)						

Motors are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

HPK-B/E1815 Motor (non-brake) Dimensions

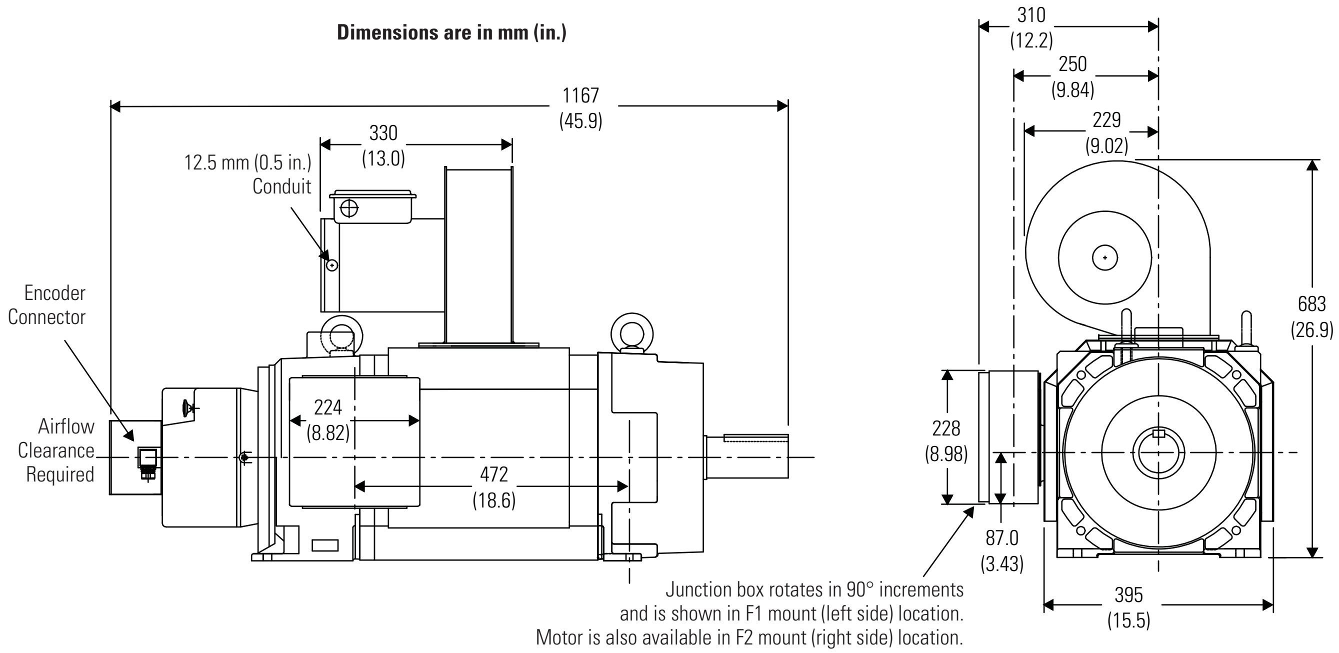


Shaft Detail with Key

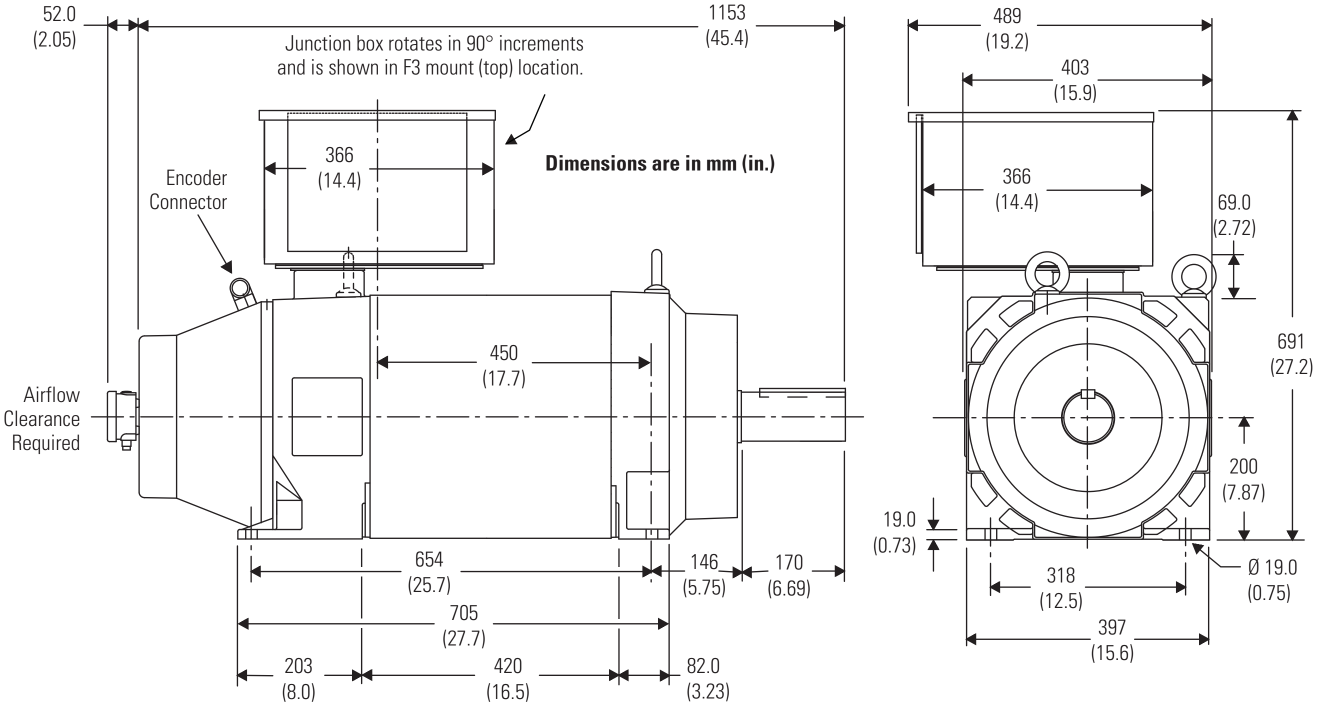


Shaft, Pilot, and Keyway Tolerances	HPK-B/E1815
Shaft Runout (T.I.R.)	0.06 (0.002)
Max Face Runout (T.I.R.)	0.13 (0.005)
Keyway Depth	73.90…74.50 (2.91…2.93)
Keyway Depth	62.00…62.50 (2.44…2.46)
Keyway Width	19.94…19.99 (0.785…0.787)

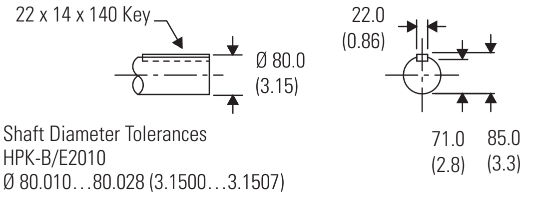
HPK-B/E1815 Motor (brake) Dimensions



HPK-B/E2010 Motor (non-brake) Dimensions



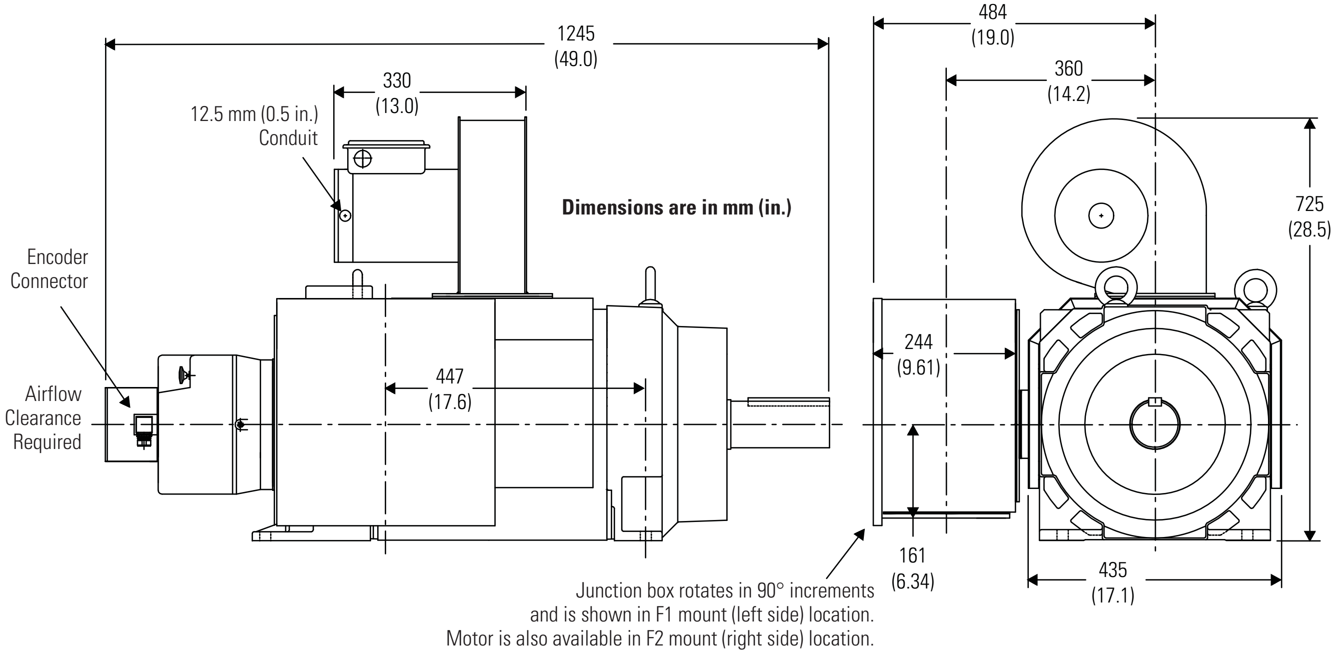
Shaft Detail with Key



Shaft Diameter Tolerances
HPK-B/E2010
Ø 80.010…80.028 (3.1500…3.1507)

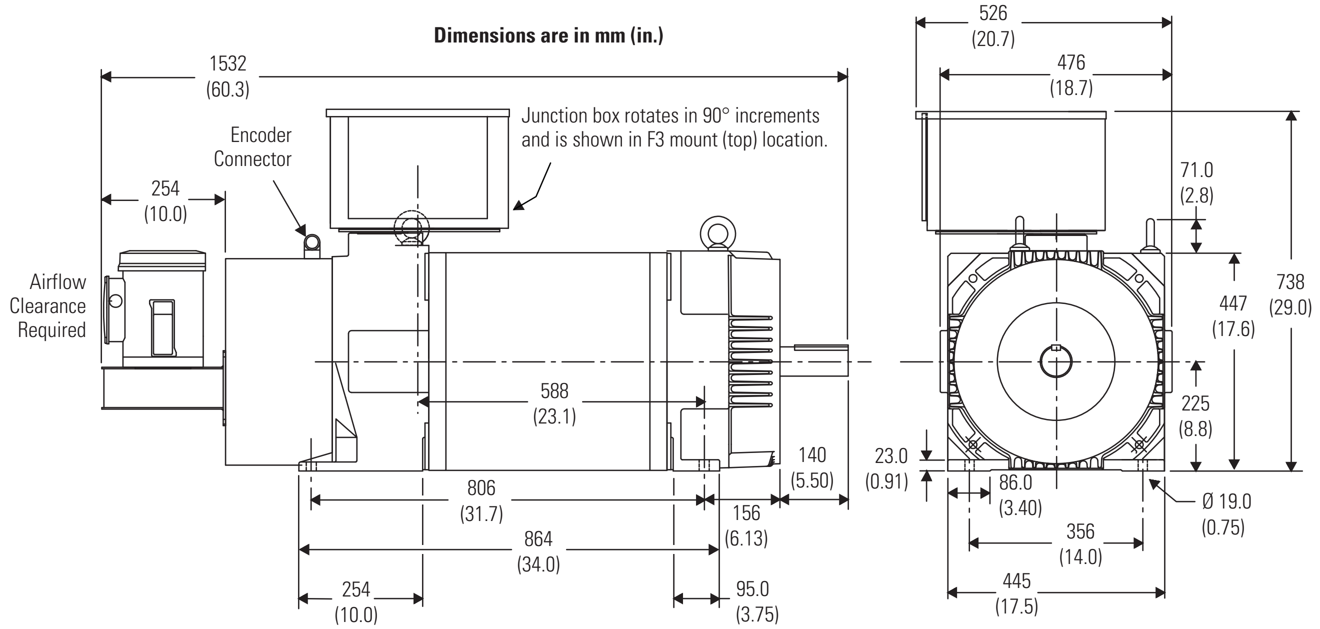
Shaft, Pilot, and Keyway Tolerances	HPK-B/E2010
Shaft Runout (T.I.R.)	0.06 (0.002)
Max Face Runout (T.I.R.)	0.13 (0.005)
Keyway Depth	84.50…85.10 (3.33…3.35)
Keyway Depth	70.60…71.10 (2.78…2.80)
Keyway Width	21.95…22.00 (0.864…0.866)

HPK-B/E2010 Motor (brake) Dimensions

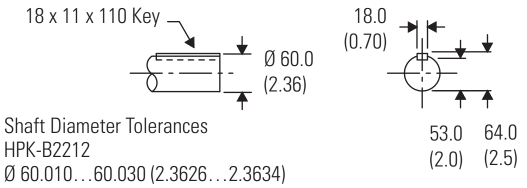


Junction box rotates in 90° increments
and is shown in F1 mount (left side) location.
Motor is also available in F2 mount (right side) location.

HPK-B2212 Motor (non-brake) Dimensions



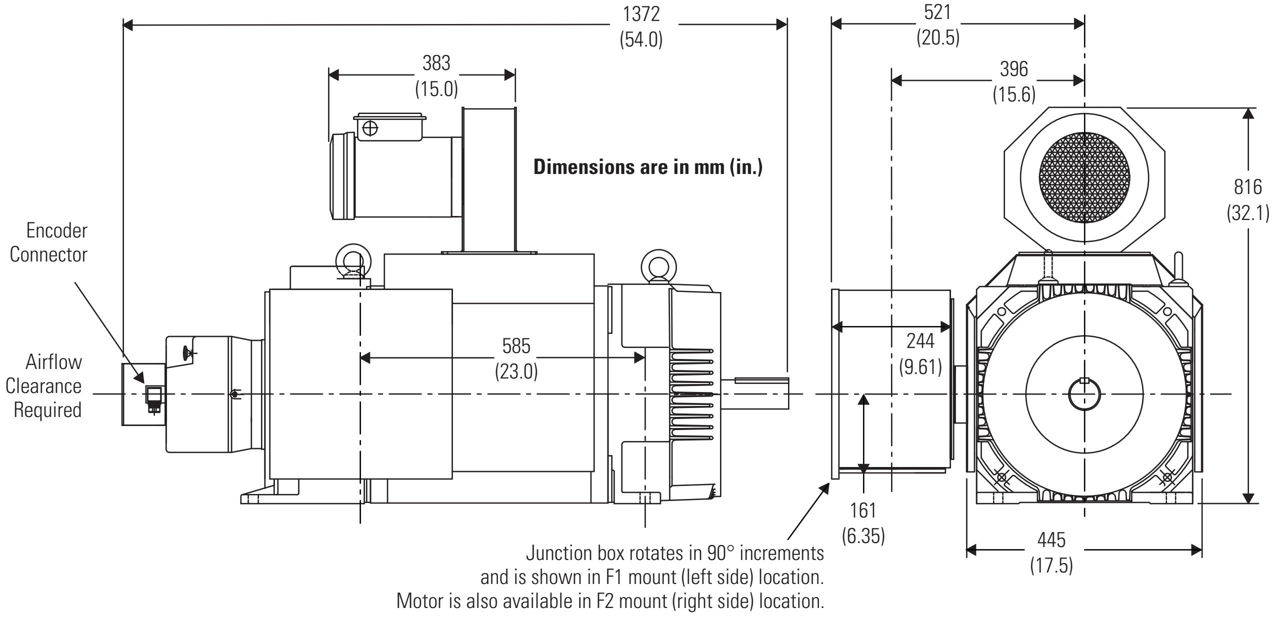
Shaft Detail with Key



Shaft Diameter Tolerances
HPK-B2212
Ø 60.010…60.030 (2.3626…2.3634)

Shaft, Pilot, and Keyway Tolerances	HPK-B2212
Shaft Runout (T.I.R.)	0.06 (0.002)
Max Face Runout (T.I.R.)	0.13 (0.005)
Keyway Depth	63.08…64.08 (2.48…2.59)
Keyway Depth	52.57…53.08 (2.07…2.09)
Keyway Width	21.95…22.00 (0.864…0.866)

HPK-B2212 Motor (brake) Dimensions



Motors are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

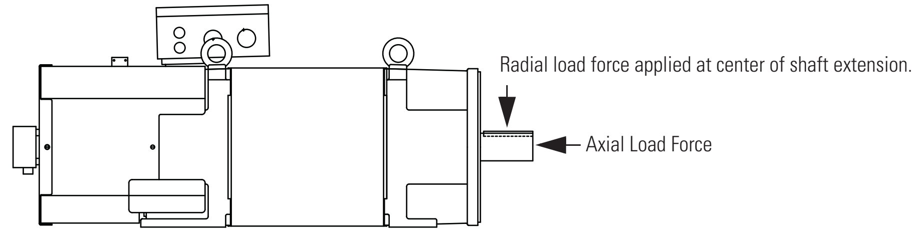
HPK-Series Asynchronous Servo Motor Load Force Ratings

HPK-Series motors are capable of operating with the maximum radial or maximum axial shaft loads listed in the following tables. Radial loads listed are applied in the middle of the shaft extension. The tables starting below represent an L_{10} bearing fatigue life of 10,000 hours. This 10,000-hour life does not account for possible application-specific life reduction that may occur due to bearing grease contamination from external sources. Maximum operating speed is limited by motor winding.

Radial Load Force Ratings (zero axial load)

Motor Cat. No.	**850 rpm** kg (lb)	**1150 rpm** kg (lb)	**1750 rpm** kg (lb)	**2500 rpm** kg (lb)
HPK-B/E1307	320 (704)	290 (638)	250 (550)	220 (485)
HPK-B/E1308	320 (704)	290 (638)	250 (550)	220 (485)
HPK-B/E1310	320 (704)	290 (638)	250 (550)	220 (485)
HPK-B/E1609	500 (1100)	450 (990)	390 (858)	350 (770)
HPK-B/E1611	500 (1100)	450 (990)	390 (858)	350 (770)
HPK-B/E1613	500 (1100)	450 (990)	390 (858)	350 (770)
HPK-B/E1815	530 (1166)	530 (1166)	530 (1166)	450 (990)
HPK-B/E2010	660 (1452)	660 (1452)	660 (1452)	580 (1276)
HPK-B2212	730 (1609)	690 (1521)	600 (1323)	530 (1168)
HPK-B2510				

HPK-*xxxxxx-xxxxx*AA Motor Load Forces



Axial Load Force Ratings (zero radial load)

Motor Cat. No.	**850 rpm** kg (lb)	**1150 rpm** kg (lb)	**1750 rpm** kg (lb)	**2500 rpm** kg (lb)
HPK-B/E1307	260 (572)	240 (528)	210 (462)	180 (396)
HPK-B/E1308	260 (572)	240 (528)	210 (462)	180 (396)
HPK-B/E1310	260 (572)	240 (528)	210 (462)	180 (396)
HPK-B/E1609	360 (796)	330 (726)	290 (638)	250 (550)
HPK-B/E1611	360 (796)	330 (726)	290 (638)	250 (550)
HPK-B/E1613	360 (796)	330 (726)	290 (638)	250 (550)
HPK-B/E1815	440 (970)	380 (838)	310 (682)	260 (572)
HPK-B/E2010	530 (1166)	460 (1012)	370 (814)	310 (682)
HPK-B2212	800 (1764)	730 (1609)	630 (1389)	560 (1235)
HPK-B2510				

HPK-Series Asynchronous Servo Motor Catalog Number

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your motor. For questions regarding product availability, contact your Allen-Bradley distributor.

HPK - x *xxxx x*-*x x x x x* A

Factory Options
A = Standard

Mounting Type/Shaft Key
A = IEC metric, free mounting holes (type FF), foot mount and flange mount/Keyed shaft (1)
B = IEC metric, free mounting holes (type FF), foot mount only/Keyed shaft (2)

Brake
2 = No brake
4 = Holding brake 380…460V AC

Connections
4 = Feedback - right angle, 180° rotatable
Power - junction box

Blower/Junction Box Location
A = Inline blower, F3 junction box (no brake only)
B = Top-mounted blower, F1 junction box (brake only)
C = Top-mounted blower, F2 junction box (brake only)

Feedback
M = Multi-turn, 1024 sin/cos, absolute encoder (Hiperface protocol)
S = Single-turn, 1024 sin/cos, absolute encoder (Hiperface protocol)

Base Speed (3)
C = 1500 rpm
E = 3000 rpm

Frame Size (3)

1307	1609	1815	2510
1308	1611	2010	
1310	1613	2212	

Voltage Class
B = 460V
E = 400V

Series
HPK = HPK-Series asynchronous servo motors

---

(1) Applies to HPK-B/E13*xx* and HPK-B/E16*xx* motors only.

(2) Applies to HPK-B/E18*xx*, HPK-B/E20*xx*, HPK-B22*xx*, and HPK-B25*xx* motors only.

(3) Not all combinations are available. Only the configurations for base speed and frame size as listed in HPK-Series Asynchronous Servo Motor (460V) Performance Specifications (page 70) and HPK-Series Asynchronous Servo Motor (400V) Performance Specifications (page 70) are available.

TL-Series Motors

The TL-Series motors are low-inertia high-performance servo motors featuring metric and NEMA frame sizes. They combine a compact size with a high torque density afforded by their superior stator design. The result is a package that provides substantial power in a small footprint.

TL-Series (Bulletin TLY) Motors

The TL-Series (Bulletin TLY) motors are equipped with circular plastic connectors and when used with the Kinetix 2000, Kinetix 6000, or Ultra3000 drives, the TL-Series (Bulletin TLY) motors are able to offer the benefits of Kinetix Integrated Motion.

For drive compatibility, refer to Servo Drives on page 14.

TL-Series (Bulletin TL) Motors

The TL-Series (Bulletin TL) motor performance specifications match the Bulletin TLY motor specifications identically, but are available only in metric frame sizes and with high-resolution absolute position encoders. Similar to the Bulletin TLY motors, Bulletin TL motors support frame sizes TL-A110, TL-A120, TL-A130, TL-A220, TL-A230, TL-A2530, TL-A2540, and TL-A410 (TL-A310 is not supported).

The Bulletin TL motors are equipped with rectangular plastic connectors and are intended for use with Ultra1500 drives.

TL-Series Motor Encoder Features

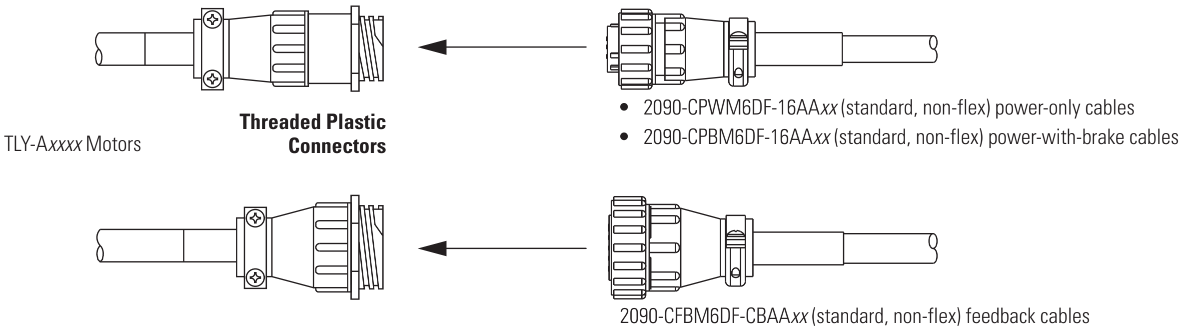
TL-Series motors are available with high-resolution or industry standard incremental encoder feedback.

- High-resolution, high performance encoders providing multi-turn absolute position feedback (131,072 counts/revolution) for smooth performance.
- Industry standard incremental encoder feedback (2,000 counts/revolution) applies to Bulletin TLY motors.

Motor Connector/Cable Compatibility

TL-Series motors are equipped with either threaded or rectangular plastic connectors.

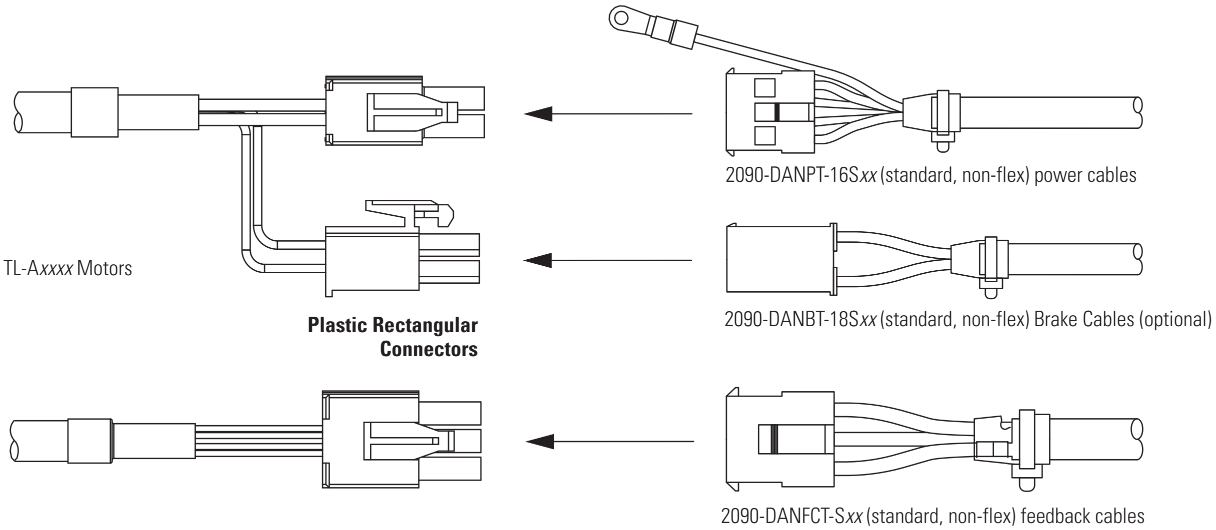
Bulletin TLY Motor Connectors



- 2090-CPWM6DF-16AA*xx* (standard, non-flex) power-only cables
- 2090-CPBM6DF-16AA*xx* (standard, non-flex) power-with-brake cables

2090-CFBM6DF-CBAA*xx* (standard, non-flex) feedback cables

Bulletin TL Motor Connectors



2090-DANPT-16S*xx* (standard, non-flex) power cables

2090-DANBT-18S*xx* (standard, non-flex) Brake Cables (optional)

2090-DANFCT-S*xx* (standard, non-flex) feedback cables

TL-Series Motor Options

TL-Series motors are available with these options:

- 24V DC brake.
- Shaft seal kit is available for field installation. Lubricant is provided with each shaft seal kit to reduce wear.

Motor Shaft Seal Kit Combinations and Dimensions

Motor Cat. No.	Seal Kit Cat. No.	Inside Diameter mm (in.)	Outside Diameter mm (in.)	Width mm (in.)
TLY/TL-A1*xx*	TL-SSN-1	8.9 (0.35)	16 (0.71)	3 (0.12)
TLY/TL-A2*xx*	TL-SSN-2	14 (0.55)	24 (0.95)	5 (0.20)
TLY/TL-A25*xx*	TL-SSN-3	19.8 (0.78)	30 (1.18)	5 (0.20)
TLY-A3*xx*				
TL-A4*xx*	TL-SSN-4	24.0 (0.95)	40 (1.57)	10 (0.39)

Transition Plates for N-Series Retrofit

Transition plates provide a means of retrofitting an existing N-Series motor with a TL-Series (Bulletin TLY) NEMA motor. In most applications, the TL-Series (Bulletin TLY) NEMA motor will be physically smaller, but will deliver the same torque rating as the N-Series motor it is replacing. Transition plates are not available for the N-56*xx* motors. Select your transition plate catalog number from the table below.

N-Series to TL-Series Transition Plates

Transition Plate Cat. No.	Description	Converts from This N-Series Motor	To This TL-Series NEMA Motor
TL-TRPLAT-17-23	TL-Series Transition Plate, NEMA 17 to 23	N-23*xx*	TLY-A1*xxx*-H*xxx*AN
TL-TRPLAT-23-34	TL-Series Transition Plate, NEMA 23 to 34	N-34*xx*	TLY-A2*xxx*-H*xxx*AN
TL-TRPLAT-34-42	TL-Series Transition Plate, NEMA 34 to 42	N-42*xx*	TLY-A25*xxx*-H*xxx*AN

TL-Series Motor Performance Specifications

TL-Series (non-brake) Motor Performance Specifications

Motor Cat. No.	**Max Speed** rpm	**Continuous Stall Torque** Nm (lb-in)	**Peak Stall Torque** Nm (lb-in)	**Motor Rated Output** kW	**Speed at Motor Rated Output** rpm	**Rotor Inertia (1)** kg-m^2 (lb-in-s^2)
TLY-A110 (2)	6000 (3)	0.096 (0.85)	0.20 (1.75)	0.041	5000	0.000001 (0.0000089)
TL-A110 (2)			0.22 (1.94)			
TLY/TL-A120		0.181 (1.60)	0.36 (3.20)	0.086	5000	0.000002 (0.000018)
TLY/TL-A130		0.325 (2.88)	0.76 (6.70)	0.14	5000	0.000003 (0.000027)
TLY/TL-A220		0.836 (7.40)	1.48 (13.1)	0.35	5000	0.000018 (0.00016)
TLY/TL-A230		1.30 (11.50)	3.05 (27.0)	0.44	5000	0.000034 (0.00030)
TLY/TL-A2530	5000	2.60 (23.0)	5.20 (46.0)	0.69	4400	0.000098 (0.00087)
TLY/TL-A2540		2.94 (26.0)	7.10 (63.0)	0.86	4575	0.00011 (0.00096)
TLY-A310	4500	3.61 (32.0)	9.0 (80.0)	0.95	4000	0.00015 (0.0013)
TL-A410 (2)		5.42 (48.0)	13.00 (115)	2.0	4500	0.00036 (0.0032)

(1) Refer to TL-Series Motor Weight Specifications on page 83 for Brake Motor Weight.

(2) The TLY/TL-A110 and TL-A410 motors are available only in metric frame sizes. All other motors are available in metric and NEMA frame sizes.

(3) Applies to TLY-A*xxx*T-H motors with incremental feedback. The TLY/TL-A*xxx*P-B motors with absolute high-resolution encoders are rated at 5000 rpm.

TL-Series (brake) Motor Performance Specifications

Motor Cat. No.	**Max Speed** rpm	**Continuous Stall Torque** Nm (lb-in)	**Peak Stall Torque** Nm (lb-in)	**Motor Rated Output** kW	**Speed at Motor Rated Output** rpm	**Rotor Inertia (1)** kg-m^2 (lb-in-s^2)
TLY-A110 (2)	6000 (3)	0.086 (0.76)	0.20 (1.75)	0.037	5000	0.000004 (0.000035)
TL-A110 (2)			0.22 (1.94)			
TLY/TL-A120		0.163 (1.44)	0.36 (3.20)	0.077	5000	0.000005 (0.000044)
TLY/TL-A130		0.293 (2.59)	0.76 (6.70)	0.13	5000	0.000006 (0.000053)
TLY/TL-A220		0.757 (6.70)	1.48 (13.1)	0.24	5000	0.000028 (0.00025)
TLY/TL-A230		1.16 (10.3)	3.05 (27.0)	0.32	4250	0.000044 (0.00039)
TLY/TL-A2530	5000	2.60 (23.0)	5.20 (46.0)	0.55	3650	0.00012 (0.0011)
TLY/TL-A2540		2.94 (26.0)	7.10 (63.0)	0.66	3750	0.00013 (0.0012)
TLY-A310	4500	3.61 (32.0)	9.0 (80.0)	0.90	3900	0.00017 (0.0017)
TL-A410 (2)		4.86 (43.0)	13.0 (115)	1.80	4500	0.00041 (0.0036)

(1) Refer to TL-Series Motor Weight Specifications on page 83 for Brake Motor Weight.

(2) The TLY/TL-A110 and TL-A410 motors are available only in metric frame sizes. All other motors are available in metric and NEMA frame sizes.

(3) Applies to TLY-A*xxx*T-H motors with incremental feedback. The TLY/TL-A*xxx*P-B motors with absolute high-resolution encoders are rated at 5000 rpm.

System Combinations

For TL-Series Motors and	Bulletin	Feedback Option Compatibility	Refer to
Kinetix 6000 (230V) drives	TLY	Incremental only	page 555
Kinetix 2000 drives		High-resolution or Incremental	page 568
Kinetix 300 (240V) drives		High-resolution or Incremental	page 613
Kinetix 3 drives		High-resolution or Incremental	page 619
	TL	High-resolution only	page 624
Ultra3000/5000 (230V) drives	TLY	Incremental only	page 660

TL-Series Motor Weight Specifications

Motor Cat. No.	High Resolution Feedback Option Bulletin TLY/TL-A*xxxx*-B Motors		Incremental Feedback Option Bulletin TLY-A*xxxx*-H Motors	
	Motor Weight, approx. kg (lb)	**Brake Motor Weight, approx.** kg (lb)	**Motor Weight, approx.** kg (lb)	**Brake Motor Weight, approx.** kg (lb)
TLY/TL-A110	0.29 (0.64)	0.55 (1.2)	0.29 (0.64)	0.55 (1.2)
TLY/TL-A120	0.34 (0.75)	0.59 (1.3)	0.35 (0.78)	0.59 (1.3)
TLY/TL-A130	0.46 (1.0)	0.68 (1.5)	0.50 (1.1)	0.68 (1.5)
TLY/TL-A220	0.95 (2.1)	1.4 (3.0)	1.1 (2.4)	1.5 (3.4)
TLY/TL-A230	1.4 (3.0)	1.8 (4.0)	1.5 (3.3)	2.0 (4.4)
TLY/TL-A2530	2.3 (5.0)	3.2 (7.0)	2.3 (5.1)	3.2 (7.0)
TLY/TL-A2540	2.6 (5.7)	3.5 (7.7)	2.6 (5.8)	3.5 (7.7)
TLY-A310	3.9 (8.6)	4.5 (10.0)	3.9 (8.6)	4.5 (10.0)
TL-A410	5.5 (12.0)	6.80 (15.0)	5.5 (12.0)	6.80 (15.0)

TL-Series Motor Brake Specifications

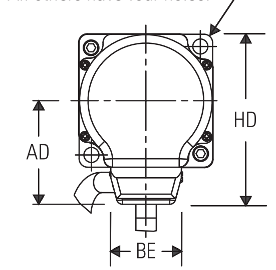
Motor Cat. No.	Max Backlash (brake engaged) arc minutes	Holding Torque Nm (lb-in)	Coil Current at 24V DC A	Brake Response Time		
				Release ms	Engage (using external arc suppression device)	
					MOV ms	**Diode** ms
TLY/TL-A110	60	0.32 (2.8)	0.18…0.22	21	7	40
TLY/TL-A120						
TLY/TL-A130						
TLY/TL-A220		1.24 (11.0)	0.333…0.407	22	13	73
TLY/TL-A230						
TLY/TL-A2530		2.5 (22.0)	0.351…0.429	42	14	86
TLY/TL-A2540						
TLY-A310						
TL-A410		9.3 (82.0)	0.648…0.792	69	20	84

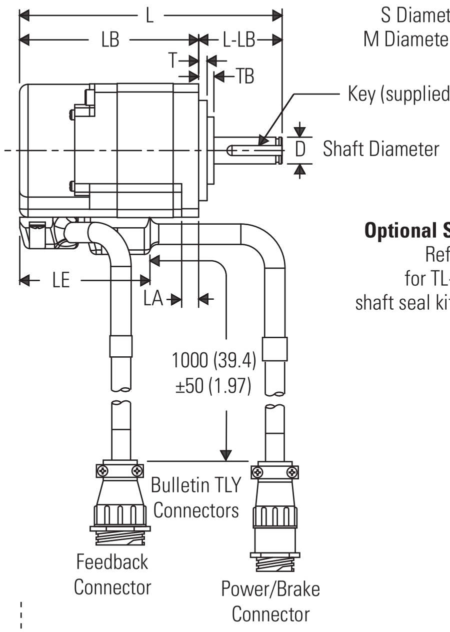
TL-Series Motor Dimensions

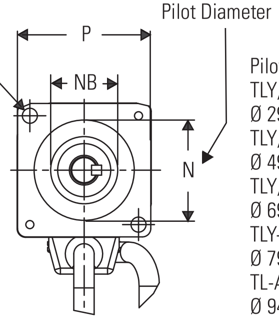
Bulletin TLY-A*xxxxx*-*xx*6*x*AA or TL-A*xxxxx*-B*x*3*x*AA Motor Dimensions

Dimensions are in mm (in.)

Mounting pattern has two holes for A110 through A130 (shown). All others have four holes.







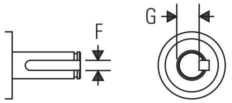
Pilot Diameter Tolerances
TLY/TL-A110, -A120, and -A130 Motors:
Ø 29.979…30.000 (1.1803…1.1811)
TLY/TL-A220 and -A230 Motors:
Ø 49.975…50.000 (1.9675…1.9685)
TLY/TL-A2530 and -A2540 Motors:
Ø 69.970…70.000 (2.7547…2.7556)
TLY-A310 Motors:
Ø 79.970…80.000 (3.1484…3.1496)
TL-A410 Motors:
Ø 94.965…95.000 (3.7388…3.7402)

Optional Shaft Seal
Refer to page 81
for TL-Series motor
shaft seal kit information.

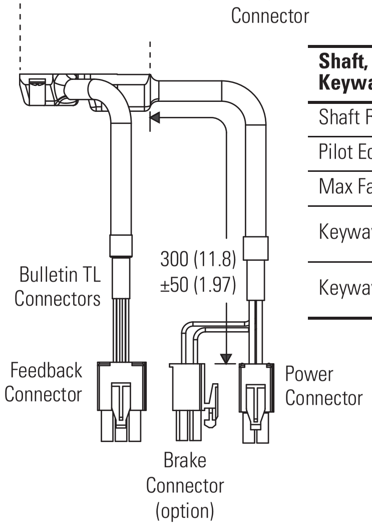
Shaft Diameter Tolerances

TLY/TL-A110, -A120, and -A130 Motors:
Ø 7.991…8.000 (0.3146…0.3150)
TLY/TL-A220 and -A230 Motors:
Ø 11.989…12.000 (0.4720…0.4724)
TLY/TL-A2530 and -A2540 Motors:
Ø 15.989…16.000 (0.6295…0.6299)
TLY-A310 Motors:
Ø 15.989…16.000 (0.6295…0.6299)
TL-A410 Motors:
Ø 21.987…22.000 (0.8656…0.8661)

Shaft Detail with Key



TLY/TL-A110, -A120, and -A130 Motors
3 +0, -0.025 x 3 +0, -0.025 x 15 Key
TLY/TL-A220 and -A230 Motors
4 +0, -0.030 x 4 +0, -0.030 x 15 Key
TLY/TL-A2530 and -A2540 Motors
5 +0, -0.030 x 5 +0, -0.030 x 20 Key
TLY-A310 Motors
5 +0, -0.030 x 5 +0, -0.030 x 20 Key
TL-A410 Motors
8 +0, -0.036 x 7 +0, -0.036 x 25 Key



Shaft, Pilot, and Keyway Tolerances	TLY/TL-A1*xx*	TLY/TL-A2*xx*	TLY/TL-A25*xx*	TLY-A3*xx*	TL-A4*xx*
Shaft Runout (T.I.R.)	0.02 (0.001)	0.02 (0.001)	0.02 (0.001)		0.02 (0.001)
Pilot Eccentricity (T.I.R.)	0.06 (0.0024)	0.06 (0.0024)	0.06 (0.0024)		0.04 (0.0016)
Max Face Runout (T.I.R.)	0.07 (0.003)	0.07 (0.003)	0.07 (0.003)		0.04 (0.0016)
Keyway (G)	6.00…6.20 (0.236…0.244)	9.30…9.50 (0.366…0.374)	12.8…13.0 (0.504…0.512)		17.8…18.0 (0.701…0.709)
Keyway (F)	2.969…2.994 (0.117…0.118)	3.958…3.998 (0.156…0.157)	4.958…4.988 (0.1952…0.1964)		7.949…7.985 (0.313…0.314)

Bulletin TLY-A*xxxxx*-*xx*6*x*AA or TL-A*xxxxx*-B*x*3*x*AA Motor Dimensions

Motor Cat. No.	**AD** mm (in.)	**BE** mm (in.)	**D** mm (in.)	**HD** mm (in.)	**L** (1) mm (in.)	**L-LB** (2) mm (in.)	**LA** mm (in.)	**LB** (1) mm (in.)	**LE** (1) mm (in.)	**M** mm (in.)	**N** mm (in.)	**NB** mm (in.)	**P** mm (in.)	**S** mm (in.)	**T** mm (in.)	**TB** mm (in.)	**G** mm (in.)	**F** mm (in.)
TLY/TL-A110	31.1 (1.22)	21.0 (0.83)	8.0 (0.31)	51.1 (2.01)	78.5 (3.09)	25.0 (0.98)	5.0 (0.20)	53.5 (2.11)	39.1 (1.54)	46.0 (1.81)	30.0 (1.18)	20.0 (0.79)	40.0 (1.57)	4.5 (0.18)	2.5 (0.10)	4.5 (0.18)	6.2 (0.24)	3.0 (0.12)
TLY/TL-A120					84.5 (3.33)			59.5 (2.34)										
TLY/TL-A130					98.5 (3.88)			73.5 (2.89)										
TLY/TL-A220	43.0 (1.69)	27.6 (1.09)	12.0 (0.47)	73.0 (2.87)	106.1 (4.18)	30.0 (1.18)	6.0 (0.24)	76.1 (3.00)	42.8 (1.69)	70.0 (2.76)	50.0 (1.97)	27.0 (1.06)	60.0 (2.36)	5.5 (0.22)	3.0 (0.12)	7.0 (0.28)	9.5 (0.37)	4.0 (0.16)
TLY/TL-A230					128.0 (5.04)			98.1 (3.86)										
TLY/TL-A2530	53.0 (2.09)	27.6 (1.09)	16.0 (0.63)	93.0 (3.66)	134.7 (5.30)	35.0 (1.38)	8.0 (0.32)	99.7 (3.93)	43.8 (1.72)	90.0 (3.54)	70.0 (2.76)	34.0 (1.34)	80.0 (3.15)	6.6 (0.26)	3.0 (0.12)	7.0 (0.28)	13.0 (0.51)	5.0 (0.20)
TLY/TL-A2540					143.7 (5.66)			108.7 (4.28)										
TLY-A310	56.0 (2.20)			99.0 (3.90)	179.2 (7.06)			144.2 (5.68)	57.1 (2.24)	100.0 (3.94)	80.0 (3.15)		86.0 (3.39)					
TL-A410	67.0 (2.64)	38.4 (1.51)	22.0 (0.87)	117.0 (4.61)	216.0 (8.50)	40.0 (1.57)	17.0 (0.67)	176.0 (6.93)	102.0 (4.02)	115.0 (4.53)	95.0 (3.74)	N/A	100.0 (3.94)	9.0 (0.35)	7.0 (0.28)	N/A	18.0 (0.71)	8.0 (0.32)

(1) If ordering an TLY/TL-A110, TLY/TL-A120 or TLY/TL-A130 motor with brake, add 35.6 mm (1.40 in.) to dimensions L, LB, and LE.
If ordering an TLY/TL-A220 or TLY/TL-A230 motor with brake, add 34.6 mm (1.36 in.) to dimensions L, LB, and LE.
If ordering an TLY/TL-A2530 or TLY/TL-A2540 motor with brake, add 36.6 mm (1.44 in.) to dimensions L, LB, and LE.
If ordering an TLY-A310 motor with brake, add 23.0 mm (0.90 in.) to dimensions L, LB, and LE.
If ordering an TL-A410 motor with brake, add 32.0 mm (1.26 in.) to dimensions L, LB, and LE.

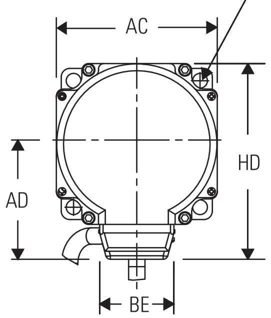
(2) Tolerance for this dimension is ±1.0 mm (±0.039 in.).

Motors are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

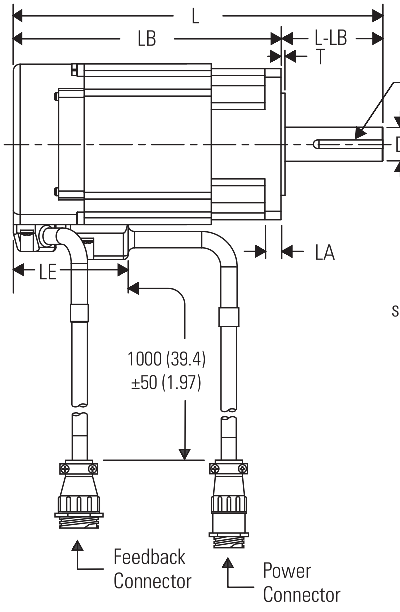
TL-Series (Bulletin TLY) NEMA Motor Dimensions

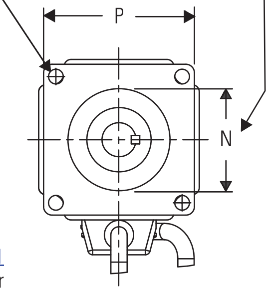
Bulletin TLY-A*xxxxx*-H*x*6*x*AN NEMA Motor Dimensions

Dimensions are in mm (in.)



Mounting pattern has two holes for A120 through A130 (shown). All others have four holes.

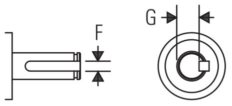




Pilot Diameter Tolerances
TLY-A120 and -A130 Motors:
Ø 21.979…22.000 (0.8653…0.8661)
TLY-A220 and -A230 Motors:
Ø 38.075…38.100 (1.4990…1.5000)
TLY-A2530 and -A2540 Motors:
Ø 72.99…73.02 (2.874…2.875)

Optional Shaft Seal
Refer to page 81 for TL-Series motor shaft seal kit information.

Shaft Detail with Key



TLY-A220 and -A230
0.125 +0, -0.002 x 0.125 +0, -0.002 x 0.9375 Key
TLY-A2530 and -A2540
0.187 +0, -0.002 x 0.187 +0, -0.002 x 1.156 Key

Shaft Diameter Tolerances

TLY-A120 and -A130 Motors:
Ø 6.341…6.350 (0.2496…0.2500)
TLY-A220 and -A230 Motors:
Ø 38.075…38.100 (0.4990…0.5000)

TLY-A2530 and -A2540 Motors:
Ø 15.864…15.875 (0.6246…0.6250)

Shaft, Pilot, and Keyway Tolerances	TLY-A1*xx*	TLY-A2*xx*	TLY-A25*xx*
Shaft Runout (T.I.R.)	0.02 (0.001)	0.02 (0.001)	0.02 (0.001)
Pilot Eccentricity (T.I.R.)	0.06 (0.0024)	0.06 (0.0024)	0.06 (0.0024)
Max Face Runout (T.I.R.)	0.07 (0.003)	0.07 (0.003)	0.07 (0.003)
Keyway (G)	N/A	10.54…10.92 (0.415…0.430)	12.75…13.13 (0.502…0.517)
Keyway (F)	N/A	3.124…3.175 (0.123…0.125)	4.763…4.814 (0.1875…0.1895)

Bulletin TLY-A*xxxxx*-H*x*6*x*AN NEMA Motor Dimensions

Motor Cat. No.	AC mm (in.)	AD mm (in.)	BE mm (in.)	D mm (in.)	HD mm (in.)	L (1) mm (in.)	L-LB (2) mm (in.)	LA mm (in.)	LB (1) mm (in.)	LE (1) mm (in.)	M mm (in.)	N mm (in.)	P mm (in.)	S mm (in.)	T mm (in.)	G mm (in.)	F mm (in.)
TLY-A120	N/A	31.10 (1.22)	21.0 (0.83)	6.35 (0.25)	52.0 (2.05)	91.5 (3.603)	27.0 (1.06)	5.0 (0.20)	64.5 (2.54)	39.1 (1.54)	43.8 (1.725)	22.0 (0.86)	42.0 (1.65)	8-32 Thread	2.0 (0.08)	N/A	N/A
TLY-A130						105.5 (4.153)			78.5 (3.09)								
TLY-A220	60 (2.36)	43.0 (1.69)	27.6 (1.09)	12.70 (0.50)	73.0 (2.87)	137.9 (5.43)	38.1 (1.50)	6.0 (0.24)	99.8 (3.93)	43.3 (1.70)	66.7 (2.625)	38.1 (1.50)	56.4 (2.22)	5.5 (0.217)	1.5 (0.06)	10.92 (0.43)	3.175 (0.125)
TLY-A230						159.9 (6.30)			121.8 (4.80)								
TLY-A2530	N/A	53.0 (2.09)		15.875 (0.625)	96.0 (3.78)	149.2 (5.872)	44.5 (1.752)	8.0 (0.32)	104.7 (4.12)	43.8 (1.72)	98.4 (3.875)	73.02 (2.87)	86.0 (3.39)	5.5 (0.217)	1.5 (0.06)	13.13 (0.517)	4.814 (0.189)
TLY-A2540						158.2 (6.205)			113.7 (4.48)								

(1) If ordering an TLY-A120 or TLY-A130 motor with brake, add 35.6 mm (1.40 in.) to dimensions L, LB, and LE.
If ordering an TLY-A220 or TLY-A230 motor with brake, add 34.6 mm (1.36 in.) to dimensions L, LB, and LE.
If ordering an TLY-A2530 or TLY-A2540 motor with brake, add 36.6 mm (1.44 in.) to dimensions L, LB, and LE.

(2) Tolerance for this dimension is ±1.0 mm (±0.039 in.).

NEMA motor flanges and shafts are designed to inch dimensions. Other frame areas are designed to metric dimensions. Conversions are approximate values.

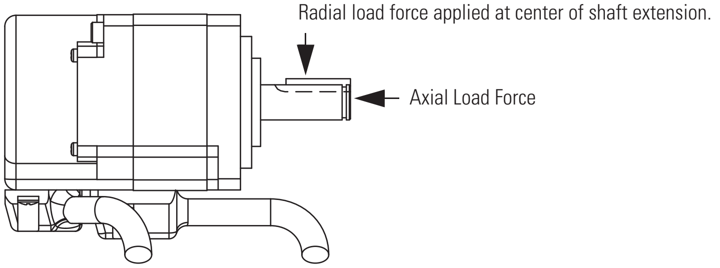
TL-Series Motor Load Force Ratings

TL-Series motors are capable of operating with the maximum radial or maximum axial shaft loads listed in the following tables. Radial loads listed are applied in the middle of the shaft extension. The tables below represent an L_{10} bearing fatigue life of 20,000 hours. This 20,000-hour life does not account for possible application-specific life reduction that may occur due to bearing grease contamination from external sources. Maximum operating speed is limited by motor winding.

Radial Load Force Ratings

Motor Cat. No.	1000 rpm kg (lb)	2000 rpm kg (lb)	3000 rpm kg (lb)	4000 rpm kg (lb)	4500 rpm kg (lb)	5000 rpm kg (lb)	6000 rpm kg (lb)
TLY/TL-A110	11 (24)	9 (19)	7 (16)	7 (16)	–	6 (13)	6 (13)
TLY/TL-A120	12 (26)	10 (21)	8 (18)	7 (16)	–	7 (15)	6 (13)
TLY/TL-A130	13 (29)	10 (23)	9 (20)	8 (18)	–	8 (17)	7 (15)
TLY/TL-A220	27 (60)	22 (48)	19 (42)	17 (37)	–	16 (35)	15 (33)
TLY/TL-A230	31 (68)	24 (54)	21 (47)	19 (42)	–	18 (40)	17 (37)
TLY/TL-A2530	48 (106)	38 (84)	34 (74)	–	–	28 (62)	–
TLY/TL-A2540	50 (110)	39 (87)	34 (76)	–	–	29 (64)	–
TLY-A310	80 (177)	63 (139)	56 (123)	–	48 (106)	–	–
TL-A410	76 (168)	60 (133)	53 (117)	–	44 (98)	–	–

Bulletin TLY-A*xxxx* or TL-A*xxxx* Motor Load Forces



Axial Load Force Ratings (maximum radial load)

Motor Cat. No.	1000 rpm kg (lb)	2000 rpm kg (lb)	3000 rpm kg (lb)	4000 rpm kg (lb)	4500 rpm kg (lb)	5000 rpm kg (lb)	6000 rpm kg (lb)
TLY/TL-A110	8 (18)	6 (13)	5 (11)	4 (9)	–	4 (9)	3 (7)
TLY/TL-A120	9 (20)	7 (16)	5 (11)	4 (9)	–	4 (9)	3 (7)
TLY/TL-A130	10 (22)	8 (17)	6 (13)	5 (12)	–	5 (11)	4 (9)
TLY/TL-A220	15 (33)	11 (24)	9 (20)	8 (17)	–	7 (16)	5 (11)
TLY/TL-A230	15 (33)	12 (26)	10 (21)	9 (20)	–	8 (17)	6 (13)
TLY/TL-A2530	18 (39)	13 (29)	11 (24)	–	–	9 (19)	–
TLY/TL-A2540	18 (39)	13 (29)	11 (25)	–	–	9 (20)	–
TLY-A310	19 (42)	14 (31)	11 (25)	–	10 (21)	–	–
TL-A410	29 (64)	21 (47)	18 (40)	–	14 (31)	–	–

Axial Load Force Ratings (zero radial load)

Motor Cat. No.	1000 rpm kg (lb)	2000 rpm kg (lb)	3000 rpm kg (lb)	4000 rpm kg (lb)	4500 rpm kg (lb)	5000 rpm kg (lb)	6000 rpm kg (lb)
TLY/TL-A110	12 (26)	9 (20)	7 (16)	6 (13)	–	6 (13)	5 (11)
TLY/TL-A120	12 (26)	9 (20)	7 (16)	6 (13)	–	6 (13)	5 (11)
TLY/TL-A130	12 (26)	9 (20)	7 (16)	6 (13)	–	6 (13)	5 (11)
TLY/TL-A220	19 (41)	14 (30)	11 (25)	10 (21)	–	9 (20)	8 (17)
TLY/TL-A230	19 (41)	14 (30)	11 (25)	10 (21)	–	9 (20)	8 (17)
TLY/TL-A2530	23 (50)	17 (37)	14 (31)	–	–	11 (25)	–
TLY/TL-A2540	23 (50)	17 (37)	14 (31)	–	–	11 (25)	–
TLY-A310	26 (57)	19 (42)	16 (35)	–	14 (31)	–	–
TL-A410	34 (75)	25 (55)	21 (47)	–	17 (37)	–	–

TL-Series Motor Catalog Number

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your motor. For questions regarding product availability, contact your Allen-Bradley distributor.

TL *x* - A *xx* *xx* *x* -*x* *x* *x* *x* A*x*

Factory Options
AA = Standard metric (Bulletin TLY or TL)
AN = NEMA (Bulletin TLY)

Brake
2 = No brake
4 = 24V DC brake

Connectors
3 = Cables with rectangular connectors, 300 mm (11.8 in.) for Bulletin TL
6 = Cables with circular connectors, 1 m (39.4 in.) for Bulletin TLY

Enclosure/Shaft Key/Shaft Seal
J = IP65 housing/Shaft key/No shaft seal
K = IP65 housing/No shaft key/No shaft seal

Feedback
B = Multi-turn (1), 17-bit (131,072) counts/revolution, absolute, battery-backed encoder (serial protocol)
H = 2000 lines/revolution, incremental encoder

Rated Speed (2)
M = 4500 rpm
P = 5000 rpm
T = 6000 rpm (Bulletin TLY only)

Magnet Stack Length

Frame Size (2)
1 = 46 mm (1.81 in.) or NEMA 173 = 100 mm (3.94 in.) Bulletin TLY only
2 = 70 mm (2.75 in.) or NEMA 234 = 115 mm (4.52 in.) Bulletin TL only
25 = 90 mm (3.54 in.) or NEMA 34

Voltage Class
A = 200V

Series Type
Y = (Bulletin TLY)
Blank = (Bulletin TL)

Series
TL = TL-Series low-inertia motors

(1) Single-turn if used without battery backup.
(2) Not all combinations are available. Only the configurations for rated speed as listed in TL-Series Motor Performance Specifications on page 82 are available.

For TL-Series connector kit catalog numbers, refer to 2090-Series Motor-end Cable Connector Kits on page 422.

General Motor Performance Definitions

This section contains a list of general motor performance definitions.

Rated Speed - The operating speed of the drive and motor combination at which approximately 70% of continuous rated torque (To) can be developed. This point is defined with the motor at 25 °C (77 °F).

Rated Operation Area - The boundary of the torque-speed curve where the motor and controller combination may operate on a servo basis without exceeding the RMS rating of either.

Intermittent Operation Area - The boundary of the torque-speed curve where the motor and controller combination may operate in acceleration/deceleration mode without exceeding peak rating of either, provided that the duty cycle RMS continuous torque limit is not exceeded.

$$\mathrm{RMSTorque} = \sqrt{\frac{(Tpa^2)(t_1) + (Tss^2)(t_2) + (Tpd^2)(t_3) + (Tr^2)(t_4)}{t_1 + t_2 + t_3 + t_4}}$$

Continuous Current - The rated current of a motor with windings at a rated temperature and an ambient temperature of 40 °C (104 °F).

Peak Current - The amount of current that can be applied to the motor without causing damage to the motor.

Mechanical Time Constant - The time required for the motor to reach 63% of its final speed when a step voltage is applied.

Electrical Time Constant - The time required for the motor to reach 63% of its rated current.

Maximum Ambient Temperature - The maximum environmental temperature in which the motor can be operated at rated loads without exceeding its insulation-type temperature rise limits.

Insulation Class - The designation of the operating temperature limits for motor insulation materials.

Thermal Time Constant - The time required for the motor windings to reach 63% of continuous temperature rise with constant watts loss.

Torque Constant - The amount of torque developed for one ampere of motor current at the stated motor temperature.

Voltage Constant - The value of the generated voltage at a specified speed when the rotor is moved mechanically in the magnetic field.

Terminal Resistance - The winding resistance.

Inductance - The winding inductance measured by a step input of zero impedance voltage applied to the locked rotor.

Rotor Polar Moment of Inertia - The moment of inertia about the axis of rotation.

Motor Weight - The approximate weight of the complete motor (including brake, if supplied) less the weight of options.

Balance - The compensation of rotor weight distribution to reduce vibrational resonance. Motors are factory balanced under running speeds.

Linear Motion

Use this chapter to become familiar with the Kinetix Motion Control linear motors and actuators and select the linear motion components required for your application. To compare features from one family of linear motion components to another, refer to Linear Servo Motors on page 11 and Linear Actuators on page 12.

Common Linear Motion Specifications

These linear motion specifications are common to Kinetix Motion Control linear motor/actuator families.

Environmental Specifications

Attribute	MP-Series Integrated Linear Stages		MP-Series and TL-Series Electric Cylinders			LDC-Series and LDL-Series Linear Motors
	Bulletin MPAS	Bulletin MPMA	Bulletin MPAR	Bulletin TLAR	Bulletin MPAI	
Temperature, ambient	0…40 °C (32…104 °F)					
Temperature, storage	-30…70 °C (-22…158 °F)		-25…60 °C (-13…140 °F)			-30…70 °C (-22…158 °F)
Relative humidity (noncondensing)	5…95%					
Shock	20 g peak, 6 ms duration					
Vibration	0.1 grms @ 30…2000 Hz		2.5 g peak @ 30…2000 Hz			

Environmental Ratings

IP Rating	Dust Protection	Liquid Protection	Actuator/Motor
IP30	Objects larger than 2.5 mm (0.098 in.)	No protection from liquids.	Bulletin MPAS and MPMA
IP40	Objects larger than 1.0 mm (0.039 in.)	No protection from liquids.	Bulletin MPAR and TLAR [1]
IP65	Total protection from dust	Protected against low-pressure jets of water from all directions.	LDC-Series and LDL-Series
IP66		Protected against strong jets of water.	Bulletin MPAR [2] [3]
IP67		Protected against the effects of temporary liquid immersion.	Bulletin MPAI [3]

(1) Applies to complete unit, including rod-end seal and breather port.

(2) Applies to electronic components.

(3) Requires use of Bulletin 2090 environmentally sealed cable connectors.

Motor Brake Application Guidelines

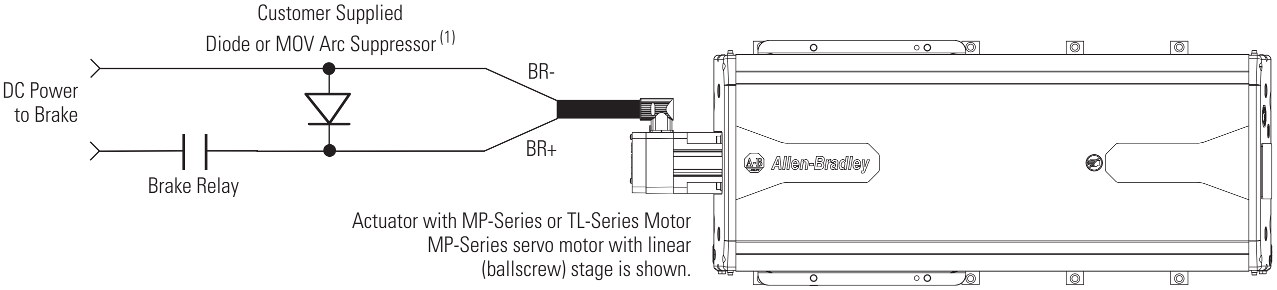
The brakes offered as options on the ballscrew driven linear stages and electric cylinders are holding brakes designed to hold the carriage/rod cylinder in position up to the rated brake holding torque. The brakes release when voltage is applied to the brake coil. Voltage and polarity supplied to the brake must be as specified to be sure of proper brake performance.

The brakes are not designed for stopping an actuator in motion. Servo drive inputs should be used to stop carriage/rod cylinder motion before the brake is activated. The recommended method of stopping motion is to command the servo drive to decelerate the carriage/rod cylinder to a complete stop and engage the brake only after the carriage/rod cylinder has stopped.

If system main power fails, the brakes can withstand use as stopping brakes. However, use of the brakes as stopping brakes can create mechanical backlash that is potentially damaging to the system, increases brake pad wear and reduces brake life. The brakes are not designed nor are they intended to be used as a safety device.

A separate power source is required to disengage the brake. This power source can be controlled by the linear stage controls, in addition to manual operator controls. Electrical arcing may occur at the relay contacts until the brake power dissipates. A customer supplied diode or metal oxide varistor (MOV) is recommended to prevent arcing. Use of an MOV can also reduce the time to mechanically engage the brake.

Example Suppression Device for Brake Relay Contacts



(1) Kinetix 2000, Kinetix 6000, Kinetix 6200, Kinetix 6500, and Kinetix 7000 servo drives provide motor/actuator brake relay outputs and supply an MOV arc suppressor, so customer-supplied arc suppressor is not required unless the coil current of the brake is greater than the maximum brake current rating of the drive relay output.

MP-Series Integrated Linear Stages

MP-Series integrated linear stages extend the performance and reliability of MP-Series servo motor technology to ballscrew and direct drive linear slide-type actuators. MP-Series linear stages are specifically designed for abusive, high throughput industrial automation applications instead of clean, high accuracy lab environments. Wear items such as cable track and strip seals can be changed in minutes to minimize machine down-time. Because the MP-Series linear stages are provided as fully integrated motor/actuators supported by Motion Analyzer and RSLogix 5000 software, the time required for mechanical design, installation, setup, and programming is dramatically reduced.

For drive compatibility, refer to Servo Drives on page 14.

MP-Series Direct-drive Linear Stage Features

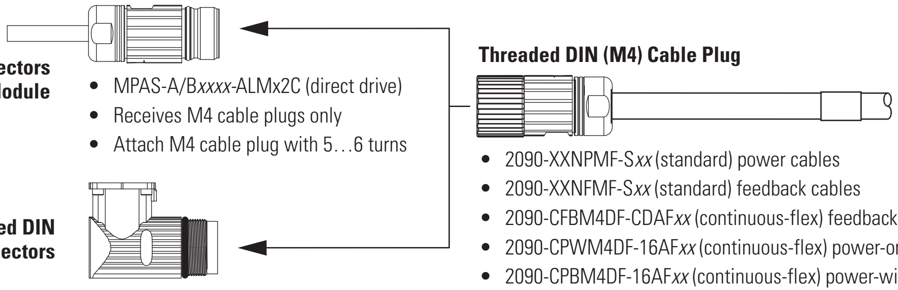
- High performance linear motor, linear guide bearings, and 5 micron resolution linear encoder integrated into single, compact package
- Extremely high speeds up to 5000 mm/s (197 in./s)
- Very long life due to elimination of mechanical transmission
- Quick-change cable track module to minimize downtime
- Available with and without cover/strip seal option
- Peak forces to 601 N (135 lb) and peak velocities to 5000 mm/s (197 in./s)
- Stroke lengths from 120…1940 mm (4.7…76.4 in.)

MP-Series Ballscrew-drive Linear Stage Features

- High performance MP-Series servo motor, ballscrew, and linear guide bearings integrated in to single, compact package
- Peak forces to 1212 N (273 lb) and peak velocities to 1124 mm/s (44 in./s)
- Stroke lengths from 120…1020 mm (4.7…40.2 in.)

Motor Connector/Cable Compatibility

MP-Series (Bulletin MPAS) linear stages are equipped with threaded DIN connectors.



Threaded DIN Connectors From Cable Track Module

- MPAS-A/B*xxxx*-ALMx2C (direct drive)
- Receives M4 cable plugs only
- Attach M4 cable plug with 5…6 turns

Threaded DIN Connectors

- MPAS-A/B*xxxx*1/2-V*xx*S*x*A (ballscrew)
- Receives M4 cable plugs only
- Attach M4 cable plug with 5…6 turns

Threaded DIN (M4) Cable Plug

- 2090-XXNPMF-S*xx* (standard) power cables
- 2090-XXNFMF-S*xx* (standard) feedback cables
- 2090-CFBM4DF-CDAF*xx* (continuous-flex) feedback cables
- 2090-CPWM4DF-16AF*xx* (continuous-flex) power-only cables
- 2090-CPBM4DF-16AF*xx* (continuous-flex) power-with-brake cables

MP-Series Integrated Linear Stage Accessory Kits

- Cable track module replacement kits
- Strip seal replacement kits
- Top cover kits
- Side cover kits
- Coupling kits
- Tee-nut kit (package of 10)
- Toe-clip kit (package of 10)
- Grease gun kit
- Grease replacement cartridge

Accessory Kits Common to All Single-axis Linear Stages

Linear Stage Cat. No.	Description	Accessory Cat. No.
MPAS-A/B*xxxx*	Kit, grease gun for all integrated linear stages.	MPAS-GPUMP
	Grease gun refill cartridge for all integrated linear stages.	MPAS-CART
	Kit, toe clamps (10 per package) for all integrated linear stages.	MPAS-TOE
MPAS-A6*xxx*, MPAS-B6*xxx*	Kit, Tee-nuts (10 per package).	MPAS-6-TNUT
MPAS-A8*xxx*, MPAS-B8*xxx*		MPAS-8-TNUT
MPAS-A9*xxx*, MPAS-B9*xxx*		MPAS-9-TNUT

Accessory Kits for Single-axis Direct-drive Linear Stages

Linear Stage Cat. No.	Description	Accessory Cat. No.
MPAS-*x*6*xxxx*-ALM*x*2A	Cable track module for direct-drive linear stage.	MPAS-6*xxx*B-CABLE (1)
MPAS-*x*8*xxxx*-ALM*x*2A		MPAS-8*xxx*E-CABLE (2)
MPAS-*x*9*xxxx*-ALM*x*2A		MPAS-9*xxx*K-CABLE (2)
MPAS-*x*6*xxxx*-ALMS2A	Kit, strip seal for direct-drive linear stage.	MPAS-6*xxx*B-SEAL (1)
MPAS-*x*8*xxxx*-ALMS2A		MPAS-8*xxx*E-SEAL (2)
MPAS-*x*9*xxxx*-ALMS2A		MPAS-9*xxx*K-SEAL (2)
MPAS-*x*6*xxxx*-ALMS2A	Kit, side covers for direct-drive linear stage.	MPAS-6*xxx*B-SIDE (1)
MPAS-*x*8*xxxx*-ALMS2A		MPAS-8*xxx*E-SIDE (2)
MPAS-*x*9*xxxx*-ALMS2A		MPAS-9*xxx*K-SIDE (2)
MPAS-*x*6*xxxx*-ALMS2A	Kit, top cover for direct-drive linear stage.	MPAS-6*xxx*B-TOP (1)
MPAS-*x*8*xxxx*-ALMS2A		MPAS-8*xxx*E-TOP (2)
MPAS-*x*9*xxxx*-ALMS2A		MPAS-9*xxx*K-TOP (2)

(1) Variable *xxx* (mm stroke x 10) is 012, 018, 024, 030, 036, 042, 054, 066, 078, 090, 102, or 114.

(2) Variable *xxx* (mm stroke x 10) is 014, 020, 026, 032, 038, 044, 056, 068, 080, 092, 104, 128, 152, 176, or 194.

Accessory Kits for Single-axis Ballscrew Linear Stages

Linear Stage Cat. No.	Description	Accessory Cat. No.
MPAS-*x*6*xxxx*-V*xx*S*x*A	Coupler, for ballscrew linear stage.	MPAS-6-COUP
MPAS-*x*8*xxxx*-V*xx*S*x*A		MPAS-8-COUP
MPAS-*x*9*xxxx*-V*xx*S*x*A		MPAS-9-COUP
MPAS-A*xxxx*1-V05S2A	Motor, 230V, without brake for 5 mm/rev ballscrew linear stage.	MPLS-A210E-VJ42AA
MPAS-A*xxxx*1-V05S4A	Motor, 230V, with brake for 5 mm/rev ballscrew linear stage.	MPLS-A210E-VJ44AA
MPAS-B*xxxx*1-V05S2A	Motor, 460V, without brake for 5 mm/rev ballscrew linear stage.	MPLS-B210E-VJ42AA
MPAS-B*xxxx*1-V05S4A	Motor, 460V, with brake for 5 mm/rev ballscrew linear stage.	MPLS-B210E-VJ44AA
MPAS-A*xxxx*2-V20S2A	Motor, 230V, without brake for 20 mm/rev ballscrew linear stage.	MPLS-A220H-VJ42AA
MPAS-A*xxxx*2-V20S4A	Motor, 230V with brake for 20 mm/rev ballscrew linear stage.	MPLS-A220H-VJ44AA
MPAS-B*xxxx*2-V20S2A	Motor, 460V, without brake for 20 mm/rev ballscrew linear stage.	MPLS-B220H-VJ42AA
MPAS-B*xxxx*2-V20S4A	Motor, 460V with brake for 20 mm/rev ballscrew linear stage.	MPLS-B220H-VJ44AA
MPAS-*x*6*xxxx*-V*xx*S*x*A	Kit, strip seal for ballscrew linear stage.	MPAS-6*xxx*1-SEAL (1)
MPAS-*x*8*xxxx*-V*xx*S*x*A		MPAS-8*xxx*1-SEAL (2)
MPAS-*x*9*xxxx*-V*xx*S*x*A		MPAS-9*xxx*1-SEAL (2)
MPAS-*x*6*xxxx*-V*xx*S*x*A	Kit, side cover for ballscrew linear stage.	MPAS-6*xxx*1-SIDE (1)
MPAS-*x*8*xxxx*-V*xx*S*x*A		MPAS-8*xxx*1-SIDE (2)
MPAS-*x*9*xxxx*-V*xx*S*x*A		MPAS-9*xxx*1-SIDE (2)
MPAS-*x*6*xxxx*-V*xx*S*x*A	Kit, top cover for ballscrew linear stage.	MPAS-6*xxx*1-TOP (1)
MPAS-*x*8*xxxx*-V*xx*S*x*A		MPAS-8*xxx*1-TOP (2)
MPAS-*x*9*xxxx*-V*xx*S*x*A		MPAS-9*xxx*1-TOP (2)

(1) Variable *xxx* (mm stroke x 10) is 012, 018, 024, 030, 036, 042, 054, or 066.

(2) Variable *xxx* (mm stroke x 10) is 012, 018, 024, 030, 036, 042, 054, 066, 078, 090, or 102.

MP-Series Integrated Linear Stage Life Specifications

Attribute	Value
Linear bearings	10,000 km (6213.7 mi) or one year minimum life with a maximum load of 22.7 kg (50 lb) centered on lubrication intervals every 5000 km (3106.8 mi) or three months.
Ballscrew	300,000 krevs or one year minimum life with a maximum load of 11.3 kg (25 lb), 1 g maximum acceleration in a clean dry, noncontaminating environment with lubrication every 150,000 krevs or three months.
Strip seal	10,000,000 cycles or 10,000 km min life in clean, dry, noncontaminating environment.
Cable track module	10,000,000 cycles minimum life.
Mechanical stop bumper	56.49 Nm (500 lb-in) potential energy.

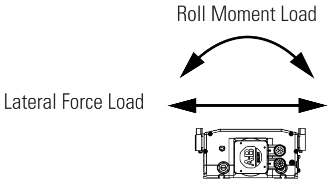
MP-Series Integrated Linear Stage Accuracy Specifications

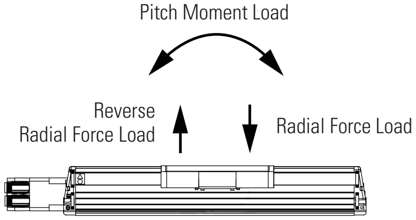
Linear Stage Cat. No.	Drive Mechanism and Feedback Type	Bi-directional Repeatability (1)	Accuracy (1)	Straightness	Flatness
MPAS-A/B6*xxx*1-V05S*x*A MPAS-A/B6*xxx*2-V20S*x*A MPAS-A/B8*xxx*1-V05S*x*A MPAS-A/B8*xxx*2-V20S*x*A MPAS-A/B9*xxx*1-V05S*x*A MPAS-A/B9*xxx*2-V20S*x*A	Ballscrew with absolute multi-turn rotary encoder	±60 µm (±0.002 in.)	60 µm + 167 µm/m (0.002 in. + 0.002 in./ft)	10 µm + 50 µm/m	10 µm + 50 µm/m
MPAS-A6*xxx*B-ALM*x*2C MPAS-A8*xxx*E-ALM*x*2C MPAS-B8*xxx*F-ALM*x*2C MPAS-A9*xxx*K-ALM*x*2C MPAS-B9*xxx*L-ALM*x*2C	Linear Motor with incremental linear encoder	±15 µm (±0.0005 in.)	100 µm + 20 µm/m (0.004 in. + 0.0002 in./ft)		

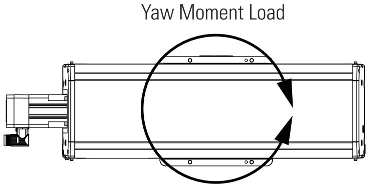
(1) Measured at 20 °C (68 °F).

MP-Series Integrated Linear Stage Load Force Ratings

The static moment and force ratings shown in the performance specifications tables are the maximum permissible values possible before permanent damage to the linear stage can occur. To determine the estimated L_{10} bearing and ballscrew life of MP-Series Integrated Linear Stages, use Motion Analyzer software, version 4.4 or later.







MP-Series Integrated Linear Stage (230V) Performance Specifications

MP-Series Integrated Linear Stage Performance Specifications (150 mm frame size)

Linear Stage Cat. No. MPAS-	Screw Lead mm (in.)	Stroke Length mm (in.)	Velocity, max mm/s (in./s)	Cont. Stall Force N (lb)	Peak Stall Force N (lb)	Max Static Loads (1)			Max Static Moment Loads (1)		
						Radial N (lb)	Reverse Radial N (lb)	Lateral N (lb)	Pitch Nm (lb-ft)	Yaw Nm (lb-ft)	Roll Nm (lb-ft)
A6*xxx*1-V05S*x*A	5.0 (0.2)	120…660 (4.7…26.0)	200 (7.9)	521 (117)	1212 (272)	5506 (1238)	2367 (532)	2753 (619)	195 (144)	227 (167)	92 (68)
A6*xxx*2-V20S*x*A	20.0 (0.8)		1124 (44.3)	462 (104)	968 (218)						
A6*xxx*B-ALMO2C (no cover)	N/A	120…1140 (4.7…44.9)	5000 (196.9)	105 (24)	359 (81)						
A6*xxx*B-ALMS2C (covered)	N/A			83 (19)	312 (70)						

(1) Values apply to bearing ratings only. Contact your Rockwell Automation sales representative for structural considerations.

MP-Series Integrated Linear Stage Performance Specifications (200 mm frame size)

Linear Stage Cat. No. MPAS-	**Screw Lead** mm (in.)	**Stroke Length** mm (in.)	**Velocity, max** mm/s (in./s)	**Cont. Stall Force** N (lb)	**Peak Stall Force** N (lb)	**Max Static Loads** [1]			**Max Static Moment Loads** [1]		
						Radial N (lb)	**Reverse Radial** N (lb)	**Lateral** N (lb)	**Pitch** Nm (lb-ft)	**Yaw** Nm (lb-ft)	**Roll** Nm (lb-ft)
A8*xxx*1-V05S*x*A	5.0 (0.2)	120…780 (4.7…30.7)	200 (7.9)	521 (117)	1212 (273)	9365 (2105)	4027 (905)	4683 (1053)	336 (248)	391 (288)	258 (190)
		900 (35.4)	176 (6.9)								
		1020 (40.1)	143 (5.6)								
A8*xxx*2-V20S*x*A	20.0 (0.8)	120…660 (4.7…26.0)	1124 (44.3)	462 (104)	968 (218)						
		780 (30.7)	889 (35.0)								
		900 (35.4)	715 (28.1)								
		1020 (40.1)	582 (22.9)								
A8*xxx*E-ALMO2C (no cover)	N/A	140…1940 (5.5…76.4)	5000 (196.9)	189 (43)	456 (103)	9100 (2046)	4100 (922)	4700 (1057)	402 (297)	492 (363)	245 (181)
A8*xxx*E-ALMS2C (covered)	N/A			159 (36)	399 (90)						

(1) Values apply to bearing ratings only. Contact your Rockwell Automation sales representative for structural considerations.

MP-Series Integrated Linear Stage Performance Specifications (250 mm frame size)

Linear Stage Cat. No. MPAS-	**Screw Lead** mm (in.)	**Stroke Length** mm (in.)	**Velocity, max** mm/s (in./s)	**Cont. Stall Force** N (lb)	**Peak Stall Force** N (lb)	**Max Static Loads** [1]			**Max Static Moment Loads** [1]		
						Radial N (lb)	**Reverse Radial** N (lb)	**Lateral** N (lb)	**Pitch** Nm (lb-ft)	**Yaw** Nm (lb-ft)	**Roll** Nm (lb-ft)
A9*xxx*1-V05S*x*A	5.0 (0.2)	120…780 (4.7…30.7)	200 (7.9)	521 (117)	1212 (273)	13,282 (2986)	5711 (1284)	6641 (1493)	477 (352)	555 (409)	468 (345)
		900 (35.4)	176 (6.9)								
		1020 (40.1)	143 (5.6)								
A9*xxx*2-V20S*x*A	20.0 (0.8)	120…660 (4.7…26.0)	1124 (44.3)	462 (104)	968 (218)						
		780 (30.7)	889 (35.0)								
		900 (35.4)	715 (28.1)								
		1020 (40.1)	582 (22.9)								
A9*xxx*K-ALMO2C (no cover)	N/A	140…1940 (5.5…76.4)	5000 (196.9)	285 (64)	680 (153)	12900 (2900)	5800 (1304)	6600 (1484)	582 (429)	714 (527)	444 (328)
A9*xxx*K-ALMS2C (covered)	N/A			245 (55)	601 (135)						

(1) Values apply to bearing ratings only. Contact your Rockwell Automation sales representative for structural considerations.

System Combinations

For MP-Series Integrated Linear Stages and	Refer to
Kinetix 6000 (230V) drives	page 667
Kinetix 2000 (230V) drives	page 696
Kinetix 300 (240V) drives	page 716
Kinetix 3 (240V) drives	page 729
Ultra3000 (230V) drives	page 742

MP-Series Integrated Linear Stage (460V) Performance Specifications

MP-Series Integrated Linear Stage Performance Specifications (150 mm frame size)

Linear Stage Cat. No. MPAS-	**Screw Lead** mm (in.)	**Stroke Length** mm (in.)	**Velocity, max** mm/s (in./s)	**Cont. Stall Force** N (lb)	**Peak Stall Force** N (lb)	**Max Static Loads**			**Max Static Moment Loads**		
						Radial N (lb)	**Reverse Radial** N (lb)	**Lateral** N (lb)	**Pitch** Nm (lb-ft)	**Yaw** Nm (lb-ft)	**Roll** Nm (lb-ft)
B6*xxx*1-V05S*x*A	5.0 (0.2)	120…660 (4.7…26.0)	200 (7.9)	521 (117)	1212 (273)	5506 (1238)	2367 (532)	2753 (619)	195 (144)	227 (167)	92 (68)
B6*xxx*2-V20S*x*A	20.0 (0.8)		1124 (44.3)	462 (104)	968 (218)						

MP-Series Integrated Linear Stage Performance Specifications (200 mm frame size)

Linear Stage Cat. No. MPAS-	**Screw Lead** mm (in.)	**Stroke Length** mm (in.)	**Velocity, max** mm/s (in./s)	**Cont. Stall Force** N (lb)	**Peak Stall Force** N (lb)	**Max Static Loads**			**Max Static Moment Loads**		
						Radial N (lb)	**Reverse Radial** N (lb)	**Lateral** N (lb)	**Pitch** Nm (lb-ft)	**Yaw** Nm (lb-ft)	**Roll** Nm (lb-ft)
B8*xxx*1-V05S*x*A	5.0 (0.2)	120…780 (4.7…30.7)	200 (7.9)	521 (117)	1212 (273)	9365 (2105)	4027 (905)	4683 (1053)	336 (248)	391 (288)	258 (190)
		900 (35.4)	176 (6.9)								
		1020 (40.1)	143 (5.6)								
B8*xxx*2-V20S*x*A	20.0 (0.8)	120…660 (4.7…26.0)	1124 (44.3)	462 (104)	968 (218)						
		780 (30.7)	889 (35.0)								
		900 (35.4)	715 (28.1)								
		1020 (40.1)	582 (22.9)								
B8*xxx*F-ALMO2C (no cover)	N/A	140…1940 (5.5…76.4)	5000 (196.9)	189 (43)	456 (103)	9100 (2046)	4100 (922)	4700 (1057)	402 (297)	492 (363)	245 (181)
B8*xxx*F-ALMS2C (covered)	N/A			159 (36)	399 (90)						

MP-Series Integrated Linear Stage Performance Specifications (250 mm frame size)

Linear Stage Cat. No. MPAS-	**Screw Lead** mm (in.)	**Stroke Length** mm (in.)	**Velocity, max** mm/s (in./s)	**Cont. Stall Force** N (lb)	**Peak Stall Force** N (lb)	**Max Static Loads**			**Max Static Moment Loads**		
						Radial N (lb)	**Reverse Radial** N (lb)	**Lateral** N (lb)	**Pitch** Nm (lb-ft)	**Yaw** Nm (lb-ft)	**Roll** Nm (lb-ft)
B9*xxx*1-V05S*x*A	5.0 (0.2)	120…780 (4.7…30.7)	200 (7.9)	521 (117)	1212 (273)	13,282 (2986)	5711 (1284)	6641 (1493)	477 (352)	555 (409)	468 (345)
		900 (35.4)	176 (6.9)								
		1020 (40.1)	143 (5.6)								
B9*xxx*2-V20S*x*A	20.0 (0.8)	120…660 (4.7…26.0)	1124 (44.3)	462 (104)	968 (218)						
		780 (30.7)	889 (35.0)								
		900 (35.4)	715 (28.1)								
		1020 (40.1)	582 (22.9)								
B9*xxx*L-ALMO2C (no cover)	N/A	140…1940 (5.5…76.4)	5000 (196.9)	285 (64)	680 (153)	12900 (2900)	5800 (1304)	6600 (1484)	582 (429)	714 (527)	444 (328)
B9*xxx*L-ALMS2C (covered)	N/A			245 (55)	601 (135)						

System Combinations

For MP-Series Integrated Linear Stages and	**Refer to**
Kinetix 6000 (460V) drives and Kinetix 6200 and Kinetix 6500 (460V) drives	page 670
Kinetix 300 (480V) drives	page 718
Ultra3000 (460V) drives	page 746

MP-Series Integrated Linear Stages Motor Brake Specifications

Linear Stage Cat. No.	**Max Backlash (brake engaged)** µm (in.)	**Holding Force** N (lb)	**Coil Current at 24V DC** A	**Brake Response Time**		
				Release ms	**Engage (using external arc suppression device)**	
					MOV ms	**Diode** ms
MPAS-A/B*xxxx*1-V05S*x*A	100 (0.004)	5187 (1166)	0.46…0.56	58	20	42
MPAS-A/B*xxxx*2-V20S*x*A		1294 (291)				

MP-Series Integrated Linear Stage Standard Stroke Length and Weight Specifications

MP-Series Integrated Linear Stages, Ballscrew (150 mm frame size)

Linear Stage Cat. No. MPAS-	**Standard Stroke Lengths** mm (in.)	**Weight, approx.** kg (lb)
A/B6*xxx*2-V20S2A 20 mm/rev (0.8 in./rev) Ballscrew without brake	120 (4.7)	14.5 (32.0)
	180 (7.1)	15.4 (33.9)
	240 (9.5)	16.3 (35.8)
	300 (11.8)	17.1 (37.7)
	360 (14.2)	18.0 (39.6)
	420 (16.5)	18.9 (41.5)
	540 (21.3)	20.6 (45.4)
	660 (26.0)	22.4 (49.2)
A/B6*xxx*2-V20S4A 20 mm/rev (0.8 in./rev) Ballscrew with brake	120 (4.7)	15.0 (33.0)
	180 (7.1)	15.9 (34.9)
	240 (9.5)	16.7 (36.8)
	300 (11.8)	17.6 (38.7)
	360 (14.2)	18.5 (40.6)
	420 (16.5)	19.3 (42.5)
	540 (21.3)	21.1 (46.4)
	660 (26.0)	22.8 (50.2)

Linear Stage Cat. No. MPAS-	**Standard Stroke Lengths** mm (in.)	**Weight, approx.** kg (lb)
A/B6*xxx*1-V05S2A 5 mm/rev (0.2 in./rev) Ballscrew without brake	120 (4.7)	13.9 (30.5)
	180 (7.1)	14.7 (32.4)
	240 (9.5)	15.6 (34.3)
	300 (11.8)	16.4 (36.2)
	360 (14.2)	17.3 (38.1)
	420 (16.5)	18.2 (40.0)
	540 (21.3)	19.9 (43.7)
	660 (26.0)	21.6 (47.5)
A/B6*xxx*1-V05S4A 5 mm/rev (0.2 in./rev) Ballscrew with brake	120 (4.7)	14.3 (31.5)
	180 (7.1)	15.2 (33.4)
	240 (9.5)	16.0 (35.3)
	300 (11.8)	16.9 (37.2)
	360 (14.2)	17.8 (39.1)
	420 (16.5)	18.6 (40.9)
	540 (21.3)	20.3 (44.7)
	660 (26.0)	22.1 (48.5)

MP-Series Integrated Linear Stages, Direct Drive (150 mm frame size)

Linear Stage Cat. No. MPAS-	**Standard Stroke Lengths** mm (in.)	**Weight, approx. without Cover** kg (lb)	**Weight, approx. with Cover** kg (lb)
A6*xxx*B-ALM*x*2C Direct drive without brake	120 (4.7)	18.7 (41.0)	20.3 (44.7)
	180 (7.1)	20.0 (44.0)	21.8 (48.0)
	240 (9.5)	21.4 (47.1)	23.4 (51.4)
	300 (11.8)	22.8 (50.1)	24.9 (54.8)
	360 (14.2)	24.1 (53.1)	26.5 (58.2)
	420 (16.5)	25.5 (56.1)	28.0 (61.6)
	540 (21.3)	28.3 (62.3)	31.1 (68.4)
	660 (26.0)	31.0 (68.3)	34.2 (75.2)
	780 (30.7)	33.8 (74.3)	37.2 (81.9)
	900 (35.4)	36.5 (80.4)	40.3 (88.7)
	1020 (40.2)	39.5 (86.9)	43.6 (95.9)
	1140 (44.9)	42.3 (93.0)	46.7 (102.7)

MP-Series Integrated Linear Stages, Ballscrew (200 mm frame size)

Linear Stage Cat. No. MPAS-	**Standard Stroke Lengths** mm (in.)	**Weight, approx.** kg (lb)
A/B8*xxx*2-V20S2A 20 mm/rev (0.8 in./rev) Ballscrew without brake	120 (4.7)	16.5 (36.3)
	180 (7.1)	17.5 (38.4)
	240 (9.5)	18.4 (40.5)
	300 (11.8)	19.4 (42.7)
	360 (14.2)	20.4 (44.8)
	420 (16.5)	21.4 (47.0)
	540 (21.3)	23.3 (51.3)
	660 (26.0)	25.2 (55.5)
	780 (30.7)	27.2 (59.8)
	900 (35.4)	29.1 (64.1)
	1020 (40.2)	31.1 (68.4)
A/B8*xxx*2-V20S4A 20 mm/rev (0.8 in./rev) Ballscrew with brake	120 (4.7)	16.9 (37.2)
	180 (7.1)	17.9 (39.4)
	240 (9.5)	18.9 (41.5)
	300 (11.8)	19.9 (43.7)
	360 (14.2)	20.8 (45.8)
	420 (16.5)	21.8 (48.0)
	540 (21.3)	23.8 (52.3)
	660 (26.0)	25.7 (56.5)
	780 (30.7)	27.7 (60.8)
	900 (35.4)	29.6 (65.1)
	1020 (40.2)	31.6 (69.4)

Linear Stage Cat. No. MPAS-	**Standard Stroke Lengths** mm (in.)	**Weight, approx.** kg (lb)
A/B8*xxx*1-V05S2A 5 mm/rev (0.2 in./rev) Ballscrew without brake	120 (4.7)	15.9 (34.9)
	180 (7.1)	16.8 (37.0)
	240 (9.5)	17.8 (39.2)
	300 (11.8)	18.8 (41.3)
	360 (14.2)	19.8 (43.5)
	420 (16.5)	20.7 (45.6)
	540 (21.3)	22.7 (49.9)
	660 (26.0)	24.6 (54.2)
	780 (30.7)	26.6 (58.5)
	900 (35.4)	28.5 (62.7)
	1020 (40.2)	30.5 (67.1)
A/B8*xxx*1-V05S4A 5 mm/rev (0.2 in./rev) Ballscrew with brake	120 (4.7)	16.3 (35.9)
	180 (7.1)	17.3 (38.0)
	240 (9.5)	18.3 (40.2)
	300 (11.8)	19.2 (42.3)
	360 (14.2)	20.2 (44.5)
	420 (16.5)	21.2 (46.6)
	540 (21.3)	23.1 (50.9)
	660 (26.0)	25.1 (55.2)
	780 (30.7)	27.0 (59.5)
	900 (35.4)	29.0 (63.8)
	1020 (40.2)	30.9 (68.0)

MP-Series Integrated Linear Stages, Direct Drive (200 mm frame size)

Linear Stage Cat. No. MPAS-	**Standard Stroke Lengths** mm (in.)	**Weight, approx. without Cover** kg (lb)	**Weight, approx. with Cover** kg (lb)
A8*xxx*E-ALM*x*2C B8*xxx*F-ALM*x*2C Direct drive without brake	140 (5.5)	21.0 (46.1)	22.6 (49.8)
	200 (7.9)	21.9 (48.2)	23.7 (52.2)
	260 (10.2)	22.8 (50.2)	24.9 (54.7)
	320 (12.6)	24.1 (53.0)	26.3 (57.8)
	380 (15.0)	24.7 (54.3)	27.1 (59.5)
	440 (17.3)	26.0 (57.1)	28.5 (62.7)
	560 (22.0)	27.9 (61.3)	30.8 (67.7)
	680 (26.8)	29.7 (65.4)	33.0 (72.5)
	800 (31.5)	31.9 (70.2)	35.5 (78.1)
	920 (36.2)	34.1 (75.1)	38.1 (83.7)
	1040 (40.9)	36.2 (79.6)	40.5 (89.1)
	1280 (50.4)	39.9 (87.8)	44.9 (98.8)
	1520 (59.8)	44.4 (97.6)	50.1 (110.1)
	1760 (69.3)	48.1 (105.8)	54.5 (119.9)
	1940 (76.4)	51.2 (112.7)	58.1 (127.9)

MP-Series Integrated Linear Stages, Ballscrew (250 mm frame size)

Linear Stage Cat. No. MPAS-	**Standard Stroke Lengths** mm (in.)	**Weight, approx.** kg (lb)
A/B9*xxx*2-V20S2A 20 mm/rev (0.8 in./rev) Ballscrew without brake	120 (4.7)	19.8 (43.6)
	180 (7.1)	21.1 (46.4)
	240 (9.5)	22.4 (49.2)
	300 (11.8)	23.6 (52.0)
	360 (14.2)	24.9 (54.8)
	420 (16.5)	26.2 (57.6)
	540 (21.3)	28.7 (63.2)
	660 (26.0)	31.3 (68.8)
	780 (30.7)	33.8 (74.3)
	900 (35.4)	36.3 (79.9)
	1020 (40.2)	38.9 (85.5)
A/B9*xxx*2-V20S4A 20 mm/rev (0.8 in./rev) Ballscrew with brake	120 (4.7)	20.3 (44.6)
	180 (7.1)	21.5 (47.4)
	240 (9.5)	22.8 (50.2)
	300 (11.8)	24.1 (53.0)
	360 (14.2)	25.4 (55.8)
	420 (16.5)	26.6 (58.6)
	540 (21.3)	29.2 (64.2)
	660 (26.0)	31.7 (69.7)
	780 (30.7)	34.2 (75.3)
	900 (35.4)	36.8 (80.9)
	1020 (40.2)	39.3 (86.5)

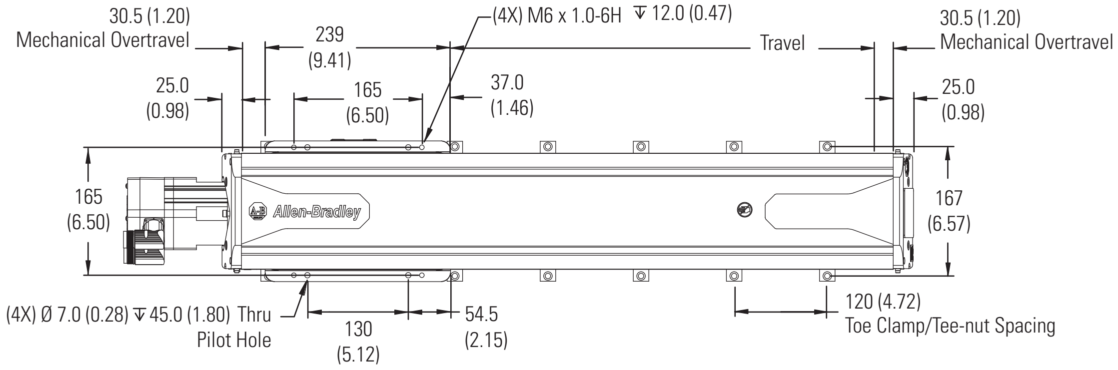
Linear Stage Cat. No. MPAS-	**Standard Stroke Lengths** mm (in.)	**Weight, approx.** kg (lb)
A/B9*xxx*1-V05S2A 5 mm/rev (0.2 in./rev) Ballscrew without brake	120 (4.7)	19.2 (42.3)
	180 (7.1)	20.5 (45.0)
	240 (9.5)	21.8 (47.9)
	300 (11.8)	23.0 (50.6)
	360 (14.2)	24.3 (53.4)
	420 (16.5)	25.6 (56.2)
	540 (21.3)	28.1 (61.8)
	660 (26.0)	30.6 (67.4)
	780 (30.7)	33.2 (73.0)
	900 (35.4)	35.7 (78.6)
	1020 (40.2)	38.2 (84.1)
A/B9*xxx*1-V05S4A 5 mm/rev (0.2 in./rev) Ballscrew with brake	120 (4.7)	19.7 (43.3)
	180 (7.1)	20.9 (46.0)
	240 (9.5)	22.2 (48.8)
	300 (11.8)	23.5 (51.6)
	360 (14.2)	24.7 (54.4)
	420 (16.5)	26.0 (57.2)
	540 (21.3)	28.5 (62.8)
	660 (26.0)	31.1 (68.4)
	780 (30.7)	33.6 (74.0)
	900 (35.4)	36.2 (79.6)
	1020 (40.2)	38.7 (85.1)

MP-Series Integrated Linear Stages, Direct Drive (250 mm frame size)

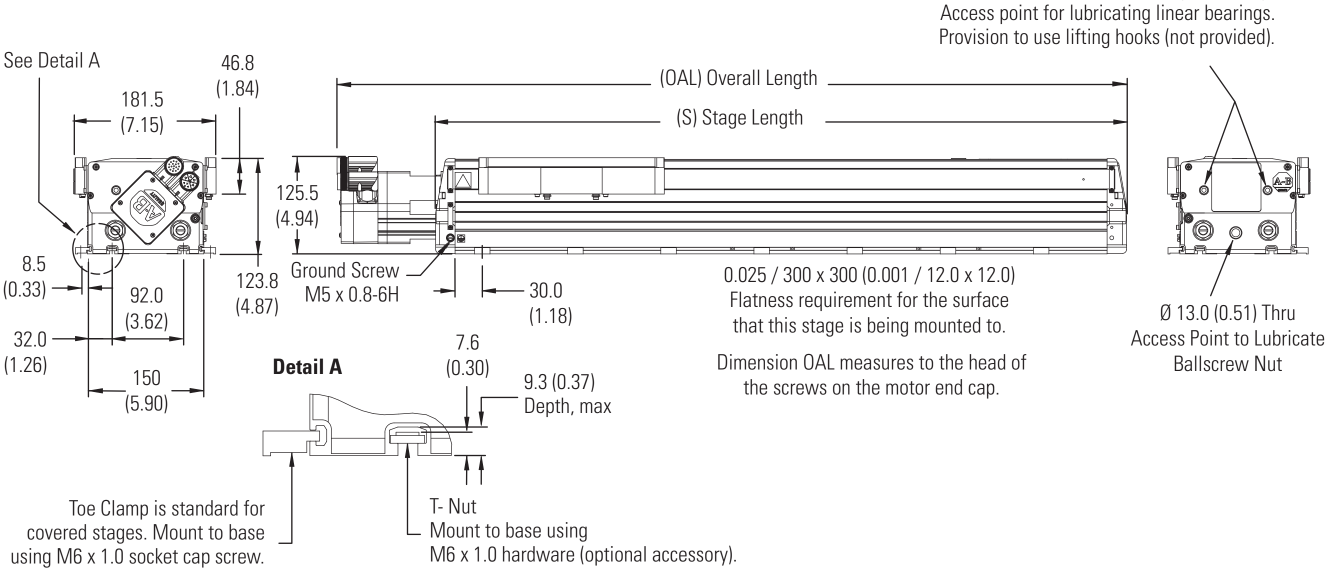
Linear Stage Cat. No. MPAS-	**Standard Stroke Lengths** mm (in.)	**Weight, approx. without Cover** kg (lb)	**Weight, approx. with Cover** kg (lb)
A9*xxx*K-ALM*x*2C B9*xxx*L-ALM*x*2C Direct drive without brake	140 (5.5)	26.8 (59.0)	29.0 (63.7)
	200 (7.9)	28.6 (62.8)	30.9 (68.0)
	260 (10.2)	29.8 (65.6)	32.5 (71.4)
	320 (12.6)	31.1 (68.5)	34.0 (74.7)
	380 (15.0)	32.4 (71.3)	35.5 (78.0)
	440 (17.3)	33.7 (74.2)	37.0 (81.4)
	560 (22.0)	36.8 (80.9)	40.5 (89.1)
	680 (26.8)	40.0 (87.9)	43.5 (95.8)
	800 (31.5)	42.4 (93.2)	47.0 (103.4)
	920 (36.2)	45.0 (98.9)	50.1 (110.1)
	1040 (40.9)	47.8 (105.1)	53.3 (117.3)
	1280 (50.4)	53.4 (117.4)	59.8 (131.6)
	1520 (59.8)	59.0 (129.8)	66.4 (146.0)
	1760 (69.3)	64.6 (142.1)	72.9 (160.3)
	1940 (76.4)	68.5 (150.6)	77.4 (170.3)

MP-Series Integrated Linear Stage Dimensions

MPAS-A/B6*xxx*1/2-V*xx*S*x*A (ballscrew) Linear Stage



Dimensions are in mm (in.)



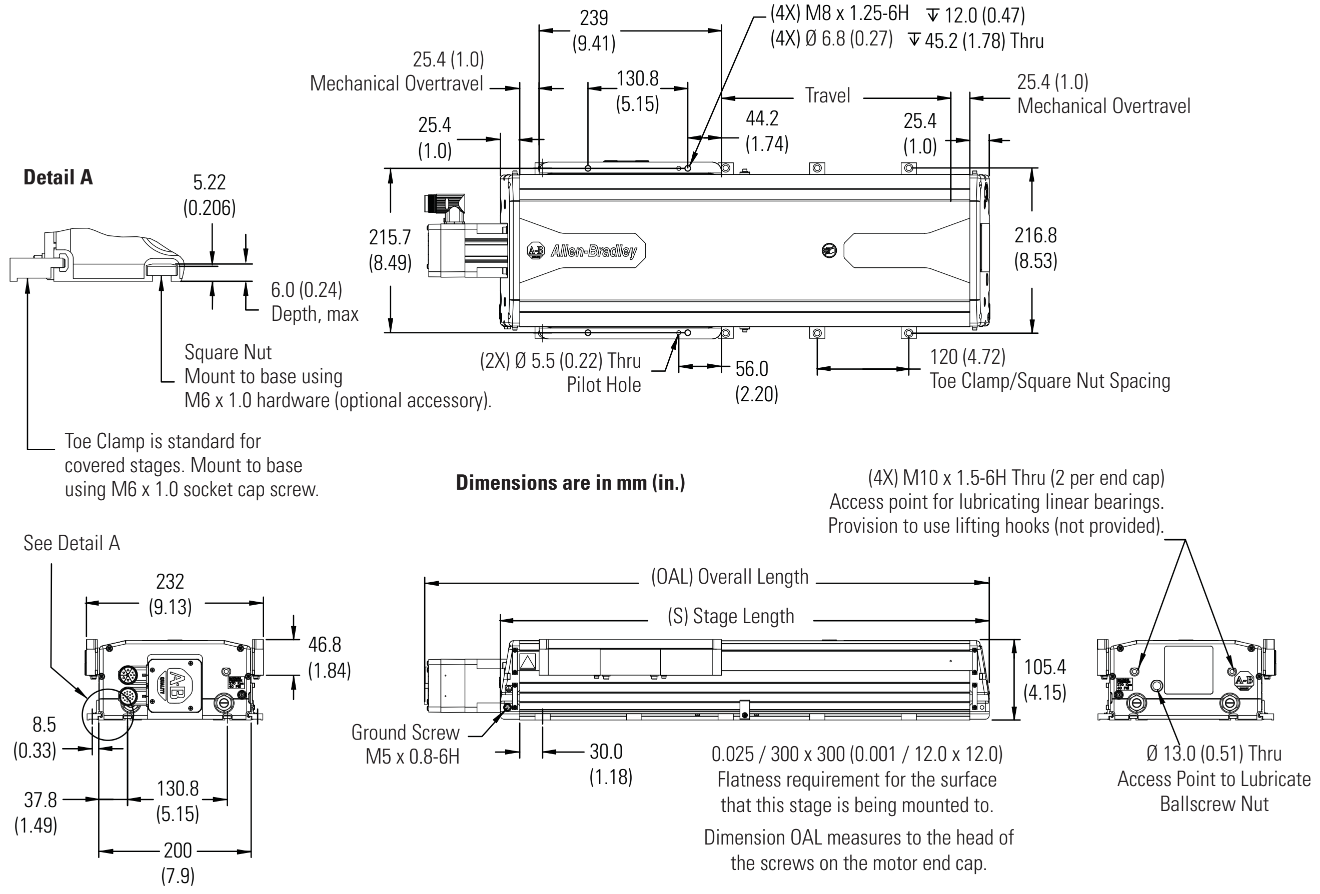
Linear Stage Cat. No. MPAS-	**S** mm (in.)	**OAL (5 mm/rev)** [1] mm (in.)	**OAL (20 mm/rev)** [2] mm (in.)
A/B6012*x*-V*xx*S*x*A	470 (18.5)	569 (22.4)	594 (23.4)
A/B6018*x*-V*xx*S*x*A	530 (20.9)	629 (24.7)	654 (25.8)
A/B6024*x*-V*xx*S*x*A	590 (23.2)	689 (27.1)	714 (28.1)
A/B6030*x*-V*xx*S*x*A	650 (25.6)	749 (29.5)	774 (30.5)
A/B6036*x*-V*xx*S*x*A	710 (28.0)	809 (31.8)	834 (32.8)
A/B6042*x*-V*xx*S*x*A	770 (30.3)	869 (34.2)	894 (35.2)
A/B6054*x*-V*xx*S*x*A	890 (35.0)	989 (38.9)	1014 (39.9)
A/B6066*x*-V*xx*S*x*A	1010 (39.8)	1109 (43.6)	1134 (44.6)

(1) If ordering an MPAS-A/B6*xxx*-V05S4A actuator with brake, add 39.0 mm (1.53 in.) to dimension OAL.

(2) If ordering an MPAS-A/B6*xxx*-V20S4A actuator with brake, add 39.0 mm (1.53 in.) to dimension OAL.

Actuators are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

MPAS-A/B8*xxx*1/2-V*xx*S*x*A (ballscrew) Linear Stage



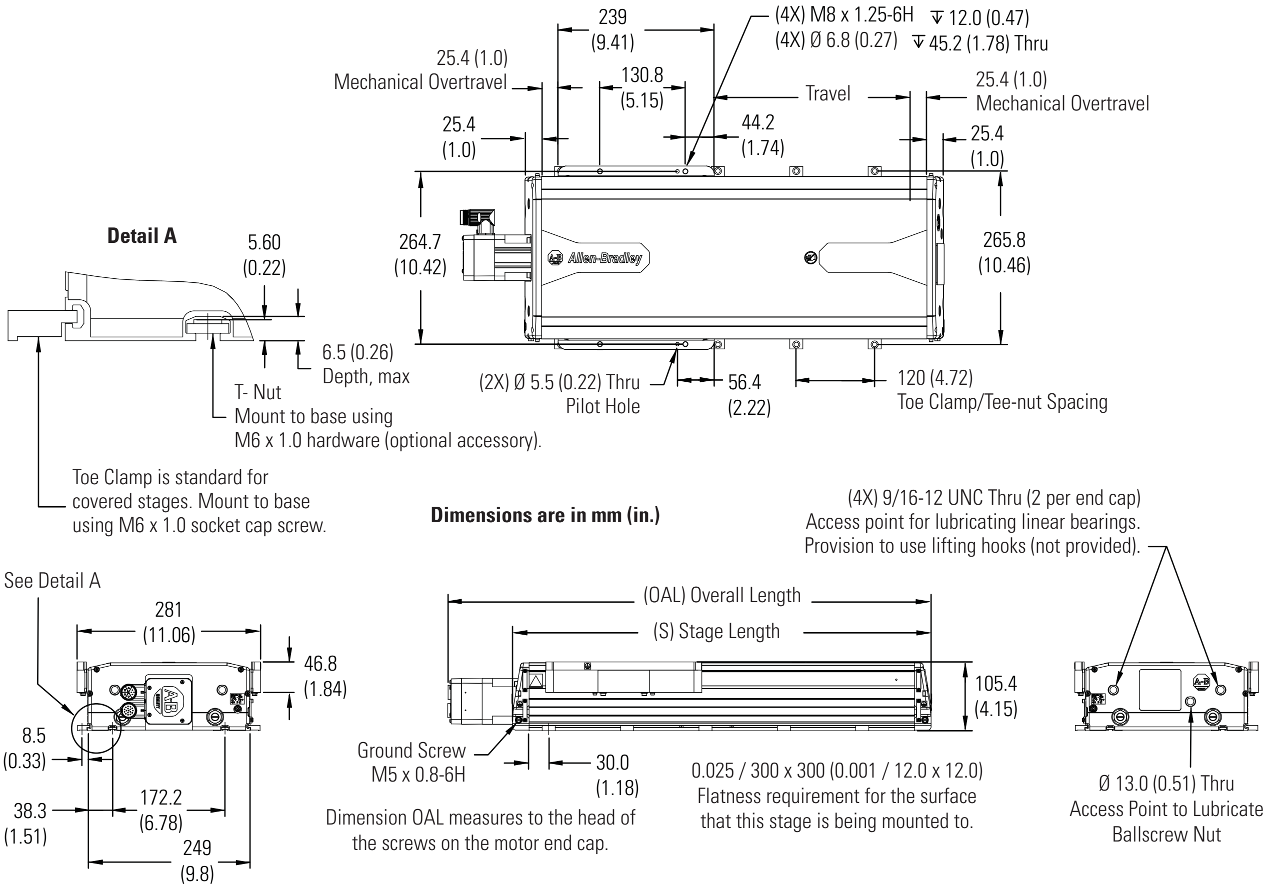
Actuator Series MPAS-	**S** mm (in.)	**OAL (5 mm/rev)** (1) mm (in.)	**OAL (20 mm/rev)** (2) mm (in.)
A/B8012*x*-V*xx*S*x*A	461 (18.1)	557 (21.9)	582 (22.9)
A/B8018*x*-V*xx*S*x*A	521 (20.5)	617 (24.3)	642 (25.3)
A/B8024*x*-V*xx*S*x*A	581 (22.9)	677 (26.6)	702 (27.6)
A/B8030*x*-V*xx*S*x*A	641 (25.2)	737 (29.0)	762 (30.0)
A/B8036*x*-V*xx*S*x*A	701 (27.6)	797 (31.4)	822 (32.4)
A/B8042*x*-V*xx*S*x*A	761 (29.9)	857 (33.7)	882 (34.7)
A/B8054*x*-V*xx*S*x*A	881 (34.7)	977 (38.5)	1002 (39.5)
A/B8066*x*-V*xx*S*x*A	1001 (39.4)	1097 (43.2)	1122 (44.2)
A/B8078*x*-V*xx*S*x*A	1121 (44.1)	1217 (47.9)	1242 (48.9)
A/B8090*x*-V*xx*S*x*A	1241 (48.8)	1337 (52.6)	1362 (53.6)
A/B8102*x*-V*xx*S*x*A	1361 (53.6)	1457 (57.4)	1482 (58.4)

(1) If ordering an MPAS-A/B8*xxx*-V05S4A actuator with brake, add 39.0 mm (1.53 in.) to dimension OAL.

(2) If ordering an MPAS-A/B8*xxx*-V20S4A actuator with brake, add 39.0 mm (1.53 in.) to dimension OAL.

Actuators are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

MPAS-A/B9*xxx*1/2-V*xx*S*x*A (ballscrew) Linear Stage



Linear Stage Cat. No. MPAS-	**S** mm (in.)	**OAL (5 mm/rev)** (1) mm (in.)	**OAL (20 mm/rev)** (2) mm (in.)
A/B9012*x*-V*xx*S*x*A	461 (18.1)	557 (21.9)	582 (22.9)
A/B9018*x*-V*xx*S*x*A	521 (20.5)	617 (24.3)	642 (25.3)
A/B9024*x*-V*xx*S*x*A	581 (22.9)	677 (26.6)	702 (27.6)
A/B9030*x*-V*xx*S*x*A	641 (25.2)	737 (29.0)	762 (30.0)
A/B9036*x*-V*xx*S*x*A	701 (27.6)	797 (31.4)	822 (32.4)
A/B9042*x*-V*xx*S*x*A	761 (29.9)	857 (33.7)	882 (34.7)

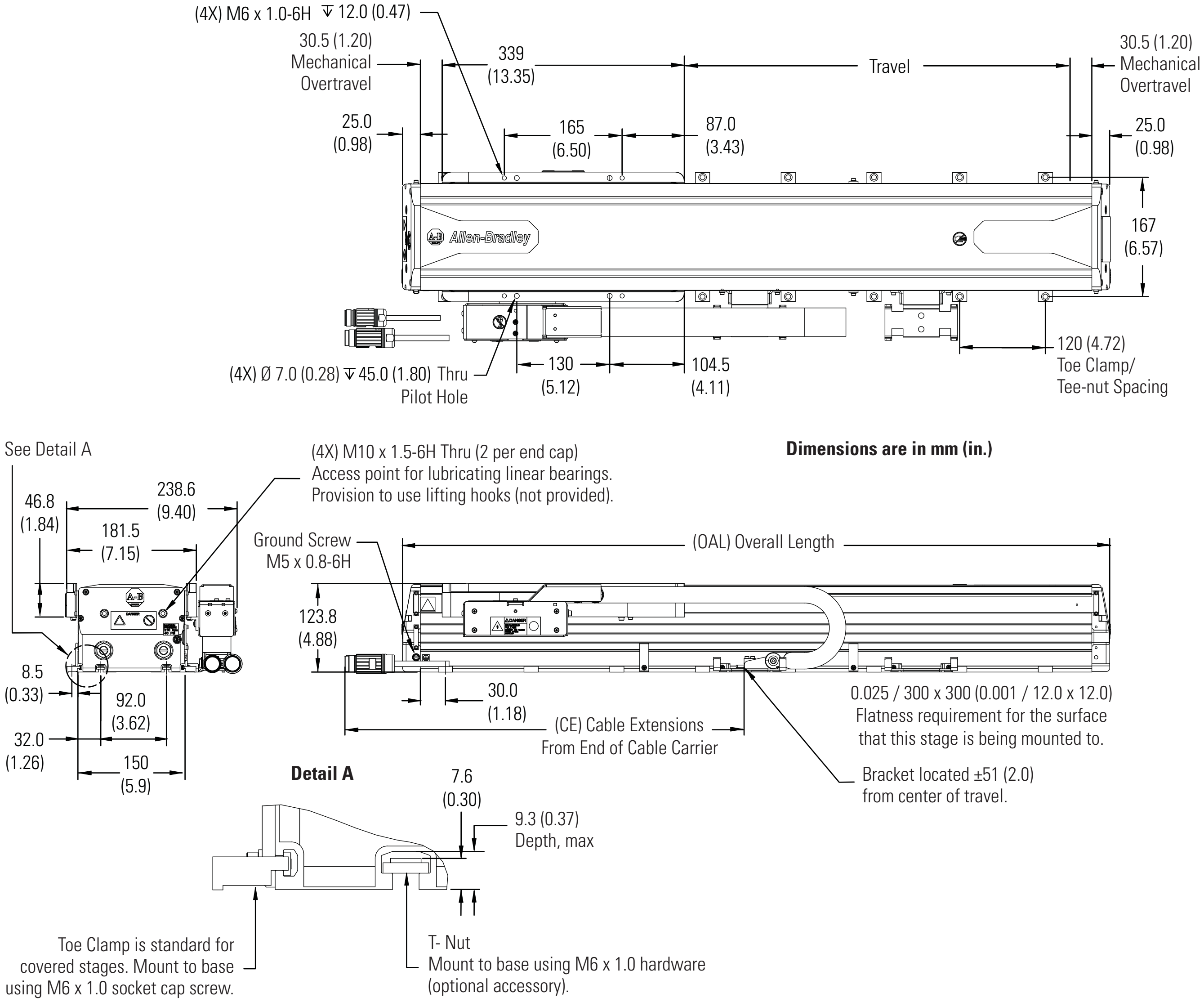
Linear Stage Cat. No. MPAS-	**S** mm (in.)	**OAL (5 mm/rev)** (1) mm (in.)	**OAL (20 mm/rev)** (2) mm (in.)
A/B9054*x*-V*xx*S*x*A	881 (34.7)	977 (38.5)	1002 (39.5)
A/B9066*x*-V*xx*S*x*A	1001 (39.4)	1097 (43.2)	1122 (44.2)
A/B9078*x*-V*xx*S*x*A	1121 (44.1)	1217 (47.9)	1242 (48.9)
A/B9090*x*-V*xx*S*x*A	1241 (48.8)	1337 (52.6)	1362 (53.6)
A/B9102*x*-V*xx*S*x*A	1361 (53.6)	1457 (57.4)	1482 (58.4)

(1) If ordering an MPAS-A/B9*xxx*-V05S4A actuator with brake, add 39.0 mm (1.53 in.) to dimension OAL.

(2) If ordering an MPAS-A/B9*xxx*-V20S4A actuator with brake, add 39.0 mm (1.53 in.) to dimension OAL.

Actuators are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

MPAS-A6*xxx*B-ALM*x*2C (direct drive) Linear Stage

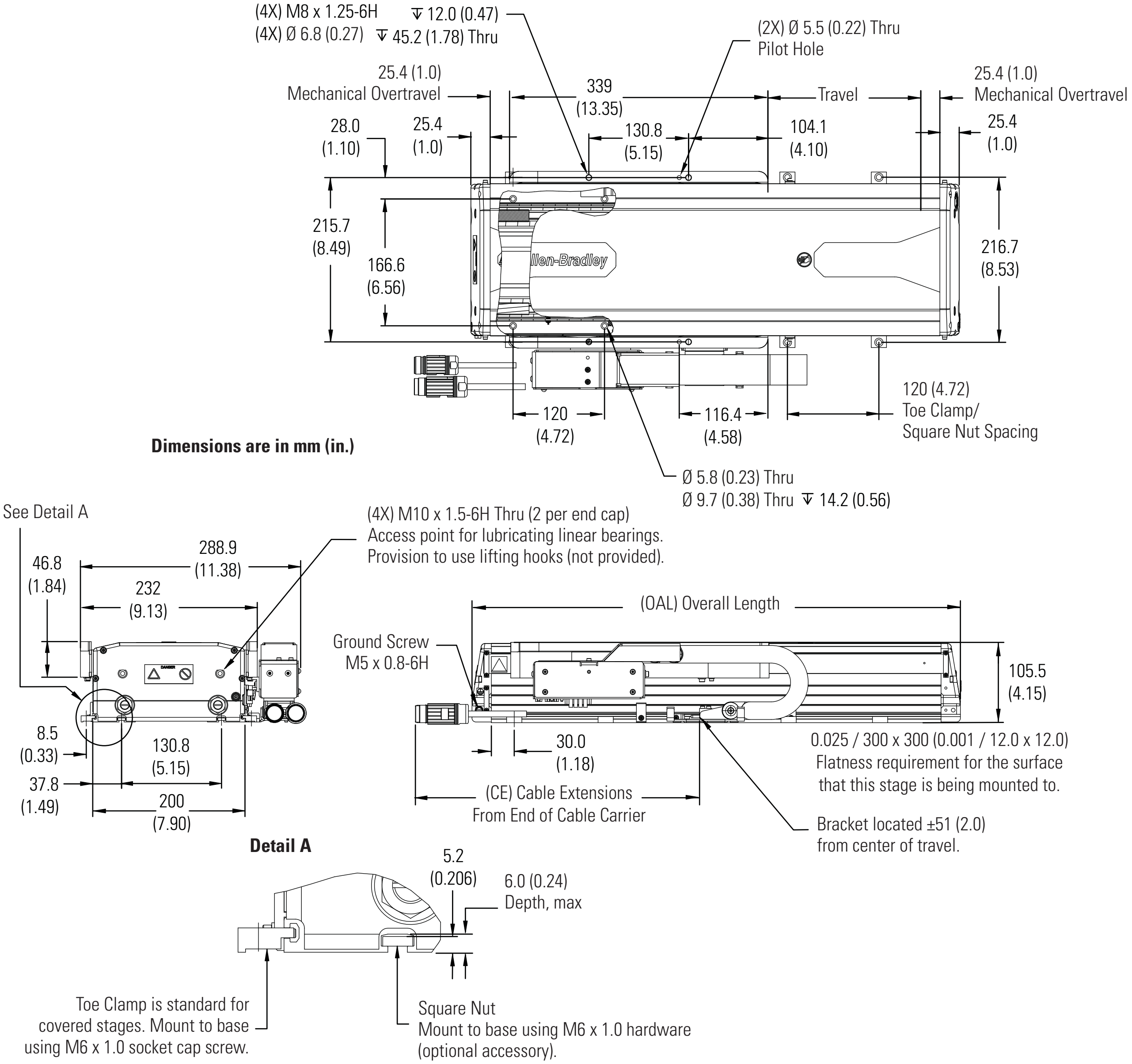


Linear Stage Cat. No. MPAS-	**OAL** mm (in.)	**CE** mm (in.)
A6012B-ALM*x*2C	570 (22.4)	1218 (48.0)
A6018B-ALM*x*2C	630 (24.8)	1193 (47.0)
A6024B-ALM*x*2C	690 (27.2)	1168 (46.0)
A6030B-ALM*x*2C	750 (29.5)	1118 (44.0)
A6036B-ALM*x*2C	810 (31.9)	1093 (43.0)
A6042B-ALM*x*2C	870 (34.3)	1068 (42.0)

Linear Stage Cat. No. MPAS-	**OAL** mm (in.)	**CE** mm (in.)
A6054B-ALM*x*2C	990 (39.0)	1018 (40.0)
A6066B-ALM*x*2C	1110 (43.7)	943 (37.0)
A6078B-ALM*x*2C	1230 (48.4)	893 (35.0)
A6090B-ALM*x*2C	1350 (53.1)	818 (32.0)
A6102B-ALM*x*2C	1470 (57.9)	1768 (70.0)
A6114B-ALM*x*2C	1590 (62.6)	1718 (68.0)

Actuators are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

MPAS-A/B8*xxxx*-ALM*x*2C (direct drive) Linear Stage



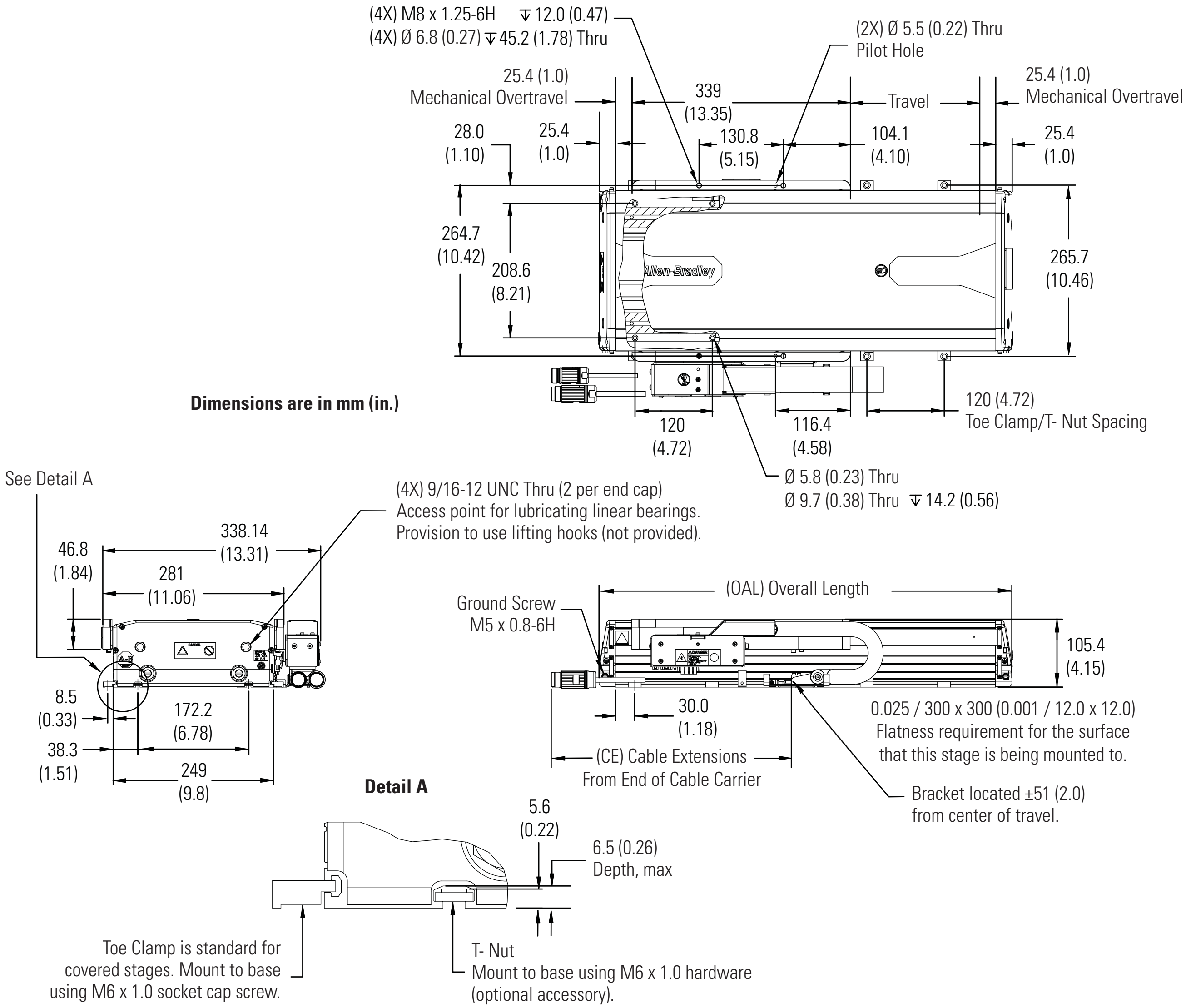
Linear Stage Cat. No. MPAS-	**OAL** mm (in.)	**CE** mm (in.)
A/B8014*x*-ALM*x*2C	581 (22.9)	1243 (49.0)
A/B8020*x*-ALM*x*2C	641 (25.2)	1218 (48.0)
A/B8026*x*-ALM*x*2C	701 (27.6)	1168 (46.0)
A/B8032*x*-ALM*x*2C	761 (30.0)	1143 (45.0)
A/B8038*x*-ALM*x*2C	821 (32.3)	1118 (44.0)

Linear Stage Cat. No. MPAS-	**OAL** mm (in.)	**CE** mm (in.)
A/B8044*x*-ALM*x*2C	881 (34.7)	1093 (43.0)
A/B8056*x*-ALM*x*2C	1001 (39.4)	1018 (40.0)
A/B8068*x*-ALM*x*2C	1121 (44.1)	968 (38.0)
A/B8080*x*-ALM*x*2C	1241 (48.9)	918 (36.0)
A/B8092*x*-ALM*x*2C	1361 (53.6)	843 (33.0)

Linear Stage Cat. No. MPAS-	**OAL** mm (in.)	**CE** mm (in.)
A/B8104*x*-ALM*x*2C	1481 (58.3)	1793 (71.0)
A/B8128*x*-ALM*x*2C	1721 (67.8)	1668 (66.0)
A/B8152*x*-ALM*x*2C	1961 (77.2)	1543 (61.0)
A/B8176*x*-ALM*x*2C	2201 (86.7)	1418 (56.0)
A/B8194*x*-ALM*x*2C	2381 (93.7)	1343 (53.0)

Actuators are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

MPAS-A/B9*xxxx*-ALM*x*2C (direct drive) Linear Stage



Linear Stage Cat. No. MPAS-	OAL mm (in.)	CE mm (in.)
A/B9014*x*-ALM*x*2C	581 (22.9)	1240 (49.0)
A/B9020*x*-ALM*x*2C	641 (25.2)	1220 (48.0)
A/B9026*x*-ALM*x*2C	701 (27.6)	1170 (46.0)
A/B9032*x*-ALM*x*2C	761 (30.0)	1140 (45.0)
A/B9038*x*-ALM*x*2C	821 (32.3)	1120 (44.0)

Linear Stage Cat. No. MPAS-	OAL mm (in.)	CE mm (in.)
A/B9044*x*-ALM*x*2C	881 (34.7)	1090 (43.0)
A/B9056*x*-ALM*x*2C	1001 (39.4)	1020 (40.0)
A/B9068*x*-ALM*x*2C	1121 (44.1)	960 (38.0)
A/B9080*x*-ALM*x*2C	1241 (48.9)	910 (36.0)
A/B9092*x*-ALM*x*2C	1361 (53.6)	840 (33.0)

Linear Stage Cat. No. MPAS-	OAL mm (in.)	CE mm (in.)
A/B9104*x*-ALM*x*2C	1481 (58.3)	1800 (71.0)
A/B9128*x*-ALM*x*2C	1721 (67.8)	1670 (66.0)
A/B9152*x*-ALM*x*2C	1961 (77.2)	1550 (61.0)
A/B9176*x*-ALM*x*2C	2201 (86.7)	1420 (56.0)
A/B9194*x*-ALM*x*2C	2381 (93.7)	1350 (53.0)

Actuators are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

MP-Series Integrated Linear Stage Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your actuator. For questions regarding product availability, contact your Allen-Bradley distributor.

MP AS - *x x xxx x* - *x xx x x x*

Cable
A = No cable track module (ballscrew only)
C = Cable track module with IP67 intercontec connectors (direct drive only)

Brake
2 = No brake
4 = 24V DC brake (ballscrew only)

Cover
S = Covered with strip seals (IP30 protection)
O = Open (direct drive only)

Screw Lead/Motor Type
05 = 5.0 mm/rev (0.19 in./rev) ballscrew with rotary motor
20 = 20.0 mm/rev (0.79 in./rev) ballscrew with rotary motor
LM = Linear motor (direct drive only)

Feedback
V = Multi-turn high-resolution encoder (absolute feedback) 128 cycle/rev. (ballscrew only)
A = 5 micron resolution incremental magnetic linear encoder (direct drive only)

Mechanical Drive/Motor Type
1 = Ballscrew/MPLS-*x*210E-V
2 = Ballscrew/MPLS-*x*220H-V
B = Direct Drive/6-frame, 230V
K = Direct Drive/9-frame, 230V
E = Direct Drive/8-frame, 230V
L = Direct Drive/9-frame, 460V
F = Direct Drive/8-frame, 460V

Stroke Length (1)
xxx = *xxx* mm x 10

Frame Size (2)
6 = 150 mm base width
8 = 200 mm base width
9 = 250 mm base width

Voltage Class
A = 200V
B = 400V

Actuator Type
AS = Actuator, linear stage

Actuator Series
MP = MP-Series

(1) For 6-frame direct-drive linear stages, variable *xxx* (mm stroke x 10) is 012, 018, 024, 030, 036, 042, 054, 066, 078, 090, 102, or 114.
For 8 and 9-frame direct-drive linear stages, variable *xxx* (mm stroke x 10) is 014, 020, 026, 032, 038, 044, 056, 068, 080, 092, 104, 128, 152, 176, or 194.
For 6-frame ballscrew linear stages, variable *xxx* (mm stroke x 10) is 012, 018, 024, 030, 036, 042, 054, or 066.
For 8 and 9-frame ballscrew linear stages, variable *xxx* (mm stroke x 10) is 012, 018, 024, 030, 036, 042, 054, 066, 078, 090, or 102.

(2) The 150 mm (6-frame) linear stages are available in ballscrew (230/460V) and direct drive (230V only).
The 200 mm (8-frame) linear stages are available in ballscrew and direct drive (230/460V).
The 250 mm (9-frame) linear stages are available in ballscrew and direct drive (230/460V).

MP-Series Integrated Multi-axis Linear Stages

Cartesian X/Y Configuration

Center Stacked X/Y Configuration

Center Stacked X/Z Configuration

MP-Series integrated multi-axis linear stages extend the Allen-Bradley actuator portfolio into pre-defined and pre-assembled multi-axis configurations to suit a variety of manufacturing needs. These configurations come pre-assembled with known orthogonality and cable management, and are selectable in Motion Analyzer sizing software, version 4.7 or later.

The cartesian X/Y configuration is typically used when you need to perform an operation on a workpiece from above such as picking and placing, dispensing, or scanning. This configuration is available with both direct-drive linear motors and ballscrew-actuated stages.

The center stacked X/Y is a configuration that is often used to position your workpiece under a stationary object such as a camera or dispenser head. This configuration is available with both direct-drive linear motors and ballscrew-actuated stages.

The center stacked X/Z configuration is often used in applications where you need to pick an object from one line and place it on a perpendicular line. This configuration is available with both direct-drive linear motors and ballscrew actuated X-axis stages, and ballscrew actuated Z-axis stages.

For drive compatibility, refer to Servo Drives on page 14.

MP-Series Integrated Multi-axis Linear Stage Features

- Your choice of direct-drive linear motor or ballscrew actuation. Direct-drive linear motor actuation provides greater repeatability, increased reliability, and higher dynamics. Ballscrew actuation provides greater linear thrust capabilities.
- Ease of selection in Motion Analyzer software, version 4.7 or later, that reduces the time and cost associated with selecting a multi-axis linear stage solution.
- Ease of machine design with the availability of CAD and Solid Models that reduce the time needed to generate machine drawings.
- Ease of configuration with the integration of RSLogix 5000 software that reduces set-up time.
- Factory-supplied cable management that is field replaceable for ease of maintenance.
- Orthogonality of 30 arc seconds.

MP-Series Integrated Multi-axis Linear Stage Accessory Kits

- Cable track module replacement kits
- Strip seal replacement kits
- Top cover kits (for Y or Z-axis only)
- Side cover kits
- Coupling kits (for Y or Z-axis only)
- Tee-nut kit (package of 10)
- Tee-nut bar kit
- Grease gun kit
- Grease replacement cartridge
- Rotary servo motor (for Y or Z-axis only)

Accessory Kits Common to All Multi-axis Linear Stages

Linear Stage Cat. No.	Description	Accessory Cat. No.
MPAS-A/B*xxxx*	Kit, grease gun for all integrated linear stages.	MPAS-GPUMP
	Grease gun refill cartridge for all integrated linear stages.	MPAS-CART
MPAS-A6*xxx*, MPAS-B6*xxx*	Kit, Tee-nuts (10 per package).	MPAS-6-TNUT
MPAS-A8*xxx*, MPAS-B8*xxx*		MPAS-8-TNUT
MPAS-A9*xxx*, MPAS-B9*xxx*		MPAS-9-TNUT

Accessory Kits for Multi-axis Direct-drive Linear Stages

Refer to Accessory Kits for Single-axis Direct-drive Linear Stages on page 94 for accessory kit descriptions and catalog numbers.

Accessory Kits for Multi-axis Ballscrew Linear Stages

Refer to Accessory Kits for Single-axis Ballscrew Linear Stages on page 95 for accessory kit descriptions and catalog numbers.

Accessory Kits for Center Stacked X/Y Cable Modules

Multi-axis Linear Stage Cat. No.	Description	Accessory Cat. No.
MPMA-*x*ABC0C0A0-*xxx*	Cable track module for MPMA-*x*AB series X/Y center stacked stages	MPMA-*x*ABC0C0A0-CBL
MPMA-*x*ABC6C6A0-*xxx*		MPMA-*x*ABC6C6A0-CBL
MPMA-*x*ABD2D2A0-*xxx*		MPMA-*x*ABD2D2A0-CBL
MPMA-*x*ABD8D8A0-*xxx*		MPMA-*x*ABD8D8A0-CBL
MPMA-*x*ABE4E4A0-*xxx*		MPMA-*x*ABE4E4A0-CBL
MPMA-*x*ABF6F6A0-*xxx*		MPMA-*x*ABF6F6A0-CBL
MPMA-*x*ACB4B4A0-*xxx*	Cable track module for MPMA-*x*AC series X/Y center stacked stages	MPMA-*x*ACB4B4A0-CBL
MPMA-*x*ACC0C0A0-*xxx*		MPMA-*x*ACC0C0A0-CBL
MPMA-*x*ACC6C6A0-*xxx*		MPMA-*x*ACC6C6A0-CBL
MPMA-*x*ACD2D2A0-*xxx*		MPMA-*x*ACD2D2A0-CBL
MPMA-*x*APB8B8A0-*xxx*	Cable track module for MPMA-*x*AP series X/Y center stacked stages	MPMA-*x*APB8B8A0-CBL
MPMA-*x*APC4C4A0-*xxx*		MPMA-*x*APC4C4A0-CBL
MPMA-*x*APD0D0A0-*xxx*		MPMA-*x*APD0D0A0-CBL
MPMA-*x*APE2E2A0-*xxx*		MPMA-*x*APE2E2A0-CBL
MPMA-*x*APG6G6A0-*xxx*		MPMA-*x*APG6G6A0-CBL

Accessory Kits for Cartesian Stacked X/Y Cable Modules

Multi-axis Linear Stage Cat. No.	Description	Accessory Cat. No.
MPMA-*x*CBD2D2A0-*xxx*	Cable track module for MPMA-*x*CB series X/Y cartesian stacked stages	MPMA-*x*CBD2D2A0-CBL
MPMA-*x*CBE4D2A0-*xxx*		MPMA-*x*CBE4D2A0-CBL
MPMA-*x*CBE4E4A0-*xxx*		MPMA-*x*CBE4E4A0-CBL
MPMA-*x*CBF6E4A0-*xxx*		MPMA-*x*CBF6E4A0-CBL
MPMA-*x*CBG8E4A0-*xxx*		MPMA-*x*CBG8E4A0-CBL
MPMA-*x*CBG8F6A0-*xxx*		MPMA-*x*CBG8F6A0-CBL
MPMA-*x*CBI0F6A0-*xxx*		MPMA-*x*CBI0F6A0-CBL
MPMA-*x*CBJ2F6A0-*xxx*		MPMA-*x*CBJ2F6A0-CBL
MPMA-*x*CQD0D0A0-*xxx*	Cable track module for MPMA-*x*CQ series X/Y cartesian stacked stages	MPMA-*x*CQD0D0A0-CBL
MPMA-*x*CQE2D0A0-*xxx*		MPMA-*x*CQE2D0A0-CBL
MPMA-*x*CQE2E2A0-*xxx*		MPMA-*x*CQE2E2A0-CBL
MPMA-*x*CQG6E2A0-*xxx*		MPMA-*x*CQG6E2A0-CBL
MPMA-*x*CQG6G6A0-*xxx*		MPMA-*x*CQG6G6A0-CBL
MPMA-*x*CQH8G6A0-*xxx*		MPMA-*x*CQH8G6A0-CBL
MPMA-*x*CQJ0G6A0-*xxx*		MPMA-*x*CQJ0G6A0-CBL

Accessory Kits for Center Stacked X/Z Cable Modules

Multi-axis Linear Stage Cat. No.	Description	Accessory Cat. No.
MPMA-*x*BDD2A0B8-*xxx*	Cable track module for MPMA-*x*BD series X/Z center stacked stages	MPMA-xBDD2A0B8-CBL
MPMA-*x*BDD2A0D0-*xxx*		MPMA-xBDD2A0D0-CBL
MPMA-*x*BDE4A0B8-*xxx*		MPMA-xBDE4A0B8-CBL
MPMA-*x*BDE4A0D0-*xxx*		MPMA-*x*BDE4A0D0-CBL
MPMA-*x*BDE4A0E2-*xxx*		MPMA-*x*BDE4A0E2-CBL
MPMA-*x*BDG8A0B8-*xxx*		MPMA-*x*BDG8A0B8-CBL
MPMA-*x*BDG8A0D0-*xxx*		MPMA-*x*BDG8A0D0-CBL
MPMA-*x*BDG8A0E2-*xxx*		MPMA-*x*BDG8A0E2-CBL
MPMA-*x*BDI0A0B8-*xxx*		MPMA-*x*BDI0A0B8-CBL
MPMA-*x*BDI0A0D0-*xxx*		MPMA-*x*BDI0A0D0-CBL
MPMA-*x*BDI0A0E2-*xxx*		MPMA-*x*BDI0A0E2-CBL
MPMA-*x*BDJ2A0D0-*xxx*		MPMA-*x*BDJ2A0D0-CBL
MPMA-*x*BDJ2A0E2-*xxx*		MPMA-*x*BDJ2A0E2-CBL
MPMA-*x*BDJ2A0F4-*xxx*		MPMA-*x*BDJ2A0F4-CBL
MPMA-*x*BED2A0B8-*xxx*	Cable track module for MPMA-*x*BE series X/Z center stacked stages	MPMA-*x*BED2A0B8-CBL
MPMA-*x*BED2A0D0-*xxx*		MPMA-*x*BED2A0D0-CBL
MPMA-*x*BEE4A0B8-*xxx*		MPMA-*x*BEE4A0B8-CBL
MPMA-*x*BEE4A0D0-*xxx*		MPMA-*x*BEE4A0D0-CBL
MPMA-*x*BEG8A0B8-*xxx*		MPMA-*x*BEG8A0B8-CBL
MPMA-*x*BEG8A0D0-*xxx*		MPMA-*x*BEG8A0D0-CBL
MPMA-*x*BID0A0B8-*xxx*	Cable track module for MPMA-*x*BI series X/Z center stacked stages	MPMA-*x*BID0A0B8-CBL
MPMA-*x*BID0A0D0-*xxx*		MPMA-*x*BID0A0D0-CBL
MPMA-*x*BIE2A0B8-*xxx*		MPMA-*x*BIE2A0B8-CBL
MPMA-*x*BIE2A0D0-*xxx*		MPMA-*x*BIE2A0D0-CBL
MPMA-*x*BIG6A0B8-*xxx*		MPMA-*x*BIG6A0B8-CBL
MPMP-*x*BIG6A0D0-*xxx*		MPMP-*x*BIG6A0D0-CBL

Mounting Bar Accessory Kits for Multi-axis X-axis Linear Stages

Multi-axis Linear Stage Cat. No.	**Description**	**Accessory Cat. No.**
MPMA-*xxx*B4*xxxx-xxx*	Mounting bar for X-axis having B4 travel	MPAS-TBAR-B4
MPMA-*xxx*B8*xxxx-xxx*	Mounting bar for X-axis having B8 travel	MPAS-TBAR-B8
MPMA-*xxx*C0*xxxx-xxx*	Mounting bar for X-axis having C0 travel	MPAS-TBAR-C0
MPMA-*xxx*C4*xxxx-xxx*	Mounting bar for X-axis having C4 travel	MPAS-TBAR-C4
MPMA-*xxx*C6*xxxx-xxx*	Mounting bar for X-axis having C6 travel	MPAS-TBAR-C6
MPMA-*xxx*D0*xxxx-xxx*	Mounting bar for X-axis having D0 travel	MPAS-TBAR-D0
MPMA-*xxx*D2*xxxx-xxx*	Mounting bar for X-axis having D2 travel	MPAS-TBAR-D2
MPMA-*xxx*D8*xxxx-xxx*	Mounting bar for X-axis having D8 travel	MPAS-TBAR-D8
MPMA-xxxE2*xxxx-xxx*	Mounting bar for X-axis having E2 travel	MPAS-TBAR-E2
MPMA-*xxx*E4*xxxx-xxx*	Mounting bar for X-axis having E4 travel	MPAS-TBAR-E4
MPMA-*xxx*F6*xxxx-xxx*	Mounting bar for X-axis having F6 travel	MPAS-TBAR-F6
MPMA-*xxx*G6*xxxx-xxx*	Mounting bar for X-axis having G6 travel	MPAS-TBAR-G6
MPMA-*xxx*G8*xxxx-xxx*	Mounting bar for X-axis having G8 travel	MPAS-TBAR-G8
MPMA-*xxx*H8*xxxx-xxx*	Mounting bar for X-axis having H8 travel	MPAS-TBAR-H8
MPMA-*xxx*I0*xxxx-xxx*	Mounting bar for X-axis having I0 travel	MPAS-TBAR-I0
MPMA-*xxx*J0*xxxx-xxx*	Mounting bar for X-axis having J0 travel	MPAS-TBAR-J0
MPMA-*xxx*J2*xxxx-xxx*	Mounting bar for X-axis having J2 travel	MPAS-TBAR-J2

MP-Series Integrated Multi-axis Linear Stage Life Specifications

Attribute	**Value**
Strip seal	10,000,000 cycles or 10,000 km min life in clean, dry, noncontaminating environment.
Cable track module	10,000,000 cycles minimum life.
Mechanical stop bumper	56.49 Nm (500 lb-in) potential energy.

MP-Series Integrated Multi-axis Linear Stages Motor Brake Specifications

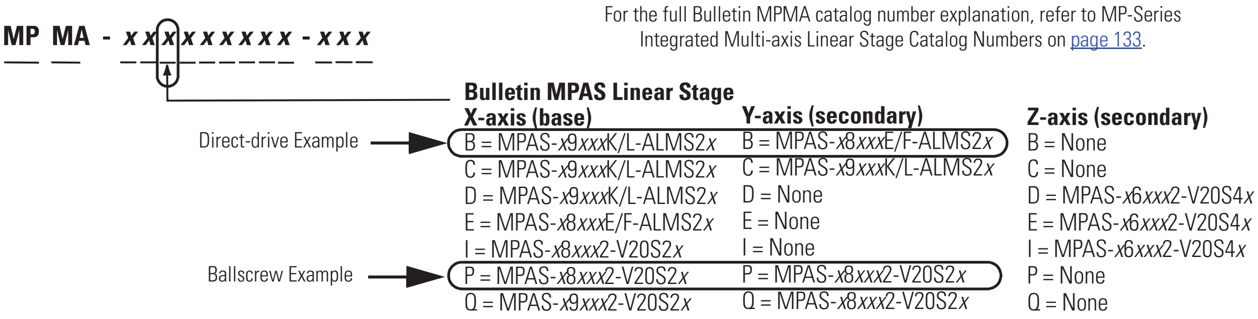
Multi-axis Linear Stage Cat. No.	**Max Backlash (brake engaged)** µm (in.)	**Holding Force** N (lb)	**Coil Current at 24V DC** A	**Brake Response Time**		
				Release ms	**Engage (using external arc suppression device)**	
					MOV ms	**Diode** ms
MPAS-A/B*xxxx*2-V20S*x*A	100 (0.004)	1294 (291)	0.46…0.56	58	20	42

MP-Series Integrated Multi-axis Linear Stage System Combinations

The Bulletin MPMA multi-axis stages are based on the individual Bulletin MPAS stages, so the Bulletin MPAS force-velocity system performance specifications also apply to the Bulletin MPMA stages. For the Bulletin MPAS linear stage performance specifications with compatible servo drives, refer to Linear Motion System Combinations, beginning on page 665.

Bulletin MPMA multi-axis stages are comprised of custom Bulletin MPAS stages. This is due to the different cable management requirements for stacked-stage assemblies. Additionally, the Bulletin MPAS individual stage identifier embedded in the Bulletin MPMA catalog number does not correspond directly to the Bulletin MPAS catalog number used in the system combinations.

Bulletin MPAS Stage Identifier



For the full Bulletin MPMA catalog number explanation, refer to MP-Series Integrated Multi-axis Linear Stage Catalog Numbers on page 133.

Bulletin MPAS Linear Stage

X-axis (base)	**Y-axis (secondary)**	**Z-axis (secondary)**
B = MPAS-*x*9*xxx*K/L-ALMS2*x*	B = MPAS-*x*8*xxx*E/F-ALMS2*x*	B = None
C = MPAS-*x*9*xxx*K/L-ALMS2*x*	C = MPAS-*x*9*xxx*K/L-ALMS2*x*	C = None
D = MPAS-*x*9*xxx*K/L-ALMS2*x*	D = None	D = MPAS-*x*6*xxx*2-V20S4*x*
E = MPAS-*x*8*xxx*E/F-ALMS2*x*	E = None	E = MPAS-*x*6*xxx*2-V20S4*x*
I = MPAS-*x*8*xxx*2-V20S2*x*	I = None	I = MPAS-*x*6*xxx*2-V20S4*x*
P = MPAS-*x*8*xxx*2-V20S2*x*	P = MPAS-*x*8*xxx*2-V20S2*x*	P = None
Q = MPAS-*x*9*xxx*2-V20S2*x*	Q = MPAS-*x*8*xxx*2-V20S2*x*	Q = None

MPAS-xxxxxx-ALM Direct-drive Linear Stage

Follow this example to determine the Bulletin MPAS stages for MPMA-AABC0C0A0-S1C multi-axis stages:

- Replace the Voltage Rating designator (*x*) with A (230V) or B (460V).
- Replace the Mechanical Drive/Motor Type designator.
  - Replace (K/L) with K if 230V or L if 460V
  - Replace (E/F) with E if 230V or F if 460V
- Replace the Cable designator (*x*), with the direct-drive standard C option.
- In this example: X-axis = MPAS-*x*9*xxx*K/L-ALMS2 and Y-axis = MPAS-*x*8*xxx*E/F-ALMS2.
  - X-axis stage = MPAS-A9*xxx*K-ALMS2C
  - Y-axis stage = MPAS-A8*xxx*E-ALMS2C

MPAS-xxxxxx-Vxx Ballscrew Linear Stage

Follow this example to determine the Bulletin MPAS stages for MPMA-BAPB8B8A0-S1C multi-axis stages:

- Replace the Voltage Rating designator (*x*) with A (230V) or B (460V).
- Holding brake (4) or no brake (2) option does not affect system performance.
- In this example: X-axis = MPAS-*x*8*xxx*2-V20S2*x* and Y-axis = MPAS-*x*8*xxx*2-V20S2*x*.
  - X-axis stage = MPAS-B8*xxx*2-V20S*x*A
  - Y-axis stage = MPAS-B8*xxx*2-V20S*x*A

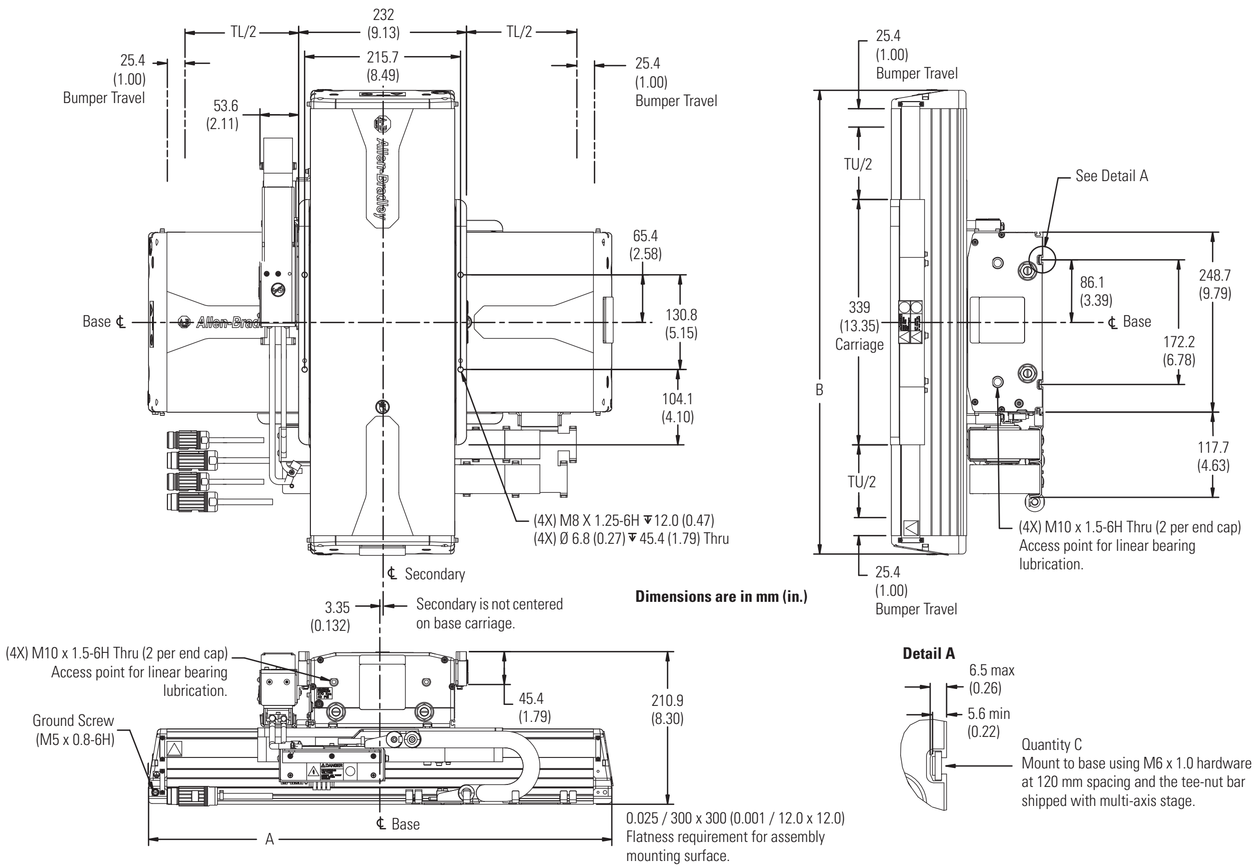
MPMA-*x*AB*xxxxxx-xxx* Product Specifications and Dimensions

These specifications apply to center stacked X/Y stages with 250 mm frame linear motor driven X-axis and 200 mm frame linear motor driven Y-axis. Maximum payload is 15 kg (33.1 lb). For heavier loads, contact your Rockwell Automation sales representative.

MPMA-*x*AB*xxxxxx-xxx* Product Specifications

Multi-axis Linear Stage Cat. No.	**Travel** mm (in.)			**Encoder Type**			**Bi-directional Repeatability** (μm)			**Weight, approx.** kg (lb)
	X-axis	**Y-axis**	**Z-axis**	**X-axis**	**Y-axis**	**Z-axis**	**X-axis**	**Y-axis**	**Z-axis**	
MPMA-*x*ABC0C0A0-S1C	200 (7.9)		N/A	5 micron resolution incremental magnetic linear encoder (direct drive only)		N/A	15	15	N/A	56.4 (124.1)
MPMA-*x*ABC6C6A0-S1C	260 (10.2)									59.2 (130.2)
MPMA-*x*ABD2D2A0-S1C	320 (12.6)									62.1 (136.6)
MPMA-*x*ABD8D8A0-S1C	380 (15.0)									64.4 (141.7)
MPMA-*x*ABE4E4A0-S1C	440 (17.3)									67.3 (148.1)
MPMA-*x*ABF6F6A0-S1C	560 (22.0)									73.1 (160.8)

MPMA-*x*AB*xxxxxx*-*xxx* Product Dimensions



Dimensions are in mm (in.)

MPMA-*x*AB*xxxxxx*-*xxx* Dimensions

Multi-axis Linear Stage Cat. No.	A Stage Length (X-axis)	TL Travel (X-axis)	B Stage Length (Y-axis)	TU Travel (Y-axis)	C Mounting Locations (X-axis)
	mm (in.)	mm (in.)	mm (in.)	mm (in.)	Qty
MPMA-AABC0C0A0-S1C	641 (25.2)	200 (7.9)	641 (25.2)	200 (7.9)	10
MPMA-AABC6C6A0-S1C	701 (27.6)	260 (10.2)	701 (27.6)	260 (10.2)	12
MPMA-AABD2D2A0-S1C	761 (30.0)	320 (12.6)	761 (30.0)	320 (12.6)	
MPMA-AABD8D8A0-S1C	821 (32.3)	380 (15.0)	821 (32.3)	380 (15.0)	14
MPMA-AABE4E4A0-S1C	881 (34.7)	440 (17.3)	881 (34.7)	440 (17.3)	
MPMA-AABF6F6A0-S1C	1001 (39.4)	560 (22.0)	1001 (39.4)	560 (22.0)	16
MPMA-BABC0C0A0-S1C	641 (25.2)	200 (7.9)	641 (25.2)	200 (7.9)	10
MPMA-BABC6C6A0-S1C	701 (27.6)	260 (10.2)	701 (27.6)	260 (10.2)	12
MPMA-BABD2D2A0-S1C	761 (30.0)	320 (12.6)	761 (30.0)	320 (12.6)	
MPMA-BABD8D8A0-S1C	821 (32.3)	380 (15.0)	821 (32.3)	380 (15.0)	14
MPMA-BABE4E4A0-S1C	881 (34.7)	440 (17.3)	881 (34.7)	440 (17.3)	
MPMA-BABF6F6A0-S1C	1001 (39.4)	560 (22.0)	1001 (39.4)	560 (22.0)	16

Actuators are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

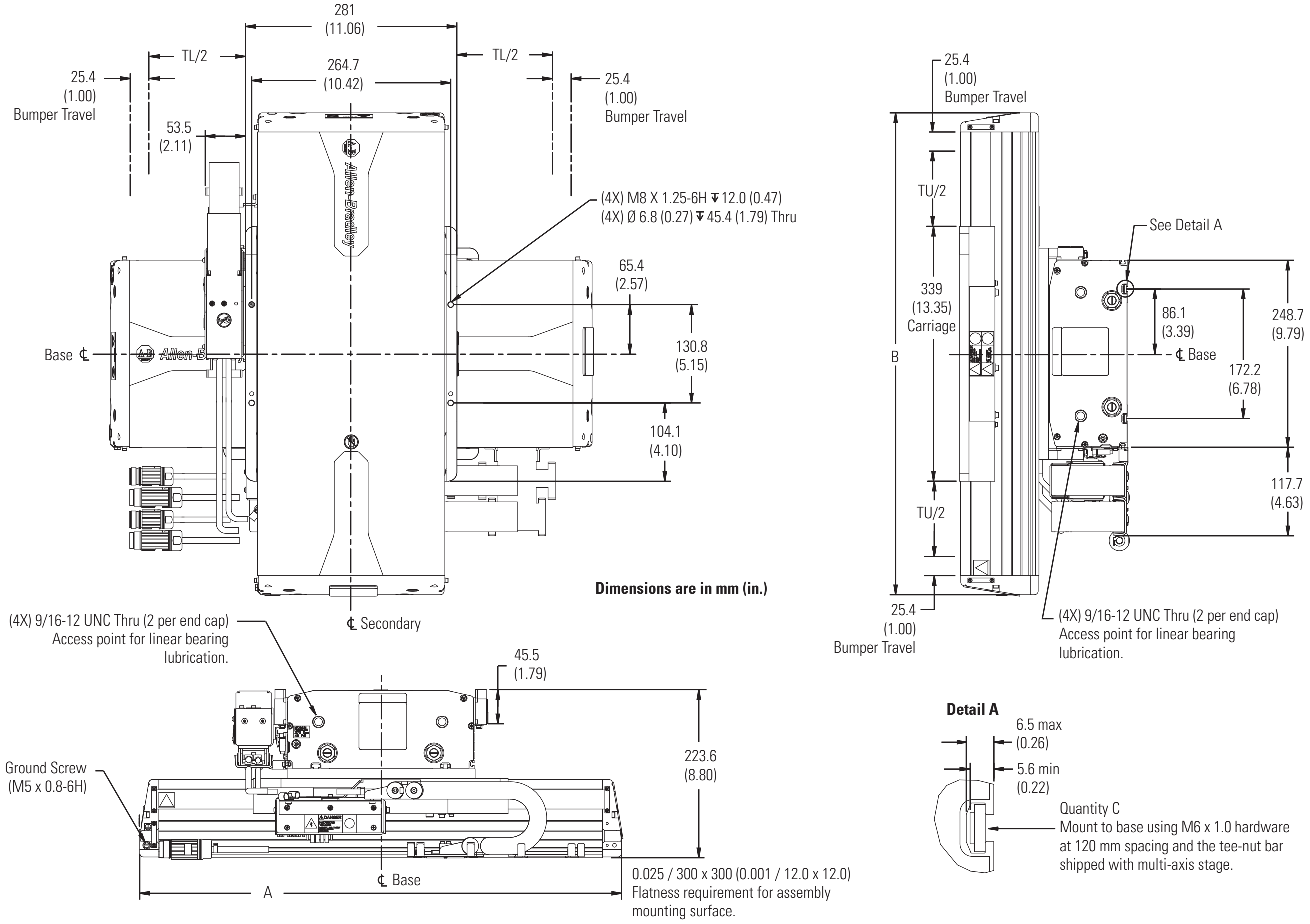
MPMA-*x*AC*xxxxxx-xxx* Product Specifications and Dimensions

These specifications apply to center stacked X/Y stages with 250 mm frame linear motor driven X-axis and 250 mm frame linear motor driven Y-axis. Maximum payload is 20 kg (44.0 lb). For heavier loads, contact your Rockwell Automation sales representative.

MPMA-*x*AC*xxxxxx-xxx* Product Specifications

Multi-axis Linear Stage Cat. No.	**Travel** mm (in.)			**Encoder Type**			**Bi-directional Repeatability** (µm)			**Weight, approx.** kg (lb)
	X-axis	**Y-axis**	**Z-axis**	**X-axis**	**Y-axis**	**Z-axis**	**X-axis**	**Y-axis**	**Z-axis**	
MPMA-*x*ACB4B4A0-S1C	140 (5.5)	140 (5.5)	N/A	5 micron resolution incremental magnetic linear encoder (direct drive only)	5 micron resolution incremental magnetic linear encoder (direct drive only)	N/A	15	15	N/A	62.2 (136.8)
MPMA-*x*ACC0C0A0-S1C	200 (7.9)	200 (7.9)								66.0 (145.2)
MPMA-*x*ACC6C6A0-S1C	260 (10.2)	260 (10.2)								69.2 (152.2)
MPMA-*x*ACD2D2A0-S1C	320 (12.6)	320 (12.6)								72.2 (158.8)

MPMA-*x*AC*xxxxxx-xxx* Product Dimensions



MPMA-*x*AC*xxxxxx-xxx* Dimensions

Multi-axis Linear Stage Cat. No.	A Stage Length (X-axis)	TL Travel (X-axis)	B Stage Length (Y-axis)	TU Travel (Y-axis)	C Mounting Locations (X-axis)
	mm (in.)	mm (in.)	mm (in.)	mm (in.)	Qty
MPMA-AACB4B4A0-S1C	581 (22.9)	140 (5.5)	581 (22.9)	140 (5.5)	10
MPMA-AACC0C0A0-S1C	641 (25.2)	200 (7.9)	641 (25.2)	200 (7.9)	
MPMA-AACC6C6A0-S1C	701 (27.6)	260 (10.2)	701 (27.6)	260 (10.2)	12
MPMA-AACD2D2A0-S1C	761 (30.0)	320 (12.6)	761 (30.0)	320 (12.6)	
MPMA-BACB4B4A0-S1C	581 (22.9)	140 (5.5)	581 (22.9)	140 (5.5)	10
MPMA-BACC0C0A0-S1C	641 (25.2)	200 (7.9)	641 (25.2)	200 (7.9)	
MPMA-BACC6C6A0-S1C	701 (27.6)	260 (10.2)	701 (27.6)	260 (10.2)	12
MPMA-BACD2D2A0-S1C	761 (30.0)	320 (12.6)	761 (30.0)	320 (12.6)	

Actuators are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

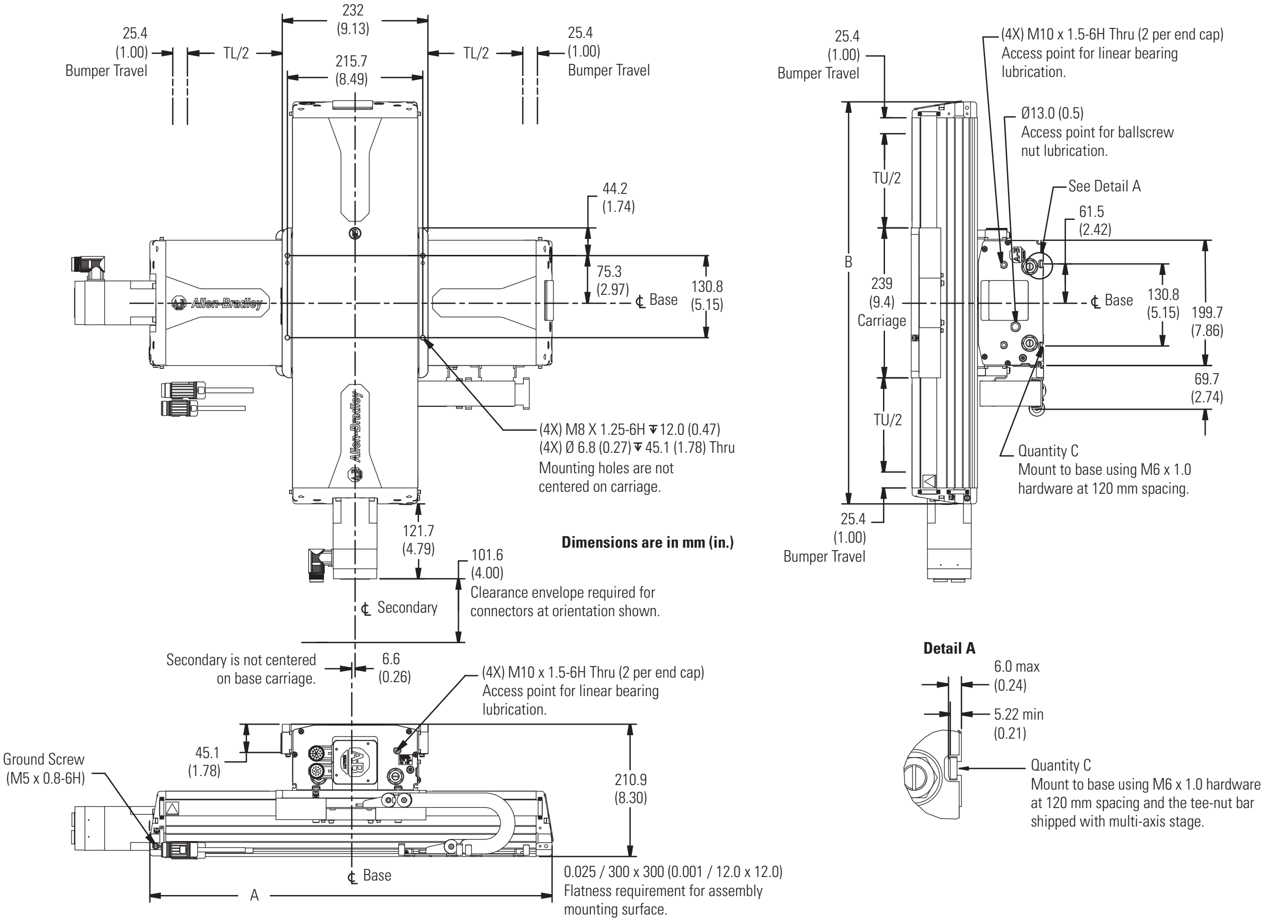
MPMA-*x*AP*xxxxxx-xxx* Product Specifications and Dimensions

These specifications apply to center stacked X/Y stages with 200 mm frame ballscrew driven X-axis and 200 mm frame ballscrew driven Y-axis. Maximum payload is 20 kg (44.0 lb). For heavier loads, contact your Rockwell Automation sales representative.

MPMA-*x*AP*xxxxxx-xxx* Product Specifications

Multi-axis Linear Stage Cat. No.	**Travel** mm (in.)			**Encoder Type**			**Bi-directional Repeatability** (µm)			**Weight, approx.** kg (lb)
	X-axis	**Y-axis**	**Z-axis**	**X-axis**	**Y-axis**	**Z-axis**	**X-axis**	**Y-axis**	**Z-axis**	
MPMA-*x*APB8B8A0-S1C	180 (7.1)		N/A	Multi-turn high resolution encoder absolute feedback, 128 cycle/rev. (ballscrew only)		N/A	60	60	N/A	36.1 (79.4)
MPMA-*x*APC4C4A0-S1C	240 (9.4)									37.9 (83.4)
MPMA-*x*APD0D0A0-S1C	300 (11.8)									39.9 (87.8)
MPMA-*x*APE2E2A0-S1C	420 (16.5)									43.9 (96.6)
MPMA-*x*APG6G6A0-S1C	660 (26.0)									51.5 (113.3)

MPMA-*xAPxxxxxx-xxx* Product Dimensions



MPMA-*xAPxxxxxx-xxx* Dimensions

Multi-axis Linear Stage Cat. No.	A Stage Length (X-axis)	TL Travel (X-axis)	B Stage Length (Y-axis)	TU Travel (Y-axis)	C Mounting Locations (X-axis)
	mm (in.)	mm (in.)	mm (in.)	mm (in.)	Qty
MPMA-AAPB8B8A0-S1C	521 (20.5)	180 (7.1)	521 (20.5)	180 (7.1)	8
MPMA-AAPC4C4A0-S1C	581 (22.9)	240 (9.4)	581 (22.9)	240 (9.4)	10
MPMA-AAPD0D0A0-S1C	641 (25.2)	300 (11.8)	641 (25.2)	300 (11.8)	
MPMA-AAPE2E2A0-S1C	761 (30.0)	420 (16.5)	761 (30.0)	420 (16.5)	12
MPMA-AAPG6G6A0-S1C	1001 (39.4)	660 (26.0)	1001 (39.4)	660 (26.0)	16
MPMA-BAPB8B8A0-S1C	521 (20.5)	180 (7.1)	521 (20.5)	180 (7.1)	8
MPMA-BAPC4C4A0-S1C	581 (22.9)	240 (9.4)	581 (22.9)	240 (9.4)	10
MPMA-BAPD0D0A0-S1C	641 (25.2)	300 (11.8)	641 (25.2)	300 (11.8)	
MPMA-BAPE2E2A0-S1C	761 (30.0)	420 (16.5)	761 (30.0)	420 (16.5)	12
MPMA-BAPG6G6A0-S1C	1001 (39.4)	660 (26.0)	1001 (39.4)	660 (26.0)	16

Actuators are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

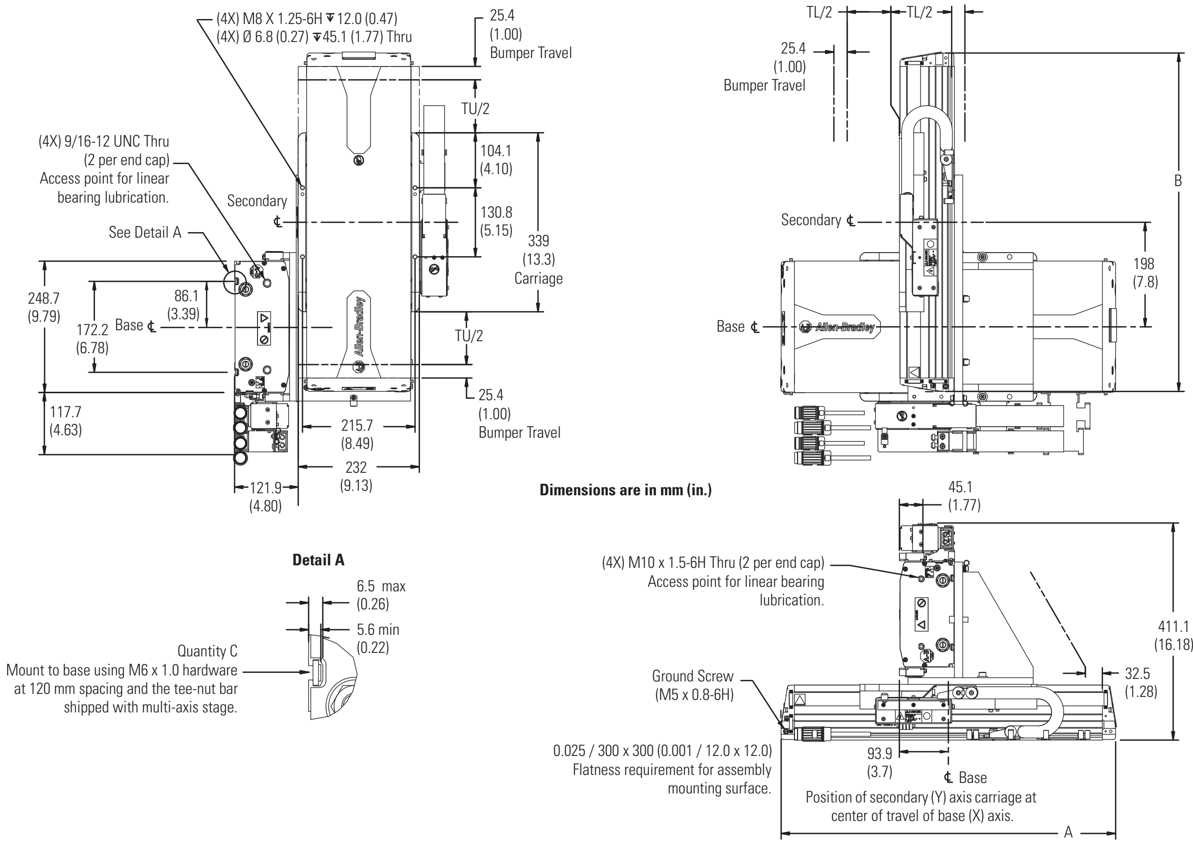
MPMA-*x*CB*xxxxxx-xxx* Product Specifications and Dimensions

These specifications apply to cartesian stacked X/Y stages with 250 mm frame linear motor driven X-axis and 200 mm frame linear motor driven Y-axis. Maximum payload is 20 kg (44.0 lb). For heavier loads, contact your Rockwell Automation sales representative.

MPMA-*x*CB*xxxxxx-xxx* Product Specifications

Multi-axis Linear Stage Cat. No.	**Travel** mm (in.)			**Encoder Type**			**Bi-directional Repeatability** (μm)			**Weight, approx.** kg (lb)
	X-axis	**Y-axis**	**Z-axis**	**X-axis**	**Y-axis**	**Z-axis**	**X-axis**	**Y-axis**	**Z-axis**	
MPMA-*x*CBD2D2A0-S1C	320 (12.6)	320 (12.6)	N/A	5 micron resolution incremental magnetic linear encoder (direct drive only)		N/A	15	15	N/A	69.7 (153.3)
MPMA-*x*CBE4D2A0-S1C	440 (17.3)									72.7 (159.9)
MPMA-*x*CBE4E4A0-S1C		440 (17.3)								74.9 (164.7)
MPMA-*x*CBF6E4A0-S1C	560 (22.0)									78.4 (172.4)
MPMA-*x*CBG8E4A0-S1C	680 (26.8)									81.4 (179.0)
MPMA-*x*CBG8F6A0-S1C		560 (22.0)								83.7 (184.1)
MPMA-*x*CBI0F6A0-S1C	800 (31.5)									87.2 (191.8)
MPMA-*x*CBJ2F6A0-S1C	920 (36.2)									90.3 (198.6)

MPMA-*x*CB*xxxxxx-xxx* Product Dimensions



MPMA-*x*CB*xxxxxx-xxx* Dimensions

Multi-axis Linear Stage Cat. No.	A Stage Length (X-axis)	TL Travel (X-axis)	B Stage Length (Y-axis)	TU Travel (Y-axis)	C Mounting Locations (X-axis)
	mm (in.)	mm (in.)	mm (in.)	mm (in.)	Qty
MPMA-ACBD2D2A0-S1C	761 (30.0)	320 (12.6)	761 (30.0)	320 (12.6)	12
MPMA-ACBE4D2A0-S1C	881 (34.7)	440 (17.3)			14
MPMA-ACBE4E4A0-S1C			881 (34.7)	440 (17.3)	
MPMA-ACBF6E4A0-S1C	1001 (39.4)	560 (22.0)			16
MPMA-ACBG8E4A0-S1C	1121 (44.1)	680 (26.8)			18
MPMA-ACBG8F6A0-S1C			1001 (39.4)	560 (22.0)	
MPMA-ACBI0F6A0-S1C	1241 (48.9)	800 (31.5)			20
MPMA-ACBJ2F6A0-S1C	1361 (53.6)	920 (36.2)			22
MPMA-BCBD2D2A0-S1C	761 (30.0)	320 (12.6)	761 (30.0)	320 (12.6)	12
MPMA-BCBE4D2A0-S1C	881 (34.7)	440 (17.3)			14
MPMA-BCBE4E4A0-S1C			881 (34.7)	440 (17.3)	
MPMA-BCBF6E4A0-S1C	1001 (39.4)	560 (22.0)			16
MPMA-BCBG8E4A0-S1C	1121 (44.1)	680 (26.8)			18
MPMA-BCBG8F6A0-S1C			1001 (39.4)	560 (22.0)	
MPMA-BCBI0F6A0-S1C	1241 (48.9)	800 (31.5)			20
MPMA-BCBJ2F6A0-S1C	1361 (53.6)	920 (36.2)			22

Actuators are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

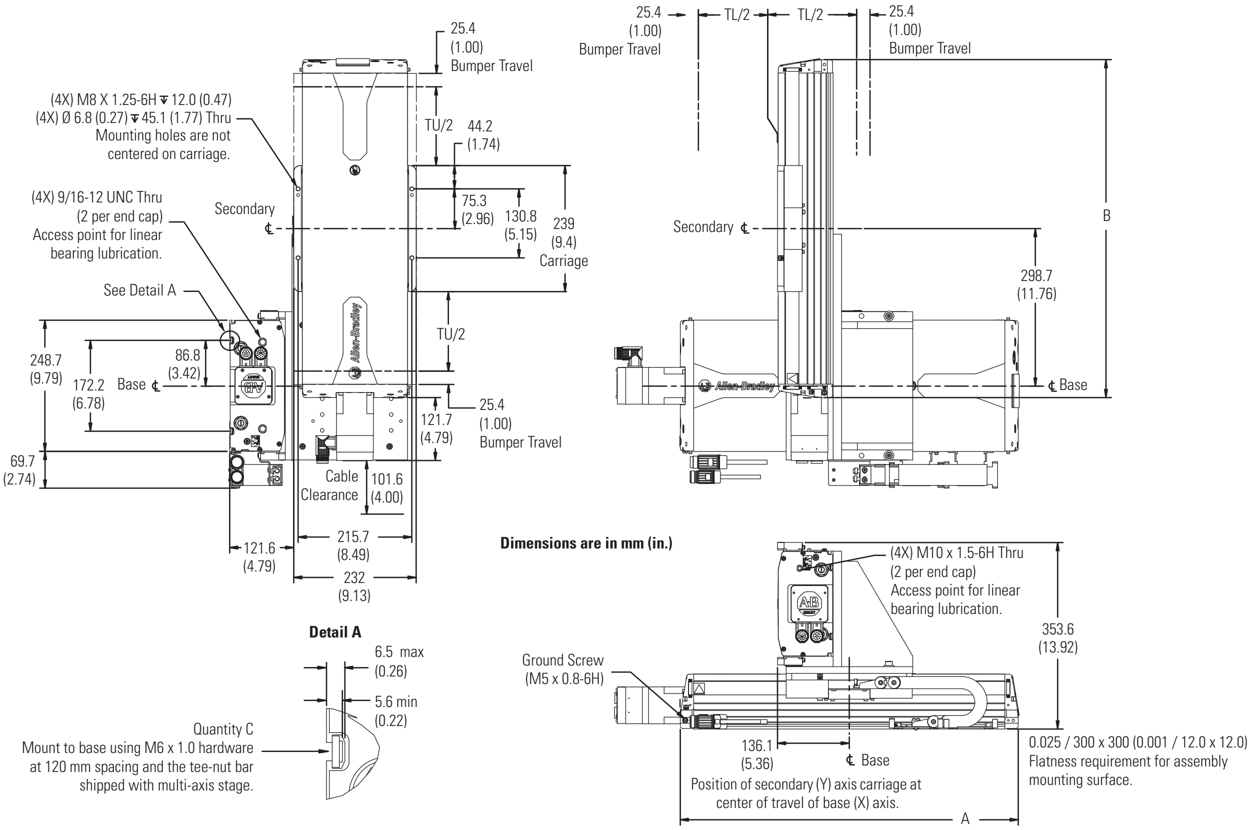
MPMA-*x*CQ*xxxxxx-xxx* Product Specifications and Dimensions

These specifications apply to cartesian stacked X/Y stages with 250 mm frame ballscrew driven X-axis and 200 mm frame ballscrew driven Y-axis. Maximum payload is 25 kg (55.1 lb). For heavier loads, contact your Rockwell Automation sales representative.

MPMA-*x*CQ*xxxxxx-xxx* Product Specifications

Multi-axis Linear Stage Cat. No.	**Travel** mm (in.)			**Encoder Type**			**Bi-directional Repeatability** (µm)			**Weight, approx.** kg (lb)
	X-axis	**Y-axis**	**Z-axis**	**X-axis**	**Y-axis**	**Z-axis**	**X-axis**	**Y-axis**	**Z-axis**	
MPMA-*x*CQD0D0A0-S1C	300 (11.8)	300 (11.8)	N/A	Multi-turn high resolution encoder absolute feedback, 128 cycle/rev. (ballscrew only)		N/A	60	60	N/A	51.7 (113.7)
MPMA-*x*CQE2D0A0-S1C	420 (16.5)									54.3 (119.4)
MPMA-*x*CQE2E2A0-S1C		420 (16.5)								56.3 (123.8)
MPMA-*x*CQG6E2A0-S1C	660 (26.0)									61.4 (135.0)
MPMA-*x*CQG6G6A0-S1C		660 (26.0)								65.2 (143.4)
MPMA-*x*CQH8G6A0-S1C	780 (30.7)									67.7 (148.9)
MPMA-*x*CQJ0G6A0-S1C	900 (35.4)									70.2 (154.4)

MPMA-*x*CQ*xxxxxx-xxx* Product Dimensions



MPMA-*x*CQ*xxxxxx-xxx* Dimensions

Multi-axis Linear Stage Cat. No.	A Stage Length (X-axis)	TL Travel (X-axis)	B Stage Length (Y-axis)	TU Travel (Y-axis)	C Mounting Locations (X-axis)
	mm (in.)	mm (in.)	mm (in.)	mm (in.)	Qty
MPMA-ACQD0D0A0-S1C	641 (25.2)	300 (11.8)	641 (25.2)	300 (11.8)	10
MPMA-ACQE2D0A0-S1C	761 (30.0)	420 (16.5)			12
MPMA-ACQE2E2A0-S1C			761 (30.0)	420 (16.5)	
MPMA-ACQG6E2A0-S1C	1001 (39.4)	660 (26.0)			16
MPMA-ACQG6G6A0-S1C			1001 (39.4)	660 (26.0)	
MPMA-ACQH8G6A0-S1C	1121 (44.1)	780 (30.7)			18
MPMA-ACQJ0G6A0-S1C	1241 (48.9)	900 (35.4)			20
MPMA-BCQD0D0A0-S1C	641 (25.2)	300 (11.8)	641 (25.2)	300 (11.8)	10
MPMA-BCQE2D0A0-S1C	761 (30.0)	420 (16.5)			12
MPMA-BCQE2E2A0-S1C			761 (30.0)	420 (16.5)	
MPMA-BCQG6E2A0-S1C	1001 (39.4)	660 (26.0)			16
MPMA-BCQG6G6A0-S1C			1001 (39.4)	660 (26.0)	
MPMA-BCQH8G6A0-S1C	1121 (44.1)	780 (30.7)			18
MPMA-BCQJ0G6A0-S1C	1241 (48.9)	900 (35.4)			20

Actuators are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

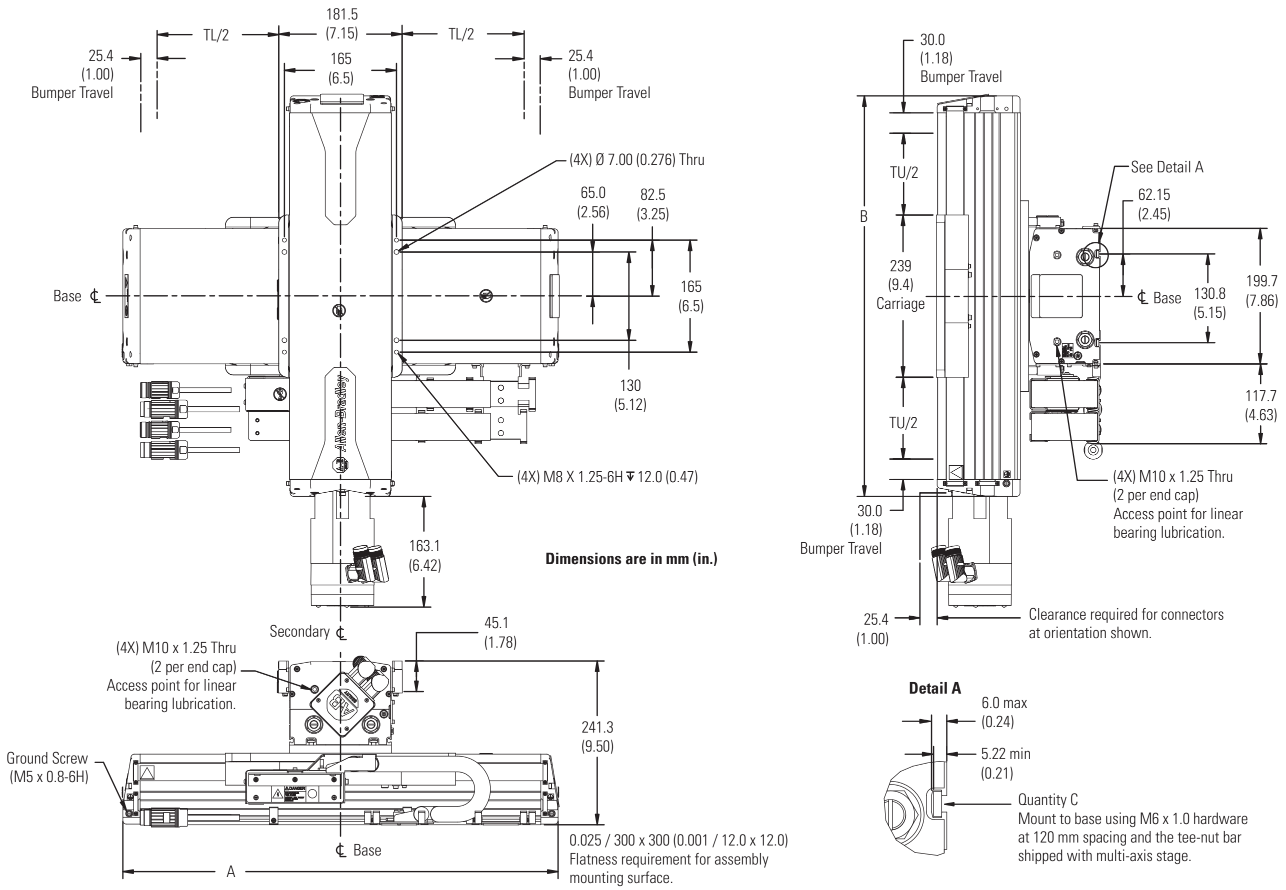
MPMA-*x*BE*xxxxxx-xxx* Product Specifications and Dimensions

These specifications apply to center stacked X/Z stages with 200 mm frame linear motor driven X-axis and 150 mm frame ballscrew driven Z-axis. Maximum payload is 20 kg (44.0 lb). For heavier loads, contact your Rockwell Automation sales representative.

MPMA-*x*BE*xxxxxx-xxx* Product Specifications

Multi-axis Linear Stage Cat. No.	**Travel** mm (in.)			**Encoder Type**			**Bi-directional Repeatability** (μm)			**Weight, approx.** kg (lb)
	X-axis	**Y-axis**	**Z-axis**	**X-axis**	**Y-axis**	**Z-axis**	**X-axis**	**Y-axis**	**Z-axis**	
MPMA-*x*BED2A0B8-S2C	320 (12.6)	N/A	180 (7.1)	5 micron resolution incremental magnetic linear encoder (direct drive only)	N/A	Multi-turn high resolution encoder absolute feedback, 128 cycle/rev. (ballscrew only)	15	N/A	60	44.5 (98.0)
MPMA-*x*BED2A0D0-S2C			300 (11.8)							46.2 (101.7)
MPMA-*x*BEE4A0B8-S2C	440 (17.3)		180 (7.1)							46.7 (102.8)
MPMA-*x*BEE4A0D0-S2C			300 (11.8)							48.4 (106.6)
MPMA-*x*BEG8A0B8-S2C	680 (26.8)		180 (7.1)							51.2 (112.7)
MPMA-*x*BEG8A0D0-S2C			300 (11.8)							52.9 (116.9)

MPMA-*x*BE*xxxxxx-xxx* Product Dimensions



MPMA-*x*BE*xxxxxx-xxx* Dimensions

Multi-axis Linear Stage Cat. No.	A Stage Length (X-axis)	TL Travel (X-axis)	B Stage Length (Z-axis)	TU Travel (Z-axis)	C Mounting Locations (X-axis)
	mm (in.)	mm (in.)	mm (in.)	mm (in.)	Qty
MPMA-ABED2A0B8-S2C	761 (30.0)	320 (12.6)	530 (20.9)	180 (7.1)	12
MPMA-ABED2A0D0-S2C			650 (25.6)	300 (11.8)	
MPMA-ABEE4A0B8-S2C	881 (34.7)	440 (17.3)	530 (20.9)	180 (7.1)	14
MPMA-ABEE4A0D0-S2C			650 (25.6)	300 (11.8)	
MPMA-ABEG8A0B8-S2C	1121 (44.1)	680 (26.8)	530 (20.9)	180 (7.1)	18
MPMA-ABEG8A0D0-S2C			650 (25.6)	300 (11.8)	
MPMA-BBED2A0B8-S2C	761 (30.0)	320 (12.6)	530 (20.9)	180 (7.1)	12
MPMA-BBED2A0D0-S2C			650 (25.6)	300 (11.8)	
MPMA-BBEE4A0B8-S2C	881 (34.7)	440 (17.3)	530 (20.9)	180 (7.1)	14
MPMA-BBEE4A0D0-S2C			650 (25.6)	300 (11.8)	
MPMA-BBEG8A0B8-S2C	1121 (44.1)	680 (26.8)	530 (20.9)	180 (7.1)	18
MPMA-BBEG8A0D0-S2C			650 (25.6)	300 (11.8)	

Actuators are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

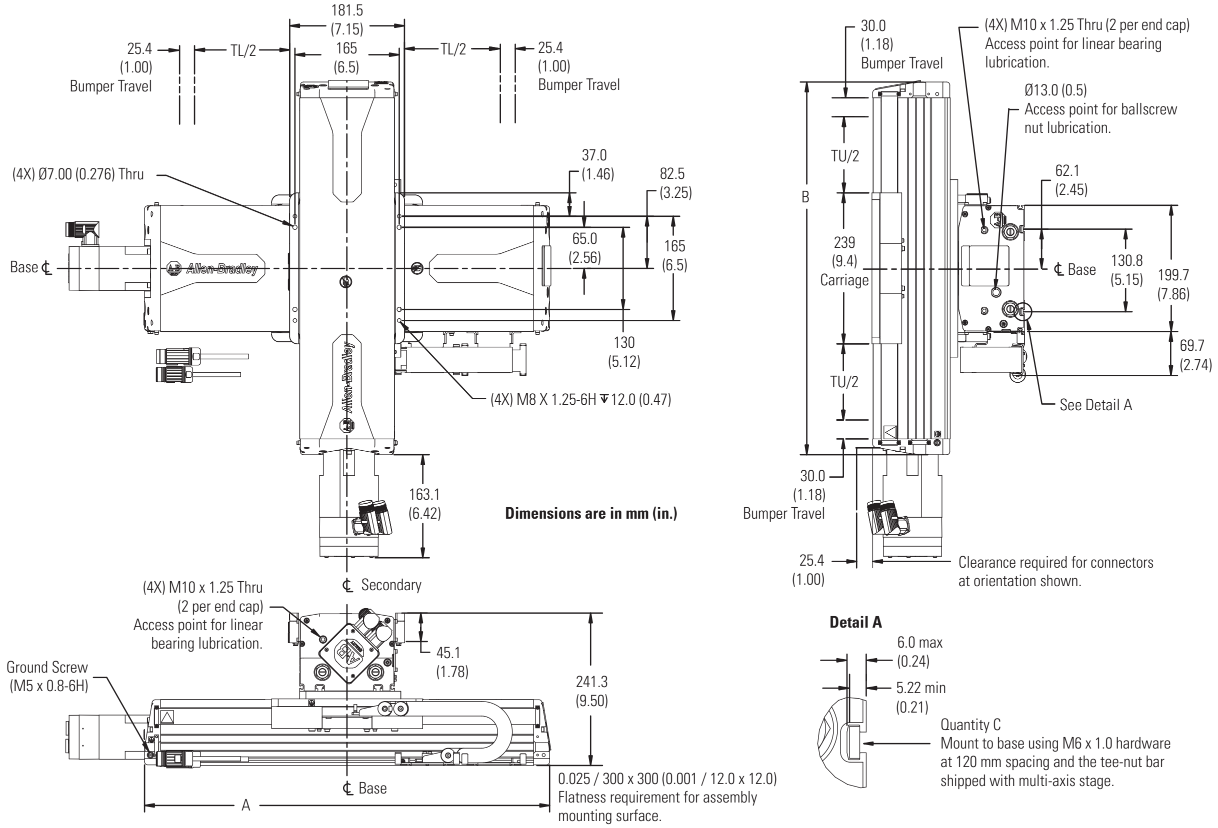
MPMA-*x*BI*xxxxxx-xxx* Product Specifications and Dimensions

These specifications apply to center stacked X/Z stages with 200 mm frame ballscrew driven X-axis and 150 mm frame ballscrew driven Z-axis. Maximum payload is 25 kg (55.1 lb). For heavier loads, contact your Rockwell Automation sales representative.

MPMA-*x*BI*xxxxxx-xxx* Product Specifications

Multi-axis Linear Stage Cat. No.	**Travel** mm (in.)			**Encoder Type**			**Bi-directional Repeatability** (µm)			**Weight, approx.** kg (lb)
	X-axis	**Y-axis**	**Z-axis**	**X-axis**	**Y-axis**	**Z-axis**	**X-axis**	**Y-axis**	**Z-axis**	
MPMA-*x*BID0A0B8-S2C	300 (11.8)	N/A	180 (7.1)	Multi-turn high resolution encoder absolute feedback, 128 cycle/rev. (ballscrew only)	N/A	Multi-turn high resolution encoder absolute feedback, 128 cycle/rev. (ballscrew only)	60	N/A	60	37.7 (83.0)
MPMA-*x*BID0A0D0-S2C			300 (11.8)							39.4 (86.8)
MPMA-*x*BIE2A0B8-S2C	420 (16.5)		180 (7.1)							39.7 (87.4)
MPMA-*x*BIE2A0D0-S2C			300 (11.8)							41.4 (91.2)
MPMA-*x*BIG6A0B8-S2C	660 (26.0)		180 (7.1)							43.5 (95.8)
MPMP-*x*BIG6A0D0-S2C			300 (11.8)							45.2 (99.5)

MPMA-*x*BI*xxxxxx*-*xxx* Product Dimensions



Dimensions are in mm (in.)

MPMA-*x*BI*xxxxxx*-*xxx* Dimensions

Multi-axis Linear Stage Cat. No.	A Stage Length (X-axis)	TL Travel (X-axis)	B Stage Length (Z-axis)	TU Travel (Z-axis)	C Mounting Locations (X-axis)
	mm (in.)	mm (in.)	mm (in.)	mm (in.)	Qty
MPMA-ABID0A0B8-S2C	641 (25.2)	300 (11.8)	530 (20.9)	180 (7.1)	10
MPMA-ABID0A0D0-S2C			650 (25.6)	300 (11.8)	
MPMA-ABIE2A0B8-S2C	761 (30.0)	420 (16.5)	530 (20.9)	180 (7.1)	12
MPMA-ABIE2A0D0-S2C			650 (25.6)	300 (11.8)	
MPMA-ABIG6A0B8-S2C	1001 (39.4)	660 (26.0)	530 (20.9)	180 (7.1)	16
MPMA-ABIG6A0D0-S2C			650 (25.6)	300 (11.8)	
MPMA-BBID0A0B8-S2C	641 (25.2)	300 (11.8)	530 (20.9)	180 (7.1)	10
MPMA-BBID0A0D0-S2C			650 (25.6)	300 (11.8)	
MPMA-BBIE2A0B8-S2C	761 (30.0)	420 (16.5)	530 (20.9)	180 (7.1)	12
MPMA-BBIE2A0D0-S2C			650 (25.6)	300 (11.8)	
MPMA-BBIG6A0B8-S2C	1001 (39.4)	660 (26.0)	530 (20.9)	180 (7.1)	16
MPMA-BBIG6A0D0-S2C			650 (25.6)	300 (11.8)	

Actuators are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

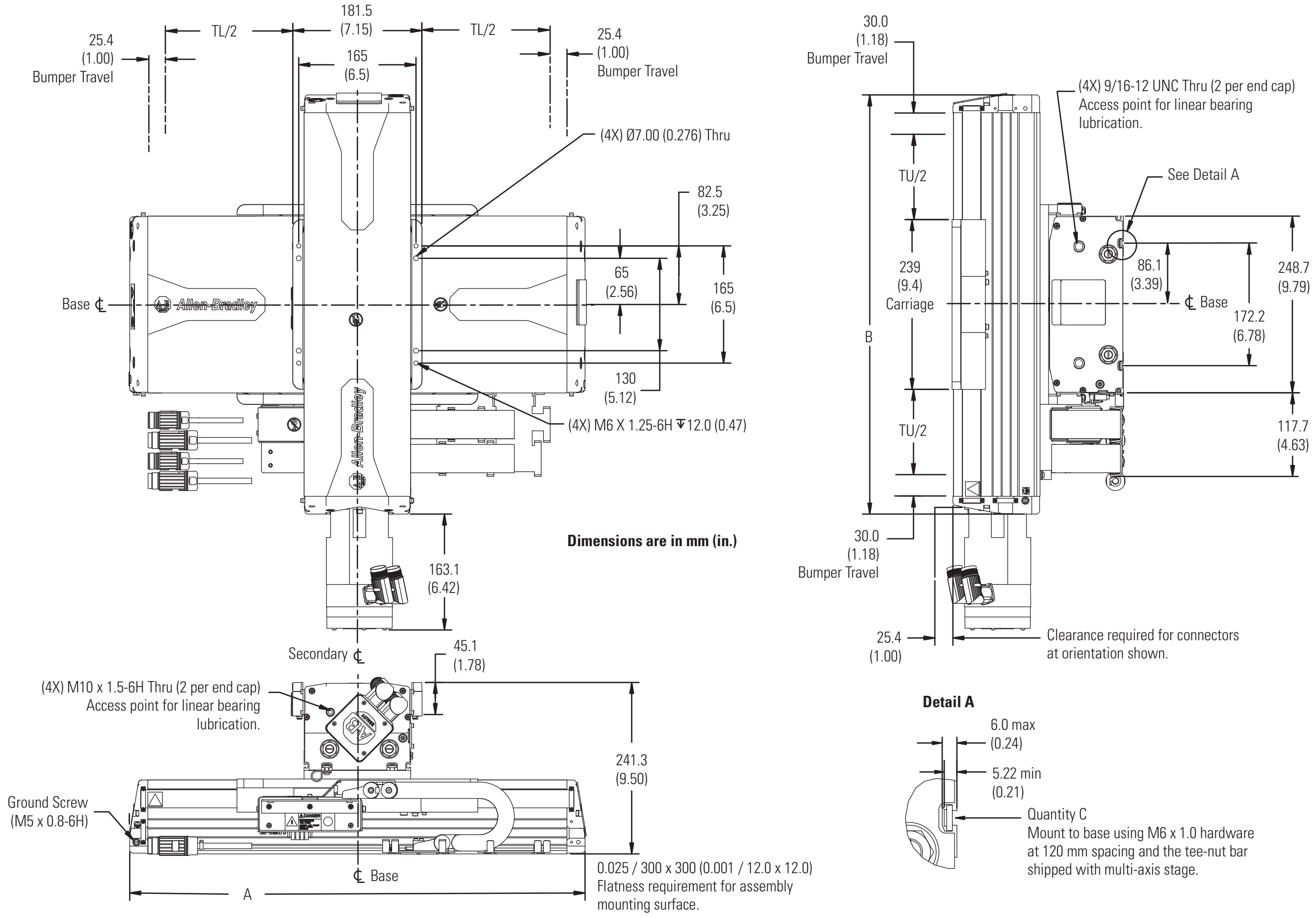
MPMA-*x*BD*xxxxxx-xxx* Product Specifications and Dimensions

These specifications apply to center stacked X/Z stages with 250 mm frame linear motor driven X-axis and 150 mm frame ballscrew driven Z-axis. Maximum payload is 20 kg (44.0 lb). For heavier loads, contact your Rockwell Automation sales representative.

MPMA-*x*BD*xxxxxx-xxx* Product Specifications

Multi-axis Linear Stage Cat. No.	Travel mm (in.)			Encoder Type			Bi-directional Repeatability (µm)			Weight, approx. kg (lb)
	X-axis	Y-axis	Z-axis	X-axis	Y-axis	Z-axis	X-axis	Y-axis	Z-axis	
MPMA-*x*BDD2A0B8-S2C	320 (12.6)	N/A	180 (7.1)	5 micron resolution incremental magnetic linear encoder (direct drive only)	N/A	Multi-turn high resolution encoder absolute feedback, 128 cycle/rev. (ballscrew only)	15	N/A	60	52.2 (114.9)
MPMA-*x*BDD2A0D0-S2C			300 (11.8)							53.9 (118.7)
MPMA-*x*BDE4A0B8-S2C	440 (17.3)		180 (7.1)							55.2 (121.5)
MPMA-*x*BDE4A0D0-S2C			300 (11.8)							56.9 (125.3)
MPMA-*x*BDE4A0E2-S2C			420 (16.5)							58.6 (129.0)
MPMA-*x*BDG8A0B8-S2C	680 (26.8)		180 (7.1)							61.7 (135.8)
MPMA-*x*BDG8A0D0-S2C			300 (11.8)							63.4 (139.6)
MPMA-*x*BDG8A0E2-S2C			420 (16.5)							65.1 (143.3)
MPMA-*x*BDI0A0B8-S2C	800 (31.5)		180 (7.1)							65.2 (143.5)
MPMA-*x*BDI0A0D0-S2C			300 (11.8)							66.9 (147.3)
MPMA-*x*BDI0A0E2-S2C			420 (16.5)							68.6 (151.0)
MPMA-*x*BDJ2A0D0-S2C	920 (36.2)		300 (11.8)							70.0 (154.1)
MPMA-*x*BDJ2A0E2-S2C			420 (16.5)							71.7 (157.8)
MPMA-*x*BDJ2A0F4-S2C			540 (21.3)							73.5 (161.8)

MPMA-*x*BD*xxxxxx*-*xxx* Product Dimensions



MPMA-*x*BD*xxxxxx-xxx* Dimensions

Multi-axis Linear Stage Cat. No.	A Stage Length (X-axis)	TL Travel (X-axis)	B Stage Length (Z-axis)	TU Travel (Z-axis)	C Mounting Locations (X-axis)
	mm (in.)	mm (in.)	mm (in.)	mm (in.)	Qty
MPMA-ABDD2A0B8-S2C	761 (30.0)	320 (12.6)	530 (20.9)	180 (7.1)	12
MPMA-ABDD2A0D0-S2C			650 (25.6)	300 (11.8)	
MPMA-ABDE4A0B8-S2C	881 (34.7)	440 (17.3)	530 (20.9)	180 (7.1)	14
MPMA-ABDE4A0D0-S2C			650 (25.6)	300 (11.8)	
MPMA-ABDE4A0E2-S2C			770 (30.3)	420 (16.5)	
MPMA-ABDG8A0B8-S2C	1121 (44.1)	680 (26.8)	530 (20.9)	180 (7.1)	18
MPMA-ABDG8A0D0-S2C			650 (25.6)	300 (11.8)	
MPMA-ABDG8A0E2-S2C			770 (30.3)	420 (16.5)	
MPMA-ABDI0A0B8-S2C	1241 (48.9)	800 (31.5)	530 (20.9)	180 (7.1)	20
MPMA-ABDI0A0D0-S2C			650 (25.6)	300 (11.8)	
MPMA-ABDI0A0E2-S2C			770 (30.3)	420 (16.5)	
MPMA-ABDJ2A0D0-S2C	1361 (53.6)	920 (36.2)	650 (25.6)	300 (11.8)	22
MPMA-ABDJ2A0E2-S2C			770 (30.3)	420 (16.5)	
MPMA-ABDJ2A0F4-S2C			890 (35.0)	540 (21.3)	
MPMA-BBDD2A0B8-S2C	761 (30.0)	320 (12.6)	530 (20.9)	180 (7.1)	12
MPMA-BBDD2A0D0-S2C			650 (25.6)	300 (11.8)	
MPMA-BBDE4A0B8-S2C	881 (34.7)	440 (17.3)	530 (20.9)	180 (7.1)	14
MPMA-BBDE4A0D0-S2C			650 (25.6)	300 (11.8)	
MPMA-BBDE4A0E2-S2C			770 (30.3)	420 (16.5)	
MPMA-BBDG8A0B8-S2C	1121 (44.1)	680 (26.8)	530 (20.9)	180 (7.1)	18
MPMA-BBDG8A0D0-S2C			650 (25.6)	300 (11.8)	
MPMA-BBDG8A0E2-S2C			770 (30.3)	420 (16.5)	
MPMA-BBDI0A0B8-S2C	1241 (48.9)	800 (31.5)	530 (20.9)	180 (7.1)	20
MPMA-BBDI0A0D0-S2C			650 (25.6)	300 (11.8)	
MPMA-BBDI0A0E2-S2C			770 (30.3)	420 (16.5)	
MPMA-BBDJ2A0D0-S2C	1361 (53.6)	920 (36.2)	650 (25.6)	300 (11.8)	22
MPMA-BBDJ2A0E2-S2C			770 (30.3)	420 (16.5)	
MPMA-BBDJ2A0F4-S2C			890 (35.0)	540 (21.3)	

Actuators are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

MP-Series Integrated Multi-axis Linear Stage Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your actuator. For questions regarding product availability, contact your Allen-Bradley distributor.

MP MA - *x x x x x x x x x* - *x x x*

Cable
C = Cable track module with IP67 SpeedTec DIN connectors (applies to all Bulletin MPMA stages)

Limits and Brake Options

X-axis	**Y-axis**	**Z-axis**
1 = No brake, no limits	1 = No brake, no limits	1 = None
2 = No brake, no limits	2 = None	2 = 24V brake, no limits (ballscrew only)

Protection Options (same as Bulletin MPAS options)
S = Covered with strip seals (applies to all Bulletin MPMA stages)

Z-axis 10s of mm
0 = 0 mm 2 = 20 mm 4 = 40 mm 6 = 60 mm 8 = 80 mm
1 = 10 mm 3 = 30 mm 5 = 50 mm 7 = 70 mm 9 = 90 mm

Z-axis 100s of mm
A = 0 mm C = 200 mm E = 400 mm
B = 100 mm D = 300 mm F = 500 mm

Y-axis 10s of mm
0 = 0 mm 2 = 20 mm 4 = 40 mm 6 = 60 mm 8 = 80 mm
1 = 10 mm 3 = 30 mm 5 = 50 mm 7 = 70 mm 9 = 90 mm

Y-axis 100s of mm
A = 0 mm C = 200 mm E = 400 mm G = 600 mm
B = 100 mm D = 300 mm F = 500 mm

X-axis 10s of mm
0 = 0 mm 2 = 20 mm 4 = 40 mm 6 = 60 mm 8 = 80 mm
1 = 10 mm 3 = 30 mm 5 = 50 mm 7 = 70 mm 9 = 90 mm

X-axis 100s of mm
A = 0 mm C = 200 mm E = 400 mm G = 600 mm I = 800 mm
B = 100 mm D = 300 mm F = 500 mm H = 700 mm J = 900 mm

Bulletin MPAS Linear Stage

X-axis (base)	**Y-axis (secondary)**	**Z-axis (secondary)**
B = MPAS-*x9xxx*K/L-ALMS2*x*	B = MPAS-*x8xxx*F/E-ALMS2*x*	B = None
C = MPAS-*x9xxx*K/L-ALMS2*x*	C = MPAS-*x9xxx*K/L-ALMS2*x*	C = None
D = MPAS-*x9xxx*K/L-ALMS2*x*	D = None	D = MPAS-*x6xxx*2-V20S4*x*
E = MPAS-*x8xxx*F/E-ALMS2*x*	E = None	E = MPAS-*x6xxx*2-V20S4*x*
I = MPAS-*x8xxx*2-V20S2*x*	I = None	I = MPAS-*x6xxx*2-V20S4*x*
P = MPAS-*x8xxx*2-V20S2*x*	P = MPAS-*x8xxx*2-V20S2*x*	P = None
Q = MPAS-*x9xxx*2-V20S2*x*	Q = MPAS-*x8xxx*2-V20S2*x*	Q = None

Configuration
A = Center stacked XY
B = Center stacked XZ
C = Cartesian stacked XY (left-side cantilever)

Voltage Class
A = 200V
B = 400V

Actuator Type
MA = Actuator, multi-axis linear stage

Actuator Series
MP = MP-Series

MP-Series and TL-Series Electric Cylinders

MP-Series Electric Cylinders

TL-Series Electric Cylinders

With the MP-Series and TL-Series Electric Cylinders, your applications will experience flexible servo control ideal for solutions requiring forces to be built up quickly and positions that need to be approached accurately. Available in three ISO 15552 pneumatic-class frame sizes (32, 40, and 63), these durable, quiet, and energy-efficient non-rotating stainless steel piston rod actuators are an excellent upgrade for pneumatic systems.

With the ability to synchronize and coordinate with multi-axis machine motions, the MP-Series and TL-Series cylinders provide a dynamic, precise response for a wide range of linear motion applications. When seamlessly integrated into the Rockwell Automation Integrated Architecture, MP-Series and TL-Series electric cylinders use RSLogix 5000 software to extend and retract with precise positioning, velocity or force. In Force mode, an electric cylinder continues to push the load in a manner similar to an air cylinder, but with the convenience of a limit established by using state-of-the art software.

For drive compatibility, refer to Servo Drives on page 14.

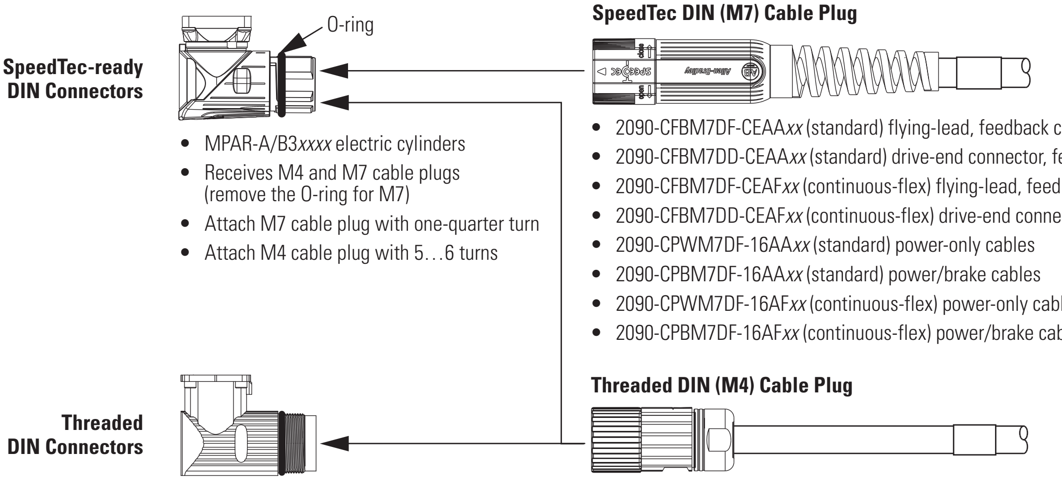
MP-Series and TL-Series Electric Cylinder Features

- Fully assembled and ready to mount cylinders contribute to reductions in mechanical design engineering, assembly, wiring, and commissioning time.
- State-of-the-art design features ballscrew construction with linear stroke lengths up to 800 mm (32 in.), absolute high-resolution feedback, and speeds up to 1.0 m/s (39.4 in./s).
- Operates without externally mounted limit or home switches and retains position during power loss for faster machine restart.
- Rated for 100% duty cycle and designed for repeatable, reproducible performance over the actuator's operating life.
- Linear feed force of up to 2500 N (562 lb).
- Positioning repeatability of ±0.02 mm (0.0008 in.).
- No piping, valving, air, or oil supply required.
- Sizing and engineering with compatible servo drives is made easy with Motion Analyzer software and online CAD files.
- Commissioning is simplified by the use of standard Allen-Bradley motor power and feedback cables.
- Full set-up and programming support by using Allen-Bradley Logix Controller platforms. RSLogix 5000 and Ultraware software make setup and commissioning fast and easy.
- Optional 24V DC holding brakes.

Motor Connector/Cable Compatibility

For linear motors and actuators with SpeedTec-ready DIN connectors, standard (non-flex) and continuous-flex cables are available. However, motors/actuators with SpeedTec-ready DIN connectors and incremental encoders must use threaded DIN (M4) feedback cables in non-flex applications. This is due to the number of conductors required for feedback connections.

MP-Series (Bulletin MPAR) electric cylinders are equipped with threaded DIN and SpeedTec DIN connectors.



SpeedTec-ready DIN Connectors

- MPAR-A/B3*xxxx* electric cylinders
- Receives M4 and M7 cable plugs (remove the O-ring for M7)
- Attach M7 cable plug with one-quarter turn
- Attach M4 cable plug with 5…6 turns

SpeedTec DIN (M7) Cable Plug

- 2090-CFBM7DF-CEAA*xx* (standard) flying-lead, feedback cables
- 2090-CFBM7DD-CEAA*xx* (standard) drive-end connector, feedback cables
- 2090-CFBM7DF-CEAF*xx* (continuous-flex) flying-lead, feedback cables
- 2090-CFBM7DD-CEAF*xx* (continuous-flex) drive-end connector, feedback cables
- 2090-CPWM7DF-16AA*xx* (standard) power-only cables
- 2090-CPBM7DF-16AA*xx* (standard) power/brake cables
- 2090-CPWM7DF-16AF*xx* (continuous-flex) power-only cables
- 2090-CPBM7DF-16AF*xx* (continuous-flex) power/brake cables

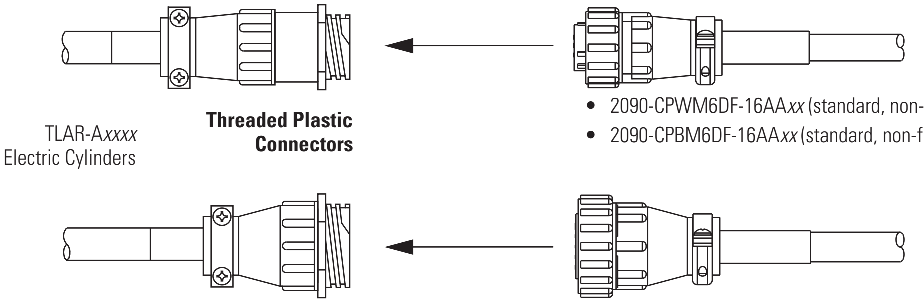
Threaded DIN Connectors

- MPAR-A/B1*xxxx* electric cylinders
- MPAR-A/B2*xxxx* electric cylinders
- Receives M4 cable plugs only
- Attach M4 cable plug with 5…6 turns

Threaded DIN (M4) Cable Plug

- 2090-XXNFMF-S*xx* (standard) feedback cables
- 2090-CFBM4DF-CDAF*xx* (continuous-flex) feedback cables
- 2090-XXNPMF-S*xx* (standard) power cables
- 2090-CPWM4DF-16AF*xx* (continuous-flex) power-only cables
- 2090-CPBM4DF-16AF*xx* (continuous-flex) power-with-brake cables

TL-Series (Bulletin TLAR) electric cylinders are equipped with circular plastic connectors.

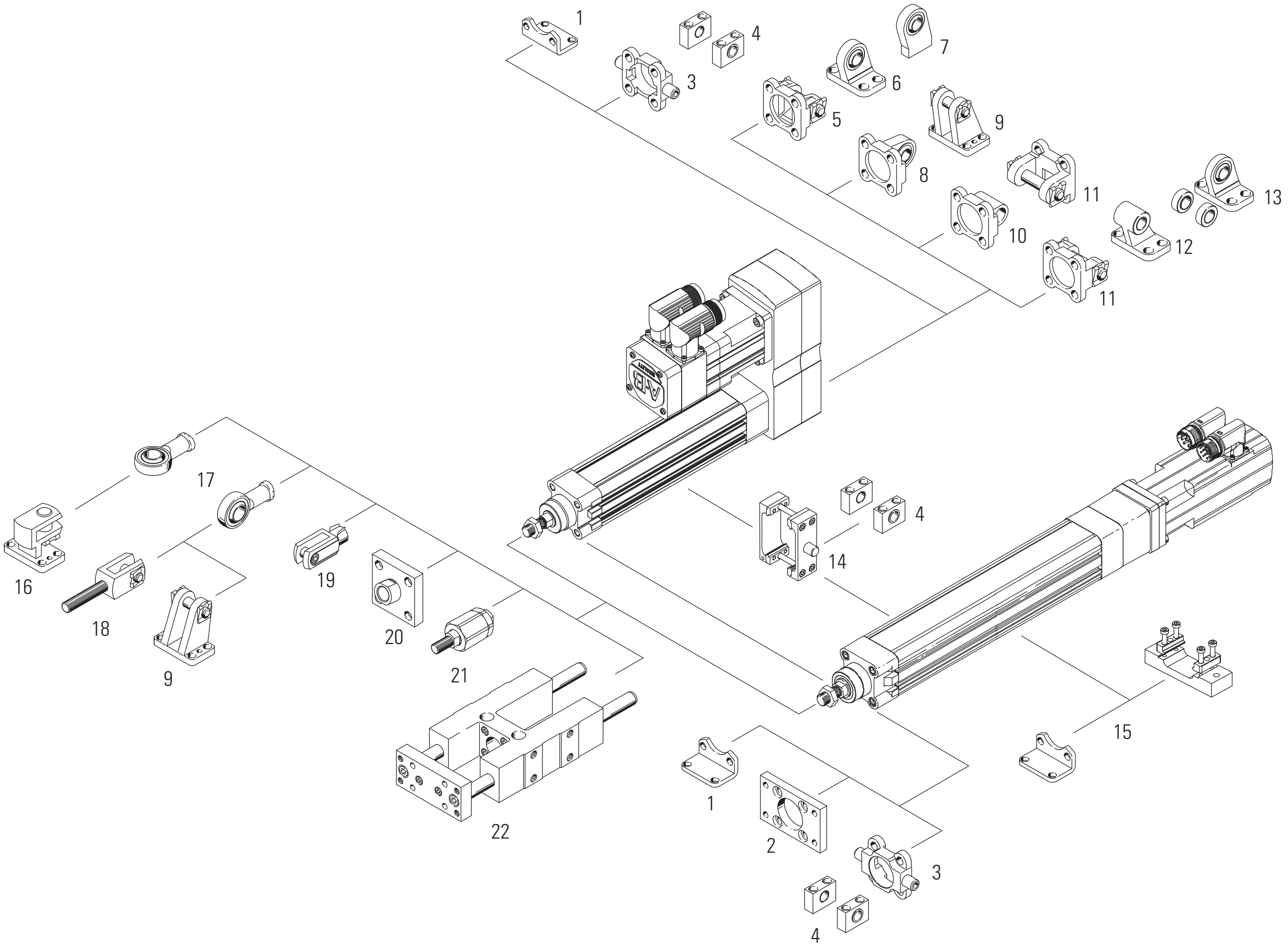


- 2090-CPWM6DF-16AA*xx* (standard, non-flex) power-only cables
- 2090-CPBM6DF-16AA*xx* (standard, non-flex) power-with-brake cables

- 2090-CFBM6DF-CBAA*xx* (standard, non-flex) flying-lead, feedback cables
- 2090-CFBM6DD-CCAA*xx* (standard, non-flex) drive-end connector, feedback cables

MP-Series and TL-Series Electric Cylinders Accessories

These accessory items apply to MP-Series and TL-Series electric cylinders.



Mounting Attachments

Accessory Item		Description	Cat. No. Standard	Cat. No. Corrosion Resistant	Dimensions
1	Foot mount	Parallel applications. Includes two foot mounts and mounting bolts.	MPAR-NP1743*xx*	MPAR-NP1769*xx*	page 162
2	Front flange mounting	In-line or parallel applications. Includes mounting bolts.	MPAR-NA17437*x*	MPAR-NA16184*x*	page 163
3	Trunnion flange		MPAR-NA17441*x*	MPAR-NA16185*x*	page 163
4	Trunnion support	In-line or parallel applications. Use with item 3.	MPAR-NA329*xx*	MPAR-NA16187*x*	page 164
5	Swivel flange (pin, narrow)	Parallel applications. Use with items 6 or 7. Includes swivel flange, pivot pin, mounting screws, and retaining clip.	MPAR-NP17438*x*	N/A	page 165
6	Clevis foot (spherical bearing)	Parallel applications. Use with item 5. Includes 1 flange with spherical bearing.	MPAR-NP3174*x*	N/A	page 166
7	Clevis foot (weld-on)	Parallel applications. Use with item 5. Includes 1 bracket with spherical bearing.	MPAR-NP317*xx*	N/A	page 165
8	Swivel flange (spherical bearing)	Parallel applications. Use with item. Includes 1 flange with spherical bearing and mounting bolts.	MPAR-NP174*xxx*	N/A	page 167

Mounting Attachments (continued)

Accessory Item		Description	Cat. No. Standard	Cat. No. Corrosion Resistant	Dimensions
9	Clevis foot (pin)	In-line or parallel applications. Use with items 8 or 17. Includes mount, pivot pin, and retaining clip.	MPAR-NA3176*x*	N/A	page 167
10	Swivel flange	Parallel applications. Use with item 11. Includes flange and mounting bolts.	MPAR-NP17440*x*	N/A	page 168
11	Swivel flange (pin, wide)	Parallel applications. Use with items 10, 12, or 13. Includes swivel flange, pivot pin, mounting screws, and retaining clips.	MPAR-NP17439*x*	MPAR-NP17694*x*	page 168
12	Clevis foot	Parallel applications. Use with item 11.	MPAR-NP3389*x*	MPAR-NP16184*x*	page 169
13	Clevis foot (spherical bearing)	Parallel applications. Use with item 11. Includes flange with spherical bearing and cylindrical spacers.	MPAR-NP556*x*	N/A	page 170
14	Trunnion mounting kit	In-line or parallel applications. For mounting anywhere along the cylinder barrel profile. Cannot be mounted in the vicinity of the motor with parallel motor attachment. Kit includes trunnion mount and supports.	MPAR-NA16352*x*	N/A	page 171
15	Foot mounting kit	In-line applications. For mounting anywhere along the cylinder barrel profile. Kit includes bracket, body clamps, foot mount attachment, and mounting bolts.	MPAR-NA17499*x*	N/A	page 172
16	Clevis foot (right angle)	In-line or parallel applications. Use with item 17. Includes clevis foot, pivot pin, and retaining clip.	MPAR-NA317*xx*	N/A	page 173

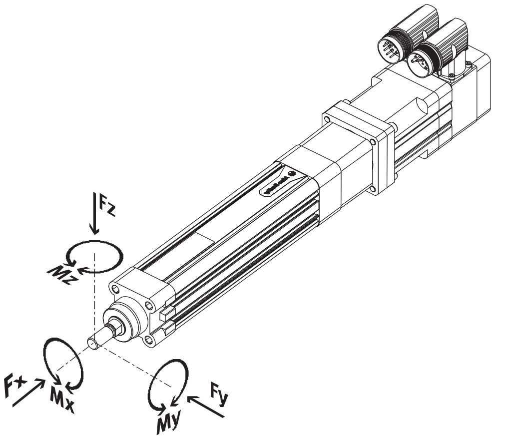
Rod-end Attachments

Accessory Item		Description	Cat. No. Standard	Cat. No. Corrosion Resistant	Dimensions
17	Rod eye	In-line or parallel applications. Use with items 16, 18, or 9. Includes rod eye and hex nut.	MPAR-NE926*x*	MPAR-NE19558*x*	page 174
18	Rod clevis (threaded rod)	In-line or parallel applications. Use with item 17. For spherical connection of cylinders with rod eye. Includes rod clevis, pivot pin, and retaining clip.	MPAR-NE*xxxxx*	N/A	page 175
19	Rod clevis (threaded hole)	In-line or parallel applications. Use with item 17. Permits a swivelling movement of the cylinder in one plane. Includes rod clevis, pivot pin, and retaining clip.	MPAR-NE614*x*	MPAR-NE135*xx*	page 175
20	Coupling piece	In-line or parallel applications. Links the piston rod-end of the cylinder to the machine parts to be moved. Used to compensate for radial misalignment. Includes flange plate and threaded coupling.	MPAR-NE3612*x*	N/A	page 177
21	Rod coupler	In-line or parallel applications. Links the piston rod-end of the cylinder to the machine parts to be moved. Used to compensate for radial and angular misalignment. Includes rod coupler and hex nut.	MPAR-NE614*x*	N/A	page 178
22	Rod guide	In-line or parallel applications. Used to protect piston rod against radial or torsional side loads.	MPAR-NE*xxxxx*	N/A	page 181

MP-Series and TL-Series Electric Cylinders Rod Load Ratings

Electric cylinders must have the weight of the load supported and guided separately so that only axial force (no radial load) is required of the piston rod throughout the complete extend and retract motion. If some residual radial and/or torsional loading remains unavoidable, it may be necessary to add a rod guide. Refer to Motion Analyzer software, version 4.7 or later, for assistance when making these calculations and to determine when a rod guide is recommended.

Maximum Permissible Lateral Forces Fy_{max} and Fz_{max} on the Piston Rod

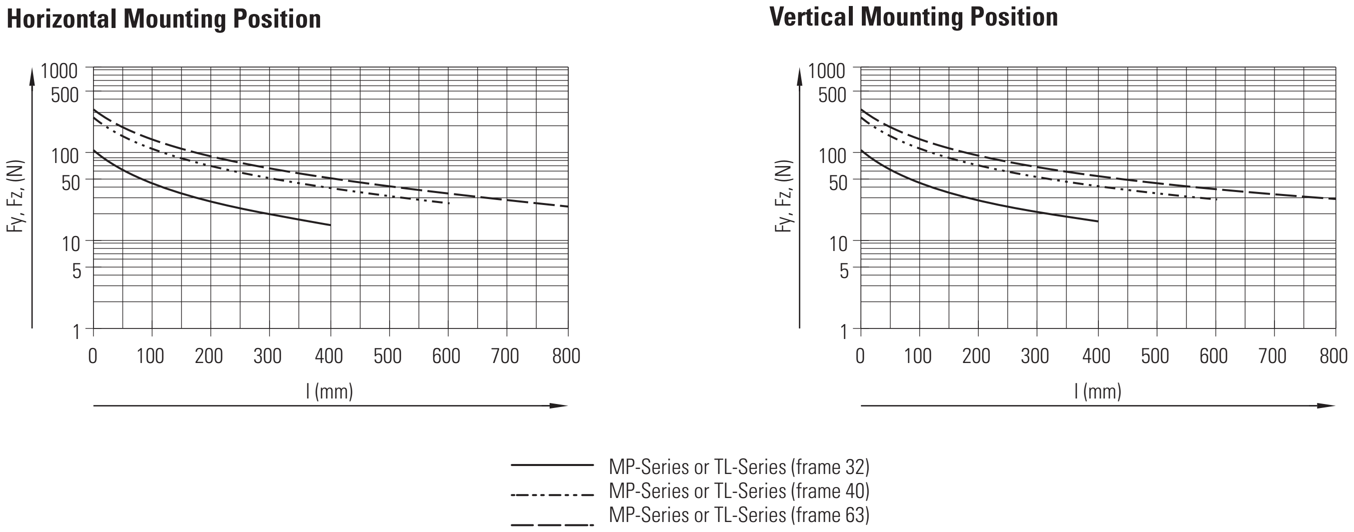


If there are two or more forces and torques acting simultaneously on the piston rod, the following equations must be true:

$$\frac{[Fy]}{Fy\ max} + \frac{[Fz]}{Fz\ max} + \frac{[My]}{My\ max} + \frac{[Mz]}{Mz\ max} \leq 1$$

$$[Fx] \leq Fx\ max$$

$$[Mx] \leq Mx\ max$$



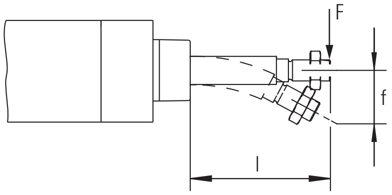
Load Force Ratings

Attribute	MP-Series and TL-Series Electric Cylinders		
	Frame 32	**Frame 40**	**Frame 63**
Fx max (static)	600 N (135 lb)	1400 N (315 lb)	3700 N (832 lb)
Mx max	1 N•m (8.8 lb•in)	1 N•m (8.8 lb•in)	1.5 N•m (13.3 lb•in)
My max, Mz max	8 N•m (70.7 lb•in)	20 N•m (177 lb•in)	27 N•m (239 lb•in)

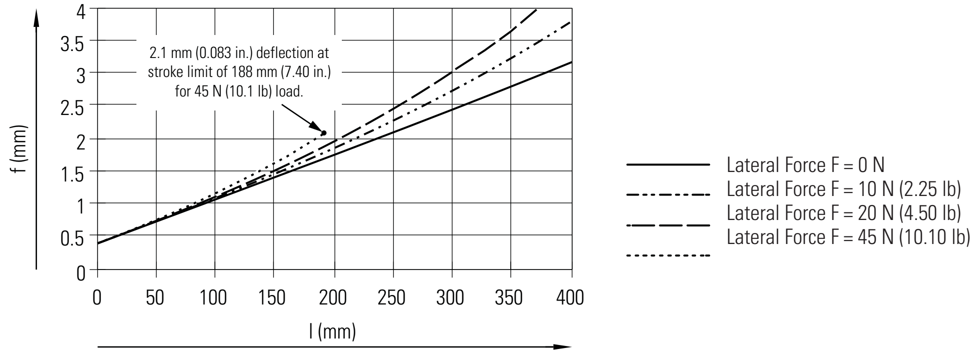
MP-Series and TL-Series Electric Cylinders Piston Rod Deflection Specifications

These specifications are for determining the amount of shaft deflection to expect from a given load.

Piston-rod deflection specifications are a function of stroke length.

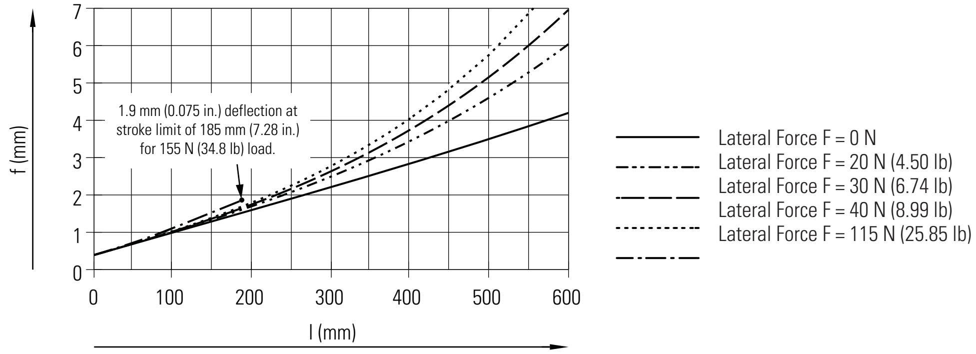


MP-Series and TL-Series Electric Cylinders (frame 32)



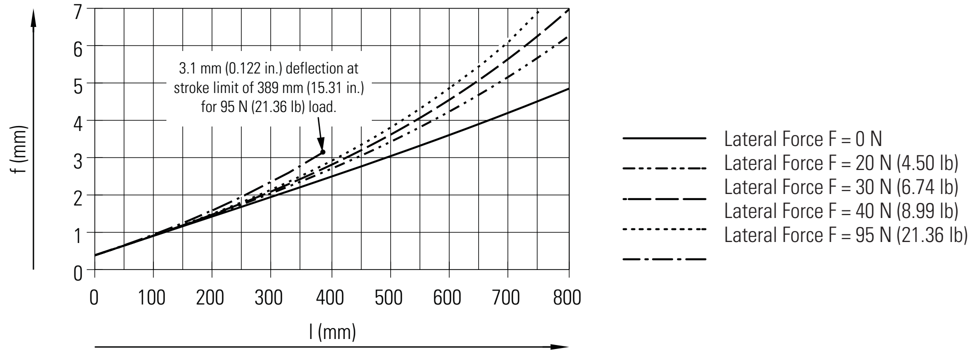
2.1 mm (0.083 in.) deflection at stroke limit of 188 mm (7.40 in.) for 45 N (10.1 lb) load.

MP-Series and TL-Series Electric Cylinders (frame 40)



1.9 mm (0.075 in.) deflection at stroke limit of 185 mm (7.28 in.) for 155 N (34.8 lb) load.

MP-Series and TL-Series Electric Cylinders (frame 63)



3.1 mm (0.122 in.) deflection at stroke limit of 389 mm (15.31 in.) for 95 N (21.36 lb) load.

MP-Series and TL-Series Electric Cylinders Life Specifications

Electric cylinder life specifications, running performance (L), are based on a combination of tested and calculated data. If the parameters of your application are different, your results could be different. The achievable running performance (L) is a function of mean feed force (F), according to DIN 69051-4 as shown below and illustrated on page 141. Refer to Motion Analyzer software, version 4.7 or later, for assistance when making these calculations and determining the running performance for your application.

Calculation of the Feed Force Fx

The peak feed force value must not exceed the maximum feed force within a movement cycle. In the case of vertical operation, the peak value is generally achieved during the acceleration phase of the upwards stroke. If the maximum feed force is exceeded, you can increase wear and shorten the service life of the ballscrew. The maximum speed must likewise not be exceeded.

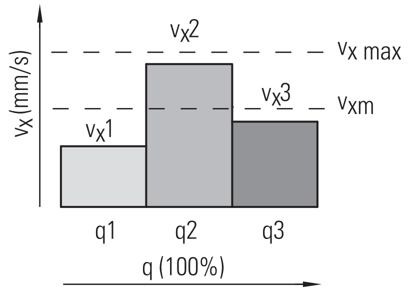
$Fx \leq Fx\ max$
and
$Vx \leq Vx\ max$

During operation, the continuous feed force may be briefly exceeded up to the maximum feed force. The continuous feed force must, however, be adhered to when averaged over a movement cycle.

$Fxm \leq Fx\ cont$

Mean Feed Speed (to DIN 69051-4)

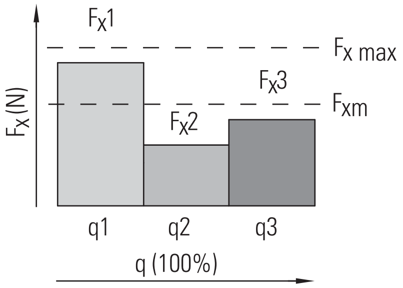
$$v_{xm} = \sum \; |v_x| \times \frac{q}{100} + |v_{x1}| \times \frac{q1}{100} + |v_{x2}| \times \frac{q2}{100} + |v_{x3}| \times \frac{q3}{100} + \ldots$$



Mean Feed Force (to DIN 69051-4)

$$F_{xm} = \sqrt[3]{\sum |F_x|^3 \times \frac{|v_x|}{|v_{xm}|} \times \frac{q}{100}} =$$

$$F_{xm} = \sqrt[3]{|F_{x1}|^3 \times \frac{|v_{x1}|}{|v_{xm}|} \times \frac{q1}{100} + |F_{x2}|^3 \times \frac{|v_{x2}|}{|v_{xm}|} \times \frac{q2}{100} + |F_{x3}|^3 \times \frac{|v_{x3}|}{|v_{xm}|} \times \frac{q3}{100} + \ldots}$$

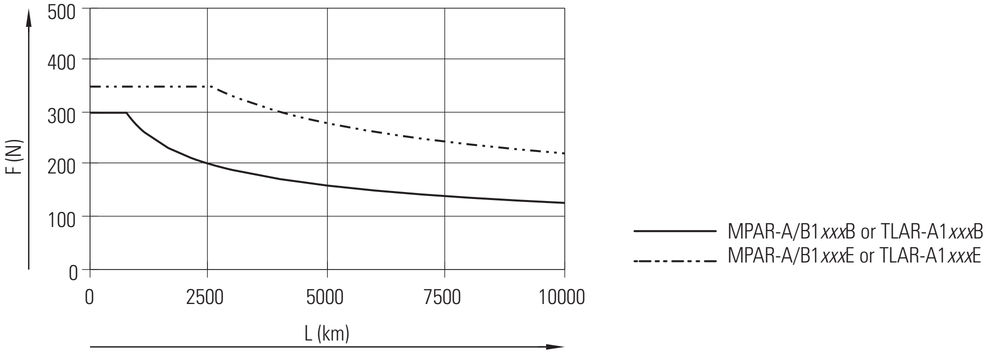


F_x = Feed Force
F_{xm} = Mean Feed Force
F_{xmax} = Maximum Feed Force
F_{xcont} = Continuous Feed Force
q = % of Time

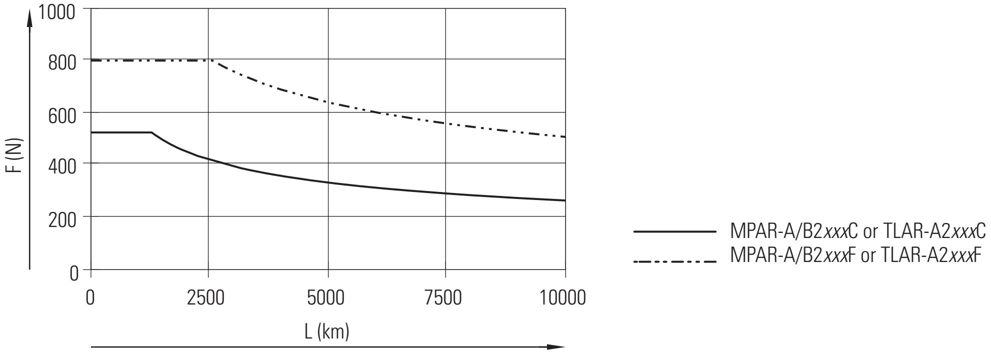
V_x = Feed Speed
V_{xm} = Mean Feed Speed
V_{xmax} = Maximum Feed Speed

The achievable running performance (L) is a function of mean feed force (F), according to DIN 69051-4 as illustrated in the figures below.

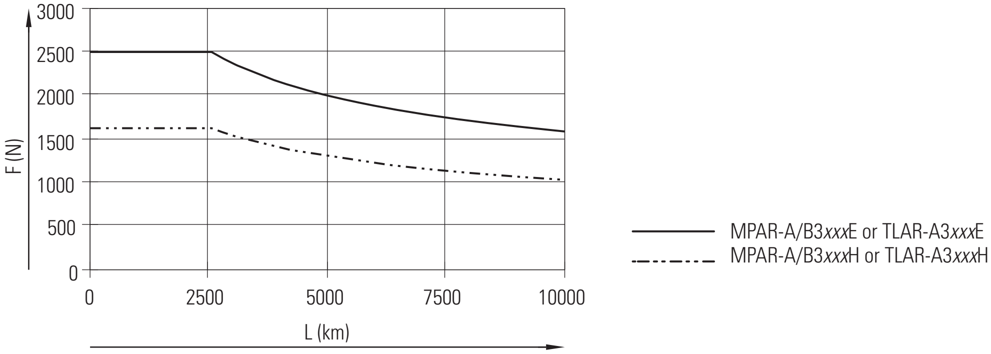
MP-Series and TL-Series Electric Cylinders (frame 32)



MP-Series and TL-Series Electric Cylinders (frame 40)



MP-Series and TL-Series Electric Cylinders (frame 63)



MP-Series and TL-Series Electric Cylinders Specifications

MP-Series and TL-Series Electric Cylinders General Specifications

Attribute	Frame 32	Frame 40	Frame 63
Construction design	Ballscrew servo-driven non-rotating piston rod (1)		
Piston rod thread	M10x1.25	M12x1.25	M16x1.50
Working stroke	100 mm (3.94 in.) 200 mm (7.87 in.) 300 mm (11.81 in.) 400 mm (15.75 in.)	100 mm (3.94 in.) 200 mm (7.87 in.) 300 mm (11.81 in.) 400 mm (15.75 in.) 600 mm (23.62 in.)	100 mm (3.94 in.) 200 mm (7.87 in.) 300 mm (11.81 in.) 400 mm (15.75 in.) 600 mm (23.62 in.) 800 mm (31.50 in.)
Protection against torsion/guide	Plain bearing guide		
Stroke reserve	0 mm		
Angle of rotation at the piston rod, max	±0.30°	±0.25°	±0.20°
Impact energy (E) at the end positions $E=0.5 \times m \times v^2$	0.0001 J	0.0002 J	0.0004 J
Positioning repeatability, max	±0.02 mm (0.0008 in.)		
Reversing backlash, max (2)	0.05 mm (0.002 in.)		
Duty cycle	100%		
Position sensing (feedback)	Multi-turn absolute encoder		
Type of mounting	Via female threads		
	Via accessories		
Mounting position	Any		

(1) The maximum rotational force (Mx) applied in the application is limited as specified in Load Force Ratings on page 138.

(2) In new condition.

MP-Series and TL-Series Electric Cylinders Moving Load Specifications

Attribute	MPAR-*x*1*xxx*B TLAR-A1*xxx*B	MPAR-*x*1*xxx*E TLAR-A1*xxx*E	MPAR-*x*2*xxx*C TLAR-A2*xxx*C	MPAR-*x*2*xxx*F TLAR-A2*xxx*F	MPAR-*x*3*xxx*E TLAR-A3*xxx*E	MPAR-*x*3*xxx*H TLAR-A3*xxx*H
	Frame 32, Weight, approx. g (oz)		**Frame 40, Weight, approx.** g (oz)		**Frame 63, Weight, approx.** g (oz)	
Moving load with 0 mm stroke	170 (6.0)	200 (7.05)	310 (10.93)	380 (13.40)	810 (28.57)	810 (1.79)
Moving load per 10 mm stroke	6.9 (0.24)	6.9 (0.24)	8.9 (0.31)	8.9 (0.31)	12.8 (0.45)	12.8 (0.028)

MP-Series Electric Cylinders Performance Specifications

Electric Cylinder [(1)] **Cat. No.**	**Frame**	**Max Feed Force** N (lb)	**Continuous Feed Force** N (lb)	**Max Speed** m/s (in./s)	**Stroke Lengths** mm (in.)	**Max Acceleration** m/s^2 ($in./s^2$)
MPAR-A1*xxx*B-V*xx* MPAR-B1*xxx*B-V*xx*	32	300 (67)	240 (54)	0.15 (5.9)	100 (3.94) 200 (7.87) 300 (11.81) 400 (15.75)	6.0 (236)
MPAR-A1*xxx*E-V*xx* MPAR-B1*xxx*E-V*xx*		350 (79)	280 (63)	0.50 (19.7)		
MPAR-A2*xxx*C-V*xx* MPAR-B2*xxx*C-V*xx*	40	525 (118)	420 (94)	0.25 (9.8)	100 (3.94) 200 (7.87) 300 (11.81) 400 (15.75) 600 (23.62)	
MPAR-A2*xxx*F-V*xx* MPAR-B2*xxx*F-V*xx*		800 (180)	640 (144)	0.64 (25.2)		
MPAR-A3*xxx*E-M*xx* MPAR-B3*xxx*E-M*xx*	63	2500 (562)	2000 (450)	0.50 (19.7)	100 (3.94) 200 (7.87) 300 (11.81) 400 (15.75) 600 (23.62) 800 (31.50)	
MPAR-A3*xxx*H-M*xx* MPAR-B3*xxx*H-M*xx*		1625 (365)	1300 (292)	1.0 (39.4)		

(1) Stroke length replaces *xxx* in each catalog number.

TL-Series Electric Cylinders Performance Specifications

Electric Cylinder [(1)] **Cat. No.**	**Frame**	**Max Feed Force** N (lb)	**Continuous Feed Force** N (lb)	**Max Speed** m/s (in./s)	**Stroke Lengths** mm (in.)	**Max Acceleration** m/s^2 ($in./s^2$)
TLAR-A1*xxx*B-B*x*A	32	300 (67)	240 (54)	0.15 (5.9)	100 (3.94) 200 (7.87) 300 (11.81) 400 (15.75)	6.0 (236)
TLAR-A1*xxx*E-B*x*A		350 (79)	280 (63)	0.50 (19.7)		
TLAR-A2*xxx*C-B*x*A	40	525 (118)	420 (94)	0.25 (9.8)	100 (3.94) 200 (7.87) 300 (11.81) 400 (15.75) 600 (23.62)	
TLAR-A2*xxx*F-B*x*A		800 (180)	520 (117)	0.64 (25.2)		
TLAR-A3*xxx*E-B*x*A	63	2500 (562)	1750 (393)	0.50 (19.7)	100 (3.94) 200 (7.87) 300 (11.81) 400 (15.75) 600 (23.62) 800 (31.50)	
TLAR-A3*xxx*H-B*x*A		1625 (365)	975 (219)	1.0 (39.4)		

(1) Stroke length replaces *xxx* in each catalog number.

MP-Series and TL-Series Electric Cylinders System Combinations

For Bulletin MPAR electric cylinders and	**Refer to**
Kinetix 6000 (230V and 460V) drives and Kinetix 6200/Kinetix 6500 (460V drives)	page 674
Kinetix 2000 (230V) drives	page 700
Kinetix 300 (240V and 480V) drives	page 720
Ultra3000 (230V and 460V) drives	page 749

For Bulletin TLAR electric cylinders and	**Refer to**
Kinetix 2000 (230V) drives	page 705
Kinetix 300 (240V) drives	page 726
Kinetix 3 (240V) drives	page 731

MP-Series and TL-Series Electric Cylinders Weight Specifications

MP-Series Electric Cylinders (weight of cylinder with non-brake motor)

Electric Cylinder Cat. No.	**Weight, (1) approx.** kg (lb)
MPAR-*x*1100B-V2A	2.6 (5.73)
MPAR-*x*1100B-V2B/D/E	3.5 (7.72)
MPAR-*x*1200B-V2A	2.9 (6.39)
MPAR-*x*1200B-V2B/D/E	3.8 (8.377)
MPAR-*x*1300B-V2A	3.2 (7.05)
MPAR-*x*1300B-V2B/D/E	4.1 (9.04)
MPAR-*x*1400B-V2A	3.5 (7.72)
MPAR-*x*1400B-V2B/D/E	4.5 (9.92)
MPAR-*x*1100E-V2A	3.0 (6.61)
MPAR-*x*1100E-V2B/D/E	3.8 (8.377)
MPAR-*x*1200E-V2A	3.3 (7.27)
MPAR-*x*1200E-V2B/D/E	4.1 (9.04)
MPAR-*x*1300E-V2A	3.6 (7.94)
MPAR-*x*1300E-V2B/D/E	4.5 (9.92)
MPAR-*x*1400E-V2A	4.0 (8.82)
MPAR-*x*1400E-V2B/D/E	4.8 (10.58)

Electric Cylinder Cat. No.	**Weight, approx.** kg (lb)
MPAR-*x*2100C-V2A	3.7 (8.16) (1)
MPAR-*x*2100C-V2B/D/E	4.4 (9.70) (1)
MPAR-*x*2200C-V2A	4.1 (9.04) (1)
MPAR-*x*2200C-V2B/D/E	4.9 (10.80) (1)
MPAR-*x*2300C-V2A	4.6 (10.14) (1)
MPAR-*x*2300C-V2B/D/E	5.3 (11.68) (1)
MPAR-*x*2400C-V2A	5.0 (11.02) (1)
MPAR-*x*2400C-V2B/D/E	5.8 (12.79) (1)
MPAR-*x*2600C-V2A	6.0 (11.02) (1)
MPAR-*x*2600C-V2B/D/E	6.7 (14.77) (1)
MPAR-*x*2100F-V2A	4.2 (9.26) (3)
MPAR-*x*2100F-V2B/D/E	6.5 (14.33) (3)
MPAR-*x*2200F-V2A	4.7 (10.36) (3)
MPAR-*x*2200F-V2B/D/E	7.0 (15.43) (3)
MPAR-*x*2300F-V2A	5.2 (11.46) (3)
MPAR-*x*2300F-V2B/D/E	7.5 (16.53) (3)
MPAR-*x*2400F-V2A	5.6 (12.34) (3)
MPAR-*x*2400F-V2B/D/E	7.9 (17.42) (3)
MPAR-*x*2600F-V2A	6.6 (14.55) (3)
MPAR-*x*2600F-V2B/D/E	8.9 (19.62) (3)

Electric Cylinder Cat. No.	**Weight, approx.** kg (lb)
MPAR-*x*3100E-M2A	9.5 (20.94) (2)
MPAR-*x*3100E-M2B/D/E	13.6 (29.98) (2)
MPAR-*x*3200E-M2A	10.3 (22.71) (2)
MPAR-*x*3200E-M2B/D/E	14.4 (31.75) (2)
MPAR-*x*3300E-M2A	11.1 (24.47) (2)
MPAR-*x*3300E-M2B/D/E	15.2 (33.51) (2)
MPAR-*x*3400E-M2A	11.9 (26.23) (2)
MPAR-*x*3400E-M2B/D/E	16.1 (35.49) (2)
MPAR-*x*3600E-M2A	13.5 (29.76) (2)
MPAR-*x*3600E-M2B/D/E	17.7 (39.02) (2)
MPAR-*x*3800E-M2A	15.2 (33.51) (2)
MPAR-*x*3800E-M2B/D/E	19.3 (42.55) (2)
MPAR-*x*3100H-M2A	9.3 (20.50) (4)
MPAR-*x*3100H-M2B/D/E	13.2 (29.10) (4)
MPAR-*x*3200H-M2A	10.1 (22.27) (4)
MPAR-*x*3200H-M2B/D/E	14.0 (30.86) (4)
MPAR-*x*3300H-M2A	10.9 (24.03) (4)
MPAR-*x*3300H-M2B/D/E	14.8 (32.63) (4)
MPAR-*x*3400H-M2A	11.7 (25.79) (4)
MPAR-*x*3400H-M2B/D/E	15.7 (34.61) (4)
MPAR-*x*3600H-M2A	13.4 (29.54) (4)
MPAR-*x*3600H-M2B/D/E	17.3 (38.14) (4)
MPAR-*x*3800H-M2A	15.0 (33.07) (4)
MPAR-*x*3800H-M2B/D/E	18.9 (41.67) (4)

(1) If ordering MPAR-*x*1*xxxx*-V4*x* or MPAR-*x*2*xxx*C-V4*x* electric cylinder with brake, add 0.2 kg (0.4 lb).

(2) If ordering MPAR-*x*3*xxx*E-M4*x* electric cylinder with brake, add 1.0 kg (2.2 lb).

(3) If ordering MPAR-*x*2*xxx*F-V4*x* electric cylinder with brake, add 0.4 kg (0.9 lb).

(4) If ordering MPAR-*x*3*xxx*H-M4*x* electric cylinder with brake, add 1.7 kg (3.7 lb).

TL-Series Electric Cylinders (weight of cylinder with non-brake motor)

Electric Cylinder Cat. No.	**Weight, approx.** kg (lb)
TLAR-A1100B-B2A	1.7 (3.75) (3)
TLAR-A1200B-B2A	2.0 (4.41) (3)
TLAR-A1300B-B2A	2.4 (5.29) (3)
TLAR-A1400B-B2A	2.7 (5.95) (3)
TLAR-A1100E-B2A	2.4 (5.29) (4)
TLAR-A1200E-B2A	2.8 (6.17) (4)
TLAR-A1300E-B2A	3.1 (6.83) (4)
TLAR-A1400E-B2A	3.4 (7.49) (4)

Electric Cylinder Cat. No.	**Weight, approx.** (1) kg (lb)
TLAR-A2100C-B2A	3.1 (6.83)
TLAR-A2200C-B2A	3.6 (7.94)
TLAR-A2300C-B2A	4.0 (8.82)
TLAR-A2400C-B2A	4.5 (9.92)
TLAR-A2600C-B2A	5.4 (11.90)
TLAR-A2100F-B2A	3.7 (8.16)
TLAR-A2200F-B2A	4.1 (9.04)
TLAR-A2300F-B2A	4.6 (10.14)
TLAR-A2400F-B2A	5.1 (11.24)
TLAR-A2600F-B2A	6.0 (13.23)

Electric Cylinder Cat. No.	**Weight, approx.** (2) kg (lb)
TLAR-A3100E-B2A	8.7 (19.18)
TLAR-A3200E-B2A	9.5 (20.94)
TLAR-A3300E-B2A	10.3 (22.71)
TLAR-A3400E-B2A	11.1 (24.47)
TLAR-A3600E-B2A	12.7 (28.0)
TLAR-A3800E-B2A	14.3 (31.52)
TLAR-A3100H-B2A	8.7 (19.18)
TLAR-A3200H-B2A	9.5 (20.94)
TLAR-A3300H-B2A	10.3 (22.71)
TLAR-A3400H-B2A	11.1 (24.47)
TLAR-A3600H-B2A	12.7 (28.0)
TLAR-A3800H-B2A	14.3 (31.52)

(1) If ordering TLAR-A2*xxxx*-B4A electric cylinder with brake, add 0.4 kg (0.9 lb).
(2) If ordering TLAR-A3*xxxx*-B4A electric cylinder with brake, add 0.6 kg (1.3 lb).
(3) If ordering TLAR-A1*xxx*B-B4A electric cylinder with brake, add 0.2 kg (0.4 lb).
(4) If ordering TLAR-A1*xxx*E-B4A electric cylinder with brake, add 0.5 kg (1.1 lb).

Actuator Cylinders (weight of replacement cylinder)

Actuator Cylinder (1) **Cat. No.**	**Weight, approx.** kg (lb)
MPAR-X1100B	1.1 (2.43)
MPAR-X1200B	1.4 (3.09)
MPAR-X1300B	1.7 (3.75)
MPAR-X1400B	2.1 (4.63)
MPAR-X1100E	1.1 (2.43)
MPAR-X1200E	1.4 (3.09)
MPAR-X1300E	1.8 (3.97)
MPAR-X1400E	2.1 (4.63)

Actuator Cylinder (1) **Cat. No.**	**Weight, approx.** kg (lb)
MPAR-X2100C	1.7 (3.75)
MPAR-X2200C	2.2 (4.85)
MPAR-X2300C	2.6 (5.73)
MPAR-X2400C	3.1 (6.83)
MPAR-X2600C	4.0 (8.82)
MPAR-X2100F	1.8 (3.97)
MPAR-X2200F	2.3 (5.07)
MPAR-X2300F	2.8 (6.17)
MPAR-X2400F	3.2 (7.05)
MPAR-X2600F	4.2 (9.26)

Actuator Cylinder (1) **Cat. No.**	**Weight, approx.** kg (lb)
MPAR-X3100E	3.8 (8.38)
MPAR-X3200E	4.6 (10.14)
MPAR-X3300E	5.4 (11.90)
MPAR-X3400E	6.3 (13.89)
MPAR-X3600E	7.9 (17.42)
MPAR-X3800E	9.5 (20.94)
MPAR-X3100H	3.8 (8.38)
MPAR-X3200H	4.6 (10.14)
MPAR-X3300H	5.4 (11.90)
MPAR-X3400H	6.3 (13.89)
MPAR-X3600H	7.9 (17.42)
MPAR-X3800H	9.5 (20.94)

(1) Replacement actuator cylinders apply to MP-Series and TL-Series electric cylinders. For example, if ordering a replacement cylinder for electric cylinder catalog numbers MPAR-A2100C-V2A or TLAR-A2100C-B2A, the replacement actuator cylinder catalog number is MPAR-X2100C.

MP-Series and TL-Series Electric Cylinders Mounting Accessories

Accessory Item		Frame	Cat. No.	Weight, approx. g (oz)
1	Foot mount attachment	32	MPAR-NP174369	140 (4.94)
		40	MPAR-NP174370	280 (9.87)
		63	MPAR-NP174372	550 (19.40)
2	Flange mounting	32	MPAR-NA174376	240 (8.46)
		40	MPAR-NA174377	280 (9.88)
		63	MPAR-NA174379	690 (24.34)
3	Trunnion flange	32	MPAR-NA174411	130 (4.58)
		40	MPAR-NA174412	240 (8.46)
		63	MPAR-NA174414	600 (21.16)
4	Trunnion support	32	MPAR-NA32959	130 (4.58)
		40	MPAR-NA32960	400 (14.11)
		63	MPAR-NA32961	480 (16.93)
5	Swivel flange (pin, narrow)	32	MPAR-NP174383	90 (3.17)
		40	MPAR-NP174384	120 (4.23)
		63	MPAR-NP174386	320 (11.29)
7	Clevis foot (weld-on)	32	MPAR-NP31747	105 (3.70)
		40	MPAR-NP31748	160 (5.64)
		63	MPAR-NP31747	365 (12.87)
9	Clevis foot (pin)	32	MPAR-NA31761	220 (7.76)
		40	MPAR-NA31762	300 (10.58)
		63	MPAR-NA31764	580 (20.46)
11	Swivel flange (pin, wide)	32	MPAR-NP174390	100 (3.53)
		40	MPAR-NP174391	150 (5.29)
		63	MPAR-NP174393	370 (13.05)
12	Clevis foot	32	MPAR-NP33890	170 (6.00)
		40	MPAR-NP33891	240 (8.46)
		63	MPAR-NP33893	520 (18.34)
13	Clevis foot (spherical bearing)	32	MPAR-NP5561	160 (5.64)
		40	MPAR-NP5562	270 (9.52)
		63	MPAR-NP5564	605 (21.34)
15	Foot mounting kit	32	MPAR-NA174991	240 (8.46)
		40	MPAR-NA174992	310 (10.93)
		63	MPAR-NA174993	510 (17.99)

Accessory Item		Frame	Cat. No.	Weight, approx. g (oz)
1	Foot mount attachment (corrosion resistant)	32	MPAR-NP176937	140 (4.94)
		40	MPAR-NP176938	280 (9.87)
		63	MPAR-NP176940	550 (19.40)
2	Flange mounting (corrosion resistant)	32	MPAR-NA161846	240 (8.46)
		40	MPAR-NA161847	300 (10.58)
		63	MPAR-NA161849	710 (25.04)
3	Trunnion flange (corrosion resistant)	32	MPAR-NA161852	150 (5.29)
		40	MPAR-NA161853	260 (9.17)
		63	MPAR-NA161855	640 (22.57)
4	Trunnion support (corrosion resistant)	32	MPAR-NA161874	200 (7.05)
		40	MPAR-NA161875	330 (11.64)
		63	MPAR-NA161876	440 (11.64)
6	Clevis foot (spherical bearing)	32	MPAR-NP31740	185 (6.53)
		40	MPAR-NP31741	295 (10.40)
		63	MPAR-NP31743	655 (23.10)
8	Swivel flange (spherical bearing)	32	MPAR-NP174397	85 (3.00)
		40	MPAR-NP174398	125 (4.41)
		63	MPAR-NP174400	280 (9.88)
10	Swivel flange	32	MPAR-NP174404	75 (2.64)
		40	MPAR-NP174405	100 (3.53)
		63	MPAR-NP174407	250 (8.82)
11	Swivel flange (corrosion resistant)	32	MPAR-NP176944	100 (3.53)
		40	MPAR-NP176945	150 (5.29)
		63	MPAR-NP176947	370 (13.05)
12	Clevis foot (corrosion resistant)	32	MPAR-NP161840	120 (4.23)
		40	MPAR-NP161841	210 (7.41)
		63	MPAR-NP161843	450 (15.87)
14	Trunnion mounting kit	32	MPAR-NA163525	210 (7.41)
		40	MPAR-NA163526	390 (13.76)
		63	MPAR-NA163528	890 (31.39)
16	Clevis foot (right angle)	32	MPAR-NA31768	290 (10.23)
		40	MPAR-NA31769	360 (12.70)
		63	MPAR-NA31771	880 (31.0)

MP-Series and TL-Series Electric Cylinders Rod-end Accessories

Accessory Item		Frame	Cat. No.	Weight, approx. g (oz)
17	Rod eye	32	MPAR-NE9261	70 (2.47)
		40	MPAR-NE9262	110 (3.53)
		63	MPAR-NE9263	210 (7.41)
18	Rod clevis (threaded rod)	32	MPAR-NE32954	140 (4.94)
		40	MPAR-NE10767	210 (7.41)
		63	MPAR-NE10768	500 (17.64)
19	Rod clevis (corrosion resistant)	32	MPAR-NE13569	110 (3.88)
		40	MPAR-NE13570	180 (6.35)
		63	MPAR-NE13571	400 (14.11)
21	Self-aligning rod coupler	32	MPAR-NE6140	210 (7.41)
		40	MPAR-NE6141	220 (7.76)
		63	MPAR-NE6142	650 (22.93)

Accessory Item		Frame	Cat. No.	Weight, approx. g (oz)
17	Rod eye (corrosion resistant)	32	MPAR-NE195582	70 (2.47)
		40	MPAR-NE195583	110 (3.53)
		63	MPAR-NE195584	210 (7.41)
19	Rod clevis	32	MPAR-NE6144	110 (3.88)
		40	MPAR-NE6145	170 (6.00)
		63	MPAR-NE6146	390 (13.76)
20	Coupling piece	32	MPAR-NE36125	110 (3.88)
		40	MPAR-NE36126	180 (6.35)
		63	MPAR-NE36127	250 (8.82)

MP-Series and TL-Series Electric Cylinders Rod Guide (item 22) Accessories

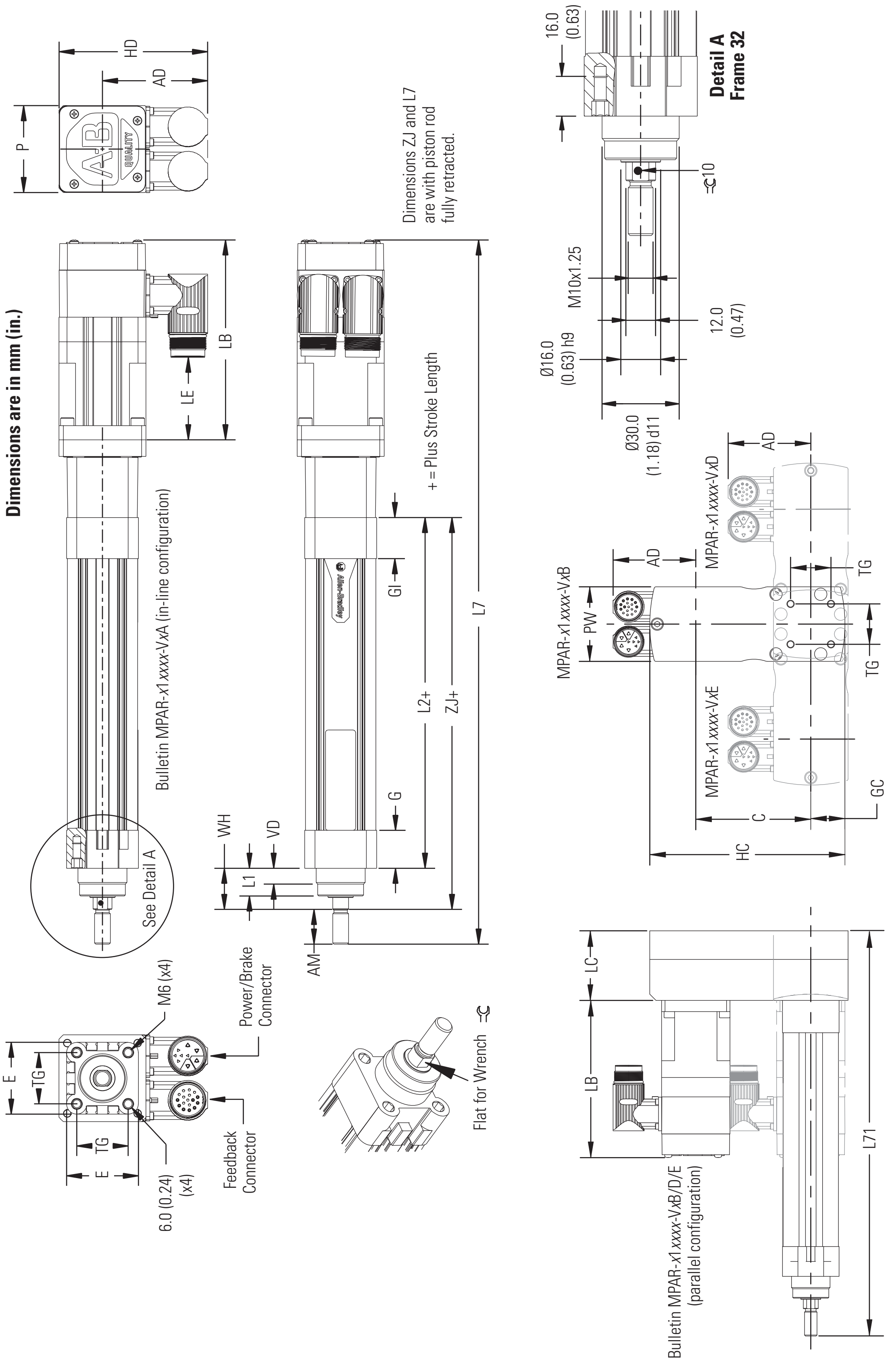
Rod Guide Cat. No.	Frame	Stroke Length mm (in.)	Weight, approx. kg (lb)
MPAR-NE34494	32	100 (3.9)	1.7 (3.75)
MPAR-NE34496		200 (7.9)	1.9 (4.19)
MPAR-NE34497		320 (12.6)	2.1 (4.63)
MPAR-NE150290		400 (15.7)	2.3 (5.07)
MPAR-NE34500	40	100 (3.9)	2.7 (5.95)
MPAR-NE34502		200 (7.9)	3.0 (6.61)
MPAR-NE34504		320 (12.6)	3.4 (7.50)
MPAR-NE150291		400 (15.7)	3.7 (8.16)
MPAR-NE34505		500 (19.7)	4.0 (8.82)
MPAR-NE34514	63	100 (3.9)	5.9 (13.01)
MPAR-NE34516		200 (7.9)	6.4 (14.11)
MPAR-NE34518		320 (12.6)	7.0 (15.43)
MPAR-NE34519		400 (15.7)	7.4 (16.31)
MPAR-NE34520		500 (19.7)	7.9 (17.42)

MP-Series and TL-Series Electric Cylinders Motor Brake Specifications

Electric Cylinder Cat. No.	**Holding Force** N (lb)	**Coil Current at 24V DC** A	**Brake Response Time**		
			Release ms	**Engage (using external arc suppression device)**	
				MOV ms	**Diode** ms
TLAR-A1*xxx*B	300 (67)	0.18…0.22	21	7	40
TLAR-A1*xxx*E	350 (79)	0.333…0.407	22	13	73
TLAR-A2*xxx*C	525 (118)				
TLAR-A2*xxx*F	552 (124)				
TLAR-A3*xxx*E	1414 (318)	0.351…0.429	42	14	86
TLAR-A3*xxx*H	707 (159)				
MPAR-A/B1*xxx*B	300 (67)	0.43…0.53	23	9	18
MPAR-A/B1*xxx*E	350 (79)				
MPAR-A/B2*xxx*C	525 (118)				
MPAR-A/B2*xxx*F	800 (180)	0.46…0.56	58	20	42
MPAR-A/B3*xxx*E	2364 (531)	0.45…0.55	50	20	110
MPAR-A/B3*xxx*H	1625 (365)	0.576…0.704	110	25	160

Notes:

MP-Series Electric Cylinder Dimensions



MP-Series Electric Cylinders Dimensions (frame 32)

MP-Series Electric Cylinder Dimensions (in-line configuration, frame 32)

Electric Cylinder Cat. No.	L7 (1) mm (in.)	LB (1) mm (in.)	LE (2) mm (in.)	P mm (in.)	AD mm (in.)	HD mm (in.)	AM mm (in.)	G mm (in.)	G1 mm (in.)	L1 mm (in.)	L2 mm (in.)	ZJ (3) mm (in.)	VD mm (in.)	WH mm (in.)	E mm (in.)	TG mm (in.)
MPAR-*x*1100B-V2A	445.7 (17.55)	126.5 (4.98)	52.4 (2.06)	55.0 (2.17)	66.5 (2.62)	94.0 (3.70)	22.0 (0.87)	24.0 (0.94)	26.0 (1.02)	18.0 (0.71)	122.0 (4.80)	148.0 (5.83)	10.0 (0.39)	26.0 (1.02)	45.5 (1.79)	32.5 (1.28)
MPAR-*x*1200B-V2A	545.7 (21.48)															
MPAR-*x*1300B-V2A	645.7 (25.42)															
MPAR-*x*1400B-V2A	745.7 (29.36)															
MPAR-*x*1100E-V2A	470.7 (18.53)	151.5 (5.96)	77.2 (3.04)													
MPAR-*x*1200E-V2A	570.7 (22.47)															
MPAR-*x*1300E-V2A	670.7 (26.41)															
MPAR-*x*1400E-V2A	770.7 (30.34)															

(1) If ordering MPAR-A/B1*xxxx*-V4A actuator with brake, add 36.1 mm (1.42 in.) to dimensions L7 and LB.

(2) If ordering MPAR-A/B1*xxxx*-V4A actuator with brake, add 33.4 mm (1.31 in.) to dimension LE.

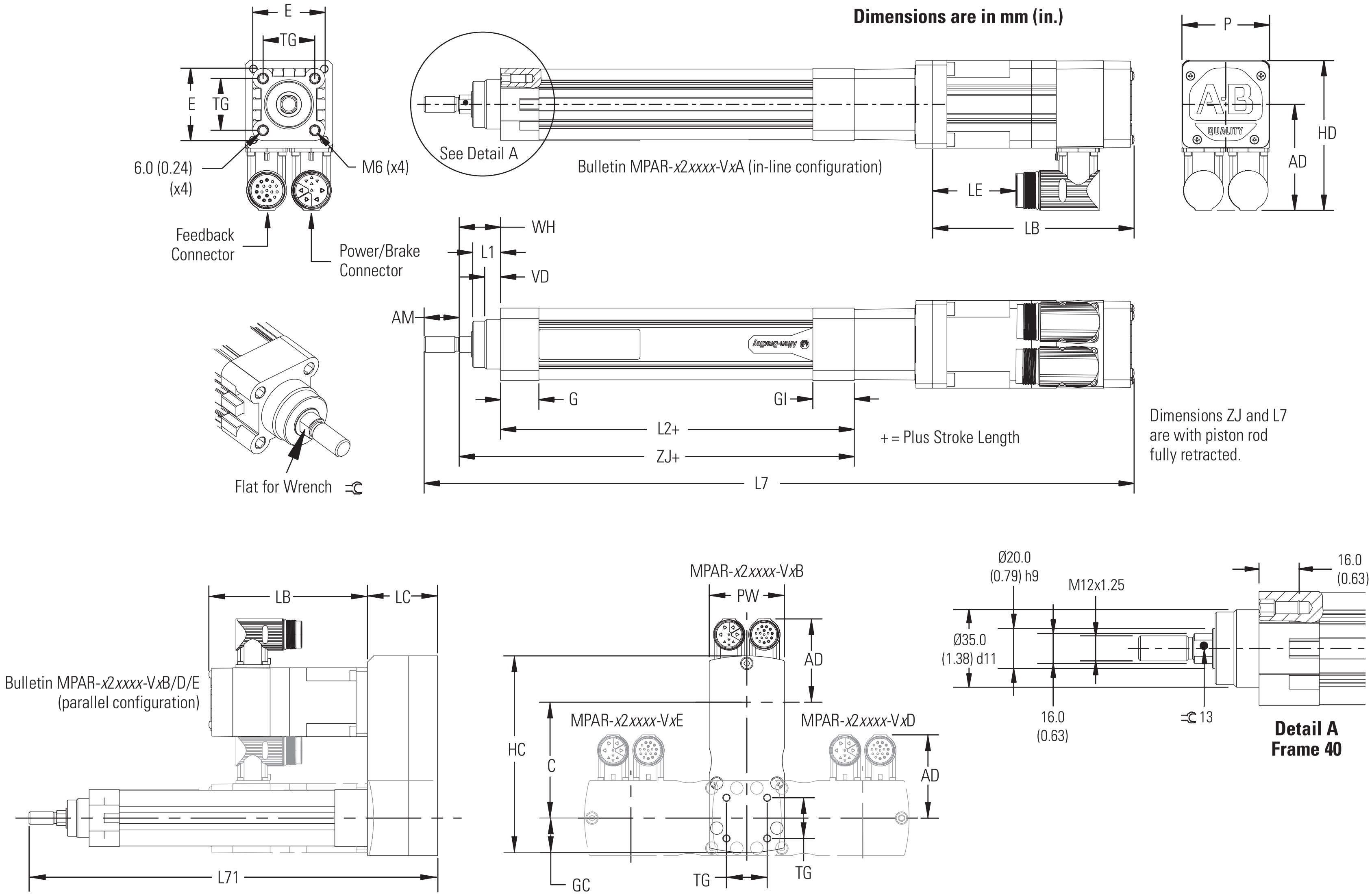
(3) The tolerance for this dimension is ±1.0 mm (0.039 in.).

MP-Series Electric Cylinder Dimensions (parallel configuration, frame 32)

Electric Cylinder Cat. No.	L71 mm (in.)	LC mm (in.)	HC mm (in.)	C (1) mm (in.)	GC mm (in.)	PW mm (in.)
MPAR-*x*1100B-V2B/D/E	326.0 (12.8)	56.0 (2.20)	157 (6.18)	92.5 (3.64)	27.5 (1.08)	60.0 (2.36)
MPAR-*x*1200B-V2B/D/E	426.0 (16.8)					
MPAR-*x*1300B-V2B/D/E	526.0 (20.7)					
MPAR-*x*1400B-V2B/D/E	626.0 (24.6)					
MPAR-*x*1100E-V2B/D/E	326.0 (12.8)					
MPAR-*x*1200E-V2B/D/E	426.0 (16.8)					
MPAR-*x*1300E-V2B/D/E	526.0 (20.7)					
MPAR-*x*1400E-V2B/D/E	626.0 (24.6)					

(1) The tolerance for this dimension is ±1.0 mm (0.039 in.)

MP-Series Electric Cylinders Dimensions (frame 40)



MP-Series Electric Cylinder Dimensions (in-line configuration, frame 40)

Electric Cylinder Cat. No.	L7 (1) mm (in.)	LB (1) mm (in.)	LE (2) mm (in.)	P mm (in.)	AD mm (in.)	HD mm (in.)	AM mm (in.)	G mm (in.)	G1 mm (in.)	L1 mm (in.)	L2 mm (in.)	ZJ (3) mm (in.)	VD mm (in.)	WH mm (in.)	E mm (in.)	TG mm (in.)
MPAR-*x*2100C-V2A	501.2 (19.73)	151.5 (5.96)	77.2 (3.04)	55.0 (2.17)	66.5 (2.62)	94.0 (3.70)	24.0 (0.94)	28.5 (1.12)	30.0 (1.18)	21.5 (0.85)	146.5 (5.77)	176.5 (6.95)	10.5 (0.41)	30.0 (1.18)	54.0 (2.13)	38.0 (1.50)
MPAR-*x*2200C-V2A	601.2 (23.67)															
MPAR-*x*2300C-V2A	701.2 (27.61)															
MPAR-*x*2400C-V2A	801.2 (31.54)															
MPAR-*x*2600C-V2A	1001.2 (39.42)															
MPAR-*x*2100F-V2A	492.1 (19.37)	140.1 (5.52)	65.1 (2.56)	70.0 (2.76)	74.0 (2.91)	109.0 (4.29)										
MPAR-*x*2200F-V2A	592.1 (23.31)															
MPAR-*x*2300F-V2A	692.1 (27.25)															
MPAR-*x*2400F-V2A	792.1 (31.19)															
MPAR-*x*2600F-V2A	992.1 (39.06)															

(1) If ordering MPAR-A/B2*xxx*C-V4A actuator with brake, add 36.1 mm (1.42 in.) to dimensions L7 and LB.
If ordering MPAR-A/B2*xxx*F-V4A actuator with brake, add 39.0 mm (1.54 in.) to dimensions L7 and LB.

(2) If ordering MPAR-A/B2*xxx*C-V4A actuator with brake, add 33.4 mm (1.31 in.) to dimension LE.
If ordering MPAR-A/B2*xxx*F-V4A actuator with brake, add 24.7 mm (0.97 in.) to dimension LE.

(3) The tolerance for this dimension is ±1.0 mm (0.039 in.).

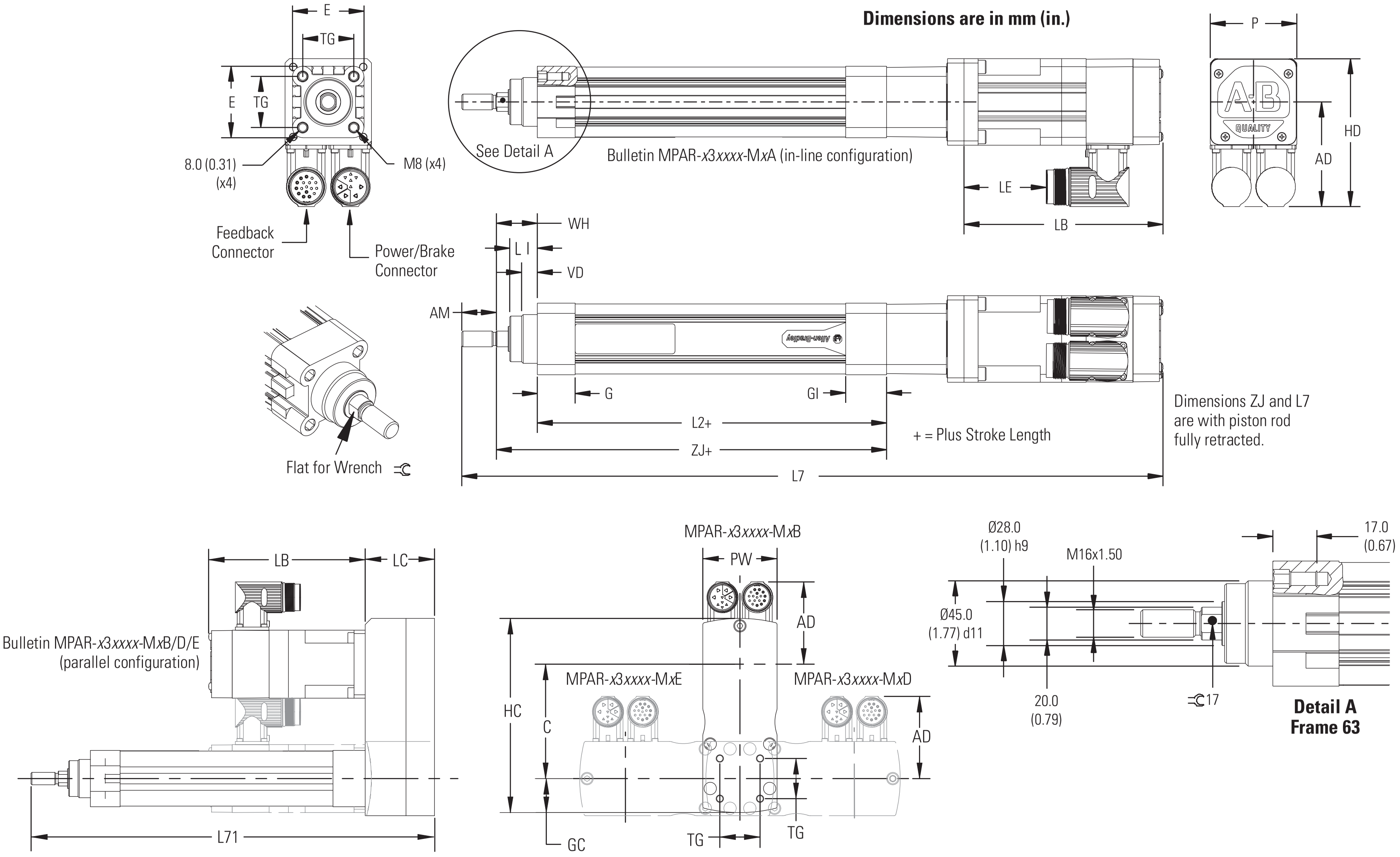
MP-Series Electric Cylinder Dimensions (parallel configuration, frame 40)

Electric Cylinder Cat. No.	L71 mm (in.)	LC mm (in.)	HC mm (in.)	C (1) mm (in.)	GC mm (in.)	PW mm (in.)
MPAR-*x*2100C-V2B/D/E	356.5 (14.03)	56.0 (2.20)	157.0 (6.18)	91.5 (3.60)	27.0 (1.06)	60.0 (2.36)
MPAR-*x*2200C-V2B/D/E	456.5 (17.97)					
MPAR-*x*2300C-V2B/D/E	556.5 (21.91)					
MPAR-*x*2400C-V2B/D/E	656.5 (25.84)					
MPAR-*x*2600C-V2B/D/E	856.5 (33.72)					
MPAR-*x*2100F-V2B/D/E	369.5 (14.55)	69.0 (2.72	188.5 (7.42)	102.5 (4.03)	38.0 (1.50)	86.0 (3.38)
MPAR-*x*2200F-V2B/D/E	469.5 (18.48)					
MPAR-*x*2300F-V2B/D/E	569.5 (22.42)					
MPAR-*x*2400F-V2B/D/E	669.5 (26.36)					
MPAR-*x*2600F-V2B/D/E	869.5 (34.23)					

(1) The tolerance for this dimension is ±1.0 mm (0.039 in.)

Actuators are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

MP-Series Electric Cylinders Dimensions (frame 63)



Dimensions are in mm (in.)

+ = Plus Stroke Length

Dimensions ZJ and L7 are with piston rod fully retracted.

MP-Series Electric Cylinder Dimensions (in-line configuration, frame 63)

Electric Cylinder Cat. No.	**L7** [1] mm (in.)	**LB** [1] mm (in.)	**LE** [1] mm (in.)	**P** mm (in.)	**AD** mm (in.)	**HD** mm (in.)	**AM** mm (in.)	**G** mm (in.)	**G1** mm (in.)	**L1** mm (in.)	**L2** mm (in.)	**ZJ** [2] mm (in.)	**VD** mm (in.)	**WH** mm (in.)	**E** mm (in.)	**TG** mm (in.)
MPAR-*x*3100E-M2A	603.8 (23.77)	178.8 (7.04)	121.5 (4.78)	89.4 (3.52)	80.9 (3.19)	125.7 (4.95)	32.0 (1.26)	34.0 (1.34)	36.0 (1.42)	28.5 (1.12)	177.0 (6.97)	214.0 (8.43)	15.0 (0.59)	37.0 (1.46)	75.5 (2.97)	56.5 (2.22)
MPAR-*x*3200E-M2A	703.8 (27.71)															
MPAR-*x*3300E-M2A	803.8 (31.65)															
MPAR-*x*3400E-M2A	903.8 (35.58)															
MPAR-*x*3600E-M2A	1103.8 (43.46)															
MPAR-*x*3800E-M2A	1303.8 (51.33)															
MPAR-*x*3100H-M2A	574.8 (22.63)	149.8 (5.90)	92.5 (3.64)	98.3 (3.87)	83.9 (3.30)	132.8 (5.23)										
MPAR-*x*3200H-M2A	674.8 (26.57)															
MPAR-*x*3300H-M2A	774.8 (30.50)															
MPAR-*x*3400H-M2A	874.8 (34.44)															
MPAR-*x*3600H-M2A	1074.8 (42.31)															
MPAR-*x*3800H-M2A	1274.8 (50.19)															

(1) If ordering MPAR-A/B3*xxx*E-M4A actuator with brake, add 34.5 mm (1.36 in.) to dimensions L7, LB, and LE.
If ordering MPAR-A/B3*xxx*H-M4A actuator with brake, add 48.5 mm (1.91 in.) to dimensions L7, LB, and LE.

(2) The tolerance for this dimension is ±1.0 mm (0.039 in.).

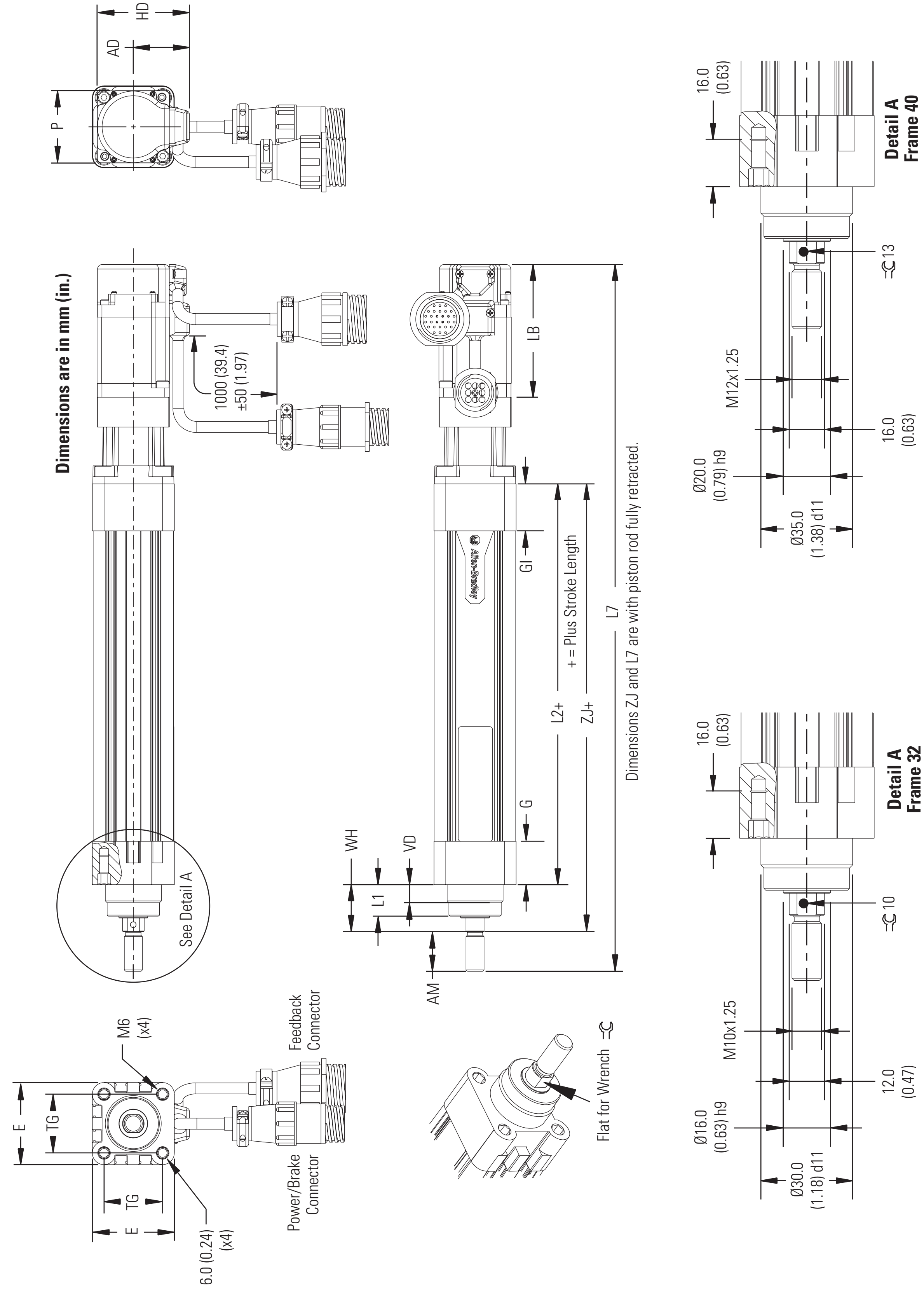
MP-Series Electric Cylinder Dimensions (parallel configuration, frame 63)

Electric Cylinder Cat. No.	**L71** mm (in.)	**LC** mm (in.)	**HC** mm (in.)	**C** [1] mm (in.)	**GC** mm (in.)	**PW** mm (in.)
MPAR-*x*3100E-M2B/D/E	428.0 (16.85)	82.0 (3.23)	225.0 (8.86)	120.0 (4.72)	45.0 (1.77)	110.0 (4.33)
MPAR-*x*3200E-M2B/D/E	528.0 (20.79)					
MPAR-*x*3300E-M2B/D/E	628.0 (24.72)					
MPAR-*x*3400E-M2B/D/E	728.0 (28.66)					
MPAR-*x*3600E-M2B/D/E	928.0 (36.53)					
MPAR-*x*3800E-M2B/D/E	1128.0 (44.41)					
MPAR-*x*3100H-M2B/D/E	428.0 (16.85)					
MPAR-*x*3200H-M2B/D/E	528.0 (20.79)					
MPAR-*x*3300H-M2B/D/E	628.0 (24.72)					
MPAR-*x*3400H-M2B/D/E	728.0 (28.66)					
MPAR-*x*3600H-M2B/D/E	928.0 (36.53)					
MPAR-*x*3800H-M2B/D/E	1128.0 (44.41)					

(1) The tolerance for this dimension is ±1.0 mm (0.039 in.)

TL-Series Electric Cylinder Dimensions

TL-Series Electric Cylinders Dimensions (frame 32 and 40)



TL-Series Electric Cylinder Dimensions (frame 32)

Electric Cylinder Cat. No.	L7 (1) mm (in.)	LB (1) mm (in.)	P mm (in.)	AD mm (in.)	HD mm (in.)	AM mm (in.)	G mm (in.)	G1 mm (in.)	L1 mm (in.)	L2 mm (in.)	ZJ (2) mm (in.)	VD mm (in.)	WH mm (in.)	E mm (in.)	TG mm (in.)
TLAR-A1100B-B2A	391.5 (15.41)	73.5 (2.89)	40.0 (1.57)	31.1 (1.22)	51.1 (2.01)	22.0 (0.87)	24.0 (0.94)	26.0 (1.02)	18.0 (0.71)	122.0 (4.80)	148.0 (5.83)	10.0 (0.39)	26.0 (1.02)	45.5 (1.79)	32.5 (1.28)
TLAR-A1200B-B2A	491.5 (19.35)														
TLAR-A1300B-B2A	591.5 (23.29)														
TLAR-A1400B-B2A	691.5 (27.22)														
TLAR-A1100E-B2A	405.5 (15.96)	76.1 (3.0)	60.0 (2.36)	43.0 (1.69)	73.0 (2.87)										
TLAR-A1200E-B2A	505.5 (19.90)														
TLAR-A1300E-B2A	605.5 (23.84)														
TLAR-A1400E-B2A	705.5 (27.78)														

(1) If ordering TLAR-A1*xxx*B-B4A actuator with brake, add 35.4 mm (1.39 in.) to dimensions L7 and LB.
If ordering TLAR-A2*xxx*E-B4A actuator with brake, add 34.6 mm (1.36 in.) to dimensions L7 and LB.

(2) The tolerance for this dimension is ±1.0 mm (0.039 in.).

TL-Series Electric Cylinder Dimensions (frame 40)

Electric Cylinder Cat. No.	L7 (1) mm (in.)	LB (1) mm (in.)	P mm (in.)	AD mm (in.)	HD mm (in.)	AM mm (in.)	G mm (in.)	G1 mm (in.)	L1 mm (in.)	L2 mm (in.)	ZJ (2) mm (in.)	VD mm (in.)	WH mm (in.)	E mm (in.)	TG mm (in.)
TLAR-A2100C-B2A	436.0 (17.17)	76.1 (3.0)	60.0 (2.36)	43.0 (1.69)	73.0 (2.87)	24.0 (0.94)	28.5 (1.12)	30.0 (1.18)	21.5 (0.85)	146.5 (5.77)	176.5 (6.95)	10.5 (0.41)	30.0 (1.18)	54.0 (2.13)	38.0 (1.50)
TLAR-A2200C-B2A	536.0 (21.10)														
TLAR-A2300C-B2A	636.0 (25.04)														
TLAR-A2400C-B2A	736.0 (28.98)														
TLAR-A2600C-B2A	936.0 (36.85)														
TLAR-A2100F-B2A	457.9 (18.03)	98.1 (3.86)													
TLAR-A2200F-B2A	557.9 (21.96)														
TLAR-A2300F-B2A	657.9 (25.90)														
TLAR-A2400F-B2A	757.9 (29.84)														
TLAR-A2600F-B2A	957.9 (37.71)														

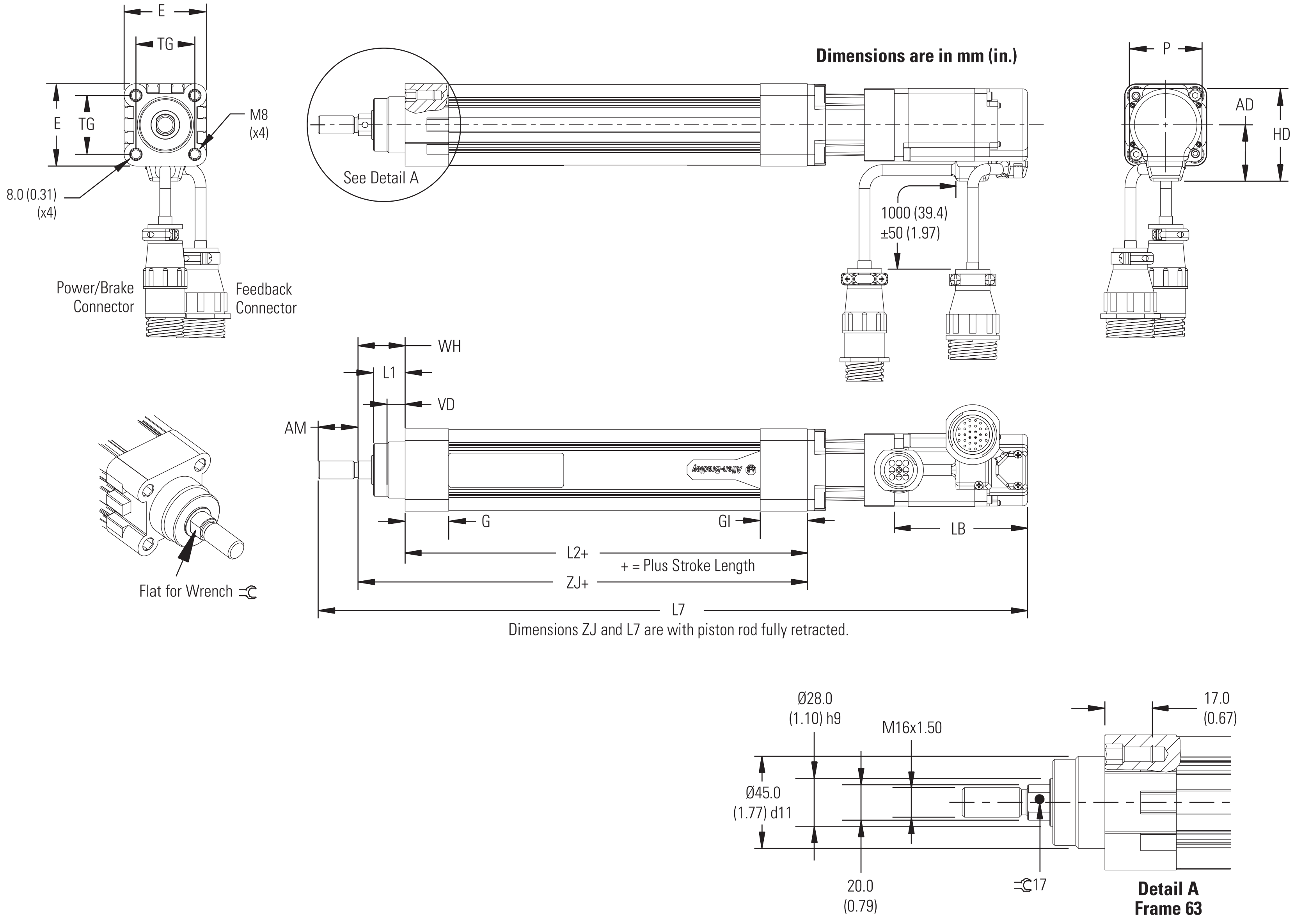
(1) If ordering TLAR-A2*xxxx*-B4A actuator with brake, add 34.6 mm (1.36 in.) to dimensions L7 and LB.

(2) The tolerance for this dimension is ±1.0 mm (0.039 in.).

Actuators are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

TL-Series Electric Cylinders Dimensions (frame 63)

Dimensions are in mm (in.)



Dimensions ZJ and L7 are with piston rod fully retracted.

TL-Series Electric Cylinder Dimensions (frame 63)

Electric Cylinder Cat. No.	L7 (1) mm (in.)	LB (1) mm (in.)	P mm (in.)	AD mm (in.)	HD mm (in.)	AM mm (in.)	G mm (in.)	G1 mm (in.)	L1 mm (in.)	L2 mm (in.)	ZJ (2) mm (in.)	VD mm (in.)	WH mm (in.)	E mm (in.)	TG mm (in.)
TLAR-A3100E-B2A	564.6 (22.23)	144.2 (5.68)	86.0 (3.39)	56.0 (2.20)	99.0 (3.90)	32.0 (1.26)	34.0 (1.34)	36.0 (1.42)	28.5 (1.12)	177.0 (6.97)	214.0 (8.43)	15.0 (0.59)	37.0 (1.46)	75.5 (2.97)	56.5 (2.22)
TLAR-A3200E-B2A	664.6 (26.17)														
TLAR-A3300E-B2A	764.6 (30.10)														
TLAR-A3400E-B2A	864.6 (34.04)														
TLAR-A3600E-B2A	1064.6 (41.91)														
TLAR-A3800E-B2A	1264.6 (49.79)														
TLAR-A3100H-B2A	564.6 (22.23)														
TLAR-A3200H-B2A	664.6 (26.17)														
TLAR-A3300H-B2A	764.6 (30.10)														
TLAR-A3400H-B2A	864.6 (34.04)														
TLAR-A3600H-B2A	1064.6 (41.91)														
TLAR-A3800H-B2A	1264.6 (49.79)														

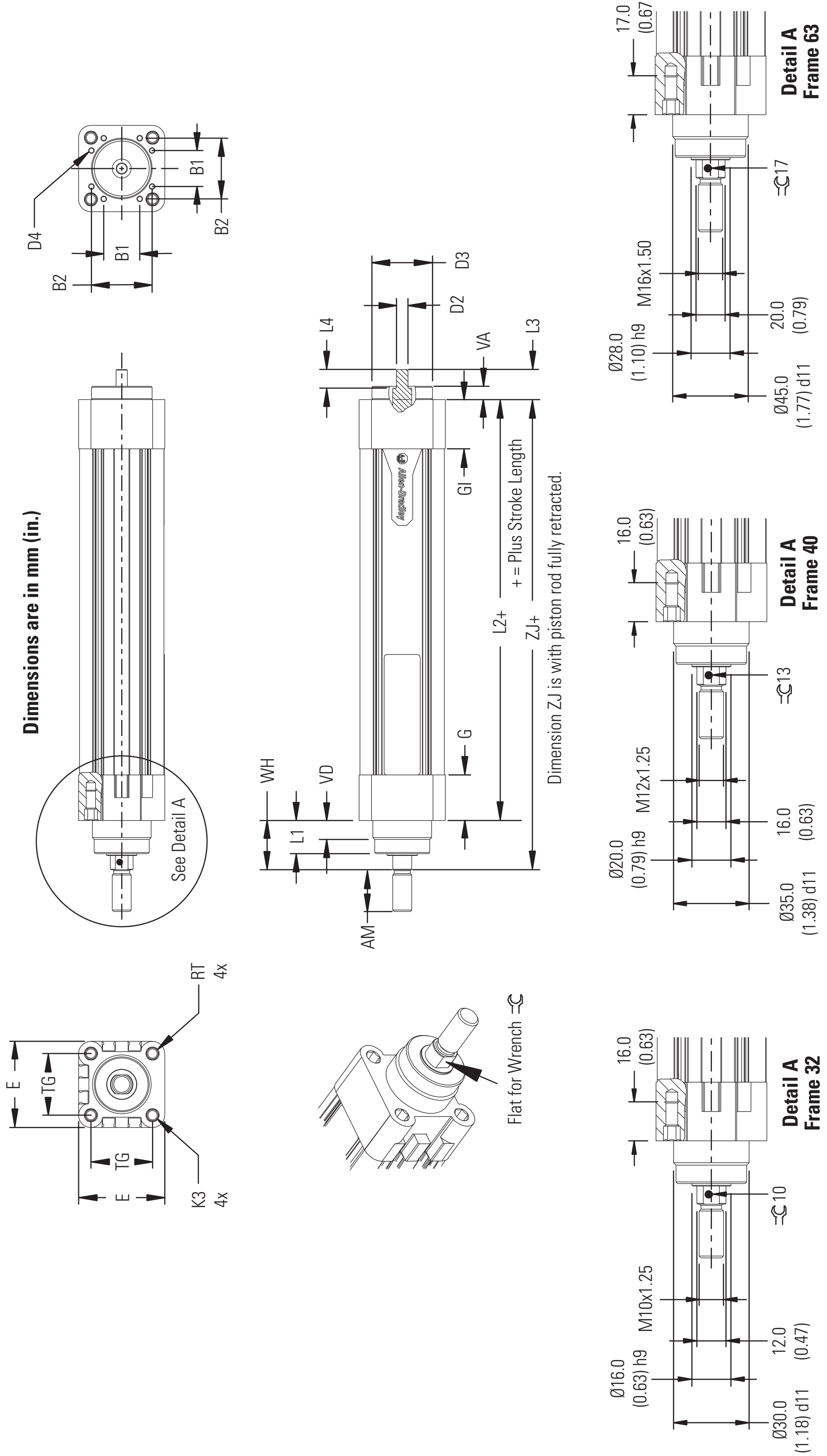
(1) If ordering TLAR-A3*xxxx*-B4A actuator with brake, add 23.0 mm (0.91 in.) to dimensions L7 and LB.

(2) The tolerance for this dimension is ±1.0 mm (0.039 in.).

Actuators are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

Electric Cylinder Dimensions (replacement components)

Electric Cylinder Dimensions (frame 32, 40, and 63)


Dimensions are in mm (in.)

Electric Cylinder Replacement Components

Electric Cylinder Cat. No. [1]	Frame	AM mm (in.)	G mm (in.)	G1 mm (in.)	L1 mm (in.)	L2 mm (in.)	L3 mm (in.)	L4 mm (in.)	ZJ [2] mm (in.)	VA mm (in.)	VD mm (in.)	WH mm (in.)	D2 mm (in.) h6	D3 mm (in.) f7	D4	B1 mm (in.)	B2 mm (in.)	E mm (in.)	RT	TG mm (in.)	K3 mm (in.)
MPAR-X1*xxx*B	32	22.0 (0.87)	24.0 (0.94)	26.0 (1.02)	18.0 (0.71)	122.0 (4.80)	15.9 (0.63)	8.0 (0.31)	148.0 (5.83)	7.0 (0.28)	10.0 (0.39)	26.0 (1.02)	6.0 (0.24)	32.0 (1.26)	M3	19.0 (0.75)	32.0 (1.26)	45.5 (1.79)	M6	32.5 (1.28)	6.0 (0.24)
MPAR-X1*xxx*E																					
MPAR-X2*xxx*C	40	24.0 (0.94)	28.5 (1.12)	30.0 (1.18)	21.5 (0.85)	146.5 (5.77)	18.4 (0.72)	14.0 (0.55)	176.5 (6.95)	7.0 (0.28)	10.5 (0.41)	30.0 (1.18)	8.0 (0.31)	40.0 (1.57)	M4	20.0 (0.79)	42.0 (1.65)	54.0 (2.13)	M6	38.0 (1.50)	6.0 (0.24)
MPAR-X2*xxx*F																					
MPAR-X3*xxx*E	63	32.0 (1.26)	34.0 (1.34)	36.0 (1.42)	28.5 (1.12)	177.0 (6.97)	23.5 (0.93)	17.0 (0.67)	214.0 (8.43)	9.0 (0.35)	15.0 (0.59)	37.0 (1.46)	12.0 (0.47)	60.0 (2.36)	M5	31.0 (1.22)	62.0 (2.44)	75.5 (2.97)	M8	56.5 (2.22)	8.0 (0.31)
MPAR-X3*xxx*H																					

(1) These catalog numbers apply to both MP-Series and TL-Series electric cylinders. For example, the replacement cylinder for MPAR-A1100B-V2A and TLAR-A1100B-B2A is catalog number MPAR-X1100B.

(2) The tolerance for this dimension is ±1.0 mm (0.039 in.).

These replacement cylinders are available in the event of bent rods or other damage to the cylinder. For replacement motors, couplings, and belts, refer to MP-Series and TL-Series Electric Cylinder Replacement Parts Installation Instructions, publication MPAR-IN002.

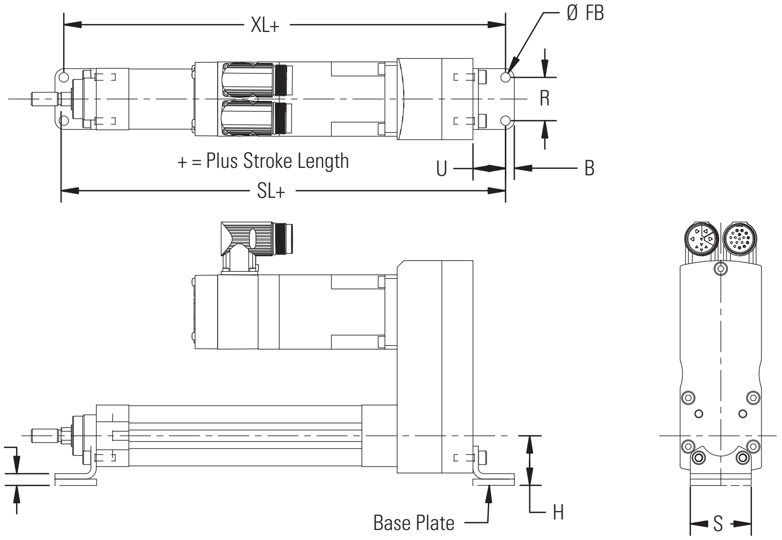
Actuators are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

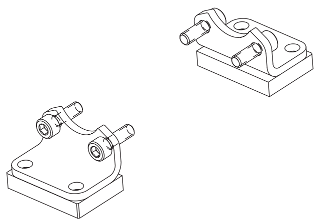
MP-Series and TL-Series Electric Cylinder Mounting Accessories

These accessories apply to MP-Series and TL-Series electric cylinders. Components are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

Foot Mount Attachment

Dimensions are in mm (in.)





Attachment includes:

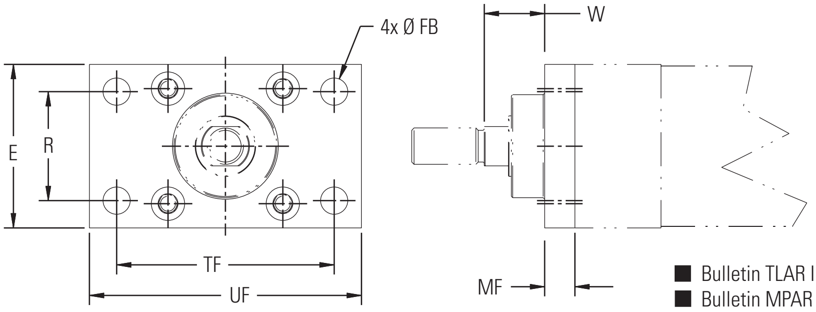
- 2 Foot mount attachments
- 4 Mounting bolts

☐ Bulletin TLAR In-line
☐ Bulletin MPAR In-line
■ Bulletin MPAR Parallel

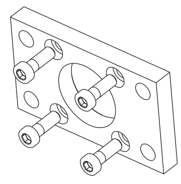
Cat. No. (1)	Cat. No. (2) Corrosion Resistant	Frame	T mm (in.)	FB mm (in.)	H mm (in.)	B mm (in.)	U mm (in.)	R mm (in.)	S mm (in.)	SL+ mm (in.)	XL+ mm (in.)
MPAR-NP174369	MPAR-NP176937	32	4.0 (0.16)	7.0 (0.27)	32.0 (1.26)	6.5 (0.25)	24.0 (0.94)	32.0 (1.26)	45.0 (1.77)	228.0 (8.97)	226.0 (8.90)
MPAR-NP174370	MPAR-NP176938	40	14.0 (3) (0.55)	10.0 (0.39)	46.0 (3) (1.81)	9.0 (0.34)	28.0 (1.10)	36.0 (1.42)	54.0 (2.12)	260.5 (4) (10.25)	258.5 (4) (10.18)
MPAR-NP174372	MPAR-NP176940	63	13.0 (3) (0.51)	10.0 (0.39)	58.0 (3) (2.28)	12.5 (0.49)	32.0 (1.26)	50.0 (1.97)	75.0 (2.95)	328.3 (12.92)	323.3 (12.73)

(1) Material is galvanized steel. Contains no copper, PTFE, or silicone.
(2) Material is high-alloy steel for high-corrosion protection. Contains no copper, PTFE, or silicone.
(3) This dimension includes base plate provided with frame 40 and frame 63 foot mount attachments.
(4) Add 13 mm (0.51 in.) for catalog number MPAR-*x*2*xxx*F-V*xx*.

Flange Mounting Attachment



Dimensions are in mm (in.)



Attachment includes:

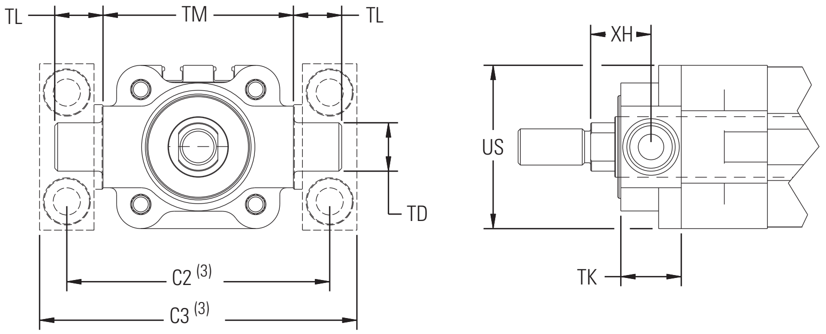
- 1 Flange mounting
- 4 Mounting bolts

■ Bulletin TLAR In-line
■ Bulletin MPAR In-line
■ Bulletin MPAR Parallel

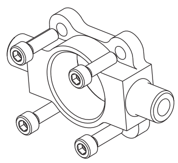
Cat. No. [1]	Cat. No. [2] Corrosion Resistant	Frame	E mm (in.)	FB mm (in.) H13	MF mm (in.)	R mm (in.)	TF mm (in.)	UF mm (in.)	W mm (in.)
MPAR-NA174376	MPAR-NA161846	32	45.0 (1.77)	7.0 (0.28)	10.0 (0.39)	32.0 (1.26)	64.0 (2.52)	80.0 (3.15)	16.0 (0.63)
MPAR-NA174377	MPAR-NA161847	40	54.0 (2.13)	9.0 (0.35)	10.0 (0.39)	36.0 (1.42)	72.0 (2.83)	90.0 (3.54)	20.0 (0.79)
MPAR-NA174379	MPAR-NA161849	63	75.0 (2.95)	9.0 (0.35)	12.0 (0.47)	50.0 (1.97)	100 (3.94)	120 (4.72)	25.0 (0.98)

(1) Material is galvanized steel and subject to moderate corrosion stress. Contains no copper, PTFE, or silicone.
(2) Material is high-alloy steel for environments requiring higher corrosion resistance. Contains no copper, PTFE, or silicone.

Trunnion Flange Attachment



Dimensions are in mm (in.)



Attachment includes:

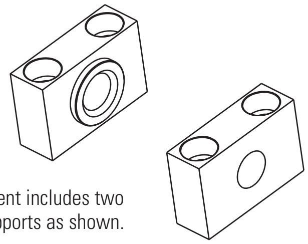
- 1 Trunnion mounting
- 4 Mounting bolts

■ Bulletin TLAR In-line
■ Bulletin MPAR In-line
■ Bulletin MPAR Parallel

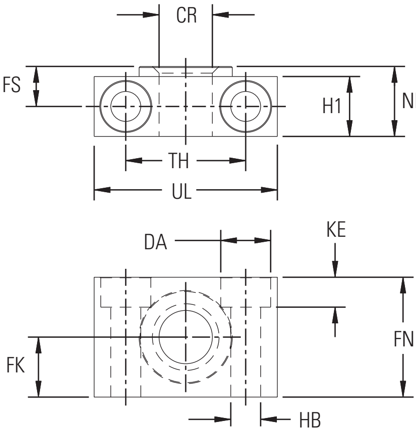
Cat. No. [1]	Cat. No. [2] Corrosion Resistant	Frame	C2 mm (in.)	C3 mm (in.)	TD mm (in.) e9	TK mm (in.)	TL mm (in.)	TM mm (in.)	US mm (in.)	XH mm (in.)
MPAR-NA174411	MPAR-NA161852	32	71.0 (2.80)	86.0 (3.39)	12.0 (0.47)	16.0 (0.63)	12.0 (0.47)	50.0 (1.97)	45.0 (1.77)	18.0 (0.71)
MPAR-NA174412	MPAR-NA161853	40	87.0 (3.43)	105 (4.13)	16.0 (0.63)	20.0 (0.79)	16.0 (0.63)	63.0 (2.48)	54.0 (2.13)	20.0 (0.79)
MPAR-NA174414	MPAR-NA161855	63	116 (4.57)	136 (5.35)	20.0 (0.79)	24.0 (0.94)	20.0 (0.79)	90.0 (3.54)	75.0 (2.95)	25.0 (0.98)

(1) Material is special steel casting and subject to moderate corrosion stress. Contains no copper, PTFE, or silicone.
(2) Material is electrolytically-polished special steel casting for environments requiring higher corrosion resistance. Contains no copper, PTFE, or silicone.
(3) These dimensions are drawn to the trunnion support attachment as shown on page 164 (not included with the trunnion flange attachment).

Trunnion Support Attachments



Attachment includes two trunnion supports as shown.



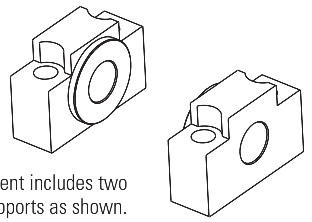
Dimensions are in mm (in.)

- ■ Bulletin TLAR In-line
- ■ Bulletin MPAR In-line
- ■ Bulletin MPAR Parallel

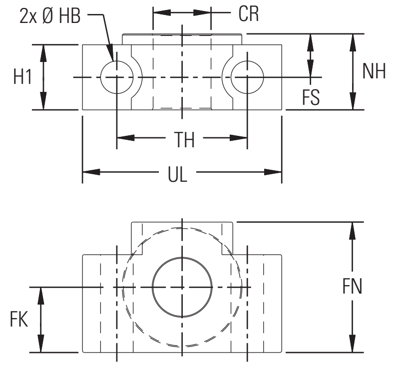
Cat. No. [1]	Frame	CR mm (in.) D11	DA mm (in.) H13	FK [2] mm (in.)	FN mm (in.)	FS mm (in.)	H1 mm (in.)	HB mm (in.) H13	KE mm (in.)	NH mm (in.)	TH [3] mm (in.)	UL mm (in.)
MPAR-NA32959	32	12.0 (0.47)	11.0 (0.43)	15.0 (0.59)	30.0 (1.18)	10.5 (0.41)	15.0 (0.59)	6.6 (0.26)	6.8 (0.27)	18.0 (0.71)	32.0 (1.26)	46.0 (1.81)
MPAR-NA32960	40	16.0 (0.63)	15.0 (0.59)	18.0 (0.71)	36.0 (1.42)	12.0 (0.47)	18.0 (0.71)	9.0 (0.35)	9.0 (0.35)	21.0 (0.83)	36.0 (1.42)	55.0 (2.17)
MPAR-NA32961	63	20.0 (0.79)	18.0 (0.71)	20.0 (0.79)	40.0 (1.57)	13.0 (0.51)	20.0 (0.79)	11.0 (0.43)	11.0 (0.43)	23.0 (0.91)	42.0 (1.65)	65.0 (2.56)

(1) Material is anodized aluminum and subject to moderate corrosion stress. Plain bearing: Polymer. Contains no copper, PTFE, or silicone.
(2) Tolerance for this dimension is ±0.1 mm (±0.0039 in.).
(3) Tolerance for this dimension is ±0.2 mm (±0.0079 in.).

Trunnion Support (corrosion resistant) Attachments



Attachment includes two trunnion supports as shown.



Dimensions are in mm (in.)

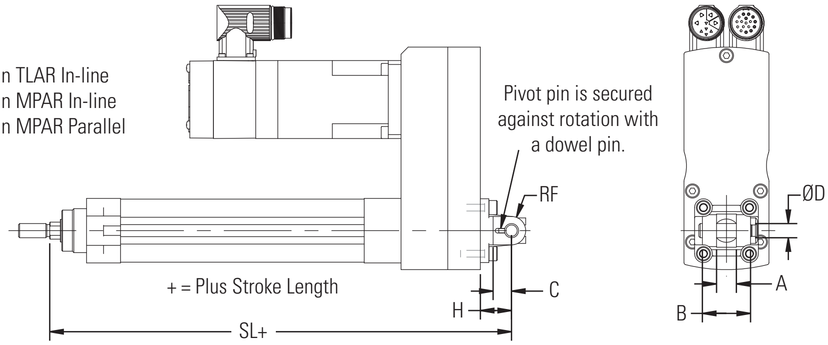
- ■ Bulletin TLAR In-line
- ■ Bulletin MPAR In-line
- ■ Bulletin MPAR Parallel

Cat. No. [1]	Frame	CR mm (in.) D11	FK [2] mm (in.)	FN mm (in.)	FS mm (in.)	H1 mm (in.)	HB mm (in.) H13	NH mm (in.)	TH [3] mm (in.)	UL mm (in.)
MPAR-NA161874	32	12.0 (0.47)	15.0 (0.59)	30.0 (1.18)	10.5 (0.41)	15.0 (0.59)	6.6 (0.26)	18.0 (0.71)	32.0 (1.26)	46.0 (1.81)
MPAR-NA161875	40	16.0 (0.63)	18.0 (0.71)	36.0 (1.42)	12.0 (0.47)	18.0 (0.71)	9.0 (0.35)	21.0 (0.83)	36.0 (1.42)	55.0 (2.17)
MPAR-NA161876	63	20.0 (0.79)	20.0 (0.79)	40.0 (1.57)	13.0 (0.51)	20.0 (0.79)	11.0 (0.43)	23.0 (0.91)	42.0 (1.65)	65.0 (2.56)

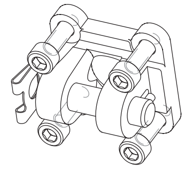
(1) Material is high-alloy steel for environments requiring higher corrosion resistance. Contains no copper, PTFE, or silicone.
(2) Tolerance for this dimension is ±0.1 mm (±0.0039 in.).
(3) Tolerance for this dimension is ±0.2 mm (±0.0079 in.).

Swivel Flange (pin, narrow) Attachment

☐ Bulletin TLAR In-line
☐ Bulletin MPAR In-line
■ Bulletin MPAR Parallel



Dimensions are in mm (in.)



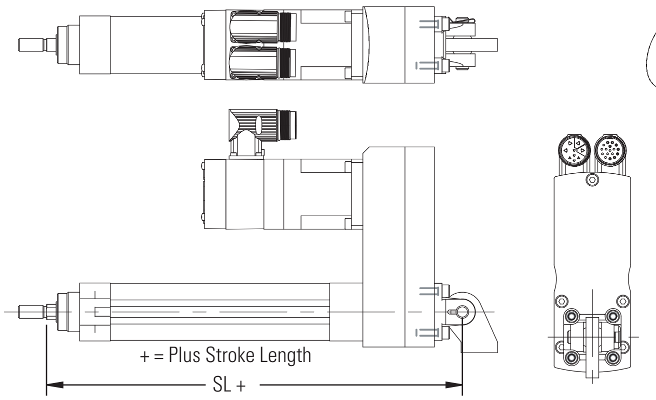
Attachment includes

- 1 Swivel flange
- 1 Pivot pin
- 4 Mounting screws
- 1 Retaining clip

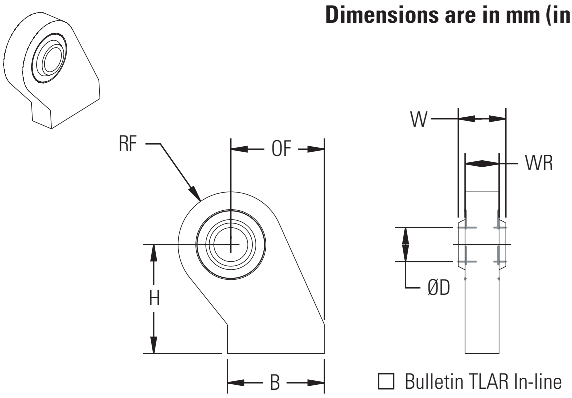
Cat. No. [1]	Frame	D mm (in.)	A mm (in.) H14	B mm (in.) h14	C mm (in.)	H [2] mm (in.)	RF mm (in.)	SL+ mm (in.)
MPAR-NP174383	32	10.0 (0.39)	14.0 (0.55)	34.0 (1.34)	13.0 (0.51)	22.0 (0.86)	10.0 (0.39)	226.0 (8.90)
MPAR-NP174384	40	12.0 (0.47)	16.0 (0.63)	40.0 (1.57)	16.0 (0.63)	25.0 (0.98)	12.0 (0.47)	257.5 (10.14) [3]
MPAR-NP174386	63	16.0 (0.63)	21.0 (0.82)	51.0 (2.01)	21.0 (0.82)	32.0 (0.26)	16.0 (0.63)	328.3 (12.92)

(1) Material is die cast aluminum. Contains no copper, PTFE, or silicone.
(2) Tolerance for this dimension is ±0.2 mm (±0.0079 in.).
(3) Add 13 mm (0.51 in.) for catalog number MPAR-*x*2*xxx*F-V*xx*.

Clevis Foot (weld-on) Attachment



Dimensions are in mm (in.)



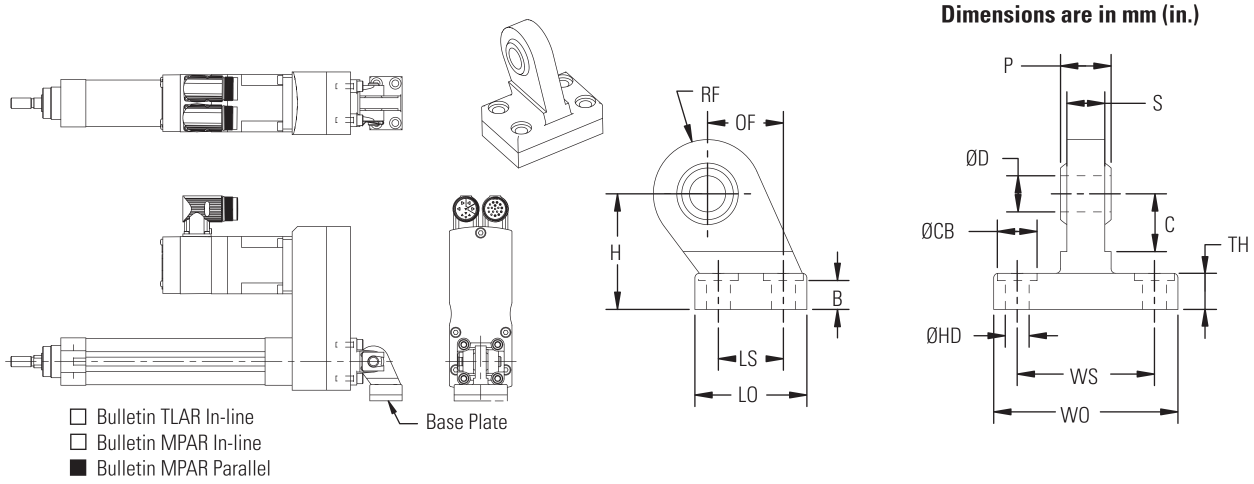
☐ Bulletin TLAR In-line
☐ Bulletin MPAR In-line
■ Bulletin MPAR Parallel

Cat. No. [1]	Frame	D mm (in.)	B mm (in.)	H mm (in.) js14	OF [2] mm (in.)	RF [2] mm (in.)	W [3] mm (in.)	WR [4] mm (in.)	SL+ mm (in.)
MPAR-NP31747	32	10.0 (0.39) [5]	28.5 (1.12)	32.0 (1.26)	27.5 (1.08)	15.5 (0.61)	14.0 (0.55)	10.5 (0.41)	226.0 (8.90)
MPAR-NP31748	40	12.0 (0.47) [6]	33.5 (1.32)	36.0 (1.42)	30.5 (1.20)	17.5 (0.69)	16.0 (0.63)	12.0 (0.47)	257.5 (10.14) [7]
MPAR-NP31750	63	16.0 (0.63) [6]	48.0 (1.89)	50.0 (1.97)	44.5 (1.75)	22.5 (0.88)	21.0 (0.82)	15.0 (0.89)	328.3 (12.92)

(1) Material is die cast aluminum. Contains no copper, PTFE, or silicone.
(2) Tolerance for this dimension is ±0.5 mm (±0.0197 in.).
(3) Tolerance for this dimension is -0.1 mm (±0.0039 in.).
(4) Tolerance for this dimension is -0.2 mm (±0.0079 in.).
(5) Tolerance for this dimension is +0.013 mm (±0.00051 in.).
(6) Tolerance for this dimension is +0.015 mm (±0.00059 in.).
(7) Add 13 mm (0.51 in.) for catalog number MPAR-*x*2*xxx*F-V*xx*.

Clevis Foot (spherical bearing) Attachment

Dimensions are in mm (in.)



Cat. No. [1]	Frame	D mm (in.)	B [2] mm (in.)	C mm (in.)	CB mm (in.)	H mm (in.) js14	HD mm (in.) H13	LO mm (in.)	LS mm (in.) js14
MPAR-NP31740	32	10.0 [3] (0.39)	8.50 (0.33)	N/A	11.0 (0.43)	32.0 (1.26)	6.6 (0.26)	31.0 (1.22)	18.0 (0.71)
MPAR-NP31741	40	12.0 [4] (0.47)	18.5 [5] (0.73)	N/A	11.0 (0.43)	46.0 [5] (1.81)	6.6 (0.26)	35.0 (1.38)	22.0 (0.86)
MPAR-NP31743	63	16.0 [4] (0.63)	18.5 [5] (0.73)	27.0 (1.06)	15.0 (0.59)	58.0 [5] (2.28)	9.0 (0.35)	50.0 (1.97)	35.0 (1.38)

(1) Frame 32 and 40 material is high-alloy steel. Frame 63 material is nodular graphite cast iron. Contains no copper, PTFE, or silicone.
(2) Tolerance for this dimension is ±0.8 mm (±0.0315 in.).
(3) Tolerance for this dimension is +0.013 mm (±0.00051 in.).
(4) Tolerance for this dimension is +0.015 mm (±0.00059 in.).
(5) This dimension includes base plate provided with frame 40 and frame 63 foot mount attachments.

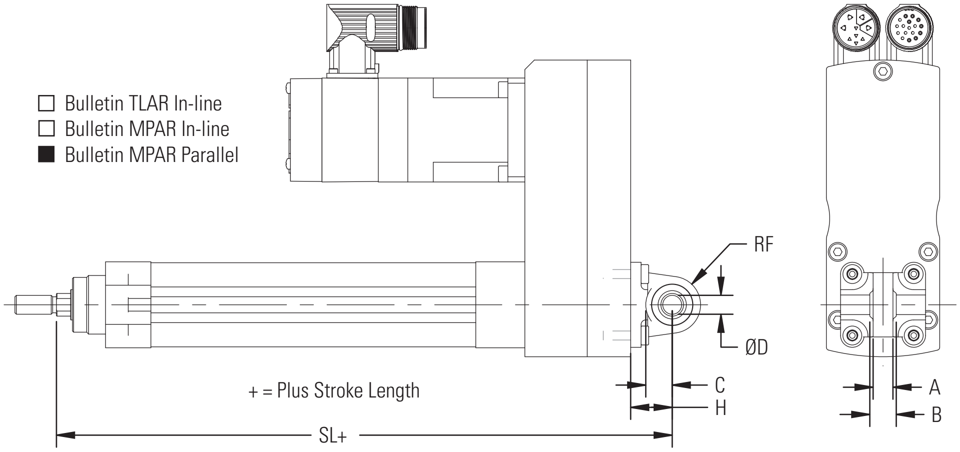
Cat. No.	Frame	OF mm (in.) js14	P [1] mm (in.) js14	RF mm (in.)	S [2] mm (in.)	TH mm (in.)	WO mm (in.)	WS mm (in.) js14
MPAR-NP31740	32	21.0 (0.82)	14.0 (0.55)	15.0 (0.59)	10.5 (0.41)	10.0 [3] (0.39)	51.0 (2.01)	38.0 (1.49)
MPAR-NP31741	40	24.0 (0.94)	16.0 (0.63)	17.0 (0.67)	12.0 (0.47)	20.0 [3] [4] (0.79)	54.0 (2.12)	41.0 (1.61)
MPAR-NP31743	63	37.0 (1.45)	21.0 (0.82)	22.0 (0.86)	15.0 (0.59)	20.0 [5] [4] (0.79)	67.0 (2.64)	52.0 (2.05)

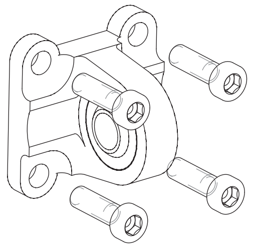
(1) Tolerance for this dimension is -0.1 mm (±0.0039 in.).
(2) Tolerance for this dimension is -0.2 mm (±0.0079 in.).
(3) Tolerance for this dimension is ±0.2 mm (±0.0079 in.).
(4) This dimension includes base plate provided with frame 40 and frame 63 foot mount attachments.
(5) Tolerance for this dimension is ±0.6 mm (±0.0236 in.).

Swivel Flange (spherical bearing) Attachment

Dimensions are in mm (in.)

- ☐ Bulletin TLAR In-line
- ☐ Bulletin MPAR In-line
- ■ Bulletin MPAR Parallel





Attachment includes
- 1 Swivel flange with spherical bearing
- 4 Mounting screws

Cat. No. [(1)]	Frame	D mm (in.) H7	A [(2)] mm (in.)	B mm (in.)	C mm (in.)	H [(2)] mm (in.)	RF mm (in.)	SL+ mm (in.)
MPAR-NP174397	32	10.0 (0.39)	10.5 (0.41)	14.0 (0.55)	13.0 (0.51)	22.0 (0.86)	15.0 (0.59)	226.0 (8.90)
MPAR-NP174398	40	12.0 (0.47)	12.0 (0.47)	16.0 (0.63)	16.0 (0.63)	25.0 (0.98)	17.0 (0.67)	257.5 [(3)] (10.14)
MPAR-NP174400	63	16.0 (0.63)	15.0 (0.59)	21.0 (0.82)	21.0 (0.82)	32.0 (0.26)	22.0 (0.86)	328.3 (12.92)

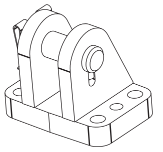
(1) Material is die cast aluminum. Contains no copper, PTFE, or silicone.

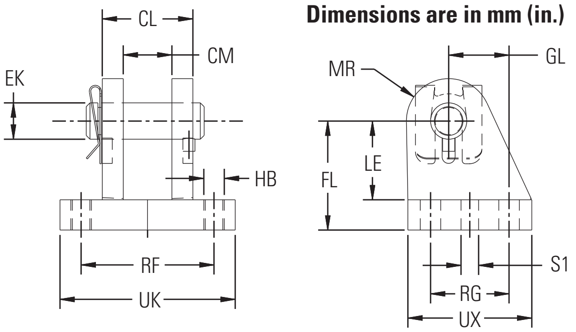
(2) Tolerance for this dimension is ±0.2 mm (±0.0079 in.).

(3) Add 13 mm (0.51 in.) for catalog number MPAR-*x*2*xxx*F-V*xx*.

Clevis Foot (pin) Attachment

Dimensions are in mm (in.)





- ■ Bulletin TLAR In-line
- ■ Bulletin MPAR In-line
- ■ Bulletin MPAR Parallel

Attachment includes:
- 1 Clevis foot
- 1 Pivot pin [(2)]
- 1 Retaining clip

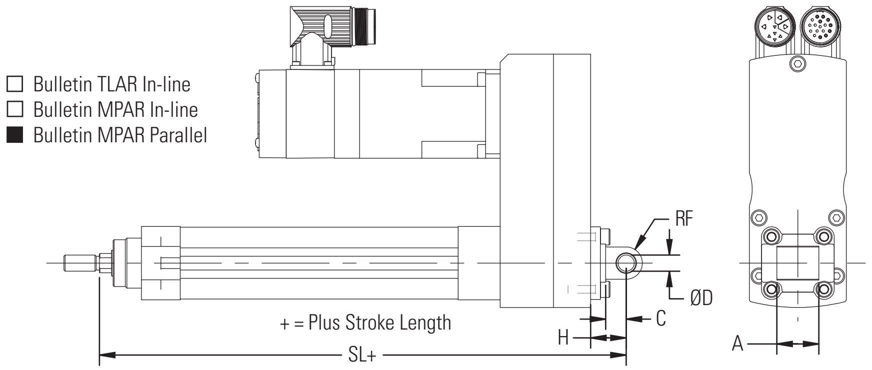
Cat. No. [(1)]	Frame	CL mm (in.)	CM mm (in.)	EK mm (in.)	FL mm (in.)	GL mm (in.)	HB mm (in.)	LE mm (in.)	MR mm (in.)	RF mm (in.)	RG mm (in.)	S1 mm (in.)	UK mm (in.)	UX mm (in.)
MPAR-NA31761	32	28.0 (1.10)	14.1 (0.56)	10.0 (0.39)	32.0 (1.26)	16.0 (0.63)	6.8 (0.27)	24.0 (0.94)	12.0 (0.47)	42.0 (1.65)	20.0 (0.79)	4.8 (0.19)	56.0 (2.20)	36.0 (1.42)
MPAR-NA31762	40	30.0 (1.18)	16.1 (0.63)	12.0 (0.47)	36.0 (1.42)	20.0 (0.79)	6.8 (0.27)	26.0 (1.02)	14.0 (0.55)	44.0 (1.73)	26.0 (1.02)	5.8 (0.23)	58.0 (2.28)	41.5 (1.63)
MPAR-NA31764	63	40.0 (1.57)	21.1 (0.83)	16.0 (0.63)	50.0 (1.97)	25.0 (0.98)	9.0 (0.35)	38.0 (1.50)	17.0 (0.67)	56.0 (2.20)	31.0 (1.22)	7.8 (0.31)	70.0 (2.76)	47.0 (1.85)

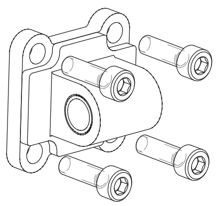
(1) Material is nodular graphite cast iron for environments requiring moderate corrosion resistance. Contains no copper, PTFE, or silicone.

(2) The pivot pin is secured against rotation with a dowel pin.

Swivel Flange Attachment

Dimensions are in mm (in.)





Attachment includes

- 1 Swivel flange
- 4 Mounting screws

Cat. No. [(1)]	Frame	D mm (in.) H9	A [(2)] mm (in.) h12	C mm (in.)	H [(2)] mm (in.)	RF mm (in.)	SL+ mm (in.)
MPAR-NP174404	32	10.0 (0.39)	26.0 (1.02)	13.0 (0.51)	22.0 (0.86)	10.0 (0.39)	226.0 (8.90)
MPAR-NP174405	40	12.0 (0.47)	28.0 (1.10)	16.0 (0.63)	25.0 (0.98)	12.0 (0.47)	257.5 (3) (10.14)
MPAR-NP174407	63	16.0 (0.63)	40.0 (1.57)	21.0 (0.82)	32.0 (0.26)	16.0 (0.63)	328.3 (12.92)

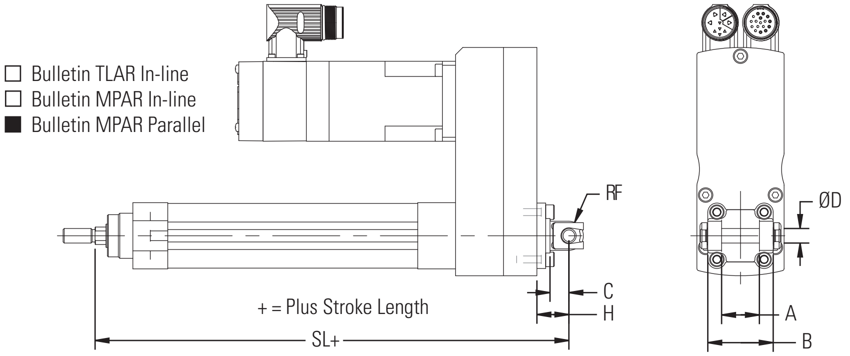
(1) Material is die cast aluminum. Contains no copper, PTFE, or silicone.

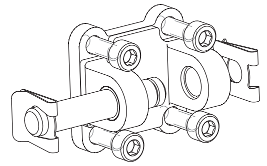
(2) Tolerance for this dimension is ±0.2 mm (±0.0079 in.).

(3) Add 13 mm (0.51 in.) for catalog number MPAR-*x*2*xxx*F-V*xx*.

Swivel Flange (pin, wide) Attachment

Dimensions are in mm (in.)





Attachment includes

- 1 Swivel flange
- 1 Pivot pin
- 2 Retaining clips
- 4 Mounting screws

Cat. No. [(1)]	Cat. No. [(2)] Corrosion Resistant	Frame	D mm (in.) e8	A mm (in.) H14	B mm (in.) h14	C mm (in.)	H [(3)] mm (in.)	RF mm (in.)	SL+ mm (in.)
MPAR-NP174390	MPAR-NP176944	32	10.0 (0.39)	26.0 (1.02)	45.0 (1.77)	13.0 (0.51)	22.0 (0.86)	10.0 (0.39)	226.0 (8.90)
MPAR-NP174391	MPAR-NP176945	40	12.0 (0.47)	28.0 (1.10)	52.0 (2.05)	16.0 (0.63)	25.0 (0.98)	12.0 (0.47)	257.5 (4) (10.14)
MPAR-NP174393	MPAR-NP176947	63	16.0 (0.63)	40.0 (1.57)	70.0 (2.75)	21.0 (0.82)	32.0 (0.26)	16.0 (0.63)	328.3 (12.92)

(1) Material is die cast aluminum. Contains no copper, PTFE, or silicone.

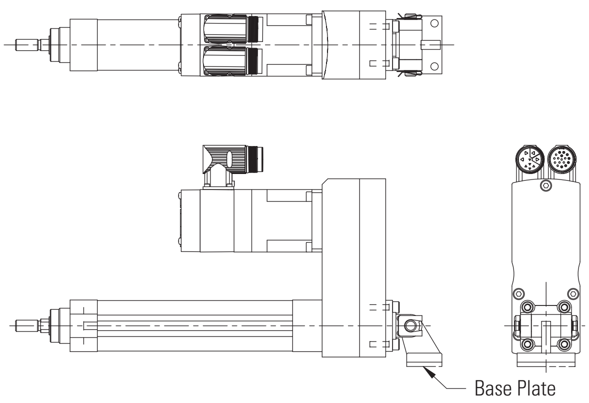
(2) Material is die cast aluminum, high-corrosion protection. Contains no copper, PTFE, or silicone.

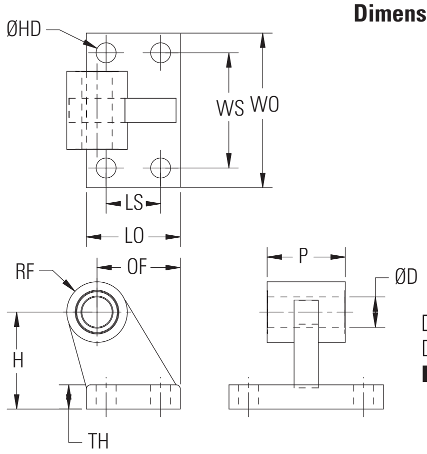
(3) Tolerance for this dimension is ±0.2 mm (±0.0079 in.).

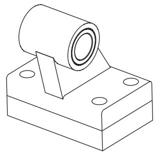
(4) Add 13 mm (0.51 in.) for catalog number MPAR-*x*2*xxx*F-V*xx*.

Clevis Foot Attachment

Dimensions are in mm (in.)







Attachment includes 1 clevis foot.

☐ Bulletin TLAR In-line
☐ Bulletin MPAR In-line
■ Bulletin MPAR Parallel

Cat. No. (1)	Cat. No. (2) Corrosion Resistant	Frame	D mm (in.) D11	H mm (in.)	HD mm (in.) H13	LO mm (in.)	LS mm (in.)	OF mm (in.)	P (3) mm (in.)	RF mm (in.)	TH mm (in.)	WO mm (in.)	WS mm (in.)
MPAR-NP33890	MPAR-NP161840	32	10.0 (0.39)	32.0 (1.26)	6.6 (0.26)	31.0 (1.22)	18.0 (0.71)	21.0 (0.82)	25.8 (1.01)	10.0 (0.39)	8.0 (0.31)	51.0 (2.01)	38.0 (1.49)
MPAR-NP33891	MPAR-NP161841	40	12.0 (0.47)	46.0 (4) (1.81)	6.6 (0.26)	35.0 (1.38)	22.0 (0.86)	24.0 (0.94)	27.8 (1.09)	11.0 (0.43)	20.0 (4) (0.79)	54.0 (2.12)	41.0 (1.61)
MPAR-NP33893	MPAR-NP161843	63	16.0 (0.63)	58.0 (4) (2.28)	9.0 (0.35)	50.0 (1.97)	35.0 (1.38)	37.0 (1.45)	39.8 (1.56)	15.0 (1.56)	20.0 (4) (0.79)	67.0 (2.64)	52.0 (2.05)

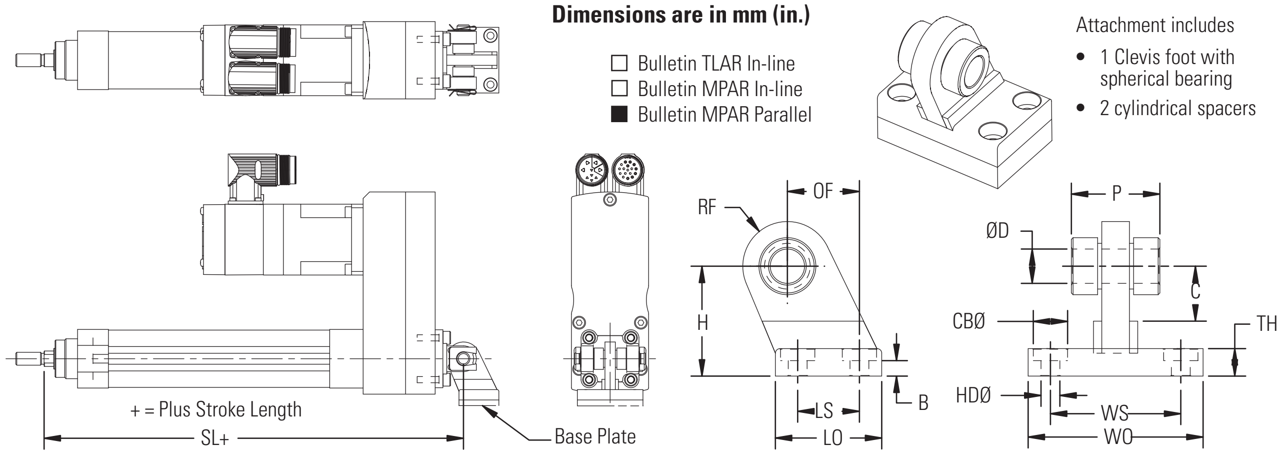
(1) Frame 32 and 40 material is high-alloy steel. Frame 63 material is nodular graphite cast iron. Contains no copper, PTFE, or silicone.

(2) Material is spacial steel casting, high corrosion protection stainless steel. Contains no copper, PTFE, or silicone.

(3) Tolerance for this dimension is -0.4 mm (-0.0157 in.).

(4) This dimension includes base plate provided with frame 40 and frame 63 foot mount attachments.

Clevis Foot (spherical bearing) Attachment



Dimensions are in mm (in.)

☐ Bulletin TLAR In-line
☐ Bulletin MPAR In-line
■ Bulletin MPAR Parallel

Attachment includes

- 1 Clevis foot with spherical bearing
- 2 cylindrical spacers

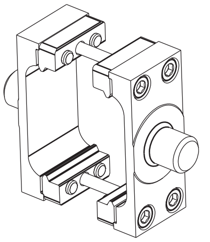
Cat. No. (1)	Frame	D (2) mm (in.)	B mm (in.)	C mm (in.)	CB mm (in.)	H (3) mm (in.)	HD mm (in.)	LO mm (in.)	LS mm (in.)
MPAR-NP5561	32	10.0 (0.39)	4.50 (0.18)	16.0 (0.63)	10.0 (0.39)	32.0 (1.26)	5.5 (0.21)	31.0 (1.22)	18.0 (0.71)
MPAR-NP5562	40	12.0 (0.47)	14.30 (4) (0.56)	20.0 (0.79)	10.0 (0.39)	46.0 (4) (1.81)	5.5 (0.21)	35.0 (1.38)	22.0 (0.86)
MPAR-NP5564	63	16.0 (0.63)	13.2 (4) (0.52)	30.0 (1.18)	11.0 (0.43)	58.0 (4) (2.28)	6.6 (0.26)	50.0 (1.97)	35.0 (1.38)

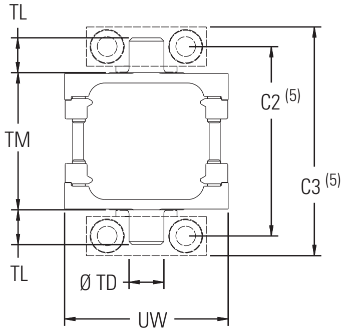
(1) Frame 32 and 40 material is high-alloy steel. Frame 63 material is nodular graphite cast iron. Contains no copper, PTFE, or silicone.
(2) Tolerance for this dimension is -0.008 mm (-0.00031 in.).
(3) Tolerance for this dimension is ±0.5 mm (±0.0197 in.).
(4) This dimension includes base plate provided with frame 40 and frame 63 foot mount attachments.

Cat. No.	Frame	OF mm (in.)	P mm (in.)	RF (1) mm (in.)	SL+ mm (in.)	TH mm (in.)	WO mm (in.)	WS mm (in.)
MPAR-NP5561	32	21.0 (0.82)	25.8 (1.01)	13.0 (0.51)	226.0 (8.90)	10.0 (0.39)	51.0 (2.01)	38.0 (1.49)
MPAR-NP5562	40	24.0 (0.94)	27.8 (1.09)	15.0 (0.59)	257.5 (2) (10.14)	20.0 (3) (0.79)	54.0 (2.12)	41.0 (1.61)
MPAR-NP5564	63	37.0 (1.45)	39.8 (1.56)	20.0 (0.79)	328.3 (12.92)	20.0 (3) (0.79)	67.0 (2.64)	52.0 (2.05)

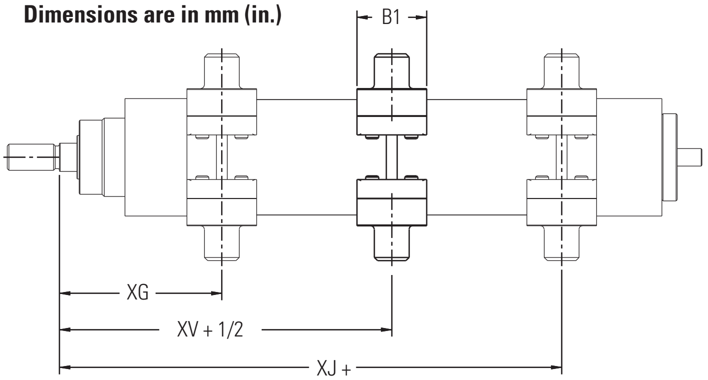
(1) Tolerance for this dimension is -0.1 mm (±0.0039 in.).
(2) Add 13 mm (0.51 in.) for catalog number MPAR-*x*2*xxx*F-V*xx*.
(3) This dimension includes base plate provided with frame 40 and frame 63 foot mount attachments.

Trunnion Mounting Kit





Dimensions are in mm (in.)



+ = Plus Stroke Length
+ 1/2 = Plus Stroke Length

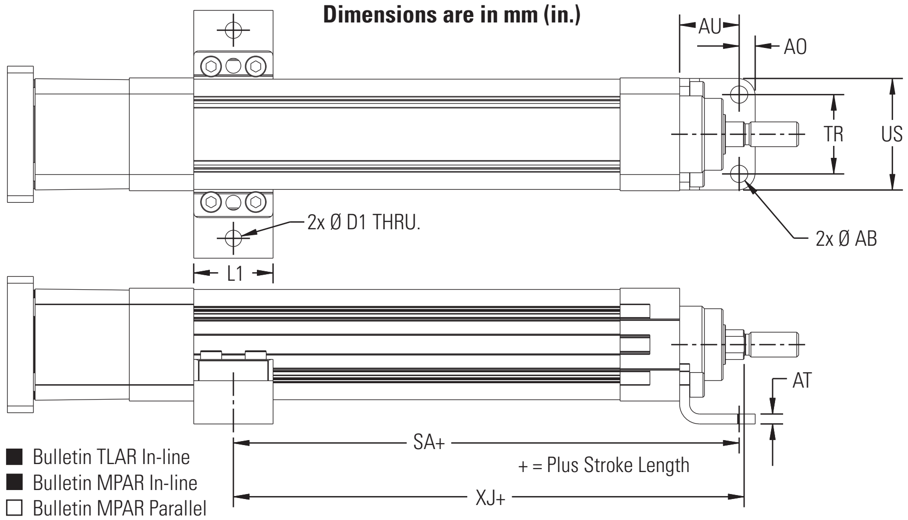
- ■ Bulletin TLAR In-line
- ■ Bulletin MPAR In-line
- ■ Bulletin MPAR Parallel

Cat. No. (1) (2)	**Frame**	**B1** mm (in.)	**C2** mm (in.)	**C3** mm (in.)	**TD** mm (in.) e9	**TL** mm (in.)	**TM** mm (in.)	**UW** mm (in.)	**XG** mm (in.)	**XJ** mm (in.)	**XV** mm (in.)	**Tightening Torque** N•m (lb•in)
MPAR-NA163525	32	30.0 (1.18)	71.0 (2.80)	86.0 (3.39)	12.0 (0.47)	12.0 (0.47)	50.0 (1.97)	65.0 (2.56)	65.0 (2.56)	107 (4.21)	86.0 (3.39)	4.0 (35.4) (3)
MPAR-NA163526	40	32.0 (1.26)	87.0 (3.43)	105 (4.13)	16.0 (0.63)	16.0 (0.63)	63.0 (2.48)	75.0 (2.95)	74.5 (2.93)	130.5 (5.14)	102.5 (4.04)	8.0 (70.7) (3)
MPAR-NA163528	63	41.0 (1.61)	116 (4.57)	136 (5.35)	20.0 (0.79)	20.0 (0.79)	90.0 (3.54)	105 (4.13)	91.5 (3.60)	157.5 (6.20)	124.5 (4.90)	18.0 (159) (4)

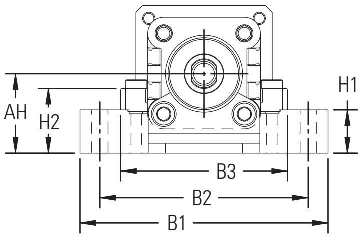
(1) Material is tempered steel and subject to moderate corrosion stress.
(2) You can attach the trunnion mounting kit anywhere along the cylinder barrel.
(3) Tolerance for this tightening torque value is +1.0 N•m (+8.8 lb•in).
(4) Tolerance for this tightening torque value is +2.0 N•m (+17.7 lb•in).
(5) These dimensions are drawn to the trunnion support attachment as shown on page 164 (not included with the trunnion flange attachment).

Foot Mounting Kit

Dimensions are in mm (in.)



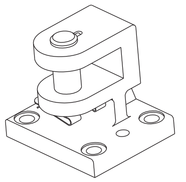
Frame	ØAB mm (in.)	ØD1 mm (in.)
32	7.0 (0.28)	6.6 (0.26)
40	10.0 (0.39)	9.0 (0.35)
63	10.0 (0.39)	9.0 (0.35)



Cat. No. [1]	Frame	AH mm (in.)	AO mm (in.)	AT mm (in.)	AU mm (in.)	B1 mm (in.)	B2 mm (in.)	B3 mm (in.)	H1 mm (in.)	H2 mm (in.)	TR mm (in.)	US mm (in.)	XJ mm (in.)	SA mm (in.)	L1 mm (in.)
MPAR-NA174991	32	32.0 (1.26)	6.5 (0.26)	4.0 (0.16)	24.0 (0.94)	100 (3.94)	84.0 (3.31)	66.1 (2.60)	17.5 (0.69)	26.1 (1.03)	32.0 (1.26)	45.0 (1.77)	106 (4.17)	104 (4.09)	32.0 (1.26)
MPAR-NA174992	40	36.0 (1.42)	9.0 (0.35)	4.0 (0.16)	28.0 (1.10)	130 (5.12)	108 (4.25)	85.2 (3.35)	15.7 (0.62)	23.3 (0.92)	36.0 (1.42)	54.0 (2.13)	129.5 (5.10)	127.5 (5.02)	34.0 (1.34)
MPAR-NA174993	63	50.0 (1.97)	12.5 (0.49)	5.0 (0.20)	32.0 (1.26)	150 (5.91)	128 (5.04)	104.8 (4.13)	22.9 (0.90)	30.4 (1.20)	50.0 (1.97)	75.0 (2.95)	157.5 (6.20)	152.5 (6.00)	41.0 (1.61)

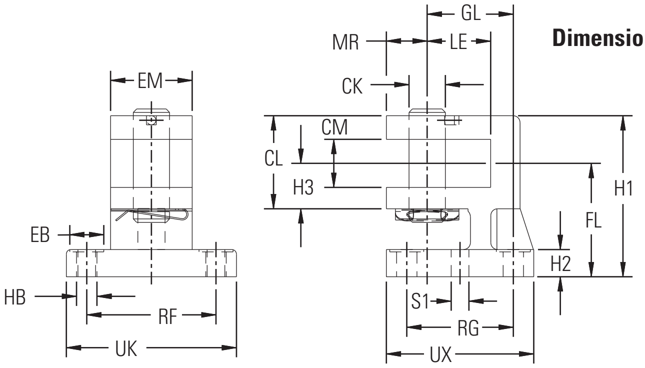
(1) Material is galvanized steel and subject to low corrosion stress. Contains no copper, PTFE, or silicone.

Clevis Foot (right-angle) Attachment



Attachment includes:

- 1 Clevis foot
- 1 Pivot pin (5)
- 1 Retaining clip



Dimensions are in mm (in.)

- ■ Bulletin TLAR In-line
- ■ Bulletin MPAR In-line
- ■ Bulletin MPAR Parallel

Cat. No. (1)	**Frame**	**CK** mm (in.) h9	**CL** mm (in.)	**CM** (2) mm (in.)	**EB** mm (in.)	**EM** mm (in.)	**FL** (3) mm (in.)	**GL** (3) mm (in.)	**H1** (4) mm (in.)	**H2** mm (in.)	**H3** mm (in.)	**HB** mm (in.)	**LE** mm (in.)
MPAR-NA31768	32	10.0 (0.39)	27.0 (1.06)	14.2 (0.56)	11.0 (0.43)	25.0 (0.98)	32.0 (1.26)	22.0 (0.87)	45.0 (1.77)	9.0 (0.35)	14.0 (0.55)	6.6 (0.26)	18.0 (0.71)
MPAR-NA31769	40	12.0 (0.47)	31.0 (1.22)	16.2 (0.64)	11.0 (0.43)	25.0 (0.98)	36.0 (1.42)	22.0 (0.87)	52.0 (2.05)	9.0 (0.35)	15.0 (0.59)	6.6 (0.26)	22.0 (0.87)
MPAR-NA31771	63	16.0 (0.63)	41.0 (1.61)	21.2 (0.83)	15.0 (0.59)	36.0 (1.42)	50.0 (1.97)	38.0 (1.50)	71.0 (2.80)	12.0 (0.47)	20.0 (0.79)	9.0 (0.35)	28.0 (1.10)

(1) Material is nodular graphite cast iron for environments requiring moderate corrosion resistance.
(2) Tolerance for this dimension is +0.2 mm (+0.008 in.).
(3) Tolerance for this dimension is ±0.3 mm (±0.012 in.) for Frame 32. Tolerance is js14 for Frame 40 and Frame 63.
(4) Tolerance for this dimension is ±0.5 mm (±0.019 in.) for Frame 32 and Frame 40.
(5) The pivot pin is secured against rotation with a dowel pin.

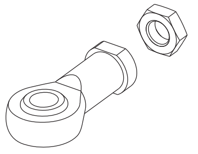
Cat. No.	**Frame**	**MR** mm (in.)	**RF** (1) mm (in.)	**RG** (1) mm (in.)	**S1** mm (in.)	**UK** mm (in.)	**UX** mm (in.)
MPAR-NA31768	32	12.5 (0.49)	40.0 (1.57)	29.0 (1.14)	4.8 (0.19)	56.0 (2.20)	45.0 (1.77)
MPAR-NA31769	40	15.0 (0.59)	40.0 (1.57)	29.0 (1.14)	5.8 (0.23)	56.0 (2.20)	45.0 (1.77)
MPAR-NA31771	63	18.0 (0.71)	57.0 (2.24)	47.0 (1.85)	7.8 (0.31)	75.0 (2.95)	65.0 (2.56)

(1) Tolerance for this dimension is ±0.3 mm (±0.012 in.) for Frame 32. Tolerance is js14 for Frame 40 and Frame 63.

MP-Series and TL-Series Electric Cylinder Rod-end Accessories

These accessories apply to MP-Series and TL-Series electric cylinders. Components are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

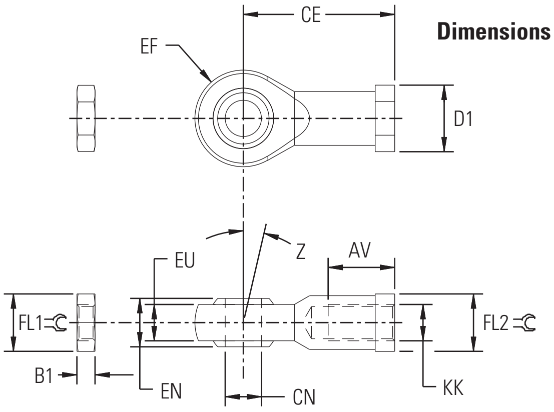
Rod-eye Attachment



Attachment includes:

- 1 Rod eye
- 1 Hex nut

■ Bulletin TLAR In-line
■ Bulletin MPAR In-line
■ Bulletin MPAR Parallel

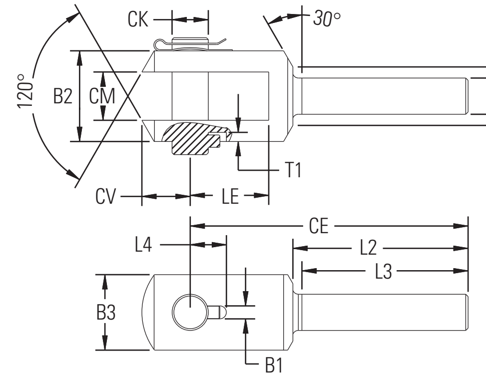


Dimensions are in mm (in.)

Cat. No. (1)	Cat. No. (2) Corrosion Resistant	Frame	AV (3) mm (in.)	B1 mm (in.)	CE mm (in.)	CN mm (in.) H7	D1 mm (in.)	EF (4) mm (in.)	EN mm (in.)	EU mm (in.)	Z angle	FL1 mm 🔧	FL2 mm 🔧	KK
MPAR-NE9261	MPAR-NE195582	32	20.0 (0.79)	5.0 (0.20)	43.0 (1.69)	10.0 (0.39)	19.0 (0.75)	14.0 (0.55)	14.0 (0.55)	10.5 (0.41)	13°	17	17	M10x1.25
MPAR-NE9262	MPAR-NE195583	40	22.0 (0.87)	6.0 (0.24)	50.0 (1.97)	12.0 (0.47)	22.0 (0.87)	16.0 (0.63)	16.0 (0.63)	12.0 (0.47)	13°	19	19	M12x1.25
MPAR-NE9263	MPAR-NE195584	63	28.0 (1.10)	8.0 (0.31)	64.0 (2.52)	16.0 (0.63)	27.0 (1.06)	21.0 (0.83)	21.0 (0.83)	15.0 (0.59)	15°	24	22	M16x1.50

(1) Material is galvanized steel and subject to moderate corrosion stress. Contains no copper, PTFE, or silicone.
(2) Material is high-alloy steel for environments requiring higher corrosion resistance. Contains no copper, PTFE, or silicone.
(3) Tolerance for this dimension is -2.0 mm (-0.079 in.).
(4) Tolerance for this dimension is ±0.5 mm (±0.020 in.).

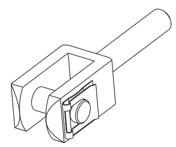
Rod Clevis (threaded rod) Attachment



Dimensions are in mm (in.)

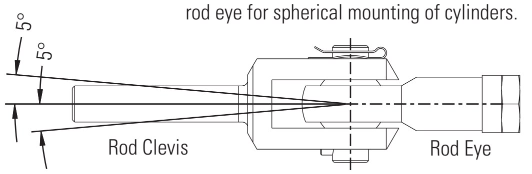
Attachment includes:

- 1 Rod clevis
- 1 Pivot pin (2)
- 1 Retaining clip



■ Bulletin TLAR In-line
■ Bulletin MPAR In-line
■ Bulletin MPAR Parallel

Use the rod clevis in combination with the rod eye for spherical mounting of cylinders.



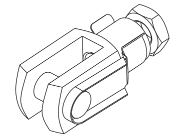
Cat. No. (1)	Frame	B1 mm (in.)	B2 mm (in.) d12	B3 mm (in.)	CE mm (in.)	CK mm (in.) F7/h9	CM mm (in.) B12	CV mm (in.)	D1 mm (in.)	L2 mm (in.)	L3 mm (in.)	L4 mm (in.)	LE mm (in.)	T1 mm (in.)	KK
MPAR-NE32954	32	3.3 (0.13)	28.0 (1.10)	20.0 (0.79)	78.0 (3.07)	10.0 (0.39)	14.0 (0.55)	12.0 (0.47)	18.0 (0.71)	53.0 (2.09)	50.0 (1.97)	11.0 (0.43)	20.0 (0.79)	3.0 (0.12)	M10x1.25
MPAR-NE10767	40	4.3 (0.17)	30.0 (1.18)	25.0 (0.98)	92.0 (3.62)	12.0 (0.47)	16.0 (0.63)	16.0 (0.63)	19.0 (0.75)	58.0 (2.28)	55.0 (2.17)	12.0 (0.47)	26.0 (1.02)	3.0 (0.12)	M12x1.25
MPAR-NE10768	63	4.3 (0.17)	40.0 (1.57)	35.0 (1.38)	108 (4.25)	16.0 (0.63)	21.0 (0.83)	21.0 (0.83)	24.0 (0.94)	65.0 (2.56)	62.0 (2.44)	14.0 (0.55)	31.0 (1.22)	3.0 (0.12)	M16x1.50

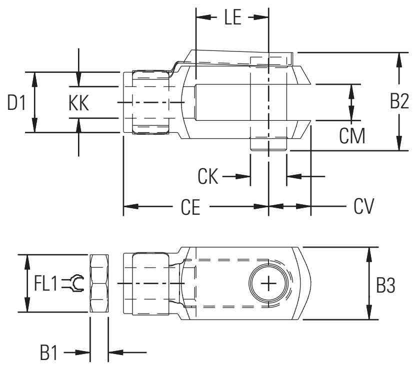
(1) Material is galvanized steel and subject to moderate corrosion stress. Contains no copper, PTFE, or silicone.

(2) The pivot pin is secured against rotation with a dowel pin.

Rod Clevis Attachment

Dimensions are in mm (in.)





Attachment includes:

- 1 Rod clevis
- 1 Hinged spring pin
- 1 Hex nut

■ Bulletin TLAR In-line
■ Bulletin MPAR In-line
■ Bulletin MPAR Parallel

Cat. No. (1)	Frame	B1 mm (in.)	B2 mm (in.)	B3 mm (in.)	CE (2) mm (in.)	CK (3) mm (in.) H9	CM mm (in.)	CV mm (in.)	D1 mm (in.)	LE (4) mm (in.)	FL1 mm	KK
MPAR-NE6144	32	5.0 (0.20)	26.0 (1.02)	20.0 (0.79)	40.0 (1.57)	10.0 (0.39)	10.0 (5) (0.39)	12.0 (0.47)	18.0 (0.71)	20.0 (0.79)	17	M10x1.25
MPAR-NE6145	40	6.0 (0.24)	31.0 (1.22)	24.0 (0.94)	48.0 (1.89)	12.0 (0.47)	12.0 (6) (0.47)	14.0 (0.55)	20.0 (0.79)	24.0 (0.94)	19	M12x1.25
MPAR-NE6146	63	8.0 (0.31)	39.0 (1.54)	32.0 (1.26)	64.0 (2.52)	16.0 (0.63)	16.0 (6) (0.63)	19.0 (0.75)	26.0 (1.02)	32.0 (1.26)	24	M16x1.50

(1) Material is galvanized steel and subject to moderate corrosion stress. Contains no copper, PTFE, or silicone.

(2) Tolerance for this dimension is ±0.4 (0.016 in.).

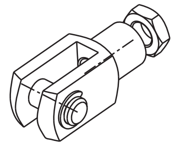
(3) Tolerance for this dimension is H9.

(4) Tolerance for this dimension is ±0.5 (0.019 in.).

(5) Tolerance for this dimension is B13.

(6) Tolerance for this dimension is a range between +0.7…+0.15 mm (+0.027…0.006 in.).

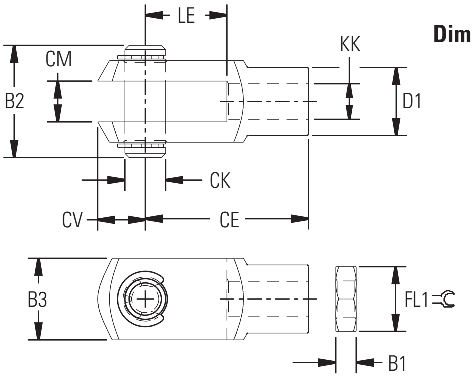
Rod Clevis Attachment (corrosion resistant)



Attachment includes:

- 1 Rod clevis
- 1 Pivot pin
- 1 Hex nut

Dimensions are in mm (in.)

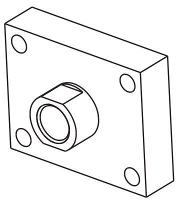


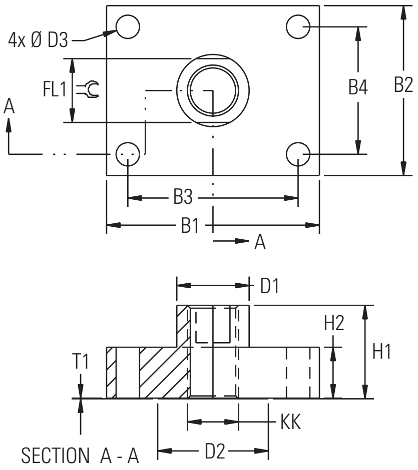
■ Bulletin TLAR In-line
■ Bulletin MPAR In-line
■ Bulletin MPAR Parallel

Cat. No. [1]	Frame	B1 mm (in.)	B2 mm (in.)	B3 mm (in.)	CE [2] mm (in.)	CK mm (in.) H9	CM mm (in.)	CV mm (in.)	D1 mm (in.)	LE [3] mm (in.)	FL1 mm	KK
MPAR-NE13569	32	5.0 (0.20)	27.0 (1.06)	20.0 (0.79)	40.0 (1.57)	10.0 (0.39)	10.0 [4] (0.39)	12.0 (0.47)	18.0 (0.71)	20.0 (0.79)	16	M10x1.25
MPAR-NE13570	40	6.0 (0.24)	33.0 (1.30)	24.0 (0.94)	48.0 (1.89)	12.0 (0.47)	12.0 [5] (0.47)	14.0 (0.55)	20.0 (0.79)	24.0 (0.94)	18	M12x1.25
MPAR-NE13571	63	8.0 (0.31)	43.0 (1.69)	32.0 (1.26)	64.0 (2.52)	16.0 (0.63)	16.0 [5] (0.63)	19.0 (0.75)	26.0 (1.02)	32.0 (1.26)	24	M16x1.50

(1) Material is high-alloy steel for environments requiring higher corrosion resistance. Contains no copper, PTFE, or silicone.
(2) Tolerance for this dimension is ±0.4 (0.016 in.).
(3) Tolerance for this dimension is ±0.5 (0.019 in.).
(4) Tolerance for this dimension is B13.
(5) Tolerance for this dimension is a range between +0.7…+0.15 mm (+0.027…0.006 in.).

Coupling Piece Attachment





Dimensions are in mm (in.)

Attachment includes:

- 1 Flange plate
- 1 Threaded coupling

■ Bulletin TLAR In-line
■ Bulletin MPAR In-line
■ Bulletin MPAR Parallel

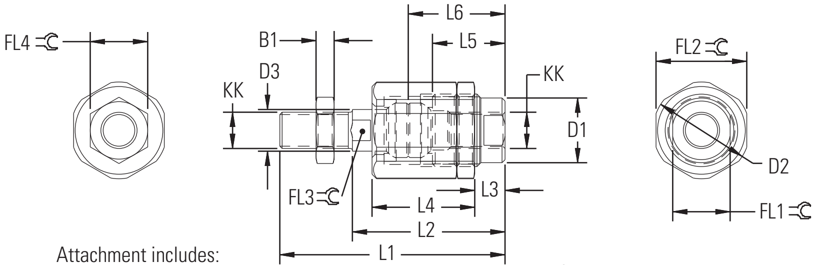
Cat. No. (1) (2)	**Frame**	**B1** mm (in.)	**B2** mm (in.)	**B3** mm (in.)	**B4** mm (in.)	**D1** (3) mm (in.)	**D2** mm (in.)	**D3** mm (in.) H13	**H1** mm (in.)	**H2** mm (in.)	**T1** mm (in.)	**FL1** mm	**KK**
MPAR-NE36125	32	40.0 (1.57)	35.0 (1.38)	30.0 (1.18)	25.0 (0.98)	17.0 (0.67)	26.0 (1.02)	5.5 (0.22)	20.0 (0.79)	10.0 (0.39)	0.10 (0.004)	15	M10x1.25
MPAR-NE36126	40	50.0 (1.97)	40.0 (1.57)	40.0 (1.57)	30.0 (1.18)	17.0 (0.67)	26.0 (1.02)	5.5 (0.22)	22.0 (0.87)	12.0 (0.47)	0.10 (0.004)	15	M12x1.25
MPAR-NE36127	63	60.0 (2.36)	45.0 (1.77)	48.0 (1.89)	33.0 (1.30)	22.0 (0.87)	34.0 (1.34)	6.6 (0.26)	25.0 (0.98)	12.0 (0.47)	0.10 (0.004)	19	M16x1.50

(1) Material is galvanized steel and subject to moderate corrosion stress. Contains no copper, PTFE, or silicone.

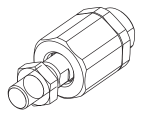
(2) Coupling is for non-rotating piston rods with male threads. You can use these coupling pieces to connect a cylinder with a non-rotating piston rod to another component with a defined orientation, without rotating either the cylinder or the other component.

(3) Tolerance for this dimension is -0.2 mm (-0.008 in.).

Self-aligning Rod Coupler Attachment



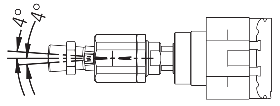
Dimensions are in mm (in.)



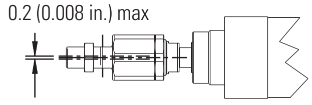
Attachment includes:

- 1 Self-aligning rod coupler
- 1 Hex nut

■ Bulletin TLAR In-line
■ Bulletin MPAR In-line
■ Bulletin MPAR Parallel



Angular Compensation



Radial Compensation of Central Axis

Cat. No. [1]	Frame	B1 mm (in.)	D1 mm (in.)	D2 mm (in.)	D3 mm (in.)	L1 mm (in.)	L2 mm (in.)	L3 mm (in.)	L4 mm (in.)	L5 mm (in.)	L6 mm (in.)
MPAR- NE6140	32	5.0 (0.20)	21.4 (0.84)	32.0 (1.26)	13.8 (0.54)	69.5 (2.74)	49.5 (1.95)	9.0 (0.35)	34.0 (1.34)	23.0 (0.91)	31.0 (1.22)
MPAR-NE6141	40	6.0 (0.24)	21.4 (0.84)	32.0 (1.26)	13.8 (0.54)	74.5 (2.93)	50.5 (1.99)	10.0 (0.39)	34.0 (1.34)	24.0 (0.94)	32.0 (1.26)
MPAR-NE6142	63	8.0 (0.31)	33.8 (1.33)	45.0 (1.77)	22.0 (0.87)	103 (4.06)	71.0 (2.80)	10.0 (0.39)	53.0 (2.09)	32.0 (1.26)	44.5 (1.75)

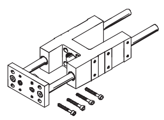
Cat. No. [1]	Frame	FL1 mm	FL2 mm	FL3 mm	FL4 mm	Radial Deviation mm (in.)	KK
MPAR-NE6140	32	19	30	12	17	±0.7 (±0.027)	M10x1.25
MPAR-NE6141	40	19	30	12	19	±0.7 (±0.027)	M12x1.25
MPAR-NE6142	63	30	41	19	24	±1.0 (±0.039)	M16x1.50

(1) Material is galvanized steel and subject to moderate corrosion stress. Contains no copper, PTFE, or silicone.

MP-Series and TL-Series Electric Cylinder Rod Guide Specifications

Rod guides protect ISO standard cylinders from torsion when subjected to radial or torsional side loads. They provide high-precision guidance for workpiece handling and other handling applications.

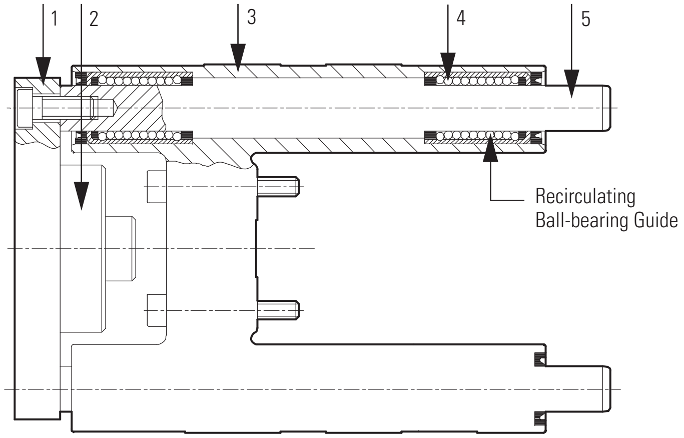
Rod Guides for Fixed Strokes

Rod Guide	Cat. No. Frame 32	Stroke mm (in.)
	MPAR-NE34494	10…100 (0.39…3.94)
	MPAR-NE34496	10…200 (0.39…7.87)
	MPAR-NE34497	10…320 (0.39…12.60)
	MPAR-NE150290	10…400 (0.39…15.75)

Cat. No. Frame 40	Stroke mm (in.)
MPAR-NE34500	10…100 (0.39…3.94)
MPAR-NE34502	10…200 (0.39…7.87)
MPAR-NE34504	10…320 (0.39…12.60)
MPAR-NE150291	10…400 (0.39…15.75)
MPAR-NE34505	10…500 (0.39…19.68)

Cat. No. Frame 63	Stroke mm (in.)
MPAR-NE34514	10…100 (0.39…3.94)
MPAR-NE34516	10…200 (0.39…7.87)
MPAR-NE34518	10…320 (0.39…12.60)
MPAR-NE34519	10…400 (0.39…15.75)
MPAR-NE34520	10…500 (0.39…19.68)

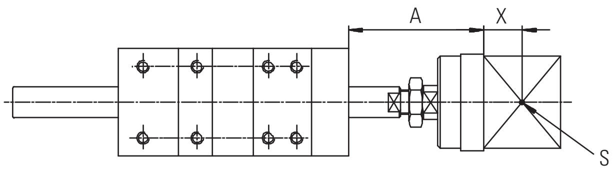
Material Specifications



Item	Attribute	Value (1)
1	Yoke Plate	Aluminium
2	Coupling	Steel
3	Guide	Aluminium
4	Bearing	Steel
5	Guide Rods	Steel

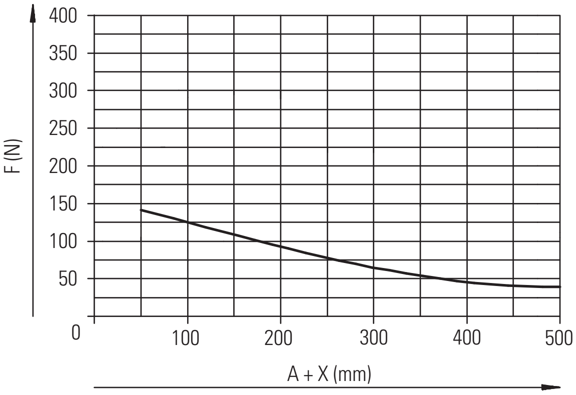
(1) Contains no copper, PTFE, or silicone.

Maximum Working Load (F) as a Function of Cantilever Extension A

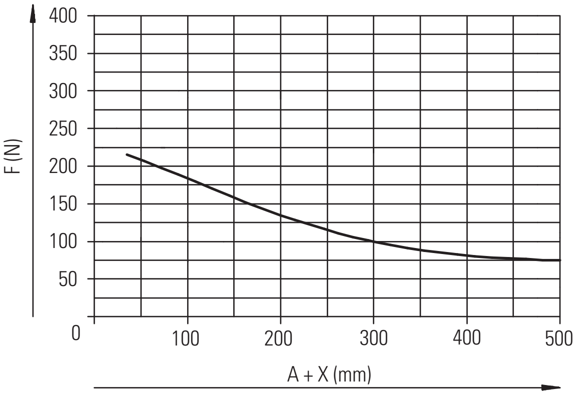


A = Cantilever Extension
X = Distance for Center of Gravity of Working Load
S = Center of Gravity of Working Load

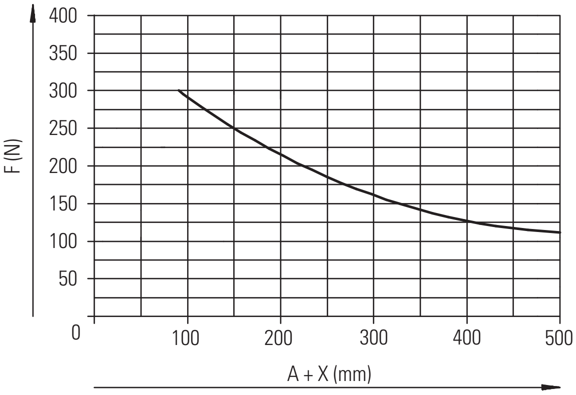
Maximum Load Forces (frame 32)



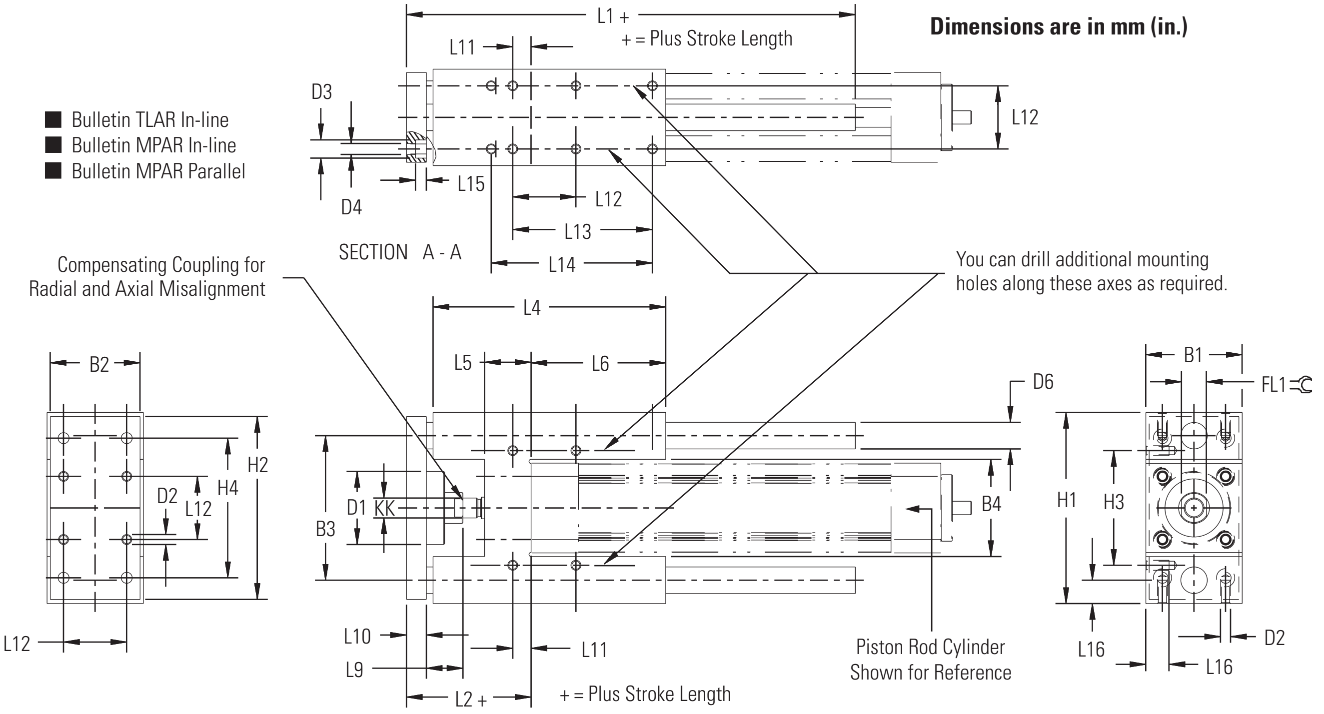
Maximum Load Forces (frame 40)



Maximum Load Forces (frame 63)



Rod Guide Dimensions



Dimensions are in mm (in.)

Frame	**B1** [1] mm (in.)	**B2** mm (in.)	**B3** [2] mm (in.)	**B4** [3] mm (in.)	**D1** mm (in.)	**D2**	**D3** mm (in.)	**D4** mm (in.)	**D6** mm (in.) h6	**H1** mm (in.)	**H2** mm (in.)	**H3** [2] mm (in.)	**H4** [2] mm (in.)	**KK**
32	50.0 (1.97)	45.0 (1.77)	74.0 (2.91)	50.5 (1.99)	44.0 (1.73)	M6	11.0 (0.43)	6.6 (0.26)	12.0 (0.47)	97.0 [4] (3.82)	90.0 (3.54)	61.0 (2.40)	78.0 (3.07)	M10x1.25
40	58.0 (2.28)	54.0 (2.13)	87.0 (3.43)	58.5 (2.30)	44.0 (1.73)	M6	11.0 (0.43)	6.6 (0.26)	16.0 (0.63)	115 [4] (4.53)	110 (4.33)	69.0 (2.72)	84.0 (3.31)	M12x1.25
63	85.0 (3.35)	80.0 (3.15)	119 (4.69)	85.5 (3.37)	60.0 (2.36)	M8	15.0 (0.59)	9.0 (0.35)	20.0 (0.79)	152 [5] (5.98)	145 (5.71)	100 (3.94)	105 (4.13)	M16x1.50

(1) Tolerance for this dimension is -0.3 mm (-0.012 in.).
(2) Tolerance for this dimension is ±0.2 mm (±0.008 in.).
(3) Tolerance for this dimension is ±0.3 mm (±0.012 in.).
(4) Tolerance for this dimension is -0.4 mm (-0.016 in.).
(5) Tolerance for this dimension is -0.5 mm (-0.019 in.).

Frame	**L1** mm (in.)	**L2** mm (in.)	**L4** mm (in.)	**L5** mm (in.)	**L6** mm (in.)	**L9** mm (in.)	**L10** mm (in.)	**L11** mm (in.)	**L12** [1] mm (in.)	**L13** [1] mm (in.)	**L14** [1] mm (in.)	**L15** mm (in.)	**L16** mm (in.)	**FL1** mm
32	155 (6.10)	67.0 [2] (2.64)	125 (4.92)	24.0 (0.94)	76.0 (2.99)	20.0 (0.79)	12.0 (0.47)	4.3 (0.17)	32.5 (1.28)	70.3 (2.77)	78.0 (3.07)	6.5 (0.26)	12.0 (0.47)	15
40	170 (6.69)	75.0 [2] (2.95)	140 (5.51)	28.0 (1.10)	81.0 (3.19)	22.0 (0.87)	12.0 (0.47)	11.0 (0.43)	38.0 (1.50)	84.0 (3.31)	–	6.5 (0.26)	14.0 (0.55)	15
63	220 (8.66)	89.0 [3] (3.50)	182 (7.17)	34.0 (1.34)	111 (4.37)	25.0 (0.98)	15.0 (0.59)	15.3 (0.60)	56.5 (2.22)	105 (4.13)	–	9.0 (0.35)	16.0 (0.63)	19

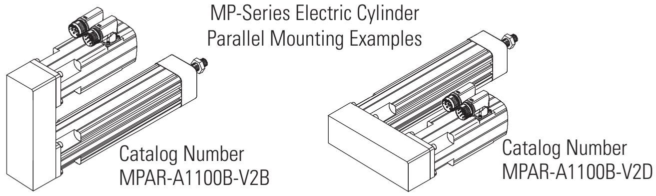
(1) Tolerance for this dimension is ±0.2 mm (±0.008 in.).
(2) Tolerance for this dimension is +5.0 mm (+0.197 in.).
(3) Tolerance for this dimension is +10.0 mm (+0.394 in.).

MP-Series and TL-Series Electric Cylinder Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your actuator. For questions regarding product availability, contact your Allen-Bradley distributor.

MP-Series and TL-Series Electric Cylinder Catalog Numbers

***x x* AR - *x x x x x x - x x* A**


MP-Series Electric Cylinder
Parallel Mounting Examples

Catalog Number
MPAR-A1100B-V2B

Catalog Number
MPAR-A1100B-V2D

Motor Mounting (1)
A = Axial (in-line)
B = Top, Bulletin MPAR parallel
D = Left, Bulletin MPAR parallel
E = Right, Bulletin MPAR parallel

Holding Brake (1)
2 = No Brake
4 = 24V DC Brake

Feedback (1)
B = Multi-turn, 17-bit (131,072 counts) absolute, battery backed, serial protocol (TL-Series, all frame sizes)
M = Multi-turn, 1024 sin/cos, absolute encoder, Hiperface protocol (MP-Series, frame size 63)
V = Multi-turn, 128 sin/cos, absolute encoder, Hiperface protocol (MP-Series, frame size 32 and 40)

Mechanical Drive/Screw Lead, Motor Type (2)
B = 3.0 mm/rev (0.118 in./rev)
C = 5.0 mm/rev (0.197 in./rev)
E = 10.0 mm/rev (0.394 in./rev)
F = 12.7 mm/rev (0.50 in./rev)
H = 20.0 mm/rev (0.787 in./rev)

Rod Stroke Length (2)
100 = 100 mm (3.94 in.)
200 = 200 mm (7.87 in.)
300 = 300 mm (11.81 in.)
400 = 400 mm (15.75 in.)
600 = 600 mm (23.62 in.)
800 = 800 mm (31.50 in.)

Actuator Frame Size
1 = 32
2 = 40
3 = 63

Voltage Class/Designator
A = 200V (applies to MP-Series and TL-Series actuators)
B = 400V (applies to MP-Series actuators)
X = Actuator cylinder replacement part (refer to Actuator Cylinders on page 145 for ordering examples)

Actuator Type
AR = Actuator Rod

Actuator Series
MP = MP-Series
TL = TL-Series

(1) This field does not apply to actuator cylinder replacement parts.
(2) Not all combinations are available. Only the configurations as listed in MP-Series Electric Cylinders Performance Specifications, on page 143, and TL-Series Electric Cylinders Performance Specifications, on page 143, are available.

MP-Series and TL-Series Electric Cylinder Accessory Catalog Numbers

MP AR - *xx xxxxxx*

Accessory Item Number

Accessory Type
NA = Axial (in-line) Mounting Accessory
NP = Parallel Mounting Accessory
NE = Rod-end Accessory

Actuator Type
AR = Actuator Rod

Actuator Series
MP = MP-Series or TL-Series Actuator Accessory

MP-Series Heavy Duty Electric Cylinders

The MP-Series (Bulletin MPAI) heavy-duty electric cylinders are compact, lightweight, high-force actuators that serve as a cost-effective alternative to fluid power solutions. These ready-to-install electric cylinders are energy efficient and provide machine flexibility, including precise, multi-point (positioning or force) motion profiles, which can be customized for movements with smooth startup and soft stops.

For drive compatibility, refer to Servo Drives on page 14.

MP-Series Heavy Duty Electric Cylinder Features

- Fully integrated servo motor and mechanics, these ready to install cylinders contribute to reductions in mechanical design engineering, assembly, wiring, and commissioning time.
- Standard and front trunnion mount configurations are available in 83 mm and 110 mm frames sizes.
- State-of-the-art design features ballscrew or roller screw construction with linear stroke lengths up to 450 mm (18 in.), absolute high-resolution feedback, and speeds up to 559 mm/s (22 in./s).
- Lower operating costs… energy efficient actuators for 230V or 460V motion control systems.
- Extend and retract accurately. End with a soft touch.
- Linear feed force of up to 8896 N (2000 lb).
- Sizing and engineering with compatible servo drives is made easy with Motion Analyzer software and online CAD files.
- Commissioning is simplified by the use of standard Allen-Bradley motor power and feedback cables.
- Full set-up and programming support by using Allen-Bradley Logix controller platforms. RSLogix 5000 and Ultraware software make setup and commissioning fast and easy.
- Closed loop control (absolute encoder feedback, standard).
- IP67 environmental rating, standard.

Motor Connector/Cable Compatibility

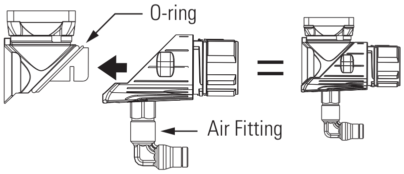
MP-Series (Bulletin MPAI) Heavy Duty Electric Cylinders

- MPAI-A/B*xxxxx* electric cylinders
- Receives M4 and M7 cable plugs (remove the O-ring for M7)
- Attach M7 cable plug with one-quarter turn

- 2090-CFBM7DF-CEAA*xx* (standard) flying-lead, feedback cables
- 2090-CFBM7DD-CEAA*xx* (standard) drive-end connector, feedback cables
- 2090-CFBM7DF-CEAF*xx* (continuous-flex) flying-lead, feedback cables
- 2090-CFBM7DD-CEAF*xx* (continuous-flex) drive-end connector, feedback cables
- 2090-CPWM7DF-16AA*xx* (standard) power-only cables
- 2090-CPBM7DF-16AA*xx* (standard) power/brake cables
- 2090-CPWM7DF-16AF*xx* (continuous-flex) power-only cables
- 2090-CPBM7DF-16AF*xx* (continuous-flex) power/brake cables

MP-Series Heavy Duty Electric Cylinder Accessories and Options

- 24V DC holding brake.
- Rod-end attachments, mounting attachments, and anti-rotation accessories.
- Positive Air Pressure kit (catalog number MPF-7-AIR-PURGE) is mounted on the feedback connector to provide positive air pressure to further reduce the chance of contamination inside the electric cylinder.



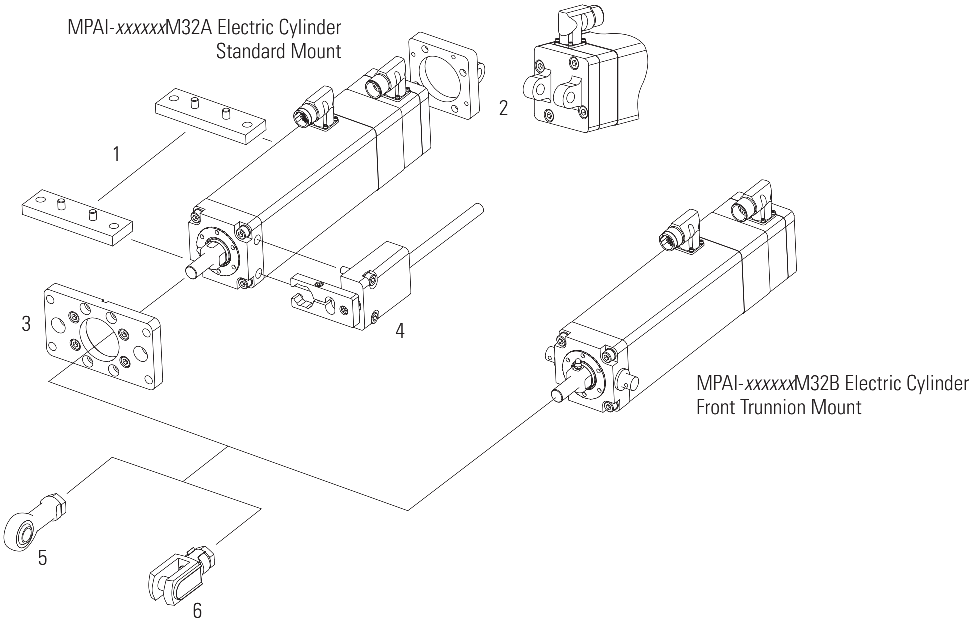
Refer to the MP-Series Heavy Duty Electric Cylinders Installation Instructions, publication MPAI-IN001, for more information on the positive air pressure kit.

MP-Series Heavy Duty Electric Cylinders and Accessories

The Bulletin MPAI electric cylinders are available in standard and front trunnion mount configurations. The standard mount cylinders provide tapped holes for mounting attachments. The front trunnion mount cylinders are compatible only with the rod-end attachments.

Mounting Attachments

Accessory Item		Description	Dimensions
1	Mounting plates	Includes two mounting plates and mounting hardware.	page 196
2	Rear clevis mounting kit	Includes replacement gasket to maintain IP rating, end-cap mounting bolts are reused.	page 196
3	Front flange mount	For front face mounting. Includes mounting hardware.	page 197
4	Anti-rotation option	Needed to prevent piston rod from rotating when piston rod is not connected to the load. Includes mounting hardware.	page 198



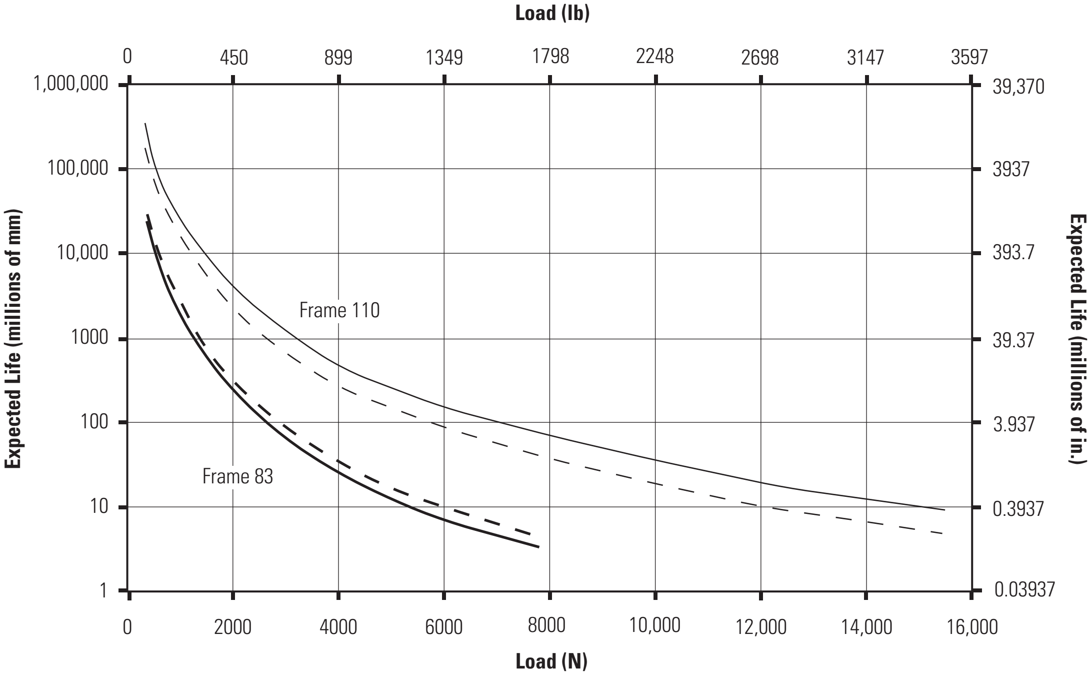
Rod-end Attachments

Accessory Item		Description	Dimensions
5	Rod eye	Includes mounting hardware.	page 199
6	Rod clevis	Permits a swivelling movement of the cylinder in one plane. Includes mounting hardware.	page 199

MP-Series Heavy Duty Electric Cylinders Life Specifications

Electric cylinder life specifications, the total running performance (L), are based on a combination of tested and calculated data. If the parameters of your application are different, your results could be different. The achievable running performance (L) is a function of mean feed force (F), according to DIN 69051-4 as illustrated below. Refer to Motion Analyzer software, version 4.8 or later, for assistance when making these calculations for determining the running performance for your application.

Ball Screw Expected Life Specifications



Load = Mean cubic feed force.
Life = Accumulated total travel running performance.
All curves represent properly lubricated and maintained actuators.

The L_{10} expected life of a ball-screw linear actuator is expressed as the linear travel distance that 90% of properly maintained ball-screws are expected to meet or exceed. This is not a guarantee and this graph should be used only for estimation purposes.

Legend

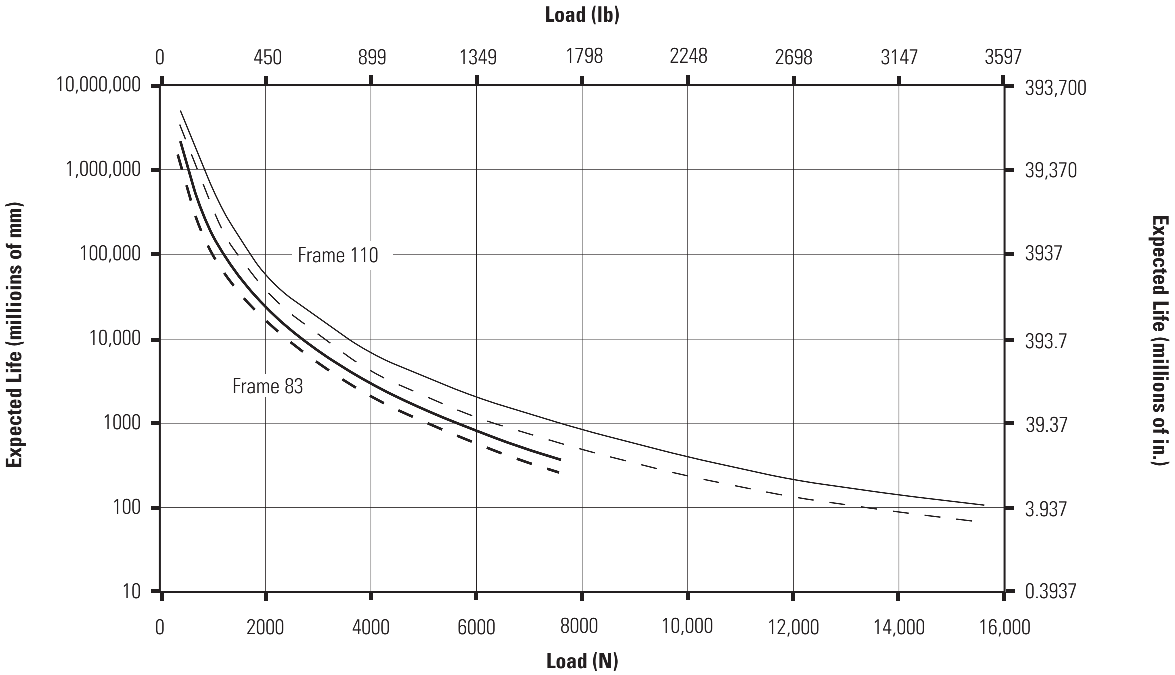
Frame 110	——	10 mm Lead
	– – –	5 mm Lead
Frame 83	**——**	10 mm Lead
	– – –	5 mm Lead

The underlying formula that defines this value is:

$$L_{10} = (P)\left(\frac{C}{F}\right)^3$$

Where:
L_{10} = Travel life in m (in.)
C = Dynamic load rating N (lb)
F = Cubic mean applied load N (lb)
P = Screw lead mm (in.)

Roller Screw Expected Life Specifications



Load = Mean cubic feed force.
Life = Accumulated total travel running performance.
All curves represent properly lubricated and maintained actuators.

The L_{10} expected life of a roller-screw linear actuator is expressed as the linear travel distance that 90% of properly maintained roller-screws are expected to meet or exceed. This is not a guarantee and this graph should be used only for estimation purposes.

Legend

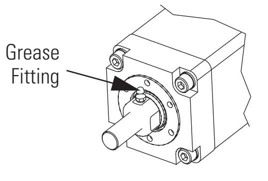
Frame 110 ——— 10 mm Lead
Frame 110 — — — 5 mm Lead
Frame 83 **———** 10 mm Lead
Frame 83 **— — —** 5 mm Lead

The underlying formula that defines this value is:

$$L_{10} = (P)\left(\frac{C}{F}\right)^3$$

Where:
L_{10} = Travel life in millions of mm (in.)
C = Dynamic load rating N (lb)
F = Cubic mean applied load N (lb)
P = Screw lead mm (in.)

Application Guidelines



All Loads must be separately supported and guided (1). Loads should align along the line-of-thrust rod motion. Side loading affects the life of the actuator. Actuators have been lubricated at the factory and are ready for installation. For moderate to light use, no lubrication is required for the life of the actuator. For more severe duty use (higher loads and/or higher duty cycles), periodic lubrication is necessary to achieve actuator expected life (2). Grease should be added via the built-in grease fitting. Removing the cylinder from the machine is seldom necessary (3).

---

(1) Bulletin MPAI actuators are not meant for use where side loading occurs.
(2) Refer to the MP-Series Heavy Duty Electric Cylinders Installation Instructions, publication MPAI-IN001, for lubrication guidelines and specifications.
(3) You can extend or retract the rod to a point where the grease fitting is more accessible.

MP-Series Heavy Duty Electric Cylinders Specifications

General Specifications

Attribute	Frame 83	Frame 110
Construction design	Ball screw or roller screw servo-driven actuator	
Piston rod thread	M16 x 1.50	M20 x 1.50
Working stroke length	152.4 mm (6.0 in.) 304.8 mm (12.0 in.) 457.2 mm (18.0 in.)	
Protection against torsion/guide	Plain bearing guide	
Stroke reserve	0 mm	
Positioning repeatability, max	±0.02 mm (0.0008 in.)	
Reversing backlash, max (1) Ball screw Roller screw	 0.10 mm (0.004 in.) 0.03 mm (0.001 in.)	 0.13 mm (0.005 in.) 0.03 mm (0.001 in.)
Duty cycle	100%	
Position sensing (feedback)	Multi-turn absolute encoder	
Mounting position	Any	
Materials	• Black anodized aluminum body, feedback housing, and end caps • Stainless steel helicoils in tapped holes • Melonited carbon steel thrust rod • Clear zinc carbon steel rod end • Black oxide high strength steel hardware	

(1) In new condition.

Performance Specifications (ball screw)

Electric Cylinder (1) **Cat. No.**	**Frame**	**Max Feed Force** N (lb)	**System Continuous** (2) **Stall Force** N (lb)		**Max Speed** mm/s (in./s)	**Ball Screw Lead** mm/rev (in./rev)	**Stroke Lengths** mm (in.)	**Dynamic Load Rating (1 million revs)** N (lb)
			25 °C (77 °F)	40 °C (104 °F)				
MPAI-*x*3*xxx*CM32*x*	83	4448 (1000)	4003 (900)	3176 (714)	279 (11)	5.0 (0.197)	152.4 mm (6.0 in.) 304.8 mm (12.0 in.) 457.2 mm (18.0 in.) (3)	7602 (1709)
MPAI-*x*3*xxx*EM32*x*		4003 (900)	2002 (450)	1588 (357)	559 (22)	10.0 (0.394)		5400 (1214)
MPAI-*x*4*xxx*CM32*x*	110	8896 (2000)	7784 (1750)	6179 (1389)	279 (11)	5.0 (0.197)		15,100 (3395)
MPAI-*x*4*xxx*EM32*x*		7784 (1750)	3892 (875)	3092 (695)	559 (22)	10.0 (0.394)		15,000 (3372)

(1) MPAI-A (230V) or MPAI-B (460V) replaces *x* in catalog number. Stroke length replaces *xxx* in catalog number.
(2) Characteristics when mounted to 279 x 279 x 12.7 mm (11 x 11 x 0.5 in.) aluminum mounting surface.
(3) For this stroke length, maximum speed is reduced by 33% for the 83 mm frame and by 12% for the 110 mm frame.

Performance Specifications (roller screw)

Electric Cylinder (1) **Cat. No.**	**Frame**	**Max Feed Force** N (lb)	**System Continuous** (2) **Stall Force** N (lb)		**Max Speed** mm/s (in./s)	**Roller Screw Lead** mm/rev	**Stroke Lengths** mm (in.)	**Dynamic Load Rating (1 million revs)** N (lb)
			25 °C (77 °F)	40 °C (104 °F)				
MPAI-*x*3*xxx*RM32*x*	83	7562 (1700)	3781 (850)	3003 (675)	279 (11)	5.0 (0.197)	152.4 mm (6.0 in.) 304.8 mm (12.0 in.) 457.2 mm (18.0 in.) (3)	29,198 (6564)
MPAI-*x*3*xxx*SM32*x* (4)		3781 (850)	1891 (425)	1499 (337)	559 (22)	10.0 (0.394)		26,013 (5848)
MPAI-*x*4*xxx*RM32*x*	110	14,679 (3300)	7340 (1650)	5827 (1310)	279 (11)	5.0 (0.197)		36,831 (8280)
MPAI-*x*4*xxx*SM32*x* (4)		7340 (1650)	3670 (825)	2914 (655)	559 (22)	10.0 (0.394)		34,193 (7687)

(1) MPAI-A (230V) or MPAI-B (460V) replaces *x* in catalog number. Stroke length replaces *xxx* in catalog number.
(2) Characteristics when mounted to 279 x 279 x 12.7 mm (11 x 11 x 0.5 in.) aluminum mounting surface.
(3) For this stroke length, maximum speed is reduced by 37% for the 83 mm frame and by 30% for the 110 mm frame.
(4) Refer to your Rockwell Automation sales representative for availability.

MP-Series Heavy Duty Electric Cylinders System Combinations

For Bulletin MPAI electric cylinders and	**Refer to**
Kinetix 6000 (230V and 460V) drives and Kinetix 6200/Kinetix 6500 (460V) drives	page 674
Kinetix 2000 (230V) drives	page 702
Kinetix 300 (240V and 480V) drives	page 722
Ultra3000 (230V and 460V) drives	page 752

MP-Series Heavy Duty Electric Cylinder Weight Specifications

Standard Mount and Trunnion Mount Electric Cylinders

Electric Cylinder (frame 83) Cat. No.	**Weight, approx.** kg (lb)
MPAI-*x*3150*x*M32*x*	8.3 (18.3)
MPAI-*x*3300*x*M32*x*	10.1 (22.2)
MPAI-*x*3450*x*M32*x*	11.9 (26.2)
MPAI-*x*3150*x*M34*x*	9.3 (20.5)
MPAI-*x*3300*x*M34*x*	11.1 (24.4)
MPAI-*x*3450*x*M34*x*	12.9 (28.4)

Electric Cylinder (frame 110) Cat. No.	**Weight, approx.** kg (lb)
MPAI-*x*4150*x*M32*x*	15.8 (34.8)
MPAI-*x*4300*x*M32*x*	18.8 (41.4)
MPAI-*x*4450*x*M32*x*	21.8 (48.0)
MPAI-*x*4150*x*M34*x*	17.3 (38.1)
MPAI-*x*4300*x*M34*x*	20.3 (44.7)
MPAI-*x*4450*x*M34*x*	23.3 (51.4)

Mounting Accessories

Accessory	Frame	Cat. No.	**Weight, approx.** g (oz)
Mounting plates	83	MPAI-NA306	920 (32.5)
	110	MPAI-NA406	1150 (40.6)
Rear clevis mount	83	MPAI-NA303	110 (3.88)
	110	MPAI-NA403	350 (12.3)

Accessory	Frame	Cat. No.	**Weight, approx.** g (oz)
Front flange mount	83	MPAI-NA301	1070 (37.7)
	110	MPAI-NA401	1740 (61.4)

Rod-end Accessories

Accessory	Frame	Cat. No.	**Weight, approx.** g (oz)
Rod eye	83	MPAI-NE303	255 (8.99)
	110	MPAI-NE403	497 (17.5)

Accessory	Frame	Cat. No.	**Weight, approx.** g (oz)
Rod clevis	83	MPAI-NE304	350 (12.3)
	110	MPAI-NE404	710 (25.0)

Anti-rotation Option Accessories

Accessory	Frame	Cat. No.	**Weight, approx.** g (oz)
Anti-rotation option	83	MPAI-NE30215	870 (30.7)
		MPAI-NE30230	1020 (36.0)
		MPAI-NE30245	1170 (41.3)

Accessory	Frame	Cat. No.	**Weight, approx.** g (oz)
Anti-rotation option	110	MPAI-NE40215	950 (33.5)
		MPAI-NE40230	1110 (39.2)
		MPAI-NE40245	1260 (44.4)

MP-Series Heavy Duty Electric Cylinder Motor Brake Specifications

Electric Cylinder Cat. No.	**Static Torque** N•m (lb•in)	**Coil Current at 24V DC** A_{peak}	**Brake Response Time**		
			Release ms	**Engage (using external arc suppression device)**	
				MOV ms	**Diode** ms
MPAI-*x*3*xxxx*M34*x*	4.0 (35.4)	0.43	50	40	80
MPAI-*x*4*xxxx*M34*x*	10.0 (88.5)	0.67	35	25	50

Estimated Brake Holding Force

Electric Cylinder Cat. No.	**MPAI-*xxxxx*CM34*x*** (1) N (lb)	**MPAI-*xxxxx*EM34*x*** (1) N (lb)	**MPAI-*xxxxx*RM34*x*** (2) N (lb)	**MPAI-*xxxxx*SM34*x*** (2) N (lb)
MPAI-*x*3*xxxx*M34*x*	5494 (1235)	2745 (617)	6423 (1444)	3212 (722)
MPAI-*x*4*xxxx*M34*x*	13,736 (3088)	6868 (1544)	16,063 (3611)	8029 (1805)

(1) MPAI-*xxxxx*CM34*x* is a 5 mm ball-screw actuator and MPAI-*xxxxx*EM34*x* is a 10 mm ball-screw actuator.

(2) MPAI-*xxxxx*RM34*x* is a 5 mm roller-screw actuator and MPAI-*xxxxx*SM34*x* is a 10 mm roller-screw actuator.

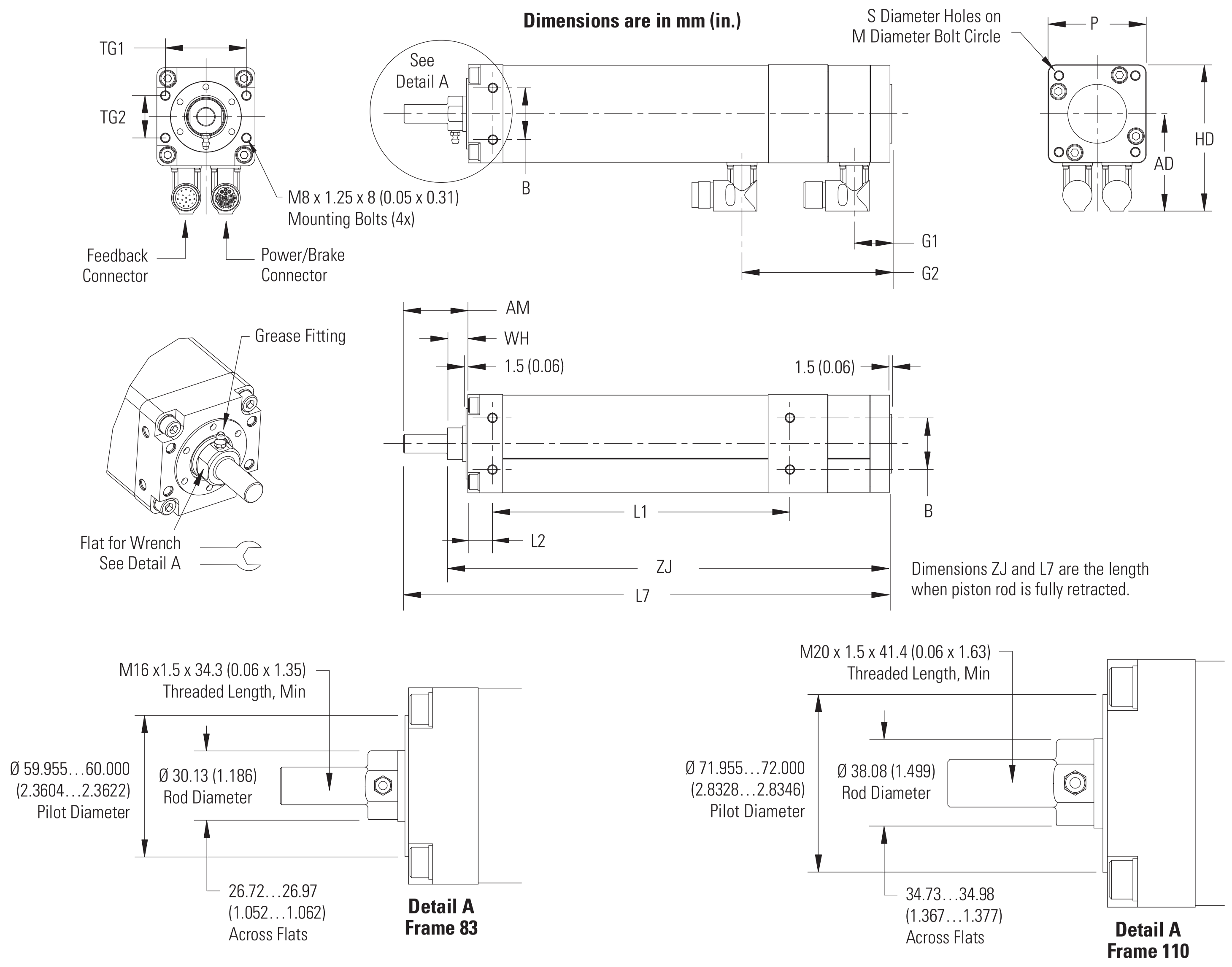
In vertical applications, an unpowered Bulletin MPAI electric cylinder requires a holding brake to maintain position if the load on the actuator exceeds these values.

Back Drive Force

Electric Cylinder Cat. No.	**MPAI-*xxxxx*CM32*x*** N (lb)	**MPAI-*xxxxx*EM32*x*** N (lb)	**MPAI-*xxxxx*RM32*x*** N (lb)	**MPAI-*xxxxx*SM32*x*** N (lb)
MPAI-*x*3*xxxx*M32*x*	267 (60)	126 (28)	592 (133)	525 (118)
MPAI-*x*4*xxxx*M32*x*	445 (100)	196 (44)	734 (165)	623 (140)

MP-Series Electric Cylinder Dimensions

MP-Series Heavy Duty Electric Cylinder Dimensions (frame 83 and 110)



MP-Series Heavy Duty Electric Cylinder Dimensions (frame 83)

Electric Cylinder Cat. No.	**AD** mm (in.)	**AM** mm (in.)	**B** mm (in.)	**G1** mm (in.)	**G2** [(1)] mm (in.)	**HD** mm (in.)	**L1** mm (in.)	**L2** mm (in.)	**L7** [(1)] mm (in.)	**M** [(2)] mm (in.)	**P** mm (in.)	**S** mm (in.)	**TG1** mm (in.)	**TG2** mm (in.)	**WH** mm (in.)	**ZJ** [(1)] mm (in.)
MPAI-A/B3150*x*M32*x*	82.8 (3.26)	54.6 (2.15)	44.0 (1.73)	30.4 (1.20)	127.5 (5.02)	124.6 (4.90)	253.1 (9.97)	21.0 (0.83)	413.6 (16.29)	92.0 (3.622)	83.6 (3.29)	M8 x 1.25 x12 (0.05 x 0.47)	69.0 (2.72)	36.0 (1.42)	17.3 (0.68)	376.3 (14.81)
MPAI-A/B3300*x*M32*x*							405.5 (15.97)		566.0 (22.29)							528.7 (20.81)
MPAI-A/B3450*x*M32*x*							557.9 (21.97)		718.4 (28.29)							681.1 (26.81)

(1) If ordering MPAI-A/B3*xxxx*M34A actuator with brake, add 47.7 mm (1.88 in.) to dimensions G2, L7, and ZJ.

(2) The tolerance for this dimension is +0.0, -0.038 mm (+0.0, -0.0015 in.).

MP-Series Heavy Duty Electric Cylinder Dimensions (frame 110)

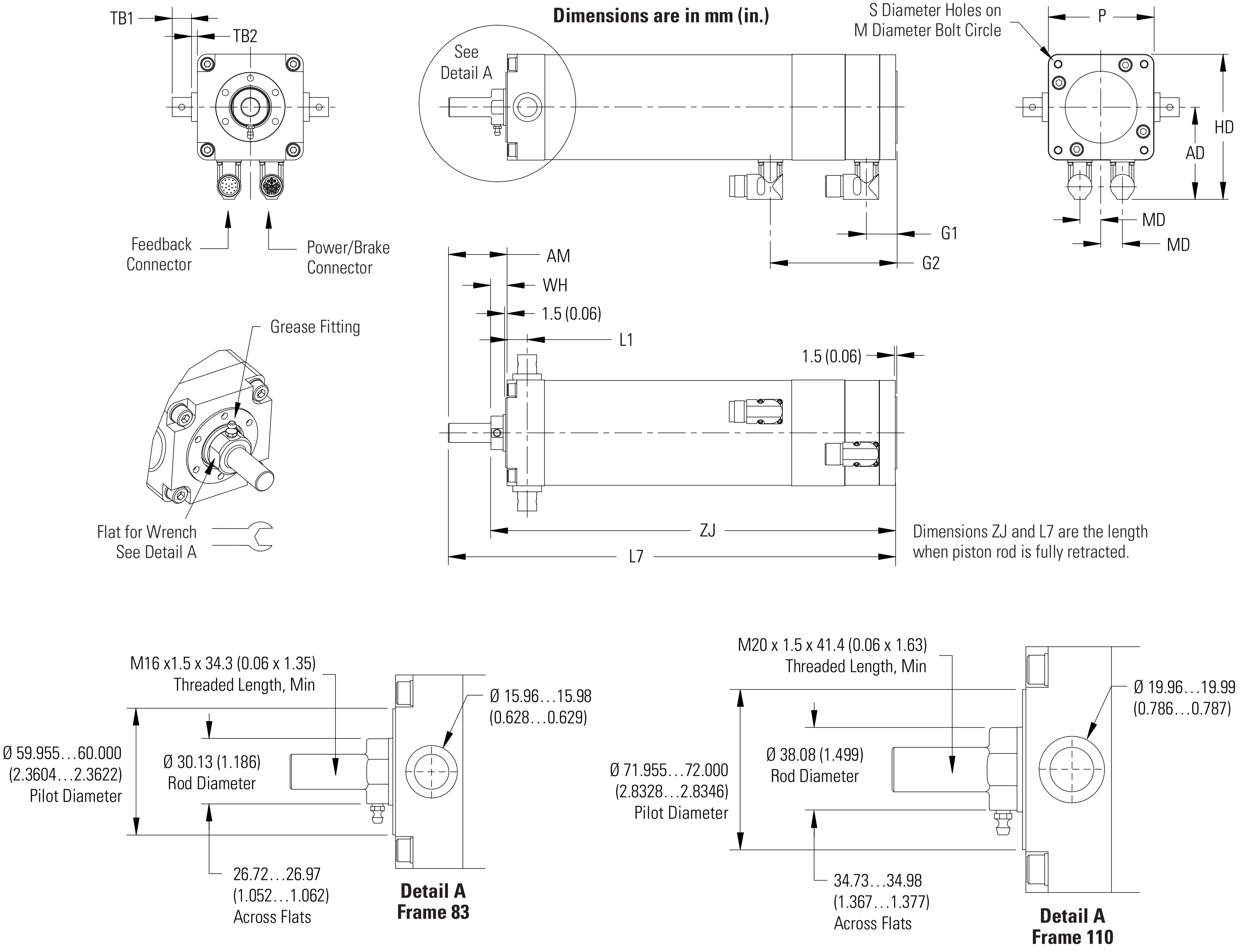
Electric Cylinder Cat. No.	**AD** mm (in.)	**AM** mm (in.)	**B** mm (in.)	**G1** mm (in.)	**G2** [(1)] mm (in.)	**HD** mm (in.)	**L1** mm (in.)	**L2** mm (in.)	**L7** [(1)] mm (in.)	**M** [(2)] mm (in.)	**P** mm (in.)	**S** mm (in.)	**TG1** mm (in.)	**TG2** mm (in.)	**WH** mm (in.)	**ZJ** [(1)] mm (in.)
MPAI-A/B4150*x*M32*x*	96.3 (3.79)	61.3 (2.41)	50.0 (1.97)	30.4 (1.20)	130.7 (5.15)	151.5 (5.96)	295.4 (11.63)	25.0 (0.98)	466.6 (18.37)	127.0 (5.0)	110.5 (4.35)	M8 x 1.25 x12 (0.05 x 0.47)	85.0 (3.35)	55.0 (2.17)	16.8 (0.66)	422.1 (16.62)
MPAI-A/B4300*x*M32*x*							447.8 (17.63)		619.0 (24.37)							574.5 (22.62)
MPAI-A/B4450*x*M32*x*							600.2 (23.63)		771.4 (30.37)							726.9 (28.62)

(1) If ordering MPAI-A/B4*xxxx*M34A actuator with brake, add 47.7 mm (1.88 in.) to dimensions G2, L7, and ZJ.

(2) The tolerance for this dimension is +0.0, -0.038 mm (+0.0, -0.0015 in.).

Actuators are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

MP-Series Heavy Duty Electric Cylinder Trunnion Mount Dimensions (frame 83 and 110)



Dimensions are in mm (in.)

Dimensions ZJ and L7 are the length when piston rod is fully retracted.

MP-Series Heavy Duty Electric Cylinder Trunnion Mount Dimensions (frame 83)

Electric Cylinder Cat. No.	AD mm (in.)	AM mm (in.)	G1 mm (in.)	G2 (1) mm (in.)	HD mm (in.)	L7 (1) mm (in.)	M (2) mm (in.)	MD mm (in.)	P mm (in.)	S mm (in.)	TB1 mm (in.)	TB2 mm (in.)	WH mm (in.)	ZJ (1) mm (in.)
MPAI-A/B3150*x*M32*x*	82.8 (3.26)	54.6 (2.15)	30.4 (1.20)	127.5 (5.02)	124.6 (4.90)	413.6 (16.29)	92.0 (3.622)	16.8 (0.66)	83.6 (3.29)	M8 x 1.25 x12 (0.05 x 0.47)	16.0 (0.63)	3.2 (0.13)	17.3 (0.68)	376.3 (14.81)
MPAI-A/B3300*x*M32*x*						566.0 (22.29)								528.7 (20.81)
MPAI-A/B3450*x*M32*x*						718.4 (28.29)								681.1 (26.81)

(1) If ordering MPAI-A/B3*xxxx*M34B actuator with brake, add 47.7 mm (1.88 in.) to dimensions G2, L7, and ZJ.
(2) The tolerance for this dimension is +0.0, -0.038 mm (+0.0, -0.0015 in.).

MP-Series Heavy Duty Electric Cylinder Trunnion Mount Dimensions (frame 110)

Electric Cylinder Cat. No.	AD mm (in.)	AM mm (in.)	G1 mm (in.)	G2 (1) mm (in.)	HD mm (in.)	L7 (1) mm (in.)	M (2) mm (in.)	MD mm (in.)	P mm (in.)	S mm (in.)	TB1 mm (in.)	TB2 mm (in.)	WH mm (in.)	ZJ (1) mm (in.)
MPAI-A/B4150*x*M32*x*	96.3 (3.79)	61.3 (2.41)	30.4 (1.20)	130.7 (5.15)	151.5 (5.96)	466.6 (18.37)	127.0 (5.0)	22.2 (0.87)	110.5 (4.35)	M8 x 1.25 x12 (0.05 x 0.47)	20.05 (0.79)	6.25 (0.25)	16.8 (0.66)	422.1 (16.62)
MPAI-A/B4300*x*M32*x*						619.0 (24.37)								574.5 (22.62)
MPAI-A/B4450*x*M32*x*						771.4 (30.37)								726.9 (28.62)

(1) If ordering MPAI-A/B4*xxxx*M34B actuator with brake, add 47.7 mm (1.88 in.) to dimensions G2, L7, and ZJ.
(2) The tolerance for this dimension is +0.0, -0.038 mm (+0.0, -0.0015 in.).

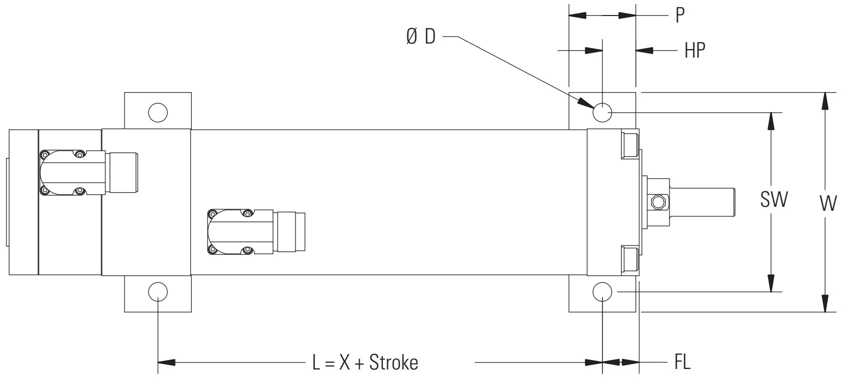
Actuators are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

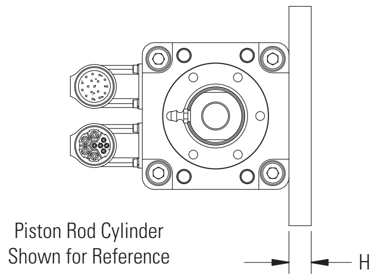
MP-Series Heavy Duty Electric Cylinder Mounting Accessories

These accessories apply to standard-mount electric cylinders. Components are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

Mounting Plates Attachment

Dimensions are in mm (in.)

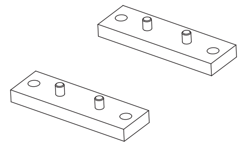




Piston Rod Cylinder Shown for Reference

Cat. No. [1]	Frame	X mm (in.)	FL mm (in.)	D mm (in.)	P mm (in.)	HP mm (in.)	SW mm (in.)	W mm (in.)	H mm (in.)
MPAI-NA306	83	100.7 (3.96)	21.0 (0.83)	10.7 (0.42)	38.1 (1.50)	19.05 (0.75)	102 (4.02)	125 (4.92)	12.7 (0.50)
MPAI-NA406	110	143 (5.63)	25.0 (0.98)				130 (5.12)	155 (6.10)	

(1) Material is carbon steel and finish is melonite. Contains no copper, PTFE, or silicone.

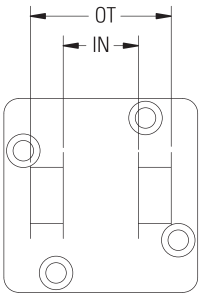


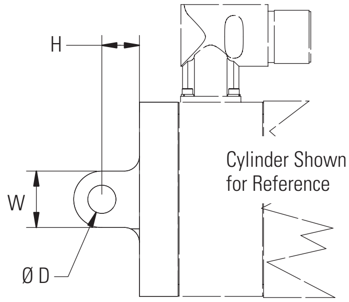
Attachment includes:

- 2 mounting plates
- Mounting hardware

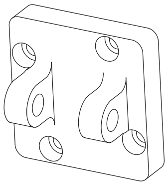
Rear Clevis Mount Attachment

Dimensions are in mm (in.)





Cylinder Shown for Reference

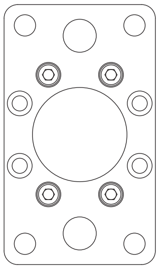


Attachment includes replacement gasket.

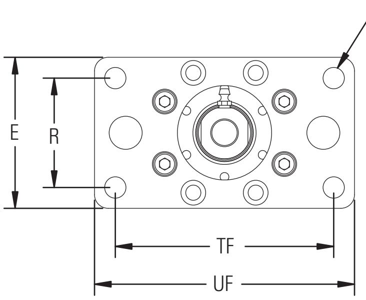
Cat. No. [1]	Frame	D [2] mm (in.)	IN [3] mm (in.)	OT [4] mm (in.)	W mm (in.)	H mm (in.)
MPAI-NA303	83	12.0 (0.47)	32.0 (1.26)	60.0 (2.36)	24.0 (0.94)	16.0 (0.63)
MPAI-NA403	110	16.0 (0.63)	50.0 (1.97)	90.0 (3.54)	36.0 (1.42)	22.0 (0.87)

(1) Material is carbon steel and finish is melonite. Contains no copper, PTFE, or silicone.
(2) Tolerance for this dimension is +0.043, -0 mm (+0.017, -0 in.).
(3) Tolerance for this dimension is +0.62, -0 mm (+0.024, -0 in.).
(4) Tolerance for this dimension is +0, -0.62 mm (+0, -0.024 in.).

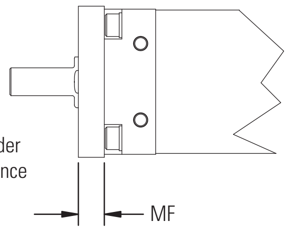
Front Flange Attachment



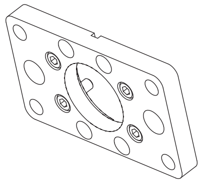
You can rotate the front flange 90°.



Dimensions are in mm (in.)



Piston Rod Cylinder Shown for Reference



Attachment includes mounting hardware.

Cat. No. [1]	Frame	E mm (in.)	FB mm (in.)	MF mm (in.)	R mm (in.)	TF mm (in.)	UF mm (in.)
MPAI-NA301	83	87.0 (3.43)	12.3 (0.48)	15.0 (0.59)	63.0 (2.48)	126 (4.96)	150 (5.90)
MPAI-NA401	110	110.5 (4.35)	14.7 (0.58)	16.0 (0.63)	75.0 (2.95)	150 (5.90)	170 (6.69)

(1) Material is carbon steel and finish is melonite. Contains no copper, PTFE, or silicone.

MP-Series Heavy Duty Electric Cylinder Anti-rotation Accessory

The Bulletin MPAI actuator design lets the extending rod rotate. This provides simple setup of the actuator letting you rotate the rod and thread it in and out of the actuator for mechanical attachment or system testing. This feature also requires that the rod be kept from rotating when used in it's dedicated application for proper linear motion. In most applications, for example, where the load is coupled to linear bearings or some support device, the load cannot rotate and thus provides anti-rotation for the extending actuator rod. For applications where the load is free to rotate, Allen-Bradley offers the anti-rotation option as shown on page 198.

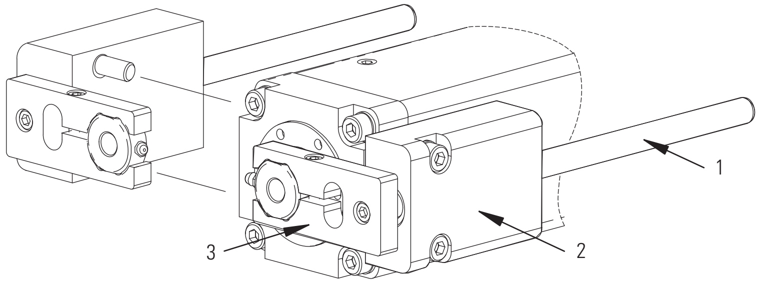
IMPORTANT The anti-rotation option is not a guide or support mechanism. It is intended only as an anti-rotation device.

Anti-rotation Guides for Fixed Strokes

Cat. No. Frame 83	Stroke mm (in.)
MPAI-NE30215	152.4 (6.0)
MPAI-NE30230	304.8 (12.0)
MPAI-NE30245	457.2 (18.0)

Cat. No. Frame 110	Stroke mm (in.)
MPAI-NE40215	152.4 (6.0)
MPAI-NE40230	304.8 (12.0)
MPAI-NE40245	457.2 (18.0)

Anti-rotation Option Material Specifications



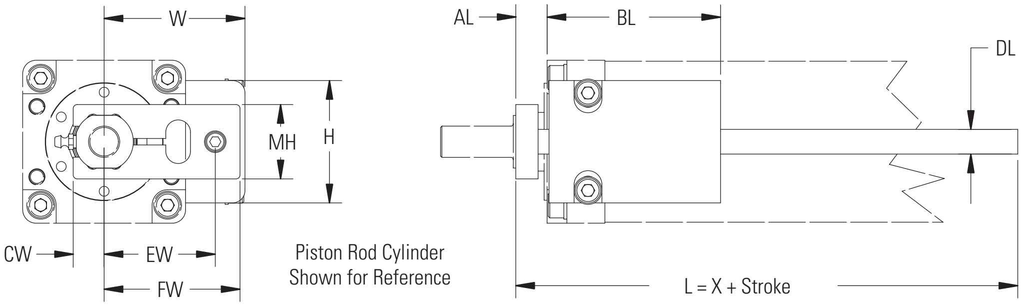
You can mount the anti-rotation option from either side of the standard mount cylinder.

Item	Attribute	Value [1]
1	Rod	C1060 ground hardened steel
2	Bearing block	Anodized aluminium
3	Clamp	Zinc-plated carbon steel

[1] Contains no copper, PTFE, or silicone.

Anti-rotation Option Dimensions

Dimensions are in mm (in.)



Piston Rod Cylinder Shown for Reference

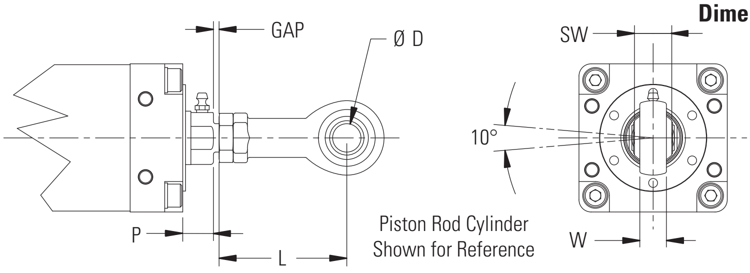
Cat. No.	Frame	X mm (in.)	L mm (in.)	AL mm (in.)	BL mm (in.)	DL mm (in.)	W mm (in.)	CW mm (in.)	EW mm (in.)	FW mm (in.)	H mm (in.)	MH mm (in.)
MPAI-NE30215	83	105.2 (4.14)	257.6 (10.14)	16.1 (0.63)	89.0 (3.50)	12.7 (0.50)	72.3 (2.85)	15.9 (0.63)	57.0 (2.24)	69.7 (2.74)	62.7 (2.47)	38.1 (1.50)
MPAI-NE30230			410.0 (16.14)									
MPAI-NE30245			562.4 (22.14)									
MPAI-NE40215	110	108.5 (4.27)	260.9 (10.27)	15.5 (0.61)	93.0 (3.66)	12.7 (0.50)	85.7 (3.37)	19.8 (0.78)	70.5 (2.78)	83.2 (3.28)	62.7 (2.47)	45.9 (1.81)
MPAI-NE40230			413.3 (16.27)									
MPAI-NE40245			565.7 (22.27)									

MP-Series Heavy Duty Electric Cylinder Rod-end Accessories

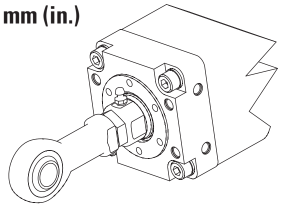
These accessories apply to standard and front trunnion-mount electric cylinders. Components are designed to metric dimensions. Inch dimensions are approximate conversions from millimeters. Dimensions without tolerances are for reference.

Rod-eye Attachment

Dimensions are in mm (in.)



Piston Rod Cylinder Shown for Reference



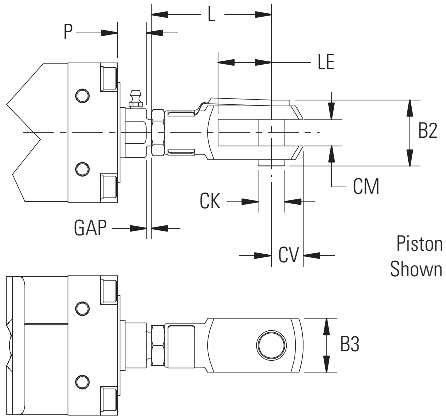
Attachment includes mounting hardware.

Cat. No. [1]	Frame	**D** mm (in.)	**P** mm (in.)	**L** mm (in.)	**W** mm (in.)	**SW** mm (in.)	**GAP** mm (in.)
MPAI-NE303	83	16.0 (0.63)	17.3 (0.68)	72.0 (2.83)	15.0 (0.59)	21.0 (0.83)	3.0…19.0 (0.12…0.75)
MPAI-NE403	110	20.0 (0.79)	16.8 (0.66)	87.0 (3.43)	18.0 (0.71)	25.0 (0.98)	3.0…24.5 (0.12…0.96)

[1] Material is steel and finish is clear zinc.

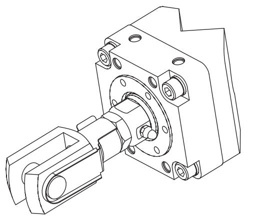
IMPORTANT The spherical rod-eye attachment must be used with an anti-rotation option or the load that the actuator is attached to must be kept from rotating. Without one of these two methods, the spherical rod-eye will let the actuator rod rotate and result in inaccurate positioning due to lost rotational motion.

Rod Clevis Attachment



Piston Rod Cylinder Shown for Reference

Dimensions are in mm (in.)



Attachment includes:

- 1 Rod clevis
- 1 Hinged spring pin
- 1 Hex nut

Cat. No. [1]	Frame	**B2** mm (in.)	**B3** mm (in.)	**CK** mm (in.) H9	**CM** mm (in.)	**CV** mm (in.)	**LE** mm (in.)	**L** mm (in.)	**P** mm (in.)	**GAP** mm (in.)
MPAI-NE304	83	40.0 (1.57)	32.0 (1.26)	16.0 (0.63)	16.0 (0.63)	19.0 (0.75)	32.0 (1.26)	72.0 (2.83)	17.3 (0.68)	3.0…19.0 (0.12…0.75)
MPAI-NE404	110	48.0 (1.89)	40.0 (1.57)	20.0 (0.79)	20.0 (0.79)	25.0 (0.98)	40.0 (1.57)	90.0 (3.54)	16.8 (0.66)	3.0…24.5 (0.12…0.96)

[1] Material is galvanized steel.

MP-Series Heavy Duty Electric Cylinder Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your actuator. For questions regarding product availability, contact your Allen-Bradley distributor.

MP-Series Heavy Duty Electric Cylinder Catalog Numbers

MP AI - *x x xxx x* M3 *x x xxx*

Special Feature (1)
Blank = IP67 rating, standard

Actuator Mounting
A = Front face, side front, bottom front, and rear tapped mounting holes
B = Front trunnion, no tapped mounting holes

Holding Brake
2 = No brake
4 = 24V DC brake

Feedback
M3 = Multi-turn, 1024 sin/cos, absolute encoder, Hiperface protocol

Mechanical Drive/Screw Lead, Motor Type (2)
C = 5.0 mm/rev ball screw (0.197 in./rev)
E = 10.0 mm/rev ball screw (0.394 in./rev)
R = 5.0 mm/rev roller screw (0.197 in./rev)
S = 10.0 mm/rev roller screw (0.394 in./rev)

Rod Stroke Length (2)
150 = 152.4 mm (6.0 in.)
300 = 304.8 mm (12.0 in.)
450 = 457.2 mm (18.0 in.)

Feedback
M3 = Multi-turn, 1024 sin/cos, absolute encoder, Hiperface protocol

Actuator Frame Size
3 = 83 mm (3.27 in.)
4 = 110 mm (4.33 in.)

Voltage Class
A = 200V
B = 400V

Actuator Type
AI = Actuator rod (threaded male rod end)

Actuator Series
MP = MP-Series

(1) The Special Feature field is used for customer-specific coding.

(2) Not all combinations are available. Only the configurations as listed in Performance Specifications (ball screw) and Performance Specifications (roller screw), on page 189, are available.

Refer to your Rockwell Automation sales representative for the availability of MPAI-*xxxxx*RM3 and MPAI-*xxxxx*SM3 (roller screw) electric cylinders.

MP-Series Heavy Duty Electric Cylinder Accessory Catalog Numbers

MP AI - ***xx x xx xx***

Stroke Length (applies to NE*x*02 anti-rotation option only)
15 = 152.4 mm (6.0 in.)
30 = 304.8 mm (12.0 in.)
45 = 457.2 mm (18.0 in.)

Option Number
01 = Front face mounting flange
02 = If NA*x*02 = Mounting flange
    If NE*x*02 = Anti-rotation option
03 = If NA*x*03 = Rear clevis
    If NE*x*03 = Spherical rod-eye
04 = Clevis
06 = Mounting plate kit

Actuator Frame Size
3 = 83 mm (3.27 in.)
4 = 110 mm (4.33 in.)

Accessory Type
NA = Mounting accessory
NE = Rod-end accessory

Actuator Type
AI = Actuator rod (threaded male rod end)

Actuator Series
MP = MP-Series

Notes:

LDC-Series Iron Core Linear Servo Motors

The LDC-Series iron core linear motors address a growing interest in linear motor technology as it becomes more affordable and is increasingly recognized as a practical means of improving machine performance. With the iron core product design, you now have cost-effective options to help you improve machine throughput while reducing maintenance and downtime.

For drive compatibility, refer to Servo Drives on page 14.

LDC-Series Iron Core Linear Motor Features

- High thrust force to cost ratio lets you use smaller, less expensive motors.
- Cogging torque less than 5% of continuous force.
- Very high acceleration and speeds up to 10 m/s (32.8 ft/s) greatly increase the throughput of your machine.
- No limits to travel distance. Ability to achieve high speeds over short and long travels.
- Direct drive technology for extreme servo responsiveness.
- No-wear, high reliability parts increase productivity.
- Peak forces to 5246 N (1179 lb).
- Ability to size and optimize LDC-Series linear motors and corresponding servo drives by using Motion Analyzer software reduces product selection time and minimizes cost.
- Full set-up and programming support through RSLogix 5000 software reduces set-up time.

LDC-Series Iron Core Linear Motor Accessories

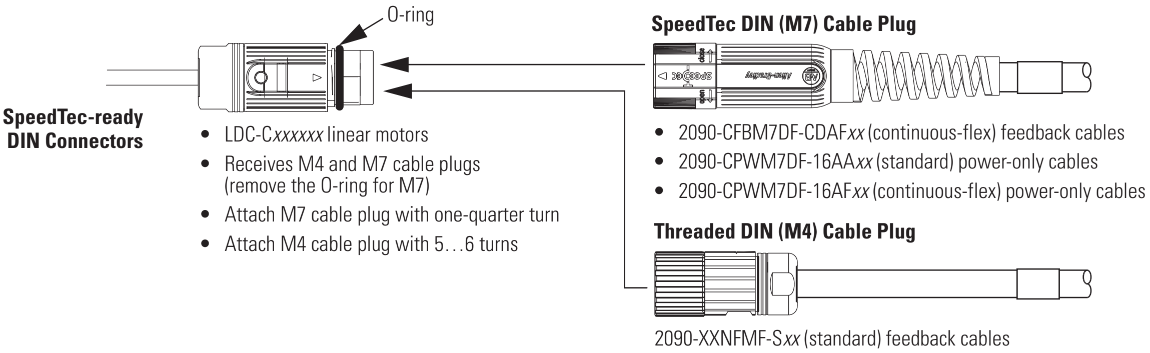
- Cooling plates.
- Bulk head connector kit.
- Encoder connector kit.
- Hall effect replacement module for connectorized coil.
- Hall effect replacement module for flying-lead coil.

Accessories for LDC-Series Iron Core Linear Motors

Cat. No.	Accessory	Description
LDC-BULK-HD	Bulk head connector kit	For easy mounting of flex cable to non-flex cables. Kit includes flange for feedback and power connectors, o-rings, and nut. Refer to Bulk Head Connector Flange Dimensions on page 225.
LDC-ENC-CNCT	Encoder connector kit	Adapts your encoder to the feedback cable on the Hall effect module.
LDC-HALL-C	Hall effect module	Replacement module for use with connectorized coil.
LDC-HALL-F		Replacement module for use with flying-lead coil.

Motor Connector/Cable Compatibility

LDC-Series Iron Core Linear Motors



SpeedTec-ready DIN Connectors

- LDC-C*xxxxxx* linear motors
- Receives M4 and M7 cable plugs (remove the O-ring for M7)
- Attach M7 cable plug with one-quarter turn
- Attach M4 cable plug with 5…6 turns

SpeedTec DIN (M7) Cable Plug

- 2090-CFBM7DF-CDAF*xx* (continuous-flex) feedback cables
- 2090-CPWM7DF-16AA*xx* (standard) power-only cables
- 2090-CPWM7DF-16AF*xx* (continuous-flex) power-only cables

Threaded DIN (M4) Cable Plug

2090-XXNFMF-S*xx* (standard) feedback cables

Cooling Plate Accessories for LDC-Series Iron Core Linear Motors

Cat. No. Coil	Cat. No. Cooling Plate
LDC-C030100-*xxxxx*	LDC-030-100-CP
LDC-C030200-*xxxxx*	LDC-030-200-CP
LDC-C050100-*xxxxx*	LCC-050-100-CP
LDC-C050200-*xxxxx*	LDC-050-200-CP
LDC-C050300-*xxxxx*	LCC-050-300-CP
LDC-C075200-*xxxxx*	LDC-075-200-CP
LDC-C075300-*xxxxx*	LDC-075-300-CP
LDC-C075400-*xxxxx*	LDC-075-400-CP
LDC-C100300-*xxxxx*	LDC-100-300-CP
LDC-C100400-*xxxxx*	LDC-100-400-CP
LDC-C100600-*xxxxx*	LDC-100-600-CP
LDC-C150400-*xxxxx*	LDC-150-400-CP
LDC-C150600-*xxxxx*	LDC-150-600-CP

LDC-Series Iron Core Linear Motor Performance Specifications

These performance specifications apply to all LDC-Series iron core linear motors.

Common Performance Specifications

Attribute	Value
Motor type	3 phase, wye winding, synchronous permanent magnet stator, non-ventilated linear motor
Operating speed, max	10 m/s (32.8 ft/s)
Operating voltage, (not for direct connection to AC line)	460V AC rms
Dielectric rating of motor power connections (U,V,W), to ground for 1.0 s (1)	2500V AC rms 50/60 Hz
Cogging torque	< 5% of the continuous force
Applied bus voltage, max (2)	650V DC
Electrical cycle length	50 mm (1.9685 in.)
Coil temperature, max	130 °C (266 °F)
Insulation class	130 °C (266 °F) Class B
Thermal time constant, Ref, winding to ambient	45 min
Paint color	Black

(1) Tested during manufacturing process. Do not re-apply test voltage. Contact Application Engineering (631.344.6600) for advice on testing coils post production.

(2) Maximum cable length is 10 m (32.8 ft). Consult Application Engineering (631.344.6600) for applications requiring longer cables.

Motor performance specifications are with sinusoidal commutation. Cooling options include NC (no cooling), AC (air cooling), and WC (water cooling).

LDC-Series Iron Core Linear Motors (30 mm frame size)

Attribute	**Units**	**Symbol**	**LDC-C030100-D*x*T*xx***			**LDC-C030200-D*x*T*xx***			**LDC-C030200-E*x*T*xx***		
			NC	**AC**	**WC**	**NC**	**AC**	**WC**	**NC**	**AC**	**WC**
Force, continuous (1) (2) (3)	N (lbf)	F_c	74 (17)	93 (21)	111 (25)	148 (33)	185 (42)	222 (50)	148 (33)	185 (42)	222 (50)
Force, peak (4)	N (lbf)	F_p	188 (42)			375 (84)					
Thermal resistance	°C/W	R_{th}	2.24	1.43	1.00	1.12	0.72	0.50	1.12	0.72	0.50
Force constant (5) (6) (7)	N/A_{pk} (lbf/A_{pk})	K_f	18.2 (4.1)			18.2 (4.1)			36.4 (8.2)		
Back EMF constant p-p (5) (6) (7)	V_p/m/s (V_p/in/s)	K_e	21.5 (0.55)			21.5 (0.55)			43.0 (1.09)		
Current, peak (4) (6)	A_{pk} (A_{rms})	I_p	12.1 (8.6)			24.3 (17.1)			12.1 (8.6)		
Current, continuous (1) (2) (3) (6)	A_{pk} (A_{rms})	I_c	4.1 (2.88)	5.1 (3.6)	6.1 (4.3)	8.1 (5.8)	10.2 (7.2)	12.2 (8.6)	4.1 (2.9)	5.1 (3.6)	6.1 (4.3)
Resistance p-p @ 20 °C (68 °F) (5) (7)	Ohms	R_{20}	2.256			1.128			4.51		
Inductance p-p (5) (7)	mH	L	21.6			10.8			43.0		
Magnetic attraction (8)	N (lbf)	F_a	393 (88)			786 (177)					

(1) Coils at maximum temperature, 130 °C (266 °F), mounted to an aluminium heat sink whose area is noted in table on page 212, and at 40 °C (104 °F) ambient.
(2) Continuous force and current based on coil moving with all phases sharing the same load in sinusoidal commutation.
(3) For standstill conditions, multiply continuous force and continuous current by 0.9.
(4) Calculated at 20% duty cycle for 1.0 second max. Some applications may produce significantly higher peak forces. Call Applications Engineering (631.344.6600) for details.
(5) Winding parameters listed are measured line-to-line (phase-to-phase).
(6) Currents and voltages listed are measured 0-peak of the sine wave unless noted as rms.
(7) Specifications are ±10%. Phase-to-phase inductance is ±30%.
(8) All specifications are at the standard reference air gap as shown in the drawing on page 214 and page 216.

LDC-Series Iron Core Linear Motors (50 mm frame size)

Attribute	Units	Symbol	LDC-C050100-D*x*T*xx*			LDC-C050200-D*x*T*xx*			LDC-C050200-E*x*T*xx*			LDC-C050300-D*x*T*xx*			LDC-C050300-E*x*T*xx*		
			NC	**AC**	**WC**	**NC**	**AC**	**WC**	**NC**	**AC**	**WC**	**NC**	**AC**	**WC**	**NC**	**AC**	**WC**
Force, continuous (1) (2) (3)	N (lbf)	F_c	119 (27)	149 (34)	179 (40)	240 (54)	299 (67)	359 (81)	240 (54)	299 (67)	359 (81)	363 (82)	453 (102)	544 (122)	363 (82)	453 (102)	544 (122)
Force, peak (4)	N (lbf)	F_p	302 (68)			600 (135)						941 (212)					
Thermal resistance	°C/W	R_{th}	1.44	0.92	0.64	0.71	0.46	0.32	0.71	0.46	0.32	0.48	0.31	0.21	0.48	0.31	0.21
Force constant (5) (6) (7)	N/A_{pk} (lbf/A_{pk})	K_f	30.3 (6.8)			30.3 (6.8)			60.7 (13.6)			30.8 (6.9)			92.4 (20.8)		
Back EMF constant p-p (5) (6) (7)	V_p/m/s (V_p/in/s)	K_e	35.8 (0.91)			35.8 (0.91)			71.7 (1.82)			36.4 (0.92)			109.1 (2.77)		
Current, peak (4) (6)	A_{pk} (A_{rms})	I_p	11.7 (8.3)			23.3 (16.5)			11.6 (8.2)			35.9 (25.4)			12.0 (8.5)		
Current, continuous (1) (2) (3) (6)	A_{pk} (A_{rms})	I_c	3.9 (2.8)	4.9 (3.5)	5.9 (4.2)	7.9 (5.6)	9.9 (7.0)	11.8 (8.4)	3.9 (2.8)	4.9 (3.5)	5.9 (4.2)	11.8 (8.3)	14.7 (10.4)	17.7 (12.5)	3.9 (2.8)	4.9 (3.5)	5.9 (4.2)
Resistance p-p @ 20 °C (68 °F) (5) (7)	Ohms	R_{20}	3.76			1.88			7.52			1.25			11.28		
Inductance p-p (5) (7)	mH	L	36			18			72			12			108		
Magnetic attraction (8)	N (lbf)	F_a	690 (155)			1379 (310)						2069 (465)					

(1) Coils at maximum temperature, 130 °C (266 °F), mounted to an aluminium heat sink whose area is noted in table on page 212, and at 40 °C (104 °F) ambient.
(2) Continuous force and current based on coil moving with all phases sharing the same load in sinusoidal commutation.
(3) For standstill conditions, multiply continuous force and continuous current by 0.9.
(4) Calculated at 20% duty cycle for 1.0 second max. Some applications may produce significantly higher peak forces. Call Applications Engineering (631.344.6600) for details.
(5) Winding parameters listed are measured line-to-line (phase-to-phase).
(6) Currents and voltages listed are measured 0-peak of the sine wave unless noted as rms.
(7) Specifications are ±10%. Phase-to-phase inductance is ±30%.
(8) All specifications are at the standard reference air gap as shown in the drawing on page 214 and page 216.

LDC-Series Iron Core Linear Motors (75 mm frame size)

Attribute	Units	Symbol	LDC-C075200-DxTxx			LDC-C075200-ExTxx			LDC-C075300-DxTxx			LDC-C075300-ExTxx			LDC-C075400-DxTxx			LDC-C075400-ExTxx		
			NC	**AC**	**WC**	**NC**	**AC**	**WC**	**NC**	**AC**	**WC**	**NC**	**AC**	**WC**	**NC**	**AC**	**WC**	**NC**	**AC**	**WC**
Force, continuous (1) (2) (3)	N (lbf)	F_c	348 (78)	435 (98)	523 (117)	348 (78)	435 (98)	523 (117)	523 (117)	653 (147)	784 (176)	523 (117)	653 (147)	784 (176)	697 (157)	871 (196)	1045 (235)	697 (157)	871 (196)	1045 (235)
Force, peak (4)	N (lbf)	F_p	882 (198)						1368 (308)						1824 (410)					
Thermal resistance	°C/W	R_{th}	0.58	0.37	0.26	0.58	0.37	0.26	0.39	0.25	0.17	0.39	0.25	0.17	0.29	0.19	0.13	0.29	0.19	0.13
Force constant (5) (6) (7)	N/A_{pk} (lbf/A_{pk})	K_f	45.5 (10.2)			91.0 (20.5)			45.5 (10.2)			136.5 (30.7)			45.5 (10.2)			91.0 (20.5)		
Back EMF constant p-p (5) (6) (7)	V_p/m/s (V_p/in/s)	K_e	53.7 (1.37)			107.5 (2.73)			53.7 (1.37)			161.2 (4.10)			53.7 (1.37)			107.5 (2.73)		
Current, peak (4) (6)	A_{pk} (A_{rms})	I_p	22.9 (16.2)			11.5 (8.1)			35.6 (25.1)			11.9 (8.4)			47.4 (33.5)			23.7 (16.8)		
Current, continuous (1) (2) (3) (6)	A_{pk} (A_{rms})	I_c	7.7 (5.4)	9.6 (6.8)	11.5 (8.1)	3.8 (2.7)	4.8 (3.4)	5.7 (4.1)	11.5 (8.1)	14.4 (10.2)	17.2 (12.2)	3.8 (2.7)	4.8 (3.4)	5.7 (4.1)	15.3 (10.8)	19.1 (13.5)	23.0 (16.2)	7.7 (5.4)	9.6 (6.8)	11.5 (8.1)
Resistance p-p @ 20 °C (68 °F) (5) (7)	Ohms	R_{20}	2.47			9.88			1.65			14.82			1.24			4.94		
Inductance p-p (5) (7)	mH	L	24			95			16			142			12			47		
Magnetic attraction (8)	N (lbf)	F_a	2000 (450)						2999 (674)						3999 (899)					

(1) Coils at maximum temperature, 130 °C (266 °F), mounted to an aluminium heat sink whose area is noted in table on page 212, and at 40 °C (104 °F) ambient.
(2) Continuous force and current based on coil moving with all phases sharing the same load in sinusoidal commutation.
(3) For standstill conditions, multiply continuous force and continuous current by 0.9.
(4) Calculated at 20% duty cycle for 1.0 second max. Some applications may produce significantly higher peak forces. Call Applications Engineering (631.344.6600) for details.
(5) Winding parameters listed are measured line-to-line (phase-to-phase).
(6) Currents and voltages listed are measured 0-peak of the sine wave unless noted as rms.
(7) Specifications are ±10%. Phase-to-phase inductance is ±30%.
(8) All specifications are at the standard reference air gap as shown in the drawing on page 214 and page 216.

LDC-Series Iron Core Linear Motors (100 mm frame size)

Attribute	Units	Symbol	LDC-C100300-D*x*T*xx*			LDC-C100300-E*x*T*xx*			LDC-C100400-D*x*T*xx*			LDC-C100400-E*x*T*xx*			LDC-C100600-D*x*T*xx*			LDC-C100600-E*x*T*xx*		
			NC	**AC**	**WC**	**NC**	**AC**	**WC**	**NC**	**AC**	**WC**	**NC**	**AC**	**WC**	**NC**	**AC**	**WC**	**NC**	**AC**	**WC**
Force, continuous (1) (2) (3)	N (lbf)	F_c	674 (152)	843 (190)	1012 (227)	674 (152)	843 (190)	1012 (227)	899 (202)	1124 (253)	1349 (303)	899 (202)	1124 (253)	1349 (303)	1349 (303)	1686 (379)	2023 (455)	1349 (303)	1686 (379)	2023 (455)
Force, peak (4)	N (lbf)	F_p	1767 (397)						2356 (530)						3534 (794)					
Thermal resistance	°C/W	R_{th}	0.33	0.21	0.15	0.33	0.21	0.15	0.25	0.16	0.11	0.25	0.16	0.11	0.17	0.11	0.07	0.17	0.11	0.07
Force constant (5) (6) (7)	N/A_{pk} (lbf/A_{pk})	K_f	60.7 (13.6)			182.0 (40.9)			60.7 (13.6)			121.3 (27.3)			60.7 (13.6)			121.3 (27.3)		
Back EMF constant p-p (5) (6) (7)	$V_p/m/s$ $(V_p/in/s)$	K_e	71.7 (1.82)			215.0 (5.46)			71.7 (1.82)			143.3 (3.64)			71.7 (1.82)			143.3 (3.64)		
Current, peak (4) (6)	A_{pk} (A_{rms})	I_p	34.3 (24.2)			11.4 (8.1)			45.7 (32.3)			22.8 (16.1)			68.5 (48.4)			34.3 (24.2)		
Current, continuous (1) (2) (3) (6)	A_{pk} (A_{rms})	I_c	11.1 (7.9)	13.9 (9.8)	16.7 (11.8)	3.7 (2.6)	4.6 (3.3)	5.6 (3.9)	14.8 (10.5)	18.5 (13.1)	22.2 (15.7)	7.4 (5.2)	9.3 (6.5)	11.1 (7.9)	22.2 (15.7)	27.8 (19.7)	33.3 (23.6)	11.1 (7.9)	13.9 (9.8)	16.7 (11.8)
Resistance p-p @ 20 °C (68 °F) (5) (7)	Ohms	R_{20}	2.04			18.36			1.53			6.12			1.02			4.08		
Inductance p-p (5) (7)	mH	L	20			184			15			61			10			41		
Magnetic attraction (8)	N (lbf)	F_a	3930 (883)						5240 (1178)						7860 (1767)					

(1) Coils at maximum temperature, 130 °C (266 °F), mounted to an aluminium heat sink whose area is noted in table on page 212, and at 40 °C (104 °F) ambient.
(2) Continuous force and current based on coil moving with all phases sharing the same load in sinusoidal commutation.

(3) For standstill conditions, multiply continuous force and continuous current by 0.9.
(4) Calculated at 20% duty cycle for 1.0 second max. Some applications may produce significantly higher peak forces. Call Applications Engineering (631.344.6600) for details.
(5) Winding parameters listed are measured line-to-line (phase-to-phase).
(6) Currents and voltages listed are measured 0-peak of the sine wave unless noted as rms.
(7) Specifications are ±10%. Phase-to-phase inductance is ±30%.
(8) All specifications are at the standard reference air gap as shown in the drawing on page 214 and page 216.

LDC-Series Iron Core Linear Motors (150 mm frame size)

Attribute	Units	Symbol	LDC-C150400-D*x*T*xx*			LDC-C150400-E*x*T*xx*			LDC-C150600-D*x*T*xx*			LDC-C150600-E*x*T*xx*		
			NC	**AC**	**WC**	**NC**	**AC**	**WC**	**NC**	**AC**	**WC**	**NC**	**AC**	**WC**
Force, continuous (1) (2) (3)	N (lbf)	F_c	1281 (288)	1601 (360)	1922 (432)	1281 (288)	1601 (360)	1922 (432)	1922 (432)	2402 (540)	2882 (648)	1922 (432)	2402 (540)	2882 (648)
Force, peak (4)	N (lbf)	F_p	3498 (786)						5246 (1179)					
Thermal resistance	°C/W	R_{th}	0.20	0.13	0.09	0.20	0.13	0.09	0.13	0.09	0.06	0.13	0.09	0.06
Force constant (5) (6) (7)	N/A_{pk} (lbf/A_{pk})	K_f	91.0 (20.5)			182.0 (40.9)			91.0 (20.5)			182.0 (40.9)		
Back EMF constant p-p (5) (6) (7)	$V_p/m/s$ $(V_p/in/s)$	K_e	107.5 (2.73)			215.0 (5.46)			107.5 (2.73)			215.0 (5.46)		
Current, peak (4) (6)	A_{pk} (A_{rms})	I_p	45.2 (32.0)			22.6 (16.0)			67.8 (47.9)			33.9 (24.0)		
Current, continuous (1) (2) (3) (6)	A_{pk} (A_{rms})	I_c	14.1 (10.0)	17.6 (12.4)	21.1 (14.9)	7.0 (5.0)	8.8 (6.2)	10.6 (7.5)	21.1 (14.9)	26.4 (18.7)	31.7 (22.4)	10.6 (7.5)	13.2 (9.3)	15.8 (11.2)
Resistance p-p @ 20 °C (68 °F) (5) (7)	Ohms	R_{20}	2.12			8.48			1.41			5.65		
Inductance p-p (5) (7)	mH	L	22			86			14			58		
Magnetic attraction (8)	N (lbf)	F_a	7860 (1768)						11790 (2652)					

(1) Coils at maximum temperature, 130 °C (266 °F), mounted to an aluminium heat sink whose area is noted in table on page 212, and at 40 °C (104 °F) ambient.
(2) Continuous force and current based on coil moving with all phases sharing the same load in sinusoidal commutation.
(3) For standstill conditions, multiply continuous force and continuous current by 0.9.
(4) Calculated at 20% duty cycle for 1.0 second max. Some applications may produce significantly higher peak forces. Call Applications Engineering (631.344.6600) for details.
(5) Winding parameters listed are measured line-to-line (phase-to-phase).
(6) Currents and voltages listed are measured 0-peak of the sine wave unless noted as rms.
(7) Specifications are ±10%. Phase-to-phase inductance is ±30%.
(8) All specifications are at the standard reference air gap as shown in the drawing on page 214 and page 216.

LDC-Series Linear Motors System Combinations

For LDC-Series linear motors and	Refer to
Kinetix 6000 (460V) drives	page 681
Kinetix 6000 (460V) drives and Kinetix 6200/Kinetix 6500 (460V) drives	page 685
Kinetix 2000 (230V drives)	page 708

For LDC-Series linear motors and	Refer to
Kinetix 3 (240V drives)	page 734
Ultra3000 (230V drives	page 756
Ultra3000 (460V) drives	page 760

LDC-Series Iron Core Linear Motor General Specifications

Weight Specifications - Motor Coil with Flying Leads and Cooling Plate

Cat. No. Coil	**Weight, approx.** kg (lb)	**Cat. No. Cooling Plate**	**Weight, approx.** kg (lb)	**Coil and Cooling Plate (combined) Weight, approx.** kg (lb)
LDC-C030100-DHT20	1.41 (3.1)	LDC-030-100-CP	0.12 (0.26)	1.53 (3.4)
LDC-C030200-*x*HT20	2.27 (5.0)	LDC-030-200-CP	0.20 (0.44)	2.47 (5.4)
LDC-C050100-DHT20	2.05 (4.5)	LDC-050-100-CP	0.15 (0.32)	2.19 (4.8)
LDC-C050200-*x*HT20	3.18 (7.0)	LDC-050-200-CP	0.25 (0.55)	3.43 (7.6)
LDC-C050300-*x*HT20	4.55 (10.0)	LDC-050-300-CP	0.36 (0.79)	4.91 (10.8)
LDC-C075200-*x*HT20	4.55 (10.0)	LDC-075-200-CP	0.39 (0.85)	4.93 (10.9)
LDC-C075300-*x*HT20	6.36 (14.0)	LDC-075-300-CP	0.56 (1.23)	6.92 (15.2)
LDC-C075400-*x*HT20	8.18 (18.0)	LDC-075-400-CP	0.73 (1.60)	8.91 (19.6)
LDC-C100300-*x*HT20	7.73 (17.0)	LDC-100-300-CP	0.73 (1.60)	8.46 (18.6)
LDC-C100400-*x*HT20	10.0 (22.0)	LDC-100-400-CP	0.96 (2.10)	10.96 (24.1)
LDC-C100600-*x*HT20	15.45 (34.0)	LDC-100-600-CP	1.39 (3.05)	16.84 (37.0)
LDC-C150400-*x*HT20	14.55 (32.0)	LDC-150-400-CP	1.93 (4.24)	16.47 (36.2)
LDC-C150600-*x*HT20	21.36 (47.0)	LDC-150-600-CP	2.86 (6.29)	24.22 (53.3)

Weight Specifications - Motor Coil with Connectors and Cooling Plate

Cat. No. Coil	**Weight, approx.** kg (lb)	**Cat. No. Cooling Plate**	**Weight, approx.** kg (lb)	**Coil and Cooling Plate (combined) Weight, approx.** kg (lb)
LDC-C030100-DHT11	1.61 (3.55)	LDC-030-100-CP	0.12 (0.26)	1.73 (3.81)
LDC-C030200-*x*HT11	2.47 (5.44)	LDC-030-200-CP	0.20 (0.44)	2.67 (5.89)
LDC-C050100-DHT11	2.25 (4.96)	LDC-050-100-CP	0.15 (0.32)	2.40 (5.29)
LDC-C050200-*x*HT11	3.38 (7.45)	LDC-050-200-CP	0.25 (0.55)	3.63 (8.00)
LDC-C050300-*x*HT11	4.75 (10.47)	LDC-050-300-CP	0.36 (0.79)	5.11 (11.3)
LDC-C075200-*x*HT11	4.75 (10.47)	LDC-075-200-CP	0.39 (0.85)	5.14 (11.33)
LDC-C075300-*x*HT11	6.56 (14.46)	LDC-075-300-CP	0.56 (1.23)	7.12 (15.70)
LDC-C075400-*x*HT11	8.38 (18.47)	LDC-075-400-CP	0.73 (1.60)	9.11 (20.08)
LDC-C100300-*x*HT11	7.91 (17.4)	LDC-100-300-CP	0.73 (1.60)	8.64 (18.6)
LDC-C100400-*x*HT11	10.2 (22.5)	LDC-100-400-CP	0.96 (2.10)	11.16 (24.60)
LDC-C100600-*x*HT11	15.65 (34.5)	LDC-100-600-CP	1.39 (3.05)	17.04 (37.57)
LDC-C150400-*x*HT11	14.75 (32.5)	LDC-150-400-CP	1.93 (4.24)	16.19 (35.69)
LDC-C150600-*x*HT11	21.56 (47.5)	LDC-150-600-CP	2.86 (6.29)	23.69 (52.23)

Weight Specifications - Motor Magnet Track

Cat. No. **Magnet Track**	**Weight, approx.** kg (lb)
LDC-M030100	0.47 (1.02)
LDC-M050100	0.66 (1.46)
LDC-M075100	0.90 (1.98)
LDC-M100100	1.14 (2.51)
LDC-M150100	1.62 (3.57)
LDC-M030500	2.35 (5.12)
LDC-M050500	3.32 (7.28)
LDC-M075500	4.5 (9.92)
LDC-M100500	5.7 (12.57)
LDC-M150500	8.08 (17.81)

Carriage Weight and Heat Sink Area Requirements

Cat. No.	**Required Heat Sink Area** cm^2 (in.^{2})	**Required Carriage Plate Weight, approx.** kg (lb)
LDC-C030100-DHT*xx*	150 X 200 (6 X 8)	1.1 (2.6)
LDC-C030200-*x*HT*xx*	150 X 300 (6 X 12)	1.6 (3.6)
LDC-C050100-DHT*xx*	200 X 200 (8 X 8)	1.8 (4)
LDC-C050200-*x*HT*xx*	200 X 300 (8 X 12)	2.7 (6)
LDC-C050300-*x*HT*xx*	200 X 400 (8 X 16)	3.6 (8)
LDC-C075200-*x*HT*xx*	250 X 300 (10 X 12)	5.4 (12)
LDC-C075300-*x*HT*xx*	250 X 400 (10 X 16)	7.3 (16)
LDC-C075400-*x*HT*xx*	250 X 500 (10 X 20)	9.1 (20)
LDC-C100300-*x*HT*xx*	300 X 400 (12 X 16)	8.7 (19.2)
LDC-C100400-*x*HT*xx*	300 X 500 (12 X 20)	10.9 (24)
LDC-C100600-*x*HT*xx*	300 X 750 (12 X 30)	19.6 (43.2)
LDC-C150400-*x*HT*xx*	400 X 500 (16 X 20)	21.8 (48)
LDC-C150600-*x*HT*xx*	400 X 750 (16 X 30)	32.7 (72)

Cooling Plate Flow Rate Specifications

Cat. No. **Coil**	**Cat. No.** **Cooling Plate**	**Air Flow Rate** (1) L/min (ft^3/hr)	**Water Flow Rate** (2) bar (psi)
LDC-C030100-*xxxxx*	LDC-030-100-CP	N/A (3)	N/A (3)
LDC-C030200-*xxxxx*	LDC-030-200-CP	N/A (3)	N/A (3)
LDC-C050100-*xxxxx*	LCC-050-100-CP	N/A (3)	0.41 (6)
LDC-C050200-*xxxxx*	LDC-050-200-CP	N/A (3)	0.48 (7)
LDC-C050300-*xxxxx*	LCC-050-300-CP	N/A (3)	0.55 (8)
LDC-C075200-*xxxxx*	LDC-075-200-CP	N/A (3)	0.48 (7)
LDC-C075300-*xxxxx*	LDC-075-300-CP	N/A (3)	0.55 (8)
LDC-C075400-*xxxxx*	LDC-075-400-CP	N/A (3)	0.69 (10)
LDC-C100300-*xxxxx*	LDC-100-300-CP	61.4 (130)	0.69 (10)
LDC-C100400-*xxxxx*	LDC-100-400-CP	N/A (3)	0.83 (12)
LDC-C100600-*xxxxx*	LDC-100-600-CP	47.2 (100)	0.97 (14)
LDC-C150400-*xxxxx*	LDC-150-400-CP	N/A (3)	0.83 (12l) (4)
LDC-C150600-*xxxxx*	LDC-150-600-CP	N/A (3)	0.93 (13.5) (4)

(1) These are the flow rates required to maintain air pressure at 0.689 bar (10 lb/in.^{2}).

(2) These are the flow rates required to maintain water pressure at 3.8 L/min (1 gal/min).

(3) This flow rate is not available. Call Application Engineering (631-344-6600) for assistance.

(4) These are the flow rates required to maintain water pressure at 7.57 L/min (2 gal/min).

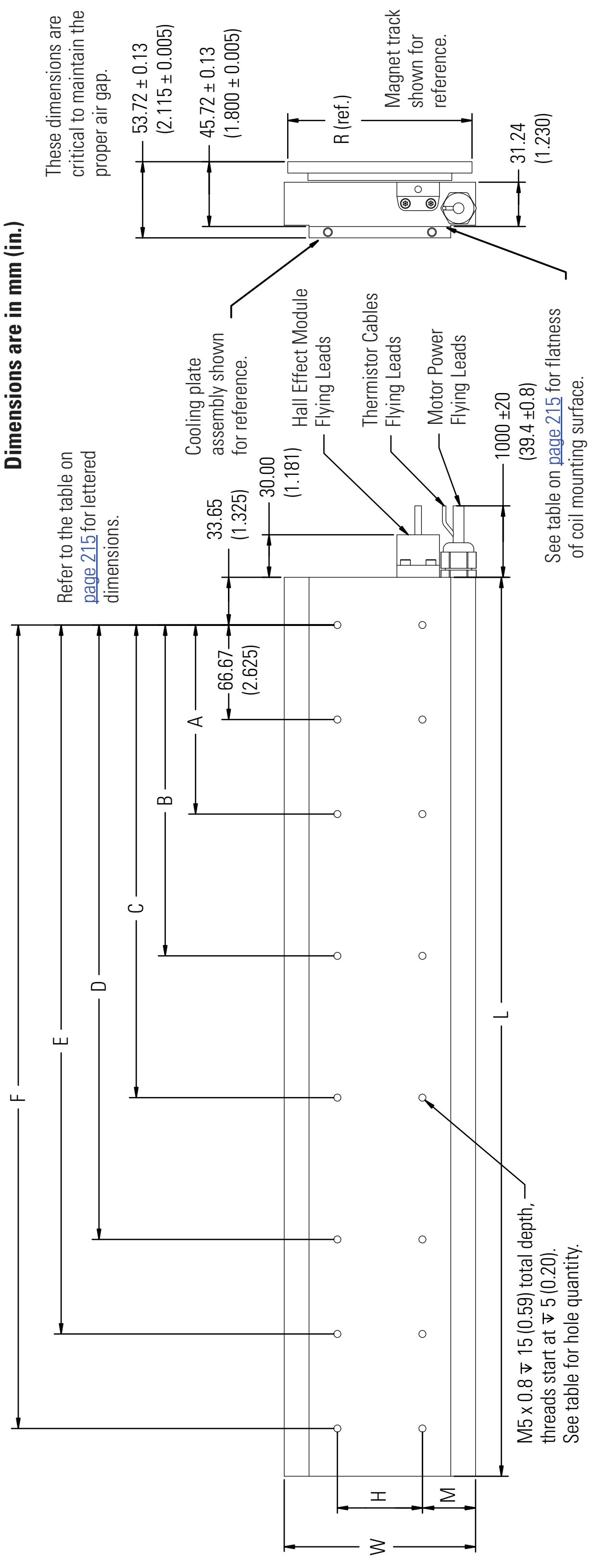
LDC-Series Iron Core Linear Motor Component Dimensions

LDC-Series iron core linear motor components are designed to metric dimensions. Inch dimensions are conversions from millimeters. Untoleranced dimensions are for reference.

LDC-Series Iron Core Linear Motor Coil Dimensions

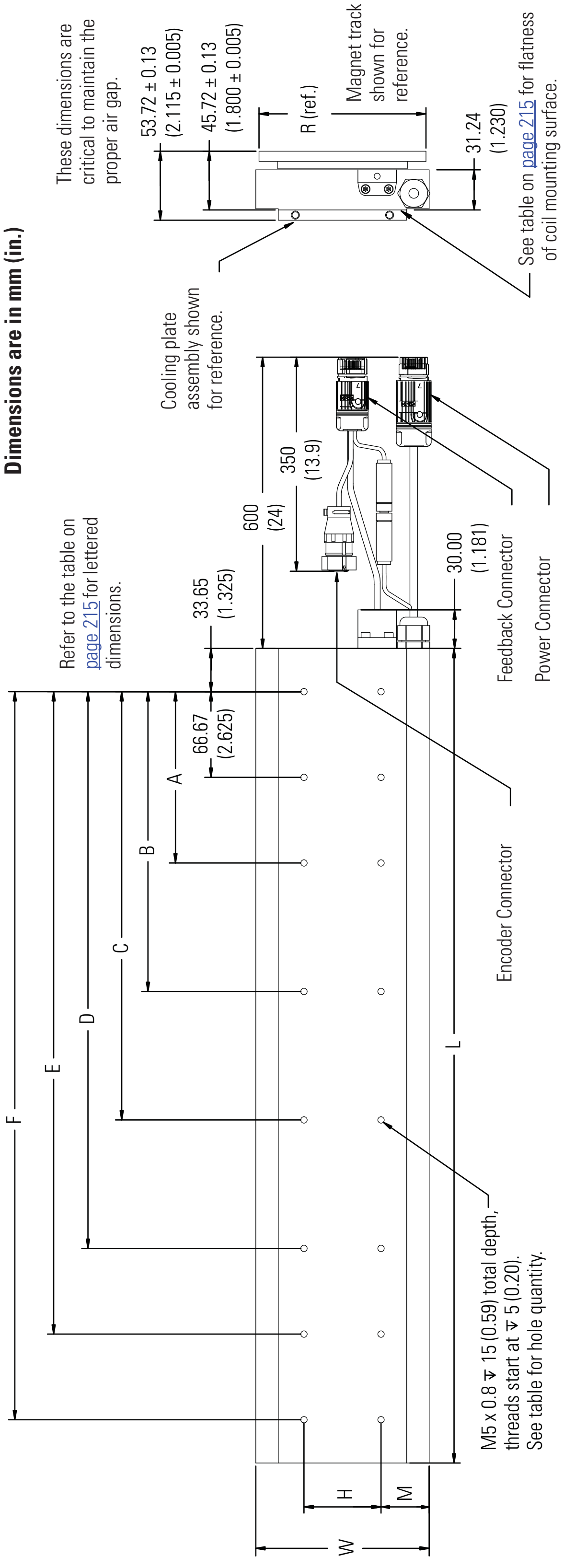
LDC-C030/050/075/100*xxx*-xHT20 Motor Coil Dimensions (flying leads)

Dimensions are in mm (in.)



LDC-C030/050/075/100*xxx*-xHT11 Motor Coil Dimensions (connectors)

Dimensions are in mm (in.)



LDC-C030/050/075/100*xxx*-*x*HT*xx* Motor Coil Dimensions

Cat. No.	L mm (in.)	W [1] mm (in.)	A mm (in.)	B mm (in.)	C mm (in.)	D mm (in.)	E mm (in.)	F mm (in.)	H mm (in.)	M mm (in.)	R mm (in.)	Hole Qty
LDC-C030100-DHT*xx*	134.0 (5.28)	65.00 (2.559)	–	–	–	–	–	–	15.00 (0.591)	25.00 (0.984)	60.00 (2.362)	4
LDC-C030200-xHT*xx*	234.0 (9.21)		100.00 (3.937)	166.67 (6.562)	–	–	–	–				8
LDC-C050100-DHT*xx*	134.0 (5.28)	85.00 (3.346)	–	–	–	–	–	–	25.00 (0.984)	30.00 (1.181)	80.00 (3.150)	4
LDC-C050200-xHT*xx*	234.0 (9.21)		100.00 (3.937)	166.67 (6.562)	–	–	–	–				8
LDC-C050300-xHT*xx*	334.0 (13.15)		133.33 (5.249)	200.00 (7.874)	266.67 (10.499)	–	–	–				10
LDC-C075200-xHT*xx*	234.0 (9.21)	110.00 (4.331)	100.00 (3.937)	166.67 (6.562)	–	–	–	–	40.00 (1.575)	35.00 (1.378)	105.00 (4.134)	8
LDC-C075300-xHT*xx*	334.0 (13.15)		133.33 (5.249)	200.00 (7.874)	266.67 (10.499)	–	–	–				10
LDC-C075400-xHT*xx*	434.0 (17.09)		133.33 (5.249)	233.33 (9.186)	300.00 (11.811)	366.67 (14.436)	–	–				12
LDC-C100300-xHT*xx*	334.0 (13.15)	135.00 (5.315)	133.33 (5.249)	200.00 (7.874)	266.67 (10.499)	–	–	–	60.00 (2.362)	37.50 (1.476)	130.00 (5.118)	10
LDC-C100400-xHT*xx*	434.0 (17.09)		133.33 (5.249)	233.33 (9.186)	300.00 (11.811)	366.67 (14.436)	–	–				12
LDC-C100600-xHT*xx*	634.0 (25.31)		133.33 (5.249)	233.33 (9.186)	333.33 (13.123)	433.33 (17.060)	500.00 (19.686)	566.66 (22.310)				16

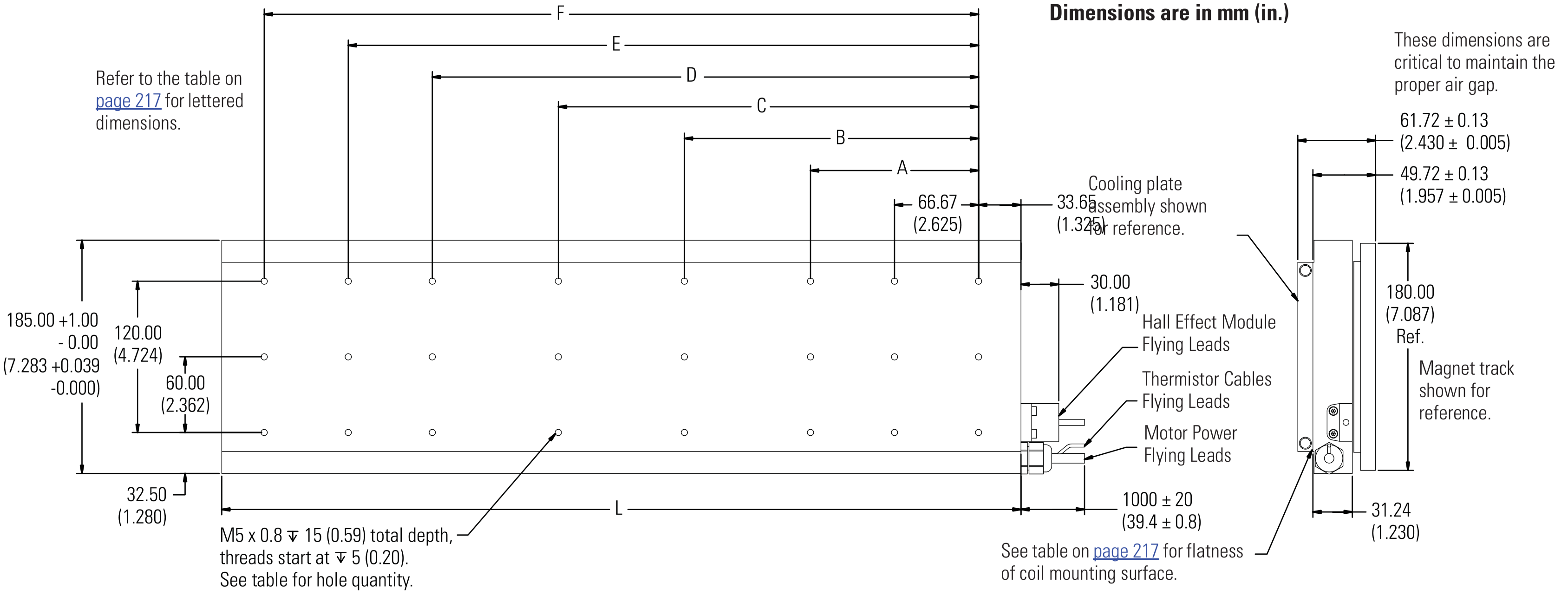
[1] Tolerance for W dimension is +1.00 mm (+0.039 in.), -0.00 mm (-0.000 in.)

Cat. No.	Power Cable Gauge [1] mm² (AWG)
LDC-C030*xxx*-*x*HT*xx*	0.75 (18)
LDC-C050*xxx*-*x*HT*xx*	
LDC-C075*xxx*-*x*HT*xx*	
LDC-C100300-*x*HT*xx*	
LDC-C100400-*x*HT*xx*	
LDC-C100600-DHT*xx*	1.5 (16)
LDC-C100600-EHT*xx*	0.75 (18)

Cat. No.	Flatness mm/300 x 300 (in./12 x 12)
LDC-C030*xxx*-*x*HT*xx*	0.25 (0.01)
LDC-C050100-DHT*xx*	
LDC-C050200-*x*HT*xx*	
LDC-C050300-*x*HT*xx*	0.38 (0.15)
LDC-C075200-*x*HT*xx*	0.25 (0.01)
LDC-C075300-*x*HT*xx*	0.38 (0.015)
LDC-C075400-*x*HT*xx*	0.64 (0.25)
LDC-C100300-*x*HT*xx*	0.38 (0.015)
LDC-C100400-*x*HT*xx*	0.64 (0.25)
LDC-C100600-*x*HT*xx*	0.89 (0.035)

[1] Refer to Cable Specifications on page 217 for additional cable specifications.

LDC-C0150*xxx*-*x*HT20 Motor Coil Dimensions (flying leads)



Dimensions are in mm (in.)

Refer to the table on page 217 for lettered dimensions.

These dimensions are critical to maintain the proper air gap.

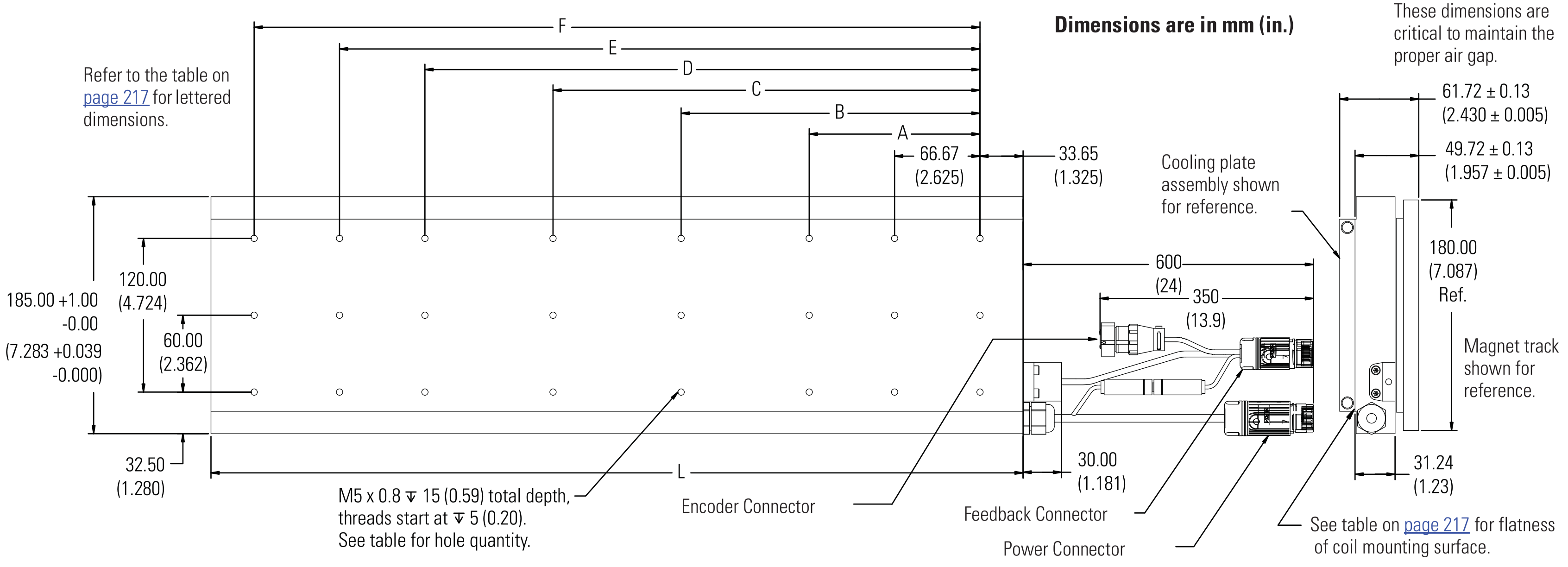
Cooling plate assembly shown for reference.

Magnet track shown for reference.

M5 x 0.8 ⩡ 15 (0.59) total depth, threads start at ⩡ 5 (0.20). See table for hole quantity.

See table on page 217 for flatness of coil mounting surface.

LDC-C150*xxx*-*x*HT11 Motor Coil Dimensions (connectors)



Dimensions are in mm (in.)

Refer to the table on page 217 for lettered dimensions.

These dimensions are critical to maintain the proper air gap.

Cooling plate assembly shown for reference.

Magnet track shown for reference.

M5 x 0.8 ⩡ 15 (0.59) total depth, threads start at ⩡ 5 (0.20). See table for hole quantity.

See table on page 217 for flatness of coil mounting surface.

LDC-C150*xxx*-*x*HT*xx* Motor Coil Dimensions

Cat. No.	L mm (in.)	A mm (in.)	B mm (in.)	C mm (in.)	D mm (in.)	E mm (in.)	F mm (in.)	Hole Qty
LDC-C150400-*x*HT*xx*	434.0 (17.09)	133.33 (5.249)	233.33 (9.186)	300.00 (11.811)	366.67 (14.436)	–	–	18
LDC-C150600-*x*HT*xx*	634.0 (25.31)	133.33 (5.249)	233.33 (9.186)	333.33 (13.123)	433.33 (17.060)	500.00 (19.686)	566.66 (22.310)	24

Cable Specifications

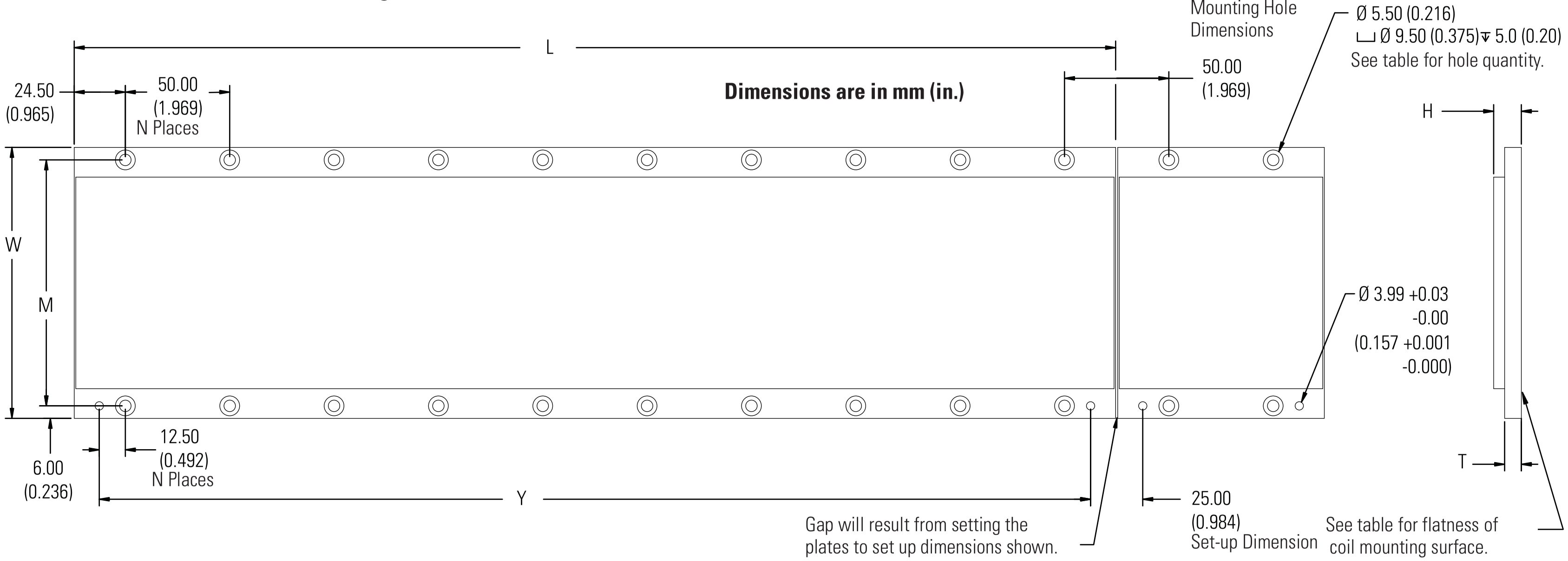
Cable [(1)]	Conductors	Gauge mm^2 (AWG)	Shield Type	Cable Dia. mm (in.)	Static Bend Radius mm (in.)
Power [(2)]	4	0.75 (18)	Braid	7.0 (0.28)	18.0 (0.70)
Power [(2)]	4	1.5 (16)	Braid	8.0 (0.32)	17.0 (0.67)
Power [(2)]	4	0.50 (20)	Braid	6.4 (0.25)	17.0 (0.67)
Thermistor	2	0.14 (26)	None	4.0 (0.16)	10.0 (0.40)
Hall Module	6	0.25 (24)	Foil	5.0 (0.20)	15.0 (0.59)

(1) All cables are non-flex.
(2) Power cable specification is dependent on coil used. See Power Cable Gauge tables on page 215 and page 217.

Cat. No.	Power Cable Gauge mm^2 (AWG)	Flatness mm/300 x 300 (in./12 x 12)
LDC-C150400-*x*HT*xx*	0.82 (18)	0.64 (0.025)
LDC-C150600-DHT*xx*		0.89 (0.035)
LDC-C150600-EHT*xx*	1.31 (16)	

LDC-Series Iron Core Linear Motor Magnet Track Dimensions

LDC-Series Iron Core Linear Motor Magnet Track Dimensions

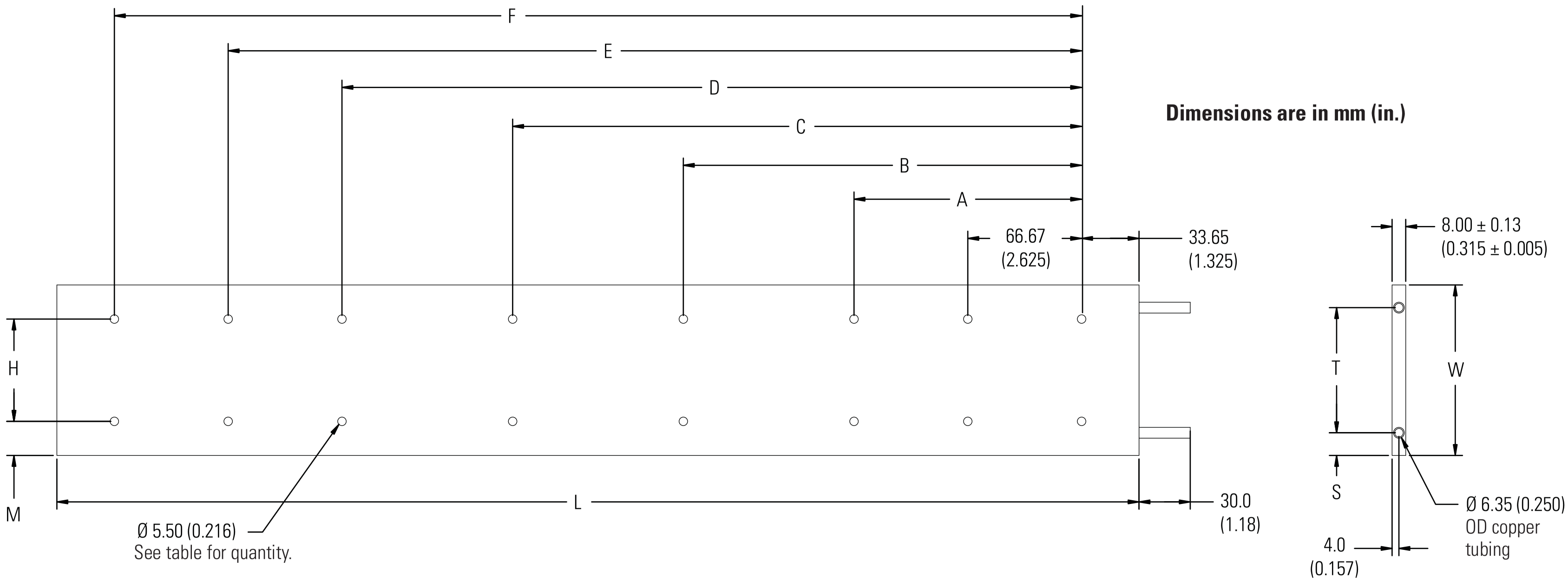


Cat. No.	L [(1)] mm (in.)	Y [(2)] mm (in.)	W mm (in.)	M mm (in.)	H [(3)] mm (in.)	T mm (in.)	N	Hole Qty	Flatness [(4)] mm/300 x 300 (in./12 x 12)
LDC-M030100	99.0 (3.90)	75.00 (2.953)	60.0 (2.36)	48.00 (1.890)	13.26 (0.522)	8.00 (0.315)	1	4	0.06 (0.002)
LDC-M050100			80.0 (3.15)	68.00 (2.677)					
LDC-M075100			105.0 (4.13)	93.00 (3.661)					0.13 (0.005)
LDC-M100100			130.0 (5.12)	118.00 (4.646)					
LDC-M150100			180.0 (7.09)	168.00 (6.614)	17.26 (0.680)	12.00 (0.472)			
LDC-M030500	499.0 (19.65)	475.00 (18.70)	60.0 (2.36)	48.00 (1.890)	13.26 (0.522)	8.00 (0.315)	9	20	0.50 (0.20)
LDC-M050500			80.0 (3.15)	68.00 (2.677)					
LDC-M075500			105.0 (4.13)	93.00 (3.661)					0.90 (0.035)
LDC-M100500			130.0 (5.12)	118.00 (4.646)					
LDC-M150500			180.0 (7.09)	168.00 (6.614)	17.26 (0.680)	12.00 (0.472)			

(1) Tolerance for L dimension is ±0.25 mm (±0.010 in.).
(2) Tolerance for Y dimension is ±0.08 mm (±0.003 in.).
(3) Tolerance for H dimension is ±0.16 mm (±0.006 in.).
(4) Specified flatness is in the free state.

LDC-Series Iron Core Linear Motor Coil Cooling Plate Dimensions

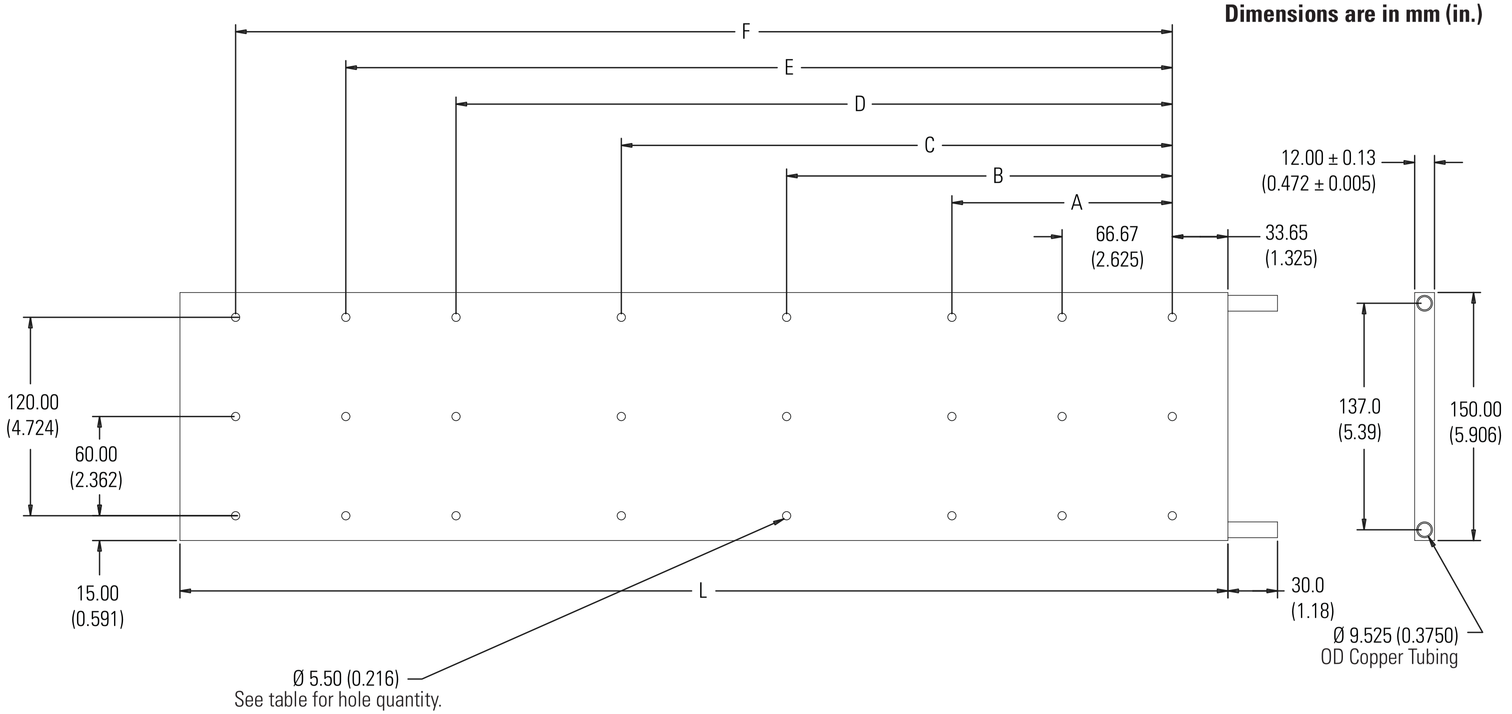
LDC-030/050/075/100-*xxx*-CP Motor Coil Cooling Plate Dimensions



Dimensions are in mm (in.)

Cat. No.	L mm (in.)	W mm (in.)	A mm (in.)	B mm (in.)	C mm (in.)	D mm (in.)	E mm (in.)	F mm (in.)	H mm (in.)	M mm (in.)	T mm (in.)	S mm (in.)	Hole Qty
LDC-030-100-CP	134.0 (5.28)	38.00 (1.496)	–	–	–	–	–	–	15.00 (0.591)	11.50 (0.453)	27.5 (1.08)	5.25 (0207)	4
LDC-030-200-CP	234.0 (9.21)		100.00 (3.937)	166.67 (6.562)	–	–	–	–					8
LCC-050-100-CP	134.0 (5.28)	50.00 (1.969)	–	–	–	–	–	–	25.00 (0.984)	12.50 (0.492)	38.1 (1.50)	5.95 (0.234)	4
LDC-050-200-CP	234.0 (9.21)		100.00 (3.937)	166.67 (6.562)	–	–	–	–					8
LCC-050-300-CP	334.0 (131.15)		133.33 (5.249)	200.00 (7.874)	266.67 (10.499)	–	–	–					10
LDC-075-200-CP	234.0 (9.21)	75.00 (2.953)	100.00 (3.937)	166.67 (6.562)	–	–	–	–	40.00 (1.575)	17.50 (0.689)	50.8 (2.00)	12.10 (0.476)	8
LDC-075-300-CP	334.0 (131.15)		133.33 (5.249)	200.00 (7.874)	266.67 (10.499)	–	–	–					10
LDC-075-400-CP	434.0 (17.09)		133.33 (5.249)	233.33 (9.186)	300.00 (11.811)	366.67 (14.436)	–	–					12
LDC-100-300-CP	334.0 (131.15)	100.00 (3.937)	133.33 (5.249)	200.00 (7.874)	266.67 (10.499)	–	–	–	60.00 (2.362)	20.00 (0.787)	73.4 (2.89)	13.30 0.524)	10
LDC-100-400-CP	434.0 (17.09)		133.33 (5.249)	233.33 (9.186)	300.00 (11.811)	366.67 (14.436)	–	–					12
LDC-100-600-CP	634.0 (25.31)		133.33 (5.249)	233.33 (9.186)	333.33 (13.123)	433.33 (17.060)	500.00 (19.686)	566.66 (22.310)					16

LDC-150-*xxx*-CP Motor Coil Cooling Plate Dimensions


Dimensions are in mm (in.)

Ø 5.50 (0.216)
See table for hole quantity.

Ø 9.525 (0.3750)
OD Copper Tubing

Cat. No.	**L** mm (in.)	**A** mm (in.)	**B** mm (in.)	**C** mm (in.)	**D** mm (in.)	**E** mm (in.)	**F** mm (in.)	**Hole Qty**
LDC-150-400-CP	434.0 (17.09)	133.33 (5.249)	233.33 (9.186)	300.00 (11.811)	366.67 (14.436)	–	–	18
LDC-150-600-CP	634.0 (25.31)	133.33 (5.249)	233.33 (9.186)	333.33 (13.123)	433.33 (17.060)	500.00 (19.686)	566.66 (22.310)	24

LDC-Series Iron Core Linear Motor Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your actuator. For questions regarding product availability, contact your Allen-Bradley distributor.

LDC-Series Iron Core Linear Motor Catalog Numbers

LDC - C *xxx xxx* - *x x* T *x x*

Cable Termination
0 = Flying leads
1 = Circular DIN connector

Cable Length
0 = 300 mm (12.45 in.)
1 = 600 mm (23.62in.)
2 = 1000 mm (39.37 in.)

Thermal Protection
T = PTC Thermal sensor and switch

Feedback
H = Hall effect (trapezoidal)
N = No feedback

Winding Code
D = D winding
E = E winding

Coil Length (1)
100 = 100 mm (4.15 in.)
200 = 200 mm (8.30 in.)
300 = 300 mm (12.45 in.)
400 = 400 mm (16.60 in.)
600 = 600 mm (23.62 in.)

Frame Size (1)
030 075150
050 100

Coil Designation
C = Standard coil

Motor Series
LDC = LDC-Series

(1) Not all combinations are available. Only the configurations as listed in LDC-Series Iron Core Linear Motor Performance Specifications on page 205 are available.

LDC-Series Iron Core Linear Motor Magnet Track and Cooling Plate Catalog Numbers

LDC - M *xxx xxx*

Magnet Track Length
100 = 100 mm (4.15 in.)
500 = 500 mm (19.68 in.)

Frame Size
030 075 150
050 100

Magnet Plate Designation
M = Magnet plate

Motor Series
LDC = LDC-Series

LDC - *xxx* - *xxx* - CP

Cooling Plate

Coil Length
100 = 100 mm (4.15 in.)
200 = 200 mm (8.30 in.)
300 = 300 mm (12.45 in.)
400 = 400 mm (16.60 in.)
600 = 600 mm (23.62 in.)

Frame Size
030 075 150
050 100

Motor Series
LDC = LDC-Series

Notes:

LDL-Series Ironless Linear Servo Motors

The LDL-Series ironless linear motors address a growing interest in linear motor technology as it becomes more affordable and is increasingly recognized as a practical means of improving machine performance. With the ironless product design, you now have cost-effective options to help you improve machine throughput while reducing maintenance and downtime.

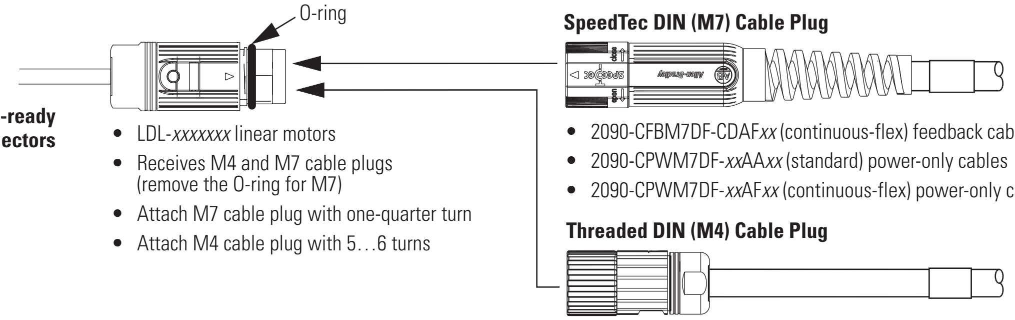
For drive compatibility, refer to Servo Drives on page 14.

LDL-Series Ironless Linear Motor Features

- No magnetic attraction between the coil and magnet channel allows for the use of smaller, less expensive linear bearings.
- No external magnetic field to have to shield in magnetic sensitive applications.
- Non-cogging technology for super smooth motion.
- Very high acceleration and speeds up to 10 m/s (32.8 ft/s) greatly increase the throughput of your machine.
- No limits to travel distance. Ability to achieve high speeds over short and long travels.
- Direct drive technology for extreme servo responsiveness.
- No-wear, high reliability parts increase productivity.
- Peak forces to 1977 N (444 lb).
- Ability to size and optimize LDC-Series linear motors and corresponding servo drives by using Motion Analyzer software reduces product selection time and minimizes cost.
- Full set-up and programming support through RSLogix 5000 software reduces set-up time.

Motor Connector/Cable Compatibility

LDL-Series Iron Core Linear Motors



SpeedTec-ready DIN Connectors

- LDL-*xxxxxxx* linear motors
- Receives M4 and M7 cable plugs (remove the O-ring for M7)
- Attach M7 cable plug with one-quarter turn
- Attach M4 cable plug with 5…6 turns

SpeedTec DIN (M7) Cable Plug

- 2090-CFBM7DF-CDAF*xx* (continuous-flex) feedback cables
- 2090-CPWM7DF-*xx*AA*xx* (standard) power-only cables
- 2090-CPWM7DF-*xx*AF*xx* (continuous-flex) power-only cables

Threaded DIN (M4) Cable Plug

2090-XXNFMF-S*xx* (standard) feedback cables

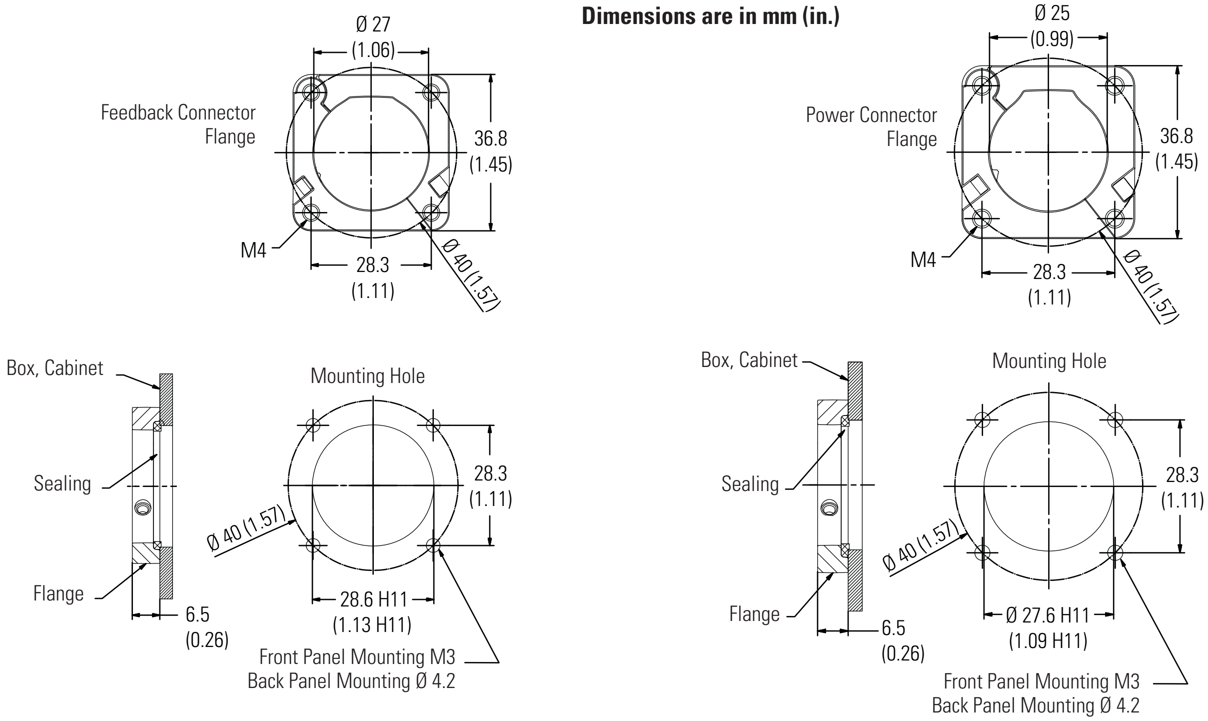
LDL-Series Ironless Linear Motor Accessories

- Bulk head connector kit.
- Encoder connector kit.
- Hall effect replacement module for connectorized coil.
- Hall effect replacement module for flying-lead coil.

Accessories for LDL-Series Ironless Linear Motors

Cat. No.	Accessory	Description
LDC-BULK-HD	Bulk head connector kit	For easy mounting of flex cable to non-flex cables. Kit includes flange for feedback and power connectors, o-rings, and nut.
LDC-ENC-CNCT	Encoder connector kit	Adapts your encoder to the feedback cable on the Hall effect module.
LDL-HALL-C	Hall effect module	Replacement module for use with connectorized coil.
LDL-HALL-F		Replacement module for use with flying-lead coil.

Bulk Head Connector Flange Dimensions



Dimensions are in mm (in.)

LDL-Series Ironless Linear Motor Performance Specifications

These performance specifications apply to all LDL-Series ironless linear motors.

Common Performance Specifications

Attribute	Value
Motor type	3 phase, wye winding, synchronous permanent magnet stator, non-ventilated linear motor.
Operating speed, max	10 m/s (32.8 ft/s)
Operating voltage, (not for direct connection to AC line)	230V AC rms
Dielectric rating of motor power connections (U,V,W), to ground for 1.0 s (1)	1500V AC rms, 50/60 Hz
Cogging torque	Zero
Applied bus voltage, max (2)	325V DC
Electrical cycle length	60 mm (2.36 in.)
Coil temperature, max	130 °C (266 °F)
Insulation class	130 °C (266 °F) Class B
Thermal time constant, Ref, winding to ambient	35 min
Paint color	Black

(1) Tested during manufacturing process, Do not re-apply test voltage. Contact Application Engineering (631.344.6600) for advice on testing coils post production.

(2) Maximum cable length 10 m (32.8 ft). Contact Application Engineering (631.344.6600) for applications requiring longer cables.

LDL-Series Ironless Linear Motor (standard 30 mm frame size)

Attribute	Units	Symbol	LDL-N030120-D*x*T*xx*	LDL-N030240-D*x*T*xx*	LDL-N030240-E*x*T*xx*
Force, continuous (1) (2) (3) (4)	N (lbf)	F_c	63 (14)	126 (28)	
Force, peak (5)	N (lbf)	F_p	209 (47)	417 (94)	
Thermal resistance	°C/W	R_{th}	1.73	0.86	
Force constant (6) (7) (8)	N/A_{pk} (lbf/A_{pk})	K_f	21.0 (4.7)	21.0 (4.7)	42.0 (9.4)
Back EMF constant p-p (6) (7) (8)	V_p/m/s (V_p/in/s)	K_e	24.8 (0.6)	24.8 (0.6)	49.6 (1.3)
Current, peak (5) (7)	A_{pk} (A_{rms})	I_p	9.9 (7.0)	19.9 (14.0)	9.9 (7.0)
Current, continuous (1) (2) (3) (4)	A_{pk} (A_{rms})	I_c	3.0 (2.1)	6.0 (4.2)	3.0 (2.1)
Resistance p-p @ 20 °C (68 °F) (6) (8)	Ohms	R_{20}	5.41	2.70	10.82
Inductance p-p (6) (8)	mH	L	8.43	4.22	16.86
Magnetic attraction	N (lbf)	F_a	0 (0)		

LDL-Series Ironless Linear Motor (thick 30 mm frame size)

Attribute	Units	Symbol	LDL-T030120-D*x*T*xx*	LDL-T030240-D*x*T*xx*	LDL-T030240-E*x*T*xx*
Force, continuous (1) (2) (3) (4)	N (lbf)	F_c	72 (16)	144 (32)	
Force, peak (5)	N (lbf)	F_p	239 (54)	479 (108)	
Thermal resistance	°C/W	R_{th}	1.31	0.65	
Force constant (6) (7) (8)	N/A_{pk} (lbf/A_{pk})	K_f	24.1 (5.4)	24.1 (5.4)	48.2 (10.8)
Back EMF constant p-p (6) (7) (8)	V_p/m/s (V_p/in/s)	K_e	28.5 (0.7)	28.5 (0.7)	56.9 (1.4)
Current, peak (5) (7)	A_{pk} (A_{rms})	I_p	9.9 (7.0)	19.9 (14.0)	9.9 (7.0)
Current, continuous (1) (2) (3) (4)	A_{pk} (A_{rms})	I_c	3.0 (2.1)	6.0 (4.2)	3.0 (2.1)
Resistance p-p @ 20 °C (68 °F) (6) (8)	Ohms	R_{20}	7.15	3.57	14.29
Inductance p-p (6) (8)	mH	L	13.40	6.70	26.80
Magnetic attraction	N (lbf)	F_a	0 (0)		

(1) Coils at maximum temperature, 130 °C (266 °F), mounted to an aluminium heat sink whose area is noted in table on page 231, and at 40 °C (104 °F) ambient.

(2) Continuous force and current based on coil moving with all phases sharing the same load in sinusoidal commutation.

(3) For standstill conditions, multiply continuous force and continuous current by 0.9.

(4) Coil mountings on either of the two narrow sides reduces continuous force by 10%.

(5) Calculated at 11% duty cycle for 1.0 second max. Some applications may produce significantly higher peak forces. Call Applications Engineering (631.344.6600) for details.

(6) Winding parameters listed are measured line-to-line (phase-to-phase).

(7) Currents and voltages listed are measured 0-peak of the sine wave unless noted as rms.

(8) Specifications are ±10%. Phase-to-phase inductance is ±30%.

LDL-Series Ironless Linear Motor (standard 50 mm frame size)

Attribute	Units	Symbol	LDL-N050120-D*x*T*xx*	LDL-N050240-D*x*T*xx*	LDL-N050240-E*x*T*xx*
Force, continuous (1) (2) (3) (4)	N (lbf)	F_c	96 (22)	191 (43)	
Force, peak (5)	N (lbf)	F_p	317 (71)	635 (143)	
Thermal resistance	°C/W	R_{th}	1.58	0.79	
Force constant (6) (7) (8)	N/A_{pk} (lbf/A_{pk})	K_f	35.0 (7.9)	35.0 (7.9)	70.0 (15.7)
Back EMF constant p-p (6) (7) (8)	$V_p/m/s$ $(V_p/in/s)$	K_e	41.3 (1.1)	41.3 (1.1)	82.7 (2.1)
Current, peak (5) (7)	A_{pk} (A_{rms})	I_p	9.1 (6.4)	18.1 (12.8)	9.1 (6.4)
Current, continuous (1) (2) (3) (4)	A_{pk} (A_{rms})	I_c	2.7 (1.9)	5.5 (3.9)	2.7 (1.9)
Resistance p-p @ 20 °C (68 °F) (6) (8)	Ohms	R_{20}	7.11	3.56	14.22
Inductance p-p (6) (8)	mH	L	11.08	5.54	22.16
Magnetic attraction	N (lbf)	F_a	0 (0)		

Attribute	Units	Symbol	LDL-N050360-D*x*T*xx*	LDL-N050360-E*x*T*xx*	LDL-N050480-D*x*T*xx*	LDL-N050480-E*x*T*xx*
Force, continuous (1) (2) (3) (4)	N (lbf)	F_c	287 (65)		383 (86)	
Force, peak (5)	N (lbf)	F_p	952 (214)		1269 (285)	
Thermal resistance	°C/W	R_{th}	0.53		0.39	
Force constant (6) (7) (8)	N/A_{pk} (lbf/A_{pk})	K_f	35.0 (7.9)	105.0 (23.6)	35.0 (7.9)	70.0 (15.7)
Back EMF constant p-p (6) (7) (8)	$V_p/m/s$ $(V_p/in/s)$	K_e	41.3 (1.1)	124.0 (3.2)	41.3 (1.1)	82.7 (2.1)
Current, peak (5) (7)	A_{pk} (A_{rms})	I_p	27.2 (19.2)	9.1 (6.4)	36.3 (25.6)	18.1 (12.8)
Current, continuous (1) (2) (3) (4)	A_{pk} (A_{rms})	I_c	8.2 (5.8)	2.7 (1.9)	10.9 (7.7)	5.5 (3.9)
Resistance p-p @ 20 °C (68 °F) (6) (8)	Ohms	R_{20}	2.37	21.33	1.78	7.11
Inductance p-p (6) (8)	mH	L	3.69	33.25	2.77	11.08
Magnetic attraction	N (lbf)	F_a	0 (0)			

(1) Coils at maximum temperature, 130 °C (266 °F), mounted to an aluminium heat sink whose area is noted in table on page 231, and at 40 °C (104 °F) ambient.
(2) Continuous force and current based on coil moving with all phases sharing the same load in sinusoidal commutation.
(3) For standstill conditions, multiply continuous force and continuous current by 0.9.
(4) Coil mountings on either of the two narrow sides reduces continuous force by 10%.
(5) Calculated at 11% duty cycle for 1.0 second max. Some applications may produce significantly higher peak forces. Call Applications Engineering (631.344.6600) for details.
(6) Winding parameters listed are measured line-to-line (phase-to-phase).
(7) Currents and voltages listed are measured 0-peak of the sine wave unless noted as rms.
(8) Specifications are ±10%. Phase-to-phase inductance is ±30%.

LDL-Series Ironless Linear Motor (thick 50 mm frame size)

Attribute	**Units**	**Symbol**	**LDL-T050120-D*x*T*xx***	**LDL-T050240-D*x*T*xx***	**LDL-T050240-E*x*T*xx***
Force, continuous (1) (2) (3) (4)	N (lbf)	F_c	110 (25)	220 (49)	
Force, peak (5)	N (lbf)	F_p	364 (82)	728 (164)	
Thermal resistance	°C/W	R_{th}	1.19	0.60	
Force constant (6) (7) (8)	N/A_{pk} (lbf/A_{pk})	K_f	40.2 (9.0)	40.2 (9.0)	80.4 (18.1)
Back EMF constant p-p (6) (7) (8)	V_p/m/s (V_p/in/s)	K_e	47.4 (1.2)	47.4 (1.2)	94.9 (2.4)
Current, peak (5) (7)	A_{pk} (A_{rms})	I_p	9.1 (6.4)	18.1 (12.8)	9.1 (6.4)
Current, continuous (1) (2) (3) (4)	A_{pk} (A_{rms})	I_c	2.7 (1.9)	5.5 (3.9)	2.7 (1.9)
Resistance p-p @ 20 °C (68 °F) (6) (8)	Ohms	R_{20}	9.42	4.71	18.83
Inductance p-p (6) (8)	mH	L	18	9	35.31
Magnetic attraction	N (lbf)	F_a	0 (0)		

Attribute	**Units**	**Symbol**	**LDL-T050360-D*x*T*xx***	**LDL-T050360-E*x*T*xx***	**LDL-T050480-D*x*T*xx***	**LDL-T050480-E*x*T*xx***
Force, continuous (1) (2) (3) (4)	N (lbf)	F_c	329 (74)		439 (99)	
Force, peak (5)	N (lbf)	F_p	1093 (246)		1457 (327)	
Thermal resistance	°C/W	R_{th}	0.40		0.30	
Force constant (6) (7) (8)	N/A_{pk} (lbf/A_{pk})	K_f	40.2 (9.0)	120.5 (27.1)	40.2 (9.0)	80.4 (18.1)
Back EMF constant p-p (6) (7) (8)	V_p/m/s (V_p/in/s)	K_e	47.4 (1.2)	142.3 (3.6)	47.4 (1.2)	94.9 (2.4)
Current, peak (5) (7)	A_{pk} (A_{rms})	I_p	27.2 (19.2)	9.1 (6.4)	36.3 (25.6)	18.1 (12.8)
Current, continuous (1) (2) (3) (4)	A_{pk} (A_{rms})	I_c	8.2 (5.8)	2.7 (1.9)	10.9 (7.7)	5.5 (3.9)
Resistance p-p @ 20 °C (68 °F) (6) (8)	Ohms	R_{20}	3.14	28.25	2.35	9.42
Inductance p-p (6) (8)	mH	L	5.88	52.96	4.41	17.65
Magnetic attraction	N (lbf)	F_a	0 (0)			

(1) Coils at maximum temperature, 130 °C (266 °F), mounted to an aluminium heat sink whose area is noted in table on page 231, and at 40 °C (104 °F) ambient.
(2) Continuous force and current based on coil moving with all phases sharing the same load in sinusoidal commutation.
(3) For standstill conditions, multiply continuous force and continuous current by 0.9.
(4) Coil mountings on either of the two narrow sides reduces continuous force by 10%.
(5) Calculated at 11% duty cycle for 1.0 second max. Some applications may produce significantly higher peak forces. Call Applications Engineering (631.344.6600) for details.
(6) Winding parameters listed are measured line-to-line (phase-to-phase).
(7) Currents and voltages listed are measured 0-peak of the sine wave unless noted as rms.
(8) Specifications are ±10%. Phase-to-phase inductance is ±30%.

LDL-Series Ironless Linear Motor (standard 75 mm frame size)

Attribute	Units	Symbol	LDL-N075480-D*x*T*xx*	LDL-N075480-E*x*T*xx*
Force, continuous (1) (2) (3) (4)	N (lbf)	F_c	519 (117)	
Force, peak (5)	N (lbf)	F_p	1723 (387)	
Thermal resistance	°C/W	R_{th}	0.37	
Force constant (6) (7) (8)	N/A_{pk} (lbf/A_{pk})	K_f	52.5 (11.8)	105.0 (23.6)
Back EMF constant p-p (6) (7) (8)	$V_p/m/s$ $(V_p/in/s)$	K_e	62.0 (1.6)	124.0 (3.2)
Current, peak (5) (7)	A_{pk} (A_{rms})	I_p	32.8 (23.2)	16.4 (11.6)
Current, continuous (1) (2) (3) (4)	A_{pk} (A_{rms})	I_c	9.9 (7.0)	4.9 (3.5)
Resistance p-p @ 20 °C (68 °F) (6) (8)	Ohms	R_{20}	2.31	9.24
Inductance p-p (6) (8)	mH	L	3.60	14.40
Magnetic attraction	N (lbf)	F_a	0 (0)	

LDL-Series Ironless Linear Motor (thick 75 mm frame size)

Attribute	Units	Symbol	LDL-T075480-D*x*T*xx*	LDL-T075480-E*x*T*xx*
Force, continuous (1) (2) (3) (4)	N (lbf)	F_c	596 (134)	
Force, peak (5)	N (lbf)	F_p	1977 (444)	
Thermal resistance	°C/W	R_{th}	0.28	
Force constant (6) (7) (8)	N/A_{pk} (lbf/A_{pk})	K_f	60.3 (13.5)	120.5 (27.1)
Back EMF constant p-p (6) (7) (8)	$V_p/m/s$ $(V_p/in/s)$	K_e	71.2 (1.8)	142.3 (3.6)
Current, peak (5) (7)	A_{pk} (A_{rms})	I_p	32.8 (23.2)	16.4 (11.6)
Current, continuous (1) (2) (3) (4)	A_{pk} (A_{rms})	I_c	9.9 (7.0)	4.9 (3.5)
Resistance p-p @ 20 °C (68 °F) (6) (8)	Ohms	R_{20}	3.06	12.25
Inductance p-p (6) (8)	mH	L	5.74	22.97
Magnetic attraction	N (lbf)	F_a	0 (0)	

(1) Coils at maximum temperature, 130 °C (266 °F), mounted to an aluminium heat sink whose area is noted in table on page 231, and at 40 °C (104 °F) ambient.
(2) Continuous force and current based on coil moving with all phases sharing the same load in sinusoidal commutation.
(3) For standstill conditions, multiply continuous force and continuous current by 0.9.
(4) Coil mountings on either of the two narrow sides reduces continuous force by 10%.
(5) Calculated at 11% duty cycle for 1.0 second max. Some applications may produce significantly higher peak forces. Call Applications Engineering (631.344.6600) for details.
(6) Winding parameters listed are measured line-to-line (phase-to-phase).
(7) Currents and voltages listed are measured 0-peak of the sine wave unless noted as rms.
(8) Specifications are ±10%. Phase-to-phase inductance is ±30%.

LDL-Series Linear Motors System Combinations

For LDL-Series linear motors and	Refer to
Kinetix 6000 (230V) drives	page 692
Kinetix 2000 (230V drives)	page 712
Kinetix 3 (240V drives)	page 738
Ultra3000 (230V) drives	page 766

LDL-Series Ironless Linear Motor General Specifications

Weight Specifications - Motor Coil with Flying Leads

Cat. No.	Weight, approx. kg (lb)
LDL-N030120-DHT20	0.63 (1.38)
LDL-T030120-DHT20	0.74 (1.64)
LDL-N030240-*x*HT20	1.14 (2.51)
LDL-T030240-*x*HT20	1.37 (3.02)
LDL-N050120-DHT20	0.75 (1.66)
LDL-T050120-DHT20	0.91 (2.01)
LDL-N050240-*x*HT20	1.39 (3.07)
LDL-T050240-*x*HT20	1.71 (3.77)
LDL-N050360-*x*HT20	2.03 (4.47)
LDL-T050360-*x*HT20	2.50 (5.52)
LDL-N050480-*x*HT20	2.67 (5.88)
LDL-T050480-*x*HT20	3.30 (7.28)
LDL-N075480-*x*HT20	3.32 (7.32)
LDL-T075480-*x*HT20	4.16 (9.18)

Weight Specifications - Motor Coil with Connectors

Cat. No.	Weight, approx. kg (lb)
LDL-N030120-DHT11	0.83 (1.83)
LDL-T030120-DHT11	0.94 (2.07)
LDL-N030240-*x*HT11	1.34 (2.95)
LDL-T030240-*x*HT11	1.57 (3.46)
LDL-N050120-DHT11	0.95 (2.09)
LDL-T050120-DHT11	1.01 (2.22)
LDL-N050240-*x*HT11	1.41 (3.11)
LDL-T050240-*x*HT11	1.91 (4.21)
LDL-N050360-*x*HT11	2.23 (4.92)
LDL-T050360-*x*HT11	2.70 (5.95)
LDL-N050480-*x*HT11	3.50 (7.72)
LDL-T050480-*x*HT11	4.36 (9.61)
LDL-N075480-*x*HT11	3.52 (7.76)
LDL-T075480-*x*HT11	4.36 (9.61)

Weight Specifications - Motor Magnet Channel

Cat. No.	Weight, approx. kg (lb)
LDL-NM030120	1.37 (3.02)
LDL-NM030480	5.51 (12.15)
LDL-TM030120	1.40 (3.08)
LDL-TM030480	5.60 (12.35)
LDL-NM050120	1.87 (4.12)
LDL-NM050480	7.48 (116.49)
LDL-TM050120	1.89 (4.17)
LDL-TM050480	7.57 (16.69)
LDL-NM075120	2.91 (6.42)
LDL-NM075480	11.64 (25.66)
LDL-TM075120	2.94 (6.48)
LDL-TM075480	11.76 (25.93)

Carriage Weight and Heat Sink Area Requirements

Cat. No.	Required Heat Sink Area cm^2 (in.^{2})	Required Carriage Plate Weight kg (lb)
LDL-*x*030120-DHT*xx*	774 (120)	1.4 (3)
LDL-*x*030240-*x*HT*xx*	1160 (180)	2.0 (4.5)
LDL-*x*050120-DHT*xx*	774 (120)	2.7 (6)
LDL-*x*050240-DHT*xx*	1160 (180)	4.0 (9)
LDL-*x*050360-DHT*xx*	1680 (260)	5.9 (13)
LDL-*x*050480-DHT*xx*	2060 (320)	7.3 (16)
LDL-*x*075480-*x*HT*xx*	2060 (320)	7.3 (16)

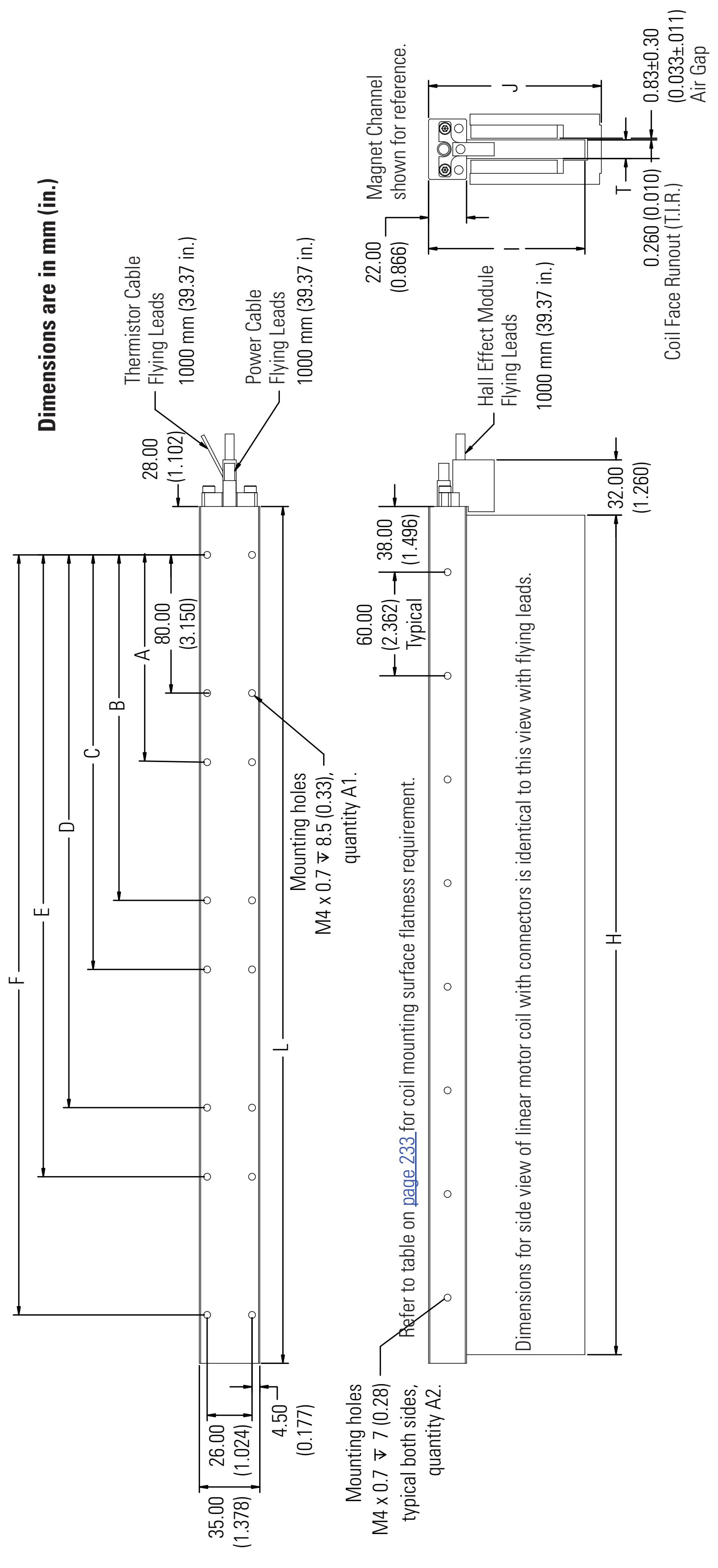
LDL-Series Ironless Linear Motor Component Dimensions

LDL-Series ironless linear motor components are designed to metric dimensions. Inch dimensions are conversions from millimeters. Untoleranced dimensions are for reference.

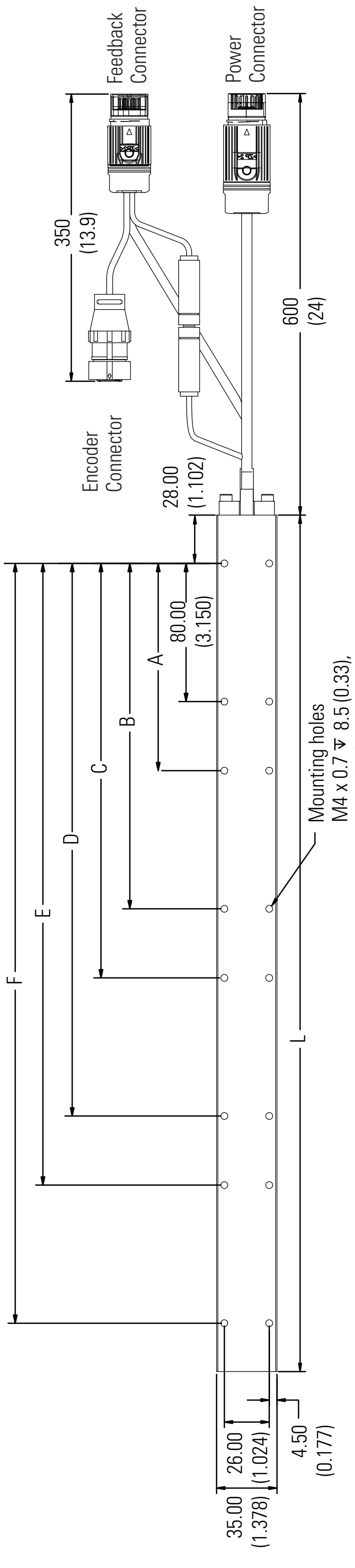
LDL-Series Ironless Linear Motor Coil Dimensions

Dimensions are in mm (in.)

LDL-xxxxxxx-xHT20 Motor Coil Dimensions (flying leads)



LDL-xxxxxxx-xHT11 Motor Coil Dimensions (connectors)



LDL-*xxxx*-*x*HT*xx* Motor Coil Dimensions

Cat. No.	L mm (in.)	A mm (in.)	B mm (in.)	C mm (in.)	D mm (in.)	E mm (in.)	F mm (in.)	G mm (in.)	H mm (in.)	I mm (in.)	J [1] mm (in.)	T mm (in.)	A1 Qty	A2 Qty	Power Cable Gauge mm^2 (AWG)	Flatness mm/300 x 300 (in./12 x 12)
LDL-N030120-DHT*xx*	136.0 (5.35)	–	–	–	–	–	–	60.00 (2.362)	126.00 (4.961)	70.50 (2.776)	80.00 (3.149)	8.30 (0.33)	4	2	0.50 (20)	0.25 (0.010)
LDL-T030120-DHT*xx*												10.80 (0.43)				
LDL-N030240-*x*HT*xx*	256.0 (10.08)	120.00 (4.724)	200.00 (7.874)	–	–	–	–	180.00 (7.087)	246.00 (9.685)			8.30 (0.33)	8	4		
LDL-T030240-*x*HT*xx*												10.80 (0.43)				
LDL-N050120-DHT*xx*	136.0 (5.35)	–	–	–	–	–	–	60.00 (2.362)	126.00 (4.961)	90.50 (3.563)	100.00 (3.937)	8.30 (0.33)	4	2		
LDL-T050120-DHT*xx*												10.80 (0.43)				
LDL-T050240-*x*HT*xx*	256.0 (10.08)	120.0 (4.724)	200 (7.874)	–	–	–	–	180.00 (7.087)	246.00 (9.685)			8.30 (0.33)	8	4		
LDL-N050240-*x*HT*xx*												10.80 (0.43)				
LDL-T050360-*x*HT*xx*	376.0 (14.80)	120.00 (4.724)	200 (7.874)	240.00 (9.449)	320.00 (12.598)	–	–	300.00 (11.811)	366.00 (14.409)			8.30 (0.33)	12	6		0.38 (0.015)
LDL-N050360-*x*HT*xx*												10.80 (0.43)				
LDL-N050480-DHT*xx*	496.0 (19.53)	120.00 (4.724)	200 (7.874)	240.00 (9.449)	320.00 (12.598)	360.00 (14.173)	440.00 (17.323)	420.00 (16.535)	486.00 (19.134)			8.30 (0.33)	16	8	0.75 (18)	0.64 (0.025)
LDL-N050480-EHT*xx*															0.50 (20)	
LDL-T050480-*x*HT*xx*												10.80 (0.43)			0.50 (20)	
LDL-N075480-DHT*xx*										115.50 (4.547)	130.00 (5.118)	8.30 (0.33)			0.75 (18)	
LDL-N075480-EHT*xx*															0.50 (20)	
LDL-T075480-DHT*xx*												10.80 (0.43)			0.75 (18)	
LDL-T075480-EHT*xx*															0.50 (20)	

(1) Tolerance for J dimension is ±0.26 mm (0.010 in.).

Cable Specifications

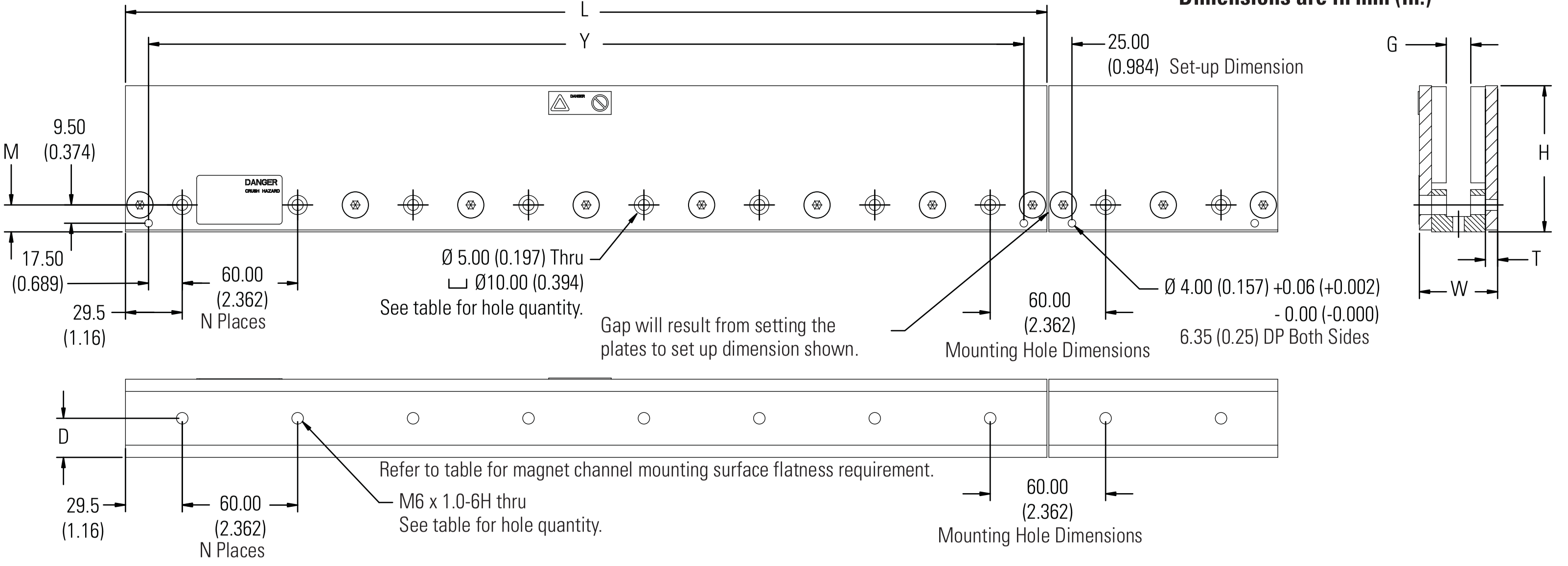
Cable [1]	Conductors	Gauge mm^2 (AWG)	Shield Type	Cable Dia. mm (in.)	Static Bend Radius mm (in.)
Power [2]	4	0.82 (18)	Braid	7.0 (0.28)	18.0 (0.70)
Power [2]	4	0.52 (20)	Braid	6.4 (0.25)	17.0 (0.67)
Thermistor	2	0.20 (26)	None	4.0 (0.16)	10.0 (0.40)
Hall Module	6	0.13 (24)	Foil	5.0 (0.20)	15.0 (0.59)

(1) All cables are non-flex.
(2) Power cable specification is dependent on coil used. See Power Cable Gauge table on this page.

LDL-Series Ironless Linear Motor Magnet Channel Dimensions

LDL-Series Ironless Linear Motor Magnet Channel Dimensions

Dimensions are in mm (in.)



Cat. No.	W mm (in.)	H mm (in.)	T mm (in.)	G [1] mm (in.)	D mm (in.)	L [2] mm (in.)	M mm (in.)	N	Hole Qty	Y [3] mm (in.)	Flatness mm/300 x 300 (in./12 x 12)
LDL-NM030120	37.80 (1.1488)	56.00 (2.205)	6.35 (0250)	9.86 (0.388)	18.90 (0.744)	119.00 (4.685)	14.00 (0.551)	1	2	95.00 (3.740)	0.13 (0.005)
LDL-NM030480						479.00 (18.858)		7	8	455.00 (17.913)	0.26 (0.010)
LDL-TM030120	40.65 (1.600)			12.57 (0.494)	20.33 (0.800)	119.00 (4.685)		1	2	95.00 (3.740)	0.13 (0.005)
LDL-TM030480						479.00 (18.858)		7	8	455.00 (17.913)	0.26 (0.010)
LDL-NM050120	37.80 (1.488)	76.00 (2.992)	6.35 (0.250)	9.86 (0.388)	18.90 (0.744)	119.00 (4.685)	14.00 (0.551)	1	2	95.00 (3.74)	0.13 (0.005)
LDL-NM050480						479.00 18.858)		7	8	455.00 (17.913)	0.26 (0.010)
LDL-TM050120	40.65 (1.600)			12.57 (0.494)	20.33 (0.800)	119.00 (4.685)		1	2	94.00 (3.740)	0.13 (0.005)
LDL-TM050480						479.00 (18.858)		7	8	455.00 (17.913)	0.26 (0.010)
LDL-NM075120	41.1 (1.62)	106.0 (4.173)	8.00 (0.315)	9.86 (0.388)	20.55 (0.809)	119.00 (4.685)	19.00 (0.748)	1	2	95.00 3.740)	0.13 (0.005)
LDL-NM075480						479.00 (18.858)		7	8	455.00 (17.913)	0.26 (0.010)
LDL-TM075120	43.7 (1.72)			12.57 (0.494)	21.85 (0.860)	119.00 (4.685)		1	2	95.00 (3.740)	0.13 (0.005)
LDL-TM075480						479.00 (18.858)		7	8	455.00 (17.913)	0.26 (0.010)

(1) Tolerance for G dimension is +0.35 mm (+0.012 in.), -0.12 mm (-0.004 in.)

(2) Tolerance for L dimension is ±0.25 mm (±0.010 in.)

(3) Tolerance for Y dimension is ±0.05 mm (±0 in.002 in.)

LDL-Series Ironless Linear Motor Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your actuator. For questions regarding product availability, contact your Allen-Bradley distributor.

LDL-Series Ironless Linear Motor Catalog Numbers

LDL - *x* *xxx* *xxx* - *x* *x* T *x* *x*

Cable Termination
0 = Flying leads
1 = Circular DIN connector

Cable Length
0 = 300 mm (12.45 in.)
1 = 600 mm (23.62in.)
2 = 1000 mm (39.37 in.)

Thermal Protection
T = PTC Thermal sensor and switch

Feedback
H = Hall effect (trapezoidal)
N = No feedback

Winding Code
D = D winding
E = E winding

Coil Length (1)
120 = 120 mm (4.72 in.)
240 = 240 mm (9.45 in.)
360 = 360 mm (14.17 in.)
480 = 480 mm (18.90 in.)

Frame Size (1)
030
050
075

Coil Designation
N = Standard coil
T = Thick coil

Motor Series
LDL = LDL-Series

(1) Not all combinations are available. Only the configurations as listed in LDL-Series Ironless Linear Motor Performance Specifications on page 225 are available.

LDL-Series Ironless Linear Motor Magnet Channel Catalog Numbers

LDL - *x* *xxx* *xxx*

Magnet Track Length
120 = 120 mm (4.72 in.)
480 = 480 mm (18.90 in.)

Frame Size
030
050
075

Coil Designation
NM = Standard coil
TM = Thick coil

Motor Series
LDL = LDL-Series

Notes:

Controller Platforms

Use this chapter to become familiar with Rockwell Automation Integrated Architecture systems featuring programmable automation controllers used for Kinetix Integrated Motion systems. It also includes information on programmable logic controllers that are used for simpler, low-cost component-level motion control.

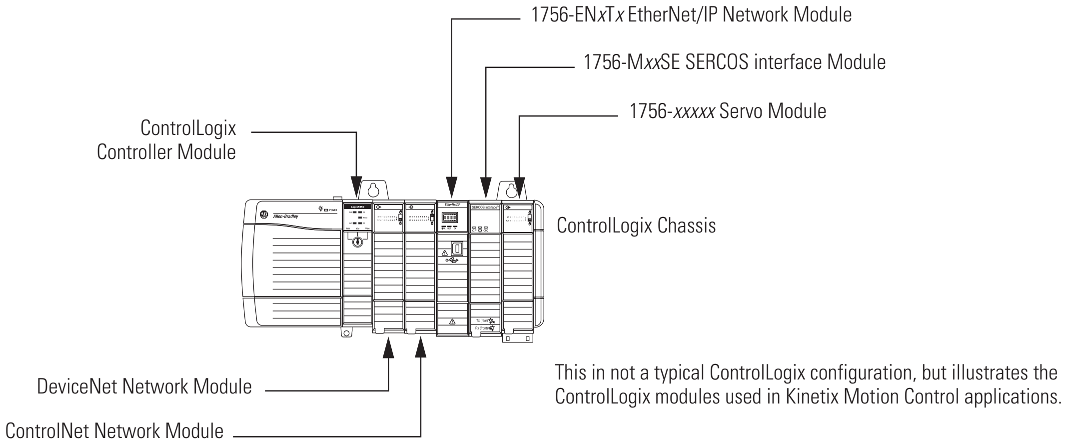
ControlLogix System Overview

The ControlLogix programmable automation controller is a modular system capable of handling your most intensive applications. Modules are inserted into slots on the ControlLogix chassis. The ControlLogix modules available for your motion control application include:

- 1756-EN2T, 1756-EN2F, 1756-EN2TR, and 1756-EN3TR EtherNet/IP network modules.
- 1756-M03SE, 1756-M08SE, and 1756-M16SE SERCOS interface modules.
- 1756-M02AE, 1756-HYD02, and 1756-M02AS analog servo modules.

For more information regarding the ControlLogix platform, including the 1756-EN*x*T*x* EtherNet/IP modules, refer to the ControlLogix Selection Guide, publication 1756-SG001.

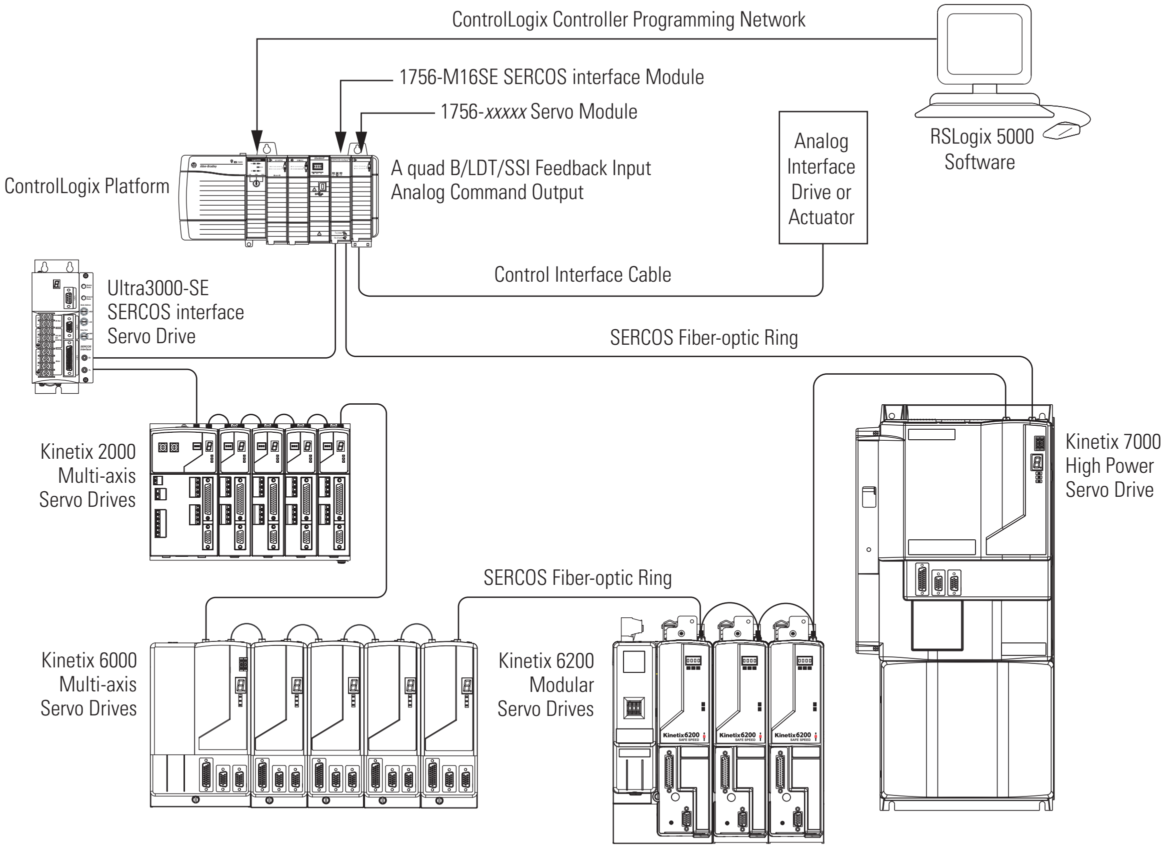
ControlLogix Platform



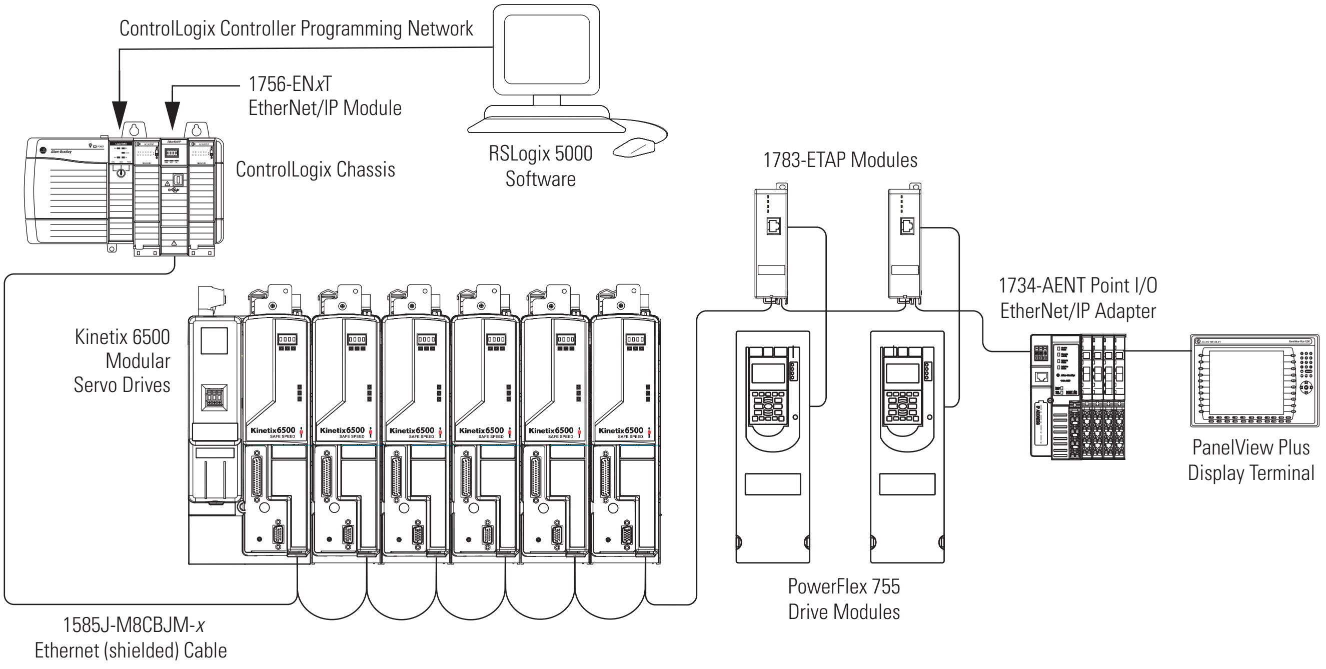
This in not a typical ControlLogix configuration, but illustrates the ControlLogix modules used in Kinetix Motion Control applications.

Select the network module and number of controllers based on the needs of your application. This flexibility lets you share system resources and divide applications across multiple controllers.

ControlLogix System (SERCOS)



ControlLogix System (Integrated Architecture on EtherNet/IP)



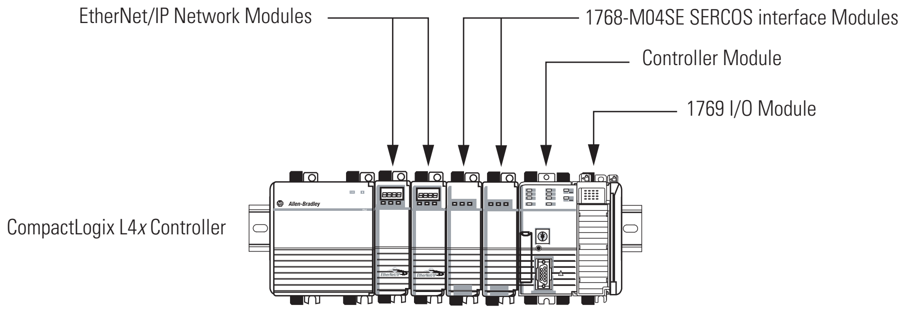
CompactLogix System Overview

The CompactLogix programmable automation controller is a modular system that provides cost-effective control for smaller applications. Modules snap together side-by-side on a DIN rail. The CompactLogix products available for your motion control application include:

- 1768-L4*x* controller with 1768-M04SE SERCOS interface module.
- 1769-L23*x* controller with an EtherNet/IP network.

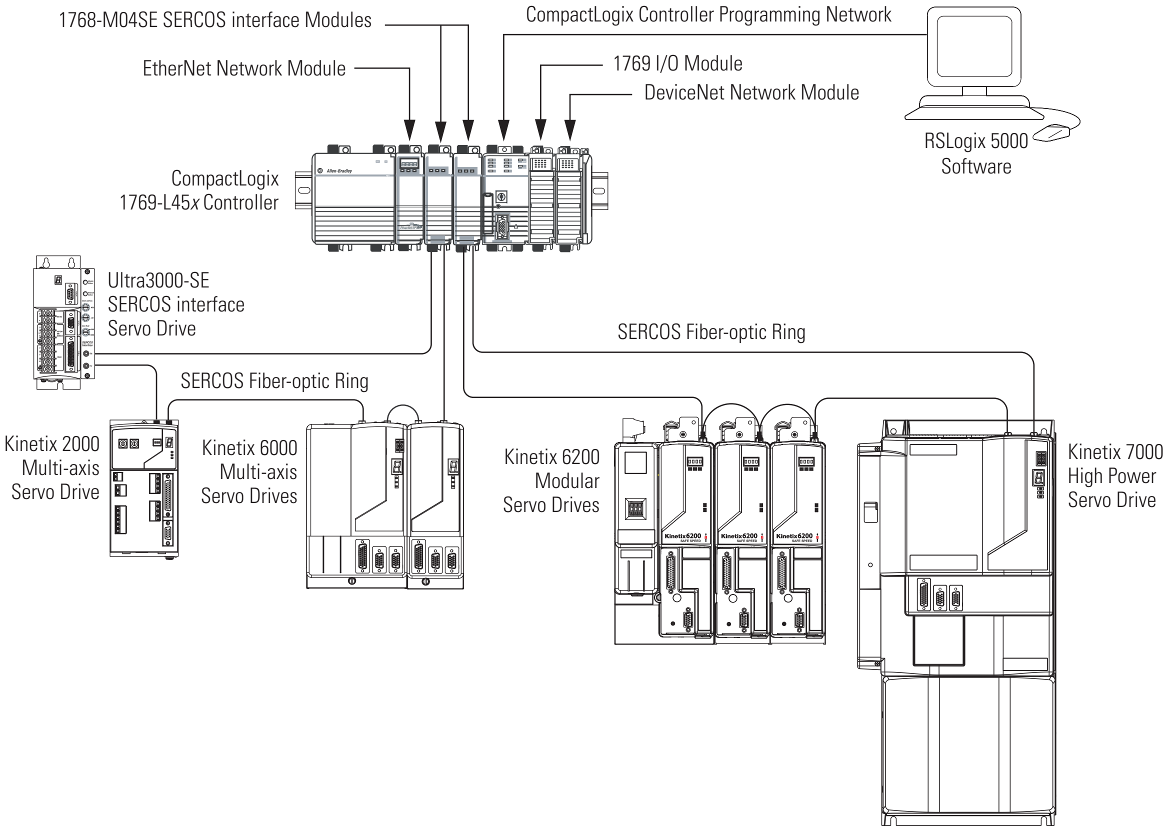
For more information regarding the CompactLogix platform, refer to the CompactLogix Selection Guide, publication 1769-SG001.

CompactLogix Platform

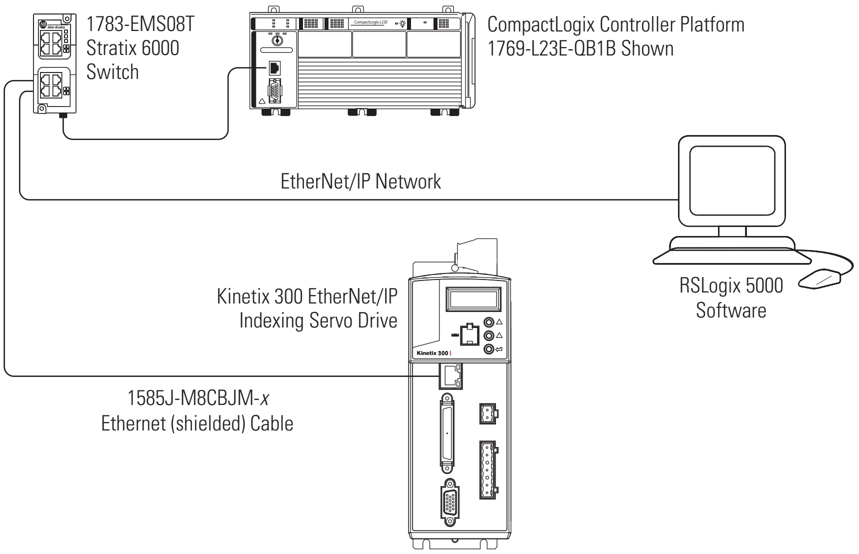


This in not a typical CompactLogix configuration, but illustrates the CompactLogix modules used in Kinetix Motion Control applications.

CompactLogix System (SERCOS)



CompactLogix System (EtherNet/IP)

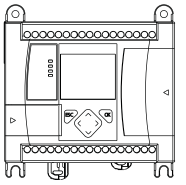


MicroLogix System Overview

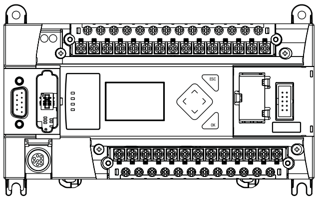
The MicroLogix programmable logic controllers with Modbus-RTU or PTO signals provide simple PLC-based motion solutions with the Kinetix 3 component servo drive. The MicroLogix controller products available for your motion control application include these controller families.

MicroLogix Controller Families

Controller Family	Cat. No.
MicroLogix 1100	1763-L16*xxx*
MicroLogix 1400	1766-L32*xxx*

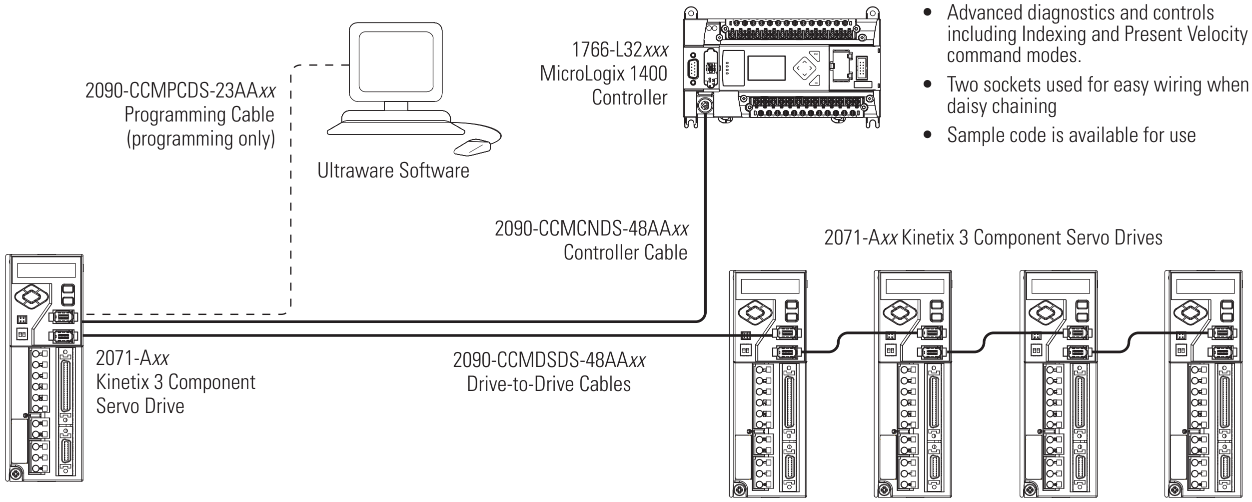


MicroLogix 1100



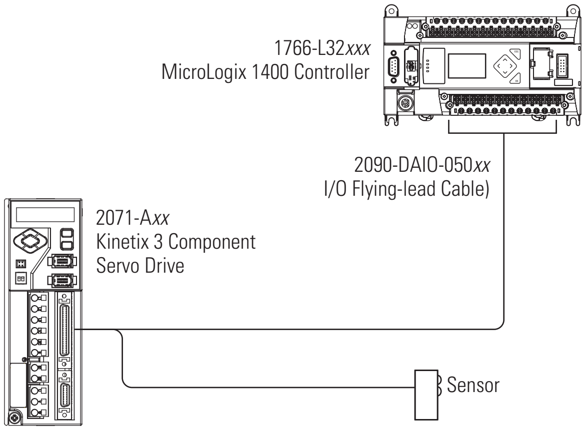
MicroLogix 1400

MicroLogix System (drive-to-drive with RS422 multi-drop network connections)



- Advanced diagnostics and controls including Indexing and Present Velocity command modes.
- Two sockets used for easy wiring when daisy chaining
- Sample code is available for use

MicroLogix System (analog or PTO control)



- Simple and cost effective
- Use of a controller is not required
- Trigger can be directly from sensors for Indexing mode or Present Velocity command mode
- Supports analog control and pulse train (PTO) control

For more information regarding the MicroLogix platform, refer to the MicroLogix Programmable Controllers Selection Guide, publication 1761-SG001.

SERCOS interface Modules

The SERCOS interface modules provide a fiber-optic link between the Logix platforms and servo drives. The communication link between the module and the drive is via IEC/EN-61491 SErial Real-time COmmunication System (SERCOS) using a fiber-optic medium. This fiber-optic medium ensures reliable high-speed data transmission with excellent noise immunity, improved communication speed, and simplified interconnect wiring between the drive and motion module.

ControlLogix
1756-M03SE, 1756-M08SE and
1756-M16SE
SERCOS interface Module

CompactLogix
1768-M04SE
SERCOS interface Module

SERCOS interface Module Features

- Drive, motor setup, and configuration by using RSLogix 5000 software.
- Real-time optical serial interface from the ControlLogix controller to Kinetix 2000, Kinetix 6200, Kinetix 6000, Kinetix 7000, and Ultra3000-SE drives for parameter updates.
- Support for up to 16 servo drive axes for greater application flexibility.
- Support for up to 32 m (105 ft) of plastic fiber-optic cable and 200 m (657 ft) of glass fiber-optic cable for distributed, convenient drive support and an overall reduction of wiring.
- Support for high-resolution motor position feedback for superior performance.
- Support for single-turn and multi-turn absolute feedback for elimination of costly downtime and homing as a result of power outages.

Certifications

- ControlLogix and CompactLogix SERCOS interface modules are UL Listed for U.S. and Canada. ControlLogix SERCOS interface combination module is UL Listed and CSA Certified.
  Refer to http://www.ab.com for more information.
- CE marked for all applicable directives.

SERCOS interface Module Specifications

This section contains general, communication, and environmental specifications for the ControlLogix and CompactLogix SERCOS interface modules.

General Specifications

Attribute	1756-M03SE	1756-M08SE	1756-M16SE	1768-M04SE
Number of nodes	3 axes max	8 axes max	16 axes max	4 axes max 2 additional feedback axes.
Module location	1756 ControlLogix chassis			1768 CompactLogix DIN rail mounted
Module keying	Electronic			
Power dissipation	5 W			5.04 W
Backplane current	• 760 mA @ 5.1V DC • 2.5 mA @ 24V DC			969 mA @ 5.2V DC
Fiber-optic cable	Refer to Chapter 12 for fiber-optic cable descriptions, dimensions, and specifications.			

Number of Axes and Data Rate

Description	Logix SERCOS Module	Number of Axes	Data Rate
Logix module specifications	1756-M03SE	Up to 3	4 or 8 Mbps
	1756-M08SE	Up to 8	
	1756-M16SE	Up to 16	
	1768-M04SE	Up to 4	

SERCOS interface Cycle Time

Description	Data Rate (1)	Number of Axes	Cycle Time (2)
SERCOS interface cycle time	4 Mbps	Up to 2	0.5 ms
		Up to 4	1 ms
		Up to 8	2 ms
		No support for 9…16 axes	
	8 Mbps	Up to 4	0.5 ms
		Up to 8	1 ms
		Up to 16	2 ms

(1) Software selectable.

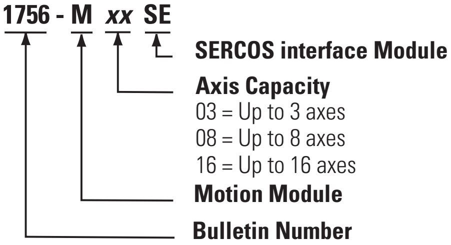
(2) Only Kinetix 2000, Kinetix 6200, Kinetix 6000, and Kinetix 7000 drives support the 0.5 ms cycle time.

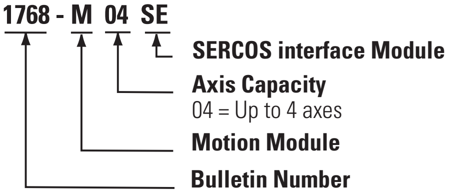
Environmental Specifications

Attribute	1756-M03SE	1756-M08SE	1756-M16SE	1768-M04SE
Temperature, ambient Operating Storage	 0…60 °C (32…140 °F) -40…85 °C (-40…185 °F)			
Relative humidity	5…95% noncondensing			
Vibration	2 g @ 10…500 Hz			5 g @ 10…500 Hz
Shock (operating)	30 g			
Shock (nonoperating)	50 g			

SERCOS interface Module Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering chart below to understand the configuration of your interface module. For questions regarding product availability, contact your Allen-Bradley distributor.





Analog Servo Modules

The ControlLogix family of analog servo modules is a cost effective option for closed or open-loop motion control devices that support an analog interface. The analog servo modules provide a ±10V analog output command reference and support a variety of different position feedback devices. As many as two axes can be controlled per module, and multiple modules can be used to provide as many as 32 axes of control per ControlLogix controller.

ControlLogix
1756-M02AE
1756-HYD02
1756-M02AS
Analog Servo Module

Analog Servo Module Features

Module	Features
1756-M02AE	The 1756-M02AE and 1756-PM02AE modules are two-axis servo modules/PCI cards optimized for control of drives or actuators that require an ±10V velocity or torque reference input. Both provide a quadrature position feedback input and are compatible with a wide range of quadrature output rotary and linear transducers.
1756-HYD02	The 1756-HYD02 module is a two-axis servo module optimized for control of hydraulic actuators that require an ±10V velocity reference input. The 1756-HYD02 module provides a LDT feedback input. Typical actuators include hydraulic motors and hydraulic cylinders. The 1756-HYD02 module is compatible with a wide range of magnostrictive linear transducers (LDT) feedback devices. Compatible LDTs include the following: • Temposonics II: RPM or DPM • Balluff: BTL-2-L2 or BTL-2-M2 • Santest: GYRP or GYRG • Gemco Quick-Stick II: 951 VP or 951 RS
1756-M02AS	The 1756-M02AS module is a two-axis servo module optimized for control of drives/actuators that require an ±10V velocity or torque reference input. The 1756-M02AS module provides a Serial Synchronous Input (SSI) position feedback input and is compatible with a wide range of SSI output rotary and linear transducers. SSI devices are available in many versions. • Linear absolute and incremental encoders • Rotary absolute and incremental encoders • Linear absolute glass scales • Linear magnostrictive • Linear laser distance

Certifications

- ControlLogix modules are UL Listed and CSA Certified. Refer to http://www.ab.com for more information.
- CE marked for all applicable directives.

IMPORTANT The servo module must be in the same chassis as the ControlLogix controller that controls the servo module. If you distribute motion control across different locations, place a ControlLogix controller in each chassis that has a servo module.

Analog Servo Module Specifications

This section contains general, input, output, servo loop, and environmental specifications for the ControlLogix analog servo modules.

General Specifications

Attribute	1756-M02AE	1756-HYD02	1756-M02AS
Number of axes per controller	32 axes max (that is, 16 cards controlled by 1 ControlLogix controller)		
Number of axes per module	2 axes max		
Module location	1756 ControlLogix chassis		
Module keying	Electronic		
Power dissipation	5.5 W max		
Backplane current	• 700 mA @ 5.1V DC • 2.5 mA@ 24V DC		

Input Specifications

Attribute	1756-M02AE	1756-HYD02	1756-M02AS
Encoder input Type	Incremental AB quadrature with marker	Linear displacement transducer	Synchronous serial interface
Number of inputs	2 per module		
Electrical interface	Optically isolated 5V differential		
Registration inputs Type	Optically isolated, current sinking input		
Number of inputs	2 per module		
24V input voltage	+24V DC, nom		
5V input voltage	+5V DC, nom		
Response time (position latched)	1 µs	1 servo update period (250…2000 µs)	
Drive faults and home inputs Type	Optically isolated, current sinking input		
Number of inputs	2 per module		
Input voltage	+24V DC, nom		

Output Specifications

Attribute	1756-M02AE	1756-HYD02	1756-M02AS
Servo output			
Type	Analog voltage		
Number of inputs	2 per module		
Voltage range	±10V		
Voltage resolution	16 bits		
Drive enable			
Type	Solid-state isolated relay contacts		
Number of inputs	2 per module		
Operating voltage	+24V DC, nom (Class 2 source)		

Servo Loop Specifications

Attribute	1756-M02AE	1756-HYD02	1756-M02AS
Servo loop type			
External drive = torque	Position Loop: PID with Velocity Feedforward. Velocity Loop: PI with Accel Feedforward (nested) with directional Scaling and Friction Compensation.	N/A	Position Loop: PID with Velocity Feedforward. Velocity Loop: PI with Accel Feedforward (nested) with directional Scaling and Friction Compensation.
External drive = velocity or hydraulic	Position Loop: PID with Velocity Feedforward and Accel Feedforward with directional Scaling and Friction Compensation. Velocity Loop: N/A (handled by drive or valve).	Proportional, Integral, and Differential (PID) with feed-forwards and directional scale.	Position Loop: PID with Velocity Feedforward and Accel Feedforward with directional Scaling and Friction Compensation. Velocity Loop: N/A (handled by drive or valve).

Environmental Specifications

Attribute	1756-M02AE	1756-HYD02	1756-M02AS
Temperature, ambient			
Operating	0…60 °C (32…140 °F)		
Storage	-40…85 °C (-40…185 °F)		
Relative humidity	5…95% noncondensing		
Vibration	2 g @ 10…500 Hz		
Shock (operating)	30 g		
Shock (nonoperating)	50 g		

Analog Servo Module Catalog Number

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering chart below to understand the configuration of your servo module. For questions regarding product availability, contact your Allen-Bradley distributor.

1756 - *xx* 02 *xx*

Encoder Type
AE = Analog encoder
AS = Synchronous serial interface

Axis Capacity
02 = Up to 2 axes

Module Type
M = Motion servo
HYD = Hydraulic servo

Bulletin Number

Analog Servo Module Terminal Block

One 1756-TB*xx*H servo module terminal block is required for each 1756-M02AE, 1756-M02AS, or 1756-HYD02 module. The terminal block provides wire terminations so that performance can be controlled. It is removable and can be installed by opening the servo module door and sliding it into place.

Terminal Block Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering chart below to understand the configuration of your terminal block. For questions regarding product availability, contact your Allen-Bradley distributor.

1756 - TB *xx*H

Options
CH = 36-position cage clamp removable terminal board.
S6H = 36-position spring clamp removable terminal board.

Terminal Block

Bulletin Number

Kinetix 6200 and Kinetix 6500 Modular Multi-axis Servo Drives

The Kinetix 6500 servo drives provide Integrated Motion on the EtherNet/IP network by using CIP Motion and CIP Sync technology from ODVA, all built on the Common Industrial Protocol (CIP).

The Kinetix 6200 servo drives provide Integrated Motion capability through SERCOS interface and compatibility with Kinetix 6000 drives, letting you migrate to the enhanced features exactly when and where you need them.

The Kinetix 6200 and Kinetix 6500 drive families are part of the Kinetix Integrated Motion solution.

These modular safe-speed servo drives help increase productivity and protect personnel with embedded safety features. Modular design and control provides ease of maintenance and greater flexibility as the drive easily transitions from safe torque-off to safe speed.

Kinetix 6200 and Kinetix 6500 Servo Drive Components

Kinetix 6200 and Kinetix 6500 modular servo drive systems consist of these required components:

- One integrated axis power module (IAM or leader IAM), 2094-AC*xx*-M*xx*-M (230V) or 2094-BC*xx*-M*xx*-M (460V)
- Up to seven axis power modules (AM), 2094-AM*xx*-S (230V) or 2094-BM*xx*-M (460V)
- Up to eight control modules, 2094-SE02F-M00-S*x* (SERCOS interface) or 2094-EN02D-M01-S*x* (EtherNet/IP network)
- One power rail, 2094-PRS1, 2094-PRS2, 2094-PRS3, 2094-PRS4, 2094-PRS5, 2094-PRS6, 2094-PRS7, or 2094-PRS8
- One to eight MP-Series, RDD-Series, or LDC-Series rotary/linear motors or linear actuators
- One to eight motor power and feedback cables
- Two to nine SERCOS fiber-optic cables

- Ethernet cables for Logix control and programming the safety configuration

Kinetix 6200 and Kinetix 6500 systems may also include any of these optional components:

- One or more IAM power modules used as a follower IAM, 2094-AC*xx*-M*xx*-M (230V) or 2094-BC*xx*-M*xx*-M (460V) and associated axis modules, power rails, motors, and cables as required for the application.
- One shunt module, 2094-BSP2 with optional Bulletin 1394 external passive shunt module
- Slot-filler modules, 2094-PRF
- Bulletin 2094 Line Interface Module (LIM)
- Bulletin 2090 Resistive Brake Module (RBM)

Kinetix 6000 Drive Component Compatibility

The 2094-BC*xx*-M*xx*-M and 2094-BM*xx*-M power modules contain the same power structure as the 2094-BC*xx*-M*xx*-S and 2094-BM*xx*-S drive modules. Because of this, the 2094-BSP2 shunt module, 2094-PRF slot-filler module, and 2094-PRS*x* power rails are supported by both drive families.

In addition, 2094-BM*xx*-M AM power modules with SERCOS interface are supported on power rails with a 2094-BC*xx*-M*xx*-S IAM drive module. Conversely, 2094-BM*xx*-S AM drive modules are supported on power rails with a 2094-BC*xx*-M*xx*-M IAM power module with SERCOS interface.

IMPORTANT	Kinetix 6500 EtherNet/IP control modules (catalog numbers 2094-EN02D-M01-S*x*) are not compatible with IAM/AM modules on the same Bulletin 2094 power rail where SERCOS interface is used.

IAM/AM Module Compatibility

IAM Module	Control Module	2094-*x*M*xx*-S Kinetix 6000 AM Module	2094-BM*xx*-M AM Power Modules	
			2094-SE02F-M00-S*x* Kinetix 6200 Control Module	2094-EN02D-M01-S*x* Kinetix 6500 Control Module
2094-*x*C*xx*-M*xx*-S	N/A	Fully compatible	Fully compatible	Not compatible
2094-BC*xx*-M*xx*-M (IAM power module)	2094-SE02F-M00-S*x* SERCOS interface			
	2094-EN02D-M01-S*x* EtherNet/IP network	Not compatible	Not compatible	Fully compatible

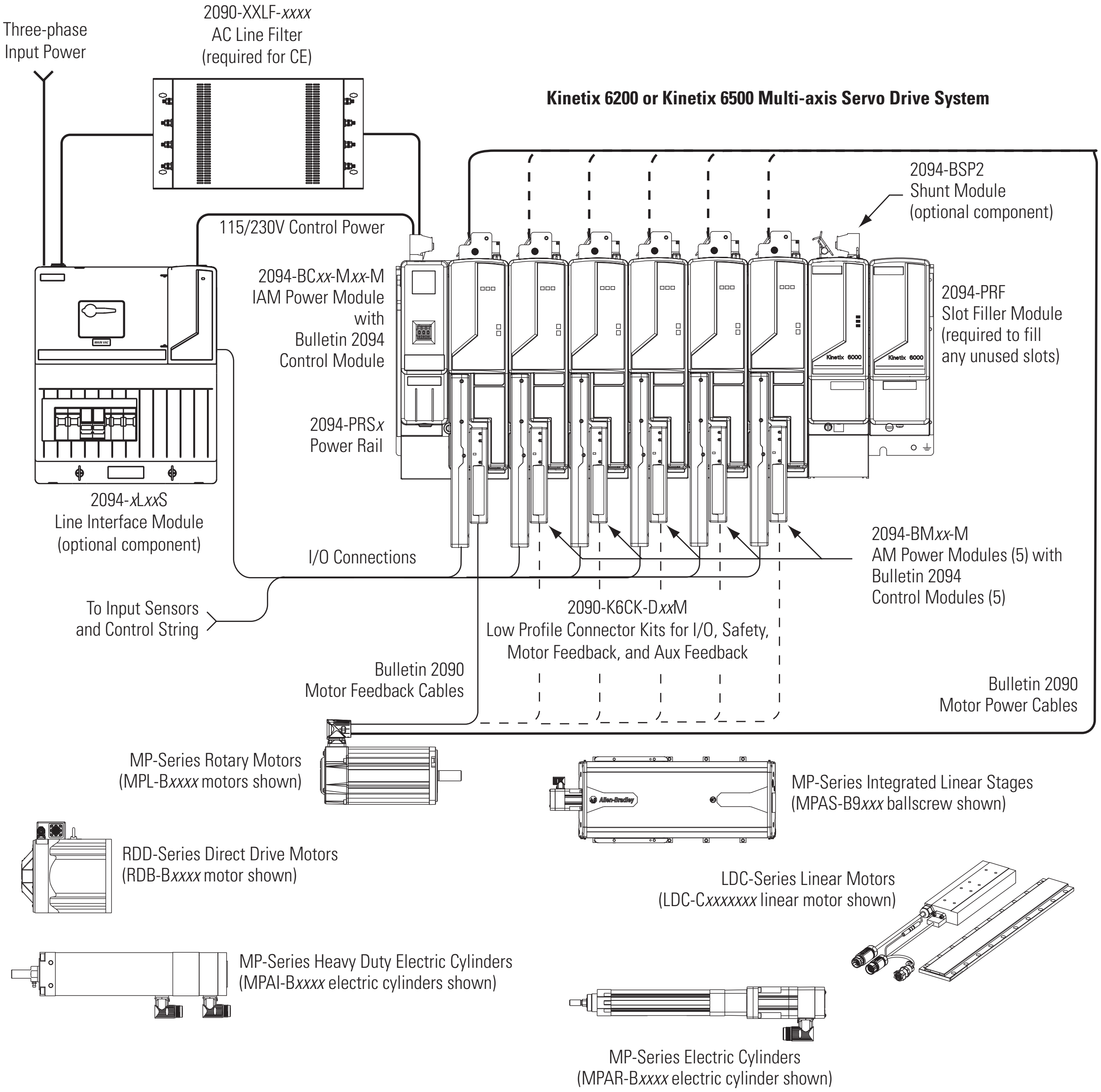
For more information on the Kinetix 6000 IAM and AM modules, catalog numbers 2094-*x*C*xx*-M*xx*-S and 2094-*x*M*xx*-S, refer to Kinetix 6000 Multi-axis Servo Drives in Chapter 5.

For more information on the Bulletin 2094 power rails, shunt module, slot-filler module, RBM and LIM modules, refer to Motion Control Accessories in Chapter 12.

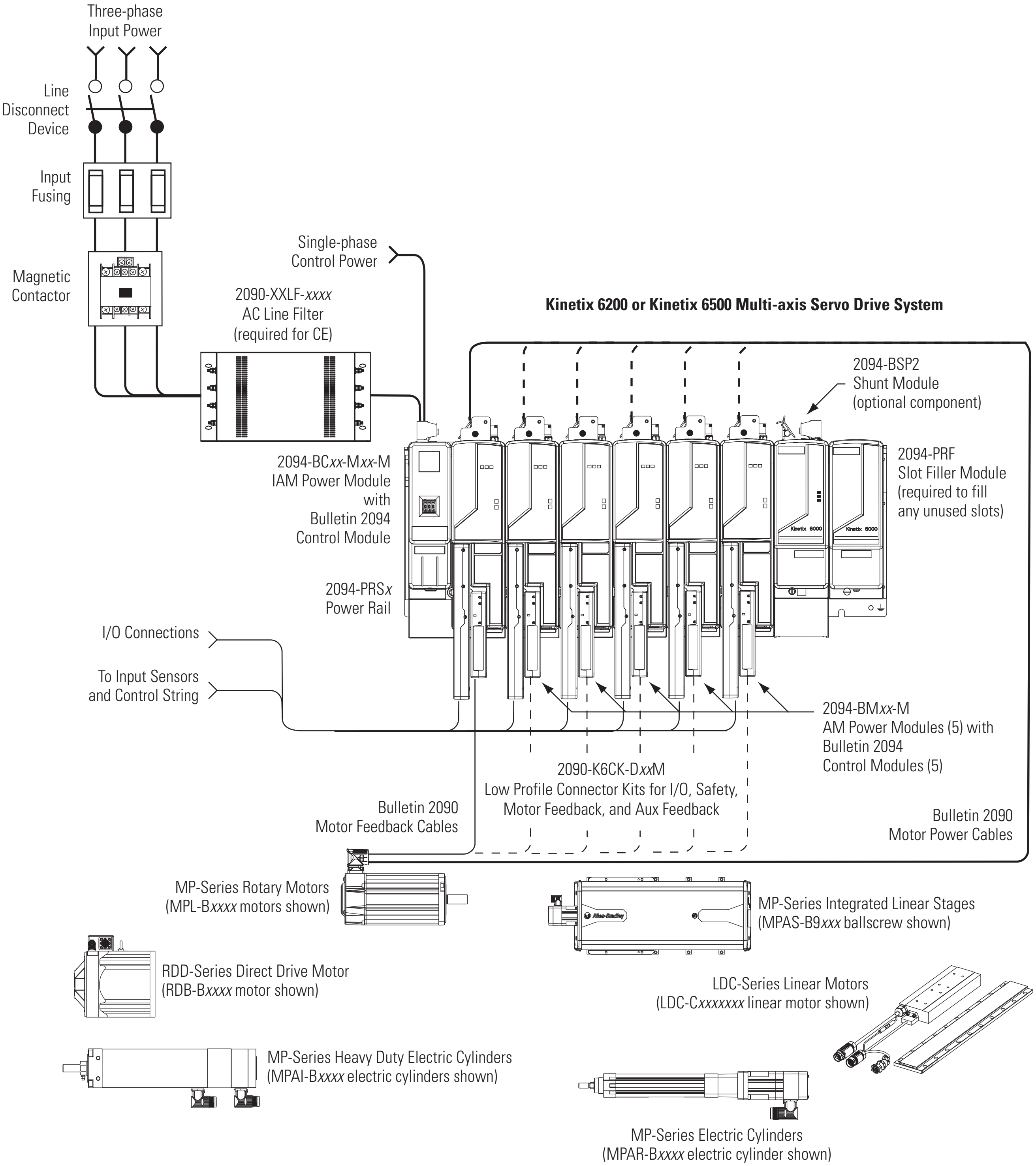
Typical Hardware Configurations

These are typical hardware configurations for Kinetix 6200 and Kinetix 6500 modular drive systems.

Modular Drive System (with LIM module)



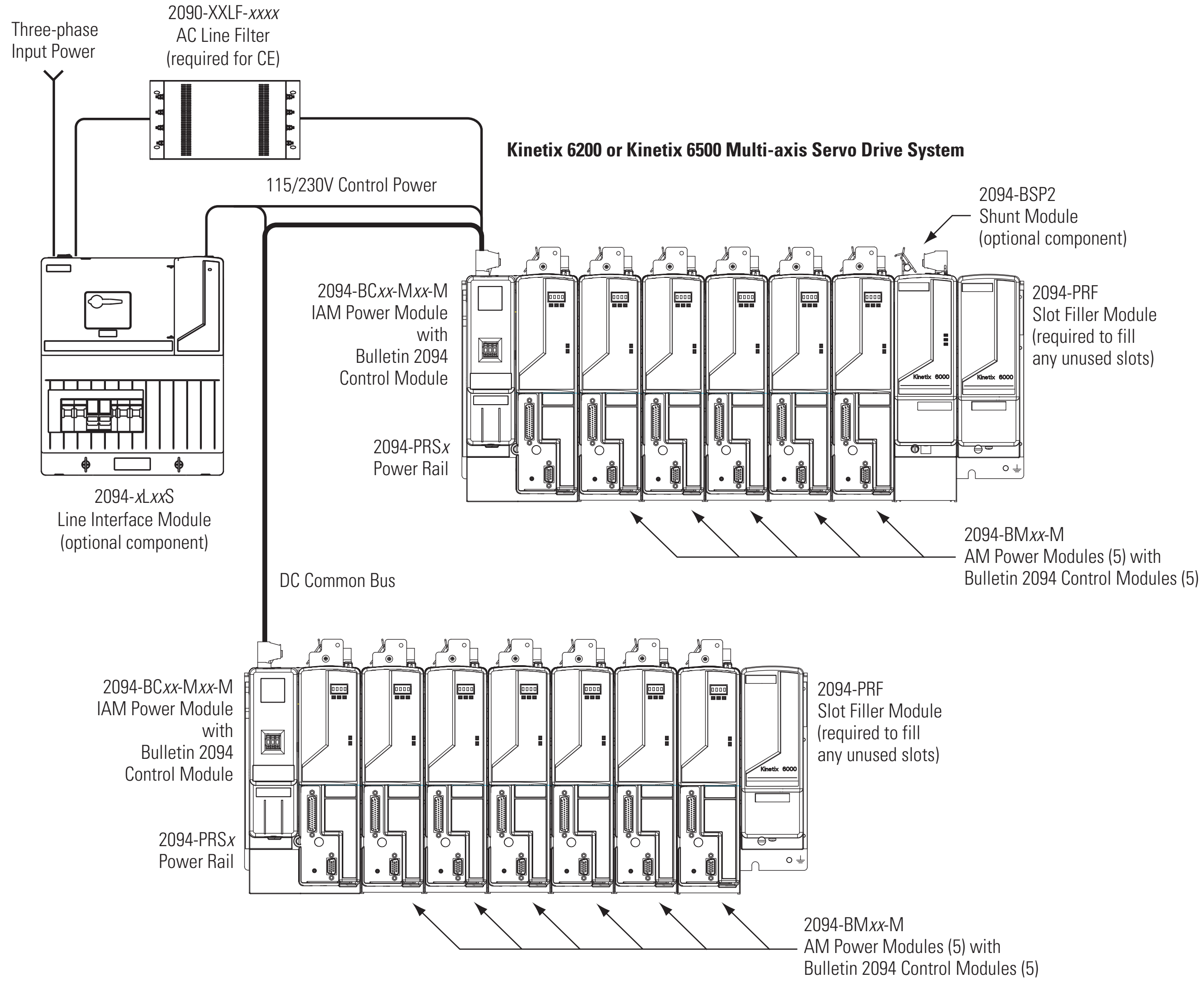
Modular Drive System (without LIM module)



In the system configuration below, the leader IAM power module is connected to the follower IAM module via the DC common bus. When planning your panel layout, you must calculate the total bus capacitance of your DC common bus system to make sure that the leader IAM power module is sized sufficiently to pre-charge the entire system. Refer to the Kinetix 6200 and Kinetix 6500 Modular Servo Drive User Manual, publication 2094-UM002, when making this calculation.

IMPORTANT	If total bus capacitance of your system exceeds the leader IAM power module pre-charge rating, the IAM module four-character display scrolls a power cycle user limit condition. If input power is applied, the display scrolls a power cycle fault limit condition. To correct this condition, you must replace the leader IAM power module with a larger module or decrease the total bus capacitance by removing AM power modules.

Modular Drive System (DC common bus)



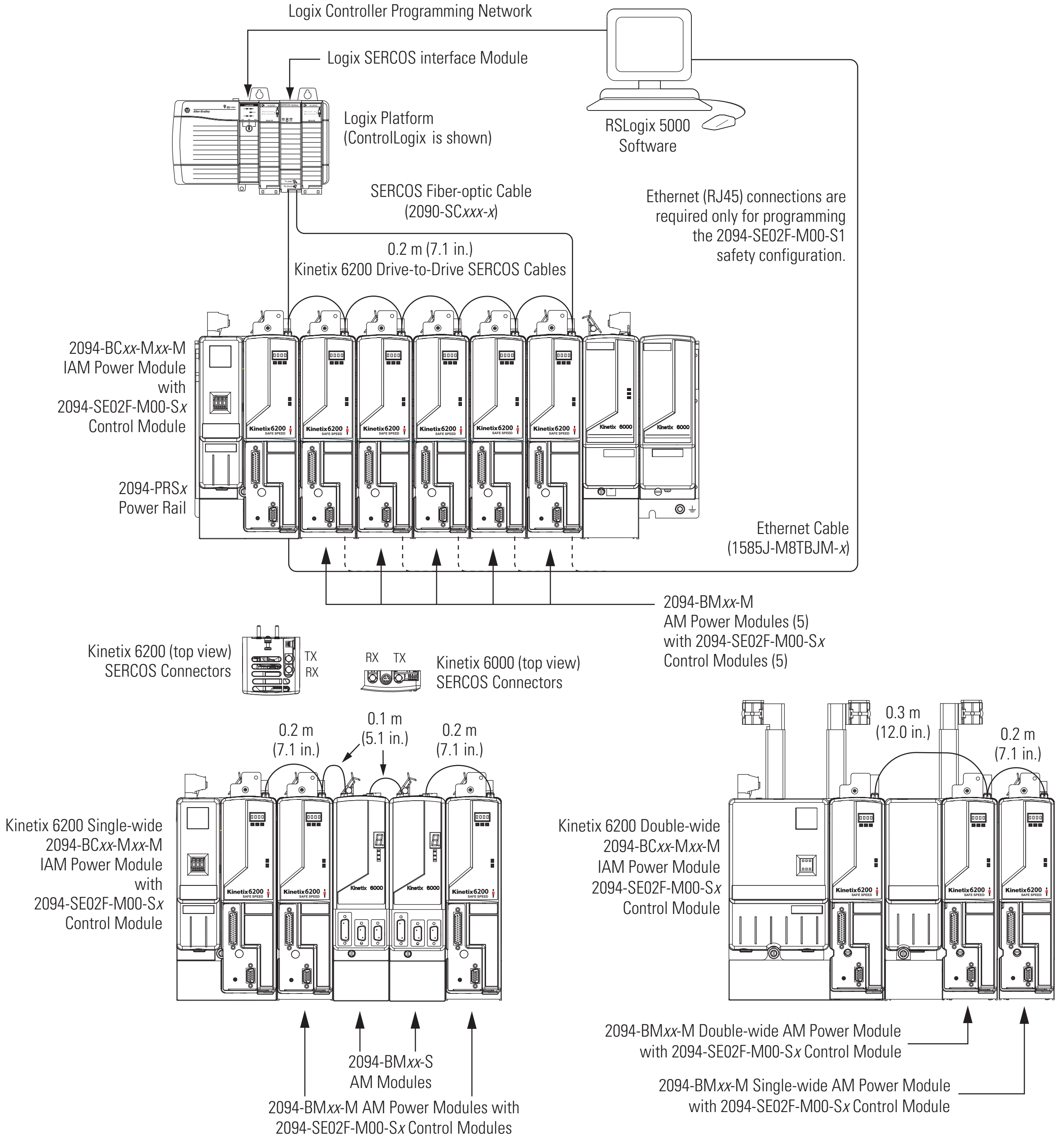
Motors and other details common to both three-phase AC and DC common-bus configurations are removed.

Typical Communication Configurations

The Kinetix 6200 control modules use SERCOS interface for configuring the Logix module and EtherNet/IP network for access to the safety configuration tool.

In this example, an ethernet cable is connected to each control module when programming the safety configuration. EtherNet/IP network connectivity is not required during runtime. Also shown are drive-to-drive SERCOS cable lengths and catalog numbers when Kinetix 6000 and Kinetix 6200 drive modules exist on the same power rail.

Kinetix 6200 Drive Communication (SERCOS)



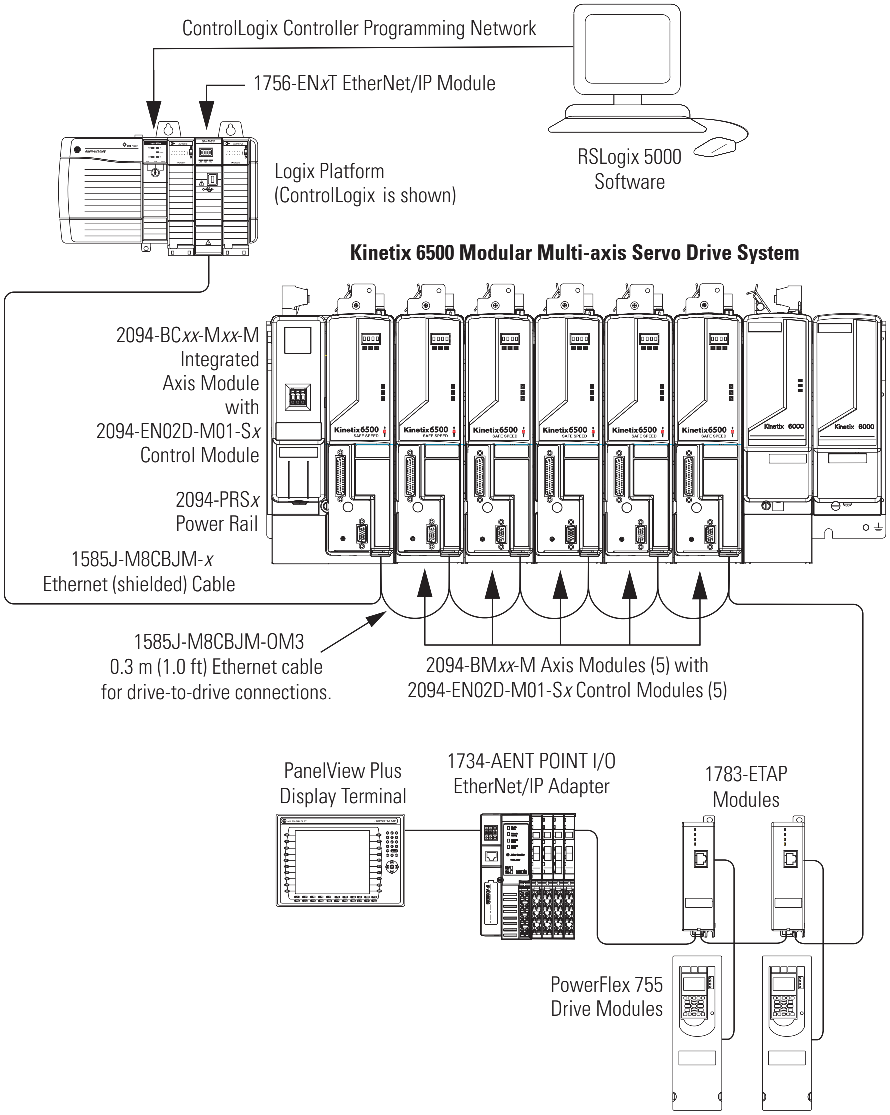
The Kinetix 6500 control modules can use any Ethernet topology including star, linear, and device-level ring (DLR). DLR is an ODVA standard and provides fault tolerant connectivity.

IMPORTANT	Shielded Ethernet cable, catalog number 1585J-M8CBJM-*x*, is available in lengths up to 78 m (256 ft). However, the total length of Ethernet cable connecting drive-to-drive, drive-to-controller, or drive-to-switch must not exceed 100 m (328 ft).

In this example, all devices are connected in linear topology. The Kinetix 6500 control module includes dual-port connectivity. Devices without dual ports should include the 1783-ETAP module or be connected at the end of the line.

- Linear configurations support up to 64 devices.
- No redundancy. If any device becomes disconnected, all the devices downstream loose communication.

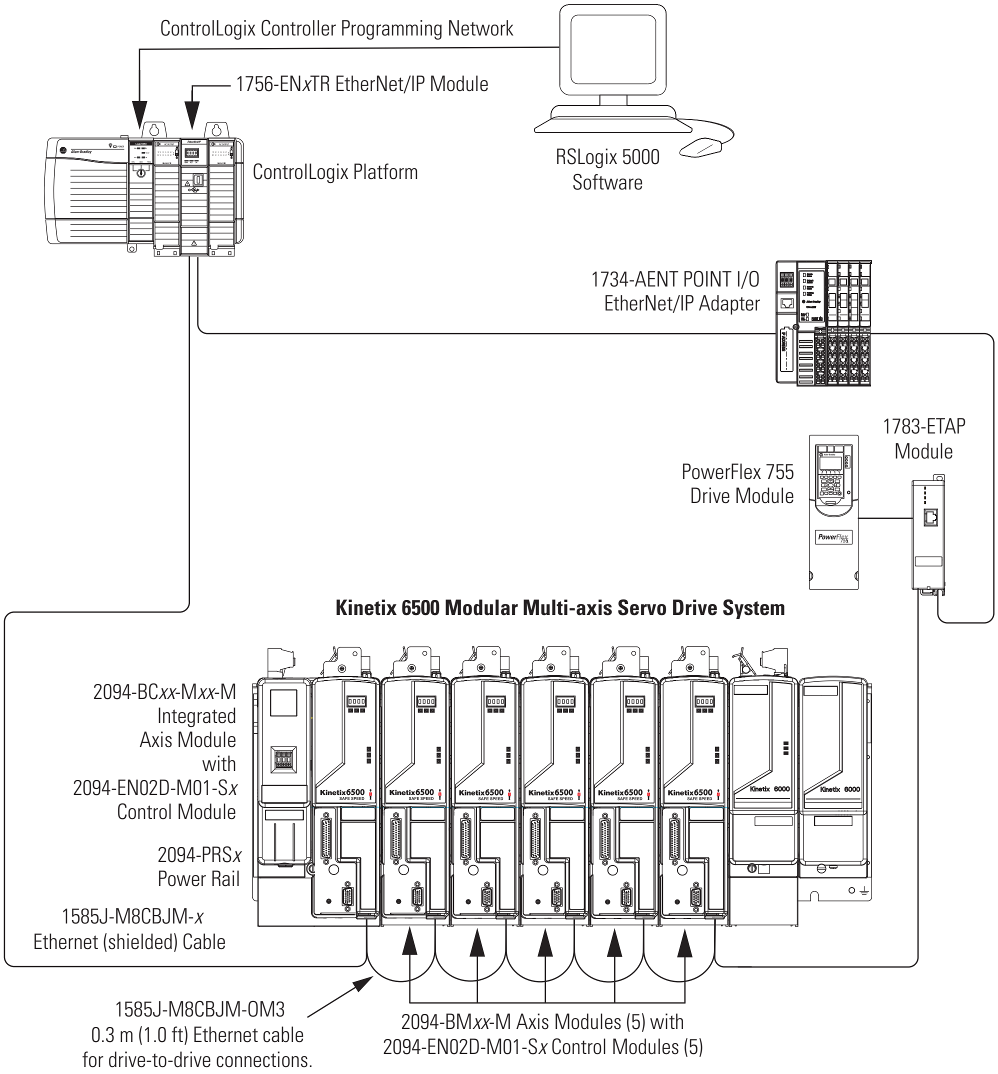
Kinetix 6500 Linear Communication (EtherNet/IP network)



In this example, the devices are connected by using device-level ring (DLR) topology. DLR topology is fault tolerant. For example, if a device in the ring is disconnected, the rest of the devices in the ring continue to maintain communication.

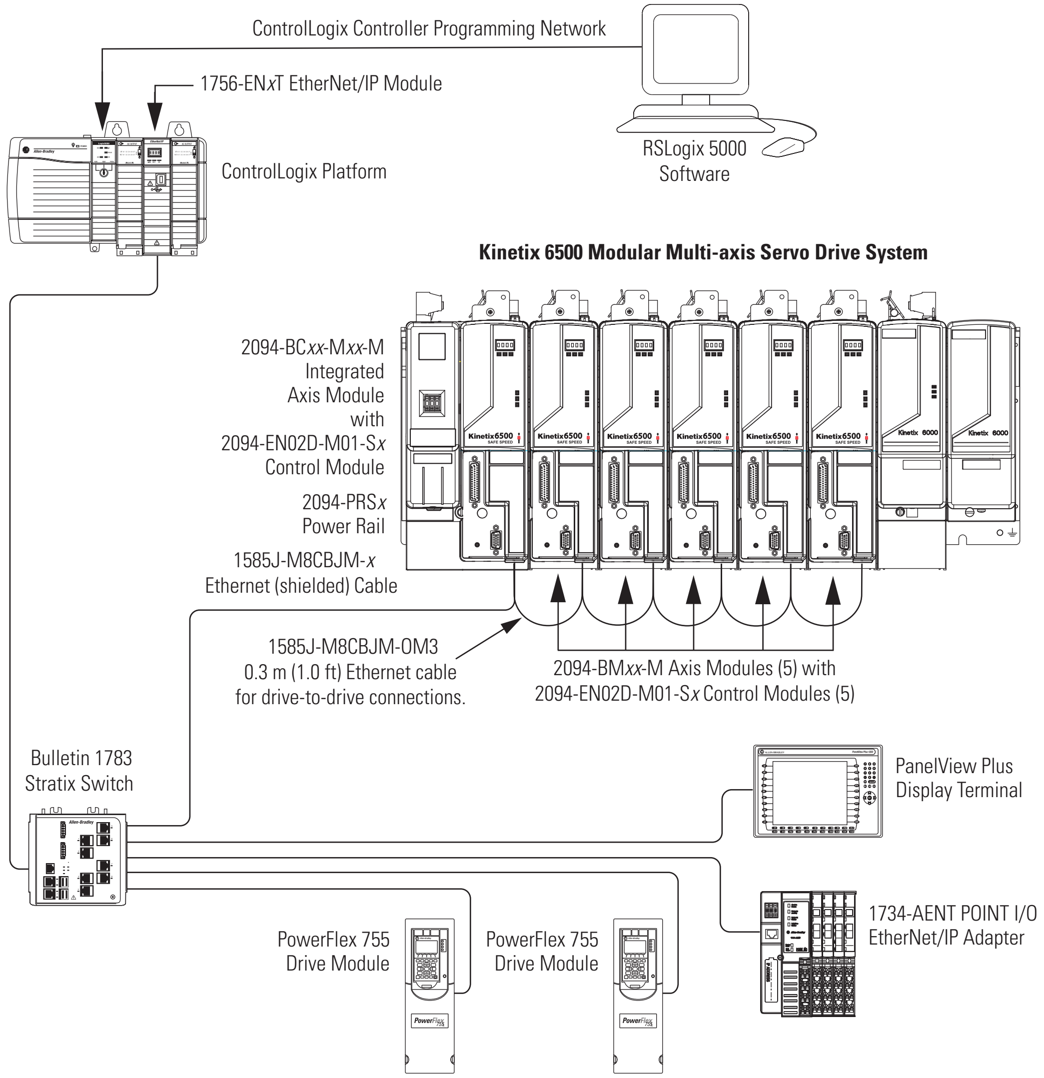
- DLR configurations support up to 64 devices.
- All devices in a DLR ring should have dual-port connectivity or be connected in the ring by using a 1783-ETAP module.

Kinetix 6500 Ring Communication (EtherNet/IP network)



In this example, the devices are connected by using star topology. Each device is connected directly to the switch, making this topology fault tolerant. The 2094 power rail modules and other devices operate independently. The loss of one device does not impact the operation of the other devices.

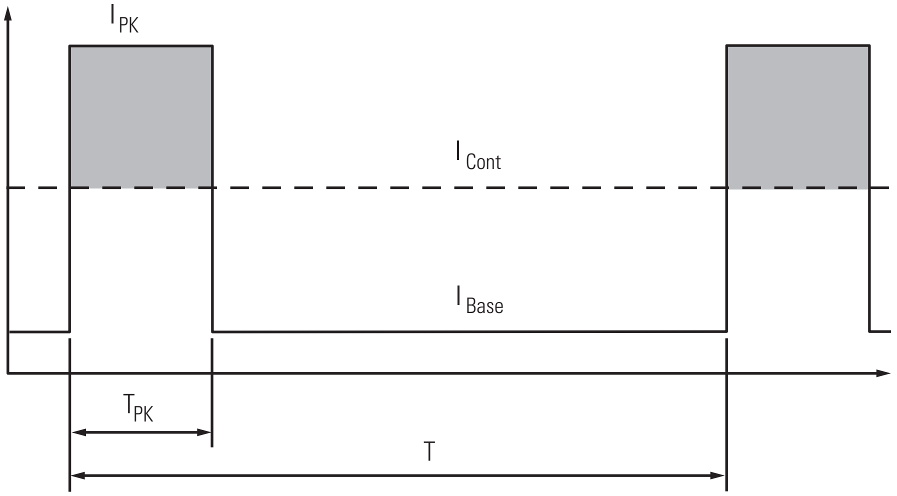
Kinetix 6500 Star Communication (EtherNet/IP network)



Peak Current Specifications

Load Duty-cycle Profile Example

$$D = \frac{T_{PK}}{T} \text{ x } 100\%$$

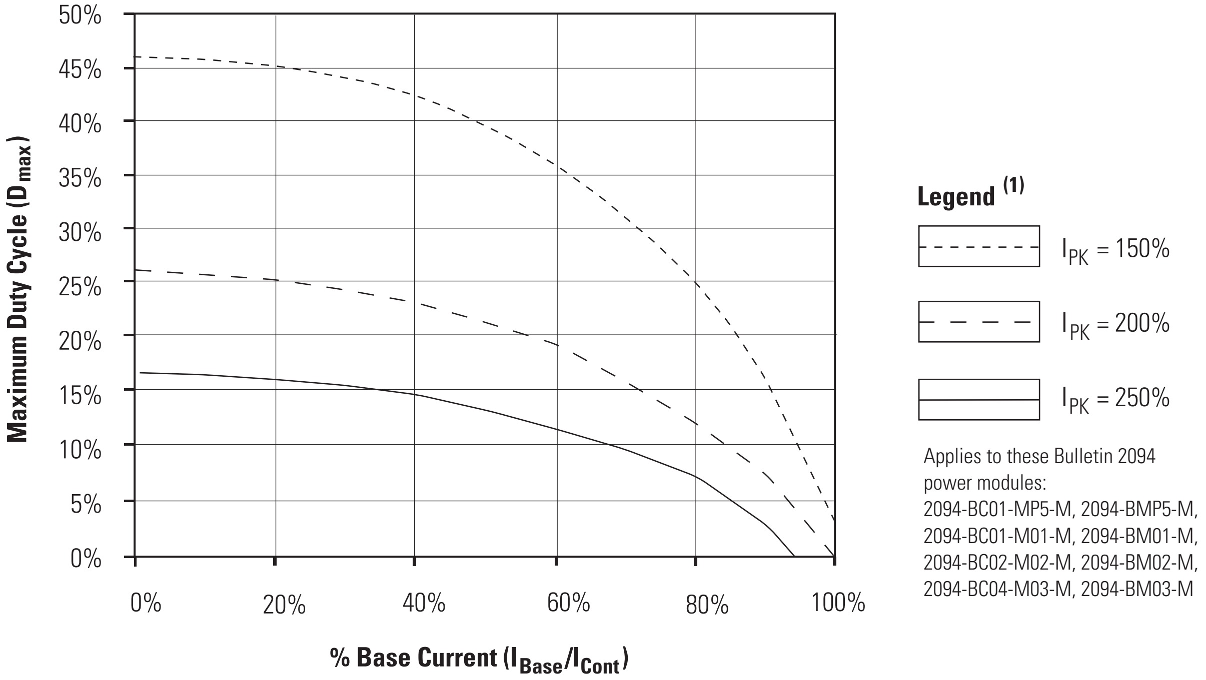


Peak Duty-cycle Definition of Terms

Term	Definition [1]
Continuous Current Rating (I_{Cont})	The maximum value of current that can be output continuously.
Peak Current Rating (I_{PKmax})	The maximum value of peak current that the drive can output. This rating is valid only for overload times less than T_{PKmax}.
Duty Cycle (D)	The ratio of time at peak to the Application Period and is defined as: $D = \frac{T_{PK}}{T} \text{ x } 100\%$
Time at Peak (T_{PK})	The time at peak current (I_{PK}) for a given loading profile. Must be less than or equal to T_{PKmax}.
Peak Current (I_{PK})	The level of peak current for a given loading profile. I_{PK} must be less than or equal to the Peak Current Rating (T_{PKMAX}) of the drive.
Base Current (I_{Base})	The level of current between the pulses of peak current for a given loading profile. I_{Base} must be less than or equal to the continuous current rating ($I_{Cont)}$ of the drive.
Loading Profile	The loading profile is comprised of I_{PK}, I_{Base}, T_{PK}, and D (or T) values and completely specify the operation of the drive in an overload situation. These values are collectively defined as the Loading Profile of the drive.
Application Period (T)	The sum of the times at I_{PK} (T_{PK}) and I_{Base}.

(1) All current values are specified as RMS.

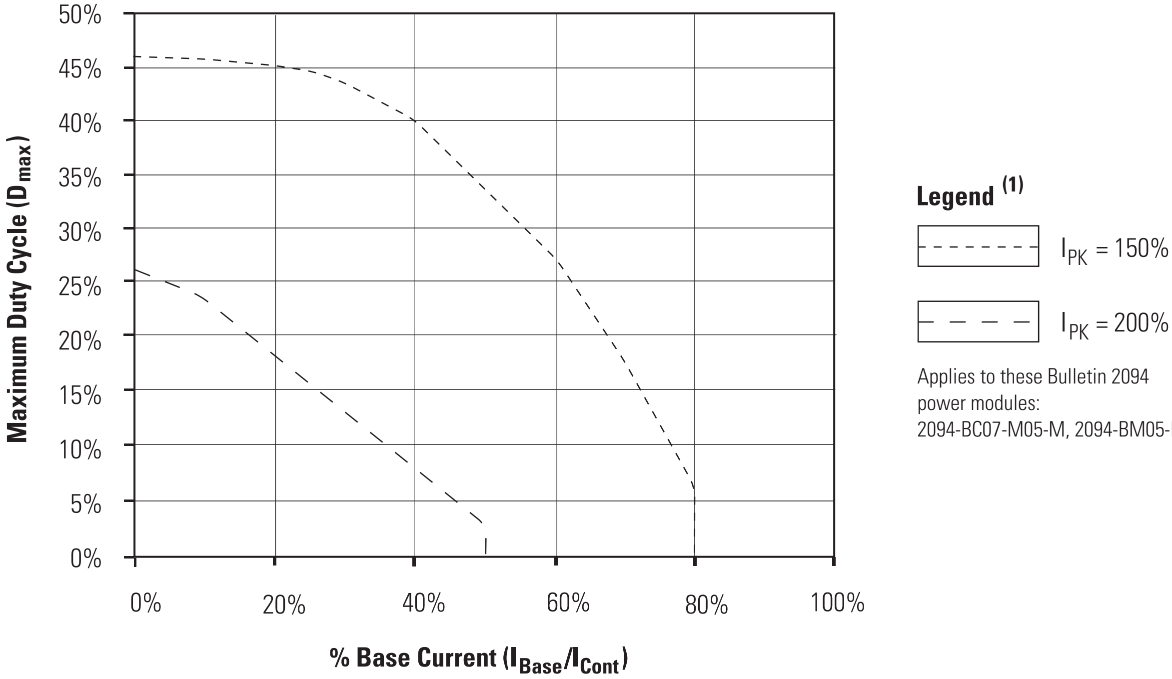
Peak Inverter Mode (T_{PK} < 2.0 s)



Applies to these Bulletin 2094 power modules:
2094-BC01-MP5-M, 2094-BMP5-M, 2094-BC01-M01-M, 2094-BM01-M, 2094-BC02-M02-M, 2094-BM02-M, 2094-BC04-M03-M, 2094-BM03-M

(1) Base current (I_{Base}) and peak current (I_{PK} are a percentage of the continuous drive current rating (I_{Cont}).

Peak Inverter Overload (T_{PK} < 2.0 s)



Applies to these Bulletin 2094 power modules:
2094-BC07-M05-M, 2094-BM05-M

(1) Base current (I_{Base}) and peak current (I_{PK}) are a percentage of the continuous drive current rating (I_{Cont}).

Kinetix 6200 and Kinetix 6500 Integrated Axis Modules

This section contains power specifications, mounting dimensions, and catalog numbers for the modular Bulletin 2094 (230V and 460V) IAM power modules. Choose your IAM power module based on the converter and inverter power requirements of your application.

Integrated Axis Module (converter) Power Specifications

IAM Module (460V) Power Specifications

Attribute	2094-BC01-MP5-M	2094-BC01-M01-M	2094-BC02-M02-M	2094-BC04-M03-M	2094-BC07-M05-M
AC input voltage	324…528V rms three-phase (360…480V nom)				
AC input frequency	47…63 Hz				
Main AC input current Nom (rms) Max inrush (0-pk) (1)	10.0 A 11.0 A		24.0 A 22.0 A	44.0 A 31.1 A	71.0 A 62.2 A
DC input voltage (common bus follower)	458…747V DC				
DC input current (common-bus follower)	9.0 A		22.6 A	41.5 A	67.7 A
Control power AC input voltage	95…264V rms single-phase (110…240V rms nom)				
Control power AC input current Nom (@ 220/230V AC) rms Nom (@ 110/115V AC) rms Max inrush (0-pk)	3 A 6 A 98 A (2)				
Nominal bus output voltage	650V DC				
Line loss ride through	20 ms				
Continuous output current to bus (A_{DC}	9.0 A		22.6 A	41.5 A	67.7 A
Peak output current to bus (A_{DC}) (3)	22.6 A		56.4 A	103.8 A	203.2 A
Bus overvoltage	825V DC				
Bus undervoltage	275V DC (when enabled) 425V DC (at the completion of pre-charge)				
Internal shunt Continuous power Peak power	50 W 5.6 kW			200W 22.5W	
Internal shunt resistor	115 Ω			27.75 Ω	
Shunt on	805V DC				
Shunt off	765V DC				
Continuous power output to bus	6 kW		15 kW	27.6 kW	45 kW
Peak power output at 480V (3)	15 kW		37.5 kW	69 kW	135 kW
Efficiency	97%				
Converter inductance	500 µH			125 µH	75 µH
Converter capacitance	110 µF		220 µF	940 µF	1410 µF
Short circuit current rating	200,000 A (rms) symmetrical				

(1) All IAM power modules are limited to 2 contactor cycles per minute (with up to 4 axes), or 1 contactor cycle per minute (with 5…8 axes). The cycle capability also depends on the converter power rating and the total system capacitance. To calculate cycle capability, refer to the Kinetix 6200 and Kinetix 6500 Modular Multi-axis Servo Drives User Manual, publication 2094-UM002.

(2) For eight axis systems with 230V AC control input voltage and 50 °C (122°F) ambient temperature the maximum inrush duration is less than 1/2 line cycle. To calculate the maximum inrush duration for other configurations, refer to the Kinetix 6200 and Kinetix 6500 Modular Multi-axis Servo Drives User Manual, publication 2094-UM002.

(3) Converter peak output duration equals 400 ms with a duty cycle of 16%.

Control Power Current Requirements

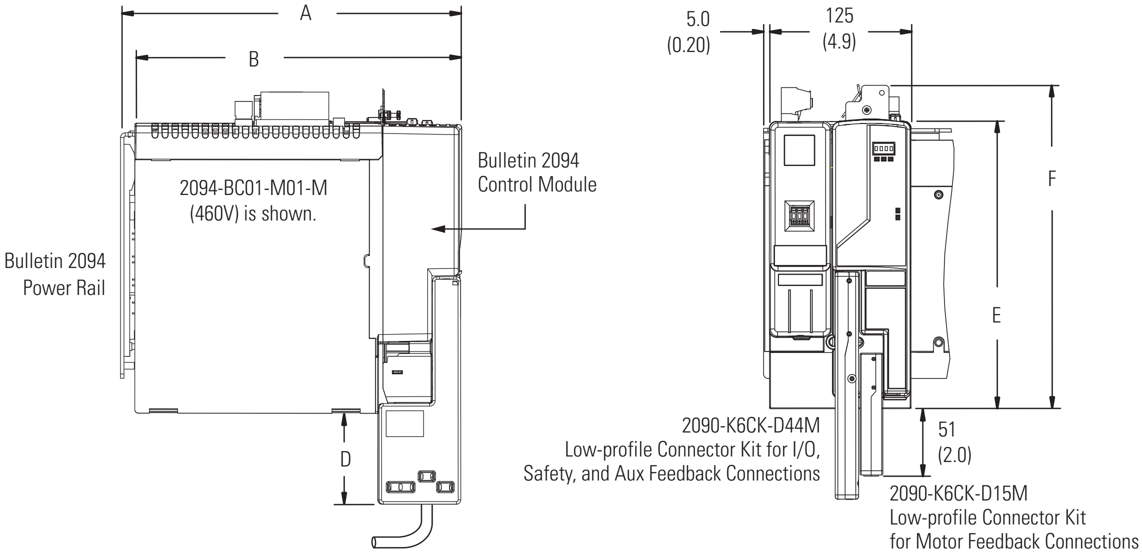
Power Modules on Power Rail	110/115V AC Input A	220/230V AC Input A	Input VA VA
IAM only	0.75	0.35	150
IAM, 1 AM	1.50	0.70	200
IAM, 2 AM	2.25	1.0	275
IAM, 3 AM	3.0	1.35	350

Power Modules on Power Rail	110/115V AC Input A	220/230V AC Input A	Input VA VA
IAM, 4 AM	3.75	1.70	450
IAM, 5 AM	4.50	2.0	550
IAM, 6 AM	5.25	2.40	650
IAM, 7 AM	6.0	3.0	750

Integrated Axis Module Dimensions

2094-BC01-MP5-M, 2094-BC01-M01-M, and 2094-BC02-M02-M IAM Modules

Dimensions are in millimeters (inches)

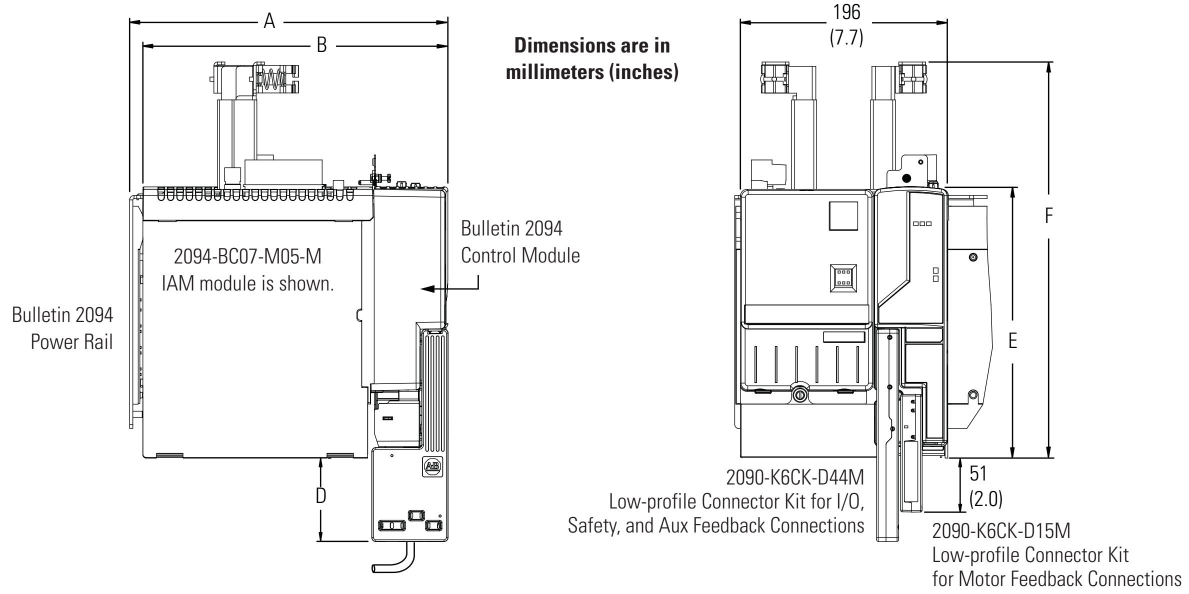


The IAM power module is shown mounted to the power rail with the control module attached.

IAM Power Module Dimensions

IAM Power Module Cat. No.	A mm (in.)	B mm (in.)	D mm (in.)	E mm (in.)	F mm (in.)
2094-BC01-MP5-M	302 (11.9)	290 (11.4)	80.0 (3.2)	260 (10.25)	285 (11.2)
2094-BC01-M01-M					
2094-BC02-M02-M					

2094-BC04-M03-M and 2094-BC07-M05-M IAM Modules



The IAM power module is shown mounted to the power rail with the control module attached.

IAM Module Dimensions

IAM Module Cat. No.	A mm (in.)	B mm (in.)	D mm (in.)	E mm (in.)	F mm (in.)
2094-BC04-M03-M	302 (11.9)	290 (11.4)	80 (3.1)	260 (10.25)	375 (14.7)
2094-BC07-M05-M					

Integrated Axis Module Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your module. For questions regarding product availability, contact your Allen-Bradley distributor.

2094 - *x* C *xx* - M *xx* - *x*

Power Module
M = Modular

Inverter Current Rating (peak of sine)
P5 = 4 A (460V input voltage)
01 = 9 A (460V input voltage)
02 = 15 A (460V input voltage)
03 = 30 A 460V input voltage)
05 = 49 A (460V input voltage)

P5 = 5 A (230V input voltage)
01 = 9 A (230V input voltage)
02 = 15 A (230V input voltage)
03 = 24 A (230V input voltage)
05 = 49 A (230V input voltage)

Inverter (Axis Module)

Converter Power Rating
01 = 6 kW (460V input voltage)
02 = 15 kW (460V input voltage)
04 = 28 kW (460V input voltage)
07 = 45 kW (460V input voltage)

05 = 3 kW (230V input voltage)
09 = 6 kW (230V input voltage)
16 = 11 kW (230V input voltage)
32 = 23 kW (230V input voltage)

Converter

Input Voltage
A = 230V AC, 50/60 Hz
B = 460V AC, 50/60 Hz

Bulletin Number

Kinetix 6200 and Kinetix 6500 Axis Modules

This section contains power specifications, mounting dimensions, and catalog numbers for the modular Bulletin 2094 (460V) axis modules (AM). Choose your AM power module based on the inverter power requirements of your application.

Axis Module (inverter) Power Specifications

These specifications apply to the axis module specified in the column heading by catalog number and the same axis module (inverter section) that resides within an IAM power module.

AM Module (inverter) 460V Power Specifications

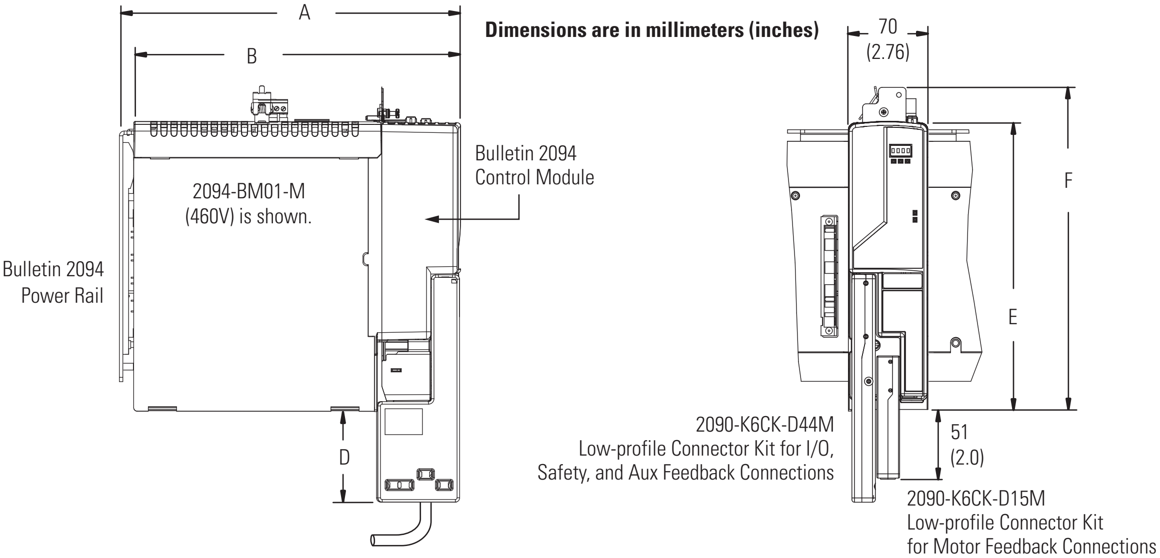
Attribute	2094-BMP5-M (2094-BC01-MP5-M)	2094-BM01-M (2094-BC01-M01-M)	2094-BM02-M (2094-BC02-M02-M)	2094-BM03-M (2094-BC04-M03-M)	2094-BM05-M (2094-BC07-M05-M)
Bandwidth (1) Velocity loop Current loop	500 Hz 1300 Hz				
PWM frequency	8 kHz		4 kHz		
Nominal input voltage	650V DC				
Continuous current (rms) (2)	2.8 A	6.1 A	10.3 A	21.2 A	34.6 A
Continuous current (sine) 0-pk (2)	4.0 A	8.6 A	14.6 A	30.0 A	48.9 A
Peak current (rms) (2)	7.0 A	15.3 A	25.8 A	53.0 A	69.2 A
Peak current (0-pk) (2)	9.9 A	21.6 A	36.4 A	75.0 A	97.9 A
Continuous power out (nom)	1.8 kW	3.9 kW	6.6 kW	13.5 kW	22.0 kW
Internal shunt Continuous power Peak power	50 W 5.6 kW			200 W 22.5 kW	
Internal shunt resistor	115 Ω			28.75 Ω	
Shunt on	805V DC				
Shunt off	765V DC				
Efficiency	98%				
Capacitance	75 μF	150 μF	270 μF	840 μF	1175 μF
Capacitive energy absorption	10 J	19 J	35 J	108 J	152 J
Short circuit current rating	200,000 A (rms) symmetrical				

(1) Bandwidth values vary based on tuning parameters and mechanical components.

(2) Continuous and peak current ratings are for high-speed operation. For constant velocity operation at an electrical output frequency below 5 Hz (75 rpm for 8-pole motors), the output current rating is reduced. See Motion Analyzer software to correctly size your drive. Refer to Peak Current Specifications on page 258 for duty cycle capability.

Axis Module Dimensions

2094-BMP5-M, 2094-BM01-M, and 2094-BM02-M AM Modules

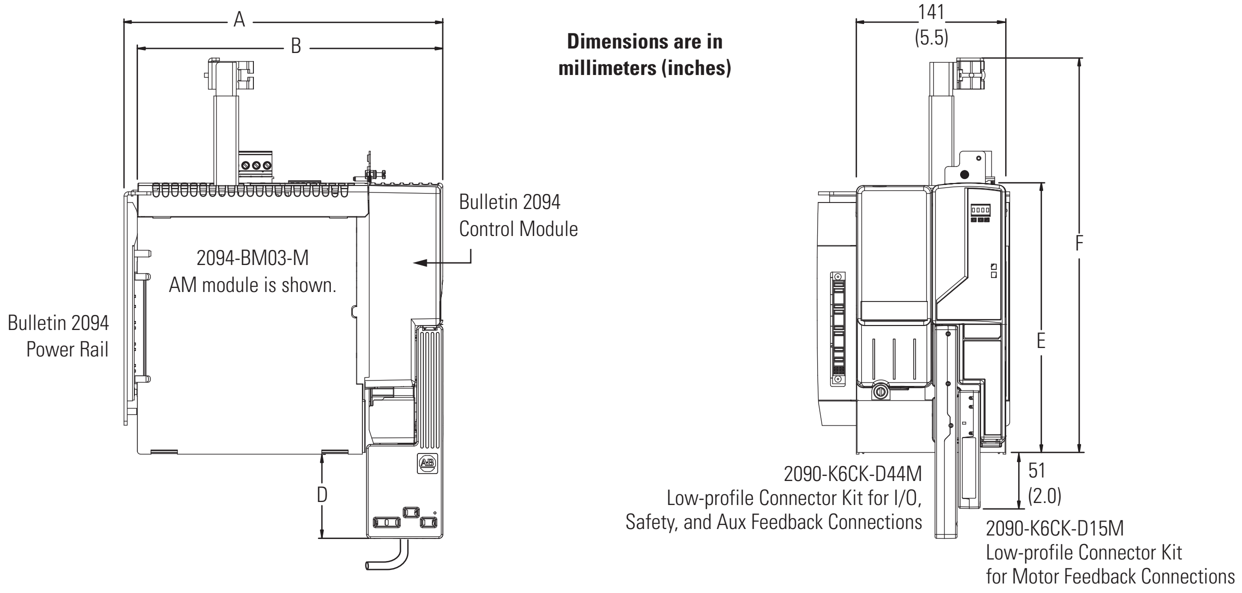


The AM power module is shown mounted to the power rail with the control module attached.

AM Power Module Dimensions

AM Module Cat. No.	A mm (in.)	B mm (in.)	D mm (in.)	E mm (in.)	F mm (in.)
2094-BMP5-M	302 (11.9)	290 (11.4)	80.0 (3.2)	260 (10.25)	285 (11.2)
2094-BM01-M					
2094-BM02-M					

2094-BM03-M and 2094-BM05-M AM Modules



The AM power module is shown mounted to the power rail with the control module attached.

AM Power Module Dimensions

AM Module Cat. No.	A mm (in.)	B mm (in.)	D mm (in.)	E mm (in.)	F mm (in.)
2094-BM03-M	302 (11.9)	290 (11.4)	80 (3.1)	260 (10.25)	375 (14.7)
2094-BM05-M					

Axis Module Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your module. For questions regarding product availability, contact your Allen-Bradley distributor.

2094 - x M xx - x

Power Module
M = Modular

Inverter Current Rating (peak of sine)

P5 = 4 A (460V input voltage)	P5 = 5 A (230V input voltage)
01 = 9 A (460V input voltage)	01 = 9 A (230V input voltage)
02 = 15 A (460V input voltage)	02 = 15 A (230V input voltage)
03 = 30 A 460V input voltage)	03 = 24 A (230V input voltage)
05 = 49 A (460V input voltage)	05 = 49 A (230V input voltage)

Module Input Voltage
AM = 230V AC, 50/60 Hz
BM = 460V AC, 50/60 Hz

Bulletin Number

Kinetix 6200 and Kinetix 6500 Control Modules

This section contains specifications and catalog numbers for the modular Bulletin 2094 SERCOS interface and EtherNet/IP network control modules. Choose your control module based on the communication and safety requirements of your application.

Control Module Specifications

Control modules couple with IAM and AM power modules to provide drive status indicators and an interface to I/O, communication, safety functionality, and feedback.

Feature	Safe Torque-off		Safe Speed Monitoring	
	2094-EN02D-M01-S0	2094-SE02F-M00-S0	2094-EN02D-M01-S1	2094-SE02F-M00-S1
Ethernet ports	2	1	2	1
SERCOS ports	–	Rx/Tx	–	Rx/Tx
DC Bus status indicator	√	√	√	√
Network status indicator	√	–	√	–
Module status indicator	√	–	√	–
Drive status indicator	–	√	–	√
Comm status indicator	–	√	–	√
Safety lock status indicator	–	–	√	√
PORT 1 status indicator	√	√	√	√
PORT 2 status indicator	√	–	√	–
IOD connector for I/O, safety, and auxiliary feedback	√	√	√	√
MF connector for motor feedback	√	√	√	√

Control Module Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your module. For questions regarding product availability, contact your Allen-Bradley distributor.

2094 - *xx* 02 *x* - M0*x* - S*x*

Safety Features
S0 = Safe torque-off
S1 = Safe speed monitoring

Control Features
M00 = SERCOS interface control
M01 = EtherNet/IP network control

Network Media
D = Twisted pair, dual port (EtherNet/IP network)
F = Fiber optic (SERCOS interface)

Axis Performance
02 = 500 µs

Network Communication
EN = EtherNet/IP network
SE = SERCOS interface

Bulletin Number

Kinetix 6200 and Kinetix 6500 General System Specifications

This section contains environmental, weight, power dissipation, circuit breaker/fuse, transformer, and contactor specifications.

Environmental Specifications

Attribute	Operational Range	Storage Range (nonoperating)
Temperature, ambient	0…50 °C (32…122 °F)	-40…70 °C (-40…158 °F)
Relative humidity	5…95% noncondensing	5…95% noncondensing
Altitude	1000 m (3281 ft) 3000 m (9843 ft) with derating	3000 m (9843 ft) during transport
Vibration	5…55 Hz @ 0.35 mm (0.014 in.) double amplitude, continuous displacement; 55…500 Hz @ 2.0 g peak constant acceleration (10 sweeps in each of 3 mutually perpendicular directions)	
Shock	15 g, 11 ms half-sine pulse (3 pulses in each direction of 3 mutually perpendicular directions)	

Weight Specifications

IAM Power Module	Cat. No.	Weight, approx. kg (lb)
IAM (460V)	2094-BC01-MP5-M	5.67 (12.5)
	2094-BC01-M01-M	5.67 (12.5)
	2094-BC02-M02-M	5.90 (13.0)
	2094-BC04-M03-M	9.53 (21.0)
	2094-BC07-M05-M	9.98 (22.0)

AM Power Module	Cat. No.	Weight, approx. kg (lb)
AM (460V)	2094-BMP5-M	3.18 (7.0)
	2094-BM01-M	3.18 (7.0)
	2094-BM02-M	3.40 (7.5)
	2094-BM03-M	5.44 (12.0)
	2094-BM05-M	5.90 (13.0)

Kinetix 6200 Control Module	Cat. No.	Weight, approx. kg (lb)
SERCOS interface	2094-SE02F-M00-S0	0.68 (1.5)
	2094-SE02F-M00-S1	

Kinetix 6500 Control Module	Cat. No.	Weight, approx. kg (lb)
EtherNet/IP network	2094-EN02D-M01-S0	0.68 (1.5)
	2094-EN02D-M01-S1	

Maximum Feedback Cable Lengths

Although motor feedback cables are available in standard lengths up to 90 m (295.3 ft), the drive/motor/feedback combination may limit the maximum feedback cable length. These tables assume the use of recommended cables as shown in the 2090-Series Motor/Actuator Cable Selection table on page 401.

Cable Lengths for Compatible Rotary Motors

Motor Cat. No.	**Absolute High-resolution (5V) Encoder** m (ft)	**Absolute High-resolution (9V) Encoder** m (ft)	**Incremental/TTL (5V) Encoder** m (ft)
MPL-B15*xxx*… MPL-B2*xxx*-E/V		90 (295.3)	
MPL-B3*xxx*… MPL-B5*xxx*-S/M		90 (295.3)	
MPL-B15*xxx*… MPL-B45*xxx*-H			30 (98.4)
MPM-B*xxxxx*-S/M		90 (295.3)	
MPF-B*xxxx*-S/M		90 (295.3)	
MPS-B*xxxx*-S/M		90 (295.3)	
RDB-B215*xx*-7/3	30 (98.4)		
RDB-B290*xx*-7/3 or RDB-B410*xx*-7/3	90 (295.3)		

Cable Lengths for Compatible Linear Actuators

Actuator Cat. No.	**Absolute High-resolution (9V) Encoder** m (ft)	**Incremental/TTL (5V) Encoder** m (ft)
MPMA-B*xxxxx* or MPAS-B*xxxxx*-V (ballscrew)	90 (295.3)	
MPMA-B*xxxxx* or MPAS-B*xxxxx*-A (direct drive)		30 (98.4)
MPAR-B*xxxxx*-V/M	90 (295.3)	
MPAI-B*xxxxx*M3	90 (295.3)	

Cable Lengths for Compatible Linear Motors

Motor Cat. No.	**Absolute High-resolution (5V) Encoder** m (ft)	**Incremental/TTL (5V) Encoder** m (ft)
LDC-Series	30 (98.4)	30 (98.4)

Maximum Power Cable Length

Although motor power cables are available in standard lengths up to 90 m (295.3 ft) and the Kinetix 6000 power rail is available in sizes up to eight axes, to meet CE requirements and improve system performance the combined motor power length for all axes on the same DC bus must not exceed 240 m (787 ft) for 460V systems.

Circuit Breaker/Fuse Specifications

While circuit breakers offer some convenience, there are limitations for their use. Circuit breakers do not handle high current inrush as well as fuses.

Make sure the selected components are properly coordinated and meet acceptable codes including any requirements for branch circuit protection. Evaluation of the short-circuit available current is critical and must be kept below the short-circuit current rating of the circuit breaker.

Use class CC, J, L, or R fuses, with current rating as indicated in the table below. The following fuse examples and Allen-Bradley circuit breakers are recommended for use with 2094-BC*xx*-M*xx*-M IAM power modules when the Line Interface Module (LIM) is not used.

IMPORTANT	LIM Modules (catalog numbers 2094-AL*xx*S, 2094-BL*xx*S, and 2094-XL75S-C*x*) provide branch circuit protection to the IAM power module. Follow all applicable NEC and local codes.

IAM Power Module Cat. No.	**V AC Input Power**			**Control Input Power**		**DC Common Bus Fuse**	
	Bussmann Fuse	**Allen-Bradley Circuit Breaker**		**Bussmann Fuse**	**Allen-Bradley Circuit Breaker**	**Bussmann Fuse**	**Ferraz Shawmut Fuse**
		Disconnect	**Magnetic Contactor**				
2094-BC01-MP5-M	KTK-R-20 (20 A)	1492-SP3D300	140M-F8E-C32	FNQ-R-10 (10 A)	1492-SP2D060	N/A	A100P20-1
2094-BC01-M01-M							
2094-BC02-M02-M	KTK-R-30 (30 A)	1492-SP3D400	140M-F8E-C45			FWJ-40A	A100P40-1
2094-BC04-M03-M	LPJ-45SP (45 A)	N/A	140U-H6C3-C50			FWJ-70A	A100P70-1
2094-BC07-M05-M	LPJ-80SP (80 A)		140U-H6C3-C90			FWJ-125A	A100P125-1

Contactor Ratings

This table provides the recommended contactor ratings for IAM power modules installed without a LIM module.

IAM Power Module Cat. No.	Contactor
2094-BC01-MP5-M 2094-BC01-M01-M	100-C23x10 (AC coil) 100-C23Zx10 (DC coil)
2094-BC02-M02-M	100-C37x10 (AC coil) 100-C37Zx10 (DC coil)
2094-BC04-M03-M	100-C60x10 (AC coil) 100-C60Zx10 (DC coil)
2094-BC07-M05-M	100-C72x10 (AC coil) 100-C72Zx10 (DC coil)

Input Transformer for Control Power

Attribute	Value
Input volt-amperes	750VA
Input voltage	460V AC
Output voltage	120…240V AC

Power Dissipation Specifications

Use this table to size an enclosure and calculate required ventilation for your Kinetix 6200 and Kinetix 6500 drive system.

Kinetix 6200 and Kinetix 6500 (1) Drive Modules	**Usage as % of Rated Power Output (watts)**				
	20%	**40%**	**60%**	**80%**	**100%**
IAM (converter) module (2)					
2094-BC01-MP5-M	18	21	25	29	34
2094-BC01-M01-M					33
2094-BC02-M02-M	36	44	54	64	75
2094-BC04-M03-M	50	67	87	110	135
2094-BC07-M05-M	71	101	137	179	226
IAM (inverter) module or AM module (2)					
2094-BC01-MP5-M or 2094-BMP5-M	46	54	61	69	77
2094-BC01-M01-M or 2094-BM01-M	57	73	90	108	126
2094-BC02-M02-M or 2094-BM02-M	53	72	93	116	142
2094-BC04-M03-M or 2094-BM03-M	94	130	169	211	255
2094-BC07-M05-M or 2094-BM05-M	121	183	252	326	407
Shunt module					
2094-BSP2	68	121	174	227	280

(1) Power dissipation for the Bulletin 2094 control modules, catalog numbers 2094-SE02F-M00-S*x* and 2094-EN02D-M01-S*x*, is included in the IAM and AM power module specifications.

(2) Internal shunt power is not included in the calculations and must be added based on utilization.

Power dissipation specifications are based on these calculations. This is an example:

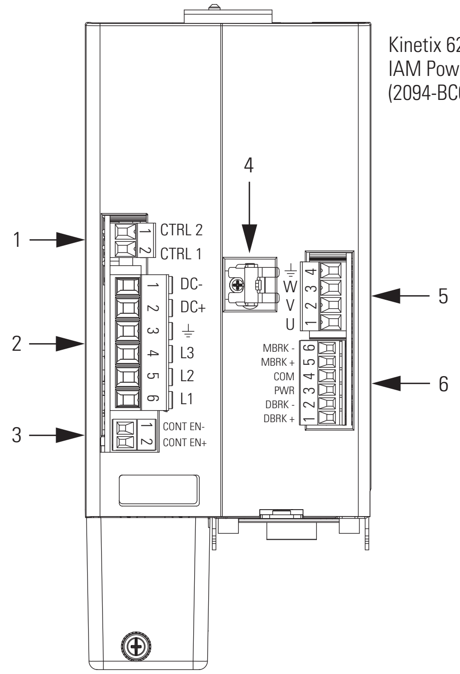
2094-BC02-M02-M with 4.52 A_{dc} (=20%) converter DC current and 10.3 A_{rms} (=100%) inverter output current.

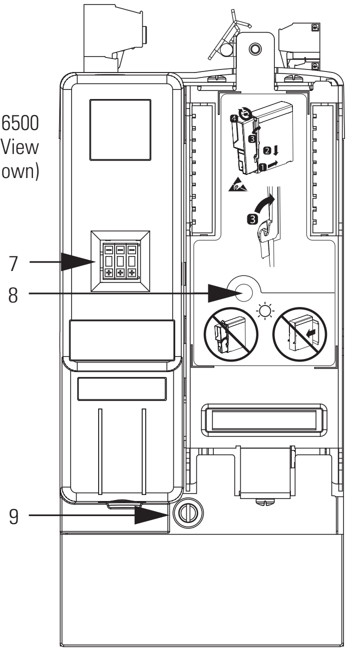
Converter loss (36 W) + Inverter loss (142 W) = 178 W total power dissipation.

Kinetix 6200 and Kinetix 6500 Features and Indicators

These are the features and indicators for the Kinetix 6200 and Kinetix 6500 IAM and AM power modules and control modules.

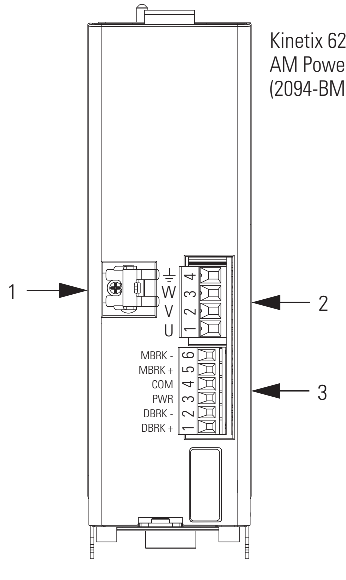
2094-AC*xx*-M*xx*-M and 2094-BC*xx*-M*xx*-M IAM Power Module Features and Indicators

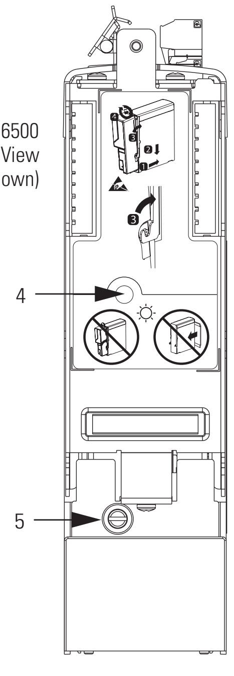

Kinetix 6200 or Kinetix 6500
IAM Power Module, Top View
(2094-BC01-MP5-M module is shown)


Kinetix 6200 or Kinetix 6500
IAM Power Module, Front View
(2094-BC01-MP5-M module is shown)

Item	Description
1	Control power (CPD) connector
2	DC bus/AC input power (IPD) connector
3	Contactor Enable (CED) connector
4	Motor cable shield clamp
5	Motor power (MP) connector
6	Motor/resistive brake (BC) connector
7	Node address switch
8	Power-applied indicator
9	Mounting screw

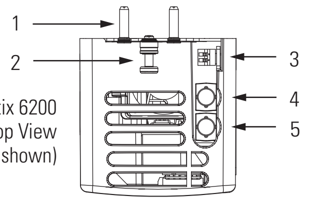
2094-AM*xx*-M and 2094-BM*xx*-M AM Power Module Features and Indicators


Kinetix 6200 or Kinetix 6500
AM Power Module, Top View
(2094-BMP5-M module is shown)


Kinetix 6200 or Kinetix 6500
AM Power Module, Front View
(2094-BMP5-M module is shown)

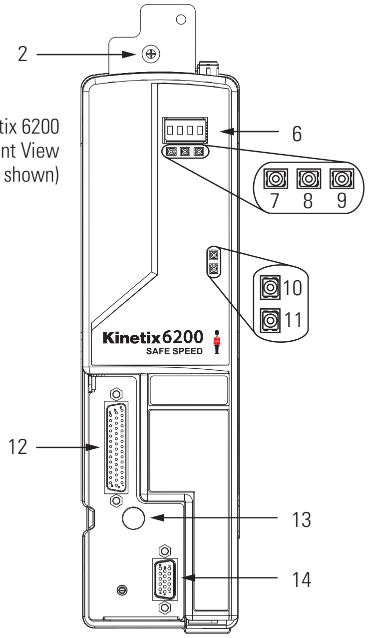
Item	Description
1	Motor cable shield clamp
2	Motor power (MP) connector
3	Motor/resistive brake (BC) connector
4	Power-applied indicator
5	Mounting screw

Control Module Features and Indicators (SERCOS)



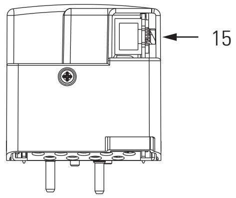
Kinetix 6200
Control Module, Top View
(2094-SE02F-M00-S1 module is shown)

Item	Description
1	Guide pins (2x)
2	Captive screw
3	SERCOS communication rate and optical power switches
4	SERCOS transmit (Tx) connector
5	SERCOS receive (Rx) connector



Kinetix 6200
Control Module, Front View
(2094-SE02F-M00-S1 module is shown)

Item	Description
6	Four-character status display
7	PORT 1 status Indicator
8	Drive status indicator
9	Comm status indicator
10	DC bus status indicator
11	Safety lock status indicator (2094-SE02F-M00-S1 modules only)
12	I/O, safety, and aux feedback (IOD) connector
13	Power module mounting screw access hole
14	Motor feedback (MF) connector

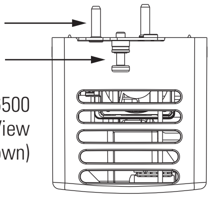


Kinetix 6200
Control Module, Bottom View
(2094-SE02F-M00-S1 module is shown)

Item	Description
15	Ethernet (PORT1) connector

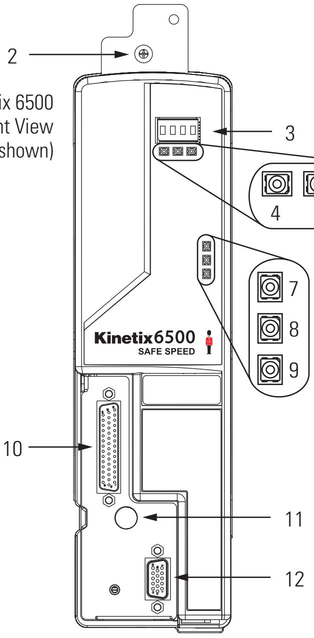
For connector kit options, refer to Breakout Components and Connector Kits beginning on page 440.

Control Module Features and Indicators (EtherNet/IP network)



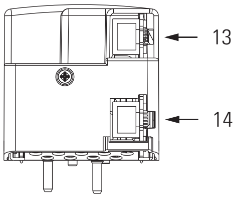
Kinetix 6500
Control Module, Top View
(2094-EN02D-M01-S1 is shown)

Item	Description
1	Guide pins (2x)
2	Captive screw



Kinetix 6500
Control Module, Front View
(2094-EN02D-M01-S1 is shown)

Item	Description
3	Four-character status display
4	PORT 1 status indicator
5	PORT 2 status indicator
6	Module status indicator
7	Network status indicator
8	DC bus status indicator
9	Safety lock status indicator (2094-EN02D-M01-S1 modules only)
10	I/O, safety, and aux feedback (IOD) connector
11	Power module mounting screw access hole
12	Motor feedback (MF) connector



Kinetix 6500
Control Module, Bottom View
(2094-EN02D-M01-S1 is shown)

Item	Description
13	Ethernet (PORT1) connector
14	Ethernet (PORT2) connector

Notes:

Kinetix 6000 Multi-axis Servo Drives

The Kinetix 6000 multi-axis servo drives provide powerful simplicity to handle even the most demanding applications quickly, easily, and cost-effectively. By providing advanced control capability along with innovative design and installation features, the Kinetix 6000 drives significantly improve system performance while saving time and money. The compact size, simplified wiring, and easy-to-use components make the Kinetix 6000 drives an ideal choice for both OEMs and end-users. Target applications for the Kinetix 6000 drives include packaging, material handling, converting, and assembly.

The Kinetix 6000 multi-axis servo drives are part of the Kinetix Integrated Motion solution.

Kinetix 6000 Servo Drive Components

Kinetix 6000 servo drive systems consist of these required components:

- One integrated axis module (IAM or leader IAM), 2094-AC*xx*-M*xx*-S (230V) or 2094-BC*xx*-M*xx*-S (460V)
- Up to seven axis modules, 2094-AM*xx*-S (230V) or 2094-BM*xx*-S (460V)
- One power rail, 2094-PRS1, 2094-PRS2, 2094-PRS3, 2094-PRS4, 2094-PRS5, 2094-PRS6, 2094-PRS7, or 2094-PRS8
- One to eight MP-Series, TL-Series, LDC-Series, LDL-Series, or RDD-Series rotary servo motors or linear motors/actuators. RDD-Series motors require the 2090-K6CK-KENDAT low-profile feedback module, all others require the 2090-K6CK-D15M low-profile connector kit for flying-lead feedback cables.
- One to eight motor power and feedback cables
- Two to nine SERCOS fiber-optic cables

Kinetix 6000 systems may also include any of these optional components:

- One or more integrated axis modules used as follower IAM, 2094-AC*xx*-M*xx*-S (230V) or 2094-BC*xx*-M*xx*-S (460V) and associated axis modules, power rails, motors, and cables as required for the application
- One shunt module, 2094-BSP2 with optional Bulletin 1394 external passive shunt module
- Slot-filler modules, 2094-PRF
- Bulletin 2094 Line Interface Module (LIM)
- Bulletin 2090 Resistive Brake Module (RBM)
- Bulletin 1336 external active shunt module (dynamic brake)

Kinetix 6000 IAM/AM Module Series Change

The peak current ratings of the Kinetix 6000 AM modules (series A and B) are configured at the factory as 150% of continuous current. You can program 460V (series B) AM modules and the equivalent IAM (inverter) modules, for up to 250% of continuous inverter current.

Kinetix 6000 Series Change

IAM Module Cat. No.	AM Module Cat. No.	Peak Current Rating	
		Series A (inverter)	Series B (inverter)
2094-BC01-MP5-S	2094-BMP5-S	150%	250%
2094-BC01-M01-S	2094-BM01-S	150%	250%
2094-BC02-M02-S	2094-BM02-S	150%	250%
2094-BC04-M03-S	2094-BM03-S	150%	250%
2094-BC07-M05-S	2094-BM05-S	150%	200%

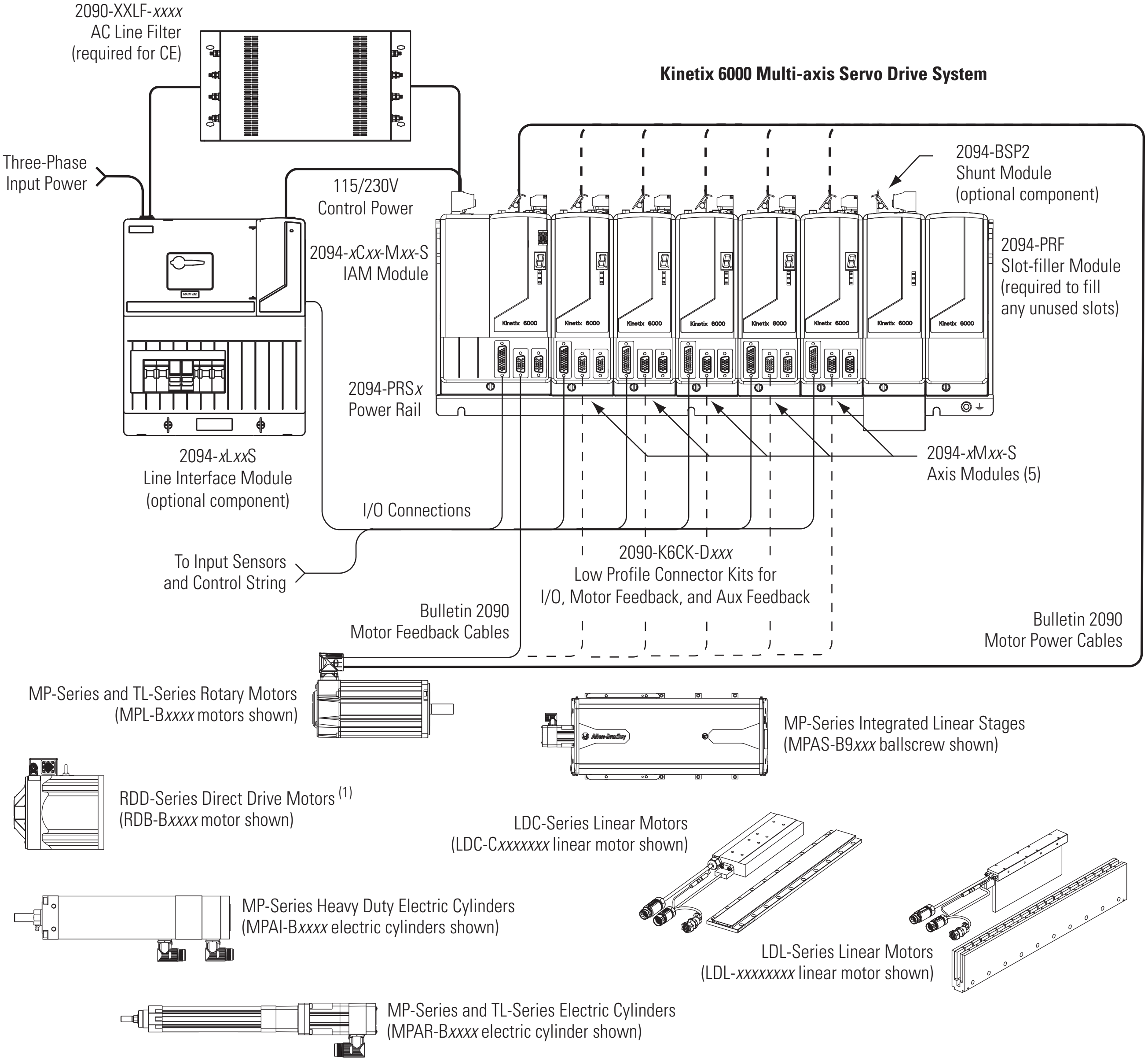
IMPORTANT Before your drive will deliver up to 250% peak performance, you must enable the peak enhancement feature by configuring your drive by using DriveExplorer or RSLogix 5000 software.

Refer to the Kinetix 6000 Multi-axis Servo Drive User Manual, publication 2094-UM001, to recalculate torque and accel/decel limit values, and paste them into the appropriate Axis Properties dialog box in RSLogix 5000 software.

For sizing your drive/motor combination by using series-B drives and the peak enhancement feature, use Motion Analyzer software, version 4.6 or later.

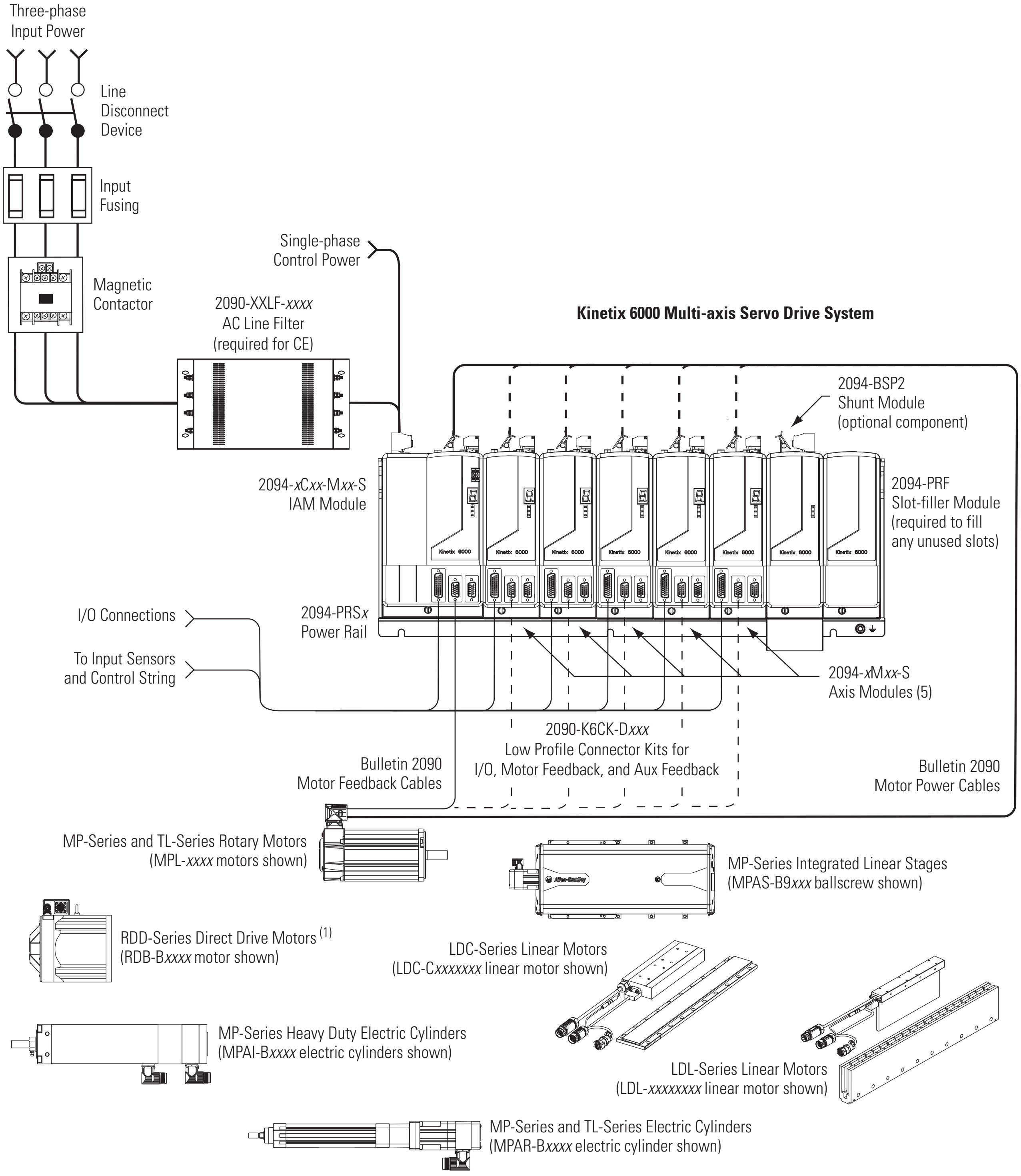
Typical Hardware Configurations

Kinetix 6000 System (with LIM module)



(1) RDD-Series direct-drive motors require the 2090-K6CK-KENDAT low-profile feedback module for Kinetix 6000 drive applications.

Kinetix 6000 System (without LIM module)

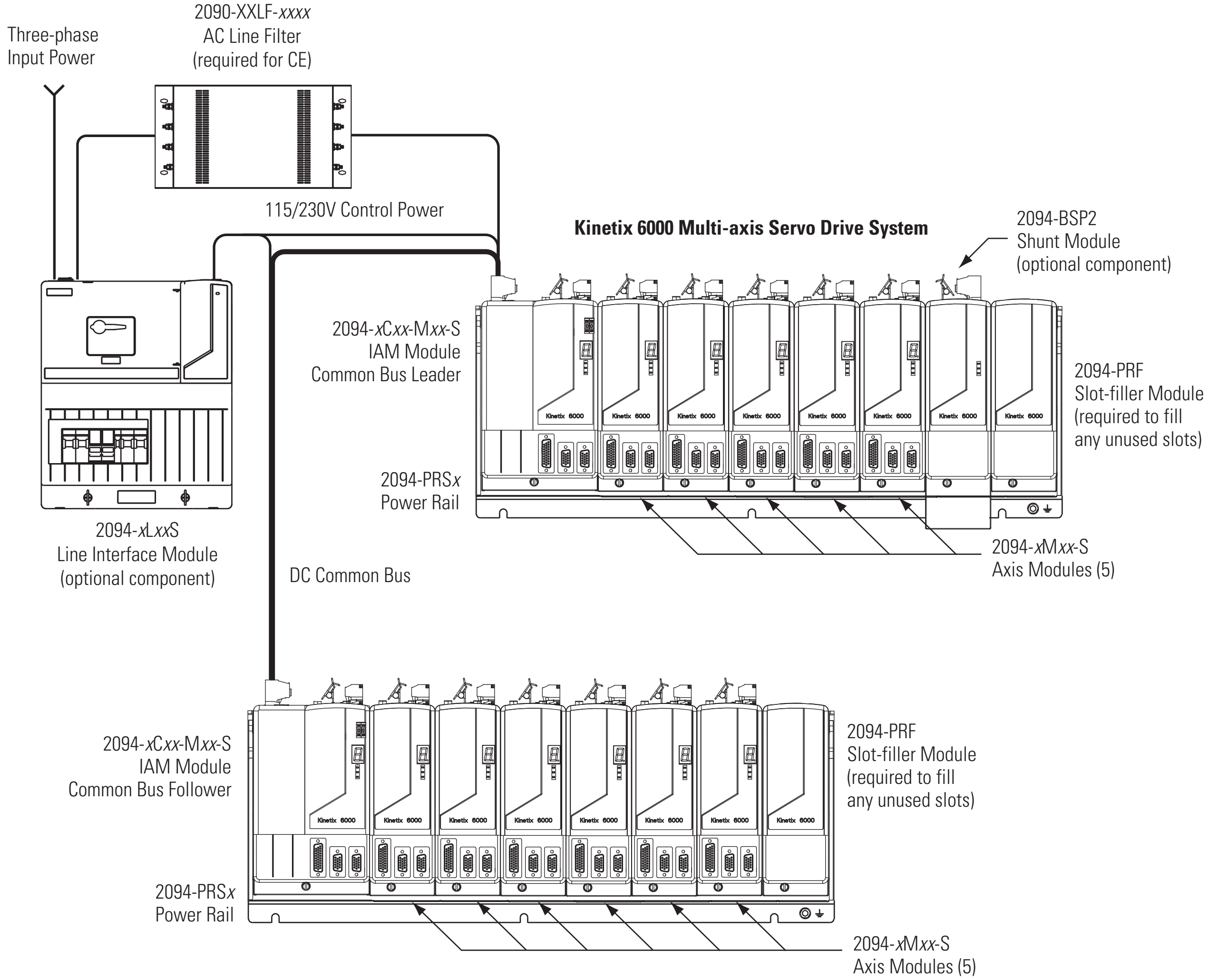


(1) RDD-Series direct-drive motors require the 2090-K6CK-KENDAT low-profile feedback module for Kinetix 6000 drive applications.

In the Kinetix 6000 system configuration below, the leader IAM module is connected to the follower IAM module via the DC common bus. When planning your panel layout, you must calculate the total bus capacitance of your DC common bus system to make sure that the leader IAM module is sized sufficiently to pre-charge the entire system. Refer to the Kinetix 6000 Servo Drive User Manual, publication 2094-UM001, when making this calculation.

IMPORTANT	If total bus capacitance of your system exceeds the leader IAM module pre-charge rating, the IAM module seven-segment status will display error code E90 (pre-charge timeout fault) if input power is applied. To correct this condition, you must replace the leader IAM module with a larger module or decrease the total bus capacitance by removing axis modules.

Kinetix 6000 System (DC common bus)

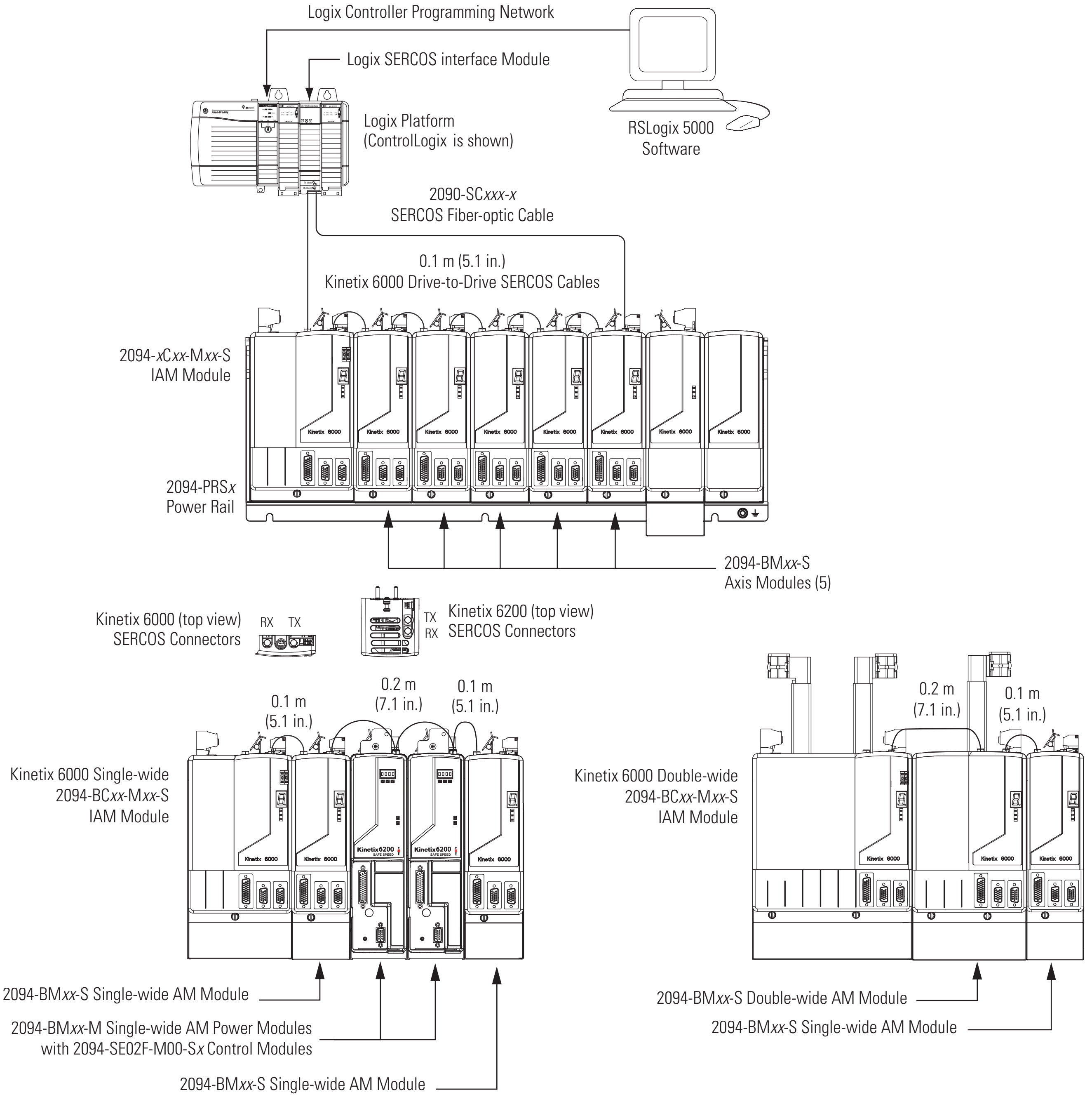


Motors and other details common to both three-phase AC and DC common-bus configurations are removed.

Typical Communication Configurations

In this example, drive-to-drive SERCOS cable lengths and catalog numbers are shown for the Kinetix 6000 drives and when Kinetix 6000 and Kinetix 6200 drive modules exist on the same power rail.

Kinetix 6000 Drive Communication (SERCOS)



Peak Enhancement Specifications

Drives that support the Peak-enhanced mode have the capability of increasing the maximum inverter peak current to achieve greater overload performance.

IMPORTANT The peak enhancement feature requires the use of RSLogix 5000 software and drive firmware as specified below.

Peak Enhancement Software and Firmware Requirements

IAM Module Cat. No.	AM Module Cat. No.	RSLogix 5000 Software Version	Kinetix 6000 Drive Firmware Revision
2094-BC01-MP5-S	2094-BMP5-S	16 or later	1.111 or later
2094-BC01-M01-S	2094-BM01-S	16 or later	1.111 or later
2094-BC02-M02-S	2094-BM02-S	16 or later	1.111 or later
2094-BC04-M03-S	2094-BM03-S	17 or later	1.117 or later
2094-BC07-M05-S	2094-BM05-S	17 or later	1.117 or later

Kinetix 6000 Peak Overload Support

IAM/AM Module Cat. No.	Module	Safe-off Drive	Series A	Series B
2094-BC*xx*-M*xx*	IAM	Non safe-off	Standard	Standard
2094-BM*xx*	AM			
2094-BC*xx*-M*xx*-S	IAM	Safe-off	Standard	Standard or Peak enhanced (1)
2094-BM*xx*-S	AM			

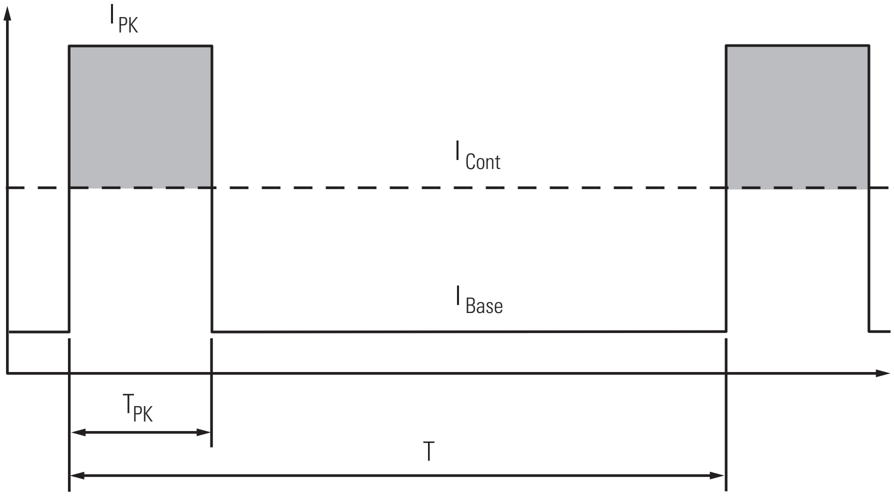
(1) Standard mode is enabled by default to preserve backward compatibility, but you can enable the Peak Enhanced mode to achieve increased peak current performance. Refer to Kinetix 6000 IAM/AM Module Series Change on page 276 for information on enabling the Peak Enhanced mode.

Kinetix 6000 Peak Current Ratings

IAM/AM Module Cat. No.	Peak Inverter Current Rating		Peak Converter Current Rating	
	Standard	Peak Enhanced	Series A	Series B
2094-BC01-MP5-S	150%	250%	200%	250%
2094-BC01-M01-S	150%	250%	200%	250%
2094-BC02-M02-S	150%	250%	200%	250%
2094-BC04-M03-S	150%	250%	200%	250%
2094-BC07-M05-S	150%	200%	200%	300%
2094-BMP5-S	150%	250%	N/A	N/A
2094-BM01-S	150%	250%	N/A	N/A
2094-BM02-S	150%	250%	N/A	N/A
2094-BM03-S	150%	250%	N/A	N/A
2094-BM05-S	150%	200%	N/A	N/A

Load Duty-cycle Profile Example

$$D = \frac{T_{PK}}{T} \times 100\%$$

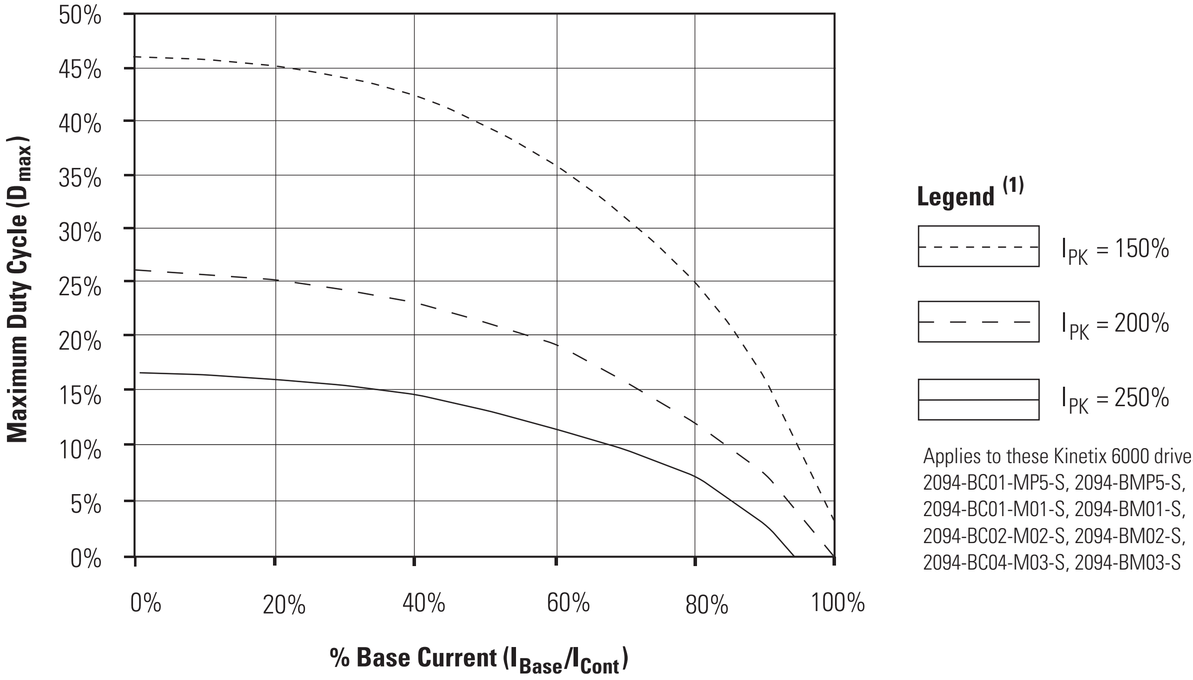


Peak Duty-cycle Definition of Terms

Term	Definition [1]
Continuous Current Rating (I_{Cont})	The maximum value of current that can be output continuously.
Peak Current Rating (I_{PKmax})	The maximum value of peak current that the drive can output. This rating is valid only for overload times less than T_{PKmax}.
Duty Cycle (D)	The ratio of time at peak to the Application Period and is defined as: $D = \frac{T_{PK}}{T} \times 100\%$
Time at Peak (T_{PK})	The time at peak current (I_{PK}) for a given loading profile. Must be less than or equal to T_{PKmax}.
Peak Current (I_{PK})	The level of peak current for a given loading profile. I_{PK} must be less than or equal to the Peak Current Rating (T_{PKMAX}) of the drive.
Base Current (I_{Base})	The level of current between the pulses of peak current for a given loading profile. I_{Base} must be less than or equal to the continuous current rating (I_{Cont}) of the drive.
Loading Profile	The loading profile is comprised of I_{PK}, I_{Base}, T_{PK}, and D (or T) values and completely specify the operation of the drive in an overload situation. These values are collectively defined as the Loading Profile of the drive.
Application Period (T)	The sum of the times at I_{PK} (T_{PK}) and I_{Base}.

(1) All current values are specified as RMS.

Peak Enhanced Inverter Mode (T_{PK} < 2.0 s)

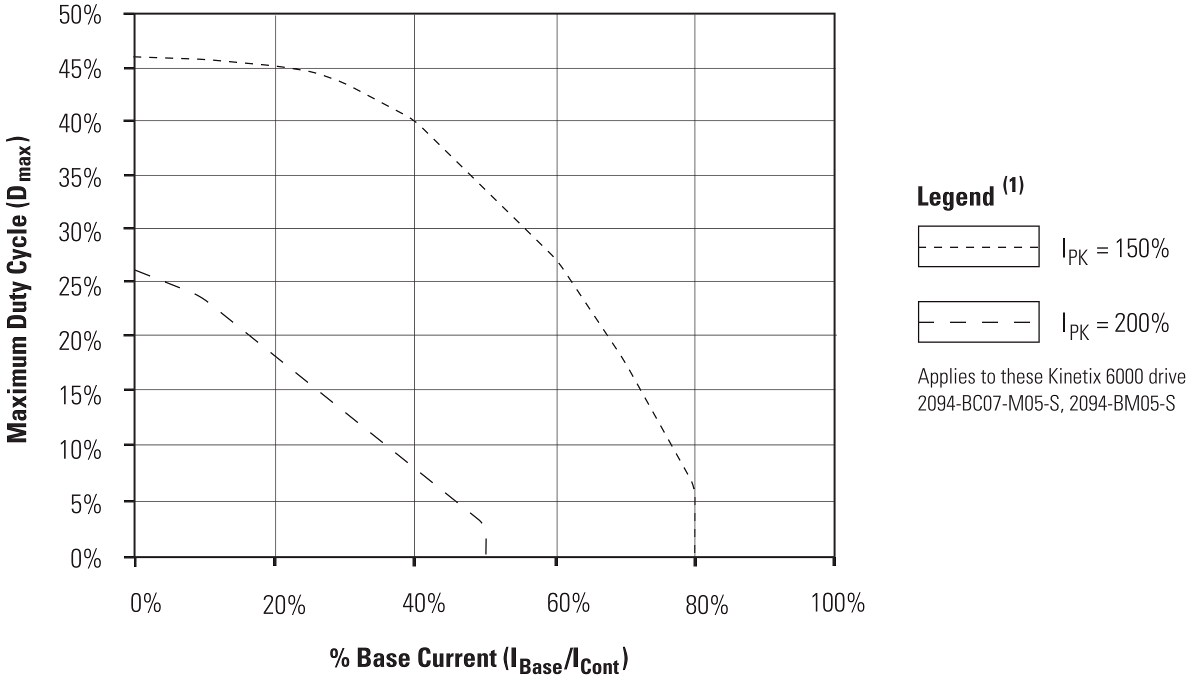


Applies to these Kinetix 6000 drives:
2094-BC01-MP5-S, 2094-BMP5-S,
2094-BC01-M01-S, 2094-BM01-S,
2094-BC02-M02-S, 2094-BM02-S,
2094-BC04-M03-S, 2094-BM03-S

---

(1) Base current (I_{Base}) and peak current (I_{PK} are a percentage of the continuous drive current rating (I_{Cont}).

Peak Inverter Overload (T_{PK} < 2.0 s)



Applies to these Kinetix 6000 drives:
2094-BC07-M05-S, 2094-BM05-S

---

(1) Base current (I_{Base}) and peak current (I_{PK}) are a percentage of the continuous drive current rating (I_{Cont}).

Kinetix 6000 Integrated Axis Modules

This section contains power specifications, mounting dimensions, and catalog numbers for the Bulletin 2094 (230V and 460V) integrated axis modules (IAM). Choose your IAM module based on the converter and inverter power requirements of your application.

Integrated Axis Module (converter) Power Specifications

IAM Module (230V) Power Specifications

Attribute	2094-AC05-MP5-S	2094-AC05-M01-S	2094-AC09-M02-S	2094-AC16-M03-S	2094-AC32-M05-S
AC input voltage	195…264V rms three-phase (230V nom)				
AC input frequency	47…63 Hz				
Main AC input current Nom (rms) Max inrush (0-pk) (1)	10 A 19 A		19 A 37 A	36 A 73 A	71 A 138 A
DC input voltage (common-bus follower)	275…375V DC				
DC input current (common-bus follower)	10 A		19 A	36 A	71 A
Control power AC input voltage	95…264V rms single-phase (110…240V rms nom)				
Control power AC input current Nom (@ 220/230V AC) rms Nom (@ 110/115V AC) rms Max inrush (0-pk)	3 A 6 A 20 A			3 A 6 A 83 A (2)	
Nominal bus output voltage	325V DC				
Line loss ride through	20 ms				
Continuous output current to bus (A_{DC})	10 A		19 A	36 A	71 A
Peak output current to bus (A_{DC}) (3)	20 A		38 A	72 A	142 A
Bus overvoltage	415V DC				
Bus undervoltage	138V DC				
Internal shunt Continuous power Peak power	N/A N/A		50 W 8200 W	200 W 5700 W	200 W 5700 W
Internal shunt resistor	N/A		20 Ω	28.75 Ω	28.75 Ω
Shunt on	N/A		405V DC		
Shunt off	N/A		375V DC		
Continuous power output to bus	3 kW		6 kW	11.3 kW	22.5 kW
Peak power output	6 kW		12 kW	22.6 kW	45.0 kW
Efficiency	95%				
Converter inductance	N/A			150 µH	75 µH
Converter capacitance	270 µF		540 µF	1320 µF	1980 µF
Short circuit current rating	200,000 A (rms) symmetrical				

(1) All 2094-*xCxx* IAM modules are limited to 2 contactor cycles per minute (with up to 4 axes), or 1 contactor cycle per minute (with 5…8 axes). The cycle capability also depends on the converter power rating and the total system capacitance. To calculate cycle capability, refer to the Kinetix 6000 Multi-axis Servo Drives User Manual, publication 2094-UM001.

(2) For eight axis systems with 230V AC control input voltage and 50 °C (122°F) ambient temperature the maximum inrush duration is less than 1/2 line cycle. To calculate the maximum inrush duration for other configurations, refer to the Kinetix 6000 Multi-axis Servo Drives User Manual, publication 2094-UM001.

(3) Peak output current duration equals 250 ms.

IAM Module (460V) Power Specifications (series A and B)

Attribute	2094-BC01-MP5-S	2094-BC01-M01-S	2094-BC02-M02-S	2094-BC04-M03-S	2094-BC07-M05-S
AC input voltage	324…528V rms three-phase (360…480V nom)				
AC input frequency	47…63 Hz				
Main AC input current					
Nom (rms)	10.0 A		24.0 A	44.0 A	71.0 A
Max inrush (0-pk) (1)	11.0 A		22.0 A	31.1 A	62.2 A
DC input voltage (common bus follower)	458…747V DC				
DC input current (common-bus follower)	9.0 A		22.6 A	41.5 A	67.7 A
Control power AC input voltage	95…264V rms single-phase (110…240V rms nom)				
Control power AC input current					
Nom (@ 220/230V AC) rms	3 A				
Nom (@ 110/115V AC) rms	6 A				
Max inrush (0-pk)	98 A (2)				
Nominal bus output voltage	650V DC				
Line loss ride through	20 ms				
Continuous output current to bus (A_{DC})	9.0 A		22.6 A	41.5 A	67.7 A
Peak output current to bus (A_{DC})					
Series A drives (3)	18.1 A		45.2 A	83.1 A	135.4 A
Series B drives	22.6 A (4)		56.4 A (4)	103.8 A (4)	203.2 A (5)
Bus overvoltage	825V DC				
Bus undervoltage	275V DC				
Internal shunt					
Continuous power	50 W			200 W	
Peak power	5.6 kW			22.5 kW	
Internal shunt resistor	115 Ω			28.75 Ω	
Shunt on	805V DC				
Shunt off	755V DC				
Continuous power output to bus	6 kW		15 kW	27.6 kW	45 kW
Peak power output					
Series A drives (3)	12 kW		30 kW	55.2 kW	90 kW
Series B drives	15 kW (4)		37.5 kW (4)	69 kW (4)	135 kW (5)
Efficiency	97%				
Converter inductance	500 µH			125 µH	75 µH
Converter capacitance	110 µF		220 µF	940 µF	1410 µF
Short circuit current rating	200,000 A (rms) symmetrical				

(1) All 2094-*x*C*xx* IAM modules are limited to 2 contactor cycles per minute (with up to 4 axis modules), or 1 contactor cycle per minute (with 5 to 8 axis modules). The cycle capability also depends on the converter power rating and the total system capacitance. Refer to the Kinetix 6000 Multi-axis Servo Drives User Manual, publication 2094-UM001 when making calculations.

(2) For eight axis systems with 230V AC control input voltage and 50 °C (122°F) ambient temperature the maximum inrush duration is less than 1/2 line cycle. To calculate the maximum inrush duration for other configurations, refer to the Kinetix 6000 Multi-axis Servo Drives User Manual, publication 2094-UM001.

(3) Peak output current duration equals 250 ms.

(4) Converter peak output duration equals 400 ms with a duty cycle of 16%.

(5) Converter peak output duration equals 200 ms with a duty cycle of 3%.

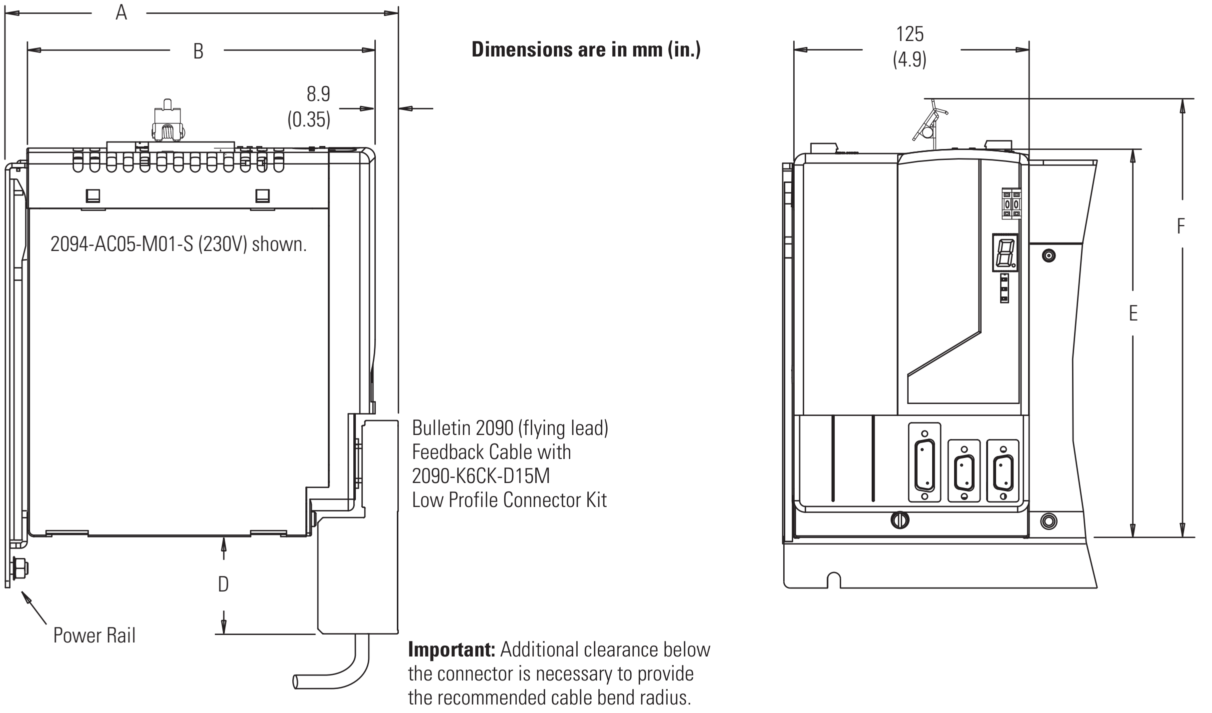
Control Power Current Requirements

Modules on Power Rail	110/115V AC Input A	220/230V AC Input A	Input VA VA
IAM only	0.75	0.35	150
IAM, 1 AM	1.50	0.70	200
IAM, 2 AM	2.25	1.0	275
IAM, 3 AM	3.0	1.35	350

Modules on Power Rail	110/115V AC Input A	220/230V AC Input A	Input VA VA
IAM, 4 AM	3.75	1.70	450
IAM, 5 AM	4.50	2.0	550
IAM, 6 AM	5.25	2.40	650
IAM, 7 AM	6.0	3.0	750

Integrated Axis Module Dimensions

2094-AC05-MP5-S, 2094-AC05-M01-S, and 2094-AC09-M02-S Dimensions (230V)
2094-BC01-MP5-S, 2094-BC01-M01-S, and 2094-BC02-M02-S Dimensions (460V)



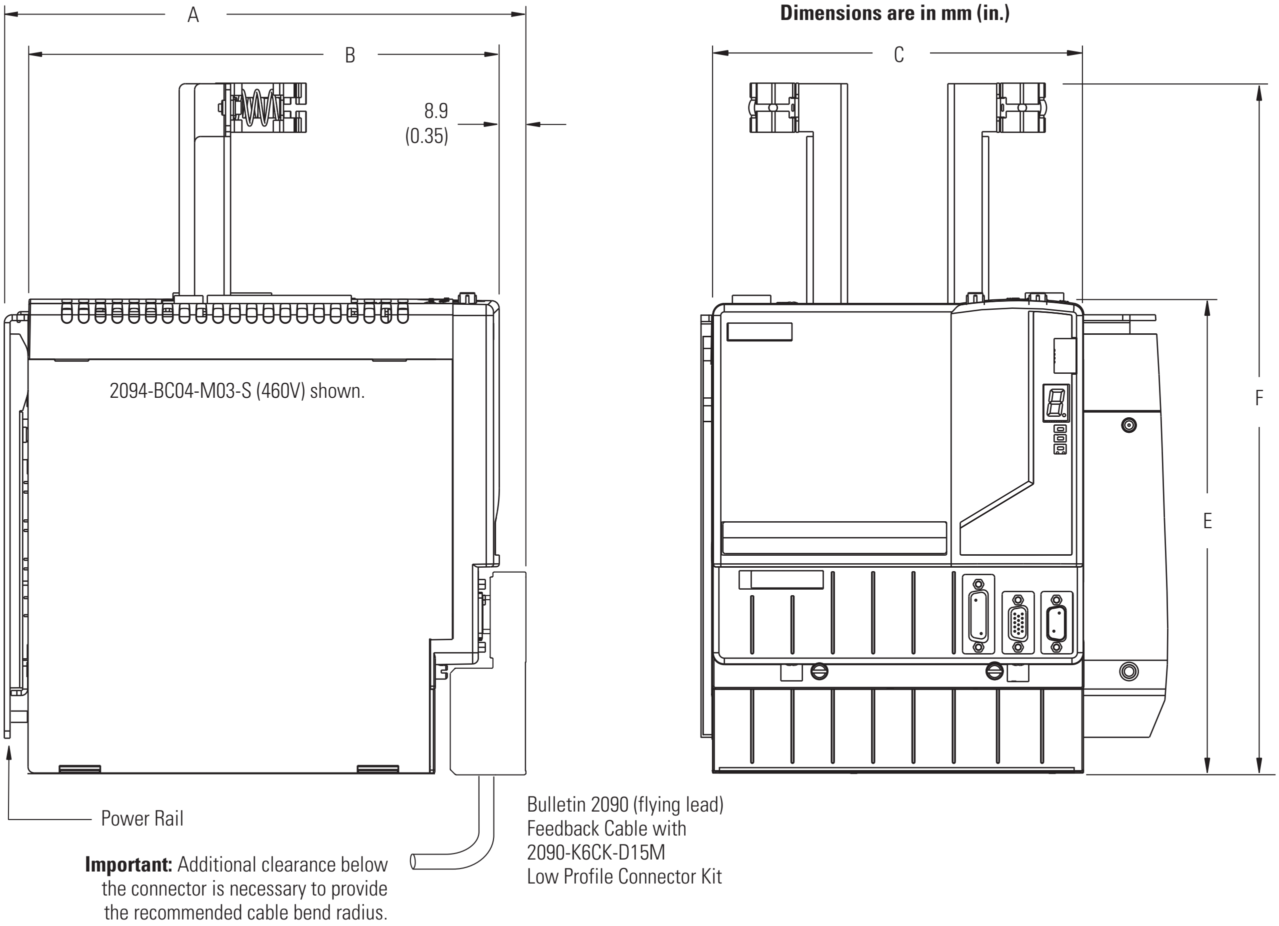
Important: Additional clearance below the connector is necessary to provide the recommended cable bend radius.

Modules are shown mounted to the power rail and the dimensions reflect that in the depth of the module.

IAM Module Dimensions (series A and B)

IAM Module Cat. No.	A mm (in.)	B mm (in.)	D mm (in.)	E mm (in.)	F mm (in.)
2094-AC05-MP5-S	198 (7.8)	176 (7.0)	51 (2.0)	206 (8.2)	237 (9.3)
2094-AC05-M01-S					
2094-AC09-M02-S					
2094-BC01-MP5-S	272 (10.7)	249 (9.8)	0 (0)	256 (10.1)	287 (11.3)
2094-BC01-M01-S					
2094-BC02-M02-S					

2094-AC16-M03-S and 2094-AC32-M05-S Dimensions (230V)
2094-BC04-M03-S and 2094-BC07-M05-S Dimensions (460V)



Modules are shown mounted to the power rail and the dimensions reflect that in the depth of the module.

IAM Module Dimensions (series A)

IAM Module Cat. No.	A mm (in.)	B mm (in.)	C mm (in.)	E mm (in.)	F mm (in.)
2094-AC16-M03-S	198 (7.8)	176 (7.0)	125 (4.9)	302 (11.9)	420 (16.5)
2094-AC32-M05-S			196 (7.7)		
2094-BC04-M03-S	272 (10.7)	249 (9.8)	196 (7.7)	256 (10.1)	374 (14.7)
2094-BC07-M05-S				318 (12.5)	436 (17.2)

IAM Module Dimensions (series B)

IAM Module Cat. No.	A mm (in.)	B mm (in.)	C mm (in.)	E mm (in.)	F mm (in.)
2094-BC04-M03-S	272 (10.7)	249 (9.8)	196 (7.7)	256 (10.1)	374 (14.7)
2094-BC07-M05-S					

Integrated Axis Module Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your module. For questions regarding product availability, contact your Allen-Bradley distributor.

2094 - *x* C *xx* - M *xx* - *x*

Safety Feature
S = Safe-off

Inverter Current Rating (peak of sine)
P5 = 5 A (230V input voltage)
01 = 9 A (230V input voltage)
02 = 15 A (230V input voltage)
03 = 24 A (230V input voltage)
05 = 49 A (230V input voltage)
P5 = 4 A (460V input voltage)
01 = 9 A (460V input voltage)
02 = 15 A (460V input voltage)
03 = 30 A 460V input voltage)
05 = 49 A (460V input voltage)

Inverter (Axis Module)

Converter Power Rating
05 = 3 kW (230V input voltage)
09 = 6 kW (230V input voltage)
16 = 11 kW (230V input voltage)
32 = 23 kW (230V input voltage)
01 = 6 kW (460V input voltage)
02 = 15 kW (460V input voltage)
04 = 28 kW (460V input voltage)
07 = 45 kW (460V input voltage)

Converter

Input Voltage
A = 230V AC, 50/60 Hz
B = 460V AC, 50/60 Hz

Bulletin Number

Kinetix 6000 Axis Modules

This section contains power specifications, mounting dimensions, and catalog numbers for the Bulletin 2094 (230V and 460V) axis modules (AM). Choose your AM module based on the inverter power requirements of your application.

Axis Module (inverter) Power Specifications

These specifications apply to the axis module specified in the column heading by catalog number and the same axis module (inverter section) that resides within an IAM module.

AM Module (inverter) 230V Power Specifications

Attribute	2094-AMP5-S (2094-AC05-MP5-S)	2094-AM01-S (2094-AC05-M01-S)	2094-AM02-S (2094-AC09-M02-S)	2094-AM03-S (2094-AC16-M03-S)	2094-AM05-S (2094-AC32-M05-S)
Bandwidth (1) Velocity loop Current loop	 500 Hz 1300 Hz				
PWM frequency	8 kHz		4 kHz		
Input voltage (nom)	325V DC				
Continuous current (rms)	3.7 A	6.0 A	10.6 A	17.3 A	34.6 A
Continuous current (0-pk)	5.2 A	8.5 A	15.0 A	24.5 A	48.9 A
Peak current (rms) (2)	7.4 A	12.0 A	21.2 A	34.6 A	51.9 A
Peak current (0-pk) (2)	10.5 A	17.0 A	30.0 A	48.9 A	73.4 A
Continuous power out (nom)	1.2 kW	1.9 kW	3.4 kW	5.5 kW	11.0 kW
Internal shunt Continuous power Peak power	 N/A N/A			 50 W 1400 W	
Internal shunt resistor	N/A			115 Ω	
Shunt on	N/A			405V DC	
Shunt off	N/A			375V DC	
Efficiency	98%				
Capacitance	390 μF	660 μF	780 μF	1320 μF	2640 μF
Capacitive energy absorption	15 J	25 J	29 J	50 J	99 J
Short circuit current rating	200,000 A (rms) symmetrical				

(1) Bandwidth values vary based on tuning parameters and mechanical components.

(2) Peak current duration equals 2.5 seconds.

IMPORTANT The peak current ratings of the Kinetix 6000 AM modules (series A and B) are configured at the factory as 150% of continuous current. You can program 2094-BMP5-S, 2094-BM01-S, 2094-BM02-S, and 2094-BM03-S series-B drives and their equivalent IAM (inverter) modules, up to 250% of continuous inverter current. You can program the 2094-BM05-S (AM module) and the 2094-BC07-M05-S (inverter) module up to 200% of continuous inverter current.

AM Module (inverter) 460V Power Specifications (series A and B)

Attribute	2094-BMP5-S (2094-BC01-MP5-S)	2094-BM01-S (2094-BC01-M01-S)	2094-BM02-S (2094-BC02-M02-S)	2094-BM03-S (2094-BC04-M03-S)	2094-BM05-S (2094-BC07-M05-S)
Bandwidth (1) Velocity loop Current loop	 500 Hz 1300 Hz				
PWM frequency	8 kHz		4 kHz		
Nominal input voltage	650V DC				
Continuous current (rms) (2)	2.8 A	6.1 A	10.3 A	21.2 A	34.6 A
Continuous current (sine) 0-pk (2)	4.0 A	8.6 A	14.6 A	30.0 A	48.9 A
Peak current (rms) (2) Series A drives Series B drives (3)	 4.2 A 7.0 A	 9.2 A 15.3 A	 15.5 A 25.8 A	 31.8 A 53.0 A	 51.9 A 69.2 A
Peak current (0-pk) (2) Series A drives Series B drives (3)	 5.9 A 9.9 A	 12.9 A 21.6 A	 21.8 A 36.4 A	 45.0 A 75.0 A	 73.4 A 97.9 A
Continuous power out (nom)	1.8 kW	3.9 kW	6.6 kW	13.5 kW	22.0 kW
Internal shunt Continuous power Peak power	 50 W 5.6 kW			 200 W 22.5 kW	
Internal shunt resistor	115 Ω			28.75 Ω	
Shunt on	805V DC				
Shunt off	755V DC				
Efficiency	98%				
Capacitance	75 μF	150 μF	270 μF	840 μF	1175 μF
Capacitive energy absorption	10 J	19 J	35 J	108 J	152 J
Short circuit current rating	200,000 A (rms) symmetrical				

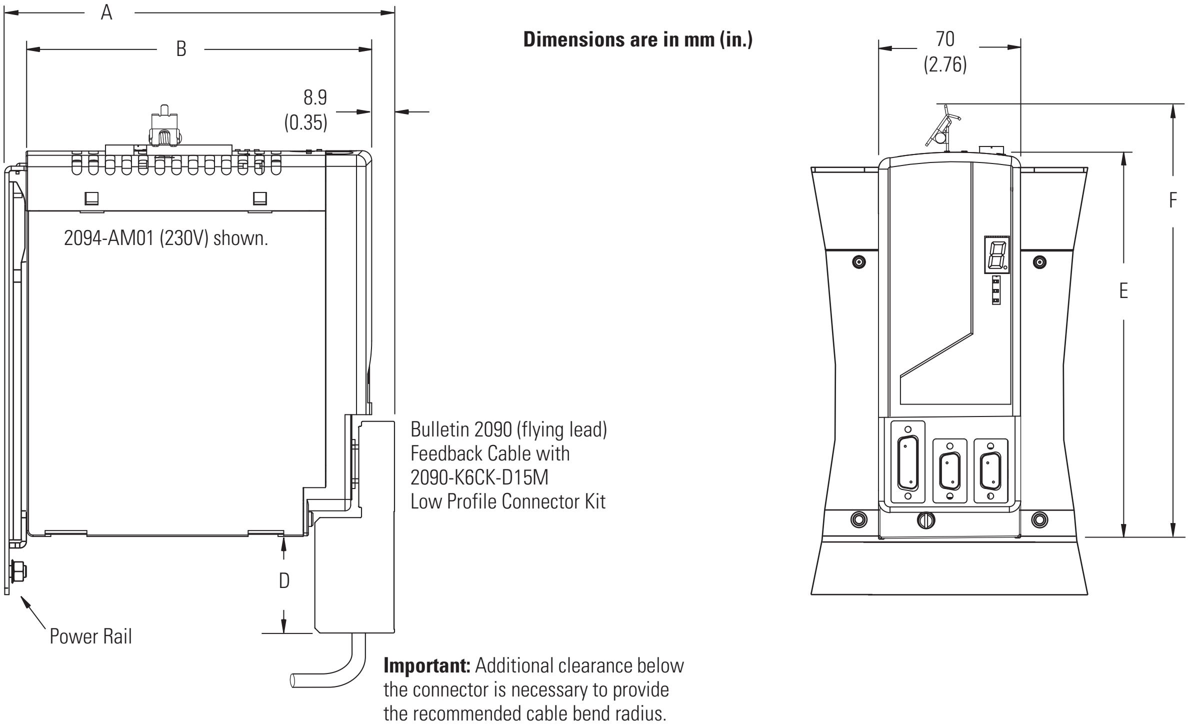
(1) Bandwidth values vary based on tuning parameters and mechanical components.

(2) Continuous and peak current ratings are for high-speed operation. For constant velocity operation at an electrical output frequency below 5 Hz (75 rpm for 8-pole motors), the output current rating is reduced. See Motion Analyzer software to correctly size your drive.

(3) Applies to series-B drives when configured for Peak-enhanced mode. For more information on drive performance in the Peak-enhanced mode, refer to Peak Enhancement Specifications on page 281.

Axis Module Dimensions

2094-AMP5-S, 2094-AM01-S, and 2094-AM02-S Dimensions (230V) 2094-BMP5-S, 2094-BM01-S, and 2094-BM02-S Dimensions (460V)

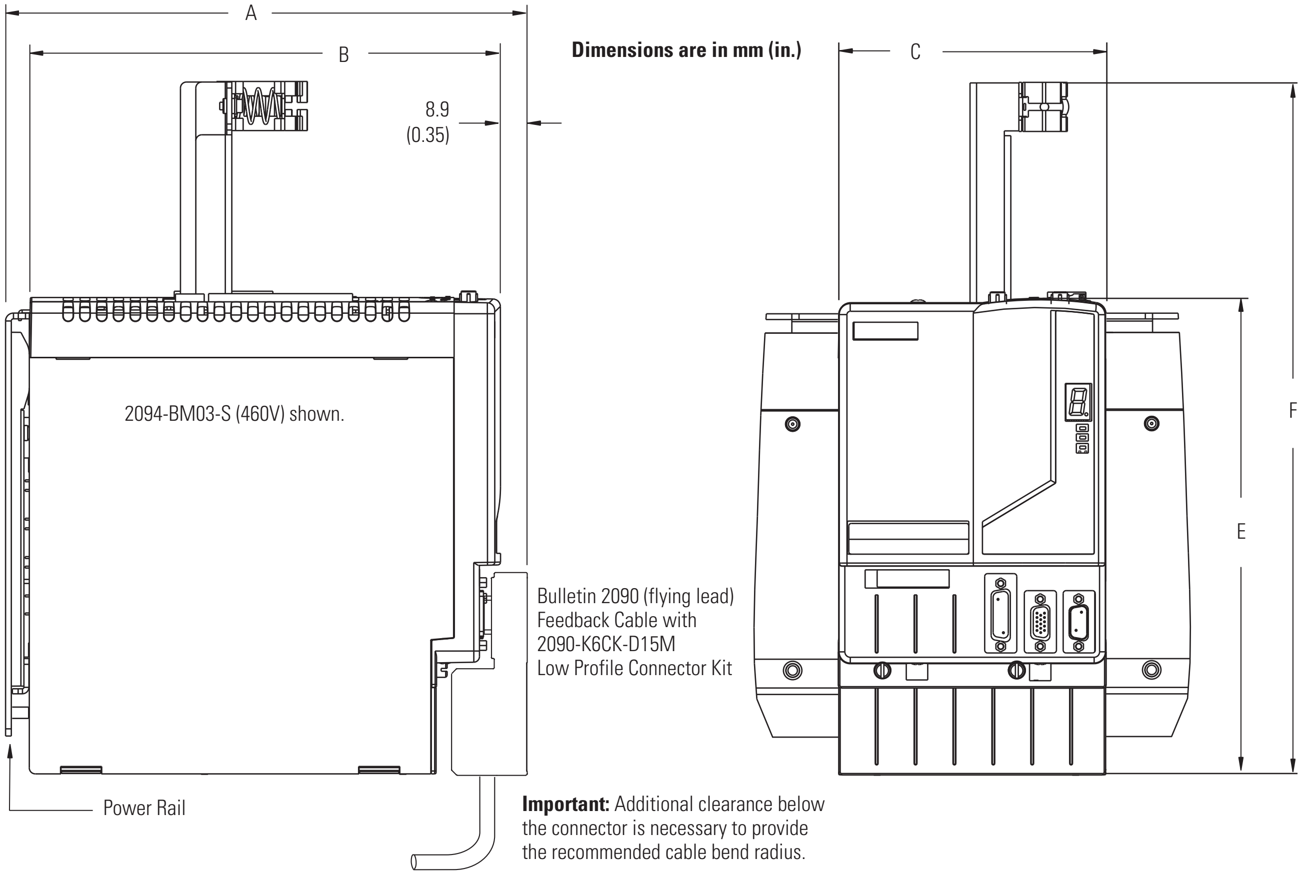


Modules are shown mounted to the power rail and the dimensions reflect that in the depth of the module.

AM Module Dimensions (series A and B)

AM Module Cat. No.	A mm (in.)	B mm (in.)	D mm (in.)	E mm (in.)	F mm (in.)
2094-AMP5-S	198 (7.8)	176 (7.0)	51 (2.0)	206 (8.2)	237 (9.3)
2094-AM01-S					
2094-AM02-S					
2094-BMP5-S	272 (10.7)	249 (9.8)	0 (0)	256 (10.1)	287 (11.3)
2094-BM01-S					
2094-BM02-S					

2094-AM03-S and 2094-AM05-S Dimensions (230V)
2094-BM03-S and 2094-BM05-S Dimensions (460V)



Dimensions are in mm (in.)

Bulletin 2090 (flying lead) Feedback Cable with 2090-K6CK-D15M Low Profile Connector Kit

Important: Additional clearance below the connector is necessary to provide the recommended cable bend radius.

Modules are shown mounted to the power rail and the dimensions reflect that in the depth of the module.

AM Module Dimensions (series A)

AM Module Cat. No.	A mm (in.)	B mm (in.)	C mm (in.)	E mm (in.)	F mm (in.)
2094-AM03-S	198 (7.8)	176 (7.0)	70 (2.8)	302 (11.9)	420 (16.5)
2094-AM05-S					
2094-BM03-S	272 (10.7)	249 (9.8)	141 (5.5)	256 (10.1)	374 (14.7)
2094-BM05-S				318 (12.5)	436 (17.2)

AM Module Dimensions (series B)

AM Module Cat. No.	A mm (in.)	B mm (in.)	C mm (in.)	E mm (in.)	F mm (in.)
2094-BM03-S	272 (10.7)	249 (9.8)	141 (5.5)	256 (10.1)	374 (14.7)
2094-BM05-S					

Axis Module Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your module. For questions regarding product availability, contact your Allen-Bradley distributor.

2094 - *x* M *xx* - *x*

Safety Feature
S = Safe-off

Inverter Current Rating (peak of sine)
P5 = 5 A (230V input voltage)
01 = 9 A (230V input voltage)
02 = 15 A (230V input voltage)
03 = 24 A (230V input voltage)
05 = 49 A (230V input voltage)
P5 = 4 A (460V input voltage)
01 = 9 A (460V input voltage)
02 = 15 A (460V input voltage)
03 = 30 A 460V input voltage)
05 = 49 A (460V input voltage)

Module Input Voltage
AM = 230V AC, 50/60 Hz
BM = 460V AC, 50/60 Hz

Bulletin Number

Kinetix 6000 General System Specifications

This section contains Kinetix 6000 drive environmental, weight, power dissipation, circuit breaker/fuse, transformer, and contactor specifications.

Environmental Specifications

Attribute	Operational Range	Storage Range (nonoperating)
Temperature, ambient	0…50 ° C (32…122 ° F)	-40…70 ° C (-40…158 ° F)
Relative humidity	5…95% noncondensing	5…95% noncondensing
Altitude	1000 m (3281 ft) 3000 m (9843 ft) with derating	3000 m (9843 ft) during transport
Vibration	5…55 Hz @ 0.35 mm (0.014 in.) double amplitude, continuous displacement; 55…500 Hz @ 2.0 g peak constant acceleration (10 sweeps in each of 3 mutually perpendicular directions).	
Shock	15 g, 11 ms half-sine pulse (3 pulses in each direction of 3 mutually perpendicular directions)	

Weight Specifications

Kinetix 6000 Module	Cat. No.	Weight, approx. kg (lb)
IAM (230V)	2094-AC05-MP5-S	2.23 (4.9)
	2094-AC05-M01-S	2.27 (5.0)
	2094-AC09-M02-S	2.31 (5.1)
	2094-AC16-M03-S	4.71 (10.4)
	2094-AC32-M05-S	7.43 (16.4)
AM (230V)	2094-AMP5-S	1.46 (3.2)
	2094-AM01-S	1.50 (3.3)
	2094-AM02-S	1.54 (3.4)
	2094-AM03-S	3.13 (6.9)
	2094-AM05-S	3.18 (7.0)
Power rails (Slim)	2094-PRS1	1.05 (2.3)
	2094-PRS2	1.59 (3.5)
	2094-PRS3	2.14 (4.7)
	2094-PRS4	2.67 (5.9)
	2094-PRS5	3.11 (6.8)
	2094-PRS6	3.55 (7.8)
	2094-PRS7	3.99 (8.8)
	2094-PRS8	4.43 (9.7)

Kinetix 6000 Module	Cat. No.	Weight, approx. kg (lb)
IAM (460V)	2094-BC01-MP5-S	4.98 (11.0)
	2094-BC01-M01-S	5.03 (11.1)
	2094-BC02-M02-S	5.08 (11.2)
	2094-BC04-M03-S	9.60 (21.1)
	2094-BC07-M05-S	10.1 (22.3)
AM (460V)	2094-BMP5-S	2.44 (5.4)
	2094-BM01-S	2.49 (5.5)
	2094-BM02-S	2.54 (5.6)
	2094-BM03-S	4.58 (10.1)
	2094-BM05-S	4.98 (11.0)
Shunt module	2094-BSP2	3.10 (6.8)
Slot-filler module	2094-PRF	0.45 (1.0)

Maximum Feedback Cable Lengths

Although motor feedback cables are available in standard lengths up to 90 m (295.3 ft), the drive/motor/feedback combination may limit the maximum feedback cable length. These tables assume the use of recommended cables as shown in the 2090-Series Motor/Actuator Cable Selection table on page 401.

Cable Lengths for Compatible Rotary Motors

Motor Cat. No.	Absolute High-resolution (5V) Encoder m (ft)	Absolute High-resolution (9V) Encoder m (ft)	Incremental/TTL (5V) Encoder m (ft)	Resolver m (ft)
MPL-A15*xxx*… MPL-A2*xxx*-E/V	30 (98.4)			
MPL-A3*xxx*… MPL-A5*xxx*-S/M [(1)]	30 (98.4)			
MPL-B15*xxx*… MPL-B2*xxx*-E/V		90 (295.3)		
MPL-B3*xxx*… MPL-B5*xxx*-S/M		90 (295.3)		
MPL-A/B15*xxx*… MPL-A/B45*xxx*-H			30 (98.4)	
MPL-B*xxxx*-R				90 (295.3)
MPM-A*xxxxx*-S/M	30 (98.4)			
MPM-B*xxxxx*-S/M		90 (295.3)		

Cable Lengths for Compatible Rotary Motors (continued)

Motor Cat. No.	**Absolute High-resolution (5V) Encoder** m (ft)	**Absolute High-resolution (9V) Encoder** m (ft)	**Incremental/TTL (5V) Encoder** m (ft)	**Resolver** m (ft)
MPM-A/B*xxxxx*-2				90 (295.3)
MPF-A*xxxx*-S/M (1)	30 (98.4)			
MPF-B*xxxx*-S/M		90 (295.3)		
MPS-A*xxxx*-S/M	30 (98.4)			
MPS-B*xxxx*-S/M		90 (295.3)		
RDB-B215*xx*-7/3	30 (98.4)			
RDB-B290*xx*-7/3 or RDB-B410*xx*-7/3	90 (295.3)			
TLY-A*xxxx*-H			30 (98.4)	

(1) MPL-A5*xxx* and MPF-A5*xxx* motor encoders are rated for 9V, the remaining Bulletin MPL and MPF (230V) motor encoders are rated for 5V.

Cable Lengths for Compatible Linear Actuators

Actuator Cat. No.	**Absolute High-resolution (5V) Encoder** m (ft)	**Absolute High-resolution (9V) Encoder** m (ft)	**Incremental/TTL (5V) Encoder** m (ft)	**Absolute High-resolution (5V) 17-bit Encoder** m (ft)
MPMA-A*xxxxx* or MPAS-A*xxxxx*-V (ballscrew)	30 (98.4)			
MPMA-A*xxxxx* or MPAS-A*xxxxx*-A (direct drive)			30 (98.4)	
MPMA-B*xxxxx* or MPAS-B*xxxxx*-V (ballscrew)		90 (295.3)		
MPMA-B*xxxxx* or MPAS-B*xxxxx*-A (direct drive)			30 (98.4)	
MPAR-A*xxxxx*-V/M	30 (98.4)			
MPAR-B*xxxxx*-V/M		90 (295.3)		
TLAR-A*xxxxx*-B				30 (98.4)
MPAI-A*xxxxx*M3	30 (98.4)			
MPAI-B*xxxxx*M3		90 (295.3)		

Cable Lengths for Compatible Linear Motors

Motor Cat. No.	**Absolute High-resolution (5V) Encoder** m (ft)	**Incremental/TTL (5V) Encoder** m (ft)
LDC-Series or LDL-Series	30 (98.4)	30 (98.4)

Maximum Power Cable Length

Although motor power cables are available in standard lengths up to 90 m (295.3 ft) and the Kinetix 6000 power rail is available in sizes up to eight axes, to meet CE requirements and improve system performance the combined motor power length for all axes on the same DC bus must not exceed 160 m (525 ft) for 230V systems and 240 m (787 ft) for 460V systems.

Circuit Breaker/Fuse Specifications

While circuit breakers offer some convenience, there are limitations for their use. Circuit breakers do not handle high current inrush as well as fuses.

Make sure the selected components are properly coordinated and meet acceptable codes including any requirements for branch circuit protection. Evaluation of the short-circuit available current is critical and must be kept below the short-circuit current rating of the circuit breaker.

Use class CC, J, L, or R fuses, with current rating as indicated in the table below. The following fuse examples and Allen-Bradley circuit breakers are recommended for use with 2094-*x*C*xx*-M*xx*-S IAM modules when the Line Interface Module (LIM) is not used.

IMPORTANT LIM Modules (catalog numbers 2094-AL*xx*S, 2094-BL*xx*S, and 2094-XL75S-C*x*) provide branch circuit protection to the IAM module. Follow all applicable NEC and local codes.

IAM Module Cat. No.	V AC Input Power			Control Input Power		DC Common Bus Fuse	
	Bussmann Fuse	Allen-Bradley Circuit Breaker		Bussmann Fuse	Allen-Bradley Circuit Breaker	Bussmann Fuse	Ferraz Shawmut Fuse
		Disconnect	Magnetic Contactor				
2094-AC05-MP5-S	KTK-R-20 (20 A)	1492-SP3D300	140M-F8E-C16	FNQ-R-10 (10 A)	1492-SP2D060	N/A	A50P20-1
2094-AC05-M01-S							
2094-AC09-M02-S	KTK-R-30 (30 A)	1492-SP3D400	140M-F8E-C20			FWH-35B	A50P35-4
2094-AC16-M03-S	LPJ-45SP (45 A)	N/A	140U-H6C3-C50		1492-SP2D200	FWH-60B	A50P60-4
2094-AC32-M05-S	LPJ-80SP (80 A)	N/A	140U-H6C3-C90			FWH-125B	A50P125-4
2094-BC01-MP5-S	KTK-R-20 (20 A)	1492-SP3D300	140M-F8E-C32		1492-SP2D060	N/A	A100P20-1
2094-BC01-M01-S							
2094-BC02-M02-S	KTK-R-30 (30 A)	1492-SP3D400	140M-F8E-C45			FWJ-40A	A100P40-1
2094-BC04-M03-S	LPJ-45SP (45 A)	N/A	140U-H6C3-C50			FWJ-70A	A100P70-1
2094-BC07-M05-S	LPJ-80SP (80 A)		140U-H6C3-C90			FWJ-125A	A100P125-1

Contactor Ratings

This table lists the recommended contactor ratings for integrated axis modules installed without a line interface module.

IAM Module (230V) Cat. No.	Contactor
2094-AC05-MP5-S	100-C23x10 (AC coil) 100-C23Zx10 (DC coil)
2094-AC05-M01-S	
2094-AC09-M02-S	100-C37x10 (AC coil) 100-C37Zx10 (DC coil)
2094-AC16-M03-S	100-C72x10 (AC coil) 100-C72Zx10 (DC coil)
2094-AC32-M05-S	100-C85x10 (AC coil) 100-C85Zx10 (DC coil)

IAM Module (460V) Cat. No.	Contactor
2094-BC01-MP5-S	100-C23x10 (AC coil) 100-C23Zx10 (DC coil)
2094-BC01-M01-S	
2094-BC02-M02-S	100-C37x10 (AC coil) 100-C37Zx10 (DC coil)
2094-BC04-M03-S	100-C60x10 (AC coil) 100-C60Zx10 (DC coil)
2094-BC07-M05-S	100-C72x10 (AC coil) 100-C72Zx10 (DC coil)

Input Transformer for Control Power

Attribute	Value
Input volt-amperes	750VA
Input voltage	460V AC
Output voltage	120…240V AC

Power Dissipation Specifications

Use this table to size an enclosure and calculate required ventilation for your Kinetix 6000 drive system.

Kinetix 6000 Modules	**Usage as % of Rated Power Output (watts)**				
	20%	**40%**	**60%**	**80%**	**100%**
IAM (converter) module (1)					
2094-AC05-MP5-S	8	11	15	19	24
2094-AC05-M01-S	9	12	16	20	25
2094-AC09-M02-S	14	20	28	36	46
2094-AC16-M03-S	19	30	43	58	74
2094-AC32-M05-S	41	68	100	136	176
2094-BC01-MP5-S	18	21	25	29	34
2094-BC01-M01-S					33
2094-BC02-M02-S	36	44	54	64	75
2094-BC04-M03-S	50	67	87	110	135
2094-BC07-M05-S	71	101	137	179	226
IAM (inverter) module or AM module (1)					
2094-AC05-MP5-S or 2094-AMP5-S	28	32	37	41	46
2094-AC05-M01-S or 2094-AM01-S	31	38	46	54	62
2094-AC09-M02-S or 2094-AM02-S	34	45	57	70	84
2094-AC16-M03-S or 2094-AM03-S	48	68	91	116	144
2094-AC32-M05-S or 2094-AM05-S	104	156	212	274	342
2094-BC01-MP5-S or 2094-BMP5-S	46	54	61	69	77
2094-BC01-M01-S or 2094-BM01-S	57	73	90	108	126
2094-BC02-M02-S or 2094-BM02-S	53	72	93	116	142
2094-BC04-M03-S or 2094-BM03-S	94	130	169	211	255
2094-BC07-M05-S or 2094-BM05-S	121	183	252	326	407
Shunt module					
2094-BSP2	68	121	174	227	280

(1) Internal shunt power is not included in the calculations and must be added based on utilization.

Power dissipation specifications are based on these calculations. This is an example:

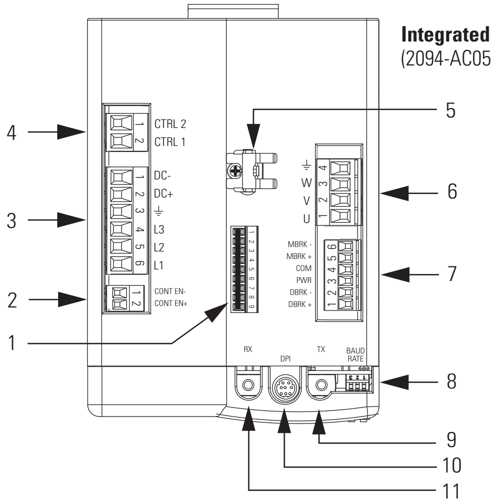
2094-BC02-M02-S with 4.52 A_{DC} (=20%) converter DC current and 10.3 A_{rms} (=100%) inverter output current.

Converter loss (36 W) + Inverter loss (142 W) = 178 W total power dissipation.

Kinetix 6000 Connector, Indicator, and Switch Locations

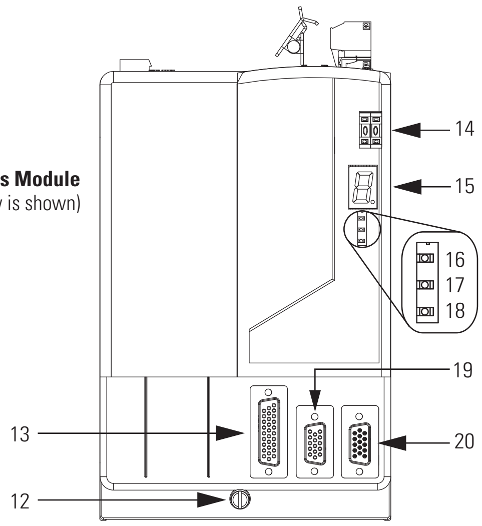
This section contains connector, indicator, and switch locations for the Kinetix 6000 IAM and AM modules.

2094-AC*xx*-M*xx*-S and 2094-BC*xx*-M*xx*-S IAM Connectors



Integrated Axis Module
(2094-AC05-M01-S top view is shown)

Integrated Axis Module
(2094-AC05-M01-S front view is shown)



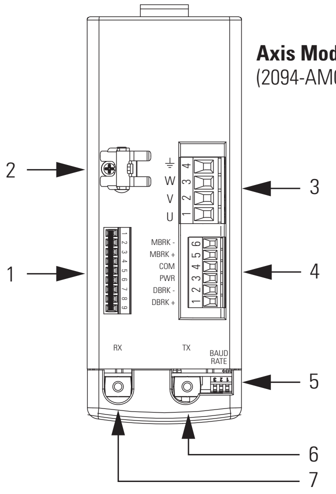
Item	Description
1	Safe-off (SO) connector
2	Contactor enable (CED) connector
3	DC bus/AC input power (IPD) connector
4	Control power (CPD) connector
5	Motor cable shield clamp
6	Motor power (MP) connector
7	Motor/resistive brake (BC) connector
8	SERCOS communication rate and optical power switches
9	SERCOS transmit (Tx) connector
10	DPI connector

Item	Description
11	SERCOS receive (Rx) connector
12	Mounting screw
13	I/O (IOD) connector
14	SERCOS node address switch
15	Seven-segment fault status indicator
16	Drive status indicator
17	COMM status indicator
18	Bus status indicator
19	Motor feedback (MF) connector
20	Auxiliary feedback (AF) connector

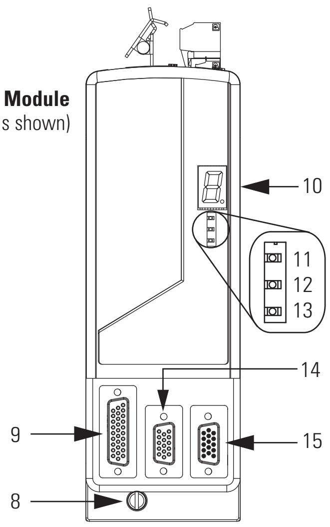
For connector kit options, refer to Breakout Components and Connector Kits beginning on page 440.

2094-AM*xx*-S and 2094-BM*xx*-S AM Connectors

Axis Module
(2094-AM01-S top view is shown)



Axis Module
(2094-AM01-S front view is shown)



Item	Description	Item	Description	Item	Description
1	Safe-off (SO) connector	6	SERCOS transmit (Tx) connector	11	Drive status indicator
2	Motor cable shield clamp	7	SERCOS receive (Rx) connector	12	COMM status indicator
3	Motor power (MP) connector	8	Mounting screw	13	Bus status indicator
4	Motor/resistive brake (BC) connector	9	I/O (IOD) connector	14	Motor feedback (MF) connector
5	SERCOS communication rate and optical power switches	10	Seven-segment fault status indicator	15	Auxiliary feedback (AF) connector

For connector kit options, refer to Breakout Components and Connector Kits beginning on page 440.

Kinetix 2000 Multi-axis Servo Drives

Extend the benefits of Kinetix Integrated Motion to low-power motion control applications with the Kinetix 2000 servo drive. This multi-axis servo drive provides simplicity at its best, letting you save time and money from initial wiring and programming to operation and diagnostics. With a continuous output current (rms) from 1.0…9.5 amps, the Kinetix 2000 offers the same compact design, exceptional performance, and cost saving features as the Kinetix 6000 servo drives. The commonality among the Kinetix drives will let you to learn once and reuse your product knowledge. Paired with the CompactLogix 1768-L4*x* controller, the Kinetix 2000 is ideal for small and mid-sized applications looking to improve productivity, quality, and time to market while reducing the total cost of ownership.

The Kinetix 2000 multi-axis servo drives are part of the Kinetix Integrated Motion solution.

Kinetix 2000 Servo Drive Components

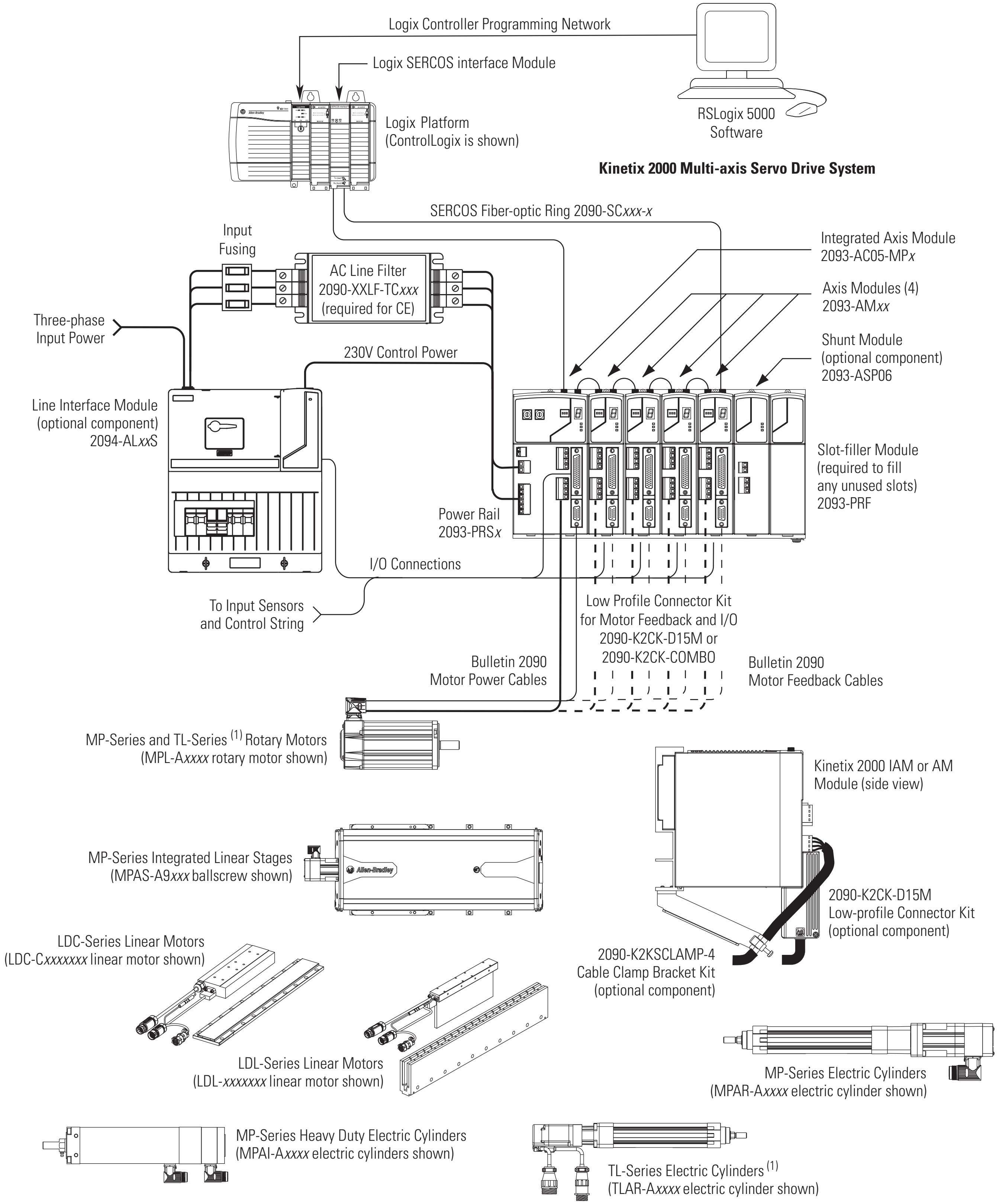
Kinetix 2000 servo drive systems consist of these required components:

- One integrated axis module (IAM or leader IAM), 2093-AC05-MP*x*
- Up to seven axis modules, 2093-AM*xx*
- One power rail, 2093-PRS1, 2093-PRS2, 2093-PRS3, 2093-PRS4, 2093-PRS5, 2093-PRS7, or 2093-PRS8S
- One to eight rotary/linear motors or linear actuators (MP-Series, TL-Series, LDC-Series, or LDL-Series)
- One to eight motor power and feedback cables
- Two to nine SERCOS fiber-optic cables

Kinetix 2000 systems may also include any of these optional components:

- One or more integrated axis modules used as a follower IAM (2093-AC05-MP*x*) and associated axis modules, power rails, motors, and cables as required for the application
- One shunt module, 2093-ASP06
- Slot-filler modules, 2093-PRF
- Bulletin 2094 Line Interface Module (LIM), 2094-AL*xx*S, 2094-XL75S-C2, or 2094-AL09

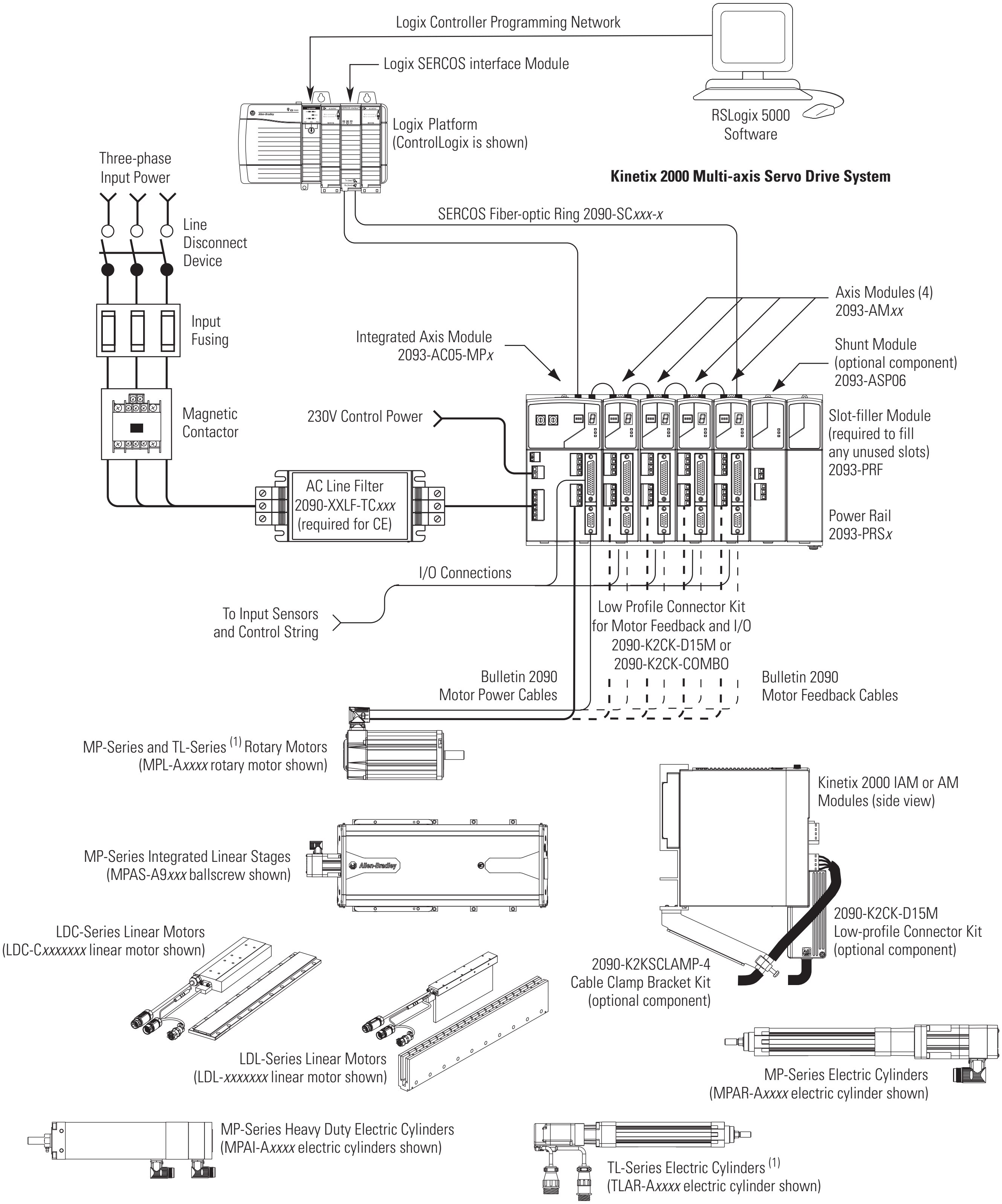
Typical Configuration - Kinetix 2000 System (with LIM module)



---

(1) TL-Series (Bulletin TLY) rotary motors and Bulletin TLAR electric cylinders require the 2090-K2CK-D15M connector kit with 2090-DA-BAT2 battery for multi-turn high-resolution encoder operation. Other Kinetix 2000 compatible motors and actuators require the connector kit for flying-lead feedback connections, but not the battery.

Typical Configuration - Kinetix 2000 System (without LIM module)



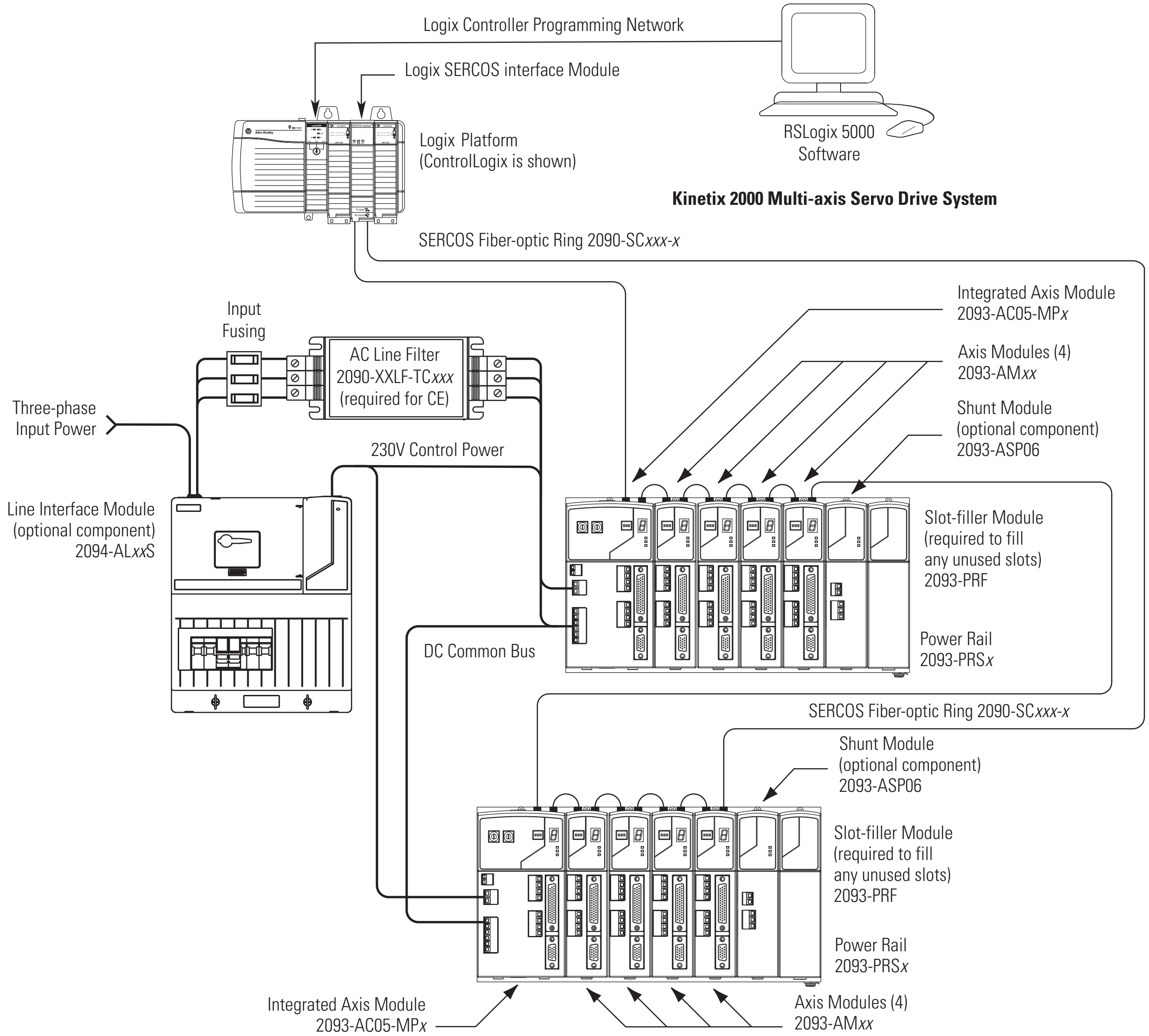
---

(1) TL-Series (Bulletin TLY) rotary motors and Bulletin TLAR electric cylinders require the 2090-K2CK-D15M connector kit with 2090-DA-BAT2 battery for multi-turn high-resolution encoder operation. Other Kinetix 2000 compatible motors and actuators require the connector kit for flying-lead feedback connections, but not the battery.

In the Kinetix 2000 drive configuration below, the leader IAM is connected to the follower IAM via the DC common bus. When planning your panel layout, you must calculate the total bus capacitance of your DC common bus system to make sure that the leader IAM is sized sufficiently to pre-charge the entire system. Refer to the Kinetix 2000 Servo Drive User Manual, publication 2093-UM001, when making this calculation.

IMPORTANT	If total bus capacitance of your system exceeds the leader IAM pre-charge rating, the IAM seven-segment status indicator will display error code E90 (pre-charge timeout fault) if input power is applied. To correct this condition, you must remove axis modules from the power rail to decrease the total bus capacitance.

Typical Configuration - Kinetix 2000 System (DC common bus)



Motors and other details common to both three-phase AC and DC common-bus configurations are removed.

Kinetix 2000 Integrated Axis Modules

This section contains power specifications, mounting dimensions, and catalog numbers for the 2093-AC05-MP1, 2093-AC05-MP2, and 2093-AC05-MP5 integrated axis modules (IAM). The converter section of these modules is identical. Choose your IAM module based on the inverter requirements of your application.

Integrated Axis Module (converter) Power Specifications

IAM (three-phase and single-phase) Power Specifications

Attribute	**Value**	
	2093-AC05-MP*x* **Three-phase Input (230V nom)**	**2093-AC05-MP*x*** **Single-phase Input (230V nom)**
AC input voltage	170…264V rms	
AC input frequency	47…63 Hz	
Main AC input current (1) Nom (rms) Max inrush (0-pk)	11.66 A 34.0 A	10.95 A 34.0 A
DC input voltage (common bus follower)	240…375V DC	
DC input current (common bus follower)	9.76 A	6.42 A
Control power AC input voltage	170…264V rms single-phase (230V nom)	
Control power AC input current Nom (@ 230V AC) rms Max inrush (0-pk)	1.25 A 93.0 A (2)	
Nominal bus output voltage	325V DC	
Line loss ride through	20 ms	
Continuous output current to bus (A_{DC})	9.67 A	6.42 A
Peak output current to bus (A_{DC}) (3)	19.34 A	12.84 A
Bus overvoltage	415V DC	
Bus undervoltage	135.5V DC	
Internal shunt Continuous power Peak power	15 W 3000 W	
Internal shunt resistor	50 Ω	
Shunt on	405V	
Shunt off	375V	
Continuous power output to bus	3.0 kW	2.0 kW
Peak power output	6.0 kW	4.0 kW
Efficiency	95%	
Converter inductance	N/A	
Converter capacitance	540 μF	
Converter leakage current (max)	2.0 mA	

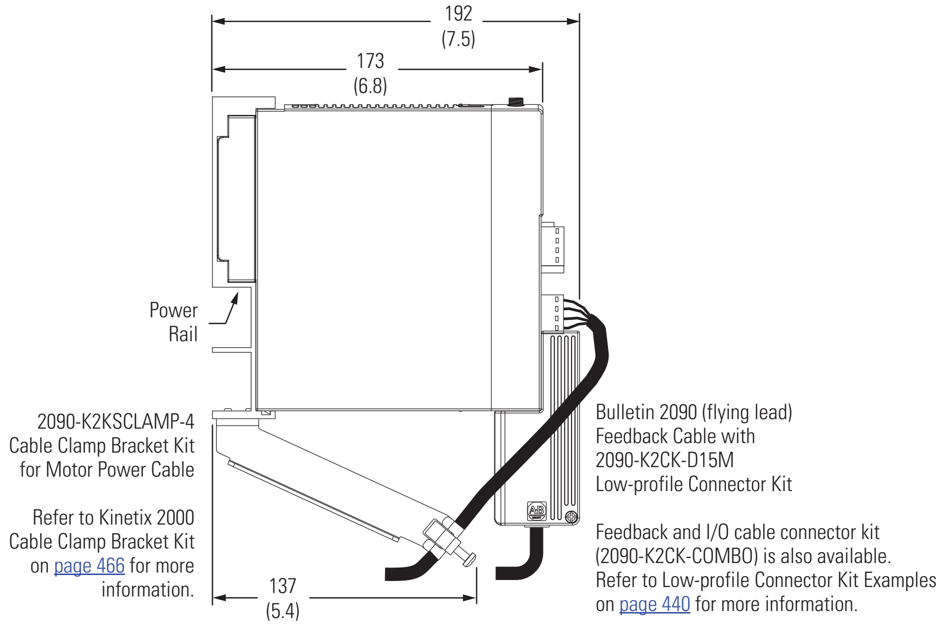
(1) All 2093-AC05 integrated axis modules are limited to 2 contactor cycles per minute (with up to 4 axis modules), or 1 contactor cycle per minute (with 5…8 axis modules).
(2) Maximum inrush duration is less than 1/2 line cycle.
(3) Peak output current duration equals 250 ms.

Control Power Current Requirements

Modules on Power Rail	220/230V AC Input A	Input VA VA
IAM only	0.15	50
IAM, 1 AM	0.30	99
IAM, 2 AM	0.45	148
IAM, 3 AM	0.60	197
IAM, 4 AM	0.75	247
IAM, 5 AM	0.90	296
IAM, 6 AM	1.05	345
IAM, 7 AM	1.20	395
IAM, 7 AM, 1 Shunt module	1.25	410

Integrated Axis Module Dimensions

2093-AC05-MP1, 2093-AC05-MP2, and 2093-AC05-MP5 Dimensions



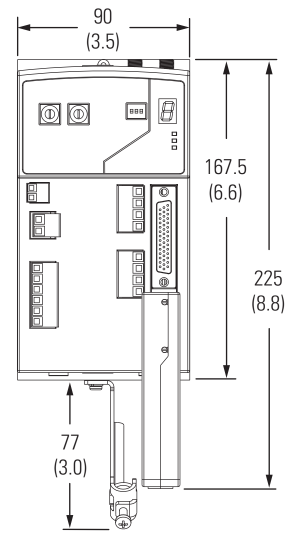
2090-K2KSCLAMP-4
Cable Clamp Bracket Kit
for Motor Power Cable

Refer to Kinetix 2000
Cable Clamp Bracket Kit
on page 466 for more
information.

Bulletin 2090 (flying lead)
Feedback Cable with
2090-K2CK-D15M
Low-profile Connector Kit

Feedback and I/O cable connector kit
(2090-K2CK-COMBO) is also available.
Refer to Low-profile Connector Kit Examples
on page 440 for more information.

Dimensions are in mm (in.)

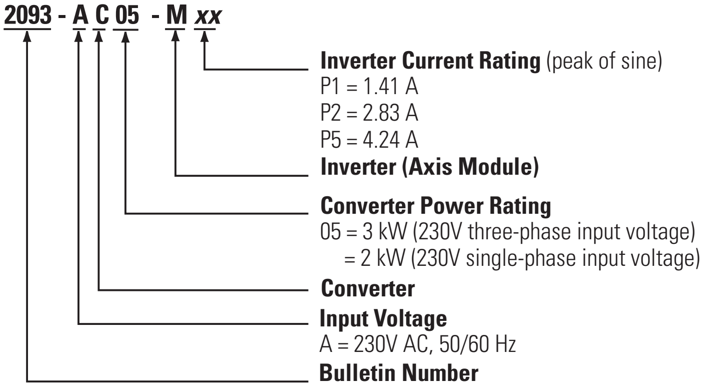


Important: Additional clearance below the connectors is necessary to provide the recommended cable bend radius.

Modules are shown mounted to the power rail and the dimensions reflect that in the height and depth of the module.

Integrated Axis Module Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your module. For questions regarding product availability, contact your Allen-Bradley distributor.



2093 - A C 05 - M *xx*

Inverter Current Rating (peak of sine)
P1 = 1.41 A
P2 = 2.83 A
P5 = 4.24 A
Inverter (Axis Module)
Converter Power Rating
05 = 3 kW (230V three-phase input voltage)
= 2 kW (230V single-phase input voltage)
Converter
Input Voltage
A = 230V AC, 50/60 Hz
Bulletin Number

Kinetix 2000 Axis Modules

This section contains power specifications, mounting dimensions, and catalog numbers for the 2093-AM*xx* axis modules (AM). Choose your AM based on the inverter power requirements of your application.

Axis Module (inverter) Power Specifications

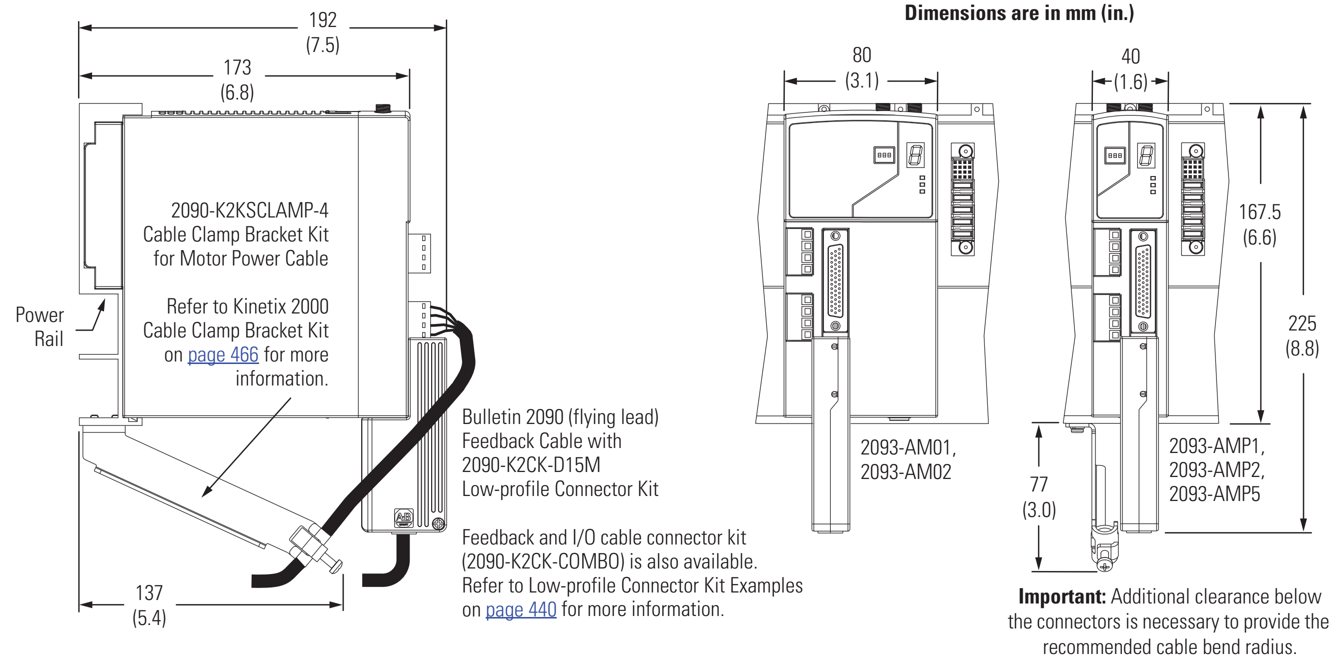
These specifications apply to the axis module specified in the column heading by catalog number and the same axis module (inverter section) that resides within an integrated axis module.

Attribute	**2093-AMP1** (2093-AC05-MP1)	**2093-AMP2** (2093-AC05-MP2)	**2093-AMP5** (2093-AC05-MP5)	**2093-AM01**	**2093-AM02**
Bandwidth [1] Velocity Loop Current Loop	 500 Hz 1300 Hz				
PWM frequency	8 kHz				
Nominal input voltage	325V DC				
Continuous current (rms)	1.0 A	2.0 A	3.0 A	6.0 A	9.5 A
Continuous current (0-pk)	1.41 A	2.83 A	4.24 A	8.48 A	13.4 A
Peak current (rms)	3.0 A	6.0 A	9.0 A	18.0 A	28.5 A
Peak current (0-pk)	4.20 A	8.48 A	12.7 A	25.5 A	40.3 A
Peak output current time (max)	3 s from 0% drive utilization (0% soak)				
Continuous power out (nom)	0.3 kW	0.6 kW	0.9 kW	1.9 kW	3.0 kW
Efficiency	98%				
Capacitance	200 µF			540 µF	
Capacitive energy absorption	7.5 J			20 J	
Inverter PCB leakage current	1 mA				

(1) Bandwidth values vary based on tuning parameters and mechanical components.

Axis Module Dimensions

2093-AMP1, 2093-AMP2, 2093-AMP5, 2093-AM01, and 2093-AM02 Dimensions



2090-K2KSCLAMP-4 Cable Clamp Bracket Kit for Motor Power Cable

Refer to Kinetix 2000 Cable Clamp Bracket Kit on page 466 for more information.

Bulletin 2090 (flying lead) Feedback Cable with 2090-K2CK-D15M Low-profile Connector Kit

Feedback and I/O cable connector kit (2090-K2CK-COMBO) is also available. Refer to Low-profile Connector Kit Examples on page 440 for more information.

Important: Additional clearance below the connectors is necessary to provide the recommended cable bend radius.

Modules are shown mounted to the power rail and the dimensions reflect that in the depth of the module.

Axis Module Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your module. For questions regarding product availability, contact your Allen-Bradley distributor.

2093 - AM *xx*

Inverter Current Rating (peak of sine)
P1 = 1.41 A
P2 = 2.83 A
P5 = 4.24 A
01 = 8.48 A
02 = 13.4 A

Module Input Voltage
AM = 230V AC, 50/60 Hz

Bulletin Number

Kinetix 2000 Power Rail

This section contains selection information, mounting dimensions, and catalog numbers for the 2093-PRS*x* power rails.

Kinetix 2000 IAM, AM, and Shunt Module Slot Requirements

IAM Cat. No.	Converter Slot Used	Inverter Slots Used
2093-AC05-MP1	1	1
2093-AC05-MP2		1
2093-AC05-MP5		1

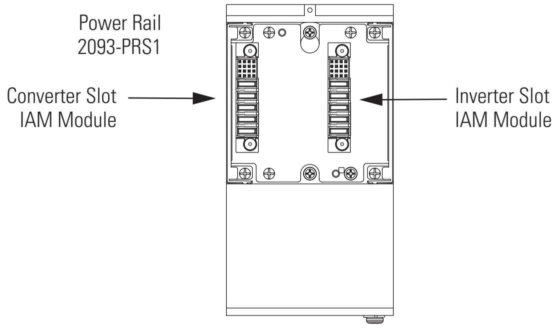
AM Cat. No.	Converter Slot Used	Inverter Slots Used
2093-AMP1	0	1
2093-AMP2		1
2093-AMP5		1
2093-AM01		2 (1)
2093-AM02		2

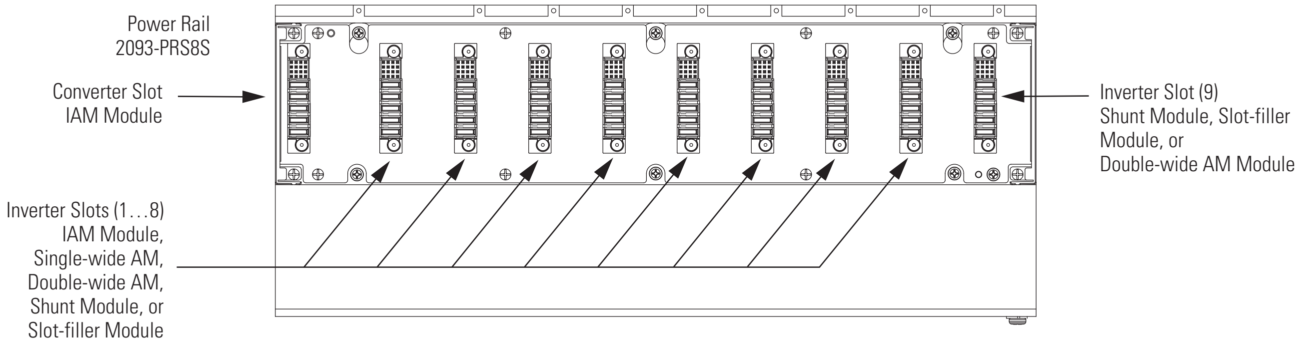
Shunt Module Cat. No.	Converter Slot Used	Inverter Slots Used
2093-ASP06	0	1

(1) 2093-AM01 and 2093-AM02 axis modules are double-wide modules and require two slots on the power rail.

The far-left slot on each power rail is the converter slot and used only by the IAM. All other slots are inverter slots and are used by the IAM, AM, or shunt module (refer to the figure below). The power rail catalog number indicates the maximum number axes that each power rail will hold.

Power Rail Slots





For example, the 2093-PRS1 power rail contains one inverter slot. This limits the use of this power rail to systems requiring only one inverter slot. Similarly, the 2093-PRS2 power rail contains two inverter slots. This limits the use of this power rail to systems requiring up to two inverter slots. When selecting a power rail, determine the number of inverter slots required by all rail-mounted modules and choose a power rail with that minimum number of inverter slots.

IMPORTANT If you select a power rail with slots exceeding the minimum required for your system, you must install a 2093-PRF slot-filler module in each unused slot.

The 2093-PRS8S power rail is unique in that it accommodates eight axes, but has nine inverter slots. The far-right (ninth) inverter slot is reserved for a shunt module, but could also be occupied by a slot-filler module or double-wide axis module. These power rail configurations are supported.

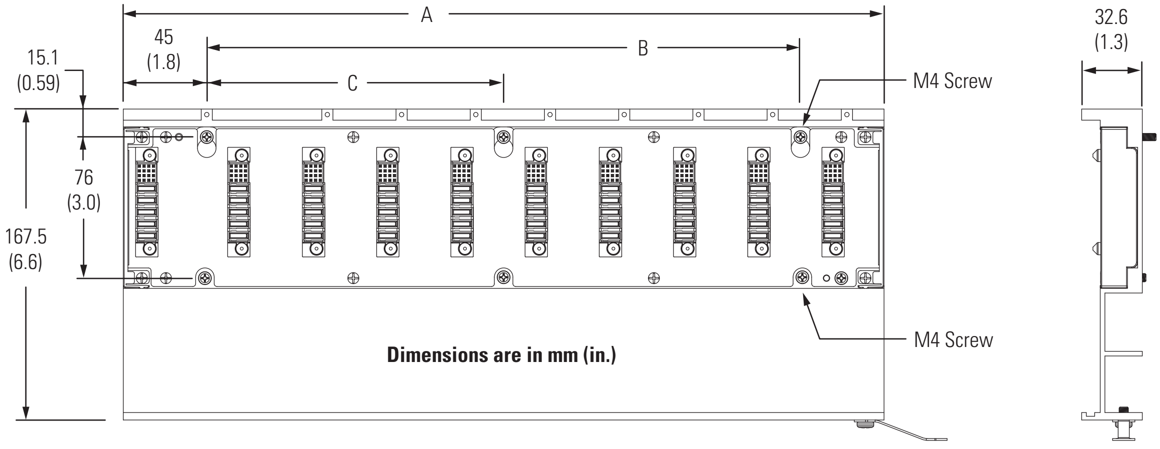
2093-PRS8S Configurations

8-axis		7-axis		6-axis		5-axis		
IAM	IAM	IAM	IAM	IAM	IAM	IAM	IAM	IAM
AM	AM	AM (double-wide)	AM (double-wide)	AM (double-wide)	AM (double-wide)	AM (double-wide)	AM (double-wide)	AM (double-wide)
AM	AM							
AM	AM	AM	AM	AM (double-wide)	AM (double-wide)	AM (double-wide)	AM (double-wide)	AM (double-wide)
AM	AM	AM	AM					
AM	AM	AM	AM	AM	AM	AM (double-wide)	AM (double-wide)	AM (double-wide)
AM	AM	AM	AM	AM	AM			
AM	AM	AM	AM	AM	AM	AM (double-wide)	AM	AM
Shunt module	Slot-filler module	Shunt module	Slot-filler module	Shunt module	Slot-filler module		Shunt module	Slot-filler module

IMPORTANT The maximum number of axes supported by the 2093-PRS8S power rail is eight axes. Do not install an axis module (AM) in the far-right (ninth) inverter slot.

Power Rail Dimensions

2093-PRS1, 2093-PRS2, 2093-PRS3, 2093-PRS4, 2093-PRS5, 2093-PRS7, and 2093-PRS8S Dimensions



Dimensions are in mm (in.)

Power Rail Cat. No.	Description	Dimension A mm (in.)	Dimension B mm (in.)	Dimension C mm (in.)
2093-PRS1	1 axis power rail	90 (3.54)	N/A	N/A
2093-PRS2	2 axis power rail	130 (5.12)	40 (1.57)	N/A
2093-PRS3	3 axis power rail	170 (6.69)	80 (3.15)	N/A
2093-PRS4	4 axis power rail	210 (8.26)	120 (4.72)	N/A
2093-PRS5	5 axis power rail	250 (9.84)	160 (6.30)	N/A
2093-PRS7	7 axis power rail	330 (12.99)	240 (9.45)	120 (4.72)
2093-PRS8S	8 axis power rail	410 (16.14)	320 (12.60)	160 (6.30)

Power Rail Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your power rail. For questions regarding product availability, contact your Allen-Bradley distributor.

2093 - PRS *x*

Module Capacity
1 = IAM and no additional modules
2 = IAM and 1 additional module
3 = IAM and up to 2 additional modules
4 = IAM and up to 3 additional modules
5 = IAM and up to 4 additional modules
7 = IAM and up to 6 additional modules
8S = IAM and up to 8 additional modules

Power Rail
PRS = Power Rail (slim), available in 1, 2, 3, 4, 5, 7, and 8 axis capacity

Bulletin Number

Kinetix 2000 Shunt Module

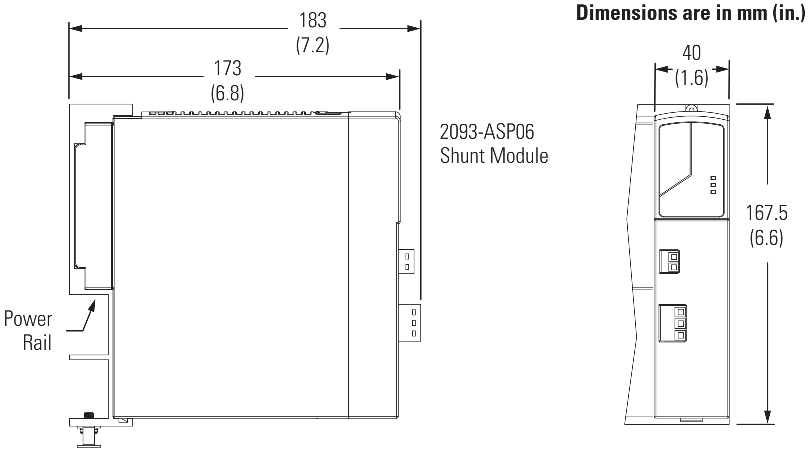
This section contains specifications, mounting dimensions, and catalog numbers for the 2093-ASP06 Shunt Module.

Shunt Module Power Specifications

Drive Cat. No.	Shunt Module Cat. No.	Specifications						Fuse Replacement
		Drive Voltage V AC	Resistance Ω	Peak Power kW	Peak Current A	Continuous Power W	Capacitance µF	
2093-AC05-MP1	2093-ASP06	230	15.0	10.9	27.0	50	164	N/A (no internal fuse)
2093-AC05-MP2								
2093-AC05-MP5								

For specifications and dimensions of external shunt resistors compatible with the Kinetix 2000 drive, refer to External Shunt Modules beginning on page 492.

Shunt Module Dimensions



Modules are shown mounted to the power rail and the dimensions reflect that.

Shunt Module Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your module. For questions regarding product availability, contact your Allen-Bradley distributor.

2093 - ASP06

Input Voltage/kW Rating
ASP06 = 230V input voltage, 50 W

Bulletin Number

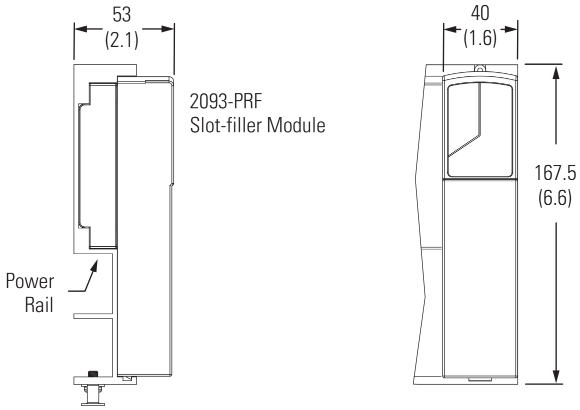
Kinetix 2000 Slot-filler Module

This section contains dimensions and catalog numbers for the 2093-PRF slot-filler module.

IMPORTANT The Kinetix 2000 slot-filler module (catalog number 2093-PRF) is compatible with all Kinetix 2000 systems. Power rail slots not occupied by an IAM, AM, or shunt module, must have a slot-filler module installed.

Slot-filler Module Dimensions

Dimensions are in mm (in.)



Slot-filler Module Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your module. For questions regarding product availability, contact your Allen-Bradley distributor.

2093 - PRF
Power Rail Filler
Bulletin Number

Kinetix 2000 General System Specifications

This section contains Kinetix 2000 environmental, weight, power dissipation, circuit breaker/fuse, transformer, and contactor specifications.

Environmental Specifications

Attribute	Operational Range	Storage Range (nonoperating)
Temperature, ambient	0…50 °C (32…122 °F)	-40…85 °C (-40…185 °F)
Relative humidity	5…95% noncondensing	5…95% noncondensing
Altitude	1000 m (3281 ft) 3000 m (9843 ft) with derating	3000 m (9843 ft) during transport
Vibration	5…55 Hz @ 0.35 mm (0.014 in.) double amplitude, continuous displacement; 55…500 Hz @ 2.0 g peak constant acceleration (10 sweeps in each of 3 mutually perpendicular directions)	
Shock	15 g, 11 ms half-sine pulse (3 pulses in each direction of 3 mutually perpendicular directions)	

Weight Specifications

Kinetix 2000 Module	Cat. No.	Weight, approx. kg (lb)
IAM	2093-AC05-MP1	1.32 (2.9)
	2093-AC05-MP2	
	2093-AC05-MP5	
AM	2093-AMP1	0.67 (1.5)
	2093-AMP2	
	2093-AMP5	
	2093-AM01	0.95 (2.1)
	2093-AM02	
Shunt module	2093-ASP06	0.59 (1.3)

Kinetix 2000 Module	Cat. No.	Weight, approx. kg (lb)
Power Rails (Slim)	2093-PRS1	0.27 (0.6)
	2093-PRS2	0.38 (0.8)
	2093-PRS3	0.51 (1.1)
	2093-PRS4	0.64 (1.4)
	2093-PRS5	0.77 (1.7)
	2093-PRS7	1.03 (2.3)
	2093-PRS8S	1.28 (2.8)
Slot-filler module	2093-PRF	0.15 (0.3)

Maximum Feedback Cable Length

Although motor feedback cables are available in standard lengths up to 90 m (295.3 ft), the Kinetix 2000 drive maximum feedback cable length is 30 m (98.4 ft). These tables assume the use of recommended cables as shown in the 2090-Series Motor/Actuator Cable Selection table on page 401.

Cable Lengths for Compatible Rotary Motors

Motor Cat. No.	**Absolute High-resolution (5V) Encoder** m (ft)	**Incremental/TTL (5V) Encoder** m (ft)	**Absolute High-resolution (5V) 17-bit Encoder** m (ft)
MPL-A3*xxx*… MPL-A5*xxx*-S/M (1)	30 (98.4)		
MPL-A15*xxx*… MPL-A2*xxx*-E/V	30 (98.4)		
MPL-A15*xxx*… MPL-A45*xxx*-H		30 (98.4)	
MPM-A*xxxxx*-S/M	30 (98.4)		
MPF-A*xxxx*-S/M (1)	30 (98.4)		
MPS-A*xxxx*-S/M	30 (98.4)		
TLY-A*xxxx*-B			30 (98.4)
TLY-A*xxxx*-H		30 (98.4)	

(1) MPL-A5*xxx* and MPF-A5*xxx* motor encoders are rated for 9V, the remaining Bulletin MPL and MPF (230V) motor encoders are rated for 5V.

Cable Lengths for Compatible Linear Actuators

Actuator Cat. No.	**Absolute High-resolution (5V) Encoder** m (ft)	**Incremental/TTL (5V) Encoder** m (ft)	**Absolute High-resolution (5V) 17-bit Encoder** m (ft)
MPMA-A*xxxxx* or MPAS-A*xxxxx*-V (ballscrew)	30 (98.4)		
MPMA-A*xxxxx* or MPAS-A*xxxxx*-A (direct drive)		30 (98.4)	
MPAR-A*xxxxx*-V/M	30 (98.4)		
TLAR-A*xxxxx*-B			30 (98.4)
MPAI-A*xxxxx*M3	30 (98.4)		

Cable Lengths for Compatible Linear Motors

Motor Cat. No.	**Absolute High-resolution (5V) Encoder** m (ft)	**Incremental/TTL (5V) Encoder** m (ft)
LDC-Series or LDL-Series	30 (98.4)	30 (98.4)

Maximum Power Cable Length

Although motor power cables are available in standard lengths up to 90 m (295.3 ft) and the Kinetix 2000 power rail is available in sizes up to eight axes, to meet CE requirements and improve system performance the combined motor power length for all axes on the same DC bus must not exceed 160 m (525 ft).

IMPORTANT Operating the Kinetix 2000 drive at maximum temperature with maximum cable length may necessitate derating of the drive.

Circuit Breaker/Fuse Specifications

While circuit breakers offer some convenience, there are limitations for their use. Circuit breakers do not handle high current inrush as well as fuses.

Make sure the selected components are properly coordinated and meet acceptable codes including any requirements for branch circuit protection. Evaluation of the short-circuit available current is critical and must be kept below the short-circuit current rating of the circuit breaker.

Use class CC, J, L, or R fuses, with current rating as indicated in the table below. The following fuse examples and Allen-Bradley circuit breakers are recommended for use with the 2093-*x*C*xx*-M*xx* IAM modules when the Line Interface Module (LIM) is not used.

IMPORTANT 2094-AL*xx*S and 2094-XL75S-C2 LIM modules provide branch circuit protection to the IAM module. Follow all applicable NEC and local codes.

Fuse Specifications

Drive Cat. No.	V AC Input Power				Control Input Power		DC Common Bus
	Input Voltage	Bussmann Fuse	Allen-Bradley Circuit Breaker		Bussmann Fuse	Allen-Bradley Circuit Breaker	Ferraz Shawmut Fuse
2093-AC05-MP1	170…264 AC three-phase	KTK-R-20 (20 A)	1492-SP3D300	140M-F8E-C16	FNQ-R-10 (10 A) Class CC or LPJ-10SP (10 A) Class J	1492-CB2H060	A50P20-1
2093-AC05-MP2							
2093-AC05-MP5							
2093-AC05-MP1	170…264 AC single-phase	KTK-R-20 (20 A)	1492-SP2D300	N/A			A50P20-1
2093-AC05-MP2							
2093-AC05-MP5							

Power Dissipation Specifications

Use this table to size an enclosure and calculate required ventilation for your Kinetix 2000 drive system.

Kinetix 2000 Modules		**Usage as % of Rated Power Output** (watts)				
		20%	**40%**	**60%**	**80%**	**100%**
Integrated Axis Module (IAM Converter) [1]						
2093-AC05-MP1	Three-phase input	7.0	10.5	14.0	17.4	20.9
2093-AC05-MP2						
2093-AC05-MP5						
2093-AC05-MP1	Single-phase input	5.8	8.0	10.3	12.6	14.8
2093-AC05-MP2						
2093-AC05-MP5						
Integrated Axis Module (IAM Inverter) or Axis Module (AM) [1]						
2093-AC05-MP1 or 2093-AMP1		31.6	33.6	35.6	37.6	39.6
2093-AC05-MP2 or 2093-AMP2		33.0	36.4	39.8	43.3	46.8
2093-AC05-MP5 or 2093-AMP5		36.2	42.9	49.8	56.8	63.9
2093-AM01		38.3	46.7	55.3	64.1	73.1
2093-AM02		44.3	55.6	67.3	79.2	91.4
Shunt Module						
2093-ASP06		35.8	45.8	55.8	65.8	75.8

(1) Internal shunt power is not included in the calculations and must be added based on utilization.

Transformer Specifications for Control Power Input

You can use any general purpose transformer with these ratings.

Attribute	**Value**
Input volt-amperes	500VA
Output voltage	200…240V AC

Contactor Ratings

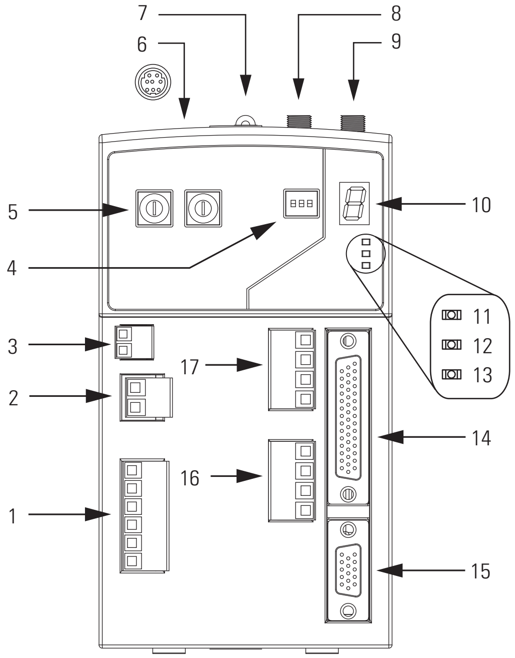
The table below lists the recommended contactor ratings for integrated axis modules installed without a line interface module.

IAM Cat. No.	**Input Voltage**	**Contactor**
2093-AC05-MP1	170…264 AC single-phase or three-phase operation	100-C23x10 (AC coil) 100-C23Zx10 (DC coil)
2093-AC05-MP2		
2093-AC05-MP5		

Kinetix 2000 Connector, Indicator, and Switch Locations

This section contains connector, indicator, and switch locations for the Kinetix 2000 Integrated Axis Module (IAM), Axis Module (AM), and Shunt Module.

2093-AC05-MP*x* IAM Connectors

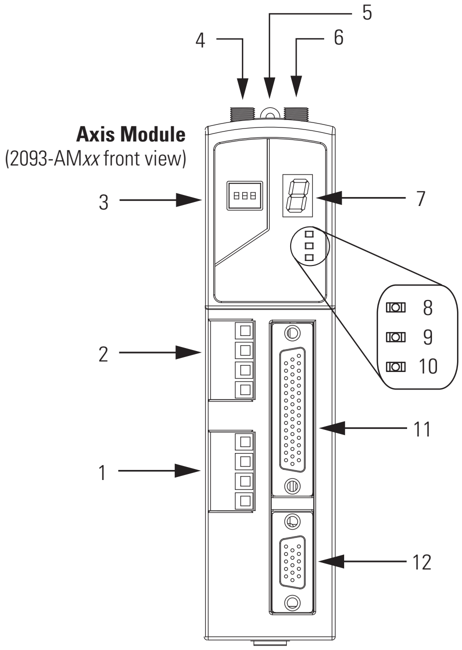


Integrated Axis Module
(2093-AC05-MP*x* front view)

Item	Description
1	DC Bus / AC input power (IPD) connector
2	Control power (CPD) connector
3	Contactor enable (CED) connector
4	SERCOS communication rate and optical power switches
5	SERCOS node address switch
6	DPI connector (facing up)
7	Mounting screw
8	SERCOS receive (Rx) connector
9	SERCOS transmit (Tx) connector
10	Seven-segment fault status indicator
11	Drive status indicator
12	COMM status indicator
13	Bus status indicator
14	I/O and auxiliary feedback (IOD/AF) connector
15	Motor feedback (MF) connector
16	Motor power (MP) connector
17	Motor brake (BC) connector

For motor feedback connector kit catalog numbers, refer to Low-profile Connector Kit Components on page 441. For replacement connector sets, refer to Connector Sets on page 463.

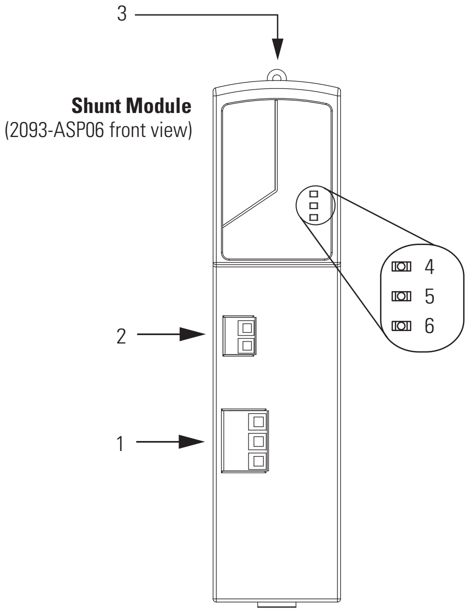
2093-AM*xx* AM Connectors



Item	Description
1	Motor power (MP) connector
2	Motor brake (BC) connector
3	SERCOS communication rate and optical power switches
4	SERCOS receive (Rx) connector
5	Mounting screw
6	SERCOS transmit (Tx) connector
7	Seven-segment fault status indicator
8	Drive status indicator
9	COMM status indicator
10	Bus status indicator
11	I/O and auxiliary feedback (IOD/AF) connector
12	Motor feedback (MF) connector

For motor feedback connector kit catalog numbers, refer to Low-profile Connector Kit Components on page 441. For replacement connector sets, refer to Connector Sets on page 463.

2093-ASP06 Shunt Module Connectors



Item	Description
1	External shunt resistor (RC) connector
2	External thermal switch (TS) connector
3	Mounting screw
4	Shunt fault status indicator
5	Over-temp fault status indicator
6	Bus status indicator

For replacement connector sets, refer to Connector Sets on page 463.

Kinetix 7000 High Power Servo Drives

The Kinetix 7000 high power servo drive is designed to accommodate the most demanding requirements and extends the benefits of Kinetix Integrated Motion to applications up to 149 kW. The Kinetix 7000 high power drive supports three-phase AC input power (380…480V AC) and DC input for common bus applications. In addition, the safe-off capability integrated into this drive increases productivity by allowing manufacturers and machine builders to implement machine solutions that provide both safety and maximum machine availability.

The Kinetix 7000 high power servo drives are part of the Kinetix Integrated Motion solution.

Kinetix 7000 Servo Drive Components

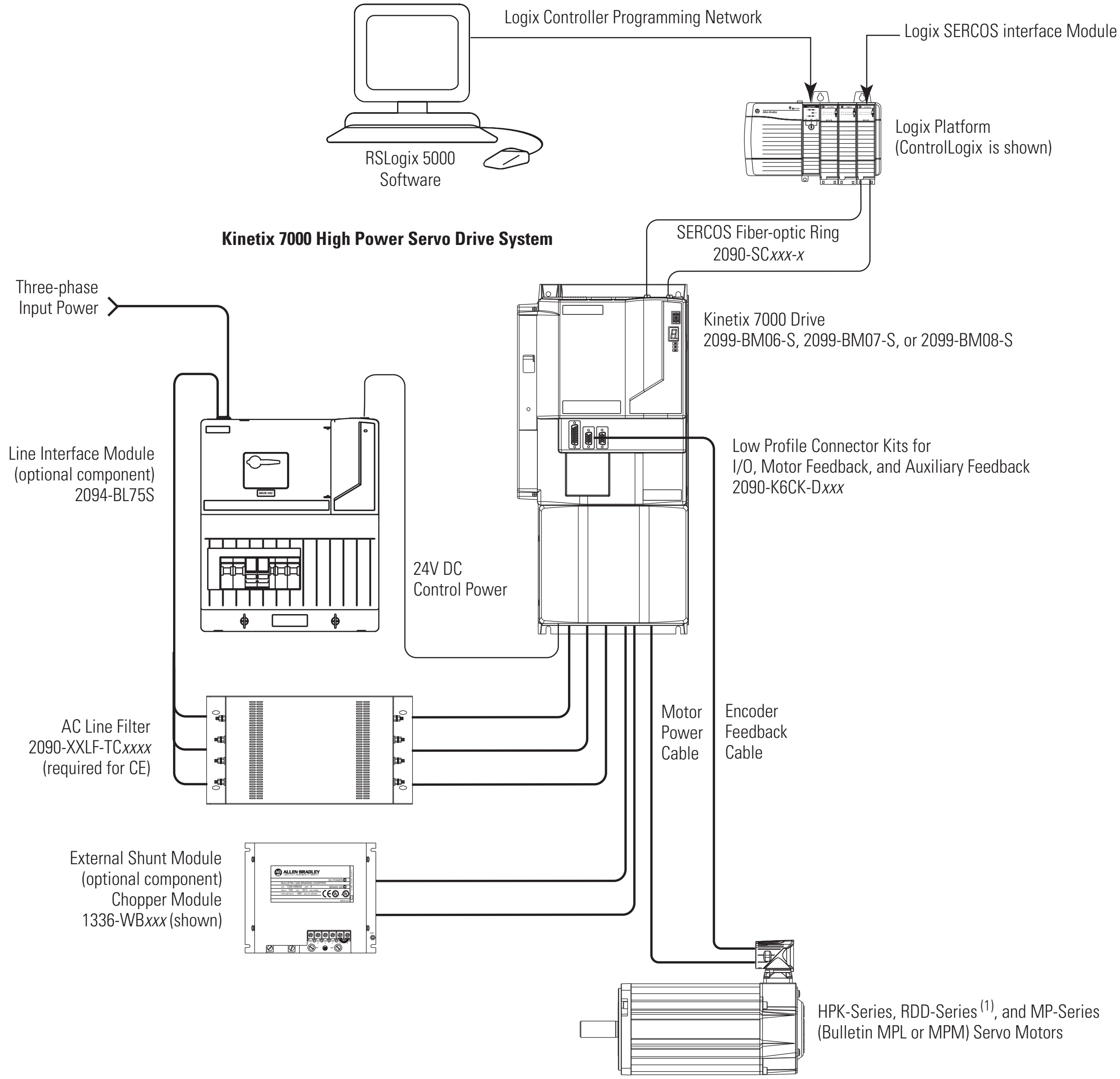
Kinetix 7000 Servo Drive systems consist of these required components:

- One Kinetix 7000 high power drive, 2099-BM*xx*-S
- One HPK-Series, MP-Series (Bulletin MPL or Bulletin MPM), or RDD-Series motor. RDD-Series motors require the 2090-K7CK-KENDAT low-profile feedback module, all others require the 2090-K6CK-D15M low-profile connector kit for flying-lead feedback cables.
- One motor power and feedback cable
- Two SERCOS fiber-optic cables, 2090-SC*xxx-x*

Kinetix 7000 systems may also include any of these optional components:

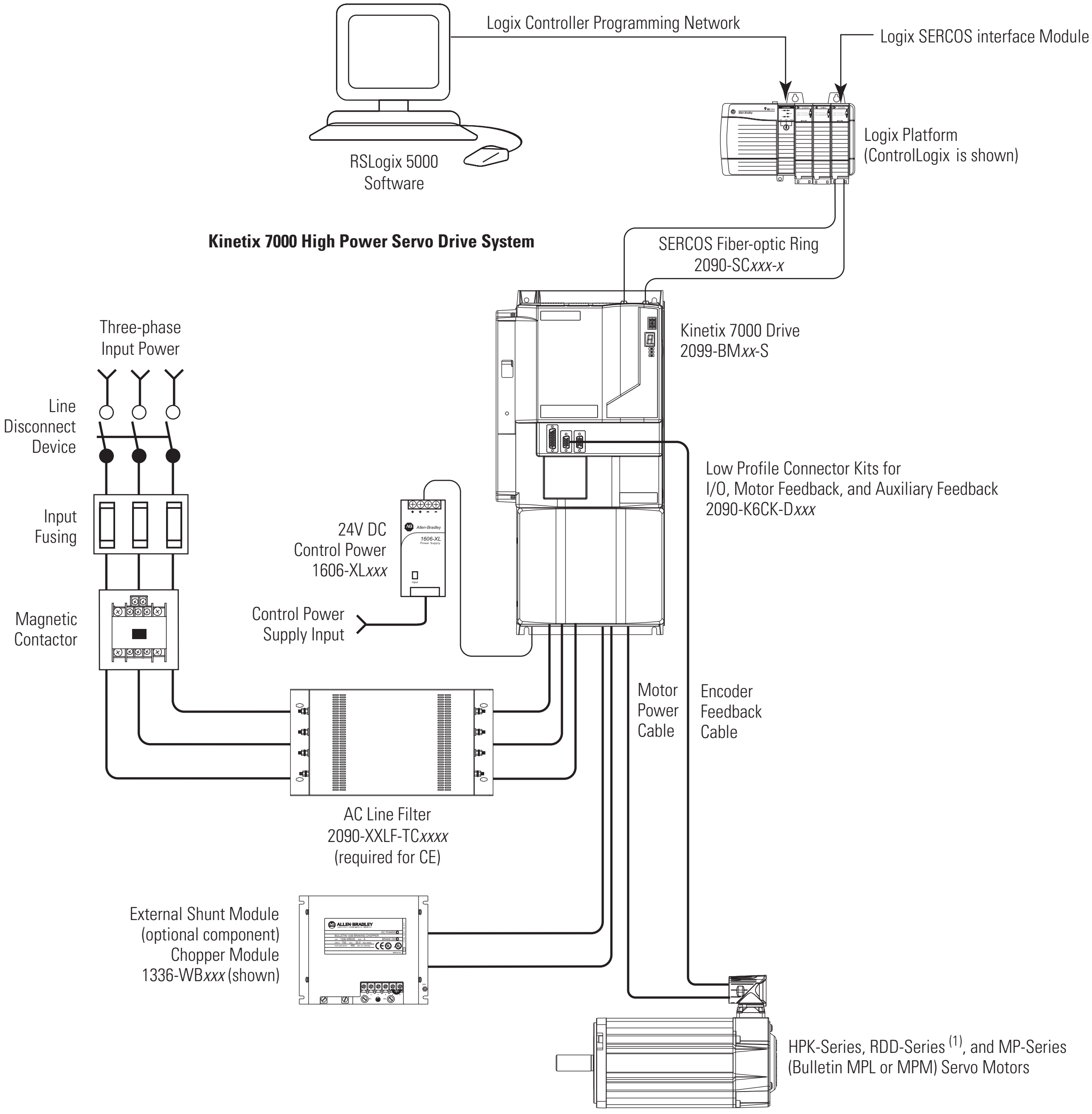
- Bulletin 8720MC Regenerative Power Supply (RPS), 8720MC-RPS*xxx* (DC common bus operation) with harmonic filter and varistor
- Bulletin 8720MC-LR*xx-xxxx* line reactor
- Bulletin 1336-WB*xxx* brake chopper module
- Bulletin 2094-BL75S (460V) Line Interface Module (LIM)

Typical Configuration - Kinetix 7000 (AC input) System (with LIM module)



(1) RDD-Series direct-drive motors require the 2090-K7CK-KENDAT low-profile feedback module for Kinetix 7000 drive applications.

Typical Configuration - Kinetix 7000 (AC input) System (without LIM module)

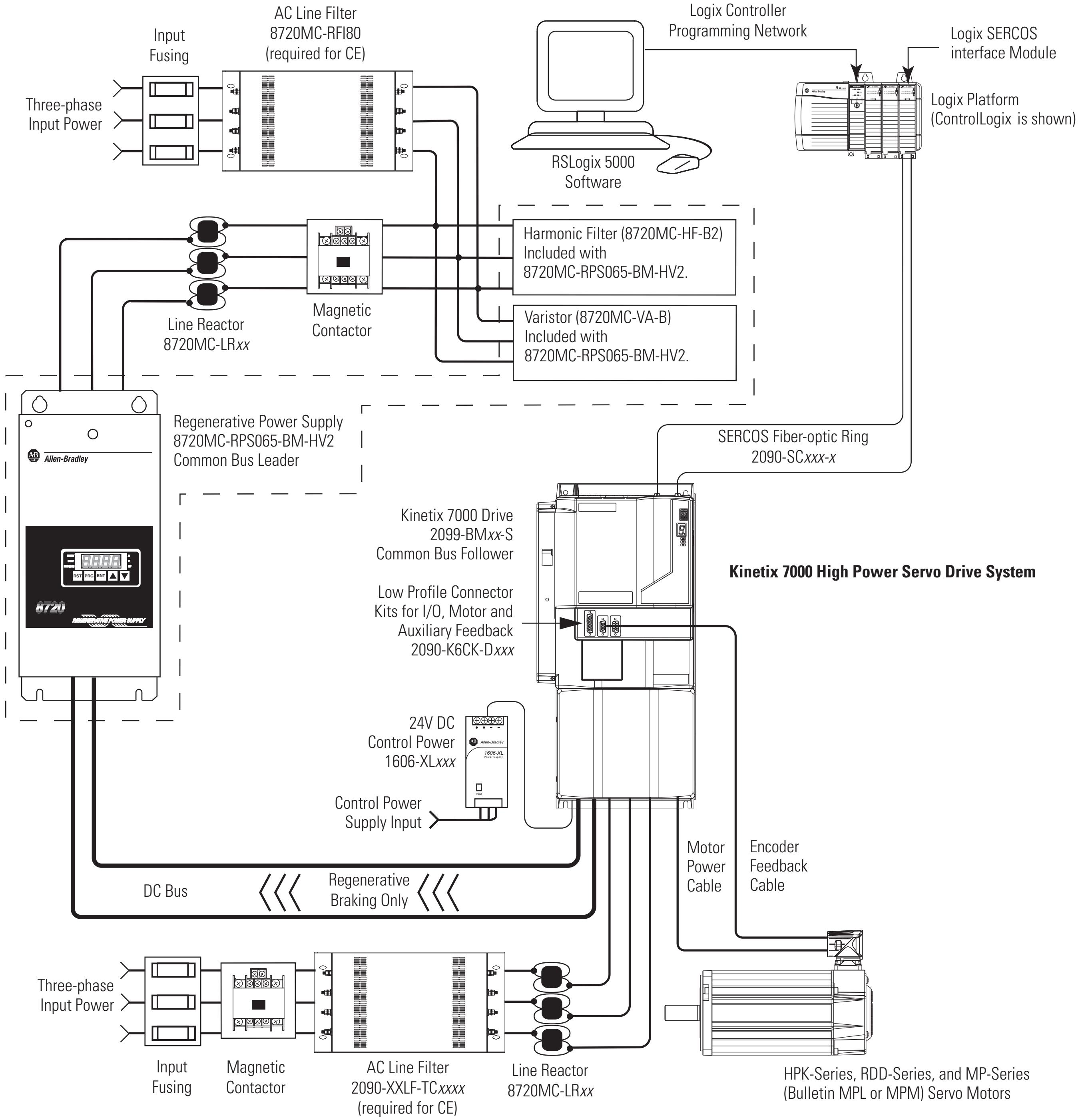


---

(1) RDD-Series direct-drive motors require the 2090-K7CK-KENDAT low-profile feedback module for Kinetix 7000 drive applications.

In the figure below, the Kinetix 7000 drive system is shown with the 8720MC Regenerative Power Supply (RPS) in a regenerative braking configuration. Harmonic filter and varistor are available separately, but are included with the RPS unit when ordering catalog number 8720MC-RPS065-BM-HV2. In this configuration, the Kinetix 7000 drive provides motoring power and the RPS unit provides regenerative braking.

Typical Configuration - Kinetix 7000 (AC input) System (with regenerative braking)

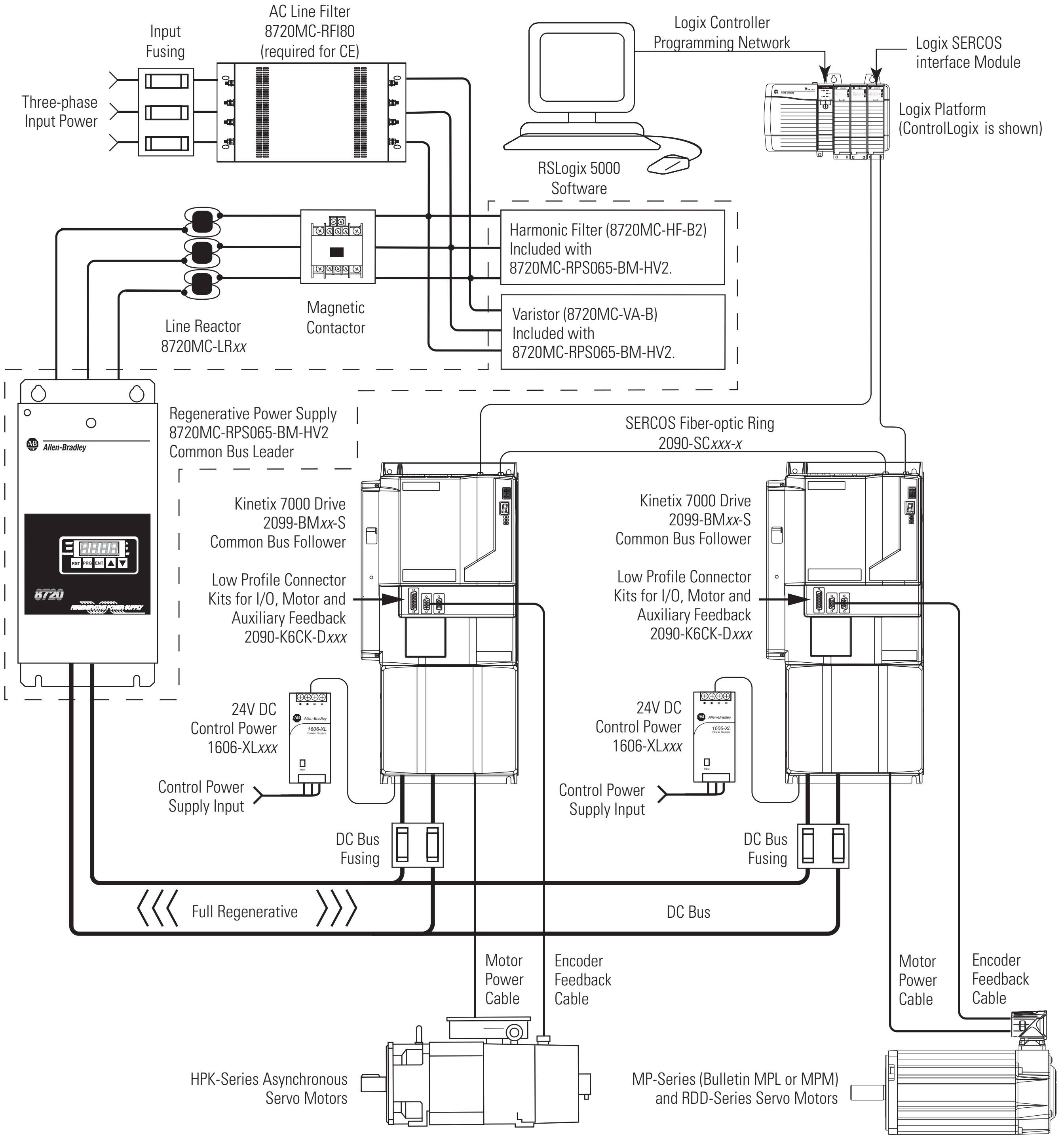


IMPORTANT Regenerative braking applications are limited to only one Kinetix 7000 common bus follower drive.

In the figure below, the Kinetix 7000 drive system is shown with the 8720MC Regenerative Power Supply (RPS) in DC common bus configuration with two follower Kinetix 7000 drives. Harmonic filter and varistor are available separately, but are included with the when ordering the 8720MC-RPS065-BM-HV2 RPS unit.

In Full-line Regenerative mode the 8720MC-RPS065 unit provides motoring power and regenerative braking.

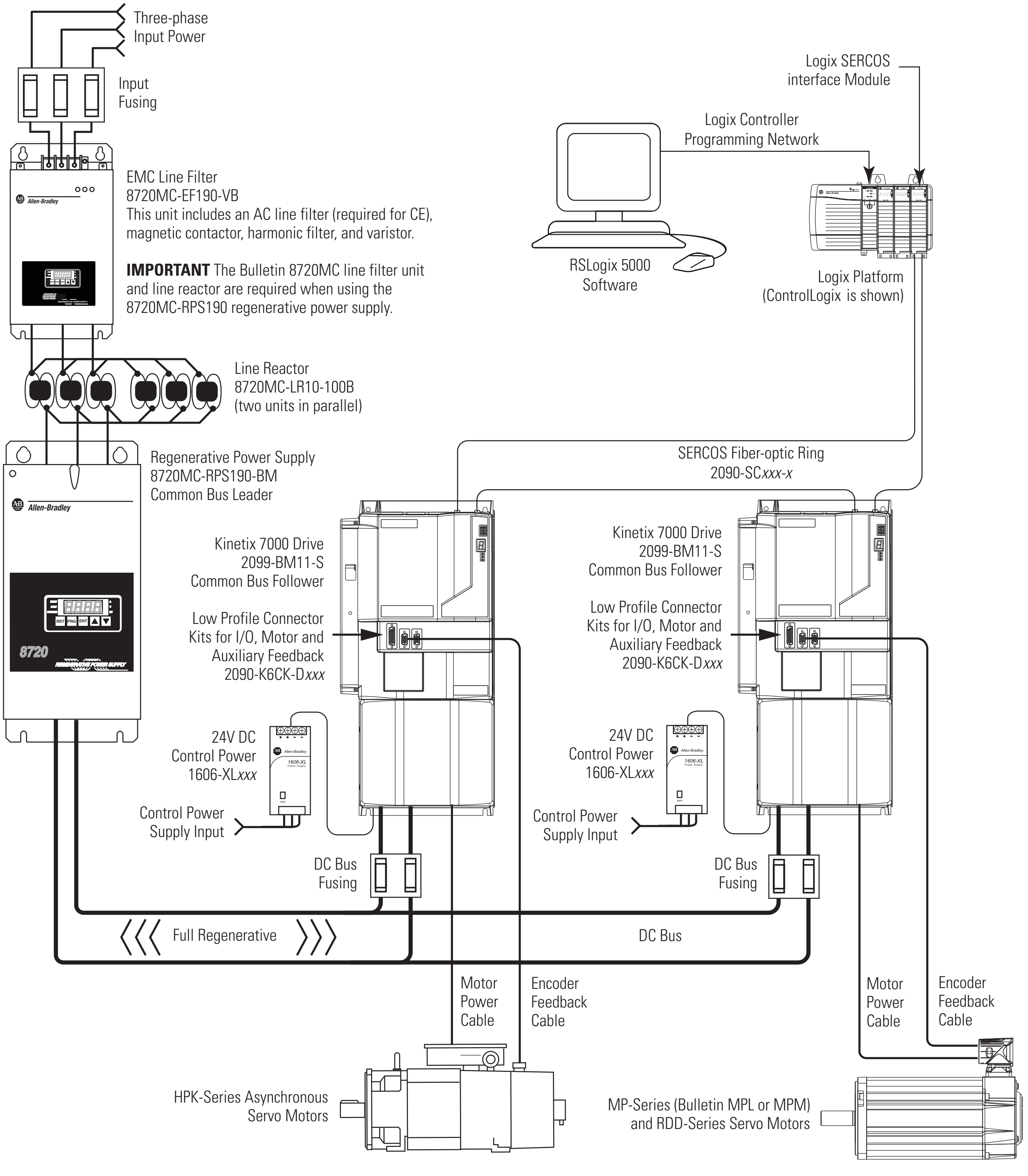
Typical Configuration - Kinetix 7000 (DC input from RPS unit) System (with full-line regeneration)



In the figure below, the Kinetix 7000 drive system is shown with the 8720MC Regenerative Power Supply (RPS) in DC common bus configuration with two follower 2099-BM11-S drives. Harmonic filter, varistor, and magnetic contactor are included when ordering the 8720MC-EF190-VB EMC line filter.

In Full-line Regenerative mode the 8720MC-RPS190 unit provides motoring power and regenerative braking.

Typical Configuration - Kinetix 7000 (DC input from RPS unit) System (with full-line regeneration)



Kinetix 7000 System Component Compatibility Charts

These tables provide input power component combinations with Kinetix 7000 compatible motors. Combinations are designed to provide peak performance.

Input Power Components with HPK-B (460V) Motors

Kinetix 7000 Drive Cat. No.	Common Bus Accessories		Compatible Motors
	8720MC Regenerative Power Supply (1) 8720MC-RPS	**8720MC Line Reactor** (1)	
2099-BM07-S	8720MC-RPS065BM	8720MC-LR05-048B	HPK-B1307C
			HPK-B1308C
2099-BM08-S			HPK-B1307E
			HPK-B1310C
2099-BM09-S		8720MC-LR10-062B	HPK-B1308E
	8720MC-RPS065BM and 8720MC-RPS065BS	8720MC-LR05-048B (two units)	HPK-B1609E
			HPK-B1613C
2099-BM11-S		8720MC-LR10-062B (two units)	HPK-B1611E
			HPK-B1815C
	8720MC-RPS190BM	8720MC-LR10-100B (two units)	HPK-B1613E
			HPK-B2010C
			HPK-B2010E
			HPK-B2212C
2099-BM12-S			HPK-B2010E
			HPK-B2212C
			HPK-B2510C

(1) Regenerative Power Supply (RPS) selection is for this single motor/drive combination. When combining multiple drives on the same RPS module, the selection will change.

Input Power Components with HPK-E (400V) Motors

Kinetix 7000 Drive Cat. No.	Common Bus Accessories		Compatible Motors
	8720MC Regenerative Power Supply (1)	8720MC Line Reactor (1) 8720MC-LR*xx-xxx*B	
2099-BM08-S	8720MC-RPS065BM	8720MC-LR05-048B	HPK-E1307C
2099-BM09-S		8720MC-LR14-070B	HPK-E1308E
		8720MC-LR10-062B	HPK-E1310C
2099-BM11-S	8720MC-RPS065BM and 8720MC-RPS065BS	8720MC-LR05-048B (two units)	HPK-E1613C
		8720MC-LR10-062B (two units)	HPK-E1609E
			HPK-E1611E
			HPK-E1815C
	8720MC-RPS190BM	8720MC-LR10-100B (two units)	HPK-E1613E
			HPK-E2010C

(1) Regenerative Power Supply (RPS) selection is for this single motor/drive combination. When combining multiple drives on the same RPS module, the selection will change.

Input Power Components with MPL-B (460V) Motors

Kinetix 7000 Drive Cat. No.	Common Bus Accessories		Compatible Motors
	8720MC Regenerative Power Supply (1)	8720MC Line Reactor (1)	
2099-BM06-S	8720MC-RPS065BM	8720MC-LR05-048B	MPL-B540K
			MPL-B560F
			MPL-B580F
			MPL-B580J
			MPL-B640F
			MPL-B660F
			MPL-B680D
			MPL-B680F
			MPL-B860D
			MPL-B960B
			MPL-B980B
2099-BM07-S			MPL-B880C
		8720MC-LR10-062B	MPL-B880D
2099-BM08-S			MPL-B960C
			MPL-B960D
			MPL-B980C
			MPL-B980D

(1) Regenerative Power Supply (RPS) selection is for this single motor/drive combination. When combining multiple drives on the same RPS module, the selection will change.

Input Power Components with MPM-B (480V) Motors

Kinetix 7000 Drive Cat. No.	Common Bus Accessories		Compatible Motors
	8720MC Regenerative Power Supply (1)	8720MC Line Reactor (1)	
2099-BM06-S	8720MC-RPS065BM	8720MC-LR05-048B	MPM-B1651M
2099-BM06-S			MPM-B1652E
2099-BM06-S			MPM-B1652F
2099-BM06-S			MPM-B1653C
2099-BM06-S			MPM-B1653E
2099-BM06-S			MPM-B1653F
2099-BM06-S			MPM-B2152C
2099-BM07-S			MPM-B2152F
2099-BM06-S			MPM-B2152M
2099-BM06-S			MPM-B2153B
2099-BM07-S			MPM-B2153E
2099-BM07-S			MPM-B2153F
2099-BM07-S			MPM-B2154B
2099-BM07-S			MPM-B2154E
2099-BM06-S			MPM-B2154F

(1) Regenerative Power Supply (RPS) selection is for this single motor/drive combination. When combining multiple drives on the same RPS module, the selection will change.

Input Power Components with RDB-B (480V) Motors

Kinetix 7000 Drive Cat. No.	Common Bus Accessories		Compatible Motors
	8720MC Regenerative Power Supply (1)	8720MC Line Reactor (1)	
2099-BM06-S	8720MC-RPS065BM	8720MC-LR05-048B	RDB-B2152C
2099-BM06-S			RDB-B2153C
2099-BM07-S			RDB-B29029
2099-BM06-S			RDB-B29036
2099-BM08-S			RDB-B29039
2099-BM06-S			RDB-B41016
2099-BM08-S			RDB-B41018
2099-BM06-S			RDB-B41024
2099-BM09-S			RDB-B41026
2099-BM09-S			RDB-B41035

(1) Regenerative Power Supply (RPS) selection is for this single motor/drive combination. When combining multiple drives on the same RPS module, the selection will change.

Kinetix 7000 High Power Drive Specifications

This section contains specifications, mounting dimensions, and catalog numbers for the Kinetix 7000 (2099-BM*xx*-S) drives.

Power Specifications

Attribute	**2099-BM06-S**	**2099-BM07-S**	**2099-BM08-S**	**2099-BM09-S**	**2099-BM10-S**	**2099-BM11-S**	**2099-BM12-S**
AC input voltage	342…528V AC rms three-phase (380…480V nom)						
AC input frequency	47…63 Hz						
Bandwidth (1)							
Velocity loop	500 Hz						500 Hz
Current loop	1300 Hz						500 Hz
PWM frequency	4 kHz						2 kHz
Main AC input current							
Nom (rms)	36.7 A	47.7 A	59.6 A	90.1 A	117 A	169 A	233 A
Max inrush (A peak)	18.0 A	18.0 A	18.0 A	96.0 A	118 A	141 A	141 A
DC input voltage	450…750V DC						
DC input current	42.9 A	55.7 A	69.7 A	105 A	137 A	204 A	281 A
Control power input Voltage	18…30V DC (24V DC, nom)						
Control power DC input current							
Nom (rms)	3.3 A						
Maximum inrush (rms)	6.0 A						
Continuous output current (rms)	40.0 A	52.0 A	65.0 A	96.0 A	125 A	180 A	248 A
Continuous output current (0-pk)	56.0 A	73.0 A	92.0 A	135 A	176 A	254 A	351 A
Peak output current (rms)							
3 s duration	68.0 A	80.0 A	104 A	154 A	163 A	312 A	372 A
60 s duration	51.0 A	60.0 A	78.0A	115 A	138 A	234 A	273 A
Peak output current (0-pk)							
3 s duration	96.0 A	113 A	147 A	217.7 A	230.5 A	441 A	526 A
60 s duration	72.0 A	84.8 A	110 A	162.6 A	195 A	331 A	386 A
Bus overvoltage	800V DC						
Bus undervoltage	275…560V DC (2)						
Continuous power output, nom	22 kW	30 kW	37 kW	56 kW	75 kW	112 kW	149 kW
Continuous power output (Hp)	30 Hp	40 Hp	50 Hp	75 Hp	100 Hp	150 Hp	200 Hp
Maximum power cycles/minute							
AC line	4 per minute (pre-charge provided by drive)						
DC bus	2 per minute (DC pre-charge provided by the regenerative power supply)						
DC bus discharge time	3 minutes after removal of main AC power						
Efficiency	97.5%						
Total capacitance (3)	1800 µF	2400 µF	3000 µF	4500 µF	6000 µF	8400 µF	8400 µF
Short circuit current rating	200,000 A (rms) symmetrical						

(1) Bandwidth values vary based on tuning parameters and mechanical components.

(2) Bus undervoltage will vary based on input line voltage.

(3) If DC input is supplied to 2099-BM09-S, 2099-BM10-S, or 2099-BM11-S drives, the precharge capability must be provided at the system level. Disconnect switches must not be used between the input of the drive and a common DC bus without the use of an external precharge device.

Environmental Specifications

Attribute	Operational Range	Storage Range (nonoperating)
Temperature, ambient	0…50 ° C (32…122 ° F)	-40…70 ° C (-40…158 ° F)
Relative humidity	5…95% noncondensing	5…95% noncondensing
Altitude	1000 m (3281 ft) 3000 m (9843 ft) with derating	3000 m (9843 ft) during transport
Vibration	5…55 Hz @ 0.35 mm (0.014 in.) double amplitude, continuous displacement; 55…500 Hz @ 2.0 g peak constant acceleration (10 sweeps in each of 3 mutually perpendicular directions).	
Shock	15 g, 11 ms half-sine pulse (3 pulses in each direction of 3 mutually perpendicular directions)	

Weight Specifications

Drive Cat. No.	Weight, approx. kg (lb)
2099-BM06-S	18.55 (40.9)
2099-BM07-S	
2099-BM08-S	
2099-BM09-S	37.2 (82.0)
2099-BM10-S	
2099-BM11-S	71.4 (157.5)
2099-BM12-S	

Power Dissipation Specifications

Use this table to size an enclosure and calculate required ventilation for your Kinetix 7000 drive system.

Drive Cat. No.	Usage as % of Rated Power Output W	
	50%	100%
2099-BM06-S	294	465
2099-BM07-S	388	619
2099-BM08-S	452	730
2099-BM09-S	645	1072
2099-BM10-S	882	1479
2099-BM11-S	1275	2125
2099-BM12-S	1438	2437

Circuit Breaker/Fuse Specifications

While circuit breakers offer some convenience, there are limitations for their use. Circuit breakers do not handle high current inrush as well as fuses.

Make sure the selected components are properly coordinated and meet acceptable codes including any requirements for branch circuit protection. Evaluation of the short-circuit available current is critical and must be kept below the short-circuit current rating of the circuit breaker.

Use class CC, T, RK1, or J fuses, with current rating as indicated in the table below. The following fuse examples and short-circuit current ratings are recommended for use with the 2099-BM*xx*-S drives when the Line Interface Module (LIM) is not used.

IMPORTANT LIM modules (catalog numbers 2094-BL*xx*S and 2094-XL75S-C*x*) provide branch circuit protection to the Kinetix 7000 drive. Follow all applicable NEC and local codes.

Fuse Specifications

Drive Cat. No.	Bussmann Fuse	Dual Element Time Delay Fuse (min/max) A rms	Non-Time Delay Fuse (min/max) A rms	Motor Circuit Protector (max) A rms
2099-BM06-S	LPJ-90SP	50/90	50/150	50
2099-BM07-S	LPJ-110SP	60/110	60/200	70
2099-BM08-S	LPJ-125SP	80/125	80/250	100
2099-BM09-S	LPJ-200SP	125/200	125/300	125
2099-BM10-S	LPJ-250SP	150/250	150/500	150
2099-BM11-S	LPJ-400SP	225/400	225/600	250
2099-BM12-S	LPJ-500SP	300/550	300/700	400

Contactor Ratings

The table below lists the recommended contactor ratings for Kinetix 7000 drives installed without a Line Interface Module.

Drive Cat. No.	Contactor	Safety Contactor	Coil Type	Coil Voltage Requirements
2099-BM06-S	100-C43DJ01	100S-C43-DJD4C	Standard with Diode	24V DC
2099-BM07-S	100-D95EN11	100S-D95EN22C	Electronic Coil (1)	24V DC for control and 480V AC for coil power
2099-BM08-S				
2099-BM09-S				
2099-BM10-S	100-D140EN11	100S-D140EN22C		
2099-BM11-S	100-D180EN11	100S-D180EN22C		
2099-BM12-S	100-D250EN11	100S-D250EN22C		

(1) Electronic coil control power requirements = 24V DC @ 15 mA.

Maximum Feedback Cable Lengths

Although motor power and feedback cables are available in standard lengths up to 90 m (295.3 ft), the drive/motor/feedback combination may limit the maximum feedback cable length. This table assumes the use of recommended cables as shown in the 2090-Series Motor/Actuator Cable Selection table on page 401.

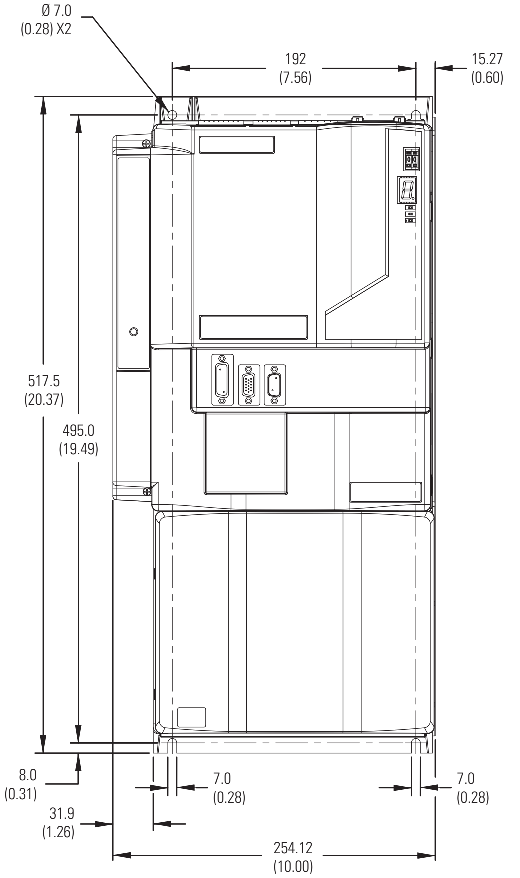
Cable Lengths for Compatible Rotary Motors

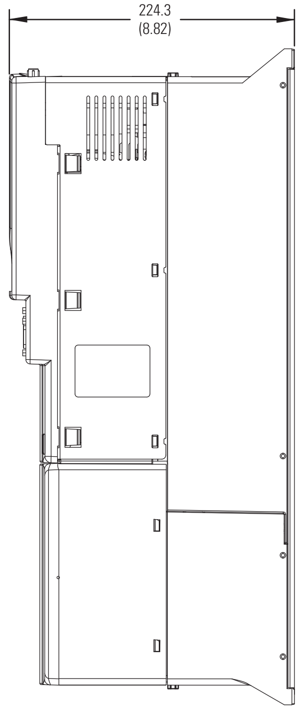
Motor Cat. No.	**Absolute High-resolution (5V) Encoder** m (ft)	**Absolute High-resolution (9V) Encoder** m (ft)
MPL-B5*xxx*… MPL-B9*xxx*-S/M		90 (295.3)
MPM-B165*xx*… MPM-B215*xx*-S/M		90 (295.3)
RDB-B215*xx*-7/3	30 (98.4)	
RDB-B290*xx*-7/3 or RDB-B410*xx*-7/3	90 (295.3)	
HPK-B*xxxxx*-S/M or HPK-E*xxxxx*-S/M		90 (295.3)

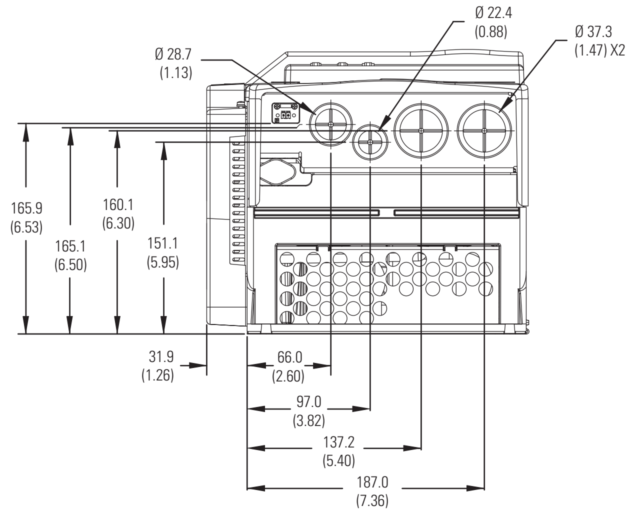
Kinetix 7000 High Power Drive Dimensions

2099-BM06-S, 2099-BM07-S, and 2099-BM08-S Dimensions

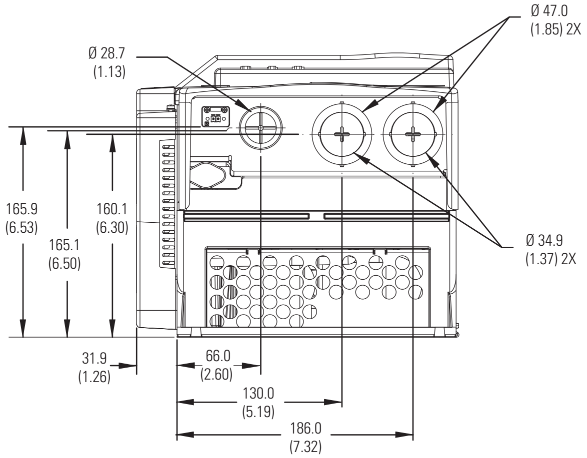
Dimensions are in mm (in.)







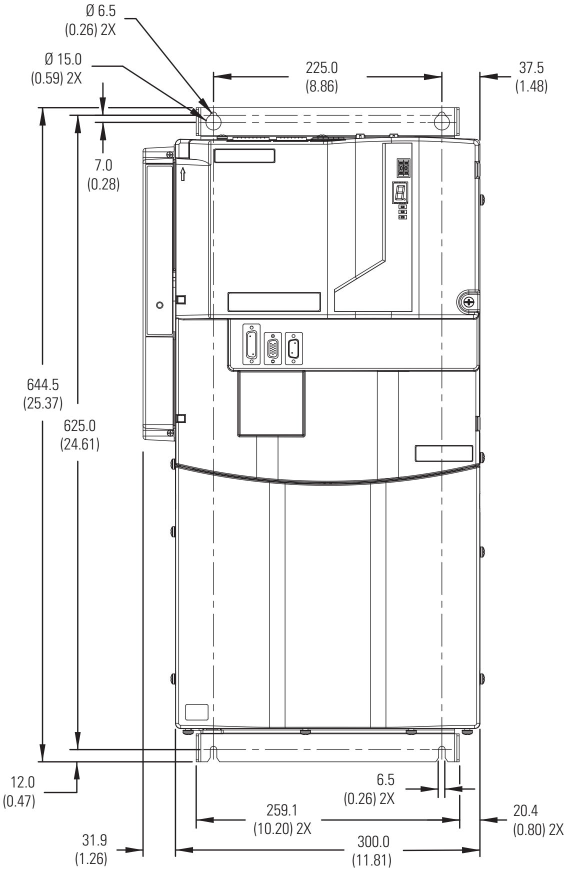
Bottom View
(2099-BM06-S and 2099-BM07-S)

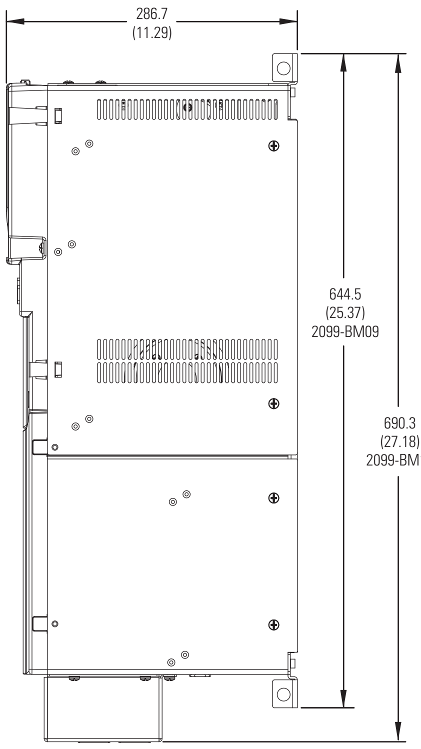


Bottom View
(2099-BM08-S)

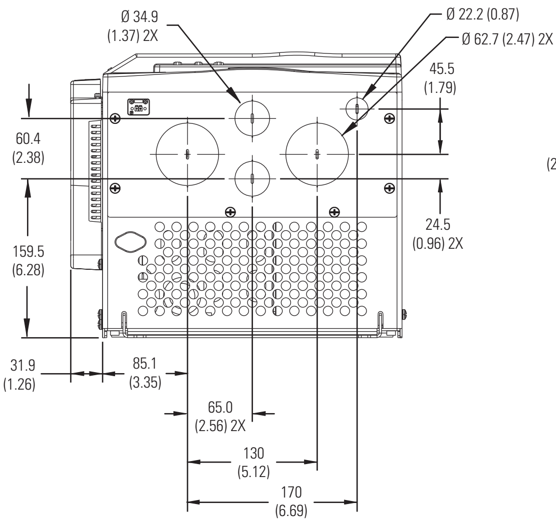
2099-BM09-S and 2099-BM10-S Dimensions

Dimensions are in mm (in.)

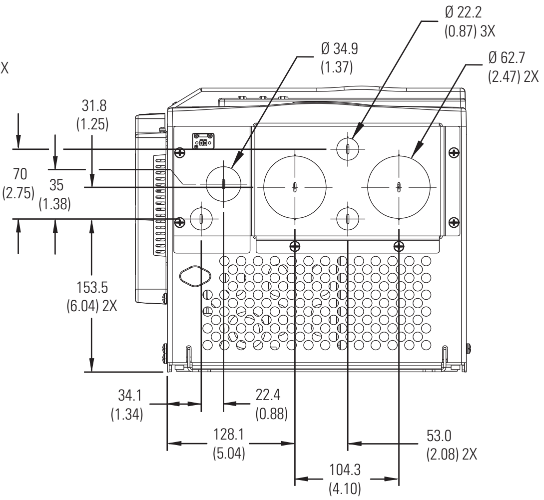




Some Components are Removed for Clarity



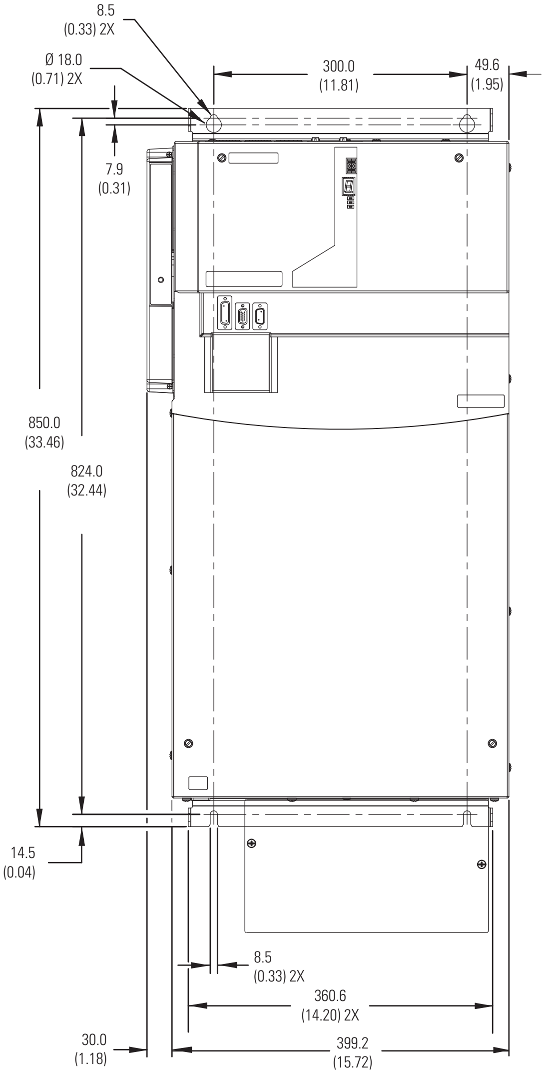
Bottom View
(2099-BM09-S)

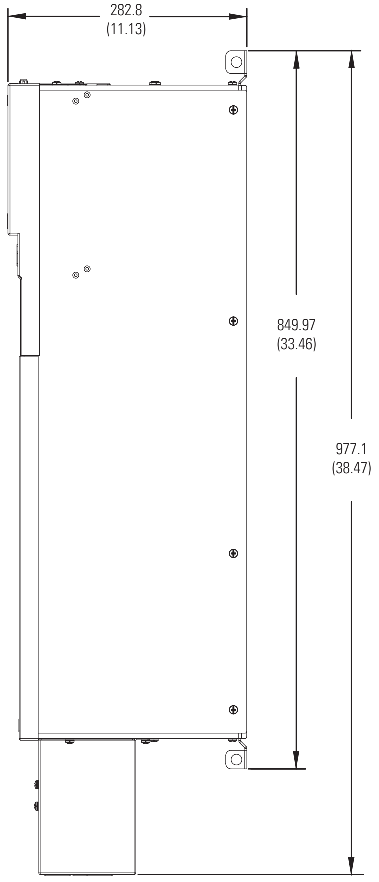


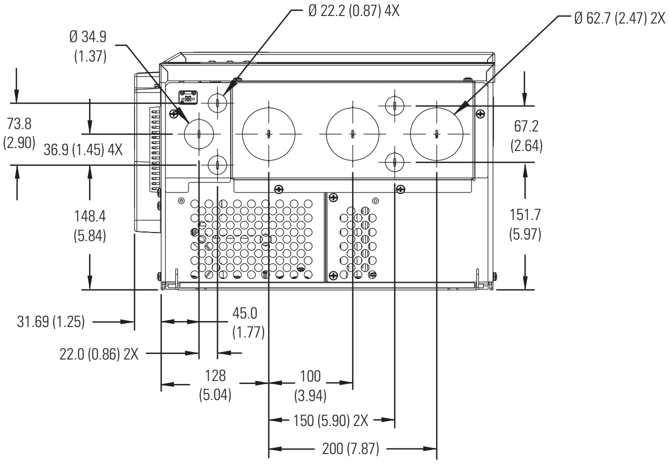
Bottom View
(2099-BM10-S)

2099-BM11-S and 2099-BM12-S Dimensions

Dimensions are in mm (in.)



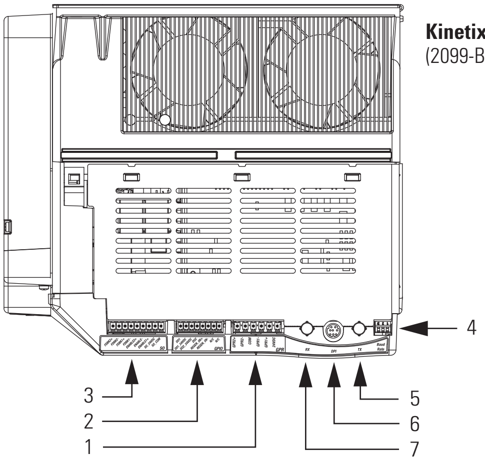




Kinetix 7000 Connector, Indicator, and Switch Locations

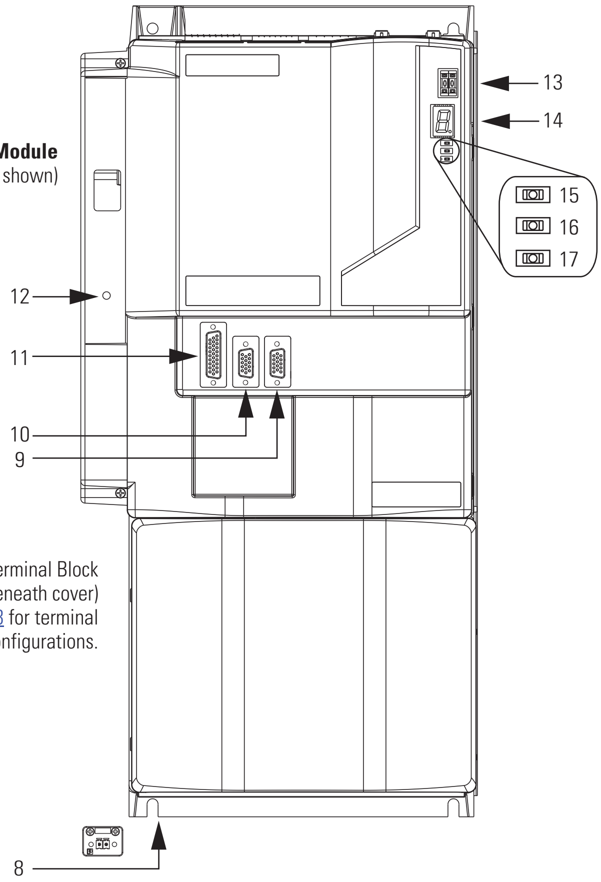
This section contains connector, indicator, switch, and terminal block locations for the Kinetix 7000 drive modules.

Kinetix 7000 Connectors



Kinetix 7000 Drive Module
(2099-BM07-S top view is shown)

Kinetix 7000 Drive Module
(2099-BM07-S front view is shown)

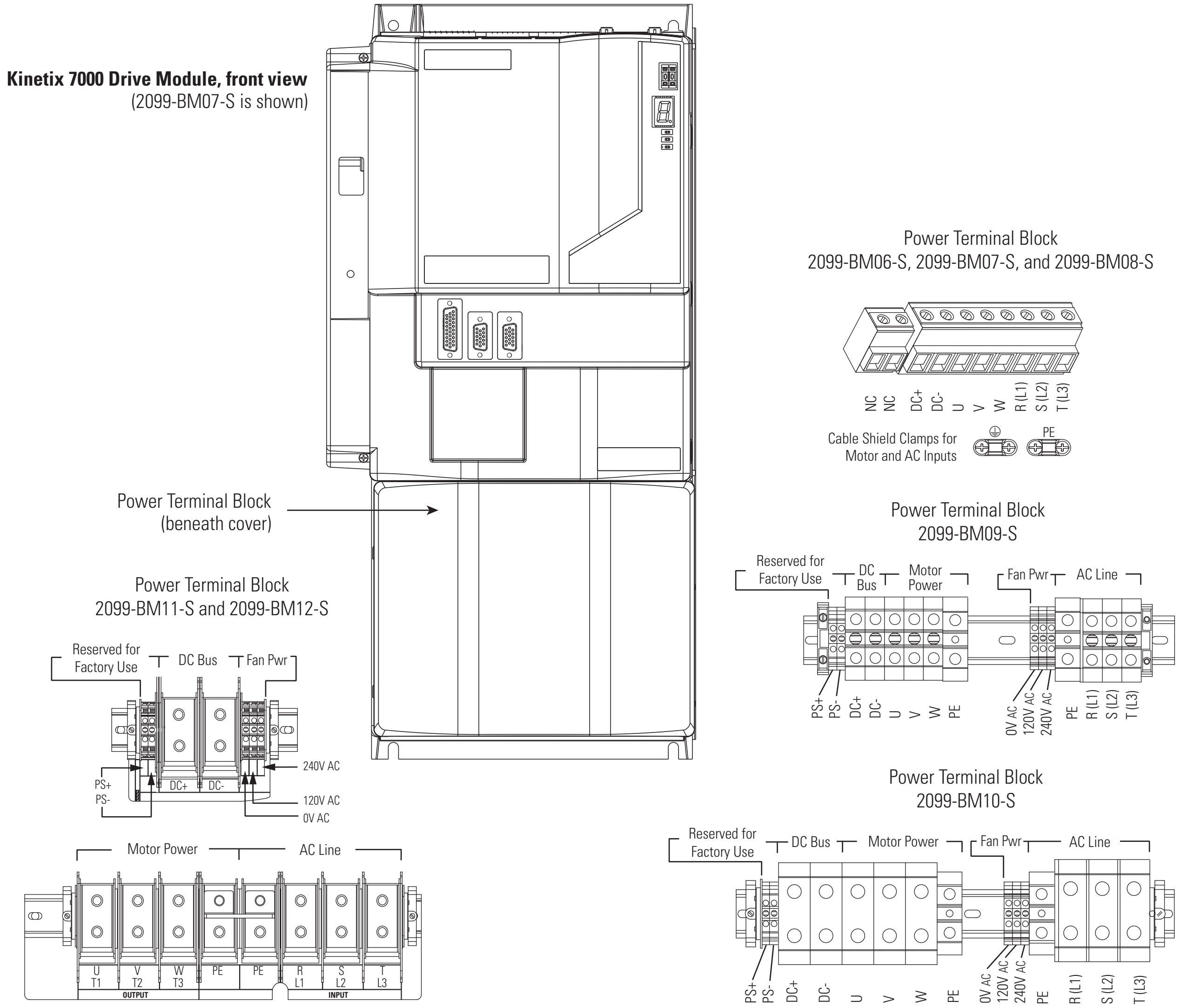


Power Terminal Block
(beneath cover)
Refer to page 338 for terminal
block configurations.

Item	Description
1	General purpose (GPR) connector
2	General purpose (GPIO) connector
3	Safe-off (SO) connector
4	SERCOS communication rate and optical power switches
5	SERCOS transmit (Tx) connector
6	DPI connector
7	SERCOS receive (Rx) connector
8	Control power (CP) connector (facing down)
9	Auxiliary feedback (AF) connector
10	Motor feedback (MF) connector
11	I/O (IOD) connector
12	Control power status indicator
13	SERCOS node address switches
14	Seven-segment fault status indicator
15	Drive status indicator
16	COMM status indicator
17	Bus status indicator

For connector kit options, refer to Breakout Components and Connector Kits beginning on page 440.

Kinetix 7000 Power Terminal Blocks



Kinetix 7000 High Power Drive Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your module. For questions regarding product availability, contact your Allen-Bradley distributor.

2099 - BM *xx* - S

Safe-off Feature

Inverter Current Rating Continuous
BM06 = 056 A 0-pk 040 A rms 22 kW
BM07 = 073 A 0-pk 052 A rms 30 kW
BM08 = 092 A 0-pk 065 A rms 37 kW
BM09 = 135 A 0-pk 096 A rms 56 kW
BM10 = 176 A 0-pk 125 A rms 75 kW
BM11 = 254 A 0-pk 180 A rms 112 kW
BM12 = 351 A 0-pk 248 A rms 149 kW

Input Voltage
B = 380…480V AC or 450…750V DC

Bulletin Number

Kinetix 300 EtherNet/IP Indexing Servo Drives

The Kinetix 300 EtherNet/IP indexing drive provides a cost-effective single-axis solution for low axis count motion control applications. Using one standard Ethernet/IP network for an entire machine - including Motion, Control, I/O, and HMI simplifies wiring, reduces panel layout costs, and allows easy integration into manufacturing and enterprise systems. In addition, safe torque-off functionality helps protect personnel while increasing machine productivity.

Kinetix 300 Servo Drive Components

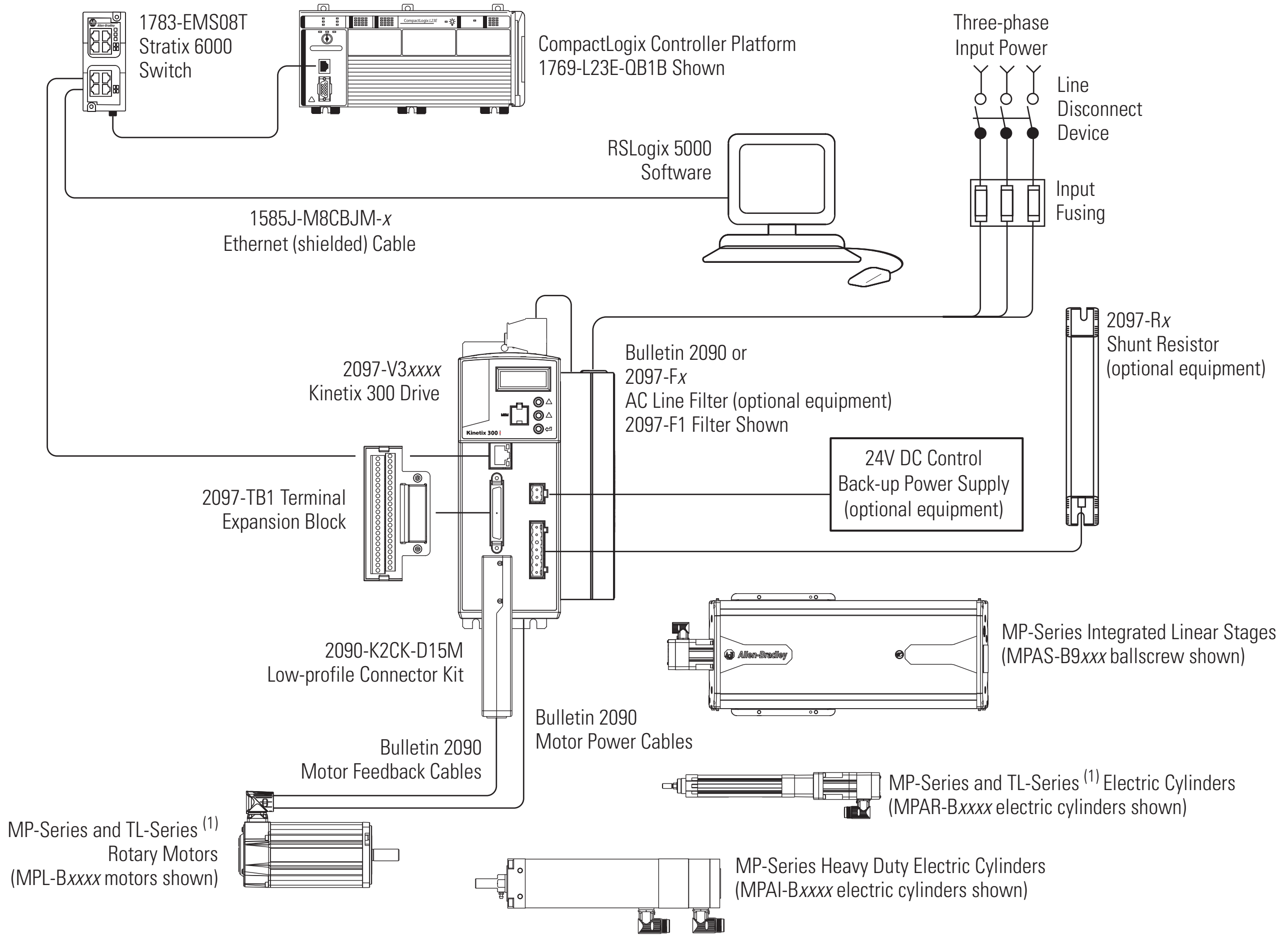
Kinetix 300 servo drive systems consist of these required components:

- One 2097-V3*xxxx* indexing drive
- One MP-Series or TL-Series servo motor or linear actuator
- One motor power and motor feedback cable
- One 2090-K2CK-D15M low-profile connector kit for motor feedback
- One 2097-TB1 I/O terminal expansion block
- 1585J-M8CBJM-*x* (shielded) Ethernet cable

Kinetix 300 servo drive systems may also include any of these optional components:

- One 2097-F*x* or Bulletin 2090 AC line filter
- One 2097-R*x* shunt resistor

Typical Configuration - Kinetix 300 Drive System



(1) TL-Series (Bulletin TLY) rotary motors and Bulletin TLAR electric cylinders require the 2090-K2CK-D15M connector kit with 2090-DA-BAT2 battery for multi-turn high-resolution encoder operation. Other Kinetix 300 compatible motors and actuators require the connector kit for flying-lead feedback connections, but not the battery.

Kinetix 300 Drive Power Specifications

The 2097-V31PR*x* drives with 120V input are capable of driving 240V motors at full speed.

Kinetix 300 Drive (single-phase) Power Specifications

Attribute	2097-V31PR0	2097-V31PR2	2097-V32PR0	2097-V32PR2	2097-V32PR4
AC input voltage	70…132V rms single-phase (120V nom) 80…264V rms single-phase (240V nom)		80…264V rms single-phase (240V nom)		
AC input frequency	48…62 Hz				
Main AC input current (1)					
Nom (rms) 120V input (voltage doubler)	9.70 A	15.0 A			
Max inrush (0-pk) 120V input	1.15 A	1.15 A			
Nom (rms) 120/240V input	5.0 A	8.6 A	5.0 A	8.6 A	15.0 A
Max inrush (0-pk) 240V input	1.1 A	1.1 A	136 A	2.3 A	2.3 A
Integrated AC line filter	No	No	Yes	Yes	Yes
Control power back-up input voltage	20…26V DC				
Control power back-up input current					
Nom	500 mA				
Max inrush (0-pk)	30 A				
Continuous output current (rms)	2.0 A	4.0 A	2.0 A	4.0 A	8.0 A
Continuous output current (0-pk)	2.8 A	5.7 A	2.8 A	5.7 A	11.3 A
Peak output current (rms) (2)	6.0 A	12.0 A	6.0 A	12.0 A	24.0 A
Peak output current (0-pk)	8.5 A	17.0 A	8.5 A	17.0 A	33.9 A
Continuous power output (3) @ 240V nom or 120V (voltage-doubler) mode	0.40 kW (4)	0.80 kW (4)	0.40 kW	0.80 kW	1.70 kW
Shunt on	390V DC				
Shunt off	375V DC				
Overvoltage	430V DC				
Short circuit current rating	100,000 A (rms) symmetrical				

(1) Kinetix 300 drive modules are limited to 1 AC mains power cycling per every 2 minutes.

(2) Peak RMS current allowed for up to 2 seconds with a 50% duty cycle.

(3) Nominal continuous power output (kW) applies to 240V AC drives. Value is approximately one-half of this kW rating when using 120V AC.

(4) The 120V voltage-doubler mode applies to only the 2097-V31PR*x* drives.

Kinetix 300 Drive (single-phase and three-phase) Power Specifications

Attribute	2097-V33PR1	2097-V33PR3	2097-V33PR5	2097-V33PR6
AC input voltage	80…132V rms single-phase (120V nom) 80…264V rms single-phase (240V nom) 80…264V rms three-phase (240V nom)			
AC input frequency	48…62 Hz			
Main AC input current (1)				
Nom (rms) 120V input	5.0 A	8.6 A	15.0 A	24.0 A
Max inrush (0-pk) 120V input	68.0 A	1.15 A	1.15 A	5.65 A
Nom (rms) 240V input	3.0 A	5.0A	8.7A	13.9 A
Max inrush (0-pk) 240V input	136 A	2.3 A	2.3 A	11.3 A
Integrated AC line filter	No	No	No	No
Control power back-up input voltage	20…26V DC			
Control power back-up input current				
Nom	500 mA			
Max inrush (0-pk)	30 A			
Continuous output current (rms)	2.0 A	4.0 A	8.0 A	12.0 A
Continuous output current (0-pk)	2.8 A	5.7 A	11.3 A	17.0 A
Peak output current (rms) (2)	6.0 A	12.0 A	24.0 A	36.0 A
Peak output current (0-pk)	8.5 A	17.0 A	33.9 A	50.9 A
Continuous power output (3) @ 240V nom	0.50 kW	1.00 kW	2.00 kW	3.00 kW
Shunt on	390V DC			
Shunt off	375V DC			
Overvoltage	430V DC			
Short circuit current rating	100,000 A (rms) symmetrical			

(1) Kinetix 300 drive modules are limited to 1 AC mains power cycling per every 2 minutes.

(2) Peak RMS current allowed for up to 2 seconds with a 50% duty cycle.

(3) Nominal continuous power output (kW) applies to 240V AC drives. Value is approximately one-half of this kW rating when using 120V AC.

Kinetix 300 Drive (three-phase) Power Specifications

Attribute	2097-V34PR3	2097-V34PR5	2097-V34PR6
AC input voltage	320…528V rms three-phase (480V nom)		
AC input frequency	48…62 Hz		
Main AC input current (1) Nom (rms) Max inrush (0-pk)	 2.7 A 4.5 A	 5.5 A 4.5 A	 7.9 A 22.6 A
Integrated AC line filter	No	No	No
Control power back-up input voltage	20…26V DC		
Control power back-up input current Nom Max inrush (0-pk)	 500 mA 30 A		
Continuous output current (rms)	2.0 A	4.0 A	6.0 A
Continuous output current (0-pk)	2.8 A	5.7 A	8.5 A
Peak output current (rms) (2)	6.0 A	12.0 A	18.0 A
Peak output current (0-pk)	8.5 A	17.0 A	25.5 A
Continuous power output @ 480V nom	1.00 kW	2.00 kW	3.00 kW
Shunt on	780V DC		
Shunt off	750V DC		
Overvoltage	850V DC		
Short circuit current rating	100,000 A (rms) symmetrical		

(1) Kinetix 300 drive modules are limited to 1 AC mains power cycling per every 2 minutes.

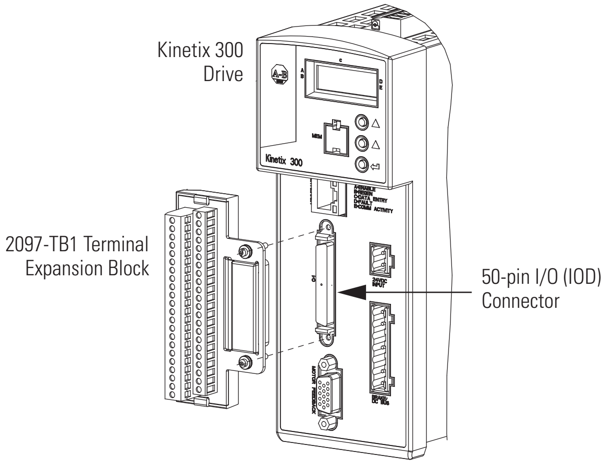
(2) Peak RMS current allowed for up to 2 seconds with a 50% duty cycle.

Kinetix 300 Drive Accessory Specifications

Kinetix 300 drive accessories include the I/O terminal block, memory module programmer, memory modules, AC line filters, and shunt resistors.

I/O Terminal Expansion Block

The 2097-TB1 I/O terminal expansion block is a drive-mounted breakout board for making flying-lead cable connections to the 50-pin IOD connector.



2097-TB1 I/O Terminal Block Specifications

Attribute	Value
Wire size	1.5…0.2 mm^2 (16…24 AWG)
Change in depth of drive (1)	11 mm (0.42 in.)
Change in width of drive (1)	10 mm (0.38 in.)

(1) Add this value to the dimensions of your Kinetix 300 drive.

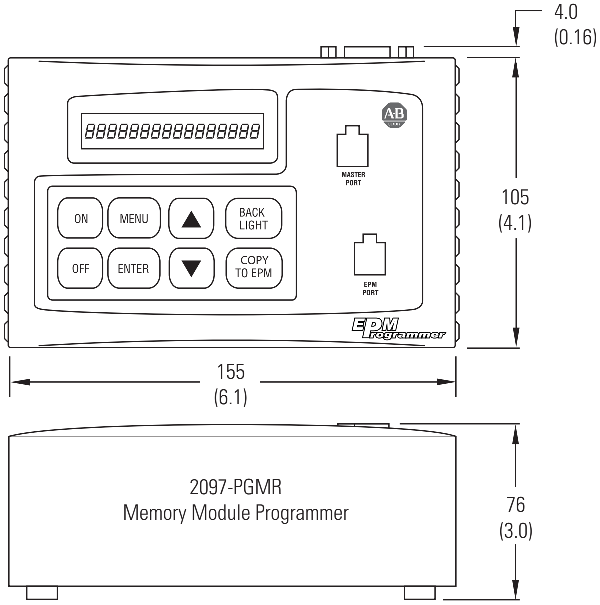
Memory Module Programmer

The 2097-PGMR memory module programmer is a hand-held device for duplicating your Kinetix 300 drive configuration to reduce down-time and troubleshooting.

2097-PGMR Memory Module Programmer Specifications

Attribute		Value
DC supply	Internal batteries	+ 6V DC, min 150 mA supply
		4 mono-cells (type D), 1.5V DC each
	External power supply unit	+ 6V DC, 300 mA, stabilized
Display	Type	LCD
	Display format	Text
	Lines x characters	1 x 16
	Contrast setting	Via menu
Memory	Data memory	Up to 120 parameter files for inverter drive controllers
Serial interface	DB9 connector	RS-232
Weight	2097-PGMR	1.3 kg (2.87 lb), with batteries

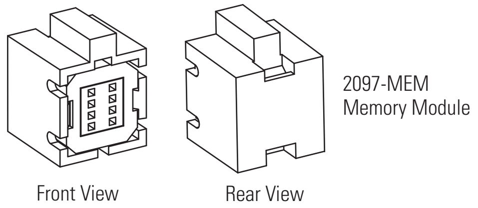
Dimensions are in mm (in.)

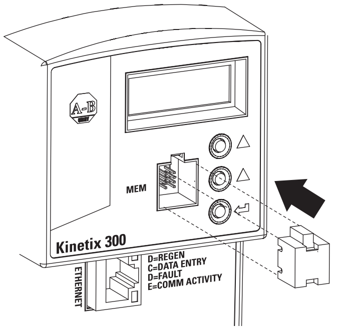


Memory Module 12-packs

The 2097-MEM memory modules use EEPROM technology in a plastic casing for protection and ruggedness to safe-guard your Kinetix 300 drive configuration.

Use the 2097-MEM memory modules to back up your drive configuration for easy Automatic Device Replacement (ADR).

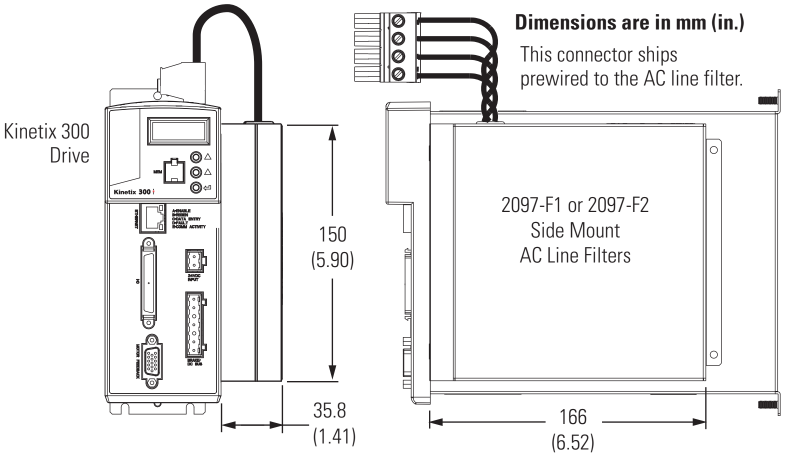




AC Line Filters

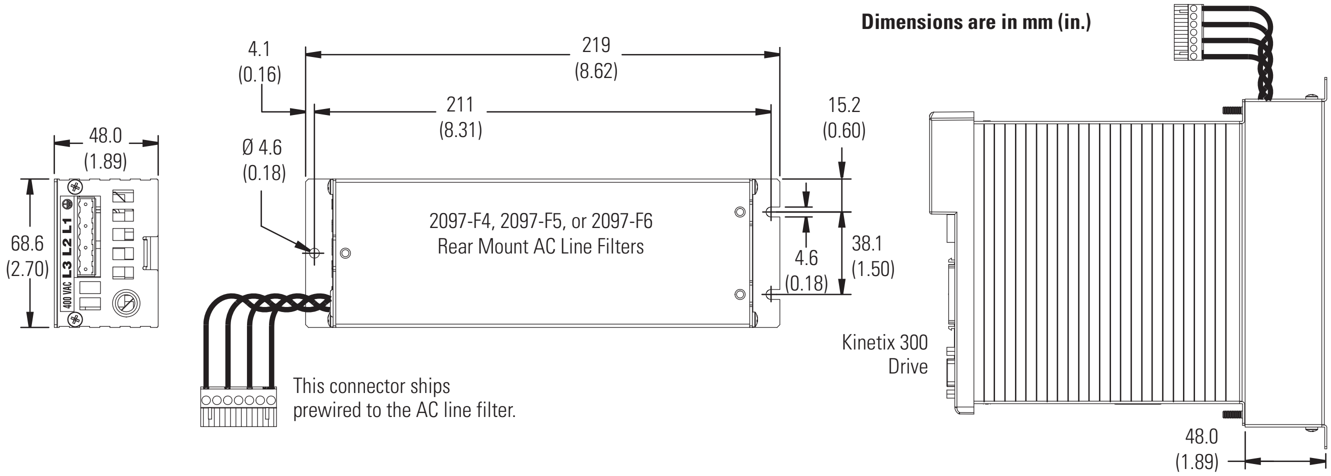
The Kinetix 300 drives were tested by using recommended line filters. Use of these filters is also needed to meet CE requirements. The 2097-V32PR0, 2097-V32PR2, and 2097-V32PR4 drives have integrated AC line filters.

AC Line Filter Dimensions



2097-F1 and 2097-F2 side-mount line filters add 35.8 mm (1.41 in.) to the width of your drive.

2097-F4, 2097-F5, and 2097-F6 rear-mount line filters add 48 mm (1.89 in.) to the depth of your drive.



AC Line Filter Specifications

AC Line Filter Cat. No.	Mount	Voltage 50/60 Hz	Phase	Current A @ 40 °C (104 °F)	Power Loss W	Leakage Current mA	Weight, approx. kg (lb)	Kinetix 300 Drive [1] Cat. No.
2097-F1	Side	120/240V AC	1	24.0	5.2	9.0	0.6 (0.13)	2097-V33PR6 [2]
2097-F2		480V AC	3	10.0	2.8	1.0		2097-V34PR6
2097-F4 [3]	Rear	120/240V AC	1 or 3	4.40	1.2		0.8 (0.18)	2097-V33PR1
		480V AC	3					2097-V34PR3
2097-F5 [3]				6.90	1.3			2097-V34PR5
		120/240V AC	1 or 3					2097-V33PR3
2097-F6 [3]			1	15.0	4.1			2097-V33PR5

(1) Use 2090-XXLF-TC116 (single-phase) line filter for 2097-V31PR0 and 2097-V31PR2 drives. Use 2090-UXLF-336 line filter for 2097-V33PR5 (three-phase) drive applications. Refer to AC Line Filter Specifications on page 484 for more information.

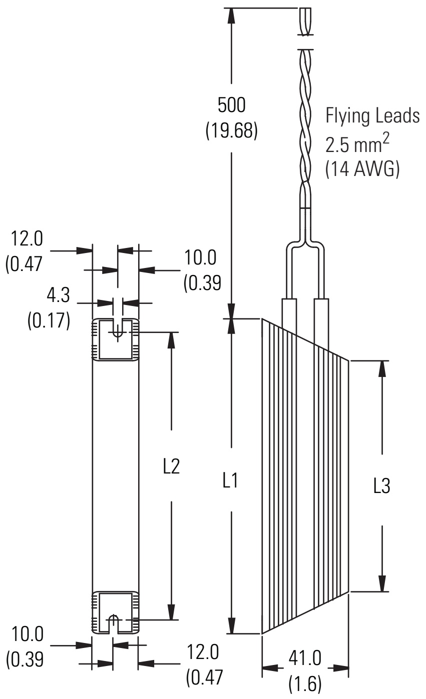
(2) Use the 2097-F1 (single-phase) line filter only for 2097-V33PR6 (single-phase) drive operation.

(3) This filter is rated for multiple voltage/phase line conditions.

Shunt Resistor Specifications

The Bulletin 2097 passive shunt resistor wires directly to the Kinetix 300 drive.

Shunt Resistor Dimensions



Dimensions are in mm (in.)

Shunt Resistor Cat. No.	L1	L2	L3
2097-R2	210 (8.3)	197 (7.7)	170 (6.7)
2097-R3	210 (8.3)	197 (7.7)	170 (6.7)
2097-R4	150 (5.9)	137 (5.4)	110 (4.3)
2097-R6	210 (8.3)	197 (7.7)	170 (6.7)
2097-R7	150 (5.9)	137 (5.4)	110 (4.3)

Shunt Resistor Power Specifications

Shunt Module Cat. No.	Specifications						Kinetix 300 Drive Cat. No.
	Resistance W	**Continuous Power** W	**Peak Power** kW	**Peak Current** A	**D_Application, Max** (1) %	**Weight** kg (lb)	
2097-R2	20	150	7.6	19.5	1.97	0.3 (0.7)	2097-V32PR4 2097-V33PR5
2097-R3	30		5.1	13.0	2.96		2097-V33PR6
2097-R4	40	80	3.8	9.8	2.10	0.2 (0.4)	2097-V31PR0 2097-V31PR2 2097-V32PR0 2097-V32PR2 2097-V33PR1 2097-V33PR3
2097-R6	75	150	7.9	10.3	1.90	0.3 (0.7)	2097-V34PR5 2097-V34PR6
2097-R7	150	80	4.0	5.1	2.02	0.2 (0.4)	2097-V34PR3

(1) D_Application is the application duty cycle in percent. For the intermittent regeneration applications, use D_Application = t/T, where t is the duration when regeneration is needed and T is the time interval between two regenerations. Both t and T must use the same time units, for example, seconds.

Kinetix 300 General System Specifications

This section contains environmental, weight, power dissipation, circuit breaker/fuse, transformer, and contactor specifications. Also included are maximum feedback cable length specifications and dimensions for mounting your Kinetix 300 drive.

Environmental Specifications

Attribute	Operational Range	Storage Range (nonoperating)
Temperature, ambient	0…40 °C (32…104 °F)	-10…70 °C (14…158 °F)
Relative humidity	5…95% noncondensing	5…95% noncondensing
Altitude	Derate by 1% per 300 m (1000 ft) above 1500 m (5000 ft)	3000 m (9842 ft) during transport
Vibration	5…2000 Hz @ 2.5 g peak, 0.015 mm (0.0006 in.) maximum displacement	
Shock	15 g, 11 ms half-sine pulse (3 pulses in each direction of 3 mutually perpendicular directions)	

Weight Specifications

Drive Cat. No.	Weight, approx. kg (lb)
2097-V31PR0	1.3 (2.9)
2097-V31PR2	1.5 (3.3)
2097-V32PR0	1.4 (3.1)
2097-V32PR2	1.7 (3.7)
2097-V32PR4	2.2 (4.9)
2097-V33PR1	1.3 (2.9)

Drive Cat. No.	Weight, approx. kg (lb)
2097-V33PR3	1.5 (3.3)
2097-V33PR5	2.0 (4.4)
2097-V33PR6	1.9 (4.2)
2097-V34PR3	1.5 (3.3)
2097-V34PR5	2.0 (4.4)
2097-V34PR6	1.8 (4.0)

Power Dissipation Specifications

Use this table to size an enclosure and calculate required ventilation for your Kinetix 300 drive system.

Drive Cat. No.	Loss, Max W
2097-V31PR0	28
2097-V31PR2	39
2097-V32PR0	28
2097-V32PR2	39
2097-V32PR4	67
2097-V33PR1	28

Drive Cat. No.	Loss, Max W
2097-V33PR3	39
2097-V33PR5	67
2097-V33PR6	117
2097-V34PR3	39
2097-V34PR5	58
2097-V34PR6	99

Circuit Breaker/Fuse Specifications

While circuit breakers offer some convenience, there are limitations for their use. Circuit breakers do not handle high current inrush as well as fuses.

Make sure the selected components are properly coordinated and meet acceptable codes including any requirements for branch circuit protection. Evaluation of the short-circuit available current is critical and must be kept below the short-circuit current rating of the circuit breaker.

Use class CC or T fast-acting current-limiting type fuses, 200,000 AIC, preferred. Use Bussman KTK-R, JJN, JJS, or equivalent. Thermal-magnetic type breakers preferred. The following fuse examples and Allen-Bradley circuit breakers are recommended for use with Kinetix 300 drives.

Fuse and Circuit Breaker Specifications

Drive Cat. No.	Drive Voltage	Main VAC		
		Bussmann Fuse	Allen-Bradley Circuit Breaker (1)	
			Disconnect (2)	Magnetic Contactor (3)
2097-V31PR0	120V	KTK-R-20 (20 A)	1492-SP1D200	140M-F8E-C20
	240V	KTK-R-10 (10 A)	1492-SP1D100	140M-F8E-C10
2097-V31PR2	120V	KTK-R-30 (30 A)	1492-SP1D300	140M-F8E-C32
	240V	KTK-R-20 (20 A)	1492-SP1D200	140M-F8E-C20
2097-V32PR0	240V	KTK-R-20 (20 A)	1492-SP3D200	140M-F8E-C20
2097-V32PR2				
2097-V32PR4	240V	KTK-R-30 (30 A)	1492-SP3D320	140M-F8E-C32
2097-V33PR1	120V	KTK-R-20 (20 A)	1492-SP1D200	140M-F8E-C20
	240V	KTK-R-15 (15 A)	1492-SP3D150	140M-F8E-C16
2097-V33PR3	120V	KTK-R-20 (20 A)	1492-SP1D200	140M-F8E-C20
	240V	KTK-R-15 (15 A)	1492-SP3D150	140M-F8E-C16
2097-V33PR5	120V	KTK-R-30 (30 A)	1492-SP1D300	140M-F8E-C32
	240V	KTK-R-20 (20 A)	1492-SP3D200	140M-F8E-C20
2097-V33PR6	120V	N/A	N/A	N/A
	240V	KTK-R-30 (30 A)	1492-SP3D300	140M-F8E-C32
2097-V34PR3	480V	KTK-R-10 (10 A)	1492-SP3D100	140M-F8E-C10
2097-V34PR5		KTK-R-10 (10 A)	1492-SP3D100	140M-F8E-C10
2097-V34PR6		KTK-R-20 (20 A)	1492-SP3D200	140M-F8E-C20

(1) When using Bulletin 1492 circuit protection devices, the maximum short circuit current available from the source is limited to 5000 A.

(2) Use fully-rated short-circuit protection circuit breaker for device branch circuit protection only when there is an upstream fully-rated breaker.

(3) Fully-rated breaker for overload current and short circuit rating.

Contactor Ratings

These table list the recommended contactor ratings for Kinetix 300 drives.

Kinetix 300 Drives (120/240V)

Cat. No.	Drive Voltage	AC Coil Contactor	DC Coil Contactor
2097-V31PR0	120V	100-C23x10	100-C23Zx10
	240V	100-C12x10	100-C12Zx10
2097-V31PR2	120V	100-C30x10	100-C30Zx10
	240V	100-C23x10	100-C23Zx10

Kinetix 300 Drives (240V)

Cat. No.	Drive Voltage	AC Coil Contactor	DC Coil Contactor
2097-V32PR0	240V	100-C23x10	100-C23Zx10
2097-V32PR2	240V	100-C23x10	100-C23Zx10
2097-V32PR4	240V	100-C30x10	100-C30Zx10
2097-V33PR1	120V	100-C23x10	100-C23Zx10
	240V	100-C16x10	100-C16Zx10
2097-V33PR3	120V	100-C23x10	100-C23Zx10
	240V	100-C16x10	100-C16Zx10
2097-V33PR5	120V	100-C30x10	100-C30Zx10
	240V	100-C23x10	100-C23Zx10
2097-V33PR6	120V	N/A	N/A
	240V	100-C30x10	100-C30Zx10

Kinetix 300 Drives (480V)

Cat. No.	Drive Voltage	AC Coil Contactor	DC Coil Contactor
2097-V34PR3	480V	100-C12x10	100-C12Zx10
2097-V34PR5		100-C12x10	100-C12Zx10
2097-V34PR6		100-C23x10	100-C23Zx10

Transformer Specifications for Control Input Power

Attribute	Value (460V system)
Input volt-amperes	750VA
Input voltage	460V AC
Output voltage	120…240V AC

Maximum Feedback Cable Lengths

Although motor power and feedback cables are available in standard lengths up to 90 m (295.3 ft), Kinetix 300 drive maximum feedback cable length is 20 m (65.6 ft). These tables assume the use of recommended cables as shown in the 2090-Series Motor/Actuator Cable Selection table on page 401.

Cable Lengths for Compatible Rotary Motors

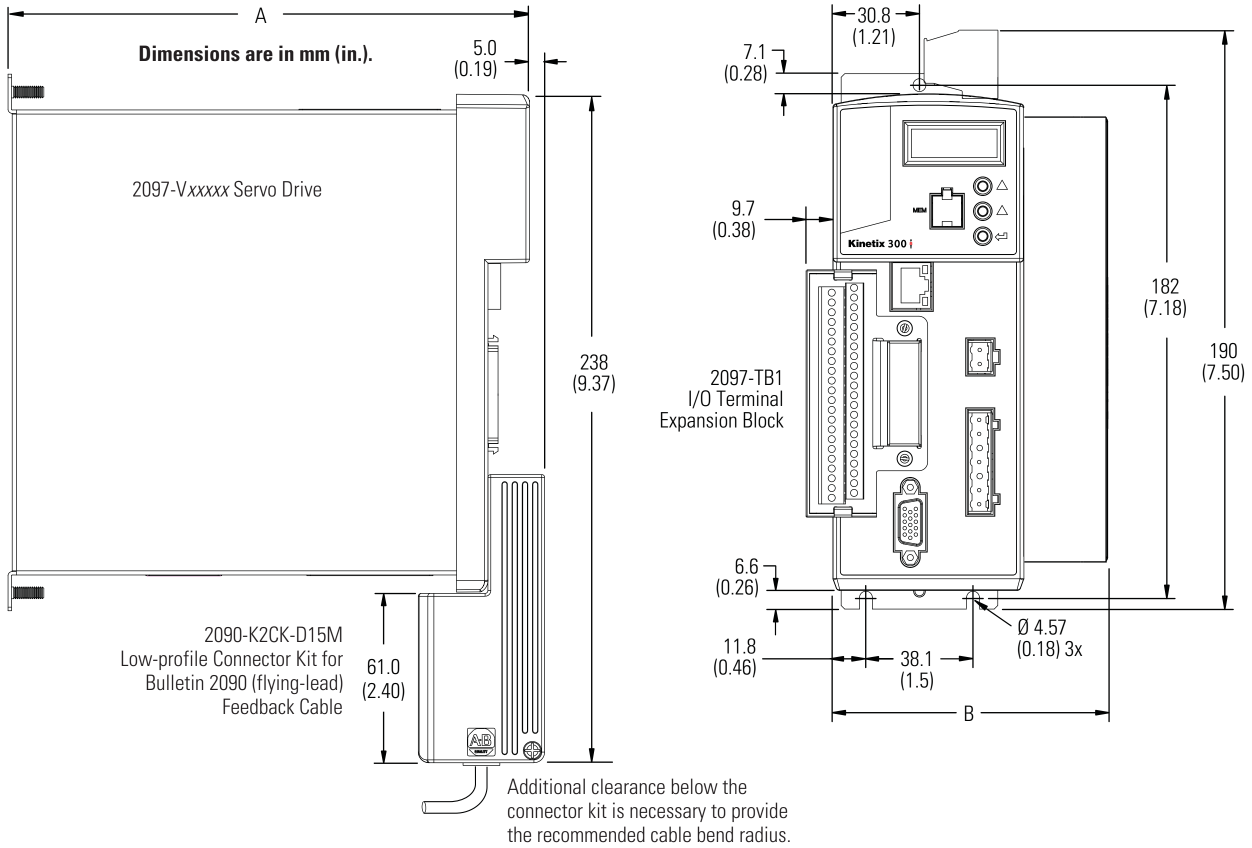
Motor Cat. No.	Absolute High-resolution (5V) Encoder m (ft)	Absolute High-resolution (9V) Encoder m (ft)	Incremental/TTL (5V) Encoder m (ft)	Absolute High-resolution (5V) 17-bit Encoder m (ft)
MPL-A3*xxx*… MPL-A5*xxx*-S/M [1]	20 (65.6)			
MPL-A15*xxx*… MPL-A2*xxx*-E/V	20 (65.6)			
MPL-B3*xxx*… MPL-B9*xxx*-S/M		20 (65.6)		
MPL-B15*xxx*… MPL-B2*xxx*-E/V		20 (65.6)		
MPL-A/B15*xxx*… MPL-A/B45*xxx*-H			20 (65.6)	
MPM-A*xxxxx*-S/M	20 (65.6)			
MPM-B*xxxxx*-S/M		20 (65.6)		
MPF-A*xxxx*-S/M [1]	20 (65.6)			
MPF-B*xxxx*-S/M		20 (65.6)		
MPS-A*xxxx*-S/M	20 (65.6)			
MPS-B*xxxx*-S/M		20 (65.6)		
TLY-A*xxxx*-B				20 (65.6)
TLY-A*xxxx*-H			20 (65.6)	

(1) MPL-A5*xxx* and MPF-A5*xxx* motor encoders are rated for 9V, the remaining Bulletin MPL and MPF (230V) motor encoders are rated for 5V.

Cable Lengths for Compatible Linear Actuators

Actuator Cat. No.	Absolute High-resolution (5V) Encoder m (ft)	Absolute High-resolution (9V) Encoder m (ft)	Absolute High-resolution (5V) 17-bit Encoder m (ft)
MPMA-A*xxxxx* or MPAS-A*xxxxx*-V (ballscrew)	20 (65.6)		
MPMA-B*xxxxx* or MPAS-B*xxxxx*-V (ballscrew)		20 (65.6)	
MPAR-A*xxxxx*-V/M	20 (65.6)		
MPAR-B*xxxxx*-V/M		20 (65.6)	
TLAR-A*xxxxx*-B			20 (65.6)
MPAI-A*xxxxx*M3	20 (65.6)		
MPAI-B*xxxxx*M3		20 (65.6)	

Kinetix 300 Drive Dimensions

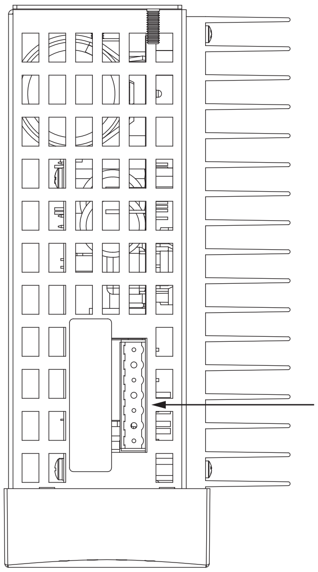


Additional clearance below the connector kit is necessary to provide the recommended cable bend radius.

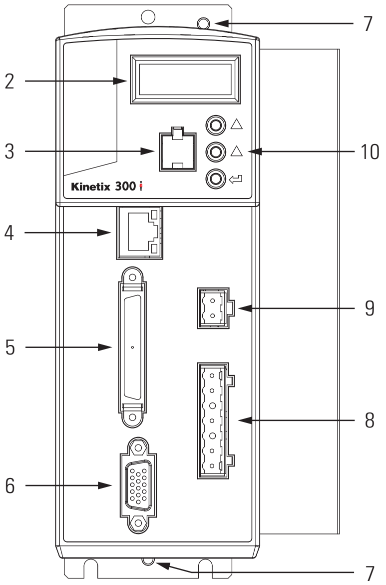
Kinetix 300 Dimensions

Cat. No.	A mm (in.)	B mm (in.)
2097-V31PR0	185 (7.29)	68.0 (2.68)
2097-V31PR2	185 (7.29)	69.0 (2.70)
2097-V32PR0	230 (9.04)	68.0 (2.68)
2097-V32PR2	230 (9.04)	69.0 (2.70)
2097-V32PR4	230 (9.04)	87.0 (3.42)
2097-V33PR1	185 (7.29)	68.0 (2.68)
2097-V33PR3	185 (7.29)	69.0 (2.70)
2097-V33PR5	185 (7.29)	94.0 (3.72)
2097-V33PR6	230 (9.04)	68.0 (2.68)
2097-V34PR3	185 (7.29)	69.0 (2.70)
2097-V34PR5	185 (7.29)	94.0 (3.72)
2097-V34PR6	230 (9.04)	68.0 (2.68)

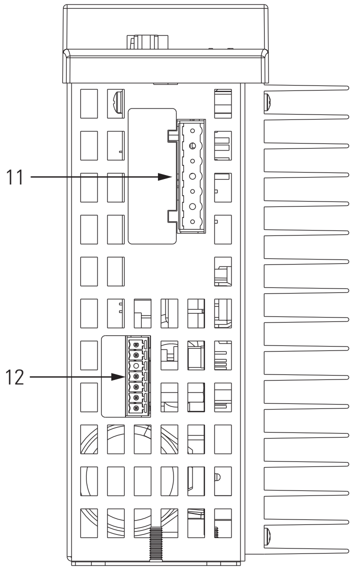
Kinetix 300 Connector and Indicator Locations



Kinetix 300 Drive, Top View
(2097-V33PR5 drive is shown)



Kinetix 300 Drive, Front View
(2097-V33PR5 drive is shown)



Kinetix 300 Drive, Bottom View
(2097-V33PR5 drive is shown)

Item	Description	Item	Description
1	Mains (IPD) connector	7	Ground lug
2	Status and diagnostic display	8	Shunt resistor and DC bus (BC) connector
3	Memory module socket	9	Back-up power (BP) connector
4	Ethernet communication port (Port 1)	10	Display control push buttons (3)
5	I/O (IOD) connector	11	Motor power (MP) connector
6	Motor feedback (MF) connector (1)	12	Safe torque-off (STO) connector

(1) The MF (15-pin) connector requires the 2090-K2CK-D15M low-profile connector kit.

For connector kit options, refer to Breakout Components and Connector Kits beginning on page 440.

Kinetix 300 Drive Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your drive. For questions regarding product availability, contact your Allen-Bradley distributor.

2097 - V3*x* PR*x*

Drive Power Rating
PR0 = 400 W
PR1 = 500 W
PR2 = 800 W
PR3 = 1.0 kW
PR4 = 1.7 kW
PR5 = 2.0 kW
PR6 = 3.0 kW

Drive Input Voltage
V31 = 120/240V AC, single-phase
V32 = 240V AC, single-phase with integrated AC (EMC) line filter
V33 = 120/240V AC, single-phase or three-phase
V34 = 480V AC, three-phase

Bulletin Number

Kinetix 3 Component Servo Drives

The Kinetix 3 component servo drive provides a cost-effective motion control solution for smaller, low-axis count applications. By providing the ability to apply the appropriate level of control for the application along with downloadable configuration software and automatic motor recognition, the Kinetix 3 servo drive delivers a motion solution that is easy to use at minimum cost. Its compact size and lower power ranges make it ideal for a variety of applications including indexing tables, medical manufacturing, laboratory automation equipment, and semiconductor processing.

Kinetix 3 Servo Drive Components

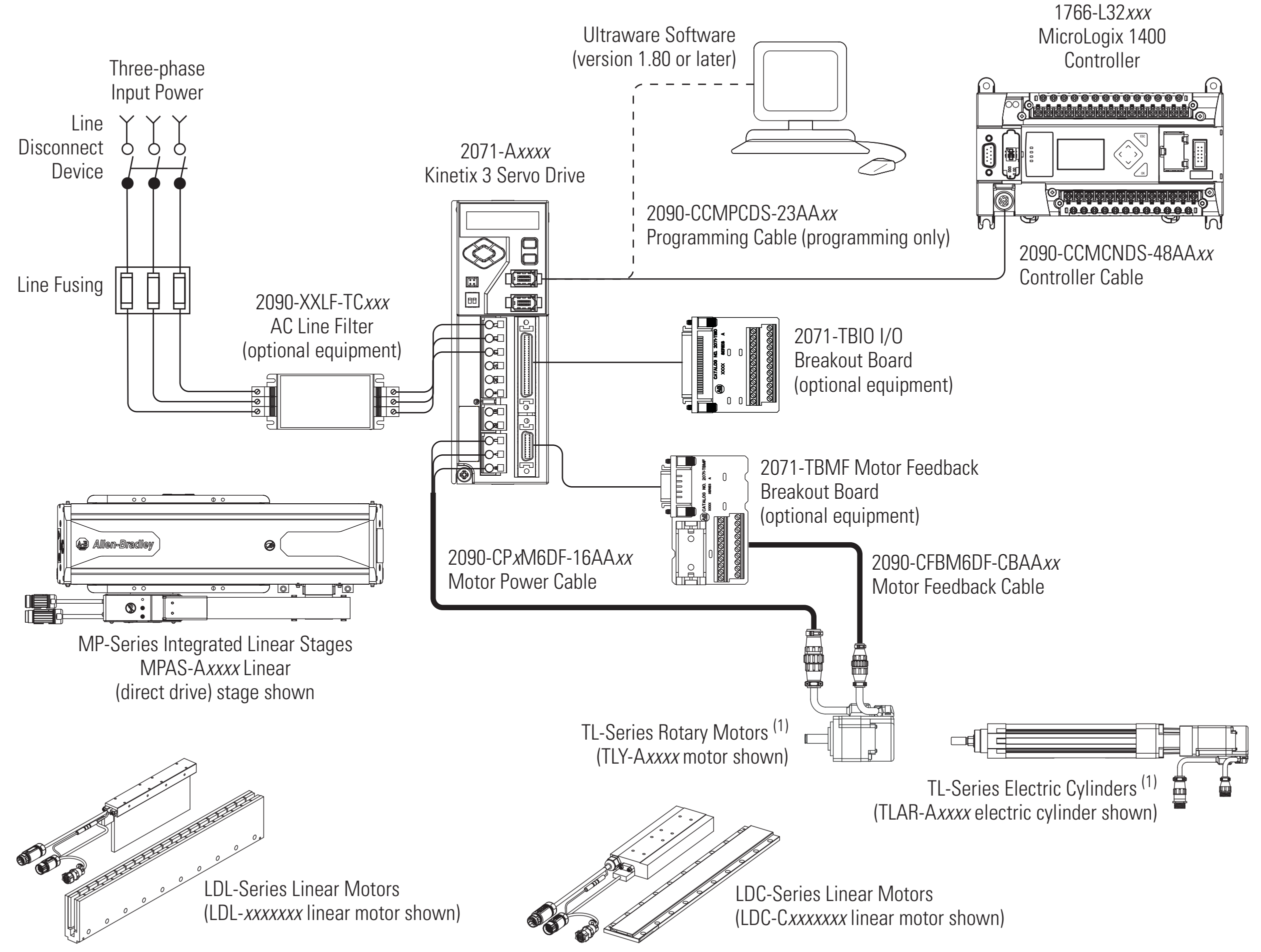
Kinetix 3 servo drive systems consist of these required components:

- One 2071-A*xxxx* servo drive
- One TL-Series servo motor/linear actuator, MP-Series linear stage, LDC-Series or LDL-Series linear motor
- One motor power and motor feedback cable

Kinetix 3 servo drive systems may also include any of these optional components:

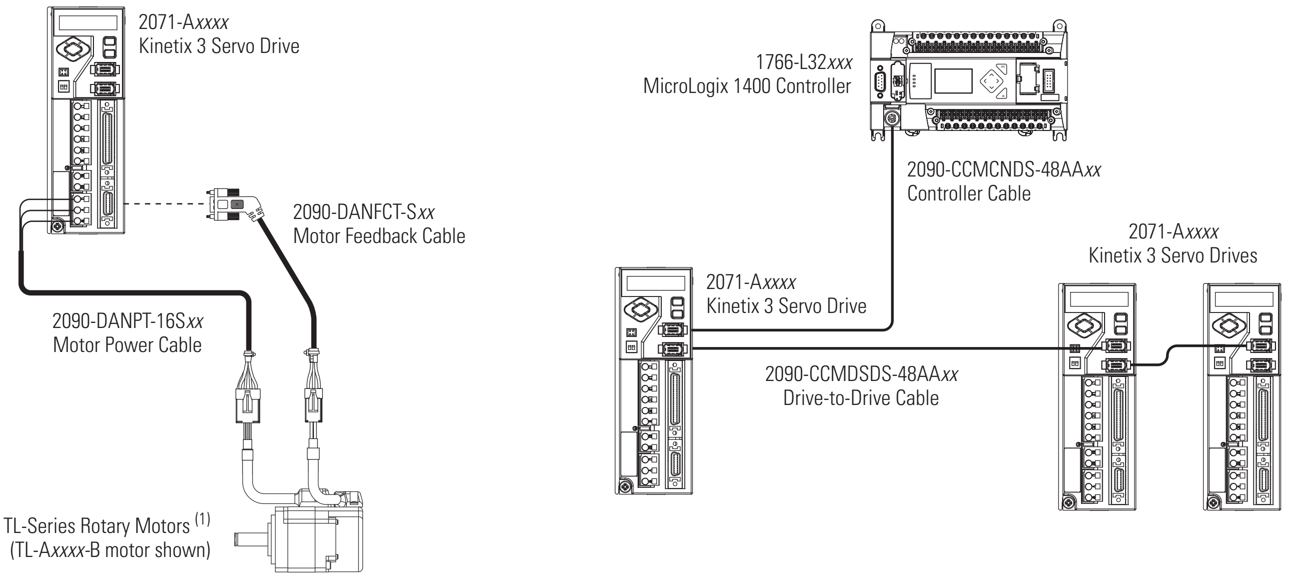
- One 2071-TBMF breakout board for motor feedback
- One 2071-TBIO breakout board for control interface
- 2090 control and configuration serial cables

Typical Configuration - Kinetix 3 Servo Drive System



(1) TL-Series (Bulletin TLY) rotary motors and Bulletin TLAR electric cylinders require the 2071-TBMF breakout board with 3.6V lithium battery (not included) for multi-turn high-resolution encoder operation. Other Kinetix 3 compatible motors and actuators require the breakout board for flying-lead feedback connections, but not the battery.

Typical Configuration - Kinetix 3 Servo Drive System (alternate configurations)



(1) TL-Series (Bulletin TL) rotary motors require the 2071-TBMF breakout board with 3.6V lithium battery (not included) for multi-turn high-resolution encoder operation or when using an overtravel limit switch. To meet this requirement, remove the drive-end connector and wire the 2090-DANFCT-S*xx* cable to the 2071-TBMF breakout board.

Kinetix 3 Drive Power Specifications

Attribute	2071-AP0	2071-AP1	2071-AP2	2071-AP4	2071-AP8	2071-A10	2071-A15
AC input voltage	170…264V rms						
AC input phase	Single-phase				Three-phase/ Single-phase	Three-phase	
AC input frequency	47…63 Hz						
Mains AC input current (1)							
Nom (rms)	1.30 A	2.38 A	3.68 A	7.14 A	6.25 A (three-phase) 10.52 A (single-phase)	8.75 A	12.37 A
Max inrush (0-pk)	21.9 A				22.6 A		
Mains AC line loss ride through	20 ms						
Nominal bus output voltage	311V DC						
Bandwidth (2)							
Velocity loop	550 Hz						
Current loop	2000 Hz						
PWM frequency	10 kHz						
Control power AC input voltage	170…264V rms						
Control power input current							
Nom (rms)	0.1 A rms						
Max inrush (0-pk)	31 A						
Continuous output current (rms)	0.61 A	1.11 A	1.72 A	3.33 A	5.05 A	7.07 A	9.90 A
Continuous output current (0-pk)	0.85 A	1.56 A	2.40 A	4.67 A	7.07 A	9.90 A	13.99 A
Peak output current (rms)	1.80 A	3.30 A	5.10 A	9.90 A	14.99 A	18.88 A	29.69 A
Peak output current (0-pk)	2.55 A	4.67 A	7.21 A	14.0 A	21.21 A	26.70 A	41.99 A
Continuous output power	50 W	100 W	200 W	400 W	800 W	1.0 kW	1.5 kW
Bus overvoltage	405V DC						
Bus undervoltage	190V DC						
Internal shunt resistor	N/A			50 Ω	30 Ω		
Internal shunt							
Continuous power	N/A			30 W	70 W		
Peak power	N/A			300 W	700 W		
Shunt on	390V DC						
Shunt off	380V DC						
Efficiency	90%						
Bus capacitance	390 µF	780 µF			1170 µF		
Capacitive energy absorption	13 J	26 J			39 J		
Short circuit current rating	100,000 A (rms) symmetrical						

(1) Kinetix 3 drive modules are limited to 1 AC mains power cycling every 2 minutes.

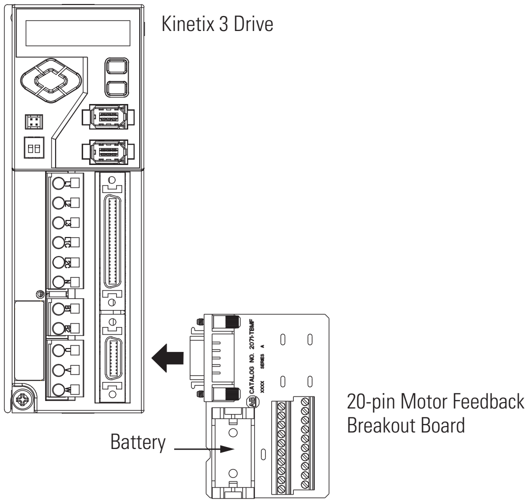
(2) Bandwidth values vary based on tuning parameters and mechanical components.

Kinetix 3 Drive Accessory Specifications

Kinetix 3 drive accessories include drive-mounted 20-pin and 50-pin breakout boards for making flying-lead motor feedback and I/O connections. For control and configuration serial interface cable specifications, refer to Interface Cable Applications and Standard Lengths on page 433.

Motor Feedback Breakout Board

Use the 2071-TBMF breakout board with 2090-CFBM6DF-CBAA*xx* flying-lead feedback cables or when your motor or actuator has high-resolution encoder feedback.

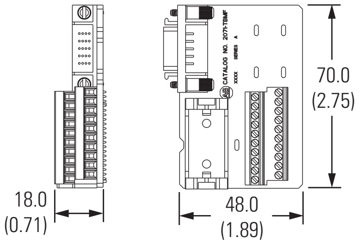


Motor Feedback Breakout Board Specifications

Attribute	Value
Wire size	1.5…0.2 mm^2 (16…24 AWG)
Change in depth of drive (1)	29 mm (1.14 in.)
Change in height of drive (1)	32 mm (1.26 in.)

(1) Add this value to the dimensions of your Kinetix 3 drive. Refer to Kinetix 3 Servo Drive Dimensions on page 359.

The customer-supplied 3.6V lithium battery, when installed in a motor feedback breakout board, provides multi-turn encoder operation to TL-Series (Bulletin TL, TLY, and TLAR) motors and actuators.



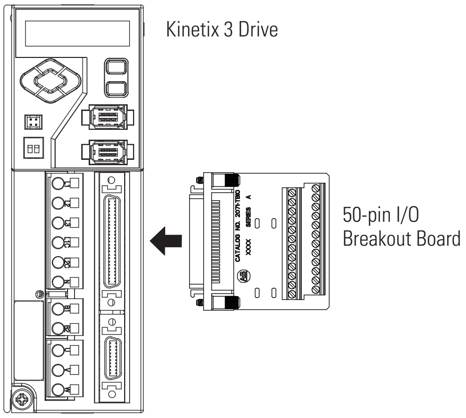
Dimensions are in mm (in.)

Battery Specifications

Attribute	Value
International size reference	1/2AA, ER14252
Nominal capacity @ 0.5 mA, to 2V	1.2 Ah
Rated voltage	3.6V
Max recommended continuous current	50 mA

I/O Breakout Board

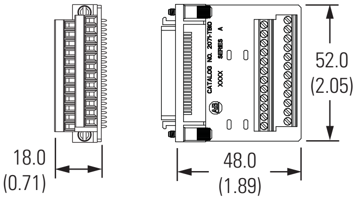
Use the 2071-TBIO breakout board for making flying-lead cable connections to twenty-four of the most commonly used terminals in the 50-pin IOD connector.



I/O Breakout Board Specifications

Attribute	Value
Wire size	1.5…0.2 mm^2 (16…24 AWG)
Change in depth of drive (1)	29 mm (1.14 in.)

(1) Add this value to the dimensions of your Kinetix 3 drive. Refer to Kinetix 3 Servo Drive Dimensions on page 359.



Kinetix 3 General System Specifications

This section contains environmental, weight, power dissipation, fuse/contactor, and maximum feedback cable length specifications.

Environmental Specifications

Attribute	Operational Range	Storage Range (nonoperating)
Temperature, ambient	0…50 ° C (32…122 ° F)	-25…85 ° C (-13…185 ° F)
Relative humidity	5…95% noncondensing	5…95% noncondensing
Altitude	1000 m (3281 ft) 3000 m (9843) with derating	3000 m (9843 ft) during transport
Vibration	5…55 Hz @ 0.35 mm (0.014 in.) double amplitude, continuous displacement; 55…500 Hz @ 2.0 g peak constant acceleration (10 sweeps in each of 3 mutually perpendicular directions).	
Shock	15 g, 11 ms half-sine pulse (3 pulses in each direction of 3 mutually perpendicular directions)	

Weight Specifications

Drive Cat. No.	Weight, approx. kg (lb)
2071-AP0	0.70 (1.5)
2071-AP1	0.75 (1.6)
2071-AP2	
2071-AP4	1.0 (2.2)
2071-AP8	1.75 (3.9)
2071-A10	
2071-A15	

Power Dissipation Specifications

Use this table to size an enclosure and calculate required ventilation for your Kinetix 3 drive system.

Drive Cat. No.	Usage as % of Rated Power Output (watts)				
	20%	40%	60%	80%	100%
2071-AP0	19.3	20.0	20.6	21.3	22.0
2071-AP1	20.1	21.6	23.1	24.6	26.2
2071-AP2	21.7	24.6	27.6	30.6	33.7
2071-AP4	25.6	31.9	38.4	45.2	52.2
2071-AP8	26.4	32.3	38.6	45.1	52.0
2071-A10	30.9	40.2	50.1	60.5	71.5
2071-A15	37.4	50.1	63.8	78.4	93.9

Fuse/Contactor Specifications

Make sure the selected components are properly coordinated and meet acceptable codes including any requirements for branch circuit protection. The following fuse examples are recommended for use with Kinetix 3 drives.

Fuse and Contactor Specifications

Drive Cat. No.	AC Input Power (1) (2) (3) Recommended Fuse	Control Power (2) (4) Recommended Fuse	Contactor (5)
2071-AP0	FNQ-R-7	FRS-R-2-1/2 FNQ-R-7-1/2 LPJ-6	100-M05N*xy*
2071-AP1			
2071-AP2			100-M09N*xy*
2071-AP2	FNQ -R-10		100-M12N*xy*
2071-AP8	FNQ-R-20 LPJ-20		100-C16*xy*
2071-A10			
2071- A15	FNQ-R-30 LPJ-30		100-C23*xy*

(1) Fuses specified are Bussmann fuses.
(2) FNQ-R fuses are described as time-delay fuses, Class CC.
(3) LPJ fuses are described as dual-element time-delay fuses, Class J.
(4) FRS-R fuses are described as dual-element time-delay fuses, Class RK5.
(5) For contactors: *x* represents coil voltage, *y* represents the number of contacts.

Maximum Feedback Cable Lengths

Although motor power and feedback cables are available in standard lengths up to 90 m (295.3 ft), Kinetix 3 drive maximum feedback cable length is 30 m (98.4 ft). These tables assume the use of recommended Bulletin 2090 cables as shown in the 2090-Series Motor/Actuator Cable Selection table on page 401.

Maximum Cable Lengths for Compatible Motors and Actuators

Rotary Motor Cat. No.	Incremental/TTL (5V) Encoder m (ft)	Absolute High-resolution (5V) 17-bit Encoder m (ft)
TL-A*xxxx*-B		30 (98.4)
TLY-A*xxxx*-B		
TLY-A*xxxx*-H	30 (98.4)	

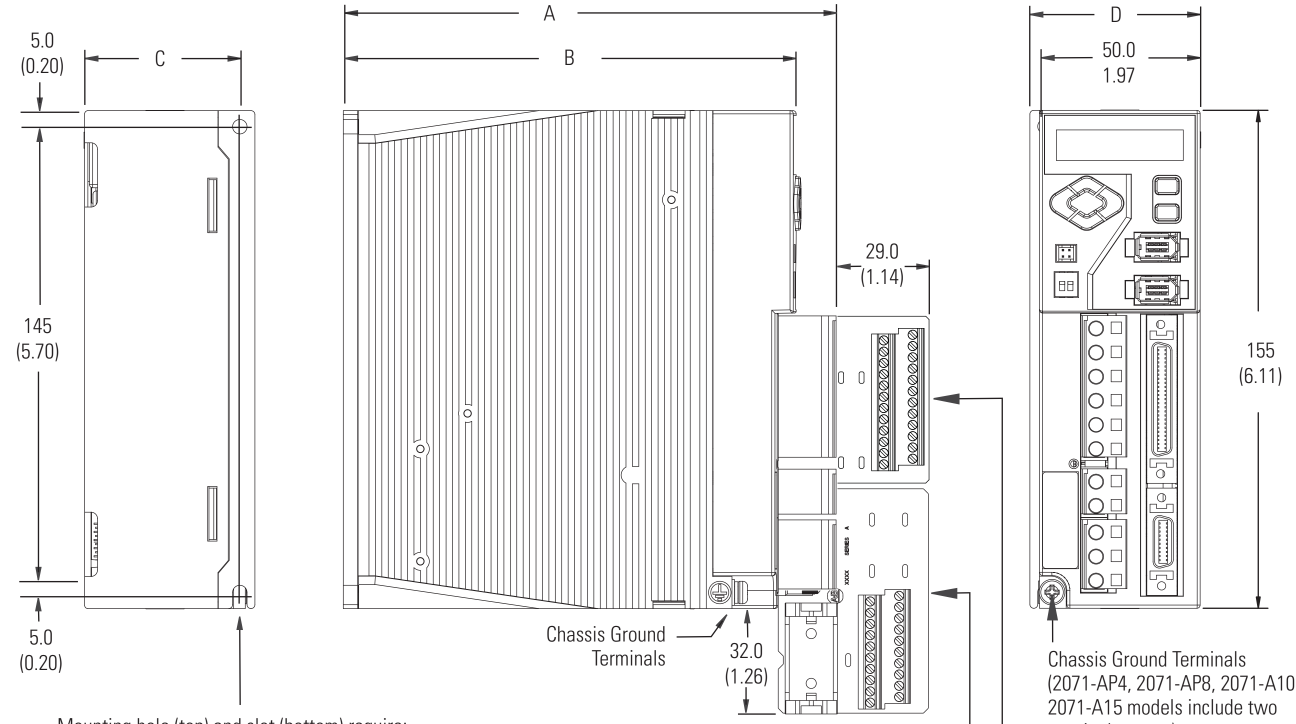
Actuator Cat. No.	Incremental/TTL (5V) Encoder m (ft)	Absolute High-resolution (5V) 17-bit Encoder m (ft)
MPAS-A*xxxx* (direct drive)	30 (98.4)	
TLAR-A*xxxxx*-B		30 (98.4)

Linear Motor Cat. No.	Incremental/TTL (5V) Encoder m (ft)
LDC-Series or LDL-Series	30 (98.4)

Kinetix 3 Servo Drive Dimensions

Dimensions are in mm (in.)

2071-AP0 drive is shown.



Chassis Ground Terminals

Mounting hole (top) and slot (bottom) require:
M4 x 10 bolts (catalog numbers 2071-AP0, 2071-AP1, 2071-AP2, and 2071-AP4)
M5 x 10 bolts (catalog numbers 2071-AP8, 2071-A10, 2071-A15)

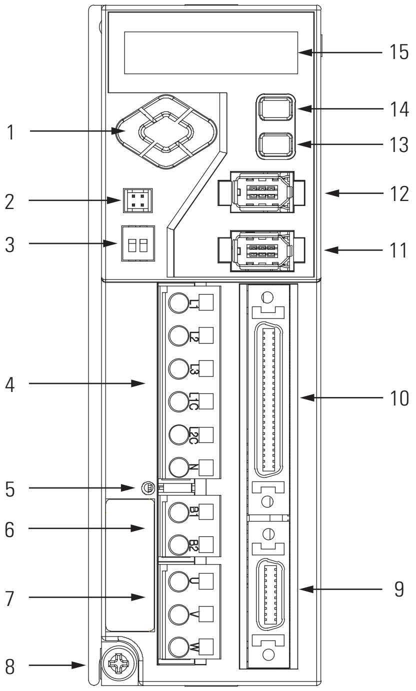
Chassis Ground Terminals (2071-AP4, 2071-AP8, 2071-A10, 2071-A15 models include two terminal screws)

2071-TBIO I/O Breakout Board

2071-TBMF Motor Feedback Breakout Board for Bulletin 2090 (flying-lead) Feedback Cables

Drive Cat. No.	**A** mm (in.)	**B** mm (in.)	**C** mm (in.)	**D** mm (in.)
2071-AP0	153 (6.04)	141 (5.55)	48.3 (1.90)	53.0 (2.09)
2071-AP1				
2071-AP2				
2071-AP4				58.0 (2.29)
2071-AP8	198 (7.82)	186 (7.33)	59.0 (2.32)	81.0 (3.19)
2071-A10				
2071-A15				

Kinetix 3 Connector and Indicator Locations



Item	Description
1	Left/right and up/down keys
2	Analog output (A.out)
3	RS-485 communication termination switch
4	Input power (IPD) connector
5	Main power status indicator
6	Shunt power (BC) connector
7	Motor power (MP) connector
8	Ground lug
9	Motor feedback (MF) connector
10	Input/output (IOD) connector
11	Serial interface (Comm0B) down
12	Serial interface (Comm0A) up
13	Enter key
14	Mode/set key
15	7-segment status indicator

For connector kit options, refer to Kinetix 3 Drive Accessory Specifications on page 356.

Kinetix 3 Servo Drive Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering chart below to understand the configuration of your drive. For questions regarding product availability, contact your Allen-Bradley distributor.

2071 - A *xx*

Continuous Output Current
P0 = 0.6 A rms
P1 = 1.1 A rms
P2 = 1.7 A rms
P4 = 3.3 A rms
P8 = 5.0 A rms
10 = 7.0 A rms
15 = 9.9 A rms

A = Amplifier

Bulletin Number

Ultraware Software Catalog Number

The Kinetix 3 drives are configured by using Ultraware software (catalog number 2098-UWCPRG), version 1.80 or later.

Ultra3000 Digital Servo Drives

The Ultra3000, Ultra3000i, Ultra3000-SE, Ultra3000-DN, and Ultra3000X-DN drives make up a family of flexible, high-performance digital servo drives for a variety of motion control applications and architectures. The wide range of power platforms, connectivity options and functions makes the Ultra3000 digital servo drive family an attractive solution for a variety of machine control architectures including Logix, SLC, and third-party machine and motion control systems. In addition, the Ultra3000i indexing, Ultra3000-SE, Ultra3000-DN, Ultra3000X-DN drives are positioning drives designed for applications requiring either simple or complex motion profiles.

Only the 2098-DSD-*xxx*-SE and 2098-DSD-HV*xxx*-SE (SERCOS interface) drives are part of the Kinetix Integrated Motion solution.

Ultra3000 Servo Drive Communication Interface

Drive Type	Drive Cat. No.	Command Interface
SERCOS interface drive	2098-DSD-*xxx*-SE and 2098-DSD-HV*xxx*-SE	Fiber-optic SERCOS ring
Analog drive	2098-DSD-*xxx* and 2098-DSD-HV*xxx*	Analog command interface
Digital drive with DeviceNet interface	2098-DSD-*xxx*-DN and 2098-DSD-HV*xxx*-DN	DeviceNet communication interface
Indexing drive	2098-DSD-*xxx*X and 2098-DSD-HV*xxx*X	Standalone control
Indexing DeviceNet interface drives	2098-DSD-*xxx*X-DN and 2098-DSD-HV*xxx*X-DN	

Ultra3000 Digital Servo Drive Components

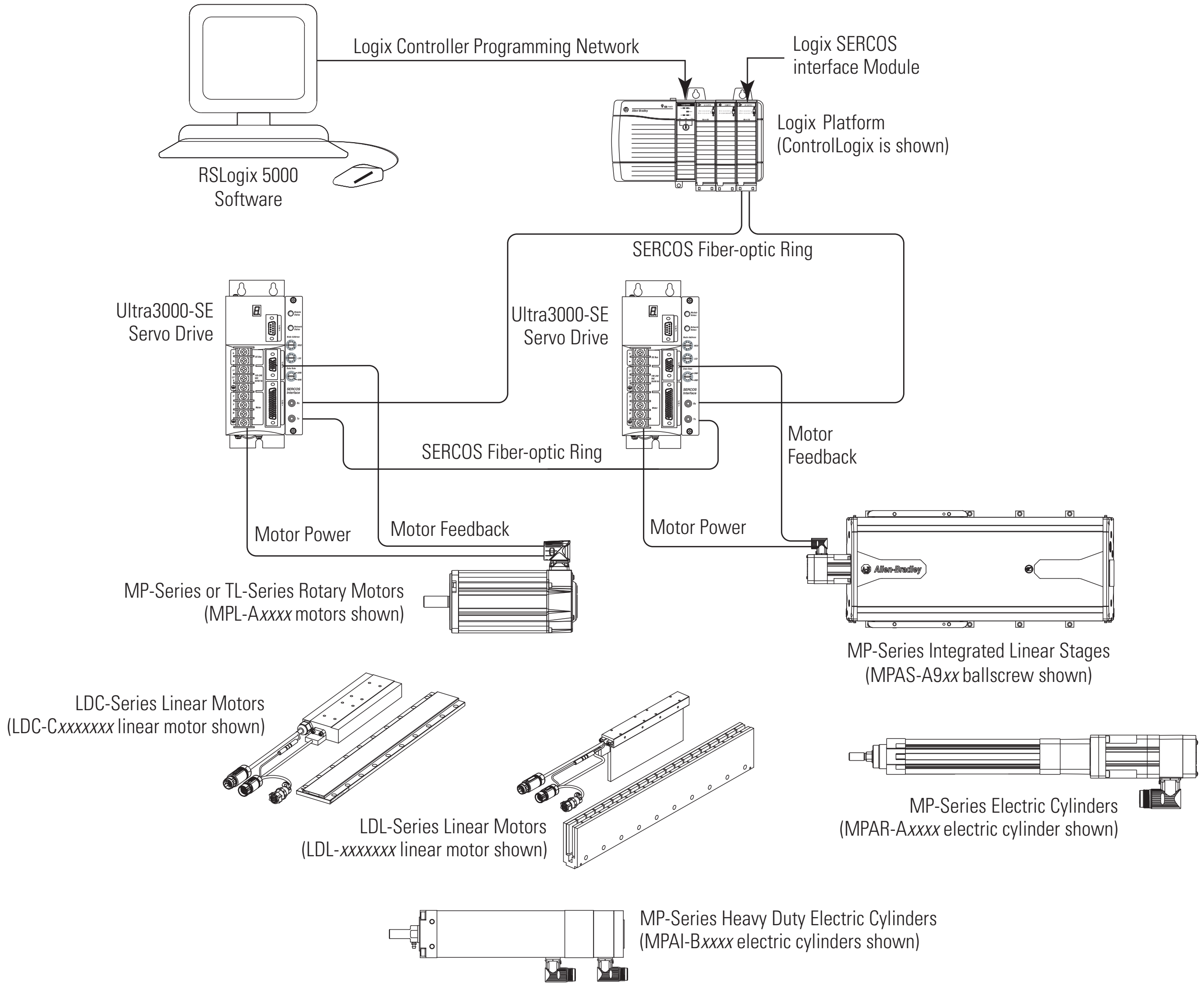
Ultra3000 digital servo drive systems consist of these required components:

- One Ultra3000 digital servo drive
- One rotary servo motor or linear motor/actuator (MP-Series, TL-Series, LDC-Series, or LDL-Series)
- One motor power and feedback cable
- Two SERCOS fiber-optic cables for Ultra3000-SE drives

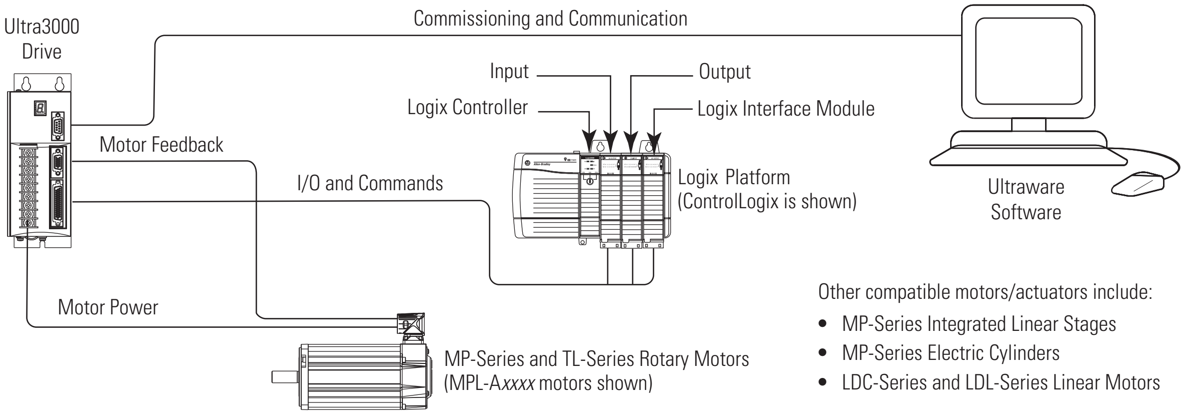
Ultra3000 systems may also include any of these optional components:

- Bulletin 2090 external active or passive shunt module
- Bulletin 2090 Resistive Brake Module (RBM)

Typical Configuration - Ultra3000-SE (SERCOS) Digital Servo Drive System



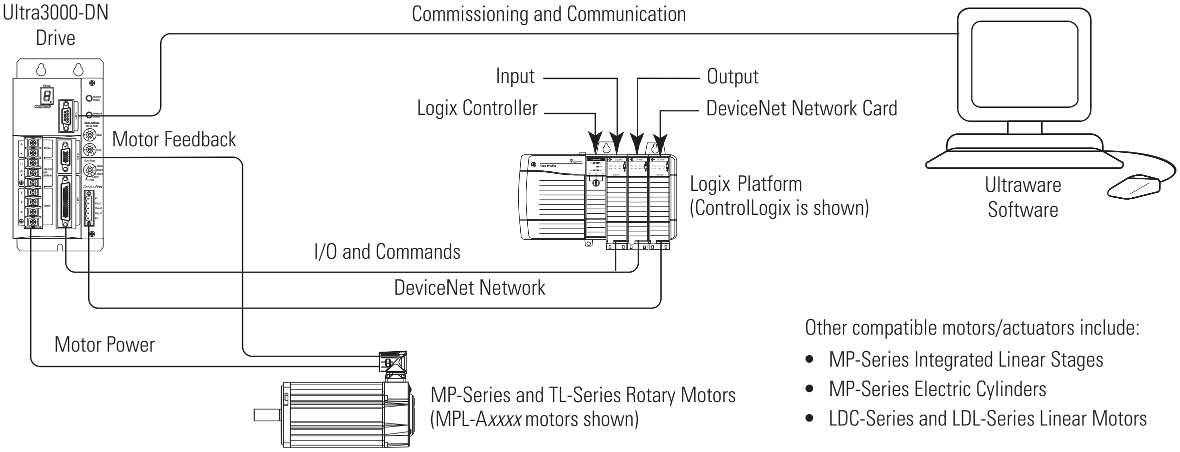
Typical Configuration - Ultra3000 Digital Servo Drive System



Other compatible motors/actuators include:

- MP-Series Integrated Linear Stages
- MP-Series Electric Cylinders
- LDC-Series and LDL-Series Linear Motors

Typical Configuration - Ultra3000-DN (DeviceNet) Digital Servo Drive System



Other compatible motors/actuators include:

- MP-Series Integrated Linear Stages
- MP-Series Electric Cylinders
- LDC-Series and LDL-Series Linear Motors

Ultra3000 Digital Servo Drive Specifications

This section contains general power, physical/environmental, power dissipation, controller, I/O, operating modes command sources, serial communication, network communication, feedback, and connector specifications for the Ultra3000 digital servo drives.

Specifications apply to these Ultra3000 drive models:

- SE indicates the 2098-DSD-*xxx*-SE SERCOS interface drive
- DN indicates the 2098-DSD-*xxx*-DN DeviceNet interface drive
- X indicates the 2098-DSD-*xxx*X indexing drive
- X-DN indicates the 2098-DSD-*xxx*X-DN indexing DeviceNet interface drive

Power Specifications

2098-DSD-005*x-xx*, 2098-DSD-010*x-xx*, and 2098-DSD-020*x-xx* Ultra3000 (230V) Drives

Attribute	2098-DSD-005	2098-DSD-010	2098-DSD-020
AC input voltage (1)	100…240V rms Single-phase		
AC input frequency	47…63 Hz		
AC input current (2) (3) Nom (rms) 230V AC (0-pk) max inrush (4)	 5 A 100 A - Series A or B 20 A - Series C	 9 A 100 A - Series A or B 20 A - Series C	 18 A 100 A - Series A or B 20 A - Series C
Continuous output current (rms)	1.8 A	3.5 A	7.1 A
Continuous output current (0-pk)	2.5 A	5.0 A	10 A
Peak output current (rms)	5.3 A	10.6 A	21.2 A
Peak output current (0-pk)	7.5 A	15 A	30 A
Bus capacitance	1410 μF	1880 μF	1880 μF
Internal shunt resistance	N/A	N/A	N/A
Shunt on	N/A	N/A	N/A
Shunt off	N/A	N/A	N/A
Bus overvoltage	400V DC	400V DC	400V DC
Energy absorption capability 115V AC input 230V AC input	 93 J 38 J	 125 J 51 J	
Continuous power output 115V AC input 230V AC input	 0.25 kW 0.5 kW	 0.5 kW 1.0 kW	 1.0 kW 2.0 kW

(1) Specification is for nominal voltage. The absolute limits are ±10%, or 88…265V rms.

(2) The 2098-DSD-005*x-xx*, 2098-DSD-010*x-xx*, and 2098-DSD-020*x-xx* (230V) drives are limited to:
Series A or B - one contactor cycle every two minutes.
Series C - one contactor cycle every 10 s for up to two minutes, not to exceed 12 cycles in five minutes.

(3) Power initialization requires a short period of inrush current. Dual element time delay (slow blow) fuses are recommended (refer to Fuse Specifications on page 367).

(4) In-rush current limiting circuitry is enabled within 3 s after removal of AC line power.

2098-DSD-030*x-xx*, 2098-DSD-075*x-xx*, and 2098-DSD-150*x-xx* Ultra3000 (230V) Drives

Attribute	2098-DSD-030	2098-DSD-075	2098-DSD-150
AC input voltage (1)	100…240V rms Single-phase	240V rms Three-phase	
AC input frequency	47…63 Hz		
Main AC input current (2) (3) Nom (rms) 230V AC (0-pk) max inrush	 28 A 50 A	 30 A 50 A	 46 A 68 A
Auxiliary AC input current 115V AC (rms) nom 230V AC (rms) nom 115V AC (0-pk) max inrush (4) 230V AC (0-pk) max inrush (4)	 1.0 A 0.5 A 47 A 95 A	 1.0 A 0.5 A 47 A 95 A	 1.0 A 0.5 A 47 A 95 A
Continuous output current (rms)	10.6 A	24.7 A	45.9 A
Continuous output current (0-pk)	15 A	35 A	65 A
Peak output current (rms)	21.2 A	53 A	106 A
Peak output current (0-pk)	30 A	75 A	150 A
Bus capacitance	2820 µF	4290 µF	7520 µF
Internal shunt resistance	35 Ω	16.5 Ω	9.1 Ω
Shunt on	420V DC	420V DC	420V DC
Shunt off	402V DC	402V DC	402V DC
Bus overvoltage	452V DC	452V DC	452V DC
Internal shunt Continuous power Peak power	 50 W 4.5 kW	 50 W 10 kW	 180 W 18 kW
External shunt Resistance Continuous power Peak power	 30 Ω(-0/+5%) 2.4 kW 6 kW	 16.5 Ω(-0/+5%) 4 kW 10 kW	 9 Ω(-0/+5%) 8 kW 19 kW
Energy absorption capability 115V AC input 230V AC input	 251 J 139 J	 381 J 211 J	 669 J 370 J
Continuous power output 115V AC input 230V AC input	 1.5 kW 3 kW	 N/A 7.5 kW	 N/A 15 kW

(1) Specification is for nominal voltage. The absolute limits are ±10%, or 88…265V rms.

(2) The 2098-DSD-030*x-xx,* 2098-DSD-075*x-xx,* and 2098-DSD-150*x-xx* (230V) drives are limited to one contactor cycles per two minutes.

(3) Power initialization requires a short period of inrush current. Dual element time delay (slow blow) fuses are recommended (refer to Fuse Specifications on page 367).

(4) 400 µs half wave sine.

Ultra3000 460V Drives

Attribute	2098-DSD-HV030	2098-DSD-HV050	2098-DSD-HV100	2098-DSD-HV150	2098-DSD-HV220
AC input voltage (1) (2)	230/480V rms Three-phase				
AC input Frequency	47…63 Hz				
Main AC input current (3) (4)					
460V AC (rms) nom	4 A	7 A	14 A	20 A	28 A
460V AC (rms) max inrush	6 A	6 A	6 A	6 A	6 A
Auxiliary AC input current					
230V AC (rms) nom	0.55 A				
360V AC (rms) nom	0.35 A				
480V AC (rms) nom	0.25 A				
230V AC (0-pk) max inrush (5)	47 A				
480V AC (0-pk) max inrush (5)	68 A				
Continuous output current (rms)	5.0 A	7.8 A	16.3 A	24.0 A	33.2 A
Continuous output current (0-pk)	7.0 A	11 A	23 A	34 A	47 A
Peak output current (rms)	9.9 A	15.6 A	32.5 A	48.1 A	66.5 A
Peak output current (0-pk)	14 A	22 A	46 A	68 A	94 A
Bus capacitance	470 µF		705 µF	940 µF	1880 µF
Internal shunt resistance	120 Ω		40 Ω	25 Ω	20 Ω
Shunt on					
230V AC input	400V DC				
480V AC input	800V DC				
Shunt off					
230V AC input	375V DC				
480V AC input	750V DC				
Bus overvoltage					
230V AC input	410V DC				
480V AC input	810V DC				
Internal shunt					
Continuous power	100 W		200 W	200 W	400 W
Peak power	5.3 kW		16 kW	25.6 kW	32 kW
External shunt					
Resistance (-0/+5%)	120 Ω		40 Ω	25 Ω	20 Ω
Continuous power	3 kW		10 kW	15 kW	22 kW
Peak power	5.3 kW		16 kW	25.6 kW	32 kW
Energy absorption capability					
230V AC input with 230V motor	15 J		22 J	29 J	59 J
230V AC input with 460V motor	129 J		194 J	259 J	517 J
480V AC input	55 J		82 J	109 J	219 J
Continuous power output					
230V AC input	1.5 kW	2.5 kW	5.0 kW	7.5 kW	11 kW
480V AC input	3.0 kW	5.0 kW	10 kW	15 kW	22 kW

(1) Specification is for nominal voltage. The absolute limits are ±10%, or 207…264V rms and 324…528V rms.

(2) The 2098-DSD-HV*xxx-xx* drives can be powered with 230V rms and used with motors designed for 230V operation. In such cases, the voltage levels used for shunting and DC bus overvoltage limits are adjusted to be compatible with the voltage limit of the motor.
The 2098-DSD-HV*xxx-xx* drives can be powered with 480V rms and used with motors designed for 480V operation. In such cases, the voltage levels used for shunting and DC bus overvoltage limits are adjusted to be compatible with the voltage limit of the motor.

(3) The 2098-DSD-HV*xxx -xx* (460V) drives are limited to three contactor cycles per minute.

(4) Power initialization requires a short period of inrush current (processor controlled via soft start circuitry). Dual element time delay (slow blow) fuses are recommended (refer to Fuse Specifications on page 367).

(5) 400 µs half wave sine.

Fuse Specifications

Use class CC, G, J, L, R, or T fuses, with current ratings as indicated in the table below. The table below lists fuse examples recommended for use with the Ultra3000 (230V and 460V) drives.

Fuse Examples for Ultra3000 Drives

Drive Cat. No.	Input Voltage	Input Type	Recommended Fuse	
			Class CC [1]	Class J [1]
2098-DSD-005	230V AC	Input Power	FNQ-R-6	LPJ-6SP
2098-DSD-010			FNQ-R-10	LPJ-10SP
2098-DSD-020			FNQ-R-20	LPJ-20SP
2098-DSD-030			FNQ-R-30	LPJ-30SP
2098-DSD-075			FNQ-R-30	LPJ-30SP
2098-DSD-150			N/A	LPJ-60SP
2098-DSD-*xxx*		Auxiliary Input Power	FNQ-R-10	LPJ-10SP
2098-DSD-HV030	460V AC	Input Power	KTK-R-5	LPJ-5SP
2098-DSD-HV050			KTK-R-8	LPJ-8SP
2098-DSD-HV100			KTK-R-20	LPJ-17-1/2SP
2098-DSD-HV150			KTK-R-30	LPJ-30SP
2098-DSD-HV220			N/A	LPJ-35SP
2098-DSD-HV*xxx*		Auxiliary Input Power	FNQ-R-10	LPJ-10SP

(1) Part numbers shown are examples of Bussmann fuses.

Circuit Breaker Specifications

While circuit breakers offer some convenience, there are limitations for their use. Circuit breakers do not handle high current inrush as well as fuses.

Make sure the selected components are properly coordinated and meet acceptable codes including any requirements for branch circuit protection. Evaluation of the short-circuit available current is critical and must be kept below the short-circuit current rating of the circuit breaker.

Circuit Breaker Examples for Ultra3000 (460V) Drives

Drive Cat. No.	Input Voltage	Circuit Breakers
2098-DSD-HV030	460V	140M-F8E-C16
2098-DSD-HV050		140M-F8E-C20
2098-DSD-HV100		140M-F8E-C32
2098-DSD-HV150		140M-F8E-C45
2098-DSD-HV220		N/A

Contactor Ratings

Drive Cat. No.	Input Voltage	Contactor
2098-DSD-HV030	460V	100-C23*x*10 (AC coil) 100-C23Z*x*10 (DC coil)
2098-DSD-HV050		100-C30*x*10 (AC coil) 100-C30Z*x*10 (DC coil)
2098-DSD-HV100		100-C37*x*10 (AC coil) 100-C37Z*x*10 (DC coil)
2098-DSD-HV150		100-C43*x*10 (AC coil) 100-C43Z*x*10 (DC coil)
2098-DSD-HV220		100-C60*x*10 (AC coil) 100-C60Z*x*10 (DC coil)

Power Dissipation Specifications

Drive Cat. No.	Max Loss W
2098-DSD-005	48
2098-DSD-010	48
2098-DSD-020	50
2098-DSD-030	150 + dissipative shunt
2098-DSD-075	300 + dissipative shunt
2098-DSD-150	500 + dissipative shunt

Drive Cat. No.	Max Loss W
2098-DSD-HV030	175 + dissipative shunt
2098-DSD-HV050	175 + dissipative shunt
2098-DSD-HV100	350 + dissipative shunt
2098-DSD-HV150	350 + dissipative shunt
2098-DSD-HV220	600 + dissipative shunt

Communication Specifications

Attribute	Value
SERCOS (option)	
Communication rates	4 and 8 Mbps
Node addresses	01…99
DeviceNet (option)	
Power consumption from network	60 mA
Data rates	125, 250, and 500 kps, and auto-baud
Node addresses	00…63
Messaging capabilities	Explicit, Polled I/O, Change of State, and Cyclic Messaging
Serial	
Ports	One RS-232/RS-422/RS-485
Communication rates	1200, 2400, 4800, 9600, 19,200, and 38,400 bps

Inputs/Outputs Specifications

Attribute	Value
Digital inputs	8 optically isolated, 12…24V, active high, current sinking
Digital outputs	4 optically isolated, 12…24V, active high, current sourcing
Relay output	One normally open relay, 30V DC, max, 1 A, max
I/O response	100 µs
Digital I/O firmware scan period	1 ms
Analog inputs COMMAND ILIMIT	 14-bit A/D, ±10V 10-bit A/D, 0 to 10V
Analog output	±10V, 8 bits, 2 mA max

Auxiliary Feedback Specifications

Attribute	Value
Input modes	A quad B, Step/Direction, CW/CCW
Maximum input frequency	2.5 MHz
Input types	Differential, single-ended, open collector (1)

(1) Differential input types are recommended.

Physical and Environmental Specifications

Attribute	Value		Attribute	Value	
Weight, approx. 2098-DSD-005 2098-DSD-010 2098-DSD-020 2098-DSD-030 2098-DSD-075 2098-DSD-150	 1.80 kg 2.10 kg 2.10 kg 6.20 kg 9.30 kg 14.1 kg	 (4.1 lb) (4.6 lb) (4.6 lb) (13.6 lb) (20.6 lb) (31.0 lb)	Weight, approx. 2098-DSD-HV030 2098-DSD-HV050 2098-DSD-HV100 2098-DSD-HV150 2098-DSD-HV220	 8.55 kg 8.55 kg 10.44 kg 10.44 kg 14.1 kg	 (18.8 lb) (18.8 lb) (22.9 lb) (22.9 lb) (31.0 lb)
Temperature, ambient Operating Storage	 0…55 °C (32…131 °F) -40…70 °C (-40…158 °F)				
Relative humidity	5…95% noncondensing				
Altitude	1500 m (4921.5 ft) - Derate 3% per 300 m (984.3 ft) above 1500 m (4921.5 ft)				
Vibration	5…2000 Hz @ 2.5 g peak, 0.0006 mm (0.015 in.) max displacement				
Shock	15 g, 11 ms half-sine				

Connector Specifications

Connector	Description	Specification
CN1	User input/output	44-pin high-density female D-sub connector
CN2	Motor feedback connector	15-pin high-density female D-sub connector
CN3	Serial port connector	9-pin female D-sub connector
TB1 and TB2	Main and auxiliary AC, DC bus, motor power, and shunt connectors	Screw terminal block

Maximum Feedback Cable Lengths

Although motor feedback cables are available in standard lengths up to 90 m (295.3 ft), the drive/motor/feedback combination may limit the maximum cable length, as shown in the tables below. These tables assume the use of cables recommended in 2090-Series Motor/Actuator Cable Selection table on page 401.

Maximum Cable Lengths for Compatible Rotary Motors

Motor Cat. No.	**Absolute High-resolution (5V) Encoder** m (ft)	**Absolute High-resolution (9V) Encoder** m (ft)	**Incremental/TTL (5V) Encoder** m (ft)
MPL-A15*xxx*…MPL-A2*xxx*-E/V	90 (295.3)		
MPL-A3*xxx*…MPL-A5*xxx*-S/M (1)	90 (295.3)		
MPL-B15*xxx*…MPL-B2*xxx*-E/V		90 (295.3)	
MPL-B3*xxx*…MPL-B9*xxx*-S/M		90 (295.3)	
MPL-A/B15*xxx*…MPL-A/B45*xxx*-H			45 (147.6)
MPM-A*xxxxx*-S/M	30 (98.4)		
MPM-B*xxxxx*-S/M		90 (295.3)	
MPF-A*xxxx*-S/M (1)	90 (295.3)		
MPF-B*xxxx*-S/M		90 (295.3)	
MPS-A*xxxx*-S/M	90 (295.3)		
MPS-B*xxxx*-S/M		90 (295.3)	
TLY-A*xxxx*-H			30 (98.4)

(1) MPL-A5*xxx* and MPF-A5*xxx* motor encoders are rated for 9V, the remaining Bulletin MPL and MPF (230V) motor encoders are rated for 5V.

Maximum Cable Lengths for Compatible Linear Actuators

Actuator Cat. No.	**Absolute High-resolution (5V) Encoder** m (ft)	**Absolute High-resolution (9V) Encoder** m (ft)	**Incremental/TTL (5V) Encoder** m (ft)
MPMA-A*xxxxx* or MPAS-A*xxxxx*-V (ballscrew)	90 (295.3)		
MPMA-A*xxxxx* or MPAS-A*xxxxx*-A (direct drive)			45 (147.6)
MPMA-B*xxxxx* or MPAS-B*xxxxx*-V (ballscrew)		90 (295.3)	
MPMA-B*xxxxx* or MPAS-B*xxxxx*-A (direct drive)			45 (147.6)
MPAR-A*xxxxx*-V/M	30 (98.4)		
MPAR-B*xxxxx*-V/M		90 (295.3)	
MPAI-A*xxxxx*M3	30 (98.4)		
MPAI-B*xxxxx*M3		90 (295.3)	

Maximum Cable Lengths for Compatible Linear Motors

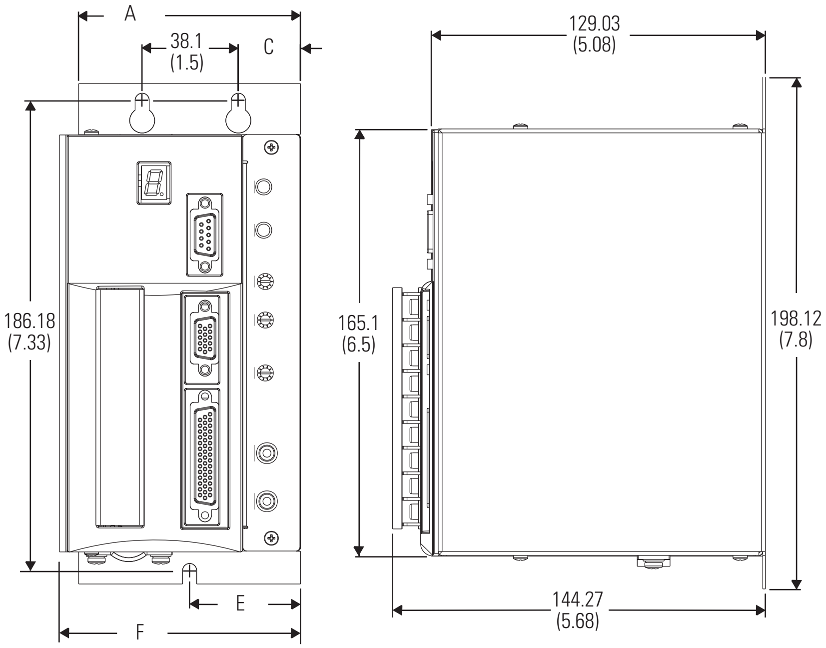
Motor Cat. No.	**Absolute High-resolution (5V) Encoder** m (ft)	**Incremental/TTL (5V) Encoder** m (ft)
LDC-Series or LDL-Series	30 (98.4)	30 (98.4)

Ultra3000 Digital Servo Drive Dimensions

This section contains dimensions for the Ultra3000 digital servo drives (X indicates indexing, -SE indicates SERCOS interface, -DN indicates DeviceNet interface, and X-DN indicates indexing DeviceNet interface).

In the figure below, -*xxx* is replaced by -005, -010, or -020 to represent the Ultra3000 500 W, 1 kW, and 2 kW drives respectively.

2098-DSD-*xxx*, 2098-DSD-*xxx*X, 2098-DSD-*xxx*-SE, 2098-DSD-*xxx*-DN, 2098-DSD-*xxx*X-DN Dimensions (230V)



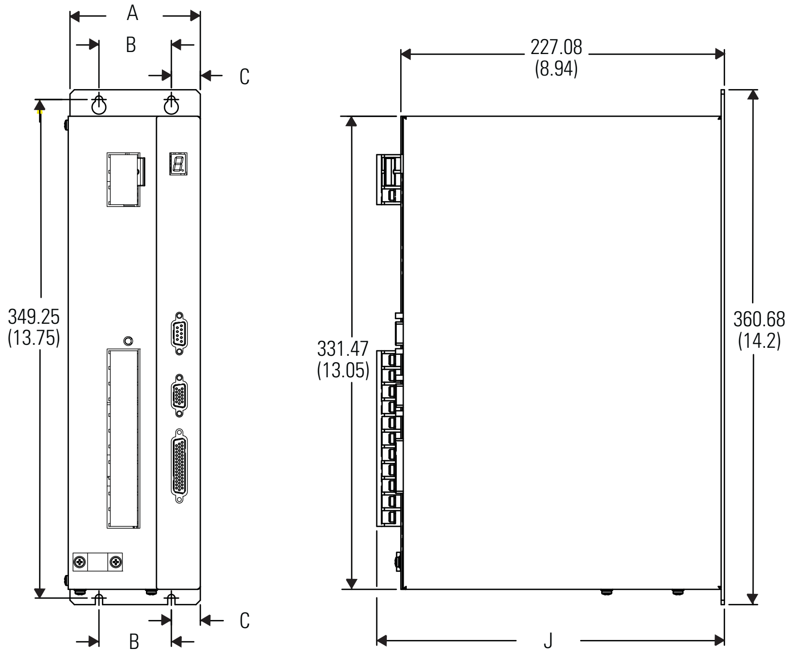
Dimensions are in mm (in.)

Unit shown is the 2098-DSD-005-SE

Ultra3000 Drives	**A** mm (in.)	**C** mm (in.)	**E** mm (in.)	**F** mm (in.)
2098-DSD-005, 2098-DSD-005X	65.02 (2.56)	13.26 (0.52)	32.77 (1.29)	72.64 (2.86)
2098-DSD-010, 2098-DSD-010X, 2098-DSD-020, 2098-DSD-020X				98.1 (3.89)
2098-DSD-005-SE, 2098-DSD-005-DN, 2098-DSD-005X-DN	87.88 (3.46)	24.64 (0.97)	43.94 (1.73)	95.5 (3.76)
2098-DSD-010-SE, 2098-DSD-010-DN, 2098-DSD-010X-DN, 2098-DSD-020-SE, 2098-DSD-020-DN, 2098-DSD-020X-DN				121.54 (4.79)

In the figure below, *-xxx* is replaced by -030, -075, or -150 to represent the Ultra3000 3, 7.5, and 15 kW drives respectively.

2098-DSD-*xxx*, 2098-DSD-*xxx*X, 2098-DSD-*xxx*-SE, 2098-DSD-*xxx*-DN, 2098-DSD-*xxx*X-DN Dimensions (230V)



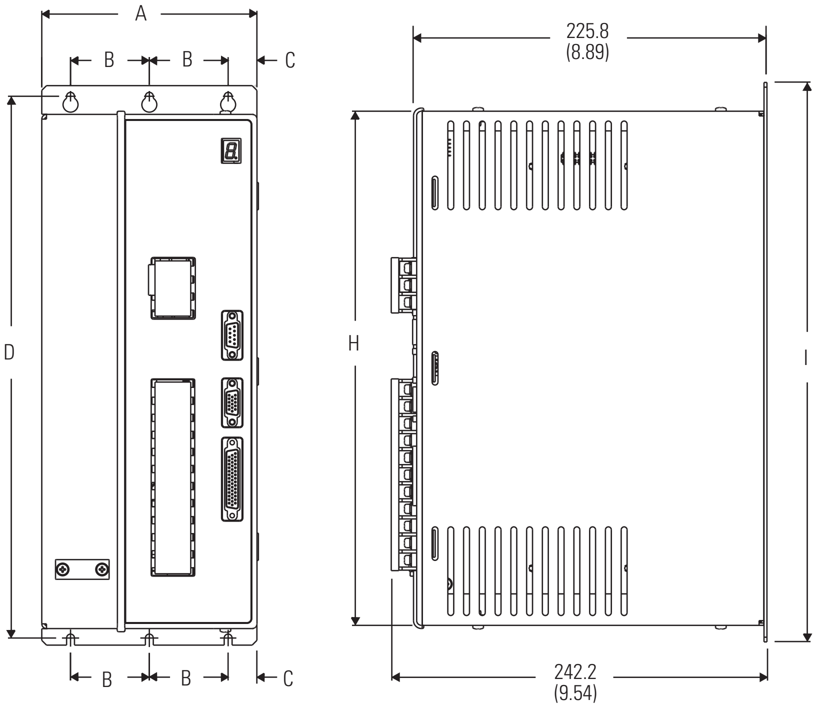
Dimensions are in mm (in.)

Unit shown is the 2098-DSD-030

Ultra3000 Drives	**A** mm (in.)	**B** mm (in.)	**C** mm (in.)	**J** mm (in.)
2098-DSD-030, 2098-DSD-030X, 2098-DSD-030-SE, 2098-DSD-030-DN, 2098-DSD-030X-DN	91.44 (3.6)	50.8 (2.0)	20.32 (0.8)	243.84 (9.6)
2098-DSD-075, 2098-DSD-075X, 2098-DSD-075-SE, 2098-DSD-075-DN, 2098-DSD-075X-DN	138.68 (5.41)	88.9 (3.5)	24.89 (0.96)	247.14 (9.73)
2098-DSD-150, 2098-DSD-150X, 2098-DSD-150-SE, 2098-DSD-150-DN, 2098-DSD-150X-DN	188.97 (7.44)	139.7 (5.5)	24.6 (0.97)	241.05 (9.49)

In the figure below, *xxx* is replaced by 030, 050, 100, 150, or 220 to represent the Ultra3000 3, 5, 10, 15, and 22 kW drives respectively.

2098-DSD-HV*xxx*, 2098-DSD-HV*xxx*X, 2098-DSD-HV*xxx*-SE, 2098-DSD-HV*xxx*-DN, 2098-DSD-HV*xxx*X-DN Dimensions (460V)



Dimensions are in mm (in.)

Unit shown is the 2098-DSD-HV030

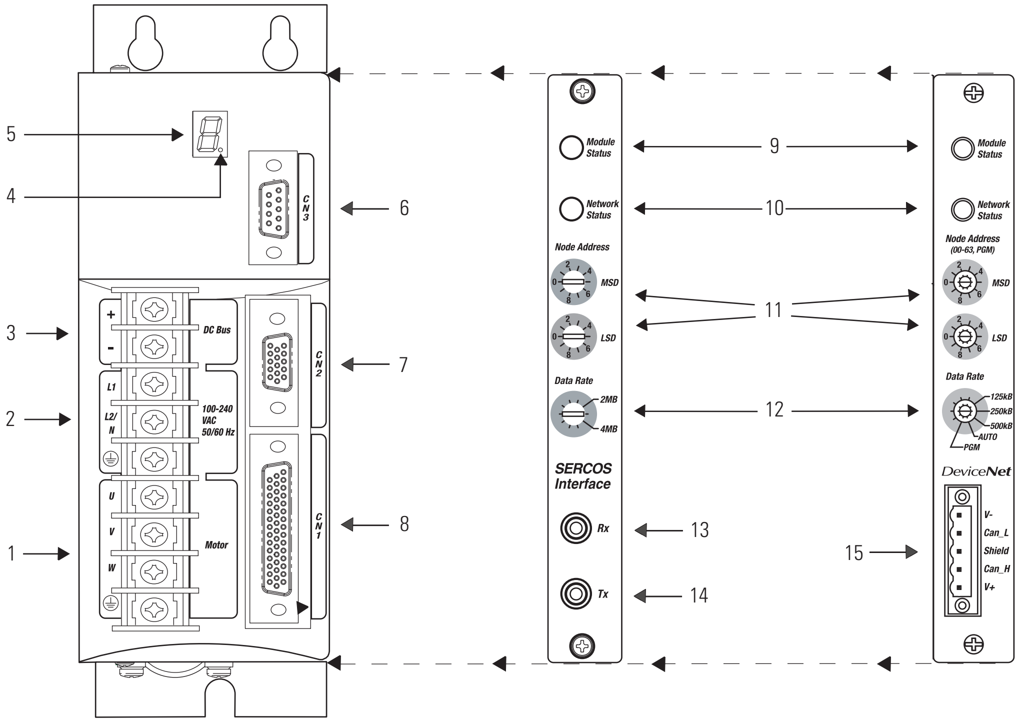
Ultra3000 Drives (1)	**A** mm (in.)	**C** mm (in.)	**B** mm (in.)	**D** mm (in.)	**H** mm (in.)	**I** mm (in.)
2098-DSD-HV030*x*, 2098-DSD-HV030-*xx*, 2098-DSD-HV050*x*, 2098-DSD-HV050-*xx*	138.7 5.46)	18.5 (0.73)	50.8 (2.0)	349.3 (13.75)	331.5 (13.05)	360.7 (14.2)
2098-DSD-HV100*x*, 2098-DSD-HV100-*xx*, 2098-DSD-HV150*x*, 2098-DSD-HV150-*xx*	151.6 (5.97)	25 (0.99)				
2098-DSD-HV220*x*, 2098-DSD-HV220-*xx*	203.2 (8.0)	25.4 (1.0)	76.2 (3.0)	380.4 (14.98)	362.6 14.26)	391.8 (15.43)

(1) The *x* represents the indexing (X) option. The -*xx* represents the SERCOS interface (SE) or DeviceNet interface (DN) option. SERCOS interface is not available with the DeviceNet interface option.

Ultra3000 Connector, Indicator, and Switch Locations

This section provides the connector, indicator, and switch locations for the Ultra3000 Digital Servo Drives (X indicates indexing, -SE indicates SERCOS interface, -DN indicates DeviceNet interface, and X-DN indicates indexing DeviceNet interface).

2098-DSD-005, 2098-DSD-010, 2098-DSD-020 Ultra3000 (230V) Connectors



2098-DSD-*xxx* and
2098-DSD-*xxx*X
Ultra3000 and Ultra3000
Indexing Drives

2098-DSD-*xxx*-SE
Ultra3000 SERCOS Interface Drives

2098-DSD-*xxx*-DN and
2098-DSD-*xxx*X-DN
Ultra3000 DeviceNet and
DeviceNet Indexing Drives

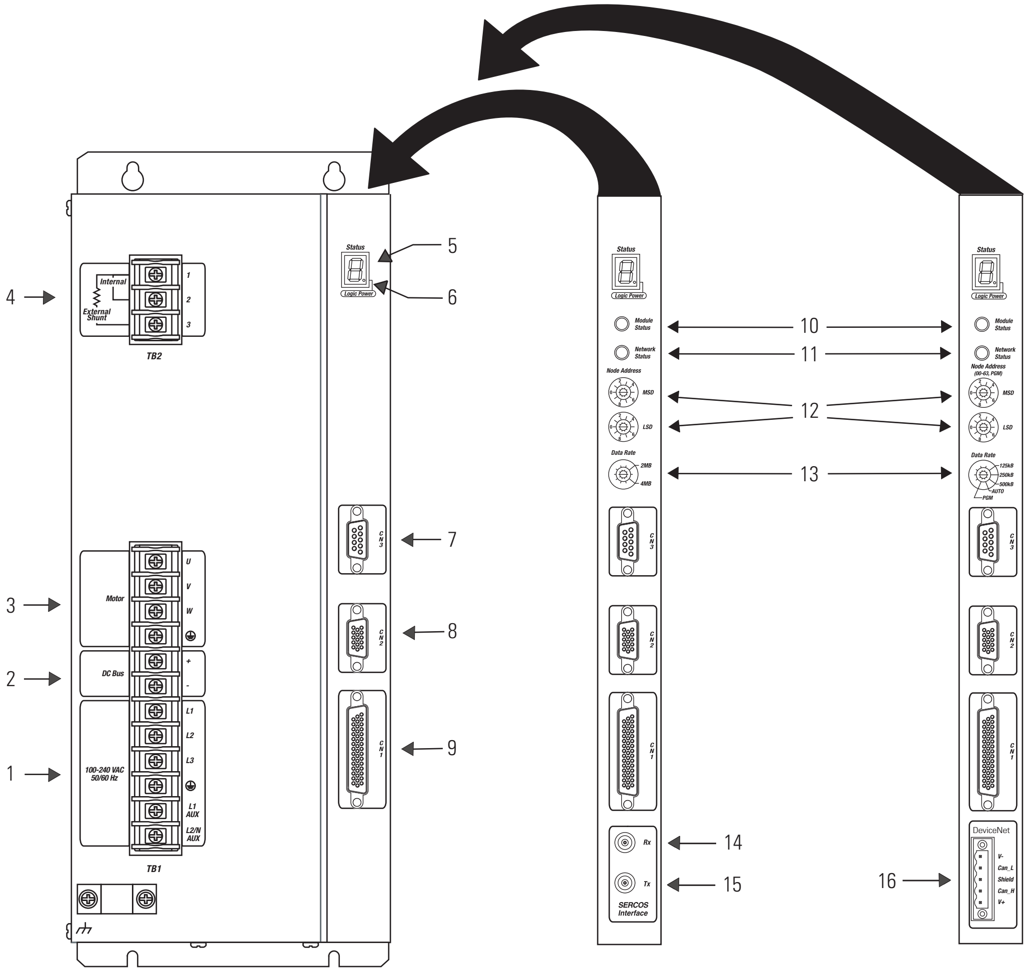
Item	Description
1	Motor power connections
2	AC input power connections
3	DC bus connections for active shunt resistor kit
4	Logic power
5	Seven-segment fault status indicator

Item	Description
6	CN3 9-pin serial port connector
7	CN2 15-pin motor feedback connector
8	CN1 44-pin user I/O connector
9	Module status indicator
10	Network status indicator

Item	Description
11	SERCOS node address switches
12	Data rate switch
13	SERCOS receive (Rx) connector
14	SERCOS transmit (Tx) connector
15	DeviceNet interface connector

For CN1, CN2, and CN3 connector options, refer to Breakout Components and Connector Kits beginning on page 440.

2098-DSD-030, 2098-DSD-075, and 2098-DSD-150 Ultra3000 (230V) Connectors



2098-DSD-*xxx* and
2098-DSD-*xxx*X
Ultra3000 and Ultra3000
Indexing Drives

2098-DSD-*xxx*-SE
Ultra3000 SERCOS Interface Drives

2098-DSD-*xxx*-DN and
2098-DSD-*xxx*X-DN
Ultra3000 DeviceNet and
DeviceNet Indexing Drives

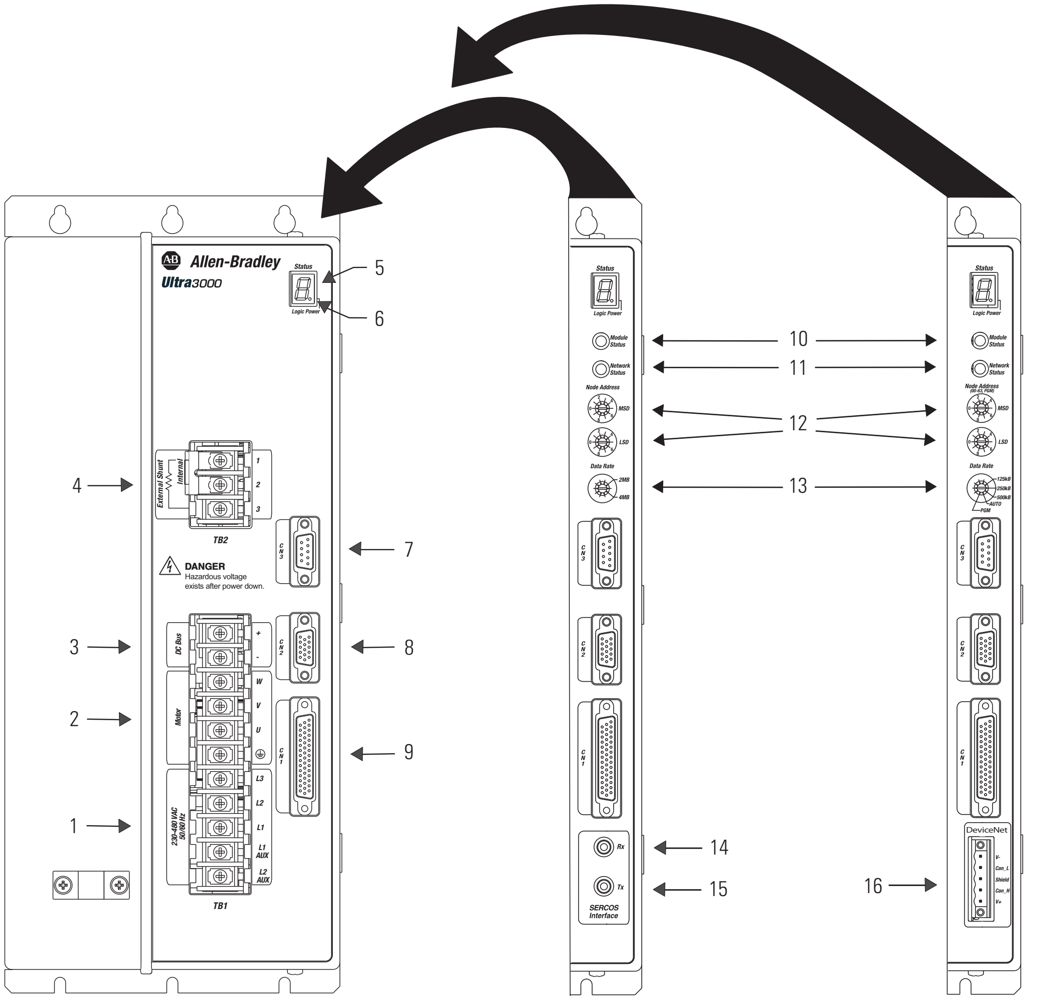
Item	Description
1	AC input power connections (1)
2	DC bus connections for active shunt resistor kit
3	Motor power connections
4	Passive shunt resistor connections
5	Seven-segment fault status indicator
6	Logic power
7	CN3 9-pin serial port connector
8	CN2 15-pin motor feedback connector

Item	Description
9	CN1 44-pin user I/O connector
10	Module status indicator
11	Network status indicator
12	SERCOS node address switches
13	Data rate switch
14	SERCOS receive (Rx) connector
15	SERCOS transmit (Tx) connector
16	DeviceNet interface connector

(1) The 2098-DSD-030*x-xx* drives do not have an L3 power terminal.

For CN1, CN2, and CN3 connector options, refer to Breakout Components and Connector Kits beginning on page 440.

2098-DSD-HV030, 2098-DSD-HV050, 2098-DSD-HV100, 2098-DSD-HV150, and 2098-DSD-HV220 Ultra3000 (460V) Connectors



2098-DSD-HV*xxx* and
2098-DSD-HV*xxx*X
Ultra3000 and Ultra3000
Indexing Drives

2098-DSD-HV*xxx*-SE
Ultra3000 SERCOS Interface Drives

2098-DSD-HV*xxx*-DN and
2098-DSD-HV*xxx*X-DN
Ultra3000 DeviceNet and
DeviceNet Indexing Drives

Item	Description
1	AC input power connections
2	DC bus connections for active shunt resistor kit
3	Motor power connections
4	Passive shunt resistor connections
5	Seven-segment fault status indicator
6	Logic power
7	CN3 9-pin serial port connector
8	CN2 15-pin motor feedback connector

Item	Description
9	CN1 44-pin user I/O connector
10	Module status indicator
11	Network status indicator
12	SERCOS node address switches
13	Data rate switch
14	SERCOS receive (Rx) connector
15	SERCOS transmit (Tx) connector
16	DeviceNet interface connector

For CN1, CN2, and CN3 connector options, refer to Breakout Components and Connector Kits beginning on page 440.

Ultra3000 Digital Servo Drive Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering chart below to understand the configuration of your drive. For questions regarding product availability, contact your Allen-Bradley distributor.

2098-DSD- ***xx xxx x-xx***

Connectivity Option
DN = DeviceNet interface
SE = SERCOS interface
Blank = No network connectivity

Indexing Capability
X = Indexing
Blank = No indexing

Continuous Output Power
005 = 500 W030 = 3 kW100 = 10 kW
010 = 1 kW050 = 5 kW150 = 15 kW
020 = 2 kW075 = 7.5 kW220 = 22 kW

Input Power (VAC)
HV = High Voltage (230…480V AC)
Blank = Standard (115…230V AC)

Digital Servo Drive

Bulletin Number

Ultraware Software Catalog Number

Ultra3000, Ultra3000 with indexing, Ultra3000-DN, and Ultra3000-DN with indexing drives are configured by using Ultraware software (catalog number 2098-UWCPRG). The Ultra3000-SE drives are configured by using RSLogix 5000 software.

Notes:

Ultra5000 Intelligent Positioning Drives

The Ultra5000 and Ultra5000-DN Intelligent Positioning Drives make up a high-performance, fully-programmable positioning servo drive family. The Ultra5000 servo drive family incorporates a state-of-the-art DSP-based control architecture and a rugged Integrated Power Module (IPM)-based power section. Ultra5000 drives are programmed by using motion commands in an ANSI C format. The Ultra5000 design is optimized for high-speed position capturing and calculations. Position capture latency is less than 1.0 µs, and processing time for position-based calculations can be as low as 125 µs.

Ultra5000 Servo Drive Communication Interface

Drive Option	Drive Cat. No.	Command Interface
Positioning drive	2098-IPD-*xxx* and 2098-IPD-HV*xxx*	Control interface
Positioning drive with DeviceNet interface	2098-IPD-*xxx*-DN and 2098-IPD-HV*xxx*-DN	Control interface with factory installed DeviceNet communication

Ultra5000 Intelligent Positioning Drive Components

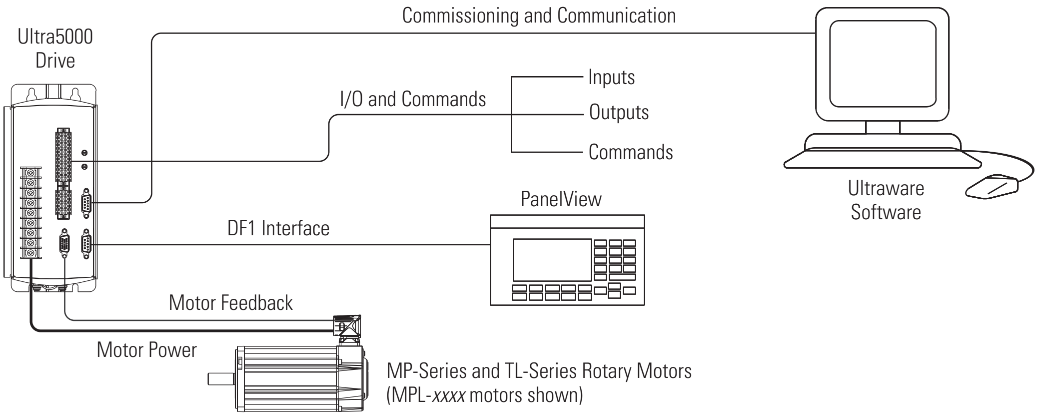
Ultra5000 intelligent positioning drive systems consist of these components:

- One Ultra5000 intelligent positioning drive
- One MP-Series or TL-Series rotary motors
- One motor power and feedback cable

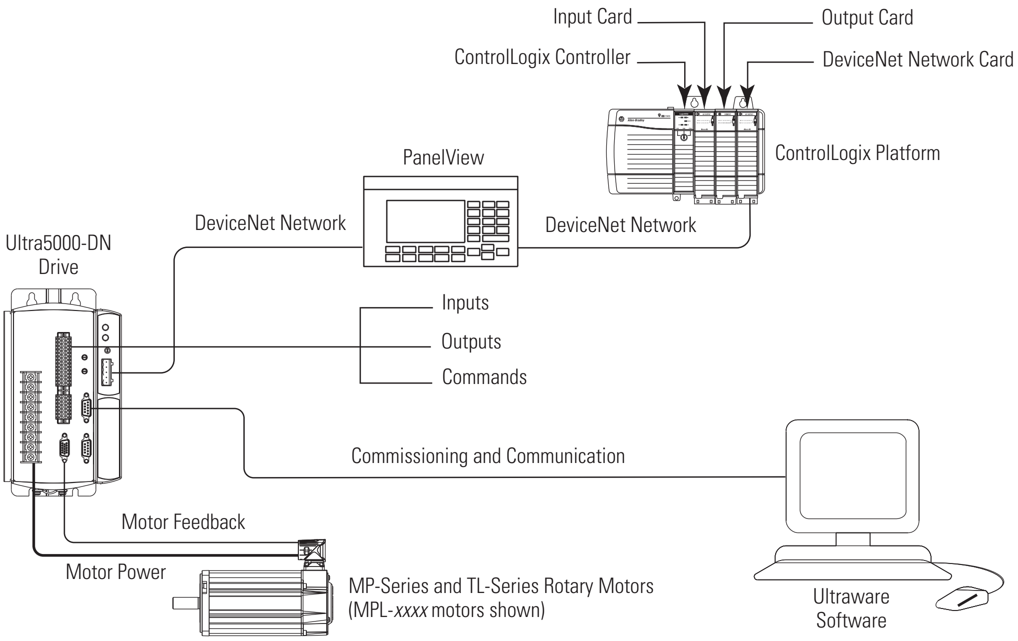
Ultra5000 systems may also include a Bulletin 2090 external active or passive shunt module.

To learn about the C programming environment or library of motion commands used to program an Ultra5000 intelligent positioning drive, refer to the Ultra5000 Programming Manual, publication 2098-PM001.

Typical Configuration - Ultra5000 Intelligent Positioning Drive System



Typical Configuration - Ultra5000-DN Intelligent Positioning Drive System



Ultra5000 Intelligent Positioning Drive Specifications

This section contains general power, physical and environmental, power dissipation, controller, inputs/outputs, auxiliary feedback, communication, and connector specifications for the Ultra5000 and Ultra5000-DN Intelligent Positioning Drives.

Power Specifications

2098-IPD-005-*xx*, 2098-IPD-010-*xx*, and 2098-IPD-020-*xx* Ultra5000 (230V) Drives

Attribute	2098-IPD-005	2098-IPD-010	2098-IPD-020
AC input voltage (1)	100…240V rms single-phase		
AC input frequency	47…63 Hz		
AC input current (2) (3) Nom (rms) 230V AC (0-pk) max inrush (4)	 5 A 100 A - Series A 20 A - Series B	 9 A 100 A - Series A 20 A - Series B	 18 A 100 A - Series A 20 A - Series B
Continuous output current (rms)	1.8 A	3.5 A	7.1 A
Continuous output current (0-pk)	2.5 A	5.0 A	10 A
Peak output current (rms)	5.3 A	10.6 A	21.2 A
Peak output current (0-pk)	7.5 A	15 A	30 A
Bus capacitance	1410 µF	1880 µF	1880 µF
Internal shunt resistance	N/A	N/A	N/A
Shunt on	N/A	N/A	N/A
Shunt off	N/A	N/A	N/A
Bus overvoltage	400V DC	400V DC	400V DC
Energy absorption capability 115V AC input 230V AC input	 93 J 38 J	 125 J 51 J	
Continuous power output 115V AC input 230V AC input	 0.25 kW 0.5 kW	 0.5 kW 1.0 kW	 1.0 kW 2.0 kW

(1) Specification is for nominal voltage. The absolute limits are ±10%, or 88…265V rms.

(2) The 2098-IPD-005*x-xx,* 2098-IPD-010*x-xx,* and 2098-IPD-020*x-xx* (230V) drives are limited to:
Series A - one contactor cycle every two minutes.
Series B - one contactor cycle every 10 s for up to two minutes, not to exceed 12 cycles in five minutes.

(3) Power initialization requires a short period of inrush current. Dual element time delay (slow blow) fuses are recommended (refer to Fuse Specifications on page 384).

(4) In-rush current limiting circuitry is enabled within 3 s after removal of AC line power.

2098-IPD-030-*xx*, 2098-IPD-075-*xx*, and 2098-IPD-150-*xx* Ultra5000 (230V) Drives

Attribute	2098-IPD-030	2098-IPD-075	2098-IPD-150
AC input voltage (1)	100…240V rms Single-phase	240V rms Three-phase	
AC input frequency	47…63 Hz		
Main AC input current (2) (3)			
Nom (rms)	28 A	30 A	46 A
230V AC (0-pk) max inrush	50 A	50 A	68 A
Auxiliary AC input current			
115V AC (rms) nom	1.0 A	1.0 A	1.0 A
230V AC (rms) nom	0.5 A	0.5 A	0.5 A
115V AC (0-pk) max inrush (4)	47 A	47 A	47 A
230V AC (0-pk) max inrush (4)	95 A	95 A	95 A
Continuous output current (rms)	10.6 A	24.7 A	45.9 A
Continuous output current (0-pk)	15 A	35 A	65 A
Peak output current (rms)	21.2 A	53 A	106 A
Peak output current (0-pk)	30 A	75 A	150 A
Bus capacitance	2820 μF	4290 μF	7520 μF
Internal shunt resistance	35 Ω	16.5 Ω	9.1 Ω
Shunt on	420V DC	420V DC	420V DC
Shunt off	402V DC	402V DC	402V DC
Bus overvoltage	452V DC	452V DC	452V DC
Internal shunt			
Continuous power	50 W	50 W	180 W
Peak power	4.5 kW	10 kW	18 kW
External shunt			
Resistance	30 Ω(-0/+5%)	16.5 Ω(-0/+5%)	9 Ω(-0/+5%)
Continuous power	2.4 kW	4 kW	8 kW
Peak power	6 kW	10 kW	19 kW
Energy absorption capability			
115V AC input	251 J	381 J	669 J
230V AC input	139 J	211 J	370 J
Continuous power output			
115V AC input	1.5 kW	N/A	N/A
230V AC input	3 kW	7.5 kW	15 kW

(1) Specification is for nominal voltage. The absolute limits are ±10%, or 88…265V rms.

(2) The 2098-IPD-030*x-xx*, 2098-IPD-075*x-xx*, and 2098-IPD-150*x-xx* (230V) drives are limited to one contactor cycles per two minutes.

(3) Power initialization requires a short period of inrush current. Dual element time delay (slow blow) fuses are recommended (refer to Fuse Specifications on page 384).

(4) 400 μs half wave sine.

Ultra5000 (460V) Drives

Attribute	2098-IPD-HV030	2098-IPD-HV050	2098-IPD-HV100	2098-IPD-HV150	2098-IPD-HV220
AC input voltage (1) (2)	230/480V rms Three-phase				
AC input frequency	47…63 Hz				
Main AC input current (3) (4) 460V AC (rms) nom 460V AC (rms) max inrush	 4 A 6 A	 7 A 6 A	 14 A 6 A	 20 A 6 A	 28 A 6 A
Auxiliary AC input current 230V AC (rms) nom 360V AC (rms) nom 480V AC (rms) nom 230V AC (0-pk) max inrush (5) 480V AC (0-pk) max inrush (5)	 0.55 A 0.35 A 0.25 A 47 A 68 A	 0.55 A 0.35 A 0.25 A 47 A 68 A	 0.55 A 0.35 A 0.25 A 47 A 68 A	 0.55 A 0.35 A 0.25 A 47 A 68 A	 0.55 A 0.35 A 0.25 A 47 A 68 A
Continuous output current (rms)	5.0 A	7.8 A	16.3 A	24.0 A	33.2 A
Continuous output current (0-pk)	7.0 A	11 A	23 A	34 A	47 A
Peak output current (rms)	9.9 A	15.6 A	32.5 A	48.1 A	66.5 A
Peak output current (0-pk)	14 A	22 A	46 A	68 A	94 A
Bus capacitance	470 µF	470 µF	705 µF	940 µF	1880 µF
Internal shunt resistance	120 Ω	120 Ω	40 Ω	25 Ω	20 Ω
Shunt on 230V AC input 480V AC input	 400V DC 800V DC				
Shunt off 230V AC input 480V AC input	 375V DC 750V DC				
Bus overvoltage 230V AC input 480V AC input	 410V DC 810V DC				
Internal shunt Continuous power Peak power	 100 W 5.3 kW	 100 W 5.3 kW	 200 W 16 kW	 200 W 25.6 kW	 400 W 32 kW
External shunt Resistance (-0/+5%) Continuous power Peak power	 120 Ω 3 kW 5.3 kW	 120 Ω 5 kW 5.3 kW	 40 Ω 10 kW 16 kW	 25 Ω 15 kW 25.6 kW	 20 Ω 22 kW 32 kW
Energy absorption capability 230V AC input with 230V motor 230V AC input with 460V motor 480V AC input	 15 J 129 J 55 J	 15 J 129 J 55 J	 22 J 194 J 82 J	 29 J 259 J 109 J	 59 J 517 J 219 J
Continuous power output 230V AC input 480V AC input	 1.5 kW 3.0 kW	 2.5 kW 5.0 kW	 5.0 kW 10 kW	 7.5 kW 15 kW	 11 kW 22 kW

(1) Specification is for nominal voltage. The absolute limits are ±10%, or 207…264V rms and 324…528V rms.

(2) The 2098-IPD-HV*xxx-xx* drives can be powered with 230V rms and used with motors designed for 230V operation. In such cases, the voltage levels used for shunting and DC bus overvoltage limits are adjusted to be compatible with the voltage limit of the motor.
The 2098-IPD-HV*xxx-xx* drives can be powered with 480V rms and used with motors designed for 480V operation. In such cases, the voltage levels used for shunting and DC bus overvoltage limits are adjusted to be compatible with the voltage limit of the motor.

(3) The 2098-IPD-HV*xxx -xx* (460V) drives are limited to three contactor cycles per minute.

(4) Power initialization requires a short period of inrush current (processor controlled via soft start circuitry). Dual element time delay (slow blow) fuses are recommended (refer to Fuse Specifications on page 384).

(5) 400 µs half wave sine.

Fuse Specifications

Use class CC, G, J, L, R, or T class fuses, with current ratings as indicated in the table below. The table below lists fuse examples recommended for use with the Ultra3000 (230V and 460V) drives.

Drive Cat. No.	Input Voltage	Input Type	Recommended Fuse	
			Class CC (1)	Class J (1)
2098-IPD-005	230V AC	Input power	FNQ-R-6	LPJ-6SP
2098-IPD-010			FNQ-R-10	LPJ-10SP
2098-IPD-020			FNQ-R-20	LPJ-20SP
2098-IPD-030			FNQ-R-30	LPJ-30SP
2098-IPD-075			FNQ-R-30	LPJ-30SP
2098-IPD-150			N/A	LPJ-60SP
2098-IPD-*xxx*		Auxiliary input power	FNQ-R-10	LPJ-10SP
2098-IPD-HV030	460V AC	Input power	KTK-R-5	LPJ-5SP
2098-IPD-HV050			KTK-R-8	LPJ-8SP
2098-IPD-HV100			KTK-R-20	LPJ-17-1/2SP
2098-IPD-HV150			KTK-R-30	LPJ-30SP
2098-IPD-HV220			N/A	LPJ-35SP
2098-IPD-HV*xxx*		Auxiliary input power	FNQ-R-10	LPJ-10SP

(1) Part numbers shown are examples of Bussmann fuses.

Circuit Breaker Specifications

While circuit breakers offer some convenience, there are limitations for their use. Circuit breakers do not handle high current inrush as well as fuses.

Make sure the selected components are properly coordinated and meet acceptable codes including any requirements for branch circuit protection. Evaluation of the short-circuit available current is critical and must be kept below the short-circuit current rating of the circuit breaker.

Circuit Breaker Examples for Ultra3000 (460V) Drives

Drive Cat. No.	Input Voltage	Circuit Breakers
2098-IPD-HV030	460V	140M-F8E-C16
2098-IPD-HV050		140M-F8E-C20
2098-IPD-HV100		140M-F8E-C32
2098-IPD-HV150		140M-F8E-C45
2098-IPD-HV220		N/A

Contactor Ratings

Drive Cat. No.	Input Voltage	Contactor
2098-IPD-HV030	460V	100-C23*x*10 (AC coil) 100-C23Z*x*10 (DC coil)
2098-IPD-HV050		100-C30*x*10 (AC coil) 100-C30Z*x*10 (DC coil)
2098-IPD-HV100		100-C37*x*10 (AC coil) 100-C37Z*x*10 (DC coil)
2098-IPD-HV150		100-C43*x*10 (AC coil) 100-C43Z*x*10 (DC coil)
2098-IPD-HV220		100-C60*x*10 (AC coil) 100-C60Z*x*10 (DC coil)

Power Dissipation Specifications

Drive Cat. No.	Max Loss W
2098-IPD-005	48
2098-IPD-010	48
2098-IPD-020	50
2098-IPD-030	150 + dissipative shunt
2098-IPD-075	300 + dissipative shunt
2098-IPD-150	500 + dissipative shunt

Drive Cat. No.	Max Loss W
2098-IPD-HV030	175 + dissipative shunt
2098-IPD-HV050	175 + dissipative shunt
2098-IPD-HV100	350 + dissipative shunt
2098-IPD-HV150	350 + dissipative shunt
2098-IPD-HV220	600 + dissipative shunt

Communication Specifications

Attribute	Value
Serial	
Ports	Two RS-232/RS-422/RS-485
Communication rates	1200, 2400, 4800, 9600, 19200, and 38400 bps
DeviceNet interface (option)	
Power consumption from network	60 mA
Data rates	125, 250, and 500 kbps
Node addresses	00…63
Messaging capabilities	Explicit, polled I/O

Inputs/Outputs Specifications

Attribute	Value
General purpose digital inputs	16 optically isolated 12…24V inputs
Inputs/outputs - sinking/sourcing selection	Software selectable as a group to be active high, current sinking or active low, current sourcing
General purpose digital outputs	7 optically isolated 12…24V Outputs - 50 mA max
General purpose relay output	1 normally open relay - 30V DC max, 1 A max current
General purpose I/O response	100 μs
High speed input response	< 1 μs (inputs 1 and 2)
Position capture response	< 1 μs (input 1, input 2, motor encoder index, and auxiliary encoder index)
General purpose analog inputs	2 12-Bit analog to digital converters (+/- 10V, single-ended)
General purpose analog outputs	2 12-Bit digital to analog converters (+/- 10V, +/- 2mA, single-ended)

Memory Specifications

Attribute	Value
User program memory capacity	512 Kbps
User program memory storage medium	Flash memory, 100,000 write cycles
Non-volatile memory capacity	32 Kbps (approximately 8000 nonvolatile user variables)
Non-volatile memory storage medium	nvSRAM (high-speed SRAM/EEPROM)

Auxiliary Feedback Specifications

Attribute	Value
Input modes	A quad B
Max signal frequency	2.5 MHz
Input types	Differential

Connector Specifications

Connector	Description	Specification
CN1A	Digital I/O connector	28-position plugable spring clamp terminal block
CN1B	Auxiliary feedback/analog I/O connector	14-position plugable spring clamp terminal block
CN2	Motor feedback connector	15-pin high-density female D-sub connector
CN3A and CN3B	Serial port connectors	9-pin female D-sub connector
TB1 and TB2	Main and auxiliary AC, DC bus, motor power, and shunt connectors	Screw terminal block

Physical and Environmental Specifications

Attribute	Value		Attribute	Value	
Weight, approx.	kg	(lb)	Weight, approx.	kg	(lb)
2098-IPD-005	1.77	(3.9)	2098-IPD-HV030	8.55	(18.8)
2098-IPD-010	2.07	(4.55)	2098-IPD-HV050	8.55	(18.8)
2098-IPD-020	2.05	(4.51)	2098-IPD-HV100	10.44	(22.96)
2098-IPD-030	6.16	(13.58)	2098-IPD-HV150	10.44	(22.96)
2098-IPD-075	9.23	(20.35)	2098-IPD-HV220	14.1	(31.0)
2098-IPD-150	13.96	(30.78)	2098-IPD-HV030-DN	8.89	(19.6)
2098-IPD-005-DN	2.11	(4.7)	2098-IPD-HV050-DN	8.89	(19.6)
2098-IPD-010-DN	2.41	(5.3)	2098-IPD-HV100-DN	10.78	(23.72)
2098-IPD-020-DN	2.39	(5.3)	2098-IPD-HV150-DN	10.78	(23.72)
2098-IPD-030-DN	6.55	(14.43)	2098-IPD-HV220-DN	14.44	(31.77)
2098-IPD-075-DN	9.62	(21.20)			
2098-IPD-150-DN	14.35	(31.63)			
Temperature, ambient					
Operating	0…55 °C (32…131 °F)				
Storage	-40…70 °C (-40…158 °F)				
Humidity	5…90% noncondensing				
Altitude	1500 m (5000 ft) Derate 3% for each 300 m above 1500 m				
Vibration Operating/Nonoperating	10…2000 Hz, 2 g peak, 0.015 in. (max displacement)				
Shock Nonoperating	15 g 11 ms half sine				
UL Listed to U.S. and Canadian safety standards	UL 508 C File E145959				

Maximum Feedback Cable Lengths

Although motor power and feedback cables are available in standard lengths up to 90 m (295.3 ft), the drive/motor/feedback combination may limit the maximum feedback cable length. This table assumes the use of cables recommended in the 2090-Series Motor/Actuator Cable Selection table on page 401.

Cable Lengths for Compatible Rotary Motors

Motor Cat. No.	Absolute High-resolution (5V) Encoder m (ft)	Absolute High-resolution (9V) Encoder m (ft)	Incremental/TTL (5V) Encoder m (ft)
MPL-A3*xxx*…MPL-A5*xxx*-S/M (1)	30 (98.4)		
MPL-B3*xxx*…MPL-B9*xxx*-S/M		30 (98.4)	
MPL-A/B3*xxx*…MPL-A/B45*xxx*-H			30 (98.4)
MPM-A*xxxxx*-S/M	30 (98.4)		
MPM-B*xxxxx*-S/M		30 (98.4)	
MPF-A*xxxx*-S/M (1)	30 (98.4)		
MPF-B*xxxx*-S/M		30 (98.4)	
MPS-A*xxxx*-S/M	30 (98.4)		
MPS-B*xxxx*-S/M		30 (98.4)	
TLY-A*xxxx*-H			30 (98.4)

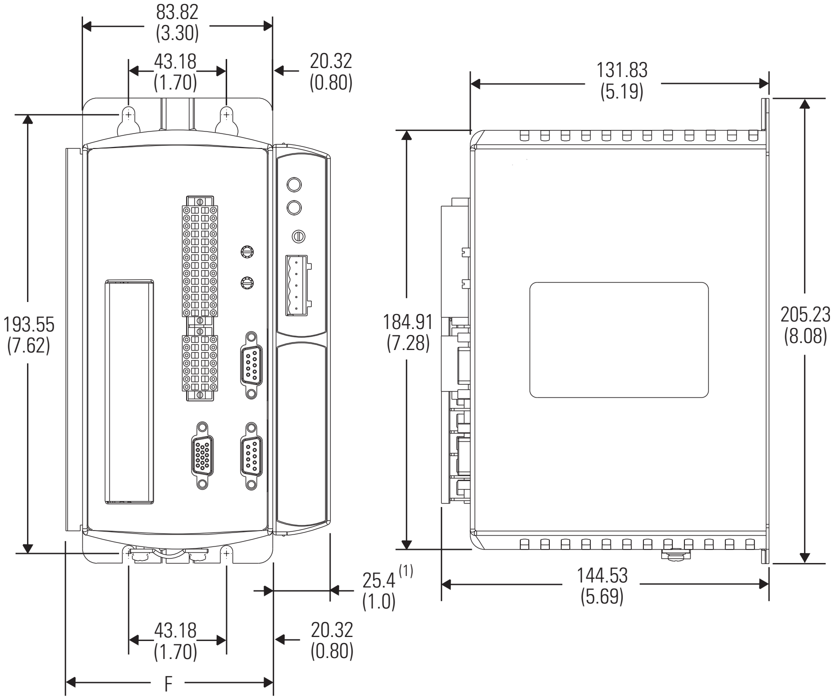
(1) MPL-A5*xxx* and MPF-A5*xxx* motor encoders are rated for 9V, the remaining Bulletin MPL and MPF (230V) motor encoders are rated for 5V.

Ultra5000 Intelligent Positioning Drive Dimensions

This section contains dimensions for the Ultra5000 and Ultra5000 DeviceNet interface drives (-DN indicates DeviceNet interface).

In the figure below, *-xxx* is replaced by -005, -010, or -020 to represent the Ultra5000 500 W, 1 kW, and 2 kW drives respectively.

2098-IPD-*xxx* and 2098-IPD-*xxx*-DN Dimensions (230V)



Dimensions are in mm (in.)

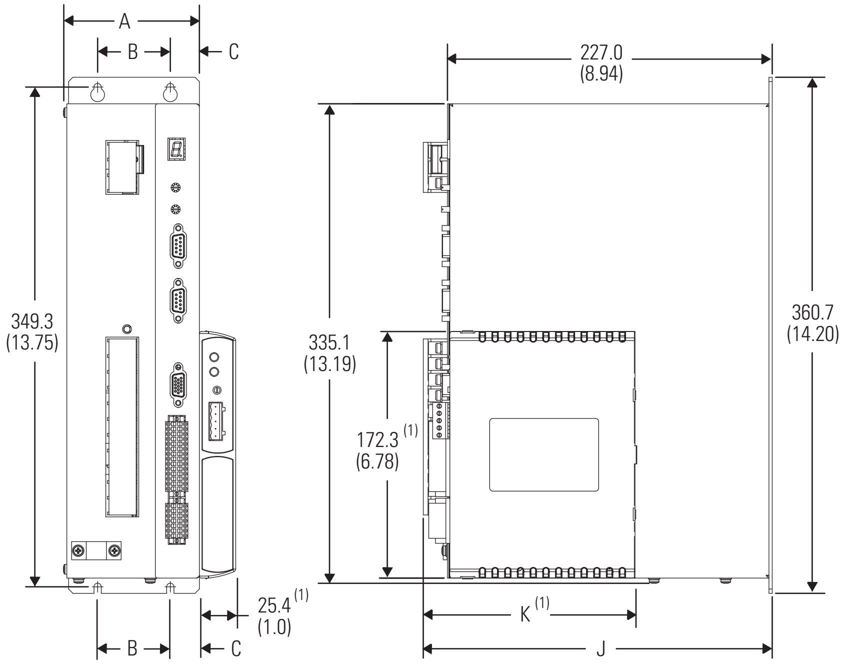
Model shown is the 2098-IPD-*xxx*-DN

(1) This dimension applies to only the 2098-IPD-005-DN, 2098-IPD-010-DN, and 2098-IPD-020-DN (Ultra5000 with DeviceNet interface) models.

Ultra5000 Drive	F mm (in.)
2098-IPD-005	91.19 (3.59)
2098-IPD-010-DN, 2098-IPD-020-DN	114.30 (4.50)

In the figure below, *-xxx* is replaced by -030, -075, or -150 to represent the Ultra5000 3, 7.5, and 15 kW drives respectively.

2098-IPD-*xxx* and 2098-IPD-*xxx*-DN Dimensions (230V)



Dimensions are in mm (in.)

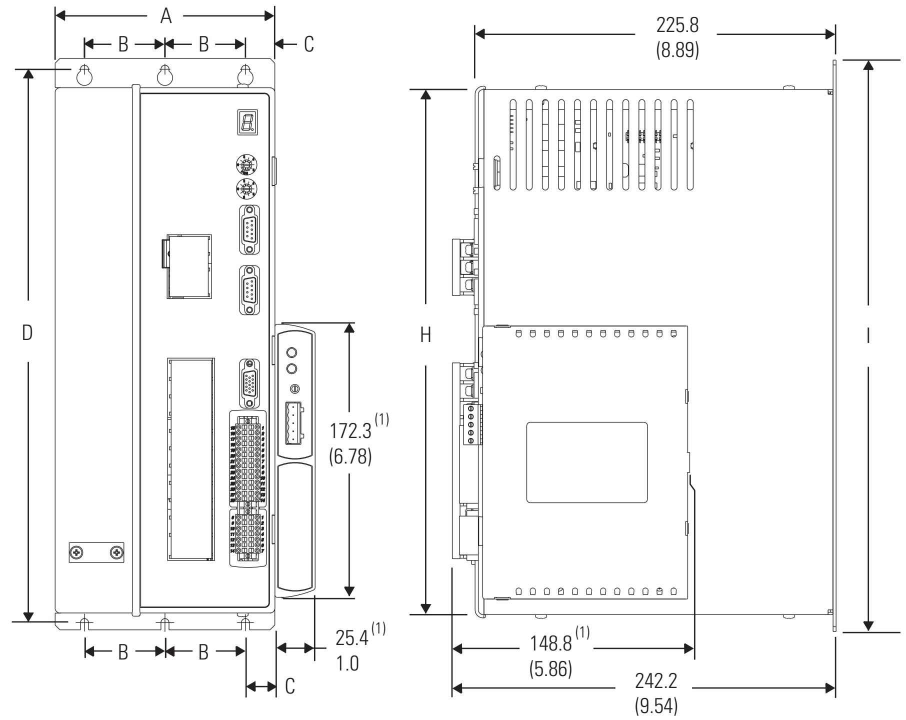
Model shown is the 2098-IPD-030-DN

(1) This dimension only applies to the 2098-IPD-030-DN, 2098-IPD-075-DN, and 2098-IPD-150-DN (Ultra5000 with DeviceNet interface) models.

Ultra5000 Drive	**A** mm (in.)	**B** mm (in.)	**C** mm (in.)	**J** mm (in.)	**K** mm (in.)
2098-IPD-030 2098-IPD-030-DN	92.7 (3.65)	50.8 (2.0)	20.3 (0.8)	243.9 (9.60)	149.3 (5.88)
2098-IPD-075 2098-IPD-075-DN	141.0 (5.55)	88.9 (3.50)	24.3 (0.96)	247.3 (9.73)	152.7 (6.01)
2098-IPD-150 2098-IPD-150-DN	190.5 (7.5)	69.9 (2.75)	24.9 (0.98)	245.1 (9.49)	146.5 (5.77)

In the figure below, -*xxx* is replaced by -030, -050, -100, -150, or -220 to represent the Ultra5000 3, 5, 10, 15, and 22 kW drives respectively.

2098-IPD-HV*xxx* and 2098-IPD-HV*xxx*-DN Dimensions (460V)



Dimensions are in mm (in.)

Model shown is the 2098-IPD-HV030-DN

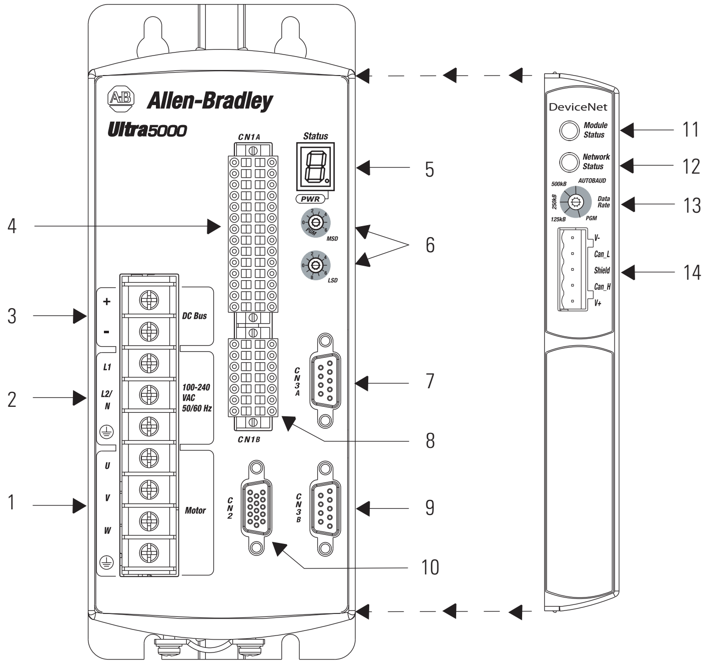
(1) This dimension applies to only the 2098-IPD-HV030-DN, 2098-IPD-HV050-DN, 2098-IPD-HV100-DN, 2098-IPD-HV150-DN, and 2098-IPD-HV220-DN (Ultra5000 with DeviceNet interface) models.

Ultra5000 Drive	**A** mm (in.)	**B** mm (in.)	**C** mm (in.)	**D** mm (in.)	**H** mm (in.)	**I** mm (in.)
2098-IPD-HV030, 2098-IPD-HV030-DN, 2098-IPD-HV050, 2098-IPD-HV050-DN	138.7 (5.46)	50.8 (2.0)	18.5 (0.73)	349.3 (13.75)	331.5 (13.05)	360.7 (14.20)
2098-IPD-HV100, 2098-IPD-HV100-DN, 2098-IPD-HV150, 2098-IPD-HV150-DN	151.6 (5.97)		25.0 (0.99)			
2098-IPD-HV220 2098-IPD-HV220-DN	203.2 (8.0)	76.2 (3.0)	25.4 (1.0)	380.41 (14.98)	362.6 (14.28)	391.8 (15.43)

Ultra5000 Connector, Indicator, and Switch Locations

This section provides the connector, indicator, and switch locations for the Ultra5000 and Ultra5000 DeviceNet interface drives (-DN indicates DeviceNet interface).

2098-IPD-005, 2098-IPD-010, and 2098-IPD-020 Ultra5000 (230V) Connectors



2098-IPD-005, 2098-IPD-010,
and 2098-IPD-020
Ultra5000 Drives

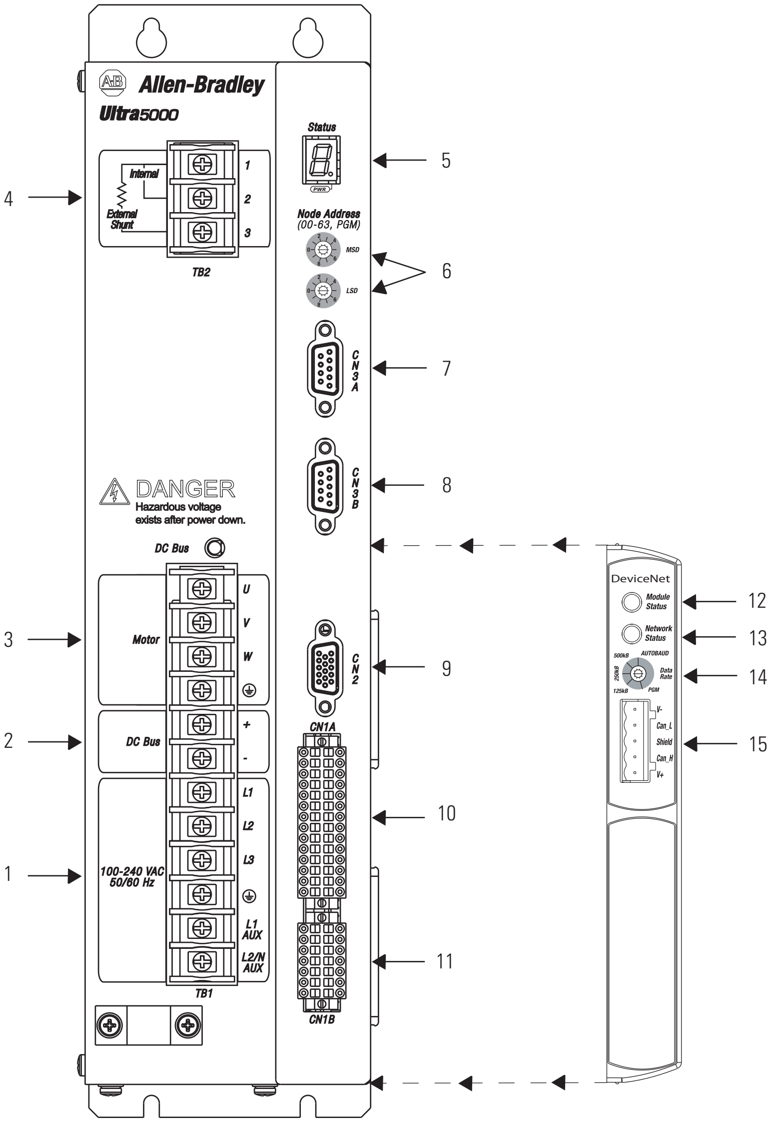
2098-IPD-005-DN, 2098-IPD-010-DN,
and 2098-IPD-020-DN
Ultra5000 DeviceNet Interface Drives

Item	Description
1	Motor power connections
2	AC input power connections
3	DC bus connections for active shunt resistor kit
4	CN1A 28-pin digital I/O connector
5	Seven-segment fault status indicator
6	Node address switches
7	CN3A 9-pin serial port connector

Item	Description
8	CN1B 14-pin auxiliary feedback and analog I/O connector
9	CN3B 9-pin auxiliary serial port connector
10	CN2 15-pin motor feedback connector
11	Module status indicator
12	Network status indicator
13	Data rate switch
14	DeviceNet interface connector

For CN1A, CN1B, CN2, and CN3 connector options, refer to Breakout Components and Connector Kits beginning on page 440.

2098-IPD-030, 2098-IPD-075, and 2098-IPD-150 Ultra5000 (230V) Connectors



2098-IPD-030, 2098-IPD-075,
and 2098-IPD-150
Ultra5000 Drives

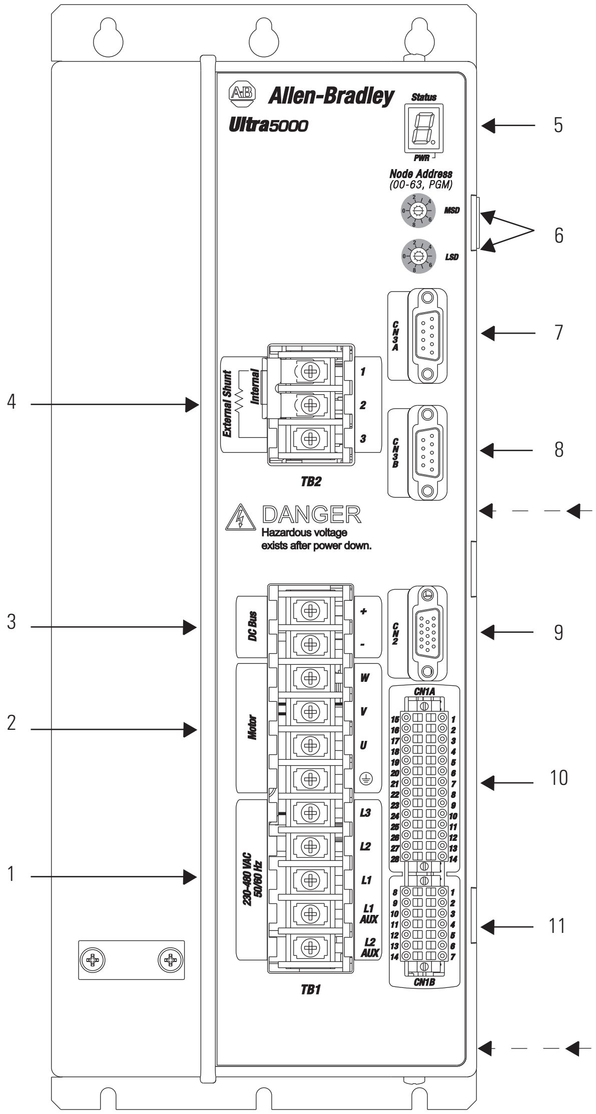
2098-IPD-030-DN, 2098-IPD-075-DN,
and 2098-IPD-150-DN
Ultra5000 DeviceNet Interface Drives

Item	Description	Item	Description	Item	Description
1	AC input power connections [1]	6	Node address switches	11	CN1B 14-pin auxiliary feedback and analog I/O connector
2	DC bus connections for active shunt resistor kit	7	CN3A 9-pin serial port connector	12	Module status indicator
3	Motor power connections	8	CN3B 9-pin auxiliary serial port connector	13	Network status indicator
4	Passive shunt connections	9	CN2 15-pin motor feedback connector	14	Data rate switch
5	Seven-segment fault status indicator	10	CN1A 28-pin digital I/O connector	15	DeviceNet interface connector

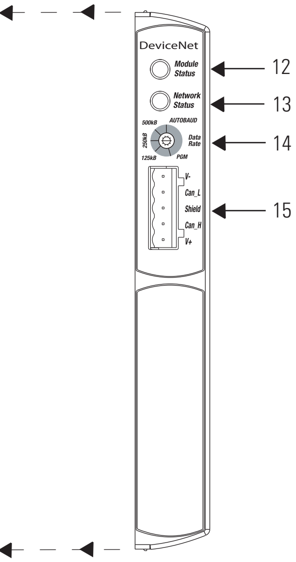
[1] The 2098-IPD-030*x-xx* drives do not have an L3 power terminal.

For CN1A, CN1B, CN2, and CN3 connector options, refer to Breakout Components and Connector Kits beginning on page 440.

2098-IPD-HV030, 2098-IPD-HV050, 2098-IPD-HV100, 2098-IPD-HV150, and 2098-IPD-HV220 Ultra5000 (460V) Connectors



2098-IPD-HV030, 2098-IPD-HV050, 2098-IPD-HV100,
2098-IPD-HV150, and 2098-IPD-HV220
Ultra5000 Drives



2098-IPD-HV030-DN, 2098-IPD-HV050-DN,
2098-IPD-HV100-DN, 2098-IPD-HV150-DN,
and 2098-IPD-HV220-DN
Ultra5000 DeviceNet Interface Drives

Item	Description	Item	Description	Item	Description
1	AC input power connections	6	Node address switches	11	CN1B 14-pin auxiliary feedback and analog I/O connector
2	DC bus connections for active shunt resistor kit	7	CN3A 9-pin serial port connector	12	Module status indicator
3	Motor power connections	8	CN3B 9-pin auxiliary serial port connector	13	Network status indicator
4	Passive shunt connections	9	CN2 15-pin motor feedback connector	14	Data rate switch
5	Seven-segment fault status indicator	10	CN1A 28-pin digital I/O connector	15	DeviceNet interface connector

For CN1A, CN1B, CN2, and CN3 connector options, refer to Breakout Components and Connector Kits beginning on page 440.

Ultra5000 Intelligent Positioning Drives Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering chart below to understand the configuration of your drive. For questions regarding product availability, contact your Allen-Bradley distributor.

2098 - IPD - *xx xxx* - *xx*

Connectivity Option
DN = DeviceNet interface
Blank = No network connectivity

Continuous Output Power
005 = 500 W 030 = 3 kW 100 = 10 kW
010 = 1 kW 050 = 5 kW 150 = 15 kW
020 = 2 kW 075 = 7.5 kW 220 = 22 kW

Input Power (VAC)
HV = High voltage (230…480V AC)
Blank = Standard (115…230V AC)

Intelligent Positioning Drive

Bulletin Number

Ultraware Software Catalog Number

The Ultra3000 drives are configured by using Ultraware software (catalog number 2098-UWCPRG).

Motion Control Accessories

This chapter includes compatibility tables, dimensions, specifications, and catalog numbers for the accessories that support the Kinetix Motion Control drive families.

2090-Series Motor/Actuator Cables

A wide variety of motor/actuator cables with rugged DIN connectors are available for connecting your motion control system. Standard motor power and feedback cables are available for all Allen-Bradley servo motors and actuators. Continuous-flex rated cables, intended for moving applications, are also available. Continuous-flex extension and standard transition cables are also available for your applications that require them.

IMPORTANT	All flying-lead feedback cables require breakout components or connector kits for drive-end terminations. Refer to Breakout Components and Connector Kits beginning on page 440 for catalog numbers and descriptions.

IMPORTANT	Standard (non-flex) cables have a regular maintenance and installation bend radius of 10 times (10x) the cable diameter. For flexing applications, continuous-flex cables have an operational bend radius of 12 times (12x) the cable diameter.

2090-Series Motor/Acutator Cables with SpeedTec DIN Connectors Overview

2090-Series motor/actuator cables with SpeedTec DIN connectors let OEMs and end-users standardize their machines on a common motor cable family. These cables, designed by Rockwell Automation for optimal performance with Allen-Bradley servo drives, servo motors, and linear actuators, offer best-in-class features and standards compliance. Innovative features, configuration options, and accessories provide machine builders with complete control of the cable requirements in their machines.

SpeedTec DIN Cable Features

- NFPA 79 Compliant
- UL Listed bulk cable with 600V insulation rating for use in cable trays and exposed run applications. Also carries applicable Canadian approvals
  - Type TC-ER: Power-only and power-with-brake cables
  - Type PLTC-ER: Feedback cable optimized for high-resolution feedback motors
- SpeedTec connection system yields ¼-turn quick connections with positive metallic keying
- DESINA compliant jacket coloring (orange for power, green for feedback) for easy identification and separation of cables in a machine
- Cables are included in the Rockwell Automation servo system Declaration of Conformity (DoC)
- Continuous flex cables are suitable for 20 million flex-cycles
  - Continuous-flex cables are available in standard and extension cable configurations
- Comprehensive accessories optimize the use of cables in machines

2090-Series Motor/Actuator Cables Overview

Feedback Cable Descriptions

Standard Cable Cat. No.	**Description**	**Connector Style**
2090-CFBM7DF-CEAA*xx*	Feedback (drive-end flying-lead)	SpeedTec DIN
2090-CFBM7DD-CEAA*xx*	Feedback (drive-end connector)	
2090-XXNFMF-S*xx*	Feedback	Threaded DIN
2090-CFBM4E2-CATR	Feedback transition (1)	
2090-CFBM6DF-CBAA*xx*	Feedback (drive-end flying-lead)	Circular Plastic
2090-CFBM6DD-CCAA*xx*	Feedback (drive-end connector)	
2090-DANFCT-S*xx*	Feedback	Rectangular Plastic

(1) Threaded DIN connector (motor end) and bayonet connector for existing 2090-XXNFMP-S*xx* cable. Refer to SpeedTec DIN Continuous-flex Extension Cables on page 398.

Continuous-flex Cable Cat. No.	**Description**	**Connector Style**
2090-CFBM7DF-CDAF*xx*	Feedback (drive-end flying-lead)	SpeedTec DIN
2090-CFBM7DF-CEAF*xx*		
2090-CFBM7DD-CEAF*xx*	Feedback (drive-end connector)	
2090-CFBM7E7-CDAF*xx*	Feedback extension	
2090-CFBM7E7-CEAF*xx*		
2090-CFBM4DF-CDAF*xx*	Feedback (drive-end flying-lead)	Threaded DIN

TIP Feedback cables with the CE designation, for example 2090-CFBM7DF-CEAF*xx*, are intended for high-resolution encoder or resolver applications. Feedback cables with the CD designation, for example 2090-CFBM7DF-CDAF*xx*, are intended for high-resolution encoder or incremental applications.

Power/Brake Cable Descriptions

Standard Cable Cat. No.	**Description**	**Connector Style**
2090-CPBM7DF-*xx*AA*xx*	Power/brake	SpeedTec DIN
2090-CPWM7DF-*xx*AA*xx*	Power-only	
2090-XXNPMF-*xx*S*xx*	Power/brake	Threaded DIN
2090-CPBM4E2-*xx*TR	Power/brake transition (1)	
2090-CPWM4E2-*xx*TR	Power-only transition (1)	
2090-CPBM6DF-16AA*xx*	Power/brake	Circular Plastic
2090-CPWM6DF-16AA*xx*	Power-only	
2090-DANPT-16S*xx*		Rectangular Plastic
2090-DANBT-18S*xx*	Brake	

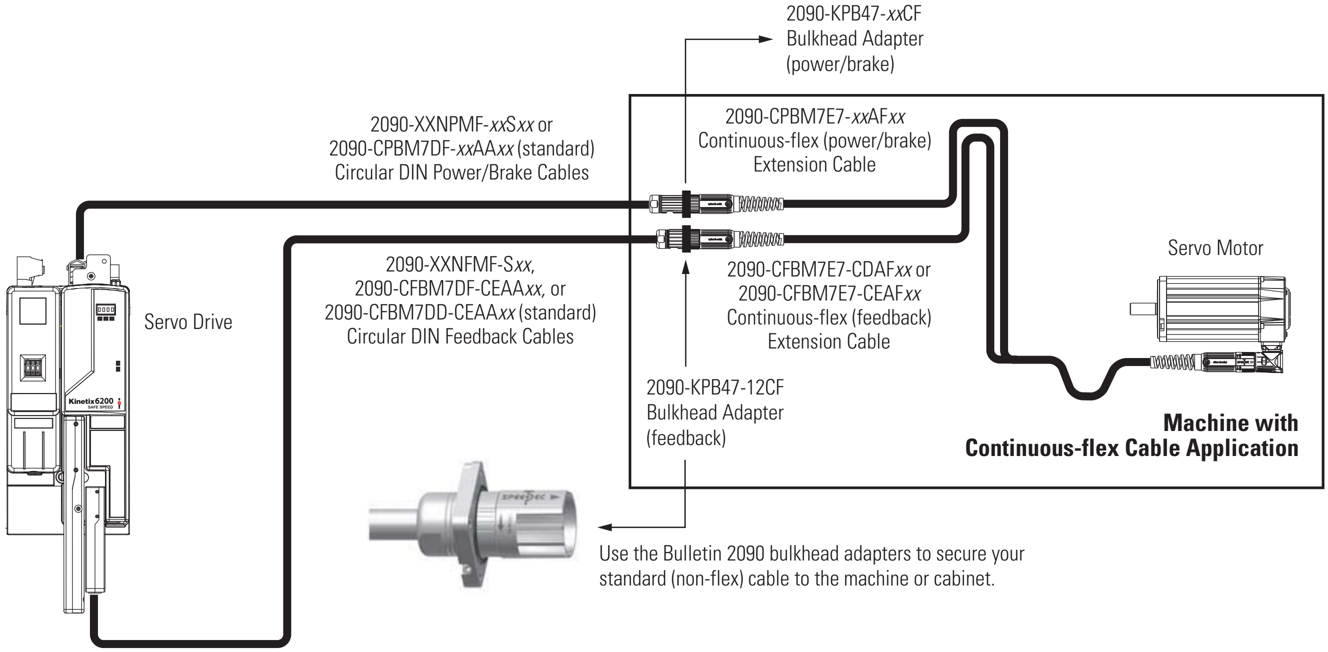
(1) Threaded DIN connector (motor end) and bayonet connector for existing 2090-XXNPMP-*xx*S*xx* cable. Refer to SpeedTec DIN Continuous-flex Extension Cables on page 398.

Continuous-flex Cable Cat. No.	**Description**	**Connector Style**
2090-CPBM7DF-*xx*AF*xx*	Power/brake	SpeedTec DIN
2090-CPWM7DF-*xx*AF*xx*	Power only	
2090-CPBM7E7-*xx*AF*xx*	Power/brake extension	
2090-CPBM4DF-*xx*AF*xx*	Power/brake	Threaded DIN
2090-CPWM4DF-*xx*AF*xx*	Power only	

SpeedTec DIN Continuous-flex Extension Cables

Motor power and feedback extension cables provide continuous-flex cable technology between your standard cable and the continuous-flex application. Extension cables are available in lengths up to 30 m (98.4 ft). Extension power cables are available in 16, 14, 10, and 8 AWG.

Typical Extension Cable Application with Bulkhead Adapter



Use the Bulletin 2090 bulkhead adapters to secure your standard (non-flex) cable to the machine or cabinet.

Continuous-flex Extension Feedback Cables

Continuous-flex Cable Cat. No.	Description	Applications
2090-CFBM7E7-CDAF*xx*	Feedback extension cable, SpeedTec DIN (male/female) connectors	Intended for high-resolution or incremental encoder applications.
2090-CFBM7E7-CEAF*xx*		Intended for high-resolution encoder or resolver applications.

Continuous-flex Extension Power Cables

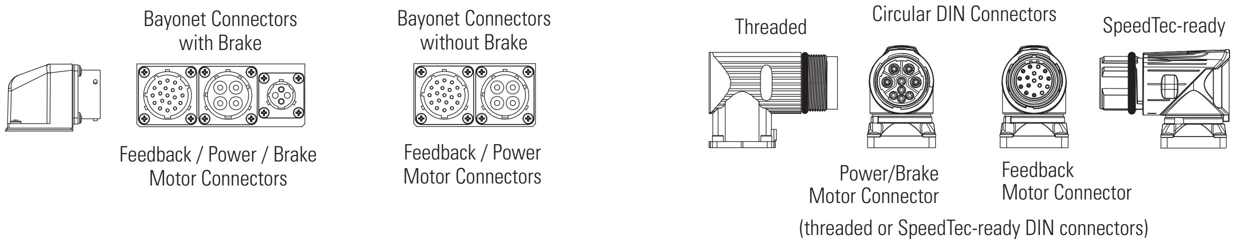
Continuous-flex Cable Cat. No.	Description
2090-CPBM7E7-16AF*xx*	Power/brake extension cable, SpeedTec DIN (male/female) connectors
2090-CPBM7E7-14AF*xx*	
2090-CPBM7E7-10AF*xx*	
2090-CPBM7E7-08AF*xx*	

2090-Series Motor Power and Feedback Transition Cables

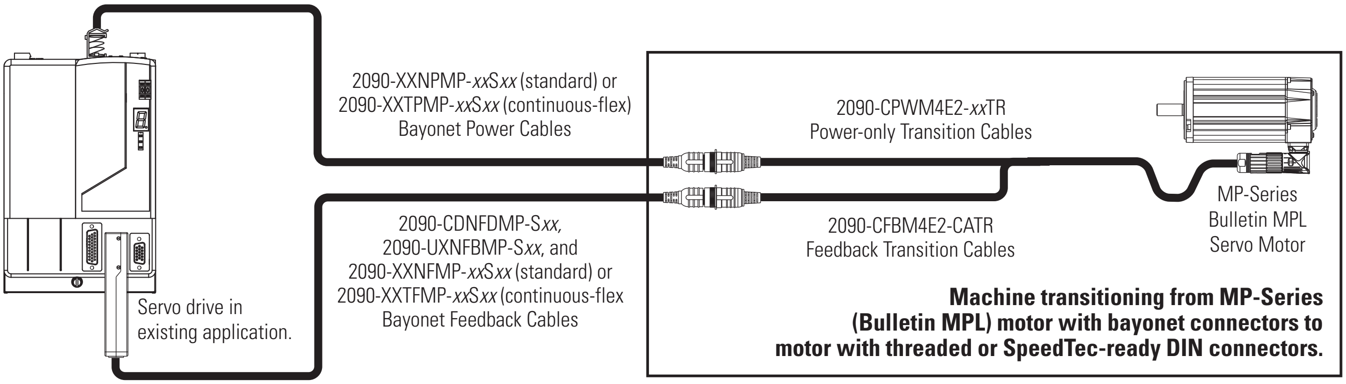
Motor power/brake and feedback transition cables support installations where MP-Series (Bulletin MPL) motors with bayonet connectors were recently replaced by the same motor with circular DIN connectors. These 0.5 m (19.7 in.) cables provide a seamless transition between your new motor and existing power, brake, and feedback cables.

TIP Brake contacts for motors with bayonet connectors are in a separate connector. Power/brake cables with circular DIN connectors (either threaded or SpeedTec) include brake contacts in the power/brake connector.

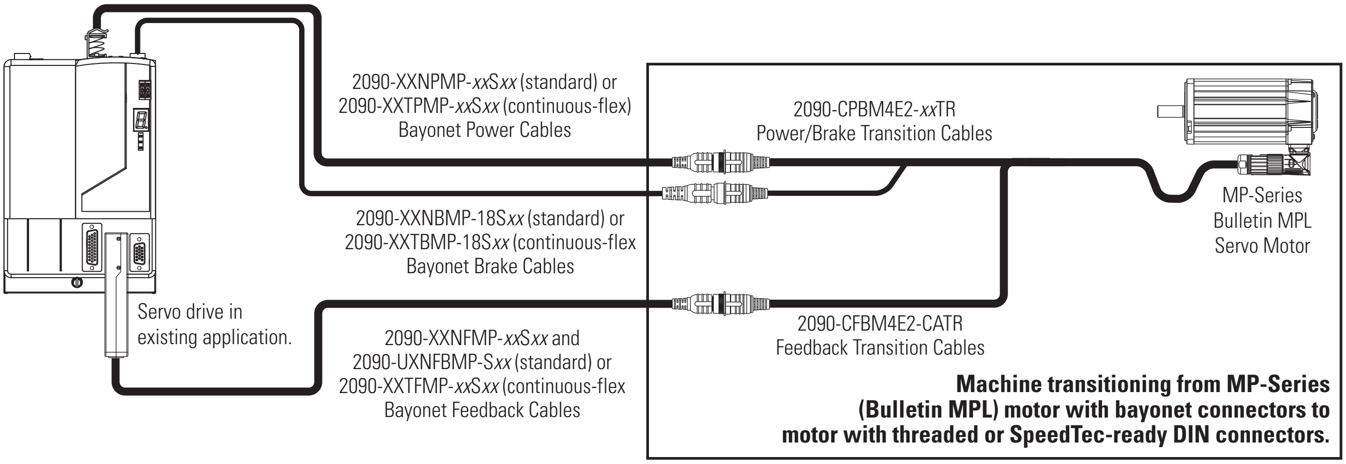
Bayonet and Circular DIN Motor Connectors



Transition Cable Application (power-only cable)



Transition Cable Application (power/brake cable)



Circular DIN Connector Compatibility Overview

Motors equipped with either threaded or SpeedTec circular DIN connectors are listed below. Circular DIN motor connectors rotate up to 180° and combine power and brake wires in the same connector.

Motor Connector/Cable Plug Compatibility

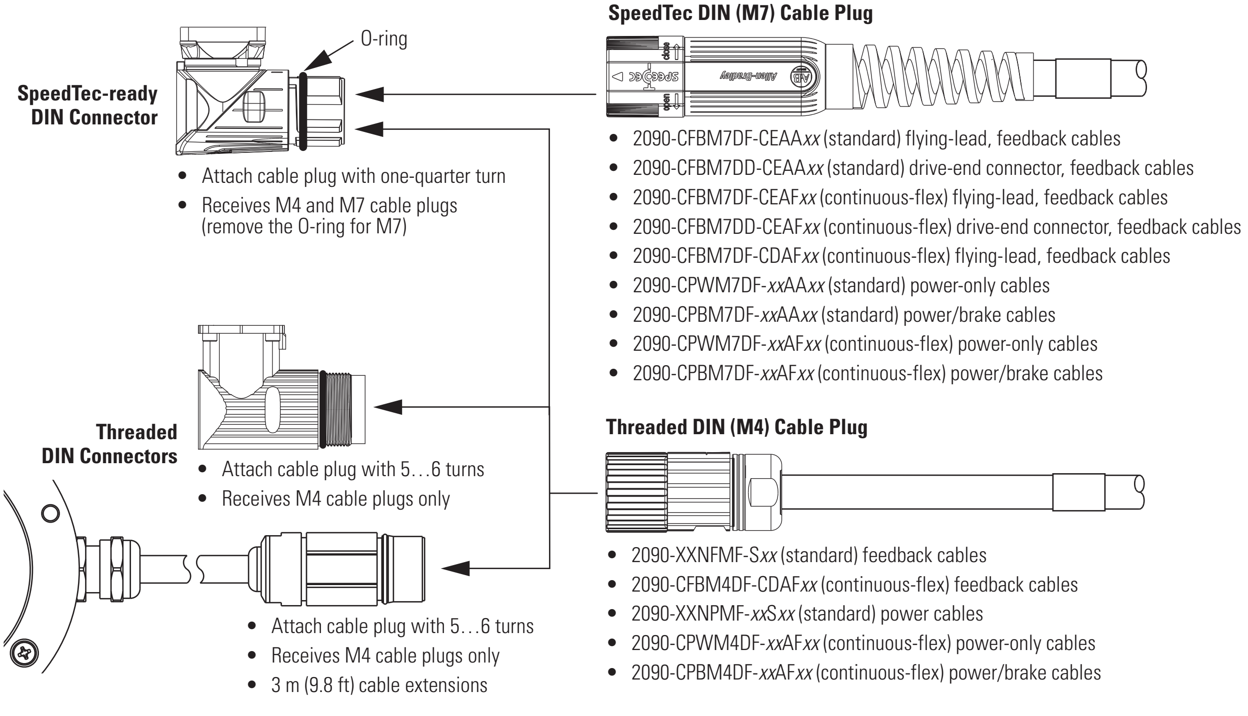
Motor/Actuator Cat. No.	Connector Type	Power-only or Power/Brake Cables	Feedback Cables
MPL-A/B3*xxx*, MPL-A/B4*xxx*, MPL-A/B45*xxx*, MPL-A/B5*xxx* MPL-B6*xxx*, MPL-B8*xxx,* and MPL-B9*xxx* MPM-A/B*xxxx* MPF-A/B*xxxx* RDB-B*xxxx* (1) MPAR-A/B3*xxxx* MPAI-A/B*xxxxx* LDC-C*xxxxxxx* and LDL-*xxxxxxx* (1)	SpeedTec-ready DIN	2090-CP*x*M7DF-*xx*AA*xx* 2090-CP*x*M7DF-*xx*AF*xx*	2090-CFBM7DF-CEAA*xx* 2090-CFBM7DD-CEAA*xx* 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* 2090-CFBM7DF-CDAF*xx*
		• M7 cable plugs • Remove the O-ring • Adapts to SpeedTec-ready connectors only	
MPL-A/B15*xxx* and MPL-A/B2*xxx* MPAS-A/B*xxxx* MPMA-A/B*xxxx* MPAR-A/B1*xxxx* and MPAR-A/B2*xxxx* HPK-B/E*xxxxx*	Threaded DIN	2090-XXNPMF-S*xx* or 2090-CP*x*M4DF-*xx*AF*xx*	2090-XXNFMF-S*xx* or 2090-CFBM4DF-CDAF*xx*
		• M4 cable plugs • Adapts to threaded or SpeedTec-ready connectors • O-ring dampens the effects of vibration to create a more secure connection	
MPS-A/B*xxxx*	Threaded DIN with 3 m (9.8 ft) cable extensions		

(1) The LDC-Series and LDL-Series linear motors and Bulletin RDB direct-drive motors have SpeedTec-ready DIN (M7) motor connectors, but require the additional conductors included with 2090-CFBM7DF-CDAF*xx* (continuous-flex) feedback cables. For standard (non-flex) applications, use 2090-XXNFMF-S*xx* feedback cables.

IMPORTANT Motors equipped with SpeedTec-ready DIN connectors are fully compatible with threaded DIN (M4) cable plugs. SpeedTec-ready DIN motor connectors are also compatible with SpeedTec DIN (M7/E7) cable plugs when the o-ring on the motor connector is removed.

Motors equipped with threaded DIN (M4) connectors are compatible only with threaded DIN (M4) cable plugs.

Typical Circular DIN Cable Applications



SpeedTec-ready DIN Connector

- Attach cable plug with one-quarter turn
- Receives M4 and M7 cable plugs (remove the O-ring for M7)

SpeedTec DIN (M7) Cable Plug

- 2090-CFBM7DF-CEAA*xx* (standard) flying-lead, feedback cables
- 2090-CFBM7DD-CEAA*xx* (standard) drive-end connector, feedback cables
- 2090-CFBM7DF-CEAF*xx* (continuous-flex) flying-lead, feedback cables
- 2090-CFBM7DD-CEAF*xx* (continuous-flex) drive-end connector, feedback cables
- 2090-CFBM7DF-CDAF*xx* (continuous-flex) flying-lead, feedback cables
- 2090-CPWM7DF-*xx*AA*xx* (standard) power-only cables
- 2090-CPBM7DF-*xx*AA*xx* (standard) power/brake cables
- 2090-CPWM7DF-*xx*AF*xx* (continuous-flex) power-only cables
- 2090-CPBM7DF-*xx*AF*xx* (continuous-flex) power/brake cables

Threaded DIN Connectors

- Attach cable plug with 5…6 turns
- Receives M4 cable plugs only

- Attach cable plug with 5…6 turns
- Receives M4 cable plugs only
- 3 m (9.8 ft) cable extensions

Threaded DIN (M4) Cable Plug

- 2090-XXNFMF-S*xx* (standard) feedback cables
- 2090-CFBM4DF-CDAF*xx* (continuous-flex) feedback cables
- 2090-XXNPMF-*xx*S*xx* (standard) power cables
- 2090-CPWM4DF-*xx*AF*xx* (continuous-flex) power-only cables
- 2090-CPBM4DF-*xx*AF*xx* (continuous-flex) power/brake cables

2090-Series Motor/Actuator Cable Selection

These tables provide flying-lead motor cable catalog numbers for drive/motor combinations. Most motor brake wires are in the power cable, so a separate brake cable is not required (except where noted).

IMPORTANT	The MP-Series low-inertia motors on this page are equipped with DIN connectors (specified by 4 or 7 in the catalog number) and are not compatible with cables designed for motors equipped with bayonet connectors (specified by 2 in the catalog number). The motors with bayonet connectors (for example, MPL-A310P-*xx*2*x*AA) are being discontinued and require 2090-XXNFMP-S*xx* (bayonet) cables. For help with migration or to select bayonet cables, contact your Rockwell Automation sales representative.

MP-Series (Bulletin MPL) Motor Feedback Cables

Motor Cat. No.	Compatible Drive Cat. No.	Feedback Type	Feedback Cable Cat. No.
MPL-A15*xxx*-V/E*x*4*x*AA, MPL-A2*xxx*-V/E*x*4*x*AA	2093-AC05-MP*x* or 2093-AM*xx* 2094-AC*xx*-M*xx*-S or 2094-AM*xx*-S 2097-V3*xxxx* 2098-DSD-*xxx*	Multi-turn High-resolution Absolute or Single-turn High-resolution Encoder Feedback	2090-XXNFMF-S*xx* (standard) 2090-CFBM4DF-CDAF*xx* (continuous-flex)
MPL-B15*xxx*-V/E*x*4*x*AA, MPL-B2*xxx*-V/E*x*4*x*AA	2094-BC*xx*-M*xx*-S or 2094-BM*xx*-S 2094-BC*xx*-M*xx*-M or 2094-BM*xx*-M 2097-V3*xxxx* 2098-DSD-HV*xxx*		
MPL-A3*xxx*-M/S*x*7*x*AA, MPL-A4*xxx*-M/S*x*7*x*AA, MPL-A45*xxx*-M/S*x*7*x*AA, MPL-A5*xxx*-M/S*x*7*x*AA	2093-AC05-MP*x* or 2093-AM*xx* 2094-AC*xx*-M*xx*-S or 2094-AM*xx*-S 2097-V3*xxxx* 2098-DSD-*xxx* 2098-IPD-*xxx*		2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex)
MPL-B3*xxx*-M/S*x*7*x*AA, MPL-B4*xxx*-M/S*x*7*x*AA, MPL-B45*xxx*-M/S*x*7*x*AA, MPL-B5*xxx*-M/S*x*7*x*AA, MPL-B6*xxx*-M/S*x*7*x*AA, MPL-B8*xxx*-M/S*x*7*x*AA, MPL-B9*xxx*-M/S*x*7*x*AA	2094-BC*xx*-M*xx*-S or 2094-BM*xx*-S 2094-BC*xx*-M*xx*-M or 2094-BM*xx*-M 2097-V3*xxxx* 2098-DSD-HV*xxx* 2098-IPD-HV*xxx* 2099-BM*xx*-S		
MPL-A15*xxx*-H*x*4*x*AA, MPL-A2*xxx*-H*x*4*x*AA	2093-AC05-MP*x* or 2093-AM*xx* 2094-AC*xx*-M*xx*-S or 2094-AM*xx*-S 2097-V3*xxxx* 2098-DSD-*xxx*	Incremental (1) Feedback	2090-XXNFMF-S*xx* (standard) 2090-CFBM4DF-CDAF*xx* (continuous-flex)
MPL-B15*xxx*-H*x*4*x*AA, MPL-B2*xxx*-H*x*4*x*AA	2094-BC*xx*-M*xx*-S or 2094-BM*xx*-S 2094-BC*xx*-M*xx*-M or 2094-BM*xx*-M 2097-V3*xxxx* 2098-DSD-HV*xxx*		
MPL-A3*xxx*-H*x*7*x*AA, MPL-A4*xxx*-H*x*7*x*AA, MPL-A45*xxx*-H*x*7*x*AA	2093-AC05-MP*x* or 2093-AM*xx* 2094-AC*xx*-M*xx*-S or 2094-AM*xx*-S 2097-V3*xxxx* 2098-DSD-*xxx* 2098-IPD-*xxx*		2090-XXNFMF-S*xx* (standard) 2090-CFBM7DF-CDAF*xx* (continuous-flex)
MPL-B*xxxx*-R*x*7*x*AA	2094-BC*xx*-M*xx*-S or 2094-BM*xx*-S	Resolver Feedback (1)	2090-CFBM7DF-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* (continuous-flex)

(1) Not all MP-Series low-inertia motors are available with incremental and resolver feedback options.

Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

IMPORTANT The MP-Series low-inertia motors on this page are equipped with DIN connectors (specified by 4 or 7 in the catalog number) and are not compatible with cables designed for motors equipped with bayonet connectors (specified by 2 in the catalog number). The motors with bayonet connectors (for example, MPL-A310P-*xx*2*x*AA) are being discontinued and require 2090-XXNPMP-*xx*S*xx* (bayonet) cables. For help with migration or to select bayonet cables, contact your Rockwell Automation sales representative.

MP-Series (230V) Low Inertia Motors	**Power Cable Cat. No.**
MPL-A15*xxx*-*xx*4*x*AA, MPL-A2*xxx*-*xx*4*x*AA	2090-XXNPMF-16S*xx* (standard) 2090-CP*x*M4DF-16AF*xx* (continuous-flex)
MPL-A3*xxx*-*xx*7*x*AA	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)
MPL-A420P-*xx*7*x*AA, MPL-A430H-*xx*7*x*AA	
MPL-A4530F-*xx*7*x*AA, MPL-A4540C-*xx*7*x*AA	
MPL-A430P-*xx*7*x*AA, MPL-A4530K-*xx*7*x*AA, MPL-A4540F-*xx*7*x*AA	2090-CP*x*M7DF-14AA*xx* (standard) 2090-CP*x*M7DF-14AF*xx* (continuous-flex)
MPL-A4560F-*xx*7*x*AA	2090-CP*x*M7DF-12AA*xx* (standard)
MPL-A520K-*xx*7*x*AA	2090-CP*x*M7DF-10AA*xx* (standard) 2090-CP*x*M7DF-10AF*xx* (continuous-flex)
MPL-A540K-*xx*7*x*AA, MPL-A560F-*xx*7*x*AA	2090-CP*x*M7DF-08AA*xx* (standard) 2090-CP*x*M7DF-08AF*xx* (continuous-flex)

MP-Series (460V) Low Inertia Motors	**Power Cable Cat. No.**
MPL-B15*xxx*-*xx*4*x*AA, MPL-B2*xxx*-*xx*4*x*AA	2090-XXNPMF-16S*xx* (standard) 2090-CP*x*M4DF-16AF*xx* (continuous-flex)
MPL-B3*xxx*-*xx*7*x*AA	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)
MPL-B4*xxx*-*xx*7*x*AA	
MPL-B45*xxx*-*xx*7*x*AA	
MPL-B520K-*xx*7*x*AA	
MPL-B540D-*xx*7*x*AA, MPL-B540K-*xx*7*x*AA, MPL-B560F-*xx*7*x*AA	2090-CP*x*M7DF-14AA*xx* (standard) 2090-CP*x*M7DF-14AF*xx* (continuous-flex)
MPL-B580F-*xx*7*x*AA, MPL-B580J-*xx*7*x*AA	2090-CP*x*M7DF-10AA*xx* (standard) 2090-CP*x*M7DF-10AF*xx* (continuous-flex)
MPL-B640F-*xx*7*x*AA	
MPL-B660F-*xx*7*x*AA, MPL-B680D-*xx*7*x*AA, MPL-B960B-*xx*7*x*AA, MPL-B980B-*xx*7*x*AA	2090-CP*x*M7DF-08AA*xx* (standard) 2090-CP*x*M7DF-08AF*xx* (continuous-flex)
MPL-B680F-*xx*7*x*AA, MPL-B860D-*xx*7*x*AA, MPL-B880C-*xx*7*x*AA,	2090-CPBM7DF-06AA*xx* (standard)
MPL-B880D-*xx*7*x*AA	2090-CPBM7DF-04AA*xx* (standard)
MPL-B960C-*xx*7*x*AA, MPL-B960D-*xx*7*x*AA, MPL-B980C-*xx*7*x*AA, MPL-B980D-*xx*7*x*AA	
MPL-B980E-*xx*7*x*AA	2090-CPBM7DF-02AA*xx* (standard)

Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

MP-Series Food Grade Motors

Motor Cat. No.	Drive Compatibility	Feedback Type	Feedback Cable Cat. No.
MPF-A*xxxx*-M/S	2093-AC05-MP*x* or 2093-AM*xx* 2094-AC*xx*-M*xx*-S or 2094-AM*xx*-S 2097-V3*xxxx* 2098-DSD-*xxx* 2098-IPD-*xxx*	Multi-turn High Resolution Absolute or Single-turn High Resolution Encoder Feedback	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex)
MPF-B*xxxx*-M/S	2094-BC*xx*-M*xx*-S or 2094-BM*xx*-S 2094-BC*xx*-M*xx*-M or 2094-BM*xx*-M 2097-V3*xxxx* 2098-DSD-HV*xxx* 2098-IPD-HV*xxx*		

MP-Series (230V) Food Grade Motors	Power Cable Cat. No.
MPF-A310P, MPF-A320H, MPF-A320P, and MPF-A330P	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)
MPF-A430H	
MPF-A430P, MPF-A4530K and MPF-A4540F	2090-CP*x*M7DF-14AA*xx* (standard) 2090-CP*x*M7DF-14AF*xx* (continuous-flex)
MPF-A540K	2090-CP*x*M7DF-08AA*xx* (standard) 2090-CP*x*M7DF-08AF*xx* (continuous-flex)

MP-Series (460V) Food Grade Motors	Power Cable Cat. No.
MPF-B310P, MPF-B320P, and MPF-B330P	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)
MPF-B430P, MPF-B4530K, and MPF-B4540F	
MPF-B540K	2090-CP*x*M7DF-10AA*xx* (standard) 2090-CP*x*M7DF-10AF*xx* (continuous-flex)

MP-Series Stainless Steel Motors

Motor Cat. No.	Drive Compatibility	Feedback Type	Feedback Cable Cat. No.
MPS-A*xxxx*-M/S	2093-AC05-MP*x* or 2093-AM*xx* 2094-AC*xx*-M*xx*-S or 2094-AM*xx*-S 2097-V3*xxxx* 2098-DSD-*xxx* 2098-IPD-*xxx*	Multi-turn High Resolution Absolute or Single-turn High Resolution Encoder Feedback	2090-XXNFMF-S*xx* (standard) 2090-CFBM4DF-CDAF*xx* (continuous-flex)
MPS-B*xxxx*-M/S	2094-BC*xx*-M*xx*-S or 2094-BM*xx*-S 2094-BC*xx*-M*xx*-M or 2094-BM*xx*-M 2097-V3*xxxx* 2098-DSD-HV*xxx* 2098-IPD-HV*xxx*		

MP-Series (230V) Stainless Steel Motors	Power Cable Cat. No.
MPS-A330P	2090-XXNPMF-16S*xx* (standard) 2090-CP*x*M4DF-16AF*xx* (continuous-flex)
MPS-A4540F	

MP-Series (460V) Stainless Steel Motors	Power Cable Cat. No.
MPS-B330P	2090-XXNPMF-16S*xx* (standard) 2090-CP*x*M4DF-16AF*xx* (continuous-flex)
MPS-B4540F	
MPS-B560F	2090-XXNPMF-14S*xx* (standard) 2090-CP*x*M4DF-14AF*xx* (continuous-flex)

Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

MP-Series Medium Inertia Motors

Motor Cat. No.	Drive Compatibility	Feedback Type	Feedback Cable Cat. No.
MPM-A*xxxxx*-M/S	2093-AC05-MP*x* or 2093-AM*xx* 2094-AC*xx*-M*xx*-S or 2094-AM*xx*-S 2097-V3*xxxx* 2098-DSD-*xxx*	Multi-turn High Resolution Absolute or Single-turn High Resolution Encoder Feedback	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex)
MPM-B*xxxxx*-M/S	2094-BC*xx*-M*xx*-S or 2094-BM*xx*-S 2094-BC*xx*-M*xx*-M or 2094-BM*xx*-M 2097-V3*xxxx* 2098-DSD-HV*xxx* 2099-BM*xx*-S		
MPM-A*xxxxx*-2	2094-AC*xx*-M*xx*-S or 2094-AM*xx*-S	Resolver Feedback (1)	2090-CFBM7DF-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* (continuous-flex)
MPM-B*xxxxx*-2	2094-BC*xx*-M*xx*-S or 2094-BM*xx*-S		

(1) Not all MP-Series medium-inertia motors are available with the resolver feedback option.

MP-Series (200V Class) Medium Inertia Motors	Power Cable Cat. No.
MPM-A115*xx*	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)
MPM-A1302F	2090-CP*x*M7DF-14AA*xx* (standard) 2090-CP*x*M7DF-14AF*xx* (continuous-flex)
MPM-A1304F	2090-CP*x*M7DF-12AA*xx* (standard)
MPM-A1651F	2090-CP*x*M7DF-10AA*xx* (standard) 2090-CP*x*M7DF-10AF*xx* (continuous-flex)
MPM-A1652F, MPM-A1653F	2090-CP*x*M7DF-08AA*xx* (standard) 2090-CP*x*M7DF-08AF*xx* (continuous-flex)
MPM-A215*xx*	2090-CPBM7DF-06AA*xx* (standard)

MP-Series (400V Class) Medium Inertia Motors	Power Cable Cat. No.
MPM-B1151*x*, MPM-B1152*x*	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)
MPM-B1153E, MPM-B1153F	
MPM-B1302F, MPM-B1302M, MPM-B1304C, MPM-B1304E	
MPM-B1651C, MPM-B1652C	
MPM-B1153T	2090-CP*x*M7DF-14AA*xx* (standard) 2090-CP*x*M7DF-14AF*xx* (continuous-flex)
MPM-B1302T, MPM-B1304M	
MPM-B1651F, MPM-B1653C	
MPM-B1651M, MPM-B1652E, MPM-B1652F, MPM-B1653E	2090-CP*x*M7DF-10AA*xx* (standard) 2090-CP*x*M7DF-10AF*xx* (continuous-flex)
MPM-B2152C, MPM-B2153B	
MPM-B1653F	2090-CP*x*M7DF-08AA*xx* (standard) 2090-CP*x*M7DF-08AF*xx* (continuous-flex)
MPM-B2152F, MPM-B2152M, MPM-B2153E, MPM-B2153F, MPM-B2154B, MPM-B2154E, MPM-B2154F	

Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

RDD-Series Direct Drive Motors

Motor Cat. No.	Drive Compatibility	Feedback Type	Feedback Cable Cat. No.
RDB-B*xxxx*-7/3	2094-BC*xx*-M*xx*-S or 2094-BM*xx*-S 2094-BC*xx*-M*xx*-M or 2094-BM*xx*-M 2099-BM*xx*-S	Multi-turn High Resolution Absolute or Single-turn High Resolution Encoder Feedback	2090-XXNFMF-S*xx* (standard) 2090-CFBM7DF-CDAF*xx* (continuous-flex)

RDD-Series (400V Class) Direct Drive Motors	Power Cable Cat. No.
RDB-B21519, RDB-B21529	2090-CPWM7DF-16AA*xx* (standard) 2090-CPWM7DF-16AF*xx* (continuous-flex)
RDB-B29014, RDB-B29016, RDB-B29024	
RDB-B2151C, RDB-B21539	2090-CPWM7DF-14AA*xx* (standard) 2090-CPWM7DF-14AF*xx* (continuous-flex)
RDB-B29019, RDB-B29034	
RDB-B2152C	2090-CPWM7DF-12AA*xx* (standard)
RDB-B29026	
RDB-B2153C	2090-CPWM7DF-10AA*xx* (standard) 2090-CPWM7DF-10AF*xx* (continuous-flex)
RDB-B29036, RDB-B41014	
RDB-B29029, RDB-B41016, RDB-B41024	2090-CPWM7DF-08AA*xx* (standard) 2090-CPWM7DF-08AF*xx* (continuous-flex)
RDB-B29039, RDB-B41018, RDB-B41026, RDB-B41035	2090-CPBM7DF-06AA*xx* (standard)

Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

HPK-Series Asynchronous Servo Motors

Cat. No.	Drive Compatibility	Feedback Type	Feedback Cable Cat. No.
HPK-B*xxxxx*-M/S HPK-E*xxxxx*-M/S	2099-BM*xx*-S	Multi-turn High Resolution Absolute or Single-turn High Resolution Encoder Feedback	2090-XXNFMF-S*xx* (standard) 2090-CFBM4DF-CDAF*xx* (continuous-flex)

HPK-Series Asynchronous Servo Motors	Power Cable Cat. No.
All HPK-B*xxxx* or HPK-E*xxxx* motors	Customer Supplied

Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

TL-Series Low Inertia Motors

Cat. No.	Drive Compatibility	Feedback Type	Feedback Cable Cat. No.
TLY-A*xxxx*-H	2093-AC05-MP*x* or 2093-AM*xx* 2094-AC*xx*-M*xx*-S or 2094-AM*xx*-S 2097-V3*xxxx* 2098-DSD-*xxx* 2098-IPD-*xxx* 2071-A*xx*	Incremental	2090-CFBM6DF-CBAA*xx* (flying lead) or 2090-CFBM6DD-CCAA*xx* (premolded connector)
TLY-A*xxxx*-B	2093-AC05-MP*x* or 2093-AM*xx* 2097-V3*xxxx*	Multi-turn High Resolution Absolute Encoder Feedback	
	2071-A*xx*		2090-CFBM6DF-CBAA*xx* (flying lead)
TL-A*xxxx*-B	2071-A*xx*		2090-DANFCT-S*xx*

TL-Series (230V) Motors	Power Cable Cat. No.
TLY-A*xxxx*-H	2090-CPBM6DF-16AA*xx* (power and brake) 2090-CPWM6DF-16AA*xx* (power without brake)
TLY-A*xxxx*-B	
TL-A*xxxx*-B	2090-DANPT-16S*xx*

TL-Series (230V) Motors	Brake Cable Cat. No.
TL-A*xxxx*-B motors	2090-DANBT-18S*xx*

Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410. For N-Series retrofit cable information, refer to Transition Plates for N-Series Retrofit on page 81.

IMPORTANT TL-A*xxxx*-B motors have rectangular plastic connectors and are intended for use with Kinetix 3 (Bulletin 2071) servo drives. The TLY-A*xxxx* motors have circular plastic connectors and are intended for use with Bulletin 2093, 2094, 2097, and 2098 (230V) servo drives.

MP-Series Integrated Linear Stages

Actuator Cat. No.	Drive Compatibility	Feedback Type	Feedback Cable Cat. No.
MPAS-A*xxxxx*-V/A or MPMA-A	2093-AC05-MP*x* or 2093-AM*xx* 2094-AC*xx*-M*xx*-S or 2094-AM*xx*-S 2098-DSD-*xxx*	Multi-turn High Resolution Absolute Encoder Feedback	2090-XXNFMF-S*xx* (standard) 2090-CFBM4DF-CDAF*xx* (continuous-flex)
MPAS-A*xxxxx*-A or MPMA-A	2097-V3*xxxx*		
MPAS-A*xxxxx*-V	2071-A*xx*		
MPAS-B*xxxxx*-V/A or MPMA-B	2094-BC*xx*-M*xx*-S or 2094-BM*xx*-S 2094-BC*xx*-M*xx*-M or 2094-BM*xx*-M 2097-V3*xxxx* 2098-DSD-HV*xxx*		
MPAS-B*xxxxx*-A or MPMA-B	2097-V3*xxxx*		

MP-Series (230V) Integrated Linear Stages	Power Cable Cat. No.
MPAS-A*xxxxx*-V/A or MPMA-A	2090-XXNPMF-16S*xx* (standard) 2090-CP*x*M4DF-16AF*xx* (continuous-flex)

MP-Series (460V) Integrated Linear Stages	Power Cable Cat. No.
MPAS-B*xxxxx*-V/A or MPMA-B	2090-XXNPMF-16S*xx* (standard) 2090-CP*x*M4DF-16AF*xx* (continuous-flex)

Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

TL-Series Electric Cylinders

Actuator Cat. No.	Drive Compatibility	Feedback Type	Motor Feedback Cable
TLAR-A*xxxxx*	2093-AC05-MP*x* or 2093-AM*xx* 2097-V3*xxxx* 2071-A*xx*	Multi-turn High Resolution Absolute Encoder Feedback	2090-CFBM6DF-CBAA*xx* (flying-lead) or 2090-CFBM6DD-CCAA*xx* (premolded connector) standard

TL-Series (230V) Electric Cylinders	Motor Power Cable
TLAR-A*xxxxx*	2090-CPBM6DF-16AA*xx* (power and brake) standard 2090-CPWM6DF-16AA*xx* (power without brake) standard

Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

MP-Series Electric Cylinders

Actuator Cat. No.	Drive Compatibility	Feedback Type	Feedback Cable Cat. No.
MPAR-A1*xxxx* MPAR-A2*xxxx*	2093-AC05-MP*x* or 2093-AM*xx* 2094-AC*xx*-M*xx*-S or 2094-AM*xx*-S 2097-V3*xxxx* 2098-DSD-*xxx*	Multi-turn High Resolution Absolute Encoder Feedback	2090-XXNFMF-S*xx* (standard) 2090-CFBM4DF-CDAF*xx* (continuous-flex)
MPAR-B1*xxxx* MPAR-B2*xxxx*	2094-BC*xx*-M*xx*-S or 2094-BM*xx*-S 2094-BC*xx*-M*xx*-M or 2094-BM*xx*-M 2097-V3*xxxx* 2098-DSD-HV*xxx*		
MPAR-A3*xxxx*	2093-AC05-MP*x* or 2093-AM*xx* 2094-AC*xx*-M*xx*-S or 2094-AM*xx*-S 2097-V3*xxxx* 2098-DSD-*xxx*		2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex)
MPAR-B3*xxxx*	2094-BC*xx*-M*xx*-S or 2094-BM*xx*-S 2094-BC*xx*-M*xx*-M or 2094-BM*xx*-M 2097-V3*xxxx* 2098-DSD-HV*xxx*		

MP-Series (230V) Electric Cylinders	Power Cable Cat. No.
MPAR-A1*xxxx* MPAR-A2*xxxx*	2090-XXNPMF-16S*xx* (standard) 2090-CP*x*M4DF-16AF*xx* (continuous-flex)
MPAR-A3*xxxx*	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)

MP-Series (460V) Electric Cylinders	Power Cable Cat. No.
MPAR-B1*xxxx* MPAR-B2*xxxx*	2090-XXNPMF-16S*xx* (standard) 2090-CP*x*M4DF-16AF*xx* (continuous-flex)
MPAR-B3*xxxx*	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)

MP-Series Heavy Duty Electric Cylinders

Actuator Cat. No.	Drive Compatibility	Feedback Type	Feedback Cable Cat. No.
MPAI-A3*xxxx* MPAI-A4*xxxx*	2093-AC05-MP*x* or 2093-AM*xx* 2094-AC*xx*-M*xx*-S or 2094-AM*xx*-S 2097-V3*xxxx* 2098-DSD-*xxx*	Multi-turn High Resolution Absolute Encoder Feedback	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex)
MPAI-B3*xxxx* MPAI-B4*xxxx*	2094-BC*xx*-M*xx*-S or 2094-BM*xx*-S 2094-BC*xx*-M*xx*-M or 2094-BM*xx*-M 2097-V3*xxxx* 2098-DSD-HV*xxx*		

MP-Series (230V) Heavy Duty Electric Cylinders	Power Cable Cat. No.
MPAI-A3*xxxx* MPAI-A4*xxxx*	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)

MP-Series (460V) Heavy Duty Electric Cylinders	Power Cable Cat. No.
MPAI-B3*xxxx* MPAI-B4*xxxx*	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)

Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

LDC-Series Linear Motors

Cat. No.	Drive Compatibility	Feedback Type	Feedback Cable Cat. No.
LDC-C*xxxxxx*-*xx*T*x*1 (230V operation)	2093-AC05-MP*x* or 2093-AM*xx* 2094-AC*xx*-M*xx*-S or 2094-AM*xx*-S 2098-DSD-*xxx* 2071-A*xx*	Sin/Cos or TTL Encoder Feedback	2090-XXNFMF-S*xx* (standard) 2090-CFBM7DF-CDAF*xx* (continuous-flex)
LDC-C*xxxxxx*-*xx*T*x*1 (460V operation)	2094-BC*xx*-M*xx*-S or 2094-BM*xx*-S 2094-BC*xx*-M*xx*-M or 2094-BM*xx*-M 2098-DSD-HV*xxx*		

LDC-Series (230V or 460V operation) Linear Motors	Power Cable Cat. No.
LDC-C*xxxxxx*-*xx*T*x*1	2090-CPWM7DF-16AA*xx* (standard) 2090-CPWM7DF-16AF*xx* (continuous-flex)

Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

LDL-Series Linear Motors

Cat. No.	Drive Compatibility	Feedback Type	Feedback Cable Cat. No.
LDL-*xxxxxxx*-*xx*T*x*1	2093-AC05-MP*x* or 2093-AM*xx* 2094-AC*xx*-M*xx*-S or 2094-AM*xx*-S 2098-DSD-*xxx* 2071-A*xx*	Sin/Cos or TTL Encoder Feedback	2090-XXNFMF-S*xx* (standard) 2090-CFBM7DF-CDAF*xx* (continuous-flex)

LDL-Series Linear Motors	Power Cable Cat. No.
LDL-*xxxxxxx*-*xx*T*x*1	2090-CPWM7DF-16AA*xx* (standard) 2090-CPWM7DF-16AF*xx* (continuous-flex)

Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

2090-Series Motor/Actuator Cable Specifications

Power Cable Specifications

Power Cables [1] **Cat. No.**	**Cable Type/ Jacket Color**	**Description**	**Wire Size** AWG	**Weight, approx.** kg/m (lb/ft)	**Standard Cable Lengths** m (ft)
2090-XXNPMF-16S*xx*	Standard cable Industrial TPE, Black	Four conductor, 600V, shielded cable for three-phase power with additional four conductors, 18 AWG, shielded, for motor brake and spares.	16	0.276 (0.186)	1 (3.2) 2 (6.5) 3 (9.8) 4 (13.1) 5 (16.4) 7 (22.9) 9 (29.5) 12 (39.4) 15 (49.2) 20 (65.6) 25 (82.0) 30 (98.4) 40 (131.2) 60 (196.8) 90 (295.3)
2090-XXNPMF-14S*xx*			14	0.315 (0.212)	
2090-CPBM7DF-16AA*xx*	Standard cable Industrial TPE, Orange (DESINA, RAL 2003)	Four conductor, 600V, shielded cable for three-phase power with additional two conductors 18 AWG for motor brake.	16	0.212 (0.143)	
2090-CPBM7DF-14AA*xx*			14	0.261 (0.175)	
2090-CPBM7DF-12AA*xx*			12	0.349 (0.235)	
2090-CPBM7DF-10AA*xx*			10	0.492 (0.331)	
2090-CPBM7DF-08AA*xx*			8	0.708 (0.476)	
2090-CPBM7DF-06AA*xx*			6	1.038 (0.698)	
2090-CPBM7DF-04AA*xx*		Four conductor, 600V, shielded cable for three-phase power with additional two conductors 16 AWG for motor brake.	4	1.549 (1.041)	
2090-CPBM7DF-02AA*xx*			2	2.166 (1.455)	
2090-CPWM7DF-16AA*xx*		Four conductor, 600V, shielded cable for three-phase power.	16	0.136 (0.091)	
2090-CPWM7DF-14AA*xx*			14	0.185 (0.124)	
2090-CPWM7DF-12AA*xx*			12	0.248 (0.167)	
2090-CPWM7DF-10AA*xx*			10	0.418 (0.281)	
2090-CPWM7DF-08AA*xx*			8	0.644 (0.433)	
2090-CPWM6DF-16AA*xx*	Standard cable Industrial TPE, Black		16	0.138 (0.093)	1 (3.2) 2 (6.5) 3 (9.8) 4 (13.1) 5 (16.4) 7 (22.9) 9 (29.5) 12 (39.4) 15 (49.2) 20 (65.6) 25 (82.0) 30 (98.4)
2090-DANPT-16S*xx*			16		
2090-CPBM6DF-16AA*xx*		Four conductor, 600V, shielded cable for three-phase power with additional two conductors 18 AWG for motor brake.	16	0.180 (0.121)	
2090-CPBM4DF-16AF*xx*	Continuous-flex cable Industrial TPE, Orange (DESINA, RAL 2003)		16	0.228 (0.153)	1 (3.2) 2 (6.5) 3 (9.8) 4 (13.1) 5 (16.4) 7 (22.9) 9 (29.5) 12 (39.4) 15 (49.2) 20 (65.6) 25 (82.0) 30 (98.4) 40 (131.2) 50 (164.0) 60 (196.8) 75 (264.0) 90 (295.3)
2090-CPBM7DF-16AF*xx*			16		
2090-CPBM7DF-14AF*xx*			14	0.289 (0.194)	
2090-CPBM7DF-10AF*xx*			10	0.513 (0.345)	
2090-CPBM7DF-08AF*xx*			8	0.697 (0.468)	
2090-CPWM4DF-16AF*xx*		Four conductor, 600V, shielded cable for three-phase power.	16	0.154 (0.104)	
2090-CPWM7DF-16AF*xx*			16		
2090-CPWM7DF-14AF*xx*			14	0.196 (0.132)	
2090-CPWM7DF-10AF*xx*			10	0.452 (0.304)	
2090-CPWM7DF-08AF*xx*			8	0.666 (0.448)	

[1] 2090-CP*x*M4DF-*xx*A*xxx* and 2090-CP*x*M7DF-*xx*A*xxx* power cables are UL Listed, bulk cable, type TC-ER.

Brake Cable Specifications

Brake Cables Cat. No.	**Cable Type/ Jacket Color**	**Description**	**Wire Size** AWG	**Weight, approx.** kg/m (lb/ft)	**Standard Cable Lengths** m (ft)
2090-DANBT-18S*xx*	Standard cable Industrial TPE, Black	Two conductor, 600V, 18 AWG, shielded cable for motor brake.	18	0.070 (0.047)	1 (3.2) 2 (6.5) 3 (9.8) 4 (13.1) 5 (16.4) 7 (22.9) 9 (29.5) 12 (39.4) 15 (49.2) 20 (65.6) 25 (82.0) 30 (98.4)

Feedback Cable Specifications

Feedback Cables (1) (2) Cat. No.	**Cable Type/ Jacket Color**	**Description**	**Wire Size** AWG	**Weight, approx.** kg/m (lb/ft)	**Standard Cable Lengths** m (ft)
2090-XXNFMF-S*xx*	Standard cable Industrial TPE, Black	Threaded DIN connector (motor end) to flying leads (drive end), 30V.	28 Feedback 16 Power, 5V 22 Power, 9V	0.120 (1.35)	1 (3.2) 2 (6.5) 3 (9.8) 4 (13.1) 5 (16.4) 7 (22.9) 9 (29.5) 12 (39.4) 15 (49.2) 20 (65.6) 25 (82.0) 30 (98.4) 40 (131.2) 60 (196.8) 90 (295.3)
2090-CFBM7DD-CEAA*xx*	Standard cable Industrial TPE, Green (DESINA, RAL 6018)	SpeedTec DIN connector (motor end) to premolded connector (drive end), 600V.	22 All conductors	0.136 (0.092)	
2090-CFBM7DF-CEAA*xx*		SpeedTec DIN connector (motor end) to flying leads (drive end), 600V.			
2090-UXNFM-S*xx* (3)	Standard cable Industrial TPE, Black	Flying-leads (motor end) to premolded connector (drive end), 30V.	28 Feedback 16 Power, 5V 22 Power, 9V	0.120 (1.35)	1 (3.2) 3 (9.8) 9 (29.5) 15 (49.2) 30 (98.4)
2090-CFBM6DF-CBAA*xx*		Circular plastic connector (motor end) to premolded connector (drive end), 300V.	28 Feedback 16 Power, 5V 22 BAT+		1 (3.2) 2 (6.5) 3 (9.8) 4 (13.1) 5 (16.4) 7 (22.9) 9 (29.5) 12 (39.4) 15 (49.2) 20 (65.6) 25 (82.0) 30 (98.4)
2090-CFBM6DD-CCAA*xx*		Circular plastic connector (motor end) to premolded connector (drive end), 300V.	28 Feedback 16 Power, 5V		
2090-DANFCT-S*xx*		Rectangular plastic connector (motor end) to premolded connector (drive end), 30V.	28 Feedback 16 Power, 5V 22 BAT+	0.130 (0.088)	
2090-CFBM4DF-CDAF*xx*	Continuous-flex cable Industrial TPE, Green (DESINA, RAL 6018)	Threaded DIN connector (motor end) to flying leads (drive end), 600V.	26 Feedback 16 Power, 5V 22 Power, 9V	0.177 (0.119)	1 (3.2) 2 (6.5) 3 (9.8) 4 (13.1) 5 (16.4) 7 (22.9) 9 (29.5) 12 (39.4) 15 (49.2) 20 (65.6) 25 (82.0) 30 (98.4) 40 (131.2) 50 (164.0) 60 (196.8) 75 (264.0) 90 (295.3)
2090-CFBM7DF-CDAF*xx*		SpeedTec DIN connector (motor end) to flying leads (drive end), 600V.			
2090-CFBM7DF-CEAF*xx*			22 All conductors	0.143 (0.096)	
2090-CFBM7DD-CEAF*xx*		SpeedTec DIN connector (motor end) to premolded connector (drive end), 600V.			

(1) 2090-CFBM7*xx*-CEA*xxx* feedback cables are UL Listed, bulk cable, type PLTC-ER.

(2) 2090-CFBM4DF-CDA*xxx* and 2090-CFBM7*xx*-CDA*xxx* feedback cables are UL Listed, bulk cable, type CM.

(3) Use with 2090-KFBM4-CAAA (threaded) or 2090-KFBM7-CAAA (SpeedTec) DIN connector kit.

Continuous-flex Extension Cable Specifications

Extension Cable [1] [2] Cat. No.	Cable Type/ Jacket Color	Description	Weight, approx. kg/m (lb/ft)	Standard Cable Lengths m (ft)
2090-CPBM7E7-16AF*xx*	Power with brake Industrial TPE, Orange (DESINA, RAL 2003)	SpeedTec DIN connector plug on motor end to SpeedTec DIN receptacle for mating with 2090-Series standard power/brake cable, 600V.	0.228 (0.153)	1 (3.2) 2 (6.5) 3 (9.8) 4 (13.1) 5 (16.4) 7 (22.9) 9 (29.5) 12 (39.4) 15 (49.2) 20 (65.6) 25 (82.0) 30 (98.4)
2090-CPBM7E7-14AF*xx*			0.289 (0.194)	
2090-CPBM7E7-10AF*xx*			0.513 (0.345)	
2090-CPBM7E7-08AF*xx*			0.697 (0.468)	
2090-CFBM7E7-CDAF*xx*	Feedback Industrial TPE, Green (DESINA, RAL 6018)	SpeedTec DIN connector plug on motor end to SpeedTec DIN receptacle for mating with 2090-Series standard feedback cable, 600V.	0.153 (0.103)	
2090-CFBM7E7-CEAF*xx*			0.143 (0.096)	

(1) 2090-CPBM7E7-*xx*AF*xx* extension power cables are UL Listed, bulk cable, type TC-ER.

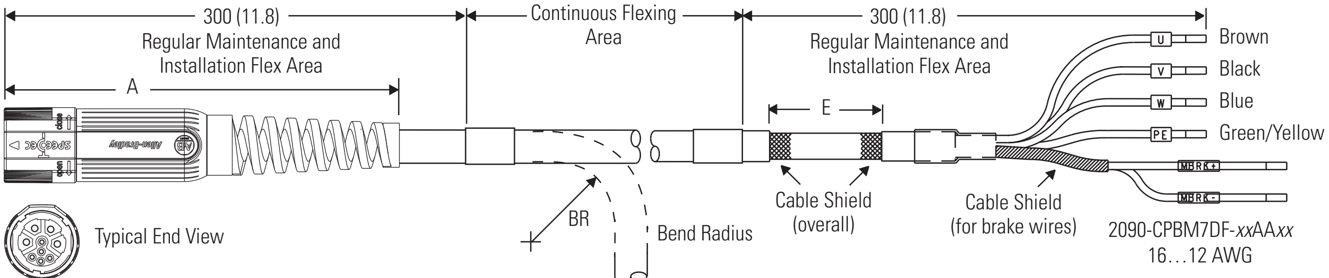
(2) 2090-CFBM7E7-CDAF*xx* extension feedback cables are UL Listed, bulk cable, type CM.
2090-CFBM7E7-CEAF*xx* extension feedback cables are UL Listed, bulk cable, type PLTC-ER.

Power and Feedback Transition Cable Specifications

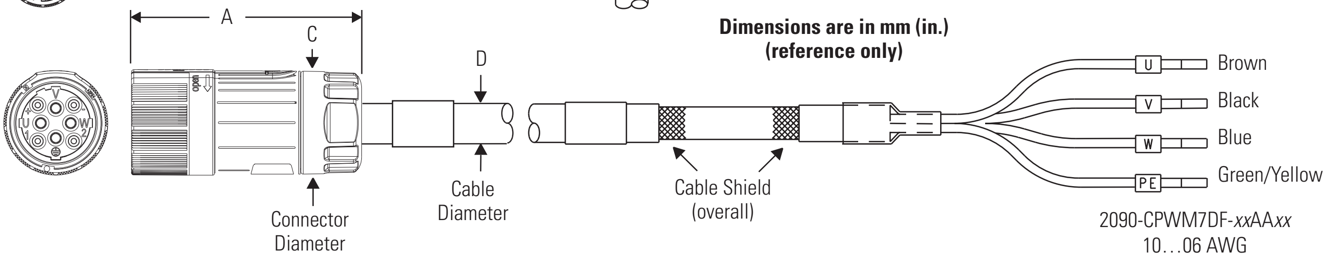
Transition Cable Cat. No.	Cable Type/ Jacket Color	Description	Standard Cable Lengths mm (in.)
2090-CPBM4E2-14TR	Power with brake Industrial TPE, Black	Threaded DIN connector on motor end to bayonet receptacle for mating with existing bayonet cable, 600V.	500 (19.7)
2090-CPBM4E2-10TR			
2090-CPBM4E2-08TR			
2090-CPBM4E2-04TR			
2090-CPWM4E2-14TR	Power (only) Industrial TPE, Black		
2090-CPWM4E2-10TR			
2090-CPWM4E2-08TR			
2090-CPWM4E2-04TR			
2090-CFBM4E2-CATR	Feedback Industrial TPE, Black	Threaded DIN connector on motor end to bayonet receptacle for mating with existing bayonet cable, 300V.	

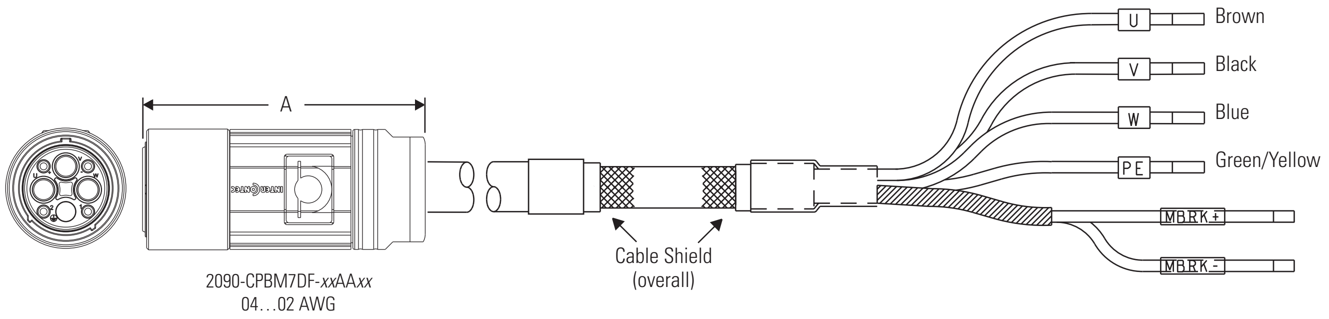
Motor Power Cable Dimensions

Power Cable Dimensions, Standard (SpeedTec DIN connector)



Dimensions are in mm (in.) (reference only)



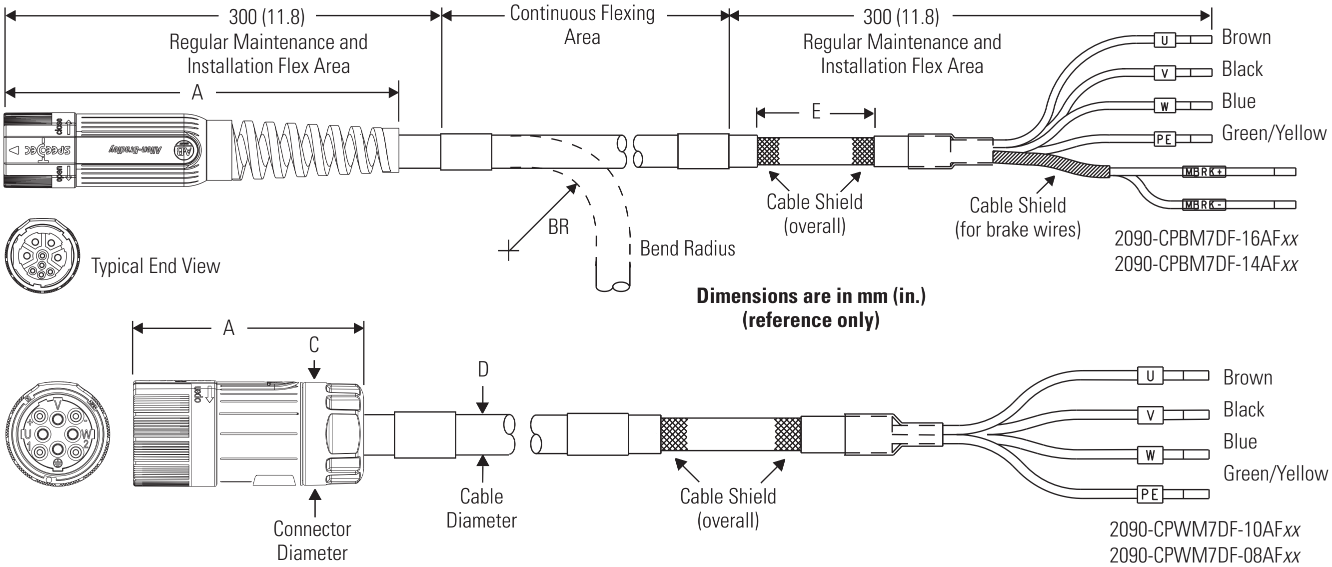


Power Cable Dimensions (standard)

Power Cable Cat. No.	A mm (in.)	BR [1] mm (in.)	C mm (in.)	D mm (in.)	E mm (in.)
2090-CPBM7DF-16AA*xx*	147 (5.8)	115 (4.5)	28 (1.1)	11.6 (0.46)	150 (5.9)
2090-CPWM7DF-16AA*xx*		95 (3.7)		9.2 (0.36)	
2090-CPBM7DF-14AA*xx*		130 (5.1)		12.7 (0.50)	
2090-CPWM7DF-14AA*xx*		105 (4.1)		10.3 (0.40)	
2090-CPBM7DF-12AA*xx*	80 (3.15)	140 (5.5)		14.3 (0.56)	
2090-CPWM7DF-12AA*xx*		115 (4.5)		11.2 (0.44)	
2090-CPBM7DF-10AA*xx*	100 (3.9)	170 (6.7)	45 (1.8)	16.8 (0.66)	90 (3.5)
2090-CPWM7DF-10AA*xx*		155 (6.1)		15.3 (0.60)	
2090-CPBM7DF-08AA*xx*		205 (8.0)		20.1 (0.79)	
2090-CPWM7DF-08AA*xx*		190 (7.5)		18.7 (0.74)	
2090-CPBM7DF-06AA*xx*		250 (9.8)		24.3 (0.96)	
2090-CPBM7DF-04AA*xx*	150 (5.9)	290 (11.4)	63 (2.5)	28.8 (1.13)	
2090-CPBM7DF-02AA*xx*		330 (13.0)		32.7 (1.29)	

(1) Standard cables have a regular maintenance and installation bend radius of 10 times (10x) the cable diameter.

Power Cable Dimensions, Continuous-flex (SpeedTec DIN connector)



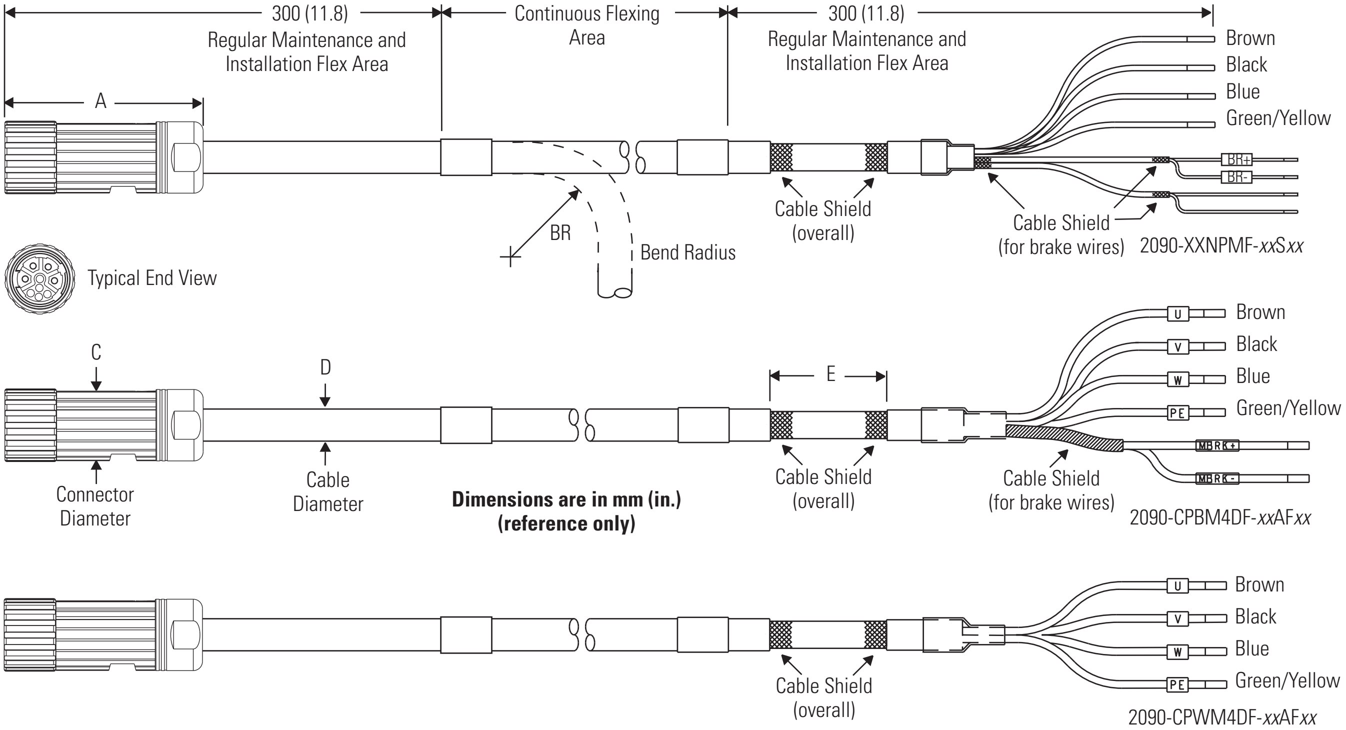
Dimensions are in mm (in.) (reference only)

Power Cable Dimensions (continuous-flex rated)

Power Cable Cat. No.	**A** mm (in.)	**BR** [1] mm (in.)	**C** mm (in.)	**D** mm (in.)	**E** mm (in.)
2090-CPBM7DF-16AF*xx*	147 (5.8)	150 (6.0)	28.0 (1.1)	12.5 (0.49)	150 (5.9)
2090-CPWM7DF-16AF*xx*		120 (5.0)		9.7 (0.38)	
2090-CPBM7DF-14AF*xx*		165 (6.5)		13.7 (0.54)	
2090-CPWM7DF-14AF*xx*		125 (5.0)		10.4 (0.41)	
2090-CPBM7DF-10AF*xx*	100 (3.9)	187 (7.4)	45.0 (1.8)	17.8 (0.70)	90 (3.5)
2090-CPWM7DF-10AF*xx*				15.7 (0.62)	
2090-CPBM7DF-08AF*xx*		250 (9.8)		20.6 (0.81)	
2090-CPWM7DF-08AF*xx*		242 (9.5)		20.2 (0.79)	

(1) Continuous-flex cables have an operational bend radius of 12 times (12x) the cable diameter. Secure the installation area, approximately 300 mm (12 in.) at both ends of the cable, with a rigid mount that prevents the cable from flexing where it connects to other components.

Power Cable Dimensions (threaded DIN connector)



Power Cable Dimensions (standard)

Power Cable Cat. No.	A mm (in.)	BR [1] mm (in.)	C mm (in.)	D mm (in.)	E mm (in.)
2090-XXNPMF-16S*xx*	75 (2.9)	142 (5.6)	28.0 (1.1)	14 (0.55)	150 (5.9)
2090-XXNPMF-14S*xx*		148 (5.8)		15 (0.59)	90.0 (3.5)
2090-XXNPMF-10S*xx*	96 (3.8)	187 (7.4)	45.0 (1.8)	19 (0.75)	

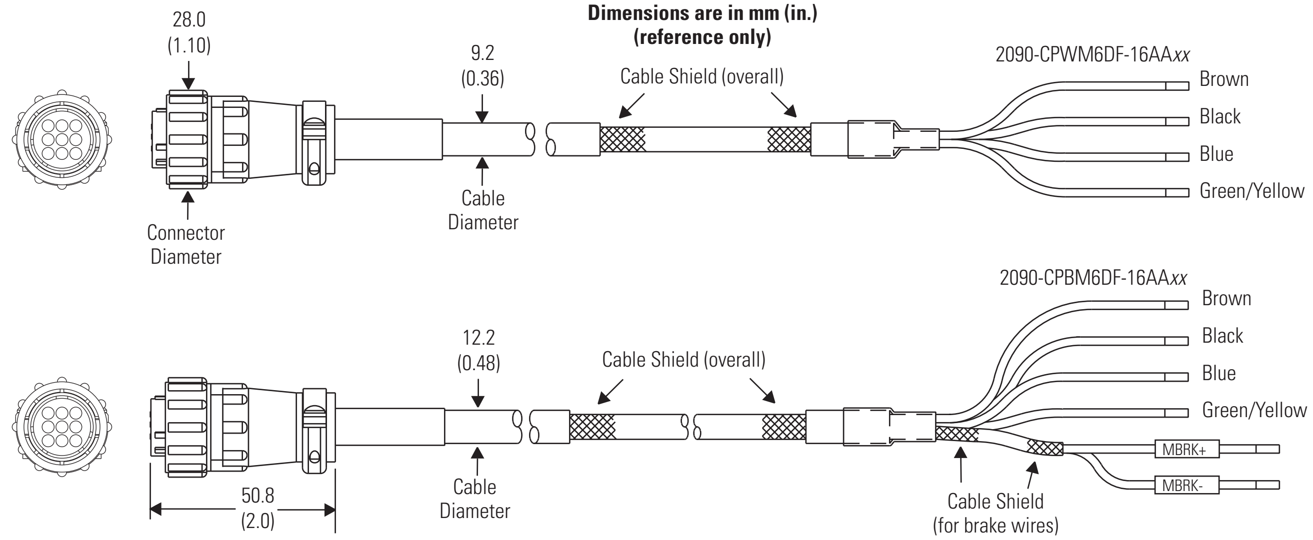
(1) Standard cables have a regular maintenance and installation bend radius of 10 times (10x) the cable diameter.

Power Cable Dimensions (continuous-flex rated)

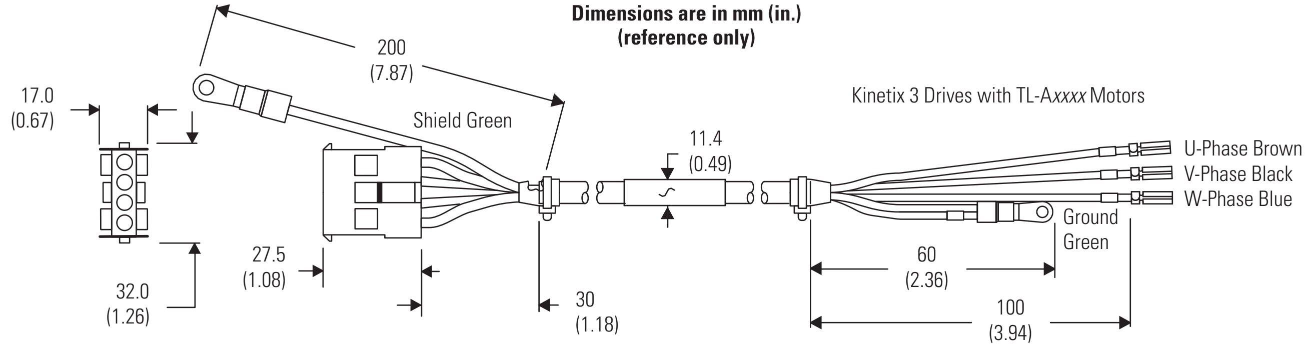
Power Cable Cat. No.	A mm (in.)	BR [1] mm (in.)	C mm (in.)	D mm (in.)	E mm (in.)
2090-CPBM4DF-16AF*xx*	75 (2.9)	150 (6.0)	28.0 (1.1)	12.5 (0.49)	150 (5.9)
2090-CPWM4DF-16AF*xx*		120 (5.0)		9.7 (0.38)	

(1) Continuous-flex cables have an operational bend radius of 12 times (12x) the cable diameter. Secure the installation area, approximately 300 mm (12 in.) at both ends of the cable, with a rigid mount that prevents the cable from flexing where it connects to other components.

Power Cable Dimensions (catalog number 2090-CP*x*M6DF-16AA*xx*)

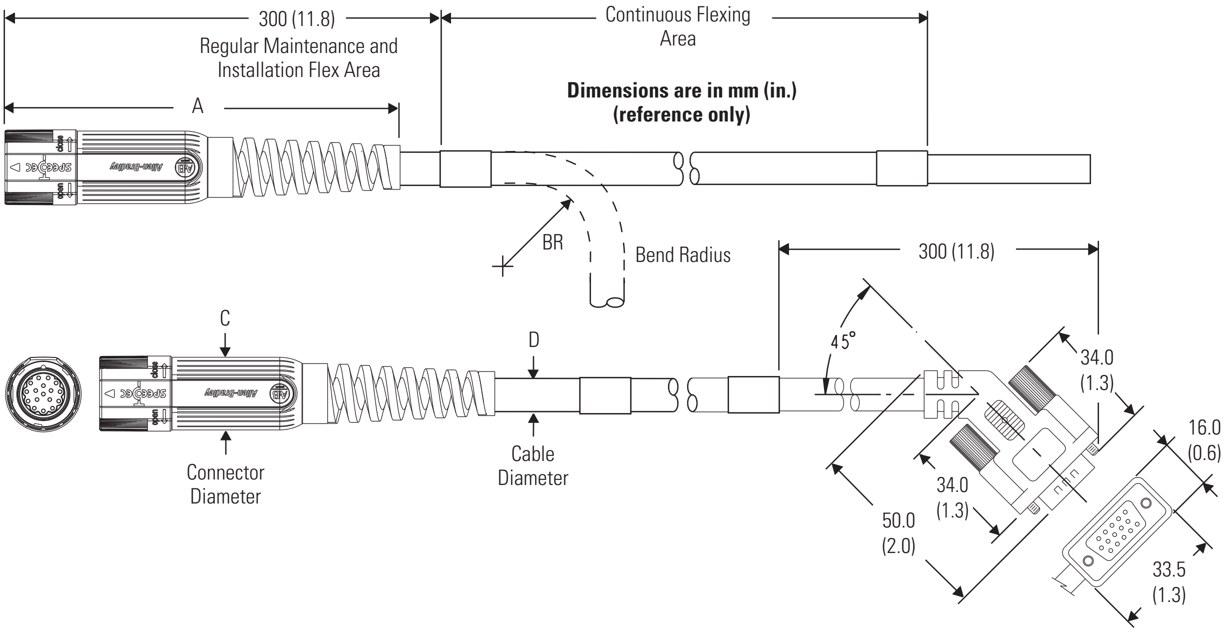


Power Cable Dimensions (catalog number 2090-DANPT-16S*xx*)



Motor Feedback Cable Dimensions

Feedback Cable Dimensions (SpeedTec DIN connector)



Feedback Cable Dimensions (standard)

Feedback Cable Cat. No.	**A** mm (in.)	**BR** [1] mm (in.)	**C** mm (in.)	**D** mm (in.)
2090-CFBM7DF-CEAA*xx*	147 (5.8)	100 (4.0)	28.0 (1.1)	9.8 (0.38)
2090-CFBM7DD-CEAA*xx*				

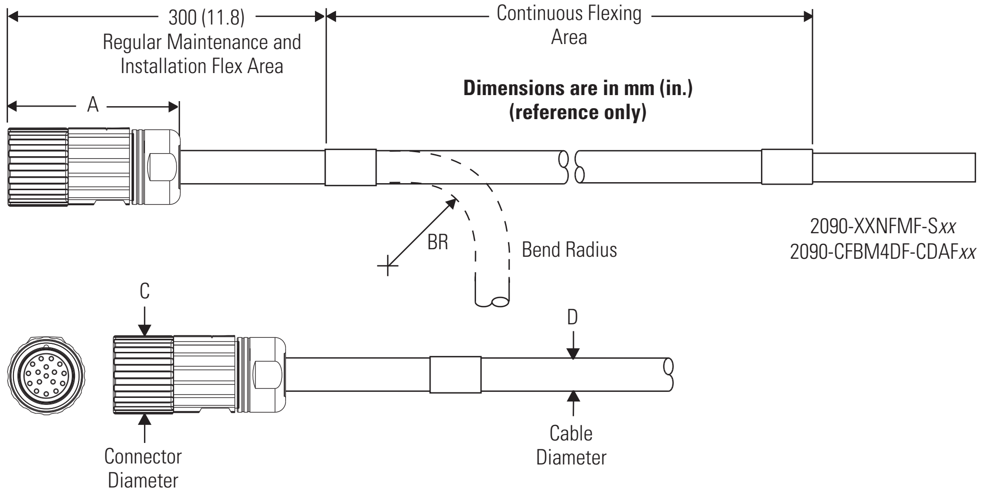
[1] Standard cables have a regular maintenance and installation bend radius of 10 times (10x) the cable diameter.

Feedback Cable Dimensions (continuous-flex rated)

Feedback Cable Cat. No.	**A** mm (in.)	**BR** [1] mm (in.)	**C** mm (in.)	**D** mm (in.)
2090-CFBM7DF-CEAF*xx*	147 (5.8)	125 (4.9)	28.0 (1.1)	10.3 (0.40)
2090-CFBM7DD-CEAF*xx*				
2090-CFBM7DF-CDAF*xx*		140 (5.5)		11.7 (0.46)

[1] Continuous-flex cables have an operational bend radius of 12 times (12x) the cable diameter. Secure the installation area, approximately 300 mm (12 in.) at both ends of the cable, with a rigid mount that prevents the cable from flexing where it connects to other components.

Feedback Cable Dimensions (threaded DIN connector)



Feedback Cable Dimensions (standard)

Feedback Cable Cat. No.	**A** mm (in.)	**BR** [(1)] mm (in.)	**C** mm (in.)	**D** mm (in.)
2090-XXNFMF-S*xx*	57.0 (2.2)	100 (4.0)	26.0 (1.0)	10.0 (0.40)

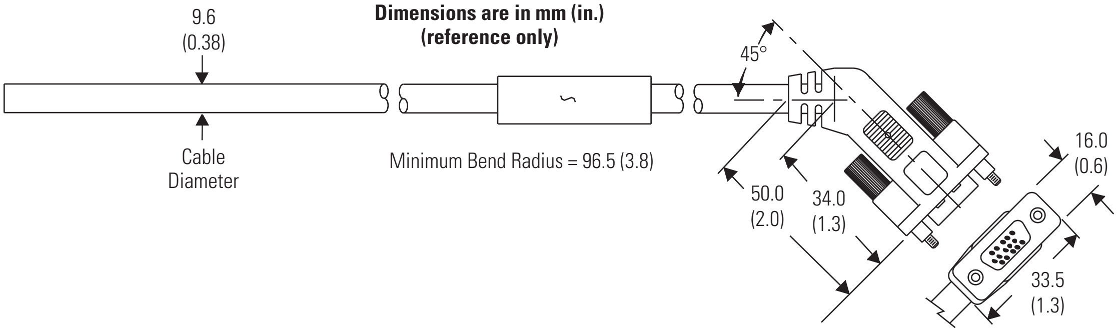
(1) Standard cables have a regular maintenance and installation bend radius of 10 times (10x) the cable diameter.

Feedback Cable Dimensions (continuous-flex rated)

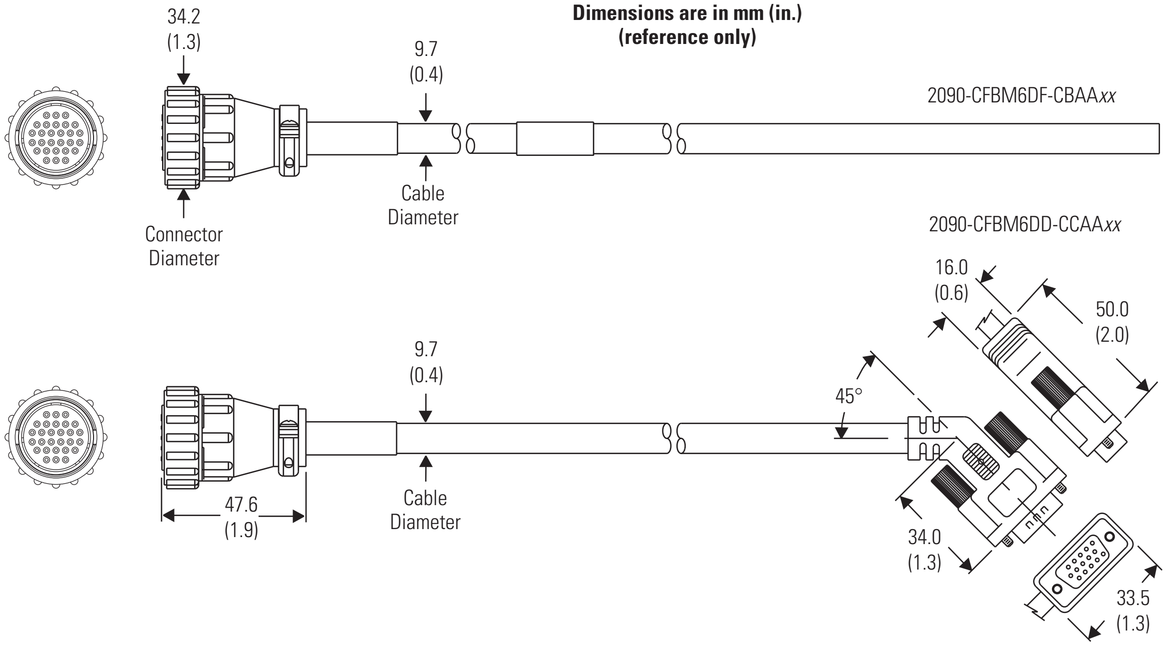
Feedback Cable Cat. No.	**A** mm (in.)	**BR** [(1)] mm (in.)	**C** mm (in.)	**D** mm (in.)
2090-CFBM4DF-CDAF*xx*	57.0 (2.2)	140 (5.5)	26.0 (1.0)	11.7 (0.46)

(1) Continuous-flex cables have an operational bend radius of 12 times (12x) the cable diameter. Secure the installation area, approximately 300 mm (12 in.) at both ends of the cable, with a rigid mount that prevents the cable from flexing where it connects to other components.

Feedback Cable Dimensions (catalog number 2090-UXNFM-S*xx*)

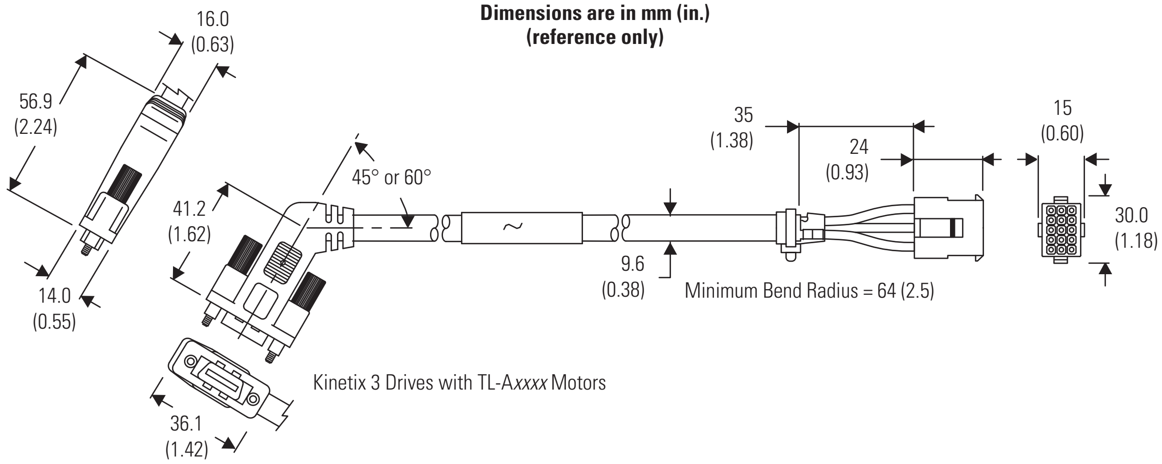


Feedback Cable Dimensions (catalog numbers 2090-CFBM6DF-CBAA*xx* and 2090-CFBM6DD-CCAA*xx*)



Dimensions are in mm (in.)
(reference only)

Feedback Cable Dimensions (catalog number 2090-DANFCT-S*xx*)



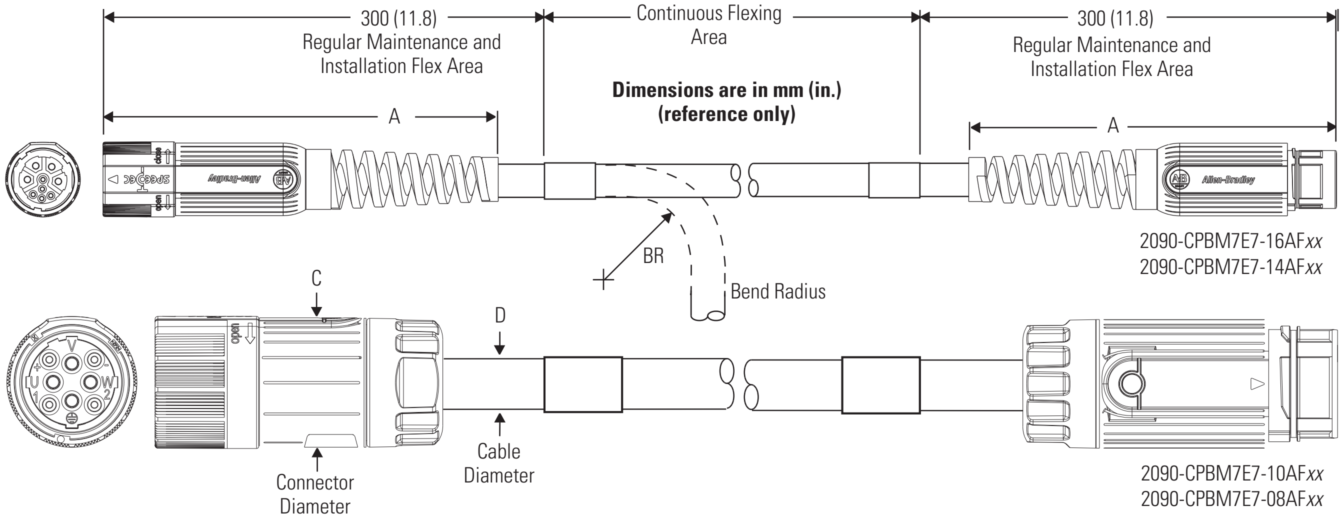
Dimensions are in mm (in.)
(reference only)

Minimum Bend Radius = 64 (2.5)

Kinetix 3 Drives with TL-A*xxxx* Motors

Extension Cable Dimensions

Power Cable Dimensions (SpeedTec DIN)

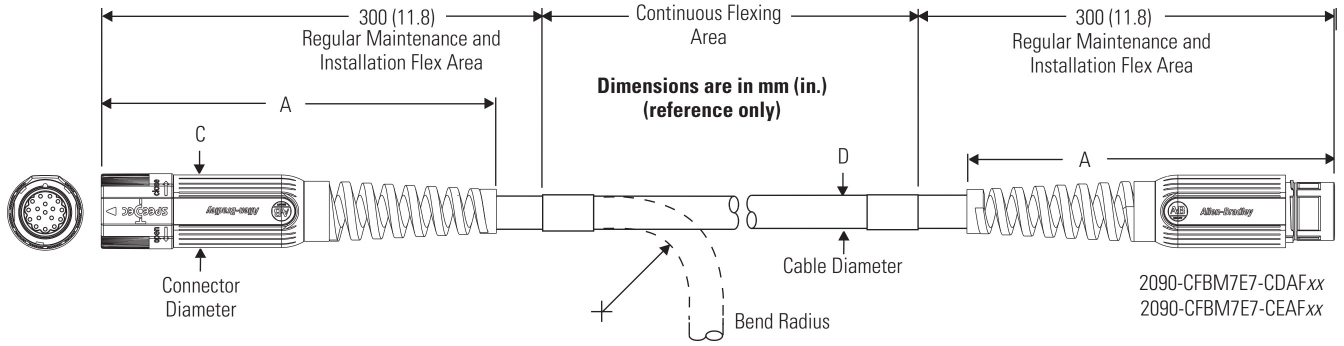


Power Cable Dimensions (continuous-flex rated)

Power Cable Cat. No.	A mm (in.)	BR [1] mm (in.)	C mm (in.)	D mm (in.)
2090-CPBM7E7-16AF*xx*	147 (5.8)	150 (5.9)	28.0 (1.1)	12.5 (0.49)
2090-CPBM7E7-14AF*xx*		165 (6.5)		13.7 (0.54)
2090-CPBM7E7-10AF*xx*	97 (3.8)	215 (8.5)	45.0 (1.8)	17.8 (0.70)
2090-CPBM7E7-08AF*xx*		250 (9.8)		20.6 (0.81)

(1) Continuous-flex cables have an operational bend radius of 12 times (12x) the cable diameter. Secure the installation area, approximately 300 mm (12 in.) at both ends of the cable, with a rigid mount that prevents the cable from flexing where it connects to other components.

Feedback Cable Dimensions (SpeedTec DIN)



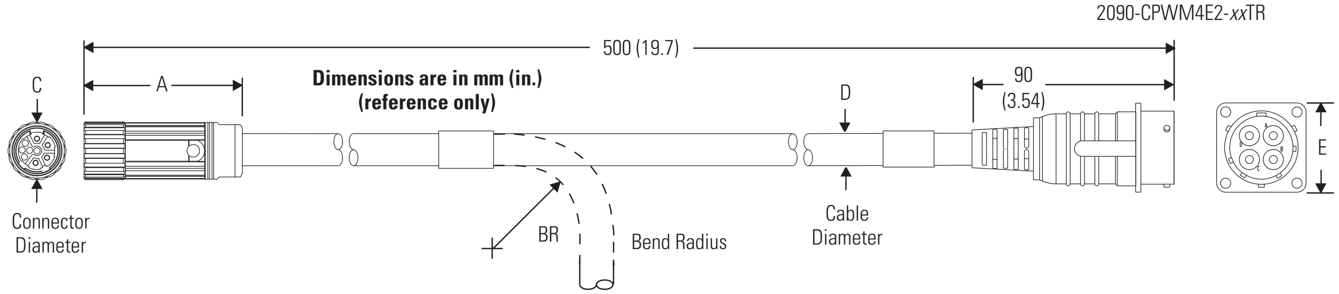
Feedback Cable Dimensions (continuous-flex rated)

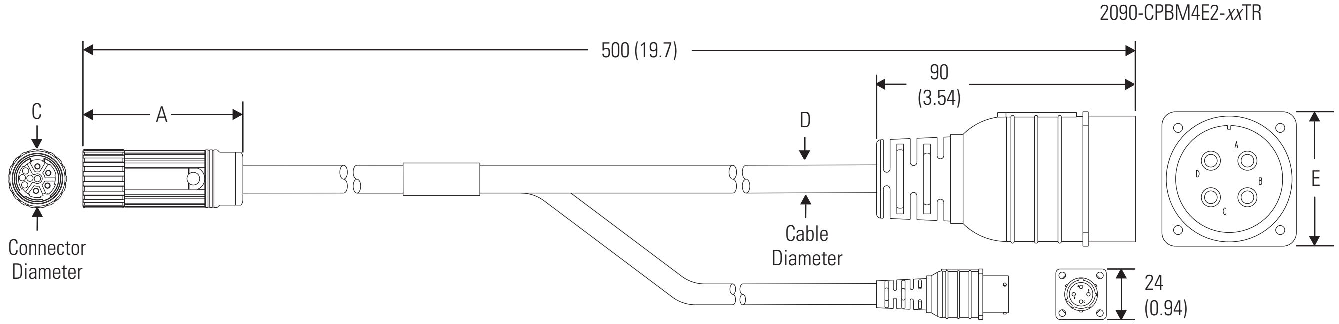
Feedback Cable Cat. No.	A mm (in.)	BR [1] mm (in.)	C mm (in.)	D mm (in.)
2090-CFBM7E7-CDAF*xx*	147 (5.8)	140 (5.5)	28.0 (1.1)	11.7 (0.46)
2090-CFBM7E7-CEAF*xx*		125 (4.9)		10.3 (0.40)

(1) Continuous-flex cables have an operational bend radius of 12 times (12x) the cable diameter. Secure the installation area, approximately 300 mm (12 in.) at both ends of the cable, with a rigid mount that prevents the cable from flexing where it connects to other components.

Transition Cable Dimensions

Power Cable Dimensions



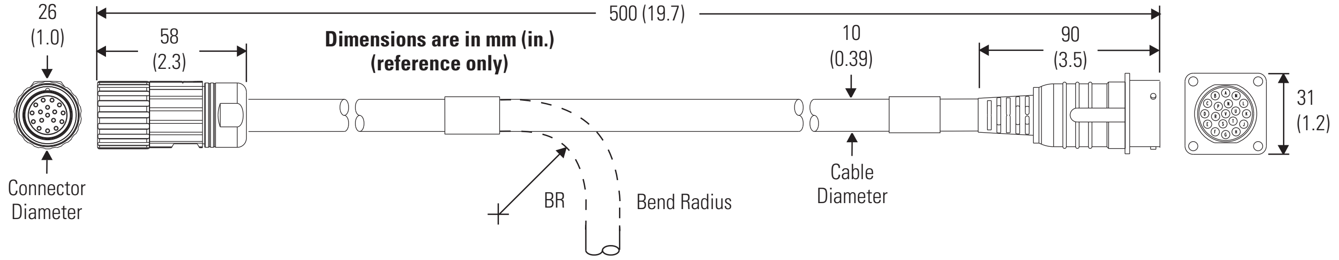


Power Cable Dimensions (standard)

Power Cable Cat. No.	A mm (in.)	BR [1] mm (in.)	C mm (in.)	D mm (in.)	E mm (in.)
2090-CPBM4E2-14TR	80 (3.15)	148 (5.83)	28.0 (1.10)	14.8 (0.58)	52.3 (2.06)
2090-CPWM4E2-14TR		104 (4.09)		10.4 (0.41)	31.0 (1.22)
2090-CPBM4E2-10TR	80 (3.15)	187 (7.36)	45.0 (1.77)	18.7 (0.74)	46.0 (1.81)
2090-CPWM4E2-10TR	95 (3.74)	156 (6.14)		15.6 (0.61)	31.0 (1.22)
2090-CPBM4E2-08TR	80 (3.15)	205 (8.07)		20.5 (0.81)	46.0 (1.81)
2090-CPWM4E2-08TR	95 (3.74)	189 (7.44)		18.9 (0.74)	35.0 (1.38)
2090-CPBM4E2-04TR	80 (3.15)	287 (11.30)	63.4 (2.48)	28.7 (1.13)	52.3 (2.06)
2090-CPWM4E2-04TR	95 (3.74)				

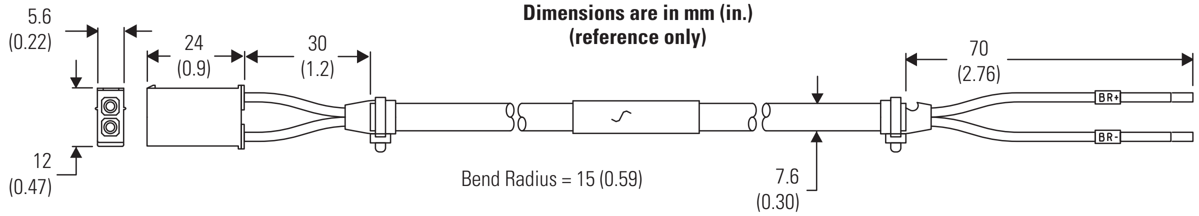
(1) Standard cables have a regular maintenance and installation bend radius of 10 times (10x) the cable diameter.

Feedback Cable Dimensions (catalog number 2090-CFBM4E2-CATR)



Motor Brake Cable Dimensions

Brake Cable Dimensions (catalog number 2090-DANBT-18S*xx*)



2090-Series Motor-end Cable Connector Kits

Motor-end connector kits are available for building your own cables. Kits are available for bayonet, circular DIN (M4 and M7), and circular plastic (M6) cable connectors.

Bayonet Motor-end Cable Connector Kits

Motor Series	Connector Kit Cat. No.	Description
MPL-A/B3*xxx-xx*2*x*AA, MPL-A/B4*xxx-xx*2*x*AA, MPL-A/B45*xxx-xx*2*x*AA MPL-A520K-*xx*2*x*AA MPL-B520K-*xx*2*x*AA, MPL-B540K-*xx*2*x*AA, MPL-B540D-*xx*2*x*AA, MPL-B560F-*xx*2*x*AA, and MPL-B580F-*xx*2*x*AA	2090-MPPC-S	Straight Power Connector Kit, 12 AWG max
MPL-B6*xxx-xx*2*x*AA, MPL-B8*xxx-xx*2*x*AA, MPL-B960B-*xx*2*x*AA, MPL-B960C-*xx*2*x*AA, MPL-B980B-*xx*2*x*AA, MPL-B980C-*xx*2*x*AA	2090-MPPC-08S	Straight Power Connector Kit, 8 AWG max
MPL-A/B3*xxx-xx*2*x*AA, MPL-A/B4*xxx-xx*2*x*AA, MPL-A/B45*xxx-xx*2*x*AA, MPL-A/B5*xxx-xx*2*x*AA MPL-B6*xxx-xx*2*x*AA, MPL-B8*xxx-xx*2*x*AA, MPL-B9*xxx-xx*2*x*AA	2090-MPFC-S	Straight Feedback Connector Kit
All MPL-A/B*xxx-xx*2*x*AA	2090-MPBC-S	Straight Brake Connector Kit

Circular DIN Motor-end Cable Connector Kits

Feedback Cable Connector Kits

M23 Cable Plug
Connector Kits

2090-KFBM4-CAAA
2090-KFBM7-CAAA
Motor-end Connector Kits

Connector pins and sockets in
these kits are soldered.

M23 Cable Extension
Connector Kits

2090-KFBE7-CAAA
Motor-end Connector Kits

Power Cable Connector Kits

M23 or M40 Cable Plug
Connector Kits

2090-KPBM4-12AA
2090-KPBM4-06AA
2090-KPBM7-12AA
2090-KPBM7-06AA
Motor-end Connector Kits

Connector pins and sockets in
these kits are crimped.

M23 or M40 Cable Extension
Connector Kits

2090-KPBE7-12AA
2090-KPBE7-06AA
Motor-end Connector Kits

Motor-end Connector Kit Cross-reference Tables

The tables beginning on page 424 provide a cross-reference for the circular DIN (M4 and M7) connector kits above to the compatible motor series catalog number. Also provided are the bulkhead adapters for securing the cables as they pass through the cabinet and crimping tools required for properly attaching the power wires to sockets and pins.

Connector kits and crimping tools are also available for circular plastic (M6) connectors. Refer to page 425 for the compatible motor series and crimp tool catalog numbers.

Power Cable Connector Kits (SpeedTec DIN)

Connector Kit Cat. No.	Description	Crimp Tool Cat. No.	Bulkhead Adapter Cat. No.	Motor Series
2090-KPBM7-12AA	Motor-end cable connector SpeedTec plug, M23 connector 16, 14, and 12 AWG motor power 18 AWG motor brake	2090-TCR47-M23	2090-KPB47-12CF	MPL-A/B3*xx*, MPL-A/B4*xx*, MPL-A/B45*xx*, MPL-B520, MPL-B540, MPL-B560
				MPM-A/B115*xx*, MPM-A/B130*xx*, MPM-B1651C, MPM-B1651F, MPM-B1652C, MPM-B1653C
				MPF-A/B3*xx*, MPF-A/B4*xx*, MPF-A/B45*xx*
				RDB-B130*xx*, RDB-B165*xx*, RDB-B21519, RDB-B2151C, RDB-B21529, RDB-B2152C, RDB-B21539, RDB-B2901*x*, RDB-B29024, RDB-B29026, RDB-B29034
2090-KPBM7-06AA	Motor-end cable connector SpeedTec plug, M40 connector 10, 8, and 6 AWG motor power 18 AWG motor brake	2090-TCR47-M40 (power pins) 2090-TCR47-M23 (brake pins)	2090-KPB47-06CF	MPL-A5*xx*, MPL-B580, MPL-B6*xx*, MPL-B860, MPL-B880C, MPL-B960B, MPL-B980B
				MPM-A1651F, MPM-B1651M, MPM-B1652E, MPM-A/B1652F, MPM-B1653E, MPM-A/B1653F, MPM-A/B215*xx*
				MPF-A/B5*xx*
				RDB-B2151F, RDB-B2152F, RDB-B2153C, RDB-B2153E, RDB-B29029, RDB-B29036, RDB-B410*xx*
2090-KPBE7-12AA	Extension cable connector SpeedTec plug, M23 connector 16, 14, and 12 AWG motor power 18 AWG motor brake	2090-TCR47-M23	2090-KPB47-12CF	MPL-A/B3*xx*, MPL-A/B4*xx*, MPL-A/B45*xx*, MPL-B520, MPL-B540, MPL-B560
				MPM-A/B115*xx*, MPM-A/B130*xx*, MPM-B1651C, MPM-B1651F, MPM-B1652C, MPM-B1653C
				MPF-A/B3*xx*, MPF-A/B4*xx*, MPF-A/B45*xx*
				RDB-B130*xx*, RDB-B165*xx*, RDB-B21519, RDB-B2151C, RDB-B21529, RDB-B2152C, RDB-B21539, RDB-B2901*x*, RDB-B29024, RDB-B29026, RDB-B29034
2090-KPBE7-06AA	Extension cable connector SpeedTec plug, M40 connector 10, 8, and 6 AWG motor power 18 AWG motor brake	2090-TCR47-M40 (power pins) 2090-TCR47-M23 (brake pins)	2090-KPB47-06CF	MPL-A5*xx*, MPL-B580, MPL-B6*xx*, MPL-B860, MPL-B880C, MPL-B960B, MPL-B980B
				MPM-A1651F, MPM-B1651M, MPM-B1652E, MPM-A/B1652F, MPM-B1653E, MPM-A/B1653F, MPM-A/B215*xx*
				MPF-A/B5*xx*
				RDB-B2151F, RDB-B2152F, RDB-B2153C, RDB-B2153E, RDB-B29029, RDB-B29036, RDB-B410*xx*

Power Cable Connector Kits (threaded DIN)

Connector Kit Cat. No.	Description	Crimp Tool Cat. No.	Bulkhead Adapter Cat. No.	Motor Series
2090-KPBM4-12AA	Motor-end cable connector Threaded plug, M23 connector 16, 14, and 12 AWG motor power 18 AWG motor brake	2090-TCR47-M23	2090-KPB47-12CF	MPL-A/B15*xx*, MPL-A/B2*xx*
				MPF-A/B3*xx*, MPF-A/B4*xx*, MPF-A/B45*xx*, MPF-A/B5*xx*
				MPS-A/B3*xx*, MPS-A/B45*xx*, MPS-B5*xx*
2090-KPBM4-06AA	Motor-end cable connector Threaded plug, M40 connector 10, 8, and 6 AWG motor power 18 AWG motor brake	2090-TCR47-M40 (power pins) 2090-TCR47-M23 (brake pins)	2090-KPB47-06CF	MPF-A/B5*xx*

Feedback Cable Connector Kits (circular DIN)

Connector Kit Cat. No.	Description	Crimp Tool Cat. No.	Bulkhead Adapter Cat. No.	Motor Series
2090-KFBM7-CAAA	Motor-end cable connector SpeedTec plug, M23 connector	N/A (soldered contacts)	2090-KFB47-CF	MPL-A/B3*xx*, MPL-A/B4*xx*, MPL-A/B45*xx*, MPL-A/B5*xx* MPL-B6*xx*, MPL-B8*xx*, MPL-B9*xx* MPM-A/B115*xx*, MPM-A/B130*xx*, MPM-A/B165*xx*, MPM-A/B215*xx* MPF-A/B3*xx*, MPF-A/B4*xx*, MPF-A/B45*xx*, MPF-A/B5*xx* RDB-B130*xx*, RDB-B165*xx*, RDB-B215*xx*, RDB-B290*xx*, RDB-B410*xx*,
2090-KFBE7-CAAA	Extension cable connector SpeedTec plug, M23 connector			MPL-A/B3*xx*, MPL-A/B4*xx*, MPL-A/B45*xx*, MPL-A/B5*xx*, MPL-B6*xx*, MPL-B8*xx*, MPL-B9*xx*
				MPM-A/B115*xx*, MPM-A/B130*xx*, MPM-A/B165*xx*, MPM-A/B215*xx*
				MPF-A/B3*xx*, MPF-A/B4*xx*, MPF-A/B45*xx*, MPF-A/B5*xx*
				RDB-B130*xx*, RDB-B165*xx*, RDB-B215*xx*, RDB-B290*xx*, RDB-B410*xx*
2090-KFBM4-CAAA	Motor-end cable connector Threaded plug, M23 connector			MPL-A/B15*xx* or MPL-A/B2*xx* MPF-A/B3*xx*, MPF-A/B4*xx*, MPF-A/B45*xx*, MPF-A/B5*xx* MPS-A/B3*xx*, MPS-A/B45*xx*, MPS-A/B5*xx*

Power and Feedback Cable Connector Kits (circular plastic)

Connector Kit Cat. No.	Description	Crimp Tool Cat. No.	Bulkhead Adapter Cat. No.	Motor Series
2090-KPBM6-16AA	Straight Power Connector Kit	58495-1 (Tyco AMP)	N/A	All TLY-A*xxx* motors All TLAR-A*xxx* electric cylinders
2090-KFBM6-AA	Straight Feedback Connector Kit	58448-1 (Tyco AMP)	N/A	

2090-Series Bulkhead Adapter Kits

These bulkhead adapter kits let you secure your cables as they pass through the cabinet and apply to circular DIN (M4 and M7) power and feedback cables.

Circular DIN Power Cable Compatibility

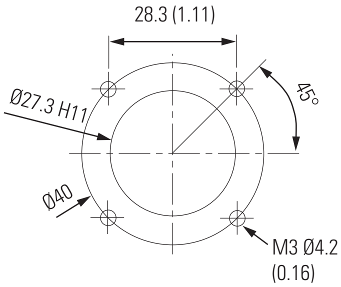
Bulkhead Adapter Cat. No.	Standard (non-flex) Power Cable Cat. No.	Continuous-flex Power Cable Cat. No.	Description		Connector Diameter mm (in.)
2090-KPB47-12CF	2090-CPWM7DF-16AA*xx*	2090-CPWM7DF-16AF*xx*	Power only	SpeedTec DIN	28.0 (1.1)
	2090-CPWM7DF-14AA*xx*	2090-CPWM7DF-14AF*xx*			
	2090-CPWM7DF-12AA*xx*	N/A			
	2090-CPBM7DF-16AA*xx*	2090-CPBM7DF-16AF*xx*	Power/brake		
	2090-CPBM7DF-14AA*xx*	2090-CPBM7DF-14AF*xx*			
	2090-CPBM7DF-12AA*xx*	N/A			
	N/A	2090-CPBM7E7-16AA*xx*			
	N/A	2090-CPBM7E7-14AA*xx*			
	N/A	2090-CPWM4DF-16AF*xx*	Power only	Threaded DIN	
	N/A	2090-CPBM4DF-16AF*xx*	Power/brake		
	2090-XXNPMF-16S*xx*	N/A			
2090-KPB47-06CF	2090-CPWM7DF-10AA*xx*	2090-CPWM7DF-10AF*xx*	Power only	SpeedTec DIN	45.0 (1.8)
	2090-CPWM7DF-08AA*xx*	2090-CPWM7DF-08AF*xx*			
	2090-CPBM7DF-10AA*xx*	2090-CPBM7DF-10AF*xx*	Power/brake		
	2090-CPBM7DF-08AA*xx*	2090-CPBM7DF-08AF*xx*			
	2090-CPBM7DF-06AA*xx*	N/A			

Circular DIN Feedback Cable Compatibility

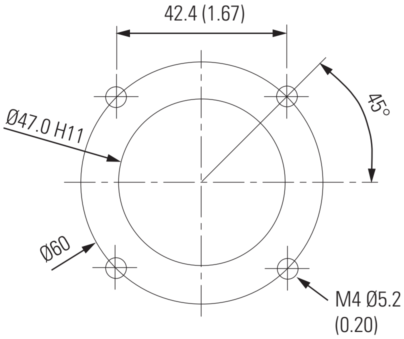
Bulkhead Adapter Cat. No.	Standard (non-flex) Feedback Cable Cat. No.	Continuous-flex Feedback Cable Cat. No.	Description		Connector Diameter mm (in.)
2090-KPB47-12CF	2090-CFBM7DF-CEAA*xx*	2090-CFBM7DF-CEAF*xx*	Feedback	SpeedTec DIN	28.0 (1.1)
	2090-CFBM7DD-CEAA*xx*	2090-CFBM7DD-CEAF*xx*			
	N/A	2090-CFBM7DF-CDAF*xx*			
	N/A	2090-CFBM7E7-CDAF*xx*			
	N/A	2090-CFBM7E7-CEAF*xx*			
2090-KFB47-CF	2090-CFBM4DF-CEAA*xx*	N/A		Threaded DIN	26.0 (1.0)
	N/A	2090-CFBM4DF-CDAF*xx*			
	2090-XXNFMF-S*xx*	N/A			

Bulkhead Adapter Kit Mounting Dimensions

Dimensions are in mm (in.)



2090-KFB47-CF and 2090-KPB47-12CF
(M23) Bulkhead Adapter Kits

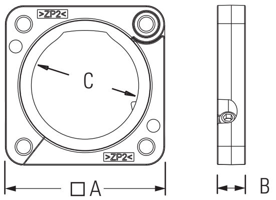


2090-KPB47-06CF
(M40) Bulkhead Adapter Kits

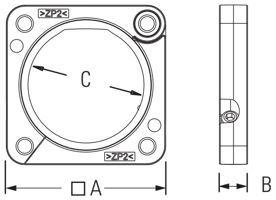
Bulkhead Adapter Kit Dimensions

Dimensions are in mm (in.)

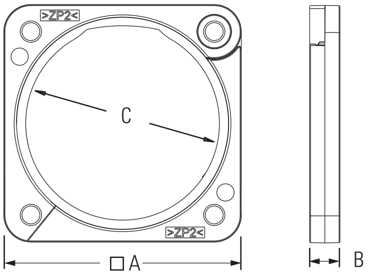
2090-KFB47-CF
Bulkhead Adapter



2090-KPB47-12CF
Bulkhead Adapter



2090-KPB47-06CF
Bulkhead Adapter



Bulkhead Adapter Cat. No.	**Size**	**Dimension A** mm (in.)	**Dimension B** mm (in.)	**Dimension C (reference)** mm (in.)
2090-KFB47-CF	M23	36.8 (1.44)	6.5 (0.26)	23.0 (0.90)
2090-KPB47-12CF				26.0 (1.02)
2090-KPB47-06CF	M40	54.8 (2.16)	7.0 (0.28)	43.0 (1.69)

2090-Series Motor/Actuator Cable Catalog Numbers

Motor Power/Brake, Feedback, and Extension Cables

2090 - C *xx* M*x* D*x* - C*x* A*x* *xx*

Cable Length
Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Cable Type
AA = Standard
AF = Continuous-flex

Encoder Type (applies to feedback cables)
CB = Serial incremental/Serial absolute - battery backup
CC = Serial incremental/Incremental
CD = SIN/COS High-resolution/Incremental
CE = SIN/COS High-resolution/Resolver

Wire Gauge Size (applies to power cables)
16, 14, 12, 10, 8, 6, 4, and 2 AWG

Drive-end Connector Type
DF = Drive-end, flying-lead
DD = Drive-end, D-sub connector
E7 = Extension receptacle (SpeedTec ready)

Motor-end Connector Type
M6 = Circular plastic connector
M4 = Threaded DIN connector
M7 = SpeedTec DIN connector

Cable Type
PB = Motor power with brake wires
PW = Motor power only
FB = Motor feedback only

Accessory Component
C = Cable

Bulletin Number

Transition Cables

Cat. No.	Cable Type	Description
2090-CPBM4E2-14TR	Power/brake	Threaded DIN connector on motor-end to bayonet receptacle for mating with existing bayonet cable, 500 mm (19.7 in.).
2090-CPBM4E2-10TR		
2090-CPBM4E2-08TR		
2090-CPBM4E2-04TR		
2090-CPWM4E2-14TR	Power (only)	
2090-CPWM4E2-10TR		
2090-CPWM4E2-08TR		
2090-CPWM4E2-04TR		
2090-CFBM4E2-CATR	Feedback	

Motor Power, Feedback, and Brake Cables

2090 - *xx x xx xx* - *xx* S *xx*

Cable Length
Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Motor Connector
S = Straight

Wire Gauge Size (AWG)
16 = Motor power cable
18 = Motor brake cables
Blank = Feedback cables

Motor/Actuator Series
MF = Threaded DIN Connectors
MPS-A/B*xxxx* (MPS-A/B*xxxx*-M/S)
HPK-B/E*xxxx* (HPK-B/E*xxxxx*-M/S)
MPAS-A/B*xxxx* (MPAS-A/B*xxxxx*-V/A) or MPMA-A/B*xxxx*
T = TLY-A*xxxx* (TLY-A*xxxx*-B/H)

Cable Type
P = Motor power
F = Motor feedback connector (flying-leads at drive)
FC = Motor feedback (connectors at both ends, TL-Series)
FM = Motor feedback (flying-leads to D-sub at drive)
B = Motor brake

Flex Option
N = Standard cable (non-flex)

Drive Family
DA = Kinetix 3 drives
XX = All other drives

Bulletin Number

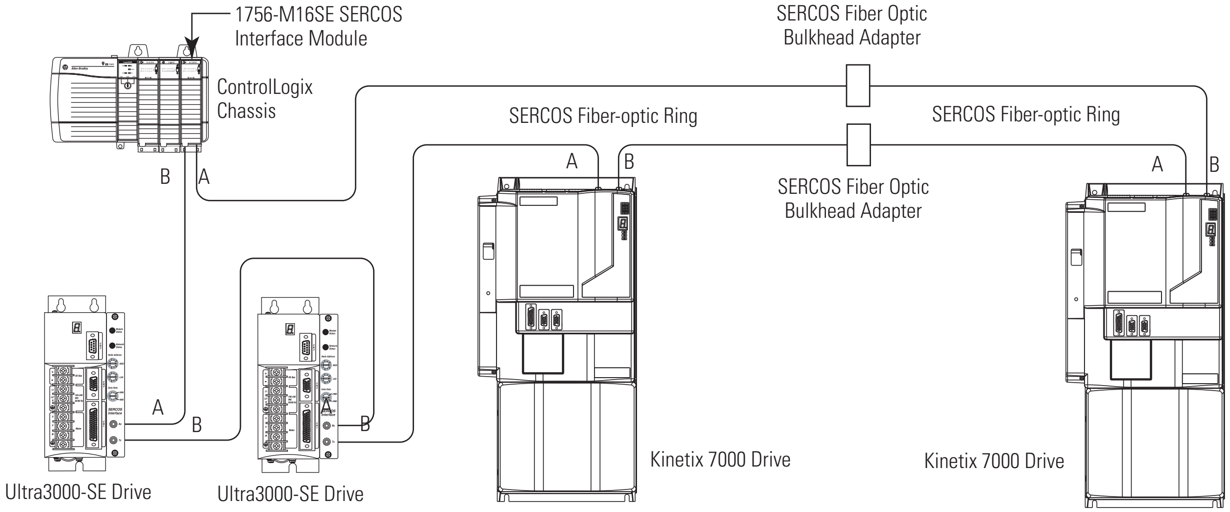
2090-Series Interface Cables

A wide variety of communication and interface cables are available for connecting servo drives to ControlLogix and CompactLogix controller modules, and to other Allen-Bradley products.

Fiber-optic Cable Connection Examples

The length of each transmission section (point A to B) can be up to 32 m (105 ft) for plastic cable and 50…200 m (164.2…656.7 ft) for glass cable. In this example, the second Kinetix 7000 drive is located in a separate cabinet and connected with bulkhead adapters.

Fiber-optic Cable Example for Single-axis Connections



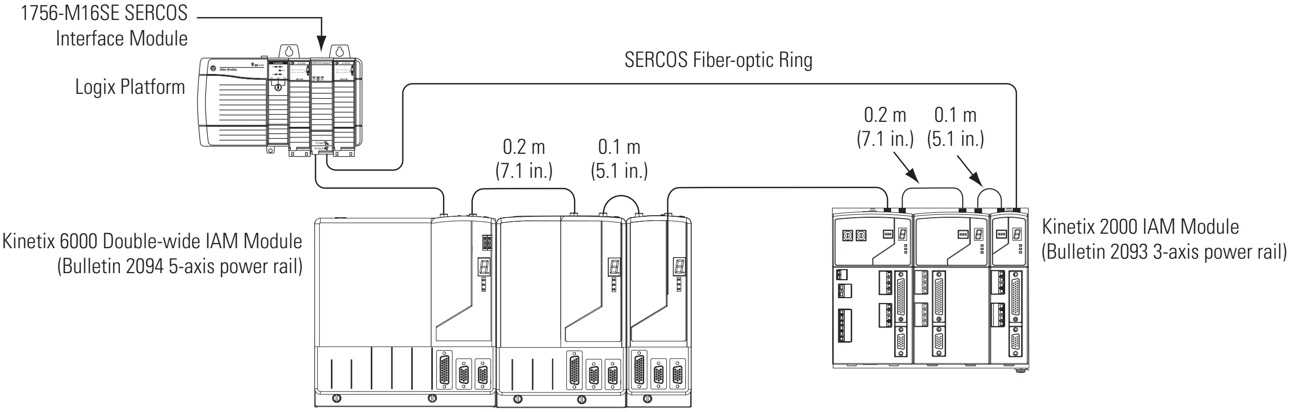
IMPORTANT To avoid signal loss, do not use bulkhead adapters to connect glass cables. Use bulkhead adapters for making plastic-to-plastic cable connections only.

Multi-axis servo drives with SERCOS interface have specific cable lengths for making drive-to-drive connections for single-wide and double-wide modules.

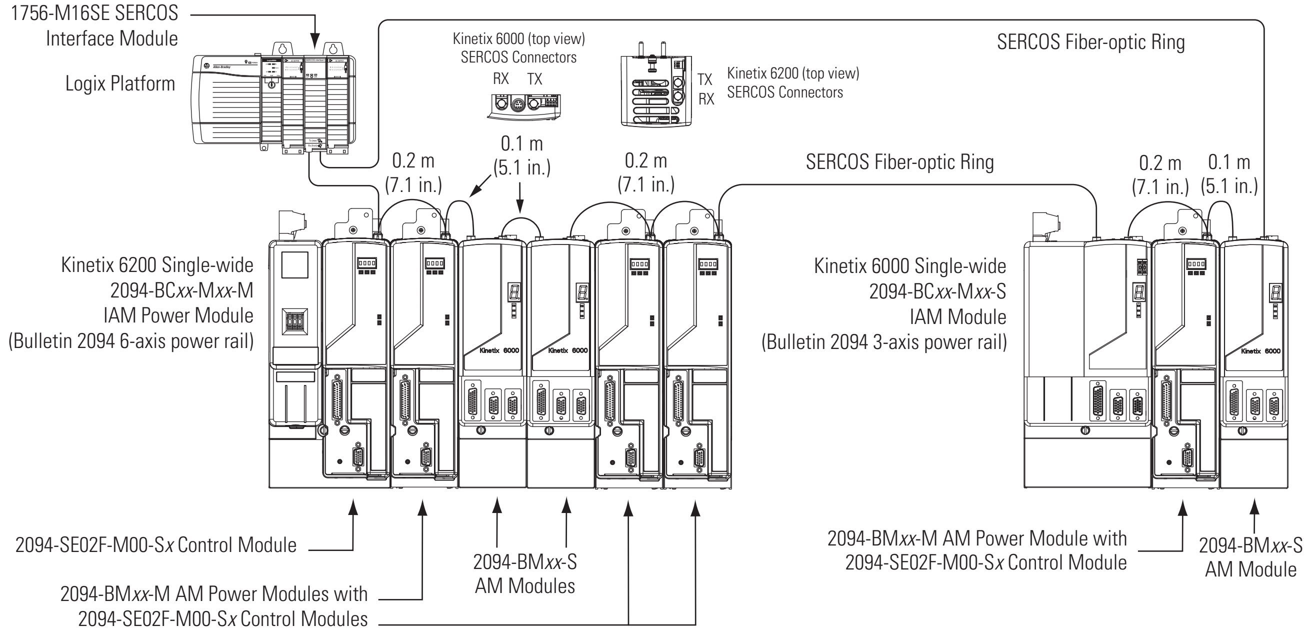
Drive-to-Drive Cable Length for Multi-axis Drive Families

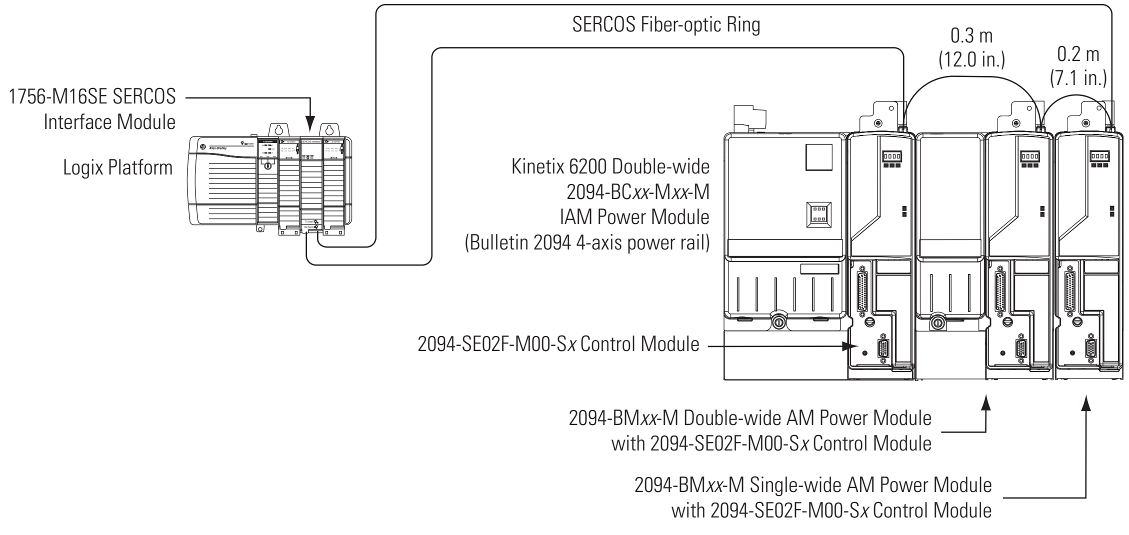
IAM Module	Adjacent AM Module		Cable Cat. No.	Cable Length m (in.)
Kinetix 2000	2093-AMP1, 2093-AMP2, or 2093-AMP5	Single-wide	2090-SCEP0-1	0.1 (5.1)
	2093-AM01 or 2093-AM02	Double-wide	2090-SCEP0-2	0.2 (7.1)
Kinetix 6000	2094-AM*xx*-S, 2094-BMP5-S, 2094-BM01-S, or 2094-BM02-S	Single-wide	2090-SCEP0-1	0.1 (5.1)
	2094-BMP5-M, 2094-BM01-M, or 2094-BM02-M		2090-SCEP0-2	0.2 (7.1)
	2094-BM03-S and 2094-BM05-S	Double-wide		
Kinetix 6200	2094-BMP5-M, 2094-BM01-M, or 2094-BM02-M	Single-wide	2090-SCEP0-2	0.2 (7.1)
	2094-AM*xx*-S, 2094-BMP5-S, 2094-BM01-S, or 2094-BM02-S		2090-SCEP0-1	0.1 (5.1)
	2094-BM03-M and 2094-BM05-M	Double-wide	2090-SCEP0-3	0.3 (12.0)

Drive-to-Drive Fiber-optic Cable Length Example (Kinetix 2000 and Kinetix 6000 drives)



Drive-to-Drive Fiber-optic Cable Length Example (Kinetix 6000 and Kinetix 6200 drives)

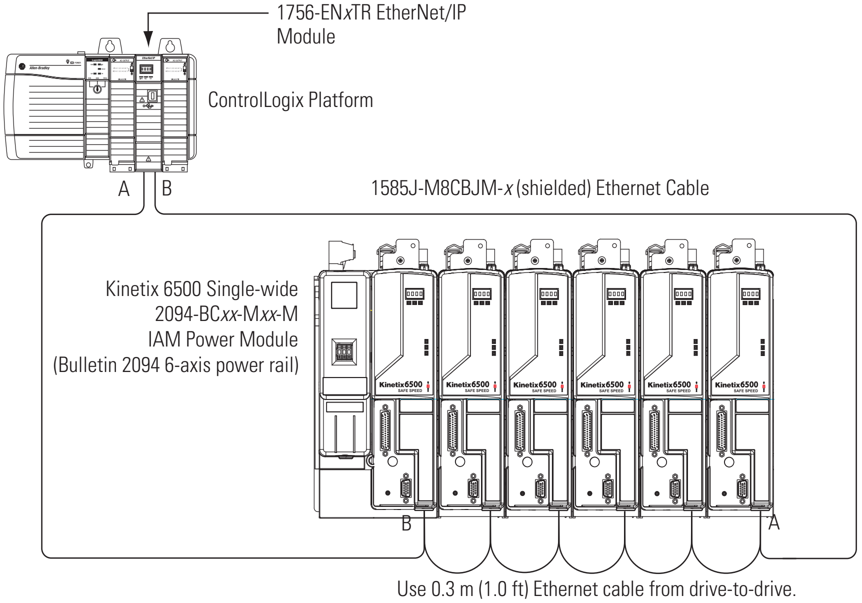




Ethernet Cable Connection Examples

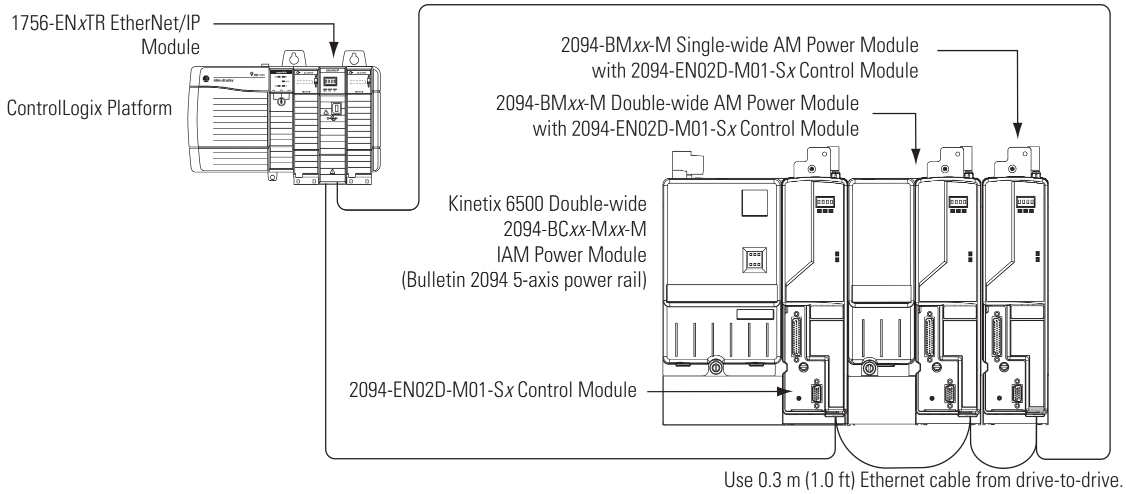
Shielded Ethernet cable is available in lengths up to 78 m (256 ft). However, the total length of Ethernet cable (point A to point B) connecting drive-to-drive, drive-to-controller, or drive-to-switch must not exceed 100 m (328 ft).

Drive-to-Drive Ethernet Cable Length Example (Kinetix 6500 drives)

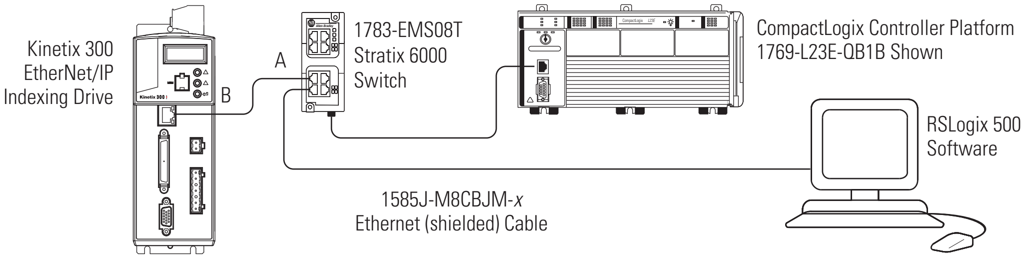


1585J-M8CBJM-*x*
Double-ended Ethernet Cable

1585J-M8CC-H
RJ45 Insulation Displacement
Connector (IDC)



Ethernet Cable Example for Single-axis Connections (Kinetix 300 drives)



Interface Cable Applications and Standard Lengths

Cat. No.	Descriptions		Standard Cable Lengths m (ft)
2090-UXPC-D09*xx*	Ultra3000/5000 serial interface cable to computer		1 (3.2) 3 (9.8) 9 (29.5) 15 (49.2) 30 (98.4)
2090-U3CC-D44*xx* (1)	Single-axis flying-lead cable, Ultra3000 drive to 1756-M02AE module		
2090-U3AE-D44*xx* (1)	Two-axis pre-wired cable, Ultra3000 drive to 1756-M02AE module		
2090-CCMDSDS-48AA*xx*	Kinetix 3 control interface cable for drive-to-drive configurations		1 (3.2) 0.3 (0.98)
2090-CCMPCDS-23AA*xx*	Kinetix 3 serial interface cable to personal computer		1 (3.2) 3 (9.8)
2090-CCMCNDS-48AA*xx*	Kinetix 3 control interface cable to MicroLogix controller		
2090-DAIO-D50*xxx*	Kinetix 3 control interface I/O cable (flying leads)		
2090-U5PM-D09*xx*	Ultra5000 drive to PanelView 300 Micro DF-1 terminal and MicroLogix system		1 (3.2) 3 (9.8) 9 (29.5)
2090-U5PV-D09*xx*	Ultra5000 drive to PanelView Standard DF-1 terminal		
2090-SCEP*x-x*	SERCOS fiber optic plastic cables suitable only for in-cabinet duty. Connectors are provided at both ends		0-1 (5.1 in.) (2) 0-2 (7.1 in.) (2) 0-3 (1.0) (2) 1-0 (3.2) 3-0 (9.8) 5-0 (16.4) 8-0 (26.2) 10-0 (32.8) 15-0 (49.2) 20-0 (65.5) 25-0 (82.0) 32-0 (105.0)
2090-SCNP*x-x*	SERCOS fiber optic plastic cables suitable for on-machine duty. Connectors are provided at both ends		
2090-SCVP*x-x*	SERCOS fiber optic plastic cables suitable for outdoor and conduit duty. Connectors are provided at both ends		
2090-SCVG*x-x*	SERCOS fiber optic glass cables suitable for outdoor and conduit duty. Connectors are provided at both ends		50-0 (164.2) 100-0 (328.3) 150-0 (492.5) 200-0 (656.7)
2090-S-BLHD	SERCOS fiber optic cable bulkhead adapter (2 per pack)		
1585J-M8CBJM-*x*	Double-ended (shielded) Ethernet cables for use when programming the safety configuration and the Logix EtherNet/IP network module		0M3 = 0.3 (1.0) 0M4 = 0.4 (1.3) 0M6 = 0.6 (2.0) 1 (3.2) 2 (6.6) 5 (16.4) 10 (32.8)
1585J-M8CB-*x*	Single-ended (shielded) Ethernet cables for use when programming the safety configuration and the Logix EtherNet/IP network module		2 (6.6) 5 (16.4) 10 (32.8)
1585J-M8CC-H	RJ45 insulation displacement connector (IDC) for use when making your own cables		100 (328) 300 (984) 600 (1968)
1585J-C8CB-S*xxx*	Shielded Ethernet (bulk) cable for use when making your own cables		
1202-C02	Drive-to-drive safety cable for connecting single-wide Kinetix 6000 axis modules		200 mm (7.9 in.)
1202-C03	Drive-to-drive safety cable for connecting double-wide Kinetix 6000 axis modules		350 mm (13.8 in.)
1202-C10	Drive-to-drive safety cable for connections between two Kinetix 6000 power rails, two Kinetix 7000 drives, or from the Kinetix 6000 power rail to Kinetix 7000 drive		1050 mm (41.3 in.)
2090-XXNRB-10F0P5	Resistive Brake Module (RBM) to Kinetix 6000 and Kinetix 6200/Kinetix 6500 drives	10 AWG	0.5 (1.6)
2090-XXNRB-8F0P6		8 AWG	0.6 (2.0)
2090-UXNRB-10F1P3	Resistive Brake Module (RBM) to Ultra3000 drives	10 AWG	1.3 (4.3)
2090-UXNRB-8F1P4		8 AWG	1.4 (4.6)
2090-UXNRB-6F1P5		6 AWG	1.5 (5.0)

(1) This cable does not carry the unbuffered motor encoder signals (CN1 pins 10-15). Contact your Allen-Bradley sales representative if these signals are required for your application.

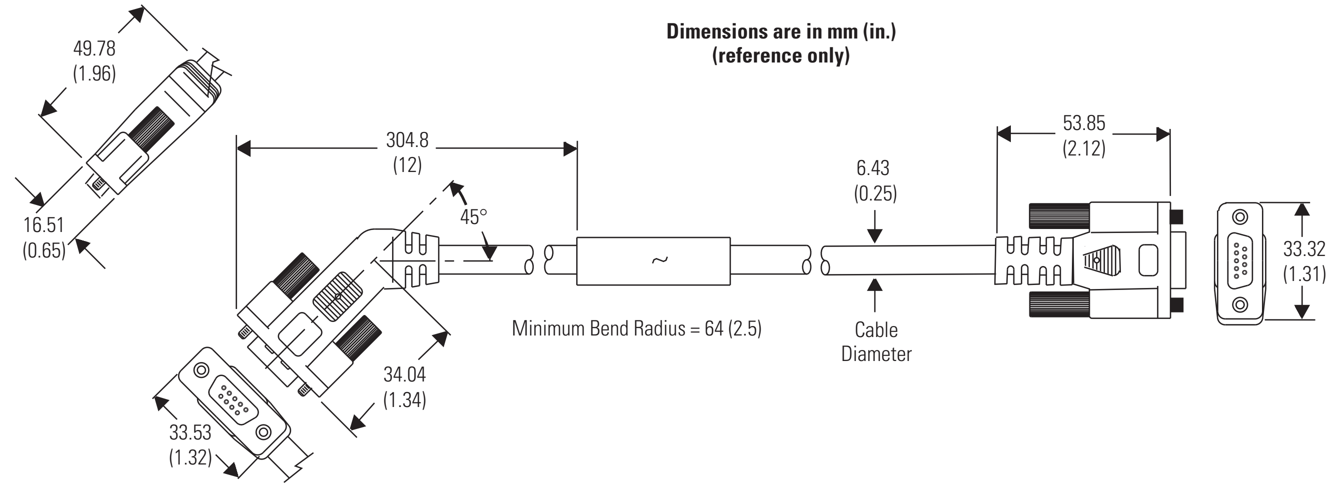
(2) Only available as 2090-SCEP*x-x*.

Interface Cable Specifications

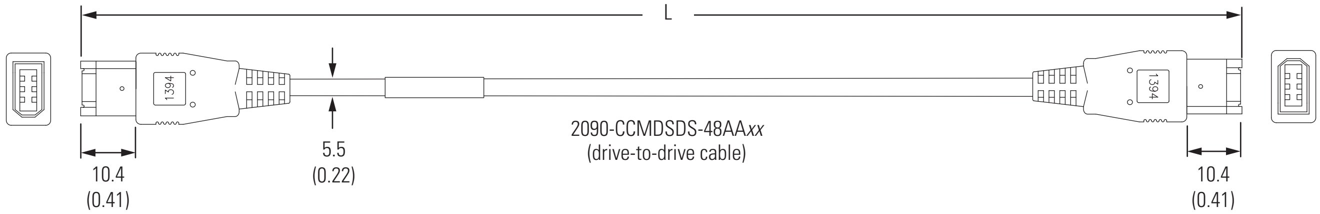
Interface Cable Cat. No.	Description	Specifications		
		Ratings	**Shield**	**Jacket Material**
2090-UXPC-D09*xx*	Ultra3000/5000 serial interface to computer	90 °C (194 °F), 30V	Aluminum Polyester 100% coverage Braid shield coverage, 85% min	TPE
2090-DAIO-D50*xx*	Kinetix 3 control interface I/O cable, flying leads			
2090-U3CC-D44*xx*	Single-axis flying lead Ultra3000 drive to 1756-M02AE module	80 °C (176 °F), 30V		
2090-U3AE-D44*xx*	Two-axis pre-wired Ultra3000 drive to 1756-M02AE module			
2090-CCMPCDS-23AA*xx*	Kinetix 3 serial interface to personal computer	80 °C (176 °F), 300V		PVC
2090-CCMCNDS-48AA*xx*	Kinetix 3 control interface to MicroLogix controller			
2090-CCMDSDS-48AA*xx*	Kinetix 3 control interface for drive-to-drive configurations			
2090-U5PM-D09*xx*	Ultra5000 drive to PanelView 300 Micro DF-1 terminal and MicroLogix system	60 °C (140 °F), 30V		
2090-U5PV-D09*xx*	Ultra5000 drive to PanelView Standard DF-1 terminal	80 °C (176 °F), 30V		TPE
2090-*x*XNRB-*xx*F*xxx*	Resistive Brake Module (RBM) to drive interface	105 °C (221 °F), 600V		
2090-SCEP*x-x*	SERCOS interface fiber-optic cable (drive to drive, drive to 1756-M*xx*SE module, or drive to 1768-M04SE module)	-55…85 °C (-67...185 °F)		Chlorinated Polyethylene
2090-SCNP*x-x*				Nylon
2090-SCVP*x-x*				Polyethylene/Kevlar covered by PVC
2090-SCVG*x-x*		-20…75 °C (-4...67 °F)		Kevlar and PVC

Interface Cable Dimensions

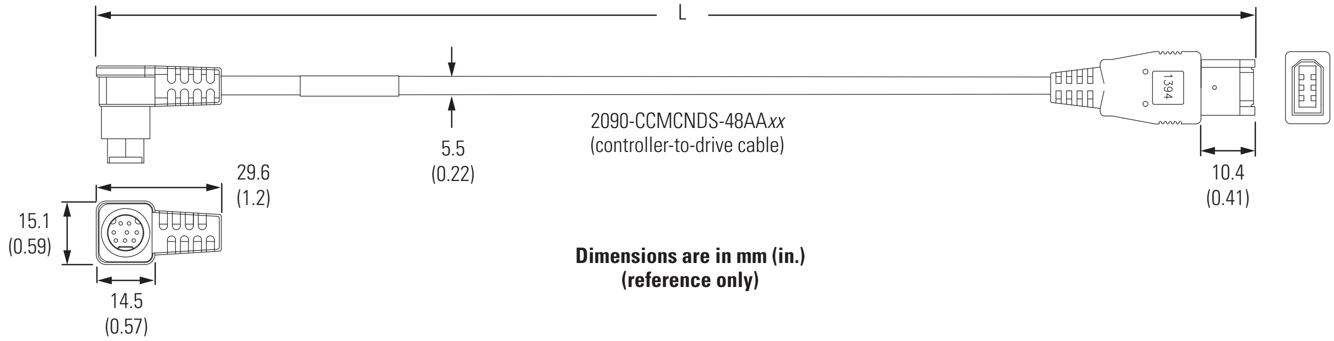
Serial Interface Cable Dimensions (catalog number 2090-UXPC-D09*xx*)



Control and Configuration Serial Cable Dimensions

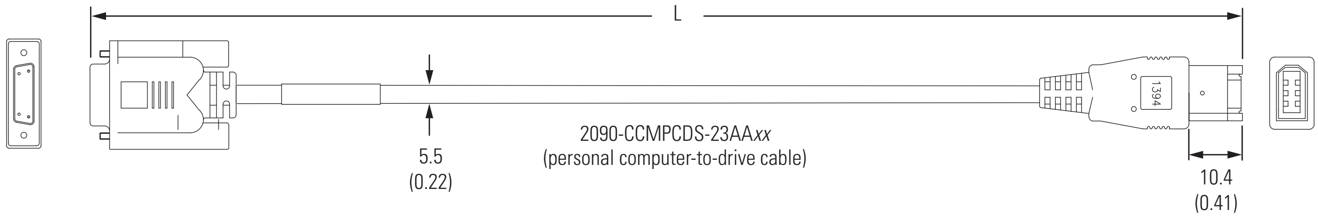


2090-CCMDSDS-48AA*xx*
(drive-to-drive cable)



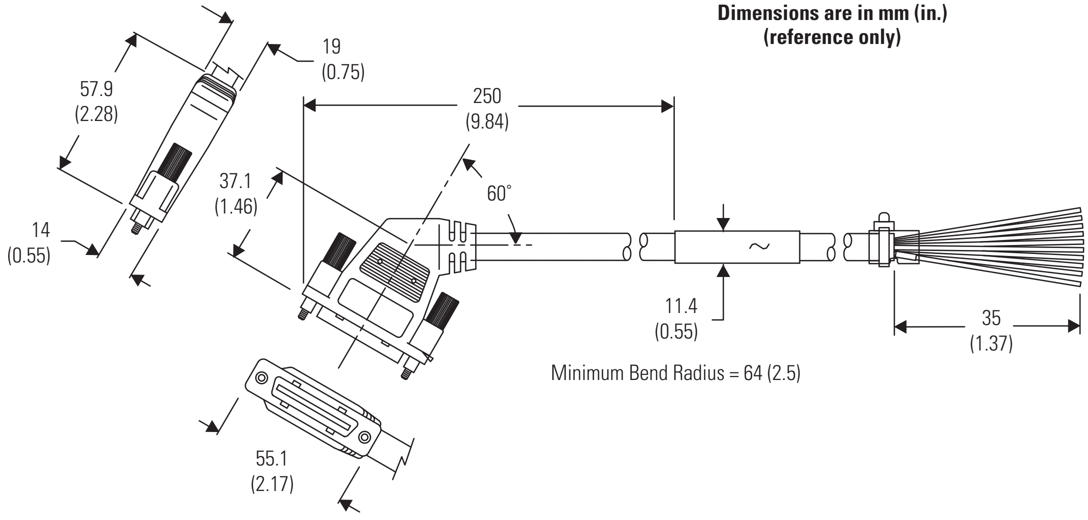
2090-CCMCNDS-48AA*xx*
(controller-to-drive cable)

Dimensions are in mm (in.)
(reference only)



2090-CCMPCDS-23AA*xx*
(personal computer-to-drive cable)

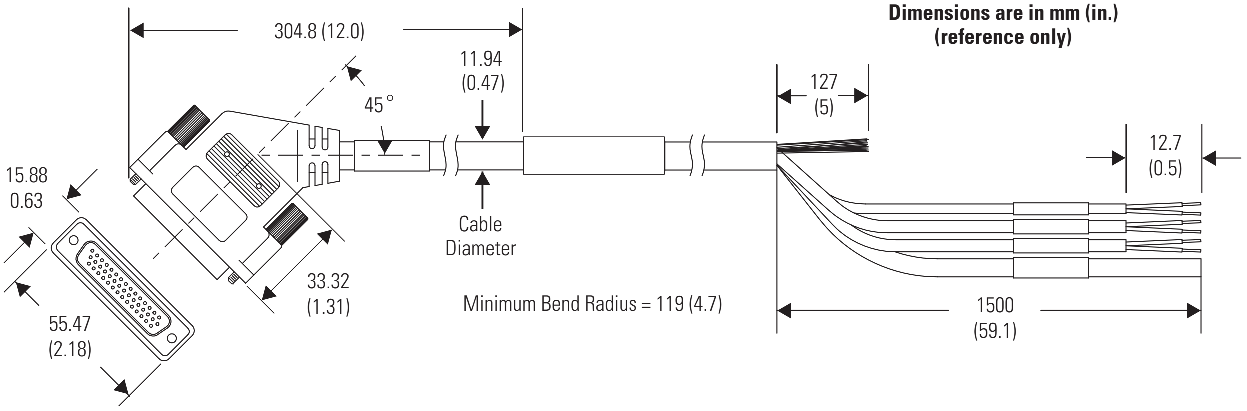
Control Interface Cable Dimensions (catalog number 2090-DAIO-D50*xx*)



**Dimensions are in mm (in.)
(reference only)**

Minimum Bend Radius = 64 (2.5)

Control Interface Cable Dimensions (catalog number 2090-U3CC-D44*xx*)

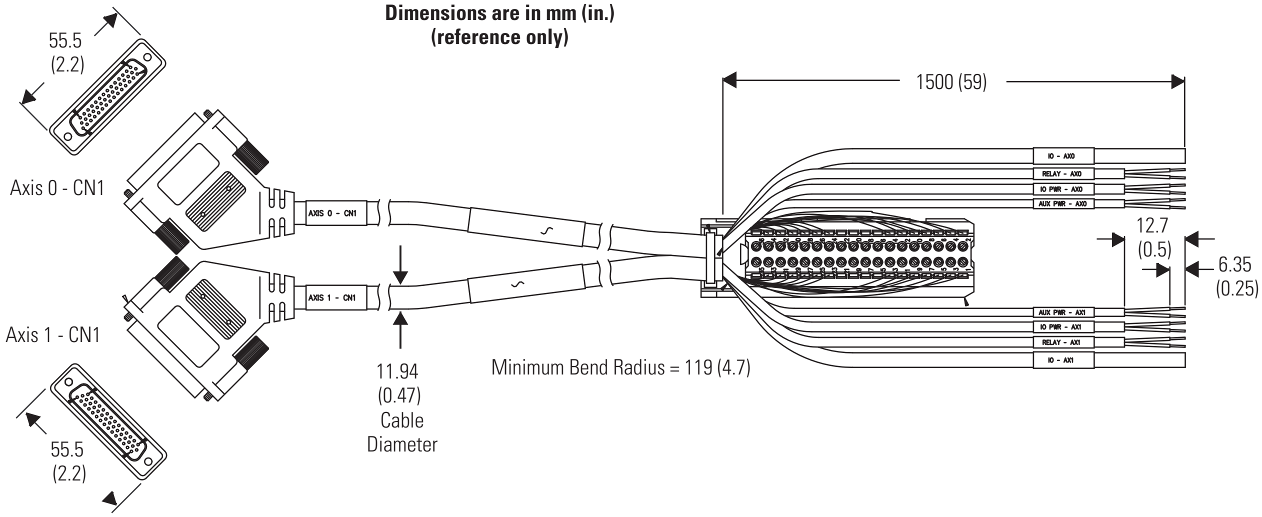


**Dimensions are in mm (in.)
(reference only)**

Minimum Bend Radius = 119 (4.7)

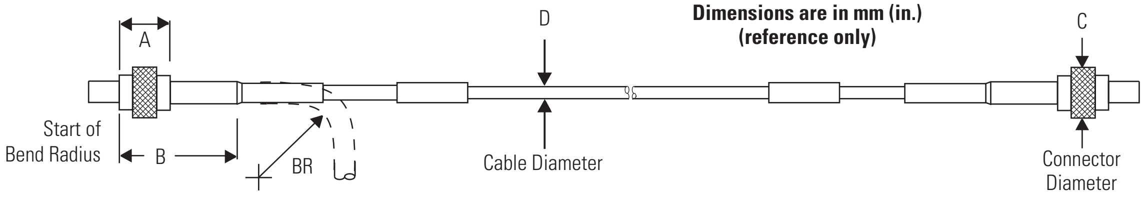
ControlLogix 1756-M02AE Card Encoder Cable Dimensions (catalog number 2090-U3AE-D44*xx*)

**Dimensions are in mm (in.)
(reference only)**



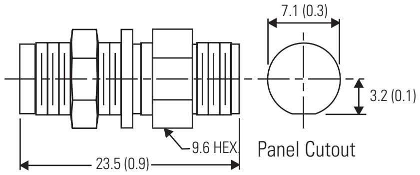
Minimum Bend Radius = 119 (4.7)

SERCOS interface Fiber-optic Cable Dimensions



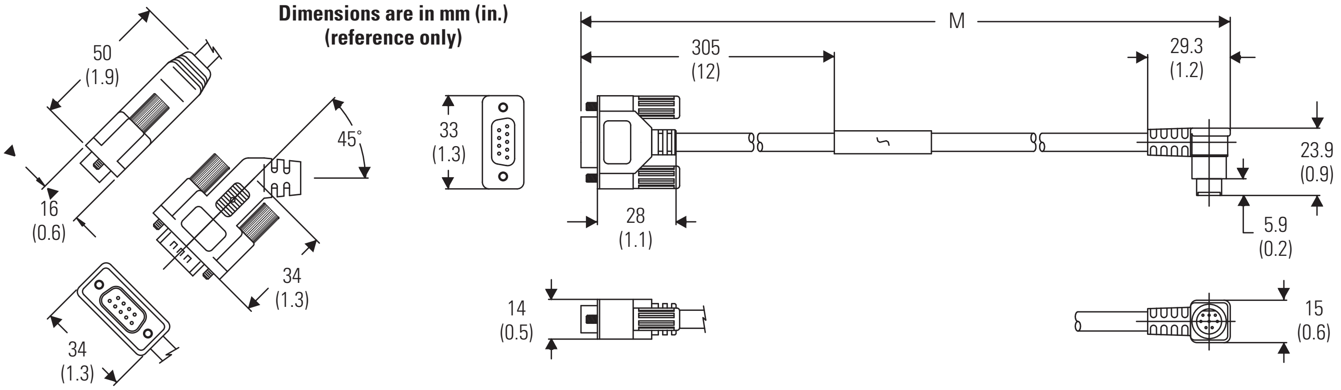
Fiber-optic Cable Cat. No.	A mm (in.)	B mm (in.)	BR [1] mm (in.)	C mm (in.)	D mm (in.)
2090-SCEP*x-x*	7 (0.27)	25 (1.0)	25 (1.0)	10 (0.39)	2.2 (0.09)
2090-SCNP*x-x*					
2090-SCVP*x-x*			40 (1.6)		5.0 (0.2)
2090-SCVG*x-x*			30 (1.2)		

Bulkhead Adapter Dimensions (2090-S-BLHD)

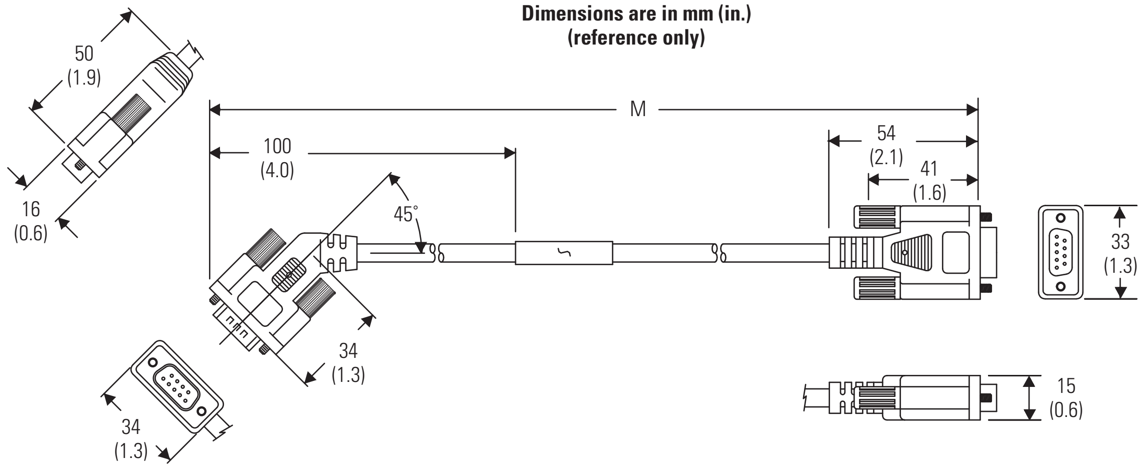


(1) Standard cables have a regular maintenance and installation bend radius of 10 times (10x) the cable diameter.

Ultra5000 Drive to PanelView Terminal Cable Dimensions (catalog number 2090-U5PM-D09*xx*)

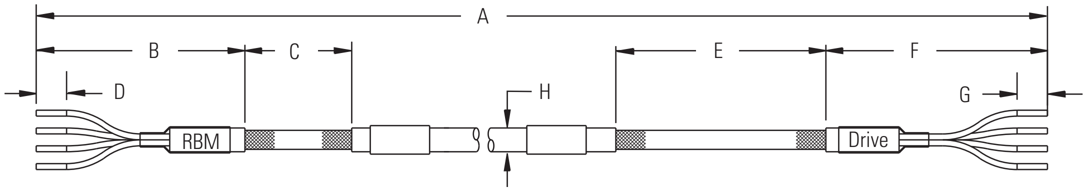


Ultra5000 Drive to PanelView Terminal Cable Dimensions (catalog number 2090-U5PV-D09*xx*)



RBM Module Interface Cable Dimensions

Dimensions are in mm (in.)
(reference only)



RBM Module Cable Cat. No.	A mm (in.)	B mm (in.)	C mm (in.)	D mm (in.)	E mm (in.)	F mm (in.)	G mm (in.)	H mm (in.)
2090-XXNRB-10F0P5	517 (20.3)	115 (4.5)	50 (1.9)	16 (0.6)	120 (4.7)	74 (2.9)	16 (0.6)	16 (0.63)
2090-UXNRB-10F1P3	1320 (52.0)					105 (4.1)		
2090-XXNRB-8F0P6	619 (24.4)					74 (2.9)		19 (0.75)
2090-UXNRB-8F1P4	1395 (54.9)					117 (4.6)		
2090-UXNRB-6F1P5	1527 (60.1)					129 (5.1)		23 (0.90)

2090-Series Interface Cable Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering charts below to understand the configuration of your cables. For questions regarding product availability, contact your Allen-Bradley distributor.

SERCOS Interface Fiber-optic Cables

2090 - S C *x* *x* *x-x*

Cable Length
Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Cable Option
P = Plastic
G = Glass

Enclosure Option
E = Enclosure only
V = PVC jacket
N = Nylon jacket

Connector Option
C = Mating connectors (at both ends)

Type
S = SERCOS interface

Bulletin Number

Control and Configuration Serial Interface Cables

2090 - C CM *xx* DS *xx* - AA *xx*

Cable Length
Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Cable Type
AA = Standard (non-flex)

Communication Method
23 = RS232 (with ASCII messaging)
48 = RS485 (with Modbus-RTU)

Connector Type (drive end of cable)
DS = Drive-side connector

Connector Type (other end of cable)
PC = DB-9-pin serial connector for personal computer connections
CN = 9-pin mini-DIN connector for MicroLogix 1100 or 1400 controller connections
DS = Drive-side connector

Cable Type
CM = Communication

Accessory Component
C = Cable

Bulletin Number

Control and Serial Interface Cables

2090 - *xx* *xx* - D*xx* *xx*

Cable Length
Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Connector Type
D09 = 9-pin D-Shell
D44 = 44-pin, D-Shell
D50 = 50-pin, mini-D

Host
AE or CC= Ultra3000 to ControlLogix 1756-M02AE cable
PM = Ultra5000 to PanelView 300 MicroLogix DF-1
PV = Ultra5000 to standard PanelView DF-1
PC = Personal computer RS232/RS485 serial interface

Drive
U3 = Ultra3000
U5 = Ultra5000
UX = Ultra3000 or Ultra5000
DA = Kinetix 3

Bulletin Number

Breakout Components and Connector Kits

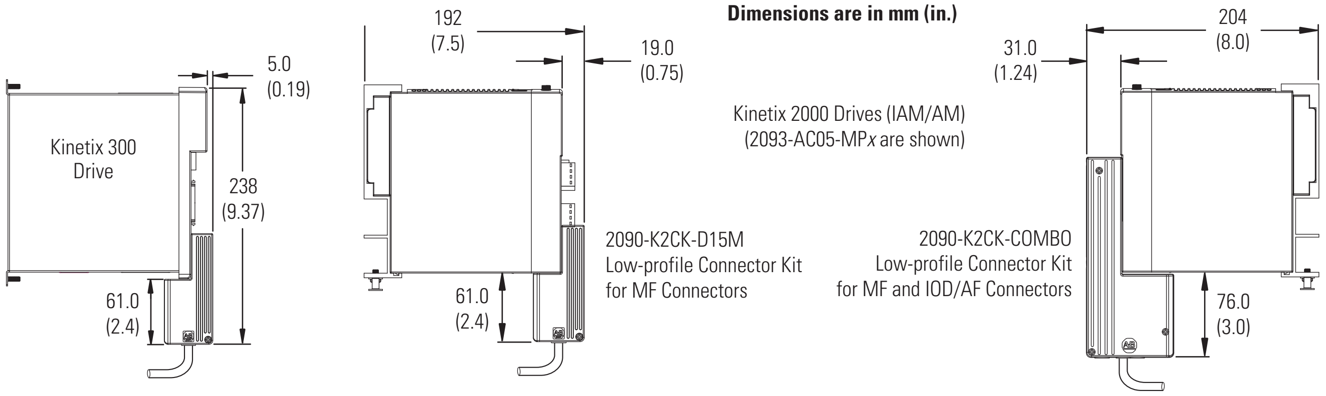
This section contain examples, descriptions, dimensions, specifications, and catalog numbers for breakout components and connector kits.

Low-profile Connector Kit Examples

Use these examples to identify the best solution for wiring flying-lead feedback and I/O cables to servo drive or Line Interface Modules (LIM).

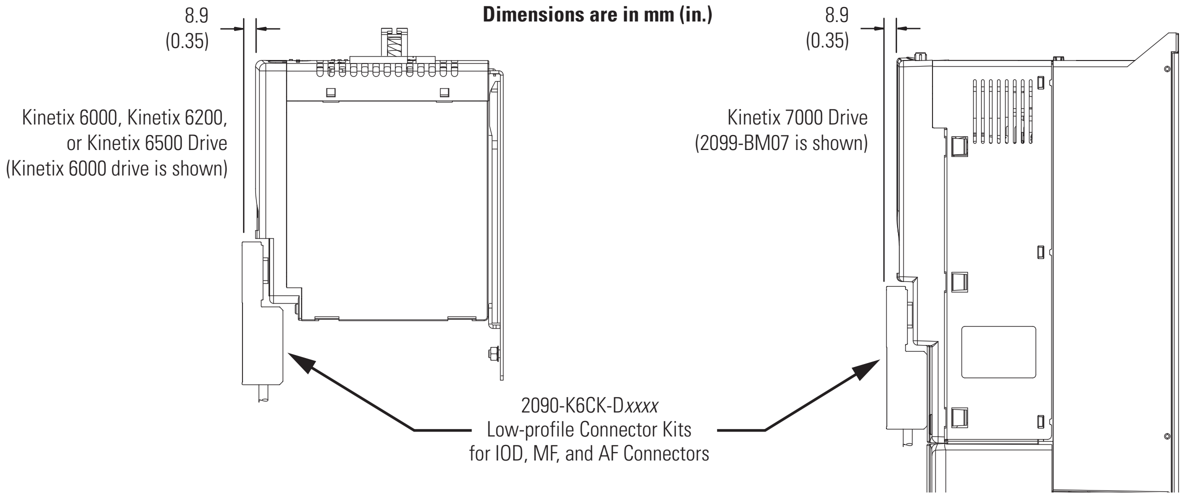
In this example, the Kinetix 2000 (IAM/AM) is shown with catalog number 2090-K2CK-D15M, for use with the motor feedback (MF) connector. Also shown is catalog number 2090-K2CK-COMBO for use with the motor feedback (MF) and I/O (IOD/AF) connectors. Refer to Low-profile Connector Kit Components on page 441 for more information.

Kinetix 2000 (IAM/AM) and Kinetix 300 Examples



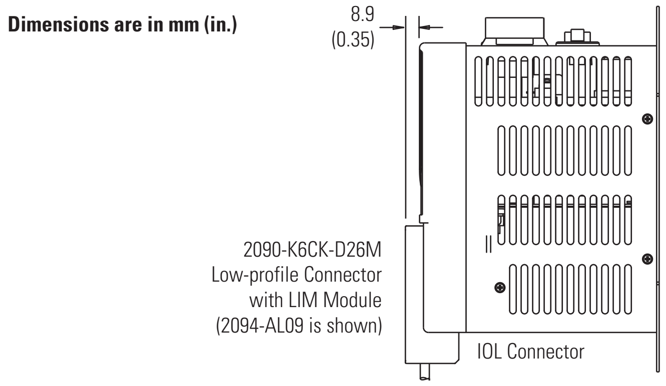
In this example, the Kinetix 6000 and Kinetix 7000 drives are shown with low-profile connector kits (catalog numbers 2090-K6CK-D*xxx*). Use these kits with the I/O (IOD), motor feedback (MF), and auxiliary feedback (AF) connectors. The 2090-K6CK-D*xxx* kits also apply to the Kinetix 6200 and Kinetix 6500 drives. The Refer to Low-profile Connector Kit Components on page 441 for more information.

Kinetix 6000, Kinetix 6200, Kinetix 6500, and Kinetix 7000 Low-profile Connector Examples



In this example, the LIM module is shown with low-profile connector kit (catalog number 2090-K6CK-D26M). Use this connector with the I/O (IOL) connector on the 2094-AL09 and 2090-BL02 LIM modules. Refer to Low-profile Connector Kit Components for more information.

LIM Module Low-profile Connector Example



Low-profile Connector Kit Components

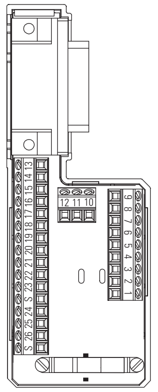
Low-profile connector kits are designed for use with the Kinetix 2000 IAM/AM, Kinetix 6000 IAM/AM, and Kinetix 7000 drives, and LIM modules. Use this table to identify the low-profile connector kit for your feedback or I/O connector.

IMPORTANT	The flying-lead compatible cables listed below require connector kits to complete feedback and I/O connections to the drive.

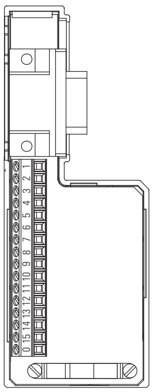
Cat. No.	Description	Cable Compatibility
2090-K2CK-D15M	Low-profile connector kit for motor feedback (15-pin, male, D-sub). Use with any Kinetix 2000 IAM/AM module or Kinetix 300 drive and compatible motors with incremental or high-resolution feedback. Does not include 3.6V battery (catalog number 2090-DA-BAT2) required for use with TLY-A*xxxx*-B high-resolution motors and 17-bit encoders.	2090-XXNFMF-S*xx* 2090-CFBM4DF-CDAF*xx* 2090-CFBM7DF-CEAA*xx* 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DF-CDAF*xx* 2090-CFBM6DF-CBAA*xx*
2090-K2CK-COMBO	Low-profile connector kit for motor feedback (15-pin, male, D-sub) and IO (44-pin, male, D-sub). Use with any Kinetix 2000 IAM/AM module and compatible motors with incremental or high-resolution feedback. Does not include 3.6V battery (catalog number 2090-DA-BAT2) required for use with TLY-A*xxxx*-B high-resolution motors and 17-bit encoders. The 2090-K2CK-COMBO kit, mounted on the Kinetix 2000 (IAM/AM) drive, fits in a standard 10 in. enclosure.	
2090-K6CK-D15M	Low-profile connector kit for motor feedback (15-pin, male, D-sub). Use with any Kinetix 6000, Kinetix 6200, Kinetix 6500, or Kinetix 7000 drive and compatible motors with incremental or high-resolution feedback.	
	Low-profile connector kit for motor feedback (15-pin, male, D-sub). Use with any Kinetix 6000 IAM/AM module and MPL-B*xxxx*-R or MPM-A/B*xxxxx*-2 (resolver feedback) motors.	2090-CFBM7DF-CEAA*xx* 2090-CFBM7DF-CEAF*xx*
2090-K6CK-D15F	Low-profile connector kit for auxiliary feedback (15-pin, female, D-sub). Use with any Kinetix 6000 IAM/AM module or Kinetix 7000 drive auxiliary feedback application.	Customer Supplied
2090-K6CK-D26M	Low-profile connector kit for I/O (26-pin, male, D-sub). For use with any Kinetix 6000 IAM/AM module, Kinetix 7000 drive, or 2094-AL09 and 2094-BL02 LIM module.	
2090-K6CK-D44M	Low-profile connector kit for I/O, safety, and auxiliary feedback (44-pin, male, D-sub). For use with any Kinetix 6200 or Kinetix 6500 control module.	

Low-profile Connector Kit Dimensions (catalog numbers 2090-K6CK-D26M, 2090-K6CK-D15M, 2090-K6CK-D15F)

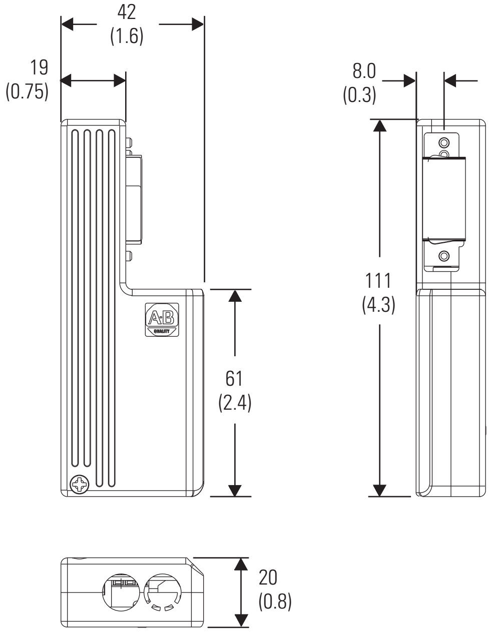
Dimensions are in mm (in.)



2090-K6CK-D26M
26-pin I/O Connector

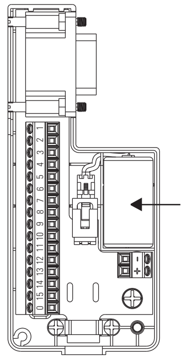


2090-K6CK-D15M (motor feedback)
2090-K6CK-D15F (auxiliary feedback)
15-pin Feedback Connector



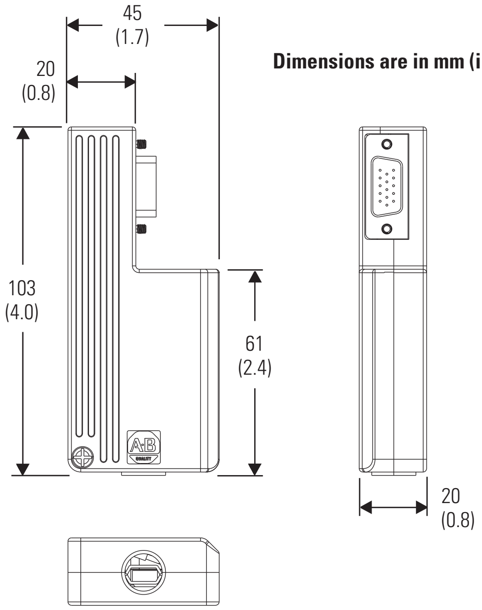
Low-profile Connector Kit Dimensions (catalog number 2090-K2CK-D15M)

Dimensions are in mm (in.)

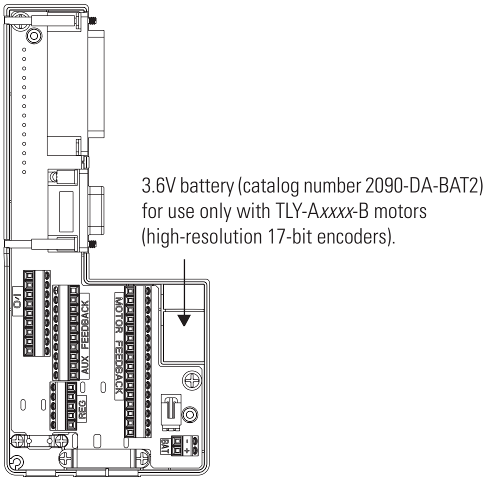


3.6V battery (catalog number 2090-DA-BAT2) for use only with TLY-A*xxxx*-B motors (high-resolution 17-bit encoders).

2090-K2CK-D15M (motor feedback)
15-pin Feedback Connector

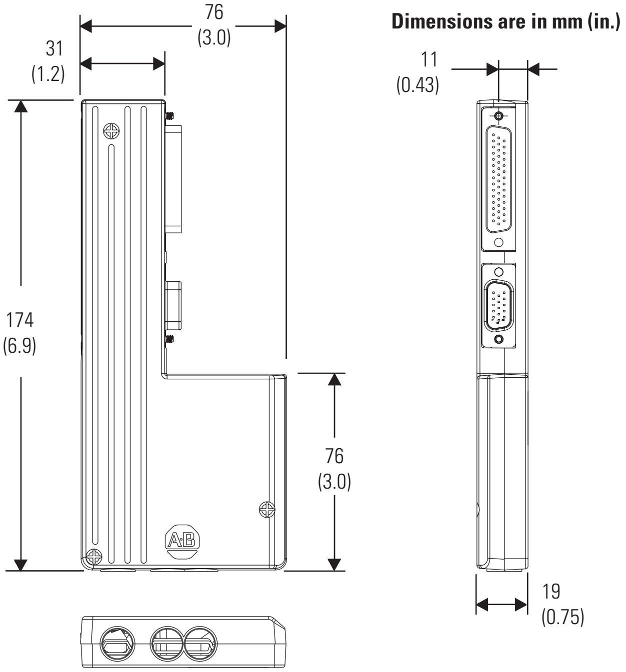


Low-profile Connector Kit Dimensions (catalog number 2090-K2CK-COMBO)

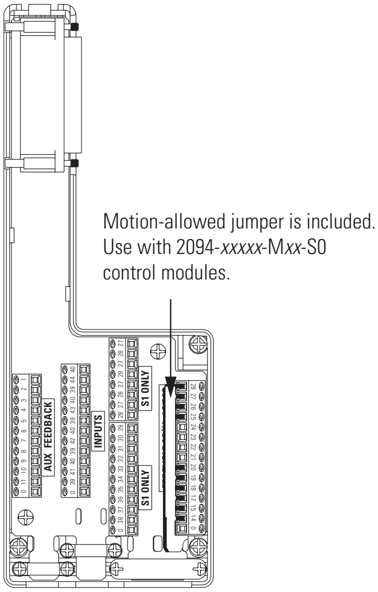


3.6V battery (catalog number 2090-DA-BAT2) for use only with TLY-A*xxxx*-B motors (high-resolution 17-bit encoders).

2090-K2CK-COMBO (motor feedback and I/O)
15-pin and 44-pin Connector

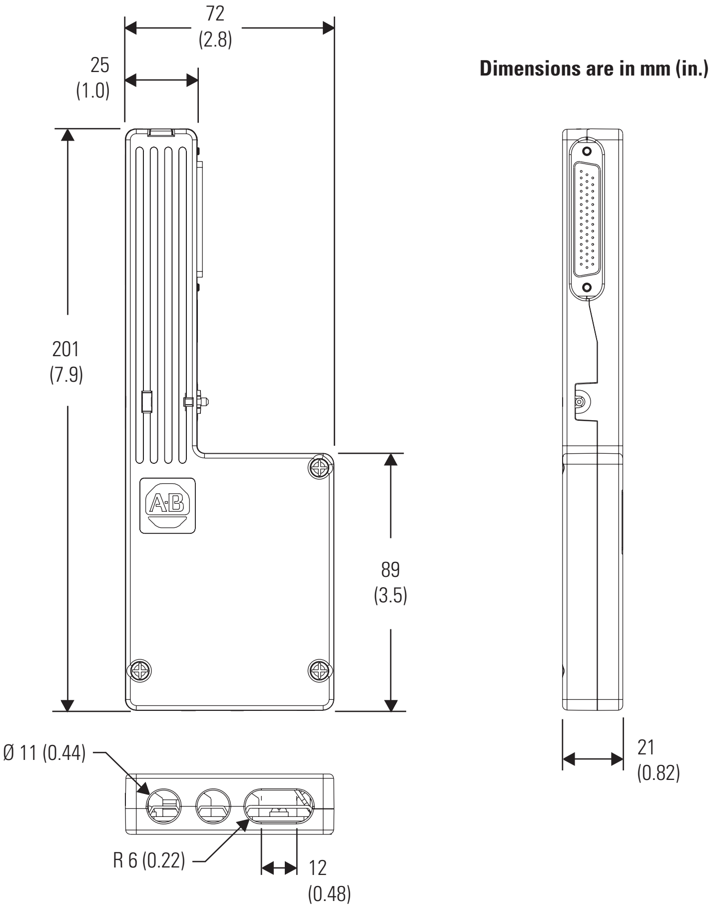


Low-profile Connector Kit Dimensions (catalog number 2090-K6CK-D44M)



Motion-allowed jumper is included. Use with 2094-*xxxxx*-M*xx*-S0 control modules.

2090-K6CK-D44M
(auxiliary feedback, safety, and I/O)
44-pin Connector

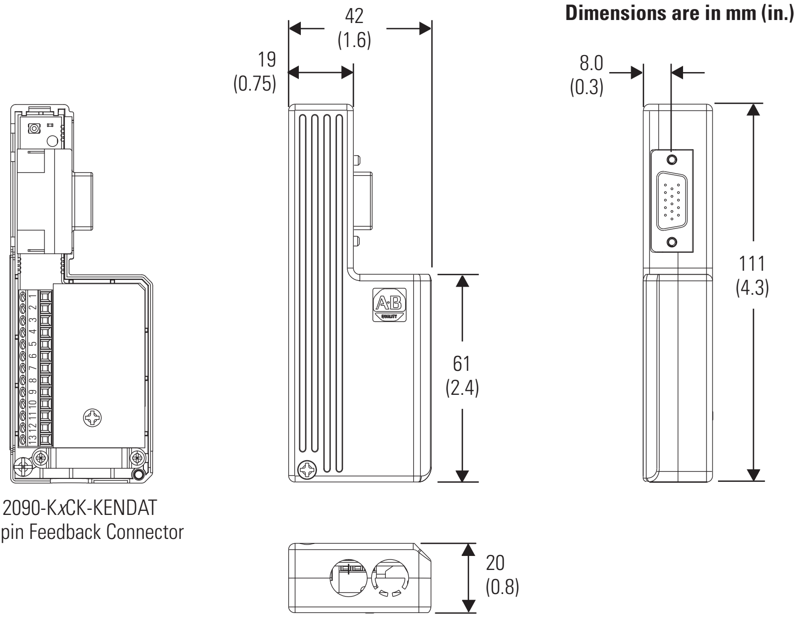


Low-profile Feedback Modules

This low-profile feedback module is designed for use with the Kinetix 6000 IAM/AM modules and Kinetix 7000 drives.

Cat. No.	Description	Cable Compatibility
2090-K6CK-KENDAT	Low-profile feedback module (15-pin, male, D-sub) used to enable operation of drives with EnDat feedback. Use with any Kinetix 6000 IAM/AM module and compatible motors with Endat encoders.	2090-XXNFMF-S*xx* 2090-CFBM4DF-CDAF*xx* 2090-CFBM7DF-CDAF*xx*
2090-K7CK-KENDAT	Low-profile feedback module (15-pin, male, D-sub) used to enable operation of drives with EnDat feedback. Use with any Kinetix 7000 drive and compatible motors with Endat encoders.	

Low-profile Feedback Module Dimensions (catalog number 2090-K*x*CK-KENDAT)



Low-profile Connector Kit Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering charts below to understand the configuration of your component. For questions regarding product availability, contact your Allen-Bradley distributor.

2090 - K*x*CK - *xxxxx*

Connector Type
D15M = 15-pin, male, D-sub, for motor feedback
D15F = 15-pin, female, D-sub, for auxiliary feedback
D15MF = 15-pin, male, with filter, D-sub, for motor feedback
D26M = 26-pin, male, D-sub, for I/O
D44M = 44-pin, D-sub, for I/O, safety, and auxiliary feedback
COMBO = 15-pin and 44-pin, D-sub, for feedback and I/O
KENDAT = 15-pin, D-sub, used to enable operation of drives with EnDat feedback

Drive
K2CK = Kinetix 2000 and Kinetix 300 drives
K6CK = Kinetix 6000 or Kinetix 7000 drives, and LIM modules (2094-AL09 and 2094-BL02 only)
K7CK = Kinetix 7000 drives

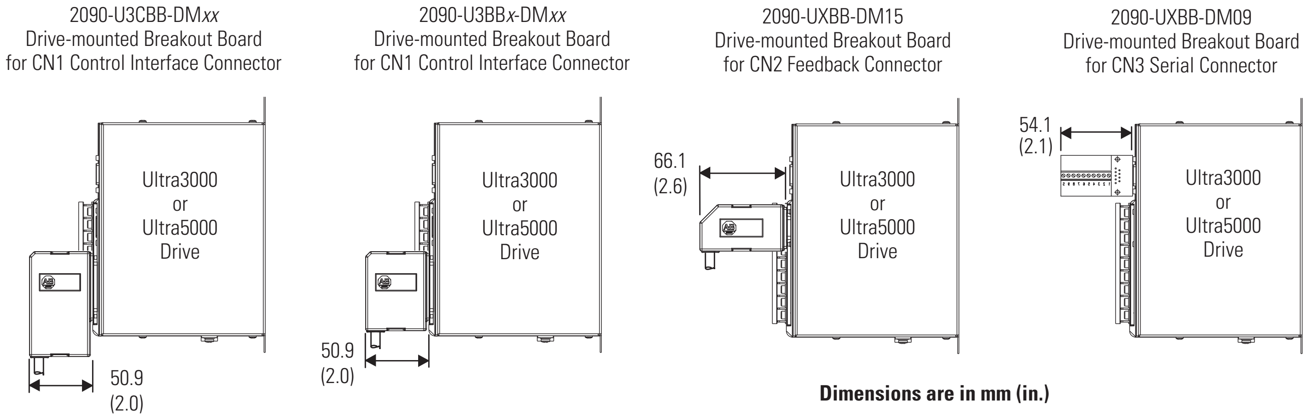
Bulletin Number

Drive-mounted Breakout Board Kit Examples

Use these examples to identify the best solution for wiring your flying-lead control interface, motor feedback, and serial cables to Ultra3000, Ultra5000, and Kinetix 3 drives.

In this example, the Ultra3000/5000 drives are shown with drive-mounted breakout board kits (catalog number 2090-U*xxx*-DM*xx*). Drive-mounted breakout board kits are available for the control interface (CN1), motor feedback (CN2), and serial interface (CN3) connectors. Refer to Drive-mounted Breakout Board Components on page 446 for more information.

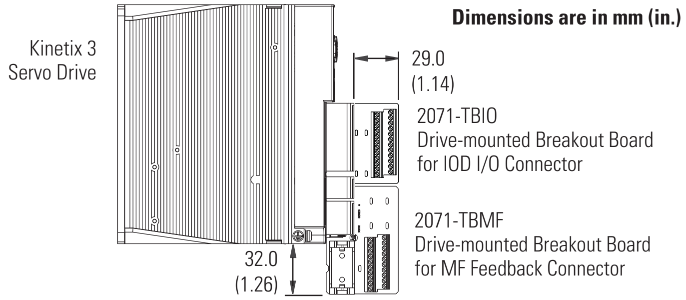
Ultra3000/5000 Drive-mounted Breakout Board Examples



TIP The 2090-UXBB-DM15 (feedback) kit is also compatible with the Kinetix 2000 IAM/AM, Kinetix 6000 IAM/AM, and Kinetix 7000 drives (MF feedback connectors only).

In this example, the Kinetix 3 drives are shown with drive-mounted breakout boards (catalog numbers 2071-TBMF and 2071-TBIO). Use the 2071-TBMF breakout board with 2090-CFBM6DF-CBAA*xx* feedback cables or when your motor or actuator has high-resolution encoder feedback. Use the 2071-TBIO breakout board for making flying-lead cable connections to twenty-four of the most commonly used terminals in the 50-pin IOD connector. Refer to Drive-mounted Breakout Board Components on page 446 for more information.

Kinetix 3 Drive-mounted Breakout Board Examples



Drive-mounted Breakout Board Components

Drive-mounted breakout boards are designed for use with Ultra3000, Ultra5000, and Kinetix 3 drives. Use this table to identify the drive-mounted breakout board for your serial, I/O or feedback connector.

IMPORTANT	The 2090-XXNFMF-S*xx* and 2090-CFBM*x*DF-*xx*A*xxx* flying-lead feedback cables require connector kits to complete feedback connections to the drive.

Drive-mounted Breakout Boards

Cat. No.	Description
2090-U3BB-DM12 (1)	12-pin, drive-mounted breakout board for Ultra3000 CN1 connector recommended for use with SERCOS interface applications.
2090-U3BB2-DM44 (1) (2)	44-pin, drive-mounted breakout board for Ultra3000 CN1 control interface connector.
2090-U3CBB-DM12 (3)	12-pin, drive-mounted breakout board for Ultra3000 CN1 connector recommended for use with SERCOS interface applications with 24…5V auxiliary power converter.
2090-U3CBB-DM44 (3)	44-pin, drive-mounted breakout board for Ultra3000 CN1 connector with 24V to 5V auxiliary power converter.
2090-UXBB-DM15 (4)	15-pin, drive-mounted breakout board for Ultra3000/5000 CN2 feedback connector.
2090-UXBB-DM09	9-pin, drive-mounted breakout board for Ultra3000/5000 CN3 serial interface.
2071-TBIO	50-pin, drive-mounted breakout board for Kinetix 3 IOD I/O connector.
2071-TBMF	20-pin, drive-mounted breakout board for Kinetix 3 MF feedback connector.

(1) For specifications, refer to the CN1 Control Interface Breakout Boards Installation Instructions, publication 2090-IN007.

(2) This breakout board accepts 1.5 to 0.14 mm^2 (16 to 26 AWG) wire. For applications that require a 44-pin drive-mounted breakout board that accepts 4 to 0.34 mm^2 (12 to 22 AWG) wire, contact your local Allen-Bradley representative.

(3) Only for use with the Ultra3000 (2098-DSD-005*x-xx*, 2098-DSD-010*x-xx*, 2098-DSD-020*x-xx*) drives. Requires an external +24V DC power supply. For specifications, refer to the CN1 Control Interface Breakout Boards with Integral 24V to 5V Auxiliary Power Converter Installation Instructions, publication 2090-IN008.

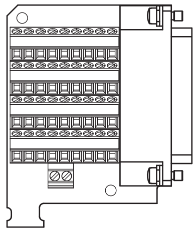
(4) For specifications, refer to the CN2 Motor Feedback Breakout Board Installation Instructions, publication 2090-IN006.

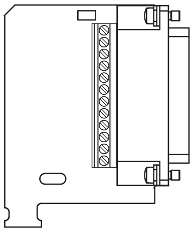
These (CN1) breakout boards apply to Ultra3000 drives (catalog numbers 2098-DSD-005, 2098-DSD-010, and 2098-DSD-020) in applications where 5V DC control power (if required) is user-supplied. The 12-pin board is intended for use with SERCOS drives, but may be used in non-SERCOS applications with minimal I/O requirements.

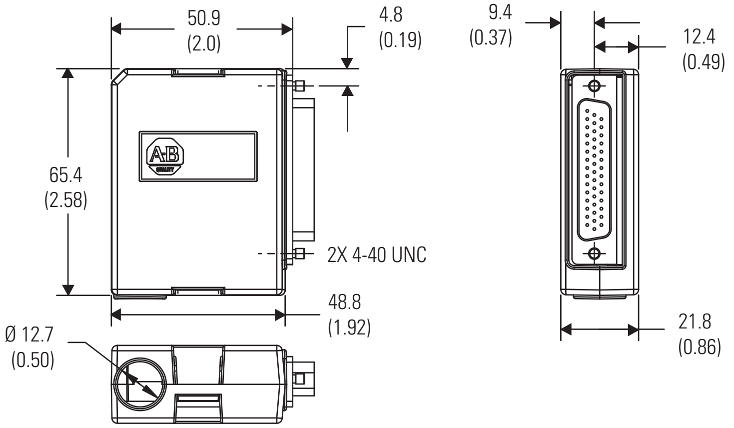
IMPORTANT	The 2090-U3BB-DM*xx* is required when wiring to the Ultra3000 (2098-DSD-030-SE/DN, 2098-DSD-075-SE/DN, 2098-DSD-150-SE/DN, or 2098-DSD-HV*xxx*-SE/DN) SERCOS/DeviceNet interface drives due to space restrictions when connecting the SERCOS or DeviceNet interface cables.

Drive-mounted Breakout Board Dimensions (catalog numbers 2090-U3BB-DM12 and 2090-U3BB2-DM44)

Dimensions are in mm (in.)


44-pin CN1 Connector
(shown without cover)
2090-U3BB2-DM44


12-pin CN1 Connector
(shown without cover)
2090-U3BB-DM12

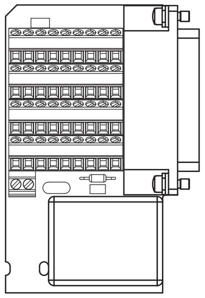


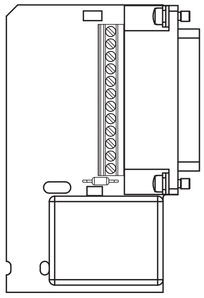
These (CN1) breakout boards apply to Ultra3000 drives (catalog numbers 2098-DSD-005, 2098-DSD-010, and 2098-DSD-020) in applications where a 24…5V DC converter for control power is required. The 12-pin board is intended for use with SERCOS drives, but may be used in non-SERCOS applications with minimal I/O requirements.

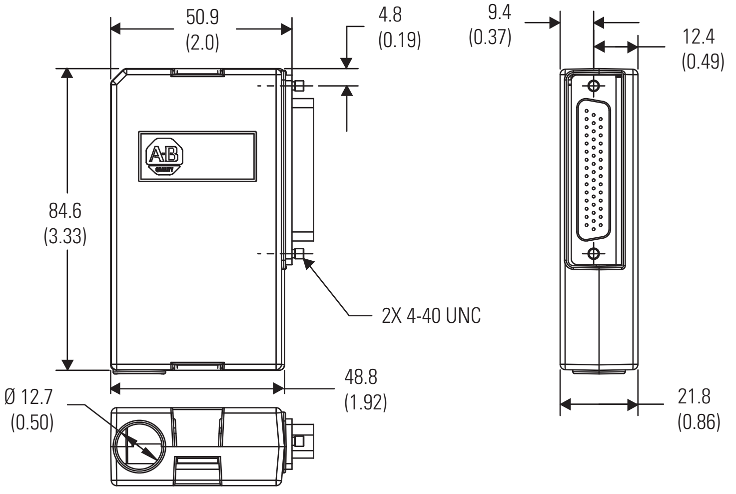
IMPORTANT	Do not use the 2090-U3CBB-DM*xx* when wiring to the Ultra3000 (2098-DSD-030-SE/DN, 2098-DSD-075-SE/DN, 2098-DSD-150-SE/DN, or 2098-DSD-HV*xxx*-SE/DN) drives.

Drive-mounted Breakout Board Dimensions (catalog numbers 2090-U3CBB-DM12 and 2090-U3CBB-DM44)

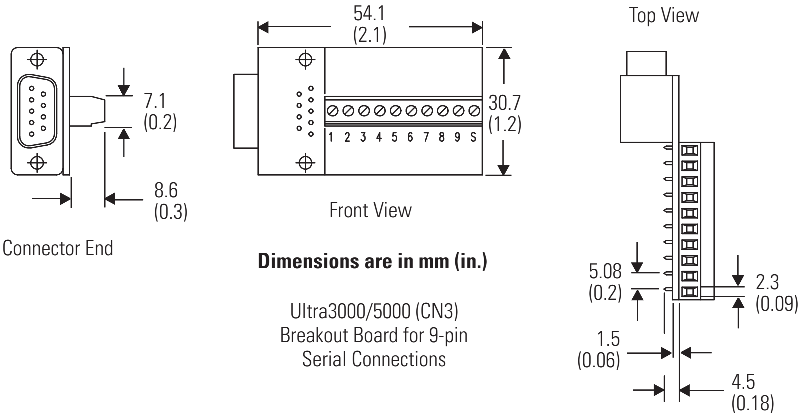
Dimensions are in mm (in.)


44-pin CN1 Connector
(shown without cover)
2090-U3CBB-DM44


12-pin CN1 Connector
(shown without cover)
2090-U3CBB-DM12



Drive-mounted Breakout Board Dimensions (catalog number 2090-UXBB-DM09)

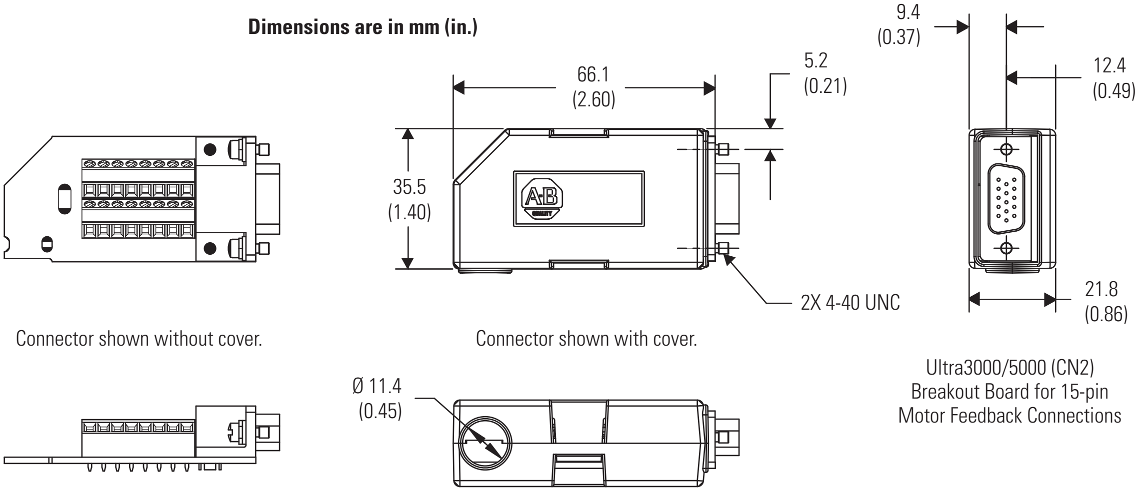


Dimensions are in mm (in.)

Ultra3000/5000 (CN3)
Breakout Board for 9-pin
Serial Connections

Drive-mounted Breakout Board Dimensions (catalog number 2090-UXBB-DM15)

Dimensions are in mm (in.)



Connector shown without cover.

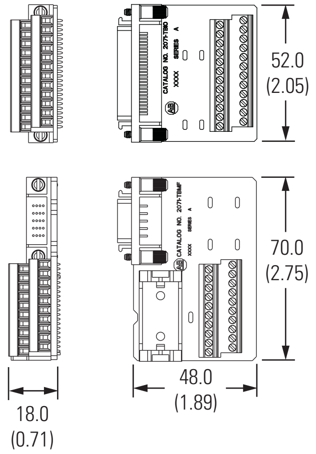
Connector shown with cover.

Ultra3000/5000 (CN2)
Breakout Board for 15-pin
Motor Feedback Connections

Drive-mounted Breakout Board Dimensions (catalog numbers 2071-TBMF and 2071-TBIO)

Kinetix 3 (IOD)
2071-TBIO Breakout Board for
50-pin Control Interface Connections

Kinetix 3 (MF)
2071-TBMF Breakout Board for
20-pin Motor Feedback Connections



Dimensions are in mm (in.)

Drive-mounted Breakout Board Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering charts below to understand the configuration of your component. For questions regarding product availability, contact your Allen-Bradley distributor.

2090 - *xx* *xxx* - DM*xx*

Connector Type
DM44 = 44-pin, male, D-sub, 44 connections
DM12 = 44-pin, male, D-sub, 12 connections
DM15 = 15-pin, male, D-sub
DM09 = 9-pin, male, D-sub

Kit Type
BB = Breakout board
BB2 = Breakout board (applies to catalog number 2090-U3BB2-DM44 only)
CBB = Breakout board with 24…5V auxiliary power converter

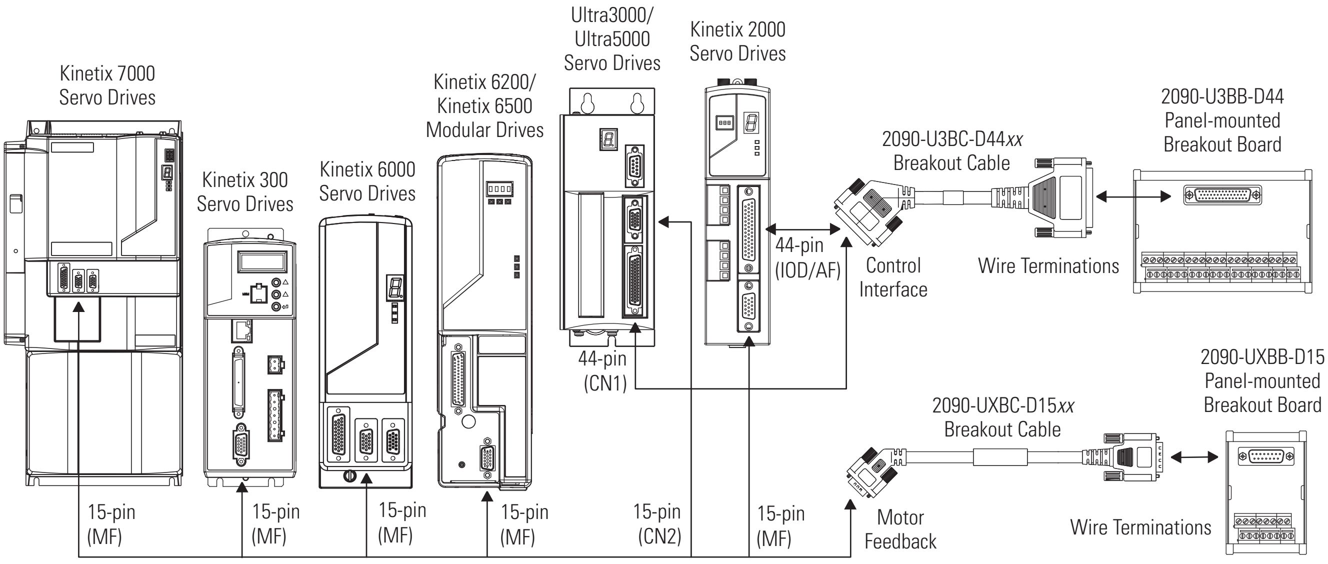
Drive
UX = Ultra3000 and Ultra5000 drives
U3 = Ultra3000 drives

Bulletin Number

Panel-mounted Breakout Board Kit Examples

Panel-mounted breakout board kits for motor feedback (catalog number 2090-UXBK-D15*xx*) and control interface (catalog number 2090-U3BK-D44*xx*) are designed for use with Ultra3000, Ultra5000, Kinetix 300, Kinetix 2000, Kinetix 6000, Kinetix 6200, Kinetix 6500, and Kinetix 7000 drives. Refer to Panel-mounted Breakout Board Components on page 450 for more information.

Panel-mounted Breakout Board Examples



Panel-mounted Breakout Board Components

Breakout boards, cables, and kits (designed for DIN rail mounting on the panel) and for use with Ultra3000/5000, Kinetix 2000, Kinetix 6000, and Kinetix 7000 drives are shown below. These breakout board components can be ordered separately, or as a kit containing both terminal block and cable.

Panel-mounted Breakout Board Kits

Cat. No.	Description	Cable Compatibility
2090-UXBK-D15*xx*	DIN rail terminal block (catalog number 2090-U*x*BB-D*xx*) and cable (catalog number 2090-U*x*BC-D*xxxx*) for motor feedback connector (15-pin, male, D-sub). Use with any Kinetix 300, Kinetix 2000, Kinetix 6000, Kinetix 6200, Kinetix 6500, or Kinetix 7000 drives (MF connector) or Ultra3000/5000 drives (CN2 connector) for motor feedback connections.	2090-XXNFMF-S*xx* 2090-CFBM4DF-CDAF*xx* 2090-CFBM7DF-CEAA*xx* 2090-CFBM7DF-CEAF*xx* 2090-CFBM6DF-CBAA*xx*
	Terminal block and cable for motor feedback connector (15-pin, male, D-sub). Use with Kinetix 6000 drives, MPL-B*xxxx*R, and MPM-A/B*xxxxx*-2 (resolver feedback) motors.	2090-CFBM7DF-CEAA*xx*
2090-U3BK-D44*xx*	Terminal block and cable for control interface connector (44-pin, male, D-sub). Use with Ultra3000 drives (CN1 connector) or Kinetix 2000 drives (IOD/AF connector).	Customer Supplied

Panel-mounted Breakout Boards

Cat. No.	Description
2090-UXBB-D15	15-pin terminal block with D-sub connector. Use with any Kinetix 300, Kinetix 2000, Kinetix 6000, Kinetix 6200, Kinetix 6500, or Kinetix 7000 drives (MF connector) or Ultra3000/5000 drives (CN2 connector) for motor feedback connections.
2090-U3BB-D44	44-pin terminal block with D-sub connector. Use with Kinetix 300 drives (CN1 connector) or Kinetix 2000 drives (IOD/AF connector) for control interface connections.

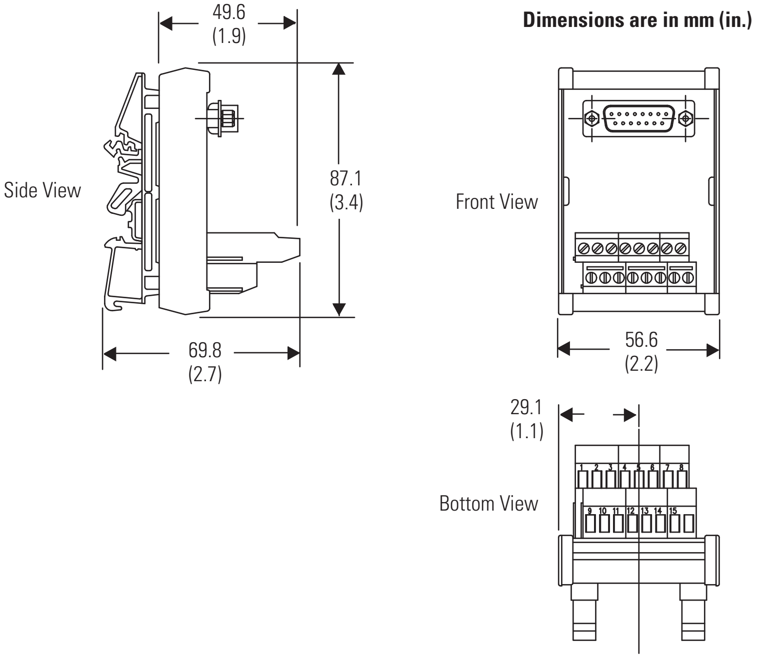
IMPORTANT The flying-lead compatible cables listed above require either 2090-UXBB-DM15 (drive-mounted) or 2090-UXBB-D15 (panel-mounted) breakout board connector kits to complete feedback and I/O connections to the drive.

Panel-mounted Breakout Cables

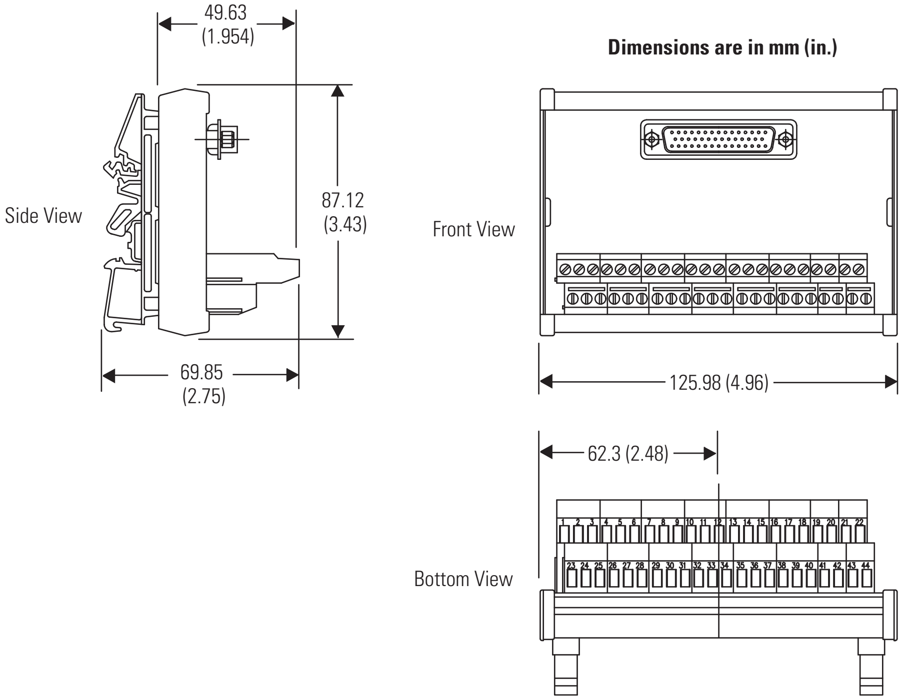
Cat. No.	Description
2090-UXBC-D15*xx*	15-pin cable with D-sub connector. Use with any Kinetix 300, Kinetix 2000, Kinetix 6000, Kinetix 6200, Kinetix 6500, or Kinetix 7000 drives (MF connector) or Ultra3000/5000 drives (CN2 connector) for motor feedback connections.
2090-U3BC-D44*xx* (1)	44-pin cable with D-sub connector. Use with Ultra3000 drives (CN1 connector) or Kinetix 2000 drives (IOD/AF connector) for control interface connections.

(1) This cable does not carry the unbuffered motor encoder signals (CN1 pins 10…15). Contact your Allen-Bradley sales representative if these signals are required for your application.

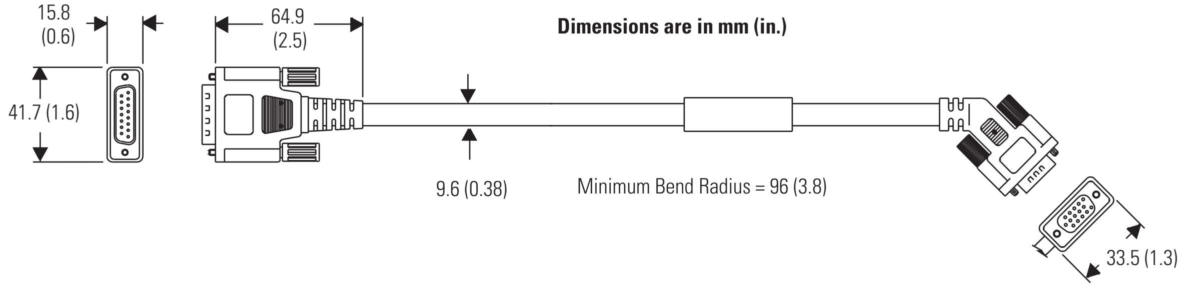
Panel-mounted Breakout Board Dimensions (catalog number 2090-UXBB-D15)



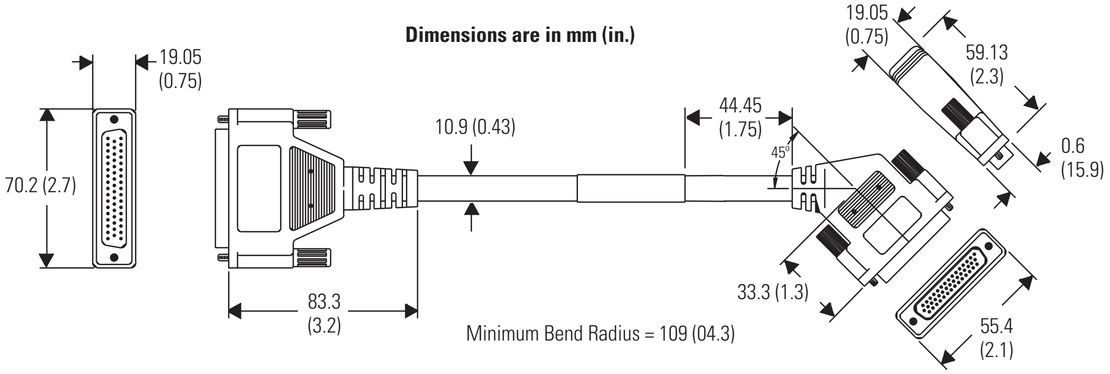
Panel-mounted Breakout Board Dimensions (catalog number 2090-U3BB-D44)



Panel-mounted Breakout Cable Dimensions (catalog number 2090-UXBC-D15*xx*)



Panel-mounted Breakout Cable Dimensions (catalog number 2090-U3BC-D44*xx*)



Panel-mounted Breakout Cable Specifications

Breakout Cable	Description	Specifications			Standard Cable Lengths m (ft)
		Rating ° C (° F)	Shield Coverage	Jacket Material	
2090-UXBC-D15*xx*	15-pin, high density D-shell for Kinetix 6000 motor/auxiliary feedback and Ultra3000/5000 CN2 feedback connector	90 ° C (194 ° F)	100% Aluminum Foil (with 85% braid overshield)	TPE	1 (3.2) 3 (9.8) 9 (29.5) 15 (49.2)
2090-U3BC-D44*xx*	44-pin, high density D-shell for Ultra3000 CN1 control interface connector				

Panel-mounted Breakout Board Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering charts below to understand the configuration of your component. For questions regarding product availability, contact your Allen-Bradley distributor.

2090 - *xx* *xx* - D*xx* *xx*

Length
Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.
Blank = N/A

Connector Type
D44 = 44-pin, male, D-sub, 44 connections
D15 = 15-pin, male, D-sub, 15 connections

Kit Type
BK = Breakout board and cable kit
BC = Breakout cable only
BB = Breakout board only

Drive
UX = Kinetix 300, Kinetix 2000, Kinetix 6000, Kinetix 6200, Kinetix 6500, Kinetix 7000, Ultra3000, and Ultra5000 drives
U3 = Ultra3000 or Kinetix 2000 drives

Bulletin Number

Drive-end Connector Kits

Drive Family	Kit Cat. No.	Description
Ultra3000/5000	2090-UXCK-D09	Mating Connector Kit (9-pin standard density D-shell) CN3
Ultra3000/5000	2090-UXCK-D15	Mating Connector Kit (15-pin high density D-shell) CN2
Ultra3000	2090-U3CK-D44	Mating Connector Kit (44-pin high density D-shell) CN1
Kinetix 3	9101-1476	Mating Connector Kit (50-pin mini-D solder cup) IOD
	9101-1477	Mating Connector Kit (20-pin mini-D solder cup) MF
Ultra5000	2090-U5CK-TB	Mating Connector Kit (28 and 15 position spring terminal) CN1A and CN1B

Bulletin 2094 Power Rail

The Bulletin 2094 power rail is compatible with Kinetix 6000, Kinetix 6200, and Kinetix 6500 drive families. This section contains selection information, mounting dimensions, and catalog numbers for the 2094-PRS*x* (slim) power rails. Bulletin 2094 power rails are compatible with all 230V and 460V drive modules.

IAM, AM, and Shunt Module Slot Requirements

IAM Module Cat. No.		Converter Slot Used	Inverter Slots Used
2094-AC05-MP5-S	230V	1	1
2094-AC05-M01-S			1
2094-AC09-M02-S			1
2094-AC16-M03-S			1
2094-AC32-M05-S			2
2094-BC01-MP5-S 2094-BC01-MP5-M	460V		1
2094-BC01-M01-S 2094-BC01-M01-M			1
2094-BC02-M02-S 2094-BC02-M02-M			1
2094-BC04-M03-S 2094-BC04-M03-M			2
2094-BC07-M05-S 2094-BC07-M05-M			2

AM Module Cat. No.		Converter Slot Used	Inverter Slots Used
2094-AMP5-S	230V	0	1
2094-AM01-S			1
2094-AM02-S			1
2094-AM03-S			1
2094-AM05-S			1
2094-BMP5-S 2094-BMP5-M	460V		1
2094-BM01-S 2094-BM01-M			1
2094-BM02-S 2094-BM02-M			1
2094-BM03-S 2094-BM03-M			2
2094-BM05-S 2094-BM05-M			2

Shunt Module Cat. No.		Converter Slot Used	Inverter Slots Used
2094-BSP2	230/ 460V	0	1

Integrated axis modules (2094-AC32-M05-S, 2094-BC04-M03-*x*, and 2094-BC07-M05-*x*) and axis modules (2094-BM03-*x* and 2094-BM05-*x*) are double-wide modules and require two slots on the power rail.

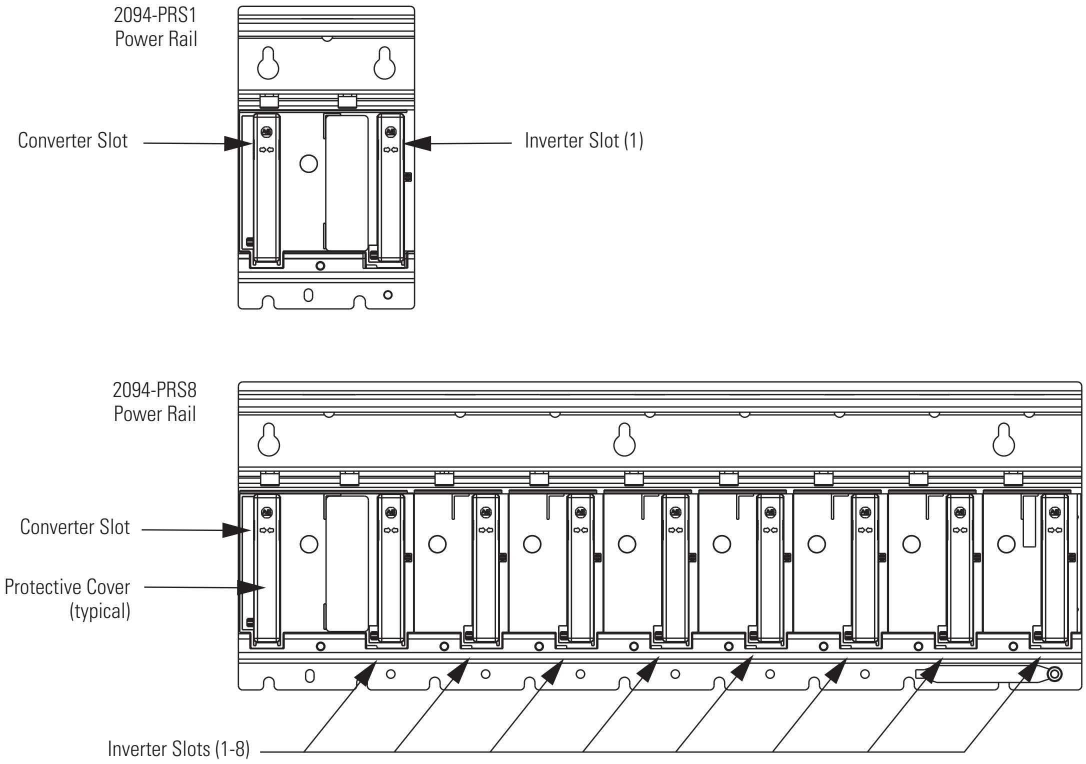
The leftmost slot on each power rail is the converter slot and only used by the IAM module. All other slots are inverter slots and are used by the IAM, AM, or shunt module (refer to the figure below). The power rail catalog number indicates the number of available inverter slots.

For example, the 2094-PRS1 power rail contains one inverter slot. This limits the use of this power rail to systems requiring only one inverter slot. Similarly, the 2094-PRS8 power rail contains eight inverter slots. This limits the use of this power rail to systems requiring up to eight inverter slots.

When selecting a power rail, determine the number of inverter slots required by all rail-mounted modules and choose a power rail with that minimum number of inverter slots.

IMPORTANT If you select a power rail with slots exceeding the minimum required for your system, you must install a 2094-PRF slot-filler module in each unused slot.

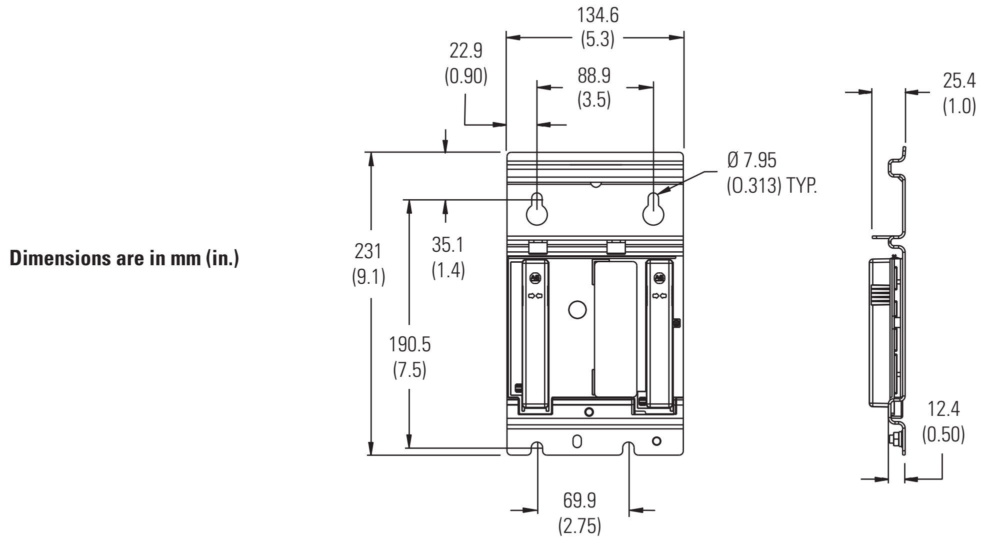
Power Rail Slots



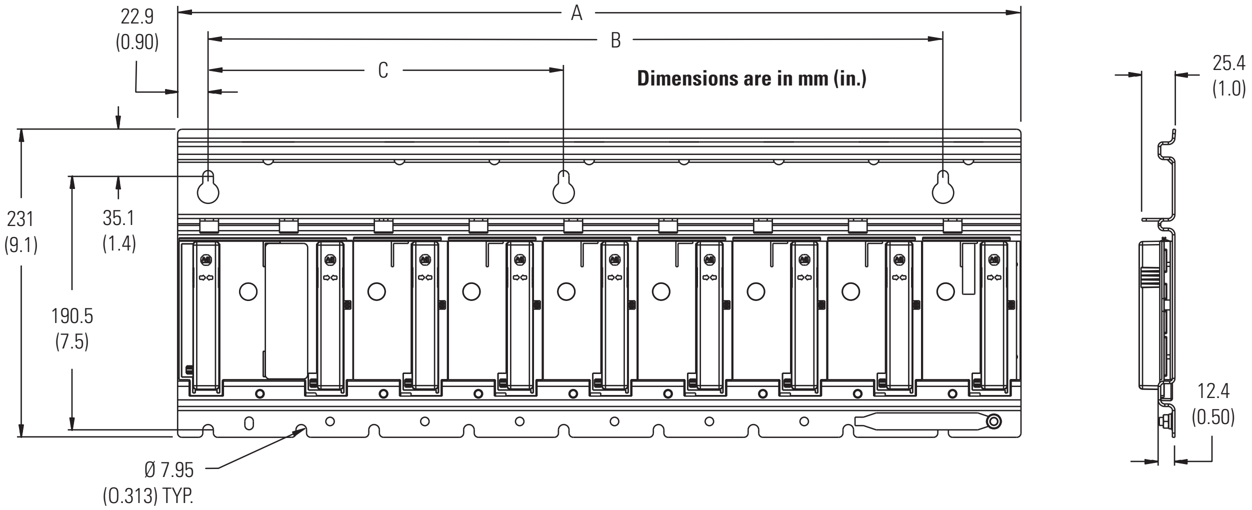
TIP The Bulletin 2094 power rails ship with a protective cover over the converter and inverter slot pins.

Power Rail Dimensions

Dimensions (catalog number 2094-PRS1)



Dimensions (catalog numbers 2094-PRS2, 2094-PRS3, 2094-PRS4, 2094-PRS5, 2094-PRS6, 2094-PRS7, and 2094-PRS8)



Power Rail Cat. No.	Description	Dimension A mm (in.)	Dimension B mm (in.)	Dimension C mm (in.)
2094-PRS1	Refer to figure on page 456.			
2094-PRS2	2 axis power rail	205.7 (8.10)	124.5 (4.90)	N/A
2094-PRS3	3 axis power rail	276.9 (10.90)	195.6 (7.70)	N/A
2094-PRS4	4 axis power rail	348.0 (13.70)	266.7 (10.50)	N/A
2094-PRS5	5 axis power rail	419.1 (16.50)	337.8 (13.30)	195.6 (7.70)
2094-PRS6	6 axis power rail	490.2 (19.30)	408.9 (16.10)	195.6 (7.70)
2094-PRS7	7 axis power rail	561.3 (22.10)	480.1 (18.90)	266.7 (10.50)
2094-PRS8	8 axis power rail	632.5 (24.90)	551.2 (21.70)	266.7 (10.50)

Power Rail Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your power rail. For questions regarding product availability, contact your Allen-Bradley distributor.

2094 - PRS *x*

Module Capacity
1 = IAM and no additional modules
2 = IAM and up to 1 additional module
3 = IAM and up to 2 additional modules
4 = IAM and up to 3 additional modules
5 = IAM and up to 4 additional modules
6 = IAM and up to 5 additional modules
7 = IAM and up to 6 additional modules
8 = IAM and up to 7 additional modules

Power Rail
PRS = Power Rail, available in 1, 2, 3, 4, 5, 6, 7, and 8 axis capacity

Bulletin Number

Bulletin 2094 Shunt Module

The Bulletin 2094 shunt module is compatible with Kinetix 6000, Kinetix 6200, and Kinetix 6500 drive families. This section contains specifications, mounting dimensions, and catalog numbers for the 2094-BSP2 shunt module.

IMPORTANT The 2094-BSP2 shunt module is compatible with all 230V and 460V systems, however, the 2094-BSP2 shunt module is physically larger than the 230V drives and additional clearance is required beneath and in front of the module.

Bulletin 2094 Shunt Module Power Specifications

The table below lists the power specifications for the Bulletin 2094 shunt module. Refer to page 458 for tables with the Bulletin 2094 shunt module in combination with an IAM module internal shunt (when present) and the various external passive shunt resistors available for 230V and 460V drive systems. Use these tables to determine the combination you need based on the requirements of your application.

Shunt Module Power Specifications

Shunt Module Cat. No.	Specifications						Short Circuit Current Rating A	Fuse Replacement
	Drive Voltage V AC	Resistance Ω	Peak Power kW	Peak Current A	Continuous Power W	Capacitance µF		
2094-BSP2	230	28.75	5.7	14	200	470	200,000 symmetrical	N/A (no internal fuse)
	460		22.5	28				

For specifications and dimensions of external shunt resistors compatible with your Kinetix 6000 or Kinetix 6200 drive, refer to External Shunt Modules beginning on page 492.

Bulletin 2094 Shunt Module (230V) System Specifications

In this table, the 230V system specifications are given for the IAM module internal shunt resistors, the 2094-BSP2 shunt module, and the Bulletin 1394 external shunt modules.

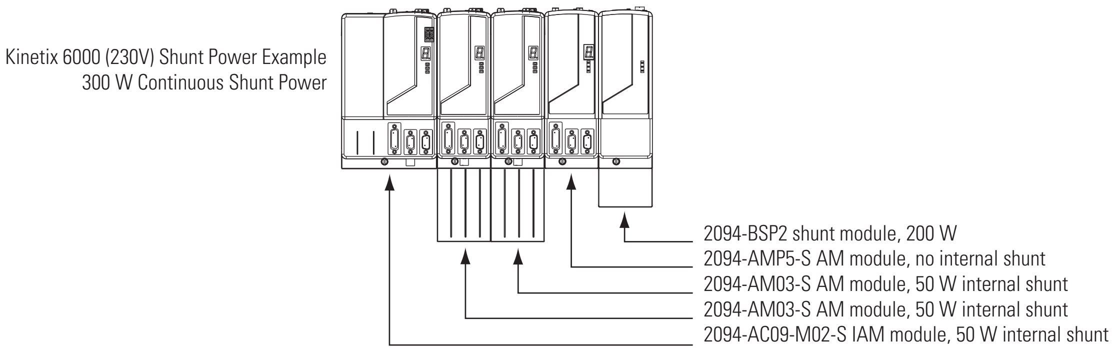
IAM Module Cat. No. 2094-	**Number of Axis Modules** Qty	**Shunt Module Cat. No.**	**Specifications**				**External Passive Shunt Module** (1)	**System Continuous Shunt Power** W
			Resistance Ω	**Peak Current** A	**Peak Power** kW	**Continuous Power** W		
AC05-MP5-S	0 to 7	N/A (2)	–	–	–	–	N/A (2)	0
AC05-M01-S			–	–	–	–		0
AC09-M02-S			–	–	–	–		50 (3)
AC16-M03-S			–	–	–	–		200 plus (4)
AC32-M05-S			–	–	–	–		
AC*xx*-M*xx*-S	0 to 6	2094-BSP2	28.75	14.1	5.7	200	N/A (2)	200 plus (5)
AC*xx*-M*xx*-S	0 to 6	2094-BSP2	4	101.3	41	300	1394-SR9A	300 (6)
AC*xx*-M*xx*-S						900	1394-SR9AF	900 (6)
AC*xx*-M*xx*-S						1800	1394-SR36A	1800 (6)
AC*xx*-M*xx*-S						3600	1394-SR36AF	3600 (6)

(1) Refer to External Shunt Modules beginning on page 492 for shunt module specifications.
(2) Module not part of system configuration.
(3) Shunt power equals 50 or the sum of the AM module internal shunt ratings.
(4) Shunt power equals 200 plus the sum of the AM module internal shunt ratings.
(5) Shunt power equals 200 plus the sum of the IAM module (2094-AC16-M03-S and 2094-AC32-M05-S only) and AM module internal shunt ratings.
(6) Use of external shunt module disables internal IAM/AM shunt modules.

IMPORTANT Use of the 2094-BSP2 shunt module in combination with the 2094-AC09-M02-S IAM module disables the shunt resistor internal to that IAM module. This situation is unique to the 2094-AC09-M02-S IAM module. Shunt resistors internal to adjacent AM modules are not disabled. Refer to the 230V Shunt Power Example (catalog number 2094-AC09-M02-S) shown below.

In this example, the continuous shunt power is 300 W. The 50 W resistor in the IAM module is disabled when used in combination with the 2094-BSP2 shunt module. This example is unique to the 2094-AC09-M02-S IAM module.

230V Shunt Power Example (catalog number 2094-AC09-M02-S)



Bulletin 2094 Shunt Module (460V) System Specifications

In this table, the 460V system specifications are given for the IAM module internal shunt resistors, the 2094-BSP2 shunt module, and the Bulletin 1394 external shunt modules.

IAM Module Cat. No. 2094-	**Number of Axis Modules** Qty	**Shunt Module Cat. No.**	**Specifications**				**External Passive Shunt Module** [1]	**System Continuous Shunt Power** W
			Resistance Ω	**Peak Current** A	**Peak Power** kW	**Continuous Power** W		
BC01-MP5-S BC01-MP5-M	0 to 7	N/A [2]	–	–	–	–	N/A [2]	50 plus [3]
BC01-M01-S BC01-M01-M			–	–	–	–		50 plus [3]
BC02-M02-S BC02-M02-M			–	–	–	–		50 plus [3]
BC04-M03-S BC04-M03-M			–	–	–	–		200 plus [4]
BC07-M05-S BC07-M05-M			–	–	–	–		
BC*xx*-M*xx*-*x*	1 to 6	2094-BSP2	28.75	28	22.5	200	N/A [2]	200 plus [5]
BC*xx*-M*xx*-*x*	1 to 6	2094-BSP2	4	201.3	162	300	1394-SR9A	300 [6]
BC*xx*-M*xx*-*x*						900	1394-SR9AF	900 [6]
BC*xx*-M*xx*-*x*						1800	1394-SR36A	1800 [6]
BC*xx*-M*xx*-*x*						3600	1394-SR36AF	3600 [6]

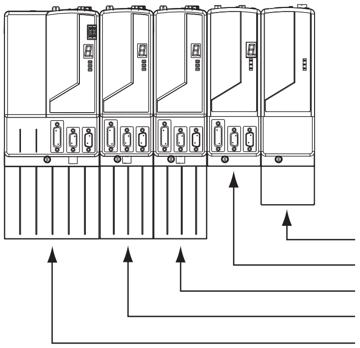
(1) Refer to External Shunt Modules beginning on page 492 for shunt module specifications.
(2) Module not part of system configuration.
(3) Shunt power equals 50 or the sum of the AM module internal shunt ratings.
(4) Shunt power equals 200 plus the sum of the AM module internal shunt ratings.
(5) Shunt power equals 200 plus the sum of the IAM and AM module internal shunt ratings.
(6) Use of external shunt module disables internal IAM/AM shunt modules.

Bulletin 2094 Shunt Power Examples

In this example, the sum of the IAM, AM, and shunt modules equal 500 W of continuous shunt power. Although a 230V system is shown, a 460V IAM, AM, and shunt module power adds up the same way.

Shunt Power Example for (230V) Kinetix 6000 Drives (without external shunt)

Shunt Power Example
500 W Continuous Shunt Power
Kinetix 6000 (230V) system is shown.



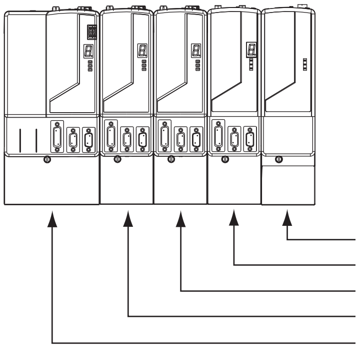
2094-BSP2 shunt module, 200 W
2094-AM02-S AM module, no internal shunt
2094-AM03-S AM module, 50 W internal shunt
2094-AM03-S AM module, 50 W internal shunt
2094-AC16-M03-S IAM module, 200 W internal

In this example, the sum of the IAM/AM modules and shunt module equal 400 W of continuous shunt power.

TIP Shunt power adds up the same way for 230V (IAM/AM, and shunt module) systems too.

Shunt Power Example for Kinetix 6000 Drives (without external shunt)

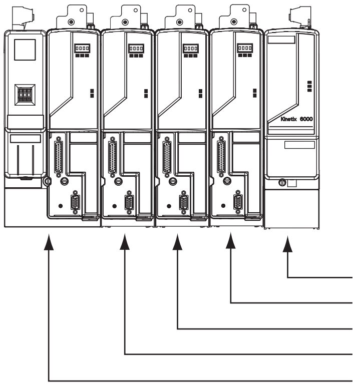
Kinetix 6000 (460V) Shunt Power Example
400 W Continuous Shunt Power



2094-BSP2 shunt module, 200 W
2094-BMP5-S AM module, 50 W internal shunt
2094-BM01-S AM module, 50 W internal shunt
2094-BM01-S AM module, 50 W internal shunt
2094-BC02-M02-S IAM module, 50 W internal shunt

Shunt Power Example for Kinetix 6200 or Kinetix 6500 Drives (without external shunt)

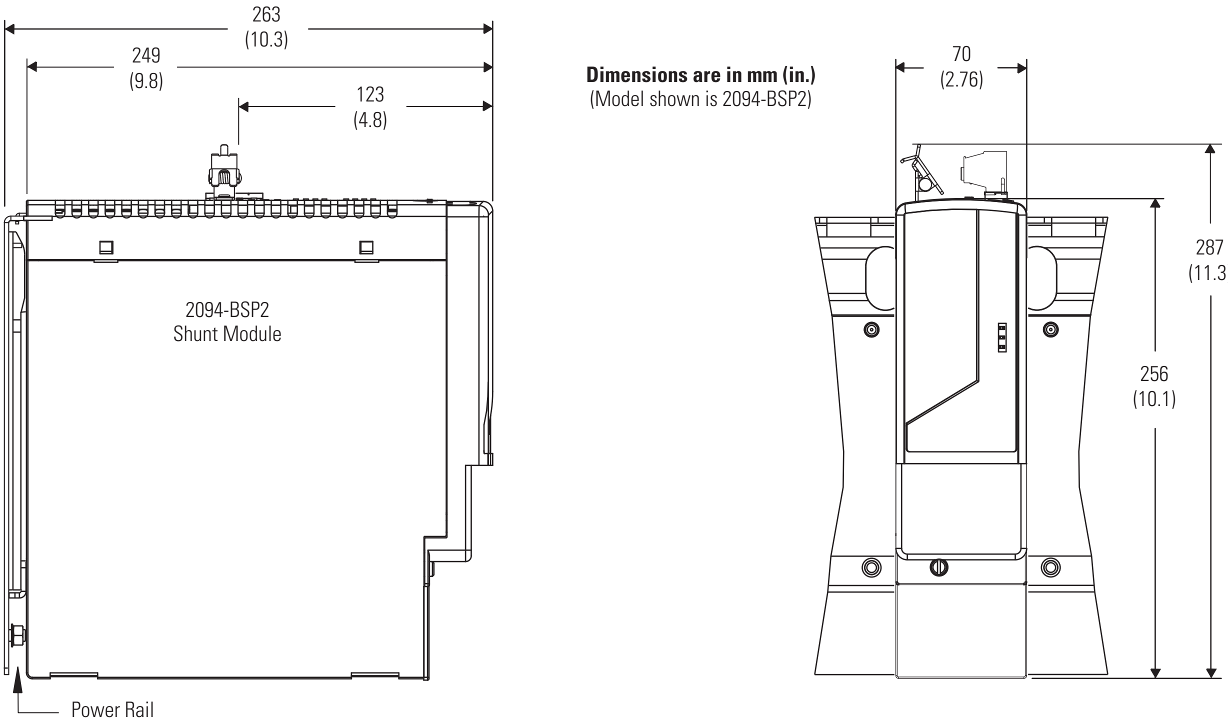
Kinetix 6200/Kinetix 6500 (460V)
Shunt Power Example
400 W Continuous Shunt Power



2094-BSP2 shunt module, 200 W
2094-BMP5-M AM power module, 50 W internal shunt
2094-BM01-M AM power module, 50 W internal shunt
2094-BM01-M AM power Module, 50 W internal shunt
2094-BC02-M02-M IAM power module, 50 W internal shunt

IMPORTANT When the 2094-BSP2 shunt module is wired to a Bulletin 1394 external shunt module, the IAM/AM (internal shunt) and 2094-BSP2 shunt module is disabled and the continuous shunt power is equal to that of the external shunt module alone.

Shunt Module Dimensions

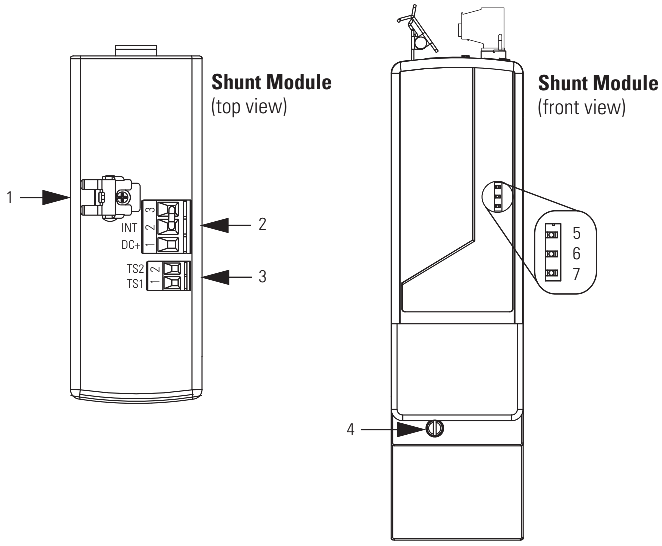


Dimensions are in mm (in.)
(Model shown is 2094-BSP2)

Modules are shown mounted to the power rail and the dimensions reflect that.

Bulletin 2094 Shunt Module Connectors and Indicators

Shunt Module Connectors (catalog number 2094-BSP2)



Item	Description
1	Motor cable shield clamp
2	External shunt resistor (RC) connector
3	External thermal switch (TS) connector
4	Mounting screw
5	Shunt fault status indicator
6	Over-temp fault status indicator
7	Bus status indicator

For replacement connector set catalog number, refer to Connector Sets on page 463.

Shunt Module Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your module. For questions regarding product availability, contact your Allen-Bradley distributor.

2094 - BSP2

Input Voltage/kW Rating
BSP2 = 230/460V input voltage, 200 W

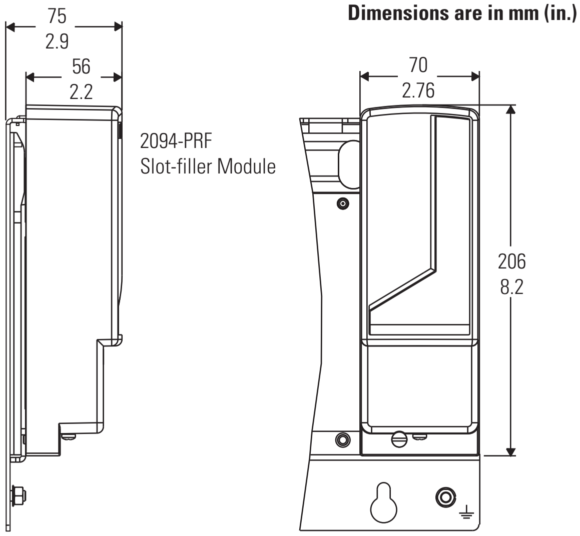
Bulletin Number

Bulletin 2094 Slot-filler Module

The Bulletin 2094 slot-filler module is compatible with Kinetix 6000, Kinetix 6200, and Kinetix 6500 drive families. This section contains dimensions and catalog numbers for the 2094-PRF slot-filler module.

IMPORTANT The 2094-PRF slot-filler module is compatible with all 230V and 460V systems. Power rail slots not occupied by an IAM, AM, or shunt module, must have a slot-filler module installed.

Slot-filler Module Dimensions



Slot-filler Module Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your module. For questions regarding product availability, contact your Allen-Bradley distributor.

2094 - PRF

Power Rail Filler

Bulletin Number

Connector Sets

Kinetix 3 Drive Family

Drive Cat. No.	Description	Cat. No.
2071-A*xx*	Includes general purpose input power (IPD), shunt resistor (BC), and motor power (MP) replacement connectors for Kinetix 3 drives.	2071-CONN1

Kinetix 300 Drive Family

Drive Cat. No.	Description	Cat. No.
2097-V3*xxxx*	Includes AC input power (IPD), back-up power (CPD), shunt and DC bus (BC), motor power (MP), and safe-off (STO) replacement connectors for Kinetix 300 drives.	2097-CONN1

Kinetix 6000, Kinetix 6200, Kinetix 6500 Drive Families

Module	Cat. No.	Description	Cat. No.
IAM Module (converter connectors)	2094-AC05-M*xx*-S and 2094-AC09-M02-S	Includes control power (CPD), DC bus/AC input (IPD), and contactor enable (CED) replacement connectors for the IAM (converter) module.	2094-ANCON-1
	2094-AC16-M03-S and 2094-AC32-M05-S		2094-XNCON-2
	2094-BC01-M*xx*-S and 2094-BC02-M02-S 2094-BC01-M*xx*-M and 2094-BC02-M02-M		2094-BNCON-1
	2094-BC04-M03-S and 2094-BC07-M05-S 2094-BC04-M03-M and 2094-BC07-M05-M		2094-XNCON-2
IAM/AM Module (inverter connectors)	2094-AC05-M*xx*-S, 2094-AC09-M02-S, 2094-AMP5-S, 2094-AM01-S, 2094-AM02-S	Includes motor power (MP), motor/resistive brake (BC), and safe-off (SO), replacement connectors for the IAM and AM (inverter) modules. Also includes bracket kit for SERCOS fiber-optic cable connectors.	2094-XNINV-1
	2094-AC16-M03-S and 2094-AC32-M05-S 2094-AM03-S, 2094-AM05-S 2094-BC04-M03-S, 2094-BM03-S		2094-ANINV-2
	2094-BC01-M*xx*-S and 2094-BC02-M02-S 2094-BC01-M*xx*-S and 2094-BC02-M02-M 2094-BMP5-M, 2094-BM01-M, 2094-BM02-S 2094-BMP5-M, 2094-BM01-M, 2094-BM02-M		2094-XNINV-1
	2094-BC07-M05-S, 2094-BM05-S 2094-BC07-M05-M, 2094-BM05-M		2094-BNINV-2
Shunt Module	2094-BSP2	Includes external shunt resistor (RC) and external thermal switch (TS) replacement connectors for the shunt module.	2094-XNSHT-1

Kinetix 7000 Drive Family

Drive Cat. No.	Description	Cat. No.
2099-BM*xx*-S	Includes safe-off (SO), general purpose I/O (GPIO), general purpose relay (GPR), and control power (CP) replacement connectors for Kinetix 7000 drives.	2099-K7KCK-1

Line Interface Module (LIM) and Resistive Brake Module (RBM)

Module	Cat. No.	Description	Cat. No.
Line Interface Module (LIM)	2094-AL09 and 2094-BL02	Includes VAC line (IPL), VAC load (OPL), control power (CPL), and 24V brake power (PSL) replacement connectors.	2094-XNLIM-1
	2094-AL*xx*S, 2094-BL*xx*S, and 2094-XL75S-C*x*	Includes I/O (IOL), VAC line (IPL), VAC load (OPL), control power (CPL), 230V auxiliary output (P2L), 24V brake power (P1L), and 230V auxiliary input (APL) replacement connectors.	2094-XNLIM-2
Resistive Brake Module (RBM)	2090-XB33-*xx*	Includes I/O connector (TB3), drive connector (TB1), and motor connector (TB2).	2090-XNRBM-1
	2090-XB120-*xx*	Includes I/O connector (TB3), 230V input power connector (TB4), drive connector (TB1), and motor connector (TB2).	2090-XNRBM-2

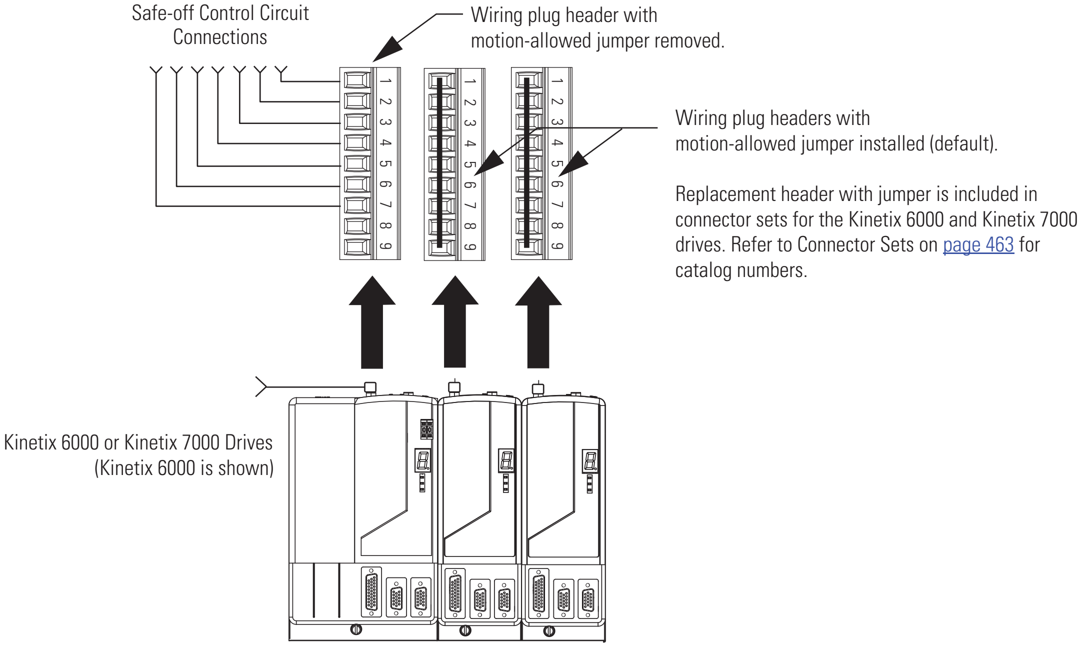
Kinetix Safe-off Headers

The safe-off feature is available with Kinetix 6000 and Kinetix 7000 drives. You can implement the safe-off function in a single drive or extended to as many as eight drives in a multiple safety drive configuration. The safe-off (SO) connector can also be jumpered to effectively remove the safe-off function (this is the default setting). For more information on wiring the safe-off connector, refer to the Kinetix Safe-off Feature Safety Reference Manual, publication GMC-RM002.

Safe-off Header Examples

In this example, a single Kinetix 6000 safe-off drive is shown using the wiring plug header. The second and third drives do not use the safe-off feature, so the motion-allowed jumpers remain installed. This single drive configuration also applies to Kinetix 7000 safe-off applications.

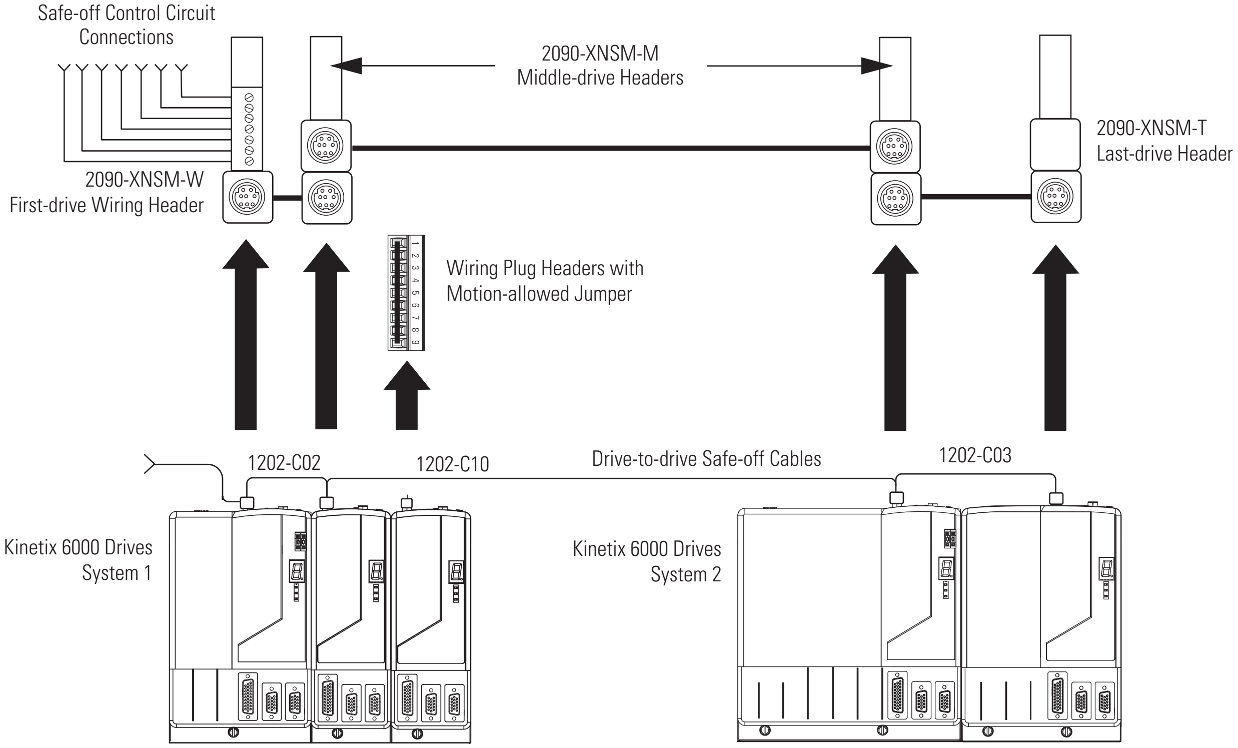
Typical Single Drive Safe-off Configuration



In this example, system 1 contains two (single-wide) Kinetix 6000 drives using the safe-off feature wired with two (double-wide) Kinetix 6000 drives in system 2. The wiring headers with motion allowed jumpers have been replaced as shown. The third axis in system 1 does not use the safe-off feature, so the wiring header and motion allowed jumper remain installed.

This multiple drive configuration also applies to the Kinetix 7000 drives. When wiring the Kinetix 7000 modules from drive-to-drive, use 1202-C10 cables.

Typical Multiple Drive Safe-off Configuration



IMPORTANT	Due to the current capacity limitation of the safe-off cable connectors, multiple safe-off drive configurations must not exceed eight Kinetix 6000 or Kinetix 7000 drive modules.

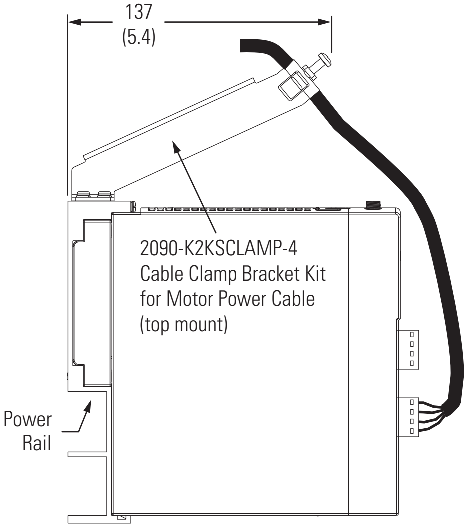
Safe-off Header Components

Description	Cat. No.
Safe-off wiring header for the first drive in multiple safety drive configurations.	2090-XNSM-W
Safe-off middle header for drive-to-drive connections in multiple safety drive configurations with three or more drives.	2090-XNSM-M
Safe-off terminating header for the last drive in multiple safety drive configurations.	2090-XNSM-T

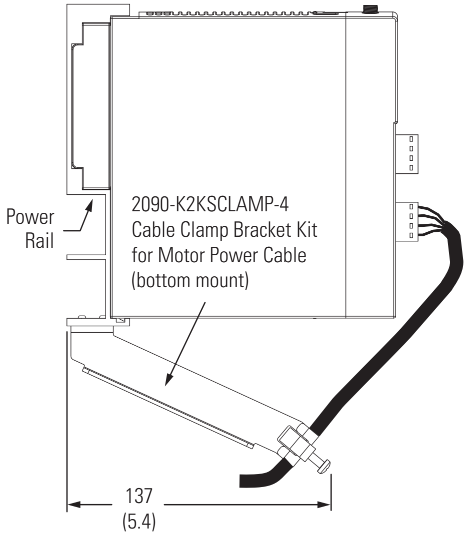
Kinetix 2000 Cable Clamp Bracket Kit

The cable clamp bracket kit (catalog number 2090-K2KSCLAMP-4) is designed for use with the Kinetix 2000 IAM and AM drive modules. The clamp mounts to the power rail and provides stress relief for the motor power cable and an electrical path from the cable shield to machine ground. You can mount the bracket to the top or bottom of the power rail, depending on the layout of cables within your panel.

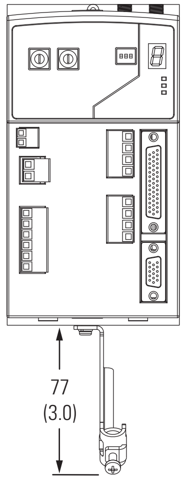
Cable Clamp Bracket Kit (catalog number 2090-K2KSCLAMP-4)



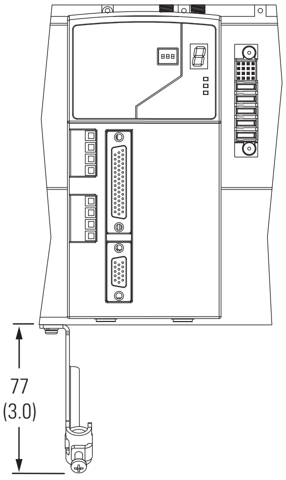
Dimensions are in mm (in.)



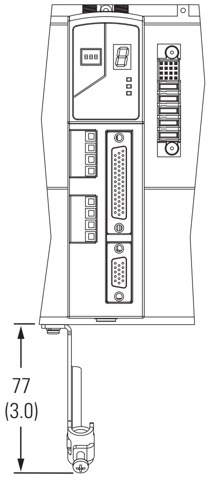
Kinetix 2000
Integrated Axis Modules
2093-AC05-MP1
2093-AC05-MP2
2093-AC05-MP5



Kinetix 2000
Axis Modules
2093-AM01
2093-AM02



Kinetix 2000
Axis Modules
2093-AMP1
2093-AMP2
2093-AMP5



IMPORTANT: Additional clearance below the connector is necessary to provide the recommended cable bend radius.

Bulletin 2094 Mounting Brackets

The Bulletin 2094 mounting brackets (catalog number 2094-XNBRKT-1) are designed to save panel space by letting you mount the Bulletin 2094 power rail or line interface module (LIM) over the AC line filter.

Each bracket provides threaded holes for mounting the 2094 power rail or LIM (catalog numbers 2094-AL*xx*S and 2094-XL75S-C*x*). The number of brackets required for use with the power rail and LIM are shown in the table below.

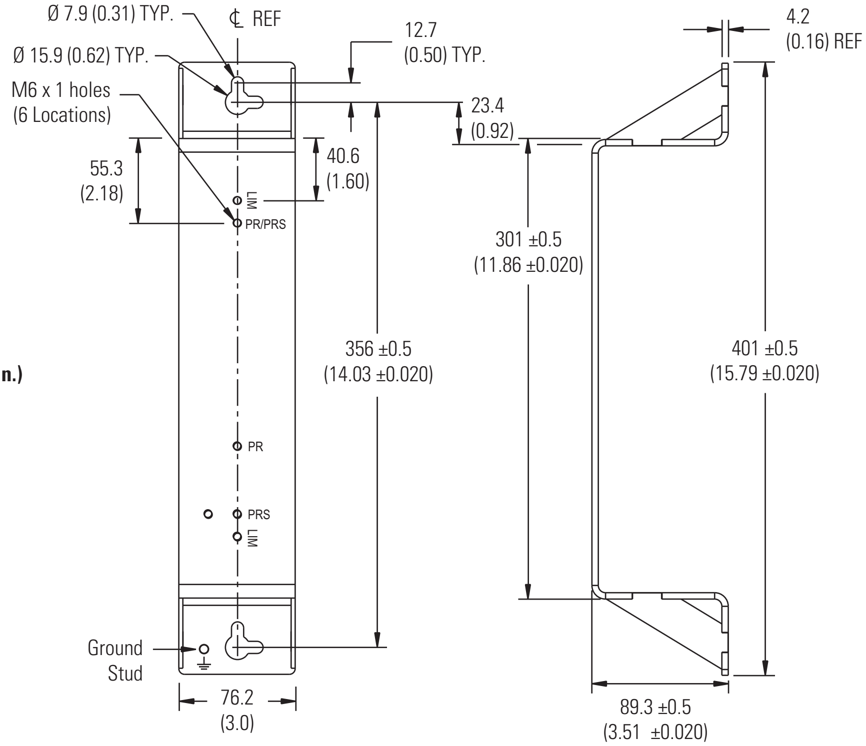
Module to Mount	Brackets Required
LIM (2094-AL*xx*S and -XL75S-C*x*)	2
Power Rail (PRS*x*) 1-4 axis	2
Power Rail (PRS*x*) 5-8 axis	3

IMPORTANT The 2094-BL*xx*S, 2094-AL09, and 2094-BL02 LIM models are not compatible with the 2094 mounting brackets.

Mounting Bracket Dimensions

The mounting bracket dimensions are shown in the figure below. Additional mounting dimensions for applications when brackets are used with the LIM, are shown on page 468.

Dimensions (catalog number 2094-XNBRKT-1)

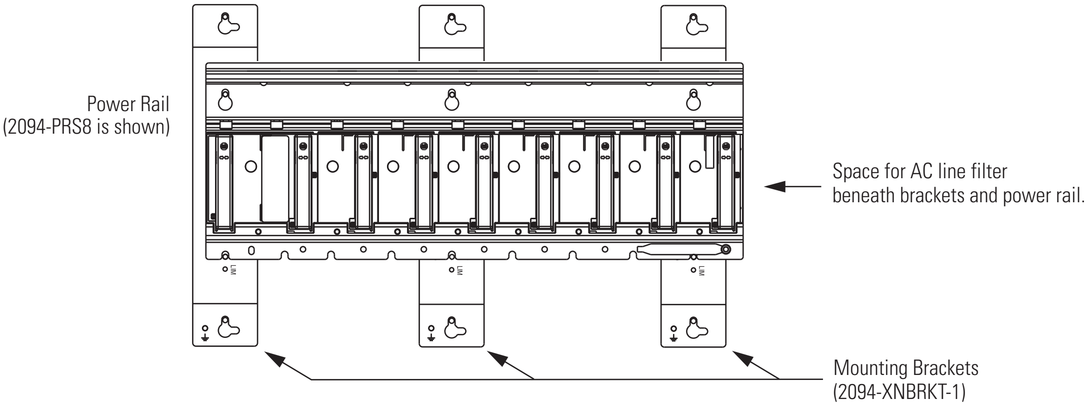


Dimensions are in mm (in.)

Mounting Bracket Configurations

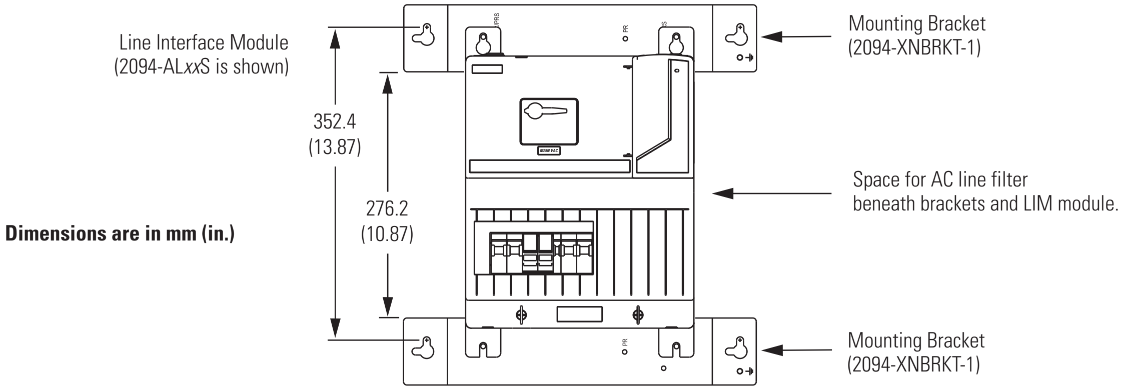
In the figure below, the power rail (catalog number 2094-PRS*x*) is shown mounted on Bulletin 2094 Mounting Brackets.

Power Rail on 2094 Mounting Brackets



In the figure below, the LIM module (catalog numbers 2094-AL*xx*S or 2094-XL75S-C*x*) is shown mounted on Bulletin 2094 Mounting Brackets.

LIM Module on 2094 Mounting Brackets



IMPORTANT Only the 2094-AL*xx*S and 2094-XL75S-C*x* Line Interface Modules are compatible with the 2094 Mounting Brackets. The 2094-BL*xx*S, 2094-AL09, and 2094-BL02 models are not compatible.

Kinetix 7000 DC-DC Converter and Control Board Kits

Cat. No.	Description
2099-K7KCB-1	Control board assembly kit. Replacement control board for 2099-BM*xx*-S drives.
2099-K7KCP-1	DC-DC converter cassette kit. Replacement DC-DC converter for 2099-BM*xx*-S drives.

External Auxiliary Encoders

These Allen-Bradley sine/cosine and incremental encoders are suitable for use when auxiliary feedback connections are required for your servo drive application.

Bulletin 842HR Sine/Cosine Encoders

Bulletin 842HR sine/cosine encoders combine the advantages of incremental and absolute encoder technologies in a single, standalone unit targeted for high-performance digital servo drive systems. The Bulletin 842HR is a 15-bit encoder featuring a hybrid digital/analog interface, transmitting sine/cosine signals via analog channels for incremental feedback and delivering absolute position information through the digital RS-485 channel.

Bulletin 842HR, Size 25, Sine/Cosine Encoder Specifications

Cat. No.	Description	Features
842HR-MJDZ115FWYD	• Square flange • 3/8 in. solid-shaft • 17-pin connector • 5…12V operating voltage	• Absolute feedback for position control • RS-485 interface • Hiperface interface compatible • IP66 (IEC 529)
842HR-SJDZ115FWYD		

Refer to 842HR Sine Cosine/Serial Encoders catalog, publication C116-CA600, for more information.

Bulletin 844D Incremental Encoders

Bulletin 844D through-shaft incremental encoders are used to electronically monitor the position or speed of a rotating shaft. Shaft position is converted to digital pulses in an A quad B format. A Zero Index Channel is also included with all models.

Bulletin 844D Hollow Shaft (HS35 Style) Encoder Specifications

Cat. No.	Resolution	Description
844D-B5CC1FW	1024 PPR	• 5/8 in. through-shaft • 3/8 in. bolt on 2.5…4.0 in. diameter radius • 10-pin connector • 5V DC input /5V DC DLD output (3487)
844D-B5CC1CS	2048 PPR	
844D-B5CC1DR	5000 PPR	

Refer to 844D Hollow Shaft Incremental Encoders catalog, publication 844D-CA500, for more information.

Bulletin 845H Incremental Encoders

The Bulletin 845H optical incremental encoders electronically digitize shaft motion of a rotating element by converting mechanical motion to an electronic digital format. Incremental square waves are accumulated in a counter as position feedback. The encoder provides code disk resolutions up to 5000 pulses per revolution at a signal frequency response of 210 kHz.

The Bulletin 845H encoder is housed in a size 25, NEMA Type 4 and 13, IP66 (IEC 529), enclosure making it suitable for many of today's industrial environments.

Bulletin 845H, Size 25, High Performance, Encoder Specifications

Cat. No.	Resolution	Description
845H-SJDN14FWY2	1024 PPR	• Square flange • 3/8 in. w/flat shaft • 5V DC input /5V DC DLD RS-422 output • Radial connector (side)
845H-SJDN14CSY2	2048 PPR	
845H-SJDN14DRY2	5000 PPR	

Refer to 845H Size 25 Incremental Encoders catalog, publication 845H-CA500, for more information.

Bulletin 845T Incremental Encoders

Bulletin 845T optical incremental encoders are used to electronically monitor the position of a rotating shaft. Shaft motion is converted to digital pulses that are accumulated and evaluated by various electronic controllers. The Bulletin 845T encoder provides code disk resolutions of up to 3000 pulses per revolution, and a frequency response of up to 100 kHz.

The Bulletin 845T encoder is a heavy duty, NEMA Type 4, and IP66 (IEC 529) rated optical incremental shaft encoder that is housed in a two inch diameter enclosure. Typical applications for the 845T include machine tools, packaging machinery, motion controls, and robotics. The heavy duty bearing assembly, rugged construction and high shaft loading capabilities make the Bulletin 845T encoder suitable for many of today's harsh industrial environments.

Bulletin 845T, Size 20, Heavy Duty Encoder Specifications

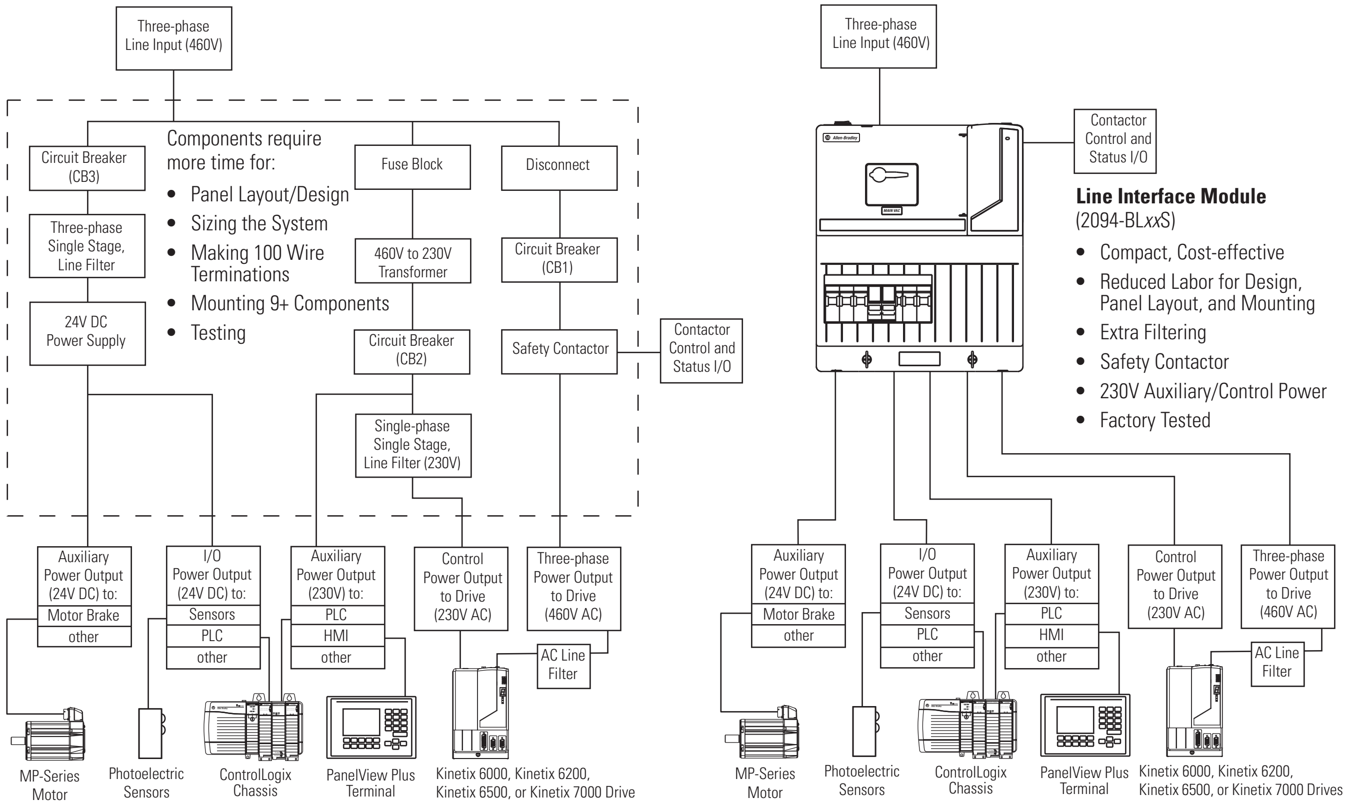
Cat. No.	Resolution	Description
845T-DN13EFW	1024 PPR	• Square flange • 3/8 in. w/flat shaft • 5V DC input /5V DC DLD output • Channel A, B, and Z signals • 10-pin connector
845T-DN13ECS	2048 PPR	

Refer to 845T Size 20 Incremental Encoders catalog, publication 845T-CA500, for more information.

Line Interface Modules

The Bulletin 2094 Line Interface Module (LIM) is designed to replace many of the common input power devices required for your servo drive system. Using the LIM module saves panel space and reduces the amount of wiring when compared with individual components mounted separately. In this example, the 2094-BL*xx*S module is compared to a similar configuration of discrete components. Auxiliary and control power (230V) is developed from the LIM module three-phase input power.

Comparing the LIM Module with Discrete Components (catalog number 2094-BL*xx*S)



An example comparing the 2094-AL*xx*S module to discrete components would be similar to the example above, but with 230V three-phase input power and without the 460V to 230V step-down transformer.

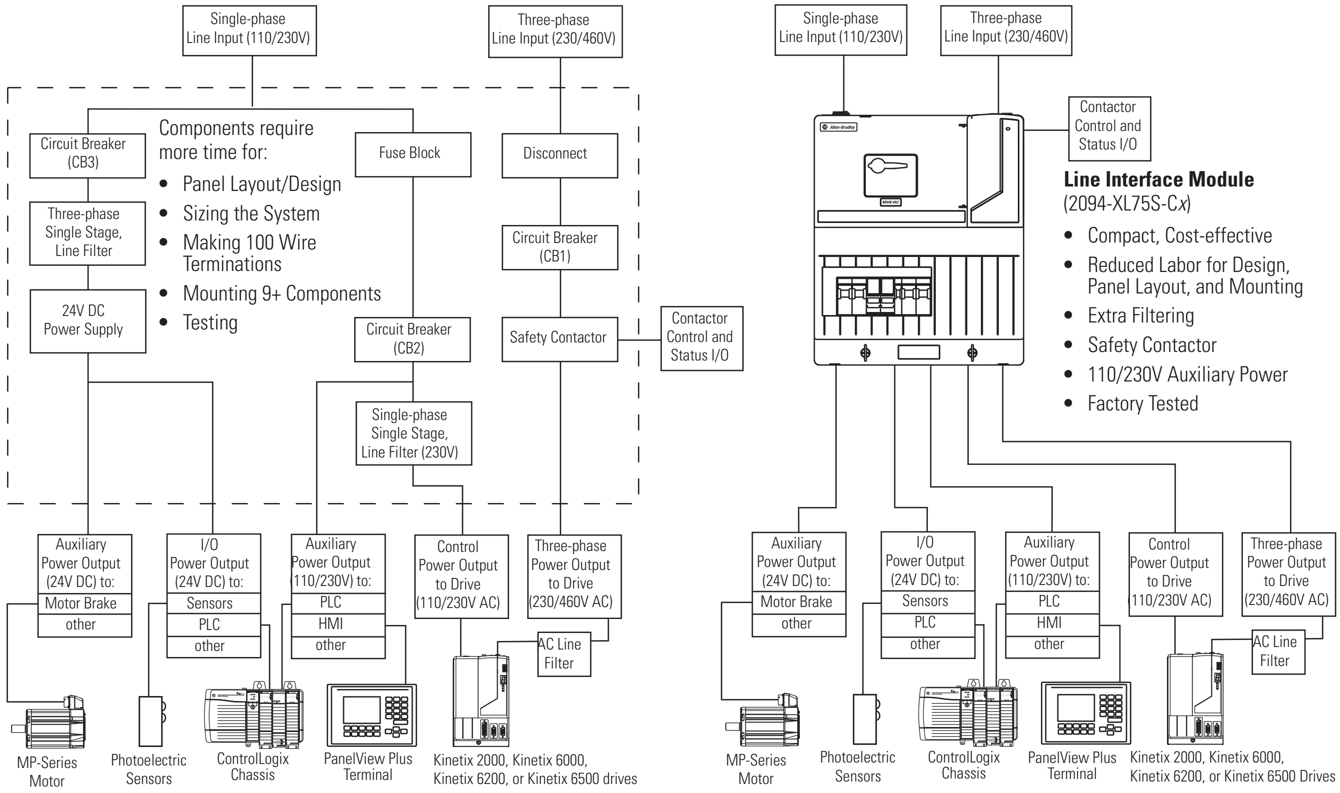
Examples comparing the 2094-AL09 and 2094-BL02 modules to discrete components would also be similar to the examples above and include the features that apply to those models.

Drive/LIM Module Compatibility

Drive Family	Drive Cat. No.	Compatible LIM Module Cat. No.
Kinetix 2000	2093-AC05-M*xx*	2094-AL*xx*S, 2094-XL75S-C2, 2094-AL09
Kinetix 6200/ Kinetix 6500	2094-BC*xx*-M*xx*-M	2094-BL*xx*S, 2094-BL02, 2094-XL75S-C*x*
Kinetix 6000	2094-*x*C*xx*-M*xx*-S	2094-AL*xx*S, 2094-AL09, 2094-BL*xx*S, 2094-BL02, 2094-XL75S-C*x*
Kinetix 7000	2099-BM06, 2099-BM07, 2099-BM08	2094-BL75S

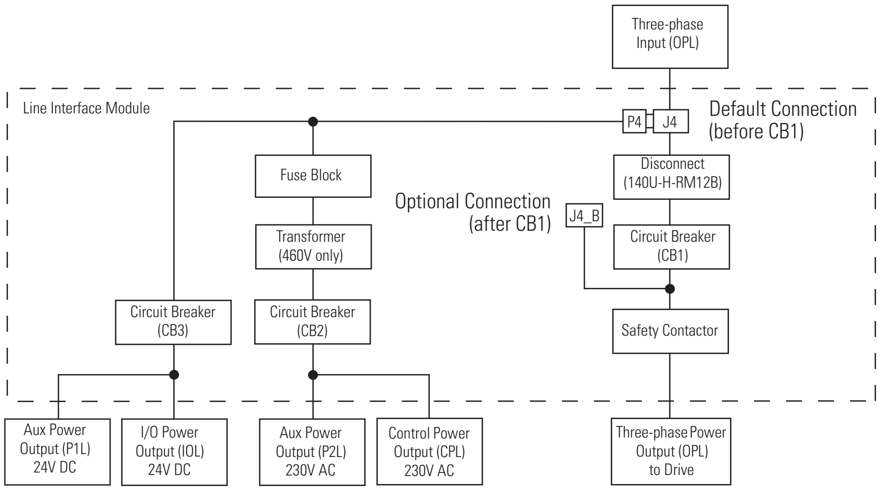
In this example, the 2094-XL75S-C1 and 2094-XL75S-C2 modules are compared to a similar configuration of discrete components. Both of these units provide a connector for an external (customer-supplied) auxiliary and control power input. The 2094-XL75S-C1 module is designed for 110V auxiliary and control power input/output. The 2094-XL75S-C2 module is designed for 230V auxiliary and control power input/output.

Comparing the LIM Module with Discrete Components (catalog number 2094-XL75S-C*x*)



Branch circuit protection for the 2094-AL*xx*S and 2094-BL*xx*S modules is customer configurable. By moving the CB2/CB3 wiring harness (P4) from one side of CB1 to the other, you can change the module operation. To understand this option, refer to the simplified block diagram below.

Simplified Block Diagram (catalog numbers 2094-AL*xx*S and 2094-BL*xx*S)



Features

Features available with the Line Interface Module (LIM) include the following:

- Power production for drive, motor brakes, and auxiliary use.
  - Branch circuit protection and disconnect for three-phase power is provided by an Allen-Bradley Bulletin 140U molded case circuit breaker. The 140U includes both magnetic and thermal protection, eliminating the need for fuses on the three-phase line. Variable Depth Rotary Mechanism (140U-H-RM12B) is required for disconnect feature.
  - Customer configurable branch circuit protection (catalog numbers 2094-AL*xx*S and 2094-BL*xx*S) provides the option of using the Bulletin 140U circuit breaker to control all output power (optional) or only three-phase output power (default).
  - Three-phase (230V or 460V) output power to drive controlled by a safety power contactor.
  - 24V DC supply rated at 20 A (catalog numbers 2094-AL*xx*S, 2094-BL*xx*S, and 2094-XL75S-C*x*) or 8 A (catalog numbers 2094-AL09 and 2094-BL02).
  - Single-phase auxiliary power and control power sourced from internal three-phase input power or external (customer-supplied) input power supply.
- Provides power to single or multiple Kinetix 2000 or Kinetix 6000 power rails. Cumulative IAM input current must not exceed LIM module output current (applies to catalog numbers 2094-AL*xx*S, 2094-BL*xx*S, and 2094-XL75S-C*x* only).
- Internal line filter (catalog numbers 2094-AL09 and 2094-BL02). External (customer-supplied) line filter (catalog numbers 2094-AL*xx*S, 2094-BL*xx*S, and 2094-XL75S-C*x).*
- 2094 mounting bracket compatibility for additional panel space saving. Using mounting brackets (catalog number 2094-XNBRKT-1) in your system (2094-AL*xx*S and 2094-XL75S-C*x* modules only) lets you mount the AC line filter behind the LIM module. Refer to Bulletin 2094 Mounting Brackets on page 467 for specifications.
- Plugable connectors let you remove and replace each connector for easy wiring.

Line Interface Module Selection

This table provides a summary of the features available with each Line Interface Module. Use this table and Line Interface Module Selection Flowchart on page 475 to select a LIM module for your drive system.

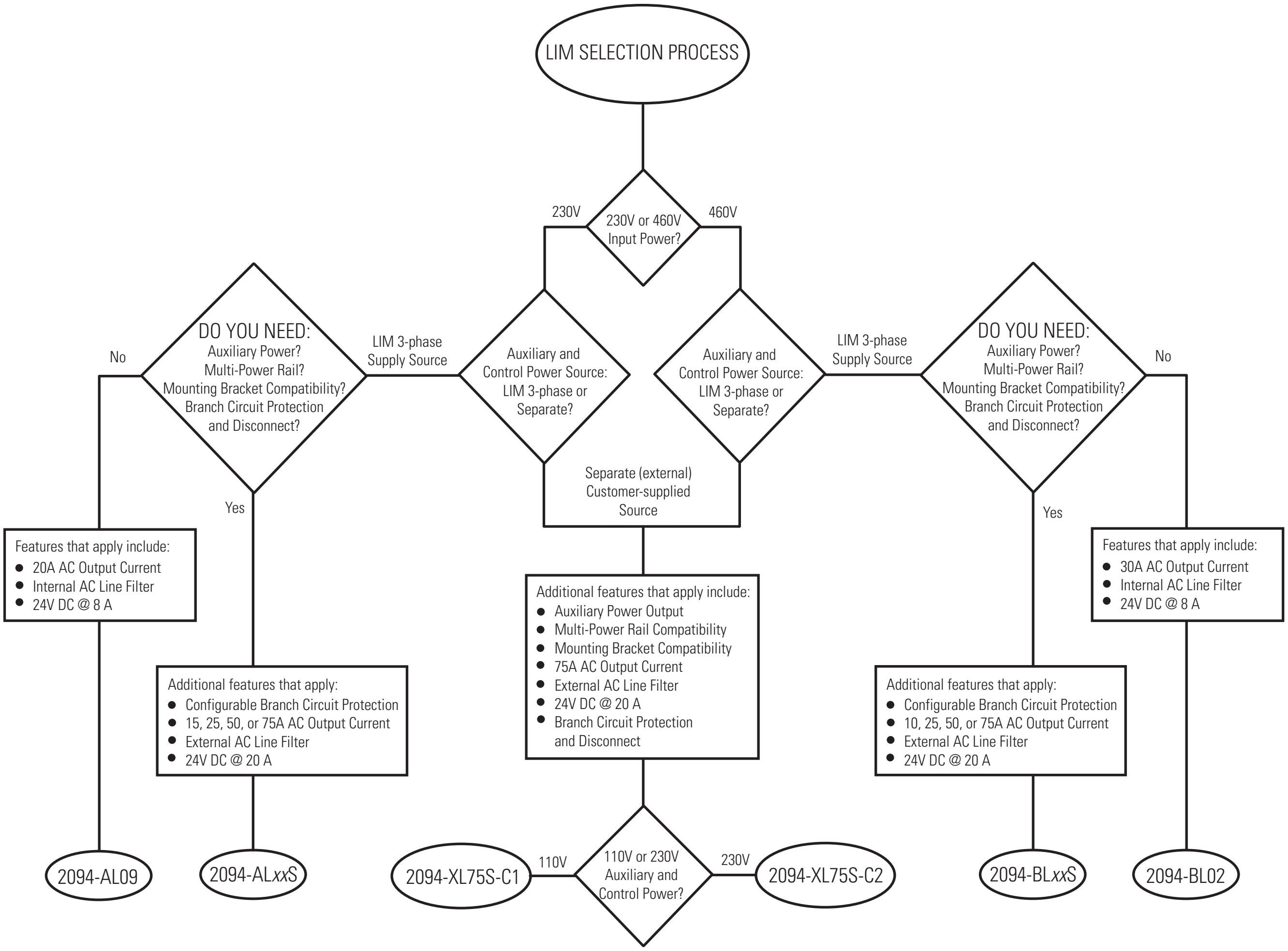
Features	**Line Interface Module Cat. No.**					
	2094-AL*xx*S	**2094-BL*xx*S**	**2094-XL75S-C1**	**2094-XL75S-C2**	**2094-AL09**	**2094-BL02**
Input Power	230V	460V	230V or 460V		230V	460V
Auxiliary and Control Power Input (customer-supplied)	N/A		110V	230V	N/A	
Auxiliary Power Output	230V (1)				N/A	
Control Power Output	230V				230V	
24V DC Power Output Current Capacity	20 A				8 A	
Configurable Branch Circuit Protection	Yes	Yes	No	No	No	No
Branch Circuit Protection and Disconnect			Yes	Yes		
Power to Multiple Power Rails (2)						
DC Line Filter						
Mounting Bracket Compatibility		No				
AC Line Filter	External (customer-supplied)				Internal	Internal
Auxiliary/Control Power Line Filter	Yes					

(1) Auxiliary power output developed internal to LIM module.

(2) For more information on powering multiple power rails from one Line Interface Module, refer to Rockwell Automation technical support.

Use this flowchart to select a LIM module for your drive system based on the input voltage and features you need.

Line Interface Module Selection Flowchart



Line Interface Module Specifications

Power Specifications (catalog numbers 2094-AL*xx*S and 2094-BL*xx*S)

Designators	**Attribute**	**2094-AL*xx*S (230V)**				**2094-BL*xx*S (460V)**			
		AL15S	**AL25S**	**AL50S**	**AL75S**	**BL10S**	**BL25S**	**BL50S**	**BL75S**
VAC Line (IPL) Connector	AC input voltage	195…265V rms three-phase (230V nom)				380…520V rms three-phase (460V nom)			
	AC input frequency	47…63 Hz				47…63 Hz			
VAC Load (OPL) Connector	Main AC output current (rms)	15 A	25 A	50 A	75 A	10 A	25 A	50 A	75 A
Control Power Output (CPL) Connector and Auxiliary Power Output (P2L) Connector	AC output current (rms)	3 A (1)				3 A (1)			
	AC output voltage	195…265V rms single-phase (230V nom)				190…260V rms single-phase (230V nom)			
Brake Power Output (P1L) Connector and I/O (IOL) Connector	24V DC Power Supply	20 A (2)				20 A (2)			
Contactor (CR1)	Contactor control voltage (3)	21.6…26.4V DC				21.6…26.4V DC			
	Contactor control current (3)	12…9 mA				12…9 mA			
	Contactor pickup current (4)	N/A (Internal)				N/A (Internal)			
	Contactor hold-in current (4)	N/A (Internal)				N/A (Internal)			
	Contactor voltage	N/A (Internal)				N/A (Internal)			
	Contactor pickup time	18.5 ms (min) 30.0 ms (max)				18.5 ms (min) 30.0 ms (max)			
	Contactor dropout time	10.0 ms (min) 60.0 ms (max)				10.0 ms (min) 60.0 ms (max)			

(1) Sum of CPL and P2L current must not exceed 3 A.
(2) Sum of P1L and IOL current must not exceed 20 A.
(3) Power specifications for DC Interface Module (100 JE) COIL_E1 and COIL_E2 input.
(4) Current provided by auxiliary VAC input.

Power Specifications (catalog number 2094-XL75S-C*x*)

Designators	Attribute	2094-XL75S-C1 (230/460V)	2094-XL75S-C2 (230/460V)
VAC Line (IPL) Connector	AC input voltage	195…520V rms three-phase (230…460V nom)	
	AC input frequency	47…63 Hz	
VAC Load (OPL) Connector	Main AC output current (rms)	75 A	
Auxiliary Power Input (APL) Connector	Auxiliary AC input voltage	93…121V rms single-phase (110V nom)	196…253V rms single-phase (230V nom)
	Auxiliary AC input current (rms)	20 A	11 A
Control Power Output (CPL) Connector and Auxiliary Power Output (P2L) Connector	AC output current (rms)	12 A	5 A
	AC output voltage	93…121V rms single-phase (110V nom)	196…253V rms single-phase (230V nom)
Brake Power Output (P1L) Connector and I/O (IOL) Connector	24V DC Power Supply	20 A	
Contactor (CR1)	Contactor control voltage (1)	21.6…26.4V DC	
	Contactor control current (1)	12…9 mA	
	Contactor pickup current (2)	1.75 A	0.87 A
	Contactor hold-in current (2)	0.14 A	0.07 A
	Contactor voltage	93…121V rms single-phase (110V nom)	196…253V rms single-phase (230V nom)
	Contactor pickup time	18.5 ms (min) 30.0 ms (max)	
	Contactor dropout Time	10.0 ms (min) 60.0 ms (max)	

(1) Power specifications for DC Interface Module (100 JE) COIL_E1 and COIL_E2 input.

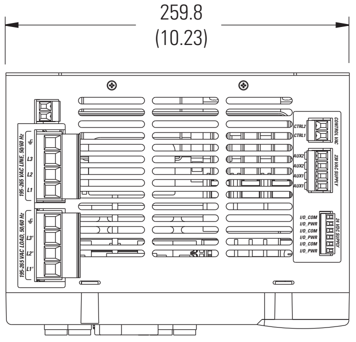
(2) Current provided by auxiliary VAC input.

Power Specifications (catalog numbers 2094-AL09 and 2094-BL02)

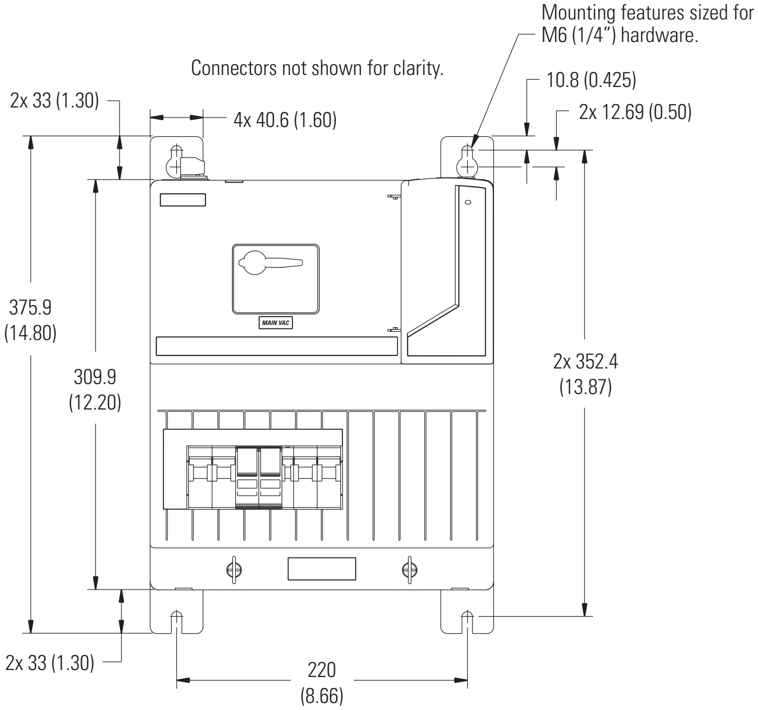
Designator	Attribute	2094-AL09 (230V)	2094-BL02 (460V)
VAC Line (IPL) Connector	AC input voltage	195-265V rms three-phase (230V nom)	380-520V rms three-phase (460V nom)
	AC input frequency	47…63 Hz	
VAC Load (OPL) Connector	Main AC output current	20 A rms	30 A rms
Control Power Output (CPL) Connector	AC output current	3 A	
	AC output voltage	195-265V rms three-phase (230V nom)	190-260V rms three-phase (230V nom)
Brake Power Output (PSL) Connector	Brake power 24V DC	2.0 A	
	I/O brake power 24V DC	5.7 A	
LIM Contactor (CR1)	Contactor pickup current	383 mA	
	Contactor hold-in current	383 mA	
	Contactor voltage	24V DC	
	Contactor pickup time	50 ms (min) 80 ms (max)	
	Contactor dropout time	80 ms (min) 125 ms (max)	

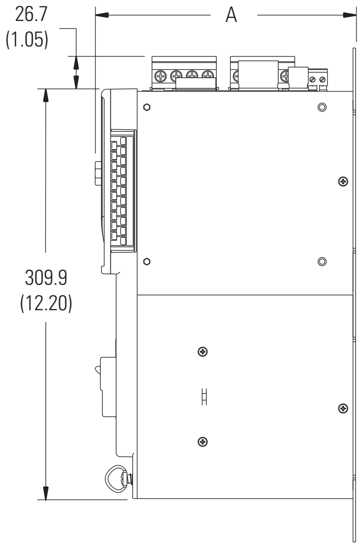
Line Interface Module Dimensions

Dimensions (catalog numbers 2094-AL*xx*S, 2094-BL*xx*S, and 2094-XL75S-C*x*)



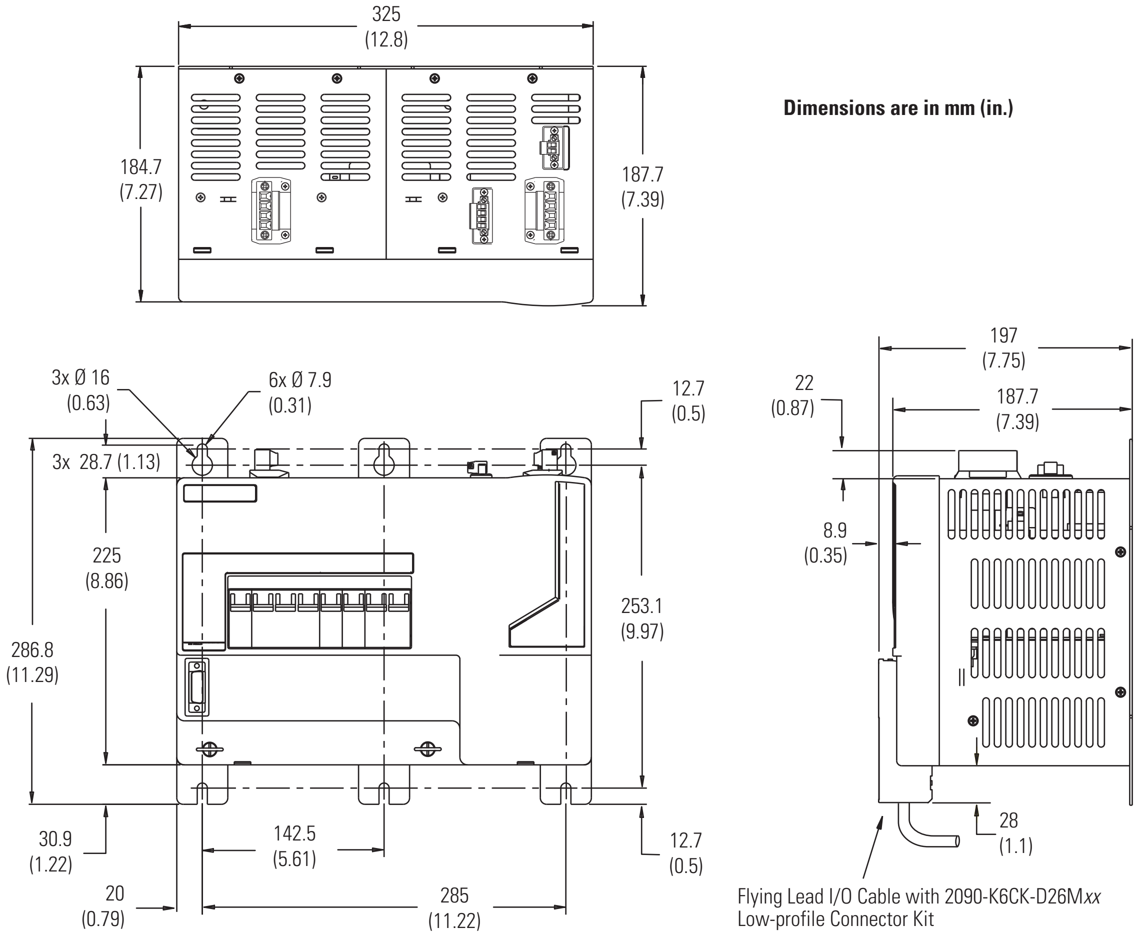
Dimensions are in mm (in.)
2094-XL75S-C*x* is shown





Cat. No.	**Dimension A** mm (in.)
2094-AL*xx*S	198.3 (7.81)
2094-XL75S-C*x*	
2094-BL*xx*S	248.0 (9.76)

Line Interface Module Dimensions (catalog number 2094-AL09)

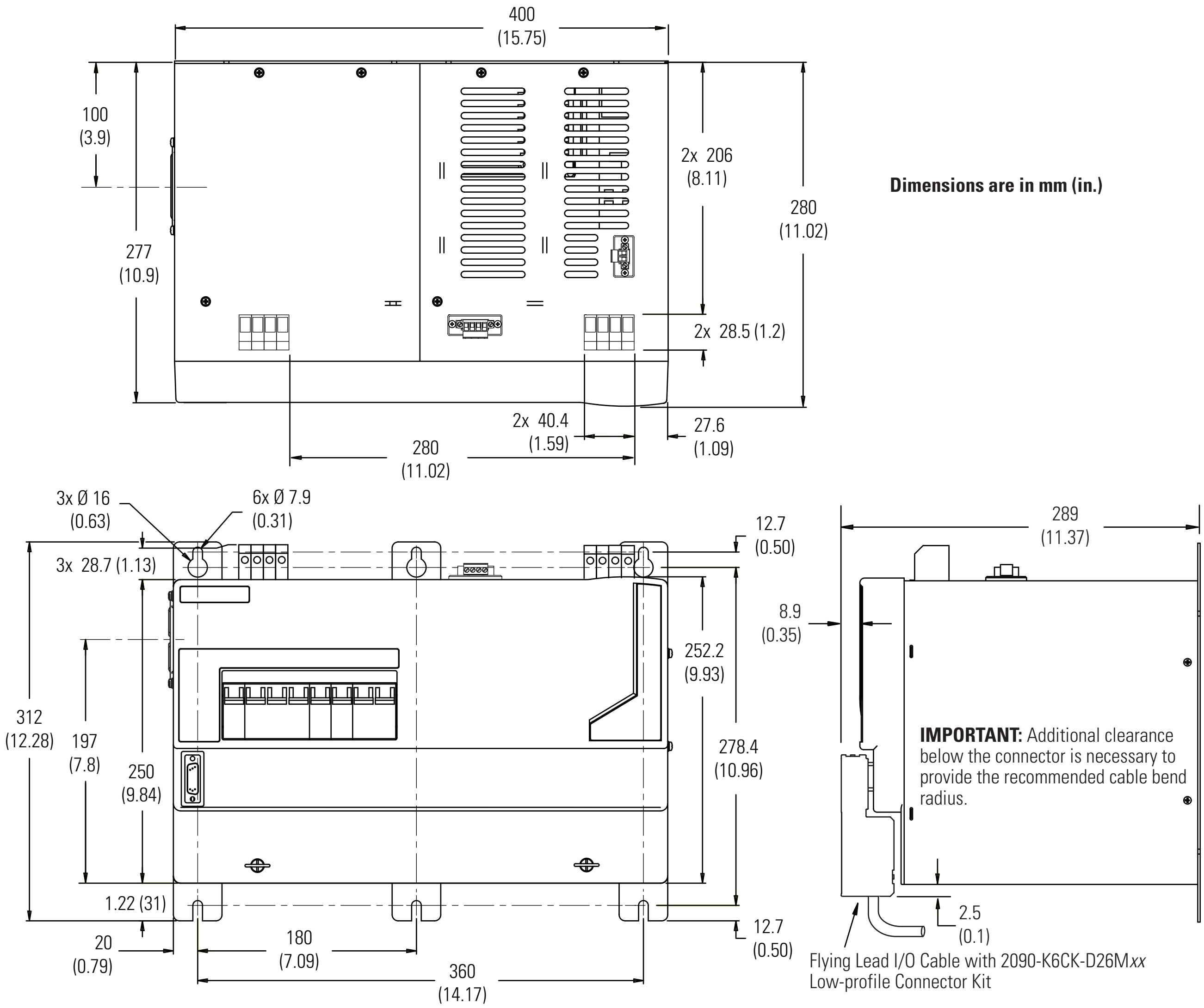


Dimensions are in mm (in.)

Flying Lead I/O Cable with 2090-K6CK-D26M*xx* Low-profile Connector Kit

IMPORTANT: Additional clearance below the connector is necessary to provide the recommended cable bend radius.

Line Interface Module Dimensions (catalog number 2094-BL02)



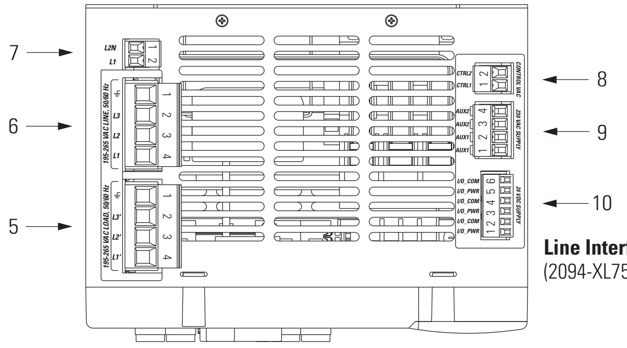
Dimensions are in mm (in.)

IMPORTANT: Additional clearance below the connector is necessary to provide the recommended cable bend radius.

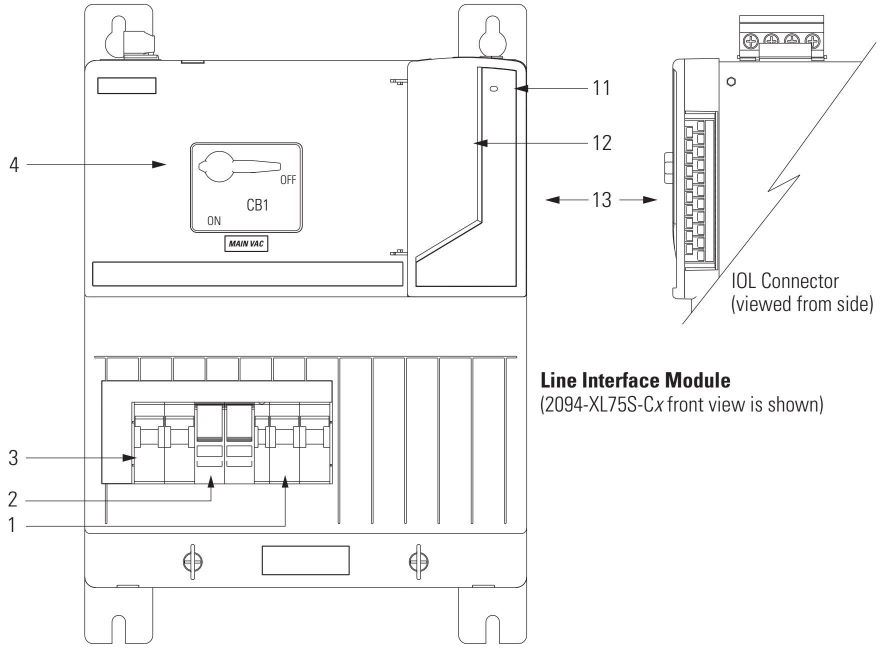
Flying Lead I/O Cable with 2090-K6CK-D26M*xx* Low-profile Connector Kit

Line Interface Module Connectors and Indicators

Catalog Numbers 2094-AL*xx*S, 2094-BL*xx*S, and 2094-XL75S-C*x*



Line Interface Module
(2094-XL75S-C*x* top view is shown)



Line Interface Module
(2094-XL75S-C*x* front view is shown)

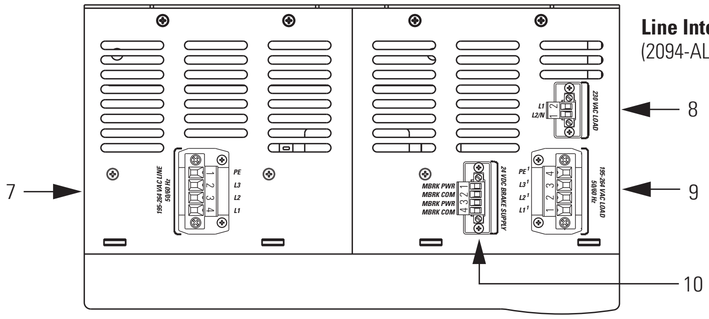
Item	Description
1	CB2 - Control and auxiliary VAC
2	FB1 - Fuse block
3	CB3 - Brake and I/O VAC
4	CB1 - Main VAC disconnect
5	VAC load (OPL) connector
6	VAC line (IPL) connector
7	Auxiliary power input (APL) connector (1)

Item	Description
8	Control power output (CPL) connector
9	Auxiliary power output (P2L) connector
10	24V DC brake power output (P1L) connector
11	24V power status indicator
12	I/O (IOL) connector access door
13	I/O (IOL) connector

(1) Auxiliary Power Input (APL) connector is present only on the 2094-XL75S-C*x* model.

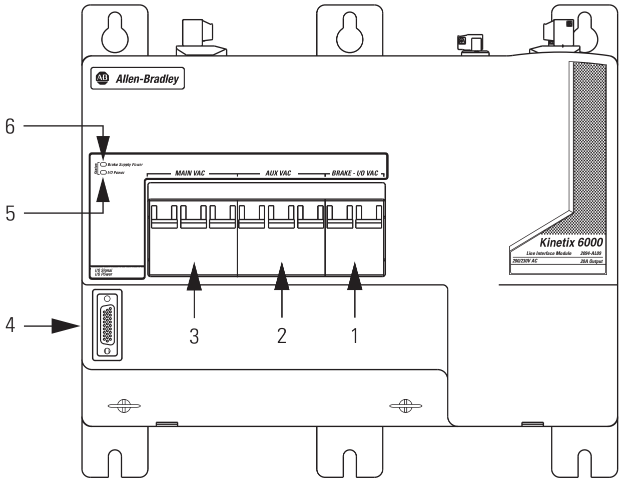
For replacement connector set catalog numbers, refer to Connector Sets on page 463.

Catalog Numbers 2094-AL09 and 2094-BL02



Line Interface Module
(2094-AL09 top view is shown)

Item	Description
1	CB3 - Brake and I/O VAC
2	CB2 - Control and auxiliary VAC
3	CB1 - Main VAC
4	I/O (IOL) connector
5	I/O power status indicator
6	Brake power status indicator
7	VAC line (IPL) connector
8	Control power output (CPL) connector
9	VAC load output (OPL) connector
10	24V DC brake power output (PSL) connector



Line Interface Module
(2094-AL09 front view is shown)

For I/O connector kit, refer to Low-profile Connector Kit Components on page 441.

Line Interface Module Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your module. For questions regarding product availability, contact your Allen-Bradley distributor.

2094 - *x* L *xxx* - C*x*

Features
AL09 = 20 A, 24V DC @ 8 A, internal three-phase line filter
BL02 = 30 A, 24V DC @ 8 A, internal three-phase line filter
AL15S = 15 A, 230V AC auxiliary power output, 24V DC @ 20 A, configurable branch circuit protection
BL10S = 10 A, 230V AC auxiliary power output, 24V DC @ 20 A, configurable branch circuit protection
AL/BL25S = 25 A, 230V AC auxiliary power output, 24V DC @ 20 A, configurable branch circuit protection
AL/BL50S = 50 A, 230V AC auxiliary power output, 24V DC @ 20 A, configurable branch circuit protection
AL/BL75S = 75 A, 230V AC auxiliary power output, 24V DC @ 20 A, configurable branch circuit protection
XL75S-C1 = 75 A, Input for 110V AC (customer-supplied) auxiliary power, 24V DC @ 20 A
XL75S-C2 = 75 A, Input for 230V AC (customer-supplied) auxiliary power, 24V DC @ 20 A

Module Input Voltage
AL = 230V AC, 50/60 Hz
BL = 460V AC, 50/60 Hz
XL = 230/460V AC, 50/60 Hz

Bulletin Number

AC Line Filters

This section contains AC line filter selection tables, specification tables, and dimension drawings. Use the tables below to match an AC line filter to your servo drive.

AC Line Filter Selection

Drive Family	Drive Cat. No.	AC Line Filter Cat. No.
Kinetix 6000	2094-AC05-MP5-S	2090-XXLF-X330B
	2094-AC05-M01-S	
	2094-AC09-M02-S	
	2094-AC16-M03-S	2090-XXLF-375
	2094-AC32-M05-S	2090-XXLF-3100
	2094-BC01-MP5-S	2090-XXLF-X330B
	2094-BC01-M01-S	
	2094-BC02-M02-S	
	2094-BC04-M03-S	2090-XXLF-375B
	2094-BC07-M05-S	2090-XXLF-3100
Kinetix 6200/ Kinetix 6500	2094-BC01-MP5-M	2090-XXLF-X330B
	2094-BC01-M01-M	
	2094-BC02-M02-M	
	2094-BC04-M03-M	2090-XXLF-375B
	2094-BC07-M05-M	2090-XXLF-3100
Kinetix 300 (2)	2097-V31PR0	2090-XXLF-TC116
	2097-V31PR2	
	2097-V33PR5	2090-UXLF-336

Drive Family	Drive Cat. No.	AC Line Filter Cat. No.
Kinetix 3	2071-AP0…2071-AP4	2090-XXLF-TC116
	2071-AP8 (single-phase)	
	2071-AP8 (three-phase)	2090-XXLF-TC316
	2071-A10…2071-A15	
Kinetix 2000	2093-AC05-M*xx*	2090-XXLF-TC116
		2090-XXLF-TC316
Kinetix 7000	2099-BM06-S	2090-XXLF-TC350
	2099-BM07-S	
	2099-BM08-S	2090-XXLF-TC365
	2099-BM09-S	2090-XXLF-TC3100
	2099-BM10-S	2090-XXLF-TC3150
	2099-BM11-S	2090-XXLF-TC3200
	2099-BM12-S	2090-XXLF-TC3250
8720MC-RPS	8720MC-RPS065-B*x*	8720MC-RFI80
	8720MC-RPS190-B*x*	8720MC-EF190-VB (1)

---

(1) Line filter unit includes magnetic contactor, harmonic filter, and varistor.

(2) For Bulletin 2097 line filters used with the Kinetix 300 drive family, refer to AC Line Filters on page 345.

AC Line Filter Selection

Drive Family	Drive Cat. No.	AC Line Filter Cat. No.	Motor Cables > 30 m
Ultra3000/ Ultra5000	2098-*xxx*-005	2090-UXLF-106	2090-UXLF-110
	2098-*xxx*-010	2090-UXLF-110	2090-UXLF-110
	2098-*xxx*-020	2090-UXLF-123	2090-UXLF-123
	2098-*xxx*-030	2090-UXLF-136	2090-UXLF-132
	2098-*xxx*-075	2090-UXLF-336	2090-UXLF-HV330
	2098-*xxx*-150	2090-UXLF-350	2090-UXLF-HV350
	2098-*xxx*-HV030, 2098-*xxx*-HV050, 2098-*xxx*-HV100, 2098-*xxx*-HV150	2090-UXLF-HV323	2090-UXLF-HV323
	2098-*xxx*-HV220	2090-UXLF-HV330	2090-UXLF-HV330

AC Line Filter Specifications

AC Line Filter Cat. No.	Specifications (1) (2)							Dimensions
	Voltage	**Phase**	**Current** A @ 50 °C (122 °F)	**Power Loss** W	**Leakage Current** mA	**Weight, approx.** kg (lb)	**Operating Temperature**	
2090-UXLF-106	250V AC 50/60 Hz	Single	6	3.5	2.26	0.3 (0.66)	-25…85 °C (-13…185 °F)	page 485
2090-UXLF-110			10	2.7	45	0.95 (2.0)		
2090-UXLF-123			23	10	90	1.6 (3.5)		
2090-UXLF-132			32	20	90			
2090-UXLF-136			36	–	200	1.75 (3.9)		page 486
2090-XXLF-TC116			16	–	87	0.80 (1.7)	-25…100 °C (-13…212 °F)	page 490
2090-UXLF-336		Three	36	–	138	2.7 (5.9)	-25…85 °C (-13…185 °F)	page 486
2090-UXLF-350			50	25	138			
2090-UXLF-HV323	520V AC 50/60 Hz		23	20	80	1.6 (3.5)		page 485
2090-UXLF-HV330			30	51	24	1.8 (4.0)		page 487
2090-XXLF-X330B	500V AC 50/60 Hz		30	38	64	2.7 (5.9)		
2090-UXLF-HV350			50	25	35	4.8 (10.6)		page 488
2090-XXLF-375			75	57	50	5.2 (11.4)		
2090-XXLF-375B					108			
2090-XXLF-3100			100	75	73	9.5 (20.9)		
2090-XXLF-TC316	520V AC 50/60 Hz		16	–	38	0.80 (1.7)	-25…100 °C (-13…212 °F)	page 489
2090-XXLF-TC350			50	–	38	2.4 (5.3)		page 489
2090-XXLF-TC365			65	–	38	2.4 (5.3)		
2090-XXLF-TC3100			100	–	38	5.2 (11.5)		
2090-XXLF-TC3150			150	–	76	7.5 (16.5)		page 490
2090-XXLF-TC3200			200	–	76	7.5 (16.5)		
2090-XXLF-TC3250			250	–	76	7.5 (16.5)		
8720MC-RFI80			80	25.9	–	5.3 (11.7)	-25…85 °C (-13…185 °F)	page 491
8720MC-EF190-VB			190	–	–	34.0 (74.8)		

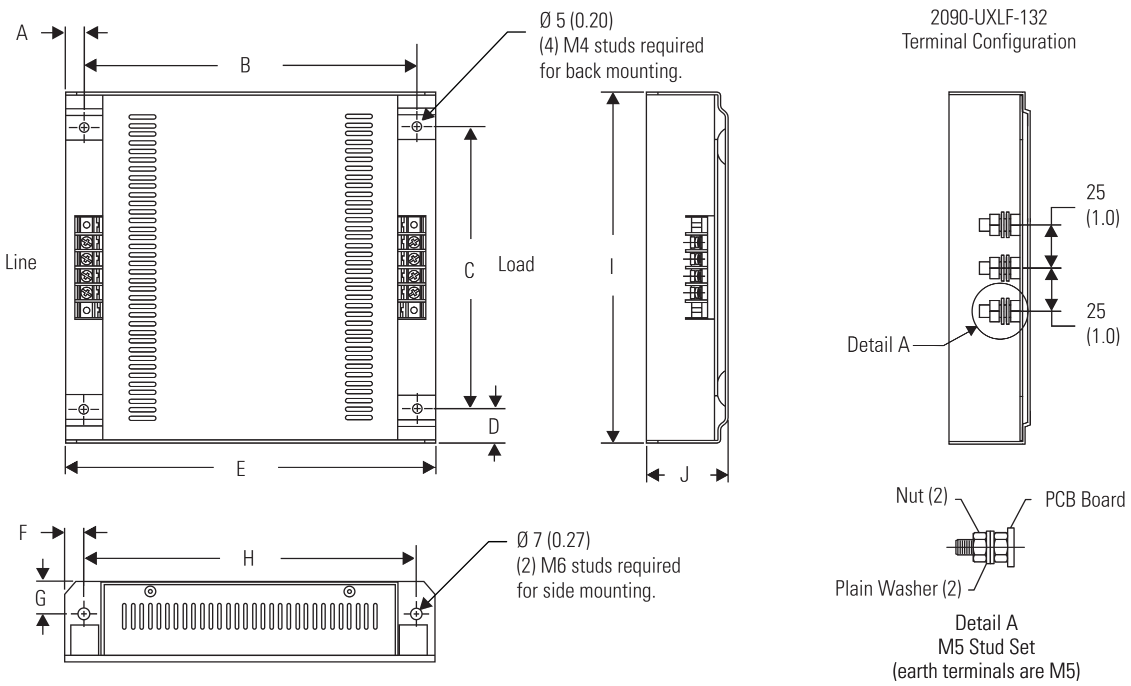
(1) For all filters, 90% relative humidity.

(2) For all filters, 10…200 Hz @ 1.8 g vibration.

AC Line Filter Dimensions

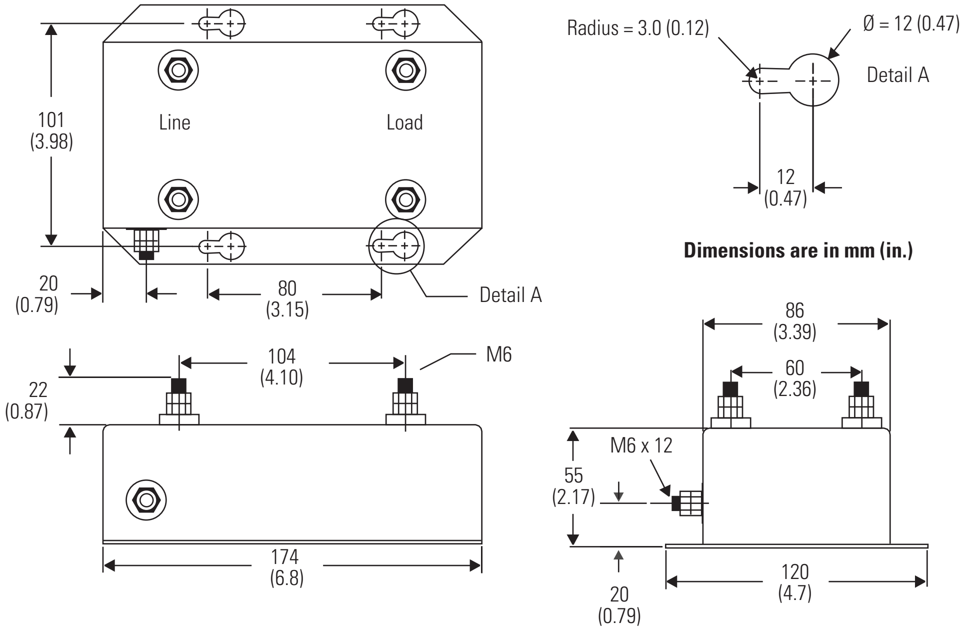
AC Line Filter Dimensions

Dimensions are in mm (in.)



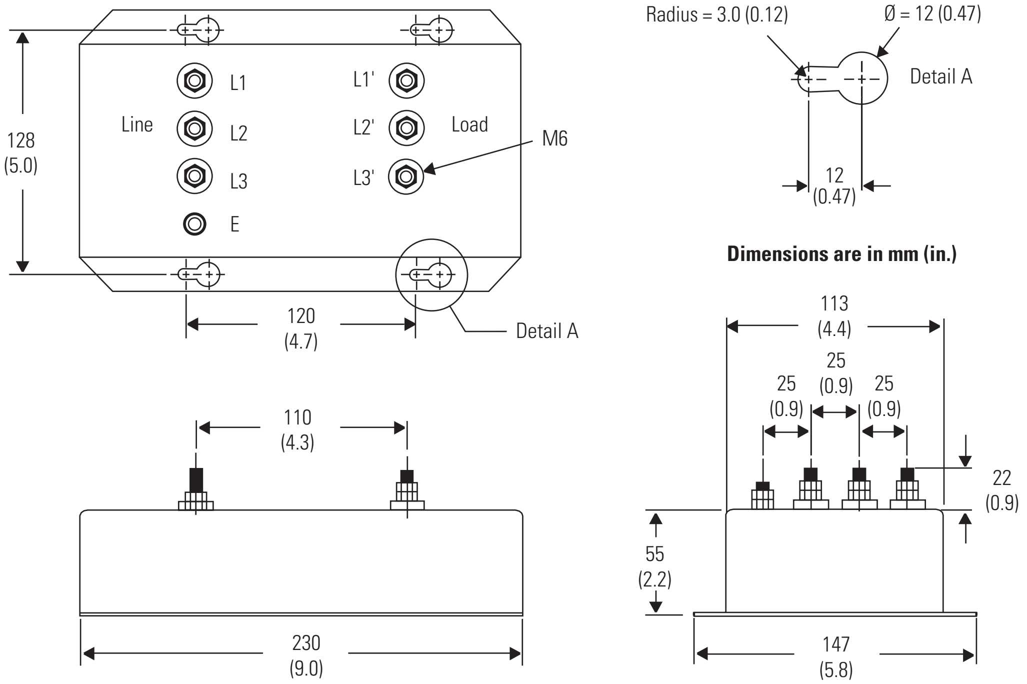
Cat. No.	A mm (in.)	B mm (in.)	C mm (in.)	D mm (in.)	E mm (in.)	F mm (in.)	G mm (in.)	H mm (in.)	I mm (in.)	J mm (in.)
2090-UXLF-106	9.0 (0.35)	152.0 (5.99)	55.0 (2.17)	18.0 (0.71)	170.0 (6.69)	9.0 (0.35)	10.0 (0.39)	152.0 (5.99)	92.0 (3.62)	25.0 (0.98)
2090-UXLF-110	11.0 (0.43)	192.0 (7.56)	104.0 (4.0)	20.0 (0.79)	214.0 (8.42)	11.0 (0.43)	16.0 (0.63)	192.0 (7.56)	145.0 (5.71)	40.0 (1.58)
2090-UXLF-123			164.0 (6.46)				19.0 (0.75)		204 (8.04)	47.0 (1.85)
2090-UXLF-132										
2090-UXLF-HV323										

AC Line Filter Dimensions (catalog number 2090-UXLF-136)



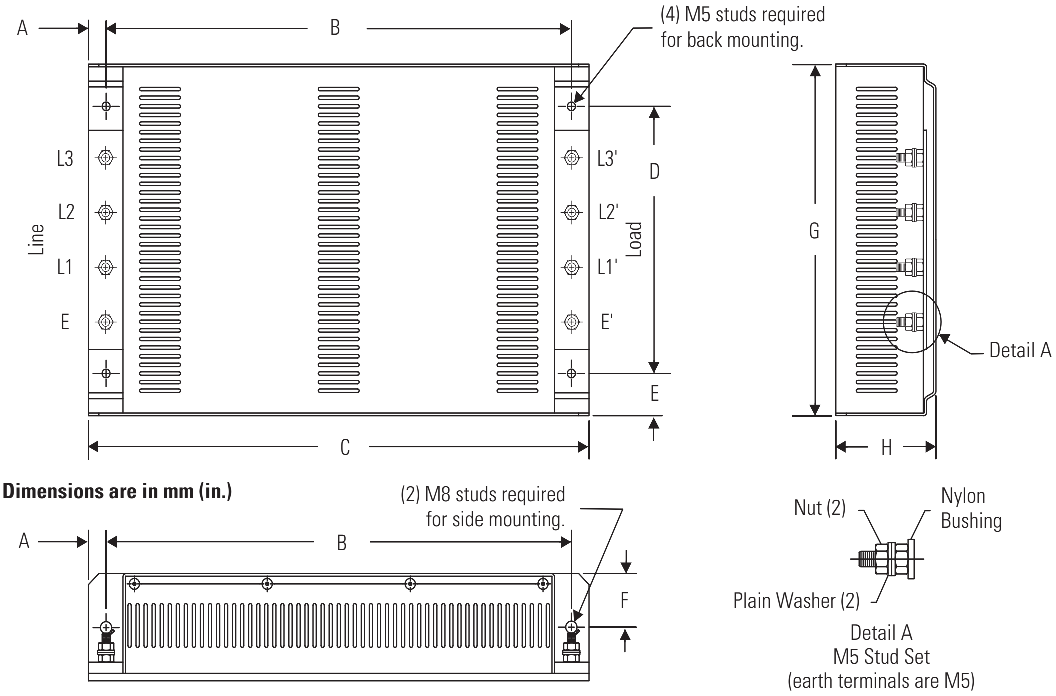
Dimensions are in mm (in.)

AC Line Filter Dimensions (catalog numbers 2090-UXLF-336 and 2090-UXLF-350)



Dimensions are in mm (in.)

AC Line Filters Dimensions

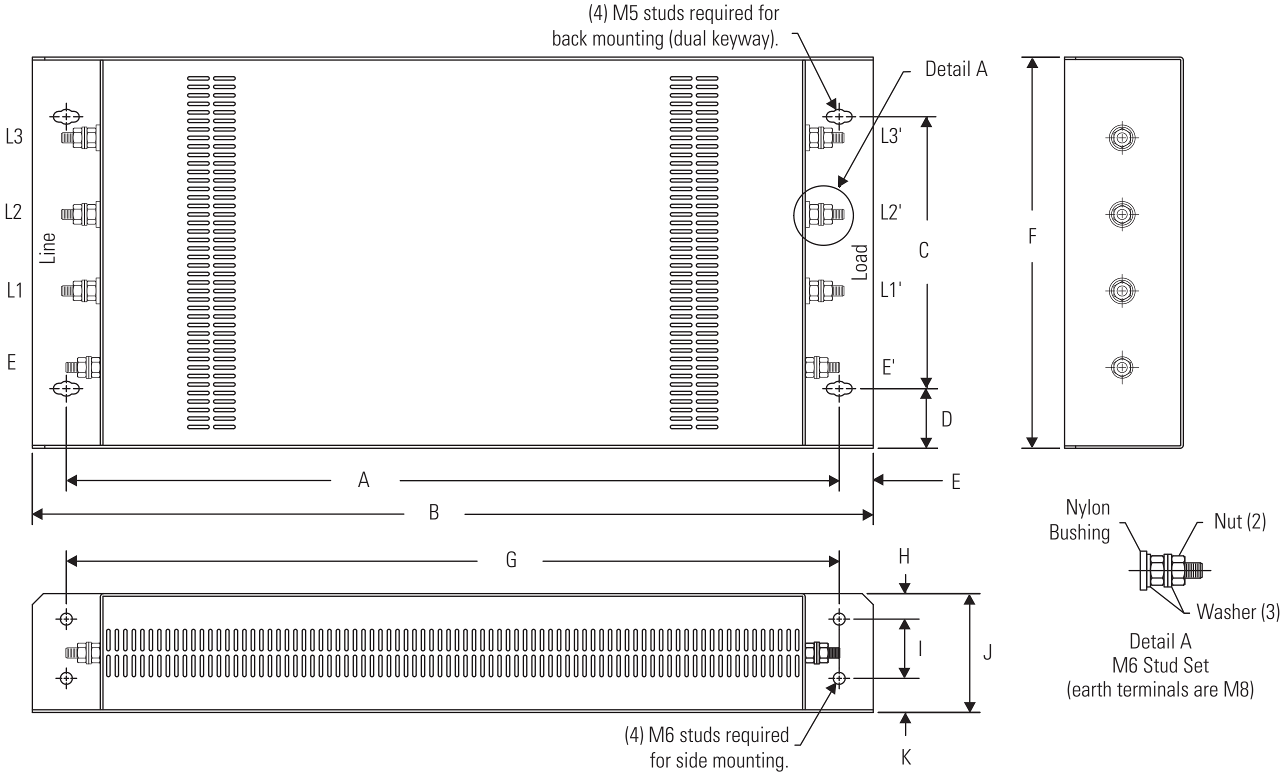


Dimensions are in mm (in.)

Cat. No.	**A** mm (in.)	**B** mm (in.)	**C** mm (in.)	**D** mm (in.)	**E** mm (in.)	**F** mm (in.)	**G** mm (in.)	**H** mm (in.)
2090-UXLF-HV330	11.0 (0.4)	338 (13.3)	360 (14.2)	145 (5.7)	29.5 (1.1)	16.0 (0.63)	204 (8.0)	40.0 (1.6)
2090-XXLF-X330B	15.0 (0.6)	330 (13.0)		155 (6.1)	20.0 (0.8)	32.5 (1.3)	195 (7.7)	65.0 (2.5)

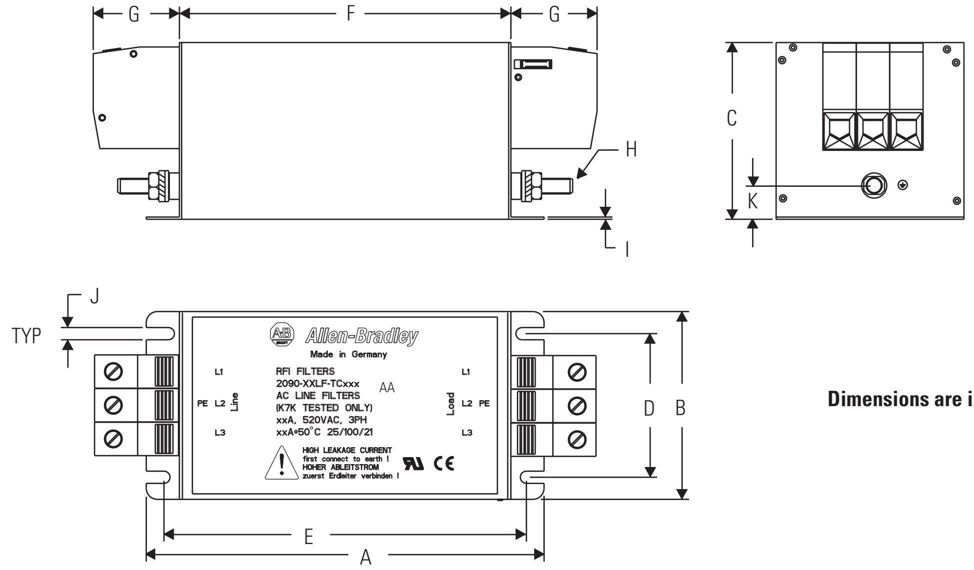
AC Line Filter Dimensions

Dimensions are in mm (in.)



Cat. No.	A mm (in.)	B mm (in.)	C mm (in.)	D mm (in.)	E mm (in.)	F mm (in.)	G mm (in.)	H mm (in.)	I mm (in.)	J mm (in.)	K mm (in.)
2090-UXLF-HV350	578 (22.7)	618 (24.3)	160 (6.3)	35 (1.4)	20 (0.8)	230 (9.0)	578 (22.7)	15 (0.6)	35 (1.4)	70 (2.7)	20 (0.8)
2090-XXLF-375 2090-XXLF-375B	646 (25.4)	686 (27.0)	192 (7.5)			262 (10.3)	646 (25.4)				
2090-XXLF-3100	741 (29.2)	785 (30.9)	215 (8.4)	30 (1.2)	21.5 (0.85)	275 (10.8)	741 (29.2)		47 (1.8)	80 (3.1)	18 (0.7)

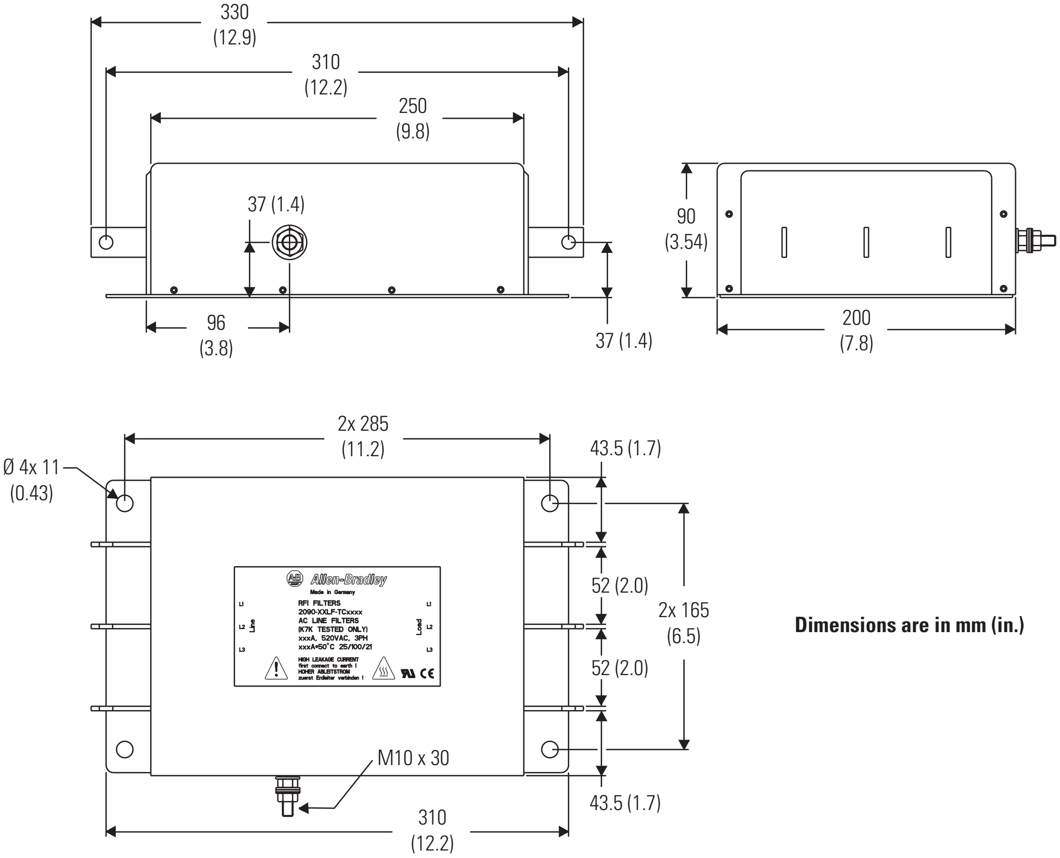
AC Line Filter Dimensions



Dimensions are in mm (in.)

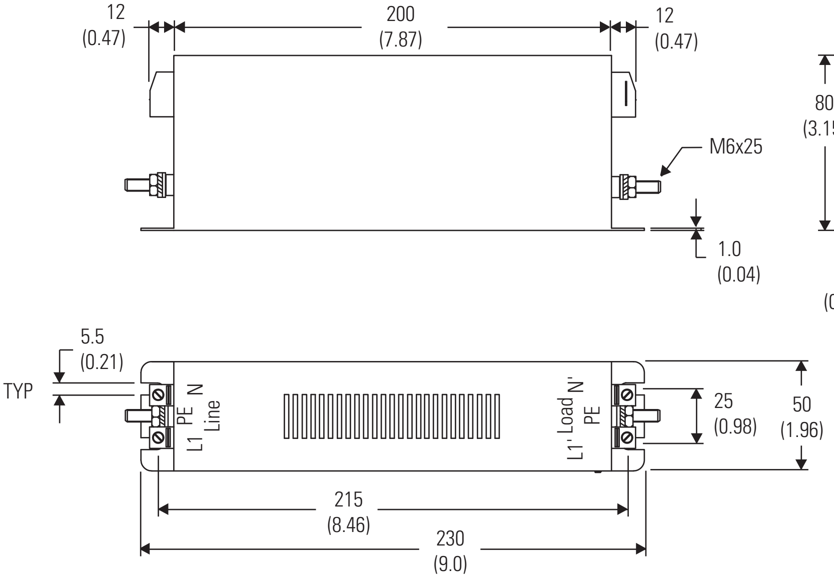
Cat. No.	**A** mm (in.)	**B** mm (in.)	**C** mm (in.)	**D** mm (in.)	**E** mm (in.)	**F** mm (in.)	**G** mm (in.)	**H** mm (in.)	**I** mm (in.)	**J** mm (in.)	**K** mm (in.)
2090-XXLF-TC316	230 (9.0)	50 (1.96)	80 (3.15)	25 (0.98)	215 (8.46)	200 (7.87)	12 (0.47)	M6x25	1.0 (0.04)	5.5 (0.21)	15 (0.59)
2090-XXLF-TC350	180 (7.08)	85 (3.35)		65 (2.56)	164 (6.45)	150 (5.90)	39 (1.53)				
2090-XXLF-TC365											
2090-XXLF-TC3100	240 (9.45)	95 (3.74)	90 (3.54)	75 (2.95)	223 (8.78)	210 (8.27)	43 (1.69)	M8x40	1.5 (0.06)	5.5 (0.21)	16 (0.63)

AC Line Filter Dimensions
(catalog numbers 2090-XXLF-TC3150, 2090-XXLF-TC3200, and 2090-XXLF-TC3250)



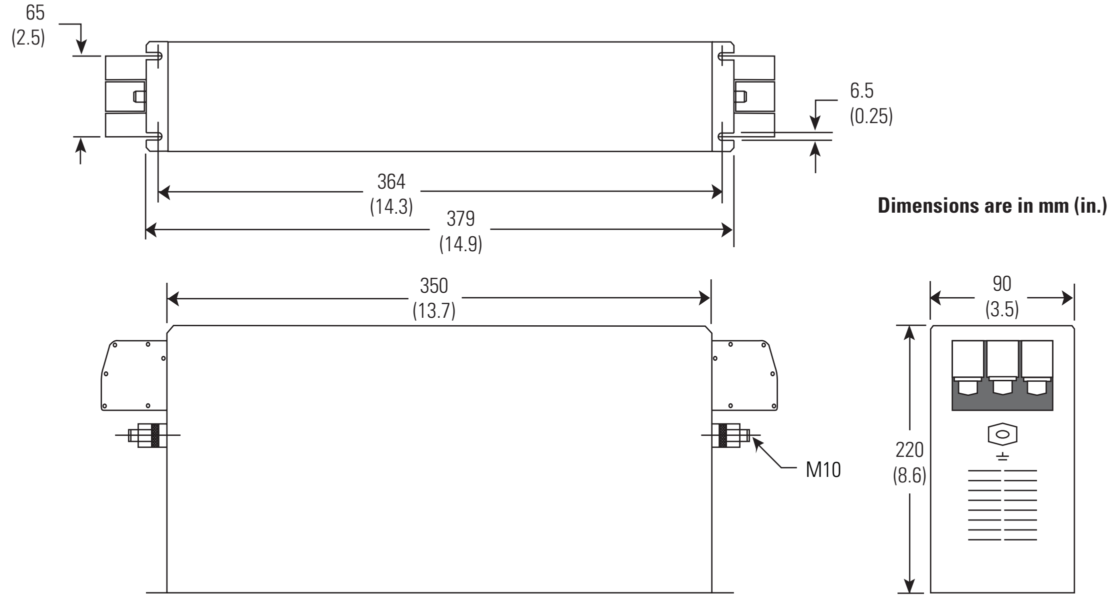
Dimensions are in mm (in.)

AC Line Filter Dimensions
(catalog number 2090-XXLF-TC116)



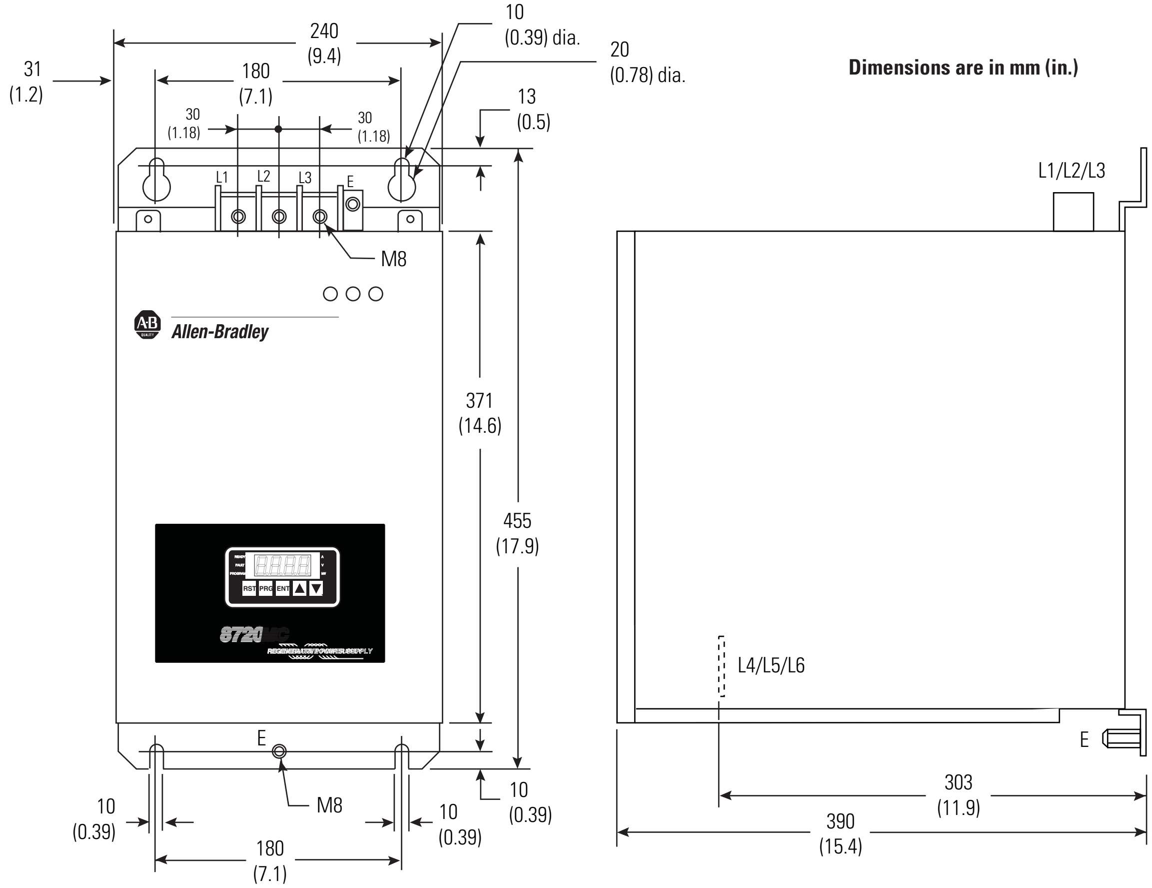
Dimensions are in mm (in.)

AC Line Filter (catalog number 8720MC-RFI80)



Dimensions are in mm (in.)

AC Line Filter (catalog number 8720MC-EF190)



Dimensions are in mm (in.)

External Shunt Modules

This section contains external shunt module/resistor kit specifications, dimensions, and catalog number information. Use the tables below to match a shunt module to your servo drive.

Refer to this table for active shunt solutions from Rockwell Automation Encompass Partners and intended for use with Kinetix 2000, Kinetix 6000, Kinetix 6200, Kinetix 6500, and Kinetix 7000 drives.

Rockwell Automation Encompass Partners	Contact Information
Powerohm Resistors, Inc.	5713 13th Street Katy, Texas 77493 Tel: (800) 838-4694 http://www.powerohm.com
Bonitron, Inc.	521 Fairground Court, Nashville, TN 37211 Tel: (615) 244-2825 http://www.bonitron.com

For Bulletin 2097 shunt modules intended for use with Kinetix 300 servo drives, refer to Shunt Resistor Specifications on page 346.

Bulletin 1394 Passive Shunt Modules

Select one of these Bulletin 1394 passive shunt modules when your Kinetix 6000, Kinetix 6200 or Kinetix 6500 drive application exceeds the capacity of the internal (IAM/AM module) shunt resistor. These external passive shunt modules wire to the Kinetix 6000 (catalog number 2094-BSP2) shunt module.

External Shunt Cat. No.	Specifications						Bussmann Replacement Fuse
	Drive Voltage	Resistance Ω	Peak Power kW	Peak Current A	Cont. Power W	Shipping Weight, approx. kg (lb)	
1394-SR9A	230V AC 1	4	41.0	101.25	300	3.63 (8)	FNQ-R-20-R1 (1)
	460V AC		160	200			FWP50A14F
1394-SR9AF	230V AC 1	4	41.0	101.25	900	3.63 (8)	FNQ-R-20-R1 (1)
	460V AC		160	200			FWP50A14F
1394-SR36A	230V AC 1	4	41.0	101.25	1800	8.6 (19)	FNQ-R-20-R1 (1)
	460V AC		160	200			FWP50A14F
1394-SR36AF	230V AC 1	4	41.0	101.25	3600	9.0 (20)	FNQ-R-25-R1 (1)
	460V AC		160	200			FWP50A14F

(1) Requires the use of an FNQ fuse with an adapter to allow the smaller body fuse to fit the larger FWP fuse holder.

Bulletin 2090 Passive Shunt Modules

Select one of these passive shunt modules when your Ultra3000 or Ultra5000 drive application exceeds the capacity of the internal shunt resistor.

Ultra3000/5000 (1) Drives	**Shunt Module Cat. No.**	**Specifications**							**Fuse Replacement**
		Drive Voltage	**Fan Voltage** V AC	**Resistance** Ω	**Peak Power** kW	**Peak Current** A	**Continuous Power** W	**Shipping Weight** kg (lb)	
2098-*xxx*-005, 2098-*xxx*-010, 2098-*xxx*-020	2090-UCSR-A300	230V AC	N/A	36	4.0	10.5	300	1.51 (3.3)	—
2098-*xxx*-030	9101-1183			30	5.9	14.0	200	—	CCMR-4-½ (2)
2098-*xxx*-075, 2098-*xxx*-150	2090-UCSR-P900			18	10.0	23.3	900	4.08 (9.0)	FWP-10A14F (3)
2098-*xxx*-HV030, 2098-*xxx*-HV050	2090-SR120-09	460V AC	N/A	120	5.3	6.7		3.63 (8.0)	FWP-2.5A14F (3)
2098-*xxx*-HV100	2090-SR040-09			40	16.0	20.0		3.63 (8.0)	FWP-5A14F (3)
	2090-SR040-18			40		20.0	1800	8.6 (19.0)	FWP-6.3A14F (3)

(1) Passive shunt solutions for Ultra3000 and Ultra5000 (460V) drives also exist with Rockwell Automation Encompass Partners. Refer to the table below for shunt module solutions outside the specifications in the table above and for the 2098-*xxx*-HV150 and 2098-*xxx*-HV220 servo drives.

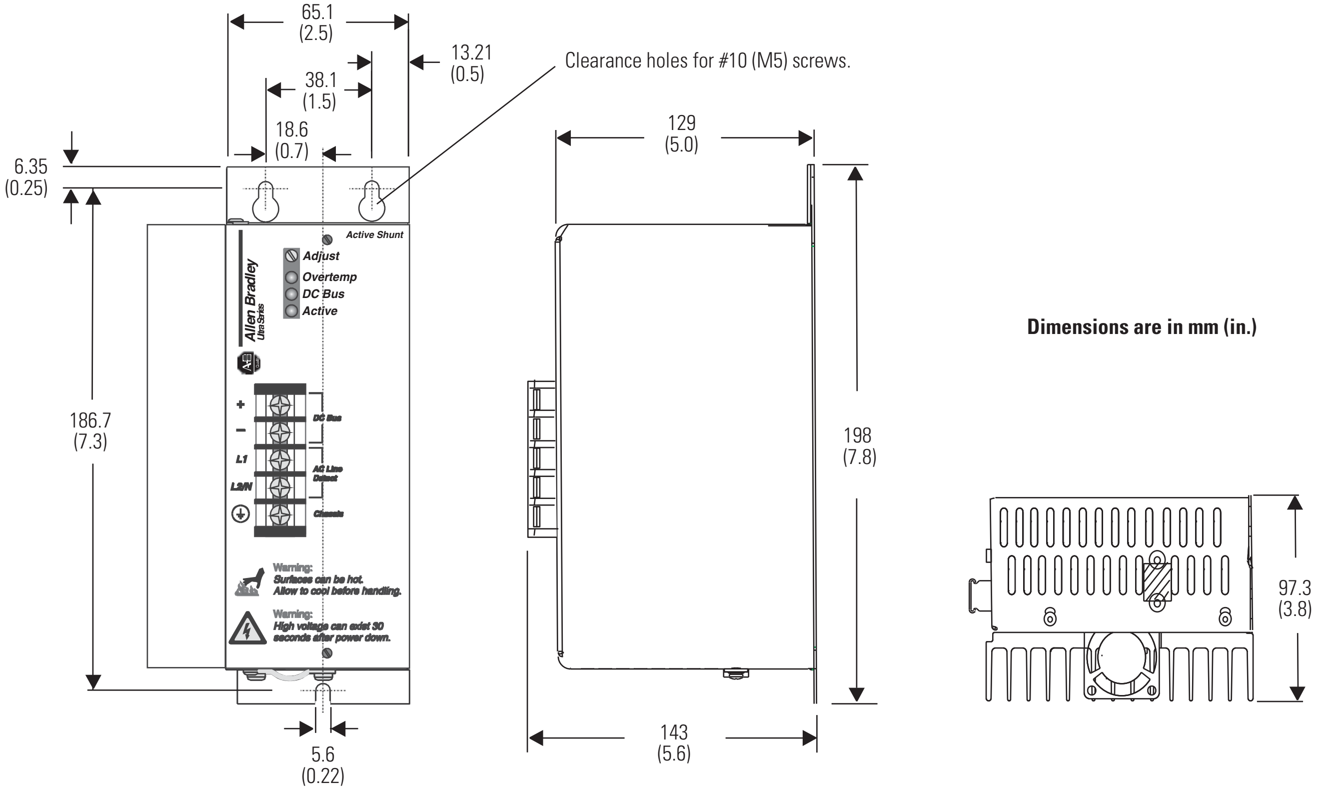
(2) LittelFuse part number.

(3) Bussmann part number.

Rockwell Automation Encompass Partners	**Contact Information**
Powerohm Resistors, Inc.	5713 13th Street Katy, Texas 77493 Tel: (800) 838-4694 http://www.powerohm.com
Bonitron, Inc.	521 Fairground Court, Nashville, TN 37211 Tel: (615) 244-2825 http://www.bonitron.com

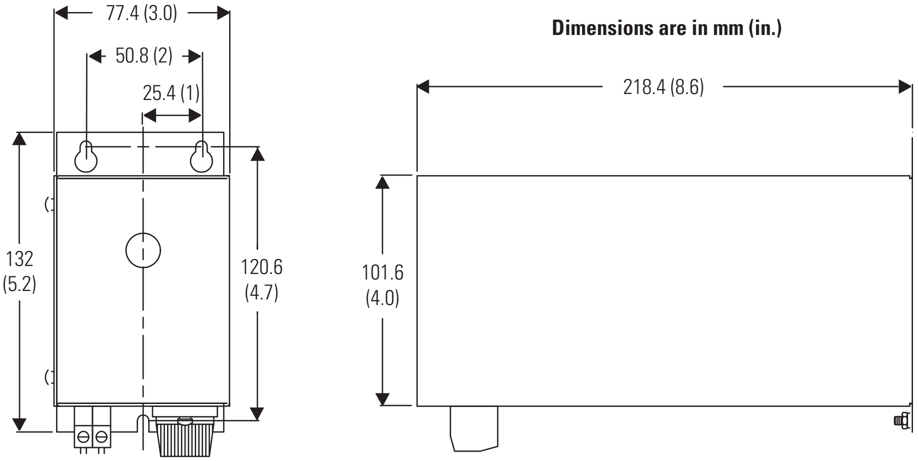
Shunt Resistor Kit Dimensions

Dimensions (catalog number 2090-UCSR-A300)



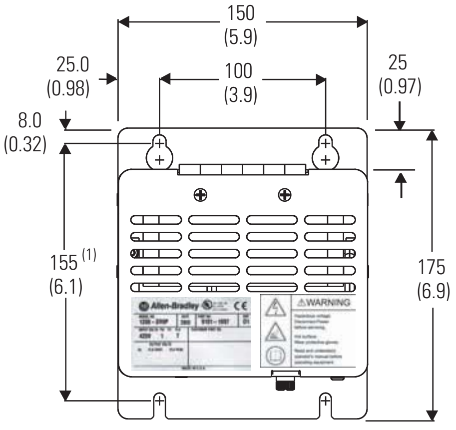
Dimensions are in mm (in.)

Dimensions (catalog number 9101-1183)



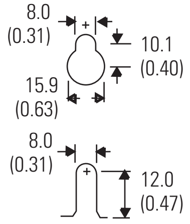
Dimensions are in mm (in.)

Dimensions (catalog number 2090-UCSR-P900)



Front View

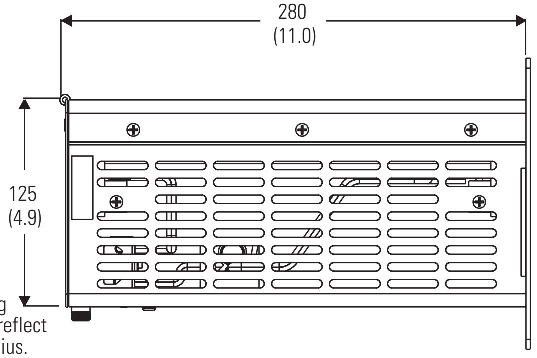
Mounting Hole Detail



All slots accept M6 or 1/4-20 mounting screws.

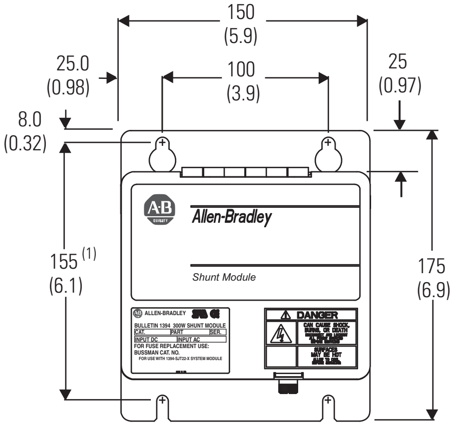
(1) Dimension shown is for mounting hardware location and does not reflect the location of the lower slot radius.

Dimensions are in mm (in.)



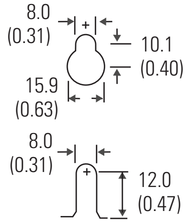
Side View

Dimensions (catalog numbers 2090-SR120-09, 2090-SR040-09, 1394-SR9A, and 1394-SR9AF)



Front View

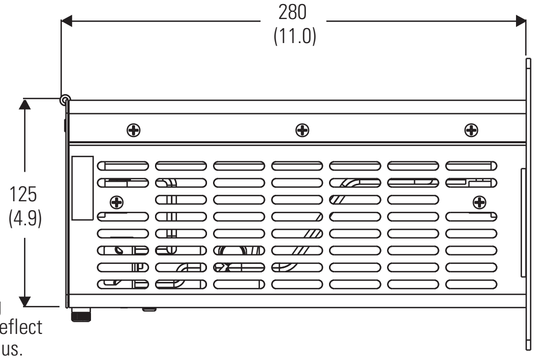
Mounting Hole Detail



All slots accept M6 or 1/4-20 mounting screws.

(1) Dimension shown is for mounting hardware location and does not reflect the location of the lower slot radius.

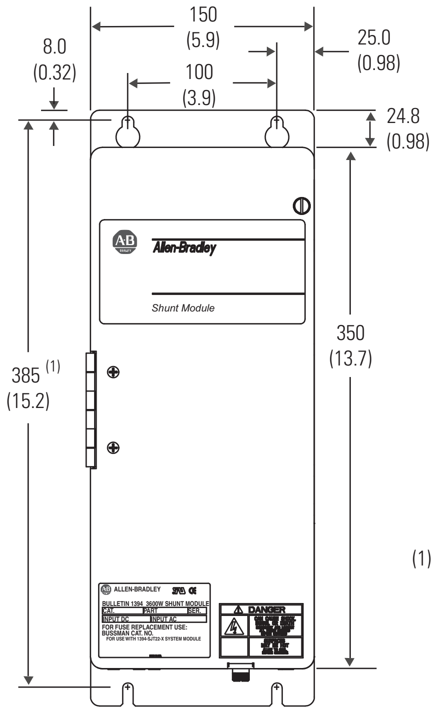
Dimensions are in mm (in.)



Side View

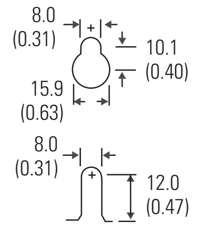
Dimensions (catalog numbers 2090-SR040-18, 1394-SR36A, and 1394-SR36AF)

Dimensions are in mm (in.)



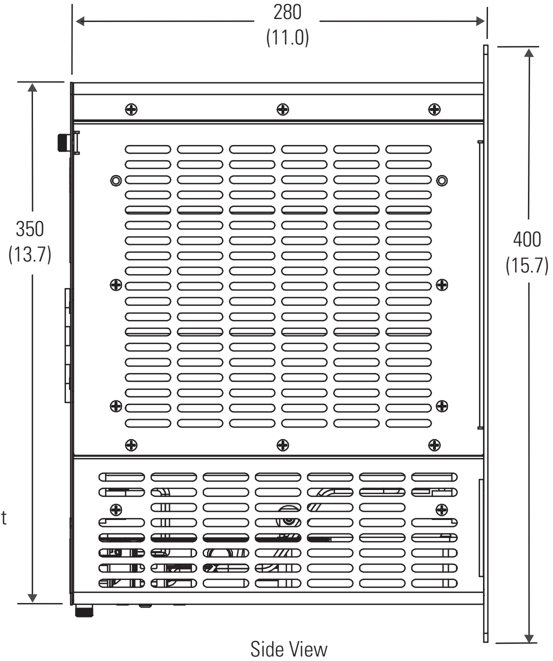
Front View

Mounting Hole Detail



All slots accept M6 or
1/4-20 mounting screws.

(1) Dimension shown is for mounting hardware location and does not reflect the location of the lower slot radius.



Side View

Shunt Module Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering charts below to understand the configuration of your shunt. For questions regarding product availability, contact your Allen-Bradley distributor.

2090 Shunt Resistor Kits for Ultra3000/5000 230V Drives

Cat. No.	Description
2090-UCSR-A300	Active shunt, 300 W for use with 500 W, 1 kW, and 2 kW drives
9101-1183	Passive shunt, 200 W for use with 3 kW drives
2090-UCSR-P900	Passive shunt, 900 W for use with 7.5 and 15 kW drives

2090 Shunt Modules for Ultra3000/5000 460V Drives

2090 - SR *xxx* - *xx*

Continuous Power Rating
09 = 900 W
18 = 1800 W

Resistance Rating
120 = 120 Ω
040 = 40 Ω

SR = Shunt Resistor

Bulletin Number

1394 Shunt Modules for Kinetix 6200, Kinetix 6500, and Kinetix 6000 Servo Drives

1394 - SR *xxxx*

kW Rating
9A = 300 W continuous, 160,000 W peak, no fan
9AF = 900 W continuous, 160,000 W peak, no fan
36A = 1800 W continuous, 160,000 W peak, no fan
36AF = 3600 W continuous, 160,000 W peak, fan-cooled

SR = Shunt Resistor

Bulletin Number

Resistive Brake Modules

This section contains Resistive Brake Module (RBM) descriptions, dimensions, and catalog number information.

Resistive Brake Module Specifications

Cat. No.	**Drive Voltage**	**Resistance** [(1)] Ω	**Peak Energy** J	**Peak Drive Current**		**Continuous Power** W	**Weight, approx.** kg (lb)
				A 0-pk	A rms		
2090-XB33-32	230 or 460V AC	32	150	33	23	30	1.91 (4.22)
2090-XB33-16		16					
2090-XB120-06		6	290	106	75	45	2.75 (6.06)
2090-XB120-03		3					
2090-XB120-01		1					

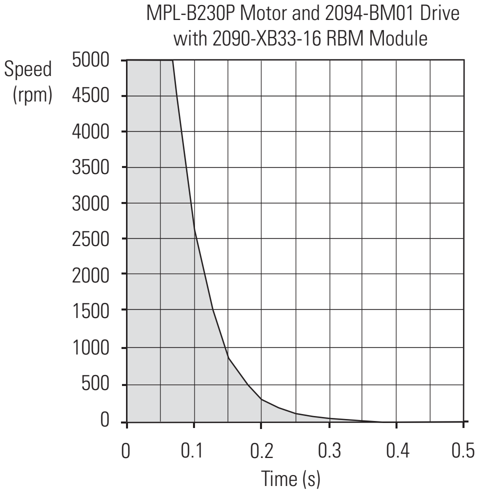
(1) Tolerance = ± 10%.

Use Motion Analyzer software to match an RBM module to your servo motor compatible with Kinetix 2000, Kinetix 6000, Kinetix 6200, Kinetix 6500, and Ultra3000-SE drive systems.

IMPORTANT Drive commands are the preferred and quickest method to bring your drive system to a controlled stop. When using drive commands, the time between braking cycles is limited by the drive/motor/load combination. When the RBM resistors are used to stop the motor, these conditions apply:

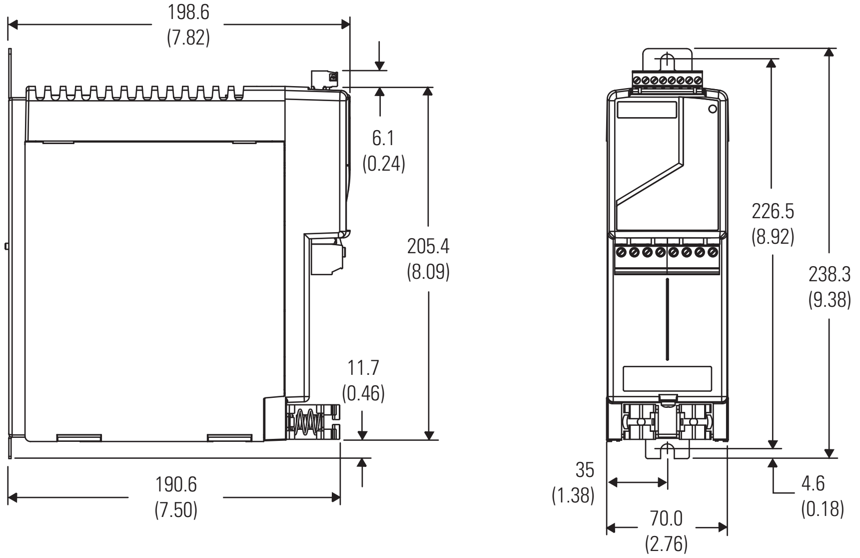
- One minute between braking cycles
- Maximum 15:1 motor inertia
- Maximum motor velocity at the start of braking
- Application must not exceed the current rating of the brake module

Typical RBM Module Curve



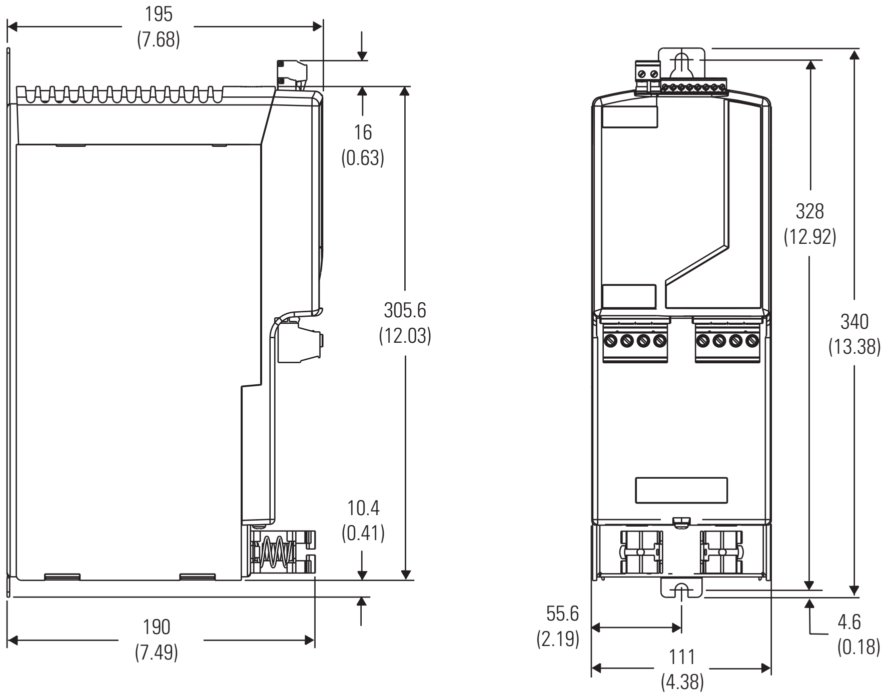
Resistive Brake Module Dimensions

Dimensions (catalog numbers 2090-XB33-16 and 2090-XB33-32)



Dimensions are in mm (in.)

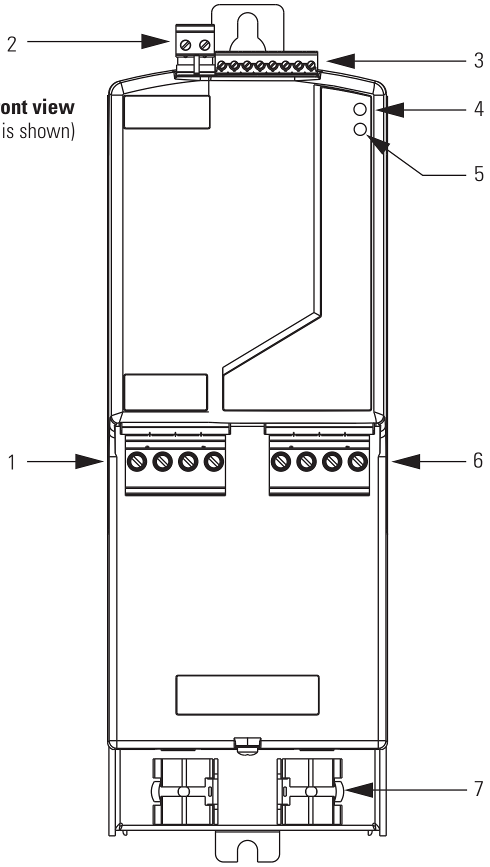
Dimensions (catalog numbers 2090-XB120-01, 2090-XB120-03, and 2090-XB120-06)



Dimensions are in mm (in.)

RBM Connectors and Indicators

Dimensions (catalog numbers 2090-XB33-*xx* and 2090-XB120-*xx*)



Resistive Brake Module, front view
(catalog number 2090-XB120-*xx* is shown)

Item	Description
1	Drive connections (TB1) connector
2	230V AC auxiliary power input (TB4) connector (1)
3	I/O (TB3) connector
4	Contactor status indicator
5	Auxiliary power status indicator (1)
6	Motor connections (TB2) connector
7	Motor cable shield clamps

(1) The 230V AC auxiliary power input (TB4) connector and auxiliary power status indicator are present only on 2090-XB120-*xx* resistive brake modules.

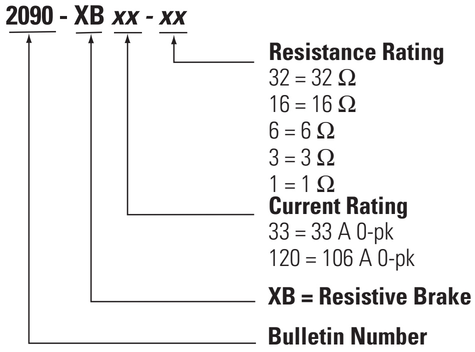
RBM to Drive Interface Cables

RBM interface cables (motor power, RBM to drive) are available for Kinetix 6000, Kinetix 6200/Kinetix 6500, and Ultra3000-SE drives. Refer to the table below for specific RBM to drive interface cable information.

For This Information	Refer to
Cable catalog numbers for compatible drives	page 433
Available cable lengths	page 433
Cable dimensions	page 438
Cable specifications	page 434

Resistive Brake Module Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering charts below to understand the configuration of your module. For questions regarding product availability, contact your Allen-Bradley distributor.



2090 - XB *xx* - *xx*

Resistance Rating
32 = 32 Ω
16 = 16 Ω
6 = 6 Ω
3 = 3 Ω
1 = 1 Ω

Current Rating
33 = 33 A 0-pk
120 = 106 A 0-pk

XB = Resistive Brake

Bulletin Number

8720MC Regenerative Power Supplies

This section contains specifications and catalog number information for the 8720MC-RPS*xxx* Regenerative Power Supplies (RPS). The 8720MC-RPS modules are compatible with Kinetix 6200, Kinetix 6500, Kinetix 6000, and Kinetix 7000 drives when used in DC common bus applications. Refer to the drive chapters for sizing information.

The 8720MC Regenerative Power Supplies support these features:

- Full line regeneration
- Regenerative braking
- Multiple DC common bus drives
- Leader and Follower modes for parallel operation of multiple units
- Adjustable DC output voltage
- CE compliance and UL Listed to U.S. and Canadian safety standards. Refer to http://www.ab.com for more information.

8720MC-RPS Power Specifications

The table below lists general power specifications and requirements for the 8720MC-RPS modules (catalog numbers 8720MC-RPS065 and 8720MC-RPS190).

Attribute	**8720MC-RPS065**		**8720MC-RPS190**	
AC input voltage	324…506V AC rms three-phase			
AC input frequency	47…63 Hz			
AC input current Nom Max (1 minute)	65 A rms 98 A rms	92 A 0-pk 138 A 0-pk	190 A rms 285 A rms	268 A 0-pk 403 A 0-pk
Continuous output current	64 A DC		190 A DC	
Output current (1 minute)	96 A DC		285 A DC	

Refer to 8720MC Regenerative Power Supply User Manual, publication 8720MC-RM001, for additional specifications and dimensions for the 8720MC-RPS modules listed in the table above.

8720MC-RPS Precharge Specifications

The table below lists internal (built-in) and external precharge capacitance of the 8720MC-RPS modules.

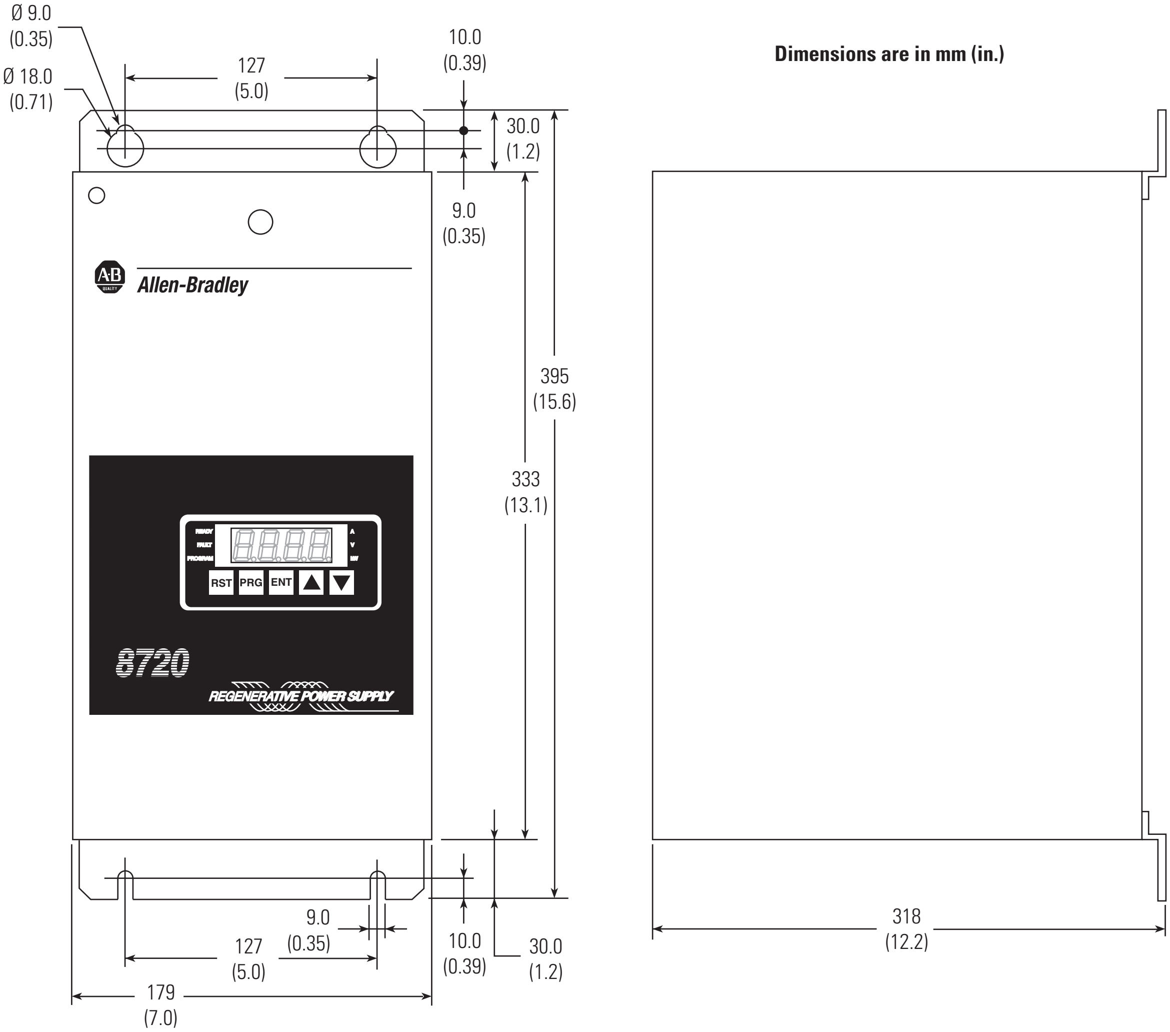
Attribute	8720MC-RPS065	8720MC-RPS190
Capacitance of built-in capacitor	1900 μF	7600 μF
Built-in resistor (value/wattage)	7000 μF (22 Ω/120 W)	25000 μF (10 Ω/400 W)
External resistor (min resistance value) Connect to PR1 and PR2	110000 μF (20 Ω)	165000 μF (10 Ω)
External circuit (min resistance value)	220000 μF (4.7 Ω)	495000 μF (1.5 Ω)

IMPORTANT Large levels of load capacitance may require modification of the 8720MC-RPS internal precharge/discharge circuit.

Refer to the wiring instructions in the 8720MC Regenerative Power Supply Installation Manual, publication 8720MC-RM001, for information on how to determine the appropriate precharge/discharge resistance power value (ohms/watt) to accommodate the capacitance of your system.

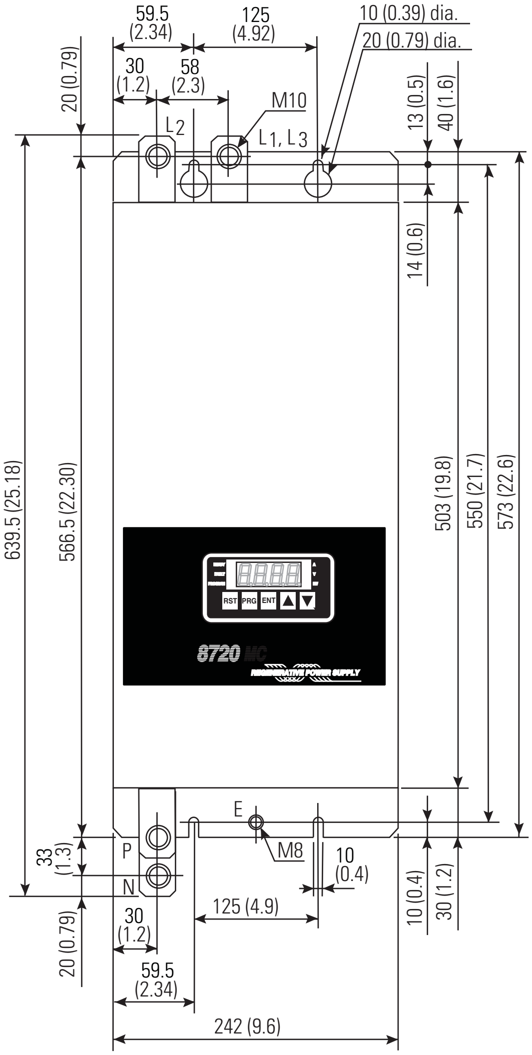
8720MC Regenerative Power Supply Dimensions

Dimensions (catalog number 8720MC-RPS065)

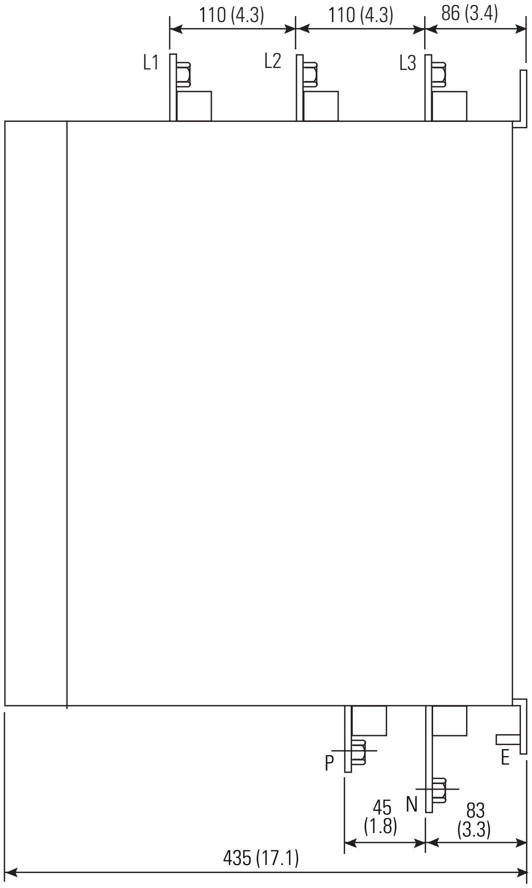


Dimensions are in mm (in.)

Dimensions (catalog number 8720MC-RPS190)

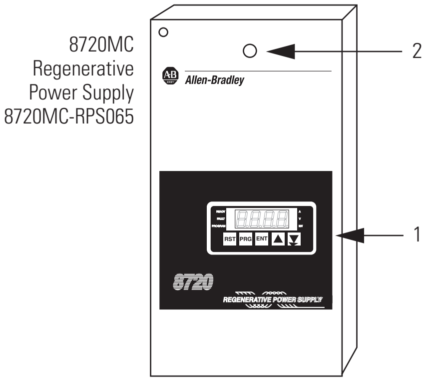


Dimensions are in mm (in.)

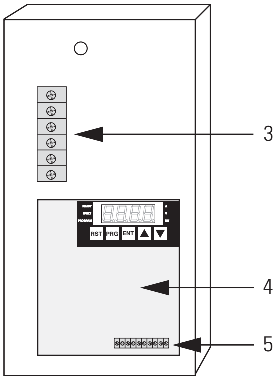


Connectors (catalog number 8720MC-RPS065)

8720MC-RPS065
(with front cover)

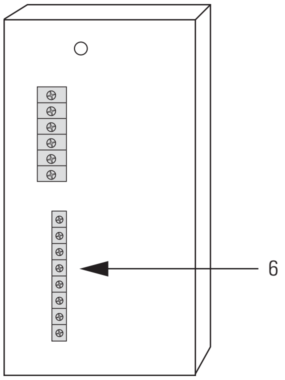


8720MC-RPS065
(with front cover removed)



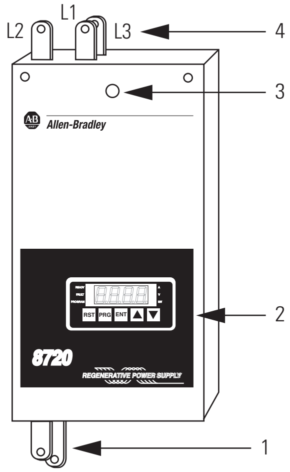
Item	Description
1	Operation panel (master unit only)
2	Power status indicator
3	Main power (TB1) terminal block
4	Regulator board (master unit only)
5	Sequence signal (TB3) terminal block
6	Control power (TB2) terminal block

8720MC-RPS065
(front cover and regulator board removed)

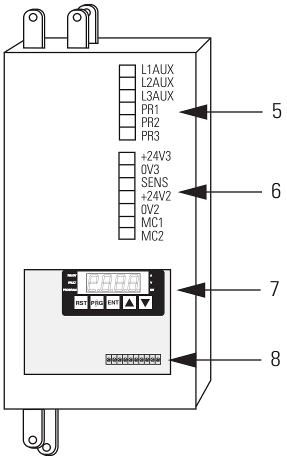


Connectors (catalog number 8720MC-RPS190)

8720MC-RPS190
(with front cover)



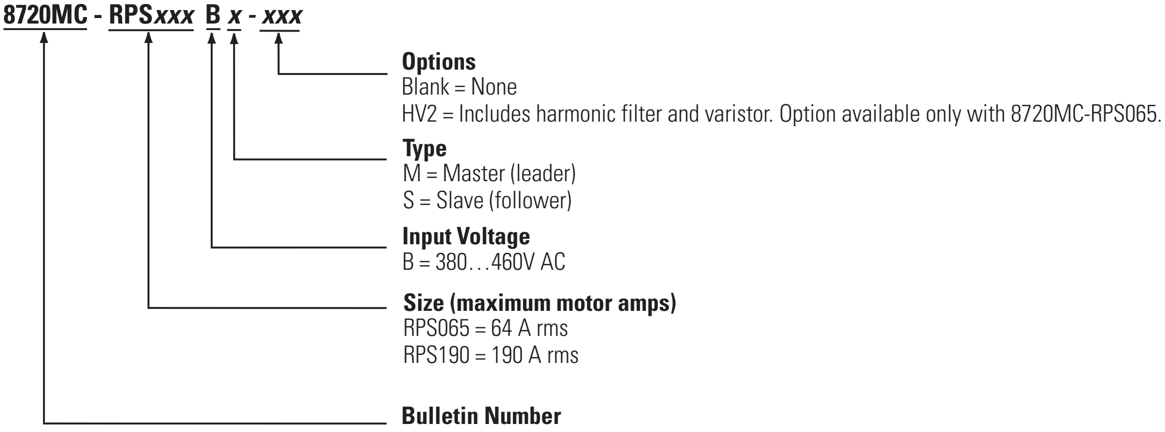
8720MC-RPS190
(with front cover removed)



Item	Description
1	DC bus terminals
2	Operation panel (master unit only)
3	Power status indicator
4	Main power terminals
5	Control power (TB2) terminal block
6	Control power (TB4) terminal block
7	Regulator board (master unit only)
8	Sequence signal (TB3) terminal block

8720MC-RPS Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your 8720MC Regenerative Power Supply. For questions regarding product availability, contact your Allen-Bradley distributor.



8720MC - RPS*xxx* B *x* - *xxx*

Options
Blank = None
HV2 = Includes harmonic filter and varistor. Option available only with 8720MC-RPS065.

Type
M = Master (leader)
S = Slave (follower)

Input Voltage
B = 380…460V AC

Size (maximum motor amps)
RPS065 = 64 A rms
RPS190 = 190 A rms

Bulletin Number

8720MC Line Reactors

This section contains 8720MC line reactor specifications, dimensions, and catalog numbers.

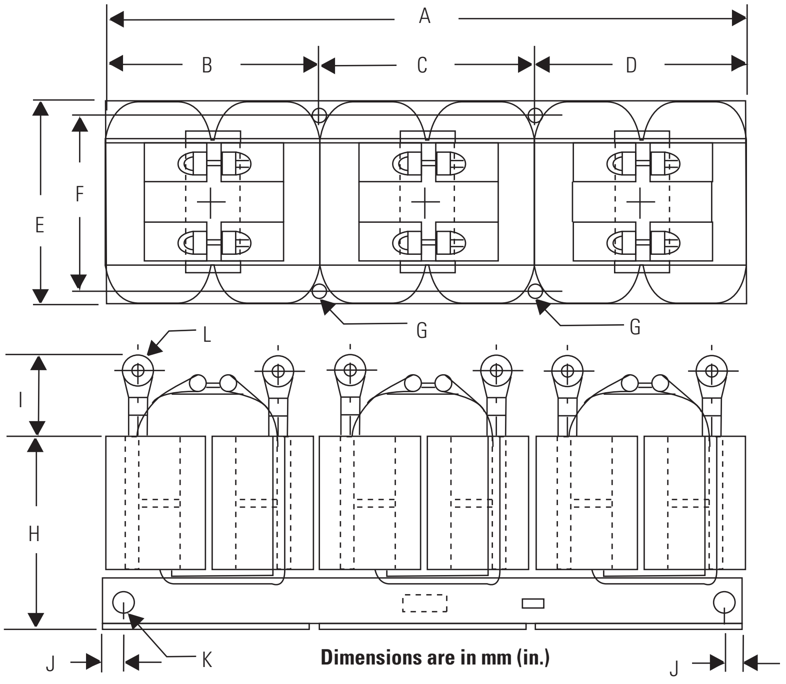
8720MC Line Reactor Specifications

Line Reactor 8720MC-	**Specifications**			
	Maximum Continuous Current A	**Inductance** uH	**Inductance as % Voltage Drop**	**Weight, approx.** kg (lb)
LR03-032B	32	850	3%	17 (37.47)
LR05-048B	48	800	5%	21 (46.29)
LR10-062B	62	1100	10%	27 (59.52)
LR14-070B	70	1200	14%	38 (83.77)
LR10-100B (1)	100	800	10%	100 (220)

(1) Order two 8720MC-LR10-100B line reactor units and wire in parallel for 200 A rating when used with the 8720MC-RPS190 RPS modules.

8720MC Line Reactor Dimensions

Dimensions (catalog numbers 8720MC-LR03-032B, 8720MC-LR048B, 8720MC-LR062B, and 8720MC-LR070B)

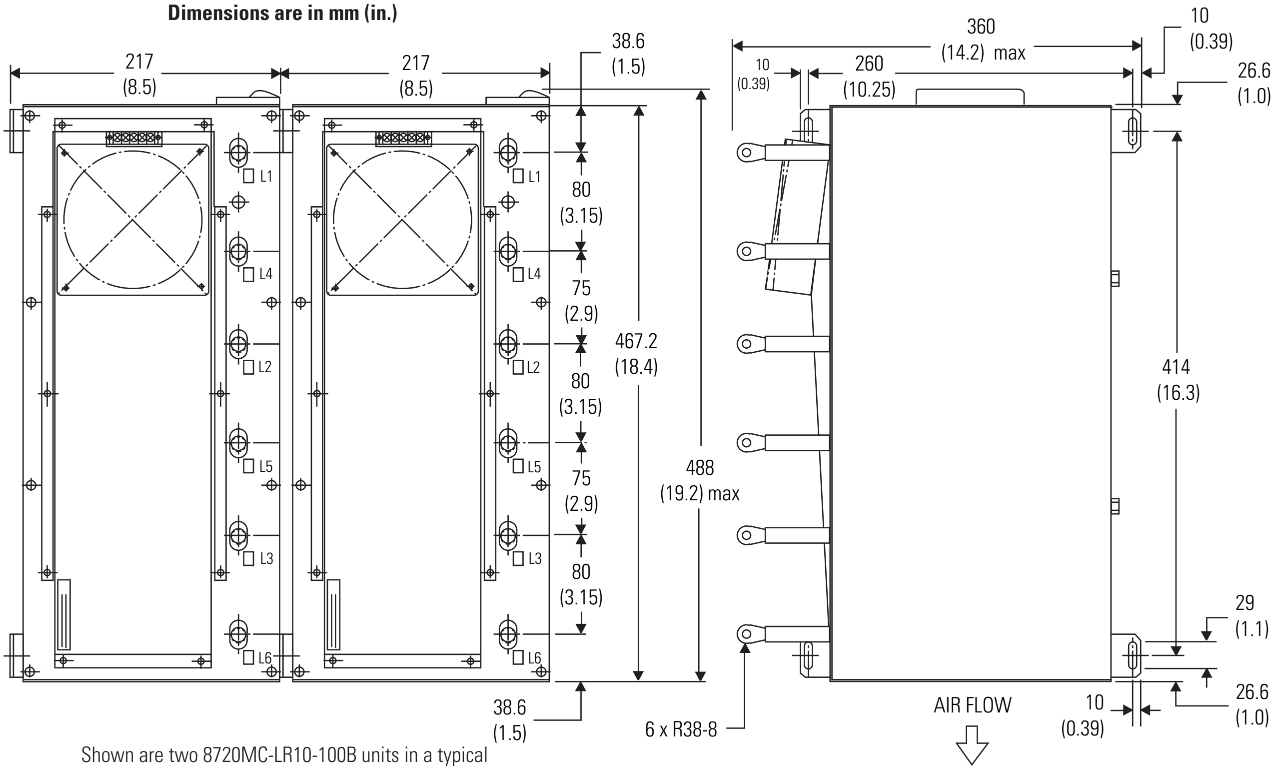


Dimensions are in mm (in.)

Line Reactor 8720MC-	**A** (1) mm (in.)	**B** mm (in.)	**C** (2) mm (in.)	**D** mm (in.)	**E** mm (in.)	**F** (3) mm (in.)	**G** mm (in.)	**H** (4) mm (in.)	**I** (5) mm (in.)	**J** mm (in.)	**K** mm (in.)	**K** mm (in.)
LR03-032B	345 (13.58)	112.5 (4.42)	120 (4.72)	112.5 (4.42)	140 (5.51)	100 (3.93)	4 to 7 (0.15 to 0.27)	127 (4.99)	80 (3.14)	15 (0.59)	4 to 15 (0.15 to 0.59)	6-(R22-6) (0.23)
LR05-048B	400 (15.74)	132.5 (5.21)	135 (5.31)	132.5 (5.21)	155 (6.10)	105 (4.13)	4 to 7 (0.15 to 0.27)	125 (4.92)	80 (3.14)	15 (0.59)	4 to 15 (0.15 to 0.59)	6-(R22-6) (0.23)
LR10-062B	440 (17.32)	145 (5.70)	150 (5.90)	145 (5.70)	160 (6.29)	110 (4.33)	4 to 9.5 (0.15 to 0.37)	125 (4.92)	80 (3.14)	15 (0.59)	4 to 15 (0.15 to 0.59)	6-(R22-6) (0.23)
LR14-070B	460 (18.11)	155 (6.10)	150 (5.90)	155 (6.10)	180 (7.08)	125 (4.92)	4 to 9.5 (0.15 to 0.37)	140 (5.51)	80 (3.14)	15 (0.59)	4 to 15 (0.15 to 0.59)	6-(R38-6) (0.23)

(1) The tolerance is +/-2 mm (0.07 in.).
(2) The tolerance is +/-1 mm (0.03 in.).
(3) The tolerance is +1/-5 mm (+0.03/-0.19 in.).
(4) The tolerance is +/-5 mm (0.19 in.).
(5) The tolerance is +/-10 mm (0.39 in.).

Dimensions (catalog number 8720MC-LR10-100B)



Shown are two 8720MC-LR10-100B units in a typical configuration for use with the 8720MC-RPS190 RPS module.

8720MC Line Reactor Catalog Numbers

Catalog numbers consist of various characters, each of which identifies a specific option for that component. Use the catalog numbering table chart below to understand the configuration of your 8720MC Line Reactors. For questions regarding product availability, contact your Allen-Bradley distributor.

8720MC - LR*xx* *x* *x* *x*

Voltage
B = 380…460V AC

Current Rating
032 = 32 A
048 = 48 A
062 = 62 A
070 = 70 A
100 = 200 A (two 100 A units)

Enclosure Type
Blank = Open
A = NEMA Type 1

Inductance as % Voltage Drop
03 = 3%
05 = 5%
10 = 10%
14 = 14%

Bulletin Number

Notes:

Rotary Motion System Combinations

This chapter provides Kinetix Motion Control servo drive and rotary motor combinations. Each drive family/motor section includes:

- a motor/cable combinations table.
- a drive/motor performance specification table.
- torque/speed curves with each motor matched to the drive with optimum performance.

Performance specification data and curves reflect nominal system performance of a typical system with motor/drive at rated ambient temperature and line voltage. For additional information on ambients, line conditions, and valid combinations not shown in this chapter, refer to Motion Analyzer software.

IMPORTANT This system combinations chapter does not include all possible motor/drive combinations. Please refer to Motion Analyzer software to verify compatibility. Download is available at http://www.ab.com/motion/software/analyzer.html.

Rotary Motion System Combinations

Drive Family	Rotary Motor Series	Class	Page
Kinetix 6000 and Kinetix 6200/ Kinetix 6500 Servo Drives	Kinetix 6000 Peak Enhancement Example		512
	MP-Series Low Inertia Motors	200V and 400V	513
	MP-Series Medium Inertia Motors		528
	MP-Series Food Grade Motors		539
	MP-Series Stainless Steel Motors		546
	RDD-Series Direct Drive Motors	400V	549
	TL-Series (Bulletin TLY) Motors	200V	555
Kinetix 2000 Servo Drives	MP-Series Low Inertia Motors	200V	559
	MP-Series Medium Inertia Motors		564
	MP-Series Food Grade Motors		565
	MP-Series Stainless Steel Motors		567
	TL-Series (Bulletin TLY) Motors		568
Kinetix 7000 Servo Drives	HPK-Series Asychronous Servo Motors	400V	574
	MP-Series Low Inertia Motors		581
	MP-Series Medium Inertia Motors		586
	RDD-Series Direct Drive Motors		591

Drive Family	Rotary Motor Series	Class	Page
Kinetix 300 Servo Drives	MP-Series Low Inertia Motors	200V and 400V	595
	MP-Series Medium Inertia Motors		604
	MP-Series Food Grade Motors		607
	MP-Series Stainless Steel Motors		611
	TL-Series (Bulletin TLY) Motors	200V	613
Kinetix 3 Servo Drives	TL-Series (Bulletin TLY) Motors	200V	619
	TL-Series (Bulletin TL) Motors		624
Ultra3000 and Ultra5000 Servo Drives	MP-Series Low Inertia Motors	200V and 400V	627
	MP-Series Medium Inertia Motors		641
	MP-Series Food Grade Motors		652
	MP-Series Stainless Steel Motors		657
	TL-Series (Bulletin TLY) Motors	200V	660

IMPORTANT You can configure Kinetix 6000 460V (series B) drives to operate with up to 250% peak current for limited duty cycles. Drive/actuator performance specifications are given with and without the peak enhancement feature enabled. For more information, refer to Kinetix 6000 Drive Performance Example with Peak Enhancement Feature on page 512.

Kinetix 6000 Drive Performance Example with Peak Enhancement Feature

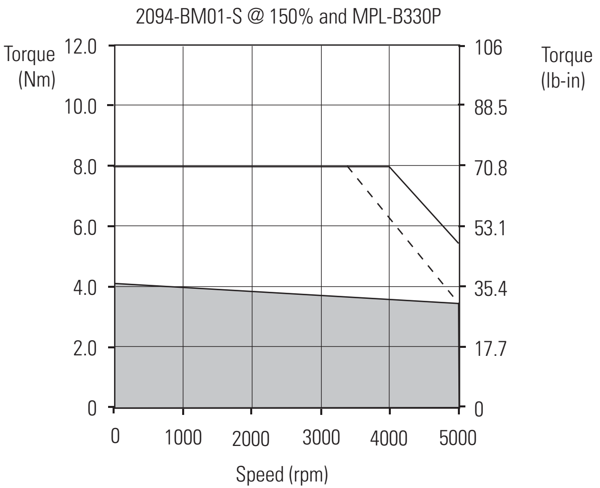
The peak current ratings of the Kinetix 6000 AM modules (series A and B) are configured at the factory as 150% of continuous current. You can program 460V (series B) AM modules and the equivalent IAM (inverter) modules, to operate with up to 250% of continuous inverter current. Refer to Peak Enhancement Specifications on page 281 for more information.

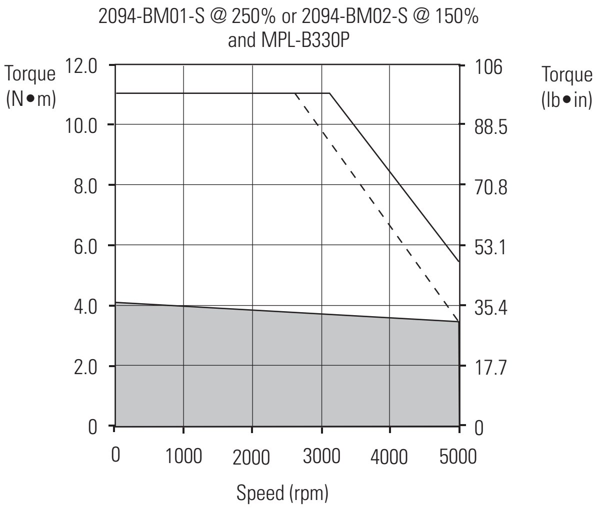
IMPORTANT	Before your Kinetix 6000 drive will deliver enhanced peak performance, you must enable the peak enhancement feature by configuring your drive by using DriveExplorer or RSLogix 5000 software. Refer to the Kinetix 6000 Multi-axis Servo Drive User Manual, publication 2094-UM001, to recalculate torque and accel/decel limit values, and paste them into the appropriate Axis Properties dialog box in RSLogix 5000 software. For sizing your drive/motor combination by using series-B drives and the peak enhancement feature, use Motion Analyzer software, version 4.6 or later.

In this example, the MPL-B330P motor, usually paired with the 2094-BM02 (series A) AM module, is shown paired with the 2094-BM01-S (series B) AM module. The two curves illustrate how the 2094-BM01-S (series B) drive, when configured for 250% peak, can achieve full motor performance.

Rotary Motor Performance Specifications Example with Kinetix 6000 Drives

Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 6000 460V Drives
MPL-B330P	5000	6.10	4.18 (37)	13.0	8.0 (71)	1.8	2094-BM01-S @ 150%
				19.0	11.1 (98)		2094-BM01-S @ 250%
							2094-BM02-S @ 150%


2094-BM01-S @ 150% and MPL-B330P


2094-BM01-S @ 250% or 2094-BM02-S @ 150% and MPL-B330P

☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

IMPORTANT	The 2094-BC07-M05-S and 2094-BM05-S (series B) modules are limited to 200% of continuous inverter current.

Kinetix 6000 (230V) Drives with MP-Series Low Inertia Motors

This section provides system combination information for the Kinetix 6000 (230V) drives when matched with MP-Series low-inertia motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

IMPORTANT The MP-Series low-inertia motors on this page are equipped with DIN connectors (specified by 4 or 7 in the catalog number) and are not compatible with cables designed for motors equipped with bayonet connectors (specified by 2 in the catalog number). The motors with bayonet connectors (for example, MPL-A310P-*xx*2*x*AA) are being discontinued and require 2090-XXN*x*MP (bayonet) cables. For help with migration or to select bayonet cables, contact your Rockwell Automation sales representative.

Bulletin MPL Motor Cable Combinations

Motor Cat. No. (200V class)	Motor Power/Brake Cable	Motor Feedback Cable (1)
MPL-A1510V-*xx*4*x*AA, MPL-A1520U-*xx*4*x*AA, MPL-A1530U-*xx*4*x*AA	2090-XXNPMF-16S*xx* (standard) 2090-CP*x*M4DF-16AF*xx* (continuous-flex)	2090-XXNFMF-S*xx* (standard) 2090-CFBM4DF-CDAF*xx* (continuous-flex) Absolute High-resolution and Incremental Feedback
MPL-A210V-*xx*4*x*AA, MPL-A220T-*xx*4*x*AA, MPL-A230P-*xx*4*x*AA		
MPL-A310F-*xx*7*x*AA, MPL-A310P-*xx*7*x*AA, MPL-A320H-*xx*7*x*AA, MPL-A320P,-*xx*7*x*AA, MPL-A330P-*xx*7*x*AA	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or (2) 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback 2090-XXNFMF-S*xx* (standard) (3) 2090-CFBM7DF-CDAF*xx* (continuous-flex) Incremental Feedback
MPL-A420P-*xx*7*x*AA, MPL-A430H-*xx*7*x*AA		
MPL-A4530F-*xx*7*x*AA, MPL-A4540C-*xx*7*x*AA		
MPL-A430P-*xx*7*x*AA	2090-CP*x*M7DF-14AA*xx* (standard) 2090-CP*x*M7DF-14AF*xx* (continuous-flex)	
MPL-A4530K-*xx*7*x*AA, MPL-A4540F-*xx*7*x*AA, MPL-A4560F-*xx*7*x*AA		
MPL-A520K-*xx*7*x*AA	2090-CP*x*M7DF-10AA*xx* (standard) 2090-CP*x*M7DF-10AF*xx* (continuous-flex)	
MPL-A540K-*xx*7*x*AA, MPL-A560F-*xx*7*x*AA	2090-CP*x*M7DF-08AA*xx* (standard) 2090-CP*x*M7DF-08AF*xx* (continuous-flex)	

(1) Use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

(2) Applies to Kinetix 6000 drives and MPL-A3*xxx*-M/S…MPL-A5*xxx*-M/S motors with absolute high-resolution feedback.

(3) Applies to Kinetix 6000 drives and MPL-A3*xxx*-H…MPL-A45*xxx*-H motors with incremental feedback.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Bulletin MPL Motor Performance Specifications with Kinetix 6000 (230V) Drives

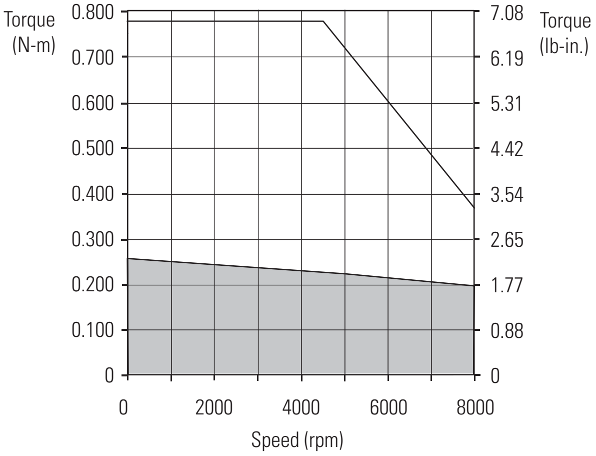
Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 6000 230V Drives
MPL-1510V	8000	1.05	0.26 (2.3)	3.40	0.77 (6.8)	0.16	2094-AMP5-S
MPL-1520U	7000	1.80	0.49 (4.3)	6.10	1.58 (13.9)	0.27	2094-AMP5-S
MPL-1530U	7000	2.82	0.90 (8.0)	10.1	2.82 (24.9)	0.39	2094-AMP5-S
MPL-210V	8000	3.09	0.55 (4.8)	10.2	1.52 (13.4)	0.37	2094-AMP5-S

Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 6000 230V Drives
MPL-220T	6000	4.54	1.61 (14.2)	10.5	3.45 (30.0)	0.62	2094-AMP5-S
				15.5	4.74 (41.9)		2094-AM01-S
MPL-230P	5000	5.40	2.10 (18.6)	17.0	8.0 (70.8)	0.86	2094-AM01-S
				23.0	8.2 (73.0)		2094-AM02-S
MPL-310F	3000	3.24	1.58 (14.0)	9.30	3.61 (31.9)	0.46	2094-AMP5-S
MPL-310P	5000	4.91	1.58 (14.0)	10.5	2.90 (25.6)	0.73	2094-AMP5-S
				14.0	3.61 (31.9)		2094-AM01-S
MPL-A320H	3500	6.10	3.05 (27.0)	17.0	7.13 (63.0)	1.0	2094-AM01-S
				19.3	7.91 (70.0)		2094-AM02-S
MPL-A320P	5000	8.50	2.88 (25.5)	17.0	5.07 (44.8)	1.3	2094-AM01-S
		9.00	3.05 (27.0)	29.5	7.91 (70.0)		2094-AM02-S
MPL-A330P	5000	12.0	4.18 (37.0)	30.0	9.10 (80.5)	1.8	2094-AM02-S
				38.0	11.1 (98.2)		2094-AM03-S
MPL-A420P	5000	12.9	4.79 (42.3)	30.0	9.67 (85.5)	2.0	2094-AM02-S
				46.0	13.6 (119)		2094-AM03-S
MPL-A430H	3500	12.2	6.21 (55.0)	30.0	13.9 (123)	1.8	2094-AM02-S
				45.0	19.8 (175)		2094-AM03-S
MPL-A430P	5000	15.0	5.35 (47.3)	30.0	9.99 (88.3)	2.2	2094-AM02-S
		16.80	5.99 (52.9)	49.0	15.4 (136)		2094-AM03-S
				67.0	19.8 (175)		2094-AM05-S
MPL-A4530F	2800	13.40	8.36 (74.0)	30.0	15.8 (139)	1.9	2094-AM02-S
				42.0	20.3 (179)		2094-AM03-S
MPL-A4530K	4000	19.50	8.13 (71.9)	49.0	17.0 (150)	2.5	2094-AM03-S
				62.0	20.3 (179)		2094-AM05-S
MPL-A4540C	1500	8.50	9.15 (80.9)	17.0	16.9 (150)	1.5	2094-AM01-S
		9.55	10.30 (91.1)	29.0	27.1 (239)		2094-AM02-S
MPL-A4540F	3000	18.40	10.19 (90.1)	49.0	23.6 (208)	2.6	2094-AM03-S
				58.0	27.1 (239)		2094-AM05-S
MPL-A4560F	3000	22.0	14.1 (125)	49.0	27.0 (239)	3.0	2094-AM03-S
				66.0	34.4 (305)		2094-AM05-S
MPL-A520K	4000	23.31	10.77 (95.2)	49.0	19.3 (171)	3.5	2094-AM03-S
				65.0	24.2 (214)		2094-AM05-S
MPL-A540K	4000	42.0	19.42 (171)	73.4	31.3 (277)	5.5	2094-AM05-S
MPL-A560F	3000	42.7	27.39 (242)	73.4	39.6 (350)	5.3	2094-AM05-S

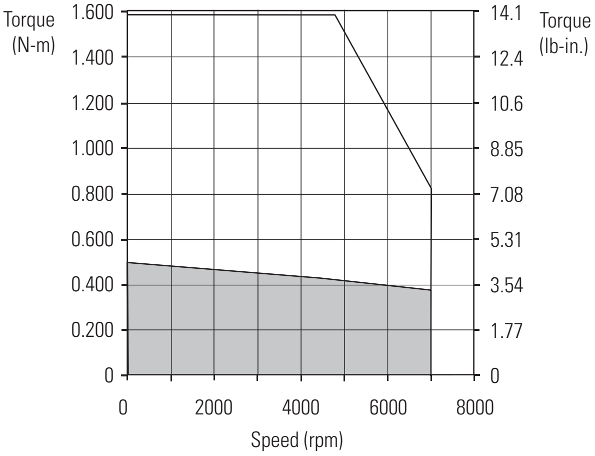
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 6000 (230V) Drives/MP-Series Low Inertia Motor Curves

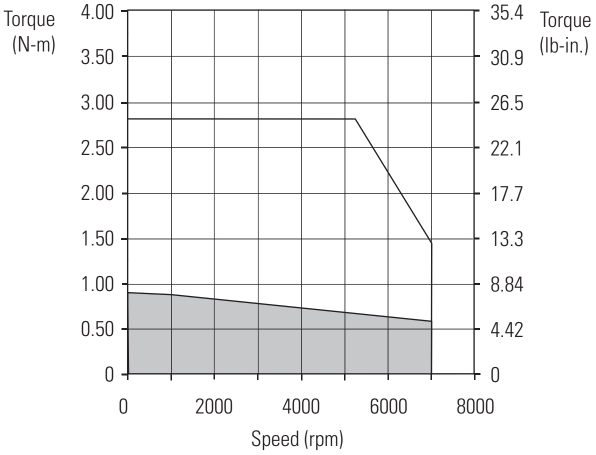
2094-AMP5-S and MPL-A1510V



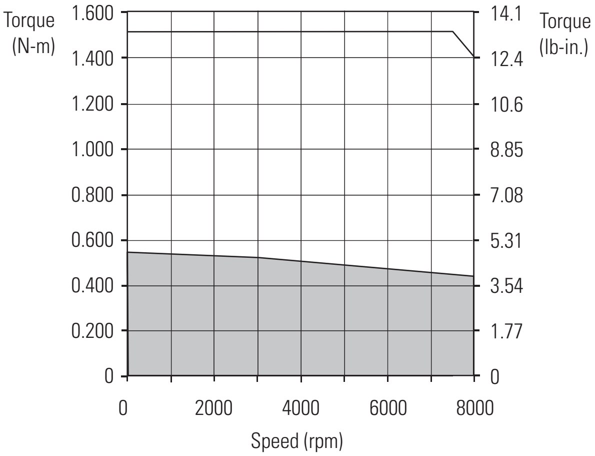
2094-AMP5-S and MPL-A1520U



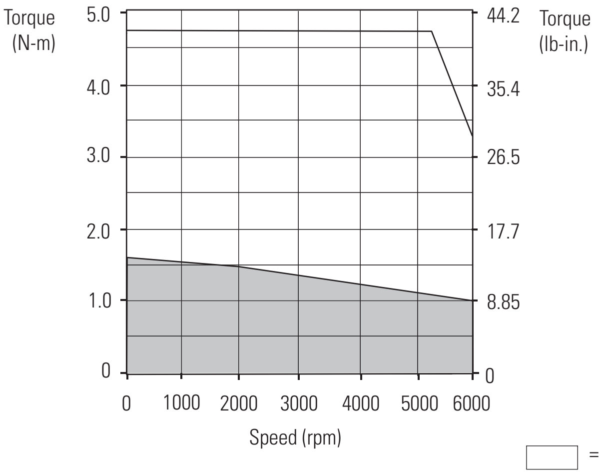
2094-AMP5-S and MPL-A1530U



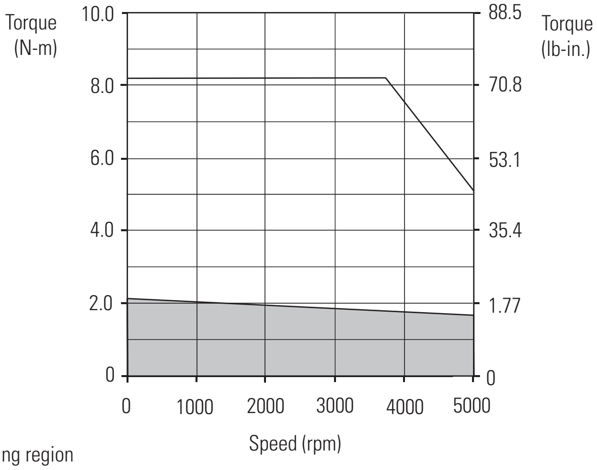
2094-AMP5-S and MPL-A210V



2094-AM01-S and MPL-A220T



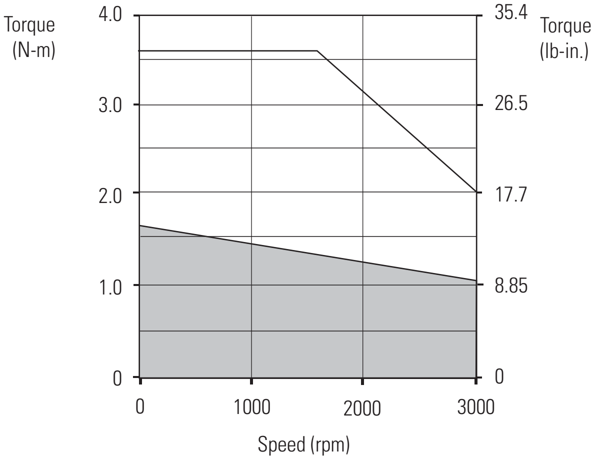
2094-AM02-S and MPL-A230P



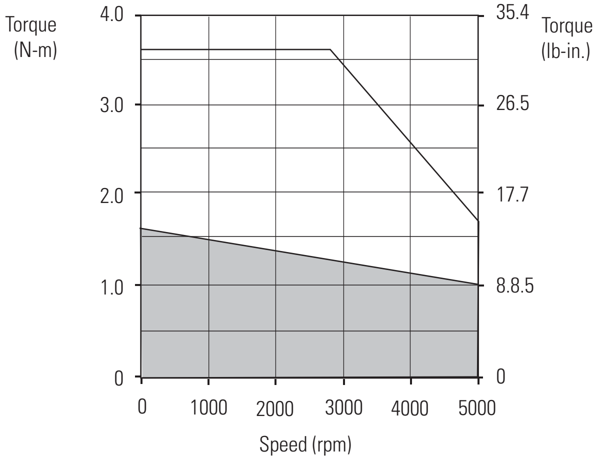
☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 6000 (230V) Drives/MP-Series Low Inertia Motor Curves, Continued

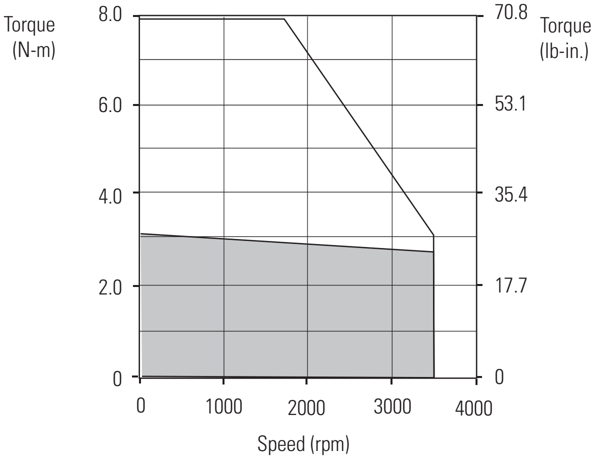
2094-AMP5-S and MPL-A310F



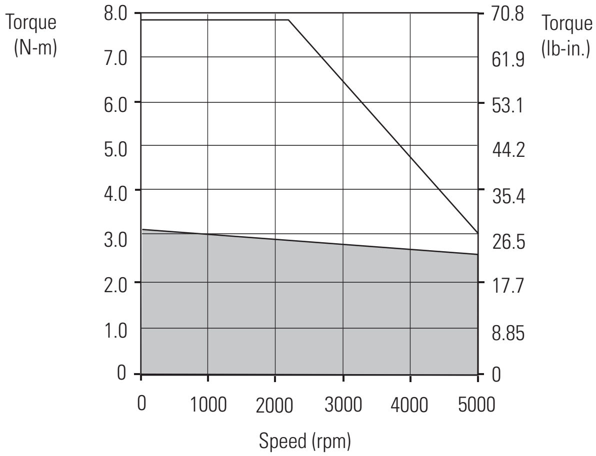
2094-AM01-S and MPL-A310P



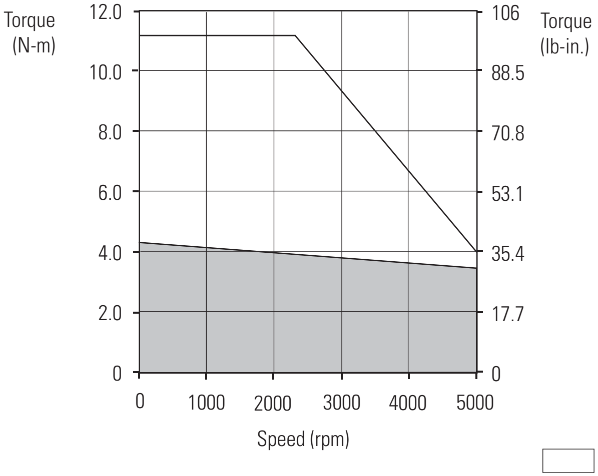
2094-AM02-S and MPL-A320H



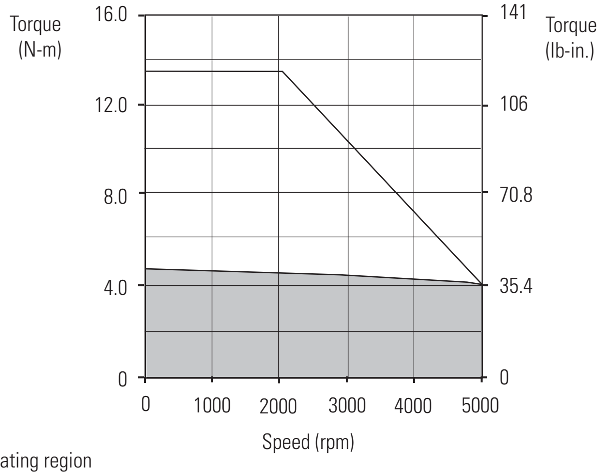
2094-AM02-S and MPL-A320P



2094-AM03-S and MPL-A330P



2094-AM03-S and MPL-A420P

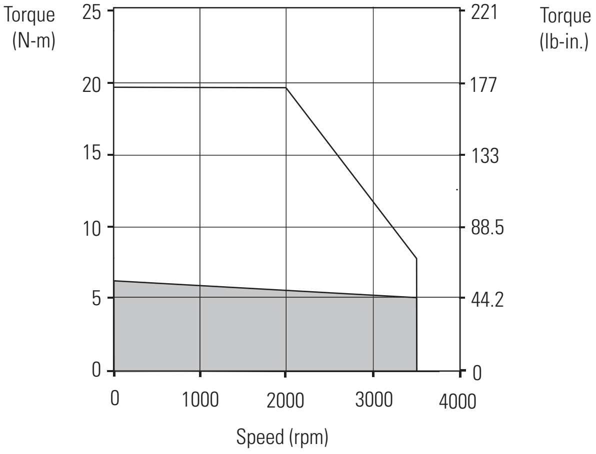


□ = Intermittent operating region

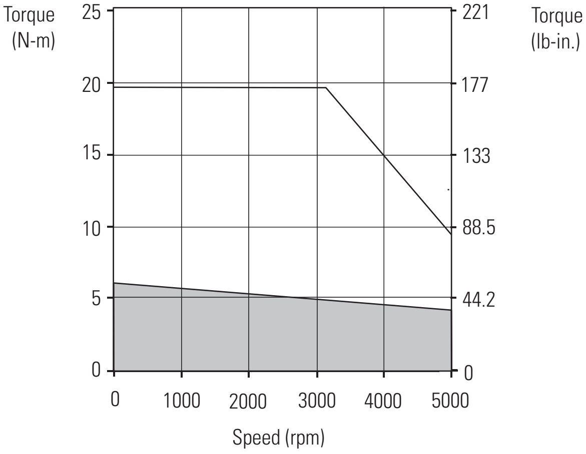
▨ = Continuous operating region

Kinetix 6000 (230V) Drives/MP-Series Low Inertia Motor Curves, Continued

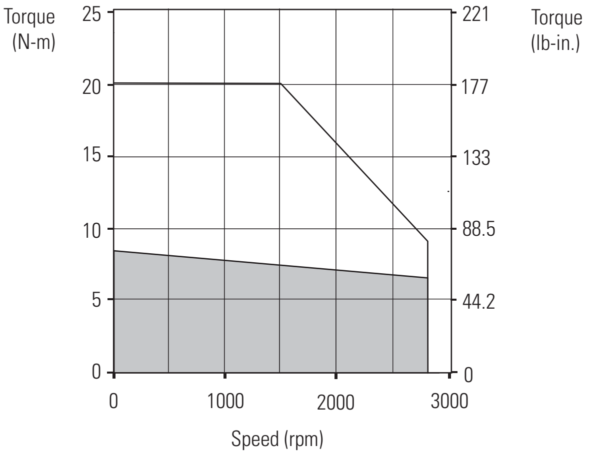
2094-AM03-S and MPL-A430H



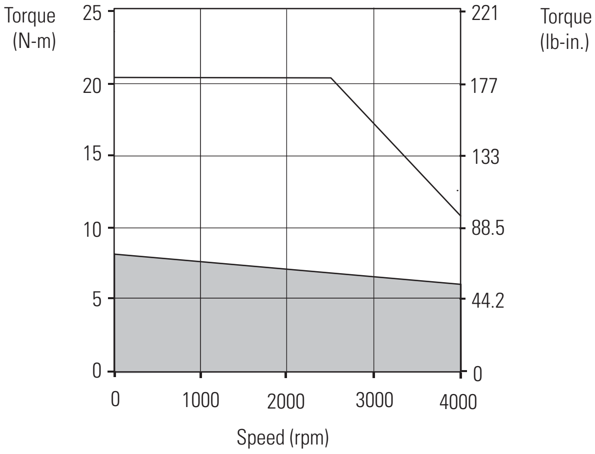
2094-AM05-S and MPL-A430P



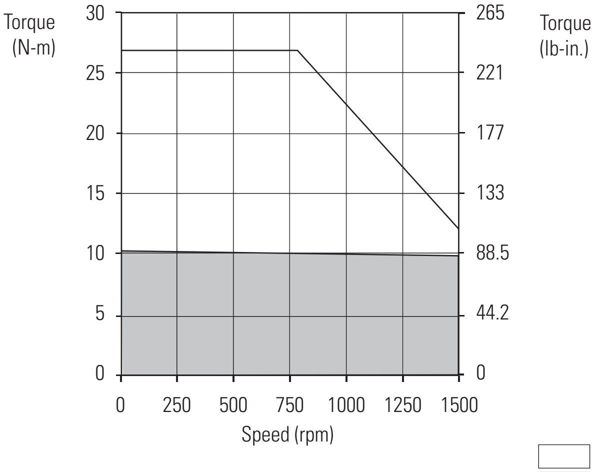
2094-AM03-S and MPL-A4530F



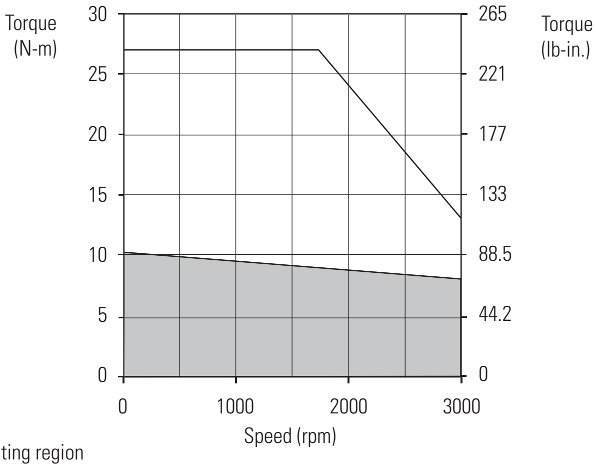
2094-AM05-S and MPL-A4530K



2094-AM02-S and MPL-A4540C



2094-AM05-S and MPL-A4540F

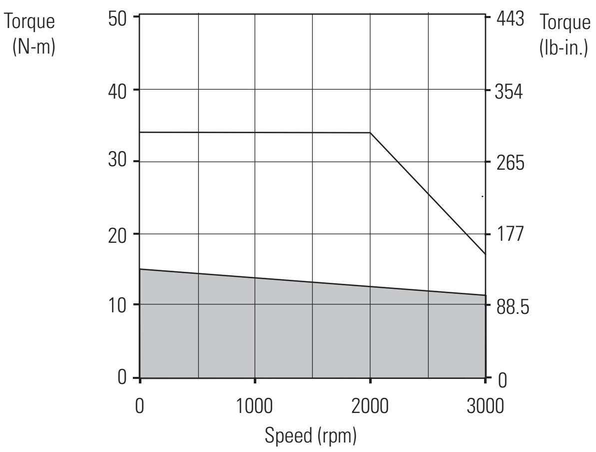


☐ = Intermittent operating region

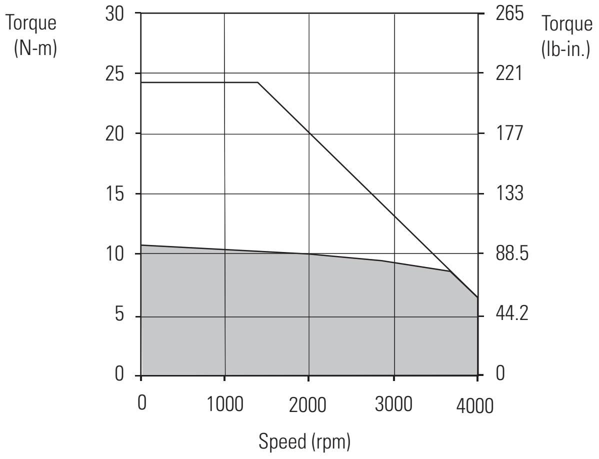
▨ = Continuous operating region

Kinetix 6000 (230V) Drives/MP-Series Low Inertia Motor Curves, Continued

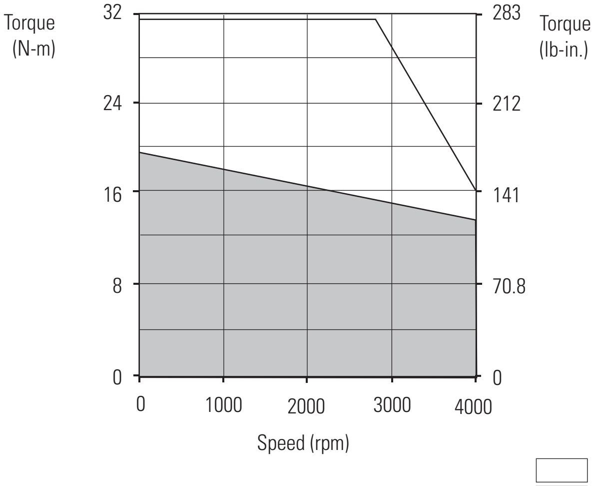
2094-AM05-S and MPL-A4560F



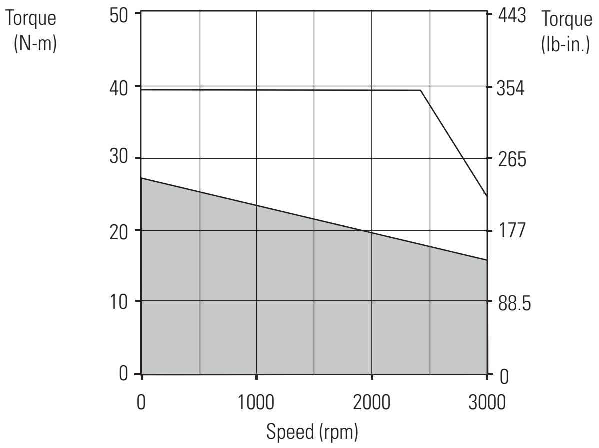
2094-AM05-S and MPL-A520K



2094-AM05-S and MPL-A540K



2094-AM05-S and MPL-A560F



☐ = Intermittent operating region

▨ = Continuous operating region

Kinetix 6000 and Kinetix 6200/6500 (460V) Drives with MP-Series Low Inertia Motors

This section provides system combination information for the Kinetix 6000 and the Kinetix 6200/6500 (460V) drives when matched with MP-Series low-inertia motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

IMPORTANT When using the Kinetix 6000 (series B) drives, configured for enhanced peak performance, you can usually achieve full motor performance with a smaller drive. Kinetix 6200 and Kinetix 6500 drives are configured for enhanced peak performance by default. Expect the same peak performance from Kinetix 6200/6500 drives and Kinetix 6000 (series B) drives configured for enhanced peak performance.

Refer to Kinetix 6000 IAM/AM Module Series Change on page 276 for more information about using the peak enhancement feature.

IMPORTANT The MP-Series low-inertia motors on this page are equipped with DIN connectors (specified by 4 or 7 in the catalog number) and are not compatible with cables designed for motors equipped with bayonet connectors (specified by 2 in the catalog number). The motors with bayonet connectors (for example, MPL-A310P-*xx*2*x*AA) are being discontinued and require 2090-XXN*x*MP (bayonet) cables. For help with migration or to select bayonet cables, contact your Rockwell Automation sales representative.

Bulletin MPL Motor Cable Combinations

Motor Cat. No. (400V class)	Motor Power/Brake Cable	Motor Feedback Cable (1)
MPL-B1510V-*xx*4*x*AA, MPL-B1520U-*xx*4*x*AA, MPL-B1530U-*xx*4*x*AA	2090-XXNPMF-16S*xx* (standard) 2090-CP*x*M4DF-16AF*xx* (continuous-flex)	2090-XXNFMF-S*xx* (standard) 2090-CFBM4DF-CDAF*xx* (continuous-flex) Absolute High-resolution and Incremental Feedback
MPL-B210V-*xx*4*x*AA, MPL-B220T-*xx*4*x*AA, MPL-B230P-*xx*4*x*AA		
MPL-B310P-*xx*7*x*AA, MPL-B320P-*xx*7*x*AA, MPL-B330P-*xx*7*x*AA	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or (2) 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback 2090-CFBM7DF-CEAA*xx* (standard) (3) 2090-CFBM7DF-CEAF*xx* (continuous-flex) Resolver Feedback
MPL-B420P-*xx*7*x*AA, MPL-B430P-*xx*7*x*AA		
MPL-B4530F-*xx*7*x*AA, MPL-B4530K-*xx*7*x*AA, MPL-B4540F-*xx*7*x*AA, MPL-B4560F-*xx*7*x*AA		
MPL-B520K-*xx*7*x*AA		
MPL-B540D-*xx*7*x*AA, MPL-B540K-*xx*7*x*AA, MPL-B560F-*xx*7*x*AA	2090-CP*x*M7DF-14AA*xx* (standard) 2090-CP*x*M7DF-14AF*xx* (continuous-flex)	
MPL-B580F-*xx*7*x*AA, MPL-B580J-*xx*7*x*AA, MPL-B640F-*xx*7*x*AA	2090-CP*x*M7DF-10AA*xx* (standard) 2090-CP*x*M7DF-10AF*xx* (continuous-flex)	
MPL-B660F-*xx*7*x*AA, MPL-B680D-*xx*7*x*AA, MPL-B960B-*xx*7*x*AA, MPL-B980B-*xx*7*x*AA	2090-CP*x*M7DF-08AA*xx* (standard) 2090-CP*x*M7DF-08AF*xx* (continuous-flex)	
MPL-B680F-*xx*7*x*AA, MPL-B860D-*xx*7*x*AA, MPL-B880C-*xx*7*x*AA	2090-CPBM7DF-06AA*xx* (standard)	
MPL-B880D-*xx*7*x*AA	2090-CPBM7DF-04AA*xx* (standard)	

(1) Use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

(2) Applies to Kinetix 6000 drives and MPL-B3*xxx*-M/S…MPL-B9*xxx*-M/S motors with absolute high-resolution feedback.

(3) Applies to Kinetix 6000 drives and MPL-B3*xxx*-R…MPL-B45*xxx*-R motors with resolver feedback.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Bulletin MPL Motor Performance Specifications with Kinetix 6200/6500 (460V) Drives

Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 6200/ Kinetix 6500 460V Drives
MPL-B1510V	8000	0.95	0.26 (2.3)	3.10	0.77 (6.8)	0.16	2094-BMP5-M
MPL-B1520U	7000	1.80	0.49 (4.3)	6.10	1.58 (13.9)	0.27	2094-BMP5-M
MPL-B1530U	7000	2.0	0.90 (8.0)	7.20	2.82 (24.9)	0.39	2094-BMP5-M
MPL-B210V	8000	1.75	0.55 (4.9)	5.80	1.52 (13.4)	0.37	2094-BMP5-M
MPL-B220T	6000	3.30	1.61 (14.2)	9.90	4.12 (36.4)	0.62	2094-BMP5-M
				11.3	4.74 (41.9)		2094-BM01-M
MPL-B230P	5000	2.60	2.10 (18.6)	9.90	7.24 (64.0)	0.86	2094-BMP5-M
				11.3	8.20 (73.0)		2094-BM01-M
MPL-B310P	5000	2.4	1.6 (14.1)	7.10	3.6 (32)	0.77	2094-BMP5-M
MPL-B320P	5000	4.0	2.7 (23.9)	9.90	5.9 (52.2)	1.5	2094-BMP5-M
		4.5	3.10 (27)	14.0	8.2 (72.5)		2094-BM01-M
MPL-B330P	5000	4.0	2.7 (23.9)	9.90	6.8 (60.2)	1.8	2094-BMP5-M
		6.1	4.18 (37)	19.0	11.1 (98)		2094-BM01-M
MPL-B420P	5000	6.3	4.74 (42)	21.6	13.1 (116)	1.9	2094-BM01-M
				22.0	13.5 (119)		2094-BM02-M
MPL-B430P	5000	8.6	6.2 (54.9)	21.6	13.9 (123)	2.2	2094-BM01-M
		9.2	6.55 (58)	32.0	19.8 (175)		2094-BM02-M
MPL-B4530F	3000	4.0	4.9 (43.3)	9.90	11.0 (97.3)	2.1	2094-BMP5-M
		6.7	8.36 (74)	21.0	20.3 (180)		2094-BM01-M
MPL-B4530K	4000	8.6	7.1 (62.8)	21.6	15.1 (133)	2.6	2094-BM01-M
		9.9	8.25 (73)	31.0	20.3 (179)		2094-BM02-M
MPL-B4540F	3000	8.6	9.5 (84.1)	21.6	20.9 (185)	2.6	2094-BM01-M
		9.1	10.20 (90)	29.0	27.1 (240)		2094-BM02-M
MPL-B4560F	3000	8.6	10.5 (92.9)	21.6	22.7 (201)	3.2	2094-BM01-M
		11.8	14.0 (124)	36.0	34.4 (304)		2094-BM02-M
MPL-B520K	4000	8.6	7.9 (69.9)	21.6	16.6 (147)	3.5	2094-BM01-M
		11.5	10.7 (95)	33.0	23.2 (205)		2094-BM02-M
MPL-B540D	2000	8.6	15.8 (139)	21.6	37.9 (335)	3.4	2094-BM01-M
		10.5	19.4 (172)	23.0	41.0 (362)		2094-BM02-M
MPL-B540K	4000	20.4	19.4 (171)	60.0	48.6 (430)	5.4	2094-BM03-M
MPL-B560F	3000	20.9	26.8 (237)	68.0	67.8 (600)	5.5	2094-BM03-M
MPL-B580F	3000	26.1	34.0 (300)	75.0	74.6 (660)	7.1	2094-BM03-M
				94.0	87.0 (770)		2094-BM05-M
MPL-B580J	3800	30.0	31.7 (280)	75.0	67.0 (592)	7.9	2094-BM03-M
		32.0	34.0 (301)	94.0	81.0 (716)		2094-BM05-M
MPL-B640F	3000	30.0	34.4 (304)	65.0	72.3 (640)	6.1	2094-BM03-M
		32.0	36.7 (325)				2094-BM05-M

Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 6200/ Kinetix 6500 460V Drives
MPL-B660F	3000	38.5	48.0 (425)	96.0	101 (895)	6.1	2094-BM05-M
MPL-B680D	2000	30.0	55.4 (490)	75.0	125 (1105)	9.3	2094-BM03-M
		34.0	62.8 (556)	94.0	154 (1365)		2094-BM05-M
MPL-B680F	3000	47.9	60.0 (531)	96.0	108 (960)	7.5	2094-BM05-M
MPL-B860D	2000	47.3	83.0 (735)	95.5	152 (1350)	12.5	2094-BM05-M
MPL-B880C	1500	47.5	110 (973)	97.5	203 (1800)	12.6	2094-BM05-M
MPL-B880D	2000	48.9	79.9 (706)	96.0	147 (1300)	12.6	2094-BM05-M
MPL-B960B	1200	42.5	130 (1150)	94.0	231 (2050)	12.7	2094-BM05-M
MPL-B980B	1000	40.0	162 (1440)	94.0	278 (2460)	15.2	2094-BM05-M

Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 5.000 or later.

Bulletin MPL Motor Performance Specifications with Kinetix 6000 (460V) Drives

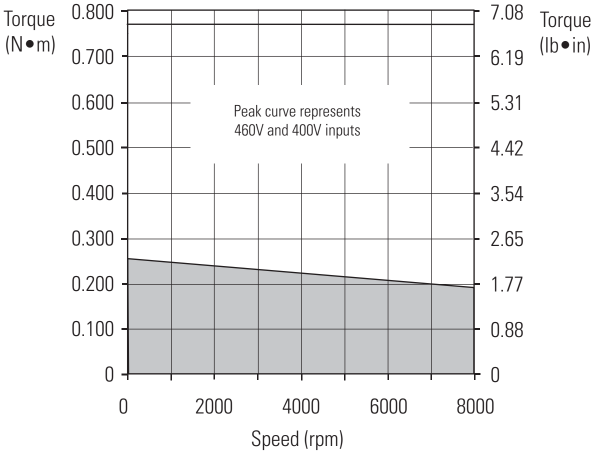
Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 6000 460V Drives
MPL-B1510V	8000	0.95	0.26 (2.3)	3.10	0.77 (6.8)	0.16	BMP5-S @ 150%
MPL-B1520U	7000	1.80	0.49 (4.3)	5.90	1.53 (13.3)	0.27	BMP5-S @ 150%
				6.10	1.58 (13.9)		BMP5-S @ 250%
MPL-B1530U	7000	2.0	0.90 (8.0)	5.90	2.34 (20.7)	0.39	BMP5-S @ 150%
				7.20	2.82 (24.9)		BMP5-S @ 250%
MPL-B210V	8000	1.75	0.55 (4.9)	5.80	1.52 (13.4)	0.37	BMP5-S @ 150%
MPL-B220T	6000	3.30	1.61 (14.2)	9.90	4.12 (36.4)	0.62	BMP5-S @ 250%
				11.3	4.74 (41.9)		BM01-S @ 150%
MPL-B230P	5000	2.60	2.10 (18.6)	9.90	7.24 (64.0)	0.86	BMP5-S @ 250%
				11.3	8.20 (73.0)		BM01-S @ 150%
MPL-B310P	5000	2.4	1.6 (14)	5.90	3.2 (28)	0.77	BMP5-S @ 150%
				7.10	3.6 (32)		BMP5-S @ 250%
MPL-B320P	5000	4.5	3.10 (27)	13.0	7.5 (66)	1.5	BM01-S @ 150%
				14.0	8.2 (72.5)		BM01-S @ 250%
MPL-B330P	5000	6.1	4.18 (37)	13.0	8.0 (71)	1.8	BM01-S @ 150%
				19.0	11.1 (98)		BM01-S @ 250%
MPL-B420P	5000	6.3	4.74 (42)	13.0	13.1 (116)	1.9	BM01-S @ 250%
				21.8	13.4 (118)		BM02-S @ 150%
				22.0	13.5 (119)		BM02-S @ 250%
MPL-B430P	5000	9.2	6.55 (58)	21.8	14.4 (127)	2.2	BM02-S @ 150%
				32.0	19.8 (175)		BM02-S @ 250%
MPL-B4530F	3000	6.7	8.36 (74)	13.0	13.9 (123)	2.1	BM01-S @ 150%
				21.0	20.3 (180)		BM01-S @ 250%

Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Stall Torque** N•m (lb•in)	**Motor Rated Output** kW	**Kinetix 6000 460V Drives**
MPL-B4530K	4000	9.9	8.25 (73)	21.8	15.5 (137)	2.6	BM02-S @ 150%
				31.0	20.3 (179)		BM02-S @ 250%
MPL-B4540F	3000	9.1	10.20 (90)	21.8	21.4 (189)	2.6	BM02-S @ 150%
				29.0	27.1 (240)		BM02-S @ 250%
MPL-B4560F	3000	11.8	14.0 (124)	21.8	23.3 (206)	3.2	BM02-S @ 150%
				36.0	34.4 (304)		BM02-S @ 250%
MPL-B520K	4000	11.5	10.7 (95)	21.8	17.0 (150)	3.5	BM02-S @ 150%
				33.0	23.2 (205)		BM02-S @ 250%
MPL-B540D	2000	10.5	19.4 (172)	21.8	38.8 (343)	3.4	BM02-S @ 150%
				23.0	41.0 (362)		BM02-S @ 250%
MPL-B540K	4000	20.4	19.4 (171)	45.0	38.1 (337)	5.4	BM03-S @ 150%
				60.0	48.6 (430)		BM03-S @ 250%
MPL-B560F	3000	20.9	26.8 (237)	45.0	49.3 (436)	5.5	BM03-S @ 150%
				68.0	67.8 (600)		BM03-S @ 250%
MPL-B580F	3000	26.1	34.0 (300)	75.0	74.6 (660)	7.1	BM03-S @ 250%
				73.4	73.5 (650)		BM05-S @ 150%
				94.0	87.0 (770)		BM05-S @ 200%
MPL-B580J	3800	32.0	34.0 (301)	73.4	66.6 (589)	7.9	BM05-S @ 150%
				94.0	81.0 (716)		BM05-S @ 200%
MPL-B640F	3000	30.0	34.4 (304)	45.0	50.4 (446)	6.1	BM03-S @ 150%
			34.4 (304)	65.0	72.3 (640)		BM03-S @ 250%
		32.0	36.7 (325)				BM05-S @ 150%
MPL-B660F	3000	38.5	48.0 (425)	73.4	81.0 (716)	6.1	BM05-S @ 150%
				96.0	101 (895)		BM05-S @ 200%
MPL-B680D	2000	30.0	55.4 (490)	75.0	125 (1105)	9.3	BM03-S @ 250%
		34.0	62.8 (556)	73.4	124 (1098)		BM05-S @ 150%
				94.0	152 (1350)		BM05-S @ 200%
MPL-B680F	3000	47.9	60.0 (531)	73.4	85.4 (755)	7.5	BM05-S @ 150%
				96.0	108 (960)		BM05-S @ 200%
MPL-B860D	2000	47.3	83.0 (735)	73.4	120 (1065)	12.5	BM05-S @ 150%
				95.5	152 (1350)		BM05-S @ 200%
MPL-B880C	1500	47.5	110 (973)	73.4	157 (1387)	12.6	BM05-S @ 150%
				97.5	203 (1800)		BM05-S @ 200%
MPL-B880D	2000	48.9	79.9 (706)	96.0	147 (1300)	12.6	BM05-S @ 200%
MPL-B960B	1200	42.5	130 (1150)	73.4	190 (1684)	12.7	BM05-S @ 150%
				94.0	231 (2050)		BM05-S @ 200%
MPL-B980B	1000	40.0	162 (1440)	73.4	235 (2077)	15.2	BM05-S @ 150%
				94.0	278 (2460)		BM05-S @ 200%

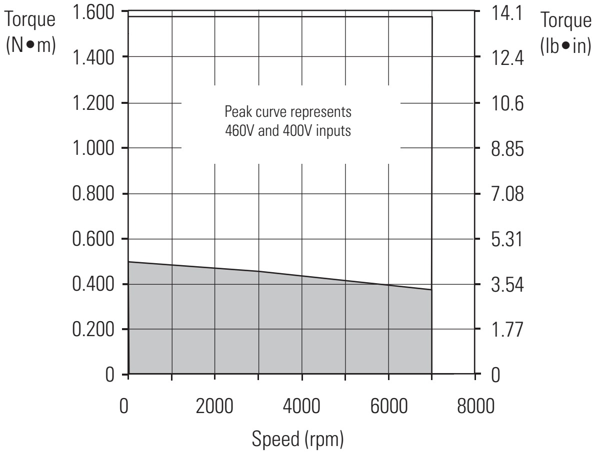
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 5.000 or later.

Kinetix 6000 and Kinetix 6200/6500 (460V) Drives/MP-Series Low Inertia Motor Curves

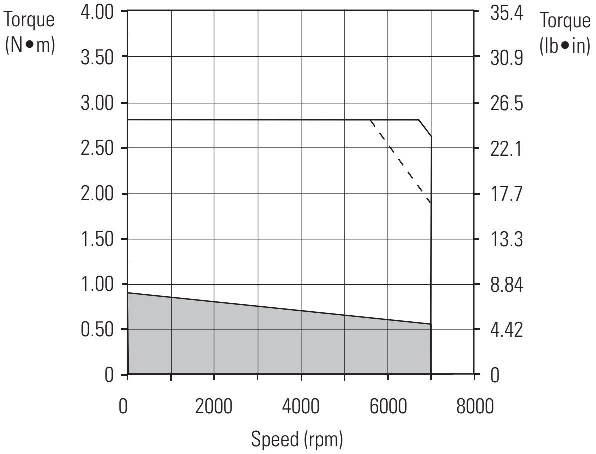
2094-BMP5-S @ 150% or 2094-BMP5-M
and MPL-B1510V



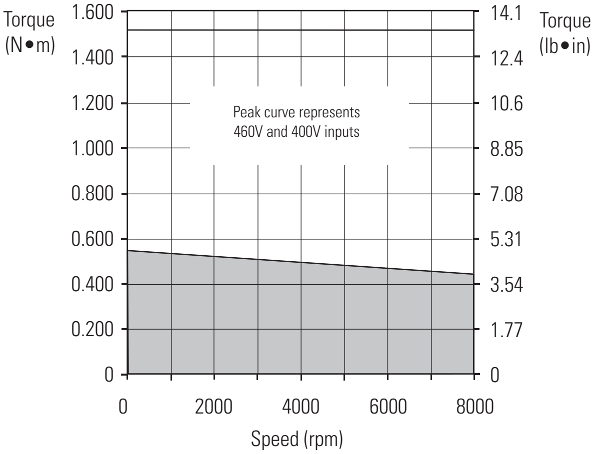
2094-BMP5-S @ 250% or 2094-BMP5-M
and MPL-B1520U



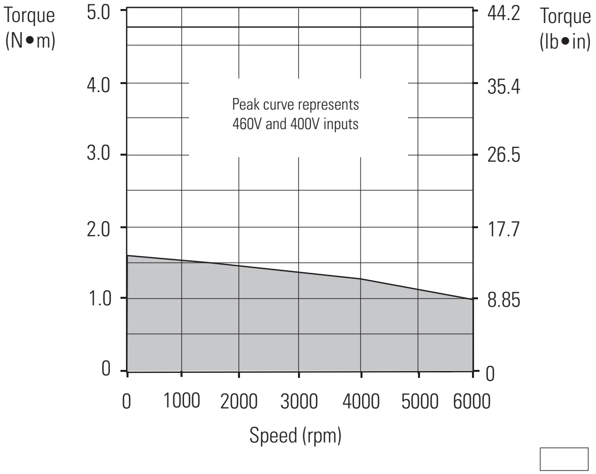
2094-BMP5-S @ 250% or 2094-BMP5-M
and MPL-B1530U



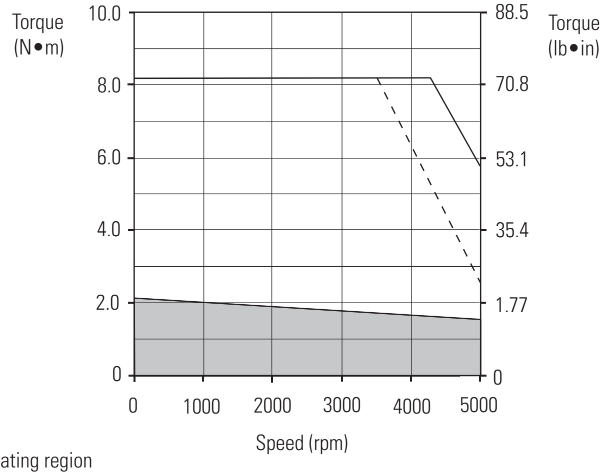
2094-BMP5-S @ 150% or 2094-BMP5-M
and MPL-B210V



2094-BM01-S @ 150% or 2094-BM01-M
and MPL-B220T



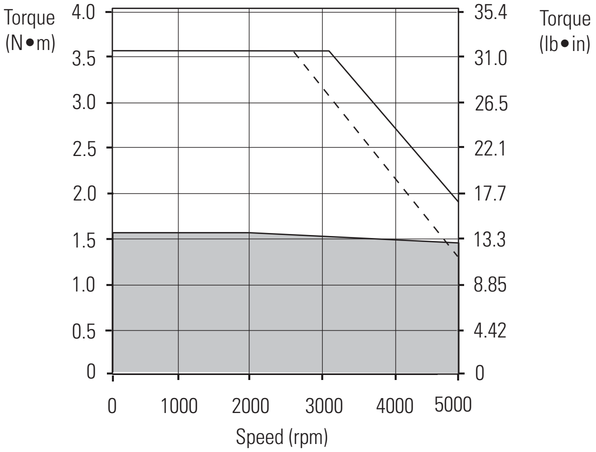
2094-BM01-S @ 150% or 2094-BM01-M
and MPL-B230P



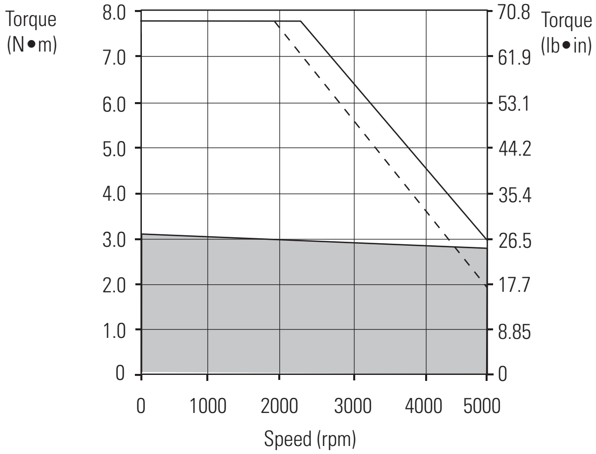
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Kinetix 6000 and Kinetix 6200/6500 (460V) Drives/MP-Series Low Inertia Motor Curves, Continued

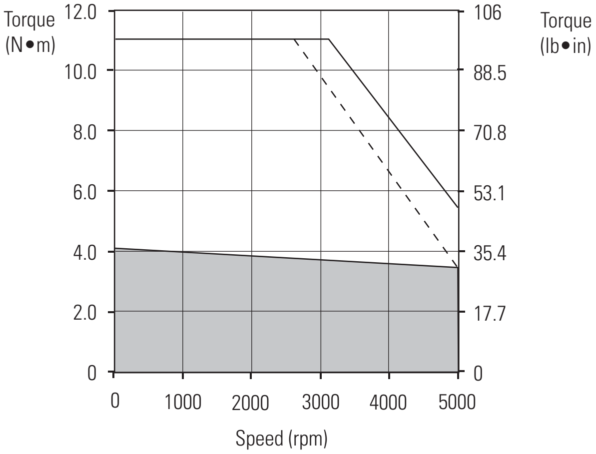
2094-BMP5-S @ 250% or 2094-BMP5-M
and MPL-B310P



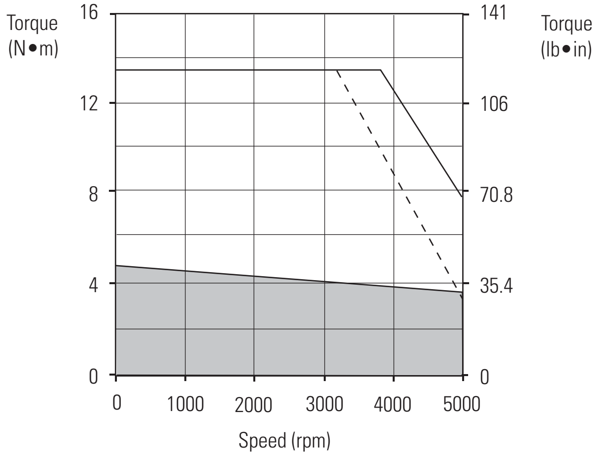
2094-BM01-S @ 250% or 2094-BM01-M
and MPL-B320P



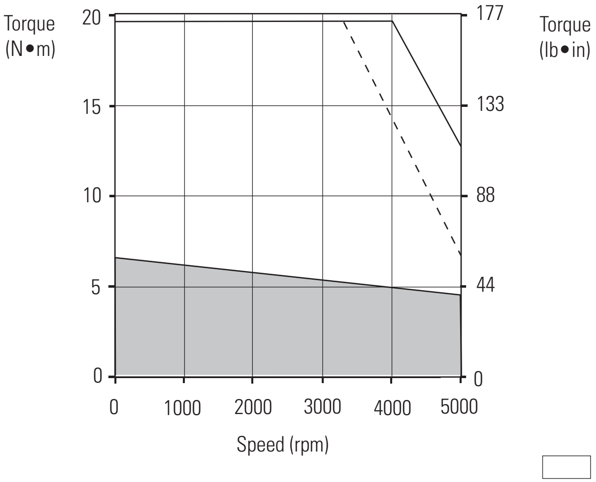
2094-BM01-S @ 250% or 2094-BM01-M
and MPL-B330P



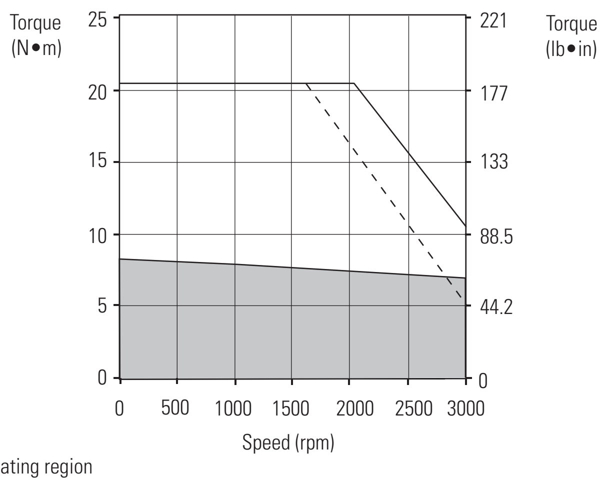
2094-BM02-S @ 250% or 2094-BM02-M
and MPL-B420P



2094-BM02-S @ 250% or 2094-BM02-M
and MPL-B430P

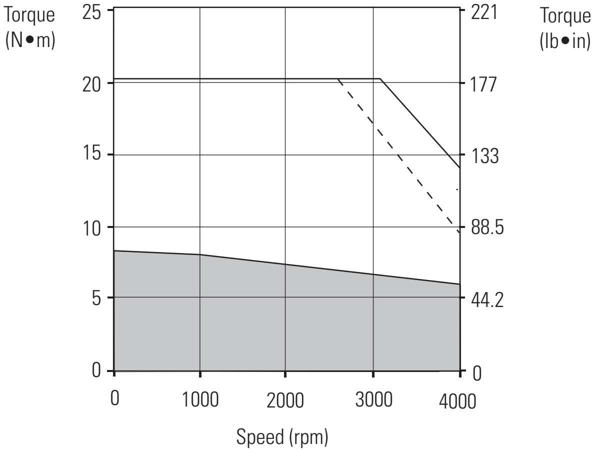


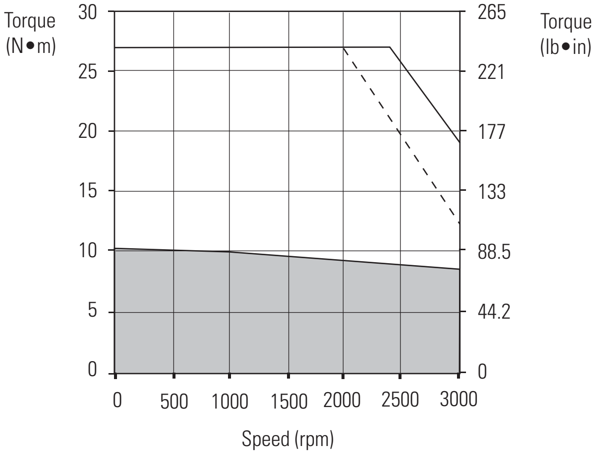
2094-BM01-S @ 250% or 2094-BM01-M
and MPL-B4530F

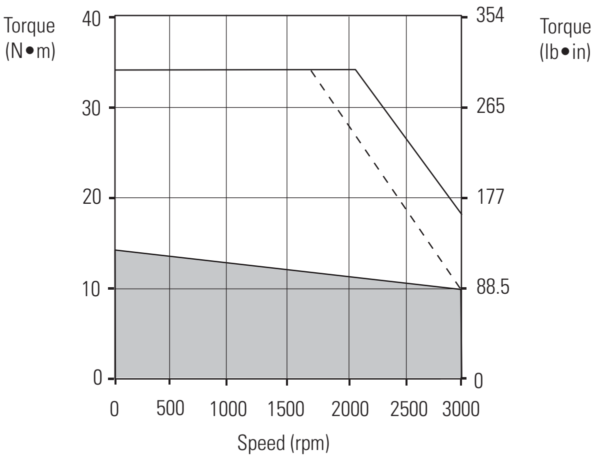


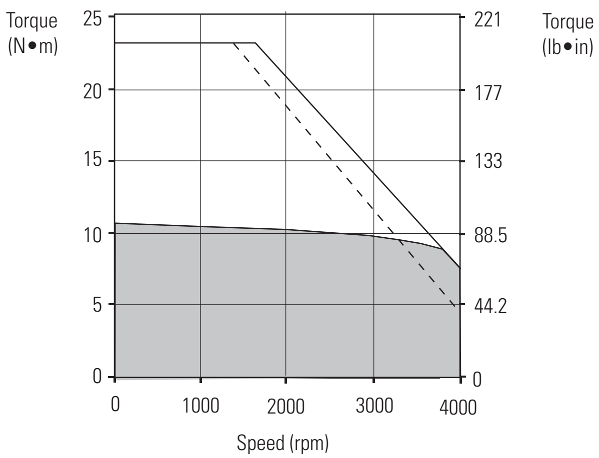
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

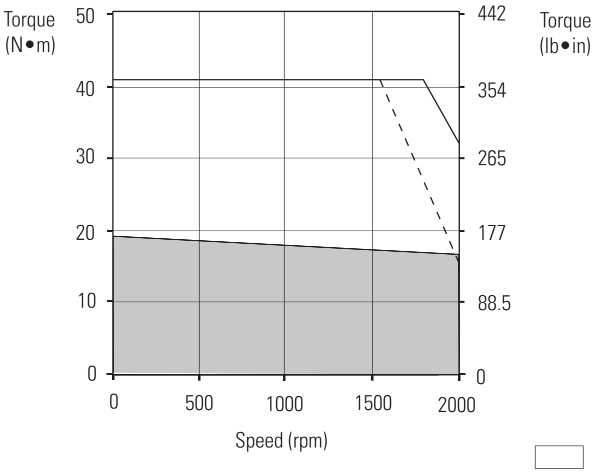
Kinetix 6000 and Kinetix 6200/6500 (460V) Drives/MP-Series Low Inertia Motor Curves, Continued

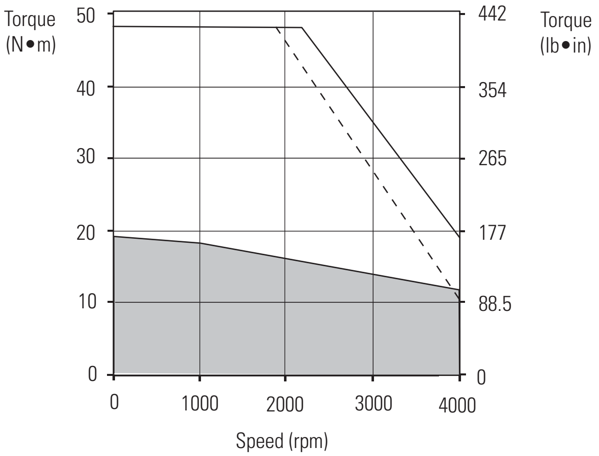

2094-BM02-S @ 250% or 2094-BM02-M
and MPL-B4530K


2094-BM02-S @ 250% or 2094-BM02-M
and MPL-B4540F


2094-BM02-S @ 250% or 2094-BM02-M
and MPL-B4560F

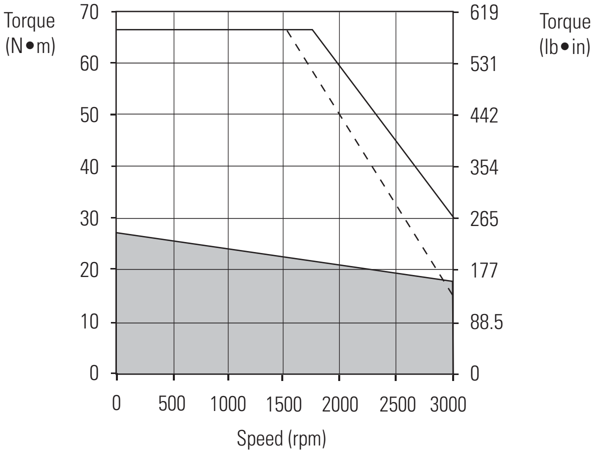

2094-BM02-S @ 250% or 2094-BM02-M
and MPL-B520K


2094-BM02-S @ 250% or 2094-BM02-M
and MPL-B540D

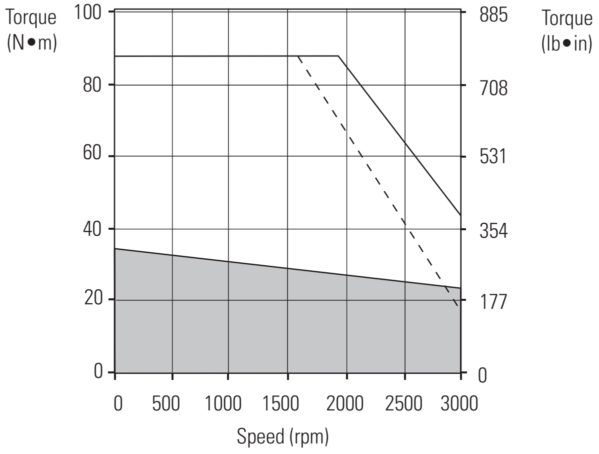

2094-BM03-S @ 250% or 2094-BM03-M
and MPL-B540K

☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

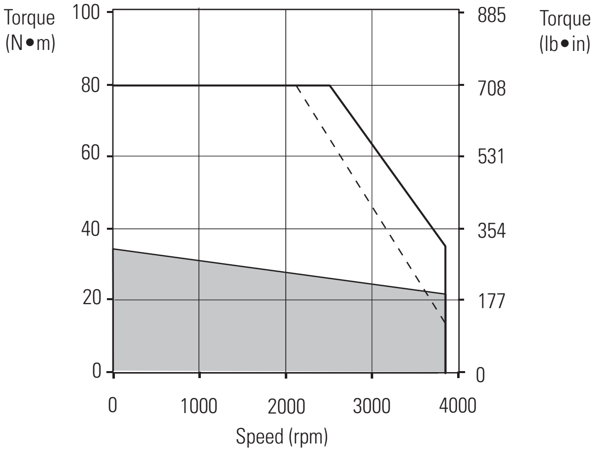
Kinetix 6000 and Kinetix 6200/6500 (460V) Drives/MP-Series Low Inertia Motor Curves, Continued



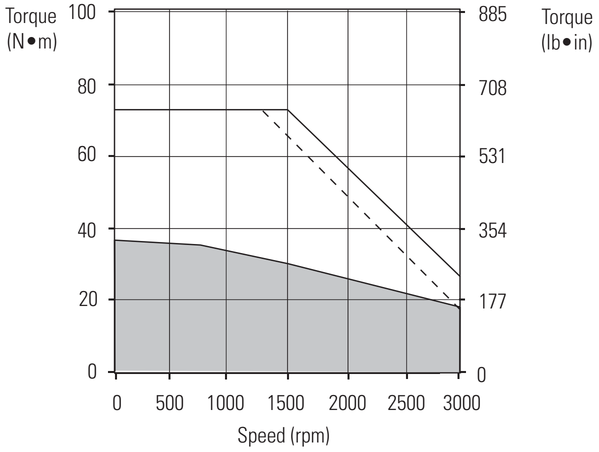
2094-BM03-S @ 250% or 2094-BM03-M
and MPL-B560F



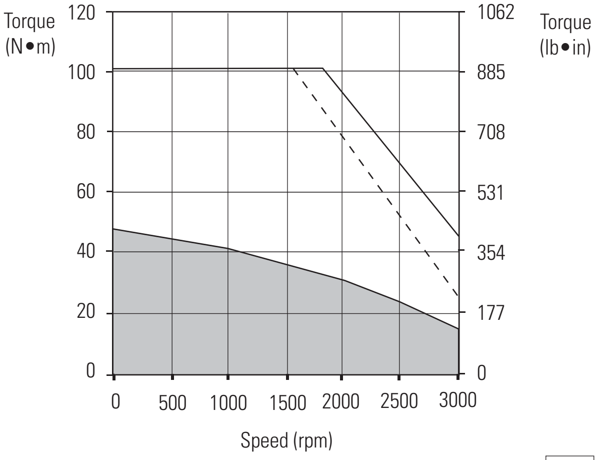
2094-BM05-S @ 200% or 2094-BM05-M
and MPL-B580F



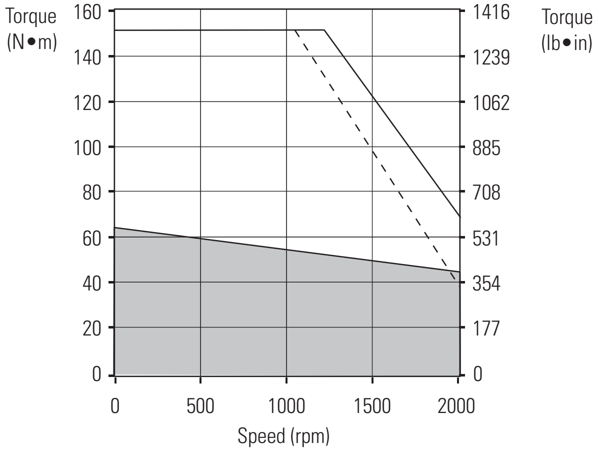
2094-BM05-S @ 200% or 2094-BM05-M
and MPL-B580J



2094-BM05-S @ 200% or 2094-BM05-M
and MPL-B640F



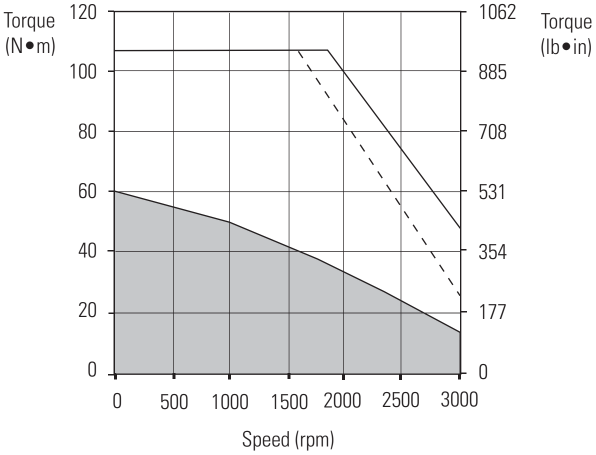
2094-BM05-S @ 200% or 2094-BM05-M
and MPL-B660F

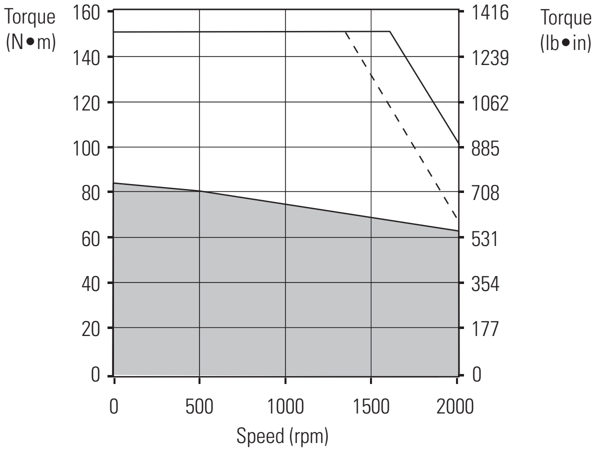


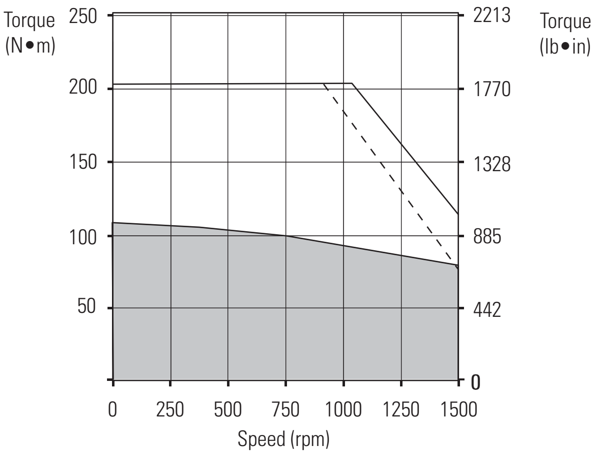
2094-BM05-S @ 200% or 2094-BM05-M
and MPL-B680D

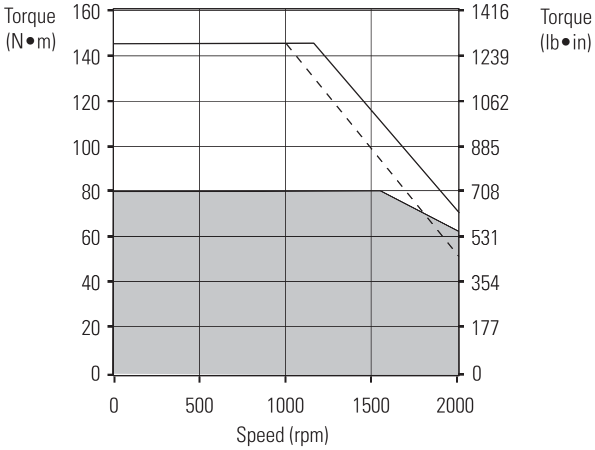
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

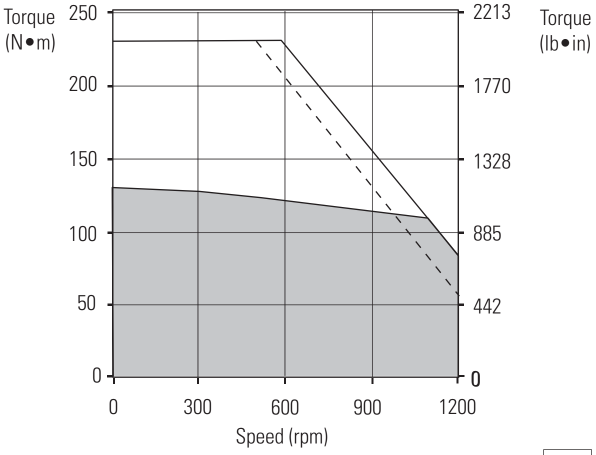
Kinetix 6000 and Kinetix 6200/6500 (460V) Drives/MP-Series Low Inertia Motor Curves, Continued

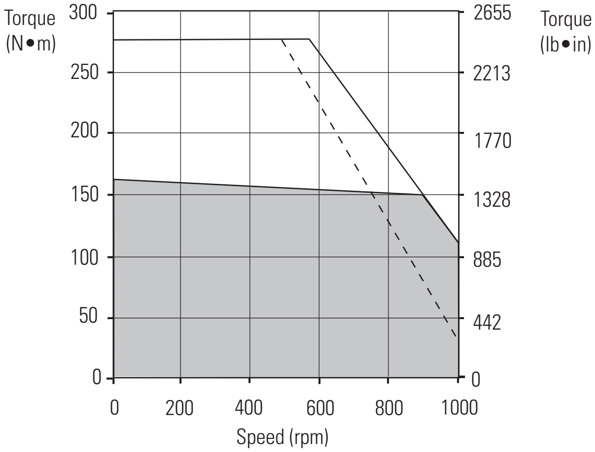

2094-BM05-S @ 200% or 2094-BM05-M and MPL-B680F


2094-BM05-S @ 200% or 2094-BM05-M and MPL-B860D


2094-BM05-S @ 200% or 2094-BM05-M and MPL-B880C


2094-BM05-S @ 200% or 2094-BM05-M and MPL-B880D


2094-BM05-S @ 200% or 2094-BM05-M and MPL-B960B


2094-BM05-S @ 200% or 2094-BM05-M and MPL-B980B

☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Kinetix 6000 (230V) Drives with MP-Series Medium Inertia Motors

This section provides system combination information for the Kinetix 6000 (230V) drives when matched with MP-Series medium-inertia motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin MPM Motor Cable Combinations

Motor Cat. No. (200V class)	Motor Power/Brake Cable	Motor Feedback Cable [1]
MPM-A1151M, MPM-A1152F, MPM-A1153F	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback 2090-CFBM7DF-CEAA*xx* (standard) [2] 2090-CFBM7DF-CEAF*xx* (continuous-flex) Resolver Feedback
MPM-A1302F	2090-CP*x*M7DF-14AA*xx* (standard) 2090-CP*x*M7DF-14AF*xx* (continuous-flex)	
MPM-A1304F	2090-CP*x*M7DF-12AA*xx* (standard)	
MPM-A1651F	2090-CP*x*M7DF-10AA*xx* (standard) 2090-CP*x*M7DF-10AF*xx* (continuous-flex)	
MPM-A1652F, MPM-A1653F	2090-CP*x*M7DF-08AA*xx* (standard) 2090-CP*x*M7DF-08AF*xx* (continuous-flex)	

(1) Use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

(2) These cables apply to Kinetix 6000 drives and MPM-A*xxxxx*-2 motors (resolver feedback).

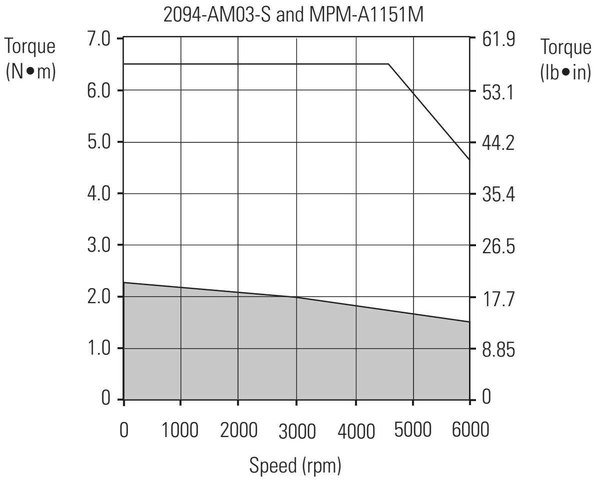
Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

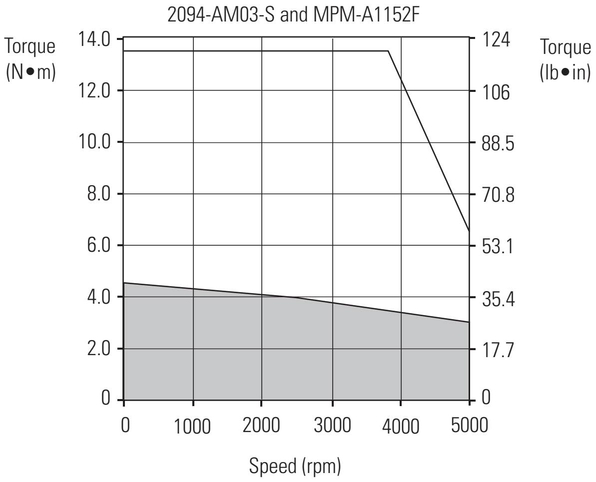
Bulletin MPM Motor Performance Specifications with Kinetix 6000 (230V) Drives

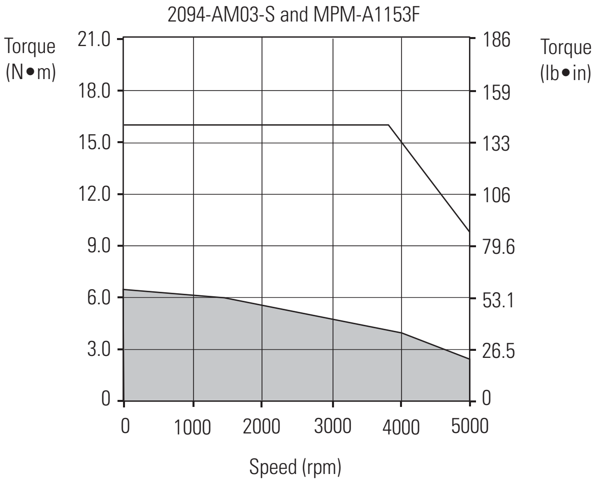
Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 6000 230V Drives
MPM-A1151M	6000	10.3	2.3 (20.3)	30.0	6.5 (57.5)	0.90	2094-AM02-S
				30.5	6.6 (58.4)		2094-AM03-S
MPM-A1152F	5000	14.9	4.7 (41.6)	30.0	9.9 (87.6)	1.40	2094-AM02-S
				44.8	13.5 (119)		2094-AM03-S
MPM-A1153F	5000	15.0	6.0 (53.1)	30.0	10.7 (94.7)	1.45	2094-AM02-S
		18.6	6.5 (57.5)	49.0	16.1 (142)		2094-AM03-S
MPM-A1302F	4500	19.8	6.6 (58.4)	49.0	13.2 (117)	1.65	2094-AM03-S
				50.2	13.5 (119)		2094-AM05-S
MPM-A1304F	4000	15.0	7.6 (67.2)	30.0	13.2 (117)	2.20	2094-AM02-S
		22.5	9.2 (81.4)	48.3	19.3 (171)		2094-AM03-S
MPM-A1651F	5000	24.5	9.3 (82.3)	49.0	15.2 (134)	2.50	2094-AM03-S
		35.6	10.7 (94.7)	73.4	20.3 (179)		2094-AM05-S
MPM-A1652F	4000	24.5	11.0 (97.3)	49.0	19.7 (174)	4.03	2094-AM03-S
		38.5	13.4 (119)	73.4	27.7 (245)		2094-AM05-S
MPM-A1653F	4000	24.5	11.7 (103)	49.0	21.1 (187)	5.10	2094-AM03-S
		48.7	18.6 (165)	73.4	29.6 (262)		2094-AM05-S

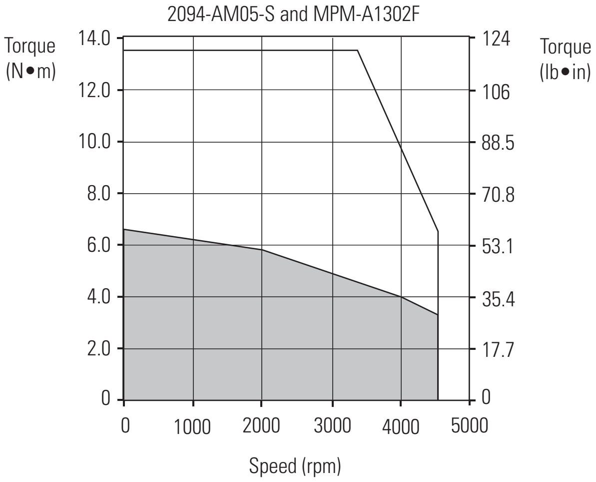
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

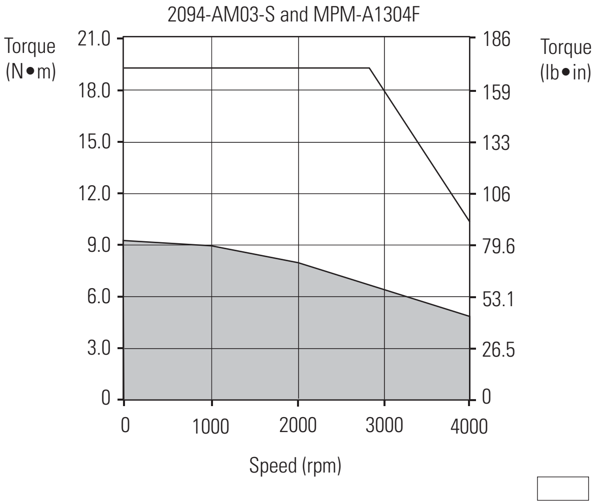
Kinetix 6000 (230V) Drives/MP-Series Medium Inertia Motor Curves

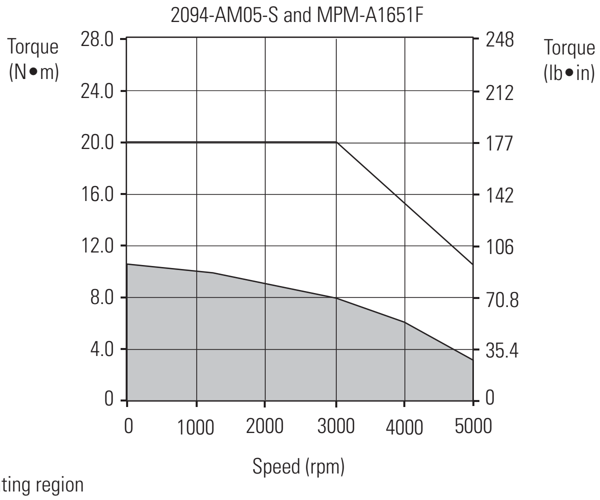

2094-AM03-S and MPM-A1151M


2094-AM03-S and MPM-A1152F


2094-AM03-S and MPM-A1153F

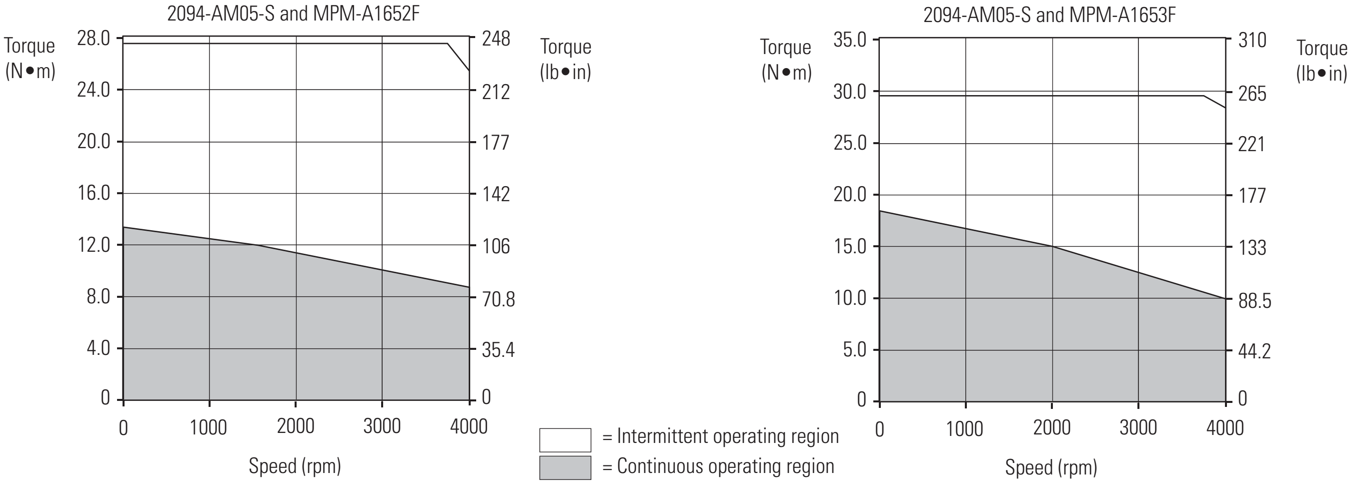

2094-AM05-S and MPM-A1302F


2094-AM03-S and MPM-A1304F


2094-AM05-S and MPM-A1651F

☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 6000 (230V) Drives/MP-Series Medium Inertia Motor Curves, Continued



Kinetix 6000 / Kinetix 6200/6500 (460V) Drives with MP-Series Medium Inertia Motors

This section provides system combination information for the Kinetix 6000 and Kinetix 6200/6500 (460V) drives when matched with MP-Series medium-inertia motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin MPM Motor Cable Combinations

Motor Cat. No. (400V class)	Motor Power/Brake Cable	Motor Feedback Cable [1]
MPM-B1151*x*, MPM-B1152*x*, MPM-B1153E, MPM-B1153F	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback 2090-CFBM7DF-CEAA*xx* (standard) [2] 2090-CFBM7DF-CEAF*xx* (continuous-flex) Resolver Feedback
MPM-B1302F, MPM-B1302M, MPM-B1304C, MPM-B1304E		
MPM-B1651C, MPM-B1652C		
MPM-B1153T	2090-CP*x*M7DF-14AA*xx* (standard) 2090-CP*x*M7DF-14AF*xx* (continuous-flex)	
MPM-B1302T, MPM-B1304M		
MPM-B1651F, MPM-B1653C		
MPM-B1651M, MPM-B1652E, MPM-B1652F, MPM-B1653E	2090-CP*x*M7DF-10AA*xx* (standard) 2090-CP*x*M7DF-10AF*xx* (continuous-flex)	
MPM-B2152C, MPM-B2153B		
MPM-B1653F	2090-CP*x*M7DF-08AA*xx* (standard) 2090-CP*x*M7DF-08AF*xx* (continuous-flex)	
MPM-B2152F, MPM-B2152M, MPM-B2153E, MPM-B2153F, MPM-B2154B, MPM-B2154E, MPM-B2154F		

(1) Use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

(2) These cables apply to Kinetix 6000 drives and MPM-B*xxxxx*-2 motors (resolver feedback).

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Bulletin MPM Motor Performance Specifications with Kinetix 6200/6500 (460V) Drives

Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Stall Torque** N•m (lb•in)	**Motor Rated Output** kW	**Kinetix 6200/ Kinetix 6500 460V Drives**
MPM-B1151F	5000	3.1	2.3 (20.3)	9.9	6.6 (58.4)	0.75	2094-BMP5-M
MPM-B1151T	7000	6.9	2.3 (20.3)	20.5	5.8 (51.3)	0.90	2094-BM01-M
MPM-B1152C	3000	4.1	5.0 (44.2)	12.4	13.5 (119)	1.20	2094-BM02-M
MPM-B1152F	5200	7.0	5.0 (44.2)	21.1	13.3 (118)	1.40	2094-BM01-M
MPM-B1152T	7000	12.6	5.0 (44.2)	36.5	13.1 (116)	1.40	2094-BM02-M
MPM-B1153E	3500	7.1	6.5 (57.5)	21.6	19.7 (174)	1.40	2094-BM01-M
MPM-B1153F	5500	10.5	6.4 (56.6)	32.0	19.7 (174)	1.40	2094-BM02-M
MPM-B1153T	7000	18.3	6.4 (56.6)	45.0	14.5 (128)	1.45	2094-BM03-M
MPM-B1302F	4500	8.6	6.6 (58.4)	21.5	13.0 (115)	1.65	2094-BM01-M
MPM-B1302M	6000	14.4	6.6 (58.4)	32.4	13.3 (118)	1.65	2094-BM02-M
MPM-B1302T	7000	19.3	6.7 (59.3)	43.4	13.3 (118)	1.65	2094-BM03-M
MPM-B1304C	2750	8.0	10.3 (91.1)	21.5	26.4 (233)	2.00	2094-BM01-M
MPM-B1304E	4000	12.3	10.2 (90.3)	34.2	27.1 (240)	2.20	2094-BM02-M
MPM-B1304M	6000	21.8	10.4 (92.0)	60.6	27.1 (240)	2.20	2094-BM03-M
MPM-B1651C	3500	11.7	11.4 (101)	29.2	23.2 (205)	2.50	2094-BM02-M
MPM-B1651F	5000	20.4	11.4 (101)	50.9	23.2 (205)	2.50	2094-BM03-M
MPM-B1651M	5000	25.8	11.3 (100)	56.8	21.4 (189)	2.50	2094-BM03-M
MPM-B1652C	2500	13.2	16.4 (145)	33.6	40.2 (356)	3.80	2094-BM02-M
MPM-B1652E	3500	24.0	21.1 (187)	60.5	48.0 (425)	4.30	2094-BM03-M
MPM-B1652F	4500	33.0	21.1 (187)	84.1	48.0 (424)	4.30	2094-BM05-M
MPM-B1653C	2500	23.0	26.7 (236)	59.2	67.7 (599)	4.60	2094-BM03-M
MPM-B1653E	3500	30.0	26.8 (237)	72.9	62.0 (549)	5.10	2094-BM03-M
MPM-B1653F	4000	40.1	31.0 (274)	94.3	56.0 (495)	5.10	2094-BM05-M
MPM-B2152C	2500	30.0	36.7 (325)	55.4	72.2 (639)	5.60	2094-BM03-M
MPM-B2152F	4500	48.9	34.1 (302)	97.8	72.3 (495)	5.90	2094-BM05-M
MPM-B2152M	5000	48.9	34.1 (302)	76.3	52.9 (468)	5.90	2094-BM05-M
MPM-B2153B	2000	27.6	48.0 (425)	60.0	101 (894)	6.80	2094-BM03-M
MPM-B2153E	3000	45.5	47.9 (424)	97.8	101 (894)	7.20	2094-BM05-M
MPM-B2153F	3800	48.9	45.6 (403)	97.8	99.0 (875)	7.20	2094-BM05-M
MPM-B2154B	2000	40.7	62.7 (555)	97.8	154 (1362)	6.90	2094-BM05-M
MPM-B2154E	3000	48.9	55.9 (495)	97.8	112 (990)	7.50	2094-BM05-M
MPM-B2154F	3300	48.9	56.2 (497)	83.6	88.0 (778)	7.50	2094-BM05-M

Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 5.000 or later.

Bulletin MPM Motor Performance Specifications with Kinetix 6000 (460V) Drives

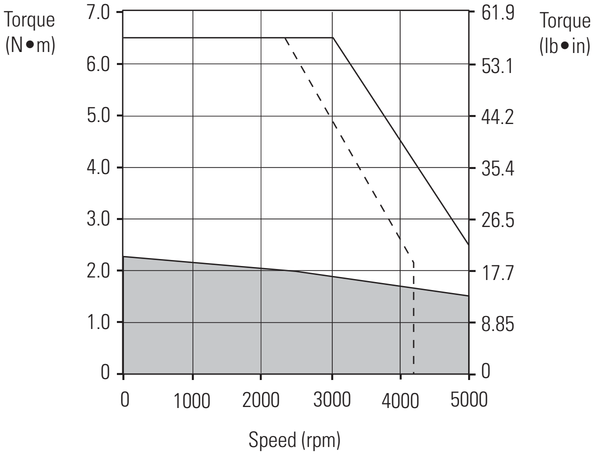
Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Stall Torque** N•m (lb•in)	**Motor Rated Output** kW	**Kinetix 6000 460V Drives**
MPM-B1151F	5000	3.1	2.3 (20.3)	5.9	4.3 (38.0)	0.75	2094-BMP5-S @ 150%
				9.9	6.6 (58.4)		2094-BMP5-S @ 250%
MPM-B1151T	7000	6.9	2.3 (20.3)	13.0	4.1 (36.3)	0.90	2094-BM01-S @ 150%
				20.5	5.8 (51.3)		2094-BM01-S @ 250%
MPM-B1152C	3000	4.1	5.0 (44.2)	5.9	7.2 (63.7)	1.20	2094-BMP5-S @ 150%
				10.0	11.3 (100)		2094-BMP5-S @ 250%
				12.4	13.5 (119)		2094-BM01-S @ 150%
MPM-B1152F	5200	7.0	5.0 (44.2)	13.0	9.0 (79.6)	1.40	2094-BM01-S @ 150%
				21.1	13.3 (118)		2094-BM01-S @ 250%
MPM-B1152T	7000	12.6	5.0 (44.2)	21.8	8.5 (75.2)	1.40	2094-BM02-S @ 150%
				36.5	13.1 (116)		2094-BM02-S @ 250%
MPM-B1153E	3500	7.1	6.5 (57.5)	21.5	13.0 (115)	1.40	2094-BM01-S @ 150%
				21.6	19.7 (174)		2094-BM01-S @ 250%
MPM-B1153F	5500	10.5	6.4 (56.6)	21.8	14.4 (127)	1.40	2094-BM02-S @ 150%
				32.0	19.7 (174)		2094-BM02-S @ 250%
MPM-B1153T	7000	18.3	6.4 (56.6)	45.0	14.5 (128)	1.45	2094-BM03-S @ 150%
MPM-B1302F	4500	8.6	6.6 (58.4)	13.0	8.9 (78.8)	1.65	2094-BM01-S @ 150%
				21.5	13.0 (115)		2094-BM01-S @ 250%
MPM-B1302M	6000	14.4	6.6 (58.4)	21.8	9.9 (87.6)	1.65	2094-BM02-S @ 150%
				32.4	13.3 (118)		2094-BM02-S @ 250%
MPM-B1302T	7000	14.6	6.0 (53.1)	36.5	11.8 (104)	1.65	2094-BM02-S @ 250%
		19.3	6.7 (59.3)	43.4	13.3 (118)		2094-BM03-S @ 150%
MPM-B1304C	2750	8.0	10.3 (91.1)	13.0	17.6 (156)	2.00	2094-BM01-S @ 150%
				21.5	26.4 (233)		2094-BM01-S @ 250%
MPM-B1304E	4000	12.3	10.2 (90.3)	21.8	19.0 (168)	2.20	2094-BM02-S @ 150%
				34.2	27.1 (240)		2094-BM02-S @ 250%
MPM-B1304M	6000	21.8	10.4 (92.0)	45.0	21.5 (190)	2.20	2094-BM03-S @ 150%
				60.6	27.1 (240)		2094-BM03-S @ 250%
MPM-B1651C	3500	11.7	11.4 (101)	21.8	19.4 (172)	2.50	2094-BM02-S @ 150%
				29.2	23.2 (205)		2094-BM02-S @ 250%
MPM-B1651F	5000	20.4	11.4 (101)	45.0	21.6 (191)	2.50	2094-BM03-S @ 150%
				50.9	23.2 (205)		2094-BM03-S @ 250%
MPM-B1651M	5000	25.8	11.3 (100)	45.0	18.8 (166)	2.50	2094-BM03-S @ 150%
				56.8	21.4 (189)		2094-BM03-S @ 250%
MPM-B1652C	2500	13.2	16.4 (145)	21.8	28.7 (254)	3.80	2094-BM02-S @ 150%
				33.6	40.2 (356)		2094-BM02-S @ 250%
MPM-B1652E	3500	24.0	21.1 (187)	45.0	38.4 (340)	4.30	2094-BM03-S @ 150%
				60.5	48.0 (425)		2094-BM03-S @ 250%

Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Stall Torque** N•m (lb•in)	**Motor Rated Output** kW	**Kinetix 6000 460V Drives**
MPM-B1652F	4500	33.0	21.1 (187)	73.4	41.1 (364)	4.30	2094-BM05-S @ 150%
				84.1	48.0 (424)		2094-BM05-S @ 200%
MPM-B1653C	2500	23.0	26.7 (236)	45.0	55.0 (487)	4.60	2094-BM03-S @ 150%
				59.2	67.7 (599)		2094-BM03-S @ 250%
MPM-B1653E	3500	30.0	26.8 (237)	45.0	42.5 (376)	5.10	2094-BM03-S @ 150%
				72.9	62.0 (549)		2094-BM03-S @ 250%
MPM-B1653F	4000	40.1	31.0 (274)	73.4	47.8 (423)	5.10	2094-BM05-S @ 150%
				94.3	56.0 (495)		2094-BM05-S @ 200%
MPM-B2152C	2500	30.0	36.7 (325)	45.0	60.3 (534)	5.60	2094-BM03-S @ 150%
				55.4	72.2 (639)		2094-BM03-S @ 250%
MPM-B2152F	4500	48.9	34.1 (302)	73.4	56.2 (497)	5.90	2094-BM05-S @ 150%
				97.8	72.3 (495)		2094-BM05-S @ 200%
MPM-B2152M	5000	48.9	34.1 (302)	73.4	51.0 (451)	5.90	2094-BM05-S @ 150%
				76.3	52.9 (468)		2094-BM05-S @ 200%
MPM-B2153B	2000	27.6	48.0 (425)	45.0	80.0 (708)	6.80	2094-BM03-S @ 150%
				60.0	101 (894)		2094-BM03-S @ 250%
MPM-B2153E	3000	45.5	47.9 (424)	73.4	79.4 (703)	7.20	2094-BM05-S @ 150%
				97.8	101 (894)		2094-BM05-S @ 200%
MPM-B2153F	3800	48.9	45.6 (403)	73.4	75.0 (664)	7.20	2094-BM05-S @ 150%
				97.8	99.0 (875)		2094-BM05-S @ 200%
MPM-B2154B	2000	40.7	62.7 (555)	73.4	121 (1071)	6.90	2094-BM05-S @ 150%
				97.8	154 (1362)		2094-BM05-S @ 200%
MPM-B2154E	3000	48.9	55.9 (495)	73.4	87.7 (776)	7.50	2094-BM05-S @ 150%
				97.8	112 (990)		2094-BM05-S @ 200%
MPM-B2154F	3300	48.9	56.2 (497)	73.4	78.8 (697)	7.50	2094-BM05-S @ 150%
				83.6	88.0 (778)		2094-BM05-S @ 200%

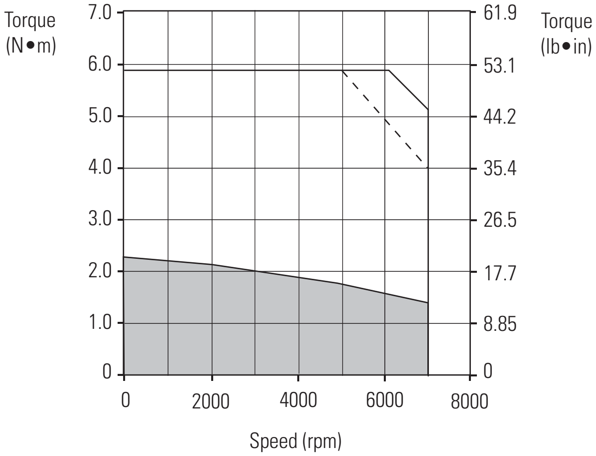
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 5.000 or later.

Kinetix 6000 / Kinetix 6200/6500 (460V) Drives/MP-Series Medium Inertia Motor Curves

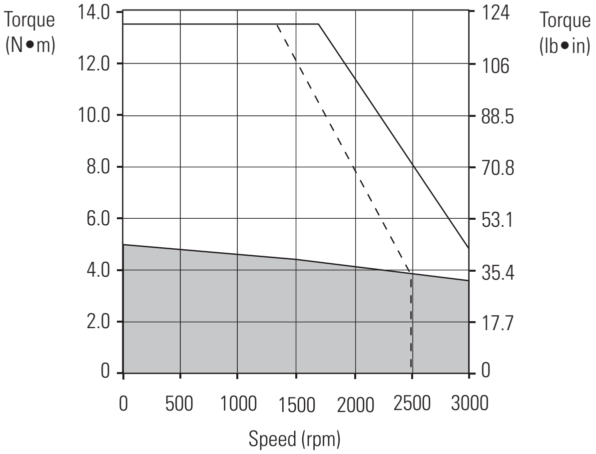
2094-BMP5-S @ 250% or 2094-BMP5-M
and MPM-B1151F



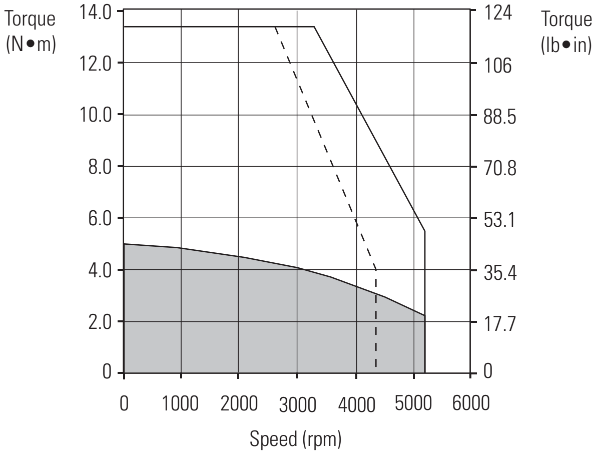
2094-BM01-S @ 250% or 2094-BM01-M
and MPM-B1151T



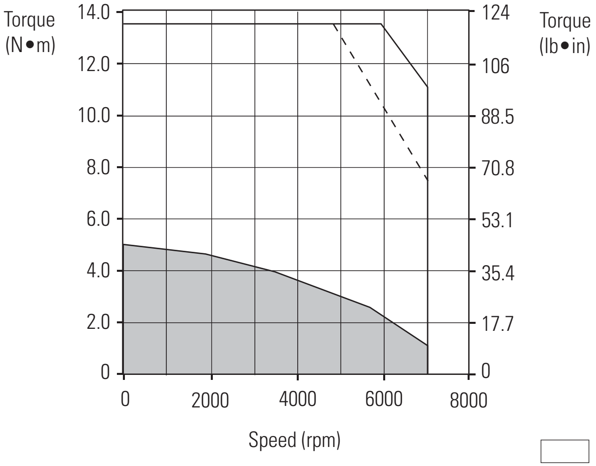
2094-BM01-S @ 150% or 2094-BM02-M
and MPM-B1152C



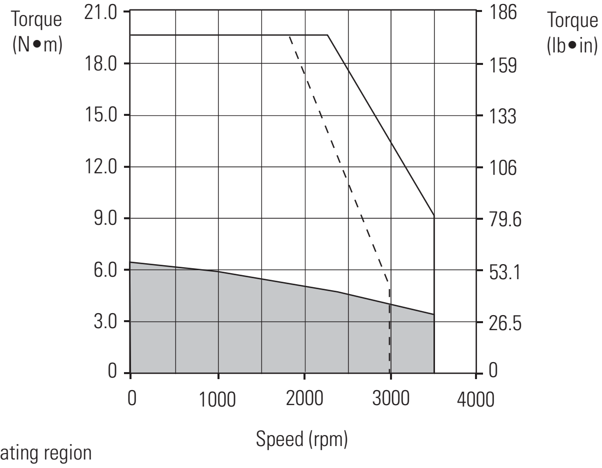
2094-BM01-S @ 250% or 2094-BM01-M
and MPM-B1152F



2094-BM02-S @ 250% or 2094-BM02-M
and MPM-B1152T



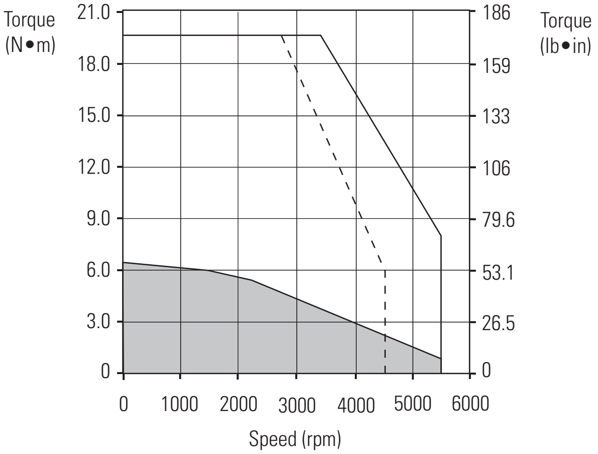
2094-BM01-S @ 250% or 2094-BM01-M
and MPM-B1153E



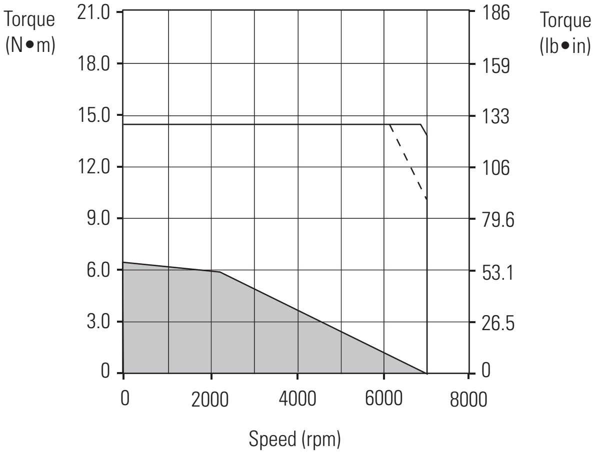
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Kinetix 6000 / Kinetix 6200/6500 (460V) Drives/MP-Series Medium Inertia Motor Curves, Continued

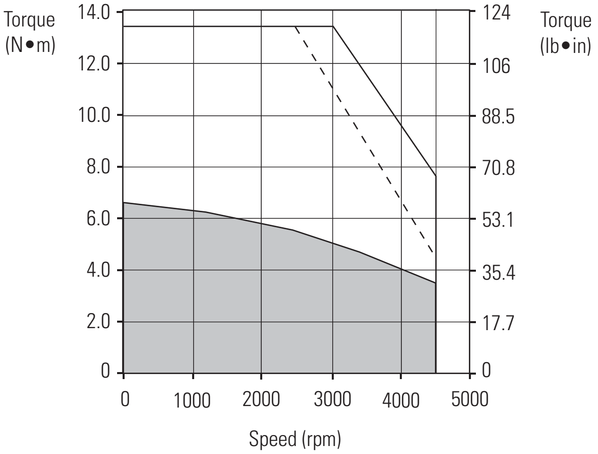
2094-BM02-S @ 250% or 2094-BM02-M
and MPM-B1153F



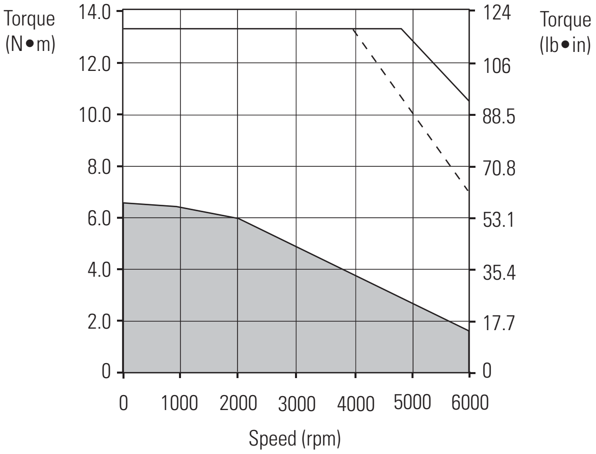
2094-BM03-S @ 150% or 2094-BM03-M
and MPM-B1153T



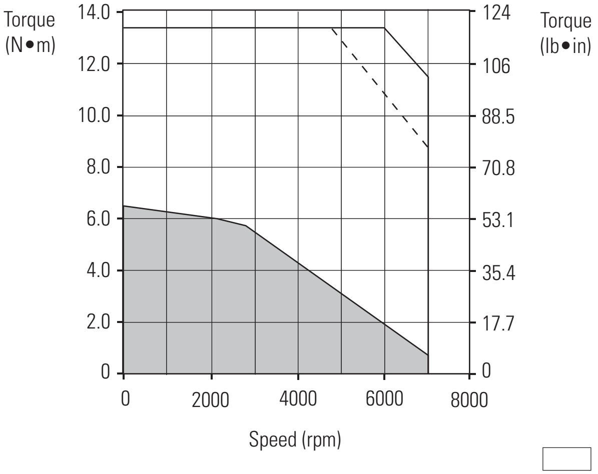
2094-BM01-S @ 250% or 2094-BM01-M
and MPM-B1302F



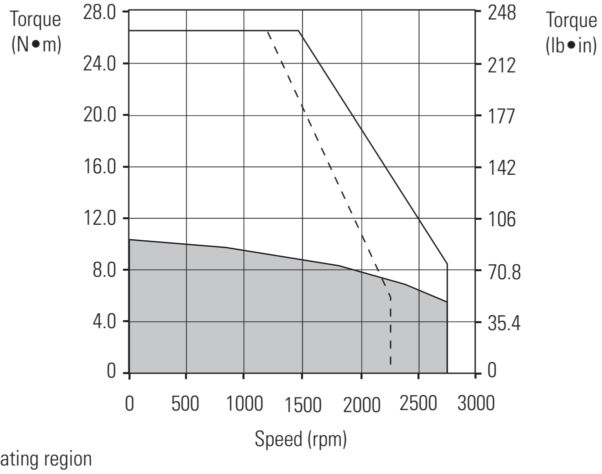
2094-BM02-S @ 250% or 2094-BM02-M
and MPM-B1302M



2094-BM03-S @ 150% or 2094-BM03-M
and MPM-B1302T

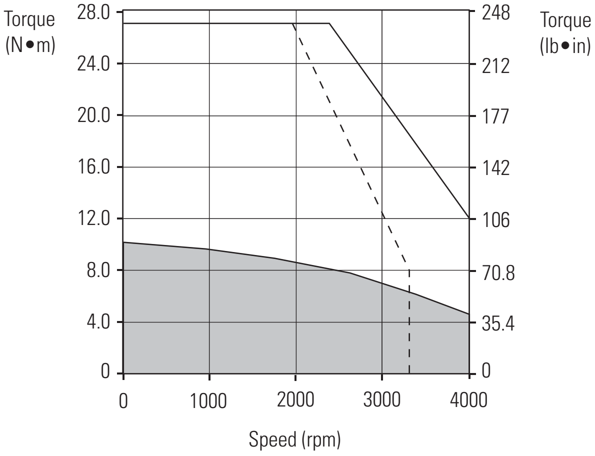


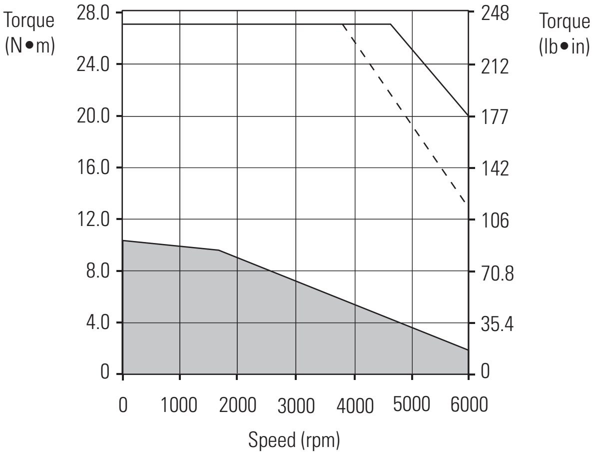
2094-BM01-S @ 250% or 2094-BM01-M
and MPM-B1304C

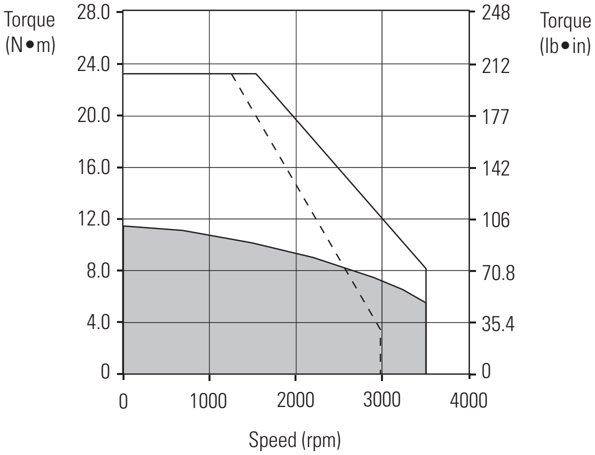


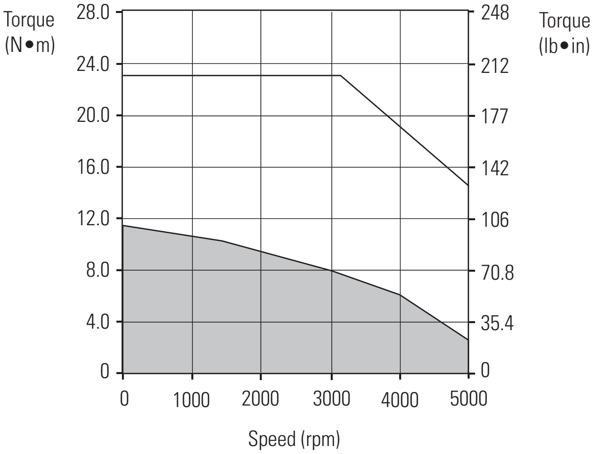
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

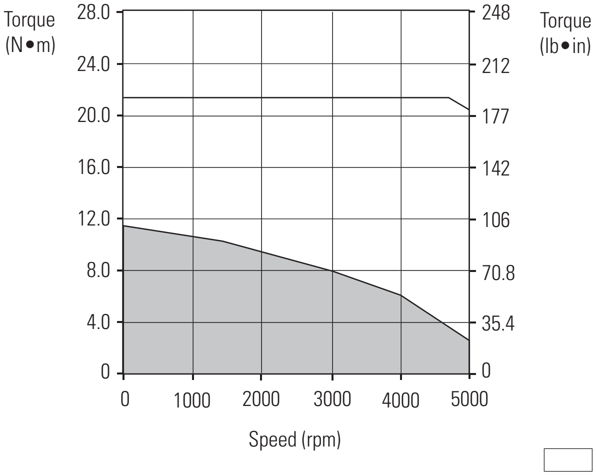
Kinetix 6000 / Kinetix 6200/6500 (460V) Drives/MP-Series Medium Inertia Motor Curves, Continued

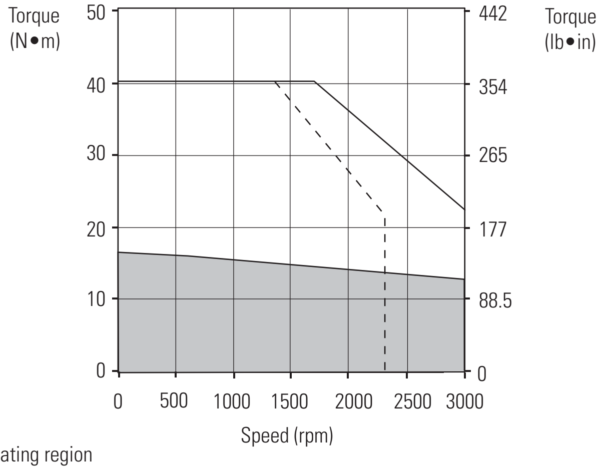

2094-BM02-S @ 250% or 2094-BM02-M and MPM-B1304E


2094-BM03-S @ 250% or 2094-BM03-M and MPM-B1304M


2094-BM02-S @ 250% or 2094-BM02-M and MPM-B1651C


2094-BM03-S @ 250% or 2094-BM03-M and MPM-B1651F

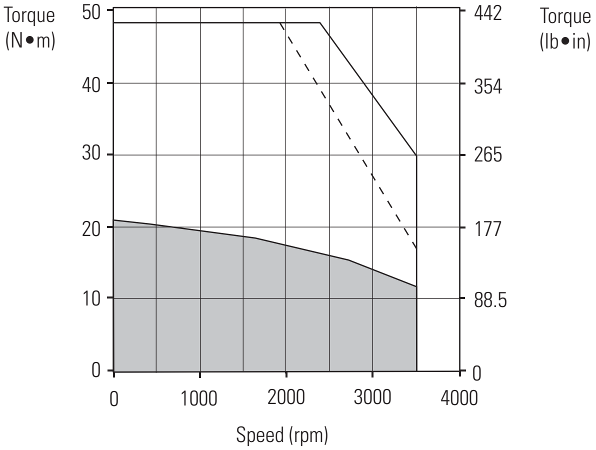

2094-BM03-S @ 250% or 2094-BM03-M and MPM-B1651M


2094-BM02-S @ 250% or 2094-BM02-M and MPM-B1652C

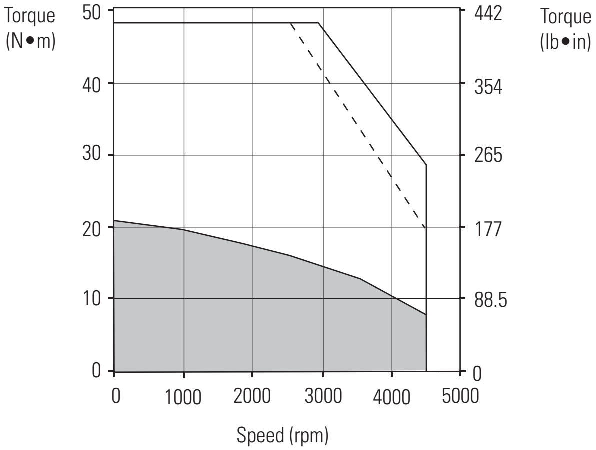
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Kinetix 6000 / Kinetix 6200/6500 (460V) Drives/MP-Series Medium Inertia Motor Curves, Continued

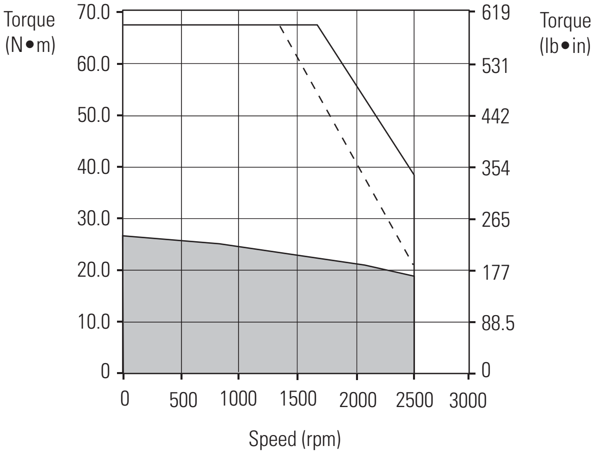
2094-BM03-S @ 250% or 2094-BM03-M
and MPM-B1652E



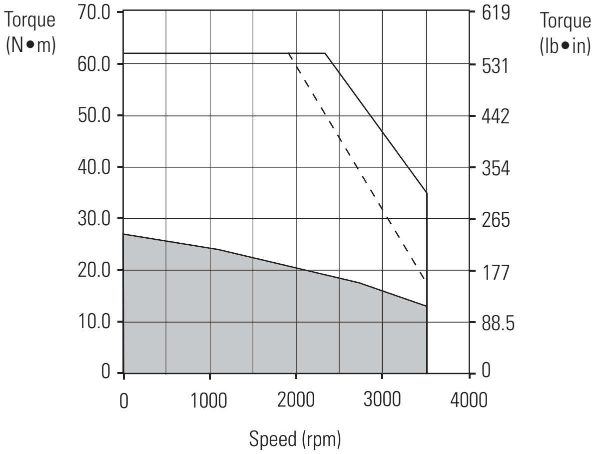
2094-BM05-S @ 200% or 2094-BM05-M
and MPM-B1652F



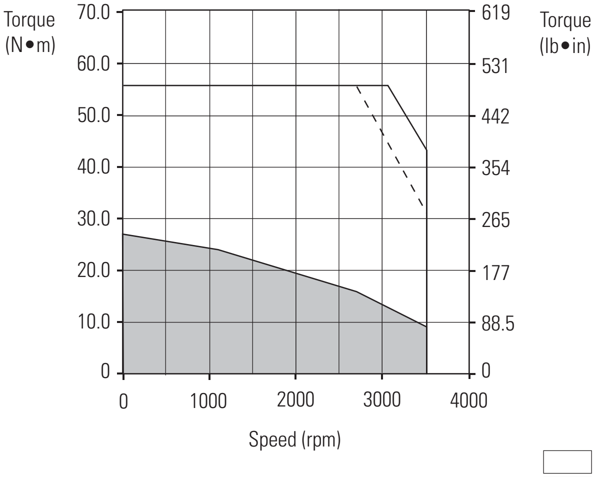
2094-BM03-S @ 250% or 2094-BM03-M
and MPM-B1653C



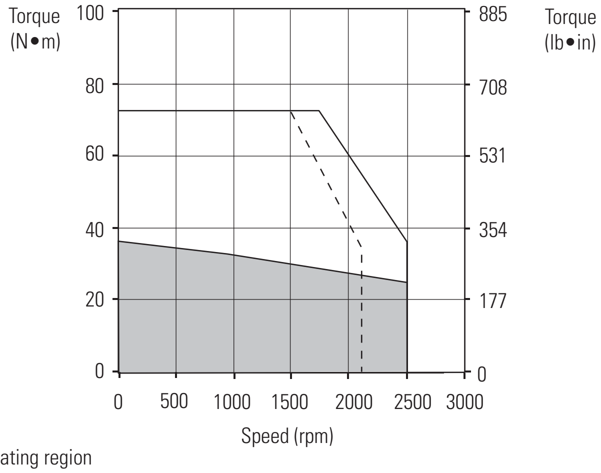
2094-BM03-S @ 250% or 2094-BM03-M
and MPM-B1653E



2094-BM05-S @ 200% or 2094-BM05-M
and MPM-B1653F



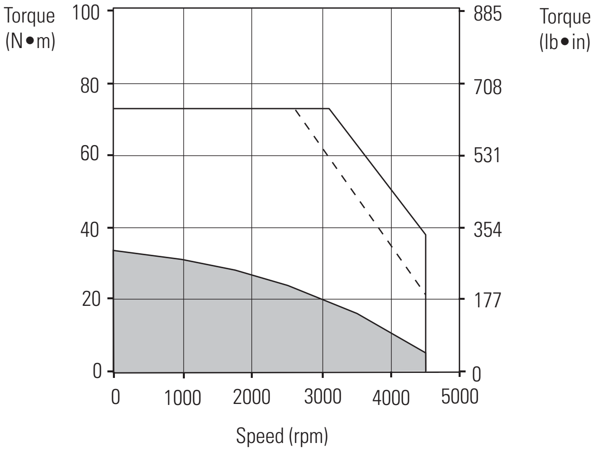
2094-BM03-S @ 250% or 2094-BM03-M
and MPM-B2152C



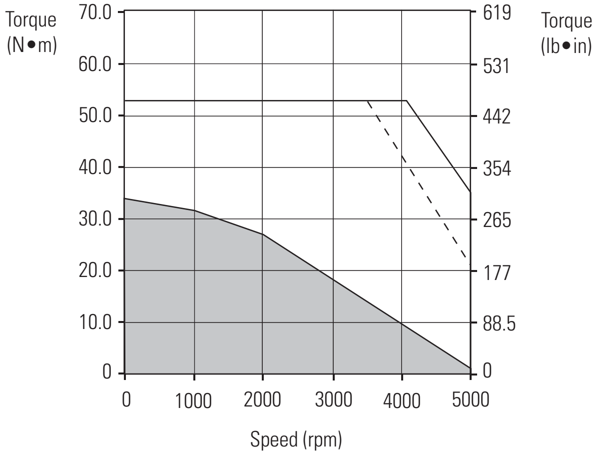
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Kinetix 6000 / Kinetix 6200/6500 (460V) Drives/MP-Series Medium Inertia Motor Curves, Continued

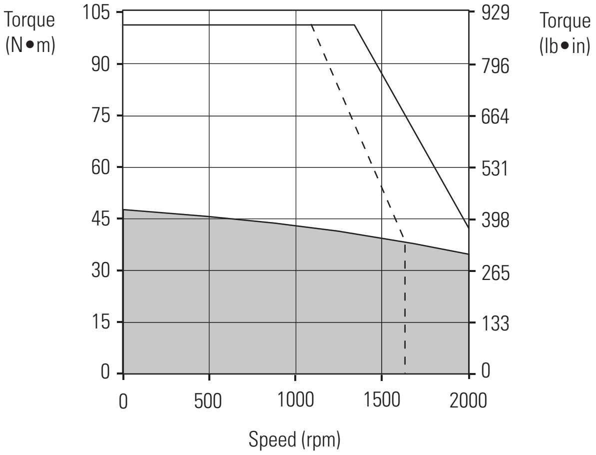
2094-BM05-S @ 200% or 2094-BM05-M
and MPM-B2152F



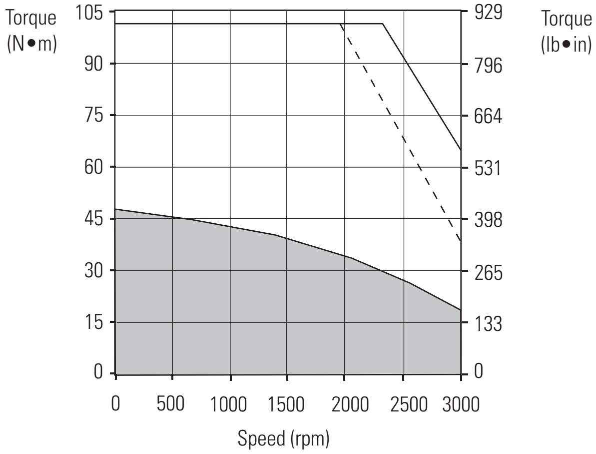
2094-BM05-S @ 200% or 2094-BM05-M
and MPM-B2152M



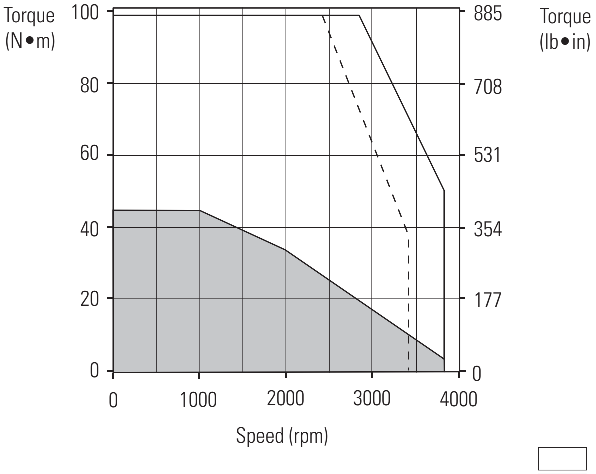
2094-BM03-S @ 250% or 2094-BM03-M
and MPM-B2153B



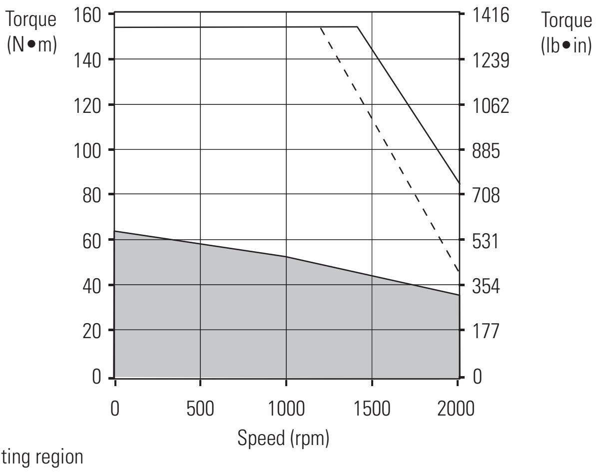
2094-BM05-S @ 200% or 2094-BM05-M
and MPM-B2153E



2094-BM05-S @ 200% or 2094-BM05-M
and MPM-B2153F

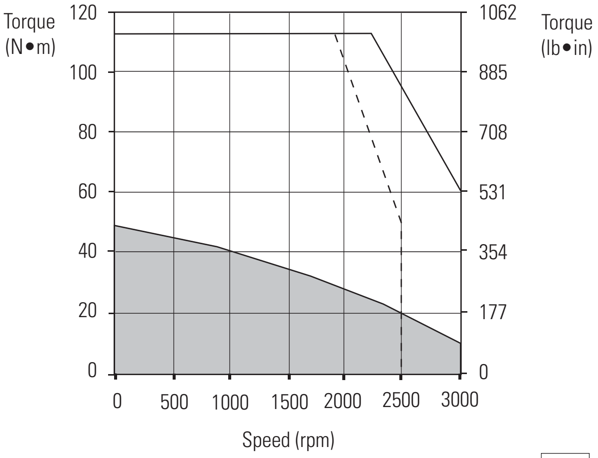


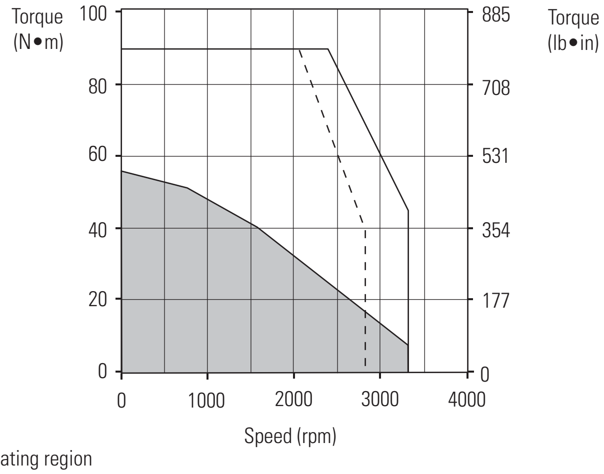
2094-BM05-S @ 200% or 2094-BM05-M
and MPM-B2154B



☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Kinetix 6000 / Kinetix 6200/6500 (460V) Drives/MP-Series Medium Inertia Motor Curves, Continued


2094-BM05-S @ 200% or 2094-BM05-M
and MPM-B2154E


2094-BM05-S @ 200% or 2094-BM05-M
and MPM-B2154F

☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Kinetix 6000 (230V) Drives with MP-Series Food Grade Motors

This section provides system combination information for the Kinetix 6000 (230V) drives when matched with MP-Series food-grade motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin MPF Motor Cable Combinations

Motor Cat. No. (200V class)	Motor Power/Brake Cable	Motor Feedback Cable [1]
MPF-A310P, MPF-A320H, MPF-A320P, MPF-A330P	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback
MPF-A430H		
MPF-A430P MPF-A4540F, MPF-A4530K	2090-CP*x*M7DF-14AA*xx* (standard) 2090-CP*x*M7DF-14AF*xx* (continuous-flex)	
MPF-A540K	2090-CP*x*M7DF-08AA*xx* (standard) 2090-CP*x*M7DF-08AF*xx* (continuous-flex)	

(1) Use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

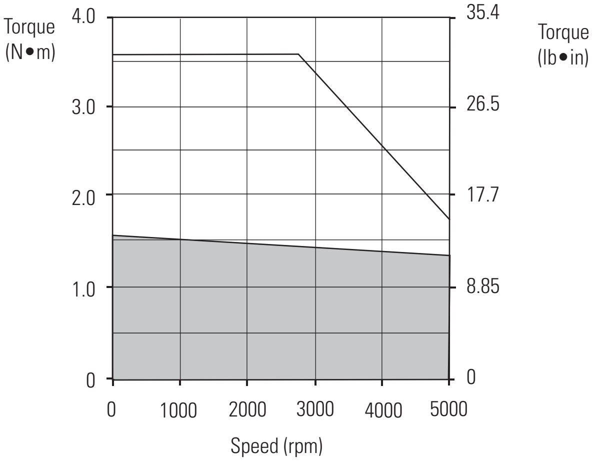
Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

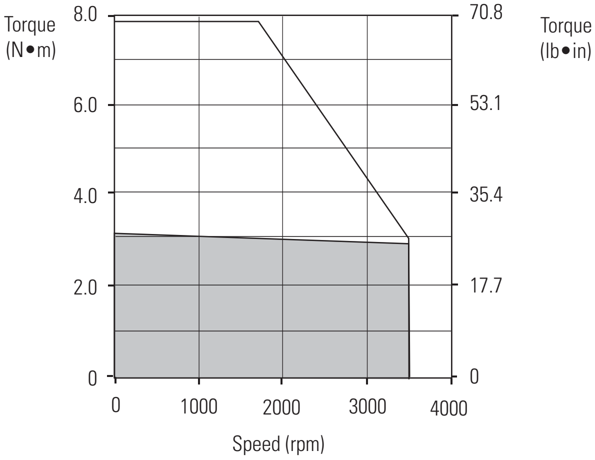
Bulletin MPF Motor Performance Specifications with Kinetix 6000 (230V) Drives

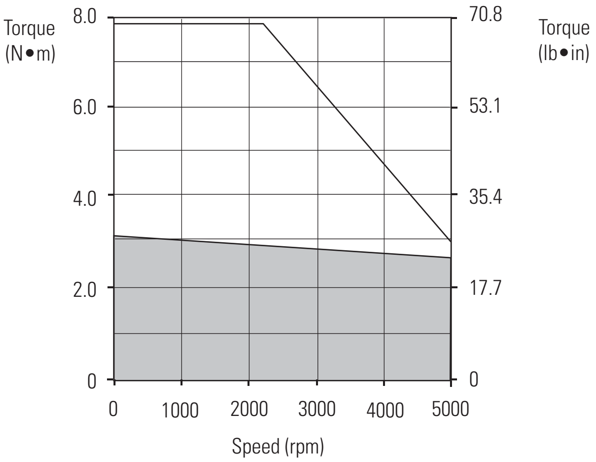
Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Stall Torque** N•m (lb•in)	**Motor Rated Output** kW	**Kinetix 6000 230V Drives**
MPF-A310P	5000	4.91	1.58 (14.0)	10.5	2.91 (25.7)	0.73	2094-AMP5-S
				14.0	3.61 (31.9)		2094-AM01-S
MPF-A320H	3500	6.10	3.05 (27.0)	17.0	6.97 (61.6)	1.0	2094-AM01-S
				19.3	7.91 (70.0)		2094-AM02-S
MPF-A320P	5000	8.50	2.88 (25.5)	17.0	5.07 (44.8)	1.3	2094-AM01-S
		9.00	3.05 (27.0)	29.5	7.91 (70.0)		2094-AM02-S
MPF-A330P	5000	12.0	3.85 (34.0)	30.0	8.47 (74.9)	1.6	2094-AM02-S
				38.0	10.32 (91.2)		2094-AM03-S
MPF-A430H	3500	12.2	6.21 (55.0)	30.0	13.20 (117)	1.8	2094-AM02-S
				45.0	19.82 (175)		2094-AM03-S
MPF-A430P	5000	16.80	5.94 (52.5)	49.0	15.36 (136)	1.9	2094-AM03-S
				67.0	19.80 (175)		2094-AM05-S
MPF-A4530K	4000	19.50	8.08 (71.4)	49.0	17.01 (150)	2.3	2094-AM03-S
				62.0	20.30 (179)		2094-AM05-S
MPF-A4540F	3000	18.40	10.15 (89.7)	49.0	23.56 (208)	2.5	2094-AM03-S
				58.0	27.10 (239)		2094-AM05-S
MPF-A540K	4000	24.5	11.40 (100)	49.0	21.68 (192)	4.1	2094-AM03-S
		41.5	19.42 (171)	73.4	31.55 (279)		2094-AM05-S

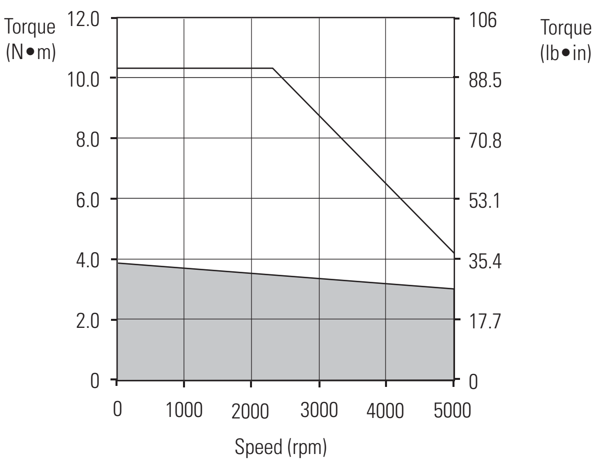
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

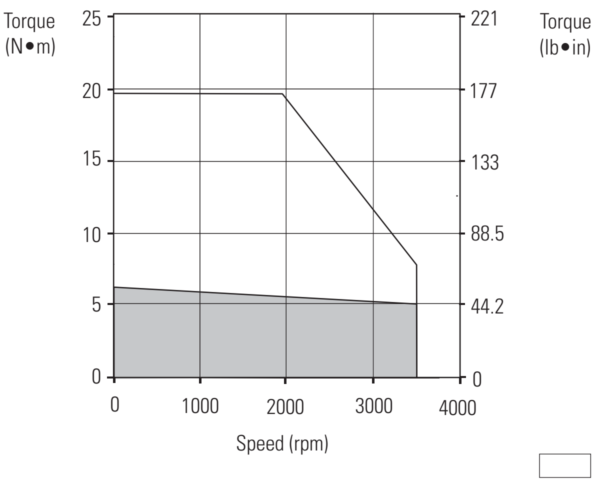
Kinetix 6000 (230V) Drives/MP-Series Food Grade Motor Curves

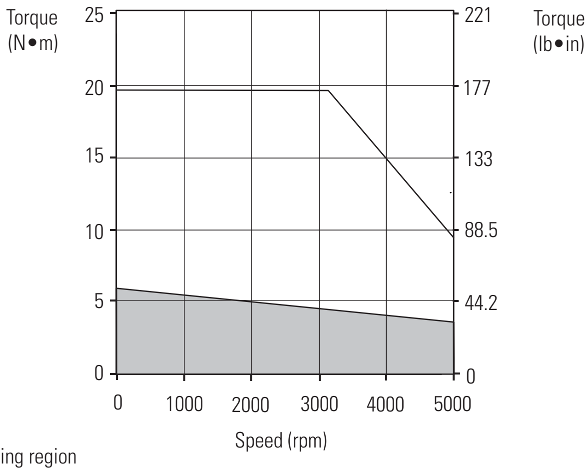

2094-AM01-S and MPF-A310P


2094-AM02-S and MPF-A320H


2094-AM02-S and MPF-A320P

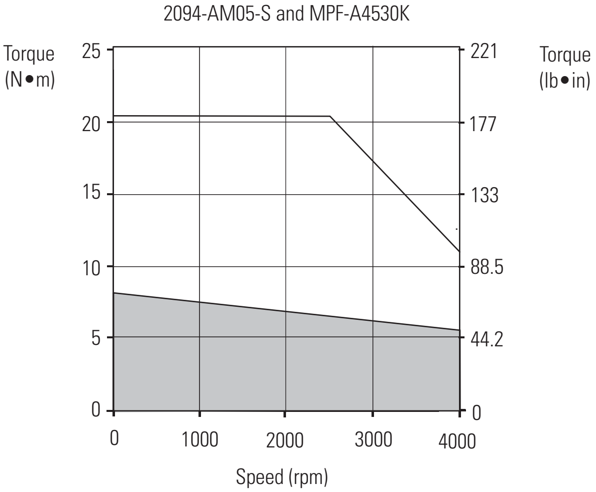

2094-AM03-S and MPF-A330P

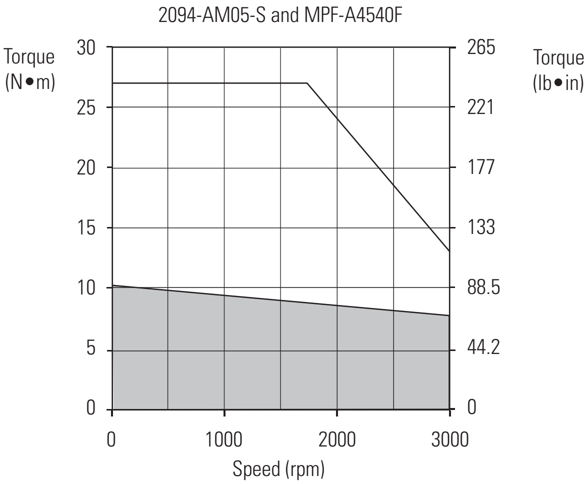

2094-AM03-S and MPF-A430H

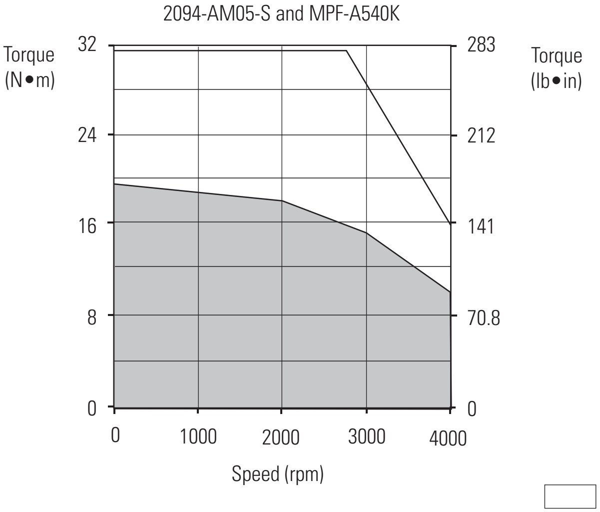

2094-AM05-S and MPF-A430P

☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 6000 (230V) Drives/MP-Series Food Grade Motor Curves, Continued







☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 6000 and Kinetix 6200/6500 (460V) Drives with MP-Series Food Grade Motors

This section provides system combination information for the Kinetix 6000 and the Kinetix 6200/6500 (460V) drives when matched with MP-Series food-grade motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

IMPORTANT When using the Kinetix 6000 (series B) drives, configured for enhanced peak performance, you can usually achieve full motor performance with a smaller drive. Kinetix 6200 and Kinetix 6500 drives are configured for enhanced peak performance by default. Expect the same peak performance from Kinetix 6200/6500 drives and Kinetix 6000 (series B) drives configured for enhanced peak performance.

Refer to Kinetix 6000 IAM/AM Module Series Change on page 276 for more information about using the peak enhancement feature.

Kinetix 6200 and Kinetix 6500 drives are configured for 250% peak performance by default. Expect the same peak performance from Kinetix 6200/6500 drives and Kinetix 6000 (series B) drives configured for 250% peak performance.

Bulletin MPF Motor Cable Combinations

Motor Cat. No. (400V class)	Motor Power/Brake Cable	Motor Feedback Cable [1]
MPF-B310P, MPF-B320P, MPF-B330P	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback
MPF-B430P		
MPF-B4530K, MPF-B4540F		
MPF-B540K	2090-CP*x*M7DF-10AA*xx* (standard) 2090-CP*x*M7DF-10AF*xx* (continuous-flex)	

(1) Use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Bulletin MPF Motor Performance Specifications with Kinetix 6200/6500 (460V) Drives

Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 6200/ Kinetix 6500 460V Drives
MPF-B310P	5000	2.30	1.60 (14)	7.10	3.6 (32)	0.77	2094-BMP5-M
MPF-B320P	5000	4.0	2.90 (25.6)	9.90	6.0 (53.1)	1.5	2094-BMP5-M
		4.24	3.10 (27)	14.0	7.8 (69)		2094-BM01-M
MPF-B330P	5000	4.0	2.90 (25.6)	9.90	6.5 (57.5)	1.6	2094-BMP5-M
		5.70	4.18 (37)	19.0	11.1 (98)		2094-BM01-M
MPF-B430P	5000	8.60	6.20 (54.9)	21.5	13.9 (123)	2.0	2094-BM01-M
		9.20	6.55 (58)	32.0	19.8 (175)		2094-BM02-M
MPF-B4530K	4000	8.60	7.10 (62.8)	21.5	15.1 (133)	2.4	2094-BM01-M
		9.90	8.25 (73)	31.0	20.3 (179)		2094-BM02-M
MPF-B4540F	3000	8.60	9.50 (84.1)	21.5	20.9 (185)	2.5	2094-BM01-M
		9.10	10.20 (90)	29.0	27.1 (240)		2094-BM02-M
MPF-B540K	4000	20.5	19.4 (171)	60.0	48.6 (430)	4.1	2094-BM03-M

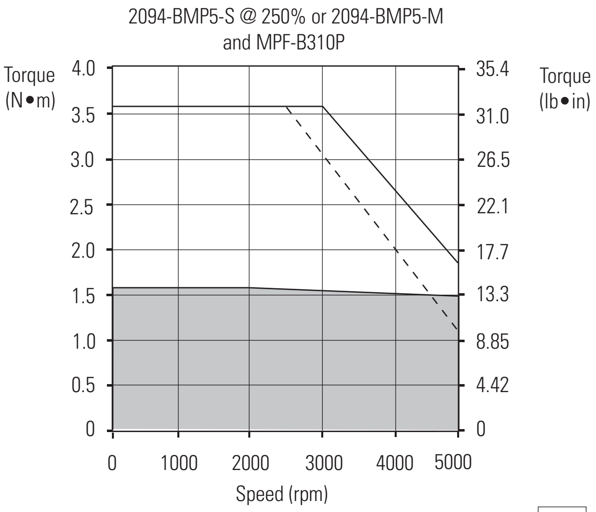
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 5.000 or later.

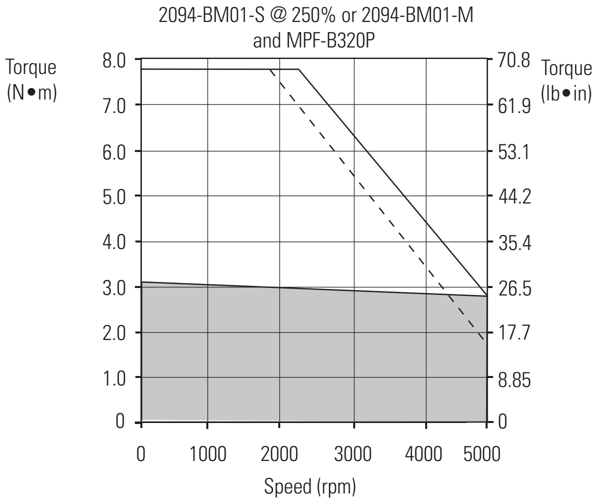
Bulletin MPF Motor Performance Specifications with Kinetix 6000 (460V) Drives

Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Stall Torque** N•m (lb•in)	**Motor Rated Output** kW	Kinetix 6000 460V Drives
MPF-B310P	5000	2.30	1.6 (14)	5.90	3.2 (28)	0.77	2094-BMP5-S @ 150%
				7.10	3.6 (32)		2094-BMP5-S @ 250%
MPF-B320P	5000	4.00	2.90 (26)	5.90	3.9 (34)	1.5	2094-BMP5-S @ 150%
		4.24	3.10 (27)	13.0	7.5 (66)		2094-BM01-S @ 150%
				14.0	7.8 (69)		2094-BM01-S @ 250%
MPF-B330P	5000	5.70	4.18 (37)	13.0	8.2 (72)	1.6	2094-BM01-S @ 150%
				19.0	11.1 (98)		2094-BM01-S @ 250%
MPF-B430P	5000	9.20	6.55 (58)	21.8	14.2 (125)	2.0	2094-BM02-S @ 150%
				32.0	19.8 (175)		2094-BM02-S @ 250%
MPF-B4530K	4000	9.90	8.25 (73)	21.8	15.4 (136)	2.4	2094-BM02-S @ 150%
				31.0	20.3 (179)		2094-BM02-S @ 250%
MPF-B4540F	3000	9.10	10.20 (90)	21.8	21.4 (189)	2.5	2094-BM02-S @ 150%
				29.0	27.1 (240)		2094-BM02-S@ 250%
MPF-B540K	4000	20.5	19.4 (171)	45.0	37.9 (335)	4.1	2094-BM03-S @ 150%
				60.0	48.6 (430)		2094-BM03-S @ 250%

Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 5.000 or later.

Kinetix 6000 and Kinetix 6200/6500 (460V) Drives/MP-Series Food Grade Motor Curves

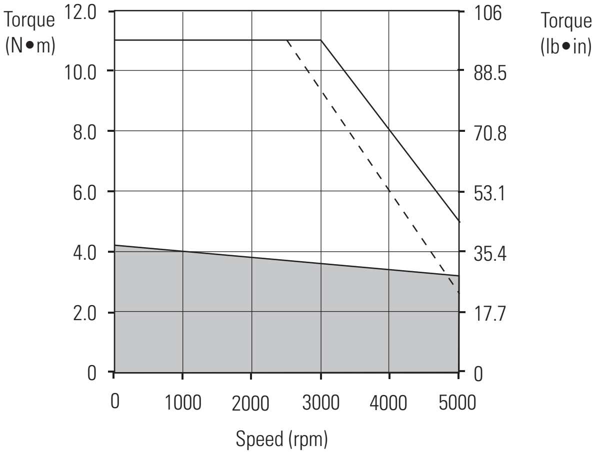




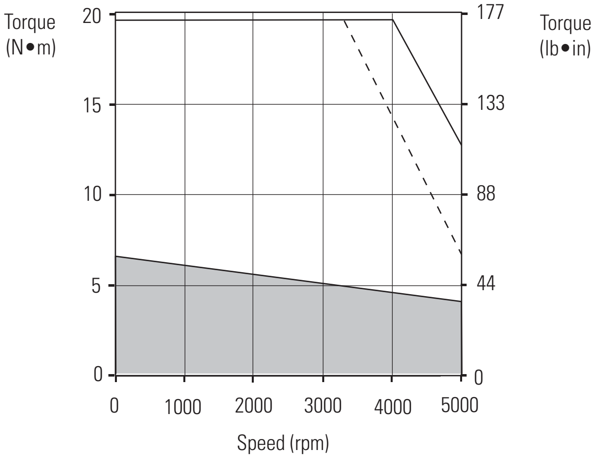
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Kinetix 6000 and Kinetix 6200/6500 (460V) Drives/MP-Series Food Grade Motor Curves, Continued

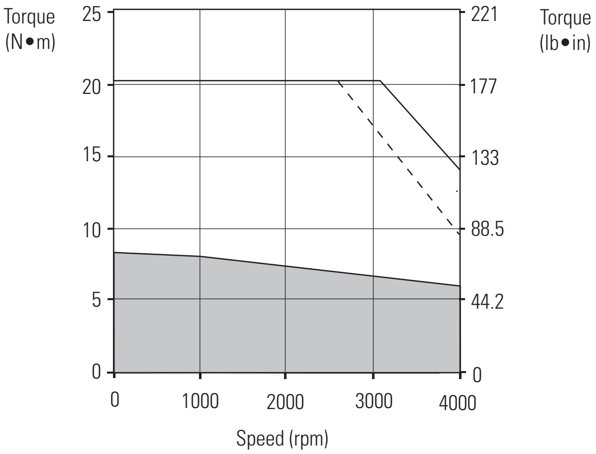
2094-BM01-S @ 250% or 2094-BM01-M
and MPF-B330P



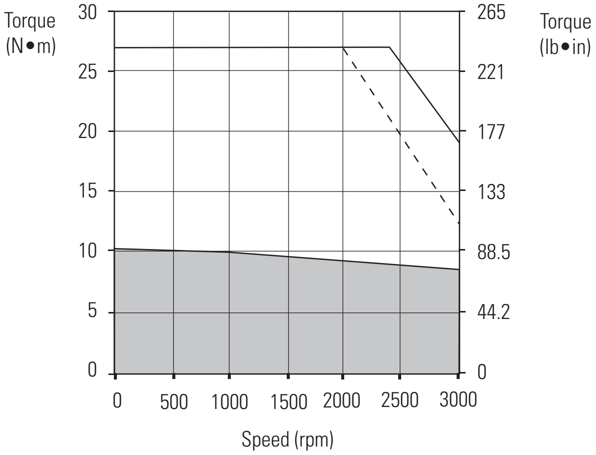
2094-BM02-S @ 250% or 2094-BM02-M
and MPF-B430P



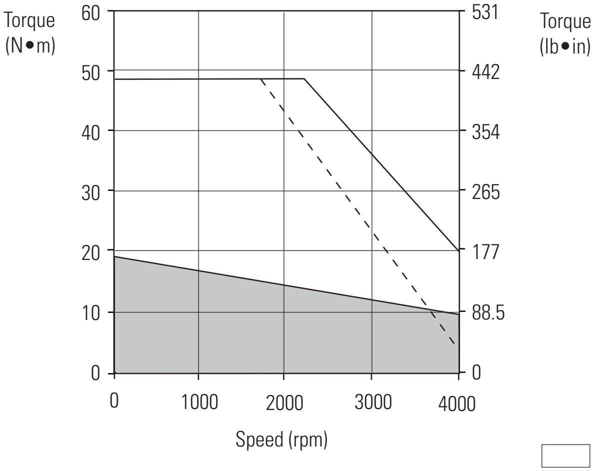
2094-BM02-S @ 250% or 2094-BM02-M
and MPF-B4530K



2094-BM02-S @ 250% or 2094-BM02-M
and MPF-B4540F



2094-BM03-S @ 250% or 2094-BM03-M
and MPF-B540K



☐ = Intermittent operating region
▨ = Continuous operating region
– – – = Drive operation with 400V AC (rms) input voltage

Kinetix 6000 (230V) Drives with MP-Series Stainless Steel Motors

This section provides system combination information for the Kinetix 6000 (230V) drives when matched with MP-Series stainless-steel motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin MPS Motor Cable Combinations

Motor Cat. No. (200V class)	Motor Power/Brake Cable	Motor Feedback Cable (1)
MPS-A330P	2090-XXNPMF-16S*xx* (2)	2090-XXNFMF-S*xx* (3) Absolute High-resolution Feedback
MPS-A4540F		

(1) Use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

(2) These cables are available as standard (catalog number 2090-XXNPMF-*xx*S*xx*) or continuous-flex (catalog number 2090-CP*x*M4DF-*xx*AF*xx*).

(3) These cables are available as standard (catalog number 2090-XXNFMF-S*xx*) or continuous-flex (catalog number 2090-CFBM4DF-CDAF*xx*).

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

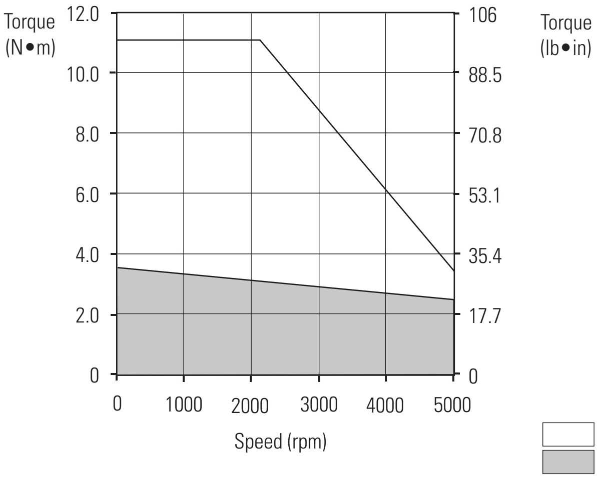
Bulletin MPS Motor Performance Specifications with Kinetix 6000 (230V) Drives

Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 6000 230V Drives
MPS-A330P	5000	8.50	3.10 (27)	17.0	5.80 (51)	1.3	2094-AM01-S
		9.80	3.60 (32.0)	30.0	9.30 (82)		2094-AM02-S
				38.0	11.10 (98)		2094-AM03-S
MPS-A4540F	3000	14.4	8.1 (72)	30.0	15.9 (140)	1.4	2094-AM02-S
				49.0	24.2 (214)		2094-AM03-S
				56.0	27.1 (240)		2094-AM05-S

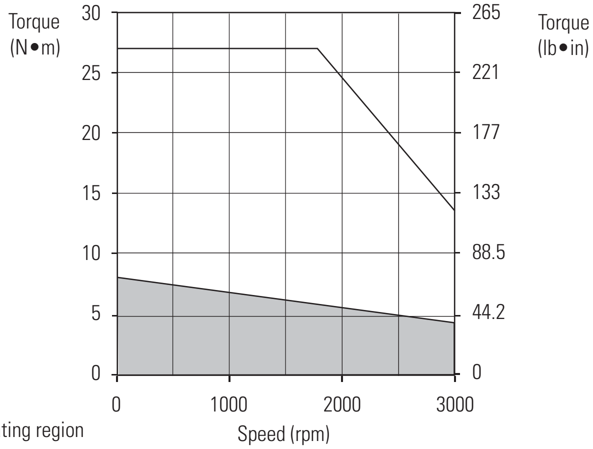
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 6000 (230V) Drives/MP-Series Stainless Steel Motor Curves

2094-AM03-S and MPS-A330P



2094-AM05-S and MPS-A4540F



☐ = Intermittent operating region

▨ = Continuous operating region

Kinetix 6000 / Kinetix 6200/6500 (460V) Drives with MP-Series Stainless Steel Motors

This section provides system combination information for the Kinetix 6000 and the Kinetix 6200/Kinetix 6500 (460V) drives when matched with MP-Series stainless-steel motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

IMPORTANT When using the Kinetix 6000 (series B) drives, configured for enhanced peak performance, you can usually achieve full motor performance with a smaller drive. Kinetix 6200 and Kinetix 6500 drives are configured for enhanced peak performance by default. Expect the same peak performance from Kinetix 6200/6500 drives and Kinetix 6000 (series B) drives configured for enhanced peak performance.

Refer to Kinetix 6000 IAM/AM Module Series Change on page 276 for more information about using the peak enhancement feature.

Kinetix 6200 and Kinetix 6500 drives are configured for 250% peak performance by default. Expect the same peak performance from Kinetix 6200/6500 drives and Kinetix 6000 (series B) drives configured for 250% peak performance.

Bulletin MPS Motor Cable Combinations

Motor Cat. No. (400V class)	Motor Power Cable	Motor Feedback Cable (1)
MPS-B330P	2090-XXNPMF-16S*xx* (2)	2090-XXNFMF-S*xx* (3) Absolute High-resolution Feedback
MPS-B4540F		
MPS-B560F	2090-XXNPMF-14S*xx* (2)	

(1) Use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

(2) These cables are available as standard (catalog number 2090-XXNPMF-*xx*S*xx*) or continuous-flex (catalog number 2090-CP*x*M4DF-*xx*AF*xx*).

(3) These cables are available as standard (catalog number 2090-XXNFMF-S*xx*) or continuous-flex (catalog number 2090-CFBM4DF-CDAF*xx*).

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Bulletin MPS Motor Performance Specifications with Kinetix 6200/6500 (460V) Drives

Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 6200/ Kinetix 6500 460V Drives
MPS-B330P	5000	4.0	3.0 (26.5)	9.90	6.6 (58.4)	1.3	2094-BMP5-M
		4.9	3.6 (32)	19.0	11.0 (97.2)		2094-BM01-M
MPS-B4540F	3000	7.1	8.1 (72)	21.5	22.8 (202)	1.4	2094-BM01-M
				26.0	27.1 (240)		2094-BM02-M
MPS-B560F	3000	17.0	21.5 (190)	68.0	67.8 (600)	3.5	2094-BM03-M

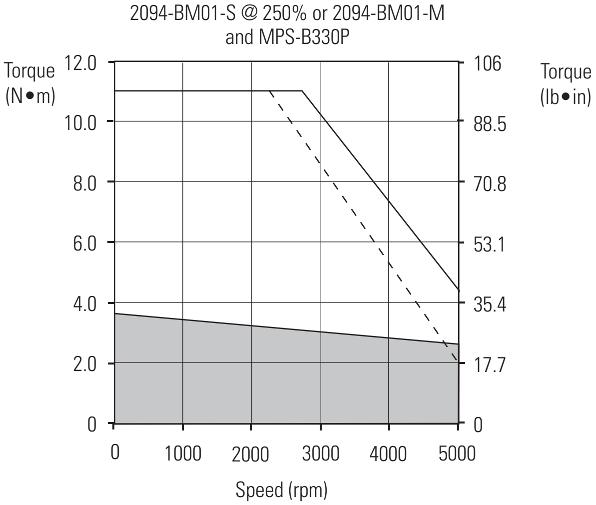
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 5.000 or later.

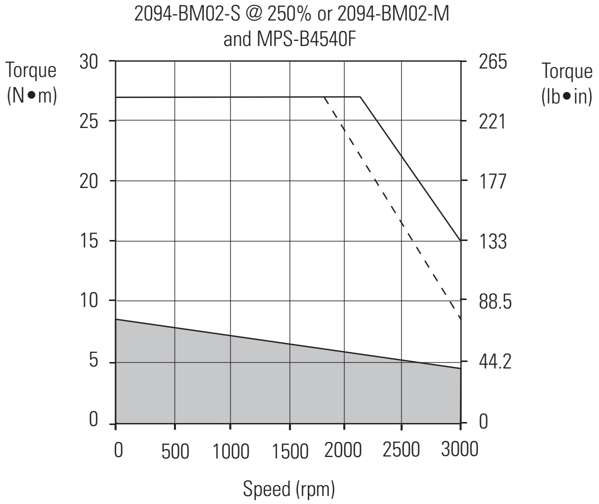
Bulletin MPS Motor Performance Specifications with Kinetix 6000 (460V) Drives

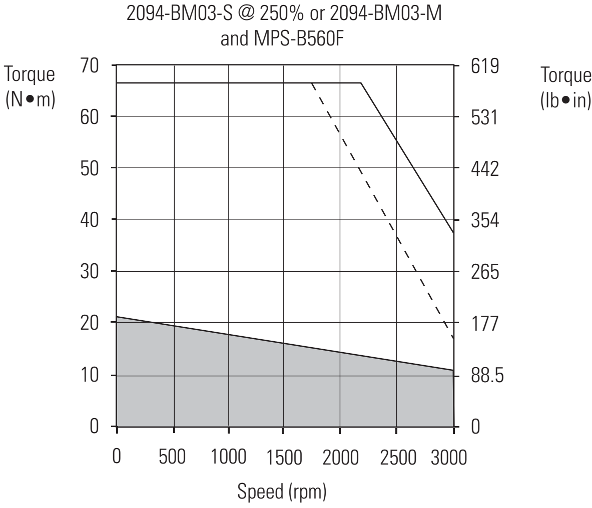
Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Stall Torque** N•m (lb•in)	**Motor Rated Output** kW	**Kinetix 6000 460V Drives**
MPS-B330P	5000	4.9	3.60 (32)	13.0	8.2 (72.5)	1.3	2094-BM01-S @ 150%
				19.0	11.0 (97.2)		2094-BM01-S @ 250%
MPS-B4540F	3000	7.1	8.1 (72)	21.5	22.8 (202)	1.4	2094-BM01-S @ 250%
				21.8	23.2 (205)		2094-BM02-S @ 150%
				26.0	27.1 (240)		2094-BM02-S @ 250%
MPS-B560F	3000	17.0	21.5 (190)	45.0	49.2 (435)	3.5	2094-BM03-S @ 150%
				68.0	67.8 (600)		2094-BM03-S @ 250%

Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 5.000 or later.

Kinetix 6000 and Kinetix 6200/6500 (460V) Drives/MP-Series Stainless Steel Motor Curves







☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Kinetix 6000 and Kinetix 6200/6500 (460V) Drives with RDD-Series Direct Drive Motors

This section provides system combination information for the Kinetix 6000 and Kinetix 6200/6500 (460V) drives when matched with RDD-Series direct drive motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin RDB Motor Cable Combinations

Motor Cat. No. (400V class)	Motor Power Cable	Motor Feedback Cable (1)
RDB-B21519, RDB-B21529	2090-CPWM7DF-16AA*xx* (standard) 2090-CPWM7DF-16AF*xx* (continuous-flex)	2090-XXNFMF-S*xx* (standard) 2090-CFBM7DF-CDAF*xx* (continuous-flex) Absolute High-resolution Feedback
RDB-B29014, RDB-B29016, RDB-B29024		
RDB-B2151C, RDB-B21539	2090-CPWM7DF-14AA*xx* (standard) 2090-CPWM7DF-14AF*xx* (continuous-flex)	
RDB-B29019, RDB-B29034		
RDB-B2152C	2090-CPWM7DF-12AA*xx* (standard)	
RDB-B29026		
RDB-B2153C	2090-CPWM7DF-10AA*xx* (standard) 2090-CPWM7DF-10AF*xx* (continuous-flex)	
RDB-B29036, RDB-B41014		
RDB-B29029, RDB-B41016, RDB-B41024	2090-CPWM7DF-08AA*xx* (standard) 2090-CPWM7DF-08AF*xx* (continuous-flex)	

(1) For Kinetix 6200/6500 drives, use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on the drive end.
For Kinetix 6000 drives, use low-profile feedback module (catalog number 2090-K6CK-KENDAT). Refer to Breakout Components and Connector Kits beginning on page 440.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers.
Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Bulletin RDB Motor Performance Specifications with Kinetix 6200/6500 (460V) Drives

Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Stall Torque** N•m (lb•in)	**Motor Rated Output** kW	**Kinetix 6200/ Kinetix 6500 460V Drives**
RDB-B21519	1235	10.3	31.2 (276)	27.3	83.1 (735)	3.64	2094-BM02-M
RDB-B2151C	2125	17.3	31.3 (277)	46.4	82.8 (733)	5.23	2094-BM03-M
RDB-B21529	1035	12.2	43.4 (384)	32.8	111 (982)	4.33	2094-BM02-M
RDB-B2152C	2125	23.5	43.4 (384)	63.2	111 (982)	6.41	2094-BM03-M
RDB-B21539	1250	15.8	51.5 (456)	47.9	137 (1212)	5.34	2094-BM03-M
RDB-B2153C	2250	29.4	51.5 (456)	82.6	137 (1212)	5.87	2094-BM03-M
RDB-B29014	450	5.9	48.9 (433)	17.6	110 (973)	1.97	2094-BM01-M
RDB-B29016	785	10.0	48.9 (433)	31.0	110 (973)	3.18	2094-BM02-M
RDB-B29019	1500	20.0	48.9 (167)	58.7	110 (973)	3.63	2094-BM03-M
RDB-B29024	435	11.0	97.8 (865)	33.0	214 (1894)	3.33	2094-BM02-M
RDB-B29026	885	22.3	97.8 (865)	67.2	214 (1894)	4.05	2094-BM03-M
RDB-B29029	1200	40.0	97.5 (863)	97.8	195 (1726)	4.05	2094-BM05-M
RDB-B29034	500	18.2	140 (1239)	56.6	321 (2841)	5.16	2094-BM03-M
RDB-B29036	750	27.0	140 (1239)	84.9	318 (2814)	5.49	2094-BM05-M
RDB-B29039	1000	48.9	113 (1000)	97.8	194 (1717)	4.41	2094-BM05-M
RDB-B41014	385	18.3	183 (1619)	51.2	340 (3009)	5.20	2094-BM03-M
RDB-B41016	700	33.8	183 (1619)	95.5	339 (3000)	4.83	2094-BM05-M
RDB-B41018	700	48.9	175 (1549)	97.8	271 (2398)	4.83	2094-BM05-M
RDB-B41024	365	31.5	330 (2929)	95.5	690 (6107)	7.29	2094-BM05-M

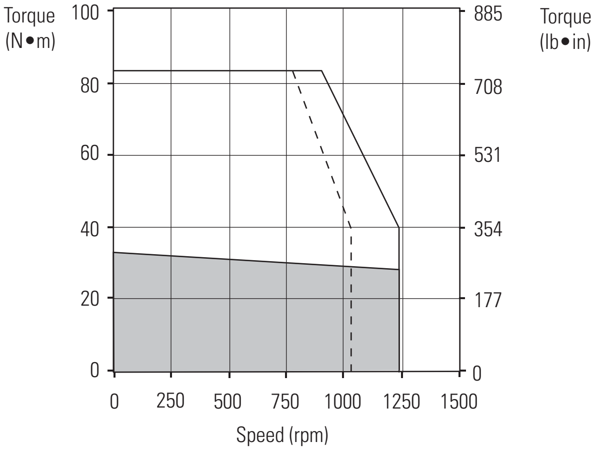
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 5.000 or later.

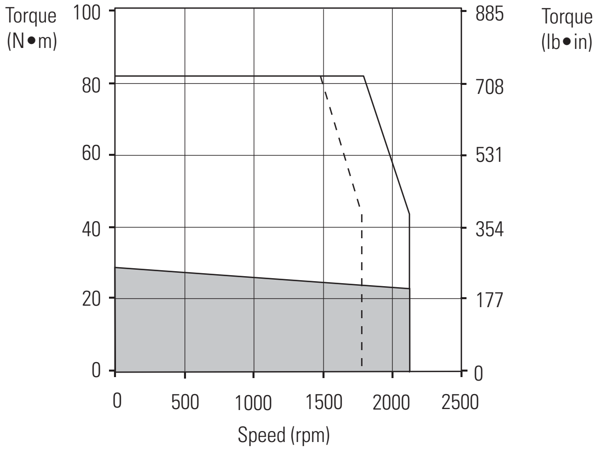
Bulletin RDB Motor Performance Specifications with Kinetix 6000 (460V) Drives

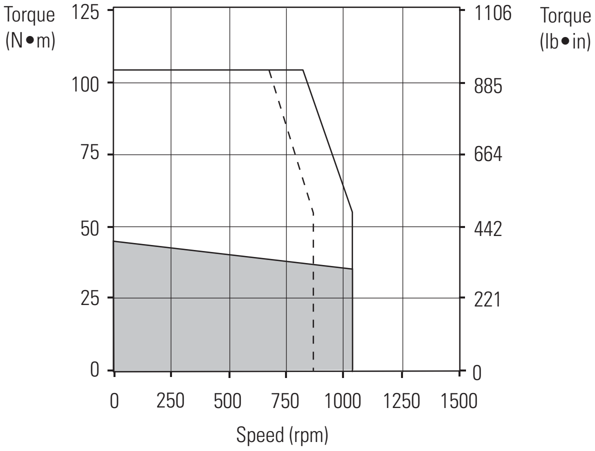
Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 6000 460V Drives
RDB-B21519	1235	10.3	31.2 (276)	21.8	66.8 (591)	3.64	2094-BM02-S @ 150%
				27.3	83.1 (735)		2094-BM02-S @ 250%
RDB-B2151C	2125	17.3	31.3 (277)	45.0	80.2 (710)	5.23	2094-BM03-S @ 150%
				46.4	82.8 (733)		2094-BM03-S @ 250%
RDB-B21529	1035	12.2	43.4 (384)	21.8	76.8 (680)	4.33	2094-BM02-S @ 150%
				32.8	111 (982)		2094-BM02-S @ 250%
RDB-B2152C	2125	23.5	43.4 (384)	45.0	80.4 (711)	6.41	2094-BM03-S @ 150%
				63.2	111 (982)		2094-BM03-S @ 250%
RDB-B21539	1250	15.8	51.5 (456)	45.0	130 (1150)	5.34	2094-BM03-S @ 150%
				47.9	137 (1212)		2094-BM03-S @ 250%
RDB-B2153C	2250	29.4	51.5 (456)	75.0	125 (1106)	5.87	2094-BM03-S @ 250%
				73.4	122 (1080)		2094-BM05-S @ 150%
				82.6	137 (1212)		2094-BM05-S @ 200%
RDB-B29014	450	5.9	48.9 (167)	13.0	89.2 (789)	1.97	2094-BM01-S @ 150%
				17.6	110 (973)		2094-BM01-S @ 250%
RDB-B29016	785	10.0	48.9 (167)	21.8	86.6 (766)	3.18	2094-BM02-S @ 150%
				31.0	110 (973)		2094-BM02-S @ 250%
RDB-B29019	1500	20.0	48.9 (167)	45.0	90.8 (803)	3.63	2094-BM03-S @ 150%
				58.7	110 (973)		2094-BM03-S @ 250%
RDB-B29024	435	11.0	97.8 (865)	21.8	159 (1407)	3.33	2094-BM02-S @ 150%
				33.0	214 (1894)		2094-BM02-S @ 250%
RDB-B29026	885	22.3	97.8 (865)	45.0	161 (1425)	4.05	2094-BM03-S @ 150%
				67.2	214 (1894)		2094-BM03-S @ 250%
RDB-B29029	1200	40.0	97.5 (863)	97.8	195 (1726)	4.05	2094-BM05-S @ 200%
RDB-B29034	500	18.2	140 (1239)	45.0	274 (2425)	5.16	2094-BM03-S @ 150%
				56.6	321 (2841)		2094-BM03-S @ 250%
RDB-B29036	750	27.0	140 (1239)	73.4	290 (2566)	5.49	2094-BM05-S @ 150%
				84.9	318 (2814)		2094-BM05-S @ 200%
RDB-B29039	1000	48.9	113 (1000)	97.8	194 (1717)	4.41	2094-BM05-S @ 200%
RDB-B41014	385	18.3	183 (1619)	45.0	317 (2805)	5.20	2094-BM03-S @ 150%
				51.2	340 (3009)		2094-BM03-S @ 250%
RDB-B41016	700	33.8	183 (1619)	73.4	292 (2584)	4.83	2094-BM05-S @ 150%
				95.5	339 (3000)		2094-BM05-S @ 200%
RDB-B41018	700	48.9	175 (1549)	97.8	271 (2398)	4.83	2094-BM05-S @ 200%
RDB-B41024	365	31.5	330 (2929)	73.4	593 (5248)	7.29	2094-BM05-S @ 150%
				95.5	690 (6107)		2094-BM05-S @ 200%

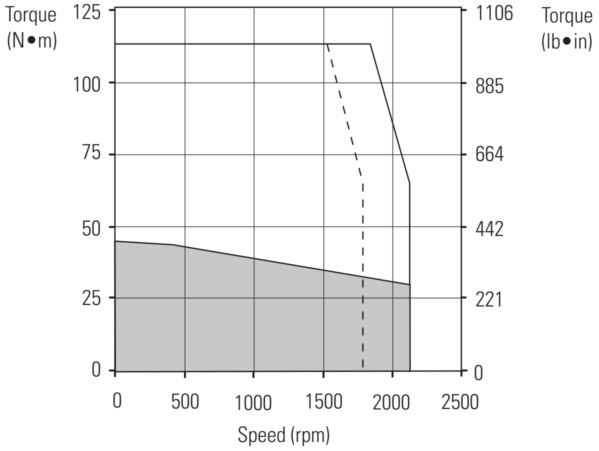
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 5.000 or later.

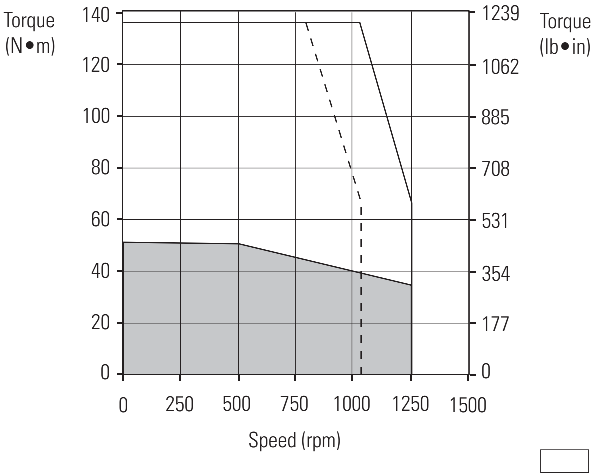
Kinetix 6000 (460V) Drives with RDD-Series Direct Drive Motor Curves

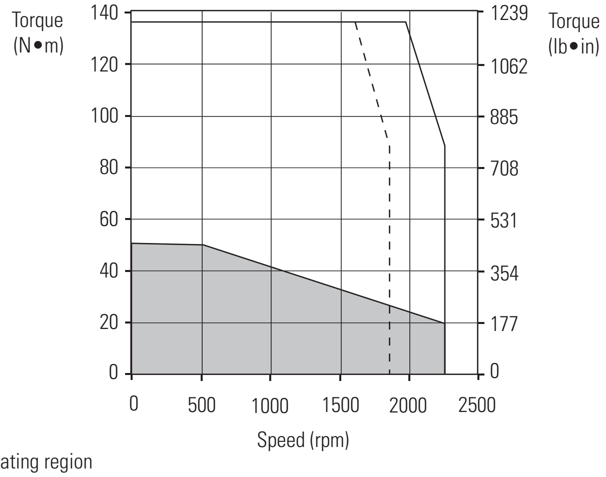

2094-BM02-S @ 250% or 2094-BM02-M
and RDB-B21519


2094-BM03-S @ 250% or 2094-BM03-M
and RDB-B2151C


2094-BM02-S @ 250% or 2094-BM02-M
and RDB-B21529


2094-BM03-S @ 250% or 2094-BM03-S
and RDB-B2152C

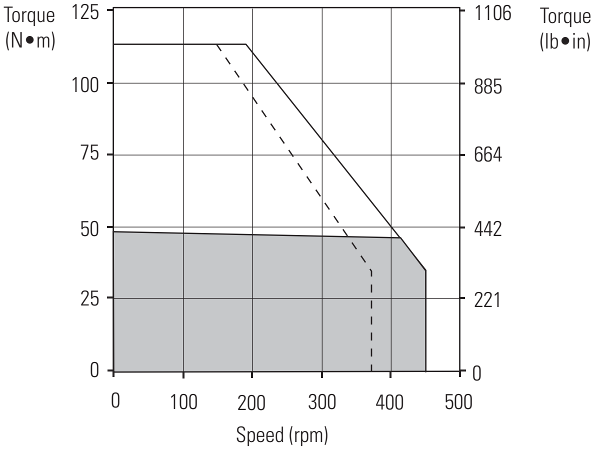

2094-BM03-S @ 250% or 2094-BM03-M
and RDB-21539


2094-BM05-S @ 200% or 2094-BM05-M
and RDB-2153C

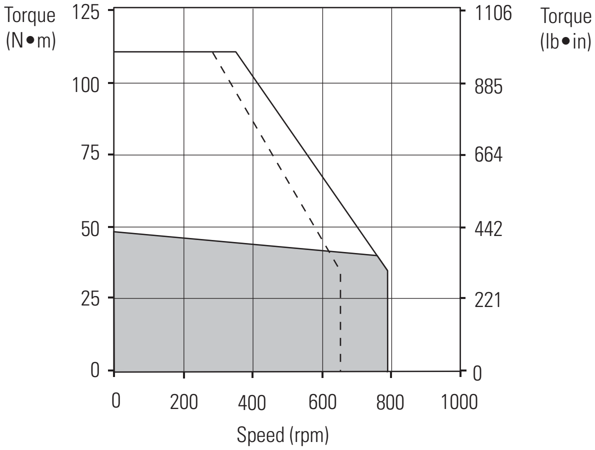
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Kinetix 6000 (460V) Drives with RDD-Series Direct Drive Motor Curves, Continued

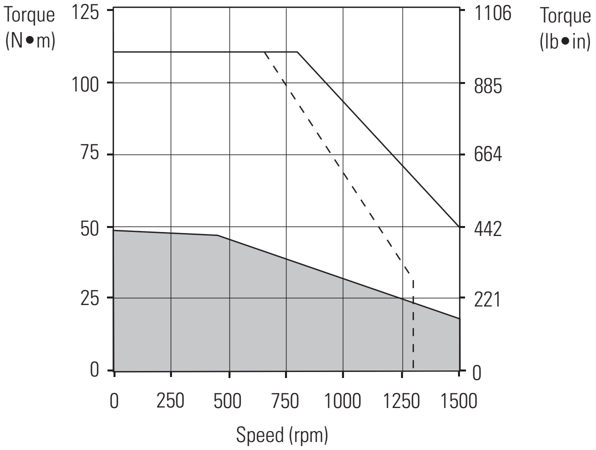
2094-BM01-S @ 250% or 2094-BM01-M
and RDB-B29014



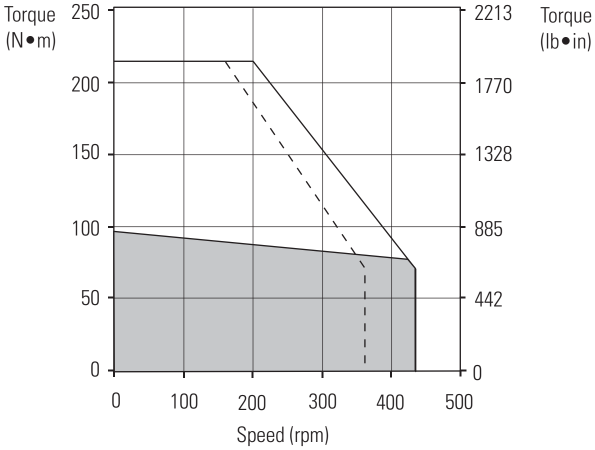
2094-BM02-S @ 250% or 2094-BM02-M
and RDB-B29016



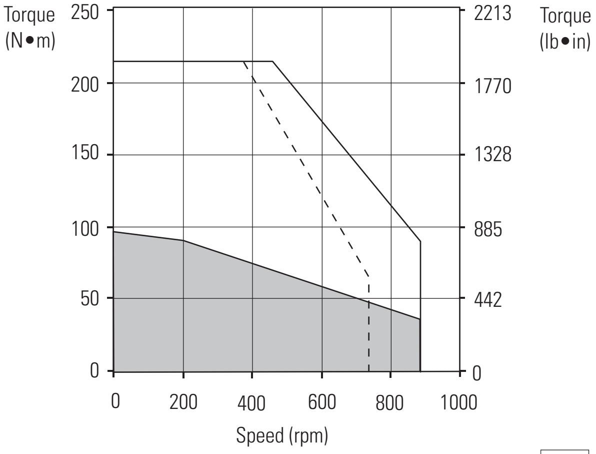
2094-BM03-S @ 250% or 2094-BM03-M
and RDB-B29019



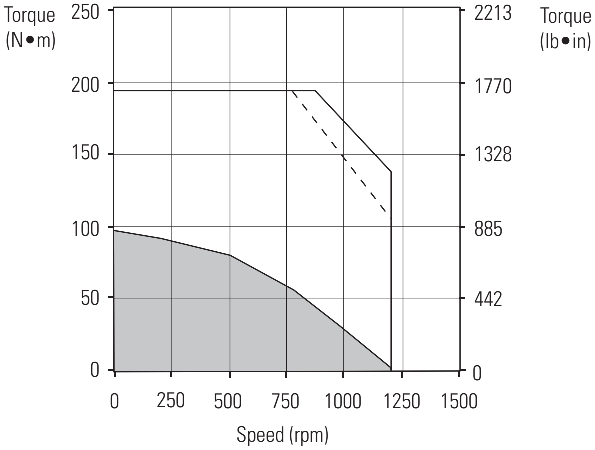
2094-BM02-S @ 250% or 2094-BM02-M
and RDB-B29024



2094-BM03-S @ 250% or 2094-BM03-M
and RDB-B29026



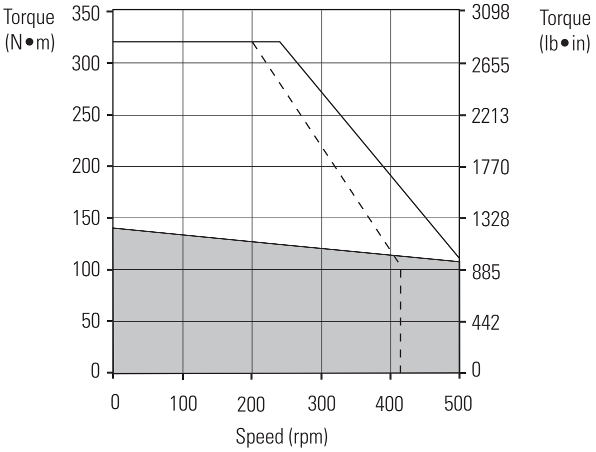
2094-BM05-S @ 200% or 2094-BM05-M
and RDB-B29029



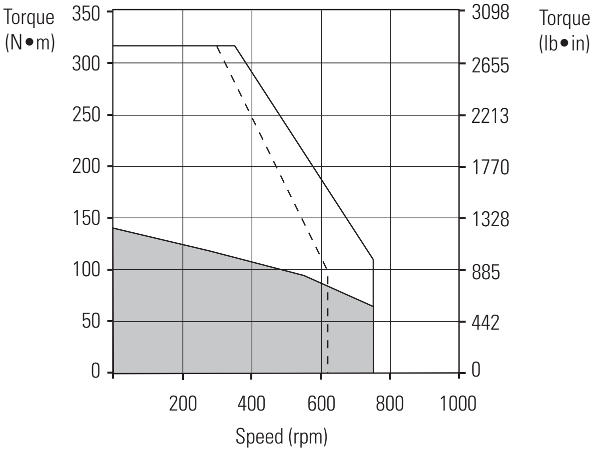
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Kinetix 6000 (460V) Drives with RDD-Series Direct Drive Motor Curves, Continued

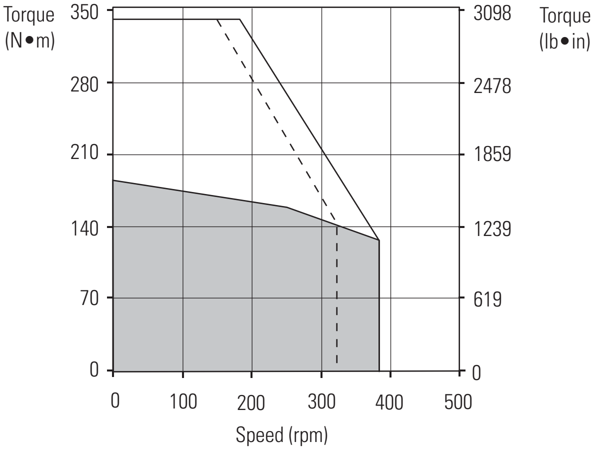
2094-BM03-S @ 250% or 2094-BM03-M
and RDB-B29034



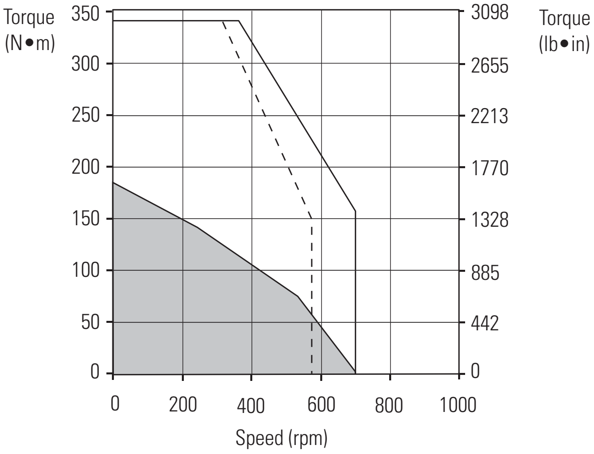
2094-BM05-S @ 200% or 2094-BM05-M
and RDB-B29036



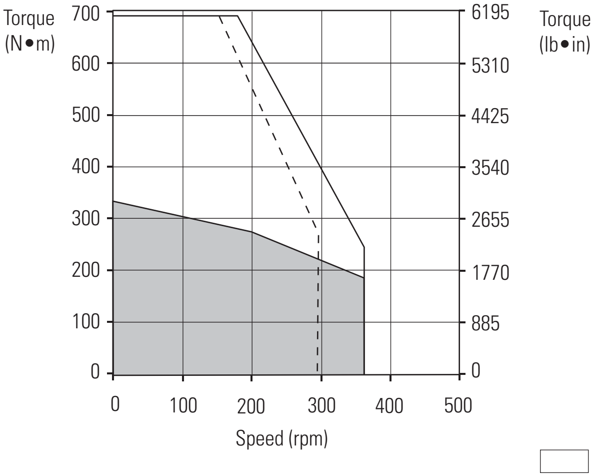
2094-BM03-S @ 250% or 2094-BM03-M
and RDB-B41014



2094-BM05-S @ 200% or 2094-BM05-M
and RDB-B41016



2094-BM05-S @ 200% or 2094-BM05-M
and RDB-B41024



☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Kinetix 6000 (230V) Drives with TL-Series Low Inertia Motors

This section provides system combination information for the Kinetix 6000 (230V) drives when matched with TL-Series (Bulletin TLY) low-inertia motors. Compatible TL-Series motors are equipped with incremental encoder feedback. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin TLY Motor Cable Combinations

Motor Cat. No. (200V class)	Motor Power/Brake Cable	Motor Feedback Cable (1)
TLY-A110T, TLY-A120T, TLY-A130T	2090-CPWM6DF-16AA*xx* (standard) without brake 2090-CPBM6DF-16AA*xx* (standard) with brake	2090-CFBM6DF-CBAA*xx* 2090-CFBM6DD-CCAA*xx* (standard) Incremental Feedback
TLY-A220T, TLY-A230T		
TLY-A2530P, TLY-A2540P		
TLY-A310M		

(1) Use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440 for more information.

TL-Series (Bulletin TLY-A*xxx*) motors are characterized as having 1000 mm (39.4 in.) cable extensions with circular plastic connectors.
Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Bulletin TLY (non-brake) Motor Performance Specifications with Kinetix 6000 (230V) Drives

Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 6000 230V Drives
TLY-A110T	6000	0.55	0.096 (0.85)	1.50	0.20 (1.75)	0.041	2094-AMP5-S
TLY-A120T		1.03	0.181 (1.60)	2.50	0.36 (3.20)	0.086	2094-AMP5-S
TLY-A130T		1.85	0.325 (2.88)	4.90	0.76 (6.70)	0.14	2094-AMP5-S
TLY-A220T		3.50	0.836 (7.40)	7.90	1.48 (13.1)	0.35	2094-AMP5-S
TLY-A230T		5.20	1.23 (10.9)	10.5	2.07 (18.3)	0.44	2094-AMP5-S
		5.50	1.30 (11.5)	15.5	3.05 (27.0)		2094-AM01-S
TLY-A2530P	5000	8.50	2.20 (19.5)	17.0	4.18 (37.0)	0.69	2094-AM01-S
		10.0	2.60 (23.0)	21.0	5.20 (46.0)		2094-AM02-S
TLY-A2540P		8.50	2.48 (22.0)	17.0	4.97 (44.0)	0.86	2094-AM01-S
		10.0	2.94 (26.0)	24.8	7.10 (63.0)		2094-AM02-S
TLY-A310M	4500	10.0	3.61 (31.9)	30.0	9.0 (79.6)	0.95	2094-AM02-S

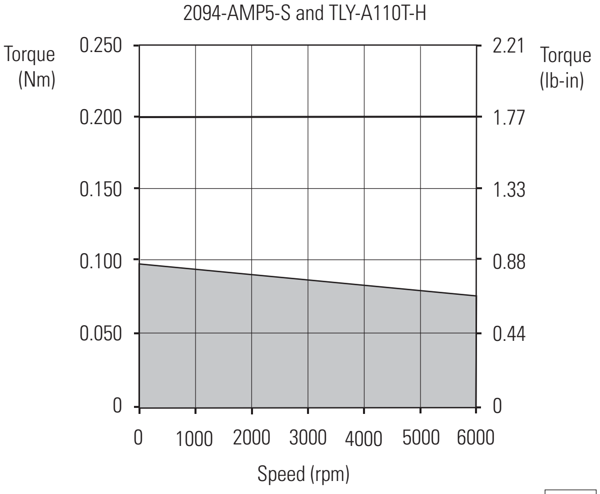
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

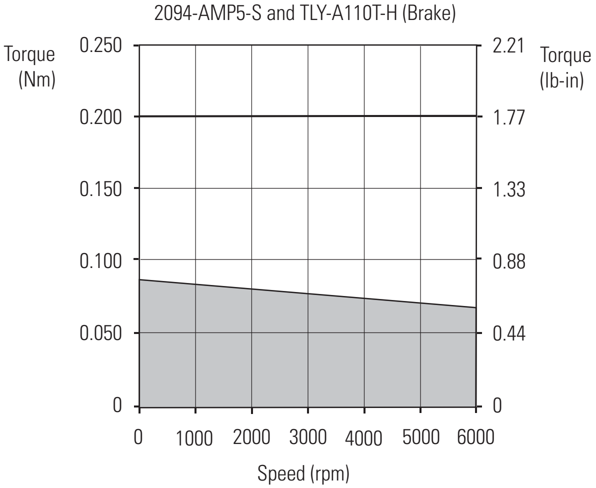
Bulletin TLY (brake) Motor Performance Specifications with Kinetix 6000 (230V) Drives

Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Torque** N•m (lb•in)	**Motor Rated Output** kW	**Kinetix 6000 230V Drives**
TLY-A110T	6000	0.50	0.086 (0.76)	1.50	0.20 (1.75)	0.037	2094-AMP5-S
TLY-A120T		0.93	0.163 (1.44)	2.50	0.36 (3.20)	0.077	2094-AMP5-S
TLY-A130T		1.67	0.293 (2.59)	4.90	0.76 (6.70)	0.13	2094-AMP5-S
TLY-A220T		3.15	0.757 (6.70)	7.90	1.48 (13.1)	0.24	2094-AMP5-S
TLY-A230T		4.95	1.16 (10.3)	10.5	2.07 (18.3)	0.32	2094-AMP5-S
		4.95	1.16 (10.3)	15.5	3.05 (27.0)		2094-AM01-S
TLY-A2530P	5000	8.50	2.20 (19.5)	17.0	4.18 (37.0)	0.55	2094-AM01-S
		10.0	2.60 (23.0)	21.0	5.20 (46.0)		2094-AM02-S
TLY-A2540P		8.50	2.48 (22.0)	17.0	4.97 (44.0)	0.66	2094-AM01-S
		10.0	2.94 (26.0)	24.8	7.10 (63.0)		2094-AM02-S
TLY-A310M	4500	10.0	3.61 (31.9)	30.0	9.0 (79.6)	0.90	2094-AM02-S

Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 6000 (230V) Drives/TLY-A*xxxx*-H (incremental) Motor Curves

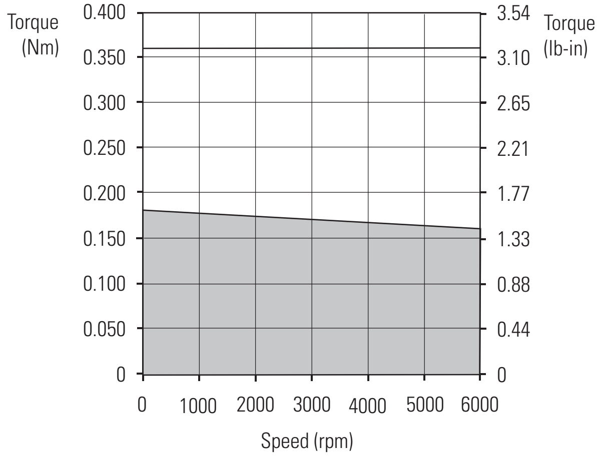

2094-AMP5-S and TLY-A110T-H


2094-AMP5-S and TLY-A110T-H (Brake)

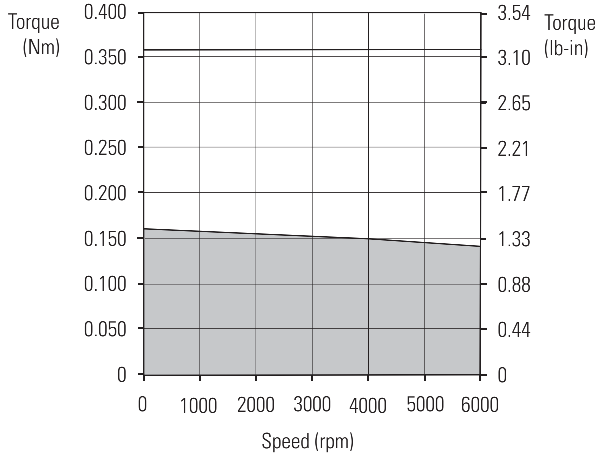
☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 6000 (230V) Drives/TLY-A*xxxx*-H (incremental) Motor Curves, Continued

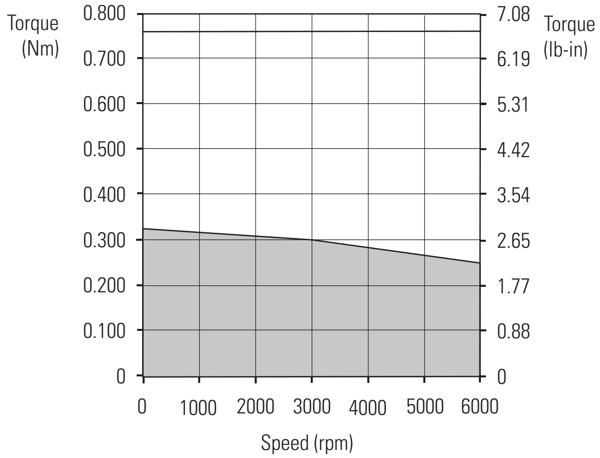
2094-AMP5-S and TLY-A120T-H



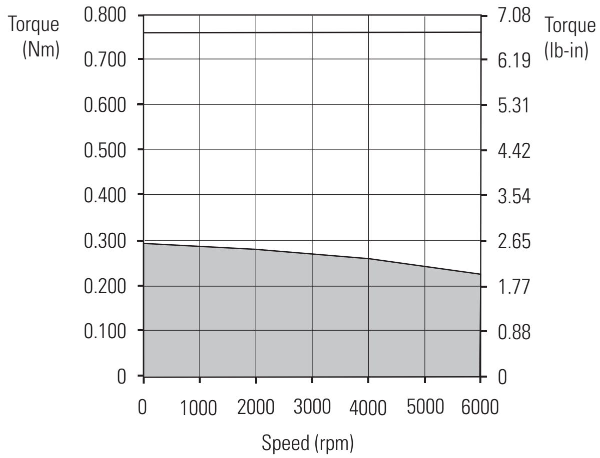
2094-AMP5-S and TLY-A120T-H (Brake)



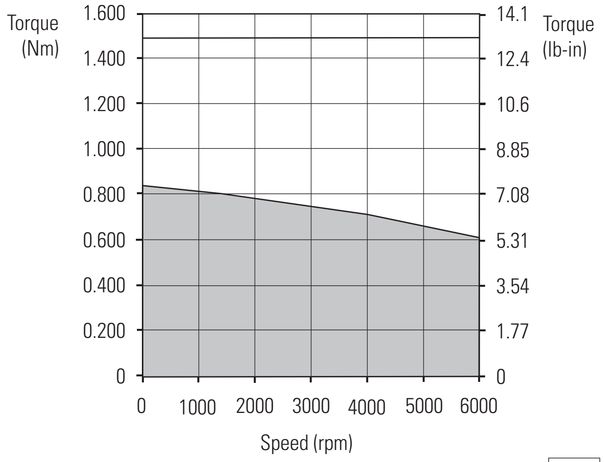
2094-AMP5-S and TLY-A130T-H



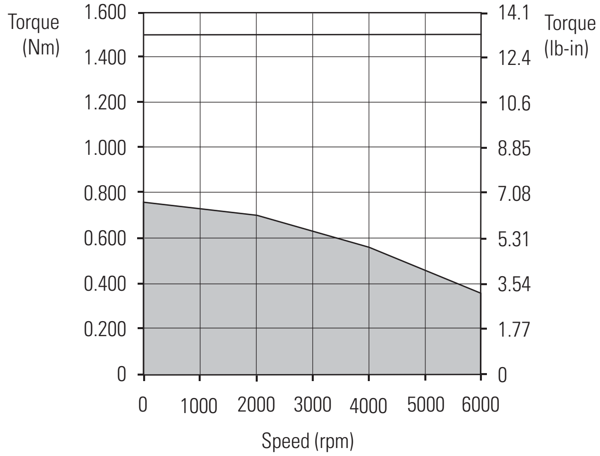
2094-AMP5-S and TLY-A130T-H (Brake)



2094-AMP5-S and TLY-A220T-H

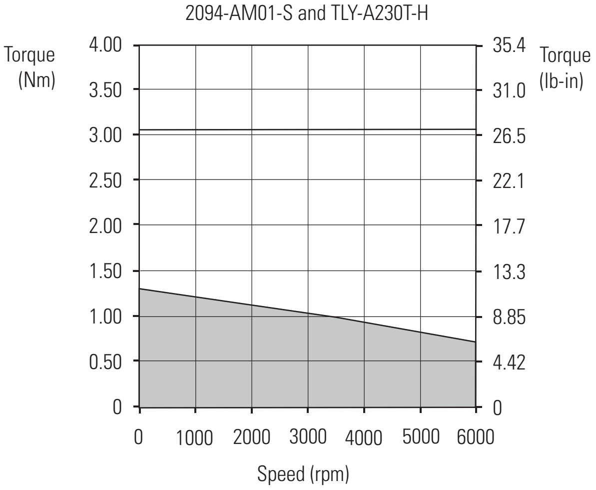


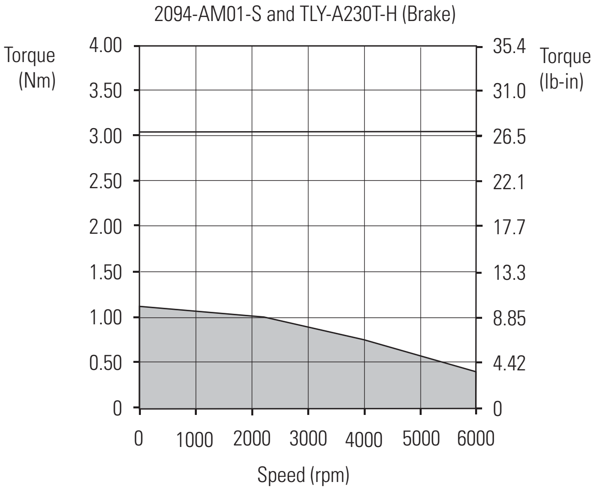
2094-AMP5-S and TLY-A220T-H (Brake)

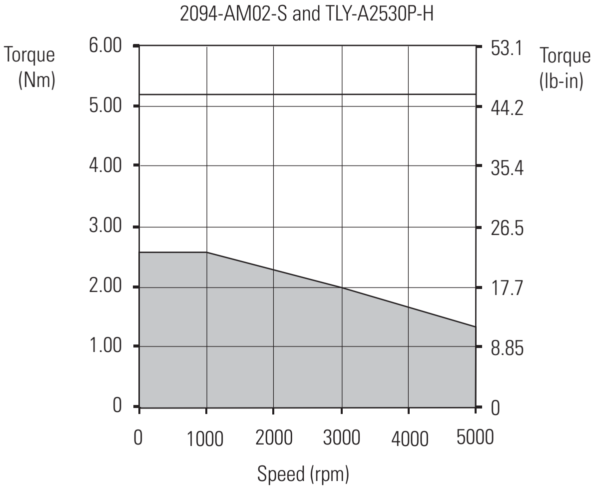


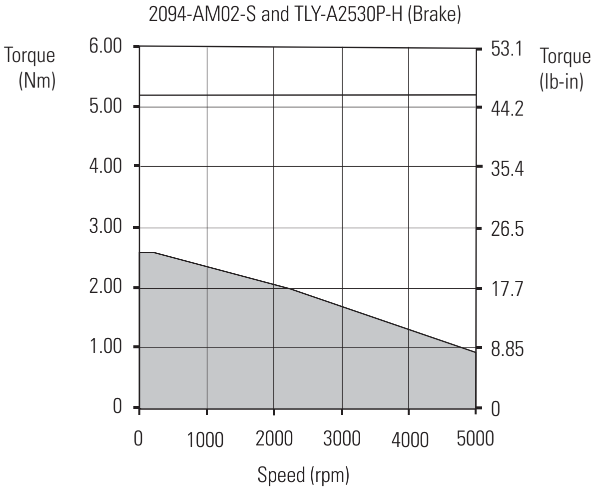
☐ = Intermittent operating region
▨ = Continuous operating region

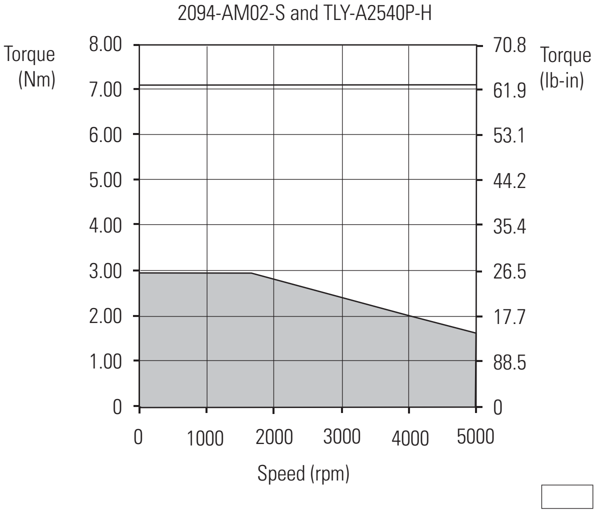
Kinetix 6000 (230V) Drives/TLY-A*xxxx*-H (incremental) Motor Curves, Continued

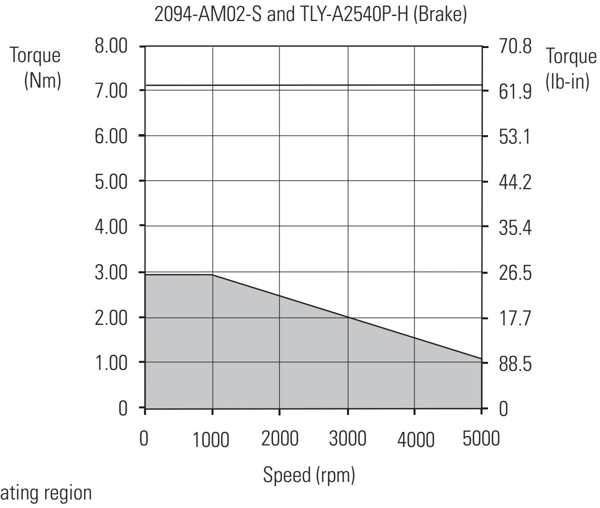






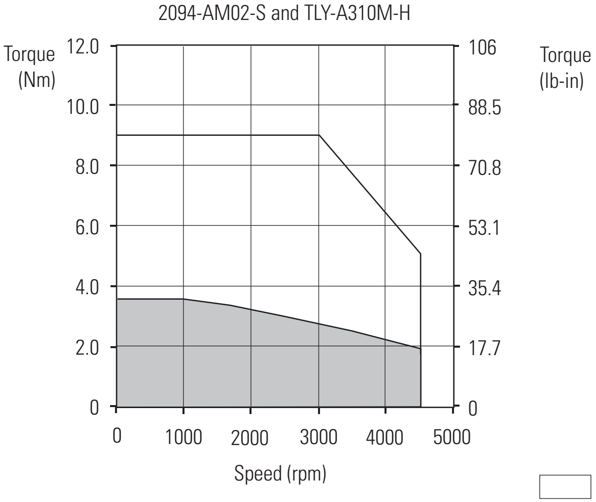


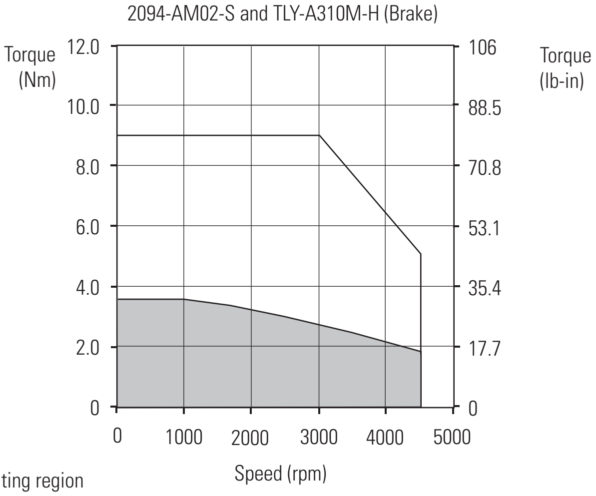




☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 6000 (230V) Drives/TLY-A*xxxx*-H (incremental) Motor Curves, Continued





☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 2000 Drives with MP-Series Low Inertia Motors

This section provides system combination information for the Kinetix 2000 drives when matched with MP-Series low-inertia motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

IMPORTANT	The MP-Series low-inertia motors on this page are equipped with DIN connectors (specified by 4 or 7 in the catalog number) and are not compatible with cables designed for motors equipped with bayonet connectors (specified by 2 in the catalog number). The motors with bayonet connectors (for example, MPL-A310P-*xx*2*x*AA) are being discontinued and require 2090-XXN*x*MP (bayonet) cables. For help with migration or to select bayonet cables, contact your Rockwell Automation sales representative.

Bulletin MPL Motor Cable Combinations

Motor Cat. No.	Motor Power/Brake Cable	Motor Feedback Cable (1)
MPL-A1510V-*xx*4*x*AA, MPL-A1520U-*xx*4*x*AA, MPL-A1530U-*xx*4*x*AA	2090-XXNPMF-16S*xx* (standard) 2090-CP*x*M4DF-16AF*xx* (continuous-flex)	2090-XXNFMF-S*xx* (standard) 2090-CFBM4DF-CDAF*xx* (continuous-flex) Absolute High-resolution and Incremental Feedback
MPL-A210V-*xx*4*x*AA, MPL-A220T-*xx*4*x*AA, MPL-A230P-*xx*4*x*AA		
MPL-A310F-*xx*7*x*AA, MPL-A310P-*xx*7*x*AA, MPL-A320H-*xx*7*x*AA, MPL-A320P-*xx*7*x*AA, MPL-A330P-*xx*7*x*AA	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or (2) 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback 2090-XXNFMF-S*xx* (standard) (3) 2090-CFBM7DF-CDAF*xx* (continuous-flex) Incremental Feedback
MPL-A420P-*xx*7*x*AA, MPL-A430H-*xx*7*x*AA		
MPL-A4530F-*xx*7*x*AA, MPL-A4540C-*xx*7*x*AA		

(1) Use low-profile connector kit (catalog number 2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

(2) Applies to Kinetix 2000 drives and MPL-A3*xxx*-M/S…MPL-A45*xxx*-M/S motors with absolute high-resolution feedback.

(3) Applies to Kinetix 2000 drives and MPL-A3*xxx*-H…MPL-A45*xxx*-H motors with incremental feedback.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

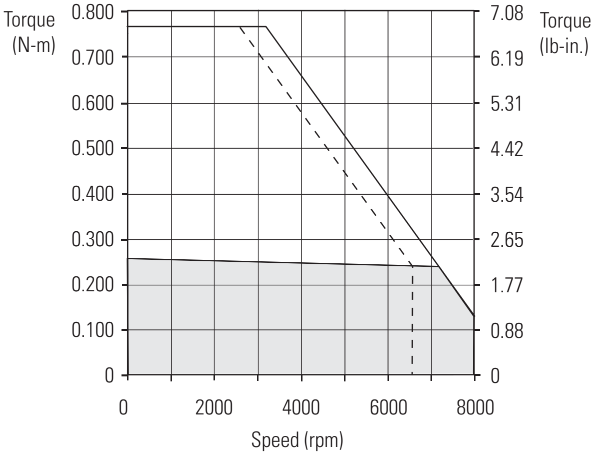
Bulletin MPL Motor Performance Specifications with Kinetix 2000 Drives

Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Stall Torque** N•m (lb•in)	**Motor Rated Output** kW	**Kinetix 2000 IAM/AM Module**
MPL-A1510V	8000	1.05	0.26 (2.3)	3.40	0.77 (6.8)	0.16	2093-AMP1
MPL-1520U	7000	1.80	0.49 (4.3)	6.10	1.58 (13.9)	0.27	2093-AMP2
MPL-1530U	7000	2.82	0.90 (8.0)	10.1	2.82 (24.9)	0.39	2093-AMP5
MPL-210V	8000	3.09	0.55 (4.8)	10.2	1.52 (13.4)	0.37	2093-AMP5
MPL-220T	6000	4.54	1.61 (14.2)	15.5	4.74 (41.9)	0.62	2093-AM01
MPL-230P	5000	5.40	2.10 (18.6)	23.0	8.2 (73.0)	0.86	2093-AM01
MPL-310F	3000	3.24	1.58 (14.0)	9.30	3.61 (31.9)	0.46	2093-AMP5
MPL-310P	5000	4.91	1.58 (14.0)	14.0	3.61 (31.9)	0.73	2093-AM01
MPL-320H	3500	6.10	3.05 (27.0)	19.3	7.91 (70.0)	1.0	2093-AM01
MPL-320P	5000	9.00	3.05 (27.0)	29.5	7.91 (70.0)	1.3	2093-AM02
MPL-330P	5000	12.0	4.18 (37.0)	38.0	11.1 (98.2)	1.8	2093-AM02
MPL-420P	5000	12.9	4.79 (42.3)	46.0	12.3 (109)	2.0	2093-AM02
MPL-430H	3500	12.2	6.21 (55.0)	45.0	18.7 (165)	1.8	2093-AM02
MPL-4530F	2800	13.40	8.36 (74.0)	42.0	19.7 (174)	1.9	2093-AM02
MPL-4540C	1500	9.55	10.30 (91.1)	29.0	27.1 (239)	1.5	2093-AM02

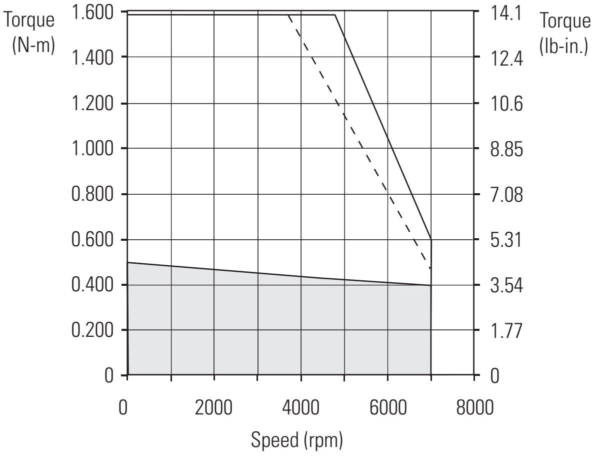
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 2000 Drives with MP-Series Low Inertia Motor Curves

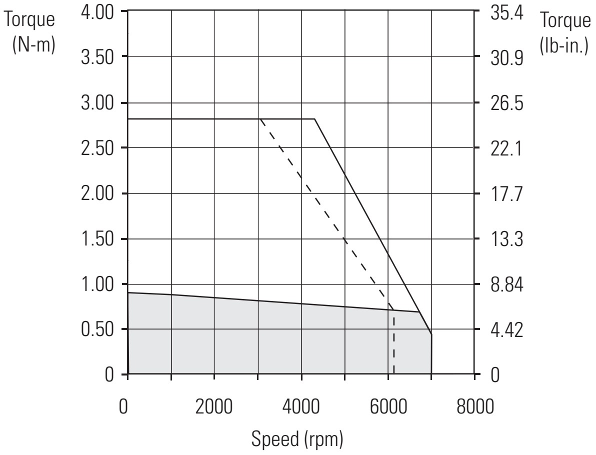
2093-AMP1 and MPL-A1510V



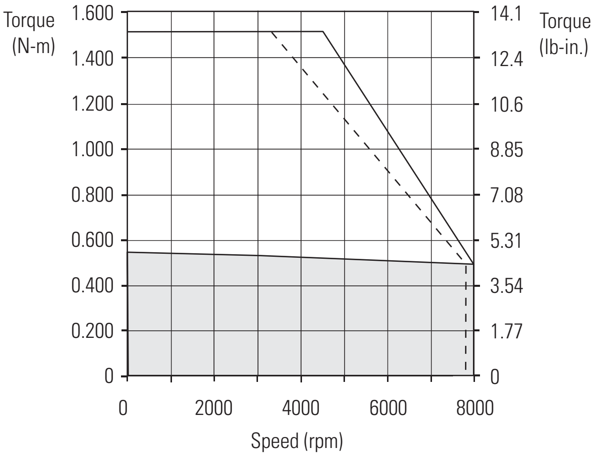
2093-AMP2 and MPL-A1520U



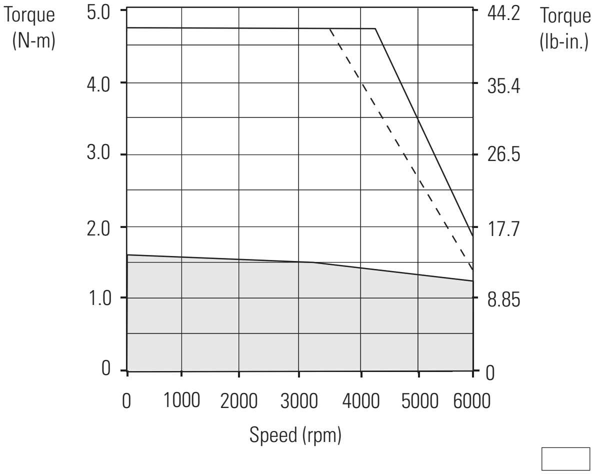
2093-AMP5 and MPL-A1530U



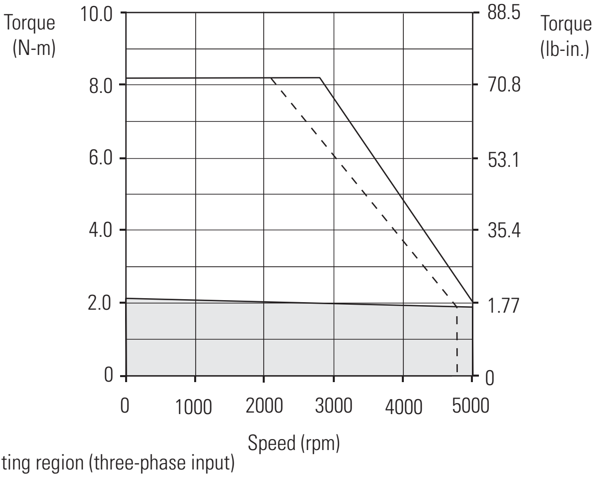
2093-AMP5 and MPL-A210V



2093-AM01 and MPL-A220T



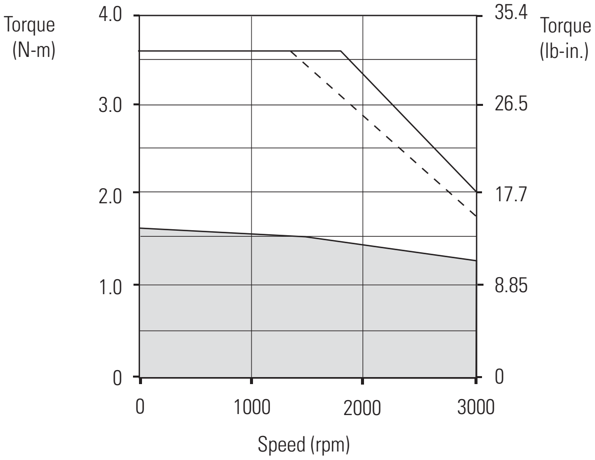
2093-AM02 and MPL-A230P



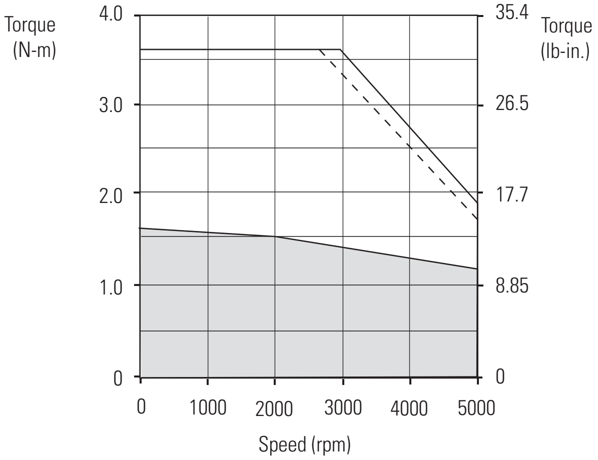
☐ = Intermittent operating region (three-phase input)
▨ = Continuous operating region
- - - = Drive operation (single-phase input)

Kinetix 2000 Drives with MP-Series Low Inertia Motor Curves, Continued

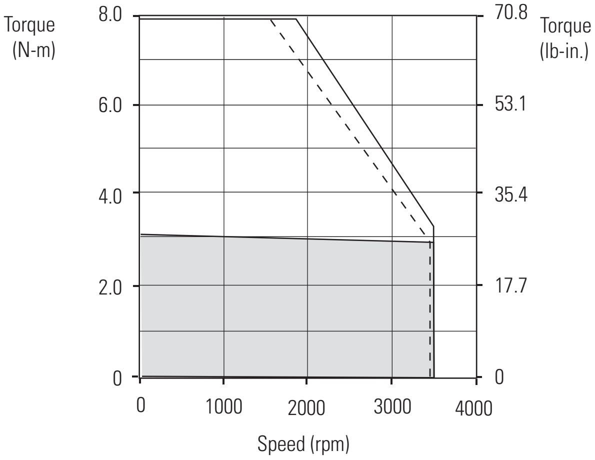
2093-AMP5 and MPL-A310F



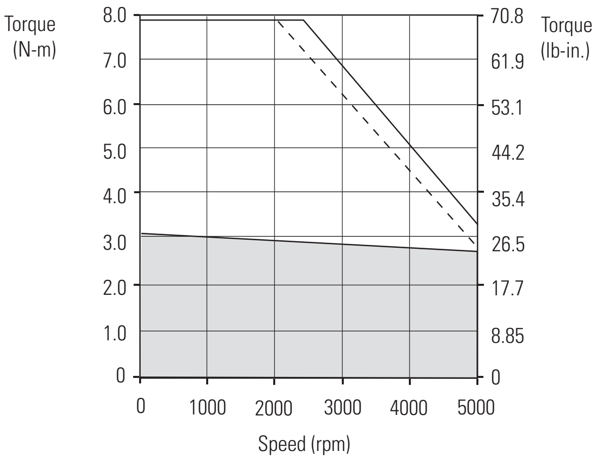
2093-AM01 and MPL-A310P



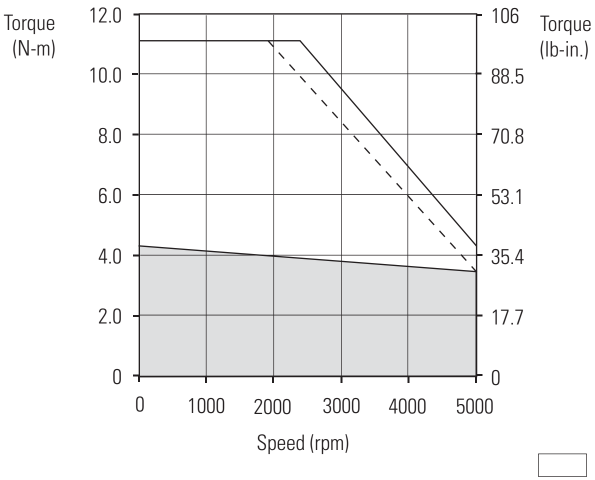
2093-AM01 and MPL-A320H



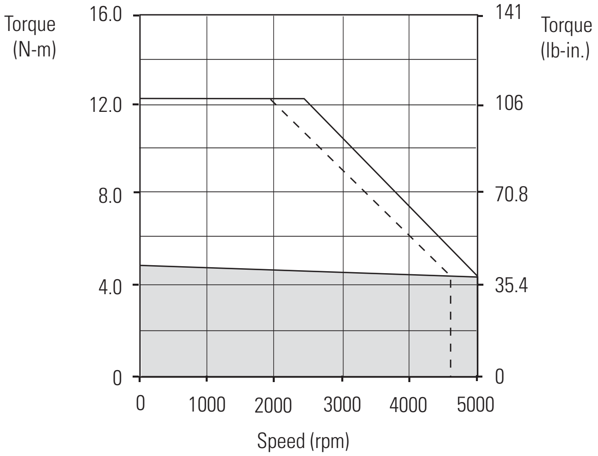
2093-AM02 and MPL-A320P



2093-AM02 and MPL-A330P

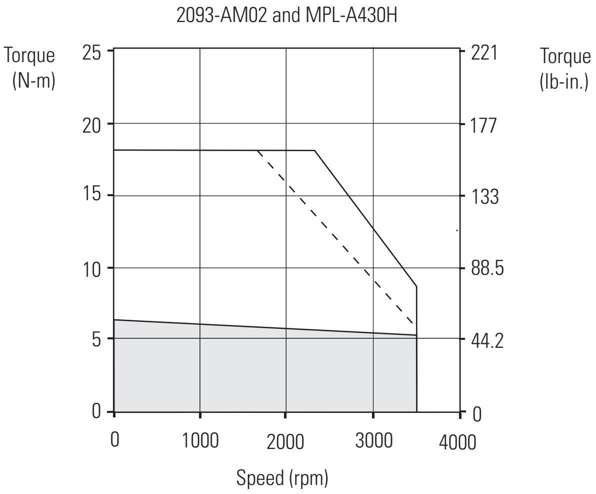


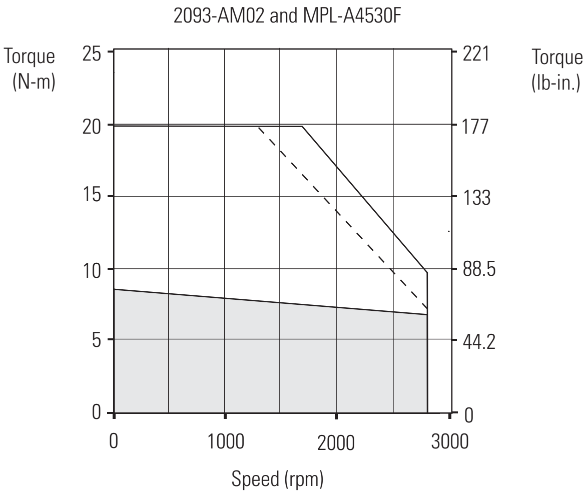
2093-AM02 and MPL-A420P

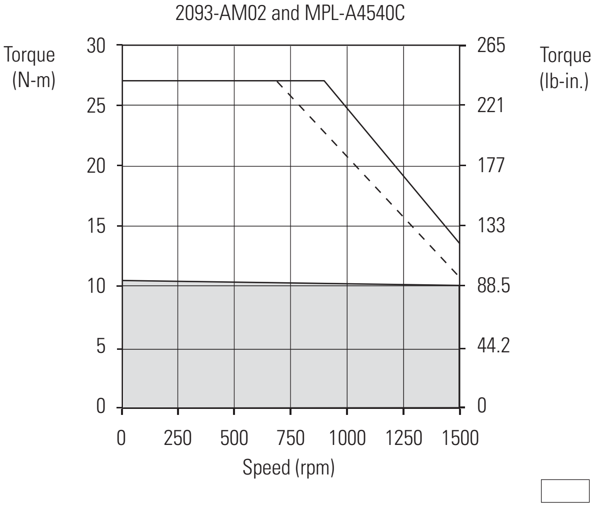


☐ = Intermittent operating region (three-phase input)
▨ = Continuous operating region
- - - = Drive operation (single-phase input)

Kinetix 2000 Drives with MP-Series Low Inertia Motor Curves, Continued







☐ = Intermittent operating region (three-phase input)
▨ = Continuous operating region
- - - = Drive operation (single-phase input)

Kinetix 2000 Drives with MP-Series Medium Inertia Motors

This section provides system combination information for the Kinetix 2000 (230V) drives when matched with MP-Series medium-inertia motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin MPM Motor Cable Combinations

Motor Cat. No. (200V class)	Motor Power/Brake Cable	Motor Feedback Cable [1]
MPM-A1151M, MPM-A1152F	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback

(1) Use low-profile connector kit (catalog number 2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

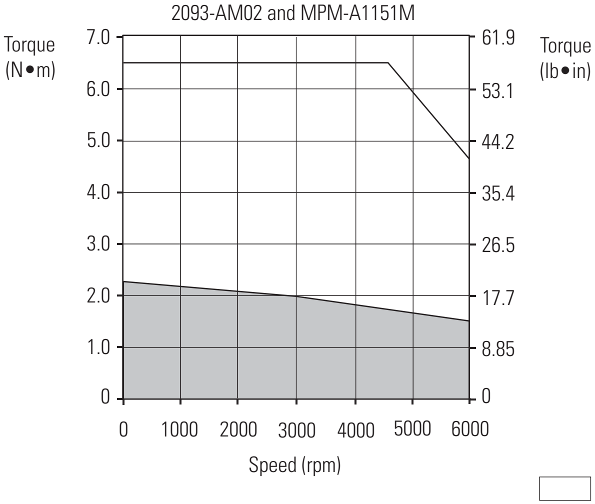
Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

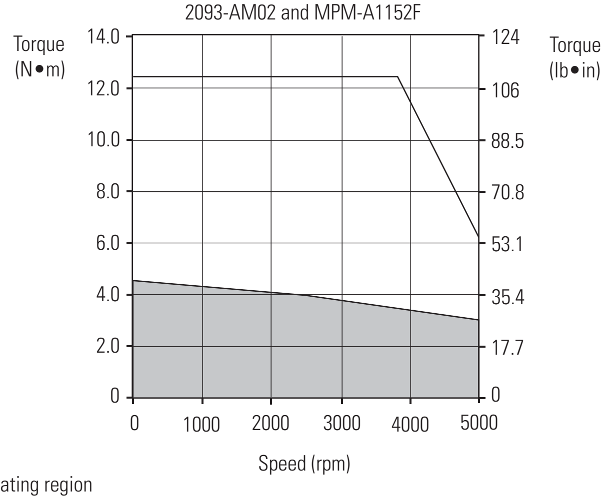
Bulletin MPM Motor Performance Specifications with Kinetix 2000 Drives

Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 2000 AM Module
MPM-A1151M	6000	8.40	2.3 (20.3)	25.5	5.7 (50.4)	0.90	2093-AM01
		10.3		30.5	6.6 (58.4)		2093-AM02
MPM-A1152F	5000	13.4	4.7 (41.6)	40.3	12.4 (110)	1.40	2093-AM02

Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 2000 Drives with MP-Series Medium Inertia Motor Curves





☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 2000 Drives with MP-Series Food Grade Motors

This section provides system combination information for the Kinetix 2000 drives when matched with MP-Series food-grade motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin MPF Motor Cable Combinations

Motor Cat. No.	Motor Power/Brake Cable	Motor Feedback Cable (1)
MPF-A310P, MPF-320H, MPF-A320P, MPF-A330P	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback
MPF-A430H		

(1) Use low-profile connector kit (catalog number 2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

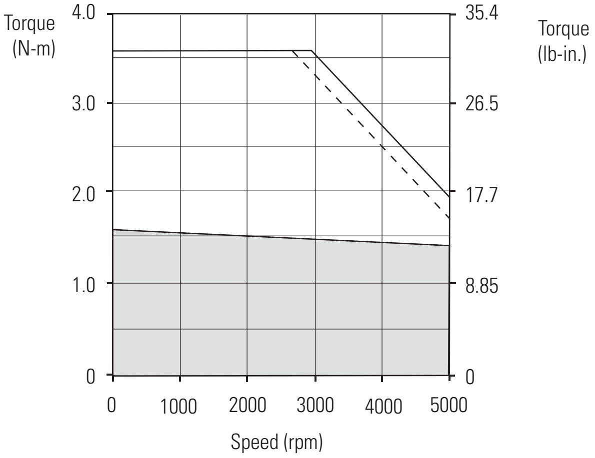
Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

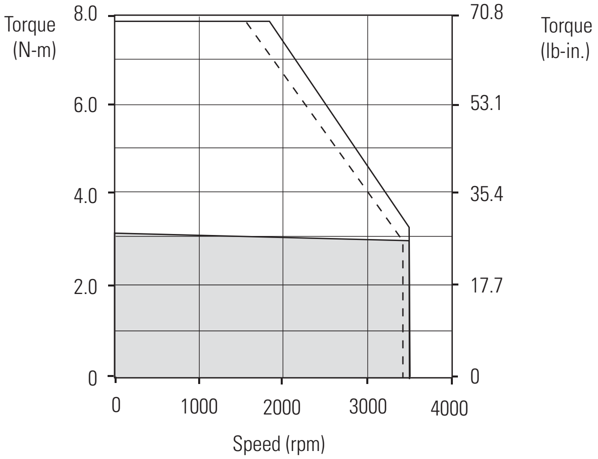
Bulletin MPF Motor Performance Specifications with Kinetix 2000 Drives

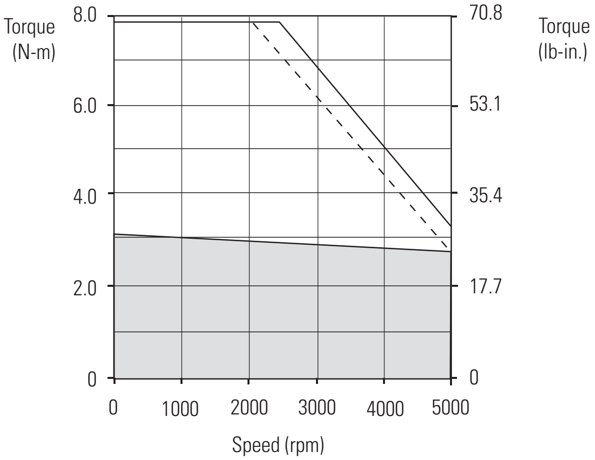
Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 2000 AM Module
MPF-A310P	5000	4.91	1.58 (14.0)	14.0	3.61 (31.9)	0.73	2093-AM01
MPF-A320H	3500	6.10	3.05 (27.0)	19.3	7.91 (70.0)	1.0	2093-AM01
MPF-A320P	5000	9.00	3.05 (27.0)	29.5	7.91 (70.0)	1.3	2093-AM02
MPF-A330P	5000	12.0	4.18 (37.0)	38.0	10.32 (91.2)	1.6	2093-AM02
MPF-A430H	3500	12.2	6.21 (55.0)	45.0	18.0 (159)	1.8	2093-AM02

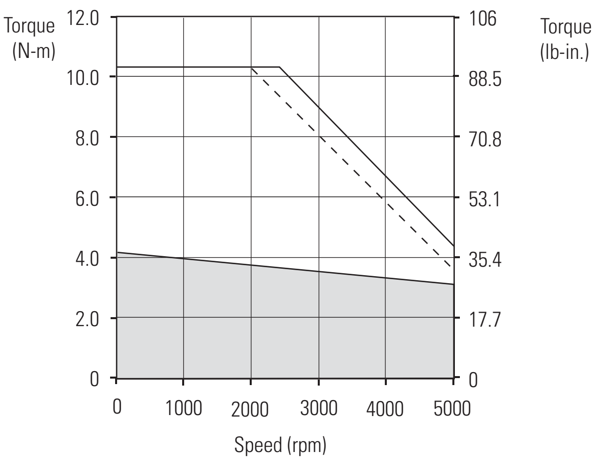
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

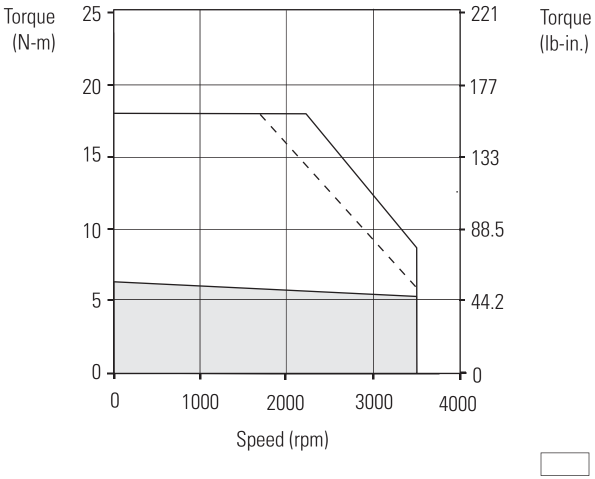
Kinetix 2000 Drives/MP-Series Food Grade Motor Curves


2093-AM01 and MPF-A310P


2093-AM01 and MPF-A320H


2093-AM02 and MPF-A320P


2093-AM02 and MPF-A330P


2093-AM02 and MPF-A430H

☐ = Intermittent operating region (three-phase input)
▨ = Continuous operating region
- - - = Drive operation (single-phase input)

Kinetix 2000 Drives with MP-Series Stainless Steel Motors

This section provides system combination information for the Kinetix 2000 drives when matched with MP-Series stainless-steel motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin MPS Motor Cable Combinations

Motor Cat. No. (200V class)	Motor Power/Brake Cable	Motor Feedback Cable [1]
MPS-A330P	2090-XXNPMF-16S*xx* [2]	2090-XXNFMF-S*xx* [3] Absolute High-resolution Feedback

(1) Use low-profile connector kit (catalog number 2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

(2) These cables are available as standard (catalog number 2090-XXNPMF-16S*xx*) or continuous-flex (catalog number 2090-CP*x*M4DF-16AF*xx*).

(3) These cables are available as standard (catalog number 2090-XXNFMF-S*xx*) or continuous-flex (catalog number 2090-CFBM4DF-CDAF*xx*).

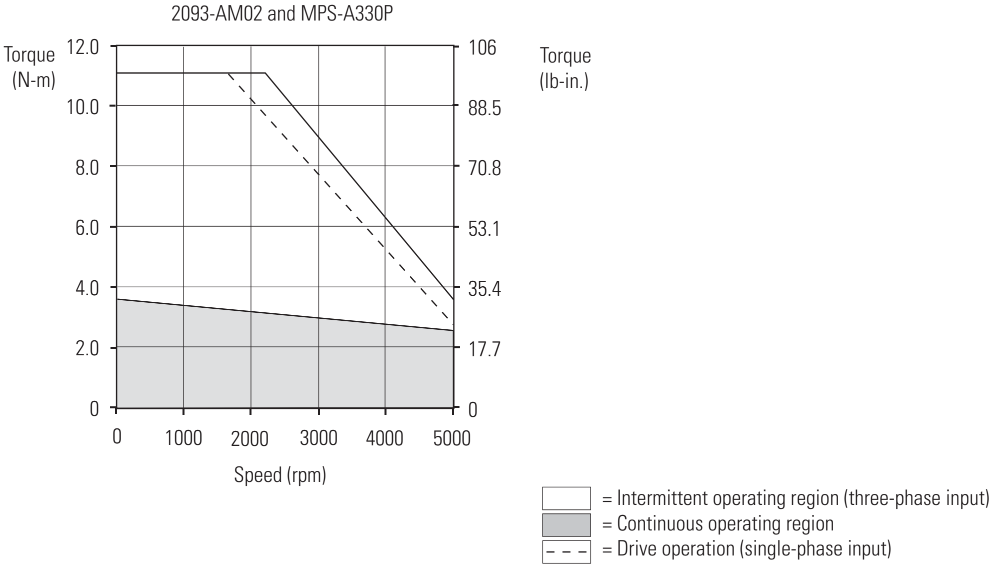
Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Bulletin MPS Motor Performance Specifications with Kinetix 2000 Drives

Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 2000 AM Module
MPS-A330P	5000	9.80	3.60 (32.0)	38.0	11.10 (98)	1.3	2093-AM02

Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 2000 Drives with MP-Series Stainless Steel Motor Curves



Kinetix 2000 Drives with TL-Series Low Inertia Motors

This section provides system combination information for the Kinetix 2000 drives when matched with TL-Series (Bulletin TLY) low-inertia motors. Compatible TL-Series motors are equipped with absolute high-resolution or incremental encoder feedback. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin TLY Motor Cable Combinations

Motor Cat. No.	Motor Power/Brake Cable	Motor Feedback Cable [1]
TLY-A110*x*, TLY-A120*x*, TLY-A130*x*	2090-CPWM6DF-16AA*xx* (standard) (without brake)	2090-CFBM6DF-CBAA*xx* (standard) Absolute High-resolution
TLY-A220*x*, TLY-A230*x*		
TLY-A2530P, TLY-A2540P	2090-CPBM6DF-16AA*xx* (standard) (with brake)	2090-CFBM6DF-CBAA*xx* or 2090-CFBM6DD-CCAA*xx* (standard) Incremental Feedback
TLY-A310M		

(1) For TLY-A*xxxx*-H motors with incremental encoder feedback, use 2090-CFBM6DF-CBAA*xx* flying-lead cables and 2090-K2CK-D15M connector kit (battery not required) or panel-mounted breakout components on the drive end. Refer to Breakout Components and Connector Kits beginning on page 440 for more information.

The TLY-A*xxxx*-B motors with 17-bit high resolution encoder feedback require the 2090-CFBM6DF-CBAA*xx* flying-lead feedback cable and 2090-K2CK-D15M connector kit with 2090-DA-BAT2 battery.

TL-Series (Bulletin TLY) motors are characterized as having 1000 mm (39.4 in.) cable extensions with circular plastic connectors and TLY-A*xxx* catalog numbers.
Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Bulletin TLY (non-brake) Motor Performance Specifications with Kinetix 2000 Drives

Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 2000 IAM/AM Module
TLY-A110*x*	6000 [1]	0.55	0.096 (0.85)	1.50	0.20 (1.75)	0.041	2093-AMP1
TLY-A120*x*		1.03	0.181 (1.60)	2.50	0.36 (3.20)	0.086	2093-AMP1
TLY-A130*x*		1.85	0.325 (2.88)	4.90	0.76 (6.70)	0.14	2093-AMP2
TLY-A220*x*		3.50	0.836 (7.40)	7.90	1.48 (13.1)	0.35	2093-AMP5
TLY-A230*x*		5.50	1.30 (11.5)	15.5	3.05 (27.0)	0.44	2093-AM01
TLY-A2530P	5000	10.0	2.60 (23.0)	21.0	5.20 (46.0)	0.69	2093-AM02
TLY-A2540P		10.0	2.94 (26.0)	24.8	7.10 (63.0)	0.86	2093-AM02
TLY-A310M	4500	10.0	3.61 (31.9)	30.0	9.0 (79.6)	0.95	2093-AM02

(1) Applies to TLY-A*xxx*T-H motors with incremental feedback. The TLY-A*xxx*P-B motors with absolute high-resolution encoders are rated at 5000 rpm.

Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Bulletin TLY (brake) Motor Performance Specifications with Kinetix 2000 Drives

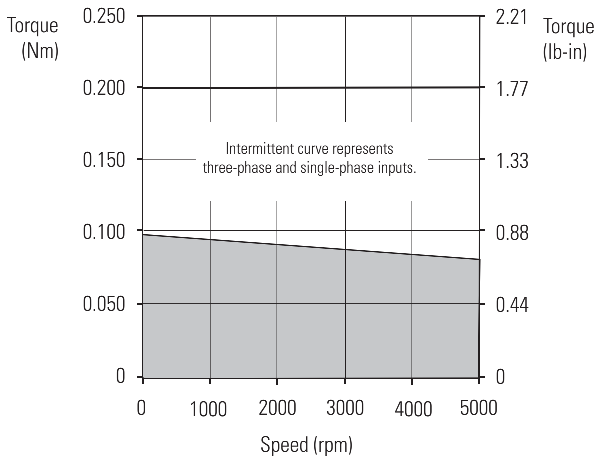
Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Torque** N•m (lb•in)	**Motor Rated Output** kW	**Kinetix 2000 IAM/AM Module**
TLY-A110*x*	6000 [(1)]	0.50	0.086 (0.76)	1.50	0.20 (1.75)	0.037	2093-AMP1
TLY-A120*x*		0.93	0.163 (1.44)	2.50	0.36 (3.20)	0.077	2093-AMP1
TLY-A130*x*		1.67	0.293 (2.59)	4.90	0.76 (6.70)	0.13	2093-AMP2
TLY-A220*x*		3.15	0.757 (6.70)	7.90	1.48 (13.1)	0.24	2093-AMP5
TLY-A230*x*		4.95	1.16 (10.3)	15.5	3.05 (27.0)	0.32	2093-AM01
TLY-A2530P	5000	10.0	2.60 (23.0)	21.0	5.20 (46.0)	0.55	2093-AM02
TLY-A2540P		10.0	2.94 (26.0)	24.8	7.10 (63.0)	0.66	2093-AM02
TLY-A310M	4500	10.0	3.61 (31.9)	30.0	9.0 (79.6)	0.90	2093-AM02

(1) Applies to TLY-A*xxx*T-H motors with incremental feedback. The TLY-A*xxx*P-B motors with absolute high-resolution encoders are rated at 5000 rpm.

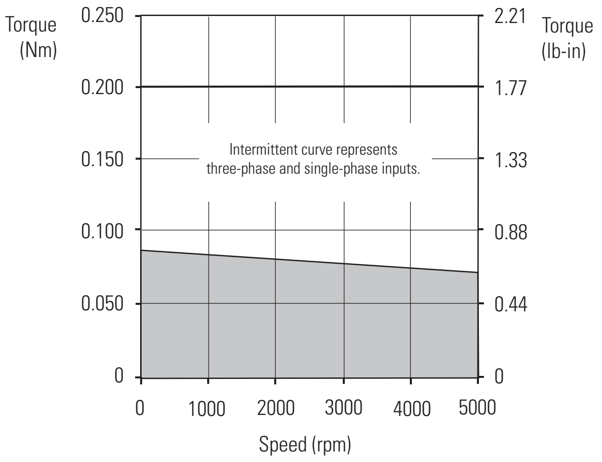
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 2000 Drives/TLY-A*xxx*P-B (absolute high-resolution) Motor Curves

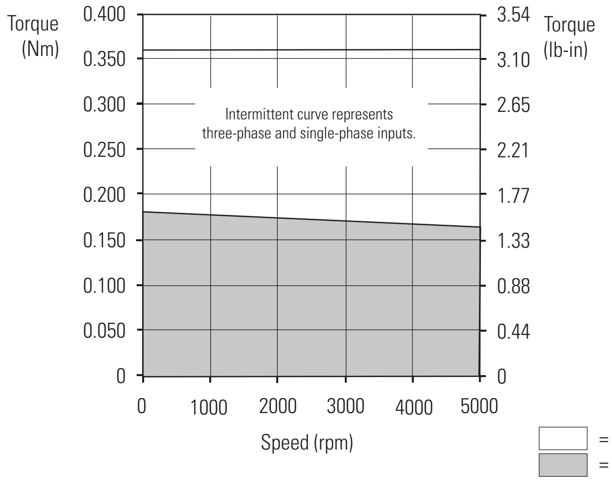
2093-AMP1 and TLY-A110P-B



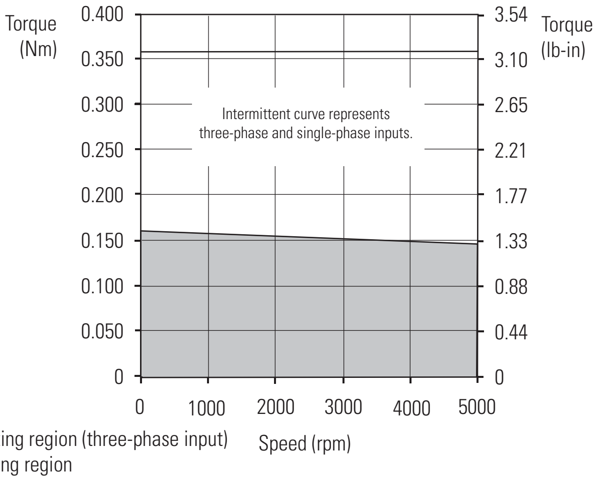
2093-AMP1 and TLY-A110P-B (Brake)



2093-AMP1 and TLY-A120P-B

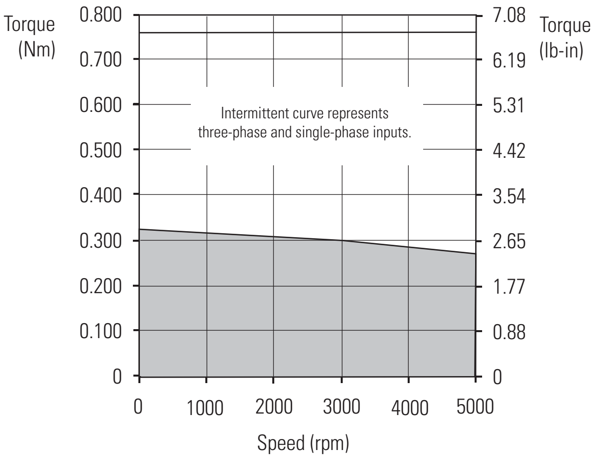


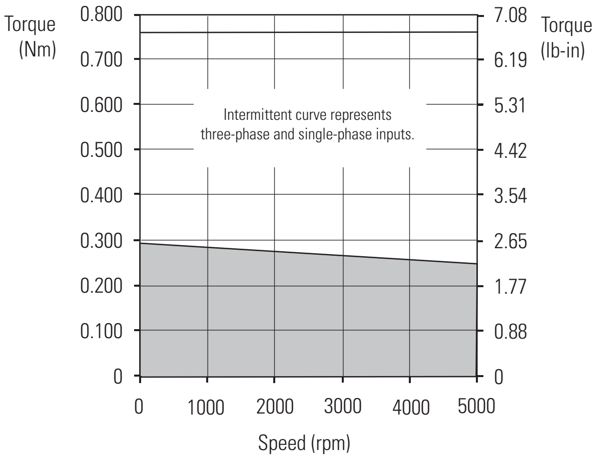
2093-AMP1 and TLY-A120P-B (Brake)

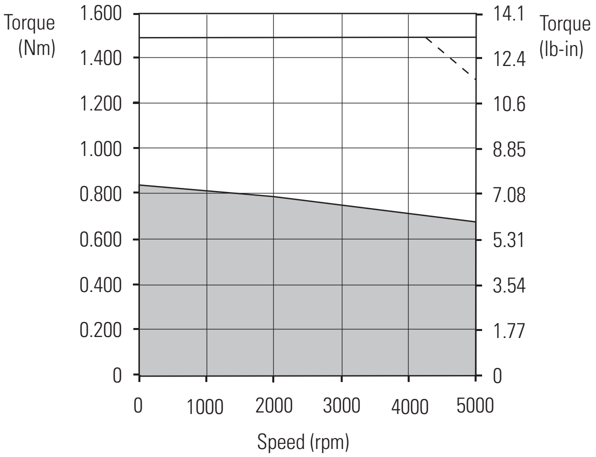


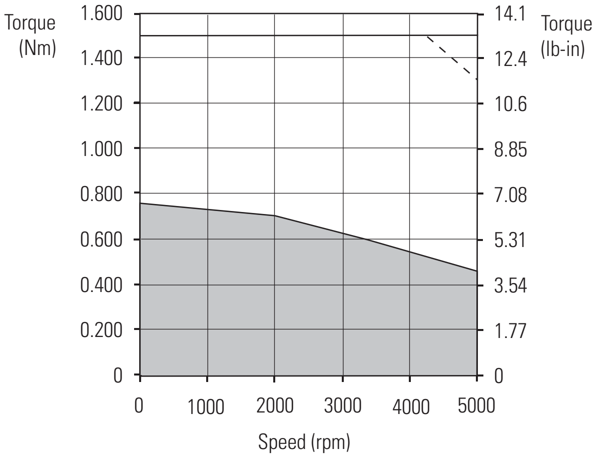
☐ = Intermittent operating region (three-phase input)
▨ = Continuous operating region
– – – = Drive operation (single-phase input)

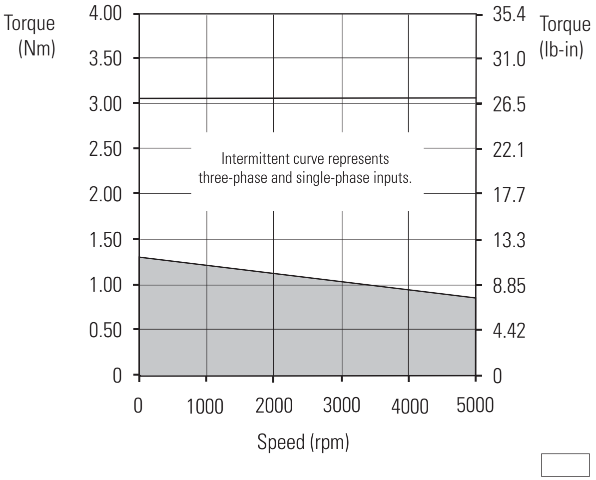
Kinetix 2000 Drives/TLY-A*xxx*P-B (absolute high-resolution) Motor Curves, Continued

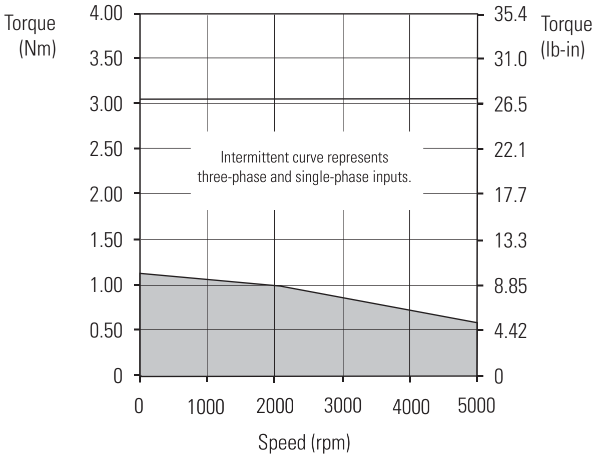

2093-AMP1 and TLY-A130P-B


2093-AMP2 and TLY-A130P-B (Brake)


2093-AMP5 and TLY-A220P-B


2093-AMP5 and TLY-A220P-B (Brake)

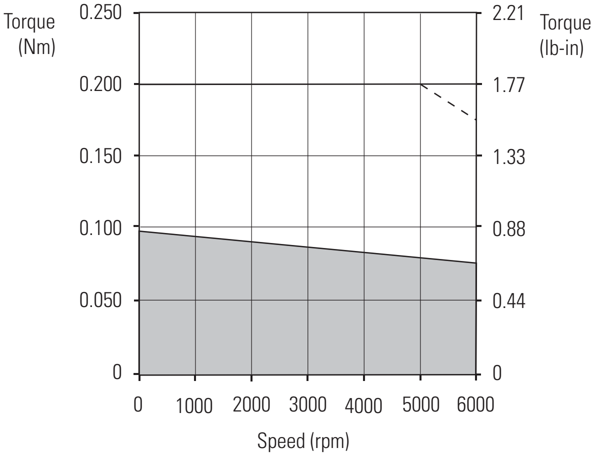

2093-AM01 and TLY-A230P-B


2093-AM01 and TLY-A230P-B (Brake)

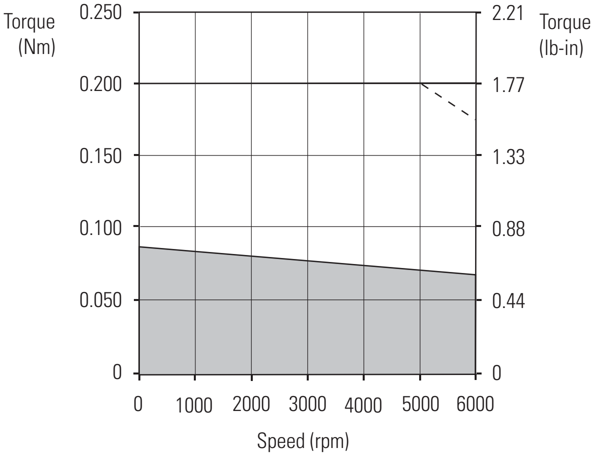
☐ = Intermittent operating region (three-phase input)
▨ = Continuous operating region
- - - = Drive operation (single-phase input)

Kinetix 2000 Drives/TLY-A*xxx*T-H (incremental) Motor Curves

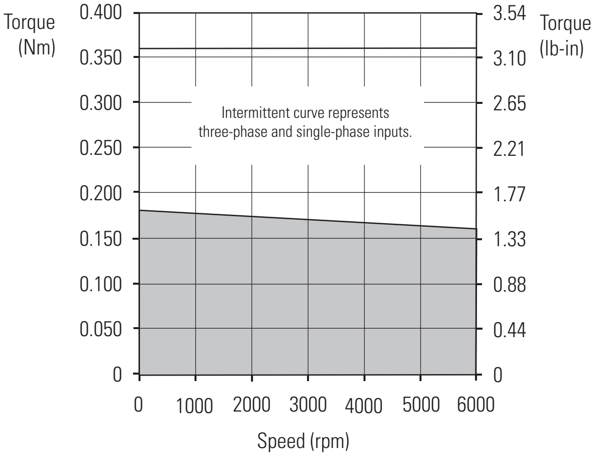
2093-AMP1 and TLY-A110T-H



2093-AMP1 and TLY-A110T-H (Brake)

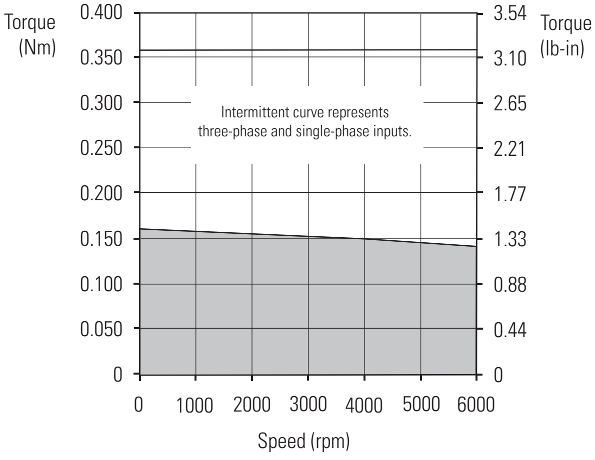


2093-AMP1 and TLY-A120T-H



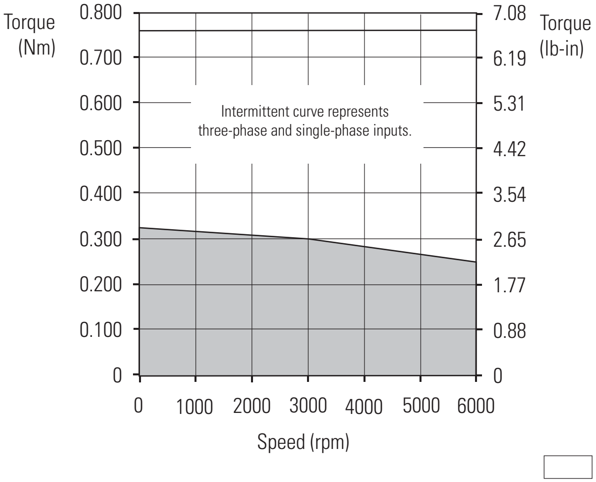
Intermittent curve represents three-phase and single-phase inputs.

2093-AMP1 and TLY-A120T-H (Brake)



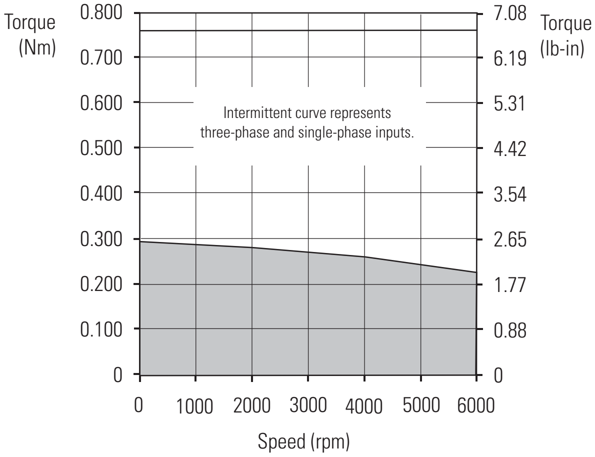
Intermittent curve represents three-phase and single-phase inputs.

2093-AMP1 and TLY-A130T-H



Intermittent curve represents three-phase and single-phase inputs.

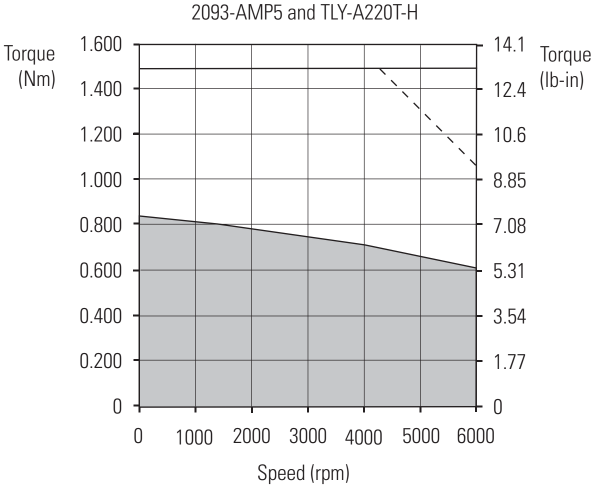
2093-AMP2 and TLY-A130T-H (Brake)

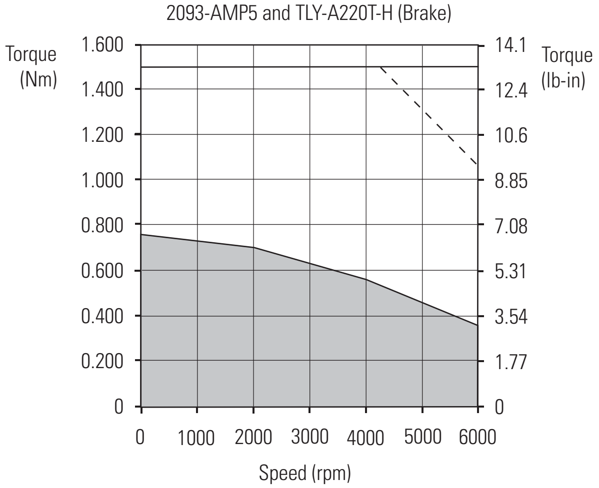


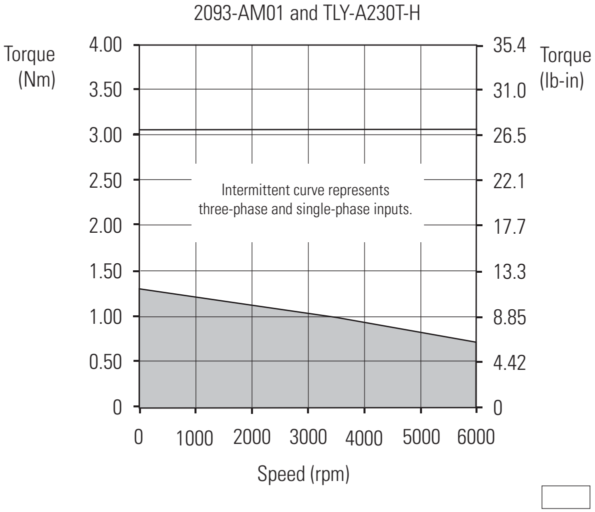
Intermittent curve represents three-phase and single-phase inputs.

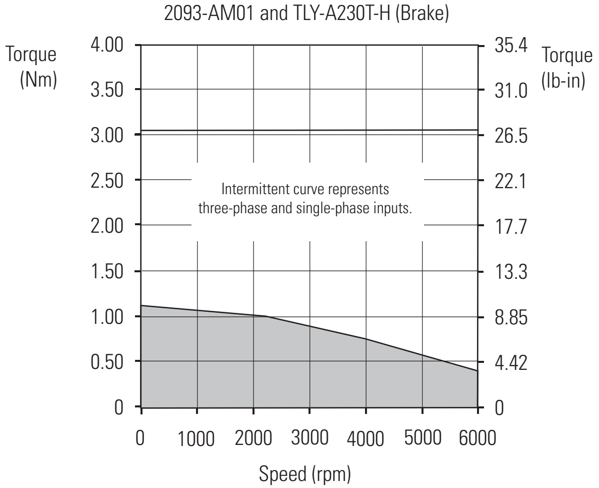
□ = Intermittent operating region (three-phase input)
▨ = Continuous operating region
- - - = Drive operation (single-phase input)

Kinetix 2000 Drives/TLY-A*xxx*T-H (incremental) Motor Curves, Continued



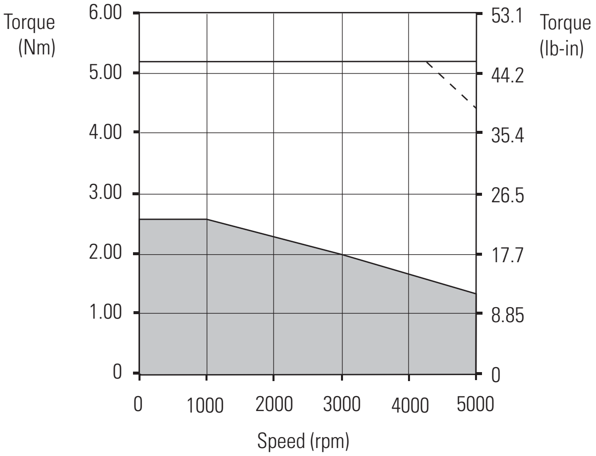


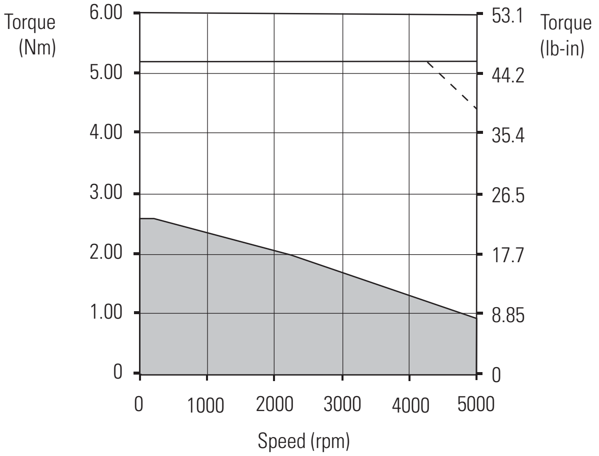


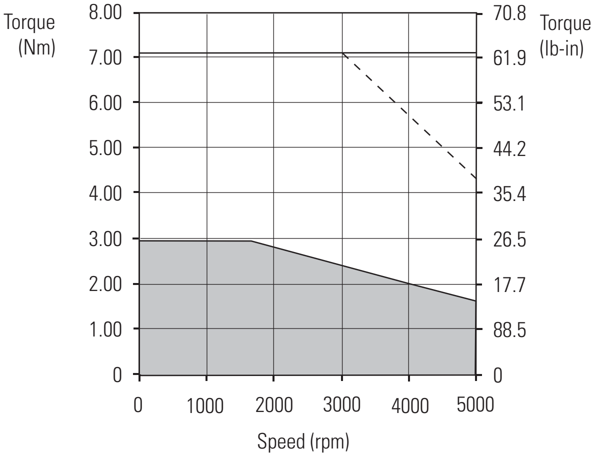


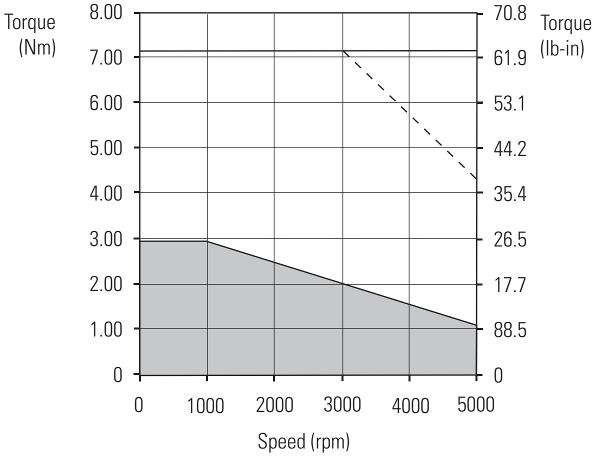
☐ = Intermittent operating region (three-phase input)
▨ = Continuous operating region
- - - = Drive operation (single-phase input)

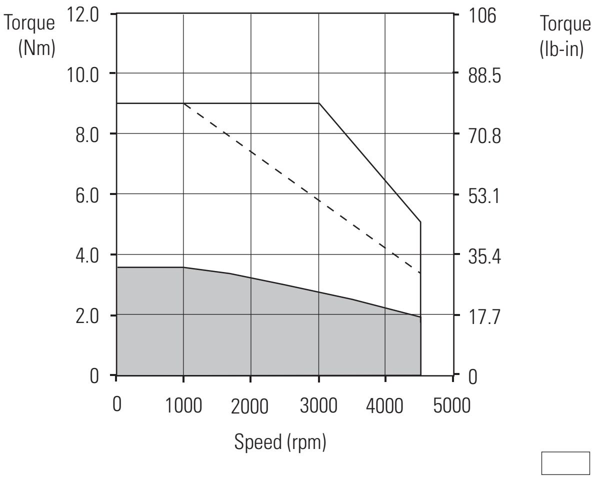
Kinetix 2000 Drives/TLY-A*xxxx*-*x* Motor Curves

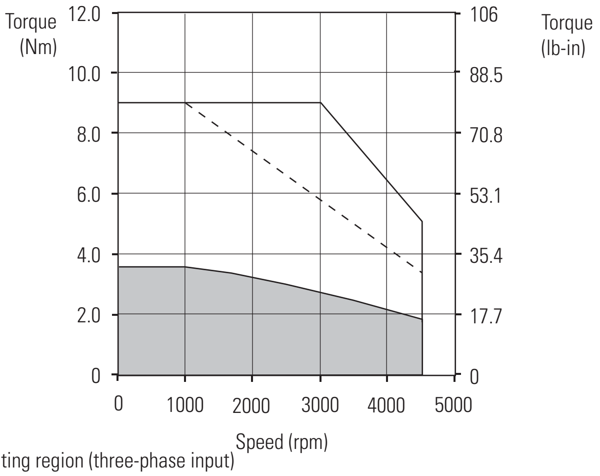

2093-AM02 and TLY-A2530P-*x*


2093-AM02 and TLY-A2530P-*x* (Brake)


2093-AM02 and TLY-A2540P-*x*


2093-AM02 and TLY-A2540P-*x* (Brake)


2093-AM02 and TLY-A310M-*x*


2093-AM02 and TLY-A310M-*x* (Brake)

☐ = Intermittent operating region (three-phase input)
▨ = Continuous operating region
- - - = Drive operation (single-phase input)

Kinetix 7000 Drives with HPK-Series Asynchronous Servo Motors

This section provides system combination information for the Kinetix 7000 drives when matched with HPK-Series motors. These motors are available with 460V and 400V windings. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin MPL Motor Cable Combinations (460V Motors)

Motor Cat. No.	Motor Power Cable	Motor Feedback Cable (1)
HPK-B1307C and HPK-B1308C	Customer Supplied	2090-XXNFMF-S*xx* (2) Absolute High-resolution Feedback
HPK-B1307E and HPK-B1310C		
HPK-B1308E, HPK-B1609E, and HPK-B1613C		
HPK-B1611E and HPK-B1613E		
HPK-B1815C		
HPK-B2010C, HPK-B2010E		
HPK-B2212C, HPK-B2510C		

(1) Use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

(2) These cables are available as standard (catalog number 2090-XXNFMF-S*xx*) or continuous-flex (catalog number 2090-CFBM4DF-CDAF*xx*).

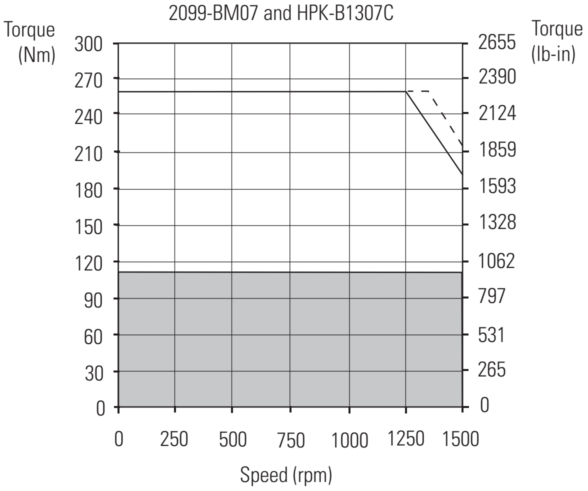
Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

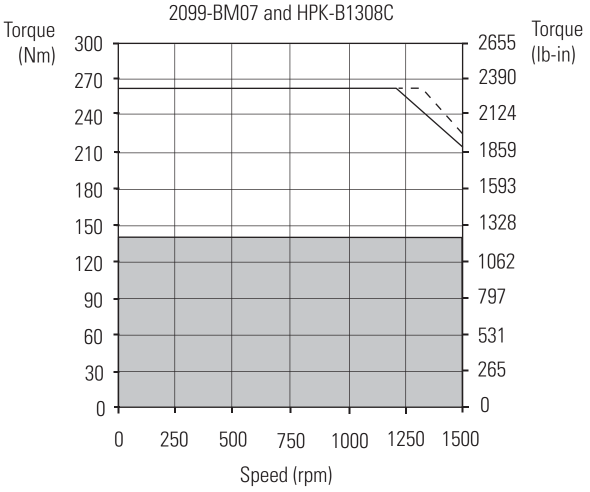
HPK-Series (460V) Performance Specifications with Kinetix 7000 (460V) Drives

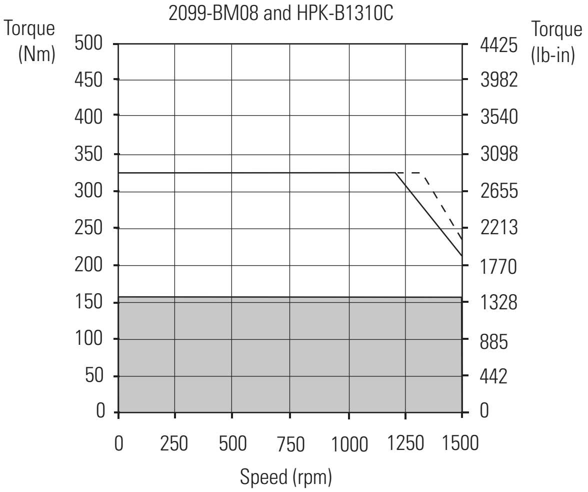
Rotary Motor	Base Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW (Hp)	Kinetix 7000 Drives
HPK-B1307C	1500	102.0	112 (991)	113.0	257 (2274)	17.1 (22.9)	2099-BM07
HPK-B1308C		59.6	141 (1248)	119.3	262 (2319)	21.6 (28.9)	2099-BM07
HPK-B1310C		64.9	155 (1372)	144.0	325 (2876)	23.8 (31.9)	2099-BM08
HPK-B1613C		109.8	271 (2398)	217.0	542 (4797)	41.7 (55.9)	2099-BM09
HPK-B1815C		153.7	360 (3186)	402.0	850 (7523)	55.9 (74.9)	2099-BM11
HPK-B2010C		196.4	482 (4266)	440.0	970 (8585)	75.0 (100.5)	2099-BM11
HPK-B2212C		278.0	714 (6319)	524.0	1356 (12,000)	112 (151)	2099-BM12
HPK-B2510C		351.0	865 (7656)	526.0	1216 (10,762)	136 (180)	2099-BM12
HPK-B1307E	3000	81.0	96.0 (849)	146.6	165 (1460)	29.8 (39.9)	2099-BM08
HPK-B1308E		91.4	115 (1018)	190.3	230 (2035)	35.7 (47.8)	2099-BM09
HPK-B1609E		124.4	156 (1381)	217.0	270 (2390)	48.4 (64.8)	2099-BM09
HPK-B1611E		149.0	183 (1619)	338.4	400 (3540)	57.0 (76.4)	2099-BM11
HPK-B1613E		191.0	237 (2097)	440.0	459 (4062)	73.7 (98.8)	2099-BM11
HPK-B2010E		254.0	295 (2610)	440.0	500 (4425)	92.0 (125.0)	2099-BM11

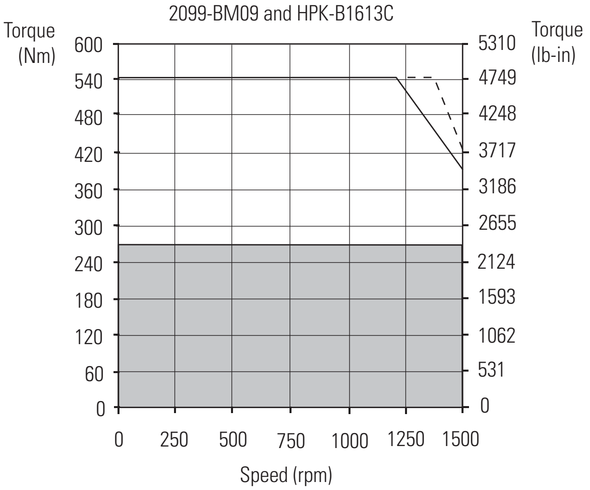
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

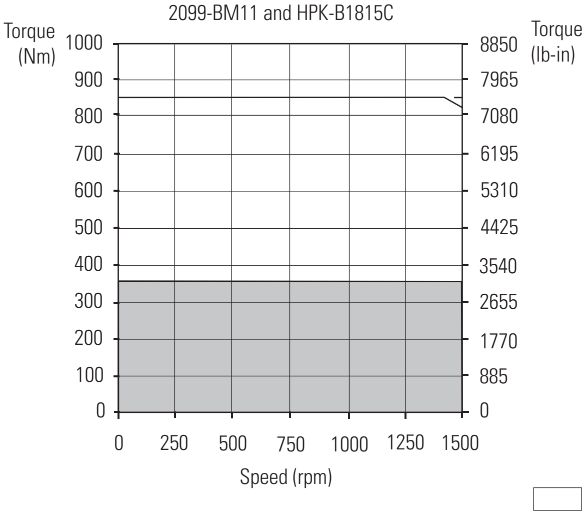
Kinetix 7000 (460V) Drives/HPK-Series (460V) Motor Curves

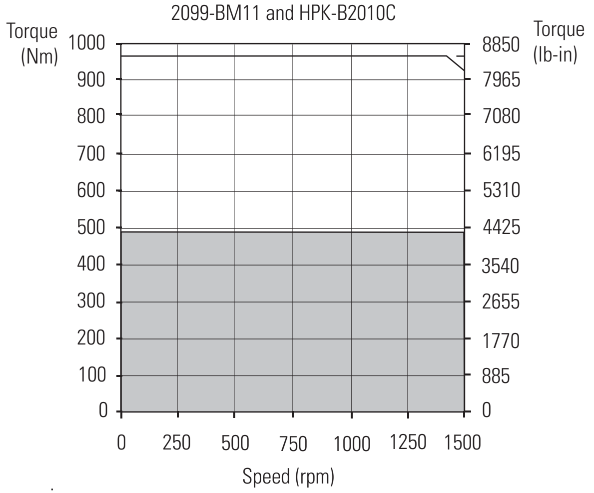






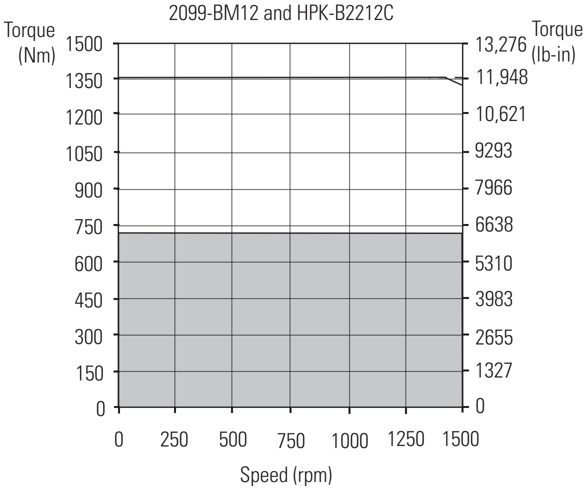


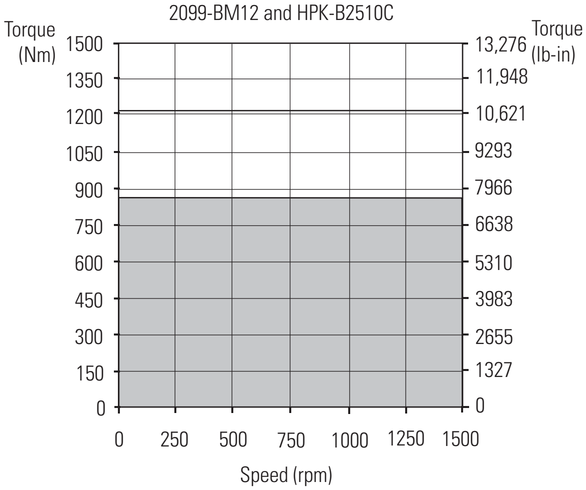


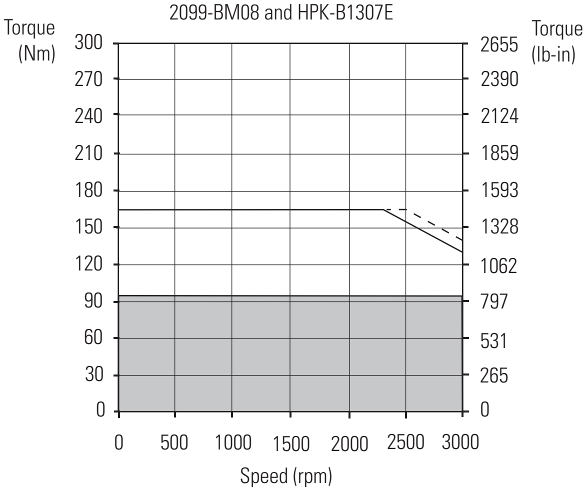


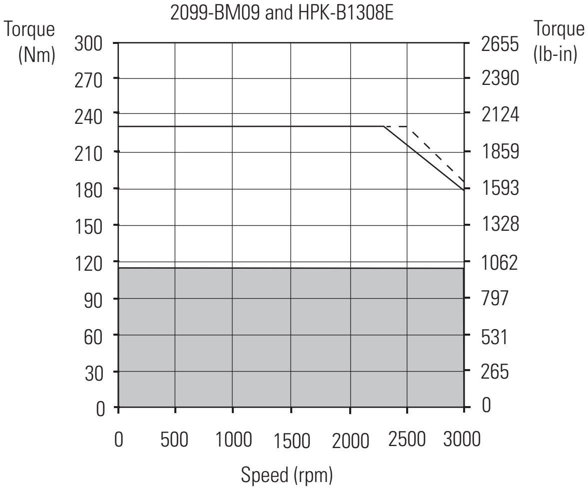
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 750V DC (rms) input voltage

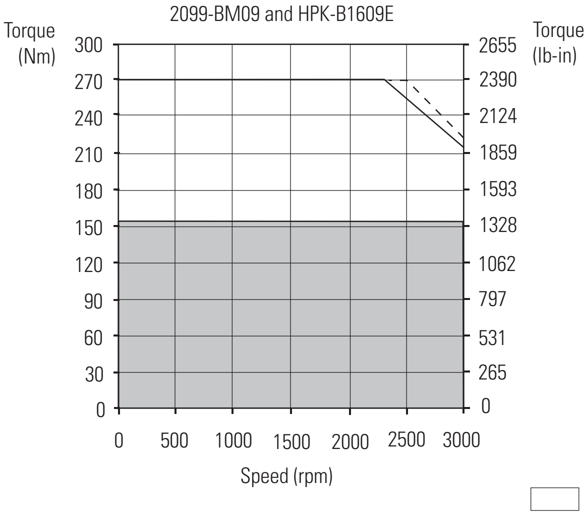
Kinetix 7000 (460V) Drives/HPK-Series (460V) Motor Curves, Continued

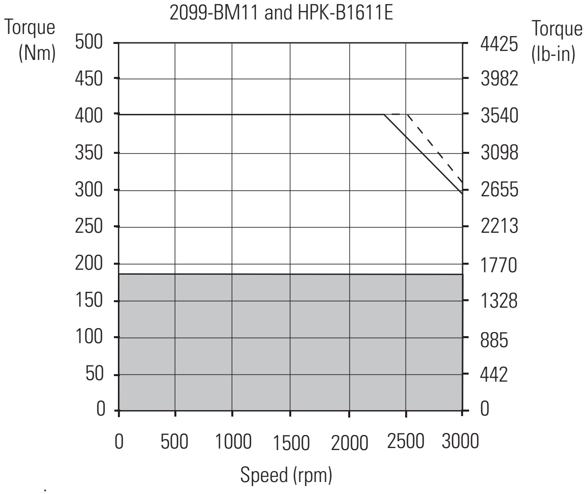






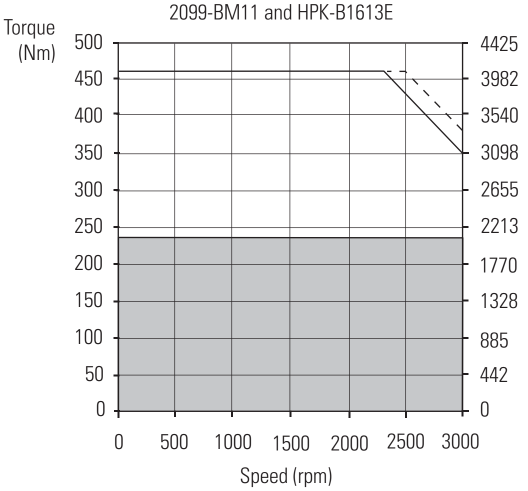


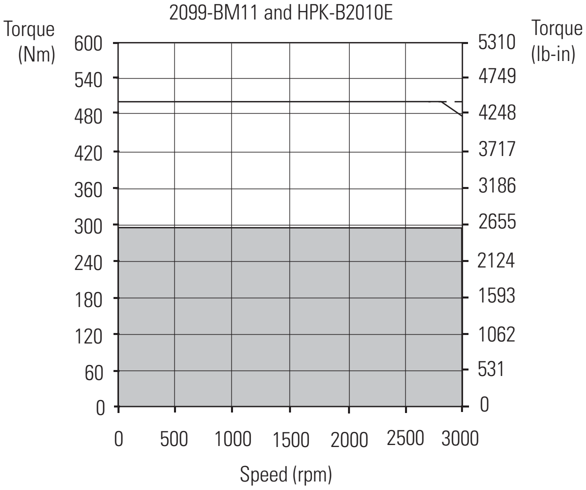




☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 750V DC (rms) input voltage

Kinetix 7000 (460V) Drives/HPK-Series (460V) Motor Curves, Continued


2099-BM11 and HPK-B1613E


2099-BM11 and HPK-B2010E

☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 750V DC (rms) input voltage

HPK-Series Motor Cable Combinations (400V Motors)

Motor Cat. No. (400V class)	Motor Power/Brake Cable	Motor Feedback Cable [1]
HPK-E1307C	Customer Supplied	2090-XXNFMF-S*xx* [2] Absolute High-resolution Feedback
HPK-E1307E, HPK-E1308E, and HPK-E1310C		
HPK-E1609E, HPK-E1611E, HPK-E1613C, and HPK-E1613E		
HPK-E1815C and HPK-E2010C		

(1) Use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

(2) These cables are available as standard (catalog number 2090-XXNFMF-S*xx*) or continuous-flex (catalog number 2090-CFBM4DF-CDAF*xx*).

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

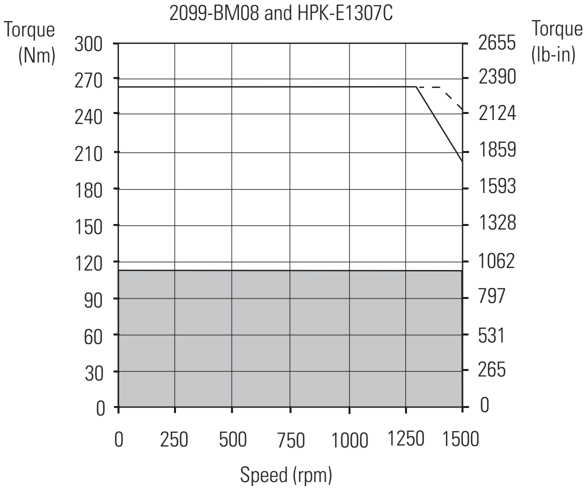
HPK-Series (400V) Performance Specifications with Kinetix 7000 (460V) Drives

Rotary Motor	**Base Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Stall Torque** N•m (lb•in)	**Motor Rated Output** kW (Hp)	**Kinetix 7000 Drives**
HPK-E1307C	1500	58.5	112 (991)	146.6	263 (2327)	17.1 (22.9)	2099-BM08
HPK-E1310C		80.0	155 (1372)	200.0	380 (3363)	23.8 (32.4)	2099-BM09
HPK-E1613C		133.0	271 (2398)	310.0	625 (5531)	41.7 (55.9)	2099-BM11
HPK-E1815C		187.0	360 (3186)	440.0	840 (7434)	55.9 (74.9)	2099-BM11
HPK-E2010C		243.0	482 (4266)	440.0	870 (7700)	75.0 (100.5)	2099-BM11
HPK-E1307E	3000	102.0	96.0 (849)	217.0	202 (1788)	29.8 (39.9)	2099-BM09
HPK-E1308E		112.8	107 (947)	217.7	200 (1770)	33.2 (45.0)	2099-BM09
HPK-E1609E		153.7	156 (1381)	356.7	359 (3176)	48.4 (64.9)	2099-BM11
HPK-E1611E		185.0	183 (1619)	440.0	430 (3805)	57.0 (76.4)	2099-BM11
HPK-E1613E		242.5	237 (2097)	440.0	430 (3805)	73.7 (98.8)	2099-BM11

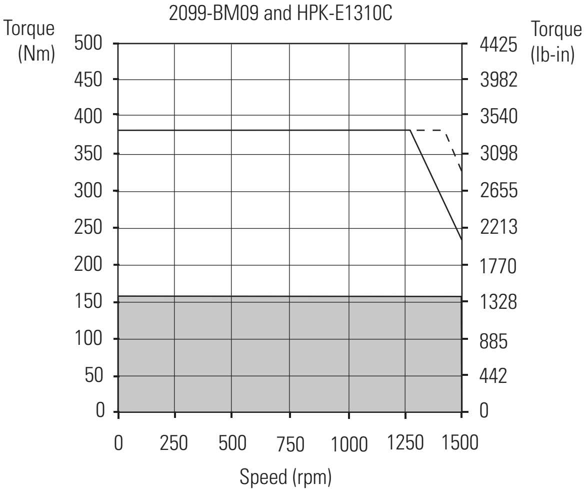
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 7000 (460V) Drives/HPK-Series (400V) Motor Curves

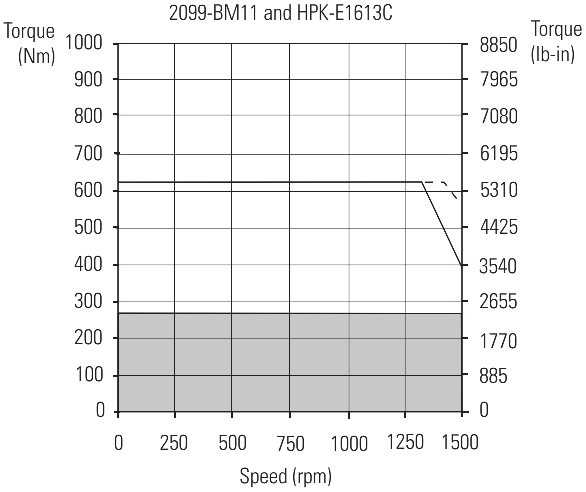
2099-BM08 and HPK-E1307C



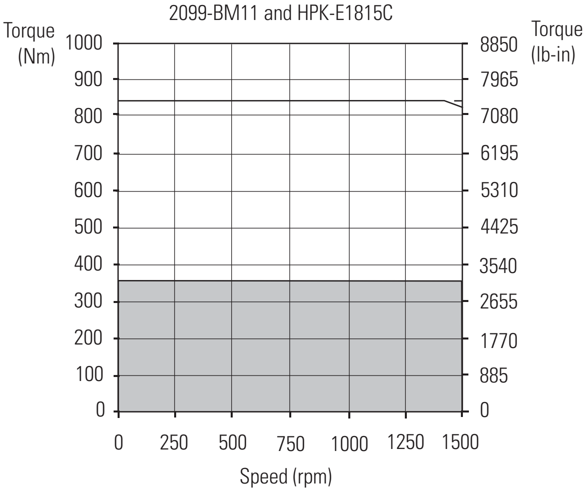
2099-BM09 and HPK-E1310C



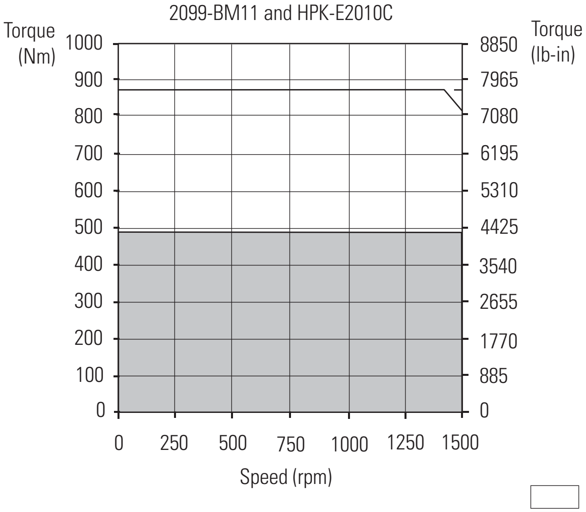
2099-BM11 and HPK-E1613C



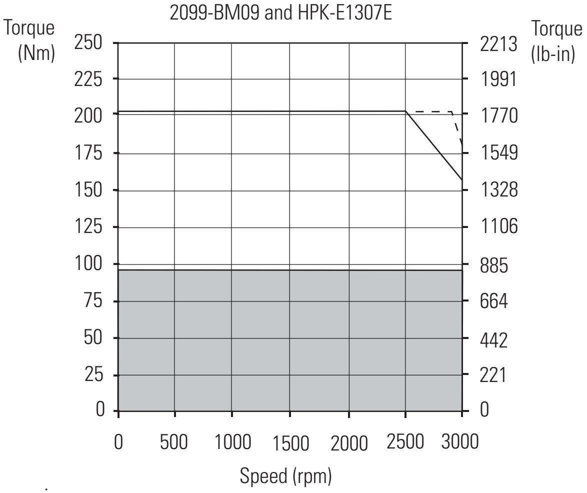
2099-BM11 and HPK-E1815C



2099-BM11 and HPK-E2010C

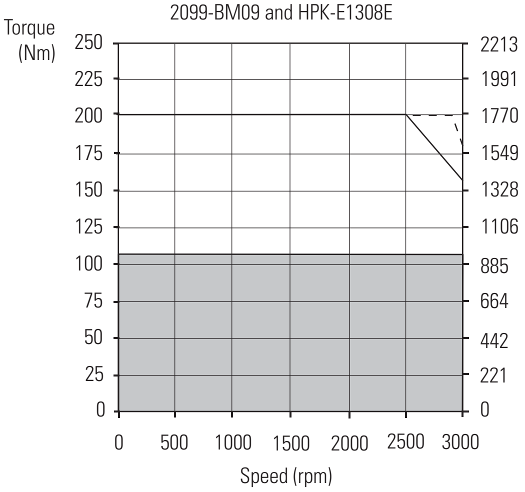


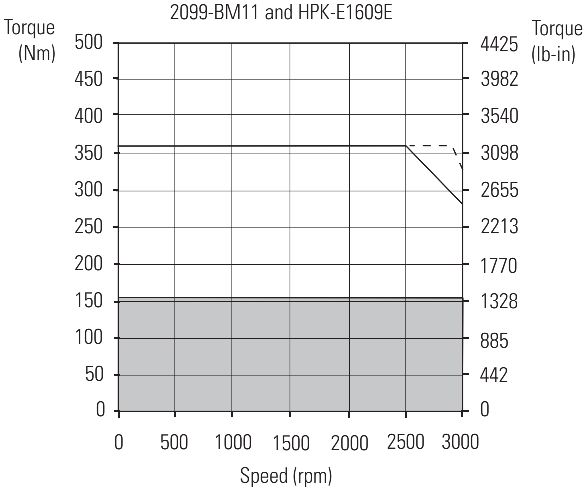
2099-BM09 and HPK-E1307E

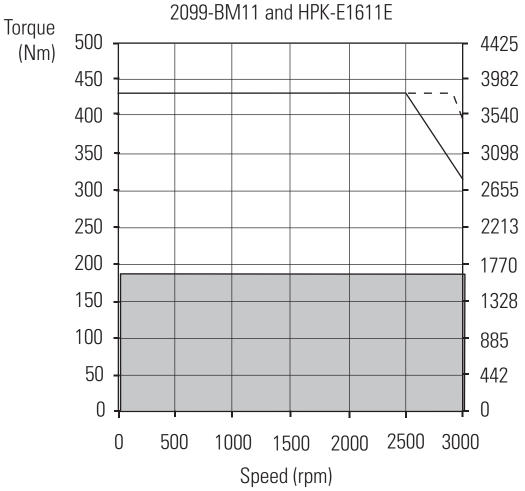


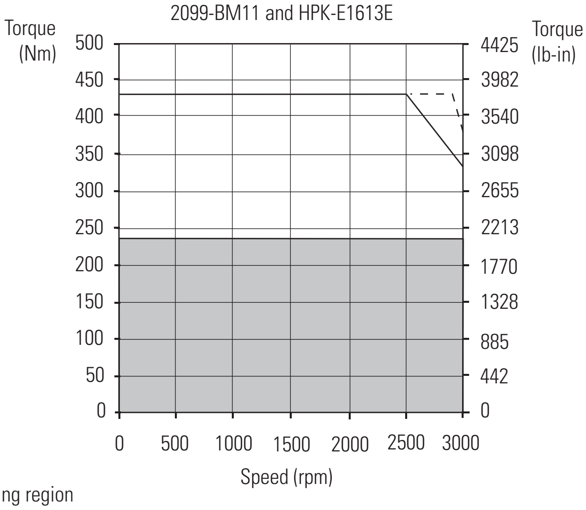
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 750V DC (rms) input voltage

Kinetix 7000 (460V) Drives/HPK-Series (400V) Motor Curves, Continued









☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 750V DC (rms) input voltage

Kinetix 7000 Drives with MP-Series Low Inertia Motors

This section provides system combination information for the Kinetix 7000 drives when matched with MP-Series low-inertia motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

IMPORTANT	The MP-Series low-inertia motors on this page are equipped with DIN connectors (specified by 7 in the catalog number) and are not compatible with cables designed for motors equipped with bayonet connectors (specified by 2 in the catalog number). The motors with bayonet connectors (for example, MPL-A310P-*xx*2*x*AA) are being discontinued and require 2090-XXN*x*MP (bayonet) cables. For help with migration or to select bayonet cables, contact your Rockwell Automation sales representative.

Bulletin MPL Motor Cable Combinations

Motor Cat. No.	Motor Power/Brake Cable	Motor Feedback Cable (1)
MPL-B540K-*xx*7*x*AA, MPL-B560F-*xx*7*x*AA	2090-CP*x*M7DF-14AA*xx* (standard) 2090-CP*x*M7DF-14AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback
MPL-B580F-*xx*7*x*AA, MPL-B580J-*xx*7*x*AA MPL-B640F-*xx*7*x*AA	2090-CP*x*M7DF-10AA*xx* (standard) 2090-CP*x*M7DF-10AF*xx* (continuous-flex)	
MPL-B660F-*xx*7*x*AA, MPL-B680D-*xx*7*x*AA, MPL-B960B-*xx*7*x*AA, MPL-B980B-*xx*7*x*AA	2090-CP*x*M7DF-08AA*xx* (standard) 2090-CP*x*M7DF-08AF*xx* (continuous-flex)	
MPL-B680F-*xx*7*x*AA, MPL-B860D-*xx*7*x*AA	2090-CPBM7DF-06AA*xx* (standard)	
MPL-B880C-*xx*7*x*AA		
MPL-B880D-*xx*7*x*AA	2090-CPBM7DF-04AA*xx* (standard)	
MPL-B960C-*xx*7*x*AA, MPL-B960D-*xx*7*x*AA, MPL-B980C-*xx*7*x*AA, MPL-B980D-*xx*7*x*AA		
MPL-B980E-*xx*7*x*AA	2090-CPBM7DF-02AA*xx* (standard)	

(1) Use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

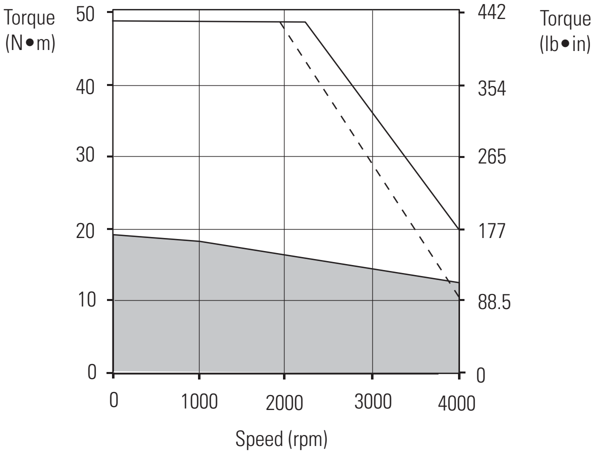
Bulletin MPL Motor Performance Specifications with Kinetix 7000 (460V) Drives

Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Stall Torque** N•m (lb•in)	**Motor Rated Output** kW	**Kinetix 7000 Drives**
MPL-B540K	3000	20.4	19.5 (172)	60.0	48.6 (430)	5.4	2099-BM06
MPL-B560F	3000	20.9	27.0 (239)	68.0	67.7 (599)	5.5	2099-BM06
MPL-B580F	3000	26.0	34.0 (301)	94.0	87.0 (770)	7.1	2099-BM06
MPL-B580J	3800	32.0	34.0 (301)	94.0	81.0 (717)	7.9	2099-BM06
MPL-B640F	3000	27.8	36.7 (325)	65.0	72.3 (640)	6.1	2099-BM06
MPL-B660F	3000	33.0	48.0 (425)	96.0	101 (894)	6.1	2099-BM06
MPL-B680D	2000	29.4	62.8 (556)	94.0	154 (1363)	9.3	2099-BM06
MPL-B680F	3000	41.5	59.4 (526)	96.0	108 (956)	7.5	2099-BM06
MPL-B860D	2000	40.9	83.1 (735)	95.5	152 (1345)	12.5	2099-BM06
MPL-B880C	1500	41.1	108 (956)	96.0	200 (1796)	12.6	2099-BM06
			109 (965)	97.5	203 (1770)		2099-BM07
MPL-B880D	2000	57.0	92.5 (818)	96.0	147 (1301)	12.6	2099-BM06
		58.0	110 (973)				2099-BM07
MPL-B960B	1200	36.8	130 (1150)	94.0	231 (2044)	12.7	2099-BM06
MPL-B960C	1500	47.6	124 (1097)	96.0	184 (1628)	14.8	2099-BM06
				113	209 (1850)		2099-BM07
				125	226 (2000)		2099-BM08
MPL-B960D	2000	57.0	100 (885)	96.0	171 (1513)	15.0	2099-BM06
		60.6	124 (1097)	113	201 (1779)		2099-BM07
				125	223 (1973)		2099-BM08
MPL-B980B	1000	34.6	162 (1434)	94.0	278 (2460)	15.2	2099-BM06
MPL-B980C	1500	57.0	131 (1159)	96.0	198 (1752)	16.8	2099-BM06
		59.0	158 (1398)	113	227 (2009)		2099-BM07
				140	270 (2389)		2099-BM08
MPL-B980D	2000	57.0	113 (1000)	96.0	183 (1619)	18.6	2099-BM06
		68.4	148 (1310)	113	213 (1885)		2099-BM07
			158 (1398)	140	259 (2292)		2099-BM08
MPL-B980E	2750	105	141 (1250)	218	237 (2100)	13.0	2099-BM09

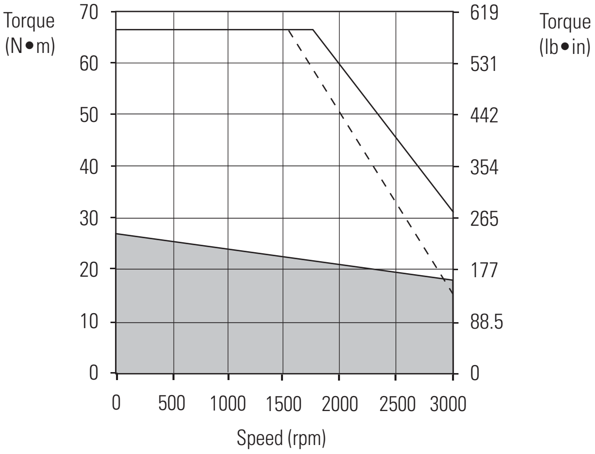
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 7000 (460V) Drives/MP-Series Low Inertia Motor Curves

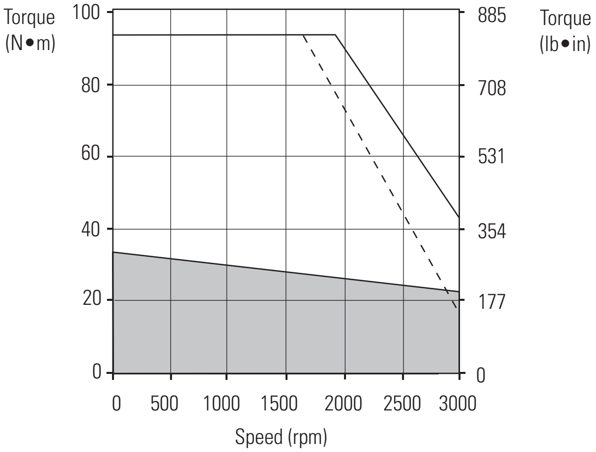
2099-BM06 and MPL-B540K



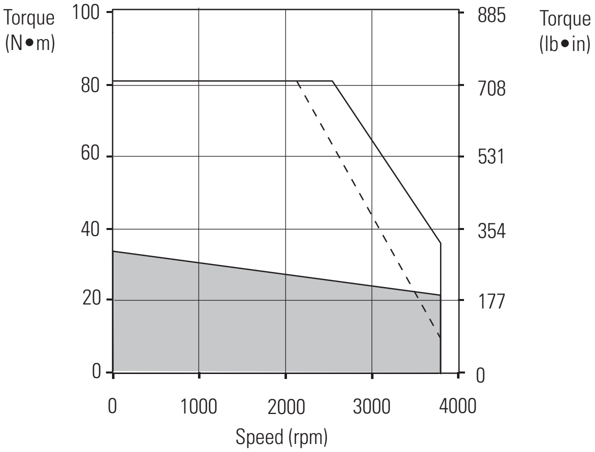
2099-BM06 and MPL-B560F



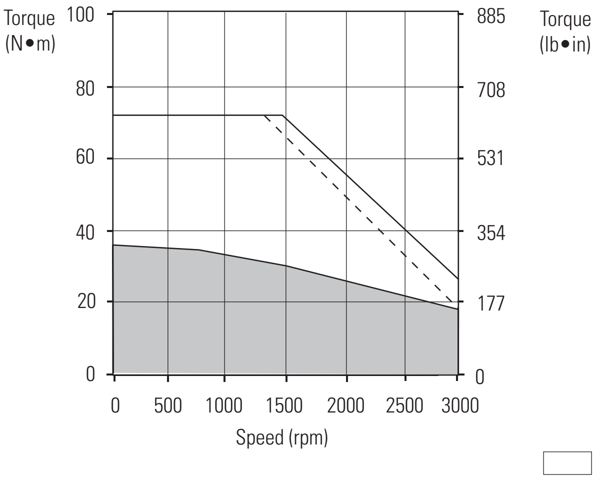
2099-BM06 and MPL-B580F



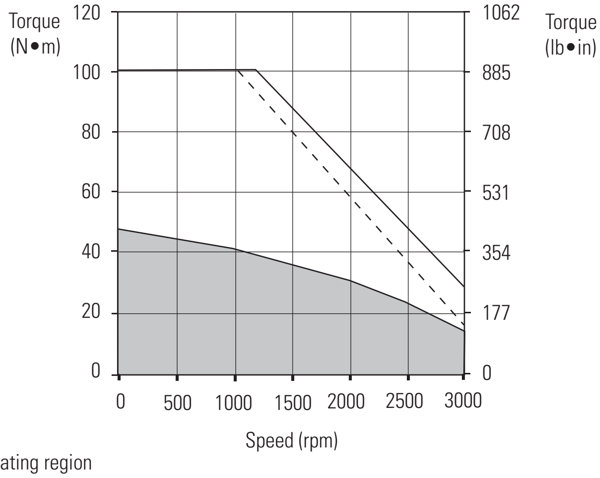
2099-BM06 and MPL-B580J



2099-BM06 and MPL-B640F



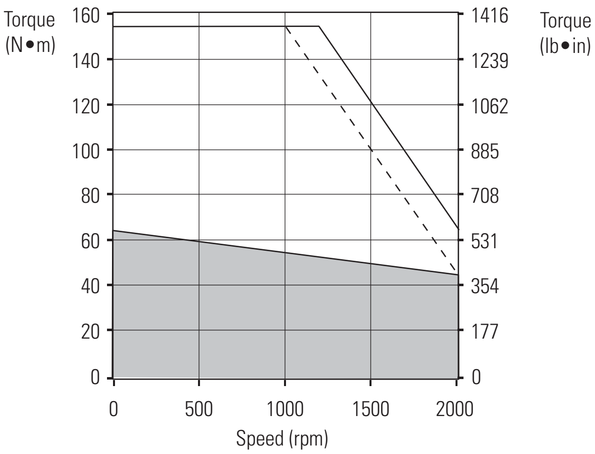
2099-BM06 and MPL-B660F



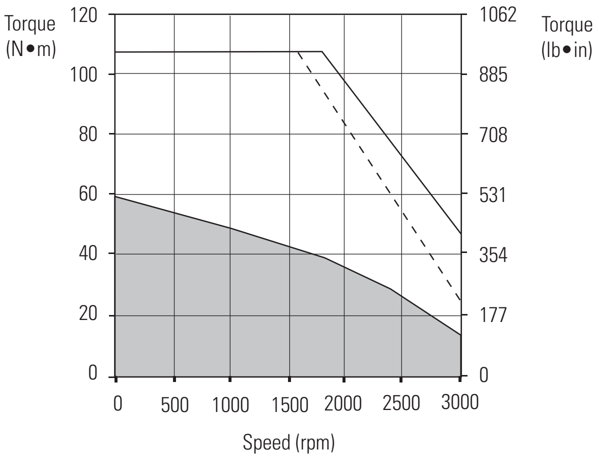
☐ = Intermittent operating region
▨ = Continuous operating region
– – – = Drive operation with 400V AC (rms) input voltage

Kinetix 7000 (460V) Drives/MP-Series Low Inertia Motor Curves, Continued

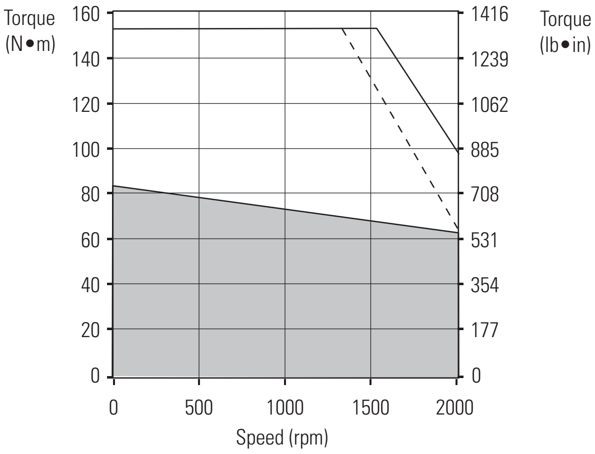
2099-BM06 and MPL-B680D



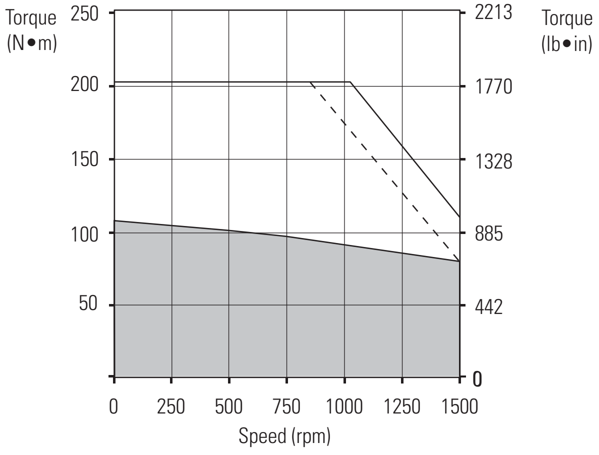
2099-BM06 and MPL-B680F



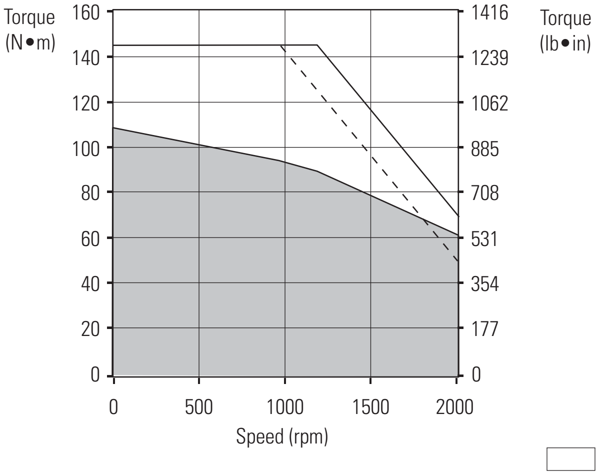
2099-BM06 and MPL-B860D



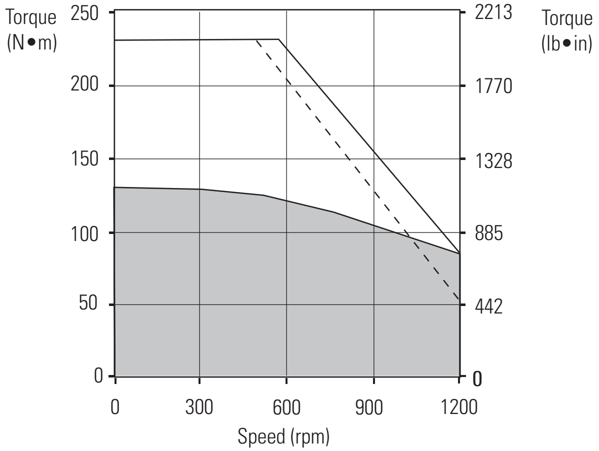
2099-BM07 and MPL-B880C



2099-BM07 and MPL-B880D



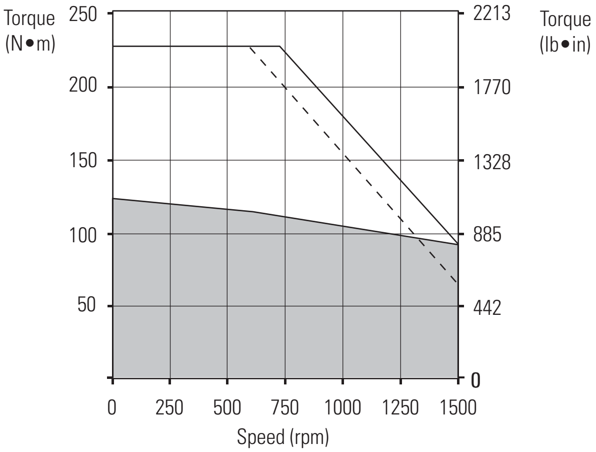
2099-BM06 and MPL-B960B



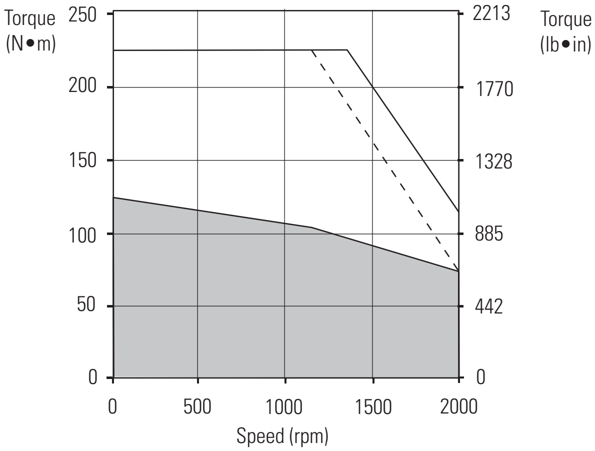
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Kinetix 7000 (460V) Drives/MP-Series Low Inertia Motor Curves, Continued

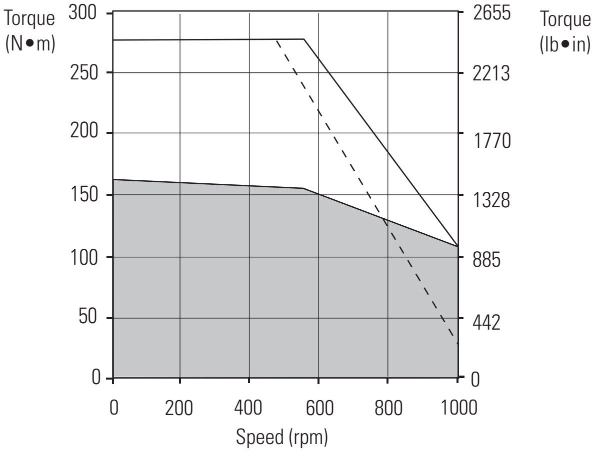
2099-BM08 and MPL-B960C



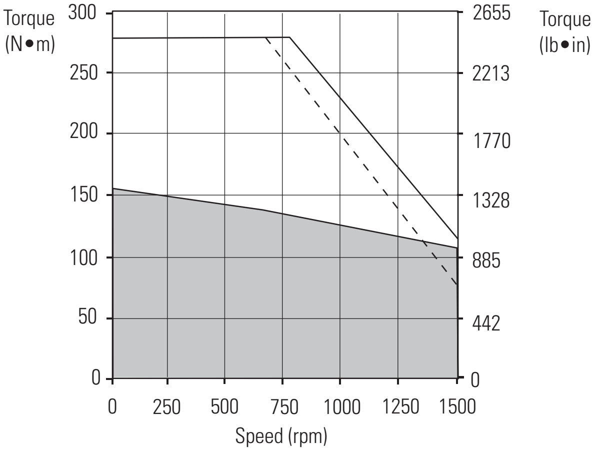
2099-BM08 and MPL-B960D



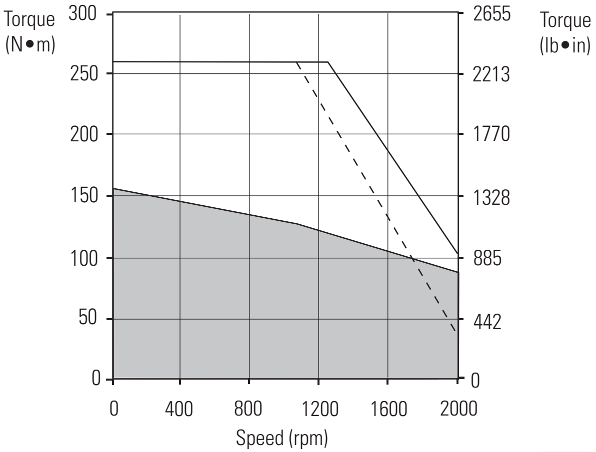
2099-BM08 and MPL-B980B



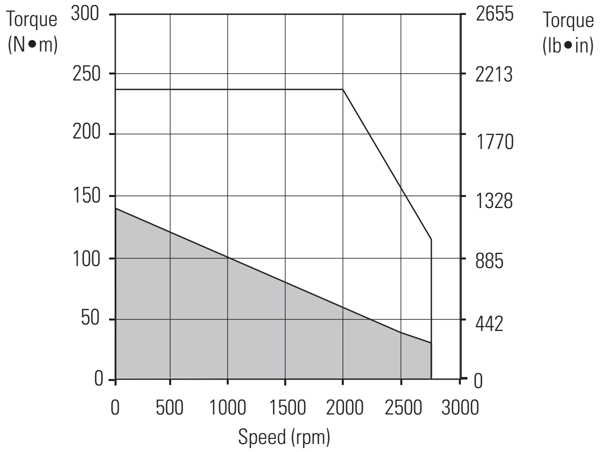
2099-BM08 and MPL-B980C



2099-BM08 and MPL-B980D



2099-BM09 and MPL-B980E



☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Kinetix 7000 (460V) Drives with MP-Series Medium Inertia Motors

This section provides system combination information for the Kinetix 7000 (460V) drives when matched with MP-Series medium-inertia motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin MPM Motor Cable Combinations

Motor Cat. No. (400V class)	Motor Power/Brake Cable	Motor Feedback Cable (1)
MPM-B1151*x*, MPM-B1152*x*, MPM-B1153E, MPM-B1153F	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback
MPM-B1302F, MPM-B1302M, MPM-B1304C, MPM-B1304E		
MPM-B1651C, MPM-B1652C		
MPM-B1153T	2090-CP*x*M7DF-14AA*xx* (standard) 2090-CP*x*M7DF-14AF*xx* (continuous-flex)	
MPM-B1302T, MPM-B1304M		
MPM-B1651F, MPM-B1653C		
MPM-B1651M, MPM-B1652E, MPM-B1652F, MPM-B1653E	2090-CP*x*M7DF-10AA*xx* (standard) 2090-CP*x*M7DF-10AF*xx* (continuous-flex)	
MPM-B2152C, MPM-B2153B		
MPM-B1653F	2090-CP*x*M7DF-08AA*xx* (standard) 2090-CP*x*M7DF-08AF*xx* (continuous-flex)	
MPM-B2152F, MPM-B2152M, MPM-B2153E, MPM-B2153F, MPM-B2154B, MPM-B2154E, MPM-B2154F		

(1) Use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

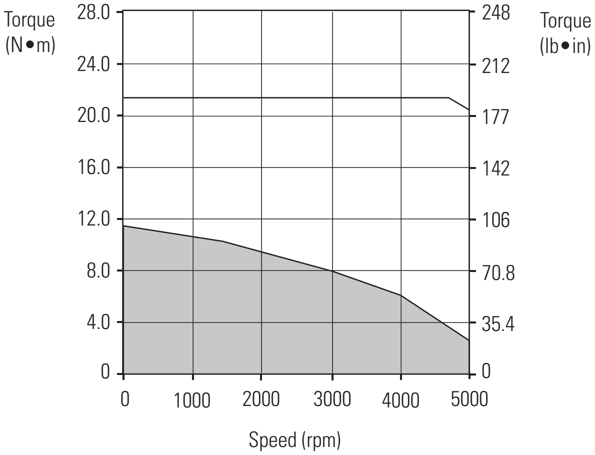
Bulletin MPM Motor Performance Specifications with Kinetix 7000 (460V) Drives

Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Stall Torque** N•m (lb•in)	**Motor Rated Output** kW	**Kinetix 7000 Drives**
MPM-B1651M	5000	25.8	11.3 (100)	56.8	21.4 (189)	2.50	2099-BM06-S
MPM-B1652E	3500	24.0	21.1 (187)	60.5	48.0 (425)	4.30	2099-BM06-S
MPM-B1652F	4500	33.0	21.1 (187)	84.1	45.0 (398)	4.30	2099-BM06-S
MPM-B1653C	2500	23.0	26.7 (236)	59.2	67.7 (599)	4.60	2099-BM06-S
MPM-B1653E	3500	31.0	26.8 (237)	72.9	62.0 (549)	5.10	2099-BM06-S
MPM-B1653F	4000	40.1	31.0 (274)	94.3	56.1 (496)	5.10	2099-BM06-S
MPM-B2152C	2500	31.5	36.7 (325)	55.4	72.2 (639)	5.60	2099-BM06-S
MPM-B2152F	4500	50.0	34.1 (302)	96.0	70.8 (626)	5.90	2099-BM06-S
				98.0	72.2 (639)		2099-BM07-S
MPM-B2152M	5000	51.2	34.1 (302)	76.3	52.9 (468)	5.90	2099-BM06-S
MPM-B2153B	2000	27.6	48.0 (425)	60.0	101 (895)	6.80	2099-BM06-S
MPM-B2153E	3000	45.5	47.9 (424)	96.0	98.8 (468)	7.20	2099-BM06-S
				98.6	101 (895)		2099-BM07-S
MPM-B2153F	3800	50.4	45.6 (403)	96.0	96.6 (855)	7.20	2099-BM06-S
				98.4	98.9 (875)		2099-BM07-S
MPM-B2154B	2000	40.7	62.7 (555)	96.0	151 (1336)	6.90	2099-BM06-S
				98.0	154 (1363)		2099-BM07-S
MPM-B2154E	3000	50.2	55.9 (495)	96.0	110 (973)	7.50	2099-BM06-S
				98.3	112 (991)		2099-BM07-S
MPM-B2154F	3300	51.0	56.2 (497)	83.6	87.9 (778)	7.50	2099-BM06-S

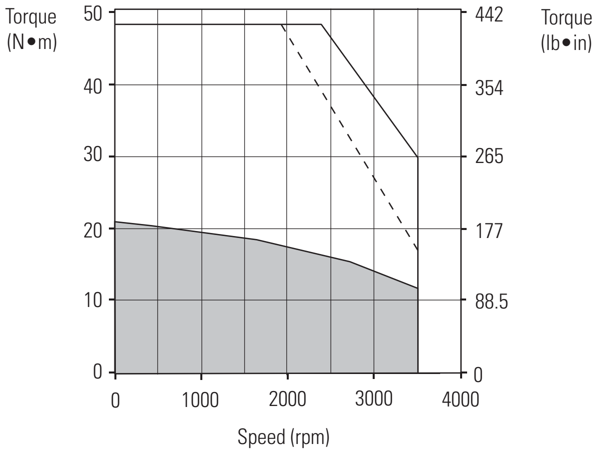
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 7000 (460V) Drives/MP-Series Medium Inertia Motor Curves

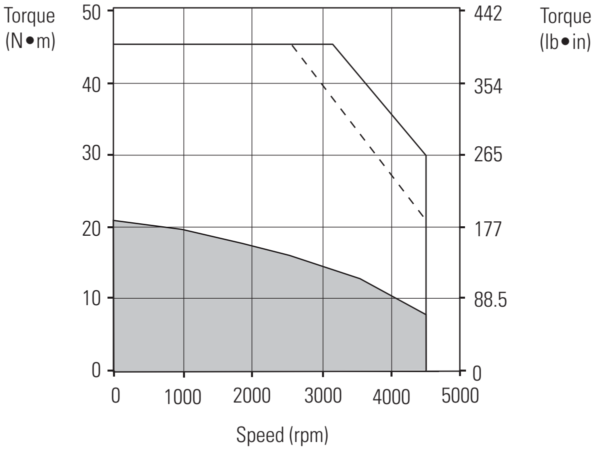
2099-BM06-S and MPM-B1651M



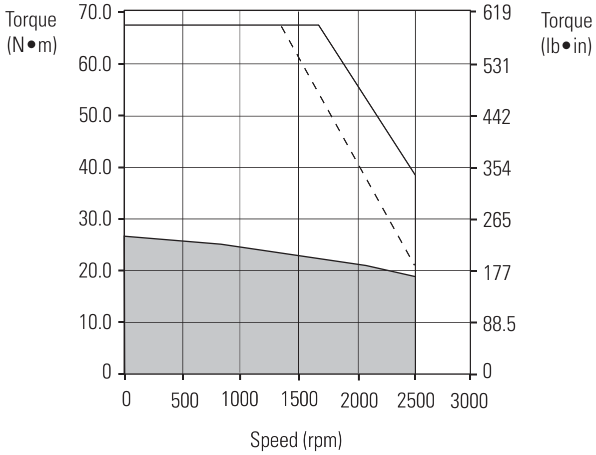
2099-BM06-S and MPM-B1652E



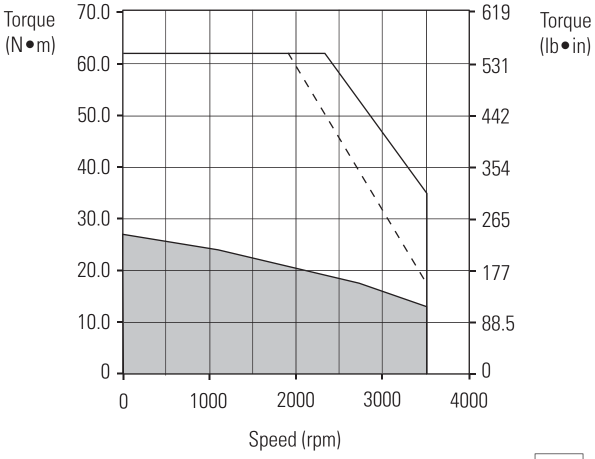
2099-BM06-S and MPM-B1652F



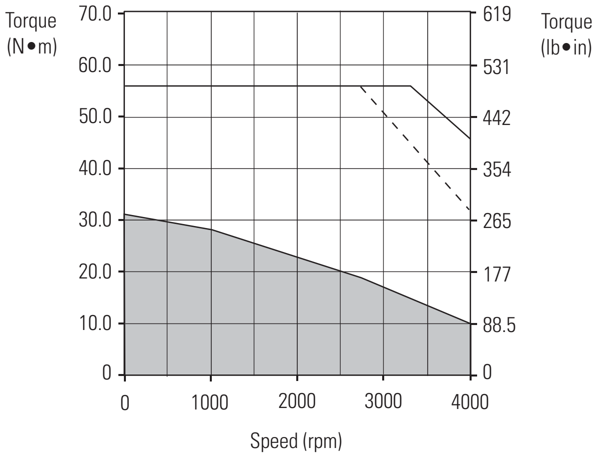
2099-BM06-S and MPM-B1653C



2099-BM06-S and MPM-B1653E

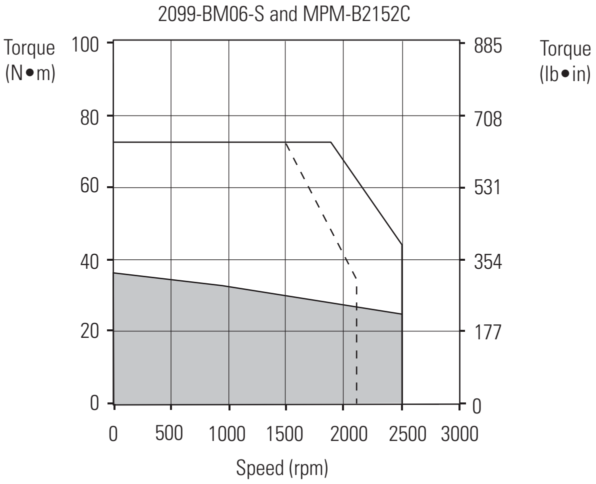


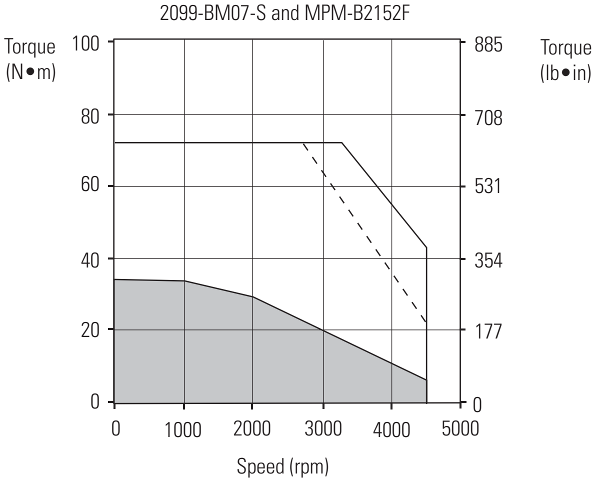
2099-BM06-S and MPM-B1653F

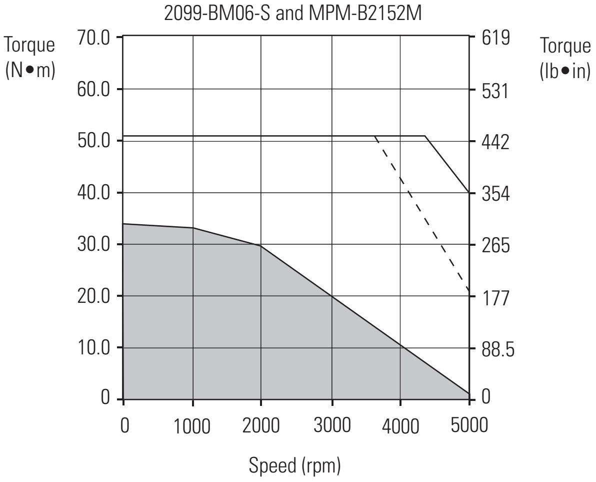


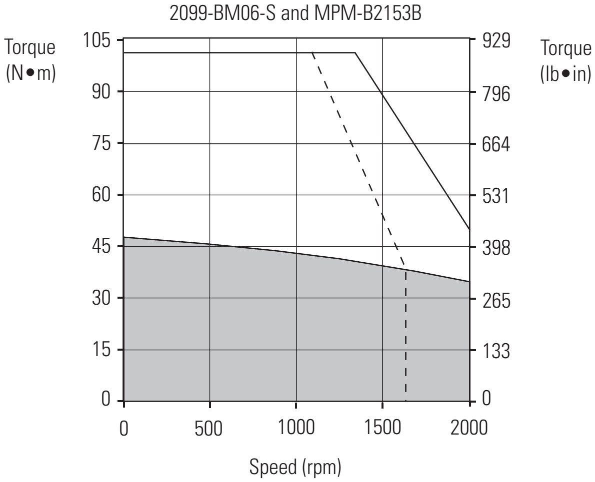
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

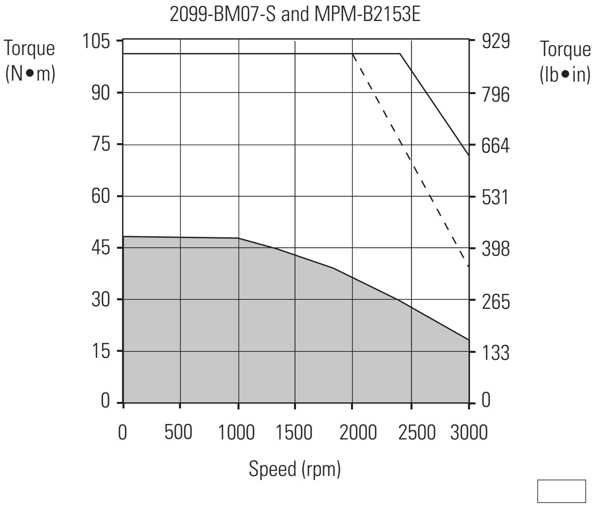
Kinetix 7000 (460V) Drives/MP-Series Medium Inertia Motor Curves, Continued

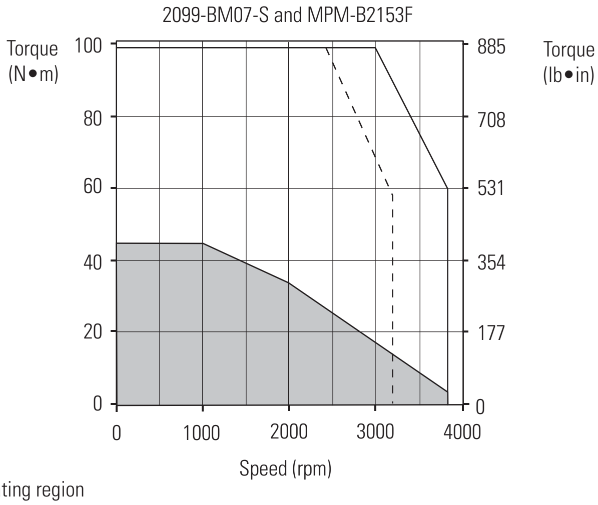






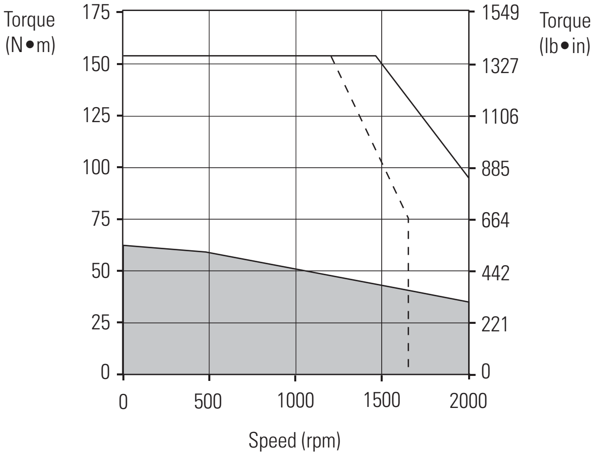


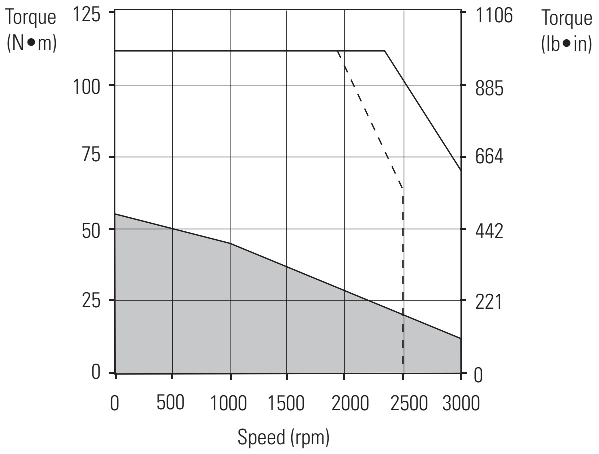


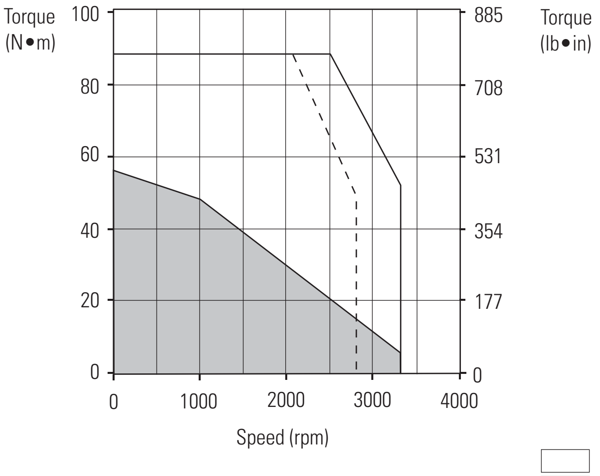


☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Kinetix 7000 (460V) Drives/MP-Series Medium Inertia Motor Curves, Continued


2099-BM07-S and MPM-B2154B


2099-BM07-S and MPM-B2154E


2099-BM06-S and MPM-B2154F

☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Kinetix 7000 Drives with RDD-Series Direct Drive Motors

This section provides system combination information for the Kinetix 7000 (460V) drives when matched with RDD-Series direct-drive motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin RDB Motor Cable Combinations

Motor Cat. No. (400V class)	Motor Power/Brake Cable	Motor Feedback Cable [1]
RDB-B21519, RDB-B21529	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-XXNFMF-S*xx* [2] Absolute High-resolution Feedback
RDB-B29014, RDB-B29016, RDB-B29024		
RDB-B2151C, RDB-B21539	2090-CP*x*M7DF-14AA*xx* (standard) 2090-CP*x*M7DF-14AF*xx* (continuous-flex)	
RDB-B29019, RDB-B29034		
RDB-B2152C	2090-CP*x*M7DF-12AA*xx* (standard)	
RDB-B29026		
RDB-B2151F, RDB-B2153C	2090-CP*x*M7DF-10AA*xx* (standard) 2090-CP*x*M7DF-10AF*xx* (continuous-flex)	
RDB-B29036, RDB-B41014		
RDB-B2152F, RDB-B2153E	2090-CP*x*M7DF-08AA*xx* (standard) 2090-CP*x*M7DF-08AF*xx* (continuous-flex)	
RDB-B29029, RDB-B41016, RDB-B41024		
RDB-B29039, RDB-B41018, RDB-B41026, RDB-B41035	2090-CPBM7DF-06AA*xx* (standard)	

(1) Use low-profile feedback module (catalog number 2090-K7CK-KENDAT). Refer to Breakout Components and Connector Kits beginning on page 440.

(2) These cables are available as standard (catalog number 2090-XXNFMF-S*xx*) or continuous-flex (catalog number 2090-CFBM7DF-CDAF*xx*).

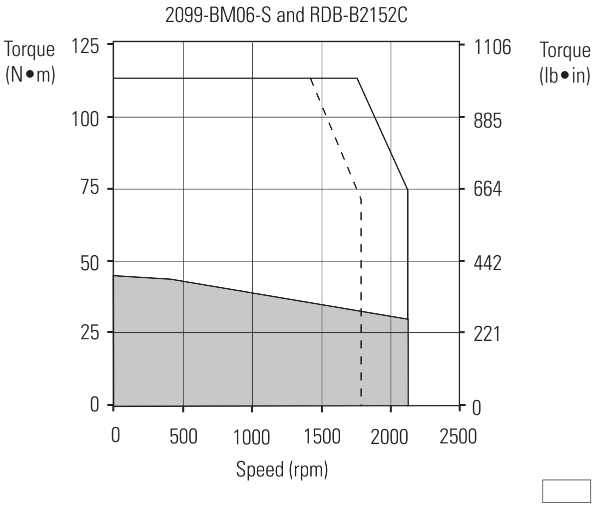
Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

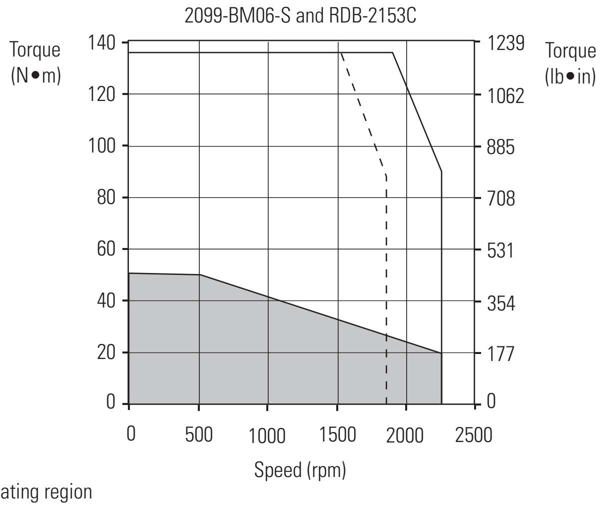
Bulletin RDB Motor Performance Specifications with Kinetix 7000 (460V) Drives

Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 7000 460V Drives
RDB-B2152C	2125	23.5	43.4 (384)	63.2	111 (982)	6.41	2099-BM06-S
RDB-B2153C	2250	29.4	51.5 (456)	82.6	137 (1212)	5.87	2099-BM06-S
RDB-B29029	1200	40.0	97.5 (863)	96.0	193 (1708)	4.05	2099-BM06-S
				111	214 (1894)		2099-BM07-S
RDB-B29036	750	27.0	140 (1239)	84.9	318 (2814)	5.49	2099-BM06-S
RDB-B29039	1000	52.7	122 (1080)	147	268 (2372)	4.41	2099-BM08-S
RDB-B41016	700	33.8	183 (1619)	95.5	339 (3000)	4.83	2099-BM06-S
RDB-B41018	700	51.3	183 (1619)	113	299 (2646)	4.83	2099-BM07-S
				140	339 (3000)		2099-BM08-S
RDB-B41024	365	31.5	330 (2929)	95.5	690 (6107)	7.29	2099-BM06-S
RDB-B41026	600	52.0	308 (2726)	147	626 (5540)	7.29	2099-BM08-S
				218	688 (6089)		2099-BM09-S
RDB-B41035	490	52.6	425 (3761)	147	897 (7939)	8.69	2099-BM08-S
				218	1050 (9293)		2099-BM09-S

Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 7000 (460V) Drives with RDD-Series Direct Drive Motor Curves

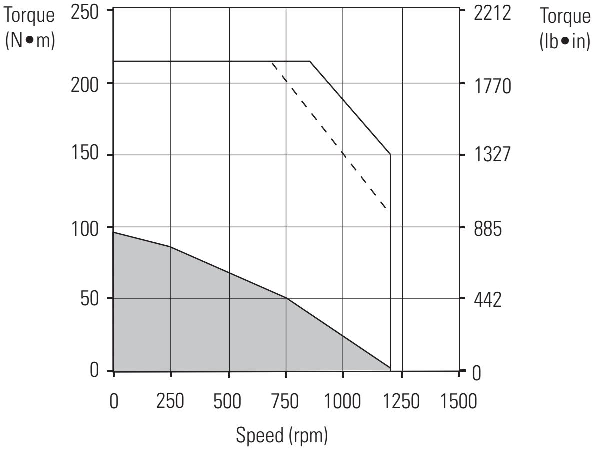




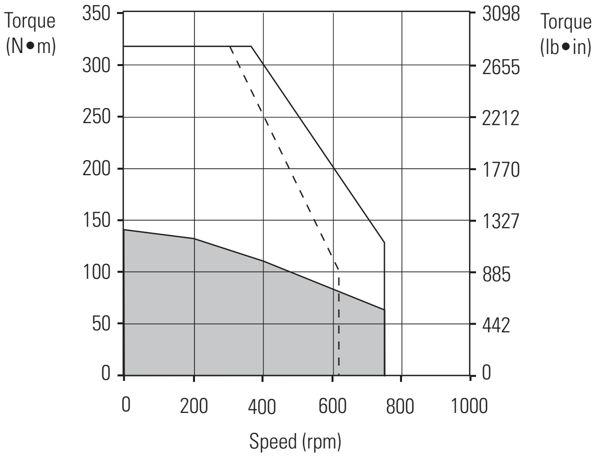
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Kinetix 7000 (460V) Drives with RDD-Series Direct Drive Motor Curves, Continued

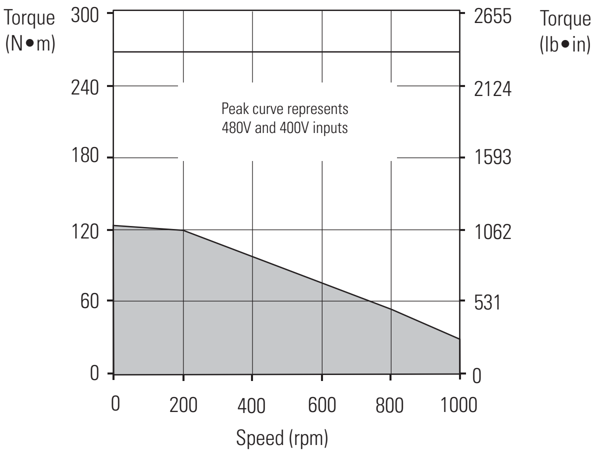
2099-BM07-S and RDB-B29029



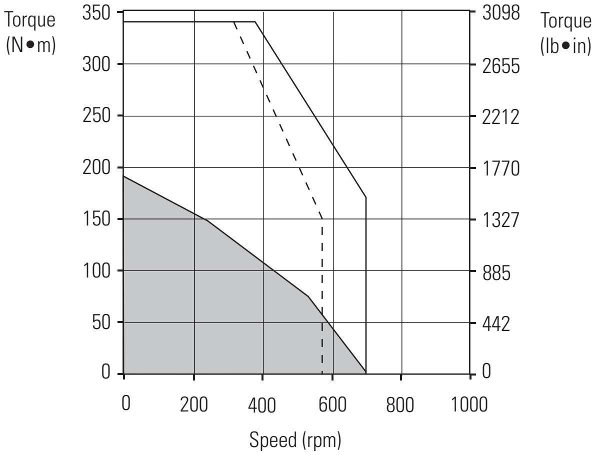
2099-BM06-S and RDB-B29036



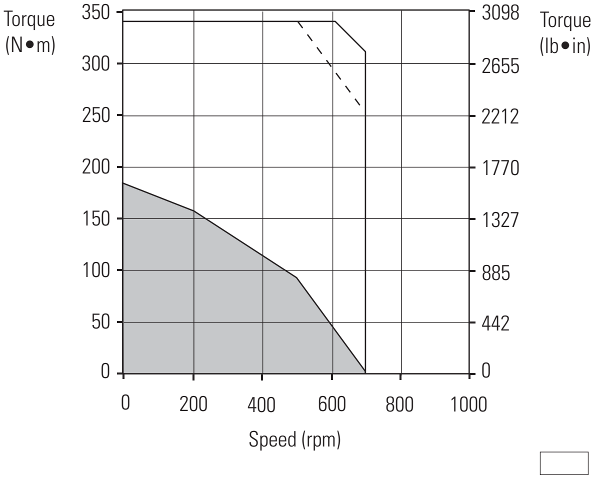
2099-BM08-S and RDB-B29039



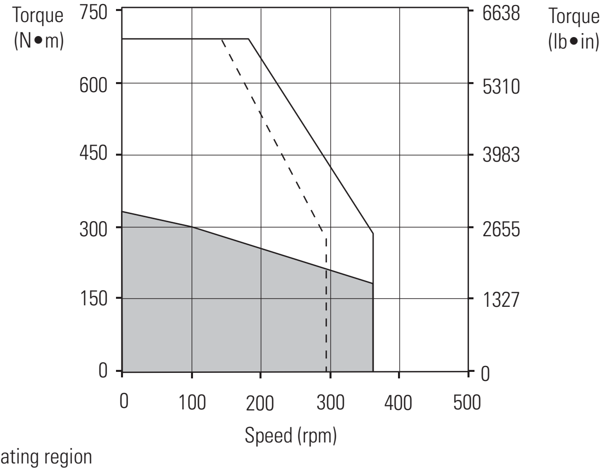
2099-BM06-S and RDB-B41016



2099-BM08-S and RDB-B41018



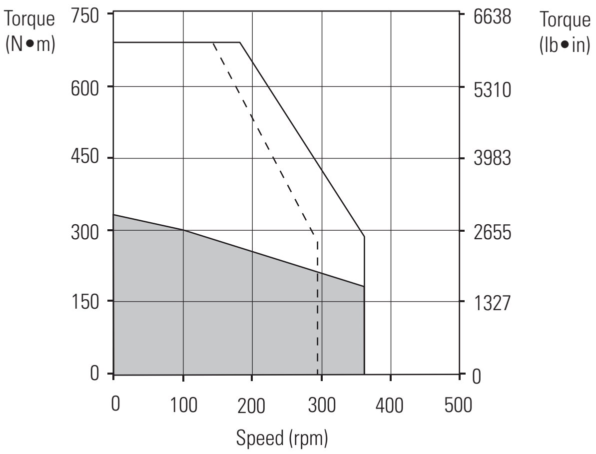
2099-BM06-S and RDB-B41024



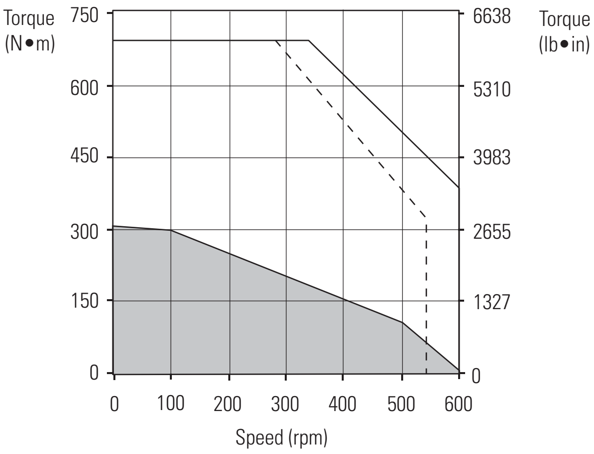
☐ = Intermittent operating region
▨ = Continuous operating region
– – – = Drive operation with 400V AC (rms) input voltage

Kinetix 7000 (460V) Drives with RDD-Series Direct Drive Motor Curves, Continued

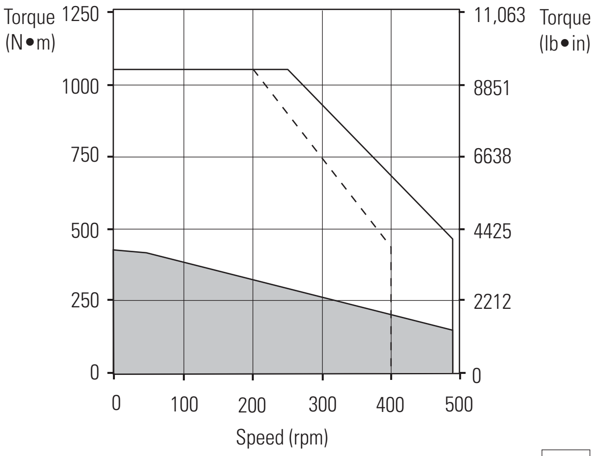
2099-BM06-S and RDB-B41024



2099-BM09-S and RDB-B41026



2099-BM09-S and RDB-B41035



☐ = Intermittent operating region
▨ = Continuous operating region
– – – = Drive operation with 400V AC (rms) input voltage

Kinetix 300 (240V) Drives with MP-Series Low Inertia Motors

This section provides system combination information for the Kinetix 300 (240V) drives when matched with MP-Series low-inertia motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

IMPORTANT The MP-Series low-inertia motors on this page are equipped with DIN connectors (specified by 4 or 7 in the catalog number) and are not compatible with cables designed for motors equipped with bayonet connectors (specified by 2 in the catalog number). The motors with bayonet connectors (for example, MPL-A310P-*xx*2*x*AA) are being discontinued and require 2090-XXN*x*MP (bayonet) cables. For help with migration or to select bayonet cables, contact your Rockwell Automation sales representative.

Bulletin MPL Motor Cable Combinations

Motor Cat. No. (200V class)	Motor Power/Brake Cable	Motor Feedback Cable (1)
MPL-A1510V-*xx*4*x*AA, MPL-A1520U-*xx*4*x*AA, MPL-A1530U-*xx*4*x*AA	2090-XXNPMF-16S*xx* (standard) 2090-CP*x*M4DF-16AF*xx* (continuous-flex)	2090-XXNFMF-S*xx* (standard) 2090-CFBM4DF-CDAF*xx* (continuous-flex) Absolute High-resolution and Incremental Feedback
MPL-A210V-*xx*4*x*AA, MPL-A220T-*xx*4*x*AA, MPL-A230P-*xx*4*x*AA		
MPL-A310F-*xx*7*x*AA, MPL-A310P-*xx*7*x*AA, MPL-A320H-*xx*7*x*AA, MPL-A320P-*xx*7*x*AA, MPL-A330P-*xx*7*x*AA	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or (2) 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback 2090-XXNFMF-S*xx* (standard) (3) 2090-CFBM7DF-CDAF*xx* (continuous-flex) Incremental Feedback
MPL-A420P-*xx*7*x*AA, MPL-A430H-*xx*7*x*AA		
MPL-A4530F-*xx*7*x*AA, MPL-A4540C-*xx*7*x*AA		
MPL-A430P-*xx*7*x*AA	2090-CP*x*M7DF-14AA*xx* (standard) 2090-CP*x*M7DF-14AF*xx* (continuous-flex)	
MPL-A4530K-*xx*7*x*AA, MPL-A4540F-*xx*7*x*AA		

(1) Use low-profile connector kit (catalog number 2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

(2) Applies to Kinetix 300 drives and MPL-A3*xxx*-M/S…MPL-A45*xxx*-M/S motors with absolute high-resolution feedback.

(3) Applies to Kinetix 300 drives and MPL-A3*xxx*-H…MPL-A45*xxx*-H motors with incremental feedback.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

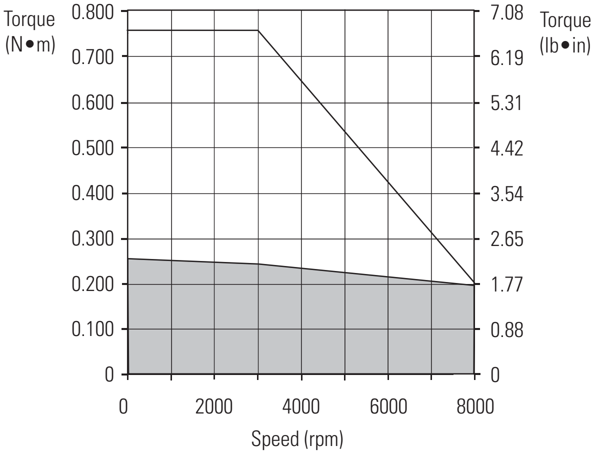
Bulletin MPL Motor Performance Specifications with Kinetix 300 (240V) Drives

Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Stall Torque** N•m (lb•in)	**Motor Rated Output** kW	**Kinetix 300 240V Drives**
MPL-A1510V	8000	1.05	0.26 (2.3)	3.40	0.77 (6.8)	0.16	2097-V33PR1
MPL-A1520U	7000	1.80	0.49 (4.3)	6.10	1.58 (13.9)	0.27	2097-V33PR1
MPL-A1530U	7000	2.82	0.90 (8.0)	10.1	2.82 (24.9)	0.39	2097-V33PR3
MPL-A210V	8000	3.09	0.55 (4.8)	10.2	1.52 (13.5)	0.37	2097-V33PR3
MPL-A220T	6000	4.54	1.61 (14.2)	15.5	4.74 (41.9)	0.62	2097-V33PR3
MPL-A230P	5000	5.40	2.10 (18)	23.0	8.2 (72.5)	0.86	2097-V33PR5
MPL-A310F	3000	3.20	1.58 (14)	9.3	3.61 (32)	0.46	2097-V33PR3
MPL-A310P	5000	4.85	1.58 (14)	14	3.61 (32)	0.73	2097-V33PR3
MPL-A320H	3500	6.1	3.05 (27)	19.3	7.91 (70)	1.0	2097-V33PR5
MPL-A320P	5000	9.0	3.05 (27)	29.5	7.91 (70)	1.3	2097-V33PR5
MPL-A330P	5000	12.0	4.18 (37)	38	11.1 (98)	1.8	2097-V33PR6
MPL-A420P	5000	12.7	4.74 (42)	46	13.5 (120)	2.0	2097-V33PR6
MPL-A430H	3500	12.2	6.21 (55)	45	19.8 (175)	1.8	2097-V33PR6
MPL-A430P	5000	16.8	5.99 (53)	51	15.7 (139)	2.2	2097-V33PR6
MPL-A4530F	2800	13.4	8.36 (74)	42	20.3 (180)	1.9	2097-V33PR6
MPL-A4540C	1500	9.4	10.2 (90)	29	27.1 (240)	1.5	2097-V33PR6

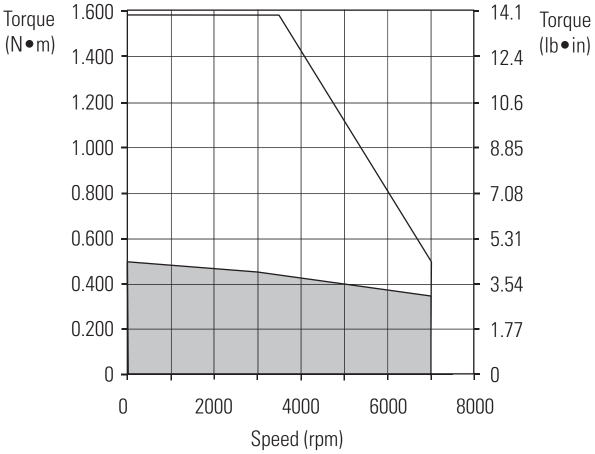
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 40 °C (104 °F) ambient, and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 300 (240V) Drives/MP-Series Low Inertia Motor Curves

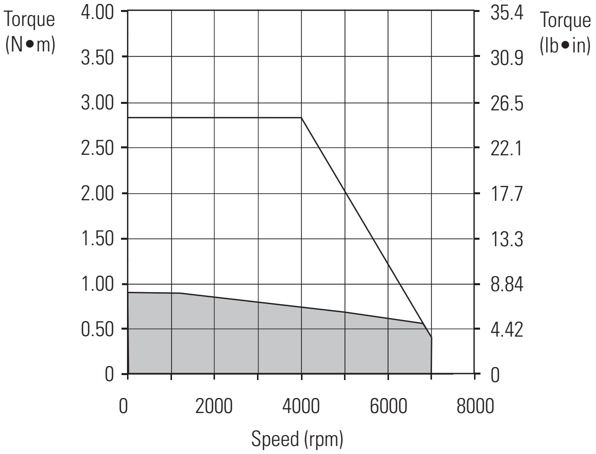
2097-V33PR1 and MPL-A1510V



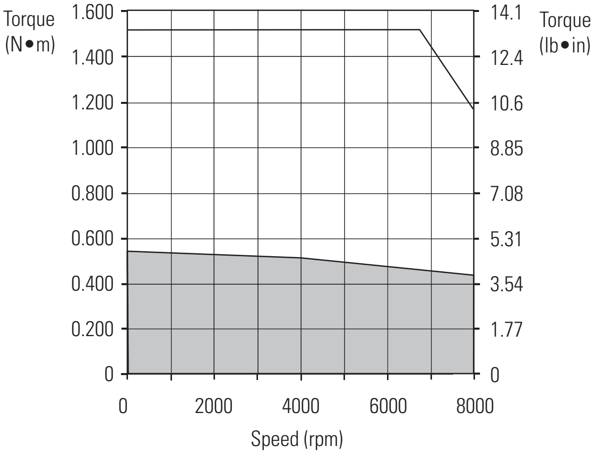
2097-V33PR1 and MPL-A1520U



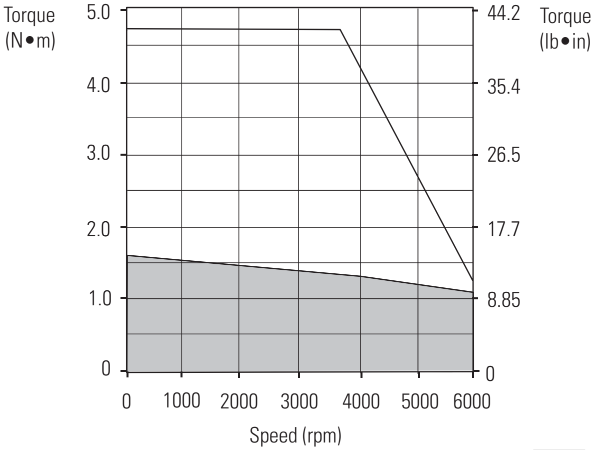
2097-V33PR3 and MPL-A1530U



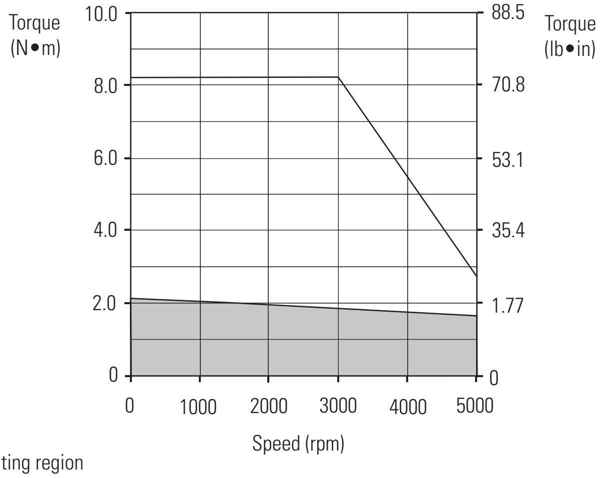
2097-V33PR3 and MPL-A210V



2097-V33PR3 and MPL-A220T

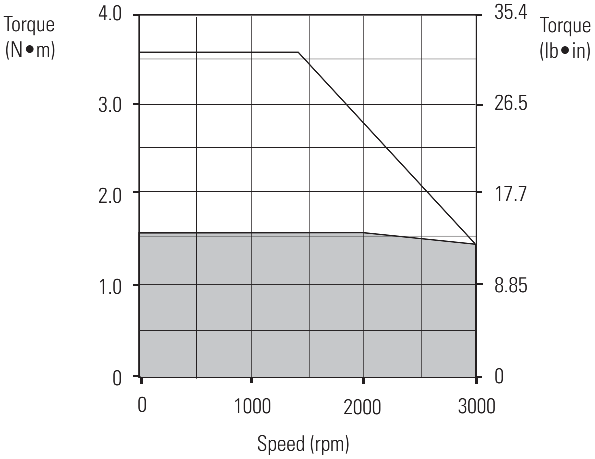


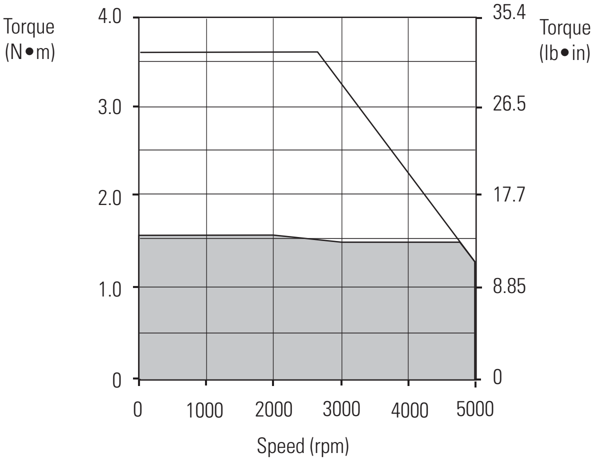
2097-V33PR5 and MPL-A230P

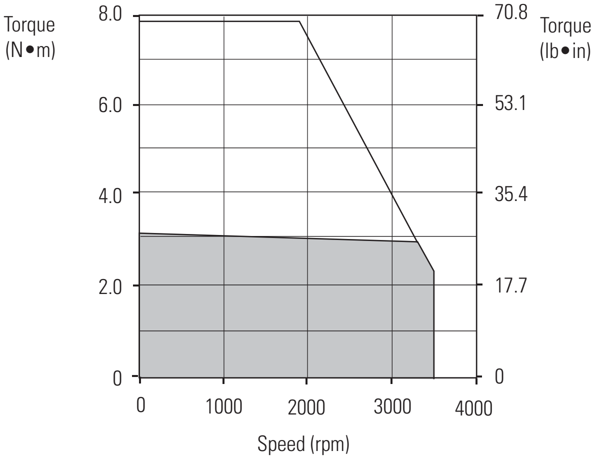


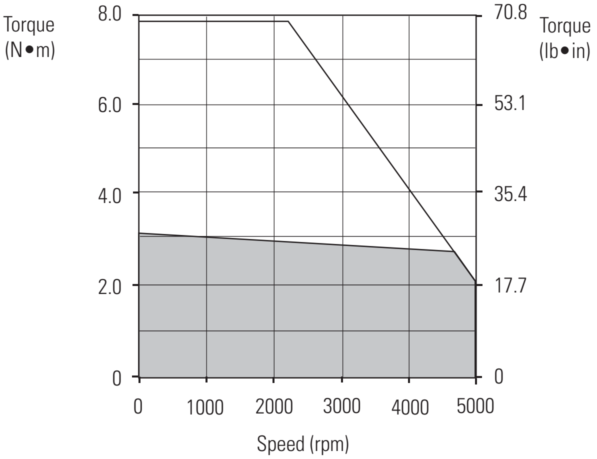
☐ = Intermittent operating region
▨ = Continuous operating region

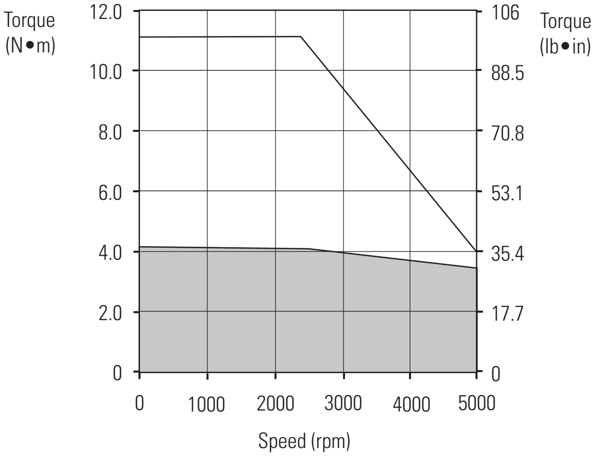
Kinetix 300 (240V) Drives/MP-Series Low Inertia Motor Curves, Continued

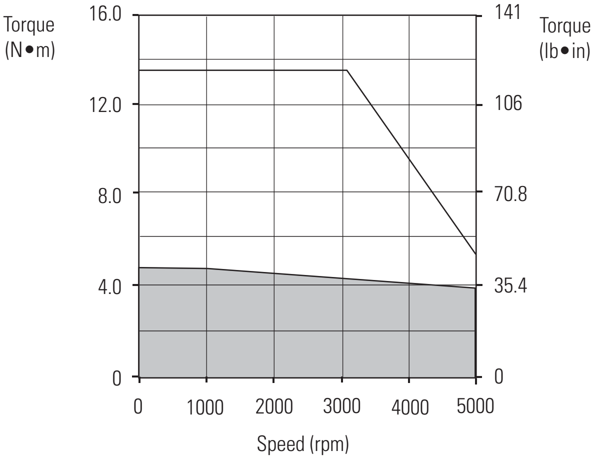

2097-V33PR3 and MPL-A310F


2097-V33PR3 and MPL-A310P


2097-V33PR5 and MPL-A320H


2097-V33PR5 and MPL-A320P

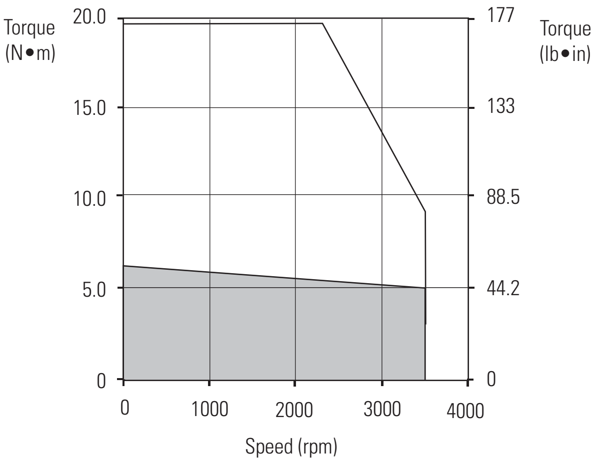

2097-V33PR6 and MPL-A330P


2097-V33PR6 and MPL-A420P

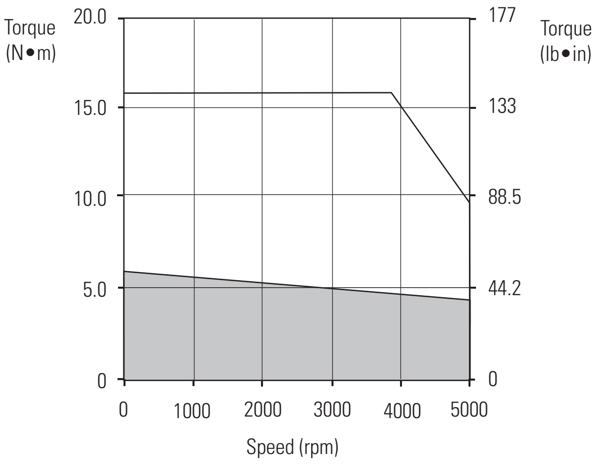
☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 300 (240V) Drives/MP-Series Low Inertia Motor Curves, Continued

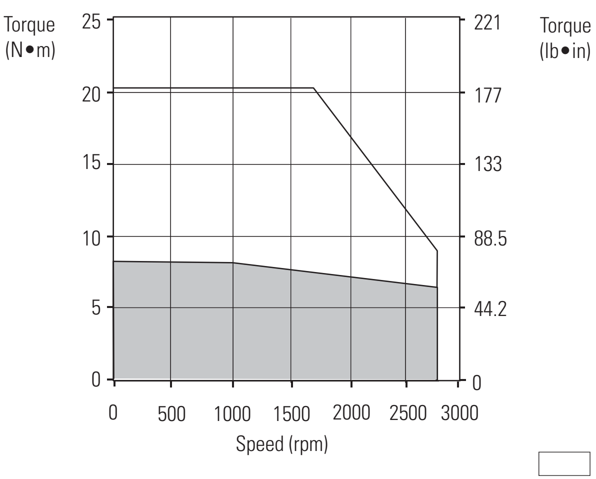
2097-V33PR6 and MPL-A430H



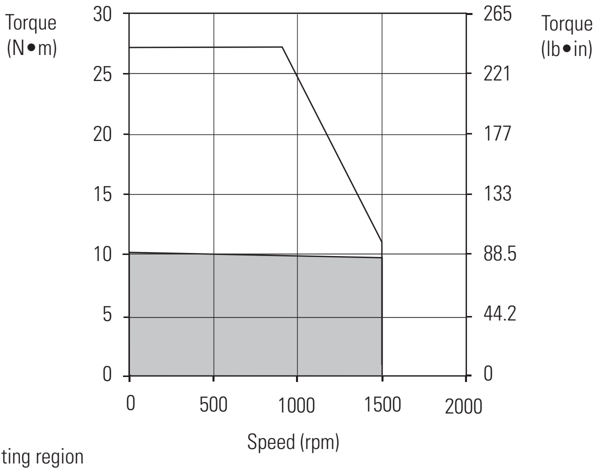
2097-V33PR6 and MPL-A430P



2097-V33PR6 and MPL-A4530F



2097-V33PR6 and MPL-A4540C



☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 300 (480V) Drives with MP-Series Low Inertia Motors

This section provides system combination information for the Kinetix 300 (480V) drives when matched with MP-Series low-inertia motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

IMPORTANT The MP-Series low-inertia motors on this page are equipped with DIN connectors (specified by 4 or 7 in the catalog number) and are not compatible with cables designed for motors equipped with bayonet connectors (specified by 2 in the catalog number). The motors with bayonet connectors (for example, MPL-A310P-*xx*2*x*AA) are being discontinued and require 2090-XXN*x*MP (bayonet) cables. For help with migration or to select bayonet cables, contact your Rockwell Automation sales representative.

Bulletin MPL Motor Cable Combinations

Motor Cat. No. (400V class)	Motor Power/Brake Cable	Motor Feedback Cable (1)
MPL-B1510V-*xx*4*x*AA, MPL-B1520U-*xx*4*x*AA, MPL-B1530U-*xx*4*x*AA	2090-XXNPMF-16S*xx* (standard) 2090-CP*x*M4DF-16AF*xx* (continuous-flex)	2090-XXNFMF-S*xx* (standard) 2090-CFBM4DF-CDAF*xx* (continuous-flex) Absolute High-resolution and Incremental Feedback
MPL-B210V-*xx*4*x*AA, MPL-B220T-*xx*4*x*AA, MPL-B230P-*xx*4*x*AA		
MPL-B310P-*xx*7*x*AA, MPL-B320P-*xx*7*x*AA, MPL-B330P-*xx*7*x*AA	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or (2) 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback 2090-XXNFMF-S*xx* (standard) (3) 2090-CFBM7DF-CDAF*xx* (continuous-flex) Incremental Feedback
MPL-B420P-*xx*7*x*AA		
MPL-B4530F-*xx*7*x*AA		

(1) Use low-profile connector kit (catalog number 2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

(2) Applies to Kinetix 300 drives and MPL-B3*xxx*-M/S…MPL-B45*xxx*-M/S motors with absolute high-resolution feedback.

(3) Applies to Kinetix 300 drives and MPL-B3*xxx*-H…MPL-B45*xxx*-H motors with incremental feedback.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

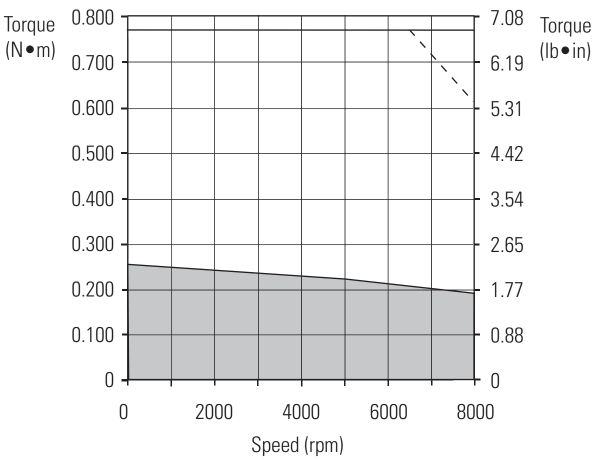
Bulletin MPL Motor Performance Specifications with Kinetix 300 (480V) Drives

Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A (0-pk)	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A (0-pk)	**System Peak Stall Torque** N•m (lb•in)	**Motor Rated Output** kW	**Kinetix 300 480V Drives**
MPL-B1510V	8000	0.95	0.26 (2.3)	3.10	0.77 (6.80)	0.16	2097-V34PR3
MPL-B1520U	7000	1.80	0.49 (4.3)	6.10	1.58 (13.9)	0.27	2097-V34PR3
MPL-B1530U	7000	2.0	0.90 (8.0)	7.20	2.82 (24.9)	0.39	2097-V34PR3
MPL-B210V	8000	1.75	0.55 (4.8)	5.80	1.52 (13.5)	0.37	2097-V34PR3
MPL-B220T	6000	3.30	1.61 (14.2)	11.3	4.74 (41.9)	0.62	2097-V34PR5
MPL-B230P	5000	2.60	2.10 (18.6)	11.3	8.20 (73.0)	0.86	2097-V34PR5
MPL-B310P	5000	2.4	1.58 (14)	7.1	3.61 (32)	0.77	2097-V34PR3
MPL-B320P	5000	4.5	2.94 (26)	14.0	7.91 (70)	1.5	2097-V34PR5
MPL-B330P	5000	6.1	4.18 (37)	19.0	11.1 (98)	1.8	2097-V34PR6
MPL-B420P	5000	6.4	4.74 (42)	22.0	13.5 (120)	1.9	2097-V34PR6
MPL-B4530F	3000	6.7	8.36 (74)	21.0	20.3 (180)	2.1	2097-V34PR6

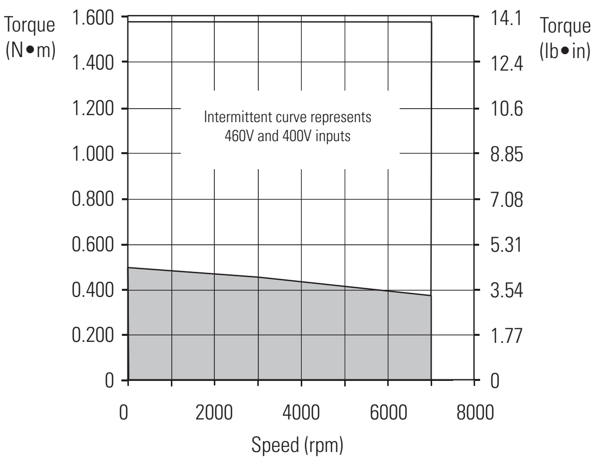
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 40 °C (104 °F) ambient, and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 300 (480V) Drives/MP-Series Low Inertia Motor Curves

2097-V34PR3 and MPL-B1510V

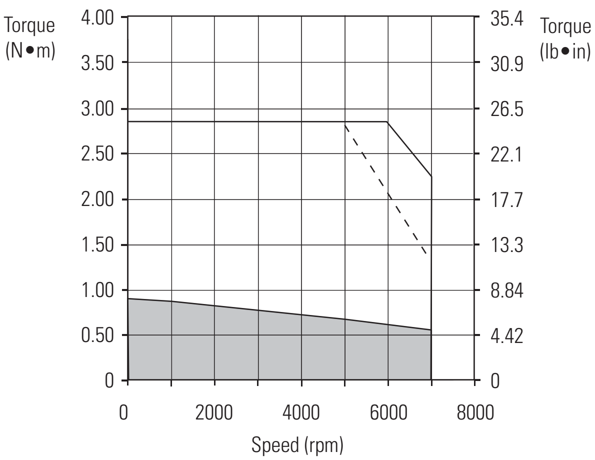


2097-V34PR3 and MPL-B1520U

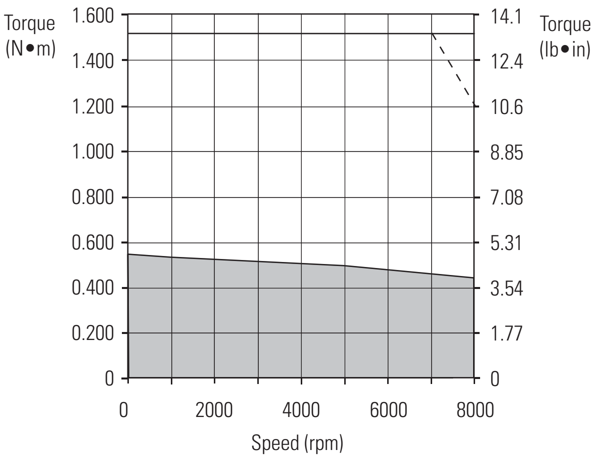


Intermittent curve represents 460V and 400V inputs

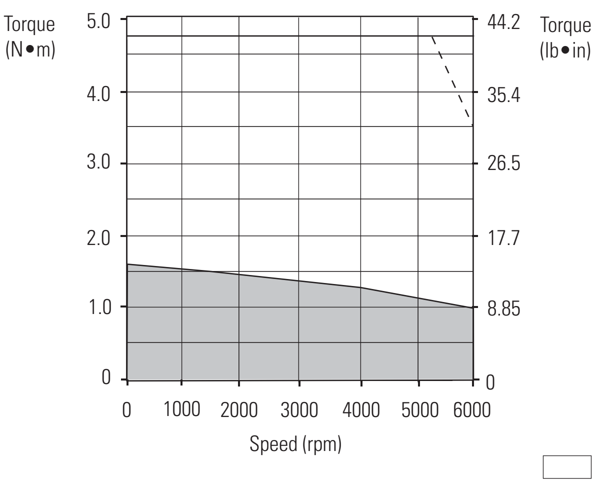
2097-V34PR3 and MPL-B1530U



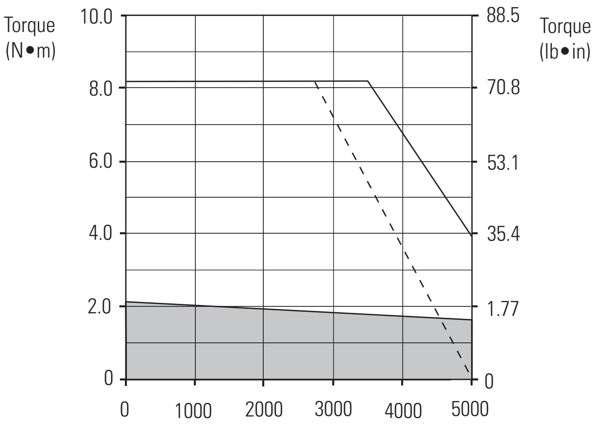
2097-V34PR3 and MPL-B210V



2097-V34PR5 and MPL-B220T



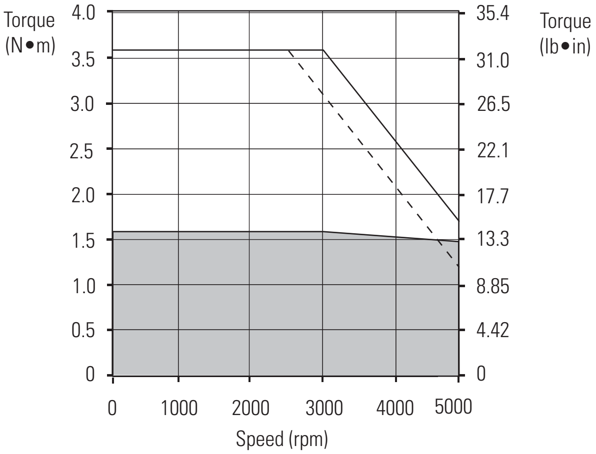
2097-V34PR5 and MPL-B230P



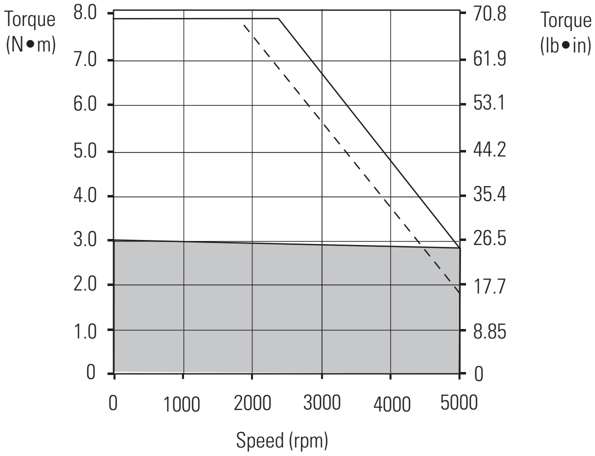
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Kinetix 300 (480V) Drives/MP-Series Low Inertia Motors, Continued

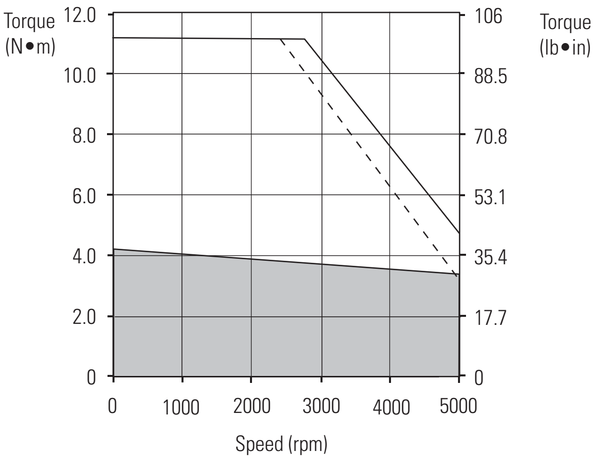
2097-V34PR3 and MPL-B310P



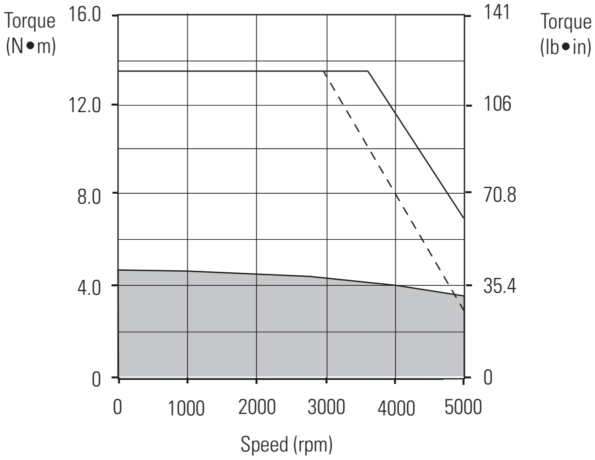
2097-V34PR5 and MPL-B320P



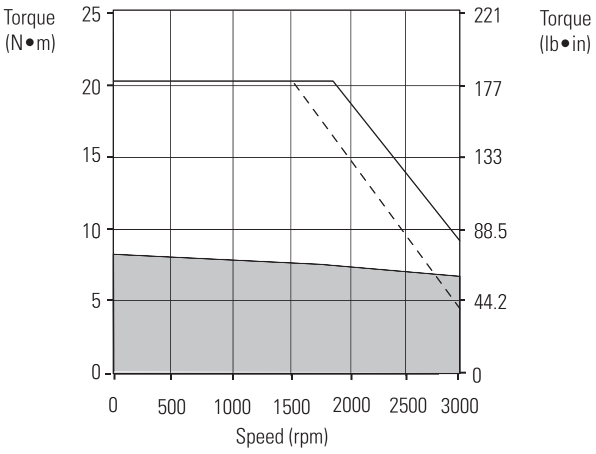
2097-V34PR6 and MPL-B330P



2097-V34PR6 and MPL-B420P



2097-V34PR6 and MPL-B4530F



☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Kinetix 300 (240V) Drives with MP-Series Medium Inertia Motors

This section provides system combination information for the Kinetix 300 (240V) drives when matched with MP-Series medium-inertia motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin MPM Motor Cable Combinations

Motor Cat. No. (200V class)	Motor Power/Brake Cable	Motor Feedback Cable [1]
MPM-A1151M, MPM-A1152F	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback

[1] Use low-profile connector kit (catalog number 2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

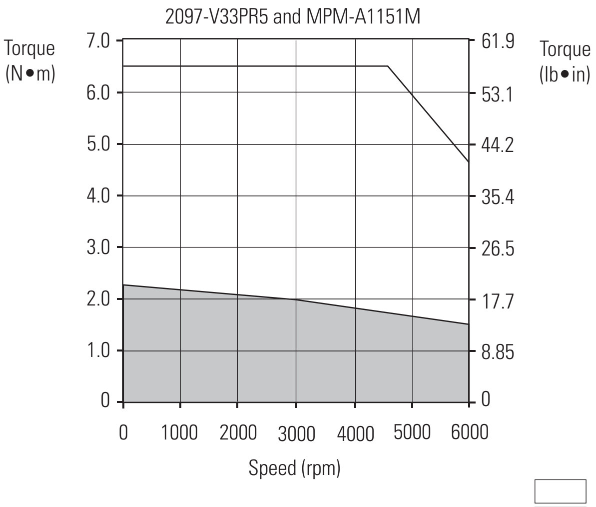
Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

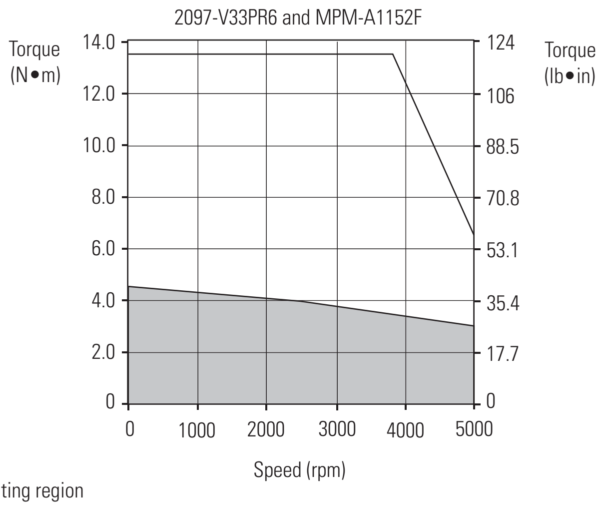
Bulletin MPM Motor Performance Specifications with Kinetix 300 (240V) Drives

Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 300 240V Drives
MPM-A1151M	6000	10.3	2.3 (20.3)	30.5	6.6 (58.4)	0.90	2097-V33PR5
MPM-A1152F	5000	14.9	4.7 (41.6)	44.8	13.5 (119)	1.40	2097-V33PR6

Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 40 °C (104 °F) ambient, and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 300 (240V) Drives/MP-Series Medium Inertia Motor Curves





☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 300 (480V) Drives with MP-Series Medium Inertia Motors

This section provides system combination information for the Kinetix 300 (480V) drives when matched with MP-Series medium-inertia motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin MPM Motor Cable Combinations

Motor Cat. No. (400V class)	Motor Power/Brake Cable	Motor Feedback Cable [1]
MPM-B1151F, MPM-B1151T, MPM-B1152C, MPM-B1152F, MPM-B1153E	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback
MPM-B1302F, MPM-B1304C		

[1] Use low-profile connector kit (catalog number 2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

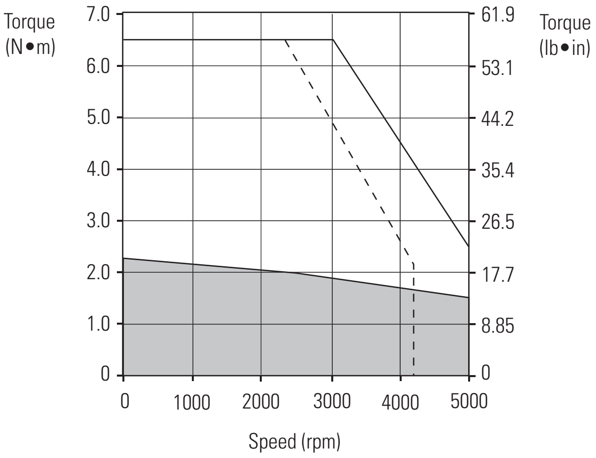
Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

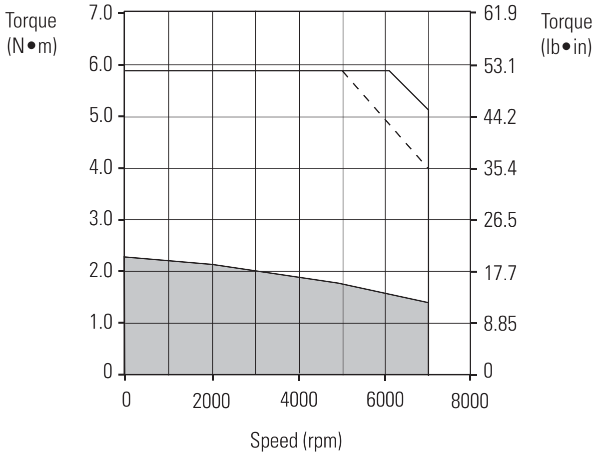
Bulletin MPM Motor Performance Specifications with Kinetix 300 (480V) Drives

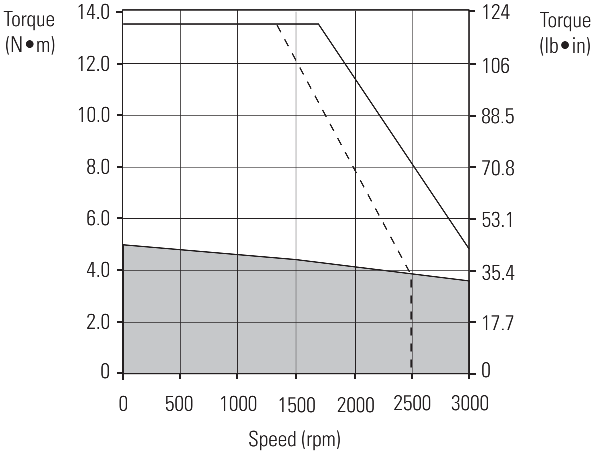
Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 300 480V Drives
MPM-B1151F	5000	3.1	2.3 (20.3)	9.9	6.6 (58.4)	0.75	2097-V34PR5
MPM-B1151T	7000	6.9	2.3 (20.3)	20.5	5.8 (51.3)	0.90	2097-V34PR6
MPM-B1152C	3000	4.1	5.0 (44.2)	12.4	13.5 (119)	1.20	2097-V34PR5
MPM-B1152F	5200	7.0	5.0 (44.2)	21.1	13.3 (118)	1.40	2097-V34PR6
MPM-B1153E	3500	7.1	6.5 (57.5)	21.6	19.7 (174)	1.40	2097-V34PR6
MPM-B1302F	4500	9.8	6.6 (58.4)	22.0	13.2 (117)	1.65	2097-V34PR6
MPM-B1304C	2750	8.0	10.3 (91.1)	22.3	27.1 (240)	2.00	2097-V34PR6

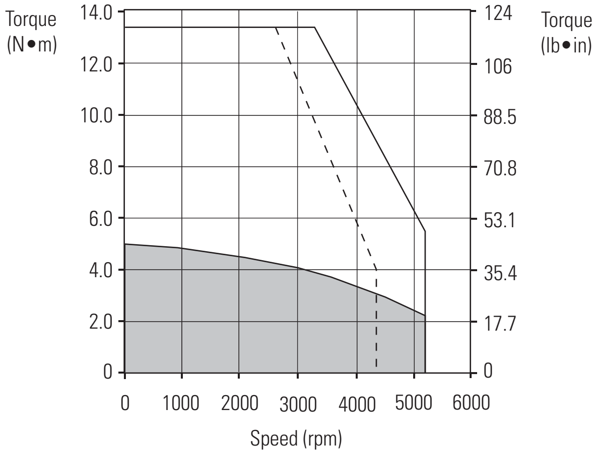
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 40 °C (104 °F) ambient, and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

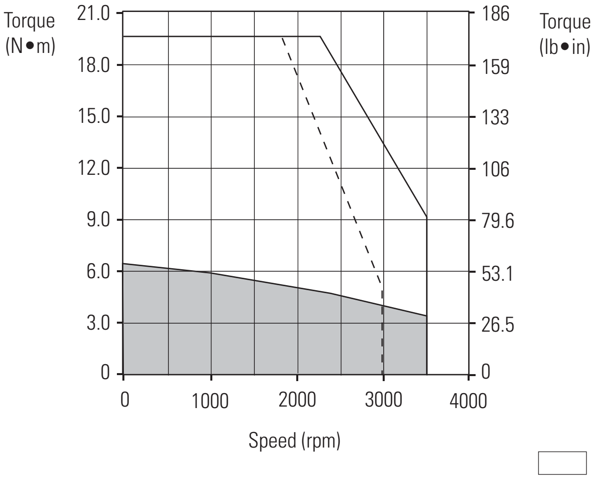
Kinetix 300 (480V) Drives/MP-Series Medium Inertia Motor Curves

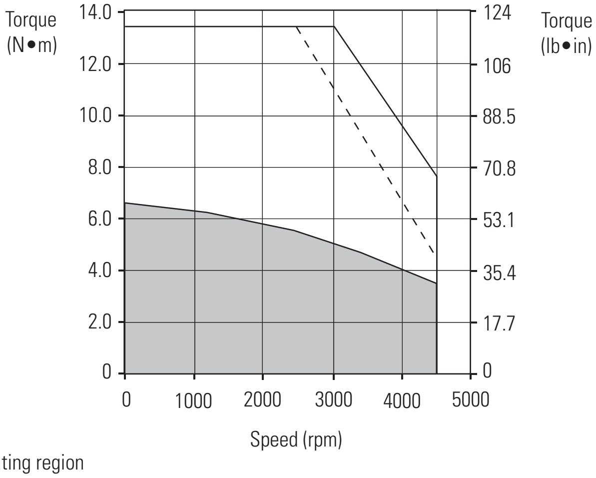

2097-V34PR5 and MPM-B1151F


2097-V34PR6 and MPM-B1151T


2097-V34PR5 and MPM-B1152C

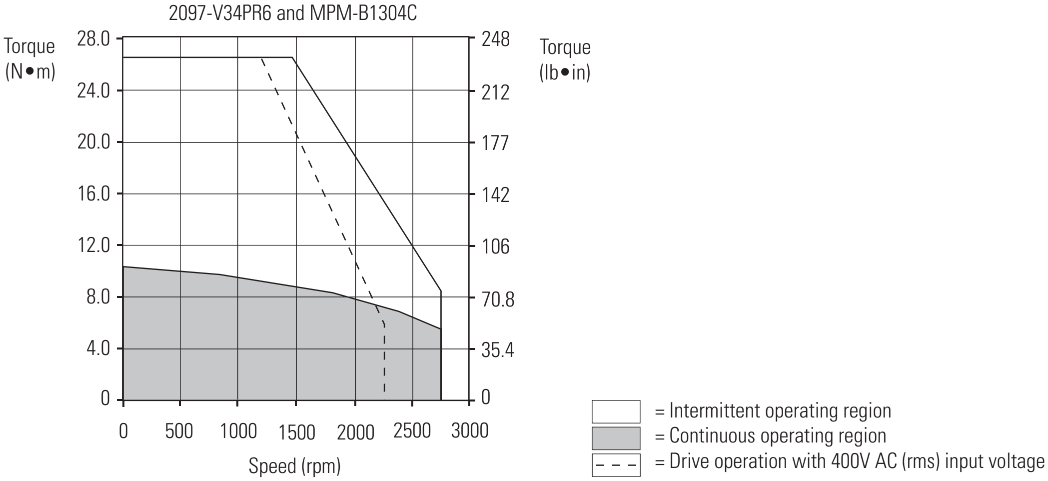

2097-V34PR6 and MPM-B1152F


2097-V34PR6 and MPM-B1153E


2097-V34PR6 and MPM-B1302F

☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Kinetix 300 (480V) Drives/MP-Series Medium Inertia Motor Curves, Continued


2097-V34PR6 and MPM-B1304C

Kinetix 300 (240V) Drives with MP-Series Food Grade Motors

This section provides system combination information for the Kinetix 300 (240V) drives when matched with MP-Series food-grade motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin MPF Motor Cable Combinations

Motor Cat. No. (200V class)	Motor Power Cable	Motor Feedback Cable [1]
MPF-A310P, MPF-A320H, MPF-A320P, MPF-A330P	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback
MPF-A430H		

(1) Use low-profile connector kit (catalog number 2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

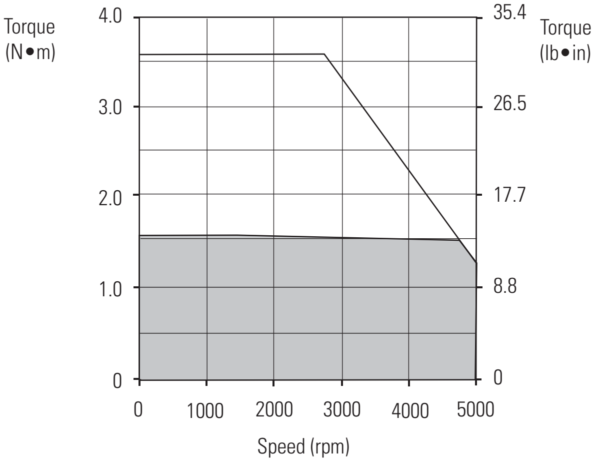
Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

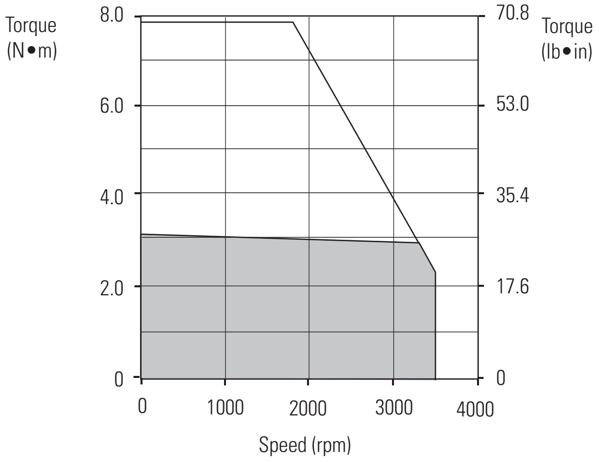
Bulletin MPF Motor Performance Specifications with Kinetix 300 (240V) Drives

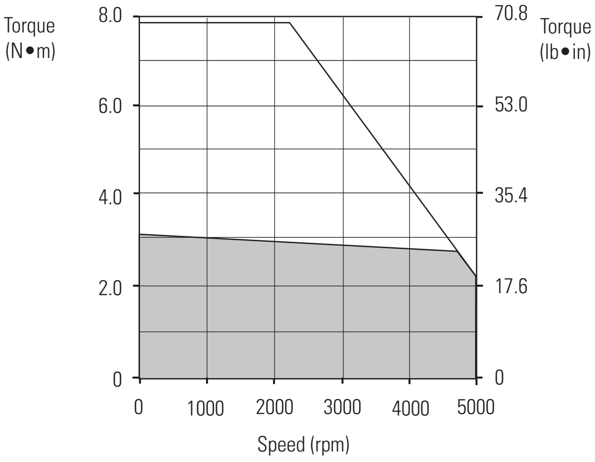
Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Stall Torque** N•m (lb•in)	**Motor Rated Output** kW	Kinetix 300 240V Drives
MPF-A310P	5000	4.85	1.58 (14)	14	3.61 (32)	0.73	2097-V33PR3
MPF-A320H	3500	6.1	3.05 (27)	19.3	7.91 (70)	1.0	2097-V33PR5
MPF-A320P	5000	9.0	3.05 (27)	29.5	7.91 (70)	1.3	2097-V33PR3
MPF-A330P	5000	12.0	4.18 (37)	38	11.1 (98)	1.6	2097-V33PR6
MPF-A430H	3500	12.2	6.21 (55)	45	19.8 (175)	1.8	2097-V33PR6

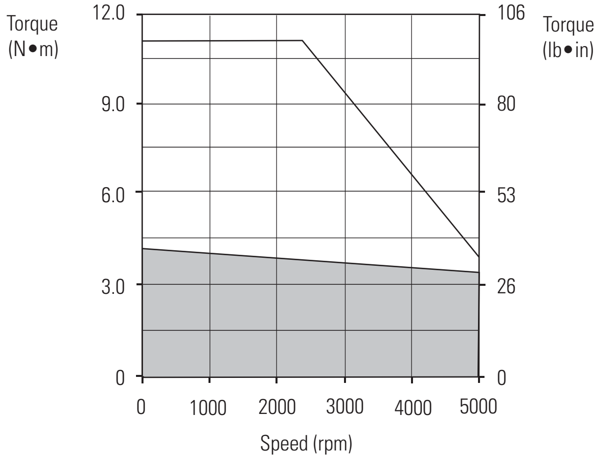
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 40 °C (104 °F) ambient, and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

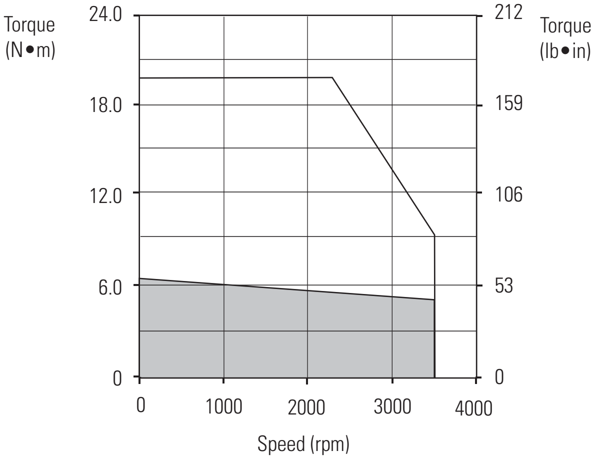
Kinetix 300 (240V) Drives/MP-Series Food Grade Motor Curves


2097-V33PR3 and MPF-A310P


2097-V33PR5 and MPF-A320H


2097-V33PR3 and MPF-A320P


2097-V33PR6 and MPF-A330P


2097-V33PR6 and MPF-A430H

☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 300 (480V) Drives with MP-Series Food Grade Motors

This section provides system combination information for the Kinetix 300 (480V) drives when matched with MP-Series food-grade motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin MPF Motor Cable Combinations

Motor Cat. No. (400V class)	Motor Power Cable	Motor Feedback Cable [1]
MPF-B310P, MPF-B320P, MPF-B330P	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback

(1) Use low-profile connector kit (catalog number 2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

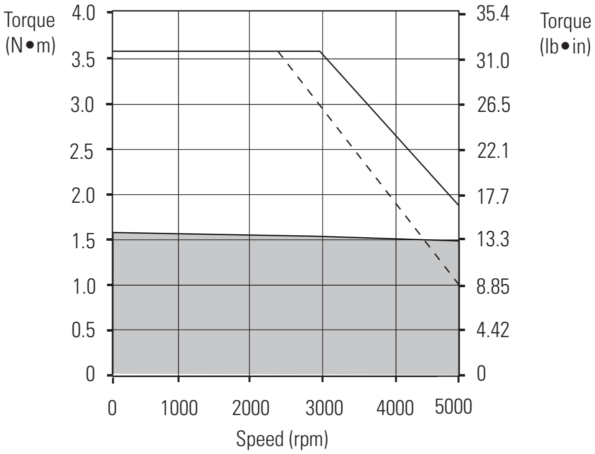
Bulletin MPF Motor Performance Specifications with Kinetix 300 (480V) Drives

Rotary Motor	Max Speed rpm	System Continuous Stall Current A (0-pk)	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A (0-pk)	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 300 480V Drives
MPF-B310P	5000	2.30	1.58 (14)	7.1	3.61 (32)	0.77	2097-V34PR3
MPF-B320P	5000	4.24	3.05 (27)	14.0	7.34 (65)	1.5	2097-V34PR5
MPF-B330P	5000	5.70	4.18 (37)	16.9	10.0 (88)	1.6	2097-V34PR5
				19.0	11.1 (98)		2097-V34PR6

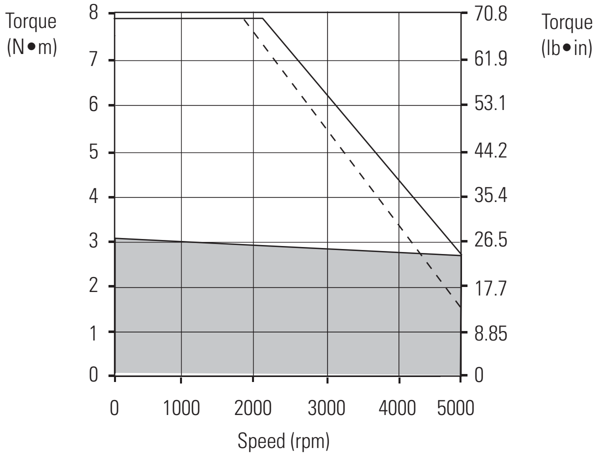
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 40 °C (104 °F) ambient, and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 300 (480V) Drives/MP-Series Food Grade Motor Curves

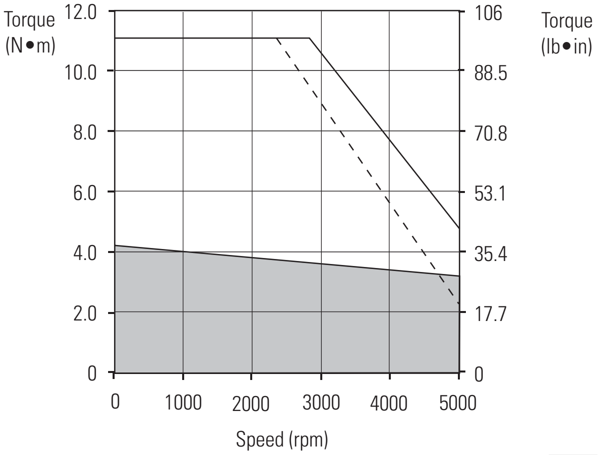
2097-V34PR3 and MPF-B310P



2097-V34PR5 and MPF-B320P



2097-V34PR6 and MPF-B330P



☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Kinetix 300 (240V) Drives with MP-Series Stainless Steel Motors

This section provides system combination information for the Kinetix 300 (240V) drives when matched with MP-Series stainless-steel motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin MPS Motor Cable Combinations

Motor Cat. No. (200V class)	Motor Power Cable	Motor Feedback Cable [1]
MPS-A330P	2090-XXNPMF-16S*xx* [2]	2090-XXNFMF-S*xx* [3] Absolute High-resolution Feedback
MPS-A4540F		

(1) Use low-profile connector kit (catalog number 2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

(2) These cables are available as standard (catalog number 2090-XXNPMF-*xx*S*xx*) or continuous-flex (catalog number 2090-CP*x*M4DF-*xx*AF*xx*).

(3) These cables are available as standard (catalog number 2090-XXNFMF-S*xx*) or continuous-flex (catalog number 2090-CFBM4DF-CDAF*xx*).

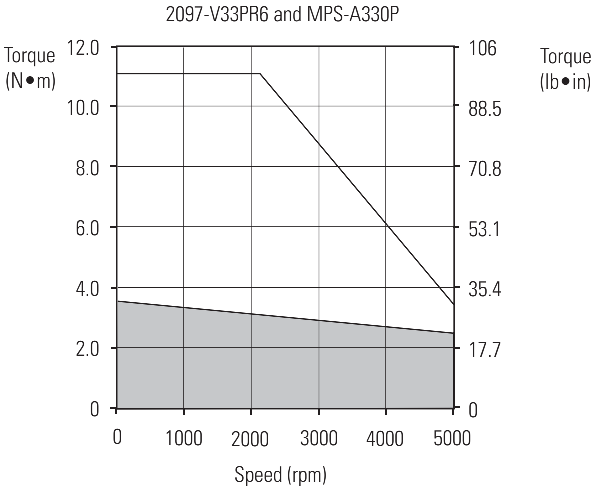
Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

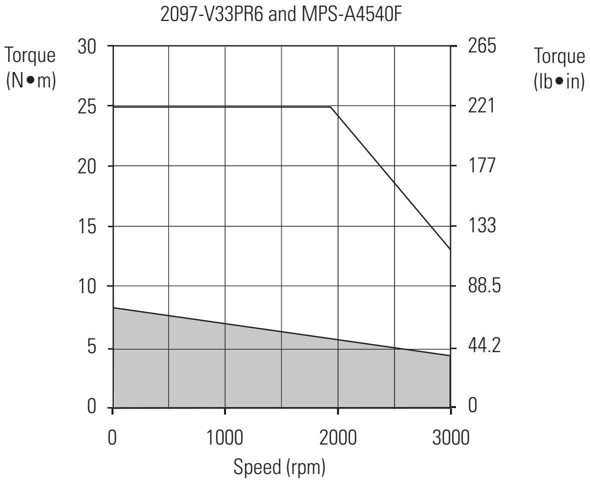
Bulletin MPS Motor Performance Specifications with Kinetix 300 (240V) Drives

Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Stall Torque** N•m (lb•in)	**Motor Rated Output** kW	**Kinetix 300 240V Drives**
MPS-A330P	5000	9.80	3.60 (32)	33.9	10.1 (89.4)	1.3	2097-V33PR5
				38.0	11.1 (98.2)		2097-V33PR6
MPS-A4540F	3000	14.4	8.1 (72)	50.9	24.8 (219)	1.4	2097-V33PR6

Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 40 °C (104 °F) ambient, and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 300 (240V) Drives/MP-Series Stainless Steel Motor Curves





☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 300 (480V) Drives with MP-Series Stainless Steel Motors

This section provides system combination information for the Ultra3000 (460V) drives when matched with MP-Series stainless-steel motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin MPS Motor Cable Combinations

Motor Cat. No. (400V class)	Motor Power Cable	Motor Feedback Cable (1)
MPS-B330P	2090-XXNPMF-16S*xx* (2)	2090-XXNFMF-S*xx* (3) Absolute High-resolution Feedback
MPS-B4540F		

(1) Use low-profile connector kit (catalog number 2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

(2) These cables are available as standard (catalog number 2090-XXNPMF-*xx*S*xx*) or continuous-flex (catalog number 2090-CP*x*M4DF-*xx*AF*xx*).

(3) These cables are available as standard (catalog number 2090-XXNFMF-S*xx*) or continuous-flex (catalog number 2090-CFBM4DF-CDAF*xx*).

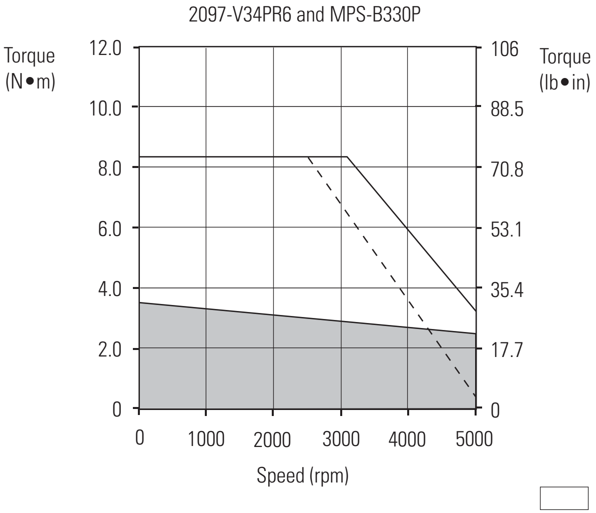
Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

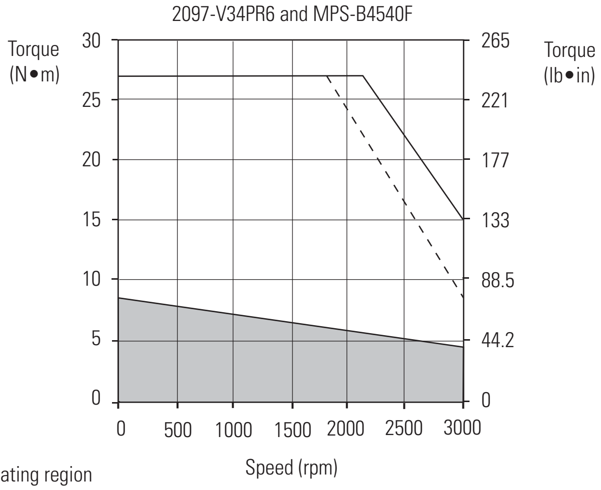
Bulletin MPS Motor Performance Specifications with Kinetix 300 (480V) Drives

Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 300 480V Drives
MPS-B330P	5000	4.90	3.6 (32)	16.9	10.1 (89.4)	1.3	2097-V34PR5
				19.0	11.1 (98.2)		2097-V34PR6
MPS-B4540F	3000	7.1	8.1 (72)	25.4	26.3 (233)	1.4	2097-V34PR6

Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 40 °C (104 °F) ambient, and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 300 (480V) Drives/MP-Series Stainless Steel Motor Curves


2097-V34PR6 and MPS-B330P


2097-V34PR6 and MPS-B4540F

☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Kinetix 300 (240V) Drives with TL-Series Low Inertia Motors

This section provides system combination information for the Kinetix 300 drives when matched with TL-Series (Bulletin TLY) low-inertia motors. Compatible TL-Series motors are equipped with absolute high-resolution or incremental encoder feedback. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin TLY Motor Cable Combinations

Motor Cat. No.	Motor Power/Brake Cable	Motor Feedback Cable [1]
TLY-A110*x*, TLY-A120*x*, TLY-A130*x*	2090-CPWM6DF-16AA*xx* (standard) (without brake)	2090-CFBM6DF-CBAA*xx* (standard) Absolute High-resolution
TLY-A220*x*, TLY-A230*x*		
TLY-A2530P, TLY-A2540P	2090-CPBM6DF-16AA*xx* (standard) (with brake)	2090-CFBM6DF-CBAA*xx* or 2090-CFBM6DD-CCAA*xx* (standard) Incremental Feedback
TLY-A310M		

[1] For TLY-A*xxxx*-H motors with incremental encoder feedback, use 2090-CFBM6DF-CBAA*xx* flying-lead cables and 2090-K2CK-D15M connector kit (battery not required) or panel-mounted breakout components on the drive end. Refer to Breakout Components and Connector Kits beginning on page 440 for more information.

The TLY-A*xxxx*-B motors with 17-bit high resolution encoder feedback require the 2090-CFBM6DF-CBAA*xx* flying-lead feedback cable and 2090-K2CK-D15M connector kit with 2090-DA-BAT2 battery.

TL-Series (Bulletin TLY) motors are characterized as having 1000 mm (39.4 in.) cable extensions with circular plastic connectors and TLY-A*xxx* catalog numbers.
Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Bulletin TLY (non-brake) Motor Performance Specifications with Kinetix 300 (240V) Drives

Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 300 240V Drives
TLY-A110*x*	6000 [1]	0.55	0.096 (0.85)	1.50	0.20 (1.75)	0.041	2097-V32PR0
TLY-A120*x*		1.03	0.181 (1.60)	2.50	0.36 (3.20)	0.086	2097-V32PR0
TLY-A130*x*		1.85	0.325 (2.88)	4.90	0.76 (6.70)	0.14	2097-V32PR0
TLY-A220*x*		3.50	0.836 (7.40)	7.90	1.48 (13.1)	0.35	2097-V33PR1
TLY-A230*x*		5.50	1.30 (11.5)	15.5	3.05 (27.0)	0.44	2097-V33PR3
TLY-A2530P	5000	10.0	2.60 (23.0)	21.0	5.20 (46.0)	0.69	2097-V33PR5
TLY-A2540P		10.0	2.94 (26.0)	24.8	7.10 (63.0)	0.86	2097-V33PR5
TLY-A310M	4500	10.0	3.61 (31.9)	30.0	9.0 (79.6)	0.95	2097-V33PR5

[1] Applies to TLY-A*xxx*T-H motors with incremental feedback. The TLY-A*xxx*P-B motors with absolute high-resolution encoders are rated at 5000 rpm.

Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 40 °C (104 °F) ambient, and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Bulletin TLY (brake) Motor Performance Specifications with Kinetix 300 (240V) Drives

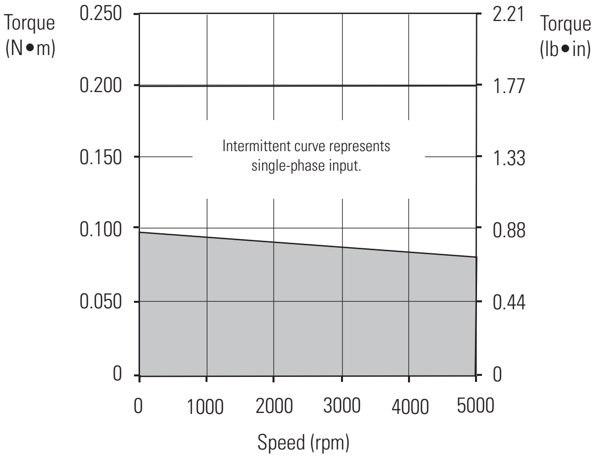
Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Torque** N•m (lb•in)	**Motor Rated Output** kW	**Kinetix 300 240V Drives**
TLY-A110*x*	6000 [1]	0.50	0.086 (0.76)	1.50	0.20 (1.75)	0.037	2097-V32PR0
TLY-A120*x*		0.93	0.163 (1.44)	2.50	0.36 (3.20)	0.077	2097-V32PR0
TLY-A130*x*		1.67	0.293 (2.59)	4.90	0.76 (6.70)	0.13	2097-V32PR0
TLY-A220*x*		3.15	0.757 (6.70)	7.90	1.48 (13.1)	0.24	2097-V33PR1
TLY-A230*x*		4.95	1.16 (10.3)	15.5	3.05 (27.0)	0.32	2097-V33PR3
TLY-A2530P	5000	10.0	2.60 (23.0)	21.0	5.20 (46.0)	0.55	2097-V33PR5
TLY-A2540P		10.0	2.94 (26.0)	24.8	7.10 (63.0)	0.66	2097-V33PR5
TLY-A310M	4500	10.0	3.61 (31.9)	30.0	9.0 (79.6)	0.90	2097-V33PR5

(1) Applies to TLY-A*xxx*T-H motors with incremental feedback. The TLY-A*xxx*P-B motors with absolute high-resolution encoders are rated at 5000 rpm.

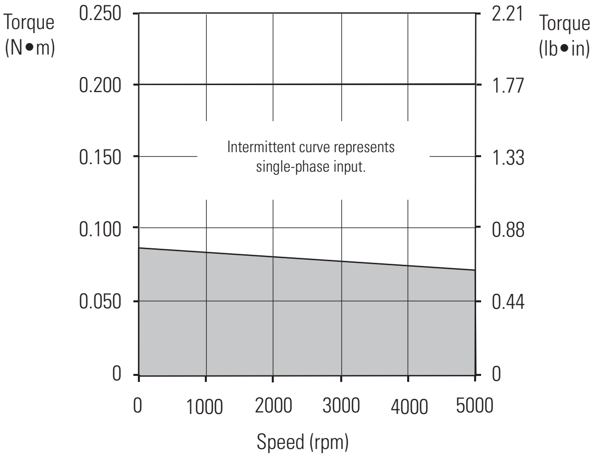
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 40 °C (104 °F) ambient, and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 300 (240V) Drives/TLY-A*xxx*P-B (absolute high-resolution) Motor Curves

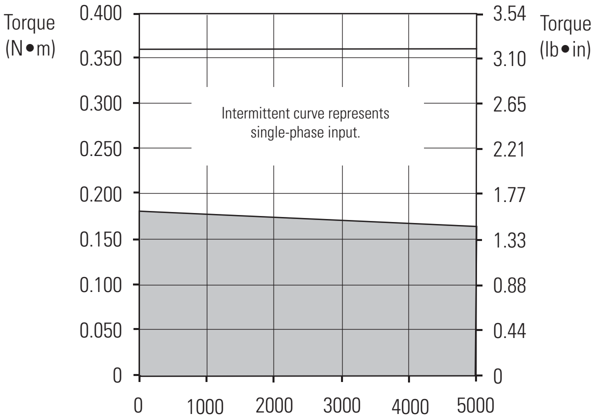
2097-V32PR0 and TLY-A110P-B



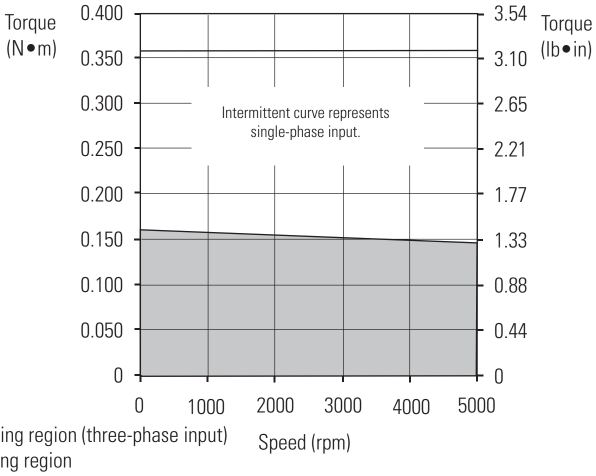
2097-V32PR0 and TLY-A110P-B (Brake)



2097-V32PR0 and TLY-A120P-B



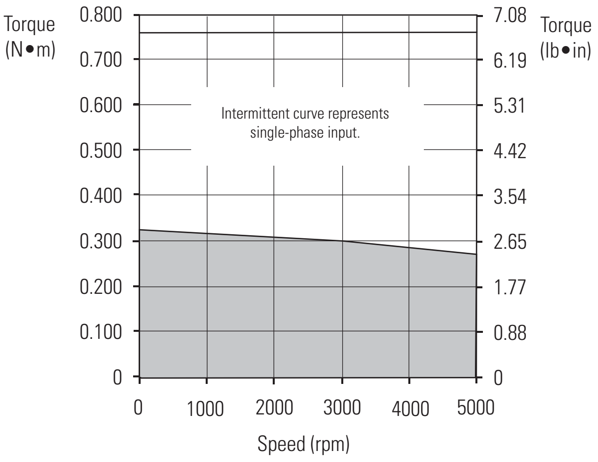
2097-V32PR0 and TLY-A120P-B (Brake)



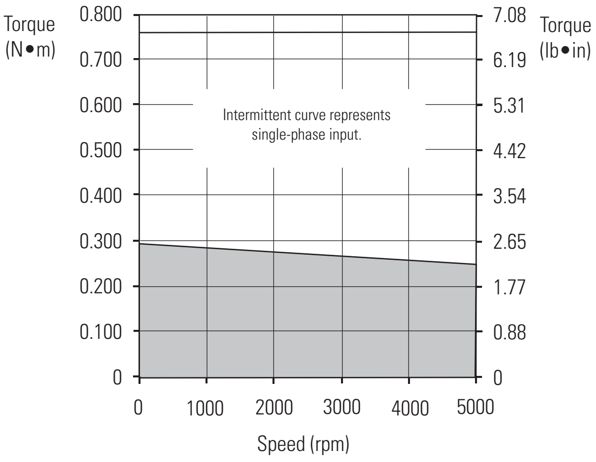
☐ = Intermittent operating region (three-phase input)
▨ = Continuous operating region
– – – = Drive operation (single-phase input)

Kinetix 300 (240V) Drives/TLY-A*xxx*P-B (absolute high-resolution) Motor Curves, Continued

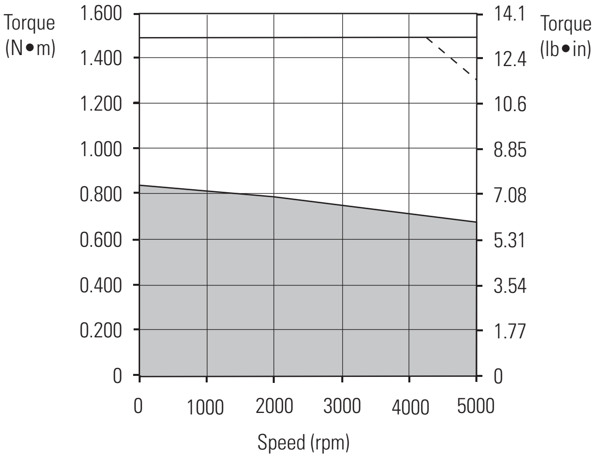
2097-V32PR0 and TLY-A130P-B



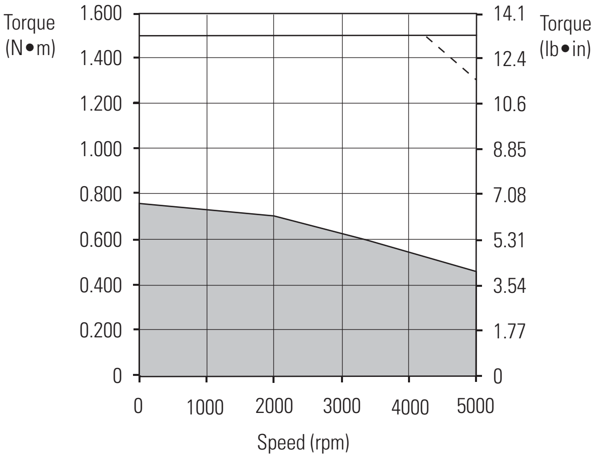
2097-V32PR0 and TLY-A130P-B (Brake)



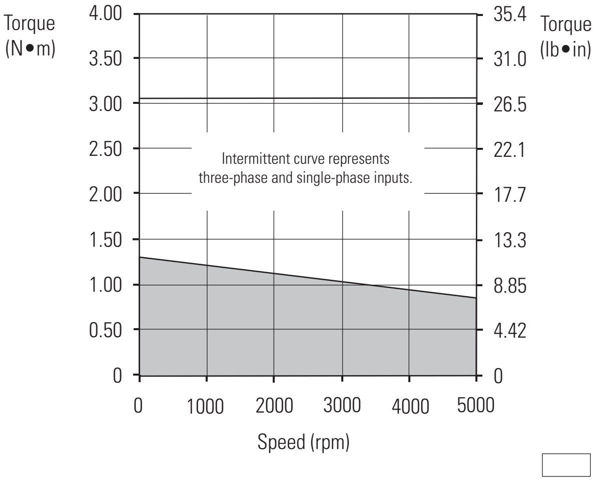
2097-V33PR1 and TLY-A220P-B



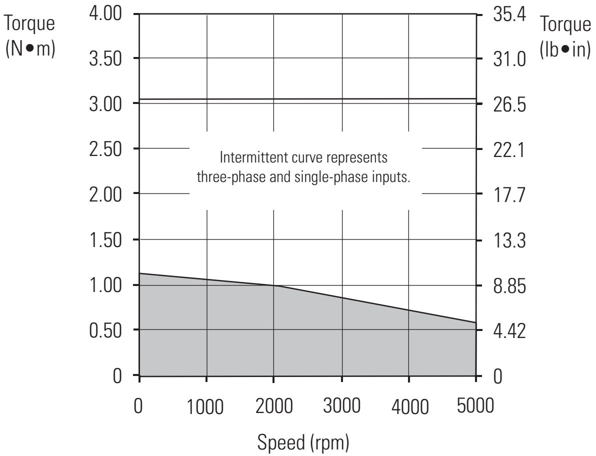
2097-V33PR1 and TLY-A220P-B (Brake)



2097-V33PR3 and TLY-A230P-B



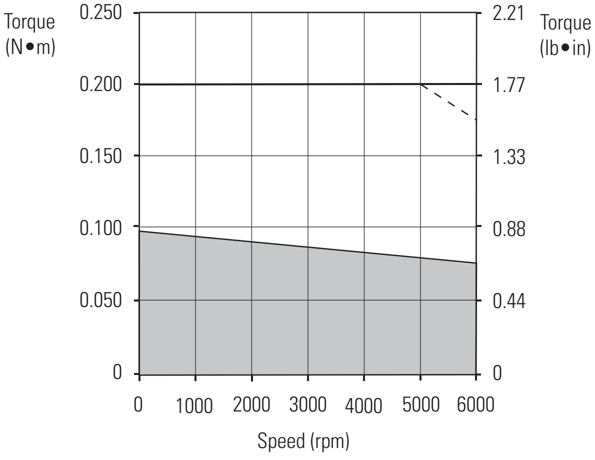
2097-V33PR3 and TLY-A230P-B (Brake)



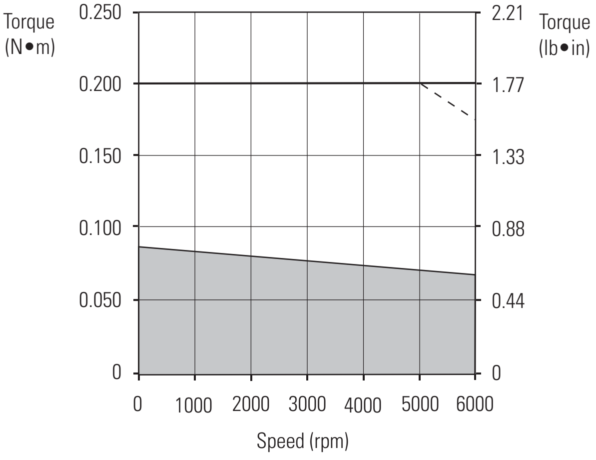
☐ = Intermittent operating region (three-phase input)
▨ = Continuous operating region
– – – = Drive operation (single-phase input)

Kinetix 300 (240V) Drives/TLY-A*xxx*T-H (incremental) Motor Curves

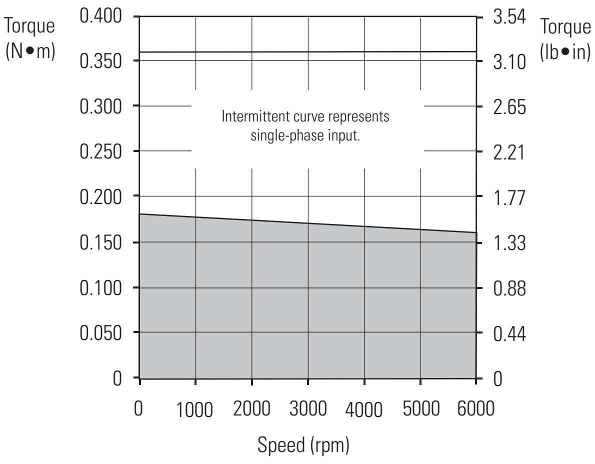
2097-V32PR0 and TLY-A110T-H



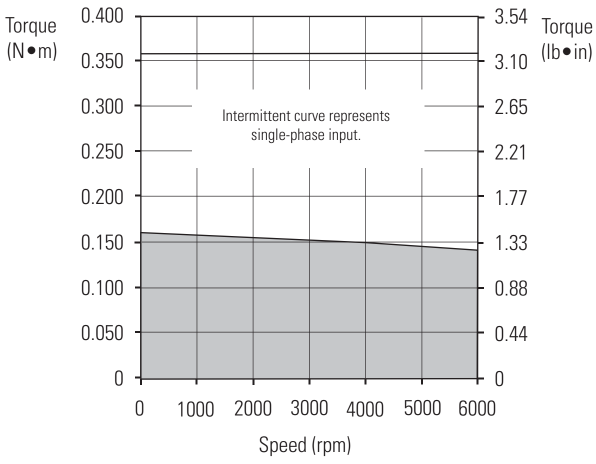
2097-V32PR0 and TLY-A110T-H (Brake)



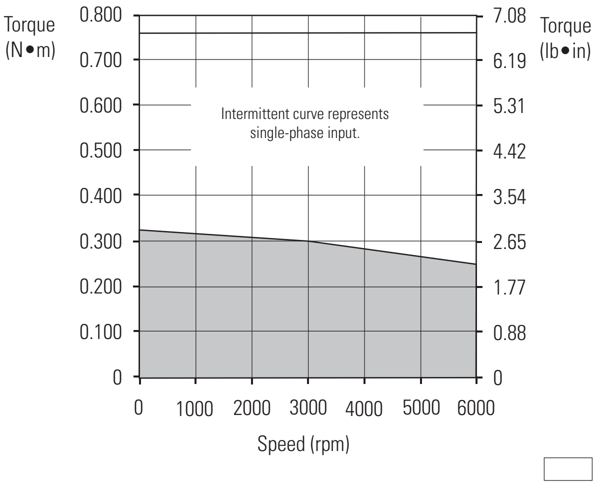
2097-V32PR0 and TLY-A120T-H



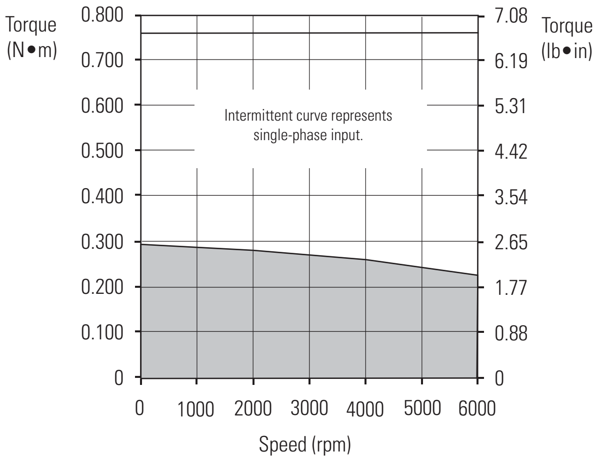
2097-V32PR0 and TLY-A120T-H (Brake)



2097-V32PR0 and TLY-A130T-H

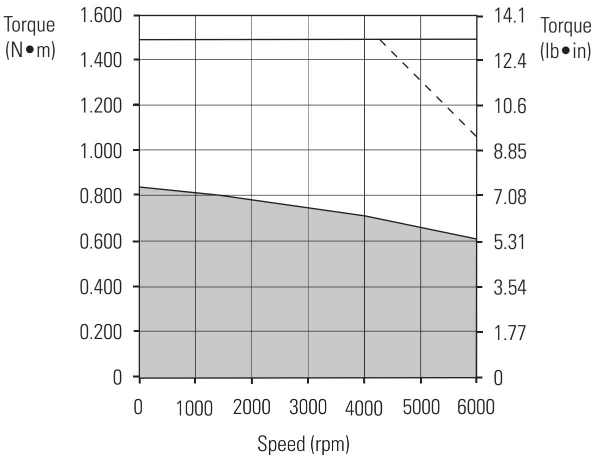


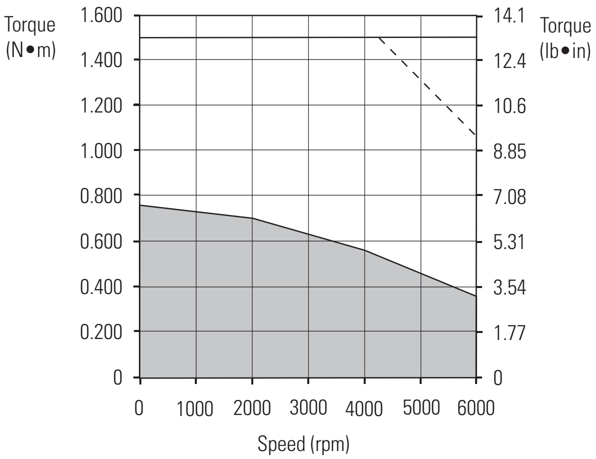
2097-V32PR0 and TLY-A130T-H (Brake)

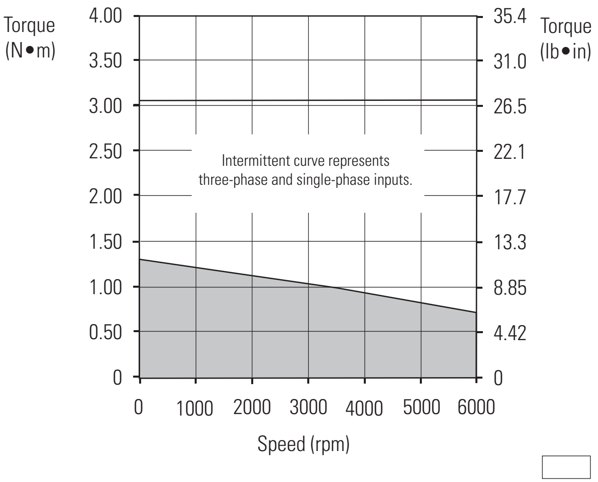


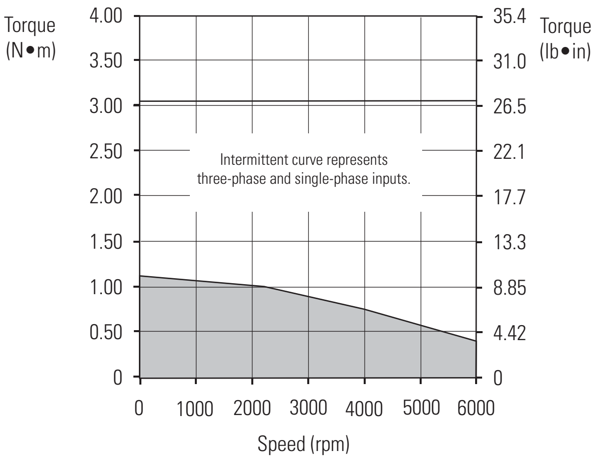
☐ = Intermittent operating region (three-phase input)
▨ = Continuous operating region
- - - = Drive operation (single-phase input)

Kinetix 300 (240V) Drives/TLY-A*xxx*T-H (incremental) Motor Curves, Continued


2097-V33PR1 and TLY-A220T-H


2097-V33PR1 and TLY-A220T-H (Brake)

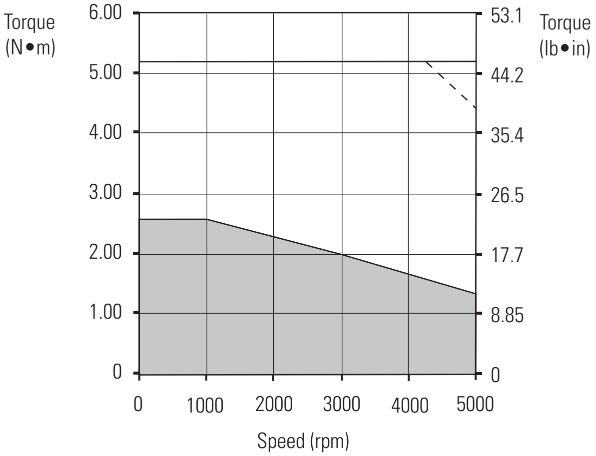

2097-V33PR3 and TLY-A230T-H


2097-V33PR3 and TLY-A230T-H (Brake)

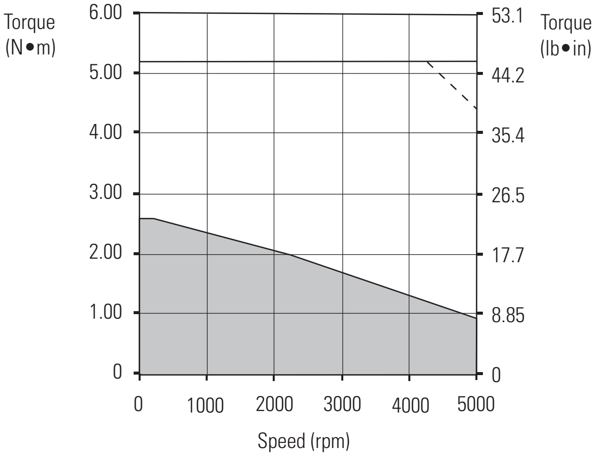
☐ = Intermittent operating region (three-phase input)
▨ = Continuous operating region
– – – = Drive operation (single-phase input)

Kinetix 300 (240V) Drives/TLY-A*xxxx*-*x* Motor Curves

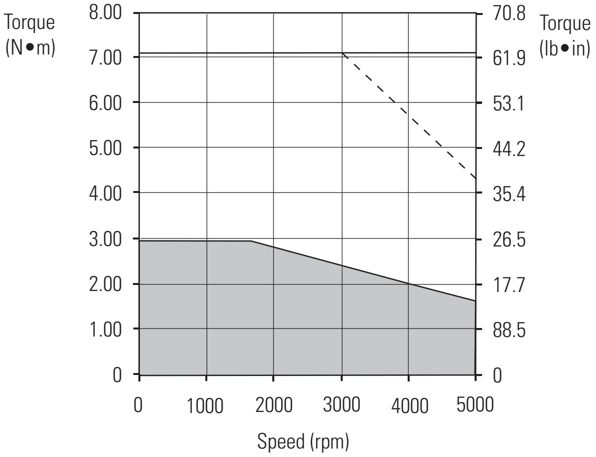
2097-V33PR5 and TLY-A2530P-*x*



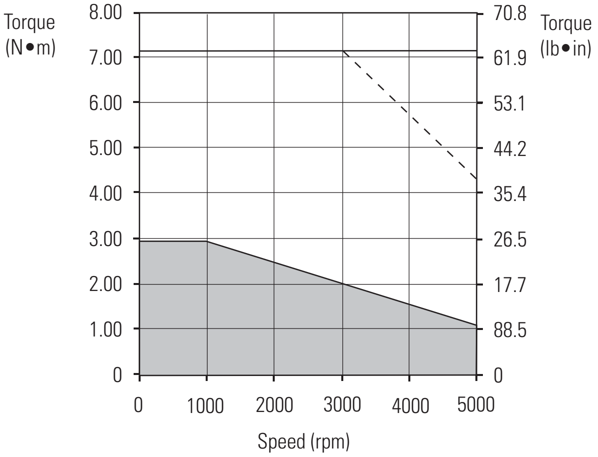
2097-V33PR5 and TLY-A2530P-*x* (Brake)



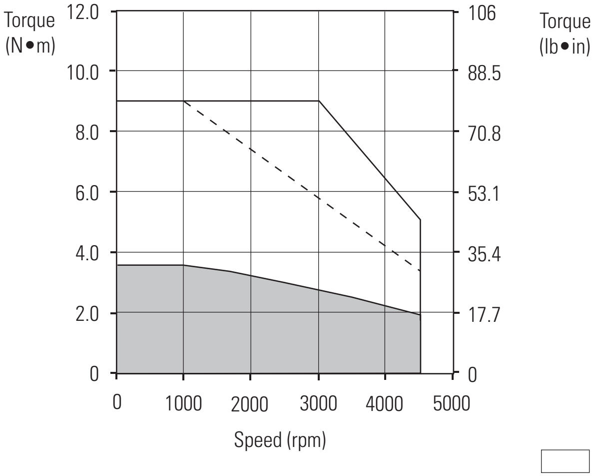
2097-V33PR5 and TLY-A2540P-*x*



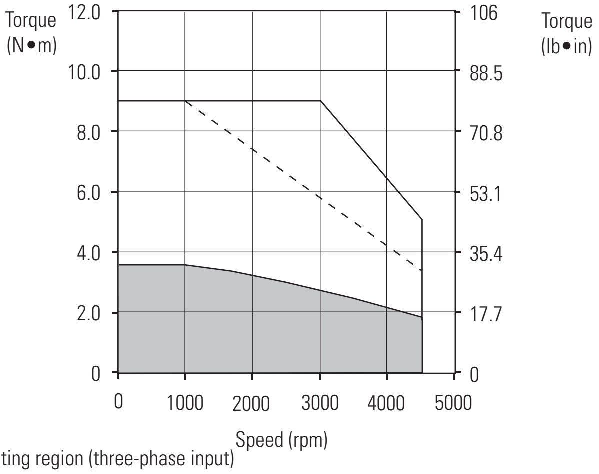
2097-V33PR5 and TLY-A2540P-*x* (Brake)



2097-V33PR5 and TLY-A310M-*x*



2097-V33PR5 and TLY-A310M-*x* (Brake)



☐ = Intermittent operating region (three-phase input)
▨ = Continuous operating region
- - - = Drive operation (single-phase input)

Kinetix 3 (240V) Drives with TL-Series (Bulletin TLY) Low Inertia Motors

This section provides system combination information for the Kinetix 3 drives when matched with TL-Series (Bulletin TLY) low-inertia motors. Compatible TL-Series motors are equipped with absolute high-resolution or incremental encoder feedback. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin TLY Motor Cable Combinations

Motor Cat. No.	Motor Power/Brake Cable	Motor Feedback Cable [1]
TLY-A120*x*, TLY-A130*x*	2090-CPWM6DF-16AA*xx* (standard) (without brake) 2090-CPBM6DF-16AA*xx* (standard) (with brake)	2090-CFBM6DF-CBAA*xx* (standard) Absolute High-resolution or Incremental Feedback
TLY-A220*x*, TLY-A230*x*		
TLY-A2540P		
TLY-A310M		

[1] For TLY-A*xxxx*-H motors with incremental encoder feedback, use 2090-CFBM6DF-CBAA*xx* flying-lead cables and 2071-TBMF breakout board (battery not required). Refer to Breakout Components and Connector Kits beginning on page 440 for more information.

The TLY-A*xxxx*-B motors with 17-bit high-resolution encoder feedback require the 2090-CFBM6DF-CBAA*xx* flying-lead feedback cable and 2071-TBMF breakout board with 3.6V lithium battery (customer-supplied). Refer to Battery Specifications on page 356.

TL-Series (Bulletin TLY) motors are characterized as having 1000 mm (39.4 in.) cable extensions with circular plastic connectors and TLY-A*xxx* catalog numbers.
Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Bulletin TLY (non-brake) Motor Performance Specifications with Kinetix 3 (240V) Drives

Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 3 240V Drives
TLY-A120*x*	6000 [1]	1.03	0.181 (1.60)	2.50	0.36 (3.20)	0.086	2071-AP1
TLY-A130*x*		1.85	0.325 (2.88)	4.90	0.76 (6.70)	0.14	2071-AP1
TLY-A220*x*		3.50	0.836 (7.40)	7.90	1.48 (13.1)	0.35	2071-AP4
TLY-A230*x*		5.50	1.30 (11.5)	15.5	3.05 (27.0)	0.44	2071-AP4
TLY-A2540P	5000	10.0	2.94 (26.0)	24.8	7.10 (63.0)	0.86	2071-AP8
TLY-A310M	4500	10.0	3.61 (31.9)	30.0	9.0 (79.6)	0.95	2071-A10

[1] Applies to TLY-A*xxx*T-H motors with incremental feedback. The TLY-A*xxx*P-B motors with absolute high-resolution encoders are rated at 5000 rpm.

Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 5.000 or later.

Bulletin TLY (brake) Motor Performance Specifications with Kinetix 3 (240V) Drives

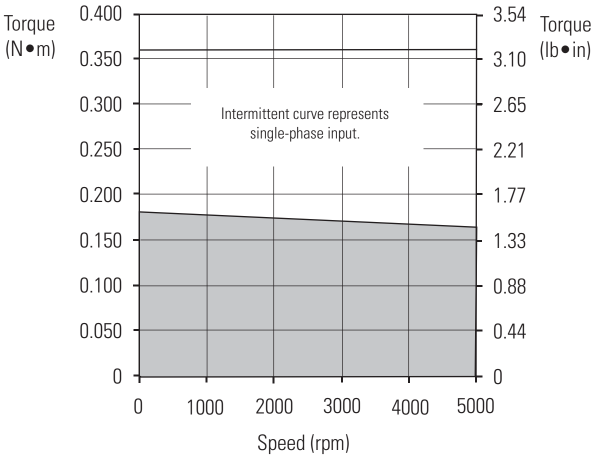
Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Torque** N•m (lb•in)	**Motor Rated Output** kW	**Kinetix 3 240V Drives**
TLY-A120*x*	6000 (1)	0.93	0.163 (1.44)	2.50	0.36 (3.20)	0.077	2071-AP1
TLY-A130*x*		1.67	0.293 (2.59)	4.90	0.76 (6.70)	0.13	2071-AP1
TLY-A220*x*		3.15	0.757 (6.70)	7.90	1.48 (13.1)	0.24	2071-AP4
TLY-A230*x*		4.95	1.16 (10.3)	15.5	3.05 (27.0)	0.32	2071-AP4
TLY-A2540P	5000	10.0	2.94 (26.0)	24.8	7.10 (63.0)	0.66	2071-AP8
TLY-A310M	4500	10.0	3.61 (31.9)	30.0	9.0 (79.6)	0.90	2071-A10

(1) Applies to TLY-A*xxx*T-H motors with incremental feedback. The TLY-A*xxx*P-B motors with absolute high-resolution encoders are rated at 5000 rpm.

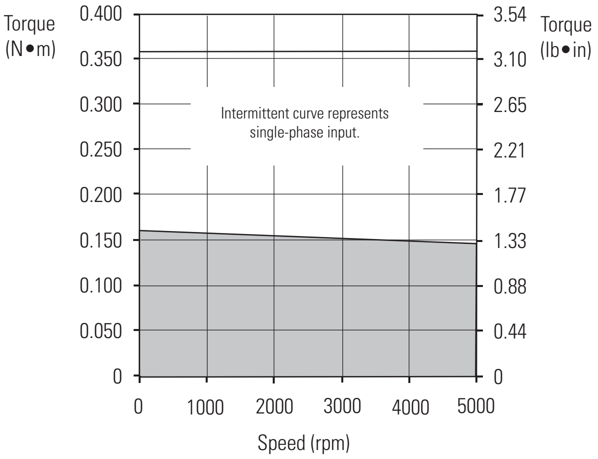
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 5.000 or later.

Kinetix 3 (240V) Drives/TLY-A*xxx*P-B (absolute high-resolution) Motor Curves

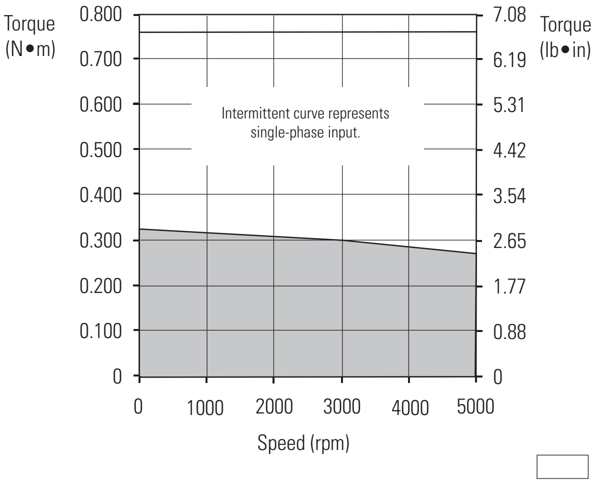
2071-AP1 and TLY-A120P-B



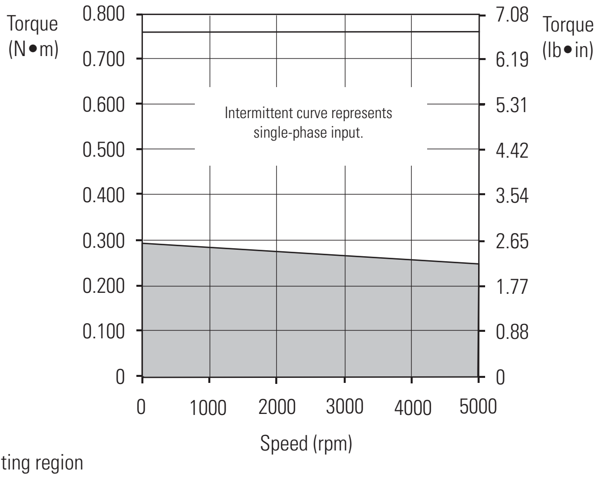
2071-AP1 and TLY-A120P-B (Brake)



2071-AP1 and TLY-A130P-B



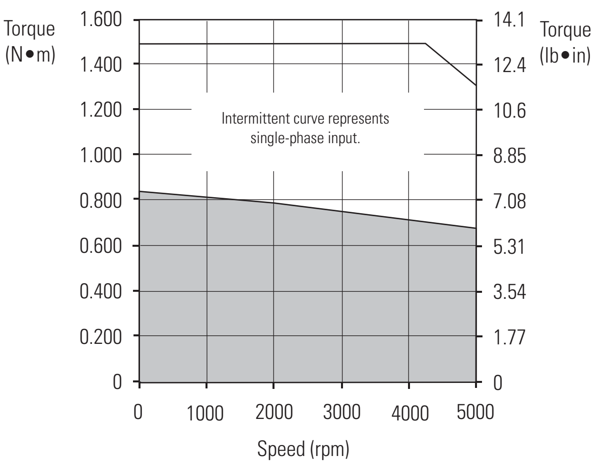
2071-AP1 and TLY-A130P-B (Brake)

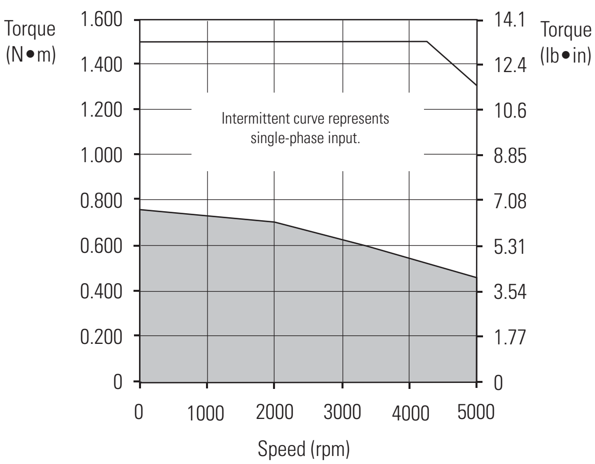


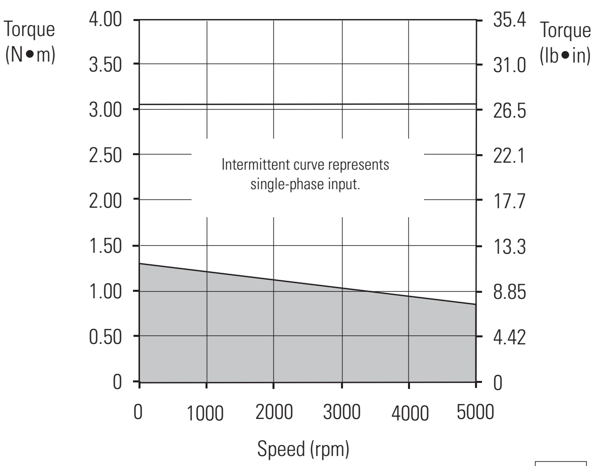
☐ = Intermittent operating region

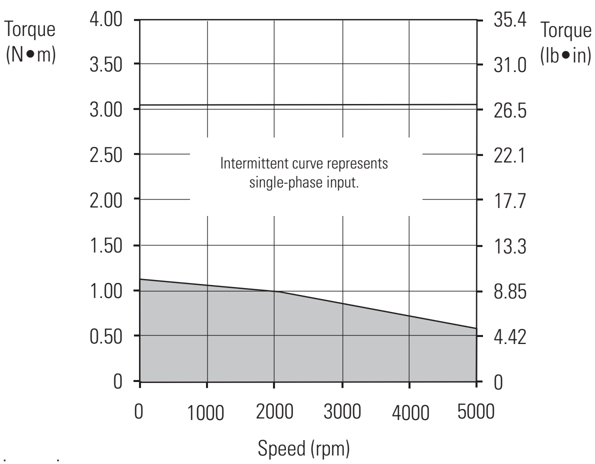
▨ = Continuous operating region

Kinetix 3 (240V) Drives/TLY-A*xxx*P-B (absolute high-resolution) Motor Curves, Continued


2071-AP4 and TLY-A220P-B


2071-AP4 and TLY-A220P-B (Brake)


2071-AP4 and TLY-A230P-B

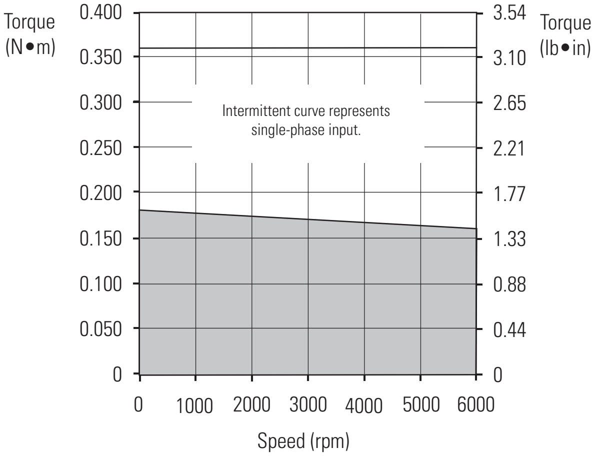

2071-AP4 and TLY-A230P-B (Brake)

□ = Intermittent operating region

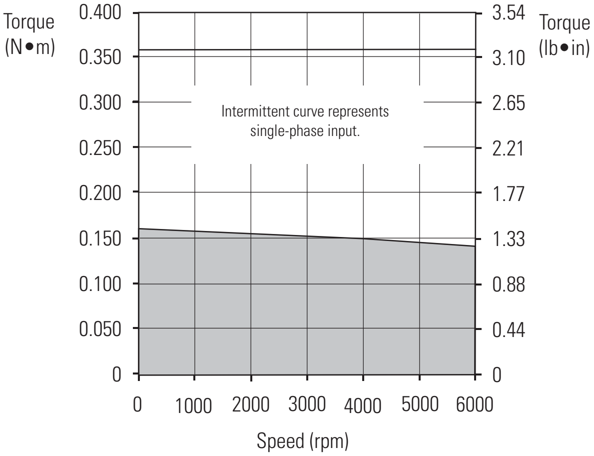
▨ = Continuous operating region

Kinetix 3 (240V) Drives/TLY-A*xxx*T-H (incremental) Motor Curves

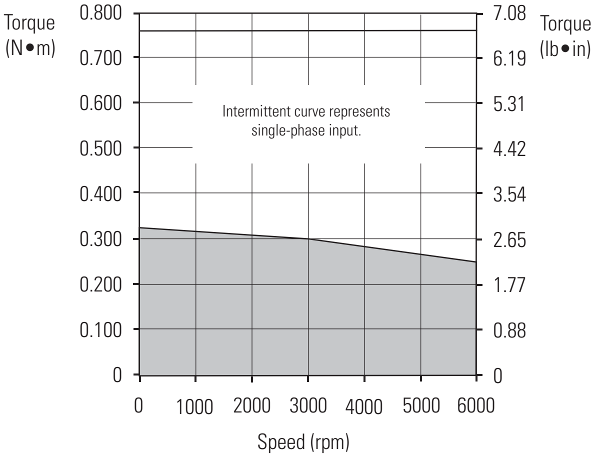
2071-AP1 and TLY-A120T-H



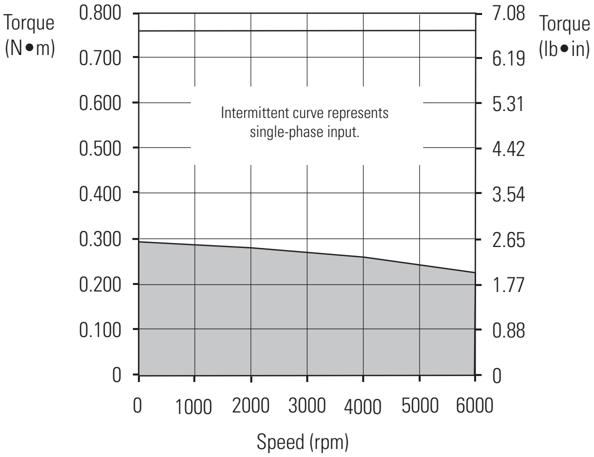
2071-AP1 and TLY-A120T-H (Brake)



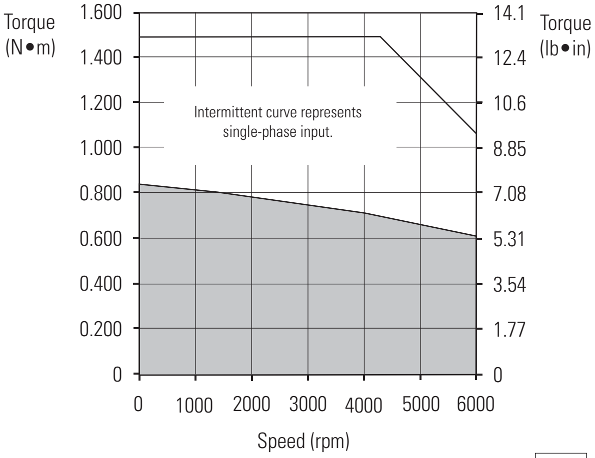
2071-AP1 and TLY-A130T-H



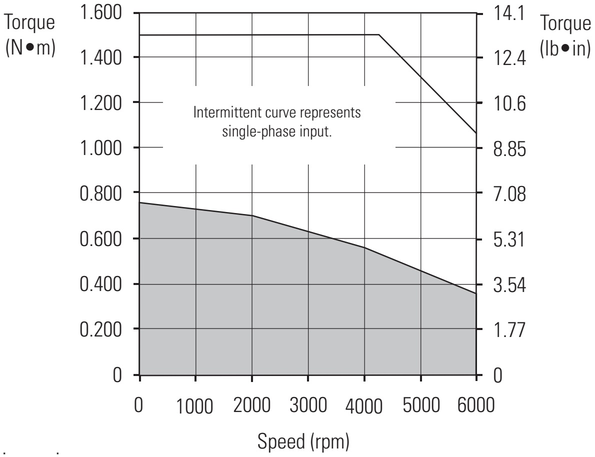
2071-AP1 and TLY-A130T-H (Brake)



2071-AP4 and TLY-A220T-H



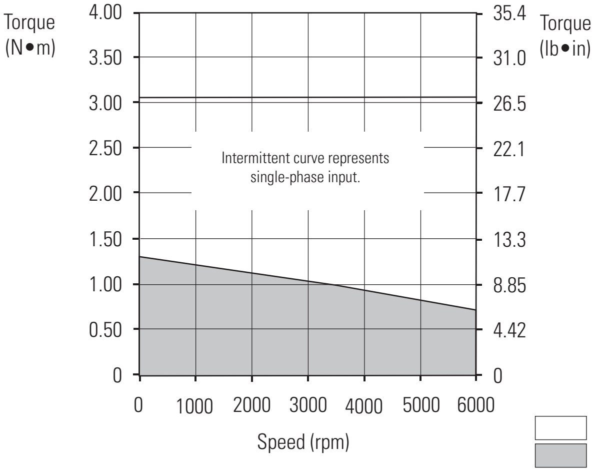
2071-AP4 and TLY-A220T-H (Brake)



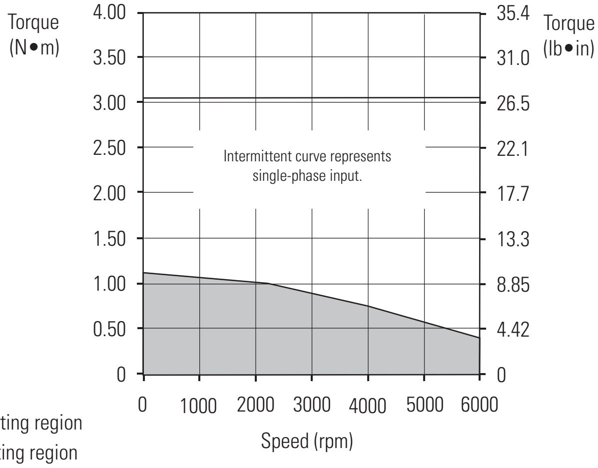
☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 3 (240V) Drives/TLY-A*xxx*T-H (incremental) Motor Curves, Continued

2071-AP4 and TLY-A230T-H



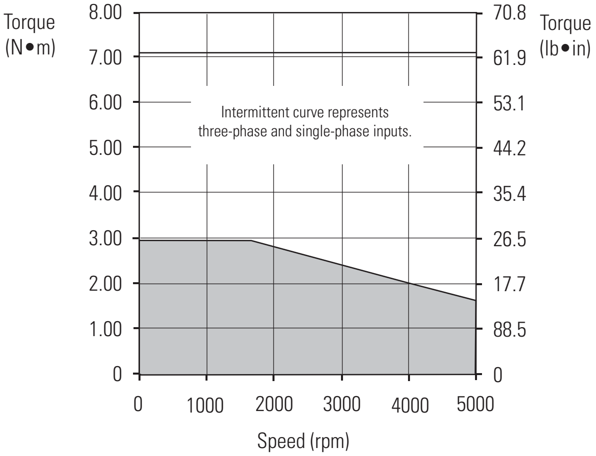
2071-AP4 and TLY-A230T-H (Brake)



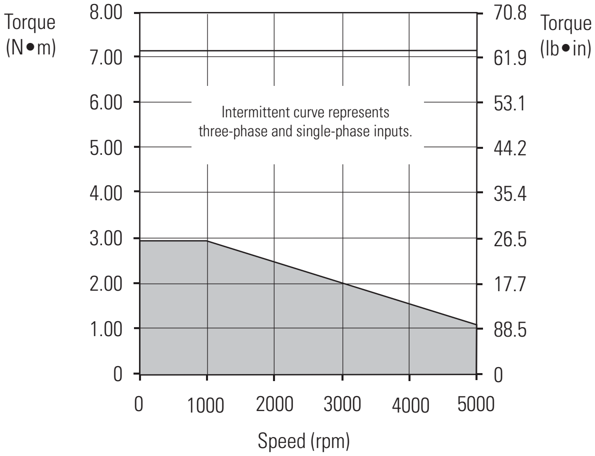
☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 3 (240V) Drives/TLY-A*xxxx*-*x* Motor Curves

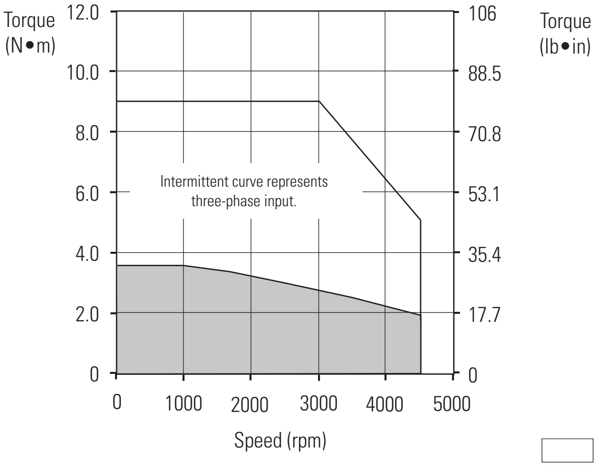
2071-AP8 and TLY-A2540P-*x*



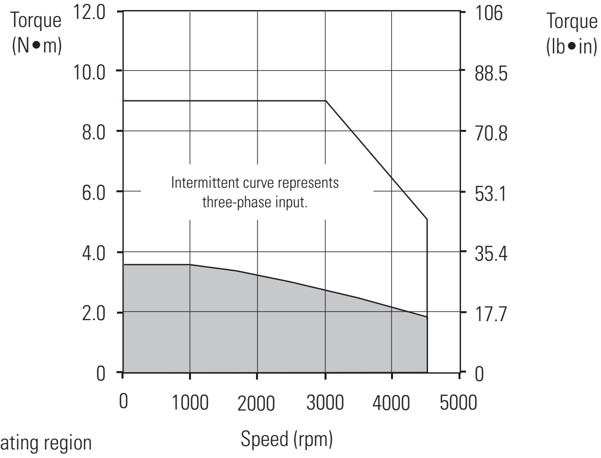
2071-AP8 and TLY-A2540P-*x* (Brake)



2071-A10 and TLY-A310M-*x*



2071-A10 and TLY-A310M-*x* (Brake)



☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 3 (240V) Drives with TL-Series (Bulletin TL) Low Inertia Motors

This section provides system combination information for the Kinetix 3 servo drives when matched with TL-Series (Bulletin TL) low-inertia motors. Bulletin TL-A*xxx* motors are equipped with absolute high-resolution encoder feedback and are characterized as having 300 mm (11.8 in.) cable extensions with rectangular connectors. Included in this section are motor power, feedback, and brake cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin TL Motor Cable Combinations

Motor Cat. No.	Motor Power Cable	Motor Feedback Cable (1)	Motor Brake Cable
TL-A120P, TL-A130P	2090-DANPT-16S*xx*	2090-DANFCT-S*xx* Absolute High-resolution	2090-DANBT-18S*xx*
TL-A220P, TL-A230P			
TL-A2540P			
TL-A410M			

(1) TL-Series (Bulletin TL) rotary motors require the 2071-TBMF breakout board with 3.6V lithium battery (not included) for multi-turn high-resolution encoder operation or when using an overtravel limit switch. To meet this requirement, remove the drive-end connector and wire the 2090-DANFCT-S*xx* cable to the 2071-TBMF breakout board.

Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

TL-Series (non-brake) Performance Specifications with Kinetix 3 (240V) Drives

Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 3 Drives
TL-A120P	5000	1.03	0.181 (1.60)	2.50	0.36 (3.20)	0.086	2071-AP1
TL-A130P		1.85	0.325 (2.88)	4.90	0.76 (6.70)	0.14	2071-AP1
TL-A220P		3.50	0.836 (7.40)	7.90	1.48 (13.1)	0.35	2071-AP4
TL-A230P		5.50	1.30 (11.5)	15.5	3.05 (27.0)	0.44	2071-AP4
TL-A2540P		10.0	2.94 (26.0)	24.8	7.10 (63.0)	0.86	2071-AP8
TL-A410M	4500	15.5	5.42 (48.0)	43.4	13.0 (115.0)	2.0	2071-A15

Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 5.000 or later.

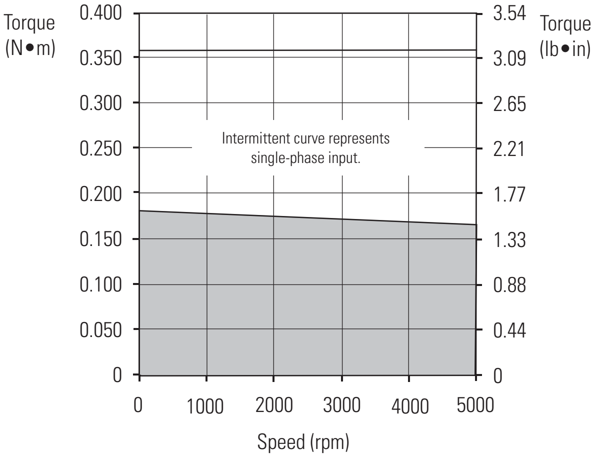
Bulletin TL (brake) Performance Specifications with Kinetix 3 (240V) Drives

Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Torque N•m (lb•in)	Motor Rated Output kW	Kinetix 3 Drives
TL-A120P	5000	0.93	0.163 (1.44)	2.50	0.36 (3.20)	0.077	2071-AP1
TL-A130P		1.67	0.293 (2.59)	4.90	0.76 (6.70)	0.13	2071-AP1
TL-A220P		3.15	0.757 (6.70)	7.90	1.48 (13.10)	0.24	2071-AP4
TL-A230P		4.95	1.160 (10.30)	15.5	3.05 (27.0)	0.32	2071-AP4
TL-A2540P		10.0	2.940 (26.00)	24.8	7.10 (63.0)	0.66	2071-AP8
TL-A410M	4500	14.0	4.860 (43.0)	43.4	13.0 (115.0)	1.80	2071-A15

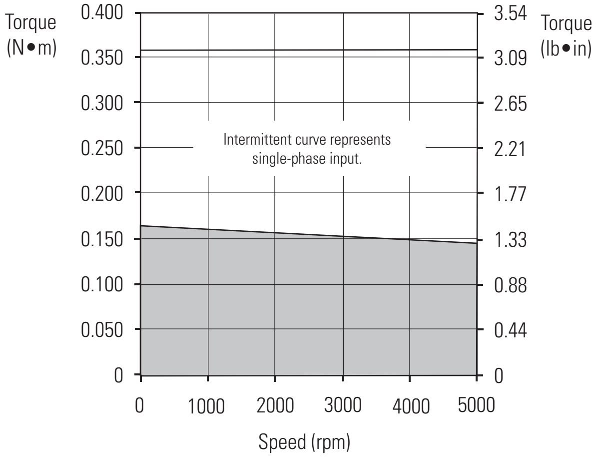
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 5.000 or later.

Kinetix 3 (240V) Drives/TL-A*xxxx*-B (absolute high-resolution) Motor Curves

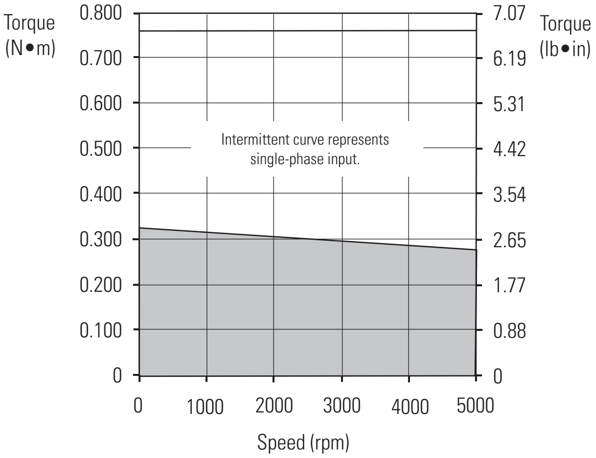
2071-AP1 and TL-A120P-B



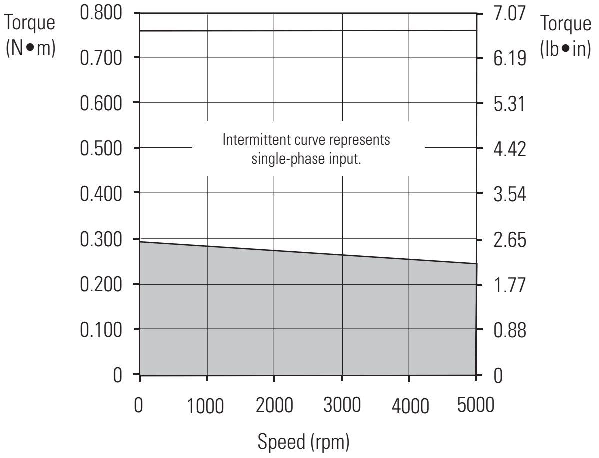
2071-AP1 and TL-A120P-B (Brake)



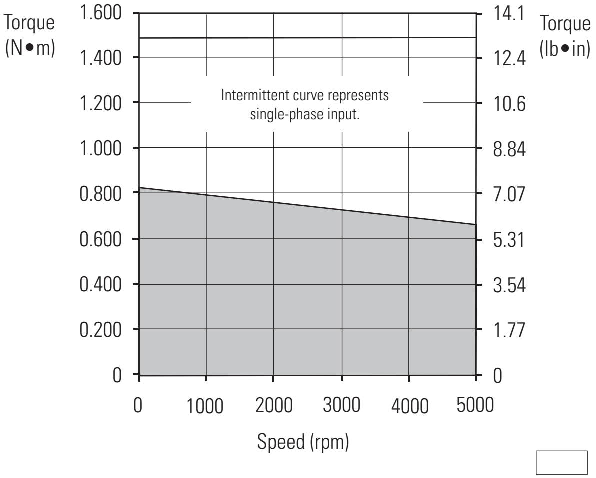
2071-AP1 and TL-A130P-B



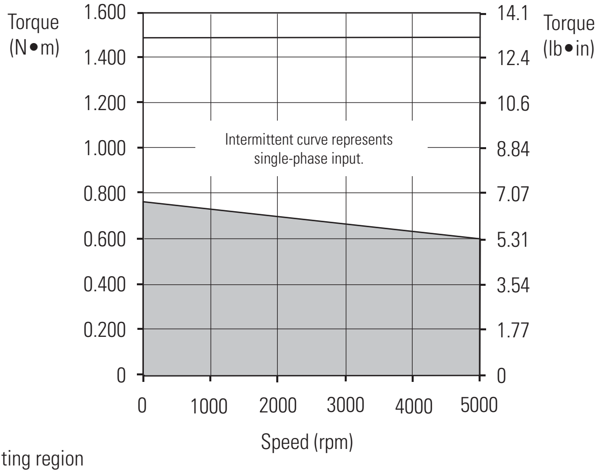
2071-AP1 and TL-A130P-B (Brake)



2071-AP4 and TL-A220P-B



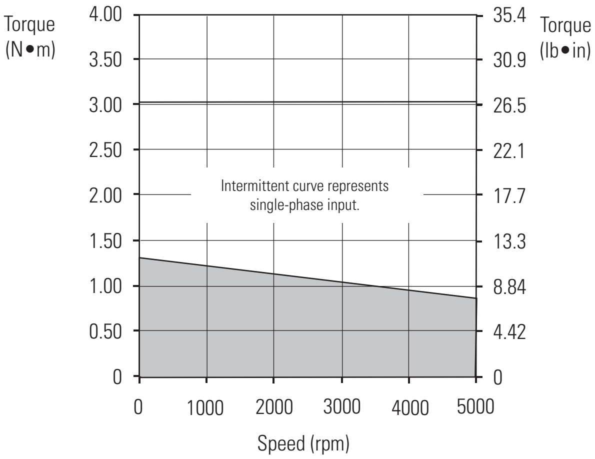
2071-AP4 and TL-A220P-B (Brake)

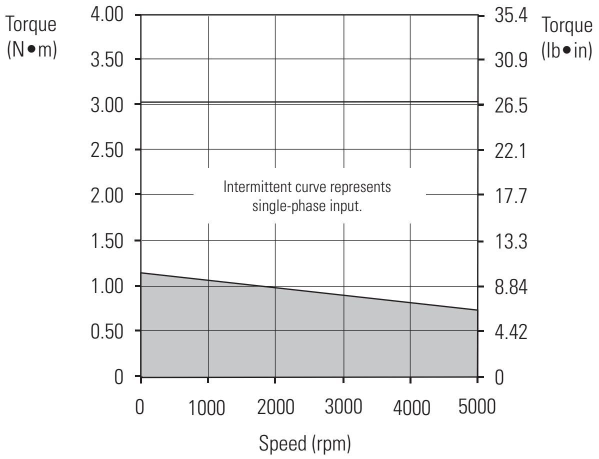


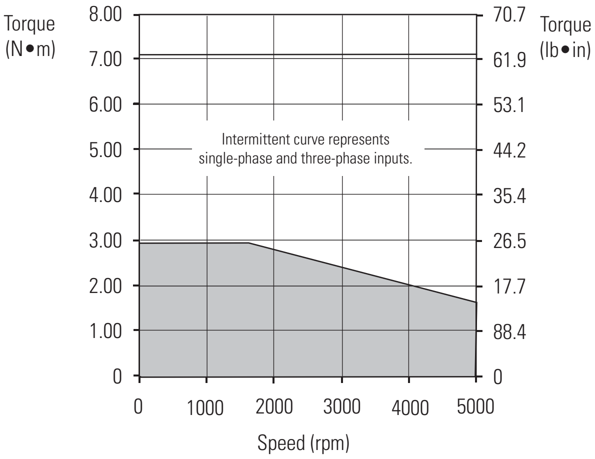
☐ = Intermittent operating region

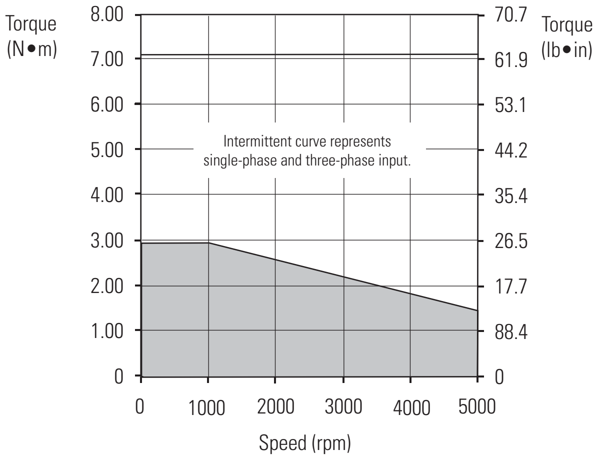
▨ = Continuous operating region

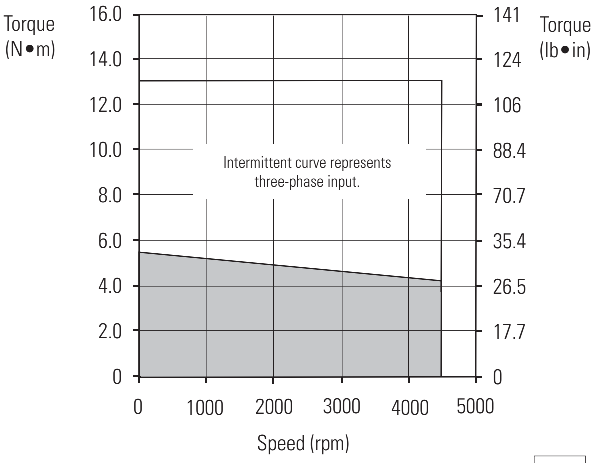
Kinetix 3 (240V) Drives/TL-A*xxxx*-B (absolute high-resolution) Motor Curves, Continued

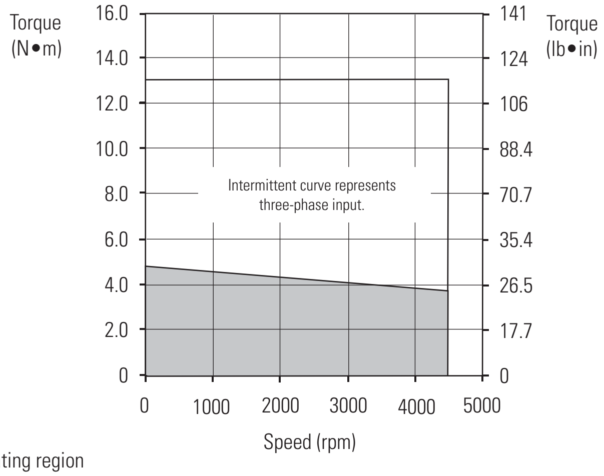

2071-AP4 and TL-A230P-B


2071-AP4 and TL-A230P-B (Brake)


2071-AP8 and TL-A2540P-B


2071-AP8 and TL-A2540P-B (Brake)


2071-A15 and TL-A410M-B


2071-A15 and TL-A410M-B (Brake)

☐ = Intermittent operating region

▨ = Continuous operating region

Ultra3000/5000 (230V) Drives with MP-Series Low Inertia Motors

This section provides system combination information for the Ultra3000/5000 (230V) drives when matched with MP-Series low-inertia motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

IMPORTANT The MP-Series low-inertia motors on this page are equipped with DIN connectors (specified by 4 or 7 in the catalog number) and are not compatible with cables designed for motors equipped with bayonet connectors (specified by 2 in the catalog number). The motors with bayonet connectors (for example, MPL-A310P-*xx*2*x*AA) are being discontinued and require 2090-XXN*x*MP (bayonet) cables. For help with migration or to select bayonet cables, contact your Rockwell Automation sales representative.

Bulletin MPL Motor Cable Combinations

Motor Cat. No. (200V class) [1]	Motor Power/Brake Cable	Motor Feedback Cable [2]
MPL-A1510V-*xx*4*x*AA, MPL-A1520U-*xx*4*x*AA, MPL-A1530U-*xx*4*x*AA	2090-XXNPMF-16S*xx* (standard) 2090-CP*x*M4DF-16AF*xx* (continuous-flex)	2090-XXNFMF-S*xx* (standard) 2090-CFBM4DF-CDAF*xx* (continuous-flex) Absolute High-resolution and Incremental Feedback
MPL-A210V-*xx*4*x*AA, MPL-A220T-*xx*4*x*AA, MPL-A230P-*xx*4*x*AA		
MPL-A310F-*xx*7*x*AA, MPL-A310P-*xx*7*x*AA, MPL-A320H-*xx*7*x*AA, MPL-A320P-*xx*7*x*AA, MPL-A330P-*xx*7*x*AA	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or [3] 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback 2090-XXNFMF-S*xx* (standard) [4] 2090-CFBM7DF-CDAF*xx* (continuous-flex) Incremental Feedback
MPL-A420P-*xx*7*x*AA, MPL-A430H-*xx*7*x*AA		
MPL-A4530F-*xx*7*x*AA, MPL-A4540C-*xx*7*x*AA		
MPL-A430P-*xx*7*x*AA	2090-CP*x*M7DF-14AA*xx* (standard) 2090-CP*x*M7DF-14AF*xx* (continuous-flex)	
MPL-A4530K-*xx*7*x*AA, MPL-A4540F-*xx*7*x*AA, MPL-A4560F-*xx*7*x*AA		
MPL-A520K-*xx*7*x*AA	2090-CP*x*M7DF-10AA*xx* (standard) 2090-CP*x*M7DF-10AF*xx* (continuous-flex)	
MPL-A540K-*xx*7*x*AA, MPL-A560F-*xx*7*x*AA	2090-CP*x*M7DF-08AA*xx* (standard) 2090-CP*x*M7DF-08AF*xx* (continuous-flex)	

(1) MPL-A15*xx* and MPL-A2*xx* motors are not compatible with Ultra5000 drives.

(2) Use drive-mounted breakout board (catalog number 2090-UXBB-DM15) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

(3) Applies to Ultra3000/5000 drives and MPL-A3*xxx*-M/S…MPL-A5*xxx*-M/S motors with absolute high-resolution feedback.

(4) Applies to Ultra3000/5000 drives and MPL-A3*xxx*-H…MPL-A45*xxx*-H motors with incremental feedback.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Bulletin MPL Motor Performance Specifications with Ultra3000/5000 (230V) Drives

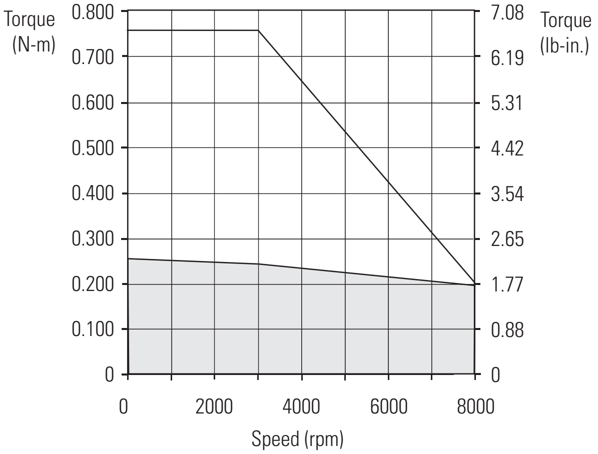
Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Ultra3000/ Ultra5000 230V Drives
MPL-A1510V	8000	1.05	0.26 (2.3)	3.40	0.77 (6.8)	0.16	2098-DSD-005
MPL-A1520U	7000	1.80	0.49 (4.3)	6.10	1.58 (13.9)	0.27	2098-DSD-005
MPL-A1530U	7000	2.82	0.90 (8.0)	10.1	2.82 (24.9)	0.39	2098-DSD-010
MPL-A210V	8000	3.09	0.55 (4.8)	10.2	1.52 (13.5)	0.37	2098-DSD-010
MPL-A220T	6000	4.54	1.61 (14.2)	15.5	4.74 (41.9)	0.62	2098-DSD-020

Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Stall Torque** N•m (lb•in)	**Motor Rated Output** kW	**Ultra3000/ Ultra5000 230V Drives**
MPL-A230P	5000	5.40	2.10 (18)	23.0	8.2 (72.5)	0.86	2098-DSD-020
MPL-A310F	3000	2.50	1.24 (11)	7.5	2.94 (26)	0.46	2098-*xxx*-005
		3.20	1.58 (14)	9.3	3.61 (32)		2098-*xxx*-010
MPL-A310P	5000	2.50	0.79 (6.9)	7.5	1.92 (17)	0.73	2098-*xxx*-005
		4.85	1.58 (14)	14	3.61 (32)		2098-*xxx*-010
MPL-A320H	3500	5.0	2.48 (22)	15	6.44 (57)	1.0	2098-*xxx*-010
		6.1	3.05 (27)	19.3	7.91 (70)		2098-*xxx*-020
MPL-A320P	5000	5.0	1.69 (15)	15	3.95 (35)	1.3	2098-*xxx*-010
		9.0	3.05 (27)	29.5	7.91 (70)		2098-*xxx*-020
MPL-A330P	5000	12.0	4.18 (37)	30	9.60 (85)	1.8	2098-*xxx*-030
				38	11.1 (98)		2098-*xxx*-075
MPL-A420P	5000	12.7	4.74 (42)	30	10.2 (90)	2.0	2098-*xxx*-030
				46	13.5 (120)		2098-*xxx*-075
MPL-A430H	3500	12.2	6.21 (55)	30	14.7 (130)	1.8	2098-*xxx*-030
				45	19.8 (175)		2098-*xxx*-075
MPL-A430P	5000	15.0	5.42 (48)	30	10.2 (90)	2.2	2098-*xxx*-030
		16.8	5.99 (53)	67	19.8 (175)		2098-*xxx*-075
MPL-A4530F	2800	13.4	8.36 (74)	30	17.5 (155)	1.9	2098-*xxx*-030
				42	20.3 (180)		2098-*xxx*-075
MPL-A4530K	4000	15.0	6.21 (55)	30	11.3 (100)	2.5	2098-*xxx*-030
		19.5	8.13 (72)	62	20.3 (180)		2098-*xxx*-075
MPL-A4540C	1500	9.4	10.2 (90)	29	27.1 (240)	1.5	2098-*xxx*-020
MPL-A4540F	3000	15.0	8.25 (73)	30	15.8 (140)	2.6	2098-*xxx*-030
		18.4	10.2 (90)	58	27.1 (240)		2098-*xxx*-075
MPL-A4560F	3000	22.0	14.1 (125)	66.0	34.4 (305)	3.0	2098-*xxx*-075
MPL-A520K	4000	23.3	10.7 (95.0)	65	24.3 (215)	3.5	2098-*xxx*-075
MPL-A540K	4000	41.5	19.4 (172)	120	48.6 (430)	5.5	2098-*xxx*-150
MPL-A560F	3000	42.0	26.8 (237)		61.0 (540)	5.3	2098-*xxx*-150

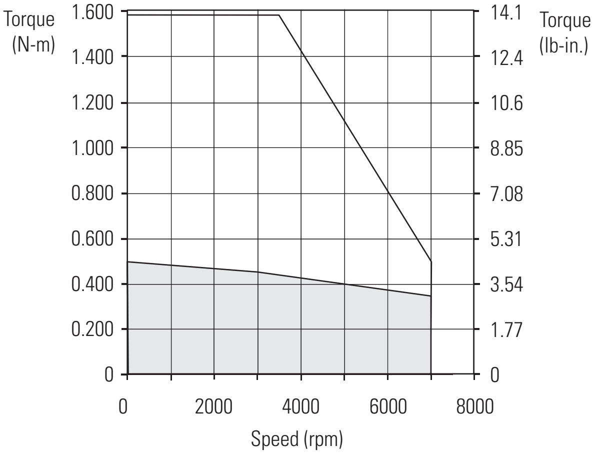
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Ultra3000 (230V) Drives/MP-Series Low Inertia Motor Curves

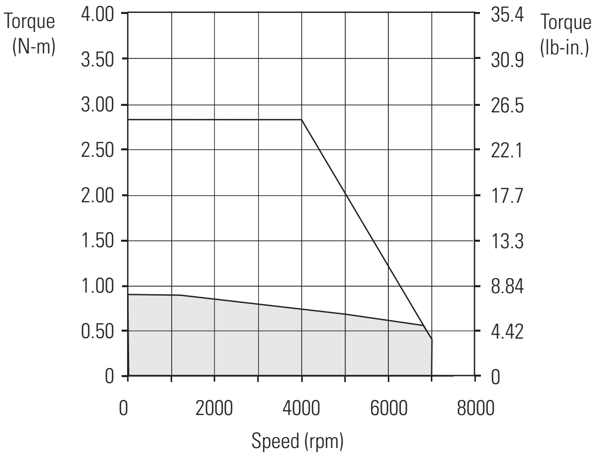
2098-DSD-005 and MPL-A1510V



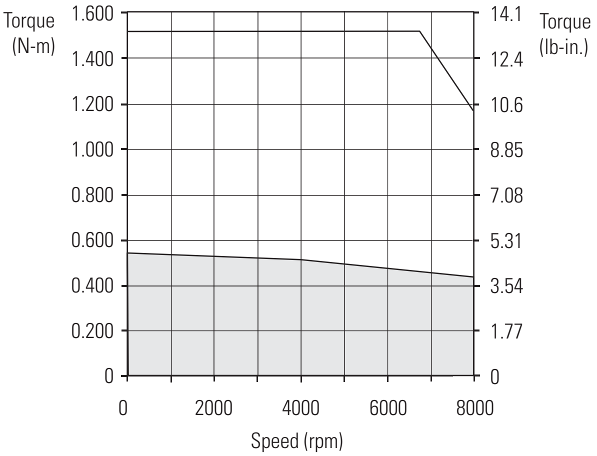
2098-DSD-005 and MPL-A1520U



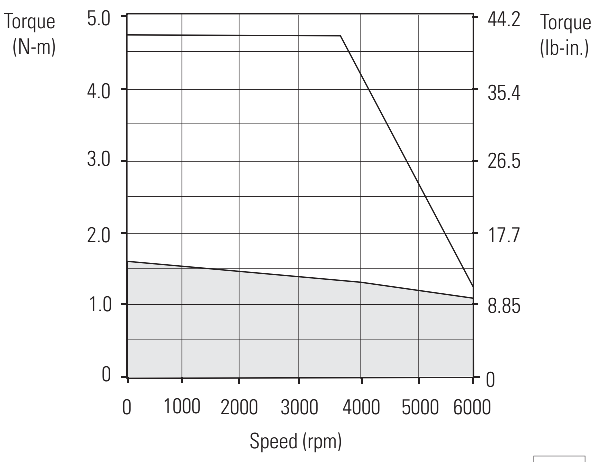
2098-DSD-010 and MPL-A1530U



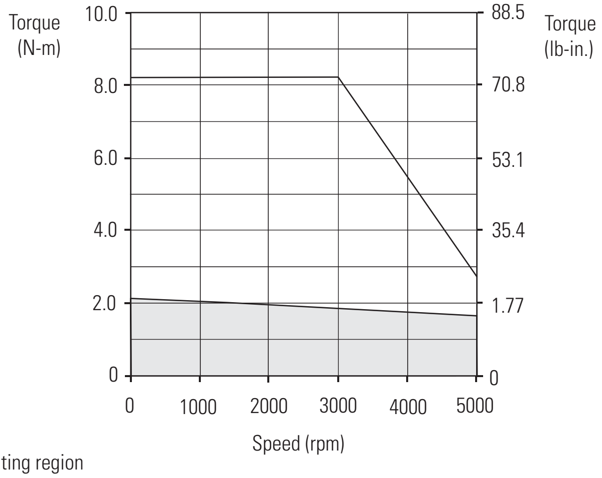
2098-DSD-010 and MPL-A210V



2098-DSD-020 and MPL-A220T



2098-DSD-020 and MPL-A230P

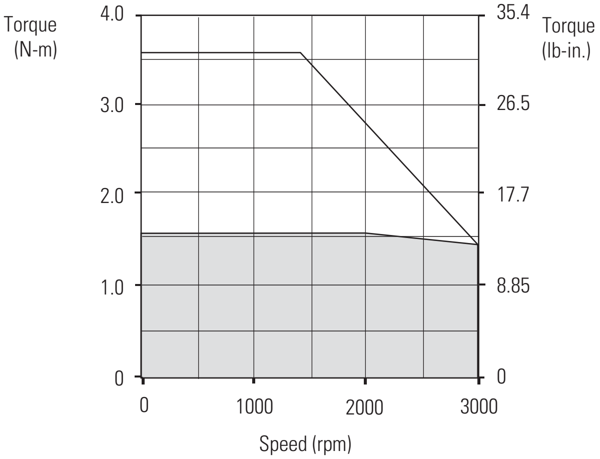


☐ = Intermittent operating region

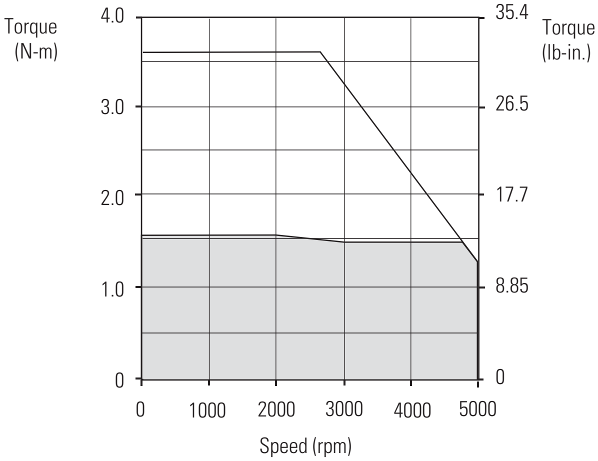
▨ = Continuous operating region

Ultra3000/5000 (230V) Drives/MP-Series Low Inertia Motor Curves, Continued

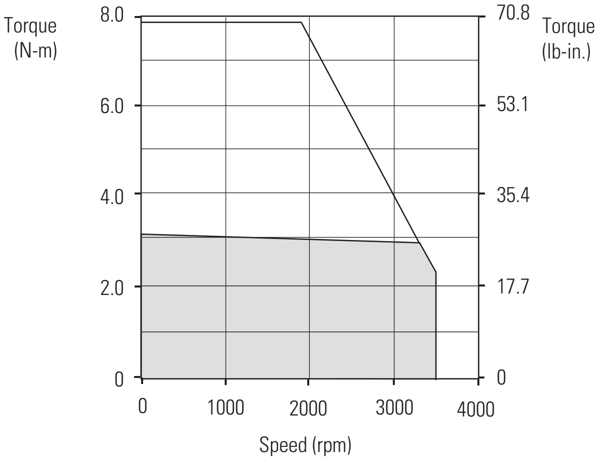
2098-*xxx*-010 and MPL-A310F



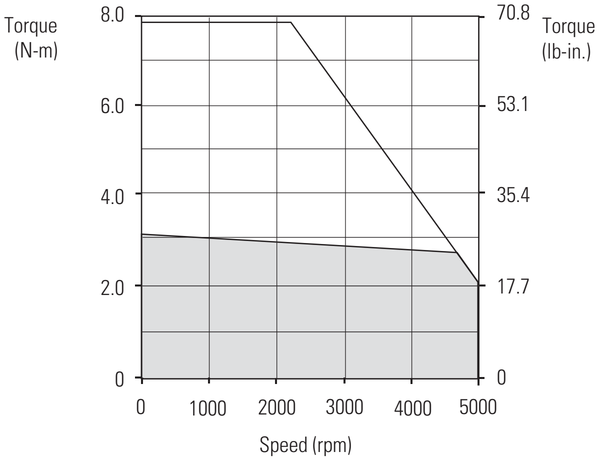
2098-*xxx*-010 and MPL-A310P



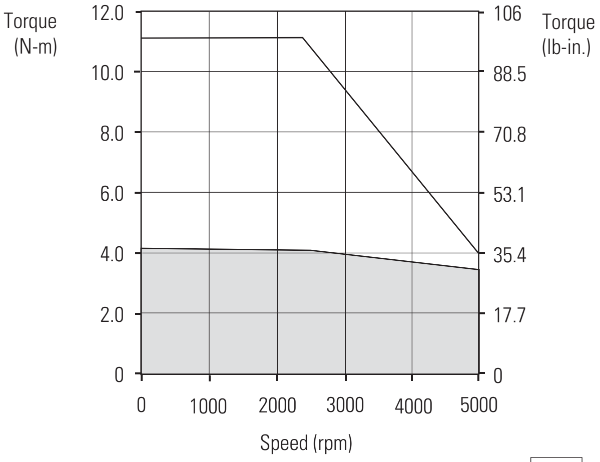
2098-*xxx*-020 and MPL-A320H



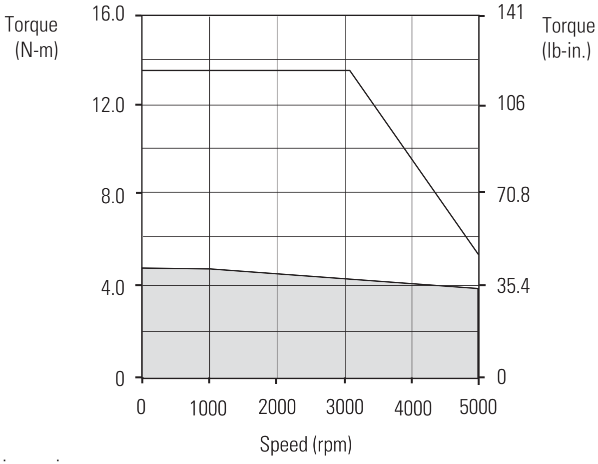
2098-*xxx*-020 and MPL-A320P



2098-*xxx*-075 and MPL-A330P



2098-*xxx*-075 and MPL-A420P

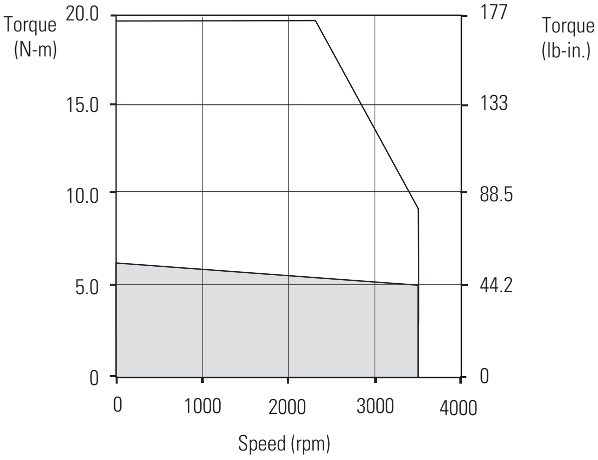


☐ = Intermittent operating region

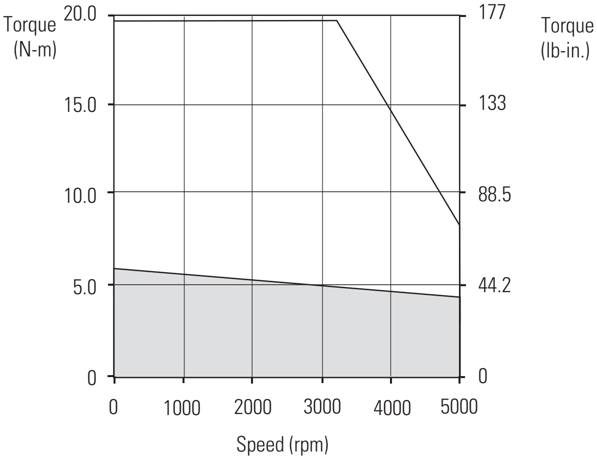
▨ = Continuous operating region

Ultra3000/5000 (230V) Drives/MP-Series Low Inertia Motor Curves, Continued

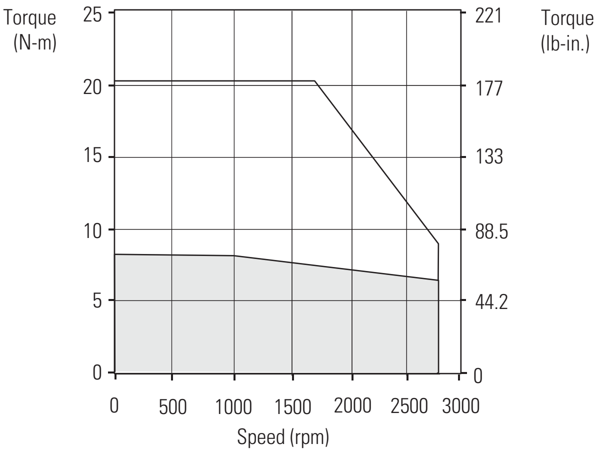
2098-*xxx*-075 and MPL-A430H



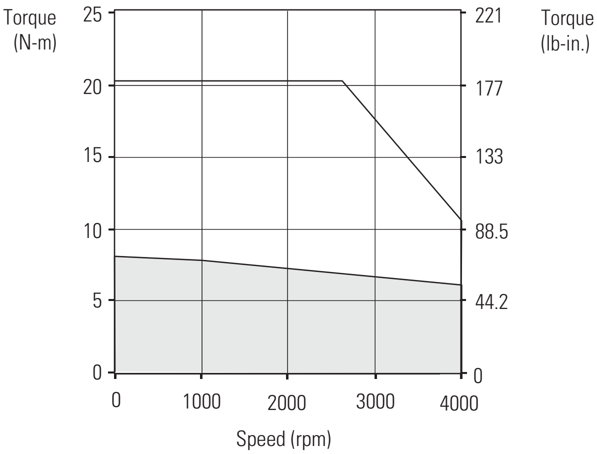
2098-*xxx*-075 and MPL-A430P



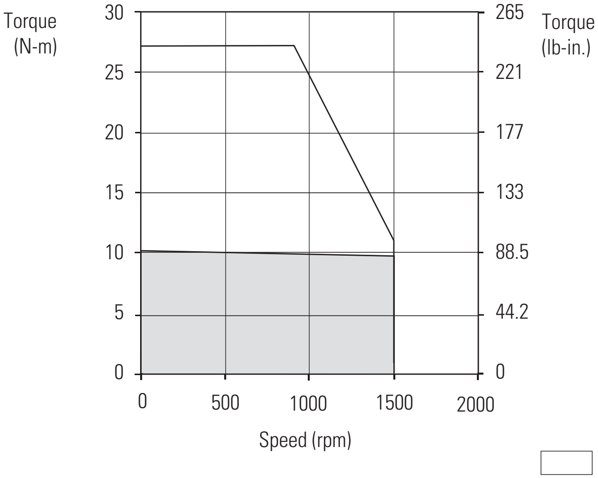
2098-*xxx*-075 and MPL-A4530F



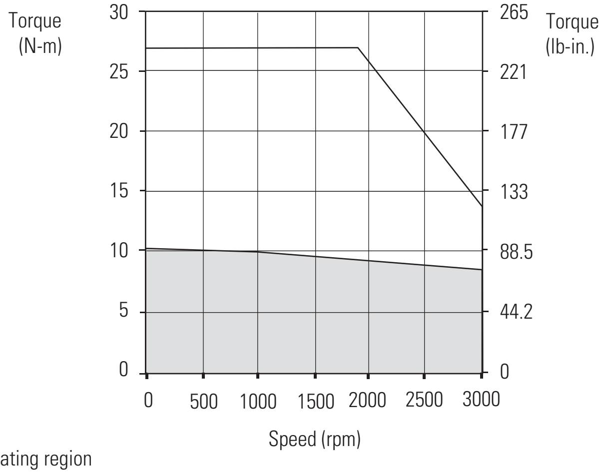
2098-*xxx*-075 and MPL-A4530K



2098-*xxx*-020 and MPL-A4540C



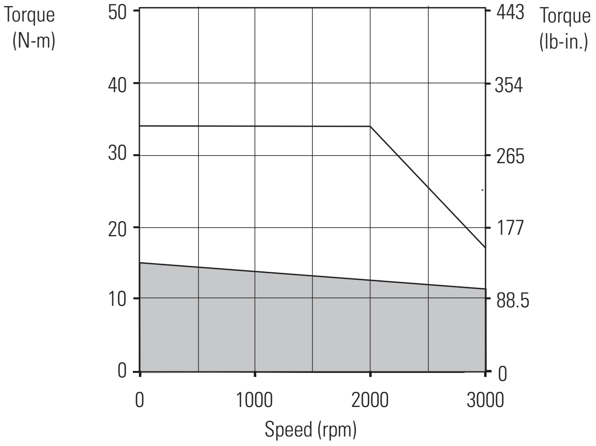
2098-*xxx*-075 and MPL-A4540F



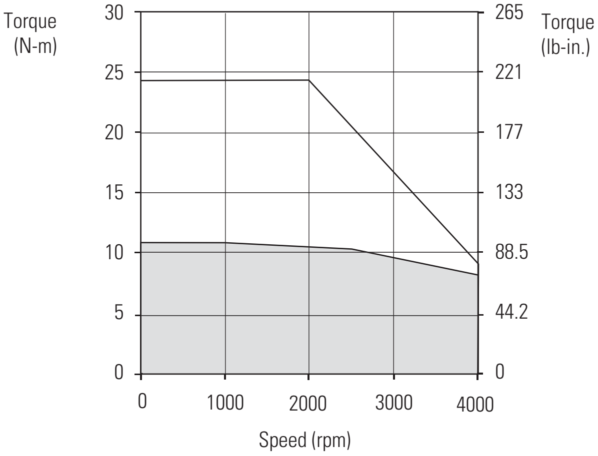
☐ = Intermittent operating region
▨ = Continuous operating region

Ultra3000/5000 (230V) Drives/MP-Series Low Inertia Motor Curves, Continued

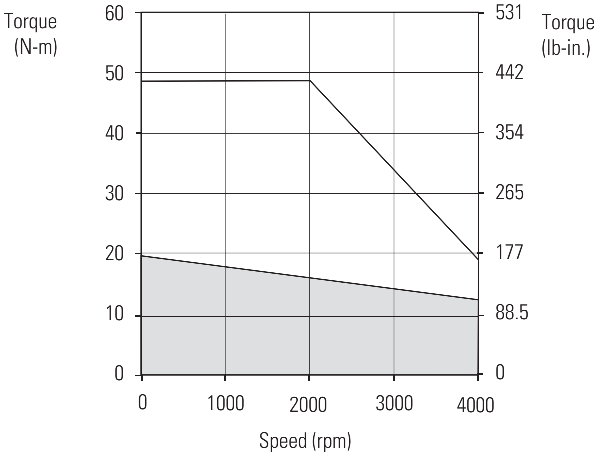
2098-*xxx*-075 and MPL-A4560F



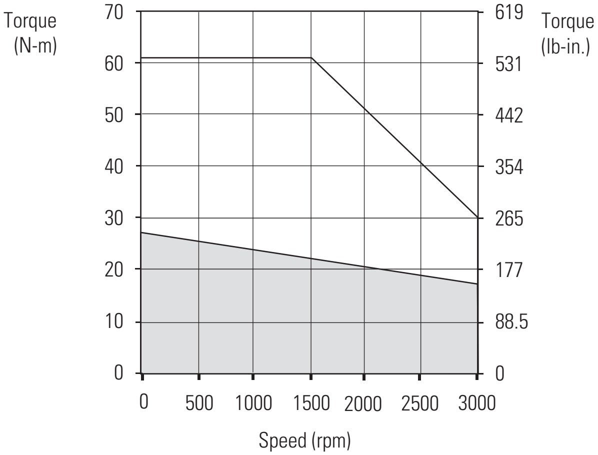
2098-*xxx*-150 and MPL-A520K



2098-*xxx*-150 and MPL-A540K



2098-*xxx*-150 and MPL-A560F



☐ = Intermittent operating region

▨ = Continuous operating region

Ultra3000/5000 (460V) Drives with MP-Series Low Inertia Motors

This section provides system combination information for the Ultra3000/5000 (460V) drives when matched with MP-Series low-inertia motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

IMPORTANT The MP-Series low-inertia motors on this page are equipped with DIN connectors (specified by 4 or 7 in the catalog number) and are not compatible with cables designed for motors equipped with bayonet connectors (specified by 2 in the catalog number). The motors with bayonet connectors (for example, MPL-A310P-*xx*2*x*AA) are being discontinued and require 2090-XXN*x*MP (bayonet) cables. For help with migration or to select bayonet cables, contact your Rockwell Automation sales representative.

Bulletin MPL Motor Cable Combinations

Motor Cat. No. (400V class) (1)	Motor Power/Brake Cable	Motor Feedback Cable (2)
MPL-B1510V-*xx*4*x*AA, MPL-B1520U-*xx*4*x*AA, MPL-B1530U-*xx*4*x*AA	2090-XXNPMF-16S*xx* (standard) 2090-CP*x*M4DF-16AF*xx* (continuous-flex)	2090-XXNFMF-S*xx* (standard) 2090-CFBM4DF-CDAF*xx* (continuous-flex) Absolute High-resolution and Incremental Feedback
MPL-B210V-*xx*4*x*AA, MPL-B220T-*xx*4*x*AA, MPL-B230P-*xx*4*x*AA		
MPL-B310P-*xx*7*x*AA, MPL-B320P-*xx*7*x*AA, MPL-B330P-*xx*7*x*AA	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or (3) 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback 2090-XXNFMF-S*xx* (standard) (4) 2090-CFBM7DF-CDAF*xx* (continuous-flex) Incremental Feedback
MPL-B420P-*xx*7*x*AA, MPL-B430P-*xx*7*x*AA		
MPL-B4530F-*xx*7*x*AA, MPL-B4530K-*xx*7*x*AA, MPL-B4540F-*xx*7*x*AA, MPL-B4560F-*xx*7*x*AA		
MPL-B520K-*xx*7*x*AA		
MPL-B540D-*xx*7*x*AA, MPL-B540K-*xx*7*x*AA, MPL-B560F-*xx*7*x*AA	2090-CP*x*M7DF-14AA*xx* (standard) 2090-CP*x*M7DF-14AF*xx* (continuous-flex)	
MPL-B580F-*xx*7*x*AA, MPL-B580J-*xx*7*x*AA, MPL-B640F-*xx*7*x*AA	2090-CP*x*M7DF-10AA*xx* (standard) 2090-CP*x*M7DF-10AF*xx* (continuous-flex)	
MPL-B660F-*xx*7*x*AA, MPL-B680D-*xx*7*x*AA, MPL-B960B-*xx*7*x*AA, MPL-B980B-*xx*7*x*AA	2090-CP*x*M7DF-08AA*xx* (standard) 2090-CP*x*M7DF-08AF*xx* (continuous-flex)	
MPL-B680F-*xx*7*x*AA, MPL-B860D-*xx*7*x*AA, MPL-B880C-*xx*7*x*AA,	2090-CPBM7DF-06AA*xx* (standard)	
MPL-B880D-*xx*7*x*AA, MPL-B960C-*xx*7*x*AA, MPL-B980C-*xx*7*x*AA,	2090-CPBM7DF-04AA*xx* (standard)	

(1) MPL-B15*xx* and MPL-B2*xx* motors are not compatible with Ultra5000 drives.

(2) Use drive-mounted breakout board (catalog number 2090-UXBB-DM15) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

(3) Applies to Ultra3000/5000 drives and MPL-B3*xxx*-M/S…MPL-B9*xxx*-M/S motors with absolute high-resolution feedback.

(4) Applies to Ultra3000/5000 drives and MPL-B3*xxx*-H…MPL-B45*xxx*-H motors with incremental feedback.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Bulletin MPL Motor Performance Specifications with Ultra3000/5000 (460V) Drives

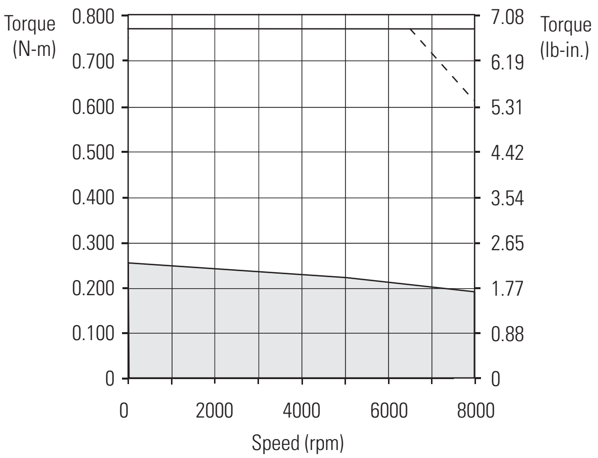
Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A (0-pk)	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A (0-pk)	**System Peak Stall Torque** N•m (lb•in)	**Motor Rated Output** kW	**Ultra3000/ Ultra5000 460V Drives**
MPL-B1510V	8000	0.95	0.26 (2.3)	3.10	0.77 (6.80)	0.16	2098-DSD-HV030
MPL-B1520U	7000	1.80	0.49 (4.3)	6.10	1.58 (13.9)	0.27	2098-DSD-HV030
MPL-B1530U	7000	2.0	0.90 (8.0)	7.20	2.82 (24.9)	0.39	2098-DSD-HV030
MPL-B210V	8000	1.75	0.55 (4.8)	5.80	1.52 (13.5)	0.37	2098-DSD-HV030
MPL-B220T	6000	3.30	1.61 (14.2)	11.3	4.74 (41.9)	0.62	2098-DSD-HV030
MPL-B230P	5000	2.60	2.10 (18.6)	11.3	8.20 (73.0)	0.86	2098-DSD-HV030
MPL-B310P	5000	2.4	1.58 (14)	7.1	3.61 (32)	0.77	2098-*xxx*-HV030
MPL-B320P	5000	4.5	2.94 (26)	13.0	7.91 (70)	1.5	2098-*xxx*-HV030
MPL-B330P	5000	6.1	4.18 (37)	14.0	8.59 (76)	1.8	2098-*xxx*-HV030
				17.0	11.1 (98)		2098-*xxx*-HV050
MPL-B420P	5000	6.4	4.74 (42)	14.0	8.59 (76)	1.9	2098-*xxx*-HV030
				22.0	12.9 (114)		2098-*xxx*-HV050
				23.0	13.5 (120)		2098-*xxx*-HV100
MPL-B430P	5000	9.2	6.55 (58)	22.0	12.9 (114)	2.2	2098-*xxx*-HV050
				31.0	19.8 (175)		2098-*xxx*-HV100
MPL-B4530F	3000	7.0	8.25 (73)	14.0	13.5 (120)	2.1	2098-*xxx*-HV030
		7.1	8.36 (74)	21.0	20.3 (180)		2098-*xxx*-HV050
MPL-B4530K	4000	11.0	8.36 (74)	22.0	14.5 (128)	2.6	2098-*xxx*-HV050
				31.0	20.3 (180)		2098-*xxx*-HV100
MPL-B4540F	3000	9.1	10.2 (90)	22.0	22.0 (195)	2.6	2098-*xxx*-HV050
				26.0	27.1 (240)		2098-*xxx*-HV100
MPL-B4560F	3000	11.0	13.1 (116)	22.0	21.0 (186)	3.2	2098-*xxx*-HV050
		11.8	14.1 (125)	36.0	34.4 (305)		2098-*xxx*-HV100
MPL-B520K	4000	11.0	10.3 (91)	22.0	15.8 (140)	3.5	2098-*xxx*-HV050
		11.5	10.7 (95)	33.0	23.2 (205)		2098-*xxx*-HV100
MPL-B540D	2000	10.5	19.4 (172)	22.0	39.2 (346)	3.4	2098-*xxx*-HV050
				23.0	41.0 (362)		2098-*xxx*-HV100
MPL-B540K	4000	20.5	19.4 (172)	46.0	33.9 (300)	5.4	2098-*xxx*-HV100
				60.0	45.2 (400)		2098-*xxx*-HV150
MPL-B560F	3000	20.6	26.8 (237)	46.0	50.4 (446)	5.5	2098-*xxx*-HV100
				68.0	67.8 (600)		2098-*xxx*-HV150
MPL-B580F	3000	26.0	34.0 (301)	68.0	70.5 (623)	7.1	2098-*xxx*-HV150
				94.0	87.0 (769)		2098-*xxx*-HV220
MPL-B580J	3800	32.0	34.0 (301)	68.0	62.4 (552)	7.9	2098-*xxx*-HV150
				94.0	81.0 (717)		2098-*xxx*-HV220
MPL-B640F	3000	32.1	36.7 (325)	65.0	72.3 (640)	6.1	2098-*xxx*-HV220
MPL-B660F	3000	34.0	40.7 (360)	68.0	73.4 (650)	6.1	2098-*xxx*-HV150
		38.5	48.0 (425)	94.0	96.0 (850)		2098-*xxx*-HV220

Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A (0-pk)	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A (0-pk)	**System Peak Stall Torque** N•m (lb•in)	**Motor Rated Output** kW	**Ultra3000/ Ultra5000 460V Drives**
MPL-B680D	2000	34.0	62.8 (556)	94.0	154.2 (1365)	9.3	2098-*xxx*-HV220
MPL-B680F	3000	48.0	58.2 (515)	94.0	101.7 (900)	7.5	2098-*xxx*-HV220
MPL-B860D	2000	47.5	83.1 (735)	94.0	151 (1335)	12.5	2098-*xxx*-HV220
MPL-B880C	1500	47.5	109.9 (973)	94.0	197 (1742)	12.6	2098-*xxx*-HV220
MPL-B880D	2000	47.0	77.4 (685)	94.0	144 (1275)	12.6	2098-*xxx*-HV220
MPL-B960B	1200	42.5	130 (1150)	94.0	231 (2050)	12.7	2098-*xxx*-HV220
MPL-B960C	1500	41.5	112 (990)	94.0	181 (1600)	14.8	2098-*xxx*-HV220
MPL-B980B	1000	40.0	163 (1440)	94.0	278 (2460)	15.2	2098-*xxx*-HV220
MPL-B980C	1500	47.5	118.6 (1050)	94.0	213 (1890)	16.8	2098-*xxx*-HV220

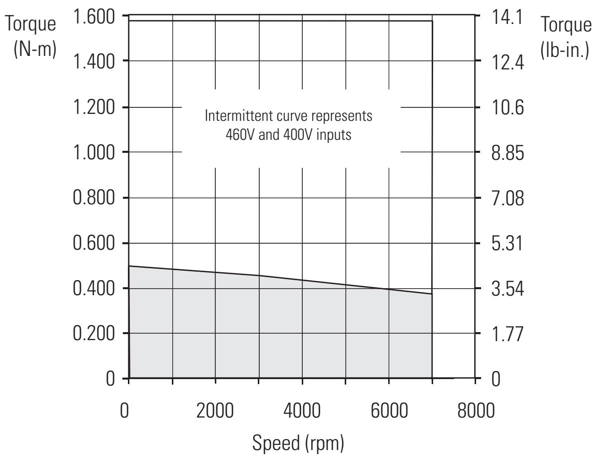
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Ultra3000 (460V) Drives/MP-Series Low Inertia Motor Curves

2098-DSD-HV030 and MPL-B1510V

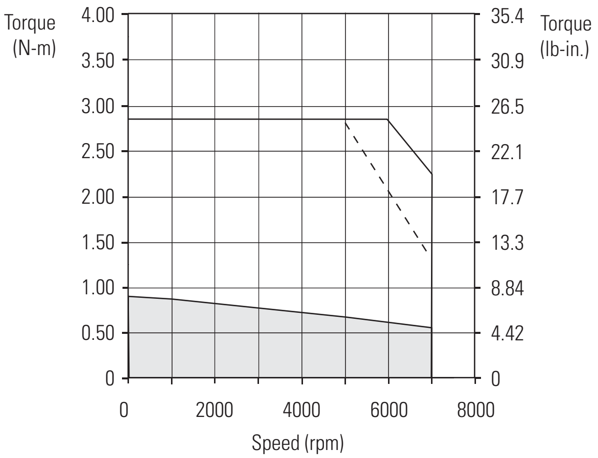


2098-DSD-HV030 and MPL-B1520U

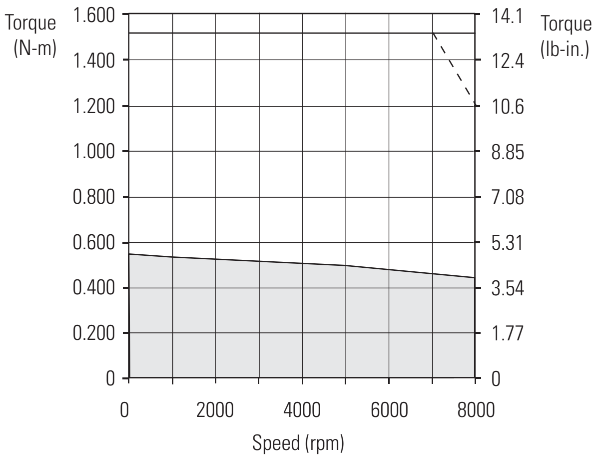


Intermittent curve represents 460V and 400V inputs

2098-DSD-HV030 and MPL-B1530U



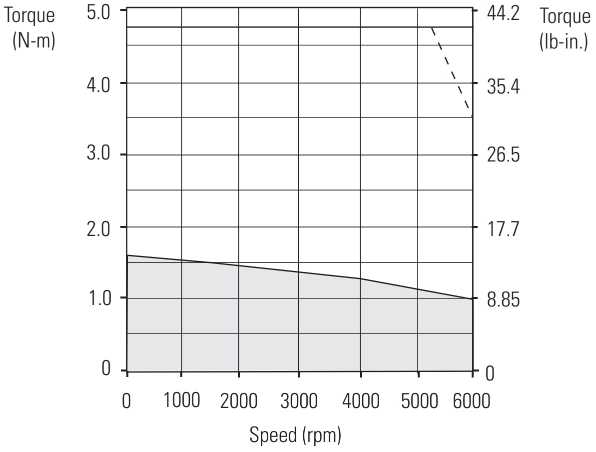
2098-DSD-HV030 and MPL-B210V



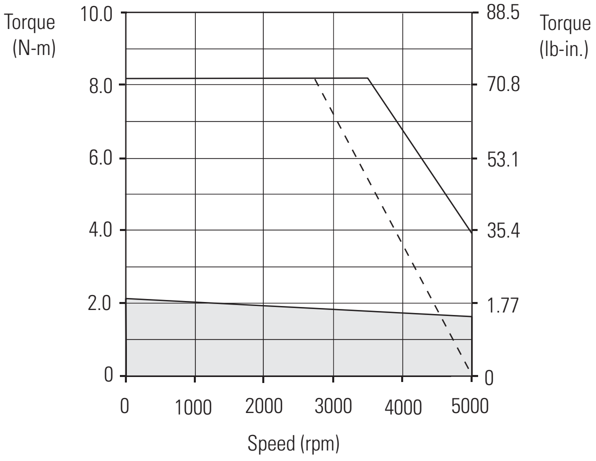
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Ultra3000/5000 (460V) Drives/MP-Series Low Inertia Motors, Continued

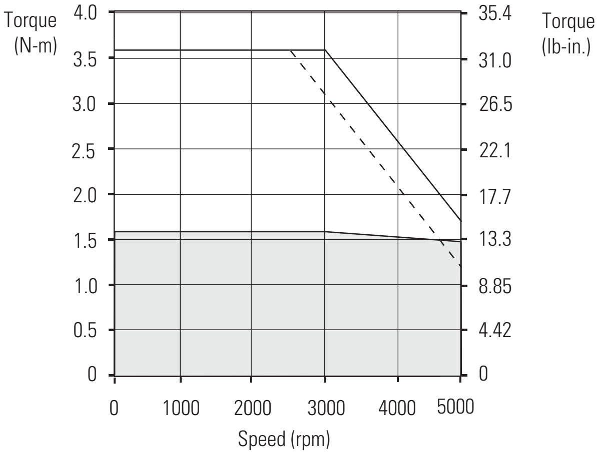
2098-DSD-HV030 and MPL-B220T



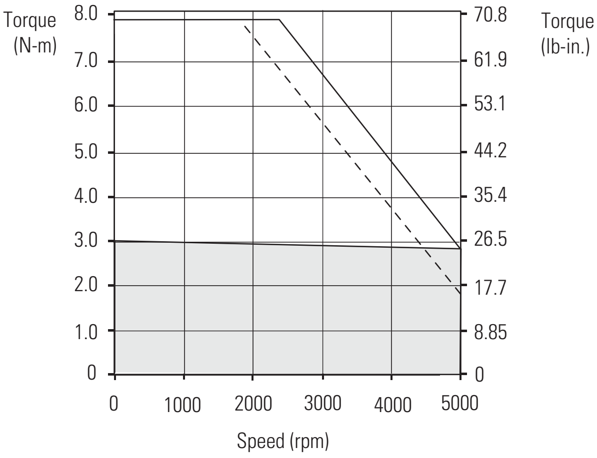
2098-DSD-HV030 and MPL-B230P



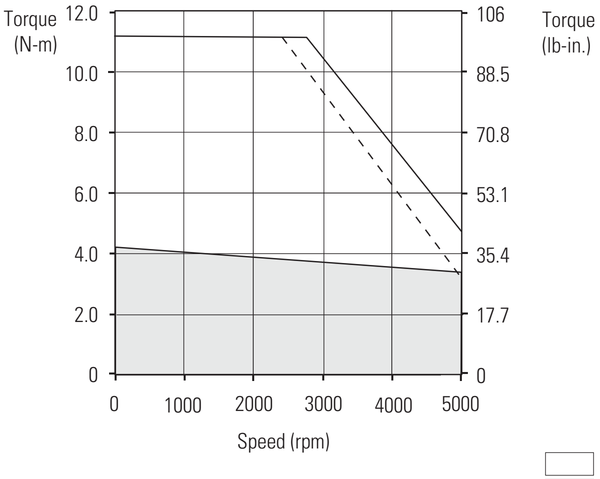
2098-*xxx*-HV030 and MPL-B310P



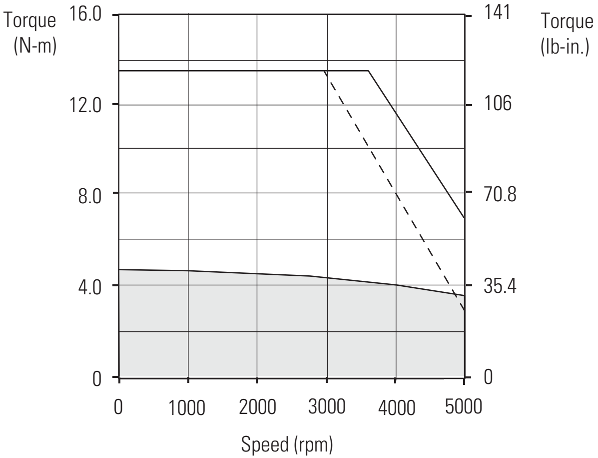
2098-*xxx*-HV030 and MPL-B320P



2098-*xxx*-HV050 and MPL-B330P



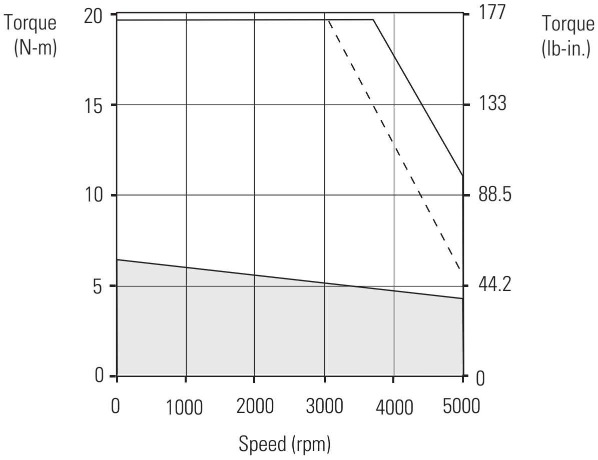
2098-*xxx*-HV100 and MPL-B420P



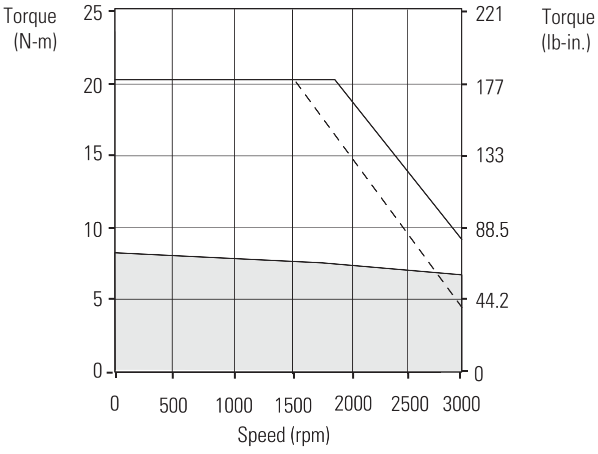
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Ultra3000/5000 (460V) Drives/MP-Series Low Inertia Motors, Continued

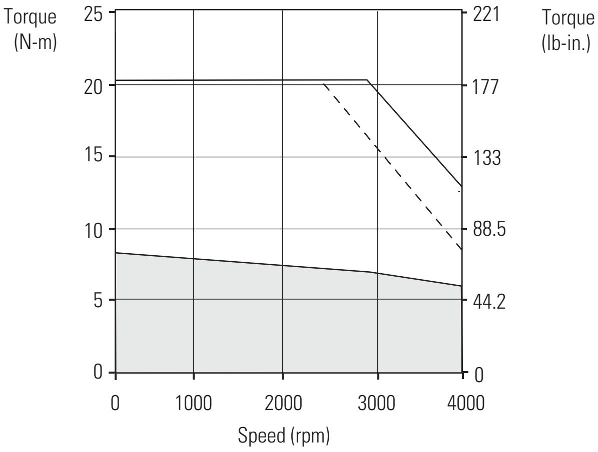
2098-*xxx*-HV100 and MPL-B430P



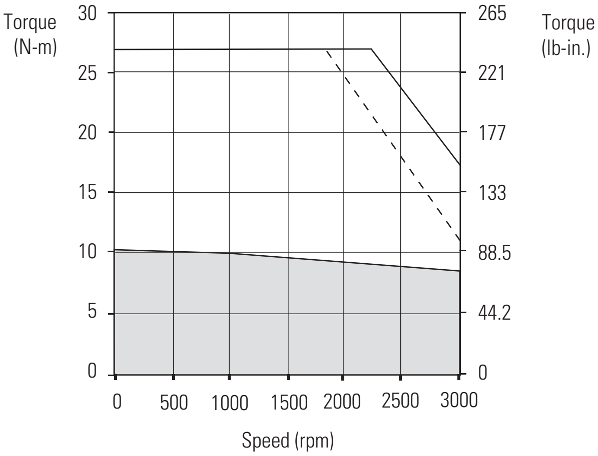
2098-*xxx*-HV050 and MPL-B4530F



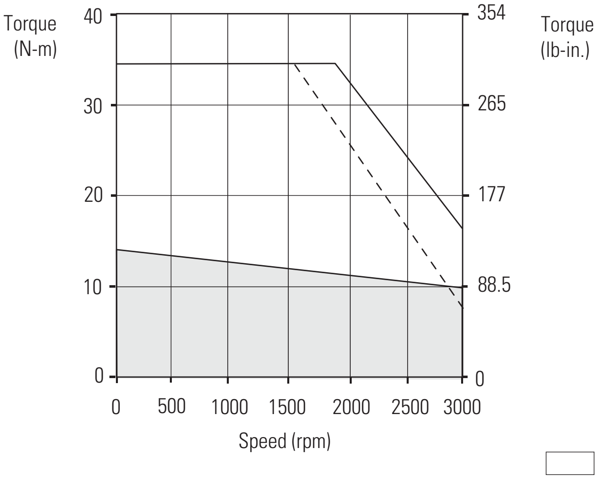
2098-*xxx*-HV100 and MPL-B4530K



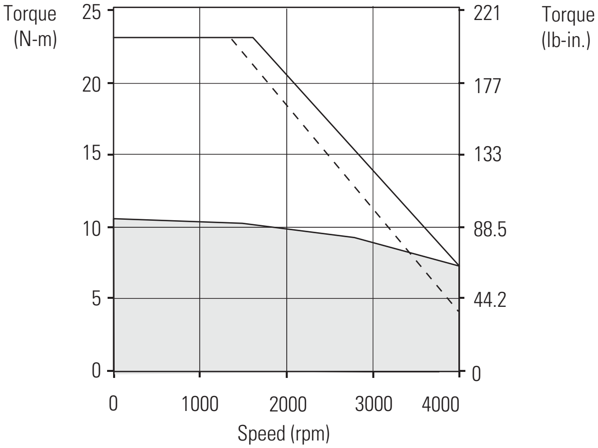
2098-*xxx*-HV100 and MPL-B4540F



2098-*xxx*-HV100 and MPL-B4560F



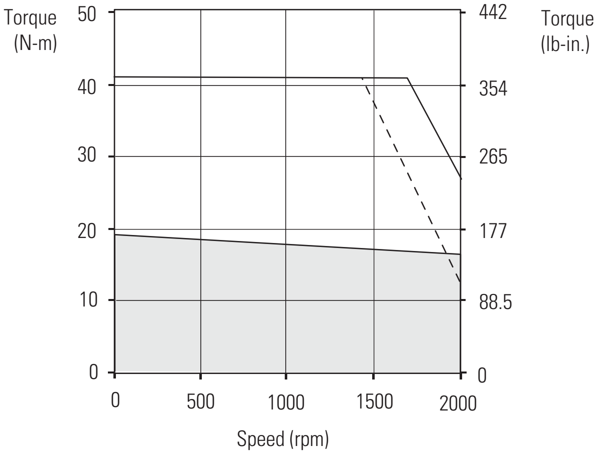
2098-*xxx*-HV100 and MPL-B520K



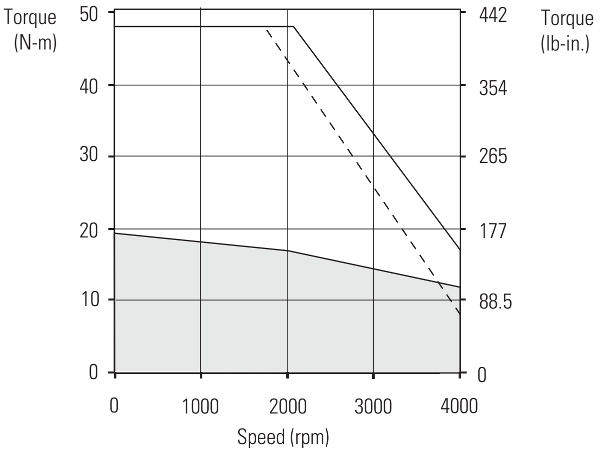
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Ultra3000/5000 (460V) Drives/MP-Series Low Inertia Motors, Continued

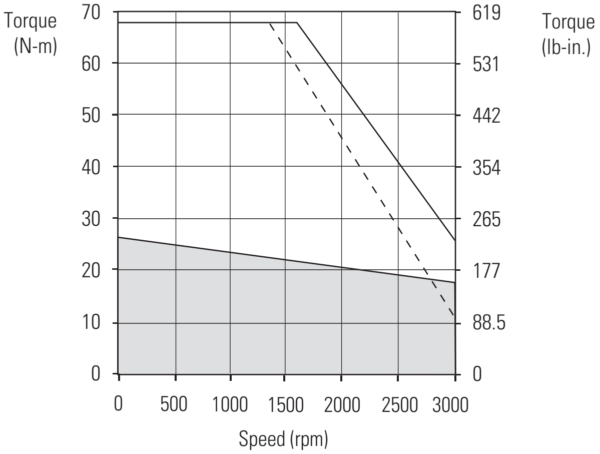
2098-*xxx*-HV100 and MPL-B540D



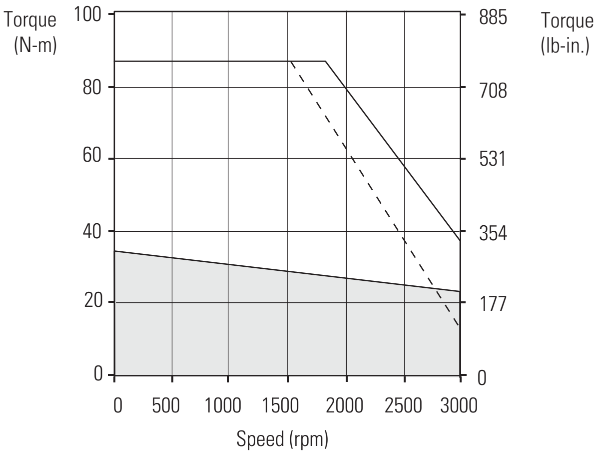
2098-*xxx*-HV150 and MPL-B540K



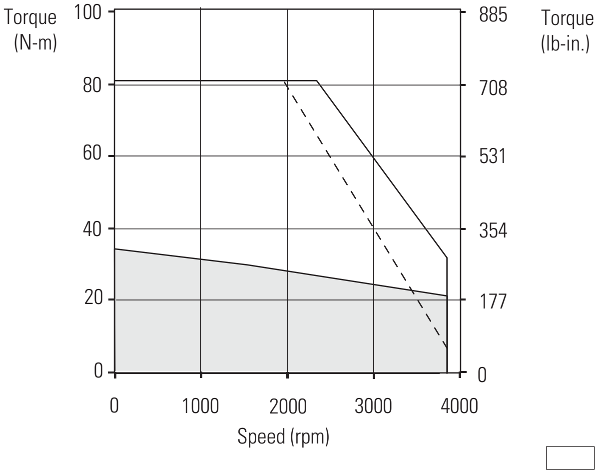
2098-*xxx*-HV150 and MPL-B560F



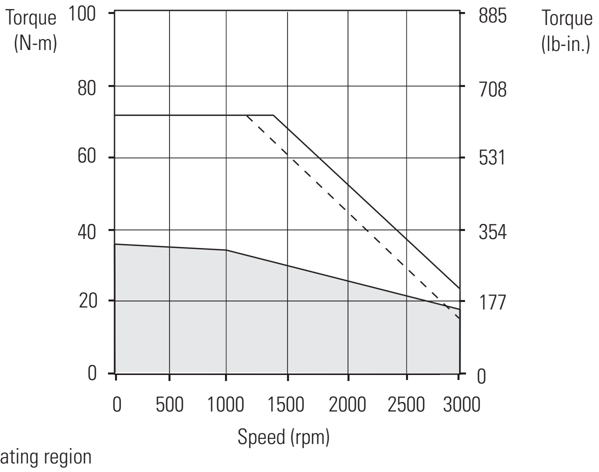
2098-*xxx*-HV220 and MPL-B580F



2098-*xxx*-HV220 and MPL-B580J



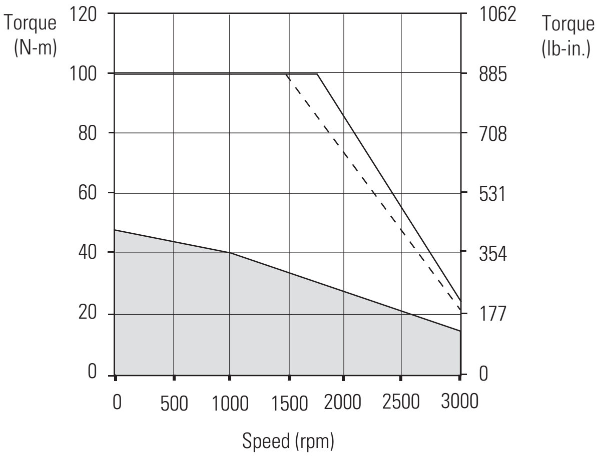
2098-*xxx*-HV220 and MPL-B640F



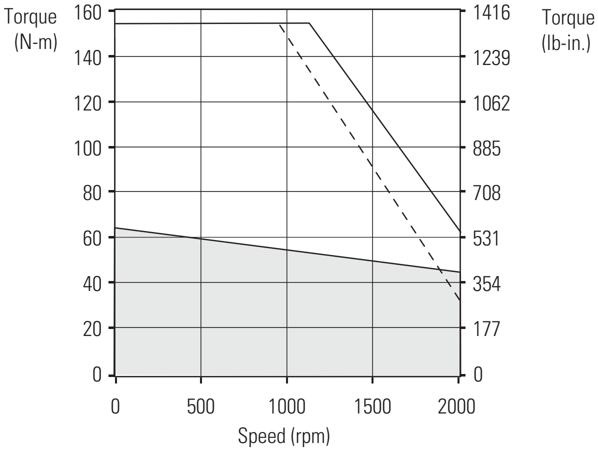
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Ultra3000/5000 (460V) Drives/MP-Series Low Inertia Motors, Continued

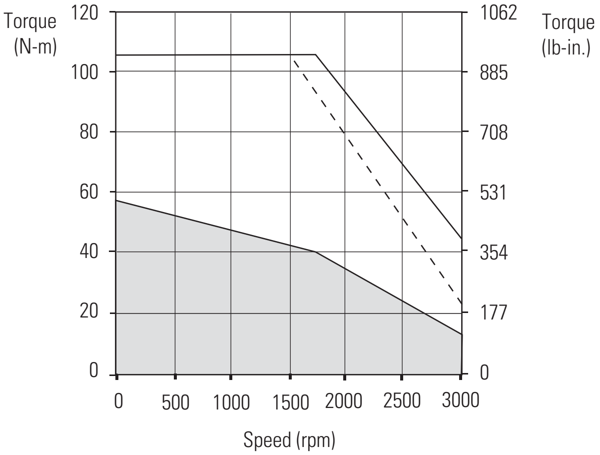
2098-*xxx*-HV220 and MPL-B660F



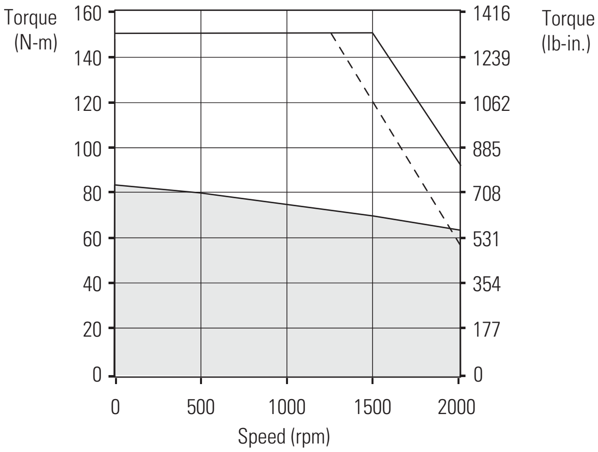
2098-*xxx*-HV220 and MPL-B680D



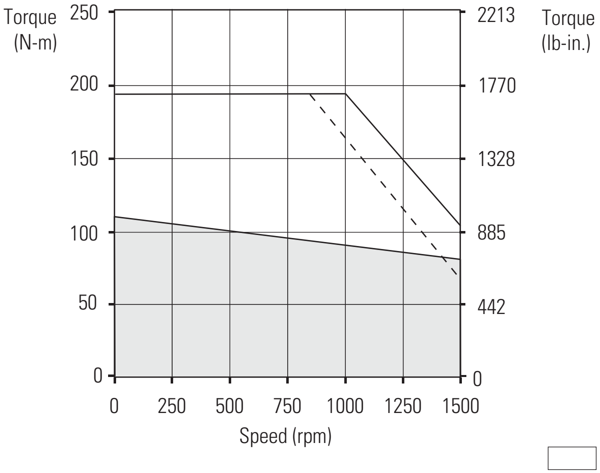
2098-*xxx*-HV220 and MPL-B680F



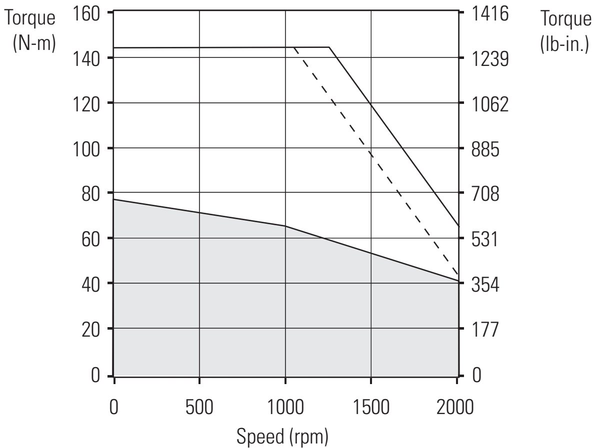
2098-*xxx*-HV220 and MPL-B860D



2098-*xxx*-HV220 and MPL-B880C



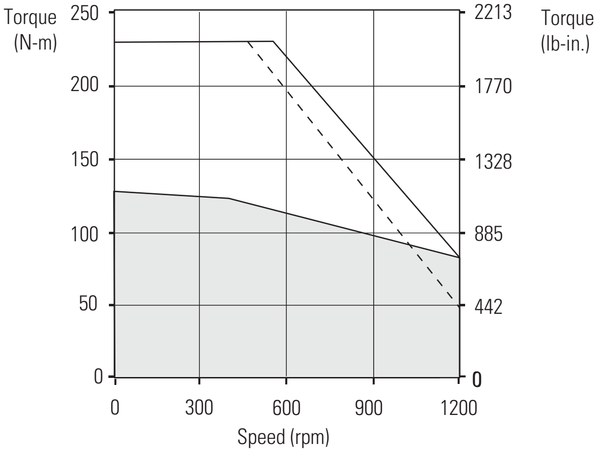
2098-*xxx*-HV220 and MPL-B880D



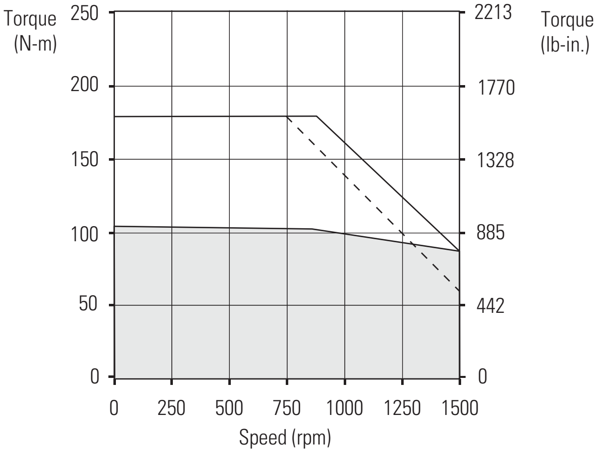
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Ultra3000/5000 (460V) Drives/MP-Series Low Inertia Motors, Continued

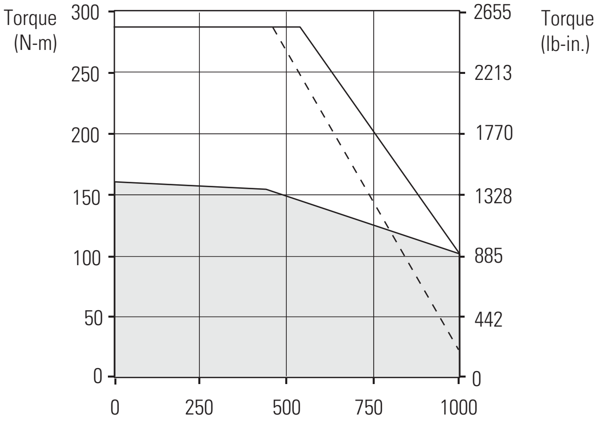
2098-*xxx*-HV220 and MPL-B960B



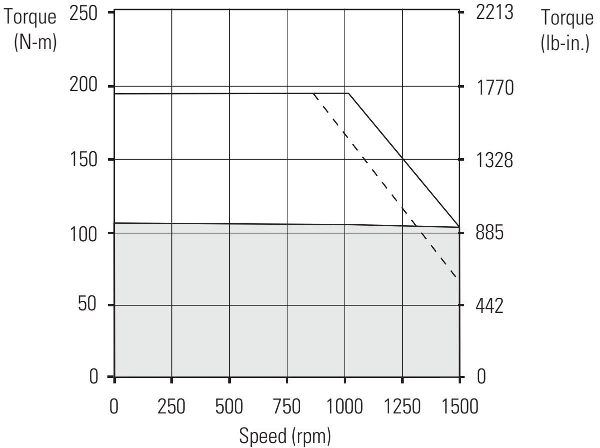
2098-*xxx*-HV220 and MPL-B960C



2098-*xxx*-HV220 and MPL-B980B



2098-*xxx*-HV220 and MPL-B980C



☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Ultra3000/5000 (230V) Drives with MP-Series Medium Inertia Motors

This section provides system combination information for the Ultra3000/5000 (230V) drives when matched with MP-Series medium-inertia motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin MPM Motor Cable Combinations

Motor Cat. No. (200V class)	Motor Power/Brake Cable	Motor Feedback Cable [1]
MPM-A1151M, MPM-A1152F, MPM-A1153F	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback
MPM-A1302F	2090-CP*x*M7DF-14AA*xx* (standard) 2090-CP*x*M7DF-14AF*xx* (continuous-flex)	
MPM-A1304F	2090-CP*x*M7DF-12AA*xx* (standard)	
MPM-A1651F	2090-CP*x*M7DF-10AA*xx* (standard) 2090-CP*x*M7DF-10AF*xx* (continuous-flex)	
MPM-A1652F, MPM-A1653F	2090-CP*x*M7DF-08AA*xx* (standard) 2090-CP*x*M7DF-08AF*xx* (continuous-flex)	
MPM-A2152F, MPM-A2153F, MPM-A2154C, MPM-A2154E	2090-CPBM7DF-06AA*xx* (standard)	

(1) Use drive-mounted breakout board (catalog number 2090-UXBB-DM15) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

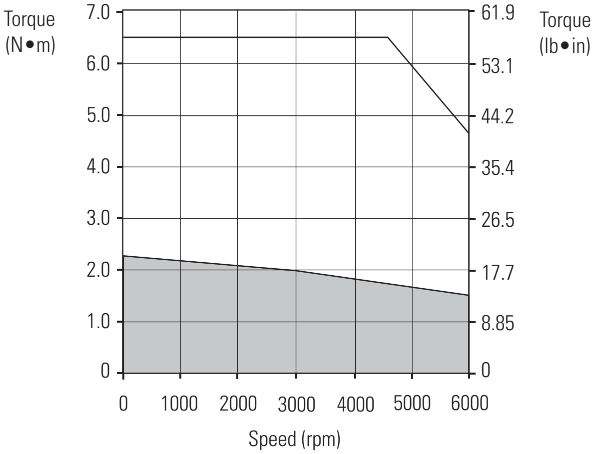
Bulletin MPM Motor Performance Specifications with Ultra3000/5000 (230V) Drives

Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Ultra3000/ Ultra5000 230V Drives
MPM-A1151M	6000	10.3	2.3 (20.3)	30.5	6.6 (58.4)	0.90	2098-*xxx*-030
MPM-A1152F	5000	14.9	4.7 (41.6)	44.8	13.5 (119)	1.40	2098-*xxx*-075
MPM-A1153F	5000	18.6	6.5 (57.5)	64.5	19.8 (175)	1.45	2098-*xxx*-075
MPM-A1302F	4500	19.8	6.6 (58.4)	50.2	13.5 (119)	1.65	2098-*xxx*-075
MPM-A1304F	4000	22.5	9.2 (81.4)	48.3	19.3 (171)	2.20	2098-*xxx*-075
MPM-A1651F	5000	35.6	10.7 (94.7)	75.0	20.4 (180)	2.50	2098-*xxx*-075
MPM-A1652F	4000	38.5	13.4 (119)	103.2	36.0 (318)	4.03	2098-*xxx*-150
MPM-A1653F	4000	48.7	18.6 (165)	119.1	41.9 (371)	5.10	2098-*xxx*-150
MPM-A2152F	4000	65.0	26.9 (238)	125.8	56.0 (495)	5.20	2098-*xxx*-150
MPM-A2153F	3600	65.0	35.2 (311)	120.4	58.0 (513)	5.80	2098-*xxx*-150
MPM-A2154C	2000	65.0	55.5 (491)	127.3	106 (938)	6.50	2098-*xxx*-150
MPM-A2154E	3000	65.0	44.0 (389)	128.2	83.9 (742)	7.00	2098-*xxx*-150

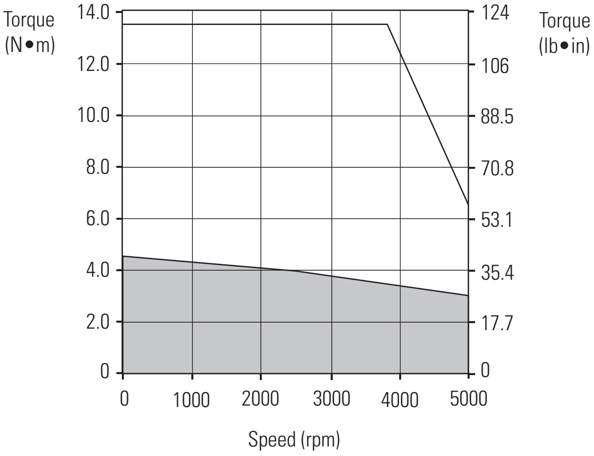
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Ultra3000/5000 (230V) Drives/MP-Series Medium Inertia Motor Curves

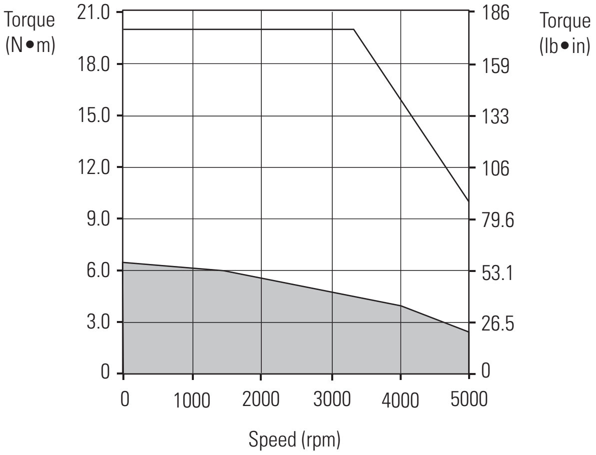
2098-*xxx*-030 and MPM-A1151M



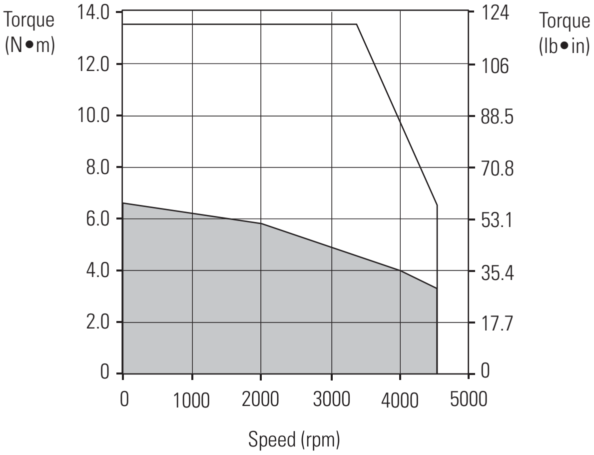
2098-*xxx*-075 and MPM-A1152F



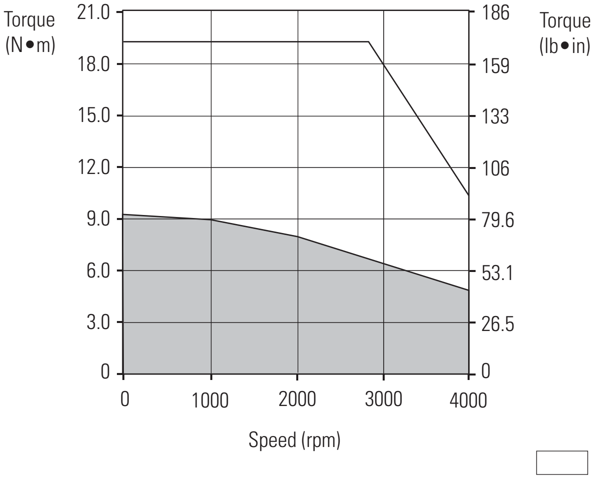
2098-*xxx*-075 and MPM-A1153F



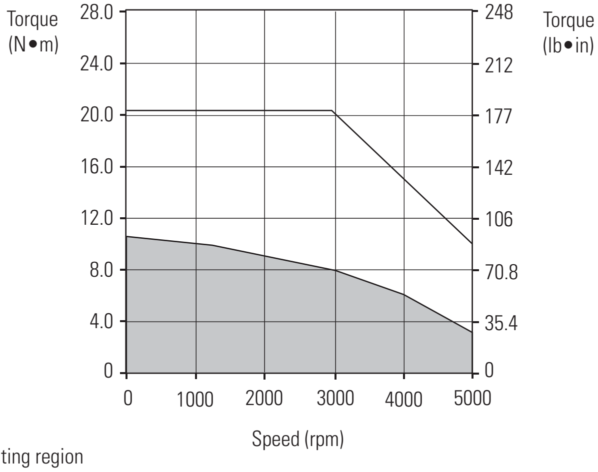
2098-*xxx*-075 and MPM-A1302F



2098-*xxx*-075 and MPM-A1304F



2098-*xxx*-075 and MPM-A1651F

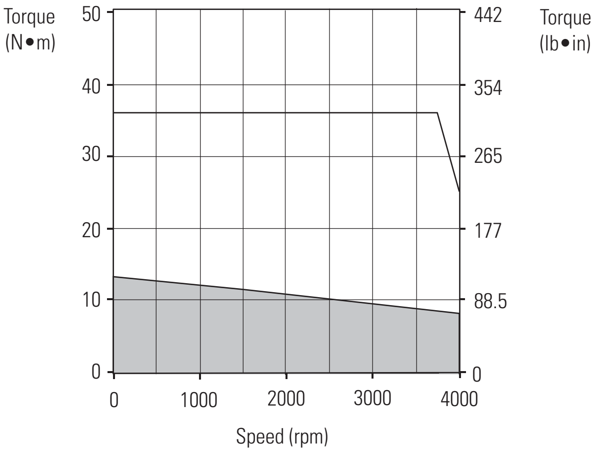


☐ = Intermittent operating region

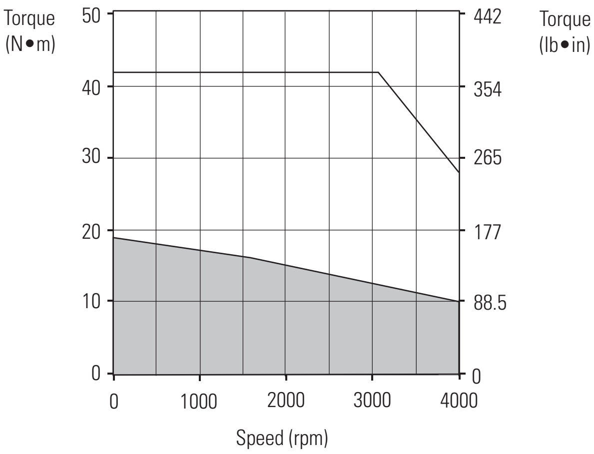
▨ = Continuous operating region

Ultra3000/5000 (230V) Drives/MP-Series Medium Inertia Motor Curves, Continued

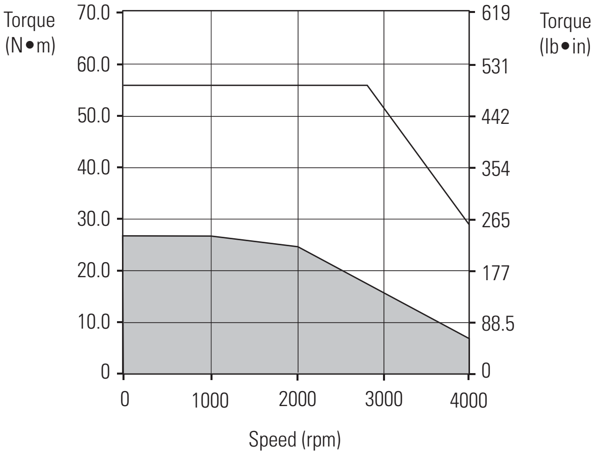
2098-*xxx*-HV150 and MPM-A1652F



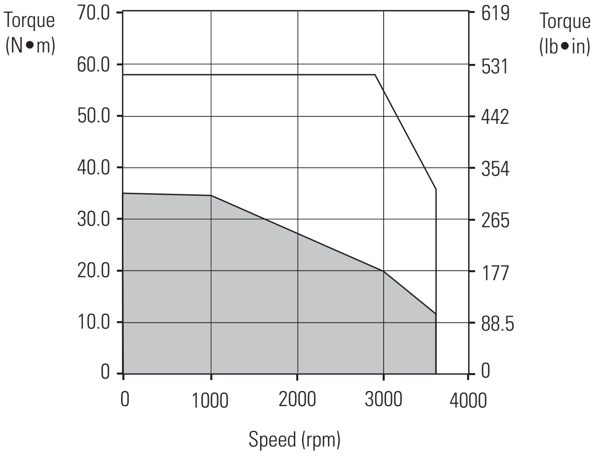
2098-*xxx*-HV150 and MPM-A1653F



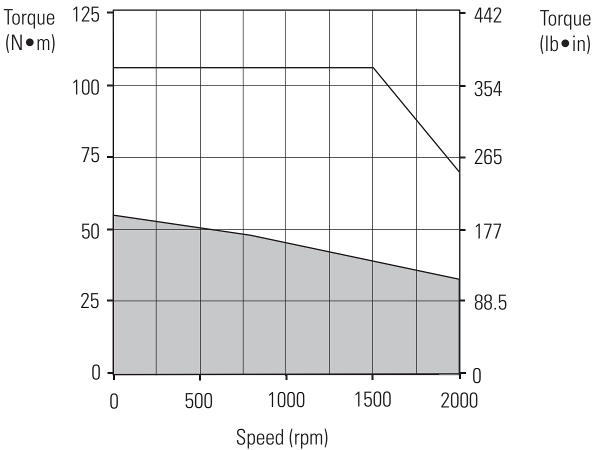
2098-*xxx*-HV150 and MPM-A2152F



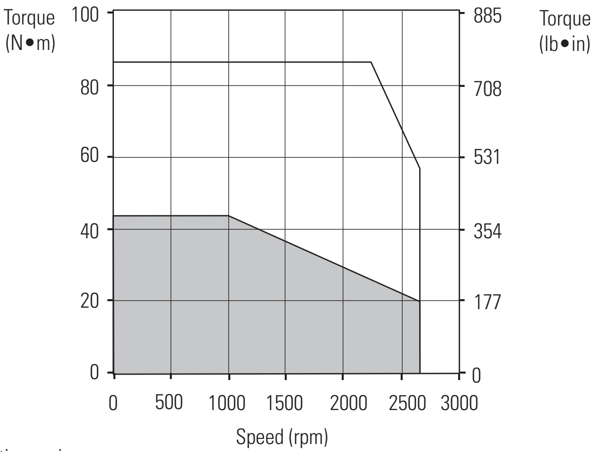
2098-*xxx*-HV150 and MPM-A2153F



2098-*xxx*-HV150 and MPM-A2154C



2098-*xxx*-150 and MPM-A2154E



☐ = Intermittent operating region
▨ = Continuous operating region

Ultra3000/5000 (460V) Drives with MP-Series Medium Inertia Motors

This section provides system combination information for the Ultra3000/5000 (460V) drives when matched with MP-Series medium-inertia motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin MPM Motor Cable Combinations

Motor Cat. No. (400V class)	Motor Power/Brake Cable	Motor Feedback Cable (1)
MPM-B1151*x*, MPM-B1152*x*, MPM-B1153E, MPM-B1153F	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback
MPM-B1302F, MPM-B1302M, MPM-B1304C, MPM-B1304E		
MPM-B1651C, MPM-B1652C		
MPM-B1153T	2090-CP*x*M7DF-14AA*xx* (standard) 2090-CP*x*M7DF-14AF*xx* (continuous-flex)	
MPM-B1302T, MPM-B1304M		
MPM-B1651F, MPM-B1653C		
MPM-B1651M, MPM-B1652E, MPM-B1652F, MPM-B1653E	2090-CP*x*M7DF-10AA*xx* (standard) 2090-CP*x*M7DF-10AF*xx* (continuous-flex)	
MPM-B2152C, MPM-B2153B		
MPM-B1653F	2090-CP*x*M7DF-08AA*xx* (standard) 2090-CP*x*M7DF-08AF*xx* (continuous-flex)	
MPM-B2152F, MPM-B2152M, MPM-B2153E, MPM-B2153F, MPM-B2154B, MPM-B2154E, MPM-B2154F		

(1) Use drive-mounted breakout board (catalog number 2090-UXBB-DM15) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

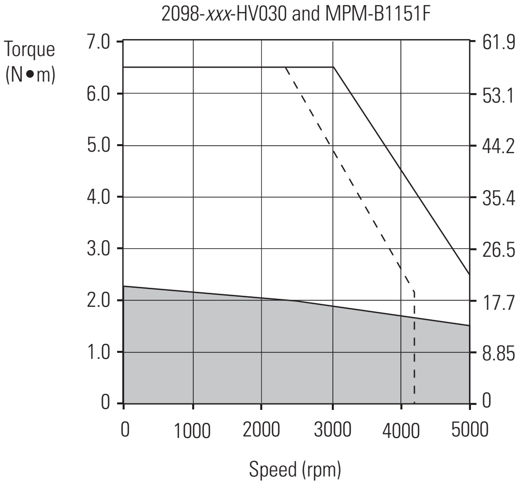
Bulletin MPM Motor Performance Specifications with Ultra3000/5000 (460V) Drives

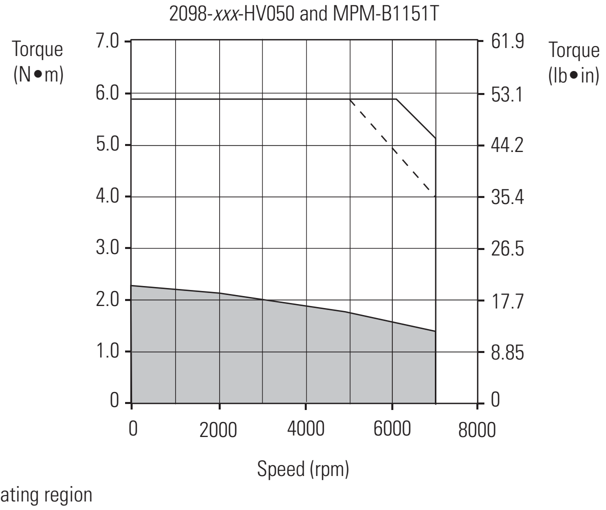
Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Stall Torque** N•m (lb•in)	**Motor Rated Output** kW	**Ultra3000/ Ultra5000 460V Drives**
MPM-B1151F	5000	3.1	2.3 (20.3)	9.9	6.6 (58.4)	0.75	2098-*xxx*-HV030
MPM-B1151T	7000	6.9	2.3 (20.3)	14.0	4.4 (38.9)	0.90	2098-*xxx*-HV030
				20.5	5.8 (51.3)		2098-*xxx*-HV050
MPM-B1152C	3000	4.1	5.0 (44.2)	12.4	13.5 (119)	1.20	2098-*xxx*-HV030
MPM-B1152F	5200	7.0	5.0 (44.2)	14.0	9.6 (84.9)	1.40	2098-*xxx*-HV030
				21.1	13.3 (118)		2098-*xxx*-HV050
MPM-B1152T	7000	12.6	5.0 (44.2)	22.0	8.6 (76.1)	1.40	2098-*xxx*-HV050
				37.8	13.5 (119)		2098-*xxx*-HV100
MPM-B1153E	3500	7.1	6.5 (57.5)	14.0	13.8 (122)	1.40	2098-*xxx*-HV030
				21.6	19.7 (174)		2098-*xxx*-HV050
MPM-B1153F	5500	10.5	6.4 (56.6)	22.0	14.6 (129)	1.40	2098-*xxx*-HV050
				32.0	19.7 (174)		2098-*xxx*-HV100
MPM-B1153T	7000	18.3	6.4 (56.6)	46.0	14.8 (131)	1.45	2098-*xxx*-HV100
				55.4	16.5 (146)		2098-*xxx*-HV150
MPM-B1302F	4500	9.8	6.6 (58.4)	22.0	13.2 (117)	1.65	2098-*xxx*-HV050
MPM-B1302M	6000	14.4	6.6 (58.4)	32.4	13.3 (118)	1.65	2098-*xxx*-HV100
MPM-B1302T	7000	19.3	6.7 (59.3)	43.4	13.3 (118)	1.65	2098-*xxx*-HV100
MPM-B1304C	2750	8.0	10.3 (91.1)	14.0	18.7 (165)	2.00	2098-*xxx*-HV030
				22.0	26.8 (237)		2098-*xxx*-HV050
MPM-B1304E	4000	12.3	10.2 (90.3)	22.0	19.1 (169)	2.20	2098-*xxx*-HV050
				34.2	27.1 (240)		2098-*xxx*-HV100
MPM-B1304M	6000	21.8	10.4 (92.0)	46.0	21.9 (194)	2.20	2098-*xxx*-HV100
				60.6	27.1 (240)		2098-*xxx*-HV150
MPM-B1651C	3500	11.7	11.4 (101)	22.0	19.5 (172)	2.50	2098-*xxx*-HV050
				29.2	23.2 (205)		2098-*xxx*-HV100
MPM-B1651F	5000	20.4	11.4 (101)	46.0	21.8 (193)	2.50	2098-*xxx*-HV100
				50.9	23.2 (205)		2098-*xxx*-HV150
MPM-B1651M	5000	25.8	11.3 (100)	46.0	18.5 (164)	2.50	2098-*xxx*-HV100
				56.8	21.4 (189)		2098-*xxx*-HV150
MPM-B1652C	2500	13.2	16.4 (145)	22.0	30.0 (265)	3.80	2098-*xxx*-HV050
				33.6	40.2 (356)		2098-*xxx*-HV100
MPM-B1652E	3500	24.0	21.1 (187)	46.0	39.1 (346)	4.30	2098-*xxx*-HV100
				60.5	48.0 (425)		2098-*xxx*-HV150
MPM-B1652F	4500	33.0	21.1 (187)	68.0	39.1 (346)	4.30	2098-*xxx*-HV150
				84.1	45.0 (398)		2098-*xxx*-HV220
MPM-B1653C	2500	23.0	26.7 (236)	46.0	56.1 (496)	4.60	2098-*xxx*-HV100
				59.2	67.7 (599)		2098-*xxx*-HV150

Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Stall Torque** N•m (lb•in)	**Motor Rated Output** kW	**Ultra3000/ Ultra5000 460V Drives**
MPM-B1653E	3500	31.0	26.8 (237)	68.0	58.8 (520)	5.10	2098-*xxx*-HV150
				72.9	62.0 (549)		2098-*xxx*-HV220
MPM-B1653F	4000	40.1	31.0 (274)	94.0	56.1 (496)	5.10	2098-*xxx*-HV220
MPM-B2152C	2500	31.5	36.7 (325)	55.4	72.2 (639)	5.60	2098-*xxx*-HV150
MPM-B2152F	4500	47.0	33.9 (300)	94.0	69.8 (618)	5.90	2098-*xxx*-HV220
MPM-B2152M	5000	47.0	34.1 (302)	94.0	52.9 (468)	5.90	2098-*xxx*-HV220
MPM-B2153B	2000	23.0	47.1 (417)	46.0	81.5 (721)	6.80	2098-*xxx*-HV100
		27.6	48.0 (425)	60.0	101.2 (895)		2098-*xxx*-HV150
MPM-B2153E	3000	45.5	47.9 (424)	94.0	97.1 (859)	7.20	2098-*xxx*-HV220
MPM-B2153F	3800	47.0	45.6 (403)	94.0	94.8 (839)	7.20	2098-*xxx*-HV220
MPM-B2154B	2000	40.7	62.7 (555)	94.0	149 (1319)	6.90	2098-*xxx*-HV220
MPM-B2154E	3000	47.0	55.9 (495)	94.0	108 (956)	7.50	2098-*xxx*-HV220
MPM-B2154F	3300	47.0	56.2 (497)	83.6	87.9 (778)	7.50	2098-*xxx*-HV220

Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Ultra3000/5000 (460V) Drives/MP-Series Medium Inertia Motor Curves

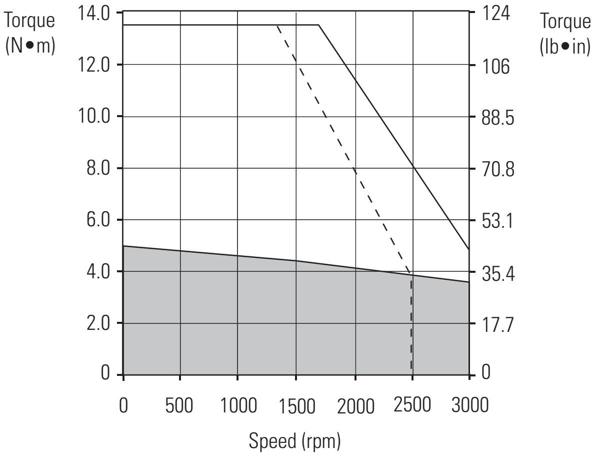




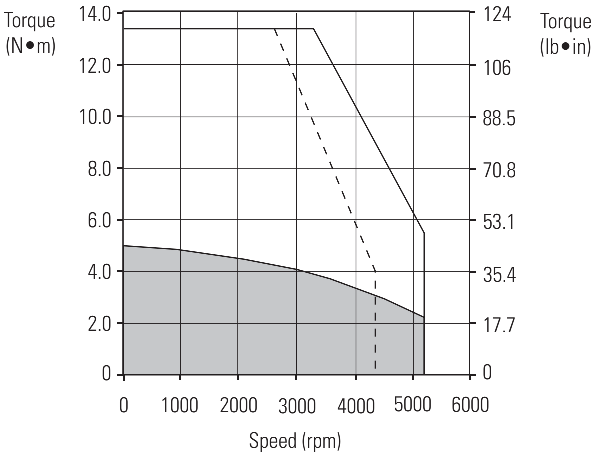
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Ultra3000/5000 (460V) Drives/MP-Series Medium Inertia Motor Curves, Continued

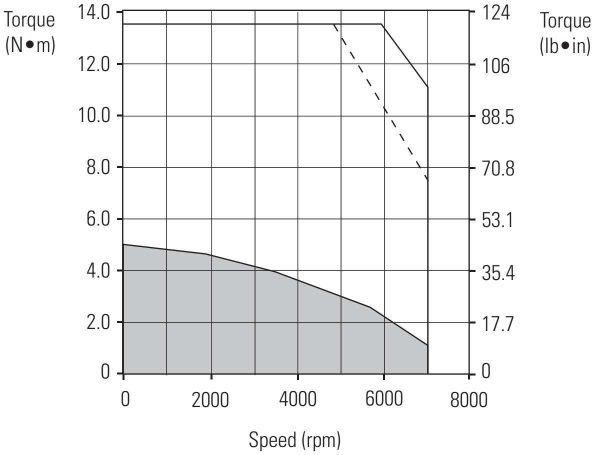
2098-*xxx*-HV030 and MPM-B1152C



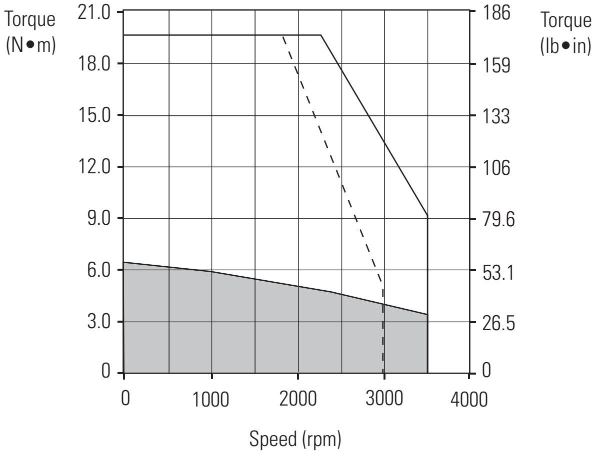
2098-*xxx*-HV050 and MPM-B1152F



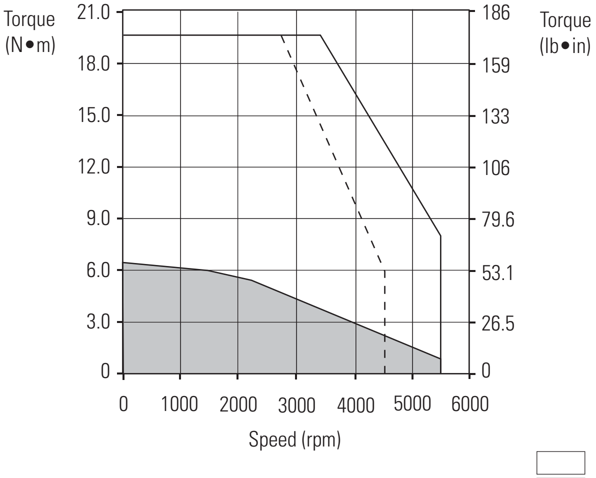
2098-*xxx*-HV100 and MPM-B1152T



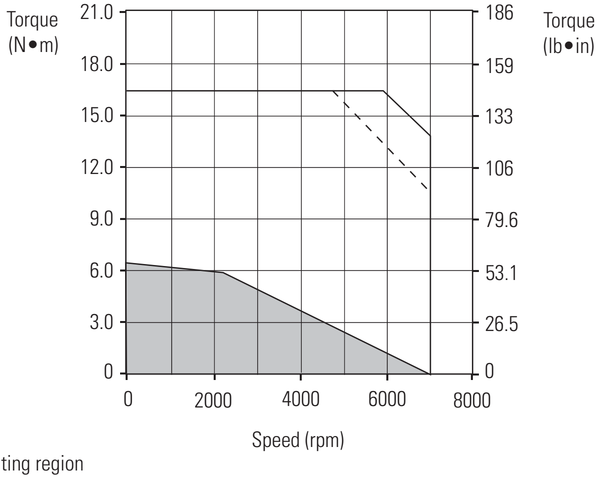
2098-*xxx*-HV050 and MPM-B1153E



2098-*xxx*-HV100 and MPM-B1153F



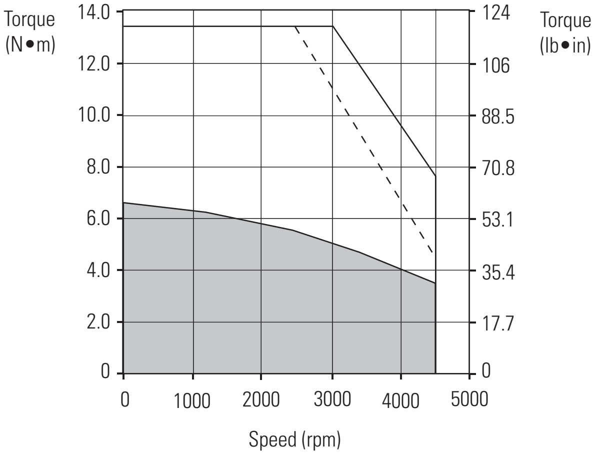
2098-*xxx*-HV150 and MPM-B1153T



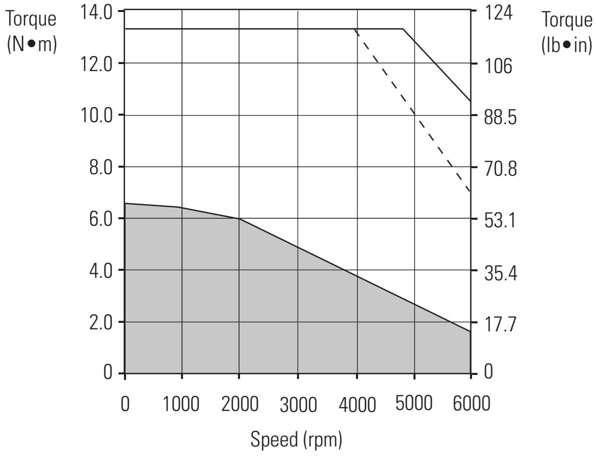
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Ultra3000/5000 (460V) Drives/MP-Series Medium Inertia Motor Curves, Continued

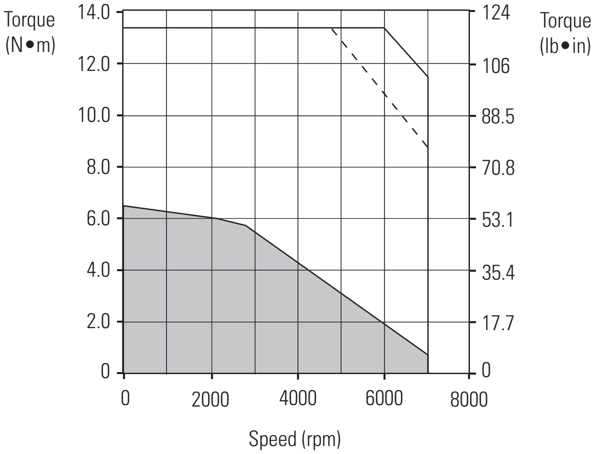
2098-*xxx*-HV050 and MPM-B1302F



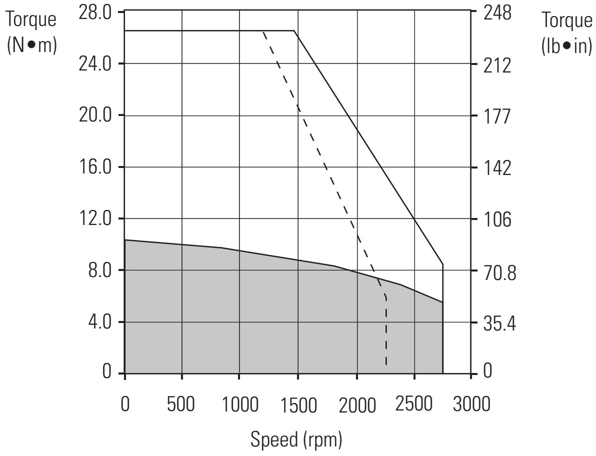
2098-*xxx*-HV100 and MPM-B1302M



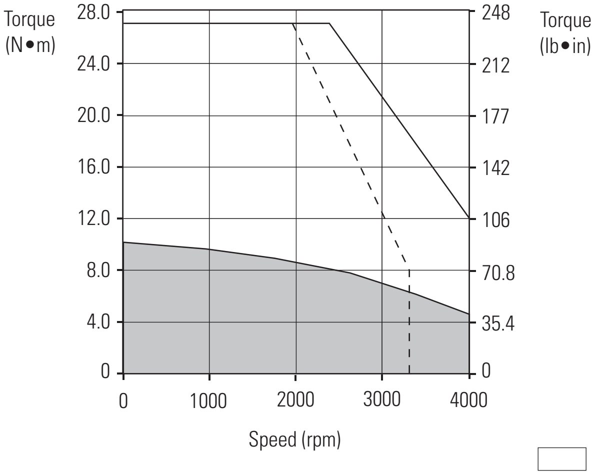
2098-*xxx*-HV100 and MPM-B1302T



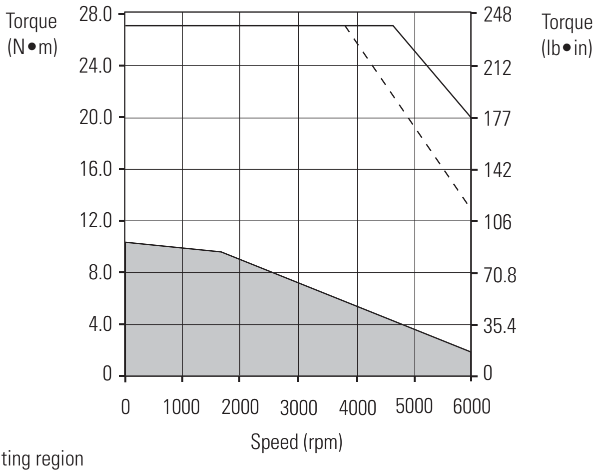
2098-*xxx*-HV050 and MPM-B1304C



2098-*xxx*-HV100 and MPM-B1304E



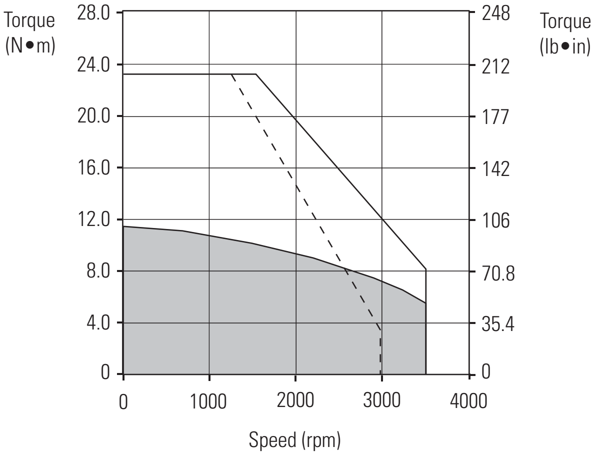
2098-*xxx*-HV150 and MPM-B1304M



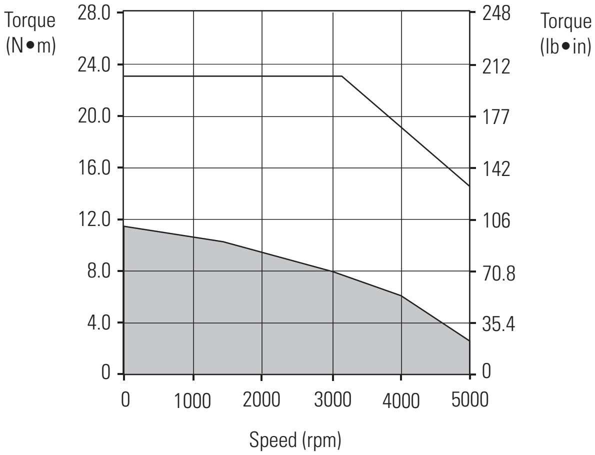
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Ultra3000/5000 (460V) Drives/MP-Series Medium Inertia Motor Curves, Continued

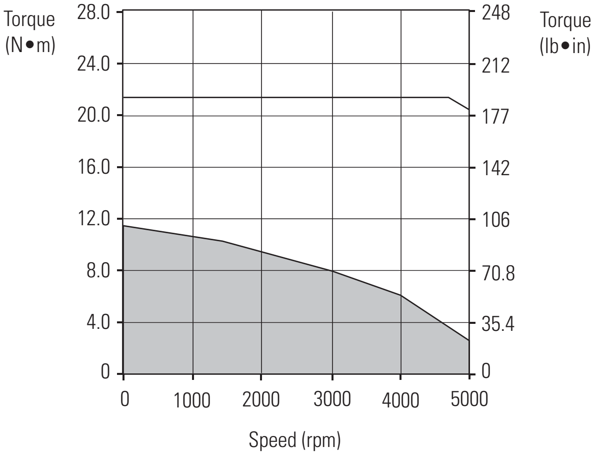
2098-*xxx*-HV100 and MPM-B1651C



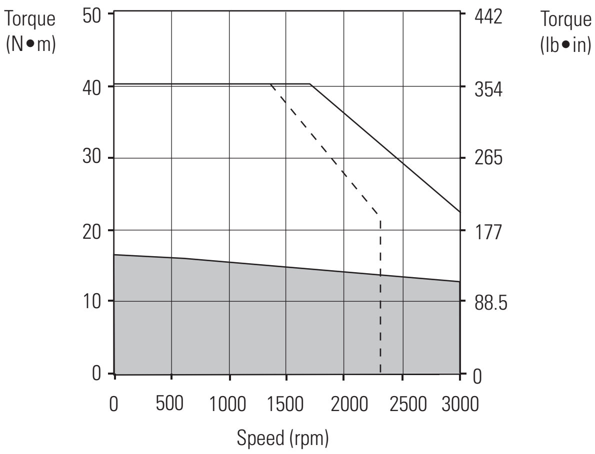
2098-*xxx*-HV150 and MPM-B1651F



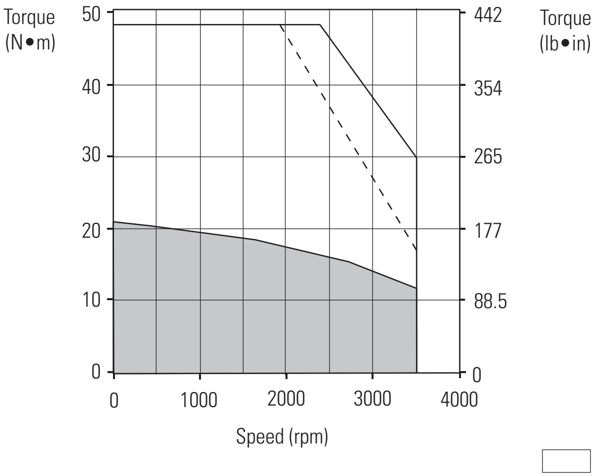
2098-*xxx*-HV150 and MPM-B1651M



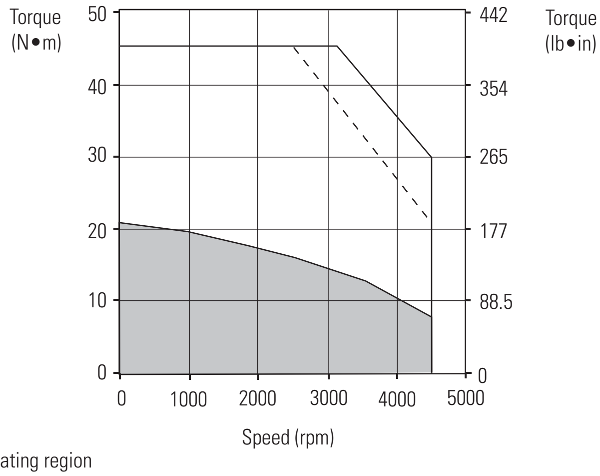
2098-*xxx*-HV100 and MPM-B1652C



2098-*xxx*-HV150 and MPM-B1652E



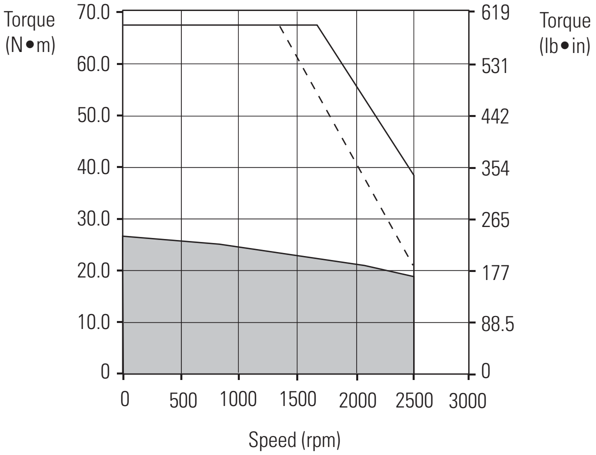
2098-*xxx*-HV220 and MPM-B1652F



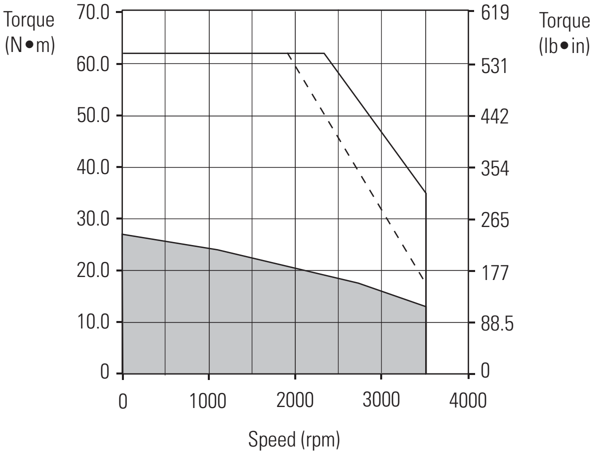
☐ = Intermittent operating region
▨ = Continuous operating region
– – – = Drive operation with 400V AC (rms) input voltage

Ultra3000/5000 (460V) Drives/MP-Series Medium Inertia Motor Curves, Continued

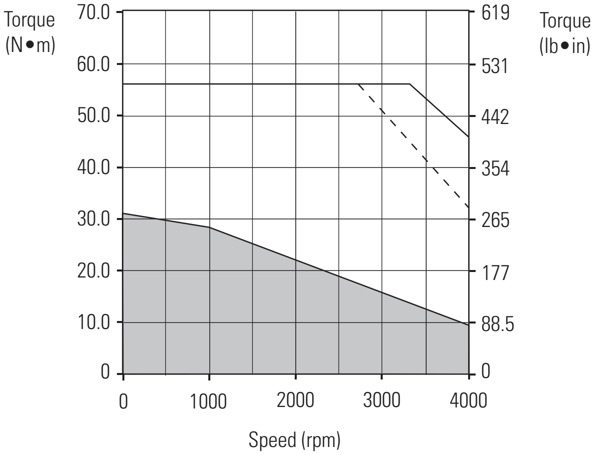
2098-*xxx*-HV150 and MPM-B1653C



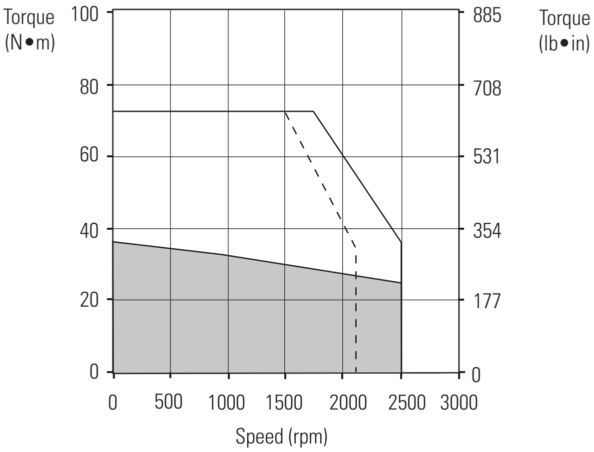
2098-*xxx*-HV220 and MPM-B1653E



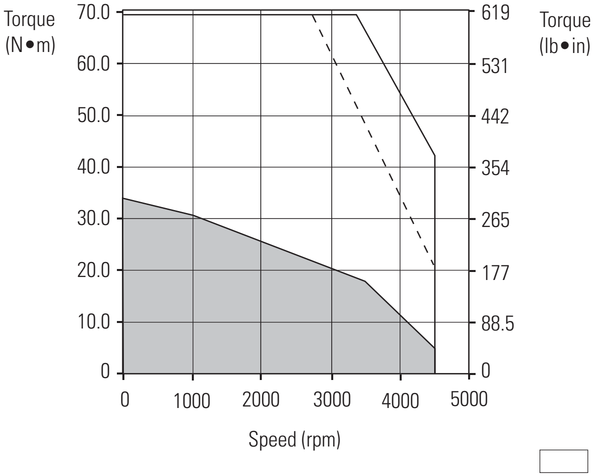
2098-*xxx*-HV220 and MPM-B1653F



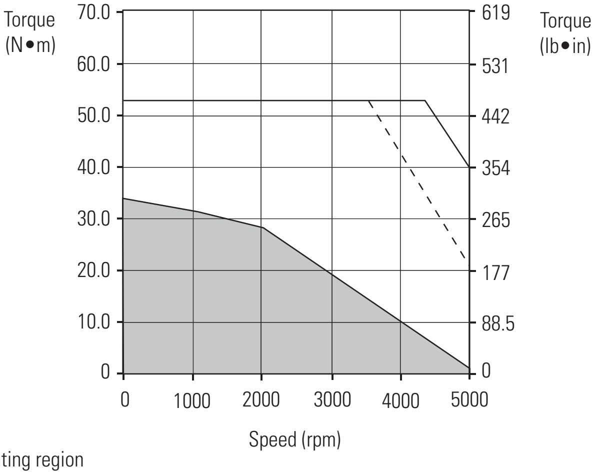
2094-BM03-S @ 250% or 2094-BM03-M and MPM-B2152C



2098-*xxx*-HV220 and MPM-B2152F



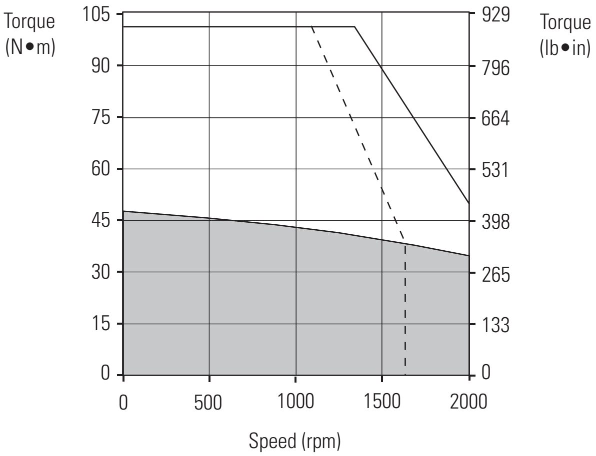
2098-*xxx*-HV220 and MPM-B2152M



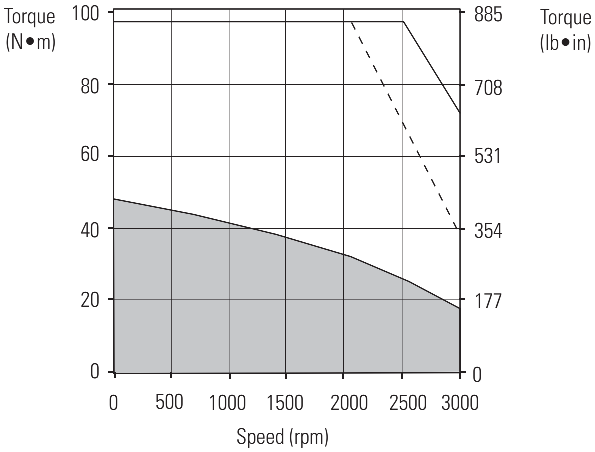
☐ = Intermittent operating region
▨ = Continuous operating region
– – – = Drive operation with 400V AC (rms) input voltage

Ultra3000/5000 (460V) Drives/MP-Series Medium Inertia Motor Curves, Continued

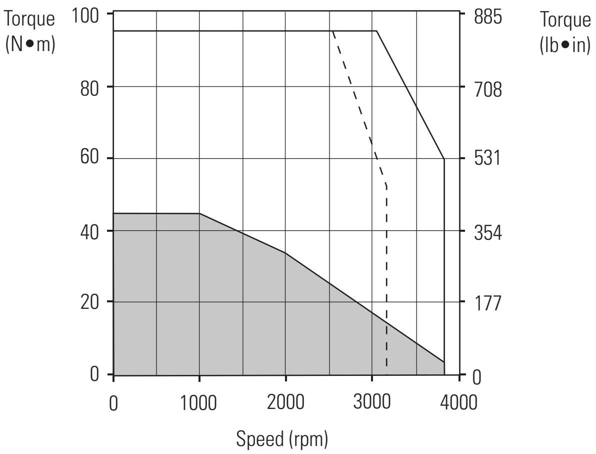
2098-*xxx*-HV150 and MPM-B2153B



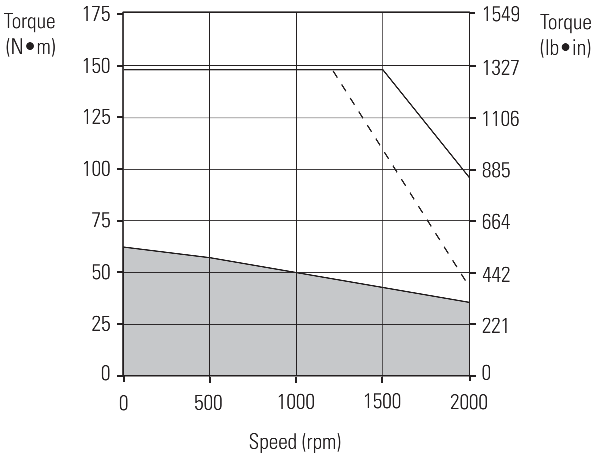
2098-*xxx*-HV220 and MPM-B2153E



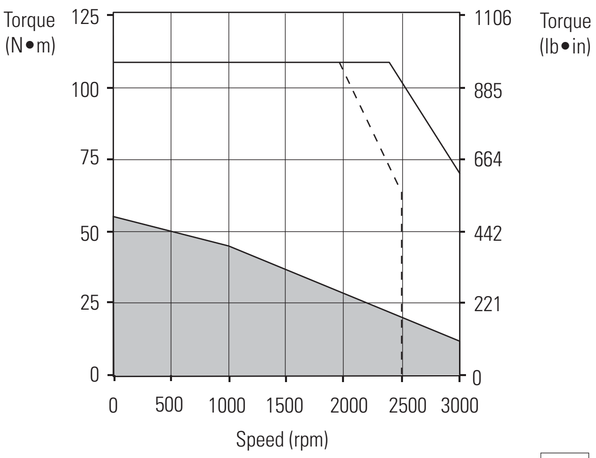
2098-*xxx*-HV220 and MPM-B2153F



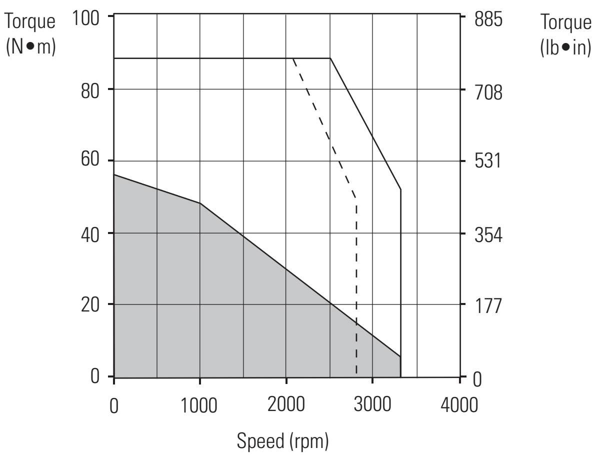
2098-*xxx*-HV220 and MPM-B2154B



2098-*xxx*-HV220 and MPM-B2154E



2098-*xxx*-HV220 and MPM-B2154F



☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Ultra3000/5000 (230V) Drives with MP-Series Food Grade Motors

This section provides system combination information for the Ultra3000/5000 (230V) drives when matched with MP-Series food-grade motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin MPF Motor Cable Combinations

Motor Cat. No. (200V class)	Motor Power Cable	Motor Feedback Cable [1]
MPF-A310P, MPF-A320H, MPF-A320P, MPF-A330P	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback
MPF-A430H		
MPF-A430P	2090-CP*x*M7DF-14AA*xx* (standard) 2090-CP*x*M7DF-14AF*xx* (continuous-flex)	
MPF-A4530K, MPF-A4540F		
MPF-A540K	2090-CP*x*M7DF-08AA*xx* (standard) 2090-CP*x*M7DF-08AF*xx* (continuous-flex)	

(1) Use drive-mounted breakout board (catalog number 2090-UXBB-DM15) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

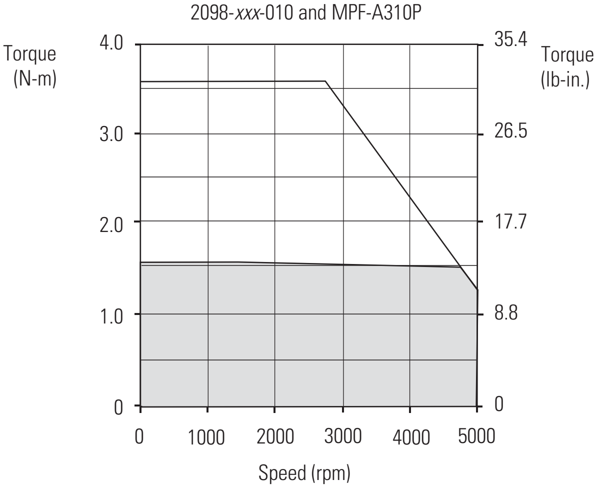
Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

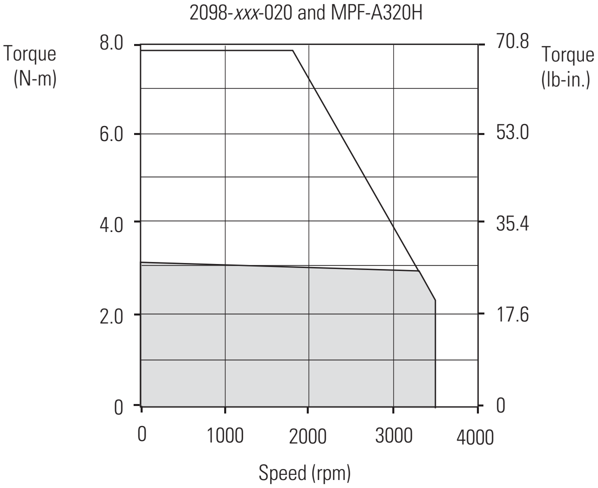
Bulletin MPF Motor Performance Specifications with Ultra3000/5000 (230V) Drives

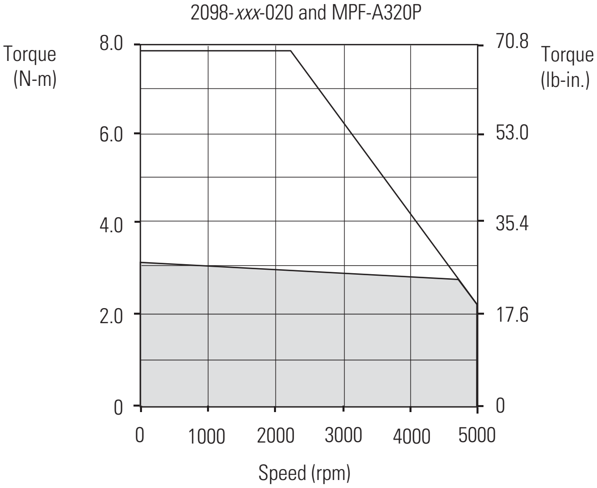
Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Ultra3000/ Ultra5000 230V Drives
MPF-A310P	5000	2.50	0.79 (7)	7.5	1.92 (17)	0.73	2098-*xxx*-005
		4.85	1.58 (14)	14	3.61 (32)		2098-*xxx*-010
MPF-A320H	3500	5.0	2.48 (22)	15	6.44 (57)	1.0	2098-*xxx*-010
		6.1	3.05 (27)	19.3	7.91 (70)		2098-*xxx*-020
MPF-A320P	5000	5.0	1.69 (15)	15	3.95 (35)	1.3	2098-*xxx*-010
		9.0	3.05 (27)	29.5	7.91 (70)		2098-*xxx*-020
MPF-A330P	5000	12.0	4.18 (37)	30	9.60 (85)	1.6	2098-*xxx*-030
				38	11.1 (98)		2098-*xxx*-075
MPF-A430H	3500	12.2	6.21 (55)	30	14.7 (130)	1.8	2098-*xxx*-030
				45	19.8 (175)		2098-*xxx*-075
MPF-A430P	5000	15.0	5.42 (48)	30	10.2 (90)	1.9	2098-*xxx*-030
		16.8	5.99 (53)	67	19.8 (175)		2098-*xxx*-075
MPF-A4530K	4000	15.0	6.21 (55)	30	11.3 (100)	2.3	2098-*xxx*-030
		19.5	8.13 (72)	62	20.3 (180)		2098-*xxx*-075
MPF-A4540F	3000	15.0	8.25 (73)	30	15.8 (140)	2.5	2098-*xxx*-030
		18.4	10.2 (90)	58	27.1 (240)		2098-*xxx*-075
MPF-A540K	4000	41.5	19.4 (172)	120	48.6 (430)	4.1	2098-*xxx*-150

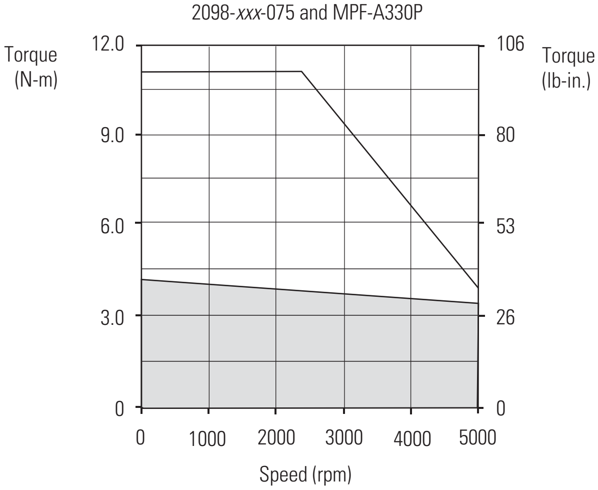
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

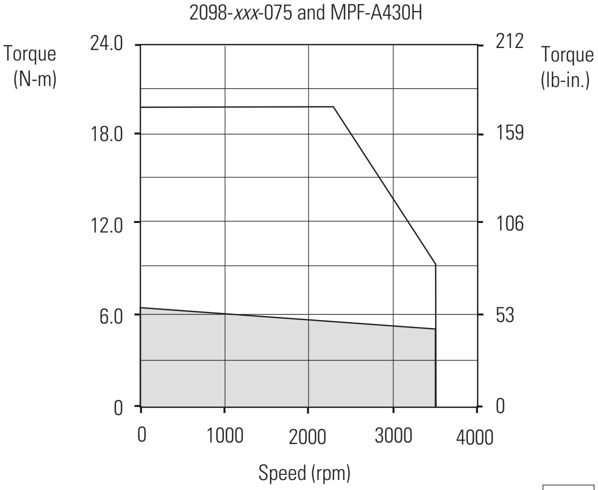
Ultra3000/5000 (230V) Drives/MP-Series Food Grade Motor Curves

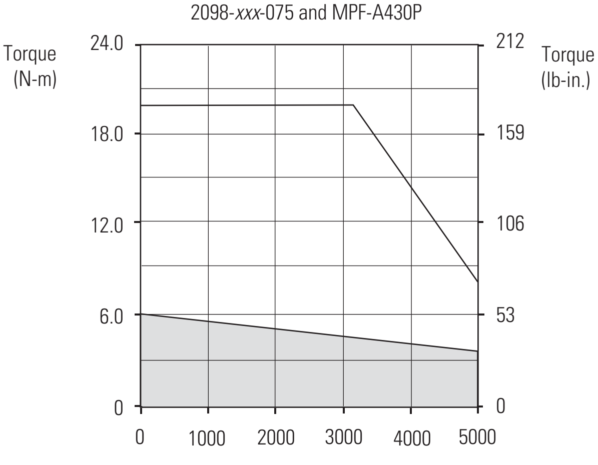








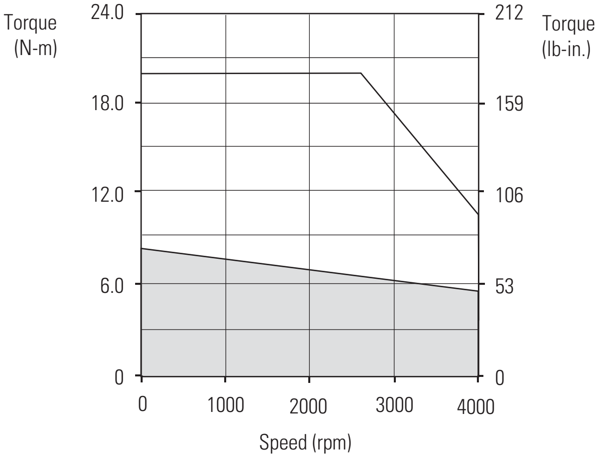


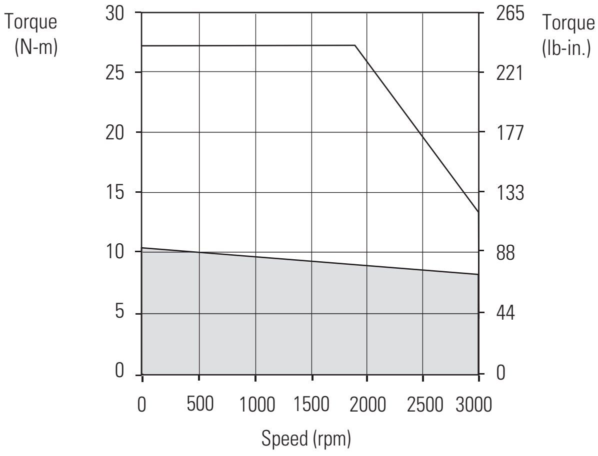


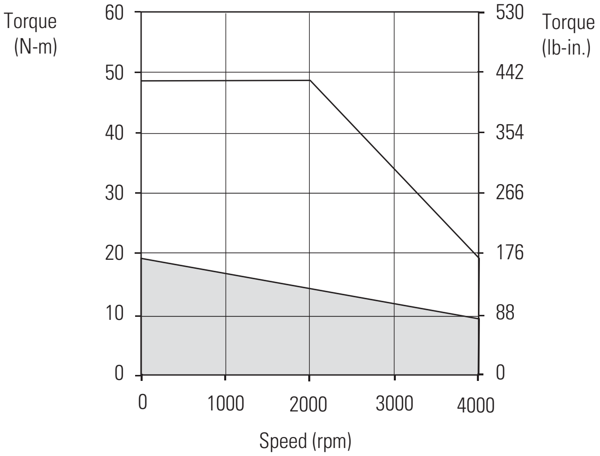
☐ = Intermittent operating region

▨ = Continuous operating region

Ultra3000/5000 (230V) Drives/MP-Series Food Grade Motor Curves, Continued


2098-*xxx*-075 and MPF-A4530K


2098-*xxx*-075 and MPF-A4540F


2098-*xxx*-150 and MPF-A540K

☐ = Intermittent operating region
▨ = Continuous operating region

Ultra3000/5000 (460V) Drives with MP-Series Food Grade Motors

This section provides system combination information for the Ultra3000/5000 (460V) drives when matched with MP-Series food-grade motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin MPF Motor Cable Combinations

Motor Cat. No. (400V class)	Motor Power Cable	Motor Feedback Cable [1]
MPF-B310P, MPF-B320P, MPF-B330P	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback
MPF-B430P		
MPF-B4530K, MPF-B4540F		
MPF-B540K	2090-CP*x*M7DF-10AA*xx* (standard) 2090-CP*x*M7DF-10AF*xx* (continuous-flex)	

[1] Use drive-mounted breakout board (catalog number 2090-UXBB-DM15) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

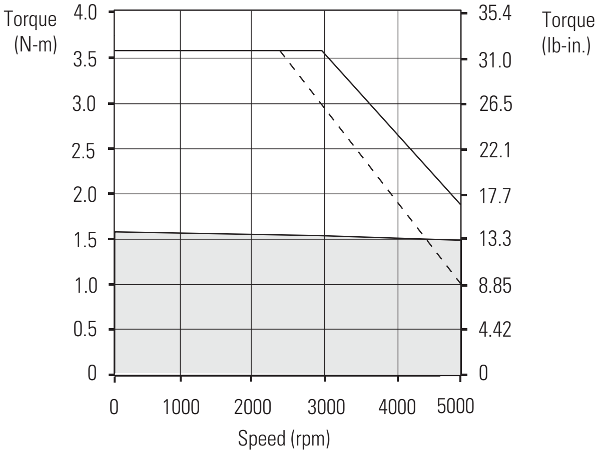
Bulletin MPF Motor Performance Specifications with Ultra3000/5000 (460V) Drives

Rotary Motor	Max Speed rpm	System Continuous Stall Current A (0-pk)	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A (0-pk)	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Ultra3000/ Ultra5000 460V Drives
MPF-B310P	5000	2.30	1.58 (14)	7.1	3.61 (32)	0.77	2098-*xxx*-HV030
MPF-B320P	5000	4.24	3.05 (27)	14.0	7.34 (65)	1.5	2098-*xxx*-HV030
MPF-B330P	5000	5.70	4.18 (37)	14.0	8.59 (76)	1.6	2098-*xxx*-HV030
				19.0	11.1 (98)		2098-*xxx*-HV050
MPF-B430P	5000	9.20	6.55 (58)	22.0	12.9 (114)	2.0	2098-*xxx*-HV050
				32.0	19.8 (175)		2098-*xxx*-HV100
MPF-B4530K	4000	9.90	8.25 (73)	22.0	14.5 (128)	2.4	2098-*xxx*-HV050
				31.0	20.3 (180)		2098-*xxx*-HV100
MPF-B4540F	3000	9.10	10.2 (90)	22.0	22.0 (195)	2.5	2098-*xxx*-HV050
				29.0	27.1 (240)		2098-*xxx*-HV100
MPF-B540K	4000	20.5	19.4 (172)	46.0	33.9 (300)	4.1	2098-*xxx*-HV100
				60.0	45.2 (400)		2098-*xxx*-HV150

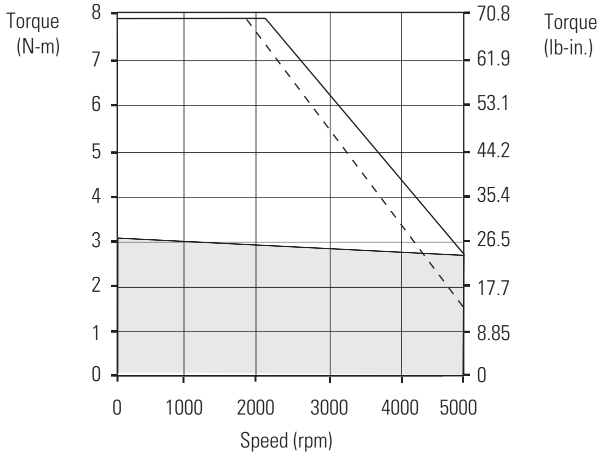
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Ultra3000/5000 (460V) Drives/MP-Series Food Grade Motor Curves

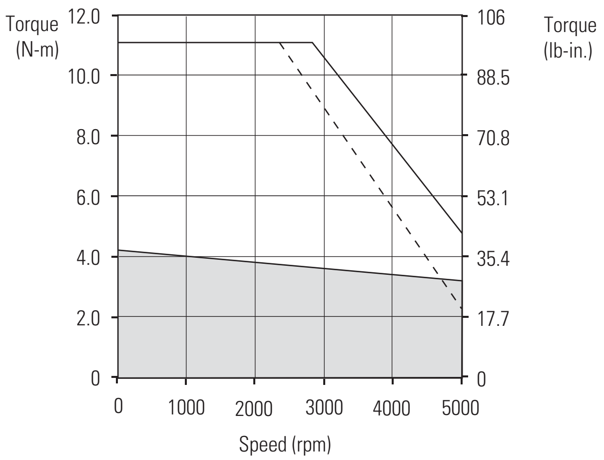
2098-*xxx*-HV030 and MPF-B310P



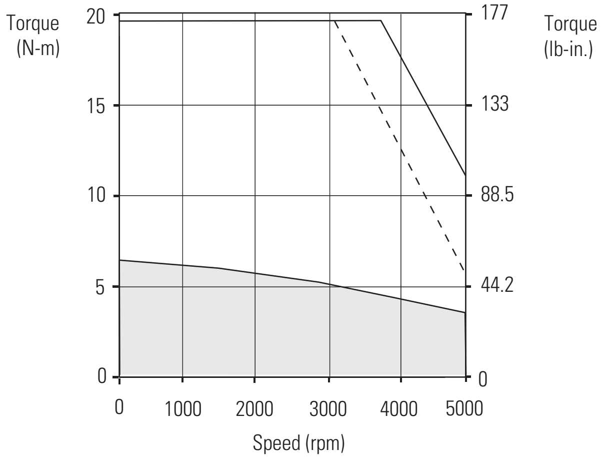
2098-*xxx*-HV030 and MPF-B320P



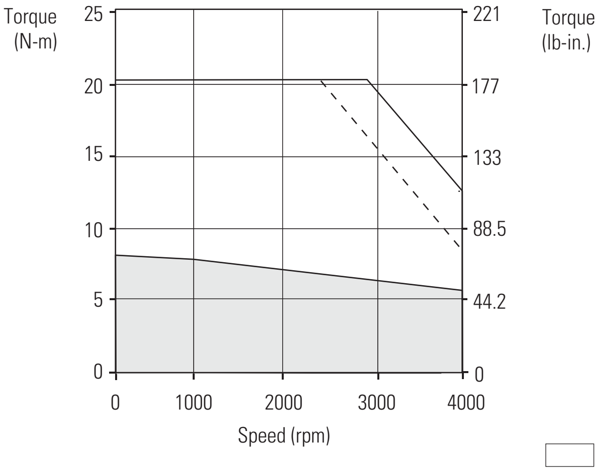
2098-*xxx*-HV050 and MPF-B330P



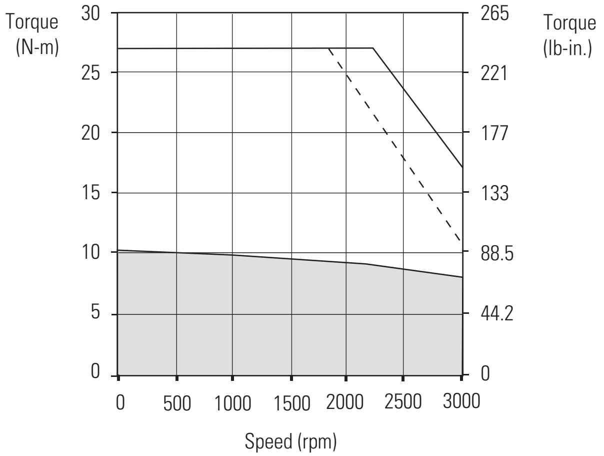
2098-*xxx*-HV100 and MPF-B430P



2098-*xxx*-HV100 and MPF-B4530K

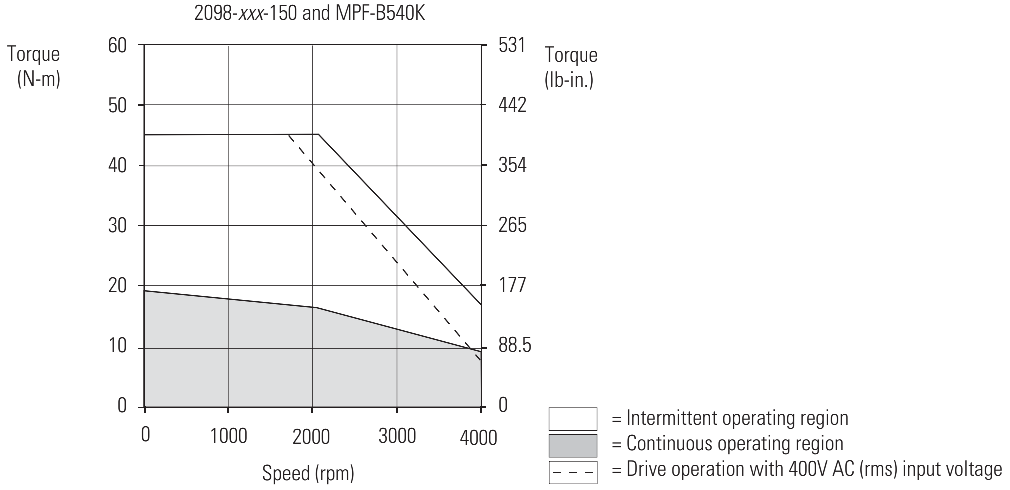


2098-*xxx*-HV100 and MPF-B4540F



☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Ultra3000/5000 (460V) Drives/MP-Series Food Grade Motor Curves, Continued



Ultra3000/5000 (230V) Drives with MP-Series Stainless Steel Motors

This section provides system combination information for the Ultra3000/5000 (230V) drives when matched with MP-Series stainless-steel motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin MPS Motor Cable Combinations

Motor Cat. No. (200V class)	Motor Power Cable	Motor Feedback Cable [1]
MPS-A330P	2090-XXNPMF-16S*xx* [2]	2090-XXNFMF-S*xx* [3] Absolute High-resolution Feedback
MPS-A4540F		

(1) Use drive-mounted breakout board (catalog number 2090-UXBB-DM15) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

(2) These cables are available as standard (catalog number 2090-XXNPMF-*xx*S*xx*) or continuous-flex (catalog number 2090-CP*x*M4DF-*xx*AF*xx*).

(3) These cables are available as standard (catalog number 2090-XXNFMF-S*xx*) or continuous-flex (catalog number 2090-CFBM4DF-CDAF*xx*).

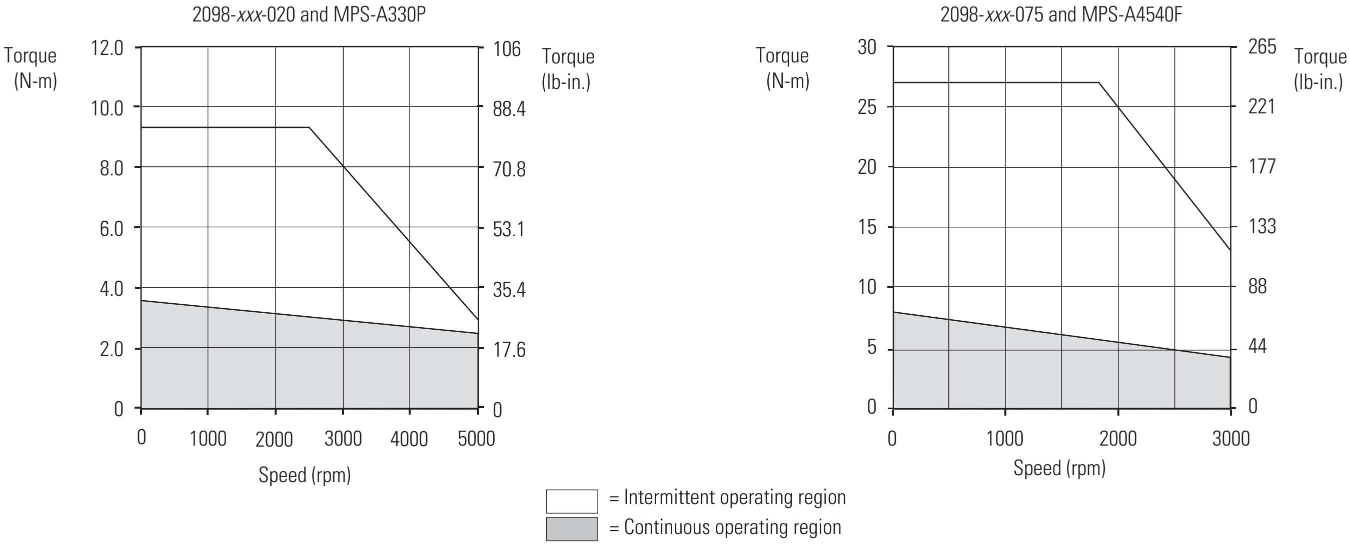
Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Bulletin MPS Motor Performance Specifications with Ultra3000/5000 (230V) Drives

Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Stall Torque N•m (lb•in)	Motor Rated Output kW	Ultra3000/Ultra 5000 230V Drives
MPS-A330P	5000	5.0	1.80 (16)	15.0	5.20 (46)	1.3	2098-*xxx*-010
		9.80	3.60 (32)	30.0	9.30 (82)		2098-*xxx*-020
MPS-A4540F	3000	14.4	8.1 (72)	30.0	15.9 (141)	1.4	2098-*xxx*-030
				56.0	27.1 (240)		2098-*xxx*-075

Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Ultra3000/5000 (230V) Drives/MP-Series Stainless Steel Motor Curves



Ultra3000/5000 (460V) Drives with MP-Series Stainless Steel Motors

This section provides system combination information for the Ultra3000/5000 (460V) drives when matched with MP-Series stainless-steel motors. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin MPS Motor Cable Combinations

Motor Cat. No. (400V class)	Motor Power Cable	Motor Feedback Cable (1)
MPS-B330P	2090-XXNPMF-16S*xx* (2)	2090-XXNFMF-S*xx* (3) Absolute High-resolution Feedback
MPS-B4540F		
MPS-B560F	2090-XXNPMF-14S*xx* (2)	

(1) Use drive-mounted breakout board (catalog number 2090-UXBB-DM15) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

(2) These cables are available as standard (catalog number 2090-XXNPMF-*xx*S*xx*) or continuous-flex (catalog number 2090-CP*x*M4DF-*xx*AF*xx*).

(3) These cables are available as standard (catalog number 2090-XXNFMF-S*xx*) or continuous-flex (catalog number 2090-CFBM4DF-CDAF*xx*).

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

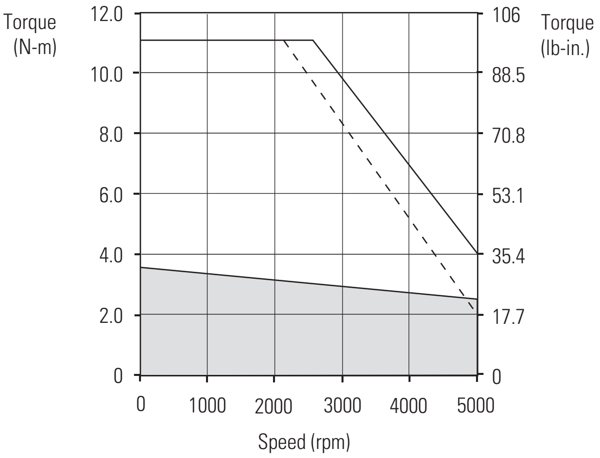
Bulletin MPS Motor Performance Specifications with Ultra3000/5000 (460V) Drives

Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Stall Torque** N•m (lb•in)	**Motor Rated Output** kW	**Ultra3000/ Ultra5000 460V Drives**
MPS-B330P	5000	4.90	3.6 (32)	14.0	8.80 (78)	1.3	2098-*xxx*-HV030
				19.0	11.10 (98)		2098-*xxx*-HV050
MPS-B4540F	3000	7.0	8.0 (71)	14.0	15.6 (138)	1.4	2098-*xxx*-HV030
		7.1	8.1 (72)	22.0	23.5 (208)		2098-*xxx*-HV050
				26.0	27.1 (240)		2098-*xxx*-HV100
MPS-B560F	3000	17.0	21.5 (190)	46.0	50.1 (443)	3.5	2098-*xxx*-HV100
				68.0	67.7 (599)		2098-*xxx*-HV150

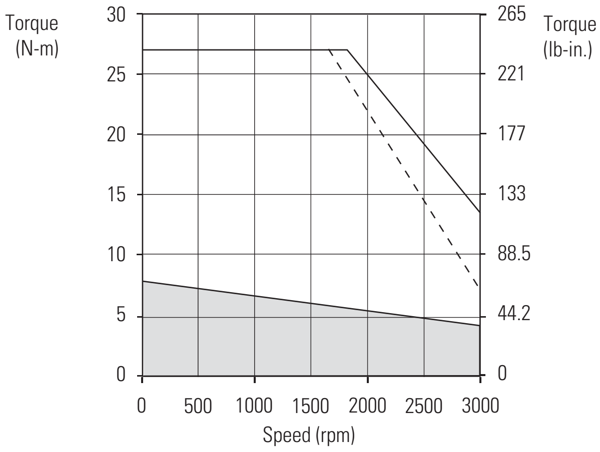
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Ultra3000/5000 (460V) Drives/MP-Series Stainless Steel Motor Curves

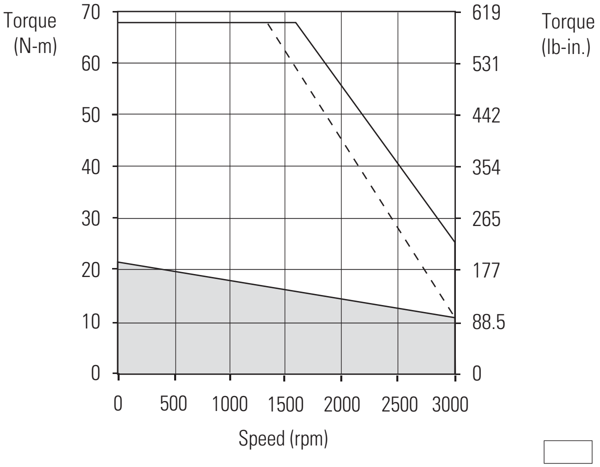
2098-*xxx*-HV050 and MPS-B330P



2098-*xxx*-HV100 and MPS-B4540F



2098-*xxx*-HV150 and MPS-B560F



☐ = Intermittent operating region
▨ = Continuous operating region
- - - = Drive operation with 400V AC (rms) input voltage

Ultra3000/5000 Drives with TL-Series Low Inertia Motors

This section provides system combination information for the Ultra3000/5000 (230V) drives when matched with TL-Series (Bulletin TLY) low-inertia motors. Compatible TL-Series motors are equipped with incremental encoder feedback. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and the optimum torque/speed curves.

Bulletin TLY Motor Cable Combinations

Motor Cat. No. (200V class)	Motor Power/Brake Cable	Motor Feedback Cable [1]
TLY-A110T, TLY-A120T, TLY-A130T	2090-CPWM6DF-16AA*xx* (standard) without brake 2090-CPBM6DF-16AA*xx* (standard) with brake	2090-CFBM6DF-CBAA*xx* 2090-CFBM6DD-CCAA*xx* (standard) Incremental Feedback
TLY-A220T, TLY-A230T		
TLY-A2530P, TLY-A2540P		
TLY-A310M		

[1] Use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440 for more information.

TL-Series (Bulletin TLY-A*xxx*) motors are characterized as having 1000 mm (39.4 in.) cable extensions with circular plastic connectors.
Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Bulletin TLY (non-brake) Performance Specifications with Ultra3000/5000 (230V) Drives

Rotary Motor	Max Speed rpm	System Continuous Stall Current A 0-pk	System Continuous Stall Torque N•m (lb•in)	System Peak Stall Current A 0-pk	System Peak Torque N•m (lb•in)	Motor Rated Output kW	Ultra3000/ Ultra5000 230V Drives
TLY-A110T	6000	0.55	0.096 (0.85)	1.50	0.20 (1.75)	0.041	2098-*xxx*-005
TLY-A120T		1.03	0.181 (1.60)	2.50	0.36 (3.20)	0.086	2098-*xxx*-005
TLY-A130T		1.85	0.325 (2.88)	4.90	0.76 (6.70)	0.14	2098-*xxx*-005
TLY-A220T		2.50	0.576 (5.10)	7.50	1.40 (12.4)	0.35	2098-*xxx*-005
		3.50	0.836 (7.40)	7.90	1.48 (13.1)		2098-*xxx*-010
TLY-A230T		5.00	1.17 (10.4)	15.0	2.94 (26.0)	0.44	2098-*xxx*-010
		5.50	1.30 (11.5)	15.5	3.05 (27.0)		2098-*xxx*-020
TLY-A2530P	5000	5.00	1.32 (11.7)	15.0	3.73 (33.0)	0.69	2098-*xxx*-010
		10.0	2.60 (23.0)	21.0	5.20 (46.0)		2098-*xxx*-020
TLY-A2540P		5.00	1.49 (13.2)	15.0	4.40 (39.0)	0.86	2098-*xxx*-010
		10.0	2.94 (26.0)	24.8	7.10 (63.0)		2098-*xxx*-020
TLY-A310M	4500	10.0	3.61 (31.9)	30.0	9.0 (79.6)	0.95	2098-*xxx*-020

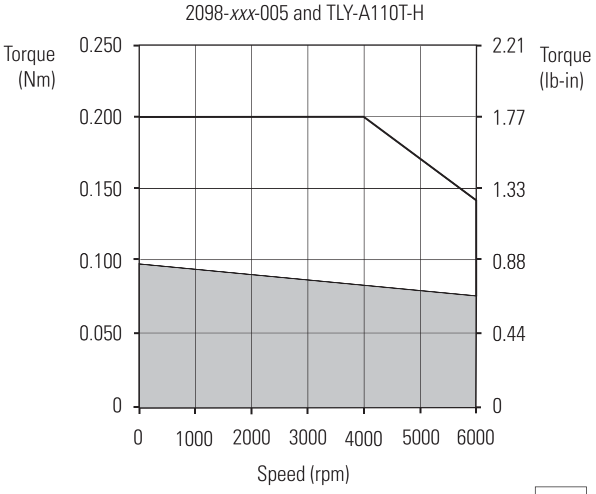
Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

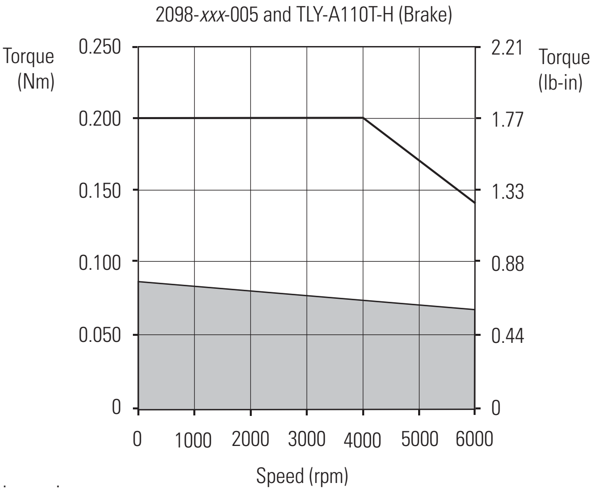
Bulletin TLY (brake) Performance Specifications with Ultra3000/5000 (230V) Drives

Rotary Motor	**Max Speed** rpm	**System Continuous Stall Current** A 0-pk	**System Continuous Stall Torque** N•m (lb•in)	**System Peak Stall Current** A 0-pk	**System Peak Torque** N•m (lb•in)	**Motor Rated Output** kW	**Ultra3000/ Ultra5000 230V Drives**
TLY-A110T	6000	0.50	0.086 (0.76)	1.50	0.20 (1.75)	0.037	2098-*xxx*-005
TLY-A120T		0.93	0.163 (1.44)	2.50	0.36 (3.20)	0.077	2098-*xxx*-005
TLY-A130T		1.67	0.293 (2.59)	4.90	0.76 (6.70)	0.13	2098-*xxx*-005
TLY-A220T		2.50	0.576 (5.10)	7.50	1.40 (12.4)	0.24	2098-*xxx*-005
		3.15	0.757 (6.70)	7.90	1.48 (13.1)		2098-*xxx*-010
TLY-A230T		4.95	1.16 (10.3)	15.0	2.94 (26.0)	0.32	2098-*xxx*-010
				15.5	3.05 (27.0)		2098-*xxx*-020
TLY-A2530P	5000	5.00	1.32 (11.7)	15.0	3.73 (33.0)	0.55	2098-*xxx*-010
		10.0	2.60 (23.0)	21.0	5.20 (46.0)		2098-*xxx*-020
TLY-A2540P		5.0	1.49 (13.2)	15.0	4.40 (39.0)	0.66	2098-*xxx*-010
		10.0	2.94 (26.0)	24.8	7.10 (63.0)		2098-*xxx*-020
TLY-A310M	4500	10.0	3.61 (31.9)	30.0	9.0 (79.6)	0.90	2098-*xxx*-020

Performance specification data and curves reflect nominal system performance of a typical system with motor at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Ultra3000/5000 (230V) Drives/TLY-A*xxxx*-H (incremental) Motor Curves

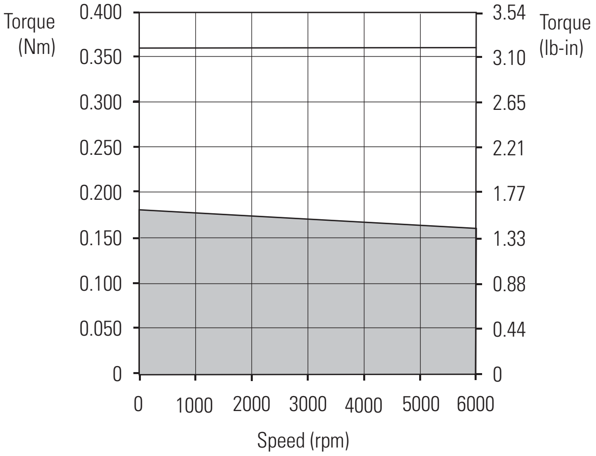




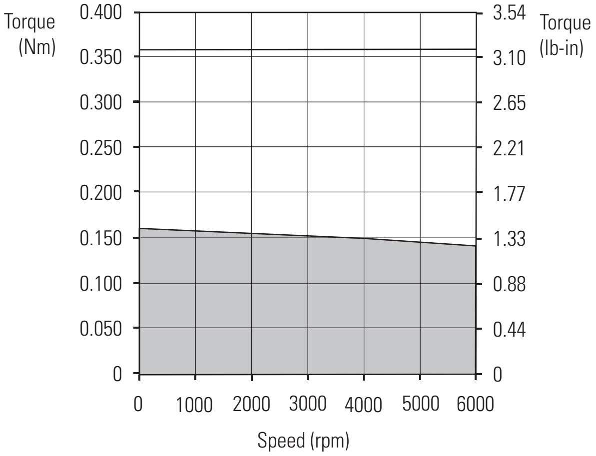
☐ = Intermittent operating region
▨ = Continuous operating region

Ultra3000/5000 (230V) Drives/TLY-A*xxxx*-H (incremental) Motor Curves, Continued

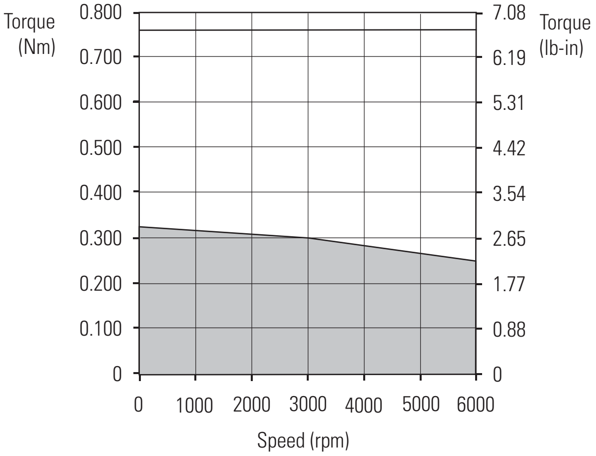
2098-*xxx*-005 and TLY-A120T-H



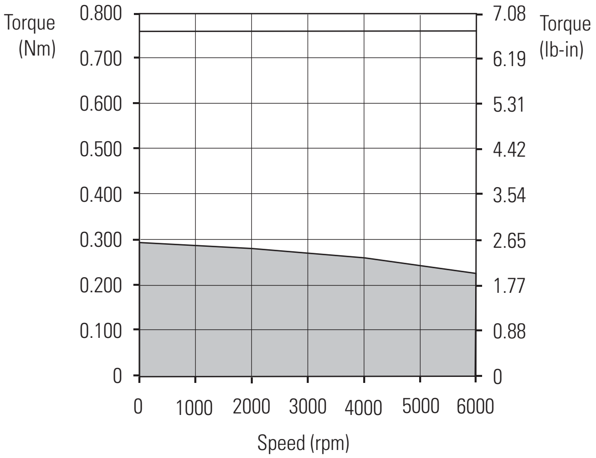
2098-*xxx*-005 and TLY-A120T-H (Brake)



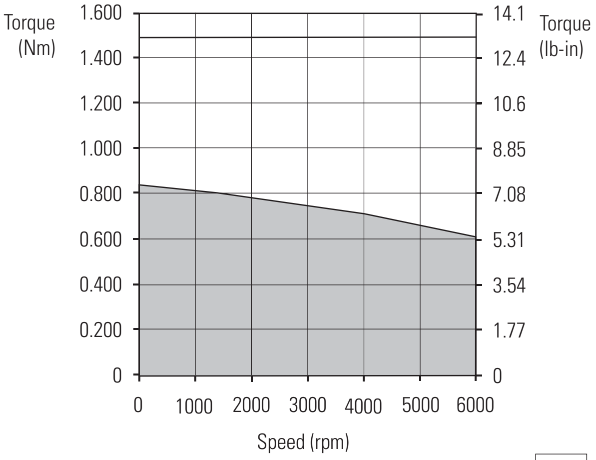
2098-*xxx*-005 and TLY-A130T-H



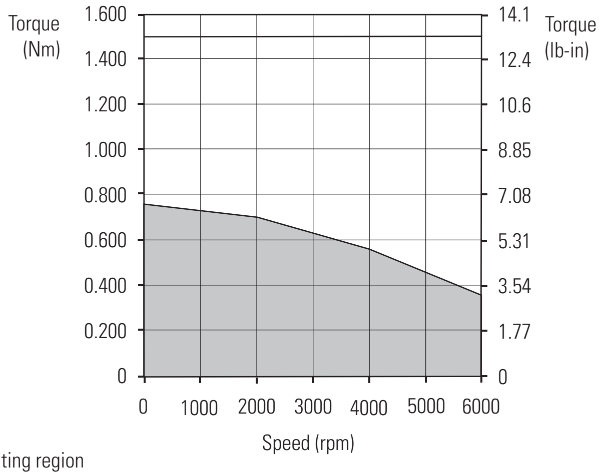
2098-*xxx*-005 and TLY-A130T-H (Brake)



2098-*xxx*-010 and TLY-A220T-H



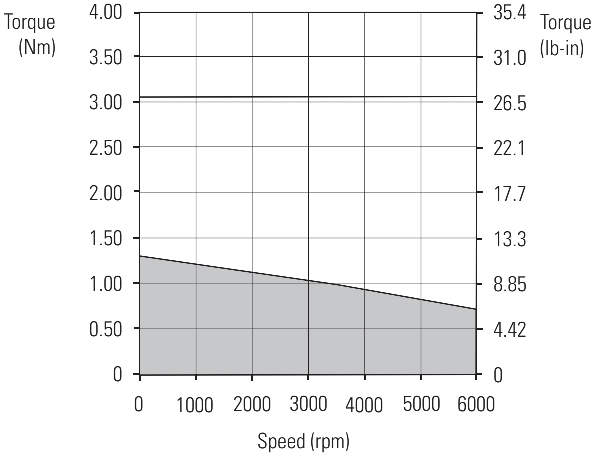
2098-*xxx*-010 and TLY-A220T-H (Brake)



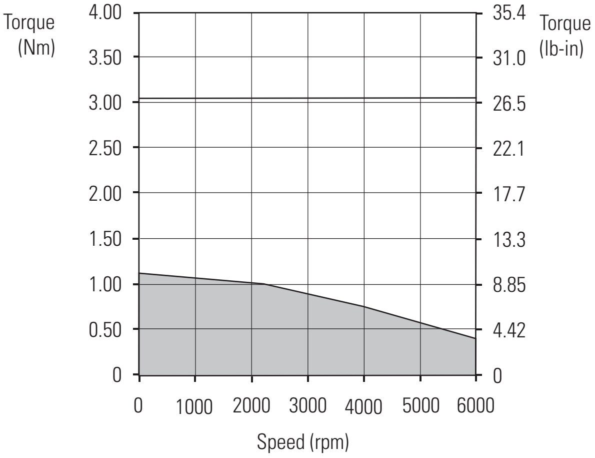
☐ = Intermittent operating region
▨ = Continuous operating region

Ultra3000/5000 (230V) Drives/TLY-A*xxxx*-H (incremental) Motor Curves, Continued

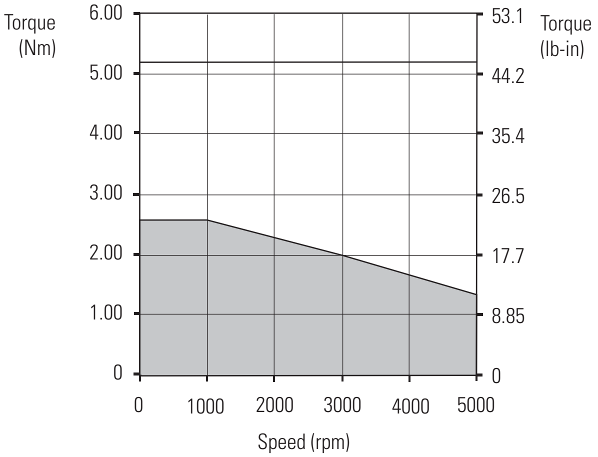
2098-*xxx*-020 and TLY-A230T-H



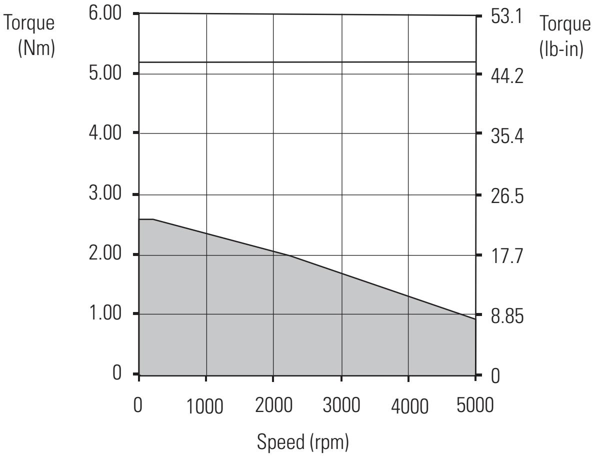
2098-*xxx*-020 and TLY-A230T-H (Brake)



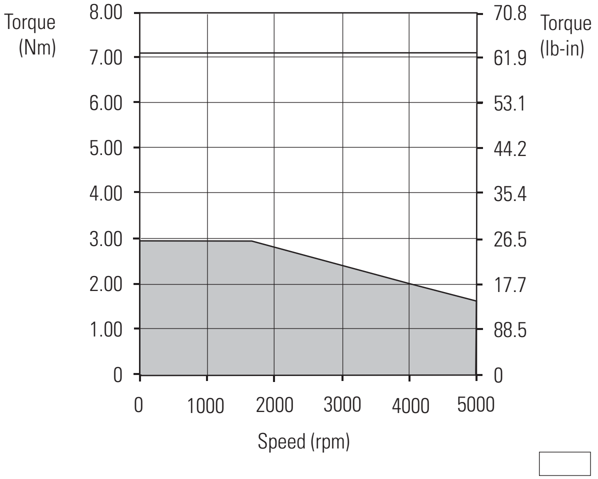
2098-*xxx*-020 and TLY-A2530P-H



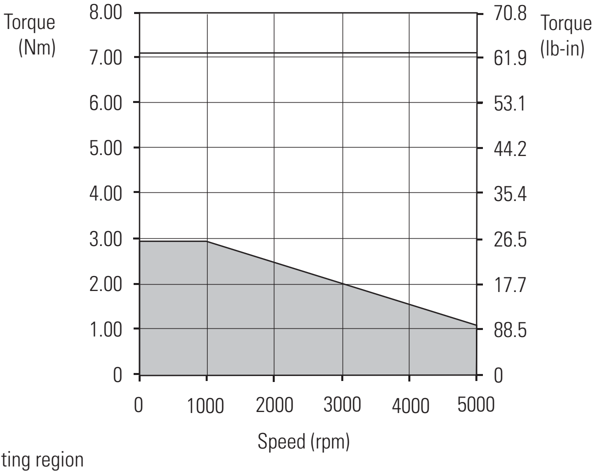
2098-*xxx*-020 and TLY-A2530P-H (Brake)



2098-*xxx*-020 and TLY-A2540P-H

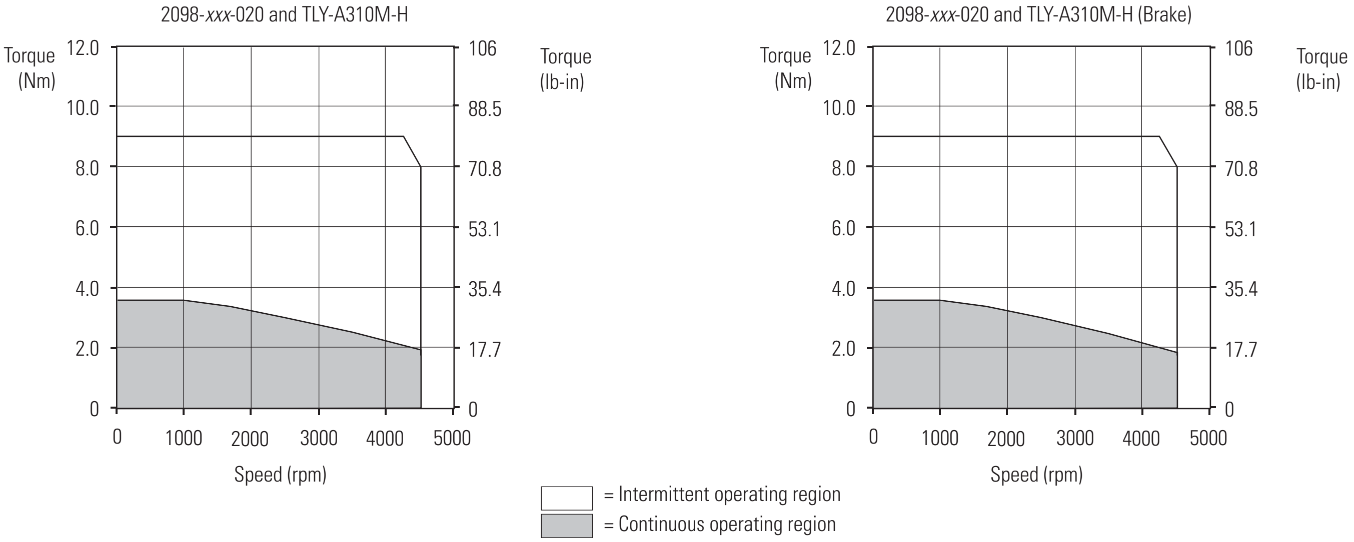


2098-*xxx*-020 and TLY-A2540P-H (Brake)



☐ = Intermittent operating region
▨ = Continuous operating region

Ultra3000/5000 (230V) Drives/TLY-A*xxxx*-H (incremental) Motor Curves, Continued



Linear Motion System Combinations

This chapter provides the Kinetix Motion Control servo drive and actuator combinations. Each drive family/actuator section includes:

- an actuator/cable combinations table.
- a drive/actuator performance specifications table.
- force/velocity curves with each actuator matched to the drive with optimum performance.

Performance specification data and curves reflect nominal system performance of a typical system with actuator/drive at rated ambient temperature and line voltage. For additional information on ambients, line conditions, and valid combinations not shown in this chapter, refer to Motion Analyzer software.

IMPORTANT This system combinations chapter does not include all possible motor/drive combinations. Please refer to Motion Analyzer software to verify compatibility. Download is available at http://www.ab.com/motion/software/analyzer.html.

Linear Motion System Combinations

Drive Family	Linear Motor/Actuator Series	Class	Page
Kinetix 6000 and Kinetix 6200/ Kinetix 6500 Servo Drives	Kinetix 6000 Peak Enhancement Example		666
	MP-Series Integrated Linear Stages	200V	667
	MP-Series Integrated Linear Stages	400V	670
	MP-Series Electric Cylinders	200V and 400V	674
	MP-Series Heavy Duty Electric Cylinders		677
	LDC-Series Linear Motors	200V	681
	LDC-Series Linear Motors	400V	685
	LDL-Series Linear Motors	200V	692
Kinetix 2000 Servo Drives	MP-Series Integrated Linear Stages	200V	696
	MP-Series Electric Cylinders		700
	MP-Series Heavy Duty Electric Cylinders		702
	TL-Series Electric Cylinders		705
	LDC-Series Linear Motors		708
	LDL-Series Linear Motors		712

Drive Family	Linear Motor/Actuator Series	Class	Page
Kinetix 300 Servo Drives	MP-Series Integrated Linear Stages	200V	716
	MP-Series Integrated Linear Stages	400V	718
	MP-Series Electric Cylinders	200V and 400V	720
	MP-Series Heavy Duty Electric Cylinders		722
	TL-Series Electric Cylinders	200V	726
Kinetix 3 Servo Drives	MP-Series Integrated Linear Stages	200V	729
	TL-Series Electric Cylinders		731
	LDC-Series Linear Motors		734
	LDL-Series Linear Motors		738
Ultra3000 Servo Drives	MP-Series Integrated Linear Stages	200V	742
	MP-Series Integrated Linear Stages	400V	746
	MP-Series Electric Cylinders	200V and 400V	749
	MP-Series Heavy Duty Electric Cylinders		752
	LDC-Series Linear Motors	200V	756
	LDC-Series Linear Motors	400V	760
	LDL-Series Linear Motors	200V	766

IMPORTANT You can configure Kinetix 6000 460V (series B) drives to operate with up to 250% peak current for limited duty cycles. Drive/actuator performance specifications are given with and without the peak enhancement feature enabled. For more information, refer to Kinetix 6000 Drive Performance Example with Peak Enhancement Feature on page 666.

Kinetix 6000 Drive Performance Example with Peak Enhancement Feature

The peak current ratings of the Kinetix 6000 AM modules (series A and B) are configured at the factory as 150% of continuous current. You can program 460V (series B) AM modules and the equivalent IAM (inverter) modules, for up to 250% of continuous inverter current. Refer to Peak Enhancement Specifications on page 281 for more information.

IMPORTANT Before your Kinetix 6000 drive will deliver 250% peak performance, you must enable the peak enhancement feature by configuring your drive by using DriveExplorer or RSLogix 5000 software.

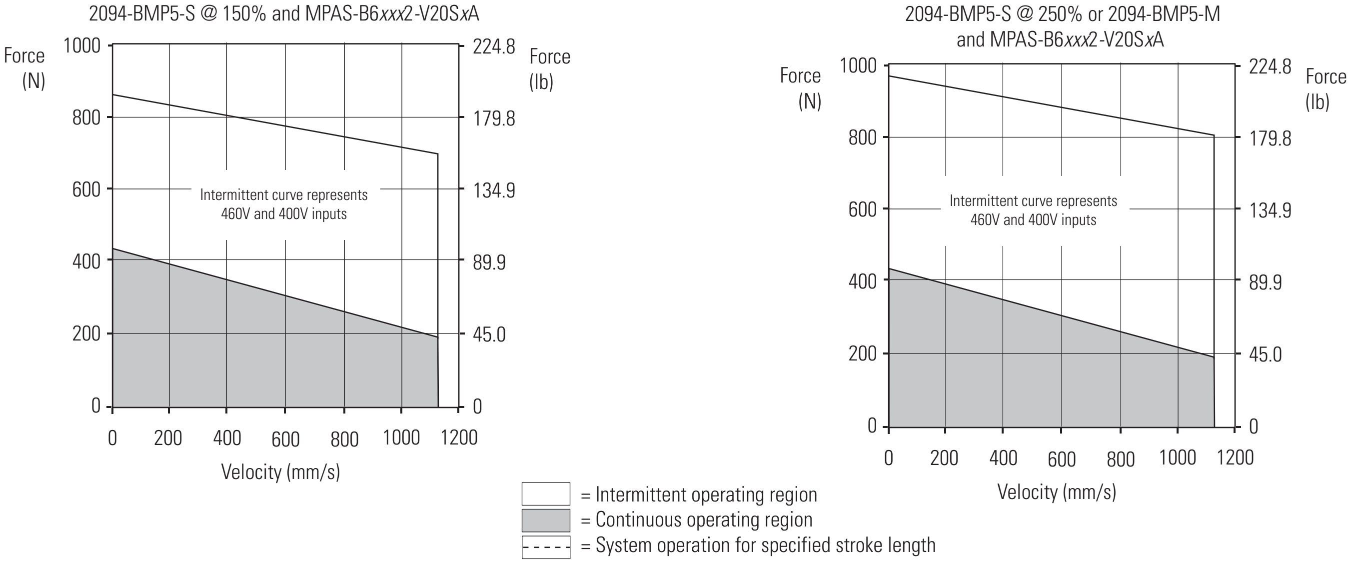
Refer to the Kinetix 6000 Multi-axis Servo Drive User Manual, publication 2094-UM001, to recalculate torque and accel/decel limit values, and paste them into the appropriate Axis Properties dialog box in RSLogix 5000 software.

For sizing your drive/motor combination by using series-B drives and the peak enhancement feature, use Motion Analyzer software, version 4.6 or later.

In this example, the MPAS-B*xxxx*2-V20S*x*A linear stage, usually paired with the 2094-BM01 (series A) AM module, is shown paired with the 2094-BMP5-S (series B) AM module. The two curves illustrate how the 2094-BMP5-S (series B) drive, when configured for 250% peak, can achieve full performance.

Linear Stage Performance Specifications Example with Kinetix 6000 Drives

Linear Stage	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Stage Rated Output** kW	**Kinetix 6000 460V Drives**
MPAS-B*xxxx*2-V20S*x*A	1124 (44.3)	3.30	462 (104)	5.90	865 (194)	0.52	2094-BMP5-S @ 150%
				6.60	968 (218)		2094-BMP5-S @ 250%
							2094-BM01-S @ 150%



Kinetix 6000 (230V) Drives with MP-Series Integrated Linear Stages

This section provides system combination information for the Kinetix 6000 (230V) drives when matched with MP-Series (230V) integrated direct-drive or ballscrew linear stages. Included are motor power and feedback cable catalog numbers, system performance specifications, and force/velocity curves.

Linear Stage Cable Combinations

Linear Stage	Motor Power/Brake Cable	Motor Feedback Cable (1)
MPAS-A*xxxx*1-V05S*x*A, MPAS-A*xxxx*2-V20S*x*A	2090-XXNPMF-16S*xx* (2)	2090-XXNFMF-S*xx* (3) Absolute High-resolution Feedback
MPAS-A6*xxx*B-ALM*x*2C, MPAS-A8*xxx*E-ALM*x*2C, MPAS-A9*xxx*K-ALM*x*2C		2090-XXNFMF-S*xx* (1) Incremental Feedback

(1) Use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

(2) These cables are available as standard (catalog number 2090-XXNPMF-16S*xx*) or continuous-flex (catalog number 2090-CP*x*M4DF-16AF*xx*).

(3) These cables are available as standard (catalog number 2090-XXNFMF-S*xx*) or continuous-flex (catalog number 2090-CFBM4DF-CDAF*xx*).

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Linear Stage Performance Specifications with Kinetix 6000 (230V) Drives

Linear Stage	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Stage Rated Output** kW	Kinetix 6000 230V Drives
MPAS-A*xxxx*1-V05S*x*A	200 (7.9) (1)	3.09	521 (117)	6.10	1212 (272)	0.37	2094-AMP5-S
MPAS-A*xxxx*2-V20S*x*A	1124 (44.3) (2)	4.54	462 (104)	9.10	968 (218)	0.62	2094-AMP5-S
MPAS-A6*xxx*B-ALMO2C	5000 (200)	5.2	103 (23.2)	10.5	231 (51.9)	0.32	2094-AMP5-S
		5.3	105 (23.6)	15.8	359 (80.7)		2094-AM01-S
MPAS-A6*xxx*B-ALMS2C		4.7	83.0 (18.7)	10.5	222 (49.9)	0.29	2094-AMP5-S
				14.2	312 (70.1)		2094-AM01-S
MPAS-A8*xxx*E-ALMO2C		7.0	189 (42.5)	17.0	417 (93.7)	0.53	2094-AM01-S
				18.5	456 (103)		2094-AM02-S
MPAS-A8*xxx*E-ALMS2C		5.2	127 (28.5)	10.5	240 (53.9)	0.48	2094-AMP5-S
		6.3	159 (35.7)	16.7	399 (89.7)		2094-AM01-S
MPAS-A9*xxx*K-ALMO2C		6.7	285 (64.1)	17.0	630 (142)	0.77	2094-AM01-S
				18.3	680 (153)		2094-AM02-S
MPAS-A9*xxx*K-ALMS2C		5.2	206 (46.3)	10.5	372 (83.6)	0.69	2094-AMP5-S
		6.1	245 (55.1)	16.5	601 (135)		2094-AM01-S

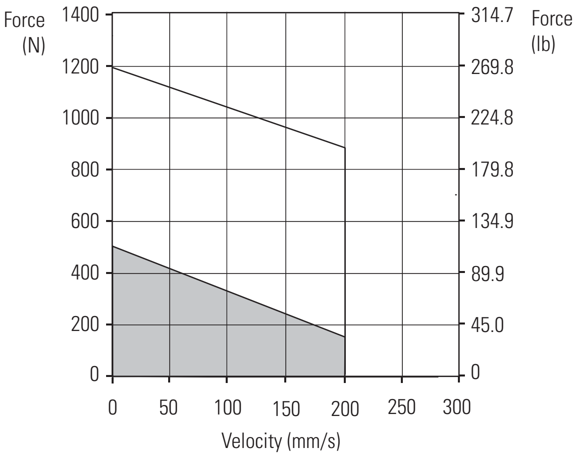
(1) For 900 mm stroke length, maximum speed is 176 mm/s (6.9 in/s). For 1020 mm stroke length, maximum speed is 143 mm/s (5.6 in/s).

(2) For 780 mm stroke length, maximum speed is 889 mm/s (35.0 in/s). For 900 mm stroke length, maximum speed is 715 mm/s (28.2 in/s). For 1020 mm stroke length, maximum speed is 582 mm/s (22.9 in/s).

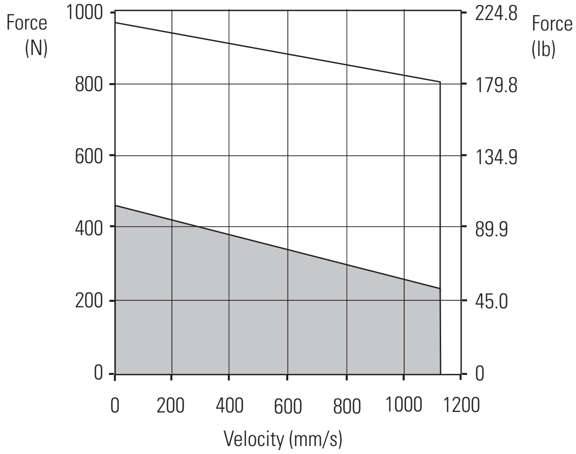
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 6000 (230V) Drives/MP-Series Integrated Linear Stage Curves

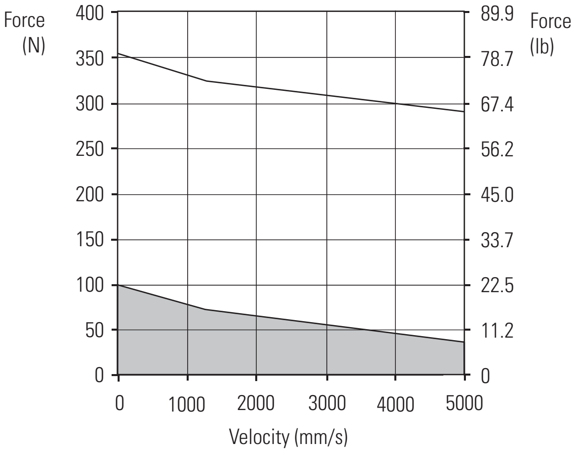
2094-AMP5-S and MPAS-A6*xxx*1-V05S*x*A



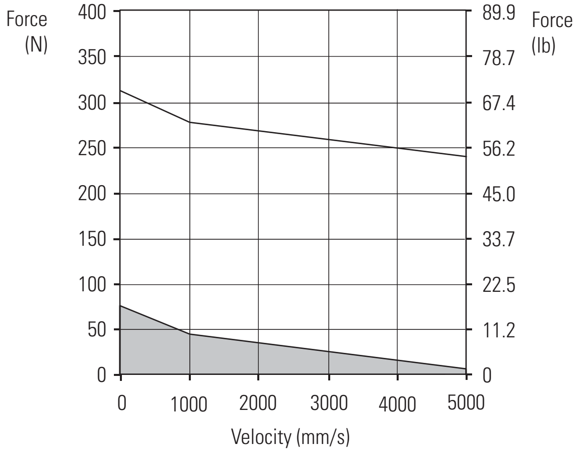
2094-AMP5-S and MPAS-A6*xxx*2-V20S*x*A



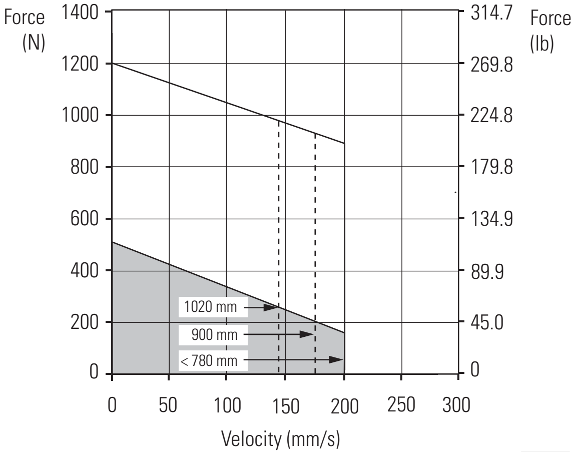
2094-AM01-S and MPAS-A6*xxx*B-ALMO2C



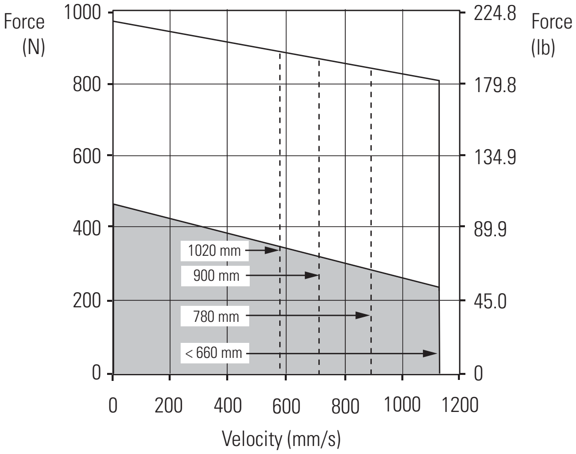
2094-AM01-S and MPAS-A6*xxx*B-ALMS2C



2094-AMP5-S and MPAS-A8*xxx*1-V05S*x*A

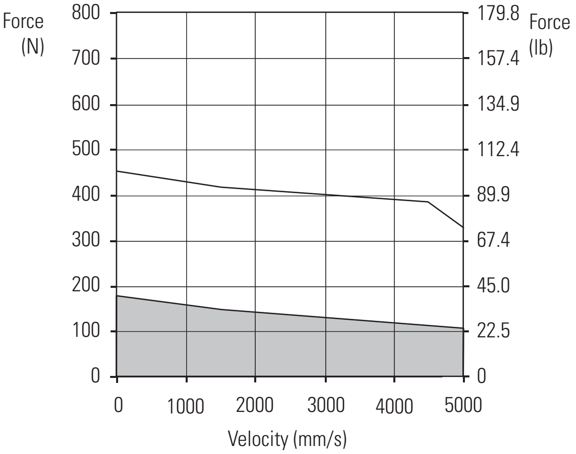


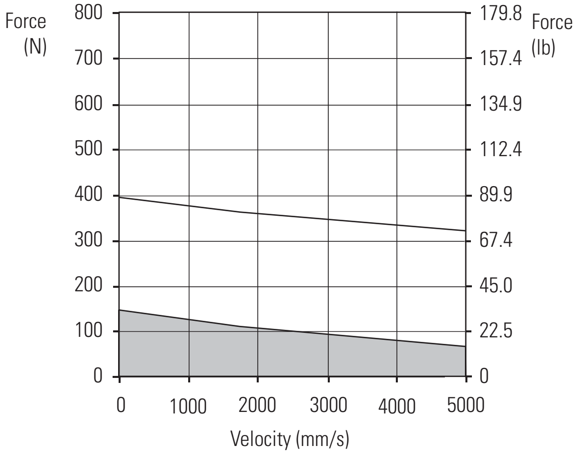
2094-AMP5-S and MPAS-A8*xxx*2-V20S*x*A

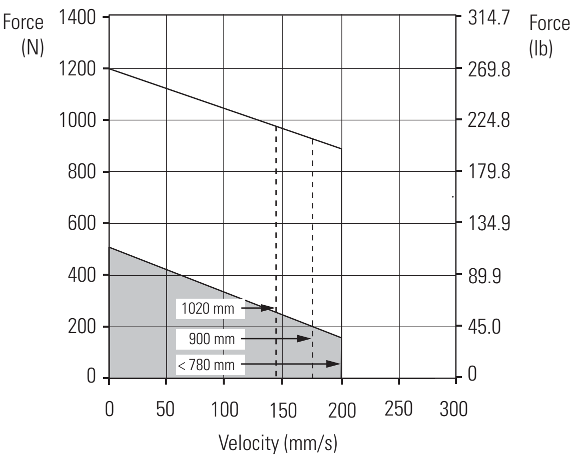


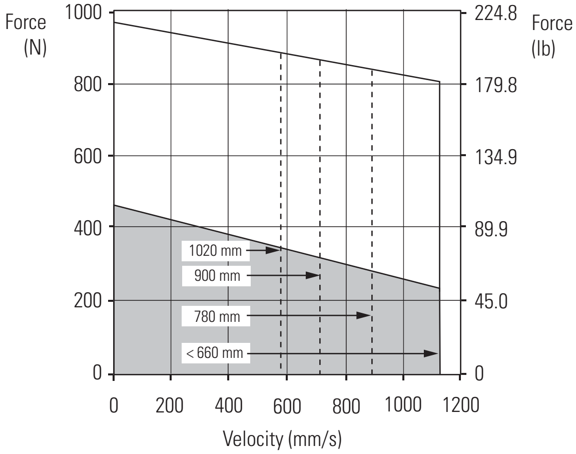
□ = Intermittent operating region
▨ = Continuous operating region
- - - - = System operation for specified stroke length

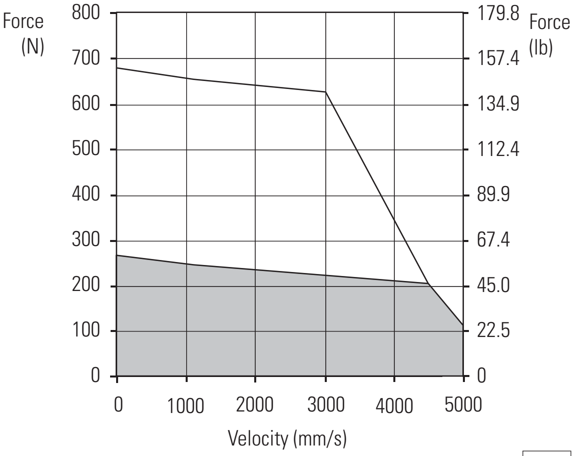
Kinetix 6000 (230V) Drives/MP-Series Integrated Linear Stage Curves, Continued

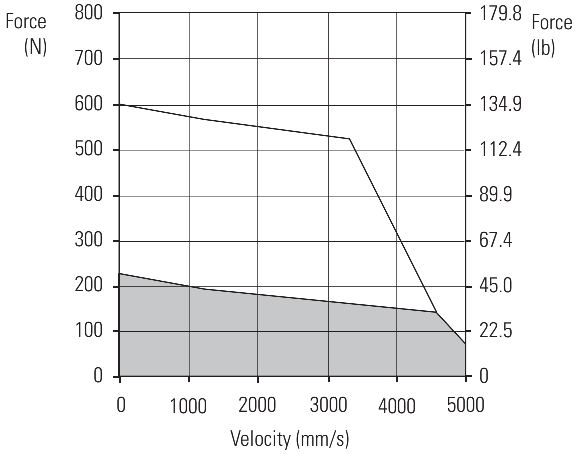

2094-AM02-S and MPAS-A8*xxx*E-ALMO2C


2094-AM01-S and MPAS-A8*xxx*E-ALMS2C


2094-AMP5-S and MPAS-A9*xxx*1-V05S*x*A


2094-AMP5-S and MPAS-A9*xxx*2-V20S*x*A


2094-AM02-S and MPAS-A9*xxx*K-ALMO2C


2094-AM01-S and MPAS-A9*xxx*K-ALMS2C

☐ = Intermittent operating region
▨ = Continuous operating region
- - - - = System operation for specified stroke length

Kinetix 6000 and Kinetix 6200/6500 (460V) Drives with MP-Series Linear Stages

This section provides system combination information for the Kinetix 6000 and Kinetix 6200/6500 (460V) drives when matched with MP-Series (460V) integrated direct-drive or ballscrew linear stages. Included are power/brake and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

IMPORTANT When using Kinetix 6000 (series B) drives, configured for 250% peak performance, you can usually achieve full actuator performance with a smaller drive. Kinetix 6200 and Kinetix 6500 drives are configured for enhanced peak performance by default. Expect the same peak performance from Kinetix 6200/6500 drives and Kinetix 6000 (series B) drives configured for enhanced peak performance.

Refer to Kinetix 6000 IAM/AM Module Series Change on page 276 for more information about using the peak enhancement feature.

Linear Stage Cable Combinations

Linear Stage	Motor Power/Brake Cable	Motor Feedback Cable [1]
MPAS-B*xxxx*1-V05S*x*A	2090-XXNPMF-16S*xx* [2]	2090-XXNFMF-S*xx* [3] Absolute High-resolution Feedback
MPAS-B*xxxx*2-V20S*x*A		
MPAS-B8*xxxx*-ALM*x*2C MPAS-B9*xxxx*-ALM*x*2C		2090-XXNFMF-S*xx* [3] Incremental Feedback

(1) Use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

(2) These cables are available as standard (catalog number 2090-XXNPMF-16S*xx*) or continuous-flex (catalog number 2090-CP*x*M4DF-16AF*xx*).

(3) These cables are available as standard (catalog number 2090-XXNFMF-S*xx*) or continuous-flex (catalog number 2090-CFBM4DF-CDAF*xx*).

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Linear Stage Performance Specifications with Kinetix 6200/Kinetix 6500 (460V) Drives

Linear Stage	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Stage Rated Output** kW	Kinetix 6200/ Kinetix 6500 460V Drives
MPAS-B*xxxx*1-V05S*x*A	200 (7.9) [1]	1.75	521 (117)	3.50	1212 (272)	0.138	2094-BMP5-M
MPAS-B*xxxx*2-V20S*x*A	1124 (44.3) [2]	3.30	462 (104)	6.60	968 (218)	0.52	2094-BMP5-M
MPAS-B8*xxx*F-ALMO2C	5000 (200)	3.50	189 (42.5)	9.30	456 (103)	0.527	2094-BMP5-M
MPAS-B8*xxx*F-ALMS2C	5000 (200)	3.15	159 (35.7)	8.37	399 (89.7)	0.475	2094-BMP5-M
MPAS-B9*xxx*L-ALMO2C	5000 (200)	3.40	285 (64.1)	9.10	680 (153)	0.768	2094-BMP5-M
MPAS-B9*xxx*L-ALMS2C	5000 (200)	3.03	245 (55.1)	8.19	601 (135)	0.69	2094-BMP5-M

(1) For 900 mm stroke length, maximum speed is 176 mm/s (6.9 in/s). For 1020 mm stroke length, maximum speed is 143 mm/s (5.6 in/s).

(2) For 780 mm stroke length, maximum speed is 889 mm/s (35.0 in/s). For 900 mm stroke length, maximum speed is 715 mm/s (28.2 in/s). For 1020 mm stroke length, maximum speed is 582 mm/s (22.9 in/s).

Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Linear Stage Performance Specifications with Kinetix 6000 (460V) Drives

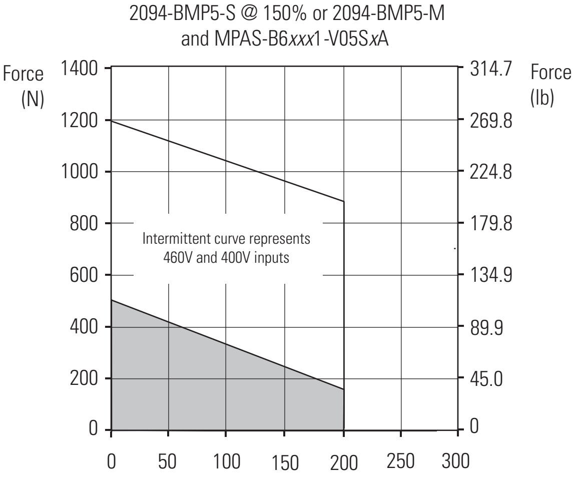
Linear Stage	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Stage Rated Output** kW	**Kinetix 6000 460V Drives**
MPAS-B*xxxx*1-V05S*x*A	200 (7.9) (1)	1.75	521 (117)	3.50	1212 (272)	0.138	2094-BMP5-S @ 150%
MPAS-B*xxxx*2-V20S*x*A	1124 (44.3) (2)	3.30	462 (104)	5.90	865 (194)	0.52	2094-BMP5-S @ 150%
				6.60	968 (218)		2094-BMP5-S @ 250%
MPAS-B8*xxx*F-ALMO2C	5000 (200)	3.50	189 (42.5)	5.90	281 (63.2)	0.527	2094-BMP5-S @ 150%
				9.30	456 (103)		2094-BMP5-S @ 250%
MPAS-B8*xxx*F-ALMS2C	5000 (200)	3.15	159 (35.7)	5.90	272 (61.1)	0.475	2094-BMP5-S @ 150%
				8.37	399 (89.7)		2094-BMP5-S @ 250%
MPAS-B9*xxx*L-ALMO2C	5000 (200)	3.40	285 (64.1)	5.90	433 (97.3)	0.768	2094-BMP5-S @ 150%
				9.10	680 (153)		2094-BMP5-S @ 250%
MPAS-B9*xxx*L-ALMS2C	5000 (200)	3.03	245 (55.1)	5.90	424 (95.3)	0.69	2094-BMP5-S @ 150%
				8.19	601 (135)		2094-BMP5-S @ 250%

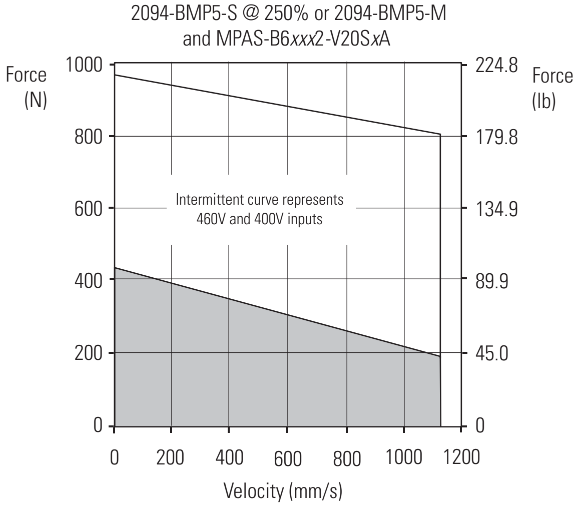
(1) For 900 mm stroke length, maximum speed is 176 mm/s (6.9 in/s). For 1020 mm stroke length, maximum speed is 143 mm/s (5.6 in/s).

(2) For 780 mm stroke length, maximum speed is 889 mm/s (35.0 in/s). For 900 mm stroke length, maximum speed is 715 mm/s (28.2 in/s). For 1020 mm stroke length, maximum speed is 582 mm/s (22.9 in/s).

Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 6000 and Kinetix 6200/6500 (460V) Drives/MP-Series Linear Stage Curves

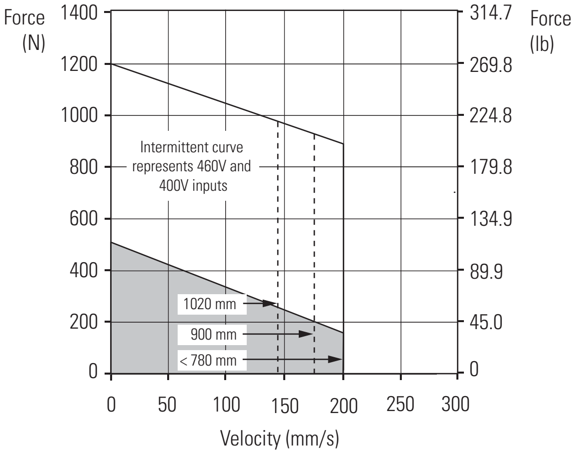

2094-BMP5-S @ 150% or 2094-BMP5-M and MPAS-B6*xxx*1-V05S*x*A


2094-BMP5-S @ 250% or 2094-BMP5-M and MPAS-B6*xxx*2-V20S*x*A

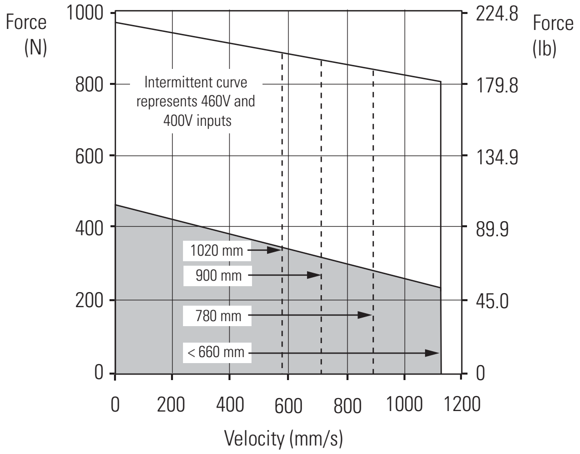
☐ = Intermittent operating region
▨ = Continuous operating region
– – – = System operation with 400V AC (rms) input voltage
- - - - = System operation for specified stroke length

Kinetix 6000 and Kinetix 6200/6500 (460V) Drives/MP-Series Linear Stage Curves, Continued

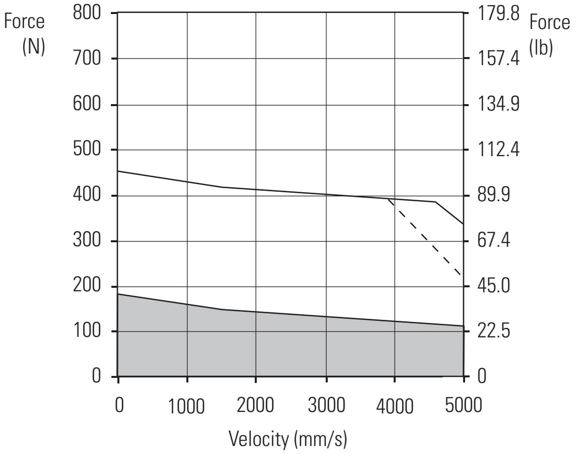
2094-BMP5-S @ 150% or 2094-BMP5-M
and MPAS-B8*xxx*1-V05S*x*A



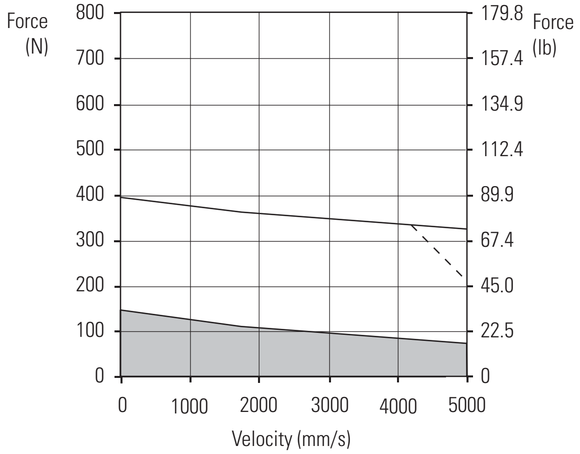
2094-BMP5-S @ 250% or 2094-BMP5-M
and MPAS-B8*xxx*2-V20S*x*A



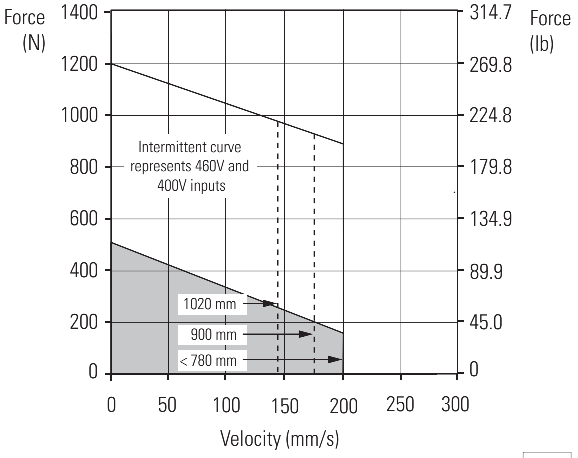
2094-BMP5-S @ 250% or 2094-BMP5-M
and MPAS-B8*xxx*F-ALMO2C



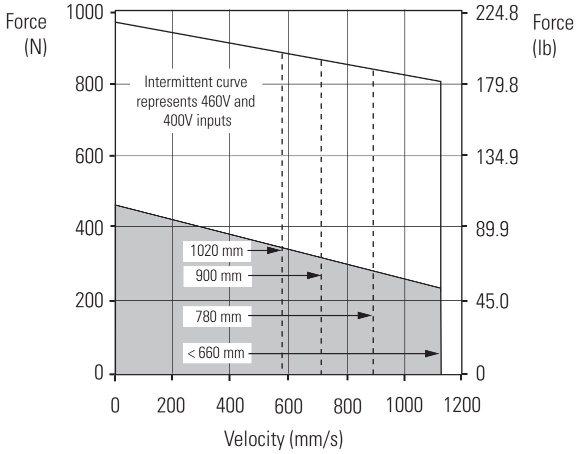
2094-BMP5-S @ 250% or 2094-BMP5-M
and MPAS-B8*xxx*F-ALMS2C



2094-BMP5-S @ 150% or 2094-BMP5-M
and MPAS-B9*xxx*1-V05S*x*A



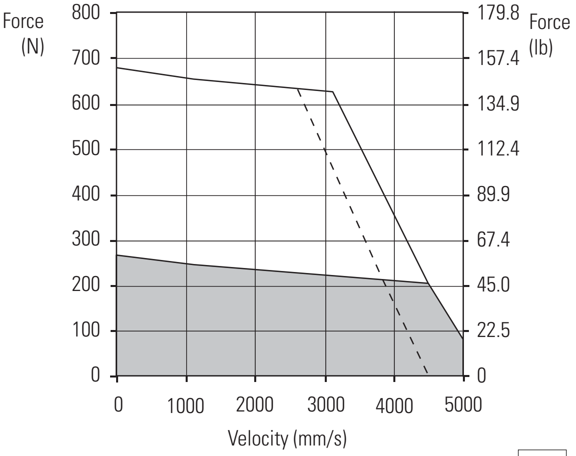
2094-BMP5-S @ 250% or 2094-BMP5-M
and MPAS-B9*xxx*2-V20S*x*A



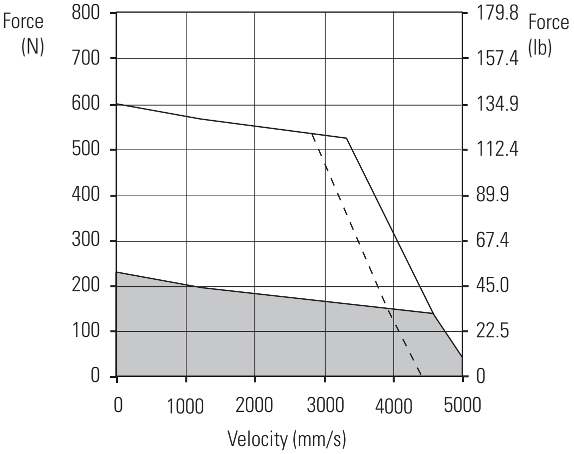
☐ = Intermittent operating region
▨ = Continuous operating region
– – – = System operation with 400V AC (rms) input voltage
- - - - = System operation for specified stroke length

Kinetix 6000 and Kinetix 6200/6500 (460V) Drives/MP-Series Linear Stage Curves, Continued

2094-BMP5-S @ 250% or 2094-BMP5-M
and MPAS-B9*xxx*L-ALMO2C



2094-BMP5-S @ 250% or 2094-BMP5-M
and MPAS-B9*xxx*L-ALMS2C



☐ = Intermittent operating region
▨ = Continuous operating region
– – – = System operation with 400V AC (rms) input voltage
- - - - = System operation for specified stroke length

Kinetix 6000 and Kinetix 6200/6500 Drives with MP-Series Electric Cylinders

This section provides system combination information for the Kinetix 6000 and Kinetix 6200/6500 drives when matched with MP-Series electric cylinders. Included are power/brake and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Electric Cylinder Cable Combinations

Electric Cylinder	Motor Power/Brake Cable	Motor Feedback Cable (1)
MPAR-A/B1*xxx*B MPAR-A/B1*xxx*E MPAR-A/B2*xxx*C MPAR-A/B2*xxx*F	2090-XXNPMF-16S*xx* (2)	2090-XXNFMF-S*xx* (3) Absolute High-resolution Feedback
MPAR-A/B3*xxx*E MPAR-A/B3*xxx*H	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback

(1) Use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

(2) These cables are available as standard (catalog number 2090-XXNPMF-16S*xx*) or continuous-flex (catalog number 2090-CP*x*M4DF-16AF*xx*).

(3) These cables are available as standard (catalog number 2090-XXNFMF-S*xx*) or continuous-flex (catalog number 2090-CFBM4DF-CDAF*xx*).

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Electric Cylinder Performance Specifications with Kinetix 6000 and Kinetix 6200/6500 Drives

Performance Specifications with Kinetix 6200/6500 (460V) Drives

Electric Cylinder	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Rated Output** kW	**Kinetix 6200/ Kinetix 6500 460V Drives**
MPAR-B1*xxx*B	150	1.15	240 (53.9)	1.35	300 (67.4)	0.036	2094-BMP5-M
MPAR-B1*xxx*E	500	1.49	280 (62.9)	1.71	350 (78.7)	0.140	2094-BMP5-M
MPAR-B2*xxx*C	250	1.67	420 (94.4)	1.90	525 (118)	0.105	2094-BMP5-M
MPAR-B2*xxx*F	640	3.29	640 (144)	3.93	800 (180)	0.410	2094-BMP5-M
MPAR-B3*xxx*E	500	5.16	2000 (450)	6.17	2500 (562)	1.00	2094-BM01-M
MPAR-B3*xxx*H	1000	6.13	1300 (292)	6.79	1625 (365)	1.30	2094-BM01-M

Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Performance Specifications with Kinetix 6000 (230V) Drives

Electric Cylinder	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Rated Output** kW	**Kinetix 6000 230V Drives**
MPAR-A1*xxx*B	150	1.15	240 (53.9)	1.35	300 (67.4)	0.036	2094-AMP5-S
MPAR-A1*xxx*E	500	2.16	280 (62.9)	2.48	350 (78.7)	0.140	2094-AMP5-S
MPAR-A2*xxx*C	250	2.42	420 (94.4)	2.72	525 (118)	0.105	2094-AMP5-S
MPAR-A2*xxx*F	640	4.54	640 (144)	5.41	800 (180)	0.410	2094-AM01-S
MPAR-A3*xxx*E	500	10.33	2000 (450)	12.34	2500 (562)	1.00	2094-AM02-S
MPAR-A3*xxx*H	1000	12.20	1300 (292)	16.40	1625 (365)	1.30	2094-AM02-S

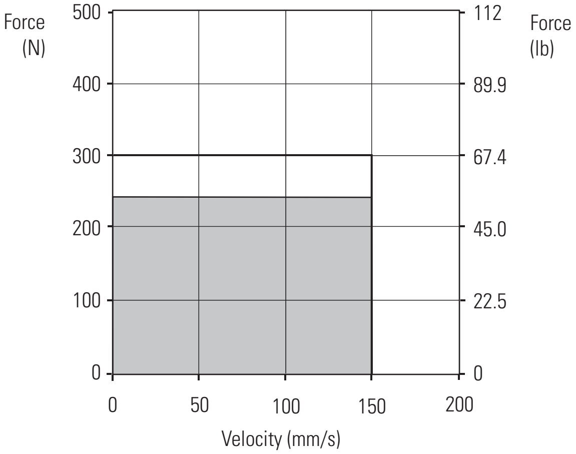
Performance Specifications with Kinetix 6000 (460V) Drives

Electric Cylinder	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Rated Output** kW	**Kinetix 6000 460V Drives**
MPAR-B1*xxx*B	150	1.15	240 (53.9)	1.35	300 (67.4)	0.036	2094-BMP5-S @ 150%
MPAR-B1*xxx*E	500	1.49	280 (62.9)	1.71	350 (78.7)	0.140	2094-BMP5-S @ 150%
MPAR-B2*xxx*C	250	1.67	420 (94.4)	1.90	525 (118)	0.105	2094-BMP5-S @ 150%
MPAR-B2*xxx*F	640	3.29	640 (144)	3.93	800 (180)	0.410	2094-BMP5-S @ 150%
MPAR-B3*xxx*E	500	4.0	1532 (344)	6.17	2500 (562)	1.00	2094-BMP5-S @ 250%
		5.16	2000 (450)				2094-BM01-S @ 150%
MPAR-B3*xxx*H	1000	4.0	841 (189)	6.79	1625 (365)	1.30	2094-BMP5-S @ 250%
		6.13	1300 (292)				2094-BM01-S @ 150%

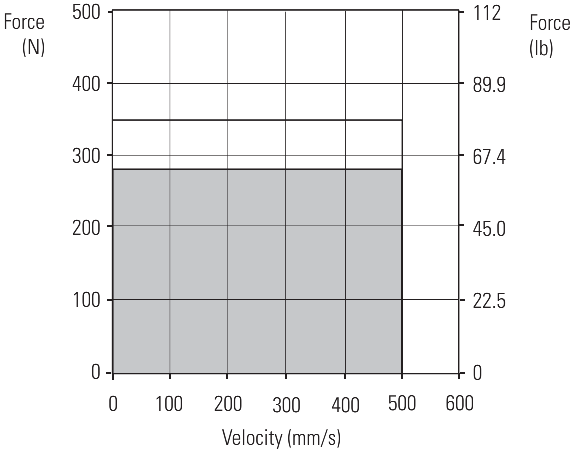
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 6000 and Kinetix 6200/6500 Drives/MP-Series Electric Cylinder Curves

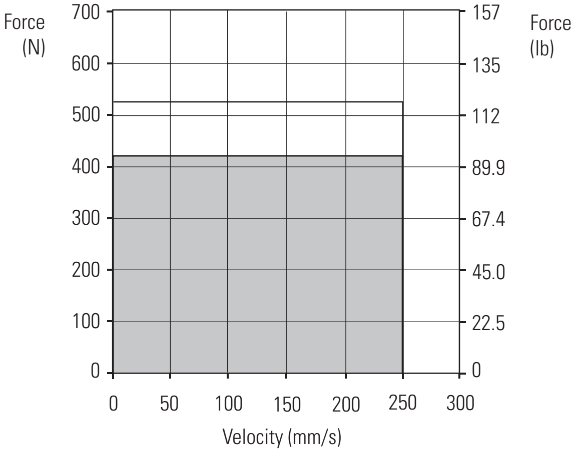
2094-AMP5-S and MPAR-A1*xxx*B-V*x*A
2094-BMP5-S @ 150% or 2094-BMP5-M
and MPAR-B1*xxx*B-V*x*A



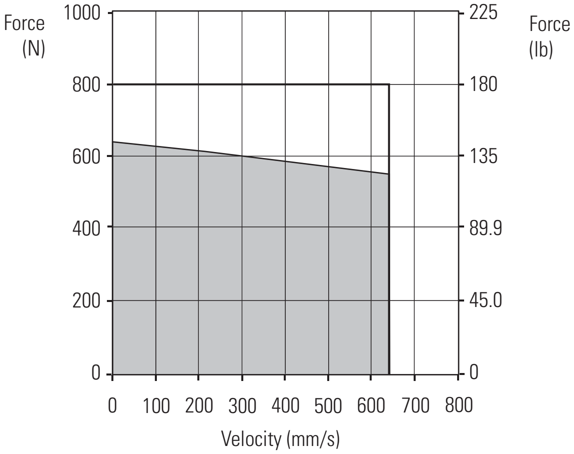
2094-AMP5-S and MPAR-A1*xxx*E-V*x*A
2094-BMP5-S @ 150% or 2094-BMP5-M
and MPAR-B1*xxx*E-V*x*A



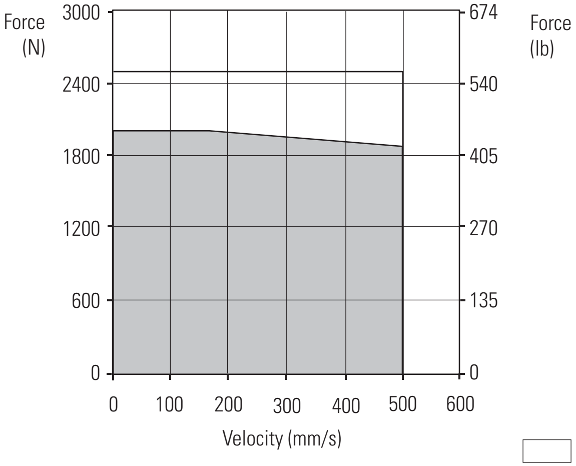
2094-AMP5-S and MPAR-A2*xxx*C-V*x*A
2094-BMP5-S @ 150% or 2094-BMP5-M
and MPAR-B2*xxx*C-V*x*A



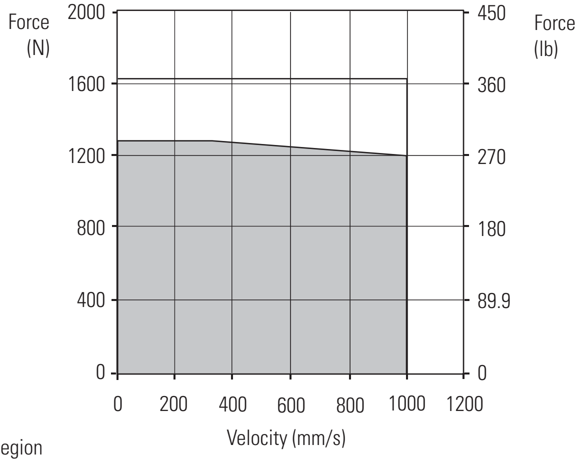
2094-AM01-S and MPAR-A2*xxx*F-V*x*A
2094-BMP5-S @ 150% or 2094-BMP5-M
and MPAR-B2*xxx*F-V*x*A



2094-AM02-S and MPAR-A3*xxx*E-M*x*A
2094-BM01-S @ 150% or 2094-BM01-M
and MPAR-B3*xxx*E-M*x*A



2094-AM02-S and MPAR-A3*xxx*H-M*x*A
2094-BM01-S @ 150% or 2094-BM01-M
and MPAR-B3*xxx*H-M*x*A



☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 6000 and Kinetix 6200/6500 Drives/MP-Series Heavy Duty Electric Cylinders

This section provides system combination information for the Kinetix 6000 and the Kinetix 6200/6500 drives when matched with MP-Series heavy-duty electric cylinders. Included are power/brake and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Electric Cylinder Cable Combinations

Electric Cylinder	Motor Power/Brake Cable	Motor Feedback Cable [1]
MPAI-A/B3*xxx*C MPAI-A/B3*xxx*E MPAI-A/B3*xxx*R MPAI-A/B3*xxx*S	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback
MPAI-A/B4*xxx*C MPAI-A/B4*xxx*E MPAI-A/B4*xxx*R MPAI-A/B4*xxx*S		

[1] Use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Electric Cylinder Performance Specifications with Kinetix 6200/6500 (460V)Drives

Performance Specifications with Ball Screw Electric Cylinders

Electric Cylinder	Maximum Speed mm/s (in/s)	System Continuous Stall Current Amps 0-pk	System Continuous Stall Force N (lb)		System Peak Stall Current Amps 0-pk	System Peak Stall Force N (lb)	Rated Output kW	Kinetix 6200/ Kinetix 6500 460V Drives
			25 °C (77 °F)	40 °C (104 °F)				
MPAI-B3*xxx*C	279 (11)	2.81	4003 (900)	3176 (714)	4.30	4448 (1000)	0.39	2094-BMP5-M
MPAI-B3*xxx*E	559 (22)		2002 (450)	1588 (357)	7.07	4003 (900)		
MPAI-B4*xxx*C	279 (11)	5.61	7784 (1750)	6179 (1389)	8.68	8896 (2000)	0.43	2094-BM01-M
MPAI-B4*xxx*E	559 (22)		3892 (875)	3092 (695)	14.14	7784 (1750)		

Performance Specifications with Roller Screw Electric Cylinders

Electric Cylinder	Maximum Speed mm/s (in/s)	System Continuous Stall Current Amps 0-pk	System Continuous Stall Force N (lb)		System Peak Stall Current Amps 0-pk	System Peak Stall Force N (lb)	Rated Output kW	Kinetix 6200/ Kinetix 6500 460V Drives
			25 °C (77 °F)	40 °C (104 °F)				
MPAI-B3*xxx*R	279 (11)	2.81	3781 (850)	3003 (675)	7.07	7562 (1700)	0.39	2094-BMP5-M
MPAI-B3*xxx*S	559 (22)		1891 (425)	1499 (337)		3781 (850)		
MPAI-B4*xxx*R	279 (11)	5.61	7340 (1650)	5827 (1310)	14.14	14,679 (3300)	0.43	2094-BM01-M
MPAI-B4*xxx*S	559 (22)		3670 (825)	2914 (655)		7340 (1650)		

Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.8 or later.

Electric Cylinder Performance Specifications with Kinetix 6000 (230V) Drives

Performance Specifications with Ball Screw Electric Cylinders

Electric Cylinder	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)		**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Rated Output** kW	**Kinetix 6000 230V Drives**
			25 °C (77 °F)	40 °C (104 °F)				
MPAI-A3*xxx*C	279 (11)	5.61	4003 (900)	3176 (714)	8.40	4448 (1000)	0.39	2094-AM01-S
MPAI-A3*xxx*E	559 (22)		2002 (450)	1588 (357)	14.14	4003 (900)		
MPAI-A4*xxx*C	279 (11)	10.89	7784 (1750)	6179 (1389)	17.07	8896 (2000)	0.43	2094-AM02-S
MPAI-A4*xxx*E	559 (22)		3892 (875)	3092 (695)	27.44	7784 (1750)		

Performance Specifications with Roller Screw Electric Cylinders

Electric Cylinder	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)		**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Rated Output** kW	**Kinetix 6000 230V Drives**
			25 °C (77 °F)	40 °C (104 °F)				
MPAI-A3*xxx*R	279 (11)	5.61	3781 (850)	3003 (675)	14.14	7562 (1700)	0.39	2094-AM01-S
MPAI-A3*xxx*S	559 (22)		1891 (425)	1499 (337)		3781 (850)		
MPAI-A4*xxx*R	279 (11)	10.89	7340 (1650)	5827 (1310)	27.44	14,679 (3300)	0.43	2094-AM02-S
MPAI-A4*xxx*S	559 (22)		3670 (825)	2914 (655)		7340 (1650)		

Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.8 or later.

Electric Cylinder Performance Specifications with Kinetix 6000 (460V) Drives

Performance Specifications with Ball Screw Electric Cylinders

Electric Cylinder	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)		**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Rated Output** kW	**Kinetix 6000 460V Drives**
			25 °C (77 °F)	40 °C (104 °F)				
MPAI-B3*xxx*C	279 (11)	2.81	4003 (900)	3176 (714)	4.30	4448 (1000)	0.39	2094-BMP5-S @ 150%
MPAI-B3*xxx*E	559 (22)		2002 (450)	1588 (357)	7.07	4003 (900)		2094-BMP5-S @ 250%
MPAI-B4*xxx*C	279 (11)	5.61	7784 (1750)	6179 (1389)	8.68	8896 (2000)	0.43	2094-BM01-S @ 150%
MPAI-B4*xxx*E	559 (22)		3892 (875)	3092 (695)	14.14	7784 (1750)		2094-BM01-S @ 250%

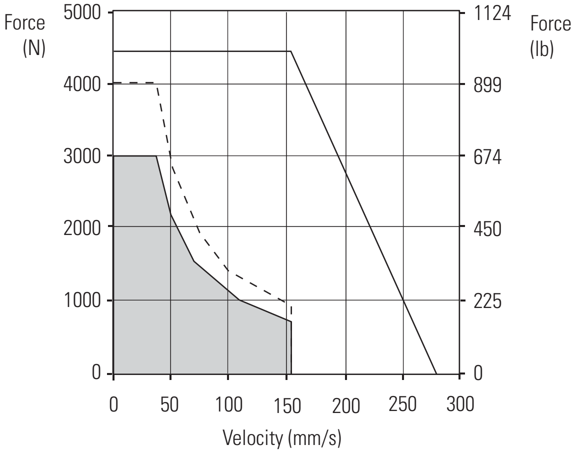
Performance Specifications with Roller Screw Electric Cylinders

Electric Cylinder	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)		**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Rated Output** kW	**Kinetix 6000 460V Drives**
			25 °C (77 °F)	40 °C (104 °F)				
MPAI-B3*xxx*R	279 (11)	2.81	3781 (850)	3003 (675)	7.07	7562 (1700)	0.39	2094-BMP5-S @ 250%
MPAI-B3*xxx*S	559 (22)		1891 (425)	1499 (337)		3781 (850)		
MPAI-B4*xxx*R	279 (11)	5.61	7340 (1650)	5827 (1310)	14.14	14,679 (3300)	0.43	2094-BM01-S @ 250%
MPAI-B4*xxx*S	559 (22)		3670 (825)	2914 (655)		7340 (1650)		

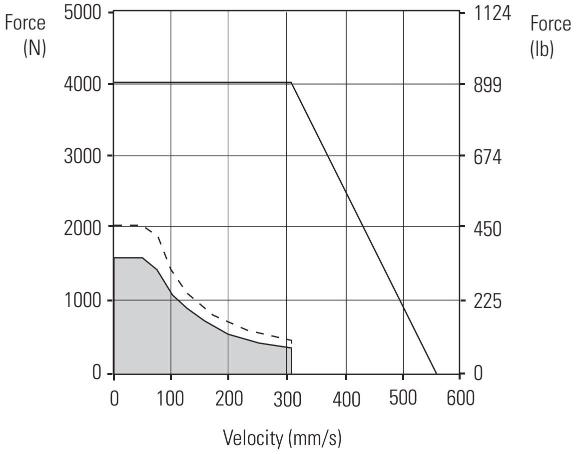
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.8 or later.

Kinetix 6000 and Kinetix 6200/6500 Drives/MP-Series (ball screw) Electric Cylinder Curves

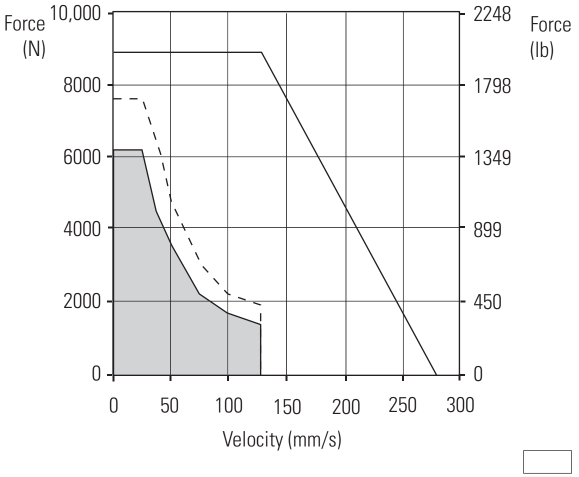
2094-AM01-S and MPAI-A3*xxx*C
2094-BMP5-S @ 150% or 2094-BMP5-M
and MPAI-B3*xxx*C



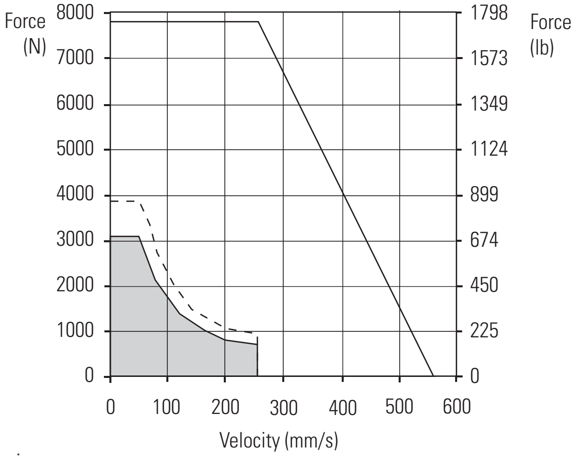
2094-AM01-S and MPAI-A3*xxx*E
2094-BMP5-S @ 250% or 2094-BMP5-M
and MPAI-B3*xxx*E



2094-AM02-S and MPAI-A4*xxx*C
2094-BM01-S @ 150% or 2094-BM01-M
and MPAI-B4*xxx*C



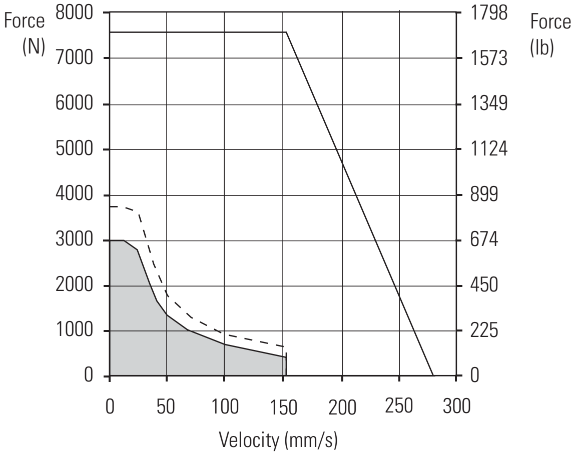
2094-AM02-S and MPAI-A4*xxx*E
2094-BM01-S @ 250% or 2094-BM01-M
and MPAI-B4*xxx*E



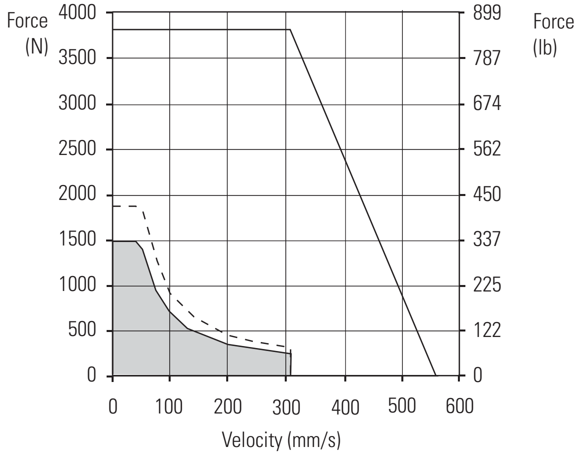
□ = Intermittent operating region
- - - = Continuous operating region @ 25 °C (77 °F)
▨ = Continuous operating region @ 40 °C (104 °F)

Kinetix 6000 and Kinetix 6200/6500 Drives/MP-Series (roller screw) Electric Cylinder Curves

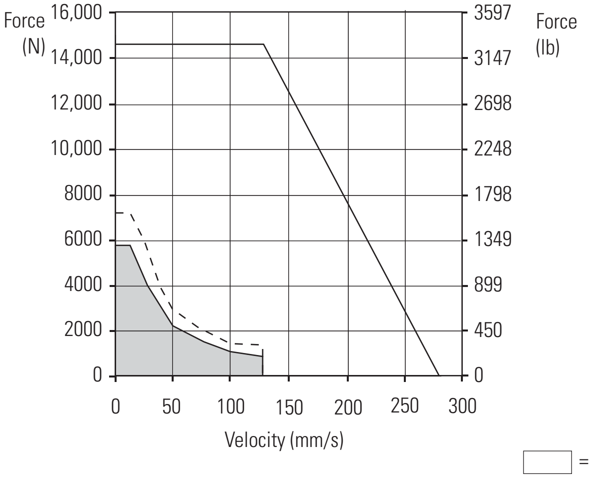
2094-AM01-S and MPAI-A3*xxx*R
2094-BMP5-S @ 250% or 2094-BMP5-M
and MPAI-B3*xxx*R



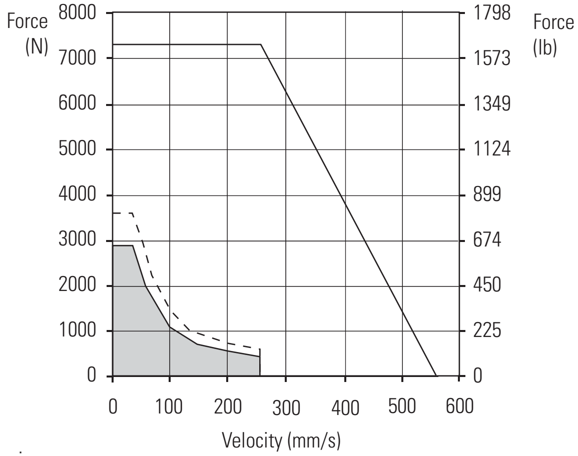
2094-AM01-S and MPAI-A3*xxx*S
2094-BMP5-S @ 250% or 2094-BMP5-M
and MPAI-B3*xxx*S



2094-AM02-S and MPAI-A4*xxx*R
2094-BM01-S @ 250% or 2094-BM01-M
and MPAI-B4*xxx*R



2094-AM02-S and MPAI-A4*xxx*S
2094-BM01-S @ 250% or 2094-BM01-M
and MPAI-B4*xxx*S



☐ = Intermittent operating region
- - - = Continuous operating region @ 25 °C (77 °F)
▨ = Continuous operating region @ 40 °C (104 °F)

Kinetix 6000 (230V) Drives with LDC-Series Linear Motors

This section provides system combination information for the Kinetix 6000 (230V) drives when matched with LDC-Series iron-core linear motors. Included are power and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Linear Motor Cable Combinations

Linear Motor	Motor Power Cable	Motor Feedback Cable [1]
LDC-C030100-DHT, LDC-C030200-DHT, LDC-C030200-EHT	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-XXNFMF-S*xx* (standard) 2090-CFBM7DF-CDAF*xx* (continuous-flex) Sin/Cos or TTL Encoder Feedback
LDC-C050100-DHT, LDC-C050200-DHT, LDC-C050200-EHT, LDC-C050300-DHT, LDC-C050300-EHT		
LDC-C075200-DHT, LDC-C075200-EHT, LDC-C075300-DHT, LDC-C075300-EHT, LDC-C075400-DHT, LDC-C075400-EHT		
LDC-C100300-DHT, LDC-C100300-EHT, LDC-C100400-DHT, LDC-C100400-EHT, LDC-C100600-DHT		
LDC-C150400-DHT, LDC-C150600-DHT		

(1) Use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

LDC-Series Performance Specifications with Kinetix 6000 (230V) Drives

Linear Motor	**Maximum Speed** m/s (ft/s)	**System Continuous Stall Current** [1] Amps 0-pk	**System Continuous Stall Force** [1] N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Motor Rated Output** kW	Kinetix 6000 230V Drives
LDC-C030100-DHT	10.0 (32.8)	4.1…6.1	74…111 (17…25)	12.1	188 (42)	0.37…0.55	2094-AM01-S
LDC-C030200-DHT		8.1…12.2	148…222 (33…50)	24.3	375 (84)	0.74…1.11	2094-AM02-S
LDC-C030200-EHT		4.1…6.1		12.1			2094-AM01-S
LDC-C050100-DHT	10.0 (32.8)	3.9…5.9	119…179 (27…40)	11.7	302 (68)	0.59…0.89	2094-AM01-S
LDC-C050200-DHT		7.9…11.8	240…359 (54…81)	23.3	600 (135)	1.20…1.79	2094-AM02-S
LDC-C050200-EHT		3.9…5.9		11.6			2094-AMP5-S
LDC-C050300-DHT		11.8…17.7	363…544 (82…122)	35.9	941 (212)	1.81…2.72	2094-AM03-S
LDC-C050300-EHT		3.9…5.9		12.0			2094-AMP5-S

(1) Values represent the range between no cooling (low value) and water cooling (high value).

Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

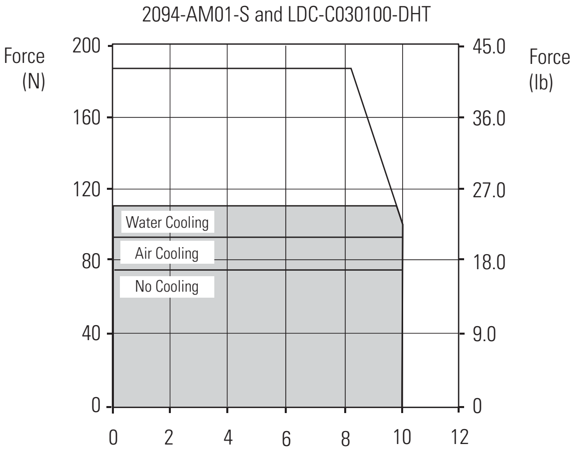
LDC-Series Performance Specifications with Kinetix 6000 (230V) Drives, Continued

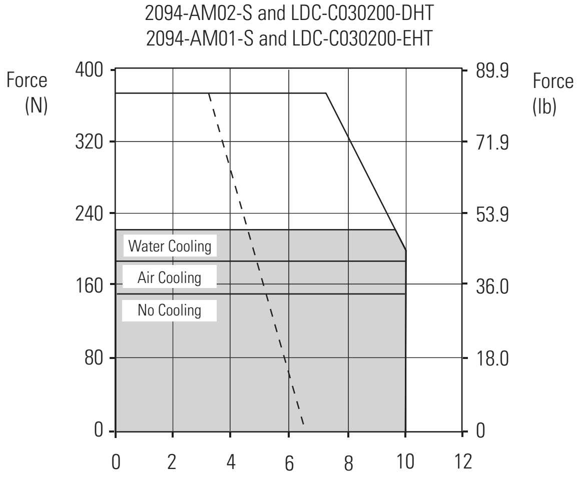
Linear Motor	**Maximum Speed** m/s (ft/s)	**System Continuous Stall Current** (1) Amps 0-pk	**System Continuous Stall Force** (1) N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Motor Rated Output** kW	**Kinetix 6000 230V Drives**
LDC-C075200-DHT	10.0 (32.8)	7.7…11.5	348…523 (78…117)	22.9	882 (198)	1.74…2.61	2094-AM02-S
LDC-C075200-EHT		3.8…5.7		11.5			2094-AMP5-S
LDC-C075300-DHT		11.5…17.2	523…784 (117…176)	35.6	1368 (308)	2.61…3.92	2094-AM03-S
LDC-C075300-EHT		3.8…5.7		11.9			2094-AM01-S
LDC-C075400-DHT		15.3…23.0	697…1045 (157…235)	47.4	1824 (410)	3.48…5.22	2094-AM03-S
LDC-C075400-EHT		7.7…11.5		23.7			2094-AM02-S
LDC-C100300-DHT	10.0 (32.8)	11.1…16.7	674…1012 (152…227)	34.3	1767 (397)	3.37…5.06	2094-AM03-S
LDC-C100300-EHT		3.7…5.6		11.4			2094-AM01-S
LDC-C100400-DHT		14.8…22.2	899…1349 (202…303)	45.7	2356 (530)	4.49…6.74	2094-AM03-S
LDC-C100400-EHT		7.4…11.1		22.8			2094-AM02-S
LDC-C100600-DHT		22.2…33.3	1349…2023 (303…455)	68.5	3534 (794)	6.74…10.11	2094-AM05-S
LDC-C150400-DHT	10.0 (32.8)	14.1…21.1	1281…1922 (288…432)	45.2	3498 (786)	6.40…9.61	2094-AM03-S
LDC-C150600-DHT		21.1…31.7	1922…2882 (432…648)	67.8	5246 (1179)	9.61…14.41	2094-AM05-S

(1) Values represent the range between no cooling (low value) and water cooling (high value).

Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 6000 (230V) Drives/LDC-Series Linear Motor Curves

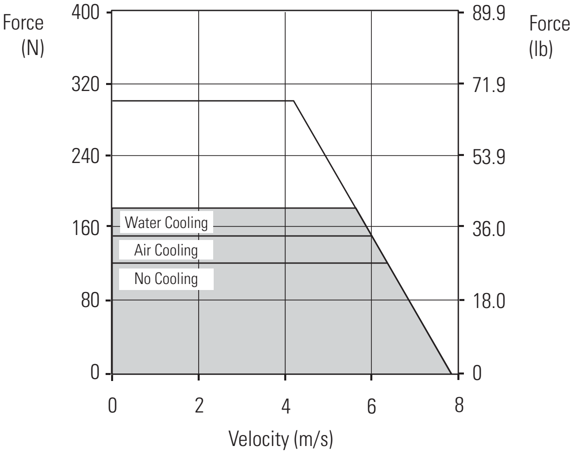




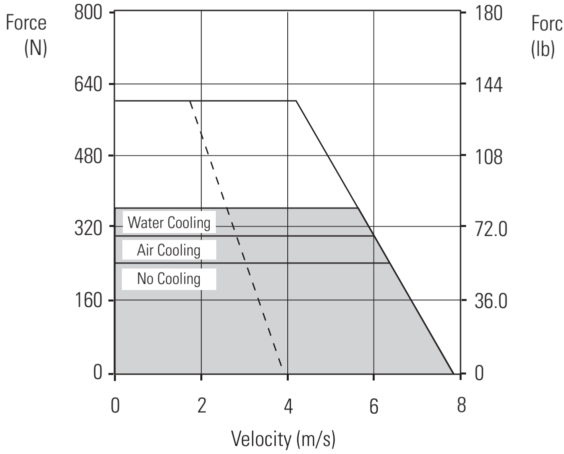
☐ = Intermittent operating region (LDC-C*xxxxxx*-DHT)
– – – = Intermittent operating region (LDC-C*xxxxxx*-EHT)
▨ = Continuous operating region

Kinetix 6000 (230V) Drives/LDC-Series Linear Motor Curves, Continued

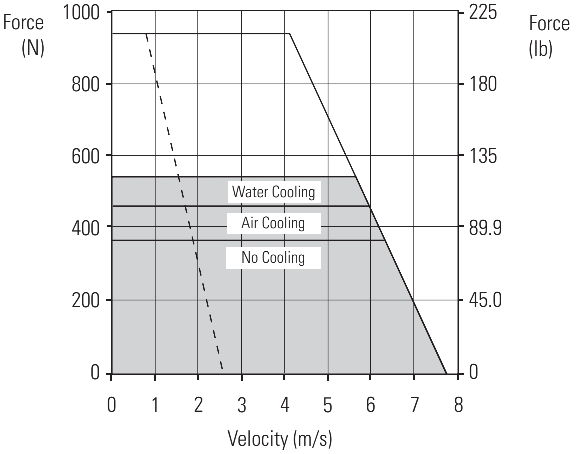
2094-AM01-S and LDC-C050100-DHT



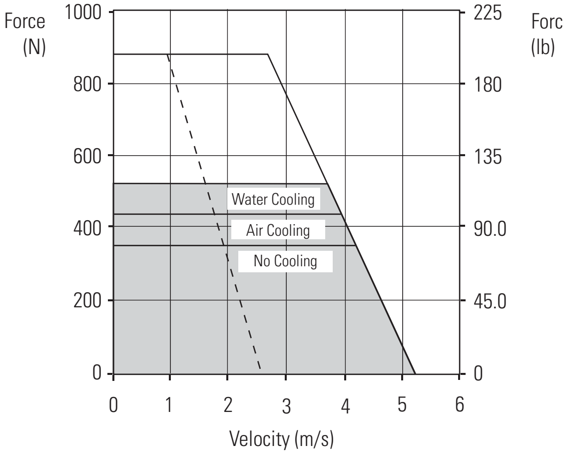
2094-AM02-S and LDC-C050200-DHT
2094-AMP5-S and LDC-C050200-EHT



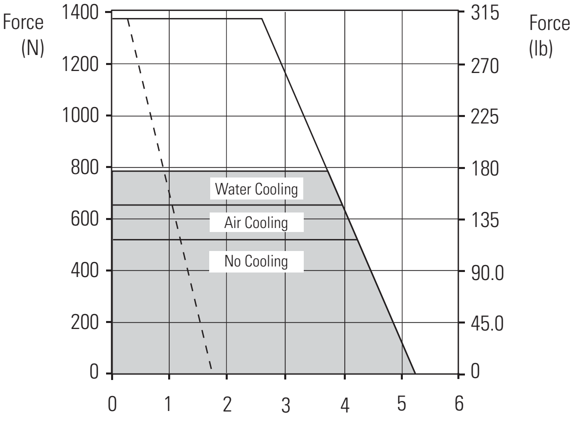
2094-AM03-S and LDC-C050300-DHT
2094-AMP5-S and LDC-C050300-EHT



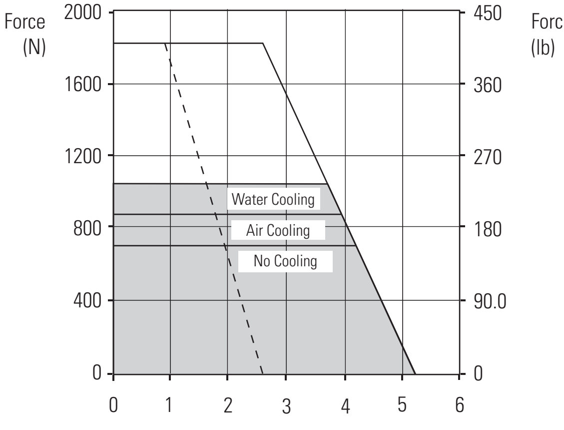
2094-AM02-S and LDC-C075200-DHT
2094-AMP5-S and LDC-C075200-EHT



2094-AM03-S and LDC-C075300-DHT
2094-AM01-S and LDC-C075300-EHT

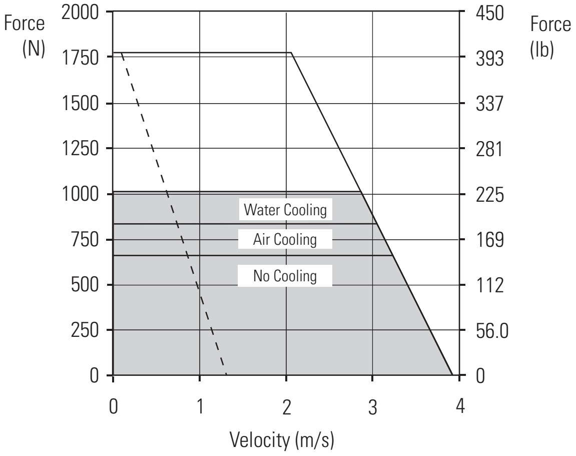


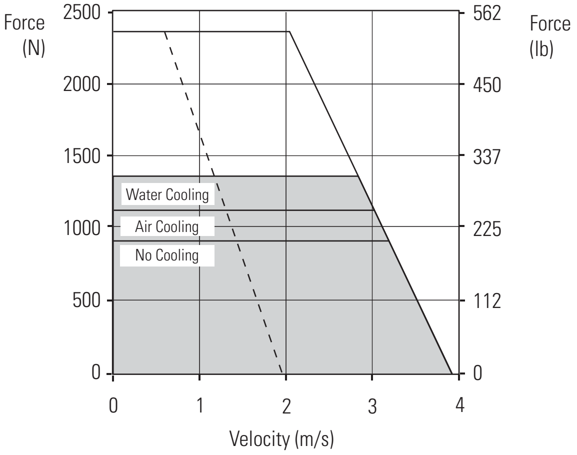
2094-AM03-S and LDC-C075400-DHT
2094-AM02-S and LDC-C075400-EHT

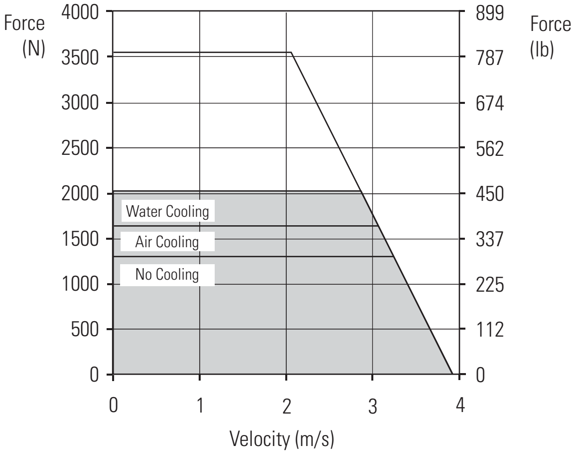


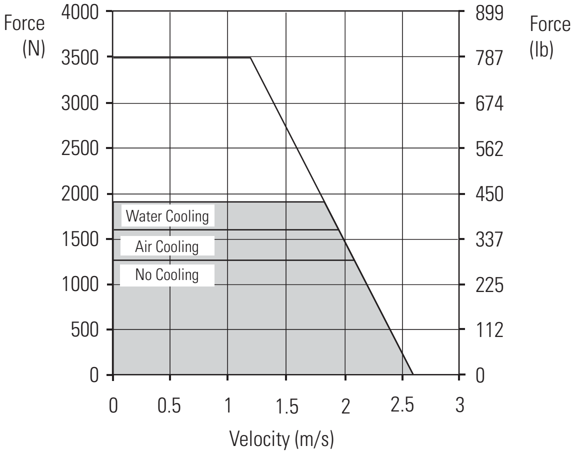
☐ = Intermittent operating region (LDC-C*xxxxxx*-DHT)
┅ = Intermittent operating region (LDC-C*xxxxxx*-EHT)
▨ = Continuous operating region

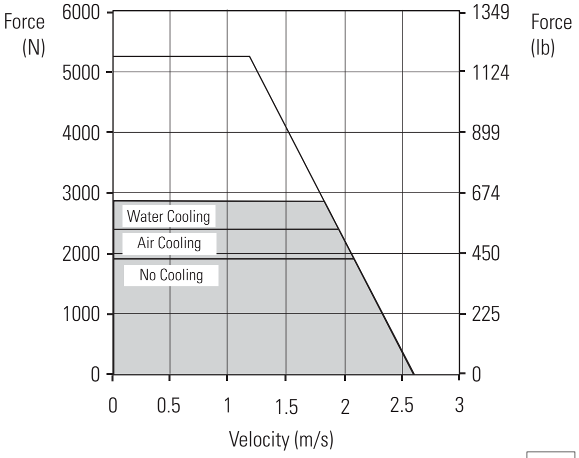
Kinetix 6000 (230V) Drives/LDC-Series Linear Motor Curves, Continued


2094-AM03-S and LDC-C100300-DHT
2094-AM01-S and LDC-C100300-EHT


2094-AM03-S and LDC-C100400-DHT
2094-AM02-S and LDC-C100400-EHT


2094-AM05-S and LDC-C100600-DHT


2094-AM03-S and LDC-C150400-DHT


2094-AM05-S and LDC-C150600-DHT

☐ = Intermittent operating region (LDC-C*xxxxxx*-DHT)
⬚ = Intermittent operating region (LDC-C*xxxxxx*-EHT)
▨ = Continuous operating region

Kinetix 6000 and Kinetix 6200/6500 (460V) Drives with LDC-Series Linear Motors

This section provides system combination information for the Kinetix 6000 and Kinetix 6200/6500 (460V) drives when matched with LDC-Series iron-core linear motors. Included are power and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Linear Motor Cable Combinations

Linear Motor	Motor Power Cable	Motor Feedback Cable (1)
LDC-C030100-DHT, LDC-C030200-DHT, LDC-C030200-EHT	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-XXNFMF-S*xx* (standard) 2090-CFBM7DF-CDAF*xx* (continuous-flex) Sin/Cos or TTL Encoder Feedback
LDC-C050100-DHT, LDC-C050200-DHT, LDC-C050200-EHT, LDC-C050300-DHT, LDC-C050300-EHT		
LDC-C075200-DHT, LDC-C075200-EHT, LDC-C075300-DHT, LDC-C075300-EHT, LDC-C075400-DHT, LDC-C075400-EHT		
LDC-C100300-DHT, LDC-C100300-EHT, LDC-C100400-DHT, LDC-C100400-EHT, LDC-C100600-DHT		
LDC-C150400-DHT, LDC-C150600-DHT		

(1) Use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

LDC-Series Performance Specifications with Kinetix 6200/6500 (460V) Drives

Linear Motor	Maximum Speed m/s (ft/s)	System Continuous Stall Current (1) Amps 0-pk	System Continuous Stall Force (1) N (lb)	System Peak Stall Current Amps 0-pk	System Peak Stall Force N (lb)	Linear Motor Rated Output kW	Kinetix 6200/ Kinetix 6500 460V Drives
LDC-C030100-DHT	10.0 (32.8)	4.1…6.1	74…111 (17…25)	12.1	188 (42)	0.37…0.55	2094-BM01-M
LDC-C030200-DHT		8.1…12.2	148…222 (33…50)	24.3	375 (84)	0.74…1.11	2094-BM02-M
LDC-C030200-EHT		4.1…6.1		12.1			2094-BM01-M
LDC-C050100-DHT	10.0 (32.8)	3.9…5.9	119…179 (27…40)	11.7	302 (68)	0.59…0.89	2094-BM01-M
LDC-C050200-DHT		7.9…11.8	240…359 (54…81)	23.3	600 (135)	1.20…1.79	2094-BM02-M
LDC-C050200-EHT		3.9…5.9		11.6			2094-BM01-M
LDC-C050300-DHT		11.8…17.7	363…544 (82…122)	35.9	941 (212)	1.81…2.72	2094-BM02-M
LDC-C050300-EHT		3.9…5.9		12.0			2094-BM01-M

(1) Values represent the range between no cooling (low value) and water cooling (high value).

Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

LDC-Series Performance Specifications with Kinetix 6200/6500 (460V) Drives, Continued

Linear Motor	**Maximum Speed** m/s (ft/s)	**System Continuous Stall Current** (1) Amps 0-pk	**System Continuous Stall Force** (1) N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Motor Rated Output** kW	**Kinetix 6200/ Kinetix 6500 460V Drives**
LDC-C075200-DHT	10.0 (32.8)	7.7…11.5	348…523 (78…117)	22.9	882 (198)	1.74…2.61	2094-BM02-M
LDC-C075200-EHT		3.8…5.7		11.5			2094-BM01-M
LDC-C075300-DHT		11.5…17.2	523…784 (117…176)	35.6	1368 (308)	2.61…3.92	2094-BM02-M
LDC-C075300-EHT		3.8…5.7		11.9			2094-BM01-M
LDC-C075400-DHT		15.3…23.0	697…1045 (157…235)	47.4	1824 (410)	3.48…5.22	2094-BM03-M
LDC-C075400-EHT		7.7…11.5		23.7			2094-BM02-M
LDC-C100300-DHT	10.0 (32.8)	11.1…16.7	674…1012 (152…227)	34.3	1767 (397)	3.37…5.06	2094-BM02-M
LDC-C100300-EHT		3.7…5.6		11.4			2094-BM01-M
LDC-C100400-DHT		14.8…22.2	899…1349 (202…303)	45.7	2356 (530)	4.49…6.74	2094-BM03-M
LDC-C100400-EHT		7.4…11.1		22.8			2094-BM02-M
LDC-C100600-DHT		22.2…33.3	1349…2023 (303…455)	68.5	3534 (794)	6.74…10.11	2094-BM03-M
LDC-C100600-EHT		11.1…16.7		34.3			2094-BM02-M
LDC-C150400-DHT	10.0 (32.8)	14.1…21.1	1281…1922 (288…432)	45.2	3498 (786)	6.40…9.61	2094-BM03-M
LDC-C150400-EHT		7.0…10.6		22.6			2094-BM02-M
LDC-C150600-DHT		21.1…31.7	1922…2882 (432…648)	67.8	5246 (1179)	9.61…14.41	2094-BM03-M
LDC-C150600-EHT		10.6…15.8		33.9			2094-BM02-M

(1) Values represent the range between no cooling (low value) and water cooling (high value).

Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 5.000 or later.

LDC-Series Performance Specifications with Kinetix 6000 (460V) Drives

Linear Motor	**Maximum Speed** m/s (ft/s)	**System Continuous Stall Current** (1) Amps 0-pk	**System Continuous Stall Force** (1) N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Motor Rated Output** kW	**Kinetix 6000 460V Drives**
LDC-C030100-DHT	10.0 (32.8)	4.1…6.1	74…111 (17…25)	12.1	188 (42)	0.37…0.55	2094-BM01-S @ 150%
LDC-C030200-DHT		8.1…12.2	148…222 (33…50)	24.3	375 (84)	0.74…1.11	2094-BM02-S @ 250%
LDC-C030200-EHT		4.1…6.1		12.1			2094-BM01-S @ 150%
LDC-C050100-DHT	10.0 (32.8)	3.9…5.9	119…179 (27…40)	11.7	302 (68)	0.59…0.89	2094-BM01-S @ 150%
LDC-C050200-DHT		7.9…11.8	240…359 (54…81)	23.3	600 (135)	1.20…1.79	2094-BM02-S @ 250%
LDC-C050200-EHT		3.9…5.9		11.6			2094-BM01-S @ 150%
LDC-C050300-DHT		11.8…17.7	363…544 (82…122)	35.9	941 (212)	1.81…2.72	2094-BM02-S @ 250%
LDC-C050300-EHT		3.9…5.9		12.0			2094-BM01-S @ 150%

(1) Values represent the range between no cooling (low value) and water cooling (high value).

Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 5.000 or later.

LDC-Series Performance Specifications with Kinetix 6000 (460V) Drives, Continued

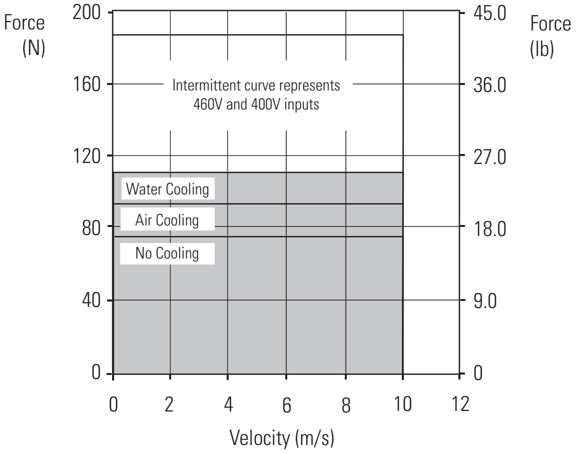
Linear Motor	**Maximum Speed** m/s (ft/s)	**System Continuous Stall Current** (1) Amps 0-pk	**System Continuous Stall Force** (1) N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Motor Rated Output** kW	**Kinetix 6000 460V Drives**
LDC-C075200-DHT	10.0 (32.8)	7.7…11.5	348…523 (78…117)	22.9	882 (198)	1.74…2.61	2094-BM02-S @ 250%
LDC-C075200-EHT		3.8…5.7		11.5			2094-BM01-S @ 150%
LDC-C075300-DHT		11.5…17.2	523…784 (117…176)	35.6	1368 (308)	2.61…3.92	2094-BM02-S @ 250%
LDC-C075300-EHT		3.8…5.7		11.9			2094-BM01-S @ 150%
LDC-C075400-DHT		15.3…23.0	697…1045 (157…235)	47.4	1824 (410)	3.48…5.22	2094-BM03-S @ 250%
LDC-C075400-EHT		7.7…11.5		23.7			2094-BM02-S @ 250%
LDC-C100300-DHT	10.0 (32.8)	11.1…16.7	674…1012 (152…227)	34.3	1767 (397)	3.37…5.06	
LDC-C100300-EHT		3.7…5.6		11.4			2094-BM01-S @ 150%
LDC-C100400-DHT		14.8…22.2	899…1349 (202…303)	45.7	2356 (530)	4.49…6.74	2094-BM03-S @ 250%
LDC-C100400-EHT		7.4…11.1		22.8			2094-BM02-S @ 250%
LDC-C100600-DHT		22.2…33.3	1349…2023 (303…455)	68.5	3534 (794)	6.74…10.11	2094-BM03-S @ 250%
LDC-C100600-EHT		11.1…16.7		34.3			2094-BM02-S @ 250%
LDC-C150400-DHT	10.0 (32.8)	14.1…21.1	1281…1922 (288…432)	45.2	3498 (786)	6.40…9.61	2094-BM03-S @ 150%
LDC-C150400-EHT		7.0…10.6		22.6			2094-BM02-S @ 250%
LDC-C150600-DHT		21.1…31.7	1922…2882 (432…648)	67.8	5246 (1179)	9.61…14.41	2094-BM03-S @ 250%
LDC-C150600-EHT		10.6…15.8		33.9			2094-BM02-S @ 250%

(1) Values represent the range between no cooling (low value) and water cooling (high value).

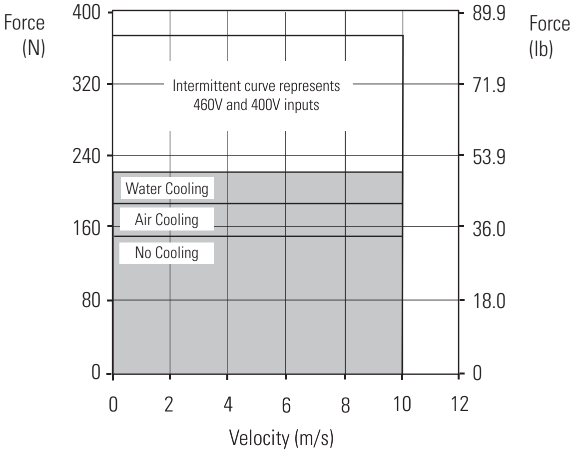
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 5.000 or later.

Kinetix 6000 and Kinetix 6200/6500 (460V) Drives/LDC-Series Linear Motor Curves

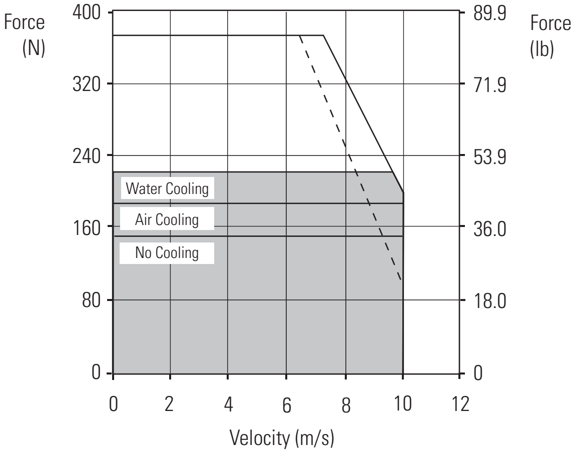
2094-BM01-S @ 150% or 2094-BM01-M
and LDC-C030100-DHT



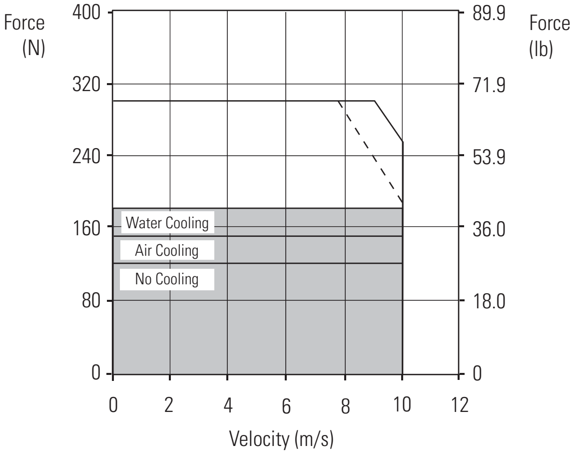
2094-BM02-S @ 250% or 2094-BM02-M
and LDC-C030200-DHT



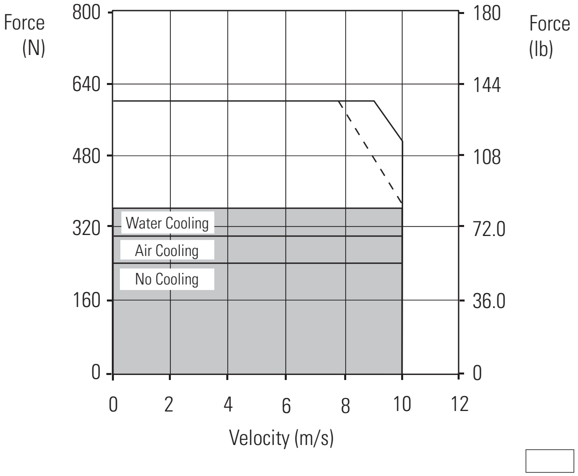
2094-BM01-S @ 150% or 2094-BM01-M
and LDC-C030200-EHT



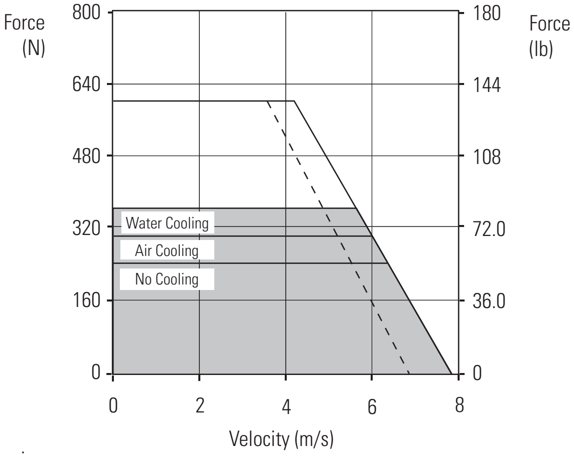
2094-BM01-S @ 150% or 2094-BM01-M
and LDC-C050100-DHT



2094-BM02-S @ 250% or 2094-BM02-M
and LDC-C050200-DHT



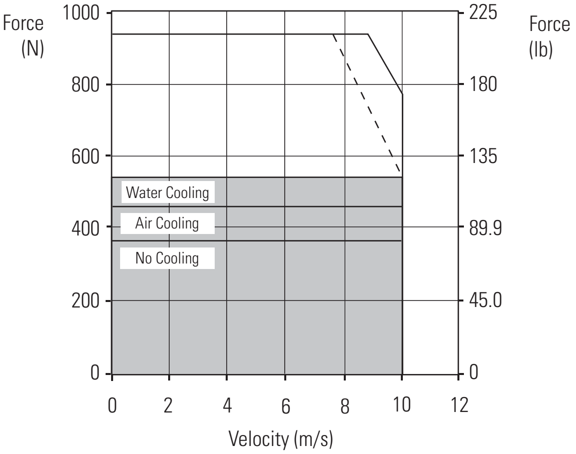
2094-BM01-S @ 150% or 2094-BM01-M
and LDC-C050200-EHT



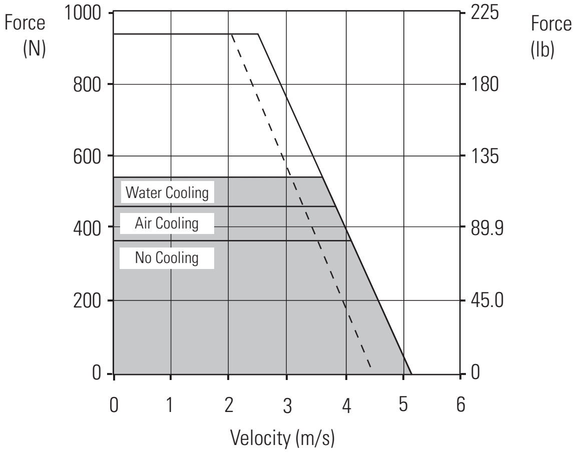
☐ = Intermittent operating region
- - - = Intermittent operating region with 400V AC (rms) input voltage
▨ = Continuous operating region

Kinetix 6000 and Kinetix 6200/6500 (460V) Drives/LDC-Series Linear Motor Curves, Continued

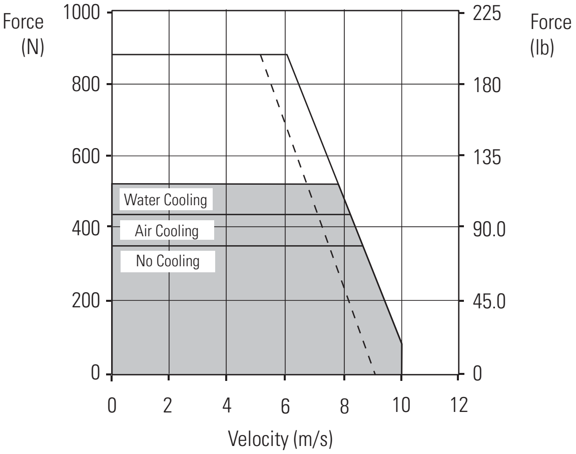
2094-BM02-S @ 250% or 2094-BM02-M
and LDC-C050300-DHT



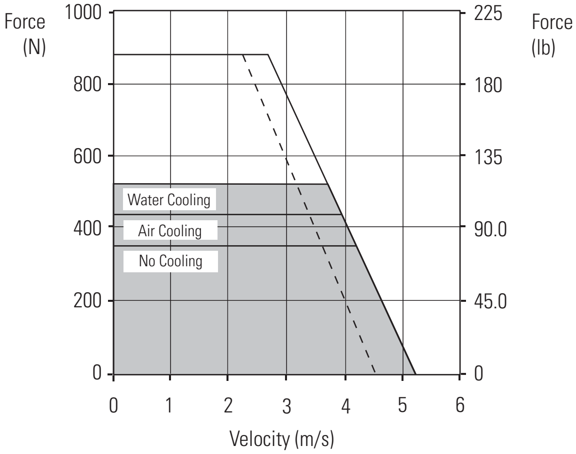
2094-BM01-S @ 150% or 2094-BM01-M
and LDC-C050300-EHT



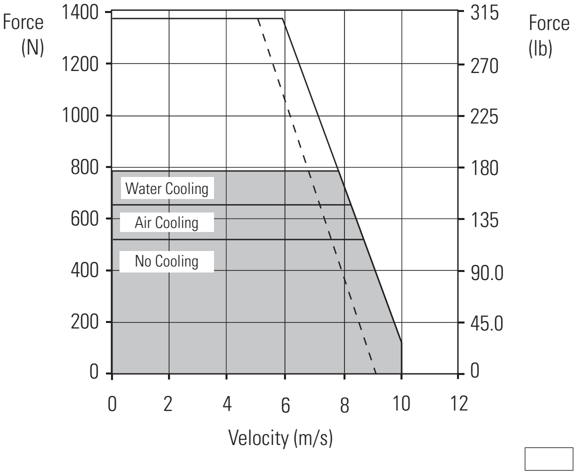
2094-BM02-S @ 250% or 2094-BM02-M
and LDC-C075200-DHT



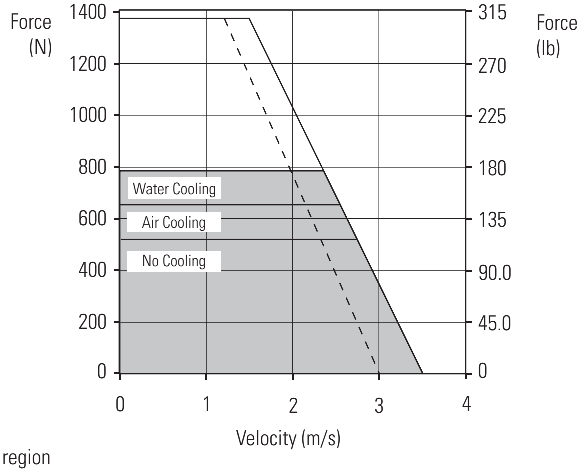
2094-BM01-S @ 150% or 2094-BM01-M
and LDC-C075200-EHT



2094-BM02-S @ 250% or 2094-BM02-M
and LDC-C075300-DHT



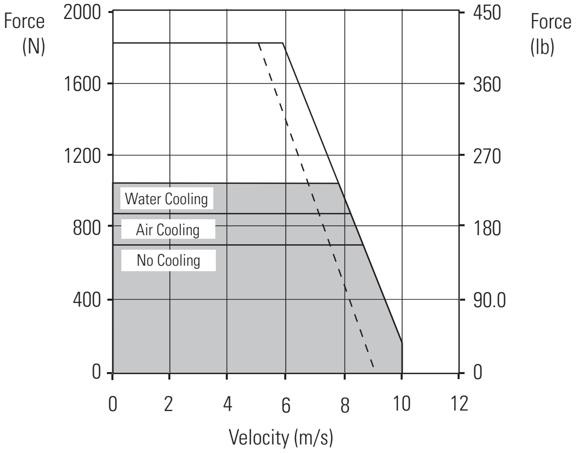
2094-BM01-S @ 150% or 2094-BM01-M
and LDC-C075300-EHT



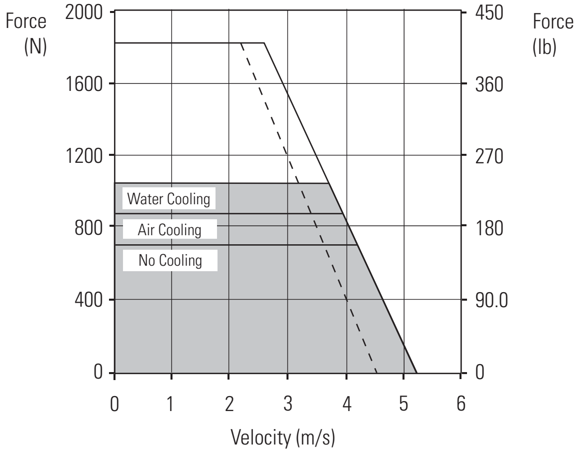
☐ = Intermittent operating region
- - - = Intermittent operating region with 400V AC (rms) input voltage
▨ = Continuous operating region

Kinetix 6000 and Kinetix 6200/6500 (460V) Drives/LDC-Series Linear Motor Curves, Continued

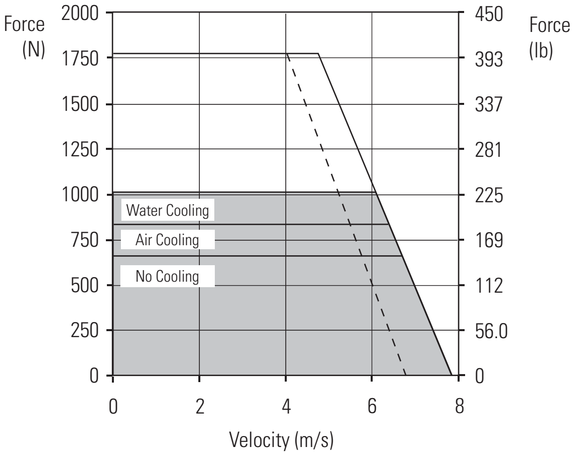
2094-BM03-S @ 250% or 2094-BM03-M
and LDC-C075400-DHT



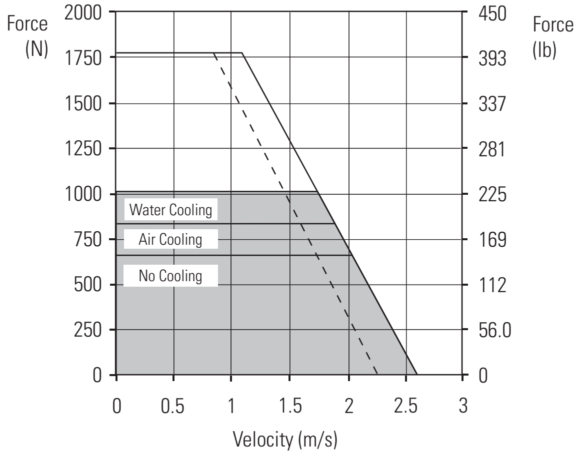
2094-BM02-S @ 250% or 2094-BM02-M
and LDC-C075400-EHT



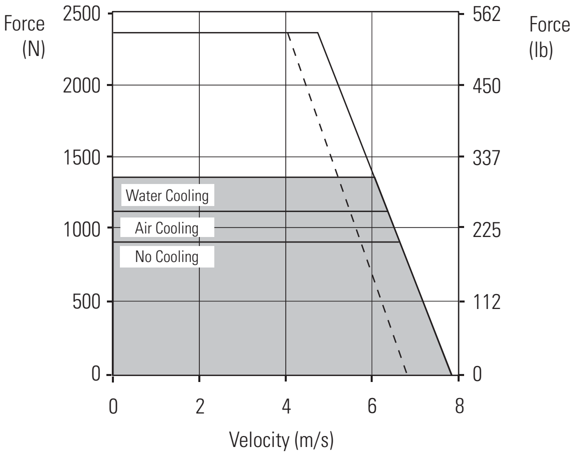
2094-BM02-S @ 250% or 2094-BM02-M
and LDC-C100300-DHT



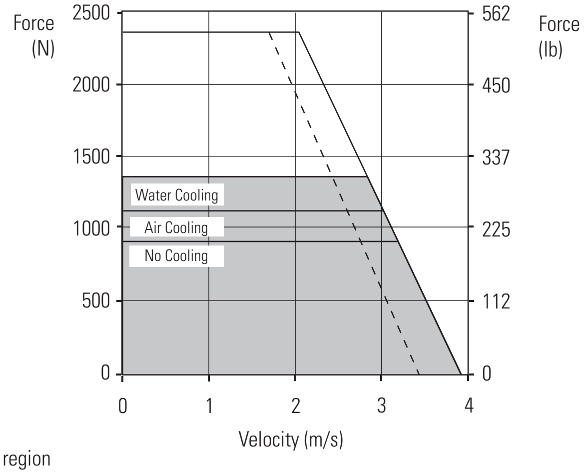
2094-BM01-S @ 150% or 2094-BM01-M
and LDC-C100300-EHT



2094-BM03-S @ 250% or 2094-BM03-M
and LDC-C100400-DHT

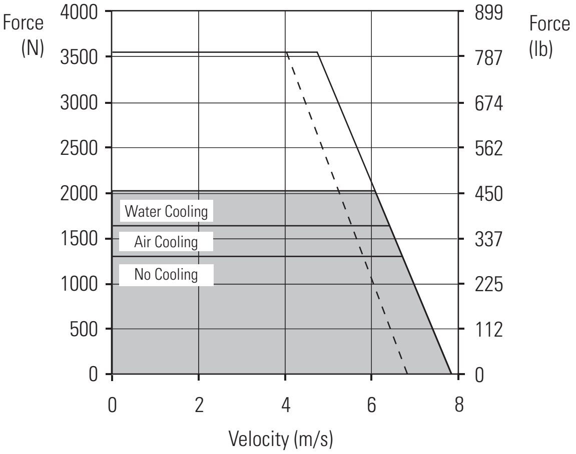


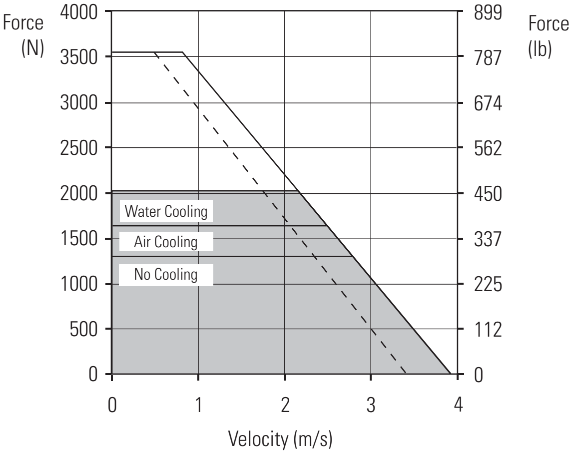
2094-BM02-S @ 250% or 2094-BM02-M
and LDC-C100400-EHT

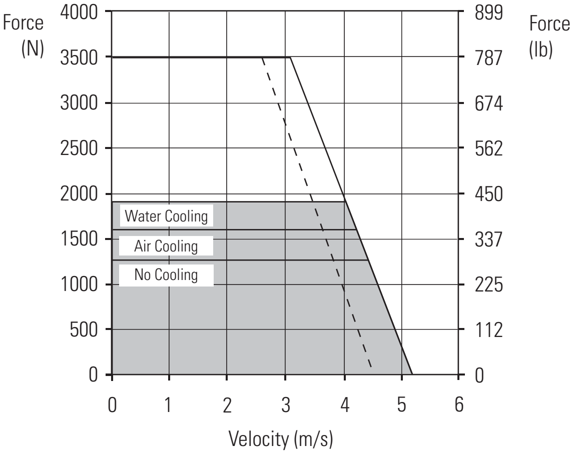


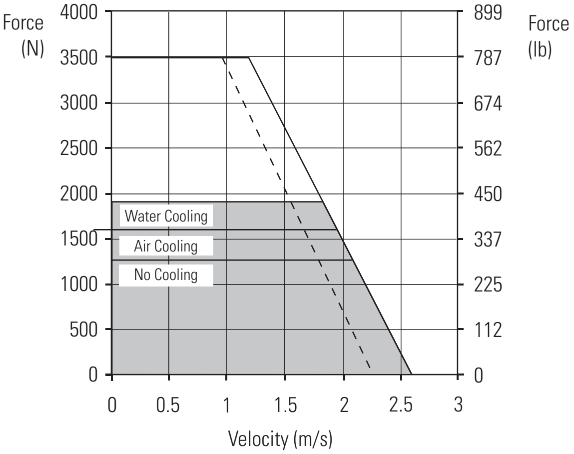
☐ = Intermittent operating region
- - - = Intermittent operating region with 400V AC (rms) input voltage
▨ = Continuous operating region

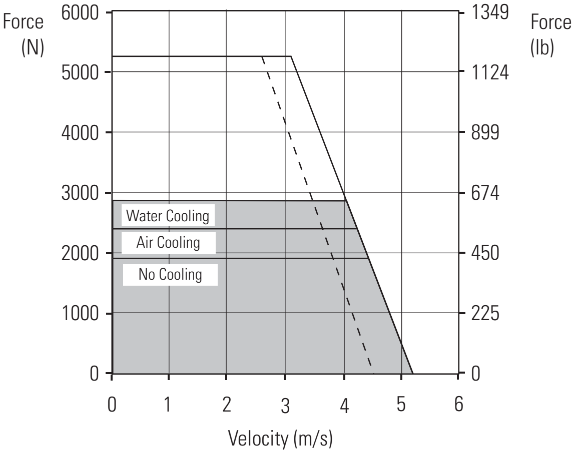
Kinetix 6000 and Kinetix 6200/6500 (460V) Drives/LDC-Series Linear Motor Curves, Continued

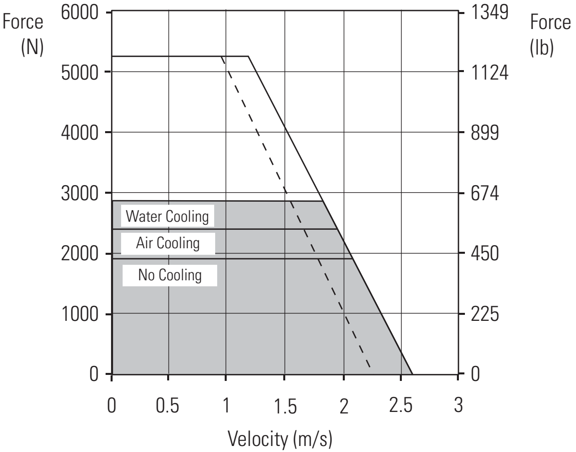

2094-BM03-S @ 250% or 2094-BM03-M
and LDC-C100600-DHT


2094-BM02-S @ 250% or 2094-BM02-M
and LDC-C100600-EHT


2094-BM03-S @ 150% or 2094-BM03-M
and LDC-C150400-DHT


2094-BM02-S @ 250% or 2094-BM02-M
and LDC-C150400-EHT


2094-BM03-S @ 250% or 2094-BM03-M
and LDC-C150600-DHT


2094-BM02-S @ 250% or 2094-BM02-M
and LDC-C150600-DHT

☐ = Intermittent operating region
- - - = Intermittent operating region with 400V AC (rms) input voltage
▨ = Continuous operating region

Kinetix 6000 (230V) Drives with LDL-Series Linear Motors

This section provides system combination information for the Kinetix 6000 (230V) drives when matched with LDL-Series ironless linear motors. Included are power and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Linear Motor Cable Combinations

Linear Motors	Motor Power Cable	Motor Feedback Cable [1]
LDL-N030120-DHT, LDL-N030240-DHT, LDL-N030240-EHT	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-XXNFMF-S*xx* (standard) 2090-CFBM7DF-CDAF*xx* (continuous-flex) Sin/Cos or TTL Encoder Feedback
LDL-N050120-DHT, LDL-N050240-DHT, LDL-N050240-EHT, LDL-N050360-DHT, LDL-N050360-EHT, LDL-N050480-DHT, LDL-N050480-EHT		
LDL-N075480-DHT, LDL-N075480-EHT		
LDL-T030120-DHT, LDL-T030240-DHT, LDL-T030240-EHT		
LDL-T050120-DHT, LDL-T050240-DHT, LDL-T050240-EHT, LDL-T050360-DHT, LDL-T050480-DHT, LDL-T050480-EHT		
LDL-T075480-DHT, LDL-T075480-EHT		

(1) Use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

LDL-Series Performance Specifications with Kinetix 6000 (230V) Drives

Linear Motor	Maximum Speed m/s (ft/s)	System Continuous Stall Current Amps 0-pk	System Continuous Stall Force N (lb)	System Peak Stall Current Amps 0-pk	System Peak Stall Force N (lb)	Linear Motor Rated Output kW	Kinetix 6000 230V Drives
LDL-N030120-DHT	10.0 (32.8)	3.0	63 (14)	9.9	209 (47)	0.31	2094-AMP5-S
LDL-N030240-DHT		6.0	126 (28)	19.9	417 (94)	0.63	2094-AM01-S
LDL-N030240-EHT		3.0		9.9			2094-AMP5-S
LDL-T030120-DHT		3.0	72 (16)	9.9	239 (54)	0.36	2094-AMP5-S
LDL-T030240-DHT		6.0	144 (32)	19.9	479 (108)	0.72	2094-AM01-S
LDL-T030240-EHT		3.0		9.9			2094-AMP5-S

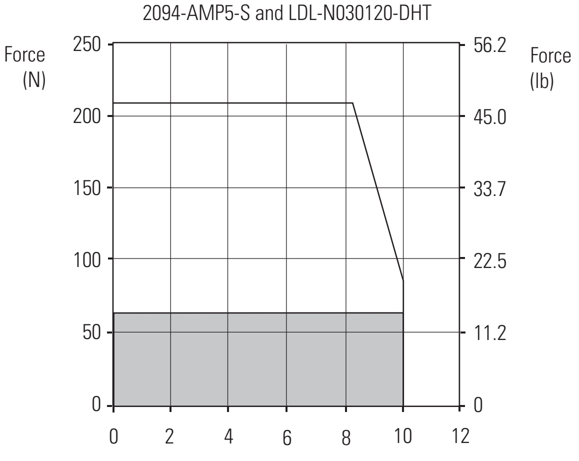
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

LDL-Series Performance Specifications with Kinetix 6000 (230V) Drives, Continued

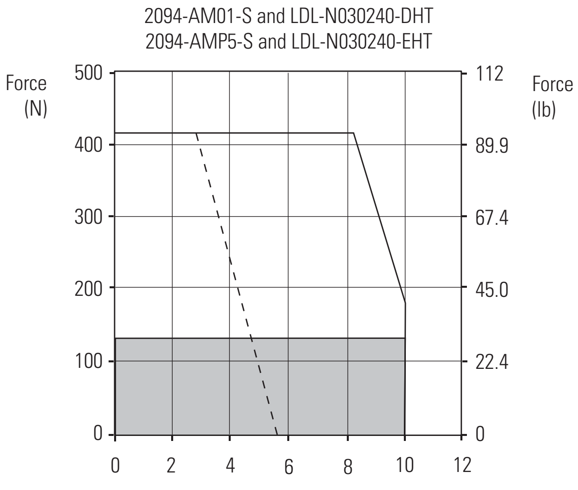
Linear Motor	**Maximum Speed** m/s (ft/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Motor Rated Output** kW	**Kinetix 6000 230V Drives**
LDL-N050120-DHT	10.0 (32.8)	2.7	96 (22)	9.1	317 (71)	0.48	2094-AMP5-S
LDL-N050240-DHT		5.5	191 (43)	18.1	635 (143)	0.95	2094-AM01-S
LDL-N050240-EHT		2.7		9.1			2094-AMP5-S
LDL-N050360-DHT		8.2	287 (65)	27.2	952 (214)	1.43	2094-AM02-S
LDL-N050360-EHT		2.7		9.1			2094-AMP5-S
LDL-N050480-DHT		10.9	383 (86)	36.3	1269 (285)	1.91	2094-AM03-S
LDL-N050480-EHT		5.5		18.1			2094-AM01-S
LDL-T050120-DHT		2.7	110 (25)	9.1	364 (82)	0.55	2094-AMP5-S
LDL-T050240-DHT		5.5	220 (49)	18.1	728 (164)	1.10	2094-AM01-S
LDL-T050240-EHT		2.7		9.1			2094-AMP5-S
LDL-T050360-DHT		8.2	329 (74)	27.2	1093 (246)	1.64	2094-AM02-S
LDL-T050480-DHT		10.9	439 (99)	36.3	1457 (327)	2.19	2094-AM03-S
LDL-T050480-EHT		5.5		18.1			2094-AM01-S
LDL-N075480-DHT	10.0 (32.8)	9.9	519 (117)	32.8	1723 (387)	2.59	2094-AM03-S
LDL-N075480-EHT		4.9		16.4			2094-AM01-S
LDL-T075480-DHT		9.9	596 (134)	32.8	1977 (444)	2.98	2094-AM03-S
LDL-T075480-EHT		4.9		16.4			2094-AM01-S

Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 6000 (230V) Drives/LDL-Series Linear Motor Curves



2094-AMP5-S and LDL-N030120-DHT

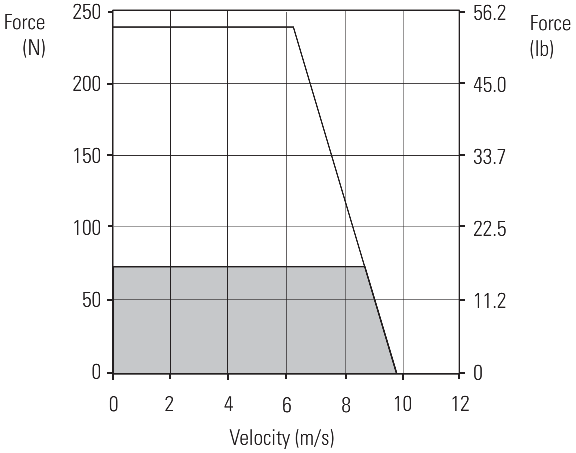


2094-AM01-S and LDL-N030240-DHT
2094-AMP5-S and LDL-N030240-EHT

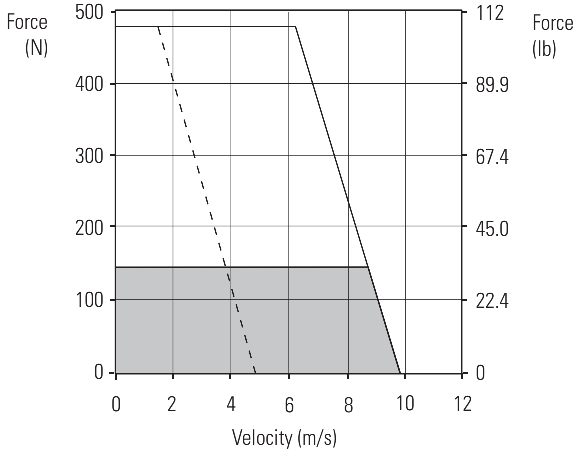
☐ = Intermittent operating region (LDL-*xxxxxxx*-DHT)
– – – = Intermittent operating region (LDL-*xxxxxxx*-EHT)
▨ = Continuous operating region

Kinetix 6000 (230V) Drives/LDL-Series Linear Motor Curves, Continued

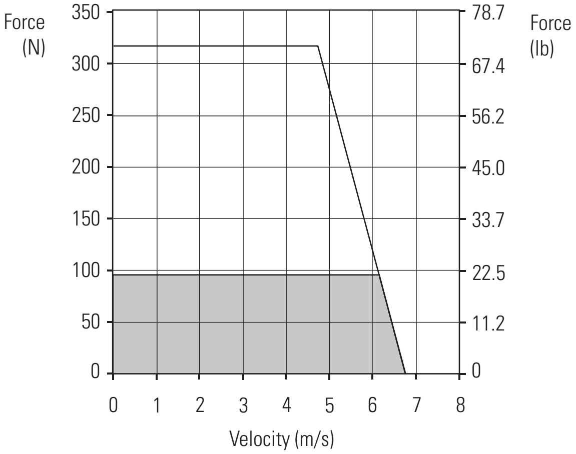
2094-AMP5-S and LDL-T030120-DHT



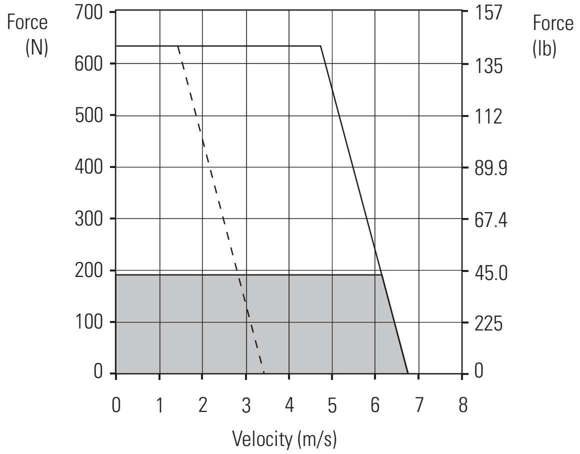
2094-AM01-S and LDL-T030240-DHT
2094-AMP5-S and LDL-T030240-EHT



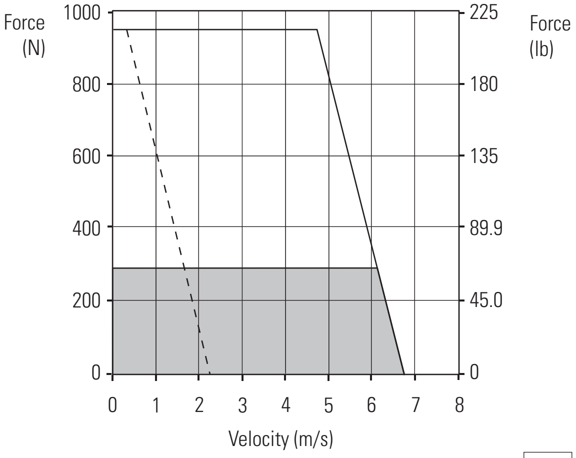
2094-AMP5-S and LDL-N050120-DHT



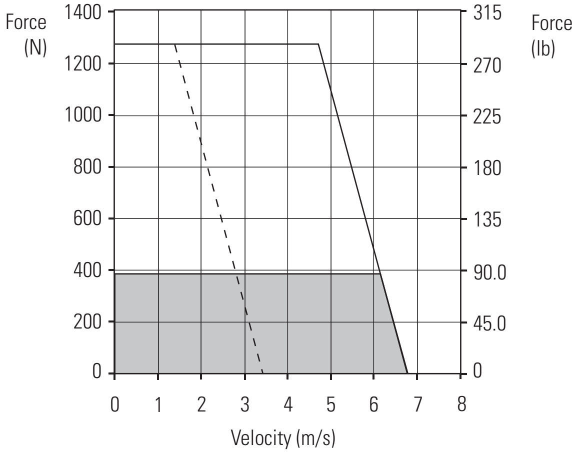
2094-AM01-S and LDL-N050240-DHT,
2094-AMP5-S and LDL-N050240-EHT



2094-AM02-S and LDL-N050360-DHT
2094-AMP5-S and LDL-N050360-EHT



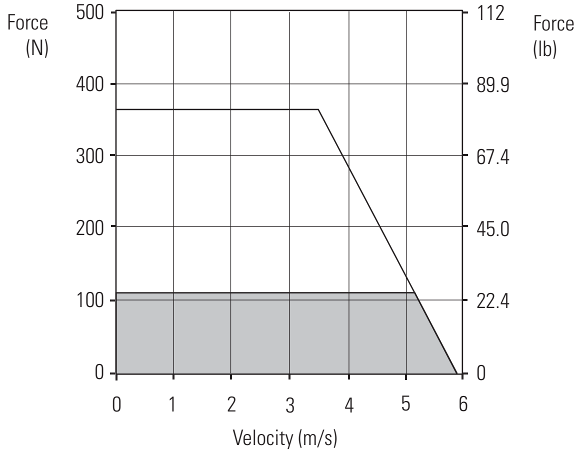
2094-AM03-S and LDL-N050480-DHT
2094-AM01-S and LDL-N050480-EHT



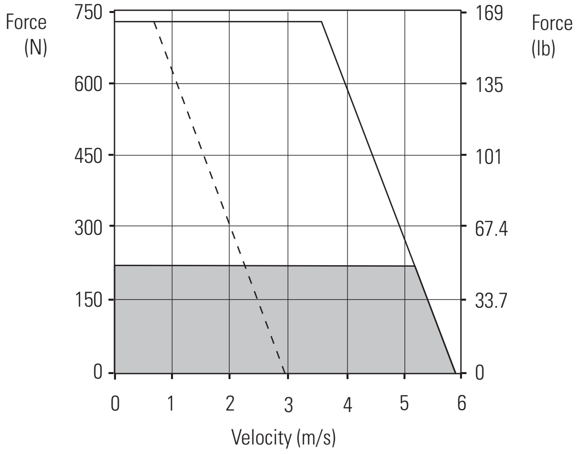
☐ = Intermittent operating region (LDL-*xxxxxxx*-DHT)
- - - = Intermittent operating region (LDL-*xxxxxxx*-EHT)
▨ = Continuous operating region

Kinetix 6000 (230V) Drives/LDL-Series Linear Motor Curves, Continued

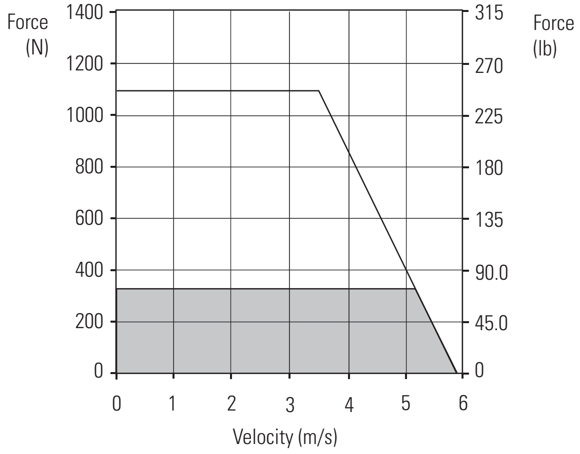
2094-AMP5-S and LDL-T050120-DHT



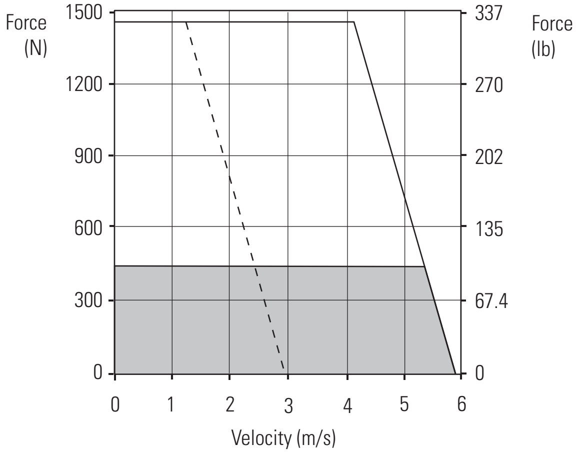
2094-AM01-S and LDL-T050240-DHT
2094-AMP5-S and LDL-T050240-EHT



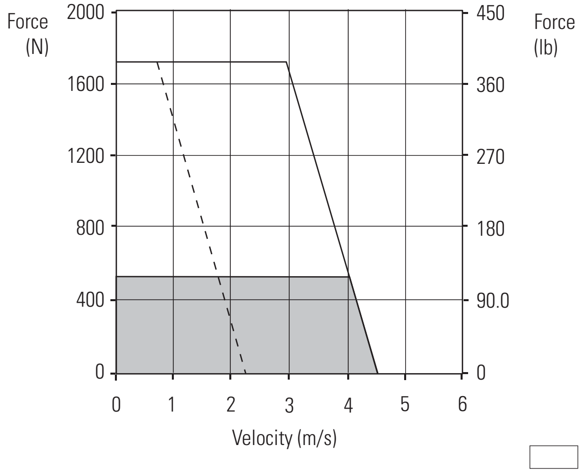
2094-AM02-S and LDL-T050360-DHT



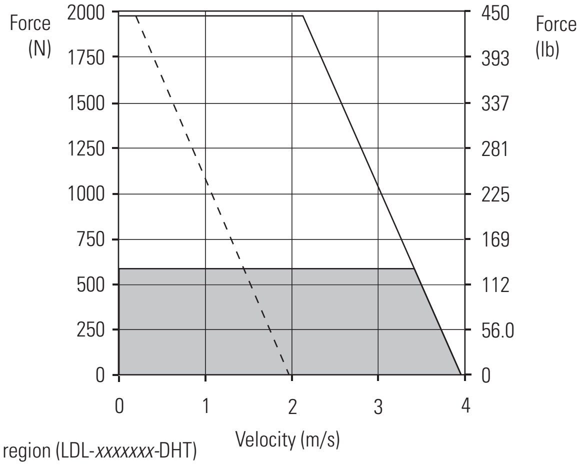
2094-AM03-S and LDL-T050480-DHT
2094-AM01-S and LDL-T050480-EHT



2094-AM03-S and LDL-N075480-DHT
2094-AM01-S and LDL-N075480-EHT



2094-AM03-S and LDL-T075480-DHT
2094-AM01-S and LDL-T075480-EHT



☐ = Intermittent operating region (LDL-*xxxxxxx*-DHT)
- - - = Intermittent operating region (LDL-*xxxxxxx*-EHT)
▨ = Continuous operating region

Kinetix 2000 (230V) Drives with MP-Series Integrated Linear Stages

This section provides system combination information for the Kinetix 2000 drives when matched with MP-Series (230V) integrated direct-drive or ballscrew linear stages. Included are power/brake and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Linear Stage Cable Combinations

Linear Stage	Motor Power/Brake Cable	Motor Feedback Cable (1)
MPAS-A*xxxx*1-V05SxA	2090-XXNPMF-16S*xx* (2)	2090-XXNFMF-S*xx* (3) Absolute High-resolution Feedback
MPAS-A*xxxx*2-V20SxA		
MPAS-A6*xxx*B-ALM*x*2C, MPAS-A8*xxx*E-ALM*x*2C, MPAS-A9*xxx*K-ALM*x*2C		2090-XXNFMF-S*xx* (3) Incremental Feedback

(1) Use low-profile connector kit (catalog number 2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

(2) These cables are available as standard (catalog number 2090-XXNPMF-16S*xx*) or continuous-flex (catalog number 2090-CP*x*M4DF-16AF*xx*).

(3) These cables are available as standard (catalog number 2090-XXNFMF-S*xx*) or continuous-flex (catalog number 2090-CFBM4DF-CDAF*xx*).

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Linear Stage Performance Specifications with Kinetix 2000 (230V) Drives

Linear Stage	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Stage Rated Output** kW	**Kinetix 2000 230V Drives**
MPAS-A*xxxx*1-V05SxA	200 (7.9) (1)	2.83	477 (100)	6.10	1212 (272)	0.37	2093-AMP2
		3.09	521 (117)				2093-AMP5
MPAS-A*xxxx*2-V20SxA	1124 (44.3) (2)	4.24	432 (97.1)	9.10	968 (218)	0.62	2093-AMP5
		4.54	462 (104)				2093-AM01
MPAS-A6*xxx*B-ALMO2C	5000 (200)	4.2	79.6 (17.9)	12.7	284 (63.8)	0.32	2093-AMP5
		5.3	105 (23.6)	15.8	359 (80.7)		2093-AM01
MPAS-A6*xxx*B-ALMS2C		4.2	71.8 (16.1)	12.7	275 (61.8)	0.29	2093-AMP5
		4.7	83.0 (18.6)	14.2	312 (70.1)		2093-AM01
MPAS-A8*xxx*E-ALMO2C		4.2	106 (23.8)	12.7	306 (68.8)	0.53	2093-AMP5
		7.0	189 (42.5)	18.5	456 (103)		2093-AM01
MPAS-A8*xxx*E-ALMS2C		4.2	96.8 (21.8)	12.7	297 (66.8)	0.48	2093-AMP5
		6.3	159 (35.7)	16.7	399 (89.7)		2093-AM01
MPAS-A9*xxx*K-ALMO2C		4.2	172 (38.7)	12.7	465 (105)	0.77	2093-AMP5
		6.7	285 (64.1)	18.3	680 (153)		2093-AM01
MPAS-A9*xxx*K-ALMS2C		4.2	161 (36.2)	12.7	456 (103)	0.69	2093-AMP5
		6.1	245 (55.1)	16.5	601 (135)		2093-AM01

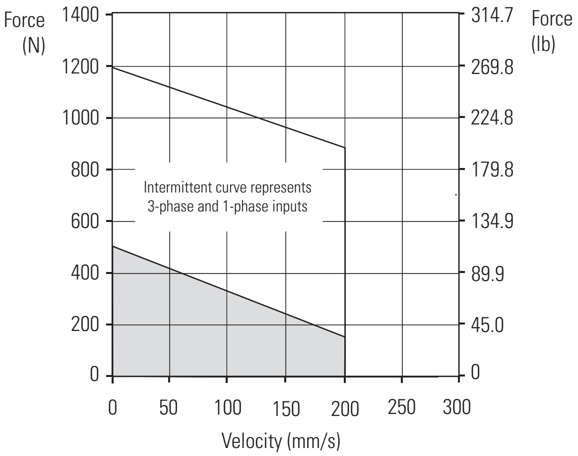
(1) For 900 mm stroke length, maximum speed is 176 mm/s (6.9 in/s). For 1020 mm stroke length, maximum speed is 143 mm/s (5.6 in/s).

(2) For 780 mm stroke length, maximum speed is 889 mm/s (35.0 in/s). For 900 mm stroke length, maximum speed is 715 mm/s (28.2 in/s). For 1020 mm stroke length, maximum speed is 582 mm/s (22.9 in/s).

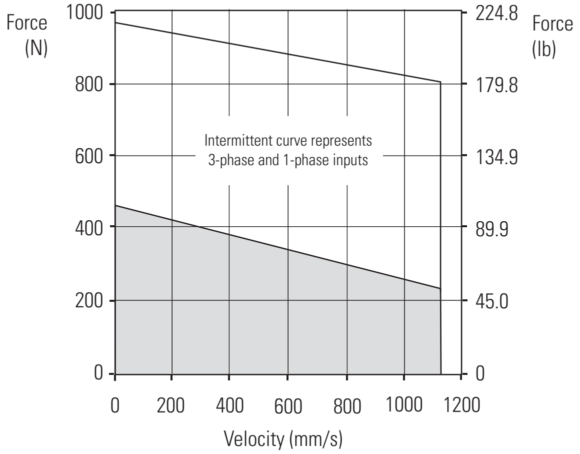
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 2000 (230V) Drives/MP-Series Integrated Linear Stage Curves

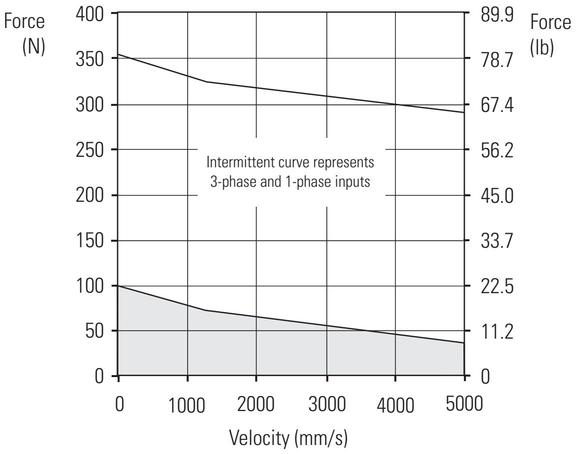
2093-AMP5 and MPAS-A6*xxx*1-V05S*x*A



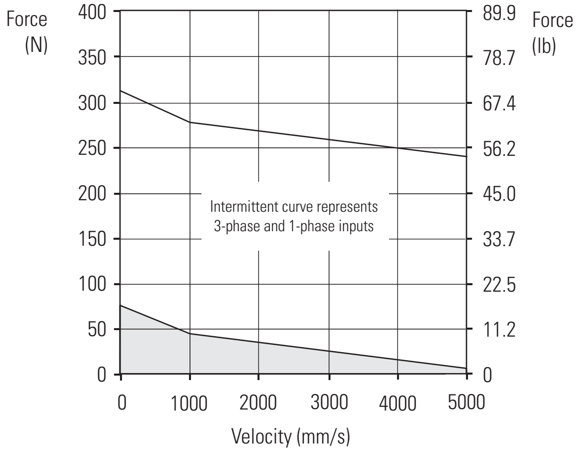
2093-AM01 and MPAS-A6*xxx*2-V20S*x*A



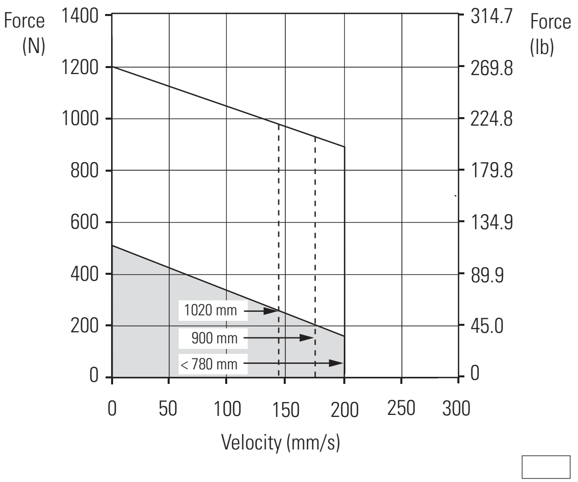
2093-AM01 and MPAS-A6*xxx*B-ALMO2C



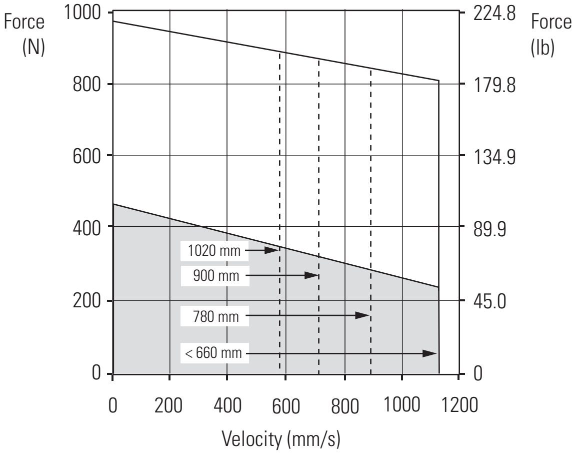
2093-AM01 and MPAS-A6*xxx*B-ALMS2C



2093-AMP5 and MPAS-A8*xxx*1-V05S*x*A



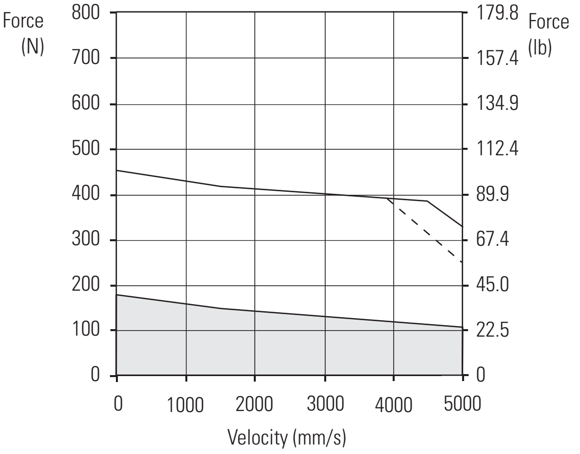
2093-AM01 and MPAS-A8*xxx*2-V20S*x*A



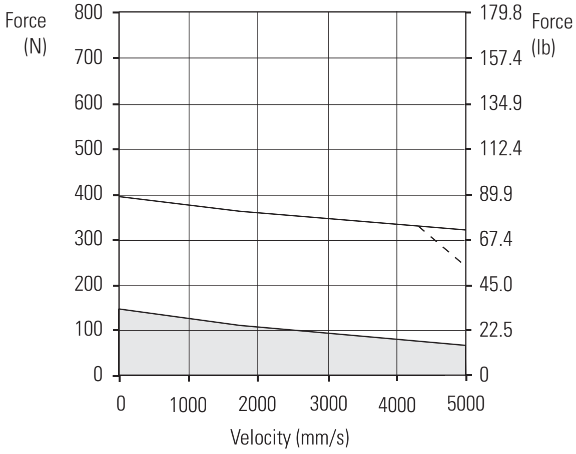
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = System operation for specified stroke length

Kinetix 2000 (230V) Drives/MP-Series Integrated Linear Stage Curves, Continued

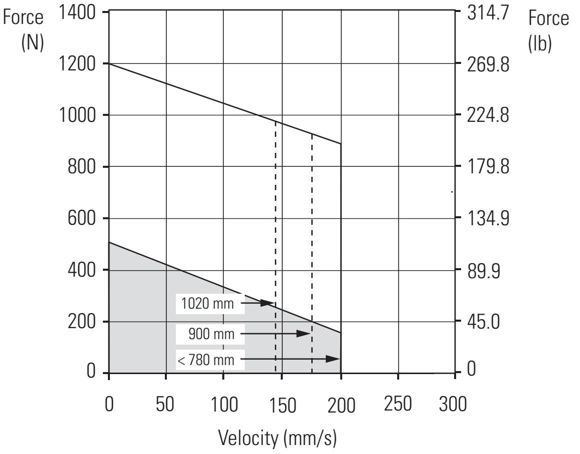
2093-AM01 and MPAS-A8*xxx*E-ALMO2C



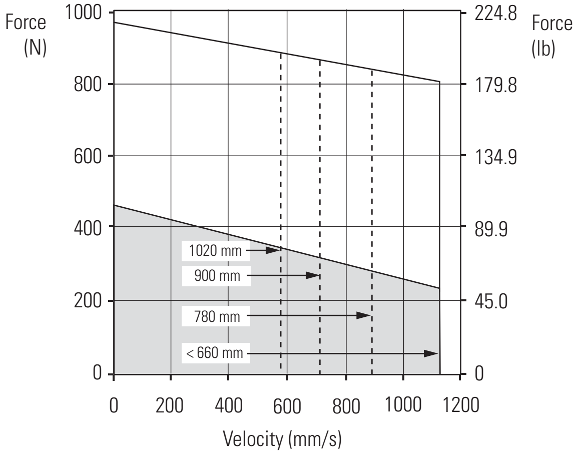
2093-AM01 and MPAS-A8*xxx*E-ALMS2C



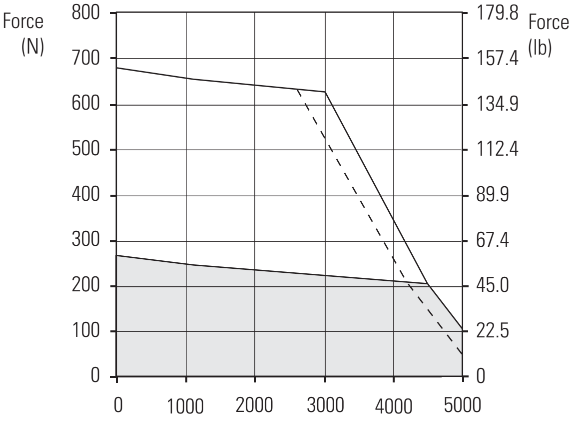
2093-AMP5 and MPAS-A9*xxx*1-V05S*x*A



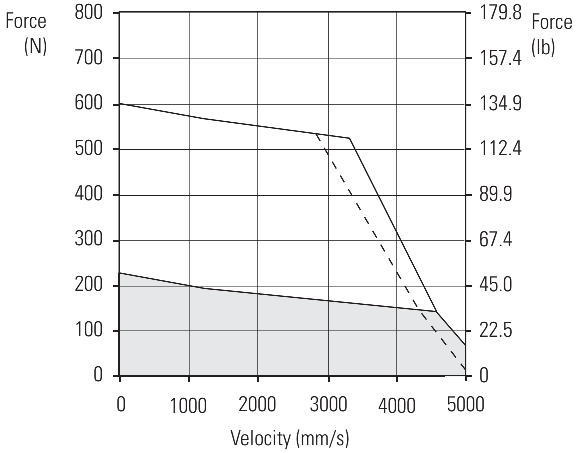
2093-AM01 and MPAS-A9*xxx*2-V20S*x*A



2093-AM01 and MPAS-A9*xxx*K-ALMO2C



2093-AM01 and MPAS-A9*xxx*K-ALMS2C



☐ = Intermittent operating region
▨ = Continuous operating region
– – = System operation (single-phase input)
- - - - = System operation for specified stroke length

Kinetix 2000 (230V) Drives with MP-Series Electric Cylinders

This section provides system combination information for the Kinetix 2000 drives when matched with MP-Series (230V) electric cylinders. Included are power/brake and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Electric Cylinder Cable Combinations

Electric Cylinder	Motor Power/Brake Cable	Motor Feedback Cable (1)
MPAR-A1*xxx*B MPAR-A1*xxx*E	2090-XXNPMF-16S*xx* (2)	2090-XXNFMF-S*xx* (3) Absolute High-resolution Feedback
MPAR-A2*xxx*C MPAR-A2*xxx*F		
MPAR-A3*xxx*E MPAR-A3*xxx*H	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback

(1) Use low-profile connector kit (catalog number 2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

(2) These cables are available as standard (catalog number 2090-XXNPMF-16S*xx*) or continuous-flex (catalog number 2090-CP*x*M4DF-16AF*xx*).

(3) These cables are available as standard (catalog number 2090-XXNFMF-S*xx*) or continuous-flex (catalog number 2090-CFBM4DF-CDAF*xx*).

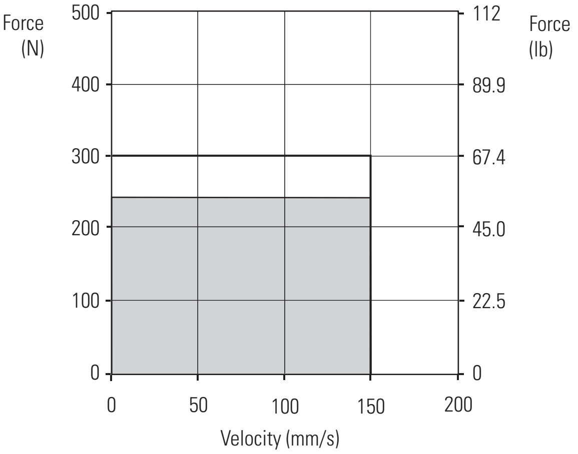
Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

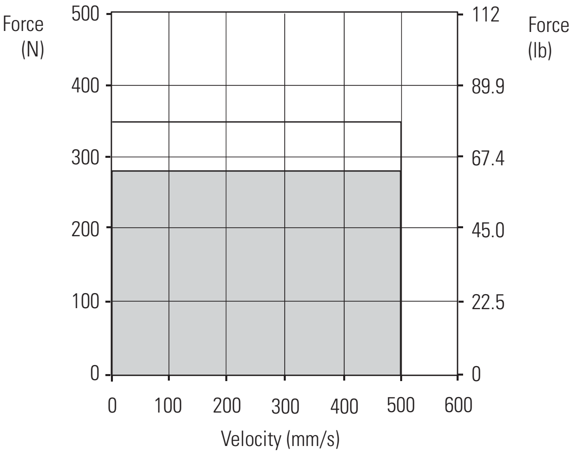
Electric Cylinder Performance Specifications with Kinetix 2000 Drives

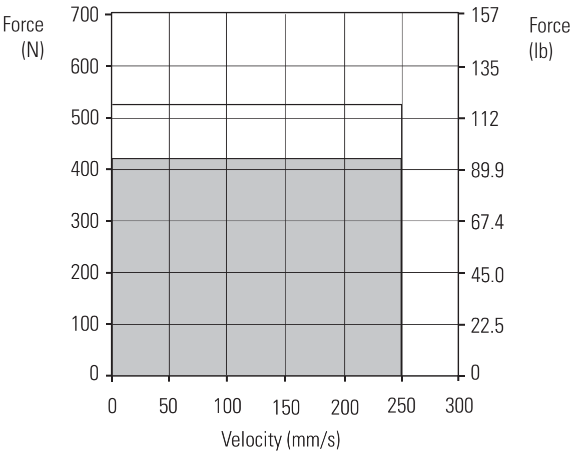
Electric Cylinder	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Rated Output** kW	**Kinetix 2000 Drives**
MPAR-A1*xxx*B	150	1.15	240 (53.9)	1.34	300 (67.4)	0.036	2093-AMP2
MPAR-A1*xxx*E	500	2.15	280 (62.9)	2.48	350 (78.7)	0.140	2093-AMP5
MPAR-A2*xxx*C	250	2.41	420 (94.4)	2.71	525 (118)	0.105	
MPAR-A2*xxx*F	640	4.54	640 (144)	5.41	800 (180)	0.409	2093-AM01
MPAR-A3*xxx*E	500	10.33	2000 (450)	12.34	2500 (562)	1.000	2093-AM02
MPAR-A3*xxx*H	1000	12.2	1300 (292)	16.4	1625 (365)	1.300	

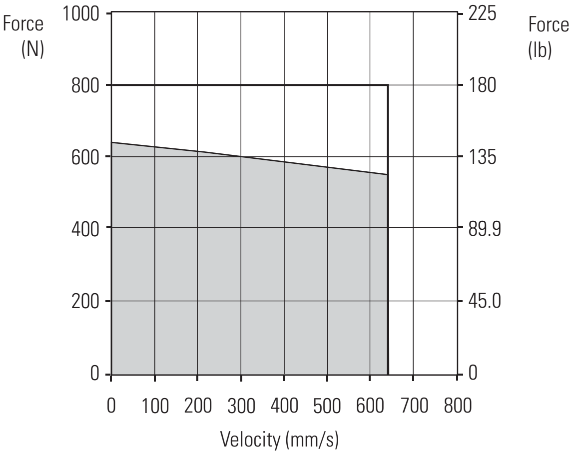
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

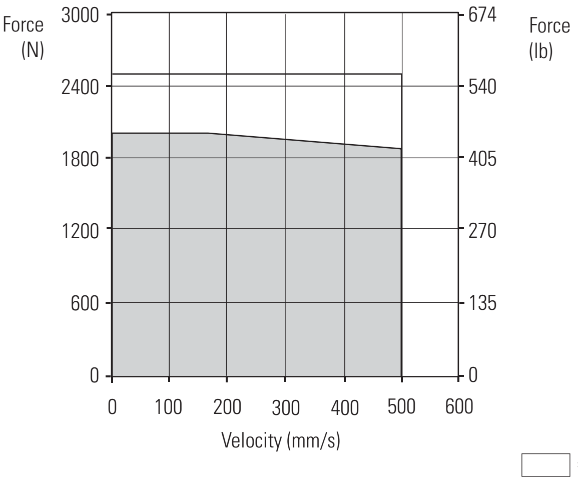
Kinetix 2000 Drives/MP-Series Electric Cylinder Curves

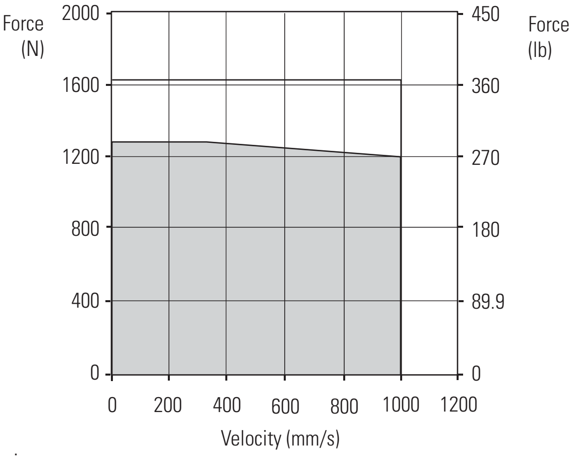

2093-AMP2 and MPAR-A1*xxx*B-V*x*A


2093-AMP5 and MPAR-A1*xxx*E-V*x*A


2093-AMP5 and MPAR-A2*xxx*C-V*x*A


2093-AM01 and MPAR-A2*xxx*F-V*x*A


2093-AM02 and MPAR-A3*xxx*E-M*x*A


2093-AM02 and MPAR-A3*xxx*H-M*x*A

☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 2000 (230V) Drives with MP-Series Heavy Duty Electric Cylinders

This section provides system combination information for the Kinetix 2000 drives when matched with MP-Series (230V) heavy-duty electric cylinders. Included are power/brake and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Electric Cylinder Cable Combinations

Electric Cylinder	Motor Power/Brake Cable	Motor Feedback Cable [1]
MPAI-A3*xxx*C MPAI-A3*xxx*E MPAI-A3*xxx*R MPAI-A3*xxx*S	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback
MPAI-A4*xxx*C MPAI-A4*xxx*E MPAI-A4*xxx*R MPAI-A4*xxx*S		

(1) Use low-profile connector kit (catalog number 2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Electric Cylinder Performance Specifications with Kinetix 2000 Drives

Performance Specifications with Ball Screw Electric Cylinders

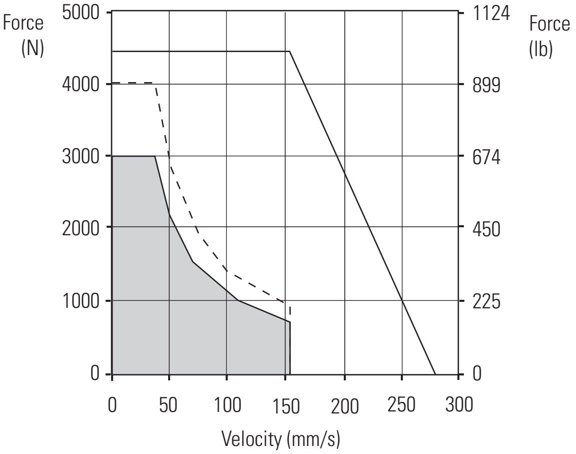
Electric Cylinder	Maximum Speed mm/s (in/s)	System Continuous Stall Current Amps 0-pk	System Continuous Stall Force N (lb)		System Peak Stall Current Amps 0-pk	System Peak Stall Force N (lb)	Rated Output kW	Kinetix 2000 Drives
			25 °C (77 °F)	40 °C (104 °F)				
MPAI-A3*xxx*C	279 (11)	5.61	4003 (900)	3176 (714)	8.40	4448 (1000)	0.39	2093-AM01
MPAI-A3*xxx*E	559 (22)		2002 (450)	1588 (357)	14.14	4003 (900)		
MPAI-A4*xxx*C	279 (11)	10.89	7784 (1750)	6179 (1389)	17.07	8896 (2000)	0.43	2093-AM02
MPAI-A4*xxx*E	559 (22)		3892 (875)	3092 (695)	27.44	7784 (1750)		

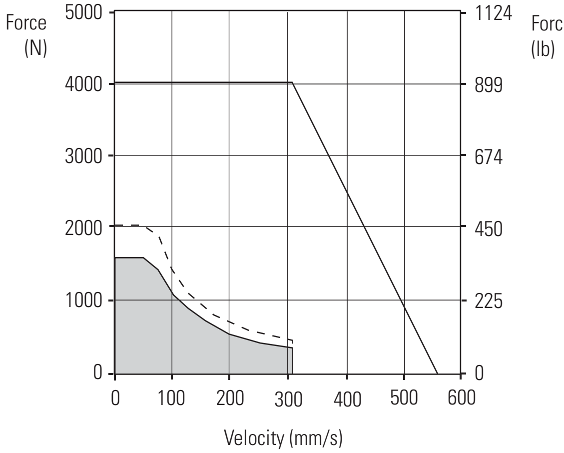
Performance Specifications with Roller Screw Electric Cylinders

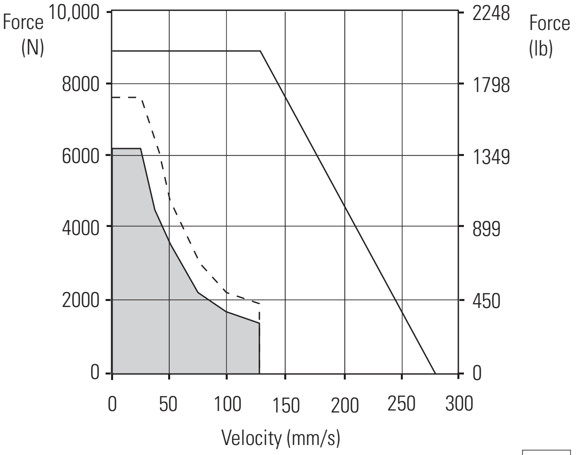
Electric Cylinder	Maximum Speed mm/s (in/s)	System Continuous Stall Current Amps 0-pk	System Continuous Stall Force N (lb)		System Peak Stall Current Amps 0-pk	System Peak Stall Force N (lb)	Rated Output kW	Kinetix 2000 Drives
			25 °C (77 °F)	40 °C (104 °F)				
MPAI-A3*xxx*R	279 (11)	5.61	3781 (850)	3003 (675)	14.14	7562 (1700)	0.39	2093-AM01
MPAI-A3*xxx*S	559 (22)		1891 (425)	1499 (337)		3781 (850)		
MPAI-A4*xxx*R	279 (11)	10.89	7340 (1650)	5827 (1310)	27.44	14,679 (3300)	0.43	2093-AM02
MPAI-A4*xxx*S	559 (22)		3670 (825)	2914 (655)		7340 (1650)		

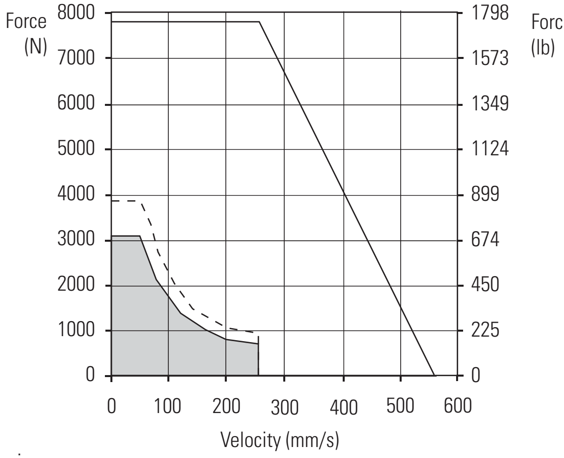
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.8 or later.

Kinetix 2000 Drives/MP-Series Heavy Duty (ball screw) Electric Cylinder Curves


2093-AM01 and MPAI-A3*xxx*C

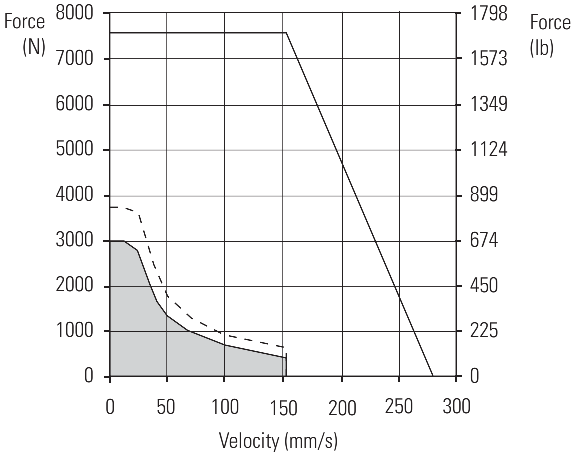

2093-AM01 and MPAI-A3*xxx*E


2093-AM02 and MPAI-A4*xxx*C

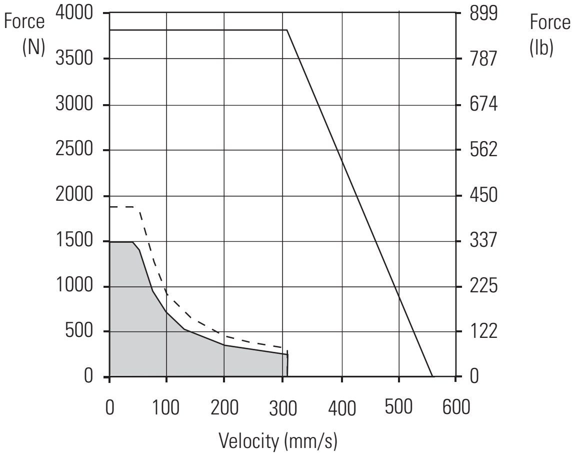

2093-AM02 and MPAI-A4*xxx*E

☐ = Intermittent operating region
- - - = Continuous operating region @ 25 °C (77 °F)
▨ = Continuous operating region @ 40 °C (104 °F)

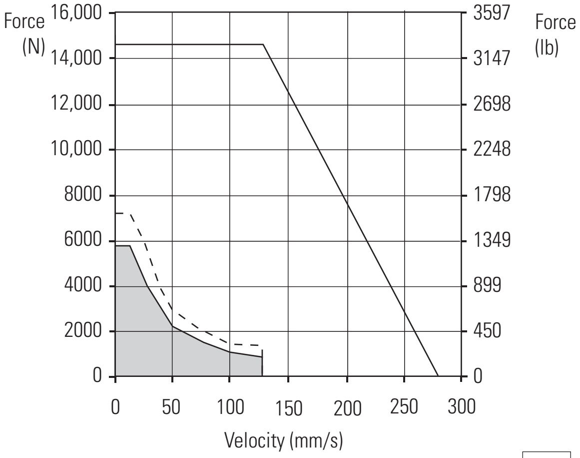
Kinetix 2000 Drives/MP-Series Heavy Duty (roller screw) Electric Cylinder Curves



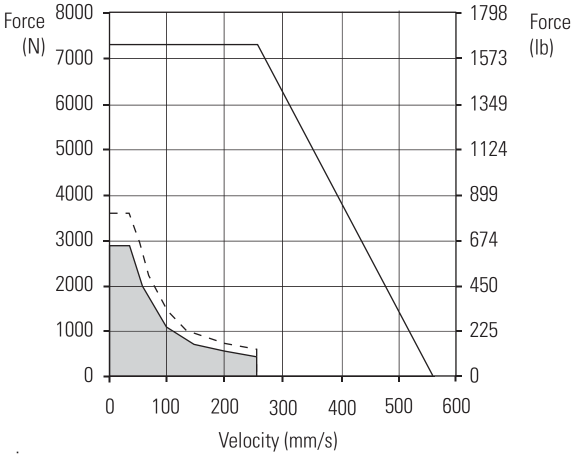
2093-AM01 and MPAI-A3*xxx*R



2093-AM01 and MPAI-A3*xxx*S



2093-AM02 and MPAI-A4*xxx*R



2093-AM02 and MPAI-A4*xxx*S

☐ = Intermittent operating region
- - - = Continuous operating region @ 25 °C (77 °F)
▨ = Continuous operating region @ 40 °C (104 °F)

Kinetix 2000 (230V) Drives with TL-Series Electric Cylinders

This section provides system combination information for the Kinetix 2000 drives when matched with TL-Series (230V) electric cylinders. Included are power/brake and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Electric Cylinder Cable Combinations

Electric Cylinder	Motor Power/Brake Cable	Motor Feedback Cable [(1)]
TLAR-A1*xxx*B TLAR-A1*xxx*E	2090-CPWM6DF-16AA*xx* (standard) (without brake) 2090-CPBM6DF-16AA*xx* (standard) (with brake)	2090-CFBM6DF-CBAA*xx* (standard) Absolute High-resolution Feedback
TLAR-A2*xxx*C TLAR-A2*xxx*F		
TLAR-A3*xxx*E TLAR-A3*xxx*H		

(1) The TLY-A*xxxx*-B motors with 17-bit high-resolution encoder feedback (mounted to the electric cylinder) require the 2090-CFBM6DF-CBAA*xx* flying-lead feedback cable and 2090-K2CK-D15M connector kit with 2090-DA-BAT2 battery. Refer to Breakout Components and Connector Kits beginning on page 440 for more information.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Electric Cylinder (non-brake) Performance Specifications with Kinetix 2000 Drives

Electric Cylinder	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Rated Output** kW	Kinetix 2000 230V Drives
TLAR-A1*xxx*B	150	1.36	240 (53.9)	1.79	300 (67.4)	0.036	2093-AMP2
TLAR-A1*xxx*E	500	2.59	280 (62.9)	3.03	350 (78.7)	0.140	2093-AMP5
TLAR-A2*xxx*C	250	3.03	420 (94.4)	3.41	525 (118)	0.105	
TLAR-A2*xxx*F	640	5.50	640 (144)	7.25	800 (180)	0.350	2093-AM01
TLAR-A3*xxx*E	500	10.0	2000 (450)	12.9	2500 (562)	0.930	2093-AM02
TLAR-A3*xxx*H	1000		1300 (292)	17.2	1625 (365)		

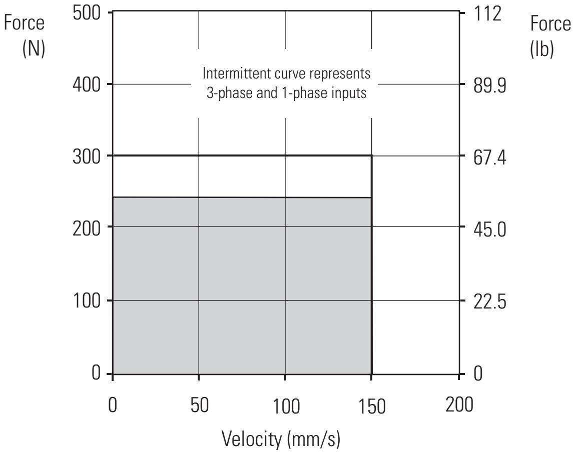
Electric Cylinder (brake) Performance Specifications with Kinetix 2000 Drives

Electric Cylinder	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Rated Output** kW	Kinetix 2000 230V Drives
TLAR-A1*xxx*B	150	1.18	240 (53.9)	1.79	300 (67.4)	0.036	2093-AMP2
TLAR-A1*xxx*E	500	2.24	280 (62.9)	3.03	350 (78.7)	0.140	2093-AMP5
TLAR-A2*xxx*C	250	2.68	420 (94.4)	3.41	525 (118)	0.105	
TLAR-A2*xxx*F	640	4.95	640 (144)	7.25	800 (180)	0.350	2093-AM01
TLAR-A3*xxx*E	500	10.0	2000 (450)	12.9	2500 (562)	0.930	2093-AM02
TLAR-A3*xxx*H	1000		1300 (292)	17.2	1625 (365)		

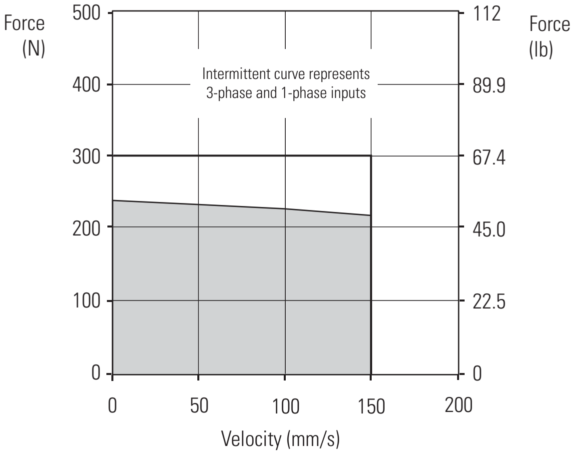
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 2000 (230V) Drives/TL-Series Electric Cylinder Curves

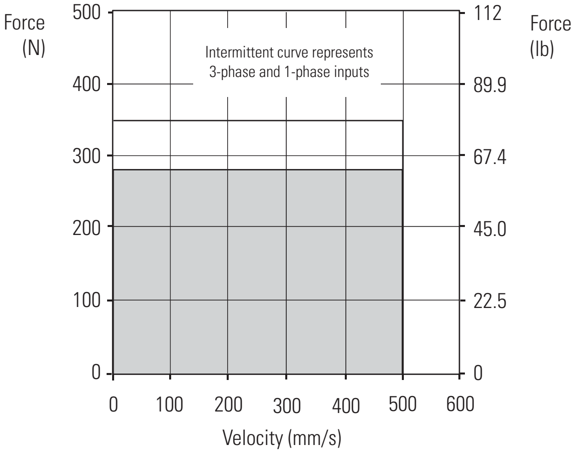
2093-AMP2 and TLAR-A1*xxx*B-B2A



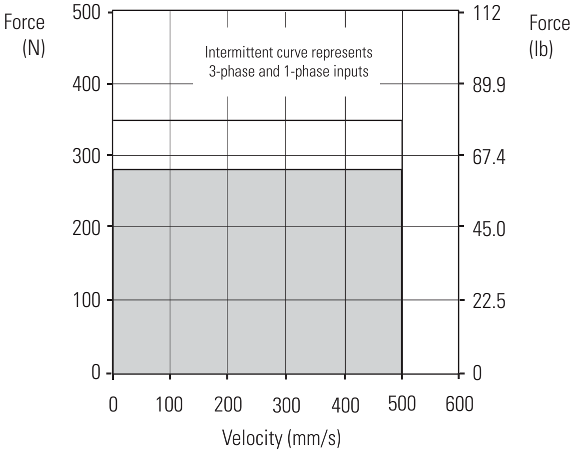
2093-AMP2 and TLAR-A1*xxx*B-B4A (Brake)



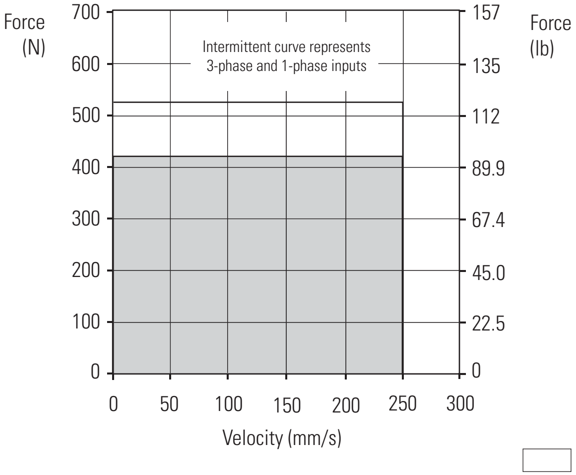
2093-AMP5 and TLAR-A1*xxx*E-B2A



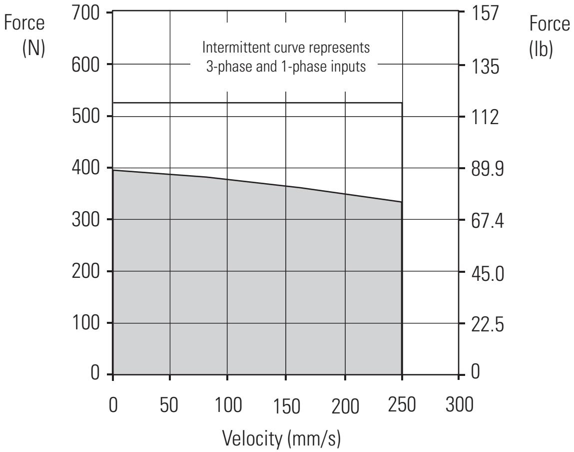
2093-AMP5 and TLAR-A1*xxx*E-B4A (Brake)



2093-AMP5 and TLAR-A2*xxx*C-B2A



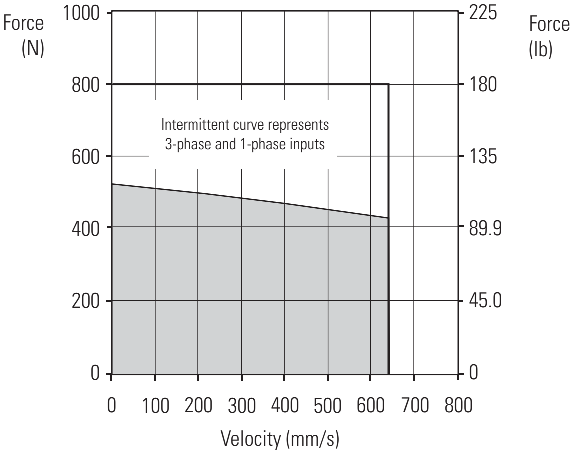
2093-AMP5 and TLAR-A2*xxx*C-B4A (Brake)



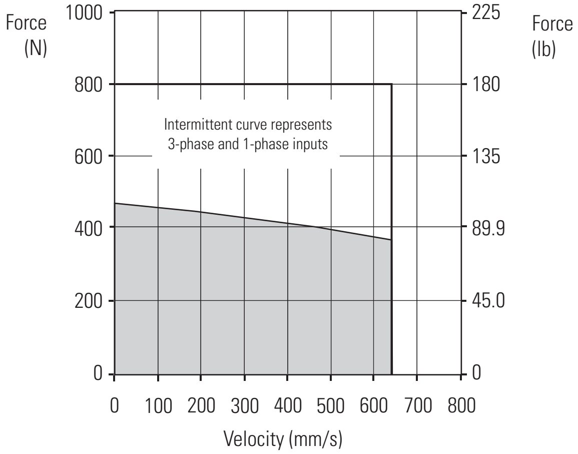
☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 2000 (230V) Drives/TL-Series Electric Cylinder Curves, Continued

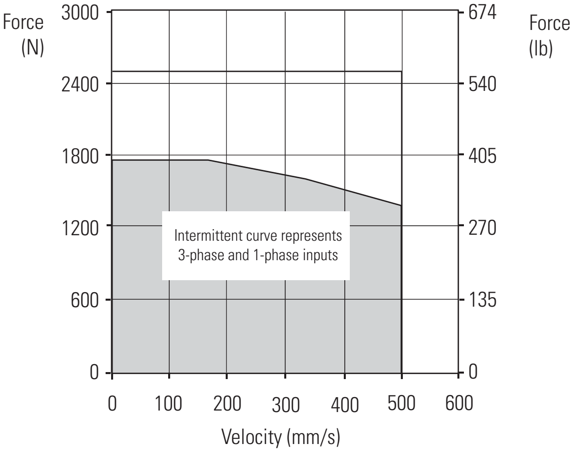
2093-AM01 and TLAR-A2*xxx*F-B2A



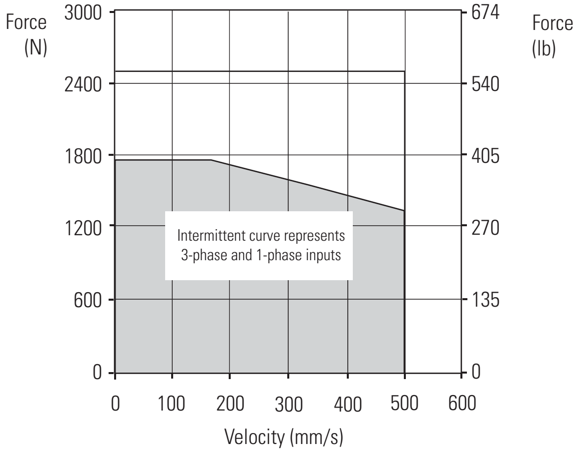
2093-AM01 and TLAR-A2*xxx*F-B4A (Brake)



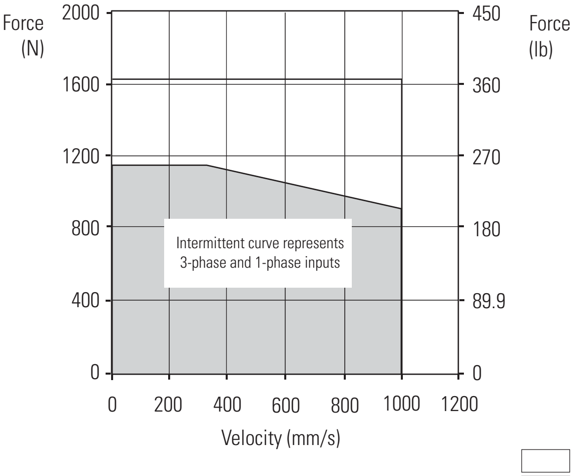
2093-AM02 and TLAR-A3*xxx*E-B2A



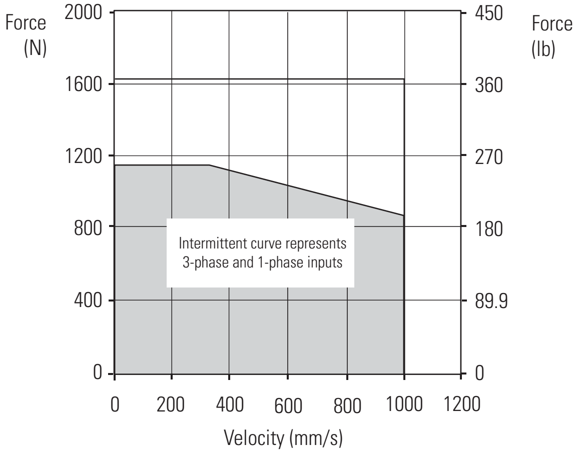
2093-AM02 and TLAR-A3*xxx*E-B4A (Brake)



2093-AM02 and TLAR-A3*xxx*H-B2A



2093-AM02 and TLAR-A3*xxx*H-B4A (Brake)



☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 2000 (230V) Drives with LDC-Series Linear Motors

This section provides system combination information for the Kinetix 2000 drives when matched with LDC-Series iron-core linear motors. Included are power and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Linear Motor Cable Combinations

Linear Motor	Motor Power Cable	Motor Feedback Cable [1]
LDC-C030100-DHT, LDC-C030200-DHT, LDC-C030200-EHT	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-XXNFMF-S*xx* (standard) 2090-CFBM7DF-CDAF*xx* (continuous-flex) Sin/Cos or TTL Encoder Feedback
LDC-C050100-DHT, LDC-C050200-DHT, LDC-C050200-EHT, LDC-C050300-DHT, LDC-C050300-EHT		
LDC-C075200-DHT, LDC-C075200-EHT, LDC-C075300-DHT, LDC-C075300-EHT, LDC-C075400-DHT, LDC-C075400-EHT		
LDC-C100300-DHT, LDC-C100300-EHT, LDC-C100400-DHT, LDC-C100400-EHT, LDC-C100600-DHT		
LDC-C150400-DHT, LDC-C150600-DHT		

(1) Use low-profile connector kit (catalog number 2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

LDC-Series Performance Specifications with Kinetix 2000 (230V) Drives

Linear Motor	**Maximum Speed** m/s (ft/s)	**System Continuous Stall Current** [1] Amps 0-pk	**System Continuous Stall Force** [1] N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Motor Rated Output** kW	**Kinetix 2000 230V Drives**
LDC-C030100-DHT	10.0 (32.8)	4.1…6.1	74…111 (17…25)	12.1	188 (42)	0.37…0.55	2093-AMP5
LDC-C030200-DHT		8.1…12.2	148…222 (33…50)	24.3	375 (84)	0.74…1.11	2093-AM01
LDC-C030200-EHT		4.1…6.1		12.1			2093-AMP5
LDC-C050100-DHT	10.0 (32.8)	3.9…5.9	119…179 (27…40)	11.7	302 (68)	0.59…0.89	2093-AMP5
LDC-C050200-DHT		7.9…11.8	240…359 (54…81)	23.3	600 (135)	1.20…1.79	2093-AM01
LDC-C050200-EHT		3.9…5.9		11.6			2093-AMP5
LDC-C050300-DHT		11.8…17.7	363…544 (82…122)	35.9	941 (212)	1.81…2.72	2093-AM02
LDC-C050300-EHT		3.9…5.9		12.0			2093-AMP5

(1) Values represent the range between no cooling (low value) and water cooling (high value).

Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

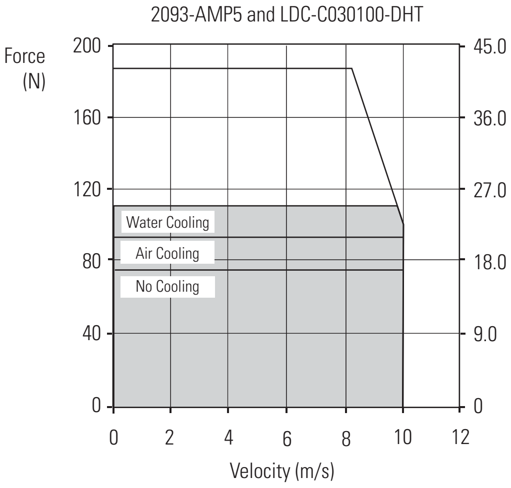
LDC-Series Performance Specifications with Kinetix 2000 (230V) Drives, Continued

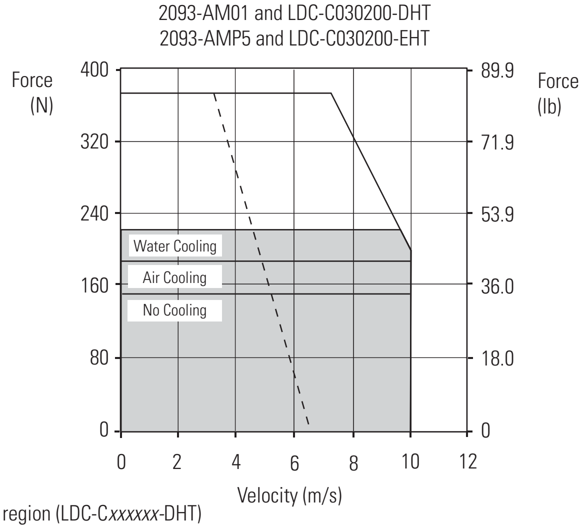
Linear Motor	**Maximum Speed** m/s (ft/s)	**System Continuous Stall Current** (1) Amps 0-pk	**System Continuous Stall Force** (1) N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Motor Rated Output** kW	**Kinetix 2000 230V Drives**
LDC-C075200-DHT	10.0 (32.8)	7.7…11.5	348…523 (78…117)	22.9	882 (198)	1.74…2.61	2093-AM01
LDC-C075200-EHT		3.8…5.7		11.5			2093-AMP5
LDC-C075300-DHT		11.5…17.2	523…784 (117…176)	35.6	1368 (308)	2.61…3.92	2093-AM02
LDC-C075300-EHT		3.8…5.7		11.9			2093-AMP5
LDC-C075400-DHT		15.3…23.0	697…1045 (157…235)	47.4	1824 (410)	3.48…5.22	2093-AM02
LDC-C075400-EHT		7.7…11.5		23.7			2093-AM01
LDC-C100300-DHT	10.0 (32.8)	11.1…16.7	674…1012 (152…227)	34.3	1767 (397)	3.37…5.06	2093-AM02
LDC-C100300-EHT		3.7…5.6		11.4			2093-AMP5
LDC-C100400-DHT		14.8…22.2	899…1349 (202…303)	45.7	2356 (530)	4.49…6.74	2093-AM02
LDC-C100400-EHT		7.4…11.1		22.8			2093-AM01
LDC-C100600-DHT		22.2…33.3	1349…2023 (303…455)	68.5	3534 (794)	6.74…10.11	2093-AM02
LDC-C150400-DHT	10.0 (32.8)	14.1…21.1	1281…1922 (288…432)	45.2	3498 (786)	6.40…9.61	2093-AM02
LDC-C150600-DHT		21.1…31.7	1922…2882 (432…648)	67.8	5246 (1179)	9.61…14.41	2093-AM02

(1) Values represent the range between no cooling (low value) and water cooling (high value).

Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 2000 (230V) Drives/LDC-Series Linear Motor Curves

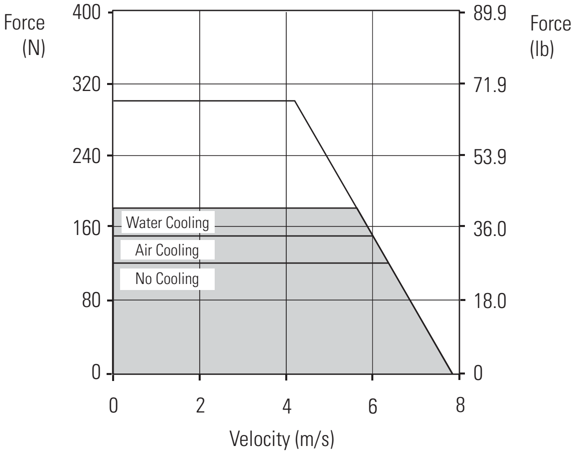




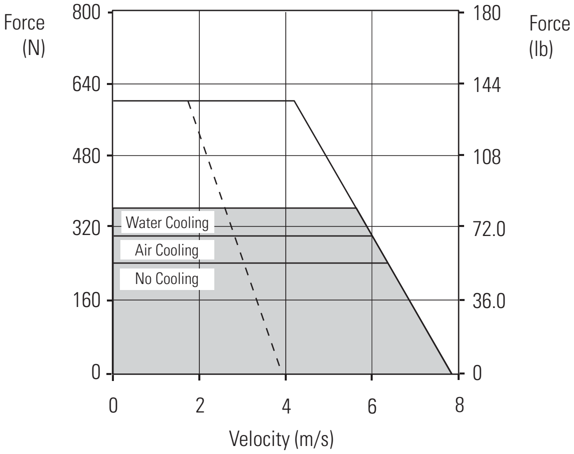
☐ = Intermittent operating region (LDC-C*xxxxxx*-DHT)
- - - = Intermittent operating region (LDC-C*xxxxxx*-EHT)
▨ = Continuous operating region

Kinetix 2000 (230V) Drives/LDC-Series Linear Motor Curves, Continued

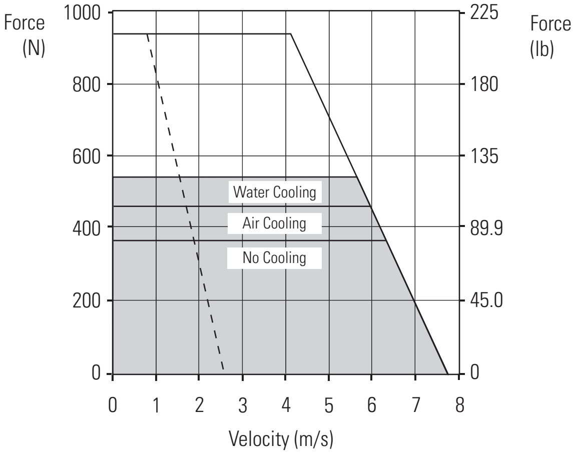
2093-AMP5 and LDC-C050100-DHT



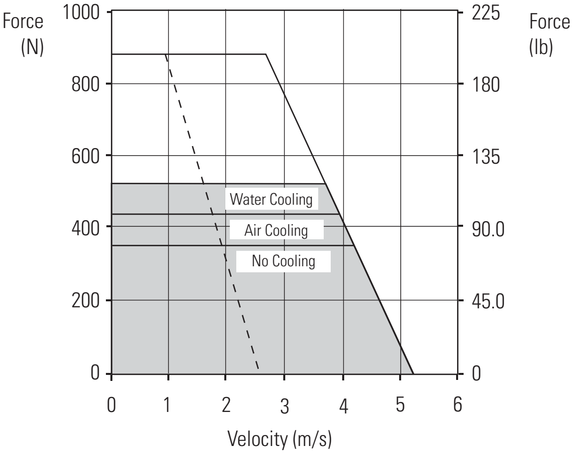
2093-AM01 and LDC-C050200-DHT
2093-AMP5 and LDC-C050200-EHT



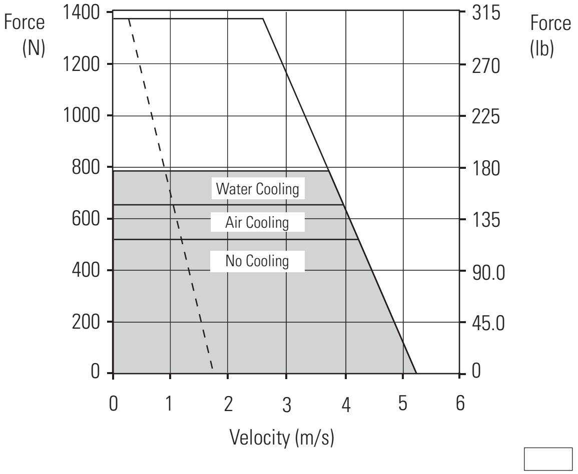
2093-AM02 and LDC-C050300-DHT
2093-AMP5 and LDC-C050300-EHT



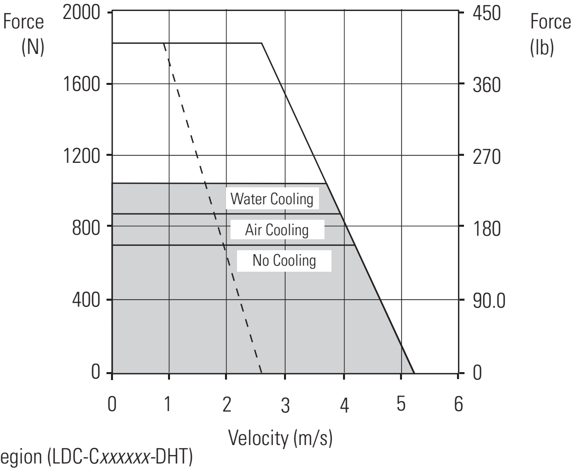
2093-AM01 and LDC-C075200-DHT
2093-AMP5 and LDC-C075200-EHT



2093-AM02 and LDC-C075300-DHT
2093-AMP5 and LDC-C075300-EHT

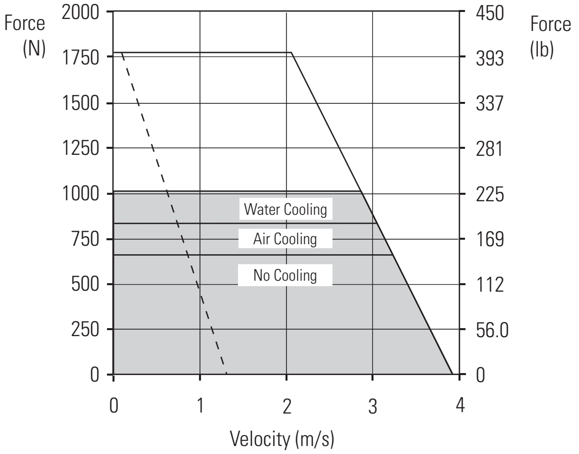


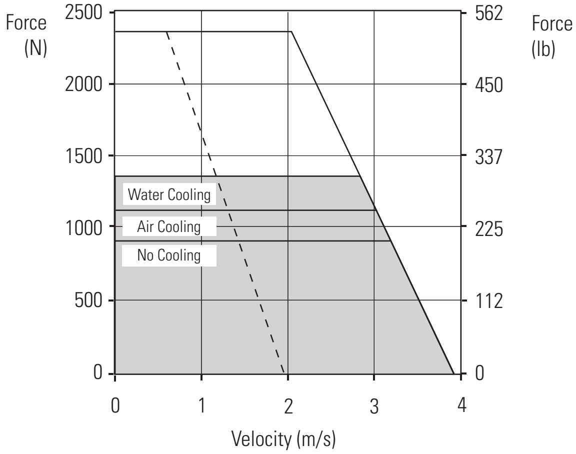
2093-AM02 and LDC-C075400-DHT
2093-AM01 and LDC-C075400-EHT

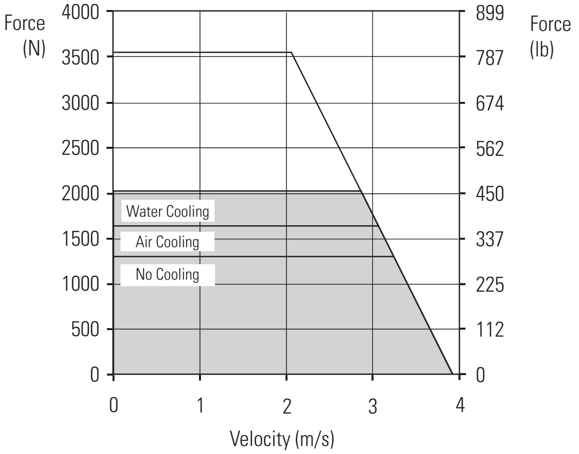


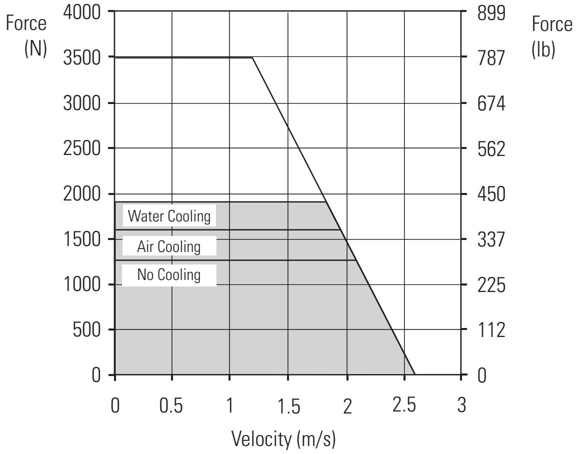
☐ = Intermittent operating region (LDC-C*xxxxxx*-DHT)
- - - = Intermittent operating region (LDC-C*xxxxxx*-EHT)
▨ = Continuous operating region

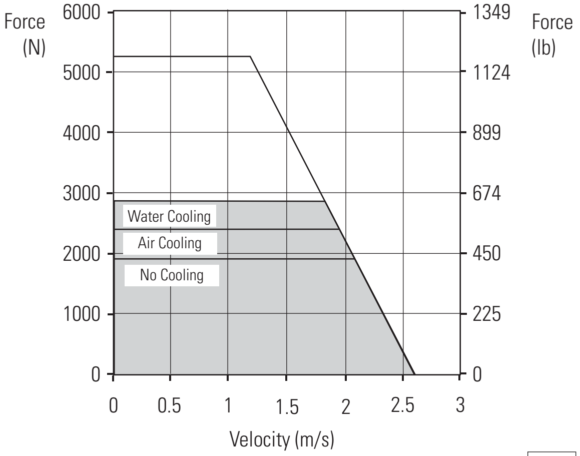
Kinetix 2000 (230V) Drives/LDC-Series Linear Motor Curves, Continued


2093-AM02 and LDC-C100300-DHT
2093-AMP5 and LDC-C100300-EHT


2093-AM02 and LDC-C100400-DHT
2093-AM01 and LDC-C100400-EHT


2093-AM02 and LDC-C100600-DHT


2093-AM02 and LDC-C150400-DHT


2093-AM02 and LDC-C150600-DHT

☐ = Intermittent operating region (LDC-C*xxxxxx*-DHT)
- - - = Intermittent operating region (LDC-C*xxxxxx*-EHT)
▨ = Continuous operating region

Kinetix 2000 (230V) Drives with LDL-Series Linear Motors

This section provides system combination information for the Kinetix 2000 drives when matched with LDL-Series ironless linear motors. Included are power and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Linear Motor Cable Combinations

Linear Motors	Motor Power Cable	Motor Feedback Cable (1)
LDL-N030120-DHT, LDL-N030240-DHT, LDL-N030240-EHT	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-XXNFMF-S*xx* (standard) 2090-CFBM7DF-CDAF*xx* (continuous-flex) Sin/Cos or TTL Encoder Feedback
LDL-N050120-DHT, LDL-N050240-DHT, LDL-N050240-EHT, LDL-N050360-DHT, LDL-N050360-EHT, LDL-N050480-DHT, LDL-N050480-EHT		
LDL-N075480-DHT, LDL-N075480-EHT		
LDL-T030120-DHT, LDL-T030240-DHT, LDL-T030240-EHT		
LDL-T050120-DHT, LDL-T050240-DHT, LDL-T050240-EHT, LDL-T050360-DHT, LDL-T050480-DHT, LDL-T050480-EHT		
LDL-T075480-EHT, LDL-T075480-EHT		

(1) Use low-profile connector kit (catalog number 2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

LDL-Series Performance Specifications with Kinetix 2000 (230V) Drives

Linear Motor	Maximum Speed m/s (ft/s)	System Continuous Stall Current Amps 0-pk	System Continuous Stall Force N (lb)	System Peak Stall Current Amps 0-pk	System Peak Stall Force N (lb)	Linear Motor Rated Output kW	Kinetix 2000 230V Drives
LDL-N030120-DHT	10.0 (32.8)	3.0	63 (14)	9.9	209 (47)	0.31	2093-AMP5
LDL-N030240-DHT		6.0	126 (28)	19.9	417 (94)	0.63	2093-AM01
LDL-N030240-EHT		3.0		9.9			2093-AMP5
LDL-T030120-DHT		3.0	72 (16)	9.9	239 (54)	0.36	2093-AMP5
LDL-T030240-DHT		6.0	144 (32)	19.9	479 (108)	0.72	2093-AM01
LDL-T030240-EHT		3.0		9.9			2093-AMP5

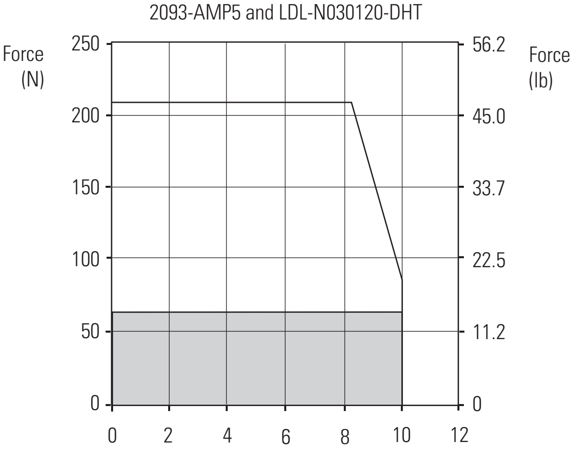
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

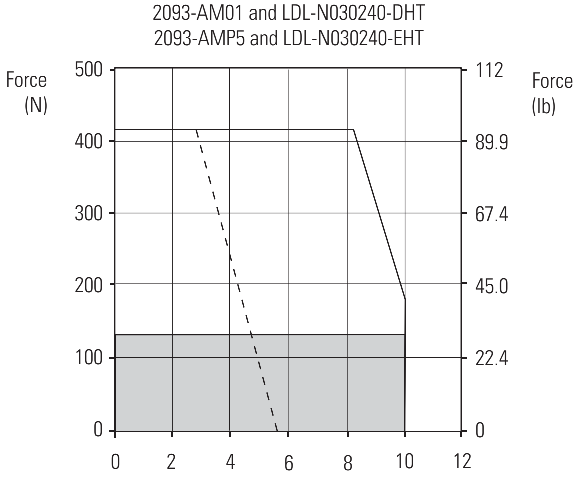
LDL-Series Performance Specifications with Kinetix 2000 (230V) Drives, Continued

Linear Motor	**Maximum Speed** m/s (ft/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Motor Rated Output** kW	**Kinetix 2000 230V Drives**
LDL-N050120-DHT	10.0 (32.8)	2.7	96 (22)	9.1	317 (71)	0.48	2093-AMP5
LDL-N050240-DHT		5.5	191 (43)	18.1	635 (143)	0.95	2093-AM01
LDL-N050240-EHT		2.7		9.1			2093-AMP5
LDL-N050360-DHT		8.2	287 (65)	27.2	952 (214)	1.43	2093-AM02
LDL-N050360-EHT		2.7		9.1			2093-AMP5
LDL-N050480-DHT		10.9	383 (86)	36.3	1269 (285)	1.91	2093-AM02
LDL-N050480-EHT		5.5		18.1			2093-AM01
LDL-T050120-DHT		2.7	110 (25)	9.1	364 (82)	0.55	2093-AMP5
LDL-T050240-DHT		5.5	220 (49)	18.1	728 (164)	1.10	2093-AM01
LDL-T050240-EHT		2.7		9.1			2093-AMP5
LDL-T050360-DHT		8.2	329 (74)	27.2	1093 (246)	1.64	2093-AM02
LDL-T050480-DHT		10.9	439 (99)	36.3	1457 (327)	2.19	2093-AM02
LDL-T050480-EHT		5.5		18.1			2093-AM01
LDL-N075480-DHT	10.0 (32.8)	9.9	519 (117)	32.8	1723 (387)	2.59	2093-AM02
LDL-N075480-EHT		4.9		16.4			2093-AM01
LDL-T075480-DHT		9.9	596 (134)	32.8	1977 (444)	2.98	2093-AM02
LDL-T075480-EHT		4.9		16.4			2093-AM01

Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 2000 (230V) Drives/LDL-Series Linear Motor Curves





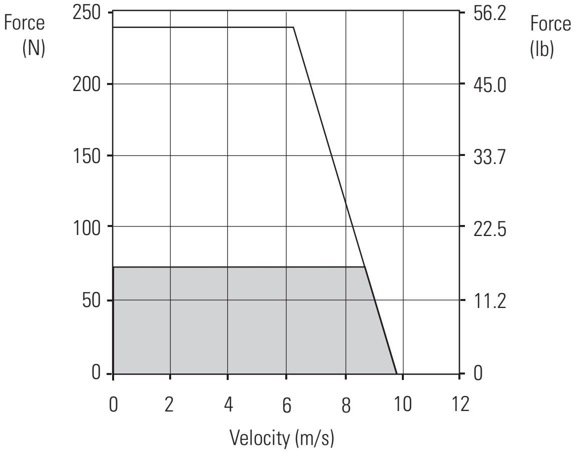
☐ = Intermittent operating region (LDL-*xxxxxxx*-DHT)

- - - = Intermittent operating region (LDL-*xxxxxxx*-EHT)

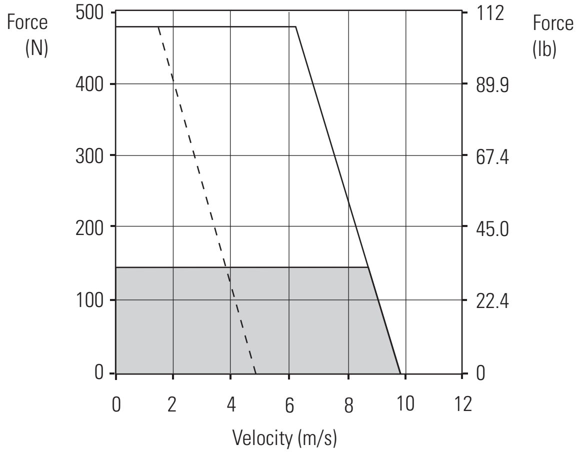
▨ = Continuous operating region

Kinetix 2000 (230V) Drives/LDL-Series Linear Motor Curves, Continued

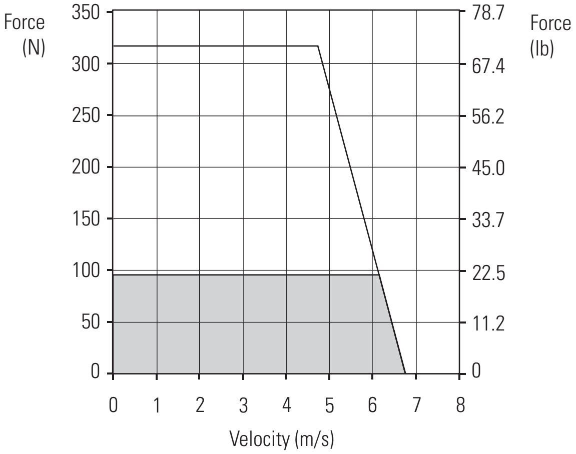
2093-AMP5 and LDL-T030120-DHT



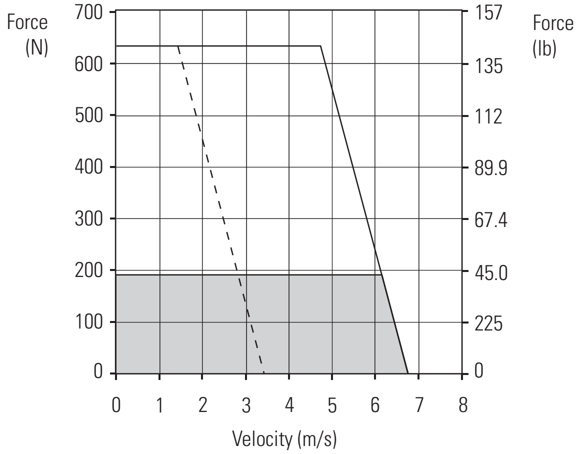
2093-AM01 and LDL-T030240-DHT
2093-AMP5 and LDL-T030240-EHT



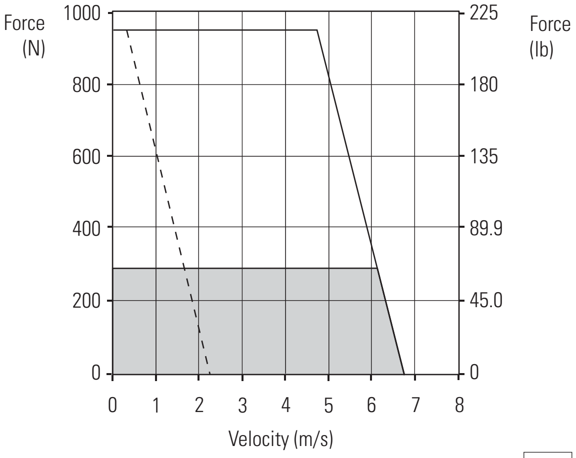
2093-AMP5 and LDL-N050120-DHT



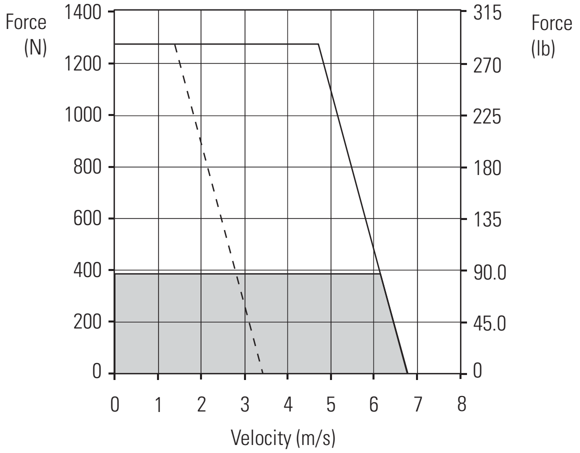
2093-AM01 and LDL-N050240-DHT,
2093-AMP5 and LDL-N050240-EHT



2093-AM02 and LDL-N050360-DHT
2093-AMP5 and LDL-N050360-EHT

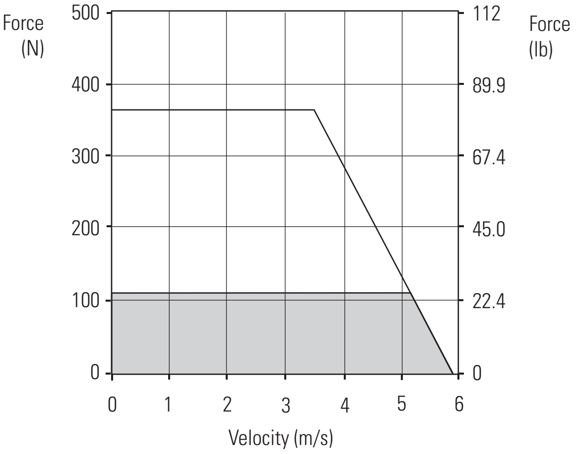


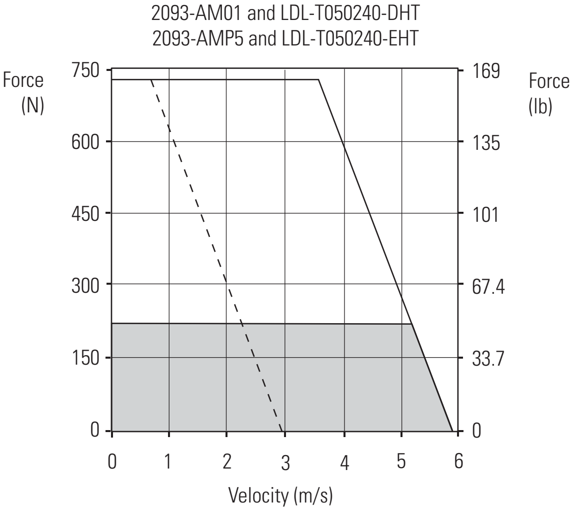
2093-AM02 and LDL-N050480-DHT
2093-AM01 and LDL-N050480-EHT

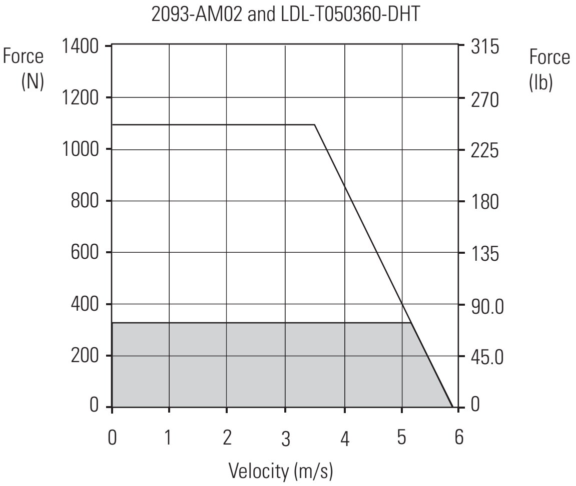


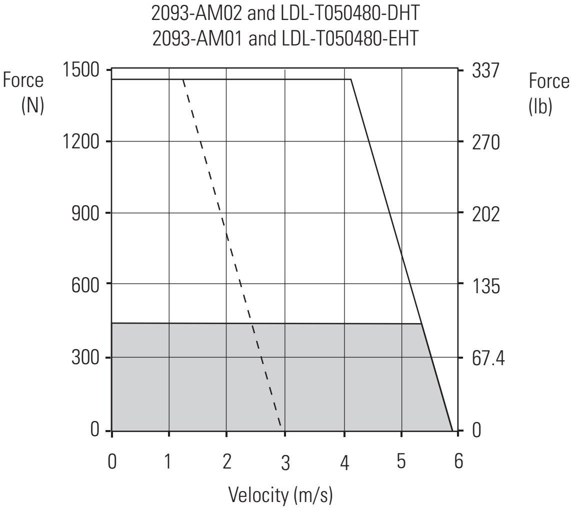
☐ = Intermittent operating region (LDL-*xxxxxxx*-DHT)
- - - = Intermittent operating region (LDL-*xxxxxxx*-EHT)
▨ = Continuous operating region

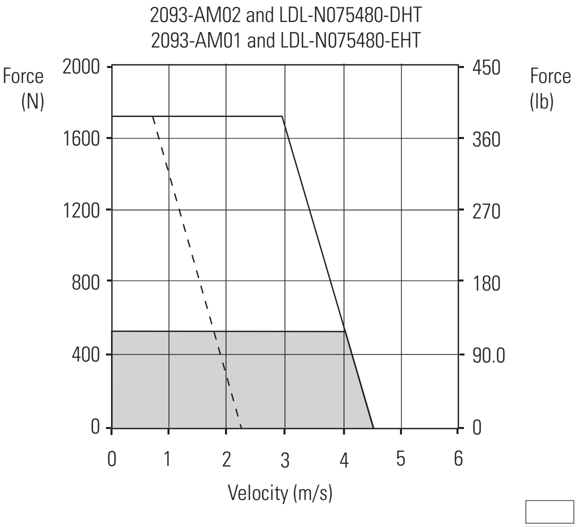
Kinetix 2000 (230V) Drives/LDL-Series Linear Motor Curves, Continued

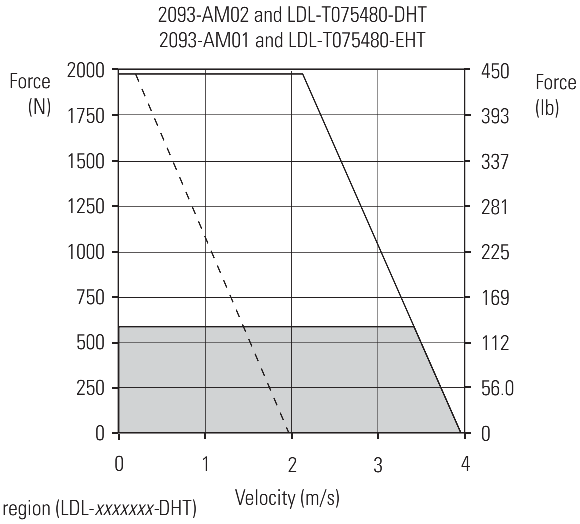

2093-AMP5 and LDL-T050120-DHT


2093-AM01 and LDL-T050240-DHT
2093-AMP5 and LDL-T050240-EHT


2093-AM02 and LDL-T050360-DHT


2093-AM02 and LDL-T050480-DHT
2093-AM01 and LDL-T050480-EHT


2093-AM02 and LDL-N075480-DHT
2093-AM01 and LDL-N075480-EHT


2093-AM02 and LDL-T075480-DHT
2093-AM01 and LDL-T075480-EHT

☐ = Intermittent operating region (LDL-*xxxxxxx*-DHT)
- - - = Intermittent operating region (LDL-*xxxxxxx*-EHT)
▨ = Continuous operating region

Kinetix 300 (240V) Drives with MP-Series Integrated Linear Stages

This section provides system combination information for the Kinetix 300 (240V) drives when matched with MP-Series (230V) integrated ballscrew linear stages. Included are motor power/brake and feedback cable catalog numbers, system performance specifications, and force/velocity curves.

Linear Stage Cable Combinations

Linear Stage	Motor Power/Brake Cable	Motor Feedback Cable [1]
MPAS-A*xxxx*1-V05S*x*A	2090-XXNPMF-16S*xx* [2]	2090-XXNFMF-S*xx* [3] Absolute High-resolution Feedback
MPAS-A*xxxx*2-V20S*x*A		

(1) Use low-profile connector kit (catalog number 2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

(2) These cables are available as standard (catalog number 2090-XXNPMF-16S*xx*) or continuous-flex (catalog number 2090-CP*x*M4DF-16AF*xx*).

(3) These cables are available as standard (catalog number 2090-XXNFMF-S*xx*) or continuous-flex (catalog number 2090-CFBM4DF-CDAF*xx*).

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Linear Stage Performance Specifications with Kinetix 300 (240V) Drives

Linear Stage	Maximum Speed mm/s (in/s)	System Continuous Stall Current Amps 0-pk	System Continuous Stall Force N (lb)	System Peak Stall Current Amps 0-pk	System Peak Stall Force N (lb)	Linear Stage Rated Output kW	Kinetix 300 240V Drives
MPAS-A*xxxx*1-V05S*x*A	200 (7.9) [1]	3.09	521 (117)	6.10	1212 (272)	0.37	2097-V33PR1
MPAS-A*xxxx*2-V20S*x*A	1124 (44.3) [2]	4.54	462 (104)	9.10	968 (218)	0.62	2097-V33PR3

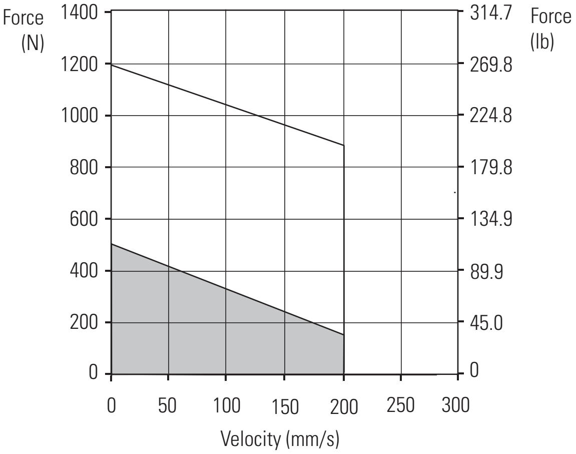
(1) For 900 mm stroke length, maximum speed is 176 mm/s (6.9 in/s). For 1020 mm stroke length, maximum speed is 143 mm/s (5.6 in/s).

(2) For 780 mm stroke length, maximum speed is 889 mm/s (35.0 in/s). For 900 mm stroke length, maximum speed is 715 mm/s (28.2 in/s). For 1020 mm stroke length, maximum speed is 582 mm/s (22.9 in/s).

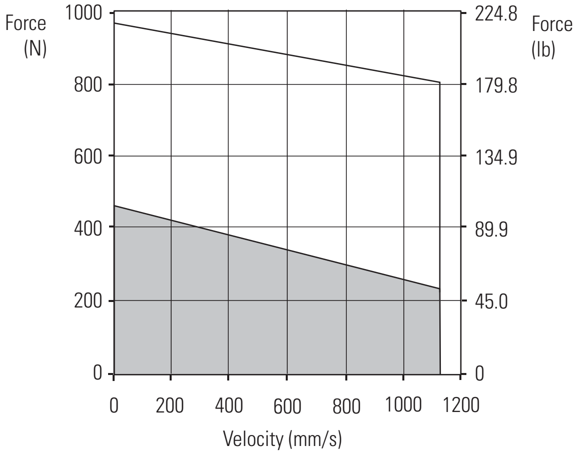
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 40 °C (104 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 300 (240V) Drives/MP-Series Integrated Linear Stage Curves

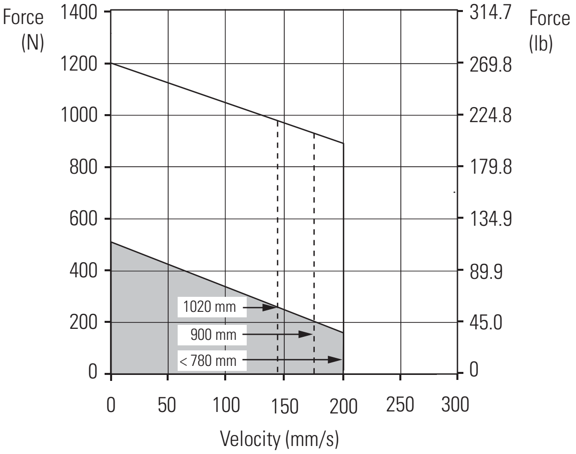
2097-V33PR1 and MPAS-A6*xxx*1-V05S*x*A



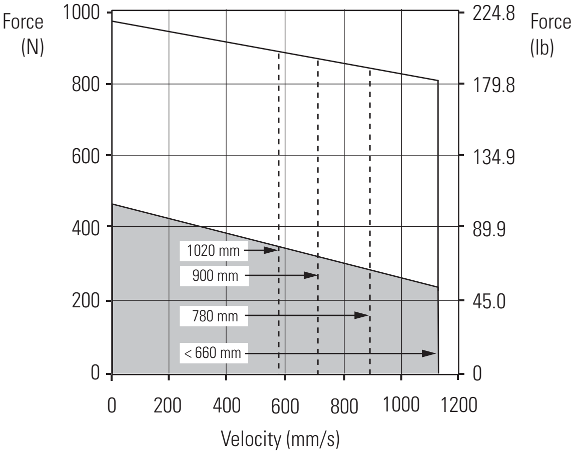
2097-V33PR3 and MPAS-A6*xxx*2-V20S*x*A



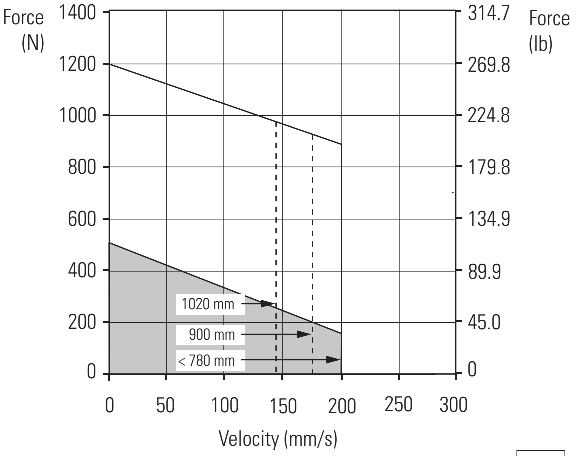
2097-V33PR1 and MPAS-A8*xxx*1-V05S*x*A



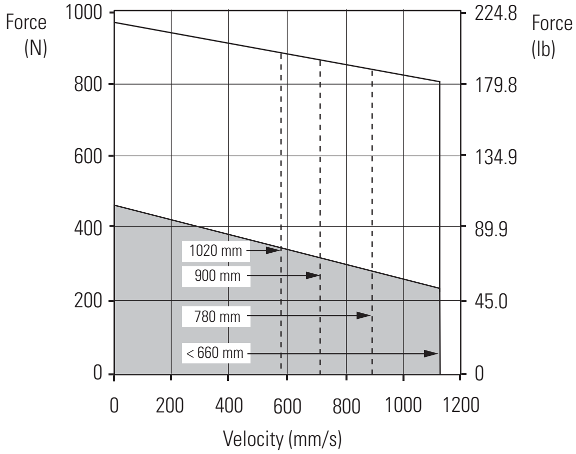
2097-V33PR3 and MPAS-A8*xxx*2-V20S*x*A



2097-V33PR1 and MPAS-A9*xxx*1-V05S*x*A



2097-V33PR3 and MPAS-A9*xxx*2-V20S*x*A



☐ = Intermittent operating region
▨ = Continuous operating region
- - - = System operation for specified stroke length

Kinetix 300 (480V) Drives with MP-Series Linear Stages

This section provides system combination information for the Kinetix 300 (480V) drives when matched with MP-Series (460V) integrated ballscrew linear stages. Included are power/brake and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Linear Stage Cable Combinations

Linear Stage	**Motor Power/Brake Cable**	**Motor Feedback Cable** (1)
MPAS-B*xxxx*1-V05S*x*A	2090-XXNPMF-16S*xx* (2)	2090-XXNFMF-S*xx* (3) Absolute High-resolution Feedback
MPAS-B*xxxx*2-V20S*x*A		

(1) Use low-profile connector kit (catalog number 2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

(2) These cables are available as standard (catalog number 2090-XXNPMF-16S*xx*) or continuous-flex (catalog number 2090-CP*x*M4DF-16AF*xx*).

(3) These cables are available as standard (catalog number 2090-XXNFMF-S*xx*) or continuous-flex (catalog number 2090-CFBM4DF-CDAF*xx*).

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Linear Stage Performance Specifications with Kinetix 300 (480V) Drives

Linear Stage	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Stage Rated Output** kW	**Kinetix 300 480V Drives**
MPAS-B*xxxx*1-V05S*x*A	200 (7.9) (1)	1.75	521 (117)	3.50	1212 (272)	0.138	2097-V34PR3
MPAS-B*xxxx*2-V20S*x*A	1124 (44.3) (2)	3.30	462 (104)	6.60	968 (218)	0.52	2097-V34PR5

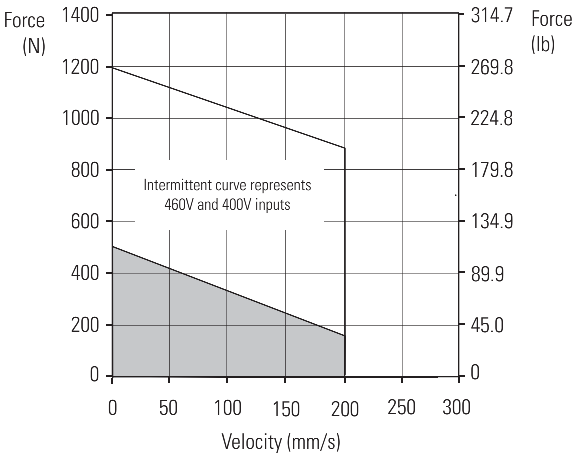
(1) For 900 mm stroke length, maximum speed is 176 mm/s (6.9 in/s). For 1020 mm stroke length, maximum speed is 143 mm/s (5.6 in/s).

(2) For 780 mm stroke length, maximum speed is 889 mm/s (35.0 in/s). For 900 mm stroke length, maximum speed is 715 mm/s (28.2 in/s). For 1020 mm stroke length, maximum speed is 582 mm/s (22.9 in/s).

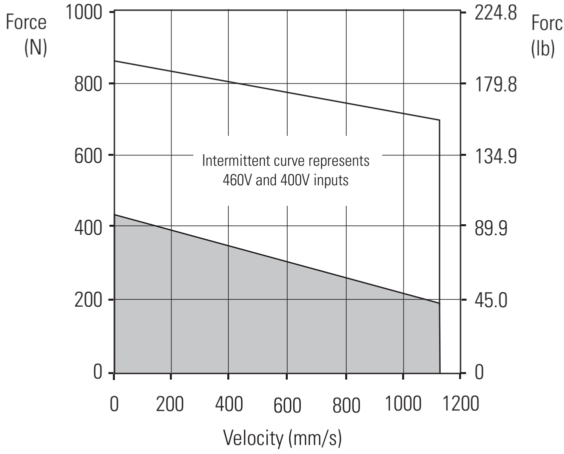
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 40 °C (104 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 300 (480V) Drives/MP-Series Integrated Linear Stage Curves

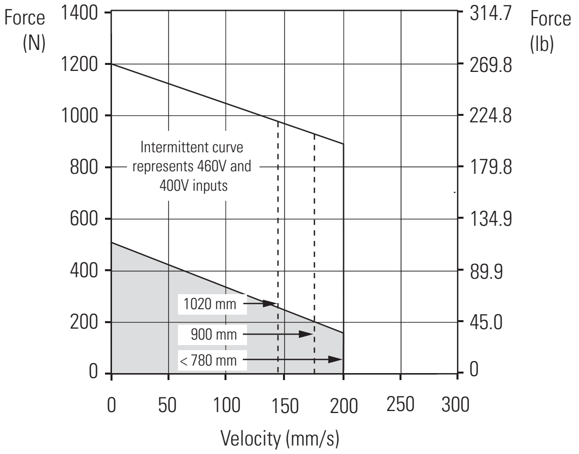
2097-V34PR3 and MPAS-B6*xxx*1-V05S*x*A



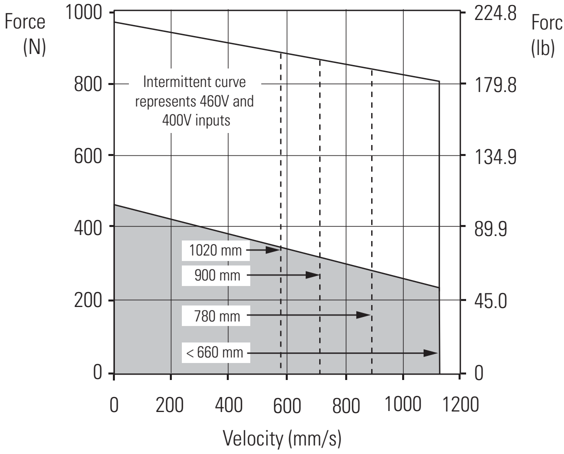
2097-V34PR5 and MPAS-B6*xxx*2-V20S*x*A



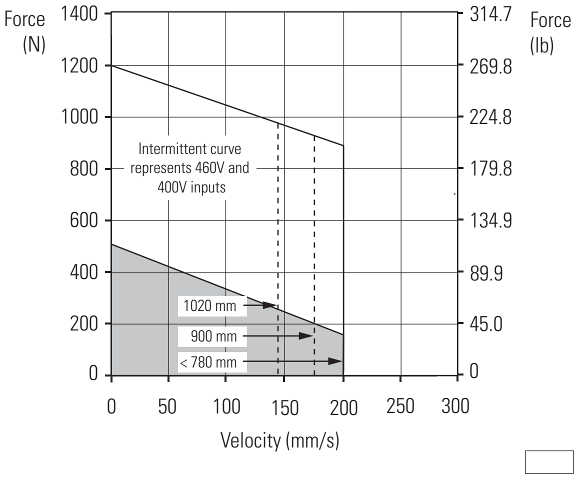
2097-V34PR3 and MPAS-B8*xxx*1-V05S*x*A



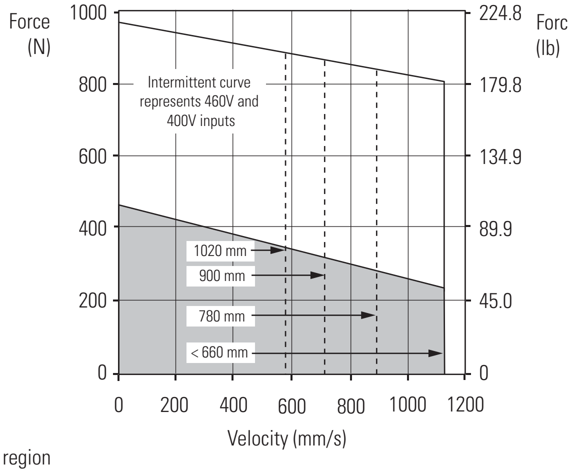
2097-V34PR5 and MPAS-B8*xxx*2-V20S*x*A



2097-V34PR3 and MPAS-B9*xxx*1-V05S*x*A



2097-V34PR5 and MPAS-B9*xxx*2-V20S*x*A



☐ = Intermittent operating region
▨ = Continuous operating region
- - - - = System operation for specified stroke length

Kinetix 300 Drives with MP-Series Electric Cylinders

This section provides system combination information for the Kinetix 300 drives when matched with MP-Series electric cylinders. Included are power/brake and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Electric Cylinder Cable Combinations

Electric Cylinder	Motor Power/Brake Cable	Motor Feedback Cable [1]
MPAR-A/B1*xxx*B MPAR-A/B1*xxx*E MPAR-A/B2*xxx*C MPAR-A/B2*xxx*F	2090-XXNPMF-16S*xx* [2]	2090-XXNFMF-S*xx* [3] Absolute High-resolution Feedback
MPAR-A/B3*xxx*E MPAR-A/B3*xxx*H	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback

(1) Use low-profile connector kit (2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

(2) These cables are available as standard (catalog number 2090-XXNPMF-16S*xx*) or continuous-flex (catalog number 2090-CP*x*M4DF-16AF*xx*).

(3) These cables are available as standard (catalog number 2090-XXNFMF-S*xx*) or continuous-flex (catalog number 2090-CFBM4DF-CDAF*xx*).

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Electric Cylinder Performance Specifications with Kinetix 300 Drives

Performance Specifications with Kinetix 300 (240V) Drives

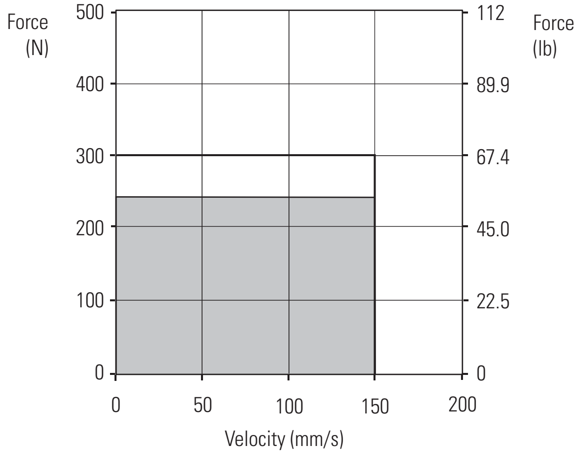
Electric Cylinder	Maximum Speed mm/s (in/s)	System Continuous Stall Current Amps 0-pk	System Continuous Stall Force N (lb)	System Peak Stall Current Amps 0-pk	System Peak Stall Force N (lb)	Rated Output kW	Kinetix 300 Drives
MPAR-A1*xxx*B	150	1.15	240 (53.9)	1.35	300 (67.4)	0.036	2097-V33PR1
MPAR-A1*xxx*E	500	2.16	280 (62.9)	2.48	350 (78.7)	0.140	
MPAR-A2*xxx*C	250	2.42	420 (94.4)	2.72	525 (118)	0.105	
MPAR-A2*xxx*F	640	4.54	640 (144)	5.41	800 (180)	0.410	
MPAR-A3*xxx*E	500	10.33	2000 (450)	12.34	2500 (562)	1.00	2097-V33PR5
MPAR-A3*xxx*H	1000	12.20	1300 (292)	16.40	1625 (365)	1.30	

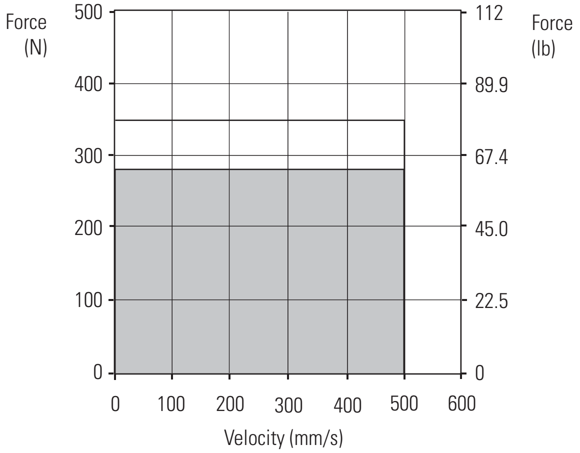
Performance Specifications with Kinetix 300 (480V) Drives

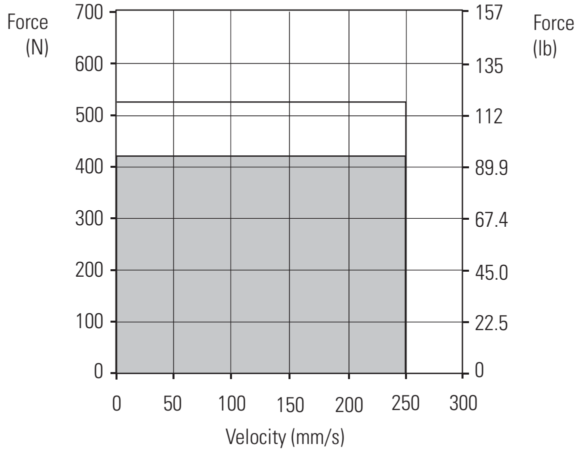
Electric Cylinder	Maximum Speed mm/s (in/s)	System Continuous Stall Current Amps 0-pk	System Continuous Stall Force N (lb)	System Peak Stall Current Amps 0-pk	System Peak Stall Force N (lb)	Rated Output kW	Kinetix 300 Drives
MPAR-B1*xxx*B	150	1.15	240 (53.9)	1.35	300 (67.4)	0.036	2097-V34PR3
MPAR-B1*xxx*E	500	1.49	280 (62.9)	1.71	350 (78.7)	0.140	
MPAR-B2*xxx*C	250	1.67	420 (94.4)	1.90	525 (118)	0.105	
MPAR-B2*xxx*F	640	3.29	640 (144)	3.93	800 (180)	0.410	
MPAR-B3*xxx*E	500	5.16	2000 (450)	6.17	2500 (562)	1.00	
MPAR-B3*xxx*H	1000	6.13	1300 (292)	6.79	1625 (365)	1.30	

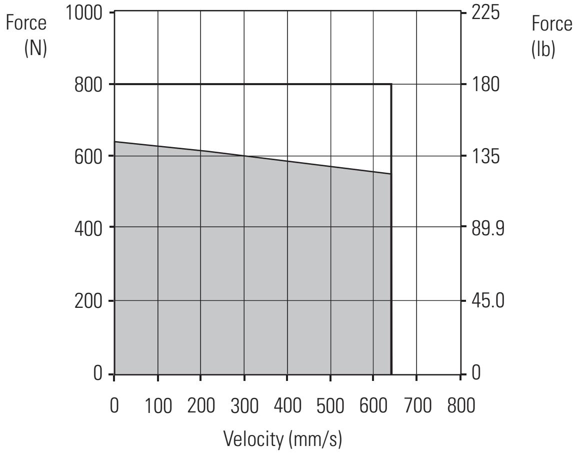
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 40 °C (104 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

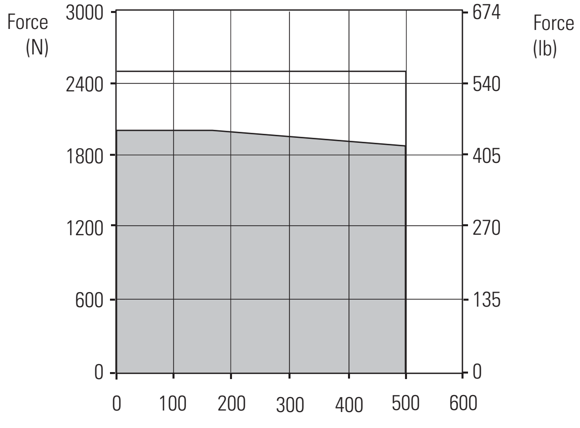
Kinetix 300 Drives/MP-Series Electric Cylinder Curves

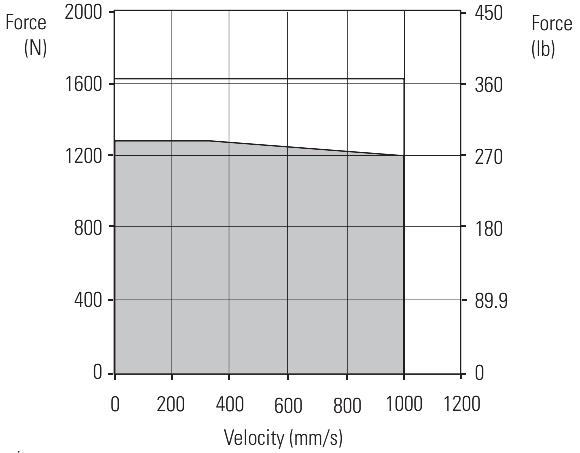

2097-V33PR1 and MPAR-A1*xxx*B-V*x*A
2097-V34PR3 and MPAR-B1*xxx*B-V*x*A


2097-V33PR1 and MPAR-A1*xxx*E-V*x*A
2097-V34PR3 and MPAR-B1*xxx*E-V*x*A


2097-V33PR1 and MPAR-A2*xxx*C-V*x*A
2097-V34PR3 and MPAR-B2*xxx*C-V*x*A


2097-V33PR1 and MPAR-A2*xxx*F-V*x*A
2097-V34PR3 and MPAR-B2*xxx*F-V*x*A


2097-V33PR5 and MPAR-A3*xxx*E-V*x*A
2097-V34PR3 and MPAR-B3*xxx*E-V*x*A


2097-V33PR5 and MPAR-A3*xxx*H-V*x*A
2097-V34PR3 and MPAR-B3*xxx*H-V*x*A

☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 300 Drives/MP-Series Heavy Duty Electric Cylinders

This section provides system combination information for the Kinetix 300 drives when matched with MP-Series heavy-duty electric cylinders. Included are power/brake and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Electric Cylinder Cable Combinations

Electric Cylinder	Motor Power/Brake Cable	Motor Feedback Cable [1]
MPAI-A/B3*xxx*C MPAI-A/B3*xxx*E MPAI-A/B3*xxx*R MPAI-A/B3*xxx*S	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback
MPAI-A/B4*xxx*C MPAI-A/B4*xxx*E MPAI-A/B4*xxx*R MPAI-A/B4*xxx*S		

(1) Use low-profile connector kit (catalog number 2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Electric Cylinder Performance Specifications with Kinetix 300 (240V) Drives

Performance Specifications with Ball Screw Electric Cylinders

Electric Cylinder	Maximum Speed mm/s (in/s)	System Continuous Stall Current Amps 0-pk	System Continuous Stall Force N (lb)		System Peak Stall Current Amps 0-pk	System Peak Stall Force N (lb)	Rated Output kW	Kinetix 300 240V Drives
			25 °C (77 °F)	40 °C (104 °F)				
MPAI-A3*xxx*C	279 (11)	5.61	4003 (900)	3176 (714)	8.40	4448 (1000)	0.39	2097-V33PR3
MPAI-A3*xxx*E	559 (22)		2002 (450)	1588 (357)	14.14	4003 (900)		
MPAI-A4*xxx*C	279 (11)	10.89	7784 (1750)	6179 (1389)	17.07	8896 (2000)	0.43	2097-V33PR5
MPAI-A4*xxx*E	559 (22)		3892 (875)	3092 (695)	27.44	7784 (1750)		

Performance Specifications with Roller Screw Electric Cylinders

Electric Cylinder	Maximum Speed mm/s (in/s)	System Continuous Stall Current Amps 0-pk	System Continuous Stall Force N (lb)		System Peak Stall Current Amps 0-pk	System Peak Stall Force N (lb)	Rated Output kW	Kinetix 300 240V Drives
			25 °C (77 °F)	40 °C (104 °F)				
MPAI-A3*xxx*R	279 (11)	5.61	3781 (850)	3003 (675)	14.14	7562 (1700)	0.39	2097-V33PR3
MPAI-A3*xxx*S	559 (22)		1891 (425)	1499 (337)		3781 (850)		
MPAI-A4*xxx*R	279 (11)	10.89	7340 (1650)	5827 (1310)	27.44	14,679 (3300)	0.43	2097-V33PR5
MPAI-A4*xxx*S	559 (22)		3670 (825)	2914 (655)		7340 (1650)		

Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 40 °C (104 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.8 or later.

Electric Cylinder Performance Specifications with Kinetix 300 (480V) Drives

Performance Specifications with Ball Screw Electric Cylinders

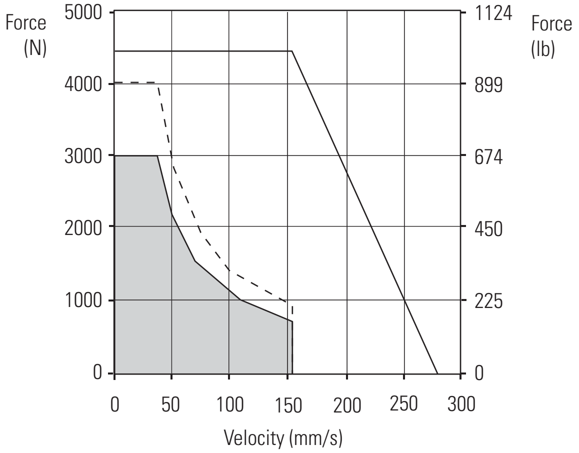
Electric Cylinder	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)		**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Rated Output** kW	**Kinetix 300 480V Drives**
			25 °C (77 °F)	40 °C (104 °F)				
MPAI-B3*xxx*C	279 (11)	2.81	4003 (900)	3176 (714)	4.30	4448 (1000)	0.39	2097-V34PR3
MPAI-B3*xxx*E	559 (22)		2002 (450)	1588 (357)	7.07	4003 (900)		
MPAI-B4*xxx*C	279 (11)	5.61	7784 (1750)	6179 (1389)	8.68	8896 (2000)	0.43	2097-V34PR5
MPAI-B4*xxx*E	559 (22)		3892 (875)	3092 (695)	14.14	7784 (1750)		

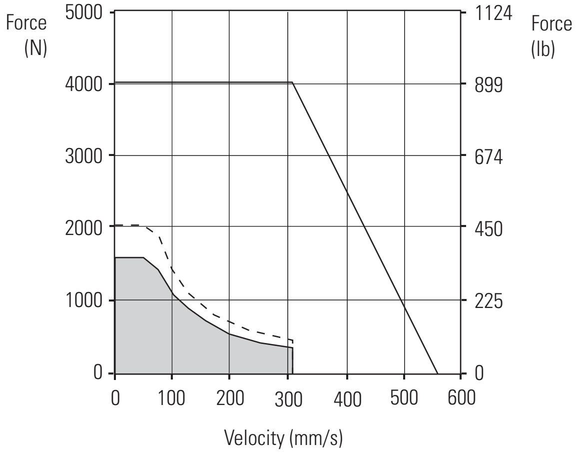
Performance Specifications with Roller Screw Electric Cylinders

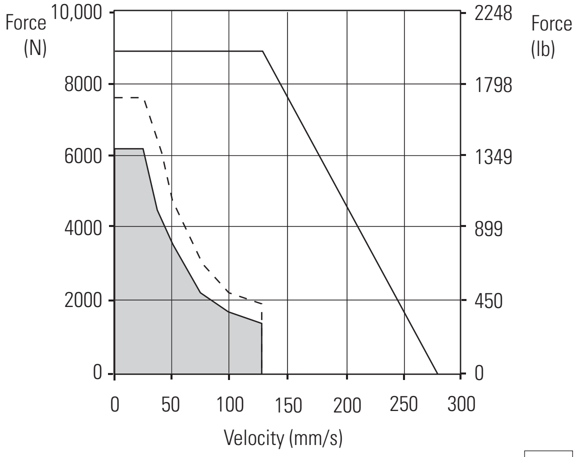
Electric Cylinder	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)		**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Rated Output** kW	**Kinetix 300 480V Drives**
			25 °C (77 °F)	40 °C (104 °F)				
MPAI-B3*xxx*R	279 (11)	2.81	3781 (850)	3003 (675)	7.07	7562 (1700)	0.39	2097-V34PR3
MPAI-B3*xxx*S	559 (22)		1891 (425)	1499 (337)		3781 (850)		
MPAI-B4*xxx*R	279 (11)	5.61	7340 (1650)	5827 (1310)	14.14	14,679 (3300)	0.43	2097-V34PR5
MPAI-B4*xxx*S	559 (22)		3670 (825)	2914 (655)		7340 (1650)		

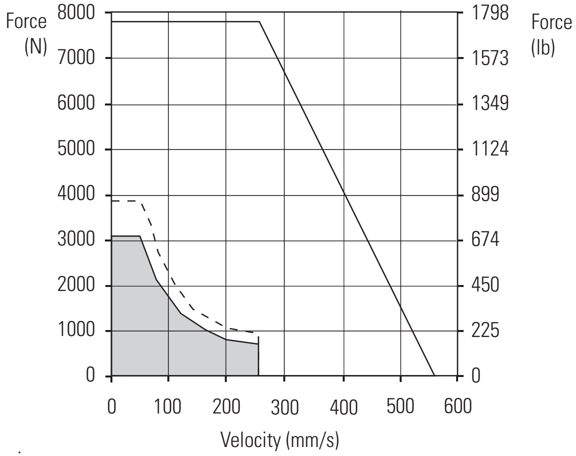
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 40 °C (104 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.8 or later.

Kinetix 300 Drives/MP-Series Heavy Duty (ball screw) Electric Cylinder Curves


2097-V33PR3 and MPAI-A3*xxx*C
2097-V34PR3 and MPAI-B3*xxx*C

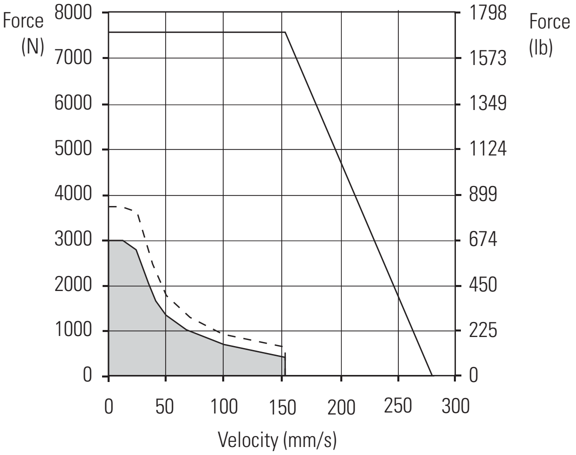

2097-V33PR3 and MPAI-A3*xxx*E
2097-V34PR3 and MPAI-B3*xxx*E

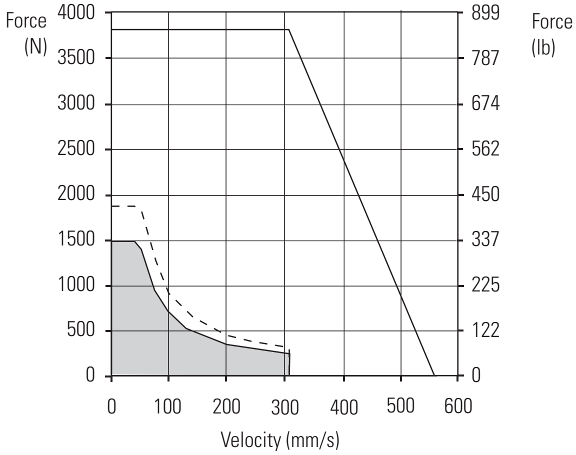

2097-V33PR5 and MPAI-A4*xxx*C
2097-V34PR5 and MPAI-B4*xxx*C

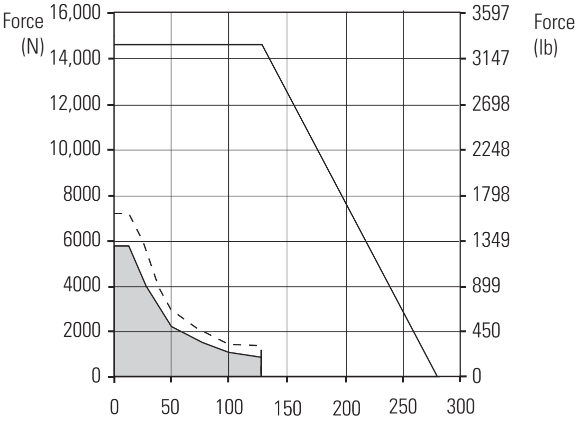

2097-V33PR5 and MPAI-A4*xxx*E
2097-V34PR5 and MPAI-B4*xxx*E

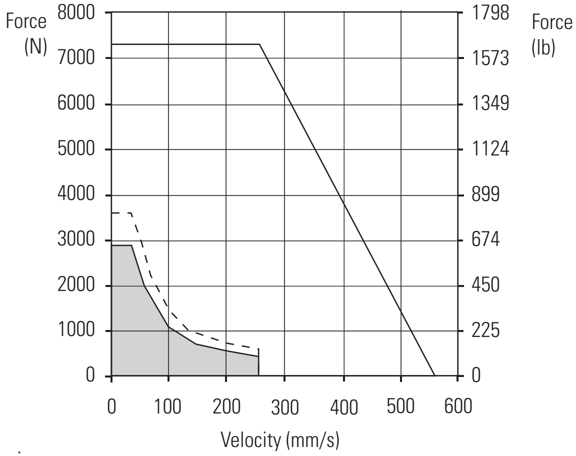
☐ = Intermittent operating region
- - - = Continuous operating region @ 25 °C (77 °F)
▨ = Continuous operating region @ 40 °C (104 °F)

Kinetix 300 Drives/MP-Series Heavy Duty (roller screw) Electric Cylinder Curves


2097-V33PR5 and MPAI-A3*xxx*R
2097-V34PR5 and MPAI-B3*xxx*R


2097-V33PR3 and MPAI-A3*xxx*S
2097-V34PR3 and MPAI-B3*xxx*S


2097-V33PR5 and MPAI-A4*xxx*R
2097-V34PR5 and MPAI-B4*xxx*R


2097-V33PR5 and MPAI-A4*xxx*S
2097-V34PR5 and MPAI-B4*xxx*S

☐ = Intermittent operating region
⬚ = Continuous operating region @ 25 °C (77 °F)
▨ = Continuous operating region @ 40 °C (104 °F)

Kinetix 300 (240V) Drives with TL-Series Electric Cylinders

This section provides system combination information for the Kinetix 300 drives when matched with TL-Series electric cylinders. Included are power/brake and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Electric Cylinder Cable Combinations

Electric Cylinder	Motor Power/Brake Cable	Motor Feedback Cable [1]
TLAR-A1*xxx*B TLAR-A1*xxx*E	2090-CPWM6DF-16AA*xx* (standard) (without brake) 2090-CPBM6DF-16AA*xx* (standard) (with brake)	2090-CFBM6DF-CBAA*xx* (standard) Absolute High-resolution Feedback
TLAR-A2*xxx*C TLAR-A2*xxx*F		
TLAR-A3*xxx*E TLAR-A3*xxx*H		

(1) The TLY-A*xxxx*-B motors with 17-bit high-resolution encoder feedback (mounted to the electric cylinder) require the 2090-CFBM6DF-CBAA*xx* flying-lead feedback cable and 2090-K2CK-D15M connector kit with 2090-DA-BAT2 battery. Refer to Breakout Components and Connector Kits beginning on page 440 for more information.

Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Electric Cylinder (non-brake) Performance Specifications with Kinetix 300 Drives

Electric Cylinder	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Rated Output** kW	**Kinetix 300 230V Drives**
TLAR-A1*xxx*B	150	1.36	240 (53.9)	1.79	300 (67.4)	0.036	2097-V33PR1
TLAR-A1*xxx*E	500	2.59	280 (62.9)	3.03	350 (78.7)	0.140	
TLAR-A2*xxx*C	250	3.03	420 (94.4)	3.41	525 (118)	0.105	
TLAR-A2*xxx*F	640	5.50	640 (144)	7.25	800 (180)	0.350	
TLAR-A3*xxx*E	500	10.0	2000 (450)	12.9	2500 (562)	0.930	2097-V33PR5
TLAR-A3*xxx*H	1000		1300 (292)	17.2	1625 (365)		

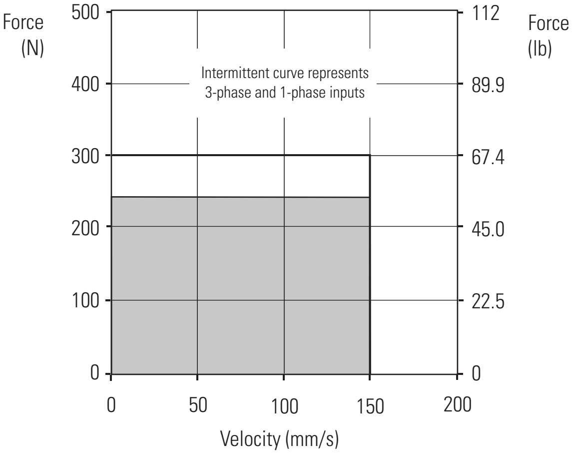
Electric Cylinder (brake) Performance Specifications with Kinetix 300 Drives

Electric Cylinder	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Rated Output** kW	**Kinetix 300 230V Drives**
TLAR-A1*xxx*B	150	1.18	240 (53.9)	1.79	300 (67.4)	0.036	2097-V33PR1
TLAR-A1*xxx*E	500	2.24	280 (62.9)	3.03	350 (78.7)	0.140	
TLAR-A2*xxx*C	250	2.68	420 (94.4)	3.41	525 (118)	0.105	
TLAR-A2*xxx*F	640	4.95	640 (144)	7.25	800 (180)	0.350	
TLAR-A3*xxx*E	500	10.0	2000 (450)	12.9	2500 (562)	0.930	2097-V33PR5
TLAR-A3*xxx*H	1000		1300 (292)	17.2	1625 (365)		

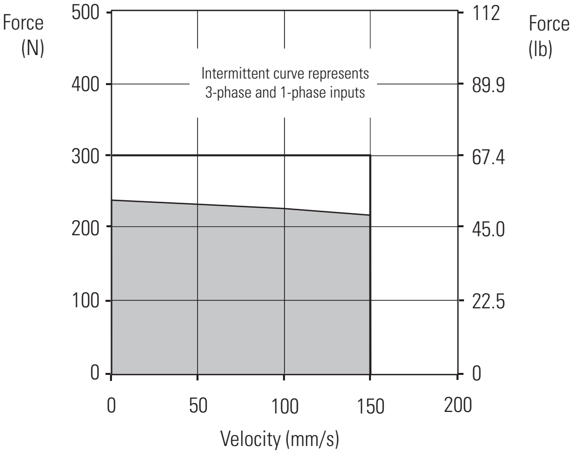
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 40 °C (104 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 300 (240V) Drives/TL-Series Electric Cylinder Curves

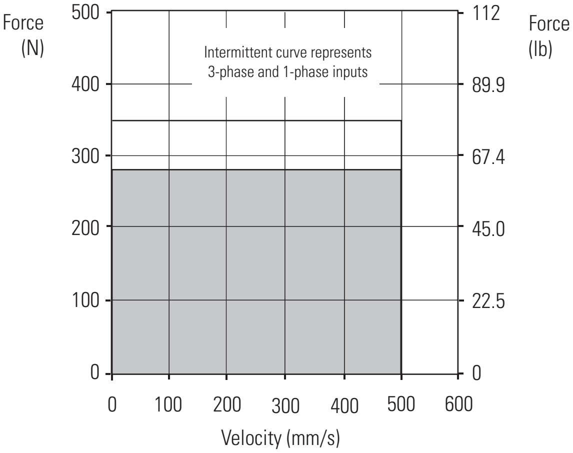
2097-V33PR1 and TLAR-A1*xxx*B-B2A



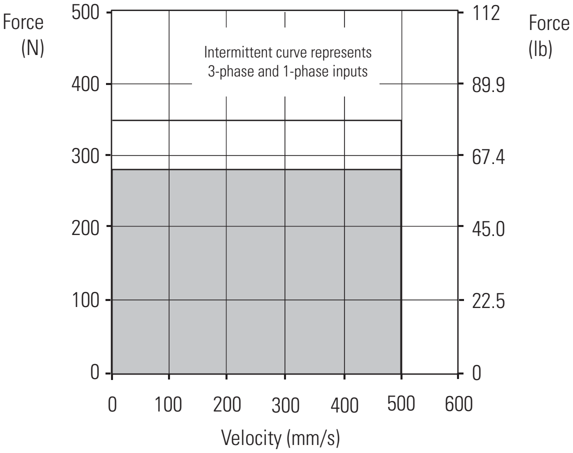
2097-V33PR1 and TLAR-A1*xxx*B-B4A (Brake)



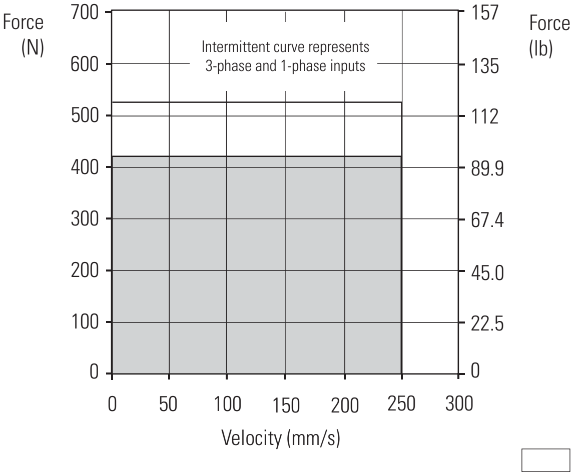
2097-V33PR1 and TLAR-A1*xxx*E-B2A



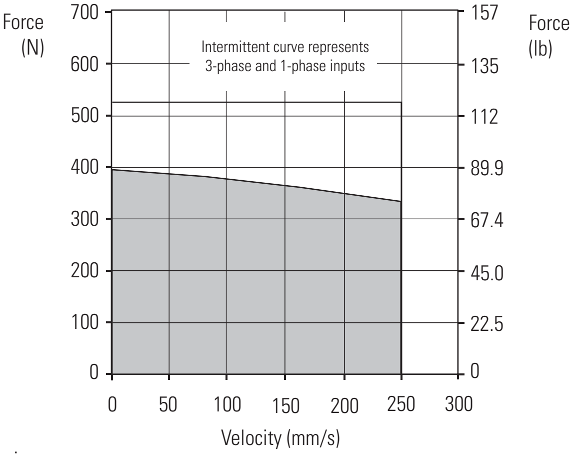
2097-V33PR1 and TLAR-A1*xxx*E-B4A (Brake)



2093-AMP5 and TLAR-A2*xxx*C-B2A



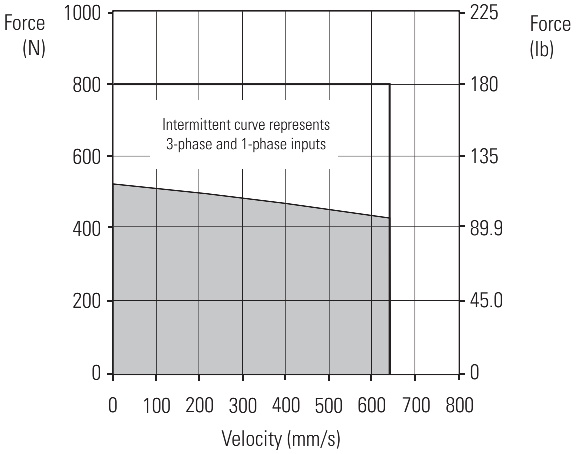
2097-V33PR1 and TLAR-A2*xxx*C-B4A (Brake)



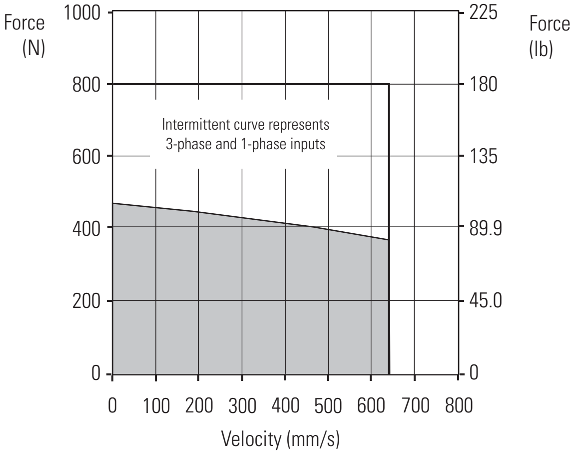
☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 300 (240V) Drives/TL-Series Electric Cylinder Curves, Continued

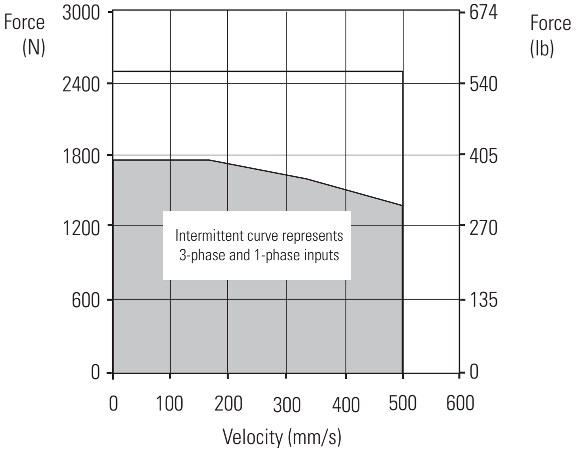
2097-V33PR1 and TLAR-A2*xxx*F-B2A



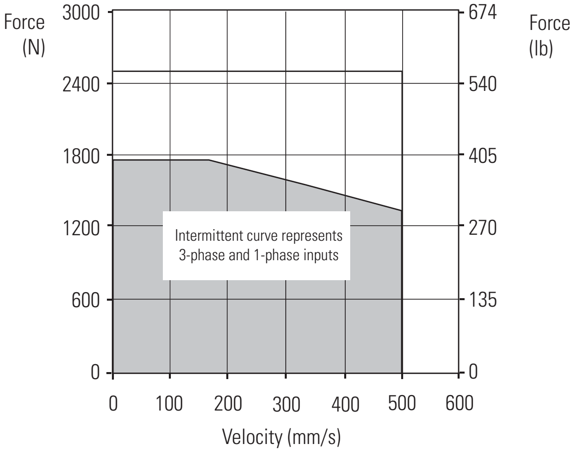
2097-V33PR1 and TLAR-A2*xxx*F-B4A (Brake)



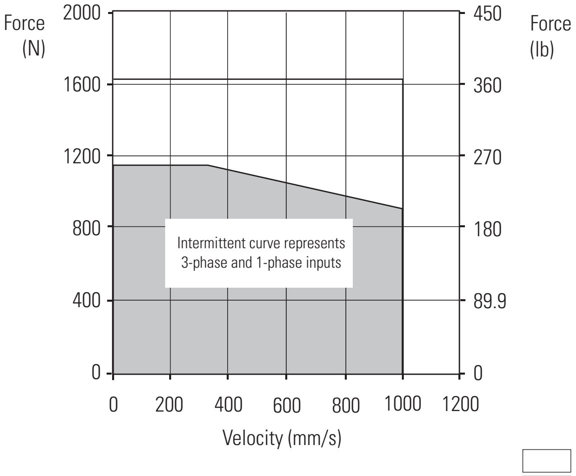
2097-V33PR5 and TLAR-A3*xxx*E-B2A



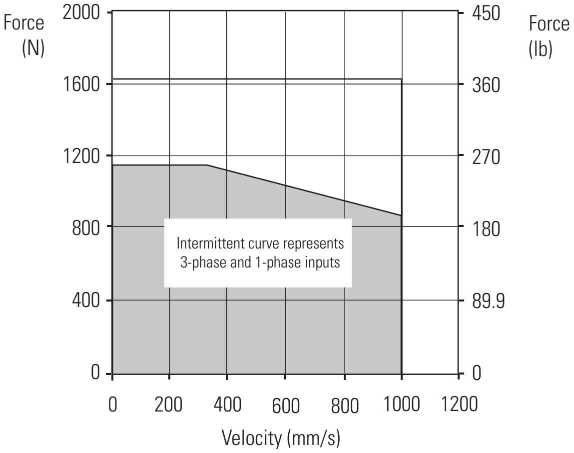
2097-V33PR5 and TLAR-A3*xxx*E-B4A (Brake)



2097-V33PR5 and TLAR-A3*xxx*H-B2A



2097-V33PR5 and TLAR-A3*xxx*H-B4A (Brake)



☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 3 (240V) Drives with MP-Series Integrated Linear Stages

This section provides system combination information for the Kinetix 3 (240V) drives when matched with MP-Series (230V) integrated direct-drive linear stages. Included are motor power and feedback cable catalog numbers, system performance specifications, and force/velocity curves.

Linear Stage Cable Combinations

Linear Stage	Motor Power/Brake Cable	Motor Feedback Cable (1)
MPAS-A6*xxx*B-ALM*x*2C, MPAS-A8*xxx*E-ALM*x*2C, MPAS-A9*xxx*K-ALM*x*2C	2090-XXNPMF-16S*xx* (2)	2090-XXNFMF-S*xx* (3) Incremental Feedback

(1) Use low-profile connector kit (catalog number 2090-K6CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

(2) These cables are available as standard (catalog number 2090-XXNPMF-16S*xx*) or continuous-flex (catalog number 2090-CP*x*M4DF-16AF*xx*).

(3) These cables are available as standard (catalog number 2090-XXNFMF-S*xx*) or continuous-flex (catalog number 2090-CFBM4DF-CDAF*xx*).

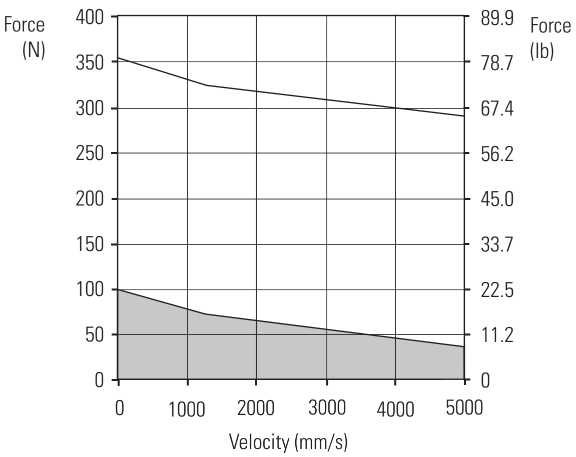
Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

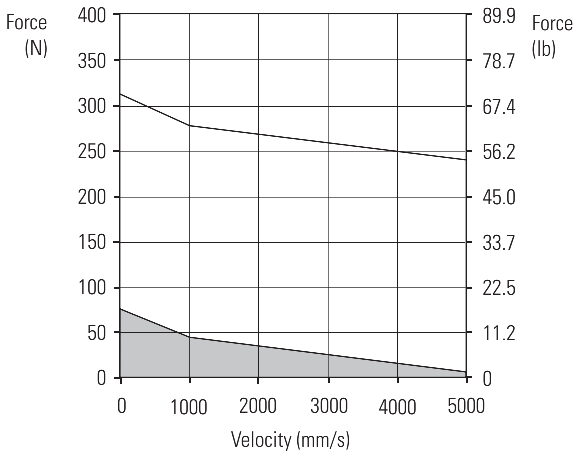
Linear Stage Performance Specifications with Kinetix 3 (240V) Drives

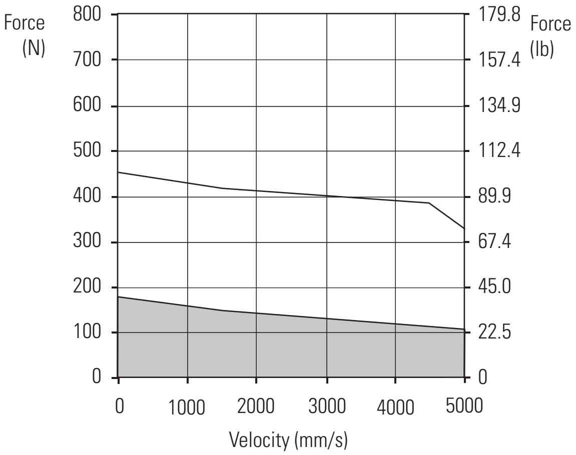
Linear Stage	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Stage Rated Output** kW	Kinetix 3 240V Drives
MPAS-A6*xxx*B-ALMO2C	5000 (200)	5.3	105 (23.6)	15.8	359 (80.7)	0.32	2071-AP8
MPAS-A6*xxx*B-ALMS2C		4.7	83.0 (18.7)	14.2	312 (70.1)	0.29	
MPAS-A8*xxx*E-ALMO2C		7.0	189 (42.5)	18.5	456 (103)	0.53	
MPAS-A8*xxx*E-ALMS2C		6.3	159 (35.7)	16.7	399 (89.7)	0.48	
MPAS-A9*xxx*K-ALMO2C		6.7	285 (64.1)	18.3	680 (153)	0.77	
MPAS-A9*xxx*K-ALMS2C		6.1	245 (55.1)	16.5	601 (135)	0.69	

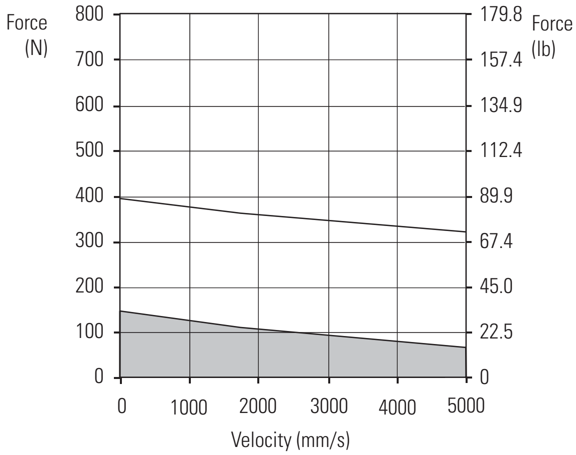
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 5.000 or later.

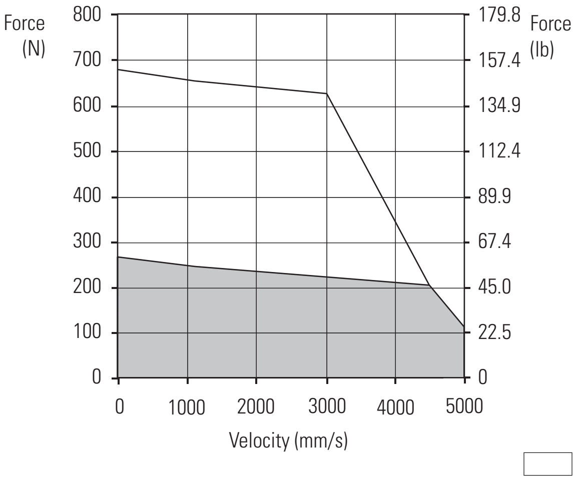
Kinetix 3 (240V) Drives/MP-Series Integrated Linear Stage Curves

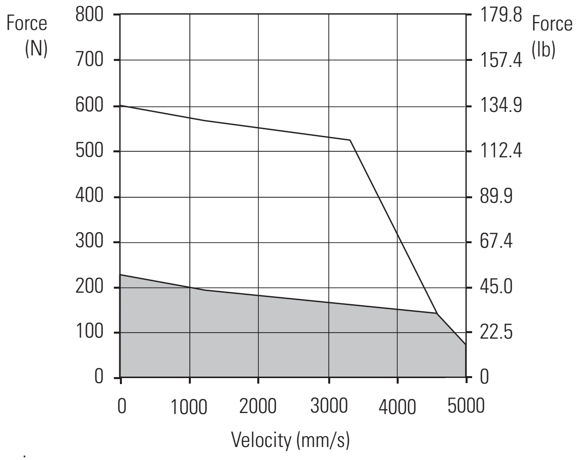

2071-AP8 and MPAS-A6*xxx*B-ALM02C


2071-AP8 and MPAS-A6*xxx*B-ALMS2C


2071-AP8 and MPAS-A8*xxx*E-ALM02C


2071-AP8 and MPAS-A8*xxx*E-ALMS2C


2071-AP8 and MPAS-A9*xxx*K-ALM02C


2071-AP8 and MPAS-A9*xxx*K-ALMS2C

☐ = Intermittent operating region
▨ = Continuous operating region
- - - - = System operation for specified stroke length

Kinetix 3 (240V) Drives with TL-Series Electric Cylinders

This section provides system combination information for the Kinetix 3 drives when matched with TL-Series (230V) electric cylinders. Included are power/brake and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Electric Cylinder Cable Combinations

Electric Cylinder	Motor Power/Brake Cable	Motor Feedback Cable [1]
TLAR-A1*xxx*B TLAR-A1*xxx*E	2090-CPWM6DF-16AA*xx* (standard) (without brake) 2090-CPBM6DF-16AA*xx* (standard) (with brake)	2090-CFBM6DF-CBAA*xx* (standard) Absolute High-resolution Feedback
TLAR-A2*xxx*C TLAR-A2*xxx*F		
TLAR-A3*xxx*E TLAR-A3*xxx*H		

(1) The TLY-A*xxxx*-B motors with 17-bit high-resolution encoder feedback (mounted to the electric cylinder) require 2090-CFBM6DF-CBAA*xx* flying-lead feedback cables and 2071-TBMF connector kit (with customer-supplied battery). Refer to Breakout Components and Connector Kits beginning on page 440 for more information.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Electric Cylinder (non-brake) Performance Specifications with Kinetix 3 Drives

Electric Cylinder	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Rated Output** kW	**Kinetix 3 240V Drives**
TLAR-A1*xxx*B	150	1.36	240 (53.9)	1.79	300 (67.4)	0.036	2071-AP0
TLAR-A1*xxx*E	500	2.59	280 (62.9)	3.03	350 (78.7)	0.140	2071-AP2
TLAR-A2*xxx*C	250	3.03	420 (94.4)	3.41	525 (118)	0.105	2071-AP2
TLAR-A2*xxx*F	640	5.50	640 (144)	7.25	800 (180)	0.350	2071-AP4
TLAR-A3*xxx*E	500	10.0	2000 (450)	12.9	2500 (562)	0.930	2071-A10
TLAR-A3*xxx*H	1000		1300 (292)	17.2	1625 (365)		2071-A15

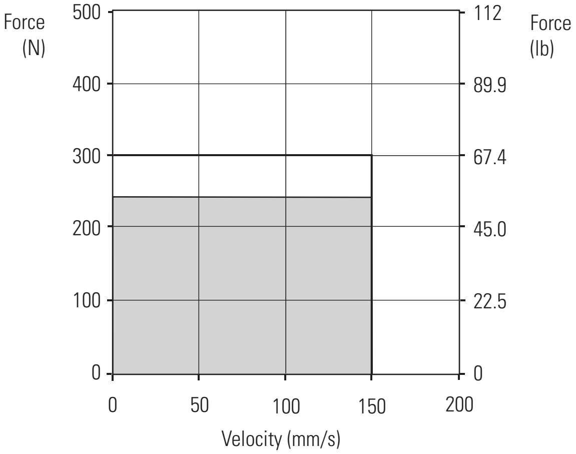
Electric Cylinder (brake) Performance Specifications with Kinetix 3 Drives

Electric Cylinder	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Rated Output** kW	**Kinetix 3 240V Drives**
TLAR-A1*xxx*B	150	1.18	240 (53.9)	1.79	300 (67.4)	0.036	2071-AP0
TLAR-A1*xxx*E	500	2.24	280 (62.9)	3.03	350 (78.7)	0.140	2071-AP2
TLAR-A2*xxx*C	250	2.68	420 (94.4)	3.41	525 (118)	0.105	2071-AP2
TLAR-A2*xxx*F	640	4.95	640 (144)	7.25	800 (180)	0.350	2071-AP4
TLAR-A3*xxx*E	500	10.0	2000 (450)	12.9	2500 (562)	0.930	2071-A10
TLAR-A3*xxx*H	1000		1300 (292)	17.2	1625 (365)		2071-A15

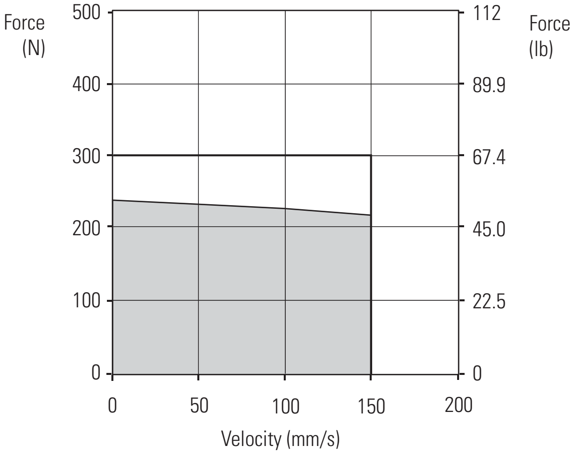
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 5.000 or later.

Kinetix 3 (240V) Drives/TL-Series Electric Cylinder Curves

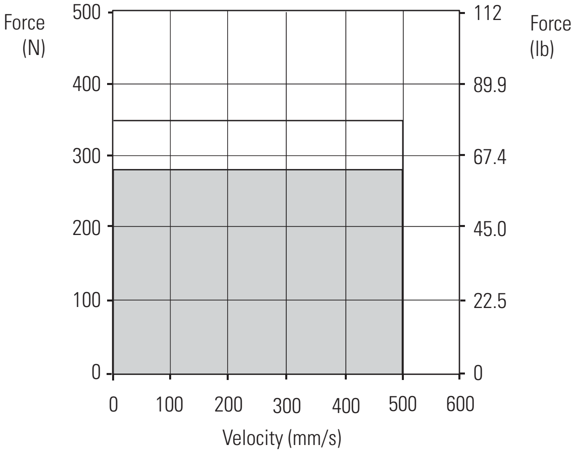
2071-AP0 and TLAR-A1*xxx*B-B2A



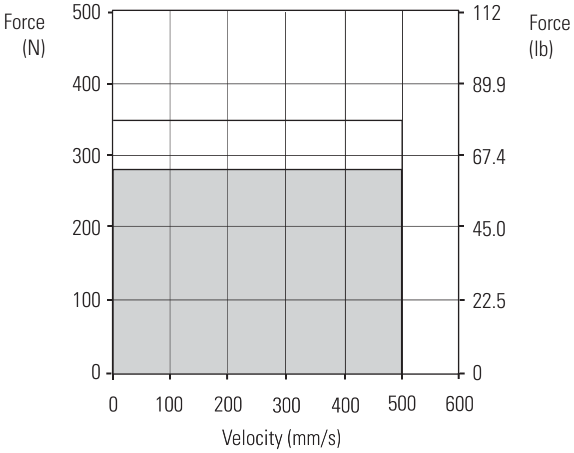
2071-AP0 and TLAR-A1*xxx*B-B4A (Brake)



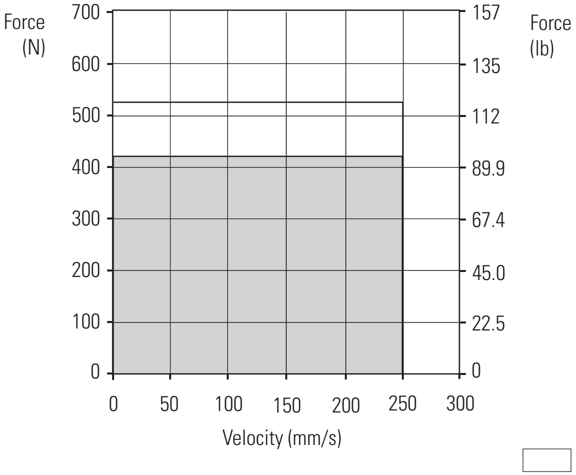
2071-AP2 and TLAR-A1*xxx*E-B2A



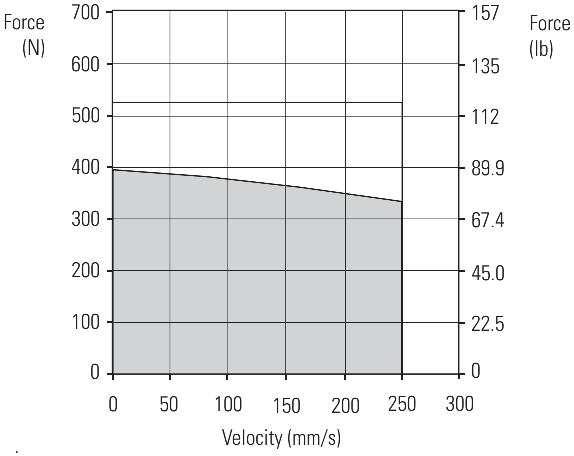
2071-AP2 and TLAR-A1*xxx*E-B4A (Brake)



2071-AP2 and TLAR-A2*xxx*C-B2A



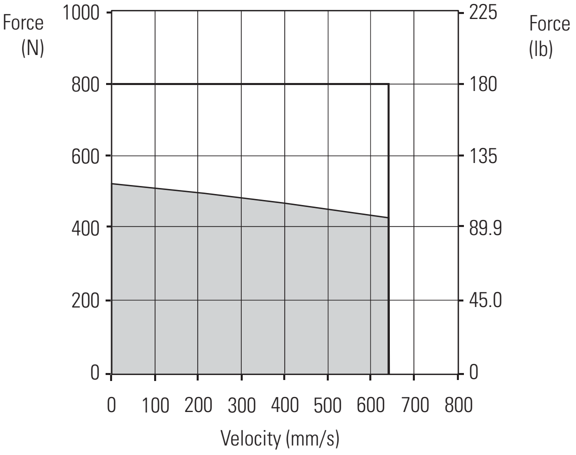
2071-AP2 and TLAR-A2*xxx*C-B4A (Brake)



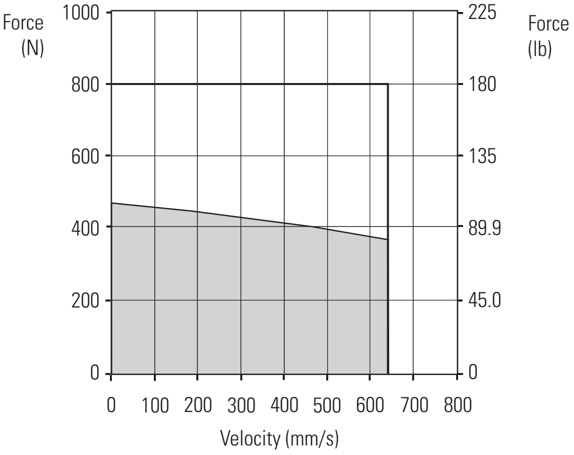
☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 3 (240V) Drives/TL-Series Electric Cylinder Curves, Continued

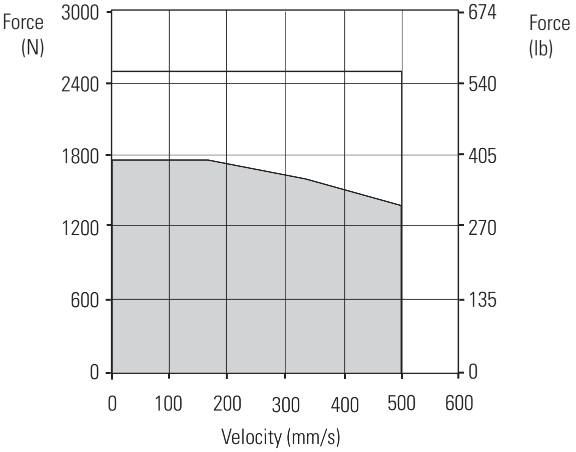
2071-AP4 and TLAR-A2*xxx*F-B2A



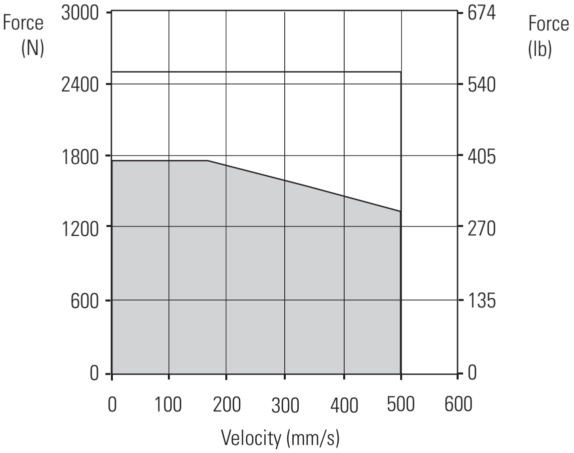
2071-AP4 and TLAR-A2*xxx*F-B4A (Brake)



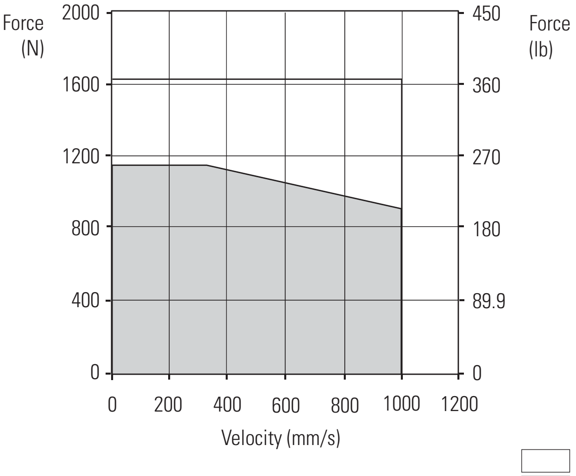
2071-A10 and TLAR-A3*xxx*E-B2A



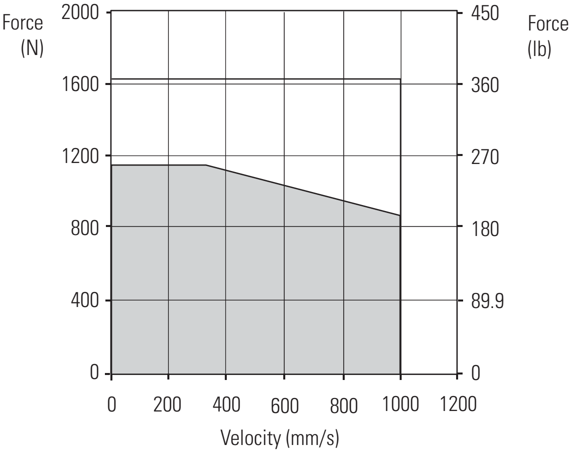
2071-A10 and TLAR-A3*xxx*E-B4A (Brake)



2071-A15 and TLAR-A3*xxx*H-B2A



2071-A15 and TLAR-A3*xxx*H-B4A (Brake)



☐ = Intermittent operating region
▨ = Continuous operating region

Kinetix 3 (240V) Drives with LDC-Series Linear Motors

This section provides system combination information for the Kinetix 3 drives when matched with LDC-Series iron-core linear motors. Included are power and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Linear Motor Cable Combinations

Linear Motor	Motor Power Cable	Motor Feedback Cable (1)
LDC-C030100-DHT, LDC-C030200-DHT, LDC-C030200-EHT	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-XXNFMF-S*xx* (standard) 2090-CFBM7DF-CDAF*xx* (continuous-flex) Sin/Cos or TTL Encoder Feedback
LDC-C050100-DHT, LDC-C050200-DHT, LDC-C050200-EHT, LDC-C050300-DHT, LDC-C050300-EHT		
LDC-C075200-DHT, LDC-C075200-EHT, LDC-C075300-DHT, LDC-C075300-EHT, LDC-C075400-DHT, LDC-C075400-EHT		
LDC-C100300-DHT, LDC-C100300-EHT, LDC-C100400-DHT, LDC-C100400-EHT, LDC-C100600-DHT		
LDC-C150400-DHT, LDC-C150600-DHT		

(1) Use low-profile connector kit (catalog number 2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

LDC-Series Performance Specifications with Kinetix 3 (240V) Drives

Linear Motor	Maximum Speed m/s (ft/s)	System Continuous Stall Current (1) Amps 0-pk	System Continuous Stall Force (1) N (lb)	System Peak Stall Current Amps 0-pk	System Peak Stall Force N (lb)	Linear Motor Rated Output kW	Kinetix 3 240V Drives
LDC-C030100-DHT	10.0 (32.8)	4.1…6.1	74…111 (17…25)	12.1	188 (42)	0.37…0.55	2071-AP4
LDC-C030200-DHT		8.1…12.2	148…222 (33…50)	24.3	375 (84)	0.74…1.11	2071-A10
LDC-C030200-EHT		4.1…6.1		12.1			2071-AP4
LDC-C050100-DHT	10.0 (32.8)	3.9…5.9	119…179 (27…40)	11.7	302 (68)	0.59…0.89	2071-AP4
LDC-C050200-DHT		7.9…11.8	240…359 (54…81)	23.3	600 (135)	1.20…1.79	2071-A10
LDC-C050200-EHT		3.9…5.9		11.6			2071-AP4
LDC-C050300-DHT		11.8…17.7	363…544 (82…122)	35.9	941 (212)	1.81…2.72	2071-A15
LDC-C050300-EHT		3.9…5.9		12.0			2071-AP4

(1) Values represent the range between no cooling (low value) and water cooling (high value).

Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 5.000 or later.

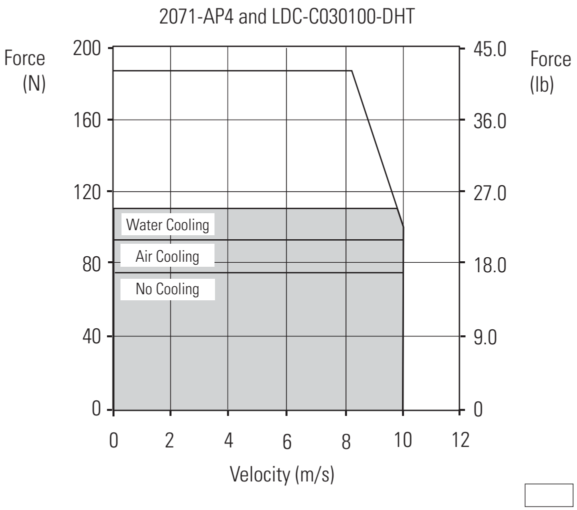
LDC-Series Performance Specifications with Kinetix 3 (240V) Drives, Continued

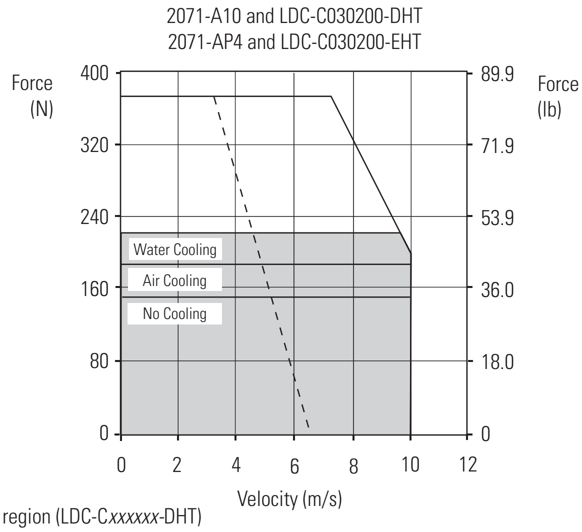
Linear Motor	**Maximum Speed** m/s (ft/s)	**System Continuous Stall Current** (1) Amps 0-pk	**System Continuous Stall Force** (1) N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Motor Rated Output** kW	**Kinetix 3 240V Drives**
LDC-C075200-DHT	10.0 (32.8)	7.7…11.5	348…523 (78…117)	22.9	882 (198)	1.74…2.61	2071-A10
LDC-C075200-EHT		3.8…5.7		11.5			2071-AP4
LDC-C075300-DHT		11.5…17.2	523…784 (117…176)	35.6	1368 (308)	2.61…3.92	2071-A15
LDC-C075300-EHT		3.8…5.7		11.9			2071-AP4
LDC-C075400-DHT		15.3…23.0	697…1045 (157…235)	47.4	1824 (410)	3.48…5.22	2071-A15
LDC-C075400-EHT		7.7…11.5		23.7			2071-A10
LDC-C100300-DHT	10.0 (32.8)	11.1…16.7	674…1012 (152…227)	34.3	1767 (397)	3.37…5.06	2071-A15
LDC-C100300-EHT		3.7…5.6		11.4			2071-AP4
LDC-C100400-DHT		14.8…22.2	899…1349 (202…303)	45.7	2356 (530)	4.49…6.74	2071-A15
LDC-C100400-EHT		7.4…11.1		22.8			2071-A10
LDC-C100600-DHT		22.2…33.3	1349…2023 (303…455)	68.5	3534 (794)	6.74…10.11	2071-A15
LDC-C150400-DHT	10.0 (32.8)	14.1…21.1	1281…1922 (288…432)	45.2	3498 (786)	6.40…9.61	2071-A10
LDC-C150400-EHT							2071-A15
LDC-C150600-DHT		21.1…31.7	1922…2882 (432…648)	67.8	5246 (1179)	9.61…14.41	2071-A15

(1) Values represent the range between no cooling (low value) and water cooling (high value).

Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 5.000 or later.

Kinetix 3 (240V) Drives/LDC-Series Linear Motor Curves

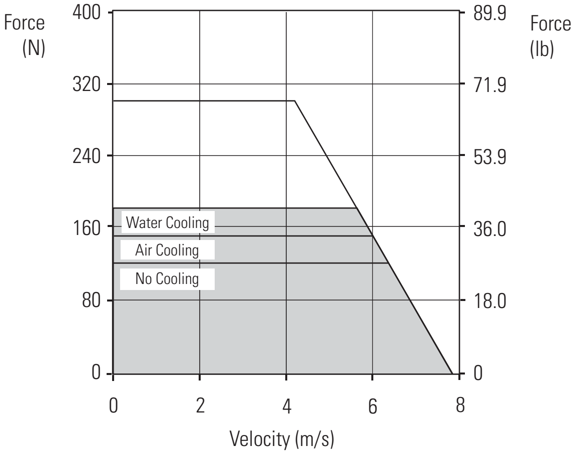

2071-AP4 and LDC-C030100-DHT


2071-A10 and LDC-C030200-DHT
2071-AP4 and LDC-C030200-EHT

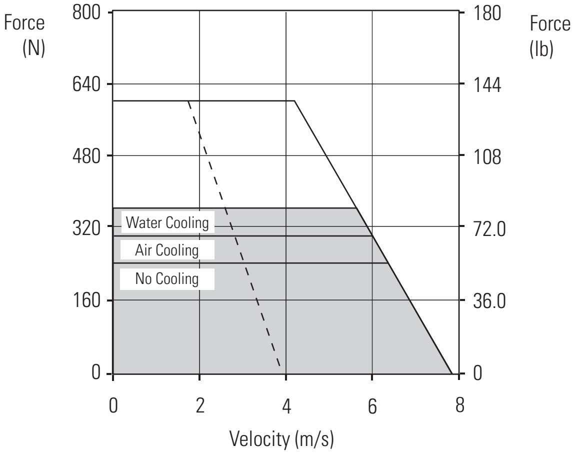
☐ = Intermittent operating region (LDC-C*xxxxxx*-DHT)
- - - = Intermittent operating region (LDC-C*xxxxxx*-EHT)
▨ = Continuous operating region

Kinetix 3 (240V) Drives/LDC-Series Linear Motor Curves, Continued

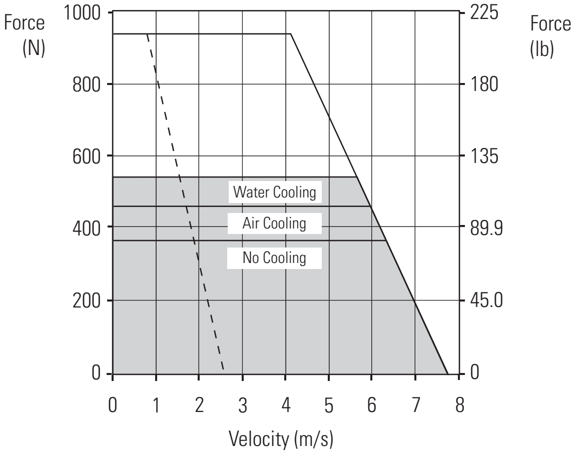
2071-AP4 and LDC-C050100-DHT



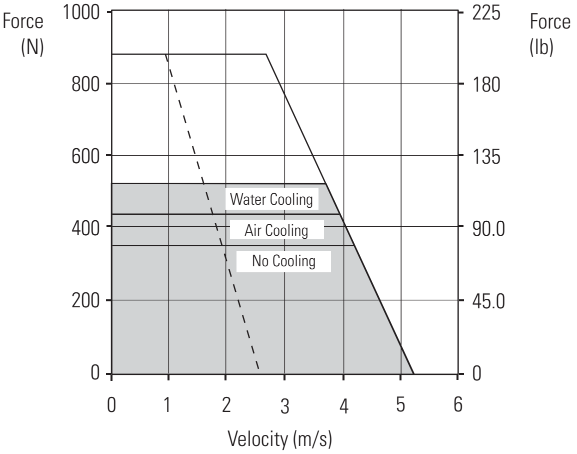
2071-A10 and LDC-C050200-DHT
2071-AP4 and LDC-C050200-EHT



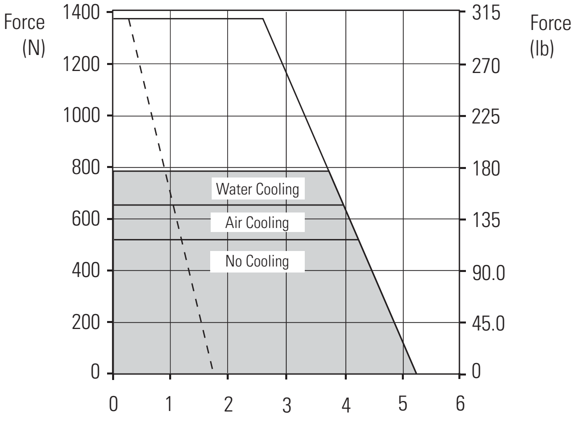
2071-A15 and LDC-C050300-DHT
2071-AP4 and LDC-C050300-EHT



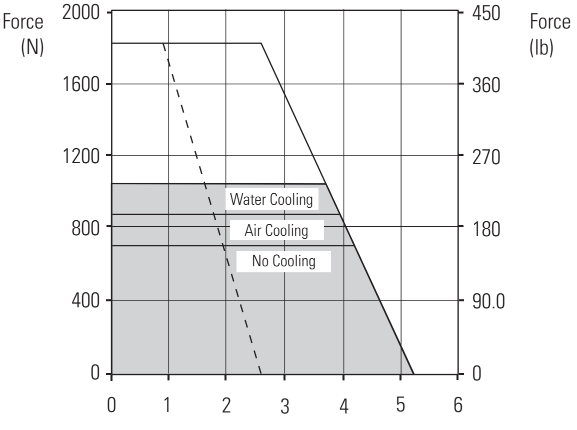
2071-A10 and LDC-C075200-DHT
2071-AP4 and LDC-C075200-EHT



2071-A15 and LDC-C075300-DHT
2071-AP4 and LDC-C075300-EHT



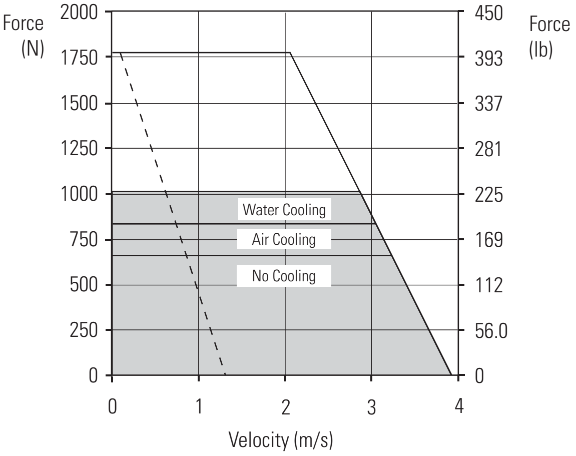
2071-A15 and LDC-C075400-DHT
2071-A10 and LDC-C075400-EHT



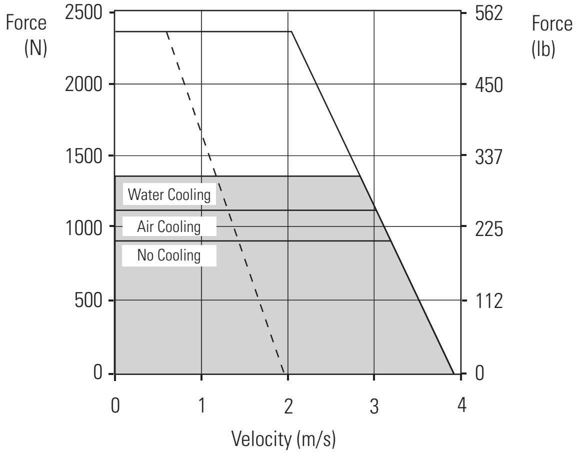
☐ = Intermittent operating region (LDC-C*xxxxxx*-DHT)
- - - = Intermittent operating region (LDC-C*xxxxxx*-EHT)
▨ = Continuous operating region

Kinetix 3 (240V) Drives/LDC-Series Linear Motor Curves, Continued

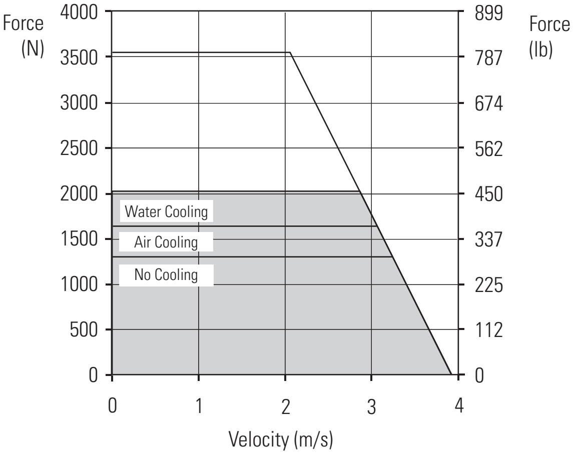
2071-A15 and LDC-C100300-DHT
2071-AP4 and LDC-C100300-EHT



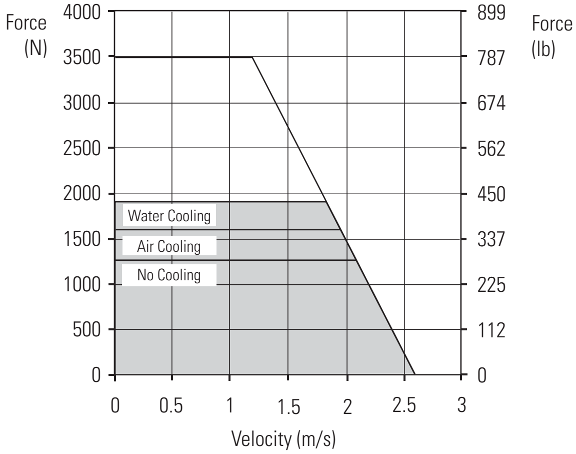
2071-A15 and LDC-C100400-DHT
2071-A10 and LDC-C100400-EHT



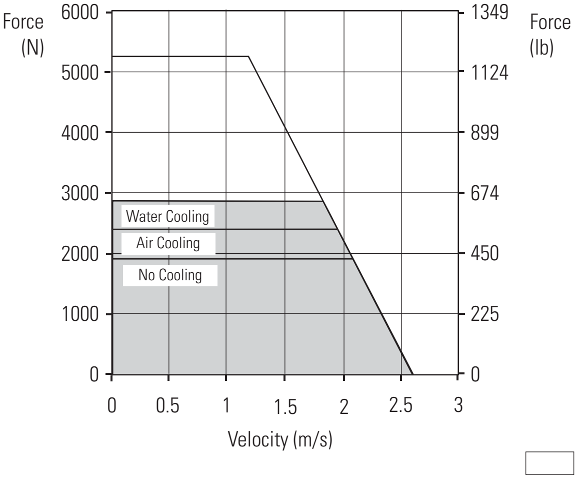
2071-A15 and LDC-C100600-EHT



2071-A15 and LDC-C150400-DHT
2071-A10 and LDC-C150400-EHT



2071-A15 and LDC-C150600-EHT



☐ = Intermittent operating region (LDC-C*xxxxxx*-DHT)
- - - = Intermittent operating region (LDC-C*xxxxxx*-EHT)
▨ = Continuous operating region

Kinetix 3 (240V) Drives with LDL-Series Linear Motors

This section provides system combination information for the Kinetix 3 drives when matched with LDL-Series ironless linear motors. Included are power and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Linear Motor Cable Combinations

Linear Motors	Motor Power Cable	Motor Feedback Cable [1]
LDL-N030120-DHT, LDL-N030240-DHT, LDL-N030240-EHT	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-XXNFMF-S*xx* (standard) 2090-CFBM7DF-CDAF*xx* (continuous-flex) Sin/Cos or TTL Encoder Feedback
LDL-N050120-DHT, LDL-N050240-DHT, LDL-N050240-EHT, LDL-N050360-DHT, LDL-N050360-EHT, LDL-N050480-DHT, LDL-N050480-EHT		
LDL-N075480-DHT, LDL-N075480-EHT		
LDL-T030120-DHT, LDL-T030240-DHT, LDL-T030240-EHT		
LDL-T050120-DHT, LDL-T050240-DHT, LDL-T050240-EHT, LDL-T050360-DHT, LDL-T050480-DHT, LDL-T050480-EHT		
LDL-T075480-EHT, LDL-T075480-EHT		

(1) Use low-profile connector kit (catalog number 2090-K2CK-D15M) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

LDL-Series Performance Specifications with Kinetix 3 (240V) Drives

Linear Motor	Maximum Speed m/s (ft/s)	System Continuous Stall Current Amps 0-pk	System Continuous Stall Force N (lb)	System Peak Stall Current Amps 0-pk	System Peak Stall Force N (lb)	Linear Motor Rated Output kW	Kinetix 3 240V Drives
LDL-N030120-DHT	10.0 (32.8)	3.0	63 (14)	9.9	209 (47)	0.31	2071-AP4
LDL-N030240-DHT		6.0	126 (28)	19.9	417 (94)	0.63	2071-AP8
LDL-N030240-EHT		3.0		9.9			2071-AP4
LDL-T030120-DHT		3.0	72 (16)	9.9	239 (54)	0.36	2071-AP4
LDL-T030240-DHT		6.0	144 (32)	19.9	479 (108)	0.72	2071-AP8
LDL-T030240-EHT		3.0		9.9			2071-AP4

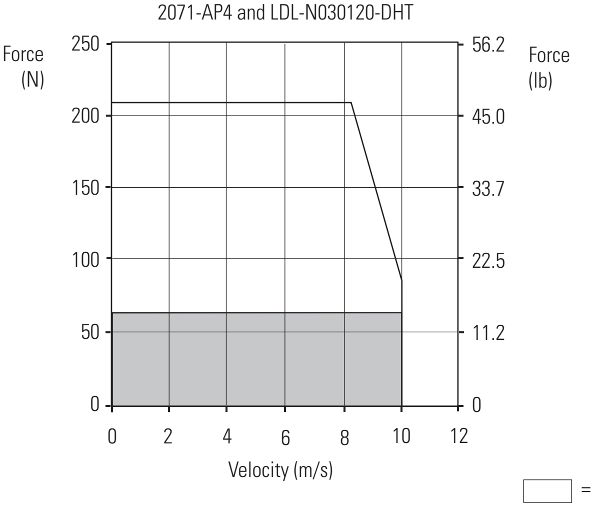
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 5.000 or later.

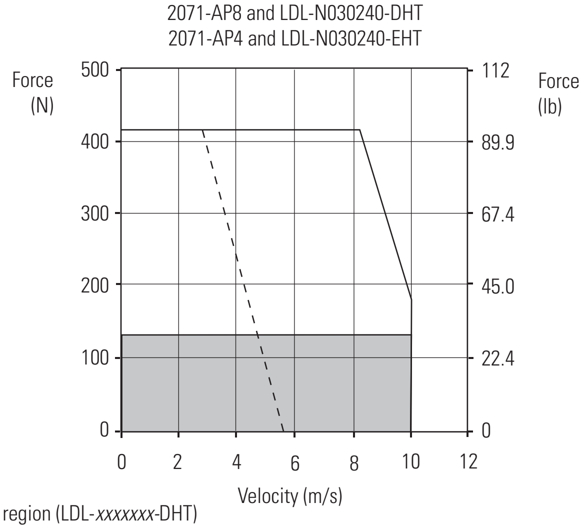
LDL-Series Performance Specifications with Kinetix 3 (240V) Drives, Continued

Linear Motor	**Maximum Speed** m/s (ft/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Motor Rated Output** kW	**Kinetix 3 240V Drives**
LDL-N050120-DHT	10.0 (32.8)	2.7	96 (22)	9.1	317 (71)	0.48	2071-AP4
LDL-N050240-DHT		5.5	191 (43)	18.1	635 (143)	0.95	2071-AP8
LDL-N050240-EHT		2.7		9.1			2071-AP4
LDL-N050360-DHT		8.2	287 (65)	27.2	952 (214)	1.43	2071-A10
LDL-N050360-EHT		2.7		9.1			2071-AP4
LDL-N050480-DHT		10.9	383 (86)	36.3	1269 (285)	1.91	2071-A15
LDL-N050480-EHT		5.5		18.1			2071-AP8
LDL-T050120-DHT		2.7	110 (25)	9.1	364 (82)	0.55	2071-AP4
LDL-T050240-DHT		5.5	220 (49)	18.1	728 (164)	1.10	2071-AP8
LDL-T050240-EHT		2.7		9.1			2071-AP4
LDL-T050360-DHT		8.2	329 (74)	27.2	1093 (246)	1.64	2071-A10
LDL-T050480-DHT		10.9	439 (99)	36.3	1457 (327)	2.19	2071-A15
LDL-T050480-EHT		5.5		18.1			2071-AP8
LDL-N075480-DHT	10.0 (32.8)	9.9	519 (117)	32.8	1723 (387)	2.59	2071-A15
LDL-N075480-EHT		4.9		16.4			2071-AP8
LDL-T075480-DHT		9.9	596 (134)	32.8	1977 (444)	2.98	2071-A15
LDL-T075480-EHT		4.9		16.4			2071-AP8

Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Kinetix 3 (240V) Drives/LDL-Series Linear Motor Curves

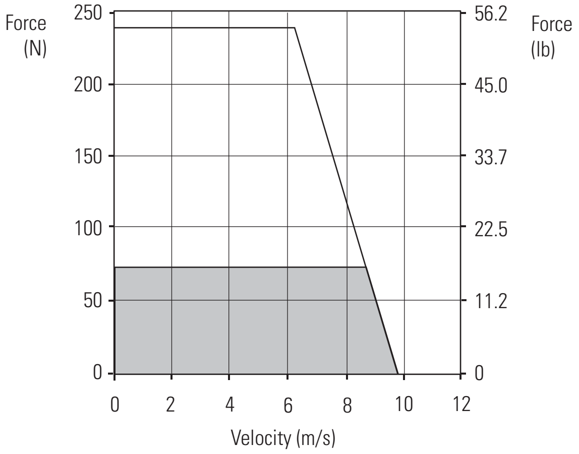




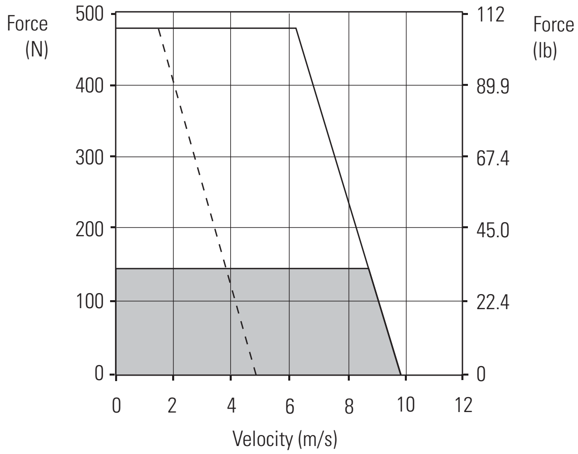
☐ = Intermittent operating region (LDL-*xxxxxxx*-DHT)
– – – = Intermittent operating region (LDL-*xxxxxxx*-EHT)
▨ = Continuous operating region

Kinetix 3 (240V) Drives/LDL-Series Linear Motor Curves, Continued

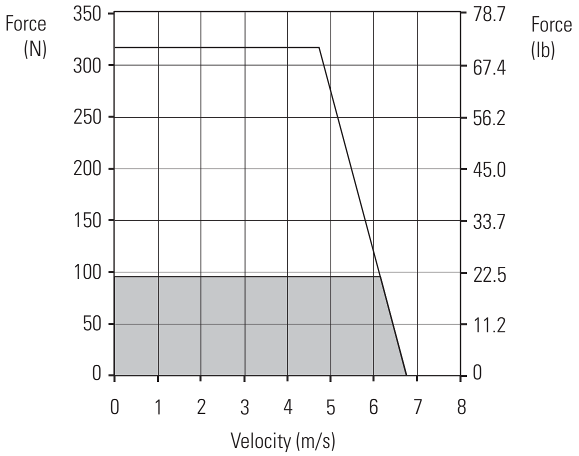
2071-AP4 and LDL-T030120-DHT



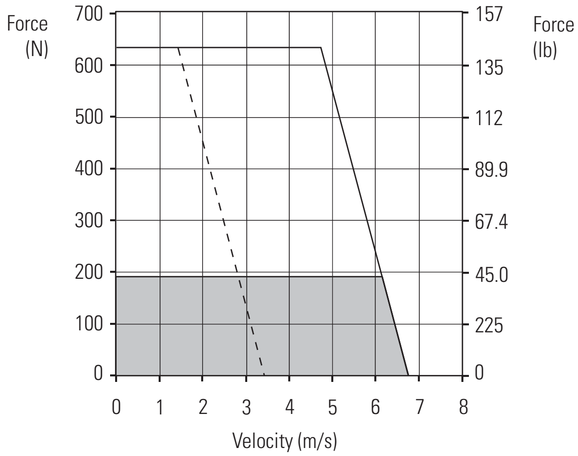
2071-AP8 and LDL-T030240-DHT
2071-AP4 and LDL-T030240-EHT



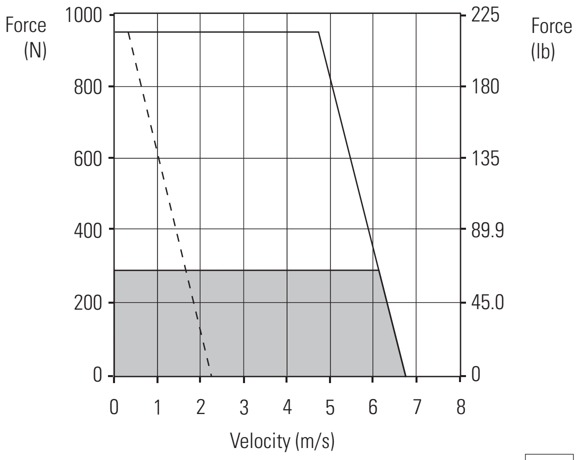
2071-AP4 and LDL-N050120-DHT



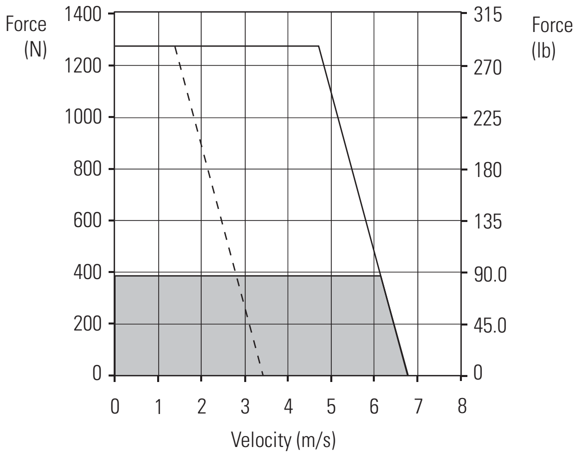
2071-AP8 and LDL-N050240-DHT,
2071-AP4 and LDL-N050240-EHT



2071-A10 and LDL-N050360-DHT
2071-AP4 and LDL-N050360-EHT



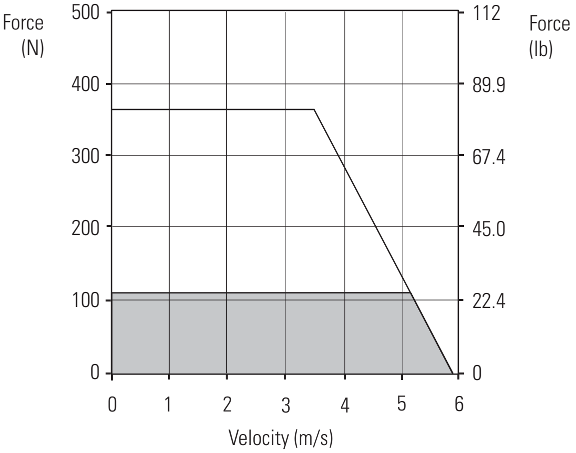
2071-A15 and LDL-N050480-DHT
2071-AP8 and LDL-N050480-EHT



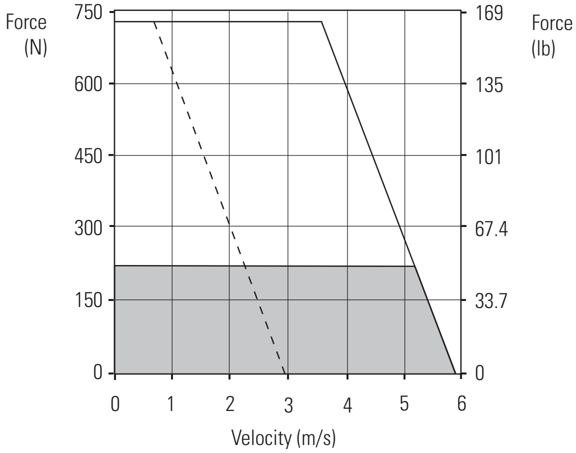
☐ = Intermittent operating region (LDL-*xxxxxxx*-DHT)
- - - = Intermittent operating region (LDL-*xxxxxxx*-EHT)
▨ = Continuous operating region

Kinetix 3 (240V) Drives/LDL-Series Linear Motor Curves, Continued

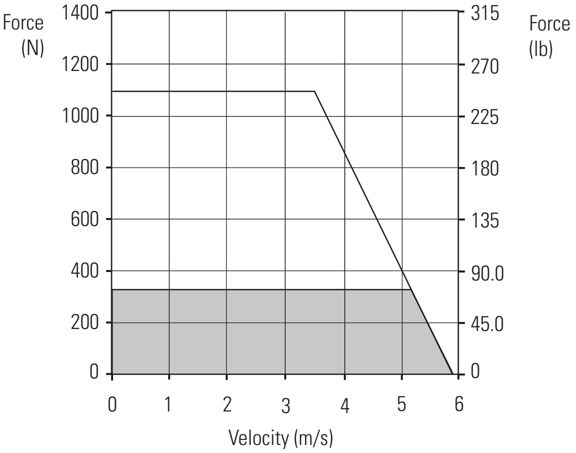
2071-AP4 and LDL-T050120-DHT



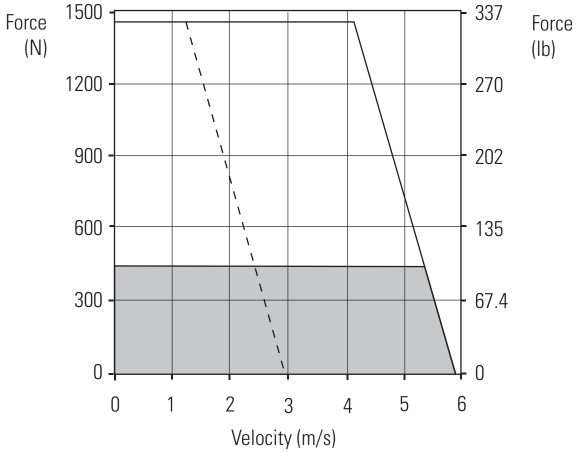
2071-AP8 and LDL-T050240-DHT
2071-AP4 and LDL-T050240-EHT



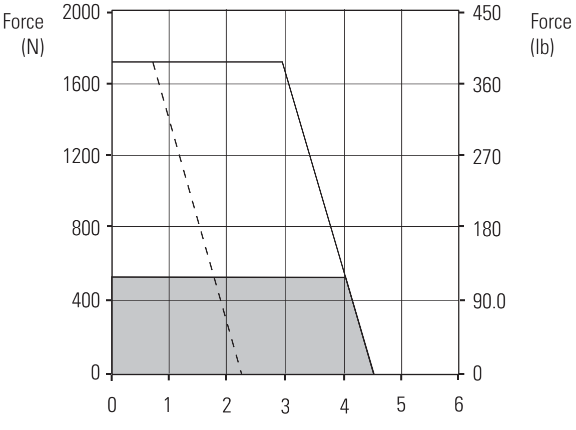
2071-A10 and LDL-T050360-DHT



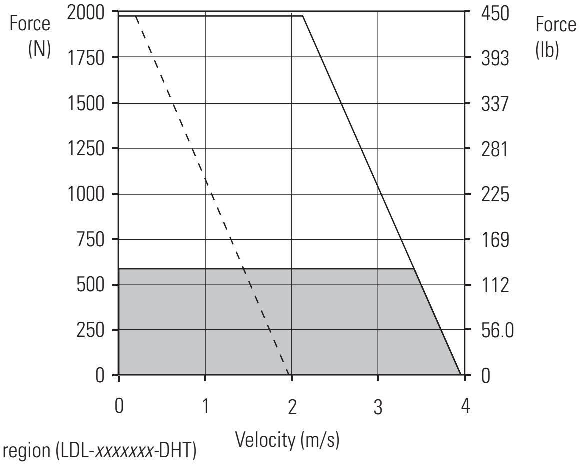
2071-A15 and LDL-T050480-DHT
2071-A08 and LDL-T050480-EHT



2071-A15 and LDL-N075480-DHT
2071-AP8 and LDL-N075480-EHT



2071-A15 and LDL-T075480-DHT
2071-A08 and LDL-T075480-EHT



☐ = Intermittent operating region (LDL-*xxxxxxx*-DHT)
- - - = Intermittent operating region (LDL-*xxxxxxx*-EHT)
▨ = Continuous operating region

Ultra3000 (230V) Drives with MP-Series Integrated Linear Stages

This section provides system combination information for the Ultra3000 (230V) drives when matched with MP-Series (230V) integrated direct-drive or ballscrew linear stages. Included are power/brake and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Linear Stage Cable Combinations

Linear Stage	Motor Power/Brake Cable	Motor Feedback Cable (1)
MPAS-A*xxxx*1-V05S*x*A, MPAS-A*xxxx*2-V20S*x*A	2090-XXNPMF-16S*xx* (2)	2090-XXNFMF-S*xx* (3) Absolute High-resolution Feedback
MPAS-A6*xxx*B-ALM*x*2C, MPAS-A8*xxx*E-ALM*x*2C, MPAS-A9*xxx*K-ALM*x*2C		2090-XXNFMF-S*xx* (3) Incremental Feedback

(1) Use drive-mounted breakout board (catalog number 2090-UXBB-DM15) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

(2) These cables are available as standard (catalog number 2090-XXNPMF-16S*xx*) or continuous-flex (catalog number 2090-CP*x*M4DF-16AF*xx*).

(3) These cables are available as standard (catalog number 2090-XXNFMF-S*xx*) or continuous-flex (catalog number 2090-CFBM4DF-CDAF*xx*).

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Linear Stage Performance Specifications with Ultra3000 (230V) Drives

Linear Stage	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Stage Rated Output** kW	**Ultra3000 230V Drives**
MPAS-A*xxxx*1-V05SxA	200 (7.9) (1)	2.50	422 (94.9)	6.10	1212 (272)	0.37	2098-DSD-005
		3.09	521 (117)				2098-DSD-010
MPAS-A*xxxx*2-V20SxA	1124 (44.3) (2)	2.50	254 (57.1)	7.50	798 (179)	0.62	2098-DSD-005
		4.54	462 (104)	9.10	968 (218)		2098-DSD-010
MPAS-A6*xxx*B-ALMO2C	5000 (200)	5.0	97.8 (22.0)	15.0	340 (76.4)	0.32	2098-DSD-010
		5.3	105 (23.6)	15.8	359 (80.7)		2098-DSD-020
MPAS-A6*xxx*B-ALMS2C		2.5	29.6 (6.65)	7.5	150 (33.7)	0.29	2098-DSD-005
		4.7	83.0 (18.7)	14.2	312 (70.1)		2098-DSD-010
MPAS-A8*xxx*E-ALMO2C		5.0	129 (29.0)	15.0	366 (82.3)	0.53	2098-DSD-010
		7.0	189 (42.5)	18.5	456 (103)		2098-DSD-020
MPAS-A8*xxx*E-ALMS2C		5.0	120 (27.0)	15.0	356 (80.0)	0.48	2098-DSD-010
		6.3	159 (35.7)	16.7	399 (89.7)		2098-DSD-020
MPAS-A9*xxx*K-ALMO2C		5.0	207 (46.5)	15.0	553 (124)	0.77	2098-DSD-010
		6.7	285 (64.1)	18.3	680 (153)		2098-DSD-020
MPAS-A9*xxx*K-ALMS2C		5.0	195 (43.8)	15.0	545 (123)	0.69	2098-DSD-010
		6.1	245 (55.1)	16.5	601 (135)		2098-DSD-020

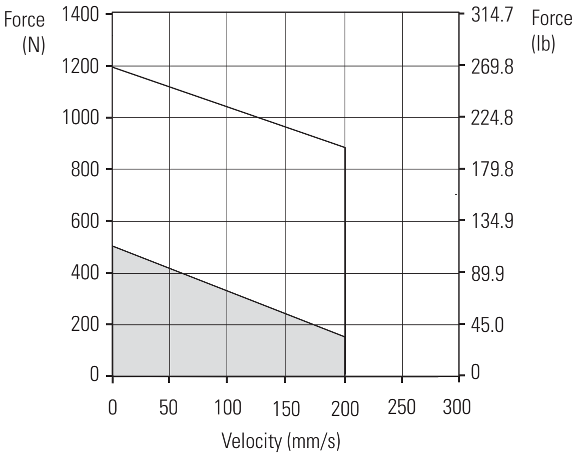
(1) For 900 mm stroke length, maximum speed is 176 mm/s (6.9 in/s). For 1020 mm stroke length, maximum speed is 143 mm/s (5.6 in/s).

(2) For 780 mm stroke length, maximum speed is 889 mm/s (35.0 in/s). For 900 mm stroke length, maximum speed is 715 mm/s (28.2 in/s). For 1020 mm stroke length, maximum speed is 582 mm/s (22.9 in/s).

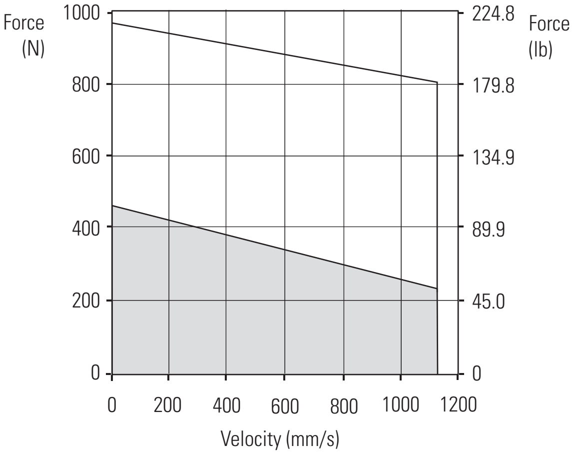
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Ultra3000 (230V) Drives/MP-Series Integrated Linear Stage Curves

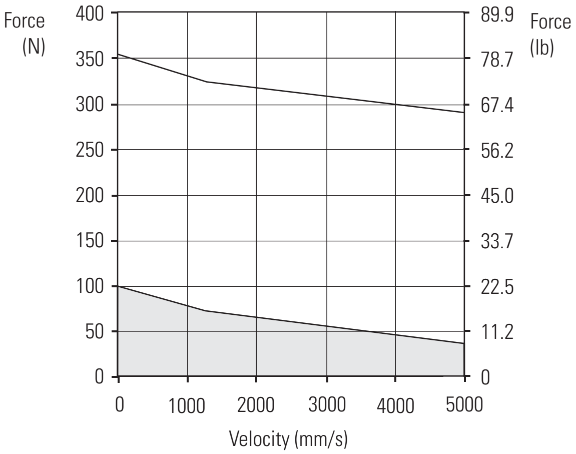
2098-DSD-010 and MPAS-A6*xxx*1-V05S*x*A



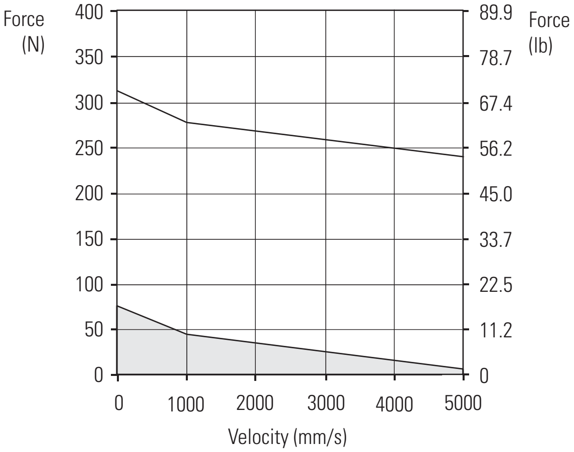
2098-DSD-010 and MPAS-A6*xxx*2-V20S*x*A



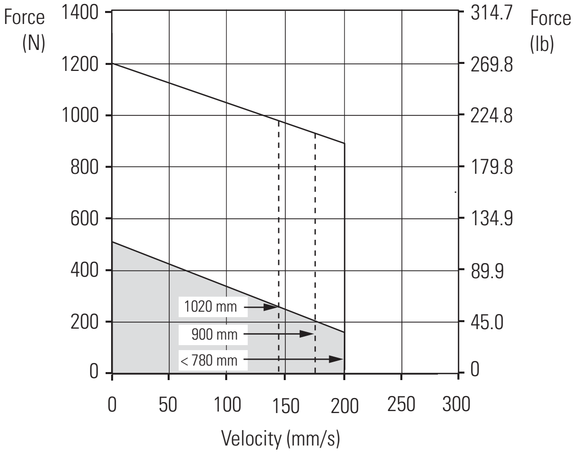
2098-DSD-020 and MPAS-A6*xxx*B-ALMO2C



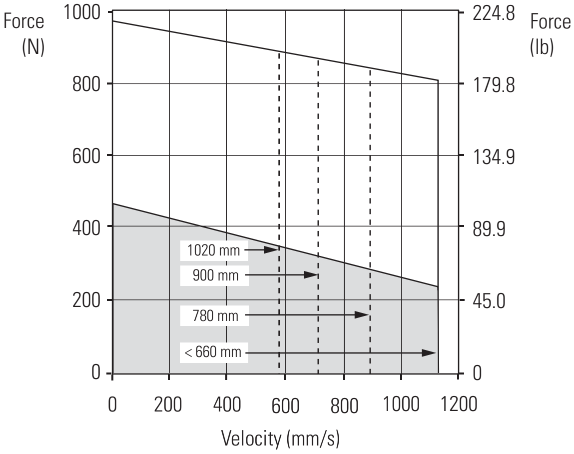
2098-DSD-010 and MPAS-A6*xxx*B-ALMS2C



2098-DSD-010 and MPAS-A8*xxx*1-V05S*x*A



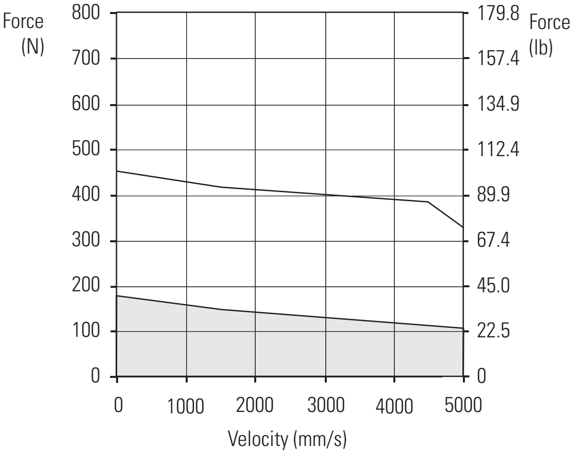
2098-DSD-010 and MPAS-A8*xxx*2-V20S*x*A



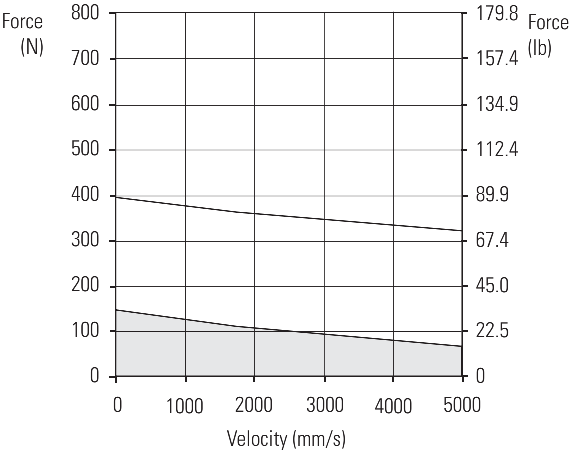
☐ = Intermittent operating region
▨ = Continuous operating region
- - - = System operation for specified stroke length

Ultra3000 (230V) Drives/MP-Series Integrated Linear Stage Curves, Continued

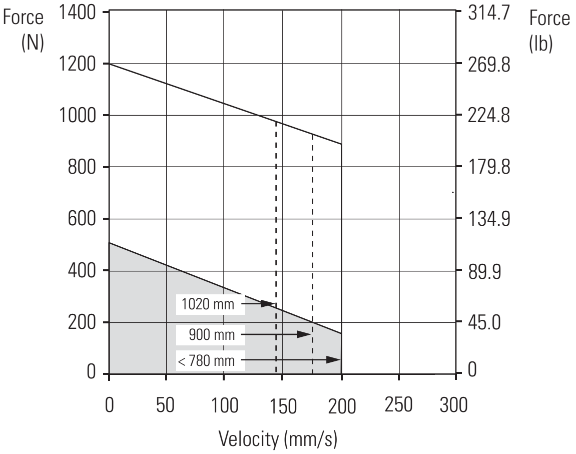
2098-DSD-020 and MPAS-A8*xxx*E-ALMO2C



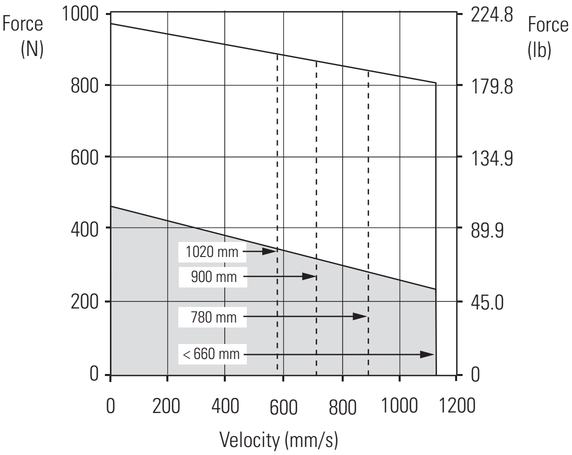
2098-DSD-020 and MPAS-A8*xxx*E-ALMS2C



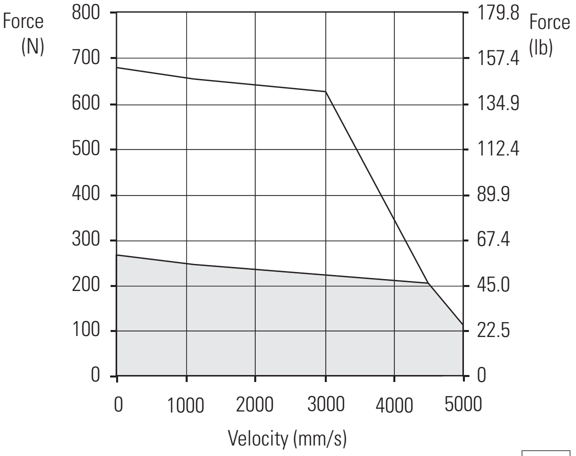
2098-DSD-010 and MPAS-A9*xxx*1-V05S*x*A



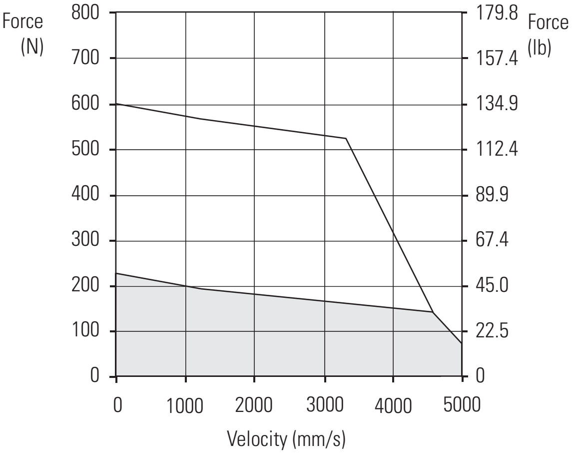
2098-DSD-010 and MPAS-A9*xxx*2-V20S*x*A



2098-DSD-020 and MPAS-A9*xxx*K-ALMO2C



2098-DSD-020 and MPAS-A9*xxx*K-ALMS2C



☐ = Intermittent operating region
▨ = Continuous operating region
- - - - = System operation for specified stroke length

Ultra3000 (460V) Drives with MP-Series Integrated Linear Stages

This section provides system combination information for the Ultra3000 (460V) drives when matched with MP-Series (460V) integrated direct-drive or ballscrew linear stages. Included are motor power and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Linear Stage Cable Combinations

Linear Stage	Motor Power/Brake Cable	Motor Feedback Cable [1]
MPAS-B*xxxx*1-V05S*x*A, MPAS-B*xxxx*2-V20S*x*A	2090-XXNPMF-16S*xx* [2]	2090-XXNFMF-S*xx* [3] Absolute High-resolution Feedback
MPAS-B8*xxx*F-ALM*x*2C, MPAS-B9*xxx*L-ALM*x*2C		2090-XXNFMF-S*xx* [1] Incremental Feedback

(1) Use drive-mounted breakout board (catalog number 2090-UXBB-DM15) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits beginning on page 440.

(2) These cables are available as standard (catalog number 2090-XXNPMF-16S*xx*) or continuous-flex (catalog number 2090-CP*x*M4DF-16AF*xx*).

(3) These cables are available as standard (catalog number 2090-XXNFMF-S*xx*) or continuous-flex (catalog number 2090-CFBM4DF-CDAF*xx*).

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Linear Stage Performance Specifications with Ultra3000 (460V) Drives

Linear Stage	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Stage Rated Output** kW	**Ultra3000 460V Drives**
MPAS-B*xxxx*1-V05S*x*A	200 (7.9) [1]	1.75	521 (117)	3.50	1212 (272)	0.138	2098-DSD-HV030
MPAS-B*xxxx*2-V20S*x*A	1124 (44.3) [2]	3.30	462 (104)	6.60	968 (218)	0.52	2098-DSD-HV030
MPAS-B8*xxx*F-ALMO2C	5000 (200)	3.50	189 (42.5)	9.30	456 (103)	0.527	2098-DSD-HV030
MPAS-B8*xxx*F-ALMS2C	5000 (200)	3.15	159 (35.7)	8.37	399 (89.7)	0.475	2098-DSD-HV030
MPAS-B9*xxx*L-ALMO2C	5000 (200)	3.40	285 (64.1)	9.10	680 (153)	0.768	2098-DSD-HV030
MPAS-B9*xxx*L-ALMS2C	5000 (200)	3.03	245 (55.1)	8.19	601 (135)	0.69	2098-DSD-HV030

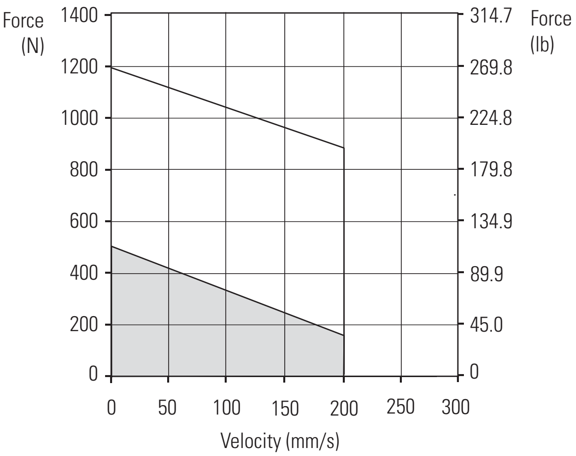
(1) For 900 mm stroke length, maximum speed is 176 mm/s (6.9 in/s). For 1020 mm stroke length, maximum speed is 143 mm/s (5.6 in/s).

(2) For 780 mm stroke length, maximum speed is 889 mm/s (35.0 in/s). For 900 mm stroke length, maximum speed is 715 mm/s (28.2 in/s). For 1020 mm stroke length, maximum speed is 582 mm/s (22.9 in/s).

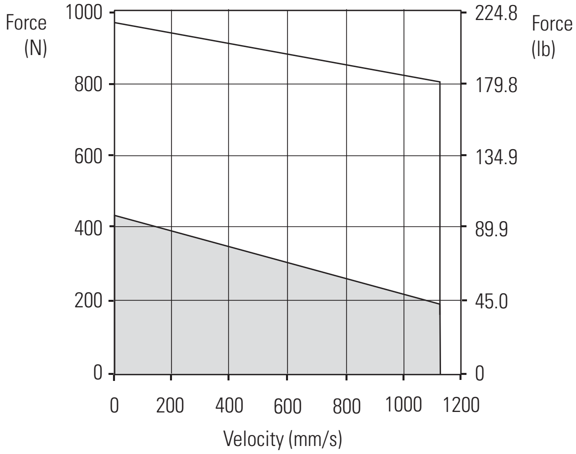
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Ultra3000 (460V) Drives/MP-Series Integrated Linear Stage Curves

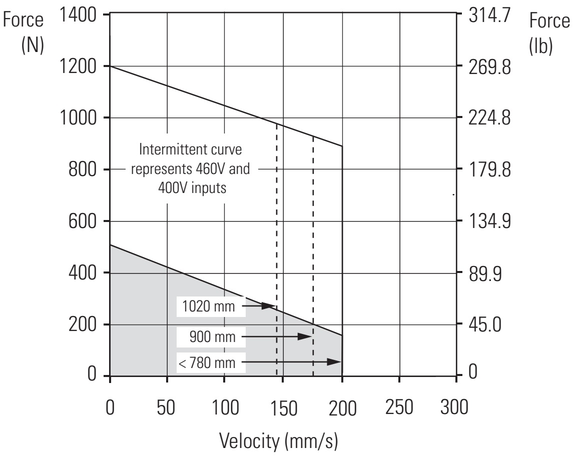
2098-DSD-HV030 and MPAS-B6*xxx*1-V05S*x*A



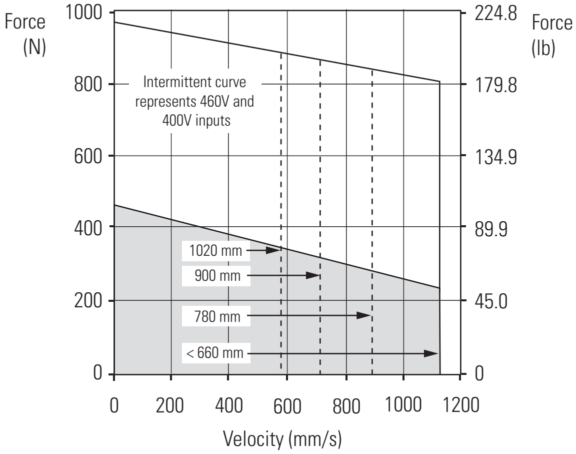
2098-DSD-HV030 and MPAS-B6*xxx*2-V20S*x*A



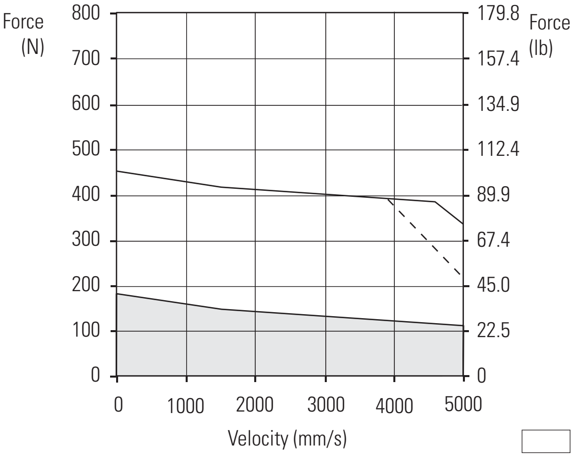
2098-DSD-HV030 and MPAS-B8*xxx*1-V05S*x*A



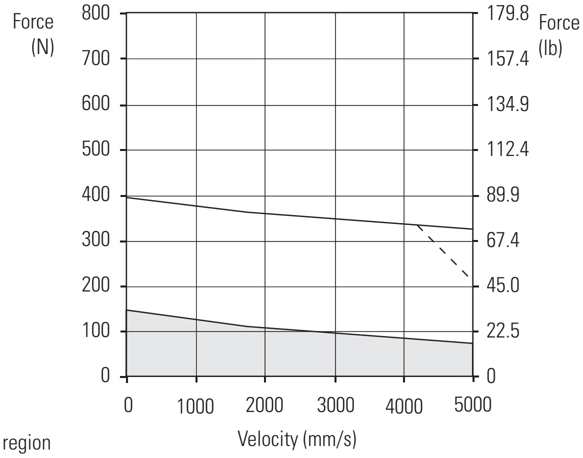
2098-DSD-HV030 and MPAS-B8*xxx*2-V20S*x*A



2098-DSD-HV030 and MPAS-B8*xxx*F-ALMO2C



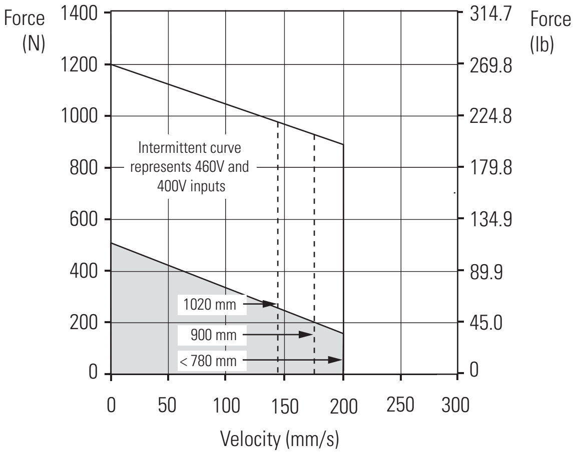
2098-DSD-HV030 and MPAS-B8*xxx*F-ALMS2C



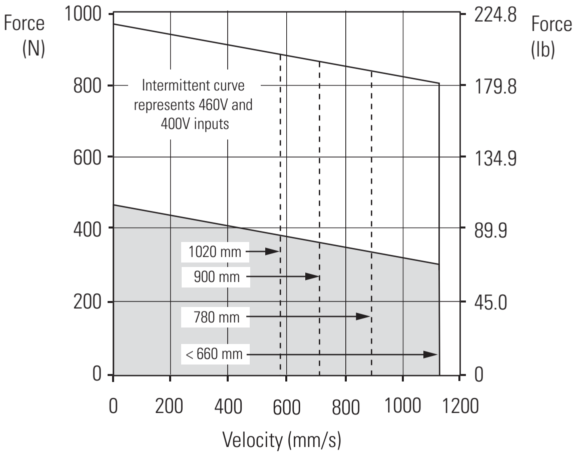
☐ = Intermittent operating region
▨ = Continuous operating region
– – – = System operation with 400V AC (rms) input voltage
- - - - = System operation for specified stroke length

Ultra3000 (460V) Drives/MP-Series Integrated Linear Stage Curves, Continued

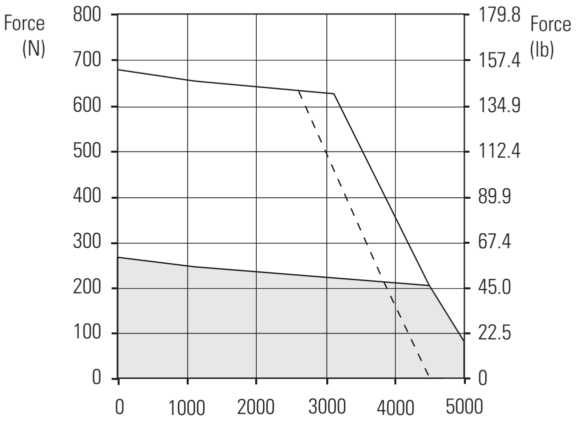
2098-DSD-HV030 and MPAS-B9*xxx*1-V05S*x*A



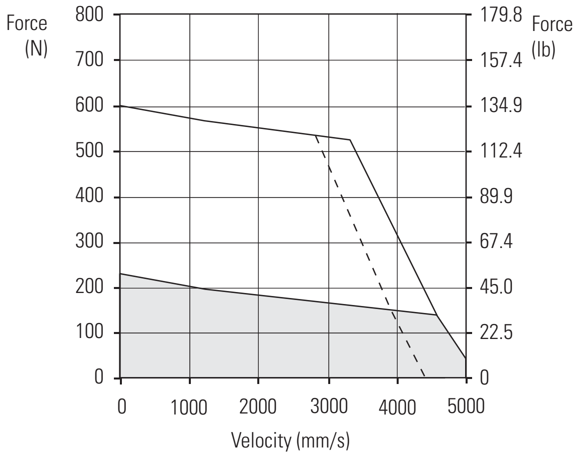
2098-DSD-HV030 and MPAS-B9*xxx*2-V20S*x*A



2098-DSD-HV030 and MPAS-B9*xxx*L-ALMO2C



2098-DSD-HV030 and MPAS-B9*xxx*L-ALMS2C



☐ = Intermittent operating region
▨ = Continuous operating region
- - - = System operation with 400V AC (rms) input voltage
- - - - = System operation for specified stroke length

Ultra3000 Drives with MP-Series Electric Cylinders

This section provides system combination information for the Ultra3000 drives when matched with MP-Series electric cylinders. Included are power/brake and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Electric Cylinder Cable Combinations

Electric Cylinders	Motor Power/Brake Cable	Motor Feedback Cable (1)
MPAR-A/B1*xxx*B MPAR-A/B1*xxx*E	2090-XXNPMF-16S*xx* (2)	2090-XXNFMF-S*xx* (3) Absolute High-resolution Feedback
MPAR-A/B2*xxx*C MPAR-A/B2*xxx*F		
MPAR-A/B3*xxx*E MPAR-A/B3*xxx*H	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback

(1) Use drive-mounted breakout board (catalog number 2090-UXBB-DM15) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

(2) These cables are available as standard (catalog number 2090-XXNPMF-16S*xx*) or continuous-flex (catalog number 2090-CP*x*M4DF-16AF*xx*).

(3) These cables are available as standard (catalog number 2090-XXNFMF-S*xx*) or continuous-flex (catalog number 2090-CFBM4DF-CDAF*xx*).

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Electric Cylinder Performance Specifications with Ultra3000 Drives

Performance Specifications with Ultra3000 (230V) Drives

Electric Cylinder	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Rated Output** kW	**Ultra3000 230V Drives**
MPAR-A1*xxx*B	150	1.15	240 (53.9)	1.35	300 (67.4)	0.036	2098-DSD-005
MPAR-A1*xxx*E	500	2.16	280 (62.9)	2.48	350 (78.7)	0.140	
MPAR-A2*xxx*C	250	2.42	420 (94.4)	2.72	525 (118)	0.105	2098-DSD-010
MPAR-A2*xxx*F	640	4.54	640 (144)	5.41	800 (180)	0.410	
MPAR-A3*xxx*E	500	10.33	2000 (450)	12.34	2500 (562)	1.00	2098-DSD-030 (1)
MPAR-A3*xxx*H	1000	12.20	1300 (292)	16.40	1625 (365)	1.30	2098-DSD-030

(1) Use of catalog number 2098-DSD020*x-xx* is acceptable for applications with actuators and continuous force derated by 5%.

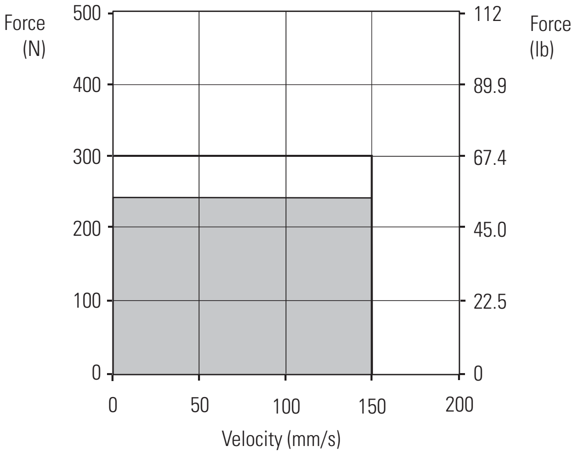
Performance Specifications with Ultra3000 (460V) Drives

Electric Cylinder	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Rated Output** kW	**Ultra3000 460V Drives**
MPAR-B1*xxx*B	150	1.15	240 (53.9)	1.35	300 (67.4)	0.036	2098-DSD-HV030
MPAR-B1*xxx*E	500	1.49	280 (62.9)	1.71	350 (78.7)	0.140	
MPAR-B2*xxx*C	250	1.67	420 (94.4)	1.90	525 (118)	0.105	
MPAR-B2*xxx*F	640	3.29	640 (144)	3.93	800 (180)	0.410	
MPAR-B3*xxx*E	500	5.16	2000 (450)	6.17	2500 (562)	1.00	
MPAR-B3*xxx*H	1000	6.13	1300 (292)	6.79	1625 (365)	1.30	

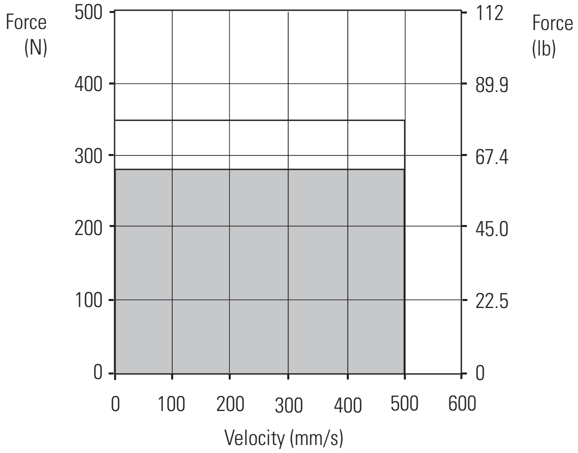
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Ultra3000 Drives/MP-Series Electric Cylinder Curves

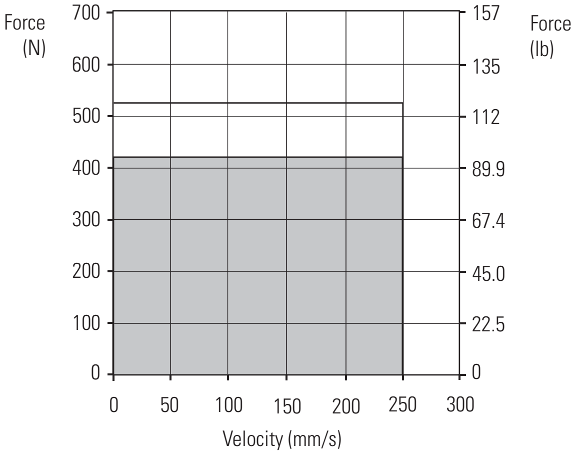
2098-DSD-005 and MPAR-A1*xxx*B-V*x*A
2098-DSD-HV030 and MPAR-B1*xxx*B-V*x*A



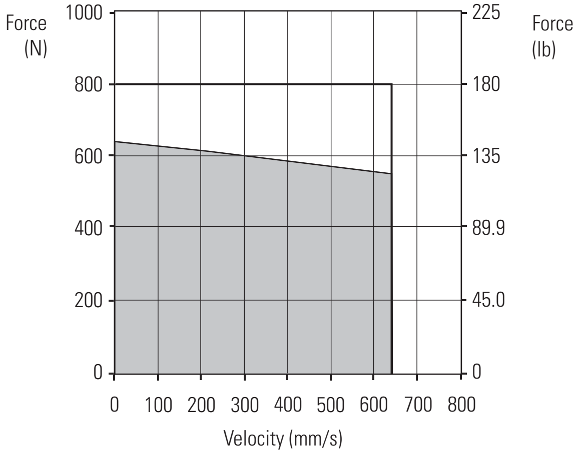
2098-DSD-005 and MPAR-A1*xxx*E-V*x*A
2098-DSD-HV030 and MPAR-B1*xxx*E-V*x*A



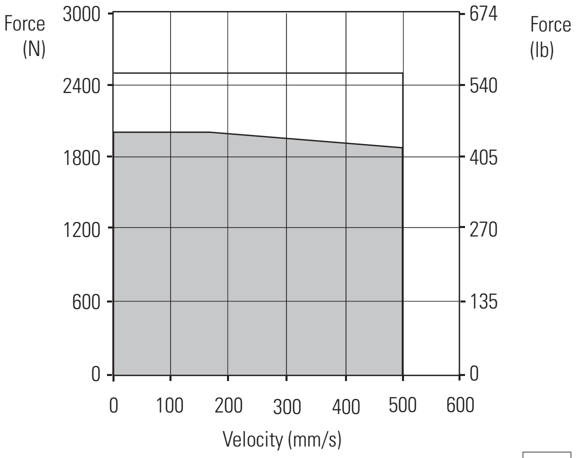
2098-DSD-010 and MPAR-A2*xxx*C-V*x*A
2098-DSD-HV030 and MPAR-B2*xxx*C-V*x*A



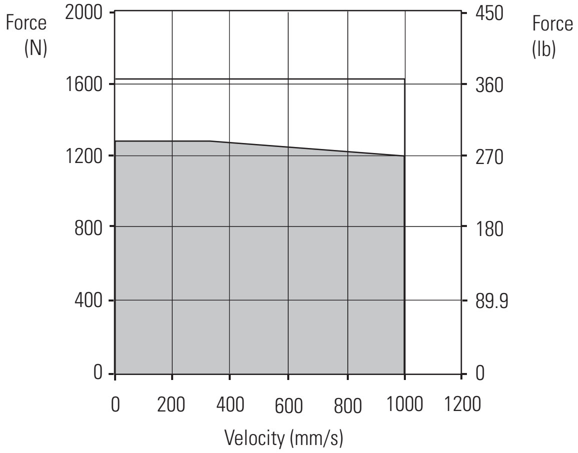
2098-DSD-010 and MPAR-A2*xxx*F-V*x*A
2098-DSD-HV030 and MPAR-B2*xxx*F-V*x*A



2098-DSD-030 and MPAR-A3*xxx*E-M*x*A
2098-DSD-HV030 and MPAR-B3*xxx*E-M*x*A



2098-DSD-030 and MPAR-A3*xxx*H-M*x*A
2098-DSD-HV030 and MPAR-B3*xxx*H-M*x*A



☐ = Intermittent operating region
▨ = Continuous operating region

Ultra3000 Drives with MP-Series Heavy Duty Electric Cylinders

This section provides system combination information for the Ultra3000 drives when matched with MP-Series heavy-duty electric cylinders. Included are power/brake and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Electric Cylinder Cable Combinations

Electric Cylinders	Motor Power/Brake Cable	Motor Feedback Cable [1]
MPAI-A/B3*xxx*C MPAI-A/B3*xxx*E MPAI-A/B3*xxx*R MPAI-A/B3*xxx*S	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-CFBM7DF-CEAA*xx* or 2090-CFBM7DD-CEAA*xx* (standard) 2090-CFBM7DF-CEAF*xx* 2090-CFBM7DD-CEAF*xx* (continuous-flex) Absolute High-resolution Feedback
MPAI-A/B4*xxx*C MPAI-A/B4*xxx*E MPAI-A/B4*xxx*R MPAI-A/B4*xxx*S		

(1) Use drive-mounted breakout board (catalog number 2090-UXBB-DM15) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

Electric Cylinder Performance Specifications with Ultra3000 (230V) Drives

Performance Specifications with Ball Screw Electric Cylinders

Electric Cylinder	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)		**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Rated Output** kW	**Ultra3000 230V Drives**
			25 °C (77 °F)	40 °C (104 °F)				
MPAI-A3*xxx*C	279 (11)	5.61	4003 (900)	3176 (714)	8.40	4448 (1000)	0.39	2098-DSD-020
MPAI-A3*xxx*E	559 (22)		2002 (450)	1588 (357)	14.14	4003 (900)		
MPAI-A4*xxx*C	279 (11)	10.89	7784 (1750)	6179 (1389)	17.07	8896 (2000)	0.43	2098-DSD-030
MPAI-A4*xxx*E	559 (22)		3892 (875)	3092 (695)	27.44	7784 (1750)		

Performance Specifications with Roller Screw Electric Cylinders

Electric Cylinder	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)		**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Rated Output** kW	**Ultra3000 230V Drives**
			25 °C (77 °F)	40 °C (104 °F)				
MPAI-A3*xxx*R	279 (11)	5.61	3781 (850)	3003 (675)	14.14	7562 (1700)	0.39	2098-DSD-020
MPAI-A3*xxx*S	559 (22)		1891 (425)	1499 (337)		3781 (850)		
MPAI-A4*xxx*R	279 (11)	10.89	7340 (1650)	5827 (1310)	27.44	14,679 (3300)	0.43	2098-DSD-030
MPAI-A4*xxx*S	559 (22)		3670 (825)	2914 (655)		7340 (1650)		

Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.8 or later.

Electric Cylinder Performance Specifications with Ultra3000 (460V) Drives

Performance Specifications with Ball Screw Electric Cylinders

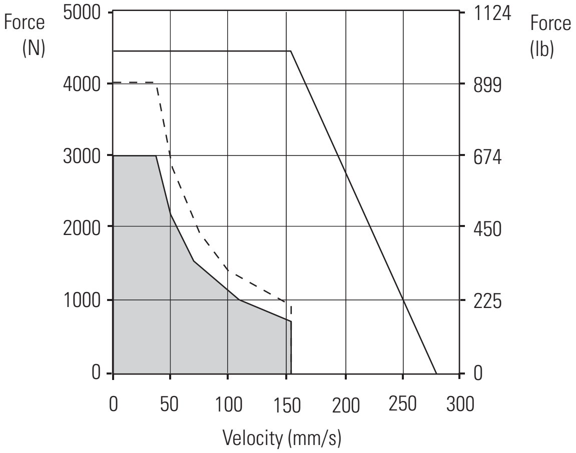
Electric Cylinder	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)		**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Rated Output** kW	**Ultra3000 460V Drives**
			25 °C (77 °F)	40 °C (104 °F)				
MPAI-B3*xxx*C	279 (11)	2.81	4003 (900)	3176 (714)	4.30	4448 (1000)	0.39	2098-DSD-HV030
MPAI-B3*xxx*E	559 (22)		2002 (450)	1588 (357)	7.07	4003 (900)		
MPAI-B4*xxx*C	279 (11)	5.61	7784 (1750)	6179 (1389)	8.68	8896 (2000)	0.43	
MPAI-B4*xxx*E	559 (22)		3892 (875)	3092 (695)	14.14	7784 (1750)		

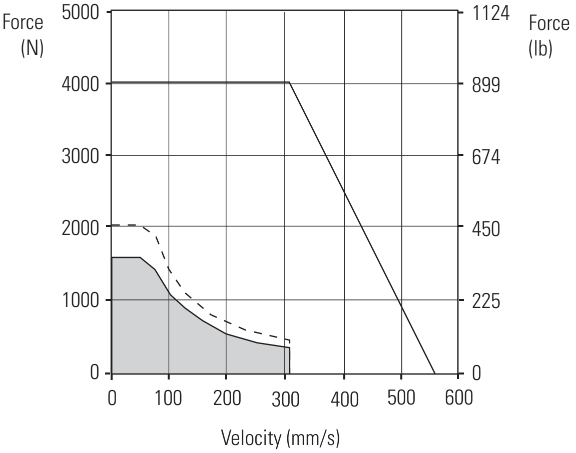
Performance Specifications with Roller Screw Electric Cylinders

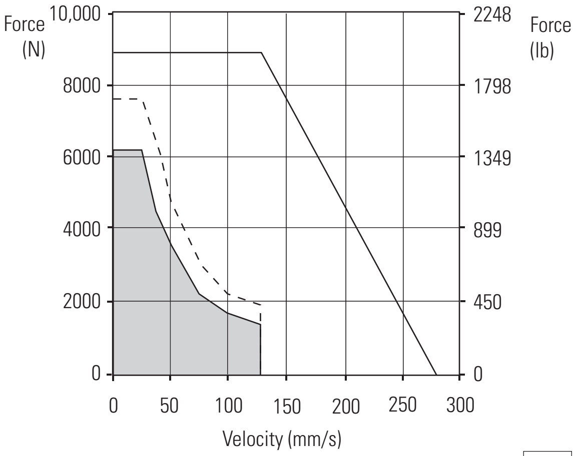
Electric Cylinder	**Maximum Speed** mm/s (in/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)		**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Rated Output** kW	**Ultra3000 460V Drives**
			25 °C (77 °F)	40 °C (104 °F)				
MPAI-B3*xxx*R	279 (11)	2.81	3781 (850)	3003 (675)	7.07	7562 (1700)	0.39	2098-DSD-HV030
MPAI-B3*xxx*S	559 (22)		1891 (425)	1499 (337)		3781 (850)		
MPAI-B4*xxx*R	279 (11)	5.61	7340 (1650)	5827 (1310)	14.14	14,679 (3300)	0.43	
MPAI-B4*xxx*S	559 (22)		3670 (825)	2914 (655)		7340 (1650)		

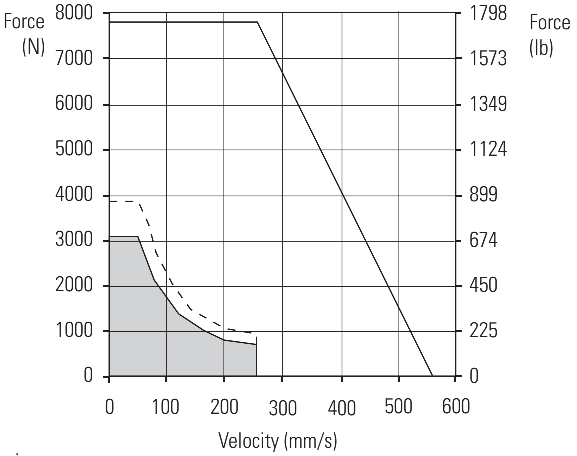
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.8 or later.

Ultra3000 Drives/MP-Series Heavy Duty (ball screw) Electric Cylinder Curves


2098-DSD-020 and MPAI-A3*xxx*C
2098-DSD-HV030 and MPAI-B3*xxx*C


2098-DSD-020 and MPAI-A3*xxx*E
2098-DSD-HV030 and MPAI-B3*xxx*E

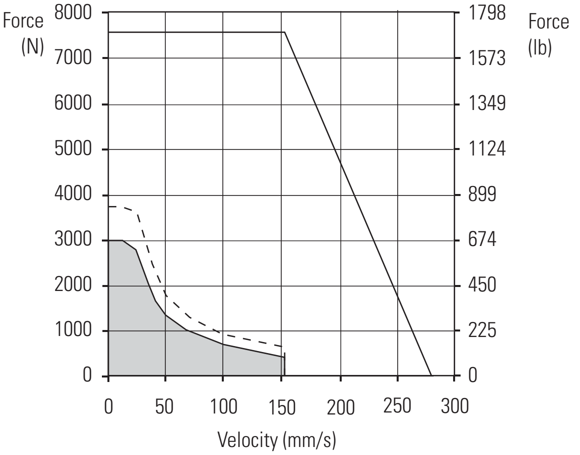

2098-DSD-030 and MPAI-A4*xxx*C
2098-DSD-HV030 and MPAI-B4*xxx*C


2098-DSD-030 and MPAI-A4*xxx*E
2098-DSD-HV030 and MPAI-B4*xxx*E

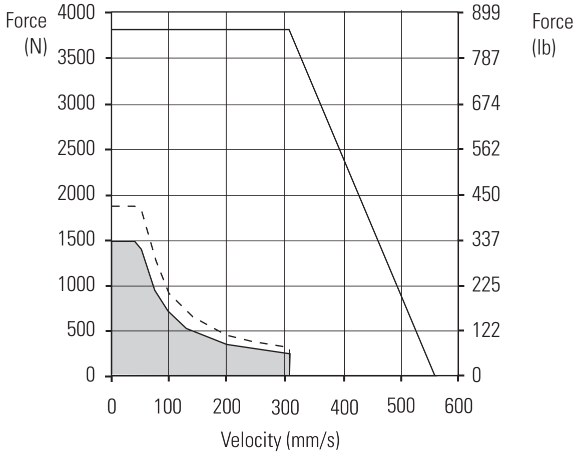
☐ = Intermittent operating region
- - - = Continuous operating region @ 25 °C (77 °F)
▨ = Continuous operating region @ 40 °C (104 °F)

Ultra3000 Drives/MP-Series Heavy Duty (roller screw) Electric Cylinder Curves

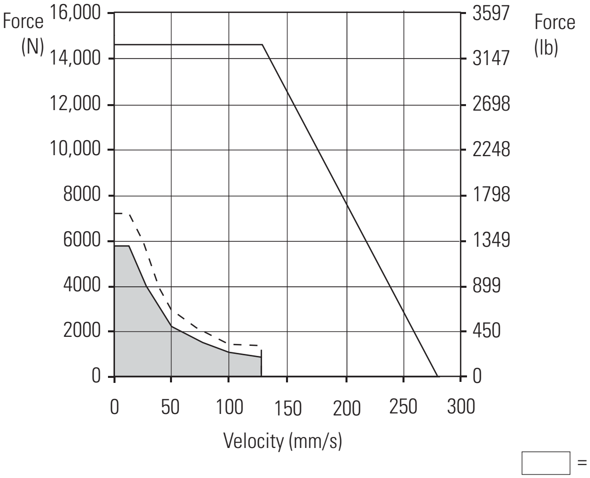
2098-DSD-020 and MPAI-A3*xxx*R
2098-DSD-HV030 and MPAI-B3*xxx*R



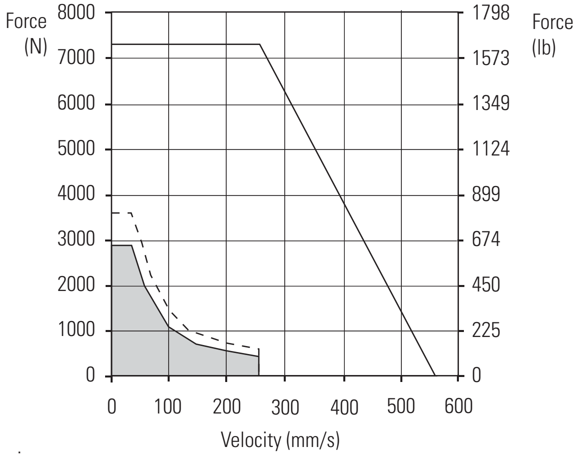
2098-DSD-020 and MPAI-A3*xxx*S
2098-DSD-HV030 and MPAI-B3*xxx*S



2098-DSD-030 and MPAI-A4*xxx*R
2098-DSD-HV030 and MPAI-B4*xxx*R



2098-DSD-030 and MPAI-A4*xxx*S
2098-DSD-HV030 and MPAI-B4*xxx*S



☐ = Intermittent operating region
- - - = Continuous operating region @ 25 °C (77 °F)
▨ = Continuous operating region @ 40 °C (104 °F)

Ultra3000 (230V) Drives with LDC-Series Linear Motors

This section provides system combination information for the Ultra3000 (230V) drives when matched with LDC-Series iron-core linear motors. Included are power and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Linear Motor Cable Combinations

Linear Motor	Motor Power Cable	Motor Feedback Cable [1]
LDC-C030100-DHT, LDC-C030200-DHT, LDC-C030200-EHT	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-XXNFMF-S*xx* (standard) 2090-CFBM7DF-CDAF*xx* (continuous-flex) Sin/Cos or TTL Encoder Feedback
LDC-C050100-DHT, LDC-C050200-DHT, LDC-C050200-EHT, LDC-C050300-DHT, LDC-C050300-EHT		
LDC-C075200-DHT, LDC-C075200-EHT, LDC-C075300-DHT, LDC-C075300-EHT, LDC-C075400-DHT, LDC-C075400-EHT		
LDC-C100300-DHT, LDC-C100300-EHT, LDC-C100400-DHT, LDC-C100400-EHT, LDC-C100600-DHT		
LDC-C150400-DHT, LDC-C150600-DHT		

(1) Use drive-mounted breakout board (catalog number 2090-UXBB-DM15) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

LDC-Series Performance Specifications with Ultra3000 (230V) Drives

Linear Motor	Maximum Speed m/s (ft/s)	System Continuous Stall Current [1] Amps 0-pk	System Continuous Stall Force [1] N (lb)	System Peak Stall Current Amps 0-pk	System Peak Stall Force N (lb)	Linear Motor Rated Output kW	Ultra3000 230V Drives
LDC-C030100-DHT	10.0 (32.8)	4.1…6.1	74…111 (17…25)	12.1	188 (42)	0.37…0.55	2098-DSD-010
LDC-C030200-DHT		8.1…12.2	148…222 (33…50)	24.3	375 (84)	0.74…1.11	2098-DSD-020
LDC-C030200-EHT		4.1…6.1		12.1			2098-DSD-010
LDC-C050100-DHT	10.0 (32.8)	3.9…5.9	119…179 (27…40)	11.7	302 (68)	0.59…0.89	2098-DSD-010
LDC-C050200-DHT		7.9…11.8	240…359 (54…81)	23.3	600 (135)	1.20…1.79	2098-DSD-020
LDC-C050200-EHT		3.9…5.9		11.6			2098-DSD-010
LDC-C050300-DHT		11.8…17.7	363…544 (82…122)	35.9	941 (212)	1.81…2.72	2098-DSD-075
LDC-C050300-EHT		3.9…5.9		12.0			2098-DSD-010

(1) Values represent the range between no cooling (low value) and water cooling (high value).

Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

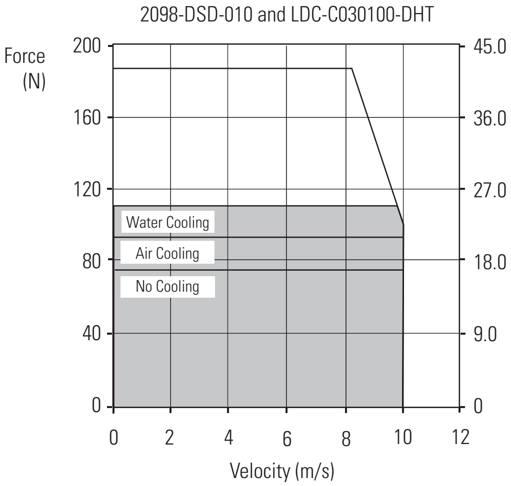
LDC-Series Performance Specifications with Ultra3000 (230V) Drives, Continued

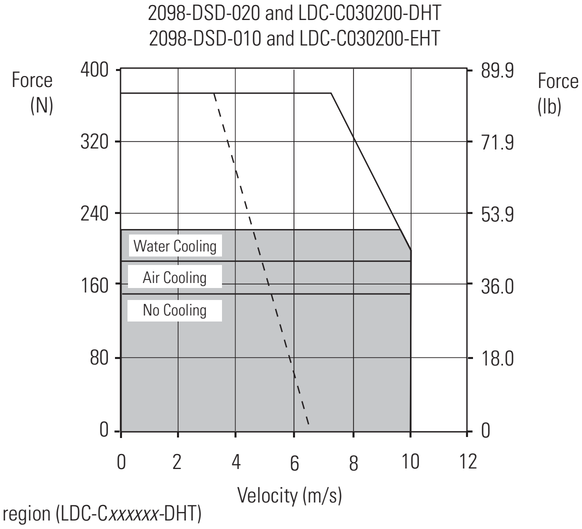
Linear Motor	Maximum Speed m/s (ft/s)	System Continuous Stall Current (1) Amps 0-pk	System Continuous Stall Force (1) N (lb)	System Peak Stall Current Amps 0-pk	System Peak Stall Force N (lb)	Linear Motor Rated Output kW	Ultra3000 230V Drives
LDC-C075200-DHT	10.0 (32.8)	7.7…11.5	348…523 (78…117)	22.9	882 (198)	1.74…2.61	2098-DSD-020
LDC-C075200-EHT		3.8…5.7		11.5			2098-DSD-010
LDC-C075300-DHT		11.5…17.2	523…784 (117…176)	35.6	1368 (308)	2.61…3.92	2098-DSD-075
LDC-C075300-EHT		3.8…5.7		11.9			2098-DSD-010
LDC-C075400-DHT		15.3…23.0	697…1045 (157…235)	47.4	1824 (410)	3.48…5.22	2098-DSD-075
LDC-C075400-EHT		7.7…11.5		23.7			2098-DSD-020
LDC-C100300-DHT	10.0 (32.8)	11.1…16.7	674…1012 (152…227)	34.3	1767 (397)	3.37…5.06	2098-DSD-075
LDC-C100300-EHT		3.7…5.6		11.4			2098-DSD-010
LDC-C100400-DHT		14.8…22.2	899…1349 (202…303)	45.7	2356 (530)	4.49…6.74	2098-DSD-075
LDC-C100400-EHT		7.4…11.1		22.8			2098-DSD-020
LDC-C100600-DHT		22.2…33.3	1349…2023 (303…455)	68.5	3534 (794)	6.74…10.11	2098-DSD-075
LDC-C150400-DHT	10.0 (32.8)	14.1…21.1	1281…1922 (288…432)	45.2	3498 (786)	6.40…9.61	2098-DSD-075
LDC-C150600-DHT		21.1…31.7	1922…2882 (432…648)	67.8	5246 (1179)	9.61…14.41	2098-DSD-075

(1) Values represent the range between no cooling (low value) and water cooling (high value).

Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Ultra3000 (230V) Drives/LDC-Series Linear Motor Curves

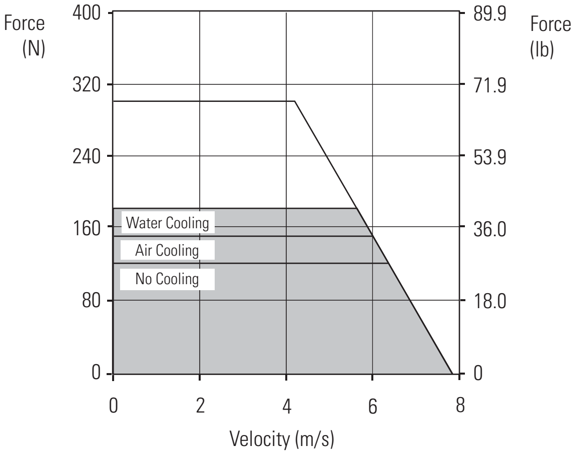




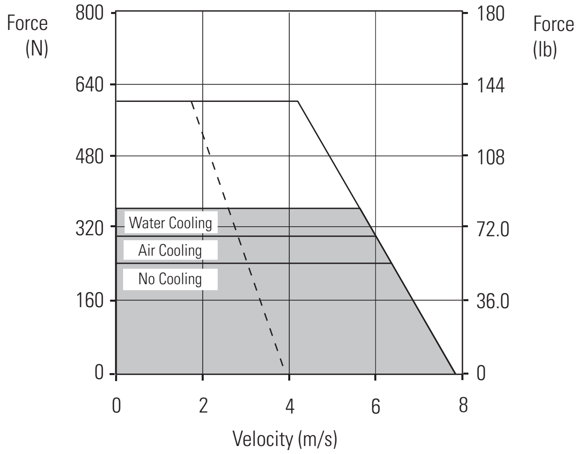
☐ = Intermittent operating region (LDC-C*xxxxxx*-DHT)
- - - = Intermittent operating region (LDC-C*xxxxxx*-EHT)
▨ = Continuous operating region

Ultra3000 (230V) Drives/LDC-Series Linear Motor Curves, Continued

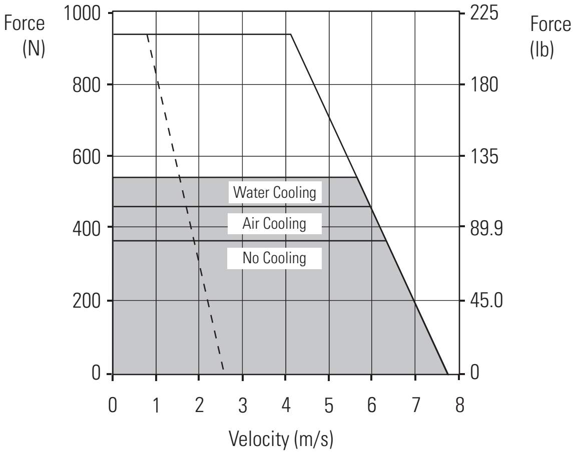
2098-DSD-010 and LDC-C050100-DHT



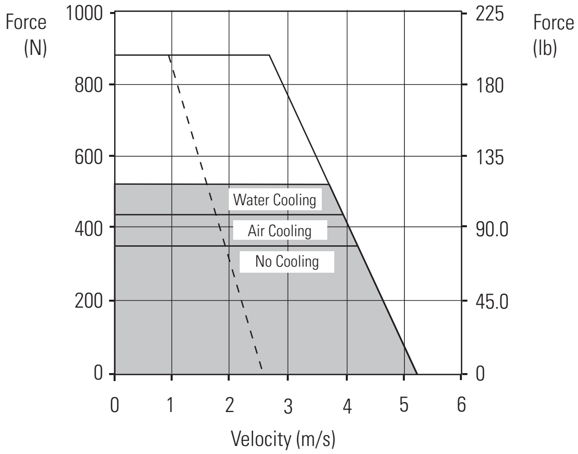
2098-DSD-020 and LDC-C050200-DHT
2098-DSD-010 and LDC-C050200-EHT



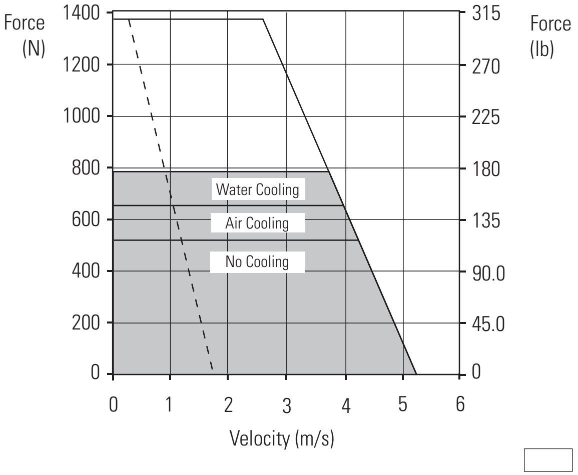
2098-DSD-075 and LDC-C050300-DHT
2098-DSD-010 and LDC-C050300-EHT



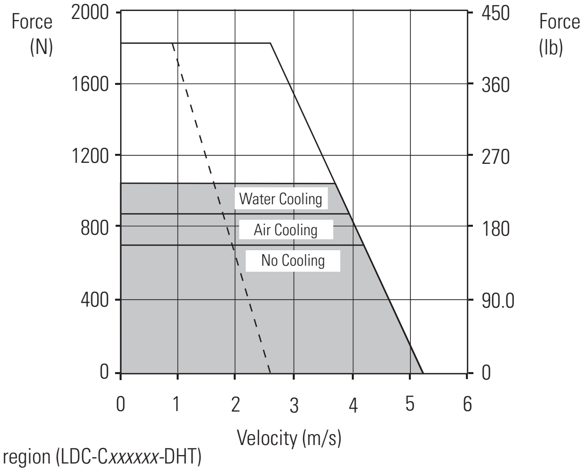
2098-DSD-020 and LDC-C075200-DHT
2098-DSD-010 and LDC-C075200-EHT



2098-DSD-075 and LDC-C075300-DHT
2098-DSD-010 and LDC-C075300-EHT



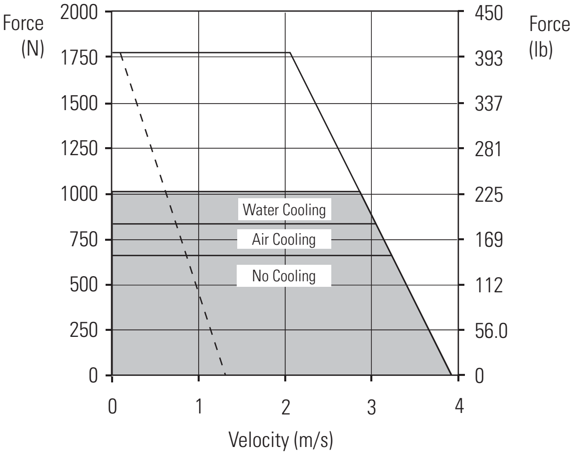
2098-DSD-075 and LDC-C075400-DHT
2098-DSD-020 and LDC-C075400-EHT



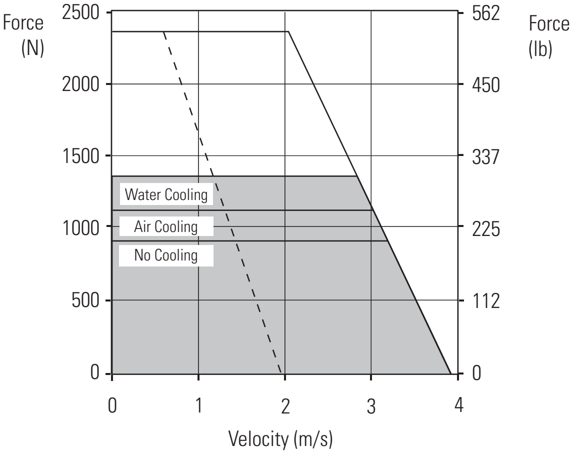
☐ = Intermittent operating region (LDC-C*xxxxxx*-DHT)
- - - = Intermittent operating region (LDC-C*xxxxxx*-EHT)
▨ = Continuous operating region

Ultra3000 (230V) Drives/LDC-Series Linear Motor Curves, Continued

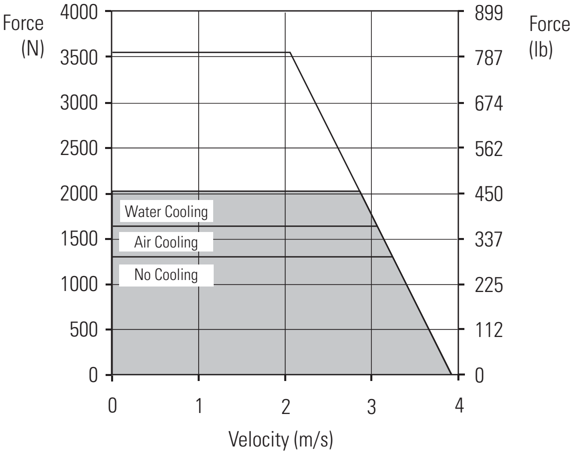
2098-DSD-075 and LDC-C100300-DHT
2098-DSD-010 and LDC-C100300-EHT



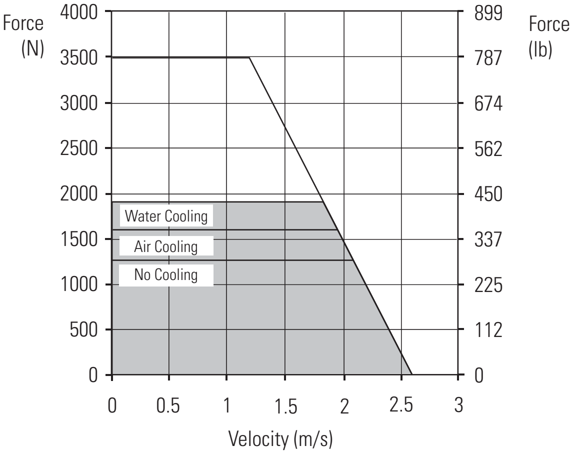
2098-DSD-075 and LDC-C100400-DHT
2098-DSD-020 and LDC-C100400-EHT



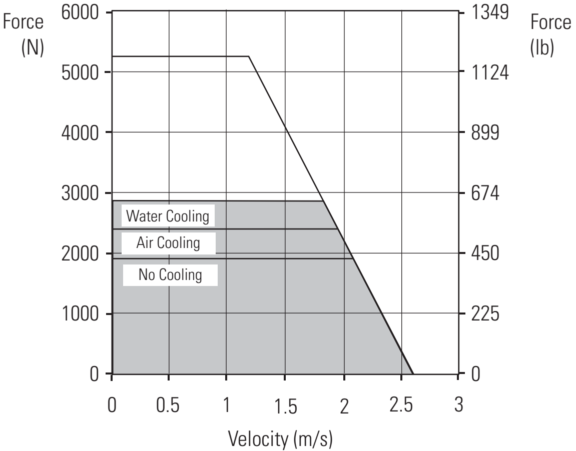
2098-DSD-075 and LDC-C100600-DHT



2098-DSD-075 and LDC-C150400-DHT



2098-DSD-075 and LDC-C150600-DHT



☐ = Intermittent operating region (LDC-C*xxxxxx*-DHT)
- - - = Intermittent operating region (LDC-C*xxxxxx*-EHT)
▨ = Continuous operating region

Ultra3000 (460V) Drives with LDC-Series Linear Motors

This section provides system combination information for the Ultra3000 (460V) drives when matched with LDC-Series iron-core linear motors. Included are power and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Linear Motor Cable Combinations

Linear Motor	Motor Power Cable	Motor Feedback Cable (1)
LDC-C030100-DHT, LDC-C030200-DHT, LDC-C030200-EHT	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-XXNFMF-S*xx* (standard) 2090-CFBM7DF-CDAF*xx* (continuous-flex) Sin/Cos or TTL Encoder Feedback
LDC-C050100-DHT, LDC-C050200-DHT, LDC-C050200-EHT, LDC-C050300-DHT, LDC-C050300-EHT		
LDC-C075200-DHT, LDC-C075200-EHT, LDC-C075300-DHT, LDC-C075300-EHT, LDC-C075400-DHT, LDC-C075400-EHT		
LDC-C100300-DHT, LDC-C100300-EHT, LDC-C100400-DHT, LDC-C100400-EHT, LDC-C100600-DHT		
LDC-C150400-DHT, LDC-C150600-DHT		

(1) Use drive-mounted breakout board (catalog number 2090-UXBB-DM15) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

LDC-Series Performance Specifications with Ultra3000 (460V) Drives

Linear Motor	**Maximum Speed** m/s (ft/s)	**System Continuous Stall Current (1)** Amps 0-pk	**System Continuous Stall Force (1)** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Motor Rated Output** kW	**Ultra3000 460V Drives**
LDC-C030100-DHT	10.0 (32.8)	4.1…6.1	74…111 (17…25)	12.1	188 (42)	0.37…0.55	2098-DSD-HV030
LDC-C030200-DHT		8.1…12.2	148…222 (33…50)	24.3	375 (84)	0.74…1.11	2098-DSD-HV100
LDC-C030200-EHT		4.1…6.1		12.1			2098-DSD-HV030
LDC-C050100-DHT	10.0 (32.8)	3.9…5.9	119…179 (27…40)	11.7	302 (68)	0.59…0.89	2098-DSD-HV030
LDC-C050200-DHT		7.9…11.8	240…359 (54…81)	23.3	600 (135)	1.20…1.79	2098-DSD-HV100
LDC-C050200-EHT		3.9…5.9		11.6			2098-DSD-HV030
LDC-C050300-DHT		11.8…17.7	363…544 (82…122)	35.9	941 (212)	1.81…2.72	2098-DSD-HV100
LDC-C050300-EHT		3.9…5.9		12.0			2098-DSD-HV030

(1) Values represent the range between no cooling (low value) and water cooling (high value).

Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

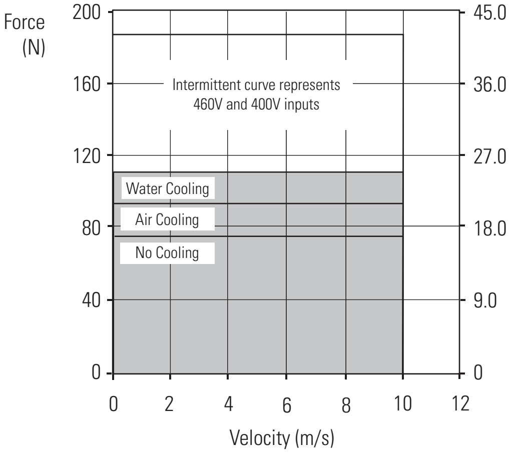
LDC-Series Performance Specifications with Ultra3000 (460V) Drives, Continued

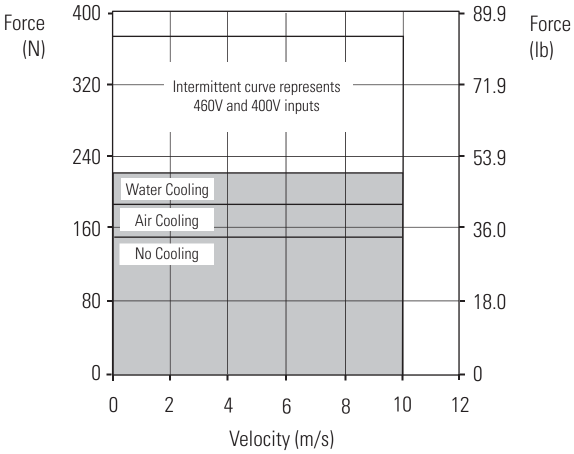
Linear Motor	**Maximum Speed** m/s (ft/s)	**System Continuous Stall Current** [1] Amps 0-pk	**System Continuous Stall Force** [1] N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Motor Rated Output** kW	**Ultra3000 460V Drives**
LDC-C075200-DHT	10.0 (32.8)	7.7…11.5	348…523 (78…117)	22.9	882 (198)	1.74…2.61	2098-DSD-HV100
LDC-C075200-EHT		3.8…5.7		11.5			2098-DSD-HV030
LDC-C075300-DHT		11.5…17.2	523…784 (117…176)	35.6	1368 (308)	2.61…3.92	2098-DSD-HV100
LDC-C075300-EHT		3.8…5.7		11.9			2098-DSD-HV030
LDC-C075400-DHT		15.3…23.0	697…1045 (157…235)	47.4	1824 (410)	3.48…5.22	2098-DSD-HV150
LDC-C075400-EHT		7.7…11.5		23.7			2098-DSD-HV100
LDC-C100300-DHT	10.0 (32.8)	11.1…16.7	674…1012 (152…227)	34.3	1767 (397)	3.37…5.06	2098-DSD-HV100
LDC-C100300-EHT		3.7…5.6		11.4			2098-DSD-HV030
LDC-C100400-DHT		14.8…22.2	899…1349 (202…303)	45.7	2356 (530)	4.49…6.74	2098-DSD-HV150
LDC-C100400-EHT		7.4…11.1		22.8			2098-DSD-HV100
LDC-C100600-DHT		22.2…33.3	1349…2023 (303…455)	68.5	3534 (794)	6.74…10.11	2098-DSD-HV220
LDC-C100600-EHT		11.1…16.7		34.3			2098-DSD-HV100
LDC-C150400-DHT	10.0 (32.8)	14.1…21.1	1281…1922 (288…432)	45.2	3498 (786)	6.40…9.61	2098-DSD-HV150
LDC-C150400-EHT		7.0…10.6		22.6			2098-DSD-HV100
LDC-C150600-DHT		21.1…31.7	1922…2882 (432…648)	67.8	5246 (1179)	9.61…14.41	2098-DSD-HV220
LDC-C150600-EHT		10.6…15.8		33.9			2098-DSD-HV100

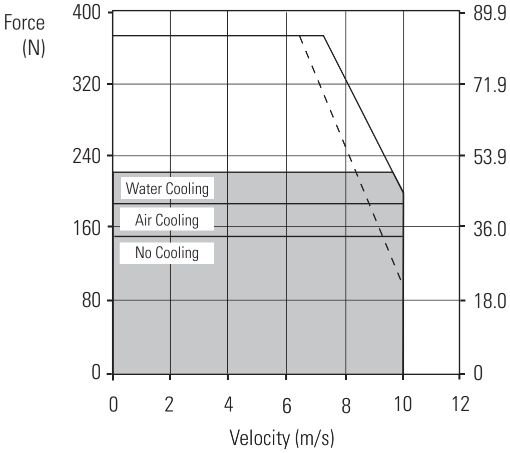
(1) Values represent the range between no cooling (low value) and water cooling (high value).

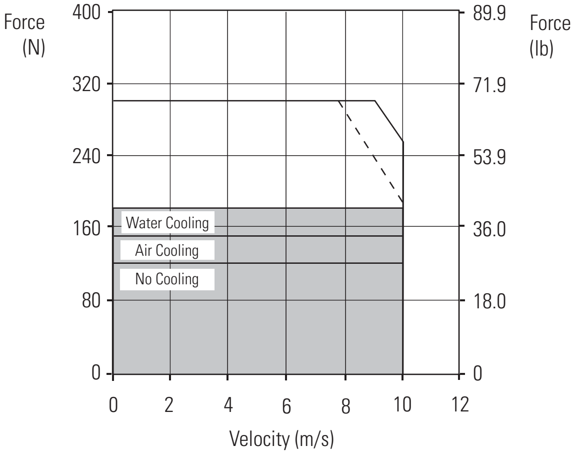
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

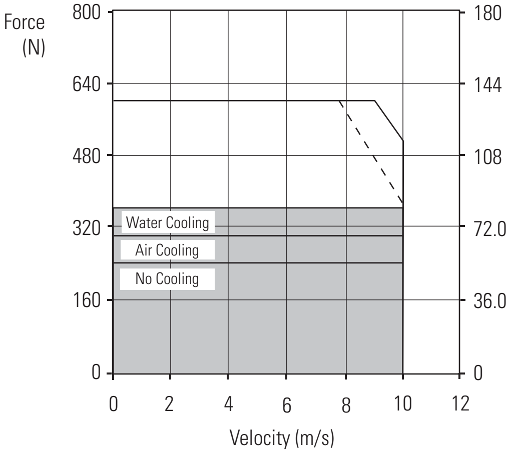
Ultra3000 (460V) Drives/LDC-Series Linear Motor Curves

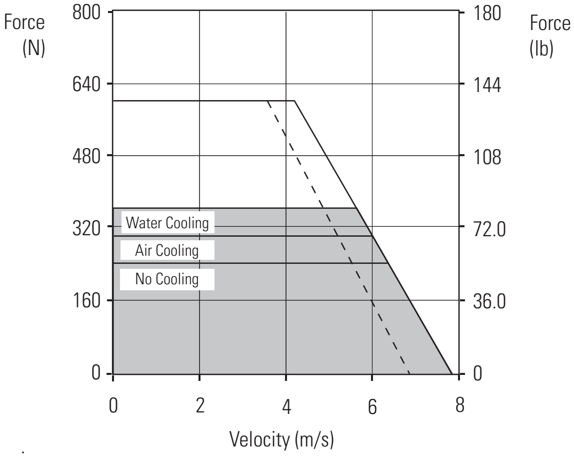

2098-DSD-HV030 and LDC-C030100-DHT


2098-DSD-HV100 and LDC-C030200-DHT


2098-DSD-HV030 and LDC-C030200-EHT

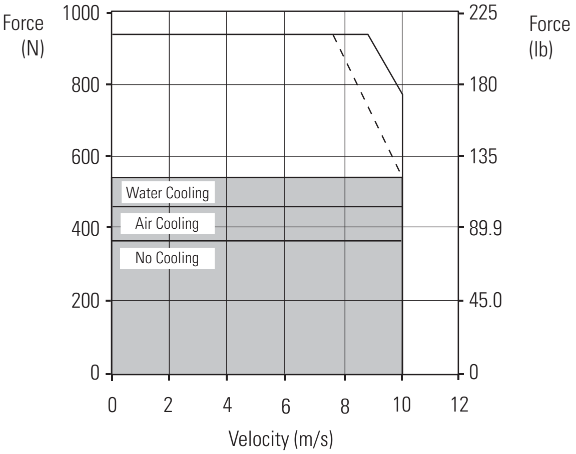

2098-DSD-HV030 and LDC-C050100-DHT

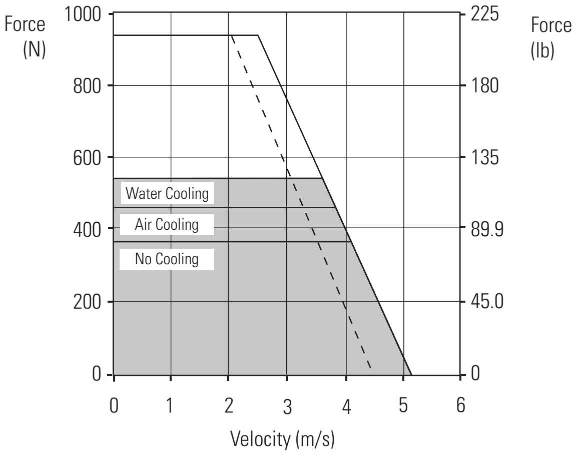

2098-DSD-HV100 and LDC-C050200-DHT

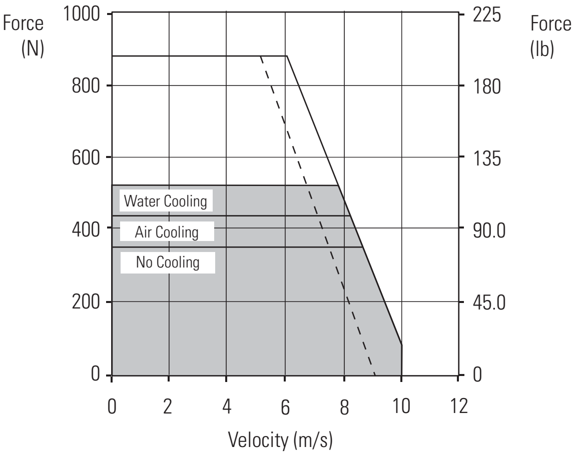

2098-DSD-HV030 and LDC-C050200-EHT

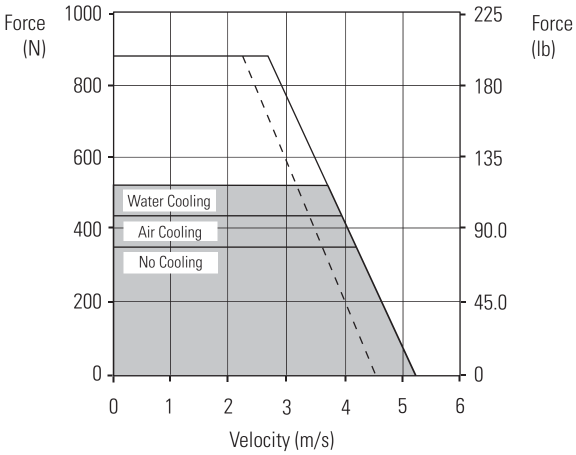
☐ = Intermittent operating region
- - - = Intermittent operating region with 400V AC (rms) input voltage
▨ = Continuous operating region

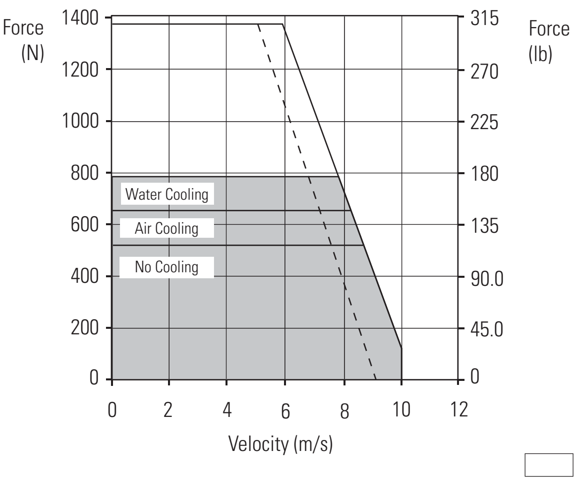
Ultra3000 (460V) Drives/LDC-Series Linear Motor Curves, Continued

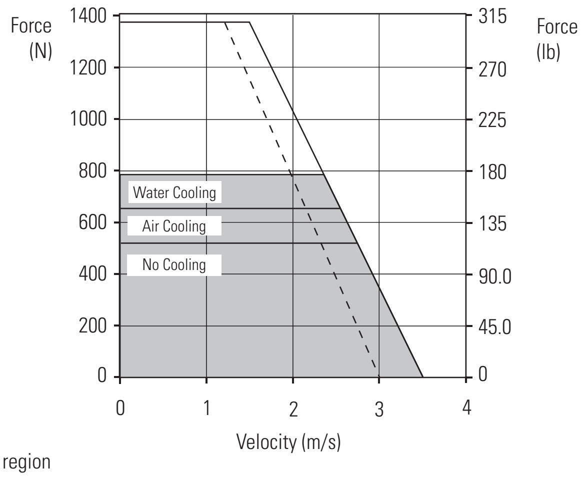

2098-DSD-HV100 and LDC-C050300-DHT


2098-DSD-HV030 and LDC-C050300-EHT


2098-DSD-HV100 and LDC-C075200-DHT


2098-DSD-HV030 and LDC-C075200-EHT

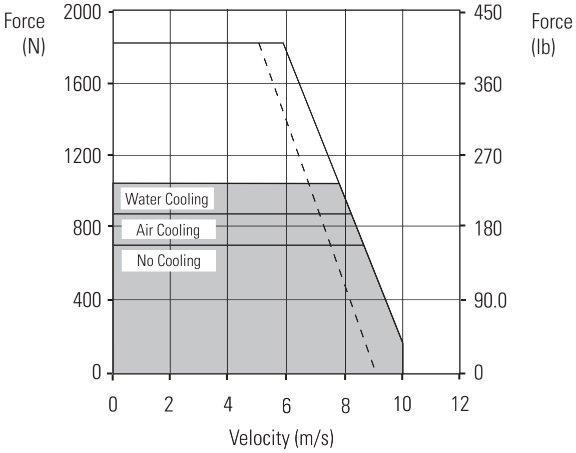

2098-DSD-HV100 and LDC-C075300-DHT


2098-DSD-HV030 and LDC-C075300-EHT

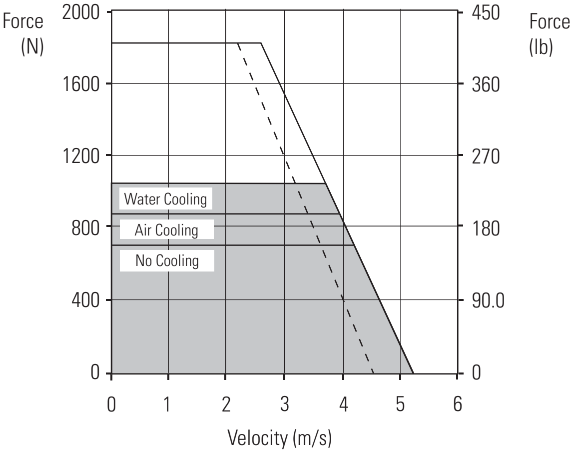
☐ = Intermittent operating region
- - - = Intermittent operating region with 400V AC (rms) input voltage
▨ = Continuous operating region

Ultra3000 (460V) Drives/LDC-Series Linear Motor Curves, Continued

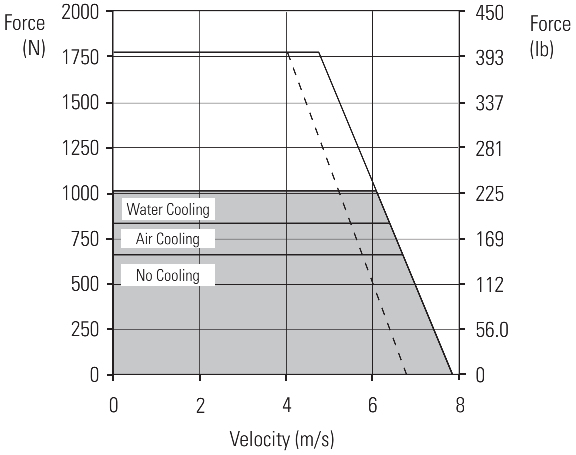
2098-DSD-HV150 and LDC-C075400-DHT



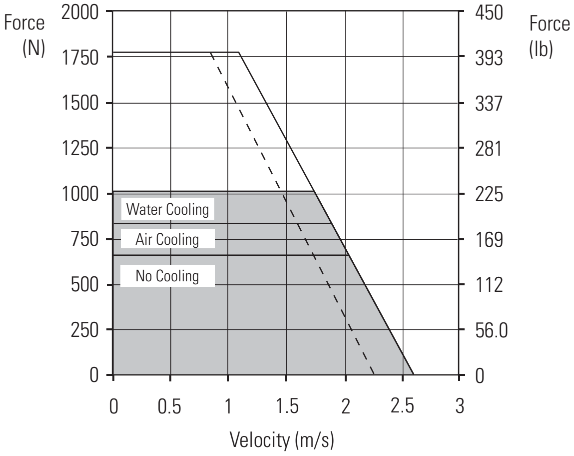
2098-DSD-HV100 and LDC-C075400-EHT



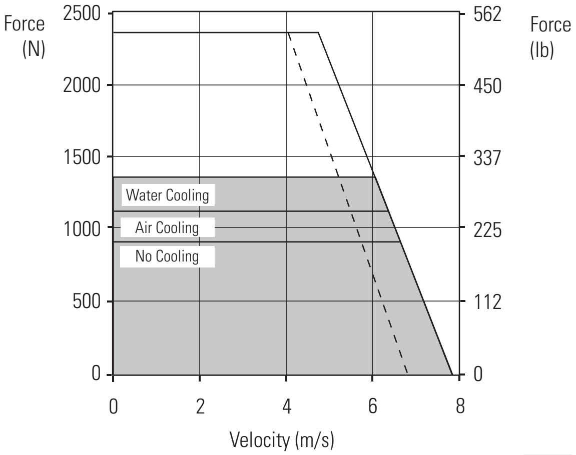
2098-DSD-HV100 and LDC-C100300-DHT



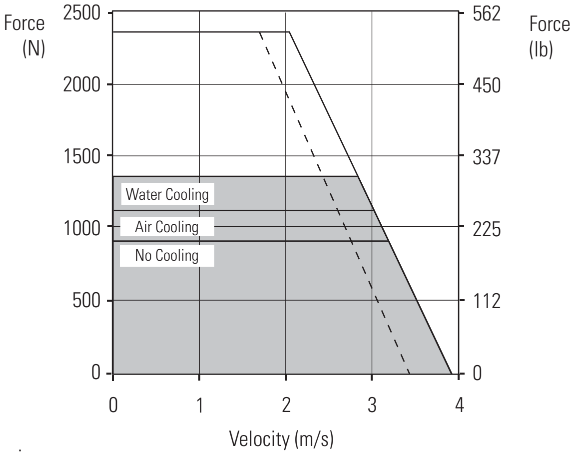
2098-DSD-HV030 and LDC-C100300-EHT



2098-DSD-HV150 and LDC-C100400-DHT



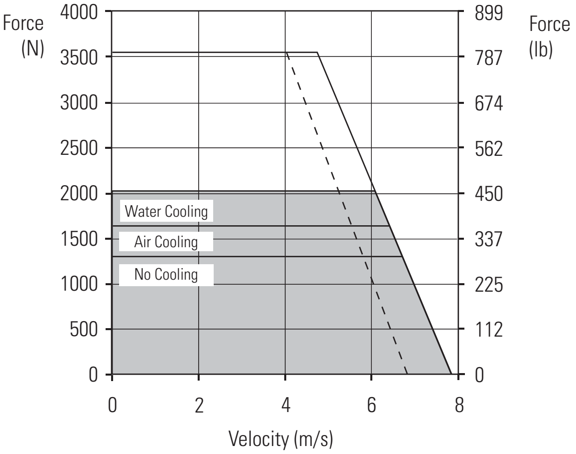
2098-DSD-HV100 and LDC-C100400-EHT



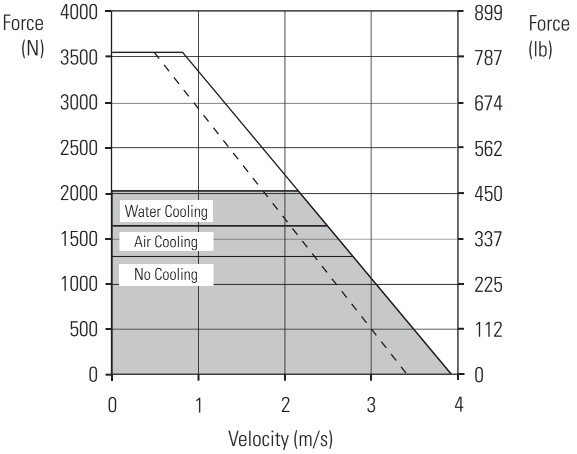
☐ = Intermittent operating region
- - - = Intermittent operating region with 400V AC (rms) input voltage
▨ = Continuous operating region

Ultra3000 (460V) Drives/LDC-Series Linear Motor Curves, Continued

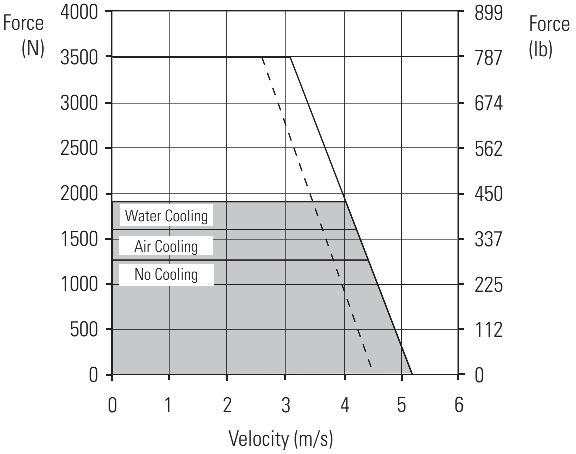
2098-DSD-HV220 and LDC-C100600-DHT



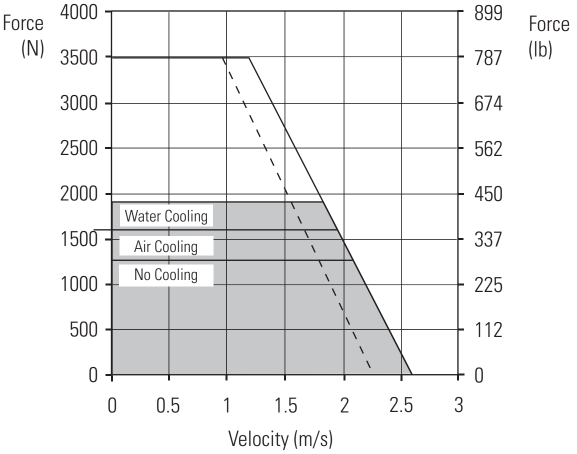
2098-DSD-HV100 and LDC-C100600-EHT



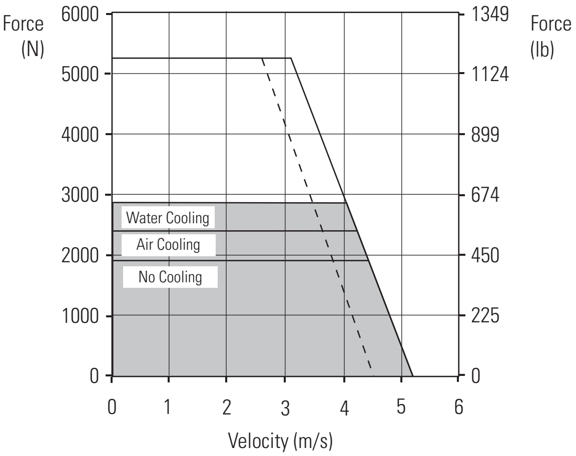
2098-DSD-HV150 and LDC-C150400-DHT



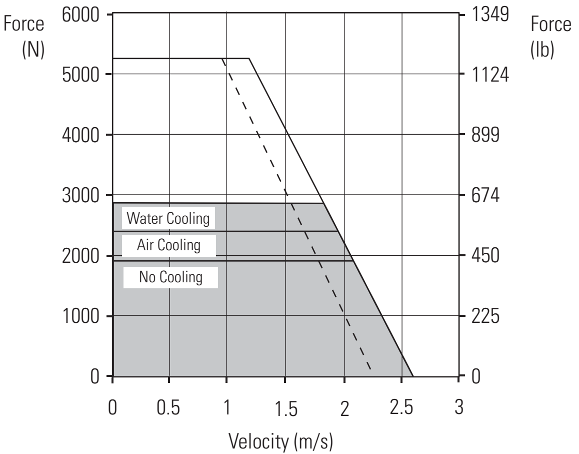
2098-DSD-HV100 and LDC-C150400-EHT



2098-DSD-HV220 and LDC-C150600-DHT



2098-DSD-HV100 and LDC-C150600-EHT



☐ = Intermittent operating region
- - - = Intermittent operating region with 400V AC (rms) input voltage
▨ = Continuous operating region

Ultra3000 (230V) Drives with LDL-Series Linear Motors

This section provides system combination information for the Ultra3000 (230V) drives when matched with LDL-Series ironless linear motors. Included are power and feedback cable catalog numbers, system performance specifications, and the optimum force/velocity curves.

Linear Motor Cable Combinations

Linear Motors	Motor Power Cable	Motor Feedback Cable [1]
LDL-N030120-DHT, LDL-N030240-DHT, LDL-N030240-EHT	2090-CP*x*M7DF-16AA*xx* (standard) 2090-CP*x*M7DF-16AF*xx* (continuous-flex)	2090-XXNFMF-S*xx* (standard) 2090-CFBM7DF-CDAF*xx* (continuous-flex) Sin/Cos or TTL Encoder Feedback
LDL-N050120-DHT, LDL-N050240-DHT, LDL-N050240-EHT, LDL-N050360-DHT, LDL-N050360-EHT, LDL-N050480-DHT, LDL-N050480-EHT		
LDL-N075480-DHT, LDL-N075480-EHT		
LDL-T030120-DHT, LDL-T030240-DHT, LDL-T030240-EHT		
LDL-T050120-DHT, LDL-T050240-DHT, LDL-T050240-EHT, LDL-T050360-DHT, LDL-T050480-DHT, LDL-T050480-EHT		
LDL-T075480-EHT, LDL-T075480-EHT		

(1) Use drive-mounted breakout board (catalog number 2090-UXBB-DM15) or panel-mounted breakout components on drive end. Refer to Breakout Components and Connector Kits on page 440.

Motor-end connector kits are available for motor power/brake and feedback cables. Refer to 2090-Series Motor-end Cable Connector Kits on page 422 for catalog numbers. Cable length *xx* is in meters. Refer to 2090-Series Motor/Actuator Cable Specifications beginning on page 410.

LDL-Series Performance Specifications with Ultra3000 (230V) Drives

Linear Motor	**Maximum Speed** m/s (ft/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Motor Rated Output** kW	Ultra3000 230V Drives
LDL-N030120-DHT	10.0 (32.8)	3.0	63 (14)	9.9	209 (47)	0.31	2098-DSD-010
LDL-N030240-DHT		6.0	126 (28)	19.9	417 (94)	0.63	2098-DSD-020
LDL-N030240-EHT		3.0		9.9			2098-DSD-010
LDL-T030120-DHT		3.0	72 (16)	9.9	239 (54)	0.36	2098-DSD-010
LDL-T030240-DHT		6.0	144 (32)	19.9	479 (108)	0.72	2098-DSD-020
LDL-T030240-EHT		3.0		9.9			2098-DSD-010

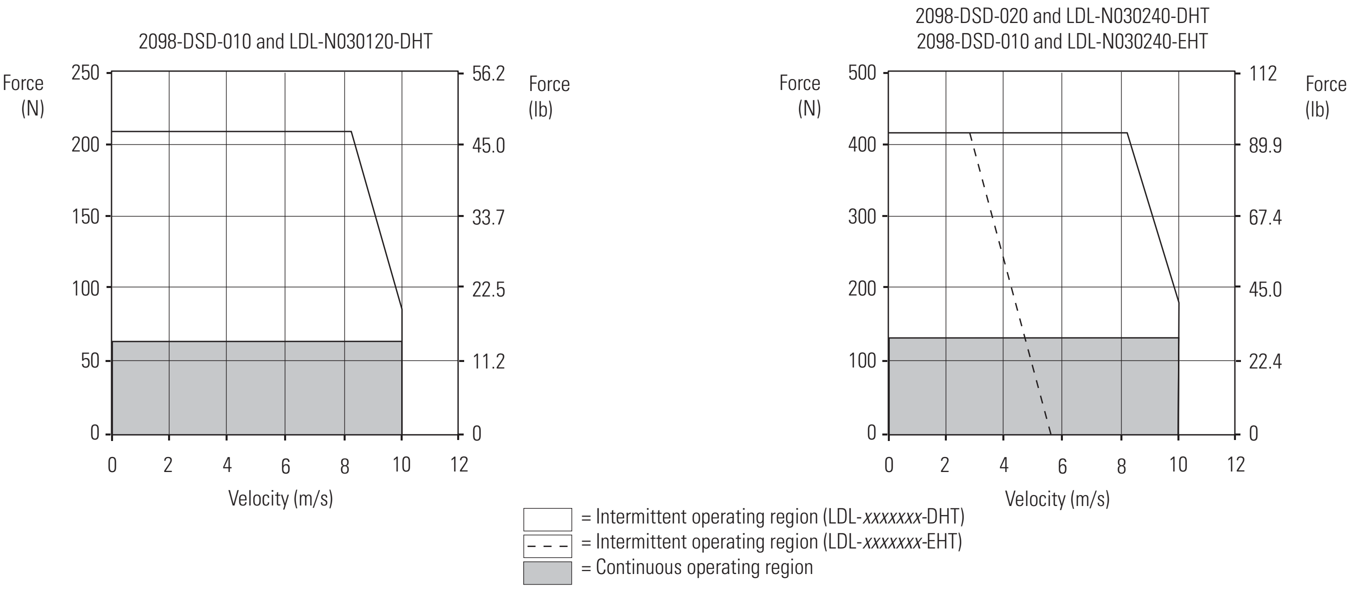
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

LDL-Series Performance Specifications with Ultra3000 (230V) Drives, Continued

Linear Motor	**Maximum Speed** m/s (ft/s)	**System Continuous Stall Current** Amps 0-pk	**System Continuous Stall Force** N (lb)	**System Peak Stall Current** Amps 0-pk	**System Peak Stall Force** N (lb)	**Linear Motor Rated Output** kW	**Ultra3000 230V Drives**
LDL-N050120-DHT	10.0 (32.8)	2.7	96 (22)	9.1	317 (71)	0.48	2098-DSD-010
LDL-N050240-DHT		5.5	191 (43)	18.1	635 (143)	0.95	2098-DSD-020
LDL-N050240-EHT		2.7		9.1			2098-DSD-010
LDL-N050360-DHT		8.2	287 (65)	27.2	952 (214)	1.43	2098-DSD-020
LDL-N050360-EHT		2.7		9.1			2098-DSD-010
LDL-N050480-DHT		10.9	383 (86)	36.3	1269 (285)	1.91	2098-DSD-075
LDL-N050480-EHT		5.5		18.1			2098-DSD-020
LDL-T050120-DHT		2.7	110 (25)	9.1	364 (82)	0.55	2098-DSD-010
LDL-T050240-DHT		5.5	220 (49)	18.1	728 (164)	1.10	2098-DSD-020
LDL-T050240-EHT		2.7		9.1			2098-DSD-010
LDL-T050360-DHT		8.2	329 (74)	27.2	1093 (246)	1.64	2098-DSD-020
LDL-T050480-DHT		10.9	439 (99)	36.3	1457 (327)	2.19	2098-DSD-075
LDL-T050480-EHT		5.5		18.1			2098-DSD-020
LDL-N075480-DHT	10.0 (32.8)	9.9	519 (117)	32.8	1723 (387)	2.59	2098-DSD-075
LDL-N075480-EHT		4.9		16.4			2098-DSD-020
LDL-T075480-DHT		9.9	596 (134)	32.8	1977 (444)	2.98	2098-DSD-075
LDL-T075480-EHT		4.9		16.4			2098-DSD-020

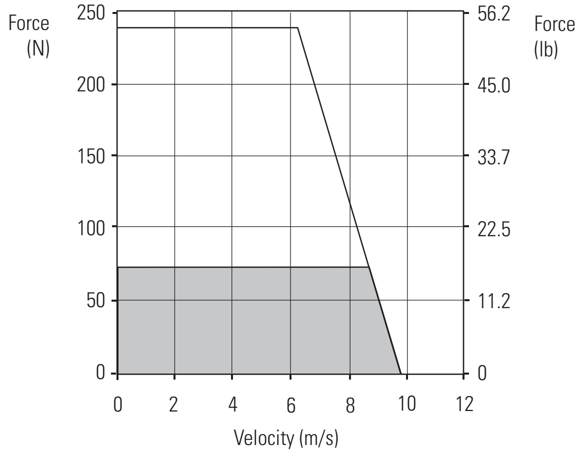
Performance specification data and curves reflect nominal system performance of a typical system with actuator at 40 °C (104 °F) and drive at 50 °C (122 °F) ambient and rated line voltage. For additional information on ambient and line conditions, refer to Motion Analyzer software, version 4.7 or later.

Ultra3000 (230V) Drives/LDL-Series Linear Motor Curves

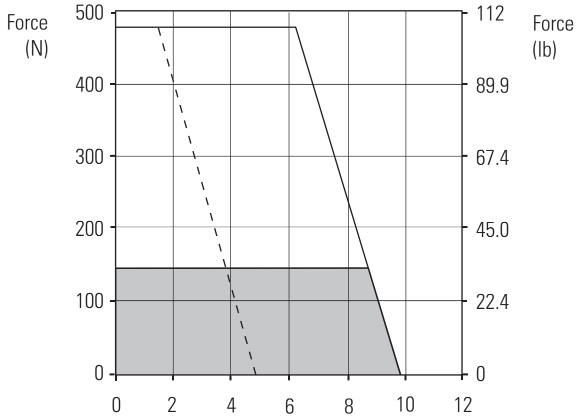


Ultra3000 (230V) Drives/LDL-Series Linear Motor Curves, Continued

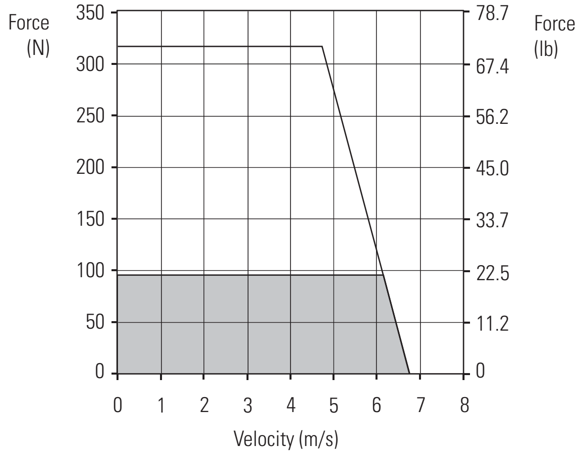
2098-DSD-010 and LDL-T030120-DHT



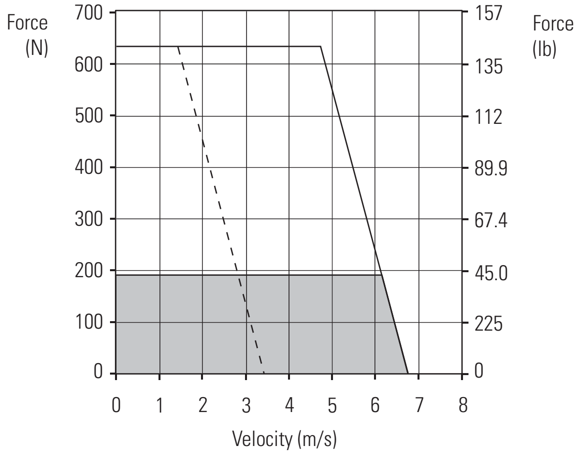
2098-DSD-020 and LDL-T030240-DHT
2098-DSD-010 and LDL-T030240-EHT



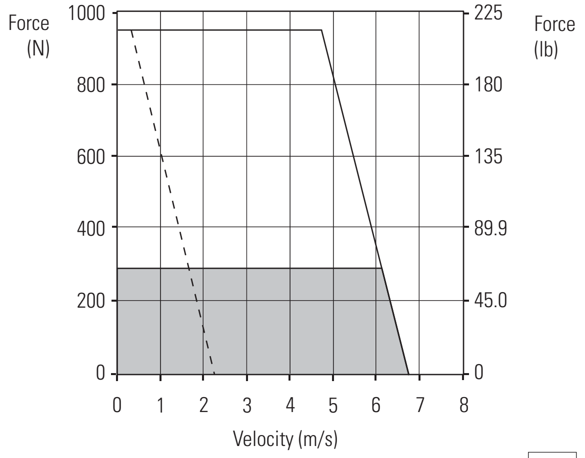
2098-DSD-010 and LDL-N050120-DHT



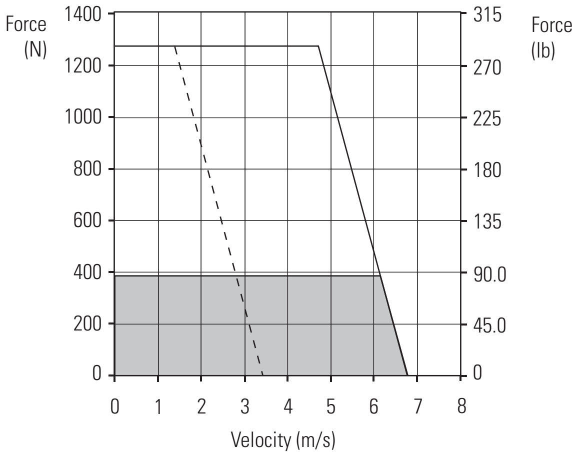
2098-DSD-020 and LDL-N050240-DHT,
2098-DSD-010 and LDL-N050240-EHT



2098-DSD-020 and LDL-N050360-DHT
2098-DSD-010 and LDL-N050360-EHT



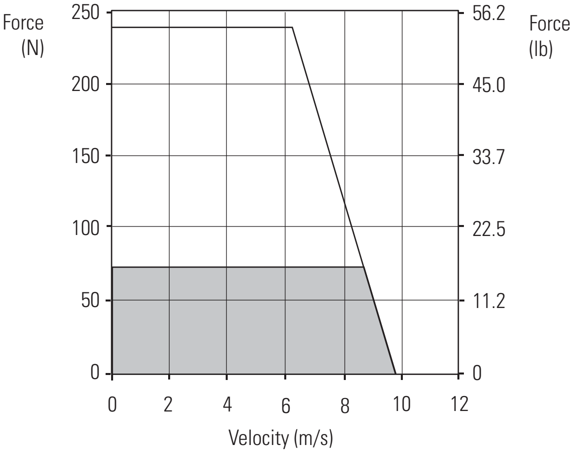
2098-DSD-075 and LDL-N050480-DHT
2098-DSD-020 and LDL-N050480-EHT



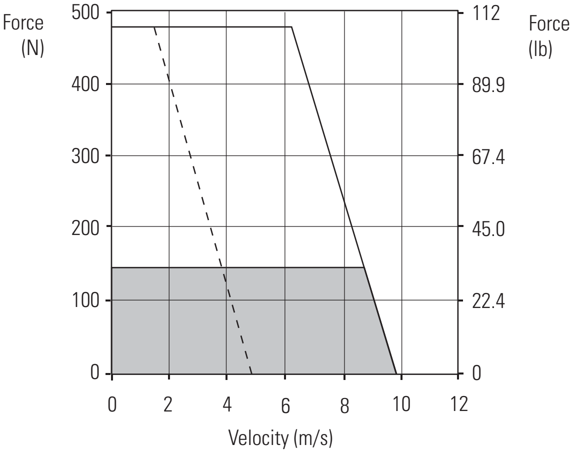
☐ = Intermittent operating region (LDL-*xxxxxxx*-DHT)
- - - = Intermittent operating region (LDL-*xxxxxxx*-EHT)
▨ = Continuous operating region

Ultra3000 (230V) Drives/LDL-Series Linear Motor Curves, Continued

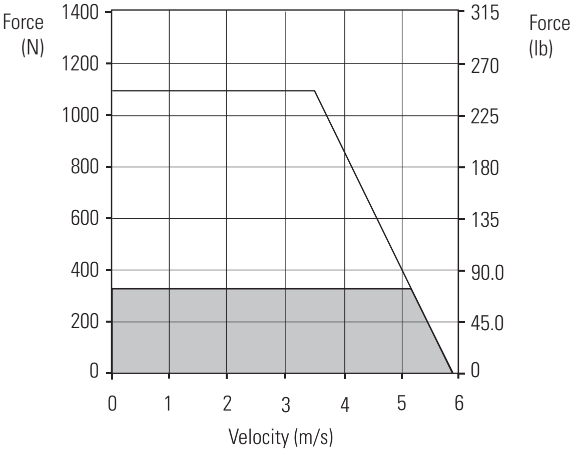
2098-DSD-010 and LDL-T030120-DHT



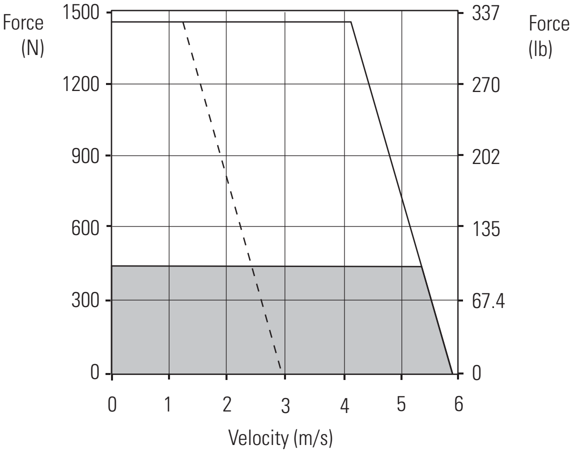
2098-DSD-020 and LDL-T030240-DHT
2098-DSD-010 and LDL-T030240-EHT



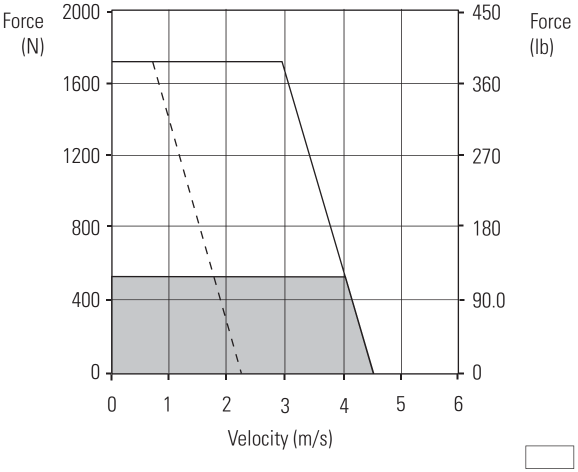
2098-DSD-020 and LDL-T050360-DHT



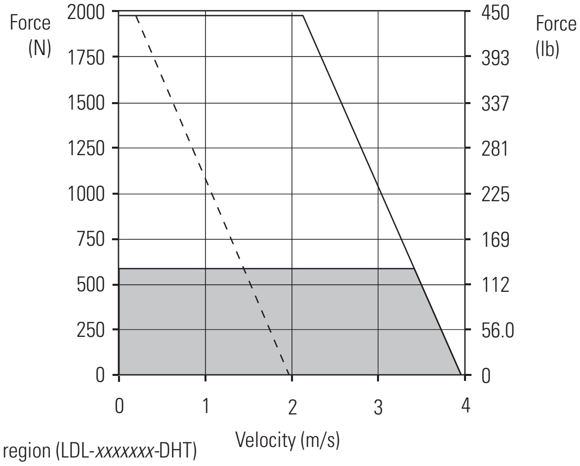
2098-DSD-075 and LDL-T050480-DHT
2098-DSD-020 and LDL-T050480-EHT



2098-DSD-075 and LDL-N075480-DHT
2098-DSD-020 and LDL-N075480-EHT



2098-DSD-075 and LDL-T075480-DHT
2098-DSD-020 and LDL-T075480-EHT



☐ = Intermittent operating region (LDL-*xxxxxxx*-DHT)
╌ = Intermittent operating region (LDL-*xxxxxxx*-EHT)
▨ = Continuous operating region

Notes:

Notes:

For more information, refer to our website: http://www.ab.com/motion.
For Allen-Bradley Technical Support information, refer to: http://www.ab.com/support or Tel: (1) 440.646.5800.

www.rockwellautomation.com

Power, Control and Information Solutions Headquarters

Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
Europe/Middle East/Africa: Rockwell Automation, Vorstlaan/Boulevard du Souverain 36, 1170 Brussels, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia Pacific: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Publication GMC-SG001P-EN-P - January 2011
Supersedes Publication GMC-SG001O-EN-P - March 2010

ANEXO J. Instrucciones de Control de Movimiento Controladores Logix5000.

Instrucciones de control de movimiento de controladores Logix5000

Manual de referencia

**Números de catálogo 1756-L1M1, 1756-L1M2,
1756-L1M3, 1756-L55M12, 1756-L55M13,
1756-L55M14, 1756-L55M16, 1756-L55M22,
1756-L55M23, 1756-L55M24, 1756-L60M03SE,
1756-L61, 1756-L62, 1756-L63, 1756-L64,
1768-L43, 1789-L60, 1789-20D**

Allen-Bradley • Rockwell Software

Rockwell Automation

Información importante para el usuario

Los equipos de estado sólido tienen características de funcionamiento distintas de las de los equipos electromecánicos. El documento Safety Guidelines for the Application, Installation and Maintenance of Solid State Controls (publicación SGI-1.1 disponible en la oficina de ventas local del Rockwell Automation o en línea en http://literature.rockwellautomation.com) describe algunas diferencias importantes entre los equipos de estado sólido y los dispositivos electromecánicos de lógica cableada. Debido a esta diferencia y también a la gran variedad de usos de los equipos de estado sólido, toda persona encargada de la aplicación de estos equipos debe asegurarse de la idoneidad de cada una de las aplicaciones concebidas con estos equipos.

En ningún caso Rockwell Automation, Inc. será responsable ni asumirá ninguna obligación por daños indirectos o consecuentes que resulten del uso o de la aplicación de estos equipos.

Los ejemplos y diagramas de este manual se incluyen únicamente con fines ilustrativos. Debido a las numerosas variables y requisitos asociados con cualquier instalación en particular, Rockwell Automation, Inc. no puede asumir responsabilidad u obligación alguna por el uso de estos equipos basado en ejemplos y diagramas.

Rockwell Automation, Inc. no asume ninguna obligación de patente con respecto al uso de la información, los circuitos, los equipos o el software descritos en este manual.

Se prohibe la reproducción total o parcial del contenido de este manual sin la autorización escrita de Rockwell Automation, Inc.

Este manual contiene notas de seguridad en cada circunstancia en que se estimen necesarias.

ADVERTENCIA ⚠	Identifica información sobre prácticas o circunstancias que pueden provocar una explosión en un ambiente peligroso, lo que podría causar lesiones personales o la muerte, daños materiales o pérdidas económicas.
IMPORTANTE	Identifica información importante para la correcta aplicación y comprensión del producto.
ATENCIÓN ⚠	Identifica información sobre prácticas o circunstancias que pueden provocar lesiones personales o la muerte, daños materiales o pérdidas económicas. Los mensajes de Atención le ayudan a identificar y evitar un peligro, y a reconocer las consecuencias.
PELIGRO DE CHOQUE ⚡	Puede haber etiquetas en el exterior o en el interior del equipo (por ejemplo, en un variador o un motor) para advertir sobre la posible presencia de un voltaje peligroso.
PELIGRO DE QUEMADURA ⚠	Puede haber etiquetas en el exterior o en el interior del equipo (por ejemplo, en un variador o un motor) para advertir sobre superficies que podrían estar a temperaturas peligrosas.

Dónde se encuentran las instrucciones

Use esta tabla de ubicación de instrucciones para encontrar detalles de referencia acerca de las instrucciones Logix (las instrucciones atenuadas en color gris están disponibles en otros manuales). Esta tabla de ubicación de instrucciones también lista los lenguajes de programación disponibles para las instrucciones

Si la tabla de localización indica	La instrucción se encuentra en
un número de página.	este manual.
General.	Logix5000 Controllers General Instructions Reference Manual, 1756-RM003.
Proceso.	Logix5000 Controllers Process Control and Drives Instructions Reference Manual, 1756-RM006.
Fase.	Logix5000 Controllers PhaseManager User Manual, LOGIX-UM001.

Instrucción	Lugar	Lenguajes
ABL ASCII Test For Buffer Line	general	lógica de escalera de relés texto estructurado
ABS Absolute Value	general	lógica de escalera de relés texto estructurado bloque de funciones
ACB ASCII Chars in Buffer	general	lógica de escalera de relés texto estructurado
ACL ASCII Clear Buffer	general	lógica de escalera de relés texto estructurado
ACOS Arc Cosine	general	texto estructurado
ACS Arc Cosine	general	lógica de escalera de relés bloque de funciones
ADD Add	general	lógica de escalera de relés texto estructurado bloque de funciones
AFI Always False Instruction	general	lógica de escalera de relés
AHL ASCII Handshake Lines	general	lógica de escalera de relés texto estructurado
ALM Alarm	proceso	texto estructurado bloque de funciones
ALMA Analog Alarm	general	lógica de escalera de relés texto estructurado bloque de funciones
ALMD Digital Alarm	general	lógica de escalera de relés texto estructurado bloque de funciones
AND Bitwise AND	general	lógica de escalera de relés texto estructurado bloque de funciones
ARD ASCII Read	general	lógica de escalera de relés texto estructurado
ARL ASCII Read Line	general	lógica de escalera de relés texto estructurado

Instrucción	Lugar	Lenguajes
ASIN Arc Sine	general	texto estructurado
ASN Arc Sine	general	lógica de escalera de relés bloque de funciones
ATAN Arc Tangent	general	texto estructurado
ATN Arc Tangent	general	lógica de escalera de relés bloque de funciones
AVE File Average	general	lógica de escalera de relés
AWA ASCII Write Append	general	lógica de escalera de relés texto estructurado
AWT ASCII Write	general	lógica de escalera de relés texto estructurado
BAND Boolean AND	general	texto estructurado bloque de funciones
BNOT Boolean NOT	general	texto estructurado bloque de funciones
BOR Boolean OR	general	texto estructurado bloque de funciones
BPT Break Points	general	lógica de escalera de relés
BRK Break	general	lógica de escalera de relés
BSL Bit Shift Left	general	lógica de escalera de relés
BSR Bit Shift Right	general	lógica de escalera de relés
BTD Bit Field Distribute	general	lógica de escalera de relés
BTDT Bit Field Distribute with Target	general	texto estructurado bloque de funciones

Instrucción	Lugar	Lenguajes
BTR Message	general	lógica de escalera de relés texto estructurado
BTW Mensaje	general	lógica de escalera de relés texto estructurado
BXOR O exclusivo booleano	general	texto estructurado bloque de funciones
CC Control coordinado	proceso	texto estructurado bloque de funciones
CLR Clear	general	lógica de escalera de relés texto estructurado
CMP Compare	general	lógica de escalera de relés
CONCAT String Concatenate	general	lógica de escalera de relés texto estructurado
COP Copy File	general	lógica de escalera de relés texto estructurado
COS Cosine	general	lógica de escalera de relés texto estructurado bloque de funciones
CPS Synchronous Copy File	general	lógica de escalera de relés texto estructurado
CPT Compute	general	lógica de escalera de relés
CTD Count Down	general	lógica de escalera de relés
CTU Count Up	general	lógica de escalera de relés
CTUD Count Up/Down	general	texto estructurado bloque de funciones
D2SD Discrete 2-State Device	proceso	texto estructurado bloque de funciones
D3SD Discrete 3-State Device	proceso	texto estructurado bloque de funciones
DDT Diagnostic Detect	general	lógica de escalera de relés
DEDT Deadtime	proceso	texto estructurado bloque de funciones
DEG Degrees	general	lógica de escalera de relés texto estructurado bloque de funciones
DELETE String Delete	general	lógica de escalera de relés texto estructurado
DERV Derivative	proceso	texto estructurado bloque de funciones
DFF D Flip-Flop	proceso	texto estructurado bloque de funciones
DIV Divide	general	lógica de escalera de relés texto estructurado bloque de funciones
DTOS DINT to String	general	lógica de escalera de relés texto estructurado
DTR Data Transitional	general	lógica de escalera de relés
EOT End of Transition	general	lógica de escalera de relés texto estructurado
EQU Equal to	general	lógica de escalera de relés texto estructurado bloque de funciones
ESEL Enhanced Select	proceso	texto estructurado bloque de funciones
EVENT Trigger Event Task	general	lógica de escalera de relés texto estructurado
FAL File Arithmetic and Logic	general	lógica de escalera de relés
FBC File Bit Comparison	general	lógica de escalera de relés
FFL FIFO Load	general	lógica de escalera de relés
FFU FIFO Unload	general	lógica de escalera de relés
FGEN Function Generator	proceso	texto estructurado bloque de funciones
FIND Find String	general	lógica de escalera de relés texto estructurado
FLL File Fill	general	lógica de escalera de relés
FOR For	general	lógica de escalera de relés
FRD Convert to Integer	general	lógica de escalera de relés bloque de funciones
FSC File Search and Compare	general	lógica de escalera de relés
GEQ Greater than or Equal to	general	lógica de escalera de relés texto estructurado bloque de funciones
GRT Greater Than	general	lógica de escalera de relés texto estructurado bloque de funciones
GSV Get System Value	general	lógica de escalera de relés texto estructurado
HLL High/Low Limit	proceso	texto estructurado bloque de funciones
HPF High Pass Filter	proceso	texto estructurado bloque de funciones

Instrucción	Lugar	Lenguajes
ICON Input Wire Connector	proceso	bloque de funciones
IMC Internal Model Control	proceso	texto estructurado bloque de funciones
INSERT Insert String	general	lógica de escalera de relés texto estructurado
INTG Integrator	proceso	texto estructurado bloque de funciones
IOT Immediate Output	general	lógica de escalera de relés texto estructurado
IREF Input Reference	proceso	bloque de funciones
JKFF JK Flip-Flop	proceso	texto estructurado bloque de funciones
JMP Jump to Label	general	lógica de escalera de relés
JSR Jump to Subroutine	general	lógica de escalera de relés texto estructurado bloque de funciones
JXR Jump to External Routine	general	lógica de escalera de relés
LBL Label	general	lógica de escalera de relés
LDL2 Second-Order Lead Lag	proceso	texto estructurado bloque de funciones
LDLG Lead-Lag	proceso	texto estructurado bloque de funciones
LEQ Less Than or Equal to	general	lógica de escalera de relés texto estructurado bloque de funciones
LES Less Than	general	lógica de escalera de relés texto estructurado bloque de funciones
LFL LIFO Load	general	lógica de escalera de relés
LFU LIFO Unload	general	lógica de escalera de relés
LIM Limit	general	lógica de escalera de relés bloque de funciones
LN Natural Log	general	lógica de escalera de relés texto estructurado bloque de funciones
LOG Log Base 10	general	lógica de escalera de relés texto estructurado bloque de funciones
LOWER Lower Case	general	lógica de escalera de relés texto estructurado
LPF Low Pass Filter	proceso	texto estructurado bloque de funciones

Instrucción	Lugar	Lenguajes
MAAT Motion Apply Axis Tuning	234	lógica de escalera de relés texto estructurado
MAFR Motion Axis Fault Reset	56	lógica de escalera de relés texto estructurado
MAG Motion Axis Gear	103	lógica de escalera de relés texto estructurado
MAH Motion Axis Home	75	lógica de escalera de relés texto estructurado
MAHD Motion Apply Hookup Diagnostics	245	lógica de escalera de relés texto estructurado
MAJ Motion Axis Jog	80	lógica de escalera de relés texto estructurado
MAM Motion Axis Move	91	lógica de escalera de relés texto estructurado
MAOC Motion Arm Output Cam	200	lógica de escalera de relés texto estructurado
MAPC Motion Axis Position Cam	132	lógica de escalera de relés texto estructurado
MAR Motion Arm Registration	191	lógica de escalera de relés texto estructurado
MASD Motion Axis Shutdown	45	lógica de escalera de relés texto estructurado
MAS Motion Axis Stop	60	lógica de escalera de relés texto estructurado
MASR Motion Axis Shutdown Reset	48	lógica de escalera de relés texto estructurado
MATC Motion Axis Time Cam	156	lógica de escalera de relés texto estructurado
MAVE Moving Average	proceso	texto estructurado bloque de funciones
MAW Motion Arm Watch	186	lógica de escalera de relés texto estructurado
MAXC Maximum Capture	proceso	texto estructurado bloque de funciones
MCD Motion Change Dynamics	114	lógica de escalera de relés texto estructurado
MCCD Motion Coordinated Change Dynamics	342	lógica de escalera de relés texto estructurado
MCCM Motion Coordinated Circular Move	299	lógica de escalera de relés texto estructurado
MCLM Motion Coordinated Linear Move	273	lógica de escalera de relés texto estructurado

Instrucción	Lugar	Lenguajes
MCCP Motion Calculate Cam Profile	126	lógica de escalera de relés texto estructurado
MCR Master Control Reset	general	lógica de escalera de relés
MCS Motion Coordinated Stop	353	lógica de escalera de relés texto estructurado
MCSD Motion Coordinated Shutdown	362	lógica de escalera de relés texto estructurado
MCSR Motion Coordinated Shutdown Reset	386	lógica de escalera de relés texto estructurado
MCSV Motion Calculate Slave Values	169	lógica de escalera de relés texto estructurado
MCT Motion Coordinated Transform	365	lógica de escalera de relés texto estructurado
MCTP Motion Calculate Transform Position	378	lógica de escalera de relés texto estructurado
MDF Motion Direct Drive Off	54	lógica de escalera de relés texto estructurado
MDOC Motion Disarm Output Cam	229	lógica de escalera de relés texto estructurado
MDO Motion Direct Drive On	51	lógica de escalera de relés texto estructurado
MDR Motion Disarm Registration	198	lógica de escalera de relés texto estructurado
MDW Motion Disarm Watch	189	lógica de escalera de relés texto estructurado
MEQ Mask Equal to	general	lógica de escalera de relés texto estructurado bloque de funciones
MGSD Motion Group Shutdown	177	lógica de escalera de relés texto estructurado
MGS Motion Group Stop	172	lógica de escalera de relés texto estructurado
MGSP Motion Group Strobe Position	182	lógica de escalera de relés texto estructurado
MGSR Motion Group Shutdown Reset	180	lógica de escalera de relés texto estructurado
MID Middle String	general	lógica de escalera de relés texto estructurado
MINC Minimum Capture	proceso	texto estructurado bloque de funciones

Instrucción	Lugar	Lenguajes
MMC Modular Multivariable Control	proceso	texto estructurado bloque de funciones
MOD Modulo	general	lógica de escalera de relés texto estructurado bloque de funciones
MOV Move	general	lógica de escalera de relés
MRAT Motion Run Axis Tuning	239	lógica de escalera de relés texto estructurado
MRHD Motion Run Hookup Diagnostics	249	lógica de escalera de relés texto estructurado
MRP Motion Redefine Position	121	lógica de escalera de relés texto estructurado
MSF Motion Servo Off	42	lógica de escalera de relés texto estructurado
MSG Message	general	lógica de escalera de relés texto estructurado
MSO Movimiento servo activado	39	lógica de escalera de relés texto estructurado
MSTD Desviación estándar de movimiento	proceso	texto estructurado bloque de funciones
MUL Multiplicar	general	lógica de escalera de relés texto estructurado bloque de funciones
MUX Multiplexor	proceso	bloque de funciones
MVM Mover con máscara	general	lógica de escalera de relés
MVMT Masked Move with Target	general	texto estructurado bloque de funciones
NEG Cambiar signo	general	lógica de escalera de relés texto estructurado bloque de funciones
NEQ Diferente a	general	lógica de escalera de relés texto estructurado bloque de funciones
NOP Sin operación	general	lógica de escalera de relés
NOT Bitwise NOT	general	lógica de escalera de relés texto estructurado bloque de funciones
NTCH Filtro de muesca	proceso	texto estructurado bloque de funciones
OCON Conector de cable de salida	proceso	bloque de funciones

Instrucción	Lugar	Lenguajes
ONS Un impulso	general	lógica de escalera de relés
OR Bitwise OR	general	lógica de escalera de relés texto estructurado bloque de funciones
OREF Referencia de salida	proceso	bloque de funciones
OSF Un impulso en flanco descendente	general	lógica de escalera de relés
OSFI One Shot Falling with Input	general	texto estructurado bloque de funciones
OSR Un impulso en flanco ascendente	general	lógica de escalera de relés
OSRI One Shot Rising with Input	general	texto estructurado bloque de funciones
OTE Activación de salida	general	lógica de escalera de relés
OTL Output Latch	general	lógica de escalera de relés
OTU Output Unlatch	general	lógica de escalera de relés
PATT Attach to Equipment Phase	fase	lógica de escalera de relés texto estructurado
PCLF Equipment Phase Clear Failure	fase	lógica de escalera de relés texto estructurado
PCMD Equipment Phase Command	fase	lógica de escalera de relés texto estructurado
PDET Detach from Equipment Phase	fase	lógica de escalera de relés texto estructurado
PFL Equipment Phase Failure	fase	lógica de escalera de relés texto estructurado
PI Proportional + Integral	proceso	texto estructurado bloque de funciones
PID Proportional Integral Derivative	general	lógica de escalera de relés texto estructurado
PIDE Enhanced PID	proceso	texto estructurado bloque de funciones
PMUL Pulse Multiplier	proceso	texto estructurado bloque de funciones
POSP Position Proportional	proceso	texto estructurado bloque de funciones
POVR Equipment Phase Override Command	fase	lógica de escalera de relés texto estructurado

Instrucción	Lugar	Lenguajes
PPD Equipment Phase Paused	fase	lógica de escalera de relés texto estructurado
PRNP Equipment Phase New Parameters	fase	lógica de escalera de relés texto estructurado
PSC Phase State Complete	fase	lógica de escalera de relés texto estructurado
PXRQ Equipment Phase External Request	fase	lógica de escalera de relés texto estructurado
RAD Radians	general	lógica de escalera de relés texto estructurado bloque de funciones
RES Reset	general	lógica de escalera de relés
RESD Reset Dominant	proceso	texto estructurado bloque de funciones
RET Return	general	lógica de escalera de relés texto estructurado bloque de funciones
RLIM Rate Limiter	proceso	texto estructurado bloque de funciones
RMPS Ramp/Soak	proceso	texto estructurado bloque de funciones
RTO Retentive Timer On	general	lógica de escalera de relés
RTOR Retentive Timer On with Reset	general	texto estructurado bloque de funciones
RTOS REAL to String	general	lógica de escalera de relés texto estructurado
SBR Subroutine	general	lógica de escalera de relés texto estructurado bloque de funciones
SCL Scale	proceso	texto estructurado bloque de funciones
SCRV S-Curve	proceso	texto estructurado bloque de funciones
SEL Select	proceso	bloque de funciones
SETD Set Dominant	proceso	texto estructurado bloque de funciones
SFP SFC Pause	general	lógica de escalera de relés texto estructurado
SFR SFC Reset	general	lógica de escalera de relés texto estructurado
SIN Sine	general	lógica de escalera de relés texto estructurado bloque de funciones

Instrucción	Lugar	Lenguajes
SIZE Size In Elements	general	lógica de escalera de relés texto estructurado
SNEG Selected Negate	proceso	texto estructurado bloque de funciones
SOC Second-Order Controller	proceso	texto estructurado bloque de funciones
SQI Sequencer Input	general	lógica de escalera de relés
SQL Sequencer Load	general	lógica de escalera de relés
SQO Sequencer Output	general	lógica de escalera de relés
SQR Square Root	general	lógica de escalera de relés bloque de funciones
SQRT Square Root	general	texto estructurado
SRT File Sort	general	lógica de escalera de relés texto estructurado
SRTP Split Range Time Proportional	proceso	texto estructurado bloque de funciones
SSUM Selected Summer	proceso	texto estructurado bloque de funciones
SSV Set System Value	general	lógica de escalera de relés texto estructurado
STD File Standard Deviation	general	lógica de escalera de relés
STOD String To DINT	general	lógica de escalera de relés texto estructurado
STOR String To REAL	general	lógica de escalera de relés texto estructurado
SUB Subtract	general	lógica de escalera de relés texto estructurado bloque de funciones
SWPB Swap Byte	general	lógica de escalera de relés texto estructurado
TAN Tangent	general	lógica de escalera de relés texto estructurado bloque de funciones
TND Temporary End	general	lógica de escalera de relés
TOD Convert to BCD	general	lógica de escalera de relés bloque de funciones
TOF Timer Off Delay	general	lógica de escalera de relés
TOFR Timer Off Delay with Reset	general	texto estructurado bloque de funciones

Instrucción	Lugar	Lenguajes
TON Timer On Delay	general	lógica de escalera de relés
TONR Timer On Delay with Reset	general	texto estructurado bloque de funciones
TOT Totalizer	proceso	texto estructurado bloque de funciones
TPT Tracepoints	general	lógica de escalera de relés
TRN Truncate	general	lógica de escalera de relés bloque de funciones
TRUNC Truncate	general	texto estructurado
UID User Interrupt Disable	general	lógica de escalera de relés texto estructurado
UIE User Interrupt Enable	general	lógica de escalera de relés texto estructurado
UPDN Up/Down Accumulator	proceso	texto estructurado bloque de funciones
UPPER Upper Case	general	lógica de escalera de relés texto estructurado
XIC Examine If Closed	general	lógica de escalera de relés
XIO Examine If Open	general	lógica de escalera de relés
XOR Bitwise Exclusive OR	general	lógica de escalera de relés texto estructurado bloque de funciones
XPY X to the Power of Y	general	lógica de escalera de relés texto estructurado bloque de funciones

Ubicación de instrucciones

Prefacio

Capítulo 1

Conceptos de movimiento

Capítulo 2

Instrucciones de estado de movimiento (MSO, MSF, MASD, MASR, MDO, MDF, MAFR)

Capítulo 3

Instrucciones que producen movimiento (MAS, MAH, MAJ, MAM, MAG, MCD, MRP, MCCP, MAPC, MATC, MCSV)

Capítulo 4

Instrucciones de grupo de movimiento (MGS, MGSD, MGSR, MGSP)

Capítulo 5

Instrucciones de evento de movimiento (MAW, MDW, MAR, MDR, MAOC, MDOC)

Capítulo 6

Instrucciones de configuración de movimiento (MAAT, MRAT, MAHD, MRHD)

Capítulo 7

Instrucciones de movimiento coordinado (MCLM, MCCM, MCCD, MCS, MCSD, MCT, MCTP, MCSR)

Capítulo 8

Ajustar un perfil de curva en S

Apéndice A

Códigos de error (ERR) para las instrucciones de control de movimiento

Apéndice B

Tipos de datos relacionados con el de movimiento (Estructuras)

Apéndice C

Programación de texto estructurado

Índice

Introducción

Este manual es uno de varios manuales de instrucciones relacionadas con el sistema Logix5000.

Tarea	Recurso
Programar el controlador para aplicaciones secuenciales	Logix5000 Controllers General Instructions Reference Manual, publicación 1756-RM003
Programar el controlador para las aplicaciones de proceso o variadores	Logix5000 Controllers Process Control and Drives Instructions Reference Manual, publicación 1756-RM006
Programar el controlador para aplicaciones de control de movimiento	El manual actual, es decir, Logix5000 Controllers Motion Instructions Reference Manual, publicación 1756-RM007
Programar el controlador para usar fases de equipo	PhaseManager User Manual, publicación LOGIX-UM001
Importar un archivo de texto o tags a un proyecto	Logix5000 Controllers Import/Export Reference Manual, publicación 1756-RM084
Exportar un proyecto o tags a un archivo de texto	
Configurar su controlador para ejes de movimiento y módulos de movimiento	Motion Configuration and Startup User Manual, publicación MOTION-UM001.
Configurar su controlador para sistema coordinado	Motion Coordinate System User Manual, publicación MOTION-UM002.

Puede usar estos controladores Logix5000 para control de movimiento:

- Controladores ControlLogix 1756
- Controladores GuardLogix 1756 (versión 16 y posterior)
- Controladores CompactLogix 1768 (versión 15 y posterior)
- Controladores SoftLogix5800 1789
- 20D PowerFlex 700S con controladores DriveLogix

Si usted tiene un variador PowerFlex 700S con controlador DriveLogix

No puede usar estas instrucciones con un controlador DriveLogix:

- Motion Direct Drive On (MDO)
- Motion Direct Drive Off (MDF)
- Motion Apply Axis Tuning (MAAT)
- Motion Run Axis Tuning (MRAT)
- Motion Apply Hookup Diagnostics (MAHD)
- Motion Run Hookup Diagnostics (MRHD)

Quién debe usar este manual

Este documento proporciona un programador con detalles acerca de las instrucciones de control de movimiento que se encuentran disponibles para un controlador Logix5000. El usuario ya debe estar familiarizado con la manera en que el controlador Logix5000 almacena y procesa datos.

Los programadores sin experiencia deben leer todos los detalles de una instrucción antes de usarla. Los programadores con experiencia pueden consultar la información sobre las instrucciones para verificar detalles.

Propósito de este manual

Este manual proporciona información acerca de cada instrucción de control de movimiento.

Tema	Proporciona este tipo de información
Nombre de la instrucción	Identifica la instrucción. Define si la instrucción es una instrucción de entrada o de salida.
Operandos	Enumera todos los operandos de la instrucción.
Texto estructurado	Describe el uso de operandos en el formato de texto estructurado.
Estructura de instrucciones de control de movimiento	Enumera los bits y valores de estado de control de la instrucción, si los hay.
Descripción	Describe el uso de la instrucción. Define las diferencias cuando la instrucción está habilitada e inhabilitada, si corresponde.
Indicadores de estado aritmético	Define si la instrucción afecta o no los indicadores de estado aritmético.
Condiciones de fallo	Define si la instrucción genera o no fallos menores o mayores. De ser así, define el tipo de fallo y el código.
Códigos de error	Indica y define los códigos de error correspondientes.
Bits de estado	Indica los bits de estado afectados, su estado y definiciones.
Ejemplo	Proporciona al menos un ejemplo de programación. Incluye una descripción que explica cada ejemplo.

Diagrama de función secuencial (SFC)

Un diagrama de función secuencial es un diagrama de flujo que controla su máquina o proceso. El SFC usa pasos y transiciones para realizar acciones u operaciones específicas. Puede usar el SFC para:

- Organizar la especificación funcional de su sistema.
- Programar y controlar su sistema como una serie de pasos y transiciones.

Al usar el diagrama de función secuencial (SFC), usted obtiene las siguientes ventajas.

- División gráfica de procesos en piezas de lógica mayores.
- Ejecución repetida más rápida de piezas individuales de su lógica.
- Una visualización en pantalla simplificada.
- Reducción del tiempo para diseñar y depurar su programa.
- La resolución de problemas es más rápida y fácil.
- Acceso directo al punto de la lógica en que la máquina entró en fallo.
- Más fácil de mejorar y actualizar.

Para obtener información más detallada acerca de cómo programar y usar un SFC, vea Logix5000 Controllers Sequential Function Charts Programming Manual, publicación 1756-PM006.

Convenciones y términos relacionados

Establecer y borrar

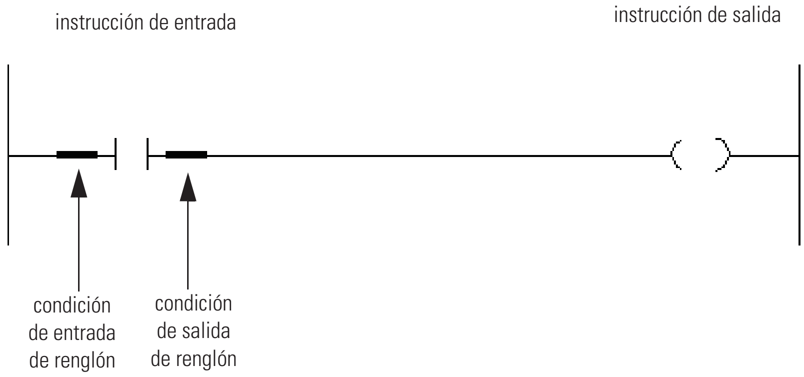
Este manual usa las funciones de establecer y restablecer para definir el estado de los bits (booleanos) y los valores (no booleanos):

Este término	Significa
establecer	el bit se establece en 1 (ACTIVADO) un valor se establece en cualquier número que no sea cero
borrar	el bit se borra a 0 (DESACTIVADO) todos los bits de un valor se restablecen a 0

Una instrucción se ejecuta más rápidamente y requiere menos memoria si todos los operandos de la instrucción usan el mismo tipo de datos óptimo, generalmente DIN o REAL.

Condición de renglón

El controlador evalúa las instrucciones de lógica de escalera según la condición de renglón que precede a la instrucción (condición de entrada del renglón). Según la condición de entrada de renglón y la instrucción, el controlador establece la condición de renglón que sigue la instrucción (condición de salida de renglón), lo cual, a su vez, afecta cualquier instrucción subsiguiente.



Si la condición de entrada de renglón de una instrucción de entrada es verdadera, el controlador evalúa la instrucción y establece la condición de renglón de entrada según los resultados de la instrucción. Si la instrucción se evalúa como verdadera, la condición de salida del renglón es verdadera; si la instrucción se evalúa como falsa, la condición de salida del renglón es falsa.

Recursos adicionales

Vea estos manuales y documentos para obtener más información acerca del uso de módulos de movimiento en un sistema de control Logix5000.

Publicación	Número de publicación
Logix5000 Controllers Quick Start	1756-QS001
Logix5000 Controllers Common Procedures	1756-PM001
Logix5000 Controller Motion Instructions Reference Manual	1756-RM007
Logix5000 Controllers General Instructions Reference Manual	1756-RM003
Logix5000 Controllers Process and Drives Instructions Reference Manual	1756-RM006
Motion Modules in Logix5000 Control Systems	LOGIX-UM002
Motion Planner Application Solution	RA-AP031
PhaseManager User Manual	LOGIX-UM001
ControlLogix Controller User Manual	1756-UM001
CompactLogix Controllers User Manual	1768-UM001
Analog Encoder (AE) Servo Module Installation Instructions	1756-IN047

Publicación	Número de publicación
ControlLogix SERCOS interface Module Installation Instructions	1756-IN572
CompactLogix SERCOS interface Module Installation Instructions	1768-IN005
1394 SERCOS Interface Multi Axis Motion Control System Installation Manual	1394-IN002
1394 SERCOS Integration Manual	1394-IN024
Ultra3000 Digital Servo Drives Installation Manual	2098-IN003
Ultra3000 Digital Servo Drives Integration Manual	2098-IN005
Kinetix 6000 Installation Manual	2094-IN001
Kinetix 6000 Integration Manual	2094-IN002
8720MC High Performance Drive Installation Manual	8720MC-IN001
8720MC High Performance Drive Integration Manual	8720MC-IN002

Notas:

Conceptos de movimiento

Introducción

Este capítulo trata los conceptos comunes a todas las instrucciones de control de movimiento.

Para esta información	Vea la página
Temporización de instrucciones	19
Programar un perfil de velocidad	24
Escoja un comando	33

Temporización de instrucciones

Las instrucciones de control de movimiento utilizan tres tipos de secuencias de temporización.

Tipo de temporización	Descripción
Inmediata	La instrucción se completa en un escán.
Mensaje	La instrucción se completa con varios escanes porque la instrucción envía mensajes al servomódulo.
Proceso	Completar la instrucción puede llevar una cantidad indefinida de tiempo.

Instrucciones de tipo inmediato

Las instrucciones de tipo inmediato de control de movimiento se ejecutan para la finalización en un escán. Si el controlador detecta un error durante la ejecución de estas instrucciones, se establece un bit de estado de error y finaliza la operación.

Los ejemplos de instrucciones de tipo inmediato incluyen:

- Instrucción Motion Change Dynamics (MCD)
- Instrucción Motion Group Strobe Position (MGSP)

Las instrucciones inmediatas funcionan de la siguiente manera:

1. Cuando el renglón que contiene la instrucción de control de movimiento se vuelve verdadero, el controlador:

   - Establece el bit Habilitar (.EN).
   - Borra el bit Listo (.DN).
   - Borra el bit Error (.ER).

   El controlador ejecuta la instrucción por completo.

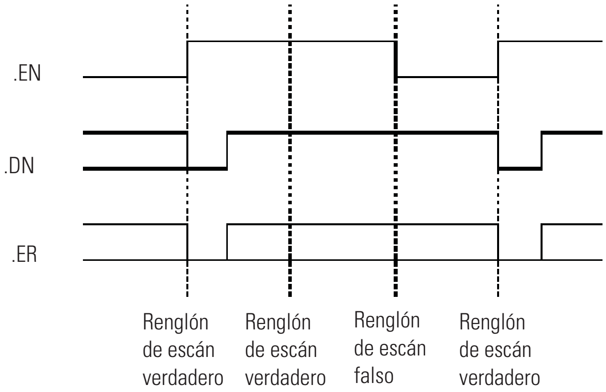
2.

Si el controlador	Entonces
No detecta un error cuando se ejecuta la instrucción	El controlador establece el bit .DN.
Detecta un error cuando se ejecuta la instrucción	El controlador establece el bit .ER y almacena un código de error en la estructura de control.

3. La próxima vez que el renglón se vuelva falso una vez establecido el bit .DN o el .ER, el controlador borra el bit .EN.

4. El controlador puede volver a ejecutar la instrucción cuando el renglón se vuelve verdadero.

Instrucciones de tipo inmediato – Condiciones del renglón



Instrucciones de tipo mensaje

Las instrucciones de control de movimiento de tipo mensaje envían uno o más mensajes al servomódulo.

Los ejemplos de instrucciones de tipo mensaje incluyen la:

- Instrucción Motion Direct Drive On (MDO)
- Instrucción Motion Redefine Position (MRP)

Las instrucciones de tipo mensaje funcionan de la siguiente manera:

1. Cuando el renglón que contiene la instrucción de control de movimiento se vuelve verdadero, el controlador:
   - Establece el bit Habilitar (.EN).
   - Borra el bit Listo (.DN).
   - Borra el bit Error (.ER).
2. El controlador comienza a ejecutar la instrucción configurando una solicitud de mensaje al servomódulo.

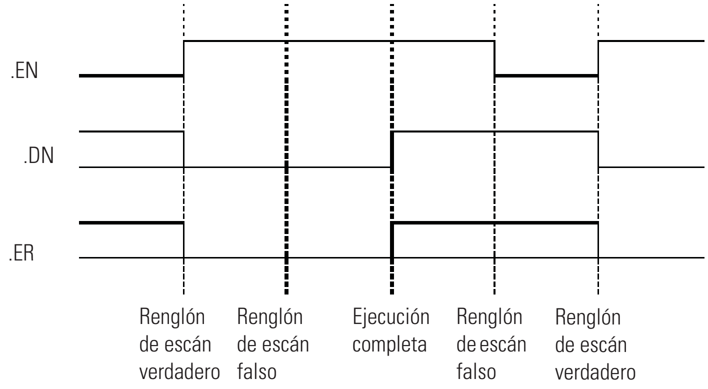
   El resto de la instrucción se ejecuta en paralelo con el escán del programa.
3. El controlador verifica si el servomódulo está listo para recibir un nuevo mensaje.
4. El controlador coloca los resultados de la verificación en la palabra de estado de mensaje de la estructura de control.
5. Cuando el módulo está listo, el controlador construye y transmite el mensaje al módulo.

   Este proceso se puede repetir varias veces si la instrucción requiere de múltiples mensajes.
6.

Si el controlador	Entonces
No detecta un error cuando se ejecuta la instrucción	El controlador establece el bit .DN si se han enviado todos los mensajes al módulo.
Detecta un error cuando se ejecuta la instrucción	El controlador establece el bit .ER y almacena un código de error en la estructura de control.

7. La próxima vez que el renglón se vuelve falso, una vez establecido el bit .DN o .ER, el controlador borra el bit .EN.
8. El controlador puede ejecutar la instrucción nuevamente cuando el renglón se vuelve verdadero.

Instrucciones de tipo mensaje – Condiciones del renglón



Instrucciones de tipo proceso

Las instrucciones de control de movimiento de tipo proceso inician procesos de control de movimiento que pueden tardar un tiempo indefinido para completarse.

Los ejemplos de instrucciones de tipo proceso incluyen la:

- Instrucción Motion Arm Watch Position (MAW)
- Instrucción Motion Axis Move (MAM)

Las instrucciones de tipo proceso funcionan de la siguiente manera:

1. Cuando el renglón que contiene la instrucción de control de movimiento se vuelve verdadero, el controlador:
    - Establece el bit Habilitar (.EN).
    - Borra el bit Listo (.DN).
    - Borra el bit Error (.ER).
    - Borra el bit Proceso Completo (.PC).
    - Establece el bit En Proceso (.IP).
2. El controlador inicia el proceso de movimiento.

3.

Si	Entonces el controlador
El controlador no detecta un error cuando se ejecuta la instrucción	· Establece el bit .DN. · Establece el bit En Proceso (.IP).
El controlador detecta un error cuando se ejecuta la instrucción	· Establece el bit .ER. · Almacena un código de error en la estructura de control.
El controlador detecta otra ocurrencia de la instrucción de control de movimiento	Borra el bit .IP para esa ocurrencia.
El proceso de control de movimiento llega al punto donde la instrucción puede volver a ejecutarse	Establece el bit .DN. Para algunas instrucciones de tipo proceso, tales como MAM, esto ocurrirá durante el primer escán. Para otras instrucciones, tales como MAH, el bit .DN no se establecerá hasta que se complete todo el proceso de vuelta a la posición inicial.
Ocurre uno de los eventos siguientes durante el proceso de control de movimiento: • Se completa el proceso de movimiento • Se ejecuta otra ocurrencia de la instrucción • Otra instrucción detiene el proceso de movimiento • Un fallo de movimiento detiene el proceso de movimiento	Borra el bit .IP.

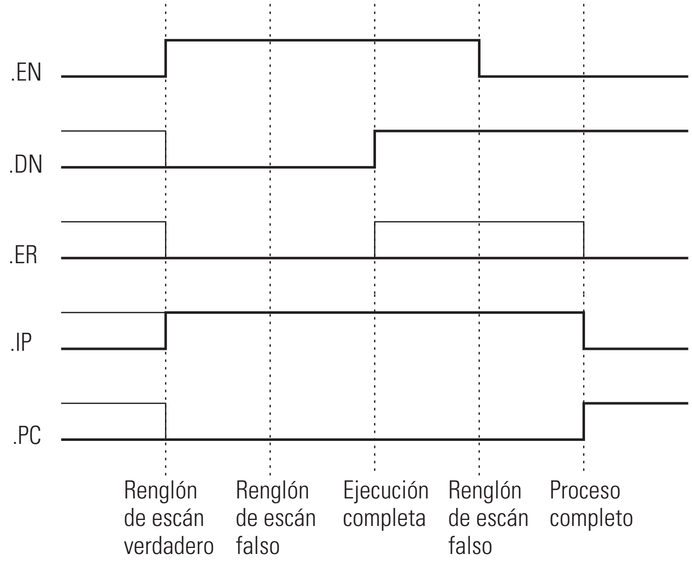
4. Luego del inicio del proceso de movimiento, el escán del programa puede continuar.

   El resto de la instrucción y el proceso de control continúan en paralelo con el escán del programa.

5. La próxima vez que el renglón se vuelve falso, una vez establecido el bit .DN o .ER, el controlador borra el bit .EN.

6. La instrucción puede ejecutarse nuevamente cuando el renglón se vuelve verdadero.

Instrucciones de tipo proceso – Condiciones del renglón



Programar un perfil de velocidad

Puede usar cualquiera de estos perfiles de movimiento para varias instrucciones:

- **Trapezoidal** perfil para la aceleración y desaceleración lineal
- **Curva en S** perfiles para el jaloneo controlado

Para obtener	Vea la página
Definición de jaloneo	24
Escoja un perfil	25
Use % de tiempo para facilitar la programación del jaloneo	26
Efectos del perfil de velocidad	27
Cálculo del régimen de jaloneo	28

Definición de jaloneo

Jaloneo es el régimen de cambio de aceleración o desaceleración.

Los parámetros de jaloneo sólo se aplican a los movimientos del perfil de curva en S utilizando estas instrucciones:

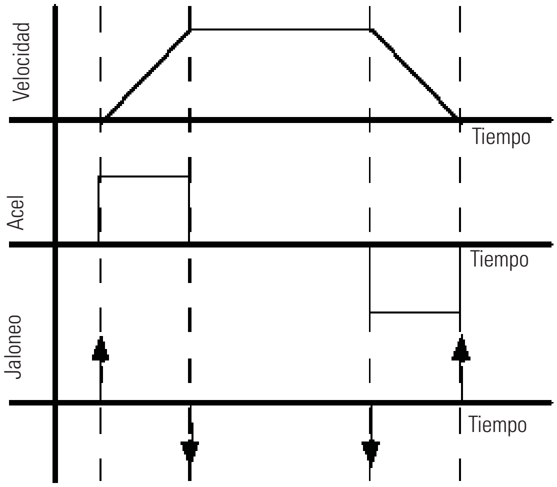
- MAJ
- MAM
- MAS
- MCD
- MCS
- MCCD
- MCCM
- MCLM

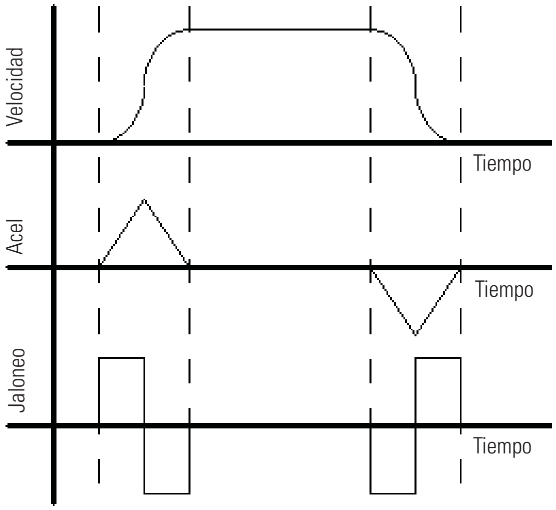
Ejemplo: Si la aceleración cambia de 0 a 40 mm/s^2 en 0.2 segundos, el jaloneo es de:

$(40 \text{ mm/s}^2 – 0 \text{ mm/s}^2)/0.2 \text{ s} = 200 \text{ mm/s}^3$

Escoja un perfil

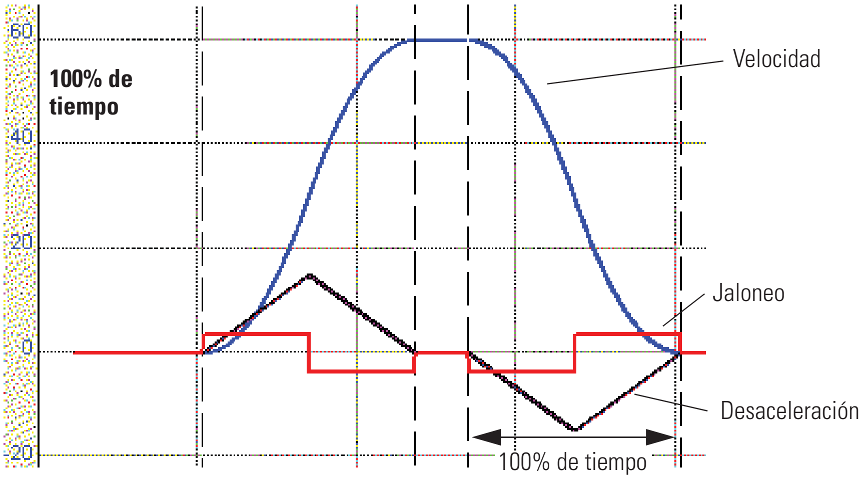
Cuando escoja un perfil, considere el tiempo de ciclo y la suavidad.

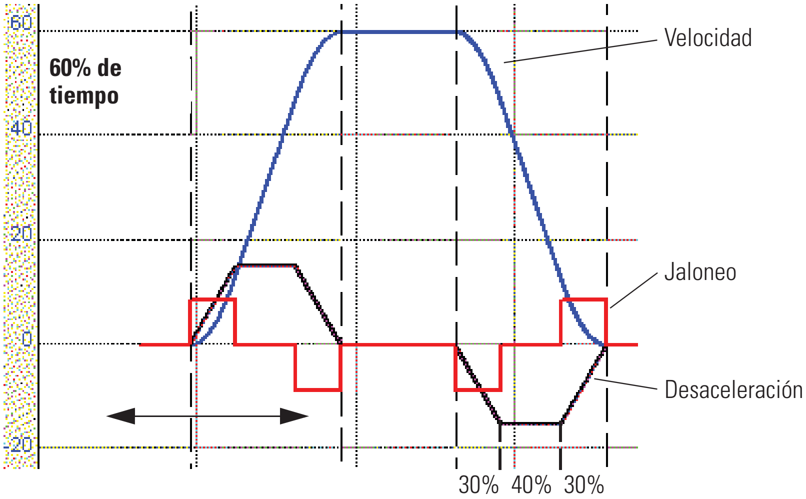
Si desea	Escoja este perfil	Consideración
• Tiempos más rápidos de aceleración y desaceleración • Mayor flexibilidad en la programación de movimientos subsiguientes	Trapezoidal 	El jaloneo **no** limita el tiempo de aceleración y desaceleración: • Las velocidades de aceleración y desaceleración controlan el cambio máximo en velocidad. • Su equipamiento y carga realizan más esfuerzo que con un perfil de curva en S. • El jaloneo es considerado infinito y se muestra en una línea vertical.

La aceleración y desaceleración más suave reducen el esfuerzo en el equipamiento y la carga	Curva en S  	El jaloneo limita el tiempo de aceleración y desaceleración: • Lleva más tiempo acelerar y desacelerar que un perfil trapezoidal. • Si la instrucción utiliza un perfil de curva en S, el controlador calcula la aceleración, la desaceleración y el jaloneo cuando inicia la instrucción. • El controlador calcula perfiles triangulares de aceleración y desaceleración.

Use % de tiempo para facilitar la programación del jaloneo

Use % de tiempo para especificar cuánto tiempo de aceleración o desaceleración tiene jaloneo. No tiene que calcular valores reales de jaloneo.

Ejemplo	Perfil
100% de tiempo	Al 100% del tiempo, la aceleración o desaceleración cambian el tiempo completo en el que el eje aumenta o disminuye la velocidad. 

Ejemplo	**Perfil**
60% de tiempo	Al 60% del tiempo, la aceleración o desaceleración cambian el tiempo en el que el eje aumenta o disminuye la velocidad. La aceleración o desaceleración es constante para el otro 40%. 

Efectos del perfil de velocidad

Esta tabla resume las diferencias entre perfiles:

Perfil	**ACEL/DESACEL**	**Motor**	**Prioridad de control**			
Tipo	Tiempo	Esfuerzo	**De más alta a más baja**			
Trapezoidal	Más rápida	Peor	Acel/Desacel	Velocidad	Posición	
Curva en S	2X más lento	Mejor	Jaloneo	Acel/Desacel	Velocidad	Posición

Cálculo del régimen de jaloneo

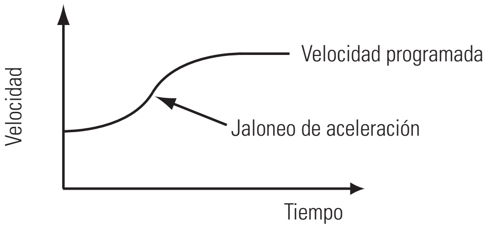
Si la instrucción utiliza o cambia un perfil de curva en S, el controlador calcula la aceleración, la desaceleración y el jaloneo cuando inicia la instrucción.

El sistema tiene un planificador de prioridad de jaloneo. En otras palabras, el jaloneo siempre tiene prioridad por sobre la aceleración y la velocidad. Por consiguiente, usted siempre obtiene el jaloneo programado. Si un movimiento tiene velocidad limitada, el movimiento no alcanza la aceleración y/o velocidad programada.

A los parámetros de jaloneo para MAJ programados en unidades de % de tiempo se los convierte en unidades de ingeniería del siguiente modo:

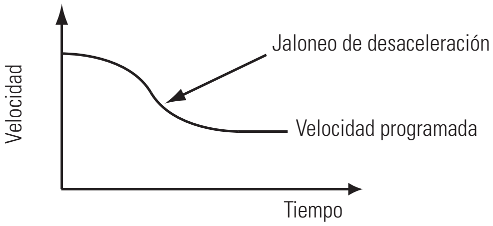
Si la velocidad de inicio < Velocidad MAJ programada

$$\text{Jaloneo de aceleración (Unidades/Seg}^3\text{)} = \frac{\text{Velocidad de aceleración programada}^2}{\text{Velocidad programada}} * \left(\frac{200}{\text{\% de tiempo}} - 1\right)$$



Si la velocidad de inicio > Velocidad MAJ programada

$$\text{Jaloneo de desaceleración (Unidades/Seg}^3\text{)} = \frac{\text{Velocidad de desaceleración programada}^2}{\text{Máx (Velocidad programada, [Velocidad de inicio – Velocidad programada])}} * \left(\frac{200}{\text{\% de tiempo}} - 1\right)$$

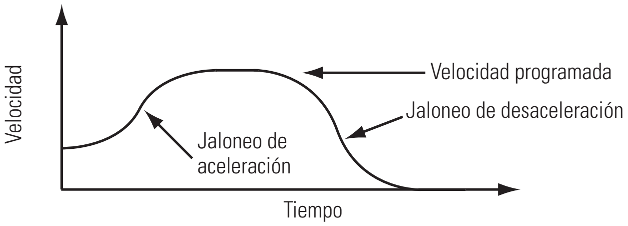


Los jaloneos para movimientos programados, como las instrucciones MAM o MCLM, en unidades de % de tiempo se convierten en unidades de ingeniería del siguiente modo:

Si la velocidad de inicio < Velocidad programada

$$\text{Jaloneo de aceleración (Unidades/Seg}^3\text{)} = \frac{\text{Velocidad de aceleración programada}^2}{\text{Velocidad programada}} * \left(\frac{200}{\text{\% de tiempo}} - 1\right)$$

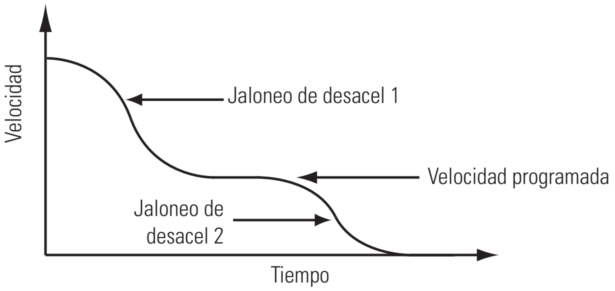
$$\text{Jaloneo de desaceleración (Unidades/Seg}^3\text{)} = \frac{\text{Velocidad de desaceleración programada}^2}{\text{Máx (Velocidad programada, [Velocidad de inicio – Velocidad programada])}} * \left(\frac{200}{\text{\% de tiempo}} - 1\right)$$



Si la velocidad de inicio > Velocidad programada

$$\text{Jaloneo de desacel 1} = \frac{\text{Velocidad de desaceleración programada}^2}{\text{Máx (Velocidad programada, [Velocidad de inicio – Velocidad programada])}} * \left(\frac{200}{\text{\% de tiempo}} - 1\right)$$

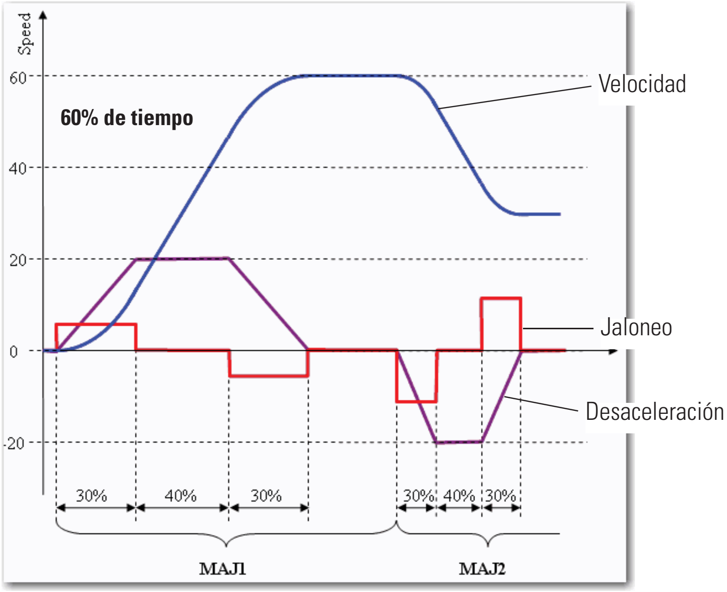
$$\text{Velocidad de desacel 2} = \frac{\text{Velocidad de desaceleración programada}^2}{\text{Velocidad programada}} * \left(\frac{200}{\text{\% de tiempo}} - 1\right)$$



El Jaloneo de desacel 1 se utiliza si la velocidad actual > Velocidad programada

El Jaloneo de desacel 2 se utiliza si la velocidad actual < Velocidad programada

Según el parámetro de la instrucción Speed, es posible el mismo jaloneo “% de tiempo” puede dar como resultado pendientes distintas para el perfil de aceleración y el perfil de desaceleración.



El algoritmo del planificador de movimiento ajusta el régimen de jaloneo real de modo que ambos perfiles, el de aceleración y el de desaceleración, contengan al menos el tiempo de rampa de “% de tiempo”. Si la velocidad de inicio es cercana al parámetro Speed programado, el porcentaje real del tiempo de rampa puede ser superior al valor programado.

En la mayoría de los casos, la condición es:

si: (la Velocidad de inicio es == 0.0) O
(la Velocidad de inicio es > 2 * Velocidad máx)

entonces: obtiene un porcentaje **programado** del tiempo de rampa

de otra manera: obtiene un valor **superior** al porcentaje **programado** de tiempo de rampa

Conversión de % de tiempo a Unidades de ingeniería

Si quiere convertir un % de Tiempo a Unidades de ingeniería, utilice estas ecuaciones.

Para jaloneo de acel:

$$j_a\,[\%\ \text{de tiempo}] = \frac{2}{1 + \dfrac{j_a\,[EU/s^3]\; v_{max}\,[EU/s]}{a_{max}\,[EU/s^2]}}\ 100$$

Para jaloneo de desacel:

$$j_d \text{ [\% de tiempo]} = \frac{2}{1 + \frac{j_d \text{ [EU/s}^3\text{] } v_{max} \text{ [EU/s]}}{d_{max} \text{ [EU/s}^2\text{]}}} 100$$

Programación de jaloneo en Unidades/Seg^3

Si desea especificar el jaloneo en “Unidades/segundos^3” en vez de en “% de tiempo”, ajuste su valor de jaloneo del siguiente modo para obtener el valor que usted ha programado:

$$\text{Velocidad temporal} = \frac{\text{Velocidad de desaceleración programada}^2}{\text{Valor de jaloneo de desaceleración deseado en Unidades/Seg}^3}$$

$$k = \frac{\text{Velocidad de inicio – Velocidad programada}}{\text{Máx (Velocidad programada, Velocidad temporal)}}$$

si (k < 1)

- Jaloneo de desacel de la plantilla de instrucción en Unidades/Seg^3 = Jaloneo de desacel deseado en Unidades/Seg^3

de otra manera

- Plantilla de instrucción del jaloneo de desaceleración en Unidades/Seg^3 = Jaloneo de desaceleración deseado en Unidades/Seg^3 * k

Consideraciones únicas del programa

Si programa un movimiento utilizando las unidades de *% de tiempo*, el software de programación RSLogix 5000 calcula un *jaloneo de aceleración* = a^2/v donde a = la *velocidad de aceleración* programada y v = *velocidad* programada. Por lo tanto, cuanto más alta sea la velocidad programada, más bajo será el jaloneo calculado. El sistema tiene un planificador de prioridad de jaloneo. En otras palabras, el jaloneo siempre tiene prioridad por sobre la aceleración y la velocidad. Por consiguiente, siempre obtiene el jaloneo programado. Si un movimiento tiene velocidad limitada, el movimiento no alcanza la aceleración y/o velocidad programada. Una vez que alcanza el límite de velocidad para la longitud del movimiento, como la velocidad ha aumentado, el movimiento demorará más tiempo en completarse.

El jaloneo de desacel se calcula de manera similar al jaloneo de aceleración descrito anteriormente. La única diferencia es que en vez de a^2/v, Jaloneo de desacel = d^2/v, donde d = la *velocidad de desacel* programada.

EJEMPLO

Ejemplo #1

Velocidad de inicio = 8.0 in/seg
Velocidad deseada = 5.0 in/seg

Velocidad de desaceleración deseada = 2.0 in/seg^2

Jaloneo de desaceleración deseado = 1.0 in/seg^3

Velocidad temporal = (Velocidad de desaceleración deseada)^2/
Valor de jaloneo deseado en Unidades/Seg^3 = 2.0^2/1.0 =
= 4.0 in/seg

k = (8.0 – 5.0)/máx(5.0, 4.0) = 3.0/5.0 =

= 0.6

Porque k < 1, podemos introducir el jaloneo de desaceleración deseado directamente en la plantilla

Plantilla de instrucción de jaloneo de desaceleración en Unidades/Seg^3 = 1.0 in/seg^3

EJEMPLO

Ejemplo #2

Velocidad de inicio = 13.0 in/seg
Velocidad deseada = 5.0 in/seg

Velocidad de desaceleración deseada = 2.0 in/seg^2

Jaloneo de desaceleración deseado = 1.0 in/seg^3

Velocidad temporal = (Velocidad de desaceleración deseada)^2/
Valor de jaloneo deseado en Unidades/Seg^3 = 2.0^2/1.0 =
= 4.0 in/seg

k = (13.0 – 5.0)/máx(5.0, 4.0) = 8.0/5.0 =

= 1.6

Porque k > 1, debemos calcular el jaloneo de desaceleración para usarlo en la plantilla de instrucción como:

> Jaloneo de desaceleración de la plantilla de instrucción en Unidades/Seg^3 =
>
> = 1.0 in/seg^3 * 1.6 =
>
> = 1.6 in/seg^3

¿Qué revisión tiene?

- 15 o anterior – % de tiempo se fija en 100.
- 16 o posterior – % de tiempo predeterminado en 100% de tiempo en proyectos convertidos desde versiones anteriores. Para nuevos proyectos, debe ingresar el valor de jaloneo.

Escoja un comando

Use esta tabla para escoger una instrucción y ver si está disponible como un comando directo de movimiento:

Si desea	**Y**	**Use esta instrucción**	**Comando directo de movimiento**
Cambiar el estado de un eje	Habilitar el servovariador y activar el lazo del servo del eje.	MSO Motion Servo On	sí
	Inhabilitar el servovariador y desactivar el lazo del servo del eje.	MSF Motion Servo Off	sí
	Forzar un eje a un estado de interrupción y bloquear cualquier instrucción que inicie este movimiento del eje.	MASD Motion Axis Shutdown	sí
	Cambiar un eje al estado listo. Si se quitan todos los ejes de un servomódulo del estado de interrupción como resultado de esta instrucción, los contactos OK de relé del módulo se cierran.	MASR Motion Axis Shutdown Reset	sí
	Habilitar el servovariador y establecer el voltaje de salida de un eje.	MDO Motion Direct Drive On	sí
	Inhabilitar el servovariador y establecer el voltaje de salida del servo en el voltaje de offset de salida.	MDF Motion Direct Drive Off	sí
	Borrar todos los fallos de movimiento para un eje.	MAFR Motion Axis Fault Reset	sí
Controlar la posición del eje	Detener cualquier proceso de movimiento de un eje.	MAS Motion Axis Stop	sí
	Volver el eje a la posición inicial.	MAH Motion Axis Home	sí
	Impulsar un eje.	MAJ Motion Axis Jog	sí
	Mover un eje a una posición específica.	MAM Motion Axis Move	sí
	Iniciar el engranaje electrónico entre 2 ejes.	MAG Motion Axis Gear	sí
	Cambiar la velocidad, aceleración, o desaceleración de un movimiento o impulso que está en progreso.	MCD Motion Change Dynamics	sí
	Cambiar el comando o la posición real de un eje.	MRP Motion Redefine Position	sí
	Calcular el perfil de una levas basado en una matriz de puntos de levas.	MCCP Perfil de leva de cálculo de movimiento	no
	Iniciar operaciones electrónicas de levas entre dos ejes.	MAPC Motion Axis Position Cam	no
	Iniciar operaciones electrónicas de levas en función del tiempo.	MATC Motion Axis Time Cam	no
	Calcular el valor esclavo, la pendiente, y la derivada de la pendiente para un perfil de levas y un valor de maestro.	MCSV Motion Calculate Slave Values	no

Si desea	Y	Use esta instrucción	Comando directo de movimiento
Iniciar la acción en todos los ejes	Detener el movimiento de todos los ejes.	MGS Motion Group Stop	sí
	Forzar todos los ejes al estado de interrupción.	MGSD Motion Group Shutdown	sí
	Cambiar todos los ejes al estado listo.	MGSR Motion Group Shutdown Reset	sí
	Enclavar el comando actual y la posición real de todos los ejes.	MGSP Motion Group Strobe Position	sí
Activar y desactivar las funciones de revisión de eventos especiales tales como registro y posición de control	Activar un evento en posición de control para verificar si hay un eje.	MAW Motion Arm Watch Position	sí
	Desactivar el evento en posición de control para determinar si hay un eje.	MDW Motion Disarm Watch Position	sí
	Activar el evento de registro del servomódulo para determinar si hay un eje.	MAR Motion Arm Registration	sí
	Desactivar el evento de registro del servomódulo para determinar si hay un eje.	MDR Motion Disarm Registration	sí
	Activar una leva de salida para un eje y una salida.	MAOC Motion Arm Output Cam	no
	Desactivar una o todas las levas de salida conectadas a un eje.	MDOC Motion Disarm Output Cam	no
Ajustar un eje y ejecutar pruebas de diagnóstico para su sistema de control. Estas pruebas incluyen: • Prueba de conexión motor/encoder • Prueba de conexión de encoder • Prueba de marcador	Use los resultados de la instrucción MAAT para calcular y actualizar las ganancias servo y los límites dinámicos de un eje.	MAAT Motion Apply Axis Tuning	no
	Ejecutar un perfil de ajuste de movimiento para un eje.	MRAT Motion Run Axis Tuning	no
	Use los resultados de una instrucción MRHD para establecer el encoder y las polaridades servo.	MAHD Motion Apply Hookup Diagnostic	no
	Ejecutar una de las pruebas de diagnóstico en el eje.	MRHD Motion Run Hookup Diagnostic	no

Si desea	**Y**	**Use esta instrucción**	**Comando directo de movimiento**
Control del movimiento coordinado de los multiejes	Iniciar un movimiento lineal coordinado para los ejes del sistema de coordinación.	MCLM Motion Coordinated Linear Move	no
	Iniciar un movimiento circular para los ejes del sistema de coordinación.	MCCM Motion Coordinated Circular Move	no
	Cambiar la dinámica de la ruta para el movimiento activo en un sistema coordinado.	MCCD Motion Coordinated Change Dynamics	no
	Detener los ejes de un sistema de coordinación o cancelar una transformación.	MCS Motion Coordinated Stop	no
	Interrumpir los ejes de un sistema de coordinación.	MCSD Motion Coordinated Shutdown	no
	Iniciar una transformación que vincula dos sistemas de coordinación juntos. Esto es como un engranaje bidireccional.	MCT Motion Coordinated Transform(1)	no
	Calcular la posición de un sistema de coordinación con respecto a otro sistema de coordinación.	MCTP Motion Calculate Transform Position(1)	no
	Cambiar los ejes de un sistema de coordinación a estado listo y borrar los fallos del eje.	MCSR Motion Coordinated Shutdown Reset	no

---

(1) Sólo puede usar estas instrucciones con los controladores 1756-L6x.

Notas:

Capítulo 2

Instrucciones de estado de movimiento

(MSO, MSF, MASD, MASR, MDO, MDF, MAFR)

ATENCIÓN ⚠ Las etiquetas usadas para el atributo de control de movimiento de las instrucciones sólo deben usarse una vez. La reutilización de la etiqueta de control de movimiento en otras instrucciones puede causar funcionamiento inesperado. Esto puede producir daños a maquinaria o lesiones personales.

Introducción

Las instrucciones de control de estado de movimiento controlan o cambian directamente los estados de operación de un eje. Las instrucciones de estado de movimiento son:

Si desea	Use esta instrucción	Disponible en estos lenguajes
Habilitar el servovariador y activar el lazo del servo del eje.	MSO	lógica de escalera de relés texto estructurado
Inhabilitar el servovariador y desactivar el lazo del servo del eje.	MSF	lógica de escalera de relés texto estructurado
Forzar un eje al estado de interrupción de operación. Una vez que el eje está en el estado de interrupción de operación, el controlador bloqueará cualquier instrucción que inicie movimiento del eje.	MASD	lógica de escalera de relés texto estructurado
Cambiar un eje de un estado existente de interrupción de funcionamiento a un estado de funcionamiento con eje listo. Si todos los ejes de un servomódulo se retiran del estado de interrupción como producto de esta instrucción, los contactos OK de relé para el módulo se cerrarán.	MASR	lógica de escalera de relés texto estructurado
Habilitar el servovariador y establecer el voltaje de salida del servo de un eje.	MDO	lógica de escalera de relés texto estructurado
Desactivar el servovariador y establecer el voltaje de salida del servo al voltaje de offset de salida.	MDF	lógica de escalera de relés texto estructurado
Borrar todos los fallos de movimiento de un eje.	MAFR	lógica de escalera de relés texto estructurado

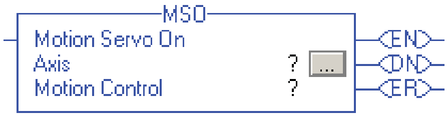
Los cinco estados de operación de un eje son:

Estado de operación	**Descripción**
Eje listo	Éste es el estado normal de encendido del eje. En este estado: • la salida de habilitación del variador del servomódulo está inactiva. • la acción del servo está inhabilitada. • no hay fallos del servo.
Control directo del variador	Este estado de funcionamiento permite al servomódulo DAC controlar directamente un variador externo. En este estado: • la salida de habilitación del variador del servomódulo está activa. • la acción del servo de posición está inhabilitada.
Control del servo	Este estado de funcionamiento permite al servomódulo realizar movimiento de lazo cerrado. En este estado: • la salida de habilitación del variador del servomódulo está activa. • la acción del servo está habilitada. • el eje está forzado a mantener la posición de servo comandada.
Eje en fallo	En este estado de funcionamiento, hay un fallo en el servo y el estado de la salida de habilitación del variador, la acción del servo y la condición del contacto OK dependen de los fallos y las acciones de fallo presentes.
Interrupción	Este estado de funcionamiento permite a los contactos de relé OK abrir un conjunto de contactos en la cadena de paro de emergencia de la fuente de alimentación del variador. En este estado: • la salida de habilitación del variador del servomódulo está inactiva. • la acción del servo está inhabilitada. • el contacto en buen estado está abierto.

Motion Servo On (MSO)

Use la instrucción MSO para activar el amplificador del variador para el eje especificado y activar el lazo de control del servo del eje.

Operandos: *Lógica de escalera de relés*



Operando	Tipo	Formato	Descripción
AXIS	AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje en el cual se realizará la operación.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura usada para obtener acceso a los parámetros de estado de instrucción.

Texto estructurado

```
MSO(Axis,MotionControl);
```

Los operandos son iguales a los de la instrucción de MSO de lógica de escaleras de relés.

Estructura de MOTION_INSTRUCTION

Mnemónico	Descripción
.EN (Habilitar) Bit 31	Se establece cuando el renglón hace una transición de falso a verdadero y permanece establecido hasta que la transacción de mensaje del servo se completa y el renglón pasa a falso.
Bit .DN (Listo) 29	Se establece cuando la acción del servo del eje ha sido habilitada correctamente y los bits Drive Enable y Servo Active Status se han establecido.
Bit .ER (Error) 28	Se establece para indicar que la instrucción ha detectado un error, como si se hubiese especificado un eje no configurado.

Descripción: La instrucción MSO activa directamente el variador y habilita los lazos del servo configurados asociados con un servoeje físico. Puede usarse en cualquier lugar en un programa, pero no debe usarse mientras el eje se mueve. Si se intenta esto, la instrucción MSO genera un error “Eje en movimiento”.

La instrucción MSO habilita automáticamente el eje especificado activando el variador y el lazo del servo asociado. El estado resultante del eje hace referencia al estado de control del servo.

El uso más común de esta instrucción es activar el lazo del servo para el eje especificado en su posición actual como preparación para movimiento de comando.

Para ejecutar correctamente una instrucción MSO, el eje especificado debe estar configurado como servoeje. Si no se cumple esta condición, la instrucción dará error.

IMPORTANTE La ejecución de la instrucción MSO puede necesitar múltiples escanes porque requiere la transmisión de un mensaje al módulo y tiempo de movimiento para que la salida del variador se estabilice y el lazo del servo se active. El bit Listo (.DN) no se establece inmediatamente, sino luego de que el eje esté en el estado de control de servo.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condiciona la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Consulte el documento Códigos de error (ERR) para las instrucciones de control de movimiento.

Códigos de error extendidos: Los códigos de error extendidos proporcionan información adicional específica de instrucciones para los códigos de error que son genéricos a muchas instrucciones. Los siguientes códigos de error extendidos ayudan a localizar el problema cuando la instrucción MSO recibe un mensaje de error Fallo de mensajes al servomódulo (12).

Código de error extendido (decimal)	Código de error asociado (decimal)	Significado
Conflicto del modo de objeto (12)	SERVO_MESSAGE_FAILURE (12)	El eje está en interrupción.
Permiso denegado (15)	SERVO_MESSAGE_FAILURE (12)	Error al habilitar el interruptor de entrada. (SERCOS)
Dispositivo en estado equivocado (16)	SERVO_MESSAGE_FAILURE (12)	El estado del dispositivo no es el correcto para la acción. (SERCOS)

MSO cambia a bits de estado:

Bits de estado de eje

Nombre del bit	Estado	Significado
ServoActStatus	VERDADERO	El eje está en estado de control del servo con el lazo del servo activo.
DriveEnableStatus	VERDADERO	La salida de habilitación del variador del eje está activa.

Bits de estado de movimiento

Ninguno

Ejemplo:

Cuando las condiciones de entrada son verdaderas, el controlador habilita el servovariador y activa el lazo del servo de eje configurado por *eje1*.

Lógica de escalera de relés

MSO
Motion Servo On
Axis Axis1
Motion Control MSO_1

Ejemplo de lógica de escalera de MSO

Texto estructurado

MSO(Axis0,MSO_1);

Motion Servo Off (MSF)

Use la instrucción MSF para desactivar la salida del variador para el eje especificado y desactivar el lazo del servo del eje.

IMPORTANTE Si ejecuta una instrucción MSF mientras el eje está en movimiento, el eje realiza por inercia un paro no controlado.

Operandos:

Lógica de escalera de relés

Operando	Tipo	Formato	Descripción
Axis	AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje en el cual se realizará la acción.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura usada para obtener acceso a los parámetros de estado de instrucción.

Texto estructurado

MSF(Axis,MotionControl);

Los operandos son iguales a los de la instrucción MSF de lógica de escalera de relés.

Estructura MOTION_INSTRUCTION

Mnemónico	Descripción
Bit .EN (Habilitar) 31	Se establece cuando el renglón hace una transición de falso a verdadero y permanece establecido hasta que la transacción de mensaje del servo se completa y el renglón pasa a falso.
Bit .DN (Listo) 29	Se establece cuando la acción del servo del eje ha sido inhabilitada correctamente y los bits Drive Enable y Servo Active Status se han borrado.
Bit .ER (Error) 28	Se establece para indicar que la instrucción ha detectado un error, como si se hubiese especificado un eje no configurado.

Descripción:

La instrucción MSF desactiva directa e inmediatamente la salida del variador e inhabilita el lazo del servo en cualquier servoeje físico. Esto coloca al eje en el estado de eje listo. La instrucción MSF también inhabilita cualquier planificador de movimiento que pueda estar activo al momento de ejecución. La instrucción MSF no requiere parámetros – simplemente introduzca o seleccione el eje deseado.

Si el eje especificado no aparece en la lista de ejes disponibles, el eje no se ha configurado para operación. Use el editor de tags para crear y configurar un eje nuevo.

Puede usar la instrucción MSF para **DESACTIVAR** la acción del servo cuando deba mover el eje a mano. Ya que la posición se continúa siguiendo incluso cuando la acción del servo está DESACTIVADA. Cuando el lazo del servo está ACTIVO nuevamente, mediante la instrucción MSO, el eje está de nuevo bajo control de lazo cerrado en la posición nueva.

El comportamiento de paro del eje varía según el tipo de variador. En algunos casos, el eje realiza por inercia un paro y en otros casos el eje desacelera hasta parar usando el par de paro disponible del variador.

Para ejecutar correctamente una instrucción MSF, el eje especificado debe estar configurado como un servoeje. Si no se cumple esta condición, la instrucción dará error. Si tiene un tipo de eje virtual, las instrucciones dan error porque, con un eje virtual, la acción del servo y el estado de habilitación del variador están forzados a ser siempre verdaderos. Un tipo de datos de eje consumido también da error porque sólo el controlador que produce puede cambiar el estado de un eje consumido.

IMPORTANTE La ejecución de la instrucción MSF puede necesitar múltiples escanes porque requiere la transmisión de un mensaje al módulo de movimiento y tiempo para que la salida del variador y el lazo del servo se desactiven completamente. El bit Listo (.DN) no se establece si este mensaje no se transmite correctamente y el eje hace una transición al estado eje listo.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Vea Códigos de error (ERR) para las instrucciones de control de movimiento.

MSF cambia a bits de estado:

Bits de estado de eje

Nombre del bit	Estado	Significado
ServoActionStatus	FALSO	El eje está en estado de servo activado con el lazo del servo activo.
DecelStatus	FALSO	La salida de habilitación del variador del eje está activa.

Bits de estado de movimiento

Nombre del bit	Estado	Significado
AccelStatus	FALSO	El eje no acelera.
DecelStatus	FALSO	El eje no desacelera.
MoveStatus	FALSO	El eje no se mueve.
JogStatus	FALSO	El eje no se impulsa.
GearingStatus	FALSO	El eje no se acopla.
HomingStatus	FALSO	El eje no vuelve a la posición inicial.
StoppingStatus	FALSO	El eje no se detiene.
PositionCamStatus	FALSO	El eje no realiza la operación de levas de posición.
TimeCamStatus	FALSO	El eje no realiza la operación de levas de tiempo.
PositionCamPendingStatus	FALSO	El eje no tiene pendiente una operación de levas de posición.
TimeCamPendingStatus	FALSO	El eje no tiene pendiente una operación de levas de tiempo.
GearingLockStatus	FALSO	El eje no está en una condición de engranaje bloqueado.
PositionCamLockStatus	FALSO	El eje no está en una condición de operación de levas bloqueadas.

Ejemplo: Cuando las condiciones de entrada son verdaderas, el controlador inhabilita el servovariador y el lazo del servoeje configurado por *Eje0*.

Lógica de escalera de relés

Ejemplo de lógica de escalera de MSF

Texto estructurado

MSF(Axis0,MSF_1);

Motion Axis Shutdown (MASD)

Use la instrucción MASD para forzar un eje especificado al estado de interrupción. El estado de interrupción de un eje es la condición en la cual la salida del variador está inhabilitada, el lazo del servo desactivado y todos los contactos OK de relé de estado sólido disponibles o asociados están abiertos. El eje permanecerá en estado de interrupción hasta que se ejecute un restablecimiento de interrupción de grupo o de eje.

Operandos:

Lógica de escalera de relés

Operando	Tipo	Formato	Descripción
AXIS	AXIS_FEEDBACK AXIS_VIRTUAL AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje en el cual se realizará la operación.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura usada para obtener acceso a los parámetros de estado de instrucción.

Texto estructurado

MASD(Axis,MotionControl);

Los operandos son iguales a los de la instrucción MASD de lógica de escalera de relés.

Estructura MOTION_INSTRUCTION

Mnemónico	Descripción
Bit .EN (Habilitar) 31	Se establece cuando el renglón hace una transición de falso a verdadero y permanece establecido hasta que la transacción de mensaje del servo se completa y el renglón pasa a falso.
Bit .DN (Listo) 29	Se establece cuando el eje se ha establecido correctamente al estado de interrupción.
Bit .ER (Error) 28	Se establece para indicar que la instrucción detectó un error, como por ejemplo si usted especifica un eje no configurado.

Descripción:

La instrucción MASD inhabilita directa e inmediatamente la salida del variador, inhabilita el lazo del servo y abre todos los contactos OK asociados. Esta acción coloca al eje en estado de interrupción.

Otra acción iniciada por la instrucción MASD es el borrado de todos los procesos de movimiento en progreso y el borrado de todos los bits de estado de movimiento. Asociado con esta acción, el comando también borra todos los bits IP de instrucción de movimiento que se encuentran establecidos actualmente para el eje especificado.

La instrucción MASD fuerza al eje especificado al estado de interrupción. Una de las características únicas del estado de interrupción es que, cuando se encuentra disponible, el contacto OK de relé en estado sólido para el módulo de movimiento o variador está abierto. Esta función puede usarse para abrir la cadena de paro de emergencia que controla la alimentación principal del sistema del variador. Tenga en cuenta que normalmente hay un sólo contacto OK por módulo de movimiento, lo que significa que la ejecución de una instrucción MASD para cualquier eje asociado con un módulo dado abre el contacto OK.

Otra característica del estado de interrupción es que cualquier instrucción que inicia movimiento de eje tiene la ejecución bloqueada. Si se intenta, se produce un error de ejecución. Sólo ejecutando una de las instrucciones de restablecimiento de interrupción puede iniciarse correctamente el movimiento.

Para ejecutar correctamente una instrucción MASD, el eje especificado debe estar configurado como un servoeje o eje de sólo retroalimentación. Si no, la instrucción dará error.

El eje permanece en el estado de interrupción hasta que se ejecuta una instrucción MASR o una instrucción MGSR.

IMPORTANTE La ejecución de la instrucción MASD puede necesitar múltiples escanes porque requiere la transmisión de un mensaje al módulo de movimiento. De este modo, el bit Listo (.DN) no se establece si este mensaje no es trasmitido correctamente y el eje está en el estado de interrupción.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Vea Códigos de error (ERR) para las instrucciones de control de movimiento.

MASD cambia a bits de estado:

Bits de estado de eje

Nombre del bit	**Estado**	**Significado**
ServoActStatus	FALSO	· El eje está en el estado de eje listo. · El lazo del servo está inactivo.
DriveEnableStatus	FALSO	La salida de habilitación del variador está inactiva.
ShutdownStatus	VERDADERO	El eje está en el estado de interrupción.

Bits de estado de movimiento

Nombre del bit	**Estado**	**Significado**
AccelStatus	FALSO	El eje no acelera.
DecelStatus	FALSO	El eje no desacelera.
MoveStatus	FALSO	El eje no se mueve.
JogStatus	FALSO	El eje no se impulsa.
GearingStatus	FALSO	El eje no se acopla.
HomingStatus	FALSO	El eje no vuelve a la posición inicial.
StoppingStatus	FALSO	El eje no se detiene.
PositionCamStatus	FALSO	El eje no realiza la operación de levas de posición.
TimeCamStatus	FALSO	El eje no realiza la operación de levas de tiempo.
PositionCamPendingStatus	FALSO	El eje no tiene pendiente una operación de levas de posición.
TimeCamPendingStatus	FALSO	El eje no tiene pendiente una operación de levas de tiempo.
GearingLockStatus	FALSO	El eje no está en una condición de engranaje bloqueado.
PositionCamLockStatus	FALSO	El eje no está en una condición de levas bloqueadas.

Ejemplo:

Cuando las condiciones de entrada son verdaderas, el controlador fuerza al *eje1* al estado de operación de interrupción.

Lógica de escalera de relés

Axis Axis0
Motion Control MASD_2

Ejemplo de lógica de escaleras de MASD

Texto estructurado

MASD(Axis0,MASD_1);

Motion Axis Shutdown Reset (MASR)

Use la instrucción MASR para hacer una transición de un eje desde un estado de interrupción existente a un estado de eje listo. Todos los fallos asociados con el eje especificado se borran automáticamente. Si, como resultado de esta instrucción, todos los ejes del módulo de movimiento asociado ya no se encuentran en estado de interrupción, los contactos OK de relé para el módulo se cierran.

Operandos: *Lógica de escalera de relés*

Operando	Tipo	Formato	Descripción
AXIS	AXIS_FEEDBACK AXIS_VIRTUAL AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje en el cual se realizará la operación.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura usada para obtener acceso a los parámetros de estado de instrucción.

Texto estructurado

MASR(Axis,MotionControl);

Los operandos son iguales a los de la instrucción MASR de lógica de escalera de relés.

Estructura MOTION_STRUCTURE

Mnemónico	Descripción
Bit .EN (Habilitar) 31	Se establece cuando el renglón hace una transición de falso a verdadero y permanece establecido hasta que la transacción de mensaje del servo se completa y el renglón pasa a falso.
Bit .DN (Listo) 29	Se establece cuando el eje se ha restablecido correctamente del estado de interrupción.
Bit .ER (Error) 28	Se establece para indicar que la instrucción ha detectado un error, como si se hubiese especificado un eje no configurado.

Descripción: La instrucción MASR borra todos los fallos de eje y saca al eje especificado del estado de interrupción. Si el módulo de movimiento es compatible con un contacto OK, y ningún otro eje de módulo está en el estado de interrupción, la instrucción MASR produce el cierre del contacto OK de relé en estado sólido del módulo. Independientemente de la condición del contacto OK, la ejecución de MASR coloca al eje en el estado de eje listo.

De la misma manera en que la instrucción MASD fuerza al eje especificado al estado de interrupción, la instrucción MASR saca al eje del estado de interrupción y lo coloca en el estado de eje listo. Una de las características únicas del estado de interrupción es que cualquier contacto OK de relé en estado sólido asociado para el módulo de movimiento está abierto. Si, como resultado de una instrucción MASR, no hay ejes asociados con un módulo de movimiento dado en estado de interrupción, los contactos OK de relé se cierran como resultado de MASR. Esta función puede usarse para cerrar la cadena de paro de emergencia que controla la alimentación principal del sistema del variador y, de este modo, permite al usuario realimentar el variador. Tenga en cuenta que normalmente hay un sólo contacto OK por módulo de movimiento, lo que significa que la ejecución de la instrucción MASR puede ser necesaria para todos los ejes asociados con un módulo dado para que se cierre el contacto OK.

Para ejecutar correctamente una instrucción MASR, el eje seleccionado debe estar configurado como un servoeje o de sólo retroalimentación. En caso contrario, la instrucción dará error.

La instrucción MASR es un comando de tipo procedimiento que se procesa desde el controlador Logix, a través del módulo SERCOS, y hacia los variadores asociados. Para aplicaciones que usan un controlador Logix con RSLogix 5000, versión 13 y anterior, el bit .DN de la instrucción se estableció cuando el módulo SERCOS confirmó el pedido de procedimiento. Los variadores confirmaron el comando y lo procesaron rápidamente. Para el usuario, la operación del bit .DN parece indicar la finalización correcta del borrado de fallos casi inmediatamente.

Para aplicaciones que usan un controlador Logix con RSLogix 5000, versión 15 y posterior, las versiones actuales de firmware del variador han incorporado una nueva parte del mecanismo de borrado de fallos que activa un proceso de recuperación de posición en el variador que recalcula la conmutación. Este recálculo puede hacer que el variador que no esté en un estado listo pase a servo activado durante 0.5 a 5 segundos después de que se realizó el restablecimiento. Para reflejar este cambio en la instrucción, la operación de bit .DN se cambió para esperar a que el variador complete el procedimiento de restablecimiento. El resultado final es que usted no verá una finalización correcta del borrado de fallos casi inmediatamente como quizás lo hacía en versiones anteriores.

IMPORTANTE La ejecución de la instrucción MASR puede necesitar múltiples escanes porque requiere la transmisión de un mensaje al módulo de movimiento. De este modo, el bit Listo (.DN) no se establece si el mensaje no ha sido transmitido correctamente.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Vea Códigos de error (ERR) para las instrucciones de control de movimiento.

Bits de estado:

Nombre del bit	Estado	Significado
ShutdownStatus	FALSO	El eje no está en el estado de interrupción.

Ejemplo: Cuando las condiciones de entrada son verdaderas, el controlador restablece el *eje1* desde un estado previo de operación de interrupción a un estado de operación de eje listo.

Lógica de escalera de relés

MASR
Motion Axis Shutdown Reset
Axis Axis2
Motion Control MASR_1

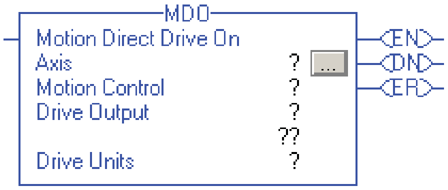
Texto estructurado

MASR(Axis0,MASR_1);

Motion Direct Drive On (MDO)

Use la instrucción MDO en combinación con módulos de movimiento compatibles con una interface del servovariador analógico externo, por ejemplo el servomódulo 1756-M02AE o 1784-PM02AE. Esta instrucción activa la habilitación del variador del módulo, lo cual habilita el servovariador externo, y también establece el voltaje de salida del servomódulo del variador al nivel de voltaje especificado. El valor para la salida del variador puede especificarse en voltios o % del límite máximo de salida del eje.

Operandos: *Lógica de escalera de relés*



Operando	Tipo de datos	Descripción
AXIS	tag	Nombre del eje en el cual se realizará la operación.
Motion Control	Tag MOTION_ INSTRUCTION	Estructura usada para obtener acceso a los parámetros de estado de instrucción.
Drive Output	REAL	Voltaje a la salida en % del límite de salida del servo o en voltios
Drive Units	Boolean	Unidades en la cuales se interpreta el valor de la salida del variador.

Texto estructurado

```
MDO(Axis,MotionControl,
DriveOutput,DriveUnits);
```

Los operandos son iguales a los de la instrucción MDO de lógica de escalera de relés.

En el caso de los operandos para los que usted debe seleccionar entre opciones disponibles, introduzca su selección como:

Este operando	Tiene estas opciones que usted...	
	introduce como texto	**o introduce como un número**
DriveUnits	volts	0
	percent	1

Estructura MOTION_STRUCTURE

Mnemónico	Descripción
Bit .EN (Enable) 31	Se establece cuando el renglón hace una transición de falso a verdadero y permanece establecido hasta que la transacción de mensaje del servo se completa y el renglón pasa a falso.
Bit .DN (Listo) 29	Se establece cuando el bit de habilitación del variador del eje está activado y la salida analógica especificada está conectada correctamente.
Bit .ER (Error) 28	Se establece para indicar que la instrucción detectó un error, como si se hubiera introducido un valor de salida del variador demasiado largo.

Descripción: Para módulos de movimiento con una interface de servovariador externo, como 1756-M02AE o 1784-PM02AE, la instrucción MDO puede usarse para habilitar directamente la salida de habilitación del variador del eje y establecer la salida analógica a un nivel especificado determinado por el parámetro de salida del variador. El parámetro de salida del variador se puede expresar como un voltaje o como un porcentaje del valor de voltaje de salida configurado máximo dado por el atributo del límite de salida.

La instrucción MDO puede usarse sólo en un eje físico cuyo tipo de eje esté configurado para servo. La instrucción sólo se ejecuta cuando el eje está en el estado de eje listo (por ejemplo, la acción del servo está DESACTIVADA). El estado resultante del eje se conoce como estado de control del variador.

La instrucción MDO habilita automáticamente el eje especificado activando la salida de habilitación del variador adecuada antes de establecer la salida analógica del servomódulo en el valor de voltaje especificado. Normalmente, hay un retardo de 500 mseg entre la activación de la salida de habilitación del variador y la configuración de la salida analógica al nivel especificado para permitir que la estructura de alimentación del variador se estabilice. Para minimizar la deriva durante este retardo de habilitación del variador, el voltaje de salida del variador se establece en el valor de atributo de offset de salida (el valor predeterminado es cero). A partir de entonces, el voltaje de salida está dado por el valor especificado de salida del variador de la instrucción MDO e indicado por el valor de atributo de estado de la salida del servo.

El hardware DAC de 16 bits asociado con varios servomódulos Logix limita la resolución efectiva del control de movimiento del variador directo a 305 µV o 0.003%. En el caso del funcionamiento de variador directo, el lazo del servo del módulo está inactivo y es evitado. La instrucción MDO sólo es afectada por el bit de configuración Servo Output Polarity, el offset de salida y los atributos de límite de salida para el eje. En el caso en que el valor de configuración del límite de salida se reduce por debajo del valor de voltaje de salida de corriente, el valor de salida del servo se fija automáticamente en el valor del límite de salida.

El uso más común de esta instrucción es para proporcionar una salida analógica programable independiente como una referencia de velocidad de lazo abierto para un variador externo o para probar un servovariador externo para el funcionamiento de lazo cerrado.

Para ejecutar correctamente una instrucción MDO, el eje especificado debe estar configurado como servoeje y estar en el estado de eje listo, con la acción del servo desactivada. Si no se cumplen estas condiciones, la instrucción dará error.

IMPORTANTE La ejecución de la instrucción MDO puede necesitar escanes múltiples porque requiere la transmisión de un mensaje al módulo de movimiento y tiempo para que la salida del variador se estabilice. El bit Listo (.DN) no se establece si el eje no está en el estado de control del variador.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Vea Códigos de error (ERR) para las instrucciones de control de movimiento.

Códigos de error extendidos: Los códigos de error extendidos proporcionan información adicional específica de instrucción para los códigos de error que son genéricos a muchas instrucciones. Los siguientes códigos de error extendidos ayudan a localizar el problema cuando la instrucción MDO recibe un mensaje de error de fallo de mensajes al servomódulo (12).

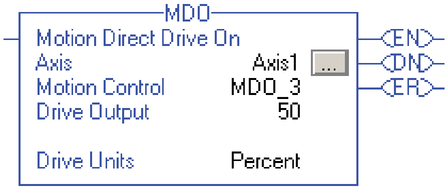
Código de error extendido (decimal)	Código de error asociado (decimal)	Significado
Conflicto del modo de objeto (12)	SERVO_MESSAGE_FAILURE (12)	El eje está en interrupción.

Bits de estado: *MDO cambia a bits de estado*

Nombre del bit	Estado	Significado
DriveEnableStatus	VERDADERO	El eje está en estado de control del variador con la salida de habilitación del variador activa.

Ejemplo: Cuando las condiciones de entrada son verdaderas, el controlador activa el servovariador para *eje1* y establece el voltaje de salida del servo de *eje1*. En este ejemplo, la salida es del 2% del valor de salida.

Lógica de escalera de relés



MDO
Motion Direct Drive On
Axis Axis1
Motion Control MDO_3
Drive Output 50
Drive Units Percent

Ejemplo de lógica de escalera de MDO

Texto estructurado

MDO(Axis0,MDO_1,4,percent);

Motion Direct Drive Off (MDF)

Use la instrucción MDF para desactivar el servovariador y para establecer el voltaje de salida del servo al voltaje de offset de salida. El voltaje de offset de salida es el voltaje de salida que genera movimiento nulo o mínimo del variador. Este valor se puede especificar durante la configuración del eje.

Operandos: *Lógica de escalera de relés*

MDF
Motion Direct Drive Off
Axis ?
Motion Control ?

MDF(Axis,MotionControl);

Operando	Tipo de datos	Descripción
AXIS	tag	Nombre del eje en el cual se realizará la operación.
Motion Control	Tag MOTION_ INSTRUCTION	Estructura usada para obtener acceso a los parámetros de estado de instrucción.

Texto estructurado

Los operandos son iguales a los de la instrucción MDF de lógica de escalera de relés.

EstructuraMOTION_INSTRUCTION

Mnemónico	Descripción
Bit .EN (Habilitar) 31	Se establece cuando el renglón hace una transición de falso a verdadero y permanece establecido hasta que la transacción de mensaje del servo se completa y el renglón pasa a falso.
Bit .DN (Listo) 29	Se establece cuando las señales del variador del eje se han inhabilitado correctamente y se borra el bit de estado de habilitación del variador.
Bit .ER (Error) 28	Se establece para indicar que la instrucción ha detectado un error, como si se hubiese especificado un eje no configurado.

Descripción: Para módulos de movimiento con una interface de servovariador externo, por ejemplo 1756-M02AE, la instrucción MDF inhabilita directamente la salida de habilitación del variador del módulo de movimiento del eje físico especificado y también “vuelve a cero” la salida del servo de los módulos al variador externo aplicando el valor de offset de salida configurado.

Se usa la instrucción MDF para detener el movimiento iniciado por una instrucción MDO previa y cambiar el eje del estado de control directo del variador a estado de eje listo.

Para ejecutar correctamente una instrucción MDF, el eje especificado debe estar configurado como un servoeje o eje de sólo retroalimentación. En caso contrario, la instrucción dará error.

IMPORTANTE La ejecución de la instrucción MDF puede necesitar múltiples escanes porque requiere la transmisión de un mensaje al módulo de movimiento. El bit Listo (.DN) no se establece si el mensaje no ha sido transmitido correctamente.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Vea Códigos de error (ERR) para las instrucciones de control de movimiento.

MDF cambia a bits de estado: *Bits de estado de eje*

Nombre del bit	Estado	Significado
DriveEnableStatus	FALSO	El eje está en estado de eje listo con la salida de habilitación del variador ahora activa.

Ejemplo: Cuando las condiciones de entrada son verdaderas, el controlador desactiva el servovariador para *eje1* y establece el voltaje de salida de servo de *eje_* al valor de offset de salida.

Lógica de escalera de relés

MDF
Motion Direct Drive Off
Axis Axis0
Motion Control MDF_1

Ejemplo de lógica de escalera de MDF

Texto estructurado

MDF(Axis0,MDF_1);

Motion Axis Fault Reset (MAFR)

Use la instrucción MAFR para borrar todos los fallos de movimiento para un eje. Éste es el único método para borrar fallos de movimiento de eje.

IMPORTANTE La instrucción MAFR elimina el estado de fallo, pero no realiza ninguna otra recuperación, tal como habilitar la acción del servo. Además, cuando el controlador elimina el estado de fallo, todavía puede existir la condición que generó el/los fallo/s. Si no se corrige la condición antes de usar la instrucción MAFR, el eje volverá a entrar en fallo inmediatamente.

Operandos: *Lógica de escalera de relés*

Operando	Tipo	Formato	Descripción
AXIS	AXIS_FEEDBACK AXIS_VIRTUAL AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje en el cual se realizará la operación.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura usada para obtener acceso a los parámetro de estado de instrucción.

Texto estructurado

MAFR(Axis,MotionControl);

Los operandos son iguales a los de la instrucción MAFR de lógica de escalera de relés.

Estructura MOTION_INSTRUCTION

Mnemónico	Descripción
Bit .EN (Habilitar) 31	Se establece cuando el renglón hace una transición de falso a verdadero y permanece establecido hasta que la transacción de mensaje del servo se completa y el renglón pasa a falso.
Bit .DN (Listo) 29	Se establece cuando se han borrado correctamente los fallos del eje.
Bit .ER (Error) 28	Se establece para indicar que la instrucción ha detectado un error, como si se hubiese especificado un eje no configurado.

Descripción: La instrucción MAFR borra directamente el estado de fallo especificado del eje seleccionado. No corrige la condición que causó el error. Si no se corrige la condición antes de ejecutar la instrucción MAFR, el eje puede volver a entrar en fallo inmediatamente dando la impresión de que el estado de fallo no se restableció.

Esta instrucción se usa generalmente como parte de un programa gestor de fallos, que proporciona una acción específica ante un fallo como respuesta a varios fallos potenciales de control de movimiento.

Una vez que se realiza la acción ante un fallo adecuada, la instrucción MAFR puede usarse para borrar todos los bits de estado de fallo activos.

Para ejecutar correctamente una instrucción MAFR, el eje especificado debe estar configurado como un servoeje o de sólo retroalimentación. En caso contrario, la instrucción dará error.

La instrucción MAFR es un comando de tipo procedimiento que se procesa desde el controlador Logix, a través del módulo SERCOS, y hacia los variadores asociados. Para aplicaciones que usan un controlador Logix con RSLogix 5000, versión 13 y anterior, el bit .DN de la instrucción se estableció cuando el módulo SERCOS confirmó el pedido de procedimiento. Los variadores confirmaron el comando y lo procesaron rápidamente. Para el usuario, la operación del bit .DN parece indicar la finalización correcta del borrado de fallos casi inmediatamente.

Para aplicaciones que usan un controlador Logix con RSLogix 5000, versión 15 y posterior, las versiones de firmware del variador actuales han incorporado una nueva parte del mecanismo de borrado de fallos que activa un proceso de recuperación de posición en el variador que recalcula la conmutación. Este recálculo puede hacer que el variador que no esté en un estado listo pase a servo activado durante 0.5 a 5 segundos después de realizado el restablecimiento. Para reflejar este cambio en la instrucción, la operación de bit .DN se cambió para esperar a que el variador complete el procedimiento de restablecimiento. El resultado final es que usted no verá una finalización correcta del borrado de fallos casi inmediatamente como quizás lo hacía en versiones anteriores.

IMPORTANTE La ejecución de la instrucción MAFR puede necesitar múltiples escanes porque requiere la transmisión de un mensaje al módulo de movimiento. El bit Listo (.DN) no se establece si este mensaje no ha sido transmitido correctamente. No se garantiza que está instrucción borrará todos los fallos, ya que uno o más fallos pueden ser provocados por una condición permanente.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Vea Códigos de error (ERR) para las instrucciones de control de movimiento.

MAFR cambia a bits de estado: ninguno

Ejemplo de MAFR: Cuando las condiciones de entrada son verdaderas, el controlador borra todos los fallos de movimiento para *eje1*.

Lógica de escalera de relés

Ejemplo de lógica de escalera de MAFR

Texto estructurado

```
MAFR(Axis0,MAFR_1);
```

Instrucciones que producen movimiento

(MAS, MAH, MAJ, MAM, MAG, MCD, MRP, MCCP, MAPC, MATC, MCSV)

Introducción

Utilice las instrucciones que producen movimiento para controlar la posición del eje.

Si desea	Use esta instrucción	Disponible en estos lenguajes
Detener algún proceso de movimiento en un eje	Motion Axis Stop (MAS)	lógica de escalera de relés texto estructurado
Volver el eje a la posición inicial	Motion Axis Home (MAH)	lógica de escalera de relés texto estructurado
Impulsar un eje	Motion Axis Jog (MAJ)	lógica de escalera de relés texto estructurado
Mover un eje a una posición específica	Motion Axis Move (MAM)	lógica de escalera de relés texto estructurado
Iniciar un engranaje electrónico entre 2 ejes	Motion Axis Gear (MAG)	lógica de escalera de relés texto estructurado
Cambiar la velocidad, aceleración o desaceleración de un movimiento o impulso en progreso	Motion Change Dynamics (MCD)	lógica de escalera de relés texto estructurado
Cambiar el comando o la posición real de un eje	Motion Redefine Position (MRP)	lógica de escalera de relés texto estructurado
Calcular un perfil de levas basado en una matriz de puntos de levas	Motion Calculate Cam Profile (MCCP)	lógica de escalera de relés texto estructurado
Iniciar una operación electrónica de levas entre 2 ejes	Motion Axis Position Cam (MAPC)	lógica de escalera de relés texto estructurado
Iniciar una operación electrónica de levas en función al tiempo	Motion Axis Time Cam (MATC)	lógica de escalera de relés texto estructurado
Calcular el valor del esclavo, la pendiente y la derivada de la pendiente para un perfil de levas y un valor de maestro	Motion Calculate Slave Values (MCSV)	lógica de escalera de relés texto estructurado

Motion Axis Stop (MAS)

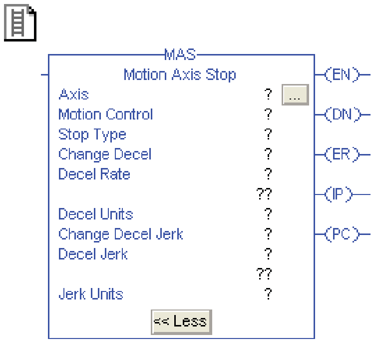
Utilice la instrucción MAS para detener un proceso de movimiento específico o para detener el eje completamente.

ATENCIÓN

⚠

Utilice un tag de control de movimiento sólo una vez. No vuelva a utilizarlo en otra instrucción. De lo contrario, podría ocasionar un movimiento inesperado del equipo y provocar lesiones a personas.

Operandos: *Lógica de escalera de relés*



Operando	Tipo	Formato	Descripción
AXIS	AXIS_VIRTUAL AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje
Motion Control	MOTION_INSTRUCTION	tag	Tag de control para la instrucción

Operando	Tipo	Formato	Descripción	
Stop Type	DINT	Inmediato	**Para detener**	**Seleccione este Stop Type**
			Todos los movimientos en proceso de este eje	All (0) – Con esta opción, la instrucción detiene todos los movimientos en un eje. El paro tiene en cuenta un movimiento coordinado en el eje cuando calcula la velocidad de desaceleración y detiene ese componente del movimiento coordinado. Los otros componentes de ejes del movimiento coordinado permanecen inalterados y continúan.
			Sólo un determinado tipo de movimiento pero dejando en marcha otros procesos de movimiento	Seleccione el tipo de movimiento que desea detener: • Impulso (1) • Movimiento (2) • Engranaje (3) • Posición inicial (4) • Ajuste (5) • Prueba (6) • Leva de posición (7) • Leva de tiempo (8) • Movimiento de offset maestro (9) El eje debe estar todavía en movimiento cuando la instrucción MAS finalice.
Change Decel	DINT	Inmediato	**Si desea**	**Seleccione**
			Utilizar la velocidad máxima de desaceleración del eje	No (0)
			Especificar la velocidad de desaceleración	Yes (1)
Decel Rate	REAL	Inmediato o tag	**Importante:** El eje podría sobreimpulsar su posición específica si reduce la desaceleración mientras un movimiento está en proceso. Velocidad de desaceleración del eje. La instrucción utiliza este valor sólo si Change Decel está en Yes.	
Decel Units	DINT	Inmediato	¿Qué unidades desea utilizar para Decel Rate? · Unidades por seg^2 (0) · % del Máximo (1)	

Operando	**Tipo**	**Formato**	**Descripción**	
Change Decel Jerk	DINT	Inmediato	**Si desea**	**Seleccione**
			Utilizar la velocidad máxima de jaloneo de desaceleración del eje	No (0)
			Programar la velocidad de jaloneo de desaceleración	Yes (1)
Decel Jerk	REAL	Inmediato o tag	**Importante:** El eje podría sobreimpulsar su posición específica si reduce el jaloneo de desaceleración mientras un movimiento está en proceso.	
Jerk Units	DINT	Inmediato	Siempre **debe** introducir un valor para el operando Decel Jerk. Esta instrucción sólo utiliza el valor si el perfil está configurado como curva en S. · Decel Jerk es la velocidad de jaloneo de desaceleración del eje. Utilice estos valores para comenzar. Decel Jerk = 100 (% de Tiempo) 0 = Unidades por seg^3 1 = % del Máximo 2 = % de Tiempo (utilice este valor para comenzar)	

MAS(Axis,MotionControl, StopType,ChangeDecel, DecelRate,DecelUnits, ChangeDecelJerk, DecelJerk,JerkUnits);

Texto estructurado

Los operandos del texto estructurado son iguales a los operandos de lógica de escalera de relés.

Este operando	**Tiene estas opciones, las cuales usted puede**	
	introducir como texto	**o introducir como**
Stop Type	all	0
	jog	1
	move	2
	gear	3
	home	4
	tune	5
	test	6
	timecam	7
	positioncam	8
	masteroffsetmove	9
Change Decel	no	0
	sí	1
Decel Units	Unidades por seg^2	0
	% del máximo	1

Este operando	**Tiene estas opciones, las cuales usted puede**	
	introducir como texto	**o introducir como**
Change Decel Jerk	no sí	0 1
Decel Jerk	sin enumeración	Siempre **debe** introducir un valor para el operando Decel Jerk. Esta instrucción sólo utiliza el valor si el perfil está configurado como curva en S. · Decel Jerk es la velocidad de jaloneo de desaceleración del eje. Utilice este valor para comenzar. Decel Jerk = 100% de Tiempo (2)
Jerk Units	unitspersec3 % del Máximo % de Tiempo	0 1 2

Tipo de datos MOTION_INSTRUCTION:

Para ver si	**Verifique si este bit está establecido**	**Tipo de datos**	**Notas**
Una transición de falso a verdadero ha provocado la ejecución de la instrucción	EN	BOOL	El bit EN queda establecido hasta que el proceso finaliza y el renglón pasa a falso.
El paro se ha iniciado correctamente	DN	BOOL	
Ha ocurrido un error	ER	BOOL	
El eje se detiene	IP	BOOL	Cualquiera de estas acciones finalizan la instrucción MAS y borran el bit IP: · El eje está detenido · Otra instrucción MAS reemplaza esta instrucción MAS · Comando cierre eléctrico · Acción ante un fallo
El eje se detuvo	PC	BOOL	El bit PC queda establecido hasta que el renglón hace una transición de falso a verdadero.

Descripción:

Utilice la instrucción MAS si desea un paro desacelerado para algún movimiento controlado en proceso en el eje. La instrucción detiene el movimiento sin inhabilitar el lazo del servo. Para la desaceleración, siempre se utiliza un perfil trapezoidal para MAS con Stop Type = ALL independientemente del tipo de perfil programado. Utilice la instrucción para:

- detener un proceso de movimiento específico como el funcionamiento por impulsos, el movimiento o el engranaje

- detener completamente el eje
- cancelar una prueba o ajustar un proceso iniciado por una instrucción MRHD o MRAT

¿Qué tipo de perfil utiliza la instrucción MAS?

Si Stop Type es	Entonces, la instrucción MAS utiliza este perfil
Jog	El mismo tipo de perfil que la instrucción MAJ que inició el impulso
Mover	El mismo tipo de perfil que la instrucción MAM que inició el movimiento
ninguno de los anteriores	Trapezoidal

Ejemplo

Supongamos que usted utiliza una instrucción MAJ con un perfil de curva en S para iniciar un impulso. Entonces, utiliza una instrucción MAS con un tipo de paro de Impulso para detener el impulso. En ese caso, la instrucción MAS utiliza un perfil de curva en S para detener el impulso.

Pautas de programación:

ATENCIÓN

Riesgo de sobreimpulso de velocidad y/o de posición final

Si cambia los parámetros de movimiento dinámicamente mediante algún método, es decir, cambiando la dinámica de movimiento (MCD o MCCD) o comenzando una nueva instrucción antes de que haya finalizado la última, tenga en cuenta el riesgo de sobreimpulso de velocidad y/o de posición final.

Un perfil de velocidad trapezoidal puede sobreimpulsar si disminuye la desaceleración máxima mientras el movimiento está desacelerando o está próximo al punto de desaceleración.

Un perfil de velocidad de curva en S puede sobreimpulsar si:

- la desaceleración máxima disminuye mientras el movimiento está desacelerando o está próximo al punto de desaceleración; o

- el jaloneo máximo de aceleración disminuye y el eje se acelera. Sin embargo, recuerde que el jaloneo se puede cambiar indirectamente si está especificado en % de tiempo.

Para obtener más información, consulte Troubleshoot Axis Motion en page 9.

Pauta	Detalles
· **En la lógica de escalera de relés, alterne la condición de renglón cada vez que desee ejecutar la instrucción.**	Ésta es una instrucción transicional: · En la lógica de escalera de relés, alterne la condición de entrada del renglón de borrado a establecido cada vez que desee ejecutar la instrucción.

<table>
<tr><th>Pauta</th><th colspan="2">Detalles</th></tr>
<tr><td>· En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición.</td><td colspan="2">En texto estructurado, las instrucciones se ejecutan cada vez que son escaneadas.
· Acondicione la instrucción de manera que sólo se ejecute en una transición. Utilice alguno de estos dos métodos:
  · calificador de una acción SFC
  · construcción de texto estructurado
Para obtener más información, vea el Apéndice C.</td></tr>
<tr><td rowspan="4">· Seleccione detener todos los movimientos o sólo un tipo específico de movimiento.</td><td>Si desea detener</td><td>Seleccione este Stop Type</td></tr>
<tr><td>Todos los movimientos en proceso para este eje</td><td>All
La instrucción utiliza un perfil trapezoidal y detiene el eje.</td></tr>
<tr><td>Detener sólo un determinado tipo de movimiento pero dejando en marcha otros procesos de movimiento</td><td>El tipo de movimiento que desea detener
El eje debe estar todavía en movimiento cuando la instrucción MAS finalice.
La instrucción utiliza un perfil de curva en S para detener el eje sólo si:
· El tipo de paro es Jog o Move, y
· El impulso o movimiento utilizó un perfil de curva en S.</td></tr>
<tr><td colspan="2">Ejemplo: Supongamos que su eje ejecuta un impulso y un movimiento al mismo tiempo. Supongamos también que desea detener sólo el impulso pero dejar el movimiento en marcha. En ese caso, seleccione Jog como tipo de paro.</td></tr>
<tr><td>· Para detener un engranaje u operación de levas, seleccione el eje esclavo.</td><td colspan="2">Para detener un proceso de engranaje u operación de levas de posición, introduzca el eje esclavo para desactivar el proceso específico y detener el eje. Si el eje maestro es un servoeje, puede detener el eje maestro el cual, a la vez, detiene el esclavo sin inhabilitar el engranaje u operación de levas de posición.</td></tr>
<tr><td>· Para detener un movimiento offset de maestro, introduzca el eje esclavo pero utilice unidades maestras.</td><td colspan="2">Para detener un movimiento offset de maestro absoluto o incremental:
· Para ejes, introduzca el eje esclavo.
· Para desaceleración y jaloneo, introduzca los valores y unidades para el eje maestro.</td></tr>
<tr><td>· Tenga cuidado si la instrucción cambia los parámetros de movimiento.</td><td colspan="2">Cuando usted ejecuta una instrucción MAS, el eje utiliza las nuevas velocidades de desaceleración y jaloneo para el movimiento que está ya en proceso. Esto puede hacer que el eje sobreimpulse su velocidad, sobreimpulse su posición final o invierta la dirección. Los perfiles de curva en S son más sensibles a los cambios de parámetros.
Para obtener más información, consulte Troubleshoot Axis Motion en page 9.</td></tr>
<tr><td>· Utilice los operandos de jaloneo para los perfiles de curva en S.</td><td colspan="2">Utilice los operandos de jaloneo cuando
· El tipo de paro es Jog o Move.
· El impulso o movimiento utiliza un perfil de curva en S.
En esas condiciones, la instrucción utiliza un perfil de curva en S para detener el eje. La instrucción utiliza una velocidad constante de desaceleración para todos los tipos de paro. Debe completar los operandos de jaloneo independientemente del tipo de paro.</td></tr>
<tr><td>· Utilice el % de tiempo para facilitar la programación y ajuste del jaloneo.</td><td colspan="2">Para facilitar la programación y ajuste del jaloneo, introdúzcalo como % de tiempo de aceleración o desaceleración.
Para obtener más información, consulte:
· Programar un perfil de velocidad en página 27
· Ajustar un perfil de curva en S en página 391.</td></tr>
</table>

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Consulte Códigos de error (ERR) para las instrucciones de control de movimiento en página 395.

Códigos de error extendidos: Utilice los códigos de error extendidos (EXERR) para obtener más información acerca de un error.

<table>
<tr><th rowspan="2">Si ERR es</th><th rowspan="2">y EXERR es</th><th colspan="2">Entonces</th></tr>
<tr><th>Causa</th><th>Acción correctiva</th></tr>
<tr><td>13</td><td>Varía</td><td>Un operando está fuera de su rango.</td><td>El EXERR es el número del operando que está fuera de rango.
El primer operando es 0.
Por ejemplo, si EXERR = 4, verifique la velocidad de desaceleración.
<table>
<tr><th>EXERR</th><th>Operando MAS</th></tr>
<tr><td>0</td><td>AXIS</td></tr>
<tr><td>1</td><td>Motion Control</td></tr>
<tr><td>2</td><td>Stop Type</td></tr>
<tr><td>3</td><td>Change Decel</td></tr>
<tr><td>4</td><td>Decel Rate</td></tr>
</table></td></tr>
</table>

Cambia a bits de estado: *Bits de estado de movimiento*

Si Stop Type es	Entonces		
NOT All	La instrucción borra el bit Motion Status para el proceso de movimiento que detuvo.		
All	La instrucción borra todos los bits Motion Status.		
	Bit	**Estado**	**Significado**
	MoveStatus	FALSO	El eje no se mueve.
	JogStatus	FALSO	El eje no se impulsa.
	GearingStatus	FALSO	El eje no se acopla.
	HomingStatus	FALSO	El eje no vuelve a la posición inicial.
	StoppingStatus	VERDADERO	El eje se detiene.
	PositionCamStatus	FALSO	El eje no realiza la operación de levas de posición.
	TimeCamStatus	FALSO	El eje no realiza la operación de levas de tiempo.
	PositionCamPendingStatus	FALSO	El eje no tiene pendiente una operación de levas de posición.
	TimeCamPendingStatus	FALSO	El eje no tiene pendiente una operación de levas de tiempo.
	GearingLockStatus	FALSO	El eje no está en una condición de engranaje bloqueado.
	PositionCamLockStatus	FALSO	El eje no está en una condición de operación de levas bloqueada.

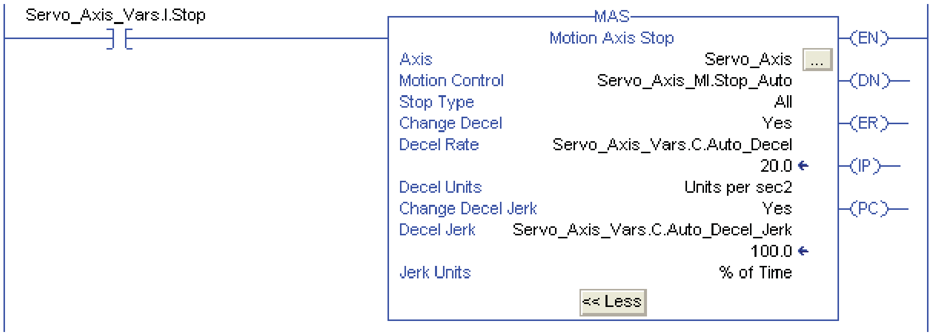
Ejemplo 1 Cuando se enciende *Servo_Axis_Vars.I.Stop*

Se detienen todos los movimientos en *Servo_Axis*.

Desacelera a 20.0 unidades por seg2.

La instrucción no utiliza el valor de Decel Jerk. Puesto que el tipo de paro es All, la instrucción utiliza un perfil trapezoidal para detener el eje.

Lógica de escalera de relés

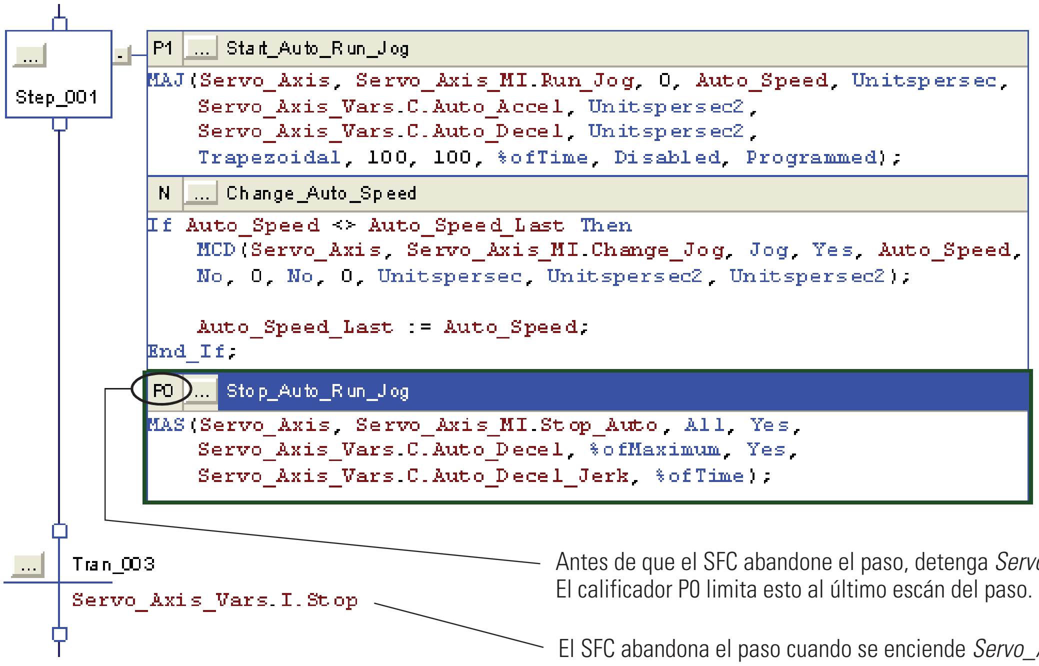


Servo_Axis_Vars.I.Stop

MAS
Motion Axis Stop

Axis	Servo_Axis
Motion Control	Servo_Axis_MI.Stop_Auto
Stop Type	All
Change Decel	Yes
Decel Rate	Servo_Axis_Vars.C.Auto_Decel 20.0 ←
Decel Units	Units per sec2
Change Decel Jerk	Yes
Decel Jerk	Servo_Axis_Vars.C.Auto_Decel_Jerk 100.0 ←
Jerk Units	% of Time

Texto estructurado



```
MAJ(Servo_Axis, Servo_Axis_MI.Run_Jog, 0, Auto_Speed, Unitspersec,
    Servo_Axis_Vars.C.Auto_Accel, Unitspersec2,
    Servo_Axis_Vars.C.Auto_Decel, Unitspersec2,
    Trapezoidal, 100, 100, %ofTime, Disabled, Programmed);
```

```
If Auto_Speed <> Auto_Speed_Last Then
    MCD(Servo_Axis, Servo_Axis_MI.Change_Jog, Jog, Yes, Auto_Speed,
    No, 0, No, 0, Unitspersec, Unitspersec2, Unitspersec2);

    Auto_Speed_Last := Auto_Speed;
End_If;
```

```
MAS(Servo_Axis, Servo_Axis_MI.Stop_Auto, All, Yes,
    Servo_Axis_Vars.C.Auto_Decel, %ofMaximum, Yes,
    Servo_Axis_Vars.C.Auto_Decel_Jerk, %ofTime);
```

```
Servo_Axis_Vars.I.Stop
```

Paro

Antes de que el SFC abandone el paso, detenga *Servo_Axis*.
El calificador P0 limita esto al último escán del paso.

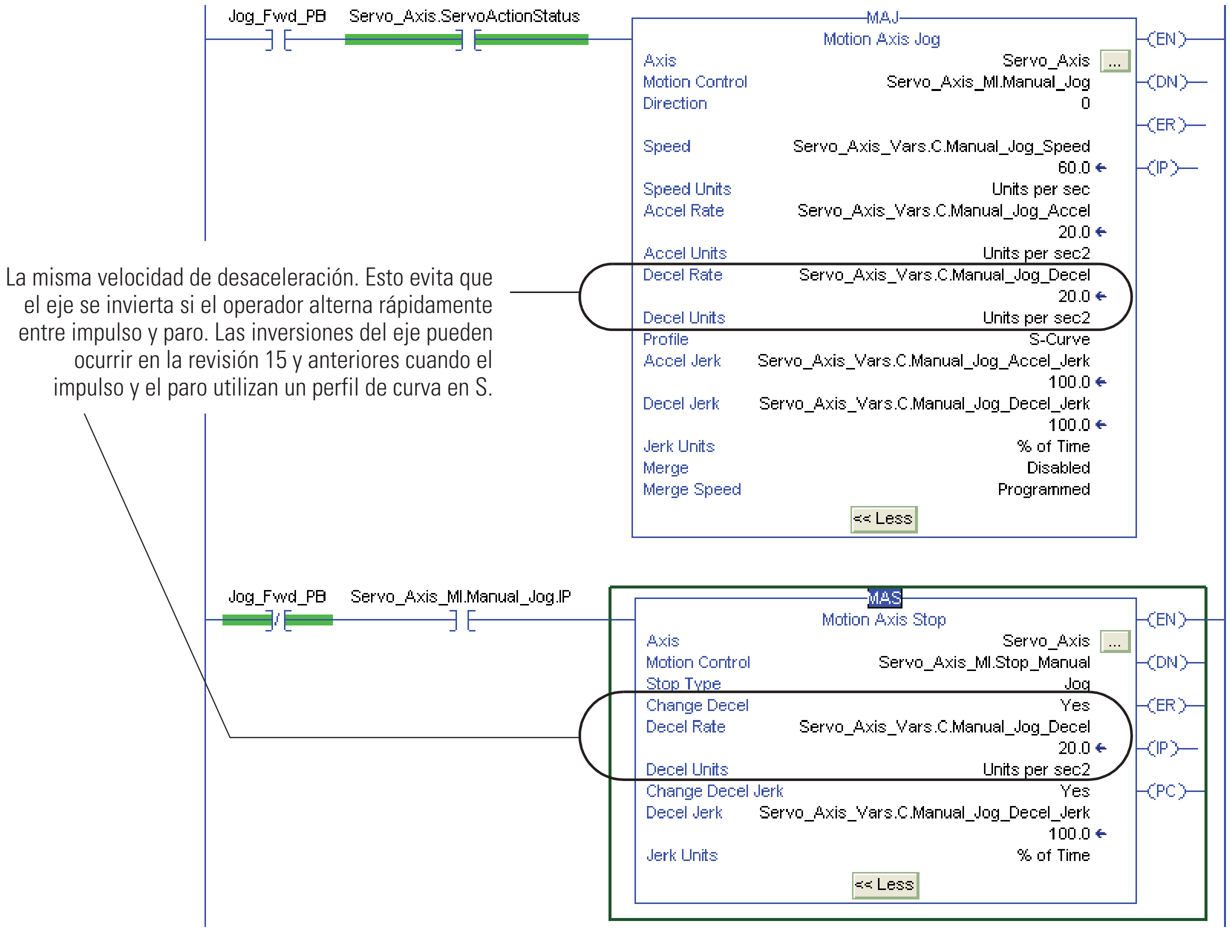
El SFC abandona el paso cuando se enciende *Servo_Axis_Vars.I.Stop*.

Ejemplo 2 El operador utiliza un botón pulsador para impulsar un eje. El botón pulsador enciende y apaga el tag *Jog_Fwd_PB*. Cuando el operador suelta el botón, la instrucción MAS detiene el eje.

La instrucción MAS utiliza un perfil de curva en S para detener el eje porque:

- La instrucción MAJ utiliza un perfil de curva en S.
- El tipo de paro es Jog para la instrucción MAS.

Lógica de escalera de relés



Jog_Fwd_PB Servo_Axis.ServoActionStatus

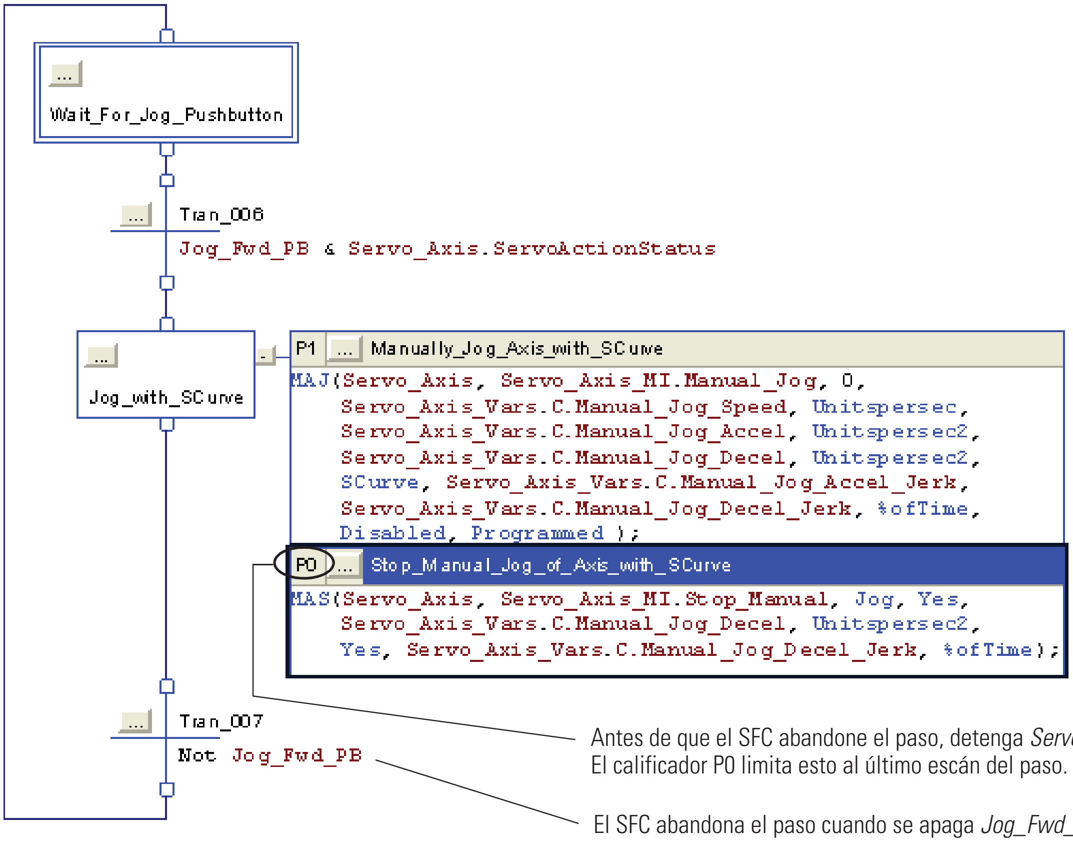
MAJ
Motion Axis Jog
Axis Servo_Axis
Motion Control Servo_Axis_MI.Manual_Jog
Direction 0
Speed Servo_Axis_Vars.C.Manual_Jog_Speed 60.0
Speed Units Units per sec
Accel Rate Servo_Axis_Vars.C.Manual_Jog_Accel 20.0
Accel Units Units per sec2
Decel Rate Servo_Axis_Vars.C.Manual_Jog_Decel 20.0
Decel Units Units per sec2
Profile S-Curve
Accel Jerk Servo_Axis_Vars.C.Manual_Jog_Accel_Jerk 100.0
Decel Jerk Servo_Axis_Vars.C.Manual_Jog_Decel_Jerk 100.0
Jerk Units % of Time
Merge Disabled
Merge Speed Programmed

Jog_Fwd_PB Servo_Axis_MI.Manual_Jog.IP

MAS
Motion Axis Stop
Axis Servo_Axis
Motion Control Servo_Axis_MI.Stop_Manual
Stop Type Jog
Change Decel Yes
Decel Rate Servo_Axis_Vars.C.Manual_Jog_Decel 20.0
Decel Units Units per sec2
Change Decel Jerk Yes
Decel Jerk Servo_Axis_Vars.C.Manual_Jog_Decel_Jerk 100.0
Jerk Units % of Time

La misma velocidad de desaceleración. Esto evita que el eje se invierta si el operador alterna rápidamente entre impulso y paro. Las inversiones del eje pueden ocurrir en la revisión 15 y anteriores cuando el impulso y el paro utilizan un perfil de curva en S.

Texto estructurado



```
Jog_Fwd_PB & Servo_Axis.ServoActionStatus
```

```
MAJ(Servo_Axis, Servo_Axis_MI.Manual_Jog, 0,
    Servo_Axis_Vars.C.Manual_Jog_Speed, Unitspersec,
    Servo_Axis_Vars.C.Manual_Jog_Accel, Unitspersec2,
    Servo_Axis_Vars.C.Manual_Jog_Decel, Unitspersec2,
    SCurve, Servo_Axis_Vars.C.Manual_Jog_Accel_Jerk,
    Servo_Axis_Vars.C.Manual_Jog_Decel_Jerk, %ofTime,
    Disabled, Programmed );
```

```
MAS(Servo_Axis, Servo_Axis_MI.Stop_Manual, Jog, Yes,
    Servo_Axis_Vars.C.Manual_Jog_Decel, Unitspersec2,
    Yes, Servo_Axis_Vars.C.Manual_Jog_Decel_Jerk, %ofTime);
```

```
Not Jog_Fwd_PB
```

Paro

Antes de que el SFC abandone el paso, detenga *Servo_Axis*. El calificador P0 limita esto al último escán del paso.

El SFC abandona el paso cuando se apaga *Jog_Fwd_PB*.

Operando Profile

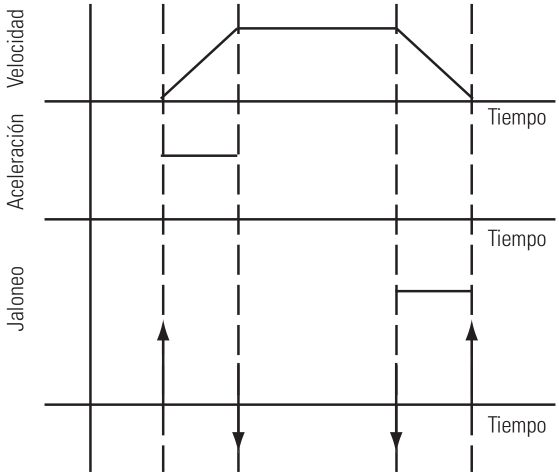
Este operando tiene dos tipos de perfil:

- Perfil de velocidad trapezoidal
- Perfil de velocidad curvan en S

Perfil de velocidad trapezoidal

El perfil de velocidad trapezoidal es el perfil utilizado más frecuentemente porque proporciona mayor flexibilidad a la hora de programar movimientos subsiguientes y tiempos de aceleración y desaceleración más rápidos. El cambio en velocidad por tiempo de unidad se especifica por la aceleración y desaceleración. El jaloneo no es un factor para los perfiles trapezoidales. Por lo tanto, se considera infinito y se muestra como una línea vertical en el siguiente diagrama.

Tiempo Aceleración/Desaceleración trapezoidal



Perfil de velocidad curvan en S

Los perfiles de velocidad de curva en S son los utilizados más frecuentemente cuando es necesario minimizar la carga y el esfuerzo en el sistema mecánico. El tiempo de aceleración y desaceleración se equilibra en comparación con el esfuerzo de la máquina utilizando dos parámetros adicionales, jaloneo de aceleración y jaloneo de desaceleración.

Según los ajustes del jaloneo, el perfil de aceleración se puede establecer como casi totalmente rectangular, consulte Tiempo Aceleración/Desaceleración trapezoidal en página 71 (el esfuerzo más rápido y alto) o como triangular, consulte Tiempo de aceleración/desaceleración de curva en S programable, Jaloneo de aceleración = 60% de tiempo en página 74, (el esfuerzo más lento y bajo).

El típico perfil de aceleración es un equilibrio entre esfuerzo y velocidad, como se muestra en Tiempo de aceleración/desaceleración de curva en S, Configuración de retrocompatibilidad: Jaloneo de aceleración = 100% de tiempo en página 74.

El Jaloneo está especificado por el usuario (en Unidades/seg^3 o como un porcentaje del máximo) o se calcula a partir del porcentaje de tiempo. (El porcentaje de tiempo es igual al porcentaje del tiempo de rampa en el perfil de aceleración/desaceleración):

$$j_a\ [EU/s^3] = \frac{a_{\text{máx}}^{2}\ [EU/s^2]}{v_{\text{máx}}\ [EU/s]}\left(\frac{200}{j_a\ [\%\ \text{de tiempo}]} - 1\right)$$

$$j_a\ [EU/s^3] = \frac{d_{\text{max}}^{2}\ [EU/s^2]}{v_{\text{max}}\ [EU/s]}\left(\frac{200}{j_a\ [\%\ \text{de tiempo}]} - 1\right)$$

Deberá tener en cuenta este dato con respecto a la retrocompatibilidad. El jaloneo del 100% de tiempo produce perfiles de aceleración y desaceleración triangulares. Estos perfiles son los producidos anteriormente, como se muestra en Tiempo de aceleración/desaceleración de curva en S, Configuración de retrocompatibilidad: Jaloneo de aceleración = 100% de tiempo en página 74.

Los regímenes de jaloneo muy pequeños, es decir, inferiores al 5% del tiempo, producen perfiles de aceleración y desaceleración cercanos a los rectangulares, como el que se muestra en Tiempo Aceleración/Desaceleración trapezoidal en página 71.

IMPORTANTE Los valores más altos del % de Tiempo dan como resultado valores más bajos en los límites de régimen de jaloneo y, por lo tanto, perfiles más bajos. Consulte la siguiente tabla como referencia.

	Perfil de velocidad trapezoidal[1]	**Perfil de velocidad en forma de S con 1<= Jaloneo <100% del tiempo**[2]	**Perfil de velocidad en forma de S con Jaloneo = 100% del tiempo**[3]
Jaloneo de aceleración/desaceleración en Unidades/seg^3	∞	$\frac{\text{Máx Acel}^2}{\text{Máx Velocidad}}$ a ∞	$\frac{\text{Máx Acel}^2}{\text{Máx Velocidad}}$
Jaloneo de aceleración/desaceleración en % del máximo	NA	0 – 100%	NA
Jaloneo de aceleración/desaceleración en % del tiempo	0%	1 – 100%	100%

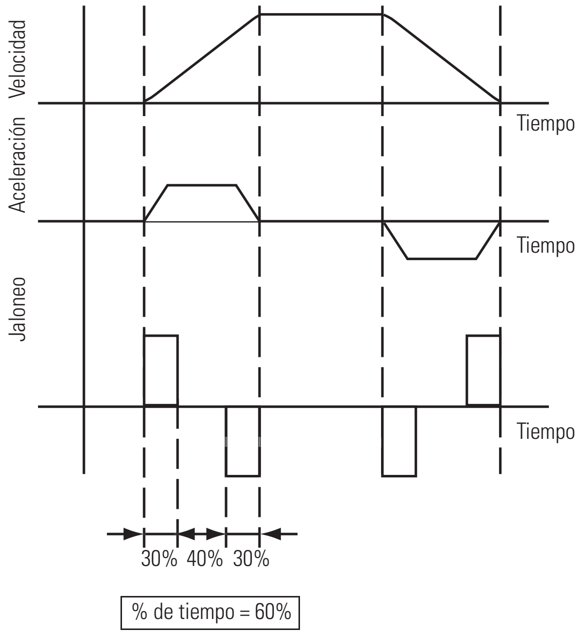
[1] El ejemplo en página 71 (Tiempo de aceleración/desaceleración trapezoidal etiquetado) utiliza un perfil de aceleración rectangular.

[2] El ejemplo de página 74 (Tiempo de aceleración/desaceleración programable etiquetado, Jaloneo de aceleración = 60% de tiempo) utiliza un perfil de aceleración trapezoidal.

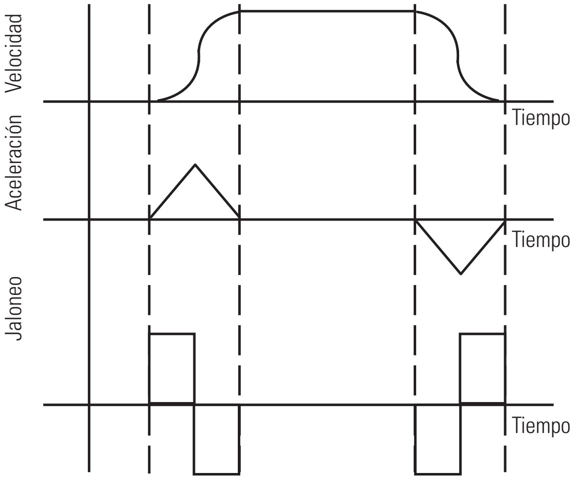
[3] El ejemplo de página 74 (Tiempo de aceleración/desaceleración de curva en S etiquetado, Configuración de retrocompatibilidad: Jaloneo de aceleración = 100% de tiempo) utiliza un perfil de aceleración triangular.

Los cálculos se realizan cuando se inicia un movimiento del eje, una dinámica de cambio o un Paro MCS o Stop Type = Move o Jog.

Tiempo de aceleración/desaceleración de curva en S programable, Jaloneo de aceleración = 60% de tiempo



Tiempo de aceleración/desaceleración de curva en S, Configuración de retrocompatibilidad: Jaloneo de aceleración = 100% de tiempo



Motion Axis Home (MAH)

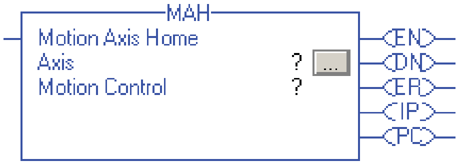
Utilice la instrucción MAH para volver el eje a su posición inicial. Durante la configuración del eje, se pueden seleccionar dos modos diferentes de vuelta a la posición inicial: Activo o pasivo. Si se selecciona una secuencia de vuelta a la posición inicial activa, el eje ejecuta el tipo configurado de secuencia de posición inicial y establece una posición absoluta del eje. No obstante, si se selecciona una vuelta pasiva a la posición inicial, no se ejecuta ninguna secuencia específica de vuelta a la posición inicial y el eje queda esperando el impulso del siguiente marcador para establecer la posición inicial.

ATENCIÓN

Utilice un tag de control de movimiento sólo una vez. No vuelva a utilizarlo en otra instrucción. De lo contrario, podría ocasionar un movimiento inesperado del equipo y provocar lesiones a personas.

Operandos:

Lógica de escalera de relés



Operando	Tipo	Formato	Descripción
AXIS	AXIS_FEEDBACK AXIS_VIRTUAL AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje en el cual se realizará la operación.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura que se utiliza para acceder a los parámetros de estado de la instrucción.

Texto estructurado

```
MAH(Axis,MotionControl);
```

Los operandos son iguales a los de la instrucción MAH de la lógica de escalera de relés.

Tipo de datos MOTION_INSTRUCTION:

Mnemónico	Descripción
Bit .EN (Habilitado) 31	Se establece cuando el renglón realiza una transición de falso a verdadero y permanece habilitado hasta que la transacción de mensaje servo se completa y el renglón cambia a falso.
Bit .DN (Listo) 29	Se establece cuando la vuelta del eje a la posición inicial se ha completado correctamente o ha sido cancelada.
Bit .ER (Error) 28	Se establece para indicar que la instrucción ha detectado un error, tal como especificar un eje no configurado.
.IP (En Proceso) Bit 27	Se establece en transición de renglón positiva y desaparece una vez que está completa la instrucción Motion Home Axis o es interrumpido por un comando de paro, un cierre eléctrico o un fallo del servo
.PC (Proceso Completo) Bit 26	Se establece cuando la vuelta del eje a la posición inicial se ha completado correctamente.

Descripción: La instrucción MAH se utiliza para calibrar la posición absoluta del eje especificado. En el caso de los ejes configurados como tipo Servo, el eje puede volver a la posición inicial utilizando la configuración del modo de vuelta a la posición inicial activa, pasiva, o absoluta. Para ejes de sólo retroalimentación, sólo están disponibles los modos de vuelta a la posición inicial pasiva y absoluta. El modo de vuelta a la posición inicial absoluta requiere que el eje esté equipado con dispositivo de retroalimentación absoluta.

Vuelta activa a la posición inicial

Cuando el modo de vuelta a la posición inicial del eje está configurado como activo, el eje físico primero se activa para funcionamiento servo. Como parte de este proceso, todos los otros movimientos en proceso se cancelan y se borran los bits de estado correspondientes. El eje vuelve a la posición inicial utilizando la secuencia de inicio configurada, la cual puede ser Inmediata, Interruptor, Marcador o Interruptor-Marcador. Las últimas tres secuencias de inicio hacen que el eje se desplace por impulsos en la dirección configurada de inicio y, una vez redefinida la posición según el evento de posición inicial detectado, el eje se mueve automáticamente a la posición inicial configurada.

IMPORTANTE Cuando la vuelta a la posición inicial activa unidireccional se realiza en un eje giratorio y el valor de offset de inicio es menor que la distancia de desaceleración cuando se detecta el evento de inicio, el control mueve el eje a la posición de desbobinado a cero. Esto garantiza que el movimiento resultante hacia la posición inicial sea unidireccional.

Vuelta pasiva a la posición inicial

Cuando el modo de vuelta a la posición inicial del eje está configurado en pasivo, la instrucción MAH redefine la posición real de un eje físico en la siguiente aparición del marcador de encoder. La vuelta pasiva a la posición inicial se utiliza normalmente para calibrar ejes de sólo retroalimentación en sus marcadores, pero también se puede utilizar en servoejes. La vuelta pasiva a la posición inicial es idéntica a la vuelta activa a la posición inicial en un marcador de encoder, excepto que el controlador de movimiento no ordena ningún movimiento del eje.

Después de iniciar la vuelta pasiva a la posición inicial, el eje se debe mover hasta pasar el marcador de encoder para que la secuencia de vuelta a la posición inicial se complete de manera apropiada. En el caso de Servoejes de lazo cerrado, esto se puede obtener con una instrucción MAM o MAJ. En el caso de los ejes físicos de sólo retroalimentación, el controlador de movimiento no puede ordenar el movimiento directamente y se debe obtener mediante otros medios.

Vuelta absoluta a la posición inicial

Si el hardware de movimiento del eje es compatible con un dispositivo de retroalimentación absoluta, se puede utilizar el modo absoluto de vuelta a la posición inicial. La única secuencia válida de inicio para el modo absoluto de vuelta a la posición inicial es "inmediato". En este caso, el proceso de vuelta la posición inicial absoluto establece la posición absoluta verdadera del eje aplicando la posición inicial configurada en la posición indicada del dispositivo de retroalimentación absoluta. Antes de la ejecución del proceso absoluto de vuelta a la posición inicial mediante la instrucción MAH, el eje debe estar en estado preparado con el lazo servo inhabilitado.

Para ejecutar correctamente una instrucción MAH en un eje configurado para el modo activo de vuelta a la posición inicial, el eje específico debe estar configurado como un tipo de servoeje. Para ejecutar correctamente una instrucción MAH, el eje específico debe estar configurado como servoeje o de sólo retroalimentación. Si no se cumple alguna de estas condiciones, la instrucción falla.

IMPORTANTE Cuando la instrucción MAH se ejecuta inicialmente, se establece el bit .IP En Proceso y se borra el bit Proceso Completo (.PC). La ejecución de la instrucción MAH puede necesitar la ejecución de múltiples escanes porque requiere la transmisión de múltiples mensajes al módulo de movimiento. Por lo tanto, el bit Listo (.DN) no se establecerá si estos mensajes no han sido correctamente transmitidos. Se borra el bit En Proceso (.IP) y se establece el bit Proceso Completo (.PC) al mismo tiempo que se establece el bit Listo (.DN).

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Consulte Códigos de error (ERR) para las instrucciones de control de movimiento en página 395.

Códigos de error extendidos: Los códigos de error extendidos proporcionan información adicional específica de instrucciones para los códigos de error que son genéricos a muchas instrucciones. Los siguientes códigos de error extendidos ayudan a localizar el problema cuando la instrucción MAH recibe un mensaje de error Fallo de mensajes al servomódulo (12) o Configuración de vuelta a la posición inicial no válida (41).

Código de error asociado (decimal)	Código de error extendido (decimal)	Significado
SERVO_MESSAGE_FAILURE (12)	Proceso finalizado a pedido (1)	La ejecución de la vuelta a la posición inicial seguida por una instrucción para interrumpir/inhabilitar el variador, o una instrucción de paro de movimiento o un cambio de procesador solicita la cancelación del inicio.
SERVO_MESSAGE_FAILURE (12)	Sin recursos (2)	No hay suficientes recursos de memoria para completar la solicitud. (SERCOS)
SERVO_MESSAGE_FAILURE (12)	Conflicto Modo de objeto (12)	El eje está en cierre eléctrico.
SERVO_MESSAGE_FAILURE (12)	Permiso denegado (15)	Error al habilitar el interruptor de entrada. (SERCOS)
SERVO_MESSAGE_FAILURE (12)	Dispositivo en estado incorrecto (16)	La Redefinición de la posición, inicio y registro 2 son excluyentes entre sí (SERCOS), el estado del dispositivo no es correcto para la acción. (SERCOS)
ILLEGAL_HOMING_CONFIG (41)	Secuencia de inicio (4)	La secuencia de inicio es incompatible con el modo de inicio.
ILLEGAL_HOMING_CONFIG (41)	Velocidad de inicio de cero (6)	La velocidad de inicio no puede ser cero.
ILLEGAL_HOMING_CONFIG (41)	Velocidad de retorno del inicio de cero (7)	La velocidad de retorno del inicio no puede ser cero.

Para el código de error 54 – El valor máximo de desaceleración es cero, si el error extendido devuelve un número positivo (0-*n*), está haciendo referencia al eje en error en el sistema de coordenadas. Diríjase a la ficha general Coordinate System Properties y mire debajo de la columna de corchetes ([ ]) de la cuadrícula del eje para determinar qué eje tiene un valor máximo de desaceleración de cero. Haga clic en el botón de elipsis al lado del eje en error para acceder a la pantalla Axis Properties. Diríjase a la ficha Dynamics y haga el cambio apropiado al valor máximo de desaceleración. Si el número del error extendido es –1, esto significa que el sistema de coordenadas tiene un valor máximo de desaceleración de 0. Diríjase a la ficha Coordinate System Properties Dynamics para corregir el valor máximo de desaceleración.

Bits de estado: *MAH Cambia a bits de estado*

Nombre del bit	Estado	Significado
HomingStatus	VERDADERO	El eje vuelve a la posición inicial
JogStatus	FALSO	El eje no funciona ya por impulsos*
MoveStatus	FALSO	El eje ya no se mueve*
GearingStatus	FALSO	El eje ya no acopla
StoppingStatus	FALSO	El eje ya no se detiene

> Durante partes de la secuencia activa de vuelta a la posición inicial, es posible establecer y borrar estos bits. La instrucción MAH utiliza los generadores de perfil de movimiento Impulso y Movimiento para mover el eje durante la secuencia de vuelta a la posición inicial. Esto significa también que cualquier trastorno en los perfiles de movimiento Impulso y Movimiento causado por otras instrucciones de control de movimiento puede afectar la correcta finalización de la secuencia de vuelta a la posición inicial iniciada por una instrucción MAH.

Si está en el modo pasivo de vuelta a la posición inicial, la instrucción MAH simplemente establece el bit Homing Status.

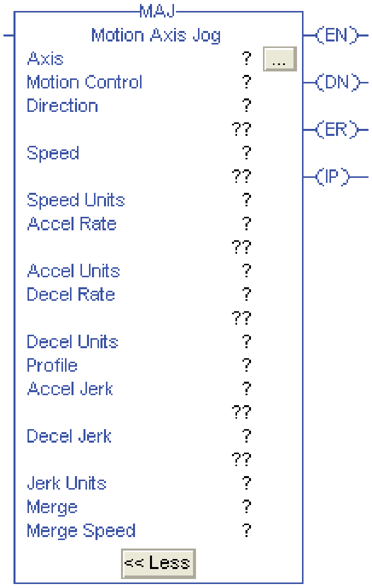
Motion Axis Jog (MAJ)

Utilice la instrucción MAJ para mover un eje a una velocidad constante hasta que usted ordene el paro.

ATENCIÓN ⚠

Utilice un tag de control de movimiento sólo una vez. No vuelva a utilizarlo en otra instrucción. De lo contrario, podría ocasionar un movimiento inesperado del equipo y provocar lesiones a personas.

Operandos: *Lógica de escalera de relés*



MAJ
Motion Axis Jog
Axis ?
Motion Control ?
Direction ?
??
Speed ?
??
Speed Units ?
Accel Rate ?
??
Accel Units ?
Decel Rate ?
??
Decel Units ?
Profile ?
Accel Jerk ?
??
Decel Jerk ?
??
Jerk Units ?
Merge ?
Merge Speed ?

<table>
<tr><th>Operando</th><th>Tipo</th><th>Formato</th><th colspan="2">Descripción</th></tr>
<tr><td>AXIS</td><td>AXIS_VIRTUAL
AXIS_GENERIC
AXIS_SERVO
AXIS_SERVO_DRIVE</td><td>Tag</td><td colspan="2">Nombre del eje por impulsar</td></tr>
<tr><td>Motion Control</td><td>MOTION_INSTRUCTION</td><td>Tag</td><td colspan="2">Tag de control para la instrucción</td></tr>
<tr><td rowspan="3">Direction</td><td rowspan="3">DINT</td><td rowspan="3">Tag
Inmediato</td><td>Para esta dirección de impulso</td><td>Introduzca</td></tr>
<tr><td>Avance</td><td>0</td></tr>
<tr><td>Retroceso</td><td>1</td></tr>
<tr><td>Speed</td><td>REAL</td><td>Tag
Inmediato</td><td colspan="2">Velocidad para mover el eje en unidades de velocidad.</td></tr>
<tr><td>Speed Units</td><td>DINT</td><td>Inmediato</td><td colspan="2">¿Qué unidades desea utilizar para la velocidad?
· Unidades por seg (0)
· % del Máximo (1)</td></tr>
<tr><td>Accel Rate</td><td>REAL</td><td>Tag
Inmediato</td><td colspan="2">Velocidad de aceleración del eje en unidades de aceleración</td></tr>
</table>

Operando	**Tipo**	**Formato**	**Descripción**
Accel Units	DINT	Inmediato	¿Qué unidades desea utilizar para la velocidad de aceleración? · Unidades por seg^2 (0) · % del Máximo (1)
Decel Rate	REAL	Tag Inmediato	Velocidad de desaceleración del eje en unidades de desaceleración.
Decel Units	DINT	Inmediato	¿Qué unidades desea utilizar para la velocidad de desaceleración? · Unidades por seg^2 (0) · % del Máximo (1)
Profile	DINT	Inmediato	Seleccione el perfil de velocidad para ejecutar el impulso: · Trapezoidal (0) · S-curve (1) Para obtener más información, consulte Programar un perfil de velocidad en página 24.
Accel Jerk	REAL	Inmediato o tag	Siempre **debe** introducir valores para los operandos Accel y Decel Jerk. Esta instrucción sólo utiliza los valores si el perfil está configurado como curva en S. · Accel Jerk es el régimen de jaloneo de aceleración del eje. · Decel Jerk es el régimen de jaloneo de desaceleración del eje. Utilice estos valores para comenzar. • Accel Jerk = 100 (% de Tiempo) • Decel Jerk = 100 (% de Tiempo) • Unidades de jaloneo = 2 Introduzca los regímenes de jaloneo en estas unidades de jaloneo. 0 = Unidades por seg^3 1 = % del Máximo 2 = % de Tiempo (utilice este valor para comenzar)
Decel Jerk	REAL	Inmediato o tag	
Jerk Units	DINT	Inmediato	
Merge	DINT	Inmediato	¿Desea convertir todos los movimientos actuales del eje en un impulso puro regido por esta instrucción independientemente de las instrucciones de control de movimiento actualmente en proceso? · NO – Seleccione Inhabilitado (0) · YES – Seleccione Habilitado (1)
Merge Speed	DINT	Inmediato	Si Merge está habilitado, ¿A qué velocidad desea desplazarse por impulsos? · Velocidad de esta instrucción – Seleccione Programmed (0) · Velocidad actual del eje – Seleccione Current (1)

MAJ(Axis,MotionControl,
Direction,Speed, SpeedUnits,
AccelRate, AccelUnits,DecelRate,
DecelUnits,Profile, AccelJerk,DecelJerk,
JerkUnits,Merge, MergeSpeed);

Texto estructurado

Los operandos del texto estructurado son iguales a los operandos de la lógica de escalera de relés.

Este operando	**Tiene estas opciones, las cuales usted**	
	introduce como texto	**o introduce como un número**
AXIS	Sin enumeración	Tag
MotionControl	Sin enumeración	Tag
Direction	Sin enumeración	Inmediato o tag
Speed	Sin enumeración	Inmediato o tag
SpeedUnits	unidades por seg % del máximo	0 1
AccelRate		
AccelUnits	unidades por seg^2 % del máximo	0 1
DecelRate		
DecelUnits	unidades por seg^2 % del máximo	0 1
Profile	trapezoidal scurve	0 1
AccelJerk	Sin enumeración	Inmediato o tag Siempre **debe** introducir un valor para los operandos Accel y Decel Jerk. Esta instrucción sólo utiliza los valores si el perfil está configurado como S-curve. Utilice estos valores para comenzar. • Accel Jerk = 100 (% de Tiempo) • Decel Jerk = 100 (% de Tiempo)
DecelJerk	Sin enumeración	
Jerk Units	unidades por seg^3 % del máximo % de tiempo	0 1 2 (utilice este valor para comenzar)
Merge	disabled habilitado	0 1
MergeSpeed	programada actual	0 1

Tipo de datos MOTION_INSTRUCTION:

Para ver si	Verifique si este bit está establecido	Tipo de datos	Notas
Una transición de falso a verdadero ha provocado la ejecución de la instrucción	EN	BOOL	El bit EN queda establecido hasta que el proceso finaliza y el renglón pasa a falso.
El impulso se ha iniciado correctamente	DN	BOOL	
Ha ocurrido un error	ER	BOOL	
El eje se impulsa	IP	BOOL	Cualquiera de estas acciones detiene este impulso y borra el bit IP: · Otra instrucción MAJ reemplaza esta instrucción MAJ · Instrucción MAS · Incorporación de otra instrucción · Comando de cierre eléctrico · Acción ante un fallo

Descripción: Utilice la instrucción MAJ para mover un eje a una velocidad constante sin tener en cuenta la posición.

Pautas de programación:

ATENCIÓN

Riesgo de sobreimpulso de velocidad y/o de posición final

Si cambia los parámetros de movimiento dinámicamente mediante algún método, es decir, cambiando la dinámica de movimiento (MCD o MCCD) o comenzando una nueva instrucción antes de que haya finalizado la última, tenga en cuenta el riesgo de sobreimpulso de velocidad y/o de posición final.

Un perfil de velocidad trapezoidal puede sobreimpulsar si disminuye la desaceleración máxima mientras el movimiento está desacelerando o está próximo al punto de desaceleración.

Un perfil de velocidad de curva en S puede sobreimpulsar si:

- la desaceleración máxima disminuye mientras el movimiento está desacelerando o está próximo al punto de desaceleración; o
- el jaloneo máximo de aceleración disminuye y el eje se acelera. Sin embargo, recuerde que el jaloneo se puede cambiar indirectamente si está especificado en % de tiempo.

Para obtener más información, consulte Troubleshoot Axis Motion en page 9.

Pauta	Detalles
· **En la lógica de escalera de relés, alterne la condición de renglón cada vez que desee ejecutar la instrucción.**	Ésta es una instrucción transicional: · En la lógica de escalera de relés, alterne la condición de entrada del renglón de borrado a establecido cada vez que desee ejecutar la instrucción.
· **En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición.**	En texto estructurado, las instrucciones se ejecutan cada vez que son escaneadas. · Acondicione la instrucción de manera que sólo se ejecute en una transición. Utilice alguno de estos dos métodos: · calificador o una acción SFC · construcción de texto estructurado Para obtener más información, vea el Apéndice C.
· **Utilice los operandos de jaloneo para los perfiles de curva en S.**	Utilice los operandos de jaloneo cuando la instrucción utiliza un perfil de curva en S. Debe completar los operandos de jaloneo independientemente del perfil.
· **Utilice % de tiempo para facilitar la programación y ajuste del jaloneo.**	Para facilitar la manera de programar y ajustar el jaloneo, introdúzcalo como un % del tiempo de aceleración o desaceleración. Para obtener más información, consulte: · Programar un perfil de velocidad en página 24 · Ajustar un perfil de curva en S en página 391.

<table>
<tr><th>Pauta</th><th>Detalles</th></tr>
<tr><td>• Utilice Merge para cancelar el movimiento de otras instrucciones.</td><td>¿Cómo desea manipular algún movimiento que está ya en proceso?
<table>
<tr><th>Si desea</th><th>Y usted desea</th><th>Establezca</th></tr>
<tr><td>Añadir el impulso a un movimiento ya en proceso</td><td>→</td><td>Merge = Inhabilitado
Merge Speed = Programada
La instrucción ignora Merge Speed, pero aún así debe completarla.</td></tr>
<tr><td>Finalizar el movimiento de otras instrucciones y sólo impulsar</td><td>Impulsar a la velocidad establecida en esta instrucción</td><td>Merge = Habilitado
Merge Speed = Programada</td></tr>
<tr><td></td><td>Impulsar a la velocidad a la cual ya se está moviendo el eje</td><td>Merge = Habilitado
Merge Speed = Actual
La instrucción ignora el valor que usted introdujo en el operando Speed.</td></tr>
</table></td></tr>
<tr><td>• Tenga cuidado si desea iniciar otro impulso mientras el eje ya se está desplazando a impulsos.</td><td>Si desea iniciar una nueva instrucción MAJ mientras una ya está en proceso, puede hacer que:
· un eje en aceleración sobreimpulse su velocidad
· un eje en desaceleración retroceda
· (revisión 15 y anterior)
Esto ocurre si las instrucciones MAJ utilizan un perfil de curva en S.
Causa: La nueva instrucción MAJ cancela la instrucción MAJ anterior. El eje utiliza la velocidad, aceleración, desaceleración y jaloneo de la nueva instrucción.
Para obtener más información, consulte Troubleshoot Axis Motion en page 9.</td></tr>
<tr><td>• Utilice una instrucción MAS para detener el impulso.</td><td>Vea los ejemplos.</td></tr>
<tr><td>• Utilice una instrucción MCD para cambiar la velocidad durante el desplazamiento a impulsos.</td><td>Consulte Ejemplo 1 en página 87.</td></tr>
</table>

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Consulte Códigos de error (ERR) para las instrucciones de control de movimiento en página 395.

Códigos de error extendidos: Utilice los códigos de error extendidos (EXERR) para obtener más información acerca de un error.

<table>
<tr><th>Si ERR es</th><th>Y EXERR es</th><th colspan="2">Entonces</th></tr>
<tr><th></th><th></th><th>Causa</th><th>Acción correctiva</th></tr>
<tr><td>13</td><td>Varía</td><td>Un operando está fuera de su rango.</td><td>El EXERR es el número del operando que está fuera de rango. El primer operando es 0.
Por ejemplo, si EXERR = 3, verifique la velocidad.
<table>
<tr><th>EXERR</th><th>Operando</th></tr>
<tr><td>0</td><td>AXIS</td></tr>
<tr><td>1</td><td>Motion Control</td></tr>
<tr><td>2</td><td>Direction</td></tr>
<tr><td>3</td><td>Speed</td></tr>
</table></td></tr>
<tr><td>54</td><td>–1</td><td>El sistema de coordenadas tiene una desaceleración máxima de 0.</td><td>Diríjase a Properties para el eje del sistema de coordenadas y establezca una desaceleración máxima.</td></tr>
<tr><td></td><td>0 o más</td><td>Un eje del sistema de coordenadas tiene una desaceleración máxima de 0.</td><td>1. Abra las propiedades del eje.
2. Utilice el valor EXERR para ver qué eje tiene la desaceleración máxima de 0.
3. El eje que está desplazando a impulsos tiene una velocidad de desaceleración de 0.</td></tr>
</table>

Cambia a bits de estado: *Bits de estado de movimiento*

Si Merge es	La instrucción cambia estos bits		
	Nombre del bit	**Estado**	**Significado**
Inhabilitado (Disabled)	JogStatus	VERDADERO	El eje se impulsa.
Habilitado	JogStatus	VERDADERO	El eje se impulsa.
	MoveStatus	FALSO	El eje ya no se mueve.
	GearingStatus	FALSO	El eje ya no se acopla.

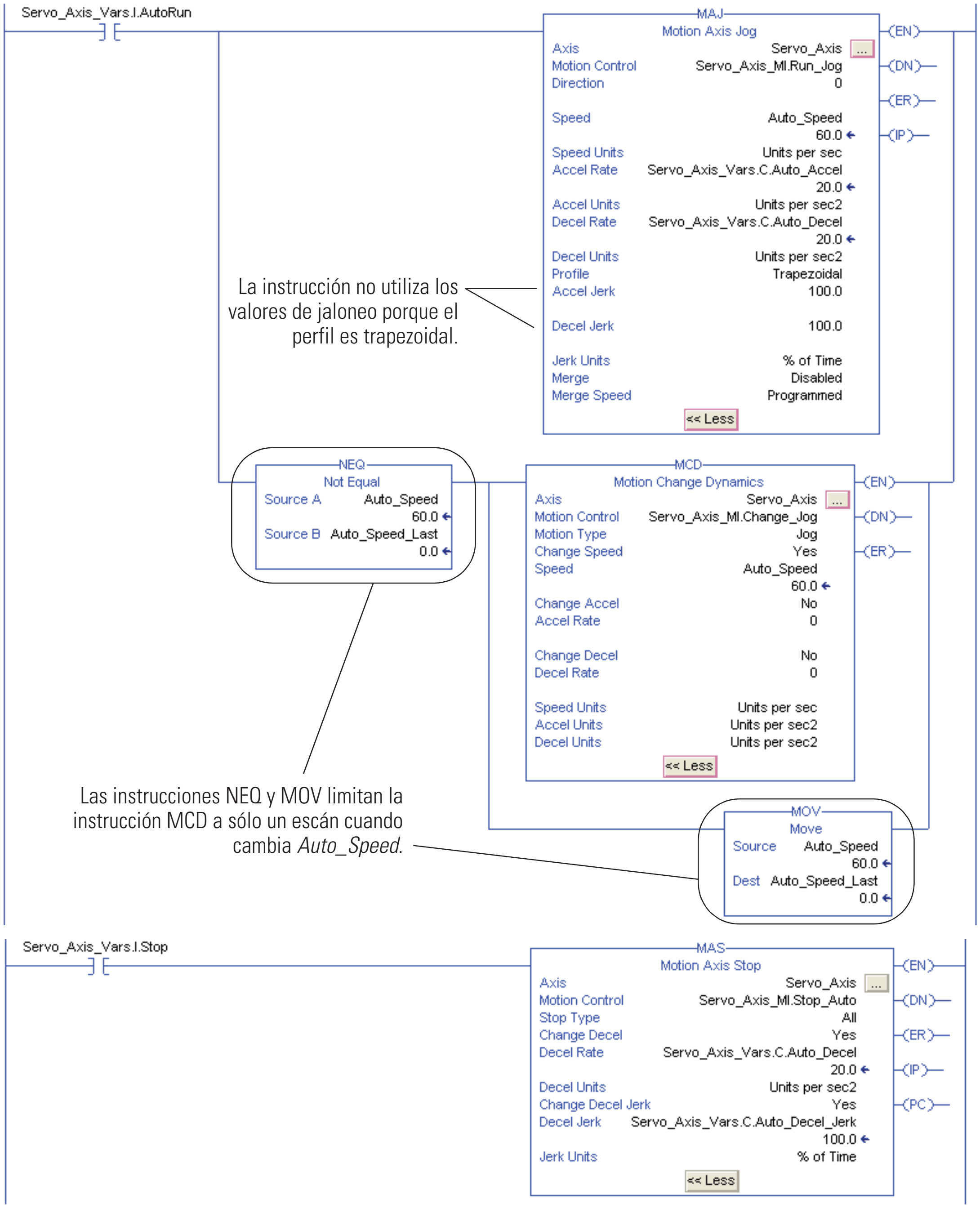
Ejemplo 1 *Impulso con cambio de velocidad – Lógica de escalera de relés*

Cuando *Servo_Axis_Vars.I.AutoRun* se activa

Ejecute *Servo_Axis* a *Auto_Speed.*

Si *Auto_Speed* cambia, entonces cambiará la velocidad del impulso al nuevo valor de *Auto_Speed*.

Cuando *Servo_Axis_Vars.I.Stop* se activa, detenga *Servo_Axis*.



Servo_Axis_Vars.I.AutoRun

MAJ
Motion Axis Jog
Axis Servo_Axis
Motion Control Servo_Axis_MI.Run_Jog
Direction 0
Speed Auto_Speed 60.0
Speed Units Units per sec
Accel Rate Servo_Axis_Vars.C.Auto_Accel 20.0
Accel Units Units per sec2
Decel Rate Servo_Axis_Vars.C.Auto_Decel 20.0
Decel Units Units per sec2
Profile Trapezoidal
Accel Jerk 100.0
Decel Jerk 100.0
Jerk Units % of Time
Merge Disabled
Merge Speed Programmed

La instrucción no utiliza los valores de jaloneo porque el perfil es trapezoidal.

NEQ
Not Equal
Source A Auto_Speed 60.0
Source B Auto_Speed_Last 0.0

MCD
Motion Change Dynamics
Axis Servo_Axis
Motion Control Servo_Axis_MI.Change_Jog
Motion Type Jog
Change Speed Yes
Speed Auto_Speed 60.0
Change Accel No
Accel Rate 0
Change Decel No
Decel Rate 0
Speed Units Units per sec
Accel Units Units per sec2
Decel Units Units per sec2

MOV
Move
Source Auto_Speed 60.0
Dest Auto_Speed_Last 0.0

Las instrucciones NEQ y MOV limitan la instrucción MCD a sólo un escán cuando cambia *Auto_Speed.*

Servo_Axis_Vars.I.Stop

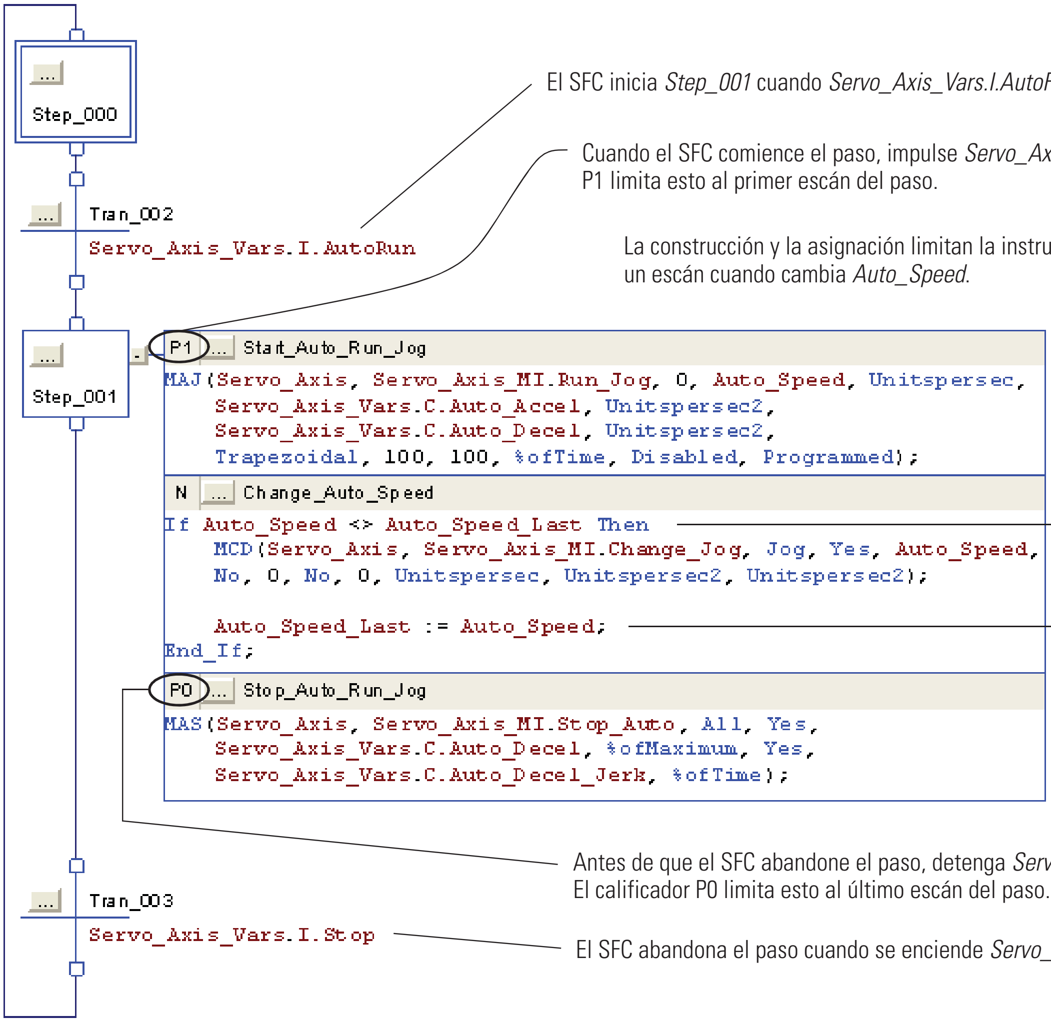
MAS
Motion Axis Stop
Axis Servo_Axis
Motion Control Servo_Axis_MI.Stop_Auto
Stop Type All
Change Decel Yes
Decel Rate Servo_Axis_Vars.C.Auto_Decel 20.0
Decel Units Units per sec2
Change Decel Jerk Yes
Decel Jerk Servo_Axis_Vars.C.Auto_Decel_Jerk 100.0
Jerk Units % of Time

Impulso con cambio de velocidad – Texto estructurado

Cuando *Servo_Axis_Vars.I.AutoRun* se activa

> Ejecute *Servo_Axis* a *Auto_Speed*.
>
> Si *Auto_Speed* cambia, entonces cambiará la velocidad del impulso al nuevo valor de *Auto_Speed*.

Cuando *Servo_Axis_Vars.I.Stop* se activa, detenga *Servo_Axis*.



```
Servo_Axis_Vars.I.AutoRun
```

```
P1 ... Start_Auto_Run_Jog
MAJ(Servo_Axis, Servo_Axis_MI.Run_Jog, 0, Auto_Speed, Unitspersec,
    Servo_Axis_Vars.C.Auto_Accel, Unitspersec2,
    Servo_Axis_Vars.C.Auto_Decel, Unitspersec2,
    Trapezoidal, 100, 100, %ofTime, Disabled, Programmed);
```

```
N ... Change_Auto_Speed
If Auto_Speed <> Auto_Speed_Last Then
    MCD(Servo_Axis, Servo_Axis_MI.Change_Jog, Jog, Yes, Auto_Speed,
    No, 0, No, 0, Unitspersec, Unitspersec2, Unitspersec2);

    Auto_Speed_Last := Auto_Speed;
End_If;
```

```
P0 ... Stop_Auto_Run_Jog
MAS(Servo_Axis, Servo_Axis_MI.Stop_Auto, All, Yes,
    Servo_Axis_Vars.C.Auto_Decel, %ofMaximum, Yes,
    Servo_Axis_Vars.C.Auto_Decel_Jerk, %ofTime);
```

```
Servo_Axis_Vars.I.Stop
```

El SFC inicia *Step_001* cuando *Servo_Axis_Vars.I.AutoRun* se activa.

Cuando el SFC comience el paso, impulse *Servo_Axis*. El calificador P1 limita esto al primer escán del paso.

La construcción y la asignación limitan la instrucción MCD a sólo un escán cuando cambia *Auto_Speed*.

Paro

Antes de que el SFC abandone el paso, detenga *Servo_Axis*. El calificador P0 limita esto al último escán del paso.

El SFC abandona el paso cuando se enciende *Servo_Axis_Vars.I.Stop*.

Ejemplo 2 *Impulso de avance y retroceso con curva en S – Lógica de escalera de relés*

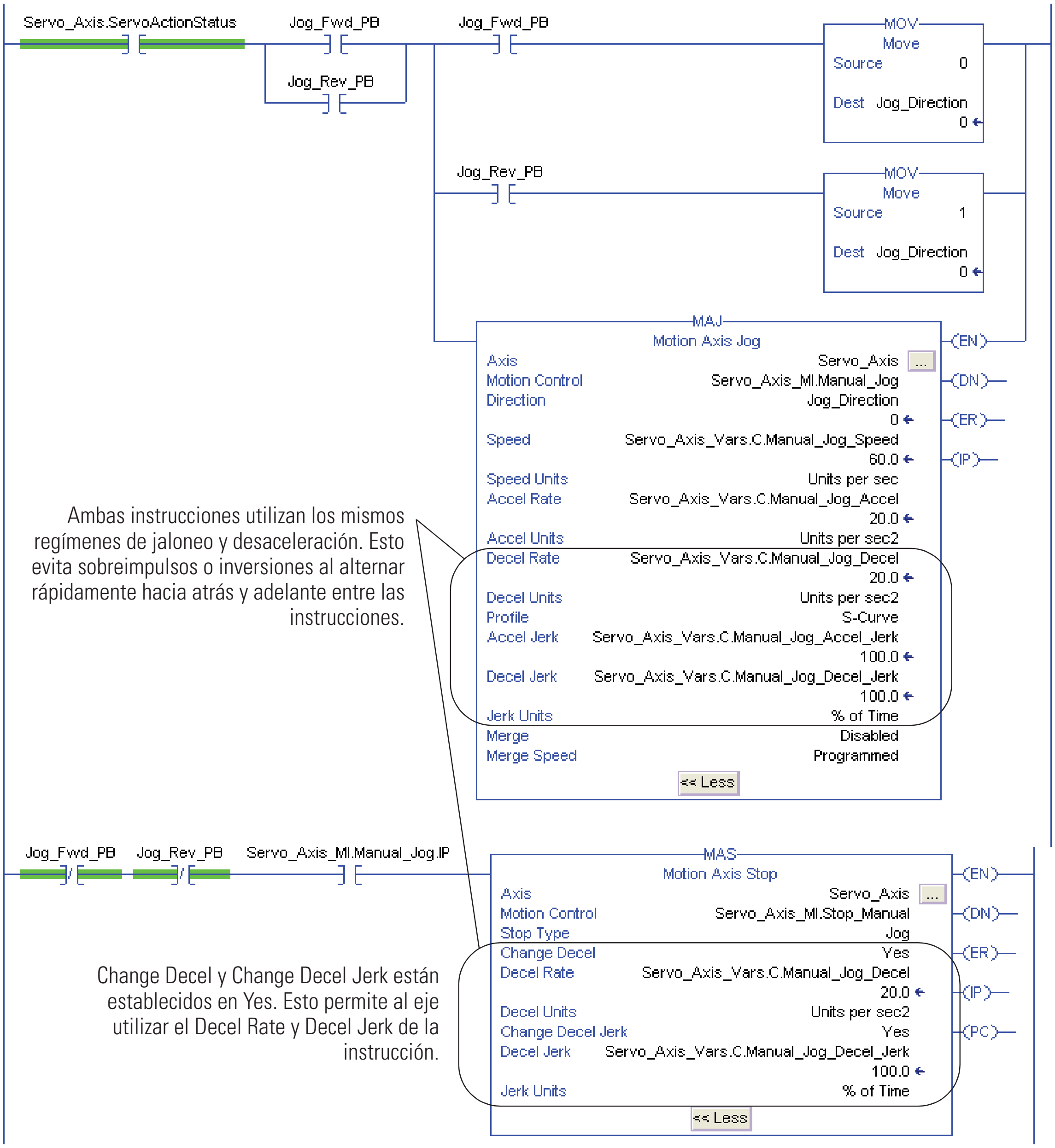
Cuando el lazo del servo está habilitado

Y *Jog_Fwd_PB* o *Jog_Rev_PB* se activa

Establezca *Jog_Direction.*

Ejecute *Servo_Axis* a *Servo_Axis_Vars.C.Manual_Jog_Speed.*

Cuando *Jog_Fwd_PB* y *Jog_Rev_PB* están desactivados, detenga *Servo_Axis*.



MOV Move

Source	0
Dest	Jog_Direction 0

MOV Move

Source	1
Dest	Jog_Direction 0

MAJ Motion Axis Jog

Axis	Servo_Axis
Motion Control	Servo_Axis_MI.Manual_Jog
Direction	Jog_Direction 0
Speed	Servo_Axis_Vars.C.Manual_Jog_Speed 60.0
Speed Units	Units per sec
Accel Rate	Servo_Axis_Vars.C.Manual_Jog_Accel 20.0
Accel Units	Units per sec2
Decel Rate	Servo_Axis_Vars.C.Manual_Jog_Decel 20.0
Decel Units	Units per sec2
Profile	S-Curve
Accel Jerk	Servo_Axis_Vars.C.Manual_Jog_Accel_Jerk 100.0
Decel Jerk	Servo_Axis_Vars.C.Manual_Jog_Decel_Jerk 100.0
Jerk Units	% of Time
Merge	Disabled
Merge Speed	Programmed

Ambas instrucciones utilizan los mismos regímenes de jaloneo y desaceleración. Esto evita sobreimpulsos o inversiones al alternar rápidamente hacia atrás y adelante entre las instrucciones.

MAS Motion Axis Stop

Axis	Servo_Axis
Motion Control	Servo_Axis_MI.Stop_Manual
Stop Type	Jog
Change Decel	Yes
Decel Rate	Servo_Axis_Vars.C.Manual_Jog_Decel 20.0
Decel Units	Units per sec2
Change Decel Jerk	Yes
Decel Jerk	Servo_Axis_Vars.C.Manual_Jog_Decel_Jerk 100.0
Jerk Units	% of Time

Change Decel y Change Decel Jerk están establecidos en Yes. Esto permite al eje utilizar el Decel Rate y Decel Jerk de la instrucción.

Impulso de avance y retroceso con curva en S – Texto estructurado

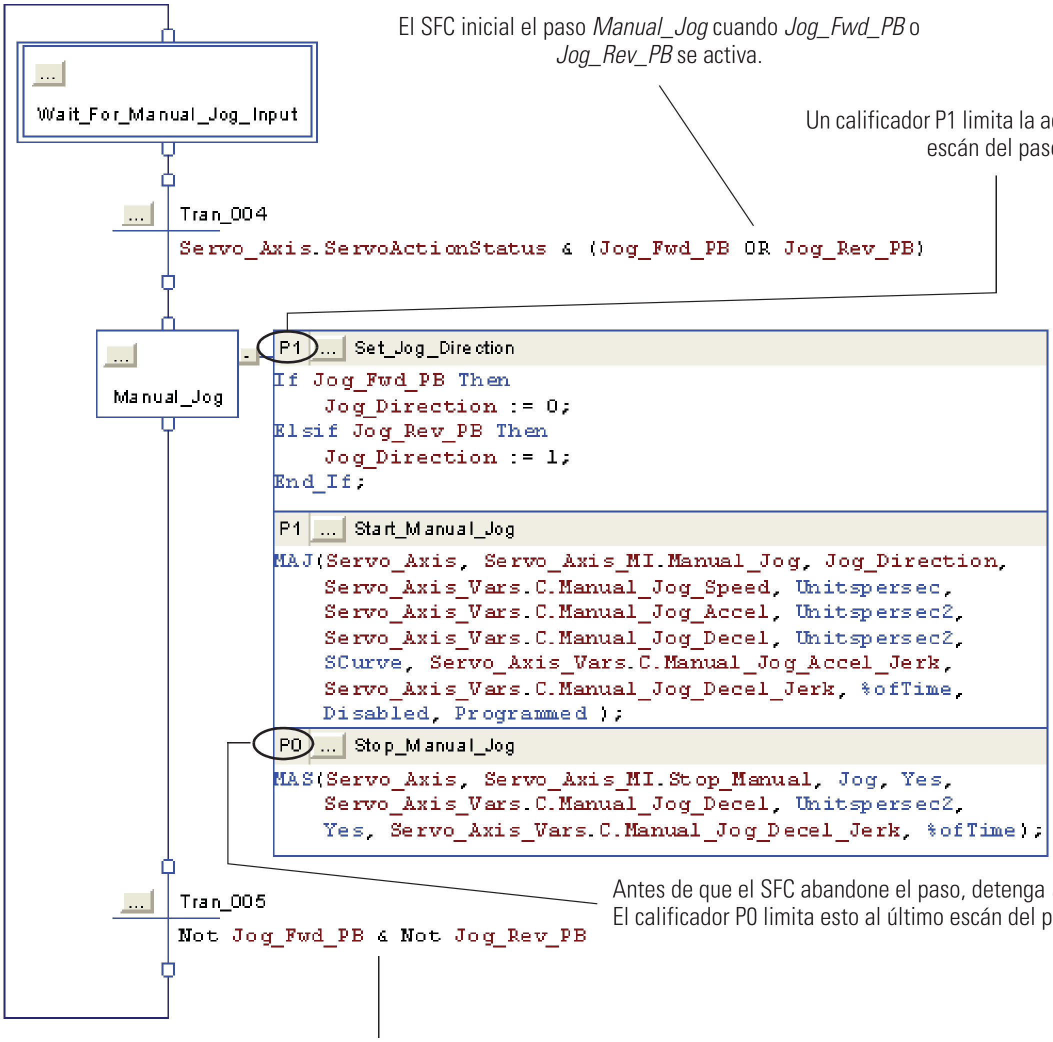
Cuando el lazo del servo está habilitado

Y *Jog_Fwd_PB* o *Jog_Rev_PB* se activa

Establezca *Jog_Direction*.

Ejecute *Servo_Axis* a *Servo_Axis_Vars.C.Manual_Jog_Speed.*

Cuando *Jog_Fwd_PB* y *Jog_Rev_PB* están desactivados, detenga *Servo_Axis*.



```
Wait_For_Manual_Jog_Input

Tran_004
Servo_Axis.ServoActionStatus & (Jog_Fwd_PB OR Jog_Rev_PB)

Manual_Jog

P1  Set_Jog_Direction
If Jog_Fwd_PB Then
    Jog_Direction := 0;
Elsif Jog_Rev_PB Then
    Jog_Direction := 1;
End_If;

P1  Start_Manual_Jog
MAJ(Servo_Axis, Servo_Axis_MI.Manual_Jog, Jog_Direction,
    Servo_Axis_Vars.C.Manual_Jog_Speed, Unitspersec,
    Servo_Axis_Vars.C.Manual_Jog_Accel, Unitspersec2,
    Servo_Axis_Vars.C.Manual_Jog_Decel, Unitspersec2,
    SCurve, Servo_Axis_Vars.C.Manual_Jog_Accel_Jerk,
    Servo_Axis_Vars.C.Manual_Jog_Decel_Jerk, %ofTime,
    Disabled, Programmed );

P0  Stop_Manual_Jog
MAS(Servo_Axis, Servo_Axis_MI.Stop_Manual, Jog, Yes,
    Servo_Axis_Vars.C.Manual_Jog_Decel, Unitspersec2,
    Yes, Servo_Axis_Vars.C.Manual_Jog_Decel_Jerk, %ofTime);

Tran_005
Not Jog_Fwd_PB & Not Jog_Rev_PB
```

El SFC inicial el paso *Manual_Jog* cuando *Jog_Fwd_PB* o *Jog_Rev_PB* se activa.

Un calificador P1 limita la acción al primer escán del paso.

Establezca la dirección.

Inicie el impulso.

Paro.

Antes de que el SFC abandone el paso, detenga *Servo_Axis*. El calificador P0 limita esto al último escán del paso.

El SFC abandona el paso cuando *Jog_Fwd_PB* y *Jog_Rev_PB* están desactivados.

Operando Profile

Al utilizar esta instrucción, deberá tener en cuenta el Operando Profile. Para obtener más información, consulte la página 71.

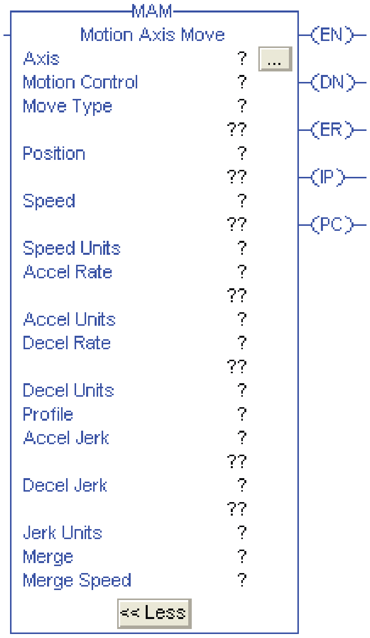
Motion Axis Move (MAM)

Utilice la instrucción MAM para mover un eje a una posición específica.

ATENCIÓN

Utilice un tag de control de movimiento sólo una vez. No vuelva a utilizarlo en otra instrucción. De lo contrario, podría ocasionar un movimiento inesperado del equipo y provocar lesiones a personas.

Operandos: *Lógica de escalera de relés*



Axis ?
Motion Control ?
Move Type ?
??
Position ?
??
Speed ?
??
Speed Units ?
Accel Rate ?
??
Accel Units ?
Decel Rate ?
??
Decel Units ?
Profile ?
Accel Jerk ?
??
Decel Jerk ?
??
Jerk Units ?
Merge ?
Merge Speed ?

Operando	Tipo	Formato	Descripción
AXIS	AXIS_VIRTUAL AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	Tag	Nombre del eje Para un movimiento absoluto u offset de maestro incremental, introduzca el eje esclavo.
Motion Control	MOTION_INSTRUCTION	Tag	Tag de control para la instrucción

Operando	Tipo	Formato	Descripción		
Move Type	DINT	Tag Inmediato	**Para**	**Use este tipo de movimiento**	**E introduzca**
			Mover un eje a una posición absoluta	Absoluto	0
			Mover un eje una distancia específica desde donde se encuentra ahora	Incremental	1
			Mover un eje giratorio a una posición absoluta en la dirección **más corta** independientemente de su posición actual	Giratorio de ruta más corta	2
			Mover un eje giratorio a una posición absoluta en la dirección **positiva** independientemente de su posición actual	Giratorio positivo	3
			Mover un eje giratorio a una posición absoluta en la dirección **negativa** independientemente de su posición actual	Giratorio negativo	4
			Desvía el valor de maestro de una leva de posición a una posición absoluta	Offset de maestro absoluto	5
			Desvía el valor de maestro de una leva de posición mediante una distancia incremental	Offset de maestro incremental	6
			Para obtener más información sobre movimientos giratorios, consulte Seleccione un tipo de movimiento para un eje giratorio en página 98.		
Position	REAL	Inmediato Tag	Posición absoluta o distancia incremental para el movimiento		
			Para este tipo de movimiento	**Introduzca este valor de posición**	
			Absolute	Posición para mover a	
			Incremental	Distancia para mover	
			Giratorio de ruta más corta	Posición para mover a. Introduzca un valor positivo menor que el valor de posición de desbobinado.	
			Giratorio positivo		
			Giratorio de negativo		
			Offset de maestro absoluto	Posición offset absoluta	
			Offset de maestro incremental	Distancia offset incremental	
Speed	REAL	Tag Inmediato	Velocidad para mover el eje en unidades de velocidad.		
Speed Units	DINT	Inmediato	¿Qué unidades desea utilizar para la velocidad? · Unidades por seg (0) · % del Máximo (1)		
Accel Rate	REAL	Tag Inmediato	Velocidad de aceleración del eje en unidades de aceleración		
Accel Units	DINT	Inmediato	¿Qué unidades desea utilizar para la velocidad de aceleración? · Unidades por seg^2 (0) · % del Máximo (1)		

Operando	Tipo	Formato	Descripción
Decel Rate	REAL	Tag Inmediato	Velocidad de desaceleración del eje en unidades de desaceleración.
Decel Units	DINT	Inmediato	¿Qué unidades desea utilizar para la velocidad de desaceleración? • Unidades por seg^2 (0) • % del Máximo (1)
Profile	DINT	Inmediato	Seleccione el perfil de velocidad para ejecutar el movimiento: • Trapezoidal (0) • S-curve (1) Para obtener más información, consulte Programar un perfil de velocidad en página 24.
Accel Jerk	REAL	Tag Inmediato	La instrucción sólo utiliza los operandos de jaloneo si el perfil es curva en S. No obstante, siempre debe completarlos. · Accel Jerk es el régimen de jaloneo de aceleración del eje. · Decel Jerk es el régimen de jaloneo de desaceleración del eje. Utilice estos valores para comenzar. • Accel Jerk = 100 • Decel Jerk = 100 • Unidades de jaloneo = 2 (% de Tiempo) Puede introducir también los regímenes de jaloneo en estas unidades de jaloneo. • Unidades por seg^3 (0) • % del Máximo (1)
Decel Jerk	REAL	Tag Inmediato	
Jerk Units	DINT	Inmediato	
Merge	DINT	Inmediato	¿Desea convertir todos los movimientos actuales del eje en un movimiento puro regido por esta instrucción independientemente de las instrucciones de control de movimiento actualmente en proceso? • NO – Seleccione Inhabilitado (0) • YES – Seleccione Habilitado (1)
Merge Speed	DINT	Inmediato	Si Merge está habilitado, ¿A qué velocidad desea moverse? • Velocidad de esta instrucción – Seleccione Programmed (0) • Velocidad actual del eje – Seleccione Current (1)

Texto estructurado

```
MAM(Axis, MotionControl, MoveType,
Position, Speed, SpeedUnits, AccelRate,
AccelUnits, DecelRate, DecelUnits,
Profile, AccelJerk, DecelJerk, JerkUnits,
Merge, MergeSpeed);
```

Los operandos son iguales a los operandos de lógica de escalera de relés.

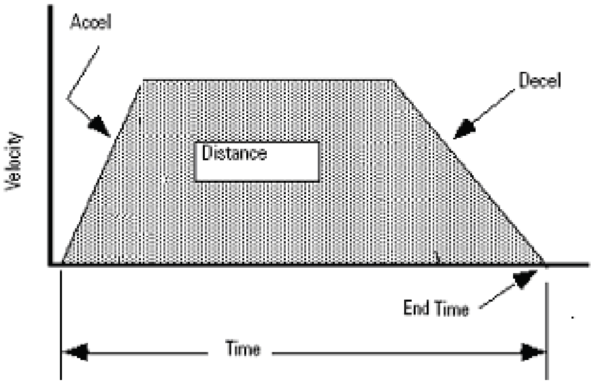
Este operando	**Tiene estas opciones, las cuales usted**	
	introduce como texto	**o introduce como un número**
SpeedUnits	unitspersec	0
	%ofmaximum	1
AccelUnits	unitspersec2	0
	%ofmaximum	1
DecelUnits	unitspersec2	0
	%ofmaximum	1
Profile	trapezoidal	0
	scurve	1
JerkUnits	unitspersec3	0
	%ofmaximum	1
	%oftime	2
Merge	inhabilitado (Disabled)	0
	habilitado	1
MergeSpeed	programada	0
	actual	1

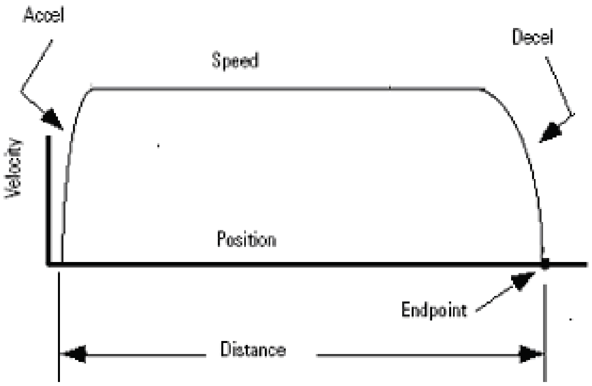
Tipo de datos MOTION_INSTRUCTION:

Para ver si	**Verifique si este bit está establecido**	**Tipo de datos**	**Notas**
Una transición de falso a verdadero ha provocado la ejecución de la instrucción	EN	BOOL	El bit EN queda establecido hasta que el proceso finaliza y el renglón pasa a falso.
El movimiento se ha iniciado correctamente	DN	BOOL	
Ha ocurrido un error	ER	BOOL	
El eje se mueve a la posición final	IP	BOOL	Cualquiera de estas acciones detiene este movimiento y borra el bit IP: · El eje llega a la posición final · Otra instrucción MAM reemplaza esta instrucción MAM · Instrucción MAS · Incorporación de otra instrucción · Comando de cierre eléctrico · Acción ante un fallo
El eje está en la posición final	PC	BOOL	· El bit PC queda establecido hasta que el renglón hace una transición de falso a verdadero. · El bit PC permanece detenido si alguna otra acción detiene el movimiento antes de que el eje llegue a la posición final.

Descripción: La instrucción MAM mueve un eje a una posición absoluta específica o mediante una distancia incremental específica. La instrucción MAM puede producir también otros tipos especiales de movimiento.

Ejemplo: Movimiento trapezoidal que comienza desde el reposo





Pautas de programación:

ATENCIÓN

Riesgo de sobreimpulso de velocidad y/o de posición final

Si cambia los parámetros de movimiento dinámicamente mediante algún método, es decir, cambiando la dinámica de movimiento (MCD o MCCD) o comenzando una nueva instrucción antes de que haya finalizado la última, tenga en cuenta el riesgo de sobreimpulso de velocidad y/o de la posición final.

Un perfil de velocidad trapezoidal puede sobreimpulsar si disminuye la desaceleración máxima mientras el movimiento está desacelerando o está próximo al punto de desaceleración.

Un perfil de velocidad de curva en S puede sobreimpulsar si:

- la desaceleración máxima disminuye mientras el movimiento está desacelerando o está próximo al punto de desaceleración; o
- el jaloneo máximo de aceleración disminuye y el eje se acelera. Sin embargo, recuerde que el jaloneo se puede cambiar indirectamente si está especificado en % de tiempo.

Para obtener más información, consulte Troubleshoot Axis Motion en page 9.

Pauta	Detalles
• **En la lógica de escalera de relés, alterne la condición de renglón cada vez que desee ejecutar la instrucción.**	Ésta es una instrucción transicional: · En la lógica de escalera de relés, alterne la condición de entrada del renglón de borrado a establecido cada vez que desee ejecutar la instrucción.

• **En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición.**	En texto estructurado, las instrucciones se ejecutan cada vez que son escaneadas.
	· Acondicione la instrucción de manera que sólo se ejecute en una transición. Utilice alguno de estos dos métodos:
	  · calificador o una acción SFC
	  · construcción de texto estructurado
	Para obtener más información, vea el Apéndice C.

• **En el caso de un movimiento offset de maestro, introduzca el eje esclavo pero utilice unidades de maestro.**	Utilice un movimiento offset de maestro incremental o absoluto para compensar el valor de maestro de una leva de posición sin cambiar, de hecho, la posición del eje maestro. Esto cambia el perfil de la leva de posición a lo largo del eje maestro. · Para Axis, introduzca el eje esclavo. · Para Position, introduzca la posición offset absoluta o la distancia offset incremental · En el caso de Speed, Acceleration, Deceleration y Jerk, introdúzcalos para el eje maestro. La instrucción suma el offset a los valores de Speed, Acceleration, Deceleration y Jerk.
• **Utilice el % de tiempo para facilitar la programación y ajuste del jaloneo.**	Para facilitar la programación y ajuste del jaloneo, introdúzcalo como % del tiempo de aceleración o desaceleración. Para obtener más información, consulte: · Programar un perfil de velocidad en página 24 · Ajustar un perfil de curva en S en página 391.

<table>
<tr><th>Pauta</th><th>Detalles</th></tr>
<tr><td>

- **Utilice Merge para cancelar el movimiento de otras instrucciones.**

</td><td>

¿Cómo desea manipular algún movimiento que está ya en proceso?

<table>
<tr><th>Si desea</th><th>Y usted desea</th><th>Establezca</th></tr>
<tr><td>Añadir el movimiento a un movimiento ya en proceso</td><td>→</td><td>Merge = Inhabilitado
Merge Speed = Programada
La instrucción ignora Merge Speed, pero aún así debe completarla.</td></tr>
<tr><td>Finalizar el movimiento de otras instrucciones y sólo mover</td><td>Mover a la velocidad establecida en esta instrucción</td><td>Merge = Habilitado
Merge Speed = Programada</td></tr>
<tr><td></td><td>Mover a la velocidad a la cual ya se está moviendo el eje</td><td>Merge = Habilitado
Merge Speed = Actual
La instrucción ignora el valor que usted introdujo en el operando Speed.</td></tr>
</table>

¿Es esto un movimiento offset de maestro incremental o absoluto?

Si es un movimiento offset de maestro incremental o absoluto y Merge está habilitado:

- El movimiento **sólo** finaliza un movimiento offset de maestro incremental o absoluto que está ya en proceso.
- El movimiento **no** afecta ningún otro movimiento que esté ya en proceso.

</td></tr>
<tr><td>

- **Use una segunda instrucción MAM para cambiar una que esté ya en proceso.**

</td><td>

Puede cambiar la posición, velocidad, aceleración o desaceleración. El cambio tiene efecto inmediatamente.

<table>
<tr><th>Para cambiar la posición de</th><th>Establezca una segunda instrucción MAM como ésta.</th></tr>
<tr><td>Movimiento a una posición absoluta</td><td>Realice uno de los siguientes:
· Establezca el tipo de movimiento en Absolute y la posición en la nueva posición.
· Establezca el tipo de movimiento en incremental y establezca la posición a la distancia para cambiar la posición final. La nueva posición final es la posición final anterior más la nueva distancia incremental.
En ambos casos, el eje se mueve hacia la nueva posición sin detenerse en la posición anterior – incluyendo cualquier cambio necesario de dirección.</td></tr>
<tr><td>Movimiento incremental</td><td>Realice uno de los siguientes:
· Establezca el tipo de movimiento en Absolute y la posición en la nueva posición. El eje va directamente a la nueva posición sin completar el movimiento incremental.
· Establezca el tipo de movimiento en incremental y establezca la posición a la distancia adicional. El eje mueve el total de ambos movimientos incrementales.</td></tr>
</table>

</td></tr>
<tr><td>

- **Combine un movimiento con engranaje para perfiles complejos y sincronización.**

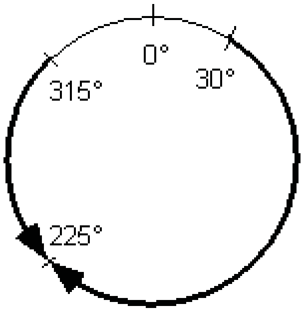
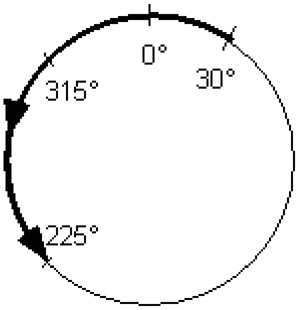
</td><td>

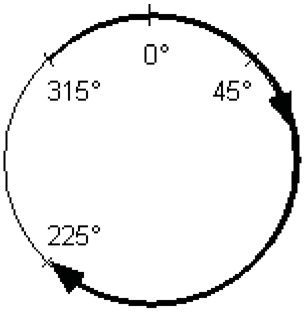
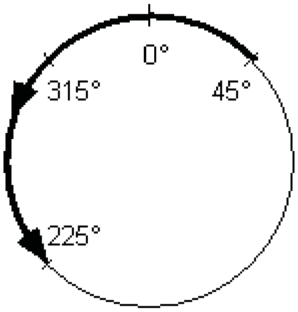
Puede utilizar una instrucción MAG junto con una instrucción MAM. Esto reemplaza el engranaje encima del movimiento o el movimiento encima del engranaje.

Ejemplo: Reemplace un movimiento incremental encima del engranaje electrónico para el avance de fase y el control de retardo.

</td></tr>
</table>

Seleccione un tipo de movimiento para un eje giratorio

Tipo de movimiento	Ejemplo	Descripción
Absolute	Movimiento a una posición absoluta a 225°. La dirección depende de la posición de inicio del eje.  	Con un movimiento Absolute, la dirección del recorrido depende de la posición actual del eje y no es necesariamente la ruta más corta a la posición final. Las posiciones de inicio menores que la posición final dan como resultado un movimiento en dirección positiva, mientras que las posiciones de inicio mayores que la posición anterior dan como resultado un movimiento en dirección negativa. La posición específica se interpreta trigonométricamente y puede ser positiva o negativa. Puede también ser mayor que el valor de posición de desbobinado. Los valores de posición negativa son equivalentes a los valores positivos correspondientes y son útiles al girar el eje hasta 0. Por ejemplo, –90° es lo mismo que +270°. Cuando la posición es mayor o igual al valor de posición de desbobinado, el eje se desplaza a más de una revolución antes de detenerse en una posición absoluta.
Incremental		La distancia específica se interpreta trigonométricamente y puede ser positiva o negativa. Puede también ser mayor que el valor de posición de desbobinado. Cuando la distancia es mayor que el valor de la posición de desbobinado, el eje se desplaza a más de una revolución antes de detenerse.
Giratorio de ruta más corta	Movimiento giratorio de ruta más corta de 30· a 225°.  	**Importante:** Utilice sólo un movimiento giratorio de ruta más corta si el modo de posicionamiento del eje es Rotary (eje giratorio). Un movimiento giratorio de ruta más corta es un tipo especial de movimiento a una posición absoluta para ejes giratorios. El eje: · se mueve a la posición específica en la dirección **más corta** independientemente de su posición actual. · se desplaza a 0° si es necesario. Con un movimiento giratorio de ruta más corta, usted: · puede iniciar el movimiento mientras el eje se está moviendo o está quieto · **no puede** mover el eje más de una revolución con un movimiento simple.

Tipo de movimiento	Ejemplo	Descripción
Giratorio positivo	Movimiento giratorio positivo de 315· a 225°.  	**Importante:** Utilice un movimiento giratorio positivo sólo mientras el eje está quieto y no se está moviendo. De lo contrario, el eje se podría mover en la dirección equivocada. Un movimiento giratorio positivo es un tipo especial de movimiento a una posición absoluta para ejes giratorios. El eje: · se mueve a la posición específica en la dirección **positiva** independientemente de su posición actual. · se desplaza a 0° si es necesario. No puede mover el eje más de una revolución con un único movimiento giratorio de ruta más corta.
Giratorio negativo	Movimiento giratorio negativo de 45· a 225°.  	**Importante:** Utilice un movimiento giratorio de ruta más corta sólo si: · El modo de posicionamiento del eje es Rotary (eje giratorio). · El eje está quieto y no se está moviendo. De lo contrario, el eje se podría mover en la dirección equivocada. Un movimiento giratorio negativo es un tipo especial de movimiento a una posición absoluta para ejes giratorios. El eje: · se mueve a la posición específica en la dirección **negativa** independientemente de su posición actual. · se desplaza a 0° si es necesario. No puede mover el eje más de una revolución con un único movimiento giratorio de ruta más corta.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Consulte Códigos de error (ERR) para las instrucciones de control de movimiento en página 395.

Códigos de error extendidos: Utilice los códigos de error extendidos (EXERR) para obtener más información acerca de un error.

<table>
<tr><th>Si ERR es</th><th>y EXERR es</th><th colspan="2">Entonces</th></tr>
<tr><th></th><th></th><th>Causa</th><th>Acción correctiva</th></tr>
<tr><td>13</td><td>Varía</td><td>Un operando está fuera de su rango.</td><td>El EXERR es el número del operando que está fuera de rango. El primer operando es 0.
Por ejemplo, si EXERR = 4, verifique la velocidad.
<table>
<tr><th>EXERR</th><th>Operando</th></tr>
<tr><td>0</td><td>AXIS</td></tr>
<tr><td>1</td><td>Motion Control</td></tr>
<tr><td>2</td><td>Move Type</td></tr>
<tr><td>3</td><td>Position</td></tr>
<tr><td>4</td><td>Speed</td></tr>
</table></td></tr>
<tr><td>54</td><td>–1</td><td>El sistema de coordenadas tiene una desaceleración máxima de 0.</td><td>Diríjase a Properties para el sistema de coordenadas y establezca una desaceleración máxima.</td></tr>
<tr><td></td><td>0 o más</td><td>Un eje del sistema de coordenadas tiene una desaceleración máxima de 0.</td><td>1. Abra las propiedades del eje.
2. Utilice el valor EXERR para ver qué eje tiene la desaceleración máxima de 0.
3. El eje que está moviendo mediante la instrucción MAM tiene una velocidad de desaceleración de 0.</td></tr>
</table>

Cambia a bits de estado: *Bits de estado de movimiento*

Si Move Type es	Y Merge es	La instrucción cambia estos bits		
		Nombre del bit	**Estado**	**Significado**
NO offset de maestro absoluto u offset de maestro incremental	Inhabilitado	MoveStatus	VERDADERO	El eje se mueve
	Habilitado	MoveStatus	VERDADERO	El eje se mueve
		JogStatus	FALSO	El eje no funciona ya por impulsos
		GearingStatus	FALSO	El eje ya no se acopla
Offset de maestro absoluto u offset de maestro incremental	→	MasterOffsetMoveStatus	VERDADERO	El eje está compensado

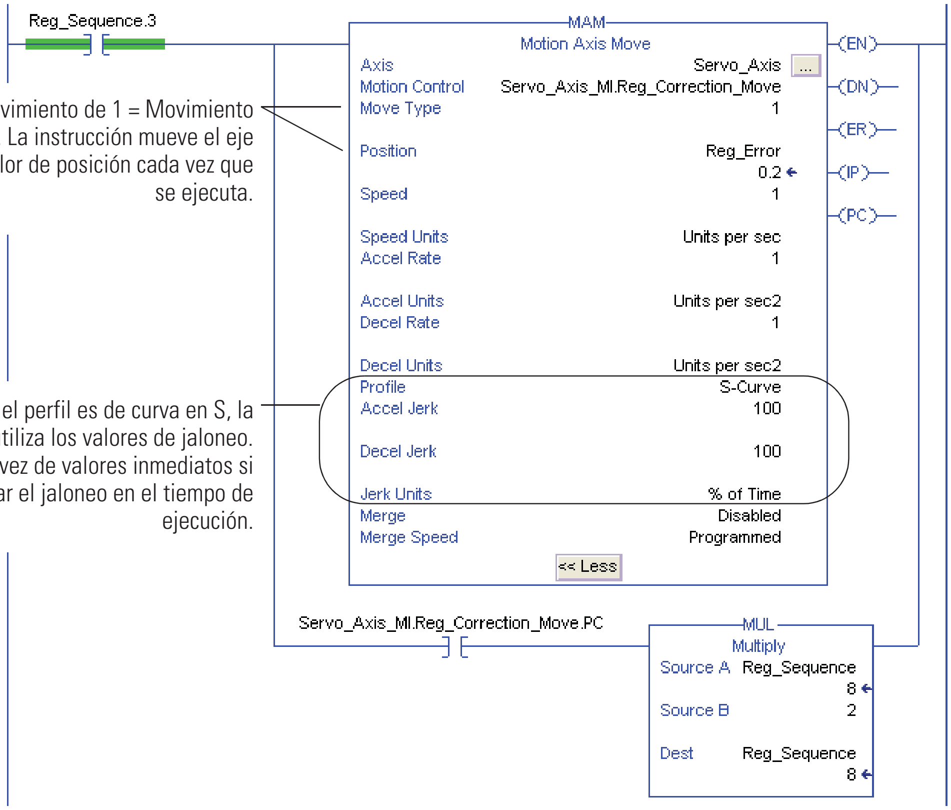
Ejemplo: *Movimiento – Lógica de escalera de relés*

Este ejemplo utiliza el patrón de bits de *Reg_Sequence* para efectuar la lógica paso a paso.

Cuando *Reg_Sequence.3* se activa

Mueva *Servo_Axis* la distancia de *Reg_Error*.

Cuando el movimiento esté completo, multiplique *Reg_Sequence* por 2. Esto desactiva el bit 3 de *Reg_Sequence* y activa el bit 4.



Axis Servo_Axis
Motion Control Servo_Axis_MI.Reg_Correction_Move
Move Type 1
Position Reg_Error 0.2
Speed 1
Speed Units Units per sec
Accel Rate 1
Accel Units Units per sec2
Decel Rate 1
Decel Units Units per sec2
Profile S-Curve
Accel Jerk 100
Decel Jerk 100
Jerk Units % of Time
Merge Disabled
Merge Speed Programmed

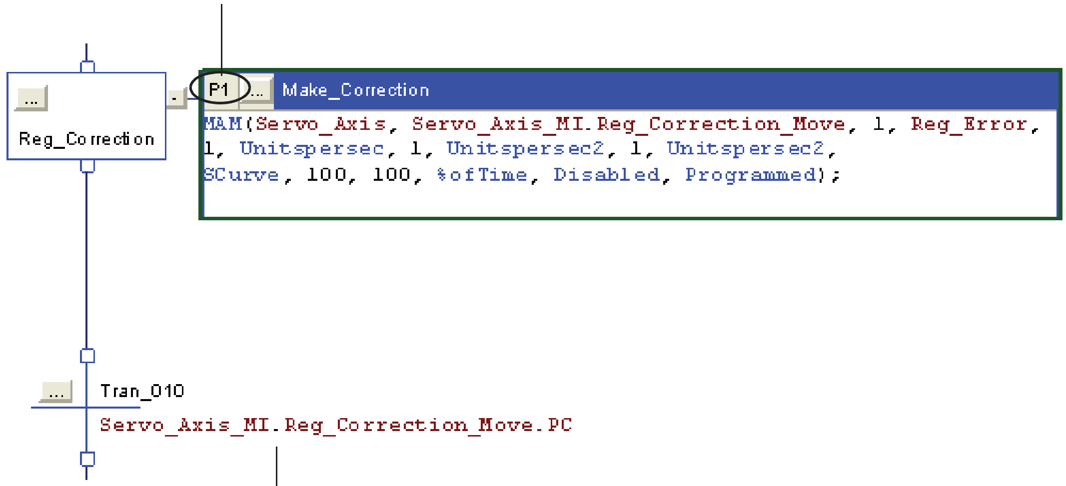
Source A Reg_Sequence 8
Source B 2
Dest Reg_Sequence 8

Un tipo de movimiento de 1 = Movimiento incremental. La instrucción mueve el eje mediante el valor de posición cada vez que se ejecuta.

Puesto que el perfil es de curva en S, la instrucción utiliza los valores de jaloneo. Utilice tags en vez de valores inmediatos si desea ajustar el jaloneo en el tiempo de ejecución.

Movimiento – Texto estructurado

Cuando el SFC inicia el paso *Reg_Correction*, mueva *Servo_Axis* la distancia de *Reg_Error*. El calificador P1 limita esto al primer escán del paso.



```
MAM(Servo_Axis, Servo_Axis_MI.Reg_Correction_Move, 1, Reg_Error,
1, Unitspersec, 1, Unitspersec2, 1, Unitspersec2,
SCurve, 100, 100, %ofTime, Disabled, Programmed);
```

```
Servo_Axis_MI.Reg_Correction_Move.PC
```

El bit PC para el movimiento se activa cuando el movimiento está completo. El SFC abandona el paso cuando el bit PC se activa.

Cómo incorporar en modo incremental

La incorporación para movimientos coordinados funciona de manera diferente que en una MCLM. Para el MAM, todo movimiento incompleto en el momento de la incorporación permanece en el movimiento. Por ejemplo, supongamos que tiene un eje simple con MAM programada en modo incremental desde una posición absoluta de inicio = 0 y con la distancia incremental programada = 4. Si tiene lugar una incorporación a una posición absoluta de 1 y la incorporación es otro movimiento incremental de 4 unidades, el movimiento se completa a una posición = 8.

Si este ejemplo tiene lugar en una MCLM programada en modo incremental, la posición final = 5. Para obtener más información sobre cómo tiene lugar esta incorporación en una MAM programada en modo incremental, consulte página 293.

Operando Profile

Al utilizar esta instrucción, deberá tener en cuenta el Operando Profile. Para obtener más información, consulte la página 71.

Motion Axis Gear (MAG)

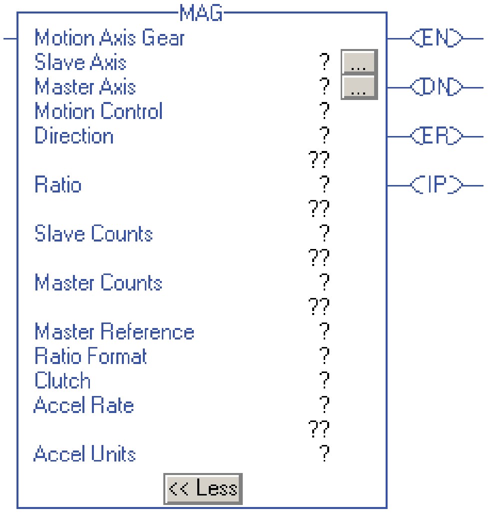
La instrucción MAG proporciona el engranaje electrónico entre dos ejes en una dirección específica y a una relación específica. Al ser llamado, el eje esclavo específico se acopla con el eje maestro a la relación específica (por ejemplo, 1.345) o Conteos esclavos con Conteos maestros (por ejemplo, 1:3). La instrucción MAG es compatible con la especificación de la relación de transmisión en uno de los dos formatos diferentes, Real o Fraccional, como se determina en la selección de entrada de formatos de relación. La dirección del movimiento del eje esclavo que corresponde al eje maestro está definida por un parámetro muy flexible de entrada de dirección. La dirección del engranaje puede establecerse de manera explícita como Same u Opposite o establecerse en relación a la dirección de engranaje actual como Reverse o Unchanged. Tenga en cuenta, además, que el valor para la Relación es sensible a las señales. La selección de referencia de maestro permite derivar la entrada de engranaje desde la posición real o de comando del eje maestro. Cuando la capacidad de embrague de la instrucción se activa, la instrucción de engranaje ordena al eje esclavo acelerar o desacelerar a una velocidad controlada antes de bloquearse en el eje maestro utilizando el valor de aceleración de la instrucción, muy similar al embrague de un coche.

ATENCIÓN

Utilice un tag de control de movimiento sólo una vez. No vuelva a utilizarlo en otra instrucción. De lo contrario, podría ocasionar un movimiento inesperado del equipo y provocar lesiones a personas.

Operandos:

Lógica de escalera de relés


MAG
Motion Axis Gear
Slave Axis ?
Master Axis ?
Motion Control ?
Direction ?
??
Ratio ?
??
Slave Counts ?
??
Master Counts ?
??
Master Reference ?
Ratio Format ?
Clutch ?
Accel Rate ?
??
Accel Units ?

Operando	Tipo	Formato	Descripción
Eje esclavo	AXIS_VIRTUAL AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje en el cual se realizará la operación.
Eje maestro	AXIS_FEEDBACK AXIS_CONSUMED AXIS_VIRTUAL AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	tag	El eje que sigue el eje esclavo.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura que se utiliza para acceder a los parámetros de estado de la instrucción.

Operando	Tipo	Formato	Descripción
Direction	UINT32	Inmediato o tag	Dirección relativa en que el eje esclavo sigue al eje maestro. Seleccione una de las siguientes opciones: 0 = el eje esclavo se mueve en la misma dirección que el eje maestro 1 = el eje esclavo se mueve en la dirección opuesta de su dirección actual 2 = el eje esclavo retrocede de su posición actual o anterior 3 = el eje esclavo continúa en su dirección actual o anterior
Ratio	REAL	Inmediato o tag	Número real con signo que establece la relación de transmisión en unidades de usuario esclavo por unidad de usuario maestro.
Slave counts	UINT32	Inmediato o tag	Número entero que representa los conteos esclavos utilizados para especificar una relación de transmisión fraccional.
Master counts	UINT32	Inmediato o tag	Número entero que representa los conteos maestros utilizados para especificar una relación de transmisión fraccional.
Master reference	BOOLEANO	Inmediato	Establece la referencia de posición maestra a la posición de comando o la posición real. 0 = Real – el movimiento del eje esclavo se genera desde la posición actual del eje maestro que mide su encoder u otro dispositivo de retroalimentación. 1 = de Comando – el movimiento del eje esclavo se genera desde la posición deseada u ordenada del eje maestro.
Ratio format	BOOLEANO	Inmediato	Formato de especificación de relación deseado. Puede seleccionar: 0 = relación de transmisión real 1 = fracción de enteros de los conteos del encoder esclavo a los conteos del encoder maestro

Operando	Tipo	Formato	Descripción
Clutch	BOOLEANO	Inmediato	Cuando Clutch está habilitado, el control de movimiento eleva el eje esclavo hasta la velocidad de engranaje al valor de aceleración definido de la instrucción. Si no está habilitado, el eje esclavo se bloquea inmediatamente en el eje maestro. Si el eje maestro se está moviendo actualmente, esta condición provoca un evento de aceleración incontrolado y brusco del eje esclavo, lo cual puede hacer que el eje entre en fallo. Puede seleccionar: 0 = habilitado 1 = inhabilitado
Accel rate	BOOLEANO	Inmediato o tag	Velocidad de aceleración del eje esclavo en % o unidades de aceleración. Se aplica cuando la característica Clutch está habilitada.
Accel units	DINT	Inmediato	Unidades utilizadas para mostrar el valor de aceleración. Puede seleccionar: 0 = unidades por seg^2 1 = % de aceleración máxima

Texto estructurado

```
MAG(SlaveAxis,MasterAxis,
MotionControl,Direction,
Ratio,SlaveCounts,
MasterCounts,
MasterReference,RatioFormat,Clutch,
AccelRate,
AccelUnits);
```

Los operandos son iguales a los de la instrucción MAG de lógica de escalera de relés.

En el caso de los operandos para los que usted debe seleccionar entre opciones disponibles, introduzca su selección como:

Este operando	Tiene estas opciones, las cuales usted...	
	introduce como texto	**o introduce como un número**
MasterReference	actual	0
	command	1
RatioFormat	REAL	0
	fraction_slave_master_counts	1
Clutch	habilitado	0
	inhabilitado	1
AccelUnits	unitspersec2	0
	%ofmaximum	1

Estructura MOTION_INSTRUCTION

Mnemónico	Descripción
Bit .EN (Habilitado) 31	Se establece cuando el renglón hace una transición de falso a verdadero y permanece establecido hasta que la transacción de mensaje servo se completa y el renglón pasa a falso.
Bit .DN (Listo) 29	Se establece cuando el engranaje del eje se ha iniciado correctamente.
Bit .ER (Error) 28	Se establece para indicar que la instrucción ha detectado un error, tal como especificar un eje no configurado.
Bit .IP (En Proceso) 26	Se establece en transición de renglón positiva y se borra si es reemplazada por otra orden de ejes de engranaje de movimiento, o interrumpida por un comando de paro, incorporación, cierre eléctrico o fallo del servo.

Descripción: La instrucción MAG habilita el engranaje electrónico entre dos ejes a una relación específica. El engranaje electrónico permite sincronizar un eje físico a la posición de comando o real de otro eje físico a una relación precisa. Proporciona un bloqueo directo de flanco a flanco entre los dos ejes – sin utilizar velocidad máxima, límites de aceleración o desaceleración. La velocidad, aceleración y desaceleración del eje esclavo están completamente determinadas por el movimiento del eje maestro y la relación de transmisión especificada.

ATENCIÓN

Los límites máximos de velocidad, aceleración o desaceleración establecidos durante la configuración del eje no se aplican al engranaje electrónico.

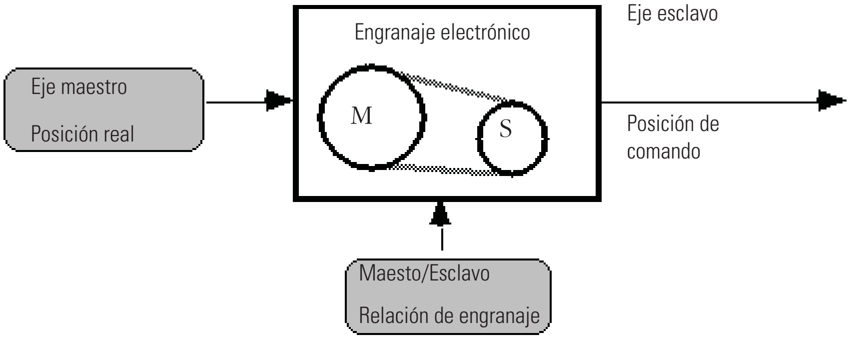
Seleccione o introduzca el eje maestro, eje esclavo y dirección deseados e introduzca un valor o variable de tag para la relación deseada. Si un eje se atenúa (gris) o no se muestra en el menú emergente del eje esclavo, el eje físico no está definido para operaciones de Servo.

Si el eje objeto no aparece en la lista de ejes disponibles, el eje no ha sido configurado para operaciones de servo. Utilice Tag Editor para crear y configurar un nuevo eje.

El engranaje electrónico permanece activo durante la ejecución subsiguiente de procesos de impulso o movimiento para el eje esclavo. Esto permite que los movimientos del engranaje electrónico sean reemplazados con perfiles de impulso o movimiento para crear un movimiento complejo y sincronización.

Cómo generar eje esclavo en la posición real

Cuando se introduce o se selecciona la posición real como la fuente de referencia maestra, el movimiento del eje maestro se genera desde la posición real del eje maestro como se muestra a continuación.

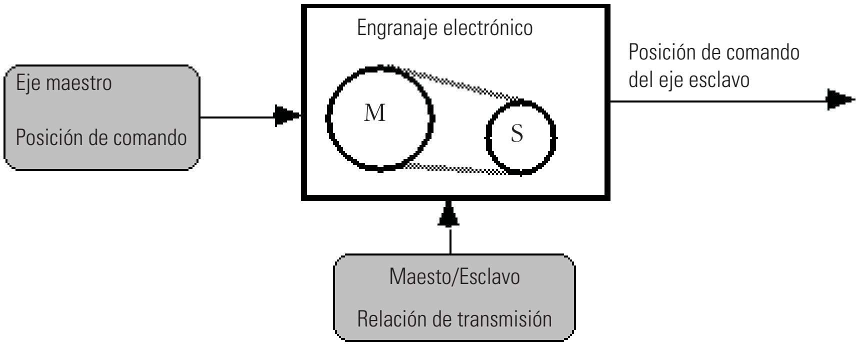


Cómo generar eje esclavo en posición real

La posición real es la posición actual de un eje físico, según la mida el encoder del eje. Ésta es la *única* selección válida cuando el tipo del eje maestro está configurado como de sólo retroalimentación.

Cómo generar eje esclavo en la posición de comando

Cuando se introduce o se selecciona la posición de comando como la fuente de referencia maestra, el movimiento del eje maestro se genera desde la posición de comando del eje maestro como se muestra a continuación.



Cómo generar eje esclavo en posición de comando

La posición de comando (sólo válida cuando el tipo del eje maestro está configurado como Servo) es la posición actual ordenada o deseada para el eje maestro.

Puesto que la posición de comando no incorpora ningún error de seguimiento asociado, perturbaciones externas de posición o ruidos de cuantificación, es una referencia más estable y precisa para el engranaje. Al acoplar en la posición de comando del maestro, se debe *comandar* al eje maestro que se mueva para provocar algún movimiento en el eje esclavo. Consulte la especificación Objeto del eje de movimiento para obtener más información sobre los parámetros de eje de la posición de comando y la posición.

Cómo acoplar en la misma dirección

Cuando se selecciona o se introduce Same como dirección, el eje esclavo se mueve en su posición *positiva* en la relación de transmisión especificada cuando el eje maestro se mueve en su dirección *positiva* y viceversa.

Cómo acoplar en la dirección opuesta

Cuando se selecciona o se introduce Opposite como dirección, el eje esclavo se mueve en su posición *negativa* en la relación de transmisión especificada cuando el eje maestro se mueve en su dirección *positiva* y viceversa.

Cómo cambiar la relación de transmisión

Cuando se selecciona o se introduce Unchanged como dirección, la relación de transmisión se puede cambiar al mismo tiempo que se conserva la dirección de engranaje actual (la misma u opuesta). Esto es útil cuando la dirección actual no es conocida o no es importante.

Cómo invertir la dirección de engranaje

Cuando se selecciona o se introduce Reverse como dirección, la dirección actual del engranaje electrónico se cambia de la misma a la opuesta o de la opuesta a la misma. Esto es muy útil para las aplicaciones de bobinado donde la relación de transmisión debe estar en retroceso en cada extremo de la bobina.

Relaciones de transmisión de números reales

Cuando se introduce o se selecciona Real como formato de relación, la relación de transmisión se especifica como un número real o variable de tag con un valor entre 0.00001 y 9.99999 (inclusive) que representa la relación deseada de las unidades de posición del eje esclavo y las unidades de posición del eje maestro. Una relación de transmisión expresada de esta manera es fácil de interpretar puesto que se define en las unidades de posición configuradas de los ejes.

Relaciones de transmisión de fracción

Cuando se introduce o se selecciona Fraction como formato de relación, la relación de transmisión se especifica como un par de números enteros o variables de tag que representa la relación entre el número de conteos de retroalimentación del eje esclavo y el número de conteos de retroalimentación del eje maestro. Consulte el Generador de variables de tag que se muestra más arriba en este manual para obtener información sobre variables de tag.

IMPORTANTE **No** se usa la constante de conversión introducida como parte del procedimiento de configuración del eje cuando el formato de relación para la instrucción MAG está especificada como una fracción.

Si su relación de transmisión no se puede expresar exactamente como un número real de un máximo de cinco dígitos a la derecha del punto decimal, use la fracción como el formato de relación.

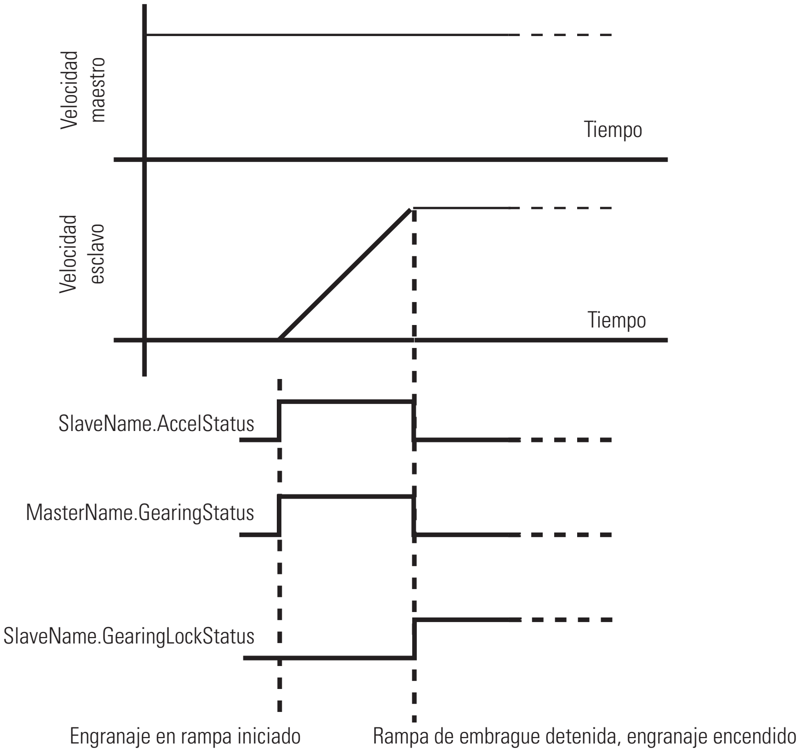
Si la relación de transmisión se especifica como una fracción, es posible implementar directamente las relaciones de transmisión irracionales (como por ejemplo, $^1/_3$) sin errores de posición acumulados o redondeo. Debido a que los valores de conteo esclavo y maestro no utilizan las constantes de conversión del eje y debido a que son números enteros, la relación de transmisión real entre los ejes maestro y esclavo coinciden exactamente con la relación específica.

Por ejemplo, la relación de transmisión irracional de $^1/_3$ se puede especificar de la misma manera como 1 conteo esclavo a 3 conteos maestros, 10 conteos esclavos a 30 conteos maestros, 3 conteos esclavos a 9 conteos maestros.

Clutch (embrague)

Cuando la casilla Clutch está marcada, el eje esclavo acelera o desacelera a la velocidad a la cual se estaría moviendo si se estuviera acoplando actualmente al eje maestro a la relación de transmisión y dirección especificadas utilizando un perfil de velocidad trapezoidal (aceleración o desaceleración lineal). Una vez que el eje esclavo alcanzó la velocidad de engranaje, el engranaje electrónico se activa automáticamente según las otras selecciones. Introduzca la velocidad de aceleración deseada como un porcentaje del valor actual de aceleración máximo configurado o directamente en las unidades para aceleración configuradas por el usuario.

Esta función de embrague funciona de manera muy similar al embrague de un coche, lo cual permite al eje esclavo acoplarse al eje maestro suavemente como se muestra a continuación.



Función de embrague

Al utilizar la característica de embrague, se evita la aceleración o desaceleración descontrolada que ocurre cuando el engranaje electrónico está habilitado mientras el eje maestro se está moviendo. La característica de embrague se puede utilizar también para incorporar rápidamente los cambios de relación de transmisión, incluso los cambios de dirección. El controlador de movimiento eleva automáticamente el eje esclavo a la velocidad que implica el eje maestro con la nueva relación y/o dirección.

El funcionamiento del generador de rampa de embrague no tiene ningún efecto en los procesos de impulso o movimiento que puedan estar en progreso en el eje esclavo.

Cómo cambiar ejes maestros

El eje maestro para engranaje electrónico se puede cambiar en cualquier momento, incluso cuando el engranaje está habilitado. Sin embargo, puesto que es posible habilitar un engranaje electrónico en más de un eje a la vez, si un servoeje maestro y un servoeje esclavo están en retroceso, los ejes se pueden acoplar de manera cruzada y provocar un movimiento inesperado.

Por ejemplo, si está acoplando el eje 0 al eje 1 (definido como un servoeje) y luego desea cambiar a acoplar el eje 1 al eje 0, primero deberá inhabilitar el engranaje en eje 0 (consulte Inhabilitar el engranaje más adelante en esta sección). Esto se debe a que, al especificar el eje 1 como el eje esclavo con el eje 0 como el eje maestro, el eje 0 *no* se inhabilita automáticamente como un eje esclavo con el eje 1 como eje maestro.

Cómo mover durante el engranaje

Una instrucción MAM incremental se puede utilizar para el eje esclavo (o eje maestro si el tipo de eje está configurado como servo) mientras el engranaje electrónico está habilitado. Esto es especialmente útil para llevar a cabo el control de retardo/avance de fase. La distancia de movimiento incremental se puede utilizar para eliminar cualquier error de fase entre el maestro y el esclavo o para crear una relación exacta de fase sin cero. La instrucción MAM incremental se puede usar también en incorporación con un engranaje electrónico para compensar el deslizamiento de material.

Generalmente, una relación de transmisión de 1 se utiliza con ajuste de fase. Una relación de 1:1 garantiza que el error de fase calculado no cambia antes de realizar el movimiento para corregirlo. El engranaje electrónico normalmente no se utiliza con movimientos a una posición absoluta, puesto que no se puede prever el último punto final.

Para ejecutar correctamente una instrucción MAG, el eje específico debe estar configurado como tipo servoeje y el eje debe estar en estado de servo activado. Si no se cumple alguna de estas condiciones, la instrucción dará error.

IMPORTANTE La ejecución de la instrucción MAG se completa en un escán único, de esta manera, el bit Ejecutado (.DN) y el bit En Proceso (.IP) se establecen y el bit Proceso Completo (.PC) se borra inmediatamente. El bit En Proceso (.IP) permanece establecido hasta que el proceso de engranaje iniciado es reemplazado por otra instrucción MAG o interrumpido por un comando de paro del eje de movimiento, operación de incorporación o acción ante un fallo del servo.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético:

no afectados

Condiciones de fallo:

ninguno

Códigos de error:

Consulte Códigos de error (ERR) para las instrucciones de control de movimiento en página 395.

Códigos de error extendidos:

Los códigos de error extendidos proporcionan información adicional específica de instrucción para los códigos de error que son genéricos a muchas instrucciones.

Los códigos de error extendidos para el código de error Axis Not Configured (11) son los siguientes:

- Código de error extendido 1 significa que el eje esclavo no está configurado.
- Código de error extendido 2 significa que el eje maestro no está configurado.

Los códigos de error extendidos para el código de error Parámetro fuera de rango (13) incluyen un número que hace referencia al número de operando tal y como aparecen en la plantilla de arriba hacia abajo, con el primer operando como cero. Por lo tanto, para la instrucción MAG, un código de error extendido de 4 haría referencia al valor del operando Ratio. Entonces, debería verificar su valor con el rango aceptado de valores para la instrucción.

Para el código de error 54 – El valor máximo de desaceleración es cero, si el error extendido devuelve un número positivo (0-*n*), está haciendo referencia al eje en error en el sistema de coordenadas. Diríjase a la ficha general Coordinate System Properties y mire debajo de la columna de corchetes ([ ]) de la cuadrícula del eje para determinar qué eje tiene un valor máximo de desaceleración de cero. Haga clic en el botón de elipsis al lado del eje en error para acceder a la pantalla Axis Properties. Diríjase a la ficha Dynamics y haga el cambio apropiado al valor máximo de desaceleración. Si el número del error extendido es –1, esto significa que el sistema de coordenadas tiene un valor máximo de desaceleración de 0. Diríjase a la ficha Coordinate System Properties Dynamics para corregir el valor máximo de desaceleración.

Bits de estado:

MAG cambia a bits de estado

Si la casilla Clutch NO está marcada, la ejecución de la instrucción MAG simplemente establece el bit Gear Status en verdadero.

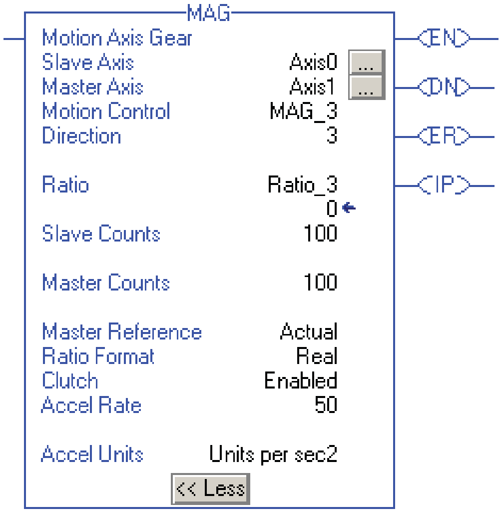
Nombre del bit	Estado	Significado
GearingStatus	VERDADERO	El eje se está acoplando

Si la casilla Clutch está marcada, la ejecución de la instrucción MAG establece el bit Gearing Lock Status en verdadero cuando se completa el proceso de embragado.

Nombre del bit	Estado	Significado
Estado de bloqueo de engranaje	VERDADERO	El eje ha terminado el embrague y se bloqueó.
GearingStatus	VERDADERO	El eje se está acoplando

Ejemplo: Cuando las condiciones de entrada son verdaderas, el controlador proporciona un engranaje electrónico entre *eje2* y *eje1*.

Lógica de escalera de relés



MAG
Motion Axis Gear
Slave Axis Axis0
Master Axis Axis1
Motion Control MAG_3
Direction 3
Ratio Ratio_3
0
Slave Counts 100
Master Counts 100
Master Reference Actual
Ratio Format Real
Clutch Enabled
Accel Rate 50
Accel Units Units per sec2

Ejemplo de lógica de escalera MAG

Texto estructurado

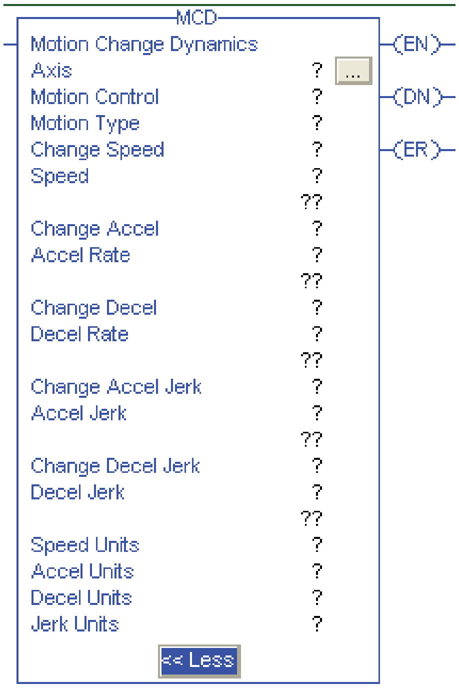
MAG(Axis0,Axis1,MAG_3,3,Ratio_3,0,100,100,Actual,Real,
Enabled,50,Unitspersec2);

Motion Change Dynamics (MCD)

Use la instrucción MCD para cambiar de manera selectiva la velocidad, la velocidad de aceleración o desaceleración de un perfil de movimiento o un perfil de impulso en proceso.

ATENCIÓN Utilice un tag de control de movimiento sólo una vez. No vuelva a utilizarlo en otra instrucción. De lo contrario, podría ocasionar un movimiento inesperado del equipo y provocar lesiones a personas.

Operandos: *Lógica de escalera de relés*



MCD
Motion Change Dynamics
Axis ?
Motion Control ?
Motion Type ?
Change Speed ?
Speed ?
??
Change Accel ?
Accel Rate ?
??
Change Decel ?
Decel Rate ?
??
Change Accel Jerk ?
Accel Jerk ?
??
Change Decel Jerk ?
Decel Jerk ?
??
Speed Units ?
Accel Units ?
Decel Units ?
Jerk Units ?

Operando	Tipo	Formato	Descripción
AXIS	AXIS_VIRTUAL AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje en el cual se realizará la operación.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura que se utiliza para acceder a los parámetros de estado de la instrucción.
Motion type	UDINT	Inmediato	Perfil de movimiento (impulso o movimiento) por cambiar. Puede seleccionar: 0 = impulso 1 = movimiento
Change speed	BOOLEANO	Inmediato	Se establece para habilitar un cambio de velocidad. Puede seleccionar: 0 = no 1 = sí
Speed	REAL	Inmediato o tag	La nueva velocidad para mover el eje en % o unidades de velocidad.
Change Accel	BOOLEANO	Inmediato	Se establece para habilitar un cambio de aceleración. Puede seleccionar: 0 = no 1 = sí
Accel rate	REAL	Inmediato o tag	Velocidad de aceleración del eje en % o unidades de aceleración.
Change Decel	BOOLEANO	Inmediato	Se establece para habilitar un cambio de desaceleración. Puede seleccionar: 0 = no 1 = sí

Operando	**Tipo**	**Formato**	**Descripción**
Decel rate	REAL	Inmediato o tag	Velocidad de desaceleración del eje en % o unidades de desaceleración. El eje podría sobreimpulsar su posición específica si reduce la desaceleración mientras un movimiento está en proceso.
Change Accel Jerk	SINT, INT o DINT	Inmediato	0 = No 1 = Sí
Accel Jerk	SINT, INT, DINT o REAL	Inmediato o tag	Siempre **debe** introducir un valor para el operando Accel Jerk. Esta instrucción sólo utiliza el valor si el perfil está configurado como curva en S. · Accel Jerk es el régimen de jaloneo de aceleración del eje. Utilice este valor para comenzar. Accel Jerk = 100 (% de Tiempo) Unidades de jaloneo = 2
Change Decel Jerk	SINT, INT o DINT	Inmediato	0 = No 1 = Sí
Decel Jerk	SINT, INT, DINT o REAL	Inmediato o Tag	Siempre **debe** introducir un valor para el operando Decel Jerk. Esta instrucción sólo utiliza el valor si el perfil está configurado como curva en S. · Decel Jerk es el régimen de jaloneo de desaceleración para el sistema de coordenadas. Utilice estos valores para comenzar. Decel Jerk = 100 (% de Tiempo) Unidades de jaloneo = 2
SpeedUnits	BOOLEANO	Inmediato	Unidades utilizadas para mostrar el valor de velocidad. Puede seleccionar: 0 = unidades por seg 1 = % de velocidad máxima

Operando	Tipo	Formato	Descripción
Accel units	BOOLEANO	Inmediato	Unidades utilizadas para mostrar el valor de aceleración. Puede seleccionar: 0 = unidades por seg^2 1 = % de aceleración máxima
Decel units	BOOLEANO	Inmediato	Unidades utilizadas para mostrar el valor de desaceleración. Puede seleccionar: 0 = unidades por seg^2 1 = % de desaceleración máxima
Jerk Units	SINT, INT o DINT	Inmediato	0 = Unidades por seg^3 1 = % del Máximo 2 = % de tiempo

Texto estructurado

Los operandos son iguales a los de la instrucción MCD de lógica de escalera de relés.

En el caso de los operandos para los que usted debe seleccionar entre opciones disponibles, introduzca su selección como:

```
MCD(Axis,MotionControl,
MotionType,ChangeSpeed,
Speed,ChangeAccel,AccelRate,
ChangeAccelJerk,AccelJerk,
ChangeDecelJerk,ChangeDecel,
DecelRate,SpeedUnits,AccelUnits,
DecelUnits,JerkUnits);
```

Este operando	Tiene estas opciones, las cuales usted…	
	introduce como texto	**o introduce como**
AXIS	Sin enumeración	Tag
MotionControl	Sin enumeración	Tag
MotionType	impulso move	0 1
ChangeSpeed	no sí	0 1
Speed	Sin enumeración	Inmediato o tag
ChangeAccel	no sí	0 1
AccelRate	Sin enumeración	Inmediato o tag
ChangeDecel	no sí	0 1
Decel Rate	Sin enumeración	Inmediato o tag
ChangeAccelJerk	Sin enumeración	0 = No 1 = Sí

Este operando	**Tiene estas opciones, las cuales usted...**	
	introduce como texto	**o introduce como**
AccelJerk	Sin enumeración	Inmediato o tag Siempre **debe** introducir un valor para el operando Accel Jerk. Esta instrucción sólo utiliza el valor si el perfil está configurado como curva en S. Utilice este valor para comenzar. Accel Jerk = 100
ChangeDecelJerk	Sin enumeración	0 = No 1 = Sí
DecelJerk	Sin enumeración	Inmediato o tag Siempre **debe** introducir un valor para el operando Decel Jerk. Esta instrucción sólo utiliza el valor si el perfil está configurado como curva en S. Utilice este valor para comenzar. Decel Jerk = 100
SpeedUnits	unitspersec	0
	%ofmaximum	1
AccelUnits	unitspersec2	0
	%ofmaximum	1
DecelUnits	unitspersec2	0
	%ofmaximum	1
JerkUnits	Unitspersec^3	0
	%ofmaximum	1
	%oftime	2

Estructura MOTION_INSTRUCTION

Mnemónico	Descripción
Bit .EN (Habilitado) 31	Se establece cuando el renglón hace una transición de falso a verdadero y permanece establecido hasta que la transacción de mensaje servo se completa y el renglón pasa a falso.
Bit .DN (Listo) 29	Se establece cuando la dinámica de cambio del eje se ha iniciado correctamente. La ejecución de la instrucción se completa en un escán individual y el bit DN se establece inmediatamente.
Bit .ER (Error) 28	Se establece para indicar que la instrucción ha detectado un error, tal como especificar un eje no configurado.

Descripción: La instrucción MCD cambia rápidamente la velocidad de los movimientos del perfil trapezoidal y la velocidad, aceleración y desaceleración de los impulsos de perfil trapezoidal rápidamente. Seleccione el eje físico deseado y el tipo de movimiento e introduzca los valores o variables de tag para Speed, Accel y Decel. Los valores de velocidad, aceleración y desaceleración se pueden introducir como porcentajes del máximo actual configurado o directamente en unidades configuradas de velocidad o aceleración del eje.

Si el eje específico no aparece en la lista de ejes disponibles, el eje no ha sido configurado para operaciones de Servo. Utilice Tag Editor para crear y configurar un nuevo eje.

ATENCIÓN

Riesgo de sobreimpulso de velocidad y/o de posición final

Si cambia los parámetros de movimiento dinámicamente mediante algún método, es decir, cambiando la dinámica de movimiento (MCD o MCCD) o comenzando una nueva instrucción antes de que haya finalizado la última, tenga en cuenta el riesgo de sobreimpulso de velocidad y/o de la posición final.

Un perfil de velocidad trapezoidal puede sobreimpulsar si disminuye la desaceleración máxima mientras el movimiento está desacelerando o está próximo al punto de desaceleración.

Un perfil de velocidad de curva en S puede sobreimpulsar si:

- la desaceleración máxima disminuye mientras el movimiento está desacelerando o está próximo al punto de desaceleración; o
- el jaloneo máximo de aceleración disminuye y el eje se acelera. Sin embargo, recuerde que el jaloneo se puede cambiar indirectamente si está especificado en % de tiempo.

Para obtener más información, consulte Troubleshoot Axis Motion en page 9.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Cómo cambiar la dinámica de movimiento

Cuando un tipo de movimiento se introduce o selecciona como movimiento, la velocidad, aceleración y/o desaceleración de un movimiento en progreso se puede cambiar a un valor especificado. El cambio de velocidad tiene lugar a la velocidad de aceleración especificada si la nueva velocidad es más alta que la velocidad actual o a la velocidad de desaceleración especificada si la nueva velocidad es más baja que la velocidad actual.

Cómo poner en pausa los movimientos

La instrucción MCD se puede utilizar para poner en pausa de manera temporaria un movimiento en progreso cambiando su velocidad a cero. Utilice otra instrucción MCD con valor de velocidad que no sea cero para completar el movimiento como se especificó originariamente.

Cómo cambiar la dinámica de impulso

Cuando un tipo de movimiento se introduce o selecciona como impulso, la velocidad, aceleración y/o desaceleración de un impulso en progreso se puede cambiar a un valor específico. El cambio de velocidad tiene lugar a la velocidad de aceleración especificada si la nueva velocidad es más alta que la velocidad actual o a la velocidad de desaceleración especificada si la nueva velocidad es más baja que la velocidad actual.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Consulte Códigos de error (ERR) para las instrucciones de control de movimiento en página 395.

Códigos de error extendidos: Los códigos de error extendidos proporcionan información adicional específica de instrucción para los códigos de error que son genéricos a muchas instrucciones.

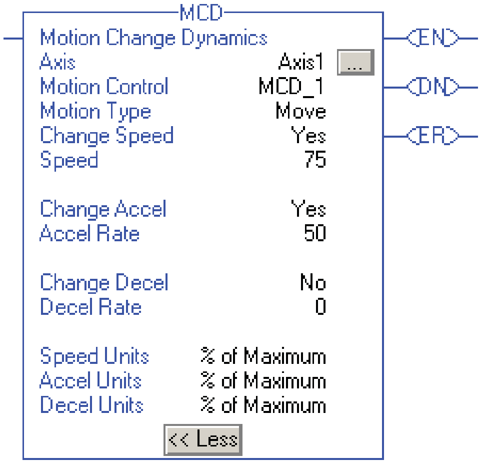
Los códigos de error extendidos para el código de error Parámetro fuera de rango (13) incluyen un número que hace referencia al número de operando, puesto que aparecen en la plantilla de arriba hacia abajo, con el primer operando como cero. Por lo tanto, para la instrucción MCD, un código de error extendido de 4 haría referencia al valor del operando Speed. Entonces, debería verificar su valor con el rango de valores aceptado para la instrucción.

Para el código de error 54 – El valor máximo de desaceleración es cero, si el error extendido devuelve un número positivo (0-*n*), está haciendo referencia al eje en error en el sistema de coordenadas. Diríjase a la ficha general Coordinate System Properties y mire debajo de la columna de corchetes ([ ]) de la cuadrícula del eje para determinar qué eje tiene un valor máximo de desaceleración de cero. Haga clic en el botón de elipsis al lado del eje en error para acceder a la pantalla de propiedades del eje. Diríjase a la ficha Dynamics y haga el cambio apropiado al valor máximo de desaceleración. Si el número del error extendido es –1, esto significa que el sistema de coordenadas tiene un valor máximo de desaceleración de 0. Diríjase a la ficha Coordinate System Properties Dynamics para corregir el valor máximo de desaceleración.

MCD cambia a bits de estado: ninguno

Ejemplo: Cuando las condiciones de entrada son verdaderas, el controlador cambia la velocidad, la velocidad de aceleración o desaceleración de un perfil de movimiento o un perfil de impulso en progreso para *eje1*.

Lógica de escalera de relés



MCD
Motion Change Dynamics
Axis Axis1
Motion Control MCD_1
Motion Type Move
Change Speed Yes
Speed 75

Change Accel Yes
Accel Rate 50

Change Decel No
Decel Rate 0

Speed Units % of Maximum
Accel Units % of Maximum
Decel Units % of Maximum

Ejemplo de lógica de escalera de MCD

Texto estructurado

MCD(Axis1,MCD_1,Move,Yes,75,Yes,50,No,0,%ofmaximum,
%ofmaximum,%ofmaximum);

Operando Profile

Al utilizar esta instrucción, deberá tener en cuenta el Operando Profile. Para obtener más información, consulte la página 71.

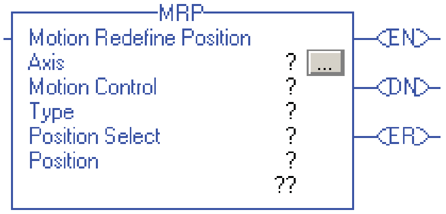
Motion Redefine Position (MRP)

Use la instrucción MRP para cambiar la posición de comando o la posición real de un eje. El valor especificado por Position se usa para actualizar la posición real o de comando del eje. La redefinición de la posición se puede calcular en una base absoluta o relativa. Si se selecciona absoluta, el valor de Position se asigna a la posición actual de comando o real. Si se selecciona relativa, el valor de Position se añade como un desplazamiento a la posición actual de comando o real. El proceso de redefinición de la posición del eje actual no afecta el movimiento en progreso puesto que la instrucción conserva el error de seguimiento del servo actual durante el proceso de redefinición. Como resultado, la posición del eje se puede redefinir rápidamente sin perturbar el movimiento del eje.

ATENCIÓN

Utilice un tag de control de movimiento sólo una vez. No vuelva a utilizarlo en otra instrucción. De lo contrario, podría ocasionar un movimiento inesperado del equipo y provocar lesiones a personas.

Operandos: *Lógica de escalera de relés*



Operando	Tipo	Formato	Descripción
AXIS	AXIS_FEEDBACK AXIS_VIRTUAL AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje en el cual se realizará la operación.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura que se utiliza para acceder a los parámetros de la instrucción.

Operando	Tipo	Formato	Descripción
Tipo	BOOLEANO	Inmediato	La manera en la desea que funcione la operación de redefinición. Puede seleccionar: 0 = absoluta 1 = relativa
Position select	BOOLEANO	Inmediato	Seleccione la posición en la cual realizar la operación de redefinición. Puede seleccionar: 0 = posición real 1 = posición de comando
Position	REAL	Inmediato o tag	Valor a utilizar para cambiar la posición del eje o para desviar a la posición actual.

MRP(Axis,MotionControl,Type,
PositionSelect,Position);

Texto estructurado

Los operandos son iguales a los de la instrucción MRP de lógica de escalera de relés.

En el caso de los operandos para los que usted debe seleccionar entre opciones disponibles, introduzca su selección como:

Este operando	Tiene estas opciones, las cuales usted...	
	introduce como texto	**o introduce como un número**
Tipo	absoluto	0
	relativo	1
PositionSelect	real	0
	comando	1

Estructura MOTION_INSTRUCTION

Mnemónico	Descripción
Bit .EN (Habilitado) 31	Se establece cuando el renglón hace una transición de falso a verdadero y permanece establecido hasta que la transacción de mensaje servo se completa y el renglón pasa a falso.
Bit .DN (Listo) 29	Se establece cuando la posición del eje ha sido redefinida correctamente.
Bit .ER (Error) 28	Se establece para indicar que la instrucción ha detectado un error, tal como especificar un eje no configurado.

Descripción: La instrucción MRP establece directamente la posición real o de comando del eje especificado a la posición relativa o absoluta especificada. Esta instrucción no provoca ningún cambio – la posición actual del eje se redefine fácilmente. Seleccione o introduzca las opciones deseadas para Axis, Type, Position Selection e introduzca un valor o variable de tag para la nueva posición deseada.

Si el eje específico no aparece en la lista de ejes disponibles, el eje no ha sido configurado para la operación. Utilice Tag Editor para crear y configurar un nuevo eje.

La instrucción MRP se puede usar mientras el eje se está moviendo como así también cuando está en reposo. MRP se usa para redefinir la posición "rápidamente" en ciertas aplicaciones de registro, compensación de deslizamiento y re-calibrado.

Modo absoluto

Cuando se introduce o selecciona Absolute como el tipo de MRP, la nueva posición especifica la nueva posición *absoluta* del eje. No tiene lugar ningún movimiento – la posición del eje actual (real o de comando) se redefine simplemente en la nueva posición especificada.

Si se usan los límites de sobrecarrera de software (consulte la especificación Objeto del eje de movimiento para obtener más información sobre la configuración de sobrecarrera del software), la nueva posición debe estar entre los valores de configuración de recorrido negativo máximo y positivo máximo. De lo contrario, se genera un fallo de sobrecarrera de software cuando se ejecuta la instrucción.

ATENCIÓN ⚠

Si la seña de comprobación del límite de sobrecarrera de software está en efecto, la ejecución de una MRP en modo absoluto puede invalidar los límites actuales de carrera negativo máximo y positivo máximo en sentido absoluto. Tenga precaución al redefinir la posición de un eje que tiene fines de carrera.

Las instrucciones MRP en modo relativo y absoluto tienen el mismo efecto cuando el eje no se está moviendo. Sin embargo, cuando el eje se está moviendo, el modo absoluto presenta un error de posición igual al movimiento del eje durante el tiempo necesario para ejecutar la instrucción MRP y asignar una nueva posición. El modo relativo no presenta este error y garantiza una corrección exacta independientemente de la velocidad o posición del eje.

Modo relativo

Cuando se introduce o selecciona Relative como el tipo de MRP, el valor de la nueva posición se usa para *compensar* la posición actual del eje. No tiene lugar ningún movimiento – la posición del eje actual (real o de comando) se redefine simplemente para ser la posición actual *más* la nueva posición especificada.

En modo relativo, la posición del eje se redefine de manera que no se presentan errores de posición si el eje se mueve. Es especialmente útil para la posición de desbobinado del eje bajo el control del programa en vez de usar la característica de eje giratorio incorporado.

Las instrucciones MRP en modo relativo y absoluto tienen el mismo efecto cuando el eje no se está moviendo. Sin embargo, cuando el eje se está moviendo, el modo absoluto presenta un error de posición igual al movimiento del eje durante el tiempo necesario para ejecutar la instrucción MRP y asignar una nueva posición. El modo relativo no presenta este error y garantiza una corrección exacta independientemente de la velocidad o posición del eje.

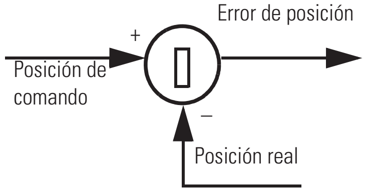
Posición real

Cuando se introduce o selecciona Actual como la selección de posición de MRP, la nueva posición se aplica directamente a la posición real del eje físico. La posición de comando del eje se ajusta también con la nueva posición real para conservar cualquier error de posición que exista. Esto garantiza que no habrá ningún movimiento inesperado del eje cuando las posiciones se redefinen. Consulte la especificación Objeto del eje de movimiento para obtener más información sobre la posición de comando, la posición real y error de posición.

Posición de comando

Cuando se introduce o selecciona Command como la selección de posición de MRP, la nueva posición se aplica directamente a la posición de comando del eje imaginario o servoeje. Debido a que los ejes de sólo retroalimentación no tienen una posición de comando, seleccione siempre Actual desde el menú de posiciones para los ejes sólo maestros. La posición real de servoejes se ajusta también con la nueva posición de comando para conservar cualquier error de posición que exista. Esto garantiza que no habrá ningún movimiento inesperado del eje cuando las posiciones se redefinen.

La posición de comando es la posición deseada o comandada de un servo tal y como fue generada por alguna instrucción de control de movimiento previa. La posición real es la posición actual de un eje físico o virtual, según la mida el encoder u otro dispositivo de retroalimentación. El error de posición es la diferencia entre ambos y se usa para activar el motor e igualar la posición real con la posición de comando. La figura a continuación muestra la relación de estas tres posiciones.



Relación de posición

Consulte la especificación Objeto del eje de movimiento para obtener una descripción general más detallada del lazo del servo digital anidado utilizado por los controladores de movimiento ControlLogix.

Para ejecutar correctamente una instrucción MRP, el eje específico debe estar configurado como un servoeje o de sólo retroalimentación. De lo contrario, la instrucción dará error.

IMPORTANTE La ejecución de la instrucción MRP puede necesitar la ejecución de múltiples escaneos porque requiere la transmisión de múltiples mensajes al módulo de movimiento. Por lo tanto, el bit Listo (.DN) no se establecerá inmediatamente, sino sólo cuando estos mensajes hayan sido correctamente transmitidos.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Consulte Códigos de error (ERR) para las instrucciones de control de movimiento en página 395.

Códigos de error extendidos: Los códigos de error extendidos proporcionan información adicional específica de instrucción para los códigos de error que son genéricos a muchas instrucciones. Los siguientes códigos de error extendidos ayudan a localizar el problema cuando la instrucción MRP recibe un mensaje de error Fallo de mensajes al servomódulo (12).

Código de error asociado (decimal)	Código de error extendido (decimal)	Significado
SERVO_MESSAGE_FAILURE (12)	Dispositivo en estado incorrecto (16)	La Redefinición de la posición, inicio y registro 2 son excluyentes entre sí. (SERCOS).

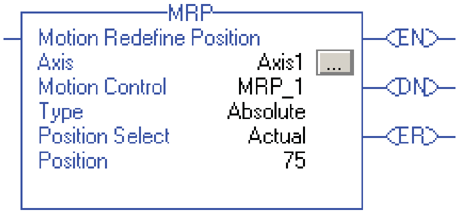
Los códigos de error extendidos para el código de error Parámetro fuera de rango (13) funcionan de manera un poco diferente. En lugar de tener una enumeración estándar, el número que aparece para el código de error extendido hace referencia al número de operando tal y como aparecen en la plantilla de arriba hacia abajo, con el primer operando como cero. Por lo tanto, para la instrucción MRP, un código de error extendido de 4 haría referencia al valor del operando Position. Entonces, debería verificar su valor con el rango de valores aceptado para la instrucción.

MRP cambia a bits de estado: Si el eje se había devuelto a su posición inicial antes de ejecutar la instrucción MRP, se establece el bit HomedStatus. El bit HomedStatus se borra cuando se ejecuta la instrucción MRP. Esto indica que la posición del eje ya no está referenciado a la posición inicial.

Si el eje ha vuelto a la posición inicial mediante el procedimiento de inicio absoluto, se establece AbsoluteReferenceStatus. Es posible que también se establezca HomedStatus si el eje no ha estado sujeto a un ciclo de potencia. AbsoluteReferenceStatus deberá borrarse al ejecutar la MRP. Esto indica que la posición del eje ya no está referenciado a la posición inicial absoluta.

Ejemplo: Cuando las condiciones de entrada son verdaderas, el controlador cambia la posición de *eje1*.

Lógica de escalera de relés



MRP
Motion Redefine Position
Axis Axis1
Motion Control MRP_1
Type Absolute
Position Select Actual
Position 75

Ejemplo de lógica de escalera de MRP

Texto estructurado

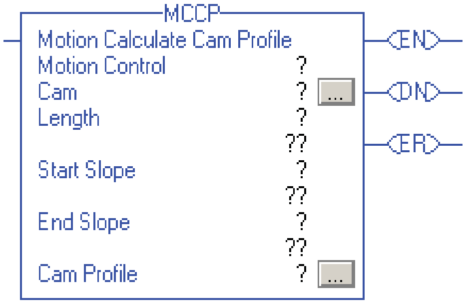
MRP(Axis1,MRP_1,Absolute,Actual,75);

Motion Calculate Cam Profile (MCCP)

La instrucción MCCP calcula un perfil de leva en una matriz de puntos de leva. Una matriz de puntos de leva se puede establecer programáticamente o mediante el uso del editor de perfil de levas RSLogix 5000. Cada punto de leva de la matriz de levas consiste en un valor de posición de esclavo, un valor de posición de maestro (leva de posición) o tiempo (leva de tiempo) y un tipo interpolación (lineal o cúbico). El perfil de leva resultante puede ser usado por una instrucción MAPC o MATC para regir el movimiento de un eje esclavo según el tiempo o la posición de maestro.

ATENCIÓN ⚠ Utilice un tag de control de movimiento sólo una vez. No vuelva a utilizarlo en otra instrucción. De lo contrario, podría ocasionar un movimiento inesperado del equipo y provocar lesiones a personas.

Operandos: *Lógica de escalera de relés*



Operando	Tipo	Formato	Descripción
Motion Control	MOTION_ INSTRUCTION	tag	Estructura que se utiliza para acceder a los parámetros de estado de bloqueo.
Cam	CAM	Matriz	Nombre del tag de la matriz de levas utilizado para calcular el perfil de levas. El índice de matriz numérico indica el elemento de leva de inicio en la matriz que se usa en el cálculo de perfil de levas. La elipsis ejecuta el editor de perfil de levas.
Length	UINT	Inmediato o tag	Determina el número de elementos de leva en la matriz utilizados en el cálculo de perfil de levas.
Start Slope	REAL	Inmediato o tag	Ésta es la condición límite para la pendiente inicial del perfil. Es sólo válida para un primer segmento cúbico y se usa para especificar una pendiente mediante el primer punto.
End Slope	REAL	Inmediato o tag	Ésta es la condición límite para la pendiente final del perfil. Es sólo válida para un último segmento cúbico y se usa para especificar una pendiente mediante el último punto.
Cam Profile	CAM_PROFILE	matriz	Nombre del tag de la matriz del perfil de leva calculado que se usa como entrada para las instrucciones MAPC y MATC. Sólo el elemento de matriz cero ([0]) está permitido para la matriz de perfil de leva. La elipsis ejecuta el editor de perfil de levas.

Texto estructurado

```
MCCP(MotionControl,Cam,
Length,StartSlope,EndSlope,
CamProfile);
```

Los operandos son iguales a los de la instrucción MCCP de lógica de escalera de relés. En el caso de los operandos de matriz, no es necesario incluir el índice de matriz. Si no incluye el índice, la instrucción comienza con el primer elemento de la matriz ([0]).

Estructura MOTION_INSTRUCTION

Mnemónico	Descripción
Bit .EN (Habilitado) 31	El bit Habilitado se establece durante la transición del renglón de falso a verdadero y queda establecido hasta que se establece el bit Listo y el renglón pasa a falso.
Bit .DN (Listo) 29	El bit Listo se establece cuando la instrucción de cálculo de levas se ha ejecutado correctamente y la matriz de perfil de levas se ha calculado.
Bit .ER (Error) 28	El bit Error indica cuando la instrucción detecta un error, como si la matriz de levas tuviera una longitud no válida.

Descripción: La instrucción MCCP calcula un perfil de levas basado en un determinado conjunto de puntos en una matriz de levas especificada. Los perfiles de levas resultantes generados por esta instrucción se pueden usar por subsiguientes instrucciones de operaciones de levas MAPC o MATC para proporcionar movimientos complejos de un eje esclavo con respecto a la posición del eje maestro o con respecto al tiempo.

Debido a que los perfiles de levas pueden ser calculados directamente por el editor de perfil de levas RSLogix 5000, el principal propósito de la instrucción MCCP es proporcionar un método para calcular los perfiles de levas en tiempo real en base a cambios programáticos en las correspondientes matrices de levas.

Cómo especificar una matriz de levas

Para ejecutar una instrucción MCCP, se debe crear un tag de matriz de levas utilizando Tag Editor o el editor de perfil de levas RSLogix. La figura a continuación ilustra cómo los tags de matriz de levas se establecen y se usan como entrada para la instrucción MCCP.

Los elementos de matriz de levas consisten en pares de un punto esclavo (yp) y maestro (xp), como así también un tipo de interpolación. Debido a que no hay ninguna asociación con un tiempo o posición de eje específico, los valores del punto x e y no tienen unidades. El tipo de interpolación se puede especificar para cada punto como “lineal” o “cúbico”.

Cómo especificar el tag de perfil de levas

Para ejecutar una instrucción MAPC, se debe crear también un tag de matriz de perfil de levas. Los tags de matriz de perfil de levas se pueden crear mediante Tag Editor RSLogix 5000 o las instrucciones MAPC/MATC usando el editor de perfil de levas incorporado.

Es posible modificar los datos dentro de la matriz de perfil de levas en tiempo de compilación usando el editor de perfil de levas o en tiempo de ejecución con la instrucción MCCP. En el caso de cambios en tiempo de ejecución, se debe crear una matriz de levas para usar la instrucción MCCP.

El parámetro Status se usa para indicar que se ha calculado el elemento de matriz de perfil de levas. Si se intenta ejecutar una instrucción de operación de levas usando elementos no calculados en un perfil de levas, las instrucciones MAPC o MATC darán error. El parámetro Type determina el tipo de interpolación aplicado entre este elemento de matriz de levas y el siguiente elemento de levas.

Miembro estado de matriz de perfil de levas

El miembro estado del primer elemento de la matriz de perfil de levas es especial y se utiliza para controlar la integridad de datos. Por esta razón, la MCCP debe especificar siempre el perfil de levas con el índice de inicio establecido en 0. Este primer miembro estado del elemento del perfil de levas puede tener los siguientes valores:

Variables de estado	Descripción
0	El elemento de perfil de levas no ha sido calculado
1	El elemento de perfil de levas se está calculado
2	El elemento de perfil de levas ha sido calculado
n	El elemento de perfil de levas se ha calculado y se está usando actualmente mediante las instrucciones MAPC (*n*-2) o MATC

Interpolación spline cúbica y lineal

Los perfiles de levas calculados resultantes se interpolan completamente. Esto significa que si la posición o tiempo del eje maestro actual no corresponde exactamente con un punto en la matriz de levas usada para generar el perfil de levas, la posición del eje esclavo está determinada por interpolación lineal o cúbica entre puntos adyacentes. De esta manera, se proporciona el movimiento esclavo más suave posible. La instrucción MCCP lleva esto a cabo calculando coeficientes en una ecuación polinominal que determina una posición de esclavo en función de la posición de maestro o tiempo.

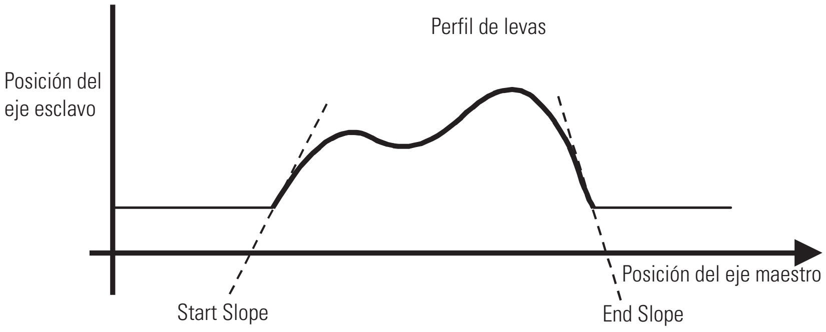
Cómo calcular el perfil de levas

Antes de calcular un perfil de levas en un eje especificado, las instrucciones MCCP primero comprueban si la matriz del perfil de levas ha sido calculada controlando el valor del miembro Status del primer elemento de perfil de levas. Si el valor Status es 0 o 2, el MCCP procede con el cálculo del perfil de levas. Cuando la matriz del perfil de levas ha sido completamente calculada, la instrucción MCCP establece el valor Estado del primer elemento de perfil de levas en “se está calculando”, o 1, y luego establece el valor Status de todos los elementos de perfil de levas en “se está calculando”. A medida que avanza el cálculo, los valores Status de los miembros del perfil de levas individuales se establecen en “calculados” o 2. Cuando todos los elementos de la matriz del perfil de levas han sido calculados, el valor Status del primer elemento de perfil de levas se establece también en “calculado”.

Sin embargo, si una instrucción MCCP se ejecuta con 1 como valor Status de perfil de levas inicial, éste está siendo calculado por otra instrucción MCCP y la instrucción MCCP dará error. Si el valor Status es >2, un proceso de instrucción MAPC o MATC está usando activamente el perfil de levas, y la instrucción MCCP dará error.

Pendiente inicial y pendiente final

Para facilitar una entrada y salida suaves de un perfil de levas cúbico, se proporciona un control de pendiente. Los parámetros Start Slope y End Slope determinan la velocidad inicial de cambio del esclavo relativo al maestro. Estos valores se utilizan en los cálculos spline cúbicos realizados en la matriz de levas. El diagrama a continuación muestra la relación de pendiente esclavo-maestro.



Pendiente inicial y final

Los valores predeterminados para Start Slope y End Slope son 0 para facilitar un inicio y final suaves al perfil de levas desde el reposo. Sin embargo, si el eje está ya en una operación de levas, se puede especificar una pendiente inicial apropiada que no sea cero para que coincida con la pendiente final de la leva en ejecución actualmente y mezclar a la perfección los perfiles de ambas levas.

Los valores de pendiente inicial y pendiente final no son aplicables al iniciar o finalizar el perfil de levas con interpolación lineal.

IMPORTANTE La ejecución de la instrucción MCCP se completa en un único escán. Por lo tanto, esta instrucción deberá colocarse en una tarea separada para evitar impactar en el tiempo de escaneo del programa de usuario.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

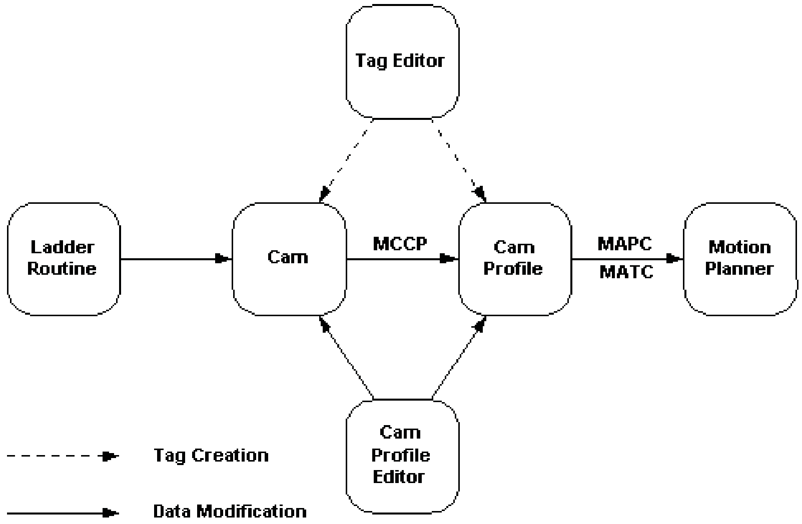


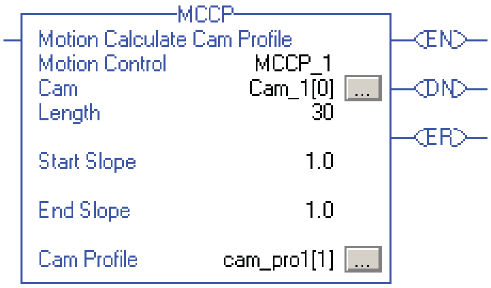
Diagrama de operación de levas

Códigos de error: Consulte Códigos de error (ERR) para las instrucciones de control de movimiento en página 395.

Códigos de error extendidos: Los códigos de error extendidos proporcionan información adicional específica de instrucción para los códigos de error que no son suficientemente específicos para ayudar a localizar el problema. Cuando la instrucción MCCP recibe un mensaje de error Longitud de leva no válida (26) indicando que el parámetro de entrada de longitud no se corresponde con lo esperado por la instrucción, el código de error extendido correspondiente proporciona el número de levas del tag de leva proporcionado para la instrucción. Cuando la instrucción MCCP recibe un mensaje de error Longitud de perfil leva no válida (27) indicando que el parámetro de entrada de longitud no se corresponde con lo esperado por la instrucción, el código de error extendido correspondiente proporciona el número de puntos de leva que la instrucción está intentando generar.

MCCP cambia a bits de estado: ninguno

Ejemplo: *Lógica de escalera de relés*



Ejemplo de lógica de escalera de MCCP

Texto estructurado

```
MCCP(MCCP_1,Cam_1[0],30,1.0,1.0,cam_pro1[1]);
```

Motion Axis Position Cam (MAPC)

La instrucción MAPC proporciona operaciones electrónicas de levas entre dos ejes según el perfil de levas especificado.

Al ejecutarla, el eje esclavo especificado se sincroniza con el eje maestro designado usando un perfil de levas de posición establecido por el editor de perfil de levas RSLogix 5000, o mediante una instrucción MCCP ejecutada previamente. La dirección del movimiento del eje esclavo que corresponde al eje maestro está definida por un parámetro flexible de entrada de dirección. La dirección de la operación de levas, ya que se aplica al esclavo, se puede establecer de manera explícita como Same o Opposite, o establecer la correspondiente a la dirección actual de la operación de levas como Reverse o Unchanged.

Para sincronizar de manera precisa la posición del eje esclavo con la posición del eje maestro, es posible especificar un ajuste del parámetro Execution Schedule y una posición asociada de bloqueo maestro para el eje maestro. Cuando el eje maestro pasa más allá de la posición de bloqueo maestro en la dirección especificada en el parámetro Execution Schedule, el eje esclavo se bloquea en la posición del eje maestro según el perfil de levas especificado que comienza en la posición de bloqueo de levas.

El perfil de levas se puede configurar también para la ejecución inmediata o pendiente de finalización de un perfil de levas de posición actualmente en ejecución mediante el parámetro Execution Schedule. El perfil de levas se puede ejecutar:

- una vez
- continua
- de manera persistente

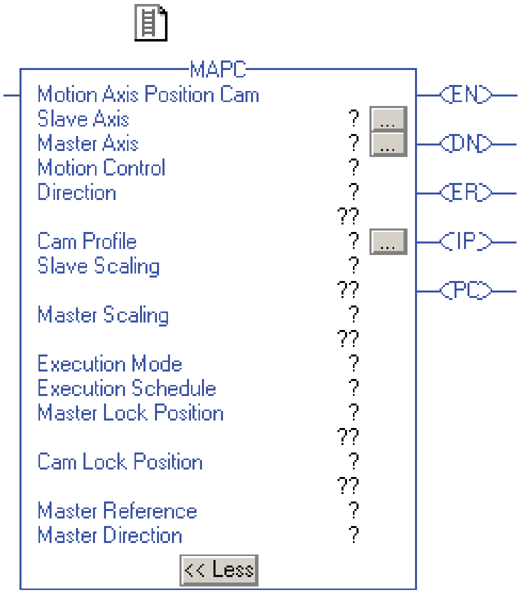
especificando el modo de ejecución deseado.

La selección de referencia de maestro permite la entrada de operaciones de levas desde el maestro para ser derivadas desde la posición Real o de Comando del eje maestro. Para ser compatible con aplicaciones que necesiten movimientos unidireccionales, se encuentra disponible la característica “deslizar embrague”, la cual evita que el esclavo “respalde datos” cuando el eje maestro revierta la dirección. Esta característica está controlada por el parámetro Master Direction.

La funcionalidad de Master Scaling y Slave Scaling se puede utilizar para escalar el movimiento esclavo en base a un perfil de levas estándar sin necesidad de crear una nueva tabla de levas y calcular un nuevo perfil de levas.

ATENCIÓN Utilice un tag de control de movimiento sólo una vez. No vuelva a utilizarlo en otra instrucción. De lo contrario, podría ocasionar un movimiento inesperado del equipo y provocar lesiones a personas.

Operandos: *Lógica de escalera de relé*



MAPC
Motion Axis Position Cam
Slave Axis ?
Master Axis ?
Motion Control ?
Direction ?
??
Cam Profile ?
Slave Scaling ?
??
Master Scaling ?
??
Execution Mode ?
Execution Schedule ?
Master Lock Position ?
??
Cam Lock Position ?
??
Master Reference ?
Master Direction ?
<< Less

Operando	Tipo	Formato	Descripción
Eje esclavo	AXIS_VIRTUAL AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje al cual se aplica el perfil de levas. La elipsis ejecuta el diálogo Axis Properties.
Eje maestro	AXIS_FEEDBACK AXIS_CONSUMED AXIS_VIRTUAL AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	tag	Eje al cual sigue el eje esclavo según el perfil de levas. La elipsis ejecuta el diálogo Axis Properties. Si se selecciona Pending como Execution Schedule, se ignorará el eje maestro.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura que se utiliza para acceder a los parámetros de estado de bloqueo.

Operando	**Tipo**	**Formato**	**Descripción**
Direction	UINT32	Inmediato o tag	Dirección relativa del eje esclavo con respecto al eje maestro: • Same – los valores de posición del eje esclavo están en el mismo sentido que los del maestro. • Opposite – los valores de posición del eje esclavo están en el sentido opuesto a los del maestro. O relativos a la dirección actual o anterior de operación de levas: • Reverse – la dirección actual o anterior de la leva de posición se revierte durante la ejecución. Cuando se ejecuta por primera vez con Reverse seleccionado, el control cambia de manera predeterminada la dirección a Opposite. • Unchanged – esto permite cambiar otros parámetros de leva sin alterar la dirección actual o anterior de operación de levas. Cuando se ejecuta por primera vez con Unchanged seleccionado, el control cambia de manera predeterminada la dirección a Same.
Cam Profile	CAM_PROFILE	matriz	Nombre del tag de la matriz del perfil de leva calculado que se usa para establecer la relación de posición maestra/esclava. Sólo el elemento de matriz cero ([0]) está permitido para la matriz de perfil de levas. La elipsis ejecuta el editor de perfil de levas.
Slave Scaling	REAL	Inmediato o tag	Escala la distancia total recorrida por el eje esclavo mediante el perfil de levas.
Master Scaling	REAL	Inmediato o tag	Escala la distancia total recorrida por el eje maestro mediante el perfil de levas.

Operando	Tipo	Formato	Descripción
Execution Mode	UINT32	Inmediato	Determina si el perfil de levas se ejecuta sólo una vez o de manera repetida: 0 = Once – el movimiento de levas del eje esclavo comienza sólo cuando el eje maestro se mueve en el rango definido por los puntos de inicio y final del perfil de levas. Cuando el eje maestro se mueve más allá del rango definido, se detiene el movimiento de levas en el eje esclavo y se establece el bit Proceso Completo. El movimiento del esclavo no se recupera si el eje maestro regresa al rango del perfil de levas. 1 = Continuous – Una vez iniciado, el perfil de levas se ejecuta indefinidamente. Esta característica es útil en aplicaciones giratorias, en las cuales es necesario que la posición de levas marche continuamente de modo giratorio o recíproco. 2 = Persistent – Cuando el eje maestro se mueve más allá del rango definido, se detiene el movimiento de levas en el eje esclavo y se borra el bit PositionCamLockStatus. El movimiento del esclavo se recupera en la dirección opuesta cuando el eje maestro se mueve en retroceso y regresa al rango del perfil de levas. En ese momento, se establece el bit PositionCamLockStatus.

Operando	Tipo	Formato	Descripción
Execution Schedule	UINT32	Inmediato	Selecciona el método utilizado para ejecutar el perfil de levas. Las opciones son: 0 = Inmediate – El eje esclavo se bloquea inmediatamente en el eje maestro y comienza el proceso de operación de levas de posición. 1 = Pending – le permite obtener una nueva ejecución de levas de posición una vez que finaliza una leva de posición en proceso. Al seleccionar Pending, se ignoran los siguientes parámetros: Master Axis, Master Lock Position y Master Reference. 2 = Forward only – el perfil de levas comienza cuando la posición del maestro cruza la posición de bloqueo del maestro en dirección de avance. 3 = Reverse only – el perfil de levas comienza cuando la posición del maestro cruza la posición de bloqueo del maestro en dirección de retroceso. 4 = Bi-directional – el perfil de levas comienza cuando la posición del maestro cruza la posición de bloqueo del maestro en cualquiera de ambas direcciones.
Master Lock Position	REAL	Inmediato o tag	La posición absoluta del eje maestro, es en la cual el eje esclavo se bloquea en el eje maestro. Si se selecciona Pending como el valor para Execution Schedule, se ignorará el parámetro Master Lock Position.

Operando	**Tipo**	**Formato**	**Descripción**
Cam Lock Position	REAL	Inmediato o tag	Esto determina el lugar de inicio del perfil de levas.
Master Reference	UINT32	Inmediato	Establece la referencia de posición maestra a la posición de comando o la posición real. Si se selecciona Pending como el valor para Execution Schedule, se ignorará el parámetro Master Reference. 0 = Real – el movimiento del eje esclavo se genera desde la posición actual del eje maestro que mide su encoder u otro dispositivo de retroalimentación. 1 = Comando – el movimiento del eje maestro se genera desde la posición deseada o comandada del eje maestro.
Master Direction	UINT32	Inmediato	Esto determina la dirección del eje maestro que genera el movimiento del esclavo según el perfil de levas. Las opciones son: 0 = Bi-directional – el eje esclavo puede seguir el eje maestro en cualquiera de las dos direcciones. 1 = Forward only – el eje esclavo sigue el eje maestro en la dirección de avance del eje maestro. 2 = Reverse only – el eje esclavo sigue el eje maestro en la dirección de opuesta del eje maestro.

Texto estructurado

```
MAPC(SlaveAxis,MasterAxis,
MotionControl,Direction,
CamProfile,SlaveScaling,
MasterScaling,ExecutionMode,
ExecutionSchedule,
MasterLockPosition,
CamLockPosition,
MasterReference,
MasterDirection);
```

Los operandos son iguales a los de la instrucción MAPC de lógica de escalera de relés. En el caso de los operandos de matriz, no es necesario incluir el índice de matriz. Si no incluye el índice, la instrucción comienza con el primer elemento de la matriz ([0]).

En el caso de los operandos para los que usted debe seleccionar entre opciones disponibles, introduzca su selección como:

Este operando	Tiene estas opciones, las cuales usted...	
	introduce como texto	**o introduce como un número**
ExecutionMode	once	0
	continuous	1
	persistent	2
ExecutionSchedule	inmediate	0
	pending	1
	forwardonly	2
	reverseonly	3
	bidirectional	4
MasterReference	real	0
	comando	1
MasterDirection	bidirectional	0
	forwardonly	1
	reverseonly	2

Estructura MOTION_INSTRUCTION

Mnemónico	Descripción
Bit .EN (Habilitado) 31	Se establece cuando el renglón hace una transición de falso a verdadero y permanece establecido hasta que el renglón pasa a falso.
Bit .DN (Listo) 29	Se establece cuando la leva de posición del eje se ha iniciado correctamente.
Bit .ER (Error) 28	Se establece para indicar que la instrucción ha detectado un error, tal como especificar un eje no configurado.
Bit .IP (En Proceso) 26	Se establece en transición de renglón positiva y se borra si es reemplazada por otro comando MAPC, o interrumpida por un comando de paro, incorporación, cierre eléctrico o fallo del servo.
Bit .PC (Proceso Completo) 27	Se borra en la transición de renglón positiva y se establece en modo de ejecución "once" cuando la posición del eje maestro abandona el rango de posición del maestro definido por el perfil de levas activo actualmente.

Descripción: La instrucción MAPC ejecuta un perfil de levas configurado por una instrucción MCCP anterior o, de modo alternativo, por el editor de perfil de levas RSLogix 5000. Las levas de posición, en efecto, proporcionan la capacidad de implementar relaciones de "engranaje electrónico" no lineales entre dos ejes. No se usan límites máximos de velocidad, aceleración o desaceleración. La velocidad, aceleración y desaceleración del eje esclavo están completamente determinadas por el movimiento del eje maestro y el perfil designado de levas derivado de la tabla de levas asociada.

ATENCIÓN ⚠ La velocidad máxima, los límites de aceleración o desaceleración establecidos durante la configuración del eje no se aplican a operaciones electrónicas de levas.

Dirección de operación de levas

Las levas se pueden configurar para sumar o restar su contribución incremental a la posición de comando del eje esclavo. Este comportamiento se controla mediante el parámetro Direction.

Operación de levas en la misma dirección

Cuando se selecciona o introduce Same como la dirección para la instrucción MAPC, los valores de posición del eje esclavo calculados desde el perfil de levas se *suman* a la posición de comando del eje esclavo. Esta es la operación más frecuente, puesto que los valores de posición de perfil se usan tal y como son introducidos en la tabla de levas original. Es decir, los valores de perfil consecutivos en aumento dan como resultado un movimiento de eje en posición *positiva* y viceversa.

Operación de levas en la dirección opuesta

Cuando se selecciona o introduce Opposite como la dirección, los valores de posición del eje esclavo calculados desde el perfil de levas *se restan* a la posición de comando del eje esclavo. De esta manera, el movimiento del esclavo está en la dirección *opuesta* de aquella indicada en la tabla original. Es decir, los valores de perfil consecutivos en aumento dan como resultado un movimiento de eje en posición *negativa* y viceversa.

Cómo conservar la dirección actual de operación de levas

Cuando se selecciona o se introduce Unchanged como Direction, otros parámetros de levas de posición pueden cambiar al conservar la dirección actual o anterior de operación de levas (la misma u opuesta). Esto es útil cuando la dirección actual no es conocida o no es importante. Para ejecutar una leva por primera vez con Unchanged seleccionado, el control cambia de manera predeterminada la dirección a Same.

Cómo invertir la dirección actual de operación de levas

Cuando se selecciona Reverse, la dirección anterior o actual de la leva de posición cambia de Same a Opposite o de Opposite a Same. Para ejecutar una leva por primera vez con Reverse seleccionado, el control cambia de manera predeterminada la dirección a Opposite.

Cómo especificar el perfil de levas

Para ejecutar una instrucción MAPC, se debe especificar un tag calculado de matriz de datos de perfil de levas. Los tags de matriz de perfil de levas se pueden crear mediante Tag Editor RSLogix 5000 o la instrucción MAPC usando el editor de perfil de levas incorporado, o ejecutando una instrucción MCCP en una matriz de levas existente.

Es posible modificar los datos dentro de la matriz de perfil de levas en tiempo de compilación usando el editor de perfil de levas o en tiempo de ejecución con la instrucción MCCP. En el caso de cambios en tiempo de ejecución, se debe crear una matriz de levas para usar la instrucción MCCP. Consulte la especificación de la instrucción MCCP para obtener más detalles sobre cómo convertir matrices de levas.

Todos los elementos de estructura de la matriz de perfil de levas, excepto el elemento de estado, están “ocultos” del Tag Editor RSLogix 5000. Estos elementos no tienen valor para el usuario. El miembro Status se usa para indicar que se ha calculado el elemento de matriz de perfil de levas correspondiente. Si se intenta ejecutar una instrucción de operación de levas con elementos no calculados en un perfil de levas, la instrucción dará error. El parámetro Type determina el tipo de interpolación aplicado entre este elemento de matriz de levas y el siguiente elemento de levas, (por ejemplo, lineal o cúbico).

Controles de la matriz del perfil de levas

El miembro Status del primer elemento de la matriz de perfil de levas es especial y se utiliza para controlar la integridad de datos. Por esta razón, la MAPC debe especificar siempre el perfil de levas con el índice de inicio establecido en 0.

Este primer miembro Status del elemento del perfil de levas puede tener los valores siguientes.

Valor del estado	Descripción
0	El elemento de perfil de levas no ha sido calculado
1	El elemento de perfil de levas se está calculado
2	El elemento de perfil de levas ha sido calculado
n	El elemento de perfil de levas se ha calculado y se está usando actualmente mediante las instrucciones MAPC (*n*-2) o MATC

Antes de iniciar una leva en un eje especificado, las instrucciones MAPC comprueban si la matriz del perfil de levas ha sido calculada controlando el valor del miembro Status del primer elemento de perfil de levas. Si el estado es 0 o 1, el perfil de levas no ha sido calculado aún y la instrucción MAPC dará error. Si la matriz del perfil de levas ha sido completamente calculada (Status > 1), la instrucción aumenta el miembro Status indicando que este eje lo está usando.

Cuando la leva finaliza, o interrumpe, el miembro Status del primer elemento de la matriz del perfil de levas disminuye para mantener un seguimiento del número de levas que están usando activamente el perfil de levas asociado.

Interpolación cúbica y lineal

Las levas de posición están totalmente interpoladas. Esto significa que si la posición del eje maestro actual no corresponde exactamente con un punto en la tabla de levas asociada con el perfil de levas, la posición del eje esclavo está determinada por interpolación lineal o cúbica entre puntos adyacentes. De esta manera, se proporciona el movimiento esclavo más suave posible.

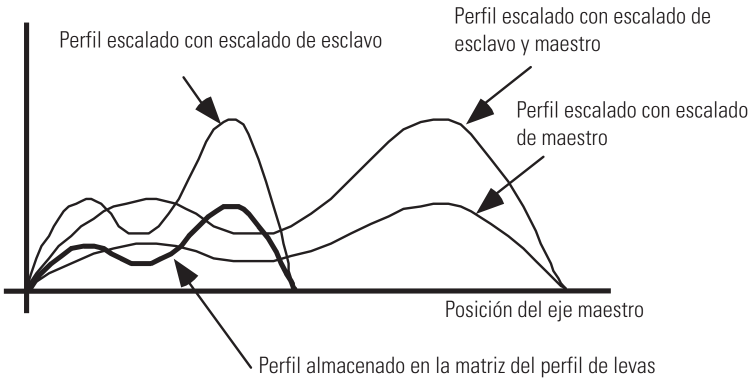
Cada punto de la matriz de levas utilizada para generar el perfil de levas se puede configurar para la interpolación cúbica o lineal.

La operación electrónica de levas permanece activa durante cualquier ejecución subsiguiente de procesos de impulso o movimiento para el eje esclavo. Esto permite que los movimientos de la operación electrónica de levas sean reemplazados con perfiles de impulso o movimiento para crear movimientos complejos o sincronización.

Cómo escalar levas de posición

Un perfil de levas de posición se puede escalar tanto en dimensión de maestro y dimensión de esclavo durante su ejecución. Esta característica de escalado es útil para permitir al perfil de levas almacenado ser usado para determinar la *forma* general del perfil de movimiento. Los parámetros de escalado se utilizan para definir el recorrido total del maestro o esclavo sobre el cual se ejecuta el perfil, como se muestra en la figura a continuación. De esta manera, un perfil de levas estándar se puede usar para generar una familia completa de perfiles de leva específicos.

Cuando una matriz de perfil de levas es especificada por una instrucción MAPC, los valores de esclavo y maestro definidos por la matriz del perfil de levas toman las unidades de posición de los ejes esclavo y maestro respectivamente. En contraste, los parámetros Master Scaling y Slave Scaling son valores sin unidades que se usan simplemente como multiplicadores del perfil de levas.



Matriz del perfil de levas

De manera predeterminada, tanto los parámetros Master Scaling y Slave Scaling se establecen en 1. Para escalar un perfil de leva de posición, introduzca un valor para Master Scaling o Slave Scaling que no sea 1.

Tenga en cuenta que, al aumentar el valor de Master Scaling de un perfil de levas, las velocidades y aceleraciones del perfil *disminuyen*, mientras que, al aumentar el valor de Slave Scaling, las velocidades y aceleraciones del perfil *aumentan*. Para mantener las velocidades y aceleraciones del perfil escalado aproximadamente igual a aquellas del perfil no escalado, los valores de Master Scaling y Slave Scaling deberán ser iguales. Por ejemplo, si el valor del escalado esclavo de un perfil es 2, el valor de Master Scaling deberá ser también 2 para mantener las velocidades y aceleraciones aproximadamente igual durante la ejecución de la leva de posición escalada.

ATENCIÓN ⚠

Si disminuye el valor de Master Scaling o aumenta el valor de Slave Scaling de una leva de posición, aumentan las velocidades y aceleraciones necesarias del perfil. Esto puede provocar un fallo de movimiento si se exceden las capacidades del sistema del variador.

Modos de ejecución del perfil de levas

Es posible seleccionar Once o Continuous como modos de ejecución para determinar cómo el movimiento de las leva se comportará cuando la posición de maestro se mueva más allá de los puntos de inicio y final del perfil definido en la tabla de levas original.

Si está seleccionado Once (predeterminado), el movimiento de la leva del eje esclavo comienza sólo cuando el eje maestro se mueve dentro de un rango definido por los puntos de inicio y final del perfil de levas. Cuando el eje maestro se mueve fuera del rango del perfil, el movimiento de levas del eje esclavo se detiene y se establece el bit Proceso Completo de la instrucción MAPC. Tenga en cuenta que, al contrario de la práctica de la Clase C actual, el movimiento del esclavo

no se recupera si el eje maestro regresa al rango del perfil especificado por los puntos de inicio y final.

Cuando se selecciona modo Continuous, una vez iniciado el perfil de levas especificado, se ejecuta indefinidamente. Con un funcionamiento continuo, las posiciones de esclavo y maestro del perfil se "desbobinan" cuando la posición del eje maestro se mueve fuera del rango del perfil, lo cual provoca la repetición del perfil de levas. Esta característica es particularmente útil en aplicaciones giratorias, en las cuales es necesario que la leva de posición marche continuamente de modo giratorio o recíproco. Para generar movimientos continuos suaves usando esta técnica, no obstante, se debe tener cuidado al designar los puntos de leva de la tabla de levas para asegurarse de que no hay discontinuidades de posición, velocidad o aceleración entre los puntos de inicio y final del perfil de levas calculado.

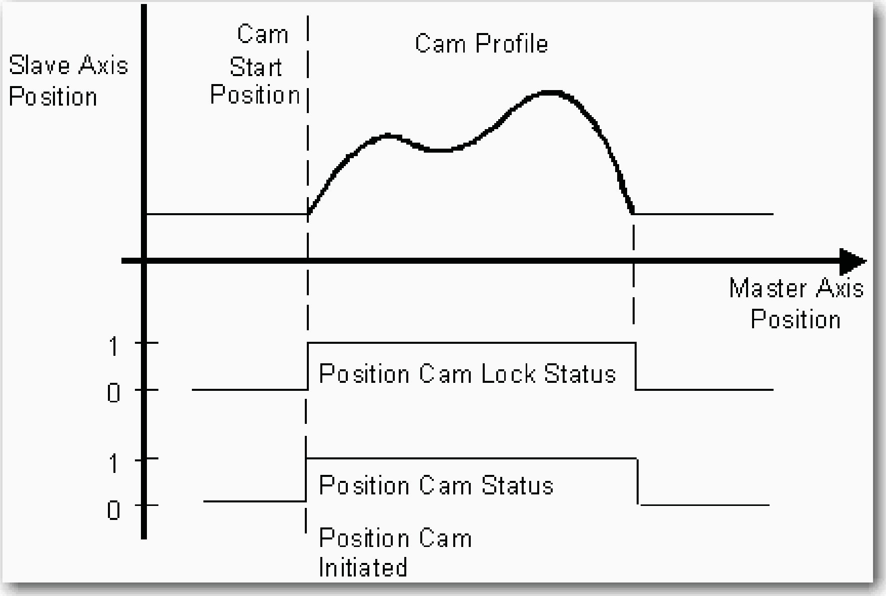
Execution Schedule

La ejecución de la instrucción MAPC se controla mediante el parámetro Execution Schedule.

Immediate Execution

Por defecto, la instrucción MAPC está programada para ejecutarse inmediatamente. En este caso, no hay retardo en la habilitación del proceso de operación de levas de posición y el parámetro Master Lock Position es irrelevante. El eje esclavo se bloquea inmediatamente en el eje maestro comenzando en la posición de bloqueo de levas del perfil de levas específico.

Como se ilustra en el diagrama a continuación, cuando se ejecuta la instrucción MAPC, el proceso de operación de levas se inicia en el eje esclavo especificado y se establece el bit Position Cam Status en la palabra Motion Status del eje esclavo. Si el parámetro Execution Schedule se establece en Inmediate, el eje esclavo se bloquea inmediatamente en el eje maestro según el perfil de levas especificado. Esto está indicado por el hecho que el bit Position Cam Lock Status se establece también para el eje esclavo especificado.

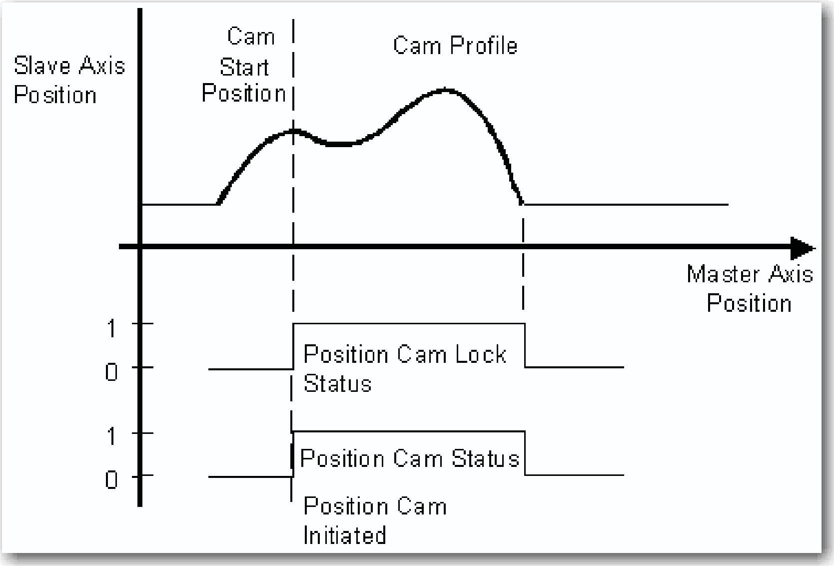


Ejecución inmediata

Cómo cambiar el parámetro Cam Lock Position

El parámetro Cam Lock Position de la instrucción MAPC determina el lugar de inicio dentro del perfil de levas cuando el esclavo se bloquea en el maestro. Generalmente, el parámetro Cam Lock Position se establece en el comienzo del perfil de levas como se muestra en la ilustración anterior. Puesto que el punto de inicio en la mayoría de las tablas de levas es 0, el parámetro Cam Lock Position se establece generalmente en 0. De forma alternativa, Cam Lock Position se puede establecer en cualquier posición dentro del rango de maestro del perfil de levas. Si se especifica que Cam Lock Position está fuera de este rango, la instrucción MAPC dará error.

El diagrama a continuación muestra el efecto de especificar un valor de Cam Lock Position que no sea el punto de inicio de la tabla de levas, en este caso, una posición dentro del perfil de levas mismo. Se debe tener la precaución de no definir un punto de inicio de levas que dé como resultado una discontinuidad de velocidad o aceleración en el eje esclavo si el eje maestro se está moviendo.

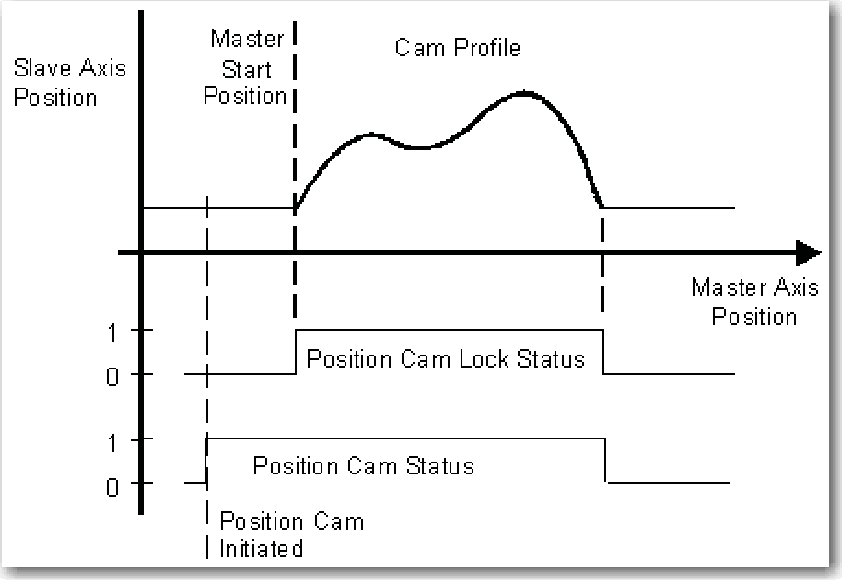


Cómo cambiar el parámetro Cam Lock Position

Ejecución Forward Only, Reverse Only o Bi-directional

En el caso en que el parámetro Execution Schedule de la instrucción se establece en Forward Only, Reverse Only o Bi-directional, el eje esclavo no se bloquea en el eje maestro si éste no cumple con la condición especificada. En este caso, el eje maestro está monitoreado por el proceso de operación de levas para determinar cuándo el eje maestro pasa la posición de bloqueo de maestro especificada en la dirección especificada. En una configuración de eje giratorio, este criterio de bloqueo es aún válido, independientemente del conteo de vueltas.

IMPORTANTE Si se redefine la referencia de posición del eje maestro (por ejemplo, una instrucción MRP) después de ejecutarse una instrucción MAPC, pero antes de que se cumpla la condición de bloqueo, el generador de perfil de levas monitorea el eje maestro según el sistema de referencia de posición absoluta en efecto antes de la operación de redefinición de la posición.



Ejecución Forward Only, Reverse Only o Bi-directional

Cuando la posición absoluta del eje maestro pasa la posición de bloqueo maestro en la dirección especificada (en la ilustración a continuación, dirección ‘Forward Only’), el bit Position Cam Status de la palabra Motion Status se establece para el eje esclavo específico. El movimiento del eje esclavo se inicia luego según el perfil de levas especificado comenzando en la posición de bloqueo de levas especificada en el perfil de levas. Desde este punto en adelante, sólo el *cambio incremental* en la posición del eje maestro se usa para determinar la posición del eje esclavo correspondiente desde el perfil de levas definido. Esto es importante para aplicaciones en las cuales el eje maestro es un eje giratorio, puesto que la leva de posición está desafectada del proceso de desbobinado de posición.

Cuando el eje maestro sale del rango definido por el perfil de levas (asumiendo que el modo de ejecución está configurado en Once), los bits Position Cam Lock Status y Position Cam Status de la palabra Motion Status se borran. Esta condición del bit Motion Status indica que el proceso de levas está completo. Este hecho se ve también reflejado en el comportamiento de segmento de la instrucción MAPC, se establece el bit PC y se borra el bit IP.

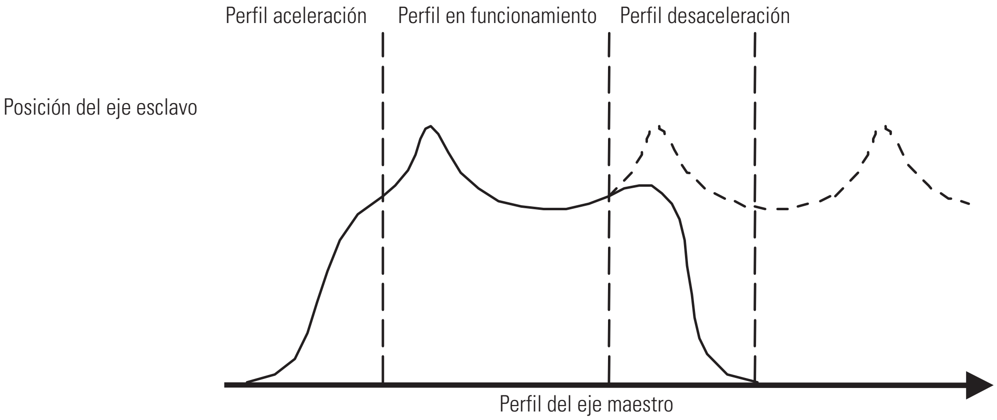
Una vez iniciado el movimiento de leva de posición cuando el eje maestro pasa la posición de bloqueo de maestro especificada en la dirección Forward Only o Reverse, el eje maestro puede cambiar la dirección y el eje esclavo retrocederá según corresponda.

Si una instrucción MAPC se ejecuta en un eje esclavo que está ya en operación de levas de posición de manera activa, se generará un error Cambio dinámico no válido (código de error 23). La única excepción a esto es cuando Execution Schedule está especificado como ‘pending’.

Ejecución de levas pendiente

Como alternativa, la ejecución de la instrucción MAPC puede dejarse con la finalización pendiente de una leva de posición actualmente en ejecución. Se puede seleccionar Pending para el parámetro Execution Schedule para mezclar a la perfección dos perfiles de leva de posición sin detener el movimiento.

La característica de ejecución pendiente es particularmente útil en aplicaciones como el envasado a alta velocidad, cuando un eje esclavo debe bloquearse en un eje maestro en movimiento y acelerar usando un perfil específico a la velocidad adecuada. Cuando este perfil de aceleración está listo, se debe mezclar suavemente con el perfil en funcionamiento, el cual generalmente se ejecuta de manera continua. Para detener el eje esclavo, el perfil en funcionamiento se mezcla suavemente con un perfil de desaceleración, de manera que el eje se detiene en un lugar conocido como se muestra a continuación.



Ejecución de levas pendiente

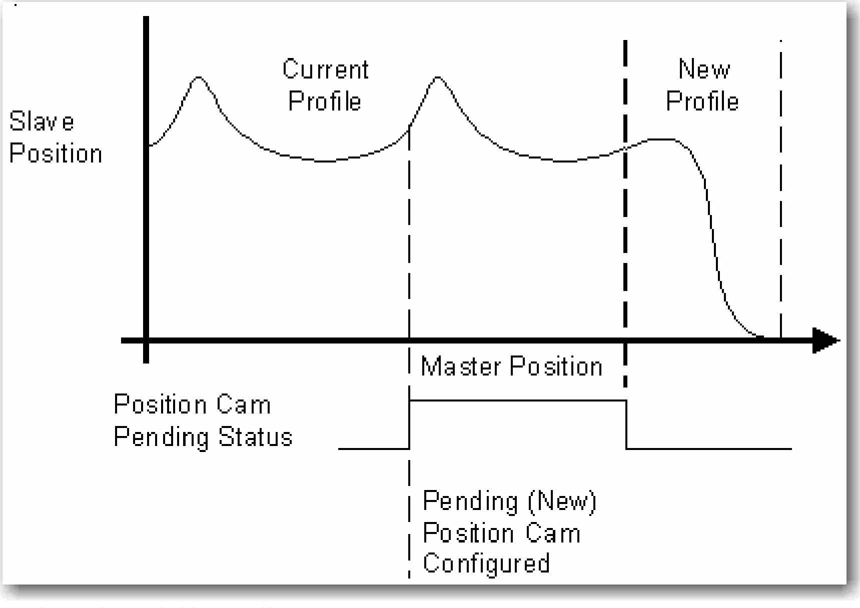
Ejecutando el perfil de levas de posición como un perfil de levas Pendiente mientras se ejecuta aún el perfil actual, los parámetros del perfil de levas adecuados se configuran con antelación. Esto hace la transición desde el perfil actual al perfil pendiente a la perfección; se mantiene la sincronización entre los ejes maestro y esclavo. Sin embargo, para garantizar un movimiento suave durante la transición, los perfiles deben estar diseñados de manera que no existan discontinuidades de posición, velocidad o aceleración entre el final del perfil actual y el inicio del nuevo. Esto se realiza usando el editor de perfil de levas RSLogix 5000.

Una vez ejecutada una instrucción pendiente de levas de posición, el nuevo perfil de levas hace efecto automáticamente (y se transforma en el perfil actual) cuando el eje maestro pasa por el punto de inicio o final del perfil actual. Si la leva actual está configurada para ejecutarse una vez, el nuevo perfil se inicia cuando termina de pasar por el perfil de levas actual y se establece el bit PC de la instrucción MAPC actualmente activa. Si la leva actual está configurada para ejecutarse de manera continua, el nuevo perfil se inicia cuando termina de pasar por el perfil de levas actual y se borra el bit IP de la instrucción MAPC actualmente activa. El controlador de movimiento mantiene un seguimiento de las posiciones del eje esclavo y el eje maestro en relación al primer perfil en el momento del cambio y usa esta información para mantener la sincronización entre los perfiles.

Si el parámetro Execution Schedule de una instrucción MAPC se establece en Inmediato y un perfil de levas de posición está actualmente en proceso, la instrucción MAPC dará error. Esto es verdadero incluso cuando el eje está esperando bloquearse en el eje maestro.

Si se selecciona un parámetro Execution Schedule como Pending sin un perfil de leva de posición correspondiente en progreso, la instrucción MAPC se ejecuta, pero no ocurrirá ningún movimiento de operación de levas si no se inicia otra instrucción MAPC con un parámetro Execution Schedule no pendiente. Esto permite precargar los perfiles de leva pendiente antes de ejecutar la leva inicial. Este método se aplica en casos en los cuales las levas inmediatas finalizarían antes de que la leva pendiente pueda cargarse de manera confiable.

Una vez configurada una leva de posición pendiente, el bit Position Cam Pending Status de la palabra Motion Status se establece en 1 (verdadero) para el eje esclavo especificado. Cuando el perfil pendiente (nuevo) se inicia y se transforma en el perfil actual, el bit Position Cam Pending Status se borra inmediatamente como se muestra a continuación.



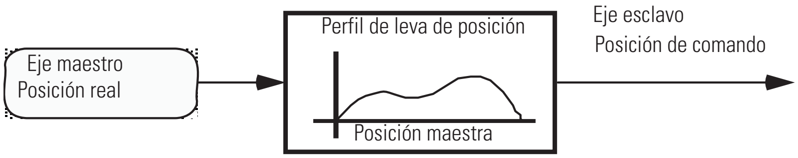
Leva de posición pendiente

Master reference

El parámetro Master Reference determina el origen de posición de maestro para vincular al generador de levas. Este origen puede ser una posición real o posición de comando del eje maestro. De la posición de comando se deriva un movimiento más suave, pero en algunos casos, por ejemplo, cuando un eje físico no está controlado por un módulo de movimiento ControlLogix, la posición real es la única opción práctica.

Cómo generar eje esclavo en la posición real

Cuando se introduce o se selecciona Actual Position como el origen de referencia maestra, el movimiento del eje maestro se genera desde la posición real del eje maestro como se muestra a continuación.

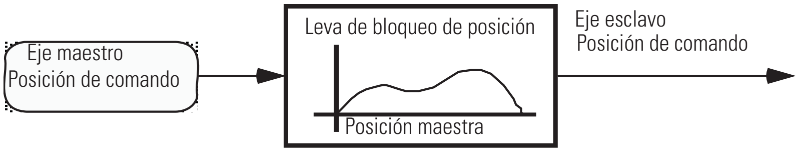


Cómo generar eje esclavo en la posición real

La posición real es la posición actual del eje maestro, según la mida su encoder u otro dispositivo de retroalimentación. Ésta es la selección predeterminada y la *única* selección cuando el tipo de eje maestro está configurado como sólo retroalimentación, puesto que frecuentemente es necesario sincronizar las posiciones reales de dos ejes.

Cómo generar eje esclavo en la posición de comando

Cuando se introduce o se selecciona la posición de comando como el origen de referencia maestra, el movimiento del eje maestro se genera desde la posición de comando del eje maestro como se muestra a continuación.



Cómo generar eje esclavo en la posición de comando

La posición de comando (sólo disponible cuando el tipo del eje maestro está configurado como Servoeje o Virtual) es la posición comandada o deseada del eje maestro.

Puesto que la posición de comando no incorpora ningún error de seguimiento asociado, perturbaciones de la posición externa, es una referencia más estable y precisa para las operaciones de levas. Al realizar operaciones de leva hacia la posición de comando del maestro, se debe *comandar* al eje maestro que se mueva para provocar algún movimiento en el eje esclavo. Consulte la especificación objeto del eje de movimiento para obtener más información sobre los parámetros de eje de la posición de comando y la posición real.

Master Direction

Normalmente, el parámetro Master Direction se establece en Bi-directional (predeterminado). Sin embargo, cuando se selecciona Forward Only para el parámetro Master Direction, el eje esclavo hace un seguimiento del eje maestro en la dirección de avance del eje maestro. Cuando se selecciona Reverse Only, el eje esclavo hace un seguimiento del eje maestro en la dirección de retroceso del eje maestro. Si el eje maestro cambia de dirección, el eje esclavo *no* invierte la dirección, pero permanece donde estaba cuando el maestro retrocedió. Esta característica unidireccional de las levas de posición se usa para proporcionar un embrague de deslizamiento electrónico, el cual evita que el generador de movimiento de levas se mueva hacia atrás en el perfil de levas si el maestro invierte su dirección.

Cuando el eje maestro retrocede otra vez, y recupera el movimiento en la dirección deseada, el eje esclavo “se recupera” otra vez cuando el maestro alcanza la posición, en la cual había retrocedido inicialmente. De esta manera, el eje esclavo mantiene la sincronización con el maestro mientras se inhiben movimientos en la dirección equivocada. Esto es especialmente útil, donde el movimiento en una cierta dirección puede provocar daños físicos a la máquina o al producto.

Cómo mover durante operaciones de levas

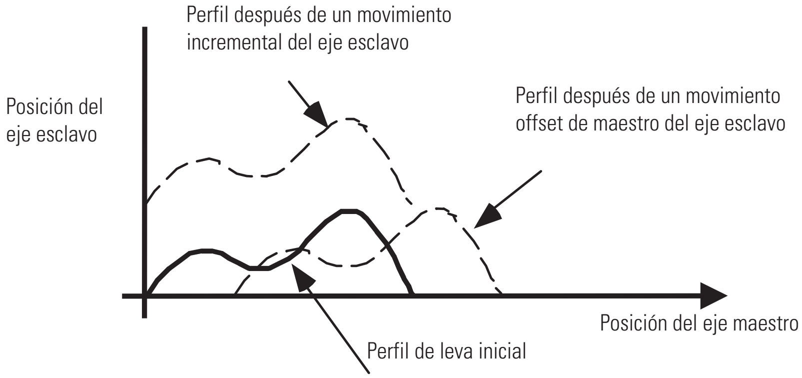
El eje de movimiento se puede mover durante una operación de levas para otorgar una fase sofisticada y control de offset mientras el eje esclavo está en funcionamiento.

Movimientos incrementales

Una instrucción MAM incremental se puede usar en el eje esclavo (o eje maestro si está configurado para funcionamiento Servo) mientras la leva de posición está en funcionamiento. Esto es especialmente útil para llevar a cabo el control de retardo/avance de fase. La distancia de movimiento incremental se puede utilizar para eliminar cualquier error de fase entre el maestro y el esclavo o para crear una relación exacta de fase.

Movimientos de offset maestro

Una instrucción MAM se puede usar también mientras la leva de posición está en funcionamiento para cambiar rápidamente la posición de referencia maestra de la leva. A diferencia de un movimiento incremental en el eje esclavo, un movimiento offset de maestro en el eje esclavo cambia el perfil de levas en relación al eje maestro, como se muestra a continuación.



Movimiento offset de maestro

Cuando la instrucción MAPC (excepto pendiente) se inicia, el movimiento offset de maestro activo correspondiente se inhabilita y el correspondiente offset de maestro, offset de estroboscopio y offset de maestro de inicio se restablecen en cero. Para obtener el cambio de posición de referencia maestra, la instrucción MAM debe iniciarse después de haber iniciado la MAPC.

Consulte la instrucción MAM para obtener más información sobre movimientos offset de maestro.

Cómo detener una leva

Al igual que otros generadores de movimiento (impulso, movimiento, engranaje), las levas activas deben detenerse con las diferentes instrucciones de paro, MAS o MGS. El movimiento de levas se debe detener también cuando el procesador ControlLogix cambia los modos OS. La instrucción MAS, en particular, debe estar habilitada para detener específicamente el proceso de operación de levas. Este proceso debe ser idéntico a la funcionalidad MAS que detiene específicamente un proceso de engranaje.

Cómo incorporar a partir de una leva

Al igual que otros generadores de movimiento (impulso, movimiento, engranaje), las levas activas deben cumplir con la funcionalidad de incorporar movimiento. Los movimientos y los impulsos, en particular, deben estar habilitados para incorporar a partir de operaciones de levas activas. Este proceso debe ser idéntico a la funcionalidad de incorporación aplicada a un proceso de engranaje.

Recuperación de fallos

A veces, es necesario reaccionar ante una condición de fallo de eje sin soltar la sincronización entre un maestro y el eje esclavo que está bloqueado en una relación de leva. Con una leva activa, existen un par de maneras de solucionar fallos de ejes.

Se puede crear un eje virtual y proporcionar una leva al mismo. Si es necesario, acople este eje maestro virtual al eje maestro real de la máquina. Establezca las diferentes acciones ante un fallo para todos los ejes en Status Only. Cuando ocurre un fallo en el eje (por ejemplo, un fallo de variador), un programa de aplicación que monitorea el estado de fallos de los ejes detecta el fallo y realiza un paro controlado de todos los ejes activos deteniendo el eje maestro. A nivel de perfil, todo está aún totalmente sincronizado. Utilice el error de seguimiento en el eje en fallo para determinar cuánto se alejó de la posición. Reestablezca el fallo en el eje en fallo, llévelo a la posición a velocidad controlada usando la instrucción MAM y el error de seguimiento calculado. Finalmente, comience a mover el eje maestro virtual.

La configuración es la misma que la anterior pero, en este caso, cuando el eje esclavo falla, la acción ante un fallo del eje inhabilita el variador. Esto, obviamente, interrumpiría el proceso de levas activo en el eje esclavo. En este punto, el programa de aplicación deberá detener todos los otros ejes mediante el eje maestro virtual. A continuación, reposicione el eje en fallo determinando dónde está el maestro y luego, recalculando dónde debería estar el eje esclavo si no hubiera ocurrido el fallo. Finalmente, haga una MAPC de bloqueo inmediato para resincronizar con el parámetro Cam Lock Position establecido en el valor calculado.

IMPORTANTE La ejecución de la instrucción MAPC se completa en un único escán, por lo tanto, el bit Listo (.DN) y el bit En Proceso (.IP) se establecen inmediatamente. El bit En Proceso (.IP) permanece establecido hasta que el proceso PCAM iniciado se completa, es reemplazado por otra instrucción MAPC, interrumpido por un comando de paro del eje de movimiento, operación de incorporación o acción ante un fallo del servo. El bit Proceso Completo se borra inmediatamente cuando MAPC se ejecuta y se establece cuando el proceso de levas se completa configurado en modo de ejecución 'Once'.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de borrado a establecido cada vez que deba ejecutarse la instrucción.

- En texto estructurado, acondicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

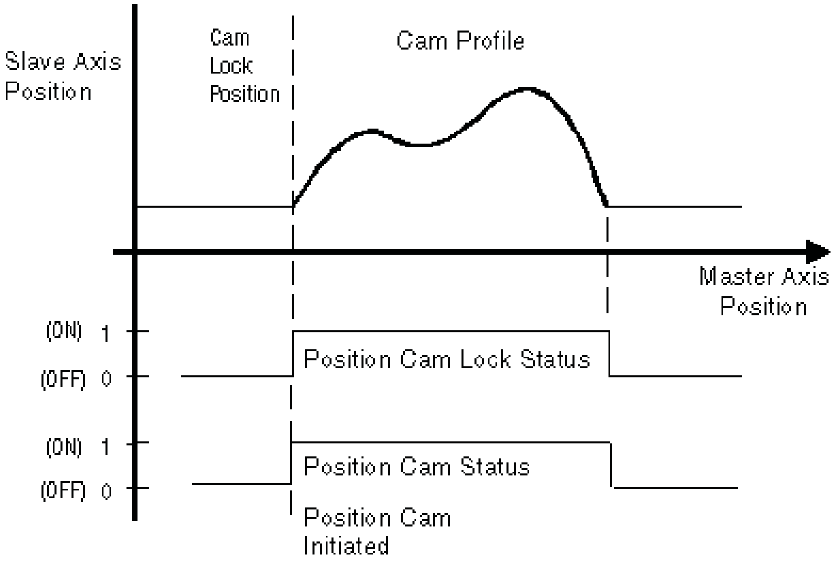


Diagrama de temporización de levas de posición

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Consulte Códigos de error (ERR) para las instrucciones de control de movimiento en página 395.

Códigos de error extendidos: Los códigos de error extendidos proporcionan información adicional específica de instrucción para los códigos de error que son genéricos a muchas instrucciones.

Los códigos de error extendidos para el código de error Eje no configurado (11) son los siguientes:

- Código de error extendido 1 significa que el eje esclavo no está configurado.
- Código de error extendido 2 significa que el eje maestro no está configurado.

Los códigos de error extendidos para el código de error Parámetro fuera de rango (13) incluyen un número que hace referencia al número de operando tal y como aparecen en la plantilla de arriba hacia abajo, con el primer operando como cero. Por lo tanto, para la instrucción MAPC, un código de error extendido de 5 haría referencia al valor del operando Scaling. Entonces, debería verificar su valor con el rango aceptado de valores para la instrucción.

Para el código de error 54 – El valor máximo de desaceleración es cero, si el error extendido devuelve un número positivo (0-*n*), está haciendo referencia al eje en error en el sistema de coordenadas. Diríjase a la ficha general Coordinate System Properties y mire debajo de la columna de corchetes ([ ]) de la cuadrícula del eje para determinar qué eje tiene un valor máximo de desaceleración de cero. Haga clic en el botón de elipsis al lado del eje en error para acceder a la pantalla de propiedades del eje. Diríjase a la ficha Dynamics y haga el cambio apropiado al valor máximo de desaceleración. Si el número del error extendido es –1, esto significa que el sistema de coordenadas tiene un valor máximo de desaceleración de 0. Diríjase a la ficha Coordinate System Properties Dynamics para corregir el valor máximo de desaceleración.

Bits de estado:

MAPC Cambia a bits de estado

Si Execution Schedule se establece en Immediate, la ejecución de la instrucción MAPC simplemente establece los bits Position Cam Status y Position Cam Lock Status en Verdadero.

Nombre del bit	Estado	Significado
Position Cam Status	VERDADERO	La operación de levas de posición está habilitada
Position Cam Lock Status	VERDADERO	El eje esclavo está bloqueado en el eje maestro según el perfil de levas.
Position Cam Lock Pending Status	FALSO	No hay leva de posición pendiente

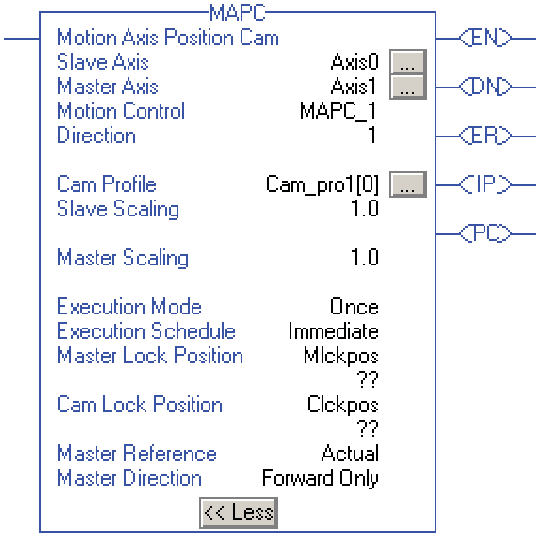
Si Execution Schedule se establece en Forward o Reverse, la ejecución de la instrucción MAPC inicialmente establece el bit Position Cam Status en Verdadero y el bit Position Cam Lock Status en Falso. Position Cam Lock Status pasa a Verdadero cuando se cumple la condición de Execution Schedule.

Nombre del bit	Estado	Significado
Position Cam Status	VERDADERO	La operación de levas de posición está habilitada
Position Cam Lock Status	FALSO	El eje esclavo espera que el eje maestro alcance la posición de bloqueo.
Position Cam Lock Pending Status	FALSO	No hay leva de posición pendiente

Si Execution Schedule se establece en Pending, la ejecución de la instrucción MAPC no afecta el estado actual de los bits Position Cam Status y Position Cam Lock Status. El bit Position Cam Pending Status se establece inmediatamente en Verdadero y pasa a Falso cuando la leva pendiente se transforma en leva activa.

Nombre del bit	Estado	Significado
Position Cam Status	N/A	La operación de levas de posición está habilitada
Position Cam Lock Status	N/A	El eje esclavo espera que el eje maestro alcance la posición de bloqueo.
Position Cam Lock Pending Status	VERDADERO	Pending Position Cam

Ejemplo: *Lógica de escalera de relés*


MAPC
Motion Axis Position Cam
Slave Axis Axis0
Master Axis Axis1
Motion Control MAPC_1
Direction 1
Cam Profile Cam_pro1[0]
Slave Scaling 1.0
Master Scaling 1.0
Execution Mode Once
Execution Schedule Immediate
Master Lock Position Mlckpos
??
Cam Lock Position Clckpos
??
Master Reference Actual
Master Direction Forward Only

Ejemplo de lógica de escalera de MAPC

Texto estructurado

MAPC(Axis0,Axis1,MAPC_1,1,Cam_pro1[0],1.0,1.0,Once, immediate,Mlckpos,Clckpos,Actual,Forwardonly);

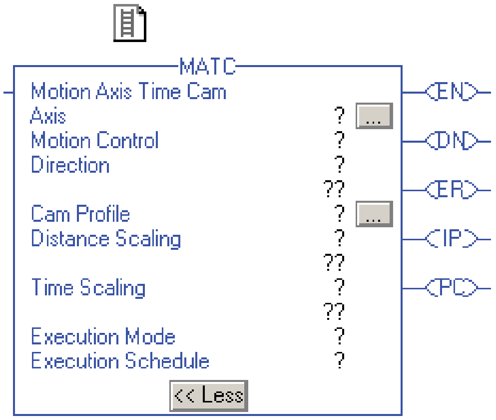
Motion Axis Time Cam (MATC)

La instrucción MATC proporciona operaciones electrónicas de levas de un eje en función del tiempo, según el perfil de levas especificado. Las levas de tiempo permiten la ejecución de perfiles de movimientos complejos, además de los perfiles de movimiento incorporados de curva en S o trapezoidal. Al ejecutarla, el eje esclavo especificado se sincroniza en tiempo usando un perfil de levas de tiempo establecido por el editor de perfil de levas RSLogix 5000, o mediante una instrucción MCCP ejecutada previamente. La dirección del movimiento del eje que corresponde al perfil de levas está definida por un parámetro muy flexible de entrada de dirección. La dirección de la operación de levas puede establecerse de manera explícita como Same o Opposite o establecerse en relación a la dirección de operación de levas actual como Reverse o Unchanged. El perfil de levas se puede configurar la ejecución Inmediata o Pendiente de finalización de un perfil de levas de tiempo actualmente en ejecución mediante el parámetro Execution Schedule. El perfil de levas se puede ejecutar también una vez o de manera continua especificando el modo de ejecución deseado. Las funciones Time Scaling y Distance Scaling se pueden utilizar para escalar el movimiento del eje en base a un perfil de levas estándar sin necesidad de crear una nueva tabla de levas y calcular un nuevo perfil de levas.

ATENCIÓN

Utilice un tag de control de movimiento sólo una vez. No vuelva a utilizarlo en otra instrucción. De lo contrario, podría ocasionar un movimiento inesperado del equipo y provocar lesiones a personas.

Operandos: *Lógica de escalera de relés*



Operando	Tipo	Formato	Descripción
AXIS	AXIS_FEEDBACK AXIS_VIRTUAL AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje al cual se aplica el perfil de levas. La elipsis ejecuta el diálogo Axis Properties.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura que se utiliza para acceder a los parámetros de estado de bloqueo.

Operando	Tipo	Formato	Descripción
Direction	UINT32	Inmediato o tag	Dirección relativa del eje esclavo con respecto al eje maestro: · Same – los valores de la posición del eje en el perfil de levas se suman a la posición de comando del eje. · Opposite – los valores de posición del eje en el perfil de levas se restan de la posición de comando del eje, lo cual crea el movimiento del eje en otra dirección de la indicada en la tabla de levas original. O relativos a la dirección actual o anterior de operación de levas: · Reverse – la dirección anterior o actual de la leva de posición cambia de Same a Opposite o viceversa. Cuando se ejecuta por primera vez con Reverse seleccionado, el control cambia de manera predeterminada la dirección a Opposite. · Unchanged – esto permite cambiar otros parámetros de leva sin alterar la dirección actual o anterior de operación de levas. Cuando se ejecuta por primera vez con Unchanged seleccionado, el control cambia de manera predeterminada la dirección a Same.
Cam Profile	CAM_PROFILE	matriz	Nombre del tag de la matriz de perfil de levas calculada. Sólo el elemento de matriz cero ([0]) está permitido para la matriz de perfil de levas. La elipsis ejecuta el editor de perfil de levas.
Distance Scaling	REAL	Inmediato o tag	Escala la distancia total recorrida por el eje mediante el perfil de levas.
Time Scaling	REAL	Inmediato o tag	Escala el intervalo de tiempo cubierto por el perfil de levas.
Execution Mode	UINT32	Inmediato	Determina cómo se comporta el movimiento de leva cuando el tiempo se mueve más allá del punto final del perfil de levas. Las opciones son: 0 = Once – Cuando el tiempo de ejecución de la leva de tiempo supera el rango de tiempo del perfil de levas, la instrucción MATC se completa, el movimiento del eje se detiene, y el bit Time Cam Status se borra. 1 = Continuous – El movimiento de perfil de levas se ejecuta indefinidamente.

Operando	Tipo	Formato	Descripción
Execution Schedule	UINT32	Inmediato	Selecciona el método utilizado para ejecutar el perfil de levas. Las opciones son: 0 = Immediate – la instrucción de programa para ejecutarse inmediatamente sin retardo al inhabilitar el proceso de operación de levas de tiempo. 1 = Pending – Retarda la ejecución de la leva de tiempo hasta que finaliza la leva de tiempo actualmente en ejecución o la siguiente inmediata. Esto es útil para mezclar un nuevo perfil de levas de tiempo con un proceso en progreso para obtener una transición perfecta.

Texto estructurado

```
MATC(Axis,MotionControl,
Direction,CamProfile,
DistanceScaling,TimeScaling,
ExecutionMode,
ExecutionSchedule);
```

Los operandos son iguales a los de la instrucción MATC de lógica de escalera de relés. En el caso de los operandos de matriz, no es necesario incluir el índice de matriz. Si no incluye el índice, la instrucción comienza con el primer elemento de la matriz ([0]).

En el caso de los operandos para los que usted debe seleccionar entre opciones disponibles, introduzca su selección como:

Este operando	Tiene estas opciones, las cuales usted…	
	introduce como texto	**o introduce como un número**
ExecutionMode	once	0
	continua	1
ExecutionSchedule	inmediato	0
	pending	1

Estructura MOTION_INSTRUCTION

Mnemónico	**Descripción**
Bit .EN (Habilitado) 31	El bit Habilitado se establece durante la transición del renglón de falso a verdadero y queda establecido hasta que el renglón pasa a falso.
Bit .DN (Listo) 29	El bit Listo se establece cuando la instrucción MATC se inicia correctamente.
Bit .ER (Error) 28	El bit Error indica cuando la instrucción detecta un error, como si el eje no estuviera configurado.
Bit .IP (En Proceso) 26	El bit En Proceso se establece en la transición de renglón positiva y se borra cuando es interrumpida por un comando de paro, incorporación, cierre eléctrico o fallo del servo.
Bit .PC (Proceso Completo) 27	El bit Proceso Completo se borra en la transición de renglón positiva y se establece en modo de ejecución "Once", cuando el tiempo sale del rango de tiempo definido por el perfil de levas activo actualmente.

Descripción:

La instrucción MATC ejecuta un perfil de levas de tiempo configurado por una instrucción MCCP anterior o, de modo alternativo, por el editor de perfil de levas RSLogix 5000. Las levas de tiempo otorgan la capacidad de implementar perfiles de movimientos complejos, además de los perfiles de movimiento incorporados de curva en S y trapezoidal. No se usan límites máximos de velocidad, aceleración o desaceleración en esta instrucción. La velocidad, aceleración y desaceleración del eje esclavo están totalmente determinadas por el perfil designado de levas derivado de la tabla de levas asociada.

ATENCIÓN

La velocidad máxima, los límites de aceleración o desaceleración establecidos durante la configuración del eje no se aplican a operaciones electrónicas de levas.

Dirección de operación de levas

Las levas se pueden configurar para sumar o restar su contribución incremental a la posición de comando del eje. Este comportamiento se controla mediante el parámetro Direction.

Operación de levas en la misma dirección

Cuando se selecciona o introduce Same como la dirección para la instrucción MATC, los valores de posición del eje calculados desde el perfil de levas se *suman* a la posición de comando del eje. Ésta es la operación más frecuente, puesto que los valores de posición de perfil se usan tal y como son introducidos en la tabla de levas original. Es decir, los valores de perfil consecutivos en aumento dan como resultado un movimiento de eje en posición *positiva* y viceversa.

Operación de levas en la dirección opuesta

Cuando se selecciona o introduce Opposite como la dirección, los valores de posición del eje calculados desde el perfil de levas *se restan* a la posición de comando del eje. De esta manera, el movimiento del esclavo está en la dirección *opuesta* de aquella indicada en la tabla original. Es decir, los valores de perfil consecutivos en aumento dan como resultado un movimiento de eje en posición *negativa* y viceversa.

Cómo cambiar el perfil de levas

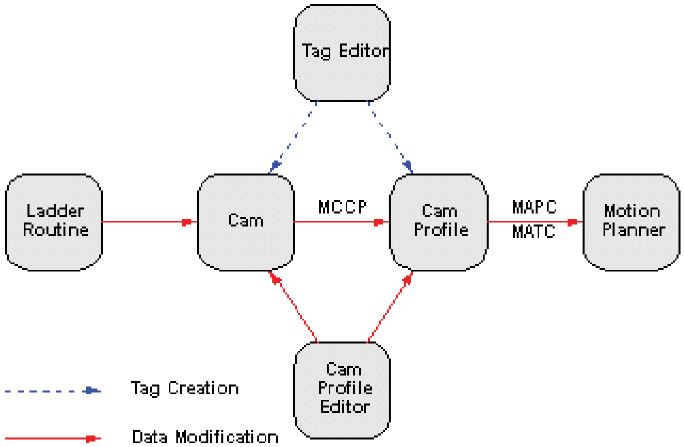
Cuando se selecciona o se introduce Unchanged como Direction, otros parámetros de levas de tiempo se pueden cambiar mientras se conserva la dirección actual o anterior de operación de levas (la misma u opuesta). Esto es útil cuando la dirección actual no es conocida o no es importante. Para ejecutar una leva por primera vez con Unchanged seleccionado, el control cambia de manera predeterminada la dirección a Same.

Cómo cambiar la dirección de operación de levas

Cuando se selecciona Reverse, la dirección anterior o actual de la leva de tiempo cambia de Same a Opposite o de Opposite a Same. Para ejecutar una leva por primera vez con Reverse seleccionado, el control cambia de manera predeterminada la dirección a Opposite.

Cómo especificar el perfil de levas

Para ejecutar una instrucción MATC, se debe especificar un tag calculado de matriz de datos de perfil de levas. Los tags de matriz de perfil de levas se pueden crear mediante Tag Editor RSLogix 5000 o la instrucción MATC usando el editor de perfil de levas incorporado, o ejecutando una instrucción MCCP en una matriz de levas existente. Consulte la siguiente figura:



Proceso MATC

Es posible modificar los datos dentro de la matriz de perfil de levas en tiempo de compilación usando el editor de perfil de levas o en tiempo de ejecución con la instrucción MCCP. En el caso de cambios en tiempo de ejecución, se debe crear una matriz de levas para usar la instrucción MCCP. Consulte las especificaciones de la instrucción MCCP para obtener más detalles sobre cómo convertir matrices de levas.

Todos los elementos de estructura de la matriz de perfil de levas, excepto los elementos de tipo y estado, están "ocultos" del Tag Editor RSLogix 5000. Estos elementos ocultos no tienen valor. El parámetro Status se usa para indicar que se ha calculado el elemento de matriz de perfil de levas. Si se intenta ejecutar una instrucción de operación de levas con algún elemento no calculado en un perfil de levas, la instrucción dará error. El parámetro Type determina el tipo de interpolación aplicado entre este elemento de matriz de levas y el siguiente elemento de levas.

Controles de la matriz del perfil de levas

El miembro Status del primer elemento de la matriz de perfil de levas es especial y se utiliza para controlar la integridad de datos. Por esta razón, la MATC debe especificar siempre el perfil de levas con el índice de inicio establecido en 0. Este primer miembro Status del elemento del perfil de levas puede tener los siguientes valores:

Valor del estado	Descripción
0	El elemento de perfil de levas no ha sido calculado
1	El elemento de perfil de levas se está calculado
2	El elemento de perfil de levas ha sido calculado
n	El elemento de perfil de levas se ha calculado y se está usando actualmente mediante las instrucciones MAPC (*n*-2) o MATC

Antes de iniciar una leva en un eje especificado, las instrucciones MATC comprueban si la matriz del perfil de levas ha sido calculada controlando el valor del miembro Status del primer elemento de perfil de levas. Si el estado es 0 o 1, el perfil de levas no ha sido calculado aún y la instrucción MATC dará error. Si la matriz del perfil de levas ha sido completamente calculada (Status > 1), la instrucción aumenta el miembro Status indicando que este eje lo está usando.

Cuando la leva finaliza, o interrumpe, el miembro Status del primer elemento de la matriz del perfil de levas disminuye para mantener un seguimiento del número de levas que están usando activamente el perfil de levas asociado.

Interpolación cúbica y lineal

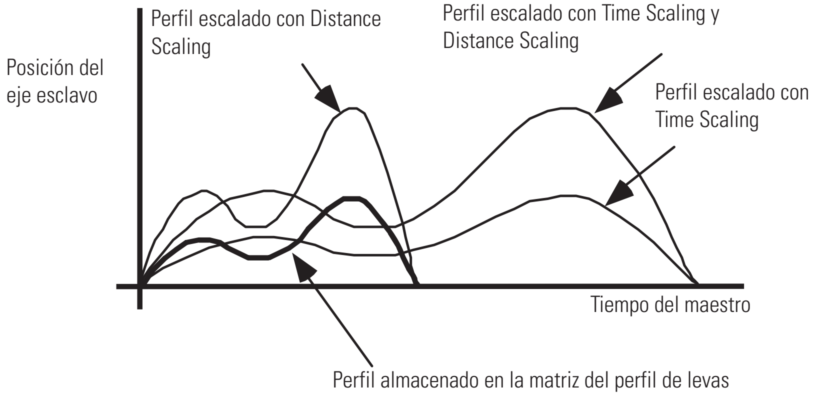
Las levas de tiempo están totalmente interpoladas. Esto significa que si el valor de tiempo del eje maestro actual no corresponde exactamente con un punto en la tabla de levas asociada con el perfil de levas, la posición del eje esclavo está determinada por interpolación lineal o cúbica entre puntos adyacentes. De esta manera, se proporciona el movimiento esclavo más suave posible.

Cada punto de la matriz de levas utilizada para generar el perfil de levas se puede configurar para la interpolación cúbica o lineal.

La operación electrónica de levas permanece activa durante cualquier ejecución subsiguiente de procesos de impulso o movimiento para el eje esclavo. Esto permite que los movimientos de la operación electrónica de levas sean reemplazados con perfiles de impulso o movimiento para crear movimientos complejos o sincronización.

Cómo escalar levas de tiempo

Un perfil de leva de tiempo se puede escalar tanto en tiempo y distancia cuando se ejecuta. Este escalado es útil para permitir al perfil de levas almacenado ser usado para la *forma* del movimiento con el escalado utilizado para definir el tiempo o distancia, a la cual el perfil se ejecuta, como se muestra a continuación.



Cómo escalar levas de tiempo

Cuando una matriz de perfil de levas es especificada por una instrucción MATC, los valores coordinados de maestro definidos por la matriz del perfil de levas toman las unidades de tiempo (segundos) y los valores de esclavo toman las unidades del eje esclavo. En contraste, los parámetros Time Scaling y Distance Scaling son valores “sin unidades” que se usan simplemente como multiplicadores del perfil de levas.

De manera predeterminada, tanto los parámetros Time Scaling y Distance Scaling se establecen en 1. Para escalar un perfil de leva de posición, introduzca un valor para Time Scaling o Distance Scaling que no sea 1.

Al aumentar el valor de Time Scaling de un perfil de levas *se disminuyen* las velocidades y aceleraciones del perfil, mientras que, al aumentar el valor de Distance Scaling *se aumentan* las velocidades y aceleraciones del perfil. Para mantener las velocidades y aceleraciones del perfil escalado aproximadamente igual a aquellas del perfil no escalado, los valores Time Scaling y Distance Scaling deberán ser iguales. Por ejemplo, si el valor de Distance Scaling de un perfil es 2, el valor de Time Scaling deberá ser también 2 para mantener las velocidades y aceleraciones aproximadamente igual durante la ejecución de la leva de tiempo escalada.

ATENCIÓN ⚠ Si disminuye el valor de Time Scaling o aumenta el valor de Distance Scaling de una leva de tiempo, aumentan las velocidades y aceleraciones necesarias del perfil. Esto puede provocar un fallo de movimiento si se exceden las capacidades del sistema del variador.

Modos de ejecución del perfil de levas

Es posible seleccionar Once o Continuous como modos de ejecución para determinar cómo el movimiento de levas se comportará cuando el tiempo se mueva más allá de los puntos de inicio y final del perfil definido en la tabla de levas original.

Si se selecciona Once (predeterminado), el movimiento del perfil de levas del eje comienza inmediatamente. Cuando el tiempo de ejecución de la leva de tiempo supera el rango de tiempo definido por el perfil de levas, la instrucción MATC se completa, el movimiento del eje se detiene, y el bit Time Cam Status en la palabra Motion Status del eje esclavo se borra.

Cuando se selecciona modo Continuous, el perfil de levas especificado comienza inmediatamente y se ejecuta indefinidamente. Con funcionamiento continuo, el tiempo se “desbobina” hacia el comienzo del perfil de levas cuando se mueve más allá del final del perfil de levas, lo cual hace que el perfil de levas se repita de manera indefinida. Esta característica es particularmente útil en aplicaciones giratorias, en las cuales es necesario que la leva de tiempo marche continuamente de modo giratorio o recíproco. Para generar movimientos continuos suaves usando esta técnica, no obstante, se debe tener cuidado al designar los puntos de leva de la tabla de levas para asegurarse de que no hay discontinuidades de posición, velocidad o aceleración entre los puntos de inicio y fin del perfil de levas calculado.

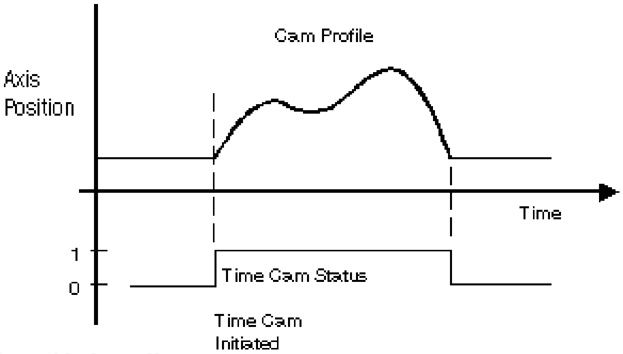
Execution Schedule

El cronograma de ejecución de la instrucción MATC se controla mediante el parámetro Execution Schedule.

Immediate Execution

De manera predeterminada, la instrucción MATC está programada para ejecutarse inmediatamente, puesto que la configuración predeterminada del parámetro Execution Schedule es Immediate. En este caso, no hay retardo en la habilitación del proceso de operación de levas de tiempo.

Como se ilustra en el diagrama a continuación, cuando se ejecuta la instrucción MATC, el proceso de operación de levas se inicia en el eje especificado y se establece el bit Time Cam Status en la palabra Motion Status del eje. Si el parámetro Execution Schedule se establece en Inmediate, el eje se bloquea inmediatamente en el eje maestro según el perfil de levas especificado.



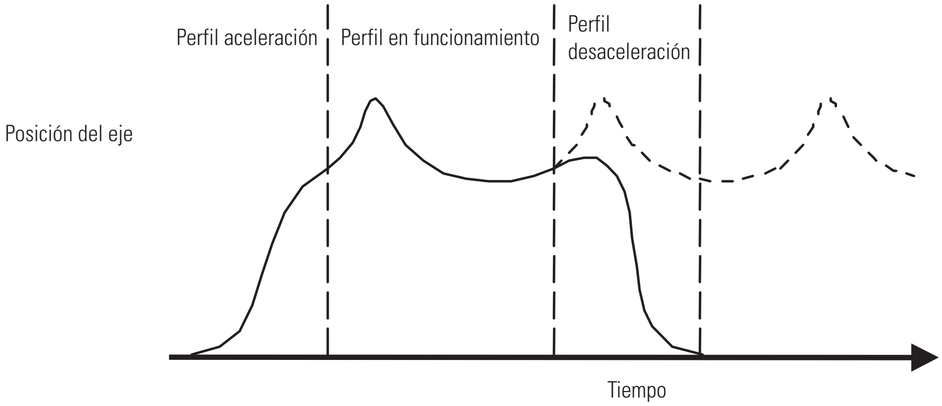
Ejecución inmediata

Si una instrucción MATC se ejecuta en un eje que está ya en operación de levas de tiempo de manera activa, se generará un error Cambio dinámico no válido (código de error 23). La única excepción a esto es cuando Execution Schedule está especificado como ‘pending’.

Ejecución de levas pendiente

Como alternativa, la ejecución de la instrucción MATC puede dejarse con la finalización pendiente de una leva de tiempo actualmente en ejecución. Se puede seleccionar Pending para el parámetro Execution Schedule para mezclar a la perfección dos perfiles de leva de tiempo sin detener el movimiento.

La característica de ejecución Pending es particularmente útil en aplicaciones en las cuales el eje debe acelerarse hasta la máxima velocidad usando un perfil de velocidad específico. Cuando este perfil de aceleración está listo, se debe mezclar suavemente con un perfil en de levas, el cual generalmente se ejecuta de manera continua. Para detener el eje, el perfil en funcionamiento se mezcla suavemente con un perfil de desaceleración, de manera que el eje se detiene en un lugar conocido como se muestra a continuación.



Ejecución de levas pendiente

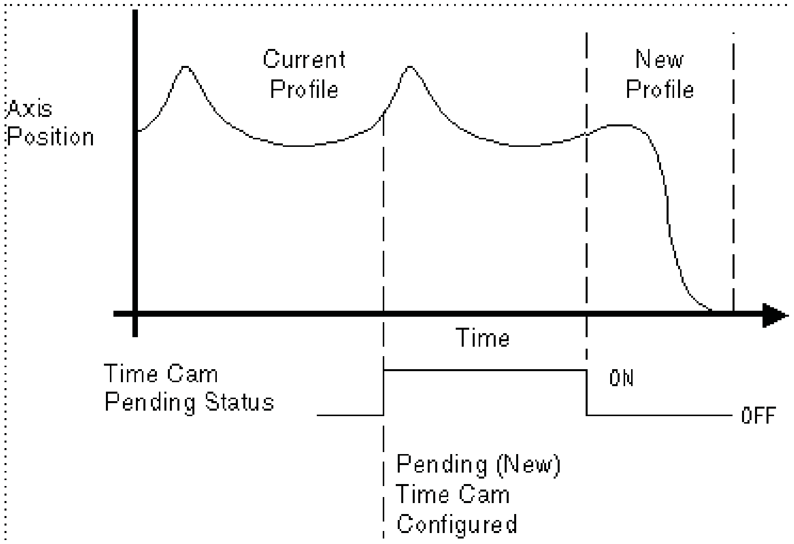
Ejecutando el perfil de levas de tiempo como un perfil de levas Pendiente mientras se ejecuta aún el perfil actual, los parámetros del perfil de levas adecuados se configuran con antelación. Esto hace la transición desde el perfil actual al perfil pendiente a la perfección – se mantiene la sincronización entre la posición de los ejes esclavo y tiempo de maestro. Sin embargo, para garantizar un movimiento suave durante la transición, los perfiles deben estar diseñados de manera que no existan discontinuidades de posición, velocidad o aceleración entre el final del perfil actual y el inicio del nuevo. Esto se realiza usando el editor de perfil de levas RSLogix 5000.

Una vez ejecutada una instrucción pendiente de levas de tiempo, el nuevo perfil de levas hace efecto automáticamente (y se transforma en el perfil actual) cuando la leva de tiempo pasa por el punto final del perfil actual. Si la leva actual está configurada para ejecutarse una vez, el nuevo perfil se inicia cuando termina de pasar por el perfil de levas actual y se establece el bit PC de la instrucción MATC actualmente activa. Si la leva actual está configurada para ejecutarse de manera continua, el nuevo perfil se inicia cuando termina de pasar por el perfil de levas actual y se borra el bit IP de la instrucción MATC actualmente activa. El controlador de movimiento mantiene un seguimiento del tiempo y las posiciones del eje en relación al primer perfil en el momento del cambio y usa esta información para mantener la sincronización entre los perfiles.

Si el parámetro Execution Schedule de una instrucción MATC se establece en Inmediate y un perfil de levas de tiempo está actualmente en proceso, la instrucción MATC genera un error Cambio dinámico no válido.

Si se selecciona un parámetro Execution Schedule como Pending sin un perfil de leva de tiempo correspondiente en progreso, la instrucción MATC se ejecuta, pero no ocurrirá ningún movimiento de operación de levas si no se inicia otra instrucción MATC con un parámetro Execution Schedule no pendiente. Esto permite precargar los perfiles de leva pendiente antes de ejecutar la leva inicial. Este método se aplica en casos en los cuales las levas inmediatas finalizarían antes de que la leva pendiente pueda cargarse de manera confiable.

Una vez configurada una leva de tiempo Pending, el bit Time Cam Pending Status de la palabra Motion Status se establece en 1 (verdadero) para el eje especificado. Cuando el perfil pendiente (nuevo) se inicia y se transforma en el perfil actual, el bit Time Cam Pending Status se borra inmediatamente como se muestra a continuación.



Leva de tiempo pendiente

Cómo detener una leva

Al igual que otros generadores de movimiento (impulso, movimiento, engranaje), las levas activas deben detenerse con las diferentes instrucciones de paro, MAS o MGS. El movimiento de levas se debe detener también cuando el procesador ControlLogix cambia los modos OS. La instrucción MAS, en particular, debe estar habilitada para detener específicamente el proceso de operación de levas. Este proceso debe ser idéntico a la funcionalidad MAS que detiene específicamente un proceso de engranaje.

Cómo incorporar a partir de una leva

Al igual que otros generadores de movimiento (impulso, movimiento, engranaje), las levas activas deben cumplir con la funcionalidad de incorporación de movimiento. Los movimientos y los impulsos, en particular, deben estar habilitados para incorporar a partir de operaciones de levas activas. Este proceso debe ser idéntico a la funcionalidad de incorporación aplicada a un proceso de engranaje.

IMPORTANTE La ejecución de la instrucción MATC se completa en un único escán, por lo tanto, el bit Listo (.DN) y el bit En Proceso (.IP) se establecen inmediatamente. El bit En Proceso (.IP) permanece establecido hasta que el proceso de operación de levas de tiempo iniciado es reemplazado por otra instrucción MATC o interrumpido por un comando de paro del eje de movimiento, operación de incorporación o acción ante un fallo del servo.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Consulte Códigos de error (ERR) para las instrucciones de control de movimiento en página 395.

Códigos de error extendidos: Los códigos de error extendidos proporcionan información adicional específica de instrucción para los códigos de error que son genéricos a muchas instrucciones. Los códigos de error extendidos para el código de error Parámetro fuera de rango (13) incluyen un número que hace referencia al número de operando tal y como aparecen en la plantilla de arriba hacia abajo, con el primer operando como cero. Por lo tanto, para la instrucción MATC, un código de error extendido de 5 haría referencia al valor del operando Time Scaling. Entonces, debería verificar su valor con el rango aceptado de valores para la instrucción.

Para el código de error 54 – El valor máximo de desaceleración es cero, si el error extendido devuelve un número positivo (0-*n*), está haciendo referencia al eje en error en el sistema de coordenadas. Diríjase a la ficha general Coordinate System Properties y mire debajo de la columna de corchetes de la cuadrícula del eje para determinar qué eje tiene un valor máximo de desaceleración de cero. Haga clic en el botón de elipsis al lado del eje en error para acceder a la pantalla de propiedades del eje. Diríjase a la ficha Dynamics y haga el cambio apropiado al valor máximo de desaceleración. Si el número del error extendido es –1, esto significa que el sistema de coordenadas tiene un valor máximo de desaceleración de 0. Diríjase a la ficha Coordinate System Properties Dynamics para corregir el valor máximo de desaceleración.

MATC cambia a bits de estado: *Bits de estado*

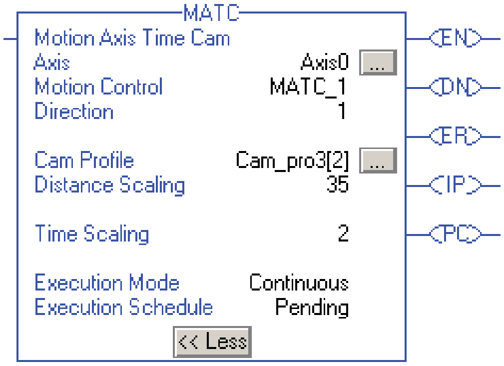
Si el parámetro Execution Schedule se establece en Immediate, la ejecución de la instrucción MATC simplemente establece el bit Time Cam Status en Verdadero.

Nombre del bit	Estado	Significado
TimeCamStatus	VERDADERO	La operación de levas de tiempo está habilitada
TimeCamPendingStatus	FALSO	No hay leva de tiempo pendiente

Si Execution Schedule se establece en Pending, la ejecución de la instrucción MATC no afecta el estado actual del bit Time Cam Status. El bit Time Cam Pending Status se establece inmediatamente en Verdadero y pasa a Falso cuando la leva pendiente se transforma en leva activa.

Nombre del bit	Estado	Significado
TimeCamStatus	N/A	La operación de levas de tiempo está habilitada
TimeCamPendingStatus	VERDADERO	Leva de tiempo pendiente

Ejemplo: *Lógica de escalera de relés*



MATC
Motion Axis Time Cam
Axis Axis0
Motion Control MATC_1
Direction 1
Cam Profile Cam_pro3[2]
Distance Scaling 35
Time Scaling 2
Execution Mode Continuous
Execution Schedule Pending

Ejemplo de lógica de escalera de MATC

Texto estructurado

MATC(Axis0,MATC_1,1,Cam_pro3[2],35,2,Continuous, Pending);

Motion Calculate Slave Values (MCSV)

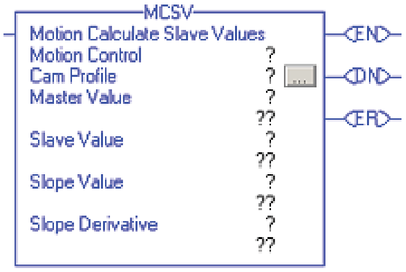
Use la instrucción MCSV para calcular el valor del esclavo, el valor de pendiente y la derivada de la pendiente para un valor de maestro y perfil de levas dados.

ATENCIÓN

Utilice un tag de control de movimiento sólo una vez. No vuelva a utilizarlo en otra instrucción. De lo contrario, podría ocasionar un movimiento inesperado del equipo y provocar lesiones a personas.

Operandos:

Lógica de escalera de relés



Operando	Tipo	Formato	Descripción
Motion Control	MOTION_INSTRUCTION	tag	Estructura que se utiliza para acceder a los parámetros de estado de la instrucción.
Cam Profile	CAM_PROFILE	tag de matriz	Matriz de elementos con el índice de matrices establecido en 0. Define el perfil de levas usado para calcular los valores del esclavo.
Master Value	SINT, INT, DINT o REAL	Inmediato o tag	Valor exacto a lo largo del eje maestro del perfil de levas que se usa para calcular los valores del esclavo.
Slave Value	REAL	tag	Valor a lo largo del eje esclavo del perfil de levas con el maestro al valor del maestro especificado.
Slope Value	REAL	tag	Primera derivada del valor a lo largo del eje esclavo del perfil de levas con el maestro al valor del maestro especificado.
Slope Derivative	REAL	tag	Segunda derivada del valor a lo largo del eje esclavo del perfil de levas con el maestro al valor del maestro especificado.

Texto estructurado

```
MCSV(MotionControl,CamProfile,
MasterValue,SlaveValue,
SlopeValue,SlopeDerivative)
```

Los operandos son iguales a los de la instrucción MCSV de lógica de escalera de relés.

Descripción:

La instrucción MCSV determina el valor del esclavo, el valor de pendiente y la derivada de la pendiente para un valor de maestro y perfil de levas dados. Como extensión de la funcionalidad de operación de levas de tiempo y posición, proporciona los valores esenciales para la recuperación de fallos durante las operaciones de levas.

Motion Control

Los siguientes bits de control están afectados por la instrucción MCSV.

Mnemónico	Descripción
Bit .EN (Habilitado) 31	El bit Habilitado se establece cuando el renglón hace la transición de falso a verdadero. Se reestablece cuando el renglón va de verdadero a falso.
Bit .DN (Listo) 29	El bit Listo se establece cuando los valores del esclavo han sido calculados correctamente. Se reestablece cuando el renglón va de falso a verdadero.
Bit .ER (Error) 28	El bit Error se establece cuando los valores del esclavo no han sido calculados correctamente. Se reestablece cuando el renglón va de falso a verdadero.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Consulte Códigos de error (ERR) para las instrucciones de control de movimiento en página 395.

Códigos de error extendidos: Los códigos de error extendidos proporcionan información adicional específica de instrucción para los códigos de error que son genéricos a muchas instrucciones. Los códigos de error extendidos para el código de error Parámetro fuera de rango (13) incluyen un número que hace referencia al número de operando tal y como aparecen en la plantilla de arriba hacia abajo, con el primer operando como cero. Por lo tanto, para la instrucción MCSV, un código de error extendido de 2 haría referencia al valor del operando Master Value. Entonces, debería verificar su valor con el rango aceptado de valores para la instrucción.

MCVS Cambia a bits de estado: ninguno

Instrucciones de grupo de movimiento

(MGS, MGSD, MGSR, MGSP)

ATENCIÓN

Los tags usados para el atributo de control de movimiento de las instrucciones se deben usar una sola vez. La reutilización del tag de control de movimiento en otras instrucciones puede causar un funcionamiento inesperado. Esto puede ocasionar daños en los equipos o lesiones personales.

Introducción

Las instrucciones de control de grupo incluyen todas las instrucciones de control de movimiento que operan en todos los ejes el en grupo especificado. Las instrucciones que pueden aplicarse a grupos incluyen estroboscopio, control de interrupción e instrucciones de paro. Tenga en cuenta que actualmente sólo un grupo es compatible con el controlador Logix.

Las instrucciones de grupo de movimiento son:

Si desea	Use esta instrucción	Disponible en estos lenguajes
Iniciar un paro de movimiento en un grupo de ejes.	MGS	lógica de escalera de relés texto estructurado
Forzar todos los ejes de un grupo en el estado de interrupción de operación.	MGSD	lógica de escalera de relés texto estructurado
Realizar una transición de un grupo de ejes del estado operativo de interrupción al estado de eje listo.	MGSR	lógica de escalera de relés texto estructurado
Enclavar la posición de comando actual y real de todos los ejes en un grupo.	MGSP	lógica de escalera de relés texto estructurado

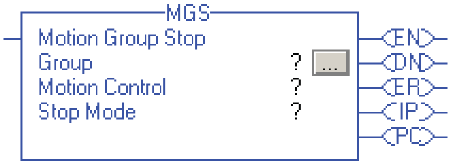
Motion Group Stop (MGS)

La instrucción MGS inicia un paro de todos los movimientos en progreso en todos los ejes del grupo especificado mediante un método configurado individualmente para cada eje, o como un grupo mediante el modo de paro de la instrucción MGS. Si se especifica el modo de paro de MGS como Programmed, se para cada eje del grupo de acuerdo con el atributo del eje del modo de paro programado. Este es el mismo mecanismo de paro empleado por el sistema operativo Logix cuando hay un cambio de estado del controlador Logix. Este atributo de modo de paro programado actualmente proporciona cinco métodos distintos para parar un eje:

- Paro rápido
- Inhabilitación rápida
- Inhabilitación basada en hardware
- Interrupción rápida
- Interrupción basada en hardware

Alternativamente, se puede seleccionar un modo de paro explícito utilizando la instrucción MGS. Si se selecciona un modo de paro de inhabilitación rápida, todos los ejes del grupo paran con un comportamiento de inhabilitación rápida. Cuando el movimiento de todos los ejes de un grupo ha sido llevado a paro, se establece el bit de Proceso completo (PC) en la estructura de control.

Operandos: *Lógica de escalera de relés*



Operando	Tipo	Formato	Descripción
Group	MOTION_ GROUP	tag	Nombre del grupo de ejes en el cual se debe realizar la operación.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura utilizada para obtener acceso a los parámetros de estado de la instrucción.
Stop Mode	UDINT	Inmediato	Controla cómo se paran los ejes del grupo. Seleccione uno de los siguientes métodos: 0 = Programmed – cada eje se detiene según como se haya configurado el eje individual. 1 = Fast Stop – se desacelera cada eje del grupo a la velocidad máxima de desaceleración y el eje detenido se deja en estado de servo activo. 2 = Fast Disable – cada eje del grupo se desacelera a la velocidad máxima de desaceleración y el eje detenido se coloca en estado de servo listo.

Texto estructurado

MGS(Group,MotionControl,
StopMode);

Los operandos son iguales a los de la instrucción MGS de lógica de escalera de relés.

En el caso de los operandos para los que usted debe seleccionar entre opciones disponibles, introduzca su selección como:

Este operando	**Tiene estas opciones que usted...**	
	introduce como texto	**o introduce como número**
StopMode	programmed	0
	faststop	1
	fastdisable	2

Estructura MOTION_INSTRUCTION

Mnemónico	**Descripción**
Bit .EN (Habilitar) 31	Se establece cuando el renglón realiza una transición de falso a verdadero y permanece establecido hasta que se completa la transacción de mensajes del servo y la condición de entrada de renglón pasa a falso.
Bit .DN (Listo) 29	Se establece cuando el grupo de paro programado se ha iniciado correctamente para todos los ejes en el grupo.
Bit .ER (Error) 28	El bit Error indica cuando la instrucción detecta un error, como si se hubiese especificado un grupo no configurado.
Bit .IP (En proceso) 26	Se establece en transición de renglón positiva y se borra luego de que el paro de grupo de movimiento se ha completado.
Bit .PC (Proceso Completo) 27	Se establece luego de que todos los ejes en el grupo hayan sido llevados al paro correctamente de acuerdo con cada configuración de modo de paro programado.

Descripción:

Con el parámetro Stop Mode establecido en Programmed, la instrucción MGS proporciona movimiento a todos los ejes del grupo especificado de acuerdo con el modo de paro programado para cada eje. Si el eje tiene tanto movimientos de eje individual como movimientos de eje coordinado, la instrucción MGS detiene el movimiento del eje individual utilizando la velocidad máxima de desaceleración del eje y detiene el movimiento coordinado de ejes utilizando la velocidad máxima de desaceleración del sistema de coordenadas. Para la desaceleración, siempre se utiliza un perfil trapezoidal independientemente del tipo de perfil programado.

La instrucción MGS inicia la misma acción de paro programado que se aplica automáticamente cuando el sistema operativo del proceso cambia el modo de funcionamiento. Esto es particularmente útil para diseñar gestores de fallos de movimiento personalizados.

Si el parámetro Stop Mode de MGS se establece en Fast Stop, se fuerza cada eje del grupo para realizar un proceso de paro rápido, independientemente del modo de paro programado. Cada eje del grupo se desacelera a la velocidad máxima de desaceleración y el eje, una vez detenido, se deja en estado de servo activado.

Si el parámetro Stop Mode de la instrucción MGS se establece en Fast Disable, se fuerza cada eje del grupo para realizar un proceso de inhabilitación rápida, independientemente del modo de paro programado. Cada eje del grupo se desacelera a la velocidad máxima de desaceleración y, una vez detenido, se coloca en el estado de eje listo (servo inactivo y variador inhabilitado).

Hay cinco modos de paro programados que actualmente son compatibles con la instrucción MGS. Fast Stop, Fast Disable, Hard Disable, Fast Shutdown, y Hard Shutdown. Cada eje se puede configurar para que use cualquiera de estos cinco modos de paro. A continuación, hay una descripción del efecto de cada uno de estos cinco modos de paro tal como se aplican en un eje individual del grupo especificado.

Fast Stop

En el caso de un eje configurado para un paro rápido, la instrucción MGS inicia un paro controlado muy parecido al iniciado por la instrucción MAS. En ese caso, la instrucción MGS lleva el eje de movimiento a un paro controlado sin inhabilitar el lazo del servo del eje. Es útil cuando se desea un paro rápido desacelerado del eje con un control de servo retenido.

La instrucción MGS usa la desaceleración máxima configurada del eje para detener sólo el movimiento individual del eje.

La porción de movimiento coordinado del eje utiliza la desaceleración máxima del sistema de coordenadas configurada para detener el eje.

Fast Disable

En el caso de un eje configurado en Fast Disable, la instrucción MGS inicia un paro controlado muy similar al iniciado por una instrucción MAS con la excepción de que el variador está inhabilitado cuando llega a un paro. Use la instrucción MGS cuando desee un paro rápido desacelerado del eje antes de que se inhabilite el variador.

La instrucción MGS usa la desaceleración máxima configurada del eje para detener sólo el movimiento individual del eje.

La porción de movimiento coordinado del eje utiliza la desaceleración máxima del sistema de coordenadas configurada para detener el eje.

Hard Disable

Para un eje configurado en Hard Disable la instrucción MGS inicia el equivalente de una instrucción MSF para el eje. Esta acción apaga automáticamente la salida apropiada del variador del eje, e inhabilita el lazo del servo. Según la configuración del variador, esto puede provocar que el eje realice un paro por inercia pero proporciona la desconexión más rápida de la alimentación eléctrica de salida del variador.

Fast Shutdown

En el caso de un eje configurado en Fast Shutdown, la instrucción MGS inicia un paro rápido y luego aplica el equivalente de una instrucción MASD en el eje. Esta acción APAGA la salida apropiada del variador del eje, inhabilita el lazo del servo, abre los contactos OK de cualquier módulo de movimiento asociado y coloca el eje en estado de interrupción.

Hard Shutdown

En el caso de un eje configurado en Hard Shutdown, la instrucción MGS inicia el equivalente a la instrucción MASD al eje. Esta acción APAGA la salida apropiada del variador del eje, inhabilita el lazo del servo, abre los contactos OK de cualquier módulo de movimiento asociado y coloca el eje en estado de interrupción. Según la configuración del variador, esto puede provocar que el eje realice un paro por inercia pero proporciona la desconexión más rápida de la alimentación eléctrica del variador mediante los contactos OK.

Para ejecutar correctamente una instrucción MGS, el grupo especificado debe estar configurado.

IMPORTANTE La ejecución de la instrucción MAH puede necesitar la ejecución de múltiples escanes porque los mensajes pueden requerir uno o más módulos de movimiento de eje en el grupo. Por ende, el bit .DN Listo puede no establecerse inmediatamente. Sin embargo, se establece el bit .IP En Proceso y se borra inmediatamente el bit .PC Proceso completo. El bit .IP En proceso permanece establecido hasta que se completa el proceso iniciado de paro programado en todos los ejes en el grupo especificado, o las instrucciones de paro reemplazadas por otra instrucción MGS o interrumpida por una acción ante un fallo del servo. El bit .PC Proceso completo sólo se establece si el perfil de desaceleración iniciado para cada grupo de ejes se ha completado antes de cualquiera de los eventos anteriores que interrumpen el proceso de paro y que borran el bit .IP En proceso.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de borrado a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Vea Códigos de error (ERR) para las instrucciones de control de movimiento.

Bits de estado: *MGS cambia a bits de estado*

Nombre del bit	Estado	Definición
StoppingStatus	VERDADERO	El eje se detiene (Según el modo de paro programado para el eje).
JogStatus	FALSO	El eje no funciona ya por impulsos.
MoveStatus	FALSO	El eje ya no se mueve.
GearingStatus	FALSO	El eje ya no se acopla.
HomingStatus	FALSO	El eje ya no vuelve a la posición inicial.

Ejemplo: Cuando las condiciones de entrada son verdaderas, el controlador detiene el movimiento en todos los ejes del *grupo1*. Luego de que el controlador detiene todo el movimiento, los ejes se inhiben.

Lógica de escalera de relés

MGS

Motion Group Stop

Group Motion

Motion Control MGS_1

Stop Mode Programmed

Ejemplo de lógica de escalera de MGS

Texto estructurado

MGS(Motion,MSG_1,Programmed);

Motion Group Shutdown (MGSD)

Use la instrucción MGSD para forzar todos los ejes de un grupo en el estado de desactivación de operación. El estado de interrupción de un eje es de servo desactivado, la salida del variador está desactivada, y se abren los contactos OK de relé de estado sólido del módulo de movimiento, si corresponde. El grupo de ejes permanece en estado de interrupción hasta que se ejecuta Group Shutdown Reset o se restablece individualmente cada eje mediante la instrucción MASD.

Operandos: *Lógica de escalera de relés*


MGS
Motion Group Stop
Group Motion
Motion Control MGS_1
Stop Mode Programmed

Operando	Tipo	Formato	Descripción
Grupo	MOTION_ GROUP	tag	Nombre del grupo de ejes en el cual se debe realizar la operación.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura utilizada para obtener acceso a los parámetros de estado de instrucción.

Texto estructurado

MGSD(Group,MotionControl);

Los operandos son iguales a los de la instrucción MGSD de lógica de escalera de relés.

Estructura MOTION_INSTRUCTION

Mnemónico	Descripción
Bit .EN (Habilitar) 31	El bit Habilitar indica cuándo la instrucción está habilitada. Permanece establecido hasta que se completa el envío de mensajes al servo y la condición de entrada del renglón pasa a falsa.
Bit .DN (Listo) 29	El bit Listo indica cuando la instrucción establece el grupo de ejes en el estado de interrupción de operación.
Bit .ER (Error) 28	El bit Error indica cuándo la instrucción detecta un error, como si un hubiera un fallo de envío de mensaje al servomódulo.

Descripción: La instrucción MGSD apaga la salida del variador, inhabilita los lazos del servo de todos los ejes en el grupo especificado, y abre todos los contactos OK asociados para todos los módulos de movimientos correspondientes en el grupo. Esta acción coloca todos los ejes del grupo en estado de interrupción. La instrucción MGSD toma sólo un parámetro; simplemente seleccione o introduzca el grupo deseado para la interrupción.

Otra acción iniciada por la instrucción MGSD es el borrado de todos los procesos de movimiento en progreso y el borrado de todos los bits de estado de movimiento. Asociado con esta acción, el comando también borra los bits .IP de todas las instrucciones de control de movimiento que puedan estar establecidos en ese momento para cada eje del grupo.

La instrucción MGSD fuerza al grupo especificado de ejes al estado de interrupción. Una de las características únicas del estado de interrupción es que el se abre contacto OK de relé de estado sólido para todos los módulos de movimiento del grupo. Esta función puede utilizarse para abrir todas las cadenas de paro de emergencia que controlan la alimentación eléctrica principal en los diferentes sistemas del variador.

Otra característica del estado de interrupción es que cualquier instrucción que inicia el movimiento del eje dentro del grupo tiene bloqueada la ejecución. Los intentos de hacer esto dan como resultado un error de ejecución. Sólo ejecutando una de las instrucciones para restablecer interrupción se puede iniciar el movimiento correctamente.

Para ejecutar correctamente una instrucción MGSD, se debe crear y configurar el grupo especificado.

IMPORTANTE La ejecución de la instrucción MGSD puede necesitar la ejecución de múltiples escanes porque requiere la transmisión de múltiples mensajes al módulo de movimiento. Por lo tanto, el bit Listo (.DN) no se establecerá inmediatamente, sino sólo cuando este mensaje haya sido correctamente transmitido.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de borrado a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Vea Códigos de error (ERR) para las instrucciones de control de movimiento.

Bits de estado: *MGSD cambia a bits de estado*

Nombre del bit	Estado	Definición
ServoActionStatus	FALSO	Axis es un estado de eje listo con el lazo del servo inactivo.
DriveEnableStatus	FALSO	La salida de habilitación del eje del variador está inactiva.
ShutdownStatus	VERDADERO	El eje está en estado de interrupción.
AccelStatus	FALSO	El eje no acelera.
DecelStatus	FALSO	El eje no desacelera.
GearingLockStatus	FALSO	El eje no está bloqueado.
JogStatus	FALSO	El eje no funciona por impulsos.
MoveStatus	FALSO	El eje no se mueve.
GearingStatus	FALSO	El eje no se acopla.
HomingStatus	FALSO	El eje no vuelve la posición inicial.

Ejemplo: Cuando las condiciones de entrada son verdaderas, el controlador fuerza todos los ejes del *grupo1* hacia un estado operativo de interrupción.

Lógica de escalera de relés

MGSD
Motion Group Shutdown
Group Motion
Motion Control MGSD_2

Ejemplo de escalera de relé de MGSD

Texto estructurado

MGSD(Motion,MGSD_2);

Motion Group Shutdown Reset (MGSR)

Use la instrucción MGSR para transferir un grupo de ejes del estado operativo de interrupción hacia el estado operativo de eje listo. Como resultado de este comando, todos los fallos asociados con los ejes del grupo se borran y todos los contactos OK de relé de los módulos de movimiento asociados con el grupo especificado se cierran.

Operandos: *Lógica de escalera de relés*

Operando	Tipo	Formato	Descripción
Grupo	MOTION_ GROUP	tag	Nombre del grupo de ejes en el cual se debe realizar la operación.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura utilizada para obtener acceso a los parámetros de estado de instrucción.

Texto estructurado

MGSR(Group,MotionControl);

Los operandos son iguales a los de la instrucción MGSR de lógica de escalera de relés.

Estructura MOTION_INSTRUCTION

Mnemónico	Descripción
Bit .EN (Habilitar) 31	El bit Habilitar indica cuándo la instrucción está habilitada. Permanece establecido hasta que se completa el envío de mensajes al servo y la condición de entrada del renglón pasa a falsa.
Bit .DN (Listo) 29	El bit Listo indica cuándo la instrucción restablece el grupo de ejes desde el estado operativo de interrupción.
Bit .ER (Error) 28	El bit Error indica cuándo la instrucción detecta un error, como si un hubiera un fallo en el envío de mensajes al servomódulo.

Descripción: La instrucción MGSR saca todos los ejes del grupo especificado del estado de interrupción borrando todos los fallos de eje y cerrando todos los contactos OK de relé en estado sólido para los módulos de movimiento dentro del grupo. Esta acción coloca todos los ejes dentro del grupo de movimiento en el estado eje listo.

Tal como la instrucción MGSD fuerza todos los ejes del grupo especificado al estado de interrupción. La instrucción MGSR saca todos los ejes del grupo especificado del estado de interrupción y los coloca en el estado eje listo. Una de las características únicas del estado de interrupción es que, si es compatible, se abre el contacto OK de relé en estado sólido para todos los módulos de grupo de movimiento. Por lo tanto, el resultado de una instrucción MGSR aplicada a un grupo de módulos de movimiento es que todos los contactos OK de relé del módulo de movimiento se cierran. Esta función puede ser utilizada para cerrar las cadenas de paro de emergencia que controlan la alimentación eléctrica principal de varios sistemas de variadores y permiten que el cliente vuelva a aplicar la alimentación eléctrica a los variadores.

Para ejecutar correctamente una instrucción MGSR, el grupo especificado debe estar configurado.

IMPORTANTE La ejecución de la instrucción MGSR puede necesitar múltiples escanes porque requiere la transmisión de un mensaje al módulo de movimiento. El bit Listo .DN no se establecerá inmediatamente, sino sólo cuando este mensaje haya sido correctamente transmitido.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de borrado a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Vea Códigos de error (ERR) para las instrucciones de control de movimiento.

Bits de estado: *MGSR cambia a bits de estado*

Nombre del bit	Estado	Definición
ServoActionStatus	FALSO	El eje está en un estado de eje listo con el lazo del servo inactivo.
DriveEnableStatus	FALSO	La salida de habilitación del eje del variador está inactiva.
ShutdownStatus	FALSO	El eje NO está en estado de interrupción.

Ejemplo: Cuando las condiciones de entrada son verdaderas, el controlador transfiere todos los ejes del *grupo1* de estado operativo de interrupción al estado operativo de eje listo.

Lógica de escalera de relés

MGSR
Motion Group Shutdown Reset
Group Motion
Motion Control MGSR_3

Ejemplo de lógica de escalera de MGSR

Texto estructurado

MGSR(Motion,MGSR_3);

Motion Group Strobe Position (MGSP)

Use la instrucción MGSP para enclavar la posición de comando actual y real de todos los ejes en el grupo especificado en un sólo punto en el tiempo. Las posiciones enclavadas están disponibles en los parámetros StrobeActualPosition y StrobeCommandPosition en el Motion Axis Object para cada eje configurado del grupo.

Operandos:

Lógica de escalera de relés

Operando	Tipo	Formato	Descripción
Grupo	MOTION_ GROUP	tag	Nombre del grupo de ejes en el cual se debe realizar la operación.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura utilizada para obtener acceso a los parámetros de estado de instrucción.

Texto estructurado

```
MGSP(Group,MotionControl);
```

Los operandos son iguales a los de la instrucción MGSP de lógica de escalera de relés.

Estructura MOTION_INSTRUCTION

Mnemónico	Descripción
Bit .EN (Habilitar) 31	Se establece cuando el renglón realiza una transición de falso a verdadero y permanece establecido hasta que se completa el envío de mensajes del servo y la condición de entrada de renglón pasa a falsa.
Bit .DN (Listo) 29	Se establece cuando el grupo de ejes ha sido establecido correctamente en el estado de interrupción.
Bit .ER (Error) 28	El bit Error indica cuando la instrucción ha detectado un error, como si se hubiese especificado un grupo no configurado.

Descripción: La instrucción MGSP enclava sincrónicamente todos los valores de posición de comando y real de todos los ejes del grupo especificado en el momento de la ejecución. La instrucción MGSP toma sólo un parámetro; simplemente seleccione o introduzca el eje deseado para el estroboscopio.

Si el grupo especificado no aparece en la lista de grupos disponibles, el grupo no ha sido configurado para la operación. Utilice el editor de tags para crear y configurar un nuevo grupo.

La instrucción MGSP se puede utilizar en cualquier momento para captar un conjunto completo de información de posición de comando real para todos los ejes del grupo especificado. Esta operación usualmente es necesaria como precursora de cálculos que afectan los valores de posición de diferentes ejes dentro del grupo.

Para ejecutar correctamente una instrucción MGSP, el grupo especificado debe estar configurado.

IMPORTANTE La ejecución de la instrucción MGSP se completa en un sólo escán, lo cual establece inmediatamente el bit Listo .DN.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de borrado a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Vea Códigos de error (ERR) para las instrucciones de control de movimiento.

Bits de estado: *MGSP cambia a bits de estado*

Ninguno

Ejemplo: Cuando las condiciones de entrada son verdaderas, el controlador se enclava en la posición de comando actual y real de todos los ejes en el *grupo1*.

Lógica de escalera de relés

MGSP
Motion Group Strobe Position
Group Motion
Motion Control MGSP_2

Ejemplo de lógica de escalera de MGSP

Texto estructurado

MGSP(Motion,MGSP_2);

Notas:

Capítulo 5

Instrucciones de evento de movimiento

(MAW, MDW, MAR, MDR, MAOC, MDOC)

ATENCIÓN

Los tags usados para el atributo de control de movimiento de las instrucciones se deben usar una sola vez. La reutilización del tag de control de movimiento en otras instrucciones puede causar un funcionamiento inesperado. Esto puede producir daños a maquinaria o lesiones personales.

Introducción

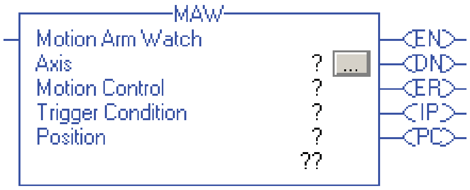
Las instrucciones de evento de movimiento controlan la activación y la desactivación de las funciones de verificación de eventos especiales, tales como la posición de registro y control. Las instrucciones de evento de movimiento son:

Si desea	Use esta instrucción	Disponible en estos lenguajes
Activar la verificación de un evento de posición de control para un eje.	MAW	lógica de escalera de relés texto estructurado
Desactivar la verificación de un evento de posición de control para un eje.	MDW	lógica de escalera de relés texto estructurado
Activar la verificación de evento de registro del servomódulo para un eje.	MAR	lógica de escalera de relés texto estructurado
Desactivar la verificación de evento de registro del servomódulo para un eje.	MDR	lógica de escalera de relés texto estructurado
Activar una leva de salida	MAOC	lógica de escalera de relés texto estructurado
Desactivar una leva de salida	MDOC	lógica de escalera de relés texto estructurado

Motion Arm Watch (MAW)

Use la instrucción MAW para activar la verificación del evento de posición de control del módulo de movimiento para un eje especificado. Cuando se solicita esta instrucción, se habilita un evento de posición de control utilizando la posición de control para el eje y para la condición especificada del evento de avance o retroceso. Después de completada la activación, se monitorea la posición real para el eje con la posición de control y cuando se cumple la condición especificada del evento de control, se establece el bit Event (PC), y también el bit Watch Event Status en la estructura de datos del eje.

Operandos: *Lógica de escalera de relés*



Operando	Tipo	Formato	Descripción
AXIS	AXIS_FEEDBACK AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje en el cual se realizará la operación.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura utilizada para obtener acceso a los parámetros de estado de la instrucción.
Trigger condition	BOOLEANO	inmediato	Seleccione la condición de activación del evento de control: 0 = avance – El servomódulo busca cambiar la posición real de un valor menor a un valor mayor que la posición de control. 1 = retroceso – El servomódulo monitorea para ver si la posición real cambia de un valor mayor a un valor menor que la posición de control.
Position	REAL	inmediato o tag	El nuevo valor para la posición de control

Texto estructurado

MAW(Axis,MotionControl,
TriggerCondition,Position);

Los operandos son iguales a los de la instrucción MAW de lógica de escalera de relés.

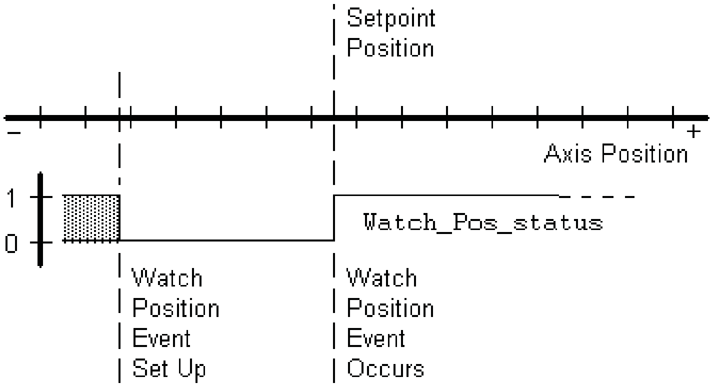
En el caso de los operandos para los que usted debe seleccionar entre opciones disponibles, introduzca su selección como:

Este operando	Tiene estas opciones que usted...	
	introduce como texto	o introduce como número
TriggerCondition	avance	0
	retroceso	1

Estructura MOTION_INSTRUCTION

Mnemónico	Descripción
Bit .EN (Habilitar) 31	Se establece cuando el renglón hace una transición de falso a verdadero y permanece establecido hasta que la transacción de mensaje servo se completa y el renglón pasa a falso.
Bit .DN (Listo) 29	Se establece cuando la verificación del evento de control del eje se ha activado correctamente.
Bit .ER (Error) 28	Se establece para indicar que la instrucción ha detectado un error, como si se hubiese especificado un eje no configurado.
Bit .IP (En proceso) 26	Se establece durante una transición positiva de renglón y se borra luego de que se ha producido un evento de control, o ha sido reemplazado por otro comando Motion Arm Watch, o interrumpido por un comando Motion Disarm Watch.
Bit .PC (Proceso completo) 27	Se establece cuando se produce un evento de control.

Descripción: La instrucción MAW establece un evento de posición de control para que se produzca cuando el eje físico especificado alcance el punto de ajuste de posición especificado, tal como se muestra a continuación.



Posición de punto de ajuste

Los eventos de posición de control son útiles para sincronizar una operación en una posición de eje especificada mientras el eje está en movimiento, como por ejemplo, activar una solenoide para empujar un envase hacia fuera del transportador en una cierta posición del eje. Seleccione o introduzca el eje físico deseado, la condición de activación deseada, e introduzca un valor o variable de tag para la posición de control deseada.

Si el eje objeto no aparece en la lista de ejes disponibles, el eje no ha sido configurado para la operación. Utilice el editor de tags para crear y configurar un nuevo eje.

Cuando se ejecuta la instrucción MAW, el bit WatchEventStatus se establece en 0 (FALSO) y se monitorea la posición real de un eje físico (en el régimen de actualización del lazo del servo) hasta que alcanza la posición de control especificada. Luego de que se produce el evento de posición de control, el bit WatchEventStatus para el eje se establece en 1 (VERDADERO).

Es posible activar múltiples eventos de posición de control en un momento dado; sin embargo, sólo uno puede estar activo a la vez para cualquier eje físico dado. Cada evento se monitorea de manera independiente y se lo puede examinar utilizando el bit WatchEventStatus apropiado.

IMPORTANTE En conexiones de E/S grandes, los valores forzados pueden disminuir la velocidad en la cual el controlador procesa las posiciones de control repetitivas.

Para ejecutar correctamente una instrucción MAW, el eje objeto debe estar configurado como un servoeje o de sólo retroalimentación. De lo contrario, la instrucción dará error.

IMPORTANTE La ejecución de la instrucción MAW puede necesitar la ejecución de múltiples escanes porque requiere la transmisión de múltiples mensajes al módulo de movimiento. El bit Listo (.DN) no se establecerá inmediatamente, sino sólo cuando este mensaje haya sido correctamente transmitido.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute durante una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Vea Códigos de error (ERR) para las instrucciones de control de movimiento.

Códigos de error extendidos: Los códigos de error extendidos proporcionan información adicional específica de instrucción para los códigos de error que son genéricos a muchas instrucciones. Los siguientes códigos de error extendidos ayudan a localizar el problema cuando la instrucción MAW recibe un mensaje de error Fallo de mensajes al servomódulo (12).

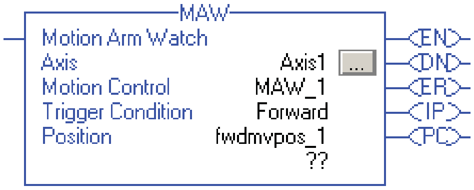
Código de error asociado (decimal)	Código de error extendido (decimal)	Significado
SERVO_MESSAGE_FAILURE (12)	Sin Recurso (2)	No hay suficientes recursos de memoria para completar la solicitud. (SERCOS)

Bits de estado: *MAW cambia a bits de estado*

Nombre del bit	Estado	Significado
WatchEventArmedStatus	Verdadera	El eje está buscando un evento de posición de control.
WatchEventStatus	Falso	Se borra el evento de control anterior.

Ejemplo: Cuando las condiciones de entrada son verdaderas, el controlador activa la verificación del evento de posición de control para *eje1*.

Lógica de escalera de relés



Motion Arm Watch
Axis Axis1
Motion Control MAW_1
Trigger Condition Forward
Position fwdmvpos_1
??

Ejemplo de lógica de escalera MAW

Texto estructurado

MAW(Axis1,MAW_1,Forward,fwdmvpos_1);

Motion Disarm Watch (MDW)

Use la instrucción MDW para desactivar la verificación del evento de posición de control para un eje. La instrucción tiene el efecto de borrado tanto del bit Watch Event Status como del bit Watch Armed Status en la estructura de datos del eje. La ejecución de esta instrucción también borra el bit En Proceso asociado con la instrucción MAW.

Operandos: *Lógica de escalera de relés*

Motion Disarm Watch
Axis ?
Motion Control ?

Operando	Tipo	Formato	Descripción
AXIS	AXIS_FEEDBACK AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje en el cual se realizará la operación.
Control de movimiento	MOTION_ INSTRUCTION	tag	Estructura utilizada para obtener acceso a los parámetros de estado de la instrucción.

Texto estructurado

MAW(Axis,MotionControl);

Los operandos son iguales a los de la instrucción MDW de lógica de escalera de relés.

Estructura MOTION_INSTRUCTION

Mnemónico	Descripción
Bit .EN (Habilitar) 31	Se establece cuando el renglón hace una transición de falso a verdadero y permanece establecido hasta que la transacción de mensaje servo se completa y el renglón pasa a falso.
Bit .DN (Listo) 29	Se establece cuando la verificación del evento de control del eje se ha desactivado correctamente.
Bit .ER (Error) 28	Se establece para indicar que la instrucción ha detectado un error, como si se hubiese especificado un eje no configurado.

Descripción:

La instrucción MDW cancela la verificación del evento de posición de control establecido por la instrucción MAW anterior. La instrucción MDW no requiere parámetros; simplemente introduzca o seleccione el eje físico deseado.

Si el eje objeto no aparece en la lista de ejes disponibles, el eje no ha sido configurado para la operación. Utilice el editor de tags para crear y configurar un nuevo eje.

Para ejecutar correctamente una instrucción MDW, el eje objeto debe estar configurado como un servoeje o de sólo retroalimentación. De lo contrario, la instrucción dará error.

IMPORTANTE La ejecución de la instrucción MDW puede necesitar múltiples escanes porque requiere la transmisión de un mensaje al módulo de movimiento. El bit Listo (.DN) no se establecerá inmediatamente, sino sólo cuando este mensaje haya sido correctamente transmitido.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético:

no afectados

Condiciones de fallo:

ninguno

Códigos de error:

Vea Códigos de error (ERR) para las instrucciones de control de movimiento.

Bits de estado: *MDW cambia a bits de estado*

Nombre del bit	Estado	Significado
WatchEventArmedStatus	falsa	El eje no está buscando un evento de posición de control.
WatchEventStatus	falsa	Se borra el evento de control anterior.

Ejemplo: Cuando las condiciones de entrada son verdaderas, el controlador desactiva la verificación del evento de posición de control para *eje1*.

Lógica de escalera de relés

Ejemplo de lógica de escalera MDW

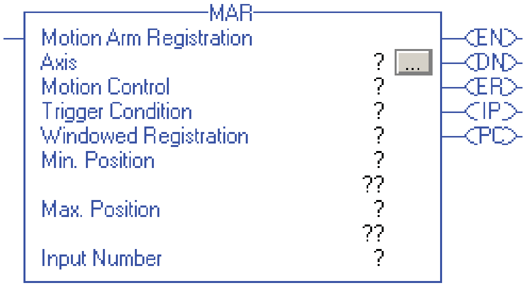
Texto estructurado

MDW(Aaxis1,MDW_1);

Motion Arm Registration (MAR)

Use la instrucción MAR para activar la verificación del evento de registro del servomódulo para un eje especificado. Cuando se solicita esta instrucción, se activa un evento de registro basado en la entrada de registro seleccionada y en la condición de activación especificada. Cuando la transición del registro de entrada especificado cumple con la condición de activación, el módulo de movimiento calcula la posición del eje en el momento en que ocurrió el evento según los datos del conteo del encoder enclavado del hardware y lo almacena en la variable de registro de posición asociada en la estructura de datos del eje. Además, se establece simultáneamente el bit Event (PC) de la instrucción, al igual que el bit Registration Event Status en la estructura de datos del eje. Si se selecciona Windowed Registration, sólo se aceptan los eventos de registro cuya posición de registro calculada esté dentro de la ventana de posición máx. y mín. Si la posición de registro está fuera de esta ventana, la verificación del evento de registro se vuelve a activar automáticamente.

Operandos: *Lógica de escalera de relés*



Operando	Tipo	Formato	Descripción
AXIS	AXIS_FEEDBACK AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje en el cual se realizará la operación.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura utilizada para obtener acceso a los parámetros de estado de la instrucción.
Trigger condition	BOOLEANO	inmediato	Define la transición del registro de entrada que define el evento de registro. Puede seleccionar: 0 = activar en el flanco positivo 1 = activar en el flanco negativo
Registro en ventana	BOOLEANO	inmediato	Establece (1) si el registro debe estar dentro de los límites, es decir, si la posición de registro calculada está dentro de los límites de posición mínimo y máximo para ser aceptado como un evento de registro válido. Puede seleccionar: 0 = inhabilitado 1 = habilitado
Posición mínima	REAL	inmediato o tag	Utilizado cuando Windowed Registration está habilitado. La posición de registro debe ser mayor que el límite mínimo de posición antes de que se acepte el evento de registro.
Posición máxima	REAL	inmediato o tag	Utilizado cuando Windowed Registration está habilitado. La posición de registro debe ser menor que el límite máximo de la posición antes de que se acepte el evento del registro.
Input Number	UINT32	1 ó 2	Especifica el registro de entrada que se debe seleccionar. 1 = Registro 1 Posición 2 = Registro 2 Posición

```
MAR(Axis,MotionControl,
TriggerCondition,
WindowedRegistration,
MinimumPosition,
MaximumPosition,
InputNumber);
```

Texto estructurado

Los operandos son iguales a los de la instrucción MAR de lógica de escalera de relés.

En el caso de los operandos para los que usted debe seleccionar entre opciones disponibles, introduzca su selección como:

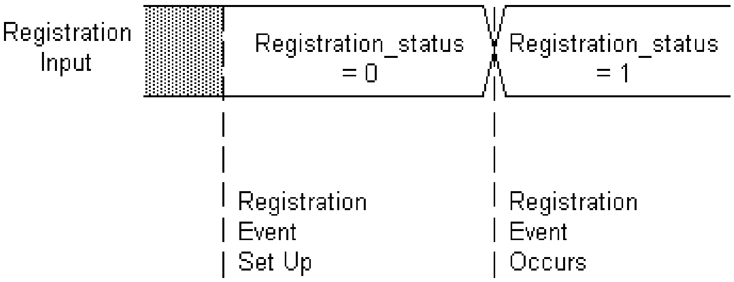
Este operando	**Tiene estas opciones que usted...**	
	introduce como texto	**o introduce como número**
TriggerCondition	positive_edge	0
	negative_edge	1
WindowedRegistration	inhabilitado	0
	habilitado	1

Estructura MOTION_INSTRUCTION

Mnemónico	**Descripción**
Bit .EN (Habilitar) 31	Se establece cuando el renglón hace una transición de falso a verdadero y permanece establecido hasta que la transacción de mensaje servo se completa y el renglón pasa a falso.
Bit .DN (Listo) 29	Se establece cuando la verificación del registro de evento del eje se ha activado correctamente.
Bit .ER (Error) 28	Se establece para indicar que la instrucción ha detectado un error, como si se hubiese especificado un eje no configurado.
Bit .IP (En proceso) 26	Se establece durante una transición positiva de renglón y se borra luego de que se ha producido un evento de control, o ha sido reemplazado por otro comando Motion Arm Reg, o interrumpido por un comando Motion Disarm Reg.
Bit .PC (Proceso completo) 27	Se establece cuando se produce un evento de registro.

Descripción: La instrucción MAR establece un evento de registro para almacenar las posiciones reales del eje físico especificado en el flanco especificado de la entrada de registro de alta velocidad dedicada y seleccionada para ese eje.

Cuando se ejecuta una instrucción MAR, el bit RegEventStatus se establece en 0 (FALSO) y la entrada de registro seleccionada para el eje especificado es monitoreada por el módulo de movimiento hasta que se produce una transición de entrada de registro del tipo seleccionado (el *evento de registro*). Cuando se produce el evento de registro, el bit RegEventStatus para el eje se establece en 1 (VERDADERO) y la posición real del eje se almacena en la variable de posición de registro correspondiente a la entrada de registro (por ejemplo, Registro 1 Posición 1 o Registro 2 Posición 2).

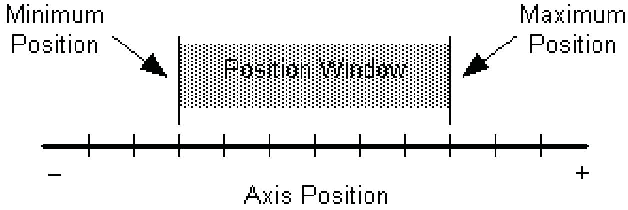


Registro

Es posible activar múltiples eventos de registro pueden estar activos en cualquier momento para un eje dado, pero sólo uno por entrada de registro puede estar activo. Cada evento se monitorea de manera independiente y se lo puede examinar utilizando el bit RegEventStatus apropiado.

Registro en ventana

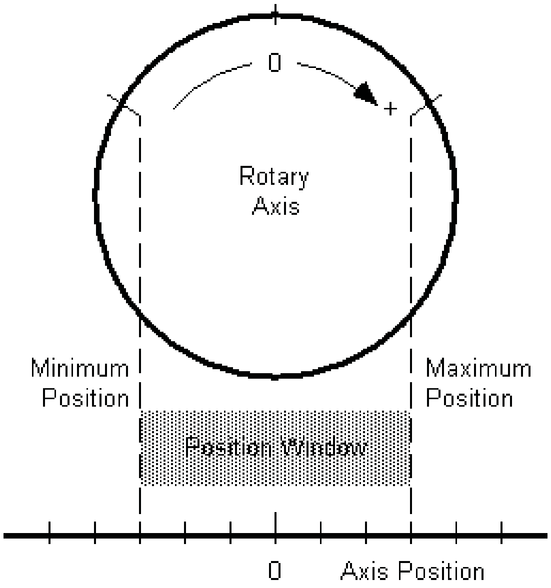
Cuando se marca la casilla de verificación Windowed Reg, el estado de activación seleccionado sólo da como resultado un evento de registro cuando el eje está dentro de la ventana definida por las posiciones mínima y máxima como se muestra a continuación.



Registro en ventana

Introduzca los valores o las variables de los tag para las posiciones absolutas deseadas que definen la posición de la ventana dentro de la cual el estado de activación seleccionado de la entrada de registro es válido. El registro en ventana es útil ya que provee un mecanismo para ignorar las transiciones falsas o aleatorias del sensor de registros, lo que mejora la inmunidad al ruido de las entradas de registro de alta velocidad.

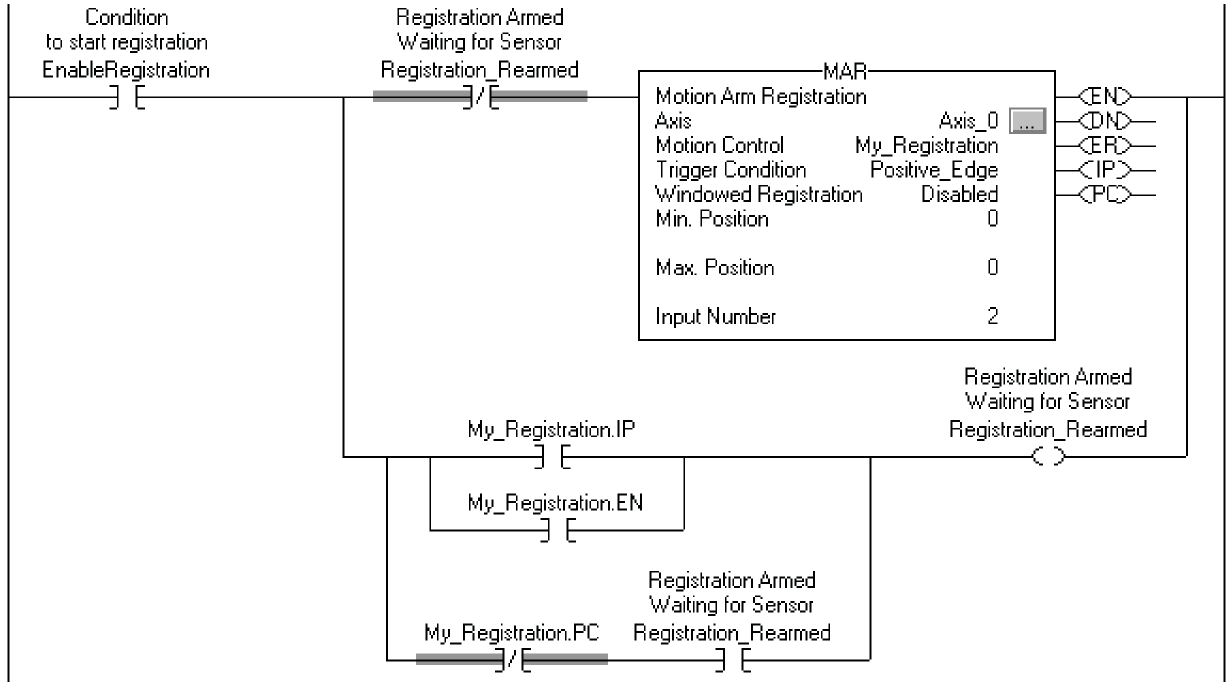
Para los ejes lineales, los valores pueden ser positivos, negativos, o una combinación. No obstante, el valor de la posición mínima debe ser menor que el valor de la posición máxima para que se produzca el evento de registro. Para los ejes giratorios, ambos valores deben ser menores al valor de desbobinado establecido en el menú de configuración de la máquina de control de movimiento. El valor de la posición mínima puede ser mayor que el valor de la posición máxima para las ventanas de registro que interceptan el punto de desbobinado del eje, como se muestra a continuación.



Posición de ventana para eje giratorio

Reactivar una instrucción MAR

Si su aplicación requiere de una detección rápida y continua de un sensor de registro, le recomendamos que utilice la siguiente lógica:



Motion Arm Registration
Axis Axis_0
Motion Control My_Registration
Trigger Condition Positive_Edge
Windowed Registration Disabled
Min. Position 0
Max. Position 0
Input Number 2

Lógica de escalera para la detección continua de registros

Para reactivar la instrucción MAR, el renglón debe cambiar de falso a verdadero. La velocidad con la que funciona la lógica depende de lo siguiente:

- tiempo de escán del programa
- velocidad de actualización del curso de la tarea de movimiento

IMPORTANTE En conexiones de E/S grandes, los valores forzados pueden disminuir la velocidad en la cual el controlador procesa las posiciones de registro repetitivas.

Para ejecutar correctamente una instrucción MAR, el eje objeto debe estar configurado como un servoeje o de sólo retroalimentación. De lo contrario, la instrucción dará error.

IMPORTANTE La ejecución de la instrucción MAR puede necesitar la ejecución de múltiples escanes porque requiere la transmisión de un mensaje al módulo de movimiento. El bit Listo (.DN) no se establecerá inmediatamente, sino sólo cuando este mensaje haya sido correctamente transmitido.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético:

no afectados

Condiciones de fallo:

ninguno

Códigos de error:

Consulte Códigos de error (ERR) para las instrucciones de control de movimiento.

Códigos de error extendidos:

Los códigos de error extendidos proporcionan información adicional específica de instrucción para los códigos de error que son genéricos a muchas instrucciones. Los siguientes códigos de error extendidos ayudan a localizar el problema cuando la instrucción MAR recibe un mensaje de error Fallo de mensajes al servomódulo (12).

Código de error asociado (decimal)	Código de error extendido (decimal)	Significado
SERVO_MESSAGE_FAILURE (12)	Sin Recurso (2)	No hay suficientes recursos de memoria para completar la solicitud. (SERCOS)
SERVO_MESSAGE_FAILURE (12)	Valor no válido (3)	La entrada de registro provista está fuera de rango.
SERVO_MESSAGE_FAILURE (12)	Dispositivo en estado equivocado (16).	Redefinir posición, Inicio y Registro 2 se excluyen mutuamente. (SERCOS)

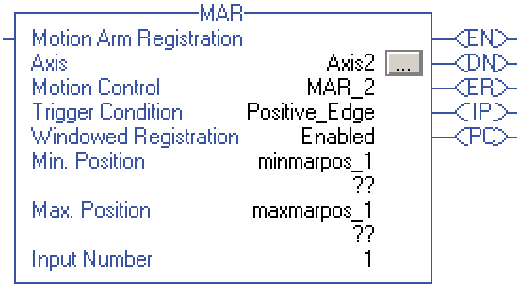
Los códigos de error extendidos para el código de error Parámetro fuera de rango (13) funcionan de un modo un poco diferente. En lugar de tener una enumeración estándar, el número que aparece para el código de error extendido hace referencia al número de operando tal y como aparecen en la plantilla de arriba hacia abajo, con el primer operando como cero. Por lo tanto, para la instrucción MAR, un código de error extendido de 4 haría referencia al valor de posición mínima. Entonces, debería verificar su valor con el rango aceptado de valores para la instrucción.

Bits de estado: *MAR cambia a bits de estado*

Nombre del bit	Estado	Significado
RegEventArmedStatus	verdadera	El eje está buscando un evento de registro.
RegEventStatus	falsa	Se borra el evento de registro anterior.

Ejemplo: Cuando las condiciones de entrada son verdaderas, el controlador activa la verificación de evento de registro del servomódulo para *eje_0*.

Lógica de escalera de relés



MAR
Motion Arm Registration
Axis Axis2
Motion Control MAR_2
Trigger Condition Positive_Edge
Windowed Registration Enabled
Min. Position minmarpos_1 ??
Max. Position maxmarpos_1 ??
Input Number 1

Ejemplo de lógica de escalera de MAR

Texto estructurado

MAR(Axis2,MAR_2,positive_edge,enabled,minmarpos_1, maxmarpos_1,1;

Motion Disarm Registration (MDR)

Use la instrucción MDR para desactivar la verificación del evento de entrada del módulo de movimiento para el eje especificado. Esta instrucción tiene el efecto de borrar los bits RegEventStatus y RegArmedEventStatus. El bit En Proceso de la instrucción de control MAR, si hubiere, se borra como resultado de la ejecución de la instrucción MDR.

Operandos: *Lógica de escalera de relés*

Operando	Tipo	Formato	Descripción
AXIS	AXIS_FEEDBACK AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje en el cual se realizará la operación.
Control de movimiento	MOTION_ INSTRUCTION	tag	Estructura utilizada para obtener acceso a los parámetros de estado de la instrucción.
Input Number	UINT32	1 ó 2	Especifica la entrada de registro que se debe seleccionar. 1 = Registro 1 Posición 2 = Registro 2 Posición

Texto estructurado

```
MDR(Axis,MotionControl,
InputNumber);
```

Los operandos son iguales a los de la instrucción MDR de lógica de escalera de relés.

Estructura MOTION_INSTRUCTION

Mnemónico	Descripción
Bit .EN (Habilitar) 31	Se establece cuando el renglón hace una transición de falso a verdadero y permanece establecido hasta que la transacción de mensaje servo se completa y el renglón pasa a falso.
Bit .DN (Listo) 29	Se establece cuando la verificación del evento de control del eje se ha desactivado correctamente.
Bit .ER (Error) 28	Se establece para indicar que la instrucción ha detectado un error, como si se hubiese especificado un eje no configurado.

Descripción: La instrucción MDR cancela la verificación de evento de registro establecida por una instrucción MAR anterior. Sólo la verificación de registro asociado con la entrada de registro especificada está inhabilitada.

Si el eje objeto no aparece en la lista de ejes disponibles, el eje no ha sido configurado para la operación. Utilice el editor de tags para crear y configurar un nuevo eje.

Para ejecutar correctamente una instrucción MDR, el eje objeto debe estar configurado como un servoeje o de sólo retroalimentación. De lo contrario, la instrucción dará error.

IMPORTANTE La ejecución de la instrucción MDR puede necesitar múltiples escanes porque requiere la transmisión de un mensaje al módulo de movimiento. El bit Listo (.DN) no se establecerá inmediatamente, sino sólo cuando este mensaje haya sido correctamente transmitido.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Consulte Códigos de error (ERR) para las instrucciones de control de movimiento.

Códigos de error extendidos: Los códigos de error extendidos proporcionan información adicional específica de la instrucción para los códigos de error que son genéricos a muchas instrucciones. Los siguientes códigos de error extendidos ayudan a localizar el problema cuando la instrucción MDR recibe un mensaje de error Fallo de mensajes al servomódulo (12).

Código de error asociado (decimal)	Código de error extendido (decimal)	Significado
SERVO_MESSAGE_FAILURE (12)	Valor no válido (3)	La entrada de registro provista está fuera de rango.

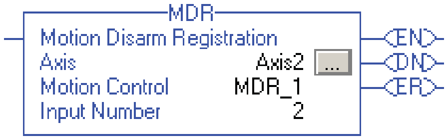
Los códigos de error extendidos para el código de error Parámetro fuera de rango (13) funcionan de un modo un poco diferente. En lugar de tener una enumeración estándar, el número que aparece para el código de error extendido hace referencia al número de operando tal y como aparece en la plantilla de arriba hacia abajo, con el primer operando como cero. Por lo tanto, para la instrucción MDR, un código de error extendido de 2 haría referencia al valor del operando Input Number. Entonces, debería verificar su valor con el rango de valores aceptado para la instrucción.

Bits de estado: *MDR cambia a bits de estado*

Nombre del bit	Estado	Significado
RegEventArmedStatus	falsa	El eje no está buscando un evento de registro.
RegEventStatus	falsa	Se borra el evento de registro anterior.

Ejemplo: Cuando las condiciones de entrada son verdaderas, el controlador desactiva la verificación de evento de registro para *eje_0*.

Lógica de escalera de relés



MDR
Motion Disarm Registration
Axis Axis2
Motion Control MDR_1
Input Number 2

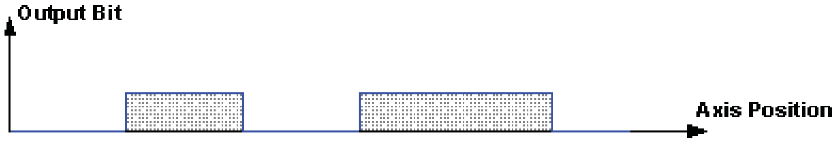
Ejemplo de lógica de escalera de MDR

Texto estructurado

MRD(Axis2,MDR_1,2);

Motion Arm Output Cam (MAOC)

La funcionalidad de la leva de salida del planificador de movimiento proporciona el establecimiento y restablecimiento de bits de salida basados en la posición de un eje.



Funcionalidad del planificador de movimiento

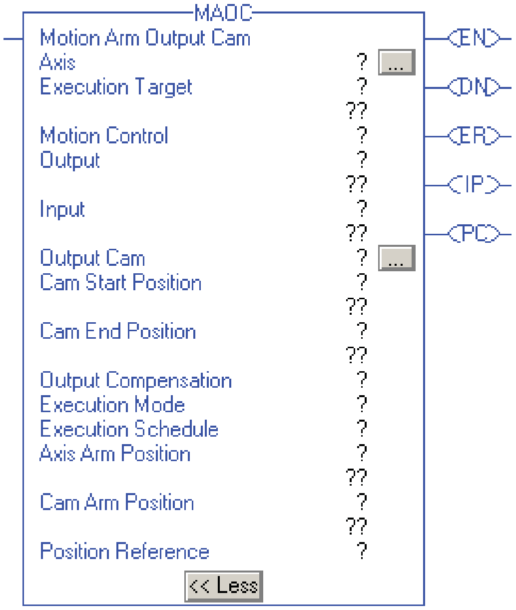
Internamente, los objetos de la leva de salida manipulan la funcionalidad de la leva de salida del planificador de movimiento. Cada objeto de la leva de salida es responsable de una salida, que consiste en 32 bits de salida. Cada bit individual de salida se puede programar por separado con un perfil de levas de salida, y puede ser compensado por el offset de posición y el tiempo de retardo.

La instrucción MAOC inicia la activación de una leva de salida específica entre el eje designado y la salida. Una vez ejecutados, los bits de perfil de leva de salida especificados se sincronizan con el eje designado utilizando un perfil de levas de salida establecido por el Editor de levas de salida RSLogix 5000. Se puede ver esta relación como una de maestro/esclavo con el eje representando al maestro y el bit de salida representando al esclavo. Por consiguiente, la funcionalidad de la leva de salida está relacionada con la funcionalidad de la leva de posición, la cual proporciona una relación entre el eje maestro y el eje esclavo. Para sincronizar con precisión las levas de salida en los ejes designados, se especifica un cronograma de ejecución y las posiciones activación de los ejes asociados y de la leva. Cuando el recorrido del eje pasa por la posición de activación del eje en la dirección especificada por el parámetro Execution Schedule, la posición de la leva se bloquea para que la posición del eje arranque en el parámetro Cam Arm Position especificado. En este momento se activa la leva de salida y se establece el estado leva de salida activada. También se puede configurar la leva de salida mediante el parámetro Execution Schedule para ejecutar de manera inmediata o pendiente de finalización una leva de salida que se está ejecutando actualmente. La leva de salida también puede ejecutarse Once, Continuously o Persistently especificando el modo de ejecución deseado. El comportamiento persistente permite que la leva de salida se desactive cuando la posición de la leva excede el rango de leva de salida, y que se reactive cuando la posición de la leva regrese dentro del rango. El rango de la leva de salida se define con los parámetros de entrada CamStartPosition y CamEndPosition. La selección de referencia de maestro permite derivar la entrada del eje desde la posición Real o de Comando del eje designado.

ATENCIÓN ⚠

Las levas de salida incrementan el potencial para exceder la velocidad de actualización aproximada. Esto puede provocar un mal comportamiento si el tiempo de ejecución de la tarea de movimiento excede el período de actualización aproximado del grupo configurado. La única manera de verificar si existe esta condición es monitorear el tiempo máximo de ejecución en la página Motion Group Properties.

Operandos: *Lógica de escalera de relés*



MAOC
Motion Arm Output Cam
Axis ?
Execution Target ?
??
Motion Control ?
Output ?
??
Input ?
??
Output Cam ?
Cam Start Position ?
??
Cam End Position ?
??
Output Compensation ?
Execution Mode ?
Execution Schedule ?
Axis Arm Position ?
??
Cam Arm Position ?
??
Position Reference ?

Operando	**Tipo**	**Formato**	**Descripción**
AXIS	AXIS_FEEDBACK AXIS_CONSUMED AXIS_VIRTUAL AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje que proporciona la entrada de posición a la leva de salida. La elipsis ejecuta el diálogo Axis Properties.
Objetivo de ejecución	UINT32	inmediato o tag	El objetivo de ejecución define la leva de salida específica del conjunto conectado al eje denominado. El comportamiento está determinado por lo siguiente: · 0...8 – Levas de salida ejecutadas en el controlador Logix. · 9...31 – Reservada para futuros usos.
Control de movimiento	MOTION_ INSTRUCTION	tag	Estructura utilizada para obtener acceso a los parámetros de estado de la instrucción.
Salida	DINT	tag	Un conjunto de 32 bits de salida que se establecen o restablecen basados en la leva de salida específica. Puede ser ya sea una ubicación de memoria o una salida física. Si se selecciona Pending como Execution Schedule, se ignorará la salida.
Entrada	DINT	tag	Un conjunto de 32 bits de salida que se pueden utilizar como bits Habilitar según la leva de salida especificada. Puede ser ya sea una ubicación de memoria o una entrada física. Si se selecciona Pending como Execution Schedule, se ignorará la entrada.
Output Cam	OUTPUT_CAM	tag de matriz	Una matriz de elementos OUTPUT_CAM. No es necesario ordenar los elementos y el tamaño de la matriz está determinado por el número de elementos de leva especificados. El tamaño de la matriz está limitado por la memoria disponible del controlador Logix.

Operando	**Tipo**	**Formato**	**Descripción**
Cam Start Position	SINT, INT, DINT o REAL	inmediato o tag	La posición de inicio de leva junto con la posición final de leva definen los límites izquierdo y derecho del rango de leva de salida.
Cam End Position	SINT, INT, DINT o REAL	inmediato o tag	La posición de fin de leva junto con la posición de inicio de leva definen los límites izquierdo y derecho del rango de leva de salida.
Output Compensation	OUTPUT_ COMPENSATION	tag de matriz	Es una matriz de 1 a 32 elementos OUTPUT_COMPENSATION. Los índices de la matriz corresponden a los números de bits de salida. El tamaño mínimo de una matriz está determinado por el bit de salida con compensación más alta.
Execution Mode	UINT32	inmediato	Existen tres (3) modos de ejecución posibles. El comportamiento está determinado por el modo seleccionado. Las opciones son: 0 = Once – Se desactiva la leva de salida y se establece el bit Proceso Completo de la instrucción de control de movimiento cuando la posición de la leva se mueva más allá de la posición de inicio o final de leva. 1 = Continuous – La leva de salida continúa en el lado opuesto del rango de leva de salida cuando la posición de la leva se mueve más allá de la posición de inicio de leva o final de leva. 2 = Persistent – La leva de salida se desactiva cuando la posición de la leva se mueve más allá de la posición de inicio de leva o de fin de leva. La leva de salida se reactiva cuando la posición de la leva vuelve al rango de la leva de salida.

Operando	**Tipo**	**Formato**	**Descripción**
Execution Schedule	UINT32	inmediato	Selecciona cuándo activar la leva de salida. Las opciones son: 0 = Immediate – La leva de salida se activa de una vez. 1 = Pending – La leva de salida se activa cuando la posición de leva de una leva de salida que se está ejecutando actualmente se mueve más allá de la posición de inicio o final de leva. Cuando se selecciona Pending se ignoran los siguientes parámetros Output, Input, Axis Arm Position y Reference. 2 = Forward only – La leva de salida se activa cuando el eje se aproxima o pasa a través de la posición de activación del eje especificado en dirección de avance. 3 = Reverse only – La leva de salida se activa cuando el eje se aproxima o pasa a través de la posición de activación del eje especificado en dirección de retroceso. 4 = Bi-directional – La leva de salida se activa cuando el eje se aproxima o pasa a través de la posición activación del eje especificado en cualquier dirección.

Operando	**Tipo**	**Formato**	**Descripción**
Axis Arm Position	SINT, INT, DINT o REAL	inmediato o tag	Esto define la posición del eje donde se activa la leva de salida cuando Execution Schedule se establece en Sólo avance, Sólo retroceso, o Bidireccíonal y el eje se mueve hacia la dirección especificada. Si se selecciona Pendiente como Execution Schedule, se ignora la posición de activación del eje.
Cam Arm Position	SINT, INT, DINT o REAL	inmediato o tag	Esto define la posición de la leva asociada con la posición de activación del eje cuando la leva de salida está activada.
Referencia	UINT32	inmediato	Establece si la leva de salida está conectada a la posición de comando o a la posición real del eje. Si se selecciona Pendiente como Execution Schedule, se ignorará la referencia. 0 = Real – la posición actual del eje según lo medido por su encoder u otro dispositivo de retroalimentación. 1 = Comando – la posición comandada o deseada del eje maestro.

Texto estructurado

```
MAOC(Axis,ExecutionTarget,
MotionControl,Output,Input,
OutputCam,CamStartPosition,
CamEndPosition,
OutputCompensation,
ExecutionMode,
ExecutionSchedule,
AxisArmPosition,
CamArmPosition,Reference);
```

Los operandos son iguales a los de la instrucción MAOC de lógica de escalera de relés. En el caso de los operandos de matriz, no es necesario incluir el índice de matriz. Si no incluye el índice, la instrucción comienza con el primer elemento de la matriz ([0]).

En el caso de los operandos para los que usted debe seleccionar entre opciones disponibles, introduzca su selección como:

Este operando	**Tiene estas opciones que usted...**	
	introduce como texto	**o introduce como número**
ExecutionMode	once	0
	continuous	1
	persistent	2
ExecutionSchedule	immediate	0
	pending	1
	forwardonly	2
	reverseonly	3
	bidirectional	4
Referencia	real	0
	command	1

Instrucción MAOC

Una posición válida de activación de leva es cualquier posición, entre e incluidas, las posiciones de inicio y final de leva. Si la posición de activación de leva se establece a un valor igual (o muy cercano) a la posición de inicio o final de leva, la compensación puede colocar una posición de leva fuera del rango de la posición de inicio y final de leva. La compensación se ve afectada por los valores de compensación de salida especificados para el Offset de posición, Retardo de enclavamiento y Retardo de desenclavamiento, al igual que los valores internos de compensación aplicados en base a los Parámetros Reference y Output de la instrucción MAOC.

No se producen efectos secundarios si se configura la instrucción MAOC con un modo Ejecución de “Continuous” o “Persistent”, y no hay una instrucción MAOC pendiente cuando la leva de salida se activa y el eje se mueve.

Se pueden producir los siguientes efectos secundarios si se configura la instrucción MAOC con un modo de Once Only, y hay una instrucción MAOC pendiente cuando se activa la leva de salida y el eje se mueve.

- Es probable que una o más salidas nunca cambien su estado.
- Es probable que la instrucción MAOC se complete inmediatamente.

Un posible efecto secundario de que haya una instrucción MAOC pendiente cuando la leva de salida está activada y el eje se mueve, es que una o más salidas pueden empezar a ejecutarse según la configuración de la instrucción MAOC pendiente.

Estructura MOTION_INSTRUCTION

Mnemónico	Descripción
Bit .EN (Habilitar) 31	Se establece cuando el renglón hace una transición de falso a verdadero y permanece establecido hasta que el renglón pasa a falso.
Bit .DN (Listo) 29	Se establece cuando la leva de salida se ha iniciado correctamente.
Bit .ER (Error) 28	Se establece para indicar que la instrucción ha detectado un error, como si se hubiese especificado un eje no configurado.
Bit .IP (En proceso) 26	Se establece cuando se ha iniciado y borrado correctamente la leva de salida si es reemplazado por otro comando Motion Arm Output Cam, interrumpido por un comando Motion Disarm Output Cam, o la posición de la leva se mueve más allá del rango de leva de salida definido mientras el modo de ejecución está establecido en 'once'.
Bit .PC (Proceso Completo) 27	Se borra en la transición del renglón positiva y se establece en el modo de ejecución 'once' cuando la posición de la leva se mueve más allá del rango de la leva de salida.
.SEGMENT	Se establece en el índice de la matriz asociado con el error 36 (Leva de salida no válida) o con el error 37 (Compensación de salida no válida). Sólo se almacenan los primeros errores múltiples.

Descripción:

La instrucción de MAOC ejecuta un perfil de leva de salida establecido de manera manual, programática o mediante el editor de levas de salida RSLogix 5000. Internamente, los objetos de la leva de salida manipulan la funcionalidad de la leva del planificador de movimiento. Cada objeto de la leva de salida es responsable de una salida, que consiste en 32 bits de salida. Cada bit de salida individual se puede programar por separado. La funcionalidad actual de la leva de salida se ejecuta en el controlador Logix en cada periodo de actualización aproximado (actualmente configurable entre 1 y 32 ms).

Axis

El eje proporciona la posición de entrada para la leva de salida. El eje puede ser virtual, físico o consumido.

Objetivo de ejecución

El objetivo de ejecución define una leva de salida específica del conjunto que está conectado al eje especificado. Actualmente, sólo pueden especificarse ocho levas de salida.

Cómo especificar el perfil de levas de salida

Para ejecutar una instrucción MAOC, se debe especificar un tag calculado de matriz de datos de leva de salida. Los tags de matriz de levas de salida se pueden crear mediante el editor de tags RSLogix 5000 o mediante la instrucción MAOC usando el editor de levas de salida incorporado. Los datos definen las especificaciones para cada elemento de leva de salida. El número de elementos de leva de salida está limitado por la cantidad de memoria disponible. Se pueden definir cero o más levas para cada bit de salida. No hay restricción en cuanto a cómo se organizan estos elementos dentro de la matriz de leva de salida.

Consulte la descripción de la estructura OUTPUT_CAM para obtener más información sobre los tipos de datos y las unidades de programación.

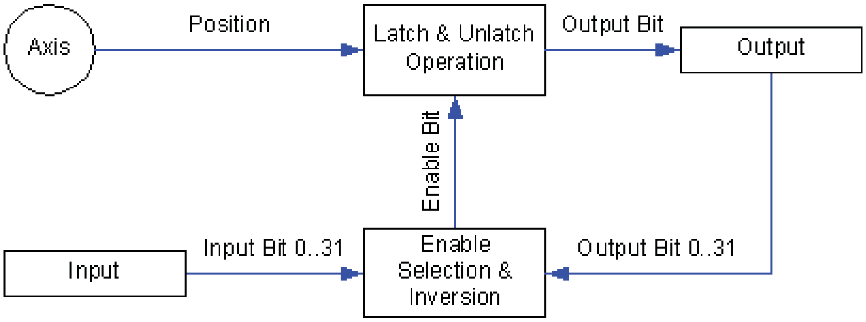
IMPORTANTE

Se produce una anomalía cuando las posiciones de la ventana CAM ON de salida se redefinen mientras la salida controlada por el elemento CAM de salida está activa. En algunos casos, el Planificador de movimiento puede no detectar un cruce off de la ventana y la salida controlada por el elemento CAM de salida permanece ON.

Esta cuestión se aplica a cualquier punto de salida o salida virtual controlada por una instrucción MAOC.

Además, le recomendamos que sólo cambie su configuración cuando el elemento CAM no esté activo.

El siguiente diagrama muestra las relaciones entre el eje, la entrada y la salida que definidas por el elemento de la leva de salida.



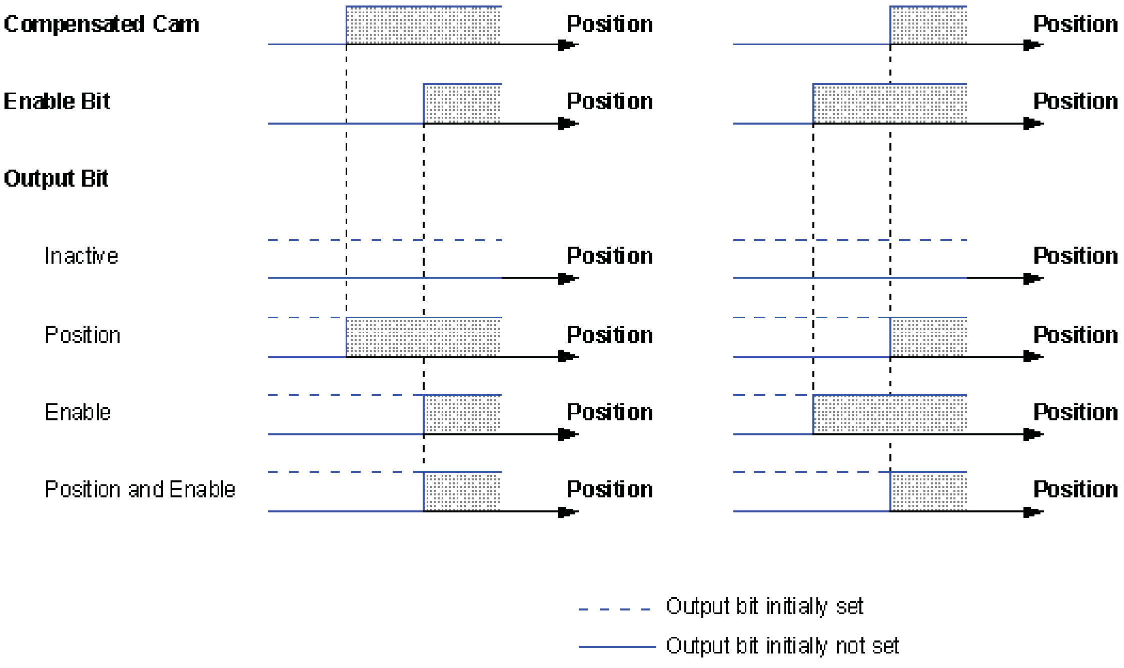
Relaciones de los elementos de levas de salida

Tipo de enclavamiento

Según el Tipo de enclavamiento seleccionado, el bit de salida correspondiente se establece según la siguiente tabla.

Tipo de enclavamiento	Comportamiento
Inactive	El bit de salida no se modifica.
Position	El bit de salida se establece cuando el eje introduce el rango de leva compensada.
Enable	El bit de salida se establece cuando el bit Habilitar se vuelve activo.
Position and Enable	El bit de salida se establece cuando el eje introduce el rango de leva compensada y el bit Habilitar se vuelve activo.

El siguiente diagrama muestra el efecto que tiene el tipo de enclavamiento seleccionado en el bit de salida para distintas combinaciones de leva compensada y bit Habilitar como función de posición.



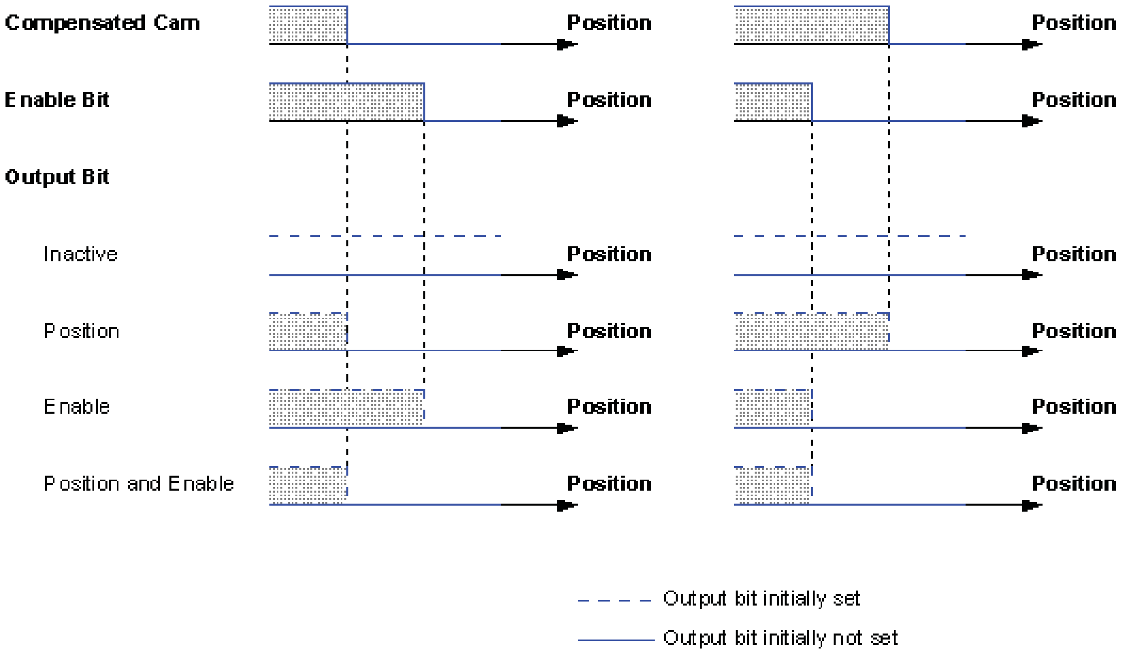
Posición de enclavamiento

Tipo de desenclavamiento

Según el Tipo de desenclavamiento, el bit de salida correspondiente se restablece según la siguiente tabla.

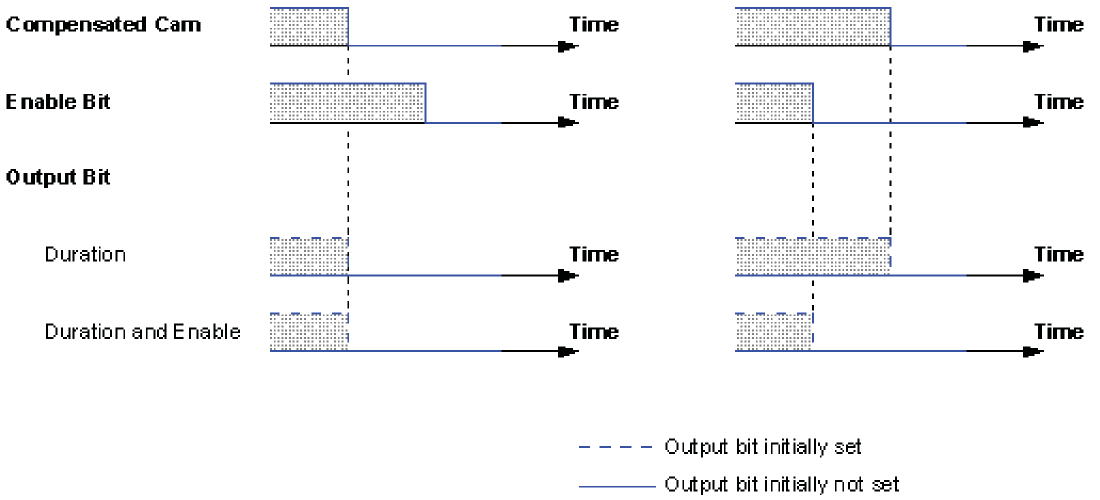
Tipo de desenclavamiento	Comportamiento
Inactive	El bit de salida no se modifica.
Position	El bit de salida se restablece cuando el eje se sale el rango de leva compensada.
Duración	El bit de salida se restablece cuando se termina la duración.
Enable	El bit de salida se restablece cuando el bit Habilitar se vuelve inactivo.
Position and Enable	El bit de salida se restablece cuando el eje se sale el rango de leva compensada o el bit Habilitar se vuelve inactivo.
Duration and Enable	El bit de salida se restablece cuando se termina la duración o el bit Habilitar se vuelve inactivo.

El siguiente diagrama muestra el efecto que tiene el tipo de desenclavamiento seleccionado en el bit de salida para distintas combinaciones de leva compensada y bit Habilitar como función de posición.



Desenclavamiento como función de posición

y como función de tiempo.



Desenclavamiento como función de tiempo

Posiciones de leva izquierda y derecha

Las posiciones de leva izquierda y derecha definen el rango del elemento de la leva de salida. Si el tipo de enclavamiento o desenclavamiento se establece en “Position” o “Position and Enable” con el bit Habilitar activo, las posiciones de leva derecha e izquierda especifican las posiciones de enclavamiento o desenclavamiento del bit de salida.

Duración

Si el tipo de desenclavamiento se establece en “Duration” o “Duration and Enable” con el bit Habilitar activo, la duración de la leva especifica el tiempo entre el enclavamiento y el desenclavamiento del bit de salida.

Tipo de habilitación

Según el tipo de habilitación seleccionado, el bit Habilitar es un elemento de la entrada, la entrada invertida, la salida o la salida invertida.

Controles de la matriz de levas de salida

Si selecciona un bit de salida menor que 0 o mayor que 31, no se considera el elemento de la leva de salida y se advierte al usuario con un error de instrucción "Leva de salida no válida".

Si selecciona un tipo de enclavamiento menor que 0 o mayor que 3, se usa un valor de "Inactive" y se advierte al usuario con un error de instrucción "Leva de salida no válida".

Si selecciona un tipo de desenclavamiento menor que 0 o mayor que 5, se usa un valor de "Inactive" y se advierte al usuario con un error de instrucción "Leva de salida no válida".

Si selecciona una posición de leva izquierda mayor o igual a la posición de la leva derecha y el tipo de enclavamiento o desenclavamiento se establece en "Position" o "Position and Enable", el elemento de la leva de salida no es considerado y se advierte al usuario con el error de instrucción "Leva de salida no válida".

Si selecciona una posición de leva izquierda menor que la posición de inicio de la leva y el tipo de enclavamiento está establecido en "Position" o "Position and Enable" se usa la posición de inicio de la leva y se advierte al usuario con el error de instrucción "Leva de salida no válida".

Si selecciona una posición de leva derecha mayor que la posición final de la leva y el tipo de desenclavamiento está establecido en "Position" o "Position and Enable", se usa la posición final de la leva y se advierte al usuario con el error de instrucción "Leva de salida no válida".

Si selecciona una duración menor o igual a 0 y el tipo de desenclavamiento está establecido en "Duration" o "Duration and Enable", no se considera el elemento de la leva de salida y se advierte al usuario con el error de instrucción "Leva de salida no válida".

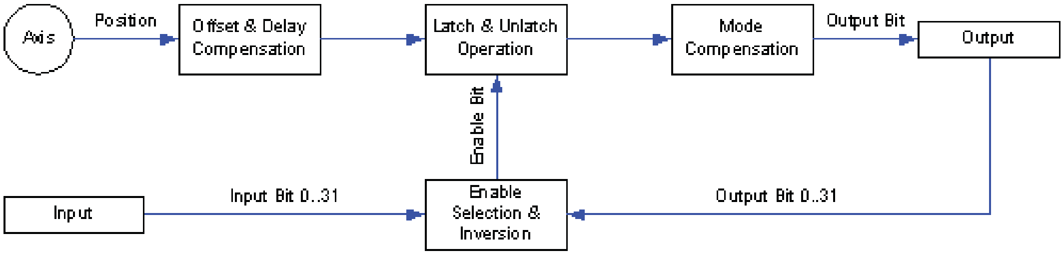
Si selecciona un tipo de habilitación menor que 0 o mayor que 3 y el tipo de enclavamiento está establecido en "Enable" o "Position and Enable", el elemento de la leva de salida no es considerado y se advierte al usuario con un error de instrucción "Leva de salida no válida".

Si selecciona un bit Habilitar menor que 0 o mayor que 31 y el tipo de enclavamiento está establecido en "Enable", "Position and Enable" o "Duration and Enable", el elemento de la leva de salida no es considerado y se advierte al usuario con un error de instrucción "Leva de salida no válida".

Cómo especificar la compensación de salida

Un tag de matriz de datos de compensación de salida se puede especificar mediante el editor de tags RSLogix 5000. El tipo de datos define las especificaciones para cada bit de salida al especificar las características de cada accionador. Los índices de la matriz corresponden a los números de bits de salida. El número del bit de salida de mayor compensación define el tamaño mínimo de esta matriz. Los cambios en la compensación de salida tienen efecto inmediatamente.

El siguiente diagrama muestra el efecto que tiene la compensación de salida en las relaciones entre el eje, la entrada y la salida.

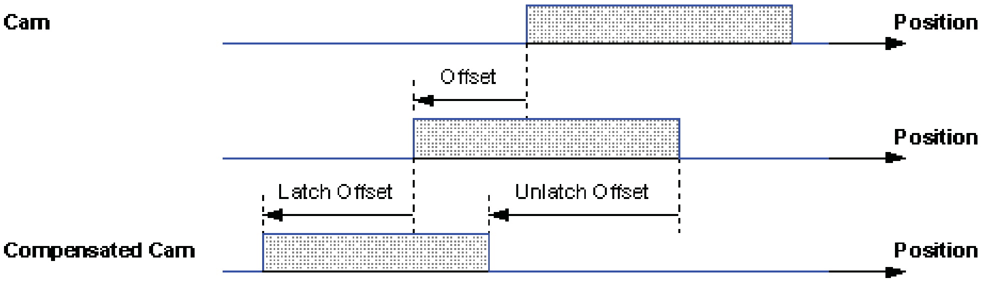


Compensación de salida

Consulte la descripción de la estructura OUTPUT_COMPENSATION para obtener más información sobre los tipos de datos y las unidades de programación.

Compensación de retardo y de offset

El offset proporciona una posición de compensación, mientras que el retardo de enclavamiento y desenclavamiento proporciona una compensación de tiempo de retardo para la operación de enclavamiento y desenclavamiento. El siguiente diagrama muestra el efecto de los valores de compensación en un elemento de leva de salida.



Compensación de retardo y de offset

El rango de leva está definido por las posiciones de leva derecha e izquierda del elemento de leva de salida. El rango de leva compensado está definido por el rango de leva, offset, y los offsets de enclavamiento y desenclavamiento. Los offsets de enclavamiento o desenclavamiento están definidos por la velocidad actual v:

Offset de enclavamiento = v * Retardo de enclavamiento

Offset de desenclavamiento = v * Retardo de desenclavamiento

El offset de compensación resultante en realidad puede ser mayor que la diferencia entre la posición de inicio de leva y final de leva.

La siguiente ecuación ilustra el efecto que tienen los valores de compensación en la duración de un elemento de leva de salida.

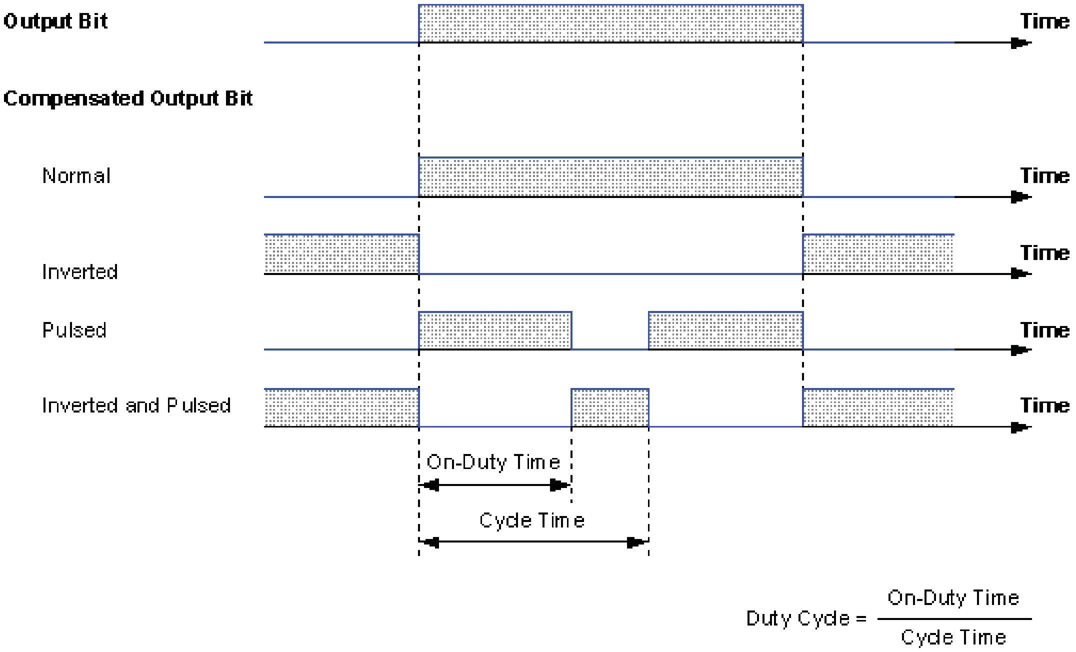
Duración compensada = Duración + Retardo de enclavamiento – Retardo de desenclavamiento

Modo Compensación

Según el modo seleccionado, el bit de salida compensado se establece según la siguiente tabla.

Mode	Comportamiento
Normal	El bit de salida se establece, cuando la salida de la operación de enclavamiento y desenclavamiento se vuelve activa. El bit de salida se restablece, cuando la salida de la operación de enclavamiento y desenclavamiento se vuelve inactiva.
Inverted	El bit de salida se establece, cuando la salida de la operación de enclavamiento y desenclavamiento se vuelve inactiva. El bit de salida se restablece, cuando la salida de la operación de enclavamiento y desenclavamiento se vuelve activa.
Pulsed	El bit de salida se impulsa, cuando la salida de la operación de enclavamiento y desenclavamiento está activa. El estado en servicio del impulso corresponde al estado activo del bit de salida. El bit de salida se restablece, cuando la operación de enclavamiento y desenclavamiento se vuelve inactiva.
Inverted and Pulsed	El bit de salida se impulsa, cuando la salida de la operación de enclavamiento y desenclavamiento está activa. El estado en servicio del impulso corresponde al estado inactivo del bit de salida. El bit de salida se establece, cuando la salida de la operación de enclavamiento y desenclavamiento se vuelve inactiva.

El siguiente diagrama muestra el efecto del modo, tiempo de ciclo y ciclo de servicio en un bit de salida.



Modo Compensación

Controles de matrices de compensación de salida

Si selecciona una combinación de retardo que dé como resultado una leva compensada de menos del ancho mínimo, el ancho de la leva compensada se establece al mínimo.

Si selecciona un tipo de modo menor que 0 o mayor que 3, se considera un modo "Normal" y se advierte al usuario con una instrucción de error "Compensación de leva no válida".

Si selecciona un ciclo de servicio menor que 0 o mayor que 100 y el modo se establece en "Pulsed" o "Inverted and Pulsed", se considera un ciclo 0 ó 100 y se advierte al usuario con un error de instrucción "Compensación de salida no válida".

Si selecciona un tipo de ciclo menor o igual a 0 y el modo está establecido en "Pulsed" o "Inverted and Pulsed", no se impulsa el bit de salida y se advierte al usuario con un error de instrucción "Compensación de salida no válida".

Salida

La salida es el conjunto de 32 bits de salida que se pueden establecer y restablecer según la leva de salida especificada. La salida puede ser una ubicación de memoria o una salida física (por ejemplo, Local.0.O.Data).

Entrada

La entrada es un conjunto de 32 bits de entrada que se pueden usar como bits Habilitar según la leva de salida especificada. La entrada puede ser una ubicación de memoria o una entrada física (por ejemplo, Local.0.I.Data).

Posiciones de inicio y final de leva

Las posiciones de inicio y final de leva definen los límites izquierdo y derecho del rango de leva de salida. Cuando la posición de la leva se mueve más allá de la posición de inicio o final de leva, el comportamiento de la leva de salida se define por el modo y el cronograma de ejecución. Los cambios en la posición de inicio de leva y final de leva no hacen efecto hasta que se completa la ejecución de la instrucción MAOC actual.

Modo de ejecución

Según el modo de ejecución seleccionado, el comportamiento de la leva de salida puede ser diferente, cuando la posición de inicio o final de leva se mueve más allá de la posición de inicio o fin de leva.

Modo de ejecución	Comportamiento
Once	Cuando la posición de la leva se mueva más allá de la posición de inicio o final de leva, se desactiva la leva de salida y se establece el bit Proceso completo de la instrucción de control de movimiento.
Persistent	Cuando la posición de la leva se mueve más allá de la posición de inicio o final de leva, la leva de salida se desactiva. Sin embargo, cuando la posición de la leva vuelve al rango de la leva de salida, se reactiva la leva de salida.
Continuous	Cuando la posición de la leva se mueve más allá de la posición de inicio de leva o final de leva, la leva de salida continua en el lado opuesto del rango de leva de salida.

Cronograma de ejecución

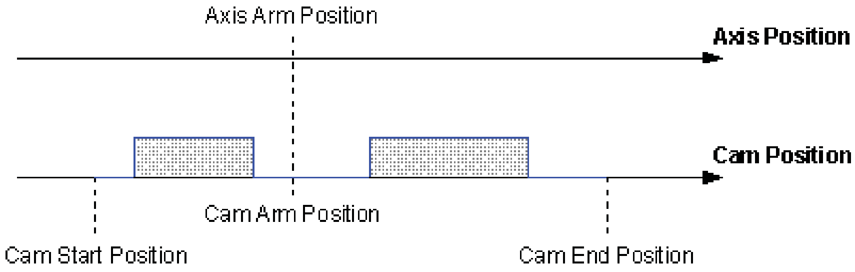
Según el parámetro de cronograma de ejecución seleccionado, se activa la leva de salida según la siguiente tabla.

Cronograma de ejecución	Comportamiento
Inmediate	La leva de salida se activa inmediatamente.
Pending	Se activa la leva de salida, cuando la posición de leva de una leva de salida activada se mueve más allá de su posición de inicio de leva o final de leva.

Cronograma de ejecución	Comportamiento
Forward Only	La leva de salida se activa cuando el eje se aproxima o pasa a través de la posición de activación del eje especificado en dirección de avance.
Reverse only	Se activa la leva de salida cuando el eje se aproxima o pasa a través de la posición de activación del eje especificado en dirección de retroceso.
Bi-Directional	Se activa la leva de salida cuando el eje se aproxima o pasa a través de la posición de activación del eje especificado en dirección de avance o retroceso.

Posiciones de activación de eje y activación de leva

La posición de activación del eje define dónde se activa la leva de salida, si se establece Execution Schedule a sólo avance o sólo retroceso, o bi-direccional y el eje se mueve en la dirección especificada. La posición de activación de leva define la posición de la leva que está asociada con la posición de activación del eje, cuando la leva de salida está activada. Los cambios en la posición de activación del eje o activación de la leva sólo hacen efecto luego de la ejecución de una instrucción MAOC.



Posiciones de activación de eje y activación de leva

Referencia

Según la referencia seleccionada, la leva de salida está conectada ya sea a la posición de comando o a la posición real del eje.

IMPORTANTE	La ejecución de la instrucción MAOC se completa con un único escán, de esta manera se establecen inmediatamente los bits Listo (.DN) y En proceso .IP. El bit En Proceso .IP permanece establecido hasta que la posición de la leva se mueve más allá de la posición de inicio o final de leva en modo de ejecución "Once", es reemplazado por otra instrucción MAOC o desactivado por la instrucción MDOC. El bit Proceso Completo se borra inmediatamente cuando se ejecuta la instrucción MAOC y se establece cuando la posición de la leva se mueve más allá de la posición de inicio o final de leva en el modo de ejecución "Once".

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: *Códigos de error (.ERR) de MAOC*

Mensaje de error	Código	Descripción
Colisión de ejecuciones	3	Se intentó una ejecución con otra leva de salida actualmente en proceso.
Error del estado de interrupción	7	Se intentó una ejecución con el eje en estado de interrupción.
Eje no configurado	11	El valor pasado del eje hace referencia a un eje no configurado, lo que significa que no ha sido asignado a un eje a un canal de módulo de movimiento físico.
Valor fuera de rango	13	Se intentó una ejecución con un parámetro de entrada que estaba fuera de rango. 1. Posición de inicio de leva ≥ posición final de leva. 2. Posición de activación de leva fuera del rango de la leva de salida. Vea la sección Error extendido para obtener más información sobre la causa del error.
Vuelta a la posición inicial en error de proceso	16	Se intentó una ejecución mientras el proceso de vuelta a la posición inicial está en progreso.
Grupo de ejes no sincronizado	19	Se intentó una ejecución en un eje cuyo grupo de ejes asociados no está actualmente sincronizado.
Eje en estado de fallo	20	Se intentó una ejecución en un eje que se encuentra en estado de fallo.
Grupo en estado de fallo	21	Se intentó una ejecución en un eje, que está en grupo, que está en estado de fallo.
Cambio dinámico no válido	23	Se intentó ejecutar un cambio de dinámica no válido como una incorporación en una curva en S, un cambio de perfil desde trampa en la curva en S rápidamente, un cambio de curva en S a una velocidad que no es cero o al cambiar la aceleración de una curva en S.
Operación no válida de modo controlador	24	Se intentó una ejecución cuando el procesador estaba en modo de prueba.
Objetivo de ejecución no válido	35	Se intentó una ejecución con una leva de salida especificada no compatible con el controlador Logix.

Mensaje de error	**Código**	**Descripción**
Leva de salida no válida	36	Se intentó una ejecución con una matriz de leva de salida que contiene al menos un miembro fuera de rango: 1. OutputBit menor que 0 o mayor que 31. 2. Valor LatchType no válido. 3. Valor UnlatchType no válido. 4. Izquierda ≥ Derecha mientras LatchType se establece en ‘position’ o ‘position and enable’. 5. Izquierda < Posición de inicio de leva mientras LatchType se establece en ‘position’ o ‘position and enable’. 6. Derecha > Posición final de leva mientras UnlatchType se establece en ‘position’ o ‘position and enable’. 7. Duración ≤ 0 mientras UnlatchType se establece en ‘duration’ o ‘duration and enable’. 8. Valor EnableType no válido mientras LatchType o UnlatchType están establecidos en ‘enable’ o ‘duration and enable’. 9. Valor EnableBit no válido mientras LatchType o UnlatchType está establecido en ‘enable’ o ‘position and enable’ o ‘duration and enable’.
Compensación de salida no válida	37	Se intentó una ejecución con una matriz de leva de salida que contiene al menos un miembro fuera de rango: 1. Valor de modo no válido. 2. CycleTime ≤ 0 mientras Mode está establecido en ‘pulsed’ o ‘inverted and pulsed’. 3. DutyCycle menor que 0 o mayor que 100 y Mode está establecido en ‘pulsed’ o ‘inverted and pulsed’.
Operando de salida MAOC no válido	80	ExErr#1: Referencia de tag de datos no válida – El operando de salida no está señalando el elemento O.Data del tag de datos de salida del modulo. ExErr#2: Formato de las comunicaciones del módulo OB16IS no válido – Se ha cambiado el formato de las comunicaciones OB16IS de su Dato de salida programado por punto predeterminado.

Códigos de error extendidos:

Los códigos de error extendidos brindan información adicional específica de la instrucción para los códigos de error que son genéricos a muchas instrucciones. Los códigos de error extendidos para el código de error Parámetro fuera de rango (13) incluyen un número que hace referencia al número de operando tal y como aparecen en la plantilla de arriba hacia abajo, con el primer operando como cero. Por lo tanto, para la instrucción MAOC, un código de error extendido 4 haría referencia al valor del operando Output. Entonces, debería verificar su valor con el rango aceptado de valores para la instrucción.

Bits de estado:

Efectos de la instrucción MAOC en los bits de estado

Se pueden utilizar los bits de estado para determinar si se puede iniciar una instrucción MAOC. La instrucción MAOC afecta las siguientes palabras de estado en la estructura del eje de movimiento:

- OutputCamStatus
- OutputCamPendingStatus
- OutputCamLockStatus
- OutputCamTransitionStatus

Si se establece el programa de ejecución en Forward Only, Reverse Only o Bi-Directional, se puede iniciar una instrucción MAOC cuando se cumpla cualquiera de las siguientes condiciones:

- OutputCamStatus bit = FALSO

  o

- Bit OutputCamStatus = VERDADERO
  Bit OutputCamLockStatus = FALSO
  Bit OutputCamTransitionStatus = FALSO

Si Execution Schedule está establecido en Pending, se inicia la instrucción MAOC si se cumple cualquiera de las dos siguientes condiciones:

- Bit OutputCamStatus = FALSO

  o

- Bit OutputCamStatus = VERDADERO
  Bit OutputCamTransitionStatus = FALSO

Las condiciones de fallo del eje y del módulo desactivan las levas de salida

Cuando el controlador detecta uno de los siguientes fallos, desactiva las levas de salida:

- En el caso de Axis_Servo y Axis_Servo_Drive, fallo de pérdida de retroalimentación del eje
- En el caso de Axis_Servo y Axis_Servo_Drive, fallo de módulo
- En el caso de Axis_Consumed, fallo de eje físico

Esos fallos producen datos de retroalimentación no confiables.

Además, si hay un fallo de eje cuando se inicia una instrucción MAOC, la instrucción dará error.

Módulo de salida programada

Se designa el módulo de salida programada 1756-OB16IS para trabajar en combinación con la instrucción de movimiento MAOC, para proporcionar un control de salida de base de posición (también conocido como PLS). La instrucción MAOC en sí misma permite el control de salida de base de posición de cualquier eje de movimiento en ControlLogix como la posición de referencia y cualquier salida o booleano como la salida. La instrucción MAOC actualiza las salidas basadas en la posición del eje de movimiento en la velocidad de actualización aproximada del grupo de movimiento (generalmente 2 ms – 10 ms). Mientras que esto puede ser adecuado para algunas aplicaciones, es demasiado lento para muchas aplicaciones de alta velocidad que se encuentran generalmente en segmentos de conversión y envasado. El módulo 1756-OB16IS mejora el rendimiento al aceptar la capacidad de programar el tiempo de encendido/apagado de 8 de sus 16 salidas (salidas 0 – 7) en 100 μs incrementos. Las salidas se programan introduciendo los datos en uno o más de los 16 programas provistos por el almacenamiento de datos de la conexión de salida.

IMPORTANTE Cuando utilice el módulo 1756-OB16IS con la instrucción MAOC, asegúrese de utilizar el formato de comunicación predeterminado, es decir, Dato de salida programado por punto. Si cambia el formato de la comunicación cuando se utiliza el módulo con una instrucción MAOC, se puede producir un error.

Operación: Las salidas programadas tal como se definen aquí no deben confundirse con la implementación anterior de salidas programadas. La implementación anterior programa salidas por módulo y todos los puntos de salida están controlados por un sello de hora. Esta implementación programa salidas por punto y cada punto de salida individual está controlado por su propio sello de hora.

Los programas individuales se crean en el controlador, se almacenan en la tabla de imagen de salida para el módulo, y se envían al módulo de salida programada mediante backplane. El programa especifica un conteo de secuencia, el punto de salida que debe asociarse con el programa, el momento en el que el valor de salida debe aplicarse al punto de salida físico y el valor que se debe aplicar en el momento programado. El módulo de E/S recibe y almacena el programa. El sello de hora de CST de cada programa está monitoreado por el módulo. Cuando un programa ha expirado, es decir el tiempo actual, coincide con el sello de hora programado, entonces se aplica el valor de salida al bit de salida correspondiente. El hardware del temporizador en el ASIC se utiliza para optimizar el algoritmo de priorización. Este hardware también reduce el tiempo de espera y las oscilaciones. El estado de cada programa se reporta en la conexión eco de salida y se refleja en la imagen de entrada para el módulo.

La funcionalidad de salida programada se basa en el sello de tiempo de CST (Hora coordinada del sistema). Al menos un controlador en el chasis debe ser un maestro de tiempo de CST.

Las salidas no usadas se pueden usar como salidas normales y se aplican inmediatamente en vez de esperar a que expire el sello de tiempo de CST. Se envía una máscara al módulo para indicar qué salidas funcionan como salidas normales.

El módulo de salida programada acepta hasta 8 salidas que se pueden programar individualmente. Las salidas programadas deben estar entre 0 y 7. El 1756-OB16IS acepta hasta 16 programas con dos programas por salida. Las salidas que no están "programadas" se usan como puntos de salida normales. Se usa una máscara para indicar qué puntos están programados y qué puntos no están programados. Las oscilaciones y el tiempo de espera son menor que 100 microsegundos. Toda la información de priorización se hace a través de la instrucción MAOC.

Si el módulo de E/S recibe un nuevo programa indicado por un cambio en el conteo de secuencia antes de que el programa actual haya expirado, el programa actual se sobrescribe. Se puede utilizar este mecanismo para cancelar el programa activo actualmente. Los bits de estado devueltos en la conexión eco de salida se pueden usar para determinar el estado actual de cada programa y para activar las tareas de evento correspondientes.

Si el controlador envía un nuevo programa y el sello de hora de CST ya ha pasado, se impone la salida hasta que la hora CST haya dado la vuelta completamente. El módulo no verifica para determinar si hay un sello de tiempo de CTS expirado.

ADVERTENCIA ⚠

Si el tiempo entre los dos programas es menor que el intervalo mínimo del programa (por ejemplo, 100 μs), entonces se producen oscilaciones. Esto quiere decir que aunque las dos salidas estén programadas en horas diferentes (por ejemplo, hora 90 y hora 110), ambas se activan a la misma hora (por ejemplo, hora 90). (El intervalo mínimo programado no se debe establecer en más de 100 μs.)

Operación remota:

Las salidas programadas que utilizan el módulo 1756-OB16IS no funcionan con un chasis remoto.

Uso con la instrucción MAOC:

Cuando se utilizan con movimiento y la instrucción MAOC, los valores de la imagen de salida están controlados por el firmware del planificador de movimiento en el controlador. El planificador de movimiento activa los datos por enviar al módulo. Sin embargo, el programa normal/escán de tareas también activa los datos por enviar al módulo. El firmware siempre mantiene la integridad de los datos estableciendo el conteo de secuencia para una duración de programa dado.

La instrucción de la leva de salida procesa los eventos de levas para las salidas programadas con un período de actualización aproximado más corto que las salidas no programadas. Cuando se detecta un evento programado de encendido o apagado, se envía un programa al módulo de salida para encender/apagar la salida en el momento apropiado dentro del próximo período de actualización aproximado. La instrucción de la leva de salida divide el período de actualización aproximado en dieciséis franjas de tiempo.

Por ejemplo, un período de actualización aproximado de 2 milisegundos producirá dieciséis franjas de tiempo de 125 microsegundos. Los eventos de encendido/apagado de levas serán asignados a las franjas de tiempo según su posición dentro del período de actualización aproximado. Si ambos eventos de enclavamiento y desenclavamiento para un elemento de leva se asignan a la misma franja de tiempo, se cancelarán mutuamente. Esto implica que la anchura de impulso mínima de un elemento de leva es mayor que una franja de tiempo.

La anchura de impulso mínima de un elemento de leva debe ser mayor que la anchura de impulso mínima de 100 microsegundos de OB16IS, o 1/16 de la anchura de impulso mínima de la actualización aproximada, cualquiera que sea mayor.

IMPORTANTE El módulo de salida programada 1756-OB16IS se puede asociar sólo con un objetivo de ejecución/eje MAOC (1).

La instrucción MAOC detecta los eventos de enclavamiento y desenclavamiento en una actualización aproximada por adelantado y programa el evento para que ocurra dentro de la siguiente actualización aproximada. Esto se logra aplicando un retardo interno de actualización aproximada a cada posición de enclavamiento o desenclavamiento de salida programada. Cuando se detecta el evento de enclavamiento o desenclavamiento, se calcula el tiempo delta desde el inicio de la actualización aproximada hasta el evento, y se programa la salida para que ocurra a la hora coordinada del sistema (CST) correspondiente al próximo período de actualización aproximado. Para facilitar esto, la funcionalidad de la leva de salida tiene acceso a la CST captada cuando se produjo el período de actualización aproximada actual.

La instrucción MAOC puede programar bits de salida tanto programados como no programados para el 1756-OB16IS.

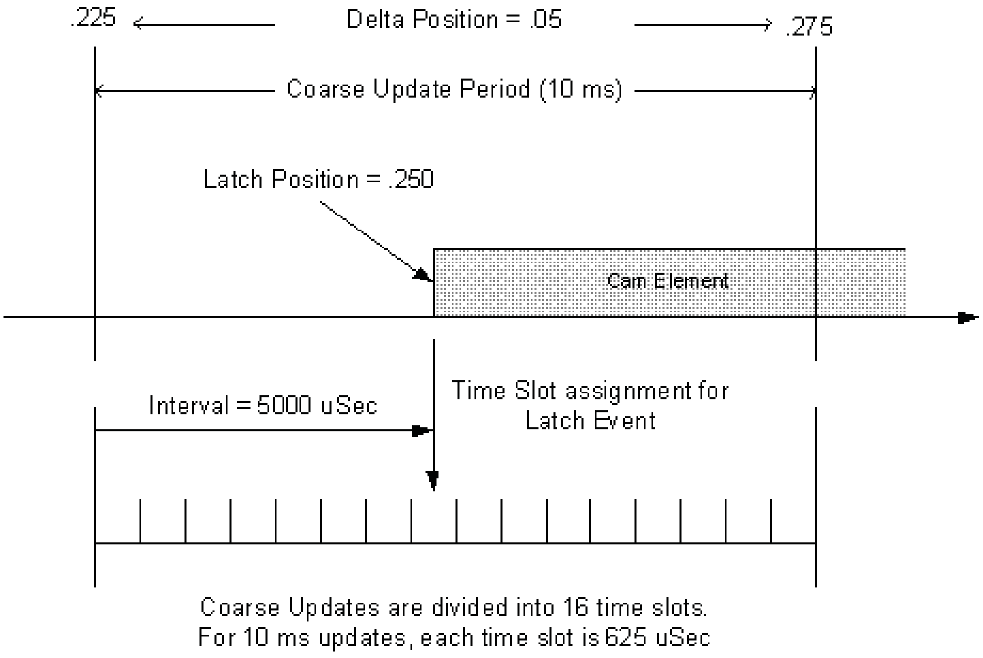
La instrucción MAOC asigna las ocho salidas programadas para uso exclusivo de la leva de salida del planificador de movimiento. La instrucción MAOC establece el campo de máscara en 0xff en cada período aproximado por si el usuario intenta cambiarlo. Esto implica que el usuario no puede afectar directamente los bits de salida 0 – 7, pero sí tiene la capacidad de modificar los bits de salida 8 – 15.

IMPORTANTE Se pueden forzar las salidas 0 – 7 forzando el bit de datos a 0 ó 1 y su bit correspondiente en ScheduleMask a 0. Para las salidas 8 – 15, sólo se debe forzar el bit de datos.

Debido al límite de dieciséis programas compatibles con el 1756-OB16IS, se aplican algunas restricciones al número de eventos que pueden ser procesados durante cada período de actualización aproximado.

Sólo ocho programas están disponibles durante cada actualización aproximada. Esto permite que se puedan realizar dos actualizaciones aproximadas consecutivas en la que cada actualización restringe ocho eventos de salida. Como un grupo de ocho programas está siendo procesado actualmente por el 1756-OB16IS, se puede configurar concurrentemente un segundo grupo de ocho programas para la próxima actualización aproximada.

El siguiente diagrama ilustra la relación entre el período de actualización aproximado, un evento de enclavamiento de leva y las franjas de tiempo.



Interrelación del período de actualización aproximado, leva de enclavamiento y franjas de tiempo

Cada franja de tiempo almacena la siguiente información:

Máscara de evento de enclavamiento – Cuando se detecta un evento de enclavamiento, se calcula la franja de tiempo a la cual pertenece y se establece el bit en la máscara del evento de enclavamiento correspondiente al bit de salida del enclavamiento.

Máscara de evento de desenclavamiento – Cuando se detecta un evento de desenclavamiento, se calcula la franja de tiempo a la cual pertenece y se establece el bit en la máscara del evento de desenclavamiento correspondiente al bit de salida del desenclavamiento.

Intervalo – El tiempo en microsegundos desde el inicio de la actualización aproximada en el que se produce el evento de enclavamiento o desenclavamiento.

Máscara de impulso activado – Para las salidas con impulso, se establece la franja de tiempo en la que se calcula un evento de impulso activado, y se establece el bit en la máscara de impulso activado correspondiente al bit de salida del evento de impulso.

Máscara de impulso desactivado – Para las salidas de impulso, se establece la franja de tiempo en la que se calcula un evento de impulso desactivado y se establece el bit en la máscara de impulso desactivado correspondiente al bit de salida del evento de impulso.

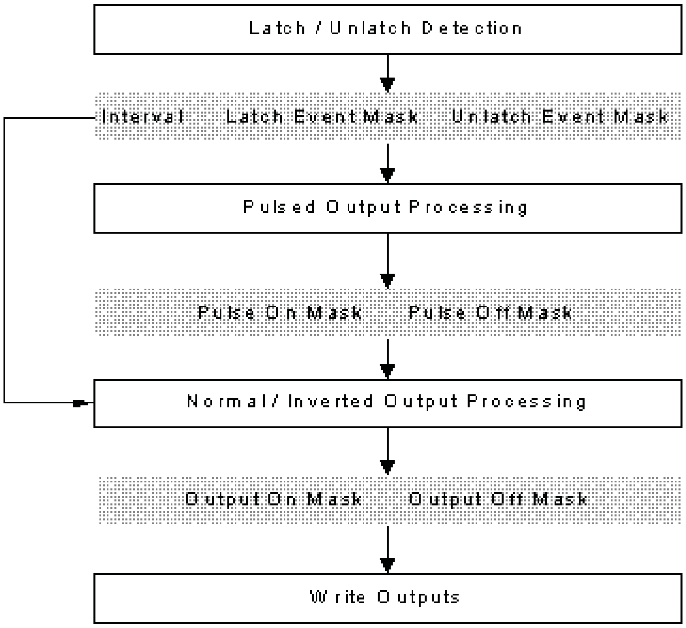
Máscara de salida activada – Para las salidas normales, se establece el bit correspondiente al bit de salida del evento de enclavamiento o impulso activado indicando que se encenderá la salida para estos eventos.

Para las salidas invertidas, se establece el bit correspondiente al bit de salida del evento de desenclavamiento o impulso desactivado indicando que se encenderá la salida para estos eventos.

Máscara de salida desactivada – Para las salidas normales, se establece el bit correspondiente al bit de salida del evento de desenclavamiento o impulso desactivado indicando que se apagará la salida para estos eventos.

Para las salidas invertidas, se establece el bit correspondiente al bit de salida del evento de enclavamiento o impulso activado indicando que se apagará la salida para estos eventos.

A continuación encontrará una descripción general simplificada de cómo se utilizan los datos de la franja de tiempo.



Descripción general del uso de los datos de la franja de tiempo

Las franjas de tiempo también se utilizan para procesar los elementos de leva superpuestos. Se mantiene un semáforo para indicar el estado activo actual de cada bit de salida. Además, si se produce un evento de enclavamiento o desenclavamiento de un elemento de leva programado en la misma franja de tiempo, se cancelan mutuamente.

Subsistema de E/S:

El usuario puede especificar el parámetro de salida de una instrucción MAOC ya sea como un tag de memoria o como un tag de datos del módulo de salida. Un puntero a tag se pasa a la instrucción MAOC. También se pasa un parámetro interno de tipo IO_MAP a la instrucción MAOC. Si el parámetro de salida hace referencia a la salida del controlador, el parámetro IO_MAP es NULO. Si el parámetro de salida hace referencia a un módulo de salida, el parámetro IO_MAP apunta a la estructura del mapa para el módulo. La instrucción MAOC puede entonces determinar si el parámetro Output está asociado con un módulo 1756-OB16IS verificando el tipo de módulo almacenado en la tabla del variador.

Estructura Output Data:

Campo	Tamaño	Descripción
Valor	4 bytes	Valores de datos para bits de salida no programados. 0 = Off 1 = On
Máscara	4 bytes	Selecciona qué bits de salida se programarán. Sólo se pueden programar los primeros ocho bits (0 – 7). 0 = No programado 1 = Programado

Matriz de 16 estructuras programadas:

Campo	Tamaño	Descripción
ID de la estructura	1 byte	Los ID válidos son 1 – 16. Otro valor indica que el programa no se tendrá en cuenta.
Número de secuencia	1 byte	El OB16IS conservará una copia del programa. Un cambio en el número de secuencia le dirá al OB16IS que procese los datos en este programa.
ID del punto	1 byte	Indica el bit de salida asociado con este programa. Introducido como un valor 00 – 07.
Valor del punto	1 byte	Próximo estado del bit de salida especificado en el ID del punto. 0 = Off 1 = On
Sello de hora	4 bytes	Los 32 bits más bajos de CST. Indica cuándo cambiar el estado del bit de salida especificado.

Procesamiento del programa:

El valor y los campos de máscara se procesan y todos los bits de datos no programados se mueven al almacenamiento de datos del módulo de salida. Estos datos se escriben en los terminales de salida luego de que todos los programas hayan sido procesados.

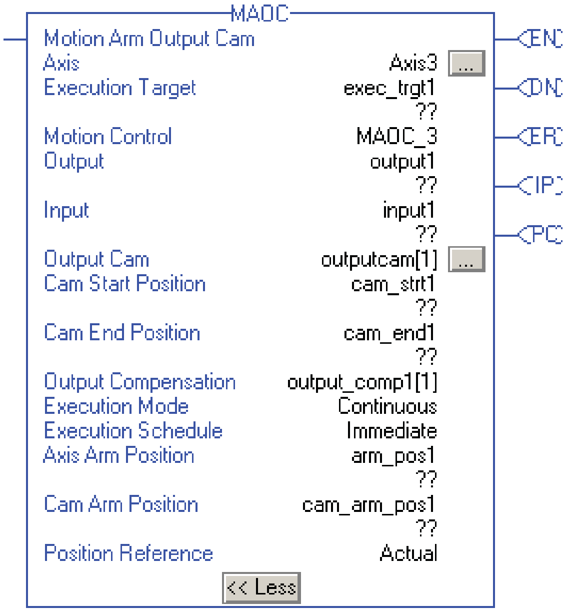
Se procesa cada programa No se tendrá en cuenta el programa si:

- El ID del programa no está en el rango de 1 – 16
- El ID del punto no está en el rango de 0 – 7
- El número de secuencia no ha cambiado

Si el programa se tendrá en cuenta, se marca como “active”.

Todos los programas “active” se examinan cada 200 micro-segundos. El sello de hora del programa se compara con el CST actual. Si el CST actual es mayor o igual al sello de hora programado, el valor del punto en el programa se mueve al almacenamiento de datos del módulo de salida para el bit de salida especificado.

Ejemplo M: *Lógica de escalera de relés*



MAOC
Motion Arm Output Cam
Axis Axis3
Execution Target exec_trgt1 ??
Motion Control MAOC_3
Output output1 ??
Input input1 ??
Output Cam outputcam[1]
Cam Start Position cam_strt1 ??
Cam End Position cam_end1 ??
Output Compensation output_comp1[1]
Execution Mode Continuous
Execution Schedule Immediate
Axis Arm Position arm_pos1 ??
Cam Arm Position cam_arm_pos1 ??
Position Reference Actual

Ejemplo de lógica de escalera de la instrucción MAOC

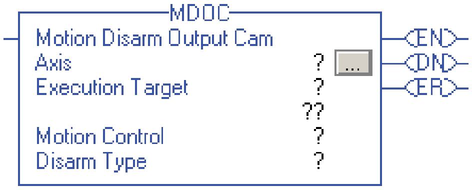
Texto estructurado

MAOC(Axis3,exec_trgt1,MAOC_3,output1, input1,outputcam1[1],cam_strt1,cam_end1, output_comp1[1],continuous,immediate,arm_pos1, cam_arm_pos1,actual);

Motion Disarm Output Cam (MDOC)

La instrucción MDOC inicia la desactivación de una o más levas de salida conectadas al eje especificado. Según el tipo de desactivación, MDOC desactiva ya sea todas las levas de salida o sólo una leva de salida específica. Las salidas correspondientes se mantienen en el último estado luego de la desactivación.

Operandos: *Lógica de escalera de relé*


MDOC
Motion Disarm Output Cam
Axis ?
Execution Target ?
??
Motion Control ?
Disarm Type ?

Operando	Tipo	Formato	Descripción
AXIS	AXIS_FEEDBACK AXIS_CONSUMED AXIS_VIRTUAL AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje que proporciona la entrada de posición a la leva de salida. La elipsis ejecuta el diálogo Axis Properties.
Objetivo de ejecución	SINT, INT o DINT	inmediato o tag	El objetivo de ejecución define la leva de salida específica del conjunto conectado al eje denominado. El comportamiento está determinado por lo siguiente: · 0...8 – Levas de salida ejecutadas en el controlador Logix. · 9...31 – Reservada para futuros usos.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura utilizada para obtener acceso a los parámetros de estado de la instrucción.
Disarm Type	UINT32	inmediato	Selecciona una o todas las levas de salida que se desactivarán para un eje específico. Puede seleccionar: 0 = Todas – Desactiva todas las levas de salida conectadas al eje especificado. 1 = Específico – Desactiva la leva de salida que está conectada al eje especificado y definida por el objetivo de ejecución.

Texto estructurado

MDOC(Axis,ExecutionTarget,
MotionControl,DisarmType);

Los operandos son iguales a los de la instrucción MDOC de lógica de escalera de relés.

En el caso de los operandos para los que usted debe seleccionar entre opciones disponibles, introduzca su selección como:

Este operando	**Tiene estas opciones que usted...**	
	introduce como texto	**o introduce como número**
Disarm Type	todas	0
	específico	1

Estructura MOTION_INSTRUCTION

Mnemónico:	**Descripción:**
Bit .EN (Habilitar) 31	Se establece cuando el renglón hace una transición de falso a verdadero y permanece establecido hasta que el renglón pasa a falso.
Bit .DN (Listo) 29	Se establece cuando la(s) leva(s) de salida se han desactivado exitosamente.
Bit .ER (Error) 28	Se establece para indicar que la instrucción detectó un error.

Descripción: La instrucción MDOC desactiva una leva de salida específica o todas para un eje especificado según el tipo de desactivación seleccionado. El eje proporciona la entrada de posición para la leva de salida. El objetivo de ejecución define la leva de salida específica del conjunto que está conectado al eje especificado.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

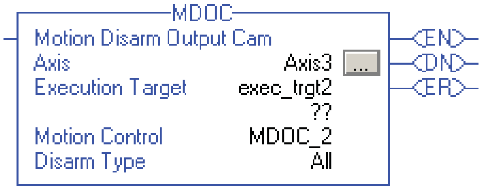
Códigos de error: Vea Códigos de error (ERR) para las instrucciones de control de movimiento.

Códigos de error extendidos: Los códigos de error extendidos proporcionan información adicional específica de instrucciones para los códigos de error que son genéricos a muchas instrucciones. Los códigos de error extendidos para el código de error Parámetro fuera de rango (13) incluyen un número que hace referencia al número de operando tal y como aparecen en la plantilla de arriba hacia abajo, con el primer operando como cero. Por lo tanto, para la instrucción MDOC, un código de error extendido de 4 haría referencia al valor del operando Disarm Type. Entonces, debería verificar su valor con el rango aceptado de valores para la instrucción.

Bits de estado: *MDOC cambia a bits de estado*

ninguno

Ejemplo: *Lógica de escalera de relés*



MDOC
Motion Disarm Output Cam
Axis Axis3
Execution Target exec_trgt2
??
Motion Control MDOC_2
Disarm Type All

Ejemplo de lógica de escalera MDOC

Texto estructurado

```
MDOC(Axis3,exec_trgt2,MDOC_2,all);
```

Notas:

Capítulo 6

Instrucciones de configuración de movimiento

(MAAT, MRAT, MAHD, MRHD)

ATENCIÓN

Los tags para instrucciones de control de movimiento se deben utilizar sólo una vez. Reutilizar el tag de control de movimiento en otras instrucciones puede provocar operaciones imprevistas. Esto puede dañar el equipo o causar lesiones personales.

Introducción

Las instrucciones de configuración incluyen todas las instrucciones de control de movimiento que se utilizan para establecer y aplicar parámetros de servo configuración a un eje. Este grupo de instrucciones incluye instrucciones de diagnóstico de prueba de conexión e instrucciones de ajuste.

Utilice las instrucciones de configuración de movimiento para ajustar un eje y ejecutar pruebas de diagnóstico para el servosistema. Estas pruebas incluyen:

- Una prueba de conexión del motor con encoder.
- Una prueba de conexión del encoder.
- Una prueba de marcador

Las instrucciones de configuración de movimiento son:

Si desea	Use esta instrucción	Disponible en estos lenguajes
Calcular un conjunto completo de ganancias servo y límites dinámicos en base a los resultados de una instrucción MRAT ejecutada previamente. La instrucción MAAT también actualiza el servo módulo con los nuevos parámetros de ganancia.	MAAT	Lógica de escalera de relés Texto estructurado
Ordenar al servo módulo que ejecute un perfil de ajuste para el eje.	MRAT	Lógica de escalera de relés Texto estructurado

Si desea	Use esta instrucción	Disponible en estos lenguajes
Aplicar los resultados de una instrucción MRHD ejecutada previamente. La instrucción MAHD genera un nuevo conjunto de polaridades de encoder y servo basado en la dirección de movimiento observada durante la instrucción MRHD.	MAHD	Lógica de escalera de relés Texto estructurado
Ordenar al servo módulo que ejecute una de las tres pruebas de diagnóstico en un eje.	MRHD	Lógica de escalera de relés Texto estructurado

Motion Apply Axis Tuning (MAAT)

La instrucción MAAT se utiliza para calcular un conjunto completo de ganancias servo y límites dinámicos en base a los resultados de una instrucción MRAT ejecutada previamente y para actualizar el módulo de control de movimiento con estos nuevos parámetros de ganancia. Ya que esta instrucción no tiene parámetros explícitos, la entrada deriva de los parámetros de configuración de ajuste para el eje como se describe en la tabla de parámetros del estado de ajuste página 242. Luego de la ejecución de la instrucción MAAT, el eje correspondiente debe estar preparado para servo activación.

Operandos: *Lógica de escalera de relés*

Operando	Tipo	Formato	Descripción
Axis	AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje en el cual se realizará la operación.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura utilizada para acceder a los parámetros de estado de la instrucción.

Texto estructurado

MAAT(Axis,MotionControl);

Los operandos son iguales a los de la instrucción MAAT de lógica de escalera de relés.

Estructura MOTION_INSTRUCTION

Mnemónico	Descripción
Bit .EN (Habilitado) 31	El bit de habilitación indica cuando la instrucción está habilitada. Permanece activo hasta completar la comunicación servo y hasta que la condición de entrada del renglón sea falsa.
Bit .DN (Listo) 29	El bit Listo indica cuando la instrucción ha efectuado un proceso de ajuste para el eje.
Bit .ER (Error) 28	El bit de error indica cuando la instrucción detecta un error, como por ejemplo, si el eje no está configurado.

Descripción: La instrucción MAAT se utiliza para realizar una serie de cálculos que arrojan como resultado los valores de ganancia y de los parámetros de configuración dinámica del eje específico. Como parte del trabajo realizado por MAAT, se aplican estos parámetros de configuración por lo tanto el eje está preparado para servo operación total. Esta instrucción ha sido diseñada para ejecutarse después de la instrucción MRAT que genera valores de configuración de entrada del eje para la instrucción MAAT. Ver la descripción de la instrucción MRAT para más información. MAAT no necesita parámetros de entrada explícitos; simplemente ingrese o seleccione el eje físico deseado.

Si el eje específico no aparece en la lista de ejes disponibles, dicho eje no ha sido configurado para la operación. Utilice Tag Editor para crear y configurar un nuevo eje.

La instrucción MAAT utiliza los parámetros entrada y salida para configuración del eje. Los parámetros de configuración de entrada que utiliza MRAT se indican en la siguiente tabla. Consulte la especificación del objeto Eje de Movimiento para una descripción más detallada de estos parámetros.

Los parámetros de entrada de configuración del eje que utiliza MAAT dependen de la configuración del variador externo. Si el parámetro de configuración del bit External Vel Servo Drive es VERDADERO, indicando interface con un servovariador de velocidad externo, se necesitan los siguientes parámetros de entrada.

Parámetro del eje	Tipo de datos	Unidades	Significado
Tuning Velocity	Real	pos unidades/seg	Máxima velocidad del perfil de ajuste.
Tune Accel	Real	pos unidades/seg^2	Tiempo de aceleración calculado del perfil de ajuste.
Tune Decel	Real	pos unidades/seg^2	Tiempo de desaceleración calculado del perfil de ajuste.
Tune Velocity Scaling	Real	mV/KCPS	Factor de escalado de velocidad medido para sistema de eje del Variador/Motor/Encoder.
Tune Velocity Bandwidth	Real	Hertz	Ancho de banda del servovariador de velocidad externo.

Si el parámetro de configuración del bit External Vel Servo Drive es FALSO, indicando interface con un servovariador de par externo, se necesitan los siguientes parámetros de entrada.

Parámetro del eje	Tipo de datos	Unidades	Significado
Damping Factor	Real	-	Factor de amortiguación utilizado para calcular las ganancias.
Tuning Velocity	Real	pos unidades/seg	Máxima velocidad del perfil de ajuste.
Tune Accel	Real	pos unidades/seg^2	Tiempo de aceleración calculado del perfil de ajuste.

Parámetro del eje	Tipo de datos	Unidades	Significado
Tune Decel	Real	pos unidades/seg^2	Tiempo de desaceleración calculado del perfil de ajuste.
Effective Inertia	Real	mV/KCPS^2	Inercia efectiva calculada del sistema de variador/motor.
Position Servo Bandwidth	Real	Hertz	Máxima posición del ancho de banda del lazo del servo.

Los parámetros de configuración del eje que genera MAAT como salida, dependen de la configuración del variador externo. Si el parámetro de configuración del bit External Vel Servo Drive es VERDADERO, indicando interface con un servovariador de velocidad externo, se necesitan los siguientes parámetros de salida.

Parámetro del eje	Tipo de datos	Unidades	Significado
Pos Proportional Gain	Real	1/mseg	Posición de la ganancia proporcional de lazo del servo.
Pos Integral Gain	Real	1/mseg^2	Posición de la ganancia integral del lazo del servo – Configurado a cero.
Velocity Feedforward	Real	-	Posición de la ganancia proporcional de lazo del servo.
Acceleration Feedforward	Real	-	Ganancia anticipada del comando de velocidad – Configurado a cero.
Max Speed	Real	pos unidades/seg	Velocidad máxima para perfiles de movimiento – Configurado a velocidad de ajuste.
Max Acceleration	Real	pos unidades/seg^2	Aceleración máxima para perfiles de movimiento.
Max Deceleration	Real	pos unidades/seg^2	Aceleración máxima para perfiles de movimiento.
Output Filter Bandwidth	Real	Hertz	Ancho de banda del filtro de paso bajo de salida del servo.
Output Scaling	Real	mV/KCPS	Factor de escalado aplicado a la salida de la posición del lazo del servo a DAC.
Position Error Tolerance	Real	pos unidades	Máximo error de posición de lazo del servo permitido sin fallo.

Si el parámetro de configuración del bit External Vel Servo Drive es FALSO, indicando interface con un servovariador de par externo, se generan los siguientes parámetros de salida.

Parámetro del eje	Tipo de datos	Unidades	Significado
Pos Proportional Gain	Real	1/mseg	Posición de la ganancia proporcional de lazo del servo
Pos Integral Gain	Real	1/mseg^2	Posición de la ganancia integral de lazo del servo
Vel Proportional Gain	Real	1/mseg	Velocidad de la ganancia proporcional de lazo del servo

Parámetro del eje	**Tipo de datos**	**Unidades**	**Significado**
Vel Integral Gain	Real	1/mseg^2	Velocidad de la ganancia integral de lazo del servo
Velocity Feedforward	Real	-	Posición de la ganancia proporcional de lazo del servo
Acceleration Feedforward	Real	-	Ganancia anticipada del comando de velocidad
Max Speed	Real	pos unidades/ seg	Velocidad máxima para perfiles de movimiento – Configurado a velocidad de ajuste
Max Acceleration	Real	pos unidades/ seg^2	Aceleración máxima para perfiles de movimiento
Max Deceleration	Real	pos unidades/ seg^2	Aceleración máxima para perfiles de movimiento
Output Filter Bandwidth	Real	Hertz	Ancho de banda del filtro de paso bajo de salida del servo
Output Scaling	Real	mV/KCPS^2	Factor de escalado aplicado a la salida de la velocidad del lazo del servo a DAC.
Position Error Tolerance	Real	pos unidades	Máximo error de posición de lazo del servo permitido sin fallo.

Estos parámetros de salida generados por la instrucción MAAT se aplican inmediatamente al eje específico, por lo que se puede realizar el movimiento posterior.

Para más información sobre parámetros de configuración del ajuste consulte la tabla de parámetros del estado de ajuste en página 242.

Para ejecutar correctamente una instrucción MAAT, el eje específico se debe configurar como un servoeje y estar en estado de Eje Preparado, con la acción servo desactivada. Si no se cumple con estas condiciones, la instrucción falla.

IMPORTANTE La ejecución de la instrucción MAAT puede tomar múltiples escanes para ejecutar ya que la misma requiere transmisión de un mensaje al módulo de control de movimiento. El bit ejecutado (.DN) no se configura inmediatamente, sólo luego de que este mensaje sea transmitido correctamente.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Consulte Códigos de error (ERR) para las instrucciones de control de movimiento.

Códigos de error extendidos: Los códigos de error extendidos brindan información adicional específica de la instrucción para los códigos de error genéricos a muchas instrucciones. Los siguientes códigos de error extendidos ayudan a establecer el problema cuando la instrucción MAAT recibe un mensaje de error de Fallo de mensajes al servo módulo (12).

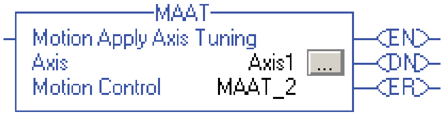
Código de error asociado (decimal)	Código de error extendido (decimal)	Significado
SERVO_MESSAGE_FAILURE (12)	Sin recursos (2)	No hay suficientes recursos de memoria para completar la demanda. (SERCOS)
SERVO_MESSAGE_FAILURE (12)	Conflicto Modo de objeto (12)	El eje está desactivado.
SERVO_MESSAGE_FAILURE (12)	Permiso denegado (15)	Error al habilitar el interruptor de entrada. (SERCOS)
SERVO_MESSAGE_FAILURE (12)	Dispositivo en estado incorrecto (16)	La Redefinición de la posición, inicio y registro 2 son excluyentes entre sí (SERCOS), el estado del dispositivo no es correcto para la acción. (SERCOS)

Bit de estado: *MAAT Cambia a Bit de estado*

Ninguno

Ejemplo: Cuando las condiciones de entrada son verdaderas, el controlador calcula un conjunto completo de ganancias servo y límites dinámicos para *eje1 en base a los resultados de una instrucción MRAT ejecutada previamente.*

Lógica de escalera de relés



MAAT
Motion Apply Axis Tuning
Axis Axis1
Motion Control MAAT_2

Ejemplo de lógica de escalera MAAT

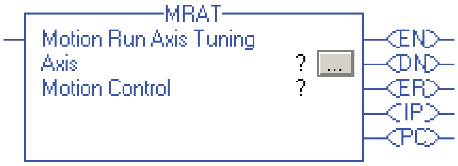
Texto estructurado

MAAT(Axis1,MAAT_2);

Motion Run Axis Tuning (MRAT)

Utilice la instrucción MRAT para ordenar al módulo de control de movimiento que ejecute un perfil de ajuste de movimiento para el eje específico. El perfil de ajuste de movimiento consta de una o más rampas de aceleración y desaceleración impulsadas mediante la aplicación de voltajes fijos a la salida del variador del servo. Observe que esta instrucción no cierra el lazo del servo en ningún momento. Ya que esta instrucción no tiene parámetros explícitos de entrada, esta deriva de los parámetros de configuración de ajuste para el eje. El resultado de la ejecución de la instrucción MRAT es un conjunto de datos de medición que se guarda en el Axis Object para su posterior uso con la instrucción MAAT.

Operandos: *Lógica de escalera de relés*



Operando	Tipo	Formato	Descripción
Axis	AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje en el cual se realizará la operación.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura utilizada para acceder a los parámetros de estado de la instrucción.

Texto estructurado

MRAT(Axis,MotionControl);

Los operandos son iguales a los de la instrucción MRAT de lógica de escalera de relés.

Estructura MOTION_INSTRUCTION

Mnemónico	Descripción
Bit .EN (Habilitado) 31	Se establece cuando el renglón realiza una transición de falso a verdadero y permanece habilitado hasta que la transacción de mensaje servo se completa y el renglón cambia a falso.
Bit .DN (Listo) 29	Se establece luego de que el proceso de ajuste se completa correctamente.
Bit .ER (Error) 28	Se establece para indicar que la instrucción ha detectado un error, tal como especificar un eje no configurado.
Bit .IP (En proceso) 26	Se establece en la transición de renglón positiva y desaparece luego que el proceso de ajuste se completa, o se interrumpe mediante un comando de paro, desactivación o falla del servo.
Bit .PC (Proceso completo) 27	Se establece luego de que el proceso de ajuste se completa correctamente.

Descripción: La instrucción MRAT se utiliza para ejecutar un perfil de ajuste de movimiento para el eje específico. Durante este breve perfil de ajuste de movimiento, el módulo de control de movimiento realiza mediciones de velocidad y tiempo que se utilizan como datos de entrada para una instrucción MAAT posterior. MRAT no necesita parámetros de entrada explícitos; simplemente ingrese o seleccione el eje físico deseado.

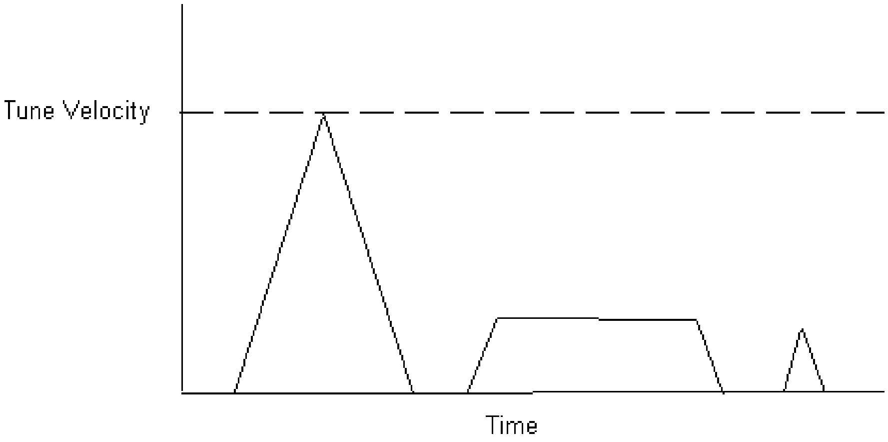
Si el eje específico no aparece en la lista de ejes disponibles, dicho eje no ha sido configurado para la operación. Utilice Tag Editor para crear y configurar un nuevo eje.

La instrucción MRAT utiliza los parámetros entrada y salida para configuración del eje. Los parámetros de configuración de entrada que utiliza MRAT se indican en la siguiente tabla.

Parámetro del eje	Tipo de datos	Unidades	Significado
Tuning Direction	Booleano	-	Dirección del movimiento de ajuste (0-hacia adelante, 1-hacia atrás).
Tuning Travel Limit	Real	pos unidades	Carrera máxima permitida del eje.
Tuning Velocity	Real	pos unidades/seg	Máxima velocidad del perfil de ajuste.
Damping Factor	Real	-	Factor de amortiguación utilizado para calcular la posición máxima del ancho de banda del servo.

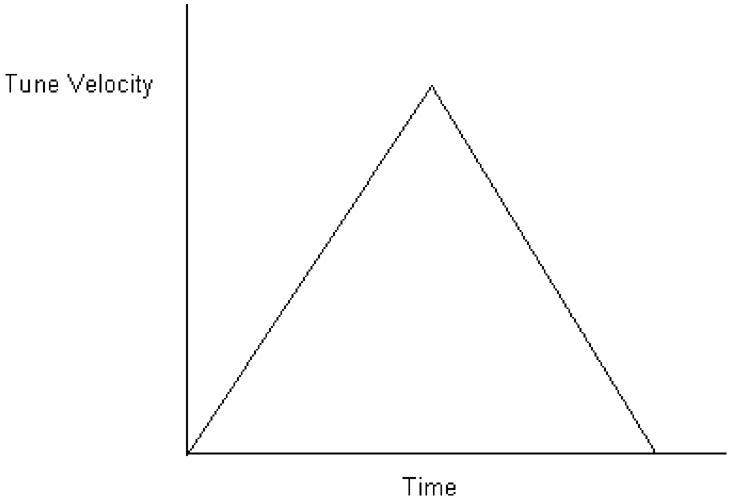
Basado en los parámetros de configuración antes mencionados, al ejecutar MRAT se genera un evento de movimiento que consiste en un simple perfil de velocidad triangular o una serie de tres de estos perfiles. La velocidad de ajuste debe estar dentro de las capacidades de velocidad máxima del variador y motor. El valor configurado para la velocidad de ajuste se debe establecer a la velocidad máxima de funcionamiento del eje, por lo tanto los parámetros de ajuste resultantes se basan en la dinámica del sistema a esa velocidad.

Si el parámetro de configuración del bit External Vel Servo Drive es VERDADERO, indicando interface con un servovariador de velocidad externo, se aplican tres impulsos al eje. El perfil de velocidad de ajuste para este caso se indica en el siguiente esquema.



Perfil de velocidad de ajuste en Verdadero

Si el parámetro de configuración del bit External Vel Servo Drive es FALSO, indicando interface con un servovariador de par externo, sólo se aplica un impulso al eje. El perfil de velocidad de ajuste se indica a continuación.



Perfil de velocidad de ajuste en Falso

Los parámetros de configuración del eje que genera MRAT como salida, dependen de la configuración del variador externo. Si el parámetro de configuración del bit External Vel Servo Drive es VERDADERO, indicando interface con un servovariador de velocidad externo, se necesitan los siguientes parámetros de salida.

Parámetro del eje	Tipo de datos	Unidades	Significado
Tune Status	Real	-	Informe de estado del proceso de ajuste.
Tune Accel Time	Real	segundos	Tiempo de aceleración medido del perfil de ajuste.
Tune Decel Time	Real	segundos	Tiempo de desaceleración medido del perfil de ajuste.
Tune Accel	Real	pos unidades/seg^2	Tiempo de aceleración calculado del perfil de ajuste.
Tune Decel	Real	pos unidades/seg^2	Tiempo de desaceleración calculado del perfil de ajuste.
Tune Velocity Scaling	Real	mV/KCPS	Factor de escalado de velocidad medido para sistema de eje del Variador/Motor/Encoder.
Tune Rise Time	Real	mV/KCPS	Tiempo de elevación medido del perfil de respuesta de paso a paso del ajuste.
Tune Velocity Bandwidth	Real	Hertz	Ancho de banda calculado del servovariador de velocidad externo.

Si el parámetro de configuración del bit External Vel Servo Drive es FALSO, indicando interface con un servovariador de par externo, se generan los siguientes parámetros de salida.

Parámetro del eje	Tipo de datos	Unidades	Significado
Tune Status	Real	-	Informe de estado del proceso de ajuste.
Tune Accel Time	Real	segundos	Tiempo de aceleración medido del perfil de ajuste.
Tune Decel Time	Real	segundos	Tiempo de desaceleración medido del perfil de ajuste.
Tune Accel	Real	pos unidades/seg^2	Tiempo de aceleración calculado del perfil de ajuste.
Tune Decel	Real	pos unidades/seg^2	Tiempo de desaceleración calculado del perfil de ajuste.
Effective Inertia	Real	mV/KCPS^2	Inercia efectiva calculada del sistema de variador/motor.
Position Servo Bandwidth	Real	Hertz	Máxima posición calculada del ancho de banda del lazo del servo.

Estos parámetros de salida generados por la instrucción MRAT se utilizan como entradas para la instrucción MAAT posterior que realiza cálculos de ajuste adicionales y aplica los resultados a diversos parámetros de configuración dinámica y del eje del servo.

Parámetro del estado de ajuste

Pueden ocurrir algunas situaciones que impidan al controlador realizar correctamente la operación de ajuste. En este caso, el proceso de ajuste se cancela automáticamente y se informa un fallo de ajuste que se almacena en el parámetro de salida del estado de ajuste (GSVable). También es posible cancelar manualmente un proceso de ajuste utilizando una instrucción MAS lo que resulta en un fallo de ajuste informado por el parámetro de estado del ajuste. Los posibles valores para el estado de ajuste se indican en la siguiente tabla.

Código de estado	Código	Significado
Tune Success	0	El proceso de ajuste se ha realizado correctamente.
Tune In Process	1	El ajuste está en progreso.
Tune Aborted	2	El proceso de ajuste fue cancelado por el usuario.
Tune Time-out	3	El proceso de ajuste ha sobrepasado el tiempo de espera
Tune Servo Fault	4	Fallo del proceso de ajuste debido a una falla del servo
Tune Travel Fault	5	El eje alcanzó el fin de carrera de ajuste

Código de estado	Código	Significado
Tune Polarity Fault	6	Dirección incorrecta del movimiento del eje debido a la configuración incorrecta de polaridad del motor/encoder.
Tune Speed Fault	7	Velocidad de ajuste del eje demasiado baja para lograr exactitud de medición mínima.
Tune Configuration Fault	8	La configuración de ajuste del eje específico no está permitida y la operación falla.

IMPORTANTE No confundir el parámetro de estado de ajuste para el sub-tag .STATUS de la instrucción MRAT.

Para ejecutar correctamente una instrucción MRAT en un eje, el eje específico se debe configurar como un tipo de servoeje y estar en estado de Eje Preparado. Si no se cumple con estas condiciones, la instrucción falla.

IMPORTANTE Cuando la instrucción MRAT se ejecuta en primer lugar, se establece el bit En proceso (.IP) y se elimina el bit de Proceso completo (.PC). La ejecución de la instrucción MRAT puede tomar múltiples escanes para ejecutar ya que la misma requiere transmisión de múltiples mensajes al módulo de control de movimiento. El bit ejecutado (.DN) no se configura inmediatamente, sólo luego de que estos mensajes sean transmitidos correctamente. Se elimina el bit En proceso (.IP) y se establece el bit de Proceso completo (.PC) al mismo tiempo que el bit ejecutado (.DN).

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.

- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error: Consulte Códigos de error (ERR) para las instrucciones de control de movimiento.

Códigos de error extendidos:

Los códigos de error extendidos brindan información adicional específica de la instrucción para los códigos de error genéricos a muchas instrucciones. Los siguientes códigos de error extendidos ayudan a establecer el problema cuando la instrucción MRAT recibe un mensaje de error de fallo de mensajes al servomódulo (12).

Código de error asociado (decimal)	Código de error extendido (decimal)	Significado
SERVO_MESSAGE_FAILURE (12)	Proceso finalizado a pedido (1)	La ejecución de ajuste seguida de una instrucción para interrupción/inhabilitación del variador, o una instrucción para detener el movimiento o un cambio de procesador requiere cancelar el ajuste.
SERVO_MESSAGE_FAILURE (12)	Conflicto Modo de objeto (12)	El eje está desactivado.
SERVO_MESSAGE_FAILURE (12)	Dispositivo en estado incorrecto (16)	Orden de proceso de ajuste incorrecta. (SERCOS)

Bit de estado:

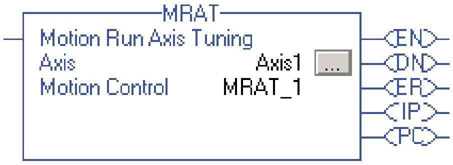
MRAT Cambia a Bit de estado

Nombre del bit:	Estado:	Significado:
DriveEnableStatus	VERDADERO	El eje está en estado de control del variador con salida habilitada del variador activa mientras el perfil de ajuste se está ejecutando.
TuneStatus	VERDADERO	El eje está ejecutando un proceso de ajuste.

Ejemplo:

Cuando las condiciones de entrada son verdaderas, el controlador ordena al servo módulo a ejecutar un perfil de movimiento de ajuste para *eje1*.

Lógica de escalera de relés



Axis Axis1
Motion Control MRAT_1

Ejemplo de lógica de escalera MRAT

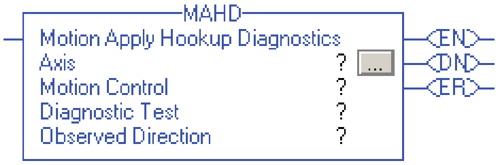
Texto estructurado

MAR(Axis1,MRAT_1);

Motion Apply Hookup Diagnostics (MAHD)

La instrucción MAHD se utiliza para aplicar los resultados de una instrucción MRHD ejecutada previamente para generar un nuevo conjunto de polaridades de encoder y servo basado en la dirección de movimiento observada durante la prueba. Como parte del proceso de aplicación, la instrucción actualiza el módulo de control de movimiento con estas nuevas configuraciones de polaridad. Luego de la ejecución de la instrucción MAHD, y asumiendo que un conjunto estable de ganancias ha sido establecido, el eje correspondiente debe estar preparado para servo activación.

Operandos: *Lógica de escalera de relés*



Operando	**Tipo**	**Formato**	**Descripción**
Axis	AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje en el cual se realizará la operación.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura utilizada para acceder a los parámetros de estado de la instrucción.
Diagnostic test	UDINT	inmediato	Escoja la prueba específica para el módulo de control de movimiento: 0 = prueba de conexión de motor/ encoder 1 = prueba de conexión de encoder 2 = prueba de marcador de encoder
Observed direction	BOOLEANO	inmediato	Establece la dirección del movimiento de la prueba. Escoja uno de los dos: 0 = avance 1 = retroceso

Texto estructurado

MAHD(Axis,MotionControl, DiagnosticTest, ObservedDirection);

Los operandos son iguales a los de la instrucción MAHD de lógica de escalera de relés.

Para los operandos necesarios para las opciones disponibles, ingrese su selección como:

Este operando	**Cuenta con estas opciones que usted...**	
	ingresa como texto	**o ingresa como número**
DiagnosticTest	motor_encoder	0
	encoder	1
	marker (marcador)	2
ObservedDirection	avance	0
	retroceso	1

Estructura MOTION_INSTRUCTION

Mnemónico	Descripción
Bit .EN (Habilitado) 31	se establece cuando el renglón realiza una transición de falso a verdadero y permanece habilitado hasta que la transacción de mensaje servo se completa y el renglón cambia a falso.
Bit .DN (Listo) 29	se establece después de que el proceso de prueba de conexión se ha ejecutado correctamente.
Bit .ER (Error) 28	se establece para indicar que la instrucción ha detectado un error, tal como especificar un eje no configurado.

Descripción: La instrucción MAHD se utiliza para ejecutar una serie de cálculos que arrojan valores para los parámetros de configuración del bit de polaridades de encoder y servo del eje específico. Como parte del trabajo realizado por MAHD, se aplican estos parámetros de configuración del bit al módulo de control de movimiento, por lo tanto el eje está preparado para servo operación total. Esta instrucción ha sido diseñada para ser ejecutada después de la instrucción MRHD que genera valores de configuración de entrada del eje para la instrucción MAHD. Ver la descripción de la instrucción MRHD para más información. MAHD requiere especificación de la Prueba de Diagnóstico para aplicar y la Dirección de movimiento observada durante el proceso de prueba MRHD previo. Ingrese o seleccione la Prueba de Diagnóstico y la Dirección Observada y el eje físico deseado.

Si el eje específico no aparece en la lista de ejes disponibles, dicho eje no ha sido configurado para la operación. Utilice Tag Editor para crear y configurar un nuevo eje.

La instrucción MAHD utiliza los parámetros entrada y salida para configuración del eje. Los parámetros de configuración de entrada que utiliza MAHD se indican en la siguiente tabla. Se establece automáticamente el bit de Prueba de Dirección de avance como salida de la instrucción MRHD. Consulte la especificación del objeto Eje de Movimiento para una descripción más detallada de estos y otros parámetros.

Parámetro del eje	Tipo de datos	Unidades	Definición
Test Direction Forward	Booleano	-	Dirección del recorrido del eje durante la prueba de conexión como establecida por el módulo de control de movimiento.

Prueba de conexión del motor encoder

Si se escoge la Prueba motor encoder, el controlador calcula la configuración correcta para la Polaridad del encoder y la Polaridad del variador basadas en el parámetro de la instrucción de dirección observada y el estado del bit de prueba de dirección de avance fue establecido por la salida de la instrucción MRHD. Una vez que las configuraciones de polaridad del encoder y polaridad del variador se calculan, MAHD aplica estos valores a los bits de parámetros de configuración del eje correspondiente como se indica en la siguiente tabla:

Parámetro del eje	Tipo de datos	Unidades	Definición
Encoder Polarity Negative	Booleano	-	Invierte el sentido la entrada de retroalimentación del encoder al módulo de control de movimiento.
Drive Polarity Negative	Booleano	-	Invierte el sentido de la salida análoga DAC desde el módulo de control de movimiento.

Prueba de conexión del encoder

Si se escoge la Prueba del encoder, el controlador calcula la configuración correcta sólo para la Polaridad del encoder basada en el parámetro de la instrucción de dirección observada y el estado del bit de prueba de dirección de avance fue establecido por la salida de la instrucción MRHD. Una vez que las configuraciones de polaridad del encoder y polaridad del variador se calculan, MAHD aplica estos valores a los bits de parámetros de configuración del eje correspondiente como se indica en la siguiente tabla:

Parámetro del eje	Tipo de datos	Unidades	Definición
Encoder Polarity Negative	Booleano	-	Invierte el sentido la entrada de retroalimentación del encoder al módulo de control de movimiento.

Para ejecutar correctamente una instrucción MAHD mediante la Prueba motor encoder, el eje específico se debe configurar como tipo de eje Servo o Sólo retroalimentación. Si no se cumple con estas condiciones, la instrucción falla.

IMPORTANTE La ejecución de la instrucción MAHD puede tomar múltiples escanes para ejecutar ya que la misma requiere transmisión de un mensaje al módulo de control de movimiento. El bit ejecutado (.DN) no se configura inmediatamente, sólo luego de que este mensaje sea transmitido correctamente.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error:

Consulte Códigos de error (ERR) para las instrucciones de control de movimiento.

Códigos de error extendidos:

Los códigos de error extendidos brindan información adicional específica de la instrucción para los códigos de error genéricos a muchas instrucciones. Los siguientes códigos de error extendidos ayudan a establecer el problema cuando la instrucción MRAT recibe un mensaje de error de fallo de mensajes al servomódulo (12).

Código de error asociado (decimal)	Código de error extendido (decimal)	Significado
SERVO_MESSAGE_FAILURE (12)	Sin recursos (2)	No hay suficientes recursos de memoria para completar la demanda. (SERCOS)
SERVO_MESSAGE_FAILURE (12)	Conflicto Modo de objeto (12)	El eje está desactivado.
SERVO_MESSAGE_FAILURE (12)	Permiso denegado (15)	Error al habilitar el interruptor de entrada. (SERCOS)
SERVO_MESSAGE_FAILURE (12)	Dispositivo en estado incorrecto (16)	La redefinición de la posición, inicio y registro 2 son excluyentes entre sí (SERCOS), el estado del dispositivo no es correcto para la acción. (SERCOS)

Bit de estado:

MAHD Cambia a Bit de estado

Ninguno

Ejemplo:

Cuando las condiciones de entrada son verdaderas, el controlador aplica los resultados de la instrucción MRHD al *eje1*.

Lógica de escalera de relés

MAHD
Motion Apply Hookup Diagnostics
Axis Axis1
Motion Control MAHD_1
Diagnostic Test Marker
Observed Direction Forward

Ejemplo de lógica de escalera MAHD

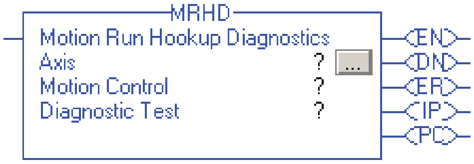
Texto estructurado

MAHD(axis1,axis1_MAHD,marker,forward);

Motion Run Hookup Diagnostics (MRHD)

Utilice la instrucción MRHD para ordenar al módulo de control de movimiento que ejecute cualquiera de tres diagnósticos diferentes en el eje especificado según la prueba ID. Se encuentran disponibles diagnósticos actuales para examinar la conexión motor/encoder para un servo eje, la conexión del encoder solamente, y la conexión del marcador encoder. Sólo el diagnóstico motor/encoder pone en marcha el movimiento en el eje. Esta acción consiste en un pequeño movimiento de un Incremento de la prueba del motor encoder. El movimiento se inicia aproximadamente con 1 volt por segundo en el nivel de rampas de la salida del variador del servo. El resultado de la ejecución de la instrucción MRHD es que los parámetros, Estado de la prueba y Dirección de la prueba de avance, se actualizan.

Operandos: *Lógica de escalera de relés*



Operando	Tipo	Formato	Descripción
Axis	AXIS_SERVO AXIS_SERVO_DRIVE	tag	Nombre del eje en el cual se realizará la operación.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura utilizada para acceder a los parámetros de estado de la instrucción.
Diagnostic test	DINT	inmediato	Escoja la prueba específica para el módulo de movimiento: 0 = prueba de conexión de motor/ encoder 1 = prueba de conexión de encoder 2 = prueba de conexión del marcador de encoder 3 = prueba de temporizador de control (Watchdog) OK

Texto estructurado

MRHD(Axis,MotionControl, DiagnosticTest);

Los operandos son iguales a los de la instrucción MRHD de lógica de escalera de relés.

Para los operandos necesarios para las opciones disponibles, ingrese su selección como:

Este operando:	Cuenta con estas opciones que usted...	
	ingresa como texto:	**o ingresa como número:**
DiagnosticTest	motor_encoder	0
	encoder	1
	marker (marcador)	2

Estructura MOTION_INSTRUCTION

Mnemónico	Descripción
Bit .EN (Habilitado) 31	Se establece cuando el renglón realiza una transición de falso a verdadero y permanece habilitado hasta que la transacción de mensaje servo se completa y el renglón cambia a falso.
Bit .DN (Listo) 29	Se establece después de que el proceso de prueba de conexión se ha ejecutado correctamente.
Bit .ER (Error) 28	Se establece para indicar que la instrucción ha detectado un error, tal como especificar un eje no configurado.
Bit .IP (En proceso) 26	Se establece en la transición de renglón positiva y desaparece luego de que el proceso de prueba de diagnóstico se completa, o se interrumpe mediante un comando de paro, desactivación o falla del servo.
Bit .PC (Proceso completo) 27	Se establece después de que el proceso de prueba de diagnóstico se ha ejecutado correctamente.

Descripción: La instrucción MRHD se utiliza para ejecutar diversas pruebas de diagnóstico en el eje específico para examinar la integridad y, en algunos casos, la polaridad de las conexiones de campo servo. Actualmente, hay pruebas de diagnóstico para conexión de variador, conexión del encoder, conexión del marcador y conexión del contacto OK del módulo de control de movimiento. Durante algunos de estos procesos de prueba, el módulo de movimiento genera una salida al variador externo para producir un pequeño movimiento. Las mediciones realizadas durante algunas de estas pruebas de diagnóstico de conexión se guardan como parámetros de configuración de salida que además, se utilizan como datos de entrada para una instrucción MAHD posterior. MRHD requiere sólo un parámetro de entrada explícito, la prueba de diagnóstico. Ingrese o escoja la prueba de diagnóstico para ejecutar y el eje a examinar.

Si el eje específico no aparece en la lista de ejes disponibles, dicho eje no ha sido configurado para la operación. Utilice Tag Editor para crear y configurar un nuevo eje.

La instrucción MRHD utiliza los parámetros entrada y salida para configuración del eje. Los parámetros de configuración de entrada que utiliza MRHD se indican en la siguiente tabla.

Parámetro del eje	Tipo de datos	Unidades	Definición
Motor Encoder Test Increment	Real	-	Distancia que el eje debe recorrer para cumplir con la prueba de diagnóstico de conexión.

Los parámetros de configuración del eje que genera MRHD como salida, dependen del Diagnóstico de conexión específica.

Prueba de conexión del motor encoder

Si escoge la prueba del motor encoder, el módulo de control de movimiento habilita el variador externo y genera 1 volt por segundo en la rampa de salida al variador, mientras monitorea la retroalimentación del encoder. Cuando el eje se ha movido una distancia mayor o igual al incremento de prueba de motor encoder configurado, el voltaje de la prueba regresa a cero y el variador está inhabilitado. El módulo de control de movimiento indica, luego, la dirección del recorrido que se guarda como uno de los siguientes parámetros:

Parámetro del eje	Tipo de datos	Unidades	Definición
Test Status	Número entero	-	Informe de estado del proceso de prueba de diagnóstico de conexión.
Test Direction Forward	Booleano	-	Dirección del recorrido del eje durante la prueba de conexión como establecida por el módulo de control de movimiento.

Si, debido a una conexión incorrecta, o algún otro problema con el sistema, la retroalimentación del eje no detecta que el eje ha llegado al incremento de prueba de motor encoder en 2 segundos, el servo establece el voltaje de la prueba en cero e inhabilita el variador. El control refleja esta condición mediante el parámetro de prueba de estado de salida del eje. Por lo general, esto indica que tanto el cableado al variador o el cableado al encoder son incorrectos. La ejecución de MRHD junto con la prueba seleccionada de conexión del encoder es un método efectivo de aislar el problema al encoder o variador.

Prueba de conexión del encoder

Si escoge la prueba del encoder, el módulo de control de movimiento no genera movimiento del eje, simplemente monitorea la retroalimentación de encoder del eje. Entonces, el eje se puede mover manualmente o por otro accionador de variador independiente para generar movimiento. Cuando el módulo de control de movimiento detecta que el eje se ha movido una distancia mayor o igual al incremento de prueba de motor encoder configurado, la prueba está completa. Entonces, el módulo de control de movimiento indica la dirección del recorrido como uno de los siguientes parámetros de salida de MRHD.

Parámetro del eje	Tipo de datos	Unidades	Definición
Test Status	Número entero	-	Informe de estado del proceso de prueba de diagnóstico de conexión.
Test Direction Forward	Booleano	-	Dirección del recorrido del eje durante la prueba de conexión como establecido por el módulo de control de movimiento.

Si, debido a una conexión incorrecta, o algún otro problema con el sistema, la retroalimentación del eje no detecta que el eje ha llegado al incremento de prueba de motor encoder luego de mover el eje, al menos esa distancia, entonces cancele la prueba mediante la instrucción MAS y controle el cableado del encoder.

Prueba de conexión de marcador

Si escoge la prueba del marcador, el módulo de control de movimiento no genera movimiento del eje, simplemente monitorea la retroalimentación de encoder del eje. Entonces, el eje se puede mover manualmente o por otro accionador de variador independiente para generar movimiento. Cuando el módulo de control de movimiento detecta un marcador (Canal Z), la prueba está completa. El módulo de control movimiento informa el resultado de la prueba mediante el Estado de la prueba

Parámetro del eje	Tipo de datos	Unidades	Definición
Test Status	Número entero	-	Informe de estado del proceso de prueba de diagnóstico de conexión.
Test Direction Forward	Booleano	-	Dirección del recorrido del eje durante la prueba de conexión como establecida por el módulo de control de movimiento.

Si, debido a una conexión incorrecta, o algún otro problema con el sistema, la retroalimentación del eje no detecta que el eje ha llegado al incremento de prueba de motor encoder luego de mover el eje, al menos esa distancia, entonces cancele la prueba mediante la instrucción MAS y controle el cableado del encoder.

Prueba del temporizador de control (Watchdog) OK

Si escoge la prueba del temporizador de control (Watchdog) OK, el módulo de control de movimiento no genera movimiento del eje, simplemente simula una falla del temporizador de control (Watchdog) de CPU que abre los contactos OK. Los contactos OK deben permanecer cerrados por 2 segundos. Esta prueba se utiliza para controlar el cableado de contactos OK en la cadena E-Stop del sistema de variador. En caso de fallo DSP del módulo de control de movimiento, este mecanismo se utiliza para desactivar la alimentación eléctrica del/de los variador(es). Al finalizar la prueba de temporizador de control (Watchdog) OK de dos segundos, el módulo de control de movimiento informa el estado a través del Estado de la prueba como se indica a continuación:

Parámetro del eje	Tipo de datos	Unidades	Definición
Test Status	Número entero	-	Informe de estado del proceso de prueba de diagnóstico de conexión.

Test Status

Pueden ocurrir algunas situaciones que impidan al controlador realizar correctamente la operación de prueba. En este caso, el proceso de prueba se cancela automáticamente y se informa un fallo de prueba que se almacena en el parámetro de salida del estado de prueba. También es posible cancelar manualmente un proceso de prueba utilizando una instrucción MAS lo que resulta en un fallo de prueba informado por el parámetro de estado del prueba. Los posibles valores para el estado de prueba se indican en la siguiente tabla:

Mensaje de error	Código	Definición
Test Success	0	El proceso de prueba se ha realizado correctamente.
Test In Process	1	La prueba está en progreso.
Test Aborted	2	El proceso de prueba fue cancelado por el usuario.
Test Time-out	3	El proceso de prueba ha sobrepasado el tiempo de espera (2 segundos).
Test Servo Fault	4	Fallo del proceso de prueba debido a una falla del servo.
Test Increment Fault	5	El proceso de prueba ha fallado debido a insuficiente distancia de incremento de la prueba. para realizar una medición confiable.

Para ejecutar correctamente una instrucción MRHD mediante una prueba motor encoder, el eje específico se debe configurar como un tipo de servoeje y estar en estado de Eje Preparado. Para otras pruebas, esta instrucción se ejecuta adecuadamente en un eje tipo Servo o sólo Retroalimentación. Si no se cumple con estas condiciones, la instrucción falla.

IMPORTANTE

Cuando la instrucción MRHD se ejecuta en primer lugar, se establece el bit En proceso (.IP) y se elimina el bit de Proceso completo (.PC). La ejecución de la instrucción MRHD puede tomar múltiples escanes para ejecutar ya que la misma requiere transmisión de múltiples mensajes al módulo de control de movimiento. El bit ejecutado (.DN) no se configura inmediatamente, sólo luego de que estos mensajes sean transmitidos correctamente. Se elimina el bit En proceso (.IP) y se establece el bit de Proceso completo (.PC) al mismo tiempo que el bit ejecutado (.DN).

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético: no afectados

Condiciones de fallo: ninguno

Códigos de error:

Consulte Códigos de error (ERR) para las instrucciones de control de movimiento.

Códigos de error extendidos:

Los códigos de error extendidos brindan información adicional específica de la instrucción para los códigos de error genéricos a muchas instrucciones. Los siguientes códigos de error extendidos ayudan a establecer el problema cuando la instrucción MRHD recibe un mensaje de error de fallo de mensajes al servomódulo (12).

Código de error asociado (decimal)	Código de error extendido (decimal)	Significado
SERVO_MESSAGE_FAILURE (12)	Proceso finalizado a pedido (1)	La ejecución de la prueba seguida de una instrucción para interrupción/inhabilitación del variador, o una instrucción para detener el movimiento o un cambio de procesador requiere cancelar la prueba.
SERVO_MESSAGE_FAILURE (12)	Conflicto Modo de objeto (12)	El eje está desactivado.
SERVO_MESSAGE_FAILURE (12)	Dispositivo en estado incorrecto (16)	Orden de proceso de ajuste incorrecta. (SERCOS)

Bit de estado:

MRHD Cambia a Bit de estado

Nombre del bit	Estado	Significado
DriveEnableStatus	VERDADERO	· El eje está en el estado de control del variador. · La salida habilitada del variador está activa mientras el perfil de ajuste se está ejecutando.
TestStatus	VERDADERO	El eje está ejecutando un proceso de prueba.

Ejemplo:

Cuando las condiciones de entrada son verdaderas, el controlador ejecuta la prueba de diagnóstico del *encoder* en el *eje1*.

Lógica de escalera de relés

MRHD
Motion Run Hookup Diagnostics
Axis Axis1
Motion Control MRHD_1
Diagnostic Test Marker

Ejemplo de lógica de escalera de MRHD

Texto estructurado

MRHD(Axis1,MRHD_1,Marker);

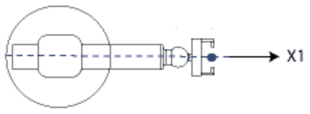
Instrucciones de movimiento coordinado

(MCLM, MCCM, MCCD, MCS, MCSD, MCT, MCTP, MCSR)

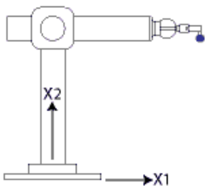
Introducción

Utilice las instrucciones de movimiento coordinado para mover los tres ejes en un sistema de coordenadas.

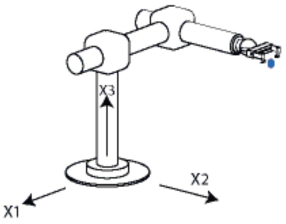
Sistemas de coordenadas con ejes ortogonales



Sistema de coordenadas cartesianas

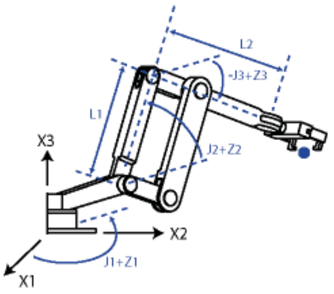


Sistema de coordenadas cartesianas bidimensional

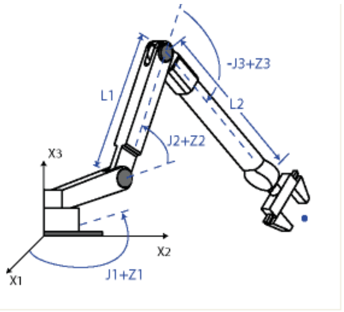


Sistema de coordenadas cartesianas tridimensional

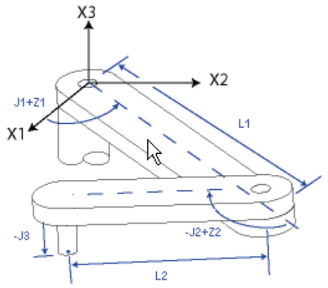
Sistemas coordinados con ejes no ortogonales



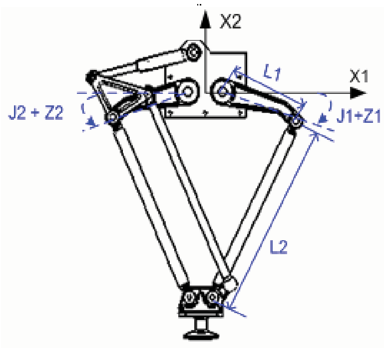
Sistema de coordenadas dependiente articulado



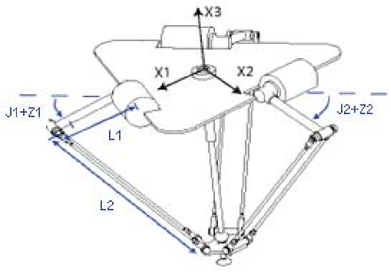
Sistema de coordenadas independiente articulado



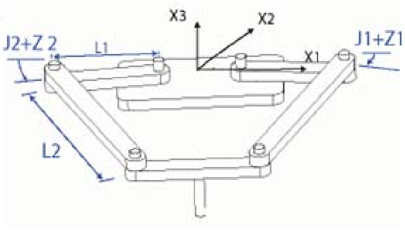
Sistema de coordenadas independiente SCARA



Sistema de coordenadas bidimensional Delta



Sistema de coordenadas tridimensional Delta



Sistema de coordenadas Delta SCARA

Utilice esta tabla para escoger una instrucción de control de movimiento coordinado.

Escoja una Instrucción de control de movimiento coordinado

Si desea	Use esta instrucción	Disponible en estos lenguajes
Iniciar un movimiento coordinado lineal sencillo o multidimensional para los ejes específicos dentro del Sistema de coordenadas cartesianas.	Motion Coordinated Linear Move (MCLM)	• Lógica de escalera de relés • Texto estructurado
Iniciar un movimiento coordinado circular bidimensional o tridimensional para los ejes específicos dentro del Sistema de coordenadas cartesianas.	Motion Coordinated Circular Move (MCCM)	• Lógica de escalera de relés • Texto estructurado
Iniciar un cambio en la dinámica de la ruta para el movimiento coordinado activo en un sistema de coordenadas específico.	Motion Coordinated Change Dynamics (MCCD)	• Lógica de escalera de relés • Texto estructurado
Detener los ejes de un sistema de coordenadas o cancelar una transformación.	Motion Coordinated Stop (MCS)	• Lógica de escalera de relés • Texto estructurado
Iniciar una interrupción controlada de todos los ejes del sistema de coordenadas específico.	Motion Coordinated Shutdown (MCSD)	• Lógica de escalera de relés • Texto estructurado
Iniciar una transformación que vincula dos sistemas coordinados juntos.	Motion Coordinated Transform (MCT)(1)	• Lógica de escalera de relés • Texto estructurado
Calcular la posición de un sistema de coordenadas con respecto a otro sistema de coordenadas.	Motion Calculate Transform Position (MCTP)(1)	• Lógica de escalera de relés • Texto estructurado
Restablecer todos los ejes de un sistema de coordenadas específico desde el estado de interrupción al estado de eje preparado y borrar los fallos del eje.	Motion Coordinated Shutdown Reset (MCSR)	• Lógica de escalera de relés • Texto estructurado

(1) Sólo puede usar esta instrucción con los controladores 1756-L6*x*.

Utilice diferentes tipos de terminación con instrucciones de combinación

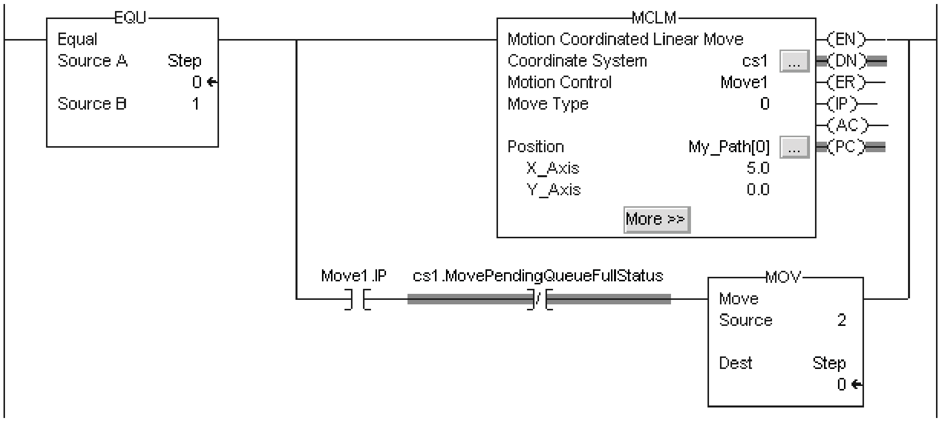
Para combinar dos instrucciones MCLM o MCCM, inicie la primera y ponga la segunda en la cola. El tag para el sistema de coordenadas proporciona dos bits para instrucciones de formación de colas.

- MovePendingStatus
- MovePendingQueueFullStatus

Por ejemplo, el siguiente diagrama de lógica de escalera utiliza el sistema de coordenadas cs1 para combinar Move1 en Move2.

Ejemplo de diagrama de lógica de escalera para instrucciones de combinación

Si Step = 1, entonces
  Move1 arranca y mueve los ejes a una posición de 5, 0
Y una vez que Move1 está en proceso
Y hay espacio para poner otro movimiento en cola
  Step = 2


EQU
Equal
Source A Step
0
Source B 1

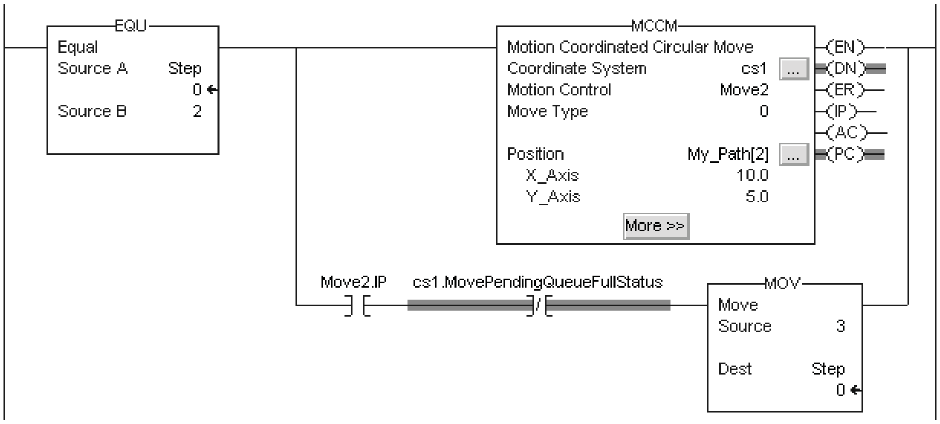
MCLM
Motion Coordinated Linear Move
Coordinate System cs1
Motion Control Move1
Move Type 0
Position My_Path[0]
X_Axis 5.0
Y_Axis 0.0

Move1.IP
cs1.MovePendingQueueFullStatus

MOV
Move
Source 2
Dest Step
0

Continua en la siguiente página

Si Step = 2, entonces
  Move1 ya se está ejecutando.
  Move2 ingresa en la cola y espera que Move1 finalice.
  Cuando Move1 finaliza, Move2 mueve los ejes a una posición de 10, 5.
  Y una vez que Move2 esté en proceso y hay espacio en la cola,
  Step = 3


EQU
Equal
Source A Step
0
Source B 2

MCCM
Motion Coordinated Circular Move
Coordinate System cs1
Motion Control Move2
Move Type 0
Position My_Path[2]
X_Axis 10.0
Y_Axis 5.0

Move2.IP
cs1.MovePendingQueueFullStatus

MOV
Move
Source 3
Dest Step
0

Cuando una instrucción finaliza, se quita de la cola y hay espacio para que otra instrucción ingrese a la cola. Ambos bits siempre tienen el mismo valor ya que se puede poner en cola sólo una instrucción pendiente a la vez. Si la aplicación requiere que varias instrucciones se ejecuten en secuencia, entonces, los bits se establecen usando estos parámetros.

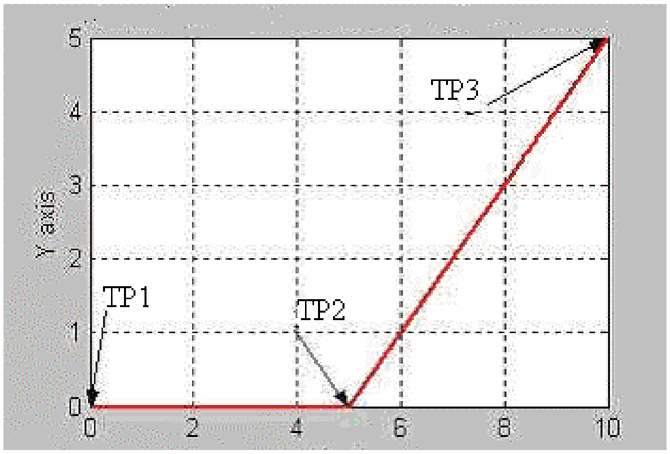
Parámetros de bit

Cuando	Entonces
Una instrucción está activa y una segunda instrucción está pendiente en la cola	• Bit MovePendingStatus = 1 • Bit MovePendingQueueFullStatus = 1 • No se puede poner otra instrucción en la cola
Una instrucción activa se completa y sale de la cola	• Bit MovePendingStatus = 0 • Bit MovePendingQueueFullStatus = 0 • Se puede poner otra instrucción en la cola

El operando de tipo de terminación para la instrucción MCLM o MCCM indica cómo se interrumpe el movimiento ejecutado actualmente. Las siguientes ilustraciones demuestran los estados de los bits de instrucción y los bits del sistema de coordenadas afectados en diferentes puntos de transición (TP).

Estados de bit en los puntos de transición del movimiento combinado usando Tolerancia Real o No Establecido

lineal → movimiento lineal



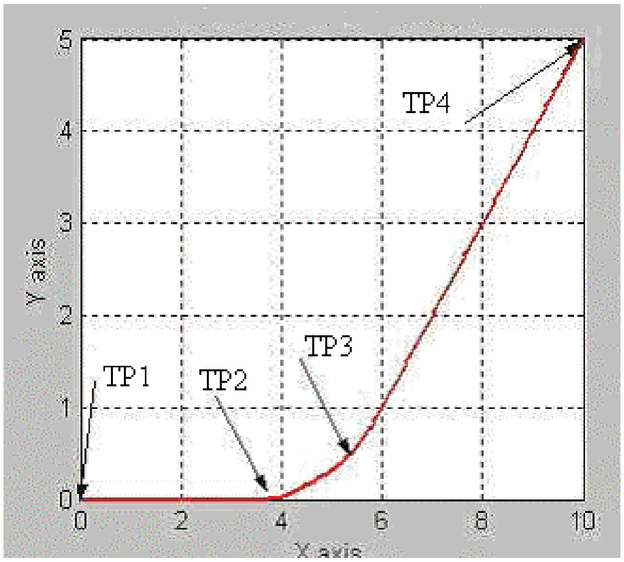
La siguiente tabla indica el estado de bit en diversos puntos de transición señalados en el diagrama anterior con el tipo de terminación de Tolerancia Real o No Establecido.

Estado de bit en los puntos de transición con Tipo de terminación de Tolerancia Real o No Establecido

Bit	TP1	TP2	TP3
Move1.DN	T	T	T
Move1.IP	T	F	F
Move1.AC	T	F	F
Move1.PC	F	T	T
Move2.DN	T	T	T
Move2.IP	T	T	F
Move2.AC	F	T	F
Move2.PC	F	F	T
cs1.MoveTransitionStatus	F	F	F
cs1.MovePendingStatus	T	F	F
cs1.MovePendingQueueFullStatus	T	F	F

Estados de bit en los puntos de transición de movimiento combinado usando No Decel

lineal → movimiento lineal



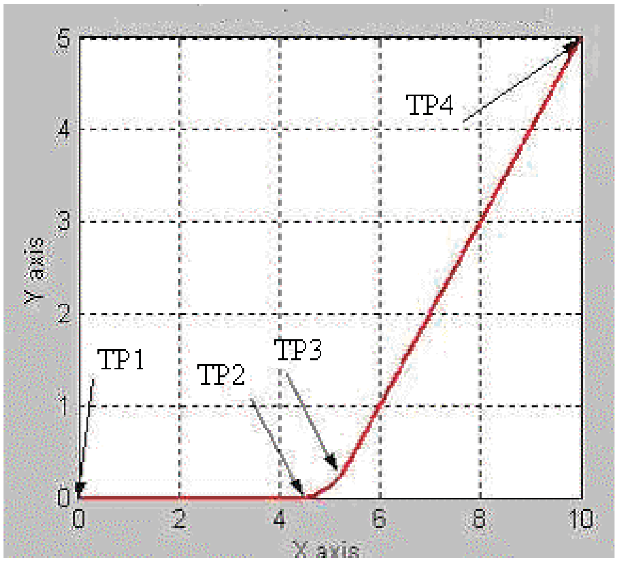
La siguiente tabla indica el estado de bit en diversos puntos de transición señalados en el diagrama anterior con el tipo de terminación de No Decel. Para el tipo de terminación No Decel la distancia a recorrer por el punto de transición TP2 es igual a la distancia de desaceleración para la instrucción Move1. Si Move 1 y Move 2 son colineales, entonces Move1.PC será verdadero en TP3 (el punto extremo programado del primer movimiento).

Estado de bit con tipo de terminación No Decel

Bit	TP1	TP2	TP3	TP4
Move1.DN	T	T	T	T
Move1.IP	T	F	F	F
Move1.AC	T	F	F	F
Move1.PC	F	T	T	T
Move2.DN	T	T	T	T
Move2.IP	T	T	T	F
Move2.AC	F	T	T	F
Move2.PC	F	F	F	T
cs1.MoveTransitionStatus	F	T	F	F
cs1.MovePendingStatus	T	F	F	F
cs1.MovePendingQueueFullStatus	T	F	F	F

Estados de bit en los puntos de transición de movimiento combinado usando Tolerancia de Comando

lineal → movimiento lineal



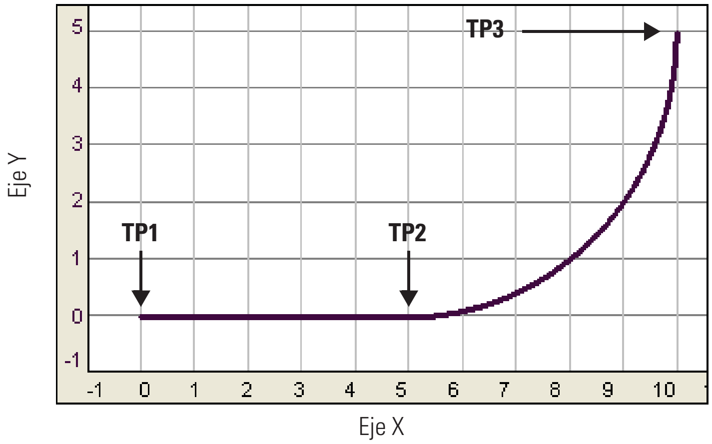
La siguiente tabla indica el estado de bit en diversos puntos de transición señalados en el diagrama anterior con el tipo de terminación de Tolerancia de Comando. Para el tipo de terminación de Tolerancia de Comando la distancia a recorrer por el punto de transición TP2 es igual a Tolerancia de Comando para el sistema de coordenadas cs1.

Estado de bit con tipo de terminación Tolerancia de Comando

Bit	TP1	TP2	TP3	TP4
Move1.DN	T	T	T	T
Move1.IP	T	F	F	F
Move1.AC	T	F	F	F
Move1.PC	F	T	T	T
Move2.DN	T	T	T	T
Move2.IP	T	T	T	F
Move2.AC	F	T	T	F
Move2.PC	F	F	F	T
cs1.MoveTransitionStatus	F	T	F	F
cs1.MovePendingStatus	T	F	F	F
cs1.MovePendingQueueFullStatus	T	F	F	F

Estados de bit en los puntos de transición de movimiento combinado usando Velocidad de contorno continua limitada y no limitada

lineal → movimiento circular



La siguiente tabla indica el estado de los bits en los puntos de transición.

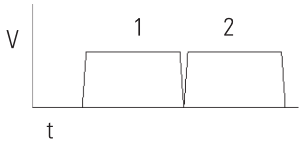
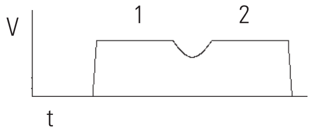
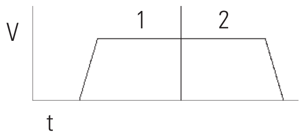
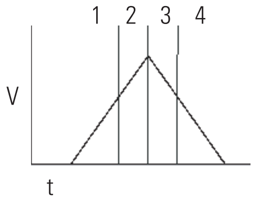
Estado del bit con tipo de terminación de velocidad de contorno limitada y no limitada

Bit	TP1	TP2	TP3
Move1.DN	T	T	T
Move1.IP	T	F	F
Move1.AC	T	F	F
Move1.PC	F	T	T
Move2.DN	T	T	T
Move2.IP	T	T	F
Move2.AC	F	T	F
Move2.PC	F	F	T
cs1.MoveTransitionStatus	F	F	F
cs1.MovePendingStatus	T	F	F
cs1.MovePendingQueueFullStatus	T	F	F

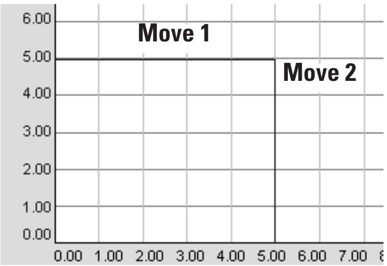
Escoja un tipo de terminación

El tipo de terminación indica cuando la instrucción está completa. También determina cómo la instrucción combina su ruta en la instrucción MCLM o MCCM en cola, si hay alguna.

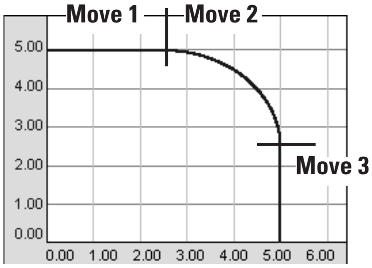
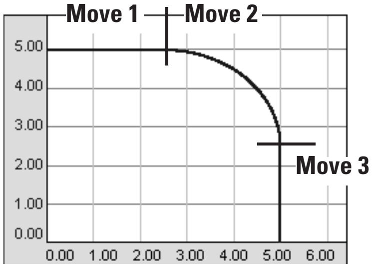
1. Escoja un tipo de terminación.

Si desea que los ejes (velocidades del vector)	Y desea que la instrucción finalice cuando	Entonces, utilice este tipo de terminación
Detenerse entre movimientos 	Ambas opciones ocurren: • La posición de comando es igual a la posición establecida. • La distancia del vector entre las posiciones establecida y real es menor o igual a la Tolerancia de posición real del sistema de coordenadas.	0 – Tolerancia real
	La posición de comando es igual a la posición establecida.	1 – No Establecido
Mantenga velocidad constante **excepto** entre movimientos 	La posición de comando se ubica dentro de la Tolerancia de posición de comando del sistema de coordenadas.	2 – Tolerancia de comando
	Los ejes alcanzan el punto en el que deben reducir la velocidad a la tasa de desaceleración.	3 – No Decel
Transición dentro o fuera del circulo sin detener 		4 – Velocidad de contorno continua limitada
Acelerar o desacelerar a través de múltiples movimientos 		5 – Velocidad de contorno continua no limitada

2. Asegúrese que esta es la opción correcta para Usted.

Tipo de terminación	Ejemplo de ruta	Descripción
0 – Tolerancia real		La instrucción permanece activa hasta que ambas opciones ocurran: • La posición de comando es igual a la posición establecida. • La distancia del vector entre las posiciones establecida y real es menor o igual a la Tolerancia de posición real del sistema de coordenadas. En este punto, la instrucción finaliza y la instrucción MCLM o MCCM en la cola puede comenzar. **Importante:** Asegúrese que configuró la Tolerancia real a un valor que sus ejes puedan alcanzar. De lo contrario, la instrucción permanece en proceso.
1 – No Establecido		La instrucción permanece activa hasta que la posición de comando es igual a la posición establecida. En este punto, la instrucción finaliza y la instrucción MCLM o MCCM en la cola puede comenzar.
2 – Tolerancia de comando		La instrucción permanece activa hasta que la posición de comando se ubica dentro de la Tolerancia de comando del sistema de coordenadas. En este punto, la instrucción finaliza y la instrucción MCLM o MCCM en la cola puede comenzar. Si no hay una instrucción MCLM o MCCM en cola, el eje se detiene en la posición establecida.

El software RSLogix 5000 compara	A	Y utiliza	Para
El 100% de la longitud configurada de la primera instrucción usando el tipo de terminación de Tolerancia de Comando	La tolerancia de comando configurada para el sistema de coordenadas	La más corta de las dos longitudes	La longitud de Tolerancia de comando utilizada para la **primera** instrucción
El 100% de la longitud configurada de la última instrucción usando el tipo de terminación de Tolerancia de Comando	La tolerancia de comando configurada para el sistema de coordenadas	La más corta de las dos longitudes	La longitud de Tolerancia de comando utilizada para la **penúltima** instrucción
El 50% de cada una de las longitudes de todas las otras instrucciones de movimiento	La tolerancia de comando configurada para el sistema de coordenadas	La más corta de las dos longitudes	La longitud de Tolerancia de comando utilizada para **cada** instrucción **individual**

Tipo de terminación	Ejemplo de ruta	Descripción
3 – No Decel		La instrucción permanece activa hasta que los ejes alcanzan el punto de desaceleración. En este punto, la instrucción finaliza y la instrucción MCLM o MCCM en la cola puede comenzar. • El punto de desaceleración depende del uso de un perfil trapezoidal o con curva en S. • Si no hay una instrucción MCLM o MCCM en cola, el eje se detiene en la posición establecida.
4 – Velocidad de contorno continua limitada		La instrucción permanece activa hasta que los ejes alcanzan la posición establecida. En este punto, la instrucción finaliza y la instrucción MCLM o MCCM en la cola puede comenzar. • Este tipo de terminación funciona mejor con transiciones tangenciales. Por ejemplo, para ir de una línea a un círculo, de un círculo a una línea, o de un círculo a un círculo. • Los ejes siguen la ruta. • La longitud del movimiento determina la velocidad máxima de los ejes. Si los movimientos son lo suficientemente largos, los ejes no reducirán la velocidad entre los movimientos. Si los movimientos son demasiado cortos, los ejes reducen la velocidad entre los movimientos.
5 – Velocidad de contorno continua no limitada		Este tipo de terminación es similar a la velocidad de contorno limitada. Tiene estas diferencias: • Utilice este tipo de terminación para alcanzar un perfil de velocidad triangular a través de diversos movimientos. Esto reduce los movimientos bruscos. • Para evitar sobreimpulso de posición al final del último movimiento, debe calcular la velocidad de desaceleración en cada punto de transición durante la desaceleración a mitad del perfil. • Debe calcular la velocidad de arranque para cada movimiento en la desaceleración a mitad del perfil.

Consideraciones importantes

Si Ud. detiene un movimiento (es decir, usando MCS o cambiando la velocidad a cero con MCCD) durante una combinación y luego, reanuda el movimiento (es decir, reprogramando el movimiento o usando otra MCCD), se desviará de la ruta que hubiera seguido si el movimiento no hubiese sido detenido y reanudado. El mismo fenómeno puede ocurrir si el movimiento se desarrolla dentro del punto de desaceleración del comienzo de la combinación. En cada caso, la desviación probablemente será una pequeña desviación.

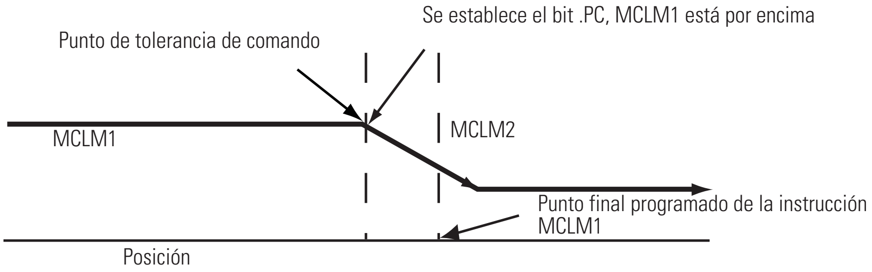
Perfiles de velocidad para movimientos colineales

Los movimientos colineales son aquellos que se encuentran en la misma línea en el área. Su dirección puede ser la misma u opuesta. Los perfiles de velocidad para movimientos colineales pueden ser complejos. En esta sección encontrará ejemplos e ilustraciones que ayudarán a comprender los perfiles de velocidad para movimientos colineales programados con instrucciones MCLM.

Perfiles de velocidad para movimientos colineales con tipo de terminación 2

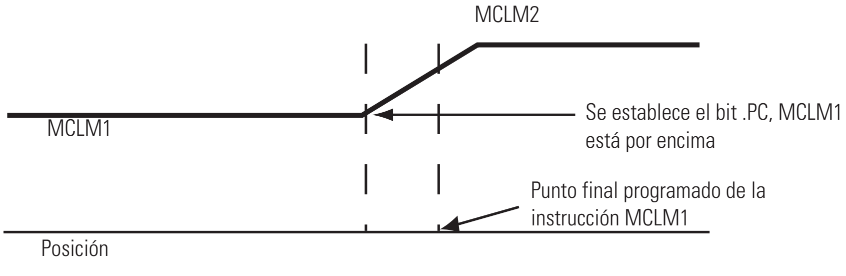
La siguiente ilustración muestra el perfil de velocidad de dos movimientos colineales usando un tipo de terminación de Tolerancia de comando (2). La segunda instrucción MCLM tiene **menor** velocidad que la primera instrucción MCLM. Cuando la primera instrucción MCLM alcanza su punto de Tolerancia de comando, el movimiento es por encima y se establece el bit .PC.

Perfil de velocidad de dos movimientos colineales cuando el segundo movimiento tiene menor velocidad que el primer movimiento y se utiliza el Tipo de terminación 2



La siguiente ilustración muestra el perfil de velocidad de dos movimientos colineales usando un tipo de terminación de Tolerancia de comando (2). La segunda instrucción MCLM tiene **mayor** velocidad que la primera instrucción MCLM. Cuando la primera instrucción MCLM alcanza su punto de Tolerancia de comando, el movimiento es por encima y se establece el bit .PC.

Perfil de velocidad de dos movimientos colineales cuando el segundo movimiento tiene mayor velocidad que el primer movimiento y se utiliza el Tipo de terminación 2



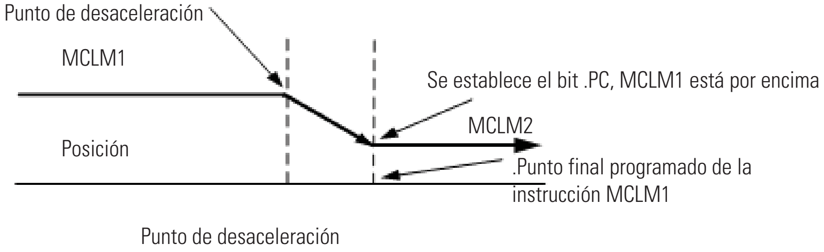
Perfiles de velocidad para movimientos colineales con Tipos de terminación 3, 4, ó 5

Esta ilustración muestra un perfil de velocidad de dos movimientos colineales. La segunda instrucción MCLM tiene una velocidad **menor** que la primera instrucción MCLM y se utiliza uno de estos tipos de terminación:

- No Decel (3)
- Velocidad de contorno continua limitada (4)
- Velocidad de contorno continua no limitada (5)

Cuando la primera instrucción MCLM alcanza el punto de desaceleración, reduce la velocidad a la velocidad programada del segundo movimiento. El primer movimiento es por encima y se establece el bit .PC.

Perfil de velocidad de dos movimientos colineales cuando el segundo movimiento tiene menor velocidad que el primer movimiento y se utiliza el Tipo de terminación 3, 4, ó 5

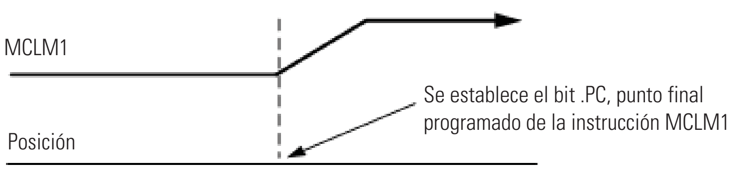


Esta ilustración muestra un perfil de velocidad de dos movimientos colineales. La segunda instrucción MCLM tiene una velocidad **mayor** que la primera instrucción MCLM y se utiliza uno de estos tipos de terminación:

- No Decel (3)
- Velocidad de contorno continua limitada (4)
- Velocidad de contorno continua no limitada (5)

Se establece el bit .PC cuando el primer movimiento alcanza su punto final programado.

Perfil de velocidad de dos movimientos colineales cuando el segundo movimiento tiene mayor velocidad que el primer movimiento y se utiliza el Tipo de terminación 3, 4, ó 5



Perfiles simétricos

Las rutas del perfil son simétricas para todos los perfiles de movimiento.

La programación de los valores de velocidad, aceleración y desaceleración simétricamente en las direcciones de avance y en retroceso genera la misma ruta desde el punto A al punto C en la dirección de avance, como desde el punto C al punto A en la dirección en retroceso.

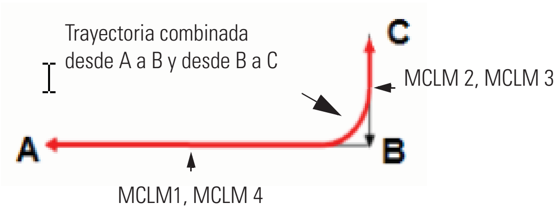
Mientras este concepto se muestra más fácilmente en una secuencia de dos instrucciones, el mismo se aplica a las secuencias de instrucción de cualquier longitud siempre y cuando estén programados simétricamente.

Consulte el Ejemplo de un perfil simétrico para más detalles.

Ejemplo de un perfil simétrico

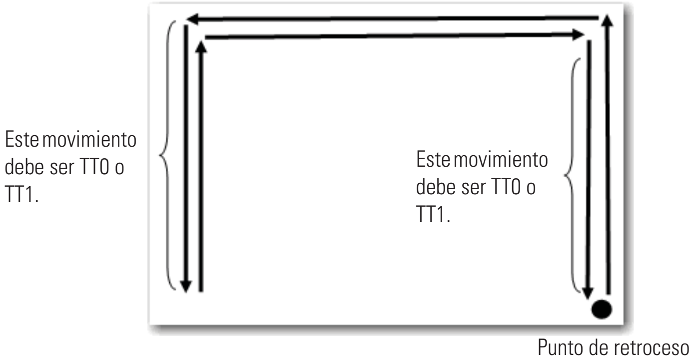
- MCLM 1 (punto A al punto B) seguido por MCLM 2 (punto B al punto C).
- MCLM 3 (punto C al punto B) seguido por MCLM 4 (punto B al punto A).
- La aceleración de MCLM 1 debe ser igual a la desaceleración de MCLM 4.
- La desaceleración de MCLM 1 debe ser igual a la aceleración de MCLM 4.
- La aceleración de MCLM 2 debe ser igual a la desaceleración de MCLM 3.
- La desaceleración de MCLM 2 debe ser igual a la aceleración de MCLM 3.

MCLM 1 (Pos = [2,0], Accel = 1, Decel = 2)
MCLM 2 (Pos = [2,1], Accel = 3, Decel = 4)
MCLM 3 (Pos = [2,0], Accel = 4, Decel = 3)
MCLM 4 (Pos = [0,0], Accel = 2, Decel = 1)



IMPORTANTE Recomendamos interrumpir cualquier secuencia de movimiento mediante el Tipo de terminación 0 ó 1; es decir, TT0 o TT1.

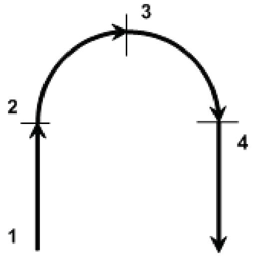
Para garantizar una trayectoria simétrica, debe interrumpir cualquier secuencia de movimiento mediante Tipos de terminación 0 ó 1. También debe usar un Tipo de terminación 0 ó 1 en el Punto de retroceso de un perfil que retrocede sobre sí mismo.



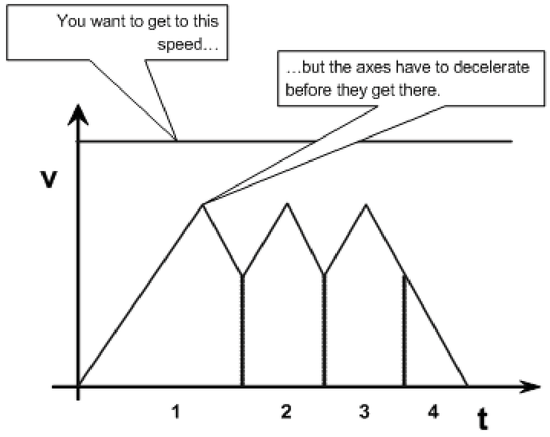
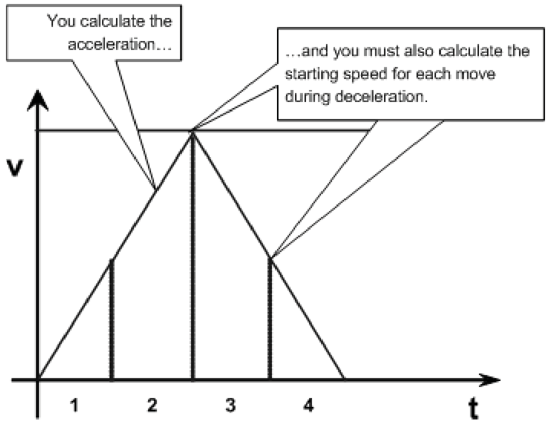
Es seguro utilizar TT2, TT3, TT4 o TT5 como el último movimiento en un perfil (o el punto de retroceso). Sin embargo, la trayectoria final desde A a B no siempre puede ser la misma que desde B a A. La terminación explícita de la secuencia de movimiento ayuda al controlador a optimizar el perfil de velocidad, a reducir la carga de la CPU, y a garantizar un perfil simétrico.

Cómo obtener un Perfil de velocidad triangular

Si desea programar una recogida y colocación de acciones en cuatro movimientos, minimizar el régimen de movimientos bruscos, y utilizar un perfil de velocidad triangular.

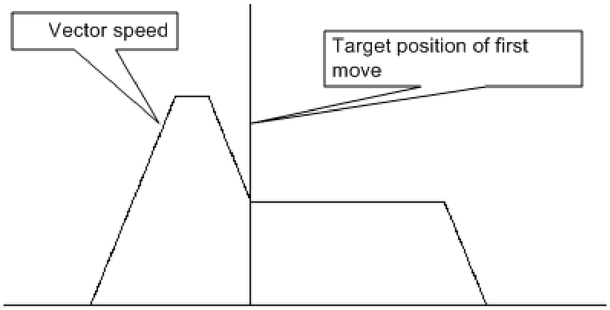
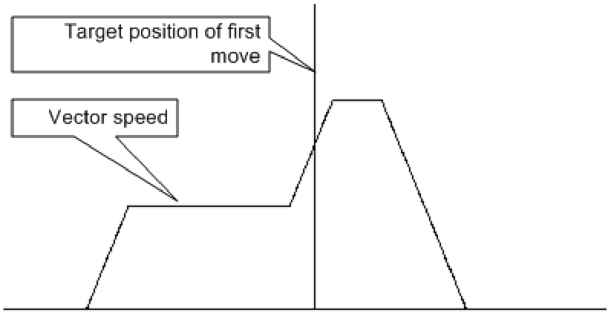
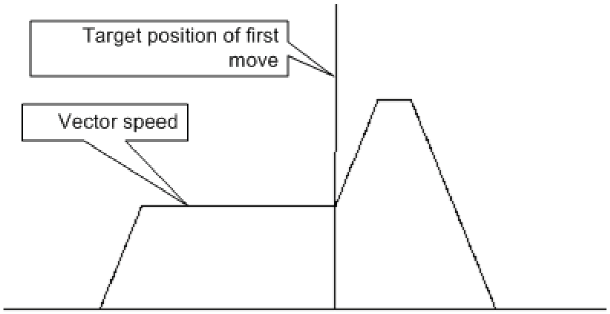


Entonces, utilice el tipo de terminación 5. Los demás tipos de terminación no permiten alcanzar la velocidad deseada.

Tipos de terminación 2, 3, ó 4	Tipo de terminación 5
	
La longitud de dicho movimiento determina su velocidad máxima. Como resultado, los ejes no alcanzarán una velocidad que provoque sobreimpulsos en la posición establecida durante la desaceleración.	Los ejes aceleran a la velocidad deseada. Debe calcular la velocidad de arranque para cada movimiento en la desaceleración a mitad del perfil.

Movimientos combinados en diferentes velocidades

Puede combinar las instrucciones MCLM y MCCM cuando la velocidad del vector de la segunda instrucción sea diferente a la velocidad del vector de la primera instrucción.

Si el próximo movimiento es	Y el tipo de terminación del primer movimiento es	Entonces
Más lento	2 – Tolerancia de comando 3 – No Decel 4 – Velocidad de contorno limitada 5 – Velocidad de contorno no limitada	
Más rápido	2 – Tolerancia de comando 3 – No Decel	
	4 – Velocidad de contorno limitada 5 – Velocidad de contorno no limitada	

Motion Coordinated Linear Move (MCLM)

Utilice la instrucción MCLM para iniciar un movimiento coordinado lineal simple o multi dimensional para los ejes específicos dentro de un sistema de coordenadas cartesianas. Puede definir la nueva posición como absoluta o incremental.

ATENCIÓN

Los tags para instrucciones de control de movimiento se deben utilizar sólo una vez. Reutilizar el tag de control de movimiento en otras instrucciones puede provocar operaciones imprevistas. Esto puede dañar el equipo o causar lesiones personales.

ATENCIÓN

Riesgo de sobreimpulso de velocidad y/o de posición final

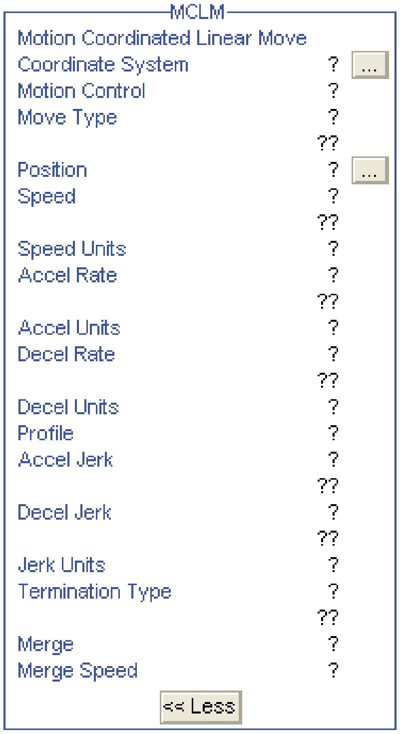
Si cambia los parámetros de movimiento dinámicamente mediante cualquier método, es decir cambiando la dinámica del movimiento (MCD o MCCD) o iniciando una nueva instrucción antes de que la última finalice, tenga en cuenta el riesgo de sobreimpulso de velocidad y/o de posición final.

Se puede exceder un perfil de velocidad trapezoidal si la máxima desaceleración se disminuye mientras el movimiento reduce la velocidad o está cerca del punto de desaceleración.

Se puede exceder un perfil de velocidad con curva en S si:

- la máxima desaceleración disminuye mientras el movimiento reduce la velocidad o está cerca del punto de desaceleración; o

- la máxima aceleración de jaloneo disminuye y el eje acelera. Recuerde, sin embargo, que el jaloneo se puede cambiar indirectamente si se encuentra especificado en % de tiempo.



MCLM
Motion Coordinated Linear Move
Coordinate System ?
Motion Control ?
Move Type ?
??
Position ?
Speed ?
??
Speed Units ?
Accel Rate ?
??
Accel Units ?
Decel Rate ?
??
Decel Units ?
Profile ?
Accel Jerk ?
??
Decel Jerk ?
??
Jerk Units ?
Termination Type ?
??
Merge ?
Merge Speed ?

La instrucción MCLM realiza un movimiento lineal usando hasta tres (3) ejes agrupados estáticamente como ejes primarios en un sistema de coordenadas cartesianas. Especifique si debe utilizar una posición establecida absoluta o incremental, la velocidad deseada, la aceleración máxima, la desaceleración máxima, el jaloneo de aceleración, el jaloneo de desaceleración y las unidades de cada uno. La velocidad real es una función de las unidades programadas de la velocidad (Unidades por seg, o % del Máximo, como configurado para el sistema de coordenadas), y la combinación de ejes primarios que reciben la orden de moverse. Cada eje recibe la orden de moverse a una velocidad que permite a todos los ejes alcanzar el punto final programado (posición establecida) al mismo tiempo.

Operandos – Lógica de escalera de relé

Operando	Tipo	Formato	Descripción
Sistema de coordenadas	COORDINATE_SYSTEM	Tag	Grupo de ejes coordinado.
Motion Control	MOTION_ INSTRUCTION	Tag	Estructura utilizada para acceder a los parámetros de estado de la instrucción.
Move Type	SINT, INT o DINT	Inmediato o tag	Escoja el tipo de movimiento: 0 = Absoluto 1 = Incremental
Position	REAL	Tag de matriz [ ]	[unidades de coordinación]
Speed	SINT, INT, DINT o REAL	Inmediato o tag	[unidades de coordinación]
Speed Units	SINT, INT o DINT	Inmediato	0 = Unidades por seg 1 = % del Máximo
Accel Rate	SINT, INT, DINT o REAL	Inmediato o tag	[unidades de coordinación]
Accel Units	SINT, INT o DINT	Inmediato	0 = Unidades por seg^2 1 = % del Máximo
Decel Rate	SINT, INT, DINT o REAL	Inmediato o tag	[unidades de coordinación]
Decel Units	SINT, INT o DINT	Inmediato	0 = Unidades por seg^2 1 = % del Máximo
Profile	SINT, INT o DINT	Inmediato	0 = Trapezoidal 1 = Curva en S

Operandos – Lógica de escalera de relé

Operando	Tipo	Formato	Descripción
Accel Jerk	SINT, INT, DINT o REAL	Inmediato o tag	**Siempre** debe ingresar los valores para los operandos Accel y Decel Jerk. Esta instrucción sólo utiliza los valores si el operando Profile está configurado como curva en S. Introduzca los regímenes de jaloneo en estas unidades de jaloneo. 0 = Unidades por seg^3 1 = % del Máximo 2 = % de tiempo Utilice estos valores para arrancar. • Accel Jerk = 100 (% del tiempo) • Decel Jerk = 100 (% del tiempo) • Unidades de jaloneo = 2
Decel Jerk	SINT, INT, DINT o REAL	Inmediato o tag	
Jerk Units	SINT, INT o DINT	Inmediato o tag	
Tipo de terminación	SINT, INT o DINT	Inmediato o tag	0 = Tolerancia real 1 = No Establecido 2 = Tolerancia de comando 3 = No Decel 4 = Velocidad de contorno continua limitada 5 = Velocidad de contorno continua no limitada Consulte Escoja un tipo de terminación en la página 264.
Merge	SINT, INT o DINT	Inmediato	0 = Inhabilitado 1 = Movimiento coordinado 2 = Todos los movimientos
Merge Speed	SINT, INT o DINT	Inmediato	0 = Programado 1 = Corriente

Texto estructurado

MCLM(CoordinateSystem,
MotionControl,MoveType,
Position,Speed,Speedunits,Accelrate,Accelunits,
Decelrate,Decelunits,Profile,Acceljerk,Deceljerk,
Jerkunits,TerminationType,Merge,
Mergespeed);

Los operandos para el texto estructurado son iguales que para la instrucción MCLM de lógica de escalera de relé.

Cuando ingresa enumeraciones para el valor del operando en el texto estructurado, las enumeraciones de múltiples palabras se **deben** ingresar sin espacios. Por ejemplo: cuando ingresa Decel Units, el valor se debe ingresar como unitspersec^2 en lugar de Unidades por Seg^2 como se indica en la lógica de escalera.

Utilice las entradas en esta tabla como guía para ingresar los operandos del texto estructurado.

Entradas para Operandos del texto estructurado

Este operando	**Cuenta con estas opciones que usted ingresa como…**	
	Texto	**O como**
Sistema de coordenadas	Sin enumeración	Tag
Motion Control	Sin enumeración	Tag
Move Type	Sin enumeración	0 (Absoluto) 1 (Incremental)
Position	Sin enumeración	Tag de matriz
Velocidad	Sin enumeración	Inmediato o tag
Speed Units	Unidades por seg % del máximo	0 1
Accel Rate	Sin enumeración	Inmediato o tag
Accel Units	Unidades por seg^2 % del máximo	0 1
Decel Rate	Sin enumeración	Inmediato o tag
Decel Units	Unidades por seg^2 % del máximo	0 1
Profile	Trapezoidal Curva S	0 1
Accel Jerk	Sin enumeración	Inmediato o tag **Siempre** debe ingresar un valor para los operandos Accel y Decel Jerk. Esta instrucción sólo utiliza los valores si Profile está configurado como curva en S. Utilice estos valores para arrancar. • Accel Jerk = 100 (% del tiempo) • Decel Jerk = 100 (% del tiempo) • Unidades de jaloneo = 2
Decel Jerk	Sin enumeración	

Entradas para Operandos del texto estructurado

Este operando	Cuenta con estas opciones que usted ingresa como...	
	Texto	**O como**
Jerk Units	Unitspersec3 %ofmaximum %oftime	0 1 2 (utilice este valor para comenzar)
Tipo de terminación	Sin enumeración	0 = Tolerancia real 1 = No Establecido 2 = Tolerancia de comando 3 = No Decel 4 = Velocidad de contorno continua limitada 5 = Velocidad de contorno continua no limitada Consulte Escoja un tipo de terminación en la página 264.
Merge	Inhabilitado (Disabled) Coordinatedmotion Allmotion	0 1 2
Merge Speed	Programada Actual	0 1

Sistema de coordenadas

El operando del sistema de coordenadas indica el conjunto de los ejes de movimiento que definen las dimensiones de un sistema de coordenadas cartesianas. Para esta versión, el sistema de coordenadas acepta hasta tres (3) ejes primarios. Sólo aquellos ejes configurados como ejes primarios se incluyen en los cálculos de velocidad coordinada.

Control de movimiento

Los siguientes bits de control están afectados por la instrucción MCLM.

Bits de control afectados por la instrucción MCLM

Mnemónico	Descripción
Bit .EN (Habilitado) 31	El bit Habilitado se establece durante la transición del renglón de falso a verdadero y se restablece cuando el renglón pasa de verdadero a falso.
Bit .DN (Listo) 29	El bit Listo se establece cuando la instrucción coordinada ha sido verificada y puesta en cola correctamente. Ya que se establece al mismo tiempo que entra en cola, puede aparecer como establecido cuando un error de tiempo de ejecución aparece durante la operación de verificación luego de salir de la cola. Se restablece cuando el renglón va de falso a verdadero.
Bit .ER (Error) 28	El bit Error se restablece cuando las transiciones del renglón cambian de falso a verdadero. Se establece cuando el movimiento coordinado no se ha iniciado correctamente. También se establece con el Bit Listo cuando una instrucción en cola identifica un error de tiempo de ejecución.
Bit .IP (En proceso) 26	El bit En proceso se establece cuando el movimiento coordinado se inicia correctamente. Se restablece cuando no hay movimiento subsiguiente y el movimiento coordinado alcanza la nueva posición, o cuando hay un movimiento subsiguiente y el movimiento coordinado alcanza las especificaciones del tipo de terminación, o cuando el movimiento coordinado es reemplazado por otra instrucción MCLM o MCCM con un tipo de incorporación de movimiento coordinado, o cuando se finaliza con una instrucción MCS.
Bit .AC (Activo) 23	Cuando tiene una instrucción de movimiento coordinado en cola, el bit Activo permite conocer qué instrucción está controlando el movimiento. Se establece cuando el movimiento coordinado se activa. Se restablece cuando el bit de Proceso completo se establece o cuando la instrucción se detiene.
Bit .PC (Proceso completo) 27	El bit Proceso Completo se establece cuando las transiciones del renglón cambian de falso a verdadero. Se establece cuando no hay movimiento subsiguiente y el movimiento coordinado alcanza una nueva posición, o cuando hay un movimiento subsiguiente y el movimiento coordinado alcanza el tipo de terminación específico.
Bit .ACCEL (Bit Aceleración) 01	El bit Aceleración se establece mientras el movimiento coordinado está en la fase de aceleración. Se restablece mientras el movimiento coordinado está en velocidad constante o fase de desaceleración, o cuando el movimiento coordinado finaliza.
Bit .DECEL (Bit Desaceleración) 02	El bit Desaceleración se establece mientras el movimiento coordinado está en la fase de desaceleración. Se restablece mientras el movimiento coordinado está en velocidad constante o en fase de aceleración, o cuando el movimiento coordinado finaliza.

Move Type

El operando Move Type indica el método utilizado para indicar la ruta del movimiento coordinado. Move Type puede ser Absoluto o Incremental.

- Absoluto – los ejes se mueven en un ruta lineal a la posición definida por la matriz de posición a la Velocidad (Speed), régimen de aceleración (Accel Rate) y régimen de desaceleración (Decel Rate) especificado por los operandos.

  Cuando el eje es configurado para operación giratoria, un tipo de movimiento absoluto funciona de la misma manera que para un eje lineal. Cuando la posición del eje excede el parámetro de desbobinado, este se desbobina. De esta manera, la posición del eje nunca es mayor que el valor de desbobinado ni menor a cero.

  La señal de la posición específica se interpreta mediante el interpolador y puede ser positiva o negativa. Los valores de posición negativos ordenan al interpolador que mueva el eje giratorio en una dirección negativa para obtener la posición absoluta deseada, mientras los valores positivos indican que el movimiento positivo es necesario para alcanzar la posición establecida. Cuando el valor de posición es mayor al valor de desbobinado, se genera un error. El eje nunca se desplaza más de un ciclo de desbobinado antes de detenerse en una posición absoluta.

- Incremental – el sistema de coordenadas se mueve la distancia definida por la matriz de posición a la velocidad específica, utilizando los regímenes de Aceleración y Desaceleración determinado por los operandos correspondientes, mediante una ruta lineal.

  La distancia específica se interpreta mediante el interpolador y puede ser positiva o negativa. Los valores de posición negativos ordenan al interpolador que mueva el eje en una dirección negativa, mientras los valores positivos indican movimiento positivo para alcanzar la posición establecida. Se permite un movimiento mayor que un ciclo desbobinado en modo Incremental.

Ejemplos de Move Type MCLM

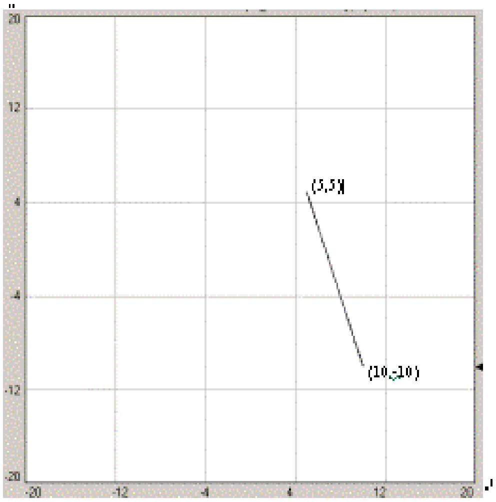
Los siguientes ejemplos indican el uso de MCLM con Tipo de movimiento (Move Type) Absoluto (primer ejemplo) e Incremental (segundo ejemplo) para llegar al mismo resultado. Los supuestos básicos son:

- los dos ejes, Axis0 y Axis1, son miembros del sistema de coordenadas, coordinate_sys.
- Axis0 y Axis1 son ortogonales entre sí.
- Coordinated_sys se inicia a (5,5) unidades.

Mueve Coordinated_sys de manera lineal a (10,–10) unidades a la velocidad vectorial de 10.0 unidades por segundo con los valores de aceleración y desaceleración de 5.0 unidades por segundo^2.

El siguiente diagrama es la ruta generada por los supuestos anteriormente mencionados.

Diagrama resultante de la ruta



La distancia total recorrida a lo largo de la ruta del vector es:

DAxis0 = 10 – 5 = 5

DAxis1 = –10 – 5 = –15

$$TotalDist = \sqrt{(DAxis0)^2 \acute{Y} (DAxis1)^2} = 15.811388$$

La velocidad vectorial de los ejes seleccionados es igual a la velocidad establecida en las unidades de posición por segundo. La velocidad de cada eje es proporcional a la distancia recorrida por el eje dividido por la raíz cuadrada de la suma de los cuadrados de la distancia recorrida por los ejes. La velocidad real del Axis0 es el siguiente porcentaje de la velocidad vectorial del movimiento.

%Axis0 Velocidad = |Daxis0/TotalDist| = |5/15.811388| = .3162 = 31.62%

%Axis1 Velocidad = |Daxis1/TotalDist| = |–15/15.811388| = .9487 = 94.87%

Para el ejemplo,

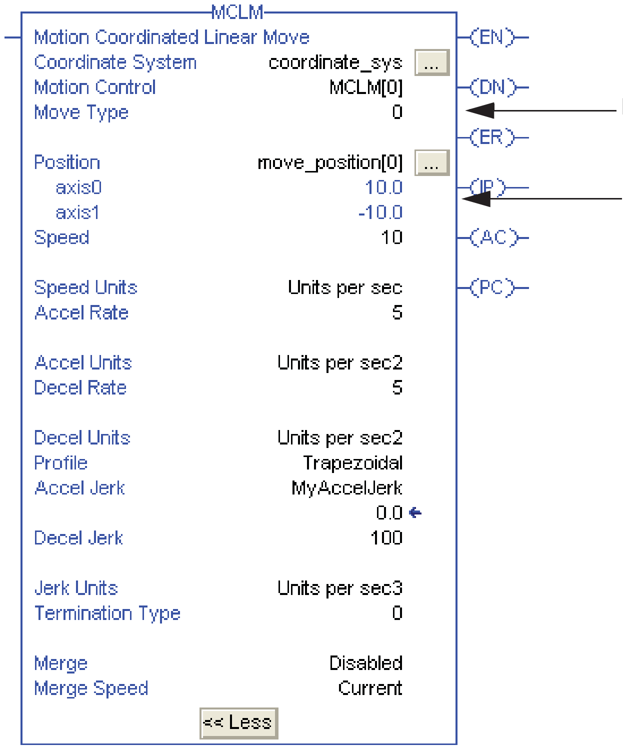
Axis0 Velocidad = .3162 * 10.0 = 3.162 unidades/seg.

Axis1 Velocidad = .9487 * 10.0 = 9.487 unidades/seg.

La aceleración y desaceleración para cada eje es el mismo porcentaje como la velocidad.

Las siguientes instrucciones de lógica de escalera indica la lógica de escalera necesaria para lograr esta ruta usando Move Type = Absolute y Move Type = Incremental, respectivamente.

Instrucción de lógica de escalera MCLM con Move Type Absolute



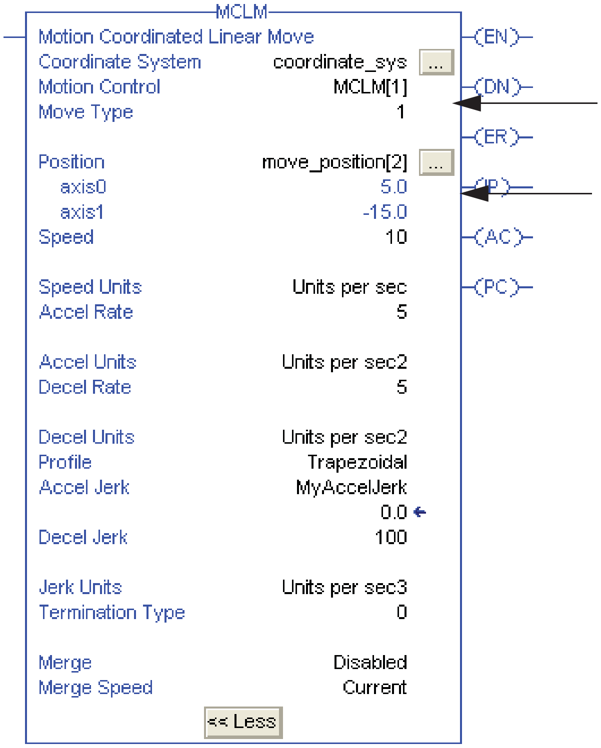
MCLM
Motion Coordinated Linear Move

Coordinate System	coordinate_sys
Motion Control	MCLM[0]
Move Type	0
Position	move_position[0]
axis0	10.0
axis1	-10.0
Speed	10
Speed Units	Units per sec
Accel Rate	5
Accel Units	Units per sec2
Decel Rate	5
Decel Units	Units per sec2
Profile	Trapezoidal
Accel Jerk	MyAccelJerk 0.0
Decel Jerk	100
Jerk Units	Units per sec3
Termination Type	0
Merge	Disabled
Merge Speed	Current

Move Type es Absolute

Posición definida en unidades absolutas.

Instrucción de lógica de escalera MCLM con Move Type Incremental



Motion Coordinated Linear Move	
Coordinate System	coordinate_sys
Motion Control	MCLM[1]
Move Type	1
Position	move_position[2]
axis0	5.0
axis1	-15.0
Speed	10
Speed Units	Units per sec
Accel Rate	5
Accel Units	Units per sec2
Decel Rate	5
Decel Units	Units per sec2
Profile	Trapezoidal
Accel Jerk	MyAccelJerk 0.0 ←
Decel Jerk	100
Jerk Units	Units per sec3
Termination Type	0
Merge	Disabled
Merge Speed	Current

Move Type es Incremental

La posición definida como la distancia incremental desde el punto de arranque de (5,5).

Ejemplos de instrucción MCLM con ejes giratorios

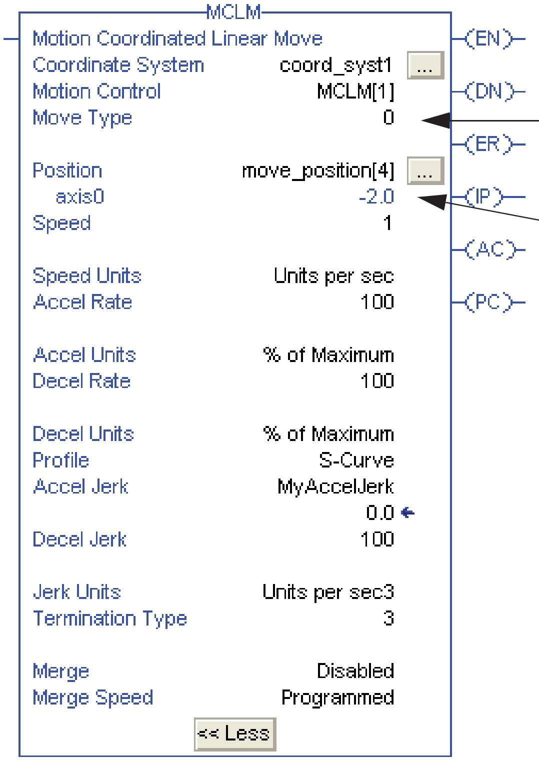
Los siguientes ejemplos indican el diagrama de las rutas para instrucciones MCLM que tienen ejes definidos como Giratorio.

MCLM con un Eje giratorio y Move Type Absolute

El primer ejemplo utiliza un sistema de coordenadas de un eje y un Move type Absolute. El diagrama de la ruta se basa en los siguientes supuestos:

- 1 eje del sistema de coordenadas llamado coord_syst1
- Axis0 es Giratorio con un desbobinado de 5 revoluciones.
- La posición de arranque es 4.
- La posición final es –2.

Instrucción de lógica de escalera MCLM con Move Type Absolute



MCLM

Motion Coordinated Linear Move

Coordinate System coord_syst1

Motion Control MCLM[1]

Move Type 0

Position move_position[4]

axis0 -2.0

Speed 1

Speed Units Units per sec

Accel Rate 100

Accel Units % of Maximum

Decel Rate 100

Decel Units % of Maximum

Profile S-Curve

Accel Jerk MyAccelJerk

0.0 ←

Decel Jerk 100

Jerk Units Units per sec3

Termination Type 3

Merge Disabled

Merge Speed Programmed

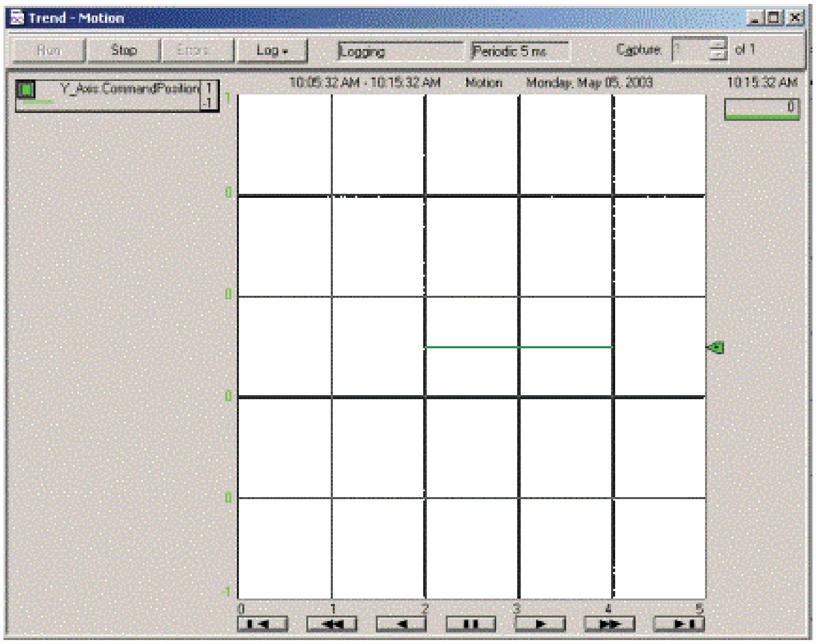
Move Type es Absoluto.

El punto final se define como negativo.

Recuerde que para los tipos de movimiento Absolute (0), la señal negativa denota la dirección del movimiento. En este ejemplo, el eje se mueve a una posición absoluta de +2.0 en la dirección negativa. Para mover a una posición de 0.0 en la dirección negativa, debe programar –360.0, ya que –0.0 se almacena internamente como 0.0.

El diagrama resultante de la ruta del movimiento se indica en la siguiente ilustración.

Diagrama de MCLM con un Eje giratorio y Move Type Absolute



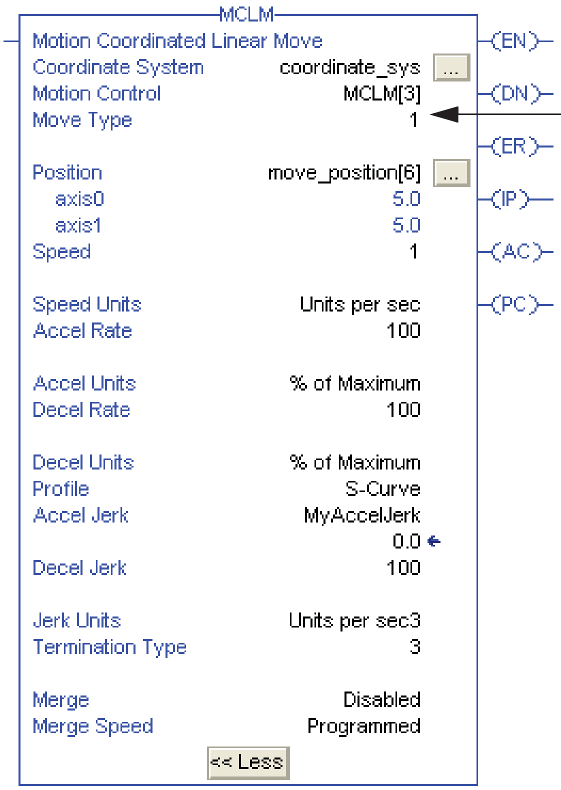
El punto final tenía un valor negativo, por lo tanto, el eje recorrió en una dirección negativa moviendo desde 4 a 2. No recorrió a través del desbobinado. Para este movimiento, se requiere el punto final para encajar dentro de la posición absoluta definida por el desbobinado giratorio del eje. Por lo tanto, un valor desbobinado de 6 o –6 no sería válido.

MCLM con dos Ejes giratorios y Move Type Incremental

El segundo ejemplo MCLM con ejes giratorios tiene dos ejes giratorios y un Move Type Incremental. El diagrama de la ruta tiene los siguientes supuestos:

- Dos ejes del sistema de coordenadas llamados coordinate_sys
- Axis0 es Giratorio con un desbobinado de 1 revolución.
- Axis1 es Giratorio con un desbobinado de 2 revoluciones.
- La posición de arranque es 0,0.
- Incremento a la posición final es 5,5.

Instrucción de lógica de escalera MCLM con Move Type Incremental

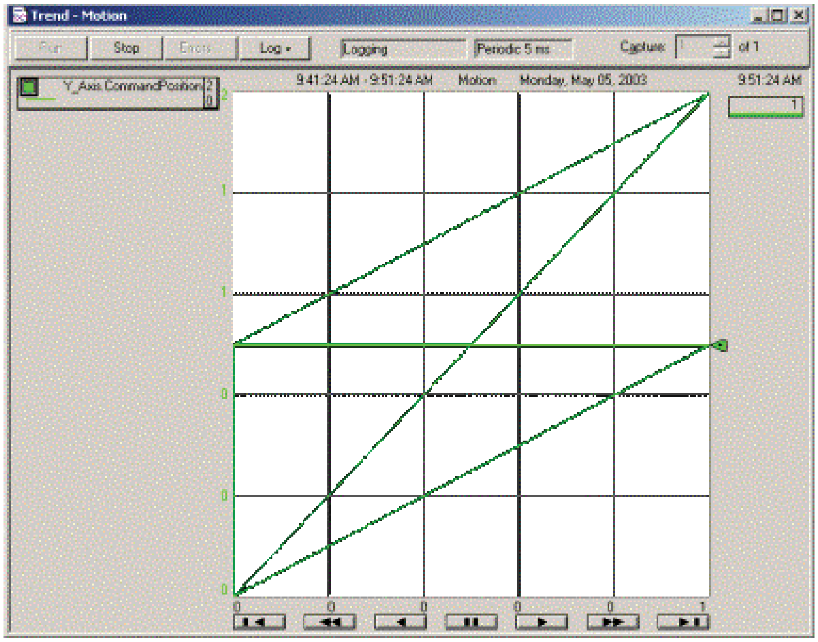


Motion Coordinated Linear Move	
Coordinate System	coordinate_sys
Motion Control	MCLM[3]
Move Type	1
Position	move_position[6]
axis0	5.0
axis1	5.0
Speed	1
Speed Units	Units per sec
Accel Rate	100
Accel Units	% of Maximum
Decel Rate	100
Decel Units	% of Maximum
Profile	S-Curve
Accel Jerk	MyAccelJerk
	0.0 ←
Decel Jerk	100
Jerk Units	Units per sec3
Termination Type	3
Merge	Disabled
Merge Speed	Programmed

Move Type es Incremental.

Esta instrucción MCLM produce el siguiente diagrama de la ruta de movimientos.

Diagrama de MCLM con dos Ejes giratorios y Move Type Incremental



En el diagrama anterior, los ejes recorren un patrón “z” en retroceso dos veces y media. Se detienen a una posición real de 0,1. Esto equipara a 5 revoluciones/desbobinados para Axis0 y 2.5 revoluciones/desbobinados para Axis1. Los incrementos de posición para este movimiento son positivos, por lo tanto, los ejes se mueven en una dirección positiva con Axis0 moviendo desde 0 a 1 y Axis1 moviendo desde 0 a 2. En este ejemplo, el punto final no es necesario para encajar dentro de la posición absoluta definida por el desbobinado giratorio de los ejes. La ruta del movimiento coordinado está determinada en un espacio lineal, pero la posición de los ejes está limitada por la configuración giratoria.

Position

Una matriz unidimensional, cuya dimensión está definida para ser, al menos, el equivalente del número de ejes indicados en el sistema de coordenadas. La matriz Position define ya sea la nueva posición absolute o incremental.

Speed

El operando Speed define la velocidad vectorial máxima a lo largo de la ruta del movimiento coordinado.

Speed Units

El operando Speed Units define las unidades aplicadas al operando Speed ya sea directamente en unidades de coordinación del sistema de coordenadas establecida o como un porcentaje de los valores máximos definidos en el sistema de coordenadas.

Accel Rate

El operando Accel Rate define la aceleración máxima a lo largo de la ruta del movimiento coordinado.

Accel Units

El operando Accel Units define las unidades aplicadas al operando Accel Rate ya sea directamente en unidades de coordinación del sistema de coordenadas establecido o como un porcentaje de los valores máximos definidos en el sistema de coordenadas.

Decel Rate

El operando Decel Rate define la desaceleración máxima a lo largo de la ruta del movimiento coordinado.

Decel Units

El operando Decel Units define las unidades aplicadas al operando Decel Rate ya sea directamente en unidades de coordinación del sistema de coordenadas establecido o como un porcentaje de los valores máximos definidos en el sistema de coordenadas.

Profile

El operando Profile determina si el movimiento utiliza un perfil trapezoidal o con curva en S.

El controlador de movimiento ControlLogix proporciona perfiles de velocidad trapezoidal (aceleración y desaceleración lineal) y con curva en S (jaloneo controlado). A continuación encontrará una guía para los efectos de estos perfiles de movimiento para diversos requisitos de aplicación.

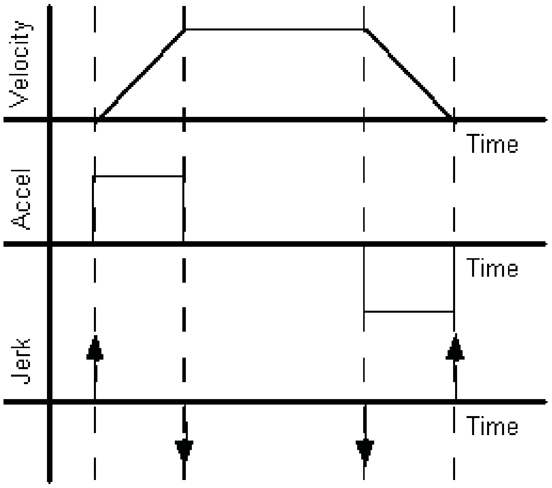
Efectos del perfil de velocidad

Perfil Tipo	**ACEL/DESACEL** Tiempo	**Motor** Tensión	**Prioridad de control** **De más alta a más baja**			
Trapezoidal	Más rápida	Peor	Acel/Desacel	Velocidad	Position	
Curva S	2X más lento	Mejor	Jaloneo	Acel/Desacel	Velocidad	Position

- Trapezoidal

El perfil de velocidad trapezoidal es el perfil utilizado más frecuentemente ya que proporciona mayor flexibilidad a la hora de programar movimientos subsiguientes y tiempos de aceleración y desaceleración más cortos. El cambio máximo en velocidad se especifica por la aceleración y desaceleración. Ya que el jaloneo no es un factor para perfiles trapezoidales, el mismo se considera infinito y se indica como una serie de líneas verticales en el siguiente gráfico.

Tiempo Aceleración/Desaceleración trapezoidal



- Curva S

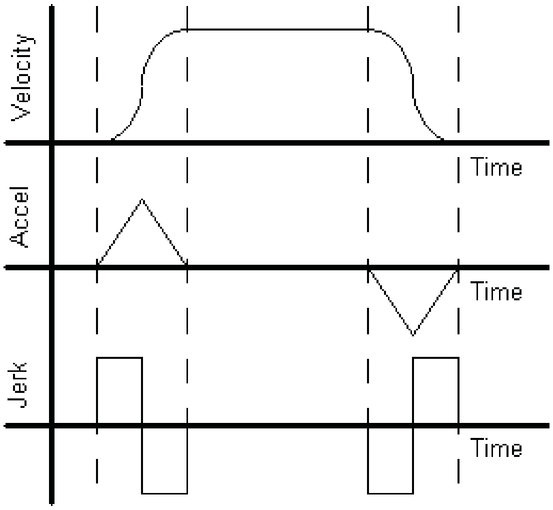
Los perfiles de velocidad de curva en S son los utilizados más frecuentemente cuando es necesario minimizar la carga y el esfuerzo en el sistema mecánico. Sin embargo, el perfil de curva en S, sacrifica el tiempo de aceleración y desaceleración comparado con el trapezoidal. El régimen máximo en el cual se puede acelerar o desacelerar la velocidad es, además, limitado por el jaloneo.

La cálculos del régimen de jaloneo de aceleración y desaceleración del movimiento coordinado se realizan una vez que estas instrucciones han comenzado.

- MAJ
- MAM
- MAS
- MCD
- MCS
- MCCD
- MCCM
- MCLM

El régimen de jaloneo calculado produce perfiles de aceleración y desaceleración triangular, como se indica en el siguiente diagrama.

Tiempo de Acel/Decel de curva en S



Para un movimiento de curva en S, el régimen de jaloneo está determinado de acuerdo a los valores programados de velocidad, aceleración y desaceleración, **no a la longitud del movimiento**. El software RSLogix 5000 trata de mantener el régimen de jaloneo constante cuando los movimientos combinados que tienen los mismos valores de aceleración y desaceleración, aunque el movimiento no puede ser lo suficientemente extenso para alcanzar la velocidad programada (movimiento con velocidad limitada).

Si un movimiento de curva en S se configura como	Entonces, al incrementar la velocidad
Sin velocidad limitada	**Disminuye** el tiempo de ejecución del movimiento
Con velocidad limitada	**Aumenta** el tiempo de ejecución del movimiento

Para movimientos de curva en S que están programados con velocidad cero, el régimen de jaloneo está determinado por el régimen de velocidad programado para la instrucción previa con una velocidad no-cero.

Consulte la instrucción MCCD para más detalles sobre los cambios del impacto producidos por una instrucción MCCD.

Accel Jerk

Accel Jerk define el jaloneo de aceleración máximo para el movimiento programado. Para más información sobre cálculos de Accel Jerk, consulte la sección de Unidades de jaloneo en este capítulo.

Decel Jerk

Decel Jerk define el jaloneo de desaceleración máximo para el movimiento programado. Para más información sobre cálculos de Decel Jerk, consulte la sección de Unidades de jaloneo en este capítulo.

Jerk Units

Jerk units define las unidades que se aplican a los valores ingresados en los operandos Accel Jerk y Decel Jerk. Los valores se ingresan directamente en las unidades de posición del sistema de coordenadas o como un porcentaje. Al configurar utilizando % de Máximo, el jaloneo se aplica como porcentaje de los operando Maximum Acceleration Jerk y Maximum Deceleration Jerk establecidos en los atributos del sistema de coordenadas. Al configurar utilizando % de Tiempo, el valor es un porcentaje basado en Speed, Accel Rate, y Decel Rate establecidos en la instrucción.

Si desea convertir las unidades de ingeniería a % de Tiempo o convertir % de Tiempo a unidades de ingeniería, utilice las ecuaciones indicadas al inicio de página 28.

Tipo de terminación

Para información sobre selección de Tipos de terminación, consulte página 264.

Merge

El operando Merge determina si cambiar o no el movimiento de todos los ejes específicos en un movimiento coordinado puro. Las opciones de Merge incluyen: Merge inhabilitado, movimiento coordinado, o Todo el movimiento.

- Merge Inhabilitado

  Cualquier instrucción de movimiento de un eje simple ejecutada actualmente que involucre cualquier eje definido en el sistema de coordenadas especifico no es afectada por la activación de esta instrucción, y trae como resultado el movimiento superpuesto en los ejes afectados. Además, cualquier instrucción de movimiento coordinado que involucre el mismo sistema de coordenadas se ejecuta hasta finalizar de acuerdo al tipo de terminación.

- Movimiento coordinado

  Cualquier instrucción de movimiento coordinado que involucre el mismo sistema de coordenadas especificado ha sido interrumpida. El movimiento activo se combina en el movimiento de corriente a la velocidad definida en el parámetro de merge speed. Cualquier instrucción de movimiento coordinado pendiente se cancela. Cualquier instrucción de movimiento de un eje simple ejecutada actualmente que involucre cualquier eje definido en el sistema de coordenadas especifico no será afectada por la activación de esta instrucción, y traerá como resultado el movimiento superpuesto en los ejes afectados.

- Todo el movimiento

  Cualquier instrucción de control de movimiento de un eje simple ejecutada actualmente que involucre cualquier eje definido en el sistema de coordenadas específico y cualquier instrucción de control de movimiento coordinado ejecutada actualmente han sido interrumpidas. El movimiento previo se incorpora en el movimiento actual a la velocidad definida en el parámetro de Merge Speed. Cualquier instrucción de movimiento coordinado pendiente se cancela.

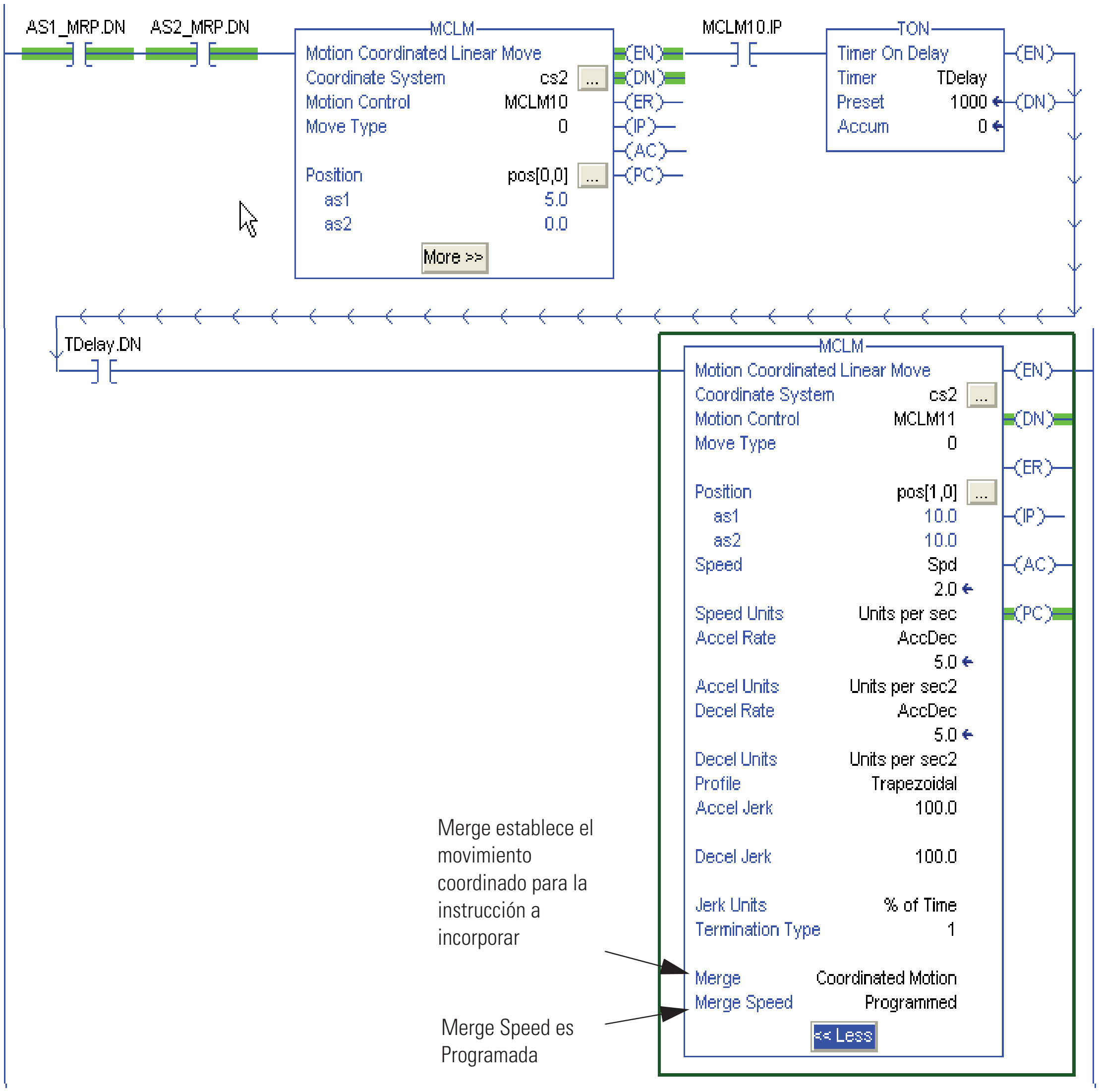
Merge Speed

El operando Merge Speed define si la velocidad actual o la velocidad programada se utiliza como velocidad máxima a lo largo de la ruta del movimiento coordinado cuando Merge es habilitado.

Ejemplo de Merge

El diagrama de lógica de escalera de MCLM utiliza el Sistema de coordenadas cs2 para incorporar una instrucción mclm10 con una posición absoluta establecida de (5,0) en una instrucción mclm11 con una posición establecida de (10,5).

Diagrama de lógica de escalera que indica Merge



MCLM
Motion Coordinated Linear Move
Coordinate System cs2
Motion Control MCLM10
Move Type 0
Position pos[0,0]
as1 5.0
as2 0.0

TON
Timer On Delay
Timer TDelay
Preset 1000
Accum 0

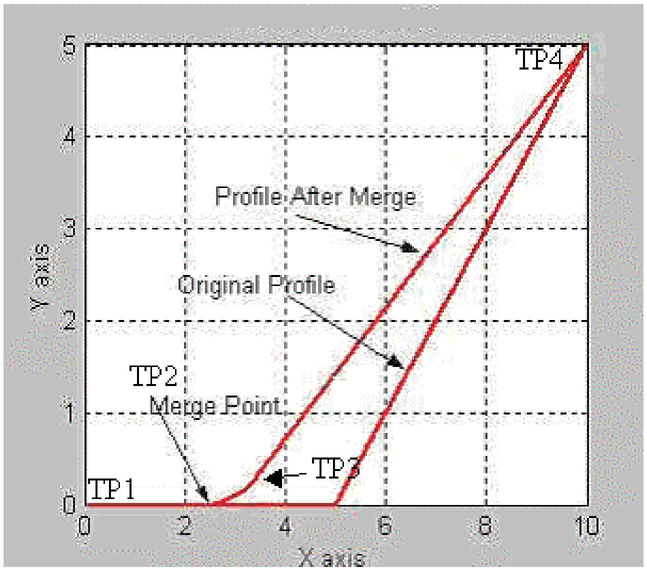
MCLM
Motion Coordinated Linear Move
Coordinate System cs2
Motion Control MCLM11
Move Type 0
Position pos[1,0]
as1 10.0
as2 10.0
Speed Spd
2.0
Speed Units Units per sec
Accel Rate AccDec
5.0
Accel Units Units per sec2
Decel Rate AccDec
5.0
Decel Units Units per sec2
Profile Trapezoidal
Accel Jerk 100.0
Decel Jerk 100.0
Jerk Units % of Time
Termination Type 1
Merge Coordinated Motion
Merge Speed Programmed

Merge establece el movimiento coordinado para la instrucción a incorporar

Merge Speed es Programada

Si los ejes son ortogonales entre sí, y el sistema de coordenadas cs2 se inicia a (0,0) unidades, entonces, el movimiento provocado por este diagrama depende del tiempo en el cual se ejecutará la segunda instrucción. La combinación comienza tan pronto como comienza el segundo movimiento y el primer movimiento se interrumpe inmediatamente. En el diagrama de lógica de escalera para este ejemplo, la transición comienza cuando el temporizador Tdelay expira.

Gráfico que indica el resultado de Merge



Estado de bit en diferentes puntos de transición para el movimiento Merge

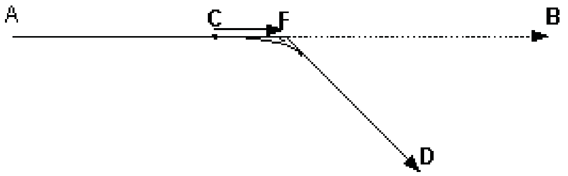
Bit	TP1	TP2	TP3	TP4
Move1.DN	T	T	T	T
Move1.IP	T	F	F	F
Move1.AC	T	F	F	F
mcclm10.PC	F	T	T	T
Move2.DN	T	T	T	T
Move2.IP	T	T	T	F
Move2.AC	F	T	T	F
Move2.PC	F	F	F	T
cs2.MoveTransitionStatus	F	T	F	F
cs2.MovePendingStatus	T	F	F	F
cs2.MovePendingQueueFullStatus	T	F	F	F

Actualmente, el Movimiento coordinado sólo acepta poner en cola una instrucción de control de movimiento coordinado. Por lo tanto, el bit MovePendingStatus y el bit MovePendingQueueFullStatus son siempre los mismos.

Información adicional sobre instrucciones de incorporación

Un movimiento desde el punto A al punto B se inicia según se indica en la siguiente figura. Cuando el eje está en el punto C, se inicia una incorporación incremental al punto D. Como resultado, la instrucción actual finaliza en el punto C. El control calcula la distancia de desaceleración necesaria en el punto C a lo largo del vector AB desde la velocidad actual a la velocidad cero. Esta distancia se indica como vector CF. El punto imaginario F se calcula sumando el vector CF al punto C. El movimiento de incorporación resultante desde C a D se indica en la siguiente ilustración. El movimiento sigue una línea curva desde C que luego une la línea recta desde F a D. El punto D se calcula desde el punto original de incorporación (punto C) utilizando el dato incremental en la instrucción de incorporación. Esta ruta es idéntica a la que se obtendría si el movimiento original programado fuera desde el punto A al F y luego desde F a D con tipo de terminación No Decel.

Ejemplo de Merge



Este ejemplo se aplica a las incorporaciones lineales.

Al intentar incorporar un movimiento circular, se puede causar un error de programación si la ruta resultante no define un círculo. El círculo central en modo incremental se calcula desde el punto C (el punto de la incorporación). Sin embargo, un círculo debe existir desde el punto F (el final de la desaceleración calculada) al final del movimiento incorporado.

Cómo incorporar en modo incremental

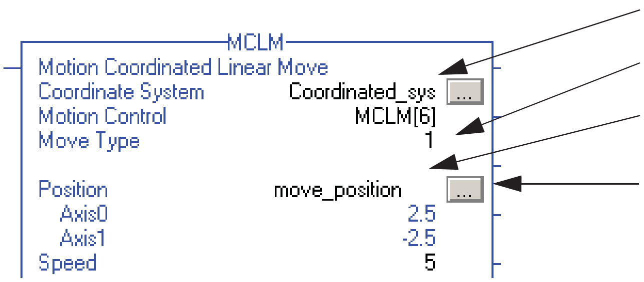
La incorporación para movimientos coordinados funciona de manera diferente que en una MAM Para MCLM, todo movimiento incompleto en el momento de la incorporación se descarta. Por ejemplo, supongamos que tiene un eje simple con MCLM programada en modo incremental desde una posición absoluta de inicio = 0 y con la distancia incremental programada = 4. Si tiene lugar una incorporación a una posición absoluta de 1 y la incorporación es otro movimiento incremental de 4 unidades, el movimiento se completa a una posición = 5.

Si este ejemplo tiene lugar en una MAM programada en modo incremental, la posición final = 8. Para obtener más información sobre cómo tiene lugar esta incorporación en una MAM programada en modo incremental, consulte página 102.

Diálogo de entrada de la posición especifica de MCLM

El Diálogo de entrada de la posición especifica para la instrucción MCLM brinda un formato sencillo para editar la Posición. Para obtener acceso al cuadro de diálogo de entrada de la posición especifica, debe introducir el nombre del sistema de coordenadas en la instrucción, debe tener un nombre de tag válido en el campo de posición con los elementos suficientes para manipular los ejes, y debe seleccionar un Move Type válido.

Para acceder al cuadro de diálogo de entrada de la posición especifica de la instrucción MCLM, presione el botón elipsis después de la línea de posición en la plantilla de instrucción.



Coordinate System Coordinated_sys
Motion Control MCLM[6]
Move Type 1
Position move_position
Axis0 2.5
Axis1 -2.5
Speed 5

Sistema de coordenadas válido

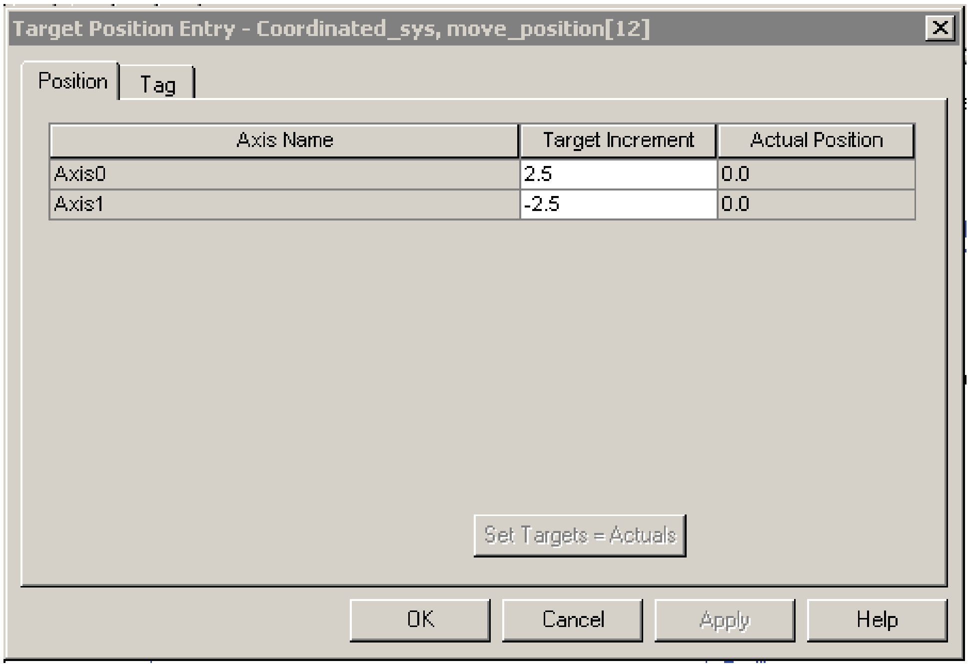
Tipo de movimiento válido

Matriz de posición válida

Presione el botón elipsis para acceder al cuadro de entrada de la posición especifica de MCLM

Valores válidos de la lógica de escalera de MCLM para acceder al cuadro de entrada de la posición especifica

Al presionar el botón elipsis en la línea de posición de la plantilla de la instrucción de la lógica de escalera, se solicita el siguiente cuadro de entrada de la posición especifica para editar los valores de posición.



Axis Name	Target Increment	Actual Position
Axis0	2.5	0.0
Axis1	-2.5	0.0

Diálogo de entrada de la posición especifica de la instrucción MCLM – Ficha de Posición

El título del diálogo indica los nombres del Sistema de coordenadas y del Tag para la instrucción.

Descripción de campo del diálogo de entrada de la posición especifica

Característica	Descripción
Nombre del eje	Estos campos enumeran los nombres de cada eje contenidos en el Sistema de Coordenadas. No puede alterar los nombres del eje en este diálogo.
Posición específica/ Incremento específico	Este campo contiene el punto final o incremento del movimiento coordinado como se estableció en la plantilla de la instrucción. Es numérico.
Posición real	Estas son las posiciones reales actuales de los ejes en el sistema de coordenadas. Estas posiciones se actualizan dinámicamente cuando la actualización del tag del sistema de coordenadas y en línea se habilita.
Objetivos configurados = Botón Actuals	Este botón copia automáticamente los valores de posición real a la columna de posición especifica.

El Move type seleccionado controla la apariencia y disponibilidad de los Objetivos configurados = Botón Actuals.

Cuando Move Type es Absolute, la columna especifica está autorizada para la Posición especifica y cuando Move Type es Incremental, la columna especifica está autorizada para el Incremento especifico y los Objetivos configurados = Botón Actuals está inhabilitado (Difuminados).

MCLM es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético

no afectados

Condiciones de fallo

ninguno

Códigos de error

Consulte Códigos de error (ERR) para las instrucciones de control de movimiento.

Códigos de error extendidos

Los códigos de error extendido ayudan a definir, además, el mensaje de error dado por esta instrucción particular. Este comportamiento depende del código de error al cual son asociados.

Los códigos de error extendido para Error de estado servo desactivado (5), Error del estado de desactivación (7), Tipo de eje no servo (8), Eje no configurado (11), Error de comando Home (16), y Tipo de dato de eje no válido (38) funcionan de la misma manera. Un número entre 0…*n* aparece en pantalla para el Código de error extendido. Este número es el índice del Sistema de Coordenadas indicando el eje que está en condición de error.

Para el código de error de Eje no configurado (11) hay un valor adicional de –1, que indica que el Sistema de coordenadas era incapaz de configurar el eje para el movimiento coordinado.

Para la instrucción MCLM, Código de error 13 – Parámetro fuera de rango, los Errores extendidos señalan un número que indica el parámetro en error como establecido en la plantilla en orden numérico de arriba hacia abajo comenzando de cero. Por ejemplo, 2 indica que el valor del parámetro para Move Type está en error.

Número y código de error referenciado	Indicador numérico del error extendido	Parámetro de instrucción	Descripción
Parámetro fuera de Rango (13)	2	Move Type	Move Type es menor que 0 o mayor que 1.
Parámetro fuera de Rango (13)	3	Position	La matriz de posición no es lo suficientemente grande como para proporcionar posiciones para todos los ejes en el sistema de coordenadas.
Parámetro fuera de Rango (13)	4	Speed	La velocidad es menor a 0.
Parámetro fuera de Rango (13)	6	Accel Rate	Accel Rate es menor o igual a 0.
Parámetro fuera de Rango (13)	8	Decel Rate	Decel Rate es menor o igual a 0.
Parámetro fuera de Rango (13)	11	Tipo de terminación	El tipo de terminación es menor que 0 o mayor que 3.

Para el código de error 54 – El valor máximo de desaceleración es cero, si el error extendido devuelve un número positivo (0-*n*), está haciendo referencia al eje en error en el sistema de coordenadas. Diríjase a la ficha general Coordinate System Properties y mire debajo de la columna de corchetes ([ ]) de la cuadrícula del eje para determinar qué eje tiene un valor máximo de desaceleración de 0. Haga clic en el botón de elipsis al lado del eje en error para acceder a la pantalla de propiedades del eje. Diríjase a la ficha Dynamics y haga el cambio apropiado al valor máximo de desaceleración. Si el número del error extendido es –1, esto significa que el sistema de coordenadas tiene un valor máximo de desaceleración de 0. Diríjase a la ficha Coordinate System Properties Dynamics para corregir el valor máximo de desaceleración.

MCLM Cambia a Bit de estado

Los Bit de estado proporcionan un medio para monitorear el progreso de la instrucción de movimiento. Hay tres tipos de bits de estado que brindan la información pertinente.

- Bits de estado del eje
- Bits de estado del sistema de coordenadas
- Bits de estado del movimiento coordinado

Cuando la instrucción MCLM comienza, los bits de estado sufren los siguientes cambios.

Bits de estado del eje

Nombre del bit	Significado
CoordinatedMotionStatus	Se establece cuando la instrucción comienza. Se borra cuando la instrucción finaliza.

Bits de estado del sistema de coordenadas

Nombre del bit	Significado
MotionStatus	Se establece cuando la instrucción MCLM está activa y el Sistema de Coordenadas está conectado a sus ejes asociados.

Bits de estado del movimiento coordinado

Nombre del bit	Significado
AccelStatus	Se establece cuando el vector está acelerando. Se borra cuando una incorporación está en proceso o cuando es movimiento vectorial está desacelerando.
DecelStatus	Se establece cuando el vector está desacelerando. Se borra cuando una incorporación está en proceso o cuando es movimiento vectorial está acelerando.
ActualPosToleranceStatus	Se establece sólo para el tipo de terminación de Tolerancia Real. Se establece luego de que estas dos condiciones se cumplen. 1) Interpolación está completa. 2) La distancia real al punto final programado es menor que el valor configurado de tolerancia real del sistema de coordenadas. El bit permanece habilitado luego de que la instrucción finaliza. El bit se restablece cuando comienza una nueva instrucción.
CommandPosToleranceStatus	Se establece para todos los tipos de terminación siempre que la distancia al punto final programado sea menor que el valor configurado de la tolerancia de comando del sistema de coordenadas. El bit permanece habilitado luego de que la instrucción finaliza. Se restablece cuando una nueva instrucción comienza.

Bits de estado del movimiento coordinado

Nombre del bit	Significado
StoppingStatus	El bit de estado de detención desaparece cuando se inicia la instrucción MCLM.
MoveStatus	Se establece cuando MCLM inicia un movimiento del eje. Borra el bit .PC de la última instrucción de movimiento o cuando una instrucción de control de movimiento ejecuta lo que provoca una detención.
MoveTransitionStatus	Se establece cuando el tipo de terminación No Decel o Command Tolerance se completa. Al combinar movimientos colineales, el bit no se establece, ya que la máquina está siempre en la ruta. Se borra cuando una combinación se completa, el movimiento de una instrucción pendiente comienza, o una instrucción de control de movimiento ejecuta lo que provoca una detención. No se encuentra en la ruta.
MovePendingStatus	Se establece cuando una instrucción de control de movimiento pendiente se encuentra en la cola de instrucción. Se borra cuando la cola de instrucción está vacía.
MovePendingQueueFullStatus	Se establece cuando la cola de instrucción está llena. Se borra cuando la cola tiene espacio para una nueva instrucción de control de movimiento coordinado.

Actualmente, el Movimiento coordinado sólo acepta poner en cola una instrucción de movimiento coordinado. Por lo tanto, el bit MovePendingStatus y el bit MovePendingQueueFullStatus son siempre los mismos.

Operando Profile

Al utilizar esta instrucción, deberá tener en cuenta el Operando Profile. Para obtener más información, consulte la página 71.

Motion Coordinated Circular Move (MCCM)

Utilice la instrucción MCCM para iniciar un movimiento coordinado circular bi- o tri-dimensional para los ejes específicos dentro de un sistema de coordenadas cartesianas. La nueva posición se define como una posición absoluta o incremental y se ejecuta a la velocidad deseada. La velocidad real de MCCM es una función del modo del movimiento (velocidad controlada o porcentaje de la velocidad máxima). La velocidad del movimiento se basa en el tiempo que toma para completar el movimiento circular usando los ejes programados. Cada eje recibe la orden de moverse a una velocidad que permite a todos los ejes alcanzar el punto final (posición establecida) al mismo tiempo.

La dimensión del círculo está definida por el número de ejes contenidos en el sistema de coordenadas. Por ejemplo, si tiene un sistema de coordenadas que contiene tres ejes con una instrucción MCCM que se mueve sólo en dos dimensiones, el movimiento resultante todavía se considera un arco o círculo tridimensional.

ATENCIÓN ⚠

Los tags para instrucciones de control de movimiento se deben utilizar sólo una vez. Reutilizar el tag de control de movimiento en otras instrucciones puede provocar operaciones imprevistas. Esto puede dañar el equipo o causar lesiones personales.

ATENCIÓN ⚠

Riesgo de sobreimpulso de velocidad y/o de posición final

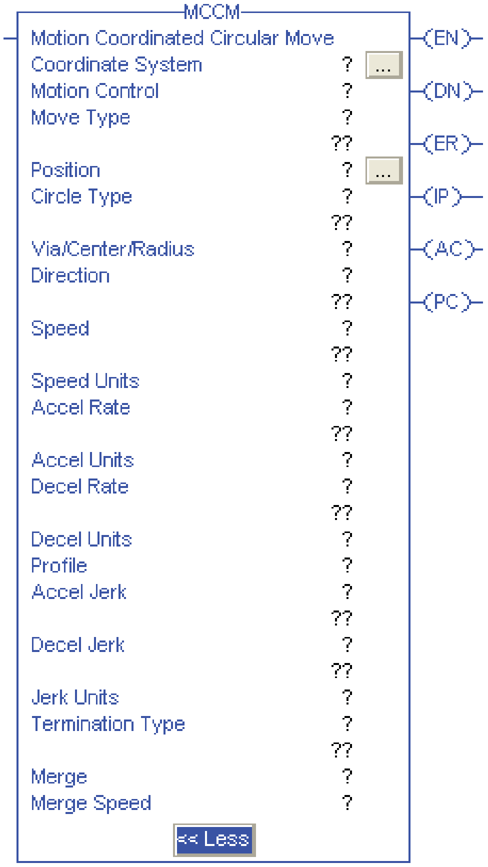
Si cambia los parámetros de movimiento dinámicamente mediante cualquier método, es decir cambiando la dinámica del movimiento (MCD o MCCD) o iniciando una nueva instrucción antes de que la última finalice, tenga en cuenta el riesgo de sobreimpulso de velocidad y/o de posición final.

Se puede exceder un perfil de velocidad trapezoidal si la máxima desaceleración se disminuye mientras el movimiento reduce la velocidad o está cerca del punto de desaceleración.

Se puede exceder un perfil de velocidad con curva en S si:

- la máxima desaceleración disminuye mientras el movimiento reduce la velocidad o está cerca del punto de desaceleración; o

- la máxima aceleración de jaloneo disminuye y el eje acelera. Recuerde, sin embargo, que el jaloneo se puede cambiar indirectamente si se encuentra especificado en % de tiempo.



MCCM
Motion Coordinated Circular Move
Coordinate System ?
Motion Control ?
Move Type ?
??
Position ?
Circle Type ?
??
Via/Center/Radius ?
Direction ?
??
Speed ?
??
Speed Units ?
Accel Rate ?
??
Accel Units ?
Decel Rate ?
??
Decel Units ?
Profile ?
Accel Jerk ?
??
Decel Jerk ?
??
Jerk Units ?
Termination Type ?
??
Merge ?
Merge Speed ?

Operandos de la instrucción MCCM – Lógica de escalera de relés

Operando	**Tipo**	**Formato**	**Descripción**
Sistema de coordenadas	COORDINATE_SYSTEM	tag	Grupo de ejes coordinado.
Motion Control	MOTION_ INSTRUCTION	tag	Estructura utilizada para acceder a los parámetros de estado de la instrucción.
Move Type	SINT, INT o DINT	inmediato o tag	0 = Absoluto 1 = Incremental
Position	REAL	tag de matriz[]	[unidades de coordinación]
Circle Type	SINT, INT o DINT	inmediato o tag	0 = Via 1 = Center 2 = Radius 3 = Center Incremental
Via/Center/ Radius	REAL	tag de matriz[] (via/center) Inmediato o tag (radius)	[unidades de coordinación]
Direction	SINT, INT o DINT	inmediato o tag	**2D** **3D** 0 = en sentido horario (CW) La más corta 1 = en sentido contrahorario (CCW) La más larga 2 = en sentido horario completo (CW Full) La más corta completa 3 = en sentido contrahorario completo (CCW Full) La más larga completa
Speed	SINT, INT, DINT o REAL	inmediato o tag	[unidades de coordinación]
Speed Units	SINT, INT o DINT	inmediato	0 = Unidades por seg 1 = % del Máximo
Accel Rate	SINT, INT, DINT o REAL	inmediato o tag	[unidades de coordinación]
Accel Units	SINT, INT o DINT	inmediato	0 = Unidades por seg^2 1 = % del Máximo

Operandos de la instrucción MCCM – Lógica de escalera de relés

Operando	Tipo	Formato	Descripción
Decel Rate	SINT, INT, DINT o REAL	inmediato o tag	[unidades de coordinación]
Decel Units	SINT, INT o DINT	inmediato	0 = Unidades por seg^2 1 = % del Máximo
Profile	SINT, INT o DINT	inmediato	0 = Trapezoidal 1 = Curva en S
Accel Jerk	SINT, INT, DINT o REAL	Inmediato o tag	**Siempre** debe ingresar los valores para los operandos Accel y Decel Jerk. Esta instrucción sólo utiliza los valores si Profile está configurado como curva en S. • Accel Jerk es el régimen de jaloneo de aceleración para el sistema de coordenadas. • Decel Jerk es el régimen de jaloneo de desaceleración para el sistema de coordenadas. Introduzca los regímenes de jaloneo en estas unidades de jaloneo. 0 = Unidades por seg^3 1 = % del Máximo 2 = % de tiempo Utilice estos valores para arrancar. • Accel Jerk = 100 (% del tiempo) • Decel Jerk = 100 (% del tiempo) • Unidades de jaloneo = 2
Decel Jerk	SINT, INT, DINT o REAL	Inmediato o tag	
Jerk Units	SINT, INT o DINT	Inmediato o tag	
Tipo de terminación	SINT, INT o DINT	inmediato o tag	0 = Tolerancia real 1 = No Establecido 2 = Tolerancia de comando 3 = No Decel 4 = Velocidad de contorno continua limitada 5 = Velocidad de contorno continua no limitada Consulte Escoja un tipo de terminación en la página 264.

Operandos de la instrucción MCCM – Lógica de escalera de relés

Operando	Tipo	Formato	Descripción
Merge	SINT, INT o DINT	inmediato	0 = Inhabilitado 1 = Movimiento coordinado 2 = Todos los movimientos
Merge Speed	SINT, INT o DINT	inmediato	0 = Programado 1 = Corriente

Texto estructurado

MCCM(CoordinateSystem,
MotionControl, MoveType,
Position, CircleType, Via/Center/Radius,
Direction, Speed, Speedunits, Accelrate,
Accelunits, Decelrate, Decelunits, Profile,
Accel Jerk, Deceljerk, Jerkunits,
Terminationtype,Merge,
Mergespeed);

Los operandos para el texto estructurado son iguales que para la instrucción MCCM de lógica de escalera de relé.

Cuando ingresa enumeraciones para el valor del operando en el texto estructurado, las enumeraciones de múltiples palabras se deben ingresar sin espacios. Por ejemplo: cuando ingresa Decel Units, el valor se debe ingresar como unitspersec^2 en lugar de Unidades por Seg^2 como se indica en la lógica de escalera.

Utilice las entradas en esta tabla como guía para ingresar los operandos del texto estructurado.

Entradas para Operandos del texto estructurado

Este operando	Cuenta con estas opciones que usted ingresa como...	
	Texto	**O como**
Sistema de coordenadas	Sin enumeración	Tag
Motion Control	Sin enumeración	Tag
Move Type	Sin enumeración	Tag 0 = Absoluto 1 = Incremental
Position	Sin enumeración	Tag de matriz
Circle Type	Sin enumeración	Tag 0 = Via 1 = Center 2 = Radius 3 = Center Incremental
Via/Center/Radius	Sin enumeración	tag de matriz (via/center) Inmediato o tag (radius)

Entradas para Operandos del texto estructurado

Este operando	**Cuenta con estas opciones que usted ingresa como...**	
	Texto	**O como**
Direction	Sin enumeración	**2D** / **3D** 0 En sentido horario / La más corta 1 En sentido contrahorario / La más larga 2 En sentido horario completo / La más corta completa 3 En sentido contrahorario completo / La más larga completa
Speed	Sin enumeración	Inmediato o tag
Speed Units	Unitspersec %ofmaximum	0 1
Accel Rate	Sin enumeración	Inmediato o tag
Accel Units	Unitspersec^2 %ofmaximum	0 1
Decel Rate	Sin enumeración	Inmediato o tag
Decel Units	Unitspersec^2 %ofmaximum	0 1
Profile	Trapezoidal Curva S	0 1
Accel Jerk	Sin enumeración	Inmediato o tag **Siempre** debe ingresar un valor para los operandos Accel y Decel Jerk. Esta instrucción sólo utiliza los valores si Profile está configurado como curva en S. Utilice estos valores para arrancar. • Accel Jerk = 100 (% del tiempo) • Decel Jerk = 100 (% del tiempo) • Unidades de jaloneo = 2
Decel Jerk	Sin enumeración	
Jerk Units	Unitspersec^3 %ofmaximum %oftime	0 1 2 (utilice este valor para comenzar)

Entradas para Operandos del texto estructurado

Este operando	Cuenta con estas opciones que usted ingresa como...	
	Texto	**O como**
Tipo de terminación	Sin enumeración	0 = Tolerancia real 1 = No Establecido 2 = Tolerancia de comando 3 = No Decel 4 = Velocidad de contorno continua limitada 5 = Velocidad de contorno continua no limitada Consulte Escoja un tipo de terminación en la página 264.
Merge	Inhabilitado (Disabled)	0
	Coordinatedmotion	1
	Allmotion	2
Merge Speed	Programada	0
	Actual	1

Sistema de coordenadas

El operando del sistema de coordenadas indica el sistema de ejes de movimiento que definen las dimensiones de un sistema de coordenadas cartesianas. Para esta versión, el sistema de coordenadas acepta hasta tres (3) ejes primarios. Sólo aquellos ejes configurados como ejes primarios (hasta 3) se incluyen en los cálculos de velocidad. Sólo los ejes primarios intervienen en el movimiento circular real.

Control de movimiento

Los siguientes bits de control están afectados por la instrucción MCCM.

Bit de control afectados por instrucción MCCM

Mnemónico	Descripción
Bit .EN (Habilitado) 31	El bit Habilitado se establece cuando las transiciones del renglón cambian de falso a verdadero. Este bit se restablece cuando las transiciones del renglón cambian de verdadero a falso.
Bit .DN (Listo) 29	El bit Listo se establece cuando la instrucción coordinada ha sido verificada y puesta en cola correctamente. Ya que se establece al mismo tiempo que entra en cola, puede aparecer como establecido cuando un error de tiempo de ejecución aparece durante la operación de verificación luego de salir de la cola. Se restablece cuando el renglón va de falso a verdadero.
Bit .ER (Error) 28	El bit Error se restablece cuando las transiciones del renglón cambian de falso a verdadero. Se establece cuando el movimiento coordinado no puede comenzar correctamente. También se establece con el Bit Listo cuando una instrucción en cola identifica un error de tiempo de ejecución.
Bit .IP (En proceso) 26	El bit En proceso se establece cuando el movimiento coordinado se inicia correctamente. Se restablece cuando hay movimiento subsiguiente y el movimiento coordinado alcanza la nueva posición, o cuando hay un movimiento subsiguiente y el movimiento coordinado alcanza las especificaciones del tipo de terminación, o cuando el movimiento coordinado es reemplazado por otra instrucción MCCM o MCLM con un tipo de incorporación de movimiento coordinado, o cuando se finaliza con una instrucción MCS o MCSD.
Bit .AC (Activo) 23	Cuando tiene una instrucción de movimiento coordinado en cola, el bit Activo permite conocer qué instrucción está controlando el movimiento. Se establece cuando el movimiento coordinado se activa. Se restablece cuando el bit de Proceso completo se establece o cuando la instrucción se detiene.
Bit .PC (Proceso completo) 27	El bit Proceso Completo se restablece cuando las transiciones del renglón cambian de falso a verdadero. Se establece cuando no hay movimiento subsiguiente y el movimiento coordinado alcanza una nueva posición, o cuando hay un movimiento subsiguiente y el movimiento coordinado alcanza el tipo de terminación específico.
Bit .ACCEL (Aceleración) 01	El bit Aceleración se establece mientras el movimiento coordinado está en la fase de aceleración. Se restablece mientras el movimiento coordinado está en velocidad constante o fase de desaceleración, o cuando el movimiento coordinado finaliza.
Bit .DECEL (Desaceleración) 02	El bit Desaceleración se establece mientras el movimiento coordinado está en la fase de desaceleración. Se restablece mientras el movimiento coordinado está en velocidad constante o en fase de aceleración, o cuando el movimiento coordinado finaliza.

Move Type

El operando Move Type determina el método utilizado por la matriz de posición para indicar la ruta del movimiento coordinado y el método que el parámetro via/center/radius utiliza para indicar las posiciones de círculo via y center. Las opciones son: Absolute o Incremental.

- Absolute – el sistema de coordenadas se mueve a la posición establecida a la velocidad definida, utilizando los regímenes de Aceleración y Desaceleración determinado por los operandos correspondientes, mediante una ruta circular.

  Cuando un eje es configurado para operación giratoria, los movimientos absolutos funcionan de la misma manera que para ejes lineales. Cuando la posición del eje excede el parámetro de desbobinado, se genera un error.

  La señal de la matriz de posición específica se interpreta mediante el controlador como la dirección para el movimiento. Los valores de posición negativa ordena al interpolador para mover el eje giratorio en una dirección negativa para obtener la posición absoluta deseada. Un valor positivo indica que se necesita un movimiento positivo para alcanzar la posición establecida. Para mover a la posición de desbobinado en la dirección negativa, se utiliza el valor de posición de desbobinado negativo 0 y –0 se consideran 0. Cuando la posición es mayor que el valor de desbobinado, se genera un error. El eje se puede desplazar a través de la posición de desbobinado, pero nunca de manera incremental más que un valor de desbobinado.

- Incremental – el sistema de coordenadas se mueve la distancia definida por la matriz de posición a la velocidad específica, utilizando los regímenes de Aceleración y Desaceleración determinado por los operandos correspondientes, mediante una ruta circular.

  La distancia específica se interpreta mediante el interpolador y puede ser positiva o negativa. Los valores de posición negativos ordenan al interpolador que mueva el eje giratorio en una dirección negativa, mientras los valores positivos indican movimiento positivo para alcanzar la posición establecida.

Position

El operando Position es una matriz unidimensional, cuya dimensión está definida para ser, al menos, el equivalente del número de ejes indicados en el sistema de coordenadas. Es la matriz Position la que define la nueva posición absolute o incremental.

Circle Type

El operando Circle Type indica como interpretar la matriz etiquetada como via/center/radius. Las opciones son: Via, Circle, Radius, Center Incremental.

- Via indica que los miembros de la matriz via/center/radius establecen un punto via entre los puntos de inicio y final.

- Center indica que los miembros de la matriz via/center/radius contienen el centro del círculo.

- Radius indica que el miembro de la primera matriz via/center/radius contiene el radio. Se ignoran lo otros miembros. Radius es válido sólo en los sistemas de coordenadas bidimensionales.

- Center Incremental indica que los miembros de la matriz via/center/radius establecen una posición que siempre define de manera incremental el centro del círculo a pesar del operando Move Type. La señal del valor incremental se mide desde el punto de inicio hasta el centro.

Ejemplos de arco bidimensional

Los siguientes ejemplos indican el uso de Move Type Absoluto e Incremental con los diversos tipos de círculo.

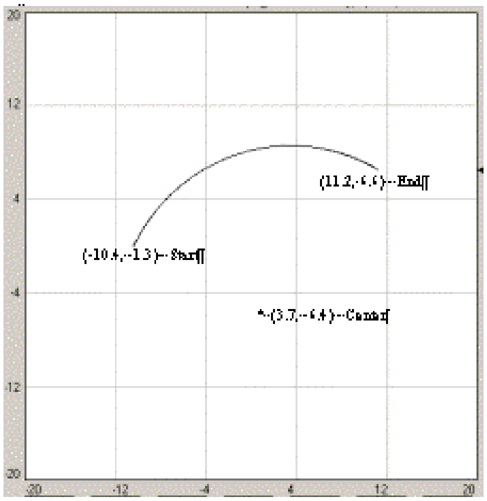
MCCM Utilizar Circle type Center

Los siguientes ejemplos indican el uso de la instrucción MCCM con un Circle Type Center y un Move type Absoluto (primer ejemplo) e Incremental (segundo ejemplo) para llegar al mismo resultado. Los supuestos básicos son:

- los dos ejes, Axis0 y Axis1, son miembros del sistema de coordenadas, Coordinated_sys.
- Axis0 y Axis1 son ortogonales entre sí.
- Coordinated_sys se inicia a (–10.4,–1.3) unidades.

Mueve el Coordinated_sys a lo largo de un arco a (11.2,6.6) unidades con un centro de (3.7,–6.4) unidades a velocidad vectorial de 10.0 unidades por segundo con los valores de aceleración y desaceleración de 5.0 unidades por segundo^2. El siguiente gráfico indica la ruta generada por la información mencionada anteriormente

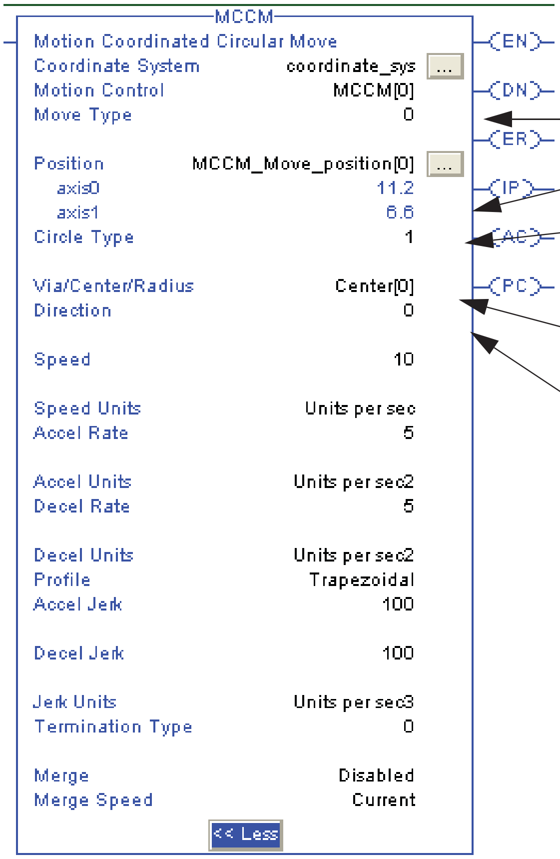
Gráfico de la instrucción MCCM con un Circle Type Center.



La velocidad vectorial de los ejes seleccionados es igual a la velocidad indicada en las unidades por segundo o el porcentaje de la velocidad máxima del sistema de coordenadas. Asimismo, la aceleración y desaceleración es igual a la aceleración/desaceleración establecida en las unidades por segundo^2 o porcentaje de aceleración máxima del sistema de coordenadas.

Esta ruta se puede lograr utilizando una instrucción MCCM en dirección en sentido horario con Move Type = Absolute o con Move Type = Incremental. Cuando se escoge Circle Type Center, la posición Via/Center/Radius define el centro del arco.

Instrucción de lógica de escalera MCCM con Move Type Absolute



MCCM

Motion Coordinated Circular Move

Coordinate System	coordinate_sys
Motion Control	MCCM[0]
Move Type	0
Position	MCCM_Move_position[0]
axis0	11.2
axis1	6.6
Circle Type	1
Via/Center/Radius	Center[0]
Direction	0
Speed	10
Speed Units	Units per sec
Accel Rate	5
Accel Units	Units per sec2
Decel Rate	5
Decel Units	Units per sec2
Profile	Trapezoidal
Accel Jerk	100
Decel Jerk	100
Jerk Units	Units per sec3
Termination Type	0
Merge	Disabled
Merge Speed	Current

Move Type es Absolute.

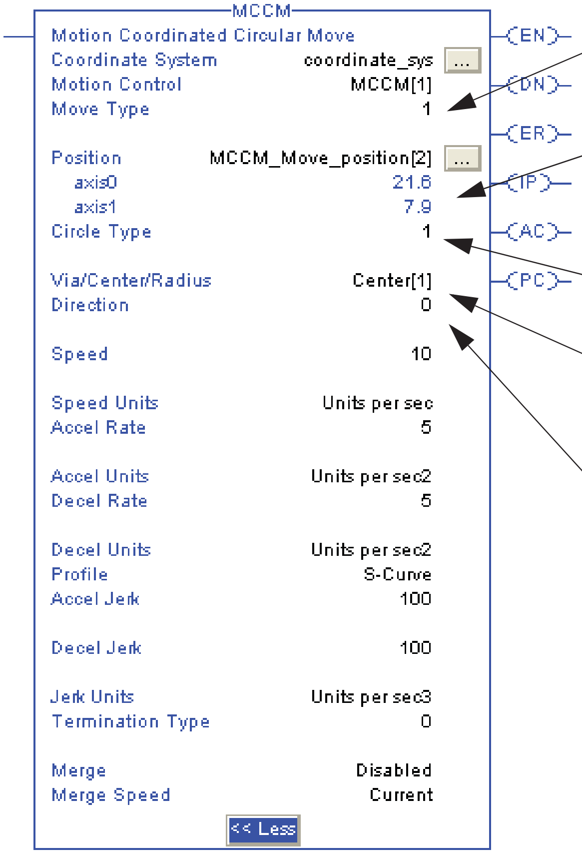
Posición definida en unidades absolutas.

CIrcle Type es center.

Posición central definida en unidades absolutas como (3.7,–6.4).

La dirección es en sentido horario.

Instrucción de lógica de escalera MCCM con Move Type Incremental



MCCM	
Motion Coordinated Circular Move	
Coordinate System	coordinate_sys
Motion Control	MCCM[1]
Move Type	1
Position	MCCM_Move_position[2]
axis0	21.6
axis1	7.9
Circle Type	1
Via/Center/Radius	Center[1]
Direction	0
Speed	10
Speed Units	Units per sec
Accel Rate	5
Accel Units	Units per sec2
Decel Rate	5
Decel Units	Units per sec2
Profile	S-Curve
Accel Jerk	100
Decel Jerk	100
Jerk Units	Units per sec3
Termination Type	0
Merge	Disabled
Merge Speed	Current

Move Type es Incremental.

La posición definida como la distancia incremental desde el punto de arranque de (–10.4,–1.3).

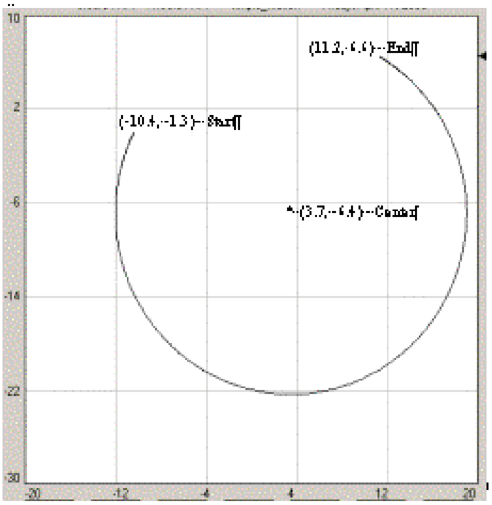
Circle Type es center.

La posición Center definida como la distancia incremental de (14.1,–5.1) desde el punto de arranque de (–10.4,–1.3).

La dirección es en sentido horario.

Si se ha seleccionado una Dirección en sentido contrahorario (Dirección = 1), los ejes se mueven a lo largo de la curva como se indica en el siguiente gráfico.

Gráfico de la ruta con dirección en sentido contrahorario



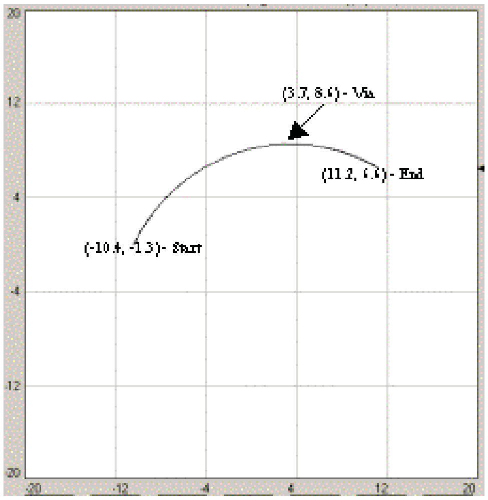
Instrucción MCCM Utilizar Circle Type Via

Los siguientes ejemplos indican el uso de la instrucción MCCM con un tipo de círculo Via y un Move type Absoluto (primer ejemplo) e Incremental (segundo ejemplo) para llegar al mismo resultado. Los supuestos básicos son:

- los dos ejes, Axis0 y Axis1, son miembros del sistema de coordenadas, coordinate_sys.
- Axis0 y Axis1 son ortogonales entre sí.
- Coordinated_sys se inicia a (–10.4,–1.3) unidades.

Mueve el Coordinated_sys a lo largo de un arco a (11.2,6.6) unidades atravesando el punto (3.7,8.6) unidades a velocidad vectorial de 10.0 unidades por segundo con los valores de aceleración y desaceleración de 5.0 unidades por segundo^2. El siguiente gráfico indica la ruta generada por la información mencionada anteriormente.

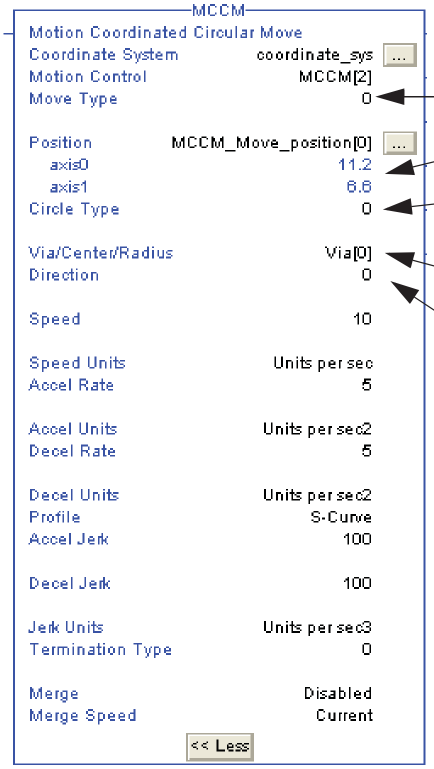
Gráfico de la ruta de la instrucción MCCM con operandos Via y Absolute



La velocidad vectorial de los ejes seleccionados es igual a la velocidad indicada en las unidades por segundo o el porcentaje de la velocidad máxima del sistema de coordenadas. Asimismo, la aceleración y desaceleración es igual a la aceleración/desaceleración establecida en las unidades por segundo^2 o porcentaje de aceleración máxima del sistema de coordenadas.

Esta ruta se puede lograr utilizando una instrucción MCCM en dirección en sentido horario con Move Type = Absolute o con Move Type = Incremental. Cuando se elige Circle Type Via, la posición Via/Center/Radius define un punto a través del cual debe pasar el arco.

Lógica de escalera de la instrucción MCCM con los valores de los operandos Via y Absolute


MCCM
Motion Coordinated Circular Move
Coordinate System coordinate_sys
Motion Control MCCM[2]
Move Type 0
Position MCCM_Move_position[0]
axis0 11.2
axis1 6.6
Circle Type 0
Via/Center/Radius Via[0]
Direction 0
Speed 10
Speed Units Units per sec
Accel Rate 5
Accel Units Units per sec2
Decel Rate 5
Decel Units Units per sec2
Profile S-Curve
Accel Jerk 100
Decel Jerk 100
Jerk Units Units per sec3
Termination Type 0
Merge Disabled
Merge Speed Current

Move Type es Absolute.

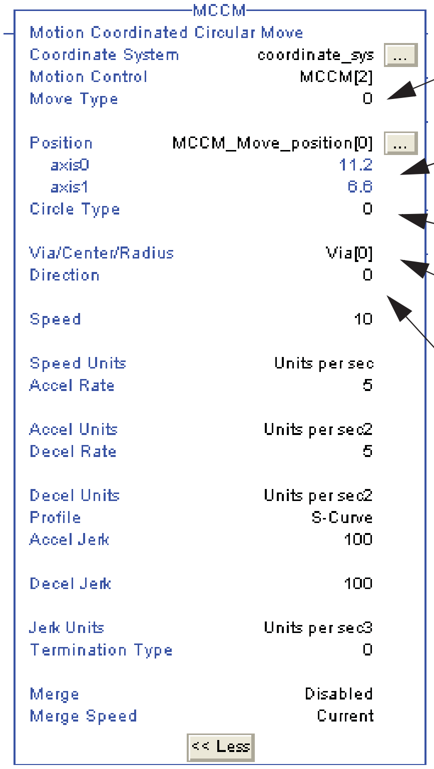
Posición definida en unidades absolutas.

CIrcle Type es Via.

Posición Via definida en unidades absolutas como (3.7,8.6).

La dirección es en sentido horario.

Lógica de escalera de la instrucción MCCM con los valores de los operandos Via e Incremental


MCCM
Motion Coordinated Circular Move
Coordinate System coordinate_sys
Motion Control MCCM[2]
Move Type 0
Position MCCM_Move_position[0]
axis0 11.2
axis1 6.6
Circle Type 0
Via/Center/Radius Via[0]
Direction 0
Speed 10
Speed Units Units per sec
Accel Rate 5
Accel Units Units per sec2
Decel Rate 5
Decel Units Units per sec2
Profile S-Curve
Accel Jerk 100
Decel Jerk 100
Jerk Units Units per sec3
Termination Type 0
Merge Disabled
Merge Speed Current

Move Type es Incremental.

La posición definida como la distancia incremental desde el punto de arranque de (–10.4,–1.3).

Circle Type es Via.

La posición Via definida como la distancia incremental de (14.1,9.9) desde el punto de arranque de (–10.4,–1.3).

La dirección es en sentido horario.

Ya que hay tres puntos (la posición actual de los ejes, el punto final indicado, y el punto via indicado), es difícil programar un arco deficiente. Ya que es posible programar un arco que no es el deseado, se producirá una falla en el tiempo de ejecución del error de programación circular con un arco si los tres puntos son colineales (los tres en la misma línea) o únicos (dos o más puntos son iguales). Además, el punto via implica la dirección del arco y por lo tanto, no es necesario (y se ignora) para indicar la dirección.

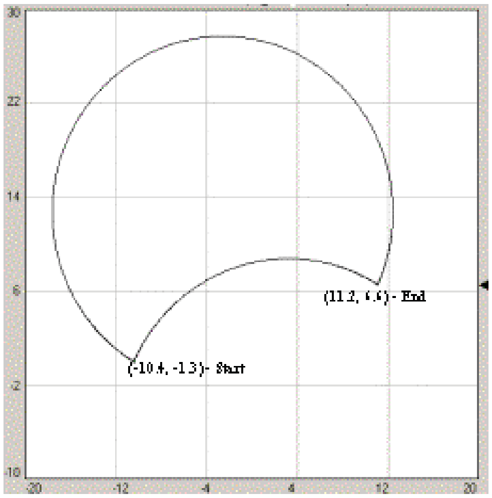
Instrucción MCCM Utilizar Circle Type Radius

Los siguientes ejemplos indican el uso de la instrucción MCCM con un tipo de círculo Radius y un Move type Absolute (primer ejemplo) e Incremental (segundo ejemplo) para llegar al mismo resultado. Los supuestos básicos son:

- los dos ejes, Axis0 y Axis1, son miembros del sistema de coordenadas, coordinate_sys.
- el valor de la dimensión del sistema de coordenadas está configurado como 2. Los Circle Types Radius sólo se pueden configurar cuando dos dimensiones se configuran para el sistema de coordenadas.
- Axis0 y Axis1 son ortogonales entre sí.
- coordinated_sys se inicia a (–10.4,–1.3) unidades.

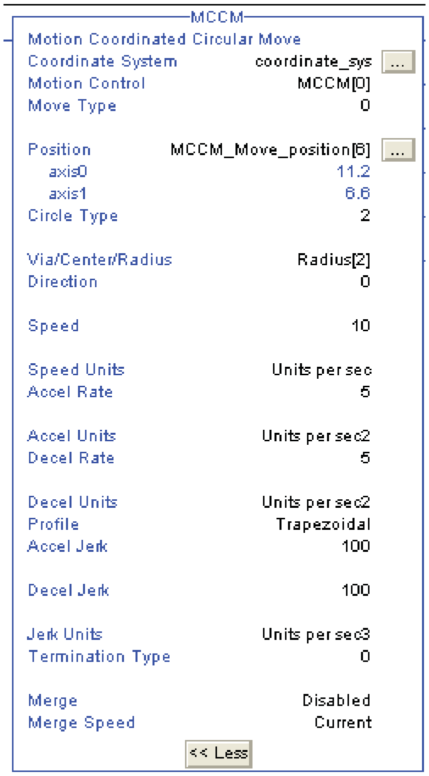
Mueve el Coordinated_sys a lo largo de un arco a (11.2,6.6) unidades con un radio de 15 unidades a velocidad vectorial de 10.0 unidades por segundo con los valores de aceleración y desaceleración de 5.0 unidades por segundo^2. El siguiente gráfico indica las rutas generadas por la información anterior utilizando un valor Radius de 15 unidades y –15 unidades.

Gráfico de la ruta con Circle Type Radius



Esta ruta se puede lograr utilizando una instrucción MCCM en dirección en sentido horario con Move Type = Absolute o con Move Type = Incremental. Cuando se escoge Circle Type Radius, la posición Via/Center/Radius define el radio del arco.

Move type Absolute de la instrucción MCCM; Circle Type Radius



Motion Coordinated Circular Move

Coordinate System	coordinate_sys
Motion Control	MCCM[0]
Move Type	0
Position	MCCM_Move_position[6]
axis0	11.2
axis1	6.6
Circle Type	2
Via/Center/Radius	Radius[2]
Direction	0
Speed	10
Speed Units	Units per sec
Accel Rate	5
Accel Units	Units per sec2
Decel Rate	5
Decel Units	Units per sec2
Profile	Trapezoidal
Accel Jerk	100
Decel Jerk	100
Jerk Units	Units per sec3
Termination Type	0
Merge	Disabled
Merge Speed	Current

Move Type es Absolute.

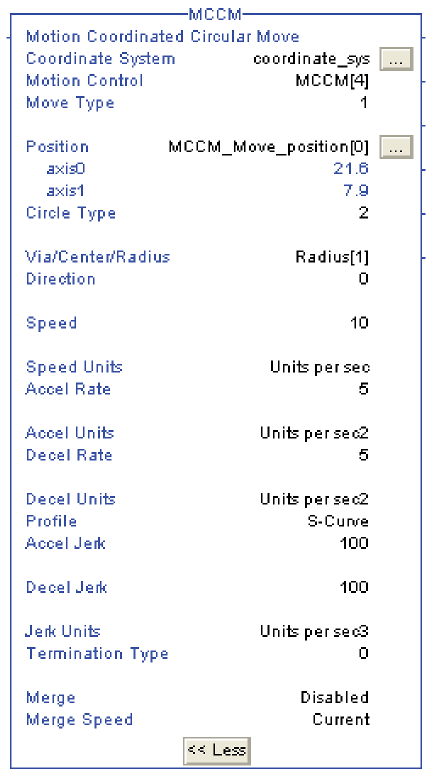
Posición definida en unidades absolutas.

CIrcle Type es Radius.

El radio definido en 15 unidades se almacena en el tag Radius [2].

La dirección es en sentido horario.

Move type Incremental de la instrucción MCCM; Circle Type Radius



Motion Coordinated Circular Move

Coordinate System	coordinate_sys
Motion Control	MCCM[4]
Move Type	1
Position	MCCM_Move_position[0]
axis0	21.6
axis1	7.9
Circle Type	2
Via/Center/Radius	Radius[1]
Direction	0
Speed	10
Speed Units	Units per sec
Accel Rate	5
Accel Units	Units per sec2
Decel Rate	5
Decel Units	Units per sec2
Profile	S-Curve
Accel Jerk	100
Decel Jerk	100
Jerk Units	Units per sec3
Termination Type	0
Merge	Disabled
Merge Speed	Current

Move Type es Incremental.

La posición definida como la distancia incremental desde el punto de arranque de (–10.4,–1.3).

Circle Type es Radius.

El radio definido en 15 unidades se almacena en el tag Radius [1].

La dirección es en sentido horario.

Move Type no afecta la especificación del valor Radius. Un radio positivo siempre crea un arco más corto (<180°) y un radio negativo crea un arco más largo (>180°) (vea gráfico de la ruta). Si el mismo es de 180°, la señal del radio es irrelevante. Un Circle Type Radius es válido en sistemas de coordenadas bidimensionales solamente.

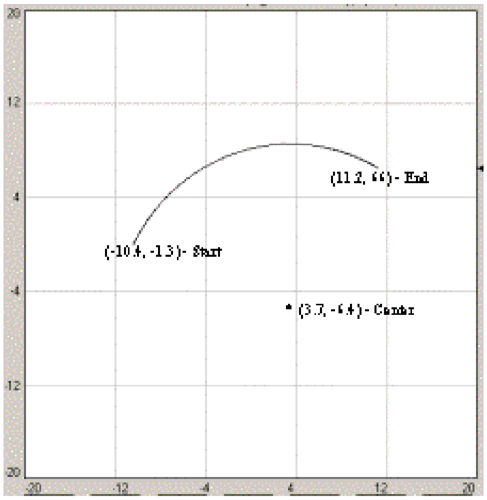
MCCM Utilizar Circle type Center Incremental

Los siguientes ejemplos indican el uso de la instrucción MCCM con un tipo de círculo Center Incremental y un Move type Absolute (primer ejemplo) e Incremental (segundo ejemplo) para llegar al mismo resultado. Los supuestos básicos son:

- los dos ejes, Axis0 y Axis1, son miembros del sistema de coordenadas, coordinate_sys.
- Axis0 y Axis1 son ortogonales entre sí.
- coordinated_sys se inicia a (–10.4,–1.3) unidades.

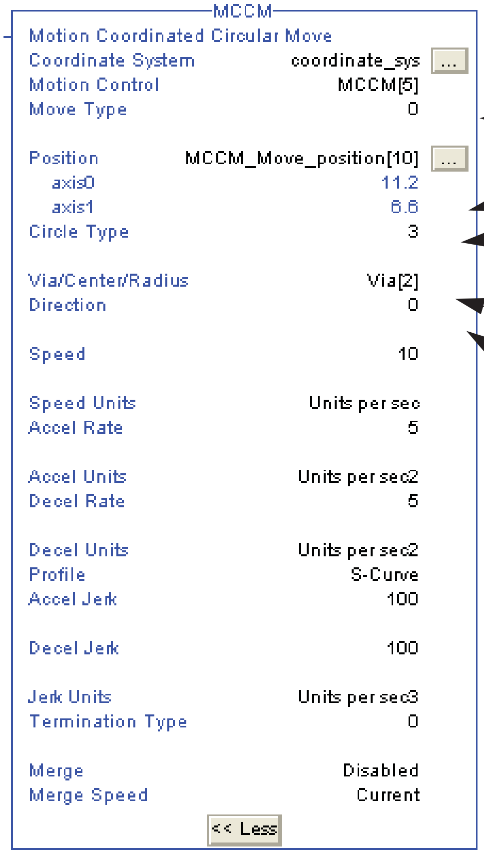
Mueve el coordinated_sys a lo largo de un arco a (11.2,6.6) unidades con un centro de (14.1,–5.1) unidades a velocidad vectorial de 10.0 unidades por segundo con los valores de aceleración y desaceleración de 5.0 unidades por segundo^2. El siguiente gráfico indica la ruta generada por la información mencionada anteriormente.

Gráfico de la ruta con Circle Type Center Incremental



Esta ruta se puede lograr utilizando una instrucción MCCM en dirección en sentido horario con Move Type = Absolute o con Move Type = Incremental. Cuando se escoge Circle Type Center Incremental, la posición Via/Center/Radius define el centro del arco.

Move type Absolute de la instrucción MCCM; Circle Type Incremental



MCCM

Motion Coordinated Circular Move

Coordinate System	coordinate_sys
Motion Control	MCCM[5]
Move Type	0
Position	MCCM_Move_position[10]
axis0	11.2
axis1	6.6
Circle Type	3
Via/Center/Radius	Via[2]
Direction	0
Speed	10
Speed Units	Units per sec
Accel Rate	5
Accel Units	Units per sec2
Decel Rate	5
Decel Units	Units per sec2
Profile	S-Curve
Accel Jerk	100
Decel Jerk	100
Jerk Units	Units per sec3
Termination Type	0
Merge	Disabled
Merge Speed	Current

Move Type es Absolute.

Posición definida en unidades absolutas.

Circle Type es Center Incremental.

Center definida como la distancia incremental de (14.1,–5.1) desde el punto de arranque de (–10.4,–1.3).

La dirección es en sentido horario.

La instrucción MCCM con Move Type Incremental y Center Type Center Incremental es igual a una instrucción MCCM con Move Type Incremental y Circle Type Center.

Ejemplo de Círculo completo bidimensional

La creación de un círculo completo es un caso especial de un arco circular. El siguiente es un ejemplo de un círculo completo bidimensional.

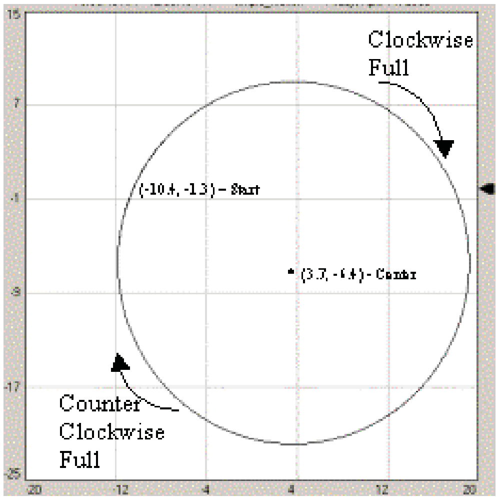
Círculo completo MCCM

Los siguientes ejemplos indican el uso de la instrucción MCCM con un tipo de círculo Center y un Move type Absolute (primer ejemplo) e Incremental (segundo ejemplo) para crear un círculo completo. Los supuestos básicos son:

- Los dos ejes, Axis0 y Axis1, son miembros del sistema de coordenadas, Coordinated_sys.
- Axis0 y Axis1 son ortogonales entre sí.
- Coordinated_sys se inicia a (–10.4,–1.3) unidades.

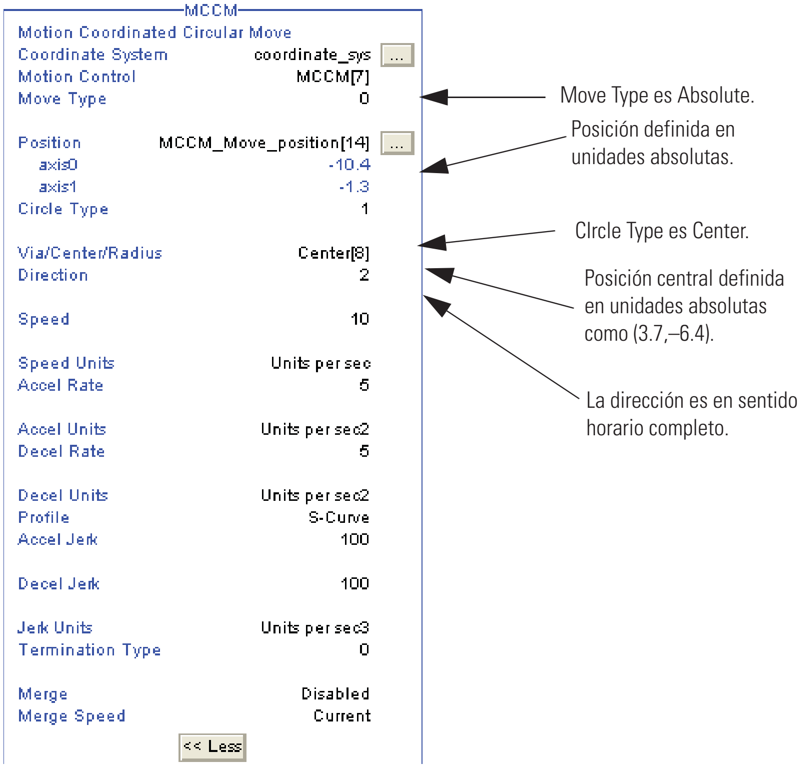
Mueve el Coordinated_sys a lo largo de un arco a (–10.4,–1.3) unidades con un centro de (3.7,–6.4) unidades a velocidad vectorial de 10.0 unidades por segundo con los valores de aceleración y desaceleración de 5.0 unidades por segundo^2. El siguiente gráfico indica el círculo generado por la información mencionada anteriormente.

Gráfico de la ruta del círculo completo de la instrucción MCCM



Esta ruta se puede lograr utilizando una instrucción MCCM en dirección en sentido horario con Move Type = Absolute o con Move Type = Incremental. Cuando se escoge Circle Type Center, la posición Via/Center/Radius define el centro del arco.

Move type Absolute de la instrucción MCCM; Circle Type Center.



Coordinate System	coordinate_sys
Motion Control	MCCM[7]
Move Type	0
Position	MCCM_Move_position[14]
axis0	-10.4
axis1	-1.3
Circle Type	1
Via/Center/Radius	Center[8]
Direction	2
Speed	10
Speed Units	Units per sec
Accel Rate	5
Accel Units	Units per sec2
Decel Rate	5
Decel Units	Units per sec2
Profile	S-Curve
Accel Jerk	100
Decel Jerk	100
Jerk Units	Units per sec3
Termination Type	0
Merge	Disabled
Merge Speed	Current

Move Type es Absolute.

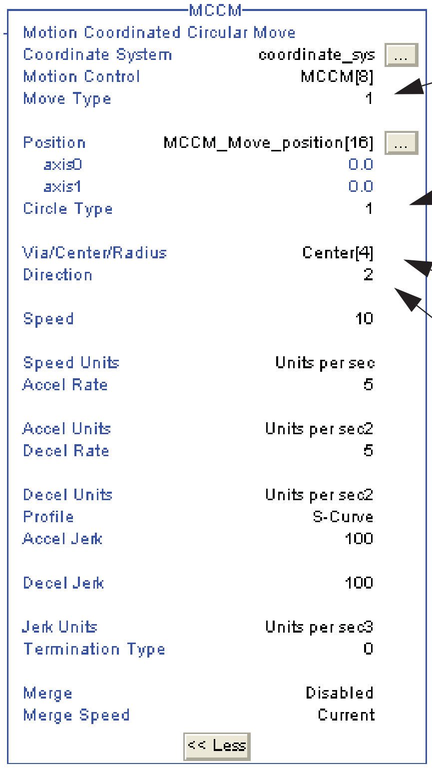
Posición definida en unidades absolutas.

CIrcle Type es Center.

Posición central definida en unidades absolutas como (3.7,–6.4).

La dirección es en sentido horario completo.

MCCM con Move Type Incremental y Center Type Center.



MCCM

Motion Coordinated Circular Move

Coordinate System	coordinate_sys
Motion Control	MCCM[8]
Move Type	1
Position	MCCM_Move_position[16]
axis0	0.0
axis1	0.0
Circle Type	1
Via/Center/Radius	Center[4]
Direction	2
Speed	10
Speed Units	Units per sec
Accel Rate	5
Accel Units	Units per sec2
Decel Rate	5
Decel Units	Units per sec2
Profile	S-Curve
Accel Jerk	100
Decel Jerk	100
Jerk Units	Units per sec3
Termination Type	0
Merge	Disabled
Merge Speed	Current

Move Type es Incremental.

Circle Type es center.

Center definida como la distancia incremental de (14.1,–5.1) desde el punto de arranque de (–10.4,–1.3).

La dirección es en sentido horario completo.

Para establecer un círculo completo usando Radius como Circle Type:

- el punto de inicio no debe ser igual al punto final.
- la dirección debe ser en sentido horario completo o en sentido contrahorario completo.
- la señal de Radius es irrelevante.

MCCM con ejemplos de ejes giratorios

Los siguientes ejemplos muestran el uso de la instrucción MCCM con ejes giratorios y Move Types Absolute e Incremental.

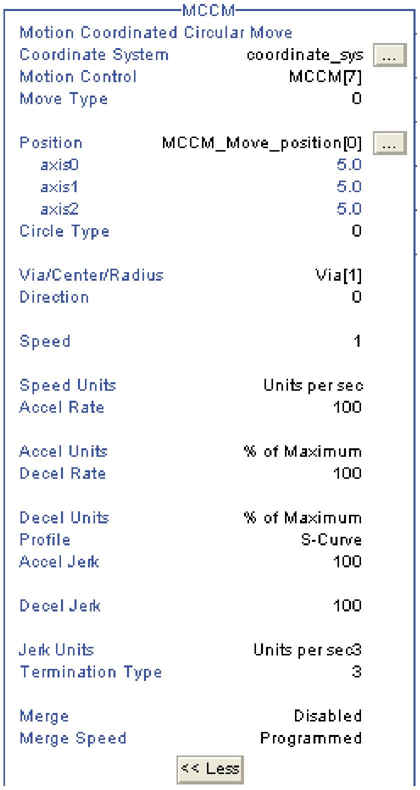
Instrucción MCCM con tres ejes, un eje giratorio y Move Type Absolute

El primer ejemplo utiliza un sistema de coordenadas de tres ejes con un eje giratorio y un Move type Absolute. El diagrama de la ruta se basa en los siguientes supuestos:

- Sistema de coordenadas de tres ejes llamado coordinate_sys (Axis2, el eje Z, no aparece en los gráficos para reducir la confusión y para ilustrar las acciones del eje giratorio (Axis0).
- Axis0 es Giratorio con un desbobinado de 5 revoluciones.
- La posición de arranque es 0, 0, 0.

- La posición final es 5, 5, 5.
- La posición Via es 5, 3.5, 3.5.

Instrucción de lógica de escalera MCCM con Move Type Absolute



MCCM

Motion Coordinated Circular Move

Coordinate System	coordinate_sys
Motion Control	MCCM[7]
Move Type	0
Position	MCCM_Move_position[0]
axis0	5.0
axis1	5.0
axis2	5.0
Circle Type	0
Via/Center/Radius	Via[1]
Direction	0
Speed	1
Speed Units	Units per sec
Accel Rate	100
Accel Units	% of Maximum
Decel Rate	100
Decel Units	% of Maximum
Profile	S-Curve
Accel Jerk	100
Decel Jerk	100
Jerk Units	Units per sec3
Termination Type	3
Merge	Disabled
Merge Speed	Programmed

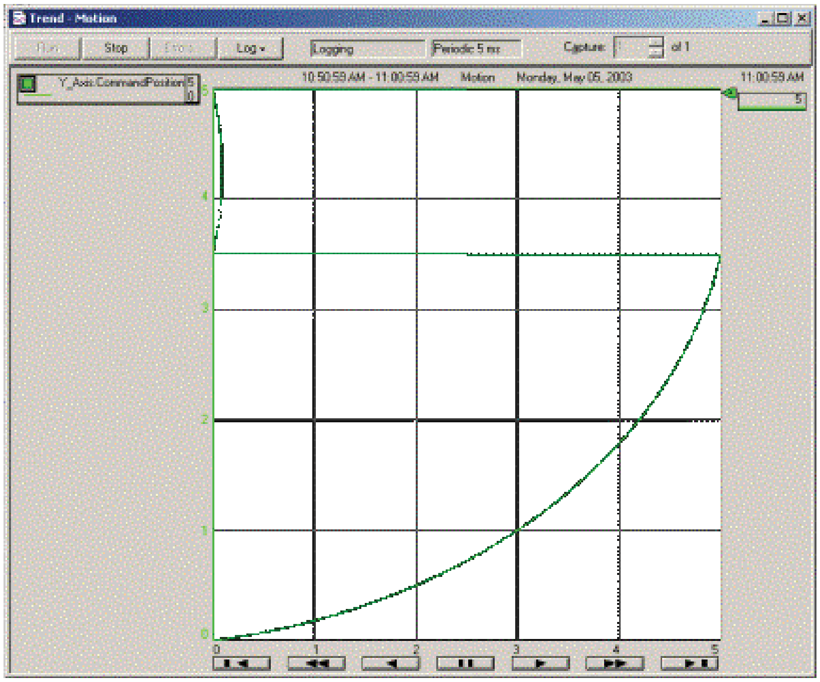
Move Type es Absolute.

Circle Type es Via.

La dirección es la más corta.

La instrucción MCCM anterior produce el siguiente gráfico.

Gráfico de MCCM con tres ejes, un eje giratorio y Move Type Absolute



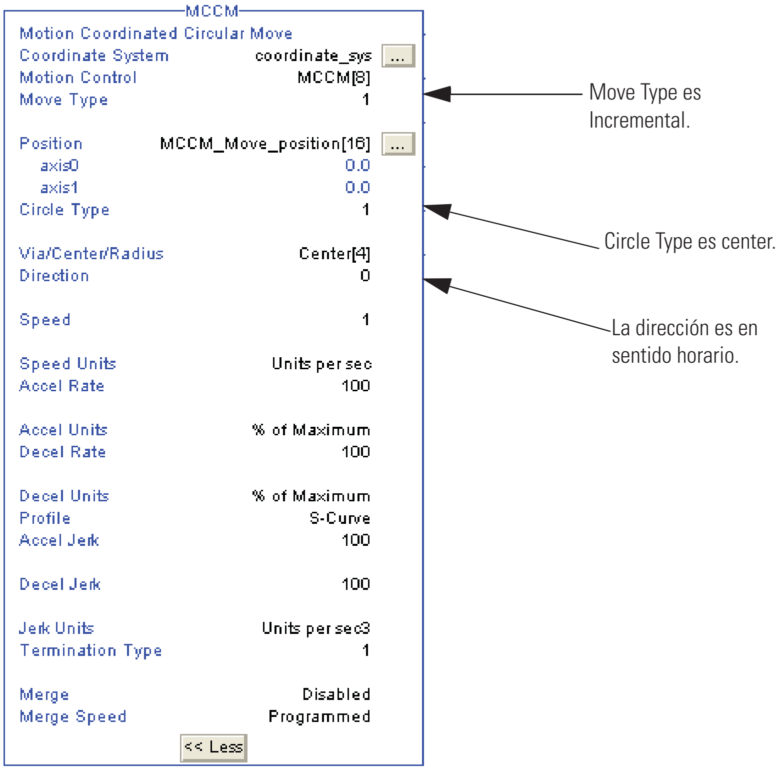
En realidad, el eje hace un recorrido en sentido contrahorario en un arco desde (0,0,0) a (5,5,5) mediante la posición (5,3.5,3.5). La dirección fue establecida en sentido horario pero Via como Circle Type, el operando Direction es omitido. El movimiento se detiene luego de generar un arco de 90 grados. Había un recorrido a través del desbobinado para Axis0 aunque estaba en Move Type Absolute. Observar que la ruta del movimiento coordinado está determinada en un espacio lineal, pero la posición de los ejes está limitada por la configuración giratoria. Los puntos final y Via son necesarios para ajustar dentro de la posición absoluta definida por el desbobinado giratorio de Axis0. Sin embargo, el movimiento resultante de estas opciones puede recorrer el desbobinado de los ejes giratorios.

Instrucción MCCM con dos Ejes giratorios y Move Type Incremental

Este ejemplo utiliza un sistema de coordenadas de dos ejes giratorios y un Move type Incremental. El diagrama de la ruta se basa en estos supuestos:

- Dos ejes del sistema de coordenadas llamados coordinate_sys.
- Axis0 es Giratorio con un desbobinado de 1 revolución.
- Axis1 es Giratorio con un desbobinado de 2 revoluciones.
- La posición de arranque es 0, 0.
- Incremento a la posición final es 0.5, –0.5.
- Incremento a la posición central es 0.5, 0.

Instrucción de lógica de escalera MCCM con Move Type Absolute



MCCM

Motion Coordinated Circular Move

Coordinate System	coordinate_sys
Motion Control	MCCM[8]
Move Type	1
Position	MCCM_Move_position[16]
axis0	0.0
axis1	0.0
Circle Type	1
Via/Center/Radius	Center[4]
Direction	0
Speed	1
Speed Units	Units per sec
Accel Rate	100
Accel Units	% of Maximum
Decel Rate	100
Decel Units	% of Maximum
Profile	S-Curve
Accel Jerk	100
Decel Jerk	100
Jerk Units	Units per sec3
Termination Type	1
Merge	Disabled
Merge Speed	Programmed

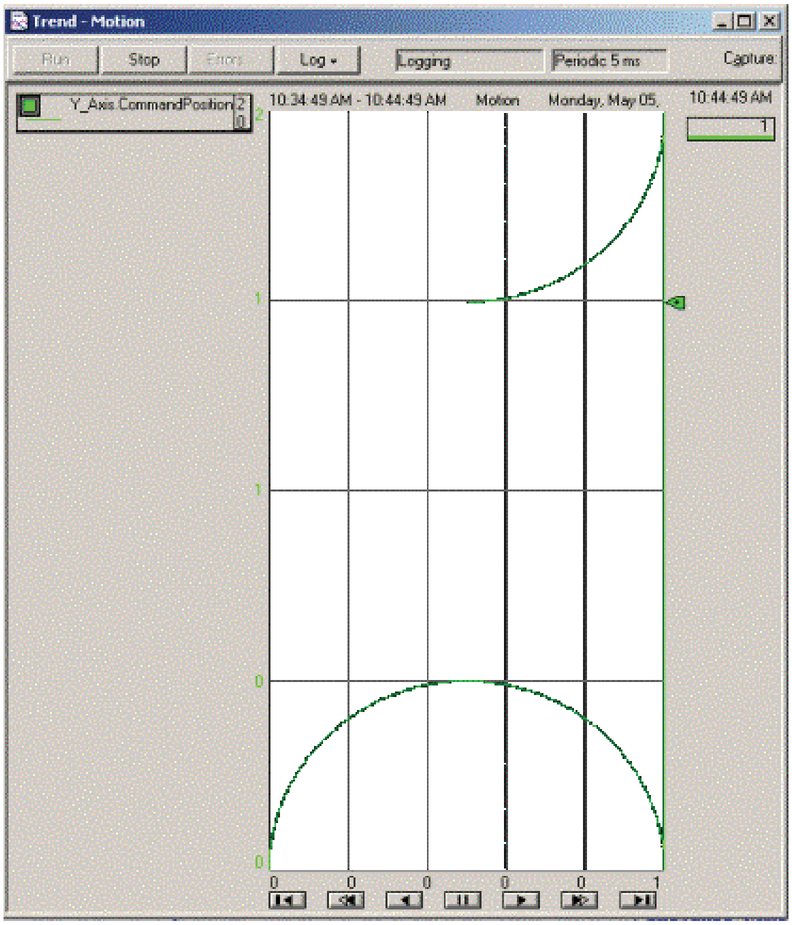
Move Type es Incremental.

Circle Type es center.

La dirección es en sentido horario.

La instrucción MCCM anterior produce el siguiente gráfico.

Diagrama de MCCM con dos Ejes giratorios y Move Type Incremental



El eje hace un recorrido en sentido horario en un círculo desde (0,0) a (0.5,1.5). El movimiento se detiene luego de generar un arco de 270 grados. Había un recorrido a través del desbobinado para Axis1. Observar que la ruta del movimiento coordinado está determinada en un espacio lineal, pero la posición de los ejes está limitada por la configuración giratoria. El punto final fue de (0.5,–0.5) para los cálculos del círculo, pero el punto final real para el movimiento fue de (0.5,1.5). La instrucción establecida y se obtiene un movimiento en sentido horario aún cuando un eje tenga una posición especifica incremental negativa. El punto final no es necesario para encajar dentro de la posición absoluta definida por el desbobinado giratorio de los ejes.

Arcos tridimensionales

Para los sistemas de coordenadas que tienen tres ejes primarios asociados a ellos, es posible crear arcos tridimensionales.

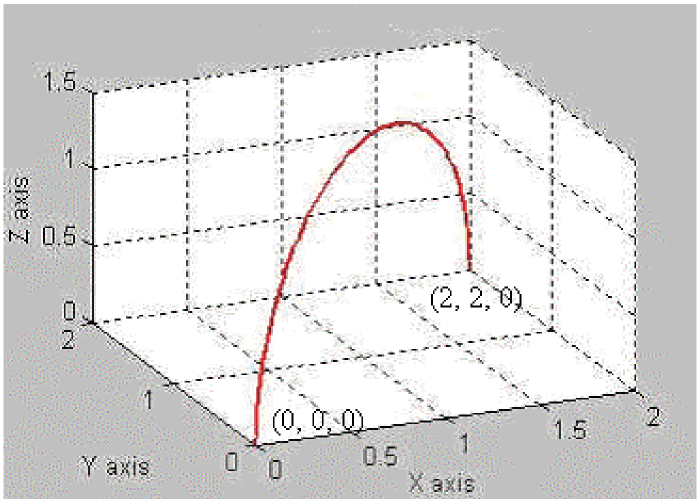
Arco tridimensional utilizando MCCM con Circle Type Via

Los siguientes ejemplos muestran el uso de MCCM con Circle Type Via y Move Type Absolute para crear un arco tridimensional. Los supuestos básicos son:

- los tres ejes, Axis0 y Axis1, Axis2 son miembros del sistema de coordenadas, coordinate_sys.
- coordinate_sys es un sistema de coordenadas tridimensional.
- Axis0, Axis1 y Axis2 son ortogonales entre sí.
- coordinated_sys se inicia a (0.0, 0.0, 0.0) unidades.

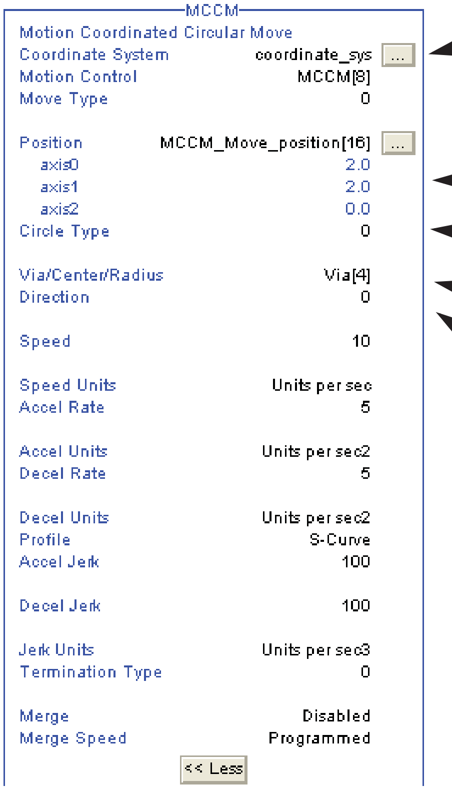
Mueve el Coordinated_sys1 a lo largo de un arco a (2.0, 2.0, 0.0) unidades atravesando el punto (1.0, 1.0, 1.414) unidades a velocidad vectorial de 10.0 unidades por segundo con los valores de aceleración y desaceleración de 5.0 unidades por segundo^2. El siguiente gráfico indica el arco 3D generado por la información mencionada anteriormente.

Arco tridimensional utilizando Circle Type Via



Se logra una ruta utilizando una instrucción MCCM con Move Type Absolute y Circle Type Via. Cuando se elige Via, la posición Via/ Center/Radius define un punto a través del cual debe pasar el arco.

Instrucción de lógica de escalera MCCM para Arco tridimensional utilizando Circle Type Via



MCCM
Motion Coordinated Circular Move
Coordinate System coordinate_sys
Motion Control MCCM[8]
Move Type 0
Position MCCM_Move_position[16]
axis0 2.0
axis1 2.0
axis2 0.0
Circle Type 0
Via/Center/Radius Via[4]
Direction 0
Speed 10
Speed Units Units per sec
Accel Rate 5
Accel Units Units per sec2
Decel Rate 5
Decel Units Units per sec2
Profile S-Curve
Accel Jerk 100
Decel Jerk 100
Jerk Units Units per sec3
Termination Type 0
Merge Disabled
Merge Speed Programmed

Sistema de coordenadas tridimensional.

Posición definida en unidades absolutas.

Circle Type es Via.

Posición Via definida en unidades absolutas como (1.0,1.0,1.414).

Direction es omitida para Via Circle Type.

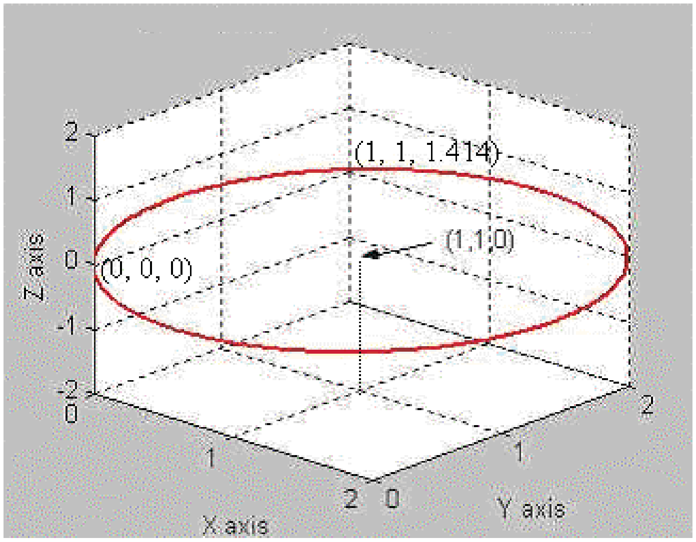
Arco tridimensional utilizando MCCM con Circle Type Center

Los siguientes ejemplos muestran el uso de MCCM con Circle Type Center y Move Type Absolute para crear un arco tridimensional. Los supuestos básicos son:

- los tres ejes, Axis0 y Axis1, Axis2 son miembros del sistema de coordenadas, coordinate_sys.
- coordinate_sys es un sistema de coordenadas tridimensional.
- Axis0, Axis1 y Axis2 son ortogonales entre sí.
- coordinate_sys se inicia a (0.0,0.0,0.0) unidades.

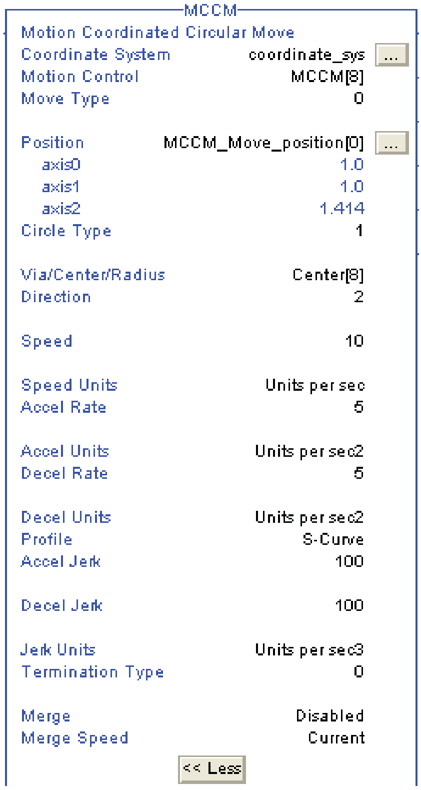
Mueve el Coordinated_sys1 a lo largo de un arco a (1.0,1.0, 1.414 unidades con centro en (1.0, 1.0, 1.0) unidades a velocidad vectorial de 10.0 unidades por segundo con los valores de aceleración y desaceleración de 5.0 unidades por segundo^2. El siguiente gráfico indica el arco tridimensional generado por la información mencionada anteriormente.

Gráfico tridimensional utilizando la dirección más corta completa para el operando Direction



Se logra esta ruta utilizando una instrucción MCCM con Move Type Absolute y Circle Type Center. Cuando se elige Via, la posición Via/Center/Radius define un punto a través del cual debe pasar el arco.

Instrucción de lógica de escalera MCCM para Arco tridimensional utilizando Circle Type Center



MCCM
Motion Coordinated Circular Move

Coordinate System	coordinate_sys
Motion Control	MCCM[8]
Move Type	0
Position	MCCM_Move_position[0]
axis0	1.0
axis1	1.0
axis2	1.414
Circle Type	1
Via/Center/Radius	Center[8]
Direction	2
Speed	10
Speed Units	Units per sec
Accel Rate	5
Accel Units	Units per sec2
Decel Rate	5
Decel Units	Units per sec2
Profile	S-Curve
Accel Jerk	100
Decel Jerk	100
Jerk Units	Units per sec3
Termination Type	0
Merge	Disabled
Merge Speed	Current

Sistema de coordenadas tridimensional.

Posición definida en unidades absolutas.

Circle Type es center.

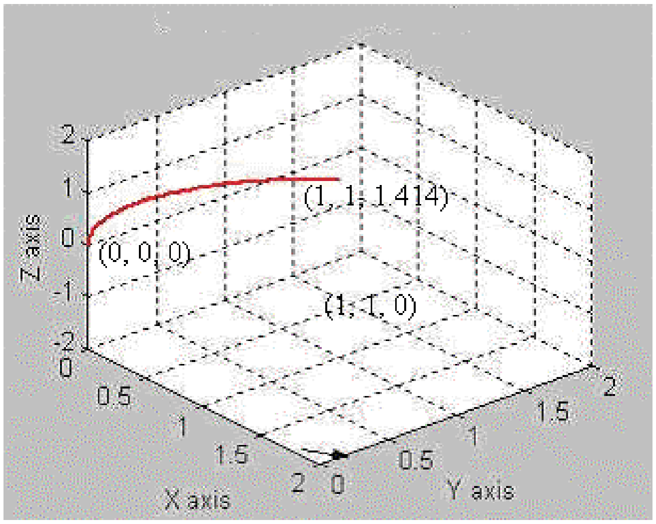
Posición Center definida en unidades absolutas como (1.0,1.0,0.0).

La dirección es la más corta completa.

Para círculos completos, establecer el operando Position en cada punto, excepto en el punto de inicio y utilizar uno de los tipos de dirección completa. Se presume que el punto final es el punto de inicio. Esto sucede por que en el espacio tridimensional se necesitan tres puntos para especificar un plano para el círculo.

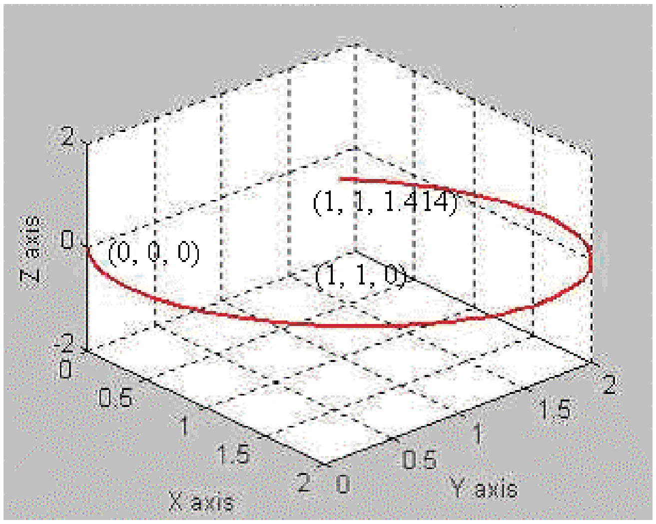
Al cambiar el operando Direction a la dirección más corta en la instrucción MCCM anterior, se genera la siguiente ruta. La opción más corta del operando Direction toma la ruta más corta desde el punto de inicio hasta el punto definido por el operando Position de la instrucción MCCM.

Gráfico 3D utilizando la dirección más corta para el operando Direction



Al cambiar el operando Direction a la dirección más larga en la instrucción MCCM anterior y la ruta seguida es la más larga desde el punto de inicio hasta el punto definido por el operando Position en la instrucción MCCM. Vea el siguiente diagrama para un ejemplo de ruta más larga.

Gráfico 3D utilizando la dirección más larga para el operando Direction



Via/Center/Radius

De acuerdo al Move Type y Circle Type seleccionado, el parámetro de posición via/center/radius define el valor absoluto o incremental de una posición a lo largo del círculo, del centro del círculo, o del radio del círculo como se define en la siguiente tabla. Si Circle Type es via o center, el parámetro de posición via/center/radius es una matriz unidimensional, cuya dimensión es definida por, al menos, el mismo número de ejes establecidos en el sistema de coordenadas. Si Circle type es radius, el parámetro de posición via/center/radius es un valor simple.

Tipo de movimiento	Circle Type	Comportamiento
Absoluto	Via	La matriz de la posición via/center/radius define una posición a lo largo del círculo. Para un caso de círculo incompleto, la matriz del parámetro de posición define el punto final del arco. Para un caso de círculo completo, la matriz del parámetro de posición define un segundo punto a lo largo del círculo, excepto el punto final.
Incremental	Via	La suma de la matriz de posición via/center/radius y la posición anterior define la posición a lo largo del círculo. Para un caso de círculo incompleto, la suma de la matriz del parámetro de posición y la posición anterior definen el punto final del arco. Para un caso de círculo completo, la suma de la matriz del parámetro de posición y la posición anterior definen un segundo punto a lo largo del círculo, excepto el punto final.
Absoluto	Center	La matriz de la posición via/center/radius define el centro del círculo. Para un caso de círculo incompleto, la matriz del parámetro de posición define el punto final del arco. Para un caso de círculo completo, la matriz del parámetro de posición define un segundo punto a lo largo del círculo, excepto el punto final.
Incremental	Center	La suma de la matriz de posición via/center/radius y la posición anterior define el centro del círculo. Para un caso de círculo incompleto, la suma de la matriz del parámetro de posición y la posición anterior definen el punto final del arco. Para un caso de círculo completo, la suma de la matriz del parámetro de posición y la posición anterior definen un segundo punto a lo largo del círculo, excepto el punto final.
Absolute o Incremental	Radius	El valor simple de la posición via/center/radius define el radio del arco. La señal del valor se utiliza para determinar el punto central para distinguir entre los dos arcos posibles. Un valor positivo indica un punto central que genera un arco menor que 180 grados. Un valor negativo indica un punto central que genera un arco mayor que 180 grados. Este Circle Type es válido sólo para círculos bidimensionales. La matriz del parámetro de posición sigue al Move Type para definir el punto final del arco.

Tipo de movimiento	**Circle Type**	**Comportamiento**
Absoluto	Center Incremental	La suma de la matriz de posición via/center/radius y la posición anterior define la posición central del círculo. Para un caso de círculo incompleto, la matriz del parámetro de posición define el punto final del arco. Para un caso de círculo completo, la matriz del parámetro de posición define un segundo punto a lo largo del círculo, excepto el punto final.
Incremental	Center Incremental	La suma de la matriz de posición via/center/radius y la posición anterior define la posición central del círculo. Para un caso de círculo incompleto, la suma de la matriz del parámetro de posición y la posición anterior definen el punto final del arco. Para un caso de círculo completo, la suma de la matriz del parámetro de posición y la posición anterior definen un segundo punto a lo largo del círculo, excepto el punto final.

Direction

El operando Direction define la dirección giratoria de un movimiento circular 2D tanto en sentido horario como contrahorario, de acuerdo al sentido del tornillo a la derecha. Para un movimiento circular 3D, la dirección es tanto la más corta como la más larga. Tanto en 2D como 3D, puede también indicar si el movimiento circular es un círculo completo.

Speed

El operando Speed define la velocidad vectorial máxima a lo largo de la ruta del movimiento coordinado.

Speed Units

El operando Speed Units define las unidades aplicadas al operando Speed ya sea directamente en unidades de coordinación o como un porcentaje de los valores máximos definidos en el sistema de coordenadas.

Accel Rate

El operando Accel Rate define la aceleración máxima a lo largo de la ruta del movimiento coordinado.

Accel Units

El operando Accel Units define las unidades aplicadas al operando Accel Rate ya sea directamente en unidades de coordinación del sistema de coordenadas establecido o como un porcentaje de los valores máximos definidos en el sistema de coordenadas.

Decel Rate

El operando Decel Rate define la desaceleración máxima a lo largo de la ruta del movimiento coordinado.

Decel Units

El operando Decel Units define las unidades aplicadas al operando Decel Rate ya sea directamente en unidades de coordinación del sistema de coordenadas establecido o como un porcentaje de los valores máximos definidos en el sistema de coordenadas.

Profile

El operando Profile determina si el movimiento coordinado utiliza un perfil de velocidad trapezoidal o con curva en S. Para obtener más información sobre perfiles Trapezoidales y con curva en S, consulte la sección Profile de la instrucción MCLM en página 286.

Accel Jerk

Accel Jerk define el jaloneo de aceleración máximo para el movimiento programado. Para más información sobre cálculos de Accel Jerk, consulte la sección de Unidades de jaloneo en este capítulo.

Decel Jerk

Decel Jerk define el jaloneo de desaceleración máximo para el movimiento programado. Para más información sobre cálculos de Decel Jerk, consulte la sección de Unidades de jaloneo en este capítulo.

Jerk Units

Jerk units define las unidades que se aplican a los valores ingresados en los operandos Accel Jerk y Decel Jerk. Los valores se ingresan directamente en las unidades de posición del sistema de coordenadas o como un porcentaje. Al configurar utilizando % de Máximo, el jaloneo se aplica como porcentaje de los operando Maximum Acceleration Jerk y Maximum Deceleration Jerk establecidos en los atributos del sistema de coordenadas. Al configurar utilizando % de Tiempo, el valor es un porcentaje basado en Speed, Accel Rate, y Decel Rate establecidos en la instrucción.

Si quiere convertir unidades de ingeniería a % de Tiempo, utilice estas ecuaciones.

Para jaloneo de acel:

$$j_a\,[\mathrm{EU/s}^3] = \frac{a_{\max}{}^2\,[\mathrm{EU/s}^2]}{v_{\max}\,[\mathrm{EU/s}]}\left(\frac{200}{j_a\,[\%\text{ of time}]} - 1\right)$$

Para jaloneo de desacel:

$$j_a\,[\mathrm{EU/s}^3] = \frac{a_{\max}{}^2\,[\mathrm{EU/s}^2]}{v_{\max}\,[\mathrm{EU/s}]}\left(\frac{200}{j_a\,[\%\text{ of time}]} - 1\right)$$

Si quiere convertir % de Tiempo a Unidades de ingeniería, utilice estas ecuaciones.

Para jaloneo de acel:

$$j_a\,[\%\text{ of time}] = \frac{2}{1 + \dfrac{j_a\,[\mathrm{EU/s}^3]\;v_{\max}\,[\mathrm{EU/s}]}{a_{\max}^2\,[\mathrm{EU/s}^2]}}\,100$$

Para jaloneo de desacel:

$$j_d\,[\%\text{ of time}] = \frac{2}{1 + \dfrac{j_d\,[\mathrm{EU/s}^3]\;v_{\max}\,[\mathrm{EU/s}]}{d_{\max}^2\,[\mathrm{EU/s}^2]}}\,100$$

Consideración importante

Si usted programa círculos tangentes con diferentes regímenes de jaloneo (Jaloneo de desaceleración del primer círculo y Jaloneo de aceleración del segundo círculo), entonces puede obtener una ligera discontinuidad de velocidad en la intersección de los dos círculos. El tamaño de la discontinuidad depende de la magnitud de la diferencia de jaloneo. En otras palabras, mientras la diferencia de jaloneo sea más pequeña, más pequeña será la perturbación de la velocidad. Por lo tanto, recomendamos no programar los regímenes de jaloneo en círculos tangentes.

Tipo de terminación

Para información sobre la selección de tipos de terminación, consulte la página 264 de este manual.

Merge

El operando Merge determina si cambiar o no el movimiento de todos los ejes específicos en un movimientocoordinado puro. Las opciones son: Merge inhabilitado, movimiento coordinado, o Todo el movimiento.

- Merge inhabilitado – Cualquier instrucción de movimiento de un eje simple ejecutada actualmente que involucre cualquier eje definido en el sistema de coordenadas especifico no será afectada por la activación de esta instrucción, y traerá como resultado el movimiento superpuesto en los ejes afectados. Se señala un error si la segunda instrucción comienza en el mismo sistema de coordenadas o en otro sistema de coordenadas que contenga ejes en común con el sistema de coordenadas activo.

- Movimiento coordinado – Cualquier instrucción de control de movimiento ejecutada actualmente que involucre el mismo sistema de coordenadas especifico será interrumpida, y el movimiento activo es combinado en el movimiento actual a la velocidad definida en el parámetro de velocidad de incorporación. Cualquier instrucción de control de movimiento coordinado pendiente en el sistema de coordenadas especifico se cancela. Cualquier instrucción de control de movimiento de un eje simple ejecutada actualmente que involucre cualquier eje definido en el sistema de coordenadas especifico no será afectada por la activación de esta instrucción, y traerá como resultado el movimiento superpuesto en los ejes afectados.

- Todos los movimientos – Cualquier instrucción de movimiento de un eje simple ejecutada actualmente que involucre cualquier eje definido en el sistema de coordenadas específico y cualquier instrucción de movimiento coordinado ejecutada actualmente han sido interrumpidas. El movimiento previo se incorpora en el movimiento actual a la velocidad definida en el parámetro de Merge Speed. Cualquier instrucción de movimiento coordinado pendiente se cancela.

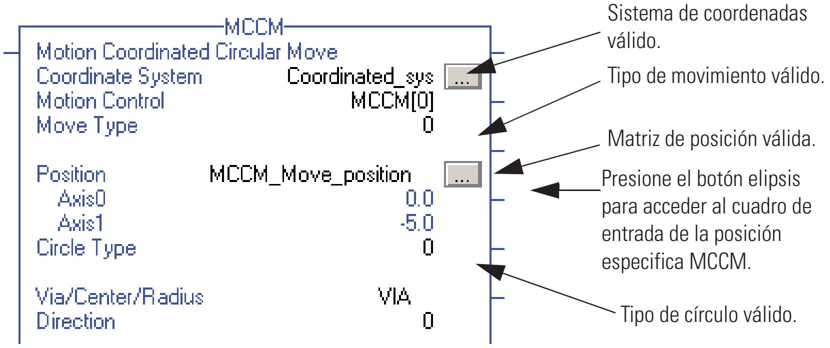
Merge Speed

El operando Merge Speed define si la velocidad actual o la velocidad programada se utiliza como velocidad máxima a lo largo de la ruta del movimiento coordinado cuando Merge es habilitado. La velocidad actual es la suma vectorial de todos los movimientos (por ejemplo, impulsos, MAM y movimiento acoplado) para los ejes definidos en el sistema de coordenadas actual.

Cuadro de diálogo de entrada de la posición especifica MCCM

Para acceder al cuadro de diálogo de entrada de la posición especifica MCCM presione el botón elipsis a la derecha del operando Position de la plantilla de instrucción de lógica de escalera. Sólo se puede acceder al cuadro de diálogo de entrada de la posición especifica si el sistema de coordenadas para la instrucción ha sido nombrado, tiene un nombre del tag para el operando Position que contiene suficientes elementos para aceptar el número de ejes, un Move Type y un Circle Type válidos han sido seleccionados. Si no se cumple con estos requisitos, aparece un mensaje de error en la barra de estado

Valores válidos de la lógica de escalera de MCCM para acceder al cuadro de entrada de la posición especifica.



MCCM
Motion Coordinated Circular Move
Coordinate System Coordinated_sys
Motion Control MCCM[0]
Move Type 0
Position MCCM_Move_position
Axis0 0.0
Axis1 -5.0
Circle Type 0
Via/Center/Radius VIA
Direction 0

Sistema de coordenadas válido.

Tipo de movimiento válido.

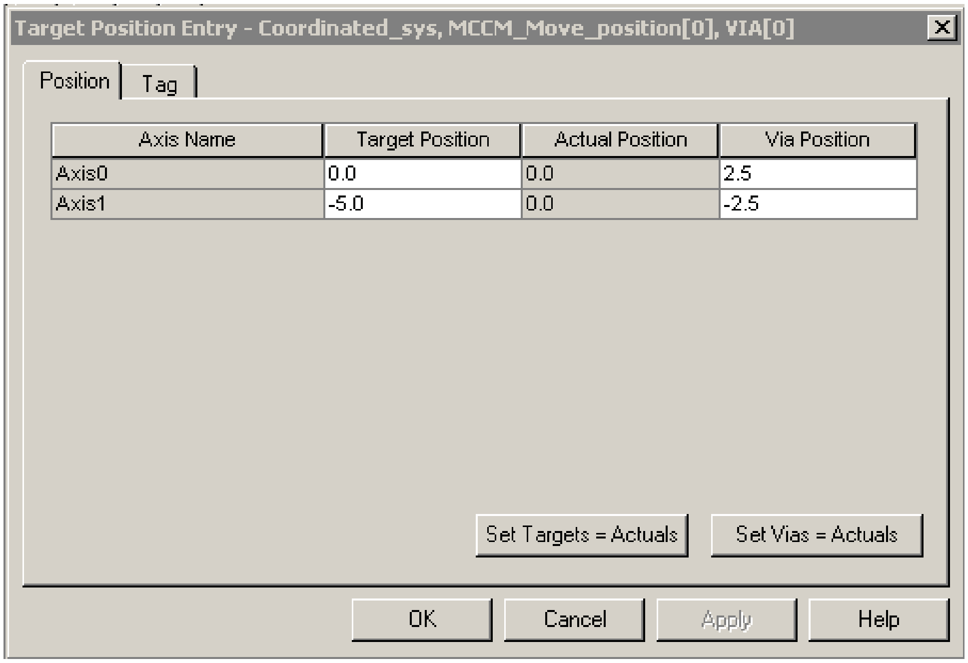
Matriz de posición válida.

Presione el botón elipsis para acceder al cuadro de entrada de la posición especifica MCCM.

Tipo de círculo válido.

Presione elipsis y aparece el siguiente cuadro de diálogo.

Cuadro de diálogo de entrada de la posición especifica de la instrucción MCCM – Ficha de Posición



Target Position Entry - Coordinated_sys, MCCM_Move_position[0], VIA[0]

Axis Name	Target Position	Actual Position	Via Position
Axis0	0.0	0.0	2.5
Axis1	-5.0	0.0	-2.5

Campos del cuadro de diálogo de entrada de la posición especifica

Característica	Descripción
Nombre del eje	Esta columna tiene los nombres de cada eje en el sistema de coordenadas nombrados en la plantilla de lógica de escalera. Estos nombres no se pueden editar.
Posición específica/ Incremento específico	Los valores en esta columna son numéricos. Indican el punto final o la partida incremental del movimiento dependiendo del Move Type activo. La columna de encabezado indica lo qué se muestra en pantalla.
Posición real	Esta columna contiene las posiciones reales actuales de los ejes en el sistema de coordenadas. Estos valores se actualizan dinámicamente cuando la actualización del tag del sistema de coordenadas y en línea se habilita.
Via Position/Via Increment Center Position/Center Increment Radius	De acuerdo al Circle Type seleccionado, esta columna contiene la posición o incremento del punto Via, la posición o incremento de Center.
Objetivos configurados = Actuals	Este botón se habilita cuando Move Type es Absolute y se utiliza para copiar el valor desde los campos de Posición real a los campos de Posición especifica.
Vias configurados = Actuals	Este botón se activa solamente si Move Type es Absolute. Se utiliza para copiar los valores desde los campos de Posición real a los campos Vias.

Move Type y Circle Type seleccionado determinada la apariencia del cuadro de diálogo. La siguiente tabla ilustra cómo cambia la pantalla mediante las combinaciones de Move Type y Circle Type seleccionados.

Cambios del cuadro de diálogo de entrada de la posición especifica

Tipo de movimiento	Circle Type	Comportamiento
Absoluto	Via	La columna Objetivo cambia a Posición especifica. La columna Via cambia a Posición Via. Objetivos configurados = Botón Actuals está activo. Vias configuradas = Botón Actuals está activo.
Incremental	Via	La columna Objetivo cambia a Incremento especifico. La columna Via cambia a Incremento Via. Objetivos configurados = Botón Actuals está inactivo (Difuminado). Vias configuradas = Botón Actuals está inactivo (Difuminado).
Absoluto	Center	La columna Objetivo cambia a Posición especifica. La columna Center cambia a Posición central. Objetivos configurados = Botón Actuals está activo. Vias configuradas = Botón Actuals está activo.

Cambios del cuadro de diálogo de entrada de la posición especifica

Tipo de movimiento	Circle Type	Comportamiento
Incremental	Center	La columna Objetivo cambia a Incremento especifico. La columna Center cambia a Incremento central. Objetivos configurados = Botón Actuals está inactivo (Difuminado). Vias configuradas = Botón Actuals está inactivo (Difuminado).
Absoluto	Radius	La columna Objetivo cambia a Posición especifica. La columna Radius cambia a Radius. Objetivos configurados = Botón Actuals está activo. Vias configuradas = Botón Actuals está inactivo (Difuminado).
Incremental	Radius	La columna Objetivo cambia a Incremento especifico. La columna Radius cambia a Radius. Objetivos configurados = Botón Actuals está inactivo (Difuminado). Vias configuradas = Botón Actuals está inactivo (Difuminado).
Absoluto	Center Incremental	La columna Objetivo cambia a Posición especifica. La columna Incremento central cambia a Center Incremental. Objetivos configurados = Botón Actuals está activo. Vias configuradas = Botón Actuals está inactivo (Difuminado).
Incremental	Center Incremental	La columna Objetivo cambia a Incremento especifico. La columna Incremento central cambia a Center Incremental. Objetivos configurados = Botón Actuals está inactivo (Difuminado). Vias configuradas = Botón Actuals está inactivo (Difuminado).

MCCM es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritméticos:

no afectados

Condiciones de fallo:

ninguno

Códigos de error:

Consulte Códigos de error (ERR) para las instrucciones de control de movimiento.

Códigos de error extendidos:

Los códigos de error extendido ayudan a definir, además, el mensaje de error dado por esta instrucción particular. Este comportamiento depende del código de error al cual son asociados.

Los códigos de error extendido para Error de estado servo desactivado (5), Error del estado de desactivación (7), Tipo de eje no servo (8), Eje no configurado (11), Error de comando Home (16), y Tipo de dato de eje no válido (38) funcionan de la misma manera. Un número entre 0…*n* aparece en pantalla para el Código de error extendido. Este número es el índice del Sistema de Coordenadas indicando el eje que está en condición de error.

Para el código de error de Eje no configurado (11) hay un valor adicional de –1, que indica que el Sistema de coordenadas era incapaz de configurar el eje para el movimiento coordinado.

Para la instrucción MCCM, Código de error 13 – Parámetro fuera de rango, los Errores extendidos señalan un número que indica el parámetro en error como establecido en la plantilla en orden numérico de arriba hacia abajo comenzando de cero. Por ejemplo, 2 indica el valor del parámetro para Move Type en error.

Código de error y (número)	Indicador numérico del error extendido	Parámetro de instrucción	Descripción
Parámetro fuera de Rango (13)	0	Sistema de coordenadas	El número de ejes primarios no es 2 ni 3.
Parámetro fuera de Rango (13)	2	Move Type	Move Type es menor que 0 o mayor que 1.
Parámetro fuera de Rango (13)	3	Position	La matriz de posición no es lo suficientemente grande como para proporcionar posiciones para todos los ejes en el sistema de coordenadas.
Parámetro fuera de Rango (13)	4	Circle Type	Circle Type es menor que 0 o mayor que 4.
Parámetro fuera de Rango (13)	5	Via/Center/Radius	El tamaño de la matriz Via/Center no es lo suficientemente grande para proporcionar las posiciones para todos los ejes en el punto via/center definido.
Parámetro fuera de Rango (13)	6	Direction	Direction es menor que 0 o mayor que 3.
Parámetro fuera de Rango (13)	7	Speed	La velocidad es menor a 0.
Parámetro fuera de Rango (13)	9	Accel Rate	Accel Rate es menor o igual a 0.
Parámetro fuera de Rango (13)	11	Decel Rate	Decel Rate es menor o igual a 0.
Parámetro fuera de Rango (13)	14	Tipo de terminación	El tipo de terminación es menor que 0 o mayor que 3.

Para el código de error 54 – El valor máximo de desaceleración es cero, si el error extendido devuelve un número positivo (0-*n*), está haciendo referencia al eje en error en el sistema de coordenadas. Diríjase a la ficha general Coordinate System Properties y mire debajo de la columna de corchetes ([ ]) de la cuadrícula del eje para determinar qué eje tiene un valor máximo de desaceleración de 0. Haga clic en el botón de elipsis al lado del eje en error para acceder a la pantalla de propiedades del eje. Diríjase a la ficha Dynamics y haga el cambio apropiado al valor máximo de desaceleración. Si el número del error extendido es –1, esto significa que el sistema de coordenadas tiene un valor máximo de desaceleración de 0. Diríjase a la ficha Coordinate System Properties Dynamics para corregir el valor máximo de desaceleración.

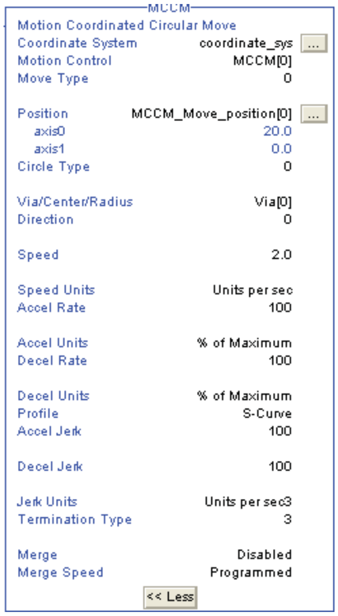
Ejemplos de error circular

Debido a la complejidad de la instrucción MCCM y a los códigos de error que puede generar, los siguientes ejemplos simples sirven de ayuda para comprender la instrucción MCCM.

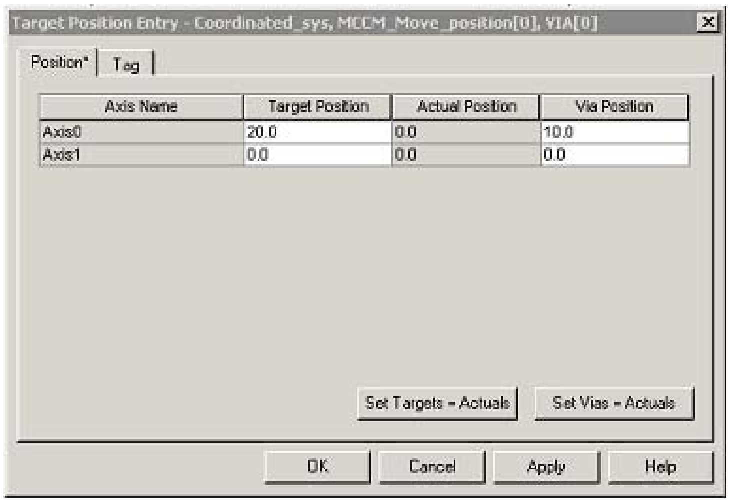
Ejemplo de CIRCULAR_COLLINEARITY_ERROR (44)

El siguiente ejemplo para error #44 indica una situación donde el punto de inicio, el punto via y el punto final se encuentran en una línea recta. El programa trata de generar un arco bidimensional desde 0,0 (posición actual) a 20,0 a través de la ubicación 10,0. Ya que estos puntos se encuentran en una línea recta, no se puede calcular un punto central circular para el círculo. Este error también se puede generar si el programa para un tipo de círculo tridimensional utiliza un punto de inicio, un punto central y un punto final en una línea recta. En este caso, un número infinito de círculos pueden encajar en los puntos programados en un número infinito de planos.

Programa de lógica de escalera y pantalla de entrada especifica que generan error #44.



Coordinate System	coordinate_sys
Motion Control	MCCM[0]
Move Type	0
Position	MCCM_Move_position[0]
axis0	20.0
axis1	0.0
Circle Type	0
Via/Center/Radius	Via[0]
Direction	0
Speed	2.0
Speed Units	Units per sec
Accel Rate	100
Accel Units	% of Maximum
Decel Rate	100
Decel Units	% of Maximum
Profile	S-Curve
Accel Jerk	100
Decel Jerk	100
Jerk Units	Units per sec3
Termination Type	3
Merge	Disabled
Merge Speed	Programmed

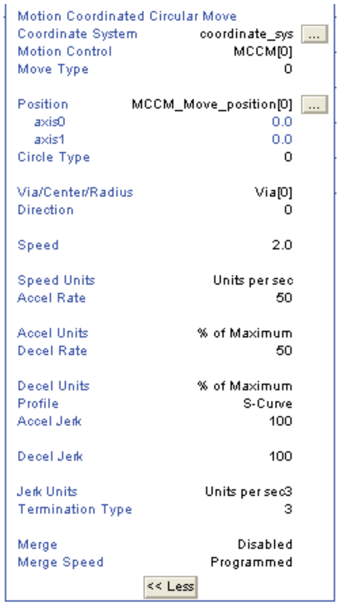


Axis Name	Target Position	Actual Position	Via Position
Axis0	20.0	0.0	10.0
Axis1	0.0	0.0	0.0

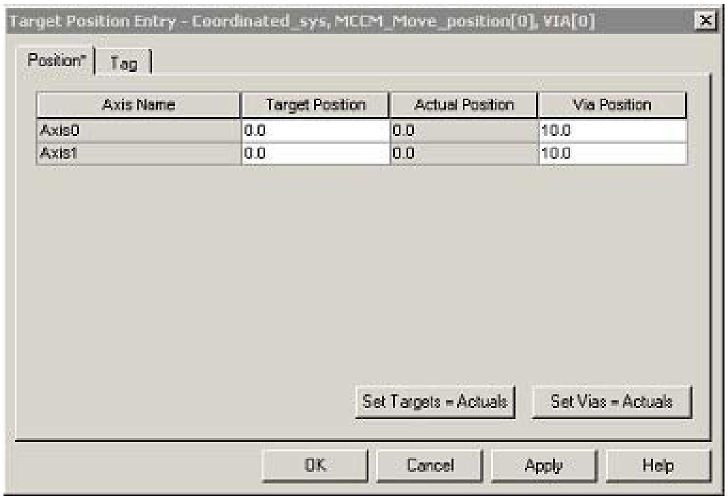
Ejemplo de CIRCULAR_START_END_ERROR (45)

El siguiente ejemplo para el error #45 describe una situación donde el punto de inicio y via son los mismo. El programa trata de generar un círculo completo bidimensional desde 0,0 (posición actual) hacia 0,0 a través de la ubicación 10,10. Ya que el punto de inicio y via son los mismo, no se puede encontrar un punto centrar circular para este círculo.

Programa de lógica de escalera y pantalla de entrada especifica que generan error #45



Motion Coordinated Circular Move	
Coordinate System	coordinate_sys
Motion Control	MCCM[0]
Move Type	0
Position	MCCM_Move_position[0]
axis0	0.0
axis1	0.0
Circle Type	0
Via/Center/Radius	Via[0]
Direction	0
Speed	2.0
Speed Units	Units per sec
Accel Rate	50
Accel Units	% of Maximum
Decel Rate	50
Decel Units	% of Maximum
Profile	S-Curve
Accel Jerk	100
Decel Jerk	100
Jerk Units	Units per sec3
Termination Type	3
Merge	Disabled
Merge Speed	Programmed



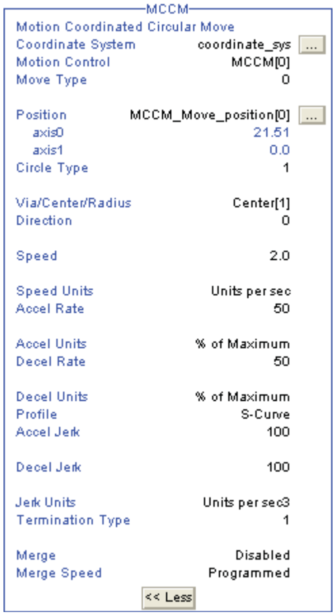
Target Position Entry - Coordinated_sys, MCCM_Move_position[0], VIA[0]

Axis Name	Target Position	Actual Position	Via Position
Axis0	0.0	0.0	10.0
Axis1	0.0	0.0	10.0

Ejemplo CIRCULAR_R1_R2_MISMATCH_ERROR (46)

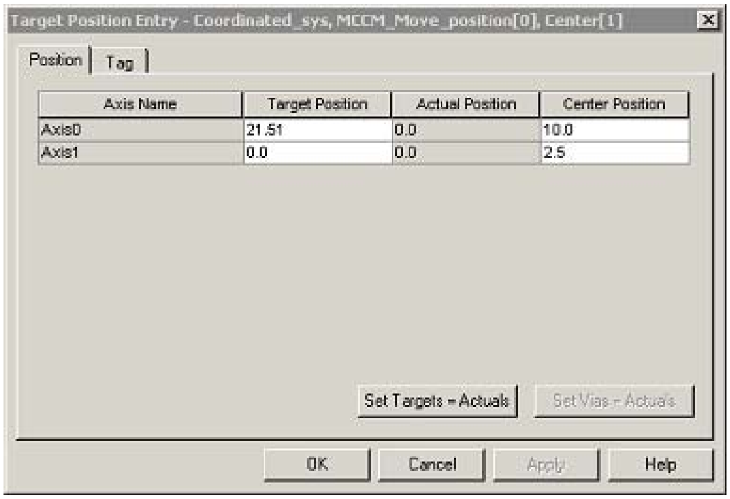
El siguiente ejemplo para el error #46 muestra una situación donde la diferencia en las longitudes inicio/final radiales exceden el 15% de la longitud radial de inicio. El programa trata de generar un arco bidimensional desde 0,0 (posición actual) hacia 21.51,0 usando un punto centrad en 10,10. Ya que la diferencia de las longitdes inicio/final radiales es de 21.51 – 10 = 1.51, el mismo excede el 15% de la longitud radial de inicio.15 * 10 = 1.5. Ya que el punto final ha sido de 21.5, este ejemplo funciona, y el punto central podría haber sido recalculado para permanecer exactamente en el punto medio entre los puntos de inicio y final.

Programa de lógica de escalera y pantalla de entrada especifica que generan error #46



Motion Coordinated Circular Move

Coordinate System	coordinate_sys
Motion Control	MCCM[0]
Move Type	0
Position	MCCM_Move_position[0]
axis0	21.51
axis1	0.0
Circle Type	1
Via/Center/Radius	Center[1]
Direction	0
Speed	2.0
Speed Units	Units per sec
Accel Rate	50
Accel Units	% of Maximum
Decel Rate	50
Decel Units	% of Maximum
Profile	S-Curve
Accel Jerk	100
Decel Jerk	100
Jerk Units	Units per sec3
Termination Type	1
Merge	Disabled
Merge Speed	Programmed



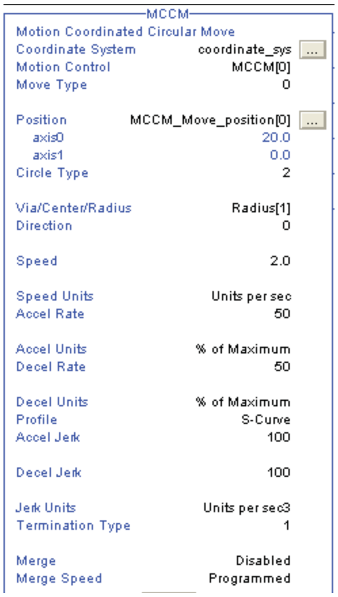
Target Position Entry - Coordinated_sys, MCCM_Move_position[0], Center[1]

Axis Name	Target Position	Actual Position	Center Position
Axis0	21.51	0.0	10.0
Axis1	0.0	0.0	2.5

Ejemplo de CIRCULAR_SMALL_R_ERROR (49)

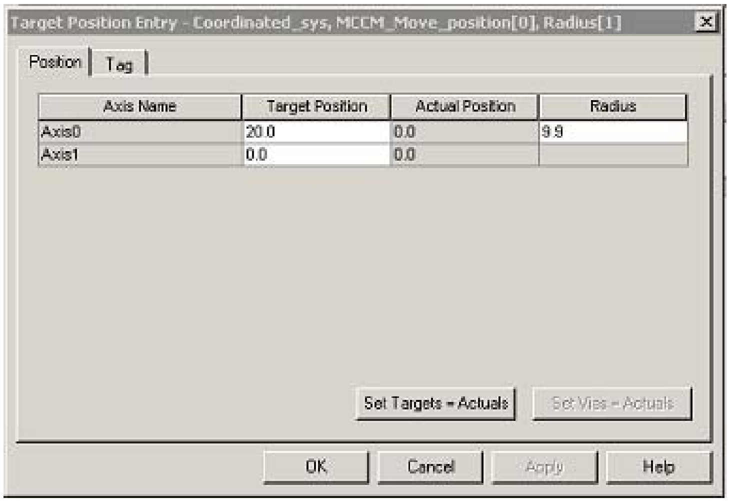
El primer ejemplo del error #49 describe una situación donde type circle radius utiliza un radio que es demasiado corto para extender la distancia entre el punto de inicio y el punto final. El programa trata de generar un arco bidimensional desde 0,0 (posición actual) hacia 20,0. Sin embargo, el usuario trata de programar un type circle radius con un radio que es demasiado corto para extender la distancia entre el punto de inicio y el punto final.

Programa de lógica de escalera y pantalla de entrada especifica que generan error #49



Motion Coordinated Circular Move

Coordinate System	coordinate_sys
Motion Control	MCCM[0]
Move Type	0
Position	MCCM_Move_position[0]
axis0	20.0
axis1	0.0
Circle Type	2
Via/Center/Radius	Radius[1]
Direction	0
Speed	2.0
Speed Units	Units per sec
Accel Rate	50
Accel Units	% of Maximum
Decel Rate	50
Decel Units	% of Maximum
Profile	S-Curve
Accel Jerk	100
Decel Jerk	100
Jerk Units	Units per sec3
Termination Type	1
Merge	Disabled
Merge Speed	Programmed



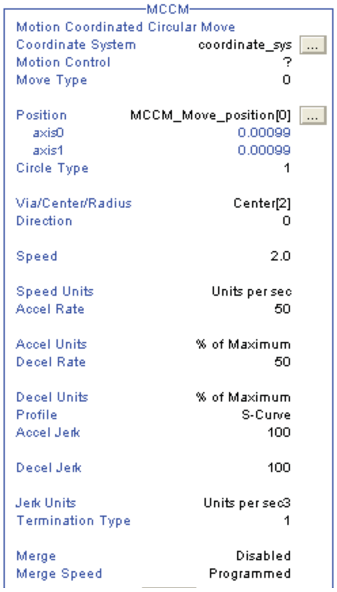
Target Position Entry - Coordinated_sys, MCCM_Move_position[0], Radius[1]

Axis Name	Target Position	Actual Position	Radius
Axis0	20.0	0.0	9.9
Axis1	0.0	0.0	

Ejemplo de CIRCULAR_SMALL_R_ERROR (49)

Este segundo ejemplo del error #49 muestra una situación donde type circle radius utiliza un radio de magnitud menor que 0.001. El programa trata de generar un arco bidimensional desde 0,0 (posición actual) a 0.00099,0.00099. Este error sucede ya que el usuario trata de programar un type circle radius con un radio de magnitud menor que 0.001 unidades.

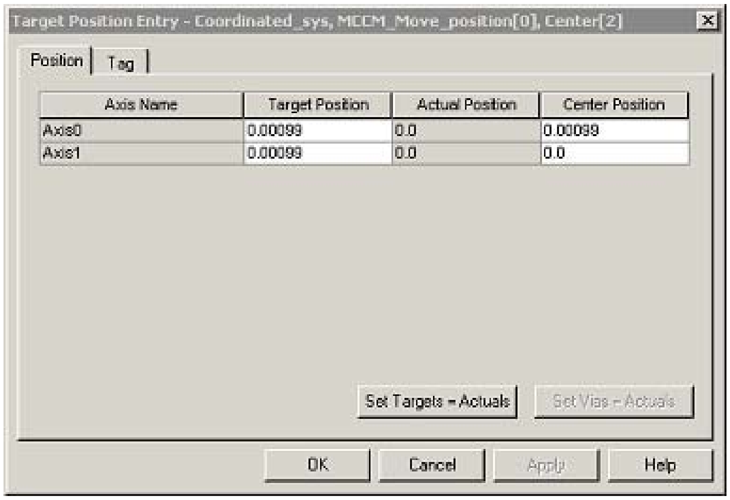
Programa de lógica de escalera y pantalla de entrada especifica que generan error #49



MCCM

Motion Coordinated Circular Move

Coordinate System	coordinate_sys
Motion Control	?
Move Type	0
Position	MCCM_Move_position[0]
axis0	0.00099
axis1	0.00099
Circle Type	1
Via/Center/Radius	Center[2]
Direction	0
Speed	2.0
Speed Units	Units per sec
Accel Rate	50
Accel Units	% of Maximum
Decel Rate	50
Decel Units	% of Maximum
Profile	S-Curve
Accel Jerk	100
Decel Jerk	100
Jerk Units	Units per sec3
Termination Type	1
Merge	Disabled
Merge Speed	Programmed



Target Position Entry - Coordinated_sys, MCCM_Move_position[0], Center[2]

Axis Name	Target Position	Actual Position	Center Position
Axis0	0.00099	0.0	0.00099
Axis1	0.00099	0.0	0.0

MCCM Cambia a bits de estado:

Los Bit de estado proporcionan un medio para monitorear el progreso de la instrucción de movimiento. Hay tres tipos de bits de estado que brindan la información pertinente.

- AXIS
- Sistema de coordenadas
- Movimiento coordinado

Cuando la instrucción MCCM comienza, los bits de estado sufren los siguientes cambios.

Bits de estado del eje

Nombre del bit	Significado
CoordinatedMotionStatus	Se establece cuando se ejecuta la instrucción MCCM y desaparece cuando finaliza la instrucción.

Bits de estado del sistema de coordenadas

Nombre del bit	Significado
MotionStatus	Se establece cuando la instrucción MCCM está activa y el Sistema de Coordenadas está conectado a sus ejes asociados.

Bits de estado del movimiento coordinado

Nombre del bit	Significado
AccelStatus	Se establece cuando el vector está acelerando. Se borra cuando una incorporación está en proceso o cuando es movimiento vectorial está en velocidad o desacelerando.
DecelStatus	Se establece cuando el vector está desacelerando. Se borra cuando una incorporación está en proceso o cuando es movimiento vectorial está desacelerando o cuando el movimiento se completa.
ActualPosToleranceStatus	Se establece sólo para el tipo de terminación de Tolerancia Real. Se establece luego de que estas dos condiciones se cumplen. 1) Interpolación está completa. 2) La distancia real al punto final programado es menor que el valor configurado de tolerancia real del sistema de coordenadas. Permanece habilitado luego de que la instrucción finaliza. Se restablece cuando una nueva instrucción comienza.
CommandPosToleranceStatus	Se establece para todos los tipos de terminación siempre que la distancia al punto final programado sea menor que el valor configurado de la tolerancia de comando del sistema de coordenadas y permanece establecido hasta que la instrucción finaliza. Se restablece cuando una nueva instrucción comienza.
StoppingStatus	El bit de estado de detención desaparece cuando se ejecuta la instrucción MCCM.
MoveStatus	Se establece cuando MCCM inicia un movimiento del eje. Borra el bit .PC de la última instrucción de control de movimiento o cuando una instrucción de control de movimiento ejecuta lo que provoca una detención.
MoveTransitionStatus	Se establece cuando el tipo de terminación No Decel o Command Tolerance se completa. Al combinar movimientos colineales, el bit no se establece, ya que la máquina está siempre en la ruta. Se borra cuando una combinación se completa, el movimiento de una instrucción pendiente comienza, o una instrucción de control de movimiento ejecuta lo que provoca una detención. No se encuentra en la ruta.
MovePendingStatus	Se establece cuando una instrucción de control de movimiento pendiente se encuentra en la cola de instrucción. Se borra cuando la cola de instrucción está vacía.
MovePendingQueueFullStatus	Se establece cuando la cola de instrucción está llena. Se borra cuando la cola tiene espacio para retener otra nueva instrucción de control de movimiento coordinado.

Actualmente, el Movimiento coordinado sólo acepta poner en cola una instrucción de movimiento coordinado. Por lo tanto, el bit MovePendingStatus y el bit MovePendingQueueFullStatus son siempre los mismos.

Guía de referencia de programación circular

Circle Type	Utilizado en 2D/3D/Ambos	Errores de validación	Dirección – 2D	Dirección – 3D	Comentarios
Radius	2D	Error 25; Instrucción no válida Error 45 Punto final = Punto de inicio Error 49; R demasiado pequeño (|R| < .001) o R demasiado corto para extenderse hacia los puntos programados.	CW/CCW visualizado desde el plano perpendicular "+" al plano circular.	N/A	Un radio "+" obliga que la longitud del arco sea <= 180° (Arco más corto). Un radio "–" obliga que la longitud del arco sea => 180° (Arco más largo). Se pueden programar círculos completos. Para círculos completos: establecer el operando Position en cada punto del círculo, excepto en el punto de inicio y utilizar uno de los tipos de dirección completa.
Punto Central	Ambos	Error 44; Colinealidad (3D solamente) Error 45; Punto final = Punto de inicio (3D solamente) Error 46; Desigualdad del radio Inicio/Final (|R1 – R2| > .15 * R1).	CW/CCW visualizado desde el plano perpendicular "+" al plano circular.	Arco más corto/más largo en círculos completos, la ubicación del punto final define las rutas más cortas/más largas referidas por el parámetro de dirección.	1. Se pueden programar círculos completos. 2. Sólo en 2D, Punto final = Punto de inicio es válido. Por lo tanto, se pueden generar círculos completos: a. Al establecer que el punto final = punto de inicio, en cuyo caso, todos los tipos de dirección producen círculos completos. b. Al establecer que el punto final no = punto de inicio y utilizar tipo de dirección completa. 3. Para círculos completos 3D: establecer el operando Position en cada punto del círculo, excepto en el punto de inicio y utilizar uno de los tipos de dirección completa. La posición define los tipos de arco y dirección más corta.
Punto Via	Ambos	Error 44; Colinealidad Error 45; Punto final = Punto de inicio	El punto Via siempre determina la dirección.	El punto Via siempre determina la dirección. El operando Direction se utiliza solamente para determinar si el círculo es parcial o completo.	1. Se pueden programar círculos completos. 2. Para círculos completos: establecer el operando Position en cada punto del círculo, excepto en el punto de inicio y utilizar uno de los tipos de dirección completa.

Operando Profile

Al utilizar esta instrucción, deberá tener en cuenta el Operando Profile. Para obtener más información, consulte la página 71.

Motion Coordinated Change Dynamics (MCCD)

La instrucción MCCD comienza un cambio en la ruta dinámica del sistema de coordenadas específico. De acuerdo a Motion Type, MCCD cambia el perfil de movimiento coordinado activo en este momento en el sistema.

ATENCIÓN

Los tags para instrucciones de control de movimiento se deben utilizar sólo una vez. Reutilizar el tag de control de movimiento en otras instrucciones puede provocar operaciones imprevistas. Esto puede dañar el equipo o causar lesiones personales.

ATENCIÓN

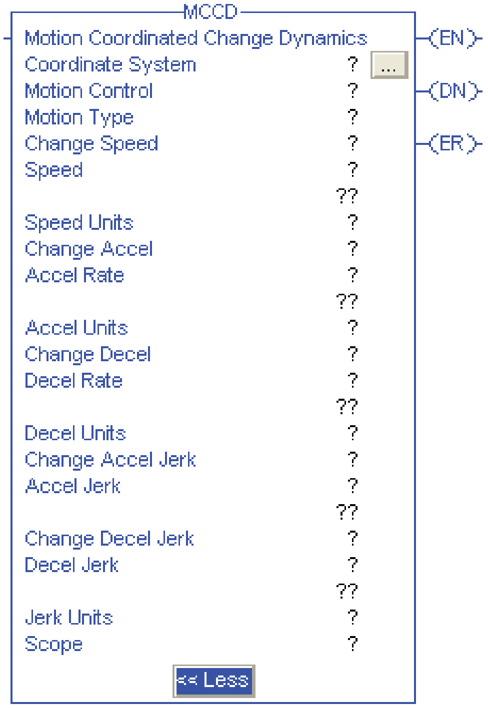
Riesgo de sobreimpulso de velocidad y/o de posición final

Si cambia los parámetros de movimiento dinámicamente mediante cualquier método, es decir cambiando la dinámica del movimiento (MCD o MCCD) o iniciando una nueva instrucción antes de que la última finalice, tenga en cuenta el riesgo de sobreimpulso de velocidad y/o de posición final.

Se puede exceder un perfil de velocidad trapezoidal si la máxima desaceleración se disminuye mientras el movimiento reduce la velocidad o está cerca del punto de desaceleración.

Se puede exceder un perfil de velocidad con curva en S si:

- la máxima desaceleración disminuye mientras el movimiento reduce la velocidad o está cerca del punto de desaceleración; o
- la máxima aceleración de jaloneo disminuye y el eje acelera. Recuerde, sin embargo, que el jaloneo se puede cambiar indirectamente si se encuentra especificado en % de tiempo.



Operandos de la instrucción MCCD – Lógica de escalera de relés

Operando	Tipo	Formato	Descripción
Sistema de coordenadas	COORDINATE_SYSTEM	Tag	Grupo de ejes coordinado.
Motion Control	MOTION_ INSTRUCTION	Tag	Estructura utilizada para acceder a los parámetros de estado de la instrucción.
Motion Type	SINT, INT o DINT	Inmediato	1 = Movimiento coordinado
Change Speed	SINT, INT o DINT	Inmediato	0 = No 1 = Sí
Speed	SINT, INT, DINT o REAL	Inmediato o tag	[Unidades de coordinación]
Speed Units	SINT, INT o DINT	Inmediato	0 = Unidades por seg 1 = % del Máximo
Change Accel	SINT, INT o DINT	Inmediato	0 = No 1 = Sí
Accel Rate	SINT, INT, DINT o REAL	Inmediato o tag	[Unidades de coordinación]
Accel Units	SINT, INT o DINT	Inmediato	0 = Unidades por seg^2 1 = % del Máximo
Change Decel	SINT, INT o DINT	Inmediato	0 = No 1 = Sí
Decel Rate	SINT, INT, DINT o REAL	Inmediato o tag	[Unidades de coordinación]
Decel Units	SINT, INT o DINT	Inmediato	0 = Unidades por seg^2 1 = % del Máximo
Change Accel Jerk	SINT, INT o DINT	Inmediato	0 = No 1 = Sí
Accel Jerk	SINT, INT, DINT o REAL	Inmediato o tag	**Siempre** debe ingresar un valor para el operando Accel Jerk. Esta instrucción sólo utiliza el valor si Profile está configurado como curva en S. • Accel Jerk es el régimen de jaloneo de aceleración para el sistema de coordenadas. Utilice estos valores para arrancar. Accel Jerk = 100 (% del tiempo) Unidades de jaloneo = 2
Change Decel Jerk	SINT, INT o DINT	Inmediato	0 = No 1 = Sí

Operandos de la instrucción MCCD – Lógica de escalera de relés

Operando	Tipo	Formato	Descripción
Decel Jerk	SINT, INT, DINT o REAL	Inmediato o tag	**Siempre** debe ingresar un valor para el operando Decel Jerk. Esta instrucción sólo utiliza el valor si Profile está configurado como curva en S. • Decel Jerk es el régimen de jaloneo de desaceleración para el sistema de coordenadas. Utilice estos valores para arrancar. • Decel Jerk = 100 (% del tiempo) • Unidades de jaloneo = 2
Jerk Units	SINT, INT o DINT	Inmediato	0 = Unidades por seg^3 1 = % del Máximo 2 = % de Tiempo (utilice este valor para comenzar)
Scope	SINT, INT o DINT	Inmediato	0 = Movimiento activo 1 = Movimiento activo y pendiente

Texto estructurado

MCCD(CoordinateSystem, MotionControl,MotionType ChangeSpeed,Speed,SpeedUnits,ChangeAccel, AccelRate,AccelUnits,ChangeDecel,DecelRate, DecelUnits,ChangeAccelJerk,AccelJerk, ChangeDecelJerk,DecelJerk,JerkUnits, Scope);

Los operandos son iguales a los de la instrucción MCCD de lógica de escalera de relés.

Cuando ingresa enumeraciones para el valor del operando en el texto estructurado, las enumeraciones de múltiples palabras se deben ingresar sin espacios. Por ejemplo: cuando ingresa Decel Units, el valor se debe ingresar como unitspersec^2 en lugar de Unidades por Seg^2 como se indica en la lógica de escalera.

Para los operandos con valores enumerados, ingrese su opción de la siguiente manera.

Este operando	Cuenta con estas opciones que usted ingresa como...	
	Texto	**O como**
Sistema de coordenadas	Sin enumeración	Tag
Motion Control	Sin enumeración	Tag
Move Type	Sin enumeración	Tag 0 = Absoluto 1 = Incremental

Este operando	**Cuenta con estas opciones que usted ingresa como…**	
	Texto	**O como**
ChangeSpeed	No Sí	0 1
Speed	Sin enumeración	Inmediato o tag
SpeedUnits	Unitspersec %ofmaximum	0 1
ChangeAccel	No Sí	0 1
Accel Rate	Sin enumeración	Inmediato o tag
Accel Units	Unitspersec^2 %ofmaximum	0 1
ChangeDecel	No Sí	0 1
Decel Rate	Sin enumeración	Inmediato o tag
Decel Units	Unitspersec^2 %ofmaximum	0 1
Change Accel Jerk	Sin enumeración	0 = No 1 = Sí
Accel Jerk	Sin enumeración	Inmediato o tag **Siempre** debe ingresar un valor para el operando Accel. Esta instrucción sólo utiliza el valor si Profile está configurado como curva en S. Utilice este valor para arrancar. Accel Jerk = 100 (% del tiempo)
Change Decel Jerk	Sin enumeración	0 = No 1 = Sí
Decel Jerk	Sin enumeración	Inmediato o tag **Siempre** debe ingresar un valor para el operando Decel Jerk. Esta instrucción sólo utiliza el valor si Profile está configurado como curva en S. Utilice este valor para arrancar. • Decel Jerk = 100 (% del tiempo) • Unidades de jaloneo = 2

Este operando	**Cuenta con estas opciones que usted ingresa como…**	
	Texto	**O como**
Jerk Units	Unitspersec^3 %ofmaximum %oftime	0 1 2 (utilice este valor para comenzar)
Scope	Sin enumeración	0 = Movimiento activo 1 = Movimiento activo y pendiente

Sistema de coordenadas

El operando del sistema de coordenadas indica el conjunto de los ejes de movimiento que definen las dimensiones de un sistema de coordenadas. El sistema de coordenadas acepta hasta tres (3) ejes primarios.

Control de movimiento

Los siguientes bits de control están afectados por la instrucción MCCD.

Mnemónico	**Descripción**
Bit .EN (Habilitado) 31	El bit Habilitado se establece cuando las transiciones del renglón cambian de falso a verdadero. Este bit se restablece cuando las transiciones del renglón cambian de verdadero a falso.
Bit .DN (Listo) 29	El bit Listo se restablece cuando las transiciones del renglón cambian de falso a verdadero. Se establece cuando la posición especifica se calcula correctamente.
Bit .ER (Error) 28	El bit Error se restablece cuando las transiciones del renglón cambian de falso a verdadero. Se establece cuando la posición especifica no logra realizar el cálculo correctamente.

Motion Type

El operando motion type determina qué perfil de movimiento cambiar. Actualmente, el movimiento coordinado es la única opción disponible.

- Movimiento coordinado – al seleccionarlo, la opción de movimiento coordinado cambia el movimiento del movimiento activo actual en el sistema de coordenadas.

Change Speed

El operando Change Speed determina si cambiar o no la velocidad del perfil de movimiento coordinado.

- No – no se producen cambios en la velocidad del movimiento coordinado.
- Si – la velocidad del movimiento coordinado cambia por el valor definido en los operandos Speed y Speed Units.

Speed

El operando Speed define la velocidad máxima a lo largo de la ruta del movimiento coordinado.

Speed Units

El operando Speed Units define las unidades aplicadas al operando Speed ya sea directamente en unidades de coordinación del sistema de coordenadas establecida o como un porcentaje de los valores máximos definidos en el sistema de coordenadas.

Change Accel

El operando Change Accel determina si cambiar o no la aceleración del perfil de movimiento coordinado.

- No – no se producen cambios en la aceleración del movimiento coordinado.
- Si – la aceleración del movimiento coordinado cambia por el valor definido en los operandos Accel Rate y Accel Units.

Accel Rate

El operando Accel Rate define la aceleración máxima a lo largo de la ruta del movimiento coordinado.

Accel Units

El operando Accel Units define las unidades aplicadas al operando Accel Rate ya sea directamente en unidades de coordinación del sistema de coordenadas establecido o como un porcentaje de los valores máximos definidos en el sistema de coordenadas.

Change Decel

El operando Change Decel determina si cambiar o no la desaceleración del perfil de movimiento coordinado.

- No – no se producen cambios en la desaceleración del movimiento coordinado.
- Si – la desaceleración del movimiento coordinado cambia por el valor definido en los operandos Decel Rate y Decel Units.

Decel Rate

El operando Decel Rate define la desaceleración máxima a lo largo de la ruta del movimiento coordinado.

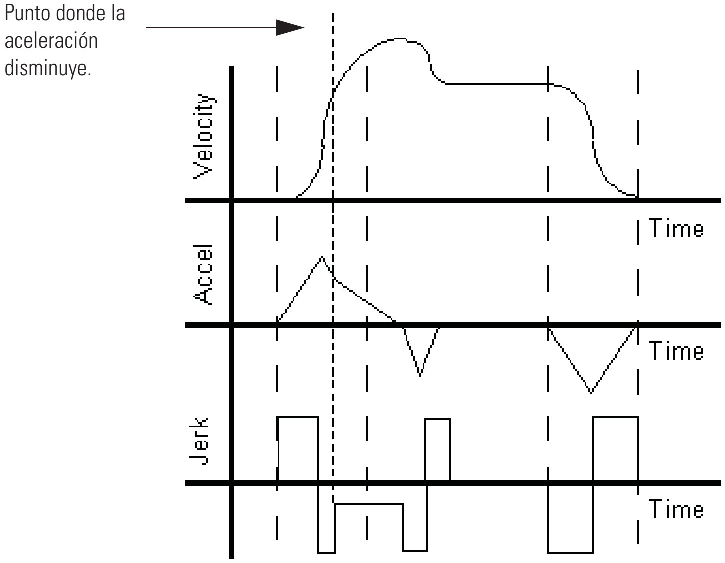
Decel Units

El operando Decel Units define las unidades aplicadas al operando Decel Rate ya sea directamente en unidades de coordinación del sistema de coordenadas establecido o como un porcentaje de los valores máximos definidos en el sistema de coordenadas.

Impacto de los cambios en los valores Aceleración y Desaceleración en el Perfil de movimiento

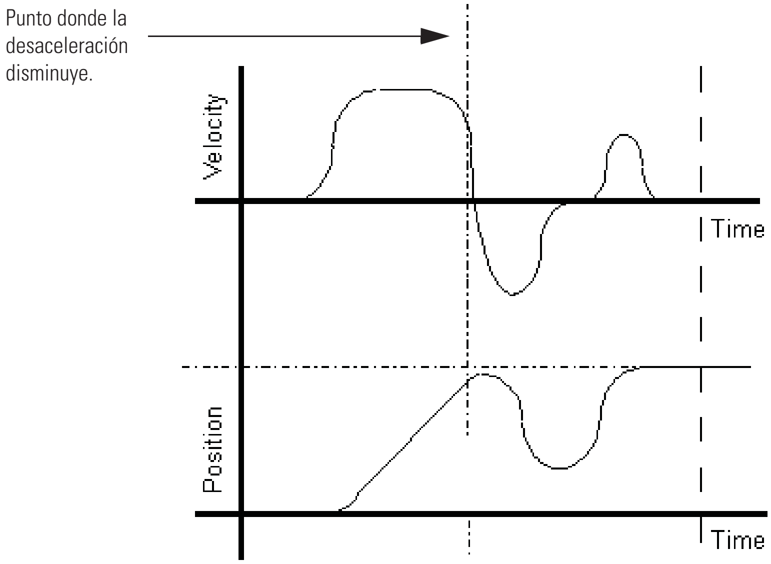
El siguiente gráfico ilustra lo que sucede cuando la instrucción MCCD se utiliza para reducir la aceleración cuando la velocidad se acerca al máximo. El nuevo régimen de jaloneo de aceleración se vuelve más pequeño, limitando aún más el cambio máximo en la aceleración. El sobreimpulso de velocidad ocurre debido al tiempo adicional requerido para que la aceleración alcance el cero. Otro perfil se genera para volver la velocidad al máximo programado.

Efecto del cambio en Aceleración



El siguiente gráfico ilustra lo que sucede cuando la instrucción MCCD se utiliza para reducir la desaceleración cuando la velocidad y la posición se acercan a sus puntos finales específicos. El nuevo régimen de jaloneo de desaceleración se vuelve más pequeño. El tiempo necesario para desacelerar a cero provoca que la velocidad se subimpulsa, pasando por cero y convirtiéndose en negativa. El movimiento del eje también retrocede hasta que la velocidad regresa a cero. Se genera un perfil adicional para regresar la posición a la programada.

Efecto del cambio en Desaceleración



Change Accel Jerk

El operando Change Accel Jerk determina si cambiar o no el jaloneo de aceleración del perfil de movimiento coordinado.

- No – no se producen cambios en el jaloneo de aceleración del movimiento coordinado.
- Si – la aceleración del movimiento coordinado cambia por el valor definido en los operandos Accel Jerk Rate y Jerk Units.

Accel Jerk

Accel Jerk define el jaloneo de aceleración máximo para el movimiento programado. Para más información sobre cálculos de Accel Jerk, consulte la sección de Unidades de jaloneo en este capítulo.

Change Decel Jerk

El operando Change Decel Jerk determina si cambiar o no el jaloneo de desaceleración del perfil de movimiento coordinado.

- No – no se producen cambios en el jaloneo de desaceleración del movimiento coordinado.
- Si – la desaceleración del movimiento coordinado cambia por el valor definido en los operandos Accel Jerk Rate y Jerk Units.

Decel Jerk

Decel Jerk define el jaloneo de desaceleración máximo para el movimiento programado. Para más información sobre cálculos de Decel Jerk, consulte la sección de Unidades de jaloneo en este capítulo.

Jerk Units

Jerk units define las unidades que se aplican a los valores ingresados en los operandos Accel Jerk y Decel Jerk. Los valores se ingresan directamente en las unidades de posición del sistema de coordenadas o como un porcentaje. Al configurar utilizando % de Máximo, el jaloneo se aplica como porcentaje de los operando Maximum Acceleration Jerk y Maximum Deceleration Jerk establecidos en los atributos del sistema de coordenadas. Al configurar utilizando % de Tiempo, el valor es un porcentaje basado en Speed, Accel Rate, y Decel Rate establecidos en la instrucción.

Si quiere convertir Unidades de ingeniería a % de Tiempo, utilice estas ecuaciones.

Para jaloneo de acel:

$$j_a\,[\mathrm{EU/s}^3] = \frac{a_{\max}{}^2[\mathrm{EU/s}^2]}{v_{\max}\,[\mathrm{EU/s}]}\left(\frac{200}{j_a\,[\%\ \mathrm{of\ time}]} - 1\right)$$

Para jaloneo de desacel:

$$j_a\,[\mathrm{EU/s}^3] = \frac{a_{\max}{}^2[\mathrm{EU/s}^2]}{v_{\max}\,[\mathrm{EU/s}]}\left(\frac{200}{j_a\,[\%\ \mathrm{of\ time}]} - 1\right)$$

Si quiere convertir % de Tiempo a Unidades de ingeniería, utilice estas ecuaciones.

Para jaloneo de acel:

$$j_a\,[\%\ \mathrm{of\ time}] = \frac{2}{1 + \frac{j_a\,[\mathrm{EU/s}^3]\ v_{\max}\,[\mathrm{EU/s}]}{a_{\max}^2\,[\mathrm{EU/s}^2]}}100$$

Para jaloneo de desacel:

$$j_d\,[\%\ \mathrm{of\ time}] = \frac{2}{1 + \frac{j_d\,[\mathrm{EU/s}^3]\ v_{\max}\,[\mathrm{EU/s}]}{d_{\max}^2\,[\mathrm{EU/s}^2]}}100$$

Scope

Al seleccionar Movimiento activo para el operando Scope indica que los cambios afectan sólo la dinámica del movimiento de la instrucción de control de movimiento coordinado activo. Al seleccionar Movimiento Activo y Pendiente se indica que los cambios afectan la dinámica de movimiento de la instrucción de control de movimiento coordinado activo y cualquier instrucción de control de movimiento coordinado pendiente en la cola. Actualmente, el tamaño de la cola está limitado a una instrucción luego de la instrucción activa.

MCCD es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético

no afectados

Condiciones de fallo

ninguno

Códigos de error:

Consulte Códigos de error (ERR) para las instrucciones de control de movimiento.

Códigos de error extendidos:

Los códigos de error extendido ayudan a definir aún más el mensaje de error arrojado por esta instrucción particular. Su comportamiento depende del Código de error al cual están asociados.

Los códigos de error extendido para Error de estado servo desactivado (5), Error del estado de desactivación (7), Tipo de eje no servo (8), Eje no configurado (11), Error de comando Home (16), y Tipo de dato de eje no válido (38) funcionan de la misma manera. Un número entre 0…*n* aparece en pantalla para el Código de error extendido. Este número es el índice del Sistema de Coordenadas indicando el eje que está en condición de error.

Para la instrucción MCCD, Código de error 13 – Parámetro fuera de rango, los Errores extendidos señalan un número que indica el parámetro en error como establecido en la plantilla en orden numérico de arriba hacia abajo comenzando de cero. Por ejemplo, 2 indica el valor del parámetro para Move Type en error.

Número y código de error referenciado	Indicador numérico del error extendido	Parámetro de instrucción	Descripción
Parámetro fuera de Rango (13)	2	Move Type	Move Type es menor que 0 o mayor que 1.
Parámetro fuera de Rango (13)	4	Speed	La velocidad es menor a 0.
Parámetro fuera de Rango (13)	7	Accel Rate	Accel Rate es menor o igual a 0.
Parámetro fuera de Rango (13)	10	Decel Rate	Decel Rate es menor o igual a 0.

Para el código de error 54 – El valor máximo de desaceleración es cero, si el error extendido devuelve un número positivo (0-*n*), está haciendo referencia al eje en error en el sistema de coordenadas. Diríjase a la ficha general Coordinate System Properties y mire debajo de la columna de corchetes ([ ]) de la cuadrícula del eje para determinar qué eje tiene un valor máximo de desaceleración de 0. Haga clic en el botón de elipsis al lado del eje en error para acceder a la pantalla de propiedades del eje. Diríjase a la ficha Dynamics y haga el cambio apropiado al valor máximo de desaceleración. Si el número del error extendido es –1, esto significa que el sistema de coordenadas tiene un valor máximo de desaceleración de 0. Diríjase a la ficha

Coordinate System Properties Dynamics para corregir el valor máximo de desaceleración.

MCCD Cambia a bits de estado:

Ningún efecto.

Operando Profile

Al utilizar esta instrucción, deberá tener en cuenta el Operando Profile. Para obtener más información, consulte la página 71.

Motion Coordinated Stop (MCS)

La instrucción MCS inicia un paro controlado de un movimiento coordinado. Cualquier perfil de movimiento pendiente se cancela.

ATENCIÓN ⚠

Los tags para instrucciones de control de movimiento se deben utilizar sólo una vez. Reutilizar el tag de control de movimiento en otras instrucciones puede provocar operaciones imprevistas. Esto puede dañar el equipo o causar lesiones personales.

ATENCIÓN ⚠

Riesgo de sobreimpulso de velocidad y/o de posición final

Si cambia los parámetros de movimiento dinámicamente mediante cualquier método, es decir cambiando la dinámica del movimiento (MCD o MCCD) o iniciando una nueva instrucción antes de que la última finalice, tenga en cuenta el riesgo de sobreimpulso de velocidad y/o de posición final.

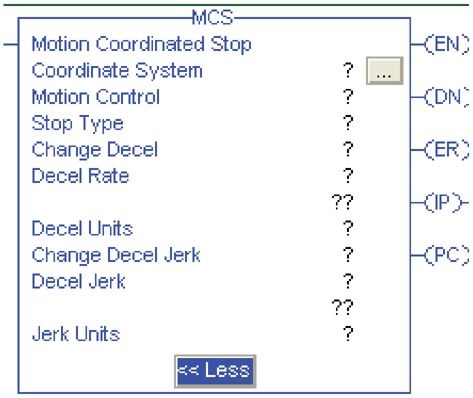
Se puede exceder un perfil de velocidad trapezoidal si la máxima desaceleración se disminuye mientras el movimiento reduce la velocidad o está cerca del punto de desaceleración.

Se puede exceder un perfil de velocidad con curva en S si:

- la máxima desaceleración disminuye mientras el movimiento reduce la velocidad o está cerca del punto de desaceleración; o
- la máxima aceleración de jaloneo disminuye y el eje acelera. Recuerde, sin embargo, que el jaloneo se puede cambiar indirectamente si se encuentra especificado en % de tiempo.

Operandos

Diagrama de lógica de escalera



MCS
Motion Coordinated Stop
Coordinate System ?
Motion Control ?
Stop Type ?
Change Decel ?
Decel Rate ?
??
Decel Units ?
Change Decel Jerk ?
Decel Jerk ?
??
Jerk Units ?

Operando	Tipo	Formato	Descripción	
Sistema de coordenadas	COORDINATE_SYSTEM	Tag	Nombre del sistema de coordenadas	
Motion Control	MOTION_ INSTRUCTION	Tag	Tag de control para la instrucción	
Stop Type	DINT	Inmediato	**Si desea**	**Seleccione este Stop Type**
			Detenga todos los movimientos para los ejes del sistema de coordenadas y detenga cualquier transformación de la cual forma parte el sistema de coordenadas	All (0) – Al calcular la dinámica inicial (es decir, régimen de aceleración y velocidad) para utilizar en Decel, se consideran todos los generadores de movimiento, incluido el movimiento coordinado para cada eje. Todos los ejes en el sistema de coordenadas se detienen independientemente utilizando la dinámica inicial calculada.
			Detenga sólo los movimientos coordinados	Movimiento coordinado (2)
			Interrumpe cualquier transformación de la cual es parte el sistema de coordenadas	Transformación coordinada (3)
Change Decel(1)	DINT	Inmediato	**Si desea**	**Seleccione**
			Utilizar el régimen de desaceleración máxima del sistema de coordenadas	No (0)
			Especificar la velocidad de desaceleración	Yes (1)

Operando	**Tipo**	**Formato**	**Descripción**
Decel Rate	REAL	Inmediato o tag	**Importante**: El eje podría sobreimpulsar su posición específica si reduce la desaceleración mientras un movimiento está en proceso. Desaceleración a lo largo de la ruta del movimiento coordinado. La instrucción utiliza esta valor: • Sólo si Change Decel está en Yes. • Sólo para movimientos coordinados. Ingrese un valor mayor que 0.
Decel Units	DINT	Inmediato	0 = Unidades por seg^2 1 = % del Máximo
Change Decel Jerk	SINT, INT o DINT	Inmediato	0 = No 1 = Sí
Decel Jerk	SINT, INT, DINT o REAL	Inmediato o tag	**Siempre** debe ingresar un valor para el operando Decel Jerk. Esta instrucción sólo utiliza el valor si Profile está configurado como curva en S. Decel Jerk es el régimen de jaloneo de desaceleración para el sistema de coordenadas. Utilice estos valores para arrancar. • Decel Jerk = 100 (% del tiempo) • Unidades de jaloneo = 2
Jerk Units	SINT, INT o DINT	Inmediato	0 = Unidades por seg^3 1 = % del Máximo 2 = % de Tiempo (utilice este valor para comenzar)

[1] El sobreimpulso puede suceder si MCS se ejecuta cerca o delante del punto de desaceleración y el límite de desaceleración disminuye. Recuerde que la desaceleración puede disminuir indirectamente cambiando ChangeDecel a NO si el régimen de desaceleración máximo es menor que el régimen de desaceleración activo.

Texto estructurado

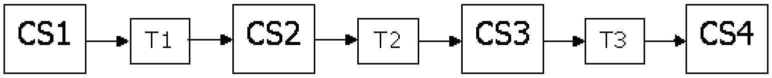
```
MCS(CoordinateSystem,
MotionControl,StopType, ChangeDecel,
DecelRate,DecelUnits,
ChangeDecelJerk,DecelJerk,
JerkUnits);
```

Los operandos del texto estructurado son iguales a los operandos de la lógica de escalera. Ingrese stop type y decel units sin espacios.

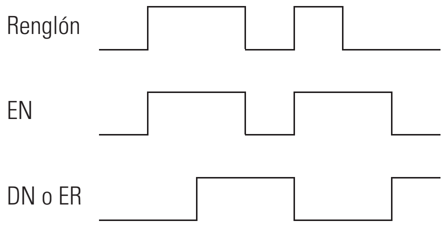
Ejemplo: Ingrese el operando Coordinate System como CoordinateSystem.

Cómo Stop Types afecta las transformaciones

La siguiente tabla describe cómo stop types afectan los sistemas de coordenadas que forman parte de una transformación.

Stop types	**Descripción**
All	Este tipo de paro: • detiene los ejes en el sistema de coordenadas especifico. También detiene los ejes de cualquier sistema de coordenadas que comparte ejes con este sistema de coordenadas. • interrumpe cualquier transformación de la cual es parte el sistema de coordenadas.
Movimiento coordinado	Este stop type detiene sólo los movimientos coordinados. Cualquier transformación permanece activa.
Transformación coordinada	Este stop type interrumpe las transformaciones relacionadas al sistema de coordenadas especifico. Todos los movimientos relacionados con la transformación se detienen en todos los sistemas de coordenadas específicos asociados. Sin embargo, los ejes coordinados de origen continuarán el movimiento como ordenado. **Ejemplo** Si cuatro sistemas de coordenadas están vinculados mediante tres transformaciones. Y el primer sistema de coordenadas (CS1) es el origen y está procesando el movimiento comandado.  Al ejecutar una instrucción MCS en CS2 y utilizar stop type de transformación coordinada, trae como resultado: • Transformaciones T1 y T2 se cancelan. • Transformación T3 permanece activa. • los ejes en CS1 permanecen en movimiento. • los ejes en los Sistemas de coordenadas CS2 y CS3 se detienen mediante el régimen de desaceleración seleccionado en la instrucción MCS o el régimen de desaceleración coordinada máxima. • los ejes en CS4 siguen a los ejes CS3 respectivamente. En una instrucción MAS, un stop type All también cancela las transformaciones.

Tipo de datos MOTION_INSTRUCTION

Para ver si	**Verifique si este bit está en**	**Tipo de datos**	**Notas**
El renglón es verdadero.	EN	BOOL	A veces, el bit EN permanece activo aún si el renglón pasa a falso. Esto sucede si el renglón cambia a falso antes de que la instrucción se ejecute o falle. 
El paro se ha iniciado correctamente.	DN	BOOL	
Ha ocurrido un error.	ER	BOOL	
El eje se detiene.	IP	BOOL	Cualquiera de estas acciones finalizan la instrucción MCS y desactivan el bit IP: • El sistema de coordenadas está detenido. • Otra instrucción MCS excede esta instrucción MCS. • Instrucción de interrupción. • Acción ante un fallo.
El eje está detenido.	PC	BOOL	El bit PC permanece activado hasta que el renglón hace una transición de falso a verdadero.

Indicadores de estado aritméticos:

no afectados

Condiciones de fallo:

ninguno

Códigos de error:

Consulte Códigos de error (ERR) para las instrucciones de control de movimiento.

Códigos de error extendidos:

Consulte Códigos de error (ERR) para las instrucciones de control de movimiento. Tiene información sobre cómo usar los códigos de error extendidos.

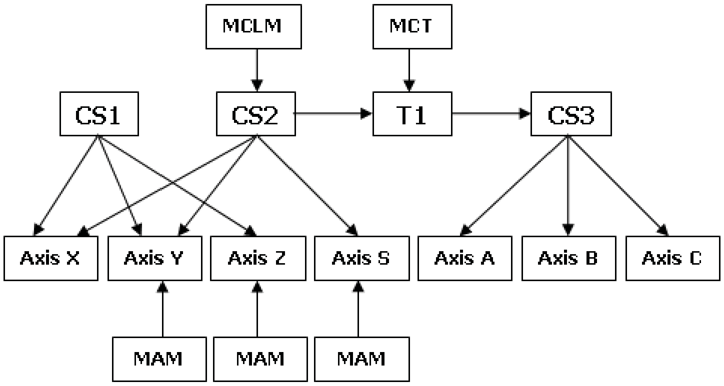
Cambia a Bit de estado

La instrucción cambia estos bits de estado cuando se ejecuta.

En el tag para	Este bit	Cuando Stop Type es	Cambia
Axis	CoordinatedMotionStatus	→	Desactivado cuando el movimiento coordinado se detiene
	TransformStateStatus	Movimiento coordinado	Sin cambio
		• All • Transformación coordinada	Desactivado
	ControlledByTransformStatus	Movimiento coordinado	Desactivado cuando los ejes se detienen y el bit PC de la instrucción MCS se activa
		• All • Transformación coordinada	Desactivado
Sistema de coordenadas	MotionStatus	→	Desactivado cuando el movimiento coordinado se detiene
	AccelStatus	→	Desactivado
	DecelStatus	→	Activado durante el paro y luego desactivado cuando la detención finaliza
	StoppingStatus	→	Activado durante el paro y luego desactivado cuando el bit PC se activa
	MoveStatus	→	Desactivado
	MoveTransitionStatus	→	Desactivado
	MovePendingStatus	→	Desactivado
	TransformSourceStatus	Movimiento coordinado	Sin cambio
		• All • Transformación coordinada	Desactivado
	TransformTargetStatus	Movimiento coordinado	Sin cambio
		• All • Transformación coordinada	Desactivado

Ejemplo de cómo Stop Types afecta las transformaciones y el movimiento del eje

Supongamos esta situación.



Donde:

- sistema de coordenadas 1 (CS1) contiene los ejes X, Y, y Z.
- sistema de coordenadas 2 (CS2) contiene los ejes Y, Z, y S.
- sistema de coordenadas 3 (CS3) contiene los ejes A, B, y C.
- la transformación (T1) vincula el origen coordinado CS2 al CS3 especifico.
- los ejes CS2 (XYS) se asignan a los ejes CS3 (ABC).
- Las instrucciones MAM se ejecutan en los ejes Y, Z, y S.
- La instrucción MCLM ejecutada en CS2.
- La instrucción MCT ejecutada con CS2 como el origen y CS3 como el especifico.
- No se ejecutan instrucciones coordinadas en CS2 o CS3.

Esta tabla muestra los resultados de la ejecución de varias instrucciones MCS y MAS con diferentes stop types.

Instrucciones MCS y MACS con Stop Types

Instrucción	Stop Type	Resultado
MCS en CS1	All	La instrucción MCLM en CS2 se detendrá.
		MAM en Y se detendrá.
		MAM en Z se detendrá.
		MAM en S continuará.
		T1 se cancela.
		Los ejes ABC se detendrán por la cancelación de la transformación.

Instrucciones MCS y MACS con Stop Types

Instrucción	Stop Type	Resultado
MCS en CS2	All	La instrucción MCLM en CS2 se detendrá.
		MAM en Y se detendrá.
		MAM en S se detendrá.
		MAM en Z continuará.
		T1 se cancela.
		Los ejes ABC se detendrán por la cancelación de la transformación.
MCS en CS3	All	La instrucción MCLM en CS2 continuará.
		MAM en Y continuará.
		MAM en S continuará.
		MAM en Z continuará.
		T1 se cancela.
		Los ejes ABC se detendrán por la cancelación de la transformación.
MCS en CS1	Movimiento coordinado	La instrucción MCLM en CS2 continuará.
		MAM en Y continuará.
		MAM en S continuará.
		MAM en Z continuará.
		T1 permanece activa.
		Los ejes ABC seguirán a los ejes CS2 respectivamente.
MCS en CS2	Movimiento coordinado	La instrucción MCLM en CS2 se detendrá.
		MAM en Y continuará.
		MAM en S continuará.
		MAM en Z continuará.
		T1 permanece activa.
		Los ejes ABC seguirán a los ejes CS2 respectivamente.
MCS en CS3	Movimiento coordinado	La instrucción MCLM en CS2 continuará.
		MAM en Y continuará.
		MAM en S continuará.
		MAM en Z continuará.
		T1 permanece activa.
		Los ejes ABC seguirán a los ejes CS2 respectivamente.
MAS en Y	All	La instrucción MCLM en CS2 se detendrá.
		MAM en Y se detendrá.
		MAM en S continuará.
		MAM en Z continuará.
		T1 se cancela.
		Los ejes ABC se detendrán por la cancelación de la transformación.

Instrucciones MCS y MACS con Stop Types

Instrucción	Stop Type	Resultado
MAS en Y	Mover	La instrucción MCLM en CS2 continuará.
		MAM en Y se detendrá.
		MAM en S continuará.
		MAM en Z continuará.
		T1 permanece activa.
		Los ejes ABC seguirán a los ejes CS2 respectivamente.
MAS en Z	All	La instrucción MCLM en CS2 continuará.
		MAM en Y continuará.
		MAM en S continuará.
		MAM en Z se detendrá.
		T1 permanece activa.
		Los ejes ABC seguirán a los ejes CS2 respectivamente.
MAS en Z	Mover	La instrucción MCLM en CS2 continuará.
		MAM en Y continuará.
		MAM en S continuará.
		MAM en Z se detendrá.
		T1 permanece activa.
		Los ejes ABC seguirán a los ejes CS2 respectivamente.
MCS en CS1	Transformación coordinada	La instrucción MCLM en CS2 continúa.
		MAM en Y continuará.
		MAM en S continuará.
		MAM en Z continuará.
		T1 permanece activa.
		Los ejes ABC seguirán a los ejes CS2 respectivamente.
MCS en CS2	Transformación coordinada	T1 se cancela.
		La instrucción MCLM en CS2 continúa.
		MAM en Y continuará.
		MAM en S continuará.
		MAM en Z continuará.
		Los ejes ABC se detendrán por la cancelación de la transformación.
MCS en CS3	Transformación coordinada	T1 se cancela.
		La instrucción MCLM en CS2 continúa.
		MAM en Y continuará.
		MAM en S continuará.
		MAM en Z continuará.
		Los ejes ABC se detendrán por la cancelación de la transformación.

Operando Profile

Al utilizar esta instrucción, deberá tener en cuenta el Operando Profile. Para obtener más información, consulte la página 71.

Motion Coordinated Shutdown (MCSD)

Utilice la instrucción MCSD para realizar una interrupción controlada de todos los ejes en el sistema de coordenadas llamado.

ATENCIÓN

Los tags para instrucciones de control de movimiento se deben utilizar sólo una vez. Reutilizar el tag de control de movimiento en otras instrucciones puede provocar operaciones imprevistas. Esto puede dañar el equipo o causar lesiones personales.

ATENCIÓN

Riesgo de sobreimpulso de velocidad y/o de posición final

Si cambia los parámetros de movimiento dinámicamente mediante cualquier método, es decir cambiando la dinámica del movimiento (MCD o MCCD) o iniciando una nueva instrucción antes de que la última finalice, tenga en cuenta el riesgo de sobreimpulso de velocidad y/o de posición final.

Se puede exceder un perfil de velocidad trapezoidal si la máxima desaceleración se disminuye mientras el movimiento reduce la velocidad o está cerca del punto de desaceleración.

Se puede exceder un perfil de velocidad con curva en S si:

- la máxima desaceleración disminuye mientras el movimiento reduce la velocidad o está cerca del punto de desaceleración; o
- la máxima aceleración de jaloneo disminuye y el eje acelera. Recuerde, sin embargo, que el jaloneo se puede cambiar indirectamente si se encuentra especificado en % de tiempo.

Operandos de lógica de escalera de relés

Operando	Tipo	Formato	Descripción
Sistema de coordenadas	COORDINATE_SYSTEM	Tag	Grupo de ejes coordinado.
Motion Control	MOTION_ INSTRUCTION	Tag	Estructura utilizada para acceder a los parámetros de estado de la instrucción.

Texto estructurado

```
MCSD(CoordinateSystem,
MotionControl);
```

Los operandos son iguales a los de la instrucción MCSD de lógica de escalera de relés.

Sistema de coordenadas

El operando del sistema de coordenadas indica el conjunto de los ejes de movimiento que definen las dimensiones de un sistema de coordenadas cartesianas. Para esta versión, el sistema de coordenadas acepta hasta tres (3) ejes primarios. Sólo aquellos ejes configurados como ejes primarios (hasta 3) se incluyen en los cálculos de velocidad coordinada.

Control de movimiento

Los siguientes bits de control están afectados por la instrucción MCSD.

Mnemónico	Descripción
Bit .EN (Habilitado) 31	El bit Habilitado se establece cuando las transiciones del renglón cambian de falso a verdadero. Este bit se restablece cuando el renglón cambia de verdadero a falso.
Bit .DN (Listo) 29	El bit Listo se establece cuando la interrupción coordinado se inicia correctamente. Se reestablece cuando el renglón va de falso a verdadero.
Bit .ER (Error) 28	El bit Error se establece cuando la interrupción coordinada no puede iniciar correctamente. Se restablece cuando el renglón va de falso a verdadero.

MCSD es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritmético

No afectados

Condiciones de fallo

Ninguno

Códigos de error

Consulte Códigos de error (ERR) para las instrucciones de control de movimiento.

MCSD Cambia a Bit de estado

Los Bit de estado proporcionan un medio para monitorear el progreso de la instrucción de movimiento. Hay tres tipos de bits de estado que brindan la información pertinente. Estos son: Bit de estado del eje, bits de estado del sistema de coordenadas y bits de estado del movimiento coordinado. Cuando la instrucción MCS comienza, los bits de estado sufren los siguientes cambios.

Bits de estado del eje

Nombre del bit	Efecto
CoordinatedMoveStatus	Borrado

Bits de estado del sistema de coordenadas

Nombre del bit	Efecto
ShutdownStatus	Se establece cuando se ejecuta MCSD y todos los ejes relacionados se interrumpen.
ReadyStatus	Borrado luego de que MCSD se ejecuta.

Bits de estado del movimiento coordinado

Nombre del bit	Efecto
AccelStatus	Borrado luego de que MCSD se ejecuta.
DecelStatus	Borrado luego de que MCSD se ejecuta.
ActualPosToleranceStatus	Borrado luego de que MCSD se ejecuta.
CommandPosToleranceStatus	Borrado luego de que MCSD se ejecuta.
StoppingStatus	Borrado luego de que MCSD se ejecuta.
MoveStatus	Borrado luego de que MCSD se ejecuta.
MoveTransitionStatus	Borrado luego de que MCSD se ejecuta.
MovePendingStatus	Borrado luego de que MCSD se ejecuta.
MovePendingQueueFullStatus	Borrado luego de que MCSD se ejecuta.

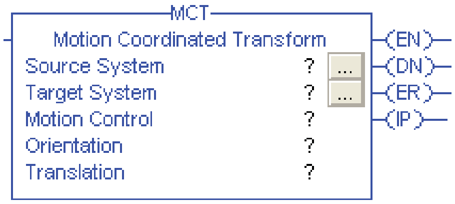
Motion Coordinated Transform (MCT)

Utilice la instrucción MCT para iniciar una trasformación que vincule dos sistemas de coordenadas juntos. Esto es como un engranaje bidireccional. Una manera de utilizar la transformación es mover un robot no cartesiano a posiciones cartesianas.

ATENCIÓN Los tags para instrucciones de control de movimiento se deben utilizar sólo una vez. Reutilizar el tag de control de movimiento en otras instrucciones puede provocar operaciones imprevistas. Esto puede dañar el equipo o causar lesiones personales.

IMPORTANTE Sólo puede usar esta instrucción con los controladores 1756-L6*x*.

Operandos: Diagrama de lógica de escalera



MCT
Motion Coordinated Transform
Source System ?
Target System ?
Motion Control ?
Orientation ?
Translation ?

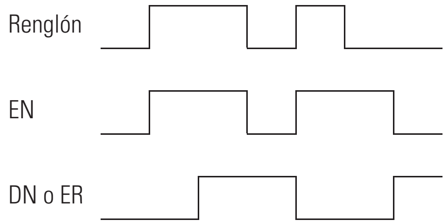
<table>
<tr><th>Operando</th><th>Tipo</th><th>Formato</th><th colspan="2">Descripción</th></tr>
<tr><td>Sistema origen</td><td>COORDINATE_SYSTEM</td><td>Tag</td><td colspan="2">El sistema de coordenadas que utiliza para programar los movimientos. Comúnmente, este es el sistema de coordenadas cartesianas.</td></tr>
<tr><td>Sistema específico</td><td>COORDINATE_SYSTEM</td><td>Tag</td><td colspan="2">Sistema de coordenadas no cartesianas que controla en equipo real.</td></tr>
<tr><td>Motion Control</td><td>MOTION_ INSTRUCTION</td><td>Tag</td><td colspan="2">Tag de control para la instrucción.</td></tr>
<tr><td rowspan="5">Orientación</td><td rowspan="5">REAL[3]</td><td rowspan="5">Matriz</td><td colspan="2">¿Desea girar la posición específica alrededor del eje X1, X2, o X3?</td></tr>
<tr><td>Si</td><td>Entonces</td></tr>
<tr><td>No</td><td>Deje los valores de matriz en cero.</td></tr>
<tr><td>Sí</td><td>Ingrese los grados de rotación en la matriz. Coloque los grados de rotación alrededor de X1 en el primer elemento de la matriz, y así consecutivamente.</td></tr>
<tr><td colspan="2">Utilice la matriz de los tres REALs aún si el sistema de coordenadas tiene sólo uno o dos ejes.</td></tr>
<tr><td rowspan="5">Traslación</td><td rowspan="5">REAL[3]</td><td rowspan="5">Matriz</td><td colspan="2">¿Desea compensar la posición específica a lo largo del eje X1, X2, o X3?</td></tr>
<tr><td>Si</td><td>Entonces</td></tr>
<tr><td>No</td><td>Deje los valores de matriz en cero.</td></tr>
<tr><td>Sí</td><td>Ingrese las distancias offset en la matriz. Ingrese las distancias offset en la unidades de coordinación. Coloque la distancia offset para X1 en el primer elemento de la matriz, y así consecutivamente.</td></tr>
<tr><td colspan="2">Utilice la matriz de los tres REALs aún si el sistema de coordenadas tiene sólo uno o dos ejes.</td></tr>
</table>

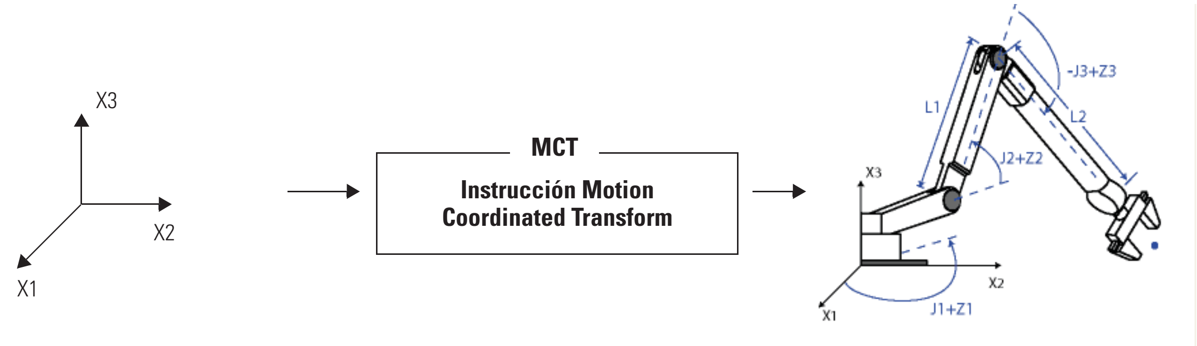
Texto estructurado

MCT(Source System,Target System,
Motion Control,Orientation,Translation);

Los operandos del texto estructurado son iguales a los operandos de la lógica de escalera.

Tipo de datos MOTION_INSTRUCTION

Para ver si	**Verifique si este bit está en**	**Tipo de datos**	**Notas**
El renglón es verdadero.	EN	BOOL	A veces, el bit EN permanece activo aún si el renglón pasa a falso. Esto sucede si el renglón cambia a falso antes de que la instrucción se ejecute o falle. 
La instrucción ha sido ejecutada.	DN	BOOL	El proceso de transformación sigue en marcha luego de la instrucción ha sido ejecutada.
Ha ocurrido un error.	ER	BOOL	Identifique el número del error en el campo de código de error del tag Motion Control, luego, consulte Códigos de error (ERR) para las instrucciones de control de movimiento en página 395 de este manual.
El proceso de transformación está en marcha.	IP	BOOL	Cualquiera de estas acciones cancela la transformación y desactivan el bit IP: • Instrucción de paro aplicable • Instrucción de interrupción • Acción ante un fallo



Mueve un sistema de ejes virtuales a las posiciones cartesianas (X1, X2, X3).

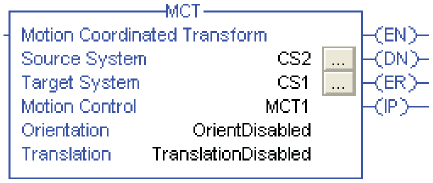
La transformación convierte el movimiento a los ángulos de articulación y mueve el robot.

La transformación controla hasta tres articulaciones del robot: J1, J2, y J3.

Flujo de datos de la instrucción MCT entre dos sistemas de coordenadas

Las siguientes ilustraciones muestran el flujo de datos cuando una instrucción MCT está activa. CS1 es un sistema de coordenadas cartesianas que contiene ejes X1, X2 y X3 como el origen de la instrucción MCT. CS2 es el sistema de coordenadas de articulación que contiene ejes J1, J2 y J3 como el específico de la instrucción MCT

Flujo de datos cuando se ejecuta un movimiento con una instrucción MCT – Transformación de avance


MCT
Motion Coordinated Transform
Source System CS2
Target System CS1
Motion Control MCT1
Orientation OrientDisabled
Translation TranslationDisabled

Datos de entrada

CS2: Datos	ORIGEN
Posiciones de la articulación (J1, J2, J3)	Máquina real Sistema de coordenadas
Longitudes de vínculo (L1, L2)	Diálogo del sistema de coordenadas
Offsets base (X1b, X2b, X3b)	Diálogo del sistema de coordenadas
Offsets de efectores finales (X1e, X2e, X3e)	Diálogo del sistema de coordenadas
Orientaciones del ángulo cero (Z1, Z2, Z3)	Diálogo del sistema de coordenadas
Orientaciones (Matriz [3])	Plantilla de la instrucción
Translación (Matriz [3])	Plantilla de la instrucción

Instrucción activa

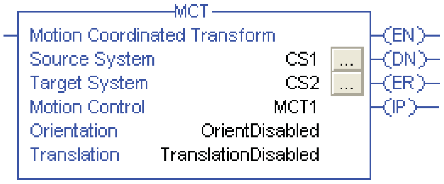
➡

MCT

Salida calculada

CS1: Datos	
Posiciones cartesianas (X1, X2, X3)	Destino Máquina virtual Sistema de coordenadas

Flujo de datos cuando se ejecuta un movimiento con una instrucción MCT – Transformación inversa


MCT
Motion Coordinated Transform
Source System CS1
Target System CS2
Motion Control MCT1
Orientation OrientDisabled
Translation TranslationDisabled

Datos de entrada

CS1: Datos	ORIGEN
Posiciones cartesianas (X1, X2, X3)	Máquina virtual Sistema de coordenadas
Longitudes de vínculo (L1, L2)	Diálogo del sistema de coordenadas
Offsets base (X1b, X2b, X3b)	Diálogo del sistema de coordenadas
Offsets de efectores finales (X1e, X2e, X3e)	Diálogo del sistema de coordenadas
Orientaciones del ángulo cero (Z1, Z2, Z3)	Diálogo del sistema de coordenadas
Orientaciones (Matriz [3])	Plantilla de la instrucción
Translación (Matriz [3])	Plantilla de la instrucción

Instrucción activa

➡

MCT

Salida calculada

CS2: Datos	Destino
Posiciones de la articulación (J1, J2, J3)	Máquina real Sistema de coordenadas

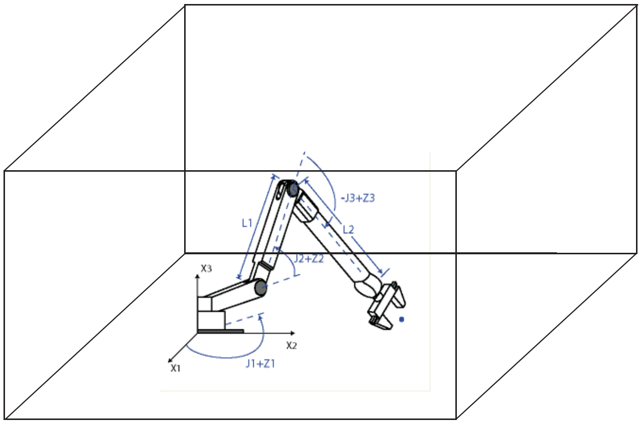
Pautas de programación

Siga estas pautas para utilizar una instrucción MCT.

ATENCIÓN

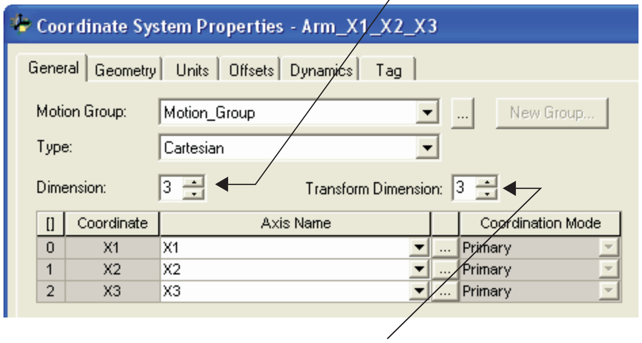
No permita que el robot se extienda o retraiga completamente. De lo contrario, puede iniciar un movimiento a muy alta velocidad. En estas posiciones, el mismo pierde su configuración como brazo derecho e izquierdo. Cuando esto sucede, puede iniciar un movimiento a muy alta velocidad.

Determine los límites de trabajo del robot y mantenga el mismo dentro de estos límites.



Pautas de la instrucción MCT

<table>
<tr><th>Pauta</th><th>Ejemplos y Notas</th></tr>
<tr><td>Establezca un sistema de coordenadas de los ejes para las posiciones cartesianas del robot. Comúnmente, estos ejes son virtuales.</td><td>
Número de los ejes de un sistema de coordenadas.



Motion Group: Motion_Group
Type: Cartesian
Dimension: 3
Transform Dimension: 3

<table>
<tr><th>[]</th><th>Coordinate</th><th>Axis Name</th><th>Coordination Mode</th></tr>
<tr><td>0</td><td>X1</td><td>X1</td><td>Primary</td></tr>
<tr><td>1</td><td>X2</td><td>X2</td><td>Primary</td></tr>
<tr><td>2</td><td>X3</td><td>X3</td><td>Primary</td></tr>
</table>

Número de ejes a transformar.

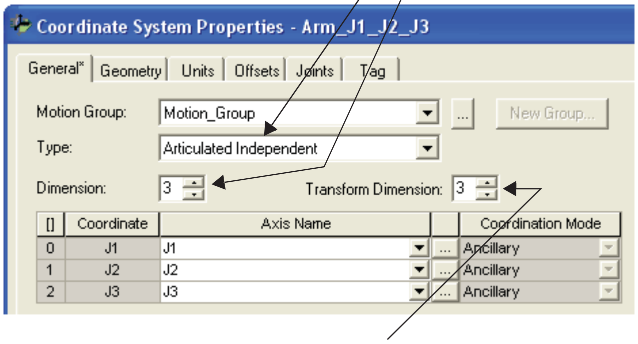
Importante: Puede ver un error de truncamiento en la precisión de los cálculos. Esto sucede cuando estas dos condiciones se cumplen:

- Los constantes de conversión de los ejes cartesianos virtuales en una transformación son pequeños, como 8000 cantidades/posición unidad.
- Las longitudes de vínculo del sistema de coordenadas no cartesianas son pequeñas, como 0,5 pulg.

Es mejor proporcionar grandes constantes de conversión a los ejes cartesianos virtuales en una transformación, como 100.000 o 1.000.000 cantidades/posición unidad. El fin de carrera máximo del robot es

$$\frac{\pm 2^{31}}{\text{Constante de conversión}} \quad \text{Unidades de coordinación}$$
</td></tr>
<tr><td>Establezca otro sistema de coordenadas para las articulaciones reales del robot.</td><td>
Tipo de geometría de robot.

Número de los ejes de un sistema de coordenadas.

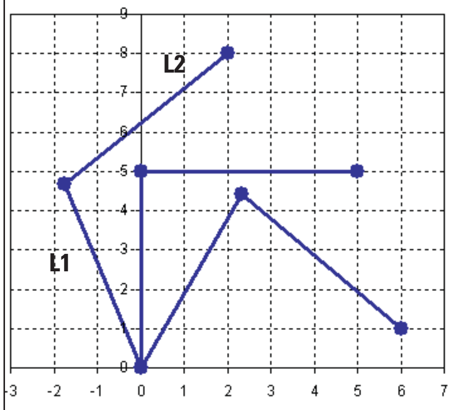
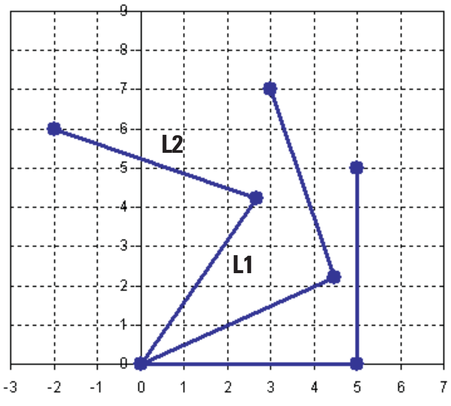
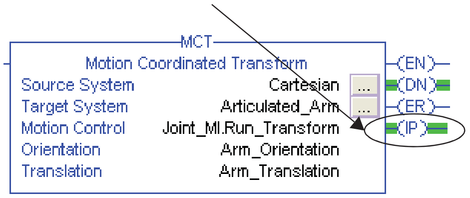


Motion Group: Motion_Group
Type: Articulated Independent
Dimension: 3
Transform Dimension: 3

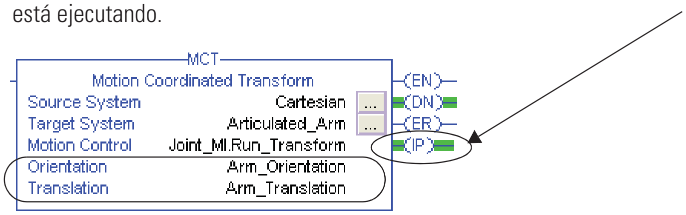
<table>
<tr><th>[]</th><th>Coordinate</th><th>Axis Name</th><th>Coordination Mode</th></tr>
<tr><td>0</td><td>J1</td><td>J1</td><td>Ancillary</td></tr>
<tr><td>1</td><td>J2</td><td>J2</td><td>Ancillary</td></tr>
<tr><td>2</td><td>J3</td><td>J3</td><td>Ancillary</td></tr>
</table>

Número de ejes a transformar.
</td></tr>
</table>

Pautas de la instrucción MCT

Pauta	Ejemplos y Notas
Mueva el robot hacia una posición de inicio de brazo derecho o izquierdo.	¿Desea que el robot se mueva como un brazo izquierdo o derecho?  **Brazos izquierdos**  **Brazos derechos** Antes de iniciar la transformación, mueva el robot a una posición de descanso que proporciona el brazo que desea (izquierdo o derecho). Una vez que se inicia la transformación y el movimiento cartesiano en el sistema de coordenadas de origen, el robot permanece como brazo izquierdo o brazo derecho. Si se inicia como brazo izquierdo, se mueve como brazo izquierdo. Si se inicia como brazo derecho, se mueve como brazo derecho. Siempre puede cambiar de brazo izquierdo a derecho o viceversa. Para hacer esto, mueva las articulaciones directamente.
Alterne el renglón de falso a verdadero para ejecutar la instrucción.	Esta es una instrucción transicional: En el diagrama de lógica de escalera, alterne la condición de entrada del renglón de falso a verdadero cada vez que desee ejecutar la instrucción. Cuando ejecute la instrucción, la transformación arranca y el bit IP se activa.  MCT Motion Coordinated Transform Source System Cartesian Target System Articulated_Arm Motion Control Joint_MI.Run_Transform Orientation Arm_Orientation Translation Arm_Translation Puede dejar que el renglón pase a falso una vez que ejecute la instrucción. Transformación permanece activa.
En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición.	En texto estructurado, las instrucciones se ejecutan cada vez que son escaneadas. Acondicione la instrucción de manera que sólo se ejecute en una transición. Utilice alguno de estos dos métodos: • Calificador de una acción SFC • Construcción de texto estructurado
Arranque la transformación antes de iniciar cualquier movimiento.	No puede iniciar una transformación si ningún proceso de movimiento está controlando un eje del sistema de coordenadas de origen o específico. **Ejemplo:** Arranque la transformación antes de iniciar el engranaje o una operación de levas.

Pautas de la instrucción MCT

Pauta	Ejemplos y Notas
Excepto el movimiento bidireccional entre los sistemas de coordenadas de origen y específico.	Una transformación es bidireccional.  Cuando inicie la transformación, la posición del sistema de coordenadas de origen cambia para coincidir con la posición correspondiente del sistema de coordenadas específico. Luego de esto, si mueve cualquiera de los dos sistemas, el otro sistema se mueve en respuesta. El controlador continúa controlando los ejes aún si detiene el escán de la instrucción MCT o su renglón pasa a falso. Utilice la instrucción MCS para detener el movimiento en el sistema de coordenadas, cancelar la transformación, o ambas.
Utilice la instrucción MCS para cancelar la transformación.	
Ejecute nuevamente la instrucción MCT si cambia la orientación o la traslación.	Si desea cambiar los valores de orientación o traslación luego de que la transformación se está ejecutando.  MCT Motion Coordinated Transform Source System Cartesian Target System Articulated_Arm Motion Control Joint_Ml.Run_Transform Orientation Arm_Orientation Translation Arm_Translation Entonces, ejecute nuevamente la instrucción. Para ejecutar la instrucción, alterne la condición de entrada del renglón de falso a verdadero. También ejecute nuevamente la instrucción si cambia la geometría del equipo.

Indicadores de estado aritmético

No afectados

Condiciones de fallo

Ninguno

Códigos de error

Consulte Códigos de error (ERR) para las instrucciones de control de movimiento.

Códigos de error extendidos:

Los códigos de error extendido ayudan a definir, además, el mensaje de error dado por esta instrucción particular. Este comportamiento depende del código de error al cual son asociados.

Códigos de error extendidos

ERR	EXERR	Acción correctiva	Notas
61	1	Asigne ambos sistemas de coordenadas al grupo de movimiento.	
	2	Controle que está utilizando los sistemas de origen y específico correctos.	No puede utilizar el mismo sistema de coordenadas como origen y específico.
	3	Establezca la dimensión de transformación del sistema de origen al número de ejes en el sistema, hasta tres.	
	4	Establezca la dimensión de transformación del sistema específico al número de ejes a transformar, hasta tres.	
	5	Utilice un sistema de origen diferente.	Puede usar solamente un sistema de coordenadas como el de origen para una transformación activa.
	6	Utilice un sistema específico diferente.	Puede usar solamente un sistema de coordenadas como el específico para una transformación activa.
	7	Busque los ejes de origen o específicos que ya haya utilizado en otra transformación. Utilice diferentes ejes en el sistema de coordenadas.	Puede utilizar un eje en un sistema de origen y en un sistema específico.
	8	Utilice un sistema específico que no es el origen para esta cadena de transformaciones.	No puede crear una cadena circular de transformaciones que devuelva al origen.
	9	Controle que haya asignado los ejes correctos a cada sistema de coordenadas.	No puede utilizar los mismos ejes en los sistemas de origen y específico.
	10	Detenga todos los procesos de movimiento para todos los ejes en ambos sistemas (por ejemplo, impulso, movimiento y engranaje).	No puede iniciar la transformación si ningún proceso de movimiento está controlando un eje de origen o específico.
	11	Recursos insuficientes disponibles para iniciar la conexión de transformación.	
	12	Establezca las longitudes de vínculo.	No puede utilizar una longitud de vínculo de cero.
	13	Busque los ejes de origen o específico que están en estado de interrupción. Utilice la instrucción MASR o comando directo para reestablecer el eje.	
	14	No inhibir todos los ejes de origen o específico.	
	15	Controle que los valores de configuración para los offset base y los offset de los efectores finales para el robot Delta o SCARA Delta.	(X1b-X1e) no puede ser menor que 0.0 para ambos robots Delta y SCARA Delta. Para los robots Delta, este error puede ocurrir si el valor de L1 + (X1b-X1e) es mayor que L2.
	16	Controle que las configuraciones del robot SCARA independiente y SCARA Delta sean las siguientes: • la dimensión de transformación para el sistema de coordenadas de origen se configura como 2. • los terceros ejes configurados para el sistema de coordenadas de origen y el sistema de coordenadas específico son los mismos.	

Cambia a Bit de estado

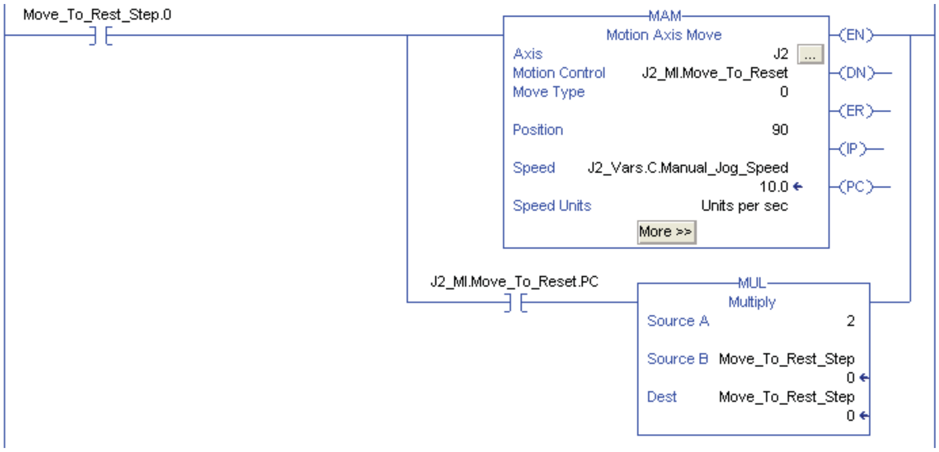
Para ver si	Controle el tag para	Y este bit	Para obtener
Un sistema de coordenadas es el origen de una transformación activa.	Sistema de coordenadas	TransformSourceStatus	activado
Un sistema de coordenadas es el objetivo de una transformación activa.	Sistema de coordenadas	TransformTargetStatus	Activado
Un eje es parte de una transformación activa.	AXIS	TransformStateStatus	Activado
Un eje se mueve por una transformación.	Axis	ControlledByTransformStatus	Activado

Ejemplo 1 – Diagrama de lógica de escalera de recogida y colocación

1. Mover a rutina de descanso

Esta rutina es una secuencia de movimientos que ponen a un robot independiente articulado en una posición de descanso en los ángulos de brazo derecho o izquierdo.

Cuando Move_To_Rest_Step.0 se activa, el eje J2 se mueve a 90·. Luego, la secuencia pasa al próximo paso.


Move_To_Rest_Step.0

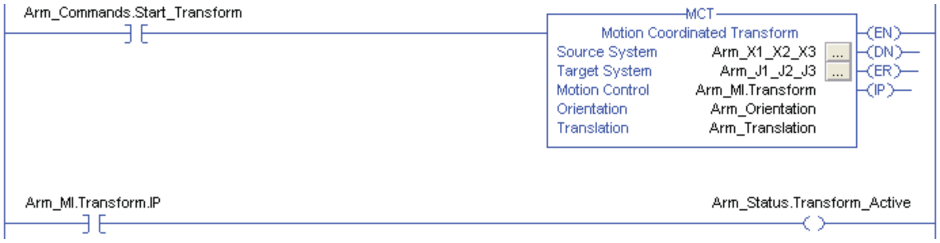
MAM Motion Axis Move
Axis J2
Motion Control J2_MI.Move_To_Reset
Move Type 0
Position 90
Speed J2_Vars.C.Manual_Jog_Speed 10.0
Speed Units Units per sec

J2_MI.Move_To_Reset.PC

MUL Multiply
Source A 2
Source B Move_To_Rest_Step 0
Dest Move_To_Rest_Step 0

2. Iniciar rutina de transformación

Cuando Arm_Commands.Start_Transform se activa, la transformación arranca. El bit IP señala que la transformación se está ejecutando.


Arm_Commands.Start_Transform

MCT Motion Coordinated Transform
Source System Arm_X1_X2_X3
Target System Arm_J1_J2_J3
Motion Control Arm_MI.Transform
Orientation Arm_Orientation
Translation Arm_Translation

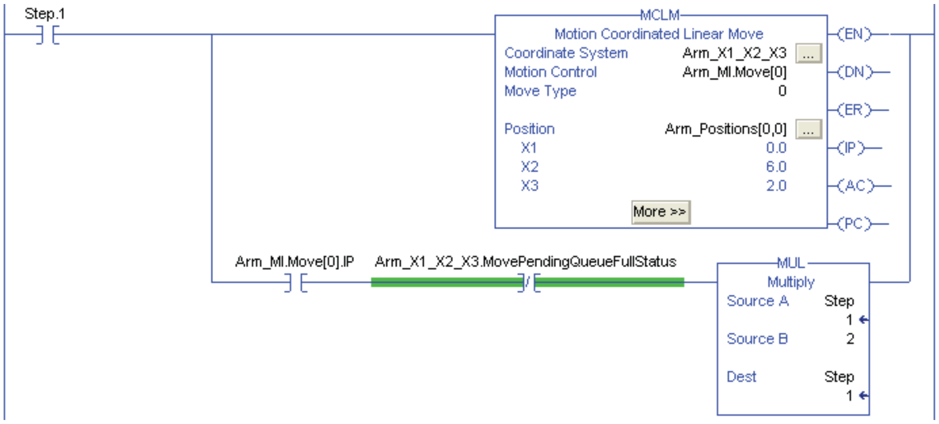
Arm_MI.Transform.IP

Arm_Status.Transform_Active

3. Rutina de recogida y colocación

Esta rutina es una en una secuencia de instrucciones MCLM que mueven el sistema cartesiano. Las articulaciones del robot siguen los movimientos.

Cuando Step.1 se activa, el sistema de coordenadas mueve a 0, 6, 2. Cuando el movimiento está en proceso (IP), la secuencia pone en cola el próximo movimiento.


Step.1

MCLM Motion Coordinated Linear Move
Coordinate System Arm_X1_X2_X3
Motion Control Arm_MI.Move[0]
Move Type 0
Position Arm_Positions[0,0]
X1 0.0
X2 6.0
X3 2.0

Arm_MI.Move[0].IP

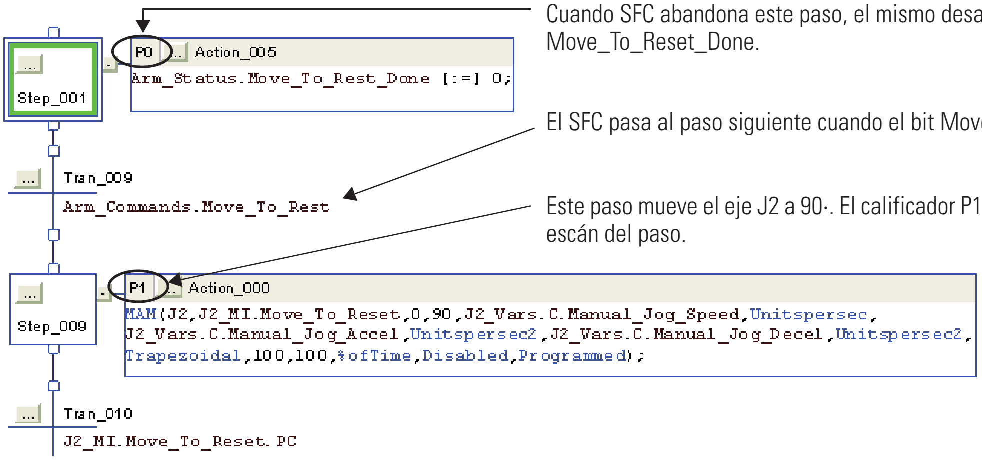
Arm_X1_X2_X3.MovePendingQueueFullStatus

MUL Multiply
Source A Step 1
Source B 2
Dest Step 1

Recogida y colocación – Ejemplo de texto estructurado

1. Mover a rutina de descanso

Esta rutina es una secuencia de movimientos que ponen a un robot en una posición de descanso en los ángulos de brazo derecho o izquierdo.



Cuando SFC abandona este paso, el mismo desactiva el bit Move_To_Reset_Done.

```
Arm_Status.Move_To_Rest_Done [:=] 0;
```

El SFC pasa al paso siguiente cuando el bit Move_To_Rest se activa.

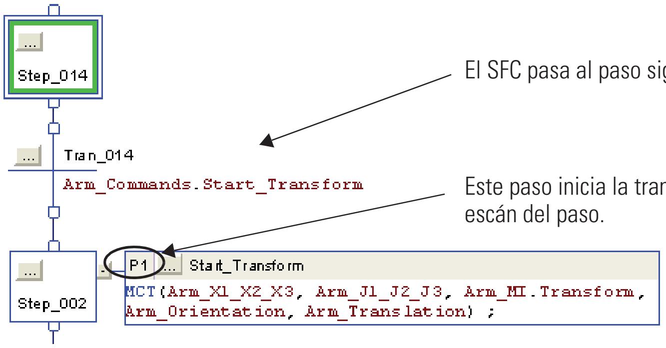
```
Arm_Commands.Move_To_Rest
```

Este paso mueve el eje J2 a 90·. El calificador P1 limita esto al primer escán del paso.

```
MAM(J2,J2_MI.Move_To_Reset,0,90,J2_Vars.C.Manual_Jog_Speed,Unitspersec,
J2_Vars.C.Manual_Jog_Accel,Unitspersec2,J2_Vars.C.Manual_Jog_Decel,Unitspersec2,
Trapezoidal,100,100,%ofTime,Disabled,Programmed);
```

```
J2_MI.Move_To_Reset.PC
```

2. Iniciar rutina de transformación



El SFC pasa al paso siguiente cuando el bit Move_To_Rest se activa.

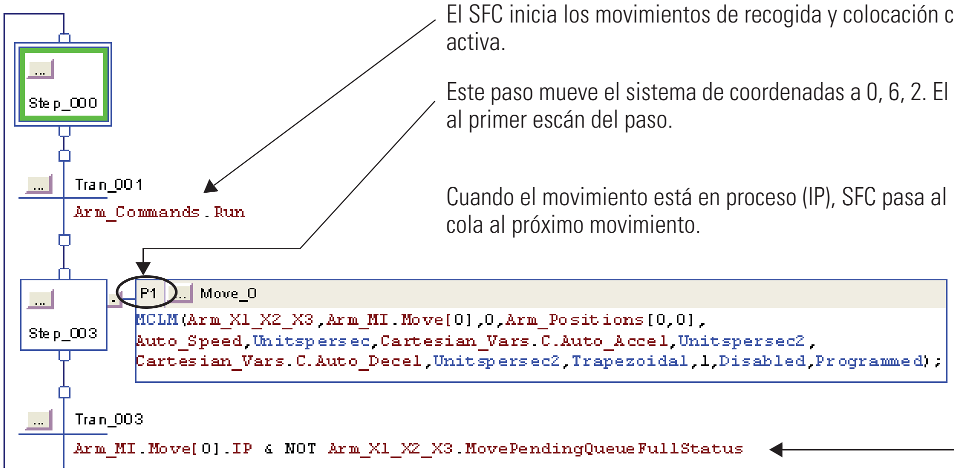
```
Arm_Commands.Start_Transform
```

Este paso inicia la transformación. El calificador P1 limita esto al primer escán del paso.

```
MCT(Arm_X1_X2_X3, Arm_J1_J2_J3, Arm_MI.Transform,
Arm_Orientation, Arm_Translation) ;
```

3. Rutina de recogida y colocación

Esta rutina es una en una secuencia de instrucciones MCLM que mueven el sistema cartesiano. Las articulaciones del robot siguen los movimientos.



El SFC inicia los movimientos de recogida y colocación cuando el bit Run se activa.

```
Arm_Commands.Run
```

Este paso mueve el sistema de coordenadas a 0, 6, 2. El calificador P1 limita este al primer escán del paso.

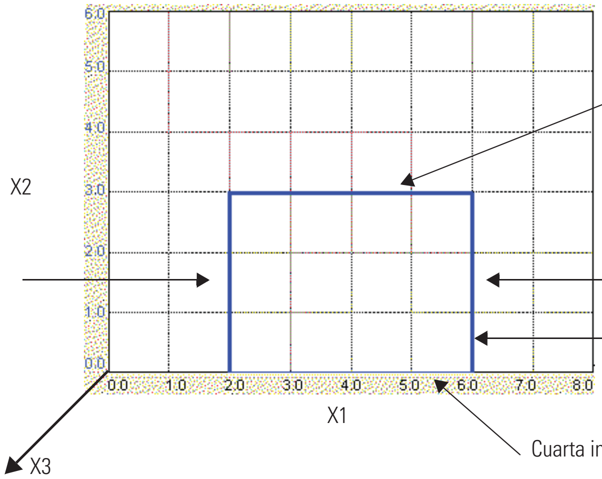
```
MCLM(Arm_X1_X2_X3,Arm_MI.Move[0],0,Arm_Positions[0,0],
Auto_Speed,Unitspersec,Cartesian_Vars.C.Auto_Accel,Unitspersec2,
Cartesian_Vars.C.Auto_Decel,Unitspersec2,Trapezoidal,1,Disabled,Programmed);
```

Cuando el movimiento está en proceso (IP), SFC pasa al próximo paso y pone en cola al próximo movimiento.

```
Arm_MI.Move[0].IP & NOT Arm_X1_X2_X3.MovePendingQueueFullStatus
```

Ejemplo de orientación de cambio

Si desea mover el sistema de coordenadas de origen en una ruta rectangular. En ese caso, ejecutar la instrucción MCT para iniciar la transformación. Luego, ejecute cuatro instrucciones MCLM para producir la ruta rectangular.



Segunda instrucción MCLM.

Tercera instrucción MCLM.

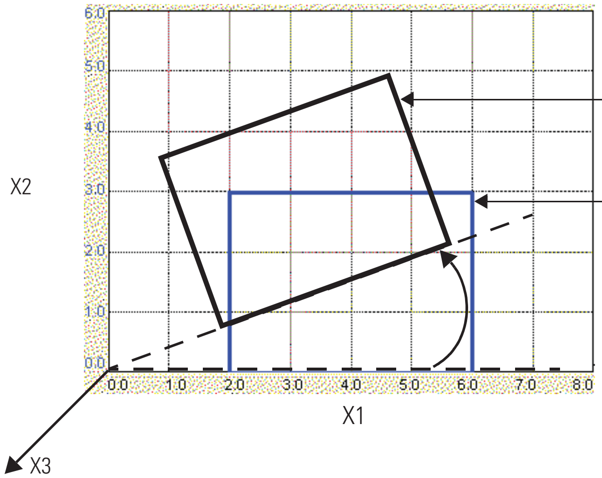
Primera instrucción MCLM.

Representa los sistemas de coordenadas de origen y específico.

Cuarta instrucción MCLM.

Si desea girar las posiciones cartesianas del sistema de coordenadas específico mediante 20· en sentido contrahorario alrededor del eje X3.

1. Ingrese los valores de orientación de 0·, 0·, 20· en la instrucción MCT.
2. Ejecute nuevamente la instrucción MCT para aplicar la orientación a la transformación.
3. Ejecute nuevamente las mismas cuatro instrucciones MCLM.



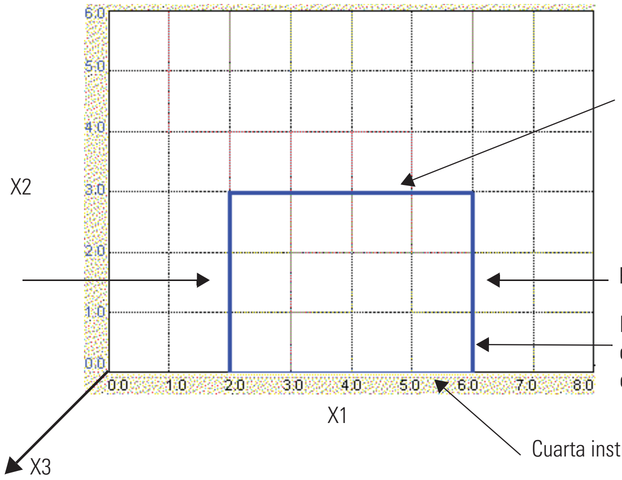
Representa el sistema de coordenadas específico.

Representa el sistema de coordenadas de origen.

Las posiciones cartesianas giran 20· en sentido contrahorario alrededor del eje X3.

Ejemplo de traslación de cambio

Si desea mover el sistema de coordenadas de origen en una ruta rectangular. En ese caso, ejecutar la instrucción MCT para iniciar la transformación. Luego, ejecute cuatro instrucciones MCLM para producir la ruta rectangular.



Segunda instrucción MCLM.

Tercera instrucción MCLM

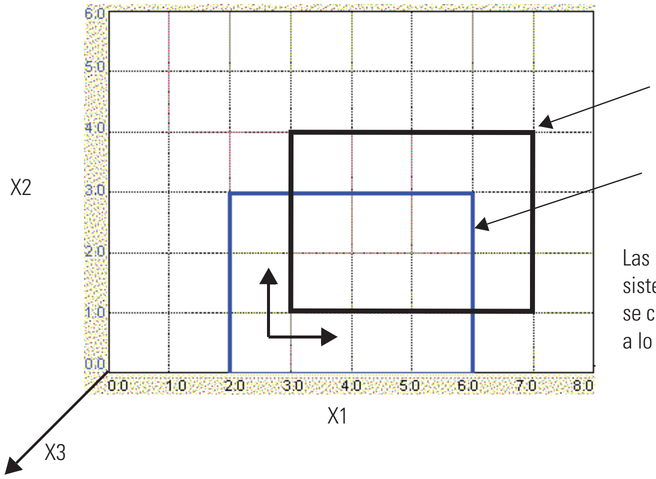
Primera instrucción MCLM.

Representa los sistemas de coordenadas de origen y específico.

Cuarta instrucción MCLM.

Si desea compensar las posiciones cartesianas del sistema de coordenadas específico mediante 1 unidad a lo largo de los ejes X1 y X2.

1. Ingrese los valores de traslación de 1, 1, 0 en la instrucción MCT.
2. Ejecute nuevamente la instrucción MCT para aplicar la traslación a la transformación.
3. Ejecute nuevamente las mismas cuatro instrucciones MCLM.



Representa el sistema de coordenadas específico.

Representa el sistema de coordenadas de origen.

Las posiciones cartesianas del sistema de coordenadas específico se compensan mediante 1 unidad a lo largo de X1 y X2.

Motion Calculate Transform Position (MCTP)

Utilice la instrucción MCTP para calcular la posición de un punto en un sistema de coordenadas al punto equivalente en un segundo sistema de coordenadas.

ATENCIÓN

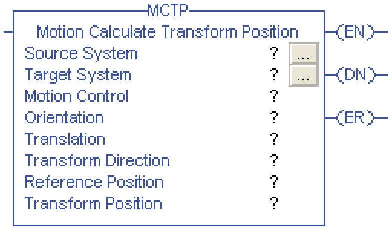
Los tags para instrucciones de control de movimiento se deben utilizar sólo una vez. Reutilizar el tag de control de movimiento en otras instrucciones puede provocar operaciones imprevistas. Esto puede dañar el equipo o causar lesiones personales.

IMPORTANTE

Sólo puede usar esta instrucción con los controladores 1756-L6x.

Operandos

Diagrama de lógica de escalera



MCTP
Motion Calculate Transform Position
Source System ?
Target System ?
Motion Control ?
Orientation ?
Translation ?
Transform Direction ?
Reference Position ?
Transform Position ?

<table>
<tr><th>Operando</th><th>Tipo</th><th>Formato</th><th colspan="2">Descripción</th></tr>
<tr><td>Sistema origen</td><td>COORDINATE_SYSTEM</td><td>Tag</td><td colspan="2">El sistema de coordenadas cartesianas para posiciones cartesianas del robot.</td></tr>
<tr><td>Sistema específico</td><td>COORDINATE_SYSTEM</td><td>Tag</td><td colspan="2">Sistema de coordenadas no cartesianas que controla en equipo real.</td></tr>
<tr><td>Motion Control</td><td>MOTION_ INSTRUCTION</td><td>Tag</td><td colspan="2">Tag de control para la instrucción.</td></tr>
<tr><td rowspan="5">Orientación</td><td rowspan="5">REAL[3]</td><td rowspan="5">Matriz</td><td colspan="2">¿Desea girar la posición específica alrededor del eje X1, X2, o X3?</td></tr>
<tr><td>Si</td><td>Entonces</td></tr>
<tr><td>No</td><td>Deje los valores de matriz en cero.</td></tr>
<tr><td>Sí</td><td>Ingrese los grados de rotación en la matriz. Coloque los grados de rotación alrededor de X1 en el primer elemento de la matriz, y así.</td></tr>
<tr><td colspan="2">Utilice la matriz de los tres REALs aún si el sistema de coordenadas tiene sólo uno o dos ejes.</td></tr>
<tr><td rowspan="5">Traslación</td><td rowspan="5">REAL[3]</td><td rowspan="5">Matriz</td><td colspan="2">¿Desea compensar la posición específica a lo largo del eje X1, X2, o X3?</td></tr>
<tr><td>Si</td><td>Entonces</td></tr>
<tr><td>No</td><td>Deje los valores de matriz en cero.</td></tr>
<tr><td>Sí</td><td>Ingrese las distancias offset en la matriz. Ingrese las distancias offset en la unidades de coordinación. Coloque la distancia offset para X1 en el primer elemento de la matriz, y así consecutivamente.</td></tr>
<tr><td colspan="2">Utilice la matriz de los tres REALs aún si el sistema de coordenadas tiene sólo uno o dos ejes.</td></tr>
</table>

<table>
<tr><th>Operando</th><th>Tipo</th><th>Formato</th><th>Descripción</th></tr>
<tr><td>Dirección de la transformación</td><td>DINT</td><td>Inmediato</td><td>
<table>
<tr><th>Para Robot Type</th><th>Para calcular</th><th>Con la base en</th><th>Y el robot es</th><th>Escoge</th></tr>
<tr><td>All</td><td>Posición cartesiana</td><td>►</td><td>►</td><td>Forward</td></tr>
<tr><td>Cartesian
Delta 2D
Delta 3D
SCARA Delta</td><td>Ángulos de articulación</td><td>►</td><td>►</td><td>Inversa</td></tr>
<tr><td rowspan="4">Independiente articulado
Dependiente articulado
Independiente SCARA</td><td rowspan="4">Ángulos de articulación</td><td rowspan="2">Mismo cuadrante que el punto</td><td>Configuración del brazo derecho</td><td>Brazo derecho inverso</td></tr>
<tr><td>Configuración del brazo izquierdo</td><td>Brazo izquierdo inverso</td></tr>
<tr><td rowspan="2">Cuadrante opuesto desde el punto</td><td>Configuración del brazo derecho</td><td>Espejo brazo derecho inverso</td></tr>
<tr><td>Configuración del brazo izquierdo</td><td>Espejo brazo izquierdo inverso</td></tr>
</table>
</td></tr>
<tr><td>Posición de referencia</td><td>REAL[3]</td><td>Matriz</td><td>
<table>
<tr><th>Si la dirección de transformación es</th><th>Entonces, ingrese una matriz que tiene</th></tr>
<tr><td>Forward</td><td>Ángulos de articulación</td></tr>
<tr><td>Inversa</td><td>Posiciones cartesianas</td></tr>
</table>
</td></tr>
<tr><td>Posición de la transformación</td><td>REAL[3]</td><td>Matriz</td><td>Matriz que almacena la posición calculada</td></tr>
</table>

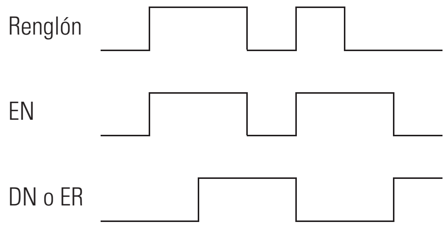
Texto estructurado

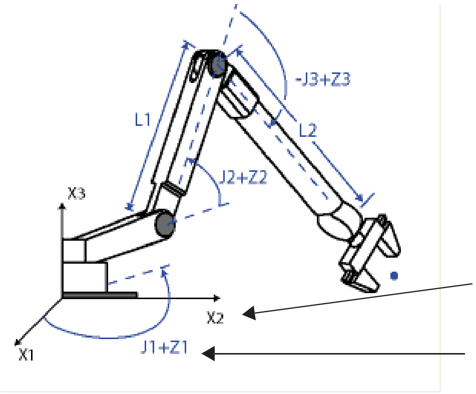
```
MCTP(Source System,Target System,
Motion Control,Orientation,Translation,
Transform Direction,Reference Position,
Transform Position);
```

Los operandos del texto estructurado son iguales a los operandos de la lógica de escalera. Ingrese la dirección de transformación sin espacios.

Ejemplo: Ingrese una dirección de transformación del Brazo izquierdo inverso como InverseLeftArm.

Tipo de datos Motion Instruction

Para ver si	**Verifique si este bit está en**	**Tipo de datos**	**Notas**
El renglón es verdadero.	EN	BOOL	A veces, el bit EN permanece activo aún si el renglón pasa a falso. Esto sucede si el renglón cambia a falso antes de que la instrucción se ejecute o falle. 
La instrucción ha sido ejecutada.	DN	BOOL	
Ha ocurrido un error.	ER	BOOL	Identifique el número del error en el campo de código de error del tag Motion Control, luego, consulte Códigos de error (ERR) para las instrucciones de control de movimiento en página 395 de este manual.



Puede proporcionar a la instrucción las posiciones X1, X2, y X3 y obtener los ángulos J1, J2, y J3 correspondientes.

O puede proporcionar a la instrucción los ángulos J1, J2 y J3 y obtener las posiciones X1, X2, y X3 correspondientes.

La instrucción MCTP es similar a la instrucción MCT, excepto la instrucción MCTP que no inicia una transformación. Esta calcula una posición cada vez que la ejecuta.

Pautas de programación

Siga estas pautas para utilizar una instrucción MCTP.

Pautas para instrucción MCTP

Pauta	Ejemplos y Notas
Alterne el renglón de falso a verdadero para ejecutar la instrucción.	Esta es una instrucción transicional: En el diagrama de lógica de escalera, alterne la condición de entrada del renglón de falso a verdadero cada vez que desee ejecutar la instrucción.
En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición.	En texto estructurado, las instrucciones se ejecutan cada vez que son escaneadas. Acondicione la instrucción de manera que sólo se ejecute en una transición. Utilice alguno de estos dos métodos: • Calificador de una acción SFC • Construcción de texto estructurado

Indicadores de estado aritmético

No afectados

Condiciones de fallo

Ninguno

Códigos de error

Consulte Códigos de error (ERR) para las instrucciones de control de movimiento.

Códigos de error extendidos

Ninguno

Cambia a Bit de estado

Ninguno

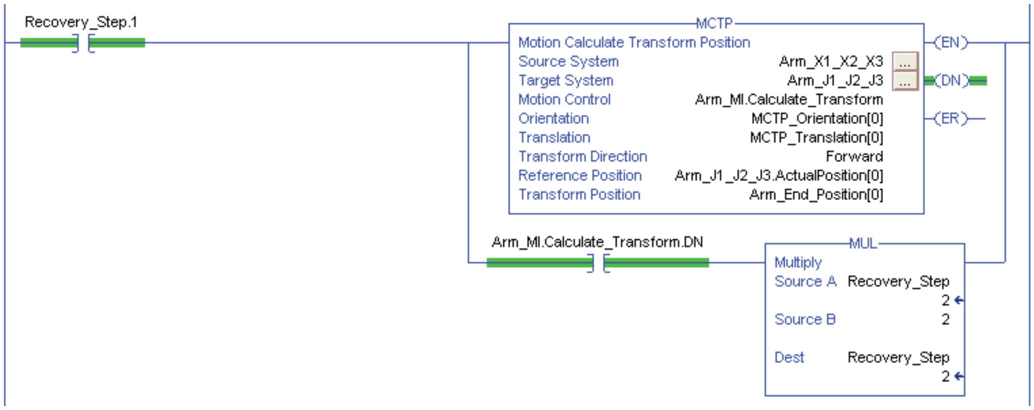
Ejemplo 1

Si desea escribir una secuencia de recuperación para fallos. Como uno de sus pasos, usted desea obtener la posición actual de un robot independiente articulado. Es ese caso, usted puede utilizar una instrucción MCTP para calcular la posición cartesiana del robot cuando conoce sus ángulos de articulación.

Calcular Posición – Diagrama de lógica de escalera

Si Recovery_Step.1 se activa, entonces calcule las posiciones X1, X2, y X3 del robot basado en sus ángulos de articulación actuales

Cuando la instrucción está ejecutada, la instrucción MUL lleva la secuencia al próximo paso.



Motion Calculate Transform Position

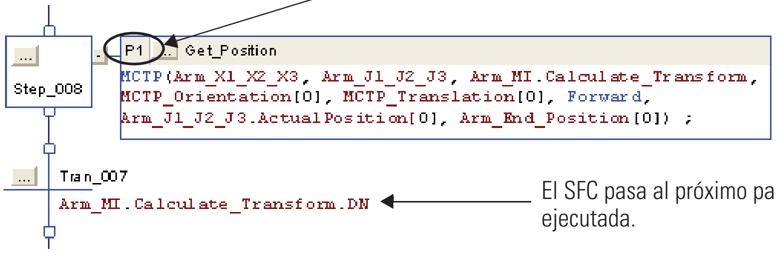
Source System	Arm_X1_X2_X3
Target System	Arm_J1_J2_J3
Motion Control	Arm_MI.Calculate_Transform
Orientation	MCTP_Orientation[0]
Translation	MCTP_Translation[0]
Transform Direction	Forward
Reference Position	Arm_J1_J2_J3.ActualPosition[0]
Transform Position	Arm_End_Position[0]

Multiply

Source A	Recovery_Step 2
Source B	2
Dest	Recovery_Step 2

Calcular Posición – Texto estructurado

Este paso calcula las posiciones X1, X2, y X3 del robot basado en sus ángulos de articulación actuales. El calificador P1 limita esto al primer escán del paso.



```
MCTP(Arm_X1_X2_X3, Arm_J1_J2_J3, Arm_MI.Calculate_Transform,
MCTP_Orientation[0], MCTP_Translation[0], Forward,
Arm_J1_J2_J3.ActualPosition[0], Arm_End_Position[0]) ;
```

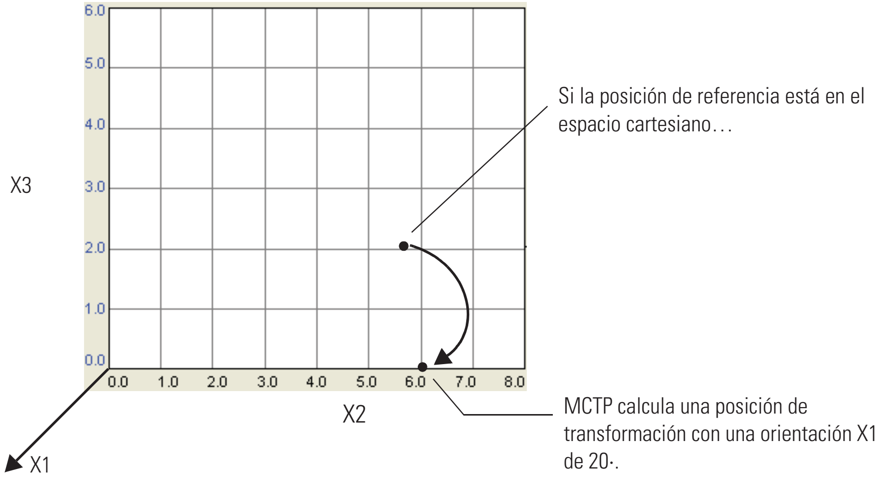
```
Arm_MI.Calculate_Transform.DN
```

El SFC pasa al próximo paso cuando la instrucción MCTP está ejecutada.

Ejemplo 2

Si desea ingresar los valores de orientación de 20·, 0·, 0· en el ejemplo 1. En este ejemplo, la instrucción MCTP realiza una transformación de avance.

Cambie orientación



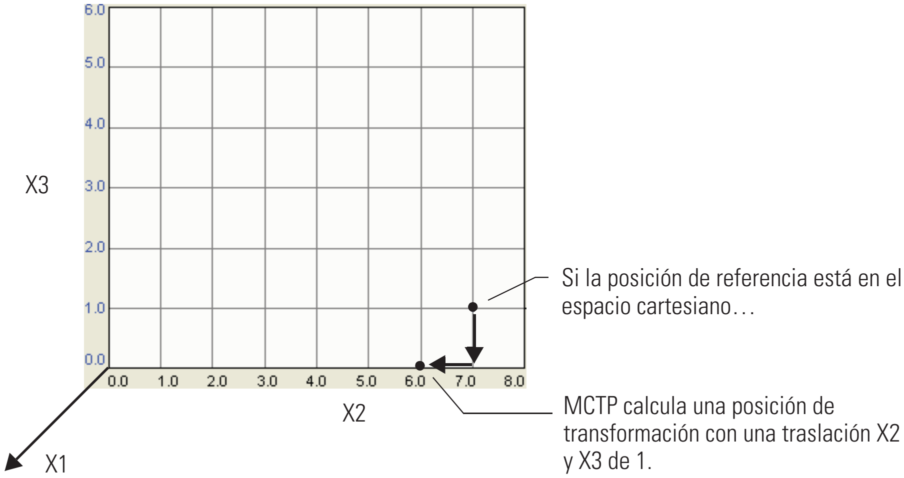
Si la posición de referencia está en el espacio cartesiano…

MCTP calcula una posición de transformación con una orientación X1 de 20·.

Ejemplo 3

Si desea ingresar los valores de traslación de 0, 1, 1 en el ejemplo 1. En este ejemplo, la instrucción MCTP realiza una transformación de avance.

Cambie traslación



Si la posición de referencia está en el espacio cartesiano…

MCTP calcula una posición de transformación con una traslación X2 y X3 de 1.

Ejemplo 4

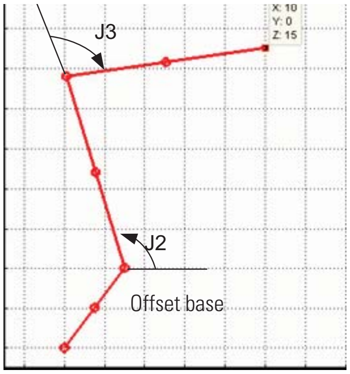
Si su robot tiene offsets base, puede haber hasta cuatro maneras diferentes de alcanzar el punto establecido. Si su robot tiene esta geometría:

- L1 = 10
- L2 = 10
- X1b = 3.0
- X3b = 4.0

Este ejemplo muestra las maneras de llegar a la posición de X1 = 10, X2 = 0, y X3 = 15

Dirección de transformación.

Brazo izquierdo inverso

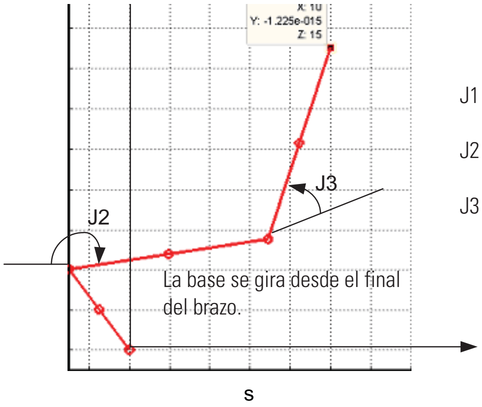


J1 = 0

J2 = 106.84

J3 = –98.63

Espejo brazo izquierdo inverso



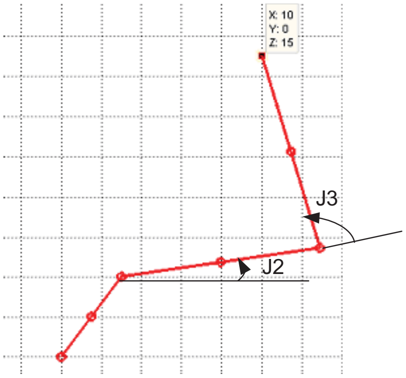
La base se gira desde el final del brazo.

J1 = 180

J2 = 171.39

J3 = –63.26

Brazo derecho inverso

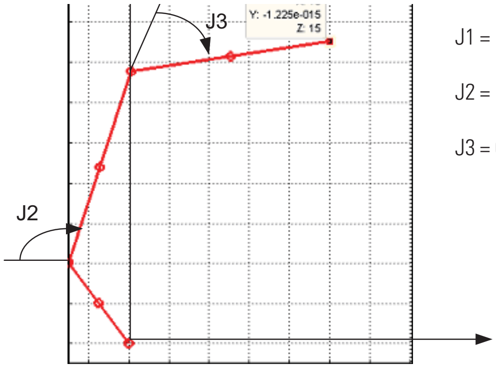


J1 = 0

J2 = 8.22

J3 = 98.63

Espejo brazo derecho inverso



J1 = 180

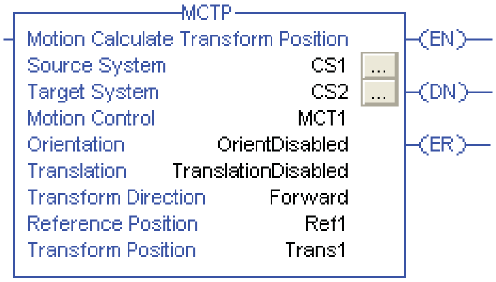
J2 = 108.14

J3 = 63.26

Flujo de datos de la instrucción MCTP entre dos sistemas de coordenadas

La siguiente ilustración muestra el flujo de datos cuando se ejecuta una instrucción MCTP para realiza una transformación de avance y una transformación inversa. El indicador CS1 representa un sistema de coordenadas cartesianas que contiene ejes X1, X2 y X3 como el origen de la instrucción MCTP. El indicador CS2 representa un sistema de coordenadas de articulación que contiene ejes J1, J2 y J3 como el específico de la instrucción MCTP.

Flujo de datos cuando se ejecuta un movimiento con una instrucción MCTP – Transformación de avance



MCTP
Motion Calculate Transform Position
Source System CS1
Target System CS2
Motion Control MCT1
Orientation OrientDisabled
Translation TranslationDisabled
Transform Direction Forward
Reference Position Ref1
Transform Position Trans1

Datos de entrada

CS2: Datos	ORIGEN
Longitudes de vínculo (L1, L2)	Diálogo del sistema de coordenadas
Offsets base (X1b, X2b, X3b)	Diálogo del sistema de coordenadas
Offsets de efectores finales (X1e, X2e, X3e)	Diálogo del sistema de coordenadas
Orientaciones del ángulo cero (Z1, Z2, Z3)	Diálogo del sistema de coordenadas
Orientación (Matriz [3])	Plantilla de la instrucción
Translación (Matriz [3])	Plantilla de la instrucción
Dirección de la transformación	Plantilla de la instrucción
Posición de referencia Articulación típica – Específico	Plantilla de la instrucción

Instrucción ejecutada

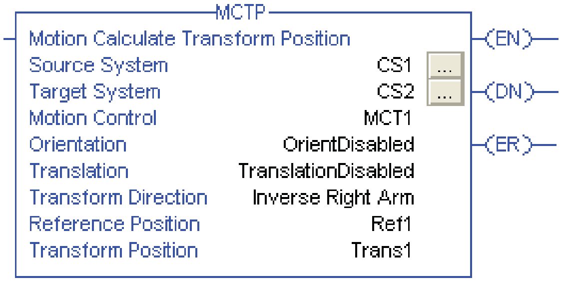
➡

MCTP

Salida calculada

CS1: Datos	Destino
Posiciones cartesianas (X1, X2, X3)	Plantilla de la instrucción
Cartesiano típico	Posición de la transformación

Flujo de datos cuando se ejecuta un movimiento con una instrucción MCTP – Transformación inversa



MCTP
Motion Calculate Transform Position
Source System CS1
Target System CS2
Motion Control MCT1
Orientation OrientDisabled
Translation TranslationDisabled
Transform Direction Inverse Right Arm
Reference Position Ref1
Transform Position Trans1

Datos de entrada

CS1: DATOS	ORIGEN
Longitudes de vínculo (L1, L2)	Diálogo del sistema de coordenadas
Offsets base (X1b, X2b, X3b)	Diálogo del sistema de coordenadas
Offsets de efectores finales (X1e, X2e, X3e)	Diálogo del sistema de coordenadas
Orientaciones del ángulo cero (Z1, Z2, Z3)	Diálogo del sistema de coordenadas
Orientación (Matriz [3])	Plantilla de la instrucción
Translación (Matriz [3])	Plantilla de la instrucción
Dirección de la transformación	Plantilla de la instrucción
Posición de referencia Cartesiano típico – Origen	Plantilla de la instrucción

Instrucción ejecutada

➡

MCTP

Salida calculada

CS2: Datos	Destino
Posiciones de la articulación (J1, J2, J3)	Plantilla de la instrucción
Articulación típica	Posición de la transformación

Motion Coordinated Shutdown Reset (MCSR)

Utilice la instrucción MCSR para restablecer todos los ejes en un sistema de coordenadas. La instrucción MCSR restablece los ejes desde un estado de interrupción a un estado de eje preparado. Esta instrucción también borra cualquier falla del eje

ATENCIÓN ⚠

Los tags para instrucciones de control de movimiento se deben utilizar sólo una vez. Reutilizar el tag de control de movimiento en otras instrucciones puede provocar operaciones imprevistas. Esto puede dañar el equipo o causar lesiones personales.

Operandos:

Lógica de escalera de relés

Operando	Tipo	Formato	Descripción
Sistema de coordenadas	COORDINATE_SYSTEM	Tag	Nombre del eje que proporciona la posición de entrada a la leva de salida. La elipsis ejecuta el diálogo Axis Properties.
Motion Control	MOTION_ INSTRUCTION	Tag	Estructura utilizada para acceder a los parámetros de estado de la instrucción.

MCSR(CoordinateSystem,
MotionControl);

Texto estructurado

Los operandos son iguales a los de la instrucción MCSR de lógica de escalera de relés.

Sistema de coordenadas

El operando del sistema de coordenadas indica el conjunto de los ejes de movimiento que definen las dimensiones de un sistema de coordenadas cartesianas. Para esta versión, el sistema de coordenadas acepta hasta tres (3) ejes primarios. Sólo aquellos ejes configurados como ejes primarios (hasta 3) se incluyen en los cálculos de velocidad coordinada.

Motion Control

Los siguientes bits de control están afectados por la instrucción MCSR.

Bit de control afectados por instrucción MCSR

Mnemónico	Descripción
Bit .EN (Habilitado) 31	El bit Habilitado se establece cuando las transiciones del renglón cambian de falso a verdadero. Este bit se restablece cuando el renglón cambia de verdadero a falso.
Bit .DN (Listo) 29	El bit Listo se establece cuando la interrupción coordinado restablecida se inicia correctamente. Este bit se restablece cuando las transiciones del renglón cambian de verdadero a falso.
Bit .ER (Error) 28	El bit Error se establece cuando el restablecimiento de la interrupción coordinada no puede iniciar correctamente. Se restablece cuando el renglón va de falso a verdadero.

Ésta es una instrucción transicional:

- En la lógica de escalera de relés, alterne la condición de entrada del renglón de restablecido a establecido cada vez que deba ejecutarse la instrucción.
- En texto estructurado, condicione la instrucción de manera que sólo se ejecute cuando ocurra una transición. Vea el Apéndice C.

Indicadores de estado aritméticos:

No afectados

Condiciones de fallo:

Ninguno

Códigos de error:

Consulte Códigos de error (ERR) para las instrucciones de control de movimiento.

MCSR Cambia a bits de estado:

Los Bit de estado proporcionan un medio para monitorear el progreso de la instrucción de control de movimiento. Hay tres tipos de bits de estado que brindan la información pertinente. Estos son: Bit de estado del eje, bits de estado del sistema de coordenadas y bits de estado del movimiento coordinado. Cuando la instrucción MCS comienza, los bits de estado sufren los siguientes cambios.

Bits de estado del eje

Nombre del bit	Efecto
CoordinatedMoveStatus	Ningún efecto.

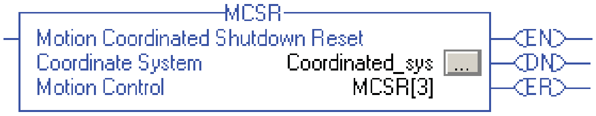
Bits de estado del sistema de coordenadas

Nombre del bit	Efecto
ShutdownStatus	Borra el bit de estado de interrupción.

Bits de estado del movimiento coordinado

Nombre del bit	Efecto
MovePendingStatus	Pone un funcionamiento la cola de instrucción y borra el bit de estado.
MovePendingQueueFullStatus	Pone un funcionamiento la cola de instrucción y borra el bit de estado.

Ejemplo de la lógica de escalera de relé



MCSR
Motion Coordinated Shutdown Reset
Coordinate System Coordinated_sys
Motion Control MCSR[3]

Texto estructurado

MCSR(Coordinated_sys,MCSR[3]);

Notas:

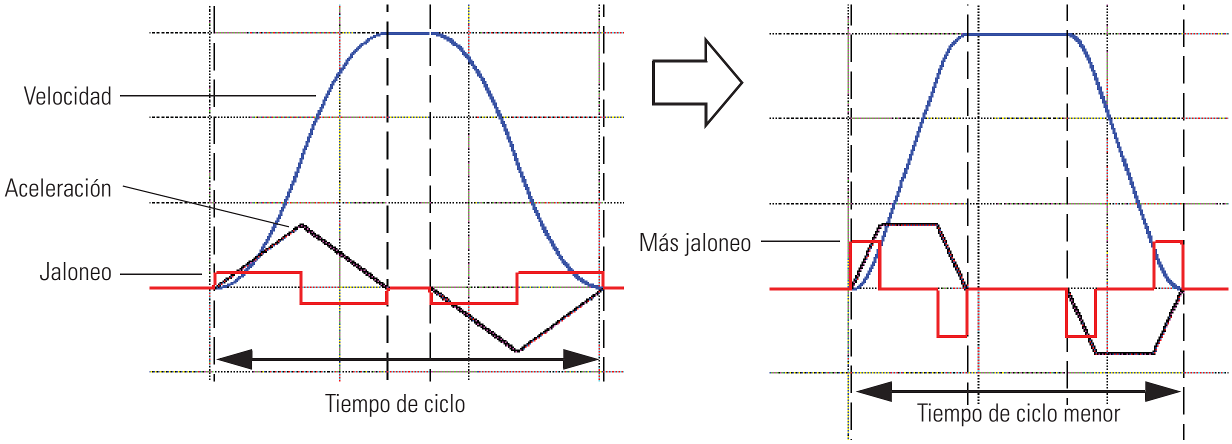
Ajustar un perfil de curva en S

Introducción

Use este procedimiento para equilibrar la suavidad y tiempo de ciclo de movimiento que usa un perfil de curva en S.

Haga esto cuando

Realice este procedimiento cuando quiera disminuir el tiempo de ciclo de un perfil de movimiento de curva en S pero manteniendo algo de suavidad del perfil.



Para usar este procedimiento, su aplicación debe cumplir estos requisitos:

- El controlador está en revisión 16 o posterior.
- Una de estas instrucciones produce el movimiento:
  - Motion Axis Move (MAM)
  - Motion Axis Jog (MAJ)
  - Motion Axis Stop (MAS)
- La instrucción usa un perfil de curva en S

Antes de empezar

IMPORTANTE En este procedimiento, aumenta el jaloneo. Esto aumenta el esfuerzo en el equipo y carga. Asegúrese de poder identificar cuándo el equipo o carga ha alcanzado su límite de jaloneo.

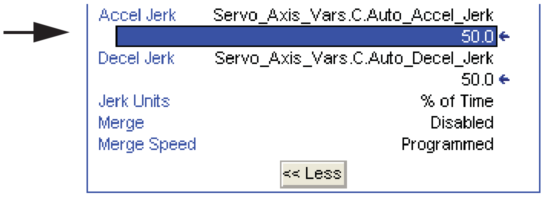
Procedimiento

1. ¿Las unidades de jaloneo están establecidas en % de tiempo?

Si las unidades de jaloneo son	Entonces
% de tiempo  Jerk Units % of Time Merge Disabled Merge Speed Programmed 	Continúe con paso 2.
% de Máximo  Jerk Units % of Maximum Merge Disabled Merge Speed Programmed 	A. Cambie las unidades de jaloneo a % de tiempo.  Jerk Units % of Time Merge Disabled Merge Speed Programmed
Unidades por seg3  Jerk Units Units per sec3 Merge Disabled Merge Speed Programmed 	B. Continúe con paso 2.

2. Establezca los valores de jaloneo en 50% de Tiempo.

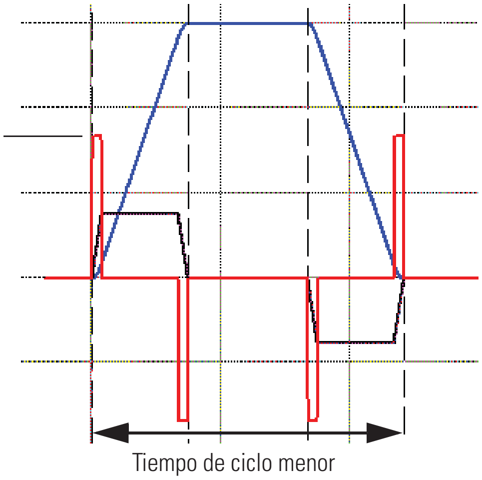
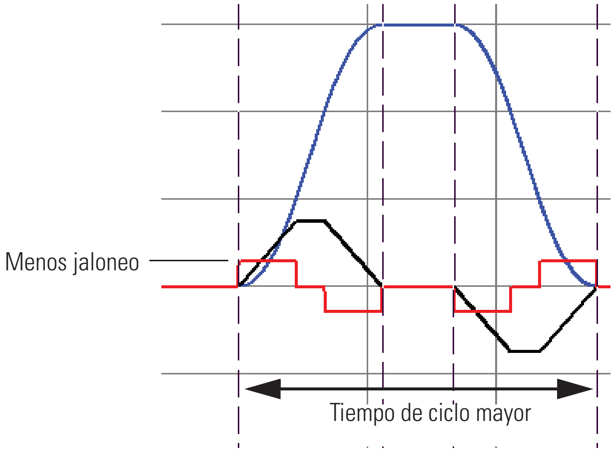
Ejemplo



Accel Jerk Servo_Axis_Vars.C.Auto_Accel_Jerk
50.0
Decel Jerk Servo_Axis_Vars.C.Auto_Decel_Jerk
50.0
Jerk Units % of Time
Merge Disabled
Merge Speed Programmed

3. Pruebe su equipo y observe el jaloneo.

4. Ajuste los valores de jaloneo.

Si hay	Entonces	Que produce
POCO jaloneo	Reduzca el % de tiempo.	
Demasiado jaloneo	Aumente el % de tiempo.	

5. Repita los pasos 3 y 4 hasta obtener el equilibrio deseado entre suavidad y tiempo de ciclo.

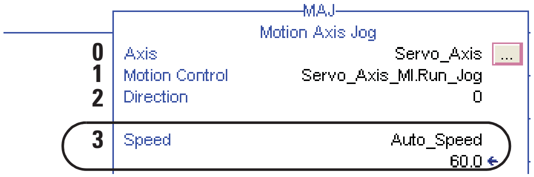
Recursos adicionales

- Programar un perfil de velocidad en la página 24
- Troubleshoot Axis Motion en la page 9

Notas:

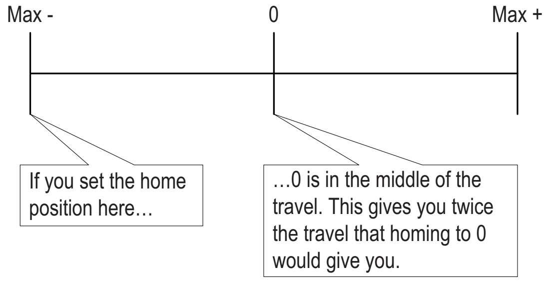
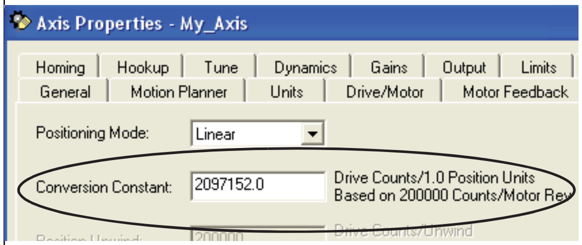
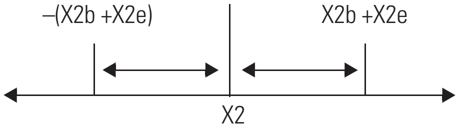
Códigos de error (ERR) para las instrucciones de control de movimiento

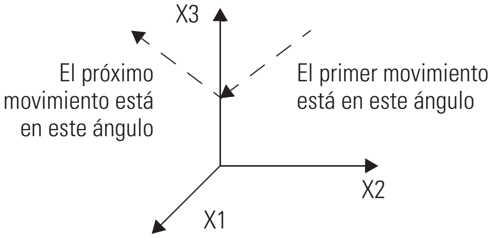
Error	Medida correctiva o causa	Notas
3	Busque otra ocurrencia de este tipo de instrucción. Vea si su bit EN está activado pero sus bits DN y ER están desactivados (habilitados pero no listo o con error). Aguarde hasta que sus bits DN o EN se activen.	Colisión de ejecuciones No puede ejecutar una instrucción si el mismo tipo de instrucción está habilitado pero no listo o con error.
4	Abra el lazo del servo antes de ejecutar esta instrucción.	Error de estado de servo activado
5	Cierre el lazo del servo antes de ejecutar esta instrucción.	Error de estado de servo desactivado Para una instrucción de movimiento coordinado, vea el código de error extendido (EXERR). Identifica qué eje provocó el error. **Ejemplo:** Si EXERR está en cero, verifique que el eje esté en dimensión cero.
6	Inhabilite el eje del variador.	Error de estado de variador activado
7	Ejecute la instrucción Restablecer interrupción del eje de movimiento (MASR) o comando directo para restablecer el eje.	Error del estado de desactivación Para una instrucción de movimiento coordinado, vea el código de error extendido (EXERR). Identifica qué eje provocó el error. **Ejemplo:** Si EXERR está en cero, verifique que el eje esté en dimensión cero.
8	El tipo de eje configurado no es correcto.	Tipo de eje incorrecto Para una instrucción de movimiento coordinado, vea el código de error extendido (EXERR). Identifica qué eje provocó el error. **Ejemplo:** Si EXERR está en cero, verifique que el eje esté en dimensión cero.
9	La instrucción intentó ejecutarse en una dirección que empeora la condición de fin de carrera actual.	Condición de fin de carrera
10	La referencia de eje maestro es idéntica a la referencia de eje esclavo.	Conflicto con el eje maestro
11	Por lo menos un eje no está configurado en un módulo de movimiento físico o no ha sido asignado a un grupo de movimiento.	Eje no configurado Para una instrucción de movimiento coordinado, vea el código de error extendido (EXERR). Identifica qué eje provocó el error. **Ejemplo:** Si EXERR está en cero, verifique que el eje esté en dimensión cero.
12	El envío de mensajes al servomódulo entró en fallo.	Fallo de envío de mensajes al servomódulo

Error	Medida correctiva o causa	Notas
13	Vea el código de error extendido (EXERR) para la instrucción. Identifica un operando que está afuera de su rango. **Ejemplo:** Haga de cuenta que una instrucción MAJ tiene un ERR = 13 y un EXERR = 3. En ese caso, cambie la velocidad para que esté en su rango.  MAJ Motion Axis Jog 0 Axis Servo_Axis ... 1 Motion Control Servo_Axis_MI.Run_Jog 2 Direction 0 3 Speed Auto_Speed 60.0	Parámetro fuera de rango Un EXERR = 0 significa que el primer operando de la instrucción está fuera de su rango.
14	La instrucción no puede aplicar los parámetros de ajuste debido a un error en la instrucción de ejecución de ajuste.	Error del proceso de ajuste
15	La instrucción no puede aplicar los parámetros de diagnóstico debido a un error en la instrucción de ejecución de la prueba de diagnóstico.	Error del proceso de prueba
16	Aguarde hasta que el proceso de vuelta a la posición inicial este listo.	Error de comando Home
17	La instrucción intentó ejecutar un movimiento giratorio en un eje no configurado para funcionamiento giratorio.	Modo de eje no giratorio
18	El tipo de eje está configurado como no usado.	Tipo de eje no usado
19	El grupo de movimiento no está en estado sincronizado. La causa puede ser un servomódulo faltante o mal configurado.	Grupo no sincronizado
20	El eje está en estado de fallo.	Eje en estado de fallo
21	El grupo está en estado de fallo.	Grupo en estado de fallo
22	Detenga el eje antes de ejecutar esta instrucción.	Eje en movimiento
23	Una instrucción intentó realizar un cambio dinámico no válido.	Cambio dinámico no válido
24	Saque el controlador del modo de prueba.	Modo op AC no válido
25	Intentó ejecutar una instrucción que no es correcta.	Instrucción no válida
26	La longitud de la matriz de leva no es correcta.	Longitud de leva no válida
27	La longitud de la matriz del perfil levas no es válida	Longitud de perfil de levas no válido
28	Tiene un tipo de segmento no válido en el elemento de la leva.	Tipo de leva no válido
29	Tiene un orden de elementos de leva no válido.	Orden de leva no válido
30	Intentó ejecutar un perfil de leva mientras se estaba calculando.	Calculando el perfil de leva
31	La matriz del perfil de levas que intentó ejecutar está en uso.	Perfil de leva en uso
32	La matriz del perfil de levas que intentó ejecutar no se ha calculado.	Perfil de leva no calculado
33	Intentó ejecutar una instrucción MAH sin una leva de posición en proceso.	Posición de leva no habilitada
34	Una instrucción MAH está intentando iniciarse mientras un registro aún está en ejecución.	Registro en progreso
35	El controlador o el Módulo de leva de salida no es compatible con la leva de salida, el eje, la entrada o la salida especificada.	Objetivo de ejecución no válido

Error	Medida correctiva o causa	Notas
36	El tamaño de la matriz de leva de salida no es compatible o el valor de uno de sus miembros está fuera de rango.	Leva de salida no válida ExErr#1: El bit de salida es menor que 0 o mayor que 31. ExErr#2: El tipo de enclavamiento es menor que 0 o mayor que 3. ExErr#3: El tipo de desenclavamiento es menor que 0 o mayor que 5. ExErr#4: La posición izquierda o derecha está fuera del rango de la leva y el tipo de enclavamiento o desenclavamiento está establecido en “Position” o “Position and Enable”. ExErr#5: La duración es menor o igual a 0 y el tipo de desenclavamiento está establecido en “Duration” o “Duration and Enable”. ExErr#6: El tipo de habilitación es menor que 0 o mayor que 3 y el tipo de enclavamiento o desenclavamiento está establecido en “Enable”, “Position and Enable”, o “Duration and Enable”. ExErr#7: El bit Habilitar es menor que 0 o mayor que 31 y el tipo de enclavamiento o desenclavamiento está establecido en “Enable”, “Position and Enable”, o “Duration and Enable”. ExErr#8: El tipo de enclavamiento está establecido en “Inactive” y el tipo de desenclavamiento está establecido ya sea en “Duration” o en “Duration and Enable”.
37	El tamaño de la matriz de compensación de salida no es compatible o el valor de uno de sus miembros está fuera de rango.	Compensación de salida no válida ExErr#1: Modo menor que 0 o mayor que 3. ExErr#2: Tiempo de ciclo menor o igual a 0 y el modo está establecido en “Pulsed” o “Inverted and Pulsed”. ExErr#3: Ciclo de servicio menor que 0 o mayor que 100 y el modo está establecido en “Pulsed” o “Inverted and Pulsed”.
38	El tipo de datos del eje no es válido. Es incorrecto para la operación.	Tipo de datos de eje no válido Para una instrucción de movimiento coordinado, vea el código de error extendido (EXERR). Identifica qué eje provocó el error. **Ejemplo:** Si EXERR está en cero, verifique que el eje esté en dimensión cero.
39	Tiene un conflicto en su proceso. La prueba y el ajuste no pueden ejecutarse al mismo tiempo.	Conflicto de proceso
40	Está intentando ejecutar una instrucción MSO o MAH cuando el variador está inhabilitado localmente.	Variador localmente inhabilitado
41	La configuración de vuelta a la posición inicial no es válida. Tiene una instrucción absoluta de vuelta a la posición inicial cuando la secuencia de vuelta a la posición inicial no es inmediata.	Config de vuelta a la posición inicial no válida

Error	Medida correctiva o causa	Notas
42	La instrucción MASD o MGSD ha sobrepasado el tiempo de espera porque no recibió el bit de estado de interrupción. Generalmente, un problema programático ocurre cuando la instrucción MASD o MGSD es seguida por una instrucción de restablecimiento, la cual es iniciada antes que el bit de Shutdown haya sido recibido por la instrucción de interrupción.	Tiempo de espera de estado de interrupción
43	Intentó activar más instrucciones de control de movimiento de las que puede retener la cola de instrucciones.	Cola del sistema de coordenadas llena
44	Ha trazado una línea con 3 puntos y no se puede determinar un punto intermedio del núcleo o un núcleo plano.	Error de colinealidad circular
45	Ha especificado un radio de 1 punto o un núcleo “trazado una línea”, punto intermedio y no se puede determinar ningún radio de núcleo o núcleo plano, punto intermedio.	Error de fin de arranque circular
46	El núcleo programado no es equidistante del punto de inicio y del punto final.	Error de desigualdad R1 R2 circular
47	Póngase en contacto con el servicio de asistencia técnica de Rockwell Automation.	Error de solución circular infinita
48	Póngase en contacto con el servicio de asistencia técnica de Rockwell Automation.	Error circular sin soluciones
49	|R| < 0.01. R es básicamente muy pequeño para ser utilizado en cálculos.	Error pequeño R circular
50	El tag del sistema de coordenadas no está asociada con el grupo de movimiento.	El sistema de coordenadas no está en el grupo
51	Ha establecido su Tipo de terminación en Posición real con un valor de 0. Este valor no es compatible.	Tolerancia real no válida
52	Al menos un eje está experimentando actualmente un movimiento coordinado en otro sistema de coordenadas.	Error de movimiento de coordinación en proceso
53	Desinhiba el eje	El eje está inhibido
54	1. Abra las propiedades para los ejes. 2. En la pestaña Dynamics, introduzca un valor para la desaceleración máxima.	Desacel máx cero No puede iniciar el movimiento si la desaceleración máxima del eje es cero.
61	Vea el código de error extendido (EXERR) para la instrucción.	Conflicto de conexión
62	Cancele la transformación que controla este eje o no utilice esta instrucción mientras la transformación está activa.	Transformación en progreso No puede ejecutar esta instrucción si el eje es parte de una transformación activa.
63	Cancele la transformación que controla este eje o espere hasta que la transformación haya terminado de mover el eje.	Eje en movimiento de transformación No puede ejecutar esta instrucción si una transformación está moviendo el eje.
64	Use un sistema de coordenadas cartesianas.	Auxiliar no compatible No puede usar un sistema de coordenadas no cartesianas con esta instrucción.

Error	Medida correctiva o causa	Notas
65	El eje se movió demasiado lejos y el controlador no puede almacenar la posición. Para evitar este error, configure límites de carrera suaves para mantener al eje dentro del rango de posición. Un modo de obtener más carrera es utilizar la posición negativa máx. o positiva máx. como su posición inicial. **Ejemplo**  	Overflow de la posición del eje El rango para la posición depende de la constante de conversión del eje.  Axis Properties - My_Axis Positioning Mode: Linear Conversion Constant: 2097152.0 Drive Counts/1.0 Position Units Based on 200000 Counts/Motor Rev • Posición positiva máxima = 2,147,483,647/constante de conversión del eje • Posición negativa máxima = –2,147,483,648/constante de conversión del eje Haga de cuenta que tiene una constante de conversión de 2,097,152 conteos/pulgada. En es caso: • Posición positiva máxima = 2,147,483,647/2,097,152 conteos/pulgada = 1023 pulgadas • Posición negativa máxima = –2,147,483,648/2,097,152 conteo/pulgada = –1023 pulgadas Para una instrucción de movimiento coordinado, vea el código de error extendido (EXERR). Identifica qué eje provocó el error. ExErr#1: El eje 0 provocó el Error ExErr#2: El eje 1 provocó el Error ExErr#1: El eje 2 provocó el Error
66	Asegúrese de mantener el robot en la solución de brazo en la que lo configuró. Puede configurar el robot ya sea en la solución de brazo izquierdo o de brazo derecho.	Está tratando de plegar hacia atrás en sí mismo un robot articulado de dos ejes independiente o dependiente en los límites del cuadrante.
67	· Cambie las posiciones específicas a los valores que están dentro del alcance del robot. · Si X2b +X2e no es cero, manténgase afuera de esta región:  	Posición de transformación no válida · Está tratando de moverse hacia un lugar que el robot no puede alcanzar. · Se intentó la instrucción MCT mientras estaba en el origen. Para evitar que el robot se pliegue hacia atrás sobre sí mismo o que se extienda más allá de su alcance, se calculan límites unidos internamente mediante firmware para robots Delta2D, Delta3D y SCARA Delta. Si intenta configurar un movimiento que viola estos límites, se produce este error.
68	Mueva las conexiones para que el extremo del robot no esté en el origen del sistema de coordenadas.	Transformación en el origen No puede iniciar la transformación si los ángulos de la conexión dan como resultado X1 = 0 y X2 = 0.

Error	Medida correctiva o causa	Notas
69	· Revise la configuración de velocidad máxima de las conexiones. · Use las posiciones específicas que evitan que el robot se estire completamente o se pliegue hacia atrás sobre sí mismo en el origen del sistema de coordenadas. · Muévase en una línea relativamente recta a través de las posiciones donde X1 = 0 y X2 = 0.	Velocidad de unión máxima excedida La velocidad calculada es muy alta. Esto ocurre cuando el robot: • se estira completamente. • se pliega hacia atrás sobre sí mismo. • se aleja de X1 = 0 y X2 = 0 en un ángulo diferente del que se aproximó a esa posición. • está configurado con el límite de velocidad incorrecto. **Ejemplo:** Estos movimientos producen este error. 
70	Busque ejes de origen u objeto que estén configurados en modo de posicionamiento giratorio. Cámbielos al modo de posicionamiento lineal.	Los ejes en la transformación deben ser lineales Una transformación funciona sólo con ejes lineales.
71	Aguarde hasta que la transformación que está cancelando esté completamente cancelada.	La transformación se está cancelando
72	Examine las posiciones objetivo. Un ángulo de unión calculado está más allá de +/– 360·.	Unión de ángulo máxima excedida
73	Verifique que cada instrucción MCT en esta cadena esté produciendo posiciones válidas.	Error de encadenamiento en el sistema de coordenadas Esta instrucción MCT es parte de una cadena de instrucciones MCT. Hay un problema con una de las instrucciones en la cadena.
74	Cambie la orientación a ángulos que estén dentro de +/– 360·.	Ángulo de orientación no válido
75	Use esta instrucción únicamente con un controlador 1756-L6x.	Instrucción no compatible Puede usar una instrucción MCT o MCTP únicamente con un controlador 1756-L6x.
76	1. Abra las propiedades para los ejes. 2. En la pestaña Dynamics, introduzca un valor para el jaloneo de desaceleración máximo.	Jaloneo de desacel máx. cero No puede iniciar un movimiento que utilice un perfil de curva en S si el jaloneo de desaceleración máximo para el eje es cero.

Error	Medida correctiva o causa	Notas
77	¿Cuántos ejes hay en su sistema de coordenadas? · 2 – Use una dirección de transformación no reflejada. · 3 – Use una dirección de transformación no inversa.	Dirección de transformación no compatible 1. Está intentando usar las direcciones reflejadas con un sistema de coordenadas de 3 ejes y un offset base sin cero (X2b) o un offset efector (X2e). 2. Un sistema de coordenadas de 2 ejes no es compatible con las direcciones reflejo. 3. Está intentando usar un sistema de coordenadas específico de 2 o tres ejes Cartesiano, Delta2D, Delta3D o SCARA Delta con direcciones de transformación que no son de avance ni inversa. Puede usar direcciones de reflejo inversas únicamente cuando estas dos condiciones son verdaderas: · Tiene un sistema de coordenadas de 3 ejes. · El offset base (X2b) y el offset efector final (X2e) de la dimensión X2 son cero.
78	No superponga las instrucciones de control de movimiento mientras las instrucciones de paro de movimiento están en progreso.	No permitido mientras se detiene No puede superponer ciertas instrucciones de control de movimiento mientras se detiene. Espere hasta que se complete la primera instrucción antes de iniciar la segunda. Para obtener más información, vea la tabla que se encuentra en las páginas 402 y 403.
79	Vuelva su eje a la posición inicial nuevamente.	Error interno de secuencia de vuelta a la posición inicial Si ve este error, vuelva a colocar su eje en la posición inicial en su programa de aplicación. Si el error persiste, póngase en contacto con el servicio de asistencia técnica de Rockwell Automation.
80	Corrija el operando Output MAOC o el módulo del formato de comunicaciones OB16IS.	Operando Output de MAOC no válido Si el operando Output de MAOC hace referencia a un módulo de salida programado OB16IS, se producen dos señas de comprobación adicionales cuando se inicia MAOC. • El operando Output debe hacer referencia al principio del tag de datos de salida del módulo, 'O.Data'. • El formato de comunicaciones del módulo OB16IS debe ser "Dato de salida programado por punto" predeterminado. Si cualquiera de estas señas de comprobación falla, verá este error. ExErr#1: Referencia de tag de datos no válida – El operando Output no está señalando el elemento O.Data del tag de datos de salida del modulo. ExErr#2: Formato de comunicaciones del módulo OB16IS no válido – Se ha cambiado el formato de las comunicaciones OB16IS de su Dato de salida programado por punto predeterminado.

Error	Medida correctiva o causa	Notas
81	No superponga la instrucción MASD o la instrucción de paro MGS con el Modo de paro = Programado en una instrucción activa MGSR.	Partial Group Shutdown Reset Si su programa de aplicación está ejecutando activamente una instrucción MGSR y usted intenta ejecutar una instrucción MASD o una instrucción de paro MGS con un Modo de paro = Programada en uno de los ejes afectados por la instrucción activa MGSR, verá este error en la instrucción MGSR.
82	Active la instrucción MDSC.	La instrucción MDSC no ha sido activada.
85	No efectúe una instrucción MAH en un eje mientras una instrucción MRP está en proceso.	Este error se produce mientras se está ejecutando la vuelta a la posición inicial mientras la instrucción MRP aún está en proceso. Este problema se produce cuando se usa una instrucción de vuelta a la posición inicial inmediata. Además, cuanto mayor sea el valor MRP mayor será el salto del eje.

Obtendrá un error si ciertas instrucciones de control de movimiento se superponen mientras las instrucciones de paro de movimiento están activas. En este caso, se detiene activamente una instrucción y se inicia una segunda instrucción que se superpone con la instrucción activa. La siguiente tabla enumera las ocurrencias de superposición que generarán errores.

En este caso:

- Error #7 = Error del estado de interrupción
- Error #61, ExErr #10 = Conflicto de conexión, Transforma los ejes en movimiento o bloqueados por otra operación
- Error #78 = No permitido mientras se detiene

	Instrucción de paro activa								
	MGS			**MGSD**	**MCS**			**MAS**	
Segunda instrucción iniciada	Modo de paro = Paro rápido	Modo de paro = Inhabilitar rápido	Modo de paro = Programado		Tipo de paro = Movimiento coordinado	Tipo de paro = Transformación coordinada	Tipo de paro = Todos	Todos los tipos de paro excepto Tipo de paro = Todos	Tipo de paro = Todos
MAAT	Error #78	Error #78	Error #78	Error #7	Error #78	Error #78	Error #78	Error #78	Error #78
MRAT	Error #78	Error #78	Error #78	Error #7	Error #78	Error #78	Error #78	Error #78	Error #78
MAHD	Error #78	Error #78	Error #78	Error #7	Error #78	Error #78	Error #78	Error #78	Error #78
MRHD	Error #78	Error #78	Error #78	Error #7	Error #78	Error #78	Error #78	Error #78	Error #78
MAH	Error #78	Error #78	Error #78	Error #7	Error #78	Error #78	Error #78	Error #78	Error #78
MAJ	Error #78	Error #78	Error #78	Error #7			Error #78		Error #78
MAM	Error #78	Error #78	Error #78	Error #7			Error #78		Error #78
MAG	Error #78	Error #78	Error #78	Error #7			Error #78		Error #78
MCD	Error #78	Error #78	Error #78	Error #7	Error #78	Error #78	Error #78	Error #78	Error #78
MAPC	Error #78	Error #78	Error #78	Error #7			Error #78		Error #78
MATC	Error #78	Error #78	Error #78	Error #7			Error #78		Error #78
MDO	Error #78	Error #78	Error #78	Error #7			Error #78		Error #78
MCT	Error #78	Error #78	Error #78	Error #7	Error #61 ExErr #10	Error #61 ExErr #10	Error #61 ExErr #10	Error #61 ExErr #10	Error #61 ExErr #10
MCCD	Error #78	Error #78	Error #78	Error #7			Error #78		Error #78
MCLM/MCCM (Incorporación = Inhabilitada)	Error #78	Error #78	Error #78	Error #7	Error #78	Error #78	Error #78		Error #78
MCLM/MCCM (Incorporación = Habilitada)	Error #78	Error #78	Error #78	Error #7		Error #78	Error #78		Error #78

La siguiente tabla enumera las instancias de superposición adicionales que generarán errores.

		Instrucción de paro activa						
		MGS			**MGSD**	**MCS**	**MAS**	**MASD**
Segunda instrucción iniciada	Tipo de paro	Modo de paro = Paro rápido	Modo de paro = Inhabilitar rápido	Modo de paro = Programado	Ninguno	Tipo de paro = Todos	Tipo de paro = Todos	Ninguno
MGS	Modo de paro = Paro rápido	Error #78	Error #78	Error #78	Error #7			
	Modo de paro = Inhabilitar rápido	Error #78	Error #78	Error #78	Error #7			
	Modo de paro = Programado	Error #78	Error #78	Error #78	Error #7			
MGSR	Ninguno	Error #78	Error #78	Error #78	Error #7			Error #7
MCS	Tipo de paro = Movimiento coordinado	Error #78	Error #78	Error #78	Error #7	Error #78	Error #78	
	Tipo de paro = Transformación coordinada	Error #78	Error #78	Error #78	Error #7	Error #78	Error #78	
	Todos los tipos de paro excepto Tipo de paro = Todos	Error #78	Error #78	Error #78	Error #7			
MAS	Tipo de paro != Todos	Error #78	Error #78	Error #78	Error #7	Error #78	Error #78	Error #7
	Tipo de paro = Todos	Error #78	Error #78	Error #78	Error #7			Error #7
MASR	Ninguno	Error #78	Error #78	Error #78	Error #7			Error #7

Información adicional sobre los códigos de error

Vea estos manuales para obtener más información sobre los códigos de error que se muestran en los variadores y/o en los sistemas multieje de control de movimiento.

Publicación	**Número de publicación**
Kinetix 2000 Multi-Axis Drive User Manual	2093-UM001
Kinetix 6000 Multi-Axis Drive User Manual	2094-UM001
Kinetix 7000 Multi-Axis Drive User Manual	2099-UM001
Ultra 3000 Digital Servo Drive Installation Instructions	2098-IN003
8720 High Performance Drive Installation Instructions	8720MC-IN001
1394 SERCOS Interface Multi-Axis Motion Control System Installation Manual	1394-IN002

Notas:

Apéndice B

Tipos de datos relacionados con el de movimiento (Estructuras)

Introducción

Use este apéndice para obtener información sobre los siguientes tipos de datos relacionados con el movimiento:

Tipo de datos	Página
Estructura CAM	405
Estructura CAM_PROFILE	406
Estructura MOTION_GROUP	406
Tipo de datos MOTION_INSTRUCTION	408
Estructura OUTPUT_CAM	409
Estructura OUTPUT_COMPENSATION	411

Recursos adicionales

Para otros tipos de datos relacionados con el movimiento, vea Módulos de movimiento en Logix5000 Control Systems User Manual, publicación LOGIX-UM002.

Estructura CAM

El tipo de datos de leva consta de pares de puntos esclavos y maestros así como también de un tipo de interpolación. Dado que no hay asociación con una posición o tiempo de eje específico, los valores de puntos no tienen unidades. El tipo de interpolación puede especificarse para cada segmento como lineal o como cúbica. El formato del elemento de la matriz de levas se muestra en la siguiente tabla.

<table>
<tr><th>Mnemónico</th><th>Tipo de datos</th><th colspan="2">Descripción</th></tr>
<tr><td>MASTER</td><td>REAL</td><td colspan="2">El valor x del punto.</td></tr>
<tr><td>SLAVE</td><td>REAL</td><td colspan="2">El valor y del punto.</td></tr>
<tr><td rowspan="4">Tipo de segmento</td><td rowspan="4">DINT</td><td colspan="2">El tipo de interpolación.</td></tr>
<tr><td>Valor</td><td>Descripción</td></tr>
<tr><td>0</td><td>lineal</td></tr>
<tr><td>1</td><td>cúbica</td></tr>
</table>

Estructura CAM_PROFILE

El tipo de datos CAM_PROFILE es una matriz de coeficientes que representa un perfil de levas calculado que puede usarse como entrada para una instrucción de leva de tiempo o leva de posición. El único elemento disponible es Status, que se define en la siguiente tabla.

<table>
<tr><th>Mnemónico</th><th>Tipo de datos</th><th>Descripción</th></tr>
<tr><td>Status</td><td>DINT</td><td>El parámetro Status se usa para indicar que se ha calculado el elemento de la matriz del perfil de levas. Si se intenta ejecutar una instrucción de operación de levas usando un elemento no calculado en un perfil de levas, la instrucción produce un error.
<table>
<tr><th>Valor</th><th>Descripción</th></tr>
<tr><td>0</td><td>El elemento de perfil de levas no ha sido calculado.</td></tr>
<tr><td>1</td><td>El elemento de perfil de levas se está calculando.</td></tr>
<tr><td>2</td><td>El elemento de perfil de levas ha sido calculado.</td></tr>
<tr><td>N</td><td>Se ha calculado el elemento del perfil de levas y está siendo usado actualmente por las instrucciones MAPC (n-2) y MATC.</td></tr>
</table></td></tr>
</table>

Estructura MOTION_GROUP

Hay una estructura MOTION_GROUP por controlador. Esta estructura contiene información de configuración y estado sobre el grupo de movimiento.

<table>
<tr><th>Mnemónico</th><th>Tipo de datos</th><th>Descripción</th></tr>
<tr><td>GroupStatus</td><td>DINT</td><td>Los bits de estado para el grupo.
<table>
<tr><th>bit</th><th>Número</th><th>Tipo de datos</th><th>Descripción</th></tr>
<tr><td>InhibStatus</td><td>00</td><td>DINT</td><td>Estado de inhibición.</td></tr>
<tr><td>GroupSynced</td><td>01</td><td>DINT</td><td>Estado de sincronización.</td></tr>
<tr><td>-no-tag</td><td>02</td><td>DINT</td><td>Temporizador de evento iniciado.</td></tr>
<tr><td>Reservado</td><td>03 – 31</td><td></td><td></td></tr>
</table></td></tr>
<tr><td>MotionFault</td><td>DINT</td><td>Los bits de fallo de movimiento para el grupo.
<table>
<tr><th>bit</th><th>Número</th><th>Tipo de datos</th><th>Descripción</th></tr>
<tr><td>ACAsyncConnFault</td><td>00</td><td>DINT</td><td>Fallo de conexión asíncrona.</td></tr>
<tr><td>ACSyncConnFault</td><td>01</td><td>DINT</td><td>Fallo de conexión síncrona.</td></tr>
<tr><td>Reservado</td><td>02 – 31</td><td></td><td></td></tr>
</table></td></tr>
</table>

Mnemónico	Tipo de datos	Descripción			
ServoFault	DINT	Los bits de fallo del servomódulo para el grupo.			
		Bit	**Número**	**Tipo de datos**	**Descripción**
		POtrvlFault	00	DINT	Fallo de sobrecarrera positiva.
		NOtrvlFault	01	DINT	Fallo de sobrecarrera negativa.
		PosErrorFault	02	DINT	Fallo de error de posición.
		EncCHALossFault	03	DINT	Fallo de pérdida del canal A del encoder.
		EncCHBLossFault	04	DINT	Fallo de pérdida del canal B del encoder.
		EncCHZLossFault	05	DINT	Fallo de pérdida del canal Z del encoder.
		EncNsFault	06	DINT	Fallo de ruido del encoder.
		DriveFault	07	DINT	Fallo del variador.
		Reservado	08 – 31		
		Bit	**Número**	**Tipo de datos**	**Descripción**
		SyncConnFault	00	DINT	Fallo de conexión síncrona.
		HardFault	01	DINT	Fallo de hardware del servo.
		Reservado	02 – 31		
GroupFault	DINT	Los bits de fallo para el grupo.			
		Bit	**Número**	**Tipo de datos**	**Descripción**
		GroupOverlapFault	00	DINT	Fallo de superposición de tareas del grupo.
		CSTLossFault	01	DINT	El controlador ha perdido sincronización con el maestro de CST.
		GroupTaskLoadingFault	02	DINT	El período de actualización aproximado del grupo es demasiado corto, las tareas de aplicación del usuario no tienen suficiente tiempo para ejecutarse.
		Reservado	03 – 31		
AxisFault	DINT	Los bits de fallo para el eje			
		Bit	**Número**	**Tipo de datos**	**Descripción**
		PhysicalAxisFault	00	BOOL	Ha ocurrido un fallo de servo o variador.
		ModuleFault	01	BOOL	Ha ocurrido un fallo grave en el módulo de movimiento asociado con el eje seleccionado. Normalmente afecta a todos los ejes asociados con el módulo de movimiento.
		ConfigFault	02	BOOL	Uno o más atributos de eje asociados con un módulo de movimiento o variador no se han actualizado correctamente para coincidir con el valor del atributo correspondiente del controlador local.
		Reservado	03 – 31		

Tipo de datos MOTION_INSTRUCTION

Debe definir un tag de control de movimiento para cada instrucción de control de movimiento que use. El tag usa el tipo de datos MOTION_INSTRUCTION y almacena información de estado sobre la instrucción.

<table>
<tr><th>Mnemónico</th><th>Tipo de datos</th><th colspan="2">Descripción</th></tr>
<tr><td rowspan="10">INDICADORES</td><td rowspan="10">DINT</td><td colspan="2">Use este DINT para obtener acceso a todos los bits de estado para la instrucción en un valor de 32 bits.</td></tr>
<tr><td>Para este bit de estado</td><td>Use este número de bit</td></tr>
<tr><td>EN</td><td>31</td></tr>
<tr><td>DN</td><td>29</td></tr>
<tr><td>ER</td><td>28</td></tr>
<tr><td>PC</td><td>27</td></tr>
<tr><td>IP</td><td>26</td></tr>
<tr><td>AC</td><td>23</td></tr>
<tr><td>DECEL</td><td>1</td></tr>
<tr><td>ACCEL</td><td>0</td></tr>
<tr><td>EN</td><td>BOOL</td><td colspan="2">El bit Habilitar indica que la instrucción está habilitada (la condición de entrada de renglón y salida de renglón es verdadera).</td></tr>
<tr><td>DN</td><td>BOOL</td><td colspan="2">El bit Listo indica que todos los cálculos y mensajes (si hay alguno) están completos.</td></tr>
<tr><td>ER</td><td>BOOL</td><td colspan="2">El bit Error indica cuando se usa la instrucción de manera no válida.</td></tr>
<tr><td>PC</td><td>BOOL</td><td colspan="2">El bit Proceso Completo indica que la operación se ha completado.
El bit .DN se establece después de que una instrucción ha terminado su ejecución. El bit .PC se establece cuando se completa el proceso iniciado.</td></tr>
<tr><td>IP</td><td>BOOL</td><td colspan="2">El bit En Proceso indica que se está ejecutando un proceso.</td></tr>
<tr><td>AC</td><td>BOOL</td><td colspan="2">El bit Active le permite saber qué instrucción está controlando el movimiento cuando tiene instrucciones en cola. Se establece cuando la instrucción se vuelve activa. Se restablece cuando el bit Proceso Completo se establece o cuando se detiene la instrucción.</td></tr>
<tr><td>ACCEL</td><td>BOOL</td><td colspan="2">El bit .ACCEL indica que la velocidad ha aumentado para la instrucción individual a la que se encuentra asociado, tales como impulso, movimiento o engranaje.</td></tr>
<tr><td>DECEL</td><td>BOOL</td><td colspan="2">El bit .DECEL indica que la velocidad ha disminuido para la instrucción individual a la que se encuentra asociado, tales como impulso, movimiento o engranaje.</td></tr>
<tr><td>ERR</td><td>INT</td><td colspan="2">El valor de error contiene el código de error asociado con una función de movimiento.
Vea Códigos de error (ERR) para las instrucciones de control de movimiento en la página 395.</td></tr>
<tr><td rowspan="7">STATUS</td><td rowspan="7">SINT</td><td colspan="2">El estado de cualquier mensaje asociado con la función de movimiento.</td></tr>
<tr><td>Estado del mensaje</td><td>Descripción</td></tr>
<tr><td>0x0</td><td>El mensaje fue correcto.</td></tr>
<tr><td>0x1</td><td>El módulo esta procesando otro mensaje.</td></tr>
<tr><td>0x2</td><td>El módulo espera una respuesta a un mensaje anterior.</td></tr>
<tr><td>0x3</td><td>La respuesta a un mensaje entró en fallo.</td></tr>
<tr><td>0x4</td><td>El módulo no está listo para enviar mensajes.</td></tr>
<tr><td>STATE</td><td>SINT</td><td colspan="2">El valor de estado de ejecución controla el estado de ejecución de una función. Muchas funciones de movimiento tienen varios pasos y este valor hace un seguimiento de estos pasos. El estado de ejecución siempre se establece en 0 cuando el controlador establece el bit EN para una instrucción de control de movimiento. Otros estados de ejecución dependen de la instrucción de control de movimiento.</td></tr>
</table>

Mnemónico	Tipo de datos	Descripción
SEGMENT	DINT	Un segmento es la distancia que hay entre un punto y el punto siguiente, sin incluir a este último. Un bit SEGMENT indica la posición relativa por número de segmento mientras se ejecuta la leva.
EXERR	SINT	Código de error extendido – se usa para obtener más información acerca de un error.

Estructura OUTPUT_CAM

El tipo de datos OUTPUT_CAM es una matriz que define la información concreta para cada elemento de leva de salida. OUTPUT_CAM contiene los siguientes miembros.

<table>
<tr><th>Mnemónico</th><th>Tipo de datos</th><th colspan="2">Descripción</th></tr>
<tr><td>OutputBit</td><td>DINT</td><td colspan="2">Debe seleccionar un bit de salida dentro del rango de 0 a 31. Una selección menor que 0 o mayor que 31 produce un error Leva de salida no válida y no se considera el elemento de leva.</td></tr>
<tr><td rowspan="6">LatchType</td><td rowspan="6">DINT</td><td colspan="2">El tipo de enclavamiento determina cómo se establece el bit de salida correspondiente. Un valor menor que 0 o mayor que 3 produce un error Leva de salida no válida y se usa un tipo de enclavamiento inactivo.</td></tr>
<tr><td>Valor</td><td>Descripción</td></tr>
<tr><td>0 = Inactivo</td><td>El bit de salida no se modifica.</td></tr>
<tr><td>1 = Position</td><td>El bit de salida se establece cuando el eje introduce el rango de leva compensado.</td></tr>
<tr><td>2 = Enable</td><td>El bit de salida se establece cuando el bit Habilitar se vuelve activo.</td></tr>
<tr><td>3 = Position and Enable</td><td>El bit de salida se establece cuando el eje introduce el rango de leva compensado y el bit Habilitar se vuelve activo.</td></tr>
<tr><td rowspan="8">UnlatchType</td><td rowspan="8">DINT</td><td colspan="2">El tipo de desenclavamiento determina cómo se restablece el bit de salida. Si se seleccionar un valor menor que 0 o mayor que 5, se produce un error Leva de salida no válida y se usa un tipo de desenclavamiento inactivo.</td></tr>
<tr><td>Valor</td><td>Descripción</td></tr>
<tr><td>0 = Inactivo</td><td>El bit de salida no se modifica.</td></tr>
<tr><td>1 = Position</td><td>El bit de salida se restablece cuando el eje se sale del rango de leva compensado.</td></tr>
<tr><td>2 = Duration</td><td>El bit de salida se restablece cuando se termina la duración.</td></tr>
<tr><td>3 = Enable</td><td>El bit de salida se restablece cuando el bit Habilitar se vuelve inactivo.</td></tr>
<tr><td>4 = Position and Enable</td><td>El bit de salida se restablece cuando el eje se sale del rango de leva compensado o el bit Habilitar se vuelve inactivo.</td></tr>
<tr><td>5 = Duration and Enable</td><td>El bit de salida se restablece cuando se termina la duración o el bit Habilitar se vuelve inactivo.</td></tr>
<tr><td>Left</td><td>REAL</td><td colspan="2">La posición de leva izquierda junto con la posición de leva derecha definen el rango de leva del elemento de leva de salida. Las posiciones de leva izquierda y derecha especifican las posiciones de enclavamiento o desenclavamiento del bit de salida cuando el tipo de enclavamiento o desenclavamiento se establece en Position o Position and Enable con el bit Habilitar activo. Si la posición izquierda es menor que la posición de inicio de leva o mayor que la posición final de leva, se produce un error Leva de salida no válida y no se considera el elemento de leva.</td></tr>
</table>

<table>
<tr><th>Mnemónico</th><th>Tipo de datos</th><th colspan="2">Descripción</th></tr>
<tr><td>Right</td><td>REAL</td><td colspan="2">La posición de leva derecha junto con la posición de leva izquierda definen el rango de leva del elemento de leva de salida. Las posiciones de leva derecha e izquierda especifican las posiciones de enclavamiento o desenclavamiento del bit de salida cuando el tipo de enclavamiento o desenclavamiento se establece en Position o Position and Enable con el bit Habilitar activo. Si la posición derecha es menor que la posición de inicio de leva o mayor que la posición final de leva, se produce un error Leva de salida no válida y no se considera el elemento de leva.</td></tr>
<tr><td>Duration</td><td>REAL</td><td colspan="2">La duración especifica el tiempo en segundos entre el enclavamiento y desenclavamiento cuando el tipo de desenclavamiento es Duration o Duration and Enable con el bit Habilitar activo. Un valor menor o igual a 0 produce un error Leva de salida no válida y no se considera el elemento de leva.</td></tr>
<tr><td rowspan="6">EnableType</td><td rowspan="6">DINT</td><td colspan="2">Esto define el origen y polaridad del bit Habilitar especificado cuando los bits LatchType o UnlatchType son Enable, Position and Enable o Duration and Enable. Un valor menor que 0 o mayor que 31 produce un error Leva de salida no válida y no se considera el elemento de leva.</td></tr>
<tr><td>Valor</td><td>Descripción</td></tr>
<tr><td>0 = Entrada</td><td>El bit Habilitar está en el parámetro Input.</td></tr>
<tr><td>1 = Entrada invertida</td><td>El bit Habilitar está en el parámetro Input y tiene actividad baja.</td></tr>
<tr><td>2 = Salida</td><td>El bit Habilitar está en el parámetro Output.</td></tr>
<tr><td>3 = Salida invertida</td><td>El bit Habilitar está en el parámetro Output y tiene actividad baja.</td></tr>
<tr><td>EnableBit</td><td>DINT</td><td colspan="2">El valor del bit Habilitar seleccionado debe ser entre 0 y 31 cuando los bits LatchType o UnlatchType son Habilitar, Posición y Habilitar o Duración y Habilitar. Un valor menor que 0 o mayor que 31 produce un error Leva de salida no válida y no se considera el elemento de leva.</td></tr>
</table>

Estructura OUTPUT_COMPENSATION

El tipo de datos OUTPUT_COMPENSATION define los detalles para cada bit de salida estableciendo las características de cada accionador. OUTPUT_COMPENSATION contiene los siguientes miembros:

Mnemónico	Tipo de datos	Descripción	
Offset	REAL	Offset proporciona compensación de posición para las operaciones de enclavamiento y de desenclavamiento.	
LatchDelay	REAL	El retardo de enclavamiento, programado en segundos, proporciona compensación de tiempo para la operación de enclavamiento.	
UnlatchDelay	REAL	El retardo de desenclavamiento, programado en segundos, proporciona compensación de tiempo para la operación de desenclavamiento.	
Mode	DINT	El modo determina el comportamiento del bit de salida. Se encuentran disponibles las siguientes cuatro opciones de modo. Un valor menor que 0 o mayor que 3 produce un error Compensación de salida no válida.	
		Valor	**Descripción**
		0 = Normal	El bit de salida se establece para la operación de enclavamiento y se restablece para la operación de desenclavamiento.
		1 = Inverted	El bit de salida se restablece para la operación de enclavamiento y se establece para la operación de desenclavamiento.
		2 = Pulsed	El bit de salida se establece para la operación de enclavamiento y para el estado de impulso en servicio y se restablece para la operación de desenclavamiento y para el estado de impulso fuera de servicio.
		3 = Inverted and Pulsed	El bit de salida se restablece para la operación de enclavamiento y para el estado de impulso en servicio y se establece para la operación de desenclavamiento y para el estado de impulso fuera de servicio.
CycleTime	REAL	El tiempo de impulso en segundos. Si el modo es **Pulsed** o **Inverted and Pulsed**, y el bit CycleTime es menor o igual a 0, se produce un error Compensación de salida no válida.	
DutyCycle	REAL	Porcentaje del bit CycleTime en que el impulso se activará (en servicio). Un valor de 50 representa 50% en servicio. Un valor menor que 0 o mayor que 100 produce un error Compensación de salida no válida.	

Notas:

Programación de texto estructurado

Introducción

Este apéndice describe aspectos que son únicos con la programación de texto estructurado. Repase la información proporcionada en este apéndice para asegurarse de que entiende cómo se ejecutará la programación de texto estructurado.

Si desea información acerca de	Vea la página
Sintaxis del texto estructurado	413
Asignaciones	415
Expresiones	417
Instrucciones	424
Construcciones	425
Comentarios	441

Sintaxis del texto estructurado

El texto estructurado es un lenguaje de programación textual que usa instrucciones para definir lo que se va a ejecutar.

- El texto estructurado trata indistintamente las mayúsculas y minúsculas.
- Use fichas y retornos de carro (líneas separadas) para facilitar la lectura del texto estructurado. Estos no tienen ningún efecto en la ejecución del texto estructurado.

El texto estructurado trata indistintamente las mayúsculas y minúsculas. El texto estructurado contiene estos componentes:

Término	Definición	Ejemplos
asignación (vea la página 415)	Use una instrucción de asignación para asignar valores a los tags. El operador es el operador de asignación. Termine la asignación con un signo de punto y coma ";".	tag := expression;

Término	Definición		Ejemplos
Expresión (vea la página 417)	Una expresión es parte de una asignación completa o instrucción de construcción. Una expresión evalúa según un número (expresión numérica) o según un estado de verdadero o falso (expresión BOOL). Una expresión contiene:		
	Tags	Un área con nombre de la memoria, donde se almacenan los datos (BOOL, SINT,INT,DINT, REAL, de cadena).	value1
	Inmediatos	Un valor constante.	4
	Operadores	Un símbolo o mnemónico que especifica una operación dentro de una expresión.	tag1 + tag2 tag1 >= value1
	Funciones	Cuando se ejecuta, una función produce un valor. Use paréntesis para contener el operando de una función. Aunque su sintaxis es similar, las funciones se diferencian de las instrucciones porque las funciones sólo pueden usarse en expresiones. Las instrucciones no pueden usarse en expresiones.	function(tag1)
Instrucción (vea la página 424)	Una instrucción es una instrucción autónoma. Una instrucción usa paréntesis para contener sus operandos. Según la instrucción, pueden haber, cero, uno o múltiples operandos. Cuando se ejecuta, una instrucción produce uno o más valores que son parte de una estructura de datos. Termine la instrucción con un signo de punto y coma “;”. Aunque su sintaxis es similar, las instrucciones se diferencias de las funciones porque las instrucciones no pueden usarse en expresiones. Las funciones sólo pueden usarse en expresiones.		*instruction();* *instruction(operand);* *instruction(operand1, operand2,operand3);*
Construcción (vea la página 425)	Una instrucción condicional usada para activar el código de texto estructurado (por ej., otras instrucciones). Termine la construcción con un signo de punto y coma “;”.		IF...THEN CASE FOR...DO WHILE...DO REPEAT...UNTIL EXIT
Comentario (vea la página 441)	Texto que explica o aclara lo que hace una sección del texto estructurado. · Use comentarios para facilitar la interpretación del texto estructurado. · Los comentarios no tienen ningún efecto en la ejecución del texto estructurado. · Los comentarios pueden aparecer en cualquier lugar del texto estructurado.		//comment (*start of comment . . . end of comment*) /*start of comment . . . end of comment*/

Asignaciones

Use una asignación para cambiar el valor almacenado dentro de un tag. Una asignación tiene esta sintaxis:

tag := *expression* ;

donde:

<table>
<tr><th>Componente</th><th>Descripción</th></tr>
<tr><td><i>tag</i></td><td>representa el tag que obtiene el nuevo valor
el tag debe ser BOOL, SINT, INT, DINT o REAL</td></tr>
<tr><td>:=</td><td>es el símbolo de la asignación</td></tr>
<tr><td><i>expression</i></td><td>representa el nuevo valor que se va a asignar al tag
<table>
<tr><th>Si el <i>tag</i> es este tipo de datos</th><th>Use este tipo de expresión</th></tr>
<tr><td>BOOL</td><td>Expresión BOOL</td></tr>
<tr><td>SINT
INT
DINT
REAL</td><td>expresión numérica</td></tr>
</table></td></tr>
<tr><td>;</td><td>finaliza la asignación</td></tr>
</table>

El *tag* retiene el valor asignado hasta que otra asignación cambia el valor.

La expresión puede ser simple, como un valor inmediato u otro nombre de tag, o la expresión puede ser compleja e incluir varios operadores y/o funciones Vea Expresiones en la página 417 para obtener más detalles.

SUGERENCIA

Los datos del módulo de E/S se actualizan de manera asíncrona a la ejecución de lógica. Si introduce referencias de una entrada múltiples veces en su lógica, la entrada puede cambiar estado entre referencias separadas. Si necesita que la entrada tenga el mismo estado para cada referencia, almacene en búfer el valor de entrada e introduzca la referencia de ese tag de almacenamiento en búfer.

Para obtener más información, vea Logix5000 Controllers Common Procedures, publicación 1756-PM001.

Consulte el índice para obtener más información.

Especifique una asignación no retentiva

La asignación no retentiva es diferente de la asignación regular descrita anteriormente porque el tag en una asignación no retentiva se restablece a cero cada vez que el controlador:

- entra al modo MARCHA
- sale del paso de un SFC si usted configura el SFC para *Restablecimiento automático* (Esto sólo se aplica si usted incorpora la asignación en la acción del paso o usa la acción para llamar a una rutina de texto estructurado mediante una instrucción JSR).

Una asignación no retentiva tiene esta sintaxis:

tag [:=] *expression* ;

Donde:

Componente	Descripción	
tag	representa el tag que obtiene el nuevo valor el tag debe ser BOOL, SINT, INT, DINT o REAL	
[:=]	es el símbolo de asignación no retentiva	
expression	representa el nuevo valor que se va a asignar al tag	
	Si el *tag* es este tipo de datos	**Use este tipo de expresión**
	BOOL	Expresión BOOL
	SINT INT DINT REAL	expresión numérica
;	finaliza la asignación	

Asigne un carácter ASCII a una cadena

Use el operador de asignación para asignar un carácter ASCII a un elemento del miembro DATA de un tag de cadena. Para asignar un carácter, especifique el valor del carácter o especifique el nombre de tag, miembro DATA y elemento del carácter. Por ejemplo:

Esto es correcto	Esto *no* es correcto
string1.DATA[0] := 65;	string1.DATA[0] := A;
string1.DATA[0] := string2.DATA[0];	string1 := string2;

Para añadir o insertar una cadena de caracteres a un tag de cadena, use cualquiera de estas instrucciones de cadena ASCII:

Para	Use esta instrucción
añadir caracteres al final de una cadena	CONCAT
insertar caracteres dentro de una cadena	INSERT

Expresiones

Una expresión es un nombre de tag, ecuación o comparación. Para escribir una expresión, use cualquiera de estos elementos:

- nombre de tag que almacene el valor (variable)
- número que usted introduce directamente en una expresión (valor inmediato)
- funciones, tales como: ABS, TRUNC
- operadores, tales como: +, –, <, >, And, Or

Cuando escriba las expresiones, siga estas reglas generales:

- Use cualquier combinación de mayúsculas y minúsculas. Por ejemplo, estas tres variaciones de “AND” son aceptables: AND, And, and.

- Para requisitos más complejos, use paréntesis para agrupar expresiones dentro de expresiones. Esto hace que toda la expresión sea más fácil de leer y asegura que la expresión se ejecute en la secuencia deseada. Consulte “Determine el orden de ejecución” en la página 423.

En texto estructurado usted usa dos tipos de expresiones:

Expresión BOOL: Una expresión que produce ya sea el valor BOOL de 1 (verdadero) o 0 (falso).

- Una expresión Bool usa tags bool, operadores con relaciones y operadores lógicos para comparar valores o verificar si las condiciones son verdaderas o falsas. Por ejemplo, tag1>65.
- Una expresión Bool simple puede ser un tag BOOL único.
- Normalmente usted usa expresiones para condicionar la ejecución de otra lógica.

Expresión numérica: Una expresión que calcula un valor entero o de coma flotante (punto flotante).

- Una expresión numérica usa operadores aritméticos, funciones aritméticas y operadores a nivel de bit. Por ejemplo, tag1+5.
- A menudo usted anida una expresión numérica dentro de una expresión Bool. Por ejemplo, (tag1+5)>65.

Use la siguiente tabla para seleccionar operadores para sus expresiones:

Si desea	**Entonces**
Calcular un valor aritmético	“Use operadores y funciones aritméticas“en la página 419.
Comparar dos valores o cadenas	“Use operadores con relaciones“en la página 420.
Verificar si las condiciones son verdaderas o falsas	“Use operadores lógicos“en la página 422.
Comparar los bits dentro de valores	“Use operadores bit a bit“en la página 423.

Use operadores y funciones aritméticas

Usted puede combinar múltiples operadores y funciones en expresiones aritméticas.

Los operadores aritméticos calculan nuevos valores.

Para	Use este operador	Tipo de datos óptimo
añadir	+	DINT, REAL
restar/cambiar signo	–	DINT, REAL
Multiplicar	*	DINT, REAL
exponente (x a la potencia de y)	**	DINT, REAL
Dividir	/	DINT, REAL
módulo-dividir	MOD	DINT, REAL

Las funciones aritméticas realizan operaciones matemáticas. Especifique una constante, un tag no booleano o una expresión para la función.

Para obtener	Use esta función	Tipo de datos óptimo
valor absoluto	ABS (*numeric_expression*)	DINT, REAL
arco coseno	ACOS (*numeric_expression*)	REAL
arco seno	ASIN (*numeric_expression*)	REAL
arco tangente	ATAN (*numeric_expression*)	REAL
coseno	COS (*numeric_expression*)	REAL
radianes a grados	DEG (*numeric_expression)*	DINT, REAL
logaritmo natural	LN (*numeric_expression*)	REAL
logaritmo base 10	LOG (*numeric_expression*)	REAL
grados a radianes	RAD (*numeric_expression*)	DINT, REAL
seno	SIN (*numeric_expression*)	REAL
raíz cuadrada	SQRT (*numeric_expression*)	DINT, REAL
tangente	TAN (*numeric_expression*)	REAL
truncar	TRUNC (*numeric_expression*)	DINT, REAL

Por ejemplo:

Use este formato	Ejemplo	
	Para esta situación	**Usted escribiría**
value1 operator value2	Si gain_4 y gain_4_adj son tags DINT y su especificación dice: "Añadir 15 a gain_4 y almacenar el resultado en gain_4_adj."	gain_4_adj := gain_4+15;
operator value1	Si alarm y high_alarm son valores DINT y su especificación dice: "Cambiar signo de high_alarm y almacenar el resultado en alarm."	alarm:= -high_alarm;
function(numeric_expression)	Si overtravel y overtravel_POS son tags DINT y su especificación dice: "Calcular el valor absoluto de overtravel y almacenar el resultado en overtravel_POS."	overtravel_POS := ABS(overtravel);
value1 operator (function((value2+value3)/2)	Si adjustment y position son tags DINT y sensor1 y sensor2 son tags REAL y su especificación dice: "Encontrar el valor absoluto del sensor1 y sensor2 promedio, añadir el ajuste y guardar el resultado en position."	position := adjustment + ABS((sensor1 + sensor2)/2);

Use operadores con relaciones

Los operadores con relaciones comparan dos valores o cadenas para proporcionar un resultado verdadero o falso. El resultado de una operación de relación es un valor BOOL:

Si la comparación es	El resultado es
verdadera	1
falsa	0

Use estos operadores de relación:

Para esta comparación:	Use este operador	Tipo de datos óptimo
igual que	=	DINT, REAL, cadena
menor que	<	DINT, REAL, cadena
menor que o igual que	<=	DINT, REAL, cadena
mayor que	>	DINT, REAL, cadena
mayor que o igual que	>=	DINT, REAL, cadena
diferente	<>	DINT, REAL, cadena

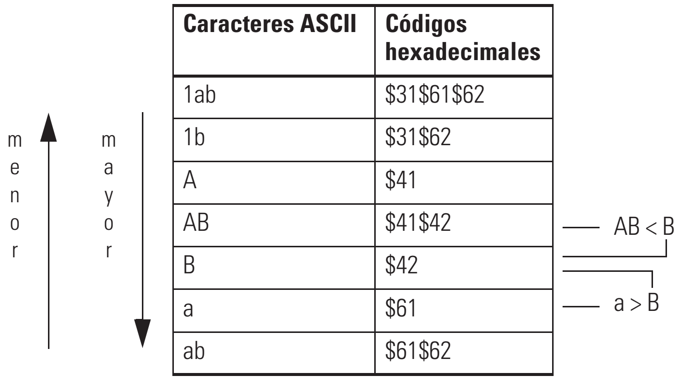
Por ejemplo:

Use este formato	**Ejemplo**	
	Para esta situación	**Usted escribiría**
value1 operator value2	Si temp es un tag DINT y su especificación dice: "Si temp es menor que 100· entonces…"	SI temp<100 ENTONCES...
stringtag1 operator stringtag2	Si bar_code y dest son tags de cadena y su especificación dice: "Si bar_code es igual que dest entonces…"	IF bar_code=dest THEN...
char1 operator char2 Para introducir un carácter ASCII directamente en la expresión, introduzca el valor decimal del carácter.	Si bar_code es un tag de cadena y su especificación dice: "Si bar_code.DATA[0] es igual que 'A' entonces…"	IF bar_code.DATA[0]=65 THEN...
bool_tag := *bool_expressions*	Si count y length son tags DINT, done es un tag BOOL y su especificación dice "Si count es mayor o igual que length, usted ya terminó el conteo."	done := (count >= length);

Cómo se evalúan las cadenas

Los valores hexadecimales de los caracteres ASCII determinan si una cadena es menor o mayor que otra cadena.

- Cuando las dos cadenas se clasifican como en un directorio telefónico, el orden de las cadenas determina cuál es mayor.



Caracteres ASCII	**Códigos hexadecimales**
1ab	$31$61$62
1b	$31$62
A	$41
AB	$41$42
B	$42
a	$61
ab	$61$62

- Las cadenas son iguales si sus caracteres coinciden.
- Los caracteres distinguen entre minúsculas y mayúsculas. La "A" mayúscula ($41) *no* es igual a la "a" minúscula ($61)

Use operadores lógicos

Los operadores lógicos le permiten verificar si múltiples condiciones son verdaderas o falsas. El resultado de una operación lógica es un valor BOOL.

Si la comparación es	El resultado es
verdadera	1
falsa	0

Use estos operadores lógicos:

Para obtener	Use este operador	Tipo de datos
Y lógico	&, AND	BOOL
O lógico	OR	BOOL
O exclusivo lógico	XOR	BOOL
complemento lógico	NOT	BOOL

Por ejemplo:

Use este formato	Ejemplo	
	Para esta situación	**Usted escribiría**
BOOLtag	Si photoeye es un tag BOOL y su especificación dice: “Si photoeye_1 está activado entonces…”	IF photoeye THEN...
NOT *BOOLtag*	Si photoeye es un tag BOOL y su especificación dice: “Si photoeye está desactivada entonces…”	IF NOT photoeye THEN...
expression1 & *expression2*	Si photoeye es un tag BOOL, temp es un tag DIN y su especificación dice: “Si photoeye está activada y temp es menor que 100· entonces…”.	IF photoeye & (temp<100) THEN...
expression1 OR *expression2*	Si photoeye es un tag BOOL, temp es un tag DIN y su especificación dice: “Si photoeye está activada o temp es menor que 100· entonces…”.	IF photoeye O (temp<100) THEN...
expression1 XOR *expression2*	Si photoeye1 y photoeye2 son tags BOOL y su especificación dice: “Si: · photoeye1 está activada mientras que photoeye2 está desactivada o · photoeye1 está desactivada mientras que photoeye2 está activada entonces…”	IF photoeye1 XOR photoeye2 THEN...
BOOLtag := *expression1* & *expression2*	Si photoeye1 y photoeye2 son tags BOOL,abrir es un tag BOOL y su especificación dice: “Si photoeye1 y photoeye2 están activadas, establezca abrir en verdadero”.	abrir := photoeye1 y photoeye2;

Use operadores bit a bit

Los operadores bit a bit manipulan los bits dentro de un valor basado en dos valores.

Para obtener	Use este operador	Tipo de datos óptimo
función AND	&, AND	DINT
función O	OR	DINT
función O exclusivo	XOR	DINT
complemento de función	NOT	DINT

Por ejemplo:

Use este formato	Ejemplo	
	Para esta situación	**Usted escribiría**
value1 operator value2	Si input1, input2, y result1 son tags DINT y su especificación dice: "Calcule el resultado de la función de input1 e input2. Guarde el resultado en result1."	result1 := input1 AND input2;

Determine el orden de ejecución

Las operaciones que usted escribe en una expresión se realizan en un orden prescrito, no necesariamente de izquierda a derecha.

- Las operaciones de igual orden se realizan de izquierda a derecha.
- Si una expresión contiene múltiples operadores o funciones, agrupe las condiciones entre paréntesis "( )". Esto asegura el orden correcto de ejecución y facilita la lectura de la expresión.

Orden	Operación
1.	()
2.	función (…)
3.	**
4.	– (cambiar signo)
5.	NOT
6.	*, /, MOD
7.	+, – (restar)
8.	<, <=, >, >=
9.	=, <>
10.	&, AND
11.	XOR
12.	OR

Instrucciones

Los comandos de texto estructurado también pueden ser instrucciones. Vea la tabla de ubicación al comienzo de este manual para obtener una lista de las instrucciones disponibles en texto estructurado. Una instrucción de texto estructurado se ejecuta cada vez que se escanea. Una instrucción de texto estructurado dentro de una construcción se ejecuta cada vez que las condiciones de la construcción son verdaderas. Si las condiciones de la construcción son falsas, las instrucciones dentro de la construcción no se escanean. No existe una condición de renglón o transición de estado que active la ejecución.

Esto es diferente de las instrucción de bloque de función que usan EnableIn para activar la ejecución. Las instrucciones de texto estructurado se ejecutan como si EnableIn siempre estuviera establecido.

Esto también es diferente de las instrucciones de lógica de escalera que usan condición de entrada de renglón para activar la ejecución. Algunas instrucciones de lógica de escalera de relés sólo se ejecutan cuando la condición de entrada de renglón cambia de falso a verdadero. Estas son instrucciones de lógica de escalera de relés transicionales. En texto estructurado, las instrucciones se ejecutarán cada vez que sean escaneadas, a menos que usted condicione previamente la ejecución de la instrucción de texto estructurado.

Por ejemplo, la instrucción ABL es una instrucción transicional en lógica de escalera de relés. En este ejemplo, la instrucción ABL sólo se ejecuta en un escán cuando tag_xic cambia de restablecido a establecido. La instrucción ABL no se ejecuta cuando tag_xic permanece establecido ni cuando tag_xic se restablece.

En texto estructurado, su usted escribe este ejemplo como:

```
IF tag_xic THEN ABL(0,serial_control);

END_IF;
```

la instrucción ABL se ejecutará en cada escán que tag_xic esté establecido, no sólo cuando tag_xic cambie de restablecido a establecido.

Si desea que la instrucción ABL se ejecute sólo cuando tag_xic cambia de restablecido a establecido, tiene que condicionar la instrucción de texto estructurado. Use un impulso para activar la ejecución.

```
osri_1.InputBit := tag_xic;
OSRI(osri_1);

IF (osri_1.OutputBit) THEN
        ABL(0,serial_control);
END_IF;
```

Construcciones

Las construcciones pueden programarse individualmente o anidadas dentro de otras construcciones.

Si desea	Use esta construcción	Disponible en estos lenguajes	Vea la página
hacer algo si o cuando ocurra una condición específica	IF...THEN	texto estructurado	426
seleccionar qué hacer en base a un valor numérico	CASE...OF	texto estructurado	429
hacer algo un número específico de veces antes de hacer otra cosa	FOR...DO	texto estructurado	432
continuar haciendo algo siempre y cuando ciertas condiciones sean verdaderas	WHILE...DO	texto estructurado	435
seguir haciendo algo hasta que una condición sea verdadera	REPEAT...UNTIL	texto estructurado	438

Algunas palabras clave están reservadas para uso futuro

Estas construcciones no están disponibles:

- GOTO
- REPEAT

El software RSLogix 5000 no permitirá que los use.

IF...THEN

Use IF…THEN (SI...ENTONCES) para hacer algo si o cuando ocurra una condición específica.

Operandos:

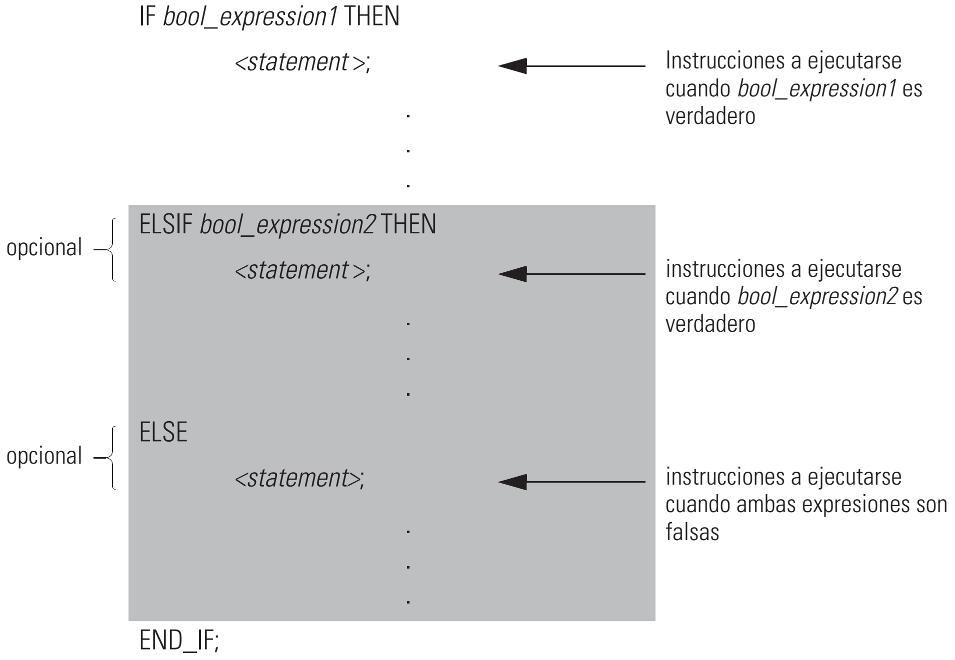
```
IF bool_expression THEN

<statement>;

END_IF;
```

Texto estructurado

Operando	Tipo	Formato	Introduzca
bool_ expression	BOOL	tag expresión	Expresión o tag BOOL que evalúa respecto a un valor BOOL (expresión BOOL)

Descripción:

La sintaxis es:



```
IF bool_expression1 THEN
          <statement >;
                                  .
                                  .
                                  .
ELSIF bool_expression2 THEN
          <statement >;
                                  .
                                  .
                                  .
ELSE
          <statement>;
                                  .
                                  .
                                  .
END_IF;
```

Para usar ELSIF o ELSE, siga estas pautas:

1. Para seleccionar entre varios posibles grupos de instrucciones, añada una o más instrucciones ELSIF.
    - Cada instrucción ELSIF representa una ruta alternativa.
    - Especifique todas las rutas ELSIF que necesite.
    - El controlador ejecuta la primera instrucción IF o ELSIF verdadera y salta el resto de las instrucciones ELSIFs y ELSE.
2. Para hacer algo cuando todas las condiciones IF o ELSIF son falsas, añada una instrucción ELSE.

Esta tabla resume las diferentes combinaciones de IF, THEN, ELSIF y ELSE.

Si desea	Y	Entonces use esta construcción
hacer algo si o cuando las condiciones son verdaderas	no hacer nada si las condiciones son falsas	IF…THEN
	hacer algo más si las condiciones son falsas	IF…THEN…ESLE
seleccionar entre instrucciones alternativas (o grupos de instrucciones) en base a condiciones de entrada	no hacer nada si las condiciones son falsas	IF…THEN…ELSIF
	asignar instrucciones predeterminadas si todas las condiciones son falsas	IF…THEN…ELSIF…ELSE

Indicadores de estado aritméticos: no afectados

Condiciones de fallo: ninguno

Ejemplo 1: **IF…THEN**

Si usted desea esto	Introduzca este texto estructurado
Si rechaza > 3 entonces	IF rejects > 3 THEN
transportador = desactivado (0)	conveyor := 0;
alarma = activada (1)	alarm:= 1;
	END_IF;

Ejemplo 2: **IF…THEN…ELSE**

Si usted desea esto	Introduzca este texto estructurado
Si el contacto de dirección de transportador = avance (1) entonces	IF conveyor_direction THEN
luz = apagada	light := 0;
De lo contrario luz = encendida	ELSE
	light [:=] 1;
	END_IF;

El [:=] le indica al controlador que restablezca light cada vez que el controlador:

- entra al modo MARCHA
- sale del paso de un SFC si usted configura el SFC para Restablecimiento automático (Esto sólo se aplica si usted incorpora la asignación en la acción del paso o usa la acción para llamar a una rutina de texto estructurado mediante una instrucción JSR).

Ejemplo 3: IF…THEN…ELSIF

Si usted desea esto	Introduzca este texto estructurado
Si el interruptor de final de carrera de azúcar baja = baja (activado) y el interruptor de final de carrera de azúcar alta = no alta (activado) entonces	IF Sugar.Low & Sugar.High THEN
válvula de entrada = abierta (activada)	Sugar.Inlet [:=] 1;
Hasta que el interruptor de final de carrera de azúcar alta = alta (desactivado)	ELSIF NOT(Sugar.High) THEN
	Sugar.Inlet := 0;
	END_IF;

El [:=] le indica al controlador que restablezca Sugar.Inlet cada vez que el controlador:

- entra al modo MARCHA
- sale del paso de un SFC si usted configura el SFC para Restablecimiento automático (Esto sólo se aplica si usted incorpora la asignación en la acción del paso o usa la acción para llamar a una rutina de texto estructurado mediante una instrucción JSR).

Ejemplo 4: IF…THEN…ELSIF…ELSE

Si usted desea esto	Introduzca este texto estructurado
Si la temperatura del tanque > 100	IF tank.temp > 200 THEN
entonces bomba = lenta	pump.fast :=1; pump.slow :=0; pump.off :=0;
Si la temperatura del tanque > 200	ELSIF tank.temp > 100 THEN
entonces bomba = rápida	pump.fast :=0; pump.slow :=1; pump.off :=0;
de lo contrario bomba = desactivada	ELSE
	pump.fast :=0; pump.slow :=0; pump.off :=1;
	END_IF;

CASE...OF

Use CASE para seleccionar qué hacer en base a un valor numérico.

Operandos:

```
CASE numeric_expression OF

selector1: statement;

selectorN: statement;

ELSE

statement;

END_CASE;
```

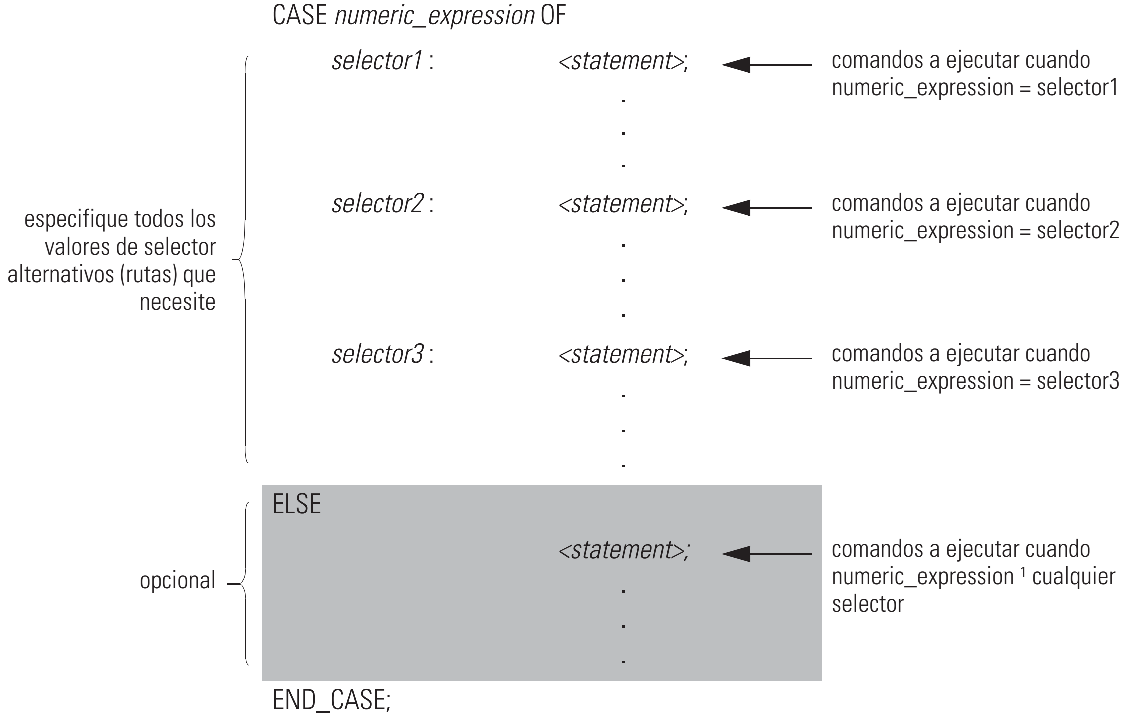
Texto estructurado

Operando	Tipo	Formato	Introduzca
numeric_ expression	SINT INT DINT REAL	tag expresión	tag o expresión que evalúa a un número (expresión numérica)
selector	SINT INT DINT REAL	Inmediato	el mismo tipo que *numeric_expression*

IMPORTANTE Si usted usa valores REAL, use un rango de valores para un selector porque un valor REAL es más probable que esté dentro de un rango de valores que una coincidencia exacta de un valor específico.

Descripción:

La sintaxis es:



```
CASE numeric_expression OF
     selector1 :          <statement>;
                               .
                               .
                               .
     selector2 :          <statement>;
                               .
                               .
                               .
     selector3 :          <statement>;
                               .
                               .
                               .
ELSE
                          <statement>;
                               .
                               .
                               .
END_CASE;
```

Vea la tabla en la siguiente página para obtener los valores de selector válidos.

La sintaxis para introducir los valores de selector es:

Cuando el selector es	Introduzca
un valor	value: statement
múltiples y diferentes valores	value1, value2, valueN : <statement> Use una coma (,) para separar cada valor.
un rango de valores	value1...valueN : <statement> Use dos puntos (..) para identificar el rango.
valores distintos más un rango de valores	valuea, valueb, value1...valueN : <statement>

La construcción CASE es similar a una instrucción de interruptor en los lenguajes de programación C o C++. Sin embargo, con la construcción CASE el controlador ejecuta sólo las instrucciones asociadas con el primer valor de selector coincidente. La ejecución siempre se interrumpe después de las instrucciones de dicho selector y va a la instrucción END_CASE.

Indicadores de estado aritméticos: no afectados

Condiciones de fallo: ninguno

Ejemplo

Si usted desea esto	Introduzca este texto estructurado
Si el número de receta = 1 entonces	CASE recipe_number OF
Ingrediente A salida 1 = abierto (1)	1: Ingredient_A.Outlet_1 :=1;
Ingrediente B salida 4 = abierto (1)	Ingredient_B.Outlet_4 :=1;
Si el número de receta = 2 ó 3 entonces	2,3: Ingredient_A.Outlet_4 :=1;
Ingrediente A salida 4 = abierto (1)	Ingredient_B.Outlet_2 :=1;
Ingrediente B salida 2 = abierto (1)	
Si el número de receta = 4, 5, 6 ó 7 entonces	4..7: Ingredient_A.Outlet_4 :=1;
Ingrediente A salida 4 = abierto (1)	Ingredient_B.Outlet_2 :=1;
Ingrediente B salida 2 = abierto (1)	
Si el número de receta = 8, 11, 12 ó 13 entonces	8,11..13 Ingredient_A.Outlet_1 :=1;
Ingrediente A salida 1 = abierto (1)	Ingredient_B.Outlet_4 :=1;
Ingrediente B salida 4 = abierto (1)	
De lo contrario, todas las salidas = cerradas (0)	ELSE
	Ingredient_A.Outlet_1 [:=]0;
	Ingredient_A.Outlet_4 [:=]0;
	Ingredient_B.Outlet_2 [:=]0;
	Ingredient_B.Outlet_4 [:=]0;
	END_CASE;

El [:=] le indica al controlador que también restablezca los tags de salida cada vez que el controlador:

- entra al modo MARCHA
- sale del paso de un SFC si usted configura el SFC para Restablecimiento automático (Esto sólo se aplica si usted incorpora la asignación en la acción del paso o usa la acción para llamar a una rutina de texto estructurado mediante una instrucción JSR).

FOR…DO

Use el lazo FOR…DO para hacer algo un número específico de veces antes de hacer otra cosa.

Operandos:

```
FOR count:=initial_value TO final_value BY
increment DO

<statement>;

END_FOR;
```

Texto estructurado

Operando	Tipo	Formato	Descripción
count	SINT INT DINT	tag	tag para almacenar la posición de conteo a medida que FOR…DO se ejecuta
initial_ value	SINT INT DINT	tag expresión Inmediato	debe evaluar con respecto a un número especifica el valor inicial para el conteo
final_ value	SINT INT DINT	tag expresión Inmediato	especifica el valor final para el conteo, el cual determina cuándo salir del lazo
increment	SINT INT DINT	tag expresión Inmediato	*(opcional)* Cantidad a incrementar el conteo cada vez a través del lazo Si usted no especifica un incremento, el conteo se incrementa por un valor de 1.

IMPORTANTE Asegúrese de no efectuar iteraciones dentro de un lazo demasiadas veces en un solo escán.

- El controlador no ejecuta ninguna otra instrucción en la rutina hasta que complete el lazo.
- Si el tiempo que se requiere para completar el lazo es mayor que el temporizador de control (watchdog) para la tarea, se produce un fallo mayor.
- Considere usar una construcción diferente, por ejemplo IF...THEN.

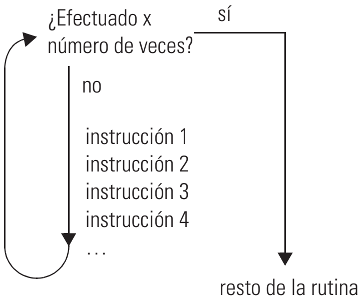
Descripción: La sintaxis es:

```
          FOR count := initial_value
             TO final_value
opcional {   BY increment

             DO
             <statement>;
          ⎧  IF bool_expression THEN
opcional ⎨       EXIT;
          ⎩  END_IF;

          END_FOR;
```

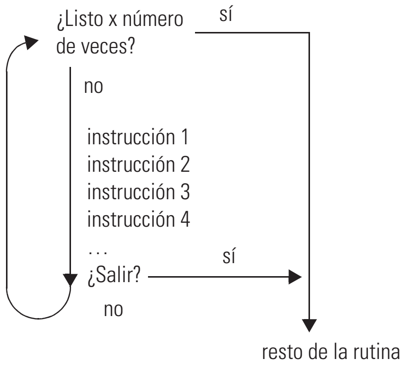
Si usted no especifica un incremento, el lazo se incrementa por un valor de 1.

Si existen condiciones cuando usted desea salir del lazo anticipadamente, use otras instrucciones, tal como la construcción IF...THEN, para condicionar una instrucción EXIT.

Estos diagramas muestran cómo se ejecuta un lazo FOR...DO y cómo una instrucción EXIT deja el lazo anticipadamente.



El lazo FOR…DO se ejecuta un número específico de veces.



Para detener el lazo antes de que el conteo llegue al último valor, use una instrucción EXIT.

Indicadores de estado aritméticos:

no afectados

Condiciones de fallo:

Ocurrirá un fallo mayor si	Tipo de fallo	Código de fallo
la construcción tiene un lazo excesivamente largo	6	1

Ejemplo 1:

Si usted desea esto	Introduzca este texto estructurado

Restablecer los bits 0 – 31 en una matriz de BOOLs:
1. Inicializar el tag subscript a 0.
2. Restablecer la matriz[subscript]. Por ejemplo, cuando subscript = 5, restablecer array[5].
3. Añadir 1 a subscript.
4. Si subscript es £ con respecto a 31, repita 2 y 3.
De lo contrario, parar.
	For subscript:=0 to 31 by 1 do
	array[subscript] := 0;
	END_FOR;

Ejemplo 2:

Si usted desea esto

Un tipo de datos definido por el usuario (estructura) almacena esta información acerca de un ítem en su inventario:

- ID de código de barras del ítem (tipo de datos de cadena)
- Cantidad en inventario del ítem (tipo de datos DINT)

Una matriz de la estructura anterior contiene un elemento para cada ítem diferente en su inventario. Usted desea buscar la matriz para un producto específico (use su código de barras) y determine la cantidad que hay en inventario.

1. Obtenga el tamaño (número de ítems) de la matriz Inventory y almacene el resultado en Inventory_Items (tag DINT).
2. Inicializar el tag position a 0.
3. Si el código de barras coincide con el ID de un ítem en la matriz, entonces:

   a. Establecer el tag Quantity = Inventory[position].Qty. Esto produce la cantidad en inventario del ítem.

   b. Pare.

   Barcode es una cadena que almacena el código de barras del ítem que usted está buscando. Por ejemplo, cuando position = 5, compare Barcode con Inventory[5].ID.
4. Añadir 1 a position.
5. Si position es £ con respecto a (Inventory_Items –1), repita 3 y 4. Puesto que los números de elemento comienzan en 0, el último elemento es 1 menos el número de elementos en la matriz.

   De lo contrario, parar.

Introduzca este texto estructurado

```
SIZE(Inventory,0,Inventory_Items);
For position:=0 to Inventory_Items –1 do
    If Barcode = Inventory[position].ID then
        Quantity := Inventory[position].Qty;
        EXIT;
    END_IF;
END_FOR;
```

WHILE…DO

Use el lazo WHILE…DO para continuar haciendo algo siempre y cuando ciertas condiciones sean verdaderas.

Operandos:

```
WHILE bool_expression DO

<statement>;

END_WHILE;
```

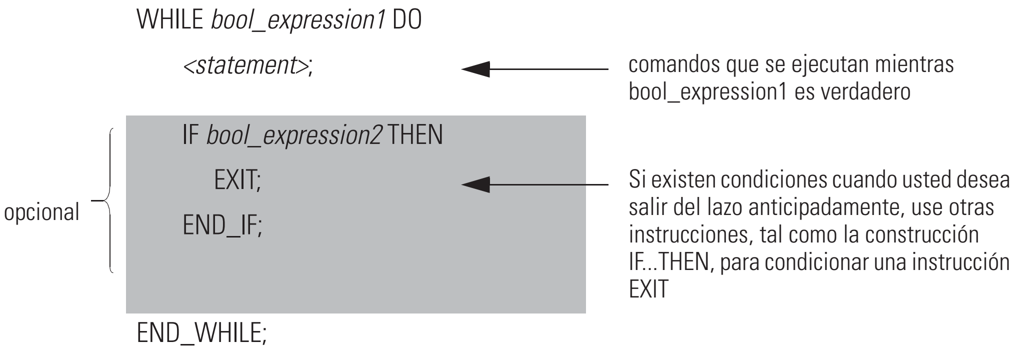
Texto estructurado

Operando	Tipo	Formato	Introduzca
bool_ expression	BOOL	tag expresión	Expresión o tag BOOL que evalúa respecto a un valor BOOL

IMPORTANTE Asegúrese de no efectuar iteraciones dentro de un lazo demasiadas veces en un solo escán.

- El controlador no ejecuta ninguna otra instrucción en la rutina hasta que complete el lazo.
- Si el tiempo que se requiere para completar el lazo es mayor que el temporizador de control (watchdog) para la tarea, se produce un fallo mayor.
- Considere usar una construcción diferente, por ejemplo IF...THEN.

Descripción: La sintaxis es:

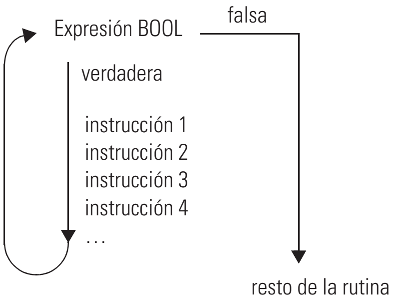


```
WHILE bool_expression1 DO
    <statement>;

    IF bool_expression2 THEN
        EXIT;
    END_IF;

END_WHILE;
```

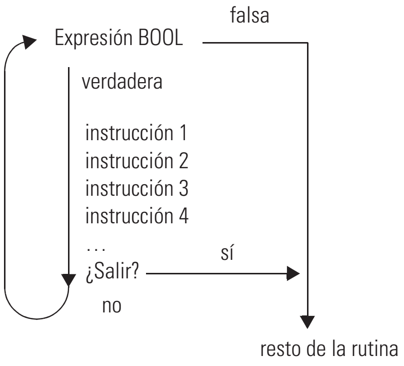
comandos que se ejecutan mientras bool_expression1 es verdadero

Si existen condiciones cuando usted desea salir del lazo anticipadamente, use otras instrucciones, tal como la construcción IF...THEN, para condicionar una instrucción EXIT

Estos diagramas muestran cómo se ejecuta un lazo WHILE...DO y cómo una instrucción EXIT deja el lazo anticipadamente.



Mientras *bool_expression* es verdadero, el controlador ejecuta sólo las instrucciones dentro del lazo WHILE…DO.



Para detener el lazo antes de que las condiciones sean verdaderas, use una instrucción EXIT.

Indicadores de estado aritméticos:

no afectados

Condiciones de fallo:

Ocurrirá un fallo mayor si	Tipo de fallo	Código de fallo
la construcción tiene un lazo excesivamente largo	6	1

Ejemplo 1:

Si usted desea esto

El lazo WHILE...DO evalúa sus condiciones primero. Si las condiciones son verdaderas, entonces el controlador ejecuta las instrucciones dentro del lazo.

Esto es diferente del lazo REPEAT...UNTIL porque el lazo REPEAT...UNTIL ejecuta las instrucciones en la construcción y luego determina si las condiciones son verdaderas antes de ejecutar las instrucciones nuevamente. Las instrucciones en un lazo REPEAT...UNTIL siempre se ejecutan por lo menos una vez. Las instrucciones en un lazo WHILE...DO es posible que nunca se ejecuten.

Introduzca este texto estructurado

```
pos := 0;
While ((pos <= 100) & structarray[pos].value <> targetvalue)) do
    pos := pos + 2;
    String_tag.DATA[pos] := SINT_array[pos];
END_WHILE;
```

Ejemplo 2:

Si usted desea esto

Mover caracteres ASCII desde una matriz SINT a un tag de cadena. (En una matriz SINT, cada elemento retiene un carácter). Pare cuando llegue al retorno de carro.

1. Inicializar Element_number a 0.
2. Cuente el número de elementos en SINT_array (la matriz que contiene los caracteres ASCII) y almacene el resultado en SINT_array_size (tag DINT).
3. Si el carácter en SINT_array[element_number] = 13 (valor decimal del retorno de carro), entonces pare.
4. Establezca String_tag[element_number] = el carácter en SINT_array[element_number].
5. Añadir 1 a element_number. Esto permite al controlador verificar el siguiente carácter en SINT_array.
6. Establezca el miembro Length de String_tag = element_number. (Así registra el número de caracteres que hay hasta el momento en String_tag).
7. Si element_number = SINT_array_size, entonces pare. (Usted está al final de la matriz y ésta no contiene un retorno de carro).
8. Vaya a 3.

Introduzca este texto estructurado

```
element_number := 0;
SIZE(SINT_array, 0, SINT_array_size);
While SINT_array[element_number] <> 13 do
    String_tag.DATA[element_number] :=
    SINT_array[element_number];
    element_number := element_number + 1;
    String_tag.LEN := element_number;
    If element_number = SINT_array_size then
        EXIT;
    END_IF;
END_WHILE;
```

REPEAT…UNTIL

Use el lazo REPEAT…UNTIL para continuar haciendo algo hasta que las condiciones sean verdaderas.

```
REPEAT
    <statement>;
UNTIL bool_expression
END_REPEAT;
```

Operandos:

Texto estructurado

Operando	Tipo	Formato	Introduzca
bool_ expression	BOOL	tag expresión	Expresión o tag BOOL que evalúa respecto a un valor BOOL (expresión BOOL)

IMPORTANTE Asegúrese de no efectuar iteraciones dentro de un lazo demasiadas veces en un solo escán.

- El controlador no ejecuta ninguna otra instrucción en la rutina hasta que complete el lazo.
- Si el tiempo que se requiere para completar el lazo es mayor que el temporizador de control (watchdog) para la tarea, se produce un fallo mayor.
- Considere usar una construcción diferente, por ejemplo IF...THEN.

Descripción: La sintaxis es:

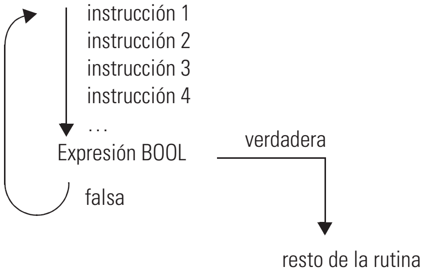
```
REPEAT
    <statement>;
    IF bool_expression2 THEN
        EXIT;
    END_IF;
UNTIL bool_expression1
END_REPEAT;
```

<statement>; ← comandos que se ejecutan mientras bool_expression1 es falso.

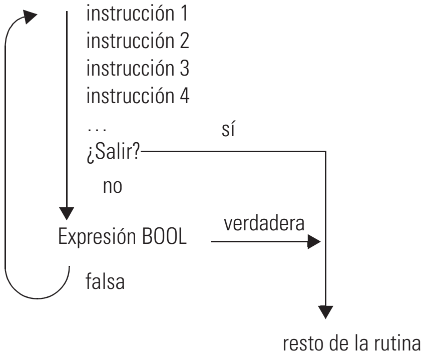
opcional: IF bool_expression2 THEN / EXIT; / END_IF;

EXIT; ← Si existen condiciones cuando usted desea salir del lazo anticipadamente, use otras instrucciones, tal como la construcción IF...THEN, para condicionar una instrucción EXIT.

Estos diagramas muestran cómo se ejecuta un lazo REPEAT...UNTIL y cómo una instrucción EXIT deja el lazo anticipadamente.



Mientras *bool_expression* es falso, el controlador ejecuta sólo las instrucciones dentro del lazo REPEAT…UNTIL.



Para detener el lazo antes de que las condiciones sean falsas, use una instrucción EXIT.

Indicadores de estado aritméticos:

no afectados

Condiciones de fallo:

Ocurrirá un fallo mayor si	Tipo de fallo	Código de fallo
la construcción tiene un lazo excesivamente largo	6	1

Ejemplo 1:

Si usted desea esto	Introduzca este texto estructurado

El lazo REPEAT...UNTIL ejecuta las instrucciones en la construcción y luego determina si las condiciones son verdaderas antes de ejecutar las instrucciones nuevamente.
Esto es diferente del lazo WHILE...DO porque el lazo WHILE...DO evalúa su condición primero. Si las condiciones son verdaderas, entonces el controlador ejecuta las instrucciones dentro del lazo. Las instrucciones en un lazo REPEAT...UNTIL siempre se ejecutan por lo menos una vez. Las instrucciones en un lazo WHILE...DO es posible que nunca se ejecuten.
	pos := –1;
	REPEAT
	pos := pos + 2;
	UNTIL ((pos = 101) OR (structarray[pos].value = targetvalue))
	END_REPEAT;

Ejemplo 2:

Si usted desea esto	Introduzca este texto estructurado
Mover caracteres ASCII desde una matriz SINT a un tag de cadena. (En una matriz SINT, cada elemento retiene un carácter). Pare cuando llegue al retorno de carro. 1. Inicializar Element_number a 0. 2. Cuente el número de elementos en SINT_array (la matriz que contiene los caracteres ASCII) y almacene el resultado en SINT_array_size (tag DINT). 3. Establezca String_tag[element_number] = el carácter en SINT_array[element_number]. 4. Añadir 1 a element_number. Esto permite al controlador verificar el siguiente carácter en SINT_array. 5. Establezca el miembro Length de String_tag = element_number. (Así registra el número de caracteres que hay hasta el momento en String_tag). 6. Si element_number = SINT_array_size, entonces pare. (Usted está al final de la matriz y ésta no contiene un retorno de carro). 7. Si el carácter en SINT_array[element_number] = 13 (valor decimal del retorno de carro), entonces pare. De lo contrario vaya a 3.	

```
element_number := 0;
SIZE(SINT_array, 0, SINT_array_size);
REPEAT
    String_tag.DATA[element_number] :=
    SINT_array[element_number];
    element_number := element_number + 1;
    String_tag.LEN := element_number;
    If element_number = SINT_array_size then
        EXIT;
    END_IF;
Until SINT_array[element_number] = 13
END_REPEAT;
```

Comentarios

Para facilitar la interpretación del texto estructurado, añádale comentarios.

- Los comentarios le permiten usar lenguaje común para describir cómo funciona su texto estructurado.
- Los comentarios no tienen ningún efecto en la ejecución del texto estructurado.

Los comentarios de texto estructurado se descargan en memoria del controlador y están disponibles para cargar. Para añadir comentarios a su texto estructurado:

Para añadir un comentario	Use uno de estos formatos
en una sola línea	//*comment* (**comment**) /**comment**/
al final de una línea de texto estructurado	
dentro de una línea de texto estructurado	(**comment**) /**comment**/
que abarque más de una línea	(**start of comment . . . end of comment**) /**start of comment . . . end of comment**/

Por ejemplo:

Formato	Ejemplo

//*comment*	**Al comienzo de una línea**
	//Check conveyor belt direction
	IF conveyor_direction THEN...
	Al final de una línea
	ELSE //If conveyor isn't moving, set alarm light
	light := 1;
	END_IF;

(**comment**)	Sugar.Inlet[:=]1;(*open the inlet*) IF Sugar.Low (*low level LS*)& Sugar.High (*high level LS*)THEN... (*Controla la velocidad de la bomba de recirculación. La velocidad depende de la temperatura del tanque*). IF tank.temp > 200 THEN...
/**comment**/	Sugar.Inlet:=0;/*close the inlet*/ IF bar_code=65 /*A*/ THEN... /*Obtiene el número de elementos en la matriz Inventory y almacena el valor en el tag Inventory_Items*/ SIZE(Inventory,0,Inventory_Items);

Servicio de soporte técnico de Rockwell Automation

Rockwell Automation ofrece información técnica en Internet para proporcionarle asistencia a la hora de utilizar sus productos. En http://support.rockwellautomation.com, puede encontrar manuales técnicos, una base de conocimientos con respuestas a preguntas frecuentes, notas técnicas y de aplicación, ejemplos de código y vínculos a Service Packs de software, así como una función llamada MySupport que usted puede personalizar para sacar el máximo provecho de estas herramientas.

Con el fin de brindarle un nivel adicional de soporte técnico por teléfono para la instalación, configuración y resolución de problemas, ofrecemos los programas de soporte técnico TechConnect. Si desea obtener más información, póngase en contacto con el representante o distribuidor local de Rockwell Automation, o visítenos en http://support.rockwellautomation.com.

Asistencia para la instalación

Si se le presenta un problema con un módulo de hardware durante las 24 horas posteriores a la instalación, revise la información proporcionada en este manual. También puede llamar a un número especial de soporte técnico al cliente a fin de recibir ayuda inicial para la puesta en marcha del módulo.

En los Estados Unidos	1.440.646.3223 Lunes a viernes de las 08:00 a las 17:00, hora oficial del Este de EE.UU.
Fuera de Estados Unidos	Póngase en contacto con el representante local de Rockwell Automation para cualquier consulta relacionada con soporte técnico.

Devolución de productos nuevos

Rockwell prueba todos sus productos para garantizar su correcto funcionamiento cuando salen de las instalaciones de fabricación. No obstante, si el producto no funciona quizás deba devolverlo.

En los Estados Unidos	Póngase en contacto con el distribuidor. Debe proporcionar al distribuidor un número de caso de soporte técnico (llame al número de teléfono indicado anteriormente para obtenerlo) a fin de completar el proceso de devolución.
Fuera de Estados Unidos	Póngase en contacto con el representante local de Rockwell Automation para obtener información sobre el procedimiento de devolución.

www.rockwellautomation.com

Oficinas corporativas de soluciones de potencia, control e información

Américas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
Europa/Medio Oriente/África: Rockwell Automation, Vorstlaan/Boulevard du Souverain 36, 1170 Bruselas, Bélgica, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia-Pacífico: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Argentina: Rockwell Automation S.A., Alem 1050, 5º Piso, CP 1001AAS, Capital Federal, Buenos Aires, Tel.: (54) 11.5554.4000, Fax: (54) 11.5554.4040, www.rockwellautomation.com.ar
Chile: Rockwell Automation Chile S.A., Luis Thayer Ojeda 166, Piso 6, Providencia, Santiago, Tel.: (56) 2.290.0700, Fax: (56) 2.290.0707, www.rockwellautomation.cl
Colombia: Rockwell Automation S.A., Edf. North Point, Carrera 7 N° 156 – 78 Piso 18, PBX: (57) 1.649.96.00 Fax: (57)649.96.15, www.rockwellautomation.com.co
España: Rockwell Automation S.A., Doctor Trueta 113-119, 08005 Barcelona, Tel.: (34) 932.959.000, Fax: (34) 932.959.001, www.rockwellautomation.es
México: Rockwell Automation S.A. de C.V., Bosques de Cierulos Nº 160, Col. Bosques de Las Lomas, C.P. 11700 México, D.F., Tel.: (52) 55.5246.2000, Fax: (52) 55.5251.1169, www.rockwellautomation.com.mx
Perú: Rockwell Automation S.A., Av Victor Andrés Belaunde N°147, Torre 12, Of. 102 – San Isidro Lima, Perú, Tel: (511) 441.59.00, Fax: (511) 222.29.87, www.rockwellautomation.com.pe
Puerto Rico: Rockwell Automation Inc., Calle 1, Metro Office # 6, Suite 304, Metro Office Park, Guaynabo, Puerto Rico 00968, Tel.: (1) 787.300.6200, Fax: (1) 787.706.3939, www.rockwellautomation.com.pr
Venezuela: Rockwell Automation S.A., Edf. Allen-Bradley, Av. González Rincones, Zona Industrial La Trinidad, Caracas 1080, Tel.: (58) 212.949.0611, Fax: (58) 212.943.3955, www.rockwellautomation.com.ve

Publicación 1756-RM007K-ES-P – Julio 2007
Reemplaza la publicación 1756-RM007H-ES-P – Diciembre de 2006

ANEXO K. Logix5000 Controllers General Instructions.

Logix5000 Controllers General Instructions

Allen-Bradley

Catalog Numbers 1756 ControlLogix, 1768 CompactLogix,
1769 CompactLogix, 1789 SoftLogix, 1794 FlexLogix,
PowerFlex 700S with DriveLogix

Reference Manual

Allen-Bradley • Rockwell Software

Rockwell
Automation

Important User Information

Solid state equipment has operational characteristics differing from those of electromechanical equipment. Safety Guidelines for the Application, Installation and Maintenance of Solid State Controls (publication SGI-1.1 available from your local Rockwell Automation sales office or online at http://www.rockwellautomation.com/literature/) describes some important differences between solid state equipment and hard-wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

In no event will Rockwell Automation, Inc. be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.

The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, Rockwell Automation, Inc. cannot assume responsibility or liability for actual use based on the examples and diagrams.

No patent liability is assumed by Rockwell Automation, Inc. with respect to use of information, circuits, equipment, or software described in this manual.

Reproduction of the contents of this manual, in whole or in part, without written permission of Rockwell Automation, Inc., is prohibited.

Throughout this manual, when necessary, we use notes to make you aware of safety considerations.

WARNING	Identifies information about practices or circumstances that can cause an explosion in a hazardous environment, which may lead to personal injury or death, property damage, or economic loss.
IMPORTANT	Identifies information that is critical for successful application and understanding of the product.
ATTENTION	Identifies information about practices or circumstances that can lead to personal injury or death, property damage, or economic loss. Attentions help you identify a hazard, avoid a hazard, and recognize the consequence
SHOCK HAZARD	Labels may be on or inside the equipment, for example, a drive or motor, to alert people that dangerous voltage may be present.
BURN HAZARD	Labels may be on or inside the equipment, for example, a drive or motor, to alert people that surfaces may reach dangerous temperatures.

Introduction

The release of this document contains new information.

New Information

New information is marked by change bars in the side column, as shown to the right.

Section	Changes
Where to Find an Instruction	The locator section has been updated to include Logix instructions from the Motion Coordinate System User Manual.
Chapter 3	New timer diagrams for the TON, TOF, and RTO instructions.
Chapter 4	Supported data types are defined for message (MSG) configuration and procedures for the Broadcast button on the Message Configuration Dialog Box. GSV/SSV object attributes are listed.
Chapter 9	Updated information for the Bit Shift Left (BSL) and Bit Shift Right (BSR) instructions.
Chapter 13	When PID control variables (MAXO, MINO) are set to equal they will reset to default values.
Chapter 18	Delete instruction is correctly defined.

Notes:

Instruction Locator

Preface

Chapter 1

FactoryTalk Alarms and Events Logix-based Instructions (ALMD, ALMA)

Chapter 2

Bit Instructions (XIC, XIO, OTE, OTL, OTU, ONS, OSR, OSF, OSRI, OSFI)

Chapter 3

Timer and Counter Instructions (TON, TOF, RTO, TONR, TOFR, RTOR, CTU, CTD, CTUD, RES)

Chapter 4

Input/Output Instructions (MSG, GSV, SSV, IOT)

Chapter 5

Compare Instructions (CMP, EQU, GEQ, GRT, LEQ, LES, LIM, MEQ, NEQ)

Chapter 6

Compute/Math Instructions (CPT, ADD, SUB, MUL, DIV, MOD, SQR, SQRT, NEG, ABS)

Chapter 7

Move/Logical Instructions (MOV, MVM, BTD, MVMT, BTDT, CLR, SWPB, AND, OR, XOR, NOT, BAND, BOR, BXOR, BNOT)

Chapter 8

Array (File)/Misc. Instructions (FAL, FSC, COP, CPS, FLL, AVE, SRT, STD, SIZE)

Chapter 9

Array (File)/Shift Instructions (BSL, BSR, FFL, FFU, LFL, LFU)

Chapter 10

Sequencer Instructions (SQI, SQO, SQL)

Chapter 11

Program Control Instructions (JMP, LBL, JSR, RET, SBR, JXR, TND, MCR, UID, UIE, AFI, NOP, EOT, SFP, SFR, EVENT)

Chapter 12

For/Break Instructions (FOR, FOR...DO, BRK, EXIT, RET)

Chapter 13

Special Instructions (FBC, DDT, DTR, PID)

Chapter 14

Trigonometric Instructions (SIN, COS, TAN, ASN, ASIN, ACS, ACOS, ATN, ATAN)

Chapter 15

Advanced Math Instructions (LN, LOG, XPY)

Chapter 16

Math Conversion Instructions (DEG, RAD, TOD, FRD, TRN, TRUNC)

Chapter 17

ASCII Serial Port Instructions (ABL, ACB, ACL, AHL, ARD, ARL, AWA, AWT)

Chapter 18

ASCII String Instructions (CONCAT, DELETE, FIND, INSERT, MID)

Chapter 19

ASCII Conversion Instructions (STOD, STOR, DTOS, RTOS, UPPER, LOWER)

Chapter 20

Debug Instructions (BPT, TPT)

Appendix A

Common Attributes

Appendix B

Function Block Attributes

Appendix C

Structured Text Programming

Index

Notes:

Where to Find an Instruction

Use this locator to find the reference details about Logix instructions (the grayed-out instructions are available in other manuals). This locator also lists which programming languages are available for the instructions.

If the locator lists	The instruction is documented in
A page number	This manual
Coordinate	Motion Coordinate System User Manual, publication MOTION-UM002
Motion	Logix5000 Controllers Motion Instructions Reference Manual, publication MOTION-RM002
PhaseManager	PhaseManager User Manual, publication LOGIX-UM001
Process control	Logix5000 Controllers Process Control and Drives Instructions Reference Manual, publication 1756-RM006

Instruction	Location	Languages
ABL ASCII Test For Buffer Line	617	Relay ladder Structured text
ABS Absolute Value	285	Relay ladder Structured text Function block
ACB ASCII Chars in Buffer	581	Relay ladder Structured text
ACL ASCII Clear Buffer	583	Relay ladder Structured text
ACS Arc Cosine	540	Relay ladder Structured text Function block
ADD Add	260	Relay ladder Structured text Function block
AFI Always False Instruction	464	Relay ladder
AHL ASCII Handshake Lines	585	Relay ladder Structured text
ALM Alarm	Process control	Structured text Function block
ALMA Analog Alarm	42	Relay ladder Structured text Function block
ALMD Digital Alarm	30	Relay ladder Structured text Function block
AND Bitwise AND	311	Relay ladder Structured text Function block
ARD ASCII Read	589	Relay ladder Structured text
ARL ASCII Read Line	593	Relay ladder Structured text
ASN Arc Sine	537	Relay ladder Structured text Function block
ATN Arc Tangent	543	Relay ladder Structured text Function block
AVE File Average	373	Relay ladder
AWA ASCII Write Append	597	Relay ladder Structured text
AWT ASCII Write	602	Relay ladder Structured text
BAND Boolean AND	325	Structured text Function block
BNOT Boolean NOT	334	Structured text Function block
BOR Boolean OR	328	Structured text Function block
BPT Breakpoints	635	Relay ladder
BRK Break	481	Relay ladder
BSL Bit Shift Left	394	Relay ladder
BSR Bit Shift Right	398	Relay ladder
BTD Bit Field Distribute	302	Relay ladder
BTDT Bit Field Distribute with Target	302	Structured text Function block
BTR Message	146	Relay ladder Structured text
BTW Message	146	Relay ladder Structured text
BXOR Boolean Exclusive OR	331	Structured text Function block
CC Coordinated Control	Process control	Structured text Function block
CLR Clear	302	Relay ladder Structured text
CMP Compare	214	Relay ladder
CONCAT String Concatenate	609	Relay ladder Structured text
COP Copy File	363	Relay ladder Structured text
COS Cosine	531	Relay ladder Structured text Function block
CPS Synchronous Copy File	363	Relay Ladder Structured text
CPT Compute	256	Relay ladder
CTD Count Down	134	Relay ladder
CTU Count Up	130	Relay ladder

Instruction	Location	Languages
CTUD Count Up/Down	138	Structured text Function block
D2SD Discrete 2-State Device	Process control	Structured text Function block
D3SD Discrete 3-State Device	Process control	Structured text Function block
DDT Diagnostic Detect	494	Relay ladder
DEDT Deadtime	Process control	Structured text Function block
DEG Degrees	558	Relay ladder Structured text Function block
DELETE String Delete	611	Relay ladder Structured text
DERV Derivative	Process control	Structured text Function block
DFF D Flip-Flop	Process control	Structured text Function block
DIV Divide	269	Relay ladder Structured text Function block
DTOS DINT to String	627	Relay ladder Structured text
DTR Data Transitional	502	Relay ladder
EOT End of Transition	466	Relay ladder Structured text
EQU Equal to	214	Relay ladder Structured text Function block
ESEL Enhanced Select	Process control	Structured text Function block
EVENT Trigger Event Task	472	Relay ladder Structured text
FAL File Arithmetic and Logic	343	Relay ladder
FBC File Bit Comparison	486	Relay ladder
FFL FIFO Load	402	Relay ladder
FFU FIFO Unload	408	Relay ladder
FGEN Function Generator	Process control	Structured text Function block

Instruction	Location	Languages
FIND Find String	613	Relay ladder Structured text
FLL File Fill	369	Relay ladder
FOR For	478	Relay ladder
FRD Convert to Integer	567	Relay ladder Function block
FSC File Search and Compare	354	Relay ladder
GEQ Greater than or Equal to	223	Relay ladder Structured text Function block
GRT Greater Than	227	Relay ladder Structured text Function block
GSV Get System Value	182	Relay ladder Structured text
HLL High/Low Limit	Process control	Structured text Function block
HPF High Pass Filter	Process control	Structured text Function block
ICON Input Wire Connector	649	Function block
IMC Internal Model Control	Process control	Structured text Function block
INSERT Insert String	615	Relay ladder Structured text
INTG Integrator	Process control	Structured text Function block
IOT Immediate Output	209	Relay ladder Structured text
IREF Input Reference	649	Function block
JKFF JK Flip-Flop	Process control	Structured text Function block
JMP Jump to Label	442	Relay ladder
JSR Jump to Subroutine	444	Relay ladder Structured text Function block
JXR Jump to External Routine	455	Relay ladder
LBL Label	442	Relay ladder

Instruction	Location	Languages
LDL2 Second-Order Lead Lag	Process control	Structured text Function block
LDLG Lead-Lag	Process control	Structured text Function block
LEQ Less Than or Equal to	231	Relay ladder Structured text Function block
LES Less Than	235	Relay ladder Structured text Function block
LFL LIFO Load	414	Relay ladder
LFU LIFO Unload	420	Relay ladder
LIM Limit	239	Relay ladder Function block
LN Natural Log	548	Relay ladder Structured text Function block
LOG Log Base 10	551	Relay ladder Structured text Function block
LOWER Lower Case	633	Relay ladder Structured text
LPF Low Pass Filter	Process control	Structured text Function block
MAAT Motion Apply Axis Tuning	Motion	Relay ladder Structured text
MAFR Motion Axis Fault Reset	Motion	Relay ladder Structured text
MAG Motion Axis Gear	Motion	Relay ladder Structured text
MAHD Motion Apply Hookup Diagnostics	Motion	Relay ladder Structured text
MAH Motion Axis Home	Motion	Relay ladder Structured text
MAJ Motion Axis Jog	Motion	Relay ladder Structured text
MAM Motion Axis Move	Motion	Relay ladder Structured text
MAOC Motion Arm Output Cam	Motion	Relay ladder Structured text
MAPC Motion Axis Position Cam	Motion	Relay ladder Structured text
MAR Motion Arm Registration	Motion	Relay ladder Structured text

Instruction	Location	Languages
MASD Motion Axis Shutdown	Motion	Relay ladder Structured text
MAS Motion Axis Stop	Motion	Relay ladder Structured text
MASR Motion Axis Shutdown Reset	Motion	Relay ladder Structured text
MATC Motion Axis Time Cam	Motion	Relay ladder Structured text
MAVE Moving Average	Process control	Structured text Function block
MAW Motion Arm Watch	Motion	Relay ladder Structured text
MAXC Maximum Capture	Process control	Structured text Function Block
MCCD Motion Coordinated Change Dynamics	Coordinate	Relay ladder Structured text
MCCM Motion Coordinated Circular Move	Coordinate	Relay ladder Structured text
MCCP Motion Calculate Cam Profile	Motion	Relay ladder Structured text
MCD Motion Change Dynamics	Motion	Relay ladder Structured text
MCLM Motion Coordinated Linear Move	Coordinate	Relay ladder Structured text
MCR Master Control Reset	460	Relay ladder
MCSD Motion Coordinated Shutdown	Coordinate	Relay ladder Structured text
MCS Motion Coordinated Stop	Coordinate	Relay ladder Structured Text
MCSR Motion Coordinated Shutdown Reset	Coordinate	Relay ladder Structured text
MCT Motion Coordinated Transform	Coordinate	Relay ladder Structured text
MCTP Motion Calculate Transform Position	Coordinate	Relay ladder Structured text
MDF Motion Direct Drive Off	Motion	Relay ladder Structured text

Instruction	Location	Languages
MDOC Motion Disarm Output Cam	Motion	Relay ladder Structured text
MDO Motion Direct Drive On	Motion	Relay ladder Structured text
MDR Motion Disarm Registration	Motion	Relay ladder Structured text
MDW Motion Disarm Watch	Motion	Relay ladder Structured text
MEQ Mask Equal to	245	Relay ladder Structured text Function Block
MGSD Motion Group Shutdown	Motion	Relay ladder Structured text
MGS Motion Group Stop	Motion	Relay ladder Structured text
MGSP Motion Group Strobe Position	Motion	Relay ladder Structured text
MGSR Motion Group Shutdown Reset	Motion	Relay ladder Structured text
MID Middle String	617	Relay ladder Structured text
MINC Minimum Capture	Process control	Structured text Function block
MMC Modular Multivariable Control	Process control	Structured text Function block
MOD Modulo	274	Relay ladder Structured text Function block
MOV Move	291	Relay ladder
MRAT Motion Run Axis Tuning	Motion	Relay ladder Structured text
MRHD Motion Run Hookup Diagnostics	Motion	Relay ladder Structured text
MRP Motion Redefine Position	Motion	Relay ladder Structured text
MSF Motion Servo Off	Motion	Relay ladder Structured text
MSG Message	146	Relay ladder Structured text
MSO Motion Servo On	Motion	Relay ladder Structured text

Instruction	Location	Languages
MSTD Moving Standard Deviation	Process control	Structured text Function block
MUL Multiply	266	Relay ladder Structured text Function block
MUX Multiplexer	Process control	Function block
MVM Masked Move	293	Relay ladder
MVMT Masked Move with Target	296	Structured text Function block
NEG Negate	282	Relay ladder Structured text Function block
NEQ Not Equal to	250	Relay ladder Structured text Function block
NOP No Operation	465	Relay ladder
NOT Bitwise NOT	322	Relay ladder Structured text Function block
NTCH Notch Filter	Process control	Structured text Function block
OCON Output Wire Connector	649	Function block
ONS One Shot	90	Relay ladder
OR Bitwise OR	314	Relay ladder Structured text Function block
OREF Output Reference	649	Function block
OSFI One Shot Falling with Input	101	Structured text Function block
OSF One Shot Falling	96	Relay ladder
OSRI One Shot Rising with Input	93	Structured text Function block
OSR One Shot Rising	93	Relay ladder
OTE Output Energize	84	Relay ladder
OTL Output Latch	86	Relay ladder
OTU Output Unlatch	88	Relay ladder

Instruction	Location	Languages
PATT Attach to Equipment Phase	PhaseManager	Relay ladder Structured text
PCLF Equipment Phase Clear Failure	PhaseManager	Relay ladder Structured text
PCMD Equipment Phase Command	PhaseManager	Relay ladder Structured text
PDET Detach from Equipment Phase	PhaseManager	Relay ladder Structured text
PFL Equipment Phase Failure	PhaseManager	Relay ladder Structured text
PIDE Enhanced PID	Process control	Structured text Function block
PID Proportional Integral Derivative	505	Relay ladder Structured text
PI Proportional + Integral	Process control	Structured text Function block
PMUL Pulse Multiplier	Process control	Structured text Function block
POSP Position Proportional	Process control	Structured text Function block
POVR Equipment Phase Override Command	PhaseManager	Relay ladder Structured text
PPD Equipment Phase Paused	PhaseManager	Relay ladder Structured text
PRNP Equipment Phase New Parameters	PhaseManager	Relay ladder Structured text
PSC Phase State Complete	PhaseManager	Relay ladder Structured text
PXRQ Equipment Phase External Request	PhaseManager	Relay ladder Structured text
RAD Radians	561	Relay ladder Structured text Function block
RESD Reset Dominant	Process control	Structured text Function block
RES Reset	143	Relay ladder
RET Return	444 and 482	Relay ladder Structured text Function block

Instruction	Location	Languages
RLIM Rate Limiter	Process control	Structured text Function block
RMPS Ramp/Soak	Process control	Structured text Function block
RTO Retentive Timer On	114	Relay ladder
RTOR Retentive Timer On with Reset	126	Structured text Function block
RTOS REAL to String	629	Relay ladder Structured text
SBR Subroutine	444	Relay ladder Structured text Function block
SCL Scale	Process control	Structured text Function block
SCRV S-Curve	Process control	Structured text Function block
SEL Select	Process control	Function block
SETD Set Dominant	Process control	Structured text Function block
SFP SFC Pause	468	Relay ladder Structured text
SFR SFC Reset	470	Relay ladder Structured text
SIN Sine	528	Relay ladder Structured text Function block
SIZE Size In Elements	389	Relay ladder Structured text
SNEG Selected Negate	Process control	Structured text Function block
SOC Second-Order Controller	Process control	Structured text Function block
SQI Sequencer Input	428	Relay ladder
SQL Sequencer Load	436	Relay ladder
SQO Sequencer Output	432	Relay ladder
SQR Square Root	278	Relay ladder Function block
SQRT Square Root	278	Structured text

Instruction	Location	Languages
SRT File Sort	378	Relay ladder Structured text
SRTP Split Range Time Proportional	Process control	Structured text Function block
SSUM Selected Summer	Process control	Structured text Function block
SSV Set System Value	182	Relay ladder Structured text
STD File Standard Deviation	383	Relay ladder
STOD String To DINT	621	Relay ladder Structured text
STOR String To REAL	624	Relay ladder Structured text
SUB Subtract	263	Relay ladder Structured text Function block
SWPB Swap Byte	307	Relay ladder Structured text
TAN Tangent	534	Relay ladder Structured text Function block
TND Temporary End	458	Relay ladder
TOD Convert to BCD	564	Relay ladder Function block
TOFR Timer Off Delay with Reset	122	Structured text Function block
TOF Timer Off Delay	110	Relay ladder
TONR Timer On Delay with Reset	118	Structured text Function block
TON Timer On Delay	106	Relay ladder
TOT Totalizer	Process control	Structured text Function block
TPT Tracepoints	639	Relay ladder
TRN Truncate	569	Relay ladder Function block
TRUNC Truncate	569	Structured text
UID User Interrupt Disable	462	Relay ladder Structured text

Instruction	Location	Languages
UIE User Interrupt Enable	462	Relay ladder Structured text
UPDN Up/Down Accumulator	Process control	Structured text Function block
UPPER Upper Case	631	Relay ladder Structured text
XIC Examine If Closed	78	Relay ladder
XIO Examine If Open	81	Relay ladder
XOR Bitwise Exclusive OR	318	Relay ladder Structured text Function block
XPY X to the Power of Y	554	Relay ladder Structured text Function block

Notes:

Preface

Introduction

This manual provides a programmer with details about each available instruction for a Logix-based controller. You should be familiar with how the Logix-based controller stores and processes data.

Novice programmers should read all the details about an instruction before using the instruction. Experienced programmers can refer to the instruction information to verify details.

This manual is one of a set of related manuals that show common procedures for programming and operating Logix5000 controllers. For a complete list of common procedures manuals, see the Logix5000 Controllers Common Procedures Programming Manual, publication 1756-PM001.

The term Logix5000 controller refers to any controller that is based on the Logix5000 operating system, such as:

- CompactLogix controllers.
- ControlLogix controllers.
- DriveLogix controllers.
- FlexLogix controllers.
- SoftLogix5800 controllers.

Description of Instruction Format

Section	Information
Instruction name	Identifies the instruction. Defines whether the instruction is an input or an output instruction.
Operands	Lists all the operands of the instruction. If available in relay ladder, describes the operands. If available in structured text, describes the operands. If available in function block, describes the operands. The pins shown on a default function block are only the default pins. The operands table lists all the possible pins for a function block.
Instruction structure	Lists control status bits and values, if any, of the instruction,
Description	Describes the instruction's use. Defines any differences when the instruction is enabled and disabled, if appropriate.
Arithmetic status flags	Defines whether or not the instruction affects arithmetic status flags.

Description of Instruction Format

Section	Information
Fault conditions	Defines whether or not the instruction generates minor or major faults. If so, defines the fault type and code.
Execution	Defines the specifics of how the instruction operates.
Example	Provides at least one programming example in each available programming language. Includes a description explaining each example

The following icons help identify language specific information.

Icon	Programming Language
	Relay ladder
	Structured text
	Function block

Common Information for All Instructions

The Logix5000 instruction set has some common attributes.

Information	Appendix
Common attributes	Common Attributes defines: • arithmetic status flags • data types • keywords
Function block attributes	Function Block Attributes defines: • program and operator control • timing modes

Conventions and Related Terms

This manual uses set and clear to define the status of bits (booleans) and values (non-booleans).

Term	Means
Set	The bit is set to 1 (ON). A value is set to any non-zero number.
Clear	The bit is cleared to 0 (OFF). All the bits in a value are cleared to 0.

If an operand or parameter support more than one data type, the **bold** data types indicate optimal data types. An instruction executes faster and requires less memory if all the operands of the instruction use the same optimal data type, typically DINT or REAL.

Relay Ladder Rung Condition

The controller evaluates ladder instructions based on the rung condition preceding the instruction (rung-condition-in). Based on the rung-condition-in and the instruction, the controller sets the rung condition following the instruction (rung-condition-out), which in turn, affects any subsequent instruction.

If the rung-in condition to an input instruction is true, the controller evaluates the instruction and sets the rung-out condition based on the results of the instruction. If the instruction evaluates to true, the rung-out condition is true; if the instruction evaluates to false, the rung-out condition is false.

The controller also prescans instructions. Prescan is a special scan of all routines in the controller. The controller scans all main routines and subroutines during prescan, but ignores jumps that could skip the execution of instructions. The controller executes all FOR loops and subroutine calls. If a subroutine is called more than once, it is executed each time it is called. The controller uses prescan of relay ladder instructions to reset non-retentive I/O and internal values.

During prescan, input values are not current and outputs are not written. The following conditions generate prescan:

- Toggle from Program to Run mode.
- Automatically enter Run mode from a power-up condition.

Prescan does not occur for a program when:

- the program becomes scheduled while the controller is running.
- the program is unscheduled when the controller enters Run mode.

Function Block States

IMPORTANT When programming in function block, restrict the range of engineering units to +/-$10^{+/-15}$ because internal floating point calculations are done by using single precision floating point. Engineering units outside of this range may result in a loss of accuracy if results approach the limitations of single precision floating point (+/-$10^{+/-38}$).

The controller evaluates function block instructions based on the state of different conditions..

Possible Condition	Description
Prescan	Prescan for function block routines is the same as for relay ladder routines. The only difference is that the EnableIn parameter for each function block instruction is cleared during prescan.
Instruction first scan	Instruction first scan refers to the first time an instruction is executed after prescan. The controller uses instruction first scan to read current inputs and determine the appropriate state to be in.
Instruction first run	Instruction first run refers to the first time the instruction executes with a new instance of a data structure. The controller uses instruction first run to generate coefficients and other data stores that do not change for a function block after initial download.

Every function block instruction also includes EnableIn and EnableOut parameters.

- Function block instructions execute normally when EnableIn is set.

- When EnableIn is cleared, the function block instruction either executes prescan logic, postscan logic, or just skips normal algorithm execution.

- EnableOut mirrors EnableIn, however, if function block execution detects an overflow condition EnableOut is also cleared.

- Function block execution resumes where it left off when EnableIn toggles from cleared to set. However, there are some function block instructions that specify special functionality, such as re-initialization, when EnableIn toggles from cleared to set. For function block instructions with time base parameters, whenever the timing mode is Oversample, the instruction always resumes were it left off when EnableIn toggles from cleared to set.

If the EnableIn parameter is not wired, the instruction always executes as normal and EnableIn remains set. If you clear EnableIn, it changes to set the next time the instruction executes.

Notes:

FactoryTalk Alarms and Events Logix-based Instructions

(ALMD, ALMA)

Introduction

These Logix-based alarm instructions are available in relay ladder, structured text, and function block diagram. When used with FactoryTalk View SE software, version 5.0 and later, these instructions create a system with your visualization package. The controller detects alarm conditions and publishes events to FactoryTalk View Alarms and Events servers that propagate alarms to Factory Talk View SE clients that subscribe to receive notifications.

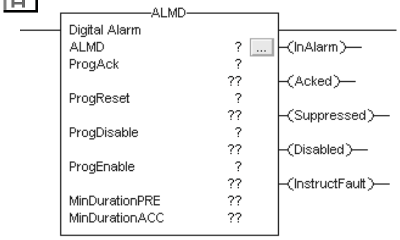
If you want to	Use this instruction	Available in	Page
Detect alarms based on Boolean (true/false) conditions	ALMD	Relay ladder Structured text Function block	30
Detect alarms based on the level or rate of change of a value	ALMA	Relay ladder Structured text Function block	42

Digital Alarm (ALMD)

The ALMD instruction detects alarms based on Boolean (true/false) conditions. Program (Prog) and operator (Oper) control parameters provide an interface for alarm commands.

Operands:

Relay Ladder



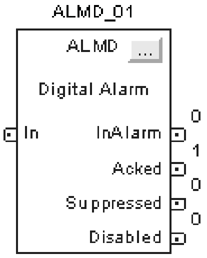
In relay ladder, the alarm condition input (In) is obtained from the rung condition.

Operand	Type	Format	Description
ALMD tag	ALARM_DIGITAL	Structure	ALMD structure.
In	BOOL	Tag Immediate	**Structured text only.** Value is copied to In when instruction executes. The alarm input value is compared to determine whether there is an alarm.
ProgAck	BOOL	Tag Immediate	Value is copied to ProgAck when instruction executes. On transition from cleared to set, acknowledges alarm (if acknowledgement is required).
ProgReset	BOOL	Tag Immediate	Value is copied to ProgReset when instruction executes. On transition from cleared to set, resets alarm (if required).
ProgDisable	BOOL	Tag Immediate	Value is copied to ProgDisable when instruction executes. When set, disables alarm (does not override Enable Commands).
ProgEnable	BOOL	Tag Immediate	Value is copied to ProgEnable when instruction executes. When set, enables alarm (takes precedence over Disable Commands).
MinDurationPRE	DINT	Immediate	**Relay ladder only.** Specifies how long the alarm condition must be met before it is reported (milliseconds).
MinDurationACC	DINT	Immediate	**Relay ladder only.** Indicates the number of milliseconds that have elapsed since the alarm condition was met.

Structured Text

```
ALMD(ALMD, In, ProgAck,
ProgReset, ProgDisable,
ProgEnable);
```

The operands are the same as those for the relay ladder ALMD instruction, with a few exceptions as indicated above.



Function Block

Operand	Type	Format	Description
ALMD tag	ALARM_DIGITAL	Structure	ALMD structure

ALARM_DIGITAL Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	**Relay Ladder** Corresponds to the rung state. Does not affect processing. **Function Block** If cleared, the instruction does not execute and outputs are not updated. If set, the instruction executes. Default is set. **Structured Text** No effect. The instruction always executes.
In	BOOL	The digital signal input to the instruction. Default is cleared. **Relay Ladder** Follows the rung condition. Set if the rung condition is true. Cleared if the rung condition is false. **Structured Text** Copied from instruction operand.
InFault	BOOL	Bad health indicator for the input. The user application may set InFault to indicate the input signal has an error. When set, the instruction sets InFaulted (Status.1). When cleared, the instruction clears InFaulted (Status.1). In either case, the instruction continues to evaluate In for alarm conditions. Default is cleared (good health).
Condition	BOOL	Specifies how alarm is activated. When Condition is set, the alarm condition is activated when In is set. When Condition is cleared, the alarm condition is activated when In is cleared. Default is set.

Input Parameter	Data Type	Description
AckRequired	BOOL	Specifies whether alarm acknowledgement is required. When set, acknowledgement is required. When cleared, acknowledgement is not required and Acked is always set. Default is set.
Latched	BOOL	Specifies whether the alarm is latched. Latched alarms remain InAlarm when the alarm condition becomes false, until a Reset command is received. When set, the alarm is latched. When cleared, the alarm is unlatched. A latched alarm can be reset only when the alarm condition is false. Default is cleared.
ProgAck	BOOL	Set by the user program to acknowledge the alarm. Requires a cleared-to-set transition while the alarm is unacknowledged. Default is cleared. **Relay Ladder** Copied from the instruction operand. **Structured Text** Copied from the instruction operand.
OperAck	BOOL	Set by the operator interface to acknowledge the alarm. Requires a cleared-to-set transition while the alarm is unacknowledged. The instruction clears this parameter. Default is cleared.
ProgReset	BOOL	Set by the user program to reset the alarm. Requires a cleared-to-set transition while the alarm is InAlarm and the In condition is not in alarm. Default is cleared. **Relay Ladder** Copied from the instruction operand. **Structured Text** Copied from the instruction operand.
OperReset	BOOL	Set by the operator interface to reset the alarm. Requires a cleared-to-set transition while the alarm is InAlarm and the In condition is not in alarm. The alarm instruction clears this parameter. Default is cleared.
ProgSuppress	BOOL	Set by the user program to suppress the alarm. Default is cleared.
OperSuppress	BOOL	Set by the operator interface to suppress the alarm. The alarm instruction clears this parameter. Default is cleared.

Input Parameter	Data Type	Description
ProgUnsuppress	BOOL	Set by the user program to unsuppress the alarm. Takes precedence over Suppress commands. Default is cleared.
OperUnsuppress	BOOL	Set by the operator interface to unsuppress the alarm. Takes precedence over Suppress commands. The alarm instruction clears this parameter. Default is cleared.
ProgDisable	BOOL	Set by the user program to disable the alarm. Default is cleared. **Relay Ladder** Copied from the instruction operand. **Structured Text** Copied from the instruction operand.
OperDisable	BOOL	Set by the operator interface to disable the alarm. The alarm instruction clears this parameter. Default is cleared.
ProgEnable	BOOL	Set by the user program to enable the alarm. Takes precedence over a Disable command. Default is cleared. **Relay Ladder** Copied from the instruction operand. **Structured Text** Copied from the instruction operand.
OperEnable	BOOL	Set by the operator interface to enable the alarm. Takes precedence over Disable command. The alarm instruction clears this parameter. Default is cleared.
AlarmCountReset	BOOL	Set by the user program to reset the alarm count. A cleared-to-set transition resets the alarm count to zero. Default is cleared.
UseProgTime	BOOL	Specifies whether to use the controller’s clock or the ProgTime value to timestamp alarm state change events. When set, the ProgTime value provides timestamp. When cleared, the controller’s clock provides timestamp. Default is cleared.

Input Parameter	Data Type	Description
ProgTime	LINT	If UseProgTime is set, this value is used to provide the timestamp value for all events. This lets the application apply timestamps obtained from the alarm source, such as a sequence-of-events input module.
Severity	DINT	Severity of the alarm. This does not affect processing of alarms by the controller, but can be used for sorting and filtering functions at the alarm subscriber. Valid = 1…1000 (1000 = most severe; 1 = least severe). Default = 500.
MinDurationPRE	DINT	Minimum duration preset (milliseconds) for the alarm condition to remain true before the alarm is marked as InAlarm and alarm notification is sent to clients. The controller collects alarm data as soon as the alarm condition is detected, so no data is lost while waiting to meet the minimum duration. Valid = 0…2,147,483,647. Default = 0.

Output Parameter	Data Type	Description
EnableOut	BOOL	Enable output.
InAlarm	BOOL	Alarm active status. Set when the alarm is active. Cleared when the alarm is not active (normal status).
Acked	BOOL	Alarm acknowledged status. Set when the alarm is acknowledged. Cleared when the alarm is not acknowledged. Acked is always set when AckRequired is cleared.
InAlarmUnack	BOOL	Combined alarm active and acknowledged status. Set when the alarm is active (InAlarm is set) and unacknowledged (Acked is cleared). Cleared when the alarm is normal (inactive), acknowledged, or both.
Suppressed	BOOL	Suppressed status of the alarm. Set when the alarm is suppressed. Cleared when the alarm is not suppressed.
Disabled	BOOL	Disabled status of the alarm. Set when the alarm is disabled. Cleared when the alarm is enabled.
MinDurationACC	DINT	Elapsed time since the alarm was detected. When this value reaches MinDurationPRE, the alarm becomes active (InAlarm is set), and a notification is sent to clients.
AlarmCount	DINT	Number of times the alarm has been activated (InAlarm is set). If the maximum value is reached, the counter leaves the value at the maximum count value.
InAlarmTime	LINT	Timestamp of alarm detection.
AckTime	LINT	Timestamp of alarm acknowledgement. If the alarm does not require acknowledgement, this timestamp is equal to alarm time.
RetToNormalTime	LINT	Timestamp of alarm returning to a normal state.
AlarmCountResetTime	LINT	Timestamp indicating when the alarm count was reset.
DeliveryER	BOOL	Delivery error of the alarm notification message. Set when there is a delivery error, either no alarm subscriber was subscribed or at least one subscriber did not receive the latest alarm change state message. Cleared when delivery is successful or is in progress.

Output Parameter	Data Type	Description
DeliveryDN	BOOL	Delivery completion of the alarm notification message. Set when delivery is successful, at least one subscriber was subscribed and all subscribers received the latest alarm change state message successfully. Cleared when delivery does not complete successfully or is in progress.
DeliveryEN	BOOL	Delivery status of the alarm notification message. Set when delivery is in progress. Cleared when delivery is not in progress.
NoSubscriber	BOOL	Alarm had no subscribers when attempting to deliver the most recent message. Set when there are no subscribers. Cleared when there is at least one subscriber.
NoConnection	BOOL	Alarm's subscribers were not connected when attempting to deliver the most recent message. Set when all subscribers are disconnected. Cleared when at least one subscriber is connected or there are no subscribers.
CommError	BOOL	Communication error when delivering an alarm message. Set when there are communication errors and all retries are used. This means that a subscriber was subscribed and it had a connection, but the controller did not receive confirmation of message delivery. Cleared when all connected subscribers confirm receipt of the alarm message.
AlarmBuffered	BOOL	Alarm message buffered due to a communication error (CommError is set) or a lost connection (NoConnection is set). Set when the alarm message is buffered for at least one subscriber. Cleared when the alarm message is not buffered.
Subscribers	DINT	Number of subscribers for this alarm.
SubscNotified	DINT	Number of subscribers successfully notified about the most recent alarm state change.
Status	DINT	Combined status indicators: Status.0 = InstructFault. Status.1= InFaulted. Status.2 = SeverityInv.
InstructFault (Status.0)	BOOL	Instruction error conditions exist. This is not a minor or major controller error. Check the remaining status bits to determine what occurred.
InFaulted (Status.1)	BOOL	User program has set InFault to indicate bad quality input data. Alarm continues to evaluate In for alarm condition.
SeverityInv (Status.2)	BOOL	Alarm severity configuration is invalid. If severity <1, the instruction uses Severity = 1. If severity >1000, the instruction uses Severity = 1000.

Description

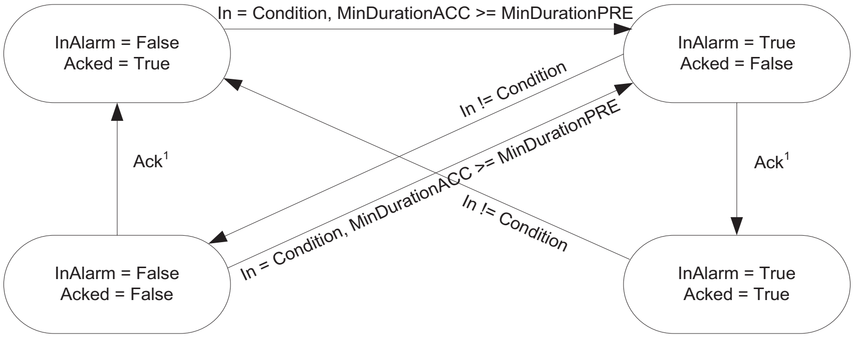
The ALMD instruction detects alarms based on Boolean (true/false) conditions.

The ALMD instruction provides additional functionality when used with RSLinx Enterprise and FactoryTalk View SE software. You can display alarms in the Alarm Summary, Alarm Banner, Alarm Status Explorer, and Alarm Log Viewer displays in FactoryTalk View SE software.

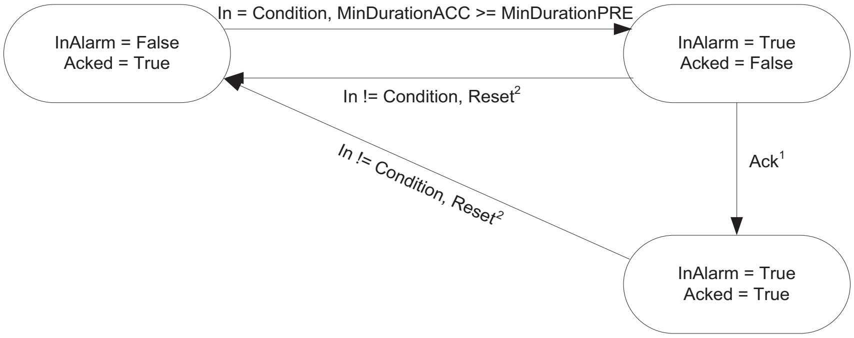
RSLinx Enterprise software subscribes to alarms in the controller. Use the output parameters to monitor the instruction to see the alarm subscription status and to display alarm status changes. If a connection to RSLinx Enterprise software is lost, the controller can briefly buffer alarm data until the connection is restored.

State Diagrams when Acknowledgement Required

Latched = False



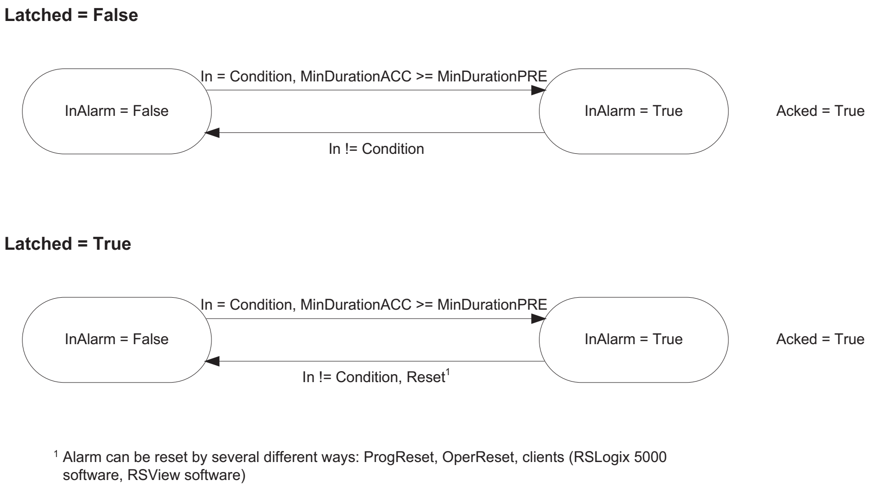
Latched = True



[1] Alarm can be acked by several different ways: ProgAck, OperAck, clients (RSLogix 5000 software, RSView software).

[2] Alarm can be reset by several different ways: ProgReset, OperReset, clients (RSLogix 5000 software, RSView software).

State Diagrams when Acknowledgment Not Required



Arithmetic Status Flags: None

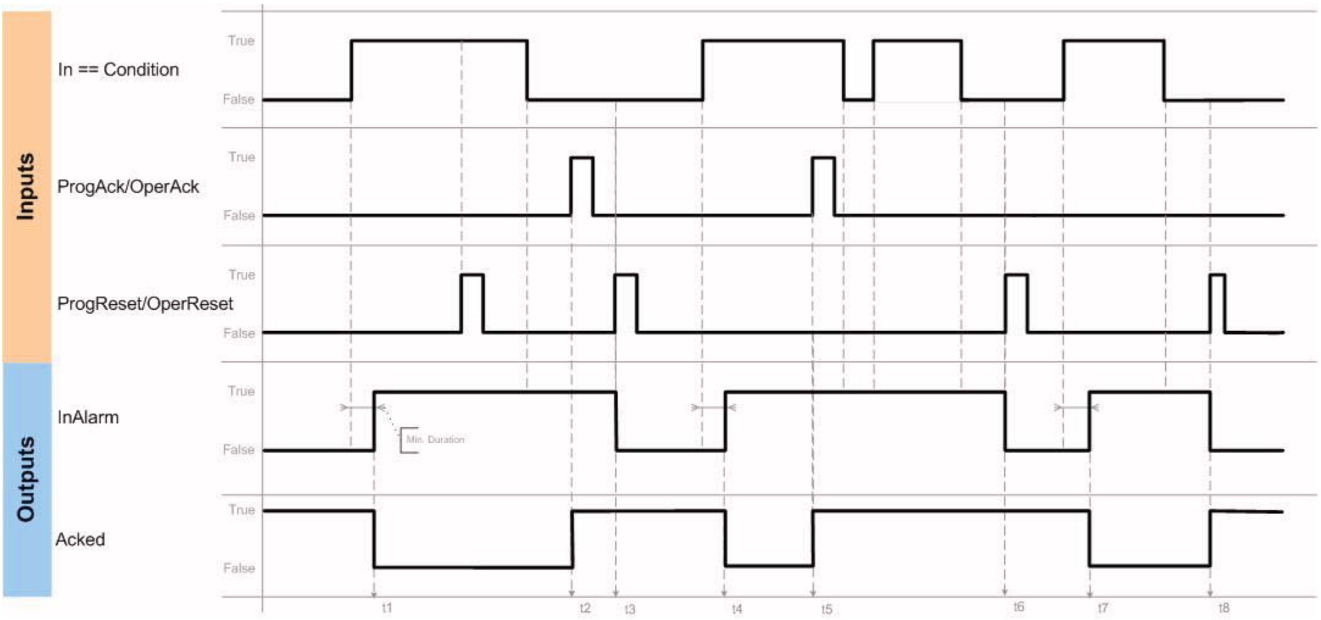
Fault Conditions: None

Execution:

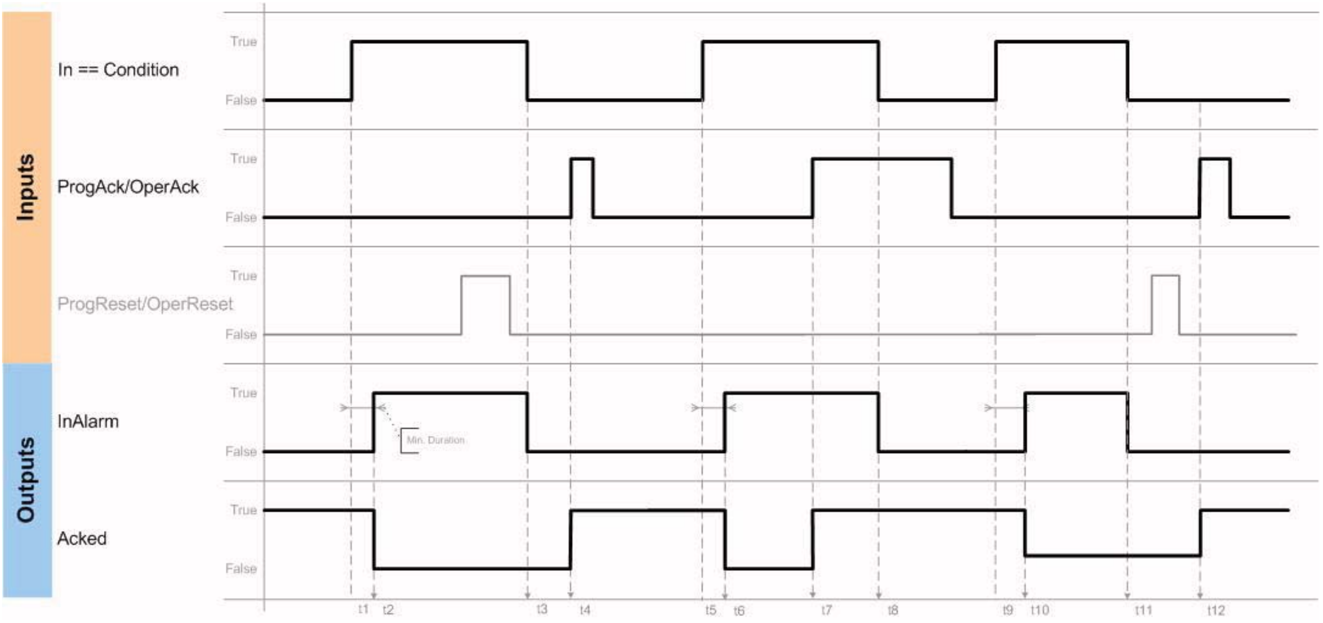
Condition	Relay Ladder Action
Prescan	The rung-condition-out is set to false. InAlarm is cleared and Acked is set. All operator requests, timestamps, and delivery flags are cleared.
Rung-condition-in is false	The rung-condition-out is set to false. EnableIn and EnableOut are cleared. The In parameter is cleared, and the instruction evaluates to determine the alarm state.
Rung-condition-in is true	The rung-condition-out is set to true. EnableIn and EnableOut are set. The In parameter is set, and the instruction evaluates to determine the alarm state.
Postscan	The rung-condition-out is set to false.

Condition	Function Block Action	Structured Text Action
Prescan	All operator requests, timestamps, and delivery flags are cleared. InAlarm is cleared and Acked is set.	All operator requests, timestamps, and delivery flags are cleared. InAlarm is cleared and Acked is set.
Instruction first scan	No action taken.	No action taken.
Instruction first run	No action taken.	No action taken.
EnableIn is cleared	The instruction does not execute. EnableOut is cleared.	The instruction executes. EnableOut is always set.
EnableIn is set	The instruction executes. EnableOut is set.	The instruction executes. EnableOut is always set.
Postscan	No action taken.	No action taken.

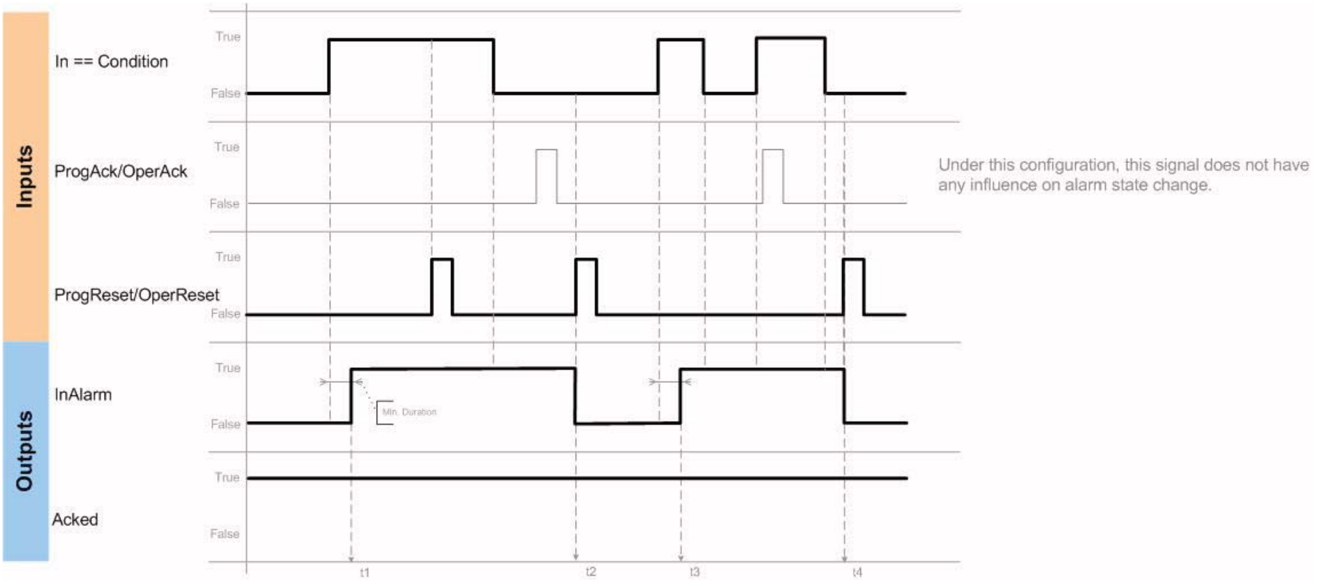
ALMD Alarm Acknowledge Required and Latched



ALMD Alarm Acknowledge Required and Not Latched

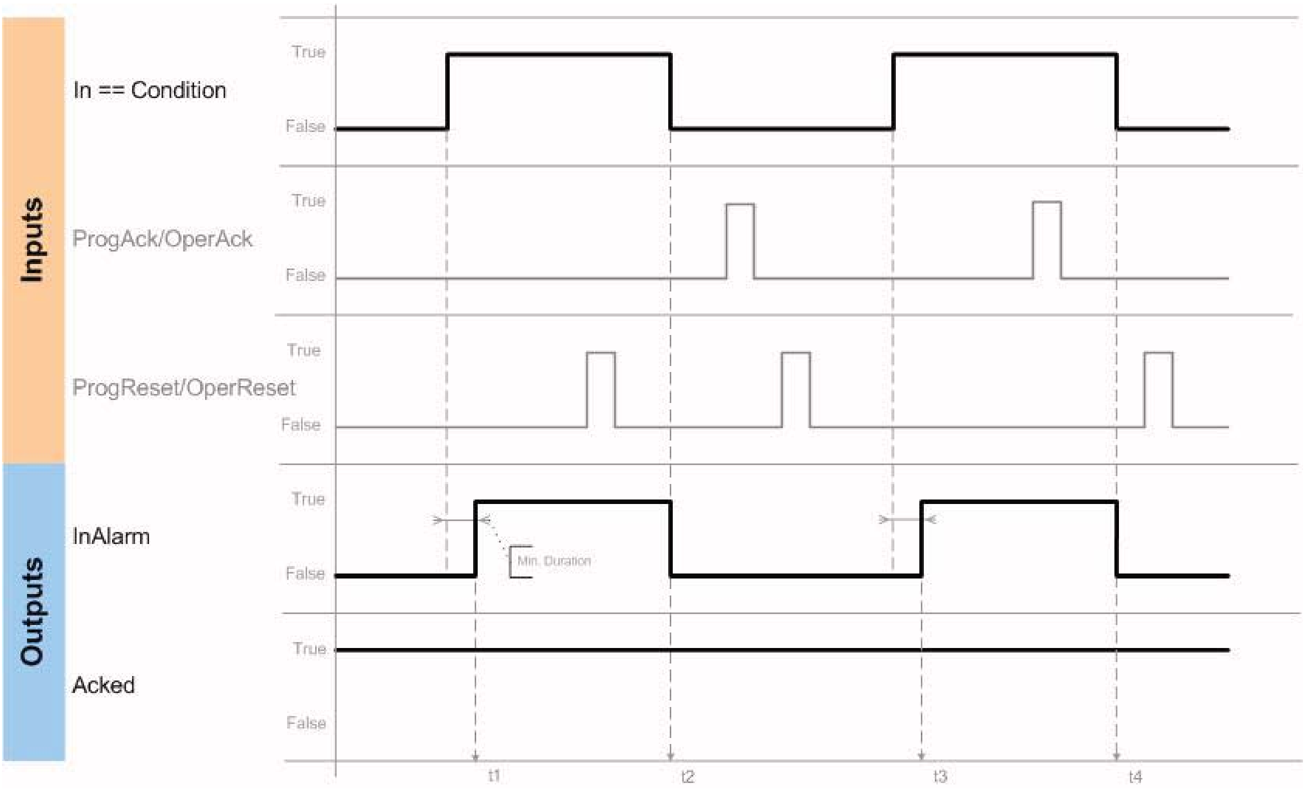


ALMD Alarm Acknowledge Not Required and Latched



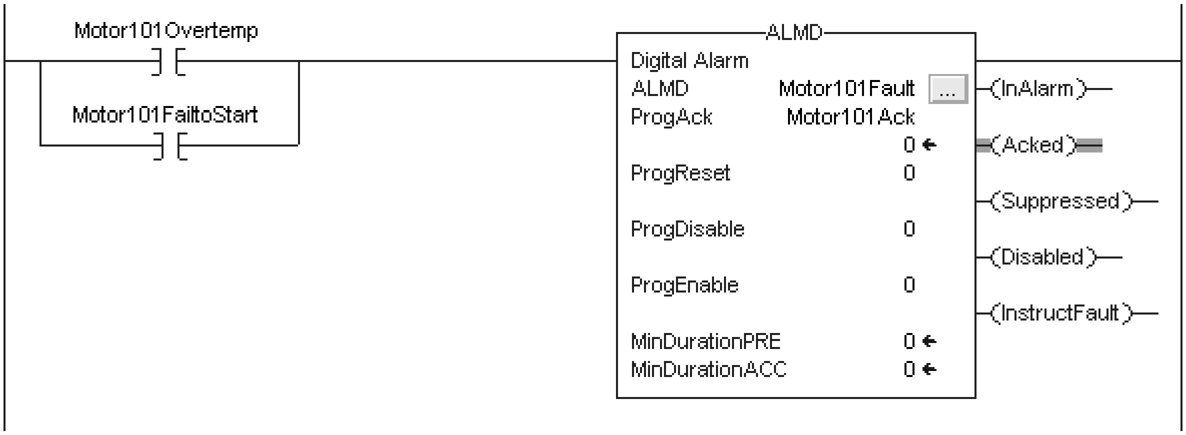
Under this configuration, this signal does not have any influence on alarm state change.

ALMD Alarm Acknowledge Not Required and Not Latched



Example: Two motor failure signals are combined such that if either one occurs, a motor fault alarm is activated. Programmatically acknowledge the alarm with a cleared-to-set transition of the Motor101Ack tag value. The application logic must clear Motor101Ack.

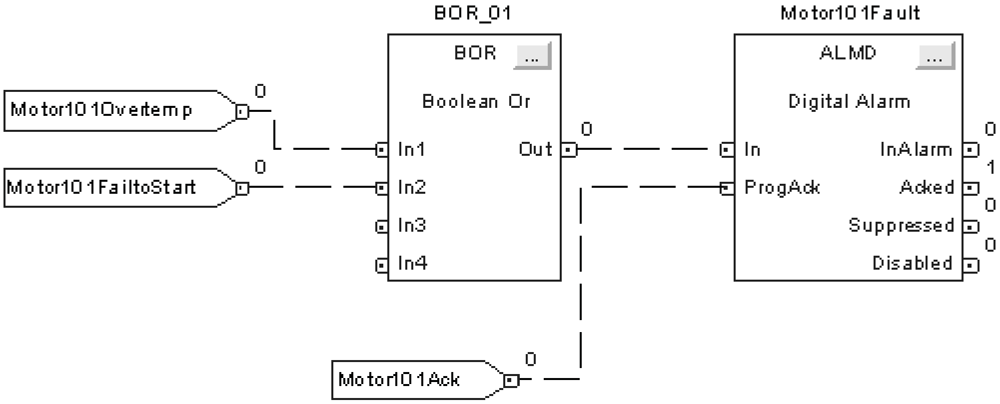
Relay Ladder



Structured Text

```
Motor101FaultConditions := Motor101Overtemp OR
Motor101FailToStart;

ALMD(Motor101Fault,Motor101FaultConditions,
Motor101Ack,0,0,0 );
```

Function Block

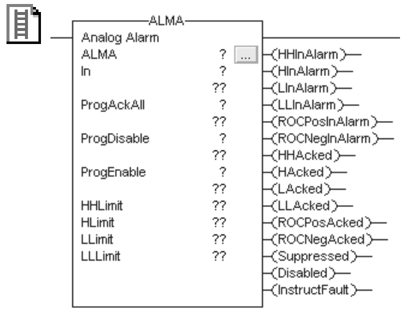


Analog Alarm (ALMA)

The ALMA instruction detects alarms based on the level or rate of change of an analog value. Program (Prog) and operator (Oper) control parameters provide an interface for alarm commands.

Operands:

Relay Ladder

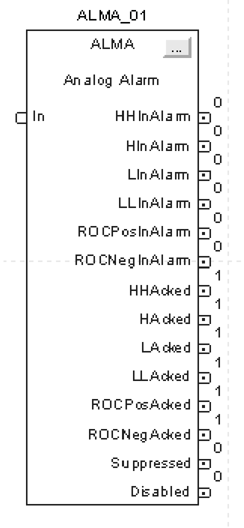


Operand	Type	Format	Description
ALMA tag	ALARM_ANALOG	Structure	ALMA structure.
In	REAL DINT INT SINT	Tag Immediate	Value is copied to In when instruction executes. The alarm input value, which is compared with alarm limits to detect the alarm conditions.
ProgAckAll	BOOL	Tag Immediate	Value is copied to ProgAckAll when instruction executes. On transition from cleared to set, acknowledges all alarm conditions that require acknowledgement.
ProgDisable	BOOL	Tag Immediate	Value is copied to ProgDisable when instruction executes. When set, disables alarm (does not override Enable Commands).
ProgEnable	BOOL	Tag Immediate	Value is copied to ProgEnable when instruction executes. When set, enables alarm (takes precedence over Disable commands).
HHlimit	REAL	Immediate	**Relay ladder only.** High High alarm limit.
HLimit	REAL	Immediate	**Relay ladder only.** High alarm limit.
LLimit	REAL	Immediate	**Relay ladder only.** Low alarm limit.
LLLimit	REAL	Immediate	**Relay ladder only.** Low Low alarm limit.

```
ALMA(ALMA, In, ProgAckAll,
ProgDisable, ProgEnable);
```

Structured Text

The operands are the same as those for the relay ladder ALMD instruction, with a few exceptions as indicated above.



Function Block

Operand	Type	Format	Description
ALMA tag	ALARM_ANALOG	Structure	ALMA structure

ALARM_ANALOG Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	**Relay Ladder** Corresponds to the rung state. If cleared, the instruction does not execute and outputs are not updated. **Structured Text** No effect. The instruction always executes. **Function Block** If cleared, the instruction does not execute and outputs are not updated. Default is set.
In	REAL	The alarm input value, which is compared with alarm limits to detect alarm conditions. Default = 0.0. **Relay Ladder** Copied from the instruction operand. **Structured Text** Copied from instruction operand.
InFault	BOOL	Bad health indicator for the input. The user application may set InFault to indicate the input signal has an error. When set, the instruction sets InFaulted (Status.1). When cleared, the instruction clears InFaulted (Status.1). In either case, the instruction continues to evaluate In for alarm conditions. Default is cleared (good health).
HHEnabled	BOOL	High High alarm condition detection. Set to enable detection of the High High alarm condition. Clear to disable detection of the High High alarm condition. Default is set.
HEnabled	BOOL	High alarm condition detection. Set to enable detection of the High alarm condition. Clear to disable detection of the High alarm condition. Default is set.
LEnabled	BOOL	Low alarm condition detection. Set to enable detection of the Low alarm condition. Clear to disable detection of the Low alarm condition. Default is set.
LLEnabled	BOOL	Low Low alarm condition detection. Set to enable detection of the Low Low alarm condition. Clear to disable detection of the Low Low alarm condition. Default is set.

Input Parameter	Data Type	Description
AckRequired	BOOL	Specifies whether alarm acknowledgement is required. When set, acknowledgement is required. When cleared, acknowledgement is not required and HHAcked, HAcked, LAcked, LLAcked, ROCPosAcked, and ROCNegAcked are always set. Default is set.
ProgAckAll	BOOL	Set by the user program to acknowledge all conditions of this alarm. Requires a cleared-to-set transition while the alarm conditions are unacknowledged. Default is cleared. **Relay Ladder** Copied from the instruction operand. **Structured Text** Copied from the instruction operand.
OperAckAll	BOOL	Set by the operator interface to acknowledge all conditions of this alarm. Requires a cleared-to-set transition while the alarm conditions are unacknowledged. The alarm instruction clears this parameter. Default is cleared.
HHProgAck	BOOL	High High program acknowledge. Set by the user program to acknowledge a High High condition. Requires a cleared-to-set transition while the alarm condition is unacknowledged. Default is cleared.
HHOperAck	BOOL	High High operator acknowledge. Set by the operator interface to acknowledge a High High condition. Requires a cleared-to-set transition while the alarm condition is unacknowledged. The alarm instruction clears this parameter. Default is cleared.
HProgAck	BOOL	High program acknowledge. Set by the user program to acknowledge a High condition. Requires a cleared-to-set transition while the alarm condition is unacknowledged. Default is cleared.
HOperAck	BOOL	High operator acknowledge. Set by the operator interface to acknowledge a High condition. Requires a cleared-to-set transition while the alarm condition is Unacknowledged. The alarm instruction clears this parameter. Default is cleared.
LProgAck	BOOL	Low program acknowledge. Set by the user program to acknowledge a Low condition. Requires a cleared-to-set transition while the alarm condition is unacknowledged. Default is cleared.
LOperAck	BOOL	Low operator acknowledge. Set by the operator interface to acknowledge a Low condition. Requires a cleared-to-set transition while the alarm condition is unacknowledged. The alarm instruction clears this parameter. Default is cleared.

Input Parameter	Data Type	Description
LLProgAck	BOOL	Low Low program acknowledge. Set by the user program to acknowledge a Low Low condition. Requires a cleared-to-set transition while the alarm condition is unacknowledged. Default is cleared.
LLOperAck	BOOL	Low Low operator acknowledge. Set by the operator interface to acknowledge a Low Low condition. Requires a cleared-to-set transition while the alarm condition is unacknowledged. The alarm instruction clears this parameter. Default is cleared.
ROCPosProgAck	BOOL	Positive rate of change program acknowledge. Set by the user program to acknowledge a positive rate-of-change condition. Requires a cleared-to-set transition while the alarm condition is unacknowledged. Default is cleared.
ROCPosOperAck	BOOL	Positive rate of change operator acknowledge. Set by the operator interface to acknowledge a positive rate-of-change condition. Requires a cleared-to-set transition while the alarm condition is unacknowledged. The alarm instruction clears this parameter. Default is cleared.
ROCNegProgAck	BOOL	Negative rate of change program acknowledge. Set by the user program to acknowledge a negative rate-of-change condition. Requires a cleared-to-set transition while the alarm condition is unacknowledged. Default is cleared.
ROCNegOperAck	BOOL	Negative rate of change operator acknowledge. Set by the operator interface to acknowledge a negative rate-of-change condition. Requires a cleared-to-set transition while the alarm condition is unacknowledged. The alarm instruction clears this parameter. Default is cleared.
ProgSuppress	BOOL	Set by the user program to suppress the alarm. Default is cleared.
OperSuppress	BOOL	Set by the operator interface to suppress the alarm. The alarm instruction clears this parameter. Default is cleared.
ProgUnsuppress	BOOL	Set by the user program to unsuppress the alarm. Takes precedence over Suppress commands. Default is cleared.
OperUnsuppress	BOOL	Set by the operator interface to unsuppress the alarm. Takes precedence over Suppress commands. The alarm instruction clears this parameter. Default is cleared.

Input Parameter	Data Type	Description
ProgDisable	BOOL	Set by the user program to disable the alarm. Default is cleared. **Relay Ladder** Copied from the instruction operand. **Structured Text** Copied from the instruction operand.
OperDisable	BOOL	Set by the operator interface to disable the alarm. The alarm instruction clears this parameter. Default is cleared.
ProgEnable	BOOL	Set by the user program to enable the alarm. Takes precedence over a Disable command. Default is cleared. **Relay Ladder** Copied from the instruction operand. **Structured Text** Copied from the instruction operand.
OperEnable	BOOL	Set by the operator interface to enable the alarm. Takes precedence over Disable command. The alarm instruction clears this parameter. Default is cleared.
AlarmCountReset	BOOL	Set by the user program to reset the alarm counts for all conditions. A cleared-to-set transition resets the alarm counts to zero. Default is cleared.
HHLimit	REAL	High High alarm limit. Valid = HLimit < HHLimit < maximum positive float. Default = 0.0.
HHSeverity	DINT	Severity of the High High alarm condition. This does not affect processing of alarms by the controller, but can be used for sorting and filtering functions at the alarm subscriber. Valid = 1…1000 (1000 = most severe; 1 = least severe). Default = 500.
HLimit	REAL	High alarm limit. Valid = LLimit < HLimit < HHLimit. Default = 0.0.

Input Parameter	Data Type	Description
HSeverity	DINT	Severity of the High alarm condition. This does not affect processing of alarms by the controller, but can be used for sorting and filtering functions at the alarm subscriber. Valid = 1…1000 (1000 = most severe; 1 = least severe). Default = 500.
LLimit	REAL	Low alarm limit. Valid = LLLimit < LLimit < HLimit. Default = 0.0.
LSeverity	DINT	Severity of the Low alarm condition. This does not affect processing of alarms by the controller, but can be used for sorting and filtering functions at the alarm subscriber. Valid = 1…1000 (1000 = most severe; 1 = least severe). Default = 500.
LLLimit	REAL	Low Low alarm limit. Valid = maximum negative float < LLLimit < LLimit. Default = 0.0.
LLSeverity	DINT	Severity of the Low Low alarm condition. This does not affect processing of alarms by the controller, but can be used for sorting and filtering functions at the alarm subscriber. Valid = 1…1000 (1000 = most severe; 1 = least severe). Default = 500.
MinDurationPRE	DINT	Minimum duration preset (milliseconds) for an alarm level condition to remain true before the condition is marked as InAlarm and alarm notification is sent to clients. The controller collects alarm data as soon as the alarm condition is detected, so no data is lost while waiting to meet the minimum duration. Does not apply to rate-of-change conditions. MinDurationPRE applies only to the first excursion from normal in either direction. For example, once the High condition times out, the High High condition will become active immediately, while a low condition will wait for the time-out period. Valid = 0…2,147,483,647. Default = 0.

Input Parameter	Data Type	Description
Deadband	REAL	Deadband for detecting that High High, High, Low, and Low Low alarm levels have returned to normal. A non-zero Deadband can reduce alarm condition chattering if the In value is continually changing but remaining near the level condition threshold. The Deadband value does not affect the transition to the InAlarm (active) state. Once a level condition is active, but before the condition will return to the inactive (normal) state, the In value must either: • drop below the threshold minus the deadband (for High and High High conditions). • rise above the threshold plus the deadband (for Low and Low Low conditions). The Deadband is not used to condition the Minimum Duration time measurement. Valid = 0 ≤ Deadband < Span from first enabled low alarm to the first enabled high alarm. Default = 0.0.
ROCPosLimit	REAL	Limit for an increasing rate-of-change in units per second. Detection is enabled for any value > 0.0 if ROCPeriod is also > 0.0. Valid = 0.0…maximum possible float. Default = 0.0.
ROCPosSeverity	DINT	Severity of the increasing rate-of-change condition. This does not affect processing of alarms by the controller, but can be used for sorting and filtering functions at the alarm subscriber. Valid = 1…1000 (1000 = most severe; 1 = least severe). Default = 500.
ROCNegLimit	REAL	Limit for a decreasing rate-of-change in units per second. Detection is enabled for any value > 0.0 if ROCPeriod is also > 0.0. Valid = 0.0…maximum possible float. Default = 0.0.
ROCNegSeverity	DINT	Severity of the decreasing rate-of-change condition. This does not affect processing of alarms by the controller, but can be used for sorting and filtering functions at the alarm subscriber. Valid = 1…1000 (1000 = most severe; 1 = least severe). Default = 500.
ROCPeriod	REAL	Time period in seconds for calculation (sampling interval) of the rate of change value. Each time the sampling interval expires, a new sample of In is stored, and ROC is recalculated. Rate-of-change detection is enabled for any value > 0.0. Valid = 0.0…maximum possible float. Default = 0.0.

Output Parameter	Data Type	Description
EnableOut	BOOL	Enable output.
InAlarm	BOOL	Alarm active status. Set when any alarm condition is active. Cleared when all alarm conditions are not active (normal status).
AnyInAlarmUnack	BOOL	Combined alarm active and acknowledged status. Set when any alarm condition is detected and unacknowledged. Cleared when all alarm conditions are normal (inactive), acknowledged, or both.
HHInAlarm	BOOL	High High alarm condition status. Set when a High High condition exists. Cleared when no High High condition exists.
HInAlarm	BOOL	High alarm condition status. Set when a High condition exists. Cleared when no High condition exists.
LInAlarm	BOOL	Low alarm condition status. Set when a Low condition exists. Cleared when no Low condition exists.
LLInAlarm	BOOL	Low Low alarm condition status. Set when a Low Low condition exists. Cleared when no Low Low condition exists.
ROCPosInAlarm	BOOL	Positive rate-of-change alarm condition status. Set when a positive rate-of-change condition exists. Cleared when no positive rate-of-change condition exists.
ROCNegInAlarm	BOOL	Negative rate-of-change alarm condition status. Set when a negative rate-of-change condition exists. Cleared when no negative rate-of-change condition exists.
ROC	REAL	Calculated rate-of-change of the In value. This value is updated when the instruction is scanned following each elapsed ROCPeriod. The ROC value is used to evaluate the ROCPosInAlarm and ROCNegInAlarm conditions. ROC = (current sample of In – previous sample of In) / ROCPeriod.
HHAcked	BOOL	High High condition acknowledged status. Set when a High High condition is acknowledged. Always set when AckRequired is cleared. Cleared when a High High condition is not acknowledged.
HAcked	BOOL	High condition acknowledged status. Set when a High condition is acknowledged. Always set when AckRequired is cleared. Cleared when a High condition is not acknowledged.
LAcked	BOOL	Low condition acknowledged status. Set when a Low condition is acknowledged. Always set when AckRequired is cleared. Cleared when a Low condition is not acknowledged.
LLAcked	BOOL	Low Low condition acknowledged status. Set when a Low Low condition is acknowledged. Always set when AckRequired is cleared. Cleared when a Low Low condition is not acknowledged.
ROCPosAcked	BOOL	Positive rate-of-change condition acknowledged status. Set when a positive rate-of-change condition is acknowledged. Always set when AckRequired is cleared. Cleared when a positive rate-of-change condition is not acknowledged.
ROCNegAcked	BOOL	Negative rate-of-change condition acknowledged status. Set when a negative rate-of-change condition is acknowledged. Always set when AckRequired is cleared. Cleared when a negative rate-of-change condition is not acknowledged.
HHInAlarmUnack	BOOL	Combined High High condition active and unacknowledged status. Set when the High High condition is active (HHInAlarm is set) and unacknowledged. Cleared when the High High condition is normal (inactive), acknowledged, or both.
HInAlarmUnack	BOOL	Combined High condition active and unacknowledged status. Set when the High condition is active (HInAlarm is set) and unacknowledged. Cleared when the High condition is normal (inactive), acknowledged, or both.

Output Parameter	Data Type	Description
LInAlarmUnack	BOOL	Combined Low condition active and unacknowledged status. Set when the Low condition is active (LInAlarm is set) and unacknowledged. Cleared when the Low condition is normal (inactive), acknowledged, or both.
LLInAlarmUnack	BOOL	Combined Low Low condition active and unacknowledged status. Set when the Low Low condition is active (LLInAlarm is set) and unacknowledged. Cleared when the Low Low condition is normal (inactive), acknowledged, or both.
ROCPosInAlarmUnack	BOOL	Combined positive rate-of-change condition active and unacknowledged status. Set when the positive rate-of-change condition is active (ROCPosInAlarm is set) and unacknowledged. Cleared when the positive rate-of-change condition is normal (inactive), acknowledged, or both.
ROCNegInAlarmUnack	BOOL	Combined negative rate-of-change condition active and unacknowledged status. Set when the negative rate-of-change condition is active (ROCNegInAlarm is set) and unacknowledged. Cleared when the negative rate-of-change condition is normal (inactive), acknowledged, or both.
Suppressed	BOOL	Suppressed status of the alarm. Set when the alarm is suppressed. Cleared when the alarm is not suppressed.
Disabled	BOOL	Disabled status of the alarm. Set when the alarm is disabled. Cleared when the alarm is enabled.
MinDurationACC	DINT	Elapsed time since an alarm condition was detected. When this value reaches MinDurationPRE, all detected alarm level conditions become active (*x*InAlarm is set), and a notification is sent to clients.
HHInAlarmTime	LINT	Timestamp when the ALMA instruction detected that the In value exceeded the High High condition limit for the most recent transition to the active state.
HHAlarmCount	DINT	The number of times the High High condition has been activated. If the maximum value is reached, the counter leaves the value at the maximum count value.
HInAlarmTime	LINT	Timestamp when the ALMA instruction detected that the In value exceeded the High condition limit for the most recent transition to the active state.
HAlarmCount	DINT	The number of times the High condition has been activated. If the maximum value is reached, the counter leaves the value at the maximum count value.
LInAlarmTime	LINT	Timestamp when the ALMA instruction detected that the In value exceeded the Low condition limit for the most recent transition to the active state.
LAlarmCount	DINT	The number of times the Low condition has been activated. If the maximum value is reached, the counter leaves the value at the maximum count value.
LLInAlarmTime	LINT	Timestamp when the ALMA instruction detected that the In value exceeded the Low Low condition limit for the most recent transition to the active state.
LLAlarmCount	DINT	The number of times the Low Low condition has been activated. If the maximum value is reached, the counter leaves the value at the maximum count value.
ROCPosInAlarmTime	LINT	Timestamp when the ALMA instruction detected that the In value exceeded the positive rate-of-change condition limit for the most recent transition to the active state.
ROCPosInAlarmCount	DINT	The number of times the positive rate-of-change condition has been activated. If the maximum value is reached, the counter leaves the value at the maximum count value.
ROCNegInAlarmTime	LINT	Timestamp when the ALMA instruction detected that the In value exceeded the negative rate-of-change condition limit for the most recent transition to the active state.
ROCNegAlarmCount	DINT	The number of times the negative rate-of-change condition has been activated. If the maximum value is reached, the counter leaves the value at the maximum count value.

Output Parameter	Data Type	Description
AckTime	LINT	Timestamp of most recent condition acknowledgement. If the alarm does not require acknowledgement, this timestamp is equal to most recent condition alarm time.
RetToNormalTime	LINT	Timestamp of alarm returning to a normal state.
AlarmCountResetTime	LINT	Timestamp indicating when the alarm count was reset.
DeliveryER	BOOL	Delivery error of the alarm notification message. Set when there is a delivery error, either no alarm subscriber was subscribed or at least one subscriber did not receive the latest alarm change state message. Cleared when delivery is successful or is in progress.
DeliveryDN	BOOL	Delivery completion of the alarm notification message. Set when delivery is successful, at least one subscriber was subscribed and all subscribers received the latest alarm change state message successfully. Cleared when delivery does not complete successfully or is in progress.
DeliveryEN	BOOL	Delivery status of the alarm notification message. Set when delivery is in progress. Cleared when delivery is not in progress.
NoSubscriber	BOOL	Alarm had no subscribers when attempting to deliver the most recent message. Set when there are no subscribers. Cleared when there is at least one subscriber.
NoConnection	BOOL	Alarm's subscribers were not connected when attempting to deliver the most recent message. Set when all subscribers are disconnected. Cleared when at least one subscriber is connected or there are no subscribers.
CommError	BOOL	Communication error when delivering an alarm message. Set when there are communication errors and all retries are used. This means that a subscriber was subscribed and it had a connection, but the controller did not receive confirmation of message delivery. Cleared when all connected subscribers confirm receipt of the alarm message.
AlarmBuffered	BOOL	Alarm message buffered due to a communication error (CommError is set) or a lost connection (NoConnection is set). Set when the alarm message is buffered for at least one subscriber. Cleared when the alarm message is not buffered.
Subscribers	DINT	Number of subscribers for this alarm.
SubscNotified	DINT	Number of subscribers successfully notified about the most recent alarm state change.
Status	DINT	Combined status indicators: Status.0 = InstructFault. Status.1 = InFaulted. Status.2 = SeverityInv. Status.3 = AlarmLimitsInv. Status.4 = DeadbandInv. Status.5 = ROCPosLimitInv. Status.6 = ROCNegLimitInv. Status.7 = ROCPeriodInv.
InstructFault (Status.0)	BOOL	Instruction error conditions exist. This is not a minor or major controller error. Check the remaining status bits to determine what occurred.

Output Parameter	Data Type	Description
InFaulted (Status.1)	BOOL	User program has set InFault to indicate bad quality input data. Alarm continues to evaluate In for alarm conditions.
SeverityInv (Status.2)	BOOL	Alarm severity configuration is invalid. If severity <1, the instruction uses Severity = 1. If severity >1000, the instruction uses Severity = 1000.
AlarmLimitsInv (Status.3)	BOOL	Alarm Limit configuration is invalid (for example, LLimit < LLLimit). If invalid, the instruction clears all level conditions active bits. Until the fault is cleared, no new level conditions can be detected.
DeadbandInv (Status.4)	BOOL	Deadband configuration is invalid. If invalid, the instruction uses Deadband = 0.0. Valid = 0 ≤Deadband < Span from first enabled low alarm to the first enabled high alarm.
ROCPosLimitInv (Status.5)	BOOL	Positive rate-of-change limit invalid. If invalid, the instruction uses ROCPosLimit = 0.0, which disables positive rate-of-change detection.
ROCNegLimitInv (Status.6)	BOOL	Negative rate-of-change limit invalid. If invalid, the instruction uses ROCNegLimit = 0.0, which disables negative rate-of-change detection.
ROCPeriodInv (Status.7)	BOOL	Rate-of-change period invalid. If invalid, the instruction uses ROCPeriod = 0.0, which disables rate-of-change detection.

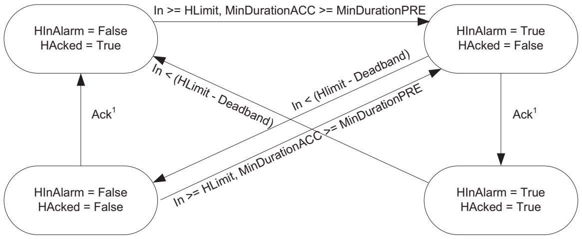
Description

The ALMA instruction detects alarms based on the level or rate of change of a value.

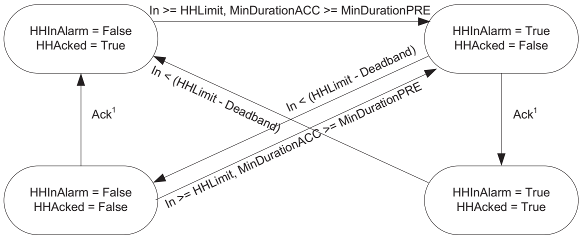
The ALMA instruction provides additional functionality when used with RSLinx Enterprise and FactoryTalk View SE software. You can display alarms in the Alarm Summary, Alarm Banner, Alarm Status Explorer, and Alarm Log Viewer displays in FactoryTalk View SE software.

RSLinx Enterprise software subscribes to alarms in the controller. Use the output parameters to monitor the instruction to see the alarm subscription status and to display alarm status changes. If a connection to RSLinx Enterprise software is lost, the controller can briefly buffer alarm data until the connection is restored.

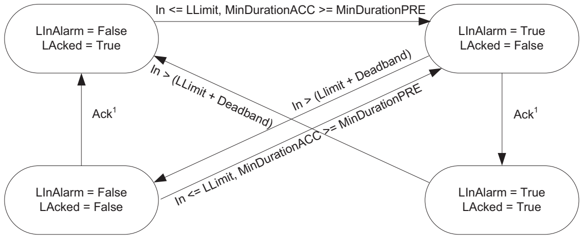
State Diagrams when Acknowledgement Required



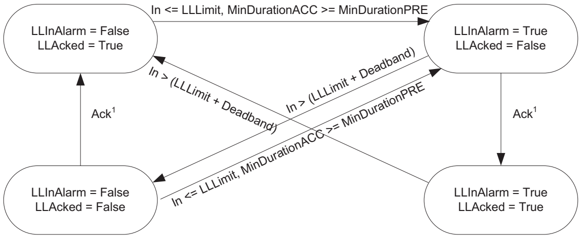
[1] H alarm condition can be acked by several different ways: HProgAck, HOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000 software, RSView software).



[1] HH alarm condition can be acked by several different ways: HHProgAck, HHOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000 software, RSView software).



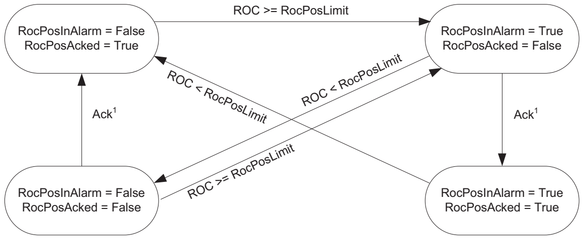
[1] L alarm condition can be acked by several different ways: LProgAck, LOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000 software, RSView software).



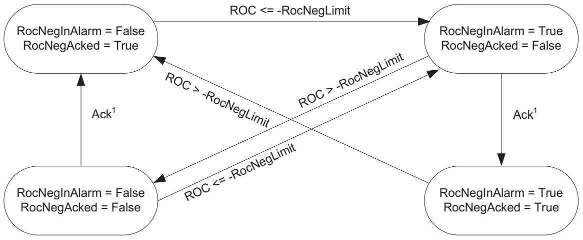
[1] LL alarm condition can be acked by several different ways: LLProgAck, LLOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000 software, RSView software).

$$\mathrm{ROC} = \frac{\mathrm{In(CurrentSample) - In(PreviousSample)}}{\mathrm{ROCPeriod}}$$

Where a new sample is collected on the next scan after the ROCPeriod has elapsed.

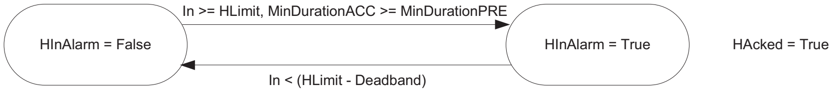


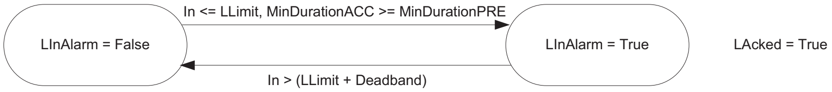
[1] ROCPos alarm condition can be acked by several different ways: RocPosProgAck, RocPosOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000 software, RSView software).

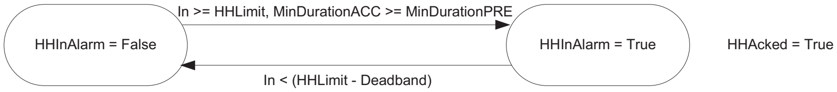


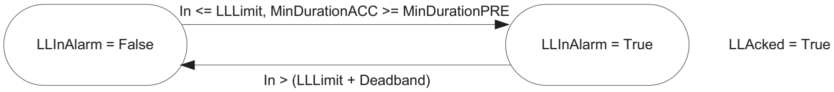
[1] ROCNeg alarm condition can be acked by several different ways: RocNegProgAck, RocNegOperAck, ProgAckAll, OperAckAll, clients (RSLogix 5000 software, RSView software).

State Diagrams when Acknowledgement Not Required



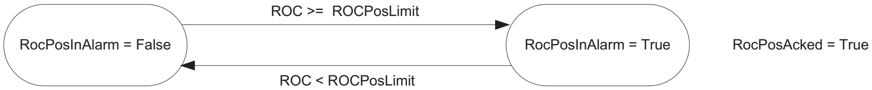


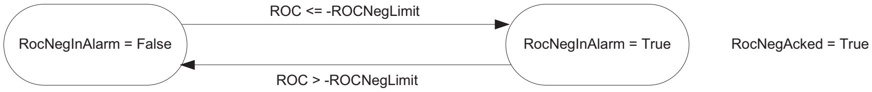




$$\mathrm{ROC} = \frac{\mathrm{In(CurrentSample) - In(PreviousSample)}}{\mathrm{ROCPeriod}}$$

Where a new sample is collected on the next scan after the ROCPeriod has elapsed.





Arithmetic Status Flags: Arithmetic status flags are set for the ROC output.

Fault Conditions:

Minor Fault	Fault Type	Fault Code
ROC overflow	4	4

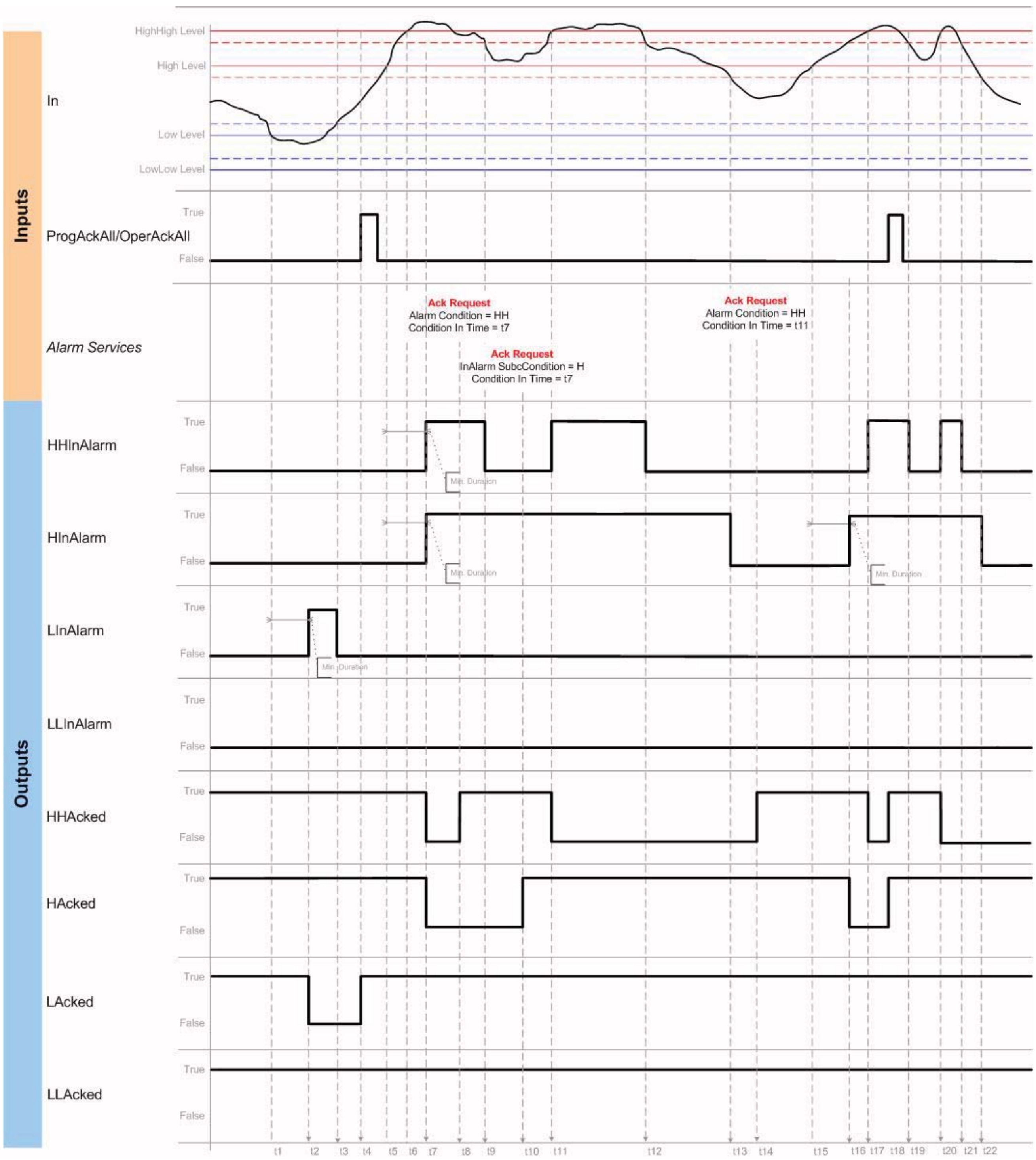
Execution:

Condition	Relay Ladder Action
Prescan	The rung-condition-out is set to false. All the *x*InAlarm parameters are cleared and all alarm conditions are acknowledged. All operator requests, timestamps, and delivery flags are cleared.
Rung-condition-in is false	The instruction does not execute. EnableOut is cleared.

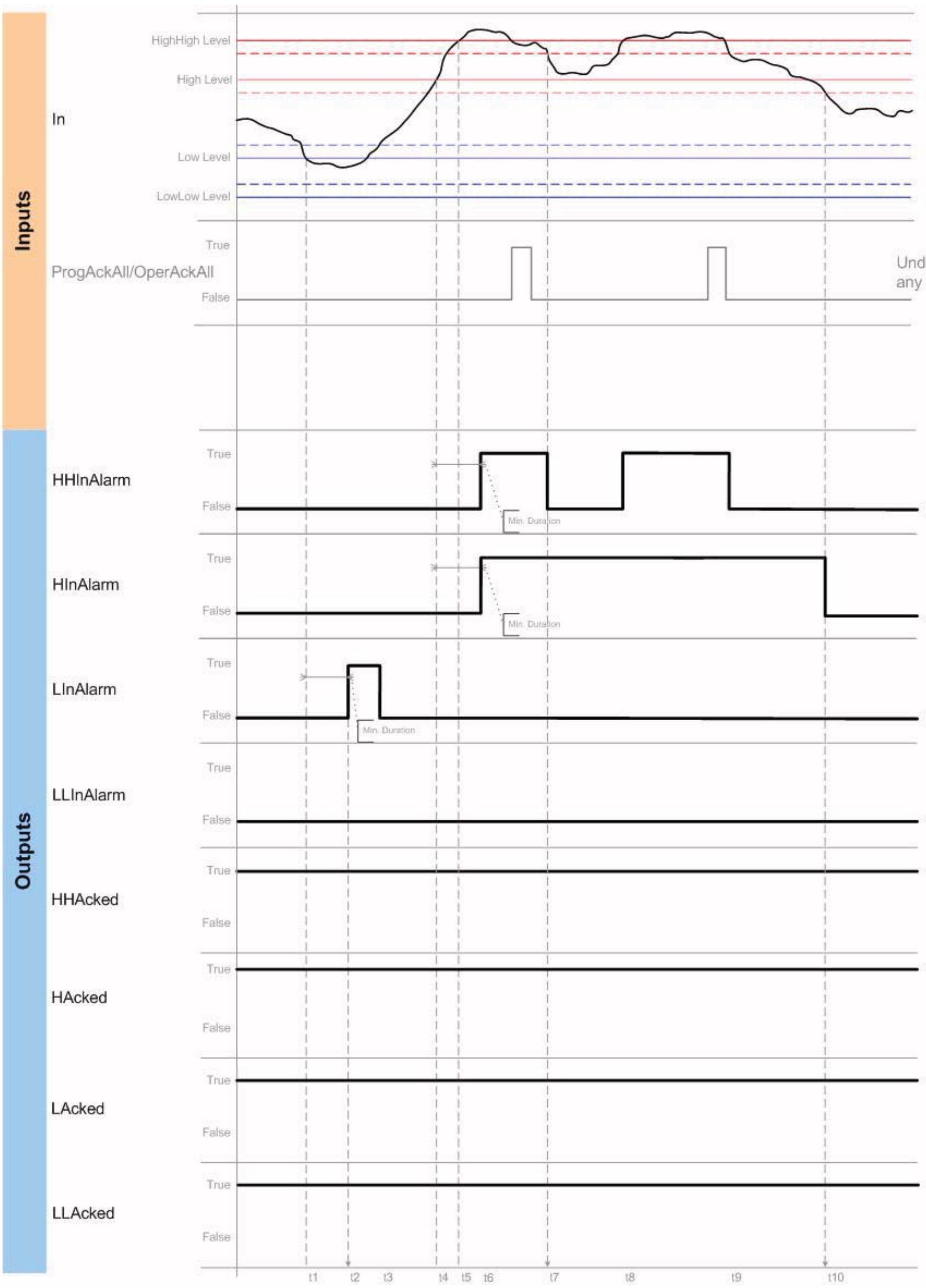
Condition	Relay Ladder Action
Rung-condition-in is true	The instruction executes. EnableOut is set.
Postscan	The rung-condition-out is set to false.

Condition	Function Block Action	Structured Text Action
prescan	All operator requests, timestamps, and delivery flags are cleared. All the *x*InAlarm parameters are cleared and all alarm conditions are acknowledged.	All operator requests, timestamps, and delivery flags are cleared. All the *x*InAlarm parameters are cleared and all alarm conditions are acknowledged.
instruction first scan	No action taken.	No action taken.
instruction first run	No action taken.	No action taken.
EnableIn is cleared	The instruction does not execute. EnableOut is cleared.	The instruction executes. EnableOut is always set.
EnableIn is set	The instruction executes. EnableOut is set.	The instruction executes. EnableOut is always set.
postscan	No action taken.	No action taken.

ALMA Level Condition Acknowledge Required

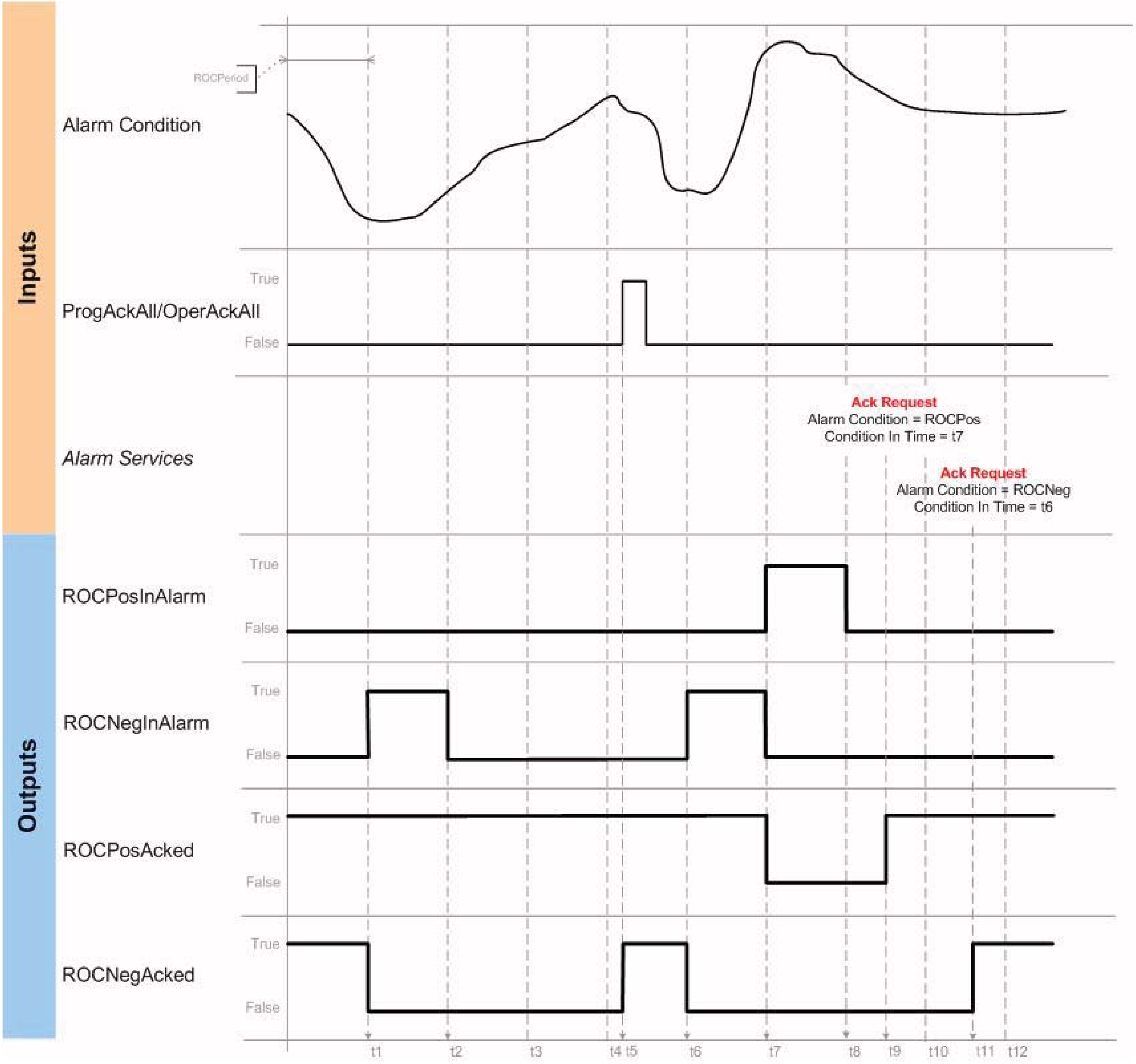


ALMA Level Condition Acknowledge Not Required

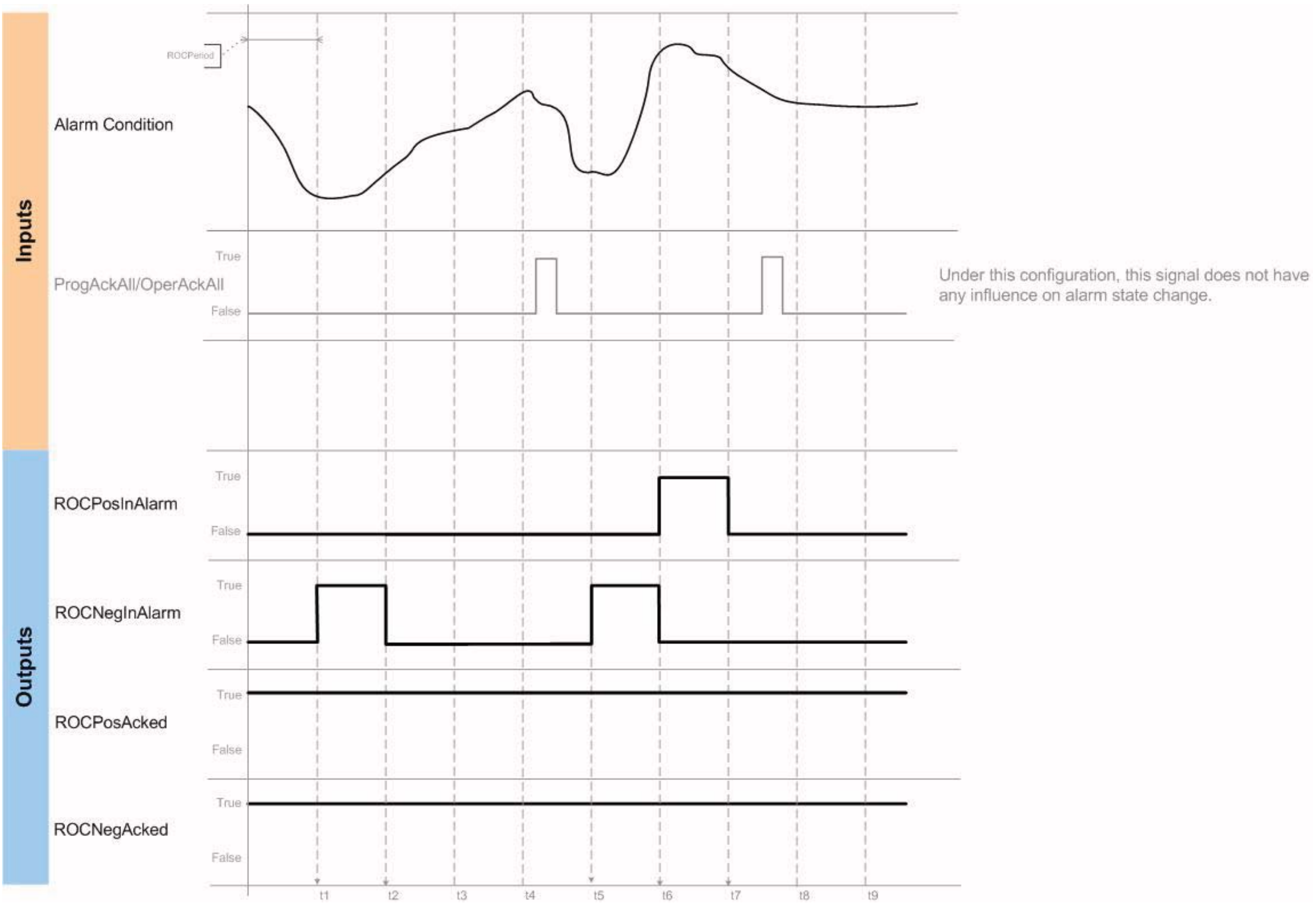


Under this configuration, this signal does not have any influence on alarm state change.

ALMA Rate of Change Acknowledge Required

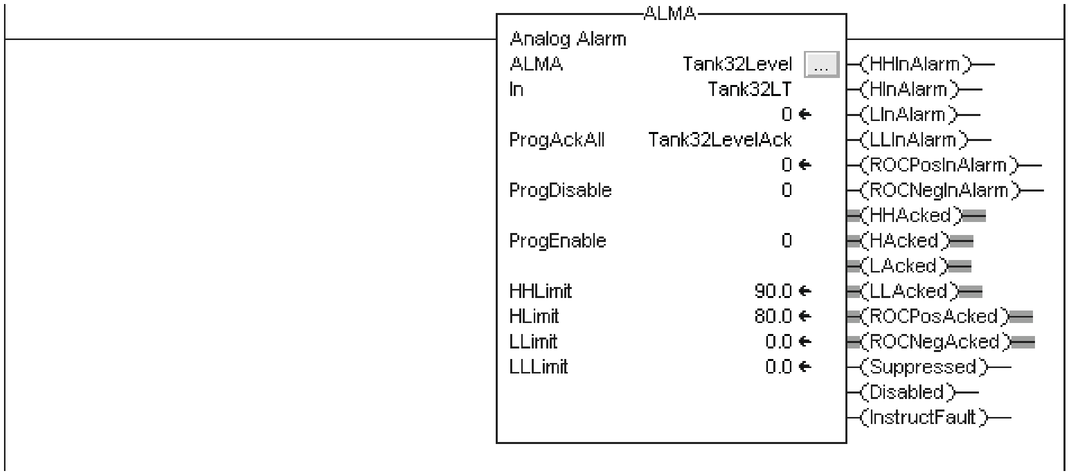


ALMA Rate of Change Acknowledge Not Required



Example: A tank alarm is activated if the tank level surpasses a High or High High limit. Programmatically acknowledge all the alarm conditions with a cleared-to-set transition of the Tank32LevelAck tag value. The application logic must clear Tank32LevelAck.

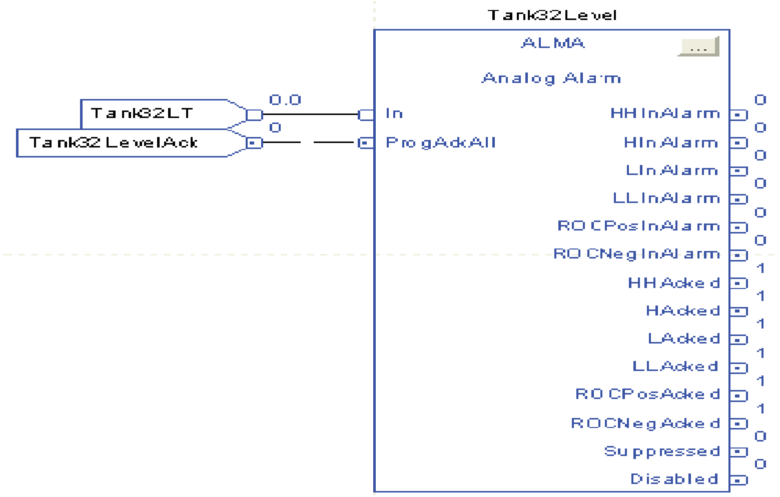
Relay Ladder



Structured Text

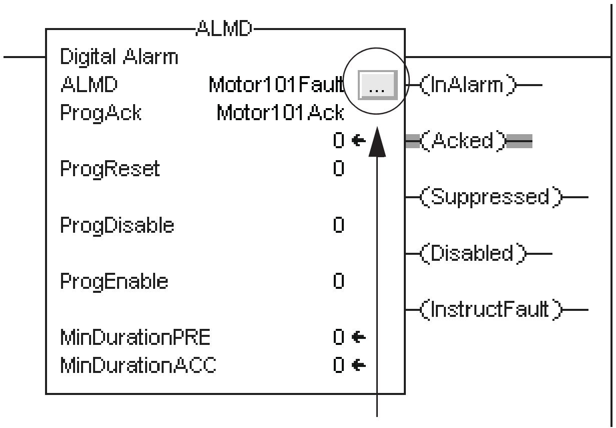
```
ALMA(Tank32Level,Tank32LT,Tank32LevelAck,0, 0);
```

Function Block



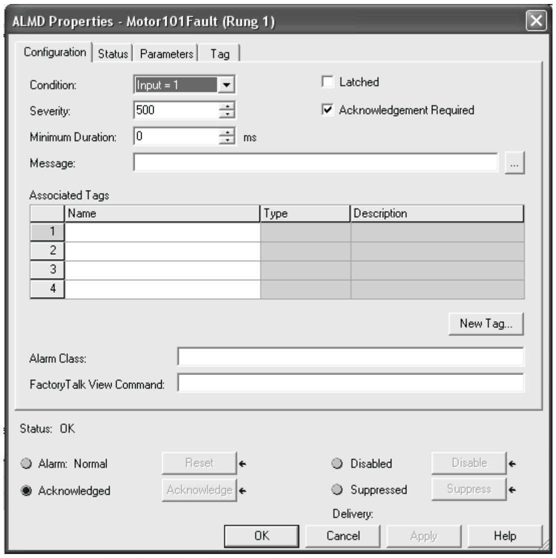
Configure an Alarm Instruction

After you enter an ALMD or ALMA instruction and specify the alarm tag name, use the Alarm Configuration dialog to specify the details of the message.



Click here to configure the instruction.

The Properties dialog for the alarm instruction includes a Configuration tab.



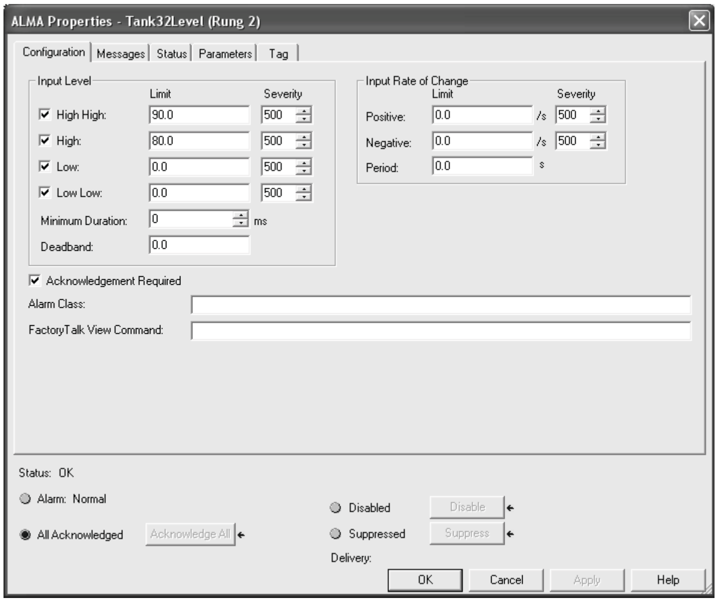
Condition: Input = 1
Severity: 500
Minimum Duration: 0 ms
Message:
☐ Latched
☑ Acknowledgement Required

Associated Tags

	Name	Type	Description
1			
2			
3			
4			

Alarm Class:
FactoryTalk View Command:

Status: OK
○ Alarm: Normal
● Acknowledged
○ Disabled
○ Suppressed
Delivery:



Input Level

	Limit	Severity
☑ High High:	90.0	500
☑ High:	80.0	500
☑ Low:	0.0	500
☑ Low Low:	0.0	500

Minimum Duration: 0 ms
Deadband: 0.0

Input Rate of Change

	Limit		Severity
Positive:	0.0	/s	500
Negative:	0.0	/s	500
Period:	0.0	s	

☑ Acknowledgement Required
Alarm Class:
FactoryTalk View Command:

Status: OK
○ Alarm: Normal
● All Acknowledged
○ Disabled
○ Suppressed
Delivery:

For each alarm instruction, configure this information.

Option	Description
Condition - ALMD instruction	Condition to trigger the alarm. Select Input=1 for an active alarm when In=1. Select Input=0 for an active alarm when In=0.
Input Level - ALMA instruction Input Rate of Change - ALMA instruction	Input Level (High High, High, Low, or Low Low) or Input Rate of Change (Positive or Negative) to trigger an alarm. Select the alarm conditions and enter the limits for those conditions. Disable rate-of-change conditions by entering a 0 for the period or limit.

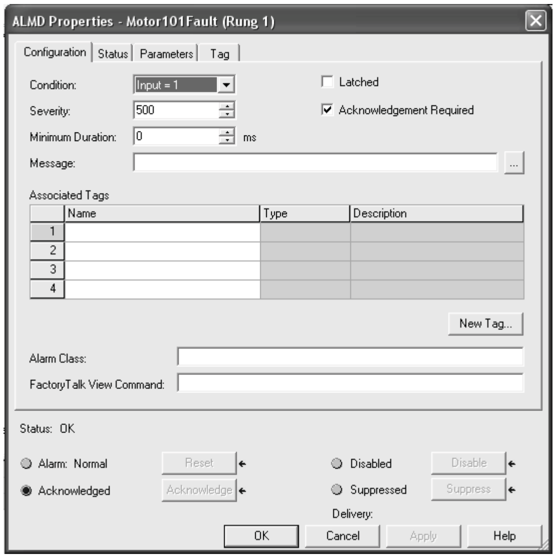
Option	**Description**
Severity	Select a severity range from 1…1000 to rank the importance of an alarm condition. A severity of 1 is for low priority alarms; a severity of 1000 is for an emergency condition. By default, in the FactoryTalk Alarms and Events system, severity ranges are mapped to priorities as follows: • 1…250 are low priority. • 251…500 are medium priority. • 501…750 are high priority. • 751…1000 are urgent priority. You can configure the severity-to-priority mapping in the FactoryTalk Alarms and Events system. See the FactoryTalk help for details.
Minimum Duration	Enter the amount of time in ms an alarm condition must be active before reporting the alarm.
Latched - ALMD instruction	Select Latched if you want the alarm to stay active (InAlarm) after the alarm condition returns to inactive (normal). Latched alarms require a reset command to transition to normal. The reset command must be received after the condition returns to normal. Acknowledge commands will not reset a latched alarm.
Deadband - ALMA instruction	Specify a Deadband value to reduce alarm condition chattering caused by small fluctuations in the In value. The deadband value does not affect the alarm limit for the transition into the active state, and is also not used during the Minimum Duration interval. Once a level condition becomes active (InAlarm), it will remain active until the In value crosses back over the limit by the specified deadband. For example, if the High limit is 80, the Low limit is 20, and the Deadband is 5, the High condition will be active at ≥ 80 and return to normal at ≤75; the Low condition will be active at ≤20 and return to normal at ≥ ≤25. The Deadband has no effect on Rate of Change alarm conditions.
Acknowledgement Required	Alarms are configured to require acknowledgement by default. Acknowledgement indicates that an operator is aware of the alarm condition, whether or not conditions have returned to normal. Clear the Acknowledgement Required setting when you want the alarm to appear and disappear from the Alarm Summary on the HMI with no operator interaction. Alarms that do not require acknowledgement always have the Acked status set. If a digital alarm is configured as latched, the reset command also acknowledges the alarm.

Option	**Description**
Alarm class	Use the alarm class to group related alarms. Specify the alarm class exactly the same for each alarm you want in the same class. The alarm class is case sensitive. For example, specify class Tank Farm A to group all the tank alarms for a specific area. Or specify class Control Loop to group all alarms for PID loops. You can then display and filter alarms at the HMI based on the class. For example, an operator can display all tank alarms or all PID loop alarms. The alarm class does not limit the alarms that an Alarm Summary object subscribes to. Use the alarm class to filter the alarms that display to an operator once they have been received by the Alarm Summary object. FactoryTalk View software can filter the alarm class substituting wild cards for characters.
View command	Execute a command on the operator station when requested by an operator for a specific alarm. This lets an operator execute any standard FactoryTalk View command, such as call specific faceplates and displays, execute macros, access help files, and launch external applications. When the alarm condition occurs and is displayed to the operator, a button on the summary and banner displays lets the operator run an associated view command. Be careful to enter the correct command syntax and test the command at runtime as there is no error checking performed when the command is entered.

You can edit all aspects of the alarm configuration offline and online. Online edits of new and existing alarms are immediately sent to FactoryTalk subscribers (legacy HMI terminals that are just polling the tags do not automatically update). FactoryTalk subscribers do not have to re-subscribe to receive updated information. Online changes automatically propagate from the controller alarm structure to the rest of the architecture.

Enter Alarm Message Text

Enter appropriate message text to display when an alarm condition is active (InAlarm). For an ALMD instruction, you enter the message information on the Configuration tab. For an ALMA instruction, you enter the message information on the Message tab.


ALMD Properties - Motor101Fault (Rung 1)

Condition: Input = 1

Severity: 500

Minimum Duration: 0 ms

☐ Latched

☑ Acknowledgement Required

Message:

Status: OK

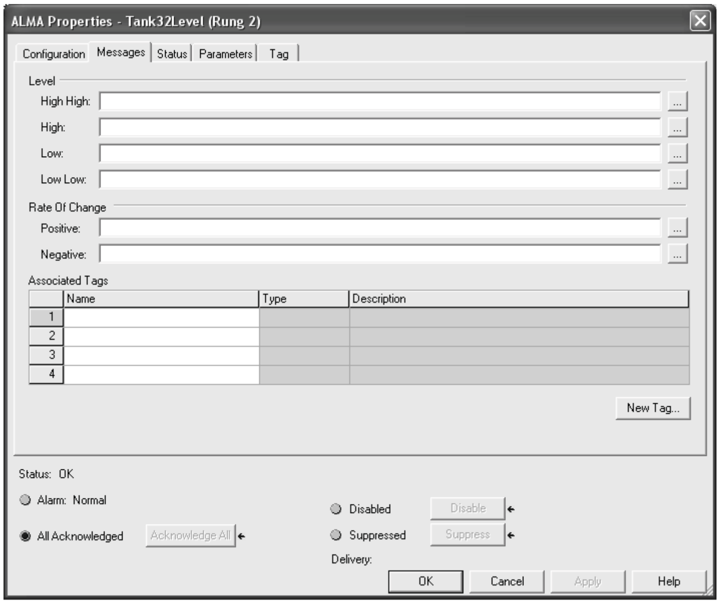
○ Alarm: Normal

● Acknowledged

○ Disabled

○ Suppressed

Delivery:


ALMA Properties - Tank32Level (Rung 2)

Level

High High:

High:

Low:

Low Low:

Rate Of Change

Positive:

Negative:

Status: OK

○ Alarm: Normal

● All Acknowledged

○ Disabled

○ Suppressed

Delivery:

To define an alarm message, specify this information.

Option	Description
Message string	The message string contains the information to display to the operator regarding the alarm. In addition to entering text, you can also embed variable information. In the alarm message editor, select the variable you want and add it anywhere in the message string. The message string can have a maximum of 255 characters, including the characters that specify any embedded variables (not the number of characters in the actual values of the embedded variables). For example, /*S:0 %Tag1*/ specifies a string tag and adds 13 characters towards the total string length, but the actual value of the string tag could contain 82 characters. You cannot programmatically access the alarm message string from the alarm tag. To change the alarm message based on specific events, configure one of the associated tags as a string data type and embed that associated tag in the message. You can have multiple language versions of messages. You enter the different language via the import/export utility. For more information, see page 67.

Option	Description
Associated tags	You can select as many as four additional tags from the controller project to associate with the alarm. The values of these tags are sent with an alarm message to the alarm server. For example, a digital alarm for a pressure relief valve might also include information such as pump speed and tank temperature. Associated tags may be any atomic data type (BOOL, DINT, INT, SINT, or REAL) or a STRING. They may be elements in a UDT or an Array. Variable array references are not allowed. If the alarm is controller-scoped, the associated tags must also be controller-scoped. Optionally, embed the associated tags into the message text string. Associated tag values are always sent with the alarm, viewable by the operator, and entered in the history log, regardless of whether you embed them in the message string.

Message String Variables

You can embed this variable information in a message string.

Variable	Embeds in the message string	Default code added to message string
Alarm name	The name of the alarm, which consists of the controller name, program name, and tag name. For example, [Zone1Controller]Program:Main.MyAlarmTagName.	`/*S:0 %AlarmName*/`
Condition name	The condition that triggers the alarm: • Digital alarm displays the trip. • Analog alarm displays HiHi, Hi, Lo, LoLo, ROC_POS, or ROC_NEG.	`/*S:0 %ConditionName*/`
Input value	The input value to the alarm: • Digital alarm displays 0 or 1. • Analog alarm displays the value of the input variable being monitored by the alarm.	`/*N:5 %InputValue NOFILL DP:0*/`
Limit value	The threshold of the alarm: • Digital alarm displays 0 or 1. • Analog alarm displays the actual configured range check for the analog alarm condition.	`/*N:5 %LimitValue NOFILL DP:0*/`
Severity	The configured severity of the alarm condition.	`/*N:5 %Severity NOFILL DP:0*/`
Values of associated tags	The value of a tag configured to be included with the alarm event.	`/*N:5 %Tag1 NOFILL DP:0*/`

The code varies depending on the type of tag you select, how many digits or characters are in a tag value, and whether you want to left fill the empty bits with spaces or zeroes. For example:

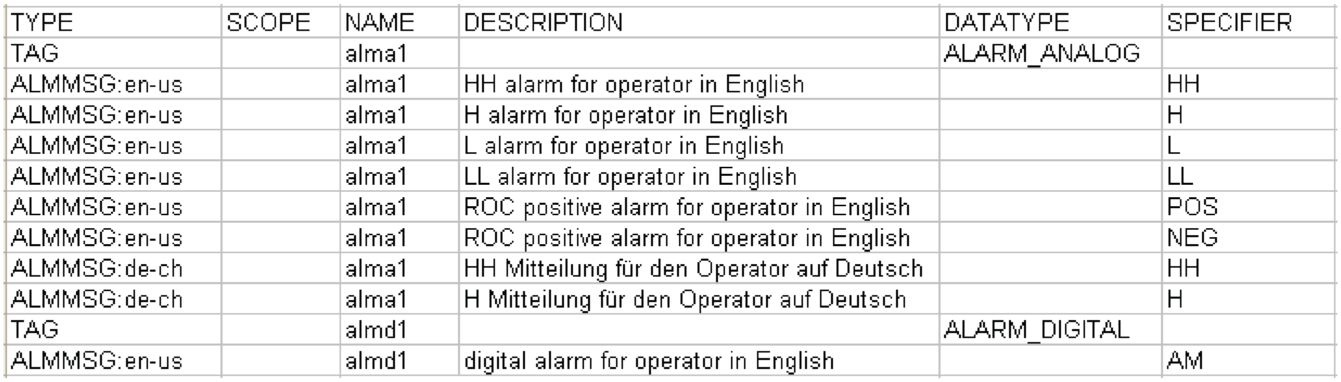
Tag	Code
BOOL value	`/*N:1 %Tag1 NOFILL DP:0*/`
DINT value, 9 digits, space left fill	`/*N:9 %Tag2 SPACEFILL DP:0*/`
REAL input value, 9 digits (includes decimal), 3 digits after decimal, zero left fill	`/*N:9 %InputValue NOFILL DP:3*/`
REAL value, 8 digits (includes decimal), 4 digits after decimal, zero left fill	`/*N:8 %Tag3 ZEROFILL DP:4*/`
String value, no fixed width	`/*S:0 %Tag4*/`
String value, 26 characters, fixed width	`/*S:26 %Tag4*/`

All of this variable information is included with the alarm data, viewable by the operator, and entered in the history log, regardless of whether you embed the information in the message text.

Multiple Language Versions of Alarm Messages

You can maintain alarm messages in multiple languages. Either enter the different languages in the associated language versions of RSLogix 5000 programming software or in an import/export (.CSV or .TXT) file.

You can access alarm message text from an import/export (.CSV or .TXT) file and add additional lines for translated versions of the original message string. Messages in different languages use ISO language codes in the TYPE column. Alarm message text, including embedded variable codes, for the operator is in the DESCRIPTION column. The SPECIFIER identifies the alarm condition.



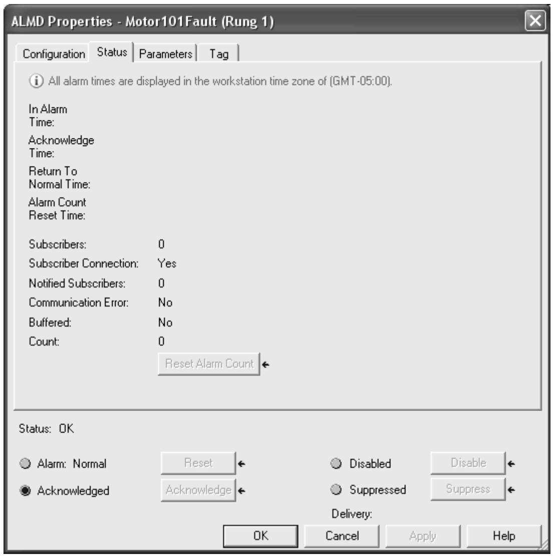
TYPE	SCOPE	NAME	DESCRIPTION	DATATYPE	SPECIFIER
TAG		alma1		ALARM_ANALOG	
ALMMSG:en-us		alma1	HH alarm for operator in English		HH
ALMMSG:en-us		alma1	H alarm for operator in English		H
ALMMSG:en-us		alma1	L alarm for operator in English		L
ALMMSG:en-us		alma1	LL alarm for operator in English		LL
ALMMSG:en-us		alma1	ROC positive alarm for operator in English		POS
ALMMSG:en-us		alma1	ROC positive alarm for operator in English		NEG
ALMMSG:de-ch		alma1	HH Mitteilung für den Operator auf Deutsch		HH
ALMMSG:de-ch		alma1	H Mitteilung für den Operator auf Deutsch		H
TAG		almd1		ALARM_DIGITAL	
ALMMSG:en-us		almd1	digital alarm for operator in English		AM

Use the import/export utility to create and translate message strings into multiple languages. The .TXT import/export format supports double-byte characters, so you can use this format for all languages, including Chinese, Japanese, and Korean. The .CSV import/export format does not support double-byte characters.

Importing and exporting messages always performs a merge. Deleting a message in a .CSV or .TXT file does not delete the message from the .ACD file. To delete a message, import the .CSV or .TXT file with the type, name, and specifier fields filled in but the description blank.

Monitor Alarm Status

On the Status tab of the alarm dialog, monitor the alarm condition, acknowledge an alarm, disable an alarm, suppress an alarm, or reset an alarm. Use the dialog selections to see how an alarm behaves, without needing an operational HMI.


ALMD Properties - Motor101Fault (Rung 1)

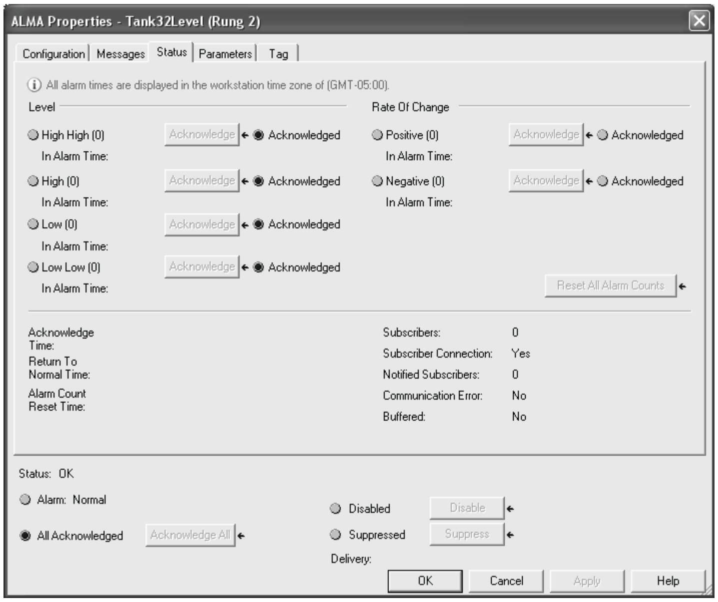
All alarm times are displayed in the workstation time zone of (GMT-05:00).

In Alarm Time:
Acknowledge Time:
Return To Normal Time:
Alarm Count Reset Time:

Subscribers: 0
Subscriber Connection: Yes
Notified Subscribers: 0
Communication Error: No
Buffered: No
Count: 0

Status: OK

○ Alarm: Normal
● Acknowledged
○ Disabled
○ Suppressed
Delivery:


ALMA Properties - Tank32Level (Rung 2)

All alarm times are displayed in the workstation time zone of (GMT-05:00).

Level
○ High High (0) ● Acknowledged
In Alarm Time:
○ High (0) ● Acknowledged
In Alarm Time:
○ Low (0) ● Acknowledged
In Alarm Time:
○ Low Low (0) ● Acknowledged
In Alarm Time:

Rate Of Change
○ Positive (0) ○ Acknowledged
In Alarm Time:
○ Negative (0) ○ Acknowledged
In Alarm Time:

Acknowledge Time:
Return To Normal Time:
Alarm Count Reset Time:

Subscribers: 0
Subscriber Connection: Yes
Notified Subscribers: 0
Communication Error: No
Buffered: No

Status: OK

○ Alarm: Normal
● All Acknowledged
○ Disabled
○ Suppressed
Delivery:

Buffer Alarms

In order to receive controller-based alarm messages, alarm clients (such as an RSLinx Enterprise server) must establish a subscription to the alarms in the Logix controller. The controller maintains a connection to each subscriber and monitors the status of that connection.

As alarm state changes occur, the alarm instructions in the controller cache the necessary information (such as timestamps and associated tag values) and request the transmission of the alarm message to all of the subscribers. The publisher mechanism delivers the alarm messages to each subscriber as quickly as possible.

If any subscriber fails to confirm receipt of the alarm message, or if the connection to a known subscriber is not good, the publisher mechanism stores the undelivered alarm messages in a 100 KB buffer. Each subscriber has its own buffer so communication problems with one subscriber do not interfere with alarm delivery to other subscribers. When the buffer is full, newer alarm messages are discarded. The buffer is created when the subscriber establishes its initial connection and is maintained for a configurable length of time (0…120 minutes, default is 20 minutes) after a subscriber loses its connection.

When the subscriber re-establishes a connection within the buffer time-out interval, it obtains the current state of all alarms, begins to receive current alarm messages, and also uploads any buffered messages that may have accumulated. Even if the buffer was full, and messages were discarded, the subscribers accurately synchronize to the current state of the alarms (including the most recent InAlarmTime, RetToNormalTime, and AckTime timestamps).

The buffer continues until is filled. Once filled, the buffer stops adding alarm transitions until space is made available in the buffer by the subscriber.

Programmatically Access Alarm Information

Each alarm instruction has an alarm structure that stores alarm configuration and execution information. The alarm structure includes both Program and Operator control elements and operator elements. The alarm instructions do not use mode settings to determine whether program access or operator access is active, so these elements are always active.

There are three ways to perform actions on an alarm instruction.

Access	Alarm Structure Elements	Considerations
User program	• ProgAck • ProgReset • ProgSuppress • ProgDisable • ProgEnable	Use controller logic to programmatically access elements of the alarming system. For example, the control program can determine whether to disable a series of alarms that are related to a single root cause. For example, the control program could disable an alarm instruction, MyDigitalAlarm of data type ALARM_DIGITAL, by accessing the tag member MyDigitalAlarm.ProgDisable.
Custom HMI	• OperAck • OperReset • OperSuppress • OperDisable • OperEnable	Create a custom HMI faceplate to access elements of the alarming system. For example, if the operator needs to remove a tool, rather than manually disable or suppress alarms individually from the alarming screens, the operator can press a disable key that accesses a tag MyDigitalAlarm.OperDisable. Operator parameters work with any Rockwell Automation or third-party operator interface to allow control of alarm states. When an operator parameter is set, the instruction evaluates whether it can respond to the request, then always resets the parameter.
Standard HMI object	Not accessible	Normal operator interaction is through the alarm summary, alarm banner, and alarm status explorer objects in the FactoryTalk View application. This interaction is similar to the custom HMI option described above, but there is no programmatic visibility or interaction.

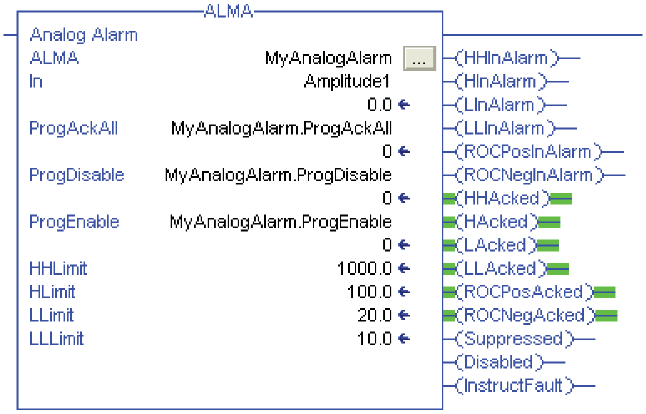
When you create an alarm instruction, you must create and assign a tag of the correct alarm data type for that alarm. For example, create MyDigitalAlarm of data type ALARM_DIGITAL. In relay ladder, these instruction parameters must be entered on the instruction:

- ProgAck
- ProgReset
- ProgDisable
- ProgEnable

In relay ladder and structured text, the value or tag you assign to an instruction parameter (such as ProgAck) is automatically written to the alarm tag member (such as MyAnalogAlarm.ProgAck) each time the instruction is scanned.

In relay ladder and structured text, if you want to programmatically access the alarm structure, assign the structure tag to the parameter on the instruction.

For example, to use MyAnalogAlarm.ProgAck in logic, assign the tag MyAnalogAlarm.ProgAck to the ProgAck parameter.



ALMA
Analog Alarm
ALMA MyAnalogAlarm
In Amplitude1
0.0
ProgAckAll MyAnalogAlarm.ProgAckAll
0
ProgDisable MyAnalogAlarm.ProgDisable
0
ProgEnable MyAnalogAlarm.ProgEnable
0
HHLimit 1000.0
HLimit 100.0
LLimit 20.0
LLLimit 10.0

Suppress or Disable Alarms

Suppress alarms to remove alarms you know exist from the HMI but still keep the alarms alive. This lets you clear the alarm summary while you are resolving a known alarm without continuing to view alarm information. A suppressed alarm does not appear on the operator summary or banner screens, but a suppressed alarm is still sent to subscribers, logged in the historical database, able to transition alarm status, time stamped, and responsive to other programmatic or operator interactions.

- When an alarm is Suppressed, it continues to function normally, monitor the In parameter for alarm conditions, and respond to Acknowledge requests. All subscribers are notified of this event, and any alarm messages generated while the alarm is in the Suppressed state include the Suppressed status. Alarm clients can respond differently to Suppressed alarms. For example, suppressed alarms can be logged to the historical database but not annunciated to the operator.
- When an alarm is Unsuppressed, all subscribers are notified and alarm messages to subscribers no longer include the Suppressed status.

Disable an alarm to treat the alarm as if it does not exist in the control program. A disabled alarm does not transition alarm status or get logged in the historical database. A disabled alarm is still tracked, and can be re-enabled, in the Alarm Status Explorer in FactoryTalk View SE software.

- When an alarm is Disabled, all of its conditions are set to the initial state (InAlarm is cleared and Acked is set). The In parameter is not monitored for alarm conditions. All subscribers are notified of this event.
- When an alarm is Enabled, it begins to monitor the In parameter for alarm conditions. All subscribers are notified of this event.

Controller-based Alarm Execution

Controller-based alarms process inputs from two sources.

Source	Description
Alarm tag members	Alarm tag members are, for the most part, processed when the user application scans the alarm instruction. This includes: • processing changes to configuration parameters. • evaluating the alarm condition. • measuring elapsed time for MinDuration. • capturing InAlarmTime and RetToNormalTime timestamps. • capturing associated tag values. • processing Prog and Oper commands. In addition, these alarm tag status members are updated as alarm messages are delivered to each subscriber, asynchronously to the program scan: • DeliveryEN, DeliveryER, DeliveryDN • NoSubscriber, NoConnection, CommError, AlarmBuffered, SubscNotified
Client messages	Client messages are processed as they are received, asynchronously to the program scan. • Reset, Acknowledge, Disable/Enable, and Suppress/Unsuppress commands from an RSLogix 5000 terminal • Reset, Acknowledge, Disable/Enable, and Suppress/Unsuppress commands from a FactoryTalk View SE alarm subscriber

Use care when determining where to place alarm instructions in the application. The accuracy of the timestamps are affected by how quickly the instruction is scanned after the alarm condition changes state. MinDuration time accumulation and Rate of Change calculations require repeated scanning, within time intervals determined by the user application. Alarm instructions must continue to be scanned after the alarm condition becomes false, so that the ReturnToNormal transition may be detected. For example, if you desire 10 ms accuracy on timestamps, you could place the alarm instructions that need that resolution in a 10 ms periodic task.

Controller Memory Use

As a guideline, use the following alarm sizes for a rough calculation of controller memory usage:

- Typically 1 KB per digital alarm with no associated tags

Digital Alarm Example	Approximate Size
Digital alarm with no associated tags and this configuration: • Alarm message: Contactor Fault • Alarm Class: Tank Farm A	1012 bytes
Digital alarm with two associated tags and this configuration: • Alarm message: Contactor Fault • Alarm Class: Tank Farm A • Associated Tag 1 = DINT data type • Associated Tag 2 = DINT data type	1100 bytes
Digital alarm with two associated tags and this configuration: • Alarm message: Contactor Fault • Alarm Class: Tank Farm A • Associated Tag 1 = DINT data type • Associated Tag 2 = STRING data type	1522 bytes

- Typically 2.2 KB per analog alarm with no associated tags

Analog Alarm Example	Approximate Size
Analog alarm with no associated tags and this configuration: • HH Alarm message: Level Alarm • H Alarm Message: Level Alarm • L Alarm Message: Level Alarm • LL Alarm Message: Level Alarm • Rate of Change Positive Message: Fill Too Fast • Rate of Change Negative Message: Empty Too Fast • Alarm Class: Tank Farm A	2228 bytes
Analog alarm with two associated tags and this configuration: • HH Alarm message: Level Alarm • H Alarm Message: Level Alarm • L Alarm Message: Level Alarm • LL Alarm Message: Level Alarm • Rate of Change Positive Message: Fill Too Fast • Rate of Change Negative Message: Empty Too Fast • Alarm Class: Tank Farm A • Associated Tag 1 = DINT data type • Associated Tag 2 = DINT data type	2604 bytes
Analog alarm with two associated tags and this configuration: • HH Alarm message: Level Alarm • H Alarm Message: Level Alarm • L Alarm Message: Level Alarm • LL Alarm Message: Level Alarm • Rate of Change Positive Message: Fill Too Fast • Rate of Change Negative Message: Empty Too Fast • Alarm Class: Tank Farm A • Associated Tag 1 = DINT data type • Associated Tag 2 = STRING data type	4536 bytes

Longer message strings, as well as message strings for multiple languages, consume additional memory from your controller.

Actual memory usage will depend on how the alarm is configured, message length, and any associated tags passed with the alarm.

Scan Time

These execution times show how ALMD instructions and ALMA instructions affect total scan time.

Rung State		**Execution Times**	
		Digital Alarm (ALMD)	**Analog Alarm (ALMA)**
No Alarm State Change	Rung False	8 μs	17 μs
	Rung True	8 μs	60 μs
Alarm State Change	Rung False	35 μs	17 μs
	Rung True	35 μs	126 μs

An alarm state change is any event that changes the condition of the alarm, such as acknowledging or suppressing the alarm. Minimize the potential for a large number of alarms changing state simultaneously (alarm bursts) by creating dependencies on related alarms. Large alarm bursts can have a significant impact on application code scan time.

Notes:

Bit Instructions

(XIC, XIO, OTE, OTL, OTU, ONS, OSR, OSF, OSRI, OSFI)

Introduction

Use the bit (relay-type) instructions to monitor and control the status of bits.

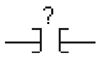
If you want to	Use this instruction	Available in	Page
Enable outputs when a bit is set	XIC	Relay ladder Structured text[1]	78
Enable outputs when a bit is cleared	XIO	Relay ladder Structured text[1]	81
Set a bit	OTE	Relay ladder Structured text[1]	84
Set a bit (retentive)	OTL	Relay ladder Structured text[1]	86
Clear bit (retentive)	OTU	Relay ladder Structured text[1]	88
Enable outputs for one scan each time a rung goes true	ONS	Relay ladder Structured text[1]	90
Set a bit for one scan each time a rung goes true	OSR	Relay ladder	93
Set a bit for one scan each time the rung goes false	OSF	Relay ladder	96
Set a bit for one scan each time the input bit is set in function block	OSRI	Structured text Function block	98
Set a bit for one scan each time the input bit is cleared in function block	OSFI	Structured text Function block	101

(1) There is no equivalent structured text instruction. Use other structured text programming to achieve the same result. See the description for the instruction.

Examine If Closed (XIC)

The XIC instruction examines the data bit to see if it is set.

Operands:



Relay Ladder

Operand	Type	Format	Description
Data bit	BOOL	Tag	Bit to be tested

Structured Text

Structured text does not have an XIC instruction, but you can achieve the same results by using an IF…THEN construct.

```
IF data_bit THEN

    <statement>;

END_IF;
```

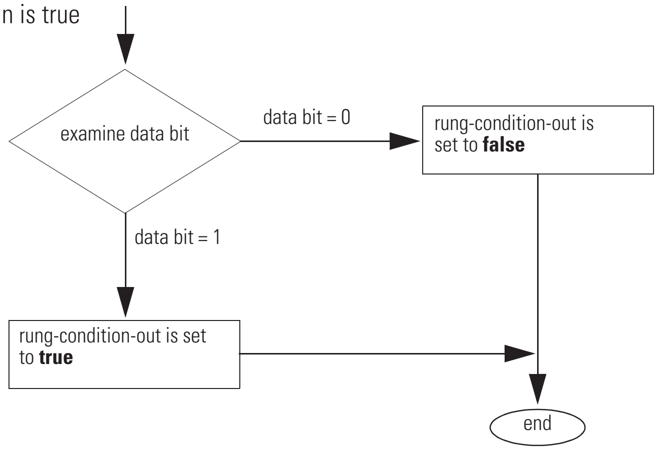
See Function Block Attributes for information on the syntax of constructs within structured text.

Description: The XIC instruction examines the data bit to see if it is set.

Arithmetic Status Flags: Not affected

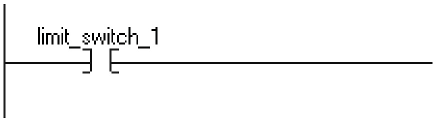
Fault Conditions: None

Execution:

Condition	Relay Ladder Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	
Postscan	The rung-condition-out is set to false.

Example 1: If *limit_switch_1* is set, this enables the next instruction (the rung-condition-out is true).

Relay Ladder



Structured Text

```
IF limit_switch THEN

    <statement>;

END_IF;
```

Example 2: If S:V is set (indicates that an overflow has occurred), this enables the next instruction (the rung-condition-out is true).

Relay Ladder

Structured Text

```
IF S:V THEN
    <statement>;
END_IF;
```

Examine If Open (XIO)

The XIO instruction examines the data bit to see if it is cleared.

Operands:

Relay Ladder

Operand	Type	Format	Description
Data bit	BOOL	Tag	Bit to be tested

Structured Text

Structured text does not have an XIO instruction, but you can achieve the same results by using an IF...THEN construct.

```
IF NOT data_bit THEN

    <statement>;

END_IF;
```

See Function Block Attributes for information on the syntax of constructs within structured text.

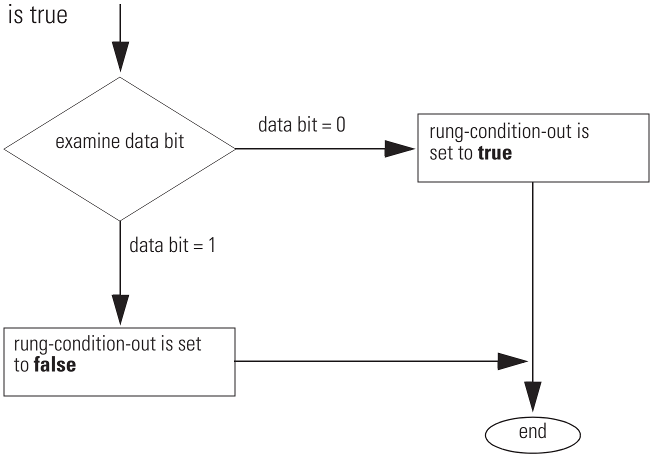
Description: The XIO instruction examines the data bit to see if it is cleared.

Arithmetic Status Flags: Not affected

Fault Conditions: None

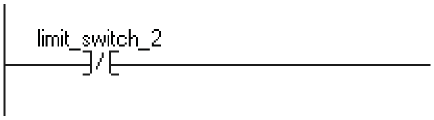
Execution:

Condition	Relay Ladder Action
Prescan	The rung-condition-out is set to false.

Condition	Relay Ladder Action
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	
Postscan	The rung-condition-out is set to false.

Example 1: If *limit_switch_2* is cleared, this enables the next instruction (the rung-condition-out is true).

Relay Ladder



Structured Text

```
IF NOT limit_switch_2 THEN

    <statement>;

END_IF;
```

Example 2: If S:V is cleared (indicates that no overflow has occurred), this enables the next instruction (the rung-condition-out is true).

Relay Ladder

Structured Text

```
IF NOT S:V THEN

    <statement>;

END_IF;
```

Output Energize (OTE)

The OTE instruction sets or clears the data bit.

Operands:

Relay Ladder

Operand	Type	Format	Description
Data bit	BOOL	Tag	Bit to be set or cleared

Structured Text

Structured text does not have an OTE instruction, but you can achieve the same results by using a non-retentive assignment.

```
data_bit [:=] BOOL_expression;
```

See Function Block Attributes for information on the syntax of assignments and expressions within structured text.

Description: When the OTE instruction is enabled, the controller sets the data bit. When the OTE instruction is disabled, the controller clears the data bit.

Arithmetic Status Flags: Not affected

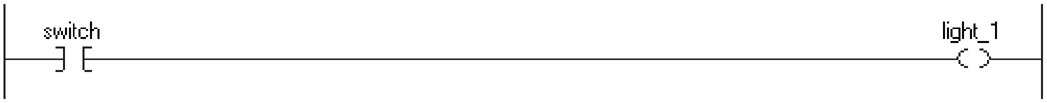
Fault Conditions: None

Execution:

Condition	Relay Ladder Action
Prescan	The data bit is cleared. The rung-condition-out is set to false.
Rung-condition-in is false	The data bit is cleared. The rung-condition-out is set to false.
Rung-condition-in is true	The data bit is set. The rung-condition-out is set to true.
Postscan	The data bit is cleared. The rung-condition-out is set to false.

Example: When switch is set, the OTE instruction sets (turns on) light_1. When *switch* is cleared, the OTE instruction clears (turns off) light_1.

Relay Ladder



Structured Text

```
light_1 [:=] switch;
```

Output Latch (OTL)

The OTL instruction sets (latches) the data bit.

Operands:

Relay Ladder

Operand	Type	Format	Description
Data bit	BOOL	Tag	Bit to be set

Structured Text

Structured text does not have an OTL instruction, but you can achieve the same results by using an IF...THEN construct and an assignment.

```
IF BOOL_expression THEN

    data_bit := 1;

END_IF;
```

See Function Block Attributes for information on the syntax of constructs, expressions, and assignments within structured text.

Description: When enabled, the OTL instruction sets the data bit. The data bit remains set until it is cleared, typically by an OTU instruction. When disabled, the OTL instruction does not change the status of the data bit.

Arithmetic Status Flags: Not affected

Fault Conditions: None

Execution:

Condition	Relay Ladder Action
Prescan	The data bit is not modified. The rung-condition-out is set to false.
Rung-condition-in is false	The data bit is not modified. The rung-condition-out is set to false.
Rung-condition-in is true	The data bit is set. The rung-condition-out is set to true.
Postscan	The data bit is not modified. The rung-condition-out is set to false.

Example: When enabled, the OTL instruction sets *light_2*. This bit remains set until it is cleared, typically by an OTU instruction.

Relay Ladder

Structured Text

```
IF BOOL_expression THEN

    light_2 := 1;

END_IF;
```

Output Unlatch (OTU)

The OTU instruction clears (unlatches) the data bit.

Operands:

Relay Ladder

Operand	Type	Format	Description
Data bit	BOOL	Tag	Bit to be cleared

Structured Text

Structured text does not have an OTU instruction, but you can achieve the same results by using an IF...THEN construct and an assignment.

```
IF BOOL_expression THEN

    data_bit := 0;

END_IF;
```

See Function Block Attributes for information on the syntax of constructs, expressions, and assignments within structured text.

Description:

When enabled, the OTU instruction clears the data bit. When disabled, the OTU instruction does not change the status of the data bit.

Arithmetic Status Flags:

Not affected

Fault Conditions:

None

Execution:

Condition	Relay Ladder Action
Prescan	The data bit is not modified. The rung-condition-out is set to false.
Rung-condition-in is false	The data bit is not modified. The rung-condition-out is set to false.
Rung-condition-in is true	The data bit is cleared. The rung-condition-out is set to true.
Postscan	The data bit is not modified. The rung-condition-out is set to false.

Example: When enabled, the OTU instruction clears *light_2*.

Relay Ladder

Structured Text

```
IF BOOL_expression THEN

    light_2 := 0;

END_IF;
```

One Shot (ONS)

The ONS instruction enables or disables the remainder of the rung, depending on the status of the storage bit.

Operands:

Relay Ladder

Operand	Type	Format	Description
Storage bit	BOOL	Tag	Internal storage bit. Stores the rung-condition-in from the last time the instruction was executed

Structured Text

Structured text does not have an ONS instruction, but you can achieve the same results by using an IF...THEN construct.

```
IF BOOL_expression AND NOT storage_bit THEN

    <statement>;

END_IF;

storage_bit := BOOL_expression;
```

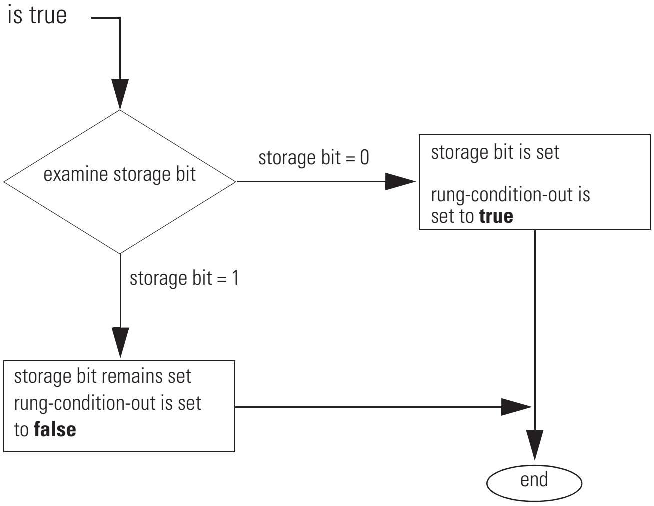
See Function Block Attributes for information on the syntax of constructs, expressions, and expressions within structured text.

Description: When enabled and the storage bit is cleared, the ONS instruction enables the remainder of the rung. When disabled or when the storage bit is set, the ONS instruction disables the remainder of the rung.

Arithmetic Status Flags: Not affected

Fault Conditions: None

Execution:

Condition	**Relay Ladder Action**
Prescan	The storage bit is set to prevent an invalid trigger during the first scan. The rung-condition-out is set to false.
Rung-condition-in is false	The storage bit is cleared. The rung-condition-out is set to false.
Rung-condition-in is true	
Postscan	The storage bit is cleared. The rung-condition-out is set to false.

Example: You typically precede the ONS instruction with an input instruction because you scan the ONS instruction when it is enabled and when it is disabled for it to operate correctly. Once the ONS instruction is enabled, the rung-condition-in must go clear or the storage bit must be cleared for the ONS instruction to be enabled again.

On any scan for which *limit_switch_1* is cleared or *storage_1* is set, this rung has no affect. On any scan for which *limit_switch_1* is set and *storage_1* is cleared, the ONS instruction sets storage_1 and the ADD instruction increments sum by 1. As long as *limit_switch_1* stays set, sum stays the same value. The *limit_switch_1* must go from cleared to set again for sum to be incremented again.

Relay Ladder

Structured Text

```
IF limit_switch_1 AND NOT storage_1 THEN

    sum := sum + 1;

END_IF;

storage_1 := limit_switch_1;
```

One Shot Rising (OSR)

The OSR instruction sets or clears the output bit, depending on the status of the storage bit.

This instruction is available in structured text and function block as OSRI, see page 98.

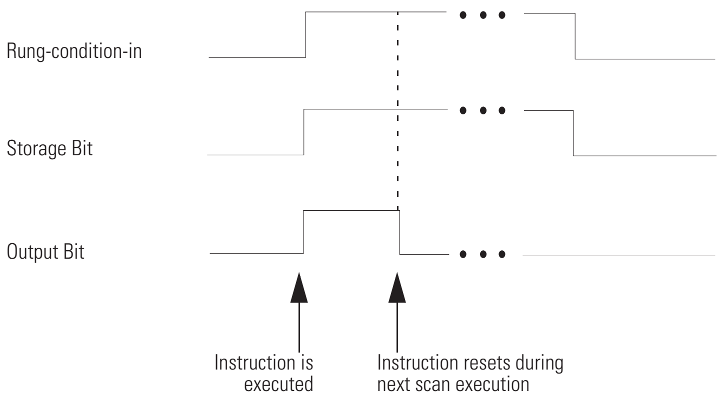
Operands:

Relay Ladder

Operand	Type	Format	Description
Storage bit	BOOL	Tag	Internal storage bit Stores the rung-condition-in from the last time the instruction was executed
Output bit	BOOL	Tag	Bit to be set

Description:

When enabled and the storage bit is cleared, the OSR instruction sets the output bit. When enabled and the storage bit is set or when disabled, the OSR instruction clears the output bit



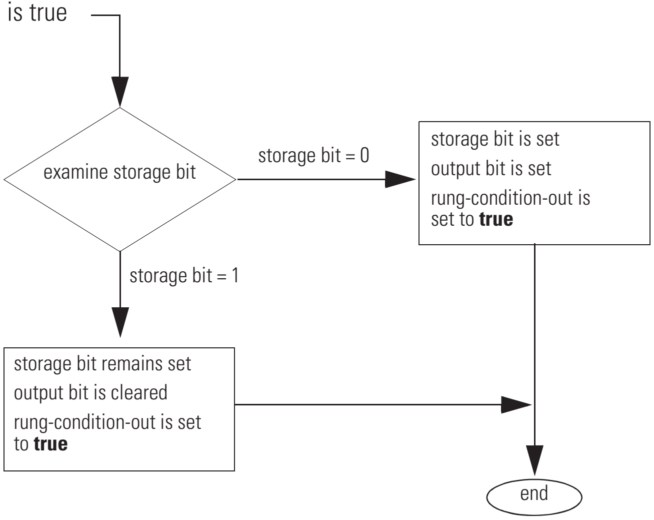
Arithmetic Status Flags:

Not affected

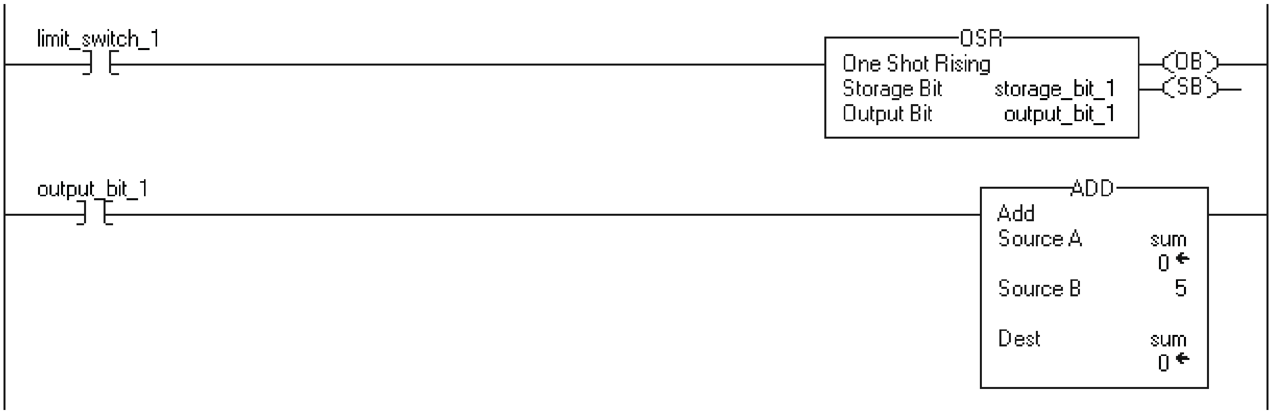
Fault Conditions:

None

Execution:

Condition	Relay Ladder Action
Prescan	The storage bit is set to prevent an invalid trigger during the first scan. The output bit is cleared. The rung-condition-out is set to false.
Rung-condition-in is false	The storage bit is cleared. The output bit is not modified. The rung-condition-out is set to false.
Rung-condition-in is true	
Postscan	The storage bit is cleared. The output bit is not modified. The rung-condition-out is set to false.

Example: Each time *limit_switch_1* goes from cleared to set, the OSR instruction sets *output_bit_1* and the ADD instruction increments sum by five. As long as *limit_switch_1* stays set, sum stays the same value. The *limit_switch_1* must go from cleared to set again for sum to be incremented again. You can use *output_bit_1* on multiple rungs to trigger other operations



One Shot Falling (OSF)

The OSF instruction sets or clears the output bit depending on the status of the storage bit.

This instruction is available in structured text and function block as OSFI, see page 101.

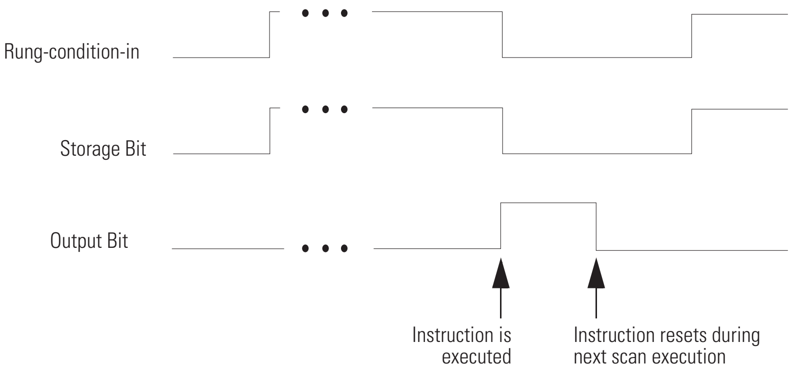
Operands:

Relay Ladder Operands

Operand	Type	Format	Description
Storage bit	BOOL	Tag	Internal storage bit Stores the rung-condition-in from the last time the instruction was executed
Output bit	BOOL	Tag	Bit to be set

Description:

When disabled and the storage bit is set, the OSF instruction sets the output bit. When disabled and the storage bit is cleared, or when enabled, the OSF instruction clears the output bit.



Arithmetic Status Flags:

Not affected

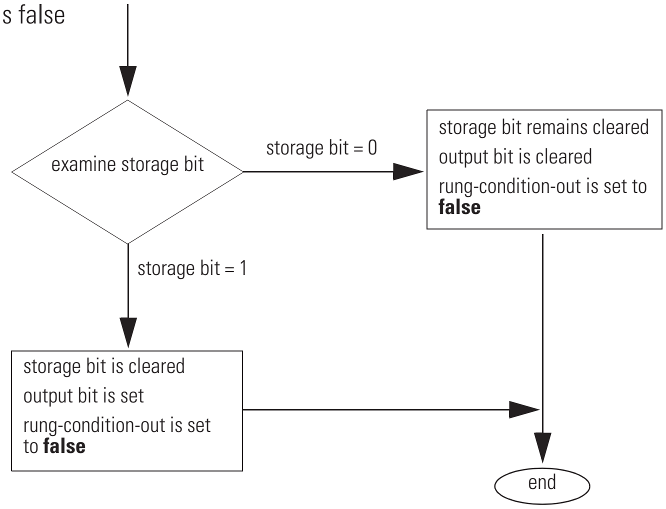
Fault Conditions:

None

Execution:

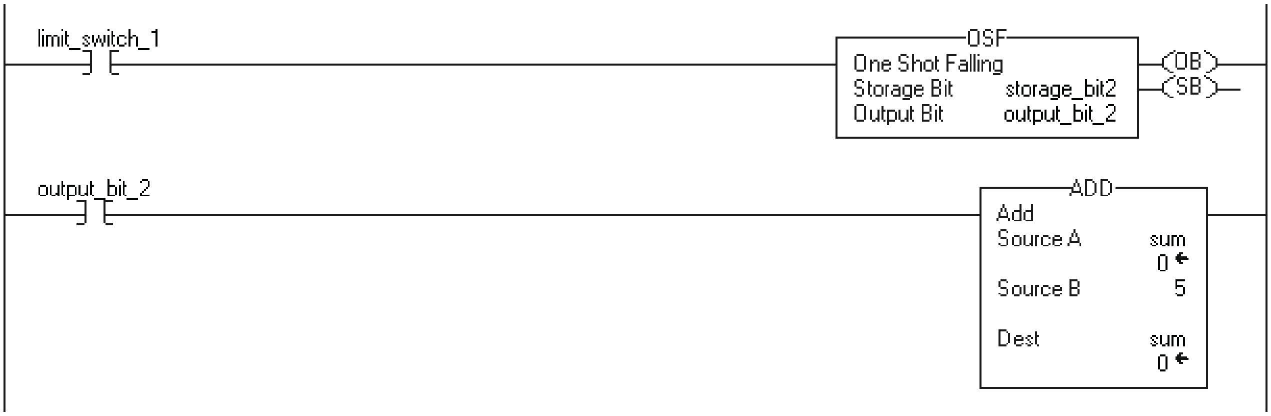
Condition	Relay Ladder Action
Prescan	The storage bit is cleared to prevent an invalid trigger during the first scan. The output bit is cleared. The rung-condition-out is set to false.

Rung-condition-in is false



Rung-condition-in is true	The storage bit is set. The output bit is cleared. The rung-condition-out is set to true.
Postscan	See rung-condition-in is false above.

Example: Each time *limit_switch_1* goes from set to cleared, the OSF instruction sets *output_bit_2* and the ADD instruction increments sum by 5. As long as *limit_switch_1* stays cleared, sum stays the same value. The *limit_switch_1* must go from set to cleared again for sum to be incremented again. You can use *output_bit_2* on multiple rungs to trigger other operations.



One Shot Rising with Input (OSRI)

The OSRI instruction sets the output bit for one execution cycle when the input bit toggles from cleared to set.

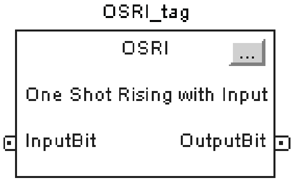
This instruction is available in relay ladder as OSR, see page 93.

Operands:

```
OSRI(OSRI_tag);
```

Structured Text

Operand	Type	Format	Description
OSRI tag	FBD_ONESHOT	Structure	OSRI structure



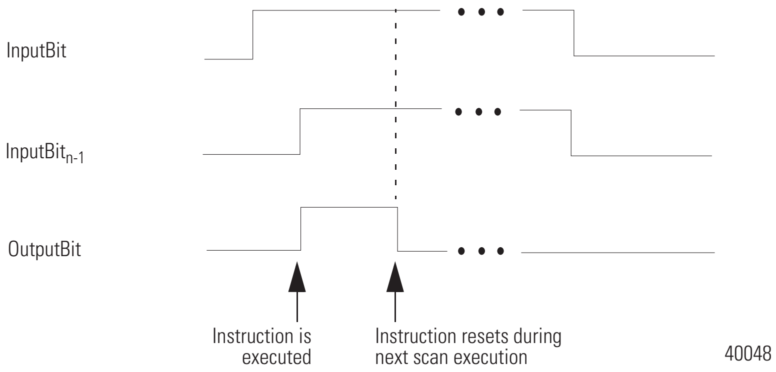
Function Block

Operand	Type	Format	Description
OSRI tag	FBD_ONESHOT	Structure	OSRI structure

FBD_ONESHOT Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	**Function Block** If cleared, the instruction does not execute and outputs are not updated. If set, the instruction executes. Default is set. **Structured Text** No effect. The instruction executes.
InputBit	BOOL	Input bit. This is equivalent to rung condition for the relay ladder OSR instruction. Default is cleared.
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
OutputBit	BOOL	Output bit

Description: When InputBit is set and InputBit_{n-1} is cleared, the OSRI instruction sets OutputBit. When InputBit_{n-1} is set or when InputBit is cleared, the OSRI instruction clears OutputBit.



Arithmetic Status Flags: Not affected

Fault Conditions: None

Execution:

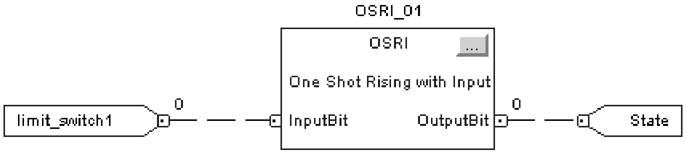
Condition	Function Block Action	Structured Text Action
Prescan	No action taken.	No action taken.
Instruction first scan	InputBit $_{n-1}$ is set.	InputBit $_{n-1}$ is set.
Instruction first run	InputBit $_{n-1}$ is set.	InputBit $_{n-1}$ is set.
EnableIn is cleared	EnableOut is cleared, the instruction does nothing, and the outputs are not updated.	N/A
EnableIn is set	On a cleared to set transition of InputBit, the instruction sets InputBit $_{n-1}$. The instruction executes. EnableOut is set.	On a cleared to set transition of InputBit, the instruction sets InputBit $_{n-1}$. EnableIn is always set. The instruction executes.
Postscan	No action taken.	No action taken.

Example: When *limit_switch1* goes from cleared to set, the OSRI instruction sets OutputBit for one scan.

Structured Text

```
OSRI_01.InputBit := limit_switch1;


OSRI(OSRI_01);


State := OSRI_01.OutputBit;
```

Function Block



One Shot Falling with Input (OSFI)

The OSFI instruction sets the OutputBit for one execution cycle when the InputBit toggles from set to cleared.

This instruction is available in relay ladder as OSF, see page 96.

Operands:

```
OSFI(OSFI_tag);
```

Structured Text

Operand	Type	Format	Description
OSFI tag	FBD_ONESHOT	Structure	OSFI structure

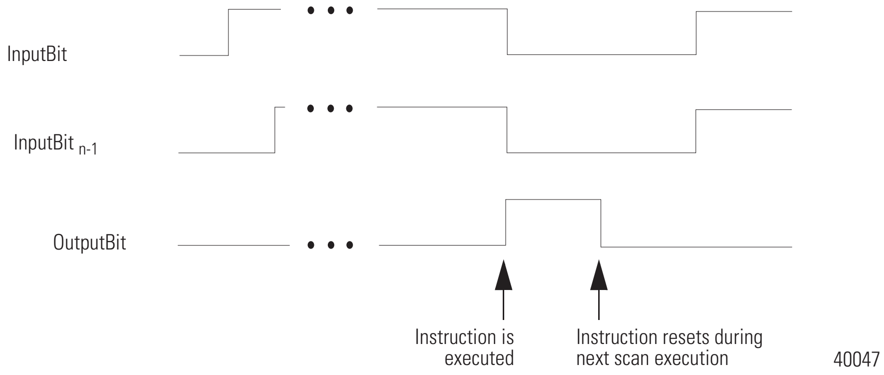
Function Block

Operand	Type	Format	Description
OSFI tag	FBD_ONESHOT	Structure	OSFI structure

FBD_ONESHOT Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	**Function Block** If cleared, the instruction does not execute and outputs are not updated. If set, the instruction executes. Default is set. **Structured Text** No effect. The instruction executes.
InputBit	BOOL	Input bit. This is equivalent to rung condition for the relay ladder OSF instruction Default is cleared.
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
OutputBit	BOOL	Output bit

Description: When the InputBit is cleared and the InputBit $_{n-1}$ is set, the OSFI instruction sets the OutputBit. When InputBit $_{n-1}$ is cleared or when InputBit is set, the OSFI instruction clears the OutputBit.



Arithmetic Status Flags: Not affected

Fault Conditions: None

Execution:

Condition	Function Block Action	Structured Text Action
Prescan	No action taken.	No action taken.
Instruction first scan	InputBit $_{n-1}$ is cleared.	InputBit $_{n-1}$ is cleared.
Instruction first run	InputBit $_{n-1}$ is cleared.	InputBit $_{n-1}$ is cleared.
EnableIn is cleared	EnableOut is cleared, the instruction does nothing, and the outputs are not updated.	na
EnableIn is set	On a cleared to set transition of InputBit, the instruction clears InputBit $_{n-1}$. The instruction executes. EnableOut is set.	On a cleared to set transition of InputBit, the instruction clears InputBit $_{n-1}$. EnableIn is always set. The instruction executes.
Postscan	No action taken.	No action taken.

Example: When *limit_switch1* goes from set to cleared, the OSFI instruction sets OutputBit for one scan.

Structured Text

```
OSFI_01.InputBit := limit_switch1;


OSFI(OSFI_01);


Output_state := OSFI_01.OutputBit;
```

Function Block

Notes:

Timer and Counter Instructions

(TON, TOF, RTO, TONR, TOFR, RTOR, CTU, CTD, CTUD, RES)

Introduction

Timers and counters control operations based on time or the number of events.

If you want to	Use this instruction	Available in these languages	Page
Time how long a timer is enabled	TON	Relay ladder	106
Time how long a timer is disabled	TOF	Relay ladder	110
Accumulate time	RTO	Relay ladder	114
Time how long a timer is enabled with built-in reset in function block	TONR	Structured text Function block	118
Time how long a timer is disabled with built-in reset in function block	TOFR	Structure text Function block	122
Accumulate time with built-in reset in function block	RTOR	Structured text Function block	126
Count up	CTU	Relay ladder	130
Count down	CTD	Relay ladder	134
Count up and count down in function block	CTUD	Structured text Function block	138
Reset a timer or counter	RES	Relay ladder	143

The time base for all timers is 1 ms.

Timer On Delay (TON)

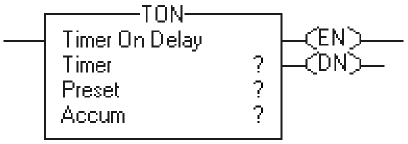
The TON instruction is a non-retentive timer that accumulates time when the instruction is enabled (rung-condition-in is true).

This instruction is available in structured text and function block as TONR.

See page 118.

Operands:

Relay Ladder



Operand	Type	Format	Description
Timer	TIMER	Tag	Timer structure
Preset	DINT	Immediate Tag	How long to delay (accumulate time)
Accum	DINT	Immediate Tag	Total milliseconds the timer has counted Initial value is typically 0

TIMER Structure

Mnemonic	Data Type	Description
.EN	BOOL	The enable bit indicates that the TON instruction is enabled.
.TT	BOOL	The timing bit indicates that a timing operation is in process.
.DN	BOOL	The done bit is set when .ACC ≥ .PRE.
.PRE	DINT	The preset value specifies the value (1 ms units) that the accumulated value must reach before the instruction sets the .DN bit.
.ACC	DINT	The accumulated value specifies the number of milliseconds that have elapsed since the TON instruction was enabled.

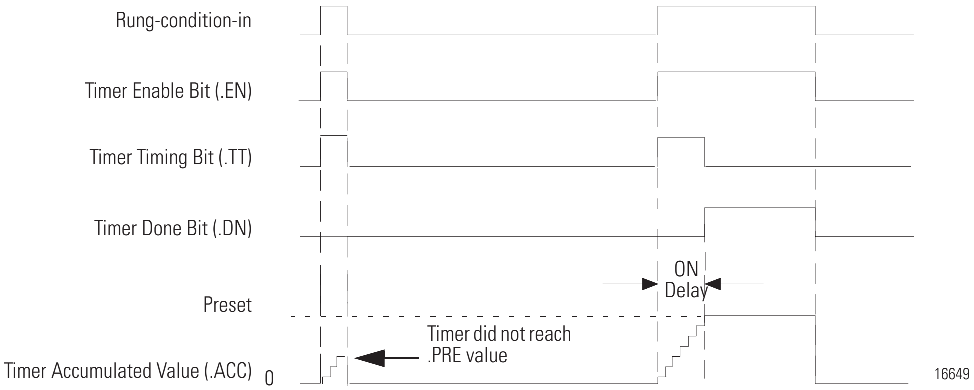
Description:

The TON instruction accumulates time until:

- the TON instruction is disabled.
- the .ACC ≥ .PRE.

The time base is always 1 ms. For example, for a two-second timer, enter 2000 for the .PRE value.

When the TON instruction is disabled, the .ACC value is cleared.



A timer runs by subtracting the time of its last scan from the time now:

```
ACC = ACC + (current_time - last_time_scanned)
```

After it updates the ACC, the timer sets last_time_scanned = current_time. This gets the timer ready for the next scan.

IMPORTANT

Make sure to scan the timer at least every 69 minutes while it runs. Otherwise, the ACC value won't be correct.

The *last_time_scanned* value has a range of up to 69 minutes. The timer's calculation rolls over if you don't scan the timer within 69 minutes. The ACC value won't be correct if this happens.

While a timer runs, scan it within 69 minutes if you put it in a:

- subroutine.
- section of code that is between JMP and LBL instructions.
- sequential function chart (SFC).
- event or periodic task.
- state routine of a phase.

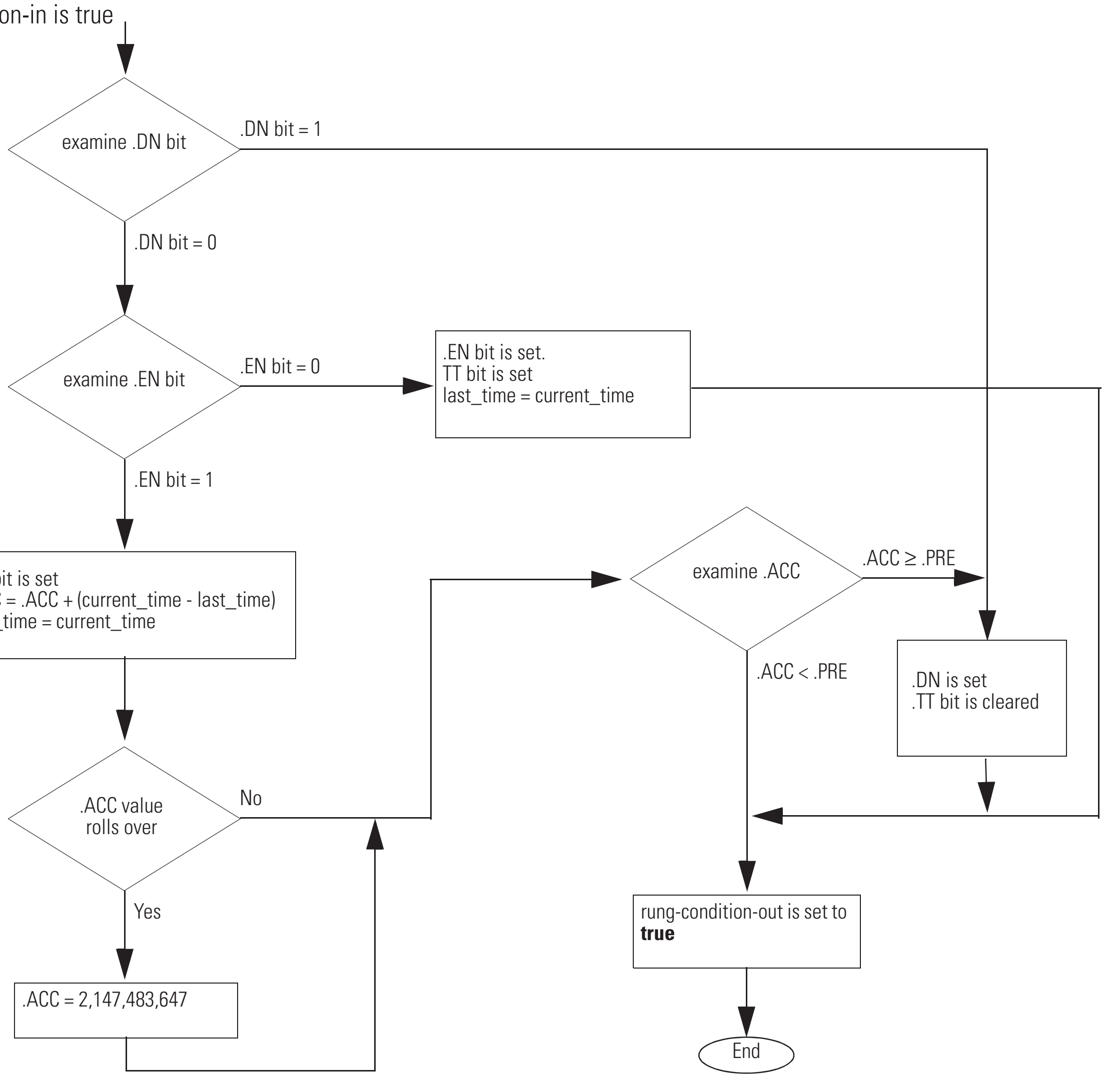
Arithmetic Status Flags:

Not affected

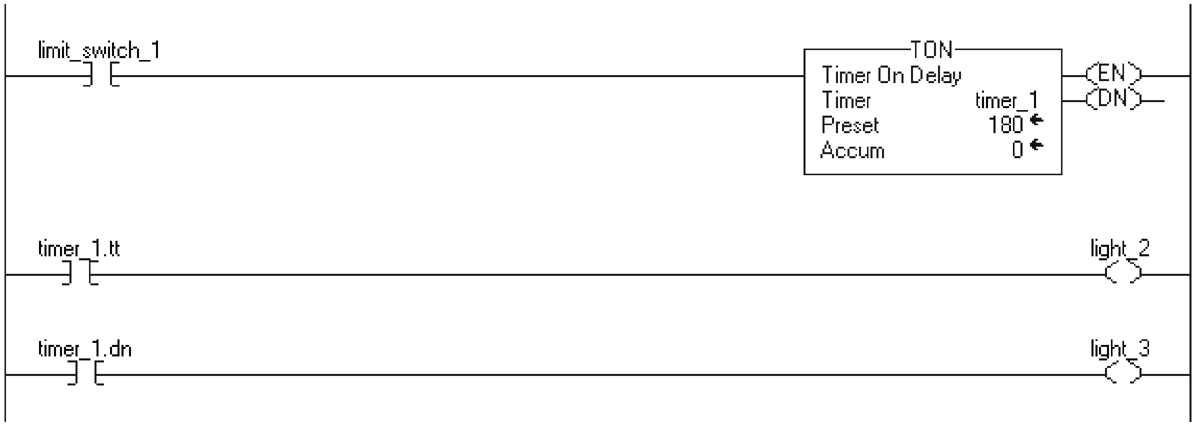
Fault Conditions:

A major fault will occur if	Fault type	Fault code
.PRE < 0	4	34
.ACC < 0	4	34

Execution:

<table>
<tr><th>Condition</th><th>Relay Ladder Action</th></tr>
<tr><td>Prescan</td><td>The .EN, .TT, and .DN bits are cleared.
The .ACC value is cleared.
The rung-condition-out is set to false.</td></tr>
<tr><td>Rung-condition-in is false</td><td>The .EN, .TT, and .DN bits are cleared.
The .ACC value is cleared.
The rung-condition-out is set to false.</td></tr>
<tr><td colspan="2">Rung-condition-in is true
 </td></tr>
<tr><td>Postscan</td><td>The rung-condition-out is set to false.</td></tr>
</table>

Example: When *limit_switch_1* is set, *light_2* is on for 180 ms (timer_1 is timing). When *timer_1.acc* reaches 180, *light_2* goes off and *light_3* goes on. *Light_3* remains on until the TON instruction is disabled. If *limit_switch_1* is cleared while *timer_1* is timing, *light_2* goes off.



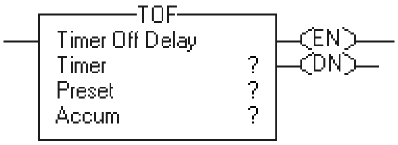
Timer Off Delay (TOF)

The TOF instruction is a non-retentive timer that accumulates time when the instruction is enabled (rung-condition-in is false).

This instruction is available in structured text and function block as TOFR.

See page 122.

Operands:



Relay Ladder

Operand	Type	Format	Description
Timer	TIMER	Tag	Timer structure
Preset	DINT	Immediate	How long to delay (accumulate time)
Accum	DINT	Immediate	Total milliseconds the timer has counted Initial value is typically 0

TIMER Structure

Mnemonic	Data Type	Description
.EN	BOOL	The enable bit indicates that the TOF instruction is enabled.
.TT	BOOL	The timing bit indicates that a timing operation is in process
.DN	BOOL	The done bit is cleared when .ACC ≥ .PRE.
.PRE	DINT	The preset value specifies the value (1 ms units) that the accumulated value must reach before the instruction clears the .DN bit.
.ACC	DINT	The accumulated value specifies the number of milliseconds that have elapsed since the TOF instruction was enabled.

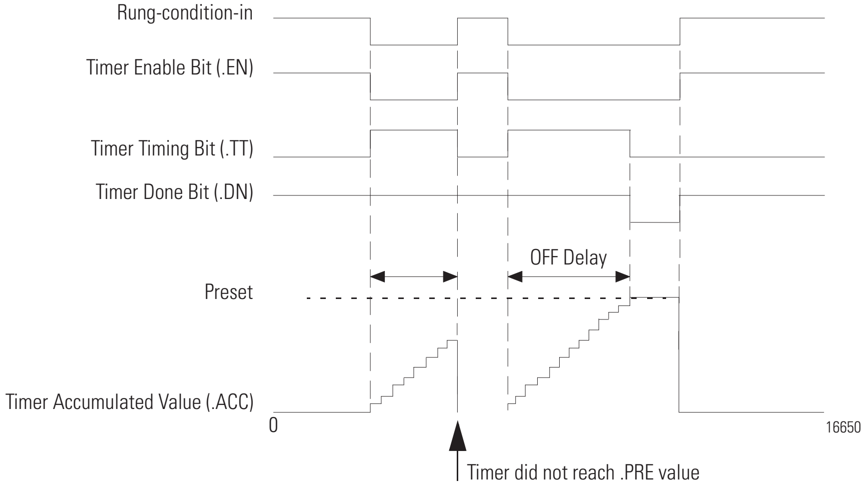
Description:

The TOF instruction accumulates time until:

- the TOF instruction is disabled.
- the .ACC ≥ .PRE.

The time base is always 1 ms. For example, for a two-second timer, enter 2000 for the .PRE value.

When the TOF instruction is disabled, the .ACC value is cleared.



A timer runs by subtracting the time of its last scan from the time now:

```
ACC = ACC + (current_time - last_time_scanned)
```

After it updates the ACC, the timer sets last_time_scanned = current_time. This gets the timer ready for the next scan.

IMPORTANT

Make sure to scan the timer at least every 69 minutes while it runs. Otherwise, the ACC value won't be correct.

The *last_time_scanned* value has a range of up to 69 minutes. The timer's calculation rolls over if you don't scan the timer within 69 minutes. The ACC value won't be correct if this happens.

While a timer runs, scan it within 69 minutes if you put it in a:

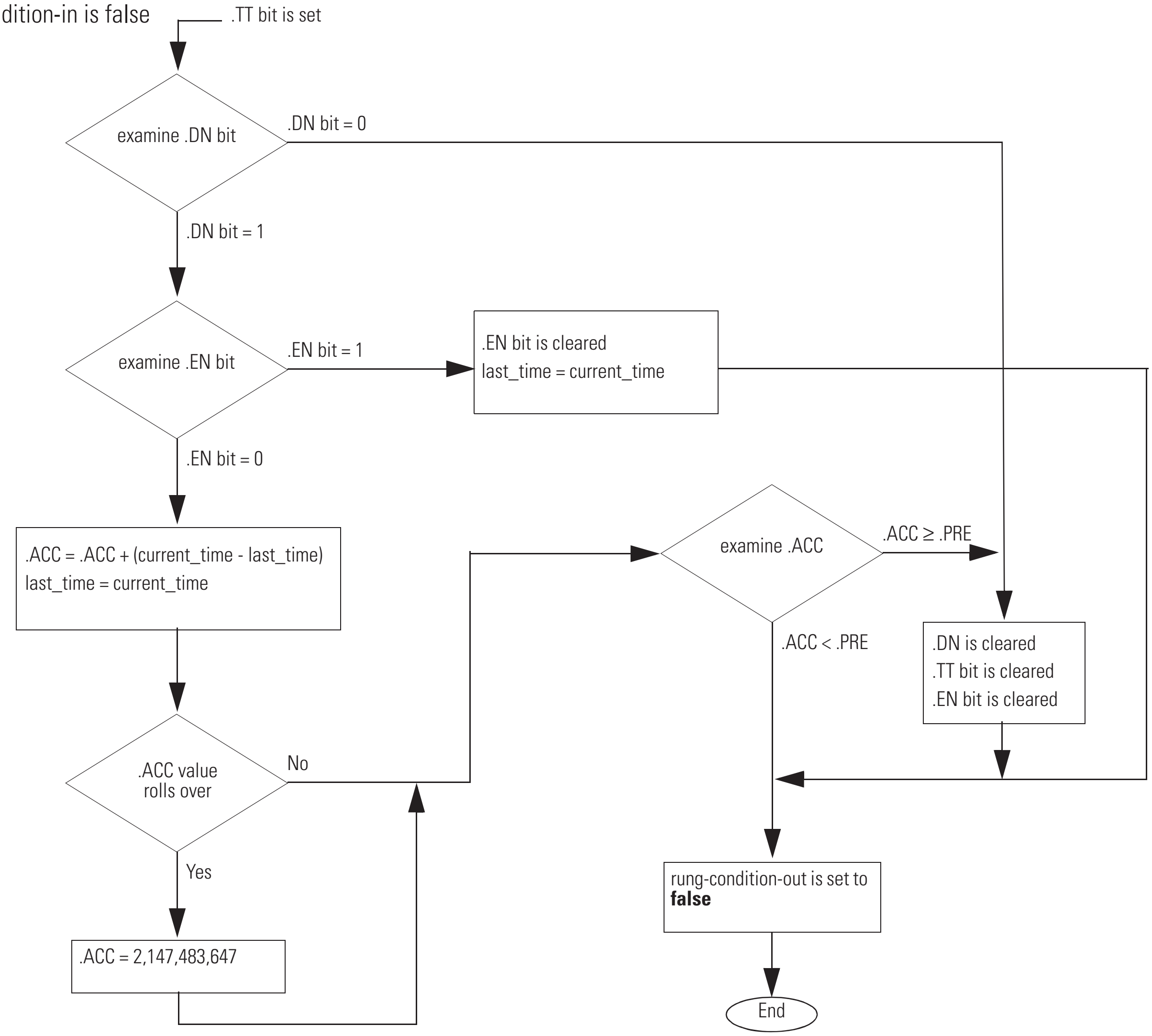
- subroutine.
- section of code that is between JMP and LBL instructions.
- sequential function chart (SFC).
- event or periodic task.
- state routine of a phase.

Arithmetic Status Flags: Not affected

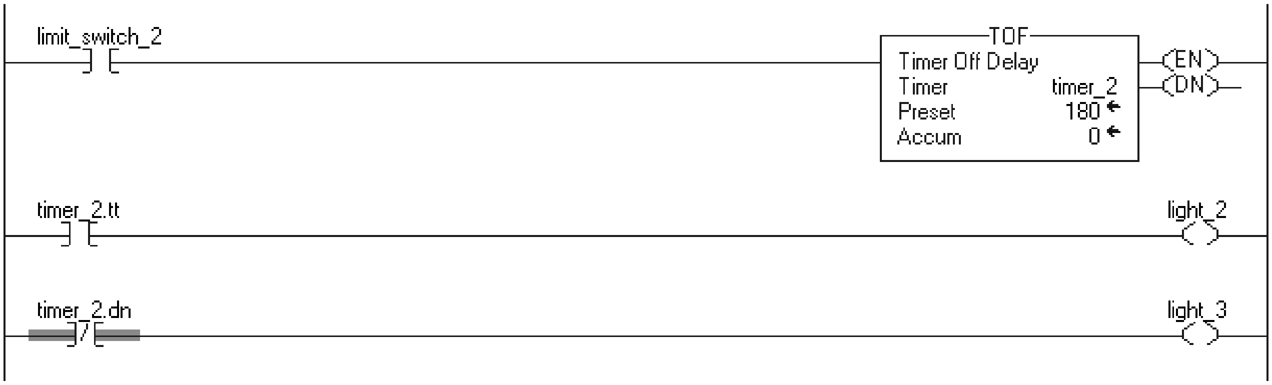
Fault Conditions:

A major fault will occur if	Fault type	Fault code
.PRE < 0	4	34
.ACC < 0	4	34

Execution:

<table>
<tr><th>Condition</th><th>Relay Ladder Action</th></tr>
<tr><td>Prescan</td><td>The .EN, .TT, and .DN bits are cleared.
The .ACC value is set to equal the .PRE value.
The rung-condition-out is set to false.</td></tr>
<tr><td colspan="2">Rung-condition-in is false
 </td></tr>
<tr><td>Rung-condition-in is true</td><td>The .EN, .TT, and .DN bits are set.
The .ACC value is cleared.
The rung-condition-out is set to true.</td></tr>
<tr><td>Postscan</td><td>The rung-condition-out is set to false.</td></tr>
</table>

Example: When *limit_switch_2* is cleared, l*ight_2* is on for 180 ms (timer_2 is timing). When *timer_2.acc* reaches 180, *light_2* goes off and *light_3* goes on. *Light_3* remains on until the TOF instruction is enabled. If *limit_switch_2* is set while *timer_2* is timing, *light_2* goes off.



Retentive Timer On (RTO)

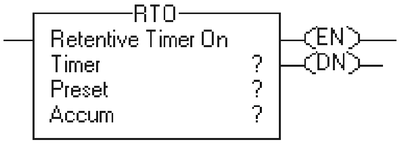
The RTO instruction is a retentive timer that accumulates time when the instruction is enabled.

This instruction is available in structured text and function block as RTOR.

See page 126.

Operands:

Relay Ladder



Operand	Type	Format	Description
Timer	TIMER	Tag	Timer structure
Preset	DINT	Immediate	How long to delay (accumulate time)
Accum	DINT	Immediate	Number of milliseconds the timer has counted Initial value is typically 0

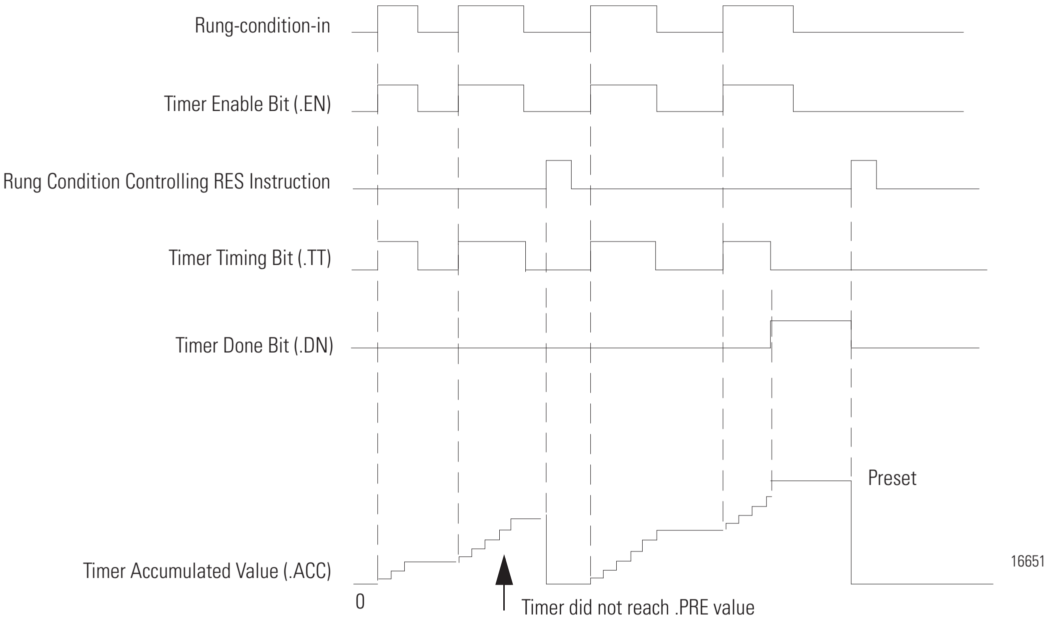
TIMER Structure

Mnemonic	Data Type	Description
.EN	BOOL	The enable bit indicates that the RTO instruction is enabled.
.TT	BOOL	The timing bit indicates that a timing operation is in process
.DN	BOOL	The done bit indicates that .ACC ≥ .PRE.
.PRE	DINT	The preset value specifies the value (1 ms units) that the accumulated value must reach before the instruction sets the .DN bit.
.ACC	DINT	The accumulated value specifies the number of milliseconds that have elapsed since the RTO instruction was enabled.

Description:

The RTO instruction accumulates time until it is disabled. When the RTO instruction is disabled, it retains its .ACC value. You must clear the .ACC value, typically with a RES instruction referencing the same TIMER structure.

The time base is always 1 ms. For example, for a 2-second timer, enter 2000 for the .PRE value.



A timer runs by subtracting the time of its last scan from the time now:

`ACC = ACC + (`*`current_time - last_time_scanned`*`)`

After it updates the ACC, the timer sets last_time_scanned = current_time. This gets the timer ready for the next scan.

IMPORTANT

Make sure to scan the timer at least every 69 minutes while it runs. Otherwise, the ACC value won't be correct.

The *last_time_scanned* value has a range of up to 69 minutes. The timer's calculation rolls over if you don't scan the timer within 69 minutes. The ACC value won't be correct if this happens.

While a timer runs, scan it within 69 minutes if you put it in a:

- subroutine.
- section of code that is between JMP and LBL instructions.
- sequential function chart (SFC).
- event or periodic task.
- state routine of a phase.

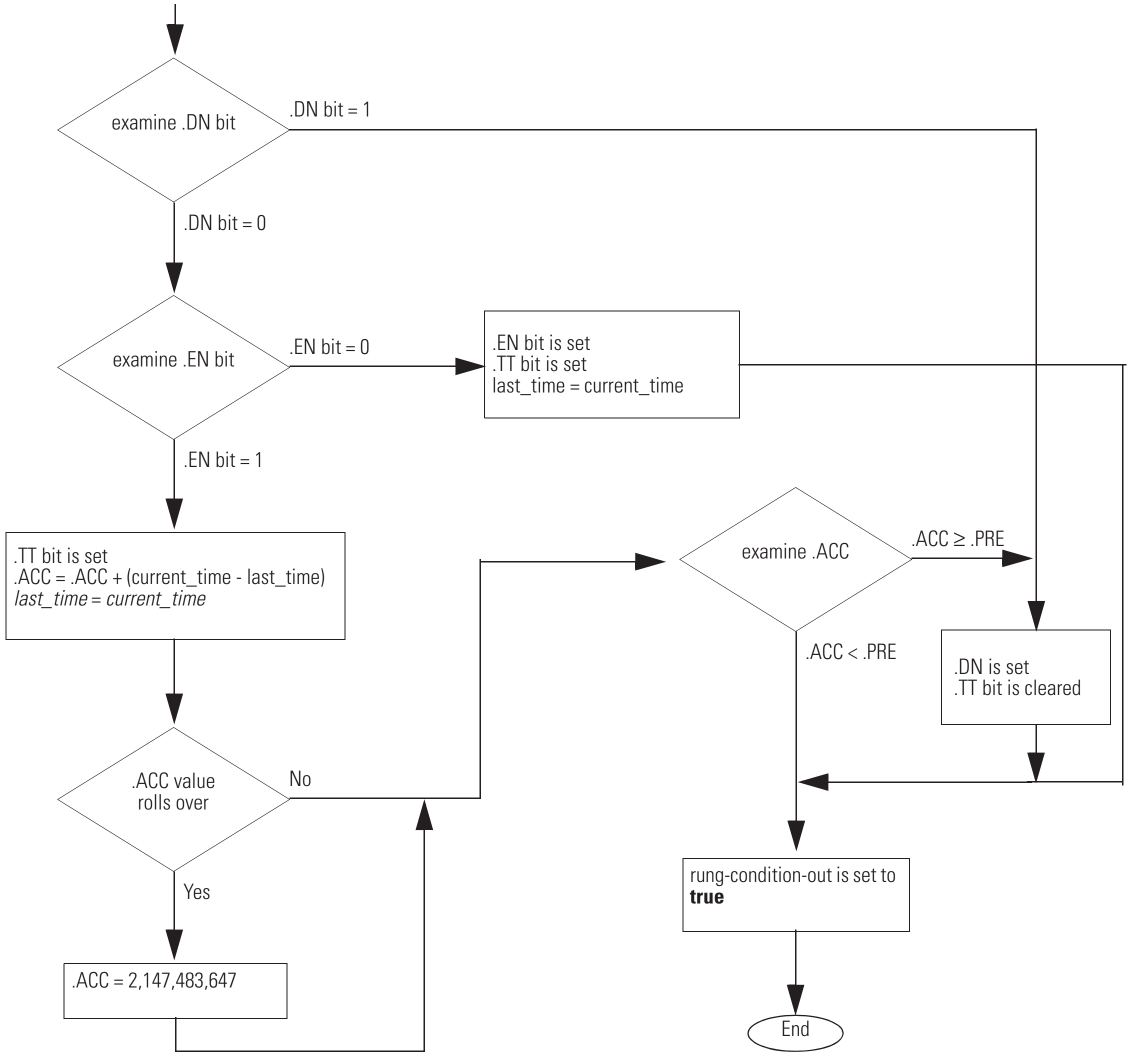
Arithmetic Status Flags:

Not affected

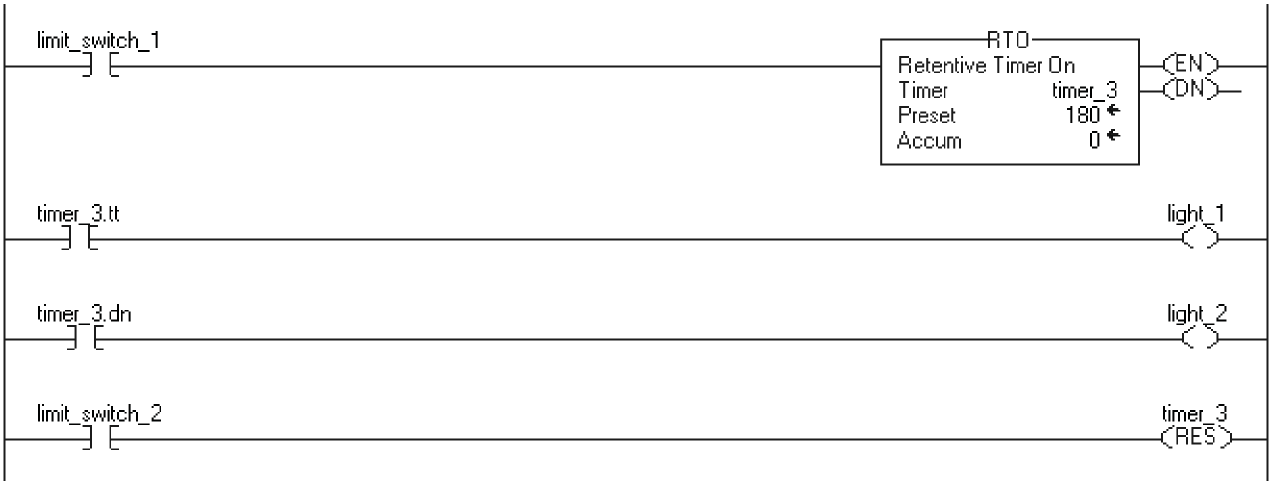
Fault Conditions:

A major fault will occur if	Fault type	Fault code
.PRE < 0	4	34
.ACC < 0	4	34

Execution:

Condition	Relay Ladder Action
Prescan	The .EN, .TT, and .DN bits are cleared. The .ACC value is not modified. The rung-condition-out is set to false.
Rung-condition-in is false	The .EN and .TT bits are cleared. The .DN bit is not modified. The .ACC value is not modified. The rung-condition-out is set to false.
Rung-condition-in is true	
Postscan	The rung-condition-out is set to false.

Example: When *limit_switch_1* is set, *light_1* is on for 180 ms (t*imer_2* is timing). When *timer_3.acc* reaches 180, *light_1* goes off and *light_2* goes on. L*ight_2* remains until *timer_3* is reset. If l*imit_switch_2* is cleared while *timer_3* is timing, *light_1* remains on. When *limit_switch_2* is set, the RES instruction resets *timer_3* (clears status bits and .ACC value).



Timer On Delay with Reset (TONR)

The TONR instruction is a non-retentive timer that accumulates time when TimerEnable is set.

This instruction is available in relay ladder as two separate instructions:

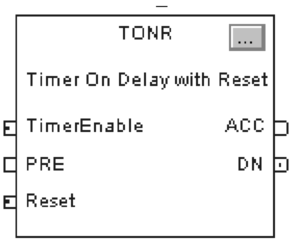
- TON (See page 106).
- RES (See page 143).

Operands:

```
TONR(TONR_tag);
```

Structured Text

Variable	Type	Format	Description
TONR tag	FBD_TIMER	Structure	TONR structure



Function Block

Operand	Type	Format	Description
TONR tag	FBD_TIMER	Structure	TONR structure

FBD_TIMER Structure

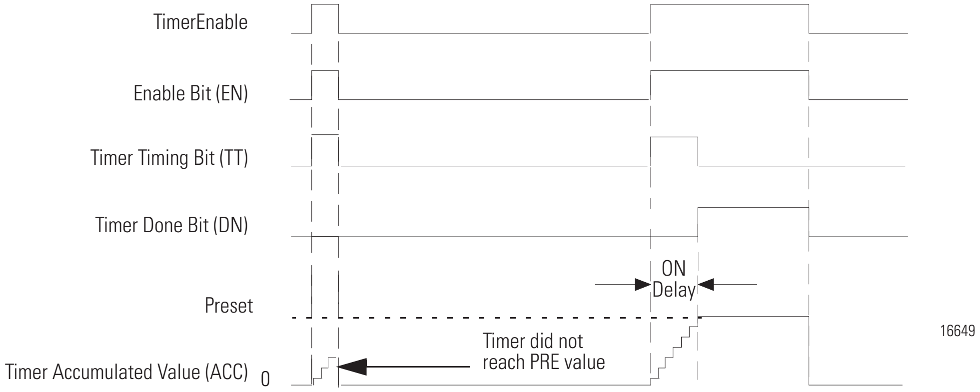
Input Parameter	Data Type	Description
EnableIn	BOOL	**Function Block** If cleared, the instruction does not execute and outputs are not updated. If set, the instruction executes. Default is set. **Structured Text** No effect. The instruction executes.
TimerEnable	BOOL	If set, this enables the timer to run and accumulate time. Default is cleared.
PRE	DINT	Timer preset value. This is the value in 1ms units that ACC must reach before timing is finished. If invalid, the instruction sets the appropriate bit in Status and the timer does not execute. Valid = 0 to maximum positive integer
Reset	BOOL	Request to reset the timer. When set, the timer resets. Default is cleared.
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
ACC	BOOL	Accumulated time in milliseconds.
EN	BOOL	Timer enabled output. Indicates the timer instruction is enabled.

Input Parameter	Data Type	Description
TT	BOOL	Timer timing output. When set, a timing operation is in progress.
DN	BOOL	Timing done output. Indicates when the accumulated time is greater than or equal to the preset value.
Status	DINT	Status of the function block.
InstructFault (Status.0)	BOOL	The instruction detected one of the following execution errors. This is not a minor or major controller error. Check the remaining status bits to determine what occurred.
PresetInv (Status.1)	BOOL	The preset value is invalid.

Description: The TONR instruction accumulates time until the:

- TONR instruction is disabled.
- ACC ≥ PRE.

The time base is always 1 ms. For example, for a two-second timer, enter 2000 for the PRE value.



Set the Reset input parameter to reset the instruction. If TimerEnable is set when Reset is set, the TONR instruction begins timing again when Reset is cleared.

A timer runs by subtracting the time of its last scan from the time now:

```
ACC = ACC + (current_time - last_time_scanned)
```

After it updates the ACC, the timer sets last_time_scanned = current_time. This gets the timer ready for the next scan.

IMPORTANT

Make sure to scan the timer at least every 69 minutes while it runs. Otherwise, the ACC value won't be correct.

The *last_time_scanned* value has a range of up to 69 minutes. The timer's calculation rolls over if you don't scan the timer within 69 minutes. The ACC value won't be correct if this happens.

While a timer runs, scan it within 69 minutes if you put it in a:

- subroutine.
- section of code that is between JMP and LBL instructions.
- sequential function chart (SFC).
- event or periodic task.
- state routine of a phase.

Arithmetic Status Flags: Not affected

Fault Conditions: None

Execution:

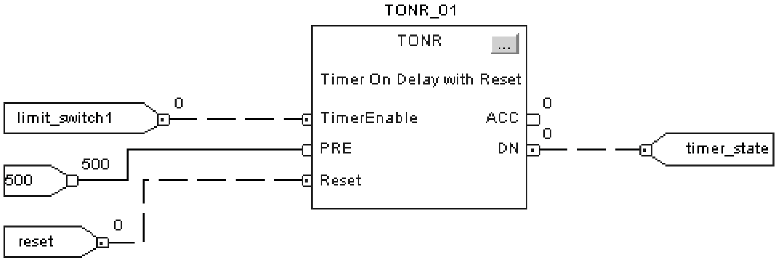
Condition	Function Block Action	Structured Text Action
Prescan	No action taken.	No action taken.
Instruction first scan	EN, TT and DN are cleared. ACC value is set to 0.	EN, TT and DN are cleared. ACC value is set to 0.
Instruction first run	EN, TT and DN are cleared. ACC value is set to 0.	EN, TT and DN are cleared. ACC value is set to 0.
EnableIn is cleared	EnableOut is cleared, the instruction does nothing, and the outputs are not updated.	N/A
EnableIn is set	When EnableIn transitions from cleared to set, the instruction initializes as described for instruction first scan. The instruction executes. EnableOut is set.	EnableIn is always set. The instruction executes.
Reset	When the Reset input parameter is set, the instruction clears EN, TT and DN and sets ACC = zero.	When the Reset input parameter is set, the instruction clears EN, TT and DN and sets ACC = zero.
Postscan	No action taken.	No action taken.

Example: Each scan that limit_switch1 is set, the TONR instruction increments the ACC value by elapsed time until the ACC value reaches the PRE value. When ACC ≥ PRE, the DN parameter is set, and timer_state is set.

Structured Text

```
TONR_01.Preset := 500;

TONR_01.Reset : = reset;

TONR_O1.TimerEnable := limit_switch1;



TONR(TONR_01);



timer_state := TONR_01.DN;
```

Function Block Example



Timer Off Delay with Reset (TOFR)

The TOFR instruction is a non-retentive timer that accumulates time when TimerEnable is cleared.

This instruction is available in relay ladder as two separate instructions:

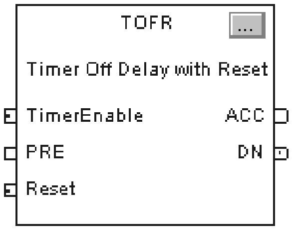
- TON (See page 106).
- RES (See page 143).

Operands:

```
TOFR(TOFR_tag);
```

Structured Text

Variable	Type	Format	Description
TOFR tag	FBD_TIMER	Structure	TOFR structure



Function Block Operands

Operand	Type	Format	Description
TOFR tag	FBD_TIMER	Structure	TOFR structure

FBD_TIMER Structure

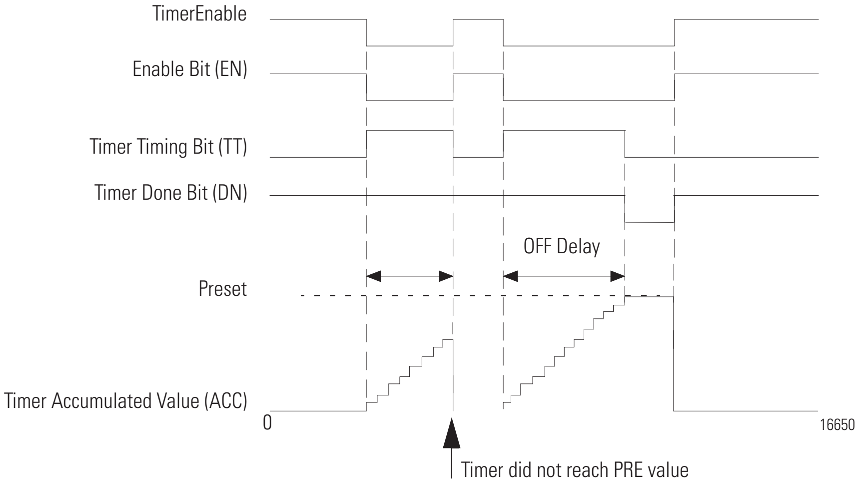
Input Parameter	Data Type	Description
EnableIn	BOOL	**Function Block** If cleared, the instruction does not execute and outputs are not updated. If set, the instruction executes. Default is set. **Structured Text** No effect. The instruction executes.
TimerEnable	BOOL	If cleared, this enables the timer to run and accumulate time. Default is cleared.
PRE	DINT	Timer preset value. This is the value in 1ms units that ACC must reach before timing is finished. If invalid, the instructions sets the appropriate bit in Status and the timer does not execute. Valid = 0 to maximum positive integer
Reset	BOOL	Request to reset the timer. When set, the timer resets. Default is cleared.
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
ACC	BOOL	Accumulated time in milliseconds.

Input Parameter	Data Type	Description
EN	BOOL	Timer enabled output. Indicates the timer instruction is enabled.
TT	BOOL	Timer timing output. When set, a timing operation is in progress.
DN	BOOL	Timing done output. Indicates when accumulated time is greater than or equal to preset.
Status	DINT	Status of the function block.
InstructFault (Status.0)	BOOL	The instruction detected one of the following execution errors. This is not a minor or major controller error. Check the remaining status bits to determine what occurred.
PresetInv (Status.1)	BOOL	The preset value is invalid.

Description: The TOFR instruction accumulates time until the:

- TOFR instruction is disabled.
- ACC ≥ PRE.

The time base is always 1 ms. For example, for a two-second timer, enter 2000 for the PRE value.



Set the Reset input parameter to reset the instruction. If TimerEnable is cleared when Reset is set, the TOFR instruction does not begin timing again when Reset is cleared.

A timer runs by subtracting the time of its last scan from the time now:

```
ACC = ACC + (current_time - last_time_scanned)
```

After it updates the ACC, the timer sets *last_time_scanned = current_time*. This gets the timer ready for the next scan.

IMPORTANT Make sure to scan the timer at least every 69 minutes while it runs. Otherwise, the ACC value won't be correct.

The *last_time_scanned* value has a range of up to 69 minutes. The timer's calculation rolls over if you don't scan the timer within 69 minutes. The ACC value won't be correct if this happens.

While a timer runs, scan it within 69 minutes if you put it in a:

- subroutine
- section of code that is between JMP and LBL instructions
- sequential function chart (SFC)
- event or periodic task
- state routine of a phase

Arithmetic Status Flags: Not affected

Fault Conditions: None

Execution:

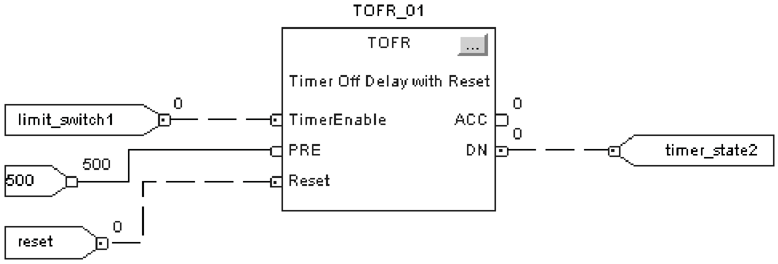
Condition	Function Block Action	Structured Text Action
Prescan	No action taken.	No action taken.
Instruction first scan	EN, TT and DN are cleared. ACC value is set to PRE.	EN, TT and DN are cleared. ACC value is set to PRE.
Instruction first run	EN, TT and DN are cleared. ACC value is set to PRE.	EN, TT and DN are cleared. ACC value is set to PRE.
EnableIn is cleared	EnableOut is cleared, the instruction does nothing, and the outputs are not updated.	na
EnableIn is set	When EnableIn transitions from cleared to set, the instruction initializes as described for instruction first scan. The instruction executes. EnableOut is set.	EnableIn is always set. The instruction executes.
Reset	When the Reset input parameter is set, the instruction clears EN, TT and DN and sets ACC = PRE. Note that this is different than using a RES instruction on a TOF instruction.	When the Reset input parameter is set, the instruction clears EN, TT and DN and sets ACC = PRE. Note that this is different than using a RES instruction on a TOF instruction.
Postscan	No action taken.	No action taken.

Example: Each scan after *limit_switch1* is cleared, the TOFR instruction increments the ACC value by elapsed time until the ACC value reaches the PRE value. When ACC ≥ PRE, the DN parameter is cleared, and *timer_state2* is set.

Structured Text

```
TOFR_01.Preset := 500

TOFR_01.Reset := reset;

TOFR_O1.TimerEnable := limit_switch1;



TOFR(TOFR_01);



timer_state2 := TOFR_01.DN;
```

Function Block



Retentive Timer On with Reset (RTOR)

The RTOR instruction is a retentive timer that accumulates time when TimerEnable is set.

This instruction is available in relay ladder as two separate instructions:

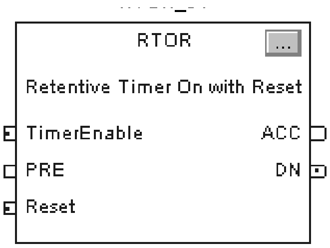
- RTO (See page 114).
- RES (See page 143).

Operands:

```
RTOR(RTOR_tag);
```

Structured Text

Variable	Type	Format	Description
RTOR tag	FBD_TIMER	Structure	RTOR structure



Function Block Operands

Operand	Type	Format	Description
RTOR tag	FBD_TIMER	Structure	RTOR structure

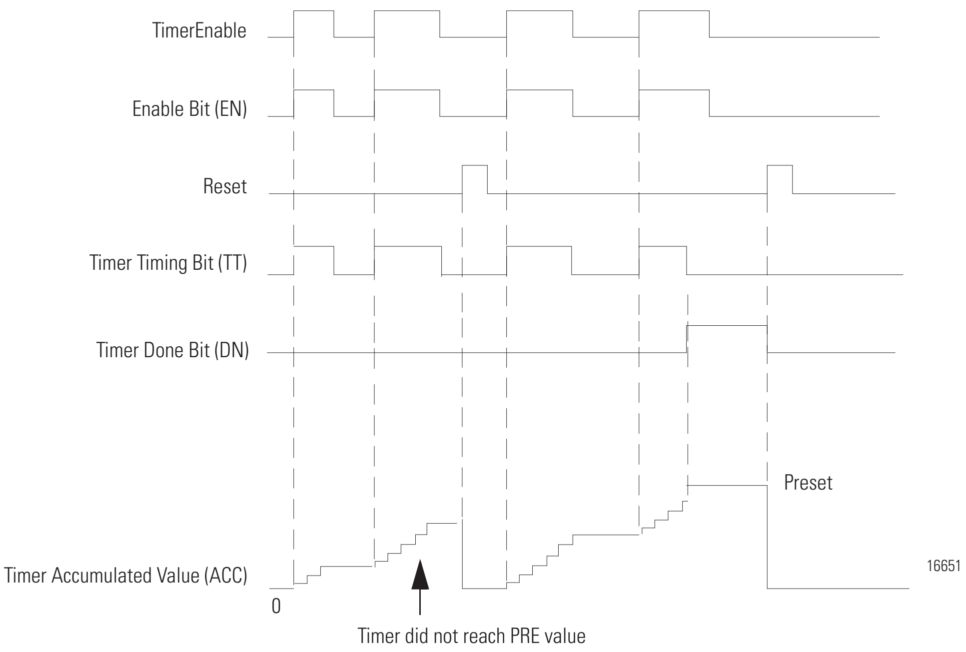
FBD_TIMER Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	**Function Block** If cleared, the instruction does not execute and outputs are not updated. If set, the instruction executes. Default is set. **Structured Text** No effect. The instruction executes.
TimerEnable	BOOL	If set, this enables the timer to run and accumulate time. Default is cleared.
PRE	DINT	Timer preset value. This is the value in 1ms units that ACC must reach before timing is finished. If invalid, the instruction sets the appropriate bit in Status and the timer does not execute. Valid = 0 to maximum positive integer
Reset	BOOL	Request to reset the timer. When set, the timer resets.
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
ACC	DINT	Accumulated time in milliseconds. This value is retained even while the TimerEnable input is cleared. This makes the behavior of this block different than the TONR block.

Input Parameter	Data Type	Description
EN	BOOL	Timer enabled output. Indicates the timer instruction is enabled.
TT	BOOL	Timer timing output. When set, a timing operation is in progress.
DN	BOOL	Timing done output. Indicates when accumulated time is greater than or equal to preset.
Status	DINT	Status of the function block.
InstructFault (Status.0)	BOOL	The instruction detected one of the following execution errors. This is not a minor or major controller error. Check the remaining status bits to determine what occurred.
PresetInv (Status.1)	BOOL	The preset value is invalid.

Description: The RTOR instruction accumulates time until it is disabled. When the RTOR instruction is disabled, it retains its ACC value. You must clear the .ACC value by using the Reset input.

The time base is always 1 ms. For example, for a two-second timer, enter 2000 for the PRE value.



Set the Reset input parameter to reset the instruction. If TimerEnable is set when Reset is set, the RTOR instruction begins timing again when Reset is cleared.

A timer runs by subtracting the time of its last scan from the time now:

```
ACC = ACC + (current_time - last_time_scanned)
```

After it updates the ACC, the timer sets last_time_scanned = current_time. This gets the timer ready for the next scan.

IMPORTANT

Make sure to scan the timer at least every 69 minutes while it runs. Otherwise, the ACC value won't be correct.

The *last_time_scanned* value has a range of up to 69 minutes. The timer's calculation rolls over if you don't scan the timer within 69 minutes. The ACC value won't be correct if this happens.

While a timer runs, scan it within 69 minutes if you put it in a:

- subroutine.
- section of code that is between JMP and LBL instructions.
- sequential function chart (SFC).
- event or periodic task.
- state routine of a phase.

Arithmetic Status Flags: Not affected

Fault Conditions: None

Execution:

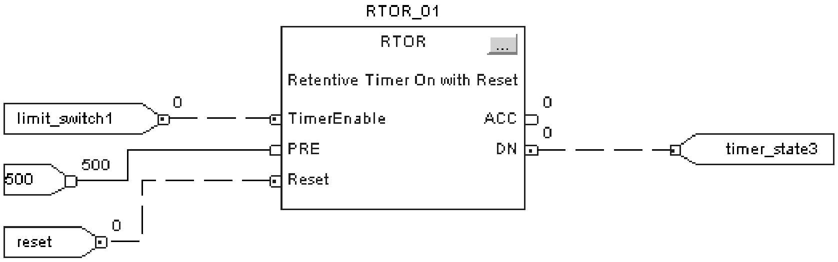
Condition	Function Block Action	Structured Text Action
Prescan	No action taken.	No action taken.
Instruction first scan	EN, TT and DN are cleared ACC value is not modified	EN, TT and DN are cleared ACC value is not modified
Instruction first run	EN, TT and DN are cleared ACC value is not modified	EN, TT and DN are cleared ACC value is not modified
EnableIn is cleared	EnableOut is cleared, the instruction does nothing, and the outputs are not updated.	N/A
EnableIn is set	**Function Block** When EnableIn transitions from cleared to set, the instruction initializes as described for instruction first scan. The instruction executes. EnableOut is set.	EnableIn is always set. The instruction executes.
Reset	When the Reset input parameter is set, the instruction clears EN, TT and DN and sets ACC = zero.	When the Reset input parameter is set, the instruction clears EN, TT and DN and sets ACC = zero.
Postscan	No action taken.	No action taken.

Example: Each scan that *limit_switch1* is set, the RTOR instruction increments the ACC value by elapsed time until the ACC value reaches the PRE value. When ACC ≥ PRE, the DN parameter is set, and *timer_state3* is set.

Structured Text

```
RTOR_01.Preset := 500

RTOR_01.Reset := reset;

RTOR_O1.TimerEnable := limit_switch1;



RTOR(RTOR_01);



timer_state3 := RTOR_01.DN;
```

Function Block



Count Up (CTU)

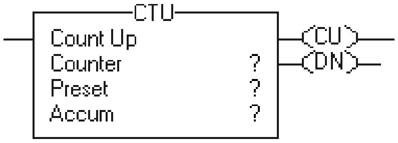
The CTU instruction counts upward.

This instruction is available in structured text and function block as CTUD.

See page 138.

Operands:

Relay Ladder

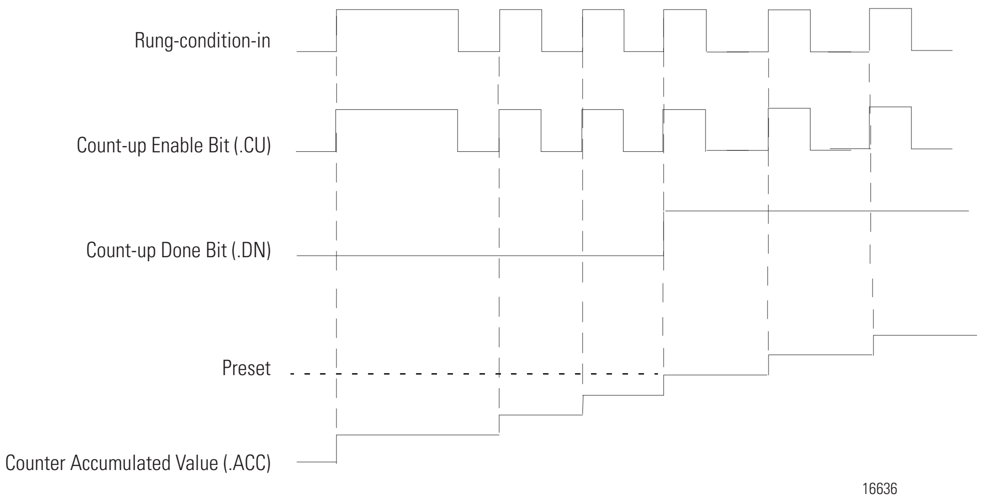


Operand	Type	Format	Description
Counter	COUNTER	Tag	Counter structure
Preset	DINT	Immediate	How high to count
Accum	DINT	Immediate	Number of times the counter has counted Initial value is typically 0

COUNTER Structure

Mnemonic	Data Type	Description
.CU	BOOL	The count up enable bit indicates that the CTU instruction is enabled.
.DN	BOOL	The done bit indicates that .ACC ≥ .PRE.
.OV	BOOL	The overflow bit indicates that the counter exceeded the upper limit of 2,147,483,647. The counter then rolls over to -2,147,483,648 and begins counting up again.
.UN	BOOL	The underflow bit indicates that the counter exceeded the lower limit of -2,147,483,648. The counter then rolls over to 2,147,483,647 and begins counting down again.
.PRE	DINT	The preset value specifies the value that the accumulated value must reach before the instruction sets the .DN bit.
.ACC	DINT	The accumulated value specifies the number of transitions the instruction has counted.

Description: When enabled and the .CU bit is cleared, the CTU instruction increments the counter by one. When enabled and the .CU bit is set, or when disabled, the CTU instruction retains its .ACC value.



The accumulated value continues incrementing, even after the .DN bit is set. To clear the accumulated value, use a RES instruction that references the counter structure or write 0 to the accumulated value.

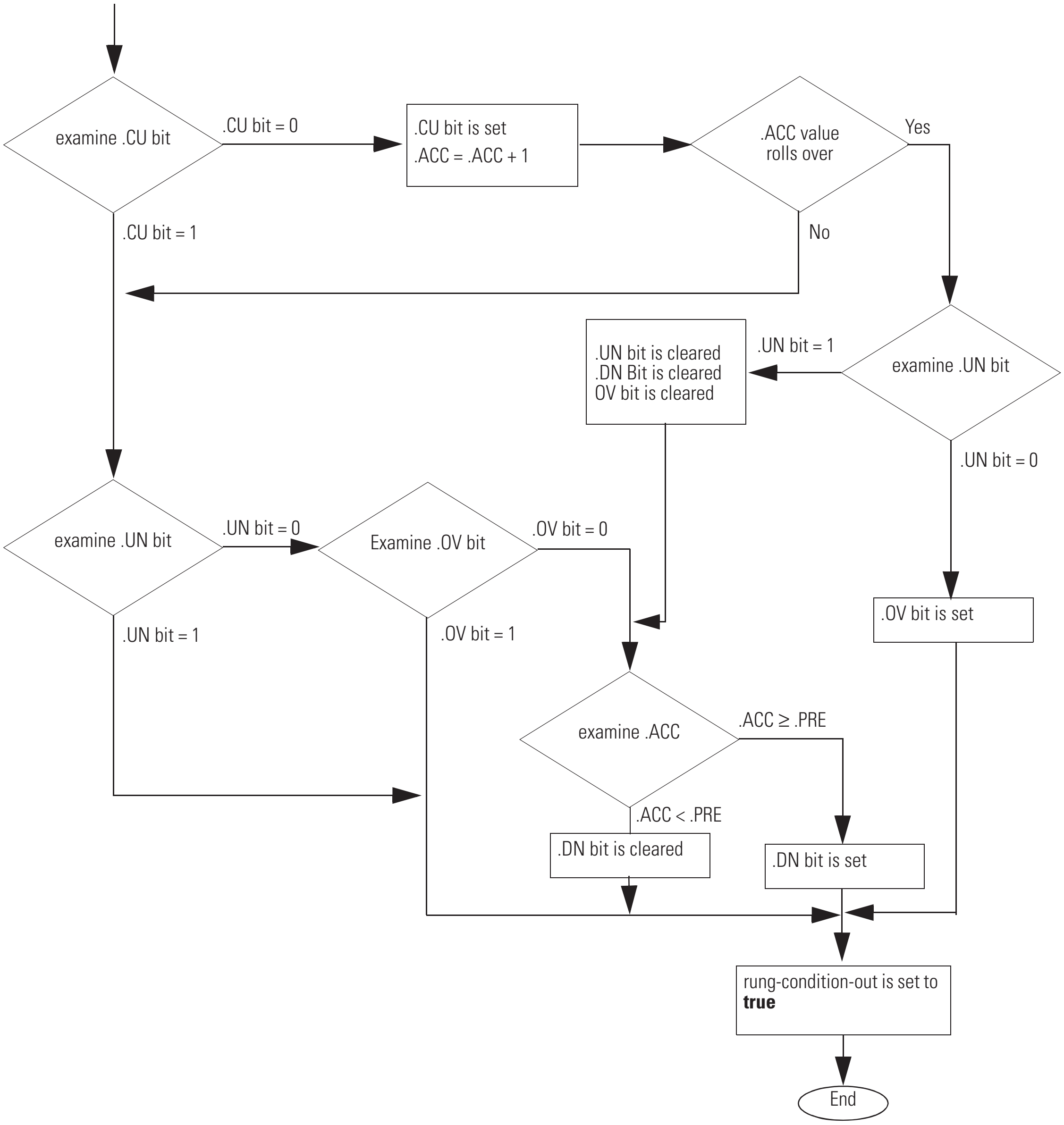
Arithmetic Status Flags: Not affected

Fault Conditions: None

Execution:

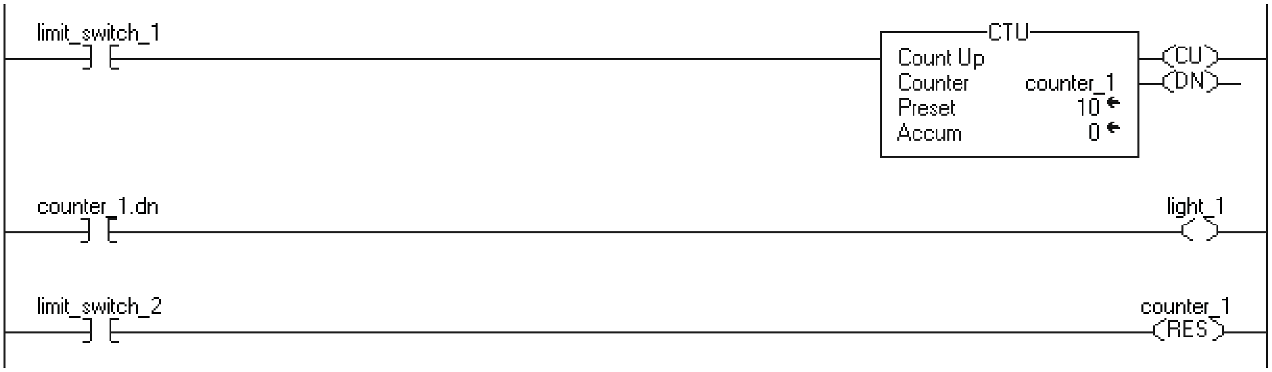
Condition	Relay Ladder Action
Prescan	The .CU bit is set to prevent invalid increments during the first program scan. The rung-condition-out is set to false.
Rung-condition-in is false	The .CU bit is cleared. The rung-condition-out is set to false.

Rung-condition-in is true



Postscan	The rung-condition-out is set to false.

Example: After *limit_switch_1* goes from disabled to enabled 10 times, the .DN bit is set and *light_1* turns on. If *limit_switch_1* continues to go from disabled to enabled, *counter_1* continues to increment its count and the .DN bit remains set. When *limit_switch_2* is enabled, the RES instruction resets *counter_1* (clears the status bits and the .ACC value) and *light_1* turns off.



Count Down (CTD)

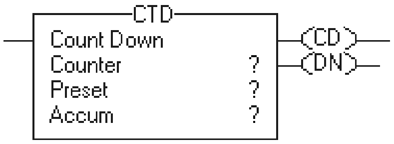
The CTD instruction counts downward.

This instruction is available in structured text and function block as CTUD.

See page 138.

Operands:

Relay Ladder



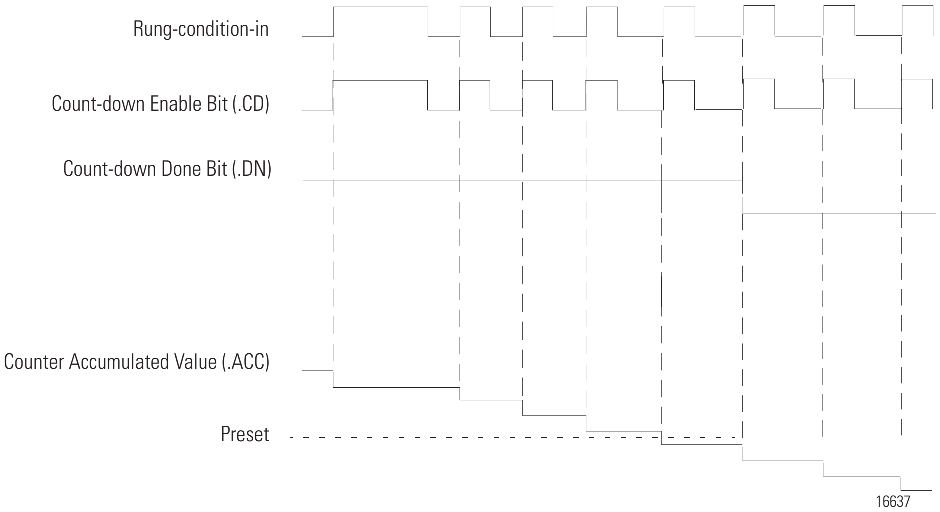
Operand	Type	Format	Description
Counter	COUNTER	Tag	Counter structure
Preset	DINT	Immediate	How low to count
Accum	DINT	Immediate	Number of times the counter has counted Initial value is typically 0

COUNTER Structure

Mnemonic	Data Type	Description
.CD	BOOL	The count down enable bit indicates that the CTD instruction is enabled.
.DN	BOOL	The done bit indicates that .ACC ≥ .PRE.
.OV	BOOL	The overflow bit indicates that the counter exceeded the upper limit of 2,147,483,647. The counter then rolls over to -2,147,483,648 and begins counting up again.
.UN	BOOL	The underflow bit indicates that the counter exceeded the lower limit of -2,147,483,648. The counter then rolls over to 2,147,483,647 and begins counting down again.
.PRE	DINT	The preset value specifies the value that the accumulated value must reach before the instruction sets the .DN bit.
.ACC	DINT	The accumulated value specifies the number of transitions the instruction has counted.

Description: The CTD instruction is typically used with a CTU instruction that references the same counter structure.

When enabled and the .CD bit is cleared, the CTD instruction decrements the counter by one. When enabled and the .CD bit is set, or when disabled, the CTD instruction retains its .ACC value.



The accumulated value continues decrementing, even after the .DN bit is set. To clear the accumulated value, use a RES instruction that references the counter structure or write 0 to the accumulated value.

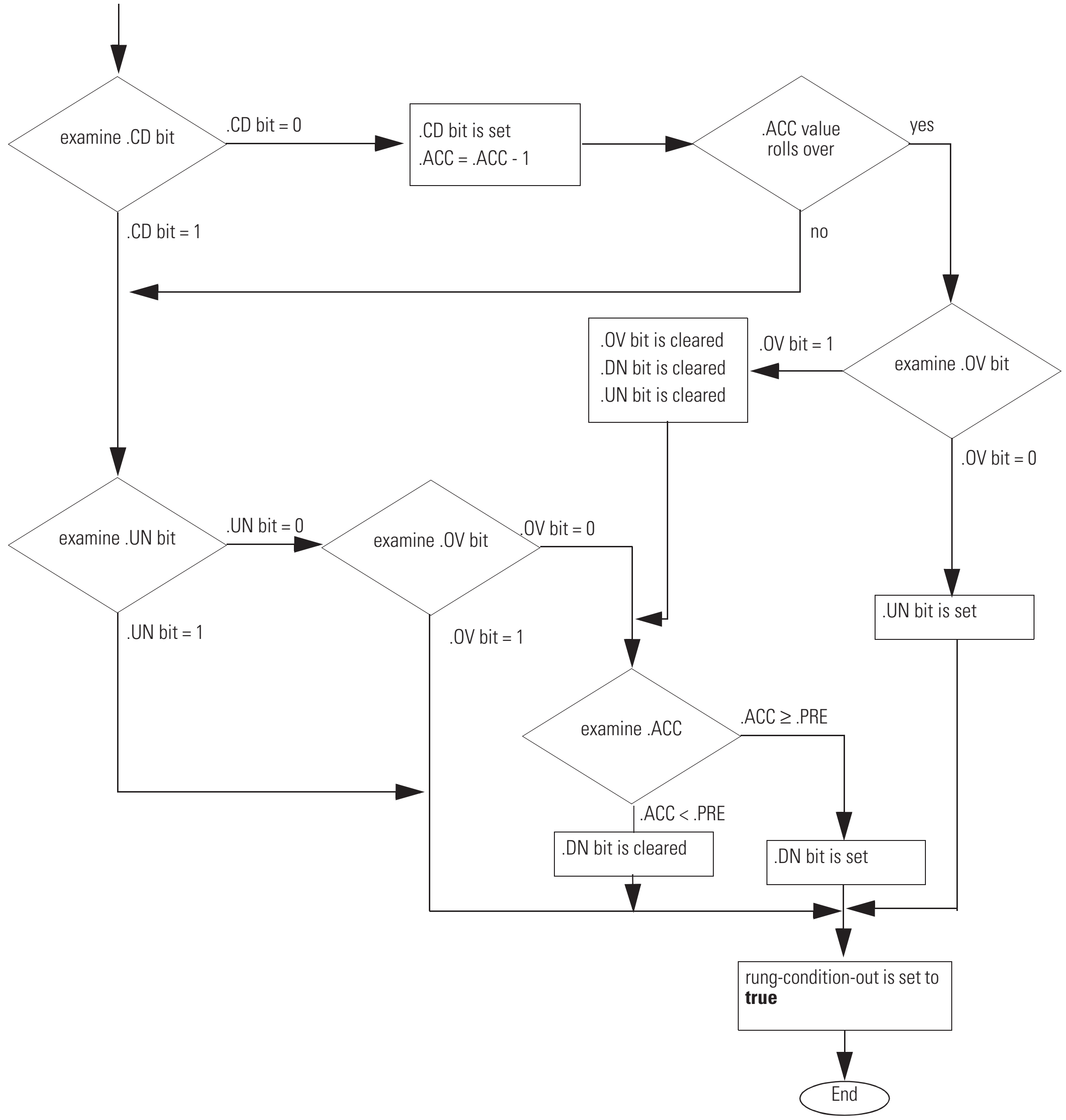
Arithmetic Status Flags: Not affected

Fault Conditions: None

Execution:

Condition	Relay Ladder Action
Prescan	The .CD bit is set to prevent invalid decrements during the first program scan. The rung-condition-out is set to false.
Rung-condition-in is false	The .CD bit is cleared. The rung-condition-out is set to false.

Rung-condition-in is true



Postscan	The rung-condition-out is set to false.

Example: A conveyor brings parts into a buffer zone. Each time a part enters, *limit_switch_1* is enabled and *counter_1* increments by one. Each time a part leaves, *limit_switch_2* is enabled and *counter_1* decrements by one. If there are 100 parts in the buffer zone (*counter_1.dn* is set), *conveyor_A* turns on and stops the conveyor from bringing in any more parts until the buffer has room for more parts.

Count Up/Down (CTUD)

The CTUD instruction counts up by one when CUEnable transitions from clear to set. The instruction counts down by one when CDEnable transitions from clear to set.

This instruction is available in relay ladder as three separate instructions:

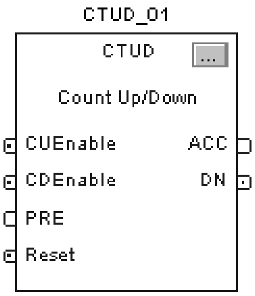
- CTU (See page 130).
- CTD (See page 134).
- RES (See page 143).

Operands:

```
CTUD(CTUD_tag);
```

Structured Text

Variable	Type	Format	Description
CTUD tag	FBD_COUNTER	Structure	CTUD structure



Function Block

Operand	Type	Format	Description
CTUD tag	FBD_COUNTER	Structure	CTUD structure

FBD_COUNTER Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	**Function Block** If cleared, the instruction does not execute and outputs are not updated. If set, the instruction executes. Default is set. **Structured Text** No effect. The instruction executes.
CUEnable	BOOL	Enable up count. When input toggles from clear to set, accumulator counts up by one. Default is cleared.

Input Parameter	Data Type	Description
CDEnable	BOOL	Enable down count. When input toggles from clear to set, accumulator counts down by one. Default is cleared.
PRE	DINT	Counter preset value. This is the value the accumulated value must reach before DN is set. Valid = any integer Default is 0.
Reset	BOOL	Request to reset the timer. When set, the counter resets. Default is cleared.

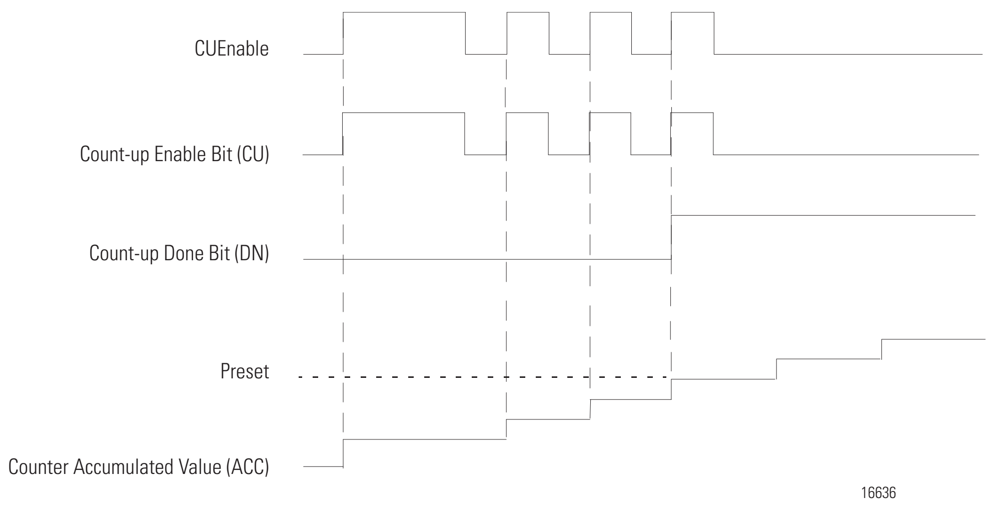
Output Parameter	Data Type	Description
EnableOut	BOOL	The instruction produced a valid result.
ACC	DINT	Accumulated value.
CU	BOOL	Count up enabled.
CD	BOOL	Count down enabled.
DN	BOOL	Counting done. Set when accumulated value is greater than or equal to preset.
OV	BOOL	Counter overflow. Indicates the counter exceeded the upper limit of 2,147,483,647. The counter then rolls over to -2,147,483,648 and begins counting down again.
UN	BOOL	Counter underflow. Indicates the counter exceeded the lower limit of -2,147,483,648. The counter then rolls over to 2,147,483,647 and begins counting down again.

Description

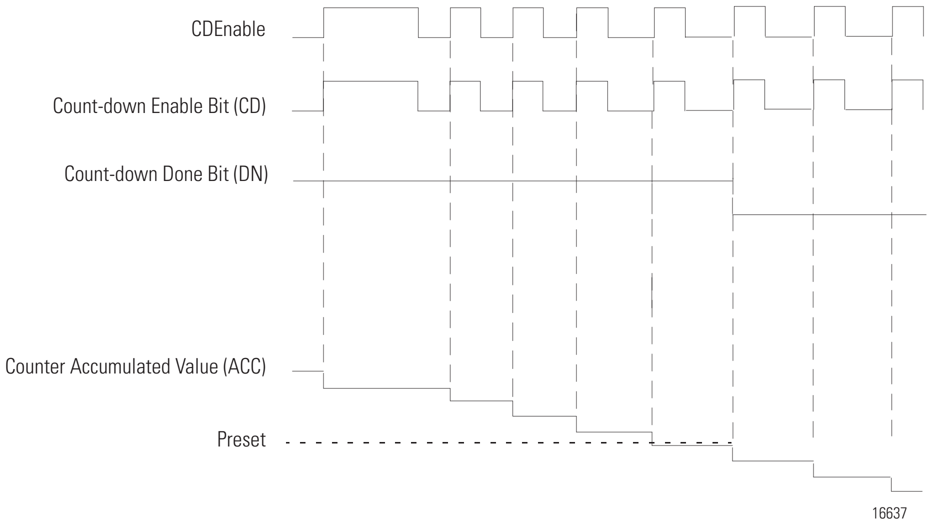
When enabled and CUEnable is set, the CTUD instructions increments the counter by one. When enabled and CDEnable is set, the CTUD instruction decrements the counter by one.

Both the CUEnable and CDEnable input parameters can both be toggled during the same scan. The instruction executes the count up prior to the count down.

Counting Up



Counting Down



When disabled, the CTUD instruction retains its accumulated value. Set the Reset input parameter of the FBD_COUNTER structure to reset the instruction.

Arithmetic Status Flags: Not affected

Fault Conditions: None

Execution:

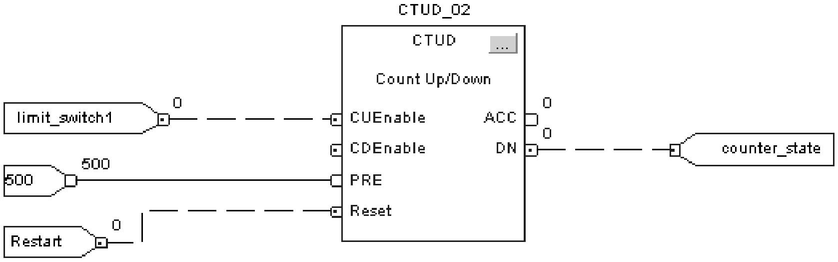
Condition	Function Block Action	Structured Text Action
Prescan	No initialization required.	No initialization required.
Instruction first scan	CUEnable$_{n-1}$ and CDEnable$_{n-1}$ are set.	CUEnable$_{n-1}$ and CDEnable$_{n-1}$ are set.
Instruction first run	CUEnable$_{n-1}$ and CDEnable$_{n-1}$ are set.	CUEnable$_{n-1}$ and CDEnable$_{n-1}$ are set.
EnableIn is cleared	EnableOut is cleared, the instruction does nothing, and the outputs are not updated.	N/A
EnableIn is set	The instruction sets CUEnable$_{n-1}$ and CDEnable$_{n-1}$. On a cleared to set transition of EnableIn: • the instruction executes. • EnableOut is set.	The instruction sets CUEnable$_{n-1}$ and CDEnable$_{n-1}$. EnableIn is always set. The instruction executes.
Reset	When set, the instruction clears CUEnable$_{n-1}$, CDEnable$_{n-1}$, CU, CD, DN, OV, and UN and sets ACC = zero.	When set, the instruction clears CUEnable$_{n-1}$, CDEnable$_{n-1}$, CU, CD, DN, OV, and UN and sets ACC = zero.
Postscan	No action taken.	No action taken.

Example: When *limit_switch1* goes from cleared to set, CUEnable is set for one scan and the CTUD instruction increments the ACC value by 1. When ACC ≥ PRE, the DN parameter is set, which enables the function block instruction following the CTUD instruction.

Structured Text

```
CTUD_01.Preset := 500;

CTUD_01.Reset := Restart;

CTUD_O1.CUEnable := limit_switch1;



CTUD(CTUD_01);



counter_state := CTUD_01.DN;
```

Function Block



Reset (RES)

The RES instruction resets a TIMER, COUNTER, or CONTROL structure.

Operands:

Relay Ladder

Operand	Type	Format	Description
Structure	TIMER CONTROL COUNTER	Tag	Structure to reset

Description:

When enabled the RES instruction clears these elements.

When using a RES instruction for a	The instruction clears
TIMER	.ACC value Control status bits
COUNTER	.ACC value Control status bits
CONTROL	.POS value Control status bits

ATTENTION

Because the RES instruction clears the .ACC value, .DN bit, and .TT bit, do not use the RES instruction to reset a TOF timer.

Arithmetic Status Flags:

Not affected

Fault Conditions:

None

Execution:

Condition	**Relay Ladder Action**
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The RES instruction resets the specified structure. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Input/Output Instructions

(MSG, GSV, SSV, IOT)

Introduction

The input/output instructions read or write data to, or from, the controller. The input/output instructions also read or write data to a block of data to, or from, another module on another network.

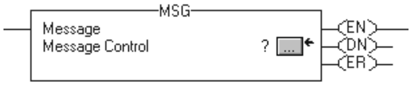
If you want to	Use this instruction	Available in these languages	Page
Send data to or from another module	MSG	Relay ladder Structured text	146
Get controller status information	GSV	Relay ladder Structured text	182
Set controller status information	SSV	Relay ladder Structured text	182
Send output values to an I/O module or consuming controller at a specific point in your logic. Trigger an event task in another controller.	IOT	Relay ladder Structured text	209

Message (MSG)

The MSG instruction asynchronously reads or writes a block of data to another module on a network.

Operands:

Relay Ladder



Operand	Type	Format	Description
Message control	MESSAGE	Tag	Message structure

Structured Text

```
MSG(MessageControl);
```

The operands are the same as those for the relay ladder MSG instruction.

MESSAGE Structure

ATTENTION

If you check the status bits more than once.

The controller changes the DN, ER, EW, and ST bits asynchronous to the scan of your logic. Use a copy of the bits if you check them in more than one place in your logic. Otherwise, the bits may change during the scan and your logic won't work as you expect it.

One way to make a copy is to use the FLAGS word. Copy the FLAGS word to another tag and check the bits in the copy.

IMPORTANT

Do not change the following status bits of a MSG instruction:

- DN
- EN
- ER
- EW
- ST

Do not change those bits either by themselves or as part of the FLAGS word. If you do, the controller may have a non-recoverable fault. The controller clears the project from its memory when it has a non-recoverable fault.

<table>
<tr><th>Mnemonic</th><th>Data Type</th><th colspan="2">Description</th></tr>
<tr><td rowspan="10">.FLAGS</td><td rowspan="10">INT</td><td colspan="2">The FLAGS member provides access to the status members (bits) in one 16-bit word.</td></tr>
<tr><td>This bit</td><td>Is this member</td></tr>
<tr><td>2</td><td>.EW</td></tr>
<tr><td>4</td><td>.ER</td></tr>
<tr><td>5</td><td>.DN</td></tr>
<tr><td>6</td><td>.ST</td></tr>
<tr><td>7</td><td>.EN</td></tr>
<tr><td>8</td><td>.TO</td></tr>
<tr><td>9</td><td>.EN_CC</td></tr>
<tr><td colspan="2">Important: Do not change the EW, ER, DN, or ST bits of the FLAGS member. For example, do not clear the entire FLAGS word. The controller ignores the change and uses the internally-stored values of the bits.</td></tr>
<tr><td>.ERR</td><td>INT</td><td colspan="2">If the .ER bit is set, the error code word identifies error codes for the MSG instruction.</td></tr>
<tr><td>.EXERR</td><td>INT</td><td colspan="2">The extended error code word specifies additional error code information for some error codes.</td></tr>
<tr><td>.REQ_LEN</td><td>INT</td><td colspan="2">The requested length specifies how many words the message instruction will attempt to transfer.</td></tr>
<tr><td>.DN_LEN</td><td>INT</td><td colspan="2">The done length identifies how many words actually transferred.</td></tr>
<tr><td>.EW</td><td>BOOL</td><td colspan="2">The enable waiting bit is set when the controller detects that a message request has entered the queue. The controller resets the .EW bit when the .ST bit is set.
Important: Do not change the EW bit. The controller ignores the change and uses the internally-stored value of the bit.</td></tr>
<tr><td>.ER</td><td>BOOL</td><td colspan="2">The error bit is set when the controller detects that a transfer failed. The .ER bit is reset the next time the rung-condition-in goes from false to true.
Important: Do not change the ER bit.</td></tr>
<tr><td>.DN</td><td>BOOL</td><td colspan="2">The done bit is set when the last packet of the message is successfully transferred. The .DN bit is reset the next time the rung-condition-in goes from false to true.
Important: Do not change the DN bit.</td></tr>
<tr><td>.ST</td><td>BOOL</td><td colspan="2">The start bit is set when the controller begins executing the MSG instruction. The .ST bit is reset when the .DN bit or the .ER bit is set.
Important: Do not change the ST bit. The controller ignores the change and uses the internally-stored value of the bit.</td></tr>
<tr><td>.EN</td><td>BOOL</td><td colspan="2">The enable bit is set when the rung-condition-in goes true and remains set until either the .DN bit or the .ER bit is set and the rung-condition-in is false. If the rung-condition-in goes false, but the .DN bit and the .ER bit are cleared, the .EN bit remains set.
Important: Do not change the EN bit.</td></tr>
<tr><td>.TO</td><td>BOOL</td><td colspan="2">If you manually set the .TO bit, the controller stops processing the message and sets the .ER bit.</td></tr>
<tr><td>.EN_CC</td><td>BOOL</td><td colspan="2">The enable cache bit determines how to manage the MSG connection. Refer to Choose a Cache Option on page 179 Connections for MSG instructions going out the serial port are not cached, even if the .EN_CC bit is set.</td></tr>
<tr><td>.ERR_SRC</td><td>SINT</td><td colspan="2">Used by RSLogix 5000 software to show the error path on the Message Configuration dialog box</td></tr>
<tr><td>.DestinationLink</td><td>INT</td><td colspan="2">To change the Destination Link of a DH+ or CIP with Source ID message, set this member to the required value.</td></tr>
</table>

<table>
<tr><th>Mnemonic</th><th>Data Type</th><th colspan="2">Description</th></tr>
<tr><td>.DestinationNode</td><td>INT</td><td colspan="2">To change the Destination Node of a DH+ or CIP with Source ID message, set this member to the required value.</td></tr>
<tr><td>.SourceLink</td><td>INT</td><td colspan="2">To change the Source Link of a DH+ or CIP with Source ID message, set this member to the required value.</td></tr>
<tr><td>.Class</td><td>INT</td><td colspan="2">To change the Class parameter of a CIP Generic message, set this member to the required value.</td></tr>
<tr><td>.Attribute</td><td>INT</td><td colspan="2">To change the Attribute parameter of a CIP Generic message, set this member to the required value.</td></tr>
<tr><td>.Instance</td><td>DINT</td><td colspan="2">To change the Instance parameter of a CIP Generic message, set this member to the required value.</td></tr>
<tr><td rowspan="4">.LocalIndex</td><td rowspan="4">DINT</td><td colspan="2">If you use an asterisk [*] to designate the element number of the local array, the LocalIndex provides the element number. To change the element number, set this member to the required value.</td></tr>
<tr><td>If the message</td><td>Then the local array is</td></tr>
<tr><td>Reads data</td><td>Destination element</td></tr>
<tr><td>Writes data</td><td>Source element</td></tr>
<tr><td>.Channel</td><td>SINT</td><td colspan="2">To send the message out a different channel of the 1756-DHRIO module, set this member to the required value. Use either the ASCII character A or B.</td></tr>
<tr><td>.Rack</td><td>SINT</td><td colspan="2">To change the rack number for a block transfer message, set this member to the required rack number (octal).</td></tr>
<tr><td>.Group</td><td>SINT</td><td colspan="2">To change the group number for a block transfer message, set this member to the required group number (octal).</td></tr>
<tr><td rowspan="4">.Slot</td><td rowspan="4">SINT</td><td colspan="2">To change the slot number for a block transfer message, set this member to the required slot number.</td></tr>
<tr><td>If the message goes over this network</td><td>Then specify the slot number</td></tr>
<tr><td>Universal remote I/O</td><td>Octal</td></tr>
<tr><td>ControlNet</td><td>Decimal (0…15)</td></tr>
<tr><td>.Path</td><td>STRING</td><td colspan="2">To send the message to a different controller, set this member to the new path.
• Enter the path as hexadecimal values.
• Omit commas [,]
For example, for a path of 1, 0, 2, 42, 1, 3, enter $01$00$02$2A$01$03.
To browse to a device and automatically create a portion or all of the new string, right-click a string tag and choose ‘Go to Message Path Editor’.</td></tr>
<tr><td rowspan="4">.RemoteIndex</td><td rowspan="4">DINT</td><td colspan="2">If you use an asterisk [*] to designate the element number of the remote array, the RemoteIndex provides the element number. To change the element number, set this member to the required value.</td></tr>
<tr><td>If the message</td><td>Then the remote array is</td></tr>
<tr><td>Reads data</td><td>Source element</td></tr>
<tr><td>Writes data</td><td>Destination element</td></tr>
<tr><td rowspan="4">.RemoteElement</td><td rowspan="4">STRING</td><td colspan="2">To specify a different tag or address in the controller to which the message is sent, set this member to the required value. Enter the tag or address as ASCII characters.</td></tr>
<tr><td>If the message</td><td>Then the remote array is</td></tr>
<tr><td>Reads data</td><td>Source element</td></tr>
<tr><td>Writes data</td><td>Destination element</td></tr>
</table>

<table>
<tr><th>Mnemonic</th><th>Data Type</th><th colspan="2">Description</th></tr>
<tr><td rowspan="4">.UnconnnectedTimeout</td><td rowspan="4">DINT</td><td colspan="2">Time out for an unconnected message or for making a connection. The default value is 30 seconds.</td></tr>
<tr><td>If the message is</td><td>Then</td></tr>
<tr><td>Unconnected</td><td>The ER bit turns on if the controller doesn’t get a response within the UnconnectedTimeout time.</td></tr>
<tr><td>Connected</td><td>The ER bit turns on if the controller doesn’t get a response for making the connection within the UnconnectedTimeout time.</td></tr>
<tr><td>.ConnectionRate</td><td>DINT</td><td colspan="2" rowspan="2">Time out for a connected message once it has a connection. This time out is for the response from the other device about the sending of the data.
• This time out applies only after the connection is made.
• The time out = ConnectionRate x TimeoutMultiplier.
• The default ConnectionRate is 7.5 seconds.
• The default TimeoutMultiplier is 0 (which is a multiplication factor of 4).
• The default time out for connected messages is 30 seconds (7.5 seconds x 4 = 30 seconds).
• To change the time out, change the ConnectionRate and leave the TimeoutMultiplier at the default value.</td></tr>
<tr><td>.TimeoutMultiplier</td><td>SINT</td></tr>
</table>

Description

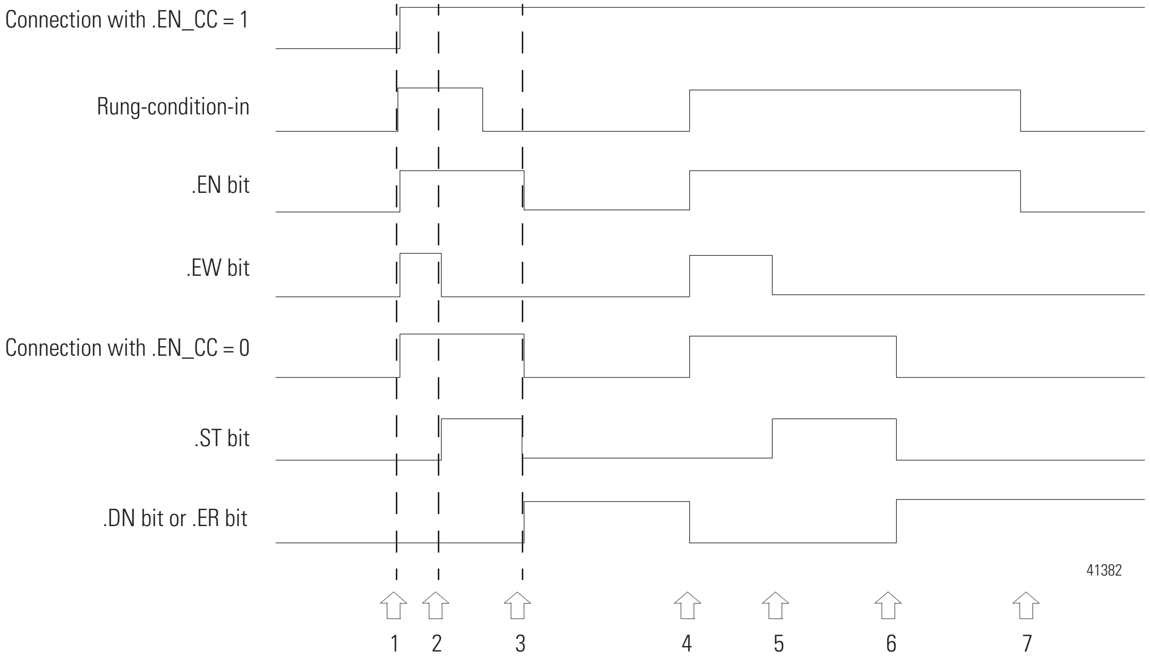
The MSG instruction transfers elements of data.

This is a transitional instruction.

- In relay ladder, toggle the rung-condition-in from cleared to set each time the instruction should execute.
- In structured text, condition the instruction so that it executes only on a transition.

See Appendix B.

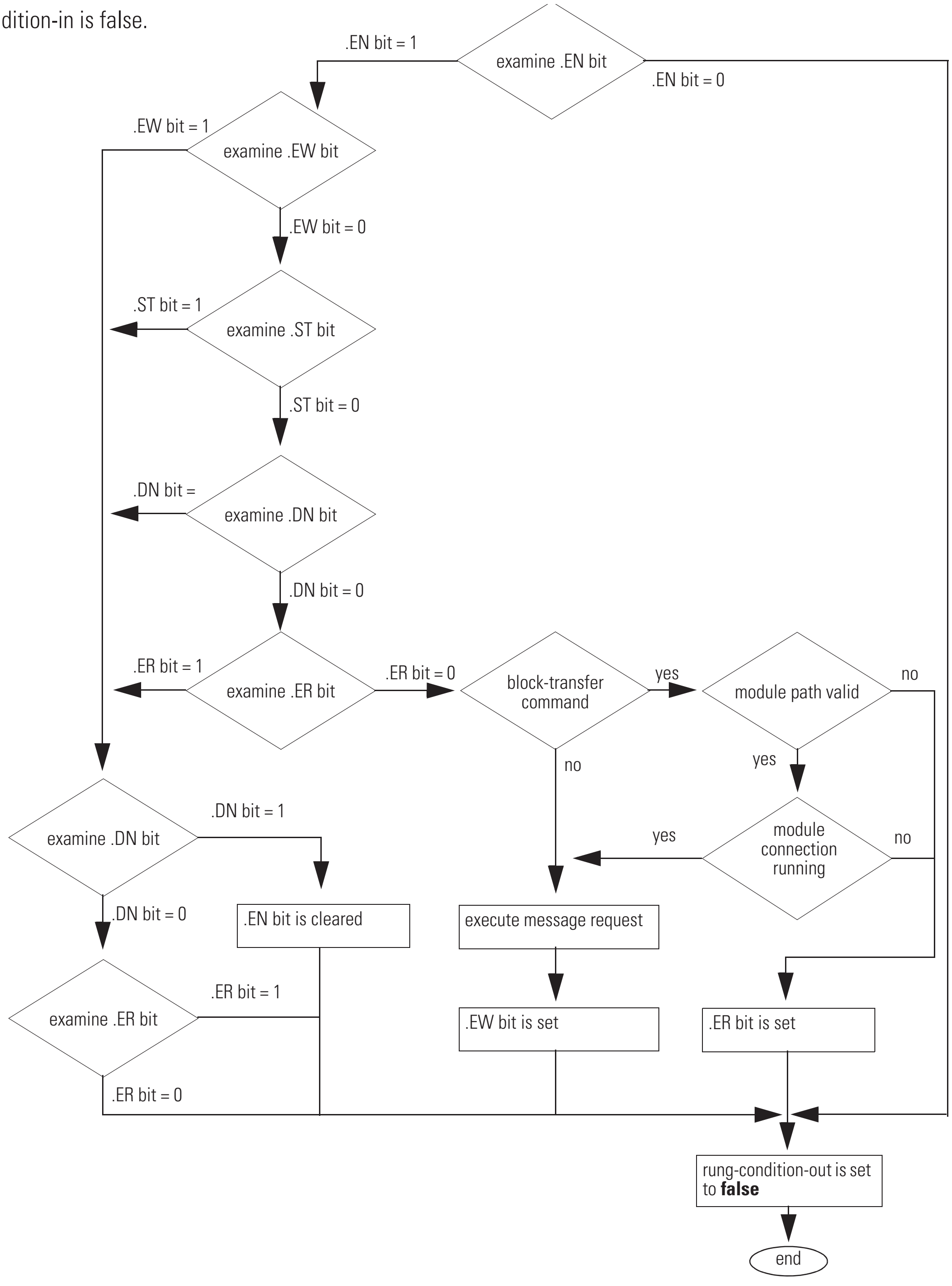
The size of each element depends on the data types you specify and the type of message command you use.



Where	Description	Where	Description
1	Rung-condition-in is true. .EN is set. .EW is set. Connection is opened*.	5	Message is sent. .ST is set. .EW is cleared.
2	Message is sent. .ST is set. .EW is cleared.	6	Message is done or errored. Rung-condition-in is still true. .DN or .ER is set. .ST is cleared. Connection is closed (if .EN_CC = 0).
3	Message is done or errored. Rung-condition-in is false. .DN or .ER is set. .ST is cleared. Connection is closed (if .EN_CC = 0). .EN is cleared (rung-condition-in is false).	7	Rung-condition-in goes false and .DN or .ER is set.. .EN is cleared.
4	Rung-condition-in is true. .DN or .ER was previously set. .EN is set. .EW is setCconnection is opened*. .DN or .ER is cleared.	N/A	N/A

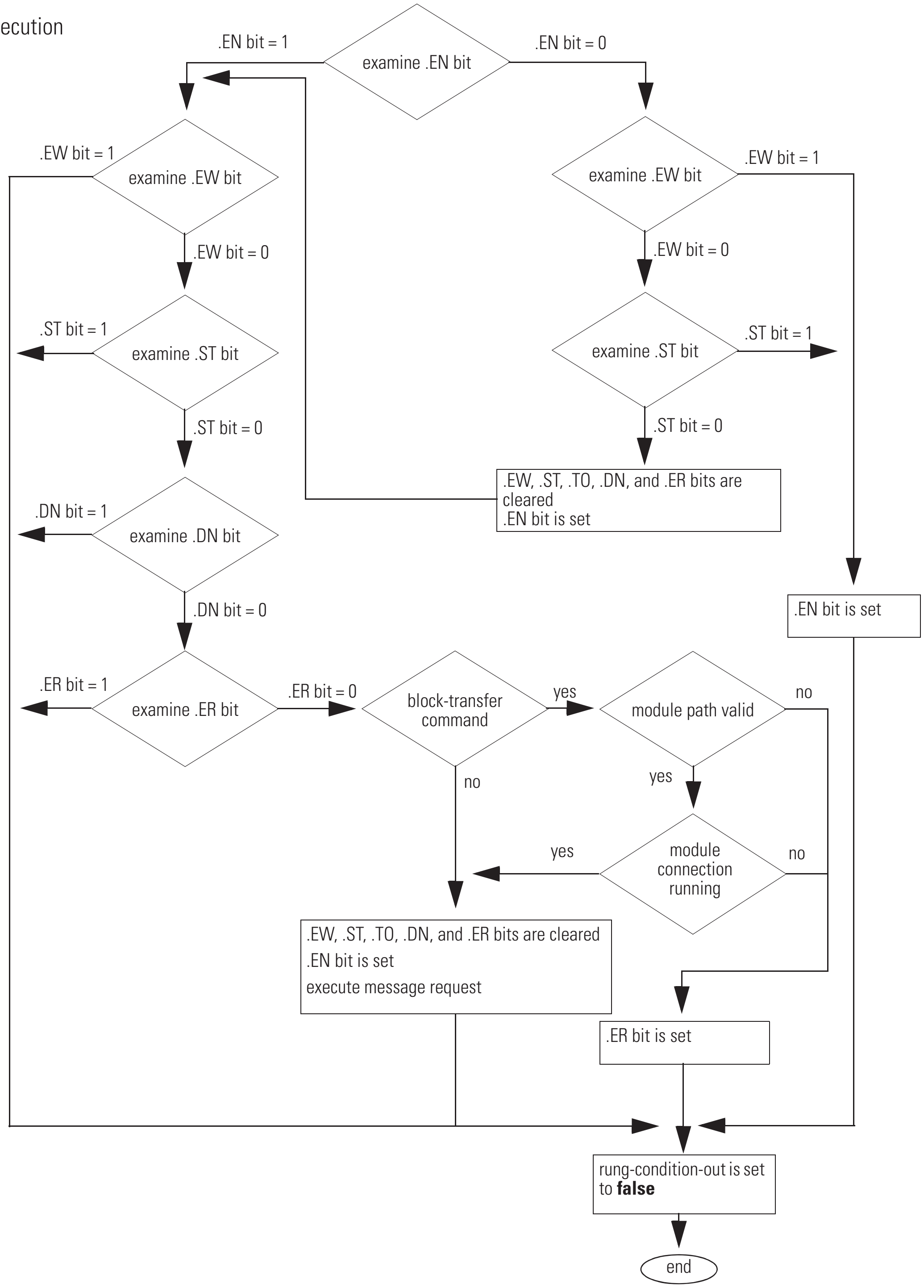
Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.

Condition	**Relay Ladder Action**	**Structured Text Action**
Rung-condition-in is false.		
Rung-condition-in is true.	The instruction executes. The rung-condition-out is set to true.	N/A

Condition	Relay Ladder Action	Structured Text Action
EnableIn is set.	N/A	EnableIn is always set. The instruction executes.

instruction execution



Postscan	The rung-condition-out is set to false.	No action taken.

Arithmetic Status Flags: Not affected

Fault Conditions: None

MSG Error Codes

The error codes depend on the type of MSG instruction.

Error Codes

The RSLogix 5000 programming software does not always display the full description.

Error Code (Hex)	Description	Display In Software
0001	Connection failure (see extended error codes)	Same as description
0002	Insufficient resource	
0003	Invalid value	
0004	IOI syntax error (see extended error codes)	
0005	Destination unknown, class unsupported, instance undefined or structure element undefined (see extended error codes)	
0006	Insufficient packet space	
0007	Connection lost	
0008	Service unsupported	
0009	Error in data segment or invalid attribute value	
000A	Attribute list error	
000B	State already exists	
000C	Object model conflict	
000D	Object already exists	
000E	Attribute cannot be set	
000F	Permission denied	
0010	Device state conflict	
0011	Reply will not fit	
0012	Fragment primitive	
0013	Insufficient command data	
0014	Attribute not supported	
0015	Too much data	
001A	Bridge request too large	
001B	Bridge response too large	
001C	Attribute list shortage	

Error Code (Hex)	Description	Display In Software
001D	Invalid attribute list	Same as description
001E	Embedded service error	
001F	Connection related failure (see extended error codes)	
0022	Invalid reply received	
0025	Key segment error	
0026	Invalid IOI error	
0027	Unexpected attribute in list	
0028	DeviceNet error - invalid member ID	
0029	DeviceNet error - member not settable	
00D1	Module not in run state	Unknown error
00FB	Message port not supported	
00FC	Message unsupported data type	
00FD	Message uninitialized	
00FE	Message timeout	
00FF	General error (see extended error codes)	

Extended Error Codes

The RSLogix 5000 programming software does not display any text for the extended error codes.

These are the extended error codes for error code **0001**.

Extended Error Code (Hex)	Description
0100	Connection in use
0103	Transport not supported
0106	Ownership conflict
0107	Connection not found
0108	Invalid connection type
0109	Invalid connection size
0110	Module not configured
0111	EPR not supported
0114	Wrong module
0115	Wrong device type
0116	Wrong revision
0118	Invalid configuration format
011A	Application out of connections

Extended Error Code (Hex)	Description
0203	Connection timeout
0204	Unconnected message timeout
0205	Unconnected send parameter error
0206	Message too large
0301	No buffer memory
0302	Bandwidth not available
0303	No screeners available
0305	Signature match
0311	Port not available
0312	Link address not available
0315	Invalid segment type
0317	Connection not scheduled

These are the extended error codes for error code **001F**.

Extended Error Code (Hex)	Description
0203	Connection timeout

These are the extended error codes for error code **0004** and **0005**.

Extended Error Code (Hex)	Description
0000	extended status out of memory
0001	extended status out of instances

These are the extended error codes for error code **00FF**.

Extended Error Code (Hex)	Description
2001	Excessive IOI
2002	Bad parameter value
2018	Semaphore reject
201B	Size too small
201C	Invalid size
2100	Privilege failure
2101	Invalid keyswitch position
2102	Password invalid
2103	No password issued
2104	Address out of range
2105	Address and how many out of range
2106	Data in use
2107	Type is invalid or not supported

Extended Error Code (Hex)	Description
2108	Controller in upload or download mode
2109	Attempt to change number of array dimensions
210A	Invalid symbol name
210B	Symbol does not exist
210E	Search failed
210F	Task cannot start
2110	Unable to write
2111	Unable to read
2112	Shared routine not editable
2113	Controller in faulted mode
2114	Run mode inhibited

PLC and SLC Error Codes (.ERR)

Logix firmware revision 10.x and later provides new error codes for errors that are associated with PLC and SLC message types (PCCC messages).

- This change lets RSLogix 5000 software display a more meaningful description for many of the errors. Previously the software did not give a description for any of the errors associated with the 00F0 error code.
- The change also makes the error codes more consistent with errors returned by other controllers, such as PLC-5 controllers.

The following table shows the change in the error codes from R9.x and earlier to R10.x and later. As a result of the change, the .ERR member returns a unique value for each PCCC error. The .EXERR is no longer required for these errors.

PLC and SLC Error Codes (hex)

R9.x And Earlier		R10.x And Later		Description
.ERR	.EXERR	.ERR	.EXERR	
0010		1000		Illegal command or format from local processor
0020		2000		Communication module not working
0030		3000		Remote node is missing, disconnected, or shut down
0040		4000		Processor connected but faulted (hardware)
0050		5000		Wrong station number
0060		6000		Requested function is not available
0070		7000		Processor is in Program mode
0080		8000		Processor's compatibility file does not exist
0090		9000		Remote node cannot buffer command
00B0		B000		Processor is downloading so it is not accessible
00F0	0001	F001		Processor incorrectly converted the address
00F0	0002	F002		Incomplete address
00F0	0003	F003		Incorrect address
00F0	0004	F004		Illegal address format - symbol not found
00F0	0005	F005		Illegal address format - symbol has 0 or greater than the maximum number of characters supported by the device
00F0	0006	F006		Address file does not exist in target processor
00F0	0007	F007		Destination file is too small for the number of words requested
00F0	0008	F008		Cannot complete request Situation changed during multipacket operation
00F0	0009	F009		Data or file is too large Memory unavailable

PLC and SLC Error Codes (hex) (Continued)

R9.x And Earlier		R10.x And Later		Description
.ERR	.EXERR	.ERR	.EXERR	
00F0	000A	F00A		Target processor cannot put requested information in packets
00F0	000B	F00B		Privilege error; access denied
00F0	000C	F00C		Requested function is not available
00F0	000D	F00D		Request is redundant
00F0	000E	F00E		Command cannot be executed
00F0	000F	F00F		Overflow; histogram overflow
00F0	0010	F010		No access
00F0	0011	F011		Data type requested does not match data available
00F0	0012	F012		Incorrect command parameters
00F0	0013	F013		Address reference exists to deleted area
00F0	0014	F014		Command execution failure for unknown reason PLC-3 histogram overflow
00F0	0015	F015		Data conversion error
00F0	0016	F016		The scanner is not available to communicate with a 1771 rack adapter
00F0	0017	F017		The adapter is no available to communicate with the module
00F0	0018	F018		The 1771 module response was not valid
00F0	0019	F019		Duplicate label
00F0	001A	F01A		File owner active - the file is being used
00F0	001B	F01B		Program owner active - someone is downloading or editing online
00F0	001C	F01C		Disk file is write protected or otherwise not accessible (offline only)
00F0	001D	F01D		Disk file is being used by another application Update not performed (offline only)

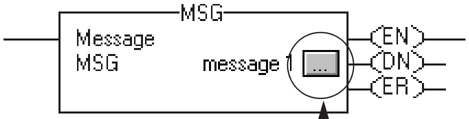
Block-Transfer Error Codes

These are the Logix5000 block-transfer specific error codes.

Error Code (Hex)	Description	Display In Software
00D0	The scanner did not receive a block-transfer response from the block-transfer module within 3.5 seconds of the request	Unknown error
00D1	The checksum from the read response did not match the checksum of the data stream	
00D2	The scanner requested either a read or write but the block-transfer module responded with the opposite	
00D3	The scanner requested a length and the block-transfer module responded with a different length	
00D6	The scanner received a response from the block-transfer module indicating the write request failed	
00EA	The scanner was not configured to communicate with the rack that would contain this block-transfer module	
00EB	The logical slot specified is not available for the given rack size	
00EC	There is currently a block-transfer request in progress and a response is required before another request can begin	
00ED	The size of the block-transfer request is not consistent with valid block-transfer size requests	
00EE	The type of block-transfer request is not consistent with the expected BT_READ or BT_WRITE	
00EF	The scanner was unable to find an available slot in the block-transfer table to accommodate the block-transfer request	
00F0	The scanner received a request to reset the remote I/O channels while there were outstanding block-transfers	
00F3	Queues for remote block-transfers are full	
00F5	No communication channels are configured for the requested rack or slot	
00F6	No communication channels are configured for remote I/O	
00F7	The block-transfer timeout, set in the instruction, timed out before completion	
00F8	Error in block-transfer protocol - unsolicited block-transfer	
00F9	Block-transfer data was lost due to a bad communication channel	
00FA	The block-transfer module requested a different length than the associated block-transfer instruction	
00FB	The checksum of the block-transfer read data was wrong	
00FC	There was an invalid transfer of block-transfer write data between the adapter and the block-transfer module	
00FD	The size of the block-transfer plus the size of the index in the block-transfer data table was greater than the size of the block-transfer data table file	

Specify the Configuration Details

After you enter the MSG instruction and specify the MESSAGE structure, use the Message Configuration dialog box to specify the details of the message.



Click here to configure the MSG instruction

The details you configure depend on the message type you select.

Message Type: CIP Data Table Write

Source Element:

Number Of Elements: 1

Destination Element:

42976

If the target device is	Select one of these message types	Page
Logix5000 controller	CIP Data Table Read	162
	CIP Data Table Write	
I/O module that you configure by using RSLogix 5000 software	Module Reconfigure	163
	CIP Generic	164
PLC-5 controller	PLC5 Typed Read	165
	PLC5 Typed Write	
	PLC5 Word Range Read	
	PLC5 Word Range Write	
SLC controller MicroLogix controller	SLC Typed Read	167
	SLC Typed Write	
Block-transfer module	Block-Transfer Read	167
	Block-Transfer Write	
PLC-3 processor	PLC3 typed read	168
	PLC3 typed write	
	PLC3 word range read	
	PLC3 word range write	
PLC-2 processor	PLC2 unprotected read	169
	PLC2 unprotected write	

Specify this Data for a Logix5000 Controller as a Target Device

Use this configuration information if a Logix5000 controller is the target device.

For this property	Specify
Source Element	• If you select a read message type, the Source Element is the address of the data you want to read in the target device. Use the addressing syntax of the target device. • If you select a write message type, the Source Tag is the first element of the tag that you want to send to the target device. If you select a write message type, the Source Tag is the first element of the tag that you want to send to the target device. • The only source data type that is not supported is Boolean. All other data types—SINT, INT, DINT, LINT, REAL—can be used. Any structure type predefined, module-defined, or user-defined also can be used to send messages.
Number of Elements	The number of elements you read/write depends on the type of data you are using. An element refers to one 'chunk' of related data. For example, tag timer1 is one element that consists of one timer control structure.
Destination Element	• If you select a read message type, the Destination Element is the first element of the tag in the Logix5000 controller where you want to store the data you read from the target device. • If you select a write message type, the Destination Element is the address of the location in the target device where you want to write the data.

Specify CIP Data Table Read and Write Messages

The CIP Data Table Read and Write message types transfer data between Logix5000 controllers.

Select this command	If you want to
CIP Data Table Read	Read data from another controller. The Source and Destination types must match.
CIP Data Table Write	Write data to another controller. The Source and Destination types must match.

Reconfigure an I/O Module

Use the Module Reconfigure message to send new configuration information to an I/O module.

During the reconfiguration:

- input modules continue to send input data to the controller.
- output modules continue to control their output devices.

A Module Reconfigure message requires this configuration properties.

In this property	Select
Message Type	Module Reconfigure

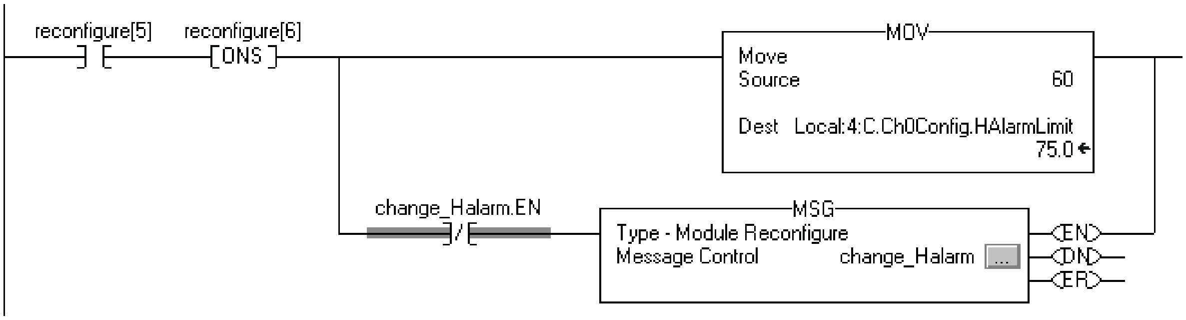
Example: Follow these steps to reconfigure an I/O module.

1. Set the required member of the configuration tag of the module to the new value.

2. Send a Module Reconfigure message to the module.

When *reconfigure[5]* is set, set the high alarm to 60 for the local module in slot 4. The Module Reconfigure message then sends the new alarm value to the module. The one shot instruction prevents the rung from sending multiple messages to the module while the *reconfigure[5]* is on.

Relay Ladder



Structured Text

```
IF reconfigure[5] AND NOT reconfigure[6]THEN
     Local:4:C.Ch0Config.HAlarmLimit := 60;
     IF NOT change_Halarm.EN THEN
          MSG(change_Halarm);
     END_IF;
END_IF;
     reconfigure[6] := reconfigure[5];
```

Specify CIP Generic Messages

A CIP Generic message performs a specific action on an I/O module.

<table>
<tr><th>If you want to</th><th>In this property</th><th colspan="2">Type or select</th></tr>
<tr><td rowspan="10">Perform a pulse test on a digital output module</td><td>Message Type</td><td colspan="2">CIP Generic</td></tr>
<tr><td>Service Type</td><td colspan="2">Pulse Test</td></tr>
<tr><td rowspan="7">Source</td><td colspan="2">*tag_name* of type INT [5]</td></tr>
<tr><td>**This array contains**</td><td>**Description**</td></tr>
<tr><td>*tag_name[0]*</td><td>Bit mask of points to test (test only one point at a time)</td></tr>
<tr><td>*tag_name[1]*</td><td>Reserved, leave 0</td></tr>
<tr><td>*tag_name[2]*</td><td>Pulse width (hundreds of µsecs, usually 20)</td></tr>
<tr><td>*tag_name[3]*</td><td>Zero cross delay for ControlLogix I/O (hundreds of µsecs, usually 40)</td></tr>
<tr><td>*tag_name[4]*</td><td>Verify delay</td></tr>
<tr><td>Destination</td><td>Leave blank</td><td></td></tr>
<tr><td rowspan="4">Reset electronic fuses on a digital output module</td><td>Message Type</td><td colspan="2">CIP Generic</td></tr>
<tr><td>Service Type</td><td>Reset Electronic Fuse</td><td></td></tr>
<tr><td>Source</td><td colspan="2">*tag name* of type DINT

This tag represents a bit mask of the points to reset fuses on.</td></tr>
<tr><td>Destination</td><td>Leave blank</td><td></td></tr>
<tr><td rowspan="3">Reset latched diagnostics on a digital input module</td><td>Message Type</td><td colspan="2">CIP Generic</td></tr>
<tr><td>Service Type</td><td colspan="2">Reset Latched Diagnostics (I)</td></tr>
<tr><td>Source</td><td colspan="2">*tag_name* of type DINT

This tag represents a bit mask of the points to reset diagnostics on.</td></tr>
<tr><td rowspan="3">Reset latched diagnostics on a digital output module</td><td>Message Type</td><td colspan="2">CIP Generic</td></tr>
<tr><td>Service Type</td><td colspan="2">Reset Latched Diagnostics (O)</td></tr>
<tr><td>Source</td><td colspan="2">*tag_name* of type DINT

This tag represents a bit mask of the points to reset diagnostics on.</td></tr>
</table>

If you want to	In this property	Type or select
Unlatch the alarm of an analog input module	Message Type	CIP Generic
	Service Type	Select which alarm that you want to unlatch. • Unlatch All Alarms (I) • Unlatch Analog High Alarm (I) • Unlatch Analog High High Alarm (I) • Unlatch Analog Low Alarm (I) • Unlatch Analog Low Low Alarm (I) • Unlatch Rate Alarm (I)
	Instance	Channel of the alarm that you want to unlatch
Unlatch the alarm of an analog output module	Message Type	CIP Generic
	Service Type	Select which alarm that you want to unlatch. • Unlatch All Alarms (O) • Unlatch High Alarm (O) • Unlatch Low Alarm (O) • Unlatch Ramp Alarm (O)
	Instance	Channel of the alarm that you want to unlatch

Specify PLC-5 Messages

Use the PLC-5 message types to communicate with PLC-5 controllers.

Select this command	If you want to
PLC5 Typed Read	Read 16-bit integer, floating-point, or string type data and maintain data integrity. See Data types for PLC5 Typed Read and Typed Write messages on page 166.
PLC5 Typed Write	Write 16-bit integer, floating-point, or string type data and maintain data integrity. See Data types for PLC5 Typed Read and Typed Write messages on page 166
PLC5 Word Range Read	Read a contiguous range of 16-bit words in PLC-5 memory regardless of data type. This command starts at the address specified as the Source Element and reads sequentially the number of 16-bit words requested. The data from the Source Element is stored, starting at the address specified as the Destination Tag.
PLC5 Word Range Write	Write a contiguous range of 16-bit words from Logix5000 memory regardless of data type to PLC-5 memory. This command starts at the address specified as the Source Tag and reads sequentially the number of 16-bit words requested. The data from the Source Tag is stored, starting at the address specified as the Destination Element in the PLC-5 processor.

The following table shows the data types to use with PLC5 Typed Read and PLC5 Typed Write messages.

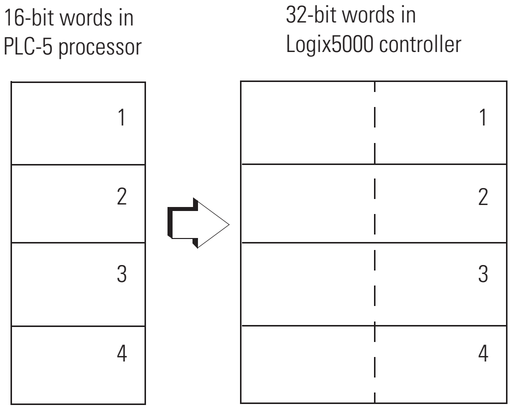
Data types for PLC5 Typed Read and Typed Write messages

For this PLC-5 data type	Use this Logix5000 data type
B	INT
F	REAL
N	INT
	DINT (Only write DINT values to a PLC-5 controller if the value is ≥ –32,768 and ≤32,767.)
S	INT
ST	STRING

The Typed Read and Typed Write commands also work with SLC 5/03 processors (OS303 and above), SLC 5/04 processors (OS402 and above), and SLC 5/05 processors.

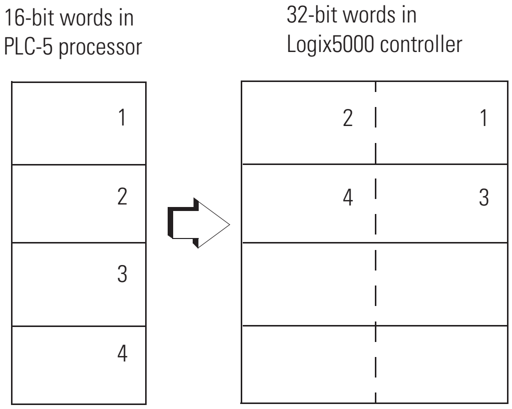
The following diagrams show how the typed and word-range commands differ. The example uses read commands from a PLC-5 processor to a Logix5000 controller.

Typed Read Command



The typed commands maintain data structure and value.

Word-range Read Command



The word-range commands fill the destination tag contiguously. Data structure and value change depending on the destination data type.

Specify SLC Messages

Use the SLC message types to communicate with SLC and MicroLogix controllers. The following table shows which data types that the instruction lets you access. The table also shows the corresponding Logix5000 data type.

For this SLC or MicroLogix data type	Use this Logix5000 data type
F	REAL
L (MicroLogix 1200 and 1500 controllers)	DINT
N	INT

Specify Block-transfer Messages

The block-transfer message types are used to communicate with block-transfer modules over a Universal Remote I/O network.

If you want to	Select this command
Read data from a block-transfer module This message type replaces the BTR instruction	Block-Transfer Read
Write data to a block-transfer module This message type replaces the BTW instruction	Block-Transfer Write

To configure a block-transfer message, follow these guidelines:

- The source (for BTW) and destination (for BTR) tags must be large enough to accept the requested data, except for MESSAGE, AXIS, and MODULE structures.
- Specify how many 16-bit integers (INT) to send or receive. You can specify from 0…64 integers.

If you want the	Then specify
Block-transfer module to determine how many 16-bit integers to send (BTR)	0 for the number of elements
Controller to send 64 integers (BTW)	

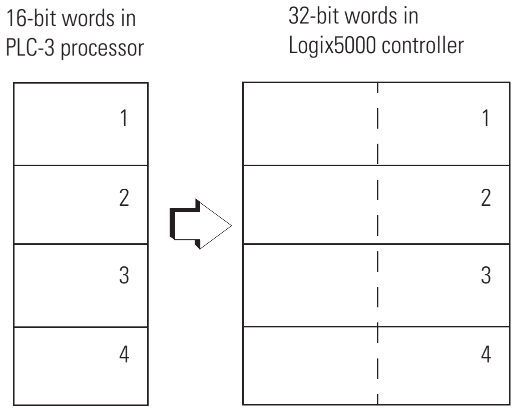
Specify PLC-3 Messages

The PLC-3 message types are designed for PLC-3 processors.

Select this command	If you want to
PLC3 Typed Read	Read integer or REAL type data. For integers, this command reads 16-bit integers from the PLC-3 processor and stores them in SINT, INT, or DINT data arrays in the Logix5000 controller and maintains data integrity. This command also reads floating-point data from the PLC-3 and stores it in a REAL data type tag in the Logix5000 controller.
PLC3 Typed Write	Write integer or REAL type data. This command writes SINT or INT data, to the PLC-3 integer file and maintains data integrity. You can write DINT data as long as it fits within an INT data type (–32,768 ≥ data ≤32,767). This command also writes REAL type data from the Logix5000 controller to a PLC-3 floating-point file.
PLC3 Word Range Read	Read a contiguous range of 16-bit words in PLC-3 memory regardless of data type. This command starts at the address specified as the Source Element and reads sequentially the number of 16-bit words requested. The data from the Source Element is stored, starting at the address specified as the Destination Tag.
PLC3 Word Range Write	Write a contiguous range of 16-bit words from Logix5000 memory regardless of data type to PLC-3 memory. This command starts at the address specified as the Source Tag and reads sequentially the number of 16-bit words requested. The data from the Source Tag is stored, starting at the address specified as the Destination Element in the PLC-3 processor.

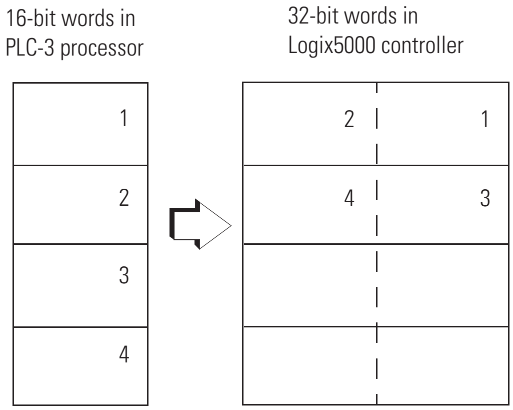
The following diagrams show how the typed and word-range commands differ. The example uses read commands from a PLC-3 processor to a Logix5000 controller.

Typed Read Command



The typed commands maintain data structure and value.

Word-range Read Command



The word-range commands fill the destination tag contiguously. Data structure and value change depending on the destination data type.

Specify PLC-2 Messages

The PLC-2 message types are designed for PLC-2 processors.

Select this command	If you want to
PLC2 Unprotected Read	Read 16-bit words from any area of the PLC-2 data table or the PLC-2 compatibility file of another processor.
PLC2 Unprotected Write	Write 16-bit words to any area of the PLC-2 data table or the PLC-2 compatibility file of another processor.

The message transfer uses 16-bit words, so make sure the Logix5000 tag appropriately stores the transferred data (typically as an INT array).

MSG Configuration Examples

The following examples show source and destination tags and elements for different controller combinations.

The table explains the path for MSG instructions originating from a Logix5000 controller and being writing to another controller.

Message Path	Example Source and Destination	
Logix5000 →Logix5000	Source tag	*array_1[0]*
	Destination tag	*array_2[0]*
	You can use an alias tag for the source tag (in originating Logix5000 controller). You **cannot** use an alias for the destination tag. The destination must be a base tag.	
Logix5000 →PLC-5 Logix5000 →SLC	Source tag	*array_1[0]*
	Destination element	*N7:10*
	You can use an alias tag for the source tag (in originating Logix5000 controller).	
Logix5000 →PLC-2	Source tag	*array_1[0]*
	Destination element	*010*

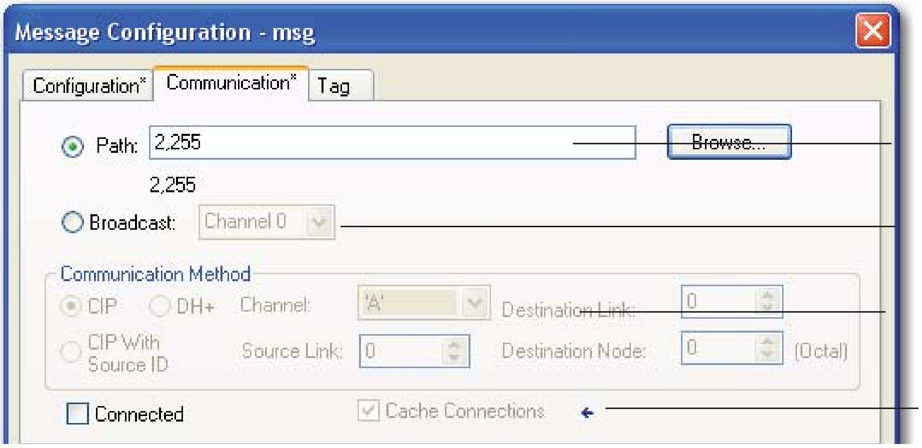
The table explains the path for MSG instructions originating from a Logix5000 controller and reading from another controller.

Message Path	Example Source and Destination	
Logix5000 →Logix5000	Source tag	*array_1[0]*
	Destination tag	*array_2[0]*
	You **cannot** use an alias tag for the source tag. The source must be a base tag. You can use an alias tag for the destination tag (in originating Logix5000 controller).	
Logix5000 →PLC-5 Logix5000 →SLC	Source element	*N7:10*
	Destination tag	*array_1[0]*
	You can use an alias tag for the destination tag (in originating Logix5000 controller).	
Logix5000 →PLC-2	Source element	*010*
	Destination tag	*array_1[0]*

Specify the Communication Details

You typically set up a broadcast in ladder logic or structured text programs. In ladder logic, you can add a rung and click on the MSG property to access the Message Configuration dialog box to set up a new message. In structured text, you type MSG(aMsg) and then right-click on aMsg to get the Message Configuration dialog to configure a message.

To configure a MSG instruction, you specify these details on the Communication tab.



● Path: 2,255

2,255

○ Broadcast: Channel 0

● CIP ○ DH+ Channel: 'A' Destination Link: 0

○ CIP With Source ID Source Link: 0 Destination Node: 0 (Octal)

☐ Connected ☑ Cache Connections

Specify a Path

Broadcast Button

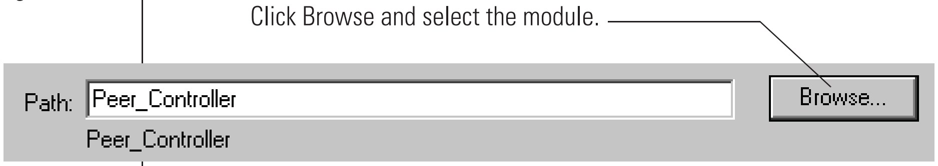
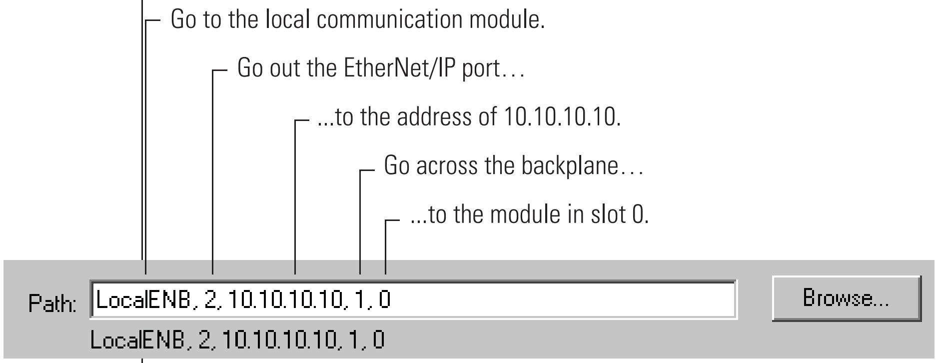
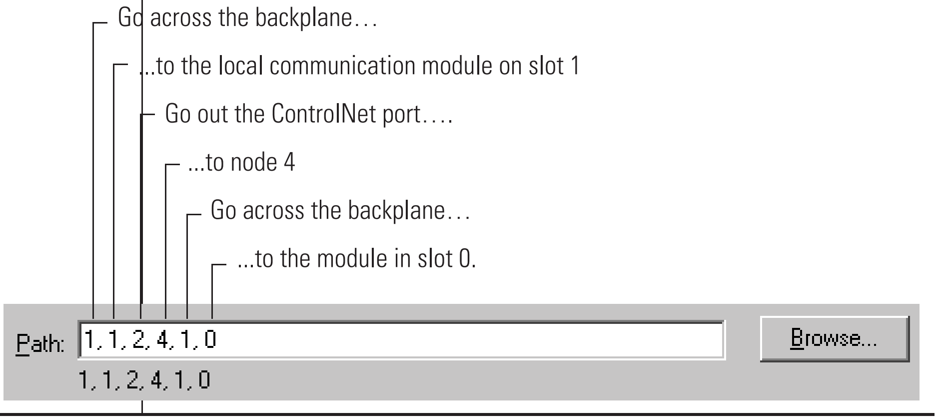
Specify a Communication Method or Module Address

Choose a Cache Option

Specify a Path

The path shows the route that the message takes to get to the destination. It uses either names from the I/O configuration of the controller, numbers that you type, or both. You can default the path by using the broadcast button, which must be enabled with the system protocol and message type.

If	Then
The I/O configuration of the controller has the module that gets the message.	Use Browse to select the module.
The I/O configuration of the controller has only the local communication module.	1. Use Browse to select the local communication module. 2. Type the rest of the path.
The I/O configuration of the controller doesn't have any of the modules that you need for the message.	Type the path.

Example	
The I/O configuration of the controller has the module that gets the message.	 Click Browse and select the module. Path: Peer_Controller Peer_Controller
The I/O configuration of the controller has only the local communication module.	 Go to the local communication module. Go out the EtherNet/IP port… ...to the address of 10.10.10.10. Go across the backplane… ...to the module in slot 0. Path: LocalENB, 2, 10.10.10.10, 1, 0 LocalENB, 2, 10.10.10.10, 1, 0
The I/O configuration of the controller doesn't have any of the modules that you need for the message.	 Go across the backplane… ...to the local communication module on slot 1 Go out the ControlNet port…. ...to node 4 Go across the backplane… ...to the module in slot 0. Path: 1, 1, 2, 4, 1, 0 1, 1, 2, 4, 1, 0

To type a path, use this format:

```
port, next_address, port, next_address, …
```

Where	**Is**	
	For this network	**Type**
`Port`	Backplane	1
	DF1 (serial, serial channel 0)	2
	ControlNet	
	EtherNet/IP	
	DH+ channel A	
	DH+ channel B	3
	DF1 channel 1 (serial channel 1)	
`Next_address`	Backplane	Slot number of the module
	DF1 (serial)	Station address (0-254)
	ControlNet	Node number (1-99 decimal)
	DH+	8# followed by the node number (1-77 octal) For example, to specify the octal node address of 37, type 8#37.
	EtherNet/IP	You can specify a module on an EtherNet/IP network by using any of these formats: IP address (for example, 10.10.10.10) IP address:Port (for example, 10.10.10.10:24) DNS name (for example, tanks) DNS name:Port (for example, tanks:24)

Broadcast Button

This functionality for RSLogix 5000 software, beginning with version 18, enhances the ability to define the route and message type that are required to send a message to its destination. You still can type in a path, such as ‘2,255’. However, manually entering a path is problematic for two reasons:

- the number entered into the path may be confusing.
- it’s error-prone, as you may forget to set up other conditions, such as system protocol and message type.

The Broadcast button, when enabled, allows you to default the path by selecting an available channel(s) in a combo box. The number of channels listed in the combo box depends on the current controller.

By default, the Path radio button on the Communication tab is active (dot in radio button). Do these steps to enable the Broadcast button and select a channel to default a path for the message.

1. On the Controller Organizer, right-click Controller, and choose Properties.

   The Controller Properties dialog box appears.

2. Click the System Protocol tab.

3. Choose DF1 Master in the Protocol box.

   The Polling mode defaults ‘Message Based’ (slave can initiate messages).

4. Click OK.

5. In ladder logic, click the box inside the MSG tag.

   The Message Configuration dialog box appears with the Configuration tab open.

6. In the Message Type box, choose CIP Data Table Write.(1)

   See page 161 for an example.

7. Click OK.

---

(1) See system protocol instructions on page 175 to enable the Broadcast button.

You have enabled the Broadcast button on the Communication tab.

8. Click the Communication tab.

9. Next to the Broadcast button, choose a channel in the combo box. The number of channels in the combo box depends on the controller.

   When you select channel 0 or 1, the corresponding message path on the Message Configuration dialog box defaults to 2,255 (channel 0) or 3,255 (channel 1). The Path grays out to not allow you to manually enter a path value.

10. Click OK.

System Protocol Tab Configuration

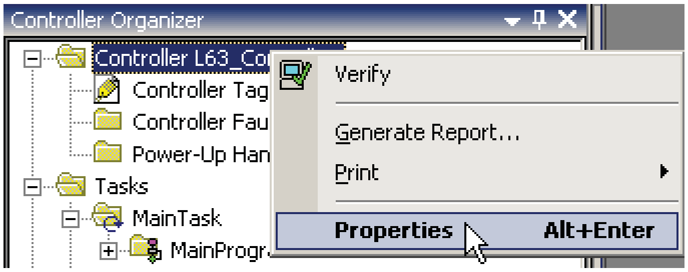
To run broadcast in ControlLogix in the RSLogix 5000 programming software, you must configure the System Protocol tab in the Controller Properties dialog box. The protocol must be compatible with the message type of 'write' on the Message Configuration dialog box.

See page 161 for an example in the Message Type box.

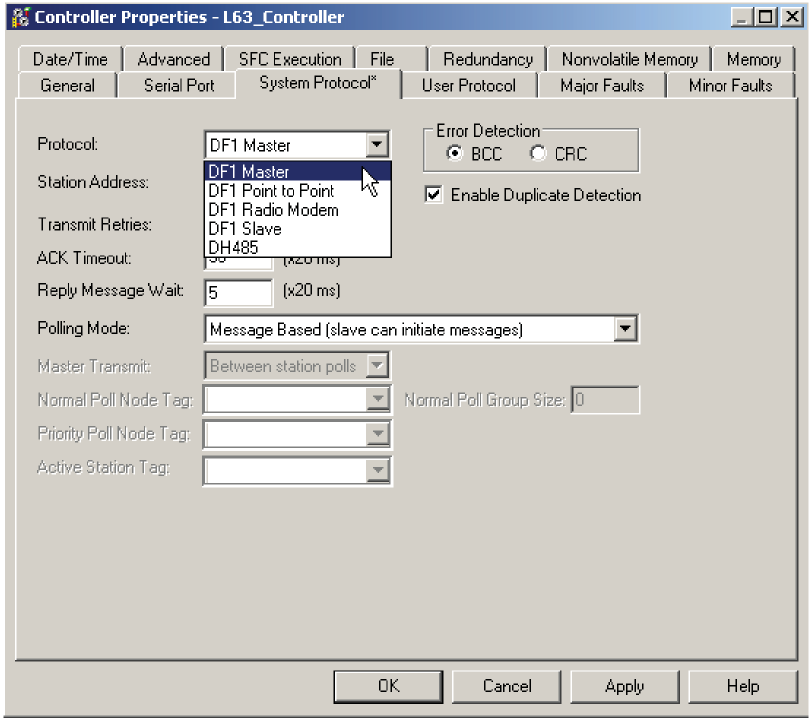
Follow these steps to set up the system protocol to be compatible with the broadcast feature.

1. Create or open an existing controller in the RSLogix 5000 programming software.

2. On the Controller Organizer, right-click the controller name, and choose Properties.



The Controller Properties dialog box appears.



Protocol: DF1 Master
DF1 Master
DF1 Point to Point
DF1 Radio Modem
DF1 Slave
DH485
Station Address:
Transmit Retries:
ACK Timeout: 50 (x20 ms)
Reply Message Wait: 5 (x20 ms)
Polling Mode: Message Based (slave can initiate messages)
Master Transmit: Between station polls
Normal Poll Node Tag:
Normal Poll Group Size: 0
Priority Poll Node Tag:
Active Station Tag:
Error Detection ● BCC ○ CRC
☑ Enable Duplicate Detection

3. Click System Protocol tab.

4. Choose a protocol from the Protocol box.

IMPORTANT The Message Type box on the Message Configuration Tab dialog box must be write-typed to be compatible with the system protocol. Otherwise, the Broadcast button is disabled.

5. Enter the information on the System Protocol tab for each protocol outlined below.

DF-1 Master

Topic	Description
Protocol	DF-1 Master
Station Address	Type controller station address number
Transmit Retries	3
ACK Timeout	50
Reply Message Wait	5

DF-1 Master

Topic	Description
Polling Mode	Choose from the following modes: • 'Message Based' to poll the slave using message instruction • 'Slave can initiate message' for slave to slave broadcast • 'Standard' to have the schedule poll for the slave
Error Detection	BCC
Duplicate Detection	Enabled (checked)

DF-1 Slave

Topic	Description
Protocol	DF-1 Slave
Station Address	Type controller station address number
Transmit Retries	3
Slave Poll Timeout	3000
EOT Suppression	Disable (unchecked)
Error Detection	BCC
Duplicate Detection	Enabled (checked)

DF-1 Radio Modem

Topic	Description
Protocol	DF-1 Slave
Station Address	Type controller station address number
Enable Store and Forward	Enable box (checkmark) to use store and forward tag
Error Detection	BCC

6. Click OK.

For Block Transfers

For block transfer messages, add the following modules to the I/O configuration of the controller.

For block-transfers over this network	**Add these modules to the I/O configuration**
ControlNet	• Local communication module (for example, 1756-CNB module) • Remote adapter module (for example, 1771-ACN module)
Universal remote I/O	• Local communication module (for example, 1756-DHRIO module) • One emote adapter module (for example, 1771-ASB module) for each rack, or portion of a rack, in the chassis • Block-transfer module (optional)

Specify a Communication Method or Module Address

Use the following table to select a communication method or module address for the message.

If the destination device is	**Then select**	**And specify**	
Logix5000 controller	**CIP**	No other specifications required.	
PLC-5 controller over an EtherNet/IP network			
PLC-5 controller over a ControlNet network			
SLC 5/05 controller			
PLC-5 controller over a DH+ network	**DH+**	Channel	Channel A or B of the 1756-DHRIO module that is connected to the DH+ network.
SLC controller over a DH+ network		Source Link	Link ID assigned to the backplane of the controller in the routing table of the 1756-DHRIO module. (The source node in the routing table is automatically the slot number of the controller.).
PLC-3 processor		Destination Link	Link ID of the remote DH+ link where the target device resides.
PLC-2 processor		Destination Node	Station address of the target device, in octal.
		If there is only one DH+ link and you did not use the RSLinx software to configure the DH/RIO module for remote links, specify 0 for both the Source Link and the Destination Link.	

If the destination device is	Then select	And specify	
Application on a workstation that is receiving an unsolicited message routed over an EtherNet/IP or ControlNet network through RSLinx software	**CIP with Source ID** (This lets the application receive data from a controller.)	Source Link	Remote ID of the topic in RSLinx software.
		Destination Link	Virtual Link ID set up in RSLinx software (0…65535).
		Destination Node	Destination ID (0…77 octal) provided by the application to RSLinx. For a DDE topic in RSLinx software, use 77.
		The slot number of the ControlLogix controller is used as the Source Node.	
Block transfer module over a universal remote I/O network	RIO	Channel	Channel A or B of the 1756-DHRIO module that is connected to the RIO network.
		Rack	Rack number (octal) of the module.
		Group	Group number of the module.
		Slot	Slot number that the module is in.
Block transfer module over a ControlNet network	ControlNet	Slot	Slot number that the module is in.

Choose a Cache Option

Depending on how you configure a MSG instruction, it may use a connection to send or receive data.

This type of message	And this communication method	Uses a connection
CIP data table read or write	→	Your option(1)
PLC-2, PLC-3, PLC-5, or SLC (all types)	CIP CIP with Source ID	
	DH+	X
CIP generic	→	Your option(2)
Block-transfer read or write	→	X

(1) CIP data table read or write messages can be connected or unconnected. But, for most applications, we recommend you leave CIP data table read or write messages connected.

(2) CIP generic messages can be connected or unconnected. But, for most applications, we recommend you leave CIP generic messages unconnected.

If a MSG instruction uses a connection, you have the option to leave the connection open (cache) or close the connection when the message is done transmitting.

If you	Then
Cache the connection	The connection stays open after the MSG instruction is done. This optimizes execution time. Opening a connection each time the message executes increases execution time.
Do not cache the connection	The connection closes after the MSG instruction is done. This frees up that connection for other uses.

The controller has the following limits on the number of connections that you can cache.

If you have this software and firmware revision	Then you can cache
11.*x* or earlier	• Block transfer messages for up to 16 connections. • Other types of messages for up to 16 connections.
12.*x* or later	Up to 32 connections.

If several messages go to the same device, the messages may be able to share a connection.

If the MSG instructions are to	And they are	Then
Different devices	→	Each MSG instruction uses 1 connection.
Same device	Enabled at the same time	Each MSG instruction uses 1 connection.
	NOT enabled at the same time	The MSG instruction uses 1 connection and 1 cached buffer. They share the connection and the buffer.

EXAMPLE

Share a Connection

If the controller alternates between sending a block-transfer read message and a block-transfer write message to the same module, then together both messages count as one connection. Caching both messages counts as one on the cache list.

Guidelines

As you plan and program your MSG instructions, follow these guidelines.

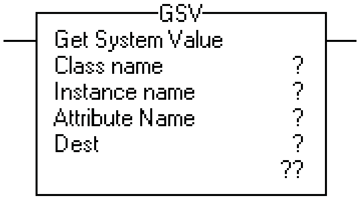
Guideline	Details
1. For each MSG instruction, create a control tag.	Each MSG instruction requires its own control tag.
	• Data type = MESSAGE
	• Scope = controller
	• The tag **cannot** be part of an array or a user-defined data type.
2. Keep the source and/or destination data at the controller scope.	A MSG instruction can access only tags that are in the Controller Tags folder (controller scope).
3. If your MSG is to a device that uses 16-bit integers, use a buffer of INTs in the MSG and DINTs throughout the project.	If your message is to a device that uses 16-bit integers, such as a PLC-5® or SLC 500™ controller, and it transfers integers (not REALs), use a buffer of INTs in the message and DINTs throughout the project.
	This increases the efficiency of your project because Logix controllers execute more efficiently and use less memory when working with 32-bit integers (DINTs).
	To convert between INTs and DINTs, see the Logix5000 Controllers Common Procedures Programming Manual, publication 1756-PM001.
4. Cache the connected MSGs that execute most frequently.	Cache the connection for those MSG instructions that execute most frequently, up to the maximum number permissible for your controller revision.
	This optimizes execution time because the controller does not have to open a connection each time the message executes.
5. If you want to enable more than 16 MSGs at one time, use some type of management strategy.	If you enable more than 16 MSGs at one time, some MSG instructions may experience delays in entering the queue. To guarantee the execution of each message, use one of these options:
	• Enable each message in sequence.
	• Enable the messages in groups.
	• Program a message to communicate with multiple devices. For more information, see the Logix5000 Controllers Common Procedures Programming Manual, publication 1756-PM001.
	• Program logic to coordinate the execution of messages. For more information, see the Logix5000 Controllers Common Procedures Programming Manual, publication 1756-PM001.
6. Keep the number of unconnected and uncached MSGs less than the number of unconnected buffers.	The controller can have 10…40 unconnected buffers. The default number is 10.
	• If all the unconnected buffers are in use when an instruction leaves the message queue, the instruction errors and does not transfer the data.
	• You can increase the number of unconnected buffers (40 max), but continue to follow guideline 5.
	• To increase the number of unconnected buffers, see the Logix5000 Controllers Common Procedures Programming Manual, publication 1756-PM001.

Get System Value (GSV) and Set System Value (SSV)

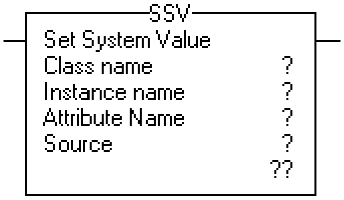
The GSV/SSV instructions get and set controller system data that is stored in objects.

Operands:

Relay Ladder



```
GSV
Get System Value
Class name          ?
Instance name       ?
Attribute Name      ?
Dest                ?
                   ??
```



```
SSV
Set System Value
Class name          ?
Instance name       ?
Attribute Name      ?
Source              ?
                   ??
```

Operand	Type	Format	Description
Class name		Name	Name of object.
Instance name		Name	Name of specific object, when object requires name.
Attribute Name		Name	Attribute of object. Data type depends on the attribute you select.
Destination (GSV)	SINT INT DINT REAL structure	Tag	Destination for attribute data.
Source (SSV)	SINT INT DINT REAL structure	Tag	Tag that contains data you want to copy to the attribute.

Structured Text

```
GSV(ClassName,InstanceName,AttributeName,Dest);
SSV(ClassName,InstanceName,AttributeName,Source);
```

The operands for are the same as those for the relay ladder GSV and SSV instructions.

Description: The GSV/SSV instructions get and set controller system data that is stored in objects. The controller stores system data in objects. There is no status file, as in the PLC-5 processor.

When enabled, the GSV instruction retrieves the specified information and places it in the destination. When enabled, the SSV instruction sets the specified attribute with data from the source.

When you enter a GSV/SSV instruction, the programming software displays the valid object classes, object names, and attribute names for each instruction. For the GSV instruction, you can get values for all the available attributes. For the SSV instruction, the software displays only those attributes are allowed to set (SSV).

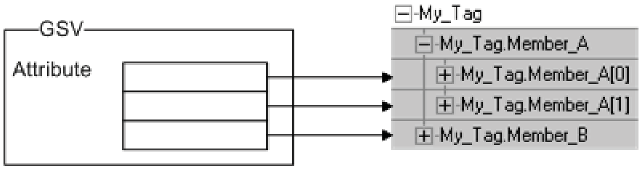
ATTENTION

Use the GSV and SSV instructions carefully. Making changes to objects may cause unexpected controller operation or injury to personnel.

You **must** test and confirm that the instructions don't change data that you don't want them to change.

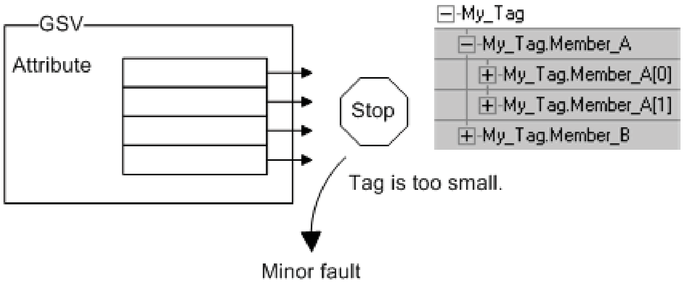
The GSV and SSV instructions write or read past a member into other members of a tag. If the tag is too small, the instructions don't write or read the data. They log a minor fault instead.

Example 1



Member_A is too small for the attribute. So the GSV instruction writes the last value to Member_B.

Example 2



My_Tag is too small for the attribute. So the GSV instruction stops and logs a minor fault.

The GSV/SSV Objects section shows each object's attributes and their associated data types. For example, the MajorFaultRecord attribute of the Program object needs a DINT[11] data type.

Arithmetic Status Flags: Not affected

Fault Conditions:

A minor fault will occur if	Fault type	Fault code
Invalid object address	4	5
Specified an object that does not support GSV/SSV	4	6
Invalid attribute	4	6
Did not supply enough information for an SSV instruction	4	6
The GSV destination was not large enough to hold the requested data	4	7

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction executes	Get or set the specified value.	Get or set the specified value.
Postscan	The rung-condition-out is set to false.	No action taken.

GSV/SSV Objects

When you enter a GSV/SSV instruction, you specify the object and its attribute that you want to access. In some cases, there will be more than one instance of the same type of object, so you might also have to specify the object name. For example, there can be several tasks in your application. Each task has its own TASK object that you access by the task name.

ATTENTION

For the GSV instruction, only the specified size of data is copied to the destination. For example, if the attribute is specified as a SINT and the destination is a DINT, only the lower 8 bits of the DINT destination are updated, leaving the remaining 24 bits unchanged.

You can access these objects.

GSV/SSV Objects

For information about this object	See this page or publication
AddOnInstructionDefinition	186
Axis	See the SERCOS Motion Configuration and Startup User Manual, publication MOTION-UM001.
Axis_Consumed	
Axis_Generic	
Axis_Generic_Drive	
Servo	
Servo_Drive	
Virtual	
Axis_CIP_Drive	See the CIP Motion Configuration and Startup User Manual, publication MOTION-UM003.
Controller	187
ControllerDevice	188
CoordinateSystem	See the Motion Coordinate System User Manual, publication MOTION-UM002.
CST	190
DF1	191
FaultLog	194
Message	195
Module	196
MotionGroup	See the Motion Reference Manual, publication MOTION-RM001.
Program	198
Redundancy	See the ControlLogix Enhanced Redundancy System User Manual, publication 1756-UM535.

GSV/SSV Objects

Routine	200
SafetyAddonInstructionDefinition, SafetyController, SafetyProgram, SafetyRoutine, SafetyTask.	See the GuardLogix Controllers User Manual, publication 1756-UM020.
SerialPort	202
Task	203
TimeSynchronization	See the Integrated Architecture and CIP Sync Configuration Application Technique, publication IA-AT003.
WallClockTime	205

AddOnInstructionDefintion Attributes

The AddOnInstructionDefinition object lets you customize instructions for sets of commonly-used logic, provides a common interface to this logic, and provides documentation for the instruction.

For details, see the Logix5000 Controllers Add-On Instructions Programming Manual, publication 1756-PM010.

Attribute	Data Type	Instruction within Standard Task	Instruction within Safety Task	Description
LastEditDate	LINT	GSV	None	Date and time stamp of the last edit to an Add-On Instruction definition.
MajorRevision	DINT	GSV	None	Major revision number of the Add-On Instruction.
MinorRevision	DINT	GSV	None	Minor revision number of the Add-On Instruction.
Name	String	GSV	GSV	Name of the Add-On Instruction.
RevisionExtended Text	String	GSV	None	Text describing the revision of the Add-On Instruction.
SafetySignature ID	DINT	GSV	None	In a safety project, the ID number, date, and timestamp of an Add-On Instruction definition.
SignatureID	DINT	GSV	None	32-bit identification number of an Add-On Instruction definition.
Vendor	String	GSV	None	Vendor that created the Add-On Instruction.

Controller Attributes

The Controller object provides status information about a controller's execution.

Attribute	Data Type	Instruction	Description
CanUseRPIFromProducer	DINT	GSV	Identifies whether to use the RPI specified by the producer. **Value** **Meaning** 0 Do not use the RPI specified by the producer 1 Use the RPI specified by the producer
ControllerLogExecutionModificationCount	DINT	GSV SSV	Number of controller log entries that originated from a program/task properties change, an online edit, or a controller timeslice change. It can also be configured to include log entries originating from forces. The number will be reset if RAM enters a bad state. The number is not capped at the largest DINT, and a rollover can occur.
ControllerLogTotalEntryCount	DINT	GSV SSV	Number of controller log entries since the last firmware upgrade. The number will be reset if RAM enters a bad state. The number is capped at the largest DINT.
DataTablePadPercentage	INT	GSV	Percentage (0…100) of free data table memory set aside.
InhibitAutomaticFirmwareUpdate	BOOL	GSV SSV	Identifies whether to enable the firmware supervisor. **Value** **Meaning** 0 Do not execute the firmware supervisor 1 Execute the firmware supervisor
KeepTestEditsOnSwitchover	SINT	GSV	Identifies whether to maintain test edits on controller switchover. **Value** **Meaning** 0 Automatically untest edits at switchover 1 Continue test edits at switchover
Name	String	GSV	Name of the controller.
RedundancyEnabled	SINT	GSV	Identifies whether the controller is configured for redundancy. **Value** **Meaning** 0 Not configured for redundancy 1 Configured for redundancy
ShareUnusedTimeSlice	INT	GSV SSV	Identifies how the continuous task and the background tasks shared any unused timeslice. **Value** **Meaning** 0 The operating system will not give control to the continuous task even if background is done (default) 1 Continuous task still runs if the background tasks are done. 2 The operating system will not give control to any tasks even if background is done
TimeSlice	INT	GSV SSV	Percentage of available CPU (10…90) that is assigned to communications. This value cannot be changed when the keyswitch is in the Run position.

ControllerDevice Attributes

The ControllerDevice object identifies the physical hardware of the controller.

Attribute	Data Type	Instruction	Description
DeviceName	SINT[33]	GSV	ASCII string that identifies the catalog number of the controller and memory board. The first byte contains a count of the number of ASCII characters returned in the array string.
ProductCode	INT	GSV	Identifies the type of controller: **Value** **Meaning** 15 SoftLogix5800 49 PowerFlex with DriveLogix5725 52 PowerFlex with DriveLogix5730 53 Emulator 54 1756-L61 ControlLogix 55 1756-L62 ControlLogix 56 1756-L63 ControlLogix 57 1756-L64 ControlLogix 64 1769-L31 CompactLogix 65 1769-L35E CompactLogix 67 1756-L61S GuardLogix 68 1756-L62S GuardLogix 69 1756-LSP GuardLogix 72 1768-L43 CompactLogix 74 1768-L45 CompactLogix 76 1769-L32C CompactLogix 77 1769-L32E CompactLogix 80 1769-L35CR CompactLogix 85 1756-L65 ControlLogix 86 1756-L63S GuardLogix 87 1769-L23E-QB1 CompactLogix 88 1769-L23-QBFC1 CompactLogix 89 1769-L23E-QBFC1 CompactLogix
ProductRev	INT	GSV	Identifies the current product revision. Display should be hexadecimal. The low byte contains the major revision; the high byte contains the minor revision.
SerialNumber	DINT	GSV	Serial number of the device. The serial number is assigned when the device is built.

Attribute	Data Type	Instruction	Description
Status	INT	GSV	**Device Status Bits** **7…4 Meaning** 0000 Reserved 0001 Flash update in progress 0010 Reserved 0011 Reserved 0100 Flash is bad 0101 Faulted modes 0110 Run 0111 Program **Controller Status Bits** **13…12 Meaning** 01 Keyswitch in run 10 Keyswitch in program 11 Keyswitch in remote **15…14 Meaning** 01 Controller is changing modes 10 Debug mode if controller in run mode **Fault Status Bits** **11…8 Meaning** 0001 Recoverable minor fault 0010 Unrecoverable minor fault 0100 Recoverable major fault 1000 Unrecoverable major fault
Type	INT	GSV	Identifies the device as a controller. Controller = 14.
Vendor	INT	GSV	Identifies the vendor of the device. Allen-Bradley = 0001.

CST Attributes

The coordinated system time (CST) object provides coordinated system time for the devices in one chassis.

Attribute	Data Type	Instruction	Description
CurrentStatus	INT	GSV	Current status of the coordinated system time. **Bit** **Meaning** 0 Timer hardware faulted: the device's internal timer hardware is in a faulted state 1 Ramping enabled: the current value of the timer's lower 16+ bits ramp up to the requested value, rather than snap to the lower value. 2 System time master: the CST object is a master time source in the ControlLogix system 3 Synchronized: the CST object's 64-bit CurrentValue is synchronized by master CST object via a system time update 4 Local network master: the CST object is the local network master time source 5 Relay mode: the CST object is acting in a time relay mode 6 Duplicate master detected: a duplicate local network time master was detected. This bit is always 0 for time-dependent nodes. 7 Unused 8-9 00 = time dependent node 01 = time master node 10 = time relay node 11 = Unused 10-15 Unused
CurrentValue	DINT[2]	GSV	Current value of the timer. DINT[0] contains the lower 32; DINT[1] contains the upper 32 bits. The timer source is adjusted to match the value supplied in update services and from local communication network synchronization. The adjustment is either a ramping to the requested value or an immediate setting to the request value, as reported in the CurrentStatus attribute.

DF1 Attributes

The DF1 object provides an interface to the DF1 communication driver that you can configure for the serial port.

Attribute	Data Type	Instruction	Description
ACKTimeout	DINT	GSV	The amount of time to wait for an acknowledgment to a message transmission (point-to-point and master only). Valid value 0-32,767. Delay in counts of 20 msec periods. Default is 50 (1 second).
Diagnostic Counters	INT[19]	GSV	Array of diagnostic counters for the DF1 communication driver.

Word offset	DF1 point-to-point	DF1 slave	Master
0	Signature (0x0043)	Signature (0x0042)	Signature (0x0044)
1	Modem bits	Modem bits	Modem bits
2	Packets sent	Packets sent	Packets sent
3	Packets received	Packets received	Packets received
4	Undelivered packets	Undelivered packets	Undelivered packets
5	Unused	Messages retried	Messages retried
6	NAKs received	NAKs received	Unused
7	ENQs received	Poll packets received	Unused
8	Bad packets NAKed	Bad packets not ACKed	Bad packets not ACKed
9	No memory sent NAK	No memory not ACKed	Unused
10	Duplicate packets received	Duplicate packets received	Duplicate packets received
11	Bad characters received	Unused	Unused
12	DCD recoveries count	DCD recoveries count	DCD recoveries count
13	Lost modem count	Lost modem count	Lost modem count
14	Unused	Unused	Priority scan time maximum
15	Unused	Unused	Priority scan time last
16	Unused	Unused	Normal scan time maximum
17	Unused	Unused	Normal scant time last
18	ENQs sent	Unused	Unused

Duplicate Detection	SINT	GSV	Enables duplicate message detection. **Value** **Meaning** 0 Duplicate message detection disabled Non zero Duplicate message detection enabled
Embedded ResponseEnable	SINT	GSV	Enables embedded response functionality (point-to-point only). **Value** **Meaning** 0 Initiated only after one is received (default) 1 Enabled unconditionally
EnableStoreFwd	SINT	GSV	Enables the store and forward behavior when receiving a message. **Value** **Meaning** 0 Do not forward message non zero See the store and forward table when receiving a message (default)
ENQTransmit Limit	SINT	GSV	The number of inquiries (ENQs) to send after an ACK timeout (point-to-point only). Valid value 0-127. Default setting is 3.

Attribute	Data Type	Instruction	Description
EOTSuppression	SINT	GSV	Enable suppressing EOT transmissions in response to poll packets (slave only). **Value** **Meaning** 0 EOT suppression disabled (disabled) Non zero EOT suppression enabled
ErrorDetection	SINT	GSV	Specifies the error-detection scheme. **Value** **Meaning** 0 BCC (default) 1 CRC
MasterMessage Transmit	SINT	GSV	Current value of the master message transmission (master only). **Value** **Meaning** 0 Between station polls (default) 1 In poll sequence (in place of master's station number)
MaxStation Address	SINT	GSV	Current value (0…31) of the maximum node address on a DH-485 network. Default is 31.
NAKReceiveLimit	SINT	GSV	The number of NAKs received in response to a message before stopping transmission (point-to-point communication only). Valid value 0…127. Default is 3.
NormalPollGroup Size	INT	GSV	Number of stations to poll in the normal poll node array after polling all the stations in the priority poll node array (master only). Valid value 0…255. Default is 0.
PollingMode	SINT	GSV	Current polling mode (master only). Default setting is 1. **Value** **Meaning** 0 Message-based, but don't allow slaves to initiate messages 1 Message-based, but allow slaves to initiate messages (default) 2 Standard, single-message transfer per node scan 3 Standard, multiple-message transfer per node scan
ReplyMessage Wait	DINT	GSV	The time (acting as a master) to wait after receiving an ACK before polling the slave for a response (master only). Valid value 0…65,535. Delay in counts of 20 msec periods. The default is 5 periods (100 msec).
SlavePollTimeout	DINT	GSV	The amount of time in msecs that the slave waits for the master to poll before the slave declares that it is unable to transmit because the master is inactive (slave only). Valid value 0…32,767. Delay in counts of 20 msec periods. The default is 3000 periods (1 minute).
StationAddress	INT	GSV	Current station address of the serial port. Valid value 0…254. Default is 0.
TokenHoldFactor	SINT	GSV	Current value (1…4) of the maximum number of messages sent by this node before passing the token on a DH-485 network. Default is 1.
TransmitRetries	SINT	GSV	Number of times to resend a message without getting an acknowledgment (master and slave only). Valid value 0…127. Default is 3.
PendingACK Timeout	DINT	SSV	Pending value for the ACKTimeout attribute.
Pending Duplicate Detection	SINT	SSV	Pending value for the DuplicateDetection attribute.

Attribute	**Data Type**	**Instruction**	**Description**
Pending Embedded ResponseEnable	SINT	SSV	Pending value for the EmbeddedResponse attribute.
PendingEnable StoreFwd	SINT	SSV	Pending value for the EnableStoreFwd attribute.
PendingENQ TransmitLimit	SINT	SSV	Pending value for the ENQTransmitLimit attribute.
PendingEOT Suppression	SINT	SSV	Pending value for the EOTSuppression attribute.
PendingError Detection	SINT	SSV	Pending value for the ErrorDetection attribute.
PendingMaster Message Transmit	SINT	SSV	Pending value for the MasterMessageTransmit attribute.
PendingMax StationAddress	SINT	SSV	Pending value for the MaxStationAddress attribute.
PendingNAK ReceiveLimit	SINT	SSV	Pending value for the NAKReceiveLimit attribute.
PendingNormal PollGroupSize	INT	SSV	Pending value for the NormalPollGroupSize attribute.
PendingPolling Mode	SINT	SSV	Pending value for the PollingMode attribute.
PendingReply MessageWait	DINT	SSV	Pending value for the ReplyMessageWait attribute.
PendingSlavePoll Timeout	DINT	SSV	Pending value for the SlavePollTimeout attribute.
PendingStation Address	INT	SSV	Pending value for the StationAddress attribute.
PendingToken HoldFactory	SINT	SSV	Pending value for the TokenHoldFactor attribute.
PendingTransmit Retries	SINT	SSV	Pending value for the TransmitRetries attribute.

FaultLog Attributes

The FaultLog object provides fault information about the controller.

Attribute	Data Type	Instruction	Description
MajorEvents	INT	GSV SSV	How many major faults have occurred since the last time this counter was reset.
MajorFaultBits	DINT	GSV SSV	Individual bits indicate the reason for the current major fault. **Bit** **Meaning** 1 Power loss 3 I/O 4 Instruction execution (program) 5 Fault handler 6 Watchdog 7 Stack 8 Mode change 11 Motion
MinorEvents	INT	GSV SSV	How many minor faults have occurred since the last time this counter was reset.
MinorFaultBits	DINT	GSV SSV	Individual bits indicate the reason for the current minor fault. **Bit** **Meaning** 4 Instruction execution (program) 6 Watchdog 9 Serial port 10 Battery

Message Attributes

You can access the Message object through the GSV/SSV instructions. Specify the message tag name to determine which Message object you want. The Message object provides an interface to setup and trigger peer-to-peer communications. This object replaces the MG data type of the PLC-5 processor.

Attribute	Data Type	Instruction	Description
ConnectionPath	SINT[130]	GSV SSV	Data to setup the connection path. The first two bytes (low byte and high byte) are the length in bytes of the connection path.
ConnectionRate	DINT	GSV SSV	Requested packet rate of the connection.
MessageType	SINT	GSV SSV	Specifies the type of message. **Value** **Meaning** 0 Not initialized
Port	SINT	GSV SSV	Indicates which port the message should be sent on. **Value** **Meaning** 1 Backplane 2 Serial port
Timeout Multiplier	SINT	GSV SSV	Determines when a connection should be considered timed out and closed. **Value** **Meaning** 0 Connection will timeout in 4 times the update rate default) 1 Connection will timeout in 8 times the update rate 2 Connection will timeout in 16 times the update rate
Unconnected Timeout	DINT	GSV SSV	Timeout in microseconds for all unconnected messages. The default is 30,000,000 microseconds (30 s).

Module Attributes

The Module object provides status information about a module. To select a particular Module object, set the Object Name operand of the GSV/SSV instruction to the module name. The specified module must be present in the I/O Configuration section of the controller organizer and must have a device name.

Attribute	Data Type	Instruction	Description
EntryStatus	INT	GSV	Specifies the current state of the specified map entry. The lower 12 bits should be masked when performing a comparison operation. Only bits 12…15 are valid. **Value** **Meaning** 16#0000 Standby: the controller is powering up. 16#1000 Faulted: any of the Module object's connections to the associated module fail. This value should not be used to determine if the module failed because the Module object leaves this state periodically when trying to reconnect to the module. Instead, test for Running state (16#4000). Check for FaultCode not equal to 0 to determine if a module is faulted. When Faulted, the FaultCode and FaultInfo attributes are valid until the fault condition is corrected. 16#2000 Validating: the Module object is verifying Module object integrity prior to establishing connections to the module. 16#3000 Connecting: the Module object is initiating connections to the module. 16#4000 Running: all connections to the module are established and data is transferring. 16#5000 Shutting down: the Module object is in the process of shutting down all connections to the module. 16#6000 Inhibited: the Module object is inhibited (the inhibit bit in the Mode attribute is set). 16#7000 Waiting: the parent object upon which this Module object depends is not running.
FaultCode	INT	GSV	A number that identifies a module fault, if one occurs.
FaultInfo	DINT	GSV	Provides specific information about the Module object fault code.
Firmware SupervisorStatus	INT	GSV	Identifies current operating state of the firmware supervisor feature. **Value** **Meaning** 0 Module updates are not being executed Module updates are being executed
ForceStatus	INT	GSV	Specifies the status of forces. **Bit** **Meaning** 0 Forces installed (1=yes, 0-no) 1 Forces enabled (1=yes, 0=no)
Instance	DINT	GSV	Provides the instance number of this module object.

Attribute	Data Type	Instruction	Description
LEDStatus	INT	GSV	Specifies the current state of the I/O status indicator on the front of the controller. **Value Meaning** 0 Status indicator off: No Module objects are configured for the controller. (There are no modules in the I/O Configuration section of the controller organizer.) 1 Flashing red: None of the Module objects are Running. 2 Flashing green: At least one Module object is not Running. 3 Solid green: All the Module objects are Running. You do not enter an object name with this attribute because this attribute applies to the entire collection of modules.
Mode	INT	GSV SSV	Specifies the current mode of the Module object. **Bit Meaning** 0 If set, causes a major fault to be generated if any of the Module object connections fault while the controller is in Run mode. 2 If set, causes the Module object to enter Inhibited state after shutting down all the connections to the module.

Program Attributes

The Program object provides status information about a program. Specify the program name to determine which Program object that you want.

Attribute	Data Type	Instruction within Standard Task	Instruction within Safety Task	Description
DisableFlag	SINT	GSV SSV	None	Controls this program's execution. **Value** **Meaning** 0 Execution enabled 1 Execution disabled
Instance	DINT	GSV	GSV	Provides the instance number of this program object.
LastScanTime	DINT	GSV SSV	None	Time it took to execute this program the last time it was executed. Time is in microseconds.
MajorFault Record	DINT[11]	GSV SSV	GSV SSV	Records major faults for this program We recommend that you create a user-defined structure to simplify access to the MajorFaultRecord attribute:

Name	Data Type	Style	Description
TimeLow	DINT	Decimal	Lower 32 bits of fault timestamp value
TimeHigh	DINT	Decimal	Upper 32 bits of fault timestamp value
Type	INT	Decimal	Fault type (program, I/O, and so forth)
Code	INT	Decimal	Unique code for the fault (depends on fault type)
Info	DINT[8]	Hexadecimal	Fault specific information (depends on fault type and code)

Attribute	Data Type	Instruction within Standard Task	Instruction within Safety Task	Description
MinorFault Record	DINT[11]	GSV SSV	GSV SSV	Records minor faults for this program We recommend that you create a user-defined structure to simplify access to the MinorFaultRecord attribute:
Name	**Data Type**	**Style**	**Description**	
TimeLow	DINT	Decimal	Lower 32 bits of fault timestamp value	
TimeHigh	DINT	Decimal	Upper 32 bits of fault timestamp value	
Type	INT	Decimal	Fault type (program, I/O, and so forth)	
Code	INT	Decimal	Unique code for the fault (depends on fault type)	
Info	DINT[8]	Hexadecimal	Fault specific information (depends on fault type and code)	
MaxScanTime	DINT	GSV SSV	None	Maximum recorded execution time for this program. Time is in microseconds.
Name	String	GSV	GSV	Name of the program.

Routine Attributes

The Routine object provides status information about a routine. Specify the routine name to determine which Routine object that you want.

Attribute	Data Type	Instruction within Standard Task	Instruction within Safety Task	Description
Instance	DINT	GSV	GSV	Provides the instance number for this routine object. Valid values are 0…65,535.
Name	String	GSV	GSV	Name of the routine.
SFCPaused	INT	GSV	None	In an SFC routine, indicates whether the SFC is paused. **Value** **Meaning** 0 SFC is not paused 1 SFC is paused
SFCResuming	INT	GSV SSV	None	In an SFC routine, indicates whether the SFC is resuming execution. **Value** **Meaning** 0 SFC is not executing. This attribute is automatically set to 0 at the end of a scan in which the chart was executed 1 SFC is executing. Step and action timers will retain their previous value if configured to do so. This attribute is automatically set to 1 on the first scan after a chart is unpaused.

Safety Attributes

The Safety Controller object provides safety status and safety signature information. The SafetyTask and SafetyFaultRecord attributes can capture information about non-recoverable faults.

See the GuardLogix Controllers User Manual, publication 1756-UM020.

Attribute	Data Type	Instruction within Standard Task	Instruction within Safety Task	Description
SafetyLocked	SINT	GSV	None	Indicates whether the controller is safety locked or unlocked.
SafetyStatus	INT	GSV	None	Specifies the safety status as: **Value** **Meaning** 1000000000000000 Safety task OK. 1000000000000001 Safety task inoperable. 00000000000000000 Partner missing. 00000000000000001 Partner unavailable. 00000000000000010 Hardware incompatible. 00000000000000011 Firmware incompatible.
SafetySignature Exists	SINT	GSV	GSV	Indicates whether the safety task signature is present.
SafetySignature ID	DINT	GSV	None	32-bit identification number.
SafetySignature	String	GSV	None	32-bit identification number.
SafetyTaskFault Record	DINT[11]	GSV	None	Records safety task faults.

SerialPort Attributes

The SerialPort object provides an interface to the serial communication port.

Attribute	Data Type	Instruction	Description
BaudRate	DINT	GSV	Specifies the baud rate. Valid values are 110, 300, 600, 1200, 2400, 4800, 9600, and 19200 (default).
ComDriverID	SINT	GSV	Specifies the specific driver. **Value** **Meaning** 0xA2 DF1 (default) 0xA3 ASCII
DataBits	SINT	GSV	Specifies the number of bits of data per character. **Value** **Meaning** 7 7 data bits (ASCII only) 8 8 data bits (default)
DCDDelay	INT	GSV	Specifies the amount of time to wait for the data carrier detect (DCD) to become low before erroring the packet. The delay is in counts of 1 s packets. Default is 0 counter.
Parity	SINT	GSV	Specifies the parity. **Value** **Meaning** 0 No parity (no default) 1 Odd parity (ASCII only) 2 Even parity
RTSOffDelay	INT	GSV	Amount of time to delay turning off the RTS line after the last character has been transmitted. Valid value 0…32,767. Delay in counts of 20 msec periods. The default is 0 msec.
RTSSendDelay	INT	GSV	Amount of time to delay transmitting the first character of a message after turning on the RTS line. Valid value 0…32,767. Delay in counts of 20 msec periods. The default is 0 msec.
StopBits	SINT	GSV	Specifies the number of stop bits. **Value** **Meaning** 1 1 stop bit (default) 2 2 stop bits (ASCII only)
PendingBaudRate	DINT	SSV	Pending value for the BaudRate attribute.
PendingCOM DriverID	SINT	SSV	Pending value for the COMDriverID attribute.
PendingDataBits	SINT	SSV	Pending value for the DataBits attribute.
PendingDCD Delay	INT	SSV	Pending value for the DCDDelay attribute.
PendingParity	SINT	SSV	Pending value for the Parity attribute.
PendingRTSOff Delay	INT	SSV	Pending value for the RTSOffDelay attribute.
PendingRTSSend Delay	INT	SSV	Pending value for the RTSSendDelay attribute.
PendingStopBits	SINT	SSV	Pending value for the StopBits attribute.

Task Attributes

The Task object provides status information about a task. Specify the task name to determine which Task object that you want.

Attribute	Data Type	Instruction within Standard Task	Instruction within Safety Task	Description
DisableUpdate Outputs	DINT	GSV SSV	None	Enables or disables the processing of outputs at the end of a task. **Value** **Meaning** 0 Enable the processing of outputs at the end of the task Non zero Disable the processing of outputs at the end of the task
EnableTimeOut	DINT	GSV SSV	None	Enables or disables the timeout function of an event task. **Value** **Meaning** 0 Disable the timeout function Non zero Enable the timeout function
InhibitTask	DINT	GSV SSV	None	Prevents the task from executing. If a task is inhibited, the controller still prescans the task when the controller transitions from program to run or test mode. **Value** **Meaning** 0 Enable the task 0 (default) Non zero Inhibit (disable) the task
Instance	DINT	GSV	GSV	Provides the instance number of this task object. Valid values are 0…31.
LastScanTime	DINT	GSV SSV	None	Time it took to execute this task the last time it was executed. Time is in microseconds.
MaximumInterval	DINT[2]	GSV SSV	None	The maximum time interval between successive executions of the task. DINT[0] contains the lower 32 bits of the value; DINT[1] contains the upper 32 bits of the value. A value of 0 indicates 1 or less executions of the task.
MaximumScan Time	DINT	GSV SSV	None	Maximum recorded execution time for this program. Time is in microseconds.
MinimumInterval	DINT[2]	GSV SSV	None	The minimum time interval between successive executions of the task. DINT[0] contains the lower 32 bits of the value; DINT[1] contains the upper 32 bits of the value. A value of 0 indicates 1 or less executions of the task.
Name	String	GSV	GSV	Name of the task.
OverlapCount	DINT	GSV SSV	GSV SSV	Number of times that the task was triggered while it was still executing. Valid for an event or a periodic task. To clear the count, set the attribute to 0.
Priority	INT	GSV SSV	GSV	Relative priority of this task as compared to the other tasks. Valid values 0…15.

Attribute	**Data Type**	**Instruction within Standard Task**	**Instruction within Safety Task**	**Description**
Rate	DINT	GSV SSV	GSV	The time interval between executions of the task. Time is in microseconds.
StartTime	DINT[2]	GSV SSV	None	Value of WALLCLOCKTIME when the last execution of the task was started. DINT[0] contains the lower 32 bits of the value; DINT[1] contains the upper 32 bits of the value.
Status	DINT	GSV SSV	None	Status information about the task. Once the controller sets one of these bits, you must manually clear the bit. **Bit** **Meaning** 0 An EVENT instruction triggered the task (event task only) 1 A timeout triggered the task (event task only) 2 An overlap occurred for this task
Watchdog	DINT	GSV SSV	GSV	Time limit for execution of all programs associated with this task. Time is in microseconds. If you enter 0, these values are assigned: **Time** **Task Type** 0.5 sec Periodic 5.0 sec Continuous

WallClockTime Attributes

The WallClockTime object provides a timestamp that the controller can use for scheduling.

Attribute	Data Type	Instruction	Description
ApplyDST	SINT	GSV SSV	Identifies whether to enable daylight savings time. **Value** **Meaning** 0 Do not adjust for daylight savings time non zero Adjust for daylight savings time
CSTOffset	DINT[2]	GSV SSV	Positive offset from the CurrentValue of the CST object (coordinated system time, see page 190). DINT[0] contains the lower 32 bits of the value; DINT[1] contains the upper 32 bits of the value. Value in µs. The default is 0.
CurrentValue	DINT[2]	GSV SSV	Current value of the wall clock time. DINT[0] contains the lower 32 bits of the value; DINT[1] contains the upper 32 bits of the value. The value is the number of microseconds that have elapsed since 0000 hours 1 January 1972. The CST and WALLCLOCKTIME objects are mathematically related in the controller. For example, if you add the CST CurrentValue and the WALLCLOCKTIME CTSOffset, the result is the WALLCLOCKTIME CurrentValue.
DateTime	DINT[7]	GSV SSV	The date and time. **Value** **Meaning** DINT[0] Year DINT[1] Month (1…12) DINT[2] Day (1…31) DINT[3] Hour (0…23) DINT[4] Minute (0…59) DINT[5] Seconds (0…59) DINT[6] Microseconds (0…999,999)
DSTAdjustment	INT	GSV SSV	The number of minutes to adjust for daylight saving time.
LocalDateTime	DINT[7]	GSV SSV	Current adjusted local time. **Value** **Meaning** DINT[0] Year DINT[1] Month (1…12) DINT[2] Day (1…31) DINT[3] Hour (0…23) DINT[4] Minute (0…59) DINT[5] Seconds (0…59) DINT[6] Microseconds (0…999,999)
TimeZoneString	INT	GSV SSV	Time zone for the time value.

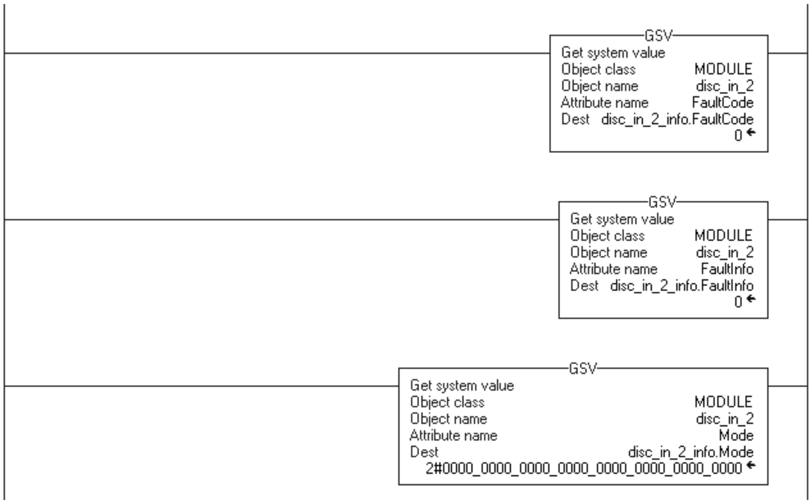
GSV/SSV Programming Example

Get Fault Information

The following examples use GSV instructions to get fault information.

Example 1: This example gets fault information from the I/O module *disc_in_2* and places the data in a user-defined structure *disc_in_2_info*.

Relay Ladder



GSV
Get system value
Object class MODULE
Object name disc_in_2
Attribute name FaultCode
Dest disc_in_2_info.FaultCode
0

GSV
Get system value
Object class MODULE
Object name disc_in_2
Attribute name FaultInfo
Dest disc_in_2_info.FaultInfo
0

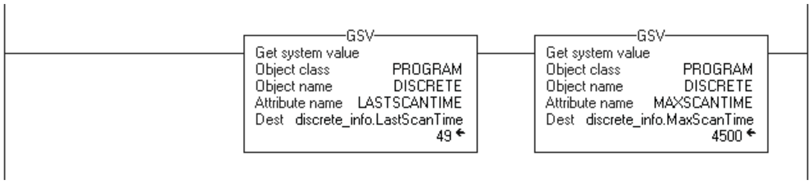
GSV
Get system value
Object class MODULE
Object name disc_in_2
Attribute name Mode
Dest disc_in_2_info.Mode
2#0000_0000_0000_0000_0000_0000_0000_0000

Structured Text

```
GSV(MODULE,disc_in_2,FaultCode,disc_in_2_info.FaultCode
);

GSV(MODULE,disc_in_2,FaultInfo,disc_in_2_info.FaultInfo
);

GSV(MODULE,disc_in_2,Mode,disc_in_2info.Mode);
```

Example 2: This example gets status information about program *discrete* and places the data in a user-defined structure *discrete_info*.

Relay Ladder

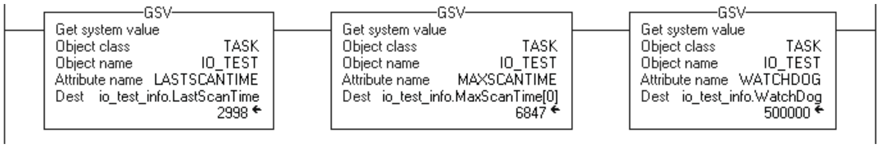


Structured Text

```
GSV(PROGRAM,DISCRETE,LASTSCANTIME,
     discrete_info.LastScanTime);

GSV(PROGRAM,DISCRETE,MAXSCANTIME,discrete_info.MaxScanT
ime);
```

Example 3: This example gets status information about task IO_test and places the data in a user-defined structure io_test_info.

Relay Ladder



Structured Text

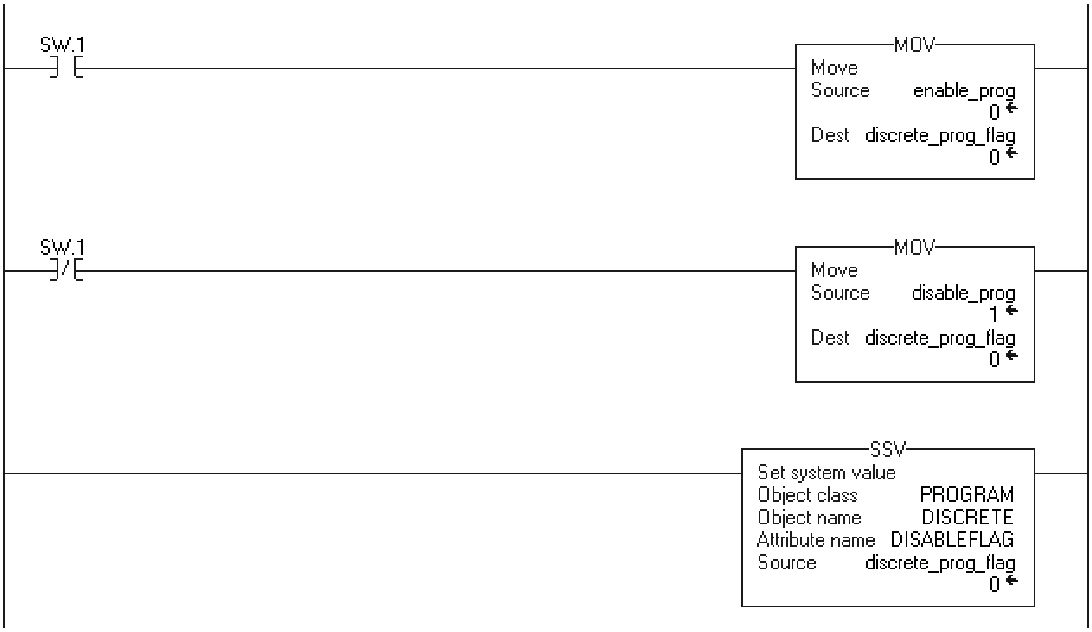
```
GSV(TASK,IO_TEST,LASTSCANTIME,io_test_info.LastScanTime
);

GSV(TASK,IO_TEST,MAXSCANTIME,io_test_info.MaxScanTime);

GSV(TASK,IO_TEST,WATCHDOG,io_test_info.WatchDog);
```

Set Enable And Disable Flags

The following example uses the SSV instruction to enable or disable a program. You could also use this method to enable or disable an I/O module, which is similar to using inhibit bits with a PLC-5 processor.

Example: Based on the status of *SW.1*, place the appropriate value in the *disableflag* attribute of program *discrete*.

Relay Ladder



Structured Text

```
IF SW.1 THEN

     discrete_prog_flag := enable_prog;

ELSE

     discrete_prog_flag := disable_prog;

END_IF;

SSV(PROGRAM,DISCRETE,DISABLEFLAG,discrete_prog_flag);
```

Immediate Output (IOT)

The IOT instruction immediately updates the specified output data (output tag or produced tag).

Operands:

Relay Ladder

Operand	Type	Format	Description
Update Tag		Tag	Tag that you want to update, either: • output tag of an I/O module • produced tag
			Do not choose a member or element of a tag. For example, Local:5:0 is OK but Local:5:0.Data is **not** OK.

`IOT(output_tag);`

Structured Text

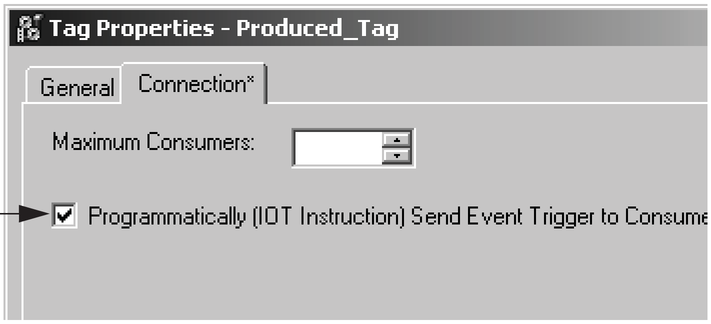
The operands are the same as those for the relay ladder IOT instruction.

Description:

The IOT instruction overrides the requested packet interval (RPI) of an output connection and sends fresh data over the connection.

- An output connection is a connection that is associated with the output tag of an I/O module or with a produced tag.
- If the connection is for a produced tag, the IOT instruction also sends the event trigger to the consuming controller. This lets the IOT instruction trigger an event task in the consuming controller.

To use an IOT instruction and a produced tag to trigger an event task in a consumer controller, configure the produced tag as shown below.



Maximum Consumers:

☑ Programmatically (IOT Instruction) Send Event Trigger to Consume

Check this box.

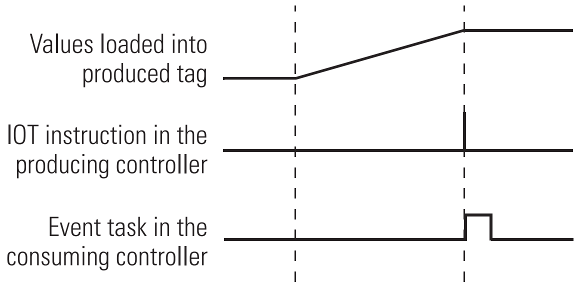
This configures the tag to update its event trigger only via an IOT instruction.

The type of network between the controllers determines when the consuming controller receives the new data and event trigger via the IOT instruction.

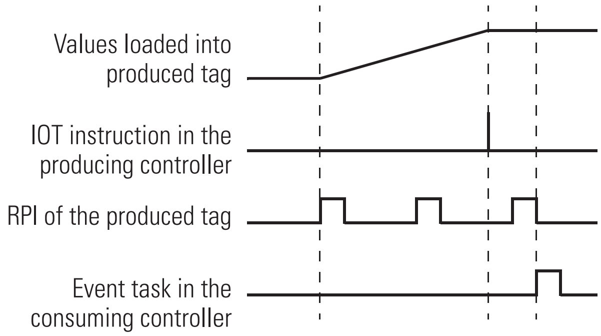
With this controller	Over this network	The consuming device receives the data and event trigger
ControlLogix	Backplane	Immediately
	EtherNet/IP network	Immediately
	ControlNet network	Within the actual packet interval (API) of the consumed tag (connection)
SoftLogix5800	You can produce and consume tags only over a ControlNet network.	Within the actual packet interval (API) of the consumed tag (connection)

The following diagrams compare the receipt of data via an IOT instruction over EtherNet/IP and ControlNet networks.

EtherNet/IP Network



ControlNet Network



Arithmetic Status Flags: Not affected

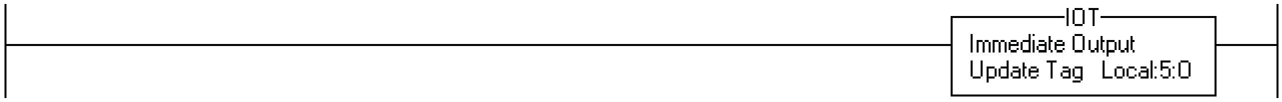
Fault Conditions: None

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction: • updates the connection of the specified tag. • resets the RPI timer of the connection.	
Postscan	The rung-condition-out is set to false.	No action taken.

Example 1: When the IOT instruction executes, it immediately sends the values of the Local:5:0 tag to the output module.

Relay Ladder



Structured Text

```
IOT (Local:5:O);
```

Example 2: This controller controls station 24 and produces data for the next station (station 25). To use an IOT instruction to signal the transmission of new data, the produced tag is configured as shown below:

Maximum Consumers:

☑ Programmatically (IOT Instruction) Send Event Trigger to Consume

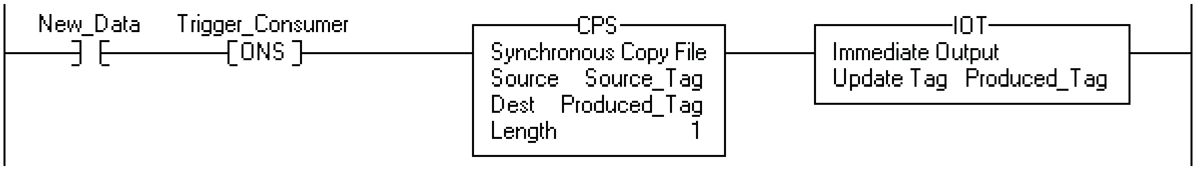
Produced_Tag is configured to update its event trigger via an IOT instruction.

Relay Ladder

If *New_Data* = on, then the following occurs for one scan:

The CPS instruction sets *Produced_Tag* = *Source_Tag*.

The IOT instruction updates *Produced_Tag* and sends this update to the consuming controller (station 25). When the consuming controller receives this update, it triggers the associated event task in that controller.



Structured Text

```
IF New_Data AND NOT Trigger_Consumer THEN
    CPS (Source_Tag,Produced_Tag,1);
    IOT (Produced_Tag);
END_IF;
    Trigger_Consumer := New_Data;
```

Compare Instructions

(CMP, EQU, GEQ, GRT, LEQ, LES, LIM, MEQ, NEQ)

Introduction

The compare instructions let you compare values by using an expression or a specific compare instruction.

If you want to	Use this instruction	Available in these languages	Page
Compare values based on an expression	CMP	Relay ladder Structured text[1]	214
Test whether two values are equal	EQU	Relay ladder Structured text[2] Function block	219
Test whether one value is greater than or equal to a second value	GEQ	Relay ladder Structured text[1] Function block	223
Test whether one value is greater than a second value	GRT	Relay ladder Structured text[1] Function block	227
Test whether one value is less than or equal to a second value	LEQ	Relay ladder Structured text[1] Function block	231
Test whether one value is less than a second value	LES	Relay ladder Structured text[1] Function block	235
Test whether one value is between two other values	LIM	Relay ladder Structured text[1] Function block	239
Pass two values through a mask and test whether they are equal	MEQ	Relay ladder Structured text[1] Function block	245
Test whether one value is not equal to a second value	NEQ	Relay ladder Structured text[1] Function block	250

(1) There is no equivalent structured text instruction. Use other structured text programming to achieve the same result. See the description for the instruction.

(2) There is no equivalent structured text instruction. Use the operator in an expression.

You can compare values of different data types, such as floating point and integer.

For relay ladder instructions, **bold** data types indicate optimal data types. An instruction executes faster and requires less memory if all the operands of the instruction use the same optimal data type, typically DINT or REAL.

Compare (CMP)

The CMP instruction performs a comparison on the arithmetic operations you specify in the expression.

Operands:

Relay Ladder

Operand	Type	Format	Description
Expression	SINT INT **DINT** **REAL** **string**	Immediate Tag	An expression consisting of tags and/or immediate values separated by operators.
	A SINT or INT tag converts to a DINT value by sign-extension.		

Structured Text

Structured text does not have a CMP instruction, but you can achieve the same results by using an IF...THEN construct and expression.

```
IF BOOL_expression THEN
    <statement>;
END_IF;
```

See Structured Text Programming for information on the syntax of constructs and expressions within structured text.

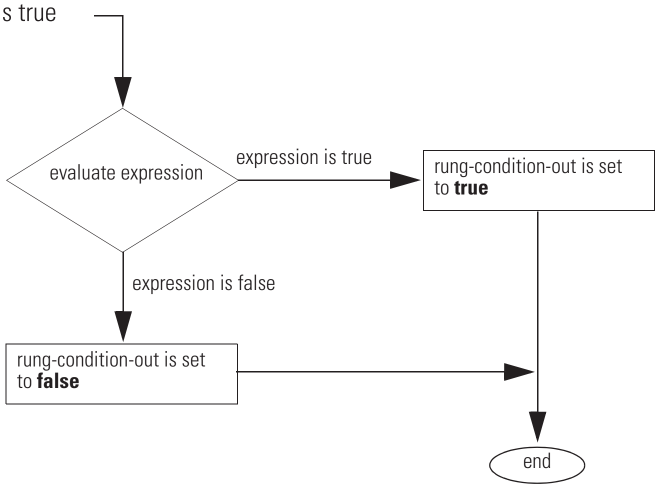
Description: Define the CMP expression by using operators, tags, and immediate values. Use parentheses ( ) to define sections of more complex expressions.

The execution of a CMP instruction is slightly slower and uses more memory than the execution of the other comparison instructions. The advantage of the CMP instruction is that it allows you to enter complex expressions in one instruction.

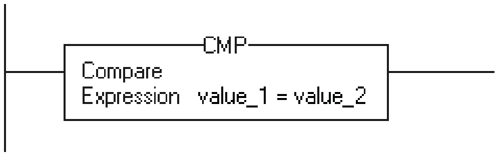
Arithmetic Status Flags: The CMP instruction affects only the arithmetic status flags if the expression contains an operator (for example, +, –, *, /) that affects the arithmetic status flags.

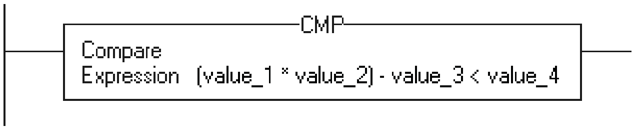
Fault Conditions: None

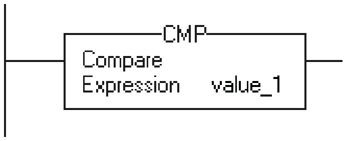
Execution:

Condition	Relay Ladder Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	
Postscan	The rung-condition-out is set to false.

Examples: If the CMP instruction finds the expression true, the rung-condition-out is set to true.







If you enter an expression without a comparison operator, such as *value_1* + *value_2*, or *value_1*, the instruction evaluates the expression as explained in the table.

If the expression is	The rung-condition-out is set to
Non-zero	True
Zero	False

CMP Expressions

You program expressions in CMP instructions the same as expressions in FSC instructions. Use the following sections for information on valid operators, format, and order of operation, which are common to both instructions.

Valid Operators

Operator:	Description	Optimal
+	Add	DINT, REAL
-	Subtract/negate	DINT, REAL
*	Multiply	DINT, REAL
/	Divide	DINT, REAL
=	Equal	DINT, REAL
<	Less than	DINT, REAL
<=	Less than or equal	DINT, REAL
>	Greater than	DINT, REAL
>=	Greater than or equal	DINT, REAL
<>	Not equal	DINT, REAL
**	Exponent (x to y)	DINT, REAL
ABS	Absolute value	DINT, REAL
ACS	Arc cosine	REAL
AND	Bitwise AND	DINT
ASN	Arc sine	REAL
ATN	Arc tangent	REAL
COS	Cosine	REAL

Operator:	Description	Optimal
DEG	Radians to degrees	DINT, REAL
FRD	BCD to integer	DINT
LN	Natural log	REAL
LOG	Log base 10	REAL
MOD	Modulo-divide	DINT, REAL
NOT	Bitwise complement	DINT
OR	Bitwise OR	DINT
RAD	Degrees to radians	DINT, REAL
SIN	Sine	REAL
SQR	Square root	DINT, REAL
TAN	Tangent	REAL
TOD	Integer to BCD	DINT
TRN	Truncate	DINT, REAL
XOR	Bitwise exclusive OR	DINT

Format Expressions

For each operator that you use in an expression, you have to provide one or two operands (tags or immediate values). Use this table to format operators and operands within an expression.

Operators that operate on	Use this format	Examples
One operand	Operator(operand)	ABS(*tag_a*)
Two operands	Operand_a operator operand_b	• *tag_b + 5* • *tag_c* AND *tag_d* • *(tag_e ** 2)* MOD *(tag_f / tag_g*)

Determine the Order of Operation

The operations you write into the expression are performed by the instruction in a prescribed order, not necessarily the order you write them. You can override the order of operation by grouping terms within parentheses, forcing the instruction to perform an operation within the parentheses ahead of other operations.

Operations of equal order are performed from left to right.

Order	Operation
1.	( )
2.	ABS, ACS, ASN, ATN, COS, DEG, FRD, LN, LOG, RAD, SIN, SQR, TAN, TOD, TRN
3.	**
4.	–(negate), NOT
5.	*, /, MOD
6.	<, <=, >, >=, =
7.	–(subtract), +
8.	AND
9.	XOR
10.	OR

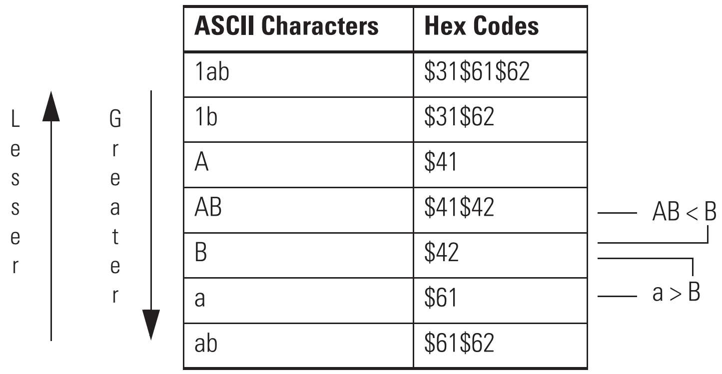
Use Strings in an Expression

Use a relay ladder or structured text expression to compare string data types. To use strings in an expression, follow these guidelines:

- An expression lets you compare two string tags.
- You **cannot** enter ASCII characters directly into the expression.
- Only the following operators are permitted.

Operator	Description
=	Equal
<	Less than
<=	Less than or equal
>	Greater than
>=	Greater than or equal
<>	Not equal

- Strings are equal if their characters match.
- ASCII characters are case sensitive. Upper case 'A' ($41) is **not** equal to lower case 'a' ($61).
- The hexadecimal values of the characters determine if one string is less than or greater than another string. For the hex code of a character, see the back cover of this manual.
- When the two strings are sorted as in a telephone directory, the order of the strings determines which one is greater.

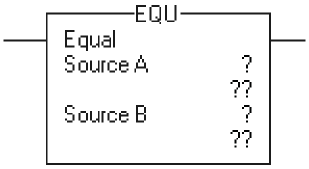


ASCII Characters	Hex Codes
1ab	$31$61$62
1b	$31$62
A	$41
AB	$41$42
B	$42
a	$61
ab	$61$62

Equal To (EQU)

The EQU instruction tests whether Source A is equal to Source B.

Operands:



Relay Ladder

Operand	Type	Format	Description
Source A	SINT INT **DINT** **REAL** **string**	Immediate Tag	Value to test against Source B
Source B	SINT INT **DINT** **REAL** **string**	Immediate Tag	Value to test against Source A

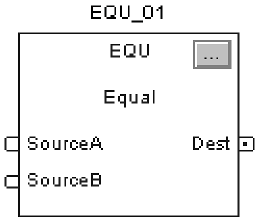
- If you enter a SINT or INT tag, the value converts to a DINT value by sign-extension.
- REAL values are rarely absolutely equal. If you need to determine the equality of two REAL values, use the LIM instruction.
- String data types are:
  - default STRING data type.
  - any new string data type that you create.
- To test the characters of a string, enter a string tag for both Source A and Source B.

Structured Text

```
IF sourceA = sourceB THEN
    <statements>;
```

Use the equal sign '=' as an operator within an expression. This expression evaluates whether *sourceA* is equal to *sourceB*.

See Structured Text Programming for information on the syntax of expressions within structured text.



Function Block

Operand	Type	Format	Description
EQU tag	FBD_COMPARE	Structure	EQU structure

FBD_COMPARE Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
SourceA	REAL	Value to test against SourceB. Valid = any float
SourceB	REAL	Value to test against SourceA. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	BOOL	Result of the instruction. This is equivalent to rung-condition-out of the relay ladder EQU instruction.

Description: Use the EQU instruction to compare two numbers or two strings of ASCII characters. When you compare strings:

- strings are equal if their characters match.
- ASCII characters are case sensitive. Upper case 'A' ($41) is **not** equal to lower case 'a' ($61).

Arithmetic Status Flags: Not affected

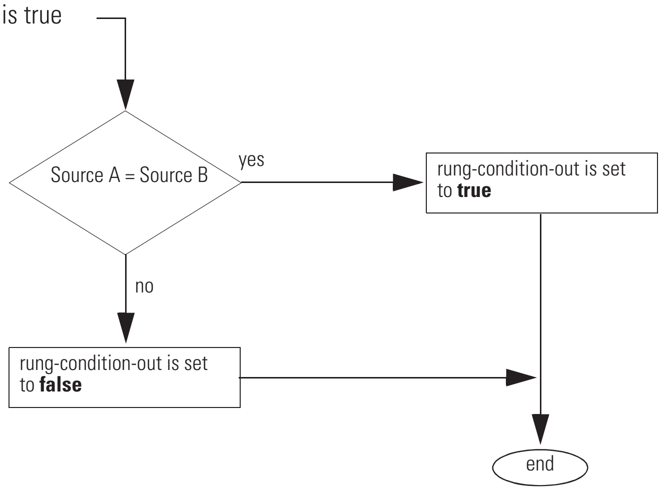
Fault Conditions: None

Execution:

Relay Ladder

Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.

Rung-condition-in is true



Condition	Action
Postscan	The rung-condition-out is set to false.

Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

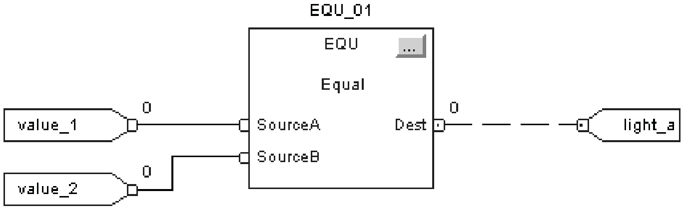
Example: If *value_1* is equal to *value_2*, set *light_a*. If *value_1* is not equal to *value_2*, clear *light_a*.

Relay Ladder

Structured Text

```
light_a := (value_1 = value_2);
```

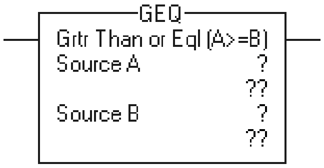
Function Block



Greater Than or Equal To (GEQ)

The GEQ instruction tests whether Source A is greater than or equal to Source B.

Operands:



Relay Ladder

Operand	Type	Format	Description
Source A	SINT INT **DINT** **REAL** **string**	Immediate Tag	Value to test against Source B
Source B	SINT INT **DINT** **REAL** **string**	Immediate Tag	Value to test against Source A

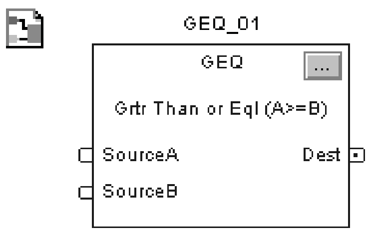
- If you enter a SINT or INT tag, the value converts to a DINT value by sign-extension.
- String data types are:
  - default STRING data type.
  - any new string data type that you create.
- To test the characters of a string, enter a string tag for both Source A and Source B.

Structured Text

```
IF sourceA >= sourceB THEN
    <statements>;
```

Use adjacent greater than and equal signs ‘>=’ as an operator within an expression. This expression evaluates whether *sourceA* is greater than or equal to *sourceB*.

See Structured Text Programming for information on the syntax of expressions within structured text.



Function Block

Operand	Type	Format	Description
GEQ tag	FBD_COMPARE	Structure	GEQ structure

FBD_COMPARE Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
SourceA	REAL	Value to test against SourceB. Valid = any float
SourceB	REAL	Value to test against SourceA. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	BOOL	Result of the instruction. This is equivalent to rung-condition-out for the relay ladder GEQ instruction.

Description: The GEQ instruction tests whether Source A is greater than or equal to Source B.

When you compare strings:

- the hexadecimal values of the characters determine if one string is less than or greater than another string. For the hex code of a character, see the back cover of this manual.
- when the two strings are sorted as in a telephone directory, the order of the strings determines which one is greater.

Lesser ↑ Greater ↓

ASCII Characters	Hex Codes
1ab	$31$61$62
1b	$31$62
A	$41
AB	$41$42
B	$42
a	$61
ab	$61$62

AB < B

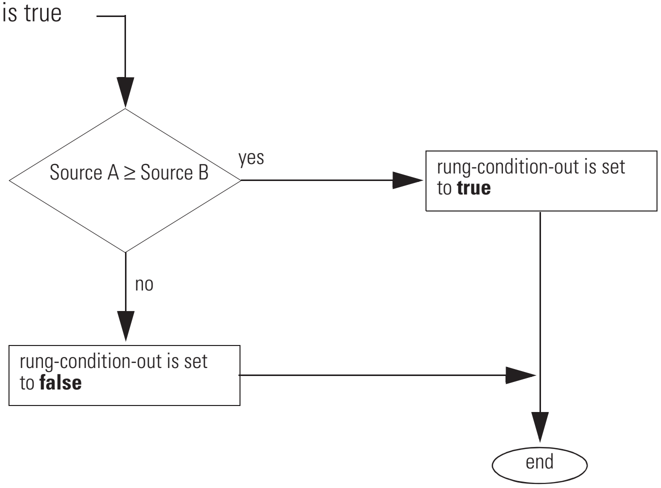
a > B

Arithmetic Status Flags: Not affected

Fault Conditions: None

Execution:

Relay Ladder

Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	
Postscan	The rung-condition-out is set to false.

Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

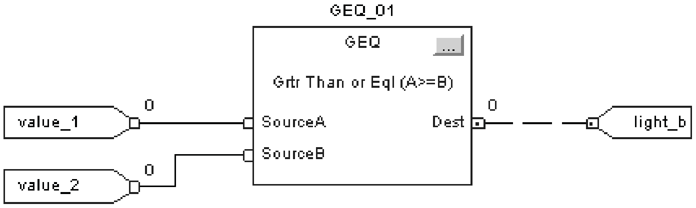
Example: If *value_1* is greater than or equal to *value_2*, set *light_b*. If *value_1* is less than *value_2*, clear *light_b*.

Relay Ladder

Structured Text

```
light_b := (value_1 >= value_2);
```

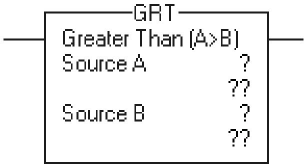
Function Block



Greater Than (GRT)

The GRT instruction tests whether Source A is greater than Source B.

Operands:



Relay Ladder

Operand	Type	Format	Description
Source A	SINT INT **DINT** **REAL** **string**	Immediate Tag	Value to test against Source B
Source B	SINT INT **DINT** **REAL** **string**	Immediate Tag	Value to test against Source A

- If you enter a SINT or INT tag, the value converts to a DINT value by sign-extension.
- String data types are:
  - default STRING data type.
  - any new string data type that you create.
- To test the characters of a string, enter a string tag for both Source A and Source B.

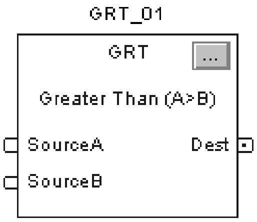
Structured Text

```
IF sourceA > sourceB THEN
    <statements>;
```

Use the greater than sign ‘>’ as an operator within an expression. This expression evaluates whether *sourceA* is greater than *sourceB*.

See Structured Text Programming for information on the syntax of expressions within structured text.

Function Block



Operand	Type	Format	Description
GRT tag	FBD_COMPARE	Structure	GRT structure

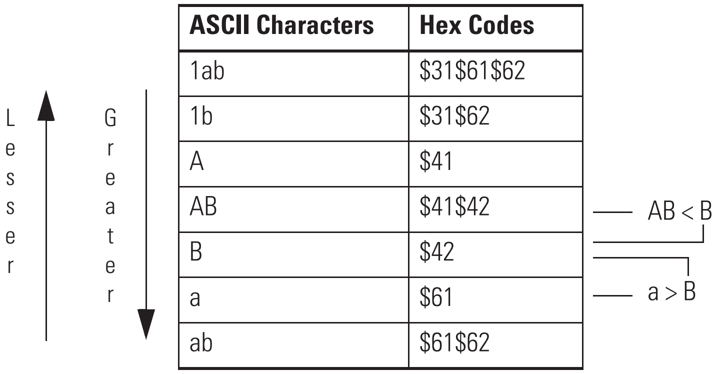
FBD_COMPARE Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
SourceA	REAL	Value to test against SourceB. Valid = any float
SourceB	REAL	Value to test against SourceA. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	BOOL	Result of the instruction. This is equivalent to rung-condition-out for the relay ladder GRT instruction.

Description: The GRT instruction tests whether Source A is greater than Source B.

When you compare strings:

- the hexadecimal values of the characters determine if one string is less than or greater than another string. For the hex code of a character, see the back cover of this manual.
- when the two strings are sorted as in a telephone directory, the order of the strings determines which one is greater.



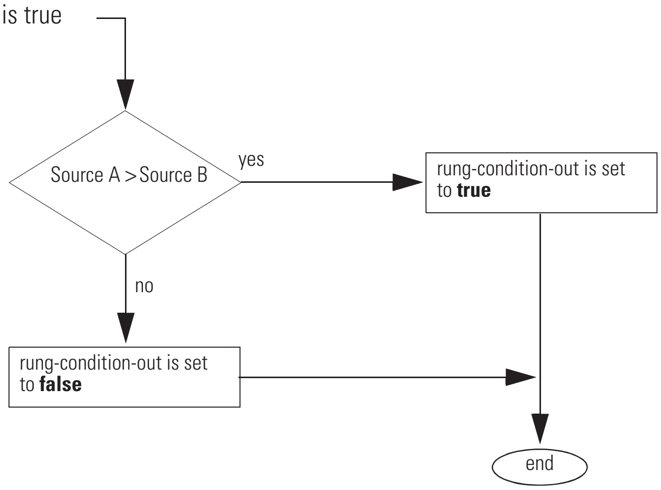
ASCII Characters	Hex Codes
1ab	$31$61$62
1b	$31$62
A	$41
AB	$41$42
B	$42
a	$61
ab	$61$62

Arithmetic Status Flags: Not affected

Fault Conditions: None

Execution:

Relay Ladder

Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	
Postscan	The rung-condition-out is set to false.

Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

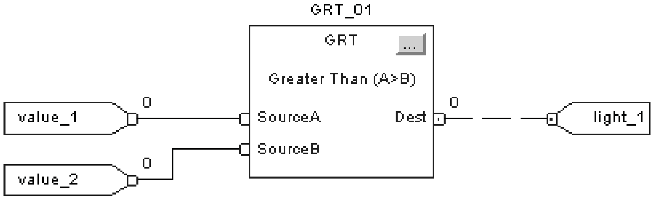
Example: If *value_1* is greater than *value_2***,** set *light_1*. If *value_1* is less than or equal to *value_2*, clear l*ight_1*.

Relay Ladder

Structured Text

```
light_1 := (value_1 > value_2);
```

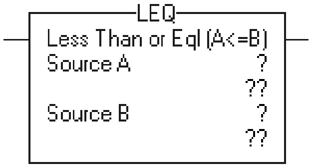
Function Block



Less Than or Equal To (LEQ)

The LEQ instruction tests whether Source A is less than or equal to Source B.

Operands:



Relay Ladder

Operand	Type	Format	Description
Source A	SINT INT **DINT** **REAL** **string**	Immediate Tag	Value to test against Source B
Source B	SINT INT **DINT** **REAL** **string**	Immediate Tag	Value to test against Source A

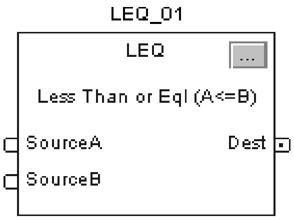
- If you enter a SINT or INT tag, the value converts to a DINT value by sign-extension.
- String data types are:
  - default STRING data type.
  - any new string data type that you create.
- To test the characters of a string, enter a string tag for both Source A and Source B.

Structured Text

```
IF sourceA <= sourceB THEN
     <statements>;
```

Use adjacent less than and equal signs ‘<=’ as an operator within an expression. This expression evaluates whether *sourceA* is less than or equal to *sourceB*.

See Structured Text Programming for information on the syntax of expressions within structured text.



Function Block

Operand	Type	Format	Description
LEQ tag	FBD_COMPARE	Structure	LEQ structure

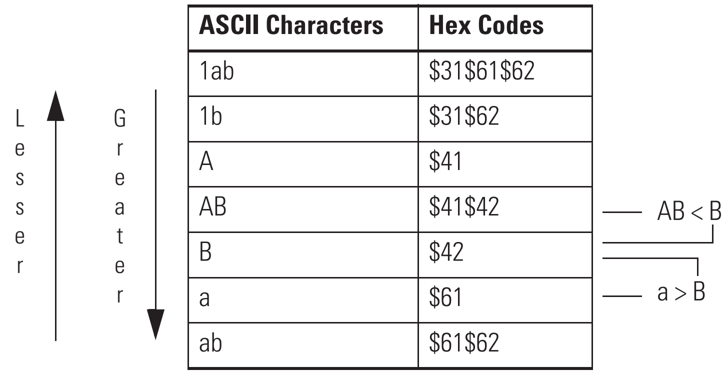
FBD_COMPARE Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
SourceA	REAL	Value to test against SourceB. Valid = any float
SourceB	REAL	Value to test against SourceA. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	BOOL	Result of the instruction. This is equivalent to rung-condition-out for the relay ladder LEQ instruction.

Description: The LEQ instruction tests whether Source A is less than or equal to Source B.

When you compare strings:

- the hexadecimal values of the characters determine if one string is less than or greater than another string. For the hex code of a character, see the back cover of this manual.
- when the two strings are sorted as in a telephone directory, the order of the strings determines which one is greater.



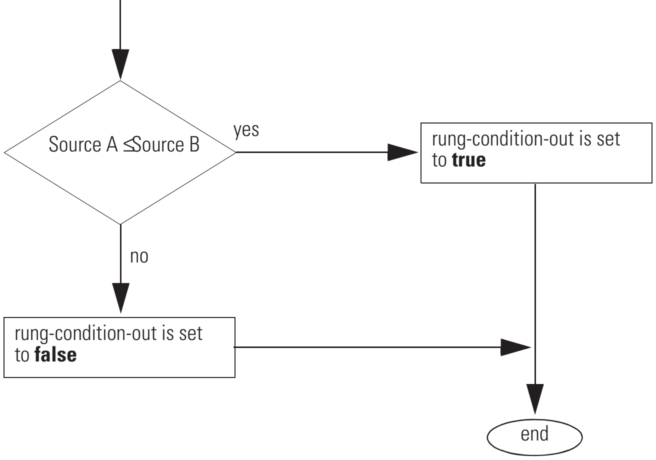
ASCII Characters	Hex Codes
1ab	$31$61$62
1b	$31$62
A	$41
AB	$41$42
B	$42
a	$61
ab	$61$62

Arithmetic Status Flags: Not affected

Fault Conditions: None

Execution:

Relay Ladder

Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	
Postscan	The rung-condition-out is set to false.

Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

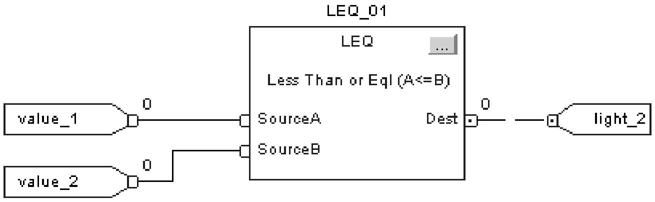
Example: If *value_1* is less than or equal to *value_2*, set *light_2*. If *value_1* is greater than *value_2*, clear *light_2*.

Relay Ladder

Structured Text

```
light_2 := (value_1 <= value_2);
```

Function Block

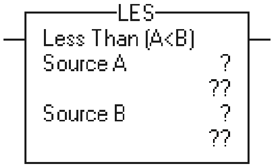


Less Than (LES)

The LES instruction tests whether Source A is less than Source B.

Operands:

Relay Ladder



Operand	Type	Format	Description
Source A	SINT INT **DINT** **REAL** **string**	Immediate Tag	Value to test against Source B
Source B	SINT INT **DINT** **REAL** **string**	Immediate Tag	Value to test against Source A

- If you enter a SINT or INT tag, the value converts to a DINT value by sign-extension.
- String data types are:
  - default STRING data type.
  - any new string data type that you create.
- To test the characters of a string, enter a string tag for both Source A and Source B.

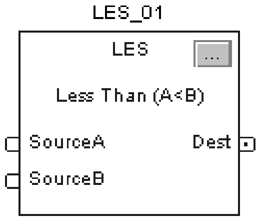
Structured Text

```
IF sourceA < sourceB THEN
    <statements>;
```

Use the less than sign ‘<‘ as an operator within an expression. This expression evaluates whether *sourceA* is less than *sourceB*.

See Structured Text Programming for information on the syntax of expressions within structured text.

Function Block



Operand	Type	Format	Description
LES tag	FBD_COMPARE	Structure	LES structure

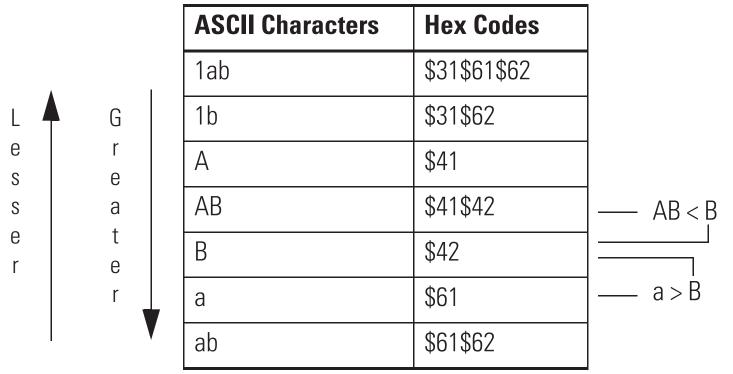
FBD_COMPARE Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
SourceA	REAL	Value to test against SourceB. Valid = any float
SourceB	REAL	Value to test against SourceA. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	BOOL	Result of the instruction. This is equivalent to rung-condition-out for the relay ladder LES instruction.

Description: The LES instruction tests whether Source A is less than Source B.

When you compare strings:

- the hexadecimal values of the characters determine if one string is less than or greater than another string. For the hex code of a character, see the back cover of this manual.
- when the two strings are sorted as in a telephone directory, the order of the strings determines which one is greater.



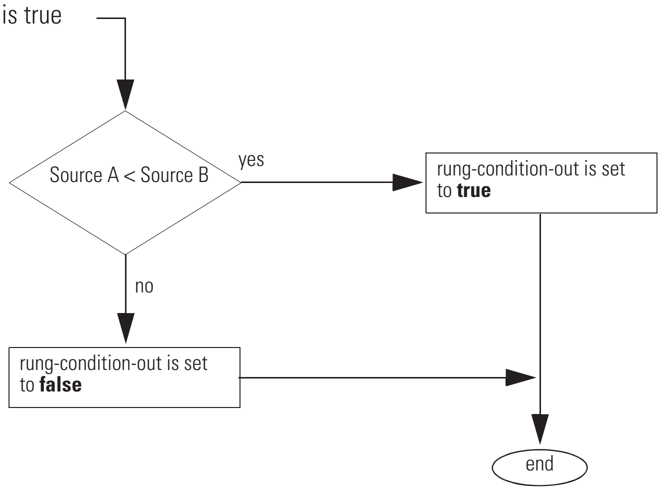
ASCII Characters	Hex Codes
1ab	$31$61$62
1b	$31$62
A	$41
AB	$41$42
B	$42
a	$61
ab	$61$62

Arithmetic Status Flags: Not affected

Fault Conditions: None

Execution:

Relay Ladder

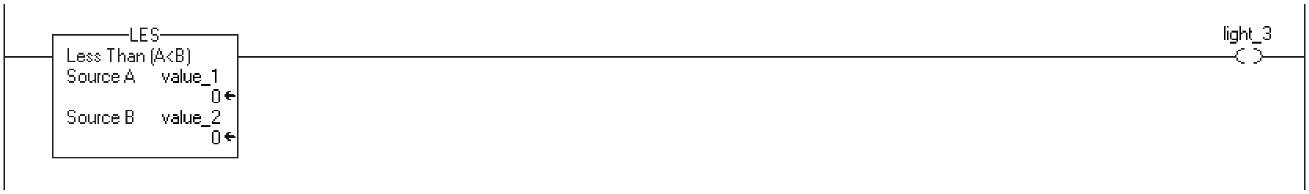
Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	
Postscan	The rung-condition-out is set to false.

Function Block

Condition:	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is false	EnableOut is cleared.
EnableIn is true	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: If *value_1* is less than *value_2*, set *light_3*. If *value_1* is greater than or equal to *value_2*, clear *light_3*.

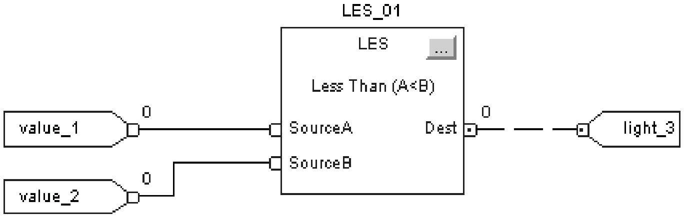
Relay Ladder



Structured Text

```
light_3 := (value_1 < value_2);
```

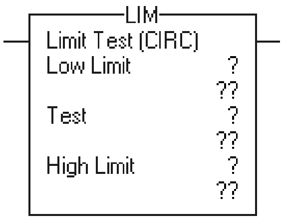
Function Block



Limit (LIM)

The LIM instruction tests whether the Test value is within the range of the Low Limit to the High Limit.

Operands:



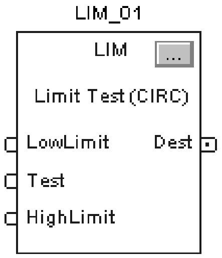
Relay Ladder

Operand	Type	Format	Description
Low limit	SINT INT **DINT** **REAL**	Immediate Tag	Value of lower limit
	A SINT or INT tag converts to a DINT value by sign-extension.		
Test	SINT INT **DINT** **REAL**	Immediate Tag	Value to test
	A SINT or INT tag converts to a DINT value by sign-extension.		
High limit	SINT INT **DINT** **REAL**	Immediate Tag	Value of upper limit
	A SINT or INT tag converts to a DINT value by sign-extension.		

Structured Text

Structured text does not have a LIM instruction, but you can achieve the same results by using structured text.

```
IF (LowLimit <= HighLimit AND
    (Test >= LowLimit AND Test <= HighLimit)) OR
    (LowLimit >= HighLimit AND
    (Test <= LowLimit OR Test >= HighLimit)) THEN

         <statement>;

END_IF;
```



Function Block

Operand	Type	Format	Description
LIM tag	FBD_LIMIT	Structure	LIM structure

FBD_LIMIT Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	If cleared, the instruction does not execute and outputs are not updated. If set, the instruction executes as described under Execution. Default is set.
LowLimit	REAL	Value of lower limit. Valid = any float
Test	REAL	Value to test against limits. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	BOOL	Result of the instruction. This is equivalent to rung-condition-out for the relay ladder LIM instruction.
HighLimit	REAL	Value of upper limit. Valid = any float

Description: The LIM instruction tests whether the Test value is within the range of the Low Limit to the High Limit.

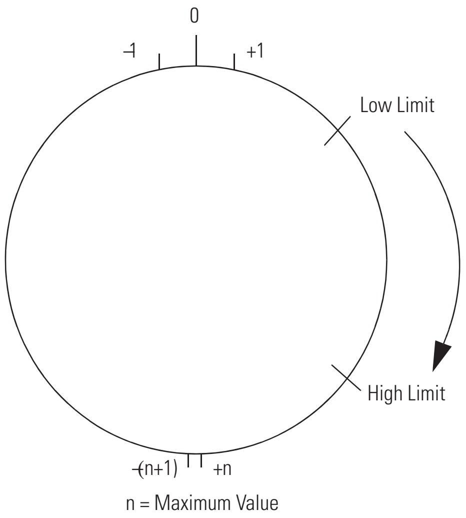
If Low Limit	And Test Value Is	The Rung-condition-out Is
≤High Limit	Equal to or between limits	True
	Not equal to or outside limits	False
≥ High Limit	Equal to or outside limits	True
	Not equal to or inside limits	False

Signed integers ‘roll over’ from the maximum positive number to the maximum negative number when the most significant bit is set. For example, in 16-bit integers (INT type), the maximum positive integer is 32,767, which is represented in hexadecimal as 16#7FFF (bits 0 …14 are all set). If you increment that number by one, the result is 16#8000 (bit 15 is set). For signed integers, hexadecimal 16#8000 is equal to -32,768 decimal. Incrementing from this point on until all 16 bits are set ends up at 16#FFFF, which is equal to -1 decimal.

This can be shown as a circular number line (see the following diagrams). The LIM instruction starts at the Low Limit and increments clockwise until it reaches the High Limit. Any Test value in the clockwise range from the Low Limit to the High Limit sets the rung-condition-out to true. Any Test value in the clockwise range from the High Limit to the Low Limit sets the rung-condition-out to false.

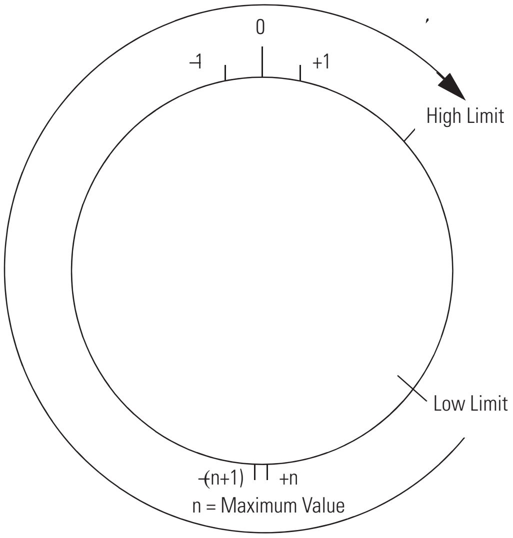
Low Limit ≤High Limit

The instruction is true if the test value is equal to or between the low and high limit



Low Limit ≥ High Limit

The instruction is true if the test value is equal to or outside the low and high limit



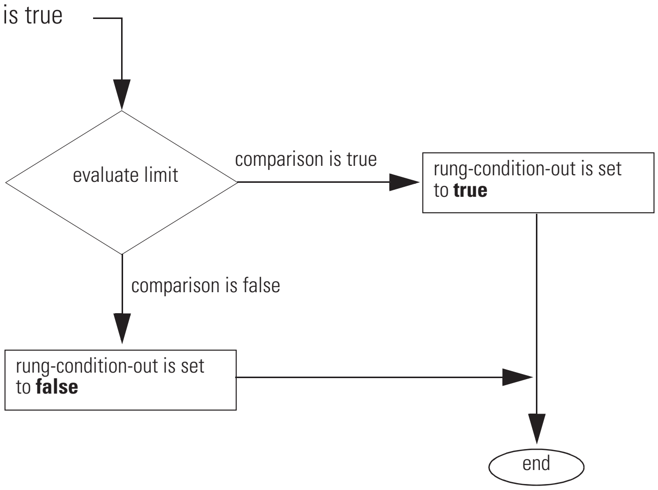
Arithmetic Status Flags: Not affected

Fault Conditions: None

Execution:

Relay Ladder

Condition	**Action**
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	



Condition	**Action**
Postscan	The rung-condition-out is set to false.

Function Block

Condition	**Action**
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared, the instruction does nothing, and the outputs are not updated.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example 1: **Low Limit ≤High Limit:**

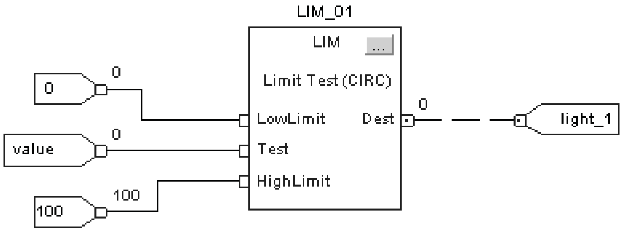
When 0 ≤*value* ≥ 100, set *light_1*. If *value* < 0 or *value* >100, clear *light_1*.

Relay Ladder

Structured Text

```
IF (value <= 100 AND(value >= 0 AND value <= 100)) OR
    (value >= 100 AND value <= 0 OR value >= 100)) THEN

         light_1 := 1;

ELSE

         light_1 := 0;

END_IF;
```

Function Block



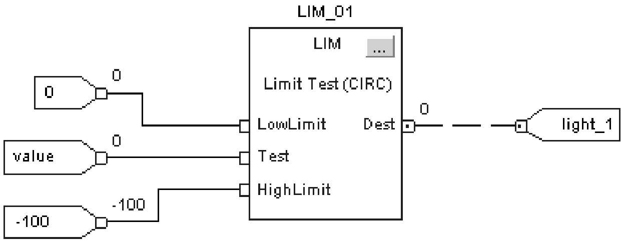
Example 2: **Low Limit ≥ High Limit:**
When *value* ≥ 0 or *value* ≤−100, set *light_1*. If *value* < 0 or *value* >−100, clear *light_1*.

Relay Ladder

Structured Text

```
IF (0 <= -100 AND value >= 0 AND value <= -100)) OR
    (0 >= -100 AND(value <= 0 OR value >= -100)) THEN

         light_1 := 1;

ELSE

         light_1 := 0;

END_IF;
```

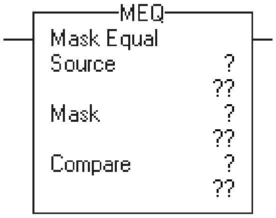
Function Block



Mask Equal To (MEQ)

The MEQ instruction passes the Source and Compare values through a Mask and compares the results.

Operands:



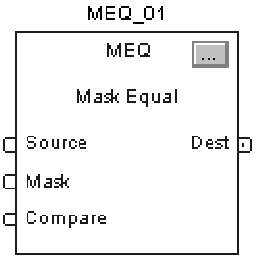
Relay Ladder

Operand	Type	Format	Description
Source	SINT INT **DINT**	Immediate Tag	Value to test against Compare
	A SINT or INT tag converts to a DINT value by zero-fill.		
Mask	SINT INT **DINT**	Immediate Tag	Defines which bits to block or pass
	A SINT or INT tag converts to a DINT value by zero-fill.		
Compare	SINT INT **DINT**	Immediate Tag	Value to test against Source
	A SINT or INT tag converts to a DINT value by zero-fill.		

Structured Text

Structured text does not have an MEQ instruction, but you can achieve the same results by using structured text.

```
IF (Source AND Mask) = (Compare AND Mask) THEN

    <statement>;

END_IF;
```



Function Block

Operand	Type	Format	Description
MEQ tag	FBD_MASK_EQUAL	Structure	MEQ structure

FBD_MASK_EQUAL Structure

Input Parameter	**Data Type**	**Description**
EnableIn	BOOL	If cleared, the instruction does not execute and outputs are not updated. If set, the instruction executes as described under Execution. Default is set.
Source	DINT	Value to test against Compare. Valid = any integer
Mask	DINT	Defines which bits to block (mask). Valid = any integer
Compare	DINT	Compare value. Valid = any integer
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	BOOL	Result of the instruction. This is equivalent to rung-condition-out for the relay ladder MEQ instruction.

Description:

A ‘1’ in the mask means the data bit is passed. A ‘0’ in the mask means the data bit is blocked. Typically, the Source, Mask, and Compare values are all the same data type.

If you mix integer data types, the instruction fills the upper bits of the smaller integer data types with 0s so that they are the same size as the largest data type.

Entering an Immediate Mask Value

When you enter a mask, the programming software defaults to decimal values. If you want to enter a mask by using another format, precede the value with the correct prefix.

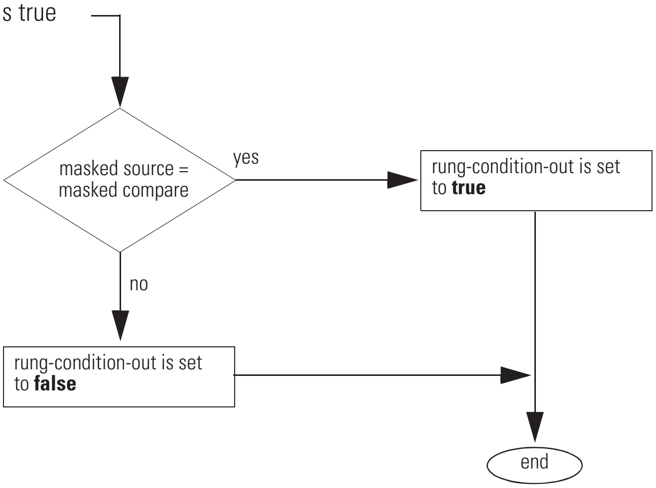
Prefix	**Description**
16#	Hexadecimal For example; 16#0F0F
8#	Octal For example; 8#16
2#	Binary For example; 2#00110011

Arithmetic Status Flags: Not affected

Fault Conditions: None

Execution:

Relay Ladder

Condition	**Action**
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	
Postscan	The rung-condition-out is set to false.

Function Block

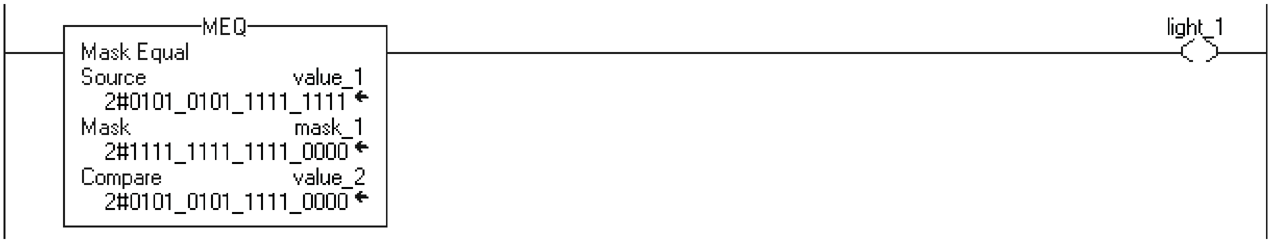
Condition	**Action**
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared, the instruction does nothing, and the outputs are not updated.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example 1: If the masked *value_1* is equal to the masked *value_2*, set *light_1*. If the masked *value_1* is not equal to the masked *value_2,* clear *light_1*. This example shows that the masked values are equal. A 0 in the mask restrains the instruction from comparing that bit (shown by x in the example).

value_1	0	1	0	1	0	1	0	1	1	1	1	1	1	1	1	1
mask_1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0
masked value_1	0	1	0	1	0	1	0	1	1	1	1	1	x	x	x	x

value_2	0	1	0	1	0	1	0	1	1	1	1	1	0	0	0	0
mask_1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0
masked value_2	0	1	0	1	0	1	0	1	1	1	1	1	x	x	x	x

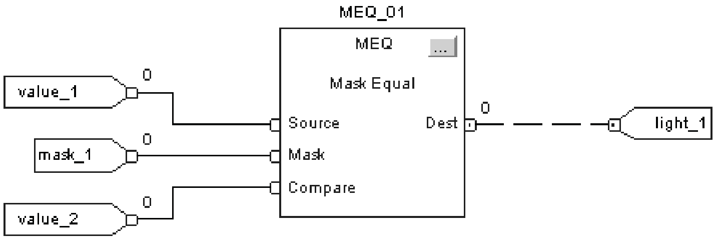
Relay Ladder



Structured Text

```
light_1 := ((value_1 AND mask_1)=(value_2 AND mask_2));
```

Function Block

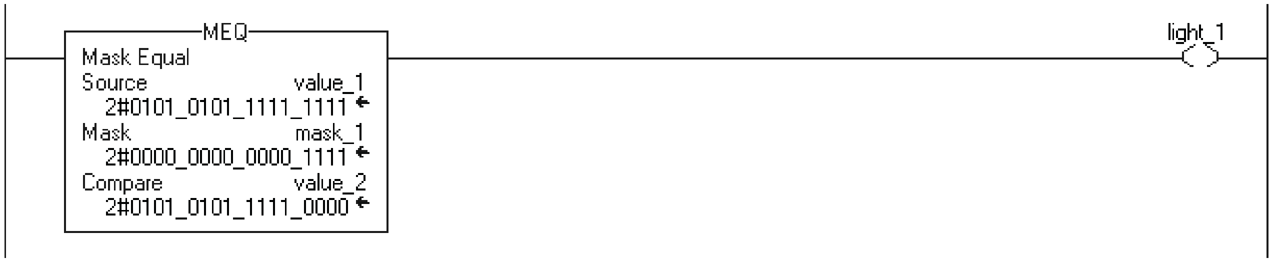


Example 2: If the masked *value_1* is equal to the masked *value_2*, set light_1. If the masked *value_1* is not equal to the masked *value_2*, clear *light_1*. This example shows that the masked values are not equal. A 0 in the mask restrains the instruction from comparing that bit (shown by x in the example).

value_1	0	1	0	1	0	1	0	1	1	1	1	1	1	1	1	1
mask_1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
masked value_1	x	x	x	x	x	x	x	x	x	x	x	x	1	1	1	1

value_2	0	1	0	1	0	1	0	1	1	1	1	1	0	0	0	0
mask_1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
masked value_2	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0

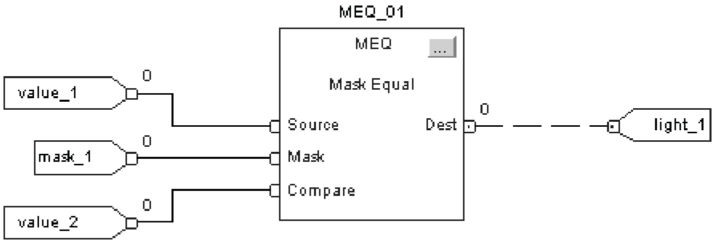
Relay Ladder



Structured Text

```
light_1 := ((value_1 AND mask_1)=(value_2 AND mask_2));
```

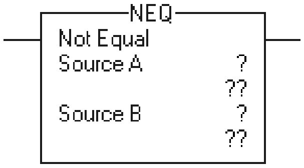
Function Block



Not Equal To (NEQ)

The NEQ instruction tests whether Source A is not equal to Source B.

Operands:



Relay Ladder

Operand	Type	Format	Description
Source A	SINT INT **DINT** **REAL** **string**	Immediate Tag	Value to test against Source B
Source B	SINT INT **DINT** **REAL** **string**	Immediate Tag	Value to test against Source A

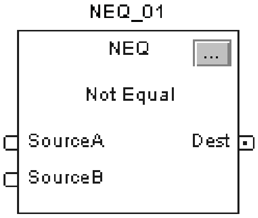
- If you enter a SINT or INT tag, the value converts to a DINT value by sign-extension.
- String data types are:
  - default STRING data type.
  - any new string data type that you create.
- To test the characters of a string, enter a string tag for both Source A and Source B.

Structured Text

```
IF sourceA <> sourceB THEN
    <statements>;
```

Use the less than and greater than signs '<>' together as an operator within an expression. This expression evaluates whether *sourceA* is not equal to *sourceB*.

See Structured Text Programming for information on the syntax of expressions within structured text.



Function Block

Operand	Type	Format	Description
NEQ tag	FBD_COMPARE	Structure	NEQ structure

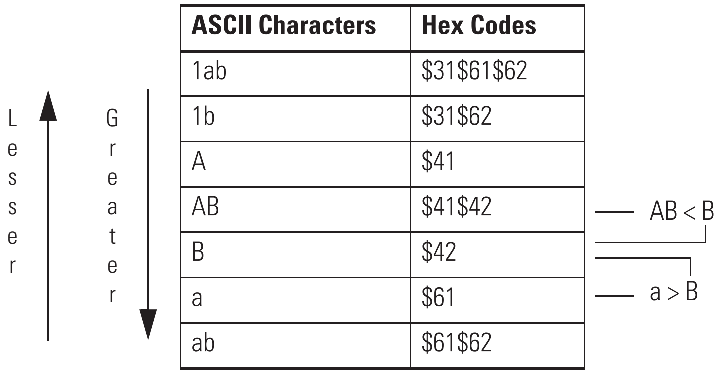
FBD_COMPARE Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
SourceA	REAL	Value to test against SourceB. Valid = any float
SourceB	REAL	Value to test against SourceA. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	BOOL	Result of the instruction. This is equivalent to rung-condition-out for the relay ladder NEQ instruction.

Description: The NEQ instruction tests whether Source A is not equal to Source B.

When you compare strings:

- strings are not equal if any of their characters do not match.
- ASCII characters are case sensitive. Upper case ‘A’ ($41) is **not** equal to lower case ‘a’ ($61).



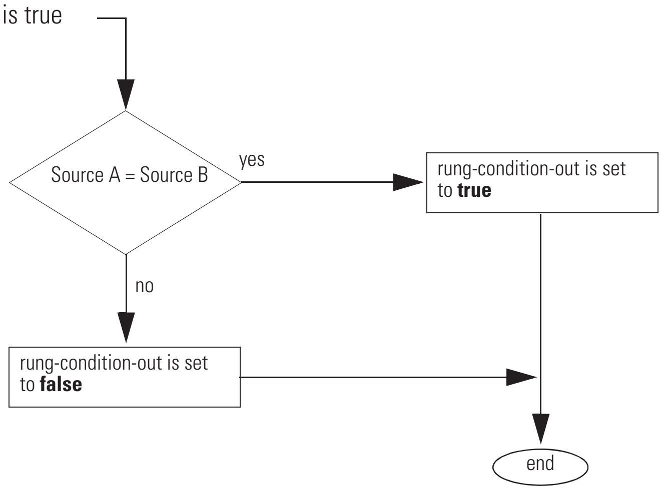
ASCII Characters	Hex Codes
1ab	$31$61$62
1b	$31$62
A	$41
AB	$41$42
B	$42
a	$61
ab	$61$62

Arithmetic Status Flags: Not affected

Fault Conditions: None

Execution:

Relay Ladder

Condition	**Action**
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	
Postscan	The rung-condition-out is set to false.

Function Block

Condition	**Action**
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

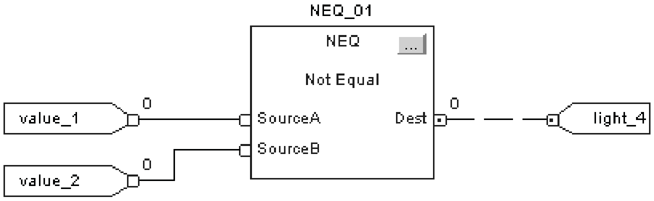
Example: If *value_1* is not equal to *value_2*, set *light_4*. If *value_1* is equal to *value_2*, clear *light_4*.

Relay Ladder

Structured Text

```
light_4 := (value_1 <> value_2);
```

Function Block



Notes:

Compute/Math Instructions

(CPT, ADD, SUB, MUL, DIV, MOD, SQR, SQRT, NEG, ABS)

Introduction

The compute/math instructions evaluate arithmetic operations by using an expression or a specific arithmetic instruction.

If you want to	Use this instruction	Available in these languages	Page
Evaluate an expression	CPT	Relay ladder Structured text(1)	256
Add two values	ADD	Relay ladder Structured text(2) Function block	260
Subtract two values	SUB	Relay ladder Structured text(2) Function block	263
Multiply two values	MUL	Relay ladder Structured text(2) Function block	266
Divide two values	DIV	Relay ladder Structured text(2) Function block	269
Determine the remainder after one value is divided by another	MOD	Relay ladder Structured text(2) Function block	274
Calculate the square root of a value	SQR SQRT(3)	Relay ladder Structured text Function block	278
Take the opposite sign of a value	NEG	Relay ladder Structured text(2) Function block	282
Take the absolute value of a value	ABS	Relay ladder Structured text Function block	285

(1) There is no equivalent structured text instruction. Use other structured text programming to achieve the same result. See the description for the instruction.

(2) There is no equivalent structured text instruction. Use the operator in an expression.

(3) Structured text only.

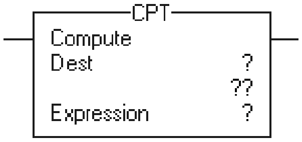
You can mix data types, but loss of accuracy and rounding error might occur and the instruction takes more time to execute. Check the S:V bit to see whether the result was truncated.

For relay ladder instructions, **bold** data types indicate optimal data types. An instruction executes faster and requires less memory if all the operands of the instruction use the same optimal data type, typically DINT or REAL.

Compute (CPT)

The CPT instruction performs the arithmetic operations you define in the expression.

Operands:



Relay Ladder

Operand	Type	Format	Description
Destination	SINT INT **DINT** **REAL**	Tag	Tag to store the result
Expression	SINT INT **DINT** **REAL**	Immediate Tag	An expression consisting of tags and/or immediate values separated by operators
	A SINT or INT tag converts to a DINT value by sign-extension.		

Structured Text

Structured text does not have a CPT instruction, but you can achieve the same results by using an assignment and expression.

```
destination := numeric_expression;
```

See Structured Text Programming for information on the syntax of assignments and expressions within structured text.

Description: The CPT instruction performs the arithmetic operations you define in the expression. When enabled, the CPT instruction evaluates the expression and places the result in the Destination.

The execution of a CPT instruction is slightly slower and uses more memory than the execution of the other compute/math instructions. The advantage of the CPT instruction is that it allows you to enter complex expressions in one instruction.

TIP There is no limit to the length of an expression.

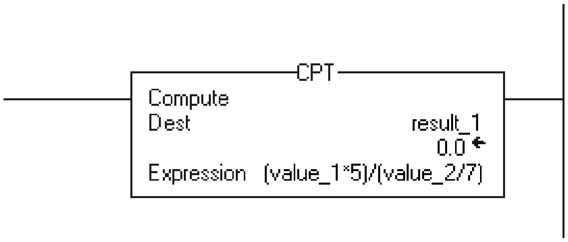
Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions: None

Execution:

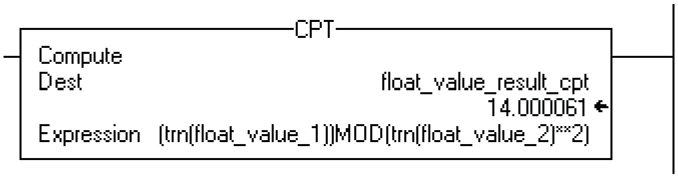
Condition	Relay Ladder Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The instruction evaluates the Expression and places the result in the Destination. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Example 1: When enabled, the CPT instruction evaluates *value_1* multiplied by 5 and divides that result by the result of *value_2* divided by 7 and places the final result in *result_1*.



Dest result_1
0.0
Expression (value_1*5)/(value_2/7)

Example 2: When enabled, the CPT instruction truncates *float_value_1* and *float_value_2*, raises the truncated *float_value_2* to the power of two and divides the truncated *float_value_1* by that result, and stores the remainder after the division in *float_value_result_cpt*.



Dest float_value_result_cpt
14.000061
Expression (trn(float_value_1))MOD(trn(float_value_2)**2)

Valid Operators

Operator	Description	Optimal
+	add	DINT, REAL
-	subtract/negate	DINT, REAL
*	multiply	DINT, REAL
/	divide	DINT, REAL
**	exponent (x to y)	DINT, REAL
ABS	absolute value	DINT, REAL
ACS	arc cosine	REAL
AND	bitwise AND	DINT
ASN	arc sine	REAL
ATN	arc tangent	REAL
COS	cosine	REAL
DEG	radians to degrees	DINT, REAL
FRD	BCD to integer	DINT
LN	natural log	REAL

Operator	Description	Optimal
LOG	log base 10	REAL
MOD	modulo-divide	DINT, REAL
NOT	bitwise complement	DINT
OR	bitwise OR	DINT
RAD	degrees to radians	DINT, REAL
SIN	sine	REAL
SQR	square root	DINT, REAL
TAN	tangent	REAL
TOD	integer to BCD	DINT
TRN	truncate	DINT, REAL
XOR	bitwise exclusive OR	DINT

Format Expressions

For each operator that you use in an expression, you have to provide one or two operands (tags or immediate values). Use the following table to format operators and operands within an expression.

For operators that operate on	Use this format	Examples
One operand	Operator(operand)	ABS(*tag_a*)
Two operands	Operand_a operator operand_b	· *tag_b* + 5 · *tag_c* AND *tag_d* · (*tag_e* ** 2) MOD (*tag_f* / *tag_g*)

Determine the Order of Operation

The operations you write into the expression are performed by the instruction in a prescribed order, not necessarily the order you write them. You can override the order of operation by grouping terms within parentheses, forcing the instruction to perform an operation within the parentheses ahead of other operations.

Operations of equal order are performed from left to right.

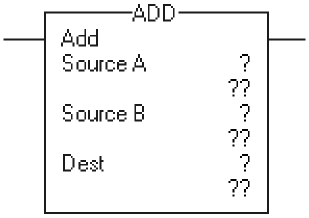
Order	Operation
1.	( )
2.	ABS, ACS, ASN, ATN, COS, DEG, FRD, LN, LOG, RAD, SIN, SQR, TAN, TOD, TRN
3.	**
4.	–(negate), NOT
5.	*, /, MOD
6.	–(subtract), +
7.	AND
8.	XOR
9.	OR

Add (ADD)

The ADD instruction adds Source A to Source B and places the result in the Destination.

Operands:

Relay Ladder



Operand	Type	Format	Description
Source A	SINT INT **DINT** **REAL**	Immediate Tag	Value to add to Source B
	A SINT or INT tag converts to a DINT value by sign-extension.		
Source B	SINT INT **DINT** **REAL**	Immediate Tag	Value to add to Source A
	A SINT or INT tag converts to a DINT value by sign-extension.		
Destination	SINT INT **DINT** **REAL**	Tag	Tag to store the result

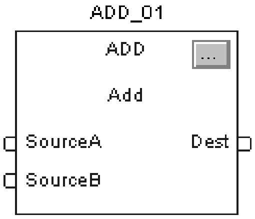
Structured Text

```
dest := sourceA + sourceB;
```

Use the plus sign "+" as an operator within an expression. This expression adds *sourceA* to *sourceB* and stores the result in *dest*.

See Structured Text Programming for information on the syntax of expressions within structured text.

Function Block



Operand	Type	Format	Description
ADD tag	FBD_MATH	Structure	ADD structure

FBD_MATH Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
SourceA	REAL	Value to add to SourceB. Valid = any float
SourceB	REAL	Value to add to SourceA. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	REAL	Result of the math instruction. Arithmetic status flags are set for this output.

Description:

The ADD instruction adds Source A to Source B and places the result in the Destination.

Arithmetic Status Flags:

Arithmetic status flags are affected.

Fault Conditions:

None

Execution:

Relay Ladder

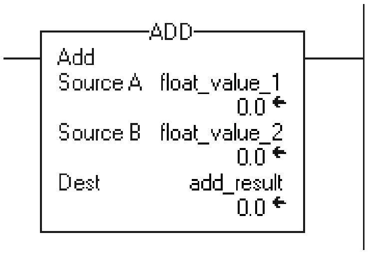
Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	Destination = Source A + Source B The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: Add *float_value_1* to *float_value_2* and place the result in *add_result*.

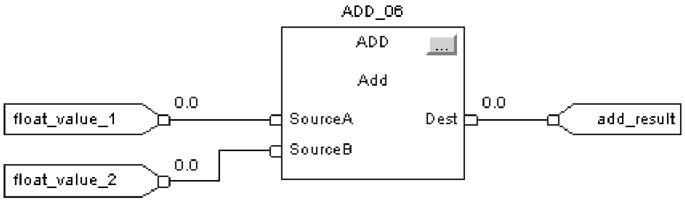
Relay Ladder



Structured Text

```
add_result := float_value_1 + float_value_2;
```

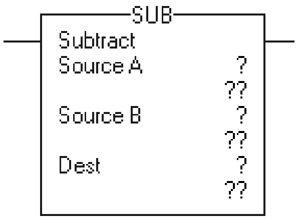
Function Block



Subtract (SUB)

The SUB instruction subtracts Source B from Source A and places the result in the Destination.

Operands:



Relay Ladder

Operand	Type	Format	Description
Source A	SINT INT **DINT** **REAL** A SINT or INT tag converts to a DINT value by sign-extension.	Immediate Tag	Value from which to subtract Source B
Source B	SINT INT **DINT** **REAL** A SINT or INT tag converts to a DINT value by sign-extension.	Immediate Tag	Value to subtract from Source A
Destination	SINT INT **DINT** **REAL**	Tag	Tag to store the result

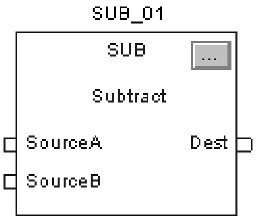
Structured Text

```
dest := sourceA - sourceB;
```

Use the minus sign “-” as an operator in an expression. This expression subtracts *sourceB* from *sourceA* and stores the result in *dest*.

See Structured Text Programming for information on the syntax of expressions within structured text.

Function Block



Operand	Type	Format	Description
SUB tag	FBD_MATH	structure	SUB structure

FBD_MATH Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
SourceA	REAL	Value from which to subtract SourceB. Valid = any float
SourceB	REAL	Value to subtract from SourceA. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	REAL	Result of the math instruction. Arithmetic status flags are set for this output.

Description: The SUB instruction subtracts Source B from Source A and places the result in the Destination.

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions: None

Execution:

Relay Ladder

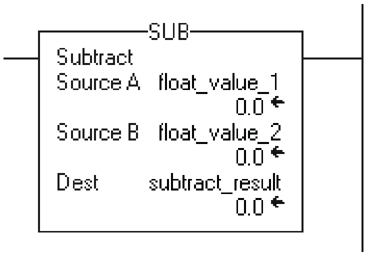
Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	Destination = Source B - Source A The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: Subtract *float_value_2* from *float_value_1* and place the result in *subtract_result*.

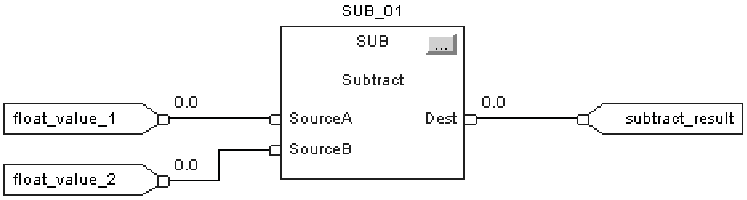
Relay Ladder



Structured Text

```
subtract_result := float_value_1 - float_value_2;
```

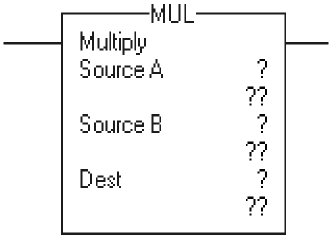
Function Block



Multiply (MUL)

The MUL instruction multiplies Source A with Source B and places the result in the Destination.

Operands:



Relay Ladder

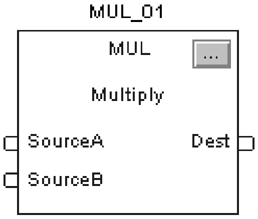
Operand	**Type**	**Format**	**Description**
Source A	SINT INT **DINT** **REAL**	Immediate Tag	Value of the multiplicand
	A SINT or INT tag converts to a DINT value by sign-extension.		
Source B	SINT INT **DINT** **REAL**	Immediate Tag	Value of the multiplier
	A SINT or INT tag converts to a DINT value by sign-extension.		
Destination	SINT INT **DINT** **REAL**	Tag	Tag to store the result

Structured Text

```
dest := sourceA * sourceB;
```

Use the multiply sign “*” as an operator in an expression. This expression multiplies *sourceA* by *sourceB* and stores the result in *dest*.

See Structured Text Programming for information on the syntax of expressions within structured text.



Function Block

Operand	**Type**	**Format**	**Description**
MUL tag	FBD_MATH	Structure	MUL structure

FBD_MATH Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
Source A	REAL	Value of the multiplicand. Valid = any float
Source B	REAL	Value of the multiplier. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	REAL	Result of the math instruction. Arithmetic status flags are set for this output.

Description: The MUL instruction multiplies Source A with Source B and places the result in the Destination.

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions: None

Execution:

Relay Ladder

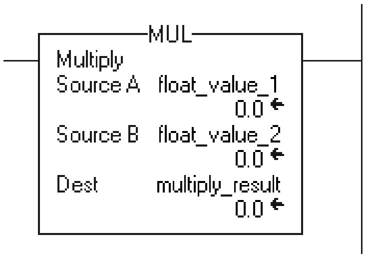
Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	Destination = Source B x Source A The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: Multiply *float_value_1* by *float_value_2* and place the result in *multiply_result*.

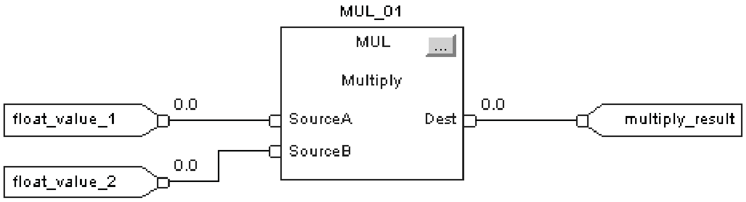
Relay Ladder



Structured Text

```
multiply_result := float_value_1 * float_value_2;
```

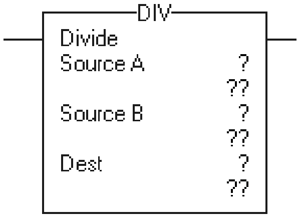
Function Block



Divide (DIV)

The DIV instruction divides Source A by Source B and places the result in the Destination.

Operands:



Relay Ladder

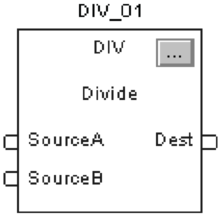
Operand	Type	Format	Description
Source A	SINT INT **DINT** **REAL**	Immediate Tag	Value of the dividend
	A SINT or INT tag converts to a DINT value by sign-extension.		
Source B	SINT INT **DINT** **REAL**	Immediate Tag	Value of the divisor
	A SINT or INT tag converts to a DINT value by sign-extension.		
Destination	SINT INT **DINT** **REAL**	Tag	Tag to store the result

Structured Text

```
dest := sourceA / sourceB;
```

Use the divide sign “/ ” as an operator in an expression. This expression divides *sourceA* by *sourceB* and stores the result in *dest*.

See Structured Text Programming for information on the syntax of expressions within structured text.



Function Block

Operand	Type	Format	Description
DIV tag	FBD_MATH	structure	DIV structure

FBD_MATH Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
Source A	REAL	Value of the dividend. Valid = any float
Source B	REAL	Value of the divisor. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	REAL	Result of the math instruction. Arithmetic status flags are set for this output.

Description: If the Destination is **not** a REAL, the instruction handles the fractional portion of the result as follows:

If source A	Then the fractional portion of the result	Example		
And Source B are *not* REALs	Truncates	Source A	DINT	5
		Source B	DINT	3
		Destination	DINT	1
Or Source B is a REAL	Rounds	Source A	REAL	5.0
		Source B	DINT	3
		Destination	DINT	2

If Source B (the divisor) is zero:

- a minor fault occurs:
  - Type 4: program fault
  - Code 4: arithmetic overflow
- the destination is set as follows:

If source B is zero and	And the destination is a	And the result is	Then the destination is set to
All operands are integers (SINT, INT, or DINT)	→	→	Source A
At least one operand is a REAL	SINT, INT, or DINT	Positive	-1
		Negative	0
	REAL	Positive	1.$ (positive infinity)
		Negative	-1.$ (negative infinity)

To detect a possible divide-by-zero, examine the minor fault bit (S:MINOR). See Logix5000 Controllers Common Procedures Programming Manual, publication 1756-PM001.

Arithmetic Status Flags:

Arithmetic status flags are affected.

Fault Conditions:

A minor fault occurs if	Fault type	Fault code
The divisor is zero	4	4

Execution:

Relay Ladder

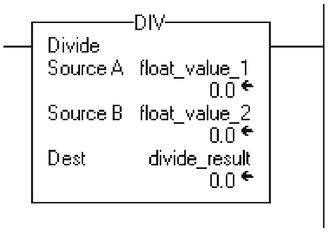
Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	Destination = Source A / Source B The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example 1: Divide *float_value_1* by *float_value_2* and place the result in *divide_result*.

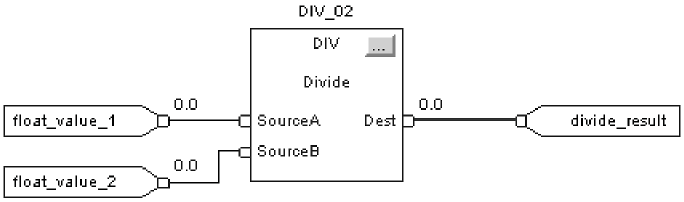
Relay Ladder



Structured Text

```
divide_result := float_value_1 / float_value_2;
```

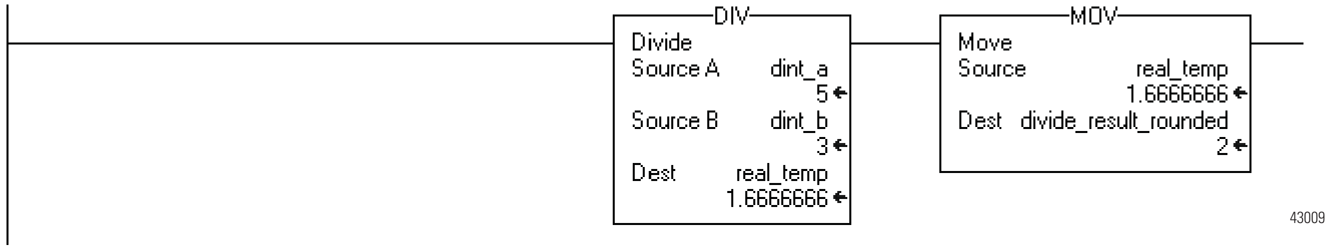
Function Block



Example 2: The DIV and MOV instructions work together to divide two integers, round the result, and place the result in an integer tag:

- The DIV instruction divides *dint_a* by *dint_b*.
- To round the result, the Destination is a REAL tag. (If the destination was an integer tag (SINT, INT, or DINT), the instruction would truncate the result.)
- The MOV instruction moves the rounded result (*real_temp*) from the DIV to *divide_result_rounded*.
- Since *divide_result_rounded* is a DINT tag the value from *real_temp* is rounded and placed in the DINT destination.

Relay Ladder



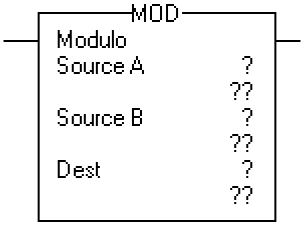
DIV
Divide
Source A dint_a
5
Source B dint_b
3
Dest real_temp
1.6666666

MOV
Move
Source real_temp
1.6666666
Dest divide_result_rounded
2

Modulo (MOD)

The MOD instruction divides Source A by Source B and places the remainder in the Destination.

Operands:



Relay Ladder

Operand	Type	Format	Description
Source A	SINT INT **DINT** **REAL**	Immediate Tag	Value of the dividend
	A SINT or INT tag converts to a DINT value by sign-extension.		
Source B	SINT INT **DINT** **REAL**	Immediate Tag	Value of the divisor
	A SINT or INT tag converts to a DINT value by sign-extension.		
Destination	SINT INT **DINT** **REAL**	Tag	Tag to store the result

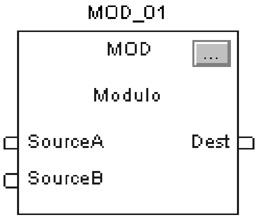
Structured Text

```
dest := sourceA MOD sourceB;
```

Use MOD as an operator in an expression. This expression divides *sourceA* by *sourceB* and stores the remainder in *dest*.

See Structured Text Programming for information on the syntax of expressions within structured text.

Function Block



Operand	Type	Format	Description
MOD tag	FBD_MATH	Structure	MOD structure

FBD_MATH Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
Source A	REAL	Value of the dividend. Valid = any float
Source B	REAL	Value of the divisor. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	REAL	Result of the math instruction. Arithmetic status flags are set for this output.

Description: If Source B (the divisor) is zero:

- a minor fault occurs:
  - Type 4: program fault
  - Code 4: arithmetic overflow
- the destination is set as follows:

If source B is zero and	And the destination is a	And the result is	Then the destination is set to
All operands are integers (SINT, INT, or DINT)	→	→	Source A
At least one operand is a REAL	SINT, INT, or DINT	Positive	-1
		Negative	0
	REAL	Positive	1.$ (positive infinity)
		Negative	-1.$ (negative infinity)

To detect a possible divide-by-zero, examine the minor fault bit (S:MINOR). See Logix5000 Controllers Common Procedures Programming Manual, publication 1756-PM001.

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions:

A minor fault occurs if	Fault type	Fault code
The divisor is zero	4	4

Execution:

Relay Ladder

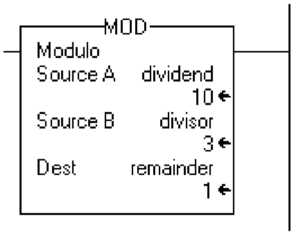
Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	Destination = Source A – (TRN (Source A / Source B) * Source B) The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
Postscan	No action taken.

Example: Divide *dividend* by *divisor* and place the remainder in *remainder*. In this example, three goes into 10 three times, with a remainder of one.

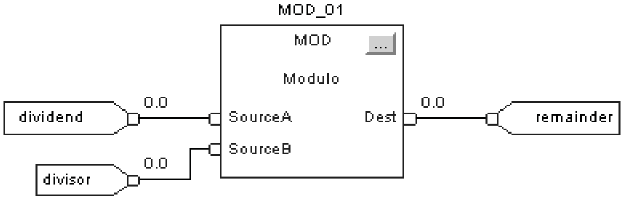
Relay Ladder



Structured Text

```
remainder := dividend MOD divisor;
```

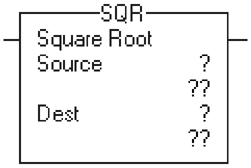
Function Block



Square Root (SQR)

The SQR instruction computes the square root of the Source and places the result in the Destination.

Operands:



Relay Ladder

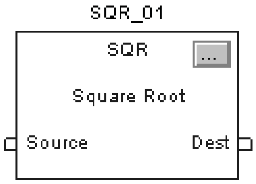
Operand	Type	Format	Description
Source	SINT INT **DINT** **REAL**	Immediate Tag	Find the square root of this value
	A SINT or INT tag converts to a DINT value by sign-extension.		
Destination	SINT INT **DINT** **REAL**	Tag	Tag to store the result

Structured Text

```
dest := SQRT(source);
```

Use SQRT as a function. This expression computes the square root of *source* and stores the result in *dest*.

See Structured Text Programming for information on the syntax of expressions within structured text.



Function Block

Operand	Type	Format	Description
SQR tag	FBD_MATH_ADVANCED	Structure	SQR structure

FBD_MATH_ADVANCED Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
Source	REAL	Find the square root of this value. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	REAL	Result of the math instruction. Arithmetic status flags are set for this output.

Description:

If the Destination is **not** a REAL, the instruction handles the fractional portion of the result as follows:

If the source is	Then the fractional portion of the result	Example		
Not a REAL	Truncates	Source	DINT	3
		Destination	DINT	1
A REAL	Rounds	Source	REAL	3.0
		Destination	DINT	2

If the Source is negative, the instruction takes the absolute value of the Source before calculating the square root.

Arithmetic Status Flags:

Arithmetic status flags are affected.

Fault Conditions:

None

Execution:

Relay Ladder

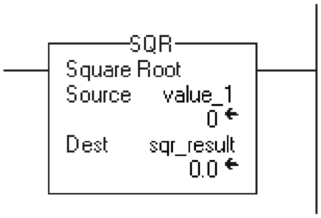
Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	$Destination = \sqrt{Source}$ The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: Calculate the square root of *value_1* and place the result in *sqr_result*.

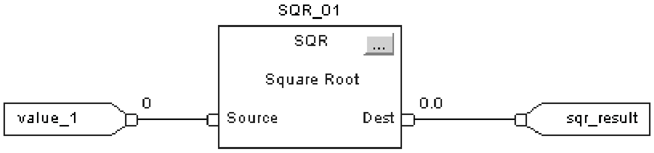
Relay Ladder



Structured Text

```
sqr_result := SQRT(value_1);
```

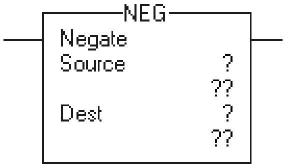
Function Block



Negate (NEG)

The NEG instruction changes the sign of the Source and places the result in the Destination.

Operands:



Relay Ladder

Operand	Type	Format	Description
Source	SINT INT **DINT** **REAL**	Immediate Tag	Value to negate
	A SINT or INT tag converts to a DINT value by sign-extension.		
Destination	SINT INT **DINT** **REAL**	Tag	Tag to store the result

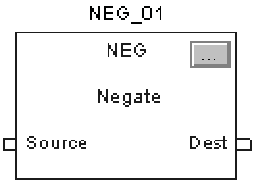
Structured Text

```
dest := -source;
```

Use the minus sign ‘–’ as an operator in an expression. This expression changes the sign of *source* and stores the result in *dest*.

See Structured Text Programming for information on the syntax of expressions within structured text.

Function Block



Operand	Type	Format	Description
NEG tag	FBD_MATH_ADVANCED	Structure	NEG structure

FBD_MATH Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
Source	REAL	Value to negate. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	REAL	Result of the math instruction. Arithmetic status flags are set for this output.

Description: If you negate a negative value, the result is positive. If you negate a positive value, the result is negative.

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions: None

Execution:

Relay Ladder

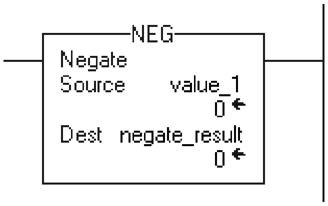
Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	Destination = 0 –Source The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: Change the sign of *value_1* and place the result in *negate_result*.

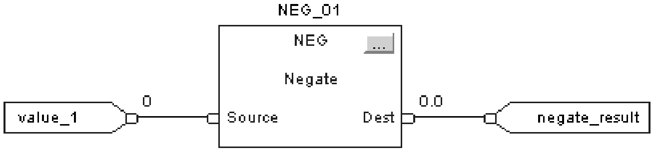
Relay Ladder



Structured Text

```
negate_result := -value_1;
```

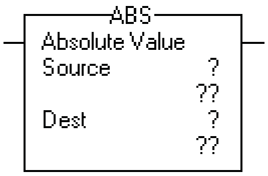
Function Block



Absolute Value (ABS)

The ABS instruction takes the absolute value of the Source and places the result in the Destination.

Operands:



Relay Ladder

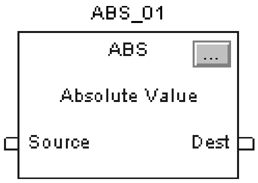
Operand	Type	Format	Description
Source	SINT INT **DINT** **REAL**	Immediate Tag	Value of which to take the absolute value
	A SINT or INT tag converts to a DINT value by sign-extension.		
Destination	SINT INT **DINT** **REAL**	Tag	Tag to store the result

Structured Text

```
dest := ABS(source);
```

Use ABS as a function. This expression computes the absolute value of *source* and stores the result in *dest*.

See Structured Text Programming for information on the syntax of expressions within structured text.



Function Block

Operand	Type	Format	Description
ABS tag	FBD_MATH_ADVANCED	structure	ABS structure

FBD_MATH_ADVANCED Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
Source	REAL	Value of which to take the absolute value. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	REAL	Result of the math instruction. Arithmetic status flags are set for this output.

Description: The ABS instruction takes the absolute value of the Source and places the result in the Destination.

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions: None

Execution:

Relay Ladder

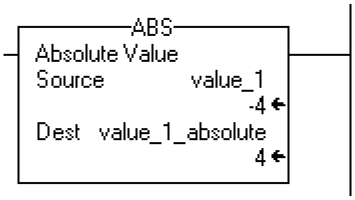
Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	Destination = | Source | The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: Place the absolute value of *value_1* into *value_1_absolute*. In this example, the absolute value of negative four is positive four.

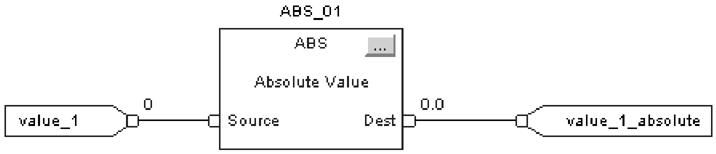
Relay Ladder



Structured Text

```
value_1_absolute := ABS(value_1);
```

Function Block



Notes:

Move/Logical Instructions

(MOV, MVM, BTD, MVMT, BTDT, CLR, SWPB, AND, OR, XOR, NOT, BAND, BOR, BXOR, BNOT)

Introduction

You can mix data types, but loss of accuracy and rounding error might occur and the instruction takes more time to execute. Check the S:V bit to see whether the result was truncated.

For relay ladder instructions, **bold** data types indicate optimal data types. An instruction executes faster and requires less memory if all the operands of the instruction use the same optimal data type, typically DINT or REAL.

The move instructions modify and move bits.

If you want to	Use this instruction	Available in these languages	Page
Copy a value	MOV	Relay ladder Structured text[1]	291
Copy a specific part of an integer	MVM	Relay ladder	293
Copy a specific part of an integer in function block	MVMT	Structured text Function block	296
Move bits within an integer or between integers	BTD	Relay ladder	299
Move bits within an integer or between integers in function block	BTDT	Structured text Function block	302
Clear a value	CLR	Structured text[1] Relay ladder	305
Rearrange the bytes of a INT, DINT, or REAL tag	SWPB	Relay ladder Structured text	307

[1] There is no equivalent structured text instruction. Use other structured text programming to achieve the same result. See the description for the instruction.

The logical instructions perform operations on bits.

If you want to	Use this instruction	Available in these languages	Page
Bitwise AND Operation	Bitwise AND &[1]	Relay ladder Structured text[2] Function block	311
Bitwise OR operation	Bitwise OR	Relay ladder Structured text[2] Function block	314
Bitwise, exclusive OR operation	Bitwise XOR	Relay ladder Structured text[2] Function block	318
Bitwise NOT operation	Bitwise NOT	Relay ladder Structured text[2] Function block	322
Logically AND as many as eight boolean inputs.	Boolean AND (BAND)	Structured text[2] Function block	325
Logically OR as many as eight boolean inputs.	Boolean OR (BOR)	Structured text[2] Function block	328
Perform an exclusive OR on two boolean inputs.	Boolean Exclusive OR (BXOR)	Structured text[2] Function block	331
Complement a boolean input.	Boolean NOT (BNOT)	Structured text[2] Function block	334

(1) Structured text only.

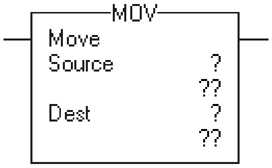
(2) In structured text, the AND, OR, XOR, and NOT operations can be bitwise or logical.

Move (MOV)

The MOV instruction copies the Source to the Destination. The Source remains unchanged.

Operands:

Relay Ladder



```
MOV
Move
Source    ?
         ??
Dest      ?
         ??
```

Operand	Type	Format	Description
Source	SINT INT **DINT** **REAL**	Immediate Tag	Value to move (copy)
	A SINT or INT tag converts to a DINT value by sign-extension.		
Destination	SINT INT **DINT** **REAL**	Tag	Tag to store the result

Structured Text

```
dest := source;
```

Use an assignment ":=" with an expression. This assignment moves the value in *source* to *dest*.

See Structured Text Programming for information on the syntax of expressions and assignments within structured text.

Description: The MOV instruction copies the Source to the Destination. The Source remains unchanged.

Arithmetic Status Flags: Arithmetic status flags are affected.

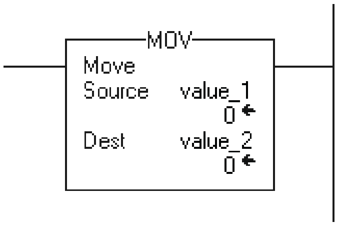
Fault Conditions: None

Execution:

Condition	Relay Ladder Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The instruction copies the Source into the Destination. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Example: Move the data in *value_1* to *value_2*.

Relay Ladder



Structured Text

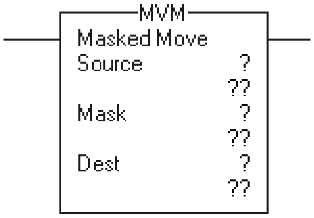
```
value_2 := value _1;
```

Masked Move (MVM)

The MVM instruction copies the Source to a Destination and allows portions of the data to be masked.

This instruction is available in structured text and function block as MVMT, see page 296.

Operands:



Relay Ladder

Operand	Type	Format	Description
Source	SINT INT **DINT**	Immediate Tag	Value to move
	A SINT or INT tag converts to a DINT value by zero-fill.		
Mask	SINT INT **DINT**	Immediate Tag	Which bits to block or pass
	A SINT or INT tag converts to a DINT value by zero-fill.		
Destination	SINT INT **DINT**	Tag	Tag to store the result

Structured Text

```
dest := (Dest AND NOT (Mask))
     OR (Source AND Mask);
```

This instruction is available in structured text as MVMT. Or you can combine bitwise logic within an expression and assign the result to the destination. This expression performs a masked move on *Source*.

See Structured Text Programming for information on the syntax of expressions and assignments within structured text.

Description:

The MVM instruction uses a Mask to either pass or block Source data bits. A "1" in the mask means the data bit is passed. A "0" in the mask means the data bit is blocked.

If you mix integer data types, the instruction fills the upper bits of the smaller integer data types with 0s so that they are the same size as the largest data type.

Enter an Immediate Mask Value

When you enter a mask, the programming software defaults to decimal values. If you want to enter a mask by using another format, precede the value with the correct prefix.

Prefix	Description
16#	Hexadecimal For example; 16#0F0F
8#	Octal For example; 8#16
2#	Binary For example; 2#00110011

Arithmetic Status Flags

Arithmetic status flags are affected.

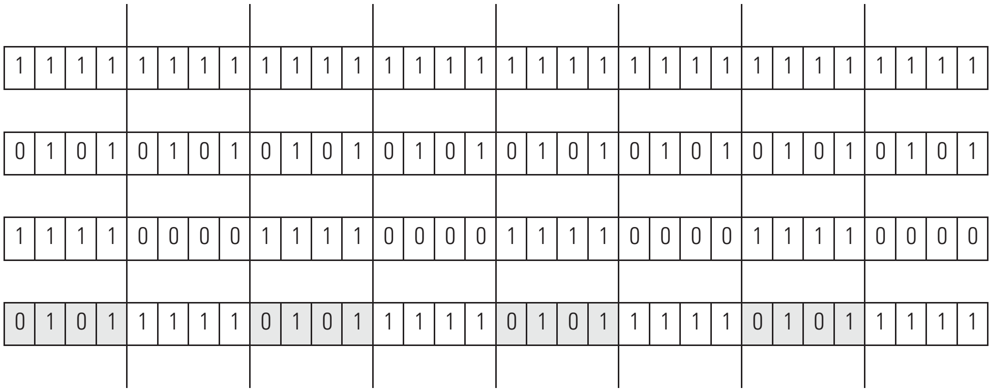
Fault Conditions

None

Execution:

Condition	Relay Ladder Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The instruction passes the Source through the Mask and copies the result into the Destination. Unmasked bits in the Destination remain unchanged. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

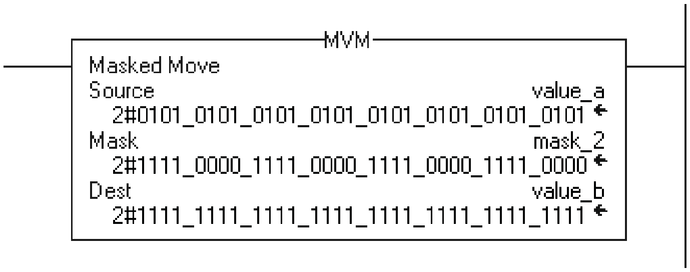
Example: Copy data from *value_a* to *value_b*, while allowing data to be masked (a 0 masks the data in *value_a*).



1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0
0	1	0	1	1	1	1	1	0	1	0	1	1	1	1	1	0	1	0	1	1	1	1	1	0	1	0	1	1	1	1	1

The shaded boxes show the bits that changed in *value_b*.

Relay Ladder



MVM
Masked Move
Source value_a
2#0101_0101_0101_0101_0101_0101_0101_0101 ←
Mask mask_2
2#1111_0000_1111_0000_1111_0000_1111_0000 ←
Dest value_b
2#1111_1111_1111_1111_1111_1111_1111_1111 ←

Structured Text

```
value_b := (value_b AND NOT (mask_2)) OR
    (value_a AND mask_2);
```

Masked Move with Target (MVMT)

The MVMT instruction first copies the Target to the Destination. Then the instruction compares the masked Source to the Destination and makes any required changes to the Destination. The Target and the Source remain unchanged.

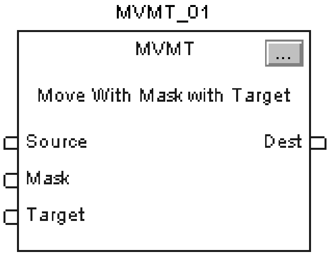
This instruction is available in relay ladder as MVM, see page 293.

Operands:

```
MVMT(MVMT_tag);
```

Structured Text

Variable	Type	Format	Description:
MVMT tag	FBD_MASKED_MOVE	Structure	MVMT structure



Function Block

Operand	Type	Format	Description
MVMT tag	FBD_MASKED_MOVE	Structure	MVMT structure

FBD_MASKED_MOVE Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	**Function Block** If cleared, the instruction does not execute and outputs are not updated. If set, the instruction executes. Default is set. **Structured Text** No effect. The instruction executes.
Source	DINT	Input value to move to Destination based on value of Mask. Valid = any integer
Mask	DINT	Mask of bits to move from Source to Dest. All bits set to one cause the corresponding bits to move from Source to Dest. All bits that are set to zero cause the corresponding bits not to move from Source to Dest. Valid = any integer
Target	DINT	Input value to move to Dest prior to moving Source bits through the Mask. Valid = any integer

Output Parameter	Data Type	Description
EnableOut	BOOL	The instruction produced a valid result.
Dest	DINT	Result of masked move instruction. Arithmetic status flags are set for this output.

Description: When enabled, the MVMT instruction uses a Mask to either pass or block Source data bits. A "1" in the mask means the data bit is passed. A "0" in the mask means the data bit is blocked.

If you mix integer data types, the instruction fills the upper bits of the smaller integer data types with 0s so that they are the same size as the largest data type.

Enter an Immediate Mask Value by Using an Input Reference

When you enter a mask, the programming software defaults to decimal values. If you want to enter a mask by using another format, precede the value with the correct prefix.

Prefix	Description
16#	Hexadecimal For example; 16#0F0F
8#	Octal For example; 8#16
2#	Binary For example; 2#00110011

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions: None

Execution:

Condition	Function Block Action	Structured Text Action
Prescan	No action taken.	No action taken.
Instruction first scan	No action taken.	No action taken.
Instruction first run	No action taken.	No action taken.
EnableIn is cleared	EnableOut is cleared, the instruction does nothing, and the outputs are not updated.	N/A
EnableIn is set	The instruction executes. EnableOut is set.	EnableIn is always set. The instruction executes.
Postscan	No action taken.	No action taken.

Example:

1. Copy Target into Dest.

Target	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Dest	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

2. Mask Source and compare it to Dest. Any required changes are made in Dest. Source and Target remain unchanged. A 0 in the mask restrains the instruction from comparing that bit (shown by x in the example).

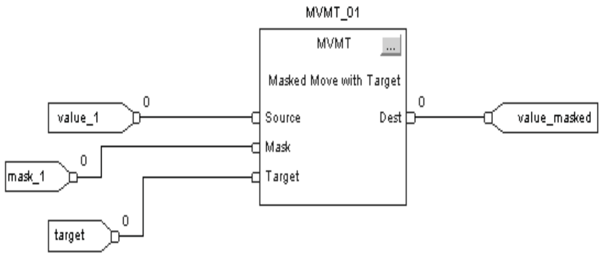
Source	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
Mask1	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0
Dest	0	1	0	1	1	1	1	1	0	1	0	1	1	1	1	1	0	1	0	1	1	1	1	1	0	1	0	1	1	1	1	1

The shaded boxes show the bits that changed.

Structured Text

```
MVMT_01.Source := value_1;
MVMT_01.Mask := mask1;
MVMT_01.Target := target;

MVMT(MVMT_01);

value_masked := MVMT_01.Dest;
```

Function Block

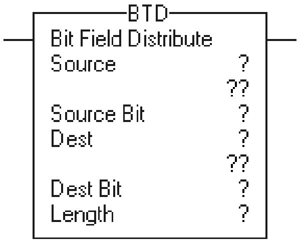


Bit Field Distribute (BTD)

The BTD instruction copies the specified bits from the Source, shifts the bits to the appropriate position, and writes the bits into the Destination.

This instruction is available in structured text and function block as BTDT, see page 302.

Operands:



Relay Ladder

Operand	Type	Format	Description
Source	SINT INT **DINT**	Immediate Tag	Tag that contains the bits to move
	A SINT or INT tag converts to a DINT value by zero-fill.		
Source bit	DINT	Immediate (0-31 DINT) (0-15 INT) (0-7 SINT)	Number of the bit (lowest bit number) from where to start the move Must be within the valid range for the Source data type
Destination	SINT INT **DINT**	Tag	Tag where to move the bits
Destination bit	DINT	Immediate (0-31 DINT) (0-15 INT) (0-7 SINT)	The number of the bit (lowest bit number) where to start copying bits from the Source Must be within the valid range for the Destination data type
Length	DINT	Immediate (1-32)	Number of bits to move

Description:

When enabled, the BTD instruction copies a group of bits from the Source to the Destination. The group of bits is identified by the Source bit (lowest bit number of the group) and the Length (number of bits to copy). The Destination bit identifies the lowest bit number bit to start with in the Destination. The Source remains unchanged.

If the length of the bit field extends beyond the Destination, the instruction does not save the extra bits. Any extra bits do not wrap to the next word.

If you mix integer data types, the instruction fills the upper bits of the smaller integer data types with 0s so that they are the same size as the largest data type.

Arithmetic Status Flags:

Not affected

Fault Conditions:

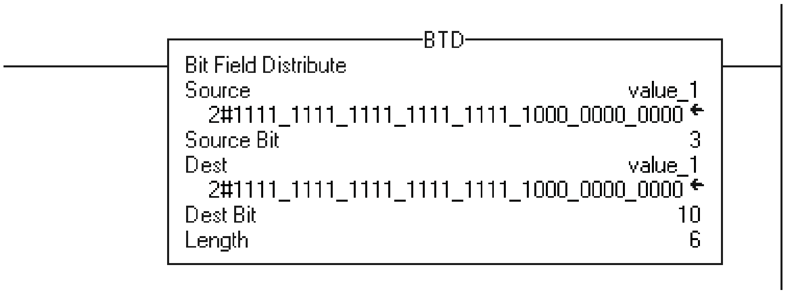
None

Execution:

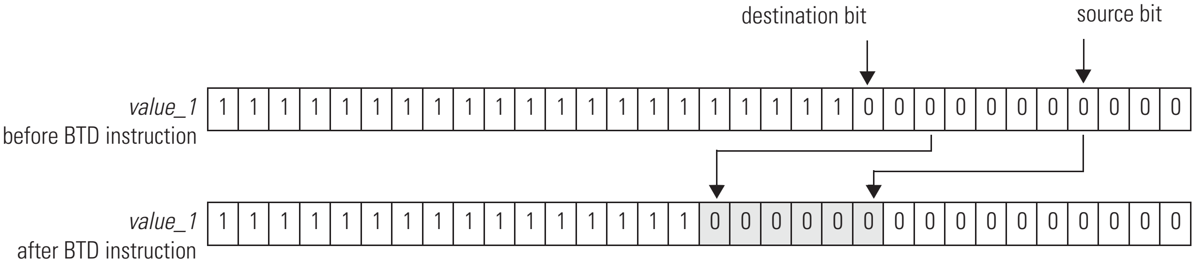
Condition	Relay Ladder Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The instruction copies and shifts the Source bits to the Destination. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Example 1:

When enabled, the BTD instruction moves bits within *value_1*.

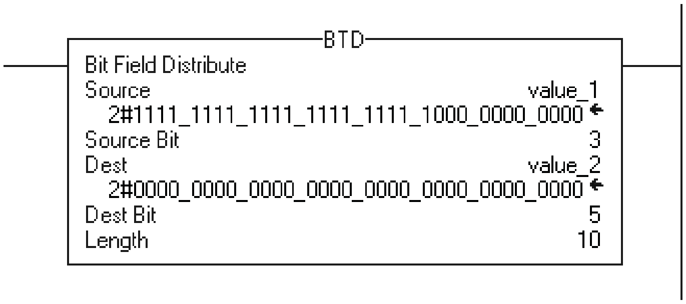


BTD
Bit Field Distribute
Source value_1
2#1111_1111_1111_1111_1111_1000_0000_0000
Source Bit 3
Dest value_1
2#1111_1111_1111_1111_1111_1000_0000_0000
Dest Bit 10
Length 6

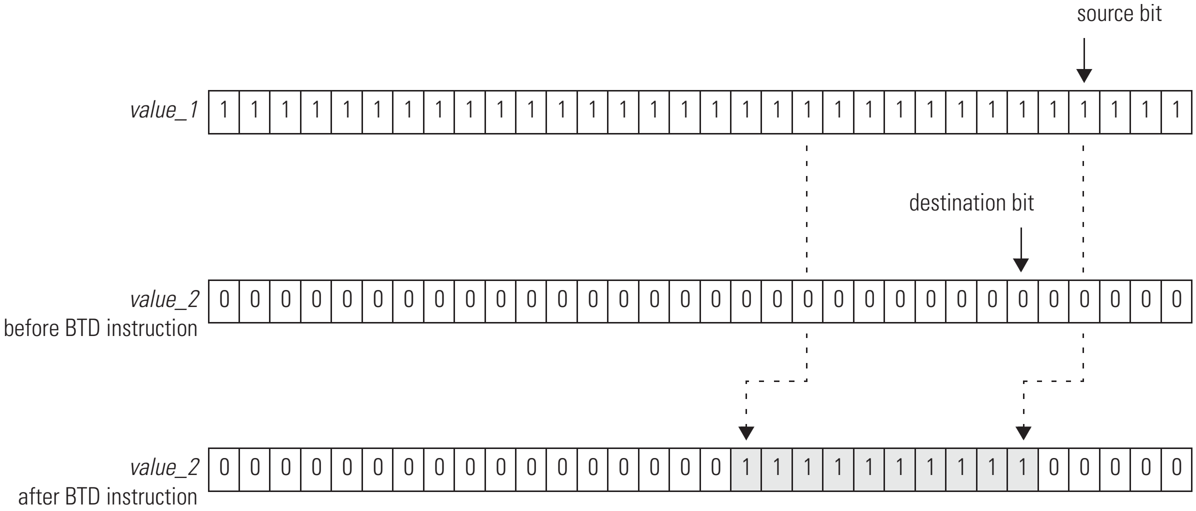


The shaded boxes show the bits that changed in *value_1*.

Example 2: When enabled, the BTD instruction moves 10 bits from *value_1* to *value_2*.



```
BTD
Bit Field Distribute
Source                                   value_1
  2#1111_1111_1111_1111_1111_1000_0000_0000 ←
Source Bit                                     3
Dest                                     value_2
  2#0000_0000_0000_0000_0000_0000_0000_0000 ←
Dest Bit                                       5
Length                                        10
```



The shaded boxes show the bits that changed in *value_2*.

Bit Field Distribute with Target (BTDT)

The BTDT instruction first copies the Target to the Destination. Then the instruction copies the specified bits from the Source, shifts the bits to the appropriate position, and writes the bits into the Destination. The Target and Source remain unchanged.

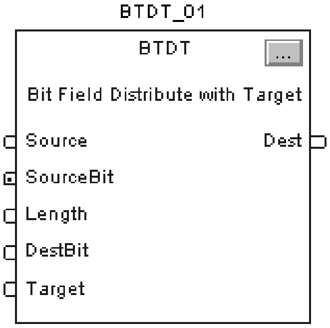
This instruction is available in relay ladder as BTD, see page 299.

Operands:

```
BTDT(BTDT_tag);
```

Structured Text

Variable	Type	Format	Description
BTDT tag	FBD_BIT_FIELD_DISTRIBUTE	Structure	BTDT structure



Function Block

Operand	Type	Format	Description
BTDT tag	FBD_BIT_FIELD_DISTRIBUTE	Structure	BTDT structure

FBD_BIT_FIELD_DISTRIBUTE Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	**Function Block:** If cleared, the instruction does not execute and outputs are not updated. If set, the instruction executes. Default is set. **Structured Text:** No effect. The instruction executes.
Source	DINT	Input value containing the bits to move to Destination. Valid = any integer
SourceBit	DINT	The bit position in Source (lowest bit number from where to start the move). Valid = 0…31
Length	DINT	Number of bits to move Valid = 1…32

Input Parameter	Data Type	Description
DestBit	DINT	The bit position in Dest (lowest bit number to start copying bits into). Valid = 0…31
Target	DINT	Input value to move to Dest prior to moving bits from the Source. Valid = any integer
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	DINT	Result of the bit move operation. Arithmetic status flags are set for this output.

Description: When enabled, the BTD instruction copies a group of bits from the Source to the Destination. The group of bits is identified by the Source bit (lowest bit number of the group) and the Length (number of bits to copy). The Destination bit identifies the lowest bit number bit to start with in the Destination. The Source remains unchanged.

If the length of the bit field extends beyond the Destination, the instruction does not save the extra bits. Any extra bits do not wrap to the next word.

Arithmetic Status Flags: Arithmetic status flags are affected

Fault Conditions: None

Execution:

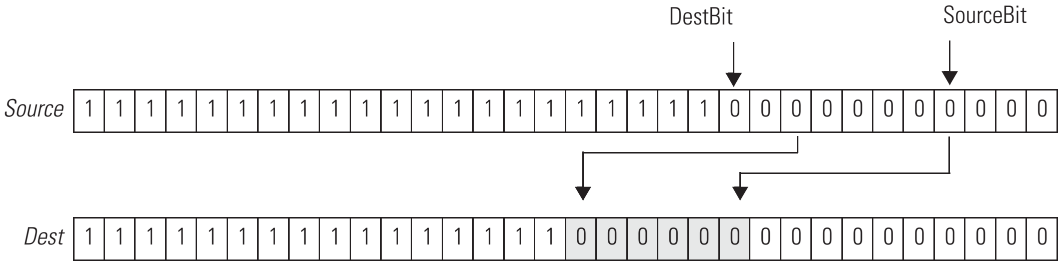
Condition	Function Block Action	Structured Text Action
Prescan	No action taken.	No action taken.
Instruction first scan	No action taken.	No action taken.
Instruction first run	No action taken.	No action taken.
EnableIn is cleared	EnableOut is cleared, the instruction does nothing, and the outputs are not updated.	N/A
EnableIn is set	The instruction executes. EnableOut is set.	EnableIn is always set. The instruction executes.
Postscan	No action taken.	No action taken.

Example:

1. The controller copies Target into Dest.

Target	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
Dest	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0

2. The SourceBit and the Length specify which bits in Source to copy into Dest, starting at DestBit. Source and Target remain unchanged.

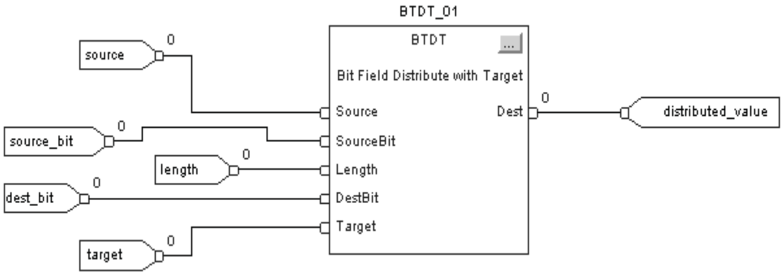


Source	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
Dest	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Structured Text

```
BTDT_01.Source := source;
BTDT_01.SourceBit := source_bit;
BTDT_01.Length := length;
BTDT_01.DestBit := dest_bit;
BTDT_01.Target := target;

BTDT(BTDT_01);

distributed_value := BTDT_01.Dest;
```

Function Block



Clear (CLR)

The CLR instruction clears all the bits of the Destination.

Operands:

Relay Ladder

Operand	Type	Format	Description
Destination	SINT INT **DINT** **REAL**	Tag	Tag to clear

```
dest := 0;
```

Structured Text

Structured text does not have a CLR instruction. Instead, assign 0 to the tag you want to clear. This assignment statement clears *dest*.

See Structured Text Programming for information on the syntax of expressions and assignment statements within structured text.

Description: The CLR instruction clears all the bits of the Destination.

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions: None

Execution:

Condition	Relay Ladder Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The instruction clears the Destination. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Example: Clear all the bits of *value* to 0.

Relay Ladder

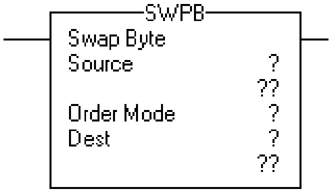
Structured Text

```
value := 0;
```

Swap Byte (SWPB)

The SWPB instruction rearranges the bytes of a value.

Operands:



Relay Ladder

Operand	Type	Format	Enter		
Source	INT DINT REAL	Tag	Tag that contains the bytes that you want to rearrange		
Order Mode			**If the source is an**	**And you want to change the bytes to this pattern (each letter represents a different byte)**	**Then select**
			INT	N/A	Any of the options
			DINT REAL	ABCD DCBA	REVERSE (or enter 0)
				ABCD CDAB	WORD (or enter 1)
				ABCD BADC	HIGH/LOW (or enter 2)
Destination	INT DINT REAL	Tag	Tag to store the bytes in the new order		
			If the source is an	**Then the destination must be an**	
			INT	INT	
				DINT	
			DINT	DINT	
			REAL	REAL	

Structured Text

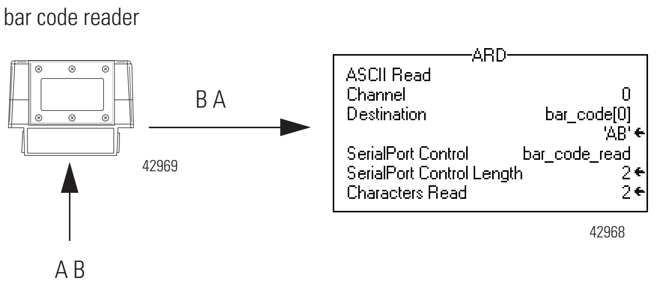
```
SWPB(Source,OrderMode,Dest);
```

The operands are the same as those for the relay ladder SWPB instruction. If you select the HIGH/LOW order mode, enter it as HIGHLOW or HIGH_LOW (without the slash).

Description:

The SWPB instruction rearranges the order of the bytes of the Source. It places the result in the Destination.

When you read or write ASCII characters, you typically *do not* need to swap characters. The ASCII read and write instructions (ARD, ARL, AWA, AWT) automatically swap characters, as shown below.



Tag Name	Value	Style	Type
bar_code[0]	AB	ASCII	INT

Arithmetic Status Flags:

Not affected

Fault Conditions:

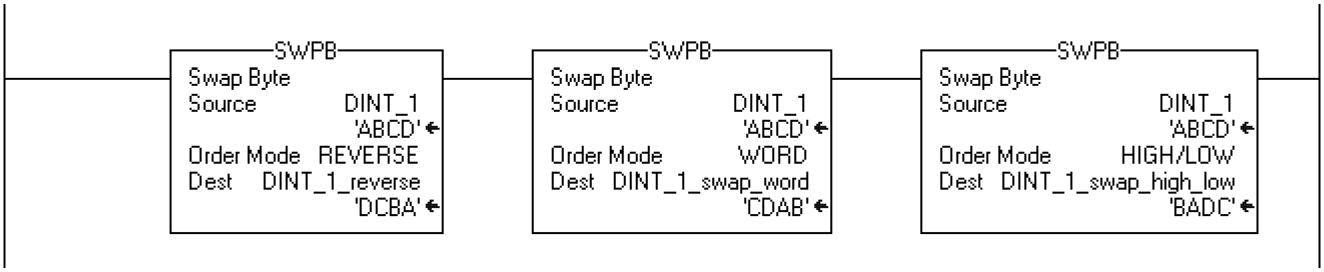
None

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction rearranges the specified bytes.	The instruction rearranges the specified bytes.
Postscan	The rung-condition-out is set to false.	No action taken.

Example 1: The three SWPB instructions each reorder the bytes of *DINT_1* according to a different order mode. The display style is ASCII, and each character represents one byte. Each instruction places the bytes, in the new order, in a different Destination.

Relay Ladder



SWPB
Swap Byte
Source DINT_1
'ABCD'
Order Mode REVERSE
Dest DINT_1_reverse
'DCBA'

SWPB
Swap Byte
Source DINT_1
'ABCD'
Order Mode WORD
Dest DINT_1_swap_word
'CDAB'

SWPB
Swap Byte
Source DINT_1
'ABCD'
Order Mode HIGH/LOW
Dest DINT_1_swap_high_low
'BADC'

Structured Text

```
SWPB(DINT_1,REVERSE,DINT_1_reverse);
SWPB(DINT_1,WORD,DINT_1_swap_word);
SWPB(DINT_1,HIGHLOW,DINT_1_swap_high_low);
```

Example 2: The following example reverses the bytes in each element of an array. For an RSLogix 5000 project that contains this example, open the RSLogix 5000\Projects\Samples folder, Swap_Bytes_in_Array.ACD file.

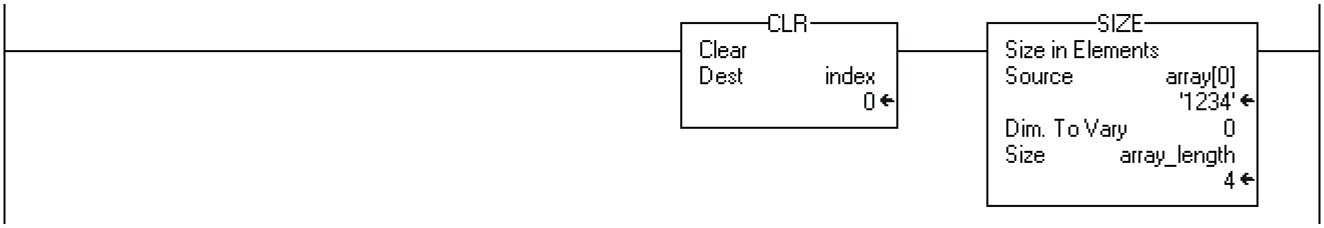
1. Initialize the tags. The SIZE instruction finds the number of elements in *array* and stores that value in *array_length*. A subsequent instruction uses this value to determine when the routine has acted on all the elements in the array.

2. Reverse the bytes in one element of *array.*
   - The SWPB instruction reverses the bytes of the element number that is indicated by the value of *index*. For example, when *index* equals 0, the SWPB instruction acts on *array[0]*.
   - The ADD instruction increments *index*. The next time the instruction executes, the SWPB instruction acts on the next element in *array.*

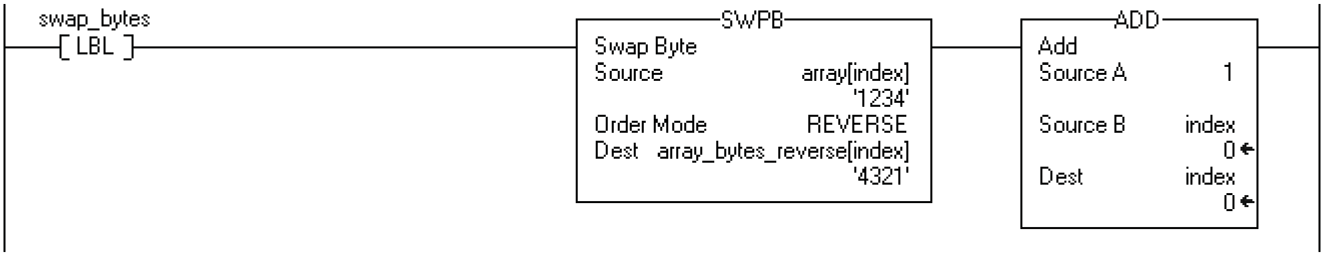
3. Determine when the SWPB instruction has acted on all the elements in the array.
   - If *index* is less then the number of elements in the array (*array_length*), then continue with the next element in the array.
   - If *index* equals *array_length*, then the SWPB has acted on all the elements in the array.

Relay Ladder

Initialize the tags.



Reverse the bytes.



Determine whether the SWPB instruction has acted on all the elements in the array.

Structured Text

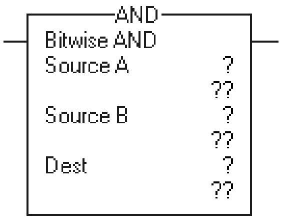
```
index := 0;
SIZE (array[0],0,array_length);
REPEAT
 SWPB(array[index],REVERSE,array_bytes_reverse[index]);
 index := index + 1;
UNTIL(index >= array_length)END_REPEAT;
```

Bitwise AND (AND)

The AND instruction performs a bitwise AND operation by using the bits in Source A and Source B and places the result in the Destination.

To perform a logical AND, see page 325.

Operands:



Relay Ladder

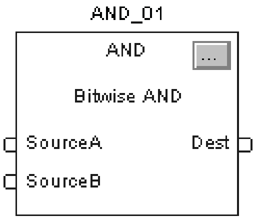
Operand	**Type**	**Format**	**Description**
Source A	SINT INT **DINT**	Immediate Tag	Value to AND with Source B
	A SINT or INT tag converts to a DINT value by zero-fill.		
Source B	SINT INT **DINT**	Immediate Tag	Value to AND with Source A
	A SINT or INT tag converts to a DINT value by zero-fill.		
Destination	SINT INT **DINT**	Tag	Stores the result

```
dest := sourceA AND sourceB
```

Structured Text

Use AND or the ampersand sign “&” as an operator within an expression. This expression evaluates *sourceA* AND *sourceB*.

See Structured Text Programming for information on the syntax of expressions within structured text.



Function Block

Operand	**Type**	**Format**	**Description**
AND tag	FBD_LOGICAL	Structure	AND structure

FBD_LOGICAL Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
SourceA	DINT	Value to AND with SourceB. Valid = any integer
SourceB	DINT	Value to AND with SourceA. Valid = any integer
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	DINT	Result of the instruction. Arithmetic status flags are set for this output.

Description: When enabled, the instruction evaluates the AND operation:

If the bit in Source A is	And the bit in Source B is	The bit in the destination is
0	0	0
0	1	0
1	0	0
1	1	1

If you mix integer data types, the instruction fills the upper bits of the smaller integer data types with 0s so that they are the same size as the largest data type.

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions: None

Execution:

Relay Ladder

Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The instruction performs a bitwise AND operation. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

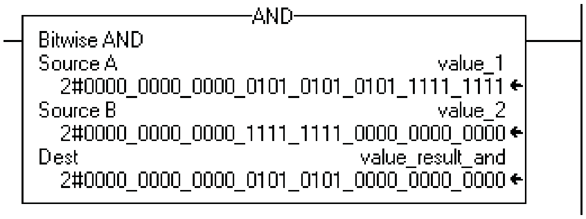
Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: When enabled, the AND instruction performs a bitwise AND operation on SourceA and SourceB and places the result in the Dest.

SourceA	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1	0	1	1	1	1	1	1	1	1	1
SourceB	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
Dest	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0

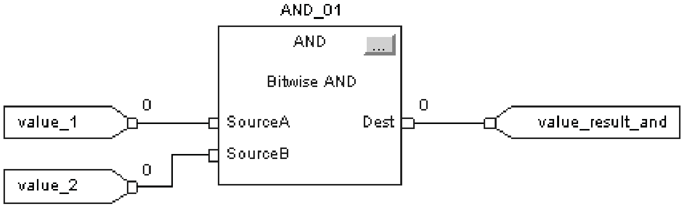
Relay Ladder


AND
Bitwise AND
Source A value_1
2#0000_0000_0000_0101_0101_0101_1111_1111
Source B value_2
2#0000_0000_0000_1111_1111_0000_0000_0000
Dest value_result_and
2#0000_0000_0000_0101_0101_0000_0000_0000

Structured Text

```
value_result_and := value_1 AND value_2;
```

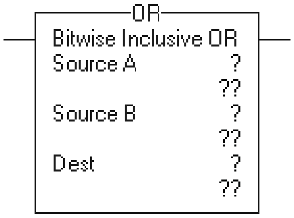
Function Block



Bitwise OR (OR)

The OR instruction performs a bitwise OR operation by using the bits in Source A and Source B and places the result in the Destination.

Operands:



Relay Ladder

Operand	Type	Format	Description
Source A	SINT INT **DINT**	Immediate Tag	Value to OR with Source B
	A SINT or INT tag converts to a DINT value by zero-fill.		
Source B	SINT INT **DINT**	Immediate Tag	Value to OR with Source A
	A SINT or INT tag converts to a DINT value by zero-fill.		
Destination	SINT INT **DINT**	Tag	Stores the result

```
dest := sourceA OR sourceB
```

Structured Text

Use OR as an operator within an expression. This expression evaluates *sourceA* OR *sourceB*.

See Structured Text Programming for information on the syntax of expressions within structured text.

Function Block

Operand	Type	Format:	Description
OR tag	FBD_LOGICAL	Structure	OR structure

FBD_LOGICAL Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
SourceA	DINT	Value to OR with SourceB. Valid = any integer
SourceB	DINT	Value to OR with SourceA. Valid = any integer
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	DINT	Result of the instruction. Arithmetic status flags are set for this output.

Description: When enabled, the instruction evaluates the OR operation.

If the bit in Source A is	And the bit in Source B is	The bit in the destination is
0	0	0
0	1	1
1	0	1
1	1	1

If you mix integer data types, the instruction fills the upper bits of the smaller integer data types with 0s so that they are the same size as the largest data type.

Arithmetic Status Flags Arithmetic status flags are affected.

Fault Conditions: None

Execution:

Relay Ladder

Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The instruction performs a bitwise OR operation. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

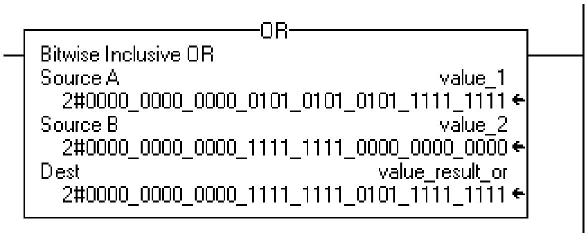
Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: When enabled, the OR instruction performs a bitwise OR operation on SourceA and SourceB and places the result in Dest.

SourceA	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1	0	1	1	1	1	1	1	1	1	1
SourceB	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
Dest	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1	1	1	1

Relay Ladder

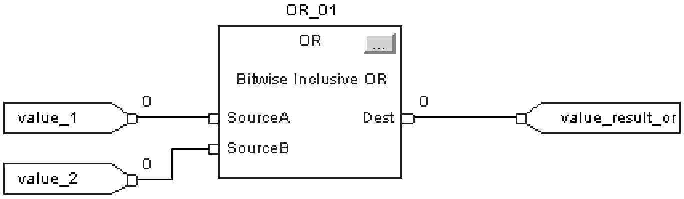


Bitwise Inclusive OR
Source A value_1
2#0000_0000_0000_0101_0101_0101_1111_1111
Source B value_2
2#0000_0000_0000_1111_1111_0000_0000_0000
Dest value_result_or
2#0000_0000_0000_1111_1111_0101_1111_1111

Structured Text

```
value_result_or := value_1 OR value_2;
```

Function Block

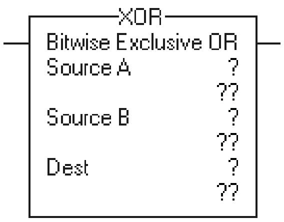


Bitwise Exclusive OR (XOR)

The XOR instruction performs a bitwise XOR operation by using the bits in Source A and Source B and places the result in the Destination.

To perform a logical XOR, see page 331.

Operands:



Relay Ladder

Operand	Type	Format	Description
Source A	SINT INT DINT	Immediate Tag	Value to XOR with Source B
	A SINT or INT tag converts to a DINT value by zero-fill.		
Source B	SINT INT DINT	Immediate Tag	Value to XOR with Source A
	A SINT or INT tag converts to a DINT value by zero-fill.		
Destination	SINT INT DINT	Tag	Stores the result

```
dest := sourceA XOR sourceB
```

Structured Text

Use XOR as an operator within an expression. This expression evaluates *sourceA* XOR *sourceB*.

See Structured Text Programming for information on the syntax of expressions within structured text.

Function Block

Operand	Type	Format	Description
XOR tag	FBD_LOGICAL	Structure	XOR structure

FBD_LOGICAL Structure

Input Parameter	**Data Type**	**Description**
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
SourceA	DINT	Value to XOR with SourceB. Valid = any integer
SourceB	DINT	Value to XOR with SourceA. Valid = any integer
Output Parameter:	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	DINT	Result of the instruction. Arithmetic status flags are set for this output.

Description:

When enabled, the instruction evaluates the XOR operation.

If the bit in Source A is	**And the bit in Source B is**	**The bit in the destination is**
0	0	0
0	1	1
1	0	1
1	1	0

If you mix integer data types, the instruction fills the upper bits of the smaller integer data types with 0s so that they are the same size as the largest data type.

Arithmetic Status Flags

Arithmetic status flags are affected.

Fault Conditions:

None

Execution:

Relay Ladder

Condition	**Action**
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The instruction performs a bitwise OR operation. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

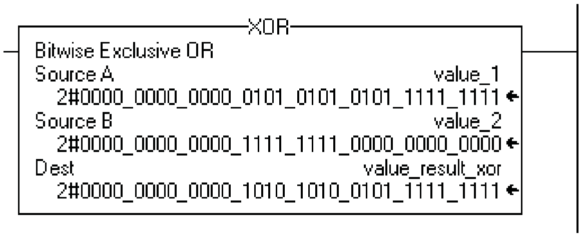
Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: When enabled, the XOR instruction performs a bitwise XOR operation on SourceA and SourceB and places the result in the destination tag.

value_1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1	0	1	1	1	1	1	1	1	1	1
value_2	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
value_result_xor	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	0	1	0	1	1	1	1	1	1	1	1	1

Relay Ladder

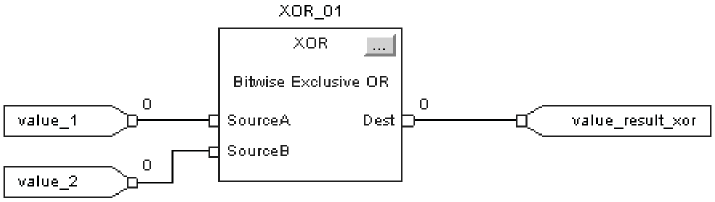


XOR
Bitwise Exclusive OR
Source A value_1
2#0000_0000_0000_0101_0101_0101_1111_1111
Source B value_2
2#0000_0000_0000_1111_1111_0000_0000_0000
Dest value_result_xor
2#0000_0000_0000_1010_1010_0101_1111_1111

Structured Text

```
value_result_xor := value_1 XOR value_2;
```

Function Block

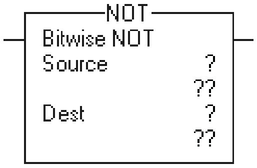


Bitwise NOT (NOT)

The NOT instruction performs a bitwise NOT operation by using the bits in the Source and places the result in the Destination.

To perform a logical NOT, see page 334.

Operands:



Relay Ladder

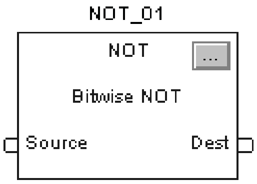
Operand	Type	Format	Description
Source	SINT INT DINT	Immediate Tag	Value to NOT
	A SINT or INT tag converts to a DINT value by zero-fill.		
Destination	SINT INT DINT	Tag	Stores the result

```
dest := NOT source
```

Structured Text

Use NOT as an operator within an expression. This expression evaluates NOT *source*.

See Structured Text Programming for information on the syntax of expressions within structured text.



Function Block

Operand	Type	Format	Description
NOT tag	FBD_LOGICAL	Structure	NOT structure

FBD_LOGICAL Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. default is set
Source	DINT	Value to NOT. valid = any integer
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	DINT	Result of the instruction. Arithmetic status flags are set for this output.

Description: When enabled, the instruction evaluates the NOT operation.

If the bit in the Source is	The bit in the destination is
0	1
1	0

If you mix integer data types, the instruction fills the upper bits of the smaller integer data types with 0s so that they are the same size as the largest data type.

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions: None

Execution:

Relay Ladder

Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The instruction performs a bitwise NOT operation. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

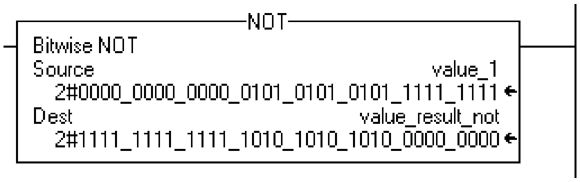
Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: When enabled, the NOT instruction performs a bitwise NOT operation on Source and places the result in Dest.

value_1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1	0	1	1	1	1	1	1	1	1	1
value_result_not	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0

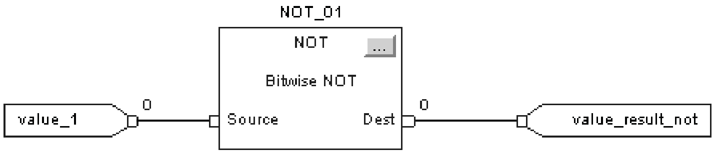
Relay Ladder



Structured Text

```
value_result_not := NOT value_1;
```

Function Block



Boolean AND (BAND)

The BAND instruction logically ANDs as many as eight boolean inputs.

To perform a bitwise AND, see page 311.

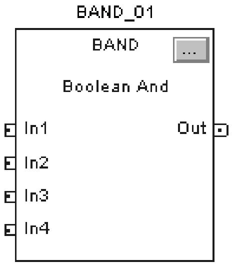
Operands:

```
IF operandA AND operandB THEN
    <statement>;
END_IF;
```

Structured Text

Use AND or the ampersand sign “&” as an operator within an expression. The operands must be BOOL values or expressions that evaluate to BOOL values. This expression evaluates whether *operandA* and *operandB* are both set (true).

See Appendix B for information on the syntax of expressions within structured text.



Function Block

Operand	Type	Format	Description
BAND tag	FBD_BOOLEAN_AND	Structure	BAND structure

FBD_BOOLEAN_AND Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
In1	BOOL	First boolean input. Default is set.
In2	BOOL	Second boolean input. Default is set.
In3	BOOL	Third boolean input. Default is set.
In4	BOOL	Fourth boolean input. Default is set.
In5	BOOL	Fifth boolean input. default is set.
In6	BOOL	Sixth boolean input. Default is set.

Input Parameter	Data Type	Description
In7	BOOL	Seventh boolean input. Default is set.
In8	BOOL	Eighth boolean input. Default is set.
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	Enable output.
Out	BOOL	The output of the instruction.

Description:

The BAND instruction ANDs as many as eight boolean inputs. If an input is not used, it defaults to set (1).

Out = In1 AND In2 AND In3 AND In4 AND In5 AND In6 AND In7 AND In8

Arithmetic Status Flags:

Not affected

Fault Conditions:

None

Execution:

Condition	Function Block Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example 1:

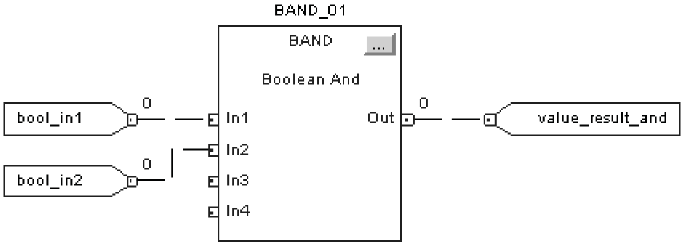
This example ANDs *bool_in1* and *bool_in2* and places the result in *value_result_and*.

If BOOL_IN1 is	If BOOL_IN2 is	Then VALUE_RESULT_AND is
0	0	0
0	1	0
1	0	0
1	1	1

Structured Text

```
value_result_and := bool_in1 AND bool_in2;
```

Function Block



Example 2: If both *bool_in1* and *bool_in2* are set (true), *light1* is set (on). Otherwise, *light1* is cleared (off).

Structured Text

```
IF bool_in1 AND bool_in2 THEN
    light1 := 1;
ELSE
    light1 := 0;
END_IF;
```

Boolean OR (BOR)

The BOR instruction logically ORs as many as eight boolean inputs.

To perform a bitwise OR, see page 314.

Operands:

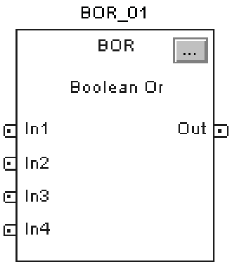
Structured Text

```
IF operandA OR operandB THEN
    <statement>;
END_IF;
```

Use OR as an operator within an expression. The operands must be BOOL values or expressions that evaluate to BOOL values. This expression evaluates whether *operandA* or *operandB* or both are set (true).

See Appendix B for information on the syntax of expressions within structured text.

Function Block



Operand	Type	Format	Description
BOR tag	FBD_BOOLEAN_OR	Structure	BOR structure

FBD_BOOLEAN_OR Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
In1	BOOL	First boolean input. Default is cleared.
In2	BOOL	Second boolean input. Default is cleared.
In3	BOOL	Third boolean input. Default is cleared.
In4	BOOL	Fourth boolean input. Default is cleared.
In5	BOOL	Fifth boolean input. Default is cleared.
In6	BOOL	Sixth boolean input. Default is cleared.
In7	BOOL	Seventh boolean input. Default is cleared.

Input Parameter	Data Type	Description
In8	BOOL	Eighth boolean input. Default is cleared.
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	Enable output.
Out	BOOL	The output of the instruction.

Description: The BOR instruction ORs as many as eight boolean inputs. If an input is not used, it defaults to cleared (0).

Out = In1 OR In2 OR In3 OR In4 OR In5 OR In6 OR In7 OR In8

Arithmetic Status Flags: Not affected

Fault Conditions: None

Execution:

Condition	Function Block Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

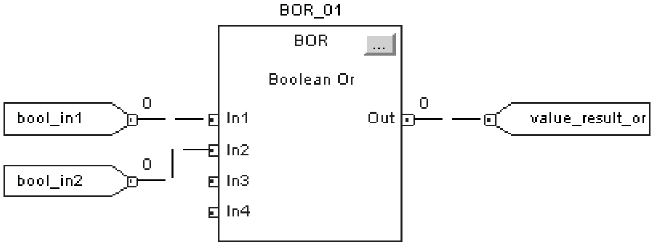
Example 1: This example ORs *bool_in1* and *bool_in2* and places the result in *value_result_or*.

If BOOL_IN1 is	If BOOL_IN2 is	Then VALUE_RESULT_OR is
0	0	0
0	1	1
1	0	1
1	1	1

Structured Text

```
value_result_or := bool_in1 OR bool_in2;
```

Function Block



Example 2: In this example, *light1* is set (on) if:

- only *bool_in1* is set (true).
- only *bool_in2* is set (true).
- both *bool_in1* and *bool_in2* are set (true).

Otherwise, *light1* is cleared (off).

Structured Text

```
IF bool_in1 OR bool_in2 THEN
    light1 := 1;
ELSE
    light1 := 0;
END_IF;
```

Boolean Exclusive OR (BXOR)

The BXOR performs an exclusive OR on two boolean inputs.

To perform a bitwise XOR, see page 318.

Operands:

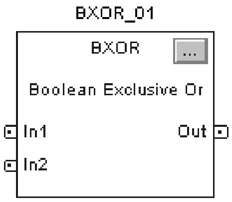
Structured Text

```
IF operandA XOR operandB THEN
    <statement>;
END_IF;
```

Use XOR as an operator within an expression. The operands must be BOOL values or expressions that evaluate to BOOL values. This expression evaluates whether only *operandA* or only *operandB* is set (true).

See Appendix B for information on the syntax of expressions within structured text.

Function Block



Operand	Type	Format	Description
BXOR tag	FBD_BOOLEAN_XOR	Structure	BXOR structure

FBD_BOOLEAN_XOR Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
In1	BOOL	First boolean input. Default is cleared.
In2	BOOL	Second boolean input. Default is cleared.
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	Enable output.
Out	BOOL	The output of the instruction.

Description: The BXOR instruction performs an exclusive OR on two boolean inputs.

Out = In1 XOR In2

Arithmetic Status Flags: Not affected

Fault Conditions: None

Execution:

Condition	Function Block Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

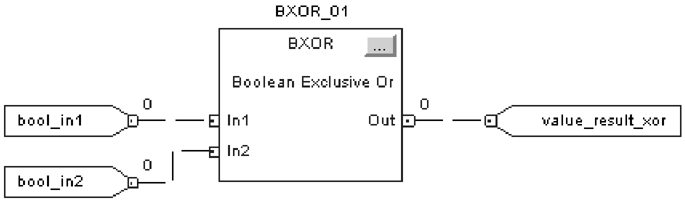
Example 1: This example performs an exclusive OR on *bool_in1* and *bool_in2* and places the result in *value_result_xor*.

If BOOL_IN1 is	If BOOL_IN2 is	Then VALUE_RESULT_XOR is
0	0	0
0	1	1
1	0	1
1	1	0

Structured Text

```
value_result_xor := bool_in1 XOR bool_in2;
```

Function Block



Example 2: In this example, *light1* is set (on) if

- only *bool_in1* is set (true).
- only *bool_in2* is set (true).

Otherwise, *light1* is cleared (off).

Structured Text

```
IF bool_in1 XOR bool_in2 THEN
    light1 := 1;
ELSE
    light1 := 0;
END_IF;
```

Boolean NOT (BNOT)

The BNOT instruction complements a boolean input.

To perform a bitwise NOT, see page 322.

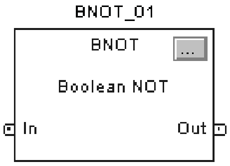
Operands:

```
IF NOT operand THEN
    <statement>;
END_IF;
```

Structured Text

Use NOT as an operator within an expression. The operand must be a BOOL values or expressions that evaluate to BOOL values. This expression evaluates whether *operand* is cleared (false).

See Structured Text Programming for information on the syntax of expressions within structured text.



Function Block

Operand	Type	Format	Description
BNOT tag	FBD_BOOLEAN_NOT	Structure	BNOT structure

FBD_BOOLEAN_NOT Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
In	BOOL	Input to the instruction. Default is set.
Output Parameter	**Data Type**	**Description:**
EnableOut	BOOL	Enable output.
Out	BOOL	The output of the instruction.

Description:

The BNOT instruction complements a boolean input.

Out = NOT In

Arithmetic Status Flags:

Not affected

Fault Conditions:

None

Execution:

Condition	Function Block Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

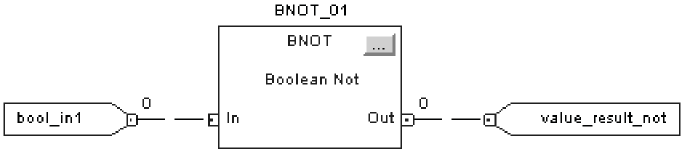
Example 1: This example complements *bool_in1* and places the result in *value_result_not*.

If BOOL_IN1 is	Then VALUE_RESULT_NOT is
0	1
1	0

Structured Text

```
value_result_not := NOT bool_in1;
```

Function Block



Example 2: If *bool_in1* is cleared, *light1* is cleared (off). Otherwise, *light1* is set (on).

Structured Text

```
IF NOT bool_in1 THEN
    light1 := 0;
ELSE
    light1 := 1;
END_IF;
```

Notes:

Array (File)/Misc. Instructions

(FAL, FSC, COP, CPS, FLL, AVE, SRT, STD, SIZE)

Introduction

The file/miscellaneous instructions operate on arrays of data.

If you want to	Use this instruction	Available in these languages	Page
Perform arithmetic, logic, shift, and function operations on values in arrays	FAL	Relay ladder Structured text[1]	343
Search for and compare values in arrays	FSC	Relay ladder	354
Copy the contents of one array into another array	COP	Relay ladder Structured text	363
Copy the contents of one array into another array without interruption	CPS	Relay ladder Structured text	363
Fill an array with specific data	FLL	Relay ladder Structured text[1]	369
Calculate the average of an array of values	AVE	Relay ladder Structured text[1]	373
Sort one dimension of array data into ascending order	SRT	Relay ladder Structured text	378
Calculate the standard deviation of an array of values	STD	Relay ladder Structured text[1]	383
Find the size of a dimension of an array	SIZE	Relay ladder Structured text	389

(1) There is no equivalent structured text instruction. Use other structured text programming to achieve the same result. See the description for the instruction.

You can mix data types, but loss of accuracy and rounding error might occur and the instruction takes more time to execute. Check the S:V bit to see whether the result was truncated.

For relay ladder instructions, **bold** data types indicate optimal data types. An instruction executes faster and requires less memory if all the operands of the instruction use the same optimal data type, typically DINT or REAL.

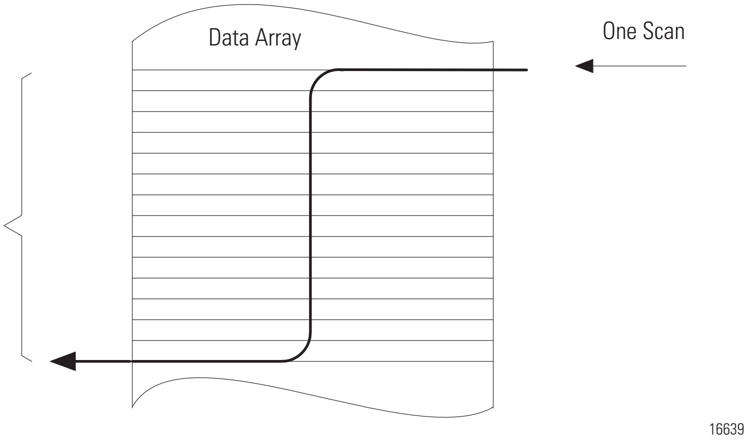
Selecting Mode of Operation

For FAL and FSC instructions, the mode tells the controller how to distribute the array operation.

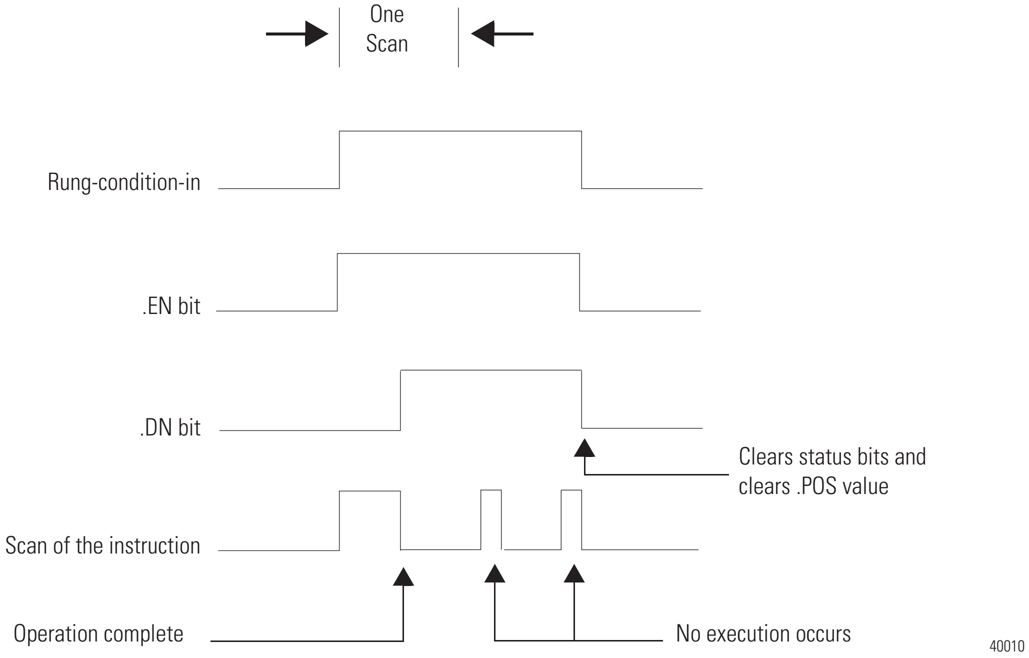
If You Want To	Select This Mode
Operate on all of the specified elements in an array before continuing on to the next instruction	All
Distribute array operation over a number of scans Enter the number of elements to operate on per scan (1-2147483647)	Numerical
Manipulate one element of the array each time the rung-condition-in goes from false to true	Incremental

All Mode

In All mode, all the specified elements in the array are operated on before continuing on to the next instruction. The operation begins when the instruction’s rung-condition-in goes from false to true. The position (.POS) value in the control structure points to the element in the array that the instruction is currently using. Operation stops when the .POS value equals the .LEN value.



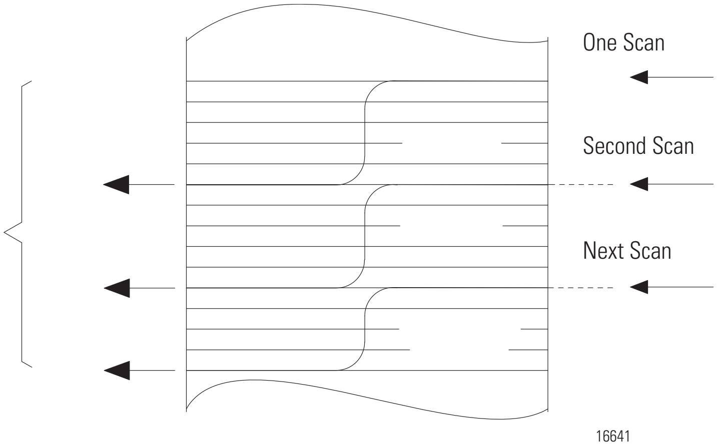
The following timing diagram shows the relationship between status bits and instruction operation. When the instruction execution is complete, the .DN bit is set. The .DN bit, the .EN bit, and the .POS value are cleared when the rung-condition-in is false. Only then can another execution of the instruction be triggered by a false-to-true transition of rung-condition-in.



Numerical Mode

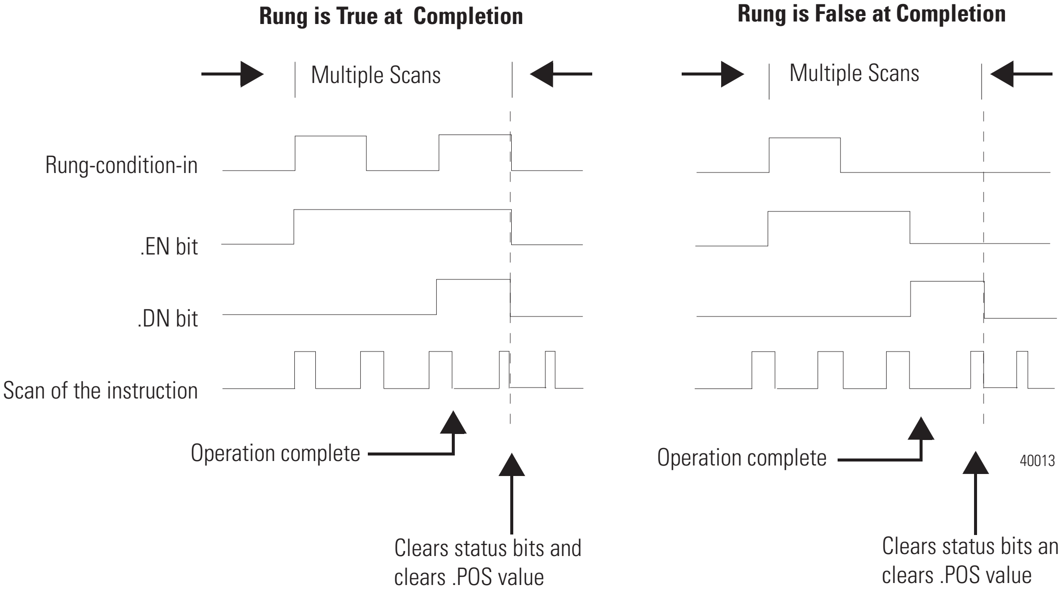
Numerical mode distributes the array operation over a number of scans. This mode is useful when working with non-time-critical data or large amounts of data. You enter the number of elements to operate on for each scan, which keeps scan time shorter.

Execution is triggered when the rung-condition-in goes from false to true. Once triggered, the instruction is executed each time it is scanned for the number of scans necessary to complete operating on the entire array. Once triggered, rung-condition-in can change repeatedly without interrupting execution of the instruction.



IMPORTANT Avoid using the results of a file instruction operating in numerical mode until the .DN bit is set.

The following timing diagram shows the relationship between status bits and instruction operation. When the instruction execution is complete, the .DN bit is set.

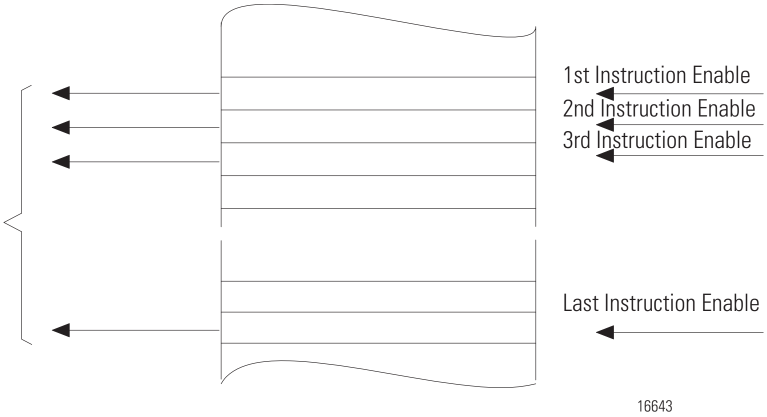


If the rung-condition-in is true at completion, the .EN and .DN bit are set until the rung-condition-in goes false. When the rung-condition-in goes false, these bits are cleared and the .POS value is cleared.

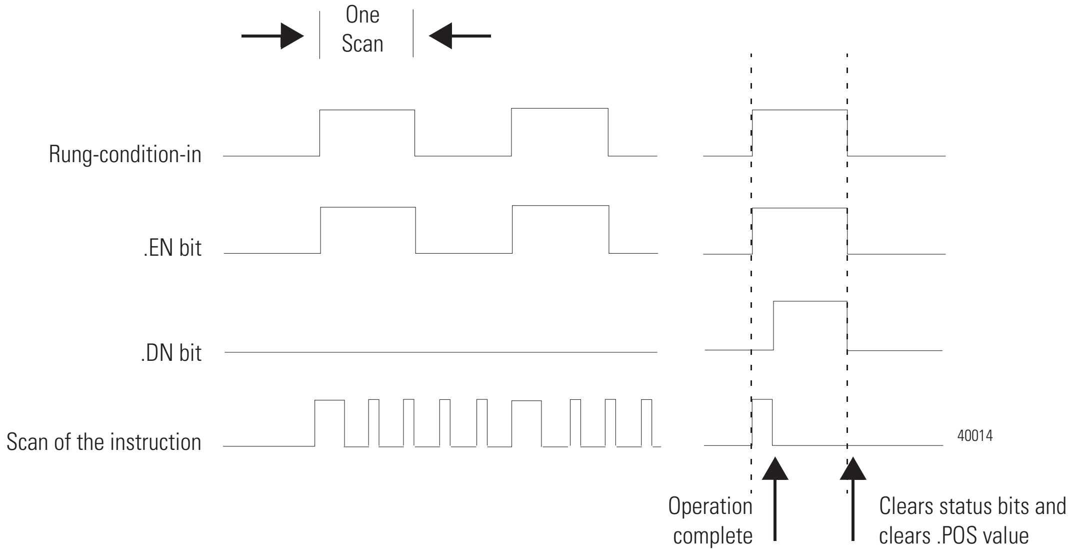
If the rung-condition-in is false at completion, the .EN bit is cleared immediately. One scan after the .EN bit is cleared, the .DN bit and the .POS value are cleared.

Incremental Mode

Incremental mode manipulates one element of the array each time the instruction's rung-condition-in goes from false to true.



The following timing diagram shows the relationship between status bits and instruction operation. Execution occurs only in a scan in which the rung-condition-in goes from false to true. Each time this occurs, only one element of the array is manipulated. If the rung-condition-in remains true for more than one scan, the instruction only executes during the first scan.



The .EN bit is set when rung-condition-in is true. The .DN bit is set when the last element in the array has been manipulated. When the last element has been manipulated and the rung-condition-in goes false, the .EN bit, the .DN bit, and the .POS value are cleared.

The difference between incremental mode and numerical mode at a rate of one element per scan is:

- Numerical mode with any number of elements per scan requires only one false-to-true transition of the rung-condition-in to start execution. The instruction continues to execute the specified number of elements each scan until completion regardless of the state of the rung-condition-in.

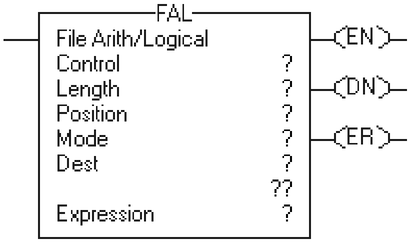
- Incremental mode requires the rung-condition-in to change from false to true to manipulate one element in the array.

File Arithmetic and Logic (FAL)

The FAL instruction performs copy, arithmetic, logic, and function operations on data stored in an array.

Operands:

Relay Ladder



Operand	Type	Format	Description
Control	CONTROL	Tag	Control structure for the operation
Length	DINT	Immediate	Number of elements in the array to be manipulated
Position	DINT	Immediate	Current element in array Initial value is typically 0
Mode	DINT	Immediate	How to distribute the operation Select INC, ALL, or enter a number
Destination	SINT INT **DINT** **REAL**	Tag	Tag to store the result
Expression	SINT INT **DINT** **REAL**	Immediate Tag	An expression consisting of tags and/or immediate values separated by operators
	A SINT or INT tag converts to a DINT value by sign-extension.		

Structured Text

Structured text does not have an FAL instruction, but you can achieve the same results by using a SIZE instruction and a FOR...DO or other loop construct.

```
SIZE(destination,0,length-1);
FOR position = 0 TO length DO
    destination[position] := numeric_expression;
END_FOR;
```

See Structured Text Programming for information on the syntax of constructs within structured text.

CONTROL Structure

Mnemonic	Data Type	Description
.EN	BOOL	The enable bit indicates that the FAL instruction is enabled.
.DN	BOOL	The done bit is set when the instruction has operated on the last element (.POS = .LEN).
.ER	BOOL	The error bit is set if the expression generates an overflow (S:V is set). The instruction stops executing until the program clears the .ER bit. The .POS value contains the position of the element that caused the overflow.
.LEN	DINT	The length specifies the number of elements in the array on which the FAL instruction operates.
.POS	DINT	The position contains the position of the current element that the instruction is accessing.

Description:

The FAL instruction performs the same operations on arrays as the CPT instruction performs on elements.

The examples that start on page 350 show how to use the .POS value to step through an array. If a subscript in the expression of the Destination is out of range, the FAL instruction generates a major fault (type 4, code 20).

Arithmetic Status Flags:

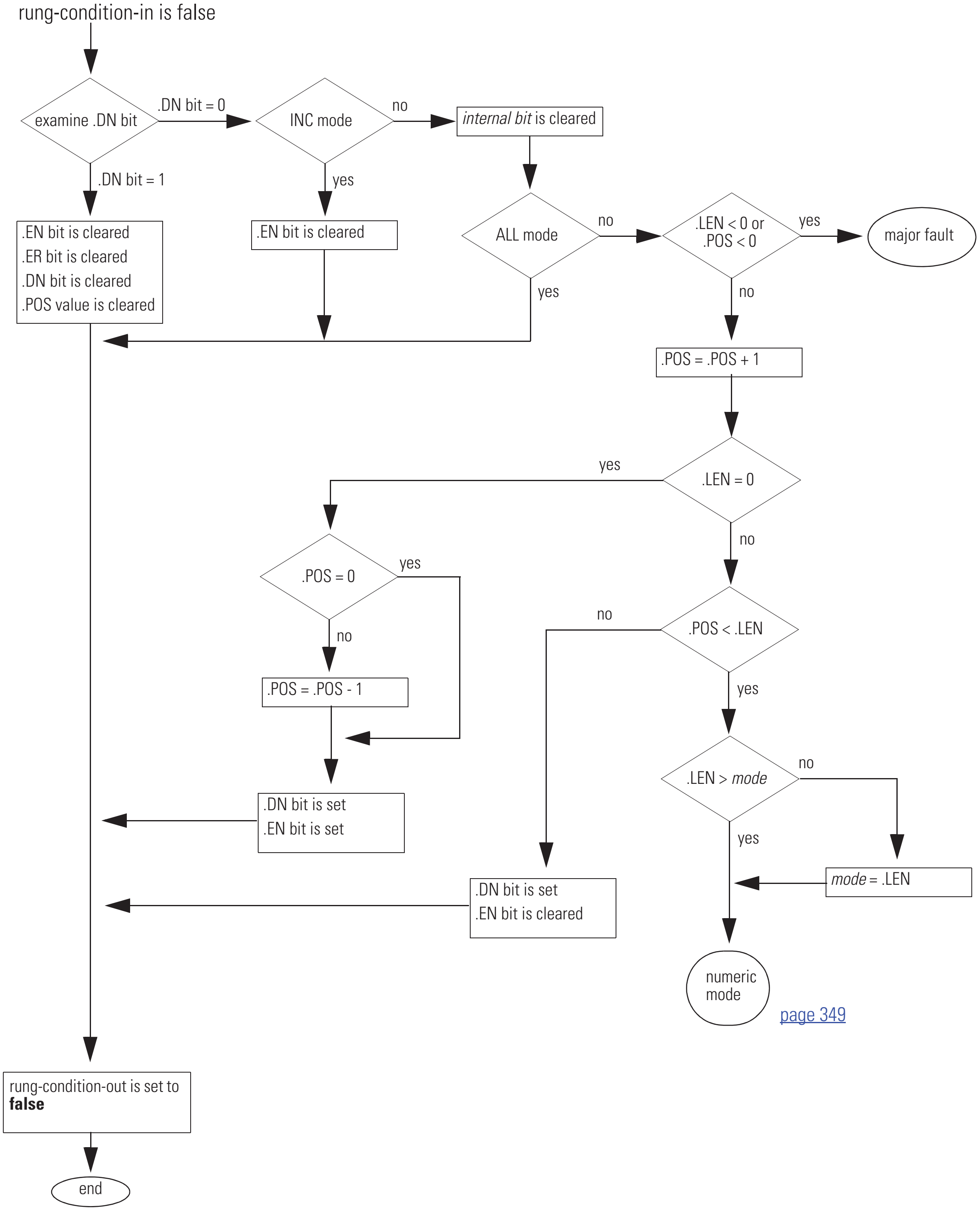
Arithmetic status flags are affected.

Fault Conditions:

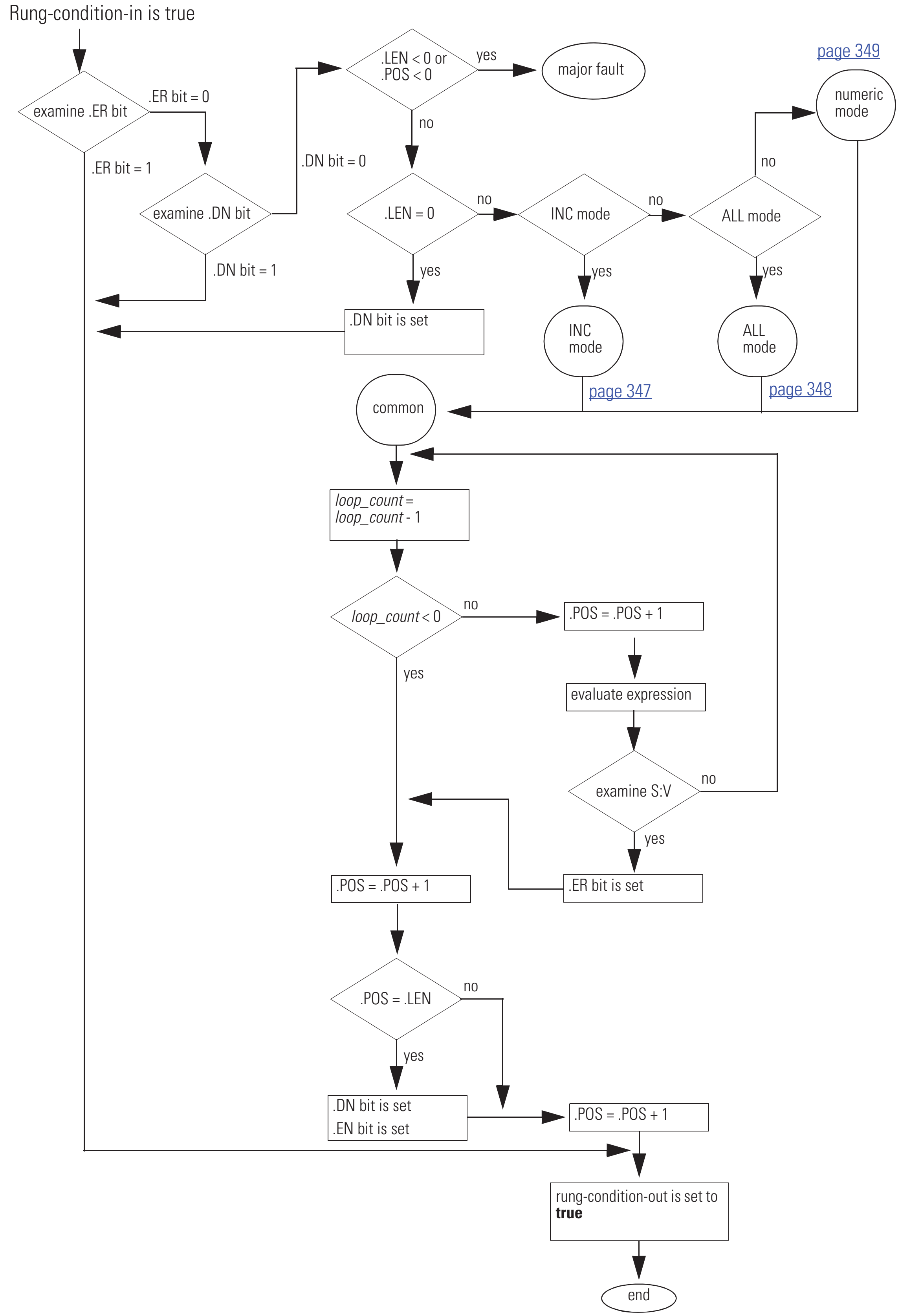
A major fault will occur if	Fault type	Fault code
Subscript is out of range	4	20
.POS < 0 or .LEN < 0	4	21

Execution:

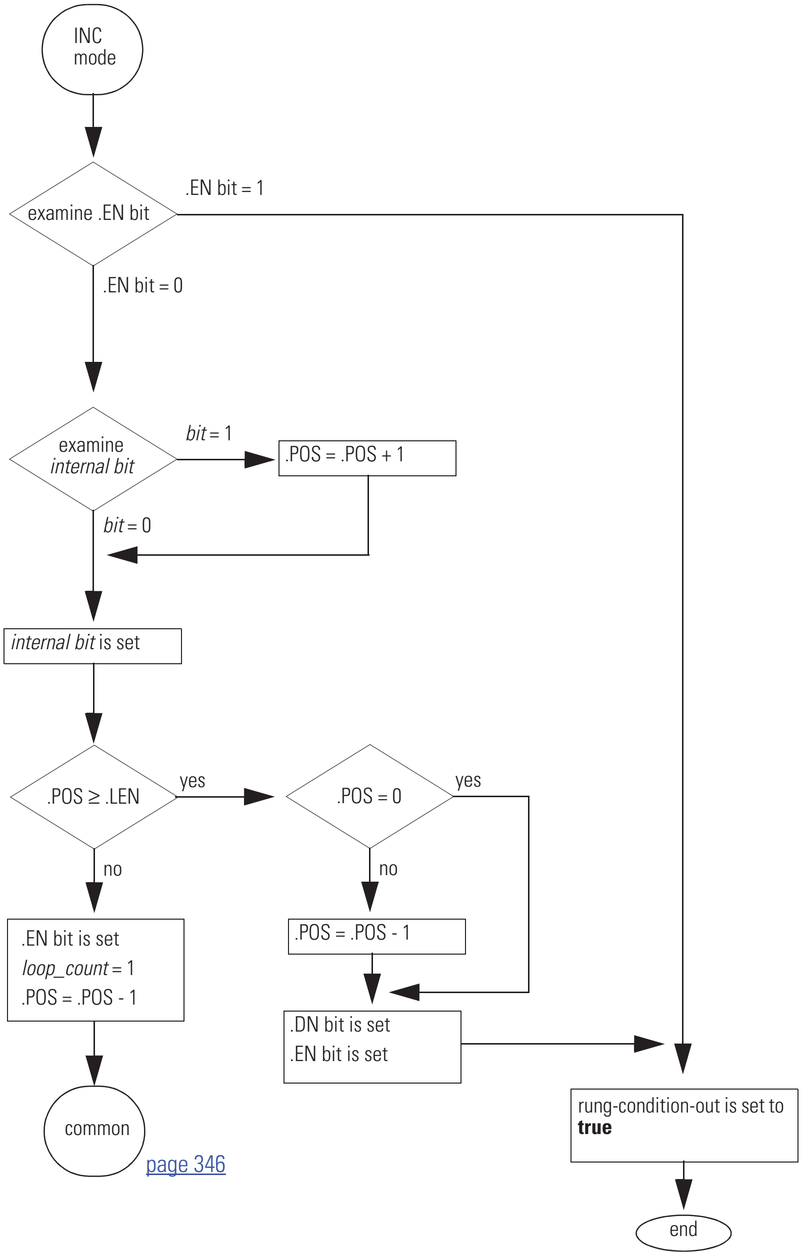
Condition	Relay Ladder Action
Prescan	The rung-condition-out is set to false.



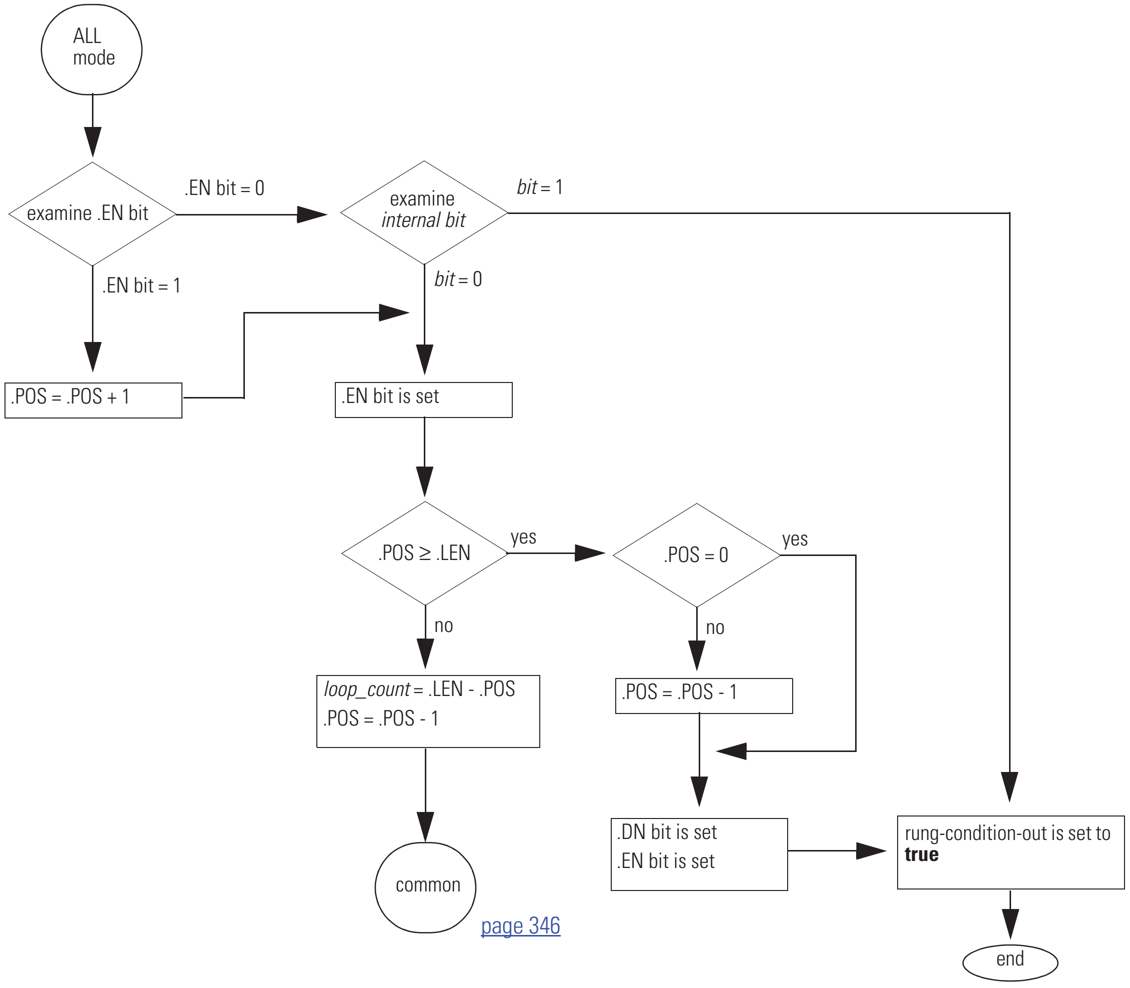
Condition	Relay Ladder Action

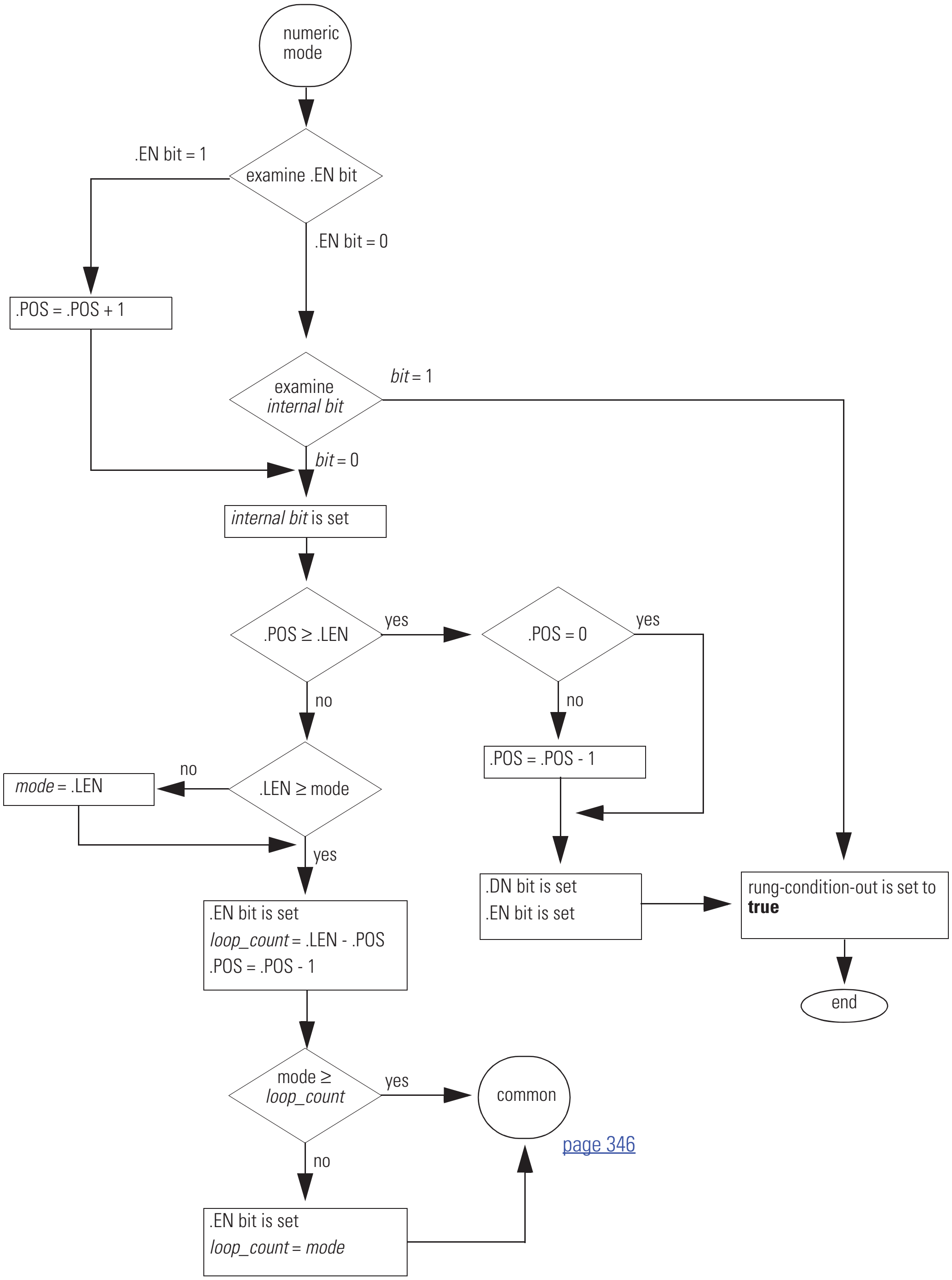


Condition	Relay Ladder Action

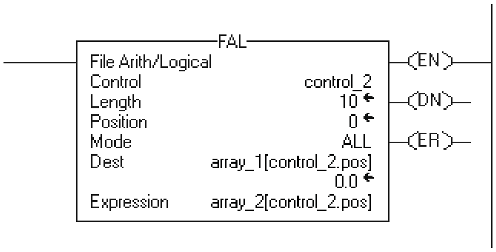


Condition	Relay Ladder Action

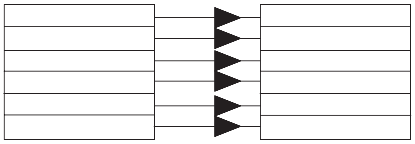


Condition	**Relay Ladder Action**
	
Postscan	The rung-condition-out is set to false.

Example 1: When enabled, the FAL instruction copies each element of *array_2* into the same position within *array_1*.



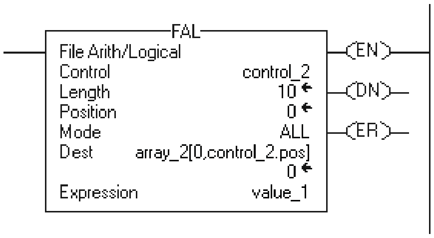
Array-to-Array Copy



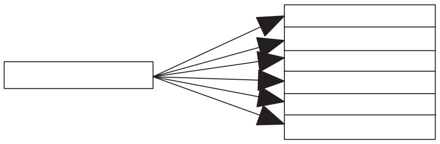
Expression: *array_2[control_2.pos]*

Destination: *array_1[control_2.pos]*

Example 2: When enabled, the FAL instruction copies *value_1* into the first 10 positions of the second dimension of *array_2*.



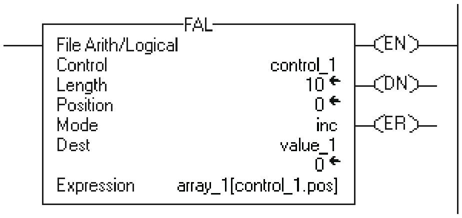
Element-to-Array Copy



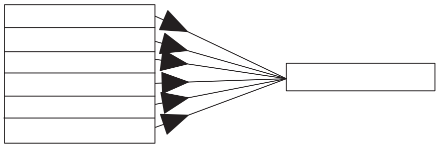
Expression: *value_1*

Destination: *array_2[0,control_2.pos]*

Example 3: Each time the FAL instruction is enabled, it copies the current value of *array_1* to *value_1*. The FAL instruction uses incremental mode, so only one array value is copied each time the instruction is enabled. The next time the instruction is enabled, the instruction overwrites *value_1* with the next value in *array_1*.



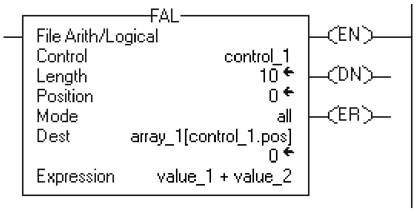
Array-to-Element Copy



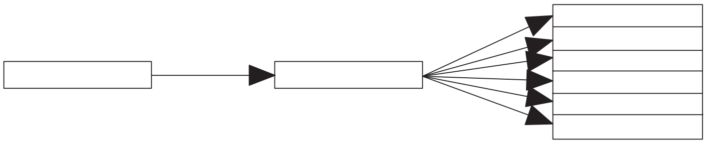
Expression: *array_1[control_1.pos]*

Destination: *value_1*

Example 4: When enabled, the FAL instruction adds *value_1* and *value_2* and stores the result in the current position of *array_1*.



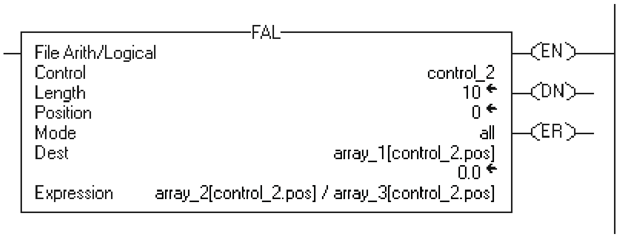
Arithmetic Operation: (Element + Element) to Array



Expression: *value_1 + value_2*

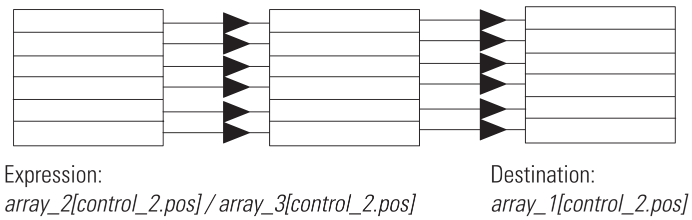
Destination: *array_1[control_1.pos]*

Example 5: When enabled, the FAL instruction divides the value in the current position of *array_2* with the value in the current position of *array_3* and stores the result in the current position of *array_1*.

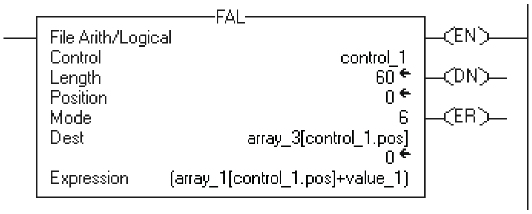


```
FAL
File Arith/Logical                                      -(EN)-
Control                                   control_2
Length                                         10 ←     -(DN)-
Position                                        0 ←
Mode                                          all       -(ER)-
Dest                          array_1[control_2.pos]
                                              0.0 ←
Expression    array_2[control_2.pos] / array_3[control_2.pos]
```

Arithmetic Operation: (Array / Array) to Array

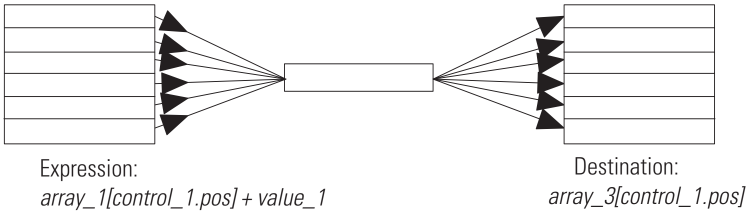


Example 6: When enabled, the FAL instruction adds the value at the current position in *array_1* to *value_1* and stores the result in the current position in *array_3*. The instruction must execute 10 times for the entire *array_1* and *array_3* to be manipulated.

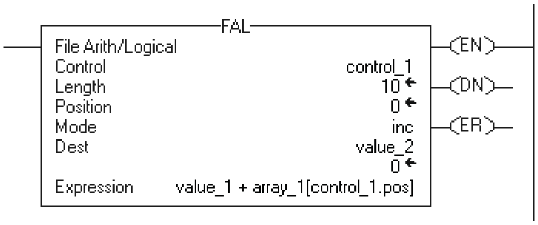


```
FAL
File Arith/Logical                          -(EN)-
Control                       control_1
Length                             60 ←     -(DN)-
Position                            0 ←
Mode                                6       -(ER)-
Dest              array_3[control_1.pos]
                                    0 ←
Expression    (array_1[control_1.pos]+value_1)
```

Arithmetic Operation: (Array + eEement) to Array

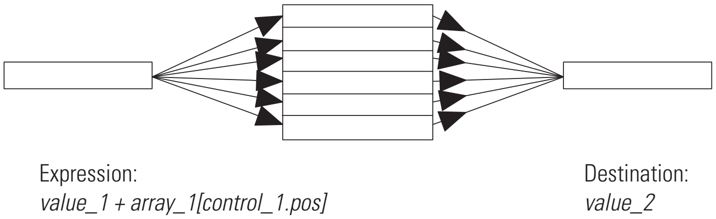


Example 7: Each time the FAL instruction is enabled, it adds *value_1* to the current value of *array_1* and stores the result in *value_2*. The FAL instruction uses incremental mode, so only one array value is added to *value_1* each time the instruction is enabled. The next time the instruction is enabled, the instruction overwrites *value_2*.

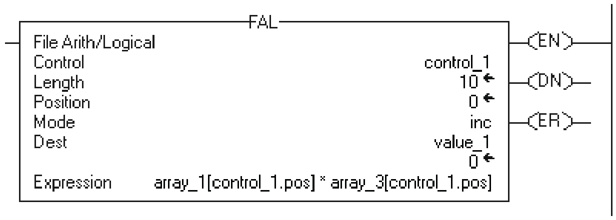


```
FAL
File Arith/Logical                          -(EN)-
Control                       control_1
Length                             10 ←     -(DN)-
Position                            0 ←
Mode                              inc       -(ER)-
Dest                          value_2
                                    0 ←
Expression    value_1 + array_1[control_1.pos]
```

Arithmetic Operation: (Element + Array) to Element

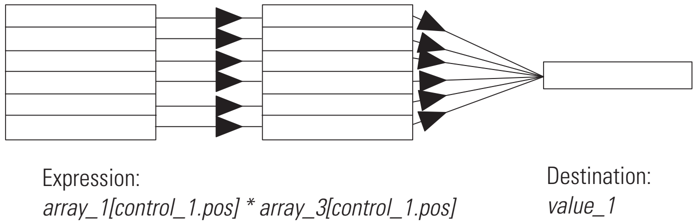


Example 8: When enabled, the FAL instruction multiplies the current value of *array_1* by the current value of *array_3* and stores the result in *value_1*. The FAL instruction uses incremental mode, so only one pair of array values is multiplied each time the instruction is enabled. The next time the instruction is enabled, the instruction overwrites *value_1*.



FAL
File Arith/Logical
Control control_1
Length 10
Position 0
Mode inc
Dest value_1
0
Expression array_1[control_1.pos] * array_3[control_1.pos]

Arithmetic Operation: (Array ∗ Array) to Element



FAL Expressions

You program expressions in FAL instructions the same as expressions in CPT instructions. Use the following sections for information on valid operators, format, and order of operation, which are common to both instructions.

Valid Operators

Operator	Description	Optimal
+	Add	DINT, REAL
-	Subtract/negate	DINT, REAL
*	Multiply	DINT, REAL
/	Divide	DINT, REAL
**	Exponent (x to y)	DINT, REAL
ABS	Absolute value	DINT, REAL
ACS	Arc cosine	REAL
AND	Bitwise AND	DINT
ASN	Arc sine	REAL
ATN	Arc tangent	REAL
COS	Cosine	REAL
DEG	Radians to degrees	DINT, REAL
FRD	BCD to integer	DINT
LN	Natural log	REAL

Operator	Description	Optimal
LOG	Log base 10	REAL
MOD	Modulo-divide	DINT, REAL
NOT	Bitwise complement	DINT
OR	Bitwise OR	DINT
RAD	Degrees to radians	DINT, REAL
SIN	Sine	REAL
SQR	Square root	DINT, REAL
TAN	Tangent	REAL
TOD	Integer to BCD	DINT
TRN	Truncate	DINT, REAL
XOR	Bitwise exclusive OR	DINT

Format Expressions

For each operator that you use in an expression, you have to provide one or two operands (tags or immediate values). Use the following table to format operators and operands within an expression.

For operators that operate on	Use this format	Examples
One operand	Operator(operand)	ABS(*tag_a*)
Two operands	Operand_a operator operand_b	• *tag_b* + 5 • *tag_c* AND *tag_d* • (*tag_e* ** 2) MOD (*tag_f* / *tag_g*)

Determine the Order of Operation

The operations you write into the expression are performed by the instruction in a prescribed order, not necessarily the order you write them. You can override the order of operation by grouping terms within parentheses, forcing the instruction to perform an operation within the parentheses ahead of other operations.

Operations of equal order are performed from left to right.

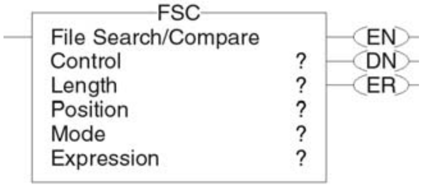
Order	Operation
1.	( )
2.	ABS, ACS, ASN, ATN, COS, DEG, FRD, LN, LOG, RAD, SIN, SQR, TAN, TOD, TRN
3.	**
4.	–(negate), NOT
5.	*, /, MOD
6.	–(subtract), +
7.	AND
8.	XOR
9.	OR

File Search and Compare (FSC)

The FSC instruction compares values in an array, element by element.

Operands:

Relay Ladder



Operand	Type	Format	Description
Control	CONTROL	Tag	Control structure for the operation
Length	DINT	Immediate	Number of elements in the array to be manipulated
Position	DINT	Immediate	Offset into array Initial value is typically 0

CONTROL Structure

Mnemonic	Data Type	Description
.EN	BOOL	The enable bit indicates that the FSC instruction is enabled.
.DN	BOOL	The done bit is set when the instruction has operated on the last element (.POS = .LEN).
.ER	BOOL	The error bit is not modified.
.IN	BOOL	The inhibit bit indicates that the FSC instruction detected a true comparison. You must clear this bit to continue the search operation.
.FD	BOOL	The found bit indicates that the FSC instruction detected a true comparison.
.LEN	DINT	The length specifies the number of elements in the array on which the instruction operates.
.POS	DINT	The position contains the position of the current element that the instruction is accessing.

Description:

When the FSC instruction is enabled and the comparison is true, the instruction sets the .FD bit and the .POS bit reflects the array position where the instruction found the true comparison. The instruction sets the .IN bit to prevent further searching.

Arithmetic Status Flags:

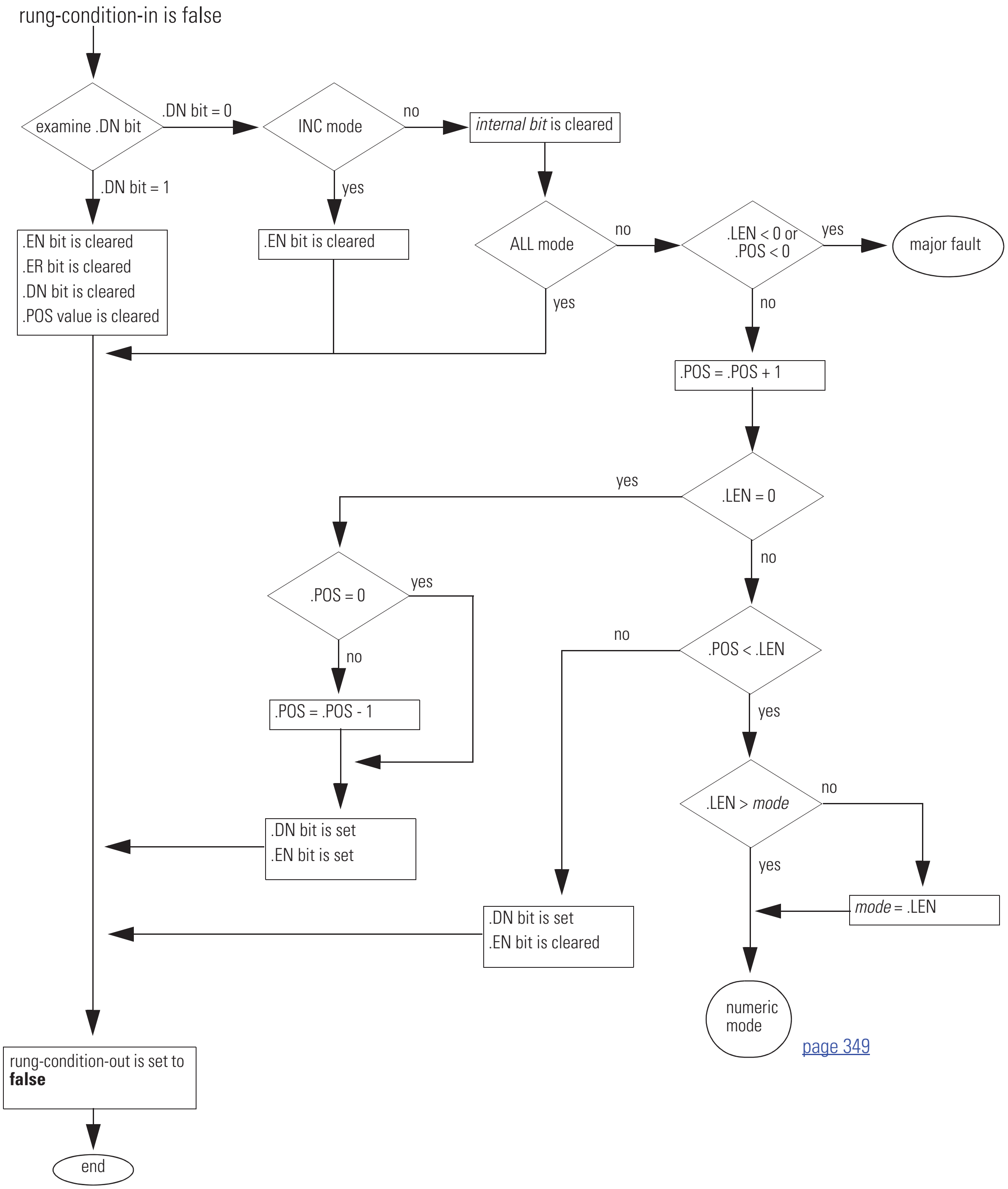
Arithmetic status flags are affected.

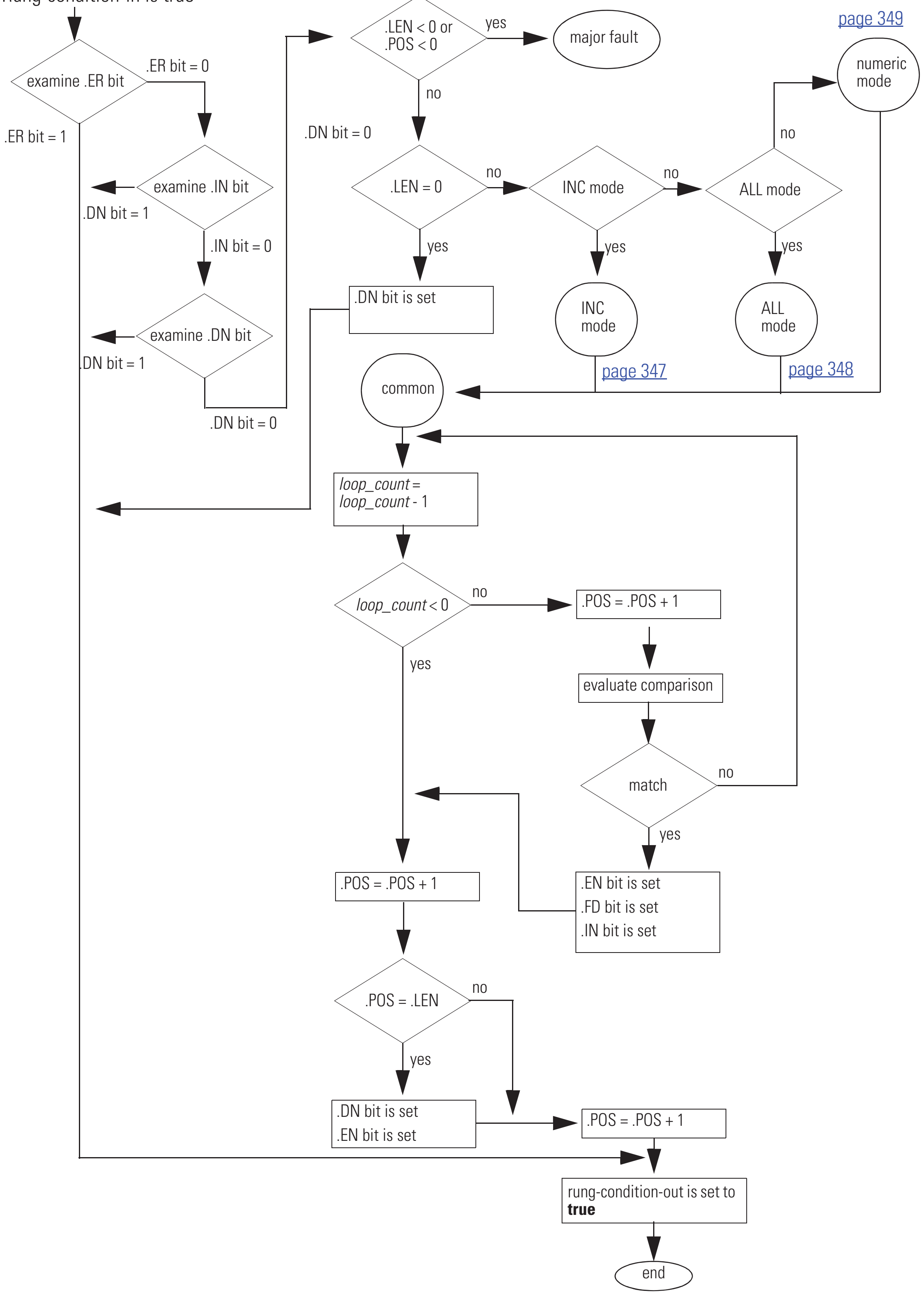
Fault Conditions:

A Major Fault Will Occur If	Fault Type	Fault Code
.POS < 0 or .LEN < 0	4	21

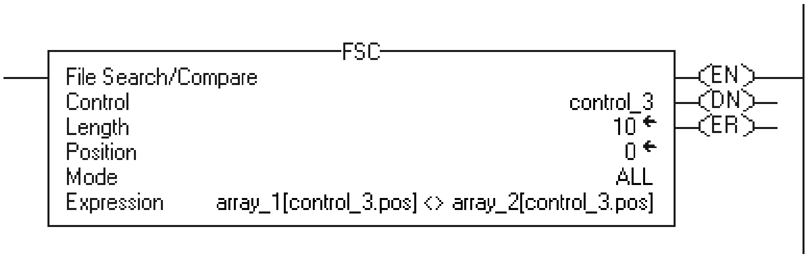
Execution:

Condition	Relay Ladder Action
Prescan	The rung-condition-out is set to false.



Condition	Relay Ladder Action
Rung-condition-in is true	
Postscan	The rung-condition-out is set to false.

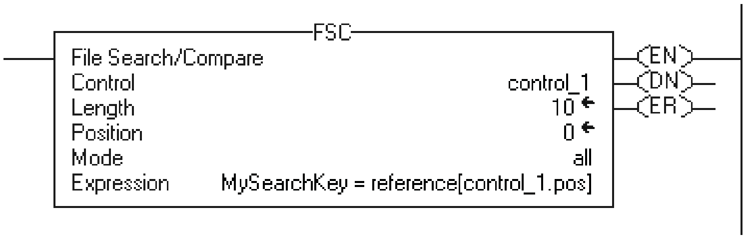
Example 1: Search for a match between two arrays. When enabled, the FSC instruction compares each of the first 10 elements in *array_1* to the corresponding elements in *array_2*.



File Search/Compare
Control control_3
Length 10
Position 0
Mode ALL
Expression array_1[control_3.pos] <> array_2[control_3.pos]

array_1	*array_2*	*control_3.pos*	
00000000000000000000000000000000	00000000000000000000000000000000	0	
00000000000000000000000000000000	00000000000000000000000000000000	1	
00000000000000000000000000000000	00000000000000000000000000000000	2	
00000000000000000000000000000000	00000000000000000000000000000000	3	
00000000000000001111111111111111	11111111111111110000000000000000	4	→ The FSC instruction finds that these elements are not equal. The instruction sets the .FD and .IN bits. The .POS value (4) indicates the position of the elements that are not equal. To continue comparing the rest of the array, clear the .IN bit.
11111111111111111111111111111111	11111111111111111111111111111111	5	
11111111111111111111111111111111	11111111111111111111111111111111	6	
11111111111111111111111111111111	11111111111111111111111111111111	7	
11111111111111111111111111111111	11111111111111111111111111111111	8	
11111111111111111111111111111111	11111111111111111111111111111111	9	

Example 2: Search for a match in an array. When enabled, the FSC instruction compares the *MySearchKey* to 10 elements in *array_1*.

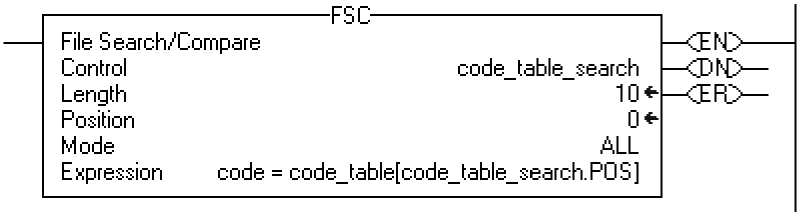


File Search/Compare
Control control_1
Length 10
Position 0
Mode all
Expression MySearchKey = reference[control_1.pos]

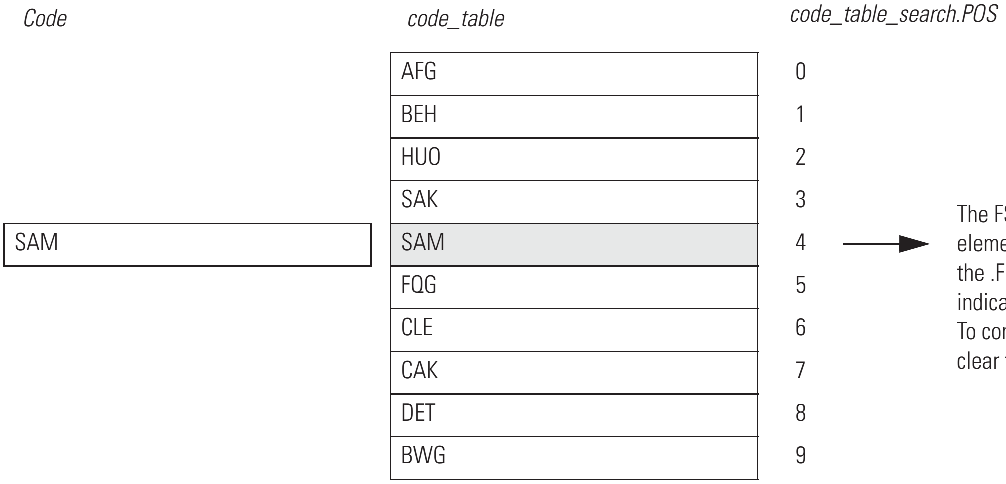
MySearchKey	*reference*	*control_3.pos*
	00000000000000000000000000000000	0
	00000000000000000000000000000000	1
	00000000000000000000000000000000	2
	00000000000000000000000000000000	3
11111111111111110000000000000000	11111111111111110000000000000000	4
	11111111111111111111111111111111	5
	11111111111111111111111111111111	6
	11111111111111111111111111111111	7
	11111111111111111111111111111111	8
	11111111111111111111111111111111	9

The FSC instruction finds that this array element equals *MySearchKey*. The instruction sets the .FD and .IN bits. The .POS (4) value indicates the position of the equal element. To continue comparing the rest of the array, clear the .IN bit.

Example 3: Search for a string in an array of strings. When enabled, the FSC instruction compares the characters in *code* to 10 elements in *code_table*.



FSC
File Search/Compare
Control code_table_search
Length 10
Position 0
Mode ALL
Expression code = code_table[code_table_search.POS]



The FSC instruction finds that this array element equals *code*. The instruction sets the .FD and .IN bits. The .POS (4) value indicates the position of the equal element. To continue comparing the rest of the array, clear the .IN bit.

FSC Expressions

You program expressions in FSC instructions the same as expressions in CMP instructions. Use the following sections for information on valid operators, format, and order of operation, which are common to both instructions.

Valid Operators

Operator	Description	Optimal
+	Add	DINT, REAL
-	Subtract/negate	DINT, REAL
*	Multiply	DINT, REAL
/	Divide	DINT, REAL
=	Equal	DINT, REAL
<	Less than	DINT, REAL
<=	Less than or equal	DINT, REAL
>	Greater than	DINT, REAL
>=	Greater than or equal	DINT, REAL
<>	Not equal	DINT, REAL
**	Exponent (x to y)	DINT, REAL
ABS	Absolute value	DINT, REAL
ACS	Arc cosine	REAL
AND	Bitwise AND	DINT
ASN	Arc sine	REAL
ATN	Arc tangent	REAL
COS	Cosine	REAL

Operator	Description	Optimal
DEG	Radians to degrees	DINT, REAL
FRD	BCD to integer	DINT
LN	Natural log	REAL
LOG	Log base 10	REAL
MOD	Modulo-divide	DINT, REAL
NOT	Bbitwise complement	DINT
OR	Bitwise OR	DINT
RAD	Degrees to radians	DINT, REAL
SIN	Sine	REAL
SQR	Square root	DINT, REAL
TAN	Tangent	REAL
TOD	Integer to BCD	DINT
TRN	Truncate	DINT, REAL
XOR	Bitwise exclusive OR	DINT

Format Expressions

For each operator that you use in an expression, you have to provide one or two operands (tags or immediate values). Use the following table to format operators and operands within an expression.

For operators that operate on	Use this format	Examples
One operand	Operator(operand)	ABS(*tag_a*)
Two operands	Operand_a operator operand_b	• *tag_b* + 5 • *tag_c* AND *tag_d* • (*tag_e* ** 2) MOD (*tag_f* / *tag_g*)

Determine the Order of Operation

The operations you write into the expression are performed by the instruction in a prescribed order, not necessarily the order you write them. You can override the order of operation by grouping terms within parentheses, forcing the instruction to perform an operation within the parentheses ahead of other operations.

Operations of equal order are performed from left to right.

Order	Operation
1.	( )
2.	ABS, ACS, ASN, ATN, COS, DEG, FRD, LN, LOG, RAD, SIN, SQR, TAN, TOD, TRN
3.	**
4.	–(negate), NOT
5.	*, /, MOD
6.	<, <=, >, >=, =
7.	–(subtract), +
8.	AND
9.	XOR
10.	OR

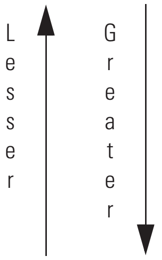
Use Strings in an Expression

To use strings of ASCII characters in an expression, follow these guidelines:

- An expression lets you compare two string tags.
- You **cannot** enter ASCII characters directly into the expression.
- Only the following operators are permitted.

Operator	Description
=	Equal
<	Less than
<=	Less than or equal
>	Greater than
>=	Greater than or equal
<>	Not equal

- Strings are equal if their characters match.
- ASCII characters are case sensitive. Upper case "A" ($41) is *not* equal to lower case "a" ($61).
- The hexadecimal values of the characters determine if one string is less than or greater than another string. For the hex code of a character, see the back cover of this manual.
- When the two strings are sorted as in a telephone directory, the order of the strings determines which one is greater.



ASCII Characters	Hex Codes
1ab	$31$61$62
1b	$31$62
A	$41
AB	$41$42
B	$42
a	$61
ab	$61$62

AB < B

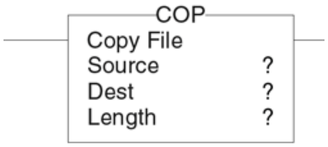
a > B

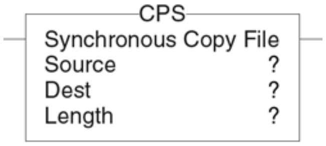
Copy File (COP) Synchronous Copy File (CPS)

The COP and CPS instructions copy the value(s) in the Source to the Destination. The Source remains unchanged.

Operands:

Relay Ladder





Operand	Type	Format	Description
Source	**SINT** **INT** **DINT** **REAL** **string** structure	Tag	Initial element to copy **Important:** the Source and Destination operands should be the same data type, or unexpected results may occur
Destination	**SINT** **INT** **DINT** **REAL** **string** structure	Tag	Initial element to be overwritten by the Source **Important:** the Source and Destination operands should be the same data type, or unexpected results may occur
Length	DINT	Immediate Tag	Number of Destination elements to copy

Structured Text

```
COP(Source,Dest,Length);
CPS(Source,Dest,Length);
```

The operands are the same as those for the relay ladder COP and CPS instructions.

Description: During execution of the COP and CPS instructions, other controller actions may try to interrupt the copy operation and change the source or destination data.

If the source or destination is	And you want to	Then select	Notes
• Produced tag • Consumed tag • I/O data • Data that another task can overwrite	Prevent the data from changing during the copy operation	CPS	• Tasks that attempt to interrupt a CPS instruction are delayed until the instruction is done. • To estimate the execution time of the CPS instruction, see *ControlLogix System User Manual*, publication 1756-UM001.
	Allow the data to change during the copy operation	COP	
None of the above	→	COP	

The number of bytes copied is:

Byte Count = Length ∗ (number of bytes in the Destination data type)

ATTENTION ⚠

If the byte count is greater than the length of the Source, unpredictable data is copied for the remaining elements.

IMPORTANT

You **must** test and confirm that the instruction doesn't change data that you don't want it to change.

The COP and CPS instructions operate on contiguous memory. They do a straight byte-to-byte memory copy. In some cases, they write past the array into other members of the tag. This happens if the length is too big and the tag is a user-defined data type.

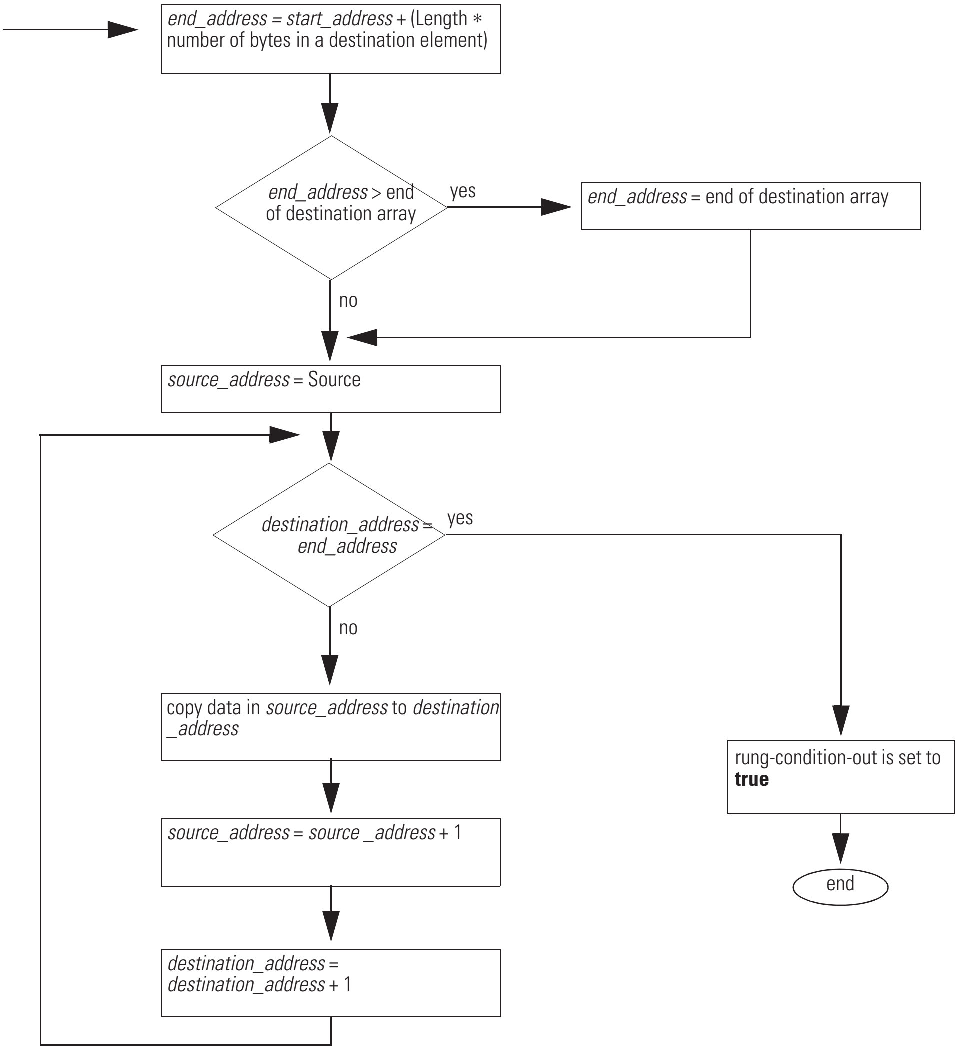
If the tag is	Then
User-defined data type	If the Length is too big, the instruction writes past the end of the array into other members of the tag. It stops at the end of the tag. No major fault is generated.
NOT user-defined data type	If the Length is too big, the instruction stops at the end of the array. No major fault is generated.

The Length is too big if it is more than the total number of elements in the Destination array.

Arithmetic Status Flags: Not affected

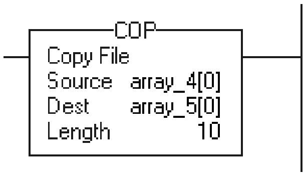
Fault Conditions: None

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution		
Postscan	The rung-condition-out is set to false.	No action taken.

Example 1: Both *array_4* and *array_5* are the same data type. When enabled, the COP instruction copies the first 10 elements of *array_4* into the first 10 elements of *array_5*.

Relay Ladder

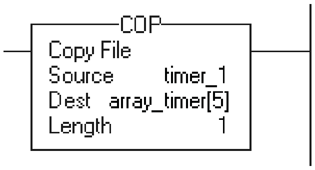

COP
Copy File
Source array_4[0]
Dest array_5[0]
Length 10

Structured Text

```
COP(array_4[0],array_5[0],10);
```

Example 2: When enabled, the COP instruction copies the structure *timer_1* into element 5 of *array_timer*. The instruction copies only one structure to one array element.

Relay Ladder


COP
Copy File
Source timer_1
Dest array_timer[5]
Length 1

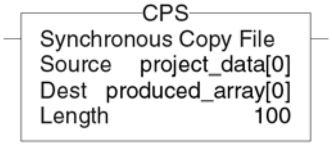
Structured Text

```
COP(timer_1,array_timer[5],1);
```

Example 3: The *project_data* array (100 elements) stores a variety of values that change at different times in the application. To send a complete image of *project_data* at one instance in time to another controller, the CPS instruction copies *project_data* to *produced_array*.

- While the CPS instruction copies the data, no I/O updates or other tasks can change the data.
- The *produced_array* tag produces the data on a ControlNet network for consumption by other controllers.
- To use the same image of data (that is, a synchronized copy of the data), the consuming controller (s) uses a CPS instruction to copy the data from the consumed tag to another tag for use in the application.

Relay Ladder



CPS
Synchronous Copy File
Source project_data[0]
Dest produced_array[0]
Length 100

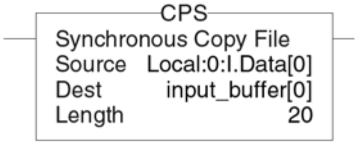
Structured Text

```
CPS(project_data[0],produced_array[0],100);
```

Example 4: *Local:0:I.Data* stores the input data for the DeviceNet network that is connected to the 1756-DNB module in slot 0. To synchronize the inputs with the application, the CPS instruction copies the input data to *input_buffer*.

- While the CPS instruction copies the data, no I/O updates can change the data.
- As the application executes, it uses for its inputs the input data in *input_buffer*.

Relay Ladder

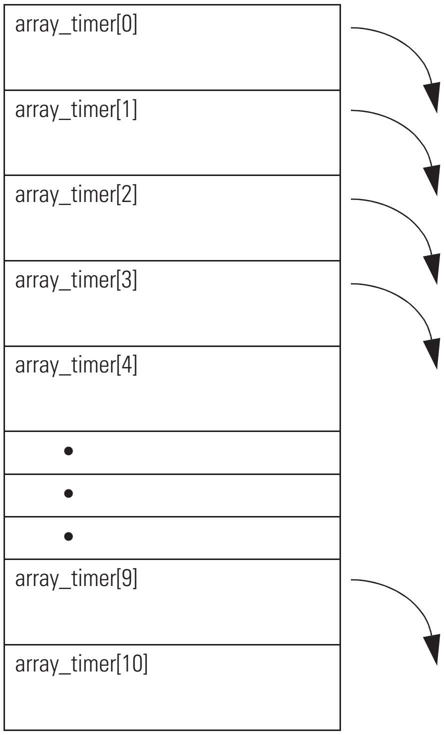


CPS
Synchronous Copy File
Source Local:0:I.Data[0]
Dest input_buffer[0]
Length 20

Structured Text

```
CPS(Local:0:I.Data[0],input_buffer[0],20);
```

Example 5: This example initializes an array of timer structures. When enabled, the MOV instructions initialize the .PRE and .ACC values of the first *array_timer* element. When enabled, the COP instruction copies a contiguous block of bytes, starting at *array_timer[0]*. The length is nine timer structures.



First the instruction copies timer[0] values to timer[1]

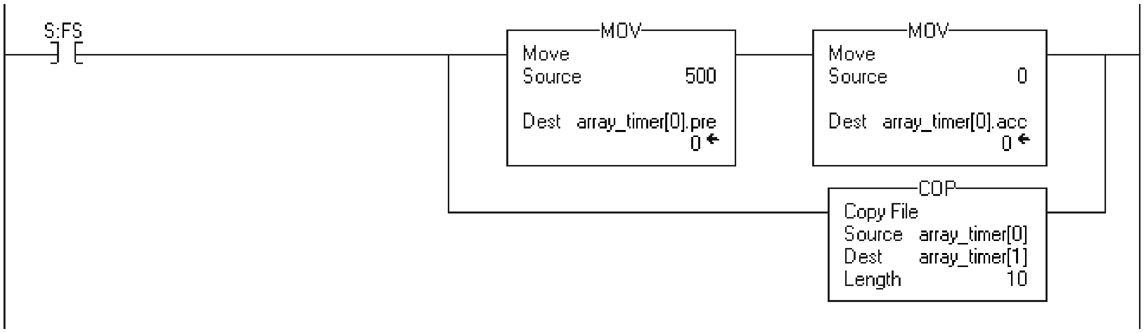
Then the instruction copies timer[1] values to timer[2]

Then the instruction copies timer[2] values to timer[3]

Then the instruction copies timer[3] values to timer[4]

Finally, the instruction copies timer[9] values to timer[10]

Relay Ladder



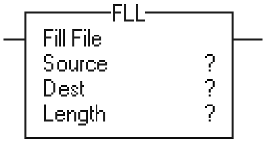
Structured Text

```
IF S:FS THEN

     array_timer[0].pre := 500;

     array_timer[0].acc := 0;

     COP(array_timer[0],array_timer[1],10);

END_IF;
```

File Fill (FLL)

The FLL instruction fills elements of an array with the Source value. The Source remains unchanged.

Operands:



Relay Ladder

Operand	Type	Format:	Description
Source	**SINT** **INT** **DINT** **REAL**	Immediate Tag	Element to copy **Important:** the Source and Destination operands should be the same data type, or unexpected results may occur
Destination	**SINT** **INT** **DINT** **REAL** structure	Tag	Initial element to be overwritten by the Source **Important:** the Source and Destination operands should be the same data type, or unexpected results may occur The preferred way to initialize a structure is to use the COP instruction.
Length	DINT	Immediate	Number of elements to fill

Structured Text

Structured text does not have an FLL instruction, but you can achieve the same results bu using a SIZE instruction and a FOR...DO or other loop construct.

```
SIZE(destination,0,length);
FOR position = 0 TO length-1 DO
    destination[position] := source;
END_FOR;
```

See Structured Text Programming for information on the syntax of constructs within structured text.

Description: The number of bytes filled is:

Byte count = Length * (number of bytes in the Destination data type)

IMPORTANT

You **must** test and confirm that the instruction doesn't change data that you don't want it to change.

The FLL instruction operates on contiguous data memory. In some cases, the instruction writes past the array into other members of the tag. This happens if the length is too big and the tag is a user-defined data type.

If the tag is	Then
User-defined data type	If the Length is too big, the instruction writes past the end of the array into other members of the tag. It stops at the end of the tag. No major fault is generated.
NOT user-defined data type	If the Length is too big, the instruction stops at the end of the array. No major fault is generated.

The Length is too big if it is more than the total number of elements in the Destination array.

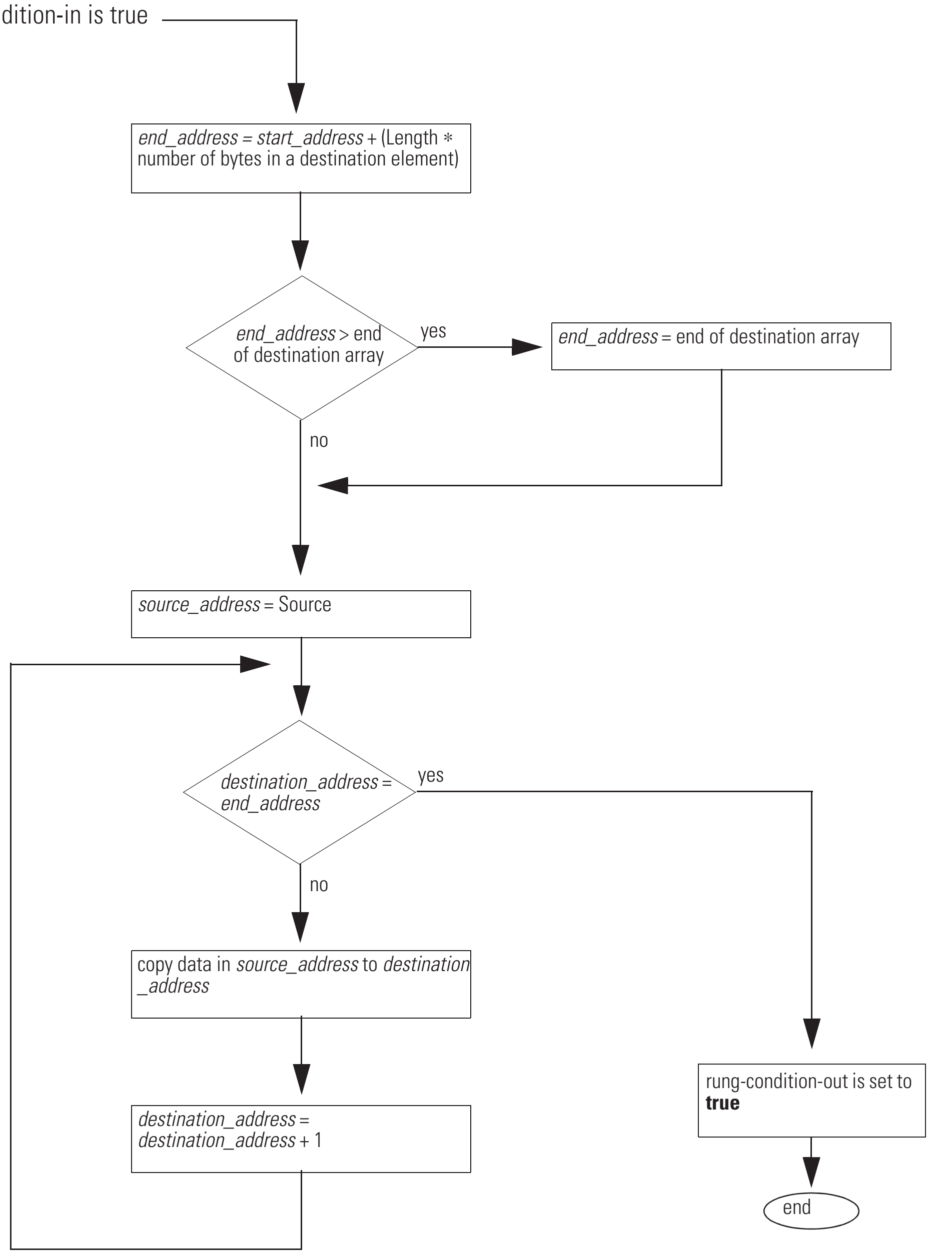
For best results, the Source and Destination should be the same type. If you want to fill a structure, use the COP instruction (see example 3 on page 359). If you mix data types for the Source and Destination, the Destination elements are filled with converted Source values.

If The Source Is	And The Destination Is	The Source Is Converted To
SINT, INT, DINT, or REAL	SINT	SINT
SINT, INT, DINT, or REAL	INT	INT
SINT, INT, DINT, or REAL	DINT	DINT
SINT, INT, DINT, or REAL	REAL	REAL
SINT	Structure	SINT (not converted)
INT	Structure	INT (not converted)
DINT	Structure	DINT (not converted)
REAL	Structure	REAL (not converted)

Arithmetic Status Flags: Not affected

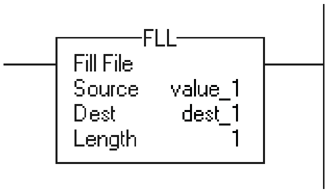
Fault Conditions: None

Execution:

Condition	Relay Ladder Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	
Postscan	The rung-condition-out is set to false.

Example: The FLL instruction copies the value in *value_1* into *dest_1*

Relay Ladder



FLL
Fill File
Source value_1
Dest dest_1
Length 1

Source (*value_1*) Data Type	Source (*value_1*) Value	Destination (*dest_1*) Data Type	Destination (*dest_1*) Value After FLL
SINT	16#80 (-128)	DINT	16#FFFF FF80 (-128)
DINT	16#1234 5678	SINT	16#78
SINT	16#01	REAL	1.0
REAL	2.0	INT	16#0002
SINT	16#01	TIMER	16#0101 0101 16#0101 0101 16#0101 0101
INT	16#0001	TIMER	16#0001 0001 16#0001 0001 16#0001 0001
DINT	16#0000 0001	TIMER	16#0000 0001 16#0000 0001 16#0000 0001

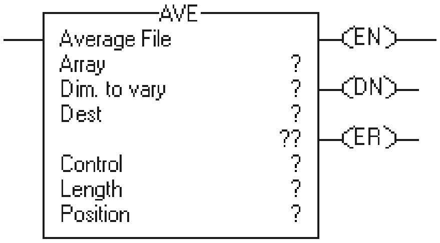
Structured Text

```
dest_1 := value_1;
```

File Average (AVE)

The AVE instruction calculates the average of a set of values.

Operands:



Relay Ladder

Operand	Type	Format	Description
Array	SINT INT DINT **REAL**	Array tag	Find the average of the values in this array Specify the first element of the group of elements to average **Do not** use CONTROL.POS in the subscript
Dimension to vary	DINT	Immediate (0, 1, 2)	Which dimension to use Depending on the number of dimensions, the order is: array[dim_0,dim_1,dim_2] array[dim_0,dim_1] array[dim_0]
Destination	SINT INT DINT **REAL**	Tag	Result of the operation
Control	CONTROL	Tag	Control structure for the operation
Length	DINT	Immediate	Number of elements of the array to average
Position	DINT	Immediate	Current element in the array Initial value is typically 0

Structured Text

Structured text does not have an AVE instruction, but you can achieve the same results by using a SIZE instruction and a FOR...DO or other loop construct.

```
SIZE(array,0,length);
sum := 0;
FOR position = 0 TO length-1 DO
    sum := sum + array[position];
END_FOR;
```

```
destination := sum / length;
```

See Structured Text Programming for information on the syntax of constructs within structured text.

CONTROL Structure

Mnemonic	Data Type	Description
.EN	BOOL	The enable bit indicates that the AVE instruction is enabled.
.DN	BOOL	The done bit is set when the instruction has operated on the last element in the array (.POS = .LEN).
.ER	BOOL	The error bit is set if the instruction generates an overflow. The instruction stops executing until the program clears the .ER bit. The position of the element that caused the overflow is stored in the .POS value.
.LEN	DINT	The length specifies the number of elements in the array on which the instruction operates.
.POS	DINT	The position contains the position of the current element that the instruction is accessing.

Description:

The AVE instruction calculates the average of a set of values.

IMPORTANT Make sure the Length does not cause the instruction to exceed the specified Dimension to vary. If this happens, the Destination will be incorrect.

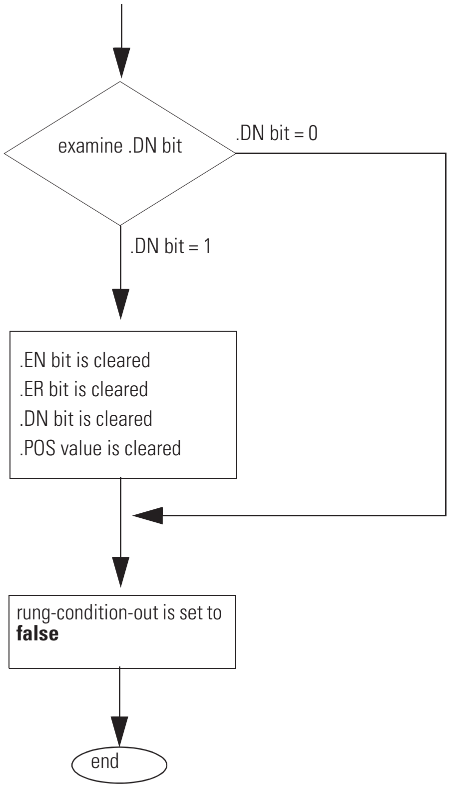
Arithmetic Status Flags:

Arithmetic status flags are affected.

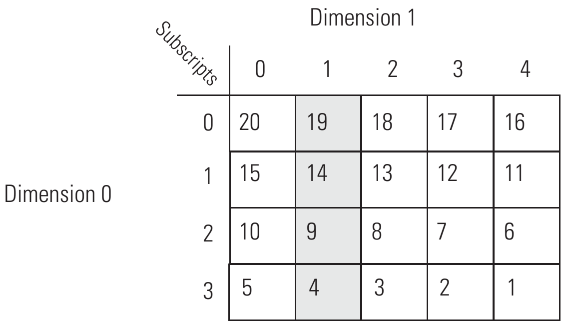
Fault Conditions:

A Major Fault Will Occur If	Fault Type	Fault Code
.POS < 0 or .LEN < 0	4	21
Dimension to vary does not exist for the specified array	4	20

Execution:

Condition	Relay Ladder Action
Prescan	The .EN bit is cleared. The .DN bit is cleared. The .ER bit is cleared. The rung-condition-out is set to false.
Rung-condition-in is false	
Rung-condition-in is true	The AVE instruction calculates the average by adding all the specified elements in the array and dividing by the number of elements. Internally, the instruction uses a FAL instruction to calculate the average: Expression = average calculation Mode = ALL For details on how the FAL instruction executes, see page 345.
Postscan	The rung-condition-out is set to false.

Example 1: Average *array_dint*, which is DINT[4,5].

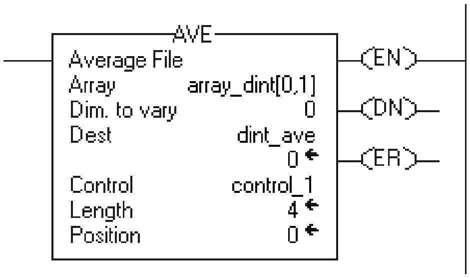


	0	1	2	3	4
0	20	19	18	17	16
1	15	14	13	12	11
2	10	9	8	7	6
3	5	4	3	2	1

$$AVE = \frac{19 + 14 + 9 + 4}{4} = \frac{46}{4} = 11.5$$

dint_ave = 12

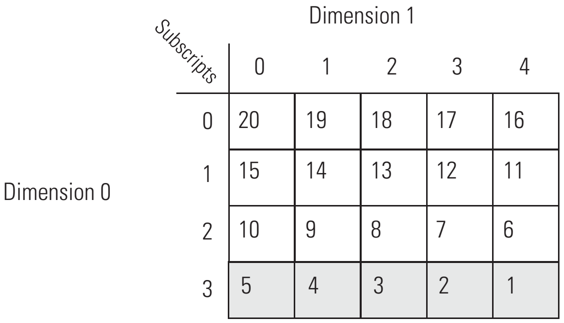
Relay Ladder



Structured Text

```
SIZE(array_dint,0,length);
sum := 0;
FOR position = 0 TO (length-1) DO
    sum := sum + array_dint[position];
END_FOR;
dint_ave := sum / length;
```

Example 2: Average *array_dint*, which is DINT[4,5].


Dimension 1

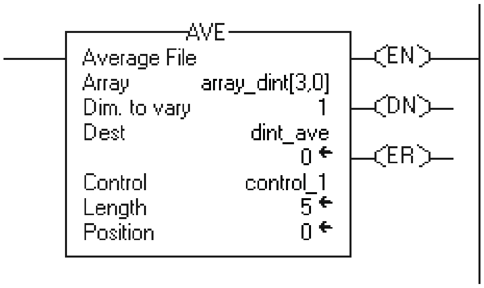
Dimension 0

Subscripts	0	1	2	3	4
0	20	19	18	17	16
1	15	14	13	12	11
2	10	9	8	7	6
3	5	4	3	2	1

$$AVE = \frac{5+4+3+2+1}{5} = \frac{15}{5} = 3$$

$dint\_ave = 3$

Relay Ladder


AVE
Average File
Array array_dint[3,0]
Dim. to vary 1
Dest dint_ave
0
Control control_1
Length 5
Position 0

Structured Text

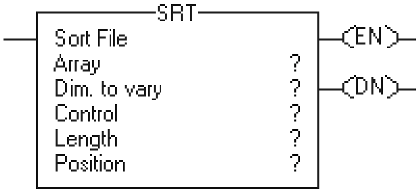
```
SIZE(array_dint,1,length);
sum := 0;
FOR position = 0 TO (length-1) DO
    sum := sum + array_dint[position];
END_FOR;
dint_ave := sum / length;
```

File Sort (SRT)

The SRT instruction sorts a set of values in one dimension (Dim to vary) of the Array into ascending order.

Operands:

Relay Ladder



Operand	Type	Format	Description
Array	**SINT** **INT** **DINT** **REAL**	Array tag	Array to sort Specify the first element of the group of elements to sort **Do not** use CONTROL.POS in the subscript
Dimension to vary	DINT	Immediate (0, 1, 2)	Which dimension to use Depending on the number of dimensions, the order is: array[dim_0,dim_1,dim_2] array[dim_0,dim_1] array[dim_0]
Control	CONTROL	Tag	Control structure for the operation
Length	DINT	Immediate	Number of elements of the array to sort
Position	DINT	Immediate	Current element in the array Initial value is typically 0

Structured Text

```
SRT(Array,Dimtovary,
    Control);
```

The operands are the same as those for the relay ladder SRT instruction. However, you specify the Length and Position values by accessing the .LEN and .POS members of the CONTROL structure, rather than by including values in the operand list.

CONTROL Structure

Mnemonic	Data Type	Description
.EN	BOOL	The enable bit indicates that the SRT instruction is enabled.
.DN	BOOL	The done bit is set when the specified elements have been sorted.
.ER	BOOL	The error bit is set when either .LEN < 0 or .POS < 0. Either of these conditions also generates a major fault.
.LEN	DINT	The length specifies the number of elements in the array on which the instruction
.POS	DINT	The position contains the position of the current element that the instruction is accessing.

Description:

The SRT instruction sorts a set of values in one dimension (Dim to vary) of the Array into ascending order.

IMPORTANT You must test and confirm that the instruction doesn't change data that you don't want it to change.

The SRT instruction operates on contiguous memory. In some cases, the instruction changes data in other members of the tag. This happens if the length is too big and the tag is a user-defined data type.

IMPORTANT Make sure the Length does not cause the instruction to exceed the specified Dimension to vary. If this happens, unexpected results will occur.

This is a transitional instruction.

- In relay ladder, toggle the rung-condition-in from cleared to set each time the instruction should execute.
- In structured text, condition the instruction so that it only executes on a transition. See Structured Text Programming.

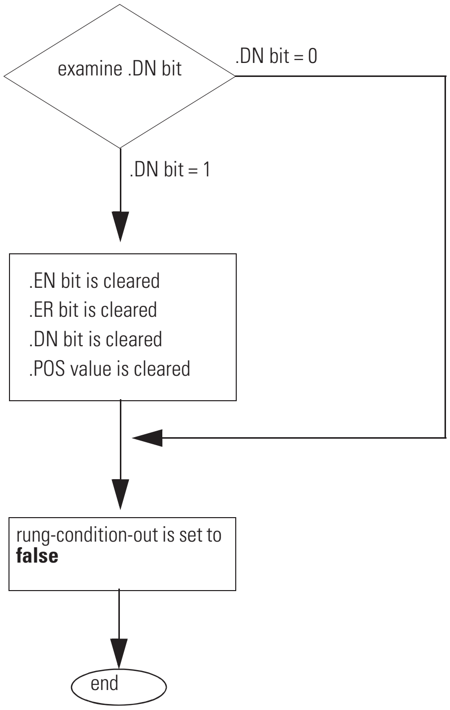
Arithmetic Status Flags:

Arithmetic status flags are affected.

Fault Conditions:

A Major Fault Will Occur If	Fault Type	Fault Code
.POS < 0 or .LEN < 0	4	21
Dimension to vary does not exist for the specified array	4	20
Instruction tries to access data outside of the array boundaries	4	20

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The .EN bit is cleared. The .DN bit is cleared. The .ER bit is cleared. The rung-condition-out is set to false.	The .EN bit is cleared. The .DN bit is cleared. The .ER bit is cleared.
Rung-condition-in is false		N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction sorts the specified elements of the array into ascending order.	The instruction sorts the specified elements of the array into ascending order.
Postscan	The rung-condition-out is set to false.	No action taken.

Example 1: Sort *int* _array, which is DINT[4,5].

Before

Dimension 1

Dimension 0

Subscripts	0	1	2	3	4
0	20	19	18	17	16
1	15	14	13	12	11
2	10	9	8	7	6
3	5	4	3	2	1

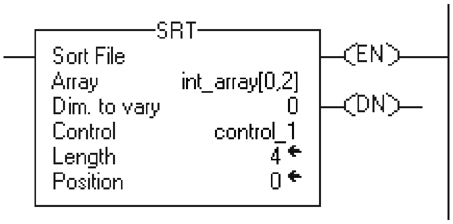
After

Dimension 1

Dimension 0

Subscripts	0	1	2	3	4
0	20	19	3	17	16
1	15	14	8	12	11
2	10	9	13	7	6
3	5	4	18	2	1

Relay Ladder



Sort File
Array int_array[0,2]
Dim. to vary 0
Control control_1
Length 4
Position 0

Structured Text

```
control_1.LEN := 4;

control_1.POS := 0;

SRT(int_array[0,2],0,control_1);
```

Example 2: Sort *int* _array, which is DINT[4,5].

Before

Dimension 1

Dimension 0

Subscripts	0	1	2	3	4
0	20	19	18	17	16
1	15	14	13	12	11
2	10	9	8	7	6
3	5	4	3	2	1

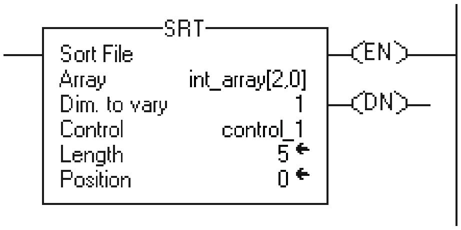
After

Dimension 1

Dimension 0

Subscripts	0	1	2	3	4
0	20	19	18	17	16
1	15	14	13	12	11
2	6	7	8	9	10
3	5	4	3	2	1

Relay Ladder



Sort File
Array int_array[2,0]
Dim. to vary 1
Control control_1
Length 5
Position 0

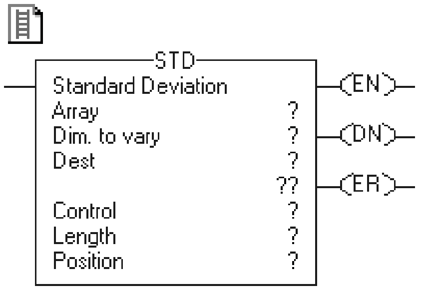
Structured Text

```
control_1.LEN := 5;

control_1.POS := 0;

SRT(int_array[2,0],1,control_1);
```

File Standard Deviation (STD)

The STD instruction calculates the standard deviation of a set of values in one dimension of the Array and stores the result in the Destination.

Operands:



Relay Ladder

Operand	Type	Format	Description
Array	SINT INT **DINT** **REAL**	Array tag	Find the standard deviation of the values in this array Specify the first element of the group of elements to use in calculating the standard deviation **Do not** use CONTROL.POS in the subscript
	A SINT or INT tag converts to a DINT value by sign-extension.		
Dimension to vary	DINT	Immediate (0, 1, 2)	Which dimension to use Depending on the number of dimensions, the order is: array[dim_0,dim_1,dim_2] array[dim_0,dim_1] array[dim_0]
Destination	REAL	Tag	Result of the operation
Control	CONTROL	Tag	Control structure for the operation
Length	DINT	Immediate	Number of elements of the array to use in calculating the standard deviation
Position	DINT	Immediate	Current element in the array Initial value is typically 0

CONTROL Structure

Mnemonic	Data Type	Description
.EN	BOOL	The enable bit indicates that the STD instruction is enabled.
.DN	BOOL	The done bit is set when the calculation is complete.
.ER	BOOL	The error bit is set when the instruction generates an overflow. The instruction stops executing until the program clears the .ER bit. The position of the element that caused the overflow is stored in the .POS value.
.LEN	DINT	The length specifies the number of elements in the array on which the instruction operates.
.POS	DINT	The position contains the position of the current element that the instruction is accessing.

Structured Text

Structured text does not have an STD instruction, but you can achieve the same results by using a SIZE instruction and a FOR...DO or other loop construct.

```
SIZE(array,0,length);
sum := 0;
FOR position = 0 TO length-1 DO
    sum := sum + array[position];
END_FOR;
average := sum / length;
sum := 0;
FOR position = 0 TO length-1 DO
    sum := sum + ((array[position] - average)**2);
END_FOR;
destination := SQRT(sum /(length-1));
```

See Structured Text Programming for information on the syntax of constructs within structured text.

Description: The standard deviation is calculated according to this formula:

$$\text{Standard Deviation} = \sqrt{\frac{\left(\sum_{i=1}^{N} [\langle X_{(start+i)} - AVE\rangle^2]\right)}{(N-1)}}$$

Where:

- start = dimension-to-vary subscript of the array operand
- x_i = variable element in the array
- N = number of specified elements in the array
- AVE = $\dfrac{\left(\sum_{i=1}^{N} x_{(start+i)}\right)}{N}$

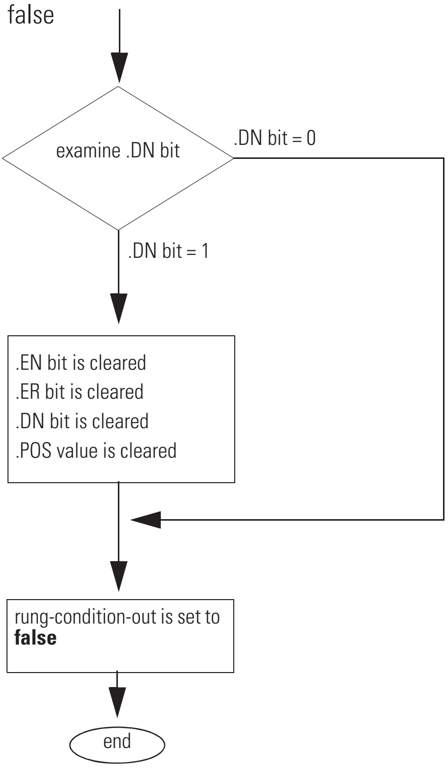
IMPORTANT Make sure the length does not cause the instruction to exceed the specified dimension to vary. If this happens, the destination will be incorrect.

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions:

A major fault will occur if	Fault type	Fault code
.POS < 0 or .LEN < 0	4	21
Dimension to vary does not exist for the specified array	4	20

Execution:

Condition	Relay Ladder Action
Prescan	The .EN bit is cleared. The .DN bit is cleared. The .ER bit is cleared. The rung-condition-out is set to false.
Rung-condition-in is false	
Rung-condition-in is true	The STD instruction calculates the standard deviation of the specified elements. Internally, the instruction uses a FAL instruction to calculate the average: Expression = standard deviation calculation Mode = ALL For details on how the FAL instruction executes, see page 345.
Postscan	The rung-condition-out is set to false.

Example 1: Calculate the standard deviation of *dint_array*, which is DINT[4,5].

Dimension 1

Dimension 0

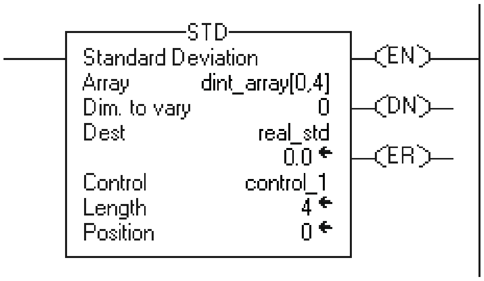
Subscripts	0	1	2	3	4
0	20	19	18	17	16
1	15	14	13	12	11
2	10	9	8	7	6
3	5	4	3	2	1

$$AVE = \frac{16 + 11 + 6 + 1}{4} = \frac{34}{4} = 8.5$$

$$STD = \sqrt{\frac{\langle 16 - 8.5\rangle^2 + \langle 11 - 8.5\rangle^2 + \langle 6 - 8.5\rangle^2 + \langle 1 - 8.5\rangle^2}{\langle 4 - 1\rangle}} = 6.454972$$

real_std = 6.454972

Relay Ladder



Structured Text

```
SIZE(dint_array,0,length);
sum := 0;
FOR position = 0 TO (length-1) DO
    sum := sum + dint_array[position];
END_FOR;
average := sum / length;
sum := 0;
FOR position = 0 TO (length-1) DO
    sum := sum + ((dint_array[position] - average)**2);
END_FOR;
real_std := SQRT(sum / (length-1));
```

Example 2: Calculate the standard deviation of d*int_array*, which is DINT[4,5].

Dimension 1

Dimension 0

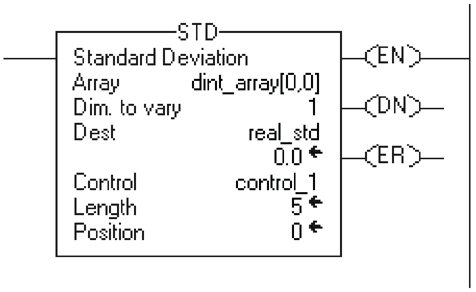
Subscripts	0	1	2	3	4
0	20	19	18	17	16
1	15	14	13	12	11
2	10	9	8	7	6
3	5	4	3	2	1

$$AVE = \frac{20 + 19 + 18 + 17 + 16}{5} = \frac{90}{5} = 18$$

$$STD = \sqrt{\frac{\langle 20 - 18\rangle^2 + \langle 19 - 18\rangle^2 + \langle 18 - 18\rangle^2 + \langle 17 - 18\rangle^2 + \langle 16 - 18\rangle^2}{\langle 5 - 1\rangle}} = 1.581139$$

real_std = 1.581139

Relay Ladder



Array dint_array[0,0]
Dim. to vary 1
Dest real_std
0.0
Control control_1
Length 5
Position 0

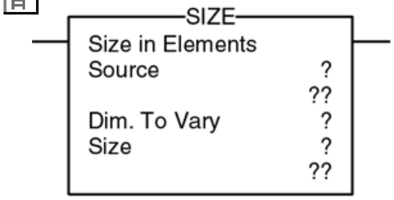
Structured Text

```
SIZE(dint_array,1,length);
sum := 0;
FOR position = 0 TO (length-1) DO
    sum := sum + dint_array[position];
END_FOR;
average := sum / length;
sum := 0;
FOR position = 0 TO (length-1) DO
    sum := sum + ((dint_array[position] - average)**2);
END_FOR;
real_std := SQRT(sum / (length-1));
```

Size In Elements (SIZE)

The SIZE instruction finds the size of a dimension of an array.

Operands:



Relay Ladder

<table>
<tr><th>Operand</th><th>Type</th><th>Format</th><th>Description</th></tr>
<tr><td>Source</td><td>SINT
INT
DINT
REAL
structure
string</td><td>Array tag</td><td>Array on which the instruction is to operate</td></tr>
<tr><td>Dimension to Vary</td><td>DINT</td><td>Immediate (0, 1, 2)</td><td>Dimension to use:<table><tr><th>For The Size Of</th><th>Enter</th></tr><tr><td>First dimension</td><td>0</td></tr><tr><td>Second dimension</td><td>1</td></tr><tr><td>Third dimension</td><td>2</td></tr></table></td></tr>
<tr><td>Size</td><td>SINT
INT
DINT
REAL</td><td>Tag</td><td>Tag to store the number of elements in the specified dimension of the array</td></tr>
</table>

Structured Text

```
SIZE(Source,Dimtovary,Size);
```

The operands are the same as those for the relay ladder SIZE instruction.

Description:

The SIZE instruction finds the number of elements (size) in the designated dimension of the Source array and places the result in the Size operand.

- The instruction finds the size of one dimension of an array.
- The instruction operates on an:
  - array
  - array in a structure
  - array that is part of a larger array

Arithmetic Status Flags:

Not affected

Fault Conditions:

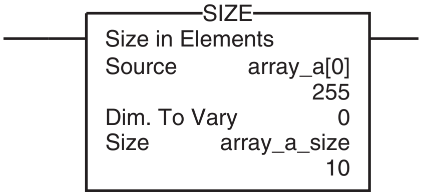
None.

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction finds the size of a dimension.	The instruction finds the size of a dimension.
Postscan	The rung-condition-out is set to false.	No action taken.

Example 1: Find the number of elements in dimension 0 (first dimension) of *array_a*. Store the size in *array_a_size*. In this example, dimension 0 of *array_a* has 10 elements.

Relay Ladder

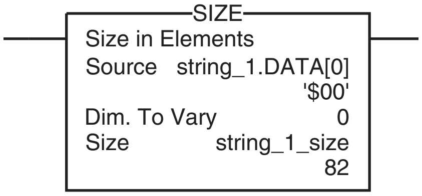


Structured Text

```
SIZE(array_a,0,array_a_size);
```

Example 2: Find the number of elements in the DATA member of *string_1*, which is a string. Store the size in *string_1_size*. In this example, the DATA member of *string_1* has 82 elements. (The string uses the default STRING data type.) Since each element holds one character, *string_1* can contain up to 82 characters.

Relay Ladder

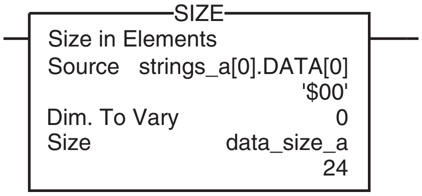


Structured Text

```
SIZE(string_1.DATA[0],0,string_1_size);
```

Example 3: *Strings_a* is an array of string structures. The SIZE instruction finds the number of elements in the DATA member of the string structure and stores the size in *data_size_a*. In this example, the DATA member has 24 elements. (The string structure has a user-specified length of 24.)

Relay Ladder



Structured Text

```
SIZE(strings_a[0].DATA[0],0,data_size_a);
```

Notes:

Array (File)/Shift Instructions

(BSL, BSR, FFL, FFU, LFL, LFU)

Introduction

Use the array (file)/shift instructions to modify the location of data within arrays.

If you want to	Use this instruction	Available in these languages	Page
Load bits into, shift bits through, and unload bits from a bit array one bit at a time	BSL	Relay ladder	394
	BSR	Relay ladder	398
Load and unload values in the same order	FFL	Relay ladder	402
	FFU	Relay ladder	408
Load and unload values in reverse order	LFL	Relay ladder	414
	LFU	Relay ladder	420

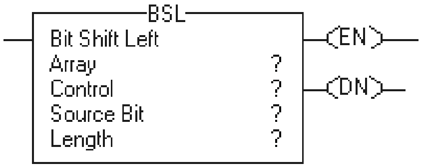
You can mix data types, but loss of accuracy and rounding errors might occur.

For relay ladder instructions, **bold** data types indicate optimal data types. An instruction executes faster and requires less memory if all the operands of the instruction use the same optimal data type, typically DINT or REAL.

Bit Shift Left (BSL)

The BSL instruction shifts the specified bits within the Array one position left.

Operands:



Relay Ladder

Operand	Type	Format	Description
Array	DINT	Array tag	Array to modify Specify the element where to begin the shift **Do not** use CONTROL.POS in the subscript
Control	CONTROL	Tag	Control structure for the operation
Source bit	BOOL	Tag	Bit to load
Length	DINT	Immediate	Number of bits in the array to shift

CONTROL Structure

Mnemonic	Data Type	Description
.EN	BOOL	The enable bit indicates that the BSL instruction is enabled.
.DN	BOOL	The done bit is set to indicate that bits shifted one position to the left.
.UL	BOOL	The unload bit is the instruction's output. The .UL bit stores the status of the bit that was shifted out of the range of bits.
.ER	BOOL	The error bit is set when .LEN < 0.
.LEN	DINT	The length specifies the number of array bits to shift.

Description: When enabled, the instruction unloads the uppermost bit of the specified bits to the .UL bit, shifts the remaining bits one position left, and loads Source bit into bit 0 of Array.

IMPORTANT You must test and confirm that the instruction doesn't change data that you don't want it to change.

The BSL instruction operates on contiguous memory. If an Array is a member array, such as contained within a structure, it is possible that the instruction could shift beyond the Array's boundary into other members following it. You must take care in choosing a length whereby this does not happen.

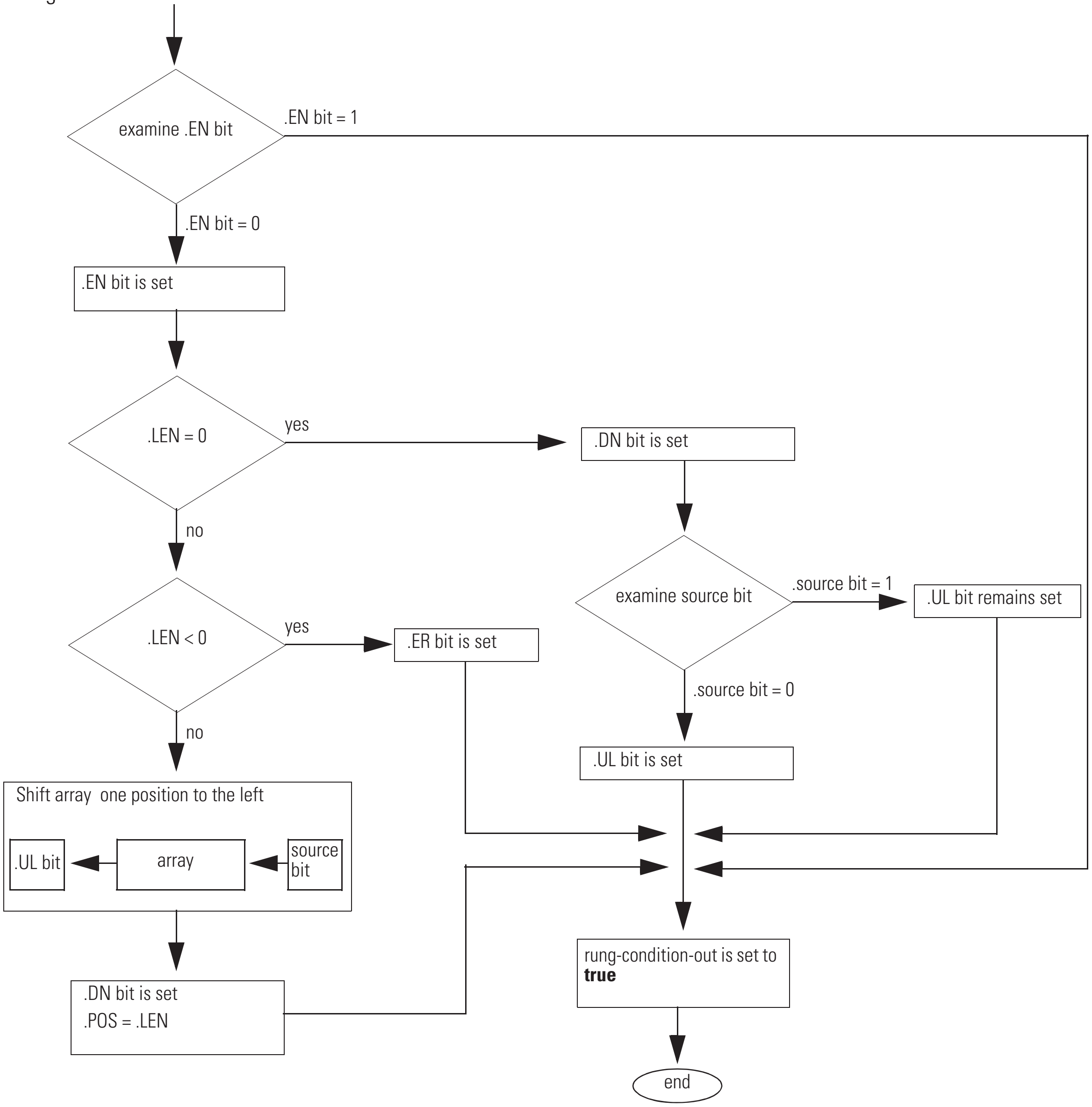
Arithmetic Status Flags: Not affected

Fault Conditions:

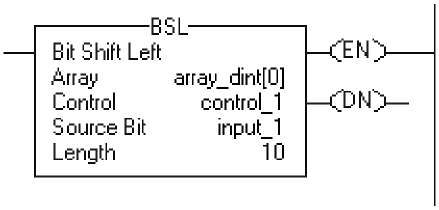
A major fault will occur if	Fault type	Fault code
Length exceeds the size of Array's storage area.	4	20

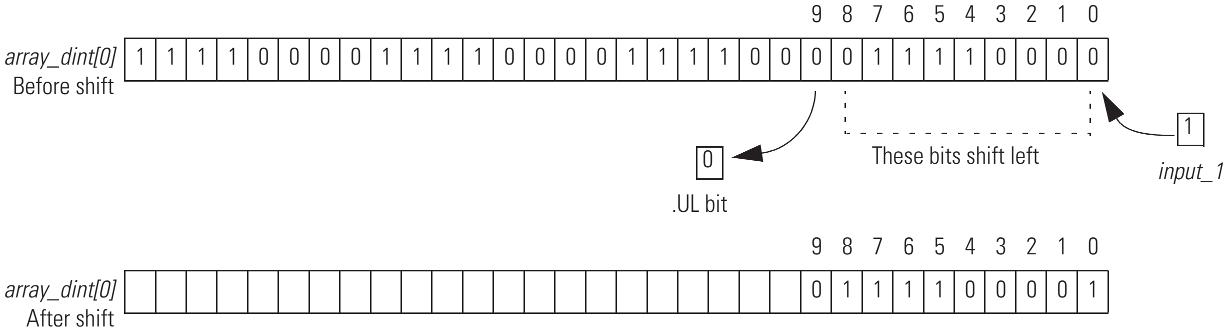
Execution:

Condition	Relay Ladder Action
Prescan	The .EN bit is cleared. The .DN bit is cleared. The .ER bit is cleared. The .POS value is cleared. The rung-condition-out is set to false.
Rung-condition-in is false	The .EN bit is cleared. The .DN bit is cleared. The .ER bit is cleared. The .POS value is cleared. The rung-condition-out is set to false.

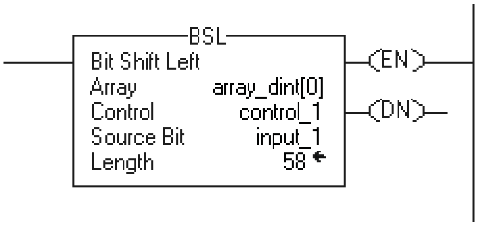
Condition	**Relay Ladder Action**
Rung-condition-in is true	
Postscan	The rung-condition-out is set to false.

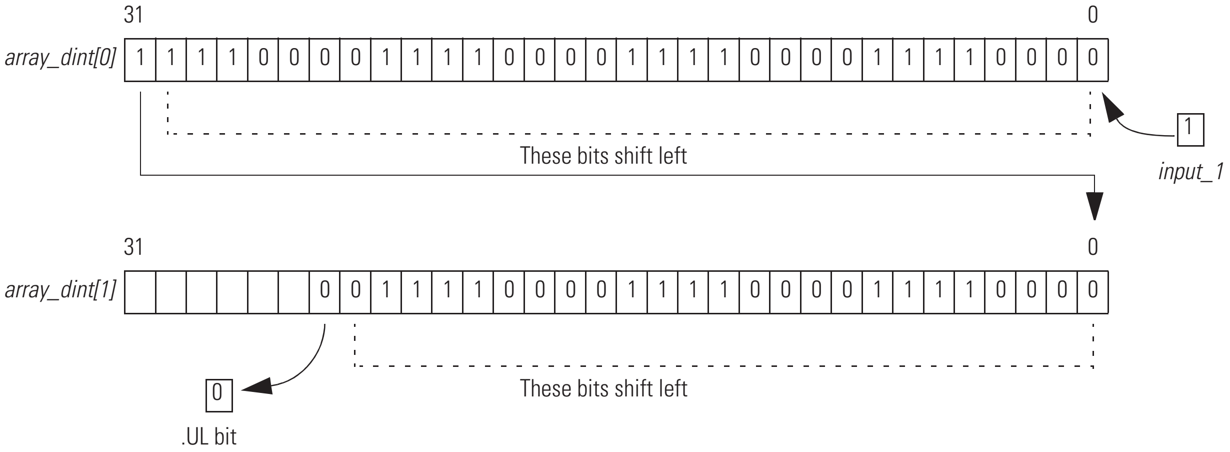
Example 1: When enabled, the BSL instruction starts at bit 0 in *array_dint[0]*. The instruction unloads *array_dint[0].9* into the .UL bit, shifts the remaining bits, and loads *input_1* into *array_dint[0].0*. The values in the remaining bits (10…31) are invalid.





Example 2: When enabled, the BSL instruction starts at bit 0 in *array_dint[0]*. The instruction unloads *array_dint[1].25* into the .UL bit, shifts the remaining bits, and loads *input_1* into *array_dint[0].0*. The values in the remaining bits (31…26 in *array_dint[1]*) are invalid. Note how *array_dint[0].31* shifts across words to *array_dint[1].0*.

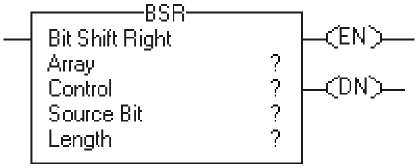




Bit Shift Right (BSR)

The BSR instruction shifts the specified bits within the Array one position right.

Operands:



Relay Ladder

Operand	Type	Format	Description
Array	DINT	Array tag	Array to modify Specify the element where to begin the shift **Do not** use CONTROL.POS in the subscript
Control	CONTROL	Tag	Control structure for the operation
Source bit	BOOL	Tag	Bit to load
Length	DINT	Immediate	Number of bits in the array to shift

CONTROL Structure

Mnemonic	Data Type	Description
.EN	BOOL	The enable bit indicates that the BSR instruction is enabled.
.DN	BOOL	The done bit is set to indicate that bits shifted one position to the right.
.UL	BOOL	The unload bit is the instruction's output. The .UL bit stores the status of the bit that was shifted out of the range of bits.
.ER	BOOL	The error bit is set when .LEN < 0.
.LEN	DINT	The length specifies the number of array bits to shift.

Description:

When enabled, the instruction unloads the value at bit 0 of Array to the .UL bit, shifts the remaining bits one position right, and loads Source bit into the uppermost bit of the specified bits.

IMPORTANT

You must test and confirm that the instruction doesn't change data that you don't want it to change.

The BSR instruction operates on contiguous memory. If an Array is a member array, such as contained within a structure, it is possible that the instruction could shift beyond the Array's boundary into other members following it. You must take care in choosing a length whereby this does not happen.

Arithmetic Status Flags:

Not affected

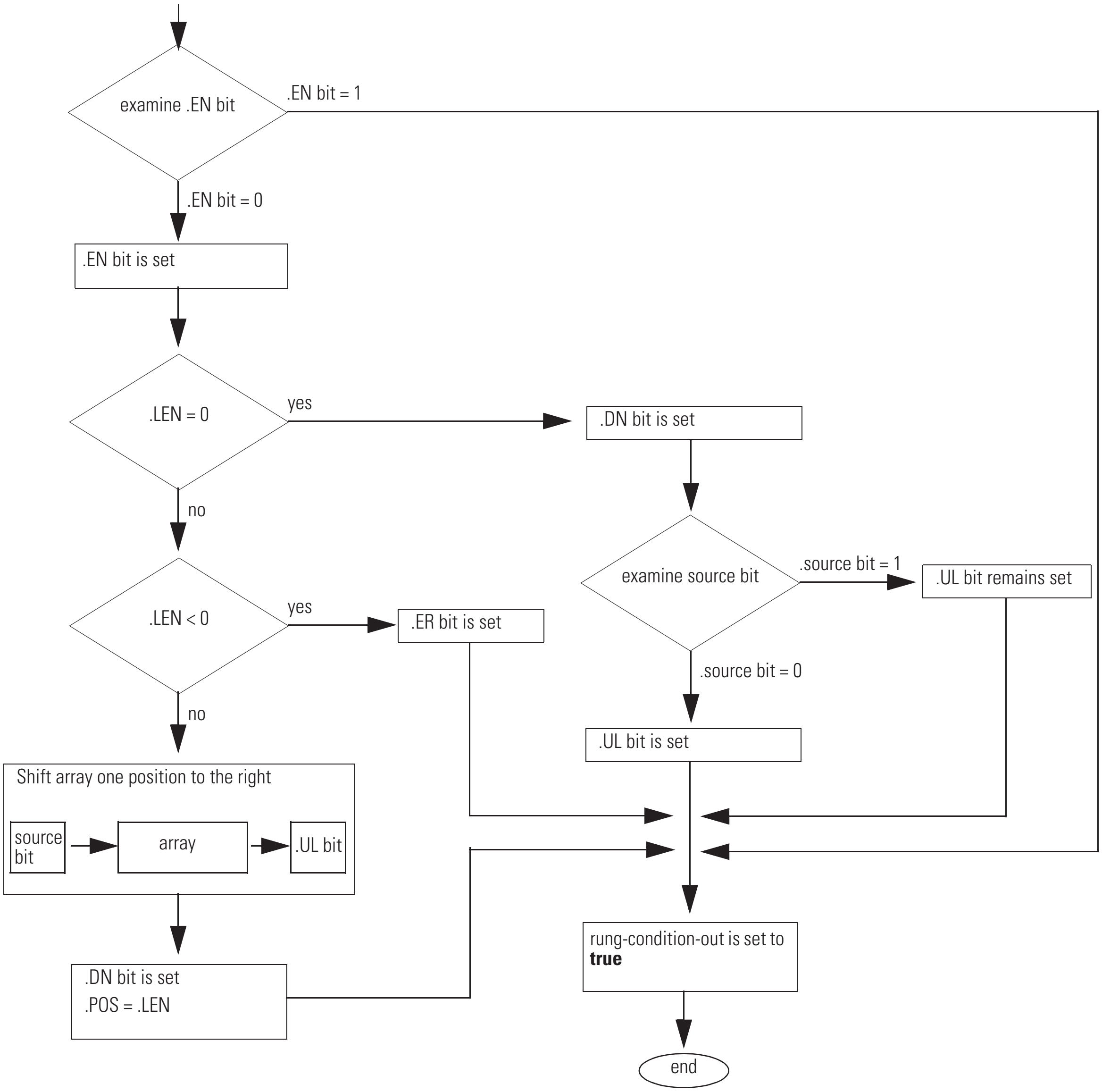
Fault Conditions:

A major fault will occur if	Fault type	Fault code
Length exceeds the size of Array's storage area.	4	20

Execution:

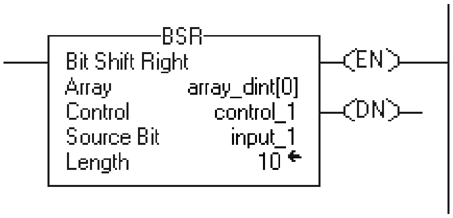
Condition	Relay Ladder Action
Prescan	The .EN bit is cleared. The .DN bit is cleared. The .ER bit is cleared. The .POS value is cleared. The rung-condition-out is set to false.
Rung-condition-in is false	The .EN bit is cleared. The .DN bit is cleared. The .ER bit is cleared. The .POS value is cleared. The rung-condition-out is set to false.

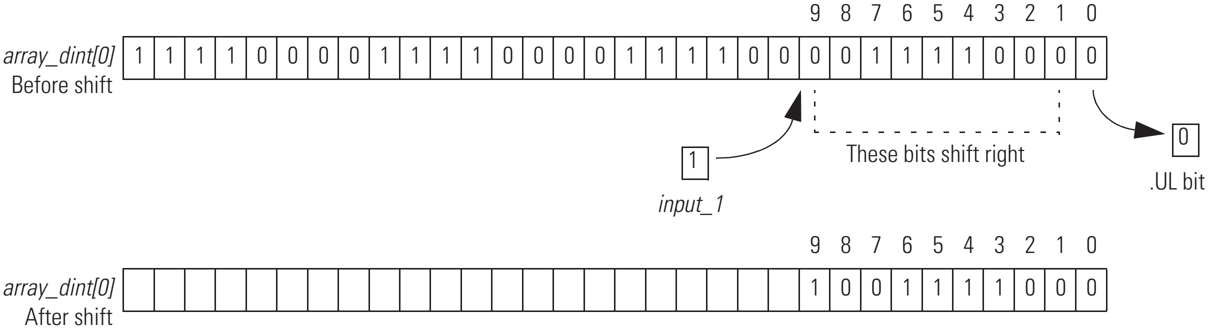
Condition	Relay Ladder Action
Rung-condition-in is true	



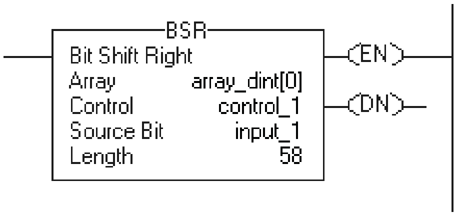
Condition	Relay Ladder Action
Postscan	The rung-condition-out is set to false.

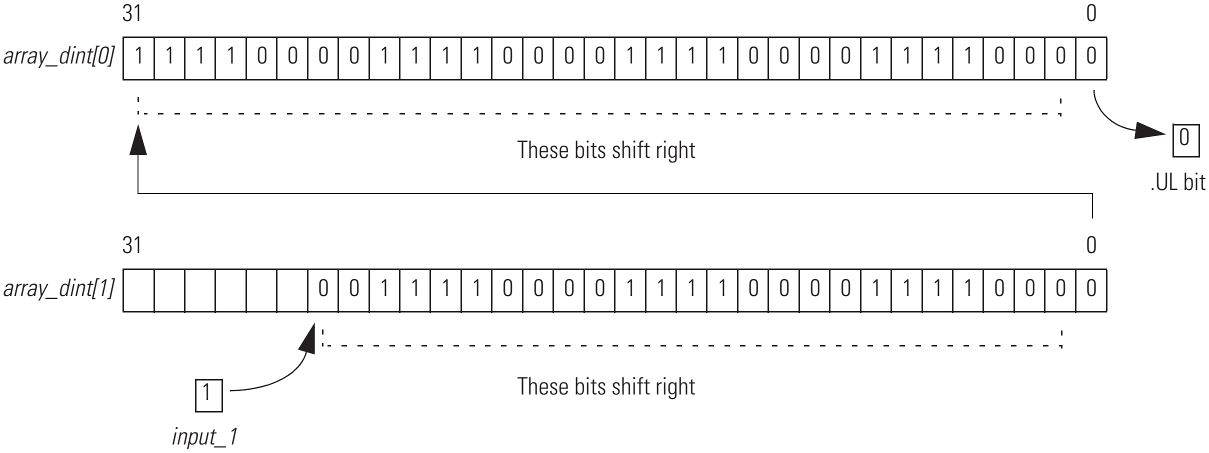
Example 1: When enabled, the BSR instruction starts at bit 9 in *array_dint[0]*. The instruction unloads *array_dint[0].0* into the .UL bit, shifts the remaining bits right, and loads *input_1* into *array_dint[0].9*. The values in the remaining bits (10…31) are invalid.





Example 2: When enabled, the BSR instruction starts at bit 25 in *array_dint[1]*. The instruction unloads *array_dint[0].0* into the .UL bit, shifts the remaining bits right, and loads *input_1* into *array_dint[1].25*. The values in the remaining bits (31…26 in *dint_array[1]*) are invalid. Note how *array_dint[1].0* shifts across words into *array_dint[0].31*.

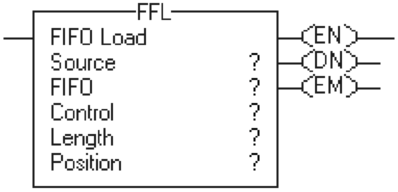




FIFO Load (FFL)

The FFL instruction copies the Source value to the FIFO.

Operands:



Relay Ladder

Operand	Type	Format	Description
Source	SINT INT DINT REAL string structure	Immediate Tag	Data to be stored in the FIFO
	The Source converts to the data type of the array tag. A smaller integer converts to a larger integer by sign-extension.		
FIFO	SINT INT DINT REAL string structure	Array tag	FIFO to modify Specify the first element of the FIFO **Do not** use CONTROL.POS in the subscript
Control	CONTROL	Tag	Control structure for the operation Typically use the same CONTROL as the associated FFU
Length	DINT	Immediate	Maximum number of elements the FIFO can hold at one time
Position	DINT	Immediate	Next location in the FIFO where the instruction loads data Initial value is typically 0

If you use a user-defined structure as the data type for the Source or FIFO operand, use the same structure for both operands.

CONTROL Structure

Mnemonic	Data Type	Description
.EN	BOOL	The enable bit indicates that the FFL instruction is enabled.
.DN	BOOL	The done bit is set to indicate that the FIFO is full (.POS = .LEN). The .DN bit inhibits loading the FIFO until .POS < .LEN.
.EM	BOOL	The empty bit indicates that the FIFO is empty. If .LEN ≤0 or .POS < 0, both the .EM bit and .DN bit are set.
.LEN	DINT	The length specifies the maximum number of elements the FIFO can hold at one time.
.POS	DINT	The position identifies the location in the FIFO where the instruction will load the next value.

Description:

Use the FFL instruction with the FFU instruction to store and retrieve data in a first-in/first-out order. When used in pairs, the FFL and FFU instructions establish an asynchronous shift register.

Typically, the Source and the FIFO are the same data type.

When enabled, the FFL instruction loads the Source value into the position in the FIFO identified by the .POS value. The instruction loads one value each time the instruction is enabled, until the FIFO is full.

IMPORTANT

You must test and confirm that the instruction doesn't change data that you don't want it to change.

The FFL instruction operates on contiguous memory. In some cases, the instruction loads data past the array into other members of the tag. This happens if the length is too big and the tag is a user-defined data type.

Arithmetic Status Flags:

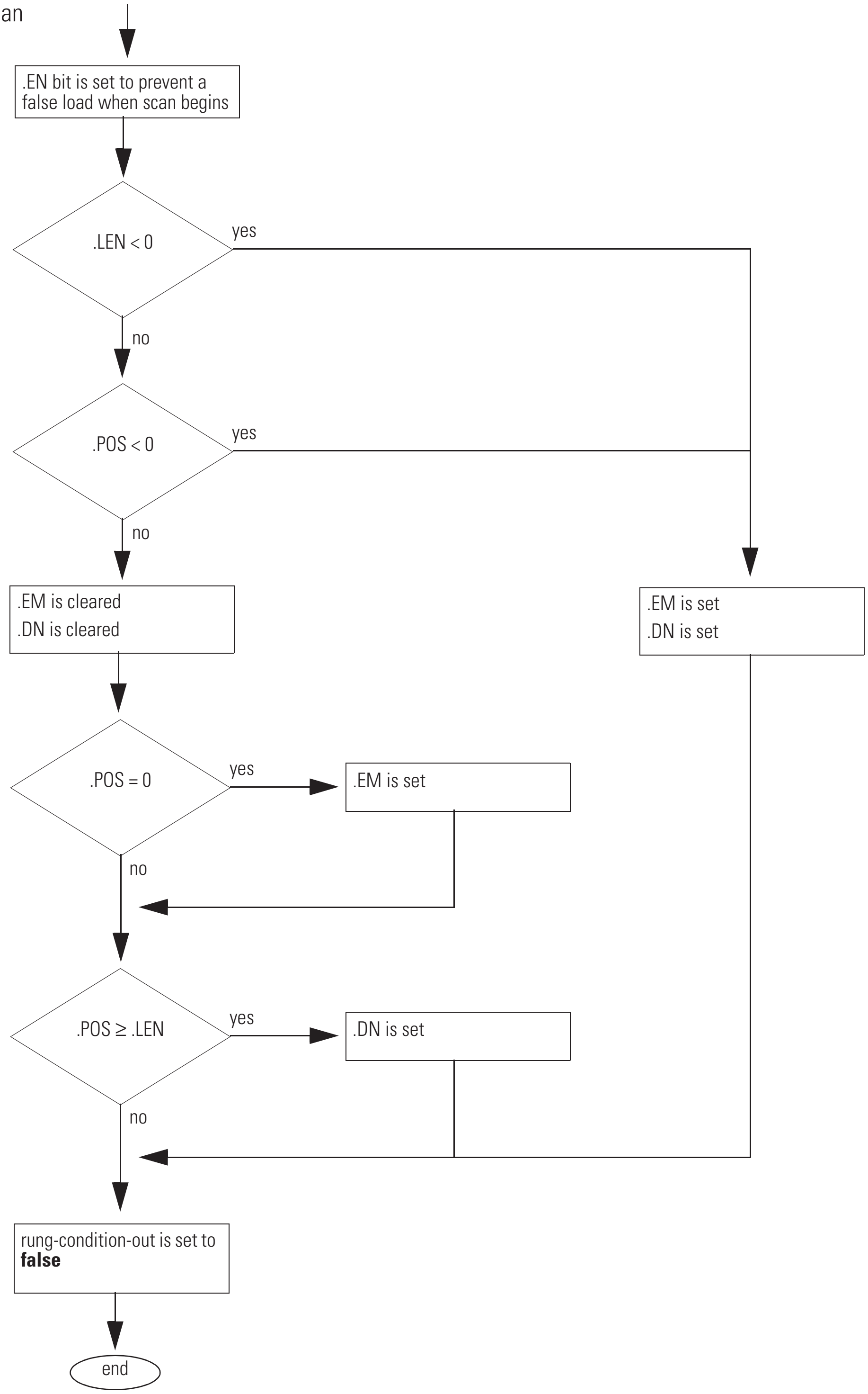
Not affected

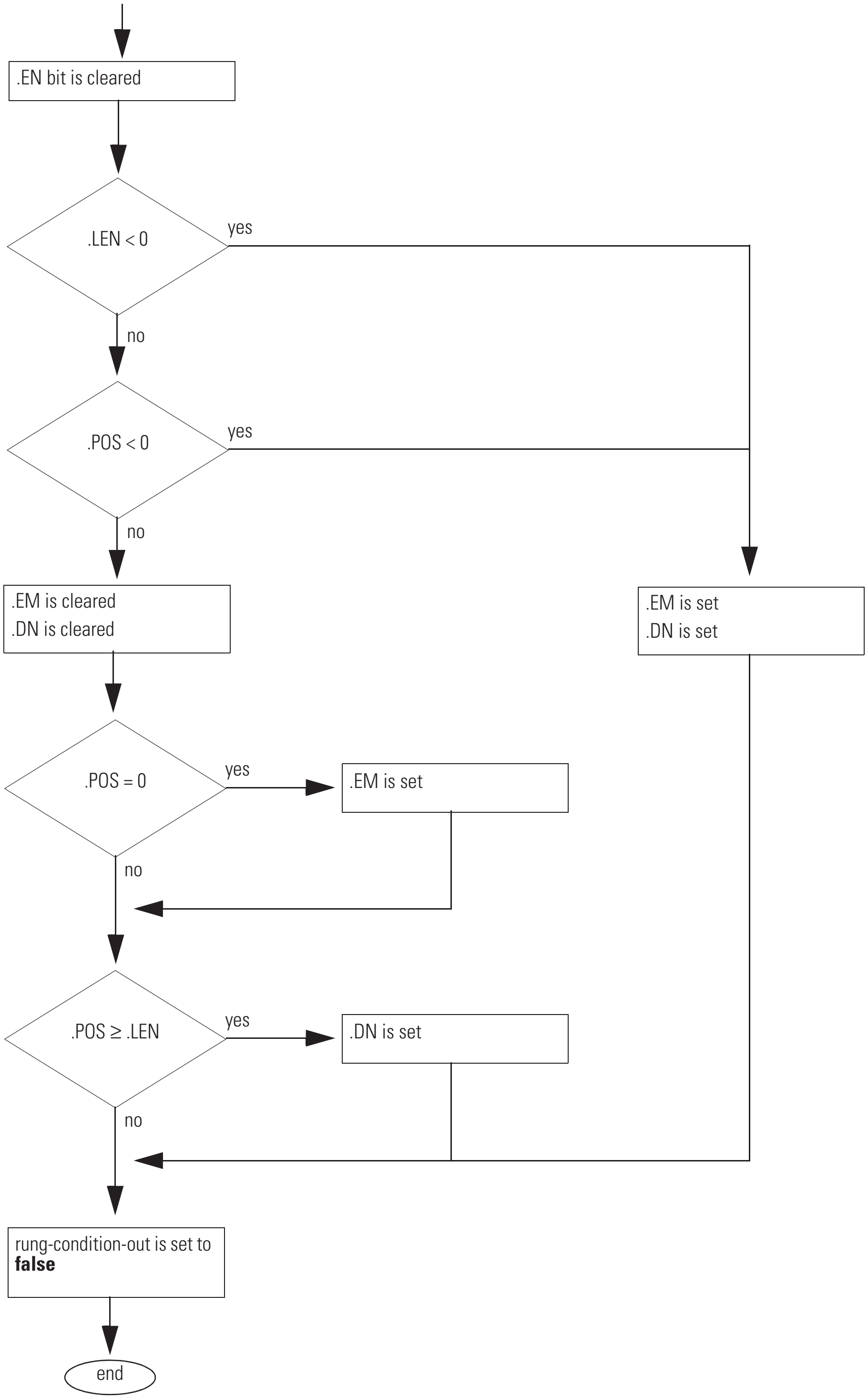
Fault Conditions:

A major fault will occur if	Fault type	Fault code
(starting element + .POS) > FIFO array size	4	20

Execution:

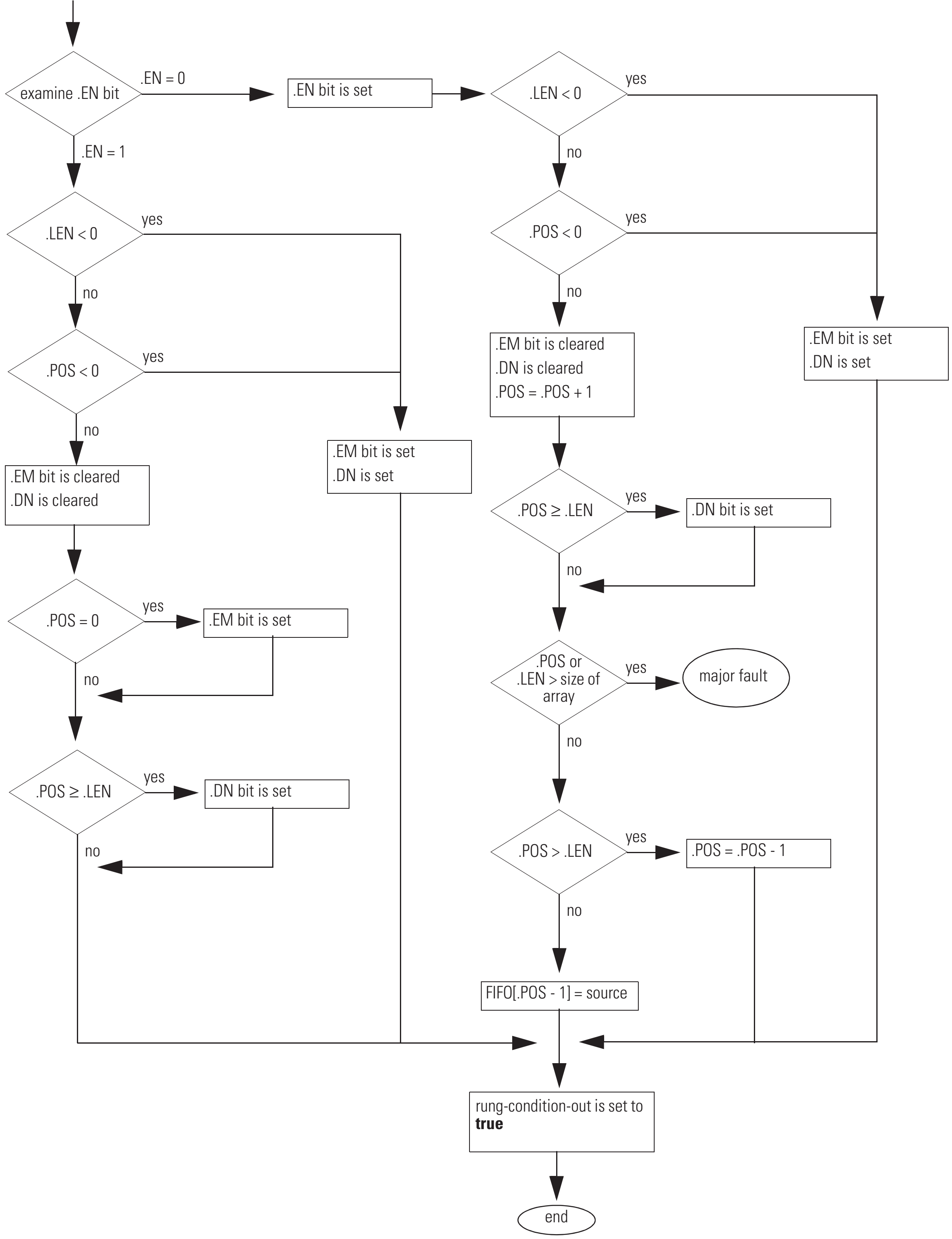
Condition	Relay Ladder Action
Prescan	



Condition	Relay Ladder Action
Rung-condition-in is false	

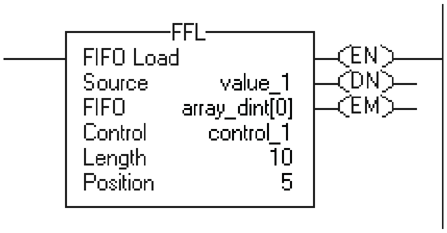
Condition	Relay Ladder Action

Rung-condition-in is true

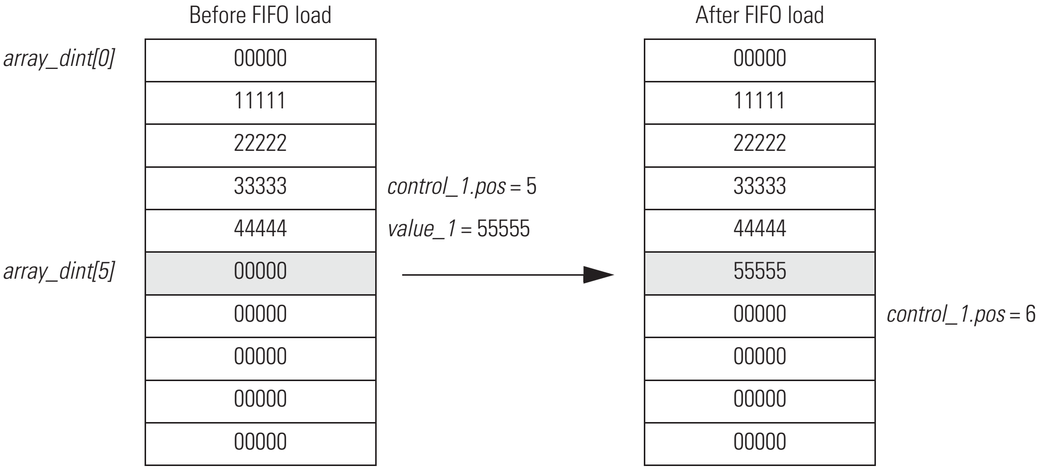


Postscan	The rung-condition-out is set to false.

Example: When enabled, the FFL instruction loads *value_1* into the next position in the FIFO, which is *array_dint[5]* in this example.



FIFO Load
Source value_1
FIFO array_dint[0]
Control control_1
Length 10
Position 5

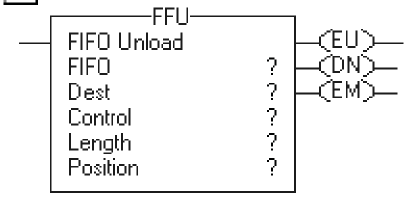


	Before FIFO load		After FIFO load	
array_dint[0]	00000		00000	
	11111		11111	
	22222		22222	
	33333	*control_1.pos* = 5	33333	
	44444	*value_1* = 55555	44444	
array_dint[5]	00000	→	55555	
	00000		00000	*control_1.pos* = 6
	00000		00000	
	00000		00000	
	00000		00000	

FIFO Unload (FFU)

The FFU instruction unloads the value from position 0 (first position) of the FIFO and stores that value in the Destination. The remaining data in the FIFO shifts down one position.

Operands:



Relay Ladder

Operand	Type	Format	Description
FIFO	SINT INT DINT REAL string structure	Array tag	FIFO to modify Specify the first element of the FIFO **Do not** use CONTROL.POS in the subscript
Destination	SINT INT DINT REAL string structure	Tag	Value that exits the FIFO
	The Destination value converts to the data type of the Destination tag. A smaller integer converts to a larger integer by sign-extension.		
Control	CONTROL	Tag	Control structure for the operation Typically use the same CONTROL as the associated FFL
Length	DINT	Immediate	Maximum number of elements the FIFO can hold at one time
Position	DINT	Immediate	Next location in the FIFO where the instruction unloads data Initial value is typically 0

If you use a user-defined structure as the data type for the FIFO or Destination operand, use the same structure for both operands.

CONTROL Structure

Mnemonic	Data Type	Description
.EU	BOOL	The enable unload bit indicates that the FFU instruction is enabled. The .EU bit is set to preset a false unload when the program scan begins.
.DN	BOOL	The done bit is set to indicate that the FIFO is full (.POS = .LEN).
.EM	BOOL	The empty bit indicates that the FIFO is empty. If .LEN ≤ 0 or .POS < 0, the .EM bit and .DN bits are set.
.LEN	DINT	The length specifies the maximum number of elements in the FIFO.
.POS	DINT	The position identifies the end of the data that has been loaded into the FIFO.

Description:

Use the FFU instruction with the FFL instruction to store and retrieve data in a first-in/first-out order.

When enabled, the FFU instruction unloads data from the first element of the FIFO and places that value in the Destination. The instruction unloads one value each time the instruction is enabled, until the FIFO is empty. If the FIFO is empty, the FFU returns 0 to the Destination.

IMPORTANT

You must test and confirm that the instruction doesn’t change data that you don’t want it to change.

The FFU instruction operates on contiguous memory. In some cases, the instruction unloads data from other members of the tag. This happens if the length is too big and the tag is a user-defined data type.

Arithmetic Status Flags:

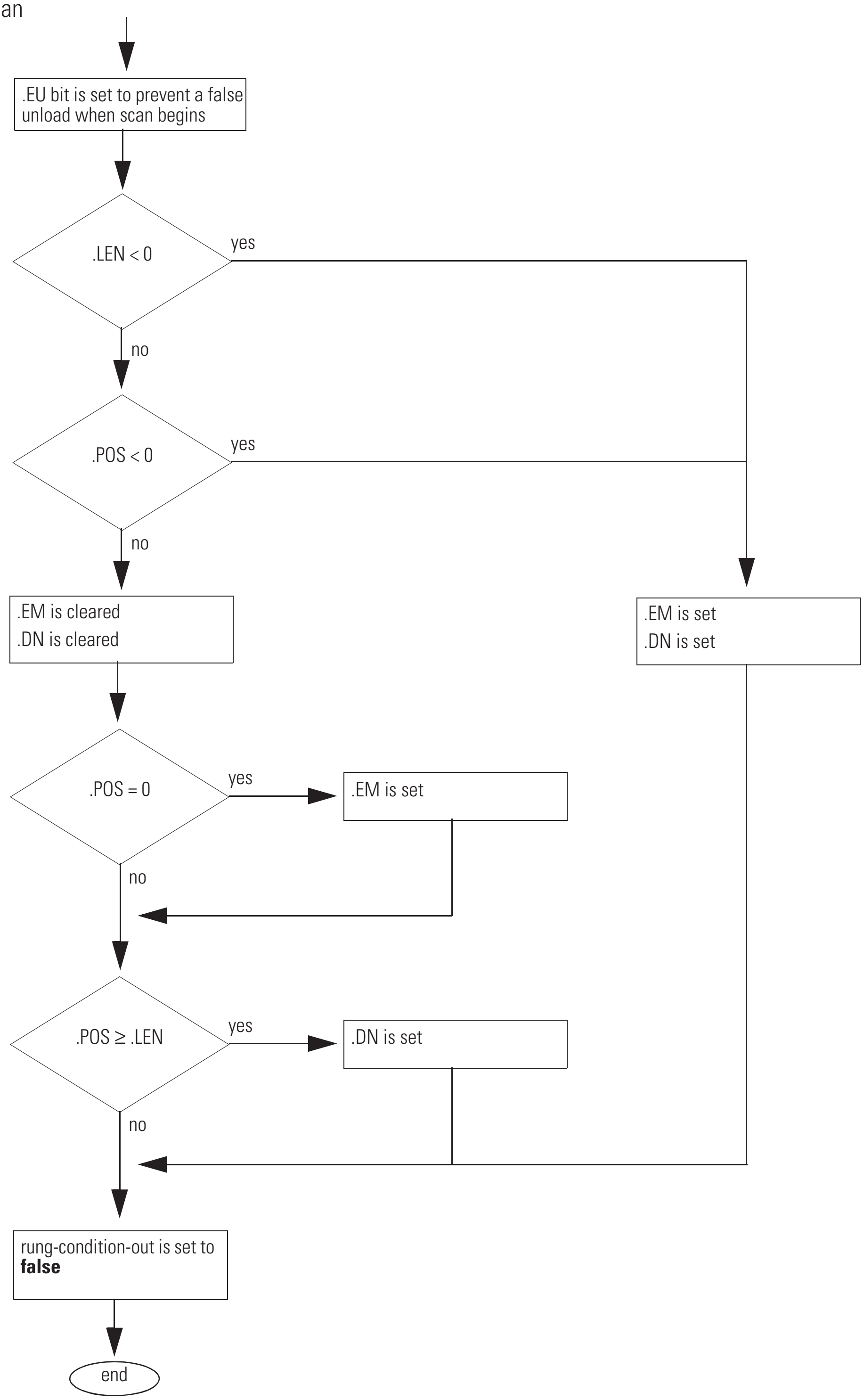
Not affected

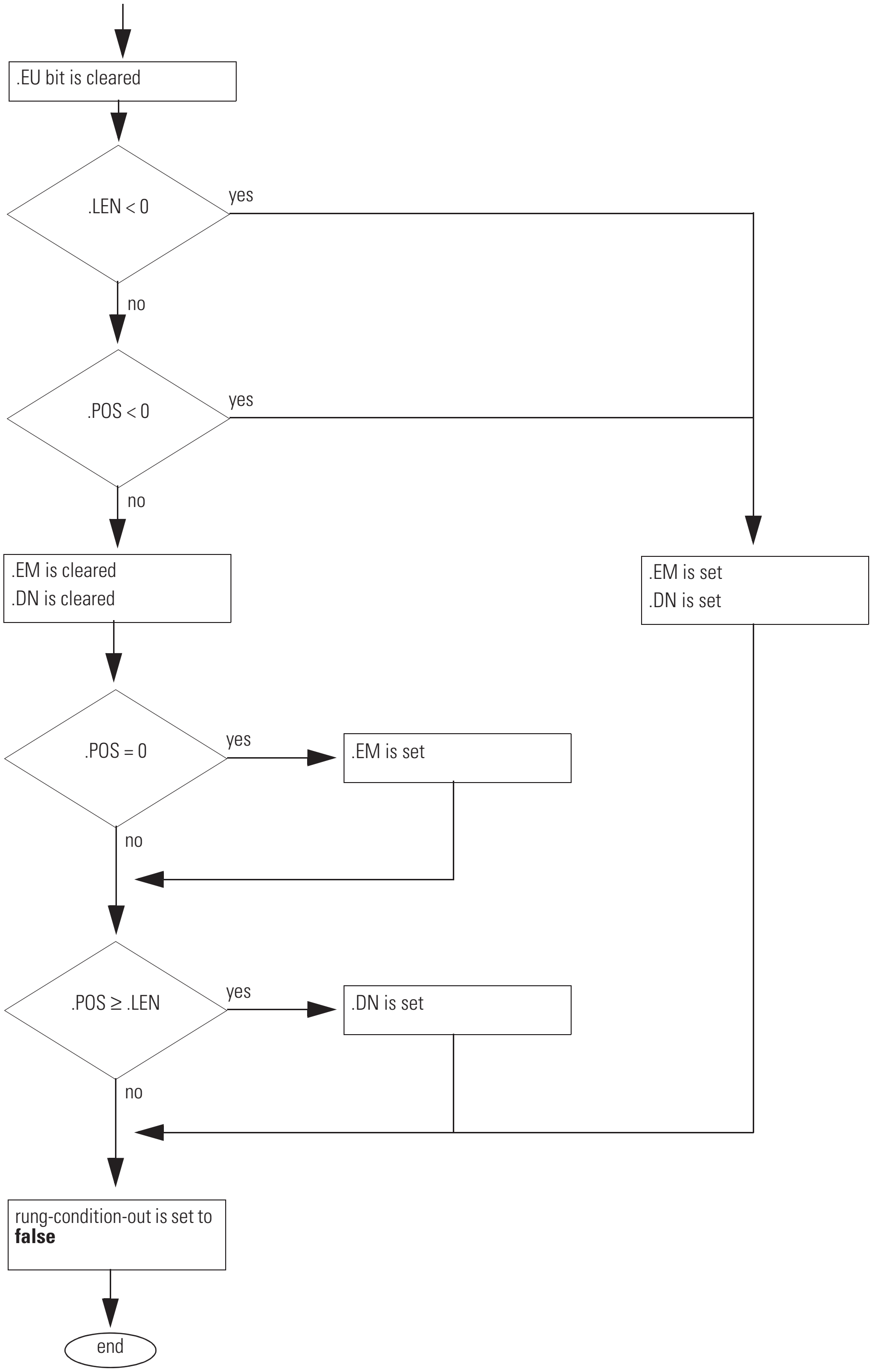
Fault Conditions:

A major fault will occur if	Fault type	Fault code
Length > FIFO array size	4	20

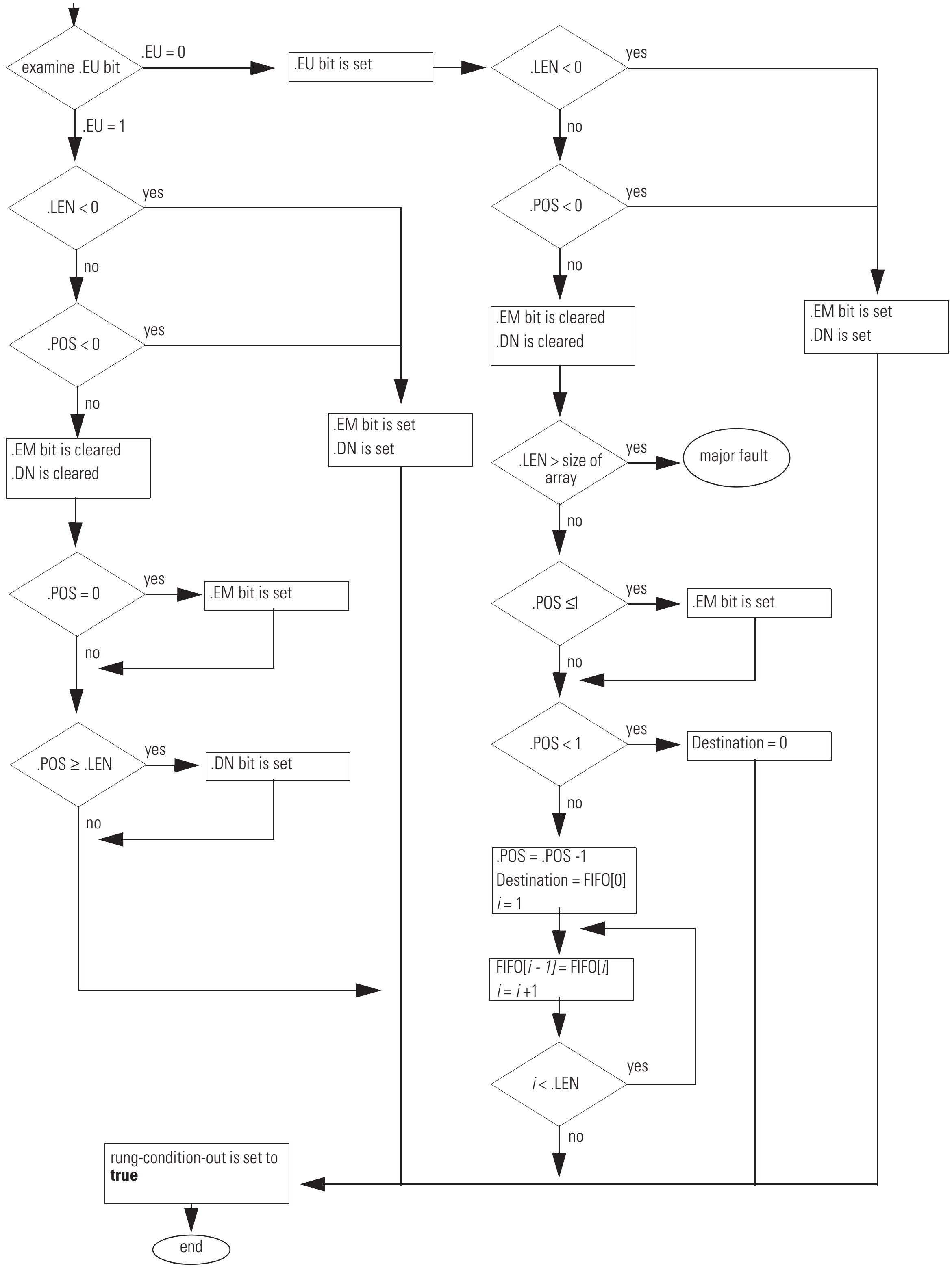
Execution:

Condition	Relay Ladder Action
Prescan	



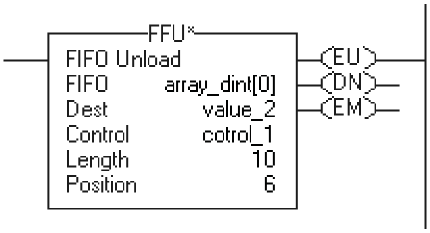
Condition	**Relay Ladder Action**
Rung-condition-in is false	

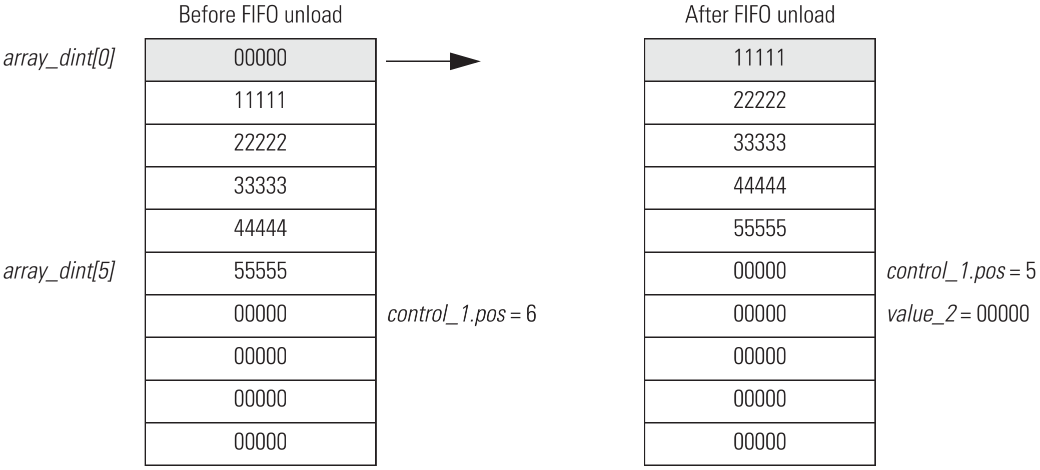
Condition	Relay Ladder Action
Rung-condition-in is true	



Postscan	The rung-condition-out is set to false.

Example: When enabled, the FFU instruction unloads *array_dint[0]* into *value_2* and shifts the remaining elements in *array_dint*.



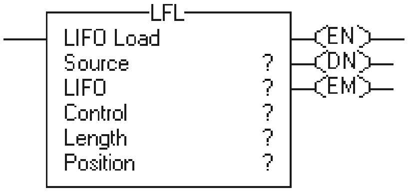


	Before FIFO unload		After FIFO unload	
array_dint[0]	00000	→	11111	
	11111		22222	
	22222		33333	
	33333		44444	
	44444		55555	
array_dint[5]	55555		00000	*control_1.pos* = 5
	00000	*control_1.pos* = 6	00000	*value_2* = 00000
	00000		00000	
	00000		00000	
	00000		00000	

LIFO Load (LFL)

The LFL instruction copies the Source value to the LIFO.

Operands:



Relay Ladder

Operand	Type	Format	Description
Source	**SINT** **INT** **DINT** **REAL** **string** **structure**	Immediate Tag	Data to be stored in the LIFO
	The Source converts to the data type of the array tag. A smaller integer converts to a larger integer by sign-extension.		
LIFO	**SINT** **INT** **DINT** **REAL** **string** **structure**	Array tag	LIFO to modify Specify the first element of the LIFO **Do not** use CONTROL.POS in the subscript
Control	CONTROL	Tag	Control structure for the operation Typically use the same CONTROL as the associated LFU
Length	DINT	Immediate	Maximum number of elements the LIFO can hold at one time
Position	DINT	Immediate	Next location in the LIFO where the instruction loads data initial value is typically 0

If you use a user-defined structure as the data type for the Source or LIFO operand, use the same structure for both operands.

CONTROL Structure

Mnemonic	Data Type	Description:
.EN	BOOL	The enable bit indicates that the LFL instruction is enabled.
.DN	BOOL	The done bit is set to indicate that the LIFO is full (.POS = .LEN). The .DN bit inhibits loading the LIFO until .POS < .LEN.
.EM	BOOL	The empty bit indicates that the LIFO is empty. If .LEN ≤0 or .POS < 0, both the .EM bit and .DN bit are set.
.LEN	DINT	The length specifies the maximum number of elements the LIFO can hold at one time.
.POS	DINT	The position identifies the location in the LIFO where the instruction will load the next value.

Description: Use the LFL instruction with the LFU instruction to store and retrieve data in a last-in/first-out order. When used in pairs, the LFL and LFU instructions establish an asynchronous shift register.

Typically, the Source and the LIFO are the same data type.

When enabled, the LFL instruction loads the Source value into the position in the LIFO identified by the .POS value. The instruction loads one value each time the instruction is enabled, until the LIFO is full.

IMPORTANT

You must test and confirm that the instruction doesn't change data that you don't want it to change.

The LFL instruction operates on contiguous memory. In some cases, the instruction loads data past the array into other members of the tag. This happens if the length is too big and the tag is a user-defined data type.

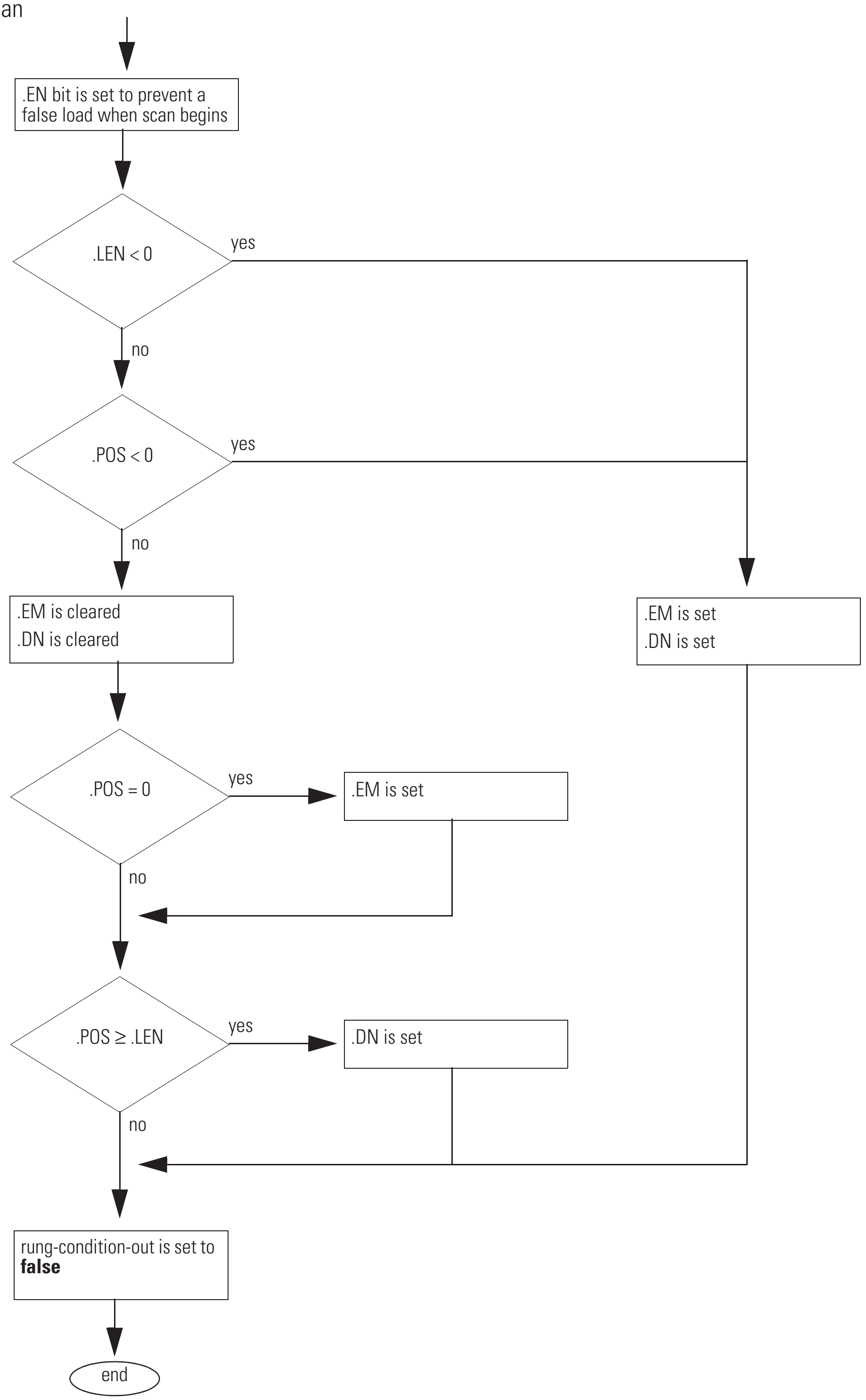
Arithmetic Status Flags: Not affected

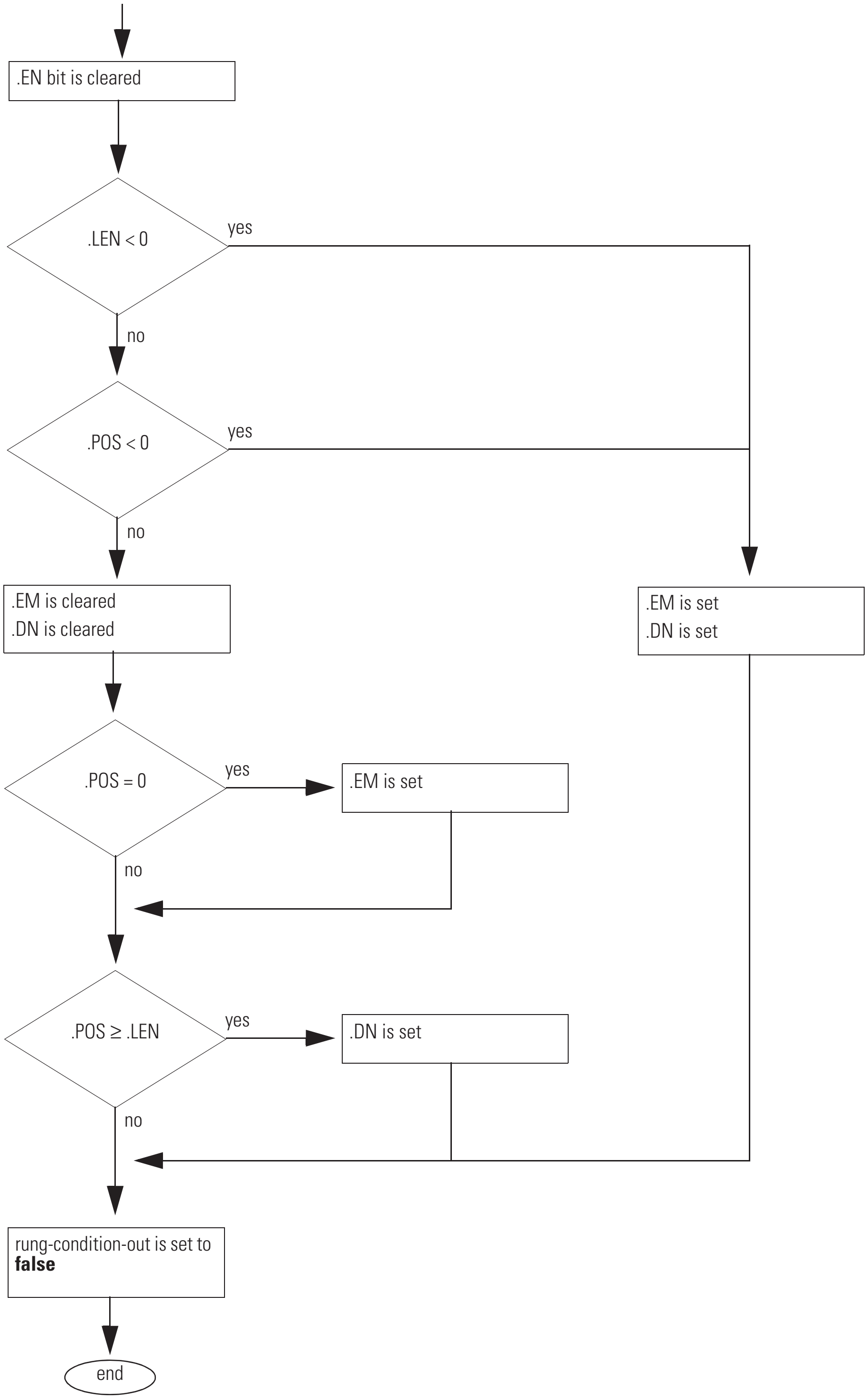
Fault Conditions:

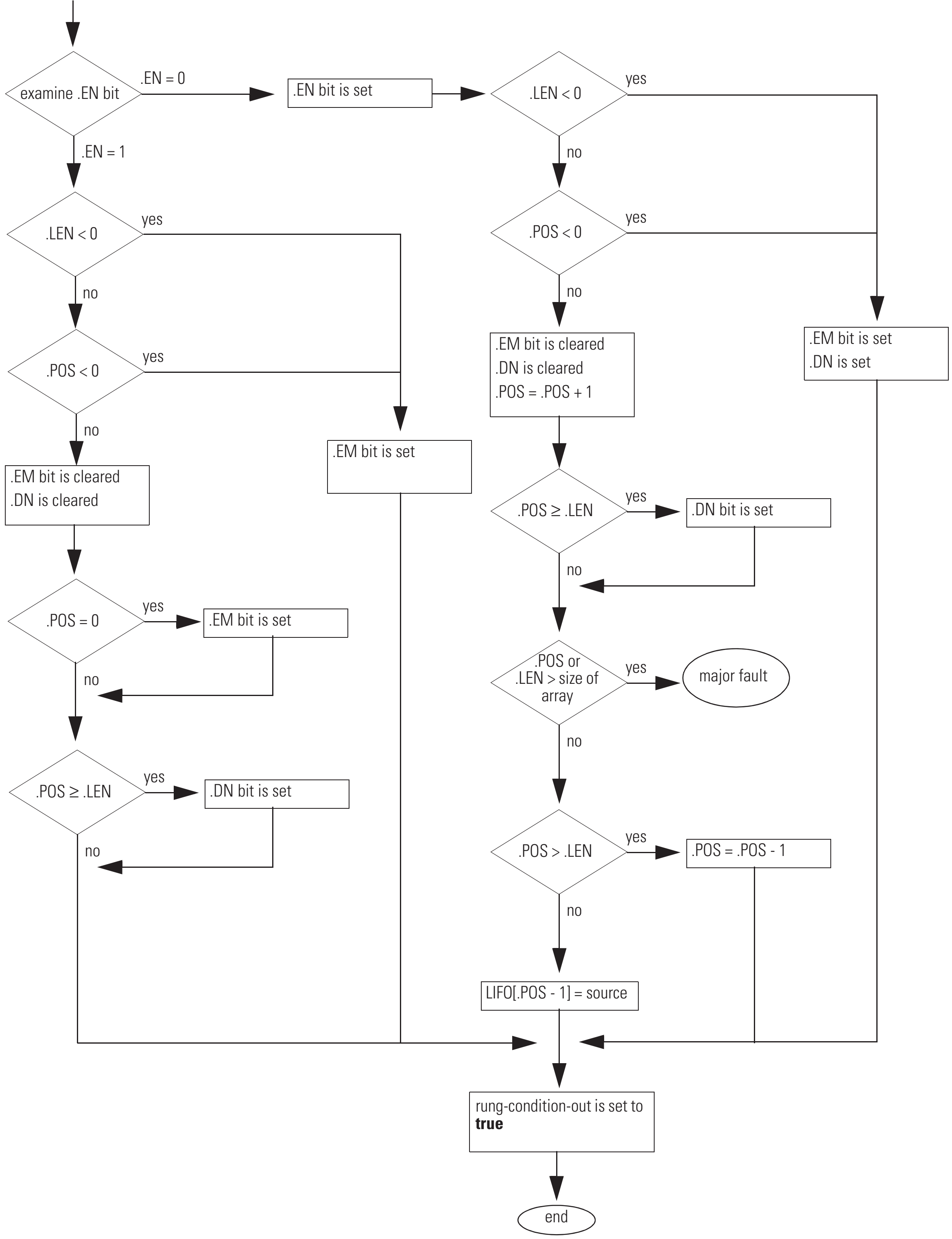
A major fault will occur if	Fault type	Fault code
(starting element + .POS) > LIFO array size	4	20

Execution:

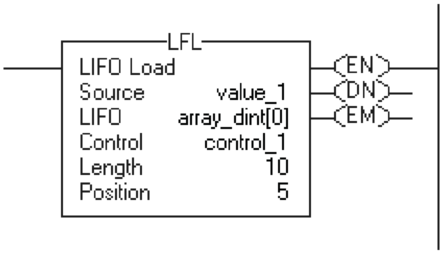
Condition:	**Relay Ladder Action**
Prescan	



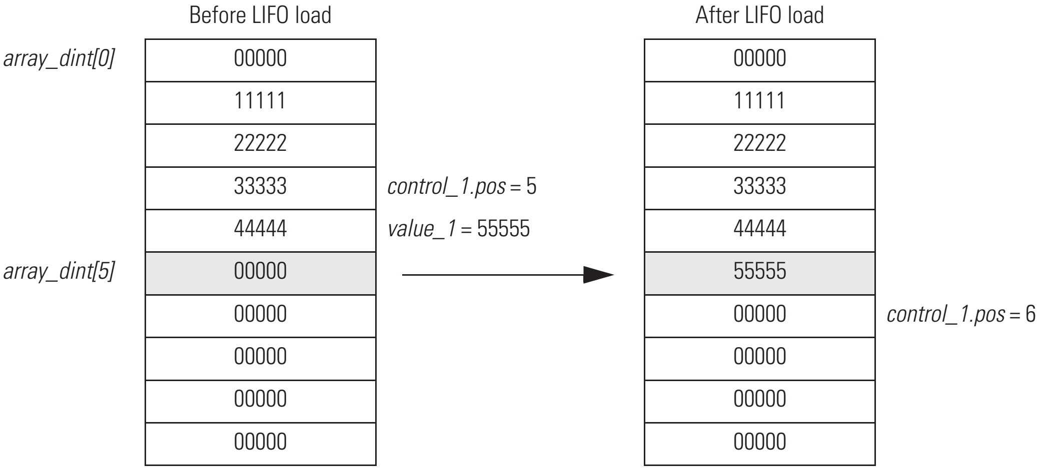
Condition:	**Relay Ladder Action**
Rung-condition-in is false	

Condition:	**Relay Ladder Action**
Rung-condition-in is true	
Postscan	The rung-condition-out is set to false.

Example: When enabled, the LFL instruction loads *value_1* into the next position in the LIFO, which is *array_dint[5]* in this example.



LFL
LIFO Load
Source value_1
LIFO array_dint[0]
Control control_1
Length 10
Position 5

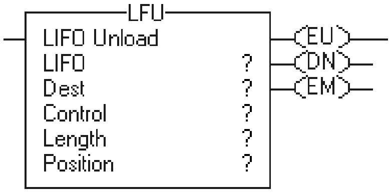


	Before LIFO load		After LIFO load	
array_dint[0]	00000		00000	
	11111		11111	
	22222		22222	
	33333	*control_1.pos* = 5	33333	
	44444	*value_1* = 55555	44444	
array_dint[5]	00000	→	55555	
	00000		00000	*control_1.pos* = 6
	00000		00000	
	00000		00000	
	00000		00000	

LIFO Unload (LFU)

The LFU instruction unloads the value at .POS of the LIFO and stores 0 in that location.

Operands:



Relay Ladder

Operand	Type	Format	Description
LIFO	SINT INT DINT REAL string structure	Array tag	LIFO to modify Specify the first element of the LIFO **Do not** use CONTROL.POS in the subscript
Destination	SINT INT DINT REAL string structure	Tag	Value that exits the LIFO
	The Destination value converts to the data type of the Destination tag. A smaller integer converts to a larger integer by sign-extension.		
Control	CONTROL	Tag	Control structure for the operation Typically use the same CONTROL as the associated LFL
Length	DINT	Immediate	Maximum number of elements the LIFO can hold at one time
Position	DINT	Immediate	Next location in the LIFO where the instruction unloads data Initial value is typically 0

If you use a user-defined structure as the data type for the LIFO or Destination operand, use the same structure for both operands.

CONTROL Structure

Mnemonic	Data Type:	Description
.EU	BOOL	The enable unload bit indicates that the LFU instruction is enabled. The .EU bit is set to preset a false unload when the program scan begins.
.DN	BOOL	The done bit is set to indicate that the LIFO is full (.POS = .LEN).
.EM	BOOL	The empty bit indicates that the LIFO is empty. If .LEN ≤0 or .POS < 0, both the .EM bit and .DN bit are set.
.LEN	DINT	The length specifies the maximum number of elements the LIFO can hold at one time.
.POS	DINT	The position identifies the end of the data that has been loaded into the LIFO.

Description:

Use the LFU instruction with the LFL instruction to store and retrieve data in a last-in/first-out order.

When enabled, the LFU instruction unloads the value at .POS of the LIFO and places that value in the Destination. The instruction unloads one value and replaces it with 0 each time the instruction is enabled, until the LIFO is empty. If the LIFO is empty, the LFU returns 0 to the Destination.

IMPORTANT

You must test and confirm that the instruction doesn’t change data that you don’t want it to change.

The LFU instruction operates on contiguous memory. In some cases, the instruction unloads data from other members of the tag. This happens if the length is too big and the tag is a user-defined data type.

Arithmetic Status Flags:

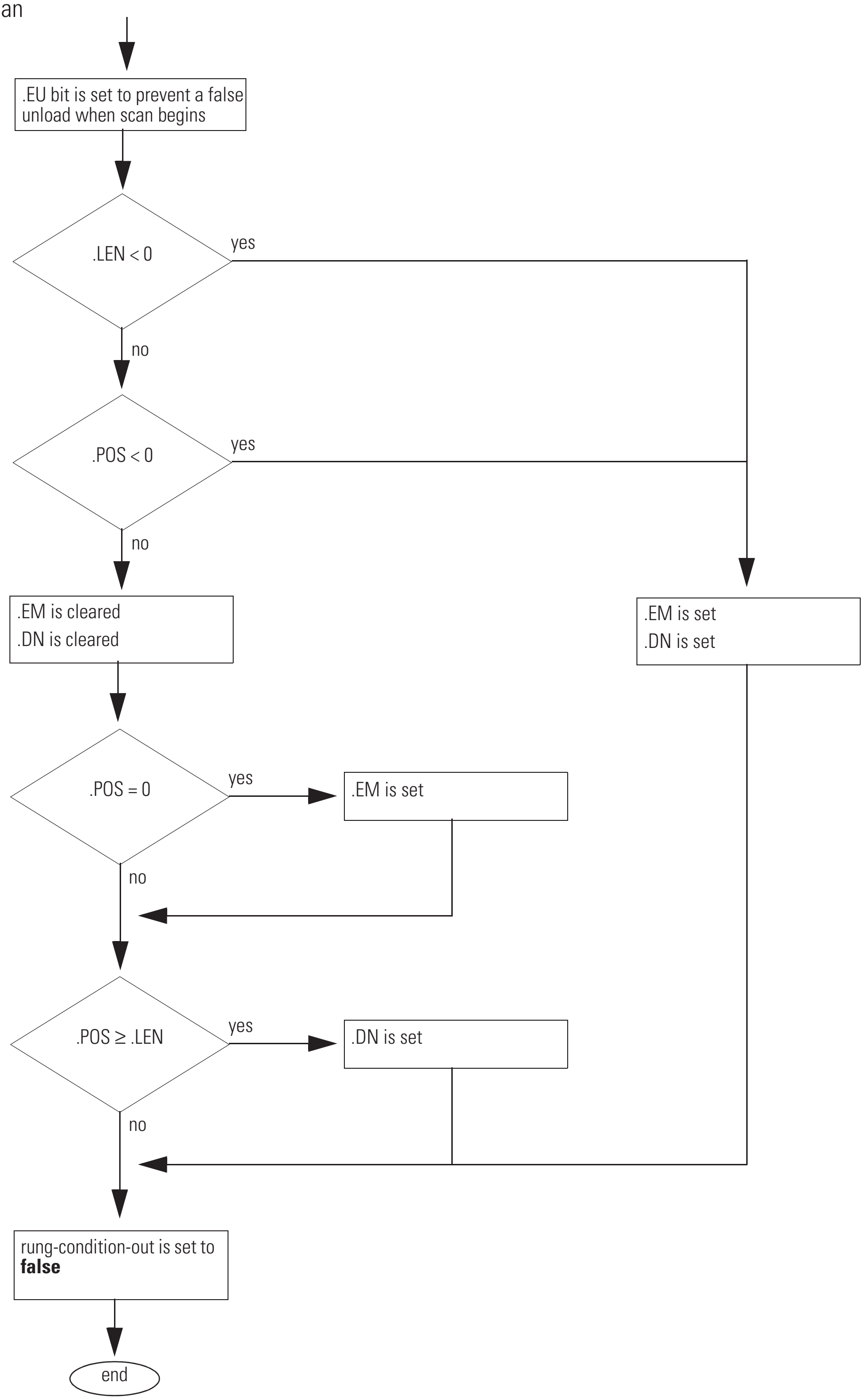
Not affected

Fault Conditions:

A major fault will occur if	Fault type	Fault code
Length > LIFO array size	4	20

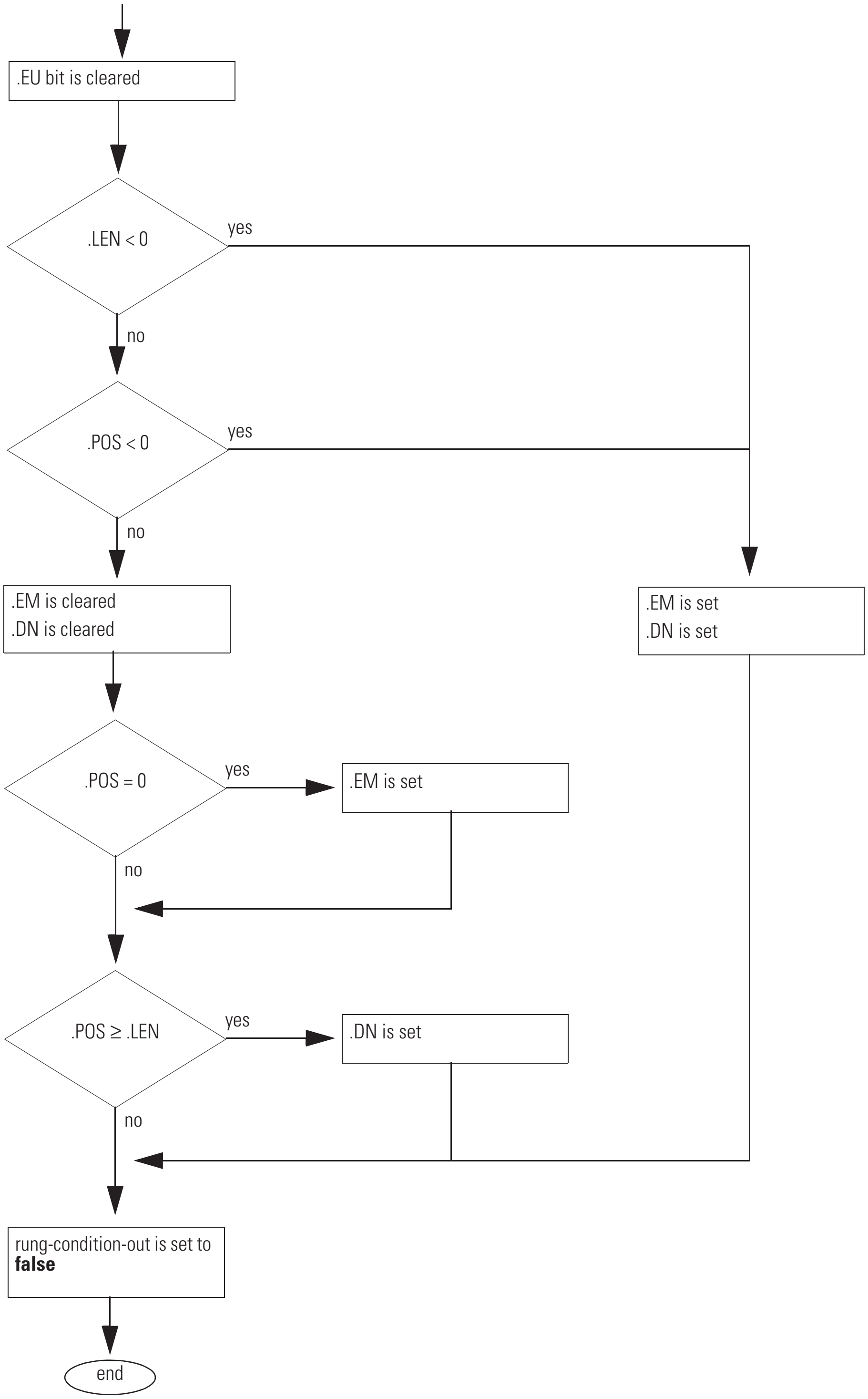
Execution:

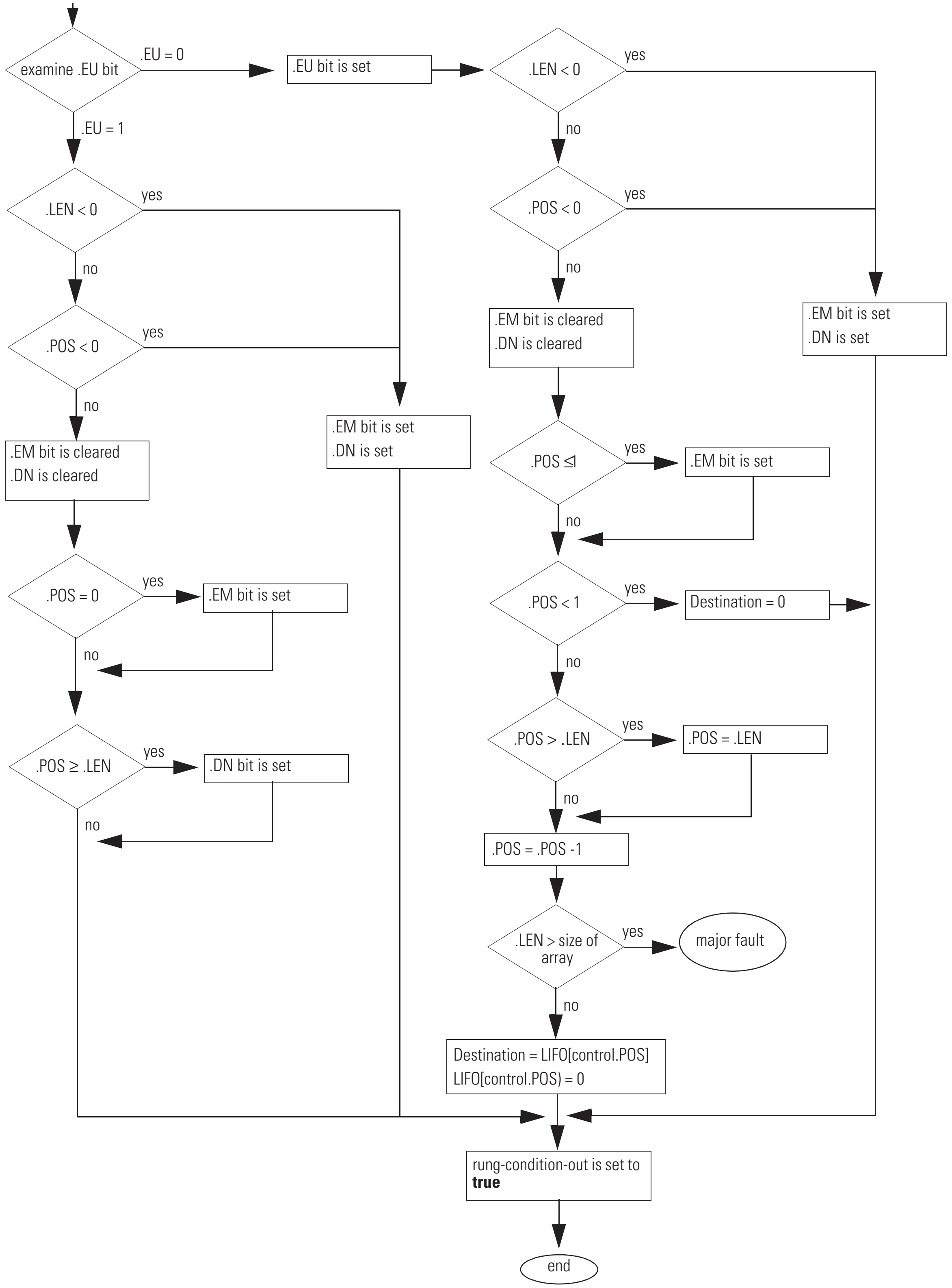
Condition	**Relay Ladder Action**
Prescan	



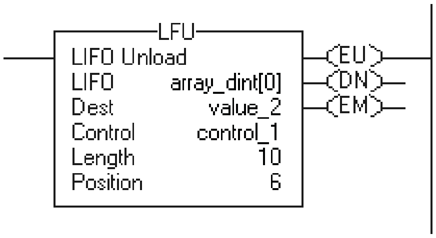
Condition	Relay Ladder Action

Rung-condition-in is false

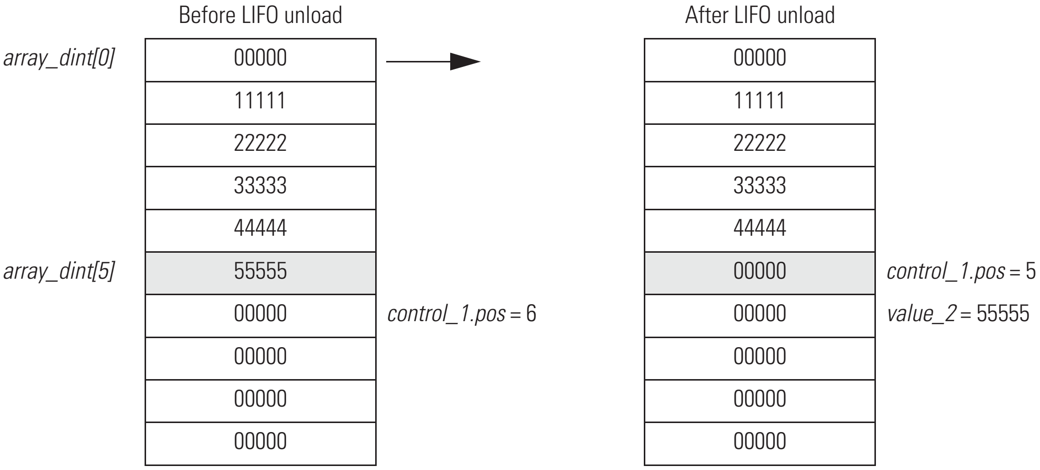


Condition	**Relay Ladder Action**
Rung-condition-in is true	
Postscan	The rung-condition-out is set to false.

Example: When enabled, the LFU instruction unloads *array_dint[5]* into *value_2*.



LFU
LIFO Unload
LIFO array_dint[0]
Dest value_2
Control control_1
Length 10
Position 6



	Before LIFO unload		After LIFO unload	
array_dint[0]	00000		00000	
	11111		11111	
	22222		22222	
	33333		33333	
	44444		44444	
array_dint[5]	55555		00000	*control_1.pos* = 5
	00000	*control_1.pos* = 6	00000	*value_2* = 55555
	00000		00000	
	00000		00000	
	00000		00000	

Notes:

Sequencer Instructions

(SQI, SQO, SQL)

Introduction

No action taken. Sequencer instructions monitor consistent and repeatable operations.

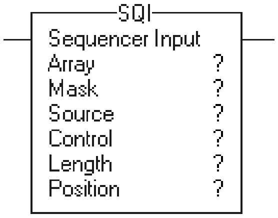
If you want to	Use this instruction	Available in these languages	Page
Detect when a step is complete	SQI	Relay ladder	428
Set output conditions for the next step	SQO	Relay ladder	432
Load reference conditions into sequencer arrays	SQL	Relay ladder	436

For relay ladder instructions, **bold** data types indicate optimal data types. An instruction executes faster and requires less memory if all the operands of the instruction use the same optimal data type, typically DINT or REAL.

Sequencer Input (SQI)

The SQI instruction detects when a step is complete in a sequence pair of SQO/SQI instructions.

Operands:



Relay Ladder

Operand	Type	Format	Description
Array	DINT	Array tag	Sequencer array Specify the first element of the sequencer array **Do not** use CONTROL.POS in the subscript
Mask	SINT INT **DINT**	Tag Immediate	Which bits to block or pass
	A SINT or INT tag converts to a DINT value by sign-extension.		
Source	SINT INT DINT	Tag	Input data for the sequencer array
	A SINT or INT tag converts to a DINT value by sign-extension.		
Control	CONTROL	Tag	Control structure for the operation Typically use the same CONTROL as the SQO and SQL instructions
Length	DINT	Immediate	Number of elements in the Array (sequencer table) to compare
Position	DINT	Immediate	Current position in the array initial value is typically 0

CONTROL Structure

Mnemonic	Data Type	Description
.ER	BOOL	The error bit is set when .LEN ≤0, .POS < 0, or .POS > .LEN.
.LEN	DINT	The length specifies the number of steps in the sequencer array.
.POS	DINT	The position identifies the element that the instruction is currently comparing.

Description: When enabled, the SQI instruction compares a Source element through a Mask to an Array element for equality.

Typically use the same CONTROL structure as the SQO and SQL instructions.

The SQI instruction operates on contiguous memory.

Enter an Immediate Mask Value

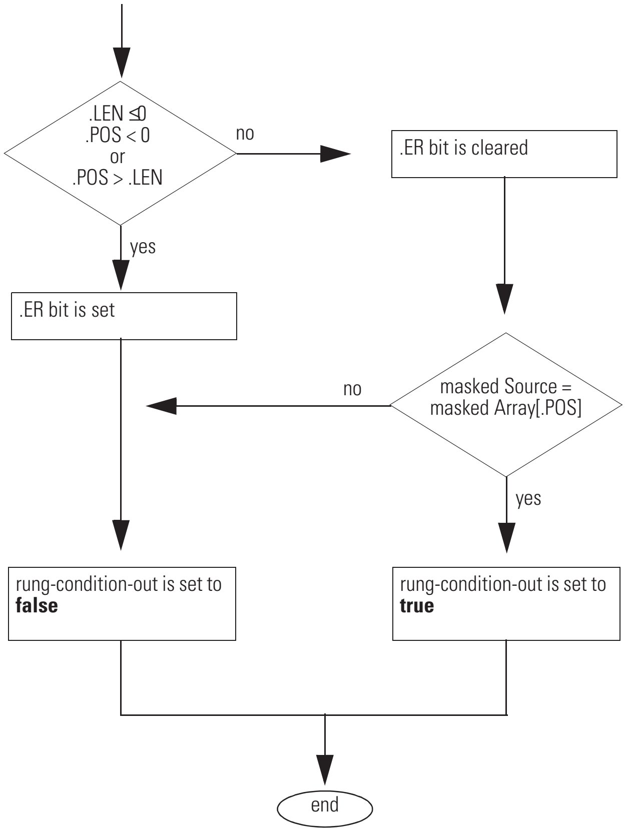
When you enter a mask, the programming software defaults to decimal values. If you want to enter a mask by using another format, precede the value with the correct prefix.

Prefix	Description
16#	Hexadecimal For example; 16#0F0F
8#	Octal For example; 8#16
2#	Binary For example; 2#00110011

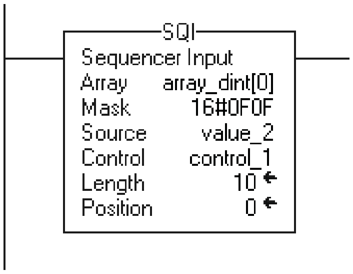
Arithmetic Status Flags: Not affected

Fault Conditions: None

Execution:

Condition:	Relay Ladder Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	
Postscan	The rung-condition-out is set to false.

Example: When enabled, the SQI instruction passes *value_2* through the mask to determine whether the result is equal to the current element in *array_dint*. The masked comparison is true, so the rung-condition-out goes true.



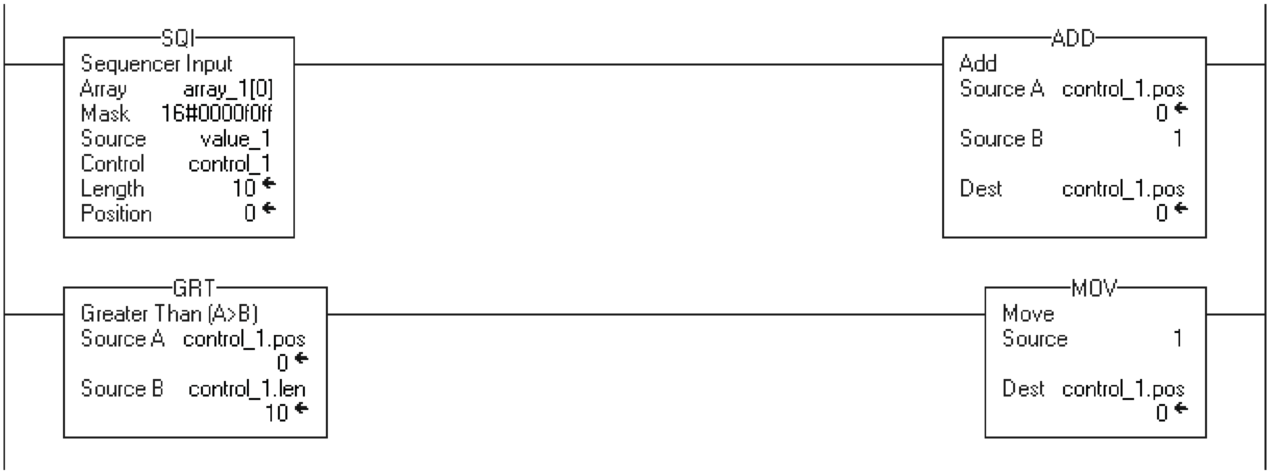
SQI Operand	Example Values (DINTs Displayed In Binary)
Source	`xxxxxxxx xxxxxxxx xxxx0101 xxxx1010`
Mask	`00000000 00000000 00001111 00001111`
Array	`xxxxxxxx xxxxxxxx xxxx0101 xxxx1010`

A 0 in the mask means the bit is not compared (designated by xxxx in this example).

Use SQI without SQO

If you use the SQI instruction without a paired SQO instruction, you have to externally increment the sequencer array.

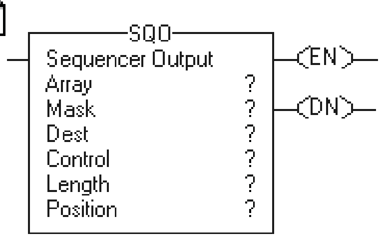
The SQI instruction compares the source value. The ADD instruction increments the sequencer array. The GRT determined whether another value is available to check in the sequencer array. The MOV instruction resets the position value after completely stepping through the sequencer array one time.



Sequencer Output (SQO)

The SQO instruction sets output conditions for the next step of a sequence pair of SQO/SQI instructions.

Operands:



Relay Ladder

Operand	Type	Format	Description
Array	DINT	Array tag	Sequencer array Specify the first element of the sequencer array **Do not** use CONTROL.POS in the subscript
Mask	SINT INT DINT	Tag Immediate	Which bits to block or pass
	A SINT or INT tag converts to a DINT value by sign-extension.		
Destination	DINT	Tag	Output data from the sequencer array
Control	CONTROL	Tag	Control structure for the operation typically use the same CONTROL as the SQI and SQL instructions
Length	DINT	Immediate	Number of elements in the Array (sequencer table) to output
Position	DINT	Immediate	Current position in the array Initial value is typically 0

CONTROL Structure

Mnemonic	Data Type	Description
.EN	BOOL	The enable bit indicates that the SQO instruction is enabled.
.DN	BOOL	The done bit is set when all the specified elements have been moved to the Destination.
.ER	BOOL	The error bit is set when .LEN ≤0, .POS < 0, or .POS > .LEN.
.LEN	DINT	The length specifies the number of steps in the sequencer array.
.POS	DINT	The position identifies the element that the controller is currently manipulating.

Description: When enabled, the SQO instruction increments the position, moves the data at the position through the Mask, and stores the result in the Destination. If .POS > .LEN, the instruction wraps around to the beginning of the sequencer array and continues with .POS = 1.

Typically, use the same CONTROL structure as the SQI and SQL instructions.

The SQO instruction operates on contiguous memory.

Enter an Immediate Mask Value

When you enter a mask, the programming software defaults to decimal values. If you want to enter a mask by using another format, precede the value with the correct prefix.

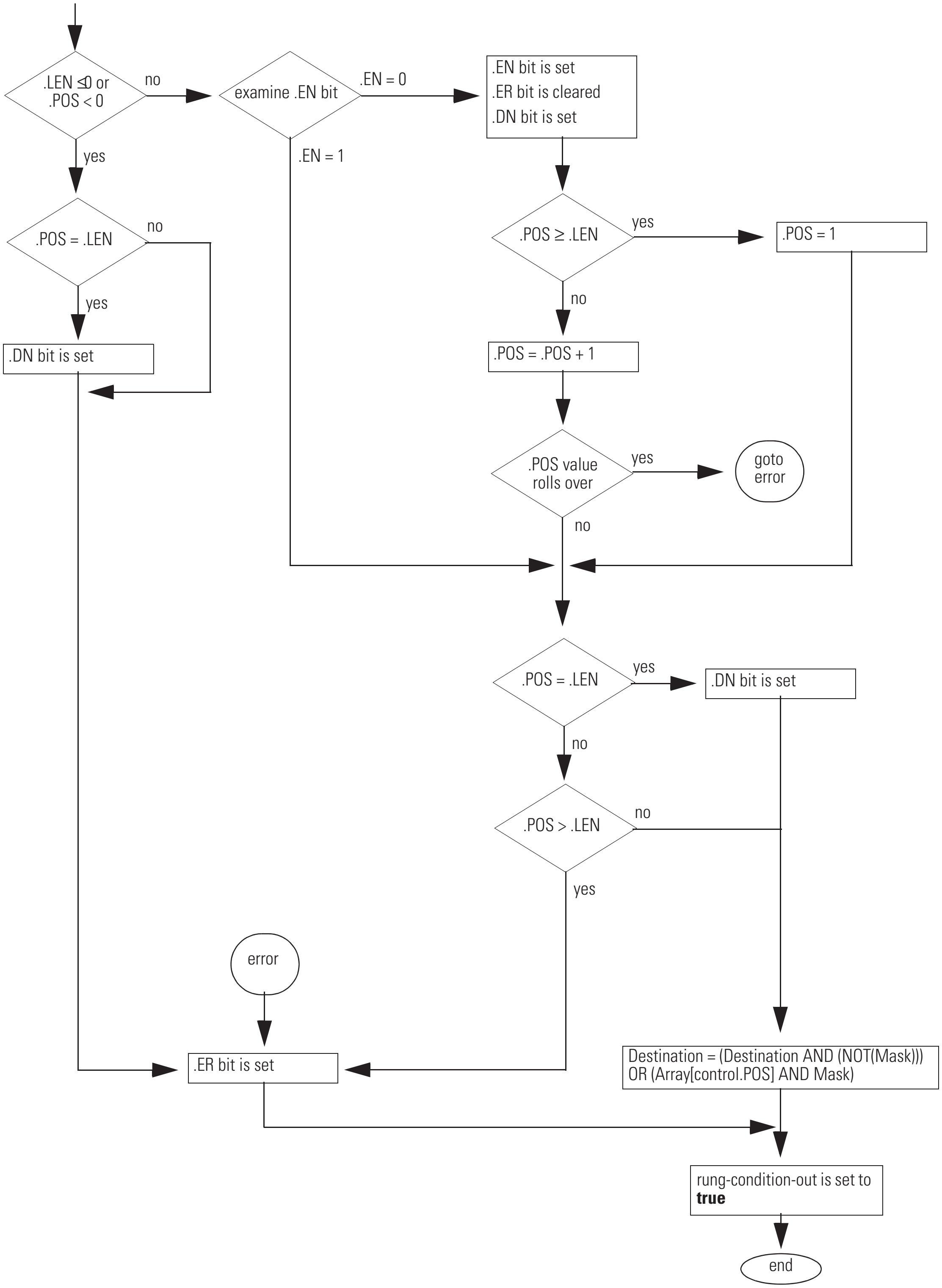
Prefix	Description
16#	Hexadecimal For example; 16#0F0F
8#	Octal For example; 8#16
2#	Binary For example; 2#00110011

Arithmetic Status Flags Not affected

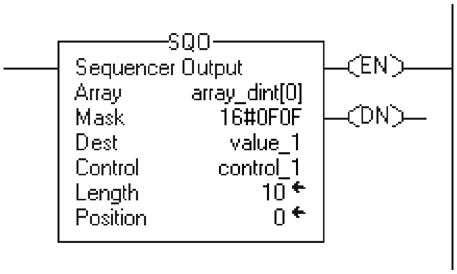
Fault Conditions: None

Execution:

Condition	Relay Ladder Action
Prescan	The .EN bit is set to prevent a false load when the program scan begins. The rung-condition-out is set to false.
Rung-condition-in is false	The .EN bit is cleared. The rung-condition-out is set to false.

Condition	Relay Ladder Action
Rung-condition-in is true	
Postscan	The rung-condition-out is set to false.

Example: When enabled, the SQO instruction increments the position, passes the data at that position in *array_dint* through the mask, and stores the result in *value_1*.



SQO Operand	Example Values (Using INTS Displayed In Binary)
Array	`xxxxxxxx xxxxxxxx xxxx0101 xxxx1010`
Mask	`00000000 00000000 00001111 00001111`
Destination	`xxxxxxxx xxxxxxxx xxxx0101 xxxx1010`

A 0 in the mask means the bit is not compared (designated by xxxx in this example).

Using SQI with SQO

If you pair an SQI instruction with an SQO instruction, make sure that both instructions use the same Control, Length, and Position values.

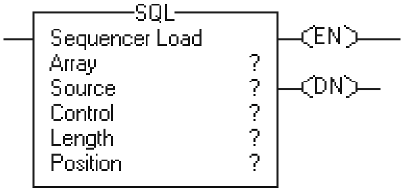
Resetting the Position of SQO

Each time the controller goes from Program to Run mode, the SQO instruction clears (initializes) the .POS value. To reset .POS to the initialization value (.POS = 0), use a RES instruction to clear the position value. This example uses the status of the first-scan bit to clear the .POS value.

Sequencer Load (SQL)

The SQL instruction loads reference conditions into a sequencer array.

Operands:



Relay Ladder

Operand	Type	Format	Description
Array	DINT	Array tag	Sequencer array Specify the first element of the sequencer array **Do not** use CONTROL.POS in the subscript
Source	SINT INT **DINT**	Tag Immediate	Input data to load into the sequencer array
	A SINT or INT tag converts to a DINT value by sign-extension.		
Control	CONTROL	Tag	Control structure for the operation Typically use the same CONTROL as the SQI and SQO instructions
Length	DINT	Immediate	Number of elements in the Array (sequencer table) to load
Position	DINT	Immediate	Current position in the array Initial value is typically 0

CONTROL Structure

Mnemonic	Data Type	Description
.EN	BOOL	The enable bit indicates that the SQL instruction is enabled.
.DN	BOOL	The done bit is set when all the specified elements have been loaded into Array.
.ER	BOOL	The error bit is set when .LEN ≤0, .POS < 0, or .POS > .LEN.
.LEN	DINT	The length specifies the number of steps in the sequencer array.
.POS	DINT	The position identifies the element that the controller is currently manipulating.

Description: When enabled, the SQL instruction increments to the next position in the sequencer array and loads the Source value into that position. If the .DN bit is set or if .POS ≥ .LEN, the instruction sets .POS=1.

Typically use the same CONTROL structure as the SQI and SQO instructions.

IMPORTANT

You must test and confirm that the instruction doesn't change data that you don't want it to change.

The SQL instruction operates on contiguous memory. In some cases, the instruction loads data past the array into other members of the tag. This happens if the length is too big and the tag is a user-defined data type.

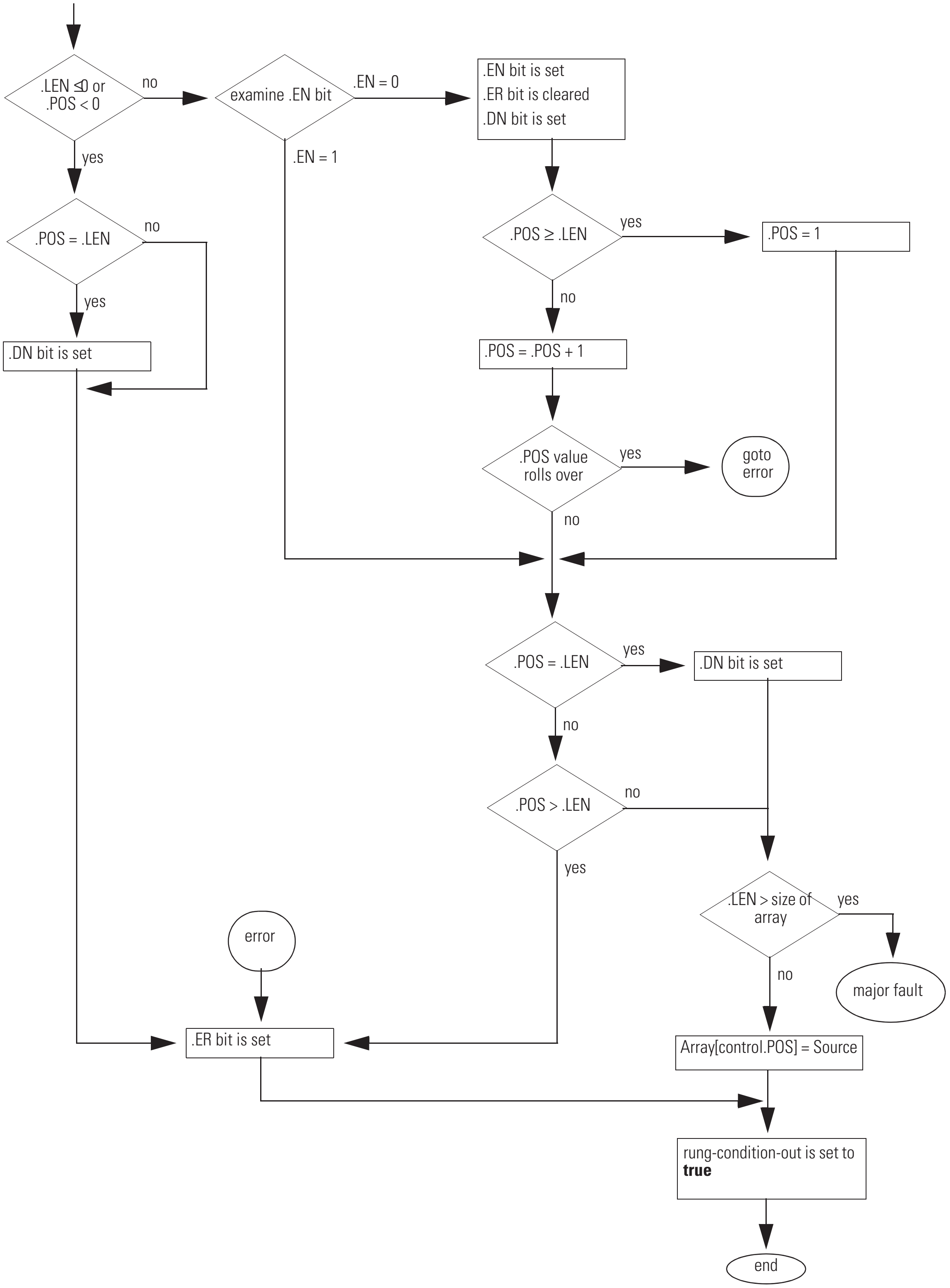
Arithmetic Status Flags: Not affected

Fault Conditions:

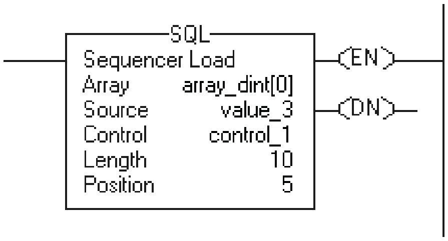
A major fault will occur if	Fault type	Fault code
Length > size of Array	4	20

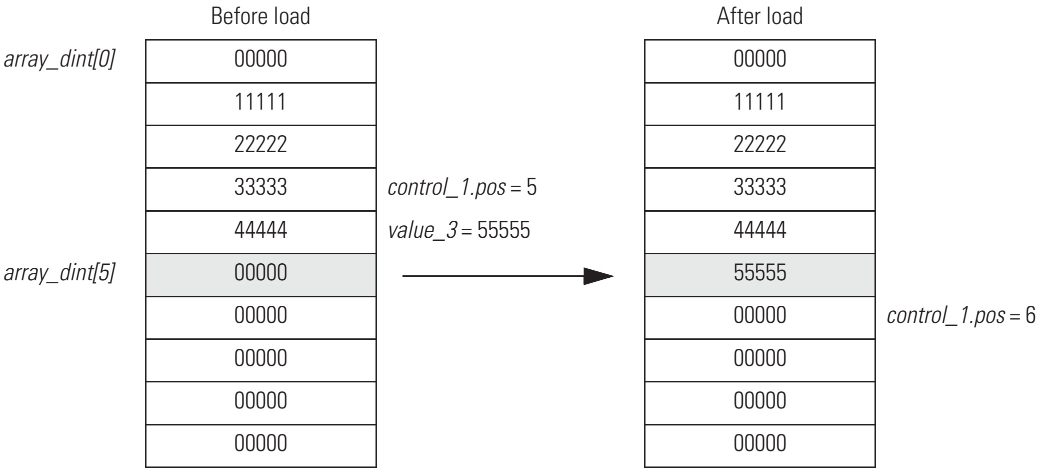
Execution:

Condition	Relay Ladder Action
Prescan	The .EN bit is set to prevent a false load when the program scan begins. The rung-condition-out is set to false.
Rung-condition-in is false	The .EN bit is cleared. The rung-condition-out is set to false.

Condition	**Relay Ladder Action**
Rung-condition-in is true	
Postscan	The rung-condition-out is set to false.

Example: When enabled, the SQL instruction loads *value_3* into the next position in the sequencer array, which is *array_dint[5]* in this example.





	Before load		After load	
array_dint[0]	00000		00000	
	11111		11111	
	22222		22222	
	33333	*control_1.pos* = 5	33333	
	44444	*value_3* = 55555	44444	
array_dint[5]	00000	→	55555	
	00000		00000	*control_1.pos* = 6
	00000		00000	
	00000		00000	
	00000		00000	

Notes:

Program Control Instructions
(JMP, LBL, JSR, RET, SBR, JXR, TND, MCR, UID, UIE, AFI, NOP, EOT, SFP, SFR, EVENT)

Introduction

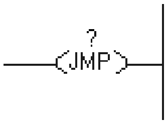
Use the program control instructions to change the flow of logic.

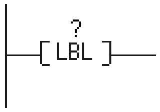
If you want to	Use this instruction	Available in these languages	Page
Jump over a section of logic that does not always need to be executed	JMP LBL	Relay ladder	442
Jump to a separate routine, pass data to the routine, execute the routine, and return results	JSR SBR RET	Relay ladder Function block Structured text	436
Jump to an external routine (SoftLogix5800 controller only)	JXR	Relay ladder	455
Mark a temporary end that halts routine execution	TND	Relay ladder Structured text	458
Disable all the rungs in a section of logic	MCR	Relay ladder	460
Disable user tasks	UID	Relay ladder Structured text	462
Enable user tasks	UIE	Relay ladder Structured text	462
Disable a rung	AFI	Relay ladder	464
Insert a placeholder in the logic	NOP	Relay ladder	465
End a transition for a sequential function chart	EOT	Relay ladder Structured text	466
Pause a sequential function chart	SFP	Relay ladder Structured text	468
Reset a sequential function chart	SFR	Relay ladder Structured text	470
Trigger the execution of an event task	EVENT	Relay ladder Structured text	472

Jump to Label (JMP) Label (LBL)

The JMP and LBL instructions skip portions of ladder logic.

Operands:





Relay Ladder

Operand	Type	Format	Description
JMP instruction			
Label name		Label name	Enter name for associated LBL instruction
LBL instruction			
Label name		Label name	Execution jumps to LBL instruction with referenced label name

Description:

When enabled, the JMP instruction skips to the referenced LBL instruction and the controller continues executing from there. When disabled, the JMP instruction does not affect ladder execution.

The JMP instruction can move ladder execution forward or backward. Jumping forward to a label saves program scan time by omitting a logic segment until it's needed. Jumping backward lets the controller repeat iterations of logic.

Be careful not to jump backward an excessive number of times. The watchdog timer could time out because the controller never reaches the end of the logic, which in turn faults the controller.

ATTENTION

Jumped logic is not scanned. Place critical logic outside the jumped zone.

The LBL instruction is the target of the JMP instruction that has the same label name. **Make sure the LBL instruction is the first instruction on its rung.**

A label name must be unique within a routine. The name can:

- have as many as 40 characters.
- contain letters, numbers, and underscores (_).

Arithmetic Status Flags: Not affected

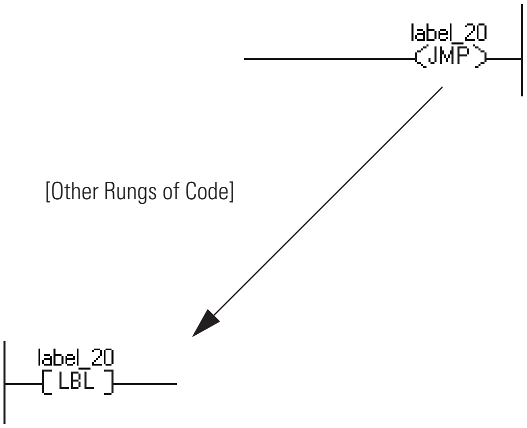
Fault Conditions:

A major fault will occur if	Fault type	Fault code
Label does not exist	4	42

Execution:

Condition:	Relay Ladder Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The rung-condition-out is set to true. Execution jumps to the rung that contains the LBL instruction with the referenced label name.
Postscan	The rung-condition-out is set to false.

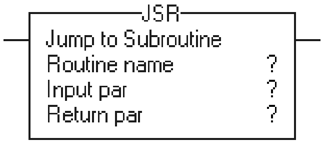
Example: When the JMP instruction is enabled, execution jumps over successive rungs of logic until it reaches the rung that contains the LBL instruction with *label_20*.



Jump to Subroutine (JSR) Subroutine (SBR) Return (RET)

The JSR instruction jumps execution to a different routine. The SBR and RET instructions are optional instructions that exchange data with the JSR instruction.

JSR Operands:



JSR
Jump to Subroutine
Routine name ?
Input par ?
Return par ?

Relay Ladder

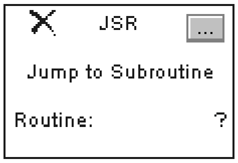
Operand	Type	Format	Description
Routine name	ROUTINE	Name	Routine to execute (that is, subroutine)
Input parameter	BOOL SINT INT DINT REAL Structure	Immediate Tag Array tag	Data from this routine that you want to copy to a tag in the subroutine • Input parameters are optional. • Enter multiple input parameters, if needed.
Return parameter	BOOL SINT INT DINT REAL Structure	Tag Array tag	Tag in this routine to which you want to copy a result of the subroutine • Return parameters are optional. • Enter multiple return parameters, if needed.

JSR Operands Continued:

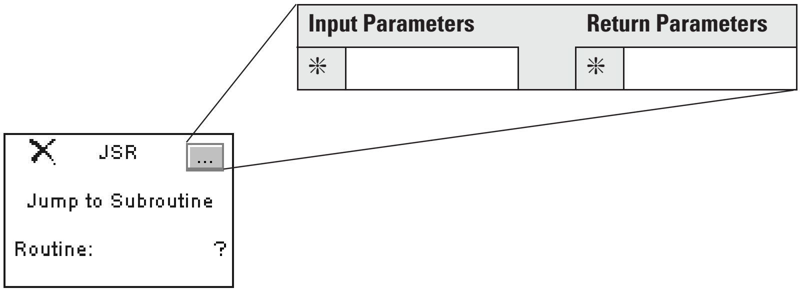
```
JSR(RoutineName,InputCount,
InputPar,ReturnPar);
```

Structured Text

Operand	Type	Format	Description
Routine name	ROUTINE	Name	Routine to execute (that is, subroutine)
Input count	SINT INT DINT REAL	Immediate	Number of input parameters
Input parameter	BOOL SINT INT DINT REAL Structure	Immediate Tag Array tag	Data from this routine that you want to copy to a tag in the subroutine • Input parameters are optional. • Enter multiple input parameters, if needed.
Return parameter	BOOL SINT INT DINT REAL Structure	Tag Array tag	Tag in this routine to which you want to copy a result of the subroutine • Return parameters are optional. • Enter multiple return parameters, if needed.



Function Block



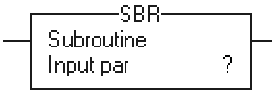
The operands are the same as those for the relay ladder JSR instruction.

ATTENTION

For each parameter in a SBR or RET instruction, use the same data type (including any array dimensions) as the corresponding parameter in the JSR instruction. Using different data types may produce unexpected results.

SBR Operands:

The SBR instruction must be the first instruction in a relay ladder or structured text routine.



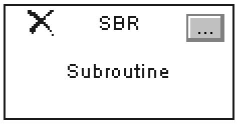
Relay Ladder

Operand	Type	Format	Description
Input parameter	BOOL SINT INT DINT REAL Structure	Tag Array tag	Tag in this routine into which you want to copy the corresponding input parameter from the JSR instruction

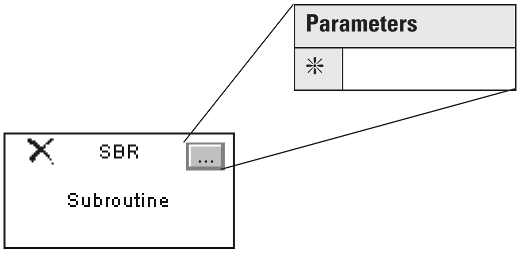
```
SBR(InputPar);
```

Structured Text

The operands are the same as those for the relay ladder SBR instruction.



Function Block



The operands are the same as those for the relay ladder SBR instruction.

RET Operands:

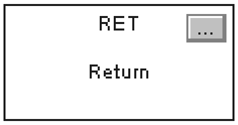
Relay Ladder

Operand	Type	Format	Description
Return parameter	BOOL SINT INT DINT REAL structure	Immediate Tag Array tag	Data from this routine that you want to copy to the corresponding return parameter in the JSR instruction

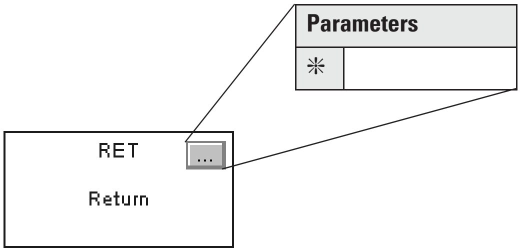
```
RET(ReturnPar);
```

Structured Text

The operands are the same as those for the relay ladder RET instruction.



Function Block



The operands are the same as those for the relay ladder RET instruction.

Description: The JSR instruction initiates the execution of the specified routine, which is referred to as a subroutine.

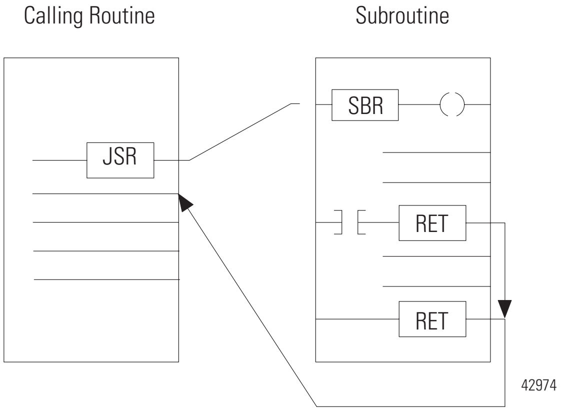
- The subroutine executes one time.
- After the subroutine executes, logic execution returns to the routine that contains the JSR instruction.

Use these guidelines to program a jump to a subroutine.

IMPORTANT Do not use a JSR instruction to call (execute) the main routine.

- You can put a JSR instruction in the main routine or any other routine.
- If you use a JSR instruction to call the main routine and then put a RET instruction in the main routine, a major fault occurs (type 4, code 31).

The following diagram illustrates how the instructions operate.



JSR

1. If you want to copy data to a tag in the subroutine, enter an input parameter.
2. If you want to copy a result of the subroutine to a tag in this routine, enter a return parameter.
3. Enter as many input and return parameters as you need.

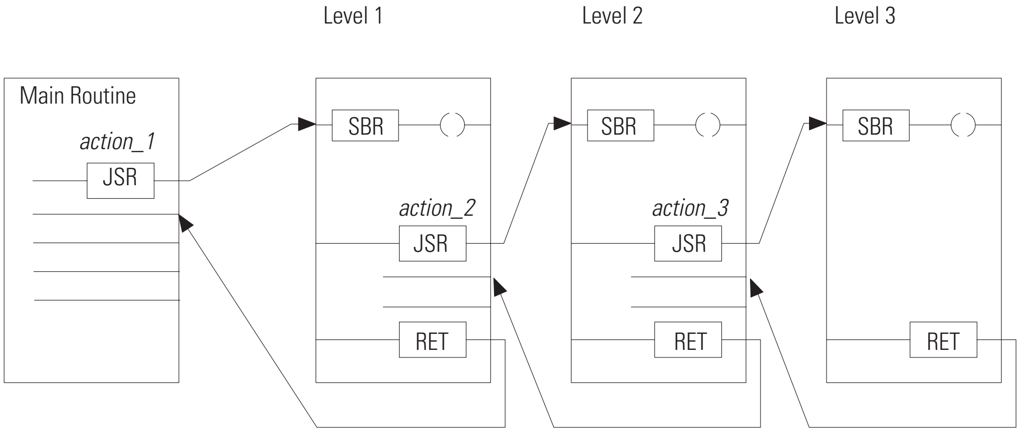
SBR

1. If the JSR instruction has an input parameter, enter an SBR instruction.
2. Place the SBR instruction as the first instruction in the routine.
3. For each input parameter in the JSR instruction, enter the tag into which you want to copy the data.

RET

1. If the JSR instruction has a return parameter, enter an RET instruction.
2. Place the RET instruction as the last instruction in the routine.
3. For each return parameter in the JSR instruction, enter a return parameter to send to the JSR instruction.
4. In a ladder routine, place additional RET instructions to exit the subroutine based on different input conditions, if required. (Function block routines only permit one RET instruction.)

There are no restrictions, other than controller memory, on the number of nested routines you can have or the number of parameters you pass or return.



Arithmetic Status Flags: Arithmetic status flags are affected.

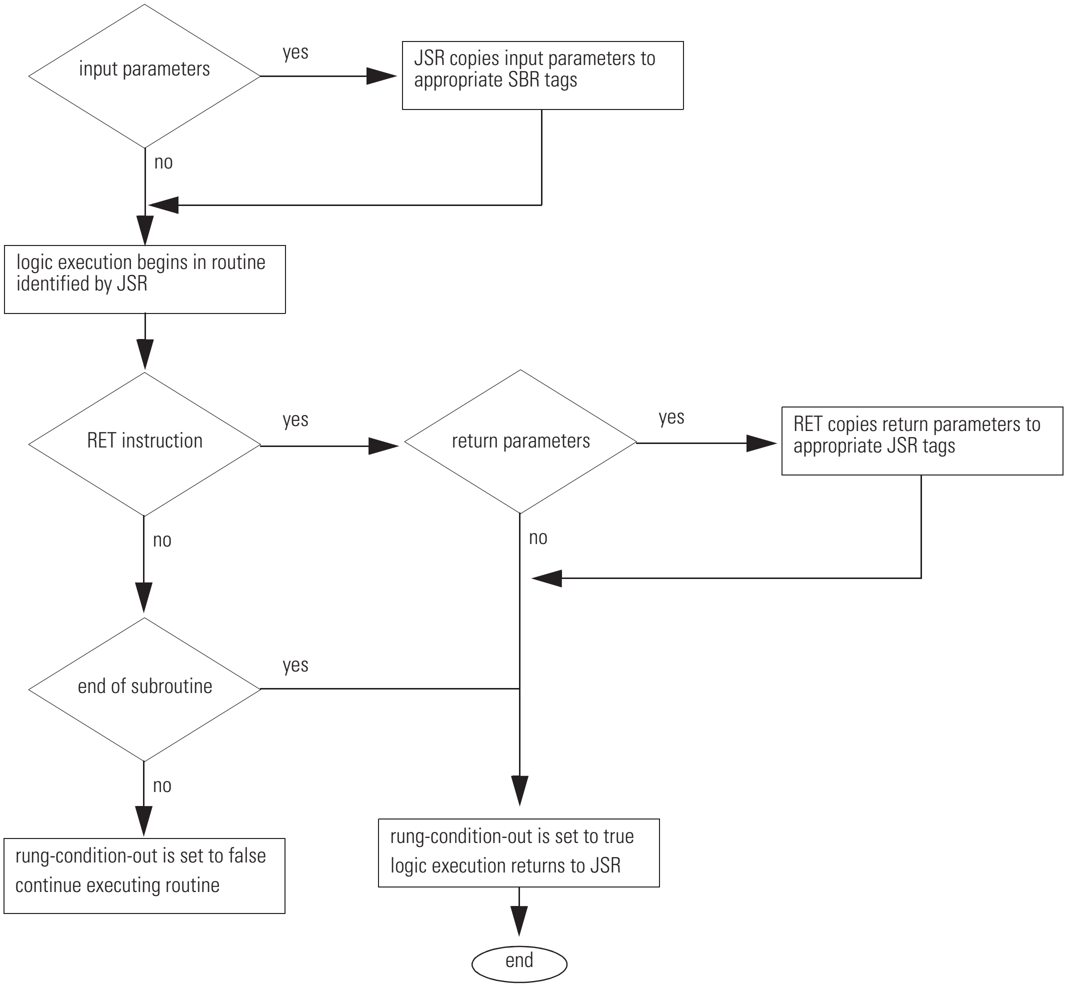
Fault Conditions:

A Major Fault Will Occur If	Fault Type	Fault Code
JSR instruction has fewer input parameters than SBR instruction	4	31
JSR instruction jumps to a fault routine	4 or user-supplied	0 or user-supplied
RET instruction has fewer return parameters than JSR instruction	4	31
Main routine contains a RET instruction	4	31

Execution:

Relay Ladder and Structured Text

Condition	Relay Ladder Action	Structured Text Action
Prescan	The controller executes all subroutines regardless of rung condition. To ensure that all rungs in the subroutine are prescanned, the controller ignores RET instructions. (that is, RET instructions do not exit the subroutine.) • Release 6.*x* and earlier, input and return parameters are passed. • Release 7.*x* and later, input and return parameters are not passed. If recursive calls exist to the same subroutine, the subroutine is prescanned only the first time. If multiple calls exist (non-recursive) to the same subroutine, the subroutine is prescanned each time. The rung-condition-out is set to false (relay ladder only).	
Rung-condition-in is false to the JSR instruction	The subroutine **does not** execute. Outputs in the subroutine remain in their last state. The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.

Condition	Relay Ladder Action	Structured Text Action
Instruction execution		
Postscan	Same action as prescan described above.	Same action as prescan described above.

Function Block

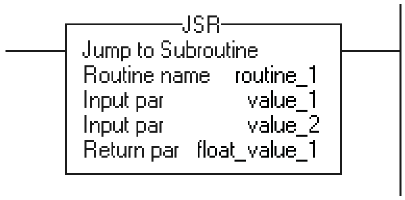
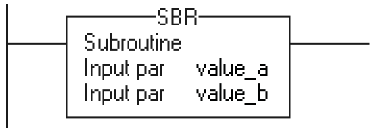
Condition	**Action**
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
Normal execution	1. If the routine contains an SBR instruction, the controller first executes the SBR instruction. 2. The controller latches all data values in IREFs. 3. The controller executes the other function blocks in the order that is determined by their wiring. This includes other JSR instructions. 4. The controller writes outputs in OREFs. 5. If the routine contains an RET instruction, the controller executes the RET instruction last.
Postscan	The subroutine is called. If the routine is an SFC routine, the routine in initialized the same as it is during prescan.

Example 1: The JSR instruction passes *value_1* and *value_2* to *routine_1*.

The SBR instruction receives *value_1* and *value_2* from the JSR instruction and copies those values to *value_a* and *value_b*, respectively. Logic execution continues in this routine.

The RET instruction sends *float_a* to the JSR instruction. The JSR instruction receives *float_a* and copies the value to *float_value_1*. Logic execution continues with the next instruction following the JSR instruction.

Relay Ladder

Routine	Program
Main routine	 JSR Jump to Subroutine Routine name routine_1 Input par value_1 Input par value_2 Return par float_value_1
Subroutine	 SBR Subroutine Input par value_a Input par value_b *[Other rungs of code]*  RET Return Return par float_a

Structured Text

Routine	Program
Main routine	`JSR(routine_1,2,value_1,value_2,float_value_1);`
Subroutine	

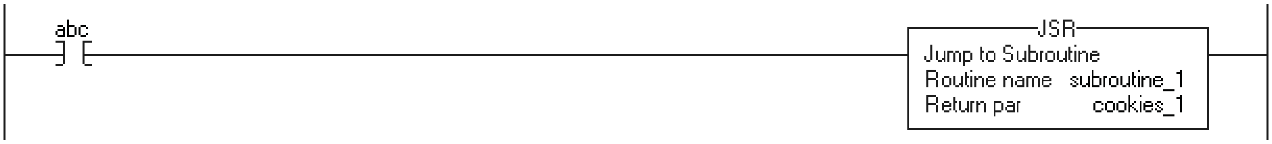
```
SBR(value_a,value_b);

     <statements>;

RET(float_a);
```

Example 2:

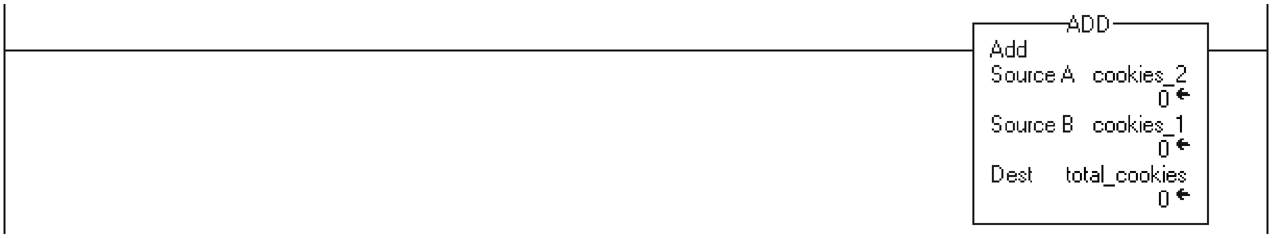
Relay Ladder

MainRoutine

When *abc* is on, *subroutine_1* executes, calculates the number of cookies, and places a value in *cookies_1*.

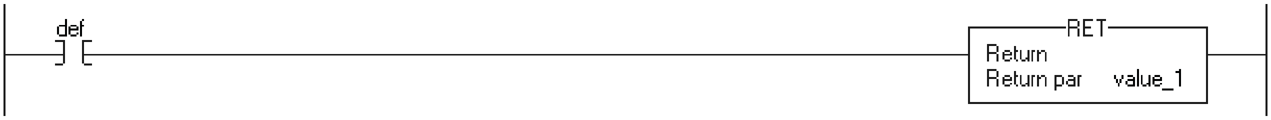


Adds the value in *cookies_1* to *cookies_2* and stores the result in *total_cookies*.

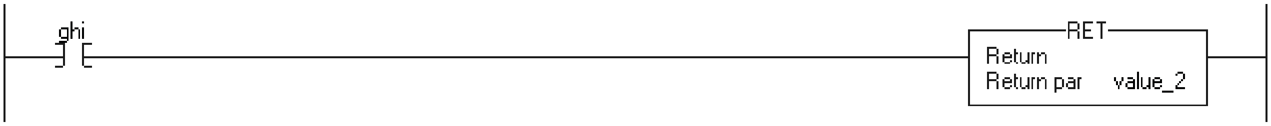


Subroutine_1

When *def* is on, the RET instruction returns *value_1* to the JSR *cookies_1* parameter and the rest of the subroutine is not scanned.



When *def* is off (previous rung) and *ghi* is on, the RET instruction returns *value_2* to the JSR *cookies_1* parameter and the rest of the subroutine is not scanned.

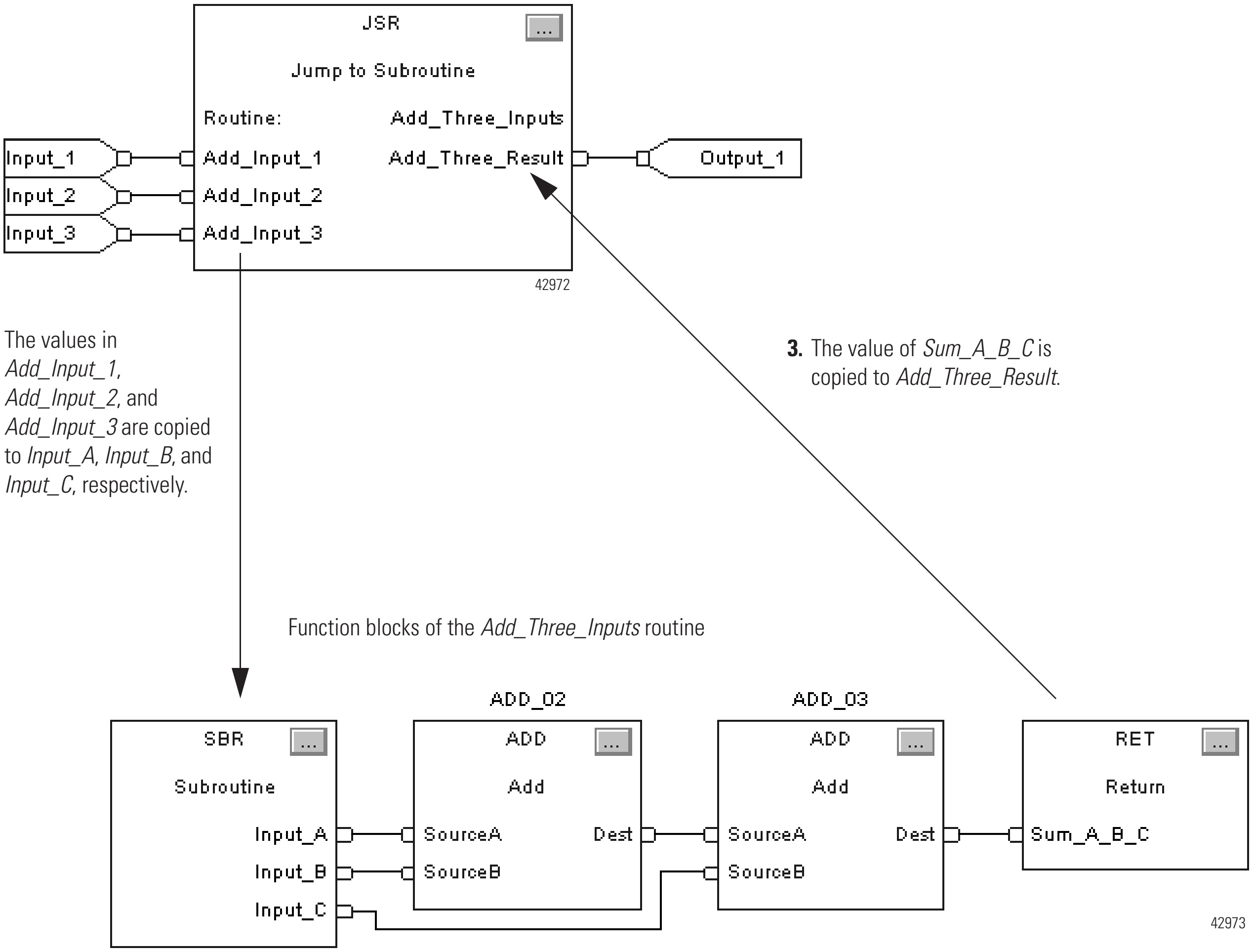


When both *def* and *ghi* are off (previous rungs), the RET instruction returns *value_3* to the JSR *cookies_1* parameter.

Example 3:

Function Block

JSR instruction in *Routine_A*



1. The values in *Add_Input_1*, *Add_Input_2*, and *Add_Input_3* are copied to *Input_A*, *Input_B*, and *Input_C*, respectively.

3. The value of *Sum_A_B_C* is copied to *Add_Three_Result*.

Function blocks of the *Add_Three_Inputs* routine

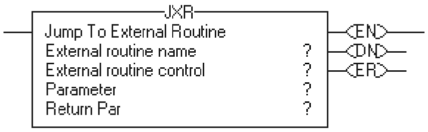
2. The ADD instructions add *Input_A*, *Input_B*, and *Input_C* and place the result in *Sum_A_B_C*.

Jump to External Routine (JXR)

The JXR instruction executes an external routine. This instruction is only supported by the SoftLogix5800 controllers.

Operands:

Relay Ladder



.

Operand	Type	Format	Description
External routine name	ROUTINE	Name	External routine to execute
External routine control	EXT_ROUTINE_ CONTROL	Tag	Control structure
Parameter	BOOL SINT INT DINT REAL structure	Immediate Tag Array tag	Data from this routine that you want to copy to a variable in the external routine • Parameters are optional. • Enter multiple parameters, if needed. • You can have as many as 10 parameters.
Return parameter	BOOL SINT INT DINT REAL	Tag	Tag in this routine to which you want to copy a result of the external routine • The return parameter is optional. • You can have only one return parameter

EXT_ROUTINE_CONTROL Structure

Mnemonic	Data Type	Description	Implementation
ErrorCode	SINT	If an error occurs, this value identifies the error. Valid values are from 0-255.	There are no predefined error codes. The developer of the external routine must provide the error codes.
NumParams	SINT	This value indicates the number of parameters associated with this instruction.	Display only - this information is derived from the instruction entry.
ParameterDefs	EXT_ROUTINE_ PARAMETERS[10]	This array contains definitions of the parameters to pass to the external routine. The instruction can pass as many as 10 parameters.	Display only - this information is derived from the instruction entry.
ReturnParamDef	EXT_ROUTIN_ PARAMETERS	This value contains definitions of the return parameter from the external routine. There is only one return parameter.	Display only - this information is derived from the instruction entry.
EN	BOOL	When set, the enable bit indicates that the JXR instruction is enabled.	The external routine sets this bit.
ReturnsValue	BOOL	If set, this bit indicates that a return parameter was entered for the instruction. If cleared, this bit indicates that no return parameter was entered for the instruction.	Display only - this information is derived from the instruction entry.
DN	BOOL	The done bit is set when the external routine has executed once to completion.	The external routine sets this bit.
ER	BOOL	The error bit is set if an error occurs. The instruction stops executing until the program clears the error bit.	The external routine sets this bit.
FirstScan	BOOL	This bit identifies whether this is the first scan after switching the controller to Run mode. Use FirstScan to initialize the external routine, if needed.	The controller sets this bit to reflect scan status.
EnableOut	BOOL	Enable output.	The external routine sets this bit.
EnableIn	BOOL	Enable input.	The controller sets this bit to reflect rung-condition-in. The instruction executes regardless of rung condition. The developer of the external routine should monitor this status and act accordingly.
User1	BOOL	These bits are available for the user. The controller does not initialize these bits.	Either the external routine or the user program can set these bits.
User0	BOOL		
ScanType1	BOOL	These bits identify the current scan type: **Bit Values:** **Scan Type:** 00 Normal 01 Pre Scan 10 Post Scan (not applicable to relay ladder programs)	The controller sets these bits to reflect scan status.
ScanType0	BOOL		

Description:

Use the Jump to External Routine (JXR) instruction to call the external routine from a ladder routine in your project. The JXR instruction supports multiple parameters so you can pass values between the ladder routine and the external routine.

The JXR instruction is similar to the Jump to Subroutine (JSR) instruction. The JXR instruction initiates the execution of the specified external routine:

- The external routine executes one time.
- After the external routine executes, logic execution returns to the routine that contains the JXR instruction.

Arithmetic Status Flags:

Arithmetic status flags are not affected.

Fault Conditions:

A major fault will occur if	Fault Type	Fault Code:
• an exception occurs in the external routine DLL. • the DLL could not be loaded. • the entry point was not found in the DLL.	4	88

Execution:

The JXR can be synchronous or asynchronous depending on the implementation of the DLL. The code in the DLL also determines how to respond to scan status, rung-condition-in status, and rung-condition-out status.

For more information on using the JXR instruction and creating external routines, see the SoftLogix5800 System User Manual, publication 1789-UM002.

Temporary End (TND)

The TND instruction acts as a boundary.

Operands:

—(TND)—

Relay Ladder Operands

None

```
TND();
```

Structured Text

None

You must enter the parentheses () after the instruction mnemonic, even though there are no operands.

Description: When enabled, the TND instruction lets the controller execute logic only up to this instruction.

When enabled, the TND instruction acts as the end of the routine. When the controller scans a TND instruction, the controller moves to the end of the current routine. If the TND instruction is in a subroutine, control returns to the calling routine. If the TND instruction is in a main routine, control returns to the next program within the current task.

Arithmetic Status Flags: Not affected

Fault Conditions: None

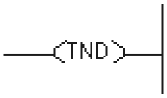
Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The current routine terminates.	The current routine terminates.
Postscan	The rung-condition-out is set to false.	No action taken.

Example: You can use the TND instruction when debugging or troubleshooting to execute logic up to a certain point. Progressively move the TND instruction through the logic as you debug each new section.

When the TND instruction is enabled, the controller stops scanning the current routine.

Relay Ladder



Structured Text

```
TND();
```

Master Control Reset (MCR)

The MCR instruction, used in pairs, creates a program zone that can disable all rungs within the MCR instructions.

Operands:

—(MCR)—

Relay Ladder

None

Description:

When the MCR zone is enabled, the rungs in the MCR zone are scanned for normal true or false conditions. When disabled, the controller still scans rungs within an MCR zone, but scan time is reduced because non-retentive outputs in the zone are disabled. The rung-condition-in is false for all the instructions inside of the disabled MCR zone.

When you program an MCR zone, note that:

- you must end the zone with an unconditional MCR instruction.
- you cannot nest one MCR zone within another.
- do not jump into an MCR zone. If the zone is false, jumping into the zone activates the zone from the point to which you jumped to the end of the zone.
- if an MCR zone continues to the end of the routine, you do not have to program an MCR instruction to end the zone.

The MCR instruction is not a substitute for a hard-wired master control relay that provides emergency-stop capability. You should still install a hard-wired master control relay to provide emergency I/O power shutdown.

ATTENTION

Do not overlap or nest MCR zones. Each MCR zone must be separate and complete. If they overlap or nest, unpredictable machine operation could occur with possible damage to equipment or injury to personnel.

Place critical operations outside the MCR zone. If you start instructions such as timers in a MCR zone, instruction execution stops when the zone is disabled and the timer is cleared.

Arithmetic Status Flags:

Not affected

Fault Conditions:

None

Execution:

Condition	Relay Ladder Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false. The instructions in the zone are scanned, but the rung-condition-in is false and non-retentive outputs in the zone are disabled.
Rung-condition-in is true	The rung-condition-out is set to true. The instructions in the zone are scanned normally.
Postscan	The rung-condition-out is set to false.

Example: When the first MCR instruction is enabled (*input_1*, *input_2*, and *input_3* are set), the controller executes the rungs in the MCR zone (between the two MCR instructions) and sets or clears outputs, depending on input conditions.

When the first MCR instruction is disabled (*input_1*, *input_2*, and *input_3* are not all set), the controller executes the rungs in the MCR zone (between the two MCR instructions) and the rung-condition-in goes false for all the rungs in the MCR zone, regardless of input conditions.

User Interrupt Disable (UID) User Interrupt Enable (UIE)

The UID instruction and the UIE instruction work together to prevent a small number of critical rungs from being interrupted by other tasks.

Operands:

—<UID>— —<UIE>—

Relay Ladder

None

```
UID();
UIE();
```

Structured Text

None

You must enter the parentheses () after the instruction mnemonic, even though there are no operands.

Description:

When the rung-condition-in is true, the:

- UID instruction prevents higher-priority tasks from interrupting the current task but **does not** disable execution of a fault routine or the Controller Fault Handler.
- UIE instruction enables other tasks to interrupt the current task.

Follow these steps to prevent a series of rungs from being interrupted.

1. Limit the number of rungs that you **do not** want interrupted to as few as possible.

   Disabling interrupts for a prolonged period of time can produce communication loss.

2. Above the first rung that you *do not* want interrupted, enter a rung and a UID instruction.

3. After the last rung in the series that you **do not** want interrupted, enter a rung and a UIE instruction.

4. If required, you can nest pairs of UID/UIE instructions.

Arithmetic Status Flags: Not affected

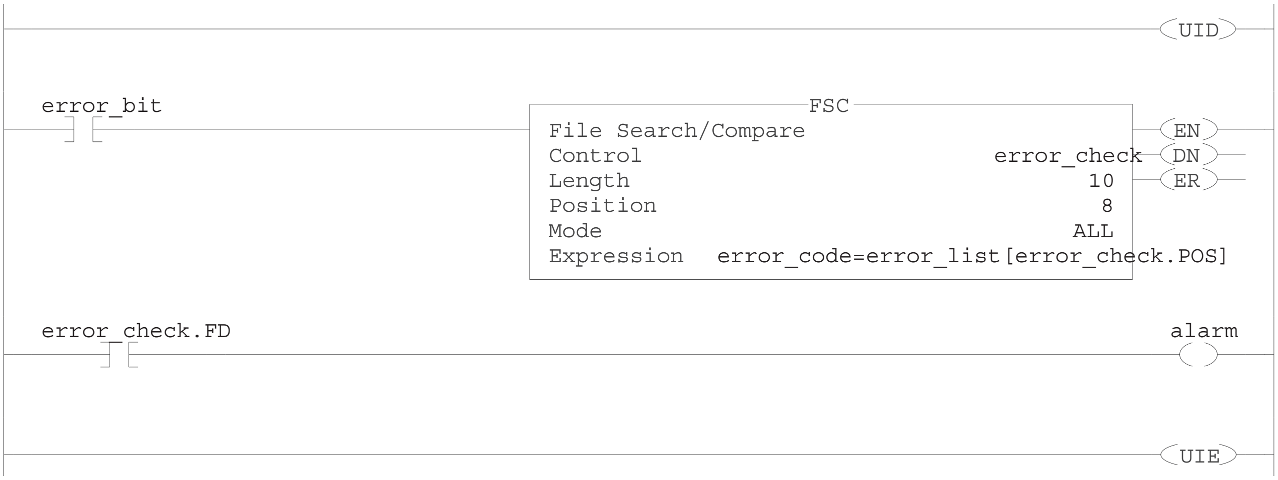
Fault Conditions: None

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The UID instruction prevents interruption by higher-priority tasks. The UIE instruction enables interruption by higher-priority tasks.	
Postscan	The rung-condition-out is set to false.	No action taken.

Example: When an error occurs (*error_bit* is on), the FSC instruction checks the error code against a list of critical errors. If the FSC instruction finds that the error is critical (error_check.FD is on), an alarm is annunciated. The UID and UIE instructions prevent any other tasks from interrupting the error checking and alarming.

Relay Ladder



Structured Text

```
UID();

    <statements>

UIE();
```

Always False Instruction (AFI)

The AFI instruction sets its rung-condition-out to false.

Operands:

Relay Ladder

None

Description: The AFI instruction sets its rung-condition-out to false.

Arithmetic Status Flags: Not affected

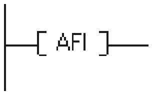
Fault Conditions: None

Execution:

Condition	Relay Ladder Action:
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The rung-condition-out is set to false.
Postscan	The rung-condition-out is set to false.

Example: Use the AFI instruction to temporarily disable a rung while you are debugging a program.

When enabled, the AFI disables all the instructions on this rung.



No Operation (NOP)

The NOP instruction functions as a placeholder.

Operands:

-[NOP]-

Relay Ladder

None

Description: You can place the NOP instruction anywhere on a rung. When enabled the NOP instruction performs no operation. When disabled, the NOP instruction performs no operation.

Arithmetic Status Flags: Not affected

Fault Conditions: None

Execution:

Condition	Relay Ladder Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Example This instruction is useful for locating unconditional branches when you place the NOP instruction on the branch.

The NOP instruction bypasses the XIC instruction to enable the output.

End of Transition (EOT)

The EOT instruction returns a boolean state to an SFC transition.

Operands:

Relay Ladder

Operand	Type	Format	Description
Data bit	BOOL	Tag	State of the transition (0=executing, 1=completed)

```
EOT(data_bit);
```

Structured Text

The operands are the same as those for the relay ladder EOT instruction.

Description:

Because the EOT instruction returns a boolean state, multiple SFC routines can share the same routine that contains the EOT instruction. If the calling routine is not a transition, the EOT instruction acts as a TND instruction (see page 458).

The Logix implementation of the EOT instruction differs from that in a PLC-5 controller. In a PLC-5 controller, the EOT instruction has no parameters. Instead, the PLC-5 EOT instruction returns rung condition as its state. In a Logix controller, the return parameter returns the transition state since rung condition is not available in all Logix programming languages.

Arithmetic Status Flags:

Not affected

Fault Conditions:

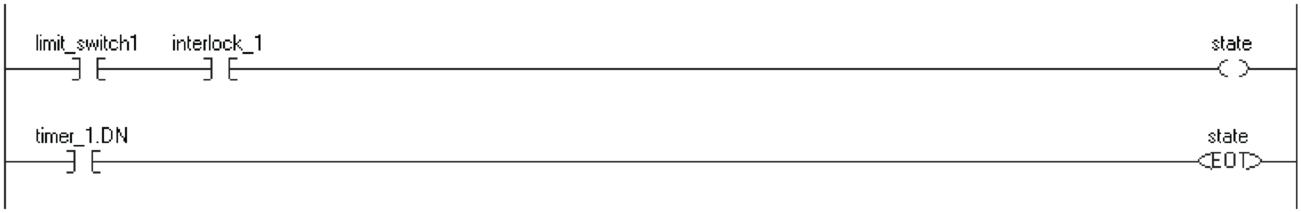
None

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction returns the data bit value to the calling routine.	
Postscan	The rung-condition-out is set to false.	No action taken.

Example: When both *limit_switch1* and *interlock_1* are set, set state. After *timer_1* completes, EOT returns the value of *state* to the calling routine.

Relay Ladder



Structured Text

```
state := limit_switch1 AND interlock_1;


IF timer_1.DN THEN

     EOT(state);

END_IF;
```

SFC Pause (SFP)

The SFP instruction pauses an SFC routine.

Operands:

Relay Ladder

Operand	Type	Format	Description
SFCRoutine Name	ROUTINE	Name	SFC routine to pause
TargetState	DINT	Immediate Tag	Select one: Executing (or enter 0) Paused (or enter 1)

Structured Text

```
SFP(SFCRoutineName,
    TargetState);
```

The operands are the same as those for the relay ladder SFP instruction.

Description:

The SFP instruction lets you pause an executing SFC routine. If an SFC routine is in the paused state, use the SFP instruction again to change the state and resume execution of the routine.

Also, use the SFP instruction to resume SFC execution after using an SFR instruction (see page 470) to reset an SFC routine.

Arithmetic Status Flags:

Not affected

Fault Conditions:

A major fault will occur if	Fault type	Fault code
Routine type is not an SFC routine	4	85

Execution:

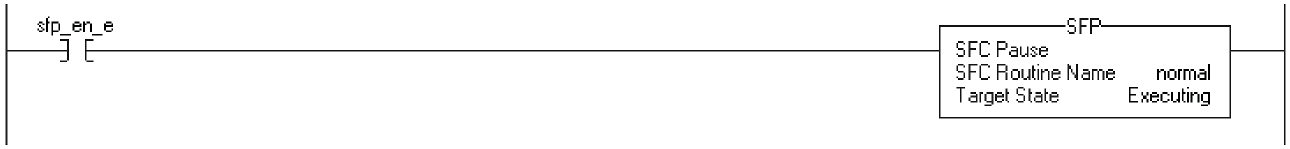
Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction pauses or resumes execution of the specified SFC routine.	
Postscan	The rung-condition-out is set to false.	No action taken.

Example: If *sfc_en_p* is set, pause the SFC routine named *normal*. Restart the SFC when *sfc_en_e* is set.

Relay Ladder

Pause the SFC routine.

Resume executing the SFC routine.



Structured Text

Pause the SFC routine:

```
IF (sfp_en_p) THEN

     SFP(normal,paused);

     sfp_en_p := 0;

END_IF;
```

Resume executing the SFC routine:

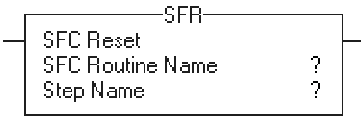
```
IF (sfp_en_e) THEN

     SFP(normal,executing);

     sfp_en_e := 0;

END_IF;
```

SFC Reset (SFR)

The SFR instruction resets the execution of a SFC routine at a specified step.

Operands:

Relay Ladder Operands



Operand	Type	Format	Description
SFCRoutine Name	ROUTINE	Name	SFC routine to reset
Step Name	SFC_STEP	Tag	Target step where to resume execution

Structured Text

`SFR(SFCRoutineName,StepName);`

The operands are the same as those for the relay ladder SFR instruction.

Description:

When the SFR instruction is enabled:

- in the specified SFC routine, all stored actions stop executing (reset).
- the SFC begins executing at the specified step.

If the target step is 0, the chart will be reset to its initial step

The Logix implementation of the SFR instruction differs from that in a PLC-5 controller. In the PLC-5 controller, the SFR executed when the rung condition was true. After reset, the SFC would remain paused until the rung containing the SFR became false. This allowed the execution following a reset to be delayed. This pause/un-pause feature of the PLC-5 SFR instruction was decoupled from the rung condition and moved into the SFP instruction.

Arithmetic Status Flags:

Not affected

Fault Conditions:

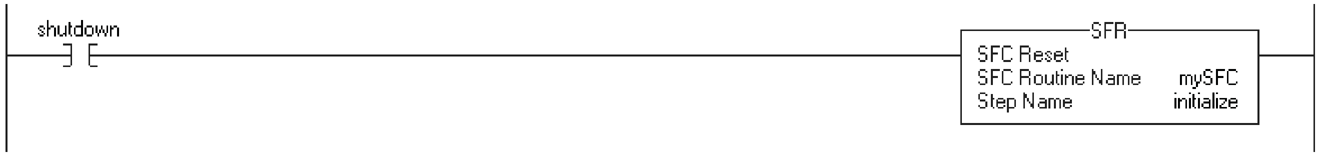
A major fault will occur if	Fault type	Fault code
Routine type is not an SFC routine	4	85
Specified target step does not exist in the SFC routine	4	89

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction resets the specified SFC routine.	The instruction resets the specified SFC routine.
Postscan	The rung-condition-out is set to false.	No action taken.

Example: If a specific condition occurs (*shutdown* is set), restart the SFC at step *initialize*.

Relay Ladder



Structured Text

```
IF shutdown THEN

    SFR(mySFC,initialize);

END_IF;
```

Trigger Event Task (EVENT)

The EVENT instruction triggers one execution of an event task.

Operands:

Relay Ladder

Operand	Type	Format	Description
Task	TASK	Name	Event task to execute The instruction lets you choose other types of tasks, but it does not execute them.

```
EVENT(task_name);
```

Structured Text

The operands are the same as those for the relay ladder EVENT instruction.

Description:

Use the EVENT instruction to programmatically execute an event task:

- each time the instruction executes, it triggers the specified event task.
- make sure that you give the event task enough time to complete its execution before you trigger it again. If not, an overlap occurs.
- if you execute an EVENT instruction while the event task is already executing, the controller increments the overlap counter but it does not trigger the event task.

Programmatically Determine if an EVENT Instruction Triggered a Task

To determine if an EVENT instruction triggered an event task, use a Get System Value (GSV) instruction to monitor the Status attribute of the task.

Status Attribute of the TASK Object

<table>
<tr><th>Attribute</th><th>Data Type</th><th>Instruction</th><th colspan="2">Description</th></tr>
<tr><td rowspan="5">Status</td><td rowspan="5">DINT</td><td rowspan="5">GSV
SSV</td><td colspan="2">Provides status information about the task. Once the controller sets a bit, you must manually clear the bit to determine if another fault of that type occurred.</td></tr>
<tr><td>To determine if</td><td>Examine this bit</td></tr>
<tr><td>An EVENT instruction triggered the task (event task only).</td><td>0</td></tr>
<tr><td>A time-out triggered the task (event task only).</td><td>1</td></tr>
<tr><td>An overlap occurred for this task.</td><td>2</td></tr>
</table>

The controller does not clear the bits of the Status attribute once they are set.

- To use a bit for new status information, you must manually clear the bit.
- Use a Set System Value (SSV) instruction to set the attribute to a different value.

Arithmetic Status Flags: Not affected

Fault Conditions: None

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes.	N/A
	The rung-condition-out is set to true.	
EnableIn is set	N/A	EnableIn is always set.
		The instruction executes.
Instruction execution	The instruction triggers one execution of the specified event task	
Postscan	The rung-condition-out is set to false.	No action taken.

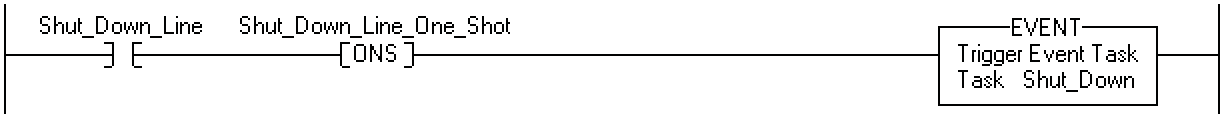
Example 1: A controller uses multiple programs but a common shut down procedure. Each program uses a program-scoped tag named *Shut_Down_Line* that turns on if the program detects a condition that requires a shut down. The logic in each program executes as follows:

If *Shut_Down_Line* = on (conditions require a shut down) then

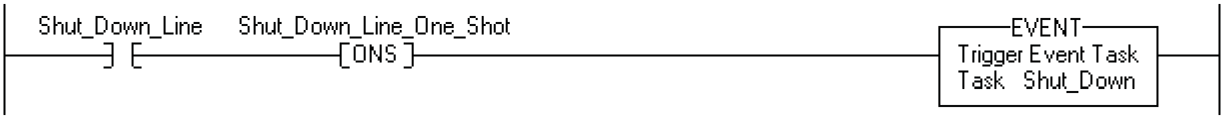
Execute the *Shut_Down* task one time

Relay Ladder

Program A



Program B



Structured Text

Program A

```
IF Shut_Down_Line AND NOT Shut_Down_Line_One_Shot THEN
  EVENT (Shut_Down);
END_IF;
Shut_Down_Line_One_Shot := Shut_Down_Line;
```

Program B

```
IF Shut_Down_Line AND NOT Shut_Down_Line_One_Shot THEN
  EVENT (Shut_Down);
END_IF;
Shut_Down_Line_One_Shot := Shut_Down_Line;
```

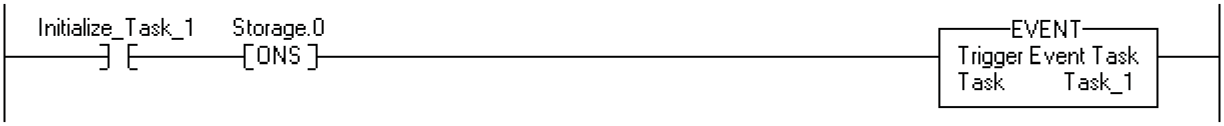
Example 2: The following example uses an EVENT instruction to initialize an event task. (Another type of event normally triggers the event task.)

Continuous task

If *Initialize_Task_1* = 1 then

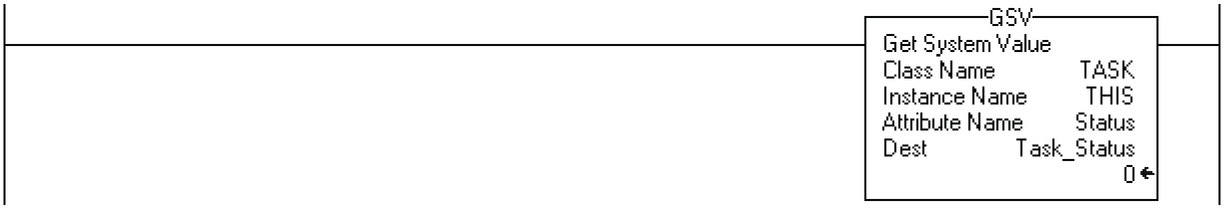
The ONS instruction limits the execution of the EVENT instruction to one scan.

The EVENT instruction triggers an execution of *Task_1* (event task).



Task_1 (event task)

The GSV instruction sets *Task_Status* (DINT tag) = Status attribute for the event task. In the Instance Name attribute, THIS means the TASK object for the task that the instruction is in (that is, *Task_1*).



If *Task_Status*.0 = 1 then an EVENT instruction triggered the event task (that is, when the continuous task executes its EVENT instruction to initialize the event task).

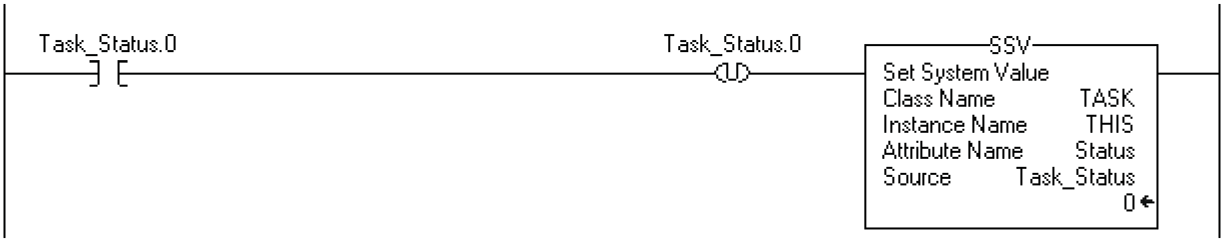
The RES instruction resets a counter that the event task uses.

The controller does not clear the bits of the Status attribute once they are set. To use a bit for new status information, you must manually clear the bit.

If *Task_Status*.0 = 1 then clear that bit.

The OTU instruction sets *Task_Status*.0 = 0.

The SSV instruction sets the Status attribute of THIS task (*Task_1*) = *Task_Status*. This includes the cleared bit.



Notes:

For/Break Instructions

(FOR, FOR...DO, BRK, EXIT, RET)

Introduction

Use the FOR instruction to repeatedly call a subroutine. Use the BRK instruction to interrupt the execution of a subroutine.

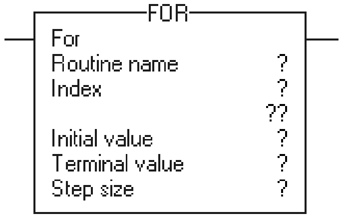
If you want to	Use this instruction	Available in these languages	Page
Repeatedly execute a routine	FOR FOR...DO[1]	Relay ladder Structured text	478
Terminate the repeated execution of a routine	BRK EXIT[1]	Relay ladder Structured text	481
Return to the FOR instruction	RET	Relay ladder	482

[1] Structured text only.

For (FOR)

The FOR instruction executes a routine repeatedly.

Operands:



```
FOR
For
Routine name        ?
Index               ?
                   ??
Initial value       ?
Terminal value      ?
Step size           ?
```

Relay Ladder

Operand	Type	Format	Description
Routine name	ROUTINE	Routine Name	Routine to execute
Index	DINT	Tag	Counts how many times the routine has been executed
Initial value	SINT INT DINT	Immediate Tag	Value at which to start the index
Terminal value	SINT INT DINT	Immediate Tag	Value at which to stop executing the routine
Step size	SINT INT DINT	Immediate Tag	Amount to add to the index each time the FOR instruction executes the routine

```
FOR count:= initial_value TO
final_value BY increment DO
    <statement>;
END_FOR;
```

Structured Text

Use the FOR...DO construct. See Structured Text Programming for information on structured text constructs.

Description:

IMPORTANT Do not use a FOR instruction to call (execute) the main routine.

- You can put a FOR instruction in the main routine or any other routine.
- If you use a FOR instruction to call the main routine and then put a RET instruction in the main routine, a major fault occurs (type 4, code 31).

When enabled, the FOR instruction repeatedly executes the Routine until the Index value exceeds the Terminal value. This instruction does not pass parameters to the routine.

Each time the FOR instruction executes the routine, it adds the Step size to the Index.

Be careful not to loop too many times in a single scan. An excessive number of repetitions can cause the controller's watchdog to timeout, which causes a major fault.

Arithmetic Status Flags:

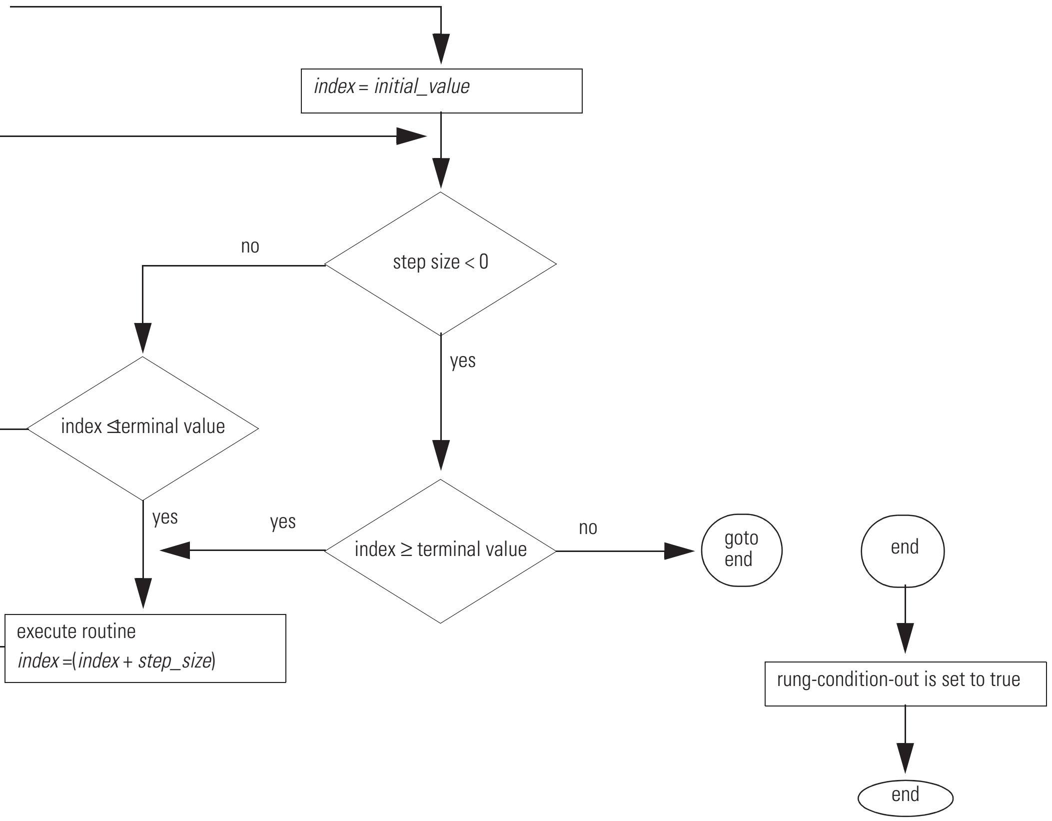
Not affected

Fault Conditions:

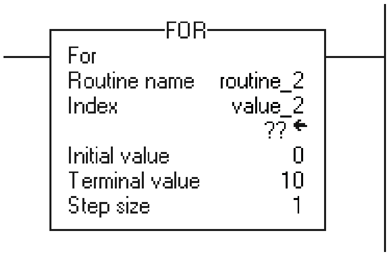
A major fault will occur if	Fault Type	Fault Code
Main routine contains a RET instruction	4	31

Execution:

Condition	Relay Ladder Action
Prescan	The rung-condition-out is set to false. The controller executes the subroutine once. If recursive FOR instruction0s exist to the same subroutine, the subroutine is prescanned only the first time. If multiple FOR instructions exist (non-recursive) to the same subroutine, the subroutine is prescanned each time.
Rung-condition-in is false	The rung-condition-out is set to false.

Condition	**Relay Ladder Action**
Rung-condition-in is true	
Postscan	The rung-condition-out is set to false.

Example: When enabled, the FOR instruction repeatedly executes *routine_2* and increments *value_2* by 1 each time. When *value_2* is >10 or a BRK instruction is enabled, the FOR instruction no longer executes *routine_2*.



For
Routine name routine_2
Index value_2
??
Initial value 0
Terminal value 10
Step size 1

Break (BRK)

The BRK instruction interrupts the execution of a routine that was called by a FOR instruction.

Operands:

—(BRK)—

Relay Ladder

None

`EXIT;`

Structured Text

Use the EXIT statement in a loop construct. See Appendix B for information on structured text constructs.

Description:

When enabled, the BRK instruction exits the routine and returns the controller to the instruction that follows the FOR.

If there are nested FOR instructions, a BRK instruction returns control to the innermost FOR instruction.

Arithmetic Status Flags:

Not affected

Fault Conditions:

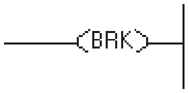
None

Execution:

Condition	Relay Ladder Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The rung-condition-out is set to true. Execution returns to the instruction that follows the calling FOR instruction.
Postscan	The rung-condition-out is set to false.

Example:

When enabled, the BRK instruction stops executing the current routine and returns to the instruction that follows the calling FOR instruction.



Return (RET)

The RET instruction returns to the calling FOR instruction.

Operands:

Relay Ladder

None

Description:

IMPORTANT Do not place a RET instruction in the main routine. If you place a RET instruction in the main routine, a major fault occurs (type 4, code 31).

When enabled, the RET instruction returns to the FOR instruction. The FOR instruction increments the Index value by the Step size and executes the subroutine again. If the Index value exceeds the Terminal value, the FOR instruction completes and execution moves on to the instruction that follows the FOR instruction.

The FOR instruction does not use parameters. The FOR instruction ignores any parameters you enter in a RET instruction.

You could also use a TND instruction to end execution of a subroutine.

Arithmetic Status Flags:

Not affected

Fault Conditions:

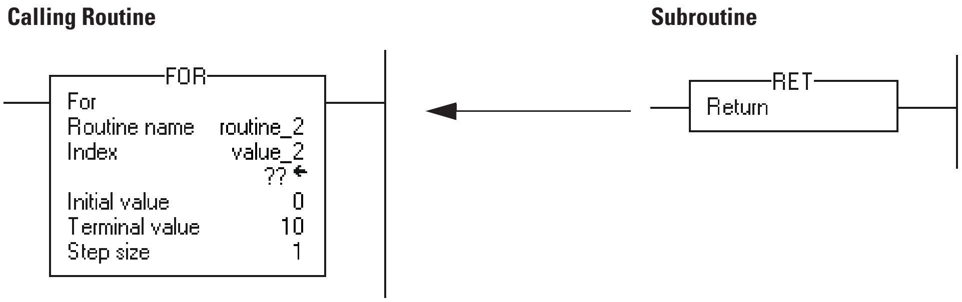
A major fault will occur if	Fault type	Fault code
Main routine contains a RET instruction	4	31

Execution:

Condition:	Relay Ladder Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	Returns the specified parameters to the calling routine. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Example: The FOR instruction repeatedly executes *routine_2* and increments *value_2* by 1 each time. When *value_2* is >10 or a BRK instruction is enabled, the FOR instruction no longer executes *routine_2*.

The RET instruction returns to the calling FOR instruction. The FOR instruction either executes the subroutine again and increments the Index value by the Step size or, if the Index value exceeds the Terminal value, the FOR instruction is complete and execution moves on to the instruction that follows the FOR instruction.



Notes:

Chapter 13

Special Instructions

(FBC, DDT, DTR, PID)

Introduction

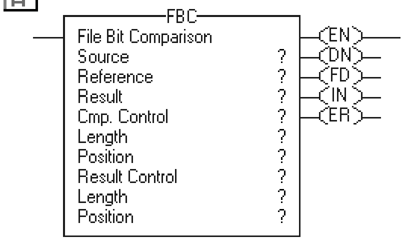
The special instructions perform application-specific operations.

If you want to	Use this instruction	Available in these languages	Page
Compare data against a known, good reference and record any mismatches	FBC	Relay ladder	486
Compare data against a known, good reference, record any mismatches, and update the reference to match the source	DDT	Relay ladder	494
Pass the source data through a mask and compare the result to reference data. Then write the source into the reference for the next comparison	DTR	Relay ladder	502
Control a PID loop	PID	Relay ladder Structured text	505

File Bit Comparison (FBC)

The FBC instruction compares bits in a Source array with bits in a Reference array.

Operands:



Relay Ladder

Operand	Type	Format	Description:
Source	DINT	Array tag	Array to compare to the reference **Do not** use CONTROL.POS in the subscript
Reference	DINT	Array tag	Array to compare to the source **Do not** use CONTROL.POS in the subscript
Result	DINT	Array tag	Array to store the result **Do not** use CONTROL.POS in the subscripts
Cmp control	CONTROL	Structure	Control structure for the compare
Length	DINT	Immediate	Number of bits to compare
Position	DINT	Immediate	Current position in the source Initial value is typically 0
Result control	CONTROL	Structure	Control structure for the results
Length	DINT	Immediate	Number of storage locations in the result
Position	DINT	Immediate	Current position in the result Initial value is typically 0

ATTENTION Use different tags for the compare control structure and the result control structure. Using the same tag for both could result in unpredictable operation, possibly causing equipment damage and/or injury to personnel.

COMPARE Structure

Mnemonic	Data Type	Description
.EN	BOOL	The enable bit indicates that the FBC instruction is enabled.
.DN	BOOL	The done bit is set when the FBC instruction compares the last bit in the Source and Reference arrays.
.FD	BOOL	The found bit is set each time the FBC instruction records a mismatch (one-at-a-time operation) or after recording all mismatches (all-per-scan operation).
.IN	BOOL	The inhibit bit indicates the FBC search mode. 0 = all mode 1 = one mismatch at a time mode
.ER	BOOL	The error bit is set if the compare .POS < 0, the compare .LEN < 0, the result .POS < 0 or the result .LEN < 0. The instruction stops executing until the program clears the .ER bit.
.LEN	DINT	The length value identifies the number of bits to compare.
.POS	DINT	The position value identifies the current bit.

RESULT Structure

Mnemonic	Data Type	Description
.DN	BOOL	The done bit is set when the Result array is full.
.LEN	DINT	The length value identifies the number of storage locations in the Result array.
.POS	DINT	The position value identifies the current position in the Result array.

Description:

When enabled, the FBC instruction compares the bits in the Source array with the bits in the Reference array and records the bit number of each mismatch in the Result array.

IMPORTANT

You must test and confirm that the instruction doesn't change data that you don't want it to change.

The FBC instruction operates on contiguous memory. In some cases, the instruction searches or writes past the array into other members of the tag. This happens if a length is too big and the tag is a user-defined data type.

The difference between the DDT and FBC instructions is that each time the DDT instruction finds a mismatch, the instruction changes the reference bit to match the source bit. The FBC instruction does not change the reference bit.

Select the Search Mode

If you want to detect	Select this mode
One mismatch at a time	Set the .IN bit in the compare CONTROL structure. Each time the rung-condition-in goes from false to true, the FBC instruction searches for the next mismatch between the Source and Reference arrays. Upon finding a mismatch, the instruction sets the .FD bit, records the position of the mismatch, and stops executing.
All mismatches	Clear the .IN bit in the compare CONTROL structure. Each time the rung-condition-in goes from false to true, the FSC instruction searches for all mismatches between the Source and Reference arrays.

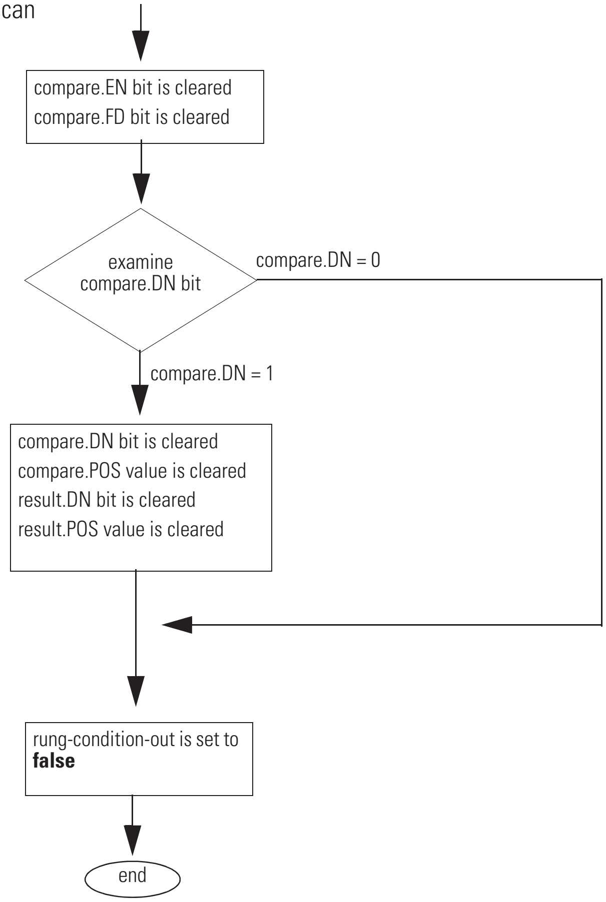
Arithmetic Status Flags: Not affected

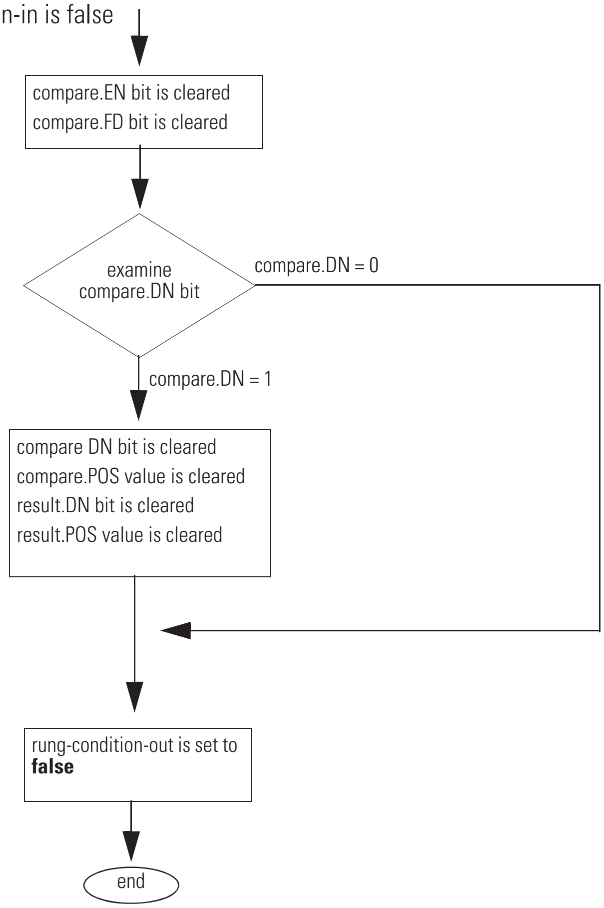
Fault Conditions:

A major fault will occur if	Fault type	Fault code
Result.POS > Size of Result array	4	20

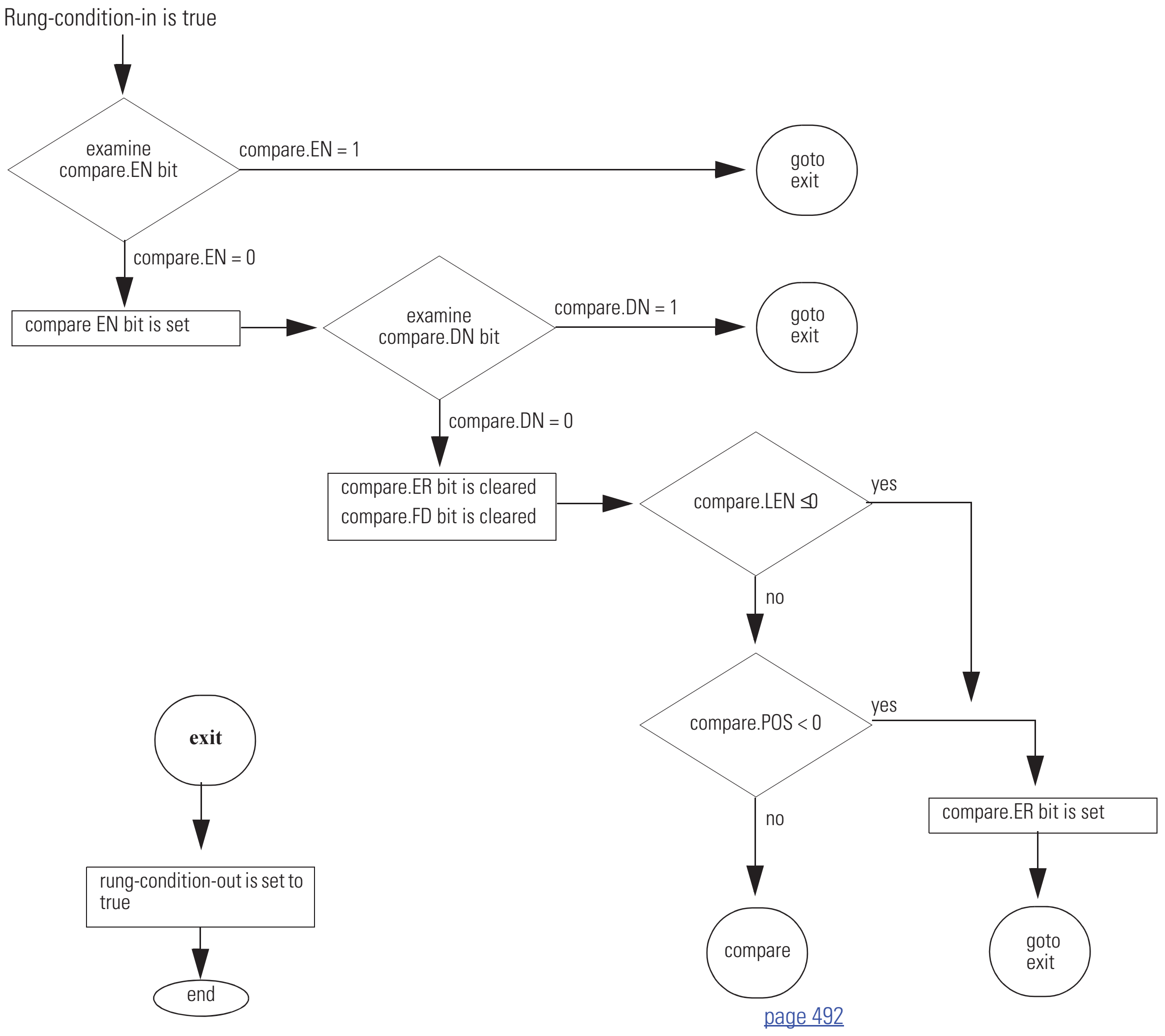
Execution:

Condition	Relay Ladder Action
Prescan	



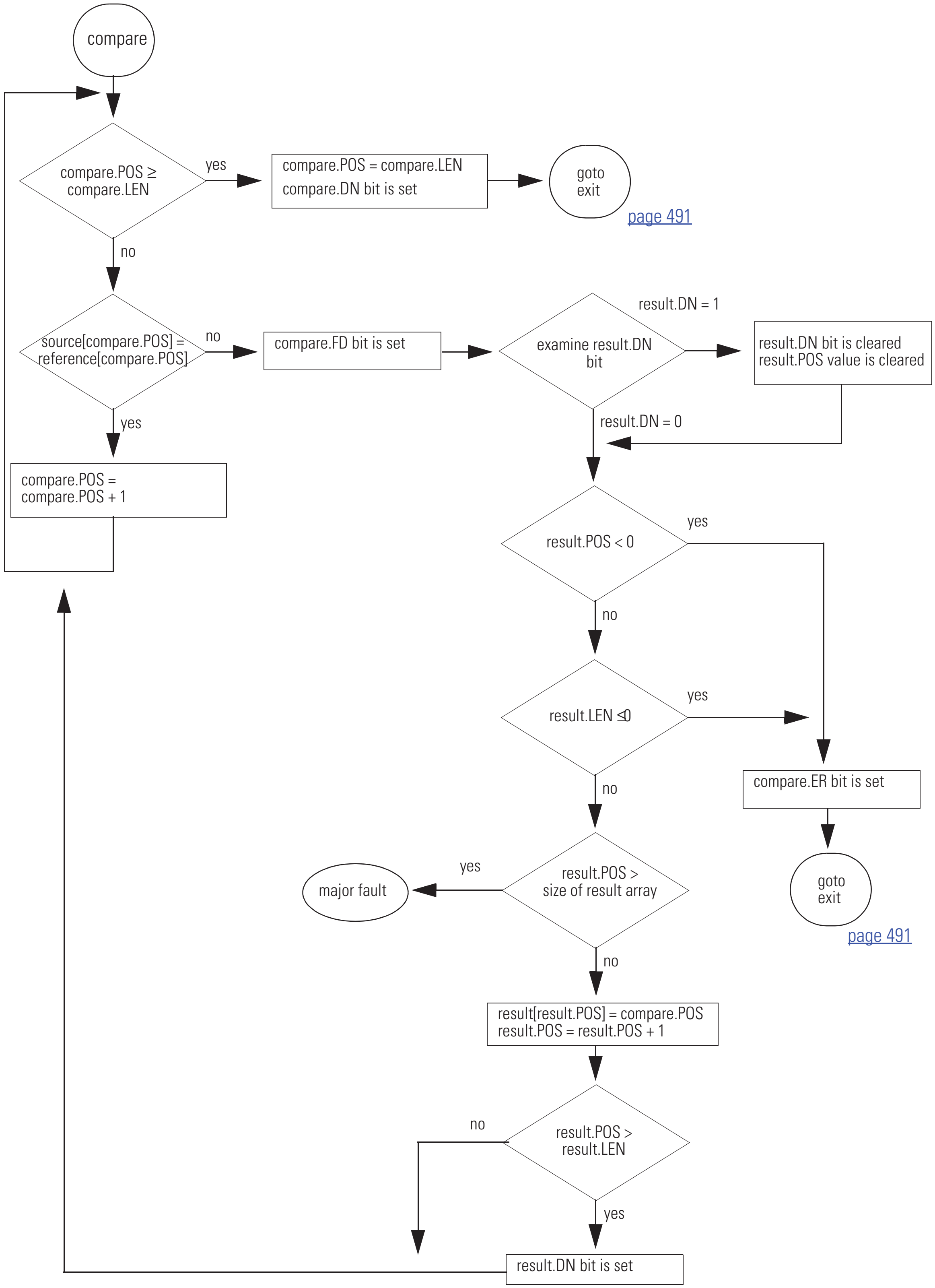
Condition	Relay Ladder Action
Rung-condition-in is false	

Condition	Relay Ladder Action



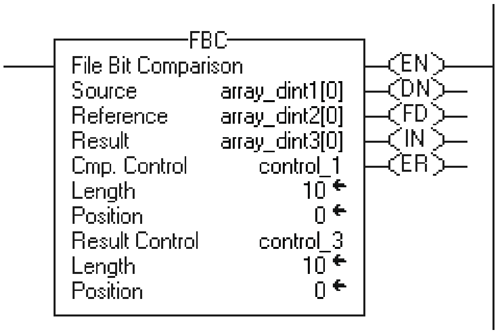
page 492

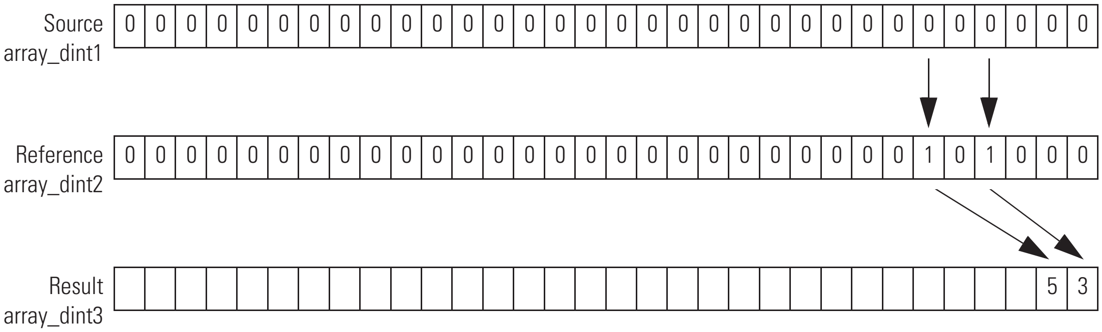
Condition	**Relay Ladder Action**



Postscan	The rung-condition-out is set to false.

Example: When enabled, the FBC instruction compares the source *array_dint1* to the reference *array_dint2* and stores the locations of any mismatches in the result *array_dint3*.

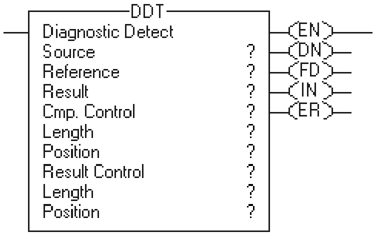




Diagnostic Detect (DDT)

The DDT instruction compares bits in a Source array with bits in a Reference array to determine changes of state.

Operands:



Relay Ladder

Operand	Type	Format	Description
Source	DINT	Array tag	Array to compare to the reference **Do not** use CONTROL.POS in the subscript
Reference	DINT	Array tag	Array to compare to the source **Do not** use CONTROL.POS in the subscript
Result	DINT	Array tag	Array to store the results **Do not** use CONTROL.POS in the subscript
Cmp control	CONTROL	Structure	Control structure for the compare
Length	DINT	Immediate	Number of bits to compare
Position	DINT	Immediate	Current position in the source Initial value typically 0
Result control	CONTROL	Structure	Control structure for the results
Length	DINT	Immediate	Number of storage locations in the result
Position	DINT	Immediate	Current position in the result Initial value typically 0

ATTENTION

Use different tags for the compare control structure and the result control structure. Using the same tag for both could result in unpredictable operation, possibly causing equipment damage and/or injury to personnel.

COMPARE Structure

Mnemonic	Data Type	Description
.EN	BOOL	The enable bit indicates that the DDT instruction is enabled.
.DN	BOOL	The done bit is set when the DDT instruction compares the last bit in the Source and Reference arrays.
.FD	BOOL	The found bit is set each time the DDT instruction records a mismatch (one-at-a-time operation) or after recording all mismatches (all-per-scan operation).
.IN	BOOL	The inhibit bit indicates the DDT search mode. 0 = all mode 1 = one mismatch at a time mode
.ER	BOOL	The error bit is set if the compare .POS < 0, the compare .LEN < 0, the result .POS < 0 or the result .LEN < 0. The instruction stops executing until the program clears the .ER bit.
.LEN	DINT	The length value identifies the number of bits to compare.
.POS	DINT	The position value identifies the current bit.

RESULT Structure

Mnemonic	Data Type	Description
.DN	BOOL	The done bit is set when the Result array is full.
.LEN	DINT	The length value identifies the number of storage locations in the Result array.
.POS	DINT	The position value identifies the current position in the Result array.

Description: When enabled, the DDT instruction compares the bits in the Source array with the bits in the Reference array, records the bit number of each mismatch in the Result array, and changes the value of the Reference bit to match the value of the corresponding Source bit.

IMPORTANT You must test and confirm that the instruction doesn't change data that you don't want it to change.

The DDT instruction operates on contiguous memory. In some cases, the instruction searches or writes past the array into other members of the tag. This happens if a length is too big and the tag is a user-defined data type.

The difference between the DDT and FBC instructions is that each time the DDT instruction finds a mismatch, the DDT instruction changes the reference bit to match the source bit. The FBC instruction does not change the reference bit.

Select the Search Mode

If you want to detect	Select this mode
One mismatch at a time	Set the .IN bit in the compare CONTROL structure. Each time the rung-condition-in goes from false to true, the DDT instruction searches for the next mismatch between the Source and Reference arrays. Upon finding a mismatch, the instruction sets the .FD bit, records the position of the mismatch, and stops executing.
All mismatches	Clear the .IN bit in the compare CONTROL structure. Each time the rung-condition-in goes from false to true, the DDT instruction searches for all mismatches between the Source and Reference arrays.

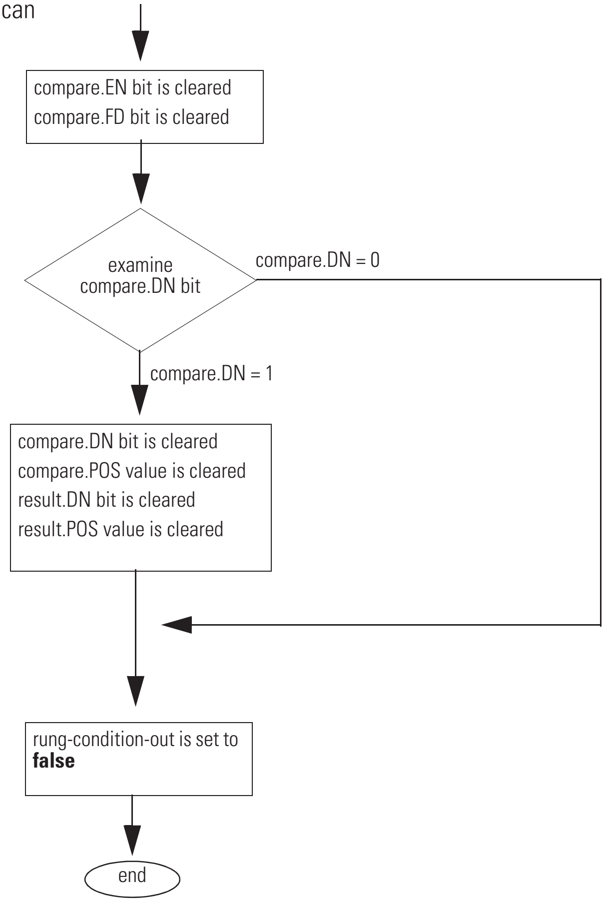
Arithmetic Status Flags: Not affected

Fault Conditions:

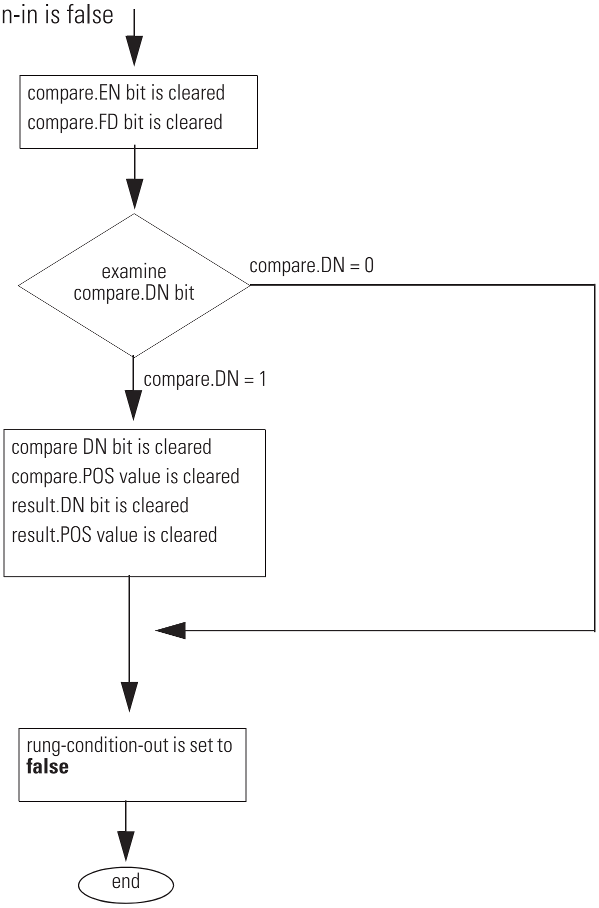
A major fault will occur if	Fault type	Fault code
Result.POS > size of Result array	4	20

Execution:

Condition:	Relay Ladder Action
Prescan	

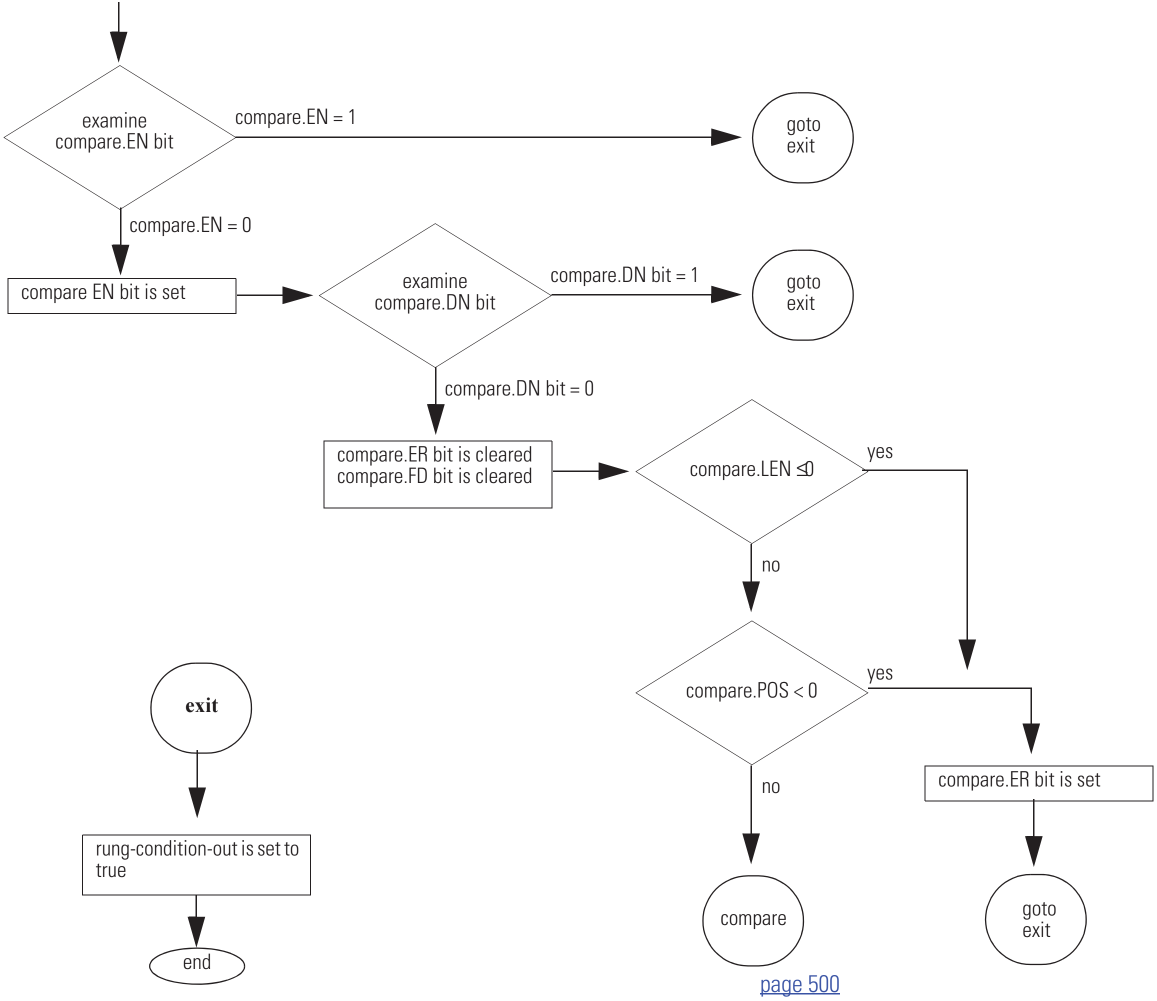


Condition:	**Relay Ladder Action**
Rung-condition-in is false	

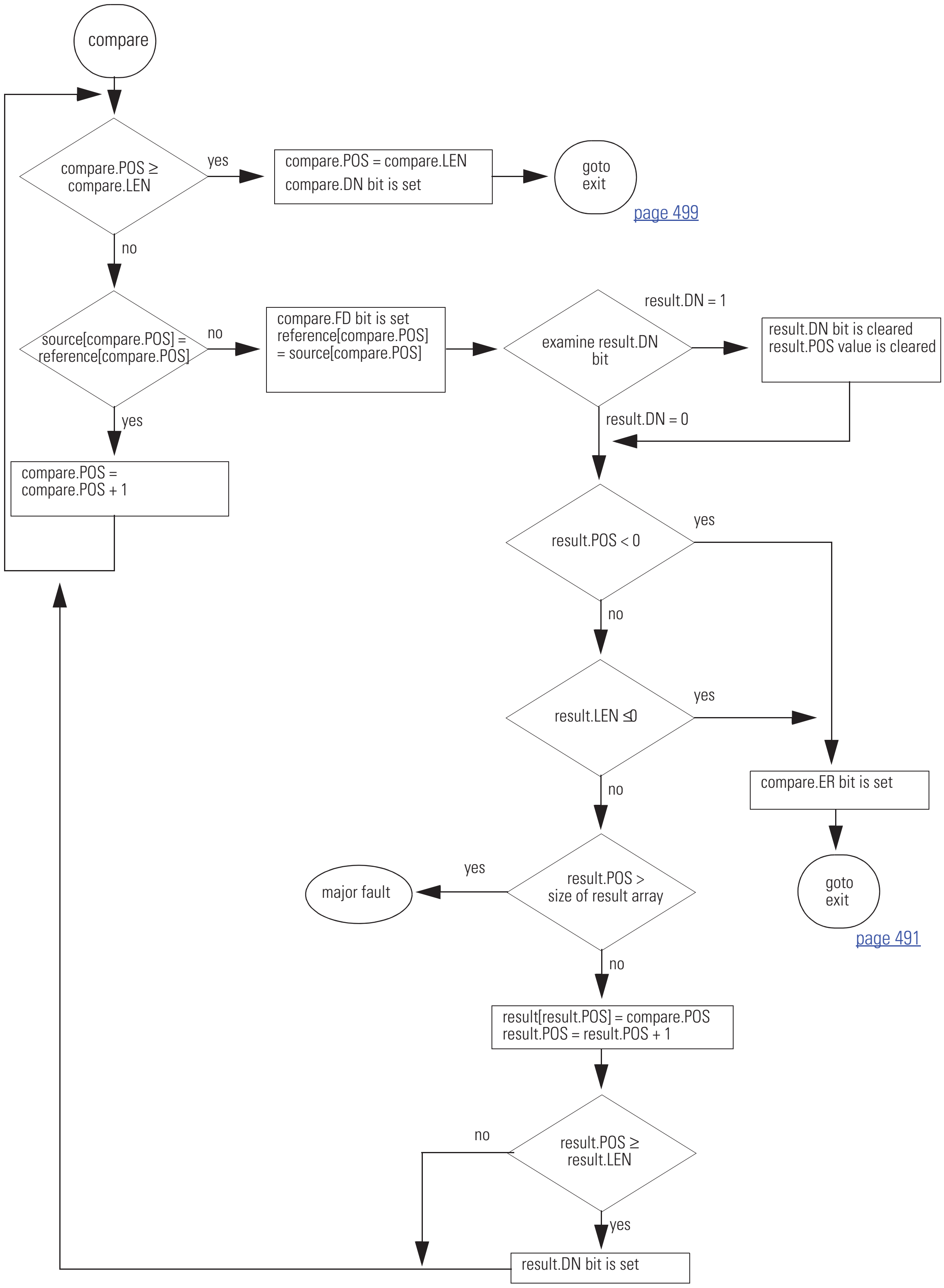


Condition:	**Relay Ladder Action**

Rung-condition-in is true

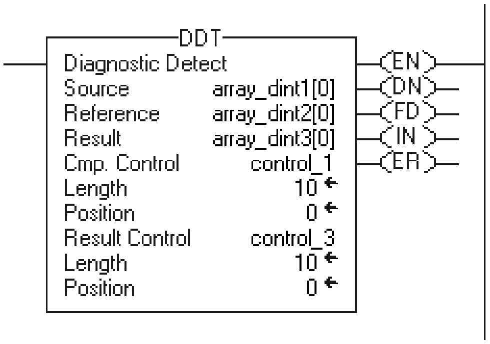


Condition:	**Relay Ladder Action**

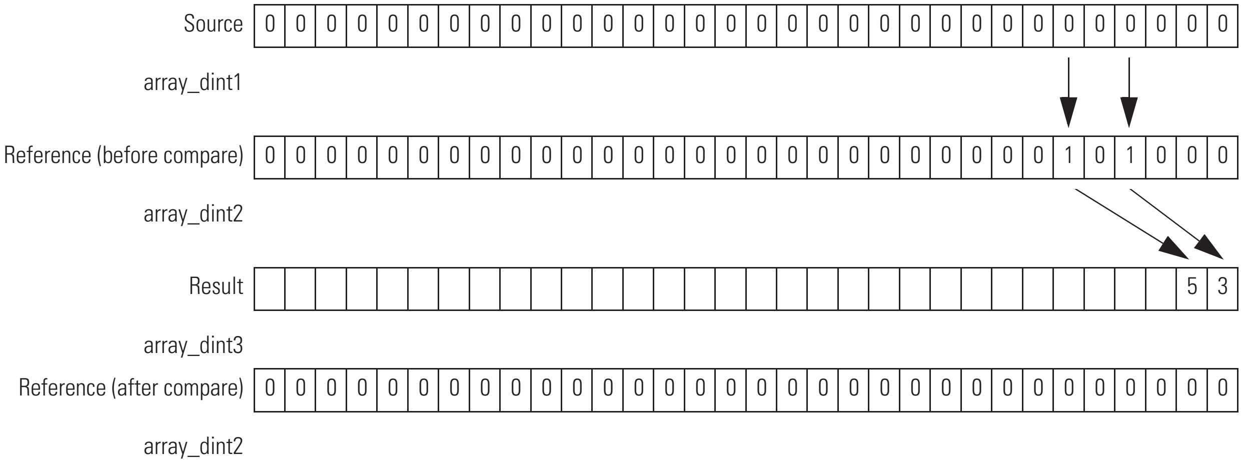


Postscan	The rung-condition-out is set to false.

Example: When enabled, the DDT instruction compares the source *array_dint1* to the reference *array_dint2* and stores the locations of any mismatches in the result *array_dint3*. The controller also changes the mismatched bits in the reference *array_dint2* to match the source *array_dint1*.



DDT	
Diagnostic Detect	
Source	array_dint1[0]
Reference	array_dint2[0]
Result	array_dint3[0]
Cmp. Control	control_1
Length	10 ←
Position	0 ←
Result Control	control_3
Length	10 ←
Position	0 ←



Source 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

array_dint1

Reference (before compare) 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0

array_dint2

Result 5 3

array_dint3

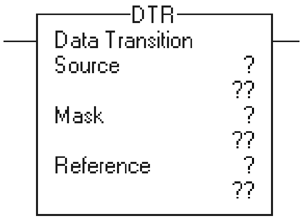
Reference (after compare) 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

array_dint2

Data Transitional (DTR)

The DTR instruction passes the Source value through a Mask and compares the result with the Reference value.

Operands:



Relay Ladder

Operand	Type	Format	Description
Source	DINT	Immediate Tag	Array to compare to the reference
Mask	DINT	Immediate Tag	Which bits to block or pass
Reference	DINT	Tag	Array to compare to the source

Description:

The DTR instruction passes the Source value through a Mask and compares the result with the Reference value. The DTR instruction also writes the masked Source value into the Reference value for the next comparison. The Source remains unchanged.

A ‘1’ in the mask means the data bit is passed. A ‘0’ in the mask means the data bit is blocked.

When the masked Source differs from the Reference, the rung-condition-out goes true for one scan. When the masked Source is the same as the Reference, the rung-condition-out is false.

ATTENTION

Online programming with this instruction can be dangerous. If the Reference value is different than the Source value, the rung-condition-out goes true. Use caution if you insert this instruction when the processor is in Run or Remote Run mode.

Enter an Immediate Mask Value

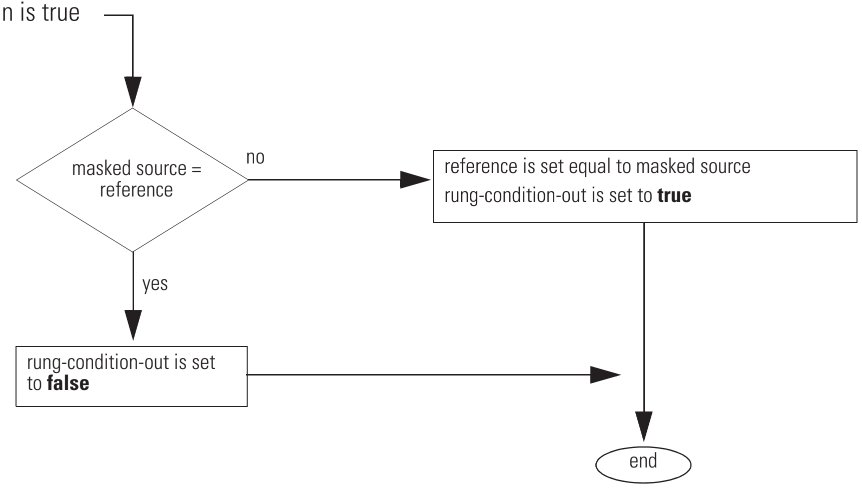
When you enter a mask, the programming software defaults to decimal values. If you want to enter a mask using another format, precede the value with the correct prefix.

Prefix	Description
16#	Hexadecimal For example; 16#0F0F
8#	Octal For example; 8#16
2#	Binary For example; 2#00110011

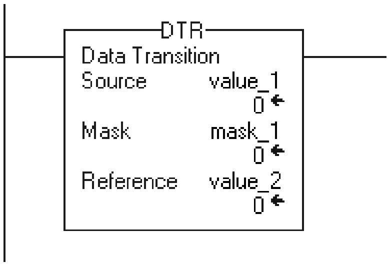
Arithmetic Status Flags: Not affected

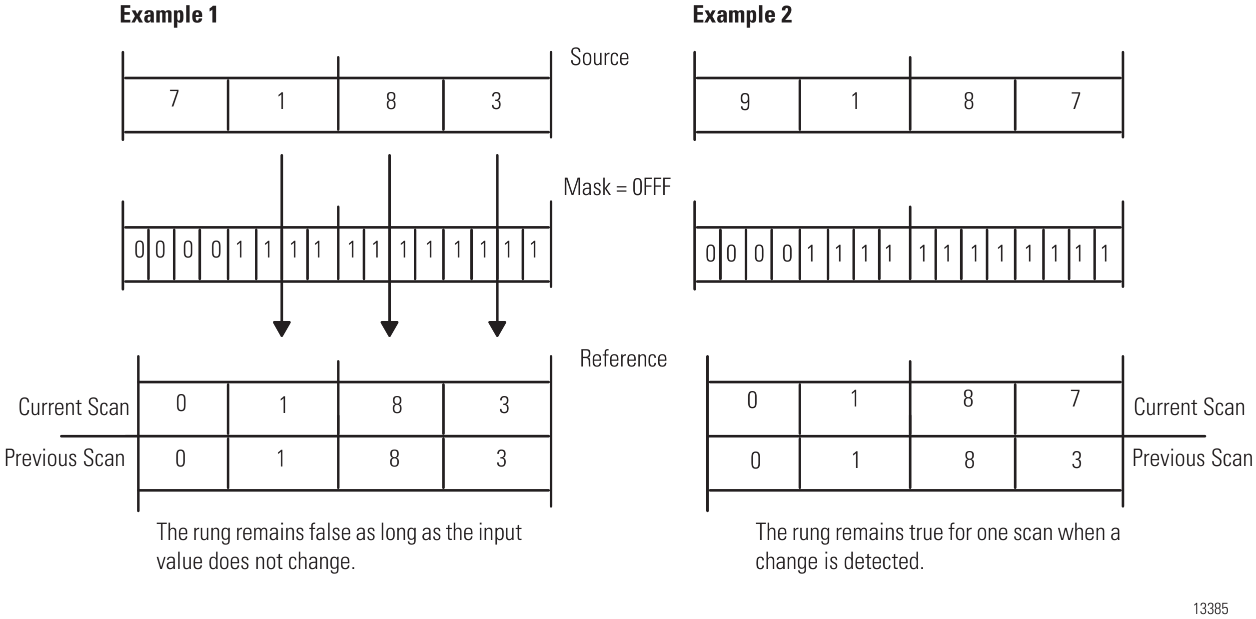
Fault Conditions: None

Execution:

Condition	Relay Ladder Action
Prescan	The Reference = Source AND Mask. The rung-condition-out is set to false.
Rung-condition-in is false	The Reference = Source AND Mask. The rung-condition-out is set to false.
Rung-condition-in is true	
Postscan	The rung-condition-out is set to false.

Example: When enabled, the DTR instruction masks *value_1*. If there is a difference in the two values, the rung-condition-out is set to true.





The rung remains false as long as the input value does not change.

The rung remains true for one scan when a change is detected.

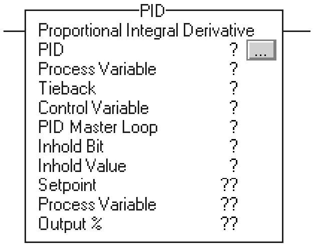
A 0 in the mask leaves the bit unchanged.

Proportional Integral Derivative (PID)

The PID instruction controls a process variable such as flow, pressure, temperature, or level.

Operands:

Relay Ladder



PID
Proportional Integral Derivative
PID ?
Process Variable ?
Tieback ?
Control Variable ?
PID Master Loop ?
Inhold Bit ?
Inhold Value ?
Setpoint ??
Process Variable ??
Output % ??

Operand	Type	Format	Description
PID	PID	Structure	PID structure
Process variable	SINT INT DINT REAL	Tag	Value you want to control
Tieback	SINT INT DINT REAL	Immediate Tag	(Optional) Output of a hardware hand/auto station that is bypassing the output of the controller Enter 0 if you don’t want to use this parameter.
Control variable	SINT INT DINT REAL	Tag	Value that goes to the final control device (valve, damper, and so forth) If you are using the deadband, the Control variable must be REAL or it will be forced to 0 when the error is within the deadband.
PID master loop	PID	Structure	(Optional) PID tag for the master PID If you are performing cascade control and this PID is a slave loop, enter the name of the master PID. Enter 0 if you don’t want to use this parameter.
Inhold bit	BOOL	Tag	(Optional) Current status of the inhold bit from a 1756 analog output channel to support bumpless restart Enter 0 if you don’t want to use this parameter.
Inhold value	SINT INT DINT REAL	Tag	(Optional) Data readback value from a 1756 analog output channel to support bumpless restart Enter 0 if you don’t want to use this parameter.

Operand	Type	Format	Description
Setpoint			Displays current value of the setpoint
Process variable			Displays current value of the scaled Process Variable
Output %			Displays current output percentage value

```
PID(PID,ProcessVariable,
Tieback,ControlVariable,
PIDMasterLoop,InholdBit,
InHoldValue);
```

Structured Text

The operands are the same as those for the relay ladder PID instruction. However, you specify the Setpoint, Process Variable, and Output percent by accessing the .SP, .PV.and .OUT members of the PID structure, rather than by including values in the operand list.

PID Structure

Mnemonic:	Data Type	Description	
.CTL	DINT	The .CTL member provides access to the status members (bits) in one, 32-bit word. The PID instruction sets bits 07…15.	
		This bit	**Is this member**
		31	.EN
		30	.CT
		29	.CL
		28	.PVT
		27	.DOE
		26	.SWM
		25	.CA
		24	.MO
		23	.PE
		22	.NDF
		21	.NOBC
		20	.NOZC
		This bit	**Is this member, which the PID instruction sets**
		15	.INI
		14	.SPOR
		13	.OLL
		12	.OLH
		11	.EWD
		10	.DVNA
		09	.DVPA
		08	.PVLA
		07	.PVHA

Mnemonic:	Data Type	Description	
.SP	REAL	Setpoint	
.KP	REAL	Independent	Proportional gain (unitless)
		Dependent	Controller gain (unitless)
.KI	REAL	Independent	Integral gain (1/sec)
		Dependent	Reset time (minutes per repeat)
.KD	REAL	Independent	Derivative gain (seconds)
		Dependent	Rate time (minutes)
.BIAS	REAL	Feedforward or bias %	
.MAXS	REAL	Maximum engineering unit scaling value	
.MINS	REAL	Minimum engineering unit scaling value	
.DB	REAL	Deadband engineering units	
.SO	REAL	Set output %	
.MAXO	REAL	Maximum output limit (% of output)	
.MINO	REAL	Minimum output limit (% of output)	
.UPD	REAL	Loop update time (seconds)	
.PV	REAL	Scaled PV value	
.ERR	REAL	Scaled error value	
.OUT	REAL	Output %	
.PVH	REAL	Process variable high alarm limit	
.PVL	REAL	Process variable low alarm limit	
.DVP	REAL	Positive deviation alarm limit	
.DVN	REAL	Negative deviation alarm limit	
.PVDB	REAL	Process variable alarm deadband	
.DVDB	REAL	Deviation alarm deadband	
.MAXI	REAL	Maximum PV value (unscaled input)	
.MINI	REAL	Minimum PV value (unscaled input)	
.TIE	REAL	Tieback value for manual control	
.MAXCV	REAL	Maximum CV value (corresponding to 100%)	
.MINCV	REAL	Minimum CV value (corresponding to 0%)	
.MINTIE	REAL	Minimum tieback value (corresponding to 100%)	
.MAXTIE	REAL	Maximum tieback value (corresponding to 0%)	

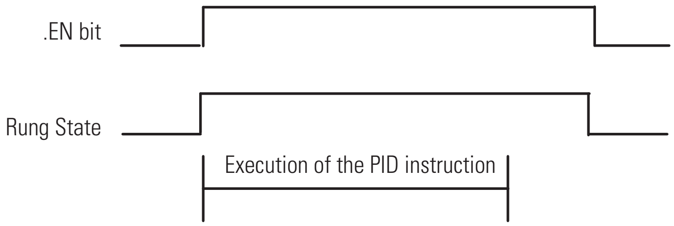
Mnemonic:	**Data Type**	**Description**	
.DATA	REAL[17]	The .DATA member stores:	
		Element	**Description**
		.DATA[0]	Integral accumulation
		.DATA[1]	Derivative smoothing temporary value
		.DATA[2]	Previous .PV value
		.DATA[3]	Previous .ERR value
		.DATA[4]	Previous valid .SP value
		.DATA[5]	Percent scaling constant
		.DATA[6]	.PV scaling constant
		.DATA[7]	Derivative scaling constant
		.DATA[8]	Previous .KP value
		.DATA[9]	Previous .KI value
		.DATA[10]	Previous .KD value
		.DATA[11]	Dependent gain .KP
		.DATA[12]	Dependent gain .KI
		.DATA[13]	Dependent gain .KD
		.DATA[14]	Previous .CV value
		.DATA[15]	.CV descaling constant
		.DATA[16]	Tieback descaling constant
.EN	BOOL	Enabled	
.CT	BOOL	Cascade type (0=slave; 1=master)	
.CL	BOOL	Cascade loop (0=no; 1=yes)	
.PVT	BOOL	Process variable tracking (0=no; 1=yes)	
.DOE	BOOL	Derivative of (0=PV; 1=error)	
.SWM	BOOL	Software manual mode (0=no-auto; 1=yes- sw manual)	
.CA	BOOL	Control action (0 means E=SP-PV; 1 means E=PV-SP)	
.MO	BOOL	Station mode (0=automatic; 1=manual)	
.PE	BOOL	PID equation (0=independent; 1=dependent)	
.NDF	BOOL	No derivative smoothing (0=derivative smoothing filter enabled; 1=derivative smoothing filter disabled)	
.NOBC	BOOL	No bias back calculation (0=bias back calculation enabled; 1=bias back calculation disabled)	
.NOZC	BOOL	No zero crossing deadband (0=deadband is zero crossing; 1=deadband is not zero crossing)	
.INI	BOOL	PID initialized (0=no; 1=yes)	
.SPOR	BOOL	Setpoint out of range (0=no; 1=yes)	
.OLL	BOOL	CV is below minimum output limit (0=no; 1=yes)	
.OLH	BOOL	CV is above maximum output limit (0=no; 1=yes)	

Mnemonic:	Data Type	Description
.EWD	BOOL	Error is within deadband (0=no; 1=yes)
.DVNA	BOOL	Deviation is alarmed low (0=no; 1=yes)
.DVPA	BOOL	Deviation is alarmed high (0=no; 1=yes)
.PVLA	BOOL	PV is alarmed low (0=no; 1=yes)
.PVHA	BOOL	PV is alarmed high (0=no; 1=yes)

Description:

The PID instruction typically receives the process variable (PV) from an analog input module and modulates a control variable output (CV) on an analog output module in order to maintain the process variable at the desired setpoint.

The .EN bit indicates execution status. The .EN bit is set when the rung-condition-in transitions from false to true. The .EN bit is cleared when the rung-condition-in becomes false. The PID instruction does not use a .DN bit. The PID instruction executes every scan as long as the rung-condition-in is true.



Arithmetic Status Flags:

Not affected

Fault Conditions:

IMPORTANT These faults were major faults in the PLC-5 controller.

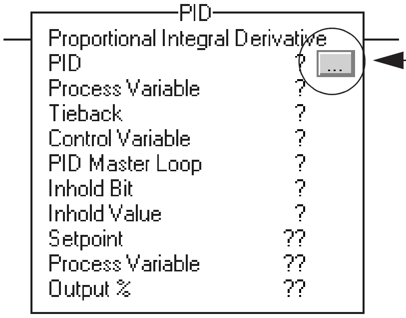
A minor fault will occur if	Fault type	Fault code
.UPD ≤0	4	35
Setpoint out of range	4	36

Execution:

Condition	Action	Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction executes the PID loop.	The instruction executes the PID loop.
Postscan	The rung-condition-out is set to false.	No action taken.

Configure a PID Instruction

After you enter the PID instruction and specify the PID structure, you use the configuration tabs to specify how the PID instruction should function.



PID
Proportional Integral Derivative
PID ?
Process Variable ?
Tieback ?
Control Variable ?
PID Master Loop ?
Inhold Bit ?
Inhold Value ?
Setpoint ??
Process Variable ??
Output % ??

Click here to configure the PID instruction

Specify Tuning

Select the Tuning tab. Changes take affect as soon as you click on another field, click OK, click Apply, or press Enter.

In this field	Specify
Setpoint (SP)	Enter a setpoint value (.SP).
Set output %	Enter a set output percentage (.SO). In software manual mode, this value is used for the output. In auto mode, this value displays the output %.
Output bias	Enter an output bias percentage (.BIAS).
Proportional gain (K_p)	Enter the proportional gain (.KP). For independent gains, it's the proportional gain (unitless). For dependent gains, it's the controller gain (unitless).
Integral gain (K_i)	Enter the integral gain (.KI). For independent gains, it's the integral gain (1/sec). For dependent gains, it's the reset time (minutes per repeat).
Derivative time (K_d)	Enter the derivative gain (.KD). For independent gains, it's the derivative gain (seconds). For dependent gains, it's the rate time minutes).
Manual mode	Select either manual (.MO) or software manual (.SWM). Manual mode overrides software manual mode if both are selected.

Specify Configuration

Select the Configuration tab. You must click OK or Apply for any changes to take effect.

In this field	Specify
PID equation	Select independent gains or dependent gains (.PE). Use independent when you want the three gains (P, I, and D) to operate independently. Use dependent when you want an overall controller gain that affects all three terms (P, I, and D).
Control action	Select either E=PV-SP or E=SP-PV for the control action (.CA).
Derivative of	Select PV or error (.DOE). Use the derivative of PV to eliminate output spikes resulting from setpoint changes. Use the derivative of error for fast responses to setpoint changes when the algorithm can tolerate overshoots.
Loop update time	Enter the update time (.UPD) for the instruction.
CV high limit	Enter a high limit for the control variable (.MAXO).(1)
CV low limit	Enter a low limit for the control variable (.MINO).(1)
Deadband value	Enter a deadband value (.DB).
No derivative smoothing	Enable or disable this selection (.NDF).
No bias calculation	Enable or disable this selection (.NOBC).
No zero crossing in deadband	Enable or disable this selection (.NOZC).
PV tracking	Enable or disable this selection (.PVT).
Cascade loop	Enable or disable this selection (.CL).
Cascade type	If cascade loop is enabled, select either slave or master (.CT).

(1) When using the ladder-based PID instruction, if you set MAXO = MINO, the PID instruction will reset these values to default. MAXO = 100.0 and MINO = 0.0

Specify Alarms

Select the Alarms tab. You must click OK or Apply for any changes to take effect.

In this field	Specify
PV high	Enter a PV high alarm value (.PVH).
PV low	Enter a PV low alarm value (.PVL).
PV deadband	Enter a PV alarm deadband value (.PVDB).
Positive deviation	Enter a positive deviation value (.DVP).
Negative deviation	Enter a negative deviation value (.DVN).
Deviation deadband	Enter a deviation alarm deadband value (.DVDB).

Specify Scaling

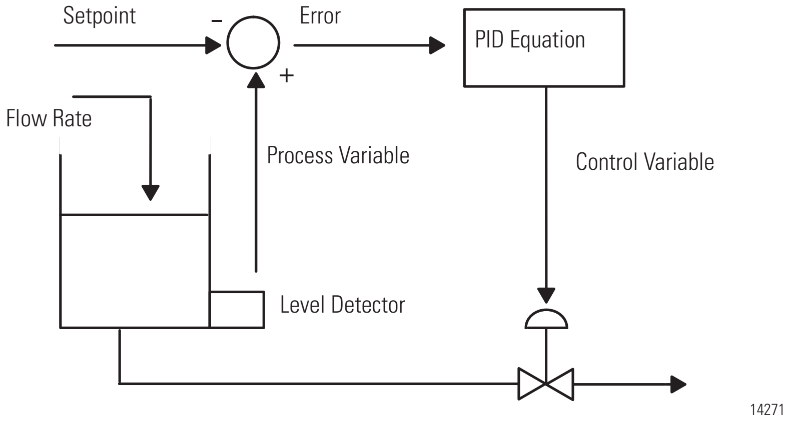
Select the Scaling tab. You must click OK or Apply for any changes to take effect.

In this field	Specify
PV unscaled maximum	Enter a maximum PV value (.MAXI) that equals the maximum unscaled value received from the analog input channel for the PV value.
PV unscaled minimum	Enter a minimum PV value (.MINI) that equals the minimum unscaled value received from the analog input channel for the PV value.
PV engineering units maximum	Enter the maximum engineering units corresponding to .MAXI (.MAXS)(1)
PV engineering units minimum	Enter the minimum engineering units corresponding to .MINI (.MINS)(1)
CV maximum	Enter a maximum CV value corresponding to 100% (.MAXCV).
CV minimum	Enter a minimum CV value corresponding to 0% (.MINCV).
Tieback maximum	Enter a maximum tieback value (.MAXTIE) that equals the maximum unscaled value received from the analog input channel for the tieback value.
Tieback minimum	Enter a minimum tieback value (.MINTIE) that equals the minimum unscaled value received from the analog input channel for the tieback value.
PID Initialized	If you change scaling constants during Run mode, turn this off to reinitialize internal descaling values (.INI).

(1) When using the ladder-based PID instruction, if you set MAXO = MINO, the PID instruction will reset these values to default. MAXO = 100.0 and MINO = 0.0

Use PID Instructions

PID closed-loop control holds a process variable at a desired set point. The illustration shows an example of a flow-rate/fluid level.



In the above example, the level in the tank is compared against the setpoint. If the level is higher than the setpoint, the PID equation increases the control variable and causes the outlet valve from the tank to open; thereby decreasing the level in the tank.

The PID equation used in the PID instruction is a positional form equation with the option of using either independent gains or dependent gains. When using independent gains, the proportional, integral, and derivative gains affect only their specific proportional, integral, or derivative terms respectively. When using dependent gains, the proportional gain is replaced with a controller gain that affects all three terms. You can use either form of equation to perform the same type of control. The two equation types are merely provided to let you use the equation type with which you are most familiar.

Gains Option	Derivative Of	Equation
Dependent gains (ISA standard)	Error (E)	$CV = K_C\left[E + \frac{1}{T_i}\int_0^t E dt + T_d\frac{dE}{dt}\right] + BIAS$
	Process variable (PV)	E = SP - PV $CV = K_C\left[E + \frac{1}{T_i}\int_0^t E dt - T_d\frac{dPV}{dt}\right] + BIAS$ E = PV - SP $CV = K_C\left[E + \frac{1}{T_i}\int_0^t E dt + T_d\frac{dPV}{dt}\right] + BIAS$
Independent gains	Error (E)	$CV = K_PE + K_i\int_0^t E dt + K_d\frac{dE}{dt} + BIAS$
	Process variable (PV)	E = SP - PV $CV = K_PE + K_i\int_0^t E dt - K_d\frac{dPV}{dt} + BIAS$ E = PV - SP $CV = K_PE + K_i\int_0^t E dt + K_d\frac{dPV}{dt} + BIAS$

Where:

Variable	Description
K_P	Proportional gain (unitless) $K_p = K_c$ unitless
K_i	Integral gain (seconds $^{-1}$) To convert between K_i (integral gain) and T_i (reset time), use: $K_i = \frac{K_C}{60 T_i}$
K_d	Derivative gain (seconds) To convert between K_d (derivative gain) and T_d (rate time), use: $K_d = K_c (T_d) 60$
K_C	Controller gain (unitless)
T_i	Reset time (minutes/repeat)
T_d	Rate time (minutes)
SP	Setpoint
PV	Process variable
E	Error [(SP-PV) or (PV-SP)]
BIAS	Feedforward or bias
CV	Control variable
dt	Loop update time

If you do not want to use a particular term of the PID equation, just set its gain to zero. For example if you want no derivative action, set K_d or T_d equal to zero.

Anti-reset Windup and Bumpless Transfer from Manual to Auto

The PID instruction automatically avoids reset windup by preventing the integral term from accumulating whenever the CV output reaches its maximum or minimum values, as set by .MAXO and .MINO. The accumulated integral term remains frozen until the CV output drops below its maximum limit or rises above its minimum limit. Then normal integral accumulation automatically resumes.

The PID instruction supports two manual modes of control.

Manual Mode of Control	Description
Software manual (.SWM)	Also known as set output mode Lets the user set the output % from the software The set output (.SO) value is used as the output of the loop. The set output value typically comes from an operator input from an operator interface device.
Manual (.MO)	Takes the tieback value, as an input, and adjusts its internal variables to generate the same value at the output The tieback input to the PID instruction is scaled to 0-100% according to the values of .MINTIE and .MAXTIE and is used as the output of the loop. The tieback input typically comes from the output of a hardware hand/auto station that is bypassing the output from the controller. **Important:** Manual mode overrides software manual mode if both mode bits are set on.

The PID instruction also automatically provides bumpless transfers from software manual mode to auto mode or from manual to auto mode. The PID instruction back-calculates the value of the integral accumulation term required to make the CV output track either the set output (.SO) value in software manual mode or the tieback input in manual mode. In this manner, when the loop switches to auto mode, the CV output starts off from the set output or tieback value and no 'bump' in output value occurs.

The PID instruction can also automatically provide a bumpless transfer from manual to auto even if integral control is not used (that is $K_i = 0$). In this case the instruction modifies the .BIAS term to make the CV output track either the set output or tieback values. When automatic control is resumed, the .BIAS term will maintain its last value. You can disable back-calculation of the .BIAS term by setting the .NOBC bit in the PID data structure. Be aware that if you set .NOBC true, the PID instruction no longer provides a bumpless transfer from manual to auto when integral control is not used.

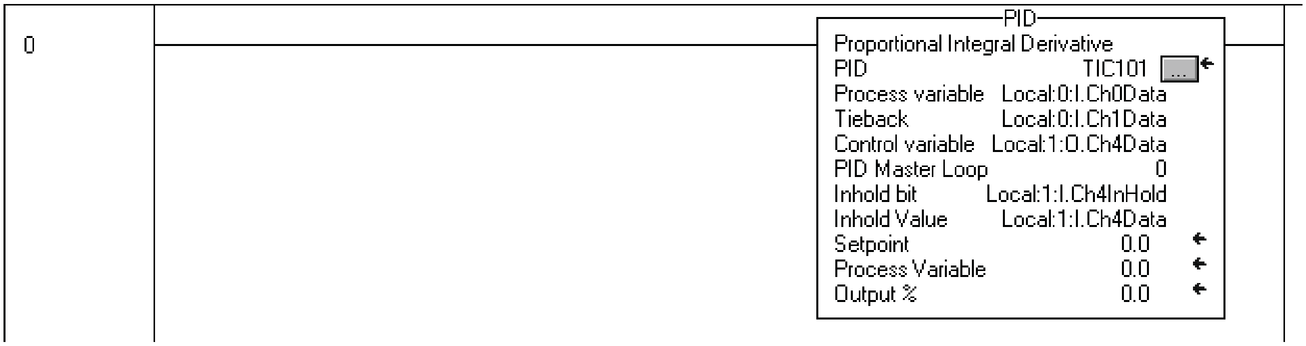
PID Instruction Timing

The PID instruction and the sampling of the process variable need to be updated at a periodic rate. This update time is related to the physical process you are controlling. For very slow loops, such as temperature loops, an update time of once per second or even longer is usually sufficient to obtain good control. Somewhat faster loops, such as pressure or flow loops, may require an update time such as once every 250 ms. Only rare cases, such as tension control on an unwinder spool, require loop updates as fast as every 10 ms or faster.

Because the PID instruction uses a time base in its calculation, you need to synchronize execution of this instruction with sampling of the process variable (PV).

The easiest way to execute the PID instruction is to put the PID instruction in a periodic task. Set the loop update time (.UPD) equal to the periodic task rate and make sure that the PID instruction is executed every scan of the periodic task.

Relay Ladder



Proportional Integral Derivative
PID TIC101
Process variable Local:0:I.Ch0Data
Tieback Local:0:I.Ch1Data
Control variable Local:1:O.Ch4Data
PID Master Loop 0
Inhold bit Local:1:I.Ch4InHold
Inhold Value Local:1:I.Ch4Data
Setpoint 0.0
Process Variable 0.0
Output % 0.0

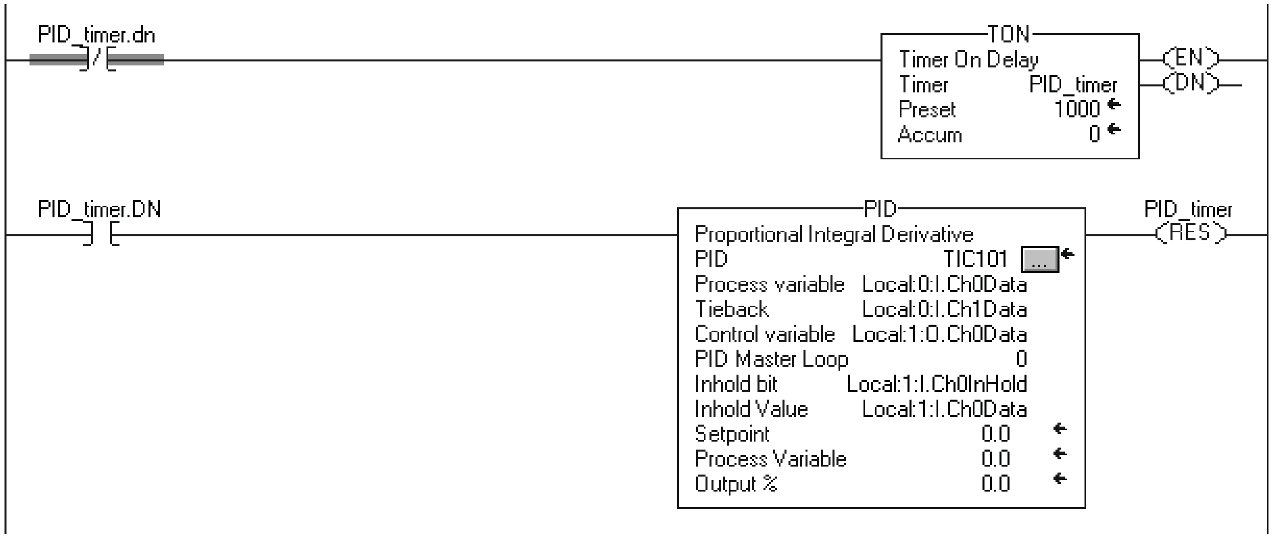
Structured Text

```
PID(TIC101,Local:0:I.Ch0Data,Local:0:I.Ch1Data,
     Local:1:O.Ch4Data,0,Local:1:I.Ch4InHold,
     Local:1:I.Ch4Data);
```

When using a periodic task, make sure that the analog input used for the process variable is updated to the processor at a rate that is significantly faster than the rate of the periodic task. Ideally, the process variable should be sent to the processor at least five…10 times faster than the periodic task rate. This minimizes the time difference between actual samples of the process variable and execution of the PID loop. For example, if the PID loop is in a 250 ms periodic task, use a loop update time of 250 ms (.UPD = .25), and configure the analog input module to produce data at least about every 25…50 ms.

Another, somewhat less accurate, method of executing a PID instruction is to place the instruction in a continuous task and use a timer done bit to trigger execution of the PID instruction.

Relay Ladder



Structured Text

```
PID_timer.pre := 1000

TONR(PID_timer);

IF PID_timer.DN THEN

     PID(TIC101,Local:0:I.Ch0Data,Local:0:I.Ch1Data,
     Local:1:O.Ch0Data,0,Local:1:I.Ch0InHold,
     Local:1:I.Ch0Data);

END_IF;
```

In this method, the loop update time of the PID instruction should be set equal to the timer preset. As in the case of using a periodic task, you should set the analog input module to produce the process variable at a significantly faster rate than the loop update time. You should only use the timer method of PID execution for loops with loop update times that are at least several times longer than the worst-case execution time for your continuous task.

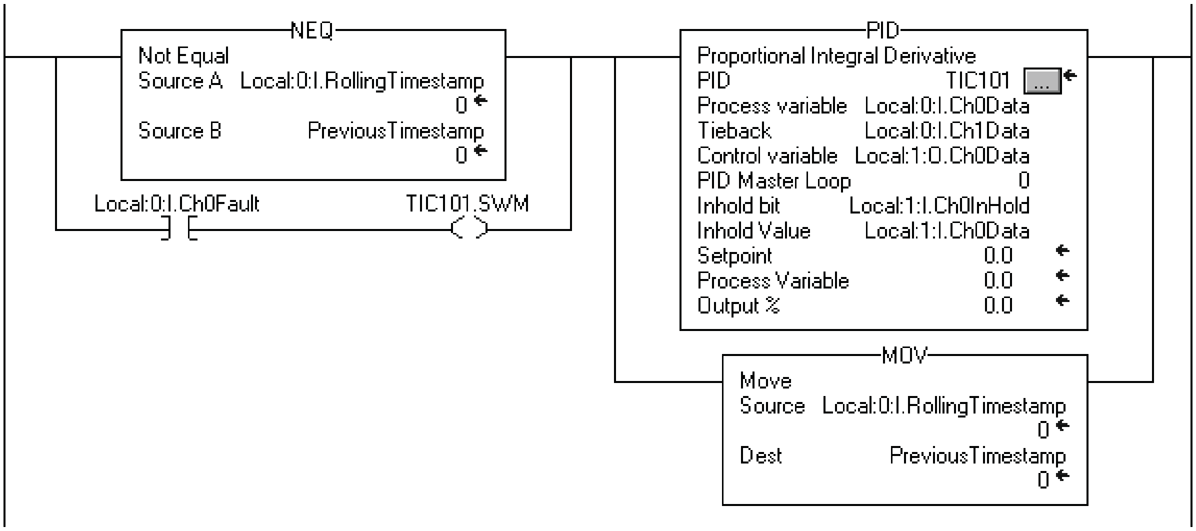
The most accurate way to execute a PID instruction is to use the real time sampling (RTS) feature of the 1756 analog input modules. The analog input module samples its inputs at the real time sampling rate you configure when you set up the module. When the module's real time sample period expires, it updates its inputs and updates a rolling timestamp (represented by the .RollingTimestamp member of the analog input data structure) produced by the module.

The timestamp ranges from 0…32,767 ms. Monitor the timestamp. When it changes, a new process variable sample has been received. Every time a timestamp changes, execute the PID instruction once. Because the process variable sample is driven by the analog input module, the input sample time is very accurate, and the loop update time used by the PID instruction should be set equal to the RTS time of the analog input module.

To make sure that you do not miss samples of the process variable, execute your logic at a rate faster than the RTS time. For example, if the RTS time is 250 ms, you could put the PID logic in a periodic task that runs every 100 ms to make sure that you never miss a sample. You could even place the PID logic in a continuous task, as long as you make sure that the logic would be updated more frequently than once every 250 ms.

An example of the RTS method of execution is shown below. The execution of the PID instruction depends on receiving new analog input data. If the analog input module fails or is removed, the controller stops receiving rolling timestamps and the PID loop stops executing. You should monitor the status bit of the PV analog input and, if it shows bad status, force the loop into software manual mode, and execute the loop every scan. This lets the operator still manually change the output of the PID loop.

Relay Ladder



Structured Text

```
IF (Local:0:I.Ch0Fault) THEN
     TIC101.SWM [:=] 1;
ELSE
     TIC101.SWM := 0;
END_IF;

IF (Local:0:I.RollingTimestamp<>PreviousTimestamp) OR
     (Local:0:I.Ch0Fault) THEN

     PreviousTimestamp := Local:0:I.RollingTimestamp;

     PID(TIC101,Local:0:I.Ch0Data,Local:0:I.Ch1Data,
     Local:1:O.Ch0Data,0,Local:1:I.Ch0InHold,
     Local:1:I.Ch0Data);

END_IF;
```

Bumpless Restart

The PID instruction can interact with the 1756 analog output modules to support a bumpless restart when the controller changes from Program to Run mode or when the controller powers up.

When a 1756 analog output module loses communications with the controller or senses that the controller is in Program mode, the analog output module sets its outputs to the fault condition values you specified when you configured the module. When the controller then returns to Run mode or re-establishes communications with the analog output module, you can have the PID instruction automatically reset its control variable output equal to the analog output by using the Inhold bit and Inhold Value parameters on the PID instruction.

Instructions for setting a bumpless restart.

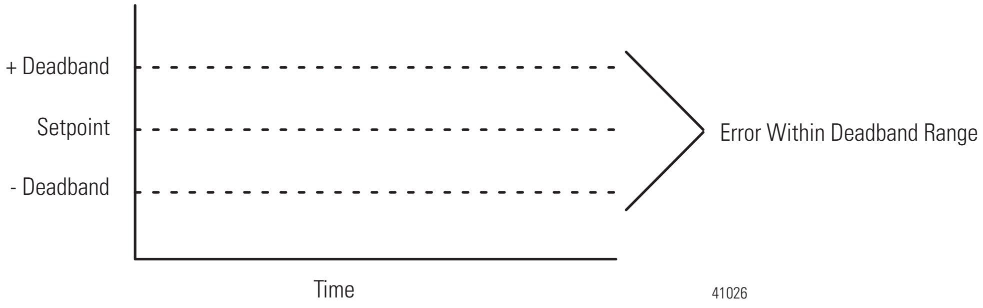
Do this	Details
Configure the 1756 analog output module's channel that receives the control variable from the PID instruction	Select the "hold for initialization" checkbox on the properties page for the specific channel of the module. This tells the analog output module that when the controller returns to Run mode or re-establishes communications with the module, the module should hold the analog output at its current value until the value sent from the controller matches (within 0.1% of span) the current value used by the output channel. The controller's output will ramp to the currently held output value by making use of the .BIAS term. This ramping is similar to auto bumpless transfer.
Enter the Inhold bit tag and Inhold Value tag in the PID instruction	The 1756 analog output module returns two values for each channel in its input data structure. The InHold status bit (.Ch2InHold, for example), when true, indicates that the analog output channel is holding its value. The Data readback value (.Ch2Data, for example) shows the current output value in engineering units. Enter the tag of the InHold status bit as the InHold bit parameter of the PID instruction. Enter the tag of the Data readback value as the Inhold Value parameter. When he Inhold bit goes true, the PID instruction moves the Inhold Value into the Control variable output and re-initializes to support a bumpless restart at that value. When the analog output module receives this value back from the controller, it turns off the InHold status bit, which allows the PID instruction to start controlling normally.

Derivative Smoothing

The derivative calculation is enhanced by a derivative smoothing filter. This first order, low pass, digital filter helps to minimize large derivative term spikes caused by noise in the PV. This smoothing becomes more aggressive with larger values of derivative gain. You can disable derivative smoothing if your process requires very large values of derivative gain ($K_d > 10$, for example). To disable derivative smoothing, select the "No derivative smoothing" option on the Configuration tab or set the .NDF bit in the PID structure.

Set the Deadband

The adjustable deadband lets you select an error range above and below the setpoint where output does not change as long as the error remains within this range. This deadband lets you control how closely the process variable matches the setpoint without changing the output. The deadband also helps to minimize wear and tear on your final control device.



Zero-crossing is deadband control that lets the instruction use the error for computational purposes as the process variable crosses into the deadband until the process variable crosses the setpoint. Once the process variable crosses the setpoint (error crosses zero and changes sign) and as long as the process variable remains in the deadband, the output will not change.

The deadband extends above and below the setpoint by the value you specify. Enter zero to inhibit the deadband. The deadband has the same scaled units as the setpoint. You can use the deadband without the zero-crossing feature by selecting the 'no zero crossing for deadband' option on the Configuration tab or set the .NOZC bit in the PID structure.

If you are using the deadband, the Control variable must be REAL or it will be forced to zero when the error is within the deadband.

Use Output Limiting

You can set an output limit (percentage of output) on the control output. When the instruction detects that the output has reached a limit, it sets an alarm bit and prevents the output from exceeding either the lower or upper limit.

Feedforward or Output Biasing

You can feedforward a disturbance from the system by feeding the .BIAS value into the PID instruction's feedforward/bias value.

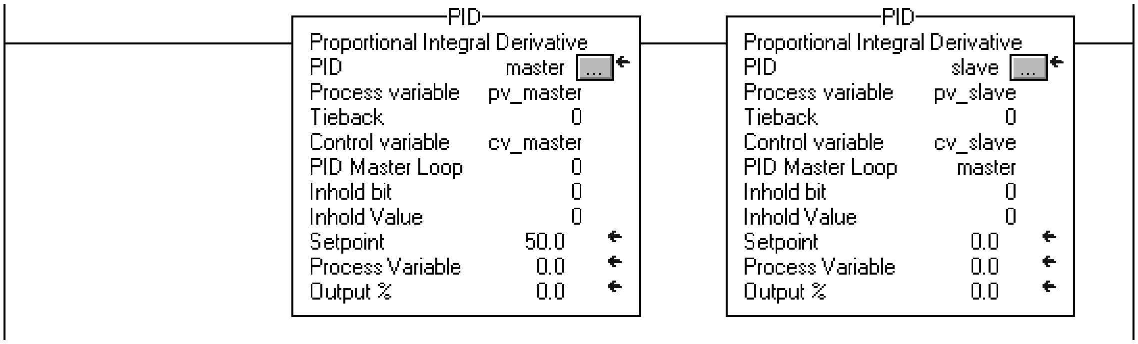
The feedforward value represents a disturbance fed into the PID instruction before the disturbance has a chance to change the process variable. Feedforward is often used to control processes with a transportation lag. For example, a feedforward value representing 'cold water poured into a warm mix' could boost the output value faster than waiting for the process variable to change as a result of the mixing.

A bias value is typically used when no integral control is used. In this case, the bias value can be adjusted to maintain the output in the range required to keep the PV near the setpoint.

Cascade Loops

The PID cascades two loops by assigning the output in percent of the master loop to the setpoint of the slave loop. The slave loop automatically converts the output of the master loop into the correct engineering units for the setpoint of the slave loop, based on the slave loop's values for .MAXS and .MINS.

Relay Ladder



PID
Proportional Integral Derivative
PID master
Process variable pv_master
Tieback 0
Control variable cv_master
PID Master Loop 0
Inhold bit 0
Inhold Value 0
Setpoint 50.0
Process Variable 0.0
Output % 0.0

PID
Proportional Integral Derivative
PID slave
Process variable pv_slave
Tieback 0
Control variable cv_slave
PID Master Loop master
Inhold bit 0
Inhold Value 0
Setpoint 0.0
Process Variable 0.0
Output % 0.0

Structured Text

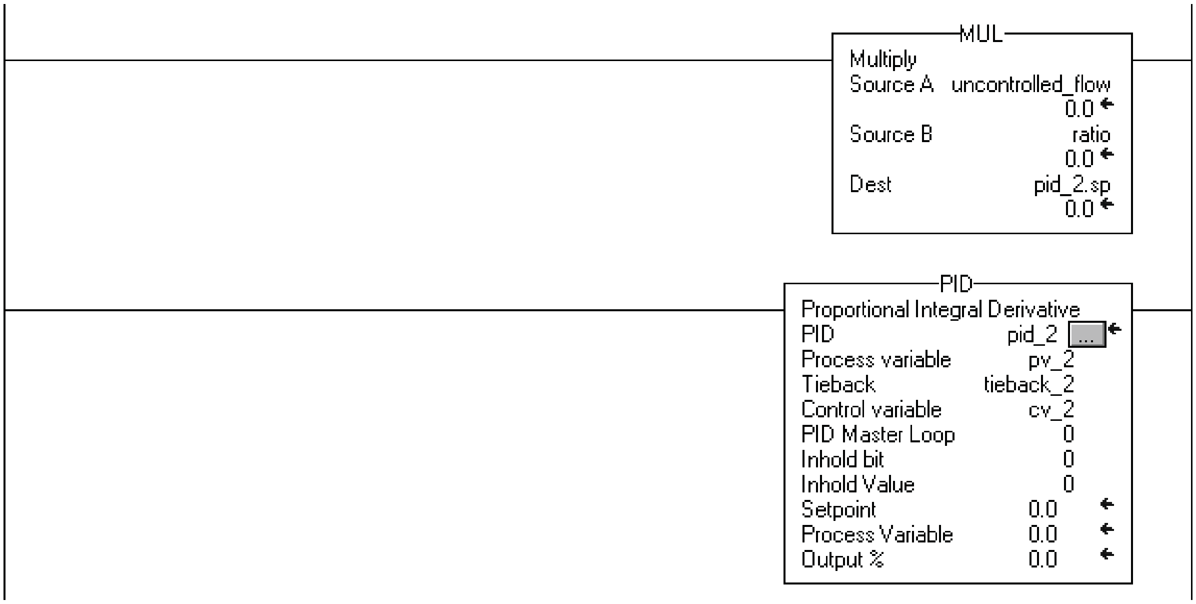
```
PID(master,pv_master,0,cv_master,0,0,0);

PID (slave,pv_slave,0,cv_slave,master,0,0);
```

Control a Ratio

You can maintain two values in a ratio by using these parameters:

- Uncontrolled value
- Controlled value (the resultant setpoint to be used by the PID instruction)
- Ratio between these two values

Relay Ladder



MUL
Multiply
Source A uncontrolled_flow
0.0
Source B ratio
0.0
Dest pid_2.sp
0.0

PID
Proportional Integral Derivative
PID pid_2
Process variable pv_2
Tieback tieback_2
Control variable cv_2
PID Master Loop 0
Inhold bit 0
Inhold Value 0
Setpoint 0.0
Process Variable 0.0
Output % 0.0

Structured Text

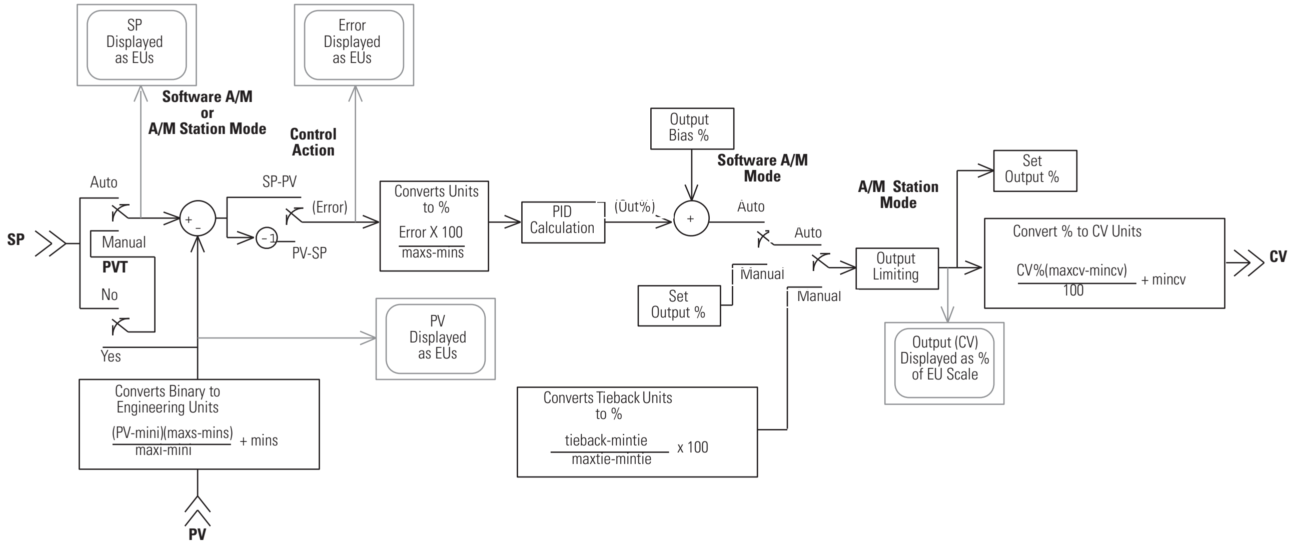
```
pid_2.sp := uncontrolled_flow * ratio

PID(pid_2,pv_2,tieback_2,cv_2,0,0,0);
```

For this multiplication parameter	Enter this value
Destination	Controlled value
Source A	Uncontrolled value
Source B	Ratio

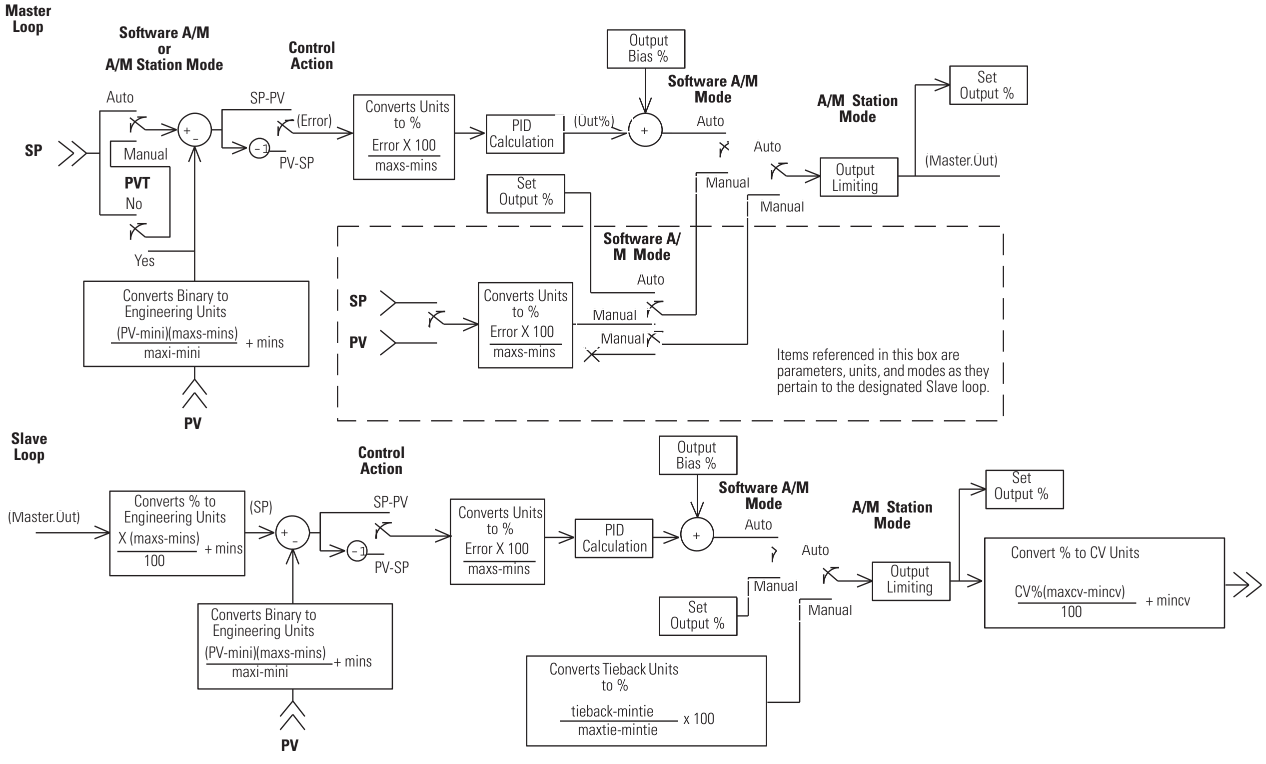
PID Theory

The following figures show the process flow for PID instructions.

PID Process



PID Process with Master/Slave Loops



Notes:

Trigonometric Instructions

(SIN, COS, TAN, ASN, ASIN, ACS, ACOS, ATN, ATAN)

Introduction

The trigonometric instructions evaluate arithmetic operations by using trigonometric operations.

If you want to	Use this instruction	Available in these languages	Page
Take the sine of a value	SIN	Relay ladder Structured text Function block	528
Take the cosine of a value	COS	Relay ladder Structured text Function block	531
Take the tangent of a value	TAN	Relay ladder Structured text Function block	534
Take the arc sine of a value	ASN ASIN(1)	Relay ladder Structured text Function block	537
Take the arc cosine of a value	ACS ACOS(1)	Relay ladder Structured text Function block	540
Take the arc tangent of a value	ATN ATAN(1)	Relay ladder Structured text Function block	543

(1) Structured text only.

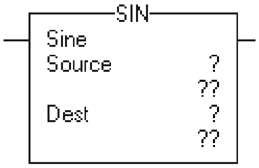
You can mix data types, but loss of accuracy and rounding error might occur and the instruction takes more time to execute. Check the overflow status bit (S:V) to see whether the result was truncated.

For relay ladder instructions, **bold** data types indicate optimal data types. An instruction executes faster and requires less memory if all the operands of the instruction use the same optimal data type, typically DINT or REAL.

Sine (SIN)

The SIN instruction takes the sine of the Source value (in radians) and stores the result in the Destination.

Operands:



Relay Ladder

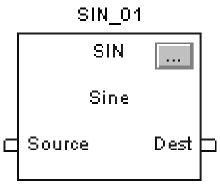
Operand	Type	Format	Description
Source	SINT INT DINT REAL	Immediate Tag	Find the sine of this value
Destination	SINT INT DINT REAL	Tag	Tag to store the result

Structured Text

```
dest := SIN(source);
```

Use SIN as a function. This function computes the sine of *source* and stores the result in *dest*.

See Structured Text Programming for information on the syntax of expressions within structured text.



Function Block

Operand	Type	Format	Description
SIN tag	FBD_MATH_ADVANCED	Structure	SIN structure

FBD_MATH_ADVANCED Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
Source	REAL	Input to the math instruction. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	REAL	Result of the math instruction. Arithmetic status flags are set for this output.

Description: The Source must be greater than or equal to -205887.4 ($-2\pi x 2^{15}$) and less than or equal to 205887.4 ($2\pi x 2^{15}$). The resulting value in the Destination is always greater than or equal to -1 and less than or equal to 1.

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions: None

Execution:

Relay Ladder

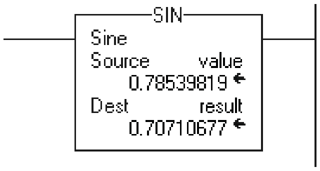
Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The controller calculates the sine of the Source and places the result in the Destination. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Iinstruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: Calculate the sine of *value* and place the result in *result*.

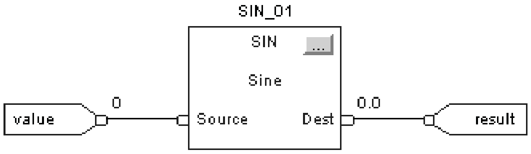
Relay Ladder



Structured Text

```
result := SIN(value);
```

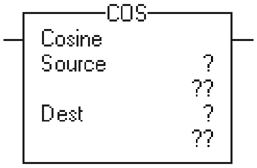
Function Block



Cosine (COS)

The COS instruction takes the cosine of the Source value (in radians) and stores the result in the Destination.

Operands:



Relay Ladder

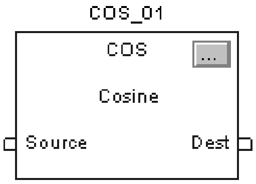
Operand	Type	Format	Description
Source	SINT INT DINT **REAL**	Immediate Tag	Find the cosine of this value
Destination	SINT INT DINT **REAL**	Tag	Tag to store the result

Structured Text

```
dest := COS(source);
```

Use COS as a function. This function computes the cosine of *source* and stores the result in *dest*.

See Structured Text Programming for information on the syntax of expressions within structured text.



Function Block

Operand	Type	Format	Description
COS tag	FBD_MATH_ADVANCED	Structure	COS structure

FBD_MATH_ADVANCED Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
Source	REAL	Input to the math instruction. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	REAL	Result of the math instruction. Arithmetic status flags are set for this output.

Description: The Source must be greater than or equal to -205887.4 ($-2\pi\text{x}2^{15}$) and less than or equal to 205887.4 ($2\pi\text{x}2^{15}$). The resulting value in the Destination is always greater than or equal to -1 and less than or equal to 1.

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions: None

Execution:

Relay Ladder

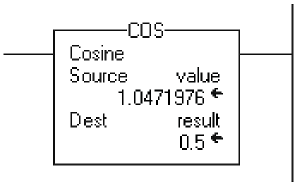
Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The controller calculates the cosine of the Source and places the result in the Destination. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: Calculate the cosine of *value* and place the result in *result*.

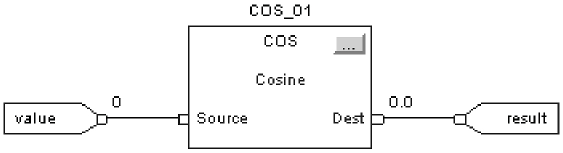
Relay Ladder



Structured Text

```
result := COS(value);
```

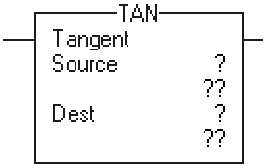
Function Block



Tangent (TAN)

The TAN instruction takes the tangent of the Source value (in radians) and stores the result in the Destination.

Operands:



Relay Ladder

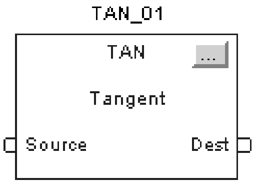
Operand	**Type**	**Format**	**Description**
Source	SINT INT DINT **REAL**	Immediate Tag	Find the tangent of this value
Destination	SINT INT DINT **REAL**	Tag	Tag to store the result

Structured Text

```
dest := TAN(source);
```

Use TAN as a function. This function computes the tangent of *source* and stores the result in *dest*.

See Structured Text Programming for information on the syntax of expressions within structured text.



Function Block

Operand	**Type**	**Format**	**Description**
TAN tag	FBD_MATH_ADVANCED	Structure	TAN structure

FBD_MATH_ADVANCED Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
Source	REAL	Input to the math instruction. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	REAL	Result of the math instruction. Arithmetic status flags are set for this output.

Description: The Source must be greater than or equal to $-102943.7(-2\pi x 2^{14})$ and less than or equal to $102943.7\ (2\pi x 2^{14})$.

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions: None

Execution:

Relay Ladder

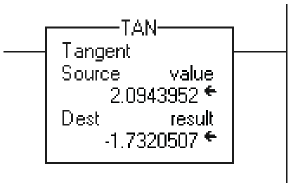
Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The controller calculates the tangent of the Source and places the result in the Destination. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: Calculate the tangent of *value* and place the result in *result*.

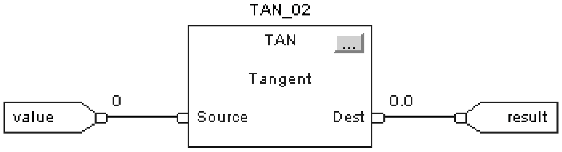
Relay Ladder



Structured Text

```
result := TAN(value);
```

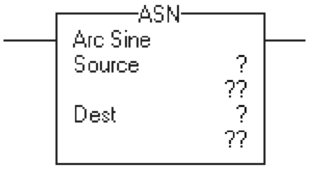
Function Block



Arc Sine (ASN)

The ASN instruction takes the arc sine of the Source value and stores the result in the Destination (in radians).

Operands:



Relay Ladder

Operand	Type	Format	Description
Source	SINT INT DINT REAL	Immediate Tag	Find the arc sine of this value
Destination	SINT INT DINT REAL	Tag	Tag to store the result

Structured Text

```
dest := ASIN(source);
```

Use ASIN as a function. This function computes the arc sine of *source* and stores the result in *dest*.

See Structured Text Programming for information on the syntax of expressions within structured text.

Function Block

Operand	Type	Format	Description
ASN tag	FBD_MATH_ADVANCED	Structure	ASN structure

FBD_MATH_ADVANCED Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
Source	REAL	Input to the math instruction. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	REAL	Result of the math instruction. Arithmetic status flags are set for this output.

Description: The Source must be greater than or equal to -1 and less than or equal to 1. The resulting value in the Destination is always greater than or equal to $-\pi/2$ and less than or equal to $\pi/2$ (where $\pi = 3.141593$).

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions: None

Execution:

Relay Ladder

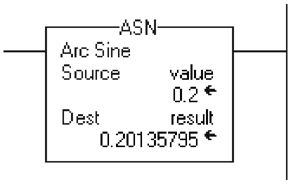
Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The controller calculates the arc sine of the Source and places the result in the Destination. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Function Block

Condition	**Action**
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: Calculate the arc sine of *value* and place the result in *result*.

Relay Ladder



Structured Text

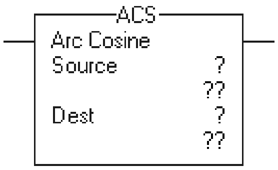
```
result := ASIN(value);
```

Function Block

Arc Cosine (ACS)

The ACS instruction takes the arc cosine of the Source value and stores the result in the Destination (in radians).

Operands:



Relay Ladder

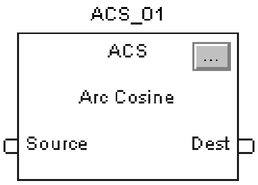
Operand	Type	Format	Description
Source	SINT INT DINT **REAL**	Immediate Tag	Find the arc cosine of this value
Destination	SINT INT DINT **REAL**	Tag	Tag to store the result

Structured Text

```
dest := ACOS(source);
```

Use ACOS as a function. This function computes the arc cosine of *source* and stores the result in *dest*.

See Structured Text Programming for information on the syntax of expressions within structured text.



Function Block

Operand	Type	Format	Description
ACS tag	FBD_MATH_ADVANCED	Structure	ACS structure

FBD_MATH_ADVANCED Structure

Input Parameter	**Data Type**	**Description:**
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
Source	REAL	Input to the math instruction. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	REAL	Result of the math instruction. Arithmetic status flags are set for this output.

Description: The Source must be greater than or equal to -1 and less than or equal to 1. The resulting value in the Destination is always greater than or equal to 0 or less than or equal to π (where π = 3.141593).

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions: None

Execution:

Relay Ladder

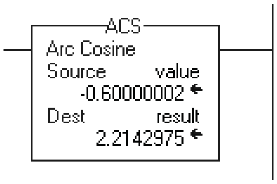
Condition	**Action**
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The controller calculates the arc cosine of the Source and places the result in the Destination. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Function Block

Condition:	**Action**
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: Calculate the arc cosine of *value* and place the result in *result*.

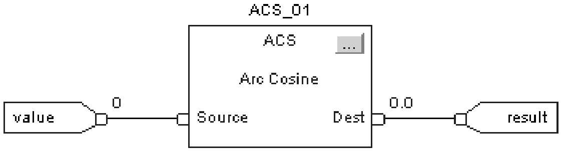
Relay Ladder



Structured Text

```
result := ACOS(value);
```

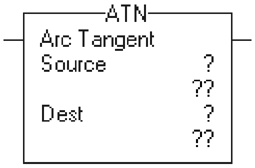
Function Block



Arc Tangent (ATN)

The ATN instruction takes the arc tangent of the Source value and stores the result in the Destination (in radians).

Operands:



Relay Ladder

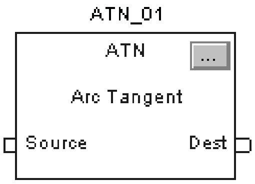
Operand:	Type	Format	Description
Source	SINT INT DINT **REAL**	Immediate Tag	Find the arc tangent of this value
Destination	SINT INT DINT **REAL**	Tag	Tag to store the result

Structured Text

```
dest := ATAN(source);
```

Use ATAN as a function. This function computes the arc tangent of *source* and stores the result in *dest*.

See Structured Text Programming for information on the syntax of expressions within structured text.



Function Block

Operand	Type	Format	Description
ATN tag	FBD_MATH_ADVANCED	Structure	ATN structure

FBD_MATH_ADVANCED Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
Source	REAL	Input to the math instruction. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	REAL	Result of the math instruction. Arithmetic status flags are set for this output.

Description: The resulting value in the Destination is always greater than or equal to $-\pi/2$ and less than or equal to $\pi/2$ (where $\pi = 3.141593$).

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions: None

Execution:

Relay Ladder

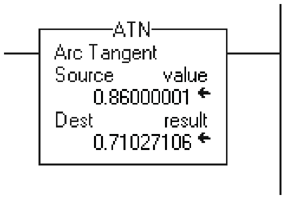
Condition	Action
Prescan	The rung-condition-out is set to false.
Rrung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The controller calculates the arc tangent of the Source and places the result in the Destination. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: Calculate the arc tangent of *value* and place the result in *result*.

Relay Ladder



Structured Text

```
result := ATAN(value);
```

Function Block

Notes:

Advanced Math Instructions

(LN, LOG, XPY)

Introduction

The advanced math instructions include these instructions.

If you want to	Use this instruction	Available in these languages	Page
Take the natural log of a value	LN	Relay ladder Structured text Function block	548
Take the log base 10 of a value	LOG	Relay ladder Structured text Function block	551
Raise a value to the power of another value	XPY	Relay ladder Structured text(1) Function block	554

(1) There is no equivalent structured text instruction. Use the operator in an expression.

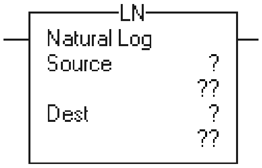
You can mix data types, but loss of accuracy and rounding error might occur and the instruction takes more time to execute. Check the S:V bit to see whether the result was truncated.

For relay ladder instructions, **bold** data types indicate optimal data types. An instruction executes faster and requires less memory if all the operands of the instruction use the same optimal data type, typically DINT or REAL.

Natural Log (LN)

The LN instruction takes the natural log of the Source and stores the result in the Destination.

Operands:



Relay Ladder

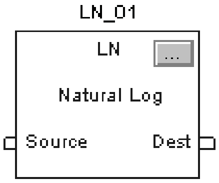
Operand	Type	Format	Description
Source	SINT INT DINT REAL	Immediate Tag	Find the natural log of this value
Destination	SINT INT DINT REAL	Tag	Tag to store the result

Structured Text

```
dest := LN(source);
```

Use LN as a function. This function computes the natural log of *source* and stores the result in *dest*.

See Structured Text Programming for information on the syntax of expressions within structured text.



Function Block

Operand	Type	Format	Description
LN tag	FBD_MATH_ADVANCED	Structure	LN structure

FBD_MATH_ADVANCED Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
Source	REAL	Input to math instruction. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.

Description: The Source must be greater than zero, otherwise the overflow status bit (S:V) is set. The resulting Destination is greater than or equal to -87.33655 and less than or equal to 88.72284.

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions: None

Execution:

Relay Ladder

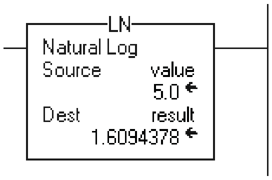
Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The controller calculates the natural log of the Source and places the result in the Destination. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: Calculate the natural log of *value* and place the result in *result*.

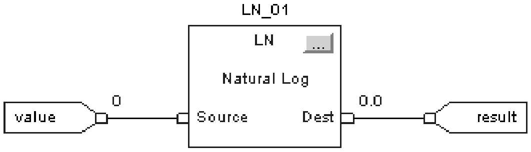
Relay Ladder Example



Structured Text

```
result := LN(value);
```

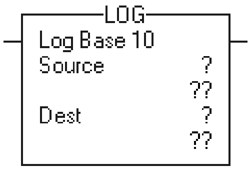
Function Block



Log Base 10 (LOG)

The LOG instruction takes the log base 10 of the Source and stores the result in the Destination.

Operands:



LOG
Log Base 10
Source ?
??
Dest ?
??

Relay Ladder

Operand	Type	Format	Description
Source	SINT INT DINT **REAL**	Immediate Tag	Find the log of this value
Destination	SINT INT DINT **REAL**	Tag	Tag to store the result

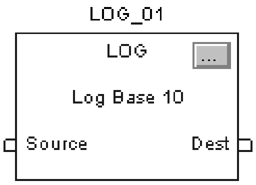
Structured Text

```
dest := LOG(source);
```

Use LOG as a function. This function computes the log of *source* and stores the result in *dest*.

See Structured Text Programming for information on the syntax of expressions within structured text.

Function Block



Operand	Type	Format	Description
LOG tag	FBD_MATH_ADVANCED	Structure	LOG structure

FBD_MATH_ADVANCED Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
Source	REAL	Input to math instruction. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	REAL	Result of the math instruction. Arithmetic status flags are set for this output.

Description: The Source must be greater than zero, otherwise the overflow status bit (S:V) is set. The resulting Destination is greater than or equal to -37.92978 and less than or equal to 38.53184.

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions: None

Execution:

Relay Ladder

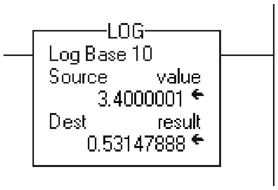
Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The controller calculates the log of the Source and places the result in the Destination. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: Calculate the log of *value* and place the result in *result*.

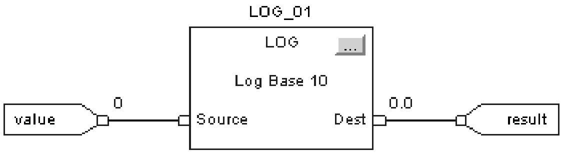
Relay Ladder



Structured Text

```
result := LOG(value);
```

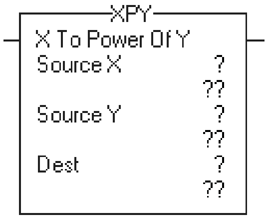
Function Block



X to the Power of Y (XPY)

The XPY instruction takes Source A (X) to the power of Source B (Y) and stores the result in the Destination.

Operands:



Relay Ladder

Operand	Type	Format	Description
Source X	SINT INT DINT **REAL**	Immediate Tag	Base value
Source Y	SINT INT DINT **REAL**	Immediate Tag	Exponent
Destination	SINT INT DINT **REAL**	Tag	Tag to store the result

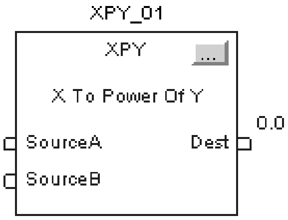
Structured Text

```
dest := sourceX ** sourceY;
```

Use two, adjacent multiply signs “**” as an operator within an expression. This expression takes *sourceX* to the power of *sourceY* and stores the result in *dest*.

See Structured Text Programming for information on the syntax of expressions within structured text.

Function Block



Operand	Type	Format	Description
XPY tag	FBD_MATH	Structure	XPY structure

FBD_MATH Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
Source X	REAL	Base value. Valid = any float
Source Y	REAL	Exponent. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	REAL	Result of the math instruction. Arithmetic status flags are set for this output.

Description:

If Source X is negative, Source Y must be an integer value or a minor fault will occur.

The XPY instruction uses this algorithm: *Destination* = X**Y

The controller evaluates $x^0=1$ and $0^x=0$.

Arithmetic Status Flags:

Arithmetic status flags are affected.

Fault Conditions:

A Minor Fault Will Occur If	Fault Type	Fault Code
Source X is negative and Source Y is not an integer value	4	4

Execution:

Relay Ladder

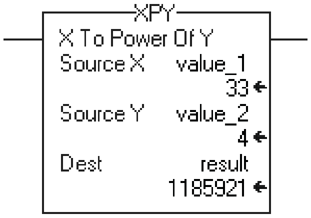
Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The controller takes Source X to the power of Source Y and places the result in the Destination. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: The XPY instruction takes *value_1* to the power of *value_2* and places the result in *result*.

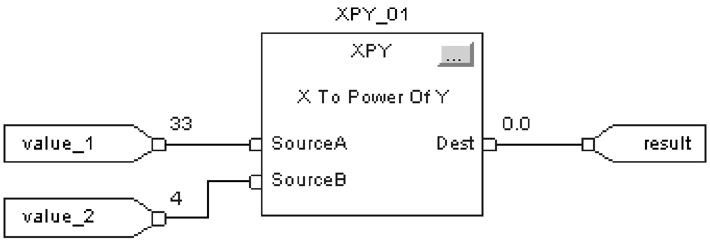
Relay Ladder



Structured Text

```
result := (value_1 ** value_2);
```

Function Block



Math Conversion Instructions

(DEG, RAD, TOD, FRD, TRN, TRUNC)

Introduction

The math conversion instructions convert values.

If you want to	Use this instruction	Available in these languages	Page
Convert radians to degrees	DEG	Relay ladder Structured text Function block	558
Convert degrees to radians	RAD	Relay ladder Structured text Function block	561
Convert an integer value to a BCD value	TOD	Relay ladder Function block	564
Convert a BCD value to an integer value	FRD	Relay ladder Function block	567
Remove the fractional part of a value	TRN TRUNC(1)	Relay ladder Structured text Function block	569

(1) Structured text only.

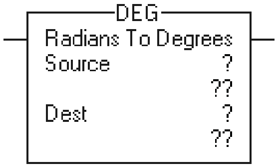
You can mix data types, but loss of accuracy and rounding error might occur and the instruction takes more time to execute. Check the S:V bit to see whether the result was truncated.

For relay ladder instructions, **bold** data types indicate optimal data types. An instruction executes faster and requires less memory if all the operands of the instruction use the same optimal data type, typically DINT or REAL.

Degrees (DEG)

The DEG instruction converts the Source (in radians) to degrees and stores the result in the Destination.

Operands:



Relay Ladder

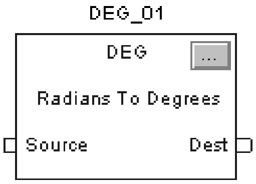
Operand	Type	Format	Description
Source	SINT INT DINT REAL	Immediate Tag	Value to convert to degrees
Destination	SINT INT DINT REAL	Tag	Tag to store the result

Structured Text

```
dest := DEG(source);
```

Use DEG as a function. This function converts *source* to degrees and stores the result in *dest*.

See Structured Text Programming for information on the syntax of expressions within structured text.



Function Block

Operand	Type	Format	Description
DEG tag	FBD_MATH_ADVANCED	Structure	DEG structure

FBD_MATH_ADVANCED Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
Source	REAL	Input to the conversion instruction. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	REAL	Result of the conversion instruction. Arithmetic status flags are set for this output.

Description: The DEG instruction uses this algorithm:
Source*180/π (where $\pi = 3.141593$)

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions: None

Execution:

Relay Ladder

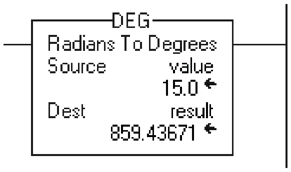
Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The controller converts the Source to degrees and places the result in the Destination. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: Convert *value* to degrees and place the result in *result*.

Relay Ladder

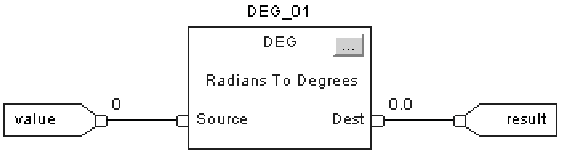


DEG
Radians To Degrees
Source value
15.0
Dest result
859.43671

Structured Text

```
result := DEG(value);
```

Function Block

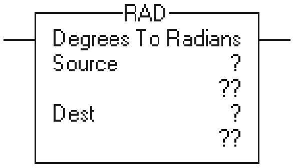


Radians (RAD)

The RAD instruction converts the Source (in degrees) to radians and stores the result in the Destination.

Operands:

Relay Ladder



Operand	Type	Format	Description
Source	SINT INT DINT REAL	Immediate Tag	Value to convert to radians
Destination	SINT INT DINT REAL	Tag	Tag to store the result

Structured Text

```
dest := RAD(source);
```

Use RAD as a function. This function converts *source* to radians and stores the result in *dest*.

See Structured Text Programming for information on the syntax of expressions within structured text.

Function Block

Operand	Type	Format	Description
RAD tag	FBD_MATH_ADVANCED	structure	RAD structure

FBD_MATH_ADVANCED Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
Source	REAL	Input to the conversion instruction. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	REAL	Result of the conversion instruction. Arithmetic status flags are set for this output.

Description: The RAD instruction uses this algorithm:
Source*π/180 (where $\pi = 3.141593$)

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions: None

Execution:

Relay Ladder

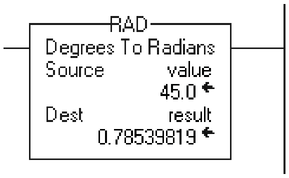
Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The controller converts the Source to radians and places the result in the Destination. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example Convert *value* to radians and place the result in *result*.

Relay Ladder



Source value
45.0
Dest result
0.78539819

Structured Text

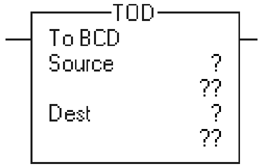
```
result := RAD(value);
```

Function Block

Convert to BCD (TOD)

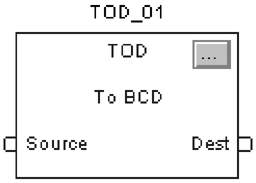
The TOD instruction converts a decimal value (0 ≤ Source ≤ 99,999,999) to a BCD value and stores the result in the Destination.

Operands:



Relay Ladder

Operand	Type	Format	Description
Source	SINT INT DINT	Immediate Tag	Value to convert to decimal
	A SINT or INT tag converts to a DINT value by zero-fill.		
Destination	SINT INT DINT	Tag	Stores the result



Function Block

Operand	Type	Format	Description
TOD tag	FBD_CONVERT	Structure	TOD structure

FBD_CONVERT Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
Source	DINT	Input to the conversion instruction. Valid = any integer
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	DINT	Result of the conversion instruction. Arithmetic status flags are set for this output.

Description: BCD is the Binary Coded Decimal number system that expresses individual decimal digits (0…9) in a 4-bit binary notation.

If you enter a negative Source, the instruction generates a minor fault and clears the Destination.

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions:

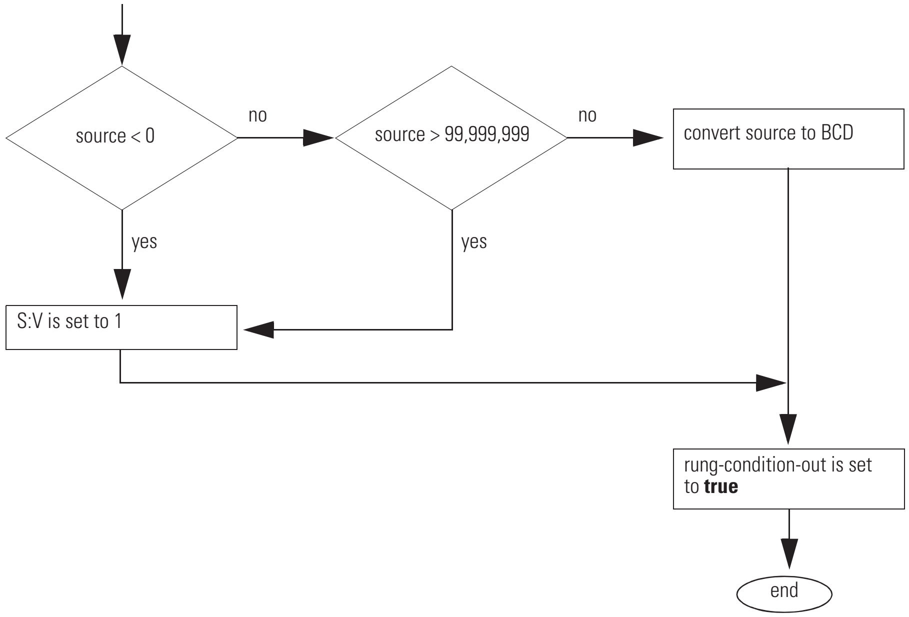
A minor fault will occur if	Fault Type	Fault Code
Source < 0	4	4

Execution:

Relay Ladder

Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.

Rung-condition-in is true



Rung-condition-in is true	The controller converts the Source to BCD and places the result in the Destination. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

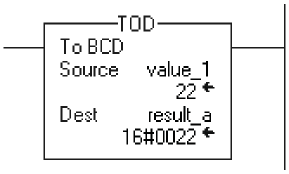
Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.

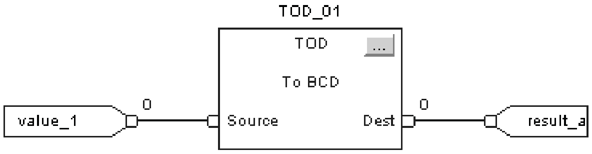
Condition	Action
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: The TOD instruction converts *value_1* to a BCD value and places the result in *result_a*.

Relay Ladder



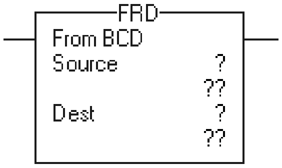
Function Block



Convert to Integer (FRD)

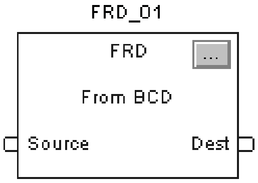
The FRD instruction converts a BCD value (Source) to a decimal value and stores the result in the Destination.

Operands:



Relay Ladder

Operand	Type	Format	Description
Source	SINT INT DINT	Immediate Tag	Value to convert to decimal
	A SINT or INT tag converts to a DINT value by zero-fill.		
Destination	SINT INT DINT	Tag	Stores the result



Function Block

Operand	Type	Format:	Description
FRD tag	FBD_CONVERT	Structure	FRD structure

FBD_CONVERT Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
Source	DINT	Input to the conversion instruction. Valid = any integer
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	DINT	Result of the conversion instruction. Arithmetic status flags are set for this output.

Description: The FRD instruction converts a BCD value (Source) to a decimal value and stores the result in the Destination.

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions: None

Execution:

Relay Ladder

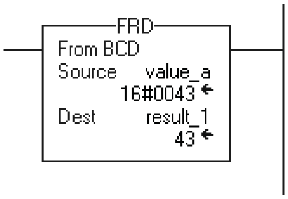
Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The controller converts the Source to a decimal value and places the result in the Destination. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Function Block

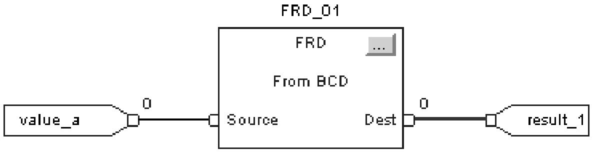
Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: The FRD instruction converts *value_a* to a decimal value and places the result in *result_1*.

Relay Ladder



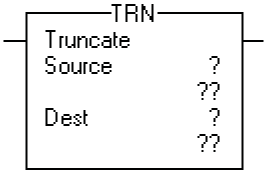
Function Block



Truncate (TRN)

The TRN instruction removes (truncates) the fractional part of the Source and stores the result in the Destination.

Operands:



Relay Ladder

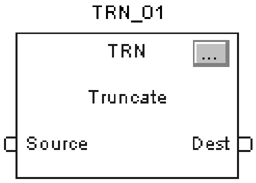
Operand	Type	Format	Description
Source	REAL	Immediate Tag	Value to truncate
Destination	SINT INT DINT REAL	Tag	Tag to store the result

Structured Text

```
dest := TRUNC(source);
```

Use TRUNC as a function. This function truncates *source* and stores the result in *dest*.

See Structured Text Programming for information on the syntax of expressions within structured text.



Function Block

Operand	Type	Format	Description
TRN tag	FBD_TRUNCATE	Structure	TRN structure

FBD_TRUNCATE Structure

Input Parameter	Data Type	Description
EnableIn	BOOL	Enable input. If cleared, the instruction does not execute and outputs are not updated. Default is set.
Source	REAL	Input to the conversion instruction. Valid = any float
Output Parameter	**Data Type**	**Description**
EnableOut	BOOL	The instruction produced a valid result.
Dest	DINT	Result of the conversion instruction. Arithmetic status flags are set for this output.

Description: Truncating does not round the value; rather, the non-fractional part remains the same regardless of the value of the fractional part.

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions: None

Execution:

Relay Ladder

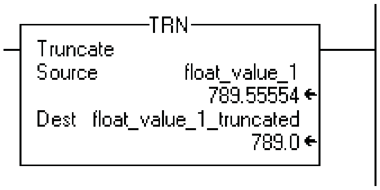
Condition	Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The controller removes the fractional part of the Source and places the result in the Destination. The rung-condition-out is set to true.
Postscan	The rung-condition-out is set to false.

Function Block

Condition	Action
Prescan	No action taken.
Instruction first scan	No action taken.
Instruction first run	No action taken.
EnableIn is cleared	EnableOut is cleared.
EnableIn is set	The instruction executes. EnableOut is set.
Postscan	No action taken.

Example: Remove the fractional part of *float_value_1*, leaving the non-fractional part the same, and place the result in *float_value_1_truncated*.

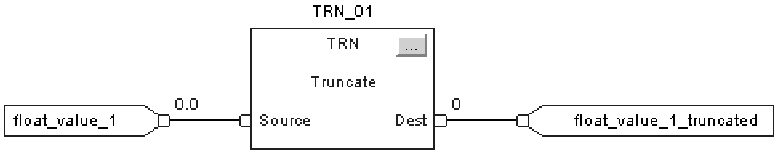
Relay Ladder



Structured Text

```
float_value_1_truncated := TRUNC(float_value_1);
```

Function Block



Notes:

ASCII Serial Port Instructions

(ABL, ACB, ACL, AHL, ARD, ARL, AWA, AWT)

Introduction

Use the ASCII serial port instructions to read and write ASCII characters.

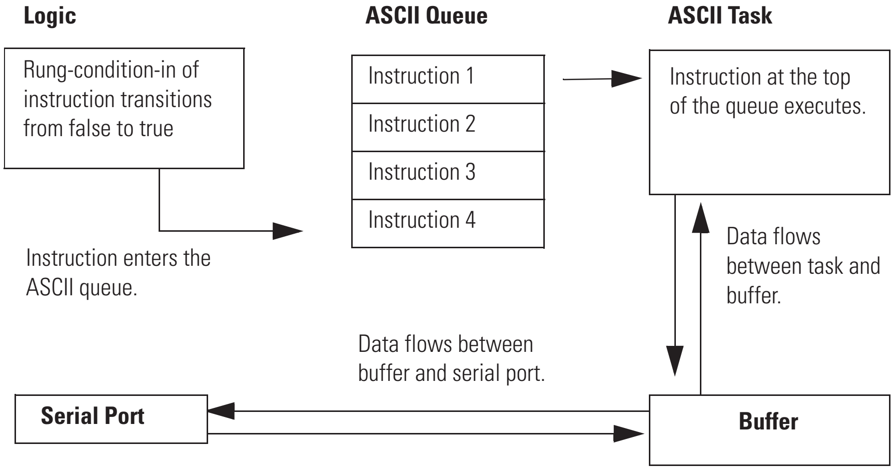
IMPORTANT To use the ASCII serial port instructions, you must configure the serial port of the controller. For procedures, see the Logix5000 Controllers Common Procedures Programming Manual, publication 1756-PM001.

The 1756-L7*x* controllers do not have a serial port and do not use the ASCII Read/Write instructions. In addition, you cannot redirect an ASCII Read/Write instruction to the USB port.

If you want to	For example	Use this instruction	Available in these languages	Page
Determine when the buffer contains termination characters	Check for data that contains termination characters	ABL	Relay ladder Structured text	578
Count the characters in the buffer	Check for the required number of characters before reading the buffer	ACB	Relay ladder Structured text	581
Clear the buffer	• Remove old data from the buffer at start-up. • Synchronize the buffer with a device.	ACL	Relay ladder Structured text	583
Clear out ASCII Serial Port instructions that are currently executing or are in the queue				
Obtain the status of the serial port control lines	Cause a modem to hang up	AHL	Relay ladder Structured text	585
Turn on or off the DTR signal				
Turn on or off the RTS signal				
Read a fixed number of characters	Read data from a device that sends the same number of characters each transmission	ARD	Relay ladder Structured text	589
Read a varying number of characters, up to and including the first set of termination characters	Read data from a device that sends a varying number of characters each transmission	ARL	Relay ladder Structured text	593
Send characters and automatically append one or two additional characters to mark the end of the data	Send messages that always use the same termination character(s)	AWA	Relay ladder Structured text	597
Send characters	Send messages that use a variety of termination characters	AWT	Relay ladder Structured text	602

Instruction Execution

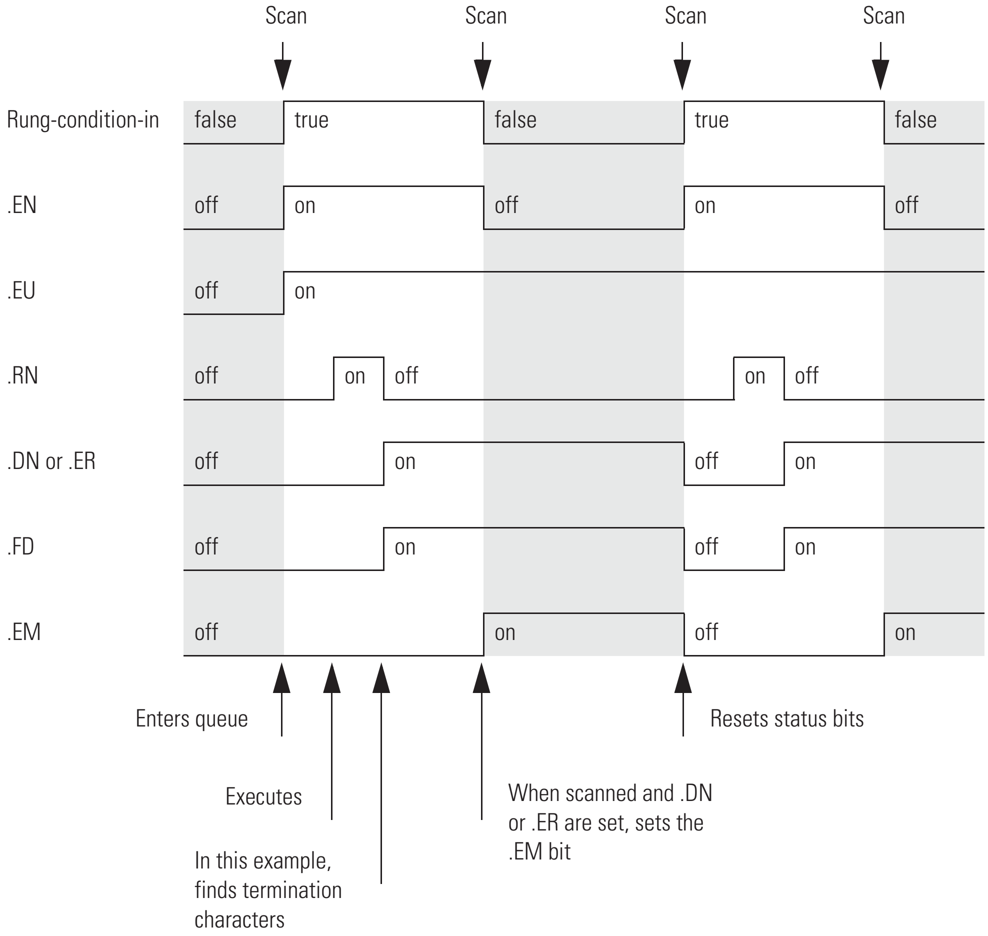
ASCII serial port instructions execute asynchronous to the scan of the logic.



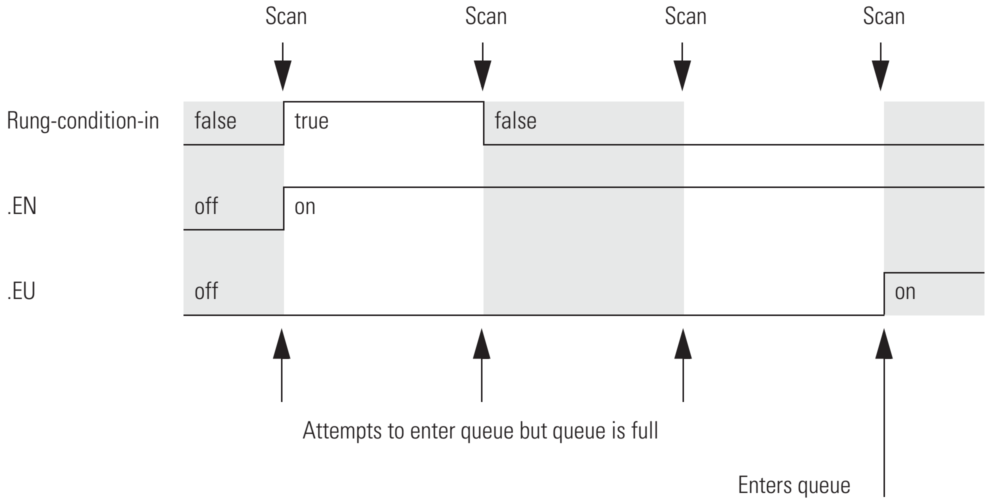
Each ASCII serial port instruction (except ACL) uses a SERIAL_PORT_CONTROL structure to perform these functions:

- Control the execution of the instruction
- Provide status information about the instruction

The following timing diagram depicts the changes in the status bits as an ABL instruction tests the buffer for termination characters.



The ASCII queue holds up to 16 instructions. When the queue is full, an instruction tries to enter the queue on each subsequent scan of the instruction, as depicted below.

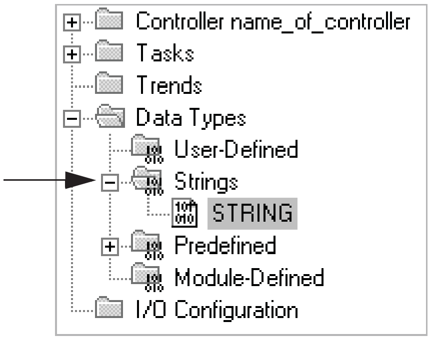


ASCII Error Codes

If an ASCII serial port instruction fails to execute, the ERROR member of its SERIAL_PORT_CONTROL structure will contain one of the following hexadecimal error codes.

This hex code	Indicates that the
16#2	Modem went offline.
16#3	CTS signal was lost during communication.
16#4	Serial port was in system mode.
16#A	Before the instruction executed, the .UL bit was set. This prevents the execution of the instruction.
16#C	The controller changed from Run mode to Program mode. This stops the execution of an ASCII serial port instruction and clears the queue.
16#D	In the Controller Properties dialog box, User Protocol tab, the buffer size or echo mode parameters were changed and applied. This stops the execution of an ASCII serial port instruction and clears the queue.
16#E	ACL instruction executed.
16#F	Serial port configuration changed from User mode to System mode. This stops the execution of an ASCII serial port instruction and clears the ASCII serial port instruction queue.
16#51	The LEN value of the string tag is either negative or greater than the DATA size of the string tag.
16#54	The Serial Port Control Length is greater than the size of the buffer.
16#55	The Serial Port Control Length is either negative or greater than the size of the Source or Destination.

String Data Types



You store ASCII characters in tags that use a string data type.

- You can use the default STRING data type. It stores up to 82 characters.
- You can create a new string data type that stores less or more characters.

To create a new string data type, see the Logix5000 Controllers Common Procedures Programming Manual, publication 1756-PM001.

Each string data type contains the following members.

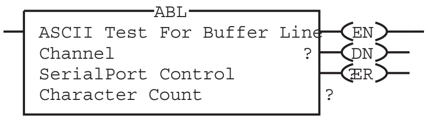
Name	Data Type	Description	Notes
LEN	DINT	Number of characters in the string	The LEN automatically updates to the new count of characters whenever you: • use the String Browser dialog box to enter characters. • use instructions that read, convert, or manipulate a string. The LEN shows the length of the current string. The DATA member may contain additional, old characters, which are not included in the LEN count.
DATA	SINT array	ASCII characters of the string	• To access the characters of the string, address the name of the tag. For example, to access the characters of the *string_1* tag, enter *string_1*. • Each element of the DATA array contains one character. • You can create new string data types that store less or more characters.

ASCII Test For Buffer Line (ABL)

The ABL instruction counts the characters in the buffer up to and including the first termination character.

Operands:

Relay Ladder



Operand	Type	Format	Description
Channel	DINT	Immediate Tag	0
Serial Port Control	SERIAL_PORT_ CONTROL	Tag	Tag that controls the operation
Character Count	DINT	Immediate	0 During execution, displays the number of characters in the buffer, including the first set of termination characters.

Structured Text

```
ABL(Channel
    SerialPortControl);
```

The operands are the same as those for the relay ladder ABL instruction. You access the Character Count value via the .POS member of the SERIAL_PORT_CONTROL structure.

SERIAL_PORT_CONTROL Structure

Mnemonic	Data Type	Description
.EN	BOOL	The enable bit indicates that the instruction is enabled.
.EU	BOOL	The queue bit indicates that the instruction entered the ASCII queue.
.DN	BOOL	The done bit indicates when the instruction is done, but it is asynchronous to the logic scan.
.RN	BOOL	The run bit indicates that the instruction is executing.
.EM	BOOL	The empty bit indicates that the instruction is done, but it is synchronous to the logic scan.
.ER	BOOL	The error bit indicates when the instruction fails (errors).
.FD	BOOL	The found bit indicates that the instruction found the termination character or characters.
.POS	DINT	The position determines the number of characters in the buffer, up to and including the first set of termination characters. The instruction only returns this number after it finds the termination character or characters.
.ERROR	DINT	The error contains a hexadecimal value that identifies the cause of an error.

Description

The ABL instruction searches the buffer for the first set of termination characters. If the instruction finds the termination characters, it:

- sets the .FD bit.
- counts the characters in the buffer up to and including the first set of termination characters.

The Controller Properties dialog box, User Protocol tab, defines the ASCII characters that the instruction considers as the termination characters.

Follow these guidelines to program the ABL instruction.

1. Configure the serial port of the controller for user mode and define the characters that serve as the termination characters.
2. This is a transitional instruction.
   - In relay ladder, toggle the rung-condition-in from cleared to set each time the instruction should execute.
   - In structured text, condition the instruction so that it executes only on a transition.

Arithmetic Status Flags:

Not affected

Fault Conditions:

None

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes when rung-condition-in toggles from cleared to set. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction counts the characters in the buffer. The .EN bit is set. The remaining status bits, except .UL, are cleared. The instruction attempts to enter the ASCII queue.	
Postscan	The rung-condition-out is set to false.	No action taken.

Example: Continuously test the buffer for the termination characters.

Relay Ladder

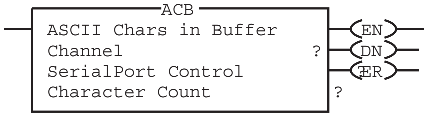
Structured Text

```
ABL(0,MV_line);
```

ASCII Chars in Buffer (ACB)

The ACB instruction counts the characters in the buffer.

Operands:



Relay Ladder

Operand	Type	Format	Enter
Channel	DINT	Immediate Tag	0
Serial Port Control	SERIAL_PORT_CONTROL	Tag	Tag that controls the operation
Character Count	DINT	Immediate	0 During execution, displays the number of characters in the buffer.

Structured Text

```
ACB(Channel
    SerialPortControl);
```

The operands are the same as those for the relay ladder ACB instruction. However, you specify the Character Count value by accessing the .POS member of the SERIAL_PORT_CONTROL structure, rather than by including the value in the operand list.

SERIAL_PORT_CONTROL Structure

Mnemonic	Data Type	Description
.EN	BOOL	The enable bit indicates that the instruction is enabled.
.EU	BOOL	The queue bit indicates that the instruction entered the ASCII queue.
.DN	BOOL	The done bit indicates when the instruction is done, but it is asynchronous to the logic scan.
.RN	BOOL	The run bit indicates that the instruction is executing.
.EM	BOOL	The empty bit indicates that the instruction is done, but it is synchronous to the logic scan.
.ER	BOOL	The error bit indicates when the instruction fails (errors).
.FD	BOOL	The found bit indicates that the instruction found a character.
.POS	DINT	The position determines the number of characters in the buffer, up to and including the first set of termination characters.
.ERROR	DINT	The error contains a hexadecimal value that identifies the cause of an error.

Description: The ACB instruction counts the characters in the buffer.

Follow these guidelines to program the ACB instruction.

1. Configure the serial port of the controller for user mode.
2. This is a transitional instruction.
    - In relay ladder, toggle the rung-condition-in from cleared to set each time the instruction should execute.
    - In structured text, condition the instruction so that it executes only on a transition.

Arithmetic Status Flags: Not affected

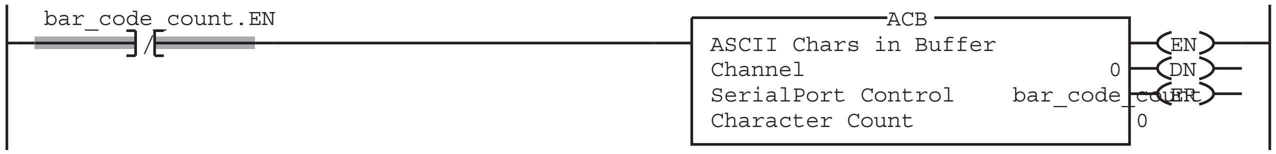
Fault Conditions: None

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes when rung-condition-in toggles from cleared to set. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction counts the characters in the buffer. The .EN bit is set. The remaining status bits, except .UL, are cleared. The instruction attempts to enter the ASCII queue.	
Postscan	The rung-condition-out is set to false.	No action taken.

Example: Continuously count the characters in the buffer.

Relay Ladder



Structured Text

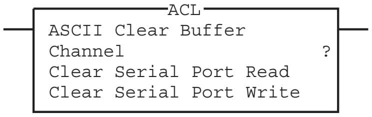
```
ACB(0,bar_code_count);
```

ASCII Clear Buffer (ACL)

The ACL instruction immediately clears the buffer and ASCII queue.

Operands:

Relay Ladder



Operand	Type	Format	Enter
Channel	DINT	Immediate Tag	0
Clear Serial Port Read	BOOL	Immediate Tag	To empty the buffer and remove ARD and ARL instructions from the queue, enter Yes.
Clear Serial Port Write	BOOL	Immediate Tag	To remove AWA and AWT instructions from the queue, enter Yes.

Structured Text

```
ACL(Channel,
     ClearSerialPortRead,
     ClearSerialPortWrite);
```

The operands are the same as those for the relay ladder ACL instruction.

Description:

The ACL instruction immediately performs one or both of the following actions:

- Clears the buffer of characters and clears the ASCII queue of read instructions
- Clears the ASCII queue of write instructions

Follow these guidelines to program the ACL instruction.

1. Configure the serial port of the controller:

If Your Application	Then
Uses ARD or ARL instructions	Select User mode
Does not use ARD or ARL instructions	Select either System or User mode

2. To determine if an instruction was removed from the queue or aborted, examine the following of the appropriate instruction.
   - .ER bit is set.
   - .ERROR member is 16#E.

Arithmetic Status Flags:

Not affected

Fault Conditions:

None

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction clears the specified instructions and buffer(s).	
Postscan	The rung-condition-out is set to false.	No action taken.

Example: When the controller enters Run mode, clear the buffer and the ASCII queue.

Relay Ladder

Structured Text

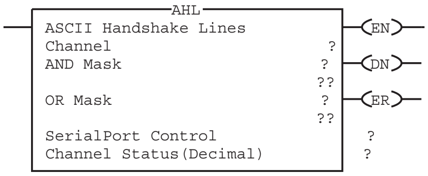
```
osri_1.InputBit := S:FS;

OSRI(osri_1);


IF (osri_1.OutputBit) THEN

     ACL(0,0,1);

END_IF;
```

ASCII Handshake Lines (AHL)

The AHL instruction obtains the status of control lines and turns on or off the DTR and RTS signals.

Operands:

Relay Ladder



<table>
<tr><th>Operand</th><th>Type</th><th>Format</th><th>Enter</th></tr>
<tr><td>Channel</td><td>DINT</td><td>Immediate
Tag</td><td>0</td></tr>
<tr><td>ANDMask</td><td>DINT</td><td>Immediate
Tag</td><td rowspan="2">Refer to the description.</td></tr>
<tr><td>ORMask</td><td>DINT</td><td>Immediate
Tag</td></tr>
<tr><td>Serial Port Control</td><td>SERIAL_PORT_CONTROL</td><td>Tag</td><td>Tag that controls the operation</td></tr>
<tr><td>Channel Status (Decimal)</td><td>DINT</td><td>Immediate</td><td>0
During execution, displays the status of the control lines.
<table>
<tr><th>For the Status Of This Control Line</th><th>Examine This Bit:</th></tr>
<tr><td>CTS</td><td>0</td></tr>
<tr><td>RTS</td><td>1</td></tr>
<tr><td>DSR</td><td>2</td></tr>
<tr><td>DCD</td><td>3</td></tr>
<tr><td>DTR</td><td>4</td></tr>
<tr><td>Received the XOFF character</td><td>5</td></tr>
</table></td></tr>
</table>

Structured Text

```
AHL(Channel,ANDMask,ORMask,
     SerialPortControl);
```

The operands are the same as those for the relay ladder AHL instruction. However, you specify the Channel Status value by accessing the .POS member of the SERIAL_PORT_CONTROL structure, rather than by including the value in the operand list.

SERIAL_PORT_CONTROL Structure

Mnemonic	Data Type	Description
.EN	BOOL	The enable bit indicates that the instruction is enabled.
.EU	BOOL	The queue bit indicates that the instruction entered the ASCII queue.
.DN	BOOL	The done bit indicates when the instruction is done, but it is asynchronous to the logic scan.
.RN	BOOL	The run bit indicates that the instruction is executing.
.EM	BOOL	The empty bit indicates that the instruction is done, but it is synchronous to the logic scan.
.ER	BOOL	The error bit indicates when the instruction fails (errors).
.FD	BOOL	The found bit does not apply to this instruction.
.POS	DINT	The position stores the status of the control lines.
.ERROR	DINT	The error contains a hexadecimal value that identifies the cause of an error.

Description: The AHL instruction can:

- obtain the status of the control lines of the serial port.
- turn on or off the data terminal ready (DTR) signal.
- turn on or off the request to send signal (RTS).

Follow these guidelines to program the AHL instruction.

1. Configure the serial port of the controller.

If your application	Then
Uses ARD or ARL instructions	Select User mode
Does not use ARD or ARL instructions	Select either System or User mode

2. Use the following table to select the correct values for the ANDMask and ORMask operands.

To turn DTR	And turn RTS	Enter this ANDMask value	And enter this ORMask value
Off	Off	3	0
	On	1	2
	Unchanged	1	0
On	Off	2	1
	On	0	3
	Unchanged	0	1
Unchanged	Off	2	0
	On	0	2
	Unchanged	0	0

3. This is a transitional instruction.
    - In relay ladder, toggle the rung-condition-in from cleared to set each time the instruction should execute.
    - In structured text, condition the instruction so that it executes only on a transition.

Arithmetic Status Flags: Not affected

Fault Conditions:

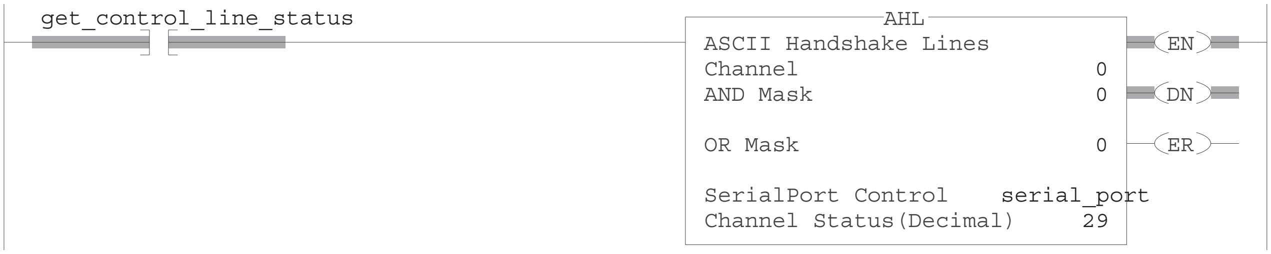
Type	Code	Cause	Recovery Method
4	57	The AHL instruction failed to execute because the serial port is set to no handshaking.	Either: • Change the Control Line setting of the serial port. • Delete the AHL instruction.

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes when rung-condition-in toggles from cleared to set. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction obtains the control line status and turns on or off DTR and RTS signals. The .EN bit is set. The remaining status bits, except .UL, are cleared. The instruction attempts to enter the ASCII queue.	
Postscan	The rung-condition-out is set to false.	No action taken.

Example: When *get_control_line_status* becomes set, obtain the status of the control lines of the serial port and store the status in the Channel Status operand. To view the status of a specific control line, monitor the SerialPortControl tag and expand the POS member.

Relay Ladder



Structured Text

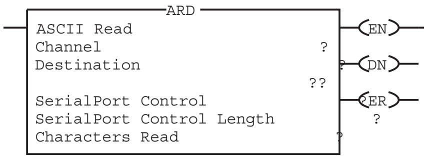
```
osri_1.InputBit := get_control_line_status;

OSRI(osri_1);


IF (osri_1.OutputBit) THEN

    AHL(0,0,0,serial_port);

END_IF;
```

ASCII Read (ARD)

The ARD instruction removes characters from the buffer and stores them in the Destination.

Operands:

Relay Ladder



Operand	Type	Format	Enter	Notes
Channel	DINT	Immediate Tag	0	
Destination	**String** SINT INT DINT	Tag	Tag into which the characters are moved (read): • For a string data type, enter the name of the tag. • For a SINT, INT, or DINT array, enter the first element of the array.	• If you want to compare, convert, or manipulate the characters, use a string data type. • String data types are: • default STRING data type • any new string data type that you create
Serial Port Control	SERIAL_PORT_CONTROL	Tag	Tag that controls the operation	
Serial Port Control Length	DINT	Immediate	Number of characters to move to the destination (read)	• The Serial Port Control Length must be less than or equal to the size of the Destination. • If you want to set the Serial Port Control Length equal to the size of the Destination, enter 0.
Characters Read	DINT	Immediate	0	During execution, displays the number of characters that were read.

Structured Text

```
ARD(Channel,Destination,
     SerialPortControl);
```

The operands are the same as those for the relay ladder ARD instruction. However, you specify the Serial Port Control Length and the Characters Read values by accessing the .LEN and .POS members of the SERIAL_PORT_CONTROL structure, rather than by including the values in the operand list.

SERIAL_PORT_CONTROL Structure

Mnemonic	Data Type	Description
.EN	BOOL	The enable bit indicates that the instruction is enabled.
.EU	BOOL	The queue bit indicates that the instruction entered the ASCII queue.
.DN	BOOL	The done bit indicates when the instruction is done, but it is asynchronous to the logic scan.
.RN	BOOL	The run bit indicates that the instruction is executing.
.EM	BOOL	The empty bit indicates that the instruction is done, but it is synchronous to the logic scan.
.ER	BOOL	The error bit indicates when the instruction fails (errors).
.FD	BOOL	The found bit does not apply to this instruction.
.LEN	DINT	The length indicates the number of characters to move to the destination (read).
.POS	DINT	The position displays the number of characters that were read.
.ERROR	DINT	The error contains a hexadecimal value that identifies the cause of an error.

Description: The ARD instruction removes the specified number of characters from the buffer and stores them in the Destination.

- The ARD instruction continues to execute until it removes the specified number of characters (Serial Port Control Length).
- While the ARD instruction is executing, no other ASCII Serial Port instruction executes.

Follow these guidelines to program the ARD instruction.

1. Configure the serial port of the controller for user mode.
2. Use the results of an ACB instruction to trigger the ARD instruction. This prevents the ARD instruction from holding up the ASCII queue while it waits for the required number of characters.
3. This is a transitional instruction.
    - In relay ladder, toggle the rung-condition-in from cleared to set each time the instruction should execute.
    - In structured text, condition the instruction so that it executes only on a transition.
4. To trigger a subsequent action when the instruction is done, examine the EM bit.

Arithmetic Status Flags: Not affected

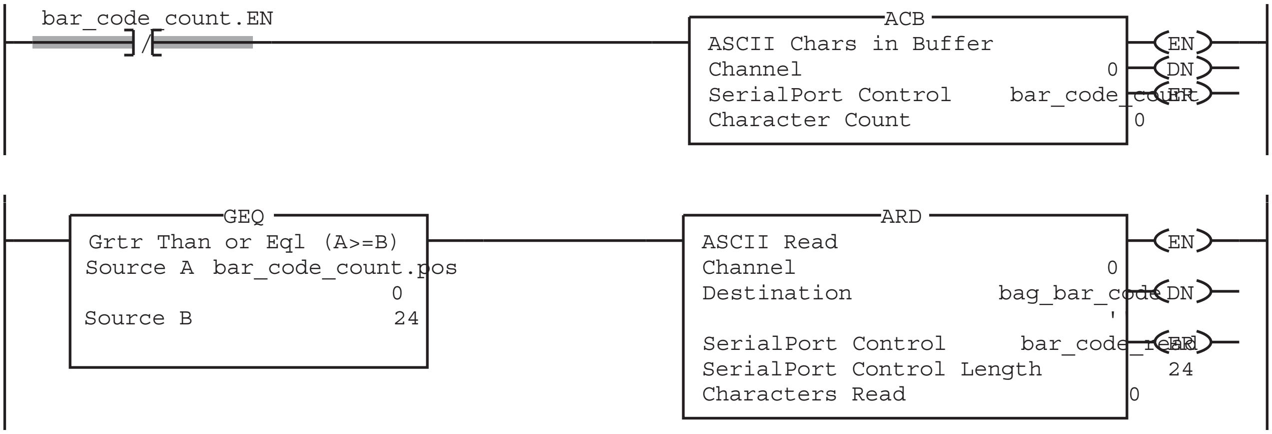
Fault Conditions: None

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes when rung-condition-in toggles from cleared to set. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction removes characters from the buffer and stores them in the destination. The .EN bit is set. The remaining status bits, except .UL, are cleared. The instruction attempts to enter the ASCII queue.	
Postscan	The rung-condition-out is set to false.	No action taken.

Example: A bar code reader sends bar codes to the serial port (channel 0) of the controller. Each bar code contains 24 characters. To determine when the controller receives a bar code, the ACB instruction continuously counts the characters in the buffer. When the buffer contains at least 24 characters, the controller has received a bar code. The ARD instruction moves the bar code to the DATA member of the *bag_bar_code* tag, which is a string.

Relay Ladder



Structured Text

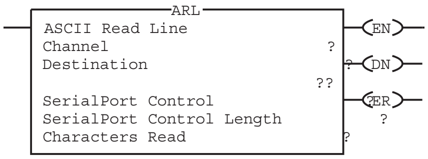
```
ACB(0,bar_code_count);

IF bar_code_count.POS >= 24 THEN

   bar_code_read.LEN := 24;

   ARD(0,bag_bar_code,bar_code_read);

END_IF;
```

ASCII Read Line (ARL)

The ARL instruction removes specified characters from the buffer and stores them in the Destination.

Operands:

Relay Ladder



Operand	Type	Format	Enter	Notes
Channel	DINT	Immediate Tag	0	
Destination	**String** SINT INT DINT	Tag	Tag into which the characters are moved (read): • For a string data type, enter the name of the tag. • For a SINT, INT, or DINT array, enter the first element of the array.	• If you want to compare, convert, or manipulate the characters, use a string data type. • String data types are: • default STRING data type. • any new string data type that you create.
Serial Port Control	SERIAL_PORT_ CONTROL	Tag	Tag that controls the operation	
Serial Port Control Length	DINT	Immediate	Maximum number of characters to read if no termination characters are found	• Enter the maximum number of characters that any message will contain (that is, when to stop reading if no termination characters are found). For example, if messages range from 3 to 6 characters in length, enter 6. • The Serial Port Control Length must be less than or equal to the size of the Destination. • If you want to set the Serial Port Control Length equal to the size of the Destination, enter 0.
Characters Read	DINT	immediate	0	During execution, displays the number of characters that were read.

Structured Text

```
ARL(Channel,Destination,
    SerialPortControl);
```

The operands are the same as those for the relay ladder ARL instruction. However, you specify the Serial Port Control Length and the Characters Read values by accessing the .LEN and .POS members of the SERIAL_PORT_CONTROL structure, rather than by including the values in the operand list.

SERIAL_PORT_CONTROL Structure

Mnemonic	Data Type	Description
.EN	BOOL	The enable bit indicates that the instruction is enabled.
.EU	BOOL	The queue bit indicates that the instruction entered the ASCII queue.
.DN	BOOL	The done bit indicates when the instruction is done, but it is asynchronous to the logic scan.
.RN	BOOL	The run bit indicates that the instruction is executing.
.EM	BOOL	The empty bit indicates that the instruction is done, but it is synchronous to the logic scan.
.ER	BOOL	The error bit indicates when the instruction fails (errors).
.FD	BOOL	The found bit does not apply to this instruction.
.LEN	DINT	The length indicates the maximum number of characters to move to the destination (that is, when to stop reading if no termination characters are found).
.POS	DINT	The position displays the number of characters that were read.
.ERROR	DINT	The error contains a hexadecimal value that identifies the cause of an error.

Description: The ARL instruction removes characters from the buffer and stores them in the Destination.

- The ARL instruction continues to execute until it removes either the:
  - first set of termination characters
  - specified number of characters (Serial Port Control Length)
- While the ARL instruction is executing, no other ASCII serial port instruction executes.

Follow these guidelines to program the ARL instruction.

1. Configure the serial port of the controller.
   a. Select User mode.
   b. Define the characters that serve as the termination characters.
2. Use the results of an ABL instruction to trigger the ARL instruction. This prevents the ARL instruction from holding up the ASCII queue while it waits for the termination characters.
3. This is a transitional instruction.
   - In relay ladder, toggle the rung-condition-in from cleared to set each time the instruction should execute.
   - In structured text, condition the instruction so that it executes only on a transition.
4. To trigger a subsequent action when the instruction is done, examine the EM bit.

Arithmetic Status Flags: Not affected

Fault Conditions: None

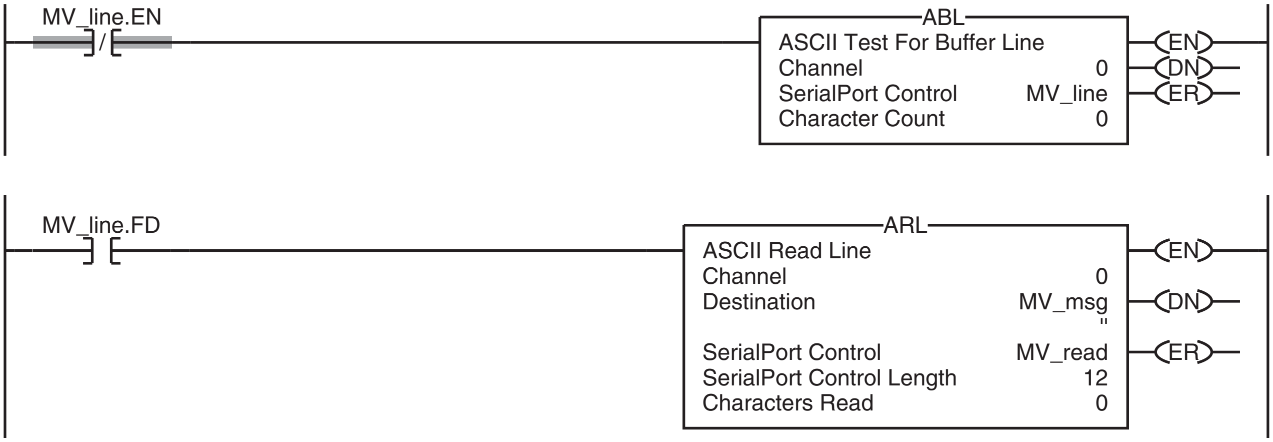
Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes when rung-condition-in toggles from cleared to set. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction removes the specified characters from the buffer and stores them in the destination. The .EN bit is set. The remaining status bits, except .UL, are cleared. The instruction attempts to enter the ASCII queue.	
Postscan	The rung-condition-out is set to false.	No action taken.

Example: Continuously test the buffer for a message from a MessageView terminal. Since each message ends in a carriage return ($r), the carriage return is configured as the termination character in the Controller Properties dialog box, User Protocol tab. When the ABL finds a carriage return, its sets the FD bit.

When the ABL instruction finds the carriage return (*MV_line.FD* is set), the controller has received a complete message. The ARL instruction removes the characters from the buffer, up to and including the carriage return, and places them in the DATA member of the *MV_msg* tag, which is a string.

Relay Ladder



Structured Text

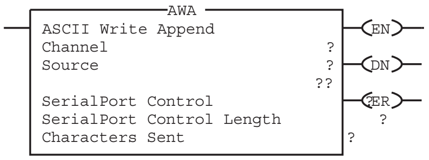
```
ABL(0,MV_line);


osri_1.InputBit := MVLine.FD;

OSRI(osri_1);


IF (osri_1.OutputBit) THEN

    mv_read.LEN := 12;

    ARL(0,MV_msg,MV_read);

END_IF;
```

ASCII Write Append (AWA)

The AWA instruction sends a specified number of characters of the Source tag to a serial device and appends either one or two predefined characters.

Operands:

Relay Ladder



Operand	Type	Format	Enter	Notes
Channel	DINT	Immediate Tag	0	
Source	**string** SINT INT DINT	Tag	Tag that contains the characters to send: • For a string data type, enter the name of the tag. • For a SINT, INT, or DINT array, enter the first element of the array.	• If you want to compare, convert, or manipulate the characters, use a string data type. • String data types are: • default STRING data type. • any new string data type that you create.
Serial Port Control	SERIAL_PORT_ CONTROL	Tag	Tag that controls the operation	
Serial Port Control Length	DINT	Immediate	Number of characters to send	• The Serial Port Control Length must be less than or equal to the size of the Source. • If you want to set the Serial Port Control Length equal to the number of characters in the Source, enter 0.
Characters Sent	DINT	Immediate	0	During execution, displays the number of characters that were sent.

Structured Text

```
AWA(Channel,Source,
     SerialPortControl);
```

The operands are the same as those for the relay ladder AWA instruction. However, you specify the Serial Port Control Length and the Characters Sent values by accessing the .LEN and .POS members of the SERIAL_PORT_CONTROL structure, rather than by including the values in the operand list.

SERIAL_PORT_CONTROL Structure

Mnemonic	Data Type	Description
.EN	BOOL	The enable bit indicates that the instruction is enabled.
.EU	BOOL	The queue bit indicates that the instruction entered the ASCII queue.
.DN	BOOL	The done bit indicates when the instruction is done, but it is asynchronous to the logic scan.
.RN	BOOL	The run bit indicates that the instruction is executing.
.EM	BOOL	The empty bit indicates that the instruction is done, but it is synchronous to the logic scan.
.ER	BOOL	The error bit indicates when the instruction fails (errors).
.FD	BOOL	The found bit does not apply to this instruction.
.LEN	DINT	The length indicates the number of characters to send.
.POS	DINT	The position displays the number of characters that were sent.
.ERROR	DINT	The error contains a hexadecimal value that identifies the cause of an error.

Description:

The AWA instruction:

- sends the specified number of characters (Serial Port Control Length) of the Source tag to the device that is connected to the serial port of the controller.
- adds to the end of the characters (appends) either one or two characters that are defined in the Controller Properties dialog box, User Protocol tab.

Follow these guidelines to program the AWA instruction.

1. Configure the serial port of the controller.

   a. Does your application also include ARD or ARL instructions?

If	Then
Yes	Select User mode
No	Select either System or User mode

   b. Define the characters to append to the data.

2. This is a transitional instruction.
   - In relay ladder, toggle the rung-condition-in from cleared to set each time the instruction should execute.
   - In structured text, condition the instruction so that it executes only on a transition.

3. Each time the instruction executes, do you always send the same number of characters?

If	Then
Yes	In the Serial Port Control Length, enter the number of characters to send.
No	Before the instruction executes, set the LEN member of the Source tag to the LEN member of the Serial Port Control tag.

Arithmetic Status Flags: Not affected

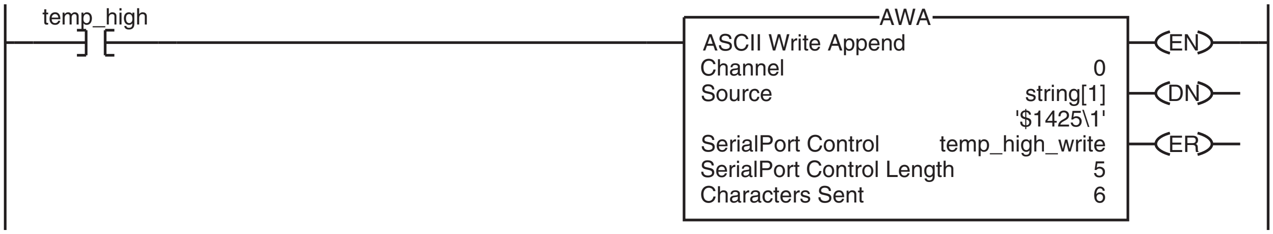
Fault Conditions: None

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes when rung-condition-in toggles from cleared to set. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction sends a specified number of characters and appends either one or two predefined characters. The .EN bit is set. The remaining status bits, except .UL, are cleared. The instruction attempts to enter the ASCII queue.	
Postscan	The rung-condition-out is set to false.	No action taken.

Example 1: When the temperature exceeds the high limit (*temp_high* is set), the AWA instruction sends a message to a MessageView terminal that is connected to the serial port of the controller. The message contains five characters from the DATA member of the *string[1]* tag, which is a string. (The *$14* counts as one character. It is the hex code for the Ctrl-T character.) The instruction also sends (appends) the characters defined in the controller properties. In this example, the AWA instruction sends a carriage return ($0D), which marks the end of the message.

Relay Ladder

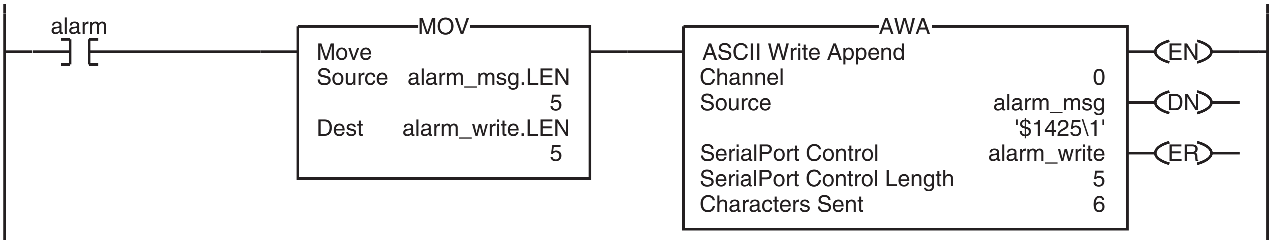


Structured Text

```
IF temp_high THEN

     temp_high_write.LEN := 5;

     AWA(0,string[1],temp_high_write);

     temp_high := 0;

END_IF;
```

Example 2: When *alarm* is set, the AWA instruction sends the specified number of characters in *alarm_msg* and appends a termination character (s). Because the number of characters in *alarm_msg* varies, the rung first moves the length of the string (*alarm_msg*.LEN) to the Serial Port Control Length of the AWA instruction (*alarm_write*.LEN). In *alarm_msg*, the *$14* counts as one character. It is the hex code for the Ctrl-T character.

Relay Ladder



Structured Text

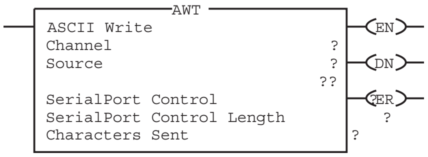
```
osri_1.InputBit := alarm;

OSRI(osri_1);


IF (osri_1.OutputBit) THEN

     alarm_write.LEN := alarm_msg.LEN;

     AWA(0,alarm_msg,alarm_write);

END_IF;
```

ASCII Write (AWT)

The AWT instruction sends a specified number of characters of the Source tag to a serial device.

Operands:

Relay Ladder



Operand	**Type**	**Format**	**Enter**	**Notes**
Channel	DINT	Immediate Tag	0	
Source	**String** SINT INT DINT	Tag	Tag that contains the characters to send: • For a string data type, enter the name of the tag. • For a SINT, INT, or DINT array, enter the first element of the array.	• If you want to compare, convert, or manipulate the characters, use a string data type. • String data types are: • default STRING data type. • any new string data type that you create.
Serial Port Control	SERIAL_PORT_CONTROL	Tag	Tag that controls the operation	
Serial Port Control Length	DINT	Immediate	Number of characters to send	• The Serial Port Control Length must be less than or equal to the size of the Source. • If you want to set the Serial Port Control Length equal to the number of characters in the Source, enter 0.
Characters Sent	DINT	Immediate	0	During execution, displays the number of characters that were sent.

Structured Text

```
AWT(Channel,Source,
    SerialPortControl);
```

The operands are the same as those for the relay ladder AWT instruction. However, you specify the Serial Port Control Length and the Characters Sent values by accessing the .LEN and .POS members of the SERIAL_PORT_CONTROL structure, rather than by including the values in the operand list.

SERIAL_PORT_CONTROL Structure

Mnemonic	Data Type	Description
.EN	BOOL	The enable bit indicates that the instruction is enabled.
.EU	BOOL	The queue bit indicates that the instruction entered the ASCII queue.
.DN	BOOL	The done bit indicates when the instruction is done, but it is asynchronous to the logic scan.
.RN	BOOL	The run bit indicates that the instruction is executing.
.EM	BOOL	The empty bit indicates that the instruction is done, but it is synchronous to the logic scan.
.ER	BOOL	The error bit indicates when the instruction fails (errors).
.FD	BOOL	The found bit does not apply to this instruction.
.LEN	DINT	The length indicates the number of characters to send.
.POS	DINT	The position displays the number of characters that were sent.
.ERROR	DINT	The error contains a hexadecimal value that identifies the cause of an error.

Description: The AWT instruction sends the specified number of characters (Serial Port Control Length) of the Source tag to the device that is connected to the serial port of the controller.

Follow these guidelines to program the AWT instruction.

1. Configure the serial port of the controller.

If your application	Then
Uses ARD or ARL instructions	Select User mode
Does not use ARD or ARL instructions	Select either System or User mode

2. This is a transitional instruction.
   - In relay ladder, toggle the rung-condition-in from cleared to set each time the instruction should execute.
   - In structured text, condition the instruction so that it executes only on a transition.

3. Each time the instruction executes, do you always send the same number of characters?

If	Then
Yes	In the Serial Port Control Length, enter the number of characters to send.
No	Before the instruction executes, move the LEN member of the Source tag to the LEN member of the Serial Port Control tag.

Arithmetic Status Flags: Not affected

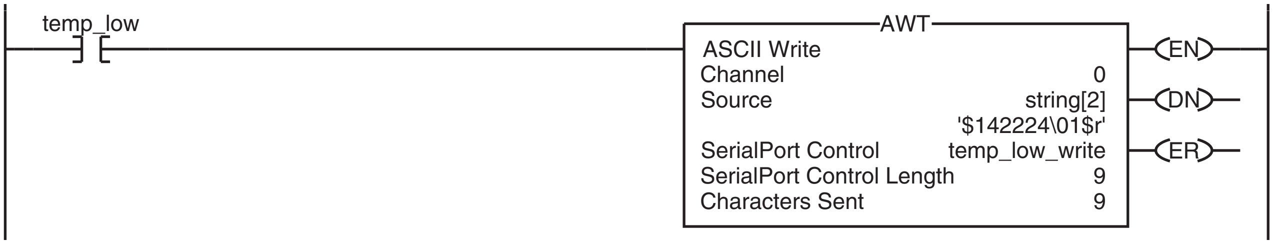
Fault Conditions: None

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes when rung-condition-in toggles from cleared to set. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction sends a specified number of characters. The .EN bit is set. The remaining status bits, except .UL, are cleared. The instruction attempts to enter the ASCII queue.	
Postscan	The rung-condition-out is set to false.	No action taken.

Example 1: When the temperature reaches the low limit (*temp_low* is set), the AWT instruction sends a message to the MessageView terminal that is connected to the serial port of the controller. The message contains nine characters from the DATA member of the *string[2]* tag, which is a string. (The *$14* counts as one character. It is the hex code for the Ctrl-T character.) The last character is a carriage return ($r), which marks the end of the message.

Relay Ladder

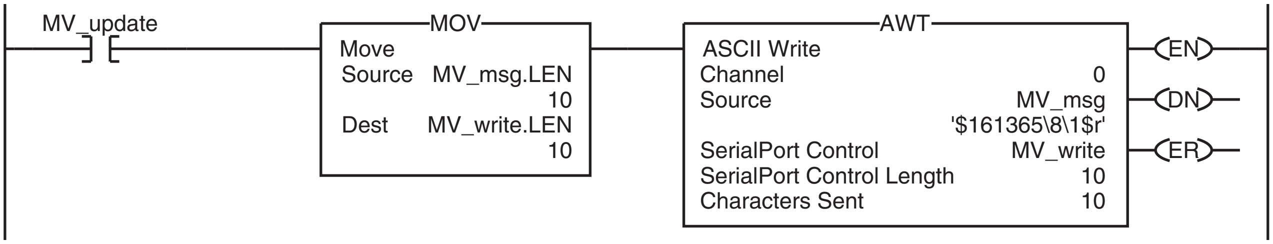


Structured Text

```
osri_1.InputBit := temp_low;

OSRI(osri_1);

IF (osri_1.OutputBit) THEN
     temp_low_write.LEN := 9;
     AWT(0,string[2],temp_low_write);
END_IF;
```

Example 2: When *MV_update* is set, the AWT instruction sends the characters in *MV_msg*. Because the number of characters in *MV_msg* varies, the rung first moves the length of the string (*MV_msg*.LEN) to the Serial Port Control Length of the AWT instruction (*MV_write*.LEN). In *MV_msg*, the *$16* counts as one character. It is the hex code for the Ctrl-V character.

Relay Ladder



Structured Text

```
osri_1.InputBit := MV_update;

OSRI(osri_1);


IF (osri_1.OutputBit) THEN

     MV_write.LEN := Mv_msg.LEN;

     AWT(0,MV_msg,MV_write);

END_IF;
```

Notes:

ASCII String Instructions
(CONCAT, DELETE, FIND, INSERT, MID)

Introduction

Use the ASCII string instructions to modify and create strings of ASCII characters.

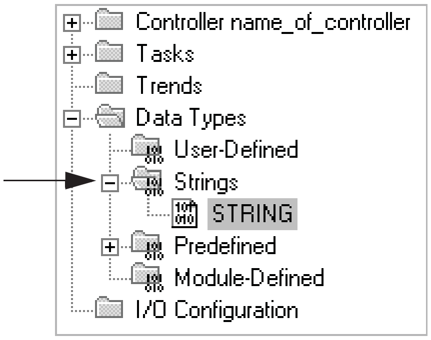
If you want to	For example	Use this instruction	Available in these languages	Page
Add characters to the end of a string	Add termination characters or delimiters to a string	CONCAT	Relay ladder Structured text	609
Delete characters from a string	Remove header or control characters from a string	DELETE	Relay ladder Structured text	611
Determine the starting character of a sub-string	Locate a group of characters within a string	FIND	Relay ladder Structured text	613
Insert characters into a string	Create a string that uses variables	INSERT	Relay ladder Structured text	615
Extract characters from a string	Extract information from a bar code	MID	Relay ladder Structured text	617

You can also use the following instructions to compare or convert ASCII characters.

If you want to	Use this instruction	Page
Compare a string to another string	CMP	214
See if the characters are equal to specific characters	EQU	219
See if the characters are not equal to specific characters	NEQ	250
See if the characters are equal to or greater than specific characters	GEQ	219
See if the characters are greater than specific characters	GRT	227
See if the characters are equal to or less than specific characters	LEQ	231
See if the characters are less than specific characters	LES	235
Rearrange the bytes of a INT, DINT, or REAL tag	SWPB	307
Find a string in an array of strings	FSC	354
Convert characters to a SINT, INT, DINT, or REAL value	STOD	621

If you want to	Use this instruction	Page
Convert characters to a REAL value	STOR	624
Convert a SINT, INT, DINT, or REAL value to a string of ASCII characters	DTOS	627
Convert REAL value to a string of ASCII characters	RTOS	629

String Data Types



You store ASCII characters in tags that use a string data type.

- You can use the default STRING data type. It stores up to 82 characters.
- You can create a new string data type that stores less or more characters.

To create a new string data type, see the Logix5000 Controllers Common Procedures Programming Manual, publication 1756-PM001.

Each string data type contains the following members.

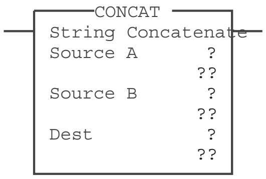
Name	Data Type	Description	Notes
LEN	DINT	Number of characters in the string	The LEN automatically updates to the new count of characters whenever you: • use the String Browser dialog box to enter characters. • use instructions that read, convert, or manipulate a string. The LEN shows the length of the current string. The DATA member may contain additional, old characters, which are not included in the LEN count.
DATA	SINT array	ASCII characters of the string	• To access the characters of the string, address the name of the tag. For example, to access the characters of the *string_1* tag, enter *string_1*. • Each element of the DATA array contains one character. • You can create new string data types that store less or more characters.

String Concatenate (CONCAT)

The CONCAT instruction adds ASCII characters to the end of a string.

Operands:

Relay Ladder



```
CONCAT
String Concatenate
Source A      ?
             ??
Source B      ?
             ??
Dest          ?
             ??
```

Operand	Type	Format	Enter	Notes
Source A	String	Tag	Tag that contains the initial characters	String data types are: • default STRING data type. • any new string data type that you create.
Source B	String	Tag	Tag that contains the end characters	
Destination	String	Tag	Tag to store the result	

Structured Text

```
CONCAT(SourceA,SourceB,
     Dest);
```

The operands are the same as those for the relay ladder CONCAT instruction.

Description:

The CONCAT instruction combines the characters in Source A with the characters in Source B and places the result in the Destination.

- The characters from Source A are first, followed by the characters from Source B.
- Unless Source A and the Destination are the same tag, Source A remains unchanged.

Arithmetic Status Flags:

Not affected

Fault Conditions:

Type	Code	Cause	Recovery Method
4	51	The LEN value of the string tag is greater than the DATA size of the string tag.	1. Check that no instruction is writing to the LEN member of the string tag. 2. In the LEN value, enter the number of characters that the string contains.

Execution:

Condition	**Relay Ladder Action**	**Structured Text Action**
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction concatenates the strings.	
Postscan	The rung-condition-out is set to false.	No action taken.

Example: To trigger a message in a MessageView terminal, the controller must send an ASCII string that contains a message number and node number. *String_1* contains the message number. When *add_node* is set, the CONCAT instruction adds the characters in *node_num_ascii* (node number) to the end of the characters in *string_1* and then stores the result in *msg*.

Relay Ladder

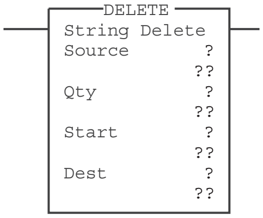
Structured Text

```
IF add_node THEN

     CONCAT(string_1,node_num_ascii,msg);

     add_node := 0;

END_IF;
```

String Delete (DELETE)

The DELETE instruction does not automatically remove all ASCII characters from a string. An algorithm determines which characters in the string are removed depending on the starting position, quantity, and size of the Source.

Operands:



Relay Ladder

The DELETE instruction:

- copies the string from the Source to the Destination, ignoring deleted characters and updating the Destination string with the number of characters copied.
- updates the length of the Destination string by the position of characters in the Source string and the number of characters being deleted.
- leaves the Source unchanged unless the Source and the Destination are the same tag.

Operand	**Type**	**Format**	**Enter**	**Notes**
Source	string	Tag	Tag that contains the string from which you want to delete characters	String data types are: • default STRING data type. • any new string data type that you create.
Quantity	SINT INT **DINT**	Immediate Tag	Number of characters to delete	The Start plus the Quantity must be less than or equal to the DATA size of the Source.
Start	SINT INT **DINT**	Immediate Tag	Position of the first character to delete	Enter a number between 1 and the DATA size of the Source.
Destination	string	Tag	Tag to store the result	

Structured Text

```
DELETE(Source,Qty,Start,
     Dest);
```

The operands are the same as those for the relay ladder DELETE instruction.

Arithmetic Status Flags: Not affected

Fault Conditions:

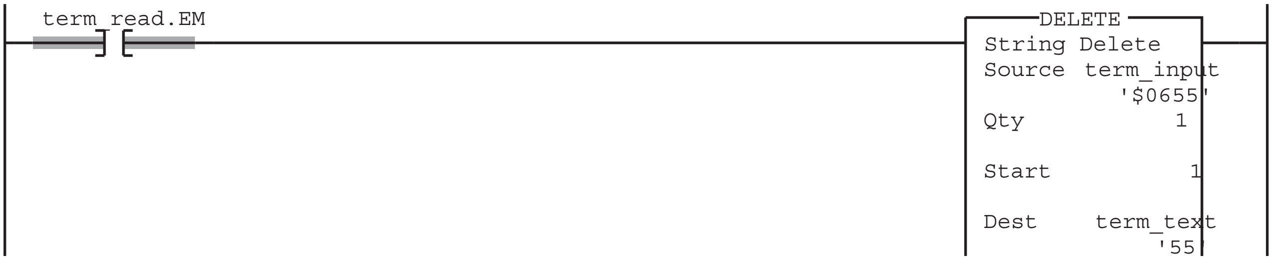
Type	Code	Cause	Recovery Method
4	51	The LEN value of the string tag is greater than the DATA size of the string tag.	1. Check that no instruction is writing to the LEN member of the string tag. 2. In the LEN value, enter the number of characters that the string contains.
4	56	The Start or Quantity value is invalid.	1. Check that the Start value is between 1 and the DATA size of the Source. 2. Check that the Start value plus the Quantity value is less than or equal to the DATA size of the Source.

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction deletes the specified characters.	
Postscan	The rung-condition-out is set to false.	No action taken.

Example: ASCII information from a terminal contains a header character. After the controller reads the data (*term_read.EM* is set) the DELETE instruction removes the header character.

Relay Ladder



Structured Text

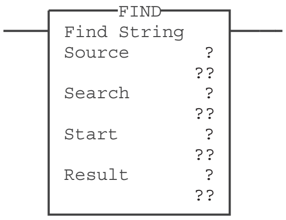
```
IF term_read.EM THEN
     DELETE(term_input,1,1,term_text);
     term_read.EM := 0;
END_IF;
```

Find String (FIND)

The FIND instruction locates the starting position of a specified string within another string.

Operands:

Relay Ladder



Operand	Type	Format	Enter	Notes
Source	String	Tag	String to search in	String data types are: • default STRING data type. • any new string data type that you create.
Search	String	Tag	String to find	
Start	SINT INT **DINT**	Immediate Tag	Position in Source to start the search	Enter a number between 1 and the DATA size of the Source.
Result	SINT INT **DINT**	Tag	Tag that stores the starting position of the string to find	

Structured Text

```
FIND(Source,Search,Start,
     Result);
```

The operands are the same as those for the relay ladder FIND instruction described above.

Description: The FIND instruction searches the Source string for the Search string. If the instruction finds the Search string, the Result shows the starting position of the Search string within the Source string.

Arithmetic Status Flags: Not affected

Fault Conditions:

Type	Code	Cause	Recovery Method
4	51	The LEN value of the string tag is greater than the DATA size of the string tag.	1. Check that no instruction is writing to the LEN member of the string tag. 2. In the LEN value, enter the number of characters that the string contains.
4	56	The Start value is invalid.	Check that the Start value is between 1 and the DATA size of the Source.

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction searches for the specified characters.	
Postscan	The rung-condition-out is set to false.	No action taken.

Example: A message from a MessageView terminal contains several pieces of information. The backslash character [\ ] separates each piece of information. To locate a piece of information, the FIND instruction searches for the backslash character and records its position in *find_pos*.

Relay Ladder

Structured Text

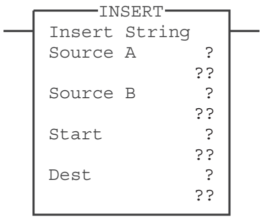
```
IF MV_read.EM THEN

     FIND(MV_msg,find,1,find_pos);

     MV_read.EM := 0;

END_IF;
```

Insert String (INSERT)

The INSERT instruction adds ASCII characters to a specified location within a string.

Operands:

Relay Ladder



Operand	Type	Format	Enter	Notes
Source A	String	Tag	String to add the characters to	String data types are: • default STRING data type. • any new string data type that you create.
Source B	String	Tag	String containing the characters to add	
Start	SINT INT **DINT**	Immediate Tag	Position in Source A to add the characters	Enter a number between 1 and the DATA size of the Source.
Result	String	Tag	String to store the result	

Structured Text

```
INSERT(SourceA,SourceB,
    Start,Dest);
```

The operands are the same as those for the relay ladder INSERT instruction.

Description:

The INSERT instruction adds the characters in Source B to a designated position within Source A and places the result in the Destination.

- Start defines where in Source A that Source B is added.
- Unless Source A and the Destination are the same tag, Source A remains unchanged.

Arithmetic Status Flags:

Not affected

Fault Conditions:

Type	Code	Cause	Recovery Method
4	51	The LEN value of the string tag is greater than the DATA size of the string tag.	1. Check that no instruction is writing to the LEN member of the string tag. 2. In the LEN value, enter the number of characters that the string contains.
4	56	The Start value is invalid.	Check that the Start value is between 1 and the DATA size of the Source.

Execution:

Condition	**Relay Ladder Action**	**Structured Text Action**
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction inserts the specified characters.	
Postscan	The rung-condition-out is set to false.	No action taken.

Example: When *temp_high* is set, the INSERT instruction adds the characters in *string_2* to position 2 within *string_1* and places the result in *string_3*:

Relay Ladder

Structured Text

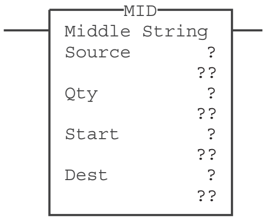
```
IF temp_high THEN

     INSERT(string_1,string_2,2,string_3);

     temp_high := 0;

END_IF;
```

Middle String (MID)

The MID instruction copies a specified number of ASCII characters from a string and stores them in another string.

Operands:

Relay Ladder



Operand	Type	Format	Enter	Notes
Source	String	Tag	String to copy characters from	String data types are: • default STRING data type. • any new string data type that you create.
Quantity	SINT INT **DINT**	Immediate Tag	Number of characters to copy	The Start plus the Quantity must be less than or equal to the DATA size of the Source.
Start	SINT INT **DINT**	Immediate Tag	Position of the first character to copy	Enter a number between 1 and the DATA size of the Source.
Destination	String	Tag	String to copy the characters to	

Structured Text

```
MID(Source,Qty,Start,
    Dest);
```

The operands are the same as those for the relay ladder MID instruction.

Description: The MID instruction copies a group of characters from the Source and places the result in the Destination.

- The Start position and Quantity define the characters to copy.
- Unless the Source and Destination are the same tag, the Source remains unchanged.

Arithmetic Status Flags: Not affected

Fault Conditions:

Type	Code	Cause	Recovery Method
4	51	The LEN value of the string tag is greater than the DATA size of the string tag.	1. Check that no instruction is writing to the LEN member of the string tag. 2. In the LEN value, enter the number of characters that the string contains.
4	56	The Start or Quantity value is invalid.	1. Check that the Start value is between 1 and the DATA size of the Source. 2. Check that the Start value plus the Quantity value is less than or equal to the DATA size of the Source.

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction copies the specified characters from a string and stores them in another string.	
Postscan	The rung-condition-out is set to false.	No action taken.

Example: In a baggage handling conveyor of an airport, each bag gets a bar code. Characters 9…17 of the bar code are the flight number and destination airport of the bag. After the bar code is read (*bag_read.EM* is set) the MID instruction copies the flight number and destination airport to the *bag_flt_and_dest* string.

Relay Ladder

```
Middle String
Source      bag_barcode
  'NWA HOP 5058 AMS 01'
Qty                   9

Start

Dest     bag_flt_and_dest
            '5058 AMS '
```

Structured Text

```
IF bag_read.EM THEN

     MID(bar_barcode,9,9,bag_flt_and_dest);

     bag_read.EM := 0;

END_IF;
```

Chapter 19

ASCII Conversion Instructions

(STOD, STOR, DTOS, RTOS, UPPER, LOWER)

Introduction

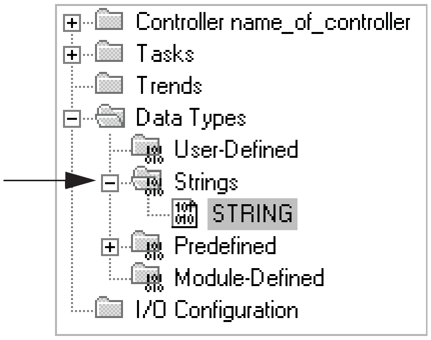
Use the ASCII conversion instructions to alter the format of data.

If you want to	For example	Use this instruction	Available in these languages	Page
Convert the ASCII representation of an integer value to a SINT, INT, DINT, or REAL value	Convert a value from a weight scale or other ASCII device to an integer so you can use it in your logic	STOD	Relay ladder Structured text	621
Convert the ASCII representation of a floating-point value to a REAL value	Convert a value from a weight scale or other ASCII device to a REAL value so you can use it in your logic	STOR	Relay ladder Structured text	624
Convert a SINT, INT, DINT, or REAL value to a string of ASCII characters	Convert a variable to an ASCII string so you can send it to a MessageView terminal	DTOS	Relay ladder Structured text	627
Convert a REAL value to a string of ASCII characters	Convert a variable to an ASCII string so you can send it to a MessageView terminal	RTOS	Relay ladder Structured text	629
Convert the letters in a string of ASCII characters to upper case	Convert an entry made by an operator to all upper case so you can search for it in an array	UPPER	Relay ladder Structured text	631
Convert the letters in a string of ASCII characters to lower case	Convert an entry made by an operator to all lower case so you can search for it in an array	LOWER	Relay ladder Structured text	633

You can also use the following instructions to compare or manipulate ASCII characters.

If you want to	Use this instruction	Page
Add characters to the end of a string	CONCAT	609
Delete characters from a string	DELETE	611
Determine the starting character of a sub-string	FIND	613
Insert characters into a string	INSERT	615
Extract characters from a string	MID	617
Rearrange the bytes of a INT, DINT, or REAL tag	SWPB	307
Compare a string to another string	CMP	214
See if the characters are equal to specific characters	EQU	219
See if the characters are not equal to specific characters	NEQ	250
See if the characters are equal to or greater than specific characters	GEQ	223
See if the characters are greater than specific characters	GRT	227
See if the characters are equal to or less than specific characters	LEQ	231
See if the characters are less than specific characters	LES	235
Find a string in an array of strings	FSC	354

String Data Types



You store ASCII characters in tags that use a string data type.

- You can use the default STRING data type. It stores up to 82 characters.
- You can create a new string data type that stores less or more characters.

To create a new string data type, see the Logix5000 Controllers Common Procedures Programming Manual, publication 1756-PM001.

Each string data type contains the following members.

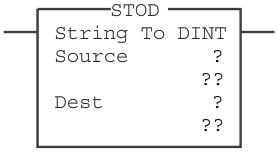
Name	Data Type	Description	Notes
LEN	DINT	Number of characters in the string	The LEN automatically updates to the new count of characters whenever you: • use the String Browser dialog box to enter characters. • use instructions that read, convert, or manipulate a string. The LEN shows the length of the current string. The DATA member may contain additional, old characters, which are not included in the LEN count.
DATA	SINT array	ASCII characters of the string	• To access the characters of the string, address the name of the tag. For example, to access the characters of the *string_1* tag, enter *string_1*. • Each element of the DATA array contains one character. • You can create new string data types that store less or more characters.

String To DINT (STOD)

The STOD instruction converts the ASCII representation of an integer to an integer or REAL value.

Operands:

Relay Ladder



Operand	Type	Format	Enter	Notes
Source	String	Tag	Tag that contains the value in ASCII	String data types are: • default STRING data type. • any new string data type that you create.
Destination	SINT INT **DINT** REAL	Tag	Tag to store the integer value	If the Source value is a floating-point number, the instruction converts only the non-fractional part of the number (regardless of the destination data type).

```
STOD(Source,Dest);
```

Structured Text

The operands are the same as those for the relay ladder STOD instruction.

Description: The STOD converts the Source to an integer and places the result in the Destination.

- The instruction converts positive and negative numbers.
- If the Source string contains non-numeric characters, the STOD converts the first set of contiguous numbers:
  - The instruction skips any initial control or non-numeric characters (except the minus sign in front of a number).
  - If the string contains multiple groups of numbers that are separated by delimiters (for example, /), the instruction converts only the first group of numbers.

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions

Type	Code	Cause	Recovery Method
4	51	The LEN value of the string tag is greater than the DATA size of the string tag.	1. Check that no instruction is writing to the LEN member of the string tag. 2. In the LEN value, enter the number of characters that the string contains.
4	53	The output number is beyond the limits of the destination data type.	Either: • Reduce the size of the ASCII value. • Use a larger data type for the destination.

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	SC is set. Destination is cleared. The instruction converts the Source. If the result is zero, then S:Z is set	
Postscan	The rung-condition-out is set to false.	No action taken.

Example: When *MV_read.EM* is set, the STOD instruction converts the first set of numeric characters in *MV_msg* to an integer value. The instruction skips the initial control character ($06) and stops at the delimiter (\ ).

Relay Ladder

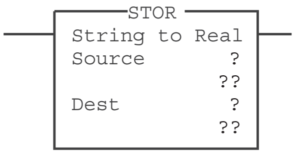
Structured Text

```
IF MV_read.EM THEN
     STOD(MV_msg,MV_msg_nmbr);
     MV_read.EM := 0;
END_IF;
```

String To REAL (STOR)

The STOR instruction converts the ASCII representation of a floating-point value to a REAL value.

Operands:



Relay Ladder Operands

Operand	Type	Format	Enter	Notes
Source	String	Tag	Tag that contains the value in ASCII	String data types are: • default STRING data type. • any new string data type that you create.
Destination	REAL	Tag	Tag to store the REAL value	

```
STOR(Source,Dest);
```

Structured Text

The operands are the same as those for the relay ladder STOR instruction.

Description: The STOR converts the Source to a REAL value and places the result in the Destination.

- The instruction converts positive and negative numbers.
- If the Source string contains non-numeric characters, the STOR converts the first set of contiguous numbers, including the decimal point [.]:
  - The instruction skips any initial control or non-numeric characters (except the minus sign in front of a number).
  - If the string contains multiple groups of numbers that are separated by delimiters (for example, /), the instruction converts only the first group of numbers.

Arithmetic Status Flags: Arithmetic status flags are affected.

Fault Conditions:

Type	Code	Cause	Recovery Method
4	51	The LEN value of the string tag is greater than the DATA size of the string tag.	1. Check that no instruction is writing to the LEN member of the string tag. 2. In the LEN value, enter the number of characters that the string contains.
4	53	The output number is beyond the limits of the destination data type.	Either: • Reduce the size of the ASCII value. • Use a larger data type for the destination.

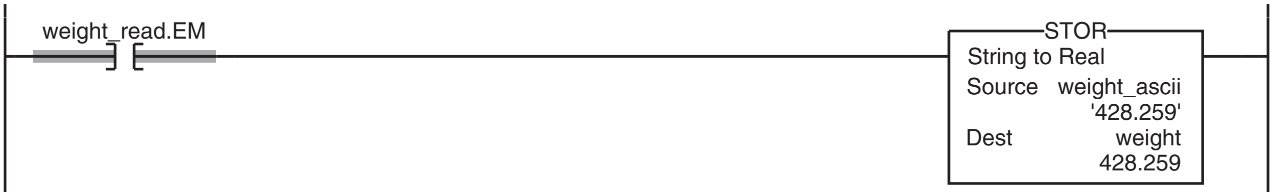
Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	S:C is set. Destination is cleared. The instruction converts the Source. If the result is zero, then S:Z is set	
Postscan	The rung-condition-out is set to false.	No action taken.

Example: After reading the weight from a scale (*weight_read.EM* is set) the STOR instruction converts the numeric characters in *weight_ascii* to a REAL value.

You may see a slight difference between the fractional parts of the Source and Destination.

Relay Ladder



Structured Text

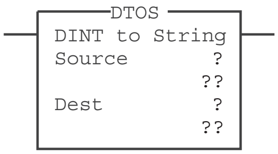
```
IF weight_read.EM THEN
     STOR(weight_ascii,weight);
     weight_read.EM := 0;
END_IF;
```

DINT to String (DTOS)

The DTOS instruction produces the ASCII representation of a value.

Operands:

Relay Ladder



Operand	Type	Format	Enter	Notes
Source	SINT INT **DINT** REAL	Tag	Tag that contains the value	If the Source is a REAL, the instruction converts it to a DINT value. Refer to REAL to an Integer on page 648.
Destination	String	Tag	Tag to store the ASCII value	String data types are: • default STRING data type. • any new string data type that you create.

Structured Text

```
DTOS(Source,Dest);
```

The operands are the same as those for the relay ladder DTOS instruction.

Description:

The DTOS converts the Source to a string of ASCII characters and places the result in the Destination.

Arithmetic Status Flags:

Not affected

Fault Conditions:

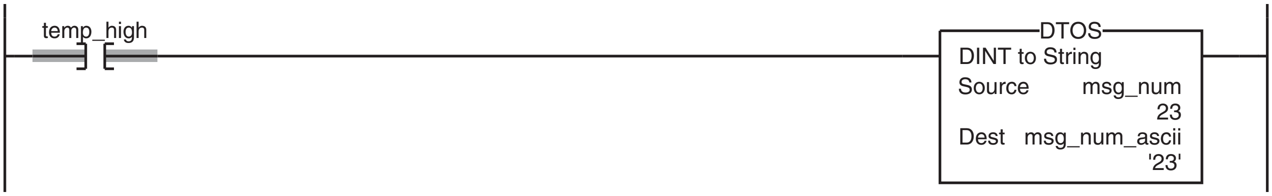
Type	Code	Cause	Recovery Method
4	51	The LEN value of the string tag is greater than the DATA size of the string tag.	1. Check that no instruction is writing to the LEN member of the string tag. 2. In the LEN value, enter the number of characters that the string contains.
4	52	The output string is larger than the destination.	Create a new string data type that is large enough for the output string. Use the new string data type as the data type for the destination.

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction converts the source.	
Postscan	The rung-condition-out is set to false.	No action taken.

Example: When *temp_high* is set, the DTOS instruction converts the value in *msg_num* to a string of ASCII characters and places the result in *msg_num_ascii*. Subsequent rungs insert or concatenate *msg_num_ascii* with other strings to produce a complete message for a display terminal.

Relay Ladder



Structured Text

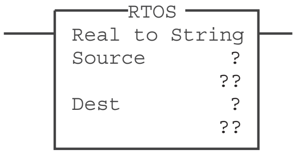
```
IF temp_high THEN

     DTOS(msg_num,msg_num_ascii);

     temp_high := 0;

END_IF;
```

REAL to String (RTOS)

The RTOS instruction produces the ASCII representation of a REAL value.

Operands:

Relay Ladder



Operand	Type	Format	Enter	Notes
Source	REAL	Tag	Tag that contains the REAL value	
Destination	String	Tag	Tag to store the ASCII value	String data types are: • default STRING data type. • any new string data type that you create.

Structured Text

`RTOS(Source,Dest);`

The operands are the same as those for the relay ladder RTOS instruction.

Description: The RTOS converts the Source to a string of ASCII characters and places the result in the Destination.

Arithmetic Status Flags: Not affected

Fault Conditions:

Type	Code	Cause	Recovery Method
4	51	The LEN value of the string tag is greater than the DATA size of the string tag.	1. Check that no instruction is writing to the LEN member of the string tag. 2. In the LEN value, enter the number of characters that the string contains.
4	52	The output string is larger than the destination.	Create a new string data type that is large enough for the output string. Use the new string data type as the data type for the destination.

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction converts the source.	
Postscan	The rung-condition-out is set to false.	No action taken.

Example: When *send_data* is set, the RTOS instruction converts the value in *data_1* to a string of ASCII characters and places the result in *data_1_ascii*. Subsequent rungs insert or concatenate *data_1_ascii* with other strings to produce a complete message for a display terminal.

You may see a slight difference between the fractional parts of the Source and Destination.

Relay Ladder

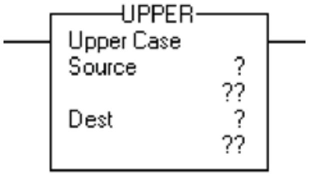
Structured Text

```
IF send_data THEN

     RTOS(data_1,data_1_ascii);

     send_data := 0;

END_IF;
```

Upper Case (UPPER)

The UPPER instruction converts the alphabetical characters in a string to upper case characters.

Operands:



Relay Ladder

Operand	Type	Format	Description
Source	String	Tag	Tag that contains the characters that you want to convert to upper case
Destination	String	Tag	Tag to store the characters in upper case

```
UPPER(Source,Dest);
```

Structured Text

The operands are the same as those for the relay ladder UPPER instruction.

Description:

The UPPER instruction converts to upper case all the letters in the Source and places the result in the Destination.

- ASCII characters are case sensitive. Upper case 'A' ($41) is **not** equal to lower case 'a' ($61).
- If operators directly enter ASCII characters, convert the characters to all upper case or all lower case before you compare them.

Any characters in the Source string that are not letters remain unchanged.

Arithmetic Status Flags:

Not affected

Fault Conditions:

None

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction converts the Source to upper case.	
Postscan	The rung-condition-out is set to false.	No action taken.

Example: To find information about a specific item, an operator enters the catalog number of the item into an ASCII terminal. After the controller reads the input from a terminal (*terminal_read.EM* is set), the UPPER instruction converts the characters in *catalog_number* to all upper case characters and stores the result in *catalog_number_upper_case*. A subsequent rung then searches an array for characters that match those in *catalog_number_upper_case*.

Relay Ladder

Structured Text

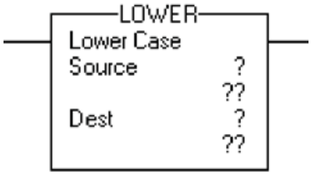
```
IF terminal_read.EM THEN

     UPPER(catalog_number,catalog_number_upper_case);

     terminal_read.EM := 0;

END_IF;
```

Lower Case (LOWER)

The LOWER instruction converts the alphabetical characters in a string to lower case characters.

Operands:

Relay Ladder



Operand	Type	Format	Description
Source	String	Tag	Tag that contains the characters that you want to convert to lower case
Destination	String	Tag	Tag to store the characters in lower case

Structured Text

```
LOWER(Source,Dest);
```

The operands are the same as those for the relay ladder LOWER instruction.

Description:

The LOWER instruction converts to lower case all the letters in the Source and places the result in the Destination.

- ASCII characters are case sensitive. Upper case ‘A’ ($41) is **not** equal to lower case ‘a’ ($61).
- If operators directly enter ASCII characters, convert the characters to all upper case or all lower case before you compare them.

Any characters in the Source string that are not letters remain unchanged.

Arithmetic Status Flags:

Not affected

Fault Conditions:

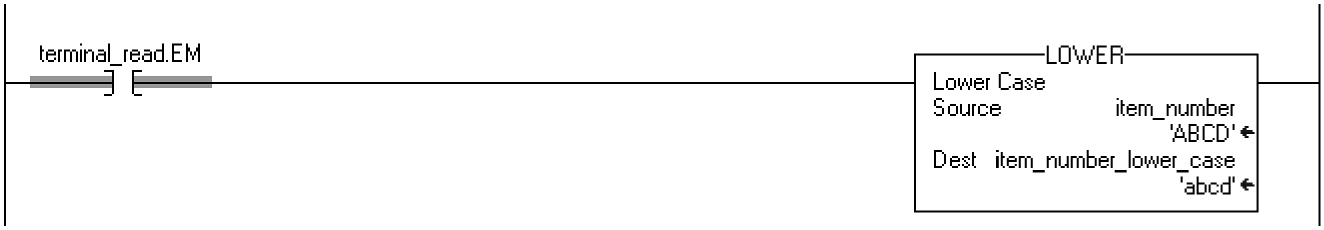
None

Execution:

Condition	Relay Ladder Action	Structured Text Action
Prescan	The rung-condition-out is set to false.	No action taken.
Rung-condition-in is false	The rung-condition-out is set to false.	N/A
Rung-condition-in is true	The instruction executes. The rung-condition-out is set to true.	N/A
EnableIn is set	N/A	EnableIn is always set. The instruction executes.
Instruction execution	The instruction converts the Source to lower case.	
Postscan	The rung-condition-out is set to false.	No action taken.

Example: To find information about a specific item, an operator enters the item number into an ASCII terminal. After the controller reads the input from a terminal (*terminal_read.EM* is set), the LOWER instruction converts the characters in *item_number* to all lower case characters and stores the result in *item_number_lower_case*. A subsequent rung then searches an array for characters that match those in *item_number_lower_case*.

Relay Ladder



Structured Text

```
IF terminal_read.EM THEN

     LOWER(item_number,item_number_lower_case);

     terminal_read.EM := 0;

END_IF;
```

Chapter 20

Debug Instructions

(BPT, TPT)

Introduction

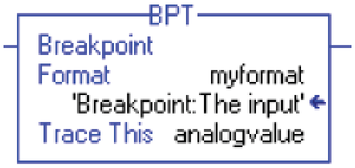
Use the debug instructions to monitor the state of your logic when it is in conditions that you determine. These instructions are compatible only with RSLogix Emulate 5000 software, with which you can emulate a Logix5000 controller on your personal computer.

If you want to	Use this instruction	Available in these languages	Page
Stop program emulation when a rung is true	BPT	Relay ladder	635
Log data you select when a rung is true	TPT	Relay ladder	639

Breakpoints (BPT)

Breakpoints stop program emulation when a rung is true.

Operands:



Relay Ladder

Operand	Type	Format	Description
Format	String	Tag	A string that sets the formatting for the text that appears in the trace window for the breakpoint.
Trace This	BOOL, SINT, INT, DINT, REAL	Tag	The tag that has a value you want to display in the trace window.

Description:

Breakpoints are programmed with the Breakpoint output instruction (BPT). When the inputs on a rung containing a BPT instruction are true, the BPT instruction stops program execution. The software displays a window indicating that the breakpoint triggered and the values that triggered it.

Stopped At Breakpoint. OK to Continue?

When a breakpoint triggers, the emulator displays a window informing you that a breakpoint occurred. The title bar of the window shows the slot containing the emulator that encountered the breakpoint.

When you click OK, the emulator resumes program execution. If the conditions that triggered the breakpoint persist, the breakpoint will recur.

In addition, the emulator opens a trace window for the breakpoint. The trace window displays information about the breakpoint and the values.

IMPORTANT

When a breakpoint triggers, you will not be able to edit your project until you permit the execution to continue. You can go online with the emulator to observe the state of your project, but you will not be able to edit it. If you try to accept a rung edit while a breakpoint is triggered, you will see a dialog box saying the controller is not in the correct mode.

String Format

With the Format string in the tracepoint and breakpoint instructions, you can control how the traced tags appear in the traces or breakpoint windows. The format of the string is:

heading:(text)%(type)

where *heading* is a text string identifying the tracepoint or breakpoint, *text* is a string describing the tag (or any other text you choose), and %(*type*) indicates the format of the tag. You need one type indicator for each tag you are tracing with the tracepoint or breakpoint instruction.

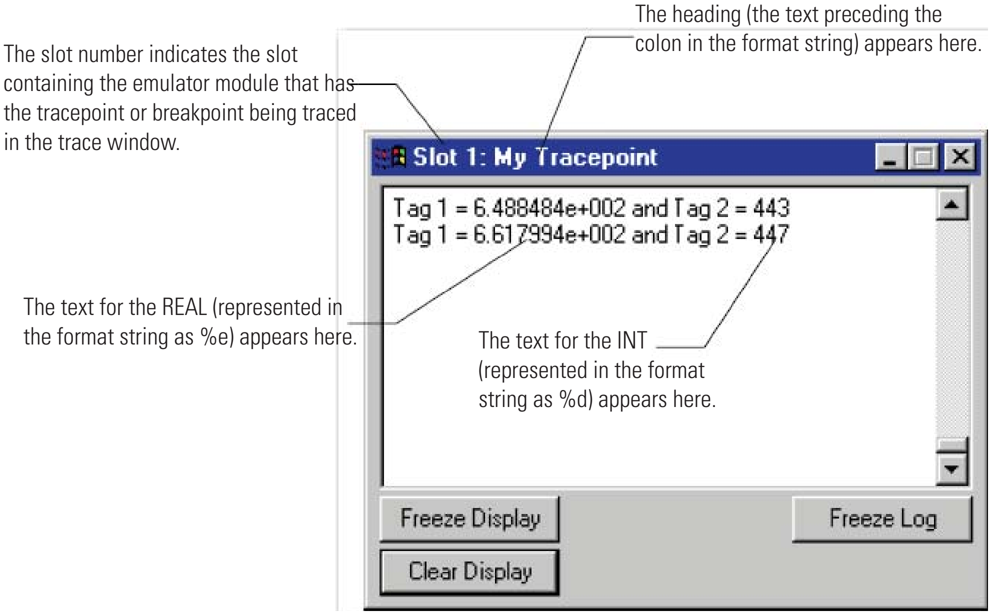
For example, you could format a tracepoint string as shown.

My tracepoint:Tag 1 = %e and Tag 2 = %d

The %e formats the first traced tag as double-precision float with an exponent, and %d formats the second traced tag as a signed decimal integer.

In this case, you would have a tracepoint instruction that has two Trace This operands (one for a REAL and one for an INT, although the value of any tag can be formatted with any flag).

The resulting tracepoint window that would appear when the tracepoint is triggered would look like the example.



```
Tag 1 = 6.488484e+002 and Tag 2 = 443
Tag 1 = 6.617994e+002 and Tag 2 = 447
```

The slot number indicates the slot containing the emulator module that has the tracepoint or breakpoint being traced in the trace window.

The heading (the text preceding the colon in the format string) appears here.

The text for the REAL (represented in the format string as %e) appears here.

The text for the INT (represented in the format string as %d) appears here.

Arithmetic Status Flags: Not affected

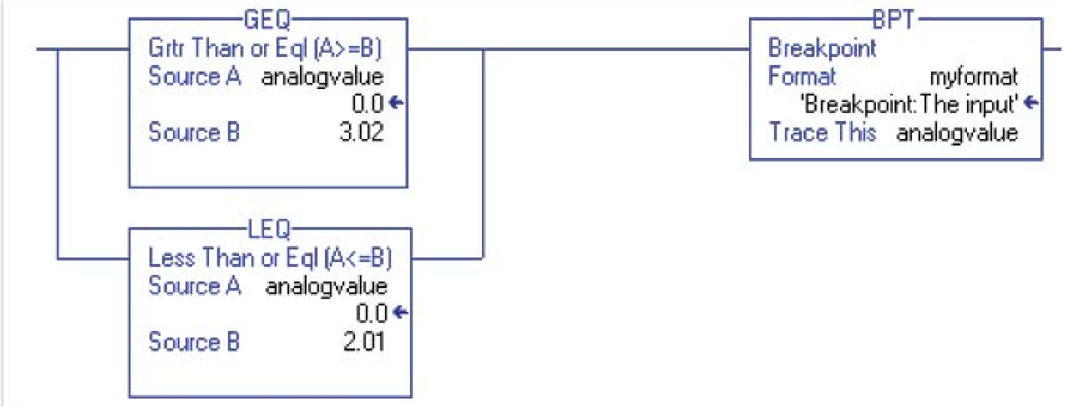
Fault Conditions: None

Execution:

Condition	Relay Ladder Action
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The rung-condition-out is set to true. Execution jumps to the rung that contains the LBL instruction with the referenced label name.
Postscan	The rung-condition-out is set to false.

Example: You can display many tag values with the BPT instruction. However, the formatting string can contain only 82 characters. Because the formatting string requires two characters for each tag you want in the breakpoint, you cannot trace more than 41 tags with a single BPT instruction. However, to separate tag data in your traces, you will need to include spaces and other formatting, thus reducing the number of tag values that one BPT instruction can effectively display to far fewer than 41.

This rung shows a breakpoint that stops program execution when an analog value is greater than 3.02 or less than 2.01.

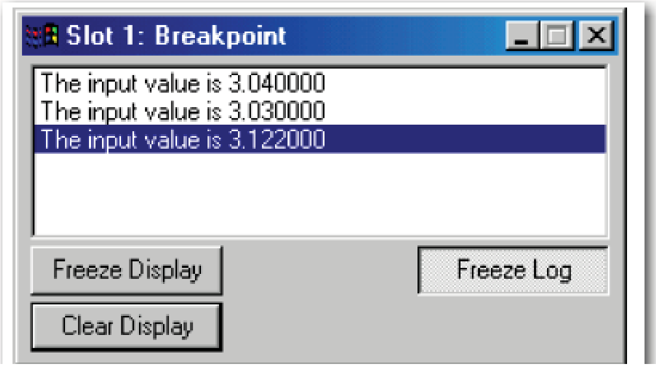


You want to display the breakpoint information in the Format string (myformat). In this case, the format string contains the following text:

```
Breakpoint:The input value is %f
```

When the breakpoint triggers, the breakpoint trace window shows the characters before the colon ('Breakpoint') in the title bar of the trace window. The other characters make up the traces. In this example, %f represents the first (and in this case, the only) tag to be traced ('analogvalue').

The resulting traces appear as shown here.

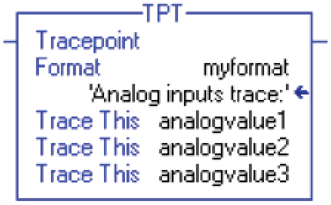


```
The input value is 3.040000
The input value is 3.030000
The input value is 3.122000
```

Tracepoints (TPT)

Trace points log data you select when a rung is true.

Operands:



Relay Ladder

Operand	Type	Format	Description
Format	String	Tag	A string that sets the formatting for the trace reports (both on-screen and logged to disk).
Trace This	BOOL, SINT, INT, DINT, REAL	Tag	The tag you want to trace.

Description:

Tracepoints are programmed with the tracepoint output instruction (TPT). When the inputs on a rung containing a TPT instruction are true, the TPT instruction writes a trace entry to a trace display or log file.

You can trace many tags with the TPT instruction. However, the formatting string can contain only 82 characters. Because the formatting string requires two characters for each tag you want to trace, you cannot trace more than 41 tags with a single TPT instruction. However, to separate tag data in your traces, you will need to include spaces and other formatting, thus reducing the number of tags that one TPT instruction can effectively trace to far fewer than 41.

String Format

With the Format string in the tracepoint and breakpoint instructions, you can control how the traced tags appear in the traces or breakpoint windows. The format of the string is as shown here:

heading:(text)%(type)

where *heading* is a text string identifying the tracepoint or breakpoint, *text* is a string describing the tag (or any other text you choose), and %(*type*) indicates the format of the tag. You need one type indicator for each tag you are tracing with the tracepoint or breakpoint instruction.

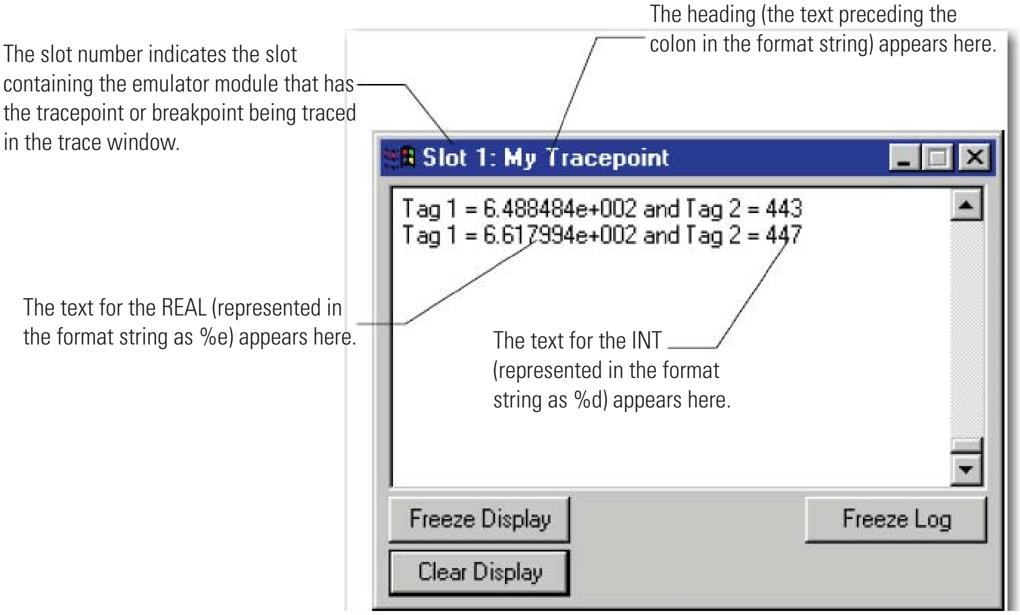
For example, you could format a tracepoint string as shown.

My tracepoint:Tag 1 = %e and Tag 2 = %d

The %e formats the first traced tag as double-precision float with an exponent, and %d formats the second traced tag as a signed decimal integer.

In this case, you would have a tracepoint instruction that has two Trace This operands (one for a REAL and one for an INT, although the value of any tag can be formatted with any flag).

The resulting tracepoint window that would appear when the tracepoint is triggered would look like the example.



```
Tag 1 = 6.488484e+002 and Tag 2 = 443
Tag 1 = 6.617994e+002 and Tag 2 = 447
```

The heading (the text preceding the colon in the format string) appears here.

The slot number indicates the slot containing the emulator module that has the tracepoint or breakpoint being traced in the trace window.

The text for the REAL (represented in the format string as %e) appears here.

The text for the INT (represented in the format string as %d) appears here.

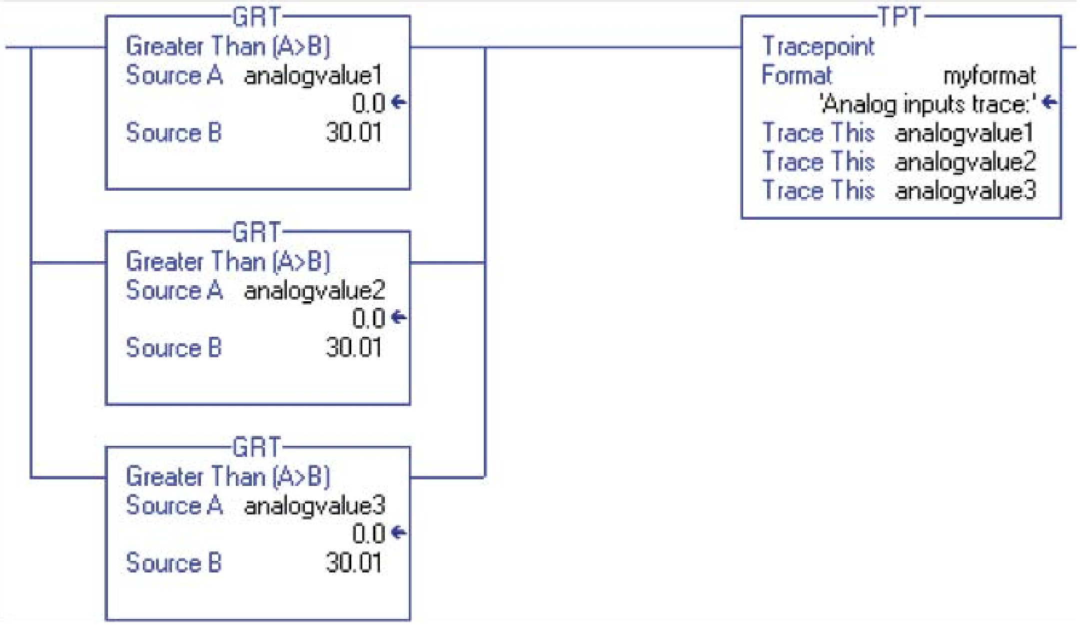
.**Arithmetic Status Flags:** Not affected

Fault Conditions: None

Execution:

Condition:	**Relay Ladder Action**
Prescan	The rung-condition-out is set to false.
Rung-condition-in is false	The rung-condition-out is set to false.
Rung-condition-in is true	The rung-condition-out is set to true. Execution jumps to the rung that contains the LBL instruction with the referenced label name.
Postscan	The rung-condition-out is set to false.

Example: This rung triggers a trace of three analog values when any one of them exceeds a given value (30.01).

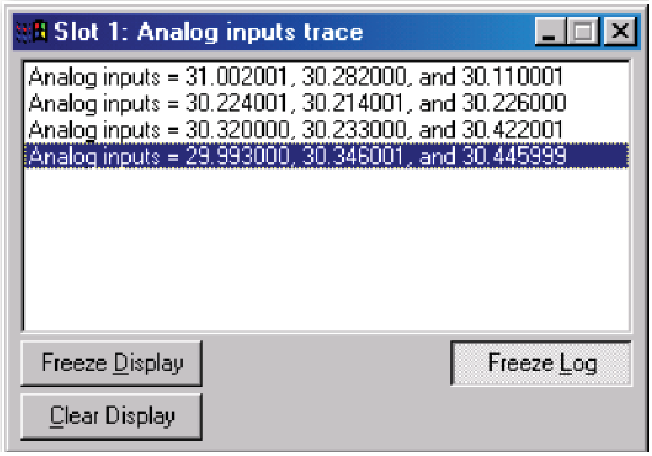


You want to display the tracepoint information in the Format string (myformat). In this case, the format string contains this text:

```
Analog inputs trace:Analog inputs = %f, %f, and %f
```

When the tracepoint triggers, the characters before the colon ('Analog inputs trace') appear in the title bar of the trace window. The other characters make up the traces. In this example, %f represents the tags to be traced ('analogvalue1,' 'analogvalue2,' and 'analogvalue3').

The resulting traces appear as shown here.



```
Analog inputs = 31.002001, 30.282000, and 30.110001
Analog inputs = 30.224001, 30.214001, and 30.226000
Analog inputs = 30.320000, 30.233000, and 30.422001
Analog inputs = 29.993000, 30.346001, and 30.445999
```

When this trace is logged to disk, the characters before the colon appear in the traces.

This indicates which tracepoint caused which trace entry. This is an example of a trace entry. ‘Analog inputs trace:’ is the heading text from the tracepoint's format string.

Analog inputs trace: Analog inputs = 31.00201, 30.282000, and 30.110001.

Common Attributes

Introduction

This appendix describes attributes that are common to the Logix instructions.

For Information About	See Page
Immediate Values	643
Data Conversions	644

Immediate Values

When you enter an immediate value (constant) in decimal format (for example, -2, 3) the controller stores the value by using 32 bits. If you enter a value in a radix other than decimal, such as binary or hexadecimal, and do not specify all 32 bits, the controller places a zero in the bits that you do not specify (zero-fill).

EXAMPLE Zero-filling of immediate values

If you enter	The controller stores
-1	16#ffff ffff (-1)
16#ffff (-1)	16#0000 ffff (65535)
8#1234 (668)	16#0000 029c (668)
2#1010 (10)	16#0000 000a (10)

Data Conversions

Data conversions occur when you mix data types in your programming.

When programming	Conversions can occur when you
Relay Ladder Logic	Mix data types for the parameters within one instruction
Function Block	Wire two parameters that have different data types

Instructions execute faster and require less memory if all the operands of the instruction use:

- the same data type.
- an optimal data type:
  - in the 'Operands' section of each instruction in this manual, a **bold** data type indicates an optimal data type.
  - the DINT and REAL data types are typically the optimal data types.
  - most function block instruction support only one data type (the optimal data type) for its operands.

If you mix data types and use tags that are not the optimal data type, the controller converts the data according to these rules.

- Are any of the operands a REAL value?

If	Then input operands (for example, source, tag in an expression, limit) convert to
Yes	REALs
No	DINTs

- After instruction execution, the result (a DINT or REAL value) converts to the destination data type, if necessary.

You cannot specify a BOOL tag in an instruction that operates on integer or REAL data types.

Because the conversion of data takes additional time and memory, you can increase the efficiency of your programs by:

- using the same data type throughout the instruction.
- minimizing the use of the SINT or INT data types.

In other words, use all DINT tags or all REAL tags, along with immediate values, in your instructions.

The following sections explain how the data is converted when you use SINT or INT tags or when you mix data types.

SINT or INT to DINT

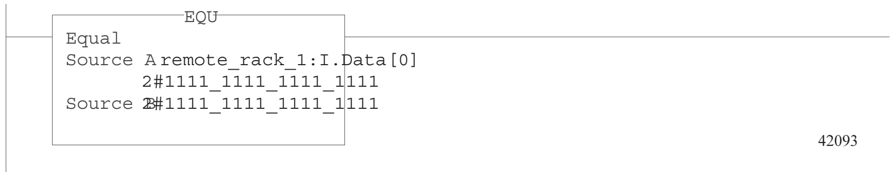
For those instructions that convert SINT or INT values to DINT values, the ‘Operands’ sections in this manual identify the conversion method.

This conversion method	Converts data by placing
Sign-extension	The value of the leftmost bit (the sign of the value) into each bit position to the left of the existing bits until there are 32 bits.
Zero-fill	Zeroes to the left of the existing bits until there are 32 bits.

The following example shows the results of converting a value using sign-extension and zero-fill.

This value	2#1111_1111_1111_1111	(-1)
Converts to this value by sign-extension	2#1111_1111_1111_1111_1111_1111_1111_1111	(-1)
Converts to this value by zero-fill	2#0000_0000_0000_0000_1111_1111_1111_1111	(65535)

Because immediate values are always zero-filled, the conversion of a SINT or INT value may produce unexpected results. In the following example, the comparison is false because Source A, an INT, converts by sign-extension; while Source B, an immediate value, is zero-filled.



Source A remote_rack_1:I.Data[0]
2#1111_1111_1111_1111
Source B 2#1111_1111_1111_1111

If you use a SINT or INT tag and an immediate value in an instruction that converts data by sign-extension, use one of these methods to handle immediate values.

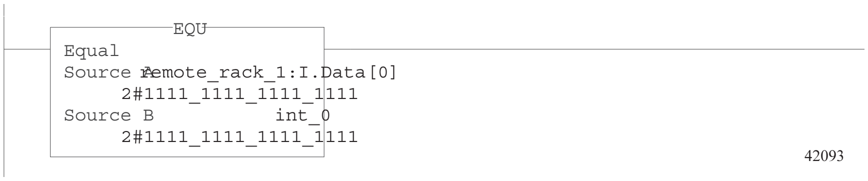
- Specify any immediate value in the decimal radix.
- If you are entering the value in a radix other than decimal, specify all 32 bits of the immediate value. To do so, enter the value of the leftmost bit into each bit position to its left until there are 32 bits.
- Create a tag for each operand and use the same data type throughout the instruction. To assign a constant value, either:
  - Enter it into one of the tags.
  - Add a MOV instruction that moves the value into one of the tags.
- Use a MEQ instruction to check only the required bits.

The following examples show two ways to mix an immediate value with an INT tag. Both examples check the bits of a 1771 I/O module to determine if all the bits are on. Since the input data word of a 1771 I/O module is an INT tag, it is easiest to use a 16-bit constant value.

EXAMPLE

Mixing an INT tag with an immediate value

Since *remote_rack_1:I.Data[0]* is an INT tag, the value to check it against is also entered as an INT tag.



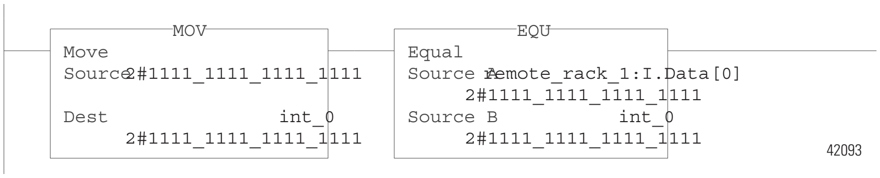
```
EQU
Equal
Source A remote_rack_1:I.Data[0]
     2#1111_1111_1111_1111
Source B           int_0
     2#1111_1111_1111_1111
```

42093

EXAMPLE

Mixing an INT tag with an immediate value

Since *remote_rack_1:I.Data[0]* is an INT tag, the value to check it against first moves into *int_0*, also an INT tag. The EQU instruction then compares both tags.



```
MOV
Move
Source 2#1111_1111_1111_1111

Dest               int_0
     2#1111_1111_1111_1111

EQU
Equal
Source A remote_rack_1:I.Data[0]
     2#1111_1111_1111_1111
Source B           int_0
     2#1111_1111_1111_1111
```

42093

Integer to REAL

The controller stores REAL values in IEEE single-precision, floating-point number format. It uses one bit for the sign of the value, 23 bits for the base value, and eight bits for the exponent (32 bits total). If you mix an integer tag (SINT, INT, or DINT) and a REAL tag as inputs in the same instruction, the controller converts the integer value to a REAL value before the instruction executes.

- A SINT or INT value always converts to the same REAL value.
- A DINT value may not convert to the same REAL value:
  - A REAL value uses up to 24 bits for the base value (23 stored bits plus a 'hidden' bit).
  - A DINT value uses up to 32 bits for the value (one for the sign and 31 for the value).
  - If the DINT value requires more than 24 significant bits, it *may not* convert to the same REAL value. If it will not, the controller rounds to the nearest REAL value by using 24 significant bits.

DINT to SINT or INT

To convert a DINT value to a SINT or INT value, the controller truncates the upper portion of the DINT and sets the overflow status flag, if necessary. The following example shows the result of a DINT to SINT or INT conversion.

EXAMPLE Conversion of a DINT to an INT and a SINT

This DINT value	Converts to this smaller value	
16#0001_0081 (65,665)	INT:	16#0081 (129)
	SINT:	16#81 (-127)

REAL to an Integer

To convert a REAL value to an integer value, the controller rounds the fractional part and truncates the upper portion of the non-fractional part. If data is lost, the controller sets the overflow status flag. Numbers round as in the following examples.

- Numbers other than *x*.5 round to the nearest whole number.
- *X*.5 rounds to the nearest even number.

The following example shows the result of converting REAL values to DINT values.

EXAMPLE Conversion of REAL values to DINT values

This REAL value	Converts to this DINT value
-2.5	-2
-1.6	-2
-1.5	-2
-1.4	-1
1.4	1
1.5	2
1.6	2
2.5	2

IMPORTANT The arithmetic status flags are set based on the value being stored. Instructions that normally do not affect arithmetic status keywords might appear to do so if type conversion occurs because of mixed data types for the instruction parameters. The type conversion process sets the arithmetic status keywords.

Function Block Attributes

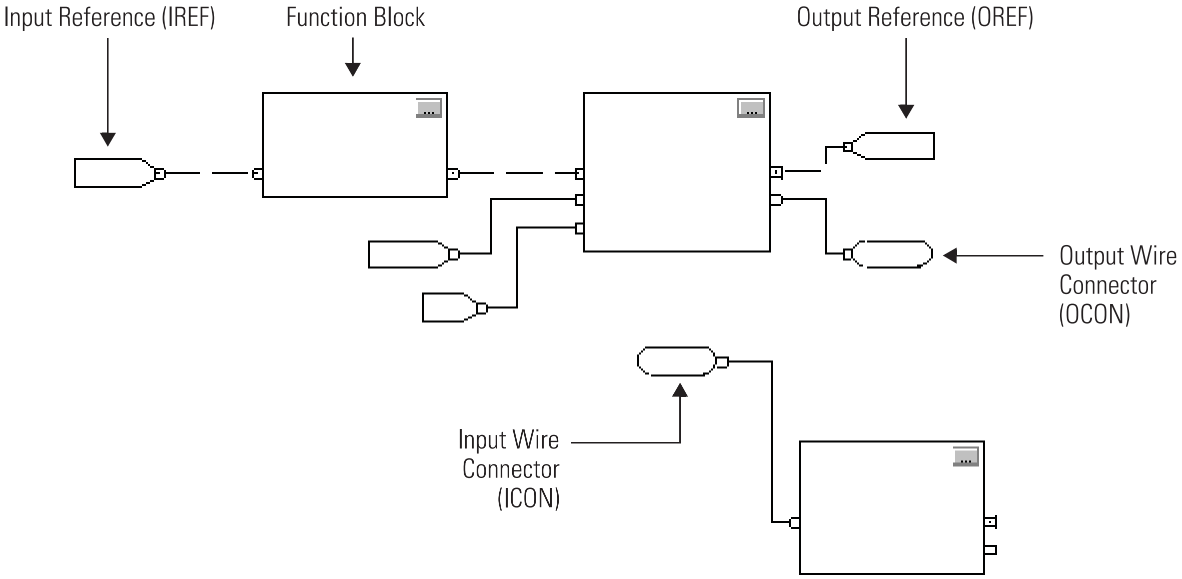
Introduction

This appendix describes issues that are unique with function block instructions. Review the information in this appendix to make sure you understand how your function block routines will operate.

IMPORTANT	When programming in function block, restrict the range of engineering units to +/-$10^{+/-15}$ because internal floating point calculations are done by using single precision floating point. Engineering units outside of this range may result in a loss of accuracy if results approach the limitations of single precision floating point (+/-$10^{+/-38}$).

Function Block Elements

To control a device, use these elements.

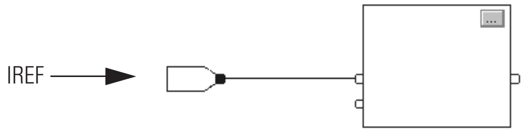


Use the table to choose your function block elements.

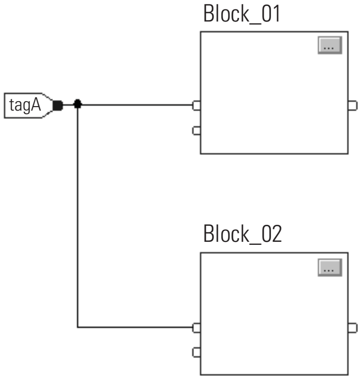
If you want to	Use a
Supply a value from an input device or tag	Input reference (IREF)
Send a value to an output device or tag	Output reference (OREF)
Perform an operation on an input value or values and produce an output value or values	Function block
Transfer data between function blocks when they are: • far apart on the same sheet. • on different sheets within the same routine.	Output wire connector (OCON) and an input wire connector (ICON)
Disperse data to several points in the routine	Single output wire connector (OCON) and multiple input wire connectors (ICON)

Latching Data

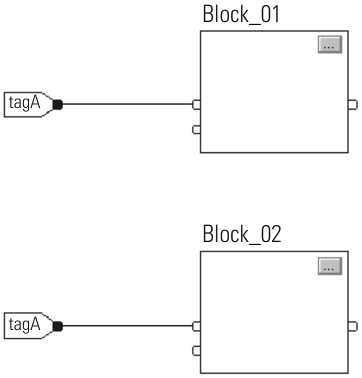
If you use an IREF to specify input data for a function block instruction, the data in that IREF is latched for the scan of the function block routine. The IREF latches data from program-scoped and controller-scoped tags. The controller updates all IREF data at the beginning of each scan.



In this example, the value of tagA is stored at the beginning of the routine's execution. The stored value is used when Block_01 executes. The same stored value is also used when Blcock_02 executes. If the value of tagA changes during execution of the routine, the stored value of tagA in the IREF does not change until the next execution of the routine.

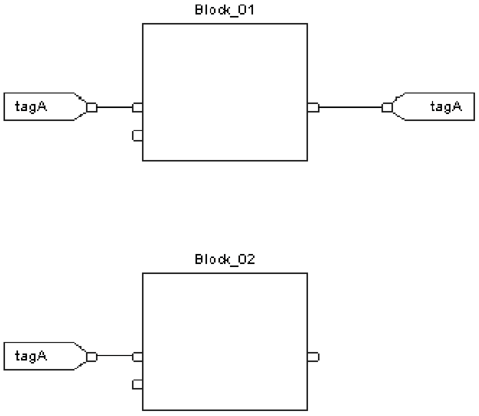


This example is the same as the one above. The value of tagA is stored only once at the beginning of the routine's execution. The routine uses this stored value throughout the routine.



Starting with RSLogix 5000 software, version 11, you can use the same tag in multiple IREFs and an OREF in the same routine. Because the values of tags in IREFs are latched every scan through the routine, all IREFs will use the same value, even if an OREF obtains a different tag value during execution of the routine.

In this example, if tagA has a value of 25.4 when the routine starts executing this scan, and Block_01 changes the value of tagA to 50.9, the second IREF wired into Block_02 will still use a value of 25.4 when Block_02 executes this scan. The new tagA value of 50.9 will not be used by any IREFs in this routine until the start of the next scan.



Order of Execution

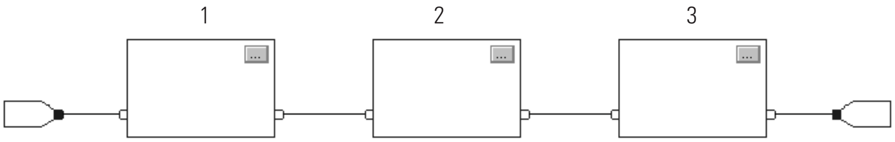
The RSLogix 5000 programming software automatically determines the order of execution for the function blocks in a routine when you:

- verify a function block routine.
- verify a project that contains a function block routine.
- download a project that contains a function block routine.

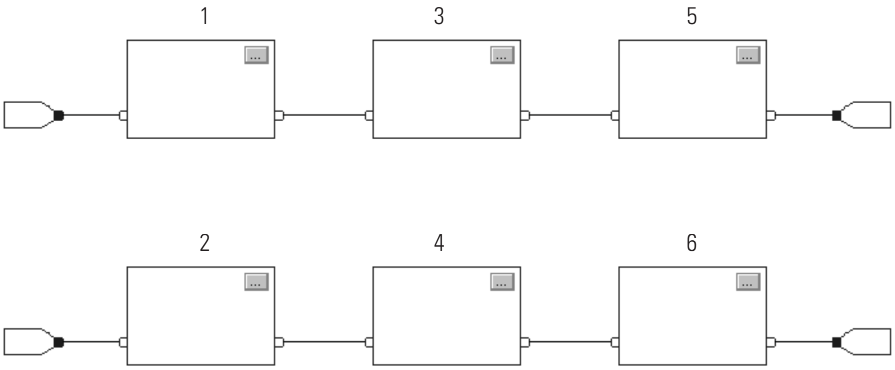
You define execution order by wiring function blocks together and indicating the data flow of any feedback wires, if necessary.

If function blocks are not wired together, it does not matter which block executes first. There is no data flow between the blocks.

If you wire the blocks sequentially, the execution order moves from input to output. The inputs of a block require data to be available before the controller can execute that block. For example, block 2 has to execute before block 3 because the outputs of block 2 feed the inputs of block 3.

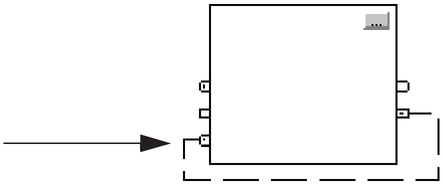


Execution order is relative only to the blocks that are wired together. The following example is fine because the two groups of blocks are not wired together. The blocks within a specific group execute in the appropriate order in relation to the blocks in that group.



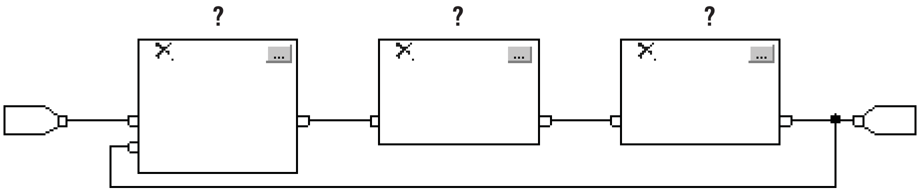
Resolve a Loop

To create a feedback loop around a block, wire an output pin of the block to an input pin of the same block. The following example is okay. The loop contains only a single block, so execution order does not matter.

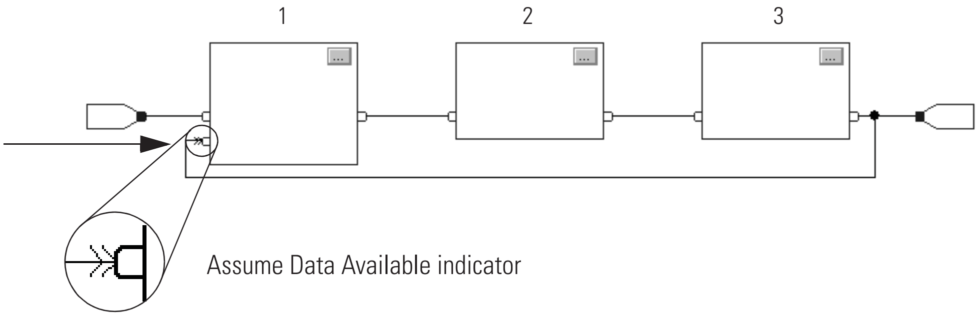


This input pin uses an output that the block produced on the previous scan.

If a group of blocks are in a loop, the controller cannot determine which block to execute first. In other words, it cannot resolve the loop.



To identify which block to execute first, mark the input wire that creates the loop (the feedback wire) with the *Assume Data Available* indicator. In the following example, block 1 uses the output from block 3 that was produced in the previous execution of the routine.

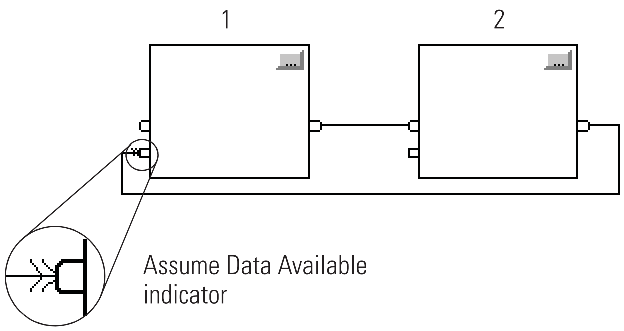
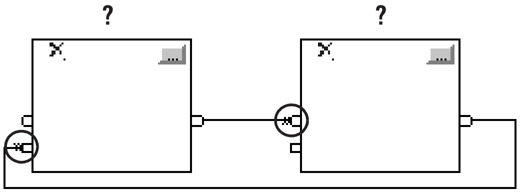


This input pin uses the output that block 3 produced on the previous scan.

Assume Data Available indicator

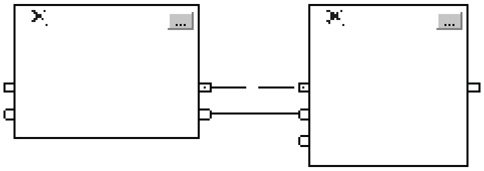
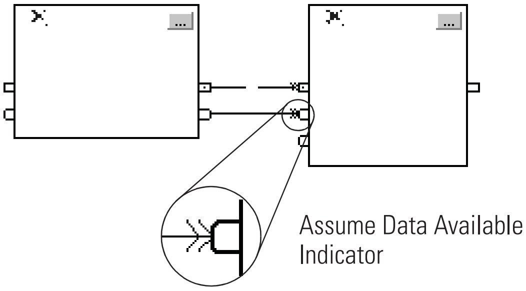
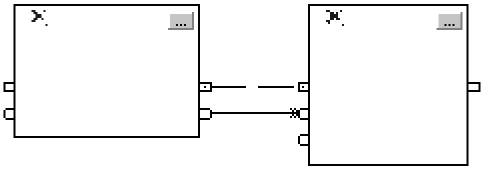
The *Assume Data Available* indicator defines the data flow within the loop. The arrow indicates that the data serves as input to the first block in the loop.

Do **not** mark all the wires of a loop with the Assume Data Available indicator.

This is okay	This is NOT okay
 The Assume Data Available indicator defines the data flow within the loop.	 The controller cannot resolve the loop because all the wires use the Assume Data Available indicator.

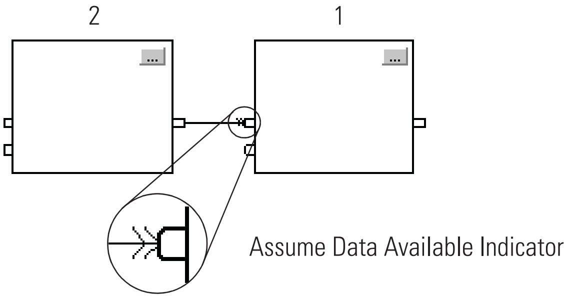
Resolve Data Flow Between Two Blocks

If you use two or more wires to connect two blocks, use the same data flow indicators for all of the wires between the two blocks.

This is okay	This is not okay
 Neither wire uses the Assume Data Available indicator.  Both wires use the Assume Data Available indicator.	 One wire uses the Assume Data Available indicator while the other wire does not.

Create a One Scan Delay

To produce a one scan delay between blocks, use the *Assume Data Available* indicator. In the following example, block 1 executes first. It uses the output from block 2 that was produced in the previous scan of the routine.



Summary

A function block routine executes in this order.

1. The controller latches all data values in IREFs.
2. The controller executes the other function blocks in the order determined by how they are wired.
3. The controller writes outputs in OREFs.

Function Block Responses to Overflow Conditions

In general, the function block instructions that maintain history do not update history with ±NAN, or ±INF values when an overflow occurs. Each instruction has one of these responses to an overflow condition.

Response 1	**Response 2**	**Response 3**
Blocks execute their algorithm and check the result for ±NAN or ±INF. If ±NAN or ±INF, the block outputs ±NAN or ±INF.	Blocks with output limiting execute their algorithm and check the result for ±NAN or ±INF. The output limits are defined by the HighLimit and LowLimit input parameters. If ±INF, the block outputs a limited result. If ±NAN, the output limits are not used and the block outputs ±NAN.	The overflow condition does not apply. These instructions typically have a boolean output.
ALMNTCH	HLL	BANDOSRI
DEDTPMUL	INTG	BNOTRESD
DERVPOSP	PI	BORRTOR
ESELRLIM	PIDE	BXORSETD
FGENRMPS	SCL	CUTDTOFR
HPFSCRV	SOC	D2SDTONR
LDL2SEL		D3SD
LDLGSNEG		DFF
LPFSRTP		JKFF
MAVESSUM		OSFI
MAXCTOT		
MINCUPDN		
MSTD		
MUX		

Timing Modes

These process control and drives instructions support different timing modes.

DEDT
DERV
HPF
INTG
LDL2
LDLG
LPF
NTCH
PI
PIDE
RLIM
SCRV
SOC
TOT

There are three different timing modes.

<table>
<tr><th>Timing Mode</th><th colspan="2">Description</th></tr>
<tr><td rowspan="5">Periodic</td><td colspan="2">Periodic mode is the default mode and is suitable for most control applications. We recommend that you place the instructions that use this mode in a routine that executes in a periodic task. The delta time (DeltaT) for the instruction is determined as follows:</td></tr>
<tr><td>If the instruction executes in a</td><td>Then DeltaT equals</td></tr>
<tr><td>Periodic task</td><td>Period of the task</td></tr>
<tr><td>Event or continuous task</td><td>Elapsed time since the previous execution
The controller truncates the elapsed time to whole milliseconds (ms). For example, if the elapsed time = 10.5 ms, the controller sets DeltaT = 10 ms.</td></tr>
<tr><td colspan="2">The update of the process input needs to be synchronized with the execution of the task or sampled 5-10 times faster than the task executes in order to minimize the sampling error between the input and the instruction.</td></tr>
<tr><td>Oversample</td><td colspan="2">In oversample mode, the delta time (DeltaT) used by the instruction is the value written into the OversampleDT parameter of the instruction. If the process input has a time stamp value, use the real time sampling mode instead.
Add logic to your program to control when the instruction executes. For example, you can use a timer set to the OversampleDeltaT value to control the execution by using the EnableIn input of the instruction.
The process input needs to be sampled 5-10 times faster than the instruction is executed in order to minimize the sampling error between the input and the instruction.</td></tr>
<tr><td>Real time sampling</td><td colspan="2">In the real time sampling mode, the delta time (DeltaT) used by the instruction is the difference between two time stamp values that correspond to the updates of the process input. Use this mode when the process input has a time stamp associated with its updates and you need precise coordination.
The time stamp value is read from the tag name entered for the RTSTimeStamp parameter of the instruction. Normally this tag name is a parameter on the input module associated with the process input.
The instruction compares the configured RTSTime value (expected update period) against the calculated DeltaT to determine if every update of the process input is being read by the instruction. If DeltaT is not within 1 millisecond of the configuration time, the instruction sets the RTSMissed status bit to indicate that a problem exists reading updates for the input on the module.</td></tr>
</table>

Time-based instructions require a constant value for DeltaT in order for the control algorithm to properly calculate the process output. If DeltaT varies, a discontinuity occurs in the process output. The severity of the discontinuity depends on the instruction and range over which DeltaT varies.

A discontinuity occurs if the:

- instruction is not executed during a scan.
- instruction is executed multiple times during a task.
- task is running and the task scan rate or the sample time of the process input changes.
- user changes the time-base mode while the task is running.
- order parameter is changed on a filter block while the task is running.

  Changing the Order parameter selects a different control algorithm within the instruction.

Common Instruction Parameters for Timing Modes

The instructions that support time-base modes have these input and output parameters.

Input Parameters

Input Parameter	Data Type	Description
TimingMode	DINT	Selects timing execution mode. **Value:** **Description:** 0 Periodic mode 1 Oversample mode 2 Real time sampling mode Valid = 0…2 Default = 0 When TimingMode = 0 and task is periodic, periodic timing is enabled and DeltaT is set to the task scan rate. When TimingMode = 0 and task is event or continuous, periodic timing is enabled and DeltaT is set equal to the elapsed time span since the last time the instruction was executed. When TimingMode = 1, oversample timing is enabled and DeltaT is set to the value of the OversampleDT parameter. When TimingMode = 2, real time sampling timing is enabled and DeltaT is the difference between the current and previous time stamp values read from the module associated with the input. If TimingMode invalid, the instruction sets the appropriate bit in Status.
OversampleDT	REAL	Execution time for oversample timing. The value used for DeltaT is in seconds. If TimingMode = 1, then OversampleDT = 0.0 disables the execution of the control algorithm. If invalid, the instruction sets DeltaT = 0.0 and sets the appropriate bit in Status. Valid = 0…4194.303 seconds Default = 0.0
RTSTime	DINT	Module update period for real time sampling timing. The expected DeltaT update period is in milliseconds. The update period is normally the value that was used to configure the module's update time. If invalid, the instruction sets the appropriate bit in Status and disables RTSMissed checking. Valid = 1…32,767ms Default = 1

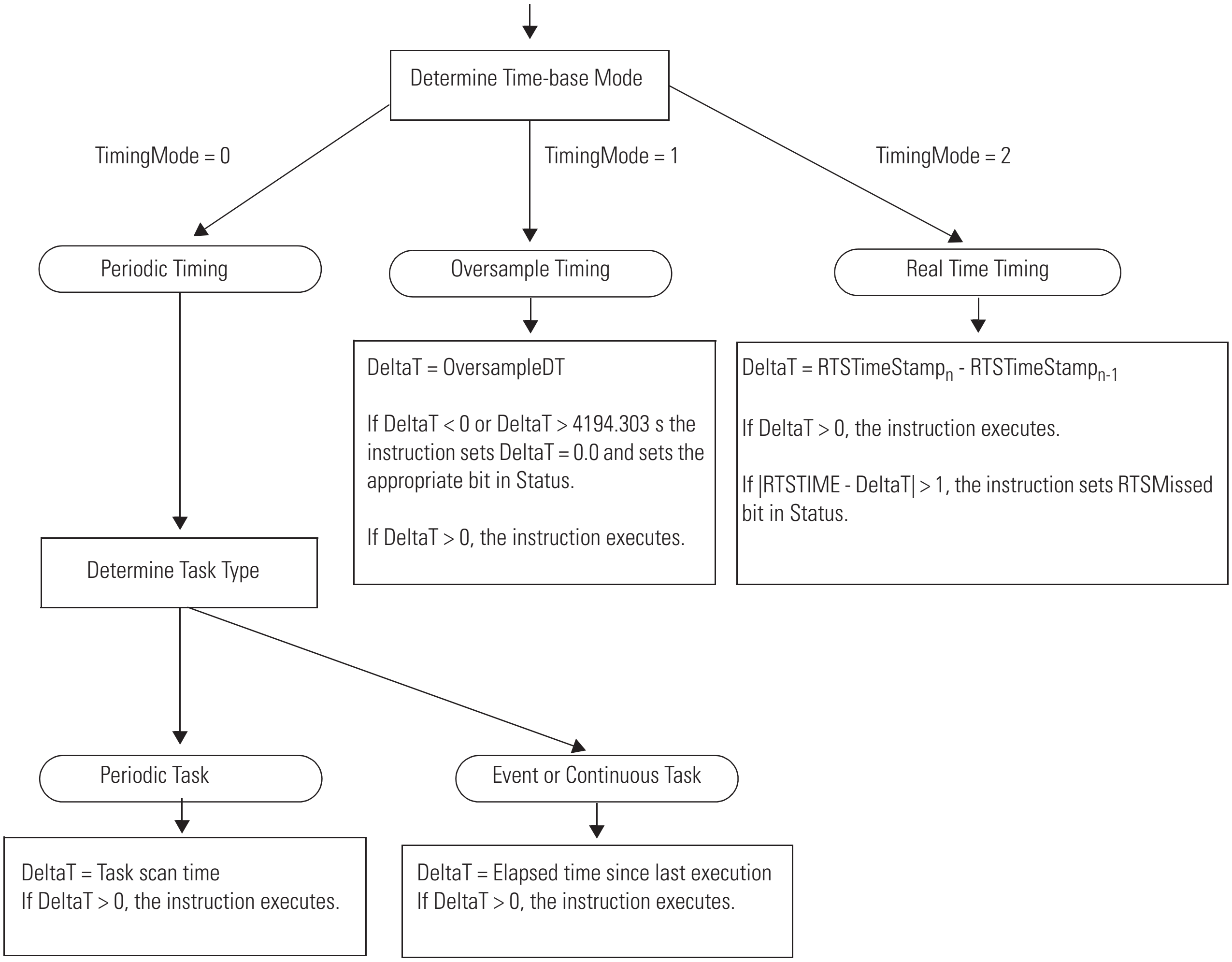
Input Parameter	Data Type	Description
RTSTimeStamp	DINT	Module time stamp value for real time sampling timing. The time stamp value that corresponds to the last update of the input signal. This value is used to calculate DeltaT. If invalid, the instruction sets the appropriate bit in Status, disables execution of the control algorithm, and disables RTSMissed checking. Valid =1…32,767ms (wraps from 32767 to 0) 1 count = 1 millisecond Default = 0

Output Parameters

Output Parameter	Data Type	Description
DeltaT	REAL	Elapsed time between updates. This is the elapsed time in seconds used by the control algorithm to calculate the process output. Periodic: DeltaT = task scan rate if task is Periodic task, DeltaT = elapsed time since previous instruction execution if task is Event or Continuous task Oversample: DeltaT = OversampleDT Real Time Sampling: DeltaT = $(\text{RTSTimeStamp}_n - \text{RTSTimeStamp}_{n-1})$
Status	DINT	Status of the function block.
TimingModeInv (Status.27)	BOOL	Invalid TimingMode value.
RTSMissed (Status.28)	BOOL	Only used in real time sampling mode. Set when ABS | DeltaT - RTSTime | > 1 (.001 second).
RTSTimeInv (Status.29)	BOOL	Invalid RTSTime value.
RTSTimeStampInv (Status.30)	BOOL	Invalid RTSTimeStamp value.
DeltaTInv (Status.31)	BOOL	Invalid DeltaT value.

Overview of Timing Modes

The following diagram shows how an instruction determines the appropriate timing mode.



Program/Operator Control

Several instructions support the concept of Program/Operator control. These instructions include:

- Enhanced Select (ESEL)
- Totalizer (TOT)
- Enhanced PID (PIDE)
- Ramp/Soak (RMPS)
- Discrete 2-State Device (D2SD)
- Discrete 3-State Device (D3SD)

Program/Operator control lets you control these instructions simultaneously from both your user program and from an operator interface device. When in Program control, the instruction is controlled by the Program inputs to the instruction; when in Operator control, the instruction is controlled by the Operator inputs to the instruction.

Program or Operator control is determined by using these inputs.

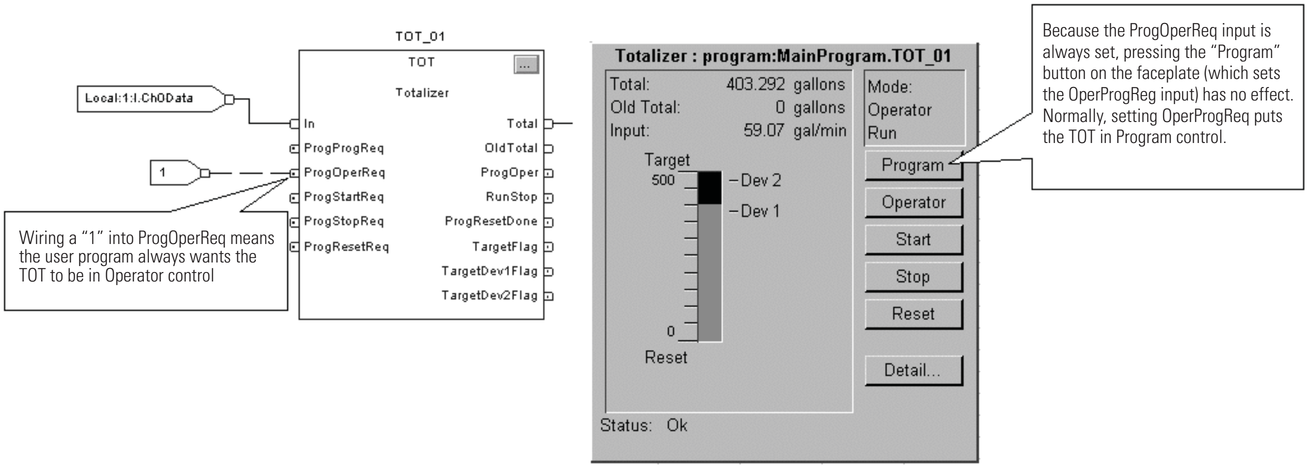
Input	Description
.ProgProgReq	A program request to go to Program control.
.ProgOperReq	A program request to go to Operator control.
.OperProgReq	An operator request to go to Program control.
.OperOperReq	An operator request to go to Operator control.

To determine whether an instruction is in Program or Control control, examine the ProgOper output. If ProgOper is set, the instruction is in Program control; if ProgOper is cleared, the instruction is in Operator control.

Operator control takes precedence over Program control if both input request bits are set. For example, if ProgProgReq and ProgOperReq are both set, the instruction goes to Operator control.

The Program request inputs take precedence over the Operator request inputs. This provides the capability to use the ProgProgReq and ProgOperReq inputs to ‘lock’ an instruction in a desired control.

For example, let's assume that a Totalizer instruction will always be used in Operator control, and your user program will never control the running or stopping of the Totalizer. In this case, you could wire a literal value of 1 into the ProgOperReq. This would prevent the operator from ever putting the Totalizer into Program control by setting the OperProgReq from an operator interface device.



Total: 403.292 gallons
Old Total: 0 gallons
Input: 59.07 gal/min
Mode: Operator Run
Status: Ok

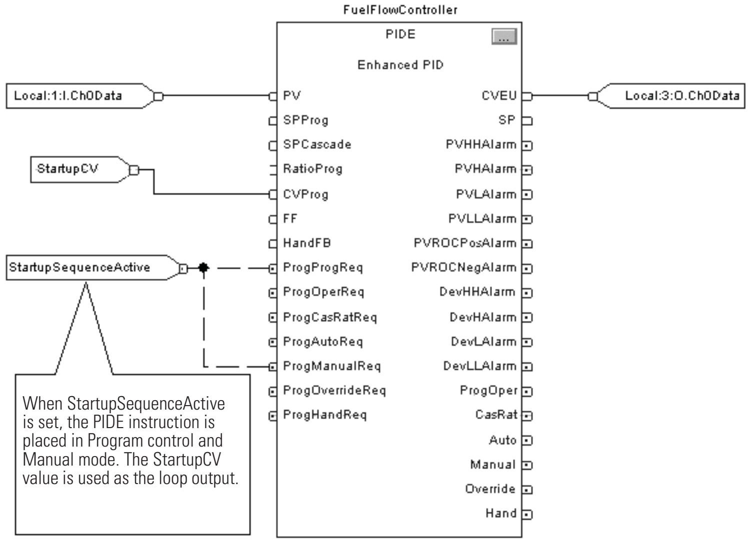
Wiring a "1" into ProgOperReq means the user program always wants the TOT to be in Operator control

Because the ProgOperReq input is always set, pressing the "Program" button on the faceplate (which sets the OperProgReg input) has no effect. Normally, setting OperProgReq puts the TOT in Program control.

Likewise, constantly setting the ProgProgReq can 'lock' the instruction into Program control. This is useful for automatic startup sequences when you want the program to control the action of the instruction without worrying about an operator inadvertently taking control of the instruction.

In this example, you have the program set the ProgProgReq input during the startup, and then clear the ProgProgReq input once the startup was complete. Once the ProgProgReq input is cleared, the instruction remains in Program control until it receives a request to change. For example, the operator could set the OperOperReq input from a faceplate to take over control of that instruction.

The following example shows how to lock an instruction into Program control.



When StartupSequenceActive is set, the PIDE instruction is placed in Program control and Manual mode. The StartupCV value is used as the loop output.

Operator request inputs to an instruction are always cleared by the instruction when it executes. This allows operator interfaces to work with these instructions by merely setting the desired mode request bit. You don't have to program the operator interface to reset the request bits. For example, if an operator interface sets the OperAutoReq input to a PIDE instruction, when the PIDE instruction executes, it determines what the appropriate response should be and clears the OperAutoReq.

Program request inputs are not normally cleared by the instruction because these are normally wired as inputs into the instruction. If the instruction clears these inputs, the input would just get set again by the wired input. There might be situations where you want to use other logic to set the Program requests in such a manner that you want the Program requests to be cleared by the instruction. In this case, you can set the ProgValueReset input and the instruction will always clear the Program mode request inputs when it executes.

In this example, a rung of ladder logic in another routine is used to one-shot latch a ProgAutoReq to a PIDE instruction when a push button is pushed. Because the PIDE instruction automatically clears the Program mode requests, you don't have to write any ladder logic to clear the ProgAutoReq after the routine executes, and the PIDE instruction will receive only one request to go to Auto every time the push button is pressed.

When the TIC101AutoReq push button is pressed, one-shot latch ProgAutoReq for the PIDE instruction TIC101. TIC101 has been configured with the ProgValueReset input set, so when the PIDE instruction executes, it automatically clears ProgAutoReq.

Notes:

Appendix C

Structured Text Programming

Introduction

This appendix describes issues that are unique with structured text programming. Review the information in this appendix to make sure you understand how your structured text programming executes.

Structured Text Syntax

Structured text is a textual programming language that uses statements to define what to execute.

- Structured text is not case sensitive.
- Use tabs and carriage returns (separate lines) to make your structured text easier to read. They have no effect on the execution of the structured text.

Structured text is not case sensitive. Structured text can contain these components.

<table>
<tr><th>Term</th><th colspan="2">Definition</th><th>Examples</th></tr>
<tr><td>Assignment
(see page 669)</td><td colspan="2">Use an assignment statement to assign values to tags.
The := operator is the assignment operator.
Terminate the assignment with a semi colon “;”.</td><td>tag := expression;</td></tr>
<tr><td rowspan="5">Expression
(see page 671)</td><td colspan="2">An expression is part of a complete assignment or construct statement. An expression evaluates to a number (numerical expression) or to a true or false state (BOOL expression).
An expression contains:</td><td></td></tr>
<tr><td>Tags</td><td>A named area of the memory where data is stored (BOOL, SINT,INT,DINT, REAL, string).</td><td>value1</td></tr>
<tr><td>Immediate</td><td>A constant value.</td><td>4</td></tr>
<tr><td>Operators</td><td>A symbol or mnemonic that specifies an operation within an expression.</td><td>tag1 + tag2
tag1 >= value1</td></tr>
<tr><td>Functions</td><td>When executed, a function yields one value. Use parentheses to contain the operand of a function.
Even though their syntax is similar, functions differ from instructions in that functions can be used only in expressions. Instructions cannot be used in expressions.</td><td>function(tag1)</td></tr>
<tr><td>Instruction
(see page 678)</td><td colspan="2">An instruction is a standalone statement.
An instruction uses parenthesis to contain its operands.
Depending on the instruction, there can be zero, one, or multiple operands.
When executed, an instruction yields one or more values that are part of a data structure.
Terminate the instruction with a semi colon(;).
Even though their syntax is similar, instructions differ from functions in that instructions cannot be used in expressions. Functions can be used only in expressions.</td><td>instruction();
instruction(operand);
instruction(operand1, operand2,operand3);</td></tr>
</table>

Term	**Definition**	**Examples**
Construct (see page 679)	A conditional statement used to trigger structured text code (that is, other statements). Terminate the construct with a semi colon (;).	IF...THEN CASE FOR...DO WHILE...DO REPEAT...UNTIL EXIT
Comment (see page 695)	Text that explains or clarifies what a section of structured text does. • Use comments to make it easier to interpret the structured text. • Comments do not affect the execution of the structured text. • Comments can appear anywhere in structured text.	//comment (*start of comment . . . end of comment*) /*start of comment . . . end of comment*/

Assignments

Use an assignment to change the value stored within a tag. An assignment has this syntax:

tag := expression ;

where:

<table>
<tr><th>Component</th><th colspan="2">Description</th></tr>
<tr><td>Tag</td><td colspan="2">Represents the tag that is getting the new value
The tag must be a BOOL, SINT, INT, DINT, or REAL</td></tr>
<tr><td>:=</td><td colspan="2">Is the assignment symbol</td></tr>
<tr><td rowspan="4">Expression</td><td colspan="2">Represents the new value to assign to the tag</td></tr>
<tr><td>If tag is this data type</td><td>Use this type of expression</td></tr>
<tr><td>BOOL</td><td>BOOL expression</td></tr>
<tr><td>SINT
INT
DINT
REAL</td><td>Numeric expression</td></tr>
<tr><td>;</td><td colspan="2">Ends the assignment</td></tr>
</table>

The tag retains the assigned value until another assignment changes the value.

The expression can be simple, such as an immediate value or another tag name, or the expression can be complex and include several operators and/or functions. See Expressions on page 671.

Specify a Non-retentive Assignment

The non-retentive assignment is different from the regular assignment described above in that the tag in a non-retentive assignment is reset to zero each time the controller:

- enters the RUN mode.
- leaves the step of an SFC if you configure the SFC for *Automatic reset*. (This applies only if you embed the assignment in the action of the step or use the action to call a structured text routine via a JSR instruction.)

A non-retentive assignment has this syntax:

tag [:=] expression ;

where:

<table>
<tr><th>Component</th><th colspan="2">Description</th></tr>
<tr><td>Tag</td><td colspan="2">Represents the tag that is getting the new value
The tag must be a BOOL, SINT, INT, DINT, or REAL</td></tr>
<tr><td>[:=]</td><td colspan="2">Is the non-retentive assignment symbol</td></tr>
<tr><td rowspan="4">Expression</td><td colspan="2">Represents the new value to assign to the tag</td></tr>
<tr><td>If tag is this data type</td><td>Use this type of expression</td></tr>
<tr><td>BOOL</td><td>BOOL expression</td></tr>
<tr><td>SINT
INT
DINT
REAL</td><td>Numeric expression</td></tr>
<tr><td>;</td><td colspan="2">Ends the assignment</td></tr>
</table>

Assign an ASCII Character to a String

Use the assignment operator to assign an ASCII character to an element of the DATA member of a string tag. To assign a character, specify the value of the character or specify the tag name, DATA member, and element of the character.

The table shows some examples.

This is okay	This is not okay
`string1.DATA[0]:= 65;`	`string1.DATA[0] := A;`
`string1.DATA[0]:= string2.DATA[0];`	`string1 := string2;`

To add or insert a string of characters to a string tag, use either of these ASCII string instructions.

To	Use this instruction
Add characters to the end of a string	CONCAT
Insert characters into a string	INSERT

Expressions

An expression is a tag name, equation, or comparison. To write an expression, use any of the following:

- Tag name that stores the value (variable)
- Number that you enter directly into the expression (immediate value)
- Functions, such as: ABS, TRUNC
- Operators, such as: +, -, <, >, And, Or

As you write expressions, follow these general rules:

- Use any combination of upper-case and lower-case letter. For example, these three variations of 'AND' are acceptable: AND, And, and.
- For more complex requirements, use parentheses to group expressions within expressions. This makes the whole expression easier to read and ensures that the expression executes in the desired sequence. See Determine the Order of Execution on page 677.

In structured text, you use two types of expressions.

BOOL expression: An expression that produces either the BOOL value of 1 (true) or 0 (false).

- A bool expression uses bool tags, relational operators, and logical operators to compare values or check if conditions are true or false. For example, tag1>65.
- A simple bool expression can be a single BOOL tag.
- Typically, you use bool expressions to condition the execution of other logic.

Numeric expression: An expression that calculates an integer or floating-point value.

- A numeric expression uses arithmetic operators, arithmetic functions, and bitwise operators. For example, tag1+5.
- Often, you nest a numeric expression within a bool expression. For example, (tag1+5)>65.

Use the table to choose operators for your expressions.

If you want to	Then
Calculate an arithmetic value	Use Arithmetic Operators and Functions on page 673.
Compare two values or strings	Use Relational Operators on page 674.
Check if conditions are true or false	Use Logical Operators on page 676.
Compare the bits within values	Use Bitwise Operators on page 677.

Use Arithmetic Operators and Functions

You can combine multiple operators and functions in arithmetic expressions.

Arithmetic operators calculate new values.

To	Use this operator	Optimal data type
Add	+	DINT, REAL
Subtract/negate	-	DINT, REAL
Multiply	*	DINT, REAL
Exponent (x to the power of y)	**	DINT, REAL
Divide	/	DINT, REAL
Modulo-divide	MOD	DINT, REAL

Arithmetic functions perform math operations. Specify a constant, a non-boolean tag, or an expression for the function.

For	Use this function	Optimal data type
Absolute value	ABS (*numeric_expression*)	DINT, REAL
Arc cosine	ACOS (*numeric_expression*)	REAL
Arc sine	ASIN (*numeric_expression*)	REAL
Arc tangent	ATAN (*numeric_expression*)	REAL
Cosine	COS (*numeric_expression*)	REAL
Radians to degrees	DEG (*numeric_expression)*	DINT, REAL
Natural log	LN (*numeric_expression*)	REAL
Log base 10	LOG (*numeric_expression*)	REAL
Degrees to radians	RAD (*numeric_expression*)	DINT, REAL
Sine	SIN (*numeric_expression*)	REAL
Square root	SQRT (*numeric_expression*)	DINT, REAL
Tangent	TAN (*numeric_expression*)	REAL
Truncate	TRUNC (*numeric_expression*)	DINT, REAL

The table shows some examples.

Use this format	Example	
	For this situation	**Write**
value1 operator value2	If gain_4 and gain_4_adj are DINT tags and your specification says: 'Add 15 to gain_4 and store the result in gain_4_adj'"	gain_4_adj := gain_4+15;
operator value1	If alarm and high_alarm are DINT tags and your specification says: 'Negate high_alarm and store the result in alarm.'	alarm:= -high_alarm;
function(numeric_expression)	If overtravel and overtravel_POS are DINT tags and your specification says: 'Calculate the absolute value of overtravel and store the result in overtravel_POS.'	overtravel_POS := ABS(overtravel);
value1 operator (function((value2+value3)/2)	If adjustment and position are DINT tags and sensor1 and sensor2 are REAL tags and your specification says: 'Find the absolute value of the average of sensor1 and sensor2, add the adjustment, and store the result in position.'	position := adjustment + ABS((sensor1 + sensor2)/2);

Use Relational Operators

Relational operators compare two values or strings to provide a true or false result. The result of a relational operation is a BOOL value.

If the comparison is	The result is
True	1
False	0

Use these relational operators.

For this comparison	Use this operator	Optimal data type
Equal	=	DINT, REAL, string
Less than	<	DINT, REAL, string
Less than or equal	<=	DINT, REAL, string
Greater than	>	DINT, REAL, string
Greater than or equal	>=	DINT, REAL, string
Not equal	<>	DINT, REAL, string

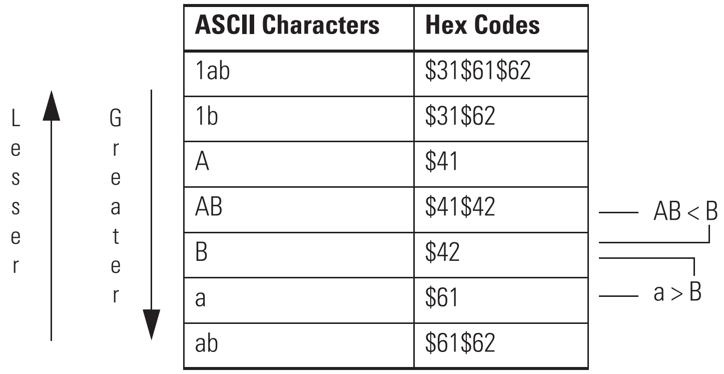
The table shows some examples.

Use this format	Example	
	For this situation	**Write**
`value1 operator value2`	If temp is a DINT tag and your specification says: 'If temp is less than 100· then…'	IF temp<100 THEN...
`stringtag1 operator stringtag2`	If bar_code and dest are string tags and your specification says: 'If bar_code equals dest then…'	IF bar_code=dest THEN...
`char1 operator char2` To enter an ASCII character directly into the expression, enter the decimal value of the character.	If bar_code is a string tag and your specification says: 'If bar_code.DATA[0] equals 'A' then…'	IF bar_code.DATA[0]=65 THEN...
`bool_tag := bool_expressions`	If count and length are DINT tags, done is a BOOL tag, and your specification says: 'If count is greater than or equal to length, you are done counting.'	Done := (count >= length);

How Strings are Evaluated

The hexadecimal values of the ASCII characters determine if one string is less than or greater than another string.

- When the two strings are sorted as in a telephone directory, the order of the strings determines which one is greater.



ASCII Characters	Hex Codes
1ab	$31$61$62
1b	$31$62
A	$41
AB	$41$42
B	$42
a	$61
ab	$61$62

- Strings are equal if their characters match.
- Characters are case sensitive. Uppercase 'A' ($41) is **not** equal to lowercase 'a' ($61).

Use Logical Operators

Logical operators let you check if multiple conditions are true or false. The result of a logical operation is a BOOL value.

If the comparison is	**The result is**
True	1
False	0

Use these logical operators.

For	**Use this operator**	**Data type**
Logical AND	&, AND	BOOL
Logical OR	OR	BOOL
Logical exclusive OR	XOR	BOOL
Logical complement	NOT	BOOL

The table shows some examples.

Use this format	**Example**	
	For this situation	**Write**
BOOLtag	If photoeye is a BOOL tag and your specification says: 'If photoeye is on then…'	IF photoeye THEN...
NOT *BOOLtag*	If photoeye is a BOOL tag and your specification says: 'If photoeye is off then…'	IF NOT photoeye THEN...
expression1 & *expression2*	If photoeye is a BOOL tag, temp is a DINT tag, and your specification says: 'If photoeye is on and temp is less than 100× then…'	IF photoeye & (temp<100) THEN...
expression1 OR *expression2*	If photoeye is a BOOL tag, temp is a DINT tag, and your specification says: 'If photoeye is on or temp is less than 100× then…'	IF photoeye OR (temp<100) THEN...
expression1 XOR *expression2*	If photoeye1 and photoeye2 are BOOL tags and your specification says: 'If: • photoeye1 is on while photoeye 2 is off • photoeye1 is off while photoeye 2 is on then…'	IF photoeye1 XOR photoeye2 THEN...
BOOLtag := *expression1* & *expression2*	If photoeye1 and photoeye2 are BOOL tags, open is a BOOL tag, and your specification says: 'If photoeye1 and photoeye2 are both on, set open to true'.	Open := photoeye1 & photoeye2;

Use Bitwise Operators

Bitwise operators manipulate the bits within a value based on two values.

For	Use this operator	Optimal data type
Bitwise AND	&, AND	DINT
Bitwise OR	OR	DINT
Bitwise exclusive OR	XOR	DINT
Bitwise complement	NOT	DINT

This is an example.

Use this format	Example	
	For this situation	**Write**
value1 operator value2	If input1, input2, and result1 are DINT tags and your specification says: 'Calculate the bitwise result of input1 and input2. Store the result in result1.'	result1 := input1 AND input2;

Determine the Order of Execution

The operations you write into an expression are performed in a prescribed order, not necessarily from left to right.

- Operations of equal order are performed from left to right.
- If an expression contains multiple operators or functions, group the conditions in parenthesis ( ) . This ensures the correct order of execution and makes it easier to read the expression.

Order	Operation
1.	( )
2.	function (…)
3.	**
4.	–(negate)
5.	NOT
6.	*, /, MOD
7.	+, - (subtract)
8.	<, <=, >, >=
9.	=, <>
10.	&, AND
11.	XOR
12.	OR

Instructions

Structured text statements can also be instructions. See the Locator Table on page 15 for a list of the instructions available in structured text. A structured text instruction executes each time it is scanned. A structured text instruction within a construct executes every time the conditions of the construct are true. If the conditions of the construct are false, the statements within the construct are not scanned. There is no rung-condition or state transition that triggers execution.

This differs from function block instructions that use EnableIn to trigger execution. Structured text instructions execute as if EnableIn is always set.

This also differs from relay ladder instructions that use rung-condition-in to trigger execution. Some relay ladder instructions only execute when rung-condition-in toggles from false to true. These are transitional relay ladder instructions. In structured text, instructions will execute each time they are scanned unless you pre-condition the execution of the structured text instruction.

For example, the ABL instruction is a transitional instruction in relay ladder. In this example, the ABL instruction only executes on a scan when *tag_xic* transitions from cleared to set. The ABL instruction does not execute when *tag_xic* stays set or when *tag_xic* is cleared.

In structured text, if you write this example as:

```
IF tag_xic THEN ABL(0,serial_control);

END_IF;
```

The ABL instruction will execute every scan that *tag_xic* is set, not just when *tag_xic* transitions from cleared to set.

If you want the ABL instruction to execute only when *tag_xic* transitions from cleared to set, you have to condition the structured text instruction. Use a one shot to trigger execution.

```
osri_1.InputBit := tag_xic;
OSRI(osri_1);

IF (osri_1.OutputBit) THEN
     ABL(0,serial_control);
END_IF;
```

Constructs

Constructs can be programmed alone or nested within other constructs.

If you want to	Use this construct	Available in these languages	Page
Do something if or when specific conditions occur	IF...THEN	Structured text	680
Select what to do based on a numerical value	CASE...OF	Structured text	683
Do something a specific number of times before doing anything else	FOR...DO	Structured text	686
Keep doing something as long as certain conditions are true	WHILE...DO	Structured text	689
Keep doing something until a condition is true	REPEAT...UNTIL	Structured text	692

Some Key Words are Reserved

These constructs are not available:

- GOTO
- REPEAT

RSLogix 5000 software will not let you use them as tag names or constructs.

IF...THEN

Use IF…THEN to do something if or when specific conditions occur.

Operands:

```
IF bool_expression THEN
    <statement>;
END_IF;
```

Structured Text

Operand	**Type**	**Format**	**Enter**
Bool_ expression	BOOL	Tag expression	BOOL tag or expression that evaluates to a BOOL value (BOOL expression)

Description:

The syntax is described in the table.

```
           IF bool_expression1 THEN
                 <statement >;     ◄────────  Statements to execute when
                          .                   bool_expression1 is true
                          .
                          .
optional { ELSIF bool_expression2 THEN
                 <statement>;      ◄────────  Statements to execute when
                          .                   bool_expression2 is true
                          .
                          .
optional { ELSE
                 <statement>;      ◄────────  Statements to execute when
                          .                   both expressions are false
                          .
                          .
           END_IF;
```

To use ELSIF or ELSE, follow these guidelines.

1. To select from several possible groups of statements, add one or more ELSIF statements.
    - Each ELSIF represents an alternative path.
    - Specify as many ELSIF paths as you need.
    - The controller executes the first true IF or ELSIF and skips the rest of the ELSIFs and the ELSE.
2. To do something when all of the IF or ELSIF conditions are false, add an ELSE statement.

The table summarizes different combinations of IF, THEN, ELSIF, and ELSE.

If you want to	And	Use this construct
Do something if or when conditions are true	Do nothing if conditions are false	IF…THEN
	Do something else if conditions are false	IF…THEN…ELSE
Choose from alternative statements (or groups of statements) based on input conditions	Do nothing if conditions are false	IF…THEN…ELSIF
	Assign default statements if all conditions are false	IF…THEN…ELSIF…ELSE

Arithmetic Status Flags

Not affected

Fault Conditions:

None

Example 1: IF…THEN

If you want this	Enter this structured text
IF rejects > 3 then	`IF rejects > 3 THEN`
conveyor = off (0)	`conveyor := 0;`
alarm = on (1)	`alarm := 1;`
	`END_IF;`

Example 2: IF…THEN…ELSE

If you want this	Enter this structured text
If conveyor direction contact = forward (1) then	`IF conveyor_direction THEN`
light = off	`light := 0;`
Otherwise light = on	`ELSE`
	`light [:=] 1;`
	`END_IF;`

The [:=] tells the controller to clear *light* whenever the controller:

- enters the RUN mode.
- leaves the step of an SFC if you configure the SFC for *Automatic reset*. (This applies only if you embed the assignment in the action of the step or use the action to call a structured text routine via a JSR instruction.)

Example 3: IF…THEN…ELSIF

If you want this		**Enter this structured text**	
If sugar low limit switch = low (on) and sugar high limit switch = not high (on) then		`IF Sugar.Low & Sugar.High THEN`	
	inlet valve = open (on)		`Sugar.Inlet [:=] 1;`
Until sugar high limit switch = high (off)		`ELSIF NOT(Sugar.High) THEN`	
			`Sugar.Inlet := 0;`
		`END_IF;`	

The [:=] tells the controller to clear *Sugar.Inlet* whenever the controller:

- enters the RUN mode.
- leaves the step of an SFC if you configure the SFC for *Automatic reset*. (This applies only if you embed the assignment in the action of the step or use the action to call a structured text routine via a JSR instruction.)

Example 4: IF…THEN…ELSIF…ELSE

If you want this		**Enter this structured text**	
If tank temperature > 100		`IF tank.temp > 200 THEN`	
	then pump = slow		`pump.fast :=1; pump.slow :=0; pump.off :=0;`
If tank temperature > 200		`ELSIF tank.temp > 100 THEN`	
	then pump = fast		`pump.fast :=0; pump.slow :=1; pump.off :=0;`
Otherwise pump = off		`ELSE`	
			`pump.fast :=0; pump.slow :=0; pump.off :=1;`
		`END_IF;`	

CASE...OF

Use CASE to select what to do based on a numerical value.

Operands:

```
CASE numeric_expression OF
     selector1: statement;
     selectorN: statement;
ELSE
     statement;
END_CASE;
```

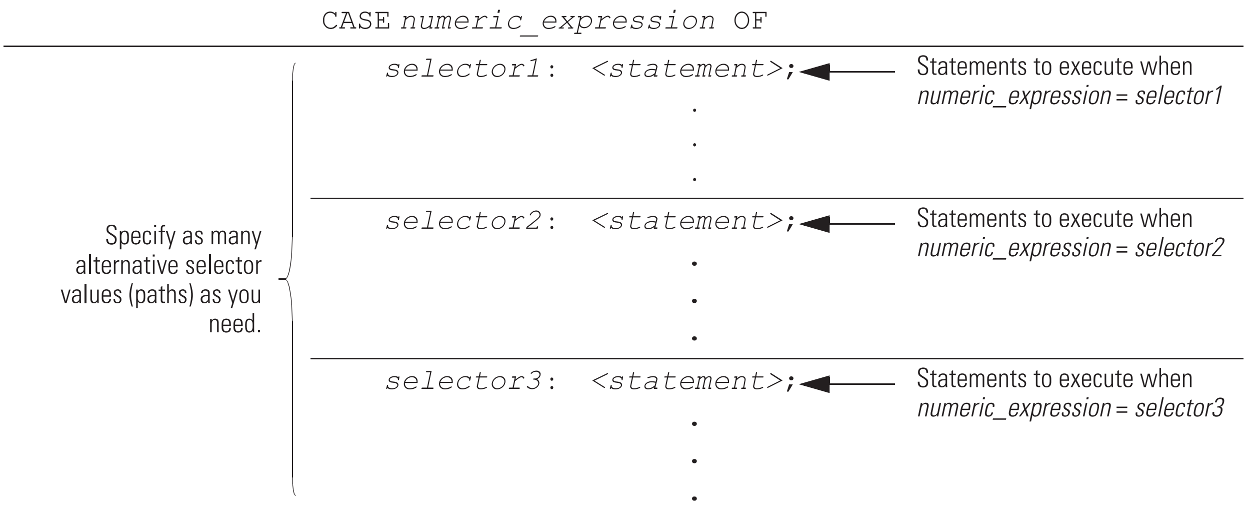
Structured Text

Operand	Type	Format	Enter
Numeric_ expression	SINT INT DINT REAL	Tag expression	Tag or expression that evaluates to a number (numeric expression)
Selector	SINT INT DINT REAL	Immediate	Same type as numeric_expression

IMPORTANT If you use REAL values, use a range of values for a selector because a REAL value is more likely to be within a range of values than an exact match of one, specific value.

Description:

The syntax is described in the table.



```
CASE numeric_expression OF
     selector1:  <statement>;
                 .
                 .
                 .
     selector2:  <statement>;
                 .
                 .
                 .
     selector3:  <statement>;
                 .
                 .
                 .
```

Specify as many alternative selector values (paths) as you need.

Statements to execute when *numeric_expression = selector1*

Statements to execute when *numeric_expression = selector2*

Statements to execute when *numeric_expression = selector3* optional

```
ELSE
                    <statement>;
                              .
                              .
                              .
END_CASE;
```

Statements to execute when *numeric_expression* ≠ any selector

See the table on page 685 for valid selector values.

These are the syntax for entering the selector values.

When selector is	Enter
One value	*value*: *statement*
Multiple, distinct values	*value1*, *value2*, *valueN* : *<statement>* Use a comma (,) to separate each value.
A range of values	*value1*..*valueN* : *<statement>* Use two periods (..) to identify the range.
Distinct values plus a range of values	*valuea, valueb, value1*..*valueN* : *<statement>*

The CASE construct is similar to a switch statement in the C or C++ programming languages. However, with the CASE construct the controller executes **only** the statements that are associated with the **first matching** selector value. Execution **always breaks after the statements of that selector** and goes to the END_CASE statement.

Arithmetic Status Flags: Not affected

Fault Conditions: None

Example

If you want this	Enter this structured text
If recipe number = 1 then	CASE recipe_number OF
Ingredient A outlet 1 = open (1) Ingredient B outlet 4 = open (1)	1: Ingredient_A.Outlet_1 :=1; Ingredient_B.Outlet_4 :=1;
If recipe number = 2 or 3 then Ingredient A outlet 4 = open (1) Ingredient B outlet 2 = open (1)	2,3: Ingredient_A.Outlet_4 :=1; Ingredient_B.Outlet_2 :=1;
If recipe number = 4, 5, 6, or 7 then Ingredient A outlet 4 = open (1) Ingredient B outlet 2 = open (1)	4…7: Ingredient_A.Outlet_4 :=1; Ingredient_B.Outlet_2 :=1;
If recipe number = 8, 11, 12, or 13 then Ingredient A outlet 1 = open (1) Ingredient B outlet 4 = open (1)	8,11…13 Ingredient_A.Outlet_1 :=1; Ingredient_B.Outlet_4 :=1;
Otherwise all outlets = closed (0)	ELSE
	Ingredient_A.Outlet_1 [:=]0; Ingredient_A.Outlet_4 [:=]0; Ingredient_B.Outlet_2 [:=]0; Ingredient_B.Outlet_4 [:=]0;
	END_CASE;

The [:=] tells the controller to also clear the outlet tags whenever the controller:

- enters the RUN mode.
- leaves the step of an SFC if you configure the SFC for *Automatic reset*. (This applies only if you embed the assignment in the action of the step or use the action to call a structured text routine via a JSR instruction.)

FOR…DO

Use the FOR…DO loop to do something a specific number of times before doing anything else.

Operands:

```
FOR count:= initial_value TO
final_value BY increment DO

    <statement>;

END_FOR;
```

Structured Text

Operand	Type	Format	Description
count	SINT INT DINT	Tag	Tag to store count position as the FOR…DO executes
initial_ value	SINT INT DINT	Tag expression Immediate	Must evaluate to a number Specifies initial value for count
final_ value	SINT INT DINT	Tag expression Immediate	Specifies final value for count, which determines when to exit the loop
increment	SINT INT DINT	Tag expression Immediate	(*Optional*) amount to increment count each time through the loop If you don't specify an increment, the count increments by 1.

IMPORTANT

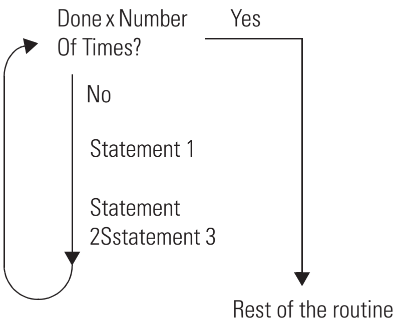
Make sure that you **do not** iterate within the loop too many times in a single scan.

- The controller does not execute any other statements in the routine until it completes the loop.
- If the time that it takes to complete the loop is greater than the watchdog timer for the task, a major fault occurs.
- Consider using a different construct, such as IF...THEN.

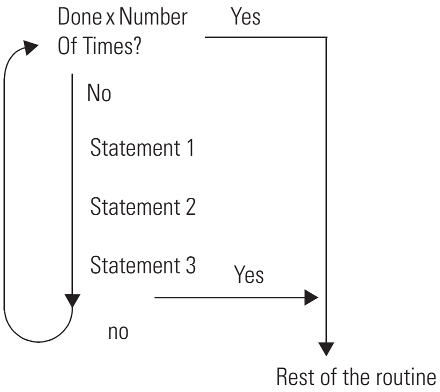
Description: The syntax is described in the table.

	FOR *count* := *initial_value*	
	TO *final_value*	
Optional {	BY *increment*	If you don't specify an increment, the loop increments by 1.
	DO	
	<statement>;	
Optional	IF *bool_expression* THEN	
	EXIT;	If there are conditions when you want to exit the loop early, use other statements, such as an IF...THEN construct, to condition an EXIT statement.
	END_IF;	
	END_FOR;	

The following diagrams show how a FOR...DO loop executes and how an EXIT statement leaves the loop early.



The FOR…DO loop executes a specific number of times.



To stop the loop before the count reaches the last value, use an EXIT statement.

Arithmetic Status Flags: Not affected

Fault Conditions:

A major fault will occur if	Fault type	Fault code
The construct loops too long.	6	1

Example 1:

If you want this	Enter this structured text
Clear bits 0…31 in an array of BOOLs: 1. Initialize the *subscript* tag to 0. 2. Clear *array[subscript]* . For example, when *subscript* = 5, clear *array[5].* 3. Add 1 to *subscript*. 4. If *subscript* is ≤to 31, repeat 2 and 3. Otherwise, stop.	See the code below

```
For subscript:=0 to 31 by 1 do
   array[subscript] := 0;
End_for;
```

Example 2:

If you want this	Enter this structured text
A user-defined data type (structure) stores the following information about an item in your inventory: • Barcode ID of the item (string data type) • Quantity in stock of the item (DINT data type) An array of the above structure contains an element for each different item in your inventory. You want to search the array for a specific product (use its bar code) and determine the quantity that is in stock. 1. Get the size (number of items) of the Inventory array and store the result in Inventory_Items (DINT tag). 2. Initialize the position tag to 0. 3. If Barcode matches the ID of an item in the array, then: a. Set the Quantity tag = Inventory[position].Qty. This produces the quantity in stock of the item. b. Stop. Barcode is a string tag that stores the bar code of the item for which you are searching. For example, when position = 5, compare Barcode to Inventory[5].ID. 4. Add 1 to position. 5. If position is ≤to (Inventory_Items -1), repeat 3 and 4. Since element numbers start at 0, the last element is 1 less than the number of elements in the array. Otherwise, stop.	See the code below

```
SIZE(Inventory,0,Inventory_Items);
For position:=0 to Inventory_Items - 1 do
    If Barcode = Inventory[position].ID then
         Quantity := Inventory[position].Qty;
         Exit;
    End_if;
End_for;
```

WHILE…DO

Use the WHILE…DO loop to keep doing something as long as certain conditions are true.

Operands:

```
WHILE bool_expression DO
    <statement>;
END_WHILE;
```

Structured Text

Operand	Type	Format	Enter
Bool_ expression	BOOL	Tag expression	BOOL tag or expression that evaluates to a BOOL value

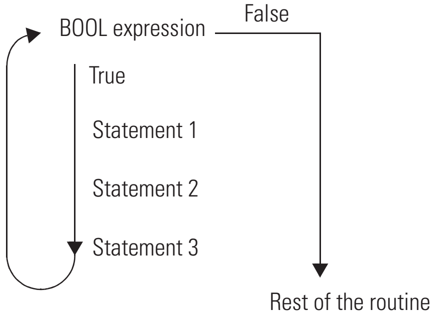
IMPORTANT Make sure that you do not iterate within the loop too many times in a single scan.

- The controller does not execute any other statements in the routine until it completes the loop.
- If the time that it takes to complete the loop is greater than the watchdog timer for the task, a major fault occurs.
- Consider using a different construct, such as IF...THEN.

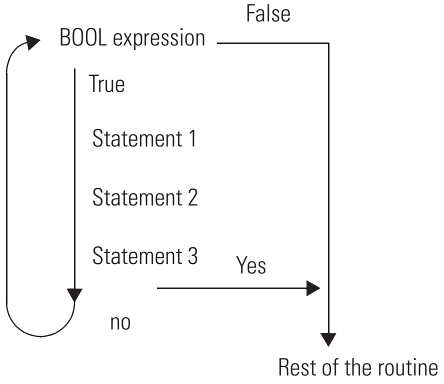
Description: The syntax is described in the table.

	`WHILE bool_expression1 DO`	
	`<statement>;`	Statements to execute while *bool_expression1* is true
Optional	`IF bool_expression2 THEN`	
	`EXIT;`	If there are conditions when you want to exit the loop early, use other statements, such as an IF...THEN construct, to condition an EXIT statement.
	`END_IF;`	
	`END_WHILE;`	

The following diagrams show how a WHILE...DO loop executes and how an EXIT statement leaves the loop early.



While the `bool_expression` is true, the controller executes only the statements within the WHILE…DO loop.



To stop the loop before the conditions are true, use an EXIT statement.

Arithmetic Status Flags: Not affected

Fault Conditions:

A major fault will occur if	Fault type	Fault code
The construct loops too long	6	1

Example 1:

If you want this	Enter this structured text
The WHILE...DO loop evaluates its conditions first. If the conditions are true, the controller then executes the statements within the loop. This differs from the REPEAT...UNTIL loop because the REPEAT...UNTIL loop executes the statements in the construct and then determines if the conditions are true before executing the statements again. The statements in a REPEAT...UNTIL loop are always executed at least once. The statements in a WHILE...DO loop might never be executed.	pos := 0;
	While ((pos <= 100) & structarray[pos].value <> targetvalue)) do
	pos := pos + 2;
	String_tag.DATA[pos] := SINT_array[pos];
	end_while;

Example 2:

If you want this	Enter this structured text
Move ASCII characters from a SINT array into a string tag. (In a SINT array, each element holds one character.) Stop when you reach the carriage return. 1. Initialize Element_number to 0. 2. Count the number of elements in SINT_array (array that contains the ASCII characters) and store the result in SINT_array_size (DINT tag). 3. If the character at SINT_array[element_number] = 13 (decimal value of the carriage return), then stop. 4. Set String_tag[element_number] = the character at SINT_array[element_number]. 5. Add 1 to element_number. This lets the controller check the next character in SINT_array. 6. Set the Length member of String_tag = element_number. (This records the number of characters in String_tag so far.) 7. If element_number = SINT_array_size, then stop. (You are at the end of the array and it does not contain a carriage return.) 8. Go to 3.	

```
element_number := 0;
SIZE(SINT_array, 0, SINT_array_size);
While SINT_array[element_number] <> 13 do
        String_tag.DATA[element_number] :=
        SINT_array[element_number];
        element_number := element_number + 1;
        String_tag.LEN := element_number;
        If element_number = SINT_array_size then
                exit;
        end_if;
end_while;
```

REPEAT…UNTIL

Use the REPEAT…UNTIL loop to keep doing something until conditions are true.

Operands:

```
REPEAT
    <statement>;
UNTIL bool_expression
END_REPEAT;
```

Structured Text

Operand	Type	Format	Enter
bool_ expression	BOOL	Tag expression	BOOL tag or expression that evaluates to a BOOL value (BOOL expression)

IMPORTANT Make sure that you **do not** iterate within the loop too many times in a single scan.

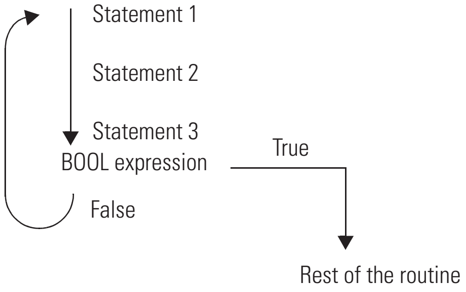
- The controller does not execute any other statements in the routine until it completes the loop.
- If the time that it takes to complete the loop is greater than the watchdog timer for the task, a major fault occurs.
- Consider using a different construct, such as IF...THEN.

Description:

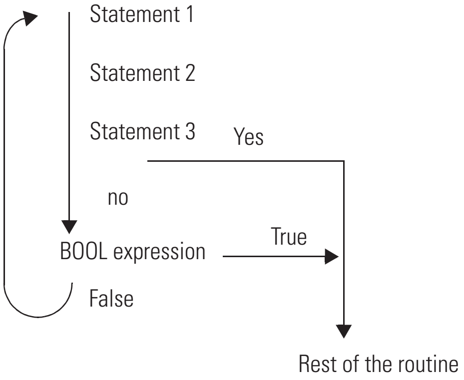
The syntax is described in the table.

	`REPEAT`	
	`<statement>;`	statements to execute while *bool_expression1* is false
Optional	`IF bool_expression2 THEN`	
	`EXIT;`	If there are conditions when you want to exit the loop early, use other statements, such as an IF...THEN construct, to condition an EXIT statement.
	`END_IF;`	
	`UNTIL bool_expression1`	
	`END_REPEAT;`	

The following diagrams show how a REPEAT...UNTIL loop executes and how an EXIT statement leaves the loop early.



While the `bool_expression` is false, the controller executes only the statements within the REPEAT…UNTIL loop.



To stop the loop before the conditions are false, use an EXIT statement.

Arithmetic Status Flags:

Not affected

Fault Conditions:

A major fault will occur if	Fault type	Fault code
The construct loops too long	6	1

Example 1:

If you want this	Enter this structured text
The REPEAT...UNTIL loop executes the statements in the construct and then determines if the conditions are true before executing the statements again. This differs from the WHILE...DO loop because the WHILE...DO The WHILE...DO loop evaluates its conditions first. If the conditions are true, the controller then executes the statements within the loop. The statements in a REPEAT...UNTIL loop are always executed at least once. The statements in a WHILE...DO loop might never be executed.	pos := -1;
	REPEAT
	pos := pos + 2;
	UNTIL ((pos = 101) OR (structarray[pos].value = targetvalue))
	end_repeat;

Example 2:

If you want this	Enter this structured text
Move ASCII characters from a SINT array into a string tag. (In a SINT array, each element holds one character.) Stop when you reach the carriage return. 1. Initialize Element_number to 0. 2. Count the number of elements in SINT_array (array that contains the ASCII characters) and store the result in SINT_array_size (DINT tag). 3. Set String_tag[element_number] = the character at SINT_array[element_number]. 4. Add 1 to element_number. This lets the controller check the next character in SINT_array. 5. Set the Length member of String_tag = element_number. (This records the number of characters in String_tag so far.) 6. If element_number = SINT_array_size, then stop. (You are at the end of the array and it does not contain a carriage return.) 7. If the character at SINT_array[element_number] = 13 (decimal value of the carriage return), then stop. Otherwise, go to step 3.	

```
element_number := 0;
SIZE(SINT_array, 0, SINT_array_size);
Repeat
        String_tag.DATA[element_number] :=
        SINT_array[element_number];
        element_number := element_number + 1;
        String_tag.LEN := element_number;
        If element_number = SINT_array_size then
                exit;
        end_if;
Until SINT_array[element_number] = 13
end_repeat;
```

Comments

To make your structured text easier to interpret, add comments to it.

- Comments let you use plain language to describe how your structured text works.
- Comments do not affect the execution of the structured text.

The table describes how to add comments to your structured text.

To add a comment	Use one of these formats
On a single line	//*comment* (**comment**) /**comment**/
At the end of a line of structured text	
Within a line of structured text	(**comment**) /**comment**/
That spans more than one line	(**start of comment . . . end of comment**) /**start of comment . . . end of comment**/

The table shows some examples.

Format	**Example**

`//`*`comment`*	**At the beginning of a line**
	`//Check conveyor belt direction`
	`IF conveyor_direction THEN...`
	At the end of a line
	`ELSE //If conveyor isn’t moving, set alarm light`
	`light := 1;`
	`END_IF;`

`(*`*`comment`*`*)`	`Sugar.Inlet[:=]1;(*open the inlet*)`
	`IF Sugar.Low (*low level LS*)& Sugar.High (*high level LS*)THEN...`
	`(*Controls the speed of the recirculation pump. The speed depends on the temperature in the tank.*)`
	`IF tank.temp > 200 THEN...`

`/*`*`comment`*`*/`	`Sugar.Inlet:=0;/*close the inlet*/`
	`IF bar_code=65 /*A*/ THEN...`
	`/*Gets the number of elements in the Inventory array and stores the value in the Inventory_Items tag*/`
	`SIZE(Inventory,0,Inventory_Items);`

Notes:

Rockwell Automation Support

Rockwell Automation provides technical information on the Web to assist you in using its products. At http://www.rockwellautomation.com/support/, you can find technical manuals, a knowledge base of FAQs, technical and application notes, sample code and links to software service packs, and a MySupport feature that you can customize to make the best use of these tools.

For an additional level of technical phone support for installation, configuration, and troubleshooting, we offer TechConnect support programs. For more information, contact your local distributor or Rockwell Automation representative, or visit http://www.rockwellautomation.com/support/.

Installation Assistance

If you experience an anomoly within the first 24 hours of installation, review the information that is contained in this manual. You can contact Customer Support for initial help in getting your product up and running.

United States or Canada	1.440.646.3434
Outside United States or Canada	Use the Worldwide Locator at http://www.rockwellautomation.com/support/americas/phone_en.html, or contact your local Rockwell Automation representative.

New Product Satisfaction Return

Rockwell Automation tests all of its products to ensure that they are fully operational when shipped from the manufacturing facility. However, if your product is not functioning and needs to be returned, follow these procedures.

United States	Contact your distributor. You must provide a Customer Support case number (call the phone number above to obtain one) to your distributor to complete the return process.
Outside United States	Please contact your local Rockwell Automation representative for the return procedure.

Documentation Feedback

Your comments will help us serve your documentation needs better. If you have any suggestions on how to improve this document, complete this form, publication RA-DU002, available at http://www.rockwellautomation.com/literature/.

www.rockwellautomation.com

Power, Control and Information Solutions Headquarters

Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
Europe/Middle East/Africa: Rockwell Automation, Vorstlaan/Boulevard du Souverain 36, 1170 Brussels, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia Pacific: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Publication 1756-RM003L-EN-P - October 2009
Supersedes Publication 1756-RM003K-EN-P - July 2008

ANEXO L. SERCOS and Analog Motion Configuration and Startup.

SERCOS and Analog Motion Configuration and Startup

Allen-Bradley

Catalog Numbers 1756-HYD02, 1756-M02AE, 1756-M02AS, 1756-M03SE, 1756-M08SE, 1756-M16SE, 1768-M04SE, 2094-SE02F-M00-S0, 2094-SE02F-M00-S1

User Manual

Allen-Bradley • Rockwell Software

Rockwell Automation

Important User Information

Solid state equipment has operational characteristics differing from those of electromechanical equipment. Safety Guidelines for the Application, Installation and Maintenance of Solid State Controls (publication SGI-1.1 available from your local Rockwell Automation sales office or online at http://www.rockwellautomation.com/literature/) describes some important differences between solid state equipment and hard-wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

In no event will Rockwell Automation, Inc. be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.

The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, Rockwell Automation, Inc. cannot assume responsibility or liability for actual use based on the examples and diagrams.

No patent liability is assumed by Rockwell Automation, Inc. with respect to use of information, circuits, equipment, or software described in this manual.

Reproduction of the contents of this manual, in whole or in part, without written permission of Rockwell Automation, Inc., is prohibited.

Throughout this manual, when necessary, we use notes to make you aware of safety considerations.

WARNING	Identifies information about practices or circumstances that can cause an explosion in a hazardous environment, which may lead to personal injury or death, property damage, or economic loss.
IMPORTANT	Identifies information that is critical for successful application and understanding of the product.
ATTENTION	Identifies information about practices or circumstances that can lead to personal injury or death, property damage, or economic loss. Attentions help you identify a hazard, avoid a hazard, and recognize the consequence
SHOCK HAZARD	Labels may be on or inside the equipment, for example, a drive or motor, to alert people that dangerous voltage may be present.
BURN HAZARD	Labels may be on or inside the equipment, for example, a drive or motor, to alert people that surfaces may reach dangerous temperatures.

Preface

Summary of Changes

Chapter 1

Configure SERCOS Motion

Chapter 2

Configure Analog Motion

Chapter 3

Commission and Tune

Chapter 4

Program

Chapter 5

Home an Axis

Appendix A

Axis Properties

Appendix B

Motion Axis Attributes

Appendix C

Wiring Diagrams

Appendix D

Servo Loop Block Diagrams

Index

Notes:

Preface

Introduction

Use this manual to configure a motion application and to start up your motion solution using Logix5000 motion modules.

Topic	Page
What You Need	11
Configuration and Start-up Scenarios	12
Help for Selecting Drives and Motors	14
Where to Find Sample Projects	14
Additional Resources	15

Before You Begin

This manual is a redesigned manual from publication LOGIX-UM002. A companion manual is available called the Coordinate System User Manual, publication MOTION-UM002.

This manual is designed to give you the quickest and easiest approach to a SERCOS or Analog control solution. If you have any comments or suggestions, please see Documentation Feedback on the back cover of this manual.

What You Need

You will need the following to set up a either a SERCOS or Analog motion system:

SERCOS

- Logix L6*x* or Logix L7*x* controller
- SERCOS interface drive (6000,6200,2000, Ultra3000)
- SERCOS interface module
- Kinetix 6000 drive/actuators pair
- RSLogix 5000 programming software

Analog

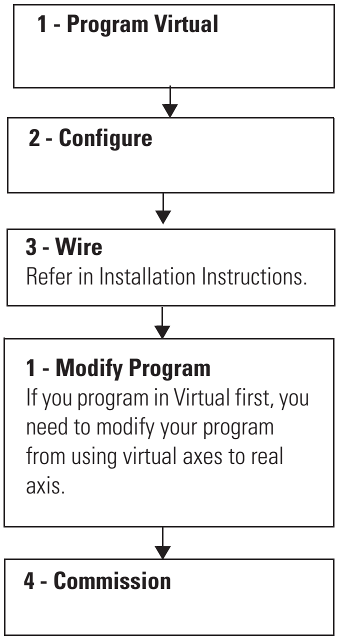
- Logix L6*x* controller
- Analog interface module
- Analog interface drive, Ultra3000
- Kinetix 6000 drive/actuators pair
- RSLogix 5000 programming software

Configuration and Start-up Scenarios

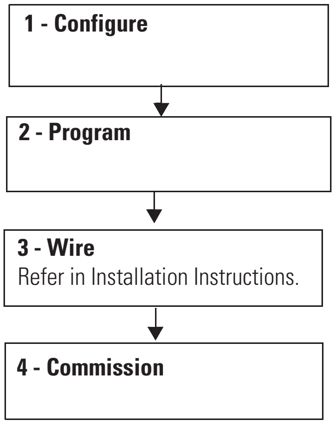
The following are three example scenarios of how you can get a motion solution up and running.

TIP Programming Virtual first is the safest method to begin with because you are separating the motion programming from the hardware.

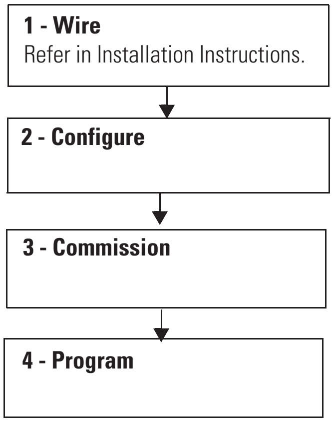
Program First



Configure First



Wire First



Description of the Modules

This table describes the Logix5000 motion modules.

Motion Module	Description
1756-M03SE 1756-M08SE 1756-M16SE 1768-M04SE	Use a SERCOS interface module to connect the controller to SERCOS interface drives. • The SERCOS interface module lets you control digital drives by using high-speed, real time, serial communication. • SERCOS is the IEC 61491 Serial Real-time Communication System protocol over a fiber optic network. • The module uses a fiber optic network for all the wiring between the drives and the module.
2094-SE02F-M00-S0, 2094-SE02F-M00-S1	Kinetix 6200 control modules use SERCOS interface to communicate with the Logix controller and EtherNet/IP to access the safety configuration tool.
1756-M02AE	The 1756-M02AE module is a two-axis servo module for drives/actuators that need a ±10V velocity or torque reference. Use the 1756-M02AE module when your equipment has quadrature encoder feedback. The module also has the following: • Home limit switch inputs • Drive fault inputs • Drive enable outputs • 5V or 24V position registration inputs • 250 μs position and velocity loop updates
1756-HYD02	The 1756-HYD02 module is a two-axis servo module for hydraulic actuators that need a ±10V velocity reference. Use the 1756-HYD02 module when your equipment has magnostrictive linear transducer (LDT) feedback. The module is similar to the 1756-M02AE module with these exceptions: • Feed Forward adjust in addition to single-step Auto Tune. • Gain ratio between extend direction and retract direction to accommodate hydraulic cylinder dynamics. • Intelligent transducer noise detection filtering in hardware and firmware replaces programmable IIR filtering.
1756-M02AS	The 1756-M02AS module is a two-axis servo module for drives/actuators that need a ±10V velocity or torque reference input. Use the 1756-M02AS module when your equipment has Serial Synchronous Input (SSI) position feedback. The module is similar to the 1756-M02AE module with these exceptions: • Gain ratio between extend direction and retract direction to accommodate hydraulic cylinder dynamics. • Intelligent transducer noise detection filtering in hardware and firmware replaces programmable IIR filtering. • SSI interface consisting of Differential Clock output and Data return signals replaces the differential encoder interface.

Help for Selecting Drives and Motors

The Motion Analyzer utility helps you select the appropriate Rockwell Automation drives and motors based upon your load characteristics and typical motion application cycles.

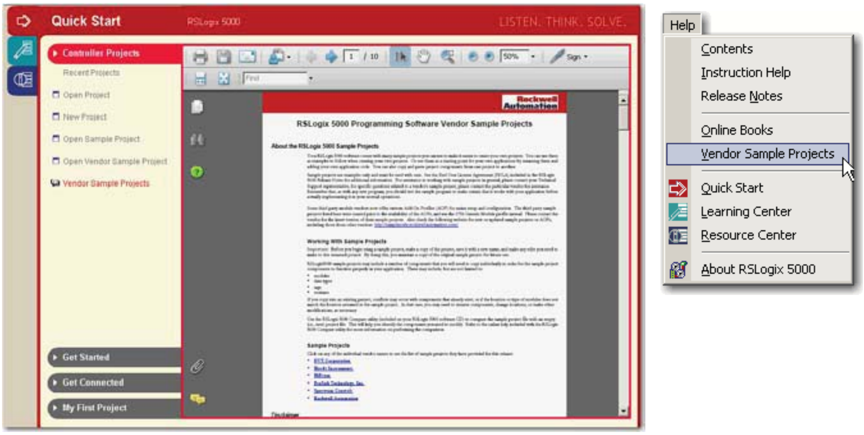
You can access and download the program at:

http://www.ab.com/motion/software/analyzer_download.html

The Motion Analyzer guides you through wizard-like screens to collect information specific to your application. After you enter the information, for example, the load inertia, gear box ratio, feedback device, and brake requirements, the Motion Analyzer generates an easy-to-read list of recommended motors, drives, and other support equipment.

Where to Find Sample Projects

Use the RSLogix 5000 software Start Page (ALT+F9) to find the sample projects.



The Rockwell Automation sample project’s default location is:

C:\RSLogix 5000\Projects\Samples\ENU\v18\Rockwell Automation

There is a PDF file named Vendor Sample Projects on the Start Page that explains how to work with the sample projects. Free sample code is available at: http://samplecode.rockwellautomation.com/.

Additional Resources

These documents contain additional information concerning related Rockwell Automation products. You can view or download publications at http://literature.rockwellautomation.com. To order paper copies of technical documentation, contact your local Rockwell Automation distributor or sales representative.

Resource	Description
Motion Coordinate System User Manual, publication MOTION-UM002.	Provides details on how to create and configure a coordinated motion system.
Logix5000 Controller Motion Instructions Reference Manual, publication MOTION-RM002.	Provides a programmer with details about motion instructions for a Logix-based controller.
Logix5000 Controllers Quick Start, publication 1756-QS001.	Describes how to get started programming and maintaining Logix5000 controllers.
Logix5000 Controllers Common Procedures, publication 1756-PM001.	Provides detailed and comprehensive information about how to program a Logix5000 controller.
Logix5000 Controllers General Instructions Reference Manual, publication 1756-RM003.	Provides a programmer with details about general instructions for a Logix-based controller.
Logix5000 Controllers Process and Drives Instructions Reference Manual, publication 1756-RM006.	Provides a programmer with details about process and drives instructions for a Logix-based controller.
PhaseManager User Manual, publication LOGIX-UM001.	Describes how to set up and program a Logix5000 controller to use equipment phases.
ControlLogix Controller User Manual, publication 1756-UM001.	Describes the necessary tasks to install, configure, program, and operate a ControlLogix system.
CompactLogix Controllers User Manual, publication 1768-UM001.	Describes the necessary tasks to install, configure, program, and operate a CompactLogix system.
Analog Encoder (AE) Servo Module Installation Instructions, publication 1756-IN047.	Provides installation instructions for the Analog Encoder (AE) Servo Module, catalog number 1756-M02AE.
ControlLogix SERCOS interface Module Installation Instructions, publication 1756-IN572.	Provides installation instructions for the ControlLogix SERCOS interface modules, catalog number 1756-M03SE, 1756-M08SE, 1756-M16SE, 1756-M08SEG.
CompactLogix SERCOS interface Module Installation Instructions, publication 1768-IN005.	Provides installation instructions for the CompactLogix SERCOS interface Module, catalog number 1768-M04SE.
Ultra3000 Digital Servo Drives Installation Manual, publication 2098-IN003.	Provides the mounting, wiring, and connecting procedures for the Ultra3000 drives and standard Rockwell Automation/Allen-Bradley motors recommended for use with the Ultra3000 drives.

Resource	Description
Ultra3000 Digital Servo Drives Integration Manual, publication 2098-IN005.	Provides powerup procedures, system integration, and troubleshooting tables for the Ultra3000 digital servo drives.
Kinetix 6000 Installation Manual, publication 2094-IN001.	Provides installation instructions for the Kinetix 6000 Integrated Axis Module and Axis Module series B drive components.
Kinetix 6000 Integration Manual, publication 2094-UM001.	Provides detailed installation instructions for mounting, wiring, and troubleshooting your Kinetix 6000 drive, and system integration for your drive/motor combination with a Logix controller.
Kinetix 6200 and Kinetix 6500 Safe Speed Monitoring Safety Reference Manual, publication 2094-RM001.	Information on wiring, configuring, and troubleshooting the safety functions of your Kinetix 6200 and Kinetix 6500 drives.
Kinetix 6200 and Kinetix 6500 Safe Torque-off Safety Reference Manual, publication 2094-RM002.	
8720MC High Performance Drive Installation Manual, publication 8720MC-IN001.	Provides the mounting, wiring, and connecting procedures for the 8720MC and standard Rockwell Automation/Allen-Bradley motors recommended for use with the 8720MC drive.
8720MC High Performance Drive Integration Manual, publication 8720MC-IN002.	This manual provides the startup, configuration, and troubleshooting procedures for the 8720MC drive.
Industrial Automation Wiring and Grounding Guidelines, publication 1770-4.1.	Provides general guidelines for installing a Rockwell Automation industrial system.
Product Certifications website, http://www.ab.com.	Provides declarations of conformity, certificates, and other certification details.

Introduction

The following are major changes from the previous version of the obsolete manual LOGIX-UM002.

Item	Description	Page
Setting Time Synchronization	You must make one controller in the chassis the Grandmaster clock for motion control. The motion modules set their clocks to the Grandmaster clock.	21
Electronic Keying	An expanded description of electronic keying for modules.	23
Set the Data Rate and Cycle Time	Changes to values.	32
Monitoring Axis Tags	You can now sort the tags alphabetically.	220
Creating Reports	Create reports, for example, tags, axis properties, and module properties.	221
AbsoluteReferenceStatus bit	Conditions for the bit to be reset.	227
Absolute Homing	Sercos Absolute Homing Process Flowchart	229
Replicated Attributes	Controller attributes that are replicated in the drive as listed.	225
Servo Loop Block Diagrams	Updated graphic quality. Updated the Dual Position Servo graphic to include Feedback Ratio under the Velocity Command signal.	361
The Motion Coordinate System instructions and programming information is now located in The Motion Coordinate System User Manual, publication MOTION-UM002		

Notes:

Chapter 1

Configure SERCOS Motion

Introduction

Use this chapter for procedures on how to set up SERCOS motion control.

Create a Controller Project

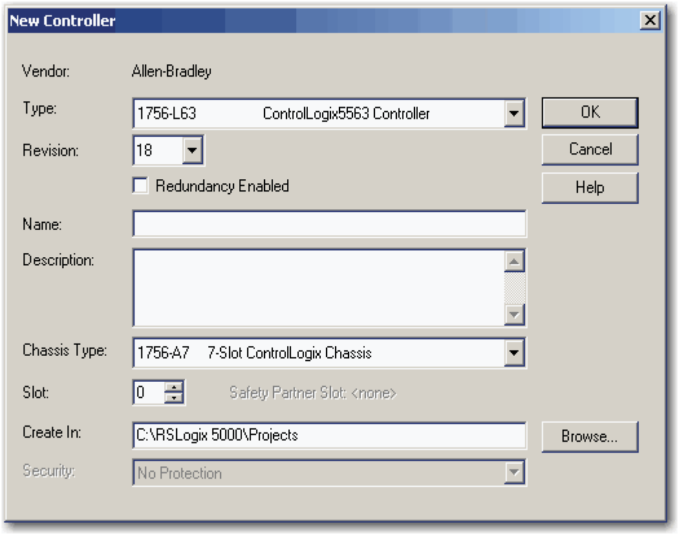
Follow these instructions to create a controller project.

1. Open RSLogix 5000 software.

   If the Start Page appears, close it.

2. From the File menu, choose New.

   The New Controller window appears.



New Controller

Vendor: Allen-Bradley

Type: 1756-L63 ControlLogix5563 Controller

Revision: 18

☐ Redundancy Enabled

Name:

Description:

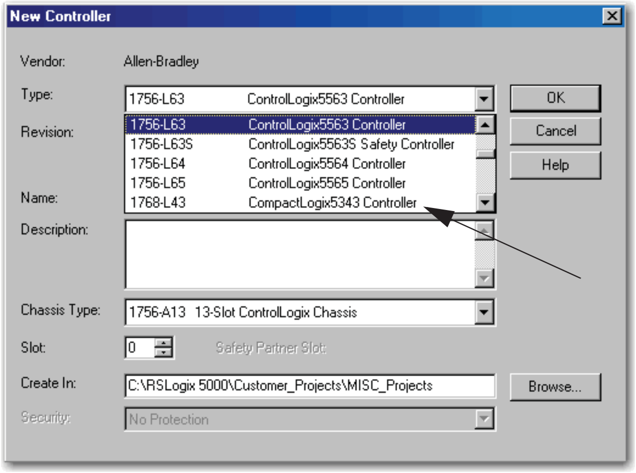
Chassis Type: 1756-A7 7-Slot ControlLogix Chassis

Slot: 0 Safety Partner Slot: <none>

Create In: C:\RSLogix 5000\Projects

Security: No Protection

You can also select a CompactLogix controller.



New Controller

Vendor: Allen-Bradley

Type: 1756-L63 ControlLogix5563 Controller

1756-L63 ControlLogix5563 Controller
1756-L63S ControlLogix5563S Safety Controller
1756-L64 ControlLogix5564 Controller
1756-L65 ControlLogix5565 Controller
1768-L43 CompactLogix5343 Controller

Revision:

Name:

Description:

Chassis Type: 1756-A13 13-Slot ControlLogix Chassis

Slot: 0 Safety Partner Slot:

Create In: C:\RSLogix 5000\Customer_Projects\MISC_Projects

Security: No Protection

For more information about the tasks to install, configure, program, and operate a CompactLogix system, see CompactLogix Controllers User Manual, publication 1768-UM001.

3. Configure the controller to match your project requirements.

   You will need to determine the following:

   - Type of Controller
   - Software Revision
   - Name of Project
   - Chassis Type
   - Controller Slot Location
   - Project Location

4. Click OK.

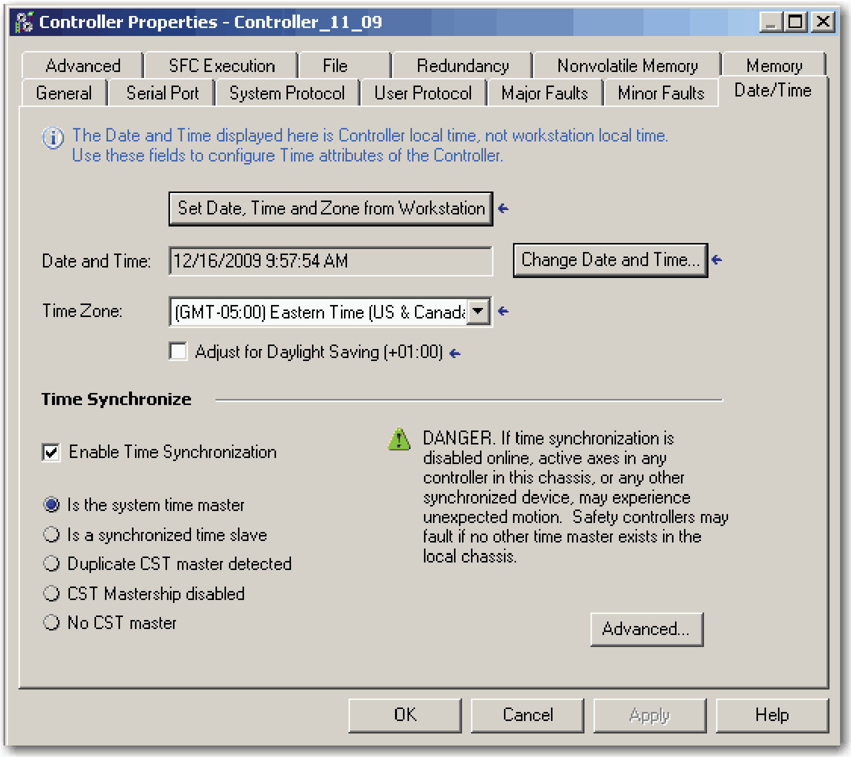
Setting Time Synchronization

Time Synchronization in ControlLogix is called CIP Sync. CIP Sync is a layer of functionality that Rockwell Automation has developed on top of the IEEE 1588 PTP protocol. CIP Sync lets you maintain accurate time synchronization of your automation solutions.

This setting establishes the module to participate in time synchronization. In systems with multiple processors all controllers need to have time synchronization enabled if they are to use CST/PTP time. The 1756-EN*x*T communication modules win the arbitration over any processor.

1. In the Controller Organizer, right-click the controller and choose Properties or double-click the controller.

2. Click the Date/Time tab.



The Date and Time displayed here is Controller local time, not workstation local time.
Use these fields to configure Time attributes of the Controller.

Date and Time: 12/16/2009 9:57:54 AM

Time Zone: (GMT-05:00) Eastern Time (US & Canada

☐ Adjust for Daylight Saving (+01:00)

Time Synchronize

☑ Enable Time Synchronization

● Is the system time master
○ Is a synchronized time slave
○ Duplicate CST master detected
○ CST Mastership disabled
○ No CST master

DANGER. If time synchronization is disabled online, active axes in any controller in this chassis, or any other synchronized device, may experience unexpected motion. Safety controllers may fault if no other time master exists in the local chassis.

3. Check Enable Time Synchronization.

4. Click OK.

Without intervention, the Grandmaster is both PTP and CST master. You can use the settings on the Advanced Dialog box to let this module to win the arbitration over other processors and communication modules in the chassis.

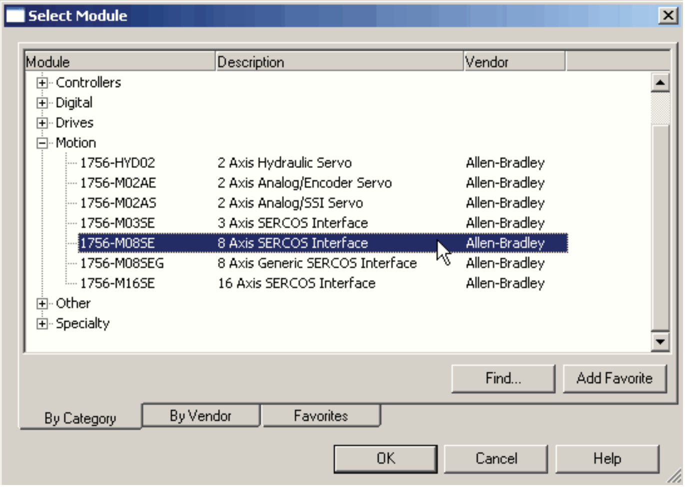
See the Integrated Architecture and CIP Sync Configuration Application Technique, publication IA-AT003 for more information.

Add a SERCOS Motion Module

Follow these instruction to add a SERCOS motion module to your system.

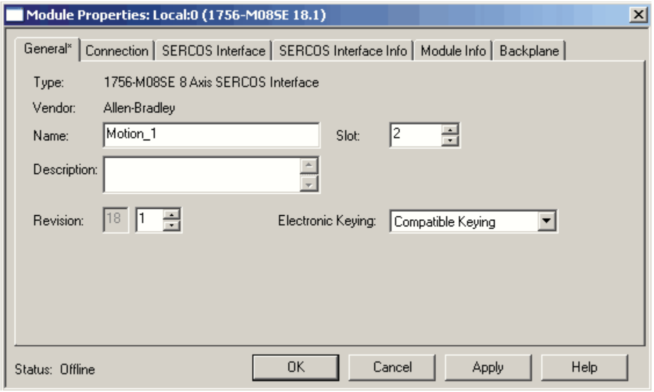
IMPORTANT For all motion modules, use the firmware revision that goes with the firmware revision of your controller. See the release notes for your controller's firmware.

1. To add a module, right-click on the backplane and choose New Module.
2. Select the module that you want to add and click OK.



Module	Description	Vendor
Controllers		
Digital		
Drives		
Motion		
1756-HYD02	2 Axis Hydraulic Servo	Allen-Bradley
1756-M02AE	2 Axis Analog/Encoder Servo	Allen-Bradley
1756-M02AS	2 Axis Analog/SSI Servo	Allen-Bradley
1756-M03SE	3 Axis SERCOS Interface	Allen-Bradley
1756-M08SE	8 Axis SERCOS Interface	Allen-Bradley
1756-M08SEG	8 Axis Generic SERCOS Interface	Allen-Bradley
1756-M16SE	16 Axis SERCOS Interface	Allen-Bradley
Other		
Specialty		

3. Name the module.
4. Assign the slot for the module.



Module Properties: Local:0 (1756-M08SE 18.1)

Type: 1756-M08SE 8 Axis SERCOS Interface
Vendor: Allen-Bradley
Name: Motion_1
Slot: 2
Description:
Revision: 18 1
Electronic Keying: Compatible Keying
Status: Offline

5. Type a description, if desired.
6. Choose a Keying option.

WARNING When using motion modules the electronic keying should be either 'Exact Match' or 'Compatible Keying'.

'Disable Keying' should **never** be used with motion module.

Electronic Keying

The electronic keying feature automatically compares the expected module, as shown in the RSLogix 5000 I/O Configuration tree, to the physical module before I/O communication begins. You can use electronic keying to help prevent communication to a module that does not match the type and revision expected.

For each module in the I/O Configuration tree, the user-selected keying option determines if, and how, an electronic keying check is performed. Typically, three keying option are available.

- Exact Match
- Compatible Keying
- Disable Keying

You must carefully consider the benefits and implications of each keying option when selecting between them. For some specific module types, fewer options are available.

Electronic keying is based on a set of attributes unique to each product revision. When a Logix5000 controller begins communicating with a module, this set of keying attributes is considered.

Keying Attributes

Attribute	Description
Vendor	The manufacturer of the module, for example, Rockwell Automation/Allen-Bradley.
Product Type	The general type of the module, for example, communication adapter, AC drive, or digital I/O.
Product Code	The specific type of module, generally represented by its catalog number, for example, the 1756-IB16I module.
Major Revision	A number that represents the functional capabilities and data exchange formats of the module. Typically, although not always, a later, that is higher, Major Revision supports at least all of the data formats supported by an earlier, that is lower, Major Revision of the same catalog number and, possibly, additional ones.
Minor Revision	A number that indicates the module's specific firmware revision. Minor Revisions typically do not impact data compatibility but may indicate performance or behavior improvement.

You can find revision information on the General tab of a module's Properties dialog box.

General Tab

IMPORTANT Changing electronic keying selections online may cause the I/O communication connection to the module to be disrupted and may result in a loss of data.

Exact Match

Exact Match keying requires all keying attributes, that is, Vendor, Product Type, Product Code (catalog number), Major Revision, and Minor Revision, of the physical module and the module created in the software to match precisely in order to establish communication. If any attribute does not match precisely, I/O communication is not permitted with the module or with modules connected through it, as in the case of a communication module.

Use Exact Match keying when you need the system to verify that the module revisions in use are exactly as specified in the project, such as for use in highly-regulated industries. Exact Match keying is also necessary to enable

Automatic Firmware Update for the module via the Firmware Supervisor feature from a Logix5000 controller.

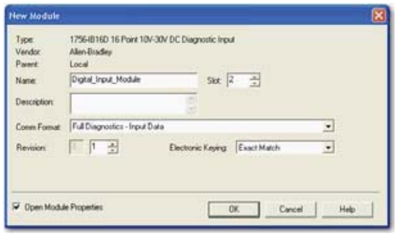
EXAMPLE

In the following scenario, Exact Match keying prevents I/O communication:

- The module configuration is for a 1756-IB16D module with module revision 3.1. The physical module is a 1756-IB16D module with module revision 3.2. In this case, communication is prevented because the Minor Revision of the module does not match precisely.

Module Configuration

Vendor = Allen-Bradley
Product Type = Digital Input Module
Catalog Number = 1756-IB16D
Major Revision = 3
Minor Revision = 1

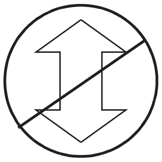


New Module

Type: 1756-IB16D 16 Point 10V-30V DC Diagnostic Input
Vendor: Allen-Bradley
Parent: Local
Name: Digital_Input_Module
Slot: 2
Description:
Comm Format: Full Diagnostics - Input Data
Revision: 1
Electronic Keying: Exact Match

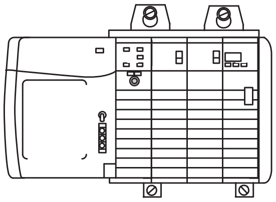
☑ Open Module Properties

Communication is prevented.



Physical Module

Vendor = Allen-Bradley
Product Type = Digital Input Module
Catalog Number = 1756-IB16D
Major Revision = 3
Minor Revision = 2



IMPORTANT

Changing electronic keying selections online may cause the I/O Communication connection to the module to be disrupted and may result in a loss of data.

Compatible Keying

Compatible Keying indicates that the module determines whether to accept or reject communication. Different module families, communication adapters, and module types implement the compatibility check differently based on the family capabilities and on prior knowledge of compatible products.

Compatible Keying is the default setting. Compatible Keying allows the physical module to accept the key of the module configured in the software, provided that the configured module is one the physical module is capable of emulating. The exact level of emulation required is product and revision specific.

With Compatible Keying, you can replace a module of a certain Major Revision with one of the same catalog number and the same or later, that is higher, Major Revision. In some cases, the selection makes it possible to use a replacement that is a different catalog number than the original. For example, you can replace a 1756-CNBR module with a 1756-CN2R module.

Release notes for individual modules indicate the specific compatibility details.

When a module is created, the module developers consider the module's development history to implement capabilities that emulate those of the previous module. However, the developers cannot know future developments. Because of this, when a system is configured, we recommend that you configure your module using the earliest, that is, lowest, revision of the physical module that you believe will be used in the system. By doing this, you can avoid the case of a physical module rejecting the keying request because it is an earlier revision than the one configured in the software.

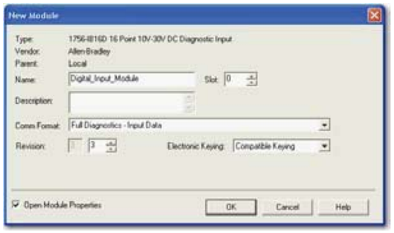
EXAMPLE

In the following scenario, **Compatible Keying prevents I/O communication**:

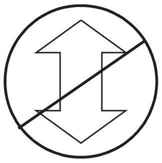
- The module configuration is for a 1756-IB16D module with module revision 3.3. The physical module is a 1756-IB16D module with module revision 3.2. In this case, communication is prevented because the minor revision of the module is lower than expected and may not be compatible with 3.3.

Module Configuration

Vendor = Allen-Bradley
Product Type = Digital Input Module
Catalog Number = 1756-IB16D
Major Revision = 3
Minor Revision = 3

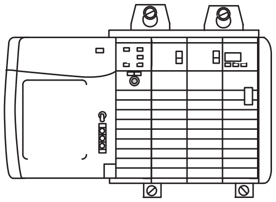

New Module
Type: 1756-IB16D 16 Point 10V-30V DC Diagnostic Input
Vendor: Allen-Bradley
Parent: Local
Name: Digital_Input_Module
Slot: 0
Description:
Comm Format: Full Diagnostics - Input Data
Revision: 3 3
Electronic Keying: Compatible Keying
☑ Open Module Properties

Communication is prevented.



Physical Module

Vendor = Allen-Bradley
Product Type = Digital Input Module
Catalog Number = 1756-IB16D
Major Revision = 3
Minor Revision = 2



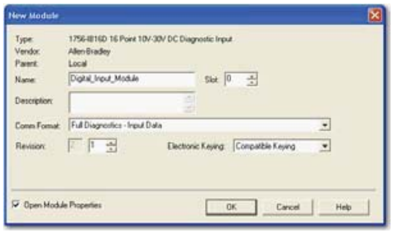
EXAMPLE

In the following scenario, **Compatible Keying allows I/O communication**:

- The module configuration is for a 1756-IB16D module with module revision 2.1. The physical module is a 1756-IB16D module with module revision 3.2. In this case, communication is allowed because the major revision of the physical module is higher than expected and the module determines that it is compatible with the prior major revision.

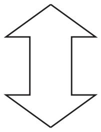
Module Configuration

Vendor = Allen-Bradley
Product Type = Digital Input Module
Catalog Number = 1756-IB16D
Major Revision = 2
Minor Revision = 1



New Module

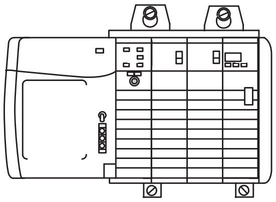
Type: 1756-IB16D 16 Point 10V-30V DC Diagnostic Input
Vendor: Allen-Bradley
Parent: Local
Name: Digital_Input_Module
Slot: 0
Description:
Comm Format: Full Diagnostics - Input Data
Revision: 2 1
Electronic Keying: Compatible Keying
☑ Open Module Properties



Communication is allowed.

Physical Module

Vendor = Allen-Bradley
Product Type = Digital Input Module
Catalog Number = 1756-IB16D
Major Revision = 3
Minor Revision = 2



IMPORTANT

Changing electronic keying selections online may cause the I/O communication connection to the module to be disrupted and may result in a loss of data.

Disabled Keying

Disabled Keying indicates the keying attributes are not considered when attempting to communicate with a module. Other attributes, such as data size and format, are considered and must be acceptable before I/O communication is established. With Disabled Keying, I/O communication may occur with a module other than the type specified in the I/O Configuration tree with unpredictable results. We generally do not recommend using Disabled Keying.

ATTENTION

Be extremely cautious when using Disabled Keying; if used incorrectly, this option can lead to personal injury or death, property damage, or economic loss.

If you use Disabled Keying, you must take full responsibility for understanding whether the module being used can fulfill the functional requirements of the application.

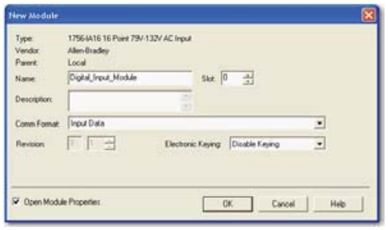
EXAMPLE

In the following scenario, **Disable Keying prevents I/O communication**:

- The module configuration is for a 1756-IA16 digital input module. The physical module is a 1756-IF16 analog input module. In this case, **communication is prevented because the analog module rejects the data formats that the digital module configuration requests**.

Module Configuration

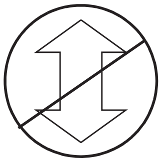
Vendor = Allen-Bradley
Product Type = Digital Input Module
Catalog Number = 1756-IA16
Major Revision = 3
Minor Revision = 1



New Module

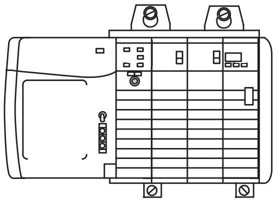
Type: 1756-IA16 16 Point 79V-132V AC Input
Vendor: Allen-Bradley
Parent: Local
Name: Digital_Input_Module
Slot: 0
Description:
Comm Format: Input Data
Revision:
Electronic Keying: Disable Keying
☑ Open Module Properties

Communication is prevented.



Physical Module

Vendor = Allen-Bradley
Product Type = Analog Input Module
Catalog Number = 1756-IF16
Major Revision = 3
Minor Revision = 2



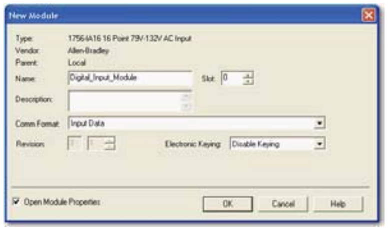
EXAMPLE

In the following scenario, **Disable Keying allows I/O communication**:

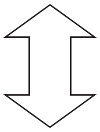
- The module configuration is for a 1756-IA16 digital input module. The physical module is a 1756-IB16 digital input module. In this case, communication is allowed because the two digital modules share common data formats.

Module Configuration

Vendor = Allen-Bradley
Product Type = Digital Input Module
Catalog Number = 1756-IA16
Major Revision = 2
Minor Revision = 1



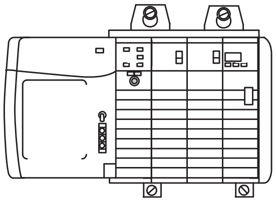
Type: 1756-IA16 16 Point 79V-132V AC Input
Vendor: Allen-Bradley
Parent: Local
Name: Digital_Input_Module
Slot: 0
Description:
Comm Format: Input Data
Revision:
Electronic Keying: Disable Keying
☑ Open Module Properties



Communication is allowed.

Physical Module

Vendor = Allen-Bradley
Product Type = Digital Input Module
Catalog Number = 1756-IB16
Major Revision = 3
Minor Revision = 2



IMPORTANT

Changing electronic keying selections online may cause the I/O communication connection to the module to be disrupted and may result in a loss of data.

7. Check Open Module Properties box, if you want the Properties dialog box to open automatically after you are done creating the new module.

☑ Open Module Properties

If you do not check the box, you will need to either double-click the module in the Controller Organizer or right-click the module and choose Properties.

8. Click OK.

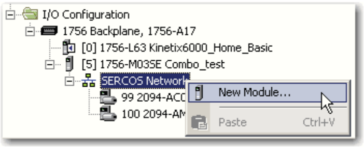
Add SERCOS Interface Drives

Add SERCOS interface drives to the I/O configuration of the controller.

IMPORTANT For all interface drives, use the firmware revision that goes with the firmware revision of your controller. See the release notes for your controller's firmware.

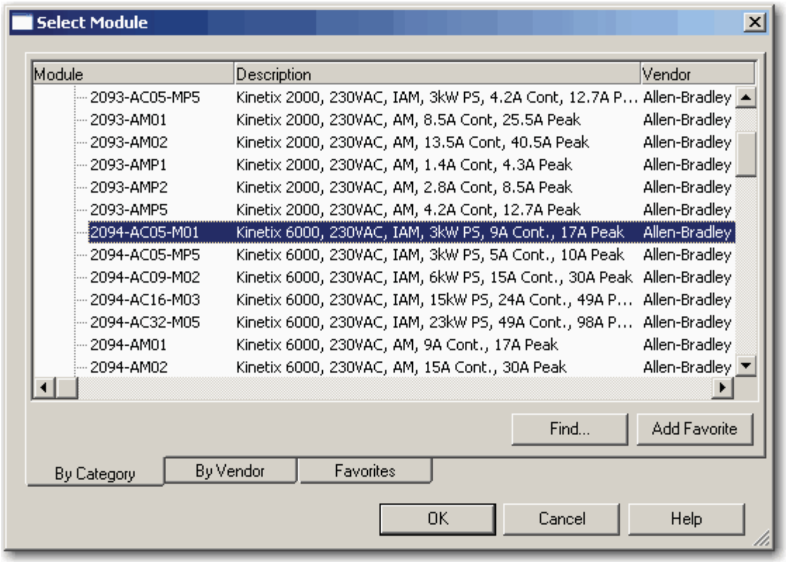
---

1. To add a module, right-click the SERCOS Network and choose New Module.



I/O Configuration
  1756 Backplane, 1756-A17
    [0] 1756-L63 Kinetix6000_Home_Basic
    [5] 1756-M03SE Combo_test
      SERCOS Network
        99 2094-ACC
        100 2094-AM

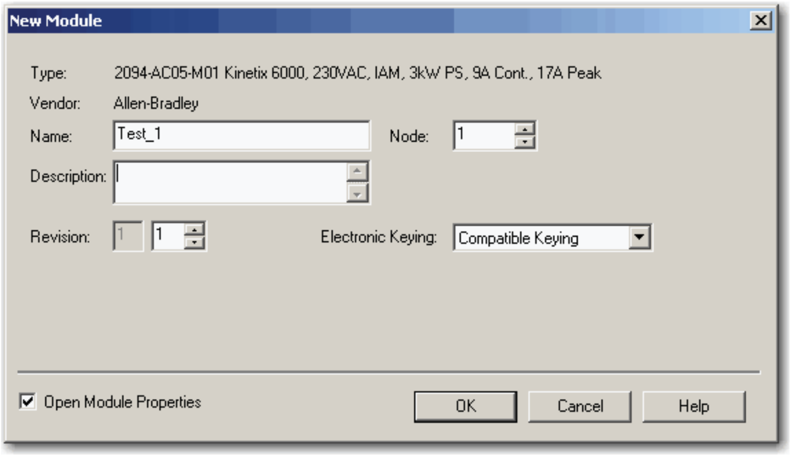
2. Select the module that you want to add and click OK.



Select Module

Module	Description	Vendor
2093-AC05-MP5	Kinetix 2000, 230VAC, IAM, 3kW PS, 4.2A Cont, 12.7A P...	Allen-Bradley
2093-AM01	Kinetix 2000, 230VAC, AM, 8.5A Cont, 25.5A Peak	Allen-Bradley
2093-AM02	Kinetix 2000, 230VAC, AM, 13.5A Cont, 40.5A Peak	Allen-Bradley
2093-AMP1	Kinetix 2000, 230VAC, AM, 1.4A Cont, 4.3A Peak	Allen-Bradley
2093-AMP2	Kinetix 2000, 230VAC, AM, 2.8A Cont, 8.5A Peak	Allen-Bradley
2093-AMP5	Kinetix 2000, 230VAC, AM, 4.2A Cont, 12.7A Peak	Allen-Bradley
2094-AC05-M01	Kinetix 6000, 230VAC, IAM, 3kW PS, 9A Cont., 17A Peak	Allen-Bradley
2094-AC05-MP5	Kinetix 6000, 230VAC, IAM, 3kW PS, 5A Cont., 10A Peak	Allen-Bradley
2094-AC09-M02	Kinetix 6000, 230VAC, IAM, 6kW PS, 15A Cont., 30A Peak	Allen-Bradley
2094-AC16-M03	Kinetix 6000, 230VAC, IAM, 15kW PS, 24A Cont., 49A P...	Allen-Bradley
2094-AC32-M05	Kinetix 6000, 230VAC, IAM, 23kW PS, 49A Cont., 98A P...	Allen-Bradley
2094-AM01	Kinetix 6000, 230VAC, AM, 9A Cont., 17A Peak	Allen-Bradley
2094-AM02	Kinetix 6000, 230VAC, AM, 15A Cont., 30A Peak	Allen-Bradley

3. Name the module.



New Module

Type: 2094-AC05-M01 Kinetix 6000, 230VAC, IAM, 3kW PS, 9A Cont., 17A Peak
Vendor: Allen-Bradley
Name: Test_1
Node: 1
Description:
Revision: 1 1
Electronic Keying: Compatible Keying

☑ Open Module Properties

4. Assign a slot for the module.
5. Type a description, if desired.
6. Choose a Keying option.

WARNING

When using motion modules the electronic keying should be either 'Exact Match' or 'Compatible Keying'.

'Disable Keying' should **never** be used with motion module.

See Electronic Keying on page 23.

7. Check Open Module Properties, if you want the Properties dialog box box to open automatically after you are done creating the new module.

   If you do not check the box, you will need to either double-click the module in the Controller Organizer or right-click the module and choose Properties.

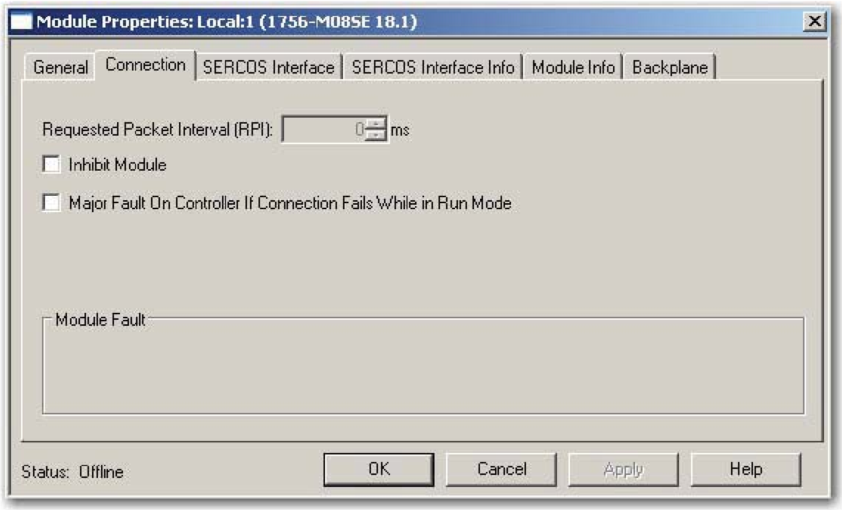
Modify Properties for a SERCOS Motion Module

Follow these instructions to modify the module properties.

1. Right-click the module you want to configure and choose Properties.
2. Click the Connection Tab.
3. Review the Major Fault on Controller function.

TIP

By default this box is unchecked. If this box is checked, then a Major Fault is generated when the SERCOS ring attempts to phase up the first time. Leave unchecked until you execute the program the first time.


Module Properties: Local:1 (1756-M08SE 18.1)

Requested Packet Interval (RPI): 0 ms

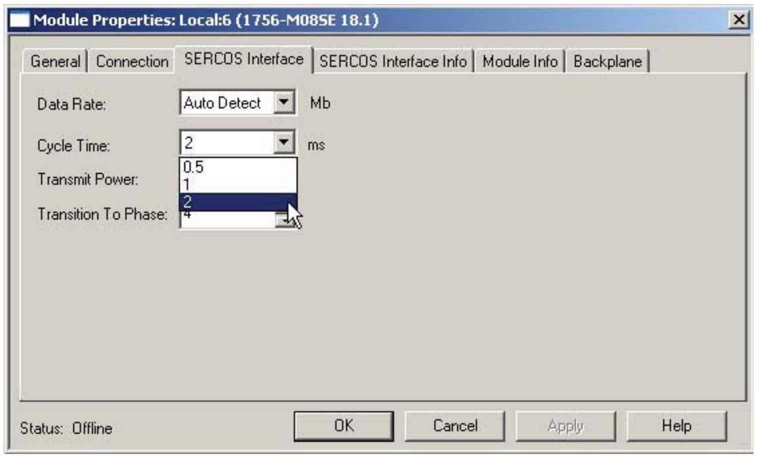
☐ Inhibit Module

☐ Major Fault On Controller If Connection Fails While in Run Mode

Status: Offline

Set the Data Rate and Cycle Time

Next you need to set the data rate and cycle time for each SERCOS interface module in your project. The cycle time is how often the SERCOS ring is read. The cycle time update rate is only applicable to a SERCOS application.


Module Properties: Local:6 (1756-M08SE 18.1)

Data Rate: Auto Detect Mb

Cycle Time: 2 ms

Transmit Power:

Transition To Phase:

Status: Offline

Based on the data rate and the number of drives associated with the module will dictate your minimum cycle time.

For more information configuring the Kinetix 6200 and Kinetix 6500 servo drive systems, see the Kinetix 6200 and Kinetix 6500 Modular Multi-axis Servo Drive User Manual, publication 2094-UM002.

This table describes the cycle time in relation to the modules.

Cycle Times

Baud Rate of Drives	Number of Drives on the Ring	Type of Drives	Cycle Time
4 Mbps	Up to 2	Kinetix 2000 Kinetix 6000 Kinetix 6200 Kinetix 7000	0.5 ms
	Up to 4	→	1 ms
	Up to 8	→	2 ms
	9…16	→	No Support
8 Mbps	Up to 4	Kinetix 2000 Kinetix 6000 Kinetix 6200 Kinetix 7000	0.5 ms
	Up to 8	→	1 ms
	Up to 16	→	2 ms

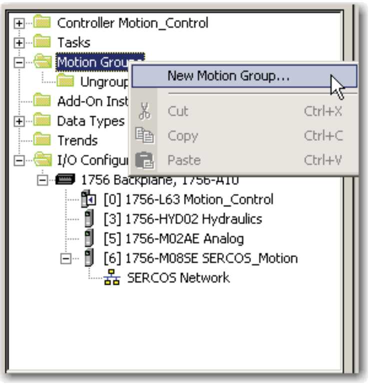
TIP The Ultra 3000 must be 1 or 2 ms SERCOS cycle time because the position loop update time minimum is 1 ms.

Add the Motion Group

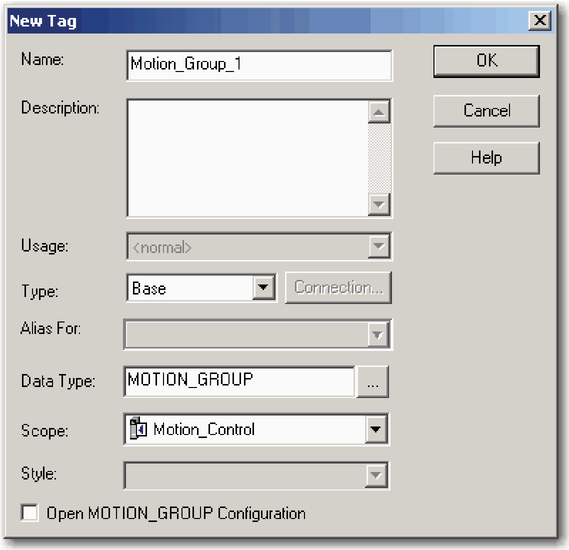
Follow these instructions to add a motion group.

IMPORTANT Only one motion group can be created for each project.

1. Right-click Motion Groups and choose New Motion Group.



2. Type a name and description, if desired and click OK.

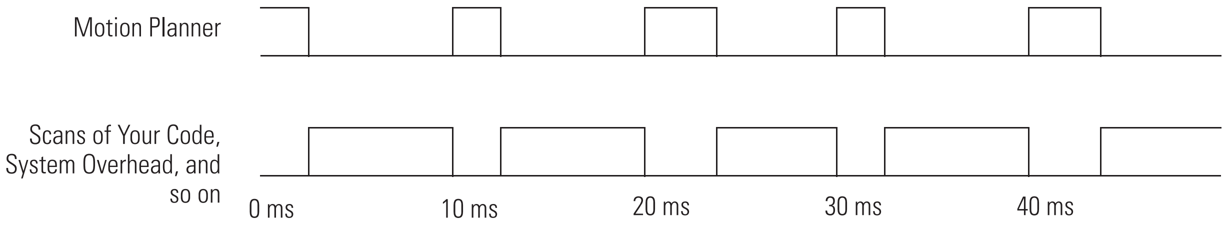


Name: Motion_Group_1
Description:
Usage: <normal>
Type: Base
Alias For:
Data Type: MOTION_GROUP
Scope: Motion_Control
Style:
☐ Open MOTION_GROUP Configuration

If you check Open Motion_Group Configuration, the Motion Group Wizard appears. If not you can access the Motion Group properties by right-clicking the group and choosing Properties.

Set the Course Update Period

The Course Update Period is how often the motion planner runs. When the motion planner runs, it interrupts most other tasks regardless of their priority. The motion planner is the part of the controller that takes care of position and velocity information for the axes.



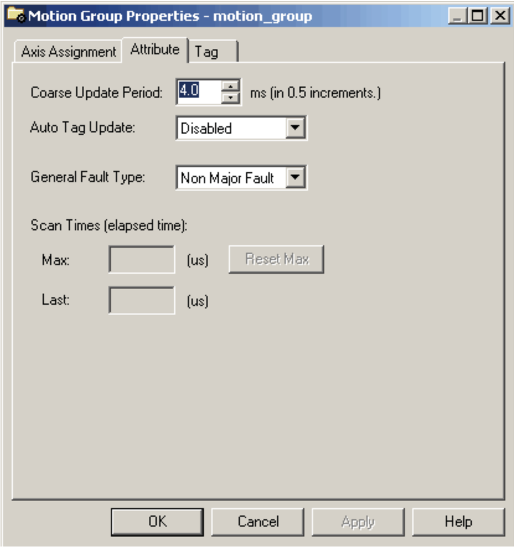
In this example, the Coarse Update Period = 10 ms. Every 10 ms the controller stops scanning your code and whatever else it is doing and runs the motion planner.

The Coarse Update Period is a trade-off between updating positions of your axes and scanning your code. Use these guidelines as a rough starting point.

Guideline	Description
Number of Axes	1756-L6*x* controller 4 axes/ms 1756-L7*x* controller 8 axes/ms
Save Controller's Time	Leave at least half the controller's time for the scan of all your code.
Coarse Update Period and SERCOS modules	If you have SERCOS interface motion modules, set the Coarse Update Period to a multiple of the cycle time of the motion module. Example: if the cycle time is 2 ms, set the Coarse Update Period to 8 ms, 10 ms, 12ms, and so on.
Coarse Update Period and Analog modules	If you have analog motion modules, set the Coarse Update Period to: • at least 3 times the servo update period of the motion module. • a multiple of the servo update period of the motion module.

3. Double-click the Motion Group in the Controller Organizer or the Wizard appears.

4. Set the Coarse Update Period.



Coarse Update Period: 4.0 ms (in 0.5 increments.)

Auto Tag Update: Disabled

General Fault Type: Non Major Fault

Scan Times (elapsed time):

Max: (us)

Last: (us)

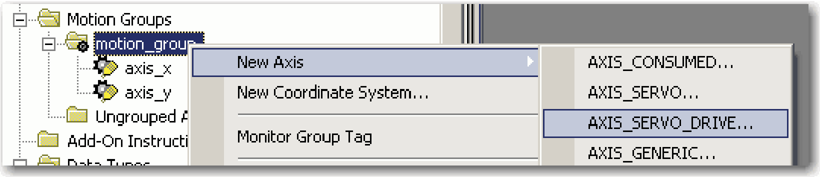
5. Set the Auto Tag Update

6. Set the General Fault Type to Non Major Fault and click OK.

Add an Axis

Add an axis for each of your drives.

1. Right-click Motion Group and choose New Axis.



If you use one of these motion modules then use the data type AXIS_SERVO_DRIVE for the axis.

- 1756-M03SE
- 1756-M08SE
- 1756-M16SE
- 1768-M04SE

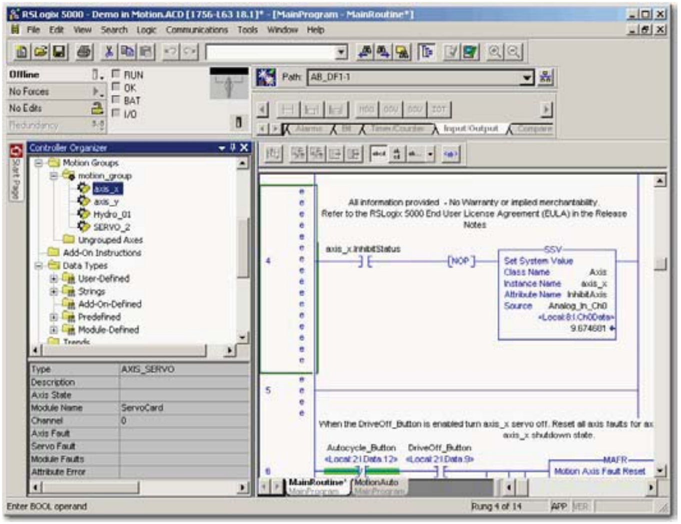
2. Choose AXIS_SERVO_DRIVE.

   If you want to use a virtual configuration (no hardware) then use the data type AXIS_VIRTUAL for the axis.

Get Axis Information

You can get information about an axis in several ways.

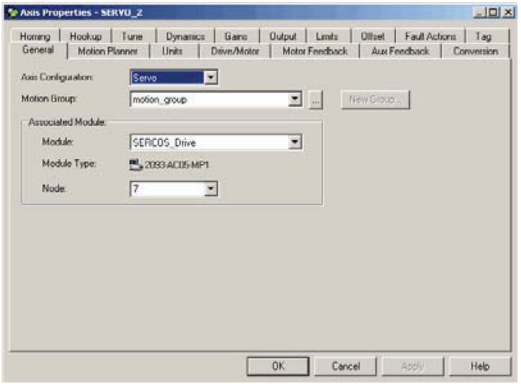
- Use the Quick View pane to see the state and faults of an axis.
- Use a Get System Value (GSV) instruction or Set System Value (SSV) instruction to read or change the configuration at run-time.
- Use the tag of the axis for status and faults.



Configure an Axis

The following steps show how to set up the axis of a SERCOS interface drive. The steps are may be different depending on the type of drive you are configuring.

1. Right-click the axis and choose Properties.
2. Click the General tab.
3. Select the drive for the axis.
4. Select the module name that you want to associate to the axis.



Axis Properties - SERVO_2

Axis Configuration: Servo

Motion Group: motion_group

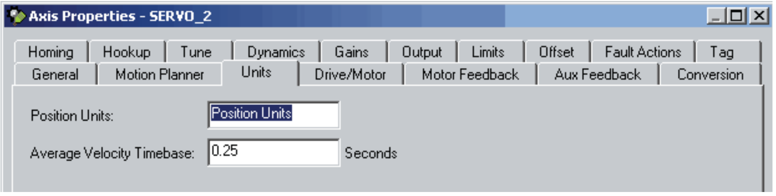
Associated Module:

Module: SERCOS_Drive

Module Type: 2093-AC05-MP1

Node: 7

5. Go to the Units tab and type the units and Average Velocity Timebase.

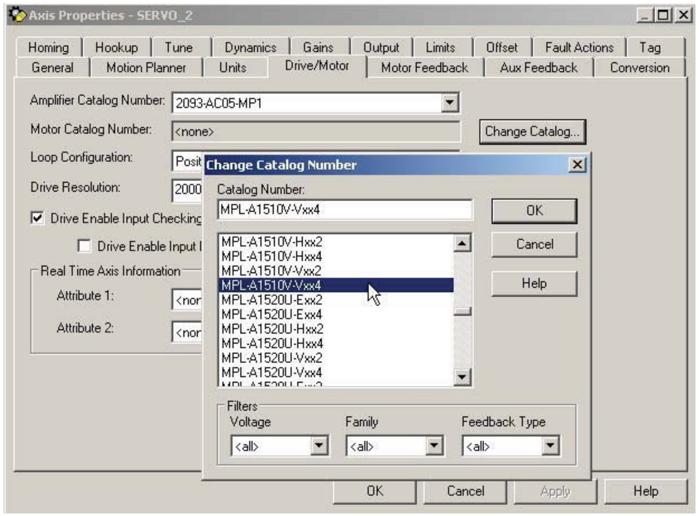


Axis Properties - SERVO_2

Position Units: Position Units

Average Velocity Timebase: 0.25 Seconds

6. Go to the Drive/Motor tab and select the Amplifier and related Motor Catalog Numbers.



Axis Properties - SERVO_2

Amplifier Catalog Number: 2093-AC05-MP1

Motor Catalog Number: <none>

Loop Configuration: Posit

Drive Resolution: 2000

☑ Drive Enable Input Checking

☐ Drive Enable Input I

Real Time Axis Information

Attribute 1: <nor

Attribute 2: <nor

Change Catalog Number

Catalog Number: MPL-A1510V-Vxx4

MPL-A1510V-Hxx2
MPL-A1510V-Hxx4
MPL-A1510V-Vxx2
MPL-A1510V-Vxx4
MPL-A1520U-Exx2
MPL-A1520U-Exx4
MPL-A1520U-Hxx2
MPL-A1520U-Hxx4
MPL-A1520U-Vxx2
MPL-A1520U-Vxx4

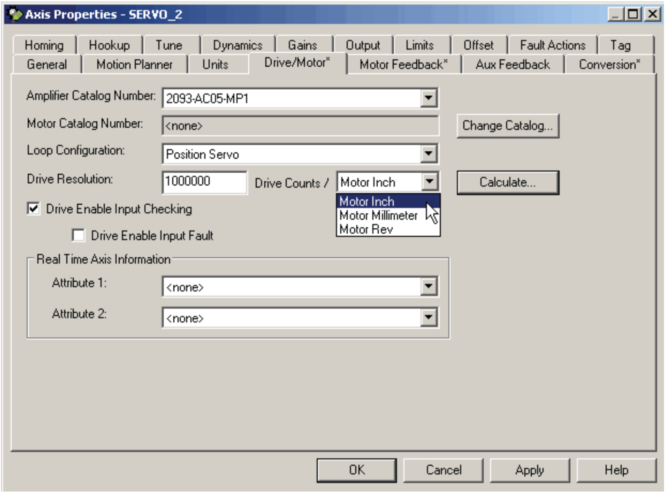
Filters

Voltage: <all>

Family: <all>

Feedback Type: <all>

7. Select the Drive/Counts for the Conversion Constant.



Amplifier Catalog Number: 2093-AC05-MP1

Motor Catalog Number: <none>

Loop Configuration: Position Servo

Drive Resolution: 1000000 Drive Counts / Motor Inch

Motor Inch
Motor Millimeter
Motor Rev

☑ Drive Enable Input Checking

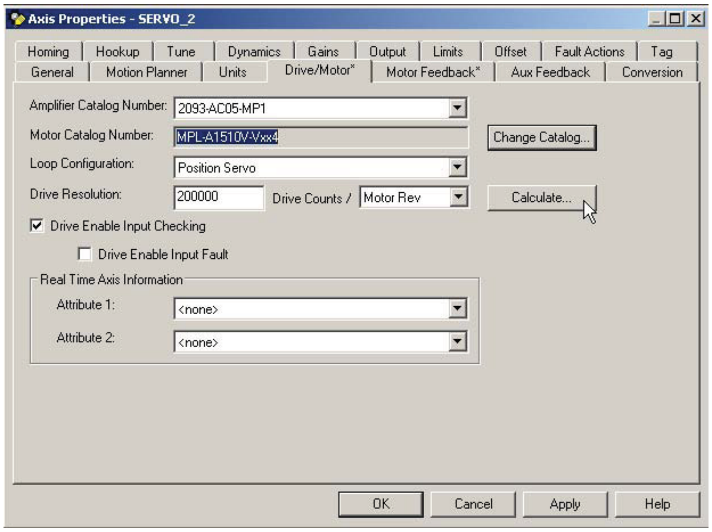
☐ Drive Enable Input Fault

Real Time Axis Information

Attribute 1: <none>

Attribute 2: <none>

8. If you want to calculate the conversion constant yourself, just put in your values, if not set the conversion between drive counts and units, and click Calculate.



Amplifier Catalog Number: 2093-AC05-MP1

Motor Catalog Number: MPL-A1510V-Vxx4

Loop Configuration: Position Servo

Drive Resolution: 200000 Drive Counts / Motor Rev

☑ Drive Enable Input Checking

☐ Drive Enable Input Fault

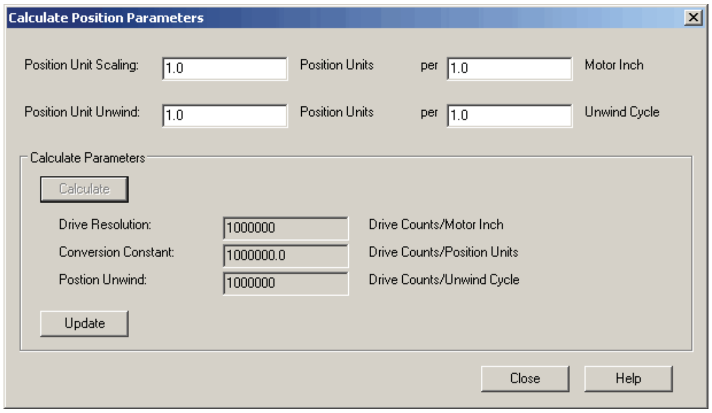
Real Time Axis Information

Attribute 1: <none>

Attribute 2: <none>

If this is a rotary axis, type the number of drive counts that you want to unwind after.

9. After you click Calculate, this dialog box appears. You can input different Position Units and click Calculate. You will see the new values. If you want to keep the new calculated values, click Update and Close.



Calculate Position Parameters

Position Unit Scaling: 1.0 Position Units per 1.0 Motor Inch

Position Unit Unwind: 1.0 Position Units per 1.0 Unwind Cycle

Calculate Parameters

Drive Resolution: 1000000 Drive Counts/Motor Inch

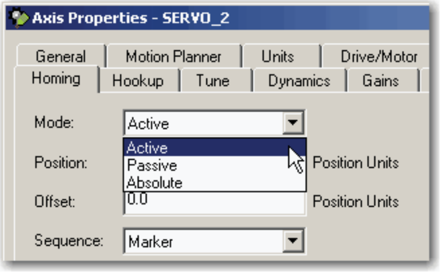
Conversion Constant: 1000000.0 Drive Counts/Position Units

Postion Unwind: 1000000 Drive Counts/Unwind Cycle

Set the Homing Sequence

For complete information about Homing modes, methods and guidelines, see Home an Axis on page 119.

Follow these instructions to set the homing sequence.
On the Homing tab select the homing mode.



Mode: Active

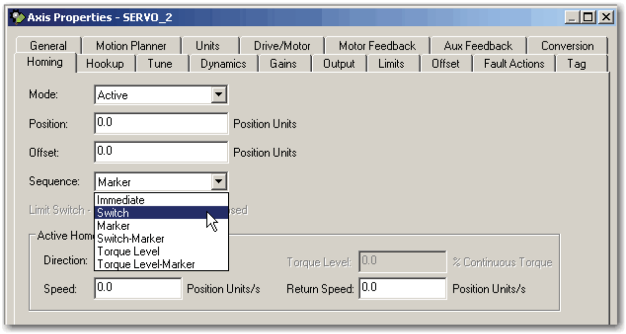
Position: Position Units

Offset: 0.0 Position Units

Sequence: Marker

Enter the Position units.

10. Select Sequence type.



Mode: Active

Position: 0.0 Position Units

Offset: 0.0 Position Units

Sequence: Marker

Direction:

Torque Level: 0.0 % Continuous Torque

Speed: 0.0 Position Units/s

Return Speed: 0.0 Position Units/s

Homing speeds appear dimmed depending on what Homing Mode and Sequence Type you select.

11. Set the homing speeds and click OK to apply the changes.

Notes:

Chapter 2

Configure Analog Motion

Introduction

Use this chapter for step-by-step procedures on how to set up analog motion control.

Create a Controller Project

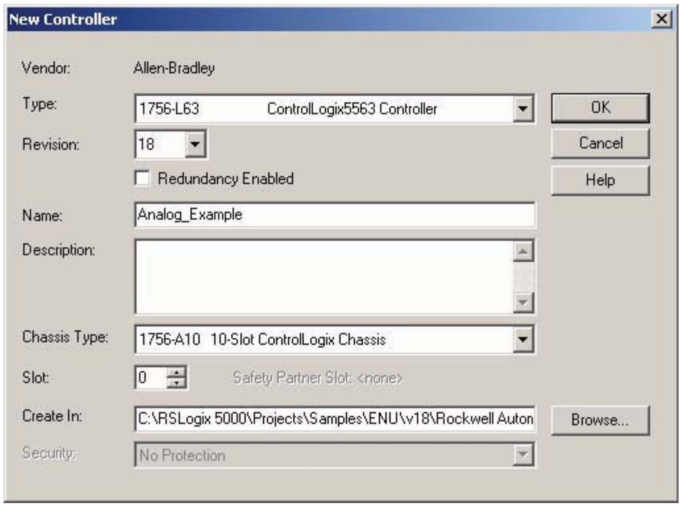
Follow these instructions to create a controller project.

1. Open RSLogix 5000 software.

   If the Start Page appears, close it.

2. From the File menu choose New.

   The New Controller window appears.



Vendor: Allen-Bradley
Type: 1756-L63 ControlLogix5563 Controller
Revision: 18
☐ Redundancy Enabled
Name: Analog_Example
Description:
Chassis Type: 1756-A10 10-Slot ControlLogix Chassis
Slot: 0 Safety Partner Slot: <none>
Create In: C:\RSLogix 5000\Projects\Samples\ENU\v18\Rockwell Auton
Security: No Protection

3. Configure the controller to match your project requirements.

   You will need to determine the following:

   - Type of Controller
   - Software Revision
   - Name of Project
   - Chassis Type
   - Controller Slot Location
   - Project Location

4. Click OK.

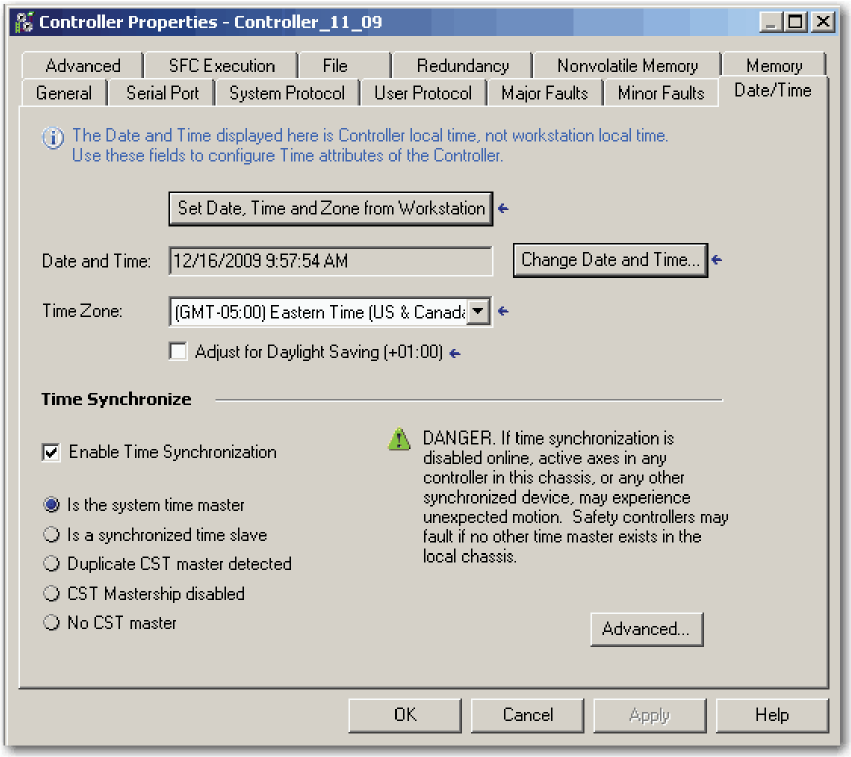
Setting Time Synchronization

Time Synchronization in ControlLogix is called CIP Sync. CIP Sync is a layer of functionality that Rockwell Automation has developed on top of the IEEE 1588 PTP protocol. CIP Sync lets you maintain accurate time synchronization of your automation solutions.

This setting establishes the module to participate in time synchronization. In systems with multiple processors all controllers need to have time synchronization enabled if they are to use CST/PTP time. The 1756-EN*x*T communication modules win the arbitration over any processor.

1. In the Controller Organizer, right-click the controller and choose Properties or double-click the controller.

2. Click the Date/Time tab.



The Date and Time displayed here is Controller local time, not workstation local time.
Use these fields to configure Time attributes of the Controller.

Date and Time: 12/16/2009 9:57:54 AM

Time Zone: (GMT-05:00) Eastern Time (US & Canad

☐ Adjust for Daylight Saving (+01:00)

☑ Enable Time Synchronization

● Is the system time master
○ Is a synchronized time slave
○ Duplicate CST master detected
○ CST Mastership disabled
○ No CST master

DANGER. If time synchronization is disabled online, active axes in any controller in this chassis, or any other synchronized device, may experience unexpected motion. Safety controllers may fault if no other time master exists in the local chassis.

3. Check Enable Time Synchronization.

4. Click OK.

Without intervention, the Grandmaster is both PTP and CST master. You can use the settings on the Advanced Dialog box to let this module to win the arbitration over other processors and communication modules in the chassis.

See the Integrated Architecture and CIP Sync Configuration Application Technique, publication IA-AP003 for more information.

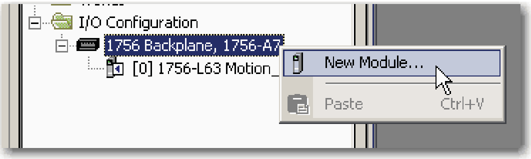
Add an Analog Interface Module

Add interface modules to the I/O configuration of the controller.

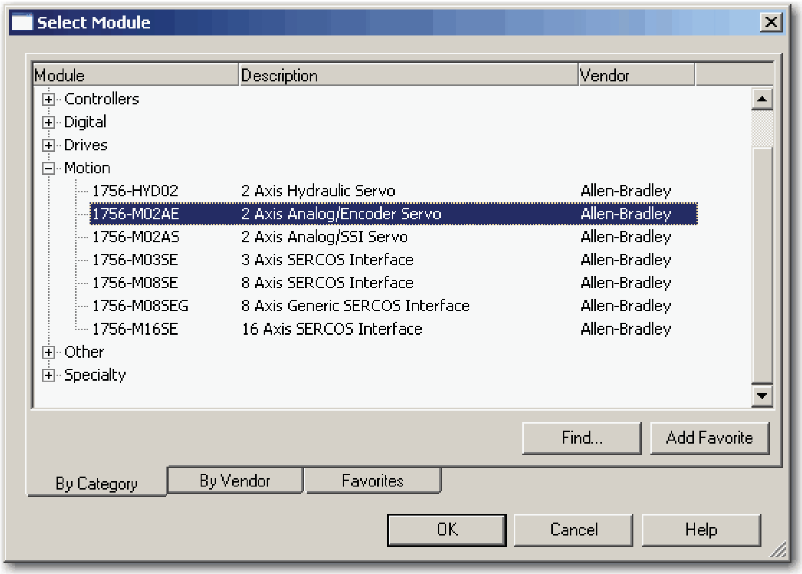
IMPORTANT For all interface drives, use the firmware revision that goes with the firmware revision of your controller. See the release notes for your controller's firmware.

---

1. To add a module, right-click the backplane and choose New Module.

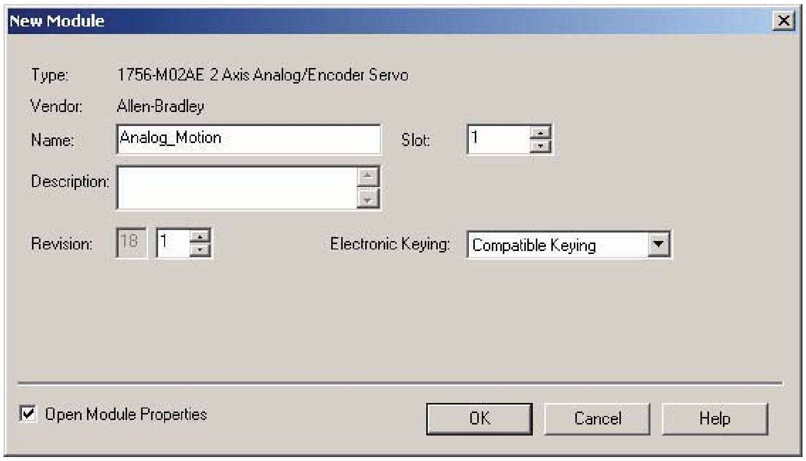


2. Select the analog module that you want to add and click OK.



Module	Description	Vendor
Controllers		
Digital		
Drives		
Motion		
1756-HYD02	2 Axis Hydraulic Servo	Allen-Bradley
1756-M02AE	2 Axis Analog/Encoder Servo	Allen-Bradley
1756-M02AS	2 Axis Analog/SSI Servo	Allen-Bradley
1756-M03SE	3 Axis SERCOS Interface	Allen-Bradley
1756-M08SE	8 Axis SERCOS Interface	Allen-Bradley
1756-M08SEG	8 Axis Generic SERCOS Interface	Allen-Bradley
1756-M16SE	16 Axis SERCOS Interface	Allen-Bradley
Other		
Specialty		

3. Name the module.



Type: 1756-M02AE 2 Axis Analog/Encoder Servo
Vendor: Allen-Bradley
Name: Analog_Motion
Slot: 1
Description:
Revision: 18 1
Electronic Keying: Compatible Keying
☑ Open Module Properties

4. Assign a slot for the module.

5. Type a description.

6. Choose a Keying option.

WARNING

When using motion modules the electronic keying should be either 'Exact Match' or 'Compatible Keying'.

'Disable Keying' should **never** be used with motion module.

See Electronic Keying on page 23.

7. Check Open Module Properties, if you want the Properties dialog box box to open automatically after you are done creating the new module.

   If you do not check the box, you will need to either double-click the module in the Controller Organizer or right-click the module and choose Properties.

Modify Properties for Analog Modules

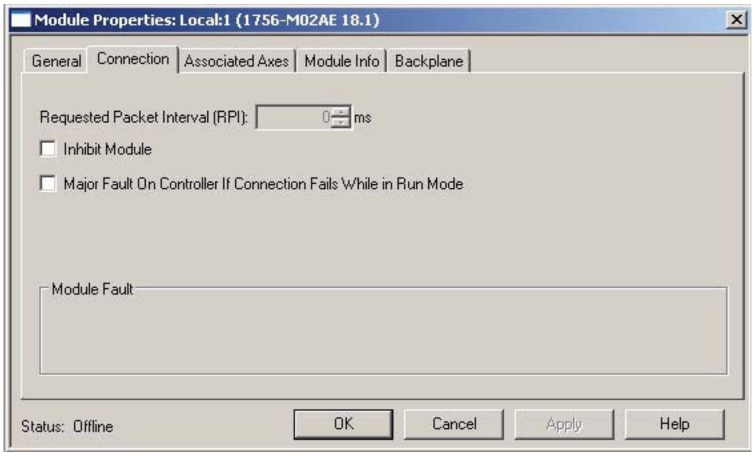
With an analog module, the main items that you need to configure from the Module Properties dialog box are:

- create an associated axis.
- assign the axis to a motion group.
- select an external drive configuration.

Follow these instructions to modify the properties for an analog module.

1. Double-click the 1756-M02AE module in the Controller Organizer.

The Module Properties dialog box appears.



Module Properties: Local:1 (1756-M02AE 18.1)

Requested Packet Interval (RPI): 0 ms

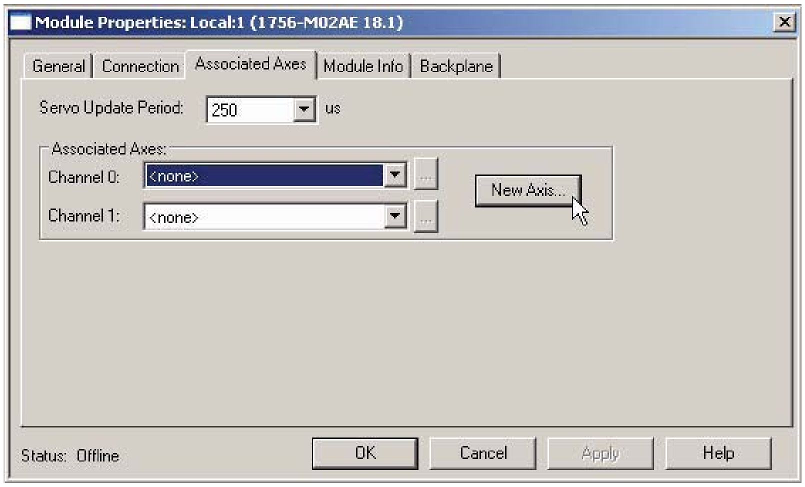
☐ Inhibit Module

☐ Major Fault On Controller If Connection Fails While in Run Mode

Module Fault

Status: Offline

2. Click the Associated Axes tab.



Module Properties: Local:1 (1756-M02AE 18.1)

Servo Update Period: 250 us

Associated Axes:

Channel 0: <none>

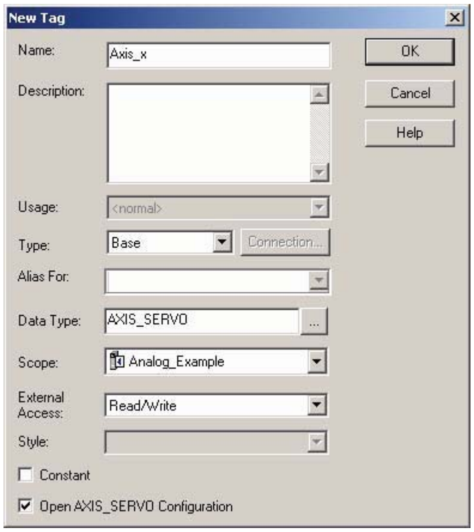
Channel 1: <none>

Status: Offline

3. Click New Axis to create an axis that will be associated to this module.

   Because you selected New Axis from the analog module dialog, the data type is defined. If you create the axis by right clicking on the Motion Group, you will need to choose the correct axis type.

1. Type a tag name for the Axis and click OK.



New Tag

Name: Axis_x

Description:

Usage: <normal>

Type: Base

Alias For:

Data Type: AXIS_SERVO

Scope: Analog_Example

External Access: Read/Write

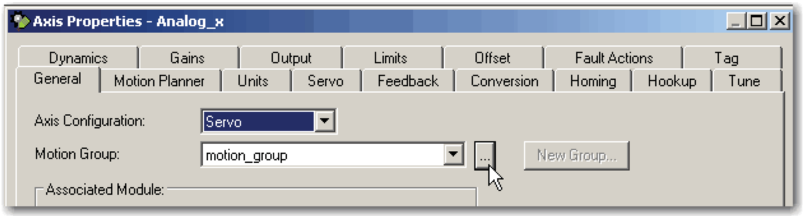
Style:

☐ Constant

☑ Open AXIS_SERVO Configuration

If you check Open AXIS_SERVO Configuration, the Axis Wizard appears and takes you through axis configuration used in motion planning.

2. Click Browse to go to the Motion Group.



Axis Properties - Analog_x

Axis Configuration: Servo

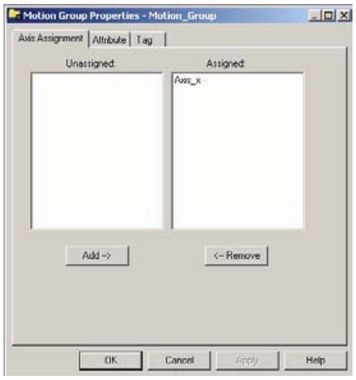
Motion Group: motion_group

Associated Module:

3. If you have already created your Motion Group, associate the axis to your group. You can also do this by dragging the axis into the Motion Group in the Controller Organizer.

   If you have not created a Motion Group, see Add the Motion Group on page 57.

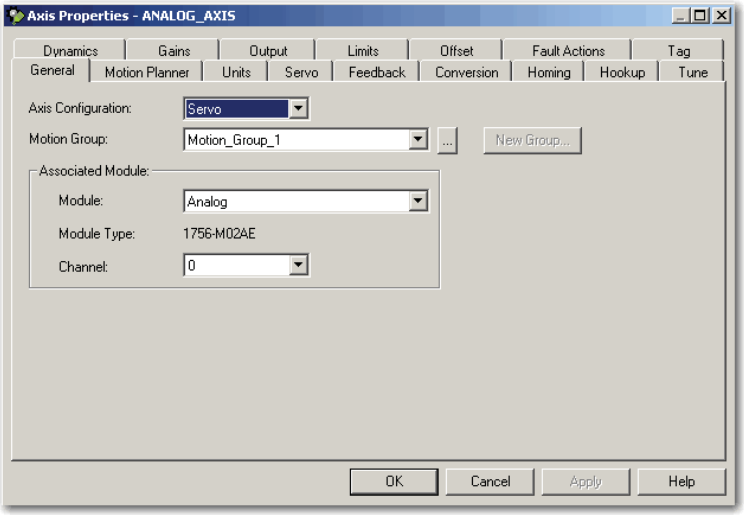
4. Assign the axis to a Motion Group, select the unassigned axis, click Add, and OK.



Unassigned:

Assigned: Axis_x

5. Under Associated Module, select the module.



Axis Configuration: Servo

Motion Group: Motion_Group_1

Associated Module:

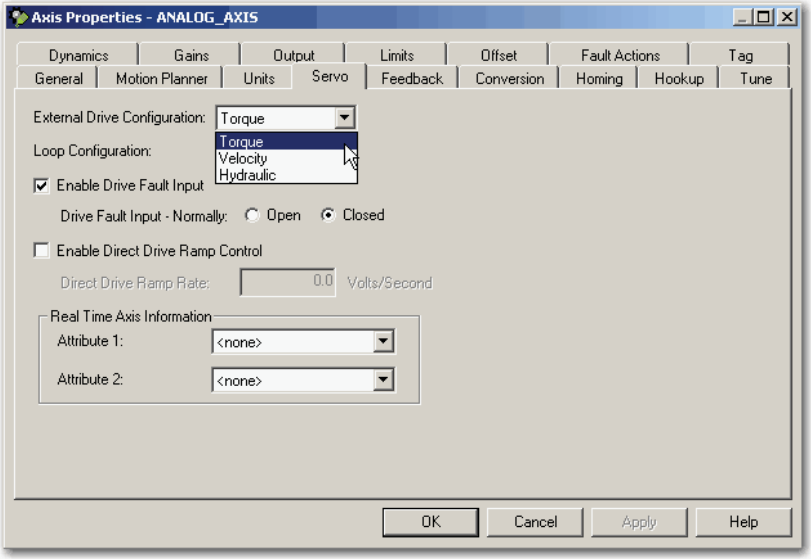
Module: Analog

Module Type: 1756-M02AE

Channel: 0

6. Click the Servo tab.

7. Assign an External Drive Configuration and click OK.



External Drive Configuration: Torque

Loop Configuration:

☑ Enable Drive Fault Input

Drive Fault Input - Normally: ○ Open ● Closed

☐ Enable Direct Drive Ramp Control

Direct Drive Ramp Rate: 0.0 Volts/Second

Real Time Axis Information

Attribute 1: <none>

Attribute 2: <none>

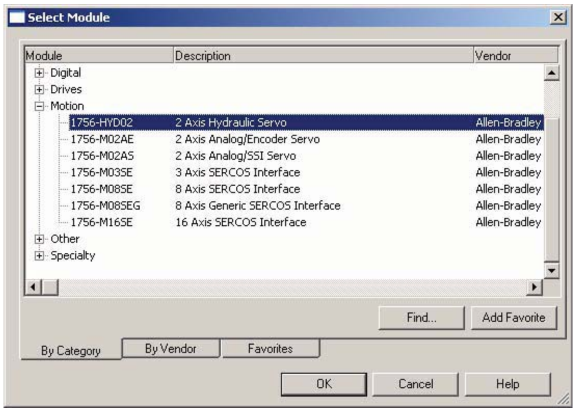
TIP If you configure a torque drive then your drive must be able to be configured for torque. Hydraulic can only be selected if it's a hydraulic module.

Add an Hydraulic Module

Follow these instructions to add an hydraulic module, if desired.

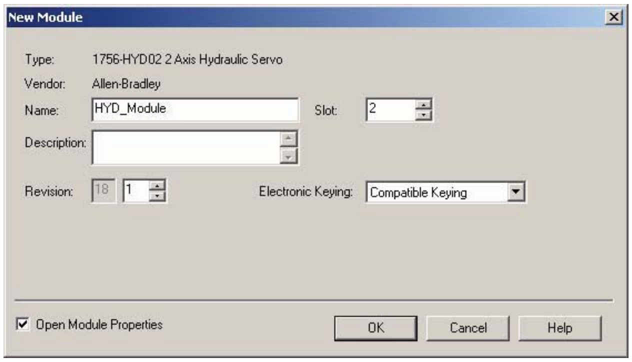
IMPORTANT For all interface module, use the firmware revision that goes with the firmware revision of your controller. See the release notes for your controller's firmware.

1. To add a module, right-click the backplane and choose New Module.
2. Select the module that you want to add and click OK.



Module	Description	Vendor
⊞ Digital		
⊞ Drives		
⊟ Motion		
1756-HYD02	2 Axis Hydraulic Servo	Allen-Bradley
1756-M02AE	2 Axis Analog/Encoder Servo	Allen-Bradley
1756-M02AS	2 Axis Analog/SSI Servo	Allen-Bradley
1756-M03SE	3 Axis SERCOS Interface	Allen-Bradley
1756-M08SE	8 Axis SERCOS Interface	Allen-Bradley
1756-M08SEG	8 Axis Generic SERCOS Interface	Allen-Bradley
1756-M16SE	16 Axis SERCOS Interface	Allen-Bradley
⊞ Other		
⊞ Specialty		

3. Name the module.



Type: 1756-HYD02 2 Axis Hydraulic Servo
Vendor: Allen-Bradley
Name: HYD_Module
Slot: 2
Description:
Revision: 18 1
Electronic Keying: Compatible Keying
☑ Open Module Properties

4. Assign a slot for the module.
5. Type a description, if desired
6. Choose a Keying option.

WARNING When using motion modules the electronic keying should be either 'Exact Match' or 'Compatible Keying'.

'Disable Keying' should **never** be used with motion module.

See Electronic Keying on page 23.

7. Check Open Module Properties, if you want the Properties dialog box box to open automatically after you are done creating the new module.

   If you do not check the box, you will need to either double-click the module in the Controller Organizer or right-click the module and choose Properties.

Modify Properties for Hydraulic Drives

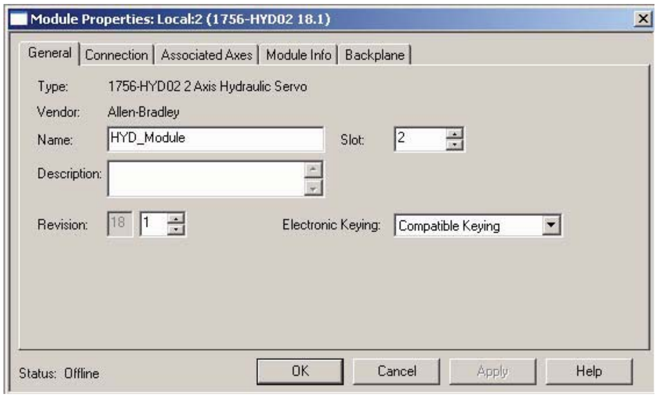
With a hydraulic drive, the main item that you need to configure is the feedback type.

Based on the length of the feedback, the Servo Update Period should be configured. This is unique for the 1756-HYD02 module. If the Servo Update Period is not configured correctly, the axis will not work.

Follow these instructions to modify the properties for an analog drive.

1. Double-click the 1756-HYD02 module in the Controller Organizer.

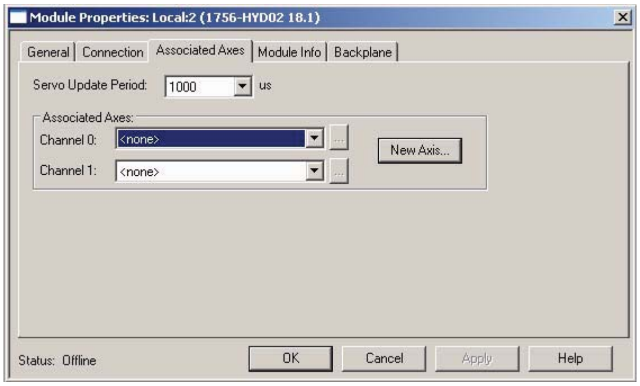
   The Module Properties dialog box appears.



Module Properties: Local:2 (1756-HYD02 18.1)

Type: 1756-HYD02 2 Axis Hydraulic Servo
Vendor: Allen-Bradley
Name: HYD_Module
Slot: 2
Description:
Revision: 18 1
Electronic Keying: Compatible Keying
Status: Offline

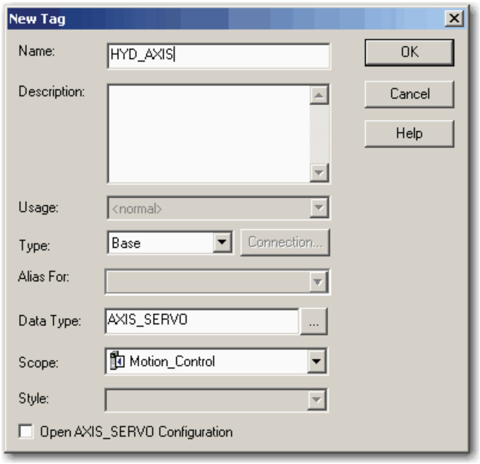
2. Click the Associated Axis tab.



Module Properties: Local:2 (1756-HYD02 18.1)

Servo Update Period: 1000 us
Associated Axes:
Channel 0: <none>
Channel 1: <none>
Status: Offline

3. Click New Axis to create a new axis that will be associated to this module.



Name: HYD_AXIS

Description:

Usage: <normal>

Type: Base

Alias For:

Data Type: AXIS_SERVO

Scope: Motion_Control

Style:

☐ Open AXIS_SERVO Configuration

Notice that the Data Type has been automatically filled in for you.

4. Type a tag name for the axis and click OK.

If you check the Open AXIS_SERVO Configuration check box, the Axis Wizard appears walks you through complete axis configuration used in Motion Planning.

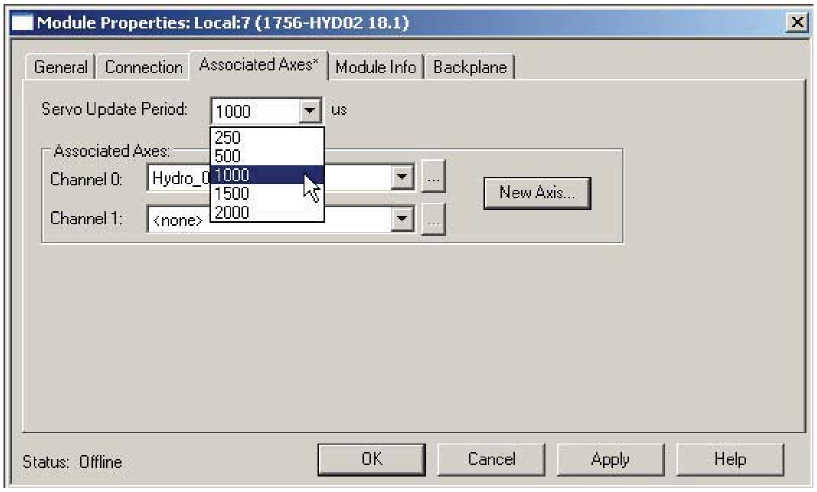
Servo Update Rate

Follow these instructions to set the Course Update Period that configures the Motion Planner.

1. Determine what your Coarse Update Period should be based on the information on page 58.

TIP A general guideline for the course rate is 4 axes/ms for a 1756-L6*x* controller and 8 axes/ms for a 1756-L7*x* controller.

2. Go to the Associated Axis tab.
3. Set the Servo Update Period.



Module Properties: Local:7 (1756-HYD02 18.1)

Servo Update Period: 1000 us

Associated Axes:

Channel 0: Hydro_0

Channel 1: <none>

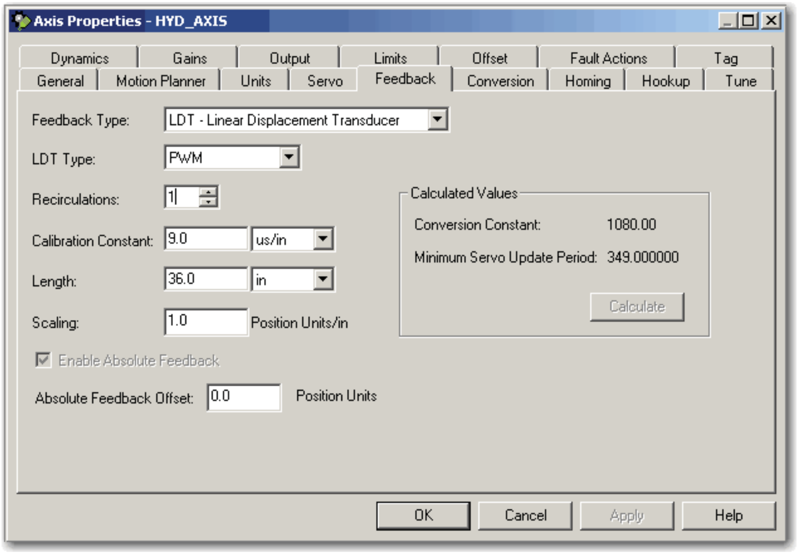
Status: Offline

Configuring the Feedback Type

1. Double-click the axis and go to the Feedback tab.

   Just changing the Feedback Type is not enough for a hydraulic axis. The rest of the feedback information must be configured.

For example, the Calibration Constant is printed on the actual feedback device and may also be available in the documentation that comes with the hardware.



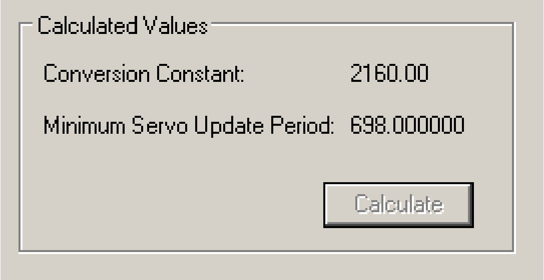
Feedback Type: LDT - Linear Displacement Transducer
LDT Type: PWM
Recirculations: 1
Calibration Constant: 9.0 us/in
Length: 36.0 in
Scaling: 1.0 Position Units/in
☑ Enable Absolute Feedback
Absolute Feedback Offset: 0.0 Position Units

Calculated Values
Conversion Constant: 1080.00
Minimum Servo Update Period: 349.000000

2. Configure the Feedback type.

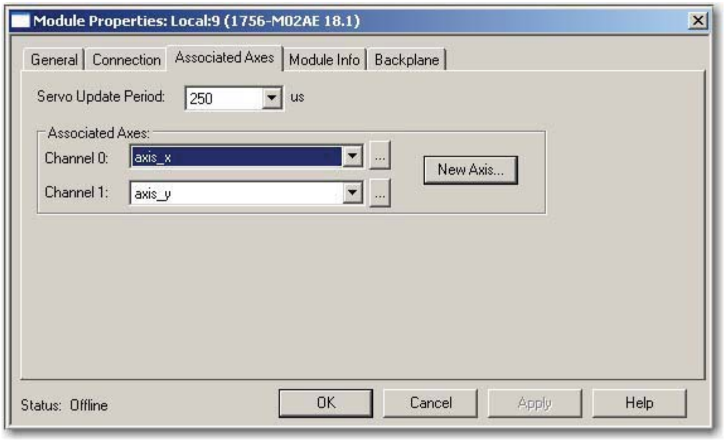
3. Set the Calibration Constant and click Calculate.

   This is the minimum servo update period for the configured feedback.



Calculated Values
Conversion Constant: 2160.00
Minimum Servo Update Period: 698.000000

4. You now have to manually update this value in the Associated Axis tab.



Servo Update Period: 250 us

Associated Axes:
Channel 0: axis_x
Channel 1: axis_y

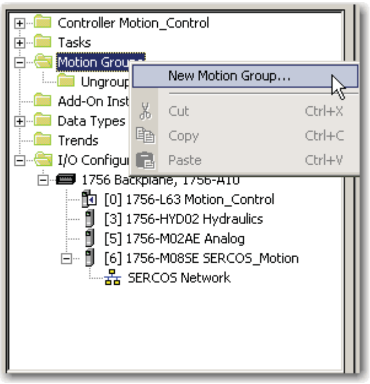
Status: Offline

Add the Motion Group

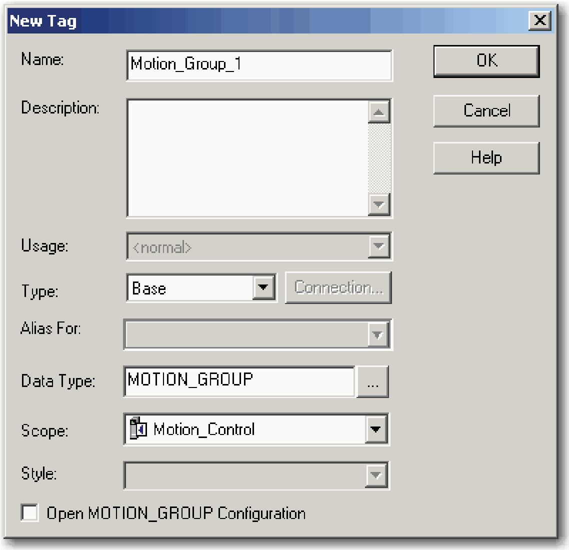
Follow these instructions to add a motion group.

IMPORTANT Only one motion group can be created for each project.

1. Right-click Motion Groups and choose New Motion Group.



2. Type a name and description and click OK.


New Tag

Name: Motion_Group_1

Description:

Usage: <normal>

Type: Base

Alias For:

Data Type: MOTION_GROUP

Scope: Motion_Control

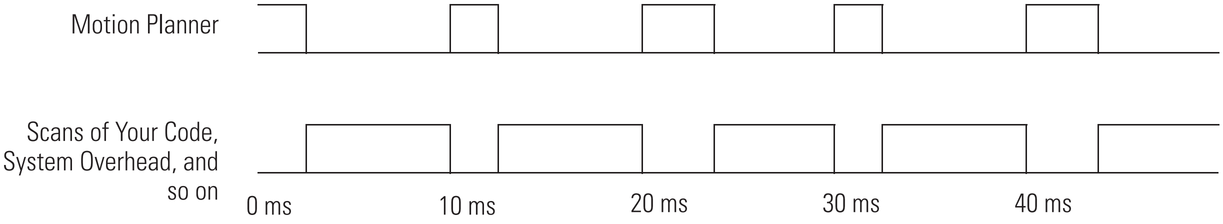
Style:

☐ Open MOTION_GROUP Configuration

If you check Open Motion_Group Configuration, the Motion Group Wizard appears. If not you can access the Motion Group properties by right-clicking the group and choosing Properties.

Course Update Period

The Course Update Period is how often the motion planner runs. When the motion planner runs, it interrupts most other tasks regardless of their priority. The motion planner is the part of the controller that takes care of position and velocity information for the axes.

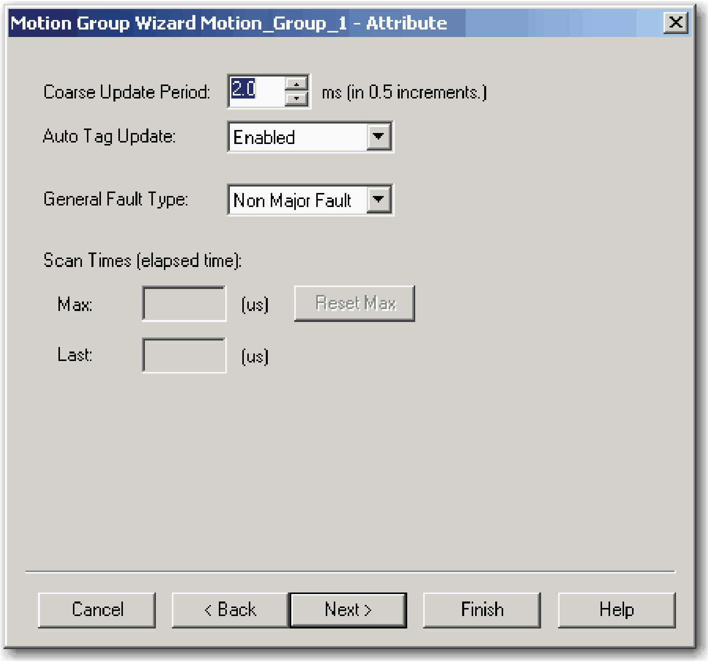


In this example, the Coarse Update Period = 10 ms. Every 10 ms the controller stops scanning your code and whatever else it is doing and runs the motion planner.

Follow these instructions to set the Course Update Period that configures the motion planner.

TIP A general guideline for the course rate is 4 axes/ms for a 1756-L6*x* controller and 8 axes/ms for a 1756-L7*x* controller.

1. Select the Coarse Update Period and click Finish.



Motion Group Wizard Motion_Group_1 - Attribute

Coarse Update Period: 2.0 ms (in 0.5 increments.)

Auto Tag Update: Enabled

General Fault Type: Non Major Fault

Scan Times (elapsed time):

Max: (us)

Last: (us)

2. Choose the Coarse Update Period.

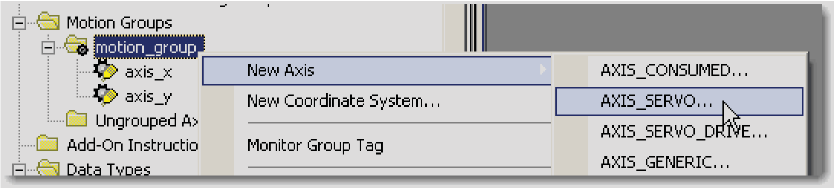
The Coarse Update Period is a trade-off between updating positions of your axes and scanning your code. Use these guidelines as a rough starting point.

Guideline	**Description**
Number of Axes	1756-L6*x* controller 4 axes/ms 1756-L7*x* controller 8 axes/ms
Save Controller's Time	Leave at least half the controller's time for the scan of all your code.
Coarse Update Period and SERCOS modules	If you have SERCOS interface motion modules, set the Coarse Update Period to a multiple of the cycle time of the motion module. Example: if the cycle time is 2 ms, set the Coarse Update Period to 8 ms, 10 ms, 12ms, and so on.
Coarse Update Period and Analog modules	If you have analog motion modules, set the Coarse Update Period to: • at least 3 times the servo update period of the motion module. • a multiple of the servo update period of the motion module.

Add an Axis

Add an axis for each of your drives.

1. Right-click Motion Group and choose New Axis.



If you use one of these motion modules then use the data type AXIS_SERVO for the axis.

- 1756-M02AE
- 1756-HYD02
- 1756-M02AS

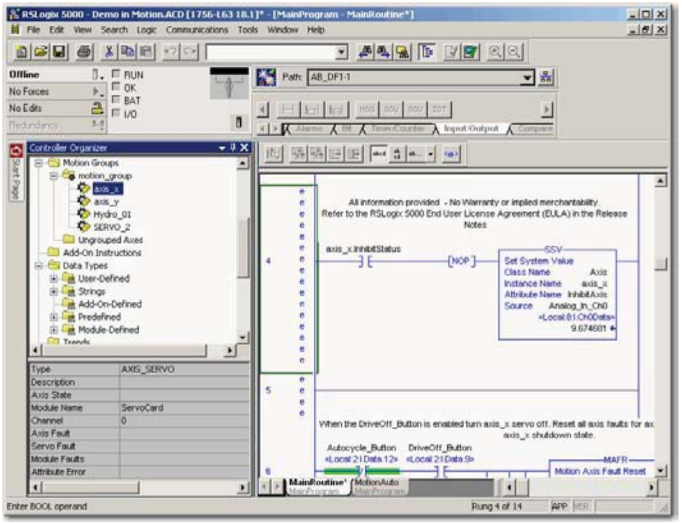
2. Choose AXIS_SERVO.

If you want to use a virtual configuration (no hardware) then use the data type AXIS_VIRTUAL for the axis.

Get Axis Information

You can get information about an axis in several ways.

- Use the Quick View pane to see the state and faults of an axis.
- Use a Get System Value (GSV) instruction or Set System Value (SSV) instruction to read or change the configuration at run-time.
- Use the tag of the axis for status and faults.



Type	AXIS_SERVO
Description	
Axis State	
Module Name	ServoCard
Channel	0
Axis Fault	
Servo Fault	
Module Faults	
Attribute Error	

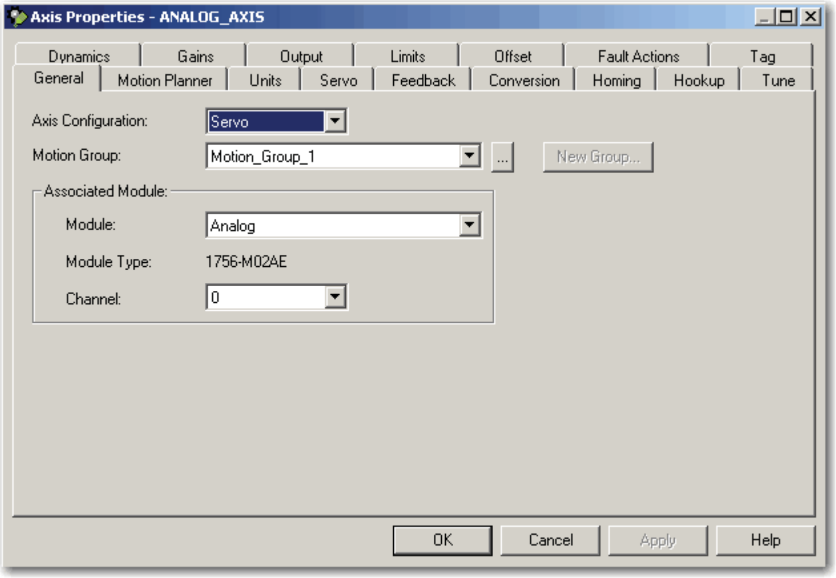
All information provided - No Warranty or implied merchantability. Refer to the RSLogix 5000 End User License Agreement (EULA) in the Release Notes

When the DriveOff_Button is enabled turn axis_x servo off. Reset all axis faults for ax axis_x shutdown state.

Configure an Axis

Follow these steps to set up the axis of an analog module.

1. Right-click the axis and choose Properties.
2. Click the General tab.
3. Select the drive for the axis.
4. Select the name that you gave to the drive for this axis.



Axis Configuration: Servo

Motion Group: Motion_Group_1

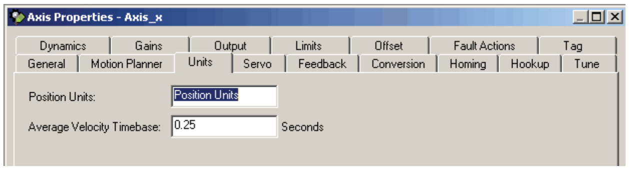
Associated Module:

Module: Analog

Module Type: 1756-M02AE

Channel: 0

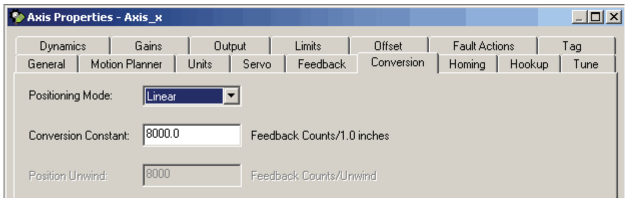
5. Set the units that you want to program in.



Position Units: Position Units

Average Velocity Timebase: 0.25 Seconds

6. On the Conversion tab, select the Positioning Mode and enter the Conversion Constant.



Positioning Mode: Linear

Conversion Constant: 8000.0 Feedback Counts/1.0 inches

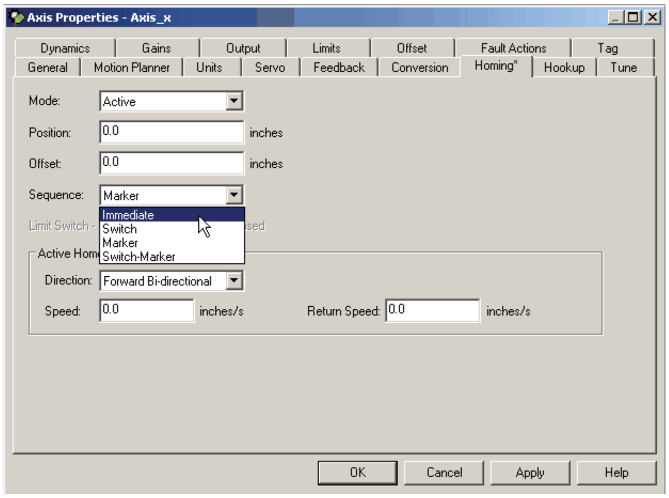
Position Unwind: 8000 Feedback Counts/Unwind

Set the Homing Sequence

For complete information about Homing modes, methods and guidelines, see Home an Axis on page 119.

Follow these instructions to set the homing sequence.

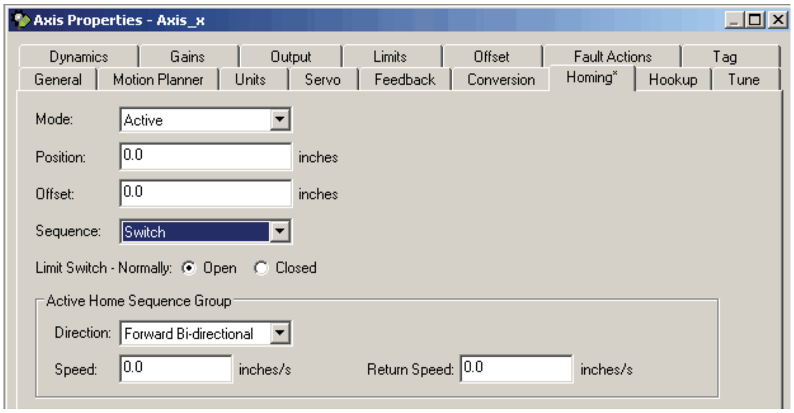
1. On the Homing tab, select the homing mode.
2. Enter the Position units.
3. Select a Sequence type.



Mode: Active
Position: 0.0 inches
Offset: 0.0 inches
Sequence: Marker
Limit Switch - ... sed
Active Hom...
Direction: Forward Bi-directional
Speed: 0.0 inches/s
Return Speed: 0.0 inches/s

Homing speeds appear dimmed depending on what Homing Mode and Sequence Type you select.

4. For all Sequence types except for Immediate, you need to select the Active Home Sequence Group and the speeds.



Mode: Active
Position: 0.0 inches
Offset: 0.0 inches
Sequence: Switch
Limit Switch - Normally: ● Open ○ Closed
Active Home Sequence Group
Direction: Forward Bi-directional
Speed: 0.0 inches/s
Return Speed: 0.0 inches/s

Commission and Tune

Introduction

This chapter discusses how to commission an axis for a motion application.

Download a Project to a Controller

Follow these steps to download your program to a controller.

1. With the keyswitch, place the controller in program or remote program mode.
2. From the Communications menu, choose Download.
3. Confirm that you wish to complete the download procedure.
4. Click Download.
5. Once the download is complete, place the controller in Run/Test mode.

After the project file is downloaded, status and compiler messages appear in the status bar.

TIP

When multiple workstations connect to the same controller using RSLogix 5000 software and invoke the Axis Wizard or Axis Properties dialog box, the firmware lets only the first workstation make any changes to axis attributes. The second workstation switches to a Read Only mode, indicated in the title bar, so that you may view the changes from that workstation, but not edit them.

If an axis in a motion group is open for edit, then any other workstation will only get read-only for any axis in that workstation. Even if it is not the axis that the first workstation is modifying.

Test Axis Wiring and Direction

Use the following procedure to check the wiring of each drive.

Test	Description
Test marker	Checks that the drive gets the marker pulse. You must manually move the axis for this test.
Test feedback	Checks the polarity of the feedback. You must manually move the axis for this test.
Test command and feedback	Checks the polarity of the drive.

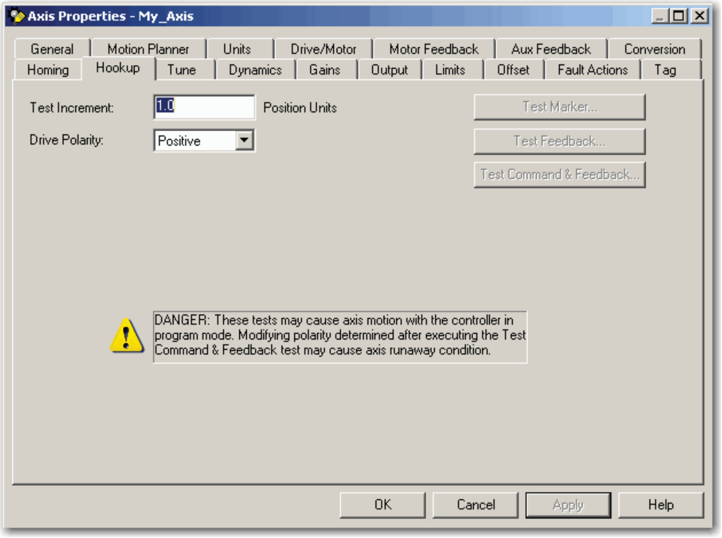
ATTENTION

These tests make the axis move even with the controller in remote program mode.

- Before you do the tests, make sure no one is in the way of the axis.
- **Do not** change the polarity after you do the tests. Otherwise you may cause an axis-runaway condition.

1. Download a program to the controller.
2. Place controller in REM.
3. Double-click the axis to open the Axis Properties.

4. Click the Hookup tab.



Test Increment: 1.0 Position Units

Drive Polarity: Positive

DANGER: These tests may cause axis motion with the controller in program mode. Modifying polarity determined after executing the Test Command & Feedback test may cause axis runaway condition.

IMPORTANT Make sure to follow all the dialogs or the information derived from the test will not be saved to the axis configuration.

5. Click Test Marker and follow the additional dialog box instructions.

6. Click Test Feedback and follow the additional dialog box instructions.

7. Click Test Command & Feedback and follow the additional dialog box instructions.

Tune a SERCOS Axis

Use the Tune tab to tune an axis.

ATTENTION When you tune an axis, it moves even with the controller in remote program mode. In that mode, your code is **not** in control of the axis.

Before you tune an axis, make sure no one is in the way of the axis.

The default tuning procedure tunes the proportional gains. Typically, tune the proportional gains first and see how your equipment runs.

TIP Where tighter positioning is required, Integral gain, and feedforward constants can be selected. However, when used incorrectly can cause axis instability.

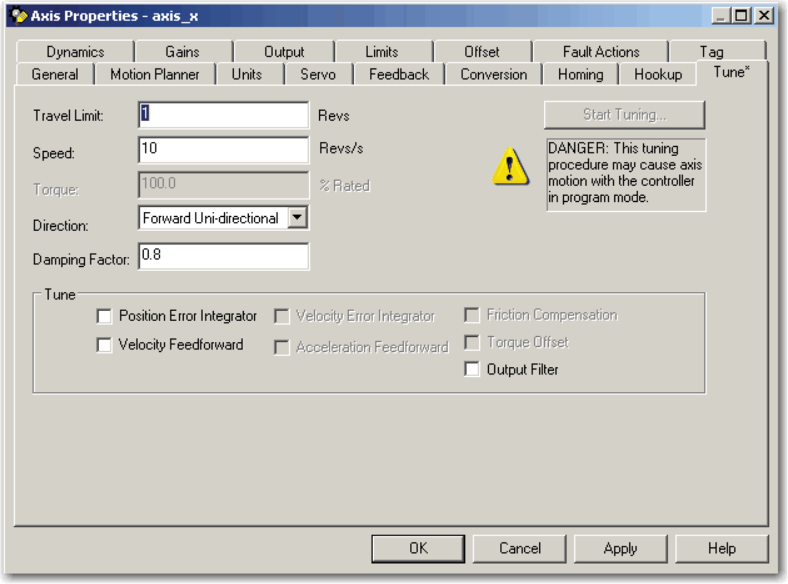
See Tune on page 175.

1. Download a program to the controller.

2. Place controller in REM.

3. Double-click the axis.

4. Click the Tune tab.



Travel Limit: 1 Revs

Speed: 10 Revs/s

Torque: 100.0 % Rated

Direction: Forward Uni-directional

Damping Factor: 0.8

DANGER: This tuning procedure may cause axis motion with the controller in program mode.

Tune

☐ Position Error Integrator ☐ Velocity Error Integrator ☐ Friction Compensation

☐ Velocity Feedforward ☐ Acceleration Feedforward ☐ Torque Offset

☐ Output Filter

5. Set the Travel Limit.

   This is the limit of movement for the axis during the tuning procedure.

6. Set the Speed.

   This is the maximum speed for your equipment.

7. Click Start Tuning.

8. Accept the changes.

   You must accept the changes or the data derived from the tune will not be saved as part of the axis configuration.

Tune an Analog Axis

Use the Tune tab to tune an axis.

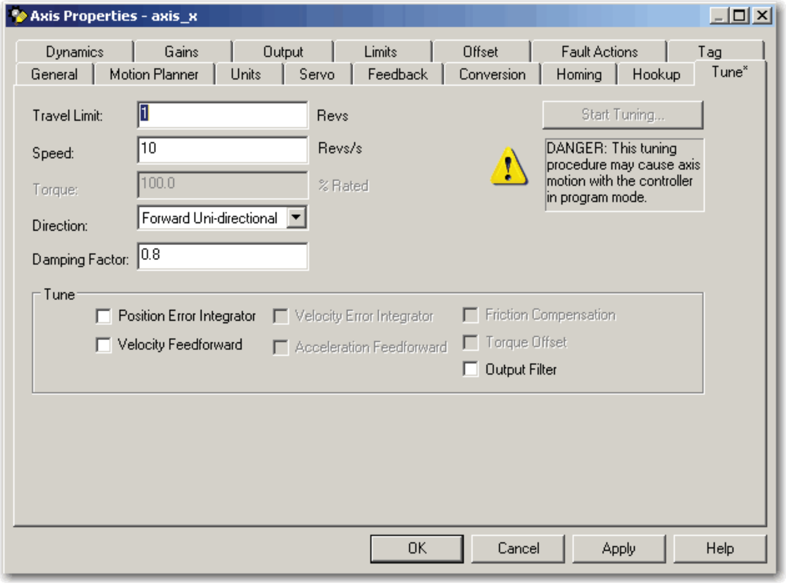
ATTENTION

When you tune an axis, it moves even with the controller in remote program mode. In that mode, your code is **not** in control of the axis.

Before you tune an axis, make sure no one is in the way of the axis.

The default tuning procedure tunes the proportional gains. Typically, tune the proportional gains first and see how your equipment runs.

1. Download a program to the controller.
2. Place controller in REM.
3. Select the axis.
4. Right-click the axis and choose Properties.
5. Click the Tune tab.



Travel Limit: 1 Revs

Speed: 10 Revs/s

Torque: 100.0 % Rated

Direction: Forward Uni-directional

Damping Factor: 0.8

DANGER: This tuning procedure may cause axis motion with the controller in program mode.

☐ Position Error Integrator
☐ Velocity Error Integrator
☐ Friction Compensation
☐ Velocity Feedforward
☐ Acceleration Feedforward
☐ Torque Offset
☐ Output Filter

6. Set the Travel Limit, type the limit of movement for the axis during the tuning procedure.
7. Set the Speed, type the maximum speed for your equipment.
8. Click Start Tuning.

Troubleshoot Faults

The controller has these types of motion faults.

Type	Description	Example
Instruction error	Caused by a motion instruction: • Instruction errors do not impact controller operation. • Look at the error code in the motion control tag to see why an instruction has an error. • Fix instruction errors to optimize execution time and make sure that your code is accurate.	A Motion Axis Move (MAM) instruction with a parameter out of range
Fault	Caused by a problem with the servo loop: • You choose whether or not motion faults give the controller major faults. • Can shutdown the controller if you do not correct the fault condition.	• Loss of feedback • Actual position exceeding an overtravel limit

Manage Motion Faults

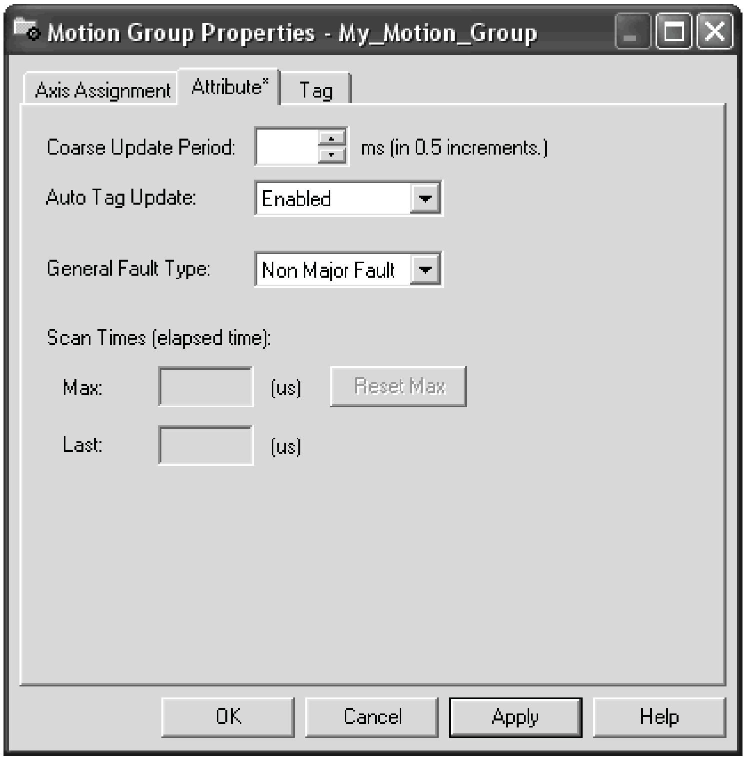
By default, the controller keeps running when there is a motion fault. As an option, you can have motion faults cause a major fault and shut down the controller.

1. Choose a Non-Major Fault.

    - If you select Major Fault, then you should develop a Fault handler.

    See Logix5000 Controllers Major and Minor Faults Programming Manual, publication 1756-PM014.

2. Right-click Motion Group and choose Properties.
3. Go to the Attribute tab.
4. Select Non Major Fault.


Motion Group Properties - My_Motion_Group

Coarse Update Period: ms (in 0.5 increments.)

Auto Tag Update: Enabled

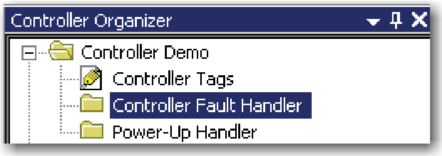
General Fault Type: Non Major Fault

Scan Times (elapsed time):

Max: (us)

Last: (us)

5. Place your programs into the Fault Handler so that when a fault occurs the program runs.



Configure the Fault Actions for an Axis

Use the fault actions to set how an axis responds to different types of faults. The type of faults depends on the type of axis and how you configure it.

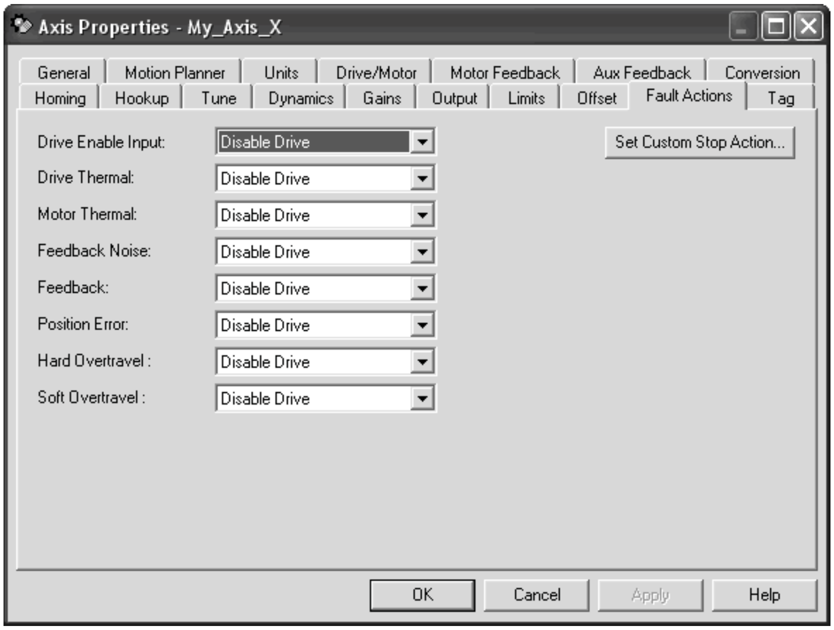
If you want to	Then choose	Description	
Shutdown the axis and let it coast to a stop	Shutdown	Shutdown is the most severe action. Use it for faults that could endanger the machine or the operator if you do not remove power quickly and completely.	
		For this axis type	**When the fault happens**
		AXIS_SERVO	• Axis servo action is disabled. • The servo amplifier output is zeroed. • The appropriate drive enable output is deactivated. • The OK contact of the servo module opens. Use this to open the E-Stop string to the drive power supply. • This impacts both axes associated with the analog motion, not just the axis with the fault.
		AXIS_SERVO_DRIVE	• Axis servo action and drive power structure are immediately disabled. • The axis coasts to a stop unless you use some form of external braking.
Disable the axis and let the drive stop the axis using it's best available stopping method	Disable Drive	**For this axis type**	**When the fault happens**
		AXIS_SERVO	• Planner decelerates axis motion to zero speed based on Maximum configured declaration using Trap Acc/Dec. • Axis servo action is off. • The servo amplifier output is zeroed. • The appropriate drive enable output is deactivated.
		AXIS_SERVO_DRIVE	• Planner decelerates axis motion to zero speed based on Maximum configured declaration using Trap Acc/Dec. • If the axis does not stop in the Stopping Time, the servo action and the power structure are disabled.
Leave the servo loop on and stop the axis at its Maximum Deceleration rate	Stop Motion	Use this fault action for less severe faults. It is the gentlest way to stop. Once the axis stops, you must clear the fault before you can move the axis.	
		For this axis type	**When the fault happens**
		AXIS_SERVO	The axis slows to a stop at the Maximum Deceleration Rate without disabling servo action or the servo module's Drive Enable output.
		AXIS_SERVO_DRIVE	• Control of the drive's servo loop is maintained. • The axis slows to a stop at the Maximum Deceleration rate without disabling the drive.
Write your own application code to handle the fault	Status Only	Use this fault action **only** when the standard fault actions are not appropriate. With this fault action, you must write code to handle the motion faults. For Stop Motion or Status Only, the drive must stay enabled for the controller to continue to control the axis. Selecting Status Only only lets motion continue if the drive itself is still enabled and tracking the command reference.	

Set the Fault Action for an Axis

Use the following steps to set the fault actions for an axis.

1. Right-click on an axis choose Properties.

2. Go to the Fault Actions tab.



Drive Enable Input: Disable Drive
Drive Thermal: Disable Drive
Motor Thermal: Disable Drive
Feedback Noise: Disable Drive
Feedback: Disable Drive
Position Error: Disable Drive
Hard Overtravel : Disable Drive
Soft Overtravel : Disable Drive

3. Set the desired attributes and click OK.

   An analog axis will have fewer faults listed than a SERCOS axis.

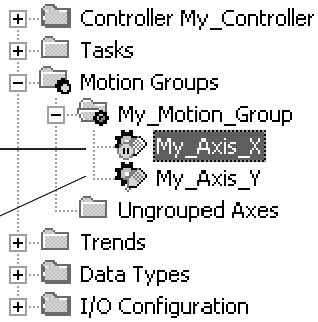
Inhibit an Axis

Follow these instructions to block the controller from using an axis.

When to Inhibit an Axis

IMPORTANT Inhibiting an axis will take down ALL axis on the motion module or ring. The non-inhibited axes will then phase back up. Un-Inhibiting an axis will cause the same thing.

Use the following information to determine when to inhibit an axis.



You want to block the controller from using an axis because the axis is faulted or not installed.

You want to let the controller use the other axes.

Example 1

If your equipment will need between 8 and 12 axes, depending on the application, set up one project for all 12 axes. When you determine how many axes you need, inhibit the axes that you don’t need.

Example 2

If you were to have two production lines that use the same SERCOS ring and one of the lines gets a fault. In that case, inhibit the axes on that line. This lets you run the other line while you take care of the fault.

TIP If an axis is faulted, then all the axes are still available. If one of the “drives” has a hardware problem. You can inhibit the axis and remove the faulty hardware. When the ring phases back up the inhibited axis (with its missing hardware) will not prevent the rest of the axes from operating.

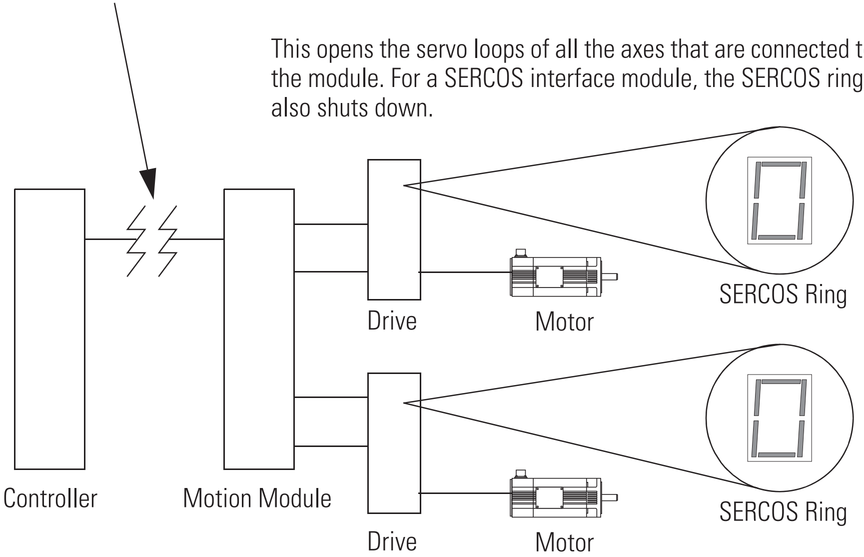
Before You Begin

Before you inhibit or uninhibit an axis, turn off all of the axes.

Follow these steps before you inhibit or uninhibit an axis.

1. Stop all motion.
2. Open the servo loops of all the axes. Use an instruction such as the Motion Servo Off (MSF) instruction.

This lets you stop motion under your control. Otherwise the axes turn off on their own when you inhibit or uninhibit one of them.



The connections to the motion module shut down when you inhibit or uninhibit an axis.

This opens the servo loops of all the axes that are connected to the module. For a SERCOS interface module, the SERCOS ring also shuts down.

The controller automatically restarts the connections. The SERCOS ring also phases back up.

Inhibit only certain types of axes.

You can inhibit only these types of axes:

- AXIS_SERVO
- AXIS_SERVO_DRIVE

To inhibit all of the axes of a motion module, inhibit the module instead.

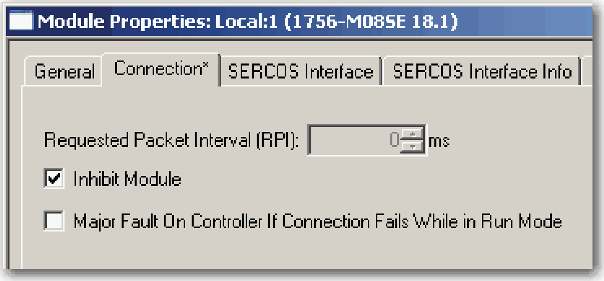
Do you want to inhibit all of the axes of a motion module?

- YES — Inhibit the motion module instead.
- NO — Inhibit the individual axes.

  You can inhibit all of the axes of a module on an individual basis. However, it is more efficient to inhibit all axes at once by inhibiting the module.

  Important: If you inhibit an axis on a drive, you inhibit all action on the drive, including any half axes. Make sure you are aware of all action on a drive before inhibiting the axis.

  Example: Suppose your motion module has two axes and you want to inhibit both of those axes. In that case, just inhibit the module.



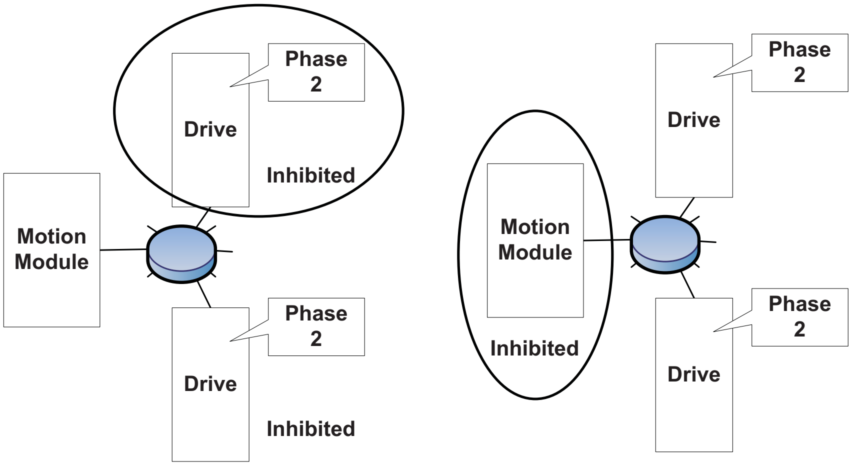
Module Properties: Local:1 (1756-M08SE 18.1)

Requested Packet Interval (RPI): 0 ms

☑ Inhibit Module

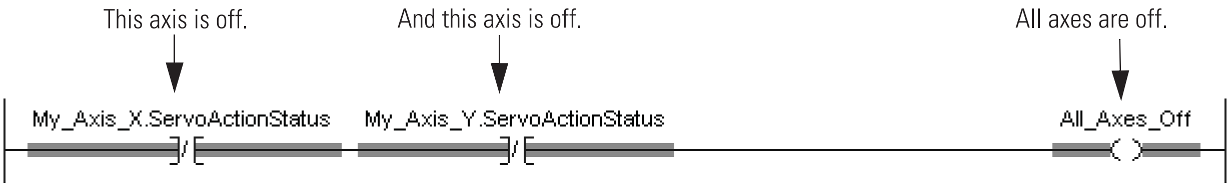
☐ Major Fault On Controller If Connection Fails While in Run Mode

If you inhibit all of the axes on a SERCOS ring, the drives phase up to phase 2. This happens whether you inhibit all the axis individually or you inhibit the motion module.

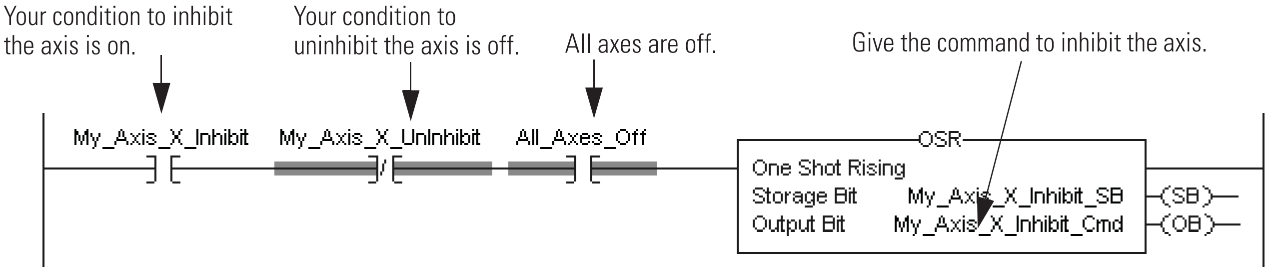


Example: Inhibit an Axis

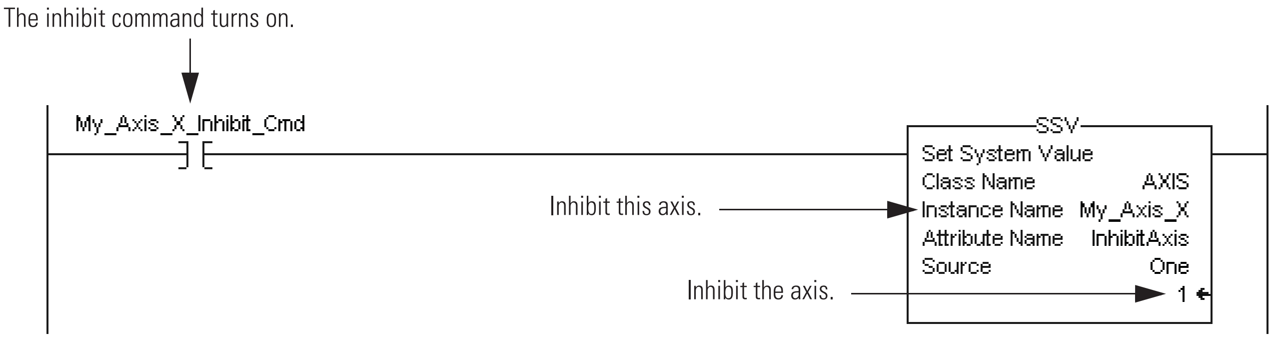
1. **Make sure all axes are off.**



2. **Use a one-shot instruction to trigger the inhibit.**



3. **Inhibit the axis.**



4. **Wait for the inhibit process to finish.**

   All of these have happened:
   - The axis is inhibited.
   - All uninhibited axes are ready.
   - The connections to the motion module are running again.
   - For a SERCOS ring, the SERCOS ring has phased up again.

Example: Uninhibit an Axis

1. Make sure all axes are off.

This axis is off.

And this axis is off.

All axes are off.

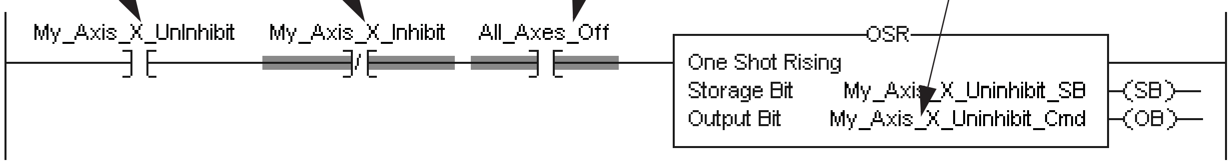
2. Use a one-shot instruction to trigger the uninhibit.

Your condition to uninhibit the axis is on.

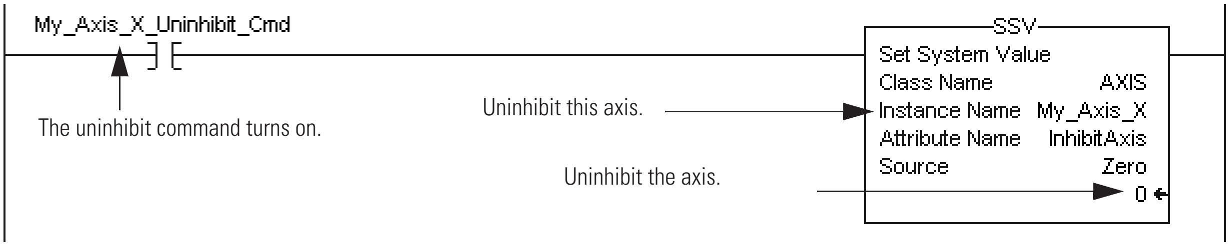
Your condition to inhibit the axis is off.

All axes are off.

Give the command to uninhibit the axis.



3. Uninhibit the axis.



The uninhibit command turns on.

Uninhibit this axis.

Uninhibit the axis.

4. Wait for the inhibit process to finish.

All of these have happened:

- The axis is uninhibited.
- All uninhibited axes are ready.
- The connections to the motion module are running again.
- For a SERCOS ring, the SERCOS ring has phased up again.

This axis is OK to run.

Test an Axis with Motion Direct Commands

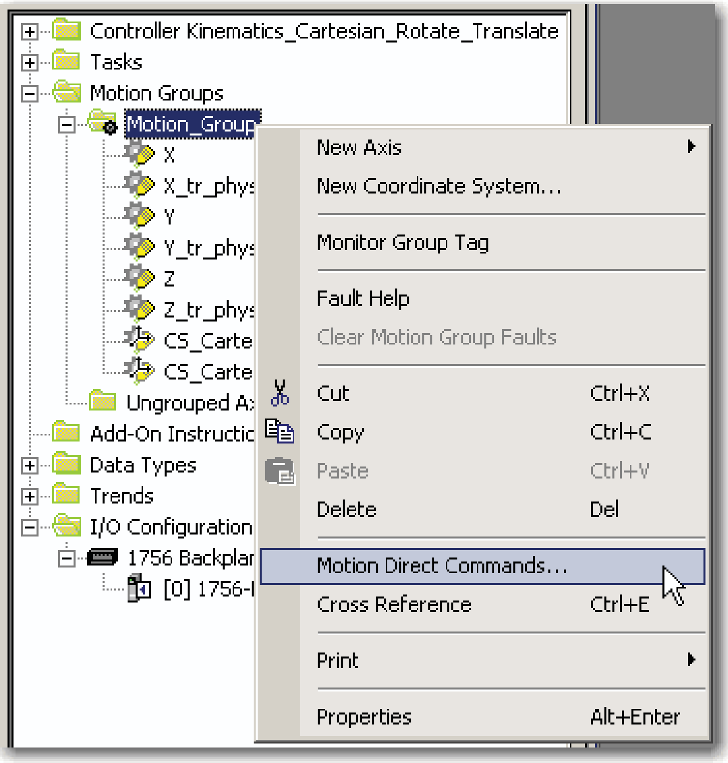
Motion direct commands let you issue motion commands while you are online without having to write or execute an application program.

Motion Direct Commands are particularly useful when you are commissioning or debugging a motion application. During commissioning, you can configure an axis and monitor the behavior using Trends in the Controller Organizer. Use of Motion Direct Commands can fine-tune the system with or without load to optimize its performance. When in the testing and or debugging cycle, you can issue Motion Direct Commands to establish or reestablish conditions such as Home. Often during initial development or enhancement to mature applications you need to test the system in small manageable areas. These tasks include:

- Home to establish initial conditions
- Incrementally Move to a physical position
- Monitor system dynamics under specific conditions

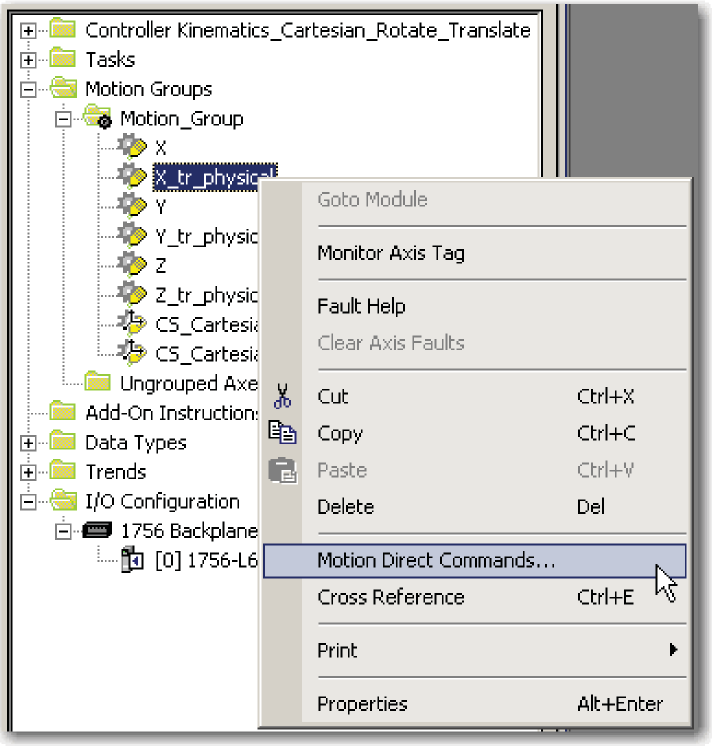
Access Motion Direct Commands for a Group

To access the Motion Direct Commands for the motion group, right- click the group in the Controller Organizer.



Access the Motion Direct Commands for an Axis

To access the Motion Direct Commands for an axis, right-click the axis in the Controller Organizer.



Choose a Command

Use this table to choose an instruction and see if it is available as a Motion Direct Command.

If you want to	And	Use this instruction	Motion Direct Command
Change the state of an axis	Enable the servo drive and activate the axis servo loop.	MSO Motion Servo On	Yes
	Turn off the servo drive and deactivate the axis servo loop.	MSF Motion Servo Off	Yes
	Force an axis into the shutdown state and block any instructions that initiate axis motion.	MASD Motion Axis Shutdown	Yes
	Transition an axis to the ready state. If all of the axes of a servo module are removed from the shutdown state as a result of this instruction, the OK relay contacts for only an analog module close.	MASR Motion Axis Shutdown Reset	Yes
	Enable the servo drive and set the servo output voltage of an axis.	MDO Motion Direct Drive On	Yes
	Turn off the servo drive and set the servo output voltage to the output offset voltage.	MDF Motion Direct Drive Off	Yes
	Clear all motion faults for an axis.	MAFR Motion Axis Fault Reset	Yes
Control axis position	Stop any motion process on an axis.	MAS Motion Axis Stop	Yes
	Home an axis.	MAH Motion Axis Home	Yes
	Jog an axis.	MAJ Motion Axis Jog	Yes
	Move an axis to a specific position.	MAM Motion Axis Move	Yes
	Start electronic gearing between 2 axes.	MAG Motion Axis Gear	Yes
	Change the speed, acceleration, or deceleration of a move or a jog that is in progress.	MCD Motion Change Dynamics	Yes
	Change the command or actual position of an axis.	MRP Motion Redefine Position	Yes
	Calculate a Cam Profile based on an array of cam points.	MCCP Motion Calculate Cam Profile	No
	Start electronic camming between 2 axes.	MAPC Motion Axis Position Cam	No
	Start electronic camming as a function of time.	MATC Motion Axis Time Cam	No
	Calculate the slave value, slope, and derivative of the slope for a cam profile and master value.	MCSV Motion Calculate Slave Values	No

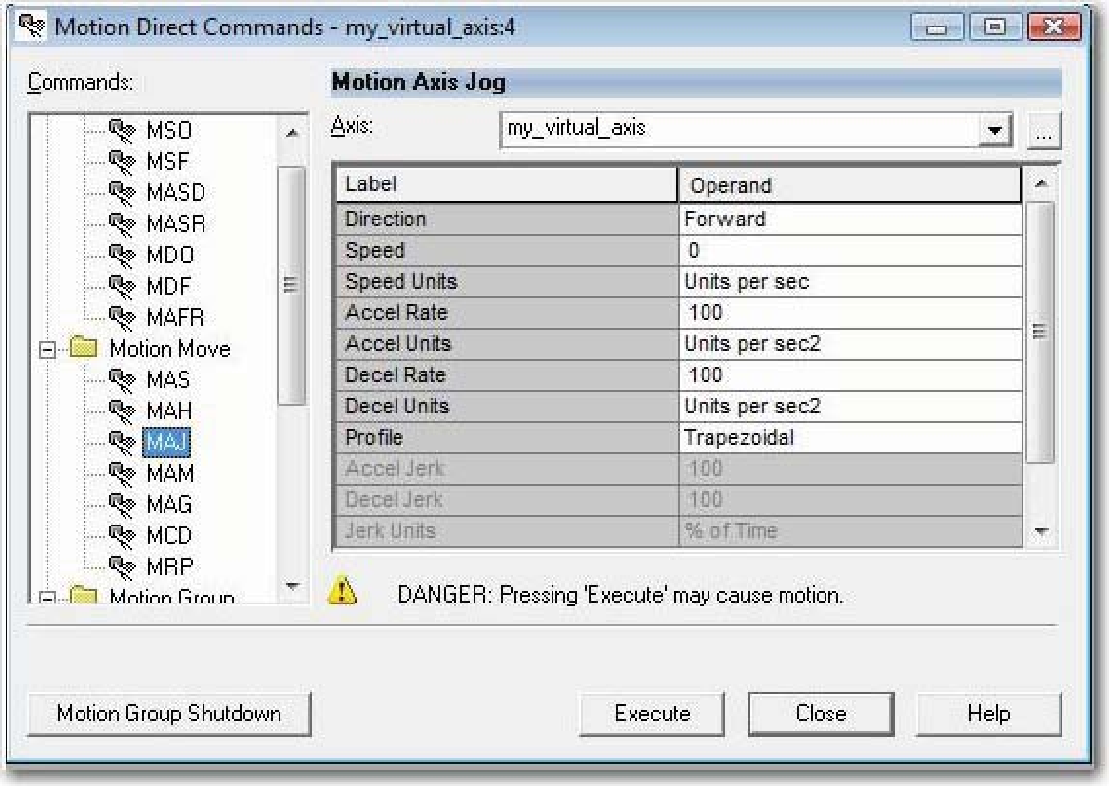
If you want to	And	Use this instruction	Motion Direct Command
Initiate action on all axes	Stop motion of all axes.	MGS Motion Group Stop	Yes
	Force all axes into the shutdown state.	MGSD Motion Group Shutdown	Yes
	Transition all axes to the ready state.	MGSR Motion Group Shutdown Reset	Yes
	Latch the current command and actual position of all axes.	MGSP Motion Group Strobe Position	Yes
Arm and disarm special event checking functions such as registration and watch position	Arm the watch-position event checking for an axis.	MAW Motion Arm Watch Position	Yes
	Disarm the watch-position event checking for an axis.	MDW Motion Disarm Watch Position	Yes
	Arm the servo-module registration-event checking for an axis.	MAR Motion Arm Registration	Yes
	Disarm the servo-module registration-event checking for an axis.	MDR Motion Disarm Registration	Yes
	Arm an output cam for an axis and output.	MAOC Motion Arm Output Cam	No
	Disarm one or all output cams connected to an axis.	MDOC Motion Disarm Output Cam	No
Tune an axis and run diagnostic tests for your control system. These tests include: • Motor/encoder hookup test • Encoder hookup test • Marker test	Use the results of an MAAT instruction to calculate and update the servo gains and dynamic limits of an axis.	MAAT Motion Apply Axis Tuning	No
	Run a tuning motion profile for an axis.	MRAT Motion Run Axis Tuning	No
	Use the results of an MRHD instruction to set encoder and servo polarities.	MAHD Motion Apply Hookup Diagnostic	No
	Run one of the diagnostic tests on an axis.	MRHD Motion Run Hookup Diagnostic	No

If you want to	**And**	**Use this instruction**	**Motion Direct Command**
Control multi-axis coordinated motion	Start a linear coordinated move for the axes of coordinate system.	MCLM Motion Coordinated Linear Move	No
	Start a circular move for the for the axes of coordinate system.	MCCM Motion Coordinated Circular Move	No
	Change in path dynamics for the active motion on a coordinate system.	MCCD Motion Coordinated Change Dynamics	No
	Stop the axes of a coordinate system.	MCS Motion Coordinated Stop	No
	Shutdown the axes of a coordinate system.	MCSD Motion Coordinated Shutdown	No
	Transition the axes of a coordinate system to the ready state and clear the axis faults.	MCSR Motion Coordinated Shutdown Reset	No
	Start a transform that links two coordinate systems together.	MCT() Motion Coordinated Transform	No
	Calculate the position of one coordinate system with respect to another coordinate system.	MCTP() Motion Calculate Transform Position	No

()You can use this instruction only with 1756-L6*x* controllers.

Motion Direct Command Dialog Box

You must be online to execute a Motion Direct Command. The online dialog box has the Motion Group Shutdown and Execute active. If you click either of these, action is taken immediately.



Motion Direct Commands - my_virtual_axis:4

Motion Axis Jog

Axis: my_virtual_axis

Label	Operand
Direction	Forward
Speed	0
Speed Units	Units per sec
Accel Rate	100
Accel Units	Units per sec2
Decel Rate	100
Decel Units	Units per sec2
Profile	Trapezoidal
Accel Jerk	100
Decel Jerk	100
Jerk Units	% of Time

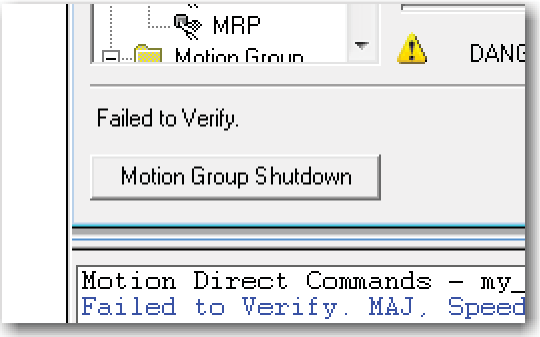
⚠ DANGER: Pressing 'Execute' may cause motion.

In the Command list, you can either type the mnemonic and the list advances to the closest match or you can scroll down the list to select a command. Click the desired command and its dialog box appears.

At the top of the dialog box, in the title bar, there is a number at the end of the axis or group that the command is being applied upon. This is the Instance reference number. This number increases each time you open a new instance of the Motion Direct Commands dialog box. The number is cleared when you exit RSLogix 5000 software.

Motion Group Shutdown

The Motion Group Shutdown is located to the left of the screen to avoid accidental invoking of this command when you really want to execute the command accessed from the Command tree.



Failed to Verify.

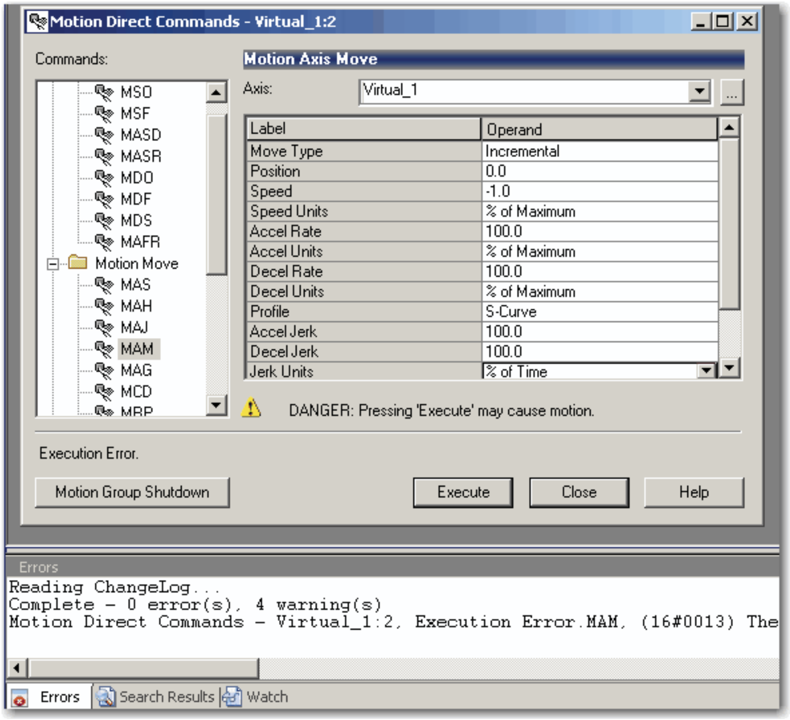
```
Motion Direct Commands - my_
Failed to Verify. MAJ, Speed
```

If you click the Motion Group Shutdown and it is successfully executed, a Result message is displayed in the results window below the dialog box. Since the use of this Motion Group Shutdown is an abrupt means of stopping motion, an additional message is displayed in the error text field. The message ‘MOTION GROUP SHUTDOWN executed!’ is displayed with the intention of giving greater awareness of the execution of this command. If the command fails then an error is indicated as per normal operation.

Clicking Execute verifies the operands and initiates the current Motion Direct Command. There is space above Motion Group Shutdown and below the line where status text is displayed when a command is executed.

Motion Direct Command Error Process

When you execute a Motion Direct Command there are two levels of error detections. The first level is verification of the command's operands. If a verification error is detected, a message 'Failed to Verify' is posted on the dialog box and an appropriate message is posted to the error result window. The second level is the initial motion direct command's error response return code. If an error code is detected, a message 'Execution Error' is posted on the dialog box.



Axis: Virtual_1

Label	Operand
Move Type	Incremental
Position	0.0
Speed	-1.0
Speed Units	% of Maximum
Accel Rate	100.0
Accel Units	% of Maximum
Decel Rate	100.0
Decel Units	% of Maximum
Profile	S-Curve
Accel Jerk	100.0
Decel Jerk	100.0
Jerk Units	% of Time

DANGER: Pressing 'Execute' may cause motion.

Execution Error.

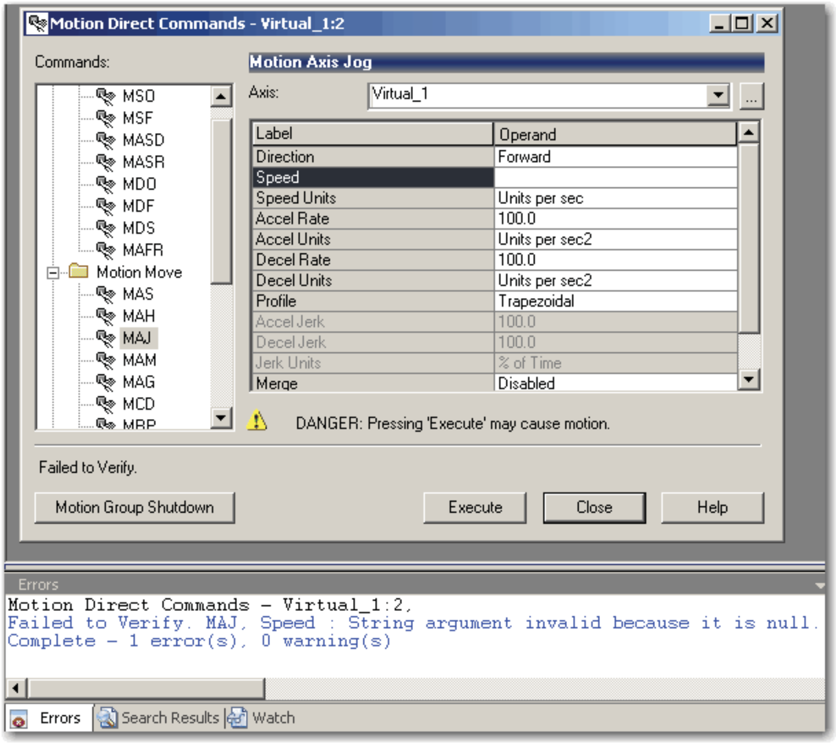
```
Reading ChangeLog...
Complete - 0 error(s), 4 warning(s)
Motion Direct Commands - Virtual_1:2, Execution Error.MAM, (16#0013) The
```

Whether or not an error is detected, a message appears to the Error result window describing the results of the executed command.

Motion Direct Command Verification

When you select Execute from a Motion Direct Command dialog box, the operands are verified. If any operand fails verification, an error message 'Failed to Verify' appears on the dialog box. A detailed error message appears in the error result window describing the fault.

This allows multiple verification errors to display and provides navigation to the error source, that is, double-clicking the error in the results window will navigate to the appropriate Motion Direct Command dialog box.



Axis: Virtual_1

Label	Operand
Direction	Forward
Speed	
Speed Units	Units per sec
Accel Rate	100.0
Accel Units	Units per sec2
Decel Rate	100.0
Decel Units	Units per sec2
Profile	Trapezoidal
Accel Jerk	100.0
Decel Jerk	100.0
Jerk Units	% of Time
Merge	Disabled

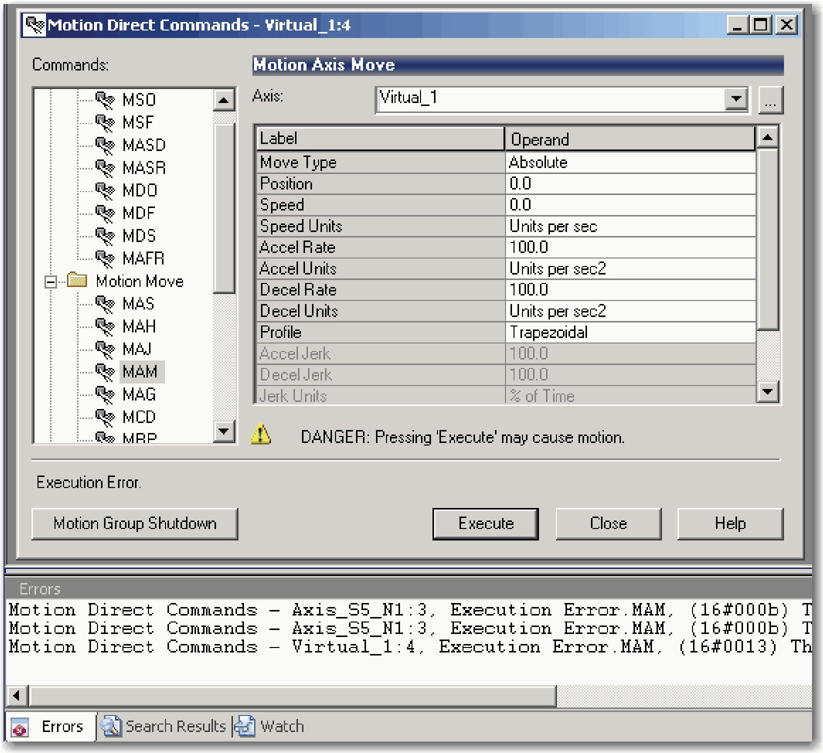
Failed to Verify.

```
Motion Direct Commands - Virtual_1:2,
Failed to Verify. MAJ, Speed : String argument invalid because it is null.
Complete - 1 error(s), 0 warning(s)
```

If no errors are detected there is still status that is displayed indicating which instruction was executed and that it had no errors.

Motion Direct Command Execution Error

When you select Execute from a Motion Direct Command dialog box and the operands are verified as valid, then the command is executed. If the command fails immediately, then an error message 'Execution Error' appears on the dialog box. Whether or not an error is detected, a detailed message appears to the Error result window describing the immediate results of the executed command.



Axis: Virtual_1

Label	Operand
Move Type	Absolute
Position	0.0
Speed	0.0
Speed Units	Units per sec
Accel Rate	100.0
Accel Units	Units per sec2
Decel Rate	100.0
Decel Units	Units per sec2
Profile	Trapezoidal
Accel Jerk	100.0
Decel Jerk	100.0
Jerk Units	% of Time

Execution Error.

```
Motion Direct Commands - Axis_S5_N1:3, Execution Error.MAM, (16#000b) T
Motion Direct Commands - Axis_S5_N1:3, Execution Error.MAM, (16#000b) T
Motion Direct Commands - Virtual_1:4, Execution Error.MAM, (16#0013) Th
```

The message 'Execution Error' clears on subsequent command execution or if you select a new command. The information provided in the error result window after an execution is not cleared provides a history of what has been executed.

What If the Software Goes Offline or the Controller Changes Modes?

When the RSLogix 5000 software transitions to Off-line and Direction Command instructions continue, but if the controller Mode is changed then the configured 'Programed Stop Mode' controls what happens to motion.

Can Two Workstations Give Motion Direct Commands?

When Execute is enabled and commands can be executed from a workstation, the group is locked. This means that another workstation cannot execute commands while this lock is in place. The lock is relinquished when all Motion Direct Command Dialogs for the Motion Group are closed.

Notes:

Program

Program a Velocity Profile and Jerk Rate

This chapter discusses how to program a velocity profile and jerk rate. You can use either of these motion profiles for various instructions:

- **Trapezoidal** profile for linear acceleration and deceleration
- **S-curve** profiles for controlled jerk

Definition of Jerk

Jerk is the rate of change of acceleration or deceleration.

The jerk parameters only apply to S-curve profile moves using these instructions:

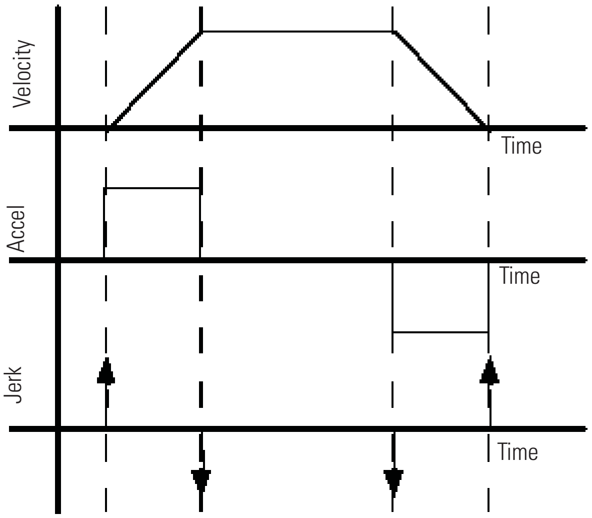
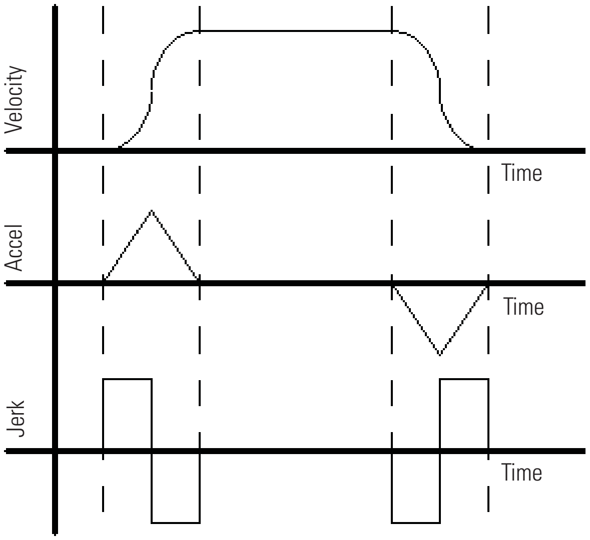
- MAJ
- MAM
- MAS
- MCD
- MCS
- MCCD
- MCCM
- MCLM

Example: If acceleration changes from 0 to 40 mm/s^2 in 0.2 seconds, the jerk is:

$(40\ mm/s^2 - 0\ mm/s^2) / 0.2\ s = 200\ mm/s^3$

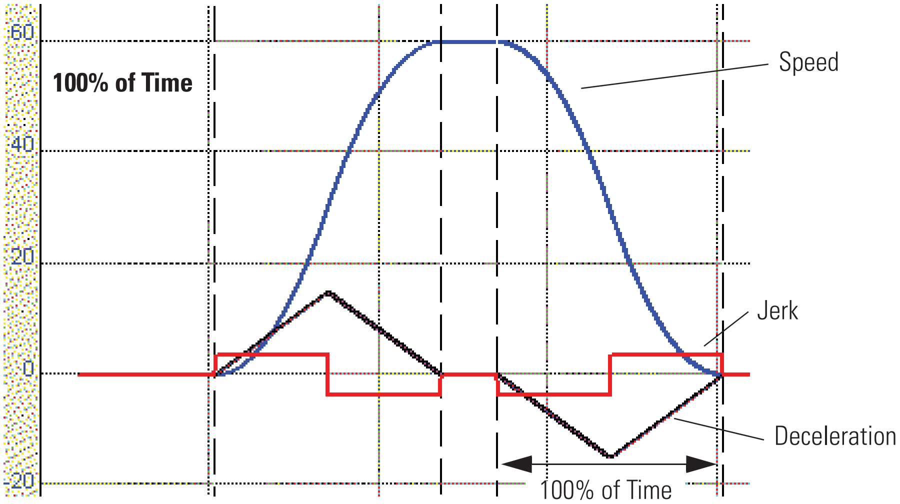
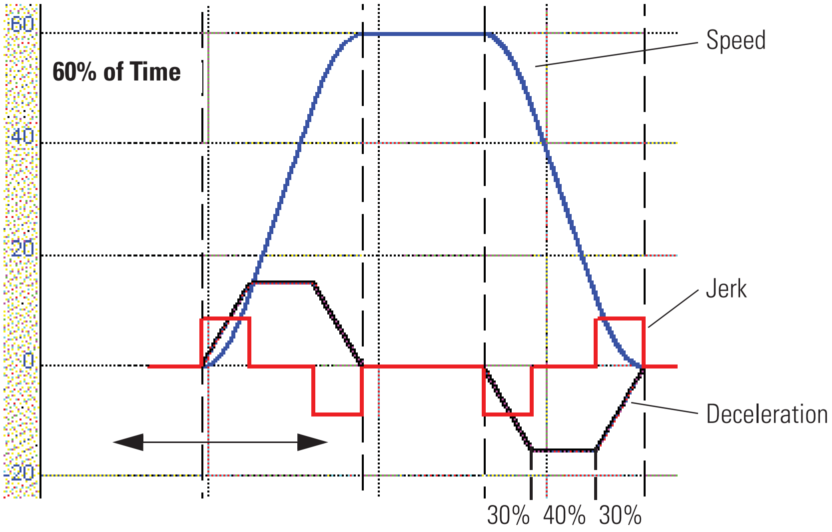
Choose a Profile

Consider cycle time and smoothness when you choose a profile.

If you want	Choose this profile	Consideration
• Fastest acceleration and deceleration times • More flexibility in programming subsequent motion	Trapezoidal 	Jerk **does not** limit the acceleration and deceleration time: • The Acceleration and Deceleration rates control the maximum change in Velocity. • Your equipment and load get more stress than with an S-curve profile. • Jerk is considered infinite and is shown as a vertical line.
Smoother acceleration and deceleration that reduces the stress on the equipment and load	S-curve 	Jerk limits the acceleration and deceleration time: • It takes longer to accelerate and decelerate than a trapezoidal profile. • If the instruction uses an S-curve profile, the controller calculates acceleration, deceleration, and jerk when you start the instruction. • The controller calculates triangular acceleration and deceleration profiles.

Use % of Time for the Easiest Programming of Jerk

Use % of Time to specify how much of the acceleration or deceleration time has jerk. You do not have to calculate actual jerk values.

Example	Profile
100% of Time	At 100% of Time, the acceleration or deceleration changes the entire time that the axis speeds up or slows down. 
60% of Time	At 60% of Time, the acceleration or deceleration changes 60% of the time that the axis speeds up or slows down. The acceleration or deceleration is constant for the other 40%. 

Velocity Profile Effects

This table summarizes the differences between profiles.

Profile Type	**ACC/DEC** Time	**Motor** Stress	**Priority of Control** **Highest to Lowest**			
Trapezoidal	Fastest	Worst	Acc/Dec	Velocity	Position	
S-curve	2X Slower	Best	Jerk	Acc/Dec	Velocity	Position

Jerk Rate Calculation

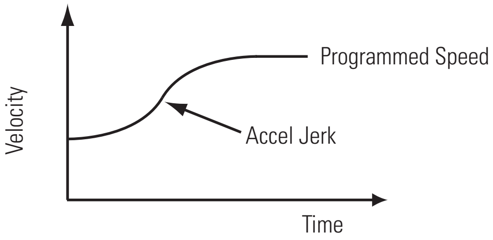
If the instruction uses or changes an S-curve profile, the controller calculates acceleration, deceleration, and jerk when you start the instruction.

The system has a Jerk priority planner. In other words, Jerk always takes priority over acceleration and velocity. Therefore, you always get the programmed Jerk. If a move is velocity-limited, the move does not reach the programmed acceleration and/or velocity.

Jerk Parameters for MAJ programmed in units of % time are converted to engineering units as follows:

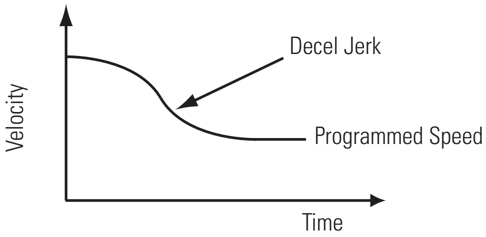
If Start Speed < MAJ Programmed Speed

$$\text{Accel Jerk (Units/Sec}^3\text{)} = \frac{\text{Programmed Accel Rate}^2}{\text{Programmed Speed}} * \left(\frac{200}{\text{\% of Time}} - 1 \right)$$



If Start Velocity > MAJ Programmed Speed

$$\text{Decel Jerk (Units/Sec}^3\text{)} = \frac{\text{Programmed Decel Rate}^2}{\text{Max (Programmed Speed, [Start Speed - Programmed Speed])}} * \left(\frac{200}{\text{\% of Time}} - 1\right)$$

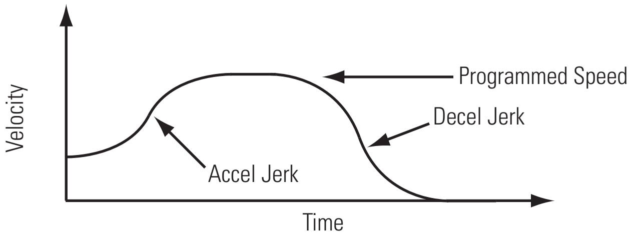


Jerks for programmed moves, such as MAM or MCLM instructions, in units of % time are converted to engineering units as follows:

If Start Speed < Programmed Speed

$$\text{Accel Jerk (Units/Sec}^3\text{)} = \frac{\text{Programmed Accel Rate}^2}{\text{Programmed Speed}} * \left(\frac{200}{\text{\% of Time}} - 1\right)$$

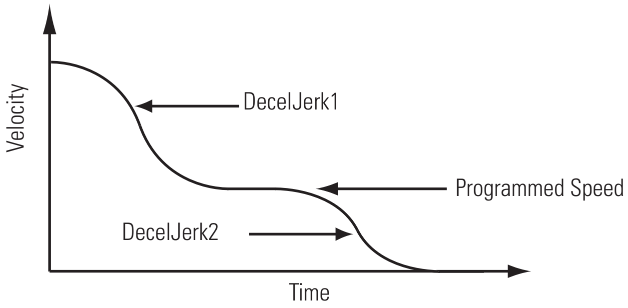
$$\text{Decel Jerk (Units/Sec}^3\text{)} = \frac{\text{Programmed Decel Rate}^2}{\text{Max (Programmed Speed, [Start Speed - Programmed Speed])}} * \left(\frac{200}{\text{\% of Time}} - 1\right)$$



If Start Speed > Programmed Speed

$$\text{DecelJerk1} = \frac{\text{Programmed Decel Rate}^2}{\text{Max (Programmed Speed, [Start Speed - Programmed Speed])}} * \left(\frac{200}{\text{\% of Time}} - 1 \right)$$

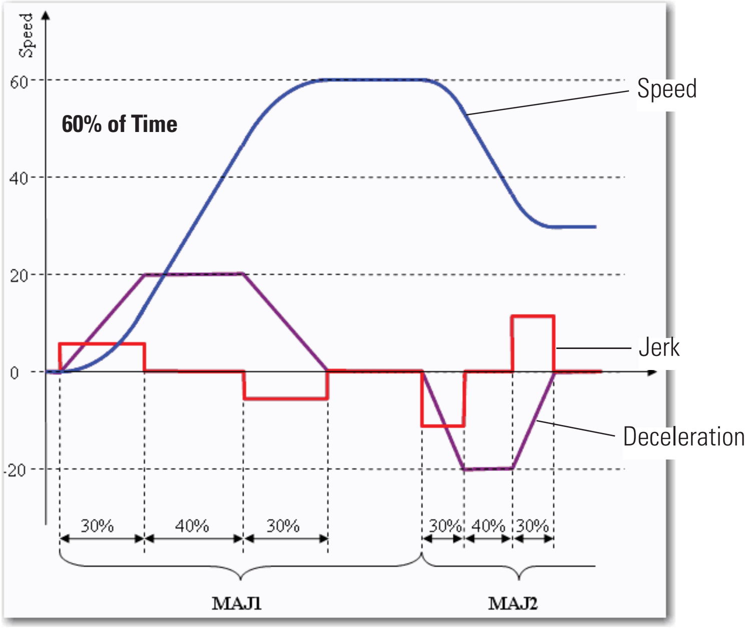
$$\text{DecelJerk2} = \frac{\text{Programmed Decel Rate}^2}{\text{Programmed Speed}} * \left(\frac{200}{\text{\% of Time}} - 1 \right)$$



DecelJerk1 is used while Current Speed > Programmed Speed

DecelJerk2 is used while Current Speed < Programmed Speed

Depending on the instruction’s Speed parameter, the same ‘% of time jerk’ can result in different slopes for the acceleration profile than on the deceleration profile.



The motion planner algorithm adjusts the actual jerk rate so that both the acceleration profile and the deceleration profile contains at least the '% of time' ramp time. If the Start Speed is close to the programmed Speed parameter, the actual percentage of ramp time may be higher than the programmed value.

In most cases the condition is:

if: (start Speed is == 0.0) OR (start Speed is > 2 * max Speed)

then: you get **programmed** percentage of ramp time

else: you get **higher than programmed** percentage of ramp time

Conversion from Engineering Units to % of Time

If you want to convert Engineering Units to % of Time, use these equations.

For Accel Jerk:

$$j_a\,[\%\text{ of Time}] = \frac{2}{1 + \dfrac{j_a\,[EU/s^3]\, v_{max}\,[EU/s]}{a_{max}\,[EU/s^2]}}\,100$$

For Decel Jerk:

$$j_d\,[\%\text{ of Time}] = \frac{2}{1 + \dfrac{j_d\,[EU/s^3]\, v_{max}\,[EU/s]}{d_{max}\,[EU/s^2]}}\,100$$

Jerk Programming in Units/Sec^3

If you want to specify the jerk in 'Units/sec^3' instead of '% of time', adjust your jerk value as follows so that you get the value that you programmed:

$$\text{Temporary Speed} = \frac{\text{Programmed Decel Rate}^2}{\text{Desired Decel Jerk value in Units/Sec}^3}$$

$$k = \frac{\text{Start Speed - Programmed Speed}}{\text{Max (Programmed Speed, Temporary Speed)}}$$

if (k < 1)

- Instruction faceplate Decel jerk in Units/Sec^3 = Desired Decel Jerk in Units/Sec^3

else

- Instruction faceplate Decel jerk in Units/Sec^3 = Desired Decel Jerk in Units/Sec^3 * k

Unique Program Considerations

If you program a move using the '% of Time units', RSLogix 5000 programming software computes an:

> Accel Jerk = a^2/v where a = the programmed Accel Rate and v = programmed Speed.

Therefore, the higher the programmed speed, the lower the computed Jerk. The system has a Jerk priority planner.

In other words, Jerk always takes priority over acceleration and velocity. Therefore, you always get the programmed Jerk. If a move is velocity-limited, the move does not reach the programmed acceleration and/or velocity.

Once you reach the velocity limit for the length of the move, as the velocity is increased, the move takes longer and longer to complete.

'Decel Jerk' is computed similarly to the Accel Jerk described above. The only difference is that instead of a^2/v, Decel Jerk = d^2/v, where d = the programmed Decel Rate.

EXAMPLE

Example #1

Start Speed = 8.0 in/sec
Desired Speed = 5.0 in/sec
Desired Decel Rate = 2.0 in/sec^2
Desired Decel Jerk = 1.0 in/sec^3

Temporary Speed = (Desired Decel Rate)^2 / Desired jerk value in Units/Sec^3 = $2.0^2 / 1.0$ =
= 4.0 in/sec

k = (8.0 - 5.0) / max(5.0, 4.0) = 3.0 / 5.0 =

= 0.6

Because k < 1, we can enter the desired Decel jerk directly in the faceplate

Instruction faceplate Decel jerk in Units/Sec^3 = 1.0 in/sec^3

EXAMPLE

Example #2

Start Speed = 13.0 in/sec
Desired Speed = 5.0 in/sec
Desired Decel Rate = 2.0 in/sec^2
Desired Decel Jerk = 1.0 in/sec^3

Temporary Speed = (Desired Decel Rate)^2 / Desired jerk value in Units/Sec^3 = 2.0^2 / 1.0 =
= 4.0 in/sec

k = (13.0 - 5.0) / max(5.0, 4.0) = 8.0 / 5.0 =

= 1.6

Because k > 1, we have to calculate the Decel jerk to use on the instruction faceplate as:

Instruction faceplate Decel jerk in Units/Sec^3 =

= 1.0 in/sec^3 * 1.6 =

= 1.6 in/sec^3

Which revision do you have?

- 15 or earlier – % of Time is fixed at 100.
- 16 or later – % of Time defaults to 100% of time on projects converted from earlier versions. For new projects, you must enter the Jerk value.

Profile Operand

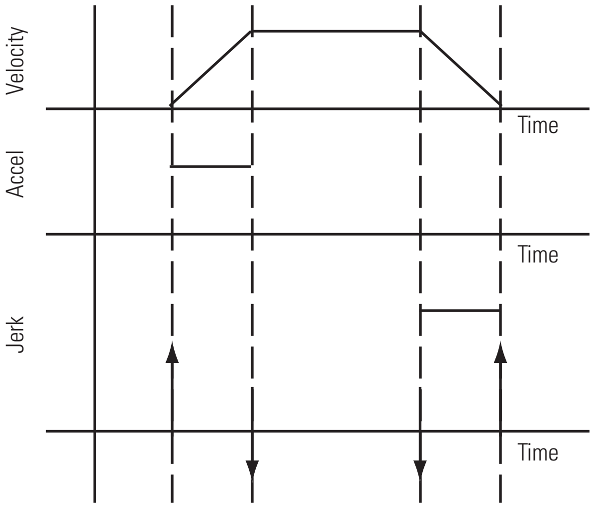
This operand has two profile types:

- Trapezoidal Velocity Profile
- S-Curve Velocity Profile

Trapezoidal Velocity Profile

The trapezoidal velocity profile is the most commonly used profile because it provides the most flexibility in programming subsequent motion and the fastest acceleration and deceleration times. The change in velocity per unit time is specified by acceleration and deceleration. Jerk is not a factor for trapezoidal profiles. Therefore, it is considered infinite and is shown as a vertical line in the following graph.

Trapezoidal Accel/Decel Time



S-Curve Velocity Profile

S-Curve velocity profiles are most often used when the stress on the mechanical system and load needs to be minimized. The acceleration and deceleration time is balanced against the machine stress using two additional parameters, accleration jerk and deceleration jerk.

Depending on the Jerk settings, the acceleration profile can be set to almost pure trapezoidal, see Trapezoidal Accel/Decel Time on page 98 (fastest and highest stress), or to S-Curve, see Programmable S-Curve Accel/Decel Time Acceleration Jerk = 60% of Time on page 101, (slowest, lowest stress).

The typical acceleration profile is a trade-off between stress and speed, as shown in S-Curve Accel/Decel Time, Backward Compatibility Setting: Acceleration Jerk = 100% of Time on page 101.

The Jerk is either specified by the user (either in Units/sec^3 or as a percentage of maximum) or it is calculated from the percentage of time. (Percentage of time is equal to the percentage of ramp time in the acceleration/deceleration profile):

$$j_a\,[\mathrm{EU/s^3}] = \frac{a_{max}{}^2\,[\mathrm{EU/s^2}]}{v_{max}\,[\mathrm{EU/s}]}\left(\frac{200}{j_a\,[\%\text{ of time}]} - 1\right)$$

$$j_a\,[\mathrm{EU/s^3}] = \frac{d_{max}{}^2\,[\mathrm{EU/s^2}]}{v_{max}\,[\mathrm{EU/s}]}\left(\frac{200}{j_a\,[\%\text{ of time}]} - 1\right)$$

Backward Compatibility

The Jerk of 100% of time produces triangular acceleration and deceleration profiles. These profiles are ones that would have been previously produced as shown in S-Curve Accel/Decel Time, Backward Compatibility Setting: Acceleration Jerk = 100% of Time on page 101.

Very small Jerk rates, that is less than 5% of time, produce acceleration and deceleration profiles close to trapezoidal ones, such as the one shown in Trapezoidal Accel/Decel Time on page 98.

IMPORTANT Higher values of the % of Time result in lower values of Jerk Rate Limits and, therefore, slower profiles. See the following table for reference.

	Trapezoidal Velocity Profile(1)	**S-shaped Velocity Profile with 1<=Jerk <100% of Time(2)**	**S-shaped Velocity Profile with Jerk = 100% of Time(3)**
Accel/Decel Jerk in Units/sec^3	∞	$\frac{\text{Max Accel}^2}{\text{Max Velocity}}$ to ∞	$\frac{\text{Max Accel}^2}{\text{Max Velocity}}$
Accel/Decel Jerk in % of Maximum	NA	0 - 100%	NA
Accel/Decel Jerk in % of Time	0%	1 - 100%	100%

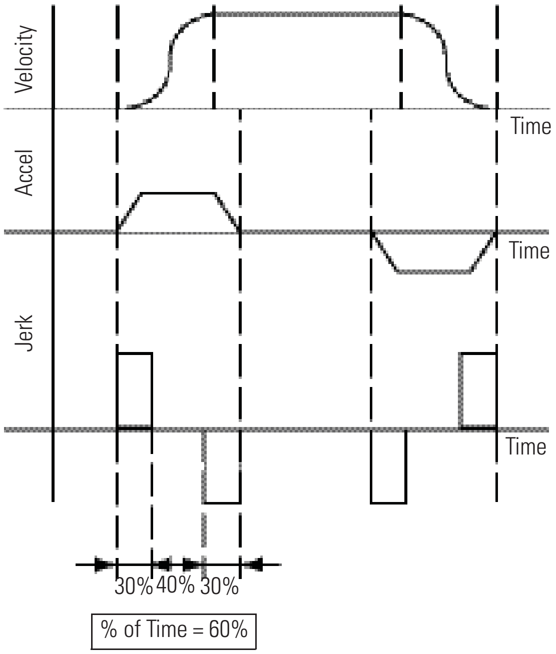
(1) The example on page 98 (labeled Trapezoidal Accel/Decel Time) uses a rectangular acceleration profile.

(2) The example on page 101 (labeled Programmable S-Curve Accel/Decel Time, Acceleration Jerk = 60% of Time) uses a trapezoidal acceleration profile.

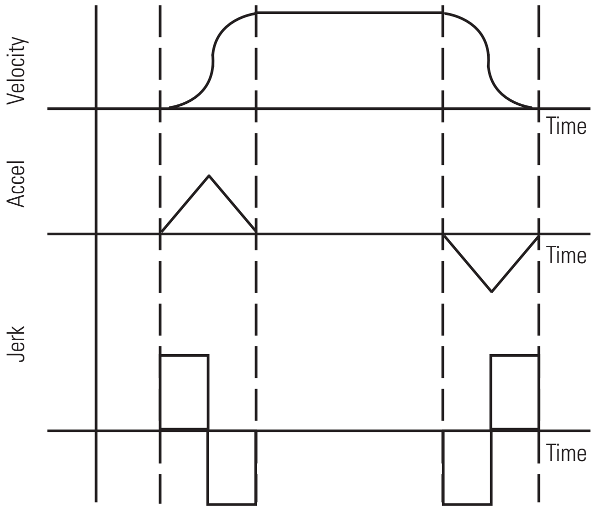
(3) The example on page 101 (labeled S-Curve Accel/Decel Time, Backward Compatibility Setting: Acceleration Jerk = 100% of Time) uses a triangular acceleration profile.

Calculations are performed when an Axis Move, Change Dynamics or an MCS Stop of StopType = Move or Jog is initiated.

Programmable S-Curve Accel/Decel Time
Acceleration Jerk = 60% of Time



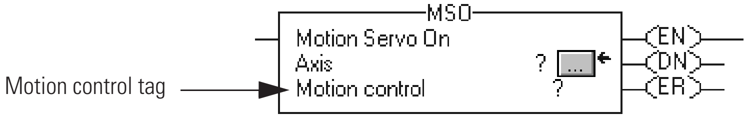
S-Curve Accel/Decel Time, Backward Compatibility Setting:
Acceleration Jerk = 100% of Time



Enter Basic Logic

The controller gives you a set of motion control instructions for your axes.

- Uses these instructions just like the rest of the Logix5000 instructions. You can program motion control in these programming languages:
  - ladder diagram (LD)
  - structured text (ST)
  - sequential function chart (SFC)
- Each motion instruction works on one or more axes.
- Each motion instruction needs a motion control tag. The tag uses a MOTION_INSTRUCTION data type. The tag stores the status information of the instruction.



ATTENTION

Use the tag for the motion control operand of motion instruction only once. Unintended operation of the control variables may happen if you re-use of the same motion control tag in other instructions.

Example Motion Control Program

This is an example of Ladder Logix that homes, jogs, and moves an axis.

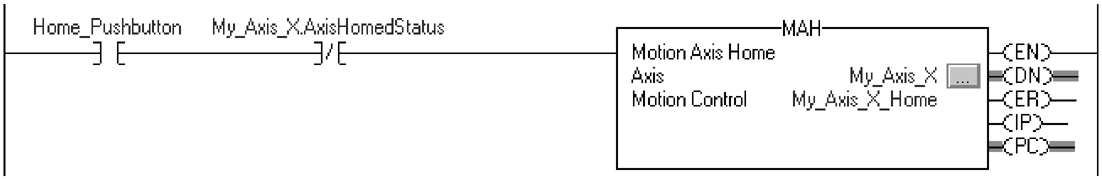
If *Initialize_Pushbutton* = on and the axis = off (*My_Axis_X.ServoActionStatus* = off) then

The MSO instruction turns on the axis.

Motion Servo On
Axis My_Axis_X
Motion Control My_Axis_X_On

If *Home_Pushbutton* = on and the axis hasn't been homed (*My_Axis_X.AxisHomedStatus* = off) then

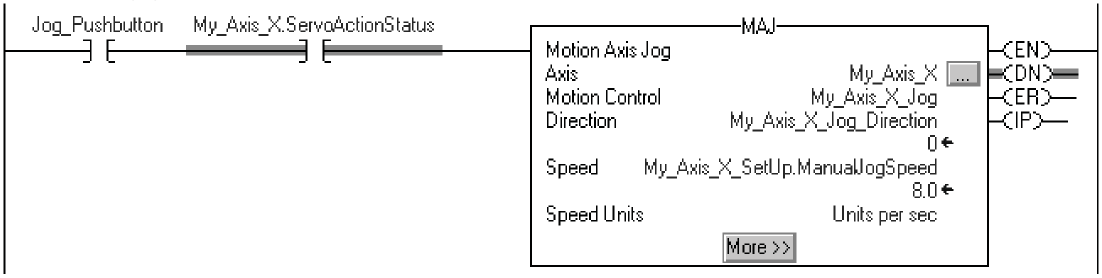
The MAH instruction homes the axis.



Motion Axis Home
Axis My_Axis_X
Motion Control My_Axis_X_Home

If *Jog_Pushbutton* = on and the axis = on (*My_Axis_X.ServoActionStatus* = on) then

The MAJ instruction jogs the axis forward at 8 units/s.

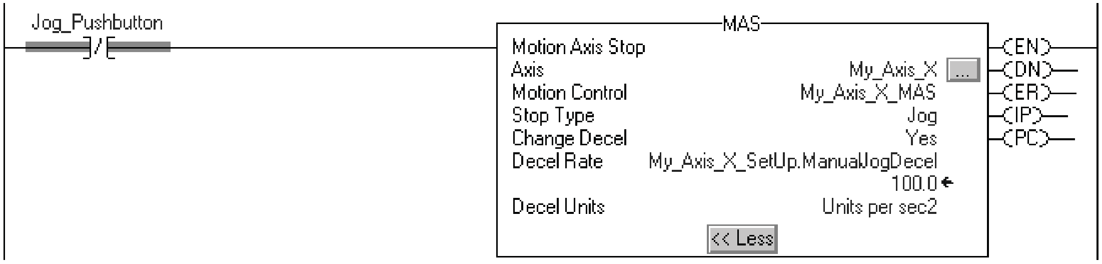


Motion Axis Jog
Axis My_Axis_X
Motion Control My_Axis_X_Jog
Direction My_Axis_X_Jog_Direction
0 ←
Speed My_Axis_X_SetUp.ManualJogSpeed
8.0 ←
Speed Units Units per sec

If *Jog_Pushbutton* = off then

The MAS instruction stops the axis at 100 units/s^2

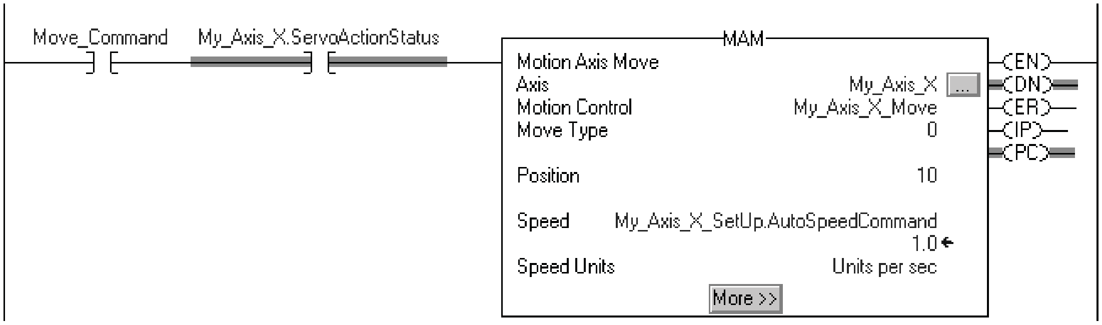
Make sure that *Change Decel* is *Yes*. Otherwise, the axis decelerates at its maximum speed.



Motion Axis Stop
Axis My_Axis_X
Motion Control My_Axis_X_MAS
Stop Type Jog
Change Decel Yes
Decel Rate My_Axis_X_SetUp.ManualJogDecel
100.0 ←
Decel Units Units per sec2

If *Move_Command* = on and the axis = on (*My_Axis_X.ServoActionStatus* = on) then

The MAM instruction moves the axis. The axis moves to the position of 10 units at 1 unit/s.



Motion Axis Move
Axis My_Axis_X
Motion Control My_Axis_X_Move
Move Type 0
Position 10
Speed My_Axis_X_SetUp.AutoSpeedCommand
1.0 ←
Speed Units Units per sec

Download a Project and Run the Logic

Follow these steps to download your program to a controller.

1. With the keyswitch, place the controller in Program or Remote Program mode.
1. From the Communications menu, choose Download.
2. Confirm that you wish to complete the download procedure.
3. Click Download.
4. Once the download is complete, place the controller in Run/Test mode.

   After the project file is downloaded, status and compiler messages appear in the status bar.

Choose a Motion Instruction

Use this table to choose an instruction and see if it is available as a Motion Direct Command.

If You Want To	And	Use This Instruction	Motion Direct Command
Change the state of an axis	Enable the drive and activate the axis servo loop.	MSO Motion Servo On	Yes
	Disable the drive and deactivate the axis servo loop.	MSF Motion Servo Off	Yes
	Force an axis into the shutdown state and block any instructions that initiate axis motion.	MASD Motion Axis Shutdown	Yes
	Reset the axis from the shutdown state.	MASR Motion Axis Shutdown Reset	Yes
	Enable the drive and set the servo output voltage of an axis.	MDO Motion Direct Drive On	Yes
	Disable the drive and set the servo output voltage to the output offset voltage.	MDF Motion Direct Drive Off	Yes
	Clear all motion faults for an axis.	MAFR Motion Axis Fault Reset	Yes
Control axis position	Stop any motion process on an axis.	MAS Motion Axis Stop	Yes
	Home an axis.	MAH Motion Axis Home	Yes
	Jog an axis.	MAJ Motion Axis Jog	Yes
	Move an axis to a specific position.	MAM Motion Axis Move	Yes
	Start electronic gearing between 2 axes.	MAG Motion Axis Gear	Yes
	Change the speed, acceleration, or deceleration of a move or a jog that is in progress.	MCD Motion Change Dynamics	Yes
	Change the command or actual position of an axis.	MRP Motion Redefine Position	Yes
	Calculate a Cam Profile based on an array of cam points.	MCCP Motion Calculate Cam Profile	No
	Start electronic camming between 2 axes.	MAPC Motion Axis Position Cam	No
	Start electronic camming as a function of time.	MATC Motion Axis Time Cam	No
	Calculate the slave value, slope, and derivative of the slope for a cam profile and master value.	MCSV Motion Calculate Slave Values	No

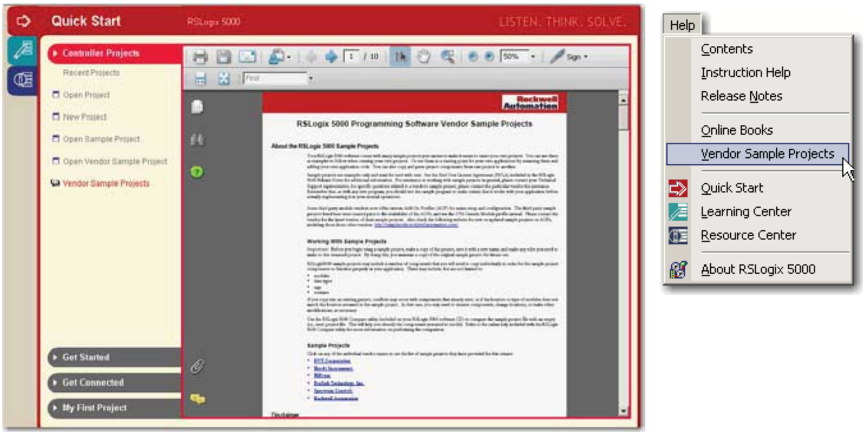
If You Want To	And	Use This Instruction	Motion Direct Command
Initiate action on all axes	Stop motion of all axes.	MGS Motion Group Stop	Yes
	Force all axes into the shutdown state.	MGSD Motion Group Shutdown	Yes
	Transition all axes to the ready state.	MGSR Motion Group Shutdown Reset	Yes
	Latch the current command and actual position of all axes.	MGSP Motion Group Strobe Position	Yes
Arm and disarm special event checking functions such as registration and watch position	Arm the watch-position event checking for an axis.	MAW Motion Arm Watch Position	Yes
	Disarm the watch-position event checking for an axis.	MDW Motion Disarm Watch Position	Yes
	Arm the servo-module registration-event checking for an axis.	MAR Motion Arm Registration	Yes
	Disarm the servo-module registration-event checking for an axis.	MDR Motion Disarm Registration	Yes
	Arm an output cam for an axis and output.	MAOC Motion Arm Output Cam	No
	Disarm one or all output cams connected to an axis.	MDOC Motion Disarm Output Cam	No
Tune an axis and run diagnostic tests for your control system. These tests include: • Motor/encoder hookup test • Encoder hookup test • Marker test	Uses the results of an MRAT instruction to calculate and update the servo gains and dynamic limits of an axis.	MAAT Motion Apply Axis Tuning	No
	Run a tuning motion profile for an axis.	MRAT Motion Run Axis Tuning	No
	Use the results of an MRHD instruction to set encoder and servo polarities.	MAHD Motion Apply Hookup Diagnostic	No
	Run one of the diagnostic tests on an axis.	MRHD Motion Run Hookup Diagnostic	No

If You Want To	And	Use This Instruction	Motion Direct Command
Control multi-axis coordinated motion	Start a linear coordinated move for the axes of coordinate system.	MCLM Motion Coordinated Linear Move	No
	Start a circular move for the for the axes of coordinate system.	MCCM Motion Coordinated Circular Move	No
	Change in path dynamics for the active motion on a coordinate system.	MCCD Motion Coordinated Change Dynamics	No
	Stop the axes of a coordinate system or cancel a transform.	MCS Motion Coordinated Stop	No
	Shutdown the axes of a coordinate system.	MCSD Motion Coordinated Shutdown	No
	Start a transform that links two coordinate systems together. This is like bi-directional gearing.	MCT Motion Coordinated Transform (1)	No
	Calculate the position of one coordinate system with respect to another coordinate system.	MCTP Motion Calculate Transform Position(1)	No
	Transition the axes of a coordinate system to the ready state and clear the axis faults.	MCSR Motion Coordinated Shutdown Reset	No

(1) You can only use this instruction with 1756-L6*x* controllers.

Sample Projects

Use the RSLogix 5000 Start Page (ALT+F9) to find the sample projects.



The Rockwell Automation sample project's default location is:

C:\RSLogix 5000\Projects\Samples\ENU\v17\Rockwell Automation

There is a PDF file named Vendor Sample Projects on the Start Page that explains how to work with the sample projects. Free sample code is available at: http://samplecode.rockwellautomation.com/.

Troubleshoot Axis Motion

This section helps you troubleshoot some situations that could happen while you are running an axis.

Situation	Page
Why does my axis accelerate when I stop it?	109
Why does my axis overshoot its target speed?	111
Why is there a delay when I stop and then restart a jog?	114
Why does my axis reverse direction when I stop and start it?	116

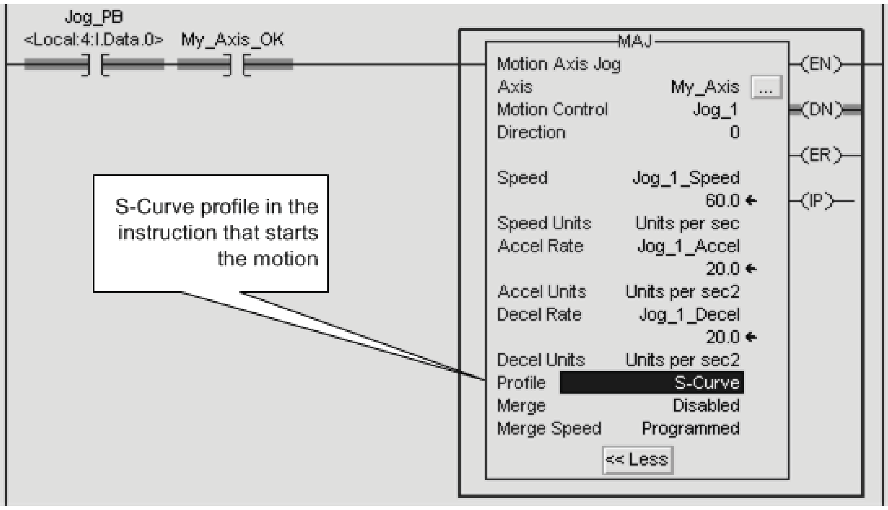
Why does my axis accelerate when I stop it?

While an axis is accelerating, you try to stop it. The axis keeps accelerating for a short time before it starts to decelerate.

Example

You start a Motion Axis Jog (MAJ) instruction. Before the axis gets to its target speed, you start a Motion Axis Stop (MAS) instruction. The axis continues to speed up and then eventually slows to a stop.

Look For



Axis	My_Axis
Motion Control	Jog_1
Direction	0
Speed	Jog_1_Speed 60.0
Speed Units	Units per sec
Accel Rate	Jog_1_Accel 20.0
Accel Units	Units per sec2
Decel Rate	Jog_1_Decel 20.0
Decel Units	Units per sec2
Profile	S-Curve
Merge	Disabled
Merge Speed	Programmed

S-Curve profile in the instruction that starts the motion

Cause

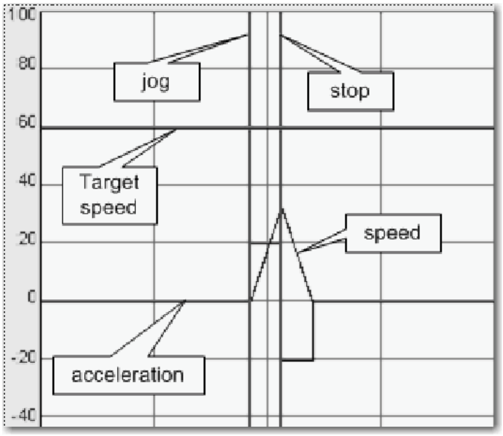
When you use an S-curve profile, jerk determines the acceleration and deceleration time of the axis.

- An S-curve profile has to get acceleration to 0 before the axis can slow down.
- The time it takes depends on the acceleration and speed.
- In the meantime, the axis continues to speed up.

The following trends show how the axis stops with a trapezoidal profile and an S-curve profile.

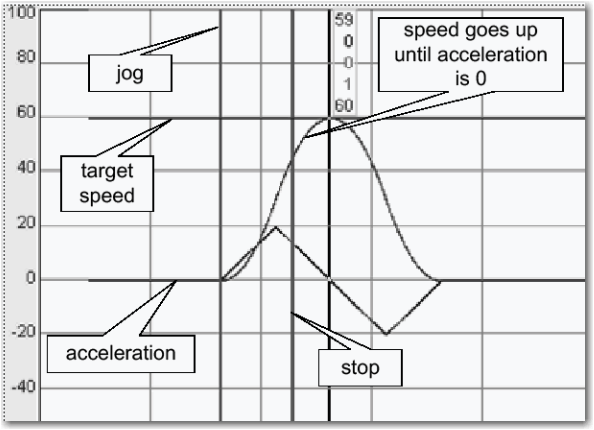
Stop while accelerating

Trapezoidal



The axis slows down as soon as you start the stopping instruction.

S-curve



The axis continues to speed up until the S-curve profile brings the acceleration rate to 0.

Corrective Action

If you want the axis to slow down right away, use a trapezoidal profile.

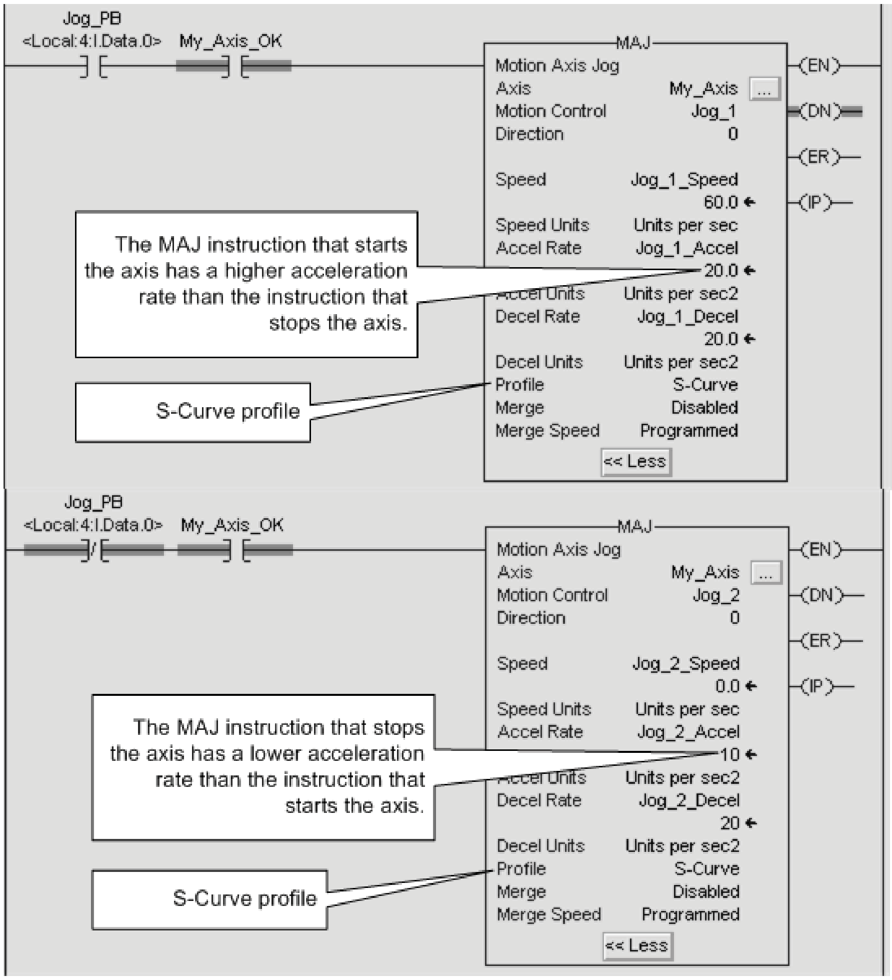
Why does my axis overshoot its target speed?

While an axis is accelerating, you try to stop the axis or change its speed. The axis keeps accelerating and goes past its initial target speed. Eventually it starts to decelerate.

Example

You start a Motion Axis Jog (MAJ) instruction. Before the axis gets to its target speed, you try to stop it with another MAJ instruction. The speed of the second instruction is set to 0. The axis continues to speed up and overshoots its initial target speed. Eventually it slows to a stop.

Look For



Motion Axis Jog
Axis My_Axis
Motion Control Jog_1
Direction 0
Speed Jog_1_Speed 60.0
Speed Units Units per sec
Accel Rate Jog_1_Accel 20.0
Accel Units Units per sec2
Decel Rate Jog_1_Decel 20.0
Decel Units Units per sec2
Profile S-Curve
Merge Disabled
Merge Speed Programmed

The MAJ instruction that starts the axis has a higher acceleration rate than the instruction that stops the axis.

S-Curve profile

Motion Axis Jog
Axis My_Axis
Motion Control Jog_2
Direction 0
Speed Jog_2_Speed 0.0
Speed Units Units per sec
Accel Rate Jog_2_Accel 10
Accel Units Units per sec2
Decel Rate Jog_2_Decel 20
Decel Units Units per sec2
Profile S-Curve
Merge Disabled
Merge Speed Programmed

The MAJ instruction that stops the axis has a lower acceleration rate than the instruction that starts the axis.

S-Curve profile

Cause

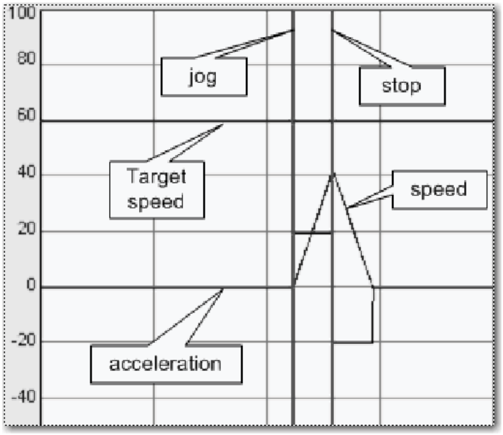
When you use an S-curve profile, jerk determines the acceleration and deceleration time of the axis.

- An S-curve profile has to get acceleration to 0 before the axis can slow down.
- If you reduce the acceleration, it takes longer to get acceleration to 0.
- In the meantime, the axis continues past its initial target speed.

The following trends show how the axis stops with a trapezoidal profile and an S-curve profile.

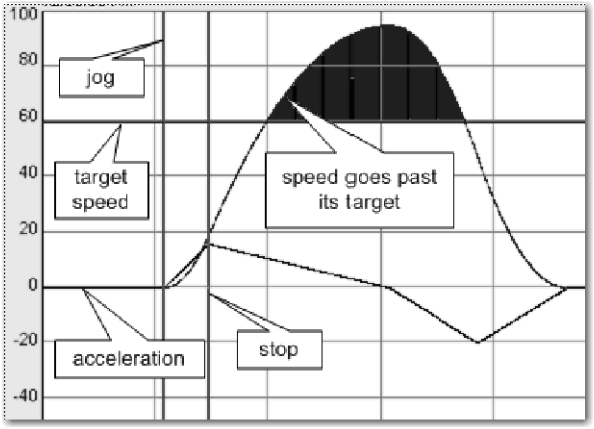
Stop while accelerating and reduce the acceleration rate

Trapezoidal



The axis slows down as soon as you start the stopping instruction. The lower acceleration does not change the response of the axis.

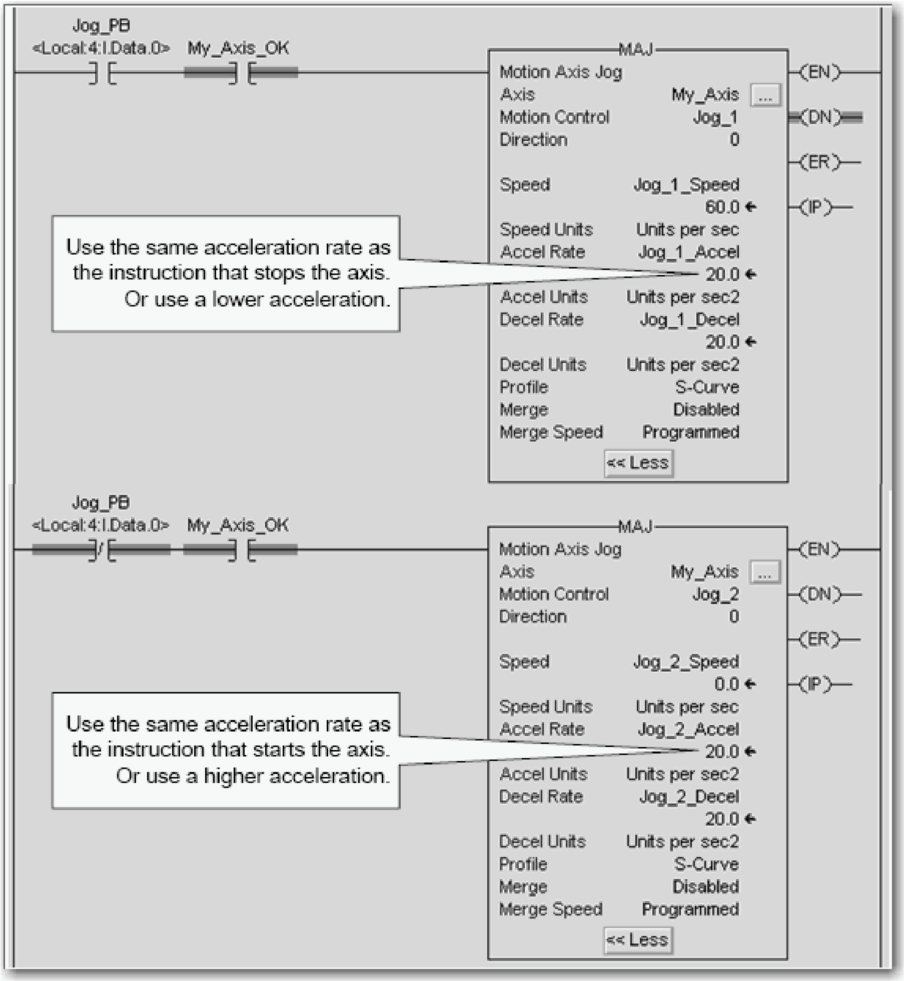
S-curve



The stopping instruction reduces the acceleration of the axis. It now takes longer to bring the acceleration rate to 0. The axis continues past its target speed until acceleration equals 0.

Corrective Action

Use a Motion Axis Stop (MAS) instruction to stop the axis or set up your instructions like this.



Motion Axis Jog
Axis My_Axis
Motion Control Jog_1
Direction 0
Speed Jog_1_Speed 60.0
Speed Units Units per sec
Accel Rate Jog_1_Accel 20.0
Accel Units Units per sec2
Decel Rate Jog_1_Decel 20.0
Decel Units Units per sec2
Profile S-Curve
Merge Disabled
Merge Speed Programmed

Use the same acceleration rate as the instruction that stops the axis. Or use a lower acceleration.

Motion Axis Jog
Axis My_Axis
Motion Control Jog_2
Direction 0
Speed Jog_2_Speed 0.0
Speed Units Units per sec
Accel Rate Jog_2_Accel 20.0
Accel Units Units per sec2
Decel Rate Jog_2_Decel 20.0
Decel Units Units per sec2
Profile S-Curve
Merge Disabled
Merge Speed Programmed

Use the same acceleration rate as the instruction that starts the axis. Or use a higher acceleration.

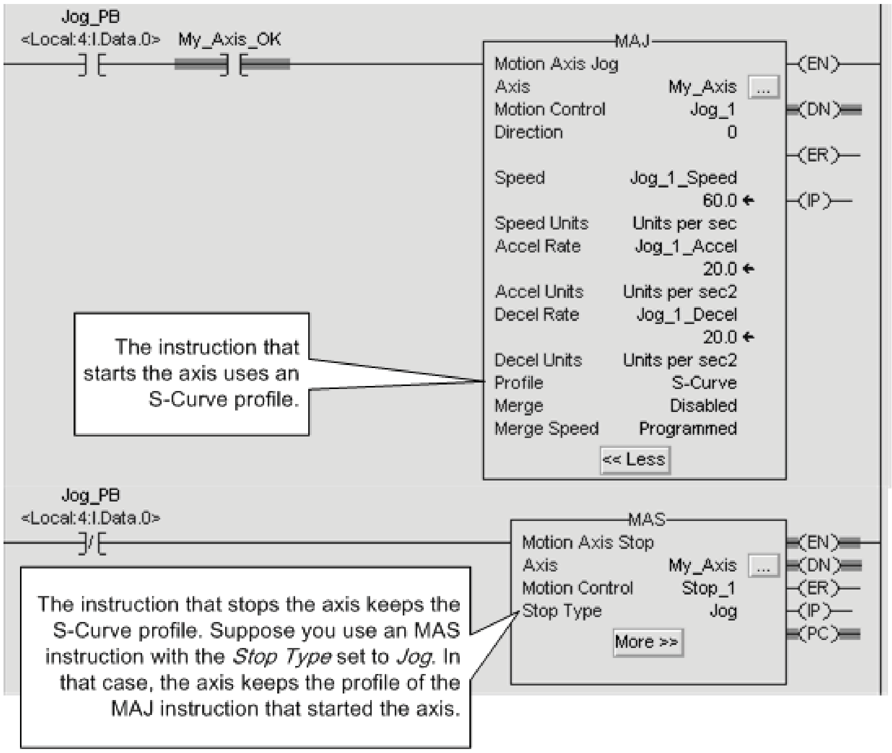
Why is there a delay when I stop and then restart a jog?

While an axis is jogging at its target speed, you stop the axis. Before the axis stops completely, you restart the jog. The axis continues to slow down before it speeds up.

Example

You use a Motion Axis Stop (MAS) instruction to stop a jog. While the axis is slowing down, you use a Motion Axis Jog (MAJ) instruction to start the axis again. The axis does not respond right away. It continues to slow down. Eventually it speeds back up to the target speed.

Look For



Motion Axis Jog
Axis My_Axis
Motion Control Jog_1
Direction 0
Speed Jog_1_Speed 60.0
Speed Units Units per sec
Accel Rate Jog_1_Accel 20.0
Accel Units Units per sec2
Decel Rate Jog_1_Decel 20.0
Decel Units Units per sec2
Profile S-Curve
Merge Disabled
Merge Speed Programmed

The instruction that starts the axis uses an S-Curve profile.

Motion Axis Stop
Axis My_Axis
Motion Control Stop_1
Stop Type Jog

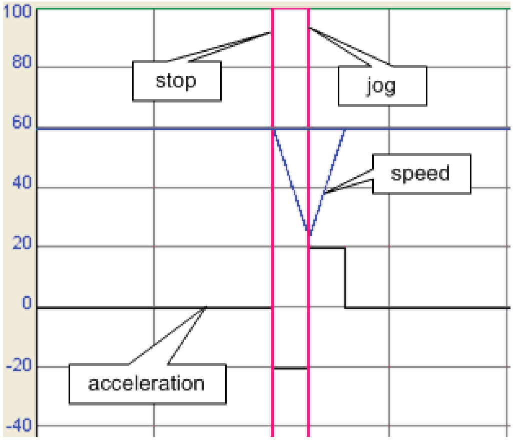
The instruction that stops the axis keeps the S-Curve profile. Suppose you use an MAS instruction with the *Stop Type* set to *Jog*. In that case, the axis keeps the profile of the MAJ instruction that started the axis.

Cause

When you use an S-curve profile, jerk determines the acceleration and deceleration time of the axis. An S-curve profile has to get acceleration to 0 before the axis can speed up again. The following trends show how the axis stops and starts with a trapezoidal profile and an S-curve profile.

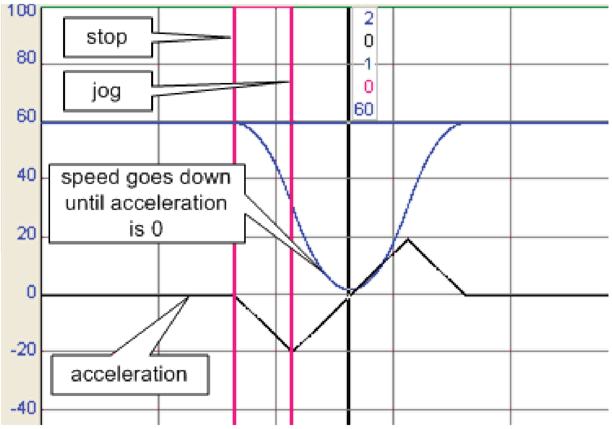
Start while decelerating

Trapezoidal



The axis speeds back up as soon as you start the jog again.

S-curve



The axis continues to slow down until the S-curve profile brings the acceleration rate to 0.

Corrective Action

If you want the axis to accelerate right away, use a trapezoidal profile.

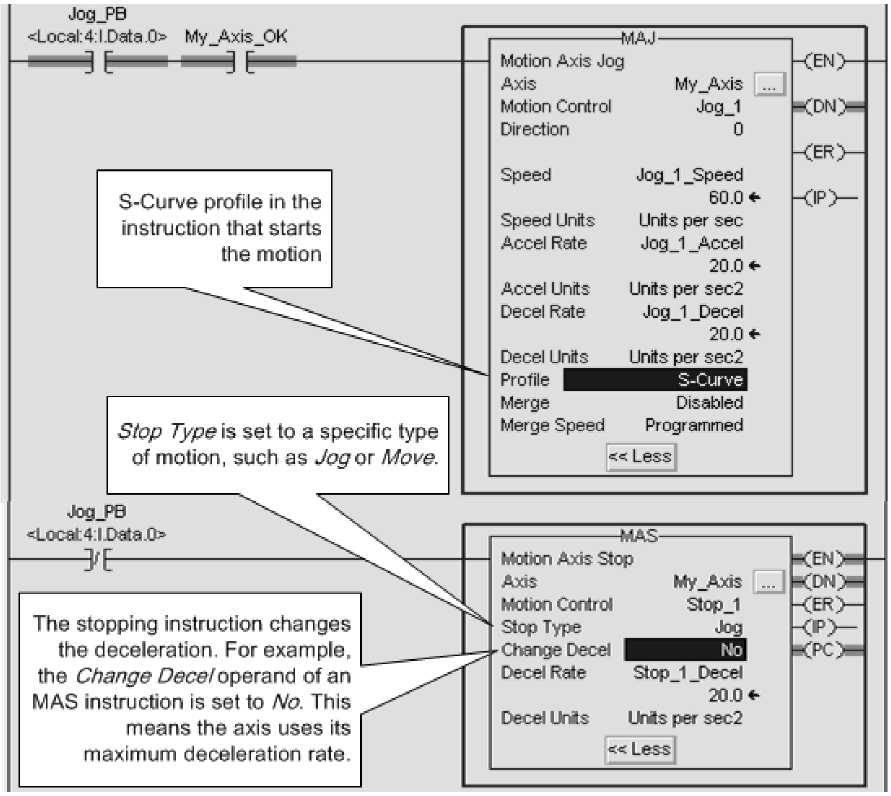
Why does my axis reverse direction when I stop and start it?

While an axis is jogging at its target speed, you stop the axis. Before the axis stops completely, you restart the jog. The axis continues to slow down and then reverse direction. Eventually the axis changes direction again and moves in the programmed direction.

Example

You use a Motion Axis Stop (MAS) instruction to stop a jog. While the axis is slowing down, you use a Motion Axis Jog (MAJ) instruction to start the axis again. The axis continues to slow down and then moves in the opposite direction. Eventually goes back to its programmed direction.

Look For



Motion Axis Jog
Axis: My_Axis
Motion Control: Jog_1
Direction: 0
Speed: Jog_1_Speed 60.0
Speed Units: Units per sec
Accel Rate: Jog_1_Accel 20.0
Accel Units: Units per sec2
Decel Rate: Jog_1_Decel 20.0
Decel Units: Units per sec2
Profile: S-Curve
Merge: Disabled
Merge Speed: Programmed

Motion Axis Stop
Axis: My_Axis
Motion Control: Stop_1
Stop Type: Jog
Change Decel: No
Decel Rate: Stop_1_Decel 20.0
Decel Units: Units per sec2

S-Curve profile in the instruction that starts the motion

Stop Type is set to a specific type of motion, such as *Jog* or *Move*.

The stopping instruction changes the deceleration. For example, the *Change Decel* operand of an MAS instruction is set to *No*. This means the axis uses its maximum deceleration rate.

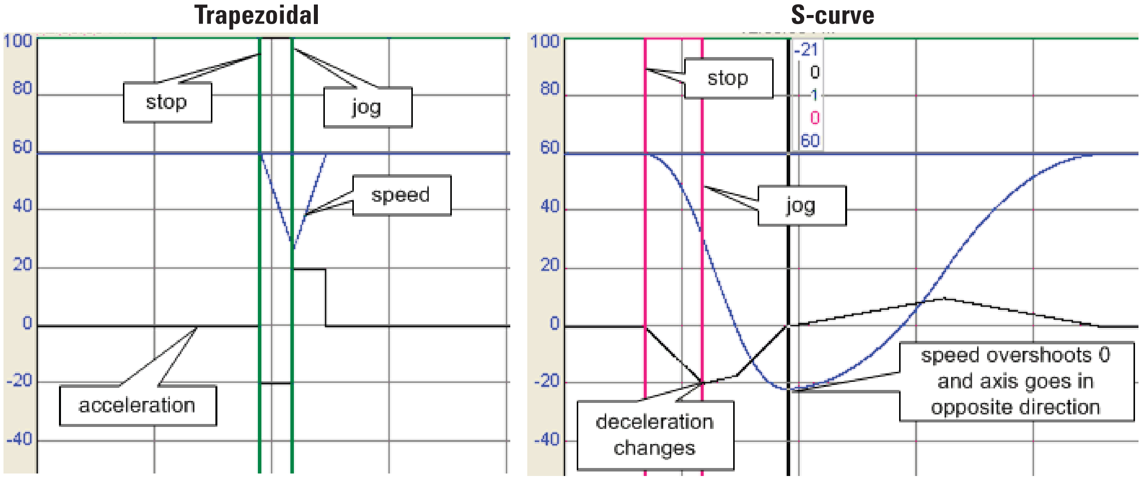
Cause

When you use an S-curve profile, jerk determines the acceleration and deceleration time of the axis.

- An S-curve profile has to get acceleration to 0 before the axis can speed up again.
- If you reduce the acceleration, it takes longer to get acceleration to 0.
- In the meantime, the axis continues past 0 speed and moves in the opposite direction.

The following trends show how the axis stops and starts with a trapezoidal profile and an S-curve profile.

Start while decelerating and reduce the deceleration rate

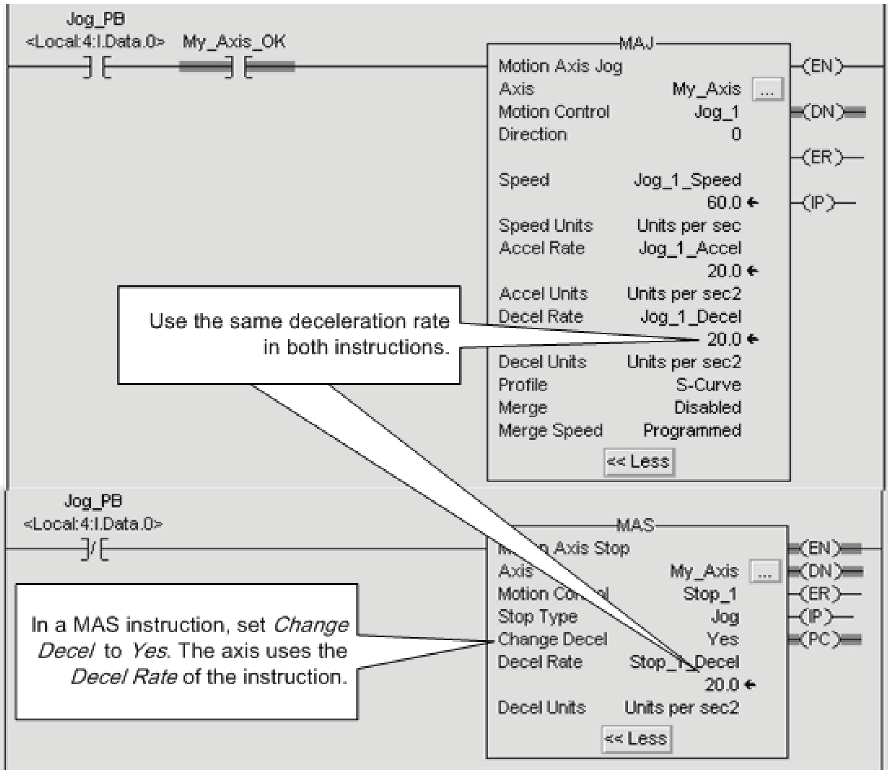


The axis speeds back up as soon as you start the jog again. The lower deceleration does not change the response of the axis.

The jog instruction reduces the deceleration of the axis. It now takes longer to bring the acceleration rate to 0. The speed overshoots 0 and the axis moves in the opposite direction.

Corrective Action

Use the same deceleration rate in the instruction that starts the axis and the instruction that stops the axis.



Motion Axis Jog
Axis My_Axis
Motion Control Jog_1
Direction 0
Speed Jog_1_Speed 60.0
Speed Units Units per sec
Accel Rate Jog_1_Accel 20.0
Accel Units Units per sec2
Decel Rate Jog_1_Decel 20.0
Decel Units Units per sec2
Profile S-Curve
Merge Disabled
Merge Speed Programmed

Motion Axis Stop
Axis My_Axis
Motion Control Stop_1
Stop Type Jog
Change Decel Yes
Decel Rate Stop_1_Decel 20.0
Decel Units Units per sec2

Use the same deceleration rate in both instructions.

In a MAS instruction, set *Change Decel* to *Yes*. The axis uses the *Decel Rate* of the instruction.

Home an Axis

Introduction

Homing puts your equipment at a specific starting point for operation. This starting point is called the home position. Typically, you home your equipment when you reset it for operation. For more information about Homing, see Motion Axis Attributes on page 223.

Guidelines for Homing

This table provides descriptions of homing guidelines.

Guidelines for Homing

Guideline	Description
To move an axis to the home position, use Active homing.	Active homing turns on the servo loop and moves the axis to the home position. Active homing also: • errors if there is any other motion on the axis. It will not stop other motion. • uses a trapezoidal profile.
For a Feedback-only device, use Passive homing.	Passive homing does not move the axis. • Use passive homing to calibrate a Feedback-only axis to its marker. • If you use passive homing on a servo axis, turn on the servo loop and use a move instruction to move the axis.
If you have an absolute feedback device, consider Absolute homing.	If the motion axis hardware supports an absolute feedback device, Absolute Homing Mode may be used. The only valid Home Sequence for Absolute Homing Mode is Immediate. In this case, the absolute homing process establishes the true absolute position of the axis by applying the configured Home Position to the reported position of the absolute feedback device. Prior to execution of the absolute homing process via the MAH instruction, the axis must be in the Axis Ready state with the servo loop disabled.
For single-turn equipment, consider homing to a marker.	The marker homing sequence is useful for single-turn rotary and linear encoder applications because these applications have only one encoder marker for full axis travel.

Guidelines for Homing

Guideline	Description
For multi-turn equipment, home to a switch or switch and marker.	These homing sequences use a home limit switch to define the home position. • You need a home limit switch if the axis moves more than one revolution when it runs. Otherwise the controller cannot tell which marker pulse to use. • For the most precise homing, use both the switch and marker.

If your equipment cannot back up, use unidirectional homing.	With unidirectional homing, the axis does not reverse direction to move to the Home Position. To help insure that the Home operation is complete, consider using an offset.
	If these are not done, the axis position is still correct and accurate.
	• Use a Home Offset that is in the same direction as the Home Direction.
	• Use a Home Offset that is greater than the deceleration distance.
	• If the Home Offset is less than the deceleration distance:
	  • The axis simply slows to a stop. The axis does not reverse direction to move to the Home Position. In this case, the MAH instruction does not set the PC bit.
	  • On a rotary axis, the controller adds 1 or more revolutions to the move distance. This makes sure that the move to the Home Position is unidirectional.

Choose a starting direction for the homing sequence.	Which direction do you want to start the homing sequence in? • Positive direction — choose a Forward direction. • Negative direction — choose a Negative direction.

Active Homing

When the axis Homing Mode is configured as Active, the physical axis is first activated for servo operation. The Home operation will not cancel other motion. It will error, Err=22.

You can home an axis using the configured Home Sequence which may be Immediate, Switch, Marker, Switch-Marker, or Torque-Level homing. The Home Sequences result in the axis being jogged in the configured Home Direction. Using bidirectional homing, after the position is re-defined based on detection of the home event, the axis is automatically moved to the configured Home Position.

IMPORTANT When unidirectional active homing is performed on a rotary axis and the Home Offset value is less than the deceleration distance when the home event is detected, it adds one or more revolutions to the move distance.

Passive Homing

When the axis Homing Mode is configured as Passive, the MAH instruction re-defines the actual position of a physical axis on the next occurrence of the encoder marker, providing that Seq = Marker. Immediate, Switch, Switch-Marker and Torque Level homing is also allowed.

Passive homing is most commonly used to calibrate Feedback Only axes to their markers, but can also be used on Servo axes. Passive homing is identical to active homing to an encoder marker except that the Home command does not command any axis motion.

After initiating passive homing, the axis must be moved past the encoder marker for the homing sequence to complete properly. For closed-loop Servo axes, this may be accomplished with a MAM or MAJ instruction. For physical Feedback Only axes, motion cannot be commanded directly by the motion controller, and must be accomplished via other means.

Absolute Homing

If the motion axis hardware supports an absolute feedback device, Absolute Homing Mode may be used. The only valid Home Sequence for an absolute Homing Mode is 'immediate'. In this case, the absolute homing process establishes the true absolute position of the axis by applying the configured Home Position, to the reported position of the absolute feedback device. Prior to execution of the absolute homing process via the MAH instruction, the axis must be in the Axis Ready state with the servo loop disabled.

To successfully execute a MAH instruction on an axis configured for Active homing mode, the targeted axis must be configured as a Servo Axis Type. To successfully execute an MAH instruction, the targeted axis must be configured as either a Servo or Feedback Only axis. If any of these conditions are not met the instruction errs.

IMPORTANT

When the MAH instruction is initially executed, the In process .IP bit is set and the Process Complete (.PC) bit is cleared.

The MAH instruction execution may take multiple scans to execute because it requires transmission of multiple messages to the motion module. Thus, the Done (.DN) bit, is not set until after these messages have been successfully transmitted.

The In process .IP bit is cleared and the Process Complete (.PC) bit is only set if the final axis position = the Home position.

This is a transitional instruction:

- In relay ladder, toggle the rung from cleared to set each time the instruction should execute.
- In structured text, condition the instruction so that it only executes on a transition.

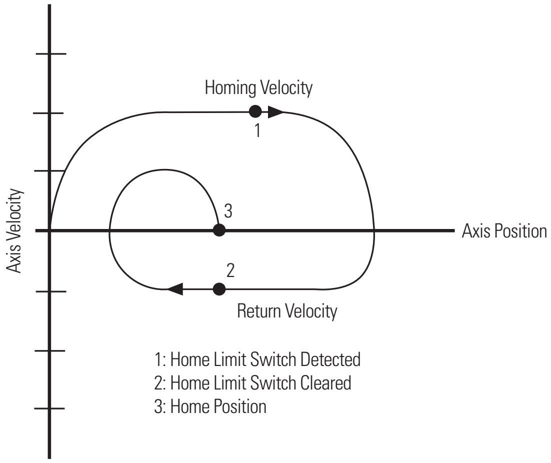
  See Motion Axis Attributes on page 223 for more information.

Examples

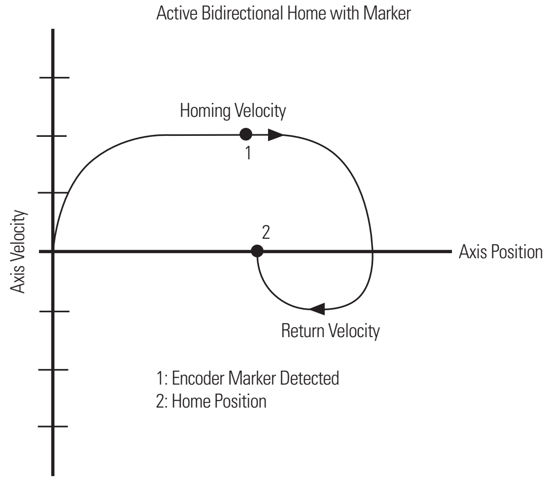
Active Homing

The following are Active Homing examples.

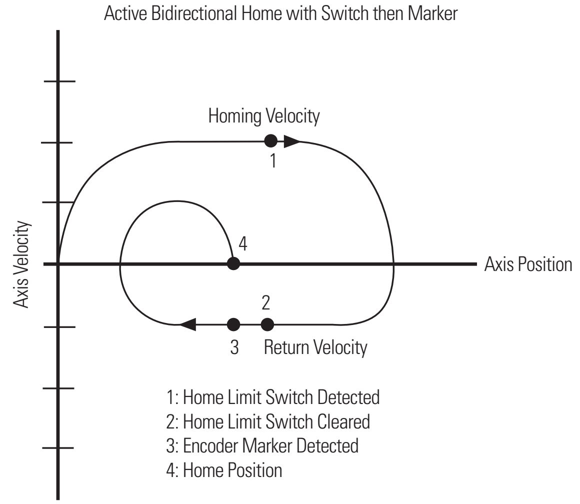
Active Homing Examples

Sequence	Description
Active immediate home	This sequence sets the axis position to the Home Position without moving the axis. If the axis isn't enabled, this sequence enables it. The feedback is always working; therefore enabled.
Active home to switch in forward bidirectional	The switch homing sequence is useful for multi-turn rotary and linear applications. Ensure that the home switch will be encountered in the direction of the home. If the axis is already ON the home limit switch, it can be past the switch, but not on the switch. Active Bidirectional Home with Switch  These steps occur during the sequence. 1. The axis moves in the Home Direction at the Home Speed to the home limit switch and decelerates to a stop (using the configured Maximum Deceleration Rate). It does not stop immediately. 2. If it is stopped as the Home Limit Switch, at position 2, when the sequence is started, it would already be at the home position. The axis reverses direction and moves at the Home Return Speed until it clears the home limit switch and then stops. 3. The axis moves back to the home limit switch or it moves to the Offset position. The axis moves at the Home Return Speed. If the axis is a Rotary Axis, the move back to the Home Position takes the shortest path (that is, no more than ½ revolution). If the axis is past the home limit switch at the start of the homing sequence, the axis reverses direction and starts the return leg of the homing sequence. Again, it decelerates to a stop. Use a Home Return Speed that is slower than the Home Speed to increase the homing accuracy. The accuracy of this sequence depends on the return speed and the delay to detect the transition of the home limit switch. Uncertainty = Home Return Speed x delay to detect the home limit switch. **Example**: Suppose your Home Return Speed is 0.1 in./s and it takes 10 ms to detect the home limit switch. Uncertainty = 0.1 in./s x 0.01 s = 0.001 in. The mechanical uncertainty of the home limit switch also affects the homing accuracy.

Active Homing Examples

Sequence	Description
Active home to marker in forward bidirectional	The marker homing sequence is useful for single-turn rotary and linear encoder applications because these applications have only one encoder marker for full axis travel.  These steps occur during the sequence. 1. The axis moves in the Home Direction at the Home Speed to the marker and decelerates to a stop. 2. The axis moves back to the marker or it moves to the Offset position. The axis moves at the Home Return Speed. If the axis is a Rotary Axis, the move back to the Home Position takes the shortest path (that is, no more than ½ revolution). The accuracy of this homing sequence depends on the homing speed and the delay to detect the marker transition. Uncertainty = Home Speed x delay to detect the marker. **Example**: Suppose your Home Speed is 1 in./s and it takes 1 µs to detect the marker. Uncertainty = 1 In./s x 0.000001 s = 0.000001 in.

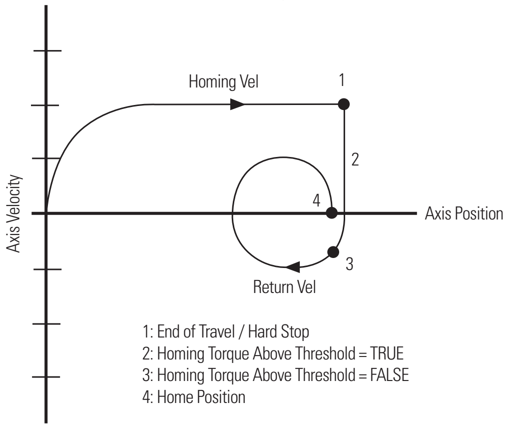
Active Homing Examples

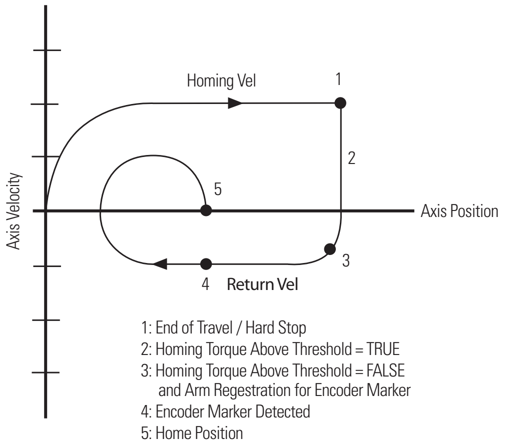
Sequence	Description
Active home to switch and marker in forward bidirectional	This is the most precise active homing sequence available.  These steps occur during the sequence. 1. The axis moves in the Home Direction at the Home Speed to the home limit switch and decelerates to a stop. 2. The axis reverses direction and moves at the Home Return Speed until it clears the home limit switch. 3. The axis keeps moving at the Home Return Speed until it gets to the marker. 4. The axis moves back to the marker or it moves to the Offset position. The axis moves at the Home Return Speed. If the axis is a Rotary Axis, the move back to the Home Position takes the shortest path (that is, no more than ½ revolution). If the axis is ON the home limit switch at the start of the homing sequence, the axis reverses direction and starts the return leg of the homing sequence.
Active home to switch in forward unidirectional	This active homing sequence is useful for when an encoder marker is not available and either unidirectional motion is required or proximity switch is being used. These steps occur during the sequence. 1. The axis moves in the Home Direction at the Home Speed to the home limit switch. 2. A decel position in calculated using the Home Offset and the decel distance. The axis keeps moving to the decel position and then decelerates to a stop. 3. The axis moves to the Home Offset position if it's in the same direction as the Home Direction. • When you position mode is linear, it decelerates to a stop. You may not be at you home position but you are correctly referenced to your home position. • When your position mode is Rotary, the rotary turns as many time that it needs to decelerate and finish at the home position.

Active Homing Examples

Sequence	Description
Active home to marker in forward unidirectional	This active homing sequence is useful for single-turn rotary and linear encoder applications when unidirectional motion is required. These steps occur during the sequence. 1. The axis moves in the Home Direction at the Home Speed to the marker. 2. If the axis is linear, it decelerates to a stop, unless the home offset is greater then the distance required to decelerate; then the home offset is applied. If the axis is rotary, it add as many revolutions as necessary so it will decelerate and stop at the home position. 3. The axis moves to the Home Offset position if it's in the same direction as the Home Direction.
Active home to switch and marker in forward unidirectional	This active homing sequence is useful for multi-turn rotary applications when unidirectional motion is required. These steps occur during the sequence. 1. The axis moves in the Home Direction at the Home Speed to the home limit switch. 2. The axis keeps moving at the Home Speed until it gets to the marker. 3. If the axis is linear, it decelerates to a stop, unless the home offset is greater then the distance required to decelerate; then the home offset is applied. If the axis is rotary, it add as many revolutions as necessary so it will decelerate and stop at the home position.

Active Homing Examples

<table>
<tr><th>Sequence</th><th>Description</th></tr>
<tr><td>Active Home to Torque</td><td>
<p>The Home to Torque Level sequence is a type of homing used when a hard stop is going to be used as the home position, as in a linear actuator. The occurrence of the hard stop is detected by the drive when the output torque to the motor reaches or exceeds the torque level specified by the user. Since the home to torque level sequence relies on the mechanical end of travel for operation, Unidirectional homing will not be possible so only Forward Bidirectional and Reverse Bidirectional are allowed.</p>
<p>In Torque Level homing, the torque event is the trigger. The motion planner decelerates the axis to a stop and reverses direction. The torque event is usually some type of hard stop. Because of this, the physical axis can't move, but the position command is changing. This causes the Position error to increase. If the distance required to decelerate is greater then the Position error Tolerance an Excessive Position error exception can occur, possibly cancelling the home operation.</p>
<p>A delay filter is implemented in the drive to reduce any false/nuisance triggers when there is a spike in the torque feedback upon enabling or jogging the motor under load.</p>
<p>Torque Level homing is very similar to Home Switch homing, with the exception that the torque level is used instead of the home switch input. This graphic depicts the Position/Velocity for Torque Level Homing.</p>
<p>Torque Level Homing</p>


<p>Torque Level-Marker homing is very similar to Home Switch-Marker homing, with the exception that the torque level is used instead of the home switch input. This graphic depicts the Position/Velocity for Torque Level-Marker Homing.</p>
<p>Torque Level - Marker Homing</p>


</td></tr>
</table>

Passive Homing

Passive Homing Examples

Sequence	Description
Passive Immediate Home	This is the simplest passive homing sequence type. When this sequence is performed, the controller immediately assigns the Home Position to the current axis actual position. This homing sequence produces no axis motion.
Passive Home with Switch	This passive homing sequence is useful for when an encoder marker is not available or a proximity switch is being used. When this sequence is performed in the Passive Homing Mode, an external agent moves the axis until the home switch is detected. The position is always preset to the Home position plus Offset Value at the moment when the switch is hit. The Offset value should be set to 0 if no Home Offset offset is wanted.
Passive Home with Marker	This passive homing sequence is useful for single-turn rotary and linear encoder applications. When this sequence is performed in the Passive Homing Mode, an external agent moves the axis until the marker is detected. The position is always preset to the Home position plus Offset Value at the moment when the switch is hit. The Offset value should be set to 0 if no Home Offset offset is wanted.
Passive Home with Switch then Marker	This passive homing sequence is useful for multi-turn rotary applications. When this sequence is performed in the Passive Homing Mode, an external agent moves the axis until the home switch and then the first encoder marker is detected. The position is always preset to the Home position plus Offset Value at the moment when the switch is hit. The Offset value should be set to 0 if no Home Offset offset is wanted.

Homed Status

The Homed Status bit is set by the MAH instruction upon successful completion of the configured homing sequence. This bit indicates that an absolute machine reference position has been established. When this bit is set, operations that require a machine reference, such as Software Overtravel checking can be meaningfully enabled.

For non-CIP Drive axis data types, the HomedStatus bit is cleared under the following conditions:

- Download
- Control power cycle
- Re-connection to Motion Module
- Feedback Loss Fault
- Shutdown

Feedback Integrity

This bit, when set, indicates that the feedback device is accurately reflecting axis position. The bit is set at power-up assuming that the feedback device passes any power-up self test required. If during operation a feedback exception occurs that could impact the fidelity of axis position, the bit is immediately cleared. The bit remains clear until either a fault reset is executed by the drive or the drive is power cycled. Note that the Feedback Integrity bit behavior applies to both absolute and incremental feedback device operation.

Notes:

Appendix A

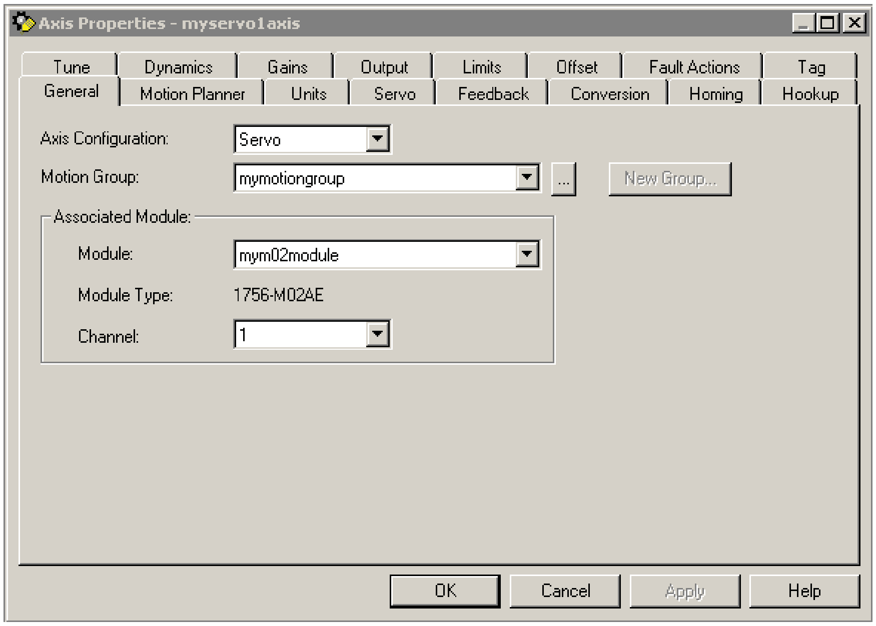
Axis Properties

Introduction

Use this appendix for a description of the properties of an axis.

General Tab – AXIS_SERVO

The General dialog box below is for an AXIS_SERVO data type.



Axis Configuration: Servo

Motion Group: mymotiongroup

Associated Module:

Module: mym02module

Module Type: 1756-M02AE

Channel: 1

General Tab – AXIS_SERVO Field Descriptions

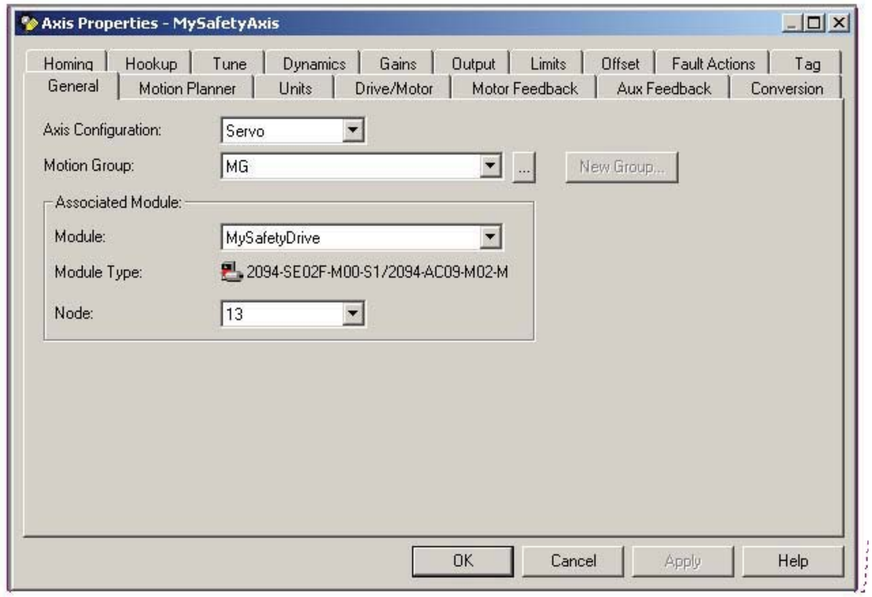
Item	Description	
Axis Configuration	Selects and displays the intended use of the axis.	
	Feedback Only	If the axis is to be used only to display position information from the feedback interface. This selection minimizes the display of axis properties tabs and parameters. The tabs for Servo, Tune, Dynamics, Gains, Output, Limits, and Offset are not displayed.
	Servo	If the axis is to be used for full servo operation. This selection maximizes the display of axis properties tabs and parameters.
Motion Group	Selects and displays the Motion Group to which the axis is associated. An axis assigned to a Motion Group appears in the Motion Groups branch of the Controller Organizer, under the selected Motion Group sub-branch. Selecting <none> terminates the Motion Group association, and moves the axis to the Ungrouped Axes sub-branch of the Motions Groups branch.	
The Associated Module selection (selected on the General tab), determines available catalog numbers.		

General Tab – AXIS_SERVO Field Descriptions

Item	Description
Module	Selects and displays the name of the motion module to which the axis is associated. Displays <none> if the axis is not associated with any motion module.
Module Type	Displays a module icon and the name of the SERCOS drive to which the axis is associated. Displays <none> if the axis is not associated with any drive. If the associated drive is a Kinetix Safety drive, a portion the module icon is red in color to signify its safety significance. For detailed information on wiring, configuring, and troubleshooting the safety functions of your Kinetix 6200 and Kinetix 6500 drives, see these publications: • Kinetix 6200 and Kinetix 6500 Control Modules Installation Instructions, publication 2094-IN012. • Kinetix 6200 and Kinetix 6500 Safe Speed Monitoring Safety Reference Manual, publication 2094-RM001. • Kinetix 6200 and Kinetix 6500 Safe Torque-off Safety Reference Manual, publication 2094-RM002.
Channel	Selects and displays the 1756-M02AE motion module channel - either 0 or 1 - to which the axis is assigned. Disabled when the axis is not associated with any motion module.

General Tab - AXIS_SERVO_DRIVE

The General screen shown below is for an AXIS_SERVO DRIVE Data Type.



Axis Configuration: Servo

Motion Group: MG

Module: MySafetyDrive

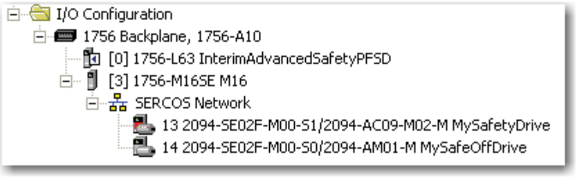
Module Type: 2094-SE02F-M00-S1/2094-AC09-M02-M

Node: 13

General Tab – AXIS_SERVO_DRIVE Field Descriptions

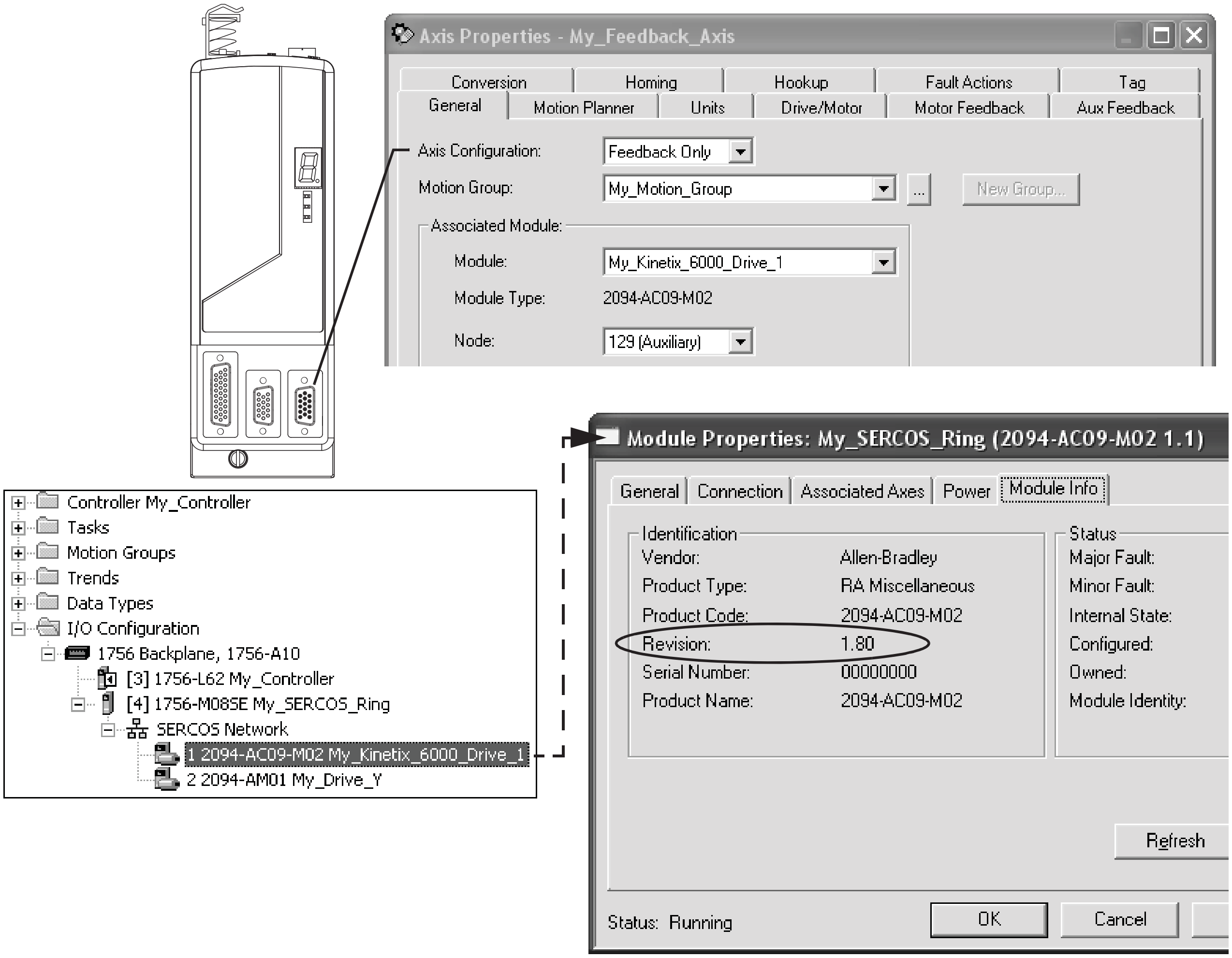
Item	Description	
Axis Configuration	Selects and displays the intended use of the axis.	
	Feedback Only	If the axis is to be used only to display position information from the feedback interface. This selection minimizes the display of axis properties tabs and parameters. The tabs for Tune, Dynamics, Gains, Output, Limits, and Offset are not displayed.
	Servo	If the axis is to be used for full servo operation. This selection maximizes the display of axis properties tabs and parameters.
Motion Group	Selects and displays the Motion Group to which the axis is associated. An axis assigned to a Motion Group appears in the Motion Groups branch of the Controller Organizer, under the selected Motion Group sub-branch. Selecting <none> terminates the Motion Group association, and moves the axis to the Ungrouped Axes sub-branch of the Motions Groups branch.	
Module	Selects and displays the name of the SERCOS drive to which the axis is associated. Displays <none> if the axis is not associated with any drive.	

General Tab – AXIS_SERVO_DRIVE Field Descriptions

Item	Description
Module Type	Displays a module icon and the name of the SERCOS drive to which the axis is associated. Displays <none> if the axis is not associated with any drive. If the associated drive is a Kinetix Safety drive, a portion the module icon is red in color to signify its safety significance. For detailed information on wiring, configuring, and troubleshooting the safety functions of your Kinetix 6200 and Kinetix 6500 drives, see these publications: • Kinetix 6200 (2094-SE02F-M00-S0, 2094-SE02F-M00-S1) and Kinetix 6500 (2094-EN02D-M01-S0, 2094-EN02D-M01-S1) Control Modules Installation Instructions, publication 2094-IN012. • Kinetix 6200 and Kinetix 6500 Safe Speed Monitoring Safety Reference Manual, publication 2094-RM001. Kinetix 6200 and Kinetix 6500 Safe Torque-off Safety Reference Manual, publication 2094-RM002. Following is a sample of the module icon (from the I/O configuration folder in RSLogix 5000 software) for: • Kinetix 6000 Advanced Safety Drive (S1) • Kinetix 6000 Enhanced Safe Torque-Off Drive (S0) 
Node	Displays the base node of the associated SERCOS drive. This is unavailable when the axis is not associated with any drive.

Node with a Kinetix 6000 Drive

If you need to use the auxiliary feedback port of a Kinetix 6000 drive as a feedback-only axis the drive must have firmware revision 1.80 or later.



Axis Properties - My_Feedback_Axis

Axis Configuration: Feedback Only

Motion Group: My_Motion_Group

Associated Module:

Module: My_Kinetix_6000_Drive_1

Module Type: 2094-AC09-M02

Node: 129 (Auxiliary)

Controller My_Controller
Tasks
Motion Groups
Trends
Data Types
I/O Configuration
  1756 Backplane, 1756-A10
    [3] 1756-L62 My_Controller
    [4] 1756-M08SE My_SERCOS_Ring
      SERCOS Network
        1 2094-AC09-M02 My_Kinetix_6000_Drive_1
        2 2094-AM01 My_Drive_Y

Module Properties: My_SERCOS_Ring (2094-AC09-M02 1.1)

Identification

Vendor: Allen-Bradley

Product Type: RA Miscellaneous

Product Code: 2094-AC09-M02

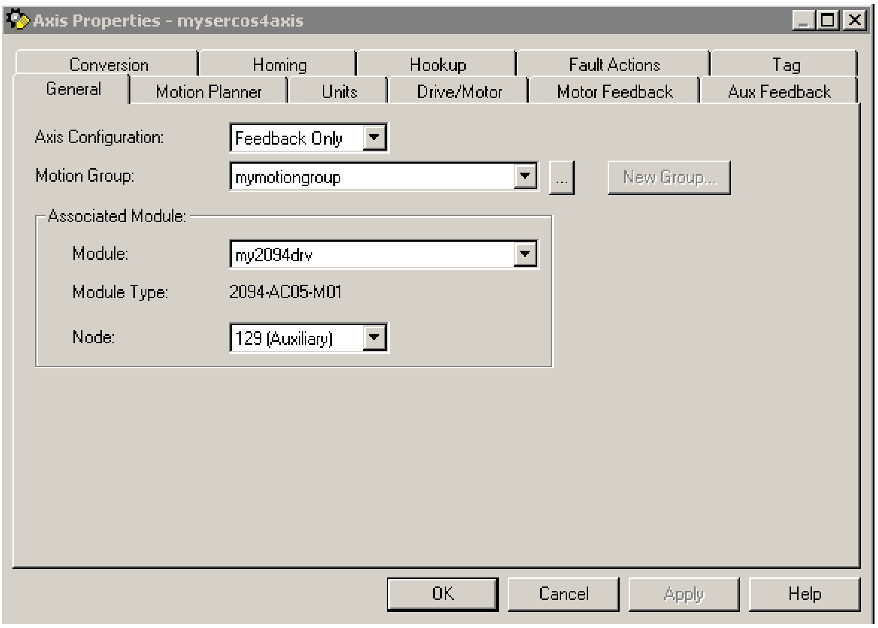
Revision: 1.80

Serial Number: 00000000

Product Name: 2094-AC09-M02

Status: Running

When a Kinetix 6000 drive is designated in the Associated Module box, there is an additional option for the Node value. It is the node associated with the drive plus 128 with (Auxiliary) after the number. The range is 129 to 234. When the Auxiliary Node assignment is chosen the axis configuration is changed to Feedback Only on the General tab and an asterisk (*) appears next to General.



Axis Configuration: Feedback Only

Motion Group: mymotiongroup

Module: my2094drv

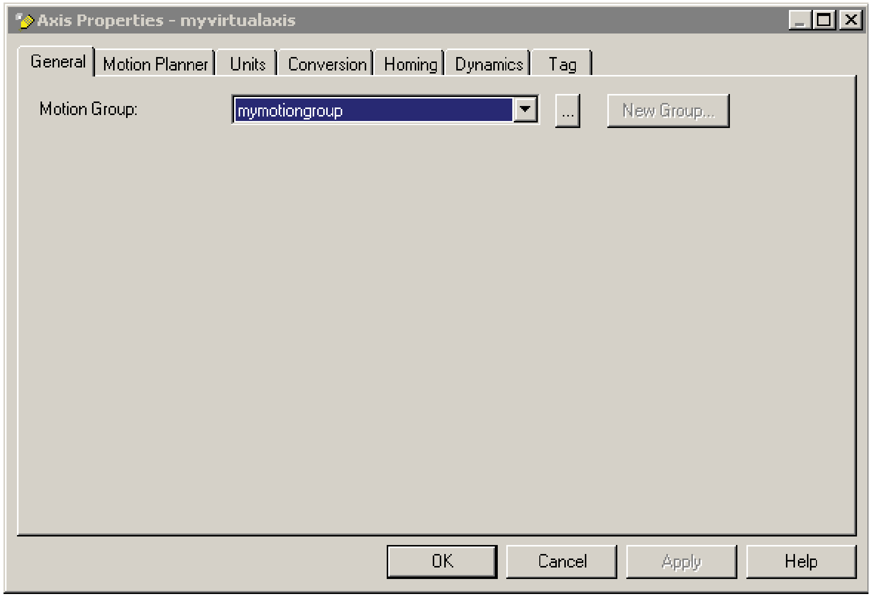
Module Type: 2094-AC05-M01

Node: 129 (Auxiliary)

This also places an asterisk (*) on the Aux Feedback tab and you must go there and choose the appropriate values. On the Drive/Motor tab the Loop Configuration is changed to Aux Feedback Only.

General Tab - AXIS_VIRTUAL

The AXIS_VIRTUAL General tab is shown below.



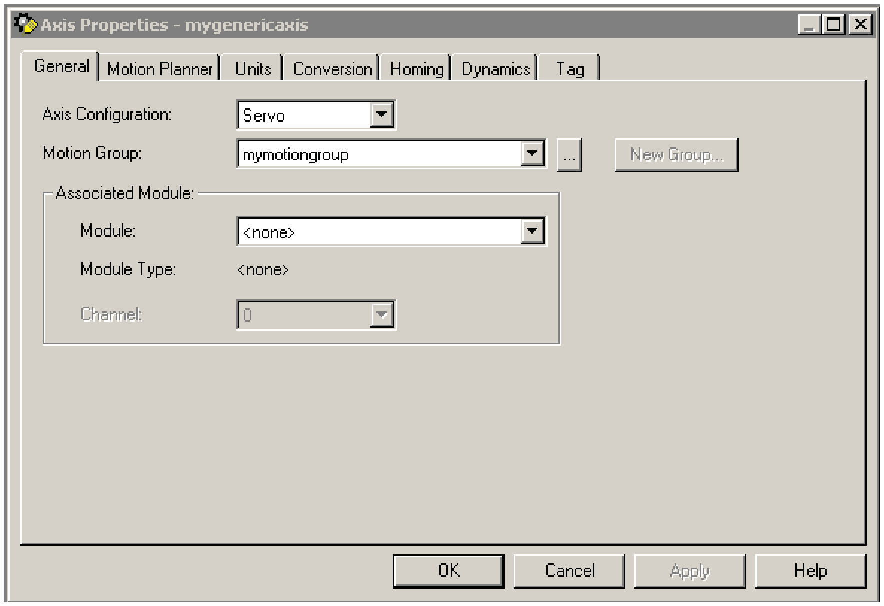
Motion Group: mymotiongroup

Motion Group

You can select the Motion Group that you want the axis to be associated with. An axis assigned to a Motion Group appears in the Motion Groups branch of the Controller Organizer. When you select 'none' it terminates the Motion Group association, and moves the axis to the Ungrouped Axes in the Controller Organizer.

General Tab – AXIS_GENERIC

The AXIS_GENERIC General tab is shown below.



Axis Configuration: Servo

Motion Group: mymotiongroup

Associated Module:

Module: <none>

Module Type: <none>

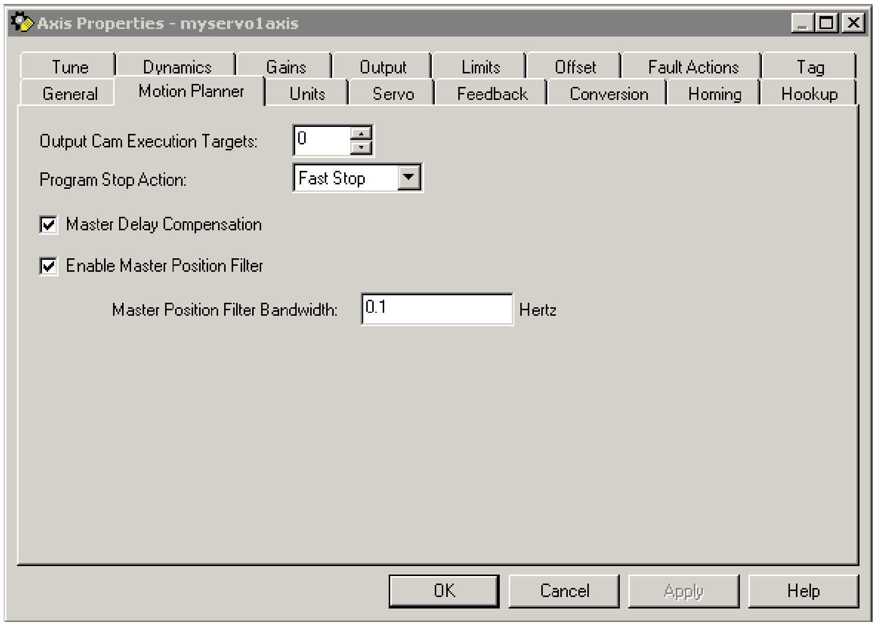
Channel: 0

General Tab – AXIS_GENERIC Field Descriptions

Item	Description	
Axis Configuration	Selects and displays the intended use of the axis.	
	Feedback Only	If the axis is to be used only to display position information from the feedback interface. This selection minimizes the display of axis properties tabs and parameters. The tab for Dynamics is not available
	Servo	If the axis is to be used for full servo operation. This selection maximizes the display of axis properties tabs and parameters.
Motion Group	Selects and displays the Motion Group to which the axis is associated. An axis assigned to a Motion Group appears in the Motion Groups branch of the Controller Organizer, under the selected Motion Group sub-branch. Selecting <none> terminates the Motion Group association, and moves the axis to the Ungrouped Axes sub-branch of the Motions Groups branch.	
Module	Selects and displays the name of the motion module to which the axis is associated. Displays <none> if the axis is not associated with any motion module.	
Channel	Selects and displays the motion module channel - either 0 or 1 - to which the axis is assigned. Disabled when the axis is not associated with any motion module.	

Motion Planner Tab

The Motion Planner tab is where you set/edit the number of Output Cam execution targets, the type of stop action to use, turn on/off the Master Delay Compensation, turn on/off the Master Position Filter, and set the bandwidth for Master Position Filter Bandwidth. The Motion Planner tab has the same fields regardless of the type of axis.



Output Cam Execution Targets: 0

Program Stop Action: Fast Stop

☑ Master Delay Compensation

☑ Enable Master Position Filter

Master Position Filter Bandwidth: 0.1 Hertz

Motion Planner Tab Field Descriptions

Item	Description
Output Cam Execution Targets	Determines how many Output Cam execution nodes (instances) are created for a specific axis. The Execution Target parameter for the MAOC/MDOC instructions specify which of the configured execution nodes the instruction is affecting. In addition, the number specified in the Axis Properties dialog box specifies the number of instances of Output Cam in which the value of zero means ‘none’, and the value specified for Execution Target in the MAOC instruction references a specific instance in which a value of zero selects the first instance.
Program Stop Action	Selects how a specific axis is stopped when the processor undergoes a mode change, or when an explicit. You can apply Program Stop Action when an MSG is programmed to Stop type.

Motion Planner Tab Field Descriptions

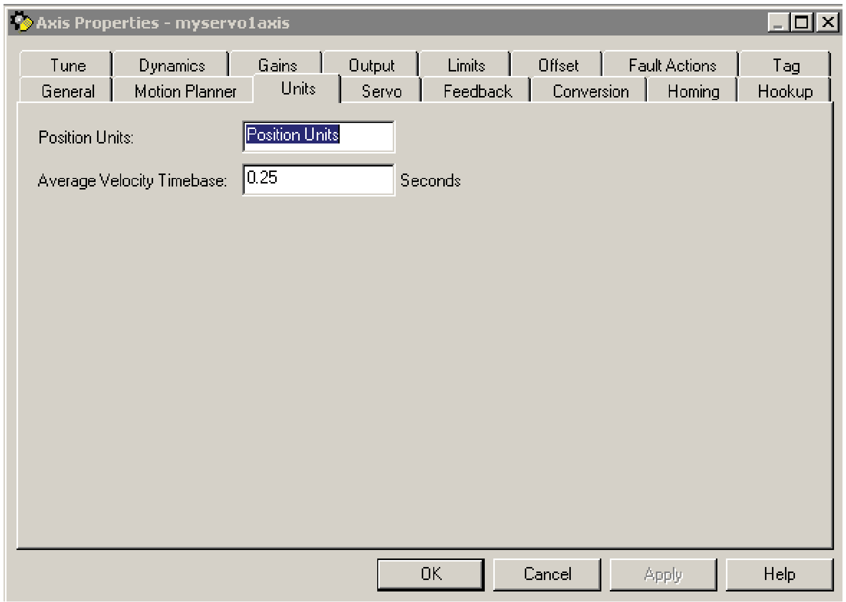
Item	Description	
	Fast Disable	The axis is decelerated to a stop using the current configured value for maximum deceleration. Servo action is maintained until the axis motion has stopped at which time the axis is turned off. (that is, Drive Enable is not checked, and Servo Action is not checked).
	Fast Shutdown	The axis is decelerated to a stop using the current configured value for maximum deceleration. Once the axis motion is stopped, the axis is placed in the shutdown state (that is, Drive Enable is not checked, Servo Action is not checked, and the OK contact is opened). To recover from this state, a Shutdown reset instruction must be executed.
	Fast Stop	The axis is decelerated to a stop using the current configured value for maximum deceleration. Servo action is maintained after the axis motion has stopped. This mode is useful for gravity or loaded systems, where servo control is needed at all times.
	Hard Disable	The axis is immediately disabled (that is, Drive Enable is disabled, Servo Action is disabled, but the OK contact is left closed). Unless the drive is configured to provide some form of dynamic breaking, this results in the axis coasting to a stop.
	Hard Shutdown	The axis is immediately placed in the shutdown state. Unless the drive is configured to provide some form of dynamic breaking, this results in the axis coasting to a stop. To recover from this state, a Shutdown reset instruction must be executed.

Motion Planner Tab Field Descriptions

Item	Description
Master Delay Compensation Checkbox	Enables or disables Master Delay Compensation. The default setting is Disabled. It must be checked to enable master delay compensation. Master Delay Compensation is used to balance the delay time between reading the master axis command position and applying the associated slave command position to the slave's servo loop. It ensures that the slave axis command position accurately tracks the actual position of the master axis; that is, zero tracking error. If the axis is configured for Feedback only, Master Delay Compensation should be disabled.
Enable Master Position Filter Checkbox	Enables or disables the Master Position Filter. The default is disabled. It must be checked to enable position filtering. Master Position Filter effectively filters the specified master axis position input to the slave axis's gearing or position camming operation. The filter smooths out the actual position signal from the master axis, and thus smooths out the corresponding motion of the slave axis. When this feature is enabled the Master Position Filter Bandwidth field is enabled.
Master Position Filter Bandwidth	The Master Position Filter Bandwidth field is enabled when the Enable Master Position Filter checkbox is selected. This field controls the bandwidth for master position filtering. Enter a value in Hz to set the bandwidth to for the Master Position Filter. Important: A value of zero for Master Position Filter Bandwidth effectively disables the master position filtering.

Units Tab

The Units tab is the same for all axis data types. Use this tab to determine the units to define your motion axis.



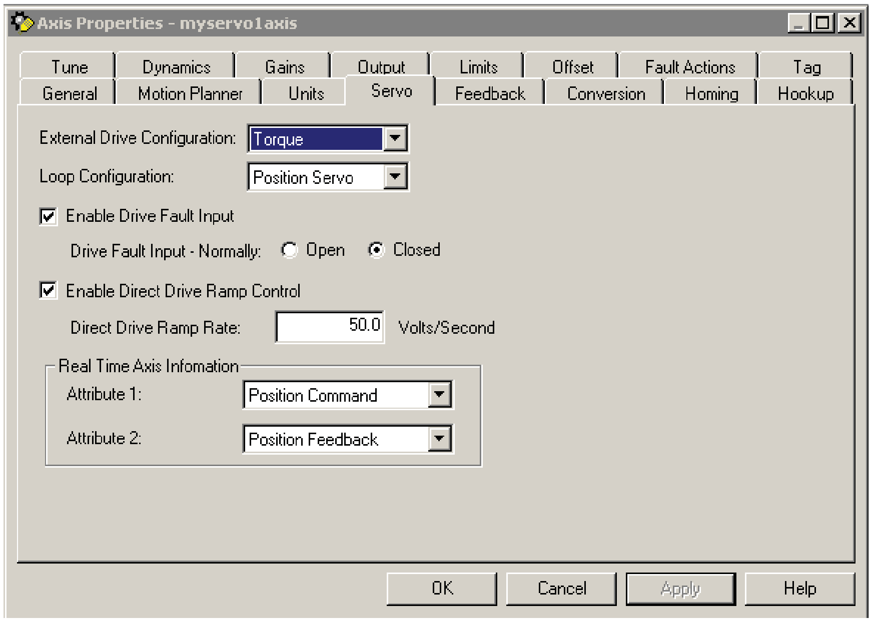
Position Units: Position Units

Average Velocity Timebase: 0.25 Seconds

Item	Description
Position Units	User-defined engineering units (rather than feedback counts) used for labeling all motion-related values, for example, position and velocity. These position units can be different for each axis. Position Units should be chosen for maximum ease of use in your application. For example, linear axes might use position units of Inches, Meters, or mm whereas rotary axes might use units of Revs or Degrees.
Average Velocity Timebase	Specifies the time (in seconds) to be used for calculating the average velocity of the axis. This value is computed by taking the total distance the axis travels in the amount of time specified, and dividing this value by the timebase. The average velocity timebase value should be large enough to filter out the small changes in velocity that would result in a noisy velocity value, but small enough to track significant changes in axis velocity. A value of 0.25 to 0.50 seconds should work well for most applications.

Servo Tab - AXIS_SERVO

Click the Servo tab from the Axis Properties for AXIS_SERVO to access the Servo dialog box.



External Drive Configuration: Torque

Loop Configuration: Position Servo

☑ Enable Drive Fault Input

Drive Fault Input - Normally: ○ Open ● Closed

☑ Enable Direct Drive Ramp Control

Direct Drive Ramp Rate: 50.0 Volts/Second

Real Time Axis Infomation

Attribute 1: Position Command

Attribute 2: Position Feedback

Servo Tab - AXIS_SERVO Tab Field Descriptions

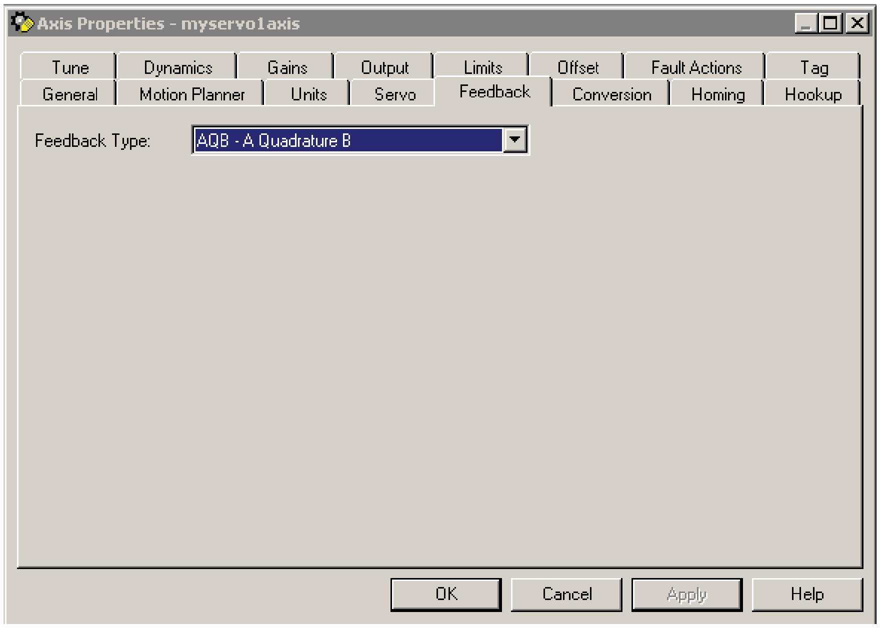
Item	Description
External Drive Configuration	Select the drive type for the servo loop: • Velocity - disables the servo module’s internal digital velocity loop. • Torque - the servo module’s internal digital velocity loop is active, which is the required configuration for interfacing the servo axis to a torque loop drive. • Hydraulic - enables features specific to hydraulic servo applications.
Loop Configuration	Select the configuration of the servo loop. For this release, only Position Servo is available.
Enable Drive Fault Input	Check this box if you wish to enable the Drive Fault Input. When active the motion module receives notice whenever the external drive detects a fault.
Drive Fault Input	Specifies the usual state of the drive fault input when a fault is detected on the drive. • Normally Open – when a drive fault is detected it opens its drive fault output contacts. • Normally Closed – when a drive fault is detected it closes its drive fault output contacts.

Servo Tab - AXIS_SERVO Tab Field Descriptions

Item	Description
Enable Direct Drive Ramp Control	Clicking on the Enable Direct drive Ramp Control check box lets you set the Direct Drive Ramp Rate in volts per second for when an MDO instruction is executed.
Direct Drive Ramp Rate	The Direct Drive Ramp Rate is a slew rate for changing the output voltage when a Direct Drive On (MDO) instruction is executed. A Direct Drive Ramp Rate of 0 disables the output rate limiter letting the Direct Drive On voltage to be applied directly.
Attribute 1/Attribute 2	Select up to two axis attributes whose status are transmitted with, for example, the actual position data to the Logix processor. The values of the selected attributes can be accessed via the standard GSV or Get Attribute List service. They can also be access using templated data. The servo status data is updated each coarse update period. If a GSV is done to one of these servo status attributes without having selected this attribute via the Drive Info Select attribute, the attribute value is static and does not reflect the true value in the servo module.

Feedback Tab – (AXIS_SERVO)

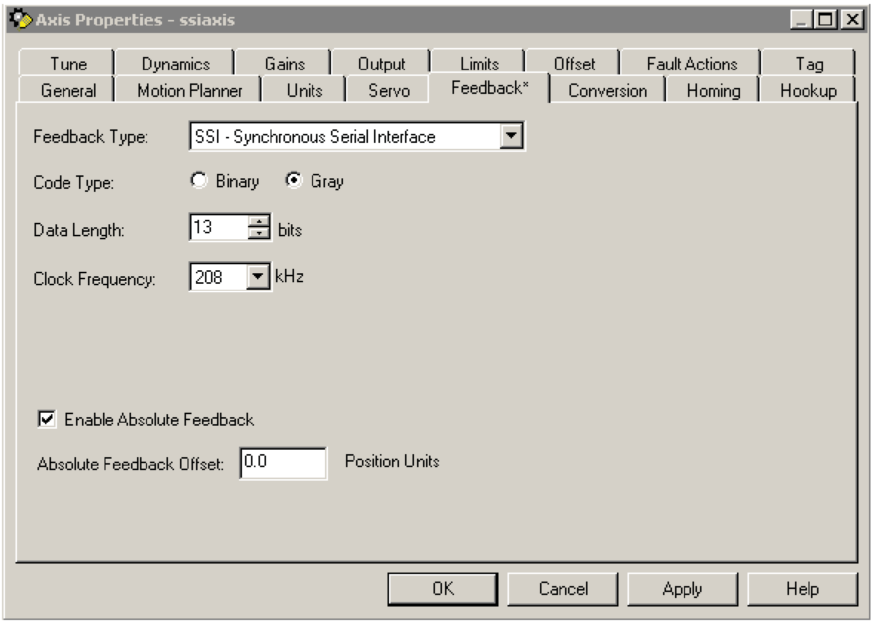
The Feedback tab lets you to select the type of Feedback used with your Servo axis.



Feedback Type: AQB - A Quadrature B

Feedback Tab (AXIS_SERVO) Field Descriptions

Item	Description
Feedback Type	Select the appropriate Feedback for your current configuration. Your options are dependent upon the motion module to which the axis is associated.
A Quadrature B Encoder Interface (AQB)	The 1756-M02AE servo module provides interface hardware to support incremental quadrature encoders equipped with standard 5-Volt differential encoder-interface signals. The AQB option has no associated attributes to configure.
Synchronous Serial Interface (SSI)	The 1756-M02AS servo module provides an interface to transducers with Synchronous Serial Interface (SSI) outputs. SSI outputs use standard 5V differential signals (RS422) to transmit information from the transducer to the controller. The signals consist of a Clock generated by the controller and Data generated by the transducer.
Linear Displacement Transducer (LDT)	The 1756-HYD02 Servo module provides an interface to the Linear Magnetostrictive Displacement Transducer, or LDT. A Field Programmable Gate Array (FPGA) is used to implement a multi-channel LDT Interface. Each channel is functionally equivalent and is capable of interfacing to an LDT device with a maximum count of 240,000. The LDT interface has transducer failure detection and digital filtering to reduce electrical noise. The Feedback screen changes in appearance depending on the selected Feedback Type. When the servo axis is associated with a 1756-M02AS motion module the only Feedback Type available is SSI-Synchronous Serial Interface and the Feedback tab dialog box looks like the following illustration.



Feedback Type: SSI - Synchronous Serial Interface

Code Type: ○ Binary ● Gray

Data Length: 13 bits

Clock Frequency: 208 kHz

☑ Enable Absolute Feedback

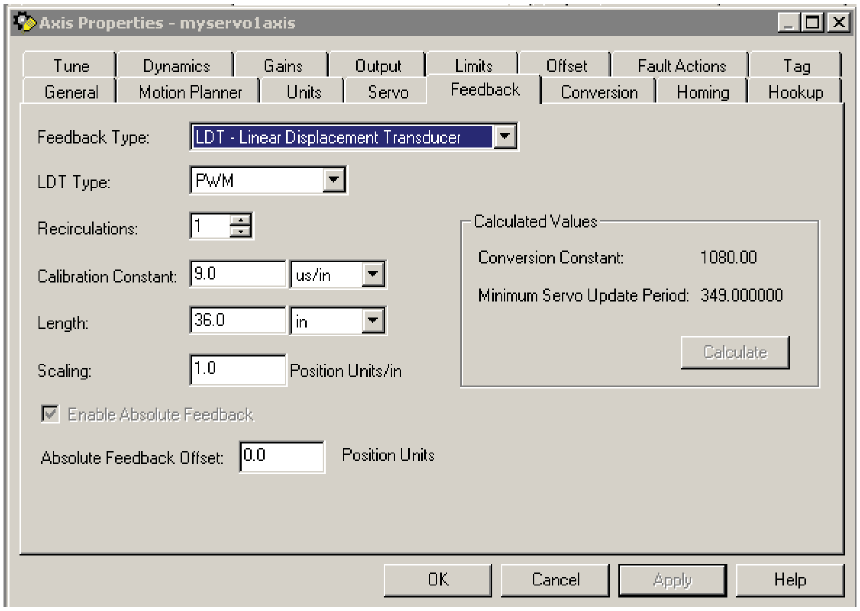
Absolute Feedback Offset: 0.0 Position Units

Feedback Tab (AXIS_SERVO) Field Descriptions

Item	Description
Code Type	The type of code, either Binary or Gray, used to report SSI output. If the module's setting does not match the feedback device, the positions jump around erratically as the axis moves.
Data Length	The length of output data in a specified number of bits between 8 and 31. The data length for the selected feedback device can be found in its specifications.

Feedback Tab (AXIS_SERVO) Field Descriptions

Item	Description
Clock Frequency	Sets the clock frequency of the SSI device to either 208 (default) or 625 kHz. When the higher clock frequency is used, the data from the feedback device is more recent, but the length of the cable to the transducer must be shorter than with the lower frequency.
Enable Absolute Feedback	The default is enabled (checked). If Enable Absolute Feedback is set, the servo module adds the Absolute Feedback Offset to the current position of the feedback device to establish the absolute machine reference position. Absolute feedback devices retain their position reference even through a power-cycle, therefore the machine reference system can be restored at powerup.
Absolute Feedback Offset	If Absolute feedback is enabled, this field becomes active. You can enter the amount of offset, in position units, to be added to the current position of the Feedback device. The SSI is an absolute feedback device. To establish an appropriate value for the Offset, the MAH instruction can be executed with the Home Mode set to Absolute. When executed, the module computes the Absolute Feedback Offset as the difference between the configured value for Home Position and the current absolute feedback position of the axis. The computed Absolute Feedback Offset is immediately applied to the axis upon completion of the MAH instruction. The actual position of the axis is re-referenced during execution of the MAH instruction therefore, the servo loop must not be active. If the servo loop is active, the MAH instruction errors. When the Enable Absolute Feedback is disabled, the servo module ignores the Absolute Feedback Offset and treats the feedback device as an incremental position transducer. A homing or redefine position operation is required to establish the absolute machine reference position. The Absolute Home Mode is invalid. If using Single-turn or Multi-turn Absolute SSI Feedback transducers, see the Homing tab information for important details concerning Absolute feedback tranducer's marker reference. When the servo axis is associated to a 1756-HYD02 motion module, then LDT - Linear Displacement Transducer is the only option for Feedback Type.



Feedback Type: LDT - Linear Displacement Transducer

LDT Type: PWM

Recirculations: 1

Calibration Constant: 9.0 us/in

Length: 36.0 in

Scaling: 1.0 Position Units/in

☑ Enable Absolute Feedback

Absolute Feedback Offset: 0.0 Position Units

Calculated Values

Conversion Constant: 1080.00

Minimum Servo Update Period: 349.000000

Feedback Tab (AXIS_SERVO) Field Descriptions

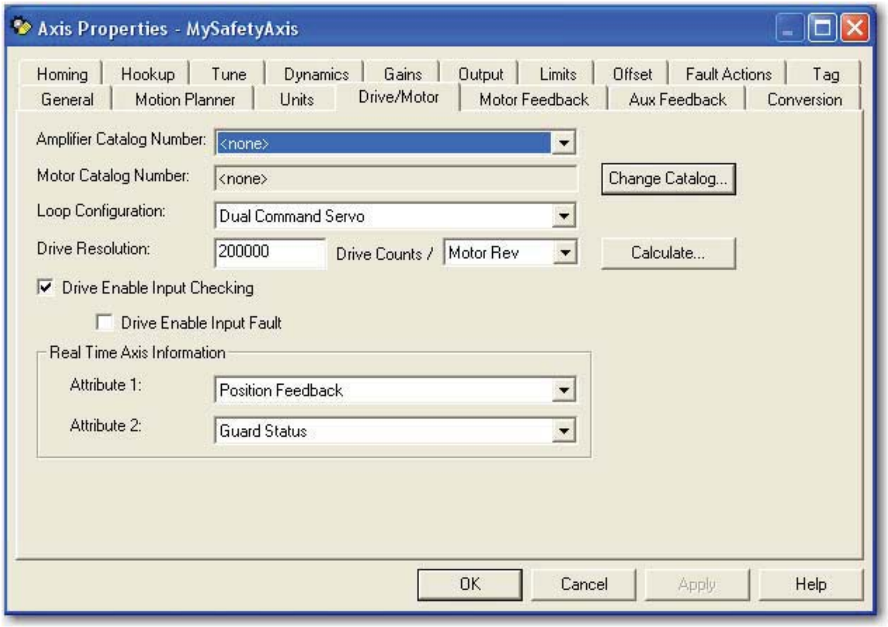
Item	Description
LDT Type	This field selects the type of LDT to use to provide feedback to the Hydraulic module. The available types are PWM, Start/Stop Rising, or Start/Stop Falling.
Recirculations	Use this field to set the number of repetitions to use to acquire a measurement from an LDT.
Calibration Constant	This is a number that is engraved on the LDT by the manufacturer. It specifies the characteristics of the individual LDT. Each LDT has its own calibration constant therefore, if you change the LDT, you must change the Calibration constant.
Length	Defines the stroke of travel of the hydraulic cylinder. The length value is used with the number of recirculations to determine the minimum servo update period.
Scaling	Scaling defines the relationship between the LDT unit of measure (length field) and the unit of measure defined at the Units tab.
Enable Absolute Feedback	This field is grayed out because it is always active when Feedback Type is LDT.

Feedback Tab (AXIS_SERVO) Field Descriptions

Item	Description	
Absolute Feedback Offset	Enter the amount of offset, in position units, to be added to the current position of the LDT. The LDT is an absolute feedback device. To establish an appropriate value for the Offset, the MAH instruction can be executed with the Home Mode set to Absolute. When executed, the module computes the Absolute Feedback Offset as the difference between the configured value for Home Position and the current absolute feedback position of the axis. The computed Absolute Feedback Offset is immediately applied to the axis upon completion of the MAH instruction. The actual position of the axis is re-referenced during execution of the MAH instruction therefore, the servo loop must not be active. If the servo loop is active, the MAH instruction errors. When the Enable Absolute Feedback is disabled, the servo module ignores the Absolute Feedback Offset and treats the feedback device as an incremental position transducer. A homing or redefine position operation is required to establish the absolute machine reference position. The Absolute Home Mode is invalid.	
Calculated Values	Conversion Constant	The Conversion Constant is calculated from the values entered on the Feedback screen when you click Calculate. This calculated value must be typed into the Conversion Constant field on the Conversion tab as it is not automatically updated.
	Minimum Servo Update Period	The Minimum Servo Update period is calculated based on the values entered for Recirculations and Length on the Feedback tab. When these values are changed, clicking Calculate recalculates the Minimum Servo Update Period based on the new values.
	Calculate Button	Calculate becomes active whenever you make changes to the values on the Feedback tab. Clicking Calculate recalculates the Conversion Constant and Minimum Servo Update Period values. However, you must then reenter the Conversion Constant value at the Conversion tab as the values are not updated automatically.

Drive/Motor Tab - (AXIS_SERVO_DRIVE)

Use this tab to configure the servo loop for an AXIS_SERVO_DRIVE axis, and open the Change Catalog dialog box.



Amplifier Catalog Number: <none>

Motor Catalog Number: <none>

Loop Configuration: Dual Command Servo

Drive Resolution: 200000 Drive Counts / Motor Rev

☑ Drive Enable Input Checking

☐ Drive Enable Input Fault

Real Time Axis Information

Attribute 1: Position Feedback

Attribute 2: Guard Status

Drive/Motor Tab - (AXIS_SERVO_DRIVE) Field Descriptions

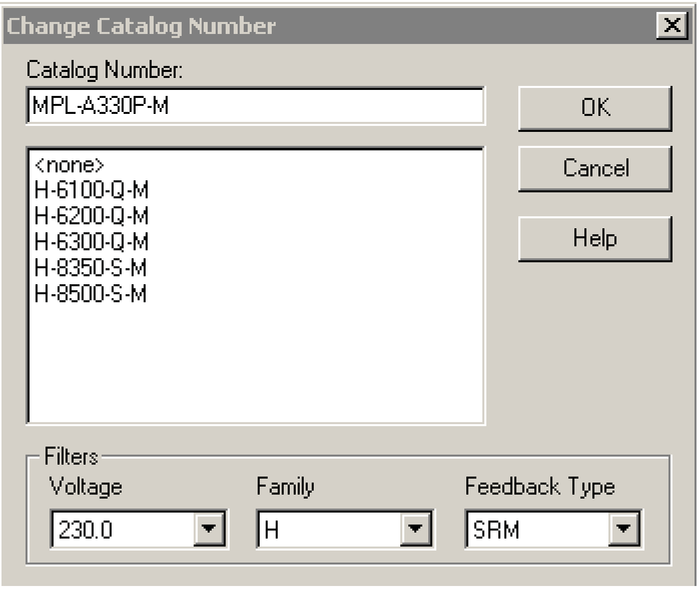
Item	Description
Amplifier Catalog Number	Select the catalog number of the amplifier to which this axis is connected.
Motor Catalog Number	Select the catalog number of the motor associated with this axis. When you change a Motor Catalog Number, the controller recalculates the values.
Change Catalog…	Change Catalog accesses the motor database and provides for selecting a new motor catalog number. There are three boxes that can be used for refine the selection process.

Drive/Motor Tab - (AXIS_SERVO_DRIVE) Recalculations

On this dialog box	These attributes are recalculated
Motor Feedback tab	Motor Feedback Type Motor Feedback Resolution
Gains tab	Position Proportional Gains Velocity Proportional Gains
Dynamics tab	Maximum Speed Maximum Acceleration Maximum Deceleration
Limits tab	Position Error Tolerance
Custom Stop Action Attributes dialog box	Stopping Torque

Drive/Motor Tab - (AXIS_SERVO_DRIVE) Recalculations

On this dialog box	These attributes are recalculated
Custom Limit Attributes dialog box	Velocity Limit Bipolar Velocity Limit Positive Velocity Limit Negative Acceleration Limit Bipolar Acceleration Limit Positive Acceleration Limit Negative Torque Limit Bipolar Torque Limit Positive Torque Limit
Tune Bandwidth dialog box	Position Loop Bandwidth Velocity Loop Bandwidth
Loop Configuration	Select the configuration of the servo loop: • Motor Feedback Only – Displayed when Axis Configuration is Feedback only • Aux Feedback Only – Displayed when Axis Configuration is Feedback only • Position Servo • Aux Position Servo (not applicable to Ultra3000 drives) • Dual Position Servo • Dual Command Servo • Aux Dual Command Servo • Velocity Servo • Torque Servo • Dual Command/Feedback Servo

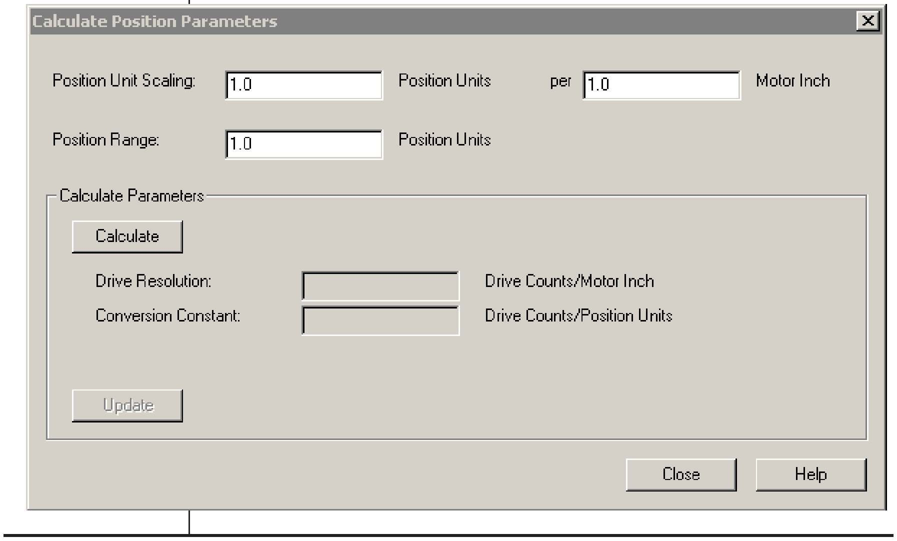


Catalog Number: MPL-A330P-M

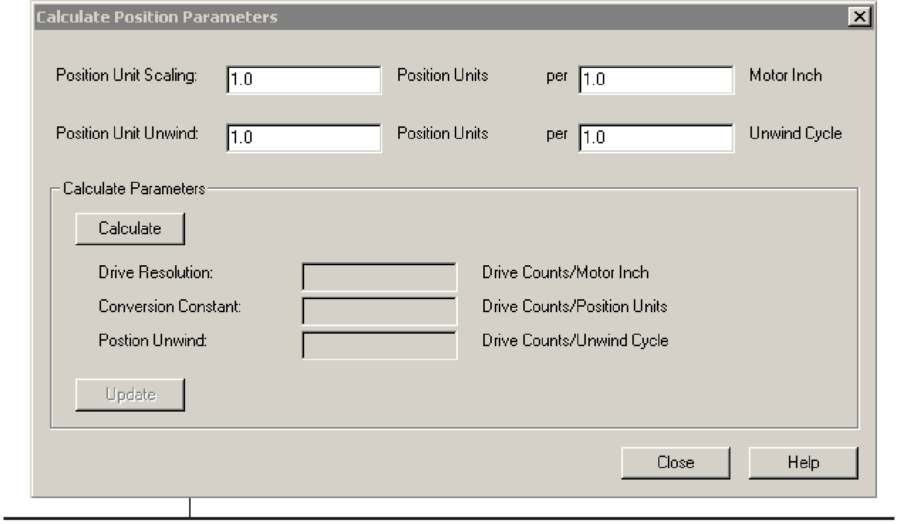
<none>
H-6100-Q-M
H-6200-Q-M
H-6300-Q-M
H-8350-S-M
H-8500-S-M

Voltage: 230.0
Family: H
Feedback Type: SRM

Item	Description	
Catalog Number	Lists the available catalog numbers from the Motor Database based on any selection criteria from the Filters fields.	
Filters	There are three optional Filter fields that allow you to refine your search of the Motor Database. The Filter boxes are defaulted to all.	
	Voltage	Lets you chooses a voltage rating from the pull-down menu to broaden or narrow your search. The default is all.
	Family	The Family filter box pull down list lets you narrow your motor search by restricting it to a particular family of motors. The default is all.
	Feedback Type	The Feedback Type filter box pull-down menu lets you manipulate your motor search by acceptable Feedback types. The default is all.
Position Unit Scaling	Position Unit Scaling defines the relationship between the Position Units defined on the Units tab and the units selected to measure position.	
Per	The units used for Position Unit Scaling. The options are: Motor Inch, Motor Millimeter, or Motor Rev	
Position Range	Maximum travel limit that your system can go.	
Position Unit Unwind	For Rotary applications, the Position Unit Unwind field appears. Enter the value for the maximum number of unwinds in position units per unwind cycle.	
Calculate Parameters	The Calculate Parameters shows the values that are to be calculated based upon the values entered for the Position Unit Scaling and Position Range.	

Item	Description
Drive Resolution	Recalculates the resolution based upon the new values entered on this screen.
Calculate…	Calculate takes you to an input screen that is designed to calculate the Drive Resolution and Conversion Constant based upon your input for Position Unit Scaling and Position Range for Linear Positioning mode. If you are in Rotary Positioning Mode then it calculates the Drive Resolution, Conversion Constant, and Position Unwind based upon your inputs for Position Unit Scaling and Position Unit Unwind. When the Conversion screen has Linear as the value for Position Mode, clicking Calculate displays the following screen.  Calculate Position Parameters Position Unit Scaling: 1.0 Position Units per 1.0 Motor Inch Position Range: 1.0 Position Units Drive Resolution: Drive Counts/Motor Inch Conversion Constant: Drive Counts/Position Units

Item	Description
Conversion Constant	Recalculates the Conversion Constant based upon the new values entered on this screen.When you edit the setting for either the Conversion Constant or the Drive Resolution, and then click either OK or Apply, you must choose whether to also recalculate the settings for these dependent attributes. The following attributes are recalculated. **On the Dynamics tab:** • Maximum Velocity • Maximum Acceleration • Maximum Deceleration **On the Limits tab:** • Position Error Tolerance **On the Custom Drive Scaling Attributes dialog:** • Torque Data Scaling **On the Custom Limit Attributes dialog:** • Velocity Limit Bipolar • Velocity Limit Positive • Velocity Limit Negative • Acceleration Limit Bipolar • Acceleration Limit Positive • Acceleration Limit Negative When the Conversion screen has Rotary as the value for Position Mode, Clicking Calculate displays the following screen.



Calculate Position Parameters

Position Unit Scaling: 1.0 Position Units per 1.0 Motor Inch

Position Unit Unwind: 1.0 Position Units per 1.0 Unwind Cycle

Calculate Parameters

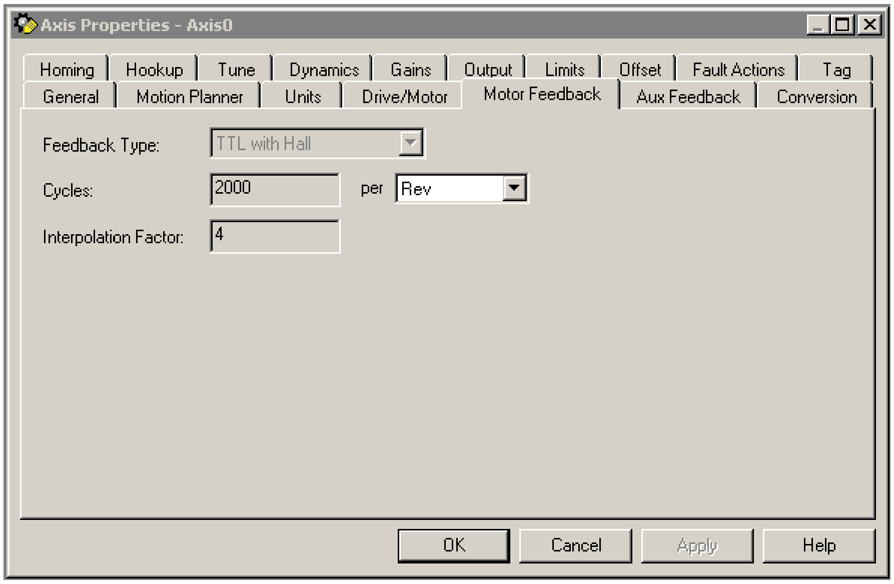
Drive Resolution: Drive Counts/Motor Inch

Conversion Constant: Drive Counts/Position Units

Postion Unwind: Drive Counts/Unwind Cycle

Motor Feedback Tab - AXIS_SERVO_DRIVE

Use this tab to configure motor and auxiliary feedback device (if any) parameters, for an axis of the type AXIS_SERVO_DRIVE.



Feedback Type: TTL with Hall

Cycles: 2000 per Rev

Interpolation Factor: 4

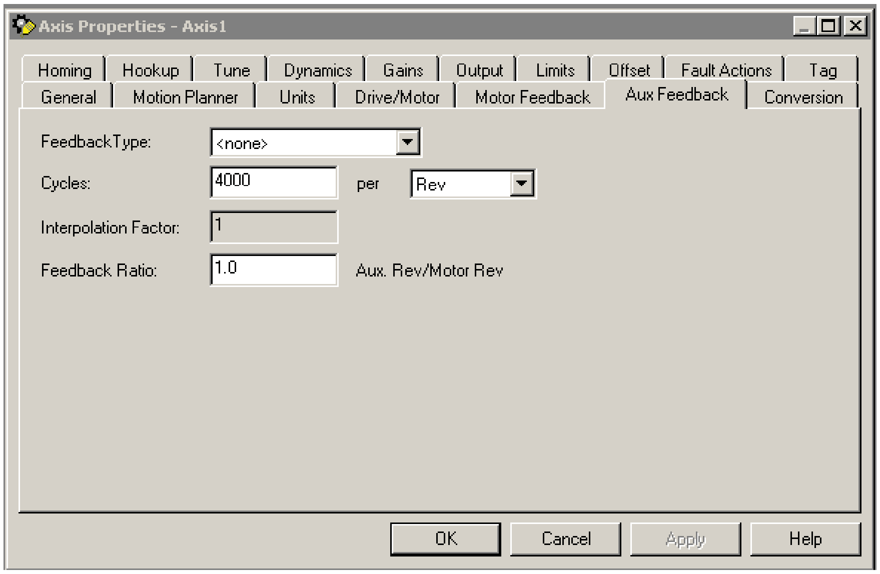
The Axis Configuration selection made on the General tab, and the Loop Configuration selection made on the Drive tab determine which sections of this dialog box – Motor and Auxiliary Feedback – are enabled.

Item	Description
Feedback Type	This field displays the type of feedback associated with the selected motor.
Cycles	The number of cycles of the associated feedback device. This helps the Drive Compute Conversion constant used to convert drive units to feedback counts. Depending on the feedback type you select, this value may be either read-only or editable.
Per	The units used to measure the cycles.
Interpolation Factor	This field displays a fixed, read-only value for each feedback type. This value is used to compute the resolution of the feedback device.

Aux Feedback Tab - AXIS_SERVO_DRIVE

The Auxiliary Feedback tab is enabled only if the Drive tab's Loop Configuration field is set to Aux Feedback Only, Aux Position Servo, Dual Position Servo, Dual Command Servo, or Aux Dual Command Servo.

Use this tab to configure motor and auxiliary feedback device (if any) parameters, for an axis of the type AXIS_SERVO_DRIVE.



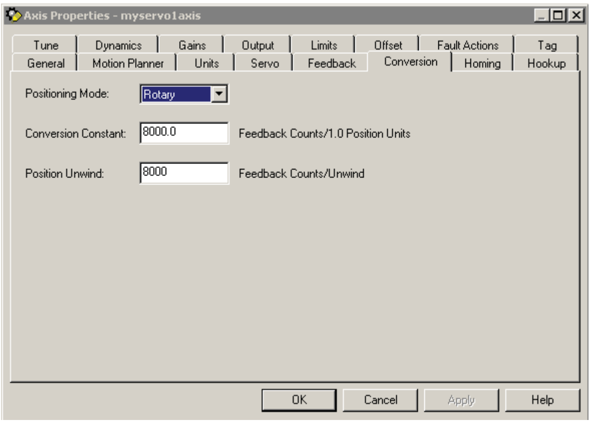
Feedback Type: <none>

Cycles: 4000 per Rev

Interpolation Factor: 1

Feedback Ratio: 1.0 Aux. Rev/Motor Rev

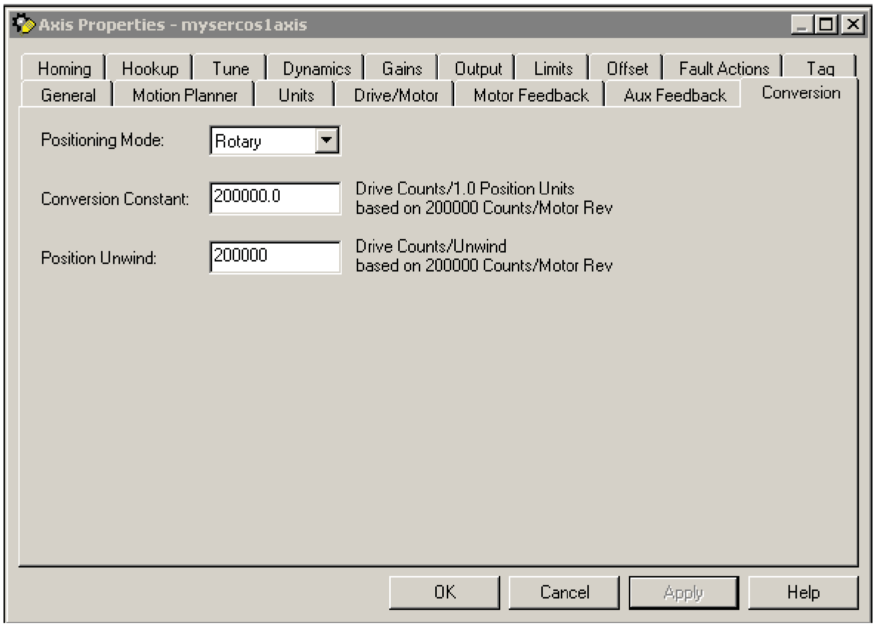
Item	Description
Feedback Type	For applications that use auxiliary feedback devices, select the type of auxiliary feedback device type. These are drive dependent.
Cycles	The number of cycles of the auxiliary feedback device. This helps the Drive Compute Conversion constant used to convert drive units to feedback counts. Depending on the feedback type selected, this value may either be read-only or editable.

Item	Description
Per	The units used to measure the cycles.
Interpolation Factor	This field displays a fixed constant value for the selected feedback type. This value is used to compute the resolution of the feedback device.
Feedback Ratio	Represents the quantitative relationship between the auxiliary feedback device and the motor. Click the Conversion tab to access the Axis Properties Conversion dialog box.  Positioning Mode: Rotary Conversion Constant: 8000.0 Feedback Counts/1.0 Position Units Position Unwind: 8000 Feedback Counts/Unwind

Conversion Tab

Use this tab to view/edit the Positioning Mode, Conversion Constant, and if configured as Rotary, the Unwind values for an axis, of the tag types AXIS_SERVO, AXIS_SERVO_DRIVE and AXIS_VIRTUAL.

The differences in the appearance of the Conversion tab screens for the AXIS_SERVO and AXIS_SERVO_DRIVE are the default values for Conversion Constant and Position Unwind and the labels for these values.



Positioning Mode: Rotary

Conversion Constant: 200000.0 Drive Counts/1.0 Position Units based on 200000 Counts/Motor Rev

Position Unwind: 200000 Drive Counts/Unwind based on 200000 Counts/Motor Rev

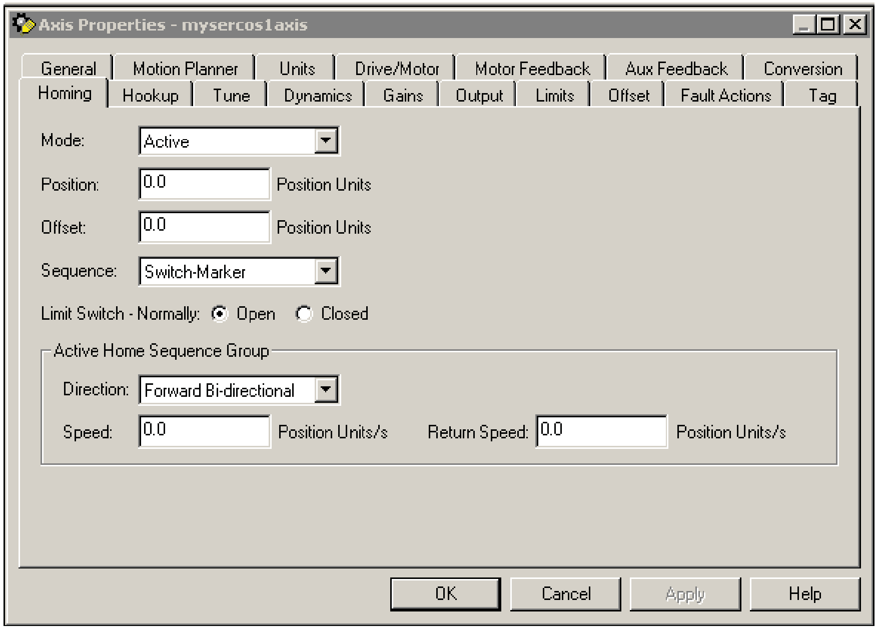
Item	Description	
Positioning Mode	This parameter is not editable for an axis of the data type AXIS_CONSUMED. Instead, this value is taken from a producing axis in a networked Logix processor. This value can be edited for AXIS_SERVO, AXIS_SERVO_DRIVE and AXIS_VIRTUAL.	
	Linear	Provides a maximum total linear travel of 2.14 (2^{31}) billion feedback counts. With this mode, the unwind feature is disabled and you can limit the linear travel distance traveled by the axis by specifying the positive and negative travel limits for the axis.

<table>
<tr><th>Item</th><th colspan="2">Description</th></tr>
<tr><td></td><td>Rotary</td><td>Enables the rotary unwind capability of the axis. This feature provides infinite position range by unwinding the axis position whenever the axis moves through a complete unwind distance. The number of encoder counts per unwind of the axis is specified by the Position Unwind parameter.</td></tr>
<tr><td>Conversion Constant</td><td colspan="2">Number of feedback counts per position unit. This conversion or ‘K’ constant lets the axis position display, and motion to be programmed, in the position units set in the Units tab.
The conversion constant is used to convert axis position units into feedback counts and vice versa for the AXIS_SERVO type and for the AXIS_SERVO_DRIVE, the number of counts per motor revolution, as set in the Drive Resolution field of the Drive tab.When you edit the setting for either the Conversion Constant or the Drive Resolution, and then click either OK or Apply, you must choose whether to also recalculate the settings for these dependent attributes. The following attributes are recalculated.
On the Dynamics tab:
• Maximum Velocity
• Maximum Acceleration
• Maximum Deceleration
On the Limits tab:
• Position Error Tolerance
On the Custom Drive Scaling Attributes dialog:
• Torque Data Scaling
On the Custom Limit Attributes dialog:
• Velocity Limit Bipolar
• Velocity Limit Positive
• Velocity Limit Negative
• Acceleration Limit Bipolar
• Acceleration Limit Positive
• Acceleration Limit Negative</td></tr>
</table>

Item	Description
Position Unwind	This parameter is not editable for an axis of the data type AXIS_CONSUMED. Instead, this value is taken from a producing axis in a networked Logix processor. For a Rotary axis (AXIS_SERVO): • This value represents the distance (in feedback counts) used to perform automatic electronic unwind. Electronic unwind lets infinite position range for rotary axes by subtracting the unwind distance from both the actual and command position, every time the axis travels the unwind distance. For axes of the type AXIS_SERVO_DRIVE: • When you save an edited Conversion Constant or a Drive Resolution value, a message box appears, asking you if you want the controller to automatically recalculate certain attribute settings. Refer to Conversion Constant and Drive Resolution Attributes. • The label indicates the number of counts per motor revolution, as set in the Drive Resolution field of the Drive tab.

Homing Tab - AXIS_SERVO

Use this tab to configure the attributes related to homing an axis of the type AXIS_SERVO.



Mode: Active

Position: 0.0 Position Units

Offset: 0.0 Position Units

Sequence: Switch-Marker

Limit Switch - Normally: ● Open ○ Closed

Active Home Sequence Group

Direction: Forward Bi-directional

Speed: 0.0 Position Units/s

Return Speed: 0.0 Position Units/s

Homing Tab - AXIS_SERVO Field Descriptions

Item	Description	
Mode	Select the homing mode.	
	Active	• Active mode - enables the axis at the beginning of the home process. • Active homing sequences always use the trapezoidal velocity profile. For LDT and SSI feedback selections, the only valid Home Sequences for Homing Mode are immediate or switch, as no physical marker exists for the LDT or SSI feedback devices. • For SSI, the selections for Home Sequence is based of if 'Enable Absolute Feedback' is checked.
	Passive	• The homing redefines the absolute position of the axis on the occurrence of a home switch or encoder marker event. • Passive homing is most commonly used to calibrate uncontrolled axes, although it can also be used with controlled axes to create a custom homing sequence. • Passive homing, for a given home sequence, works similar to the corresponding active homing sequence, except that no motion is commanded; the controller just waits for the switch and marker events to occur.

Homing Tab - AXIS_SERVO Field Descriptions

<table>
<tr><th>Item</th><th colspan="2">Description</th></tr>
<tr><td></td><td>Absolute</td><td>AXIS_SERVO_DRIVE, and AXIS_SERVO when associated with a 1756-HYD02 [LDT feedback] or 1756-M02AS [SSI feedback] module only.

Absolute homing only available on Axis_Servo_Drive if the position feedback devices support absolute homing. Absolute Homing is not available on a 1756-M02AS module if ‘Enable Absolute Feedback’ is not enabled.

• The absolute homing process establishes the true absolute position of the axis by applying the configured Home Position to the reported position of the absolute feedback device.
• The only valid Home Sequence for an absolute Homing Mode is immediate. In the LDT and SSI cases, the absolute homing process establishes the true absolute position of the axis by applying the configured Home Position less any enabled Absolute Feedback Offset to the reported position of the absolute feedback device.
• Prior to execution of the absolute homing process using the MAH instruction, the axis must be in the Axis Ready state with the servo loop disabled.</td></tr>
<tr><td></td><td colspan="2">No Physical Marker Pulse Exists

For the SSI feedback transducer no physical marker pulse exists. However, a pseudo marker reference is established by the M02AS module firmware at the feedback device’s roll over point. A single-turn Absolute SSI feedback device rolls over at its maximum ‘turns count’ = 1 rev. A multi-turn Absolute SSI feedback device (there are multiple revs or feedback-base unit-distances) the device rolls over at its maximum ‘turns count’ which is usually either 1024 or 2048.

Actual, on a it is called ‘Home to Rollover‘. And it is available in the Homing properties pages if ‘Enable Absolute Feedback‘ is not enabled.

If you need to establish the rollover of the feedback device, a ladder rung using an SSV to set Home_Sequence equal ‘Home to Rollover’. It is available in the Homing properties pages if ‘Enable Absolute Feedback’ is not enabled.

The following parameters must be added to the application program:

• Class Name = Axis,
• Attribute_Name = Home_Sequence,
• and Value = 2 (to Marker)

These can not be set in Axis Properties. They must be reset back to the initial values: 0 = Immediate or 1 = Switch, after establishing the rollover.

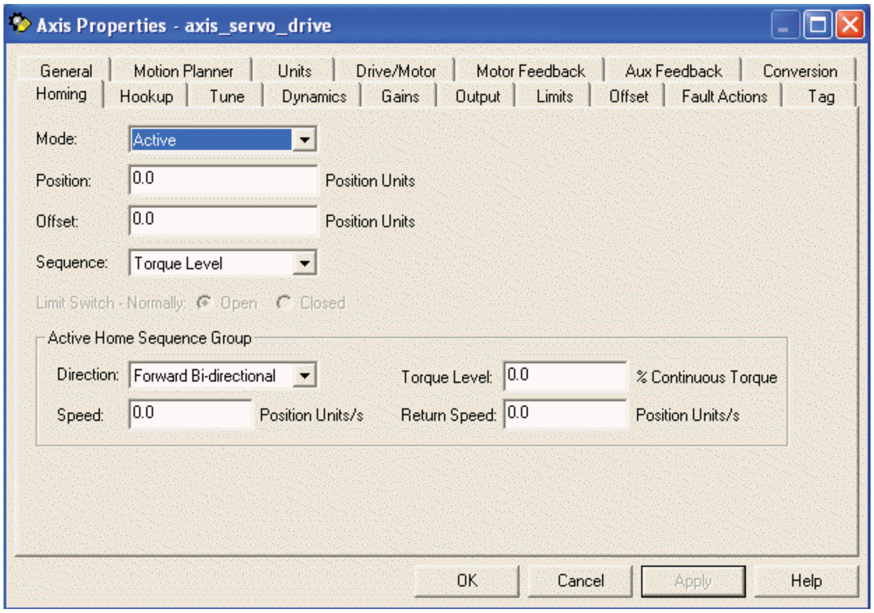
The Home Sequence = to Marker must be used to allow feedback to travel until the rollover (that is, pseudo marker) is found. This must be done without the motor attached to any axis as this could cause up to Maximum number of turn’s before pseudo marker is found.</td></tr>
</table>

Homing Tab - AXIS_SERVO Field Descriptions

<table>
<tr><th>Item</th><th colspan="2">Description</th></tr>
<tr><td>Position</td><td colspan="2">The desired absolute position, in position units, for the axis after the specified homing sequence has been completed. In most cases, this position is set to zero, although any value within the software travel limits can be used. After the homing sequence is complete, the axis is left in this position.
If the Positioning Mode (set in the Conversion tab) of the axis is Linear, then the home position should be within the travel limits, if enabled. If the Positioning Mode is Rotary, then the home position should be less than the unwind distance in position units.</td></tr>
<tr><td>Offset</td><td colspan="2">The desired offset (if any) in position units the axis is to move, upon completion of the homing sequence, to reach the home position. In most cases, this value is zero.</td></tr>
<tr><td rowspan="6">Sequence</td><td colspan="2">Select the event that causes the Home Position to be set. See the section 'Homing Configurations,' below, for a detailed description of each combination of homing mode, sequence and direction.</td></tr>
<tr><td>Sequence Type</td><td>Description</td></tr>
<tr><td>Immediate</td><td>Sets the Actual Position to the Home Position.</td></tr>
<tr><td>Switch</td><td>Sets the Actual Position to the Home Position when axis motion encounters a home limit switch.</td></tr>
<tr><td>Marker</td><td>Sets the Actual Position to the Home Position when axis encounters an encoder marker.</td></tr>
<tr><td>Switch-Marker</td><td>Sets the Actual Position to the Home position when a marker is encountered after a home switch is encountered.</td></tr>
<tr><td>Limit Switch</td><td colspan="2">If a limit switch is used, indicate the normal state of that switch (that is, before being engaged by the axis during the homing sequence):
• Normally Open
• Normally Closed</td></tr>
<tr><td rowspan="6">Direction</td><td colspan="2">For active homing sequences, except for the Immediate Sequence type, select the desired homing direction.</td></tr>
<tr><td>Direction</td><td>Description</td></tr>
<tr><td>Forward Uni-directional</td><td>The axis jogs in the positive axial direction until a homing event (switch or marker) is encountered, then continues in the same direction until axis motion stops (after decelerating or moving the Offset distance).</td></tr>
<tr><td>Forward Bi-directional</td><td>The axis jogs in the positive axial direction until a homing event (switch or marker) is encountered, then reverses direction until motion stops (after decelerating or moving the Offset distance).</td></tr>
<tr><td>Reverse Uni-directional</td><td>The axis jogs in the negative axial direction until a homing event (switch or marker) is encountered, then continues in the same direction until axis motion stops (after decelerating or moving the Offset distance).</td></tr>
<tr><td>Reverse Bi-directional</td><td>The axis jogs in the negative axial direction until a homing event (switch or marker) is encountered, then reverses direction until motion stops (after decelerating or moving the Offset distance).</td></tr>
<tr><td>Speed</td><td colspan="2">Type the speed of the jog profile used in the first leg of an active homing sequence. The homing speed specified should be less than the maximum speed and greater than zero.</td></tr>
<tr><td>Return Speed</td><td colspan="2">The speed of the jog profile used in the return leg(s) of an active homing sequence. The home return speed specified should be less than the maximum speed and greater than zero.</td></tr>
</table>

Homing Tab - AXIS_SERVO_DRIVE

Use this tab to configure the attributes related to homing an axis of the type AXIS_SERVO_DRIVE.



Axis Properties - axis_servo_drive

Mode: Active

Position: 0.0 Position Units

Offset: 0.0 Position Units

Sequence: Torque Level

Limit Switch - Normally: ● Open ○ Closed

Active Home Sequence Group

Direction: Forward Bi-directional

Torque Level: 0.0 % Continuous Torque

Speed: 0.0 Position Units/s

Return Speed: 0.0 Position Units/s

Homing Tab - AXIS_SERVO_DRIVE Field Descriptions

Item	Description	
Mode	Select the homing mode.	
	Active	• Active enables the axis at the beginning of the home process. • The desired homing sequence is selected by specifying whether a home limit switch and/or the encoder marker is used for this axis. • Active homing sequences always use the trapezoidal velocity profile. For LDT and SSI feedback selections, the only valid Home Sequences for Homing Mode are immediate or switch, as no physical marker exists for the LDT or SSI feedback devices.
	Passive	• In this mode, homing redefines the absolute position of the axis on the occurrence of a home switch or encoder marker event. • Passive homing is most commonly used to calibrate uncontrolled axes, although it can also be used with controlled axes to create a custom homing sequence. • Passive homing, for a given home sequence, works similar to the corresponding active homing sequence, except that no motion is commanded; the controller just waits for the switch and marker events to occur.

Homing Tab - AXIS_SERVO_DRIVE Field Descriptions

Item	Description	
	Passive Absolute	AXIS_SERVO_DRIVE, and AXIS_SERVO when associated with a 1756-HYD02 [LDT feedback] or 1756-M02AS [SSI feedback] module only. • In this mode, the absolute homing process establishes the true absolute position of the axis by applying the configured Home Position to the reported position of the absolute feedback device. • The only valid Home Sequence for an absolute Homing Mode is immediate. In the LDT and SSI cases, the absolute homing process establishes the true absolute position of the axis by applying the configured Home Position less any enabled Absolute Feedback Offset to the reported position of the absolute feedback device. • Prior to execution of the absolute homing process using the MAH instruction, the axis must be in the Axis Ready state with the servo loop disabled. **No Physical Marker Pulse Exists** For the SSI feedback transducer no physical marker pulse exists. However, a pseudo marker reference is established by the M02AS module firmware at the feedback device's roll over point. A single-turn Absolute SSI feedback device rolls over at its maximum 'turns count' = 1 rev. A multi-turn Absolute SSI feedback device (there are multiple revs or feedback-base unit-distances) the device rolls over at its maximum 'turns count' which is usually either 1024 or 2048. If you need to establish the rollover of the feedback device, a ladder rung using an SSV to set Home_Sequence equal 'Home to marker' with the following parameters: Class Name = SSI_Axis, Attribute_Name = Home_Sequence, and Value = 2 (to Marker) must be added to the application program (cannot be set Axis Properties and must be reset back to its initial value 0 = Immediate or 1 = Switch after establishing the rollover). The Home Sequence = to Marker must be used to allow feedback to travel until the rollover (that is, pseudo marker) is found. This must be done without the motor attached to any axis as this could cause up to Maximum number of turn's before pseudo marker is found.
	Position	Type the desired absolute position, in position units, for the axis after the specified homing sequence has been completed. In most cases, this position is set to zero, although any value within the software travel limits can be used. After the homing sequence is complete, the axis is left in this position. If the Positioning mode (set in the Conversion tab) of the axis is Linear, then the home position should be within the travel limits, if enabled. If the Positioning mode is Rotary, then the home position should be less than the unwind distance in position units.
	Offset	Type the desired offset (if any) in position units the axis is to move, upon completion of the homing sequence, to reach the home position. In most cases, this value is zero.

Homing Tab - AXIS_SERVO_DRIVE Field Descriptions

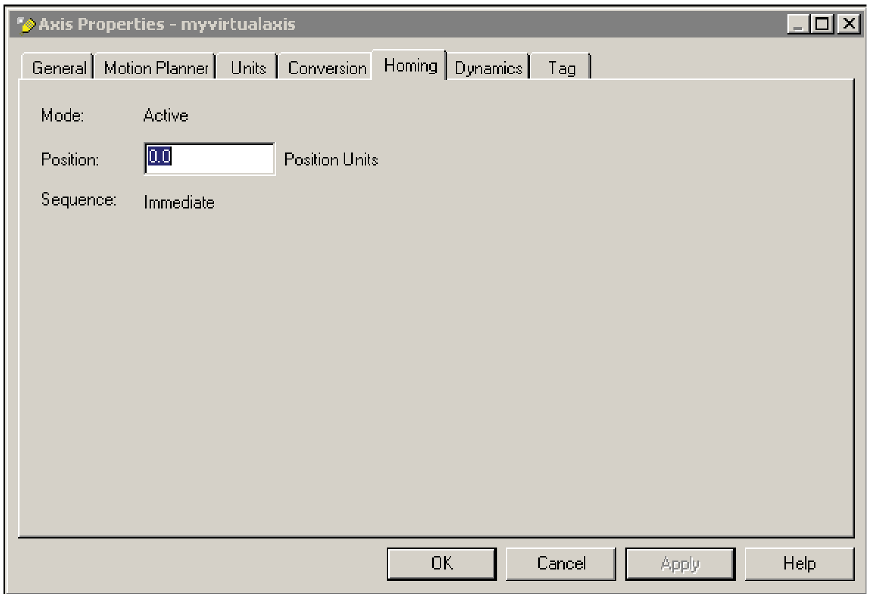
Item	Description	
Sequence	Select the event that causes the Home Position to be set.	
	Immediate	Sets the Actual Position to the Home Position.
	Switch	Sets the Actual Position to the Home Position when axis motion encounters a home limit switch.
	Marker	Sets the Actual Position to the Home Position when axis encounters an encoder marker.
	Switch-Marker	Sets the Actual Position to the Home position when a marker is encountered after a home switch is encountered.
	Torque Level	Sets the Home Position when the specified Homing Torque level is achieved on the assigned axis. IMPORTANT: For more information on the Home to Torque-level sequence, see Home to Torque-level Example Application Note, publication MOTION-AT001.
	Torque Level-marker	Sets the Home Position when the specified Homing Torque level is achieved on the assigned axis, only after the axis encounters an encoder marker. IMPORTANT: For more information on the Home to Torque-level sequence, see Home to Torque-level Example Application Note, publication MOTION-AT001. Refer to the section Homing Configurations, for a detailed description of each combination of homing mode, sequence and direction.
Limit Switch	If a limit switch is used, indicate the normal state of that switch (that is, before being engaged by the axis during the homing sequence). • Normally Open • Normally Closed	
Direction	For active homing sequences, except for the Immediate Sequence type, select the desired homing direction	
	Direction	**Description**
	Forward Uni-directional	The axis jogs in the positive axial direction until a homing event (switch or marker) is encountered, then continues in the same direction until axis motion stops (after decelerating or moving the Offset distance).
	Forward Bi-directional	The axis jogs in the positive axial direction until a homing event (switch or marker) is encountered, then reverses direction until motion stops (after decelerating or moving the Offset distance).
	Reverse Uni-directional	The axis jogs in the negative axial direction until a homing event (switch or marker) is encountered, then continues in the same direction until axis motion stops (after decelerating or moving the Offset distance).
	Reverse Bi-directional	The axis jogs in the negative axial direction until a homing event (switch or marker) is encountered, then reverses direction until motion stops (after decelerating or moving the Offset distance).

Homing Tab - AXIS_SERVO_DRIVE Field Descriptions

Item	Description
Speed	Type the speed of the jog profile used in the first leg of an active homing sequence. The homing speed specified should be less than the maximum speed and greater than zero.
Torque Level	The torque level, with units % continuous torque, that the axis motor must reach to complete the Home-to-Torque sequence. This feature is only available on the Kinetix family of drives.
Return Speed	The speed of the jog profile used in the return leg(s) of an active homing sequence. The home return speed specified should be less than the maximum speed and greater than zero.

Homing Tab - AXIS_VIRTUAL

Use this tab to configure the attributes related to homing an axis of the type AXIS_VIRTUAL.



Mode: Active

Position: 0.0 Position Units

Sequence: Immediate

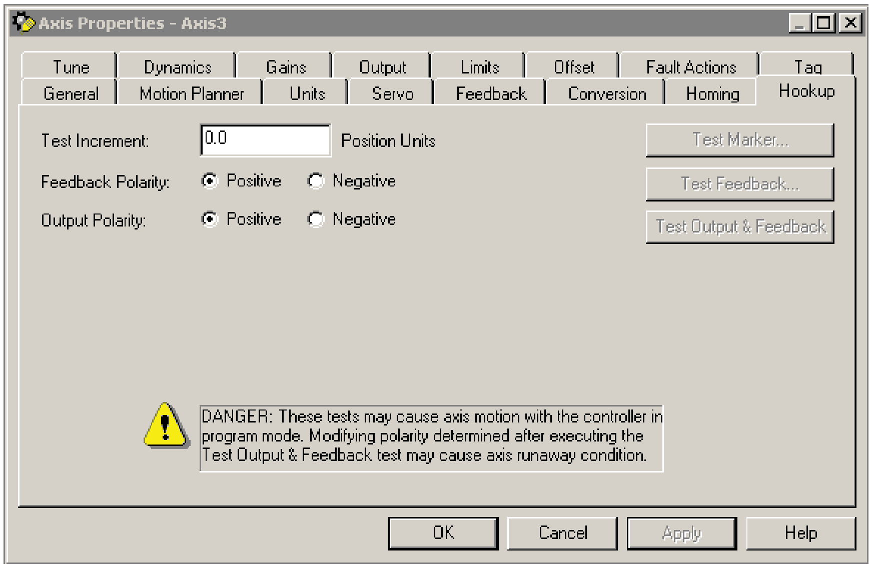
Only an Active Immediate Homing sequence can be performed for an axis of the type AXIS_VIRTUAL. A virtual axis is always enabled. The Controller assigns the Home Position to the current axis actual position and command position. This homing sequence produces no axis motion.

Item	Description
Mode	This read-only parameter is always set to Active.
Position	Desired absolute position, in position units, for the axis after the specified homing sequence has been completed. In most cases, this position is set to zero, although any value within the software travel limits can be used. After the homing sequence is complete, the axis is left at this position. If the Positioning Mode (set in the Conversion tab) of the axis is Linear, then the home position should be within the travel limits, if enabled. If the Positioning Mode is Rotary, then the home position should be less than the unwind distance in position units.
Sequence	This read-only parameter is always set to Immediate.

Hookup Tab - AXIS_SERVO

Use this tab to configure and initiate axis hookup and marker test sequences for an axis of the type AXIS_SERVO.

When a parameter transitions to a read-only state, any pending changes to parameter values are lost, and the parameter reverts to the most recently saved parameter value.



Test Increment: 0.0 Position Units

Feedback Polarity: ● Positive ○ Negative

Output Polarity: ● Positive ○ Negative

DANGER: These tests may cause axis motion with the controller in program mode. Modifying polarity determined after executing the Test Output & Feedback test may cause axis runaway condition.

Hookup Tab - AXIS_SERVO Field Description

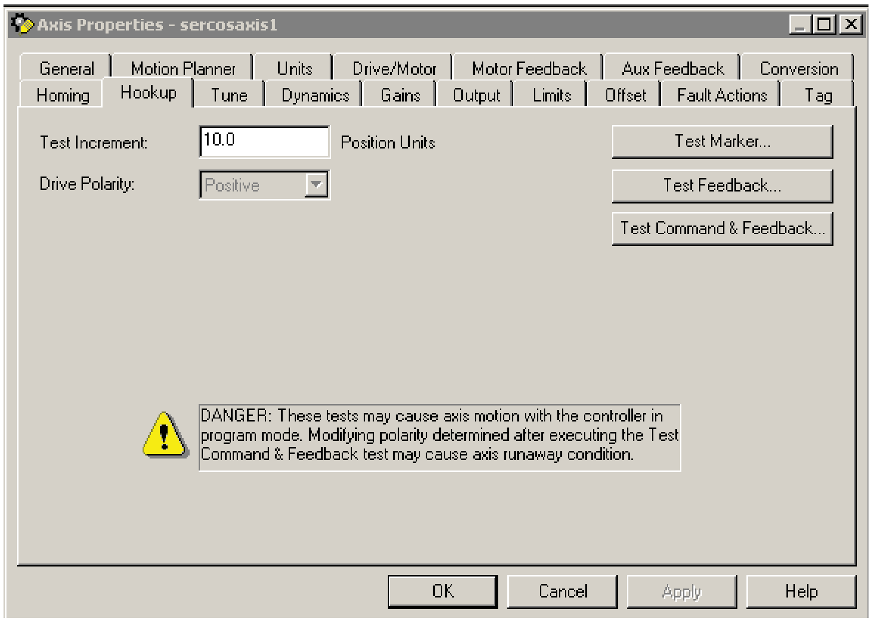
Item	Description
Test Increment	Specifies the amount of distance traversed by the axis when executing the Output & Feedback test. Test Increment is also used for the Marker and Feedback test. The test is complete when the distance is traveled. For example, if the distance is set to 1/4 of the revolution, then the marker test will fail 75% of the time because the marker will never be seen. For the Marker test, the test increment has to be a distance large enough to ensure that a marker is passed. The default value is set to approximately a quarter of a revolution of the motor in position units.
Feedback Polarity	The polarity of the encoder feedback, this field is automatically set by executing either the Feedback Test or the Output & Feedback Test. This field is set only after the test is executed and the user accepts the results. • Positive • Negative When properly configured, this setting insures that axis Actual Position value increases when the axis is moved in the user defined positive direction. This bit can be configured automatically using the MRHD and MAHD motion instructions.
	ATTENTION ⚠ Modifying input polarity values by running the Feedback or Output & Feedback Tests can cause a runaway condition resulting in unexpected motion, damage to the equipment, and physical injury or death.

Hookup Tab - AXIS_SERVO Field Description

Item	Description
Output Polarity	The polarity of the servo output to the drive, this field is automatically set by executing and accepting the results of the Output & Feedback Test. • Positive • Negative When properly configured, this setting and the Feedback Polarity setting insure that, when the axis servo loop is closed, it is closed as a negative feedback system and not an unstable positive feedback system. This bit can be configured automatically using the MRHD and MAHD motion instructions.
Test Marker	Runs the Marker test, which ensures that the encoder A, B, and Z channels are connected correctly and phased properly for marker detection. When the test is initiated, you must manually move the axis the distance specified by the Travel Limit for the system to detect the marker. If the marker is not detected, check the encoder wiring and try again.
Test Feedback	Runs the Feedback Test, which checks and, if necessary, reconfigures the Feedback Polarity setting. When the test is initiated, you must manually move the axis one revolution for the system to detect the marker. If the marker is not detected, check the encoder wiring and try again.
Test Output & Feedback	Runs the Output & Feedback Test, which checks and, if necessary, reconfigures both the polarity of encoder feedback (the Feedback Polarity setting) and the polarity of the servo output to the drive (the Output Polarity setting), for an axis configured for Servo operation in the General tab. Executing and accepting the values automatically saves all changes to Axis Properties. It is possible to execute the test but not accept (or apply) the values.

Hookup Tab Overview - AXIS_SERVO_DRIVE

Use this tab to configure and initiate axis hookup and marker test sequences for an axis of the type AXIS_SERVO_DRIVE.



Test Increment: 10.0 Position Units

Drive Polarity: Positive

DANGER: These tests may cause axis motion with the controller in program mode. Modifying polarity determined after executing the Test Command & Feedback test may cause axis runaway condition.

When a parameter transitions to a read-only state, any pending changes to parameter values are lost, and the parameter reverts to the most recently saved parameter value.

Test Increment

Specifies the amount of distance traversed by the axis when executing the Command & Feedback test. The default value is set to approximately a quarter of a revolution of the motor in position units.

Drive Polarity

The polarity of the servo loop of the drive, set by executing the Command & Feedback Test.

- Positive
- Negative

Proper wiring guarantees that the servo loop is closed with negative feedback. However there is no guarantee that the drive has the same sense of forward direction as the user for a given application. Negative Polarity inverts the polarity of both the command position and actual position data of the drive. Thus, selecting either Positive or Negative Drive Polarity makes it possible to configure the positive direction sense of the drive to agree with that of the user. This attribute can be configured automatically using the MRHD and MAHD motion instructions.

ATTENTION

Modifying polarity values, automatically input by running the Command & Feedback Test, can cause a runaway condition.

Test Marker

Runs the Marker test, which ensures that the encoder A, B, and Z channels are connected correctly and phased properly for marker detection. When the test is initiated, you must manually move the axis one revolution for the system to detect the marker. If the marker is not detected, check the encoder wiring and try again.

Test Feedback

Runs the Feedback Test, which checks and, if necessary, reconfigures the Feedback Polarity setting. When the test is initiated, you must manually move the axis one revolution for the system to detect the marker. If the marker is not detected, check the encoder wiring and try again.

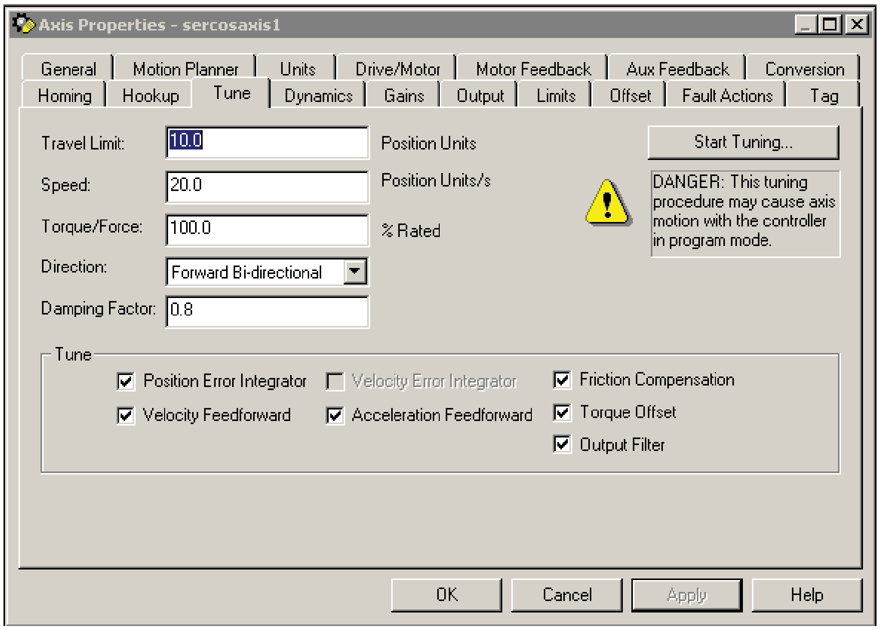
Test Command & Feedback

Runs the Command & Feedback Test, which checks and, if necessary, reconfigures both the polarity of encoder feedback (the Feedback Polarity setting) and the polarity of the servo output to the drive (the Output Polarity setting), for an axis configured for Servo operation in the General tab.

Executing any test operation automatically saves all changes to axis properties.

Tune Tab - AXIS_SERVO, AXIS_SERVO_DRIVE

Use this tab to configure and initiate the axis tuning sequence for an axis of the types AXIS_SERVO or AXIS_SERVO_DRIVE.



Travel Limit: 10.0 Position Units

Speed: 20.0 Position Units/s

Torque/Force: 100.0 % Rated

Direction: Forward Bi-directional

Damping Factor: 0.8

DANGER: This tuning procedure may cause axis motion with the controller in program mode.

Tune

☑ Position Error Integrator ☐ Velocity Error Integrator ☑ Friction Compensation

☑ Velocity Feedforward ☑ Acceleration Feedforward ☑ Torque Offset

☑ Output Filter

Travel Limit

Specifies a limit to the excursion of the axis during the tune test. If the servo module determines that the axis is not able to complete the tuning process before exceeding the tuning travel limit, it terminates the tuning profile and report that this limit was exceeded.

Speed

Determines the maximum speed for the tune process. This value should be set to the desired maximum operating speed of the motor (in engineering units) prior to running the tune test.

Torque/Force (AXIS_SERVO_DRIVE)

The maximum torque of a Rotary motor, or Force for a linear motor. Force is used only when a linear motor is connected to the application. This attribute should be set to the desired maximum safe torque level prior to running the tune test. The default value is 100%, which yields the most accurate measure of the acceleration and deceleration capabilities of the system.

In some cases a lower tuning torque limit value may be desirable to limit the stress on the mechanics during the tuning procedure. In this case the acceleration and deceleration capabilities of the system are extrapolated based on the ratio of the tuning torque to the maximum torque output of the system. Extrapolation error increases as the Tuning Torque value decreases.

Torque (AXIS_SERVO)

The maximum torque of the tune test. This attribute should be set to the desired maximum safe torque level prior to running the tune test. The default value is 100%, which yields the most accurate measure of the acceleration and deceleration capabilities of the system.

In some cases a lower tuning torque limit value may be desirable to limit the stress on the mechanics during the tuning procedure. In this case the acceleration and deceleration capabilities of the system are extrapolated based on the ratio of the tuning torque to the maximum torque output of the system. Extrapolation error increases as the Tuning Torque value decreases.

Direction

The direction of the tuning motion profile.

- Forward Uni-directional – the tuning motion profile is initiated in the forward tuning direction only.
- Forward Bi-directional – the tuning motion profile is first initiated in the forward tuning direction and then, if successful, is repeated in the reverse direction. Information returned by the Bi-directional Tuning profile can be used to tune Backlash Compensation and Torque Offset.
- Reverse Uni-directional – the tuning motion profile is initiated in the reverse tuning direction only.
- Reverse Bi-directional – the tuning motion profile is first initiated in the reverse tuning direction and then, if successful, is repeated in the forward direction. Information returned by the Bi-directional Tuning profile can be used to tune Backlash Compensation and Torque Offset.

Damping Factor

Specifies the dynamic response of the servo axis. The default is set to 0.8. When gains are tuned using a small damping factor, a step response test performed on the axis may generate uncontrolled oscillation. The gains generated using a larger damping factor would produce a system step response that has no overshoot and is stable, but may be sluggish in response to changes.

The tuning procedure uses the Damping Factor that is set in this field. However, when the controller recalculates certain attributes in response to a Motor Catalog Number change (on the Motor/Feedback tab), the controller uses the default Damping Factor value of 0.8, and not a different value set in this field.

Tune

Select the gains to be determined by the tuning test.

- Position Error Integrator – determines whether or not to calculate a value for the Position Integral Gain.
- Velocity Feedforward – determines whether or not to calculate a value for the Velocity Feedforward Gain.
- Velocity Error Integrator – determines whether or not to calculate a value for the Velocity Integral Gain.
- Acceleration Feedforward – determines whether or not to calculate a value for the Acceleration Feedforward Gain.
- Backlash Compensation – determines whether or not to calculate a value for the Backlash Compensation Gain.
- Torque Offset – determines whether or not to calculate a value for the Torque Offset. This tuning configuration is only valid if configured for bidirectional tuning.
- Output Filter – determines whether or not to calculate a value for the Output Filter Bandwidth.

Start Tuning

Once the tune process completes successfully, you are prompted to accept the values. If the tuning process completes successfully the following attributes are set. It is possible to complete the tuning process successfully and not accept (apply) the value (changes); therefore the attributes are not set.

On this tab	**These attributes are set**
Gains tab	Velocity Feedforward Gain, if checked. Acceleration Feedforward Gain, if checked. Position Proportional Gain Position Integral Gain, if checked. Velocity Proportional Gain/Velocity Integral Gain, if checked. If not checked, the values are set to zero.
Dynamics tab	Maximum Speed Maximum Acceleration Maximum Deceleration Maximum Acceleration Jerk Maximum Deceleration Jerk
Output tab	Torque Scaling Velocity Scaling (AXIS_SERVO only) Low Pass Output Filter
Limits	Position Error Tolerance

The Tune Bandwidth dialog box opens for drives, where you can tweak bandwidth values.

During tuning, if the controller detects a high degree of tuning inertia, it enables the Low Pass Output Filter and calculates and sets a value for Low Pass Output Filter Bandwidth.

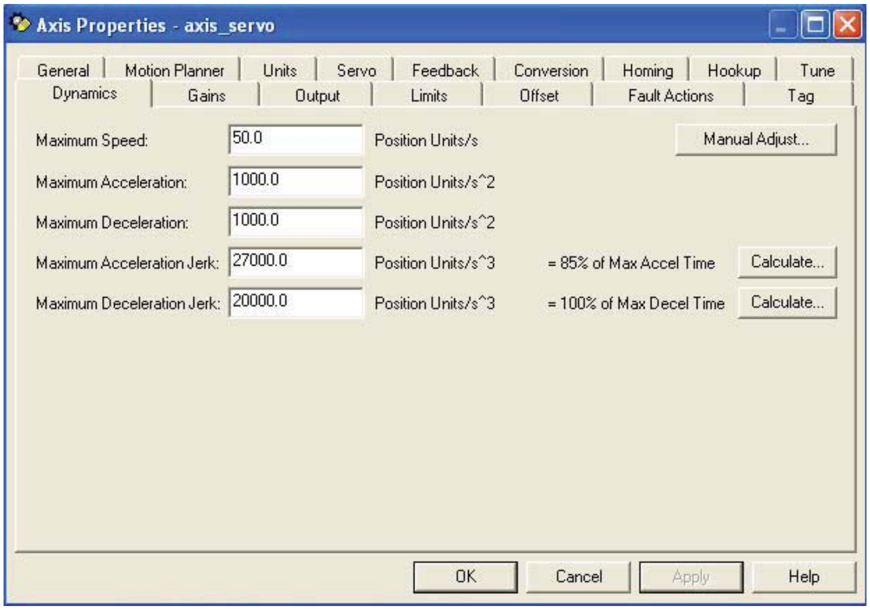
Executing a Tune operation automatically saves all changes, only if the tune values are applied, to axis properties.

ATTENTION

This tuning procedure may cause axis motion with the controller in program mode. Unexpected motion may cause damage to the equipment, personal injury, or death.

Dynamics Tab - AXIS_SERVO, AXIS_SERVO_DRIVE, AXIS_VIRTUAL

Use this tab to view or edit the dynamics related parameters for an axis of the type AXIS_SERVO or AXIS_SERVO_DRIVE configured for Servo operations in the General tab of this dialog box, or AXIS_VIRTUAL.



Maximum Speed: 50.0 Position Units/s

Maximum Acceleration: 1000.0 Position Units/s^2

Maximum Deceleration: 1000.0 Position Units/s^2

Maximum Acceleration Jerk: 27000.0 Position Units/s^3 = 85% of Max Accel Time

Maximum Deceleration Jerk: 20000.0 Position Units/s^3 = 100% of Max Decel Time

IMPORTANT The parameters on this tab can be edited in either of two ways:

- Use this tab to edit parameter changes and then click OK or Apply to save your edits.
- Many attributes can not be changed when online and/or the axis is enabled. Use Manual Adjust to make modifications to these attributes when online and the axis is enabled. Your changes are saved the moment a spin control changes any parameter value.

The parameters on this tab become read-only and cannot be edited when the controller is online if the controller is set to Hard Run mode, or if a Feedback On condition exists.

When RSLogix 5000 software is offline, the following parameters can be edited and the program saved to disk using either the Save command or when you click Apply. You must re-download the edited program to the controller before it can be run.

Maximum Speed

The steady-state speed of the axis, it is initially set to Tuning Speed by the tuning process. This value is typically set to about 90% of the maximum speed rating of the motor. This provides sufficient 'head-room' for the axis to operate at all times within the speed limitations of the motor. The Maximum Speed value entered is used when the motion instruction is set with Speed Units = % of Maximum. If a motion instruction has a Speed Units = units per sec value entered then, the speed is taken from the motion instruction faceplate.

Maximum Acceleration

The maximum acceleration rate of the axis, in Position Units/second, it is initially set to about 85% of the measured tuning acceleration rate by the tuning process. If set manually, this value should typically be set to about 85% of the maximum acceleration rate of the axis. This provides sufficient 'head-room' for the axis to operate at all times within the acceleration limits of the drive and motor. The Maximum Acceleration value entered is used when the motion instruction is set with Accel Units = % of Maximum. When a motion instruction is configured with Accel Units = units per sec^2 field then, the Maximum Acceleration is taken from the motion instruction faceplate.

Maximum Deceleration

The maximum deceleration rate of the axis, in Position Units/second, it is initially set to approximately 85% of the measured tuning deceleration rate by the tuning process. If set manually, this value should typically be set to about 85% of the maximum deceleration rate of the axis. This provides sufficient 'head-room' for the axis to operate at all times within the deceleration limits of the drive and motor. The Maximum Deceleration value entered is used when the motion instruction is set with decel Units=% of Maximum. When a motion instruction is configured with Decel Units=units per sec^2 field then, the Maximum Deceleration is taken from the motion instruction faceplate.

Maximum Acceleration Jerk

The jerk parameters only apply to S-curve profile moves using these instructions:

- MAJ
- MAM
- MAS
- MCD

The Maximum Acceleration Jerk rate of the axis, in Position Units/second^3, defaults to 100% of the maximum acceleration time after tuning. The speed and acceleration rate for this calculation are determined during S-curve the tuning process.

$$\frac{\text{MaxAccel}^2}{\text{Speed}} = \text{Maximum Acceleration Jerk}$$

The Maximum Accel Jerk value entered is used when the motion instruction is set with Jerk Units=% of Maximum. When a Single-axis Motion Instruction has Jerk Units=units per sec^3 then the maximum acceleration jerk value is derived from the motion instruction faceplate. The jerk units for the motion instruction also allow for Jerk Units=% of Time, with 100% of Time. This means that the entire S-curve move will have Jerk limiting. This is the default mode. An S-curve move with 0% of Time will result in a trapezoidal profile, and have 0% Jerk limiting. If set manually, enter the value in units=Position Units/second^3 units. You can also use the Calculate to view this value in terms of units=% of Time.

Maximum Deceleration Jerk

The jerk parameters only apply to S-curve profile moves using these instructions:

- MAJ
- MAM
- MAS
- MCD

The Maximum Deceleration Jerk rate of the axis, in Position Units/second^3, defaults to 100% of the maximum deceleration time after tuning. The speed and deceleration rate for the calculation are determined during the tuning process.

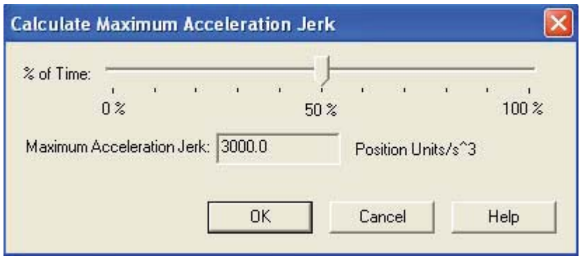
$$\frac{\text{MaxDecel}^2}{\text{Speed}} = \text{Maximum Deceleration Jerk}$$

The Maximum Decel Jerk value entered is used when the motion instruction is set with Jerk Units=% of Maximum. When a Single-axis motion instruction has Jerk Units=units per sec^3 then the Max Deceleration Jerk value is derived from the Motion Instruction faceplate. The jerk units for the motion instruction also allow for Jerk Units=% of Time, with 100% of Time meaning the entire S-curve move will have Jerk limiting, which is the default mode. An S-curve move with 0% of Time will result in a trapezoidal profile, and have 0% Jerk limiting. If set manually, enter the value in units=Position Units/second^3 units. You can also use the optional Calculate to view the value in terms of units=% of Time.

Calculate Button

This dialog box lets you set and view the Maximum Acceleration or Deceleration Jerk in Jerk Units=% of Time. Use the slider to select the value unit=% of Time. The numeric value in the Maximum Accel\Decel Jerk status box updates as the slider is moved. click OK to accept the new value, or click Cancel to leave without changing the value.

The Unit=% of Time is allowed for Jerk limiting only via the Instruction Faceplate. Only the Profile=S-curve allows Jerk control (Programmable S-curve). The units for programming Jerk limiting are more easily expressed in terms of % of Time rather than Position Units/s^3.


Calculate Maximum Acceleration Jerk

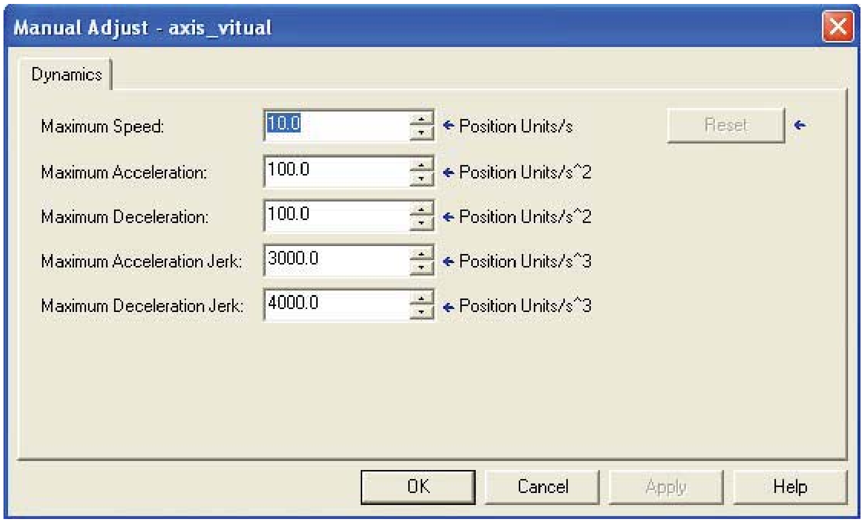
% of Time: 0 % … 50 % … 100 %

Maximum Acceleration Jerk: 3000.0 Position Units/s^3

Manual Adjust

Use Manual Adjust for online editing of the Maximum Speed, Maximum Acceleration, Maximum Deceleration, Maximum Acceleration Jerk, and Maximum Deceleration Jerk.

When values are changed on this dialog box, either manually using the spin control or entering numeric values, a blue arrow appears. This means that the values have instantaneously sent to the controller.


Manual Adjust - axis_vitual

Dynamics

Maximum Speed: 10.0 Position Units/s

Maximum Acceleration: 100.0 Position Units/s^2

Maximum Deceleration: 100.0 Position Units/s^2

Maximum Acceleration Jerk: 3000.0 Position Units/s^3

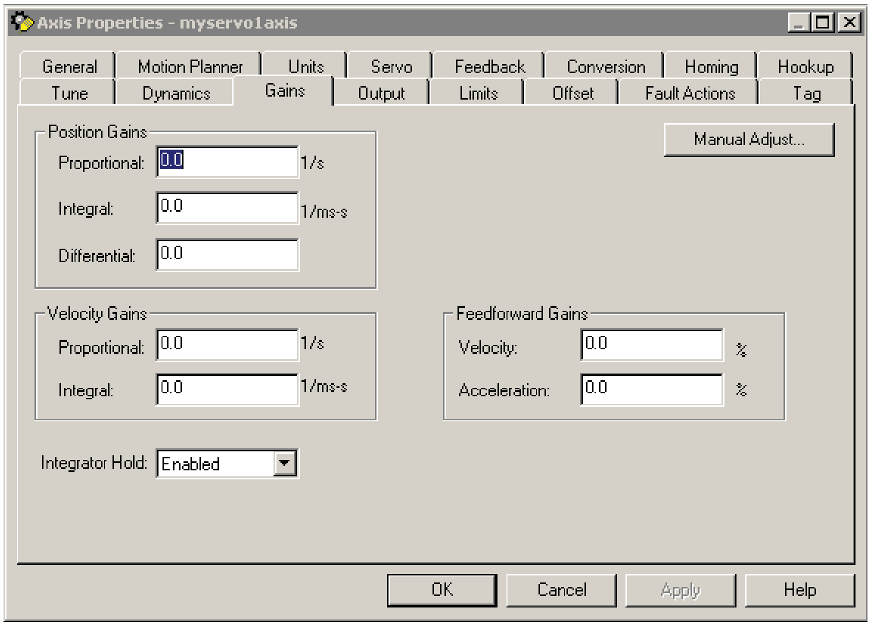
Maximum Deceleration Jerk: 4000.0 Position Units/s^3

Manual Adjust is unavailable when RSLogix 5000 software is in Wizard mode, and when offline edits to the above parameters have not yet been saved or applied.

Gains Tab - AXIS_SERVO

Use this tab to perform these offline functions for an axis of the type AXIS_SERVO, which has been configured for Servo operations (set in the General tab of this dialog box), with Position Loop Configuration.

- Adjust gain values that have been automatically set by the tuning process.
- Manually configure gains for the velocity and position loops.



Position Gains
Proportional: 0.0 1/s
Integral: 0.0 1/ms-s
Differential: 0.0

Velocity Gains
Proportional: 0.0 1/s
Integral: 0.0 1/ms-s

Feedforward Gains
Velocity: 0.0 %
Acceleration: 0.0 %

Integrator Hold: Enabled

The drive module uses a nested digital servo control loop consisting of a position loop with proportional, integral and feed-forward gains around an optional digitally synthesized inner velocity loop.

Use Manual Adjust to edit parameter settings. Values with a blue arrow have been sent to the controller. Use Manual Adjust to modify values when online and the axis is enabled. When online and the axis is enabled, the gain boxes on this dialog box are dimmed. The parameters on this tab become read-only and cannot be edited when the controller is online if the controller is set to Run mode, or if a Feedback On condition exists.

When RSLogix 5000 software is offline, the following parameters can be edited and the program saved. You must download the edited program to the controller before it can be run.

Proportional (Position) Gain

Position Error is multiplied by the Position Loop Proportional Gain, or Pos P Gain, to produce a component to the Velocity Command that ultimately attempts to correct for the position error. Too little Pos P Gain results in excessively compliant, or mushy, axis behavior. Too large a Pos P Gain, on the other hand, can result in axis oscillation due to classical servo instability.

To set the gain manually, you must first set the appropriate output scaling factor (either the Velocity Scaling factor or Torque Scaling factor) in the Output tab of this dialog box. Your selection of External Drive Configuration type – either Torque or Velocity – in the Servo tab of this dialog box determines which scaling factor you must configure before manually setting gains.

If you know the desired loop gain in inches per minute per mil or millimeters per minute per mil, use the following formula to calculate the corresponding P gain:

Pos P Gain = 16.667 * Desired Loop Gain (IPM/mil)

If you know the desired unity gain bandwidth of the position servo in Hertz, use the following formula to calculate the corresponding P gain:

Pos P Gain = Bandwidth (Hertz) * 6.28

The typical value for the Position Proportional Gain is ~100 Sec-1.

Integral (Position) Gain

The Integral (that is, summation) of Position Error is multiplied by the Position Loop Integral Gain, or Pos I Gain, to produce a component to the Velocity Command that ultimately attempts to correct for the position error. Pos I Gain improves the steady-state positioning performance of the system. Increasing the integral gain generally increases the ultimate positioning accuracy of the system. Excessive integral gain, however, results in system instability.

In certain cases, Pos I Gain control is disabled. One such case is when the servo output to the axis' drive is saturated. Continuing integral control behavior in this case would only exacerbate the situation. When the Integrator Hold parameter is set to Enabled, the servo loop automatically disables the integrator during commanded motion.

While the Pos I Gain, if employed, is typically established by the automatic servo tuning procedure (in the Tuning tab of this dialog box), the Pos I Gain value may also be set manually. Before doing this it must be stressed that the Output Scaling factor for the axis must be established for the drive system. Once this is done, the Pos I Gain can be computed based on the current or computed value for the Pos P Gain using the following formula:

Pos I Gain = .025 * 0.001 Sec/mSec * (Pos P Gain)2

Assuming a Pos P Gain value of 100 Sec-1 this results in a Pos I Gain value of 2.5 ~0.1 mSec-1 - Sec-1.

Differential

Position Differential Gain helps predict a large overshoot before it happens and makes the appropriate attempt to correct it before the overshoot actually occurs.

Proportional (Velocity) Gain

Velocity Error is multiplied by the Velocity Proportional Gain to produce a component to the Servo Output or Torque Command that ultimately attempts to correct for the velocity error, creating a damping effect. Thus, increasing the Velocity Proportional Gain results in smoother motion, enhanced acceleration, reduced overshoot, and greater system stability. However, too much Velocity Proportional Gain leads to high frequency instability and resonance effects.

If you know the desired unity gain bandwidth of the velocity servo in Hertz, you can use the following formula to calculate the corresponding P gain.

Velocity P Gain = Bandwidth (Hertz) / 6.28

The typical value for the Velocity Proportional Gain is 250.

Integral (Velocity) Gain

This parameter is enabled for all loop types except Torque loop.

At every servo update the current Velocity Error is accumulated in a variable called the Velocity Integral Error. This value is multiplied by the Velocity Integral Gain to produce a component to the Servo Output or Torque Command that attempts to correct for the velocity error. The higher the Vel I Gain value, the faster the axis is driven to the zero Velocity Error condition. Unfortunately, I Gain control is intrinsically unstable. Too much I Gain results in axis oscillation and servo instability.

In certain cases, Vel I Gain control is disabled. One such case is when the servo output to the axis' drive is saturated. Continuing integral control behavior in this case would only exacerbate the situation. When the Integrator Hold parameter is set to Enabled, the servo loop automatically disables the integrator during commanded motion.

Due to the destabilizing nature of Integral Gain, it is recommended that Position Integral Gain and Velocity Integral Gain be considered mutually exclusive. If Integral Gain is needed for the application, use one or the other, but not both. In general, where static positioning accuracy is required, Position Integral Gain is the better choice.

The typical value for the Velocity Proportional Gain is ~15 mSec-2.

Velocity Feedforward

Velocity Feedforward Gain scales the current Command Velocity by the Velocity Feedforward Gain and adds it as an offset to the Velocity Command. Hence, the Velocity Feedforward Gain allows the following error of the servo system to be reduced to nearly zero when running at a constant speed. This is important in applications such as electronic gearing, position camming, and synchronization applications, where it is necessary that the actual axis position not significantly lag behind the commanded position at any time. The optimal value for Velocity Feedforward Gain is 100%, theoretically. In reality, however, the value may need to be tweaked to accommodate velocity loops with non-infinite loop gain and other application considerations.

Acceleration Feedforward

Acceleration Feedforward Gain scales the current Command Acceleration by the Acceleration Feedforward Gain and adds it as an offset to the Servo Output generated by the servo loop. With this done, the servo loops do not need to generate much of a contribution to the Servo Output, hence the Position and/or Velocity Error values are significantly reduced. Hence, when used in conjunction with the Velocity Feedforward Gain, the Acceleration Feedforward Gain allows the following error of the servo system during the acceleration and deceleration phases of motion to be reduced to nearly zero. This is important in applications such as electronic gearing, position camming, and synchronization applications, where it is necessary that the actual axis position not significantly lag behind the commanded position at any time. The optimal value for Acceleration Feedforward is 100%, theoretically. In reality, however, the value may need to be tweaked to accommodate velocity loops with non-infinite loop gain and other application considerations.

Integrator Hold

If the Integrator Hold parameter is:

- Checked, the servo loop temporarily stops any enabled position or velocity integrators while the command position is changing. This feature is used by point-to-point moves to minimize the integrator wind-up during motion.

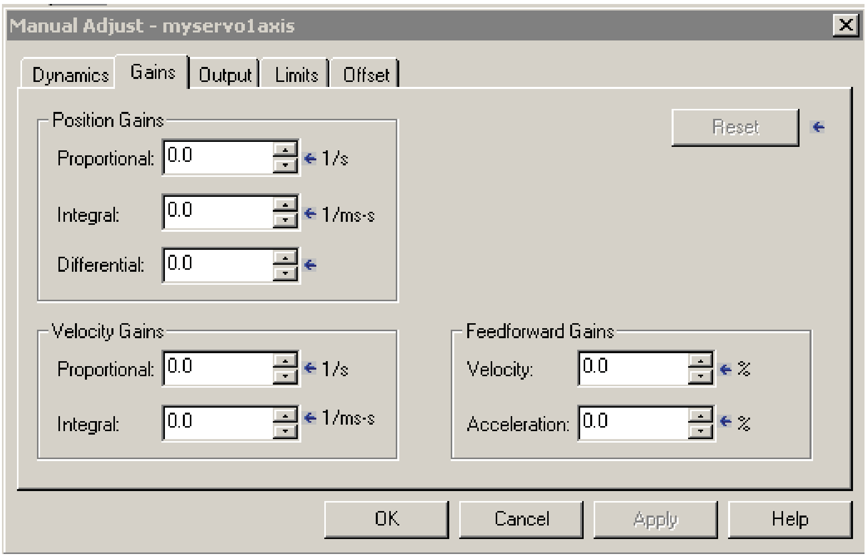
- Unchecked, all active position or velocity integrators enabled.

- Enabled, the servo loop temporarily disables any enabled position or velocity integrators while the command position is changing. This feature is used by point-to-point moves to minimize the integrator wind-up during motion.

- Disabled, all active position or velocity integrators are always enabled.

Manual Adjust

Opens the Gains tab of the Manual Adjust dialog box for online editing.



Position Gains
Proportional: 0.0 1/s
Integral: 0.0 1/ms-s
Differential: 0.0

Velocity Gains
Proportional: 0.0 1/s
Integral: 0.0 1/ms-s

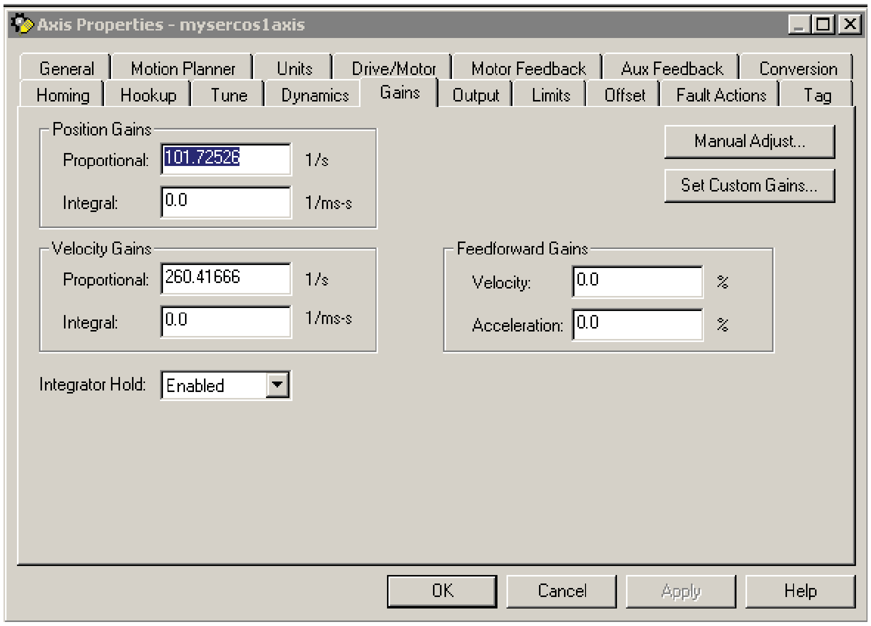
Feedforward Gains
Velocity: 0.0 %
Acceleration: 0.0 %

Manual Adjust is unavailable when RSLogix 5000 software is in Wizard mode, and when you have not yet saved or applied your offline edits to the above parameters.

Gains Tab - AXIS_SERVO_DRIVE

Use this tab to perform the following offline functions:

- Adjust, or tweak gain values that have been automatically set by the tuning process (in the Tune tab of this dialog box),
- Manually configure gains for the velocity and position loops,
- for an axis of the type AXIS_SERVO_DRIVE.



Axis Properties - mysercos1axis

Position Gains
Proportional: 101.72526 1/s
Integral: 0.0 1/ms-s

Velocity Gains
Proportional: 260.41666 1/s
Integral: 0.0 1/ms-s

Feedforward Gains
Velocity: 0.0 %
Acceleration: 0.0 %

Integrator Hold: Enabled

The drive module uses a nested digital servo control loop consisting of a position loop with proportional, integral and feed-forward gains around an optional digitally synthesized inner velocity loop. The specific design of this nested loop depends upon the Loop Configuration selected in the Drive tab. For a discussion, including a diagram, of a loop configuration, click the following loop configuration types:

- Motor Position Servo Loop
- Auxiliary Position Servo Loop
- Dual Position Servo Loop
- Motor Dual Command Servo Loop
- Auxiliary Dual Command Servo Loop
- Velocity Servo Loop
- Torque Servo Loop

The parameters on this tab can be edited in either of two ways:

- edit on this tab by typing your parameter changes and then clicking OK or Apply to save your edits

- edit in the Manual Adjust dialog box: click Manual Adjust to open the Manual Adjust dialog box to this tab and use the spin controls to edit parameter settings. Your changes are saved the moment a spin control changes any parameter value.

The parameters on this tab become read-only and cannot be edited when the controller is online if the controller is set to Hard Run mode, or if a Feedback On condition exists.

When RSLogix 5000 software is offline, the following parameters can be edited and the program saved to disk using either the Save command or by clicking Apply. You must re-download the edited program to the controller before it can be run.

Velocity Feedforward

Velocity Feedforward Gain scales the current command velocity (derivative of command position) by the Velocity Feedforward Gain and adds it as an offset to the Velocity Command. Hence, the Velocity Feedforward Gain allows the following error of the servo system to be reduced to nearly zero when running at a constant speed. This is important in applications such as electronic gearing and synchronization applications, where it is necessary that the actual axis position not significantly lag behind the commanded position at any time. The optimal value for Velocity Feedforward Gain is 100%, theoretically. In reality, however, the value may need to be tweaked to accommodate velocity loops with non-infinite loop gain and other application considerations.

Acceleration Feedforward

Acceleration Feedforward Gain scales the current Command Acceleration by the Acceleration Feedforward Gain and adds it as an offset to the Servo Output generated by the servo loop. With this done, the servo loops do not need to generate much of a contribution to the Servo Output, hence the Position and/or Velocity Error values are significantly reduced. Hence, when used in conjunction with the Velocity Feedforward Gain, the Acceleration Feedforward Gain allows the following error of the servo system during the acceleration and deceleration phases of motion to be reduced to nearly zero. This is important in applications such as electronic gearing and synchronization applications, where it is necessary that the actual axis position not significantly lag behind the commanded position at any time. The optimal value for Acceleration Feedforward is 100%, theoretically. In reality, however, the value may need to be tweaked to accommodate velocity loops with non-infinite loop gain and other application considerations.

Proportional (Position) Gain

Position Error is multiplied by the Position Loop Proportional Gain, or Pos P Gain, to produce a component to the Velocity Command that ultimately attempts to correct for the position error. Too little Pos P Gain results in excessively compliant, or mushy, axis behavior. Too large a Pos P Gain, on the other hand, can result in axis oscillation due to classical servo instability.

To set the gain manually, you must first set the Torque scaling in the Output tab of this dialog box.

If you know the desired loop gain in inches per minute per mil or millimeters per minute per mil, use the following formula to calculate the corresponding P gain:

Pos P Gain = 16.667 * Desired Loop Gain (IPM/mil)

If you know the desired unity gain bandwidth of the position servo in Hertz, use the following formula to calculate the corresponding P gain:

Pos P Gain = Bandwidth (Hertz) * 6.28

The typical value for the Position Proportional Gain is ~100 Sec-1.

Integral (Position) Gain

The Integral (that is, summation) of Position Error is multiplied by the Position Loop Integral Gain, or Pos I Gain, to produce a component to the Velocity Command that ultimately attempts to correct for the position error. Pos I Gain improves the steady-state positioning performance of the system. Increasing the integral gain generally increases the ultimate positioning accuracy of the system. Excessive integral gain, however, results in system instability.

In certain cases, Pos I Gain control is disabled. One such case is when the servo output to the axis' drive is saturated. Continuing integral control behavior in this case would only exacerbate the situation. When the Integrator Hold parameter is set to Enabled, the servo loop automatically disables the integrator during commanded motion.

While the Pos I Gain, if employed, is typically established by the automatic servo tuning procedure (in the Tuning tab of this dialog box), the Pos I Gain value may also be set manually. Before doing this it must be stressed that the Torque Scaling factor for the axis must be established for the drive system (in the Output tab of this dialog box.). Once this is done, the Pos I Gain can be computed based on the current or computed value for the Pos P Gain using the following formula:

Pos I Gain = .025 * 0.001 Sec/mSec * (Pos P Gain)2

Assuming a Pos P Gain value of 100 Sec-1 this results in a Pos I Gain value of 2.5 ~0.1 mSec-1 - Sec-1.

Proportional (Velocity) Gain

This parameter is enabled only for external drives configured for Torque loop operation in the Servo tab.

Velocity Error is multiplied by the Velocity Proportional Gain to produce a component to the Torque Command that ultimately attempts to correct for the velocity error, creating a damping effect. Thus, increasing the Velocity Proportional Gain results in smoother motion, enhanced acceleration, reduced overshoot, and greater system stability. However, too much Velocity Proportional Gain leads to high frequency instability and resonance effects.

If you know the desired unity gain bandwidth of the velocity servo in Hertz, you can use the following formula to calculate the corresponding P gain.

Vel P Gain = Bandwidth (Hertz) / 6.28

The typical value for the Velocity Proportional Gain is ~250 mSec-1.

Integral (Velocity) Gain

This parameter is enabled only for external drives configured for Torque loop operation in the Servo tab.

At every servo update the current Velocity Error is accumulated in a variable called the Velocity Integral Error. This value is multiplied by the Velocity Integral Gain to produce a component to the Torque Command that attempts to correct for the velocity error. The higher the Vel I Gain value, the faster the axis is driven to the zero Velocity Error condition. Unfortunately, I Gain control is intrinsically unstable. Too much I Gain results in axis oscillation and servo instability.

In certain cases, Vel I Gain control is disabled. One such case is when the servo output to the axis' drive is saturated. Continuing integral control behavior in this case would only exacerbate the situation. When the Integrator Hold parameter is set to Enabled, the servo loop automatically disables the integrator during commanded motion.

Due to the destabilizing nature of Integral Gain, it is recommended that Position Integral Gain and Velocity Integral Gain be considered mutually exclusive. If Integral Gain is needed for the application, use one or the other, but not both. In general, where static positioning accuracy is required, Position Integral Gain is the better choice.

While the Vel I Gain, if employed, is typically established by the automatic servo tuning procedure (in the Tune tab of this dialog box), the Pos I Gain value may also be set manually. Before doing this it must be stressed that the Torque Scaling factor for the axis must be established for the drive system, in the Output tab. Once this is done the Vel I Gain can be computed based on the current or computed value for the Vel P Gain using the following formula:

Vel I Gain = 0.25 * 0.001 Sec/mSec * (Vel P Gain)2

The typical value for the Velocity Proportional Gain is ~15 mSec-2.

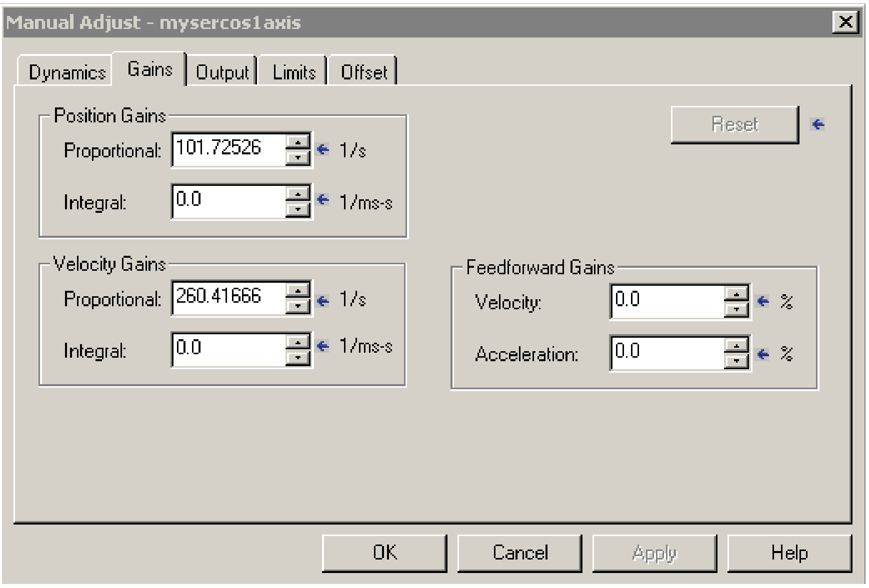
Integrator Hold

If the Integrator Hold parameter is set to:

- Enabled, the servo loop temporarily disables any enabled position or velocity integrators while the command position is changing. This feature is used by point-to-point moves to minimize the integrator wind-up during motion.
- Disabled, all active position or velocity integrators are always enabled.

Manual Adjust

Opens the Gains tab of the Manual Adjust dialog box for online editing.



Manual Adjust - mysercos1axis

Position Gains

Proportional: 101.72526 1/s

Integral: 0.0 1/ms-s

Velocity Gains

Proportional: 260.41666 1/s

Integral: 0.0 1/ms-s

Feedforward Gains

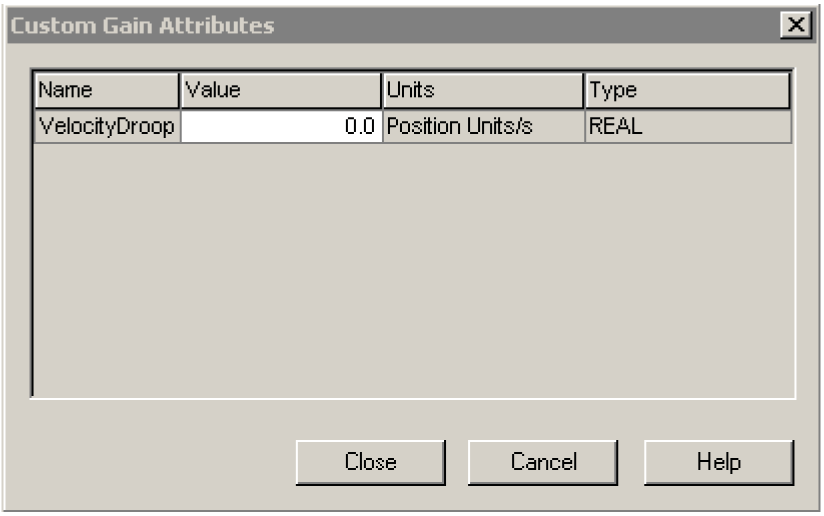
Velocity: 0.0 %

Acceleration: 0.0 %

Manual Adjust is unavailable when RSLogix 5000 software is in Wizard mode, and when you have not yet saved or applied your offline edits to the above parameters.

Set Custom Gains

Set Custom Gains dialog box is where you can edit the VelocityDroop attribute.



Name	Value	Units	Type
VelocityDroop	0.0	Position Units/s	REAL

When a parameter transitions to a read-only state, any pending changes to parameter values are lost, and the parameter reverts to the most recently saved parameter value.

Attribute

The following attribute value can be monitored and edited in this dialog box box.

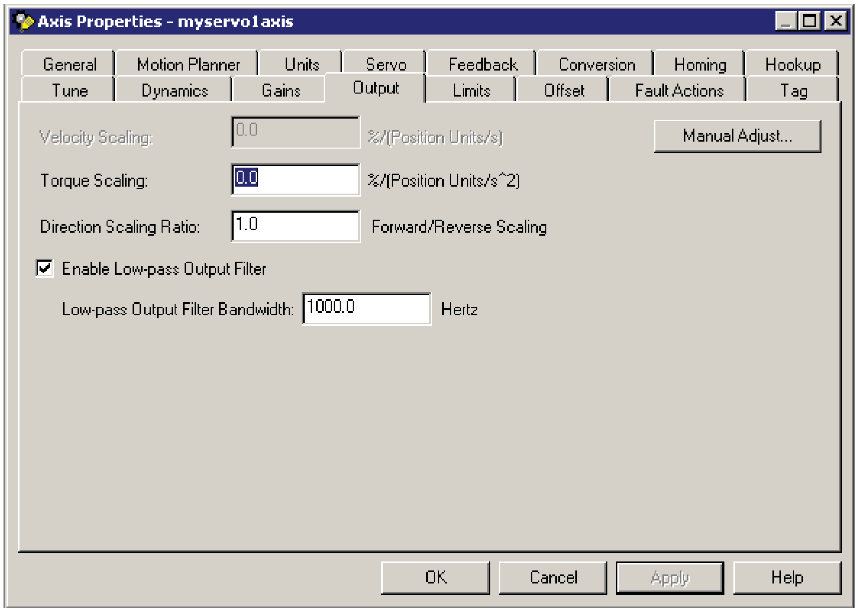
Attribute	Description
VelocityDroop	This 32-bit unsigned attribute – also referred to as static gain – acts as a very slow discharge of the velocity loop integrator. VelocityDroop may be used as a component of an external position loop system where setting this parameter to a higher, nonzero value eliminates servo hunting due to load/stick friction effects. This parameter only has effect if VelocityIntegralGain is not zero. Its value ranges from 0 to 2.14748x10^12. This value is not applicable for Ultra3000 drives.

Output Tab - AXIS_SERVO

Use this dialog box for offline configuration of:

- scaling values, which are used to generate gains, and
- the servo's low-pass digital output filter

for an axis of the type AXIS_SERVO configured as a Servo drive in the General tab of this dialog box.



Velocity Scaling: 0.0 %/(Position Units/s)

Torque Scaling: 0.0 %/(Position Units/s^2)

Direction Scaling Ratio: 1.0 Forward/Reverse Scaling

☑ Enable Low-pass Output Filter

Low-pass Output Filter Bandwidth: 1000.0 Hertz

The parameters on this tab can be edited in either of two ways:

- edit on this tab by typing your parameter changes and then click OK or Apply to save your edits

- edit in the Manual Adjust dialog box: click Manual Adjust to open the Manual Adjust dialog box to this tab and use the spin controls to edit parameter settings. Your changes are saved the moment a spin control changes any parameter value.

The parameters on this tab become read-only and cannot be edited when the controller is online if the controller is set to Hard Run mode, or if a Feedback On condition exists.

When RSLogix 5000 software is offline, the following parameters can be edited and the program saved to disk using either the Save command or by clicking Apply. You must re-download the edited program to the controller before it can be run.

Velocity Scaling

The Velocity Scaling attribute is used to convert the output of the servo loop into equivalent voltage to an external velocity drive. This has the effect of 'normalizing' the units of the servo loop gain parameters so that their values are not affected by variations in feedback resolution, drive scaling, or mechanical gear ratios. The Velocity Scaling value is typically established by servo's automatic tuning procedure but these values can be calculated, if necessary, using the following guidelines.

If the axis is configured for a velocity external drive (in the Servo tab of this dialog box), the software velocity loop in the servo module is disabled. In this case the Velocity Scaling value can be calculated by the following formula:

Velocity Scaling = 100% / (Speed @ 100%)

For example, if this axis is using position units of motor revolutions (revs), and the drive is scaled such that with an input of 100% (for example, 10 Volts) the motor goes 5,000 RPM (or 83.3 RPS), the Velocity Scaling attribute value would be calculated as:

Velocity Scaling = 100% / (83.3 RPS) = 1.2% / Revs Per Second

Torque/Force Scaling

The Torque Scaling attribute is used to convert the acceleration of the servo loop into equivalent % rated torque to the motor. This has the effect of 'normalizing' the units of the servo loops gain parameters so that their values are not affected by variations in feedback resolution, drive scaling, motor and load inertia, and mechanical gear ratios. The Torque Scaling value is typically established by the controller's automatic tuning procedure but the value can be manually calculated, if necessary, using the following guidelines:

Torque Scaling = 100% Rated Torque / (Acceleration @ 100% Rated Torque)

For example, if this axis is using position units of motor revolutions (revs), with 100% rated torque applied to the motor, if the motor accelerates at a rate of 3000 Revs/Sec2, the Torque Scaling attribute value would be calculated as shown below:

Torque Scaling = 100% Rated / (3000 RPS2) = 0.0333% Rated/ Revs Per Second2

If the Torque Scaling value does not reflect the true torque to acceleration characteristic of the system, the gains also does not reflect the true performance of the system.

Enable Low-pass Output Filter

Select this to enable the servo's low-pass digital output filter. De-select this to disable this filter.

During tuning, if the controller detects a high degree of tuning inertia, it enables the Low Pass Output Filter and calculates and sets a value for Low Pass Output Filter Bandwidth.

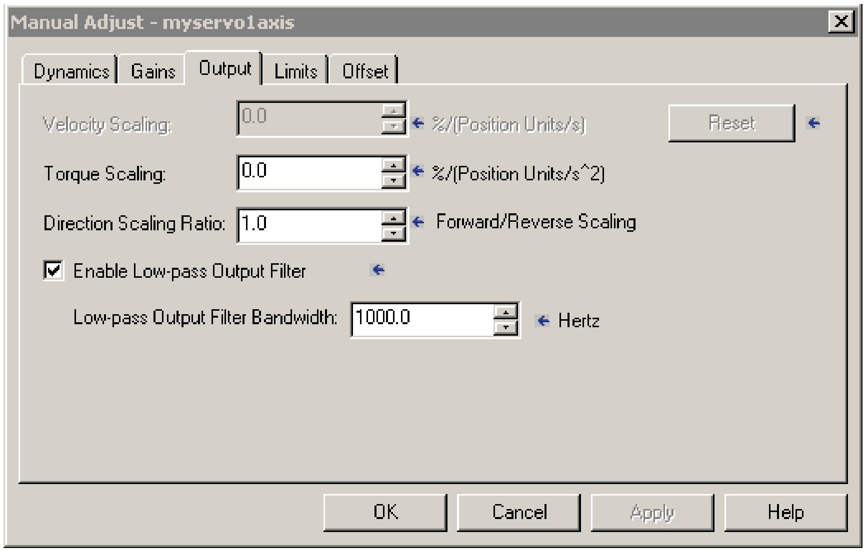
Low-pass Output Filter Bandwidth

With Enable Low-pass Output Filter selected, this value sets the bandwidth, in Hertz, of the servo's low-pass digital output filter. Use this output filter to filter out high frequency variation of the servo module output to the drive. All output from the servo module greater than the Filter Bandwidth setting is filtered-out, and not sent to the drive.

If the Low-pass Output Filter Bandwidth value is set to zero, the low-pass output filter is disabled. The lower the Filter Bandwidth value, the greater the attenuation of these high frequency components of the output signal. Because the low-pass filter adds lag to the servo loop, which pushes the system towards instability, decreasing the Filter Bandwidth value usually requires lowering the Position or Velocity Proportional Gain settings to maintain stability. The output filter is particularly useful in high inertia applications where resonance behavior can severely restrict the maximum bandwidth capability of the servo loop.

Manual Adjust

Opens the Output tab of the Manual Adjust dialog box for online editing.



Manual Adjust - myservo1axis

Velocity Scaling: 0.0 %/(Position Units/s)

Torque Scaling: 0.0 %/(Position Units/s^2)

Direction Scaling Ratio: 1.0 Forward/Reverse Scaling

☑ Enable Low-pass Output Filter

Low-pass Output Filter Bandwidth: 1000.0 Hertz

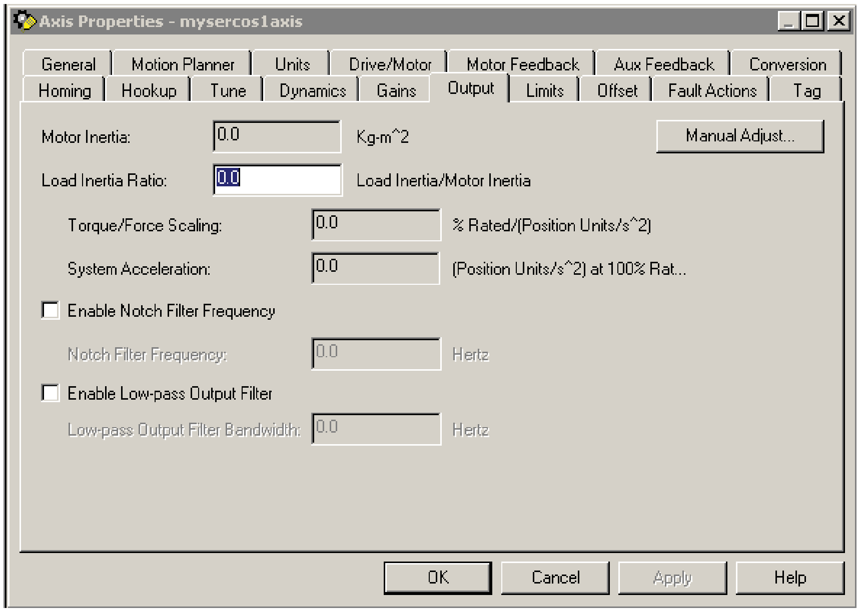
Manual Adjust is unavailable when RSLogix 5000 software is in Wizard mode, and when you have not yet saved or applied your offline edits to the above parameters.

Output Tab Overview - AXIS_SERVO_DRIVE

Use this dialog box to make the following offline configurations:

- set the torque scaling value, which is used to generate gains
- enable and configure the Notch Filter
- enable and configure servo's low-pass digital output filter

For an axis of the type AXIS_SERVO_DRIVE, configured as a Servo drive in the General tab of this dialog box.



Motor Inertia: 0.0 Kg-m^2

Load Inertia Ratio: 0.0 Load Inertia/Motor Inertia

Torque/Force Scaling: 0.0 % Rated/(Position Units/s^2)

System Acceleration: 0.0 (Position Units/s^2) at 100% Rat...

☐ Enable Notch Filter Frequency

Notch Filter Frequency: 0.0 Hertz

☐ Enable Low-pass Output Filter

Low-pass Output Filter Bandwidth: 0.0 Hertz

The parameters on this tab can be edited in either of two ways:

- edit on this tab by typing your parameter changes and then click OK or Apply to save your edits
- edit in the Manual Adjust dialog box: click Manual Adjust to open the Manual Adjust dialog box to this tab and use the spin controls to edit parameter settings. Your changes are saved the moment a spin control changes any parameter value.

The parameters on this tab become read-only and cannot be edited when the controller is online if the controller is set to Hard Run mode, or if a Feedback On condition exists.

When RSLogix 5000 software is offline, the following parameters can be edited and the program saved to disk using either the Save command or by clicking Apply. You must re-download the edited program to the controller before it can be run.

Motor Inertia

The Motor Inertia value represents the inertia of the motor without any load attached to the motor shaft in Torque Scaling units.

Load Inertia Ratio

The Load Inertia Ratio value represents the ratio of the load inertia to the motor inertia.

Torque Scaling

The Torque Scaling attribute is used to convert the acceleration of the servo loop into equivalent % rated torque to the motor. This has the effect of normalizing the units of the servo loops gain parameters so that their values are not affected by variations in feedback resolution, drive scaling, motor and load inertia, and mechanical gear ratios. The Torque Scaling value is typically established by the controller's automatic tuning procedure but the value can be manually calculated, if necessary, using the following guidelines:

Torque Scaling = 100% Rated Torque / (Acceleration @ 100% Rated Torque)

For example, if this axis is using position units of motor revolutions (revs), with 100% rated torque applied to the motor, if the motor accelerates at a rate of 3000 Revs/Sec2, the Torque Scaling attribute value would be calculated as shown below:

Torque Scaling = 100% Rated / (3000 RPS2) = 0.0333% Rated/ Revs Per Second2

If the Torque Scaling value does not reflect the true torque to acceleration characteristic of the system, the gains also do not reflect the true performance of the system.

Enable Notch Filter

Select this to enable the drive's notch filter. De-select this to disable this filter.

Notch Filter

With Enable Notch Filter selected, this value sets the center frequency of the drive's digital notch filter. If the Notch Filter value is set to zero, the notch filter is disabled.

Currently implemented as a 2nd order digital filter with a fixed Q, the Notch Filter provides approximately 40DB of output attenuation at the Notch Filter frequency. This output notch filter is particularly useful in attenuating mechanical resonance phenomena. The output filter is particularly useful in high inertia applications where mechanical resonance behavior can severely restrict the maximum bandwidth capability of the servo loop.

This value is not applicable for Ultra3000 drives.

Enable Low-pass Output Filter

Select this to enable the servo’s low-pass digital output filter. De-select this to disable this filter.

During tuning, if the controller detects a high degree of tuning inertia, the controller enables the Low Pass Output Filter and calculates and sets a value for Low Pass Output Filter Bandwidth.

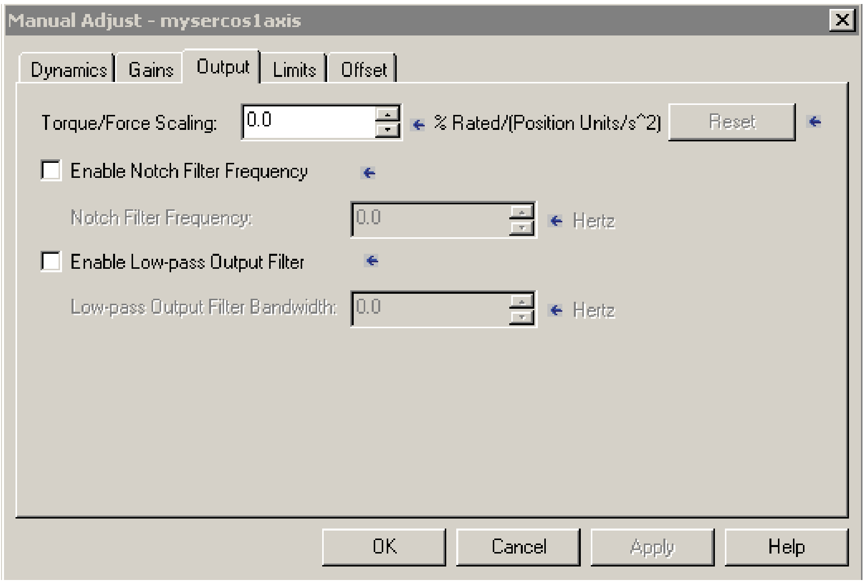
Low-pass Output Filter Bandwidth

With Enable Low-pass Output Filter selected, this value sets the bandwidth, in Hertz, of the servo’s low-pass digital output filter. Use this output filter to filter out high frequency variation of the servo module output to the drive. All output from the servo module greater than the Filter Bandwidth setting is filtered-out, and not sent to the drive.

If the Low-pass Output Filter Bandwidth value is set to zero, the low-pass output filter is disabled. The lower the Filter Bandwidth value, the greater the attenuation of these high frequency components of the output signal. Because the low-pass filter adds lag to the servo loop, which pushes the system towards instability, decreasing the Filter Bandwidth value usually requires lowering the Position or Velocity Proportional Gain settings to maintain stability. The output filter is particularly useful in high inertia applications where resonance behavior can severely restrict the maximum bandwidth capability of the servo loop.

Manual Adjust

Opens the Output tab of the Manual Adjust dialog box for online editing of Torque/Force Scaling, the Notch Filter Frequency, and the Low-pass Output Filter parameters.



Torque/Force Scaling: 0.0 % Rated/(Position Units/s^2)

☐ Enable Notch Filter Frequency

Notch Filter Frequency: 0.0 Hertz

☐ Enable Low-pass Output Filter

Low-pass Output Filter Bandwidth: 0.0 Hertz

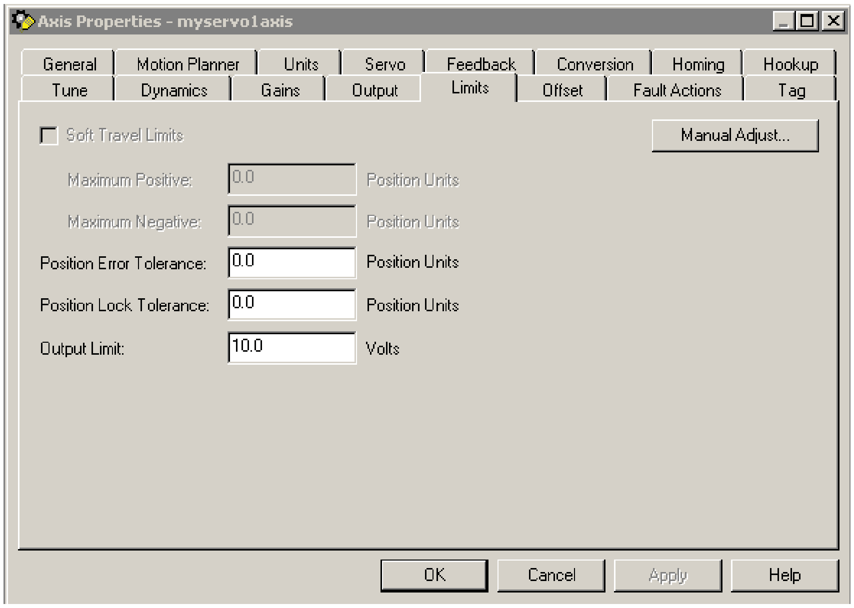
The Manual Adjust is unavailable when RSLogix 5000 software is in Wizard mode and when offline edits to the above parameters have not yet been applied.

Limits Tab - AXIS_SERVO

Use this tab to make the following offline configurations:

- enable and set maximum positive and negative software travel limits, and
- configure both Position Error Tolerance and Position Lock Tolerance, and
- set the drive's Output Limit

For an axis of the type AXIS_SERVO configured as a Servo drive in the General tab of this dialog box.



☐ Soft Travel Limits

Maximum Positive: 0.0 Position Units

Maximum Negative: 0.0 Position Units

Position Error Tolerance: 0.0 Position Units

Position Lock Tolerance: 0.0 Position Units

Output Limit: 10.0 Volts

The parameters on this tab can be edited in either of two ways:

- edit on this tab by typing your parameter changes and then click OK or Apply to save your edits

- edit in the Manual Adjust dialog box: click Manual Adjust to open the Manual Adjust dialog box to this tab and use the spin controls to edit parameter settings. Your changes are saved the moment a spin control changes any parameter value.

The parameters on this tab become read-only and cannot be edited when the controller is online if the controller is set to Run mode, or if a Feedback On condition exists.

When RSLogix 5000 software is offline, the following parameters can be edited and the program saved or applied. You must download the edited program to the controller before it can be run.

Soft Travel Limits

Enables software overtravel checking for an axis when Positioning Mode is set to Linear (in the Conversion tab of this dialog box). If an axis is configured for software overtravel limits and if that axis passes beyond these maximum travel limits (positive or negative), a software overtravel fault is issued. The response to this fault is specified by the Soft Overtravel setting (in the Fault Actions tab of this dialog box). Software overtravel limits are disabled during the tuning process.

Maximum Positive

Type the maximum positive position to be used for software overtravel checking, in position units.

The Maximum Positive limit must always be greater than the Maximum Negative limit.

Maximum Negative

Type the maximum negative position to be used for software overtravel checking, in position units.

The Maximum Negative limit must always be less than the Maximum Positive limit.

Position Error Tolerance

Specifies how much position error the servo tolerates before issuing a position error fault. This value is interpreted as a +/- quantity.

For example, setting Position Error Tolerance to 0.75 position units means that a position error fault is generated whenever the position error of the axis is greater than 0.75 or less than -0.75 position units, as shown here:

This value is set to twice the following error at maximum speed based on the measured response of the axis, during the autotuning process. In most applications, this value provides reasonable protection in case of an axis fault or stall condition without nuisance faults during normal operation. If you need to change the calculated position error tolerance value, the recommended setting is 150% to 200% of the position error while the axis is running at its maximum speed.

Position Lock Tolerance

Specifies the maximum position error the servo module accepts in order to indicate the Position Lock status bit is set. This is useful in determining when the desired end position is reached for position moves. This value is interpreted as a +/- quantity.

For example, specifying a lock tolerance of 0.01 provides a minimum positioning accuracy of +/- 0.01 position units, as shown here:

Output Limit

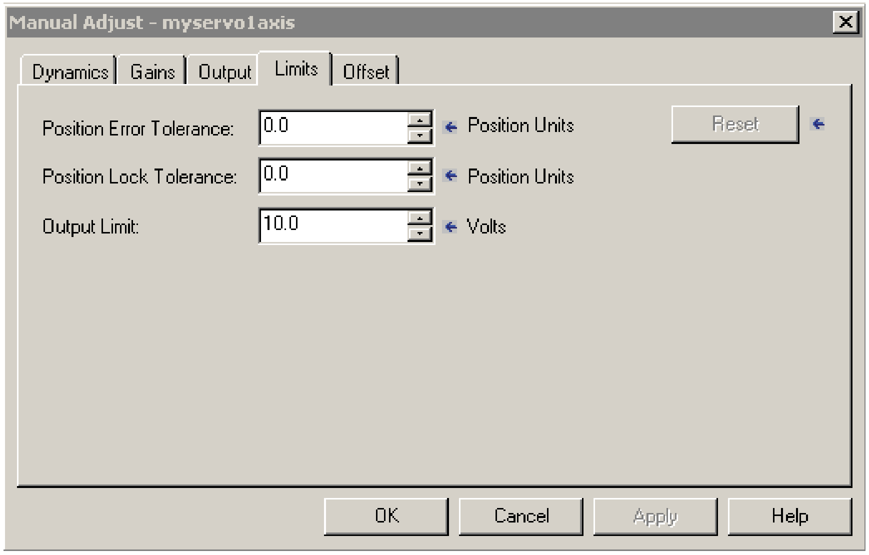
Provides a method of limiting the maximum servo output voltage of a physical axis to a specified level. The servo output for the axis as a function of position servo error, both with and without servo output limiting, is shown below.

The servo output limit may be used as a software current or torque limit if you are using a drive in torque loop mode. The percentage of the drive's maximum current that the servo controller ever commands is equal to the specified servo output limit. For example, if the drive is capable of 30 Amps of current for a 10 Volt input, setting the servo output limit to 5V limits the maximum drive current to 15 Amps.

The servo output limit may also be used if the drive cannot accept the full ±10 Volt range of the servo output. In this case, the servo output limit value effectively limits the maximum command sent to the amplifier. For example, if the drive can only accept command signals up to ±7.5 Volts, set the servo output limit value to 7.5 volts.

Manual Adjust

Opens the Limits tab of the Manual Adjust dialog box for online editing of the Position Error Tolerance, Position Lock Tolerance, and Output Limit parameters.



Manual Adjust - myservo1axis

Position Error Tolerance: 0.0 Position Units

Position Lock Tolerance: 0.0 Position Units

Output Limit: 10.0 Volts

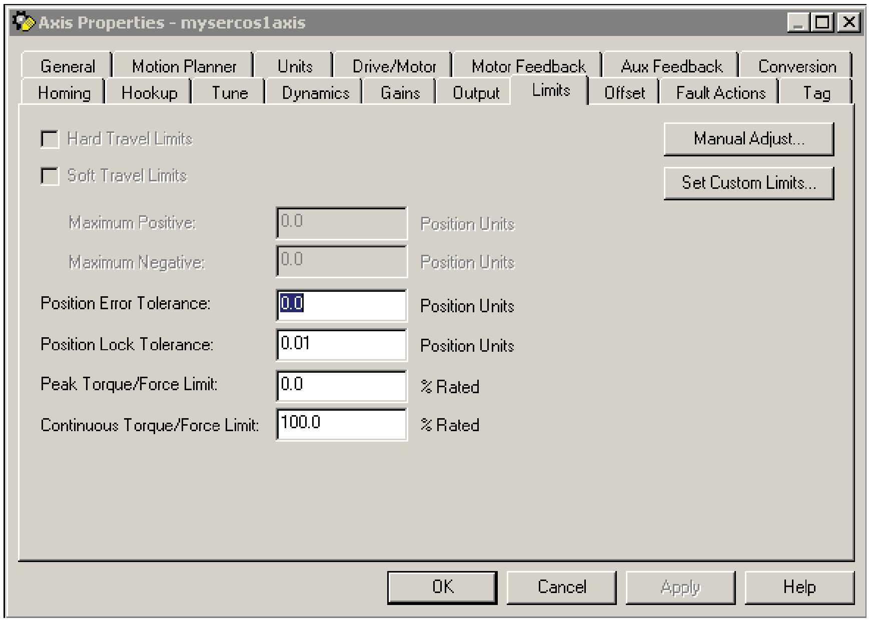
Manual Adjust is disabled when RSLogix 5000 software is in Wizard mode, and when offline edits to the above parameters have not yet been saved or applied.

Limits Tab - AXIS_SERVO_DRIVE

Use this tab to make the following offline configurations:

- enable and set maximum positive and negative software travel limits, and
- configure both Position Error Tolerance and Position Lock Tolerance,

for an axis of the type AXIS_SERVO_DRIVE configured as a Servo drive in the General tab of this dialog box.



☐ Hard Travel Limits

☐ Soft Travel Limits

Maximum Positive: 0.0 Position Units

Maximum Negative: 0.0 Position Units

Position Error Tolerance: 0.0 Position Units

Position Lock Tolerance: 0.01 Position Units

Peak Torque/Force Limit: 0.0 % Rated

Continuous Torque/Force Limit: 100.0 % Rated

The parameters on this tab can be edited in either of two ways:

- edit on this tab by typing your parameter changes and then click OK or Apply to save your edits

- edit in the Manual Adjust dialog box: click Manual Adjust to open the Manual Adjust dialog box to this tab and use the spin controls to edit parameter settings. Your changes are saved the moment a spin control changes any parameter value.

The parameters on this tab become read-only and cannot be edited when the controller is online if the controller is set to Hard Run mode, or if a Feedback On condition exists.

When RSLogix 5000 software is offline, the following parameters can be edited and the program saved to disk using either the Save command or by clicking Apply. You must re-download the edited program to the controller before it can be run.

Hard Travel Limits

Enables a periodic test that monitors the current state of the positive and negative overtravel limit switch inputs, when Positioning Mode is set to Linear (in the Conversion tab of this dialog box). If an axis is configured for hardware overtravel checking and if that axis passes beyond a positive or negative overtravel limit switch, a Positive Hard Overtravel Fault or Negative Hard Overtravel Fault is issued. The response to this fault is specified by the Hard Overtravel setting (in the Fault Actions tab of this dialog box).

Soft Travel Limits

Enables software overtravel checking for an axis when Positioning Mode is set to Linear (in the Conversion tab of this dialog box). If an axis is configured for software overtravel limits and if that axis passes beyond these maximum travel limits (positive or negative), a software overtravel fault is issued. The response to this fault is specified by the Soft Overtravel setting (in the Fault Actions tab of this dialog box). Software overtravel limits are disabled during the tuning process.

Maximum Positive

Type the maximum positive position to be used for software overtravel checking, in position units.

The Maximum Positive limit must always be greater than the Maximum Negative limit.

Maximum Negative

Type the maximum negative position to be used for software overtravel checking, in position units.

The Maximum Negative limit must always be less than the Maximum Positive limit.

Position Error Tolerance

Specifies how much position error the servo tolerates before issuing a position error fault. This value is interpreted as a +/- quantity.

For example, setting Position Error Tolerance to 0.75 position units means that a position error fault is generated whenever the position error of the axis is greater than 0.75 or less than -0.75 position units, as shown here:

This value is set to twice the following error at maximum speed based on the measured response of the axis, during the autotuning process. In most applications, this value provides reasonable protection in case of an axis fault or stall condition without nuisance faults during normal operation. If you need to change the calculated position error tolerance value, the recommended setting is 150% to 200% of the position error while the axis is running at its maximum speed.

Position Lock Tolerance

Specifies the maximum position error the servo module accepts in order to indicate the Position Lock status bit is set. This is useful in determining when the desired end position is reached for position moves. This value is interpreted as a +/- quantity.

For example, specifying a lock tolerance of 0.01 provides a minimum positioning accuracy of +/- 0.01 position units, as shown here:

Peak Torque/Force Limit

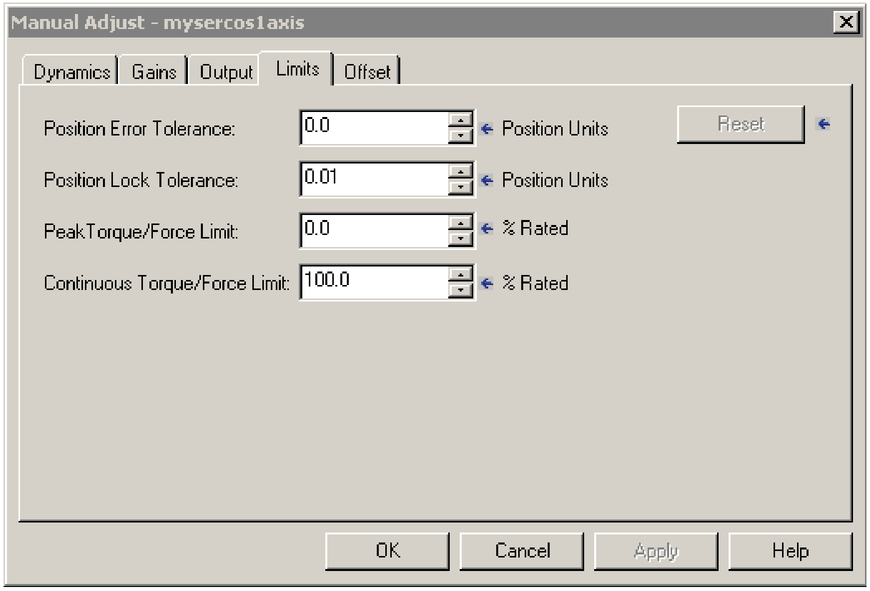
The Peak Torque/Force Limit specifies the maximum percentage of the motors rated current that the drive can command as either positive or negative torque/force. For example, a torque limit of 150% shall limit the current delivered to the motor to 1.5 times the continuous current rating of the motor.

Continuous Torque/Force Limit

The Continuous Torque/Force Limit specifies the maximum percentage of the motors rated current that the drive can command on a continuous or RMS basis. For example, a Continuous Torque/Force Limit of 150% limits the continuous current delivered to the motor to 1.5 times the continuous current rating of the motor.

Manual Adjust

Opens the Limits tab of the Manual Adjust dialog box for online editing of the Position Error Tolerance, Position Lock Tolerance, Peak Torque/Force Limit, and Continuous Torque/Force Limit parameters.



Position Error Tolerance: 0.0 Position Units

Position Lock Tolerance: 0.01 Position Units

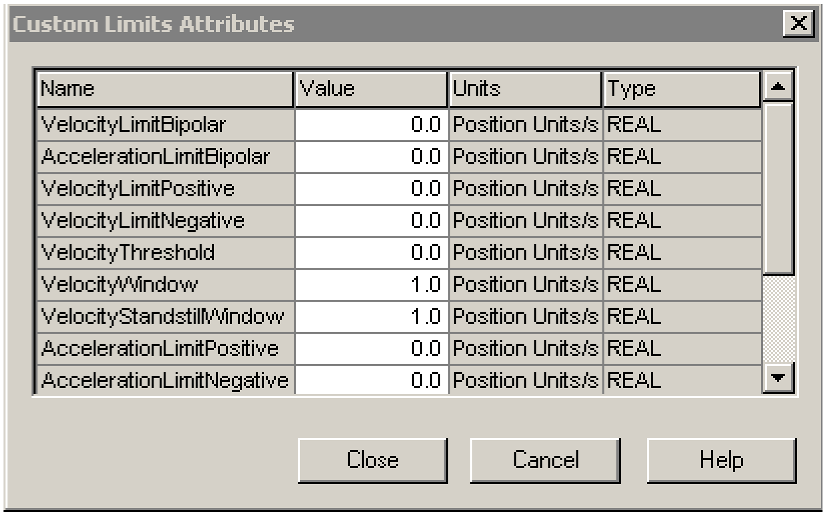
PeakTorque/Force Limit: 0.0 % Rated

Continuous Torque/Force Limit: 100.0 % Rated

Manual Adjust is unavailable when RSLogix 5000 software is in Wizard mode, and when offline edits to the above parameters have not yet been saved or applied.

Set Custom Limits

Opens the Custom Limit Attributes dialog box.



Name	Value	Units	Type
VelocityLimitBipolar	0.0	Position Units/s	REAL
AccelerationLimitBipolar	0.0	Position Units/s	REAL
VelocityLimitPositive	0.0	Position Units/s	REAL
VelocityLimitNegative	0.0	Position Units/s	REAL
VelocityThreshold	0.0	Position Units/s	REAL
VelocityWindow	1.0	Position Units/s	REAL
VelocityStandstillWindow	1.0	Position Units/s	REAL
AccelerationLimitPositive	0.0	Position Units/s	REAL
AccelerationLimitNegative	0.0	Position Units/s	REAL

From this dialog box you can monitor and edit the limit-related attributes.

When RSLogix 5000 software is online, the parameters on this tab transition to a read-only state. When a parameter transitions to a read-only state, any pending changes to parameter values are lost, and the parameter reverts to the most recently saved parameter value.

When multiple workstations connect to the same controller using RSLogix 5000 software and invoke the Axis Wizard or Axis Properties dialog box, the firmware allows only the first workstation to make any changes to axis attributes. The second workstation switches to a Read Only mode, indicated in the title bar, so that you may view the changes from that workstation, but not edit them.

Attributes

The following attribute values can be monitored and edited in this dialog box.

Monitored Attributes

Attribute	Description
VelocityLimitBipolar	This attribute sets the velocity limit symmetrically in both directions. If the command velocity exceeds this value, VelocityLimitStatusBit of the DriveStatus attribute is set. This attribute has a value range of 0 to 2.14748×10^{12}.
AccelerationLimitBipolar	This attribute sets the acceleration and deceleration limits for the drive. If the command acceleration exceeds this value, AccelLimitStatusBit of the DriveStatus attribute is set. This attribute has a value range of 0 to 2.14748×10^{15}.
TorqueLimitBipolar	This attribute sets the torque limit symmetrically in both directions. When actual torque exceeds this value TorqueLimitStatus of the DriveStatus attribute is set. This attribute has a value range of 0 to 1000.

Monitored Attributes

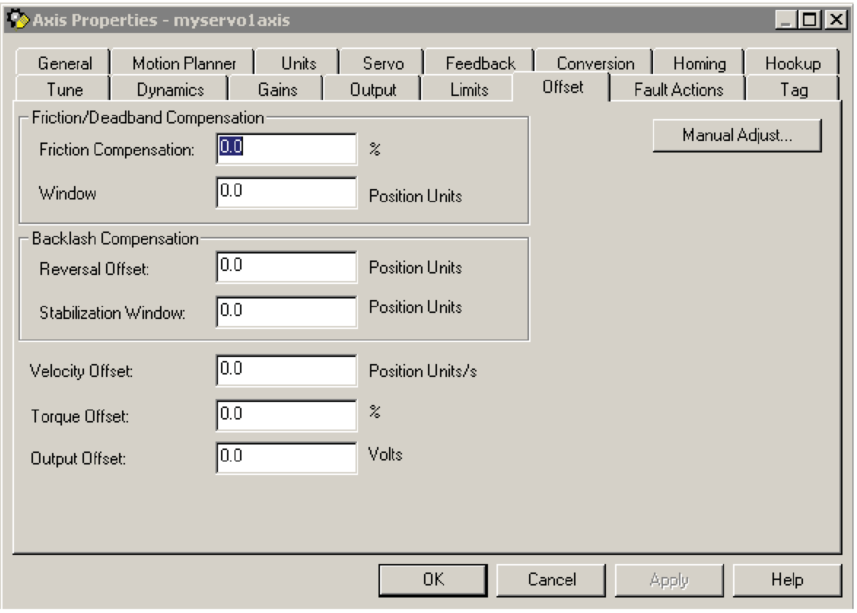
Attribute	Description
VelocityLimitPositive	This attribute displays the maximum allowable velocity in the positive direction. If the velocity limit is exceeded, bit 5 ('Velocity Command Above Velocity Limit') VelocityLimitStatusBit of the DriveStatus attribute is set. This attribute has a value range of 0 to 2.14748×10^{12}.
VelocityLimitNegative	This attribute displays the maximum allowable velocity in the negative direction. If the velocity limit is exceeded, bit 5 ('Velocity Command Above Velocity Limit') VelocityLimitStatusBit of the DriveStatus attribute is set. This attribute has a value range of -2.14748×10^{12} to 0.
VelocityThreshold	This attribute displays the velocity threshold limit. If the motor velocity is less than this limit, VelocityThresholdStatus of the DriveStatus attribute is set. This attribute has a value range of 0 to 2.14748×10^{12}.
VelocityWindow	This attribute displays the limits of the velocity window. If the motor's actual velocity differs from the command velocity by an amount less that this limit VelocityLockStatus of the DriveStatus attribute is set. This attribute has a value range of 0 to 2.14748×10^{12}.
VelocityStandstillWindow	This attribute displays the velocity limit for the standstill window. If the motor velocity is less than this limit VelocityStandStillStatus of the DriveStatus bit is set. This attribute has a value range of 0 to 2.14748×10^{12}.
AccelerationLimitPositive	This attribute limits the maximum acceleration ability of the drive to the programmed value. If the command acceleration exceeds this value, AccelLimitStatusBit of the DriveStatus attribute is set. This attribute has a value range of 0 to 2.14748×10^{15}.
AccelerationLimitNegative	This attribute limits the maximum acceleration ability of the drive to the programmed value. If the command acceleration exceeds this value, the AccelLimitStatus bit of the DriveStatus attribute is set. This attribute has a value range of -2.14748×10^{15} to 0.
TorqueLimitPositive	This attribute displays the maximum torque in the positive direction. If the torque limit is exceeded, the TorqueLimitStatus bit of the DriveStatus attribute is set. This attribute has a value range of 0 to 1000.
TorqueLimitNegative	This attribute displays the maximum torque in the negative direction. If the torque limit is exceeded, the TorqueLimitStatus bit of the DriveStatus attribute is set. This attribute has a value range of -1000 to 0.
TorqueThreshold	This attribute displays the torque threshold. If this limit is exceeded, the TorqueThreshold bit of the DriveStatus attribute is set. This attribute has a value range of 0 to 1000.

Offset Tab - AXIS_SERVO

Use this tab to make offline adjustments to the following Servo Output values:

- Backlash Compensation
- Velocity Offset
- Torque Offset
- Output Offset

For an axis of the type AXIS_SERVO configured as a Servo drive in the General tab of this dialog box.



Axis Properties - myservo1axis

Friction/Deadband Compensation

Friction Compensation: 0.0 %

Window 0.0 Position Units

Backlash Compensation

Reversal Offset: 0.0 Position Units

Stabilization Window: 0.0 Position Units

Velocity Offset: 0.0 Position Units/s

Torque Offset: 0.0 %

Output Offset: 0.0 Volts

The parameters on this tab can be edited in either of two ways:

- edit on this tab by typing your parameter changes and then click OK or Apply to save your edits.

- edit in the Manual Adjust dialog box: click Manual Adjust to open the Manual Adjust dialog box to this tab and use the spin controls to edit parameter settings. Your changes are saved the moment a spin control changes any parameter value.

The parameters on this tab become read-only and cannot be edited when the controller is online if the controller is set to Hard Run mode, or if a Feedback On condition exists.

When RSLogix 5000 software is offline, the following parameters can be edited and the program saved to disk using either the Save command or by clicking Apply. You must re-download the edited program to the controller before it can be run.

Friction/Deadband Compensation and Backlash Compensation

The percentage of output level added to a positive current Servo Output value, or subtracted from a negative current Servo Output value, for the purpose of moving an axis that is stuck in place due to static friction.

It is not unusual for an axis to have enough static friction (called 'sticktion') that, even with a significant position error, the axis refuses to budge. Backlash Compensation is used to break 'sticktion' in the presence of a nonzero position error. This is done by adding, or subtracting, a percentage output level), called Backlash Compensation to the Servo Output value.

The Backlash Compensation value should be just less than the value that would break the 'sticktion'. A larger value can cause the axis to 'dither', that is, move rapidly back and forth about the commanded position.

This controller attribute is replicated in the motion module.

Backlash Compensation Window

To address the issue of dither when applying Backlash Compensation and hunting from the integral gain, a Backlash Compensation Window is applied around the current command position when the axis is not being commanded to move. If the actual position is within the Backlash Compensation Window the Backlash Compensation value is applied to the Servo Output but scaled by the ratio of the position error to the Backlash Compensation Window. Within the window, the servo integrators are also disabled. Thus, once the position error reaches or exceeds the value of the Backlash Compensation Window attribute, the full Backlash Compensation value is applied. If the Backlash Compensation Window is set to zero, this feature is effectively disabled.

A nonzero Backlash Compensation Window has the effect of softening the Backlash Compensation as its applied to the Servo Output and reducing the dithering effect that it can create. This generally allows higher values of Backlash Compensation to be applied. Hunting is also eliminated at the cost of a small steady-state error.

Backlash Compensation and Reversal Offset

Backlash Reversal Offset provides the capability to compensate for positional inaccuracy introduced by mechanical backlash. For example, power-train type applications require a high level of accuracy and repeatability during machining operations. Axis motion is often generated by a number of mechanical components, a motor, a gearbox, and a ball-screw that may introduce inaccuracies and that are subject to wear over their lifetime. Therefore, when an axis is commanded to reverse direction, mechanical play in the machine (through the gearing, ball-screw, and so on) may result in a small amount of motor motion without axis motion. As a result, the feedback device may indicate movement even though the axis has not physically moved.

If a value of zero is applied to the Backlash Reversal Offset, the feature is effectively disabled. Once enabled by a nonzero value, and the load is engaged by a reversal of the commanded motion, changing the Backlash Reversal

Offset can cause the axis to shift as the offset correction is applied to the command position.

Stabilization Window

The Backlash Stabilization Window controls the Backlash Stabilization feature in the servo control loop.

Properly configured with a suitable value for the Backlash Stabilization Window, entirely eliminates the gearbox buzz without sacrificing any servo performance. In general, this value should be set to the measured backlash distance. A Backlash Stabilization Window value of zero effectively disables the feature.

Velocity Offset

Provides a dynamic velocity correction to the output of the position servo loop, in position units per second.

Torque Offset

Provides a dynamic torque command correction to the output of the velocity servo loop, as a percentage of velocity servo loop output.

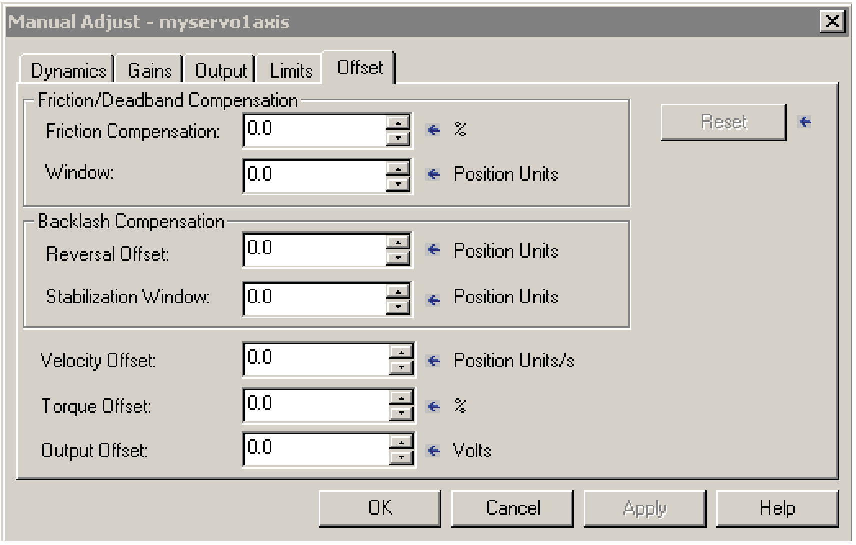
Output Offset

Corrects the problem of axis 'drift', by adding a fixed voltage value (not to exceed ±10 Volts) to the Servo Output value. Input a value to achieve near zero drive velocity when the uncompensated Servo Output value is zero.

When interfacing an external drive – especially for velocity drives, it is necessary to compensate for the effect of drive offset. Cumulative offsets of the servo module's DAC output and the Servo Drive Input result in a situation where a zero commanded Servo Output value causes the axis to 'drift'. If the drift is excessive, it can cause problems with the Hookup Diagnostic and Tuning procedures, as well as result in a steady-state nonzero position error when the servo loop is closed.

Manual Adjust

Opens the Offset tab of the Manual Adjust dialog box for online editing of the Friction/Deadband Compensation, Backlash Compensation, Velocity Offset, Torque Offset, and Output Offset parameters.



Friction/Deadband Compensation
Friction Compensation: 0.0 %
Window: 0.0 Position Units

Backlash Compensation
Reversal Offset: 0.0 Position Units
Stabilization Window: 0.0 Position Units

Velocity Offset: 0.0 Position Units/s
Torque Offset: 0.0 %
Output Offset: 0.0 Volts

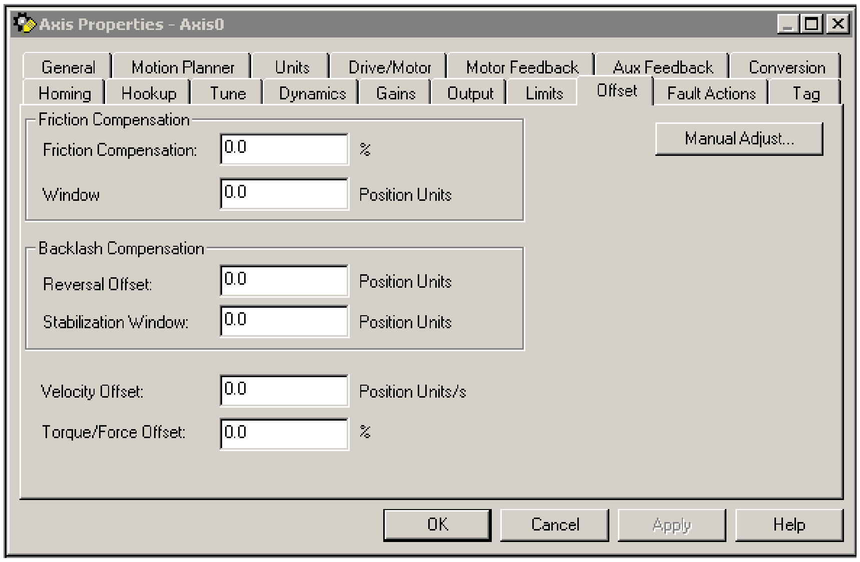
Manual Adjust is unavailable when RSLogix 5000 software is in Wizard mode, and when offline edits to the above parameters have not yet been saved or applied.

Offset Tab - AXIS_SERVO_DRIVE

Use this tab to make offline adjustments to the following Servo Output values:

- Backlash Compensation,
- Velocity Offset, and
- Torque Offset

For an axis of the type AXIS_SERVO_DRIVE configured as a Servo drive in the General tab of this dialog box.



Friction Compensation

Friction Compensation: 0.0 %

Window 0.0 Position Units

Backlash Compensation

Reversal Offset: 0.0 Position Units

Stabilization Window: 0.0 Position Units

Velocity Offset: 0.0 Position Units/s

Torque/Force Offset: 0.0 %

The parameters on this tab can be edited in either of two ways:

- edit on this tab by typing your parameter changes and then click OK or Apply to save your edits
- edit in the Manual Adjust dialog box: click Manual Adjust to open the Manual Adjust dialog box to this tab and use the spin controls to edit parameter settings. Your changes are saved the moment a spin control changes any parameter value.

The parameters on this tab become read-only and cannot be edited when the controller is online if the controller is set to Hard Run mode, or if a Feedback On condition exists.

When RSLogix 5000 software is offline, the following parameters can be edited and the program saved to disk using either the Save command or by clicking Apply. You must re-download the edited program to the controller before it can be run.

Backlash Compensation

The percentage of output level added to a positive current Servo Output value, or subtracted from a negative current Servo Output value, for the purpose of moving an axis that is stuck in place due to static friction.

It is not unusual for an axis to have enough static friction – called sticktion – that, even with a significant position error, the axis refuses to budge. Backlash Compensation is used to break sticktion in the presence of a nonzero position error. This is done by adding, or subtracting, a percentage output level), called Backlash Compensation to the Servo Output value.

The Backlash Compensation value should be just less than the value that would break the 'sticktion'. A larger value can cause the axis to 'dither', that is, move rapidly back and forth about the commanded position.

Backlash Compensation Window

To address the issue of dither when applying Backlash Compensation and hunting from the integral gain, a Backlash Compensation Window is applied around the current command position when the axis is not being commanded to move. If the actual position is within the Backlash Compensation Window the Backlash Compensation value is applied to the Servo Output but scaled by the ratio of the position error to the Backlash Compensation Window. Within the window, the servo integrators are also disabled. Thus, once the position error reaches or exceeds the value of the Backlash Compensation Window attribute, the full Backlash Compensation value is applied. If the Backlash Compensation Window is set to zero, this feature is effectively disabled.

A nonzero Backlash Compensation Window has the effect of softening the Backlash Compensation as its applied to the Servo Output and reducing the dithering effect that it can create. This generally allows higher values of Backlash Compensation to be applied. Hunting is also eliminated at the cost of a small steady-state error.

Backlash Compensation and Reversal Offset

Backlash Reversal Offset provides the capability to compensate for positional inaccuracy introduced by mechanical backlash. For example, power-train type applications require a high level of accuracy and repeatability during machining operations. Axis motion is often generated by a number of mechanical components, a motor, a gearbox, and a ball-screw that may introduce inaccuracies and that are subject to wear over their lifetime. Therefore, when an axis is commanded to reverse direction, mechanical play in the machine (through the gearing, ball-screw, and so on) may result in a small amount of motor motion without axis motion. As a result, the feedback device may indicate movement even though the axis has not physically moved.

If a value of zero is applied to the Backlash Reversal Offset, the feature is unavailable. Once enabled by a nonzero value, and the load is engaged by a reversal of the commanded motion, changing the Backlash Reversal Offset can cause the axis to shift as the offset correction is applied to the command position.

Stabilization Window

The Backlash Stabilization Window controls the Backlash Stabilization feature in the servo control loop.

Properly configured with a suitable value for the Backlash Stabilization Window, entirely eliminates the gearbox buzz without sacrificing any servo performance. In general, this value should be set to the measured backlash distance. This feature is unavailable when a Backlash Stabilization Window value is zero.

Velocity Offset

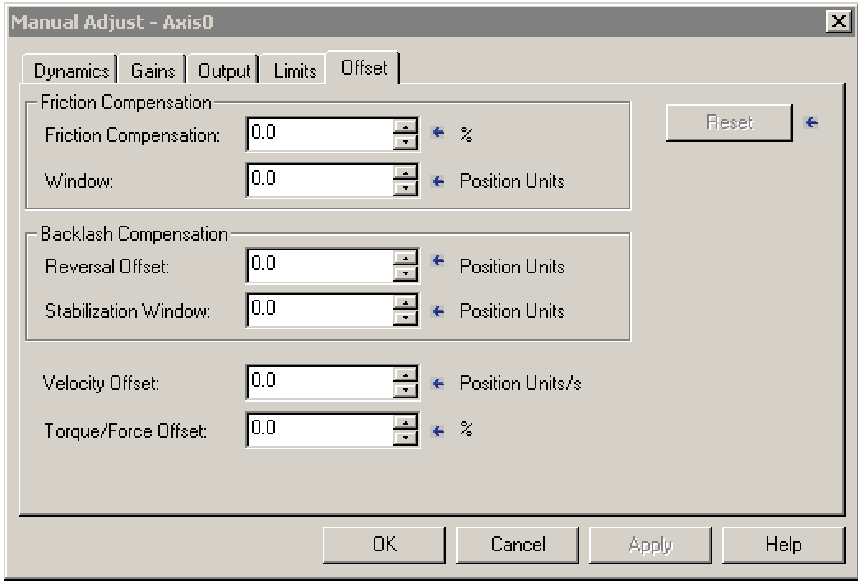
Provides a dynamic velocity correction to the output of the position servo loop, in position units per second.

Torque/Force Offset

Provides a dynamic torque command correction to the output of the velocity servo loop, as a percentage of velocity servo loop output.

Manual Adjust

Opens the Offset tab of the Manual Adjust dialog box for online editing of the Friction/Deadband Compensation, Backlash Compensation, Velocity Offset, Torque Offset, and Output Offset parameters.



Friction Compensation
Friction Compensation: 0.0 %
Window: 0.0 Position Units

Backlash Compensation
Reversal Offset: 0.0 Position Units
Stabilization Window: 0.0 Position Units

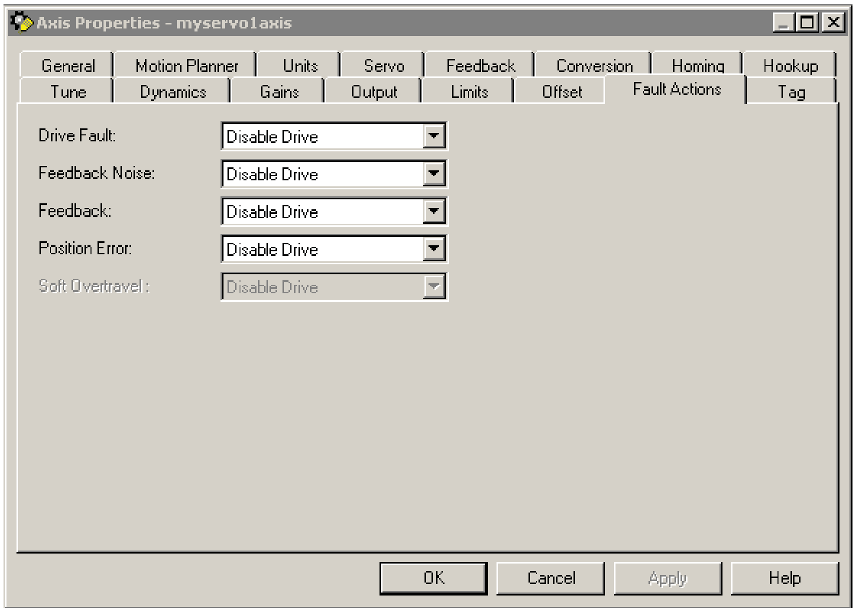
Velocity Offset: 0.0 Position Units/s
Torque/Force Offset: 0.0 %

Manual Adjust is unavailable when RSLogix 5000 software is in Wizard mode, and when offline edits to the above parameters have not yet been saved or applied.

Fault Actions Tab - AXIS_SERVO

Use this tab to specify the actions that are taken in response to these faults:

- Drive Fault
- Feedback Noise Fault
- Feedback Loss Fault
- Position Error Fault
- Soft Overtravel Fault



Drive Fault: Disable Drive
Feedback Noise: Disable Drive
Feedback: Disable Drive
Position Error: Disable Drive
Soft Overtravel : Disable Drive

When a parameter transitions to a read-only state, any pending changes to parameter values are lost, and the parameter reverts to the most recently saved parameter value.

When multiple workstations connect to the same controller using RSLogix 5000 software and invoke the Axis Wizard or Axis Properties dialog box, the firmware allows only the first workstation to make any changes to axis attributes. The second workstation switches to a Read Only mode, indicated in the title bar, so that you may view the changes from that workstation, but not edit them.

Select one of the following fault actions for each fault type:

- Shutdown - If a fault action is set to Shutdown, then when the associated fault occurs, axis servo action is unavailable, the servo amplifier output is zeroed, and the appropriate drive enable output is deactivated. Shutdown is the most severe action to a fault and it is usually reserved for faults that could endanger the machine or the operator if power is not removed as quickly and completely as possible.

- Disable Drive - If a fault action is set to Disable Drive, then when the associated fault occurs, axis servo action is unavailable, the servo amplifier output is zeroed, and the appropriate drive enable output is deactivated.

- Stop Motion - If a fault action is set to Stop Motion, then when the associated fault occurs, the axis immediately starts decelerating the axis command position to a stop at the configured Maximum Deceleration Rate without disabling servo action or the servo modules Drive Enable output. This is the gentlest stopping mechanism in response to a fault. It is usually used for less severe faults. After the stop command fault action has stopped the axis, no further motion can be generated until the fault is first cleared.

- Status Only - If a fault action is set to Status Only, then when the associated fault occurs, no action is taken. The application program must handle any motion faults. In general, this setting should only be used in applications where the standard fault actions are not appropriate.

ATTENTION ⚠ Selecting the wrong fault action for your application can cause a dangerous condition resulting in unexpected motion, damage to the equipment, and physical injury or death. Keep clear of moving machinery.

Drive Fault

The Drive Fault field lets you specify the fault action to be taken when a drive fault condition is detected, for an axis with the Drive Fault Input enabled (in the Servo tab of this dialog box) that is configured as Servo (in the General tab of this dialog box). The available actions for this fault are Shutdown, Disable Drive, Stop Motion, and Status Only.

Feedback Noise

The Feedback noise field lets you specify the fault action to be taken when excessive feedback noise is detected. The available actions for this fault are Shutdown, Disable Drive, Stop Motion, and Status Only.

Feedback Loss

The Feedback Loss field lets you specify the fault action to be taken when feedback loss condition is detected. The available actions for this fault are Shutdown, Disable Drive, Stop Motion, and Status Only.

Position Error

The Position Error field lets you specify the fault action to be taken when position error exceeds the position tolerance set for the axis, for an axis configured as Servo (in the General tab of this dialog box). The available actions for this fault are Shutdown, Disable Drive, Stop Motion, and Status Only.

Soft Overtravel

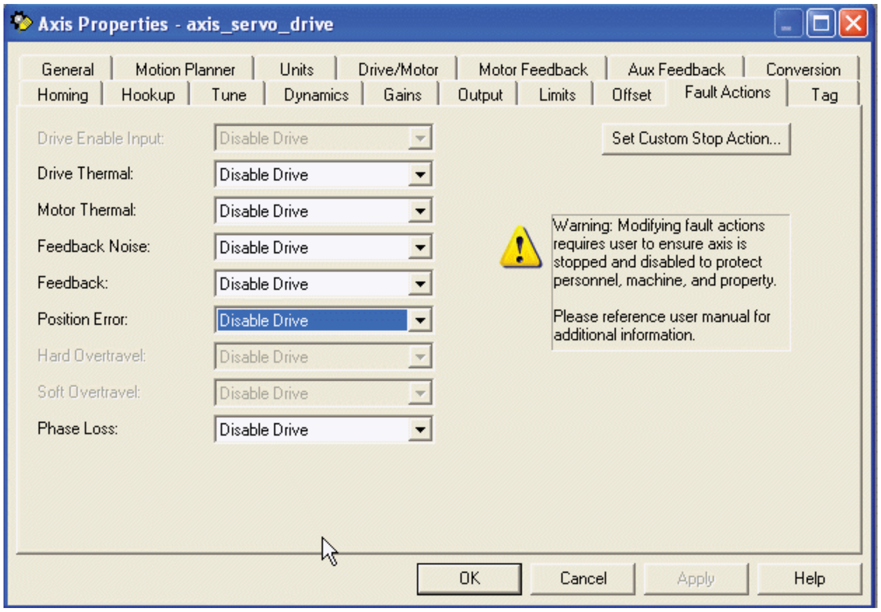
The Soft Overtravel field lets you specify the fault action to be taken when a software overtravel error occurs, for an axis with Soft Travel Limits enabled and configured (in the Limits tab of this dialog box) that is configured as Servo (in the General tab of this dialog box). The available actions for this fault are Shutdown, Disable Drive, Stop Motion, and Status Only.

Fault Actions Tab - AXIS_SERVO_DRIVE

Use this tab to specify the actions that are taken in response to the following faults:

- Drive Enable Input
- Drive Thermal Fault
- Motor Thermal Fault
- Feedback Noise Fault
- Feedback Fault
- Position Error Fault
- Hard Overtravel Fault
- Soft Overtravel Fault
- Phase Loss

for an axis of the type AXIS_SERVO_DRIVE.



Drive Enable Input: Disable Drive
Drive Thermal: Disable Drive
Motor Thermal: Disable Drive
Feedback Noise: Disable Drive
Feedback: Disable Drive
Position Error: Disable Drive
Hard Overtravel: Disable Drive
Soft Overtravel: Disable Drive
Phase Loss: Disable Drive

Warning: Modifying fault actions requires user to ensure axis is stopped and disabled to protect personnel, machine, and property.

Please reference user manual for additional information.

When a parameter transitions to a read-only state, any pending changes to parameter values are lost, and the parameter reverts to the most recently saved parameter value.

When multiple workstations connect to the same controller using RSLogix 5000 software and invoke the Axis Wizard or Axis Properties dialog box, the firmware allows only the first workstation to make any changes to axis attributes. The second workstation switches to a Read Only mode, indicated in the title bar, so that you may view the changes from that workstation, but not edit them.

Select one of the following fault actions for each fault type:

- Shutdown - If a fault action is set to Shutdown, then when the associated fault occurs, axis servo action is immediately disabled, the servo amplifier output is zeroed, and the appropriate drive enable output is deactivated. Shutdown is the most severe action to a fault and it is usually reserved for faults that could endanger the machine or the operator if power is not removed as quickly and completely as possible.

- Disable Drive - If a fault action is set to Disable Drive, then when the associated fault occurs, it brings the axis to a stop by applying the Stopping Torque for up to the Stopping Time Limit. During this period the servo is active but no longer tracking the command reference from logix. Once the axis is stopped (or the stopping limit is exceeded) the servo and power structure are disabled.

- Stop Motion - If a fault action is set to Stop Motion, then when the associated fault occurs, the axis immediately starts decelerating the axis command position to a stop at the configured Maximum Deceleration Rate without disabling servo action or the servo modules Drive Enable output. This is the gentlest stopping mechanism in response to a fault. It is usually used for less severe faults. After the stop command fault action has stopped the axis, no further motion can be generated until the fault is first cleared.

- Status Only - If a fault action is set to Status Only, then when the associated fault occurs, no action is taken. The application program must handle any motion faults. In general, this setting should only be used in applications where the standard fault actions are not appropriate.

ATTENTION ⚠ Selecting the wrong fault action for your application can cause a dangerous condition. Keep clear of moving machinery.

Drive Enable Input

Check this box if you wish to enable the Drive Fault detection. When the drive fault is detected, appropriate action is taken based on the Drive Fault Action specified in the Fault Actions tab of this dialog box.

Drive Thermal

Specifies the fault action to be taken when a Drive Thermal Fault is detected, for an axis configured as Servo (in the General tab of this dialog box). The available actions for this fault are Shutdown, Disable Drive, Stop Motion, and Status Only.

Motor Thermal

Specifies the fault action to be taken when a Motor Thermal Fault is detected, for an axis configured as Servo (in the General tab of this dialog box). The available actions for this fault are Shutdown, Disable Drive, Stop Motion, and Status Only.

Feedback Noise

Specifies the fault action to be taken when excessive feedback noise is detected. The available actions for this fault are Shutdown, Disable Drive, Stop Motion, and Status Only.

Feedback

Specifies the fault action to be taken when Feedback Fault is detected. The available actions for this fault are Shutdown, Disable Drive, Stop Motion, and Status Only.

Position Error

Specifies the fault action to be taken when position error exceeds the position tolerance set for the axis, for an axis configured as Servo (in the General tab of this dialog box). The available actions for this fault are Shutdown, Disable Drive, Stop Motion, and Status Only.

Hard Overtravel

Specifies the fault action to be taken when an axis encounters a travel limit switch, for an axis configured as Servo (in the General tab of this dialog box). The available actions for this fault are Shutdown, Disable Drive, Stop Motion, and Status Only.

Soft Overtravel

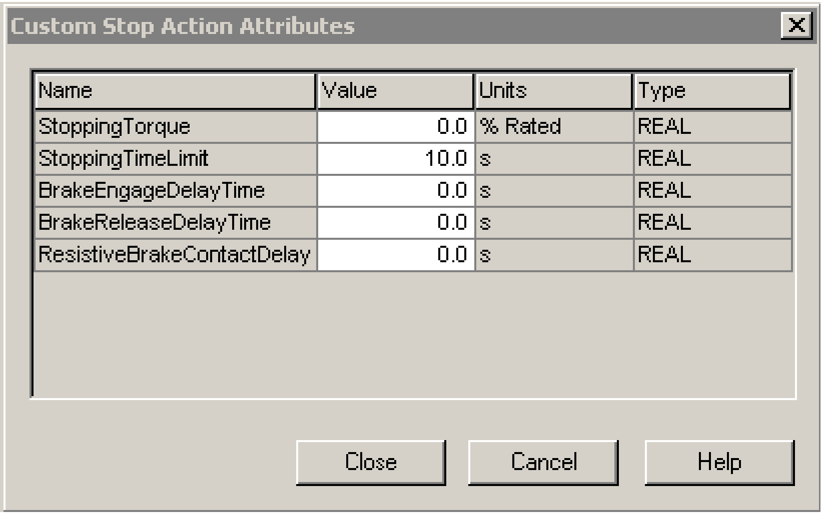
Specifies the fault action to be taken when a software overtravel error occurs, for an axis with Soft Travel Limits enabled and configured (in the Limits tab of this dialog box) that is configured as Servo (in the General tab of this dialog box). The available actions for this fault are Shutdown, Disable Drive, Stop Motion, and Status Only.

Phase Loss

Specifies the fault action to be taken when a phase loss situation occurs for an axis configured as Servo (on the General tab of this dialog box). The available actions for this fault are Shutdown, Disable Drive, Stop Motion and Status Only. The default is Shutdown. When Status Only is chosen Logix 5000 motion commands continue and the drive uses available stored DC bus energy to operate the axes.

Set Custom Stop Action

Opens the Custom Stop Action Attributes dialog box.



Name	Value	Units	Type
StoppingTorque	0.0	% Rated	REAL
StoppingTimeLimit	10.0	s	REAL
BrakeEngageDelayTime	0.0	s	REAL
BrakeReleaseDelayTime	0.0	s	REAL
ResistiveBrakeContactDelay	0.0	s	REAL

Use this dialog box to monitor and edit the Stop Action-related attributes.

When a parameter transitions to a read-only state, any pending changes to parameter values are lost, and the parameter reverts to the most recently saved parameter value.

When multiple workstations connect to the same controller using RSLogix 5000 software and invoke the Axis Wizard or Axis Properties dialog box, the firmware allows only the first workstation to make any changes to axis attributes. The second workstation switches to a Read Only mode, indicated in the title bar, so that you may view the changes from that workstation, but not edit them.

Attributes

The following attribute, or parameter, values can be monitored and edited in this dialog box.

Monitored Attributes

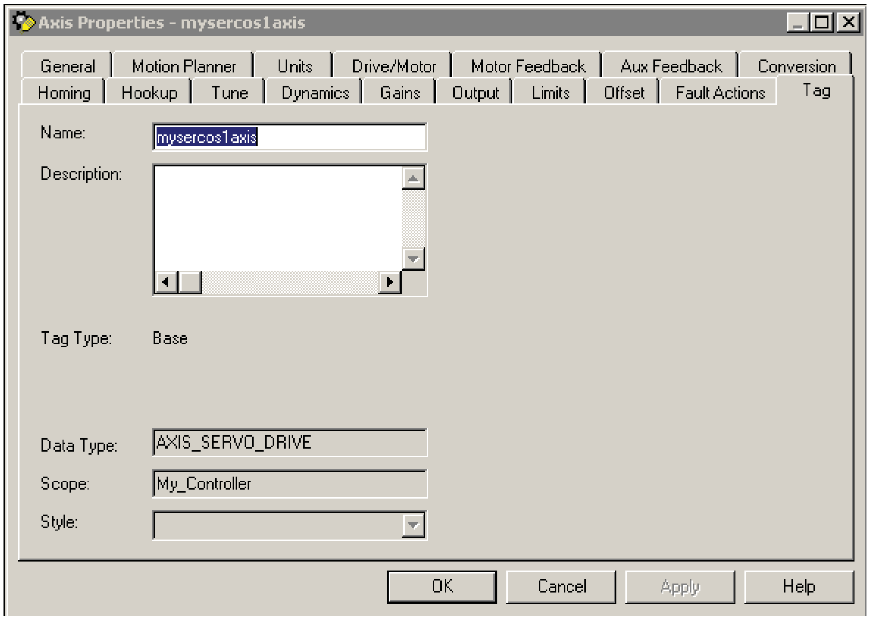
Attribute	Description
StoppingTorque	This attribute displays the amount of torque available to stop the motor. This attribute has a value range of 0…1000.
StoppingTimeLimit	This attribute displays the maximum amount of time that the drive amplifier remains enabled while trying to stop. It is useful for very slow velocity rate change settings. This attribute has a value range of 0…6553.5 units.

Monitored Attributes

Attribute	Description
BrakeEngageDelayTime	When servo axis is disabled and the drive decelerates to a minimum speed, the drive maintains torque until this time has elapsed. This time allows the motor's brake to be set. This attribute has a value range of 0…6.5535.
BrakeReleaseDelayTime	When the servo axis is enabled, the drive activates the torque to the motor but ignores the command values from the Logix controller until this time has elapsed. This time allows the motor's brake to release. This attribute has a value of 0…6.5535.
ResistiveBrakeContactDelay	The Resistive Brake Contact Delay attribute is used to control an optional external Resistive Brake Module (RBM). The RBM sits between the drive and the motor and uses an internal contactor to switch the motor between the drive and a resisted load.

Tag Tab

Use this tab to modify the name and description of the axis. When you are online, all of the parameters on this tab transition to a read-only state, and cannot be modified. If you go online before you save your changes, all pending changes revert to their previously-saved state.


Name: mysercos1axis

Description:

Tag Type: Base

Data Type: AXIS_SERVO_DRIVE

Scope: My_Controller

Style:

Name

Displays the name of the current tag. You can rename this tag, if you wish.

Description

Displays the description of the current tag, if any is available. You can edit this description, if you wish.

Tag Type

Indicates the type of the current tag. This type may be:

- Base
- Alias
- Consumed

Displays the data type associated with the current tag.

Data Type

Displays the axis data type of the current tag.

Scope

Displays the scope of the current tag. The scope is either controller scope, or program scope, based on one of the existing programs in the controller.

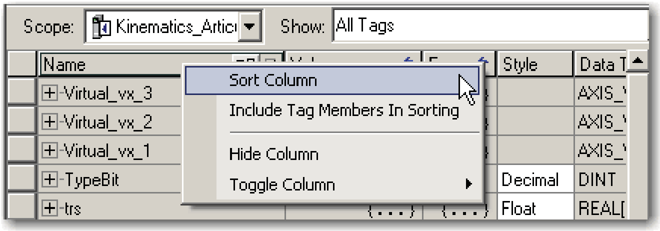
Style

Displays the default style in which to display the value of the tag. Note that style is only applicable to an atomic tag; a structure tag does not have a display style.

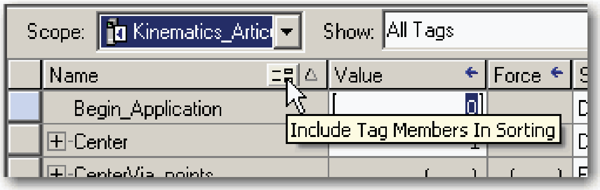
Monitoring Axis Tags

In Edit Tags or Monitor Tags, you can sort the tags alphabetically.

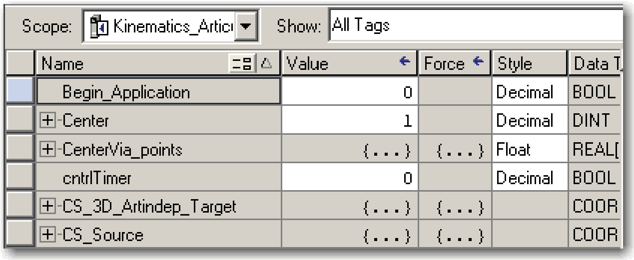
Right click on the Name column and click Sort Column.



You also have the option to: Include Tag Members in Sorting.



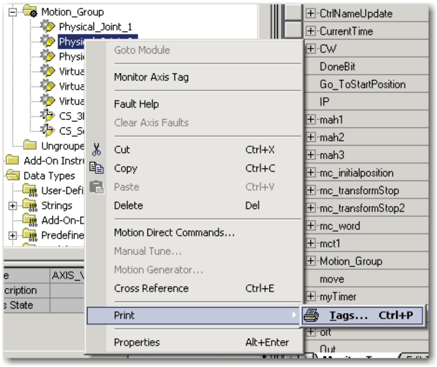
Ascending Tag Sort Order



Name	Value	Force	Style	Data T
Begin_Application	0		Decimal	BOOL
Center	1		Decimal	DINT
CenterVia_points	{...}	{...}	Float	REAL[
cntrlTimer	0		Decimal	BOOL
CS_3D_Artindep_Target	{...}	{...}		COOR
CS_Source	{...}	{...}		COOR

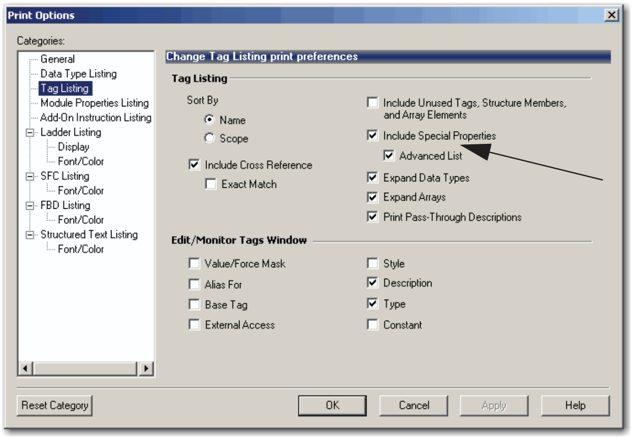
Creating Reports

In RSLogix 5000 software you are able to print a variety of reports. For example, right click on Controller Tags, MainTask, MainProgram, axis, Add-On Instructions, Data Types.



On the Print dialog box, select the Adobe PDF and click Print Options.

When printing a Tag Listing, be sure to check the Include Special Properties and Advanced List to see all the information.



Tag Listing

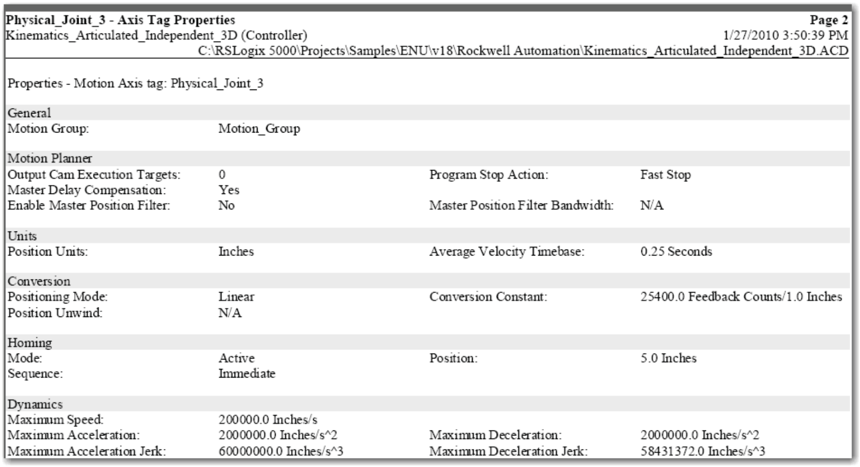
Sort By
- ● Name
- ○ Scope

☑ Include Cross Reference
☐ Exact Match
☐ Include Unused Tags, Structure Members, and Array Elements
☑ Include Special Properties
☑ Advanced List
☑ Expand Data Types
☑ Expand Arrays
☑ Print Pass-Through Descriptions

Edit/Monitor Tags Window

☐ Value/Force Mask
☐ Alias For
☐ Base Tag
☐ External Access
☐ Style
☑ Description
☑ Type
☐ Constant

Axis Properties Example

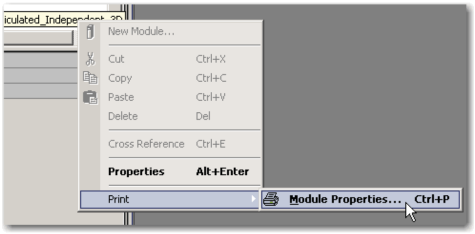


Physical_Joint_3 - Axis Tag Properties **Page 2**
Kinematics_Articulated_Independent_3D (Controller) 1/27/2010 3:50:39 PM
C:\RSLogix 5000\Projects\Samples\ENU\v18\Rockwell Automation\Kinematics_Articulated_Independent_3D.ACD

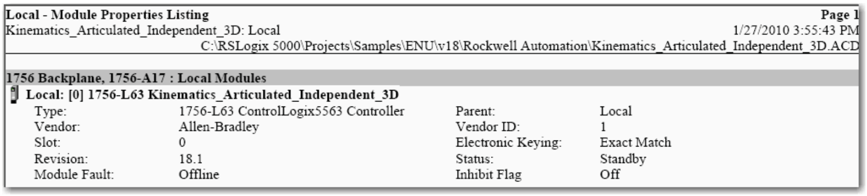
Properties - Motion Axis tag: Physical_Joint_3

General			
Motion Group:	Motion_Group		
Motion Planner			
Output Cam Execution Targets:	0	Program Stop Action:	Fast Stop
Master Delay Compensation:	Yes		
Enable Master Position Filter:	No	Master Position Filter Bandwidth:	N/A
Units			
Position Units:	Inches	Average Velocity Timebase:	0.25 Seconds
Conversion			
Positioning Mode:	Linear	Conversion Constant:	25400.0 Feedback Counts/1.0 Inches
Position Unwind:	N/A		
Homing			
Mode:	Active	Position:	5.0 Inches
Sequence:	Immediate		
Dynamics			
Maximum Speed:	200000.0 Inches/s		
Maximum Acceleration:	2000000.0 Inches/s^2	Maximum Deceleration:	2000000.0 Inches/s^2
Maximum Acceleration Jerk:	60000000.0 Inches/s^3	Maximum Deceleration Jerk:	58431372.0 Inches/s^3

You can also right click on a controller, communication module, and any motion module to print the Module Properties you have configured.



Module Properties Example



Local - Module Properties Listing **Page 1**

Kinematics_Articulated_Independent_3D: Local 1/27/2010 3:55:43 PM

C:\RSLogix 5000\Projects\Samples\ENU\v18\Rockwell Automation\Kinematics_Articulated_Independent_3D.ACD

1756 Backplane, 1756-A17 : Local Modules

Local: [0] 1756-L63 Kinematics_Articulated_Independent_3D

Type:	1756-L63 ControlLogix5563 Controller	Parent:	Local
Vendor:	Allen-Bradley	Vendor ID:	1
Slot:	0	Electronic Keying:	Exact Match
Revision:	18.1	Status:	Standby
Module Fault:	Offline	Inhibit Flag	Off

Motion Axis Attributes

Introduction

Use this chapter to get configuration, status, and fault information about an axis. The controller stores information about an axis as attributes of the axis.

Access Attributes

The Access column shows how to access the attribute.

Example

Use a Get System Value (GSV) instruction to get the value.

Use a Set System Value (SSV) instruction to set or change the value. The attribute can only be modified when the axis is not enabled.

Attribute	Axis Type	Data Type	Access	Description
Acceleration Feedforward Gain			GSV SSV	
Accel Status			Tag	
Actual Acceleration			GSV Tag	

Use the tag for the axis to get the value.

Use the tag for the axis or a GSV instruction to get the value. It's easier to use the tag.

Accessing an MSG Instruction

For complete information on how to access data using an MSG instruction, refer to Logix5000 Controllers Messages Programming Manual, publication 1756-UM012.

If you need to find Attribute and Class IDs, refer to the specific drive documentation.

Interpreting the Attribute Tables

This table provides an explanation of the information, nomenclature, and abbreviations used in the attribute tables.

Attribute Table Column Heading Descriptions

Column Heading	Description
Attribute	Each attribute table begins with the attribute name. The tag, GSV/SSV, and MSG names for each of these attributes are the same as the attribute name listed, but with the spaces removed. For example, Absolute Feedback Enable would be AbsoluteFeedbackEnable.
Axis Type	For each attribute, the related axis is listed.
Data Type	For example, DINT, UINT, SINT, REAL, BOOL
Access	GSV Can be read via the GSV instruction. SSV Can be written via the SSV instruction. Tag Can be created to allocate and reference data. MSG Message is only used to access drive attributes for which there is not GSV/SSV access. In order to use a MSG instruction to access information from a drive, you will need the Attribute and Class IDs. Refer to the drive documentation for ID information.
Description	The meaning of the attribute values. For example: Position Units / Seconds. Tag access is supported but value is valid only when Auto Tag Update of the Motion Group Object is enabled.

Replicated Attributes

These are the controller attributes that are replicated in the motion module.

- AccelerationFeedForwardGain
- AxisType
- DriveFaultAction
- FeedbackFaultAction
- FeedbackNoiseFaultAction
- FrictionCompensation
- MaximumNegativeTravel
- MaximumPositiveTravel
- OutputLPFilterBandwidth
- OutputLimit
- OutputOffset
- PositionErrorFaultAction
- PositionErrorTolerance
- PositionIntegralGain
- PositionProportionalGain
- PositionUnwind
- SoftOvertravelFaultAction
- TorqueScaling
- VelocityFeedforwardGain
- VelocityIntegralGain
- VelocityProportionalGain
- VelocityScaling

Axis Attributes

This table describes each attribute of an axis.

Attribute	Axis Type	Data Type	Access	Description
Absolute Feedback Enable	AXIS_SERVO	SINT	GSV SSV	**Important**: Use this attribute only for an axis of a 1756-HYD02 or 1756-M02AS module. This attribute controls whether or not the servo module uses the absolute position capability of the feedback device. If Absolute Feedback Enable is set to True, the servo module adds the Absolute Feedback Offset to the current position of the feedback device to establish the absolute machine reference position. Since absolute feedback devices retain their position reference even through a power-cycle, the machine reference system can be restored at powerup. To establish a suitable value for the Absolute Feedback Offset attribute the MAH instruction may be executed with the Home Mode configured for Absolute (the only valid option when Absolute Feedback Enable is True). When executed, the servo module will compute the Absolute Feedback Offset as the difference between the configured value for Home Position and the current absolute feedback position of the axis. The computed Absolute Feedback Offset is immediately applied to the axis upon completion of the MAH instruction. Because the actual position of the axis is re-referenced during execution of the MAH instruction, the servo loop must not be active. If the servo loop is active the MAH instruction errors. If Absolute Feedback Enable is set to False, the servo module ignores the Absolute Feedback Offset and treats the feedback device as an incremental position transducer. In this case, a homing or redefine position operation is therefore needed to establish the absolute machine reference position. The Absolute Home Mode in this case is considered invalid. This attribute is configurable if the Transducer Type is set to SSI. For an LDT transducer the Absolute Feedback Enable is forced to True. For an AQB transducer the Absolute Feedback Enable is forced to False.

Attribute	Axis Type	Data Type	Access	Description
Absolute Feedback Offset	AXIS_SERVO	REAL	GSV SSV	Position Units **Important** • Use this attribute only for an axis of a 1756-HYD02 or 1756-M02AS module. • Set the Absolute Feedback Enable attribute to True. This attribute is used to determine the relative distance between the absolute position of the feedback device and the absolute position of the machine. At powerup this attribute is sent to the servo module and added to the current position of the feedback device to restore the absolute machine position reference. If the axis is configured for Linear operation, absolute position may be recovered after power cycle as long as the feedback device has not exceeded its range limit. If the feedback device rolls over its count range, the absolute position of the axis is no longer valid. If the axis is configured for Rotary operation, the servo module is responsible for adjusting the Absolute Feedback Offset dynamically based on the configured Unwind value and the rollover of the absolute feedback device. If necessary, absolute position may be recovered after power cycle by periodically updating the controller's Absolute Feedback Offset value. This can be done by selecting the Absolute Feedback Offset enumeration for one of the Axis Info Select attributes.
Absolute Reference Status	AXIS_SERVO_DRIVE	BOOL	Tag	<table><tr><th>If the bit is</th><th>Then</th></tr><tr><td>ON</td><td>An absolute homing procedure happened. The bit stays set until either of these happen: • The drive resets its configuration parameters to default values. • The axis does an active or passive home or redefine position. • MRP also clears Absolute Reference Status.</td></tr><tr><td>OFF</td><td>The position of the axis has not been, or is no longer, referenced to the absolute machine reference system</td></tr></table>

Attribute	Axis Type	Data Type	Access	Description

The AbsoluteReferenceStatus bit provides an indication whether the system has been absolutely referenced. When all conditions have been configured correctly, the AbsoluteReferenceStatus will be remain set through a power cycle, and the absolute position will remain intact.

Bit = 1 when (see chapter 5 - guidelines for homing), in addition to the data listed below

Bit = 0 when an absolute home has not taken place, in addition to the data listed below

Conditions that will cause the AbsoluteReferenceStatus bit to reset back to 0:

- Completing an MotionRedefinePosition (MRP) instruction (U3K & K6K)
- Completing an MAH and specifying non-abs homing type
- Replacing motor
- Successful execution of a non-absolute MAH (U3K & K6K)
- Offline, changing rotary to linear and vice versa and then downloading over the existing configuration (U3K)
- Changing Motor / Abs Feedback device
- Power cycle to an axis with a single turn feedback device configured as a linear axis (U3K)
- Power cycle to a single turn rotary axis with a Drive Resolution not equal to the Unwind (U3K & K6K)
- E20 fault - Motor Encoder State Error. Motor encoder has encountered an illegal state transition (U3K)
- E32 fault - S/C Frequency Exceeded. Maximum frequency if hardware exceeded (U3K)
- E73 Sercos Fault - Backplane Comm, Power Rail Backplane CAN communication failed. Typically a hardware failure or bent backplane pins (K6K)
- E76 Drive Hard Fault - CAN bit, either DPI or Backplane CAN initialization failed. Typically a hardware failure or bent backplane pins (K6K)
- Power cycle while auxiliary powered devices are producing excessive regenerative energy. Example: a fan or pump powered from the same supply powering aux power to the K6K
- Reset drive to defaults using Drive Explorer (K6K)
- Reset drive to defaults using Ultraware (U3K)
- Transitioning from Ring phase 3 or Ring phase 4 if the encoder is not a multi-turn or single-turn absolute device (K6K & U3K)

A few conditions that will not cause the AbsoluteReferenceStatus bit to reset to 0:

- Feedback Loss, even through a power cycle.
- Changing the unwind value
- Battery replacement or low battery if control power remains active
- Downloading Program
  Should stay intact after an upload or download as long the user uploads attribute to the offline image, once he is offline - otherwise the offline image will not have the bit set.

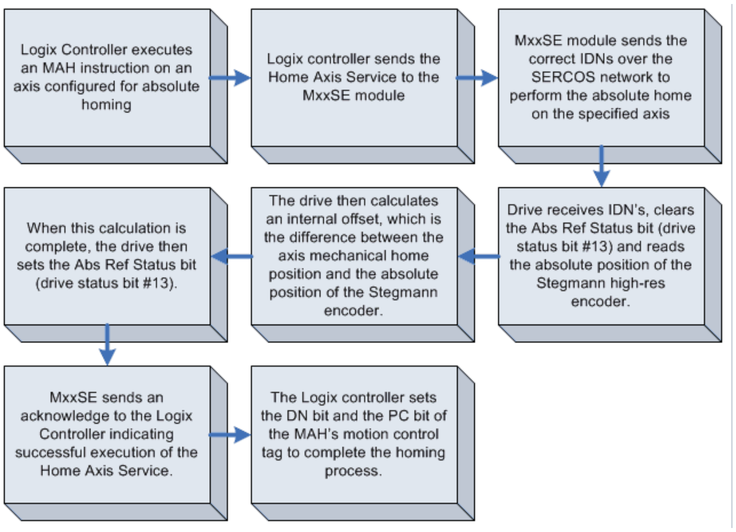
Auxiliary Axis (feedback only) absolute capabilities - K6K (U3K)

Auxiliary axes have the same capabilities for maintaining an absolute reference as the main feedback, except the auxiliary axis channel cannot generate a marker from any sine / cosine device. This would include the SRS / SRM feedback devices

Types of Absolute devices allowed for "AxisType - Axis_Servo_Drive"

- Stegmann Hiperface SRM / SKM Encoder
- Stegmann Hiperface SRS / SKS Encoder
- Tamagawa TL5669 Encoder

Sercos Absolute Homing Process Flowchart



Attribute	Axis Type	Data Type	Access	Description
Accel Limit Status	AXIS_SERVO_DRIVE	BOOL	Tag	Set when the magnitude of the commanded acceleration to the velocity servo loop input is greater than the configured Velocity Limit.
Accel Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set if the axis is currently being commanded to accelerate. Use the Accel Status bit and the Decel Status bit to see if the axis is accelerating or decelerating. If both bits are off, then the axis is moving at a steady speed or is at rest.
Acceleration Command	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV Tag	**Important**: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won't see the right value as the axis runs. See Axis Info Select 1. Acceleration Command in Position Units / Sec^2 Acceleration Command is the current acceleration reference to the output summing junction, in the configured axis Position Units per Second^2, for the specified axis. The Acceleration Command value, hence, represents the output of the inner velocity control loop. Acceleration Command is not to be confused with Command Velocity, which represents the rate of change of Command Position input to the position servo loop.
Acceleration Data Scaling	AXIS_SERVO_DRIVE	INT	GSV	This attribute is derived from the Drive Units attribute. See IDN 160 in IEC 1491.
Acceleration Data Scaling Exp	AXIS_SERVO_DRIVE	INT	GSV	This attribute is derived from the Drive Units attribute. See IDN 162 in IEC 1491.
Acceleration Data Scaling Factor	AXIS_SERVO_DRIVE	DINT	GSV	This attribute is derived from the Drive Units attribute. See IDN 161 in IEC 1491.

Attribute	Axis Type	Data Type	Access	Description
Acceleration Feedback	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV Tag	**Important**: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won't see the right value as the axis runs. See Axis Info Select 1. Acceleration Feedback in Position Units / Sec^2 Acceleration Feedback is the actual velocity of the axis as estimated by the servo module, in the configured axis Position Units per Second^2. The Estimated Acceleration is calculated by taking the difference in the Estimated Velocity over the servo update interval. Acceleration Feedback is a signed value—the sign (+ or -) depends on which direction the axis is currently moving.

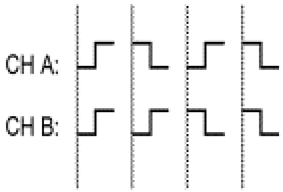
Attribute	Axis Type	Data Type	Access	Description
Acceleration Feedforward Gain	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV	% This controller attribute is also replicated in the motion module. **AXIS_SERVO** When you connect to a torque servo drive, use the Acceleration Feedforward Gain to give the Torque Command output necessary to generate the commanded acceleration. It does this by scaling the current Command Acceleration by the Acceleration Feedforward Gain and adding it as an offset to the Servo Output generated by the servo loop. With this done, the servo loops do not need to generate much of a contribution to the Servo Output, hence the Position and/or Velocity Error values are significantly reduced. Hence, when used in conjunction with the Velocity Feedforward Gain, the Acceleration Feedforward Gain lets the following error of the servo system during the acceleration and deceleration phases of motion be reduced to nearly zero. This is important in applications such as electronic gearing and synchronization where the actual axis position must not significantly lag behind the commanded position at any time. When you connect to a velocity servo drive, use Acceleration Feedforward to add a term to the Velocity Command that is proportional to the commanded acceleration. This can be effective in cases where the external drive shows a steady-state velocity error during acceleration and deceleration. The best value for Acceleration Feedforward depends on the drive configuration. Excessive Acceleration Feedforward values tend to produce axis overshoot. For torque servo drive applications the best value for Acceleration Feedforward is theoretically 100%. However, the value may need to be increased slightly to accommodate servo loops with non-infinite loop gain and other application considerations. For velocity servo drive applications the best value for Acceleration Feedforward is highly dependent on the drive's speed scaling and servo loop configuration. A value of 100%, in this case, means only that 100% of the commanded acceleration value is applied to the velocity command summing junction and may not be even close to the optimal value. To find the best Acceleration Feedforward Gain, run a simple project that jogs the axis in the positive direction and monitors the Position Error of the axis during the jog. Usually Acceleration Feedforward is used in tandem with Velocity Feedforward to achieve near zero following error during the entire motion profile. To fine tune the Acceleration Feedforward Gain, the Velocity Feedforward Gain must first be optimized using the procedure described above. While capturing the peak Position Error during the acceleration phase of the jog profile, increase the Acceleration Feedforward Gain until the peak Position Error is as small as possible, but still positive. If the peak Position Error during the acceleration ramp is negative, the actual position of the axis is *ahead* of the command position during the acceleration ramp. If this occurs, decrease the Acceleration Feedforward Gain such that the Position Error is again positive. To be thorough the same procedure should be done for the deceleration ramp to verify that the peak Position Error during deceleration is acceptable. Note that reasonable maximum velocity, acceleration, and deceleration values must be entered to jog the axis.

Attribute	Axis Type	Data Type	Access	Description
Acceleration Feedforward Gain (continued)				**AXIS_SERVO_DRIVE** The Acceleration Feedforward Gain attribute is used to provide the Torque Command output necessary to generate the commanded acceleration. It does this by scaling the current Command Acceleration by the Acceleration Feedforward Gain and adding it as an offset to the Servo Output generated by the servo loop. With this done, the servo loops do not need to generate much control effort, hence the Position and/or Velocity Error values are significantly reduced. When used in conjunction with the Velocity Feedforward Gain, the Acceleration Feedforward Gain allows the following error of the servo system during the acceleration and deceleration phases of motion to be reduced to nearly zero. This is important in applications such as electronic gearing and synchronization applications where it is necessary that the actual axis position not significantly lag behind the commanded position at any time. The optimal value for Acceleration Feedforward is 100% theoretically. In reality, however, the value may need to be tweaked to accommodate torque loops with non-infinite loop gain and other application considerations. One thing that may force a smaller Acceleration Feedforward value is that increasing amounts of feedforward tends to exacerbate axis overshoot. When necessary, the Acceleration Feedforward Gain may be 'tweaked' from the 100% value by running a simple user program that jogs the axis in the positive direction and monitors the Position Error of the axis during the jog. Usually Acceleration Feedforward is used in tandem with Velocity Feedforward to achieve near zero following error during the entire motion profile. To fine-tune the Acceleration Feedforward Gain, the Velocity Feedforward Gain must first be optimized using the procedure described above. While capturing the peak Position Error during the acceleration phase of the jog profile, increase the Acceleration Feedforward Gain until the peak Position Error is as small as possible, but still positive. If the peak Position Error during the acceleration ramp is negative, the actual position of the axis is **ahead** of the command position during the acceleration ramp. If this occurs, decrease the Acceleration Feedforward Gain such that the Position Error is again positive. To be thorough the same procedure should be done for the deceleration ramp to verify that the peak Position Error during deceleration is acceptable. Note that reasonable maximum velocity, acceleration, and deceleration values must be entered to jog the axis.
Acceleration Limit Bipolar	AXIS_SERVO_DRIVE	REAL	GSV SSV	Position Units / Sec^2 This attribute maps directly to a SERCOS IDN. See the SERCOS Interface standard for a description. This attribute is automatically set. You usually do not have to change it.
Acceleration Limit Negative	AXIS_SERVO_DRIVE	REAL	GSV SSV	Position Units / Sec^2 This attribute maps directly to a SERCOS IDN. See the SERCOS Interface standard for a description. This attribute is automatically set. You usually do not have to change it.

Attribute	Axis Type	Data Type	Access	Description
Acceleration Limit Positive	AXIS_SERVO_DRIVE	REAL	GSV SSV	Position Units / Sec^2 This attribute maps directly to a SERCOS IDN. See the SERCOS Interface standard for a description. This attribute is automatically set. You usually do not have to change it.
Actual Acceleration	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV Tag	**Important**: To use this attribute, make sure Auto Tag Update is Enabled for the motion group (default setting). Otherwise, you won't see the right value as the axis runs. Actual Acceleration in Position Units / Sec^2 Actual Acceleration is the current instantaneously measured acceleration of an axis, in the configured axis Position Units per second per second. It is calculated as the current increment to the actual velocity per coarse update interval. Actual Acceleration is a signed value — the sign (+ or -) depends on which direction the axis is currently accelerating. Actual Acceleration is a signed floating-point value. Its resolution does not depend on the Averaged Velocity Timebase, but rather on the conversion constant of the axis and the fact that the internal resolution limit on actual velocity is 1 feedback counts per coarse update period per coarse update period.
Actual Position	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV Tag	**Important**: To use this attribute, make sure Auto Tag Update is Enabled for the motion group (default setting). Otherwise, you won't see the right value as the axis runs. Actual Position in Position Units Actual Position is the current absolute position of an axis, in the configured Position Units of that axis, as read from the feedback transducer. Note, however, that this value is based on data reported to the controller as part of an ongoing synchronous data transfer process which results in a delay of one coarse update period. Thus, the Actual Position value that is obtained is the actual position of the axis one coarse update period ago.
Actual Velocity	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV Tag	**Important**: To use this attribute, make sure Auto Tag Update is Enabled for the motion group (default setting). Otherwise, you won't see the right value as the axis runs. Actual Velocity in Position Units / Sec Actual Velocity is the current instantaneously measured speed of an axis, in the configured axis Position Units per second. It is calculated as the current increment to the actual position per coarse update interval. Actual Velocity is a signed value—the sign (+ or -) depends on which direction the axis is currently moving. Actual Velocity is a signed floating-point value. Its resolution does not depend on the Averaged Velocity Timebase, but rather on the conversion constant of the axis and the fact that the internal resolution limit on actual velocity is 1 feedback counts per coarse update.

Attribute	Axis Type	Data Type	Access	Description
Analog Input1 Analog Input 2	AXIS_SERVO_DRIVE	REAL	GSV SSV	This attribute applies only to an axis associated with a Kinetix 7000 drive. This attribute has an integer range +/-16384 representing the analog value of an analog device connected to the Kinetix 7000 drive analog input. These inputs are useful for web/converting applications with load cell (measuring web force on a roller) or dancer (measuring web force/position directly) that can be directly connected to the drive controlling the web.
Attribute Error Code	AXIS_SERVO AXIS_SERVO_DRIVE	INT	GSV* Tag	CIP Error code returned by erred set attribute list service to the module. When an Axis Configuration Fault occurs, one or more axis parameters associated with a motion module or device has not been successfully updated to match the value of the corresponding parameter of the local controller. The fact that the configuration of the axis no longer matches the configuration of the local controller is a serious fault and results in the shutdown of the faulted axis. The Attribute Error Code is reset to zero by reconfiguration of the motion module. Axis Configuration Fault information is passed from the motion module or device to the controller via a 16-bit CIP status word contained in the Set Attribute List service response received by the controller. A Set Attribute List service to the motion module can be initiated by a software Set Attribute List service to the controller, or by an SSV instruction within the controller's program, referencing a servo attribute. Various routines that process responses to motion services are responsible for updating these attributes. The Set and Get service responses provide a status response with each attribute that was processed. That status value is defined by CIP as follows: UINT16, Values 0-255 (0x00-0xFF) are reserved to mirror common service status codes. Values 256 – 65535 are available for object/class attribute specific errors.
Attribute Error ID	AXIS_SERVO AXIS_SERVO_DRIVE	INT	GSV* Tag	Attribute ID associated with non-zero Attribute Error Code. The Attribute Error ID is used to retain the ID of the servo attribute that returned a non-zero attribute error code resulting in an Axis Configuration Fault. The Attribute Error ID defaults to zero and, after a fault has occurred may be reset to zero by reconfiguration of the motion module. To quickly see the Attribute Error in RSLogix 5000 software: 1. Select the axis in the Controller Organizer. 2. Look at the bottom of the Controller Organizer for the Attribute Error.

<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Aux Feedback Configuration</td><td>AXIS_SERVO_DRIVE</td><td>INT</td><td>GSV</td><td>The controller and drive use this for scaling the feedback device counts. These attributes are derived from the corresponding Motor and Auxiliary Feedback Unit attributes.
Bit
0 = Feedback type
• 0 — rotary (default)
• 1 — linear
1 = (reserved)
2 = Linear feedback unit
• 0 — metric
• 1 — english
3 = Feedback Polarity (Aux Only)
• 0 — not inverted
• 1 — inverted
<table>
<tr><th colspan="3">If the bits are</th><th rowspan="2">Then Feedback Resolution is scaled to</th></tr>
<tr><th>2</th><th>1</th><th>0</th></tr>
<tr><td>0</td><td></td><td>0</td><td>Feedback Cycles per Feedback Rev</td></tr>
<tr><td>1</td><td></td><td>0</td><td>Feedback Cycles per Feedback Rev</td></tr>
<tr><td>0</td><td></td><td>1</td><td>Feedback Cycles per mm</td></tr>
<tr><td>1</td><td></td><td>1</td><td>Feedback Cycles per inch</td></tr>
</table>
Feedback Polarity
The Feedback Polarity bit attribute can be used to change the sense of direction of the feedback device. This bit is only valid for auxiliary feedback devices. When performing motor/feedback hookup diagnostics on an auxiliary feedback device using the MRHD and MAHD instructions, the Feedback Polarity bit is configured for the auxiliary feedback device to insure negative feedback into the servo loop. Motor feedback devices must be wired properly for negative feedback since the Feedback Polarity bit is forced to 0, or non-inverted.</td></tr>
<tr><td>Aux Feedback Fault</td><td>AXIS_SERVO
AXIS_SERVO_DRIVE</td><td>BOOL</td><td>Tag</td><td>Set for an auxiliary feedback source when one of these happens:
• The differential electrical signals for one or more of the feedback channels (for example, A+ and A-, B+ and B-, or Z+ and Z-) are at the same level (both high or both low). Under normal operation, the differential signals are always at opposite levels. The most common cause of this situation is a broken wire between the feedback transducer and the servo module or drive;
• Loss of feedback 'power' or feedback 'common' electrical connection between the servo module or drive and the feedback device.
The controller latches this fault. Use a Motion Axis Fault Reset (MAFR) or Motion Axis Shutdown Reset (MASR) instruction to clear the fault.</td></tr>
</table>

Attribute	Axis Type	Data Type	Access	Description
Aux Feedback Interpolation Factor	AXIS_SERVO_DRIVE	DINT	GSV	Feedback Counts per Cycle The Feedback Interpolation attributes establish how many Feedback Counts there are in one Feedback Cycle. The Feedback Interpolation Factor depends on both the feedback device and the drive feedback circuitry. Quadrature encoder feedback devices and the associated drive feedback interface typically support 4x interpolation, so the Interpolation Factor for these devices would be set to 4 Feedback Counts per Cycle (Cycles are sometimes called Lines). High Resolution Sin/Cosine feedback device types can have interpolation factors as high as 2048 Counts per Cycle. The product of the Feedback Resolution and the corresponding Feedback Interpolation Factor is the overall resolution of the feedback channel in Feedback Counts per Feedback Unit. In our example, a Quadrature encoder with a 2000 line/rev resolution and 4x interpolation factor would have an overall resolution of 8000 counts/rev.
Aux Feedback Noise Fault	AXIS_SERVO_DRIVE	BOOL	Tag	Set when there is noise on the feedback device's signal lines. • For example, simultaneous transitions of the feedback A and B channels of an A Quad B is referred to generally as feedback noise. • Feedback noise (shown below) is most often caused by loss of quadrature in the feedback device itself or radiated common-mode noise signals being picked up by the feedback device wiring. You can see both of these on an oscilloscope.  • To troubleshoot the loss of channel quadrature, look for: > physical misalignment of the feedback transducer components > excessive capacitance (or other delays) on the encoder signals • Proper grounding and shielding usually cures radiated noise problems. The controller latches this fault. Use a Motion Axis Fault Reset (MAFR) or Motion Axis Shutdown Reset (MASR) instruction to clear the fault.

Attribute	Axis Type	Data Type	Access	Description
Aux Feedback Ratio	AXIS_SERVO_DRIVE	FLOAT	GSV	Aux Feedback Units per Motor Feedback Unit The Aux Feedback Ratio attribute represents the quantitative relationship between auxiliary feedback device and the motor. For a rotary auxiliary feedback device, this attribute's value should be the turns ratio between the auxiliary feedback device and the motor shaft. For linear auxiliary feedback devices, this attribute value would typically represent the feed constant between the motor shaft and the linear actuator. The Aux Feedback Ratio attribute is used in calculating range limits and default value calculations during configuration based on the selected motor's specifications. The value is also used by the drive when running the dual feedback servo loop configuration.
Aux Feedback Resolution	AXIS_SERVO_DRIVE	DINT	GSV	Cycles per Aux Feedback Unit The Motor and Aux Feedback Resolution attributes are used to provide the A-B drive with the resolution of the associated feedback device in cycles per feedback unit. These parameters provide the SERCOS drive with critical information needed to compute scaling factors used to convert Drive Counts to Feedback counts.

<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Aux Feedback Type</td><td>AXIS_SERVO_DRIVE</td><td>INT</td><td>GSV</td><td>The Motor and Aux Feedback Type attributes are used to identify the motor mounted or auxiliary feedback device connected to the drive.
<table>
<tr><th>Feedback Type</th><th>Code</th><th>Rotary Only</th><th>Linear Only</th><th>Rotary or Linear</th></tr>
<tr><td><None></td><td>0x0000</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>SRS</td><td>0x0001</td><td>X</td><td></td><td></td></tr>
<tr><td>SRM</td><td>0x0002</td><td>X</td><td></td><td></td></tr>
<tr><td>SCS</td><td>0x0003</td><td>X</td><td></td><td></td></tr>
<tr><td>SCM</td><td>0x0004</td><td>X</td><td></td><td></td></tr>
<tr><td>SNS</td><td>0x0005</td><td>X</td><td></td><td></td></tr>
<tr><td>MHG</td><td>0x0006</td><td>X</td><td></td><td></td></tr>
<tr><td>Resolver</td><td>0x0007</td><td>X</td><td></td><td></td></tr>
<tr><td>Analog Reference</td><td>0x0008</td><td>X</td><td></td><td></td></tr>
<tr><td>Sin/Cos</td><td>0x0009</td><td></td><td></td><td>X</td></tr>
<tr><td>TTL</td><td>0x000A</td><td></td><td></td><td>X</td></tr>
<tr><td>UVW</td><td>0x000B</td><td></td><td></td><td>X</td></tr>
<tr><td>Unknown Stegmann</td><td>0x000C</td><td></td><td></td><td>X</td></tr>
<tr><td>Endat</td><td>0x000D</td><td></td><td></td><td>X</td></tr>
<tr><td>RCM21S-4</td><td>0x000E</td><td>X</td><td></td><td></td></tr>
<tr><td>RCM21S-6</td><td>0x000F</td><td>X</td><td></td><td></td></tr>
<tr><td>RCM21S-8</td><td>0x0010</td><td>X</td><td></td><td></td></tr>
<tr><td>LINCODER</td><td>0x0011</td><td></td><td>X</td><td></td></tr>
<tr><td>Sin/Cos with Hall</td><td>0x0012</td><td></td><td></td><td>X</td></tr>
<tr><td>TTL with Hall</td><td>0x0013</td><td></td><td></td><td>X</td></tr>
</table></td></tr>
<tr><td>Aux Feedback Units</td><td>AXIS_SERVO_DRIVE</td><td>INT</td><td>GSV</td><td>The Motor Feedback Units attribute establishes the unit of measure that is applied to the Motor Feedback Resolution attribute value. The Aux Feedback Units attribute establishes the unit of measure that is applied to the Aux Feedback Resolution attribute value. Units appearing in the enumerated list cover linear or rotary, english or metric feedback devices.
0 = revs
1 = inches
2 = mm</td></tr>
<tr><td>Aux Position Feedback</td><td>AXIS_SERVO
AXIS_SERVO_DRIVE</td><td>REAL</td><td>GSV
Tag</td><td>**Important**: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won’t see the right value as the axis runs. See Axis Info Select 1.
Auxiliary Position Feedback in Position Units
Aux Position Feedback is the current value of the position feedback coming from the auxiliary feedback input.</td></tr>
</table>

Attribute	Axis Type	Data Type	Access	Description
Average Velocity	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV Tag	**Important**: To use this attribute, make sure Auto Tag Update is Enabled for the motion group (default setting). Otherwise, you won't see the right value as the axis runs. Average Velocity in Position Units / Seconds Average Velocity is the current speed of an axis in the configured Position Units per second of the axis. Unlike the Actual Velocity attribute value, it is calculated by averaging the actual velocity of the axis over the configured Average Velocity Timebase for that axis. Average velocity is a signed value. The sign doesn't necessarily show the direction that the axis is currently moving. It shows the direction the average move is going. The axis may be currently moving in the opposite direction. The resolution of the Average Velocity variable is determined by the current value of the Averaged Velocity Timebase parameter and the configured Conversion Constant (feedback counts per Position Unit) for the axis. • The greater the Average Velocity Timebase value, the better the speed resolution but the slower the response to changes in speed. • The minimum Average Velocity Timebase value is the Coarse Update period of the motion group. The Average Velocity resolution in Position Units per second may be calculated using the equation below. $\text{Average Velocity Timebase [Seconds] x } K \left[\frac{\text{Feedback Counts}}{\text{Position Unit}}\right]$ For example, on an axis with position units of inches and a conversion constant (K) of 20000, an averaged velocity time-base of 0.25 seconds results in an average velocity resolution of: $\frac{1}{0.25 \times 20000} = 0.0002 \frac{\text{Inches}}{\text{Second}} = 0.012 \frac{\text{Inches}}{\text{Minute}}$
Average Velocity Timebase	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV SSV	Seconds The Average Velocity Timebase attribute is used to specify the desired time in seconds to be used for calculating the Average Velocity of the axis. When the Average Velocity Value is requested, the value is computed by taking the total distance traveled by the axis in the amount of time given by the Average Velocity Timebase and dividing this value by the timebase. The Average Velocity Timebase value should be large enough to filter out the small changes in velocity which would otherwise result in a 'noisy' velocity value, but small enough to track significant changes in axis velocity. Typically, a value between 0.25 and 0.5 seconds works well for most applications

Attribute	Axis Type	Data Type	Access	Description
Axis Configuration State	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	SINT	GSV	State of the axis configuration state machine The Axis Configuration State attribute is used for debugging to indicate where in the axis configuration state-machine this axis presently is. Even consumed and virtual axes will utilize this attribute. If the attribute is: • 128 — the axis is configured and ready for use. • Not 128 — the axis isn't configured.
Axis Control Bits	AXIS_SERVO AXIS_SERVO_DRIVE	DINT	GSV	Bits 0 = Abort Process Request 1 = Shutdown Request 2 = Zero DAC Request 3 = Abort Home Request 4 = Abort Event Request 5-14 = Reserved 15 = Change Cmd Reference **Abort Process** If this bit is set, any active tuning or test process on the axis is aborted. **Shutdown Request** If this bit is set, the axis is forced into the shutdown state. For an AXIS_SERVO data type, the OK contact opens and the DAC output goes to 0. **Zero DAC Request — Only for AXIS_SERVO Data Type** If this bit is set, the servo module forces the DAC output for the axis to zero volts. This bit only has an affect if the axis is in the Direct Drive State with the drive enabled but no servo action. **Abort Home Request** If this bit is set, any active homing procedures are cancelled. **Abort Event Request** If this bit is set, any active registration or watch event procedures are cancelled. **Change Cmd Reference** If this bit is set, the controller switches to a new position coordinate system for command position. The servo module or drive uses this bit when processing new command position data from the controller to account for the offset implied by the shift in the reference point. The bit is cleared when the axis acknowledges completion of the reference position change by clearing its Change Position Reference bit.

Attribute	Axis Type	Data Type	Access	Description
Axis Data Type	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	SINT	MSG	Associated motion axis tag data type: 0 = Feedback 1 = Consumed 2 = Virtual 3 = Generic 4 = Servo 5 = Servo Drive 6 = Generic Drive The Axis Data Type attribute and is used to determine which data template, memory format, and set of attributes are created and applicable for this axis instance. This attribute can only be set as part of an axis create service. **Feedback** A feedback-only axis associated with feedback-only modules like PLS II and CFE, supporting quadrature encoder, resolver, HiperFace, and so on. **Consumed** A consumed axis which consumes axis motion data produced by a motion axis on another controller. **Virtual** A virtual axis having full motion planner operation but not associated with any physical device. **Generic** An axis with full motion planner functionality but no integrated configuration support; associated with devices such as DriveLogix, 1756-DM. **Servo** An axis with full motion planner functionality and integrated configuration support; associated with modules closing a servo loop and sending an analog command to an external drive; that is, 1756-M02AE, 1756-HYD02, and 1756-M02AS modules. **Servo Drive** An axis with full motion planner functionality and integrated configuration support; associated with digital drive interface modules sending a digital command to the external drive; that is, 1756-M03SE, 1756-M08SE, and 17556-M16SE (SERCOS interface). **Generic Drive** An axis of a SERCOS interface drive that is Extended Pack Profile compliant and on the ring of a 1756-M08SEG module.

<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Axis Event</td><td>AXIS_CONSUMED
AXIS_GENERIC
AXIS_SERVO
AXIS_SERVO_DRIVE
AXIS_VIRTUAL</td><td>DINT</td><td>Tag</td><td>Lets you access all the event status bits in one 32-bit word. This tag is the same as the Axis Event Bits attribute.
<table>
<tr><th>Event Status</th><th>Bit</th></tr>
<tr><td>Watch Event Armed Status</td><td>0</td></tr>
<tr><td>Watch Event Status</td><td>1</td></tr>
<tr><td>Reg Event 1 Armed Status</td><td>2</td></tr>
<tr><td>Reg Event 1 Status</td><td>3</td></tr>
<tr><td>Reg Event 2 Armed Status</td><td>4</td></tr>
<tr><td>Reg Event 2 Status</td><td>5</td></tr>
<tr><td>Home Event Armed Status</td><td>6</td></tr>
<tr><td>Home Event Status</td><td>7</td></tr>
</table></td></tr>
<tr><td>Axis Event Bits</td><td>AXIS_CONSUMED
AXIS_GENERIC
AXIS_SERVO
AXIS_SERVO_DRIVE
AXIS_VIRTUAL</td><td>DINT</td><td>GSV</td><td>Lets you access all the event status bits in one 32-bit word. This attribute is the same as the Axis Event tag.
<table>
<tr><th>Event Status</th><th>Bit</th></tr>
<tr><td>Watch Event Armed Status</td><td>0</td></tr>
<tr><td>Watch Event Status</td><td>1</td></tr>
<tr><td>Reg Event 1 Armed Status</td><td>2</td></tr>
<tr><td>Reg Event 1 Status</td><td>3</td></tr>
<tr><td>Reg Event 2 Armed Status</td><td>4</td></tr>
<tr><td>Reg Event 2 Status</td><td>5</td></tr>
<tr><td>Home Event Armed Status</td><td>6</td></tr>
<tr><td>Home Event Status</td><td>7</td></tr>
</table></td></tr>
<tr><td>Axis Fault</td><td>AXIS_CONSUMED
AXIS_GENERIC
AXIS_SERVO
AXIS_SERVO_DRIVE
AXIS_VIRTUAL</td><td>DINT</td><td>Tag</td><td>The axis faults for your axis:
<table>
<tr><th>Type of Fault</th><th>Bit</th></tr>
<tr><td>Physical Axis Fault</td><td>0</td></tr>
<tr><td>Module Fault</td><td>1</td></tr>
<tr><td>Config Fault</td><td>2</td></tr>
</table>
This attribute is the same as the Axis Fault Bits attribute.</td></tr>
<tr><td>Axis Fault Bits</td><td>AXIS_CONSUMED
AXIS_GENERIC
AXIS_SERVO
AXIS_SERVO_DRIVE
AXIS_VIRTUAL</td><td>DINT</td><td>GSV*</td><td>The axis faults for your axis:
<table>
<tr><th>Type of Fault</th><th>Bit</th></tr>
<tr><td>Physical Axis Fault</td><td>0</td></tr>
<tr><td>Module Fault</td><td>1</td></tr>
<tr><td>Config Fault</td><td>2</td></tr>
</table>
This attribute is the same as the Axis Fault tag.</td></tr>
</table>

Attribute	Axis Type	Data Type	Access	Description
Axis Info Select 1 Axis Info Select 2	AXIS_SERVO AXIS_SERVO_DRIVE	DINT	GSV SSV	An axis has a group of attributes that do not get updated by default. • To use one of them, you must choose it for Real Time Axis Information for the axis. Otherwise, its value won't change and you won't see the right value as the axis runs. • You can choose up to 2 of these attributes. To use a GSV instruction to choose an attribute for Real Time Axis Information, set the Axis Info Select 1 or Axis Info Select 2 attribute.

AXIS_SERVO	AXIS_SERVO_DRIVE	Value
None (default)	None (default)	0
Position Command	Position Command	1
Position Feedback	Position Feedback	2
Aux Position Feedback	Aux Position Feedback	3
Position Error	Position Error	4
Position Integrator Error	Position Integrator Error	5
Velocity Command	Velocity Command	6
Velocity Feedback	Velocity Feedback	7
Velocity Error	Velocity Error	8
Velocity Integrator Error	Velocity Integrator Error	9
Acceleration Command	Acceleration Command	10
Acceleration Feedback	Acceleration Feedback	11
Servo Output Level		12
Marker Distance	Marker Distance	13
	Torque Command	14
	Torque Feedback	15
	Positive Dynamic Torque Limit	16
	Negative Dynamic Torque Limit	17
	Motor Capacity	18
	Drive Capacity	19
	Power Capacity	20
	Bus Regulator Capacity	21
	Motor Electrical Angle	22
	Torque Limit Source	23
	DC Bus Voltage	24
Absolute Offset		25
		26
		27
	Guard Status	28
	Guard Faults	29

Attribute	Axis Type	Data Type	Access	Description
Axis Instance	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	INT	GSV	Instance Number assigned to Axis The Axis Instance attribute is used to return the instance number of an axis. Major fault records generated for an axis major fault contains only the instance of the offending axis. This attribute would then typically be used by a user to determine if this was the offending axis; that is, if the instance number matches.
Axis Response Bits	AXIS_SERVO AXIS_SERVO_DRIVE	DINT	GSV	Bits 0 = Abort Process Acknowledge 1 = Shutdown Acknowledge 2 = Zero DAC Acknowledge 3 = Abort Home Acknowledge 4 = Abort Event Acknowledge 5…14 = Reserved 15 = Change Pos Reference **Abort Process Acknowledge** If this bit is set, the tuning or test process has been aborted. **Shutdown Acknowledge** If this bit is set, the axis has been forced into the shutdown state. **Zero DAC Acknowledge — Only for AXIS_SERVO Data Type** If this bit is set, the DAC output for the axis has been set to zero volts. **Abort Home Acknowledge** If this bit is set, the active home procedure has been aborted. **Abort Event Acknowledge** If this bit is set, the active registration or watch position event procedure has been aborted. **Change Pos Reference** If this bit is set, the Servo loop has switched to a new position coordinate system. The controller uses this bit when processing new position data from the servo module or drive to account for the offset implied by the shift in the reference point. The bit is cleared when the controller acknowledges completion of the reference position change by clearing its Change Cmd Reference bit.

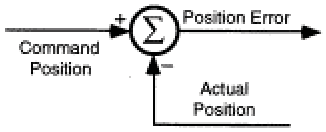
<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Axis State</td><td>AXIS_CONSUMED
AXIS_GENERIC
AXIS_SERVO
AXIS_SERVO_DRIVE
AXIS_VIRTUAL</td><td>SINT</td><td>GSV</td><td>Operating state of the axis.
0 = Axis Ready
1 = Direct Drive Control
2 = Servo Control
3 = Axis Faulted
4 = Axis Shutdown
5 = Axis Inhibited
6 = Axis Ungrouped
7 = No Module
8 = Configuring</td></tr>
<tr><td>Axis Status</td><td>AXIS_CONSUMED
AXIS_GENERIC
AXIS_SERVO
AXIS_SERVO_DRIVE
AXIS_VIRTUAL</td><td>DINT</td><td>Tag</td><td>Lets you access all the axis status bits in one 32-bit word. This tag is the same as the Axis Status Bits attribute.
<table>
<tr><th>Axis Status</th><th>Bit</th></tr>
<tr><td>Servo Action Status</td><td>0</td></tr>
<tr><td>Drive Enable Status</td><td>1</td></tr>
<tr><td>Shutdown Status</td><td>2</td></tr>
<tr><td>Config Update In Process</td><td>3</td></tr>
<tr><td>Inhibit Status</td><td>4</td></tr>
</table></td></tr>
<tr><td>Axis Status Bits</td><td>AXIS_CONSUMED
AXIS_GENERIC
AXIS_SERVO
AXIS_SERVO_DRIVE
AXIS_VIRTUAL</td><td>DINT</td><td>GSV*</td><td>Lets you access all the axis status bits in one 32-bit word. This attribute is the same as the Axis Status tag.
<table>
<tr><th>Axis Status</th><th>Bit</th></tr>
<tr><td>Servo Action Status</td><td>0</td></tr>
<tr><td>Drive Enable Status</td><td>1</td></tr>
<tr><td>Shutdown Status</td><td>2</td></tr>
<tr><td>Config Update In Process</td><td>3</td></tr>
<tr><td>Inhibit Status</td><td>4</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Axis Type</td><td>AXIS_GENERIC
AXIS_SERVO
AXIS_SERVO_DRIVE</td><td>INT</td><td>GSV
SSV</td><td>The Axis Type attribute is used to establish the intended use of the axis.
<table>
<tr><th>If</th><th>Then set the attribute to</th></tr>
<tr><td>The axis is unused in the application, which is a common occurrence when there are an odd number of axes in the system</td><td>0</td></tr>
<tr><td>You only want the position information from the feedback interface</td><td>1</td></tr>
<tr><td>The axis is intended for full servo operation</td><td>2</td></tr>
</table>
<p>Axis Type is not only used to qualify many operations associated with the axis servo loop, it also controls the behavior of the servo module's Axis Status indicators. An Axis Type of '1' (Feedback Only) results in the DRIVE LED being blanked, while a value of '0' (Unused) blanks both the FDBK and DRIVE status indicators.</p>
<p>This controller attribute is also replicated in the motion module.</p>
<p>RSLogix 5000 software also uses the current configured value for Axis Type to control the look of many of the dialog boxes associated with configurating an axis.</p></td></tr>
<tr><td>Backlash Reversal Offset</td><td>AXIS_SERVO
AXIS_SERVO_DRIVE</td><td>REAL</td><td>GSV
SSV</td><td><p>Backlash Reversal Offset provides the user the capability to compensate for positional inaccuracy introduced by mechanical backlash. For example, power-train type applications require a high level of accuracy and repeatability during machining operations. Axis motion is often generated by a number of mechanical components such as a motor, a gearbox, and a ball-screw, which can introduce inaccuracies and which are subject to wear over their lifetime. Hence, when an axis is commanded to reverse direction, mechanical play in the machine (through the gearing, ball-screw, and so on.) may result in a small amount of motor motion without axis motion. As a result, the feedback device may indicate movement even though the axis has not physically moved.</p>
<p>Compensation for mechanical backlash can be achieved by adding a directional offset, specified by the Backlash Reversal Offset attribute, to the motion planner's command position as it is applied to the associated servo loop. Whenever the commanded velocity changes sign (a reversal), the Logix controller adds, or subtracts, the Backlash Distance value from the current commanded position. This causes the servo to immediately move the motor to the other side of the backlash window and engage the load. It is important to note that the application of this directional offset is completely transparent to the user; the offset does not have any affect on the value of the Command Position attribute.</p>
<p>If a value of zero is applied to the Backlash Reversal Offset, the feature is effectively disabled. Once enabled by a non-zero value, and the load is engaged by a reversal of the commanded motion, changing the Backlash Reversal Offset can cause the axis to shift as the offset correction is applied to the command position..</p></td></tr>
</table>

Attribute	Axis Type	Data Type	Access	Description
Backlash Stabilization Window	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV	The Backlash Stabilization Window attribute is used to control the Backlash Stabilization feature in the servo control loop. What follows is a description of this feature and the general backlash instability phenomenon. Mechanical backlash is a common problem in applications that utilize mechanical gearboxes. The problem stems from the fact that until the input gear is turned to the point where its proximal tooth contacts an adjacent tooth of the output gear, the reflected inertia of the output is not felt at the motor. In other words, when the gear teeth are not engaged, the system inertia is reduced to the motor inertia. If the servo loop is tuned for peak performance with the load applied, the axis is at best under-damped and at worst unstable in the condition where the gear teeth are not engaged. In the worst case scenario, the motor axis and the input gear oscillates wildly between the limits imposed by the output gear teeth. The net effect is a loud buzzing sound when the axis is at rest. If this situation persists the gearbox wears out prematurely. To prevent this condition, the conventional approach is to de-tune the servo so that the axis is stable without the gearbox load applied. Unfortunately, system performance suffers.' Due to its non-linear, discontinuous nature, adaptive tuning algorithms generally fall short of addressing the backlash problem. However, a very effective backlash compensation algorithm can be demonstrated using the Torque Scaling gain. The key to this algorithm is the tapered Torque Scaling profile as a function of the position error of the servo loop. The reason for the tapered profile, as opposed to a step profile, is that when the position error exceeds the backlash distance a step profile would create a very large discontinuity in the torque output. This repulsing torque tends to slam the axis back against the opposite gear tooth and perpetuate the buzzing effect. The tapered Torque Scaling profile is only run when the acceleration command to the servo loop is zero, that is, when we are not commanding any acceleration or deceleration that would engage the teeth of the gearbox. Properly configured with a suitable value for the Backlash Stabilization Window, this algorithm entirely eliminates the gearbox buzz without sacrificing any servo performance. The Backlash Stabilization parameter determines the width of the window over which backlash stabilization is applied. In general, this value should be set to the measured backlash distance. A Backlash Stabilization Window value of zero effectively disables the feature. (Patent Pending)

Attribute	Axis Type	Data Type	Access	Description
Brake Engage Delay Time	AXIS_SERVO_DRIVE	REAL	GSV SSV	Seconds The Brake Engage Delay attribute controls the amount of time that the drive continues to apply torque to the motor after the motor brake output is changed to engage the brake. This gives time for the motor brake to engage. This is the sequence of events associated with engaging the motor brake. • Disable axis is initiated (via MSF or drive disable fault action) • Drive stops tracking command reference. (Servo Action Status bit clears.) • Decel to zero speed using configured Stopping Torque. • Zero speed or Stopping Time Limit is reached. • Turn motor brake output off to engage the motor brake. • Wait Brake Engage Delay Time. • Disable the drive power structure. (Drive Enable Status bit clears.) If the axis is shutdown through either a fault action or motion instruction the drive power structure is disabled immediately and the motor brake is engaged immediately. • Drive stops tracking command reference. (Servo Action Status bit clears.) • Disable drive power structure, (Drive Enable Status bit clears.) • Turn off brake output to engage brake.
Brake Release Delay Time	AXIS_SERVO_DRIVE	REAL	GSV SSV	Seconds The Brake Release Delay attribute controls the amount of time that the drive holds off tracking command reference changes after the brake output is changed to release the brake. This gives time for the brake to release. This is the sequence of events associated with engaging the brake. • Enable axis is initiated (via MSO or MAH) • Drive power structure enabled. (Drive Enable Status bit sets.) • Turn motor brake output on to release the brake.** • Wait Brake Release Delay Time. • Track command reference. (Servo_Action_Status bit sets) **The drive does not release the brake unless there is holding torque.
Bus Ready Status	AXIS_SERVO_DRIVE	BOOL	Tag	If the bit is: • ON — The voltage of the drive's dc bus is high enough for operation. • OFF — The voltage of the drive's dc bus is too low.

Attribute	Axis Type	Data Type	Access	Description
Bus Regulator Capacity	AXIS_SERVO_DRIVE	REAL	GSV Tag	**Important**: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won't see the right value as the axis runs. See Axis Info Select 1. The present utilization of the axis bus regulator as a percent of rated capacity.
Bus Regulator ID	AXIS_SERVO_DRIVE	INT	GSV	The Bus Regulator ID attribute contains the enumeration of the specific A-B Bus Regulator or System Shunt catalog numbers associated with the axis. If the Bus Regulator ID does not match that of the actual bus regulator or shunt hardware, an error is generated during the drive configuration process.
C2C Connection Instance	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	SINT	GSV	Producer/Consumed axis's associated C2C connection instance in reference to the C2C map instance When Axis Data Type is specified to be 'Consumed' then this axis is associated to the consumed data by specifying both the C2C Map Instance and the C2C Connection Instance. This attribute is the connection instance under the C2C map instance, which provides the axis data being sent to it from another axis via a C2C connection. For all other Axis Data Types if this axis is to be produced then this attribute is set to the connection instance under the local controller's map instance (1) that is used to send the remote axis data via the C2C connection.
C2C Map Instance	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	SINT	GSV	Producer/Consumed axis's associated C2C map instance When the Axis Data Type attribute is specified to be 'Consumed' then this axis is associated to the consumed data by specifying both the C2C Map Instance and the C2C Connection Instance. For all other Axis Data Types if this axis is to be produced then this attribute is set to 1 (one) to indicate that the connection is off of the local controller's map instance.
Command Acceleration	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV Tag	**Important**: To use this attribute, make sure Auto Tag Update is Enabled for the motion group (default setting). Otherwise, you won't see the right value as the axis runs. Command Acceleration in Position Units / Sec^2 Command Acceleration is the commanded speed of an axis, in the configured axis Position Units per second per second, as generated by any previous motion instructions. It is calculated as the current increment to the command velocity per coarse update interval. Command Acceleration is a signed value—the sign (+ or -) depends on which direction the axis is being commanded to move. Command Acceleration is a signed floating-point value. Its resolution does not depend on the Averaged Velocity Timebase, but rather on the conversion constant of the axis and the fact that the internal resolution limit on command velocity is 0.00001 feedback counts per coarse update period per coarse update period.

Attribute	Axis Type	Data Type	Access	Description
Command Position	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV Tag	**Important**: To use this attribute, make sure Auto Tag Update is Enabled for the motion group (default setting). Otherwise, you won't see the right value as the axis runs. Command Position in Position Units Command Position is the desired or commanded position of a physical axis, in the configured Position Units of that axis, as generated by the controller in response to any previous motion Position Control instruction. Command Position data is transferred by the controller to a physical axis as part of an ongoing synchronous data transfer process which results in a delay of one coarse update period. Thus, the Command Position value that is obtained is the command position that is acted upon by the physical servo axis one coarse update period from now. The figure below shows the relationship between Actual Position, Command Position, and Position Error for an axis with an active servo loop. Actual Position is the current position of the axis as measured by the feedback device (for example, encoder). Position error is the difference between the Command and Actual Positions of the servo loop, and is used to drive the motor to make the actual position equal to the command position.  Command position is useful when performing motion calculations and incremental moves based on the current position of the axis while the axis is moving. Using command position rather than actual position avoids the introduction of cumulative errors due to the position error of the axis at the time the calculation is performed.
Command Velocity	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV Tag	**Important**: To use this attribute, make sure Auto Tag Update is Enabled for the motion group (default setting). Otherwise, you won't see the right value as the axis runs. Command Velocity in Position Units / Seconds Command Velocity is the commanded speed of an axis, in the configured axis Position Units per second, as generated by any previous motion instructions. It is calculated as the current increment to the command position per coarse update interval. Command Velocity is a signed value—the sign (+ or -) depends on which direction the axis is being commanded to move. Command Velocity is a signed floating-point value. Its resolution does not depend on the Averaged Velocity Timebase, but rather on the conversion constant of the axis and the fact that the internal resolution limit on command velocity is 0.00001 feedback counts per coarse update.
Common Bus Fault	AXIS_SERVO_DRIVE	BOOL	Tag	The drive shuts down if you give it 3-phase power while it's configured for Common Bus Follower mode. If that happens, this bit turns on.

Attribute	Axis Type	Data Type	Access	Description
Commutation Fault	AXIS_SERVO_DRIVE	DINT	BOOL	Set when the commutation feedback source associated with the drive axis has a problem that prevents the drive from receiving accurate or reliable motor shaft information to perform commutation.
Config Fault	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set when an update operation targeting an axis configuration attribute of an associated motion module has failed. Specific information concerning the Configuration Fault may be found in the Attribute Error Code and Attribute Error ID attributes associated with the motion module. Do you want this fault to give the controller a major fault? • YES — Set the General Fault Type of the motion group = Major Fault. • NO — You must write code to handle these faults.
Config Update In Process	AXIS_CONSUMED AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	When you use an SSV instruction to change an attribute, the controller sends the change to the motion module. If you want to wait until the change is done, monitor the ConfigUpdateInProcess bit of the axis. If the bit is: • ON — The controller is changing the attribute. • OFF — The change is done.
Continuous Torque Limit	AXIS_SERVO_DRIVE	REAL	GSV SSV	%Rated The Torque limit attribute provides a method for controlling the continuous torque limit imposed by the drive's thermal model of the motor. Increasing the Continuous Torque Limit increases the amount of continuous motor torque allowed before the drive either folds back the motor current or the drive declares a motor thermal fault. Motors equipped with special cooling options can be configured with a Continuous Torque Limit of greater than 100% rated to attain higher continuous torque output from the motor. Motors operating in high ambient temperature conditions can be configured with a Continuous Torque Limit of less than 100% rated torque to protect the motor from overheating. The Continuous Torque Limit specifies the maximum percentage of the motor's rated current that the drive can command on a continuous or RMS basis. For example, a Continuous Torque Limit of 150% limits the continuous current delivered to the motor to 1.5 times the continuous current rating of the motor.
Control Sync Fault	AXIS_CONSUMED AXIS_SERVO AXIS_SERVO_DRIVE	BOOL	Tag	If this bit is set, the controller lost communication with the motion module and missed several position updates in a row. • The controller can miss up to 4 position updates. After that, the Control Sync Fault bit is set. The motion module may fault later or may already be faulted. • For a consumed axis, this bit means that communication is lost with the producing controller. This bit clears when communication is reestablished.

Attribute	Axis Type	Data Type	Access	Description
Controlled By Transform Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	If the bit is: • ON — A transform is moving the axis. • OFF — A transform isn't moving the axis.
Conversion Constant	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV SSV	Counts / Position Unit Range = 0.1 - 1e^{12} Axis_Servo Default = 8000 Axis_Servo_Drive Default = 2000000 To allow axis position to be displayed and motion to be programmed in the position units specified by the Position Unit string attribute, a Conversion Constant must be established for each axis. The Conversion Constant, sometimes known as the K constant, allows the Axis Object to convert the axis position units into feedback counts and vice versa. Specifically, K is the number of feedback counts per Position Unit. The 1756-M02AE encoder based servo module uses 4X encoder feedback decoding (both edges of channel A and B are counted). The count direction is determined from both the direction of the edge and the state of the opposite channel. Channel A leads channel B for increasing count. This is the most commonly used decode mode with incremental encoders, since it provides the highest resolution. For example, suppose this servo axis utilizes a 1000 line encoder in a motor coupled directly to a 5 pitch lead screw (5 turns per inch). With a user defined Position Unit of Inches, the conversion constant is calculated as shown below: K = 1000 Lines/Rev * 4 Counts/Line * 5 Revs/Inch = 20,000 Counts/Inch. **Attention**: If 'Conversion Constant' is changed it invalidates all of the attributes you can set with 'Position Unit' conversions in 'Description' column. To be valid the 'Conversion Constant' must be set to the desired value prior to setting (including defaulting) any of the affected attributes.

Attribute	Axis Type	Data Type	Access	Description
Coordinated Motion Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	DINT	GSV Tag	Set if any coordinated motion profile is currently active upon the axis. It is cleared as soon as Coordinated Motion is complete or stopped.
Damping Factor	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV	The Damping Factor attribute value is used in calculating the maximum Position Servo Bandwidth (see below) during execution of the MRAT (Motion Run Axis Tune) instruction. In general the Damping Factor attribute controls the dynamic response of the servo axis. When gains are tuned using a small damping factor (like 0.7), a step response test performed on the axis would demonstrate under-damped behavior with velocity overshoot. A gain set generated using a larger damping factor, like 1.0, would produce a system step response that has no overshoot but has a significantly lower servo bandwidth. The default value for the Damping Factor of 0.8 should work fine for most applications.
DC Bus Voltage	AXIS_SERVO_DRIVE	DINT	GSV Tag	**Important**: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won't see the right value as the axis runs. See Axis Info Select 1. Volts This parameter is the present voltage on the DC Bus of the drive.
Decel Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set if the axis is currently being commanded to decelerate. Use the Accel Status bit and the Decel Status bit to see if the axis is accelerating or decelerating. If both bits are off, then the axis is moving at a steady speed or is at rest.

Coordinated Motion Status bits:

Motion Status	Bit
Accel Status	0
Decel Status	1
Actual Pos Tolerance Status	2
Command Pos Tolerance Status	3
Stopping Status	4
Reserved	5
Move Status	6
Transition Status	7
Move Pending Status	8
Move Pending Queue Full Status	9
Reserved	10
Reserved	11
Reserved	12
Coordinate System in a Source CS	13
Coordinate System in a Target CS	14

Attribute	Axis Type	Data Type	Access	Description
Direct Drive Ramp Rate	AXIS_SERVO	REAL	GSV SSV	Volts/Second The Direct Drive Ramp Rate attribute contains a slew rate for changing the output voltage when the Direct Drive On (MDO) instruction is executed. A Direct Drive Ramp Rate of 0, disables the output ramp rate limiter, allowing the Direct Drive On voltage to be applied directly.
Directional Scaling Ratio	AXIS_SERVO	REAL	GSV SSV	In some cases, the speed or velocity scaling of the external drive actuator may be directionally dependent. This non-linearity can be substantial in hydraulic applications. To compensate for this behavior, the Directional Scaling Ratio attribute can be applied to the Velocity Scaling based on the sign of the Servo Output. Specifically, the Velocity Scaling value is scaled by the Directional Scaling Ratio when the sign of the Servo Output is positive. Thus, the Directional Scaling Ratio is the ratio of the Velocity Scaling in the positive direction (positive servo output) to the Velocity Scaling in the negative direction (negative servo output). The value for the Directional Scaling ratio can be empirically determined by running the auto-tune procedure in the positive direction and then in the negative direction and calculating the ratio of the resulting Velocity/Torque Scaling values.
Drive Axis ID	AXIS_SERVO_DRIVE	INT	GSV	Product Code of Drive Amplifier The Drive ID attribute contains the ASA Product Code of the drive amplifier associated with the axis. If the Product Code does not match that of the actual drive amplifier, an error is generated during the configuration process.
Drive Capacity	AXIS_SERVO_DRIVE	REAL	GSV Tag	**Important**: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won't see the right value as the axis runs. See Axis Info Select 1. The present utilization of drive capacity as a percent of rated capacity.
Drive Control Voltage Fault	AXIS_SERVO_DRIVE	BOOL	Tag	Set when the power supply voltages associated with the drive circuitry fall outside of acceptable limits.
Drive Cooling Fault	AXIS_SERVO_DRIVE	BOOL	Tag	Set when the ambient temperature surrounding the drive's control circuitry temperature exceeds the drive ambient shut-down temperature.

Attribute	Axis Type	Data Type	Access	Description
Drive Enable Input Fault	AXIS_SERVO_DRIVE	BOOL	Tag	This fault would be declared if either one of two possible conditions occur: 1) If an attempt is made to enable the axis (typically via MSO or MAH instruction) while the Drive Enable Input is inactive. 2) If the Drive Enable Input transitions from active to inactive while the axis is enabled. This fault can only occur when the Drive Enable Input Fault Handling bit is set in the Fault Configuration Bits attribute. If the Drive Enable Input Fault Action is set for Stop Command and the axis is stopped as a result of a Drive Enable Input Fault, the faulted axis cannot be moved until the fault is cleared. Any attempt to move the axis in the faulted state using a motion instruction results in an instruction error. If the Drive Enable Fault Action setting is Status Only or Stop Command and an attempt is made to enable the axis (typically via MSO or MAH instruction) while the Drive Enable Input is active, the axis enables in the faulted state indicating a Drive Enable Input Fault. When the Drive Enable Fault Action setting is Stop Command, instructions that both enable the axis and initiate motion (MAH, MRAT, MAHD) abort the motion process leaving the instruction with both the IP and PC bits clear. This fault condition is latched and requires execution of an explicit MAFR (Motion Axis Fault Reset) or MASR (Motion Axis Shutdown Reset) instruction to clear. Any attempt to clear the fault while the drive enable input is still inactive and the drive is enabled is unsuccessful. However, the drive enable input fault may be cleared with the drive enable input inactive if the drive is disabled. If the Drive Enable Input Checking bit is clear, then the state of the Drive Enable Input is irrelevant so no fault would be declared in any of the above conditions.
Drive Enable Input Fault Action	AXIS_SERVO_DRIVE	SINT	GSV SSV	<table><tr><th>Fault Action</th><th>Value</th></tr><tr><td>Shutdown</td><td>0</td></tr><tr><td>Disable Drive</td><td>1</td></tr><tr><td>Stop Motion</td><td>2</td></tr><tr><td>Status Only</td><td>3</td></tr></table>
Drive Enable Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	**AXIS_SERVO** If this bit is: • ON — The Drive Enable output of the axis is on. • OFF — Drive Enable output of the axis is off. **AXIS_SERVO_DRIVE** If this bit is: • ON — The drive's power structure is active. • OFF — The drive's power structure is not active. **AXIS_VIRTUAL** • Bit is always OFF.

Attribute	Axis Type	Data Type	Access	Description
Drive Fault	AXIS_SERVO	BOOL	Tag	If this bit is set, then the external servo drive has detected a fault and has communicated the existence of this fault to the servo module via the Drive Fault input. This fault condition is latched and requires execution of an explicit MAFR (Motion Axis Fault Reset) or MASR (Motion Axis Shutdown Reset) instruction to clear.

<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Drive Fault</td><td>AXIS_SERVO_DRIVE</td><td>DINT</td><td>Tag</td><td>Lets you access all the drive fault bits in one 32-bit word. This tag is the same as the Drive Fault Bits attribute.
<table>
<tr><th>Tag</th><th>Bit</th></tr>
<tr><td>Pos Soft Overtravel Fault</td><td>0</td></tr>
<tr><td>Neg Soft Overtravel Fault</td><td>1</td></tr>
<tr><td>Pos Hard Overtravel Fault</td><td>2</td></tr>
<tr><td>Neg Hard Overtravel Fault</td><td>3</td></tr>
<tr><td>Mot Feedback Fault</td><td>4</td></tr>
<tr><td>Mot Feedback Noise Fault</td><td>5</td></tr>
<tr><td>Aux Feedback Fault</td><td>6</td></tr>
<tr><td>Aux Feedback Noise Fault</td><td>7</td></tr>
<tr><td><i>Reserved</i></td><td>8</td></tr>
<tr><td>Drive Enable Input Fault</td><td>9</td></tr>
<tr><td>Common Bus Fault</td><td>10</td></tr>
<tr><td>Precharge Overload Fault</td><td>11</td></tr>
<tr><td>Guard Fault Exists</td><td>12</td></tr>
<tr><td>Ground Short Fault</td><td>13</td></tr>
<tr><td>Drive Hard Fault</td><td>14</td></tr>
<tr><td>Overspeed Fault</td><td>15</td></tr>
<tr><td>Overload Fault</td><td>16</td></tr>
<tr><td>Drive Overtemp Fault</td><td>17</td></tr>
<tr><td>Motor Overtemp Fault</td><td>18</td></tr>
<tr><td>Drive Cooling Fault</td><td>19</td></tr>
<tr><td>Drive Control Voltage Fault</td><td>20</td></tr>
<tr><td>Feedback Fault</td><td>21</td></tr>
<tr><td>Commutation Fault</td><td>22</td></tr>
<tr><td>Drive Overcurrent Fault</td><td>23</td></tr>
<tr><td>Drive Overvoltage Fault</td><td>24</td></tr>
<tr><td>Drive Undervoltage Fault</td><td>25</td></tr>
<tr><td>Power Phase Loss Fault</td><td>26</td></tr>
<tr><td>Position Error Fault</td><td>27</td></tr>
<tr><td>SERCOS Fault</td><td>28</td></tr>
<tr><td>Overtravel Fault</td><td>29</td></tr>
<tr><td><i>Reserved</i></td><td>30</td></tr>
<tr><td>Manufacturer Specific Fault</td><td>31</td></tr>
</table>
Do you want any of these faults to give the controller a major fault?
• YES - Set the General Fault Type of the motion group = Major Fault.
• NO - You must write code to handle these faults.
For more information about error codes displayed on drives and/or multi-axis motion control systems, see page 344.</td></tr>
</table>

<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Drive Fault Action</td><td>AXIS_SERVO</td><td>SINT</td><td>GSV
SSV</td><td>
<table>
<tr><th>Fault Action</th><th>Value</th></tr>
<tr><td>Shutdown</td><td>0</td></tr>
<tr><td>Disable Drive</td><td>1</td></tr>
<tr><td>Stop Motion</td><td>2</td></tr>
<tr><td>Status Only</td><td>3</td></tr>
</table>
This controller attribute is also replicated in the motion module.</td></tr>
</table>

<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Drive Fault Bits</td><td>AXIS_SERVO_DRIVE</td><td>DINT</td><td>GSV</td><td>Lets you access all the drive fault bits in one 32-bit word. This attribute is the same as the Drive Fault tag.
<table>
<tr><th>Tag</th><th>Bit</th></tr>
<tr><td>Pos Soft Overtravel Fault</td><td>0</td></tr>
<tr><td>Neg Soft Overtravel Fault</td><td>1</td></tr>
<tr><td>Pos Hard Overtravel Fault</td><td>2</td></tr>
<tr><td>Neg Hard Overtravel Fault</td><td>3</td></tr>
<tr><td>Mot Feedback Fault</td><td>4</td></tr>
<tr><td>Mot Feedback Noise Fault</td><td>5</td></tr>
<tr><td>Aux Feedback Fault</td><td>6</td></tr>
<tr><td>Aux Feedback Noise Fault</td><td>7</td></tr>
<tr><td><i>Reserved</i></td><td>8</td></tr>
<tr><td>Drive Enable Input Fault</td><td>9</td></tr>
<tr><td>Common Bus Fault</td><td>10</td></tr>
<tr><td>Precharge Overload Fault</td><td>11</td></tr>
<tr><td>GuardFaultExists</td><td>12</td></tr>
<tr><td>Ground Short Fault</td><td>13</td></tr>
<tr><td>Drive Hard Fault</td><td>14</td></tr>
<tr><td>Overspeed Fault</td><td>15</td></tr>
<tr><td>Overload Fault</td><td>16</td></tr>
<tr><td>Drive Overtemp Fault</td><td>17</td></tr>
<tr><td>Motor Overtemp Fault</td><td>18</td></tr>
<tr><td>Drive Cooling Fault</td><td>19</td></tr>
<tr><td>Drive Control Voltage Fault</td><td>20</td></tr>
<tr><td>Feedback Fault</td><td>21</td></tr>
<tr><td>Commutation Fault</td><td>22</td></tr>
<tr><td>Drive Overcurrent Fault</td><td>23</td></tr>
<tr><td>Drive Overvoltage Fault</td><td>24</td></tr>
<tr><td>Drive Undervoltage Fault</td><td>25</td></tr>
<tr><td>Power Phase Loss Fault</td><td>26</td></tr>
<tr><td>Position Error Fault</td><td>27</td></tr>
<tr><td>SERCOS Fault</td><td>28</td></tr>
<tr><td>Overtravel Fault</td><td>29</td></tr>
<tr><td><i>Reserved</i></td><td>30</td></tr>
<tr><td>Manufacturer Specific Fault</td><td>31</td></tr>
</table>
Do you want any of these faults to give the controller a major fault?

YES — Set the General Fault Type of the motion group = Major Fault.
NO — You must write code to handle these faults.

</td></tr>
</table>

<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Drive Fault Input Status</td><td>AXIS_SERVO</td><td>BOOL</td><td>Tag</td><td>Digital output from the drive that shows if there is a fault.
If this bit is:
• ON — The drive is has a fault.
• OFF — The drive does not have a fault.</td></tr>
<tr><td>Drive Hard Fault</td><td>AXIS_SERVO_DRIVE</td><td>BOOL</td><td>Tag</td><td>Set when the drive detects a serious hardware fault.</td></tr>
<tr><td>Drive Model Time Constant</td><td>AXIS_SERVO
AXIS_SERVO_DRIVE</td><td>REAL</td><td>GSV
SSV</td><td>Seconds

The value for the Drive Model Time Constant represents the lumped model time constant for the drive's current loop used by the MRAT instruction to calculate the Maximum Velocity and Position Servo Bandwidth values. The Drive Model Time Constant is the sum of the drive's current loop time constant, the feedback sample period, and the time constant associated with the velocity feedback filter. This value is set to a default value when you configure the axis.
<table>
<tr><th>For this Axis type</th><th>Details</th></tr>
<tr><td>AXIS_SERVO</td><td>This value is only used by MRAT when the axis is configured for an External Torque Servo Drive..</td></tr>
<tr><td>AXIS_SERVO_DRIVE</td><td>Since the bandwidth of the velocity feedback filter is determined by the resolution of the feedback device, the value for the Drive Model Time Constant is smaller when high resolution feedback devices are selected.</td></tr>
</table></td></tr>
<tr><td>Drive Overcurrent Fault</td><td>AXIS_SERVO_DRIVE</td><td>BOOL</td><td>Tag</td><td>Set when drive output current exceeds the predefined operating limits for the drive.</td></tr>
<tr><td>Drive Overtemp Fault</td><td>AXIS_SERVO_DRIVE</td><td>BOOL</td><td>Tag</td><td>Set when the drive's temperature exceeds the drive shutdown temperature.</td></tr>
<tr><td>Drive Overvoltage Fault</td><td>AXIS_SERVO_DRIVE</td><td>BOOL</td><td>Tag</td><td>Set when drive DC bus voltage exceeds the predefined operating limits for the bus.</td></tr>
</table>

Attribute	Axis Type	Data Type	Access	Description
Drive Polarity	AXIS_SERVO_DRIVE	DINT	GSV SSV	0 = Custom Polarity 1 = Positive Polarity 2 = Negative Polarity **Custom Polarity** Custom Polarity is used to enable custom polarity configurations using the various polarity parameters defined by the SERCOS Interface standard. **Positive/Negative Polarity** Positive and Negative Polarity bit attribute determines the overall polarity of the servo loop of the drive. All the advanced polarity parameters are automatically set based on whether the Drive Polarity is configured as Positive or Negative. Proper wiring guarantees that the servo loop is closed with negative feedback. However there is no such guarantee that the servo drive has the same sense of forward direction as the user for a given application. Negative Polarity inverts the polarity of both the command position and actual position data of the servo drive. Thus, selecting either Positive or Negative Drive Polarity makes it possible to configure the positive direction sense of the drive to agree with that of the user. This attribute is configured automatically using the MRHD and MAHD motion instructions. Refer to the Logix Motion Instruction Specification for more information on these hookup diagnostic instructions.

Attribute	Axis Type	Data Type	Access	Description
Drive Resolution	AXIS_SERVO_DRIVE	DINT	GSV	Drive Counts / Drive Unit The Drive Resolution attribute determines how many Drive Counts there are in a Drive Unit. Drive Units may be configured as Revs, Inches, or Millimeters depending on the specific drive application. Furthermore, the configured Drive Unit may apply to either a motor or auxiliary feedback device. All position, velocity, and acceleration data to the drive is scaled from the user's Position Units to Drive Units based on the Drive Resolution and Conversion Constant. The ratio of the Conversion Constant to Drive Resolution determines the number of Position Units in a Drive Unit. Conversion Constant / Drive Resolution = Drive Units (rev, inch, or mm) / Position Unit Conversely, all position, velocity, and acceleration data from the drive is scaled from the user's Position Units to Drive Units based on the Drive Resolution and Conversion Constant. The ratio of Drive Resolution and the Conversion Constant determines the number of Position Units in a Drive Unit. Drive Resolution / Conversion Constant = Position Units / Drive Unit (rev, inch, or mm) In general, the Drive Resolution value may be left at its default value of 200000 Drive Counts per Drive Unit, independent of the resolution of the feedback device(s) used by the drive. This is because the drive has its own set of scale factors that it uses to relate feedback counts to drive counts. **Drive Travel Range Limit** Because the drive's position parameters are ultimately limited to signed 32-bit representation per the SERCOS standard, the Drive Resolution parameter impacts the drive's travel range. The equation for determining the maximum travel range based on Drive Resolution is as follows: Drive Travel Range Limit = +/- 2,147,483,647 / Drive Resolution. Based on a default value of 200,000 Drive Counts per Drive Unit, the drive's range limit is 10,737 Drive Units. While it is relatively rare for this travel range limitation to present a problem, it is a simple matter to lower the Drive Resolution to increase the travel range. The downside of doing so is that the position data is then passed with lower resolution that could affect the smoothness of motion. **Fractional Unwind** In some cases, however, the user may also want to specifically configure Drive Resolution value to handle fractional unwind applications or multi-turn absolute applications requiring cyclic compensation. In these cases where the Unwind value for a rotary application does not work out to be an integer value, the Rotational Position Scaling attribute may be modified to a value that is integer divisible by the Unwind value. The following examples demonstrate how the Drive Resolution value may be used together with the Conversion Constant to handle various applications. ***Continued on next page***

Attribute	Axis Type	Data Type	Access	Description
Drive Resolution (continued)				**Rotary Gear-Head WITHOUT Aux Feedback Device** Based on a rotary motor selection, Drive Resolution would be expressed as Drive Counts per Motor Rev and be applied to the Rotational Position Resolution IDN. The user would set the Conversion Constant to Drive Counts per user-defined Position Unit. If it is a 3:1 gearbox, and the user's Position Unit is, say, Revs of the gear output shaft, the Conversion Constant is 200,000/3, which is irrational! But, in this case, you could simply set the Drive Resolution to 300,000 Drive Counts/Motor Rev and the Conversion Constant could then be set to 100,000 Drive Counts/Output Shaft Rev. This system would work with this configuration without any loss of mechanical precision, that is, a move of 1 output shaft revolution would move the output shaft exactly 1 revolution. **Linear Ball-Screw WITHOUT Aux Feedback Device** Based on a rotary motor selection, Drive Resolution would be expressed as Drive Counts per Motor Rev and be applied to the Rotational Position Resolution IDN. The user would set the Conversion Constant to Drive Counts per user-defined Position Unit. If it is a 5mm pitch ball-screw, and the user's Position Unit is, say, mm, the user simply sets the Conversion Constant to 200,000/5 or 40,000 Drive Counts per mm based on the default Drive Resolution value of 200,000 Drive Counts/Motor Rev. If the pitch is irrational, the method for addressing this is the same as described in Rotary Gear-Head WITHOUT Aux Feedback Device. **Rotary Gear-Head WITH Aux Feedback Device** Based on a rotary motor feedback selection, Drive Resolution would be expressed as Drive Counts per Aux Rev and be applied to the Rotational Position Resolution IDN. Now that position is based on the auxiliary feedback device according to the Servo Loop Configuration, the Data Reference bit of the various Scaling Types should be Load Referenced rather than Motor Referenced. The motor feedback would be rotary and resolution expressed in cycles per motor rev. The aux feedback device is also rotary and its resolution expressed in cycles per aux rev. The Aux Feedback Ratio would be set to the number of aux feedback revs per motor rev and internally applied to IDNs 121 and 122 for the purpose of relating position servo loop counts to velocity servo loop counts in a dual servo loop configuration. The Aux Feedback Ratio attribute is also used in range limit and default value calculations during configuration based on the selected motor’s specifications. If the application uses a 3:1 gearbox, and the user's Position Unit is, say, Revs of the gearbox output shaft, the Conversion Constant is still rational, since our scaling is Load Referenced! The user simply sets the Conversion Constant to 200,000 Drive Counts/Output Shaft Rev based on the default Drive Resolution value of 200,000 Drive Counts/Aux Rev. The system would work in this configuration without any loss of mechanical precision, that is, a move of 1 output shaft revolution would move the output shaft exactly 1 revolution. ***Continued on next page***

Attribute	Axis Type	Data Type	Access	Description
Drive Resolution (continued)				**Linear Ball-Screw/Ball-Screw Combination WITH Aux Feedback Device** Based on a linear aux feedback selection, Drive Resolution would be expressed as Drive Counts per Linear Unit, say Millimeters (Metric bit selection), and be applied to the Linear Position Data Scaling IDNs. Now that position is based on the auxiliary feedback device according to the Servo Loop Configuration, the Data Reference bit of the various Scaling Types should again be Load Referenced rather than Motor Referenced. The motor feedback would be rotary and resolution expressed in cycles per motor rev. The aux feedback device is now linear and its resolution expressed in cycles per, say, mm. The Aux Feedback Ratio would be set to the number of aux feedback units (mm) per motor rev and internally applied to IDN 123 to relate position servo loop counts to velocity servo loop counts in a dual servo loop configuration. The Aux Feedback Ratio attribute is also used in range limit and default value calculations during configuration based on the selected motor's specifications. If the application uses a 3:1 gearbox and a 5 mm pitch ball-screw, and the user's Position Unit is, say, cm, the Conversion Constant is again rational, since we are Load Referenced! The user sets the Conversion Constant to 20,000 Drive Counts/cm based on the default Drive Resolution value of 200000 Drive Counts/mm. This system would work in this configuration without any loss of mechanical precision, that is, a move of 10 cm would move the actuator exactly 10 cm.

Attribute	Axis Type	Data Type	Access	Description
Drive Scaling Bits	AXIS_SERVO_DRIVE	DINT	GSV	The Drive Scaling Bits attribute configuration is derived directly from the Drive Units attribute. Bits 0 = Scaling type 0 – standard 1 – custom 1 = Scaling unit 0 – rotary 1 – linear 2 = Linear scaling unit 0 – metric 1 – english 3 = Data Reference 0 – motor 1 – load **Scaling Type** The Scaling Type bit attribute is used to enable custom scaling using the position, velocity, acceleration, and torque scaling parameters defined by the SERCOS Interface standard. When the bit is clear (default), these scaling parameters are all set based on the preferred Rockwell Automation SERCOS drive scaling factors. Currently there is no Logix support for custom scaling. **Scaling Unit** The Scaling Unit attribute is used to determine whether the controller scales position, velocity, and acceleration attributes based on rotary or linear scaling parameters and their associated Drive Units that are defined by the SERCOS Interface standard. When the bit is clear (default), the corresponding bits in the SERCOS Position Data Scaling, Velocity Data Scaling, and Acceleration Data Scaling parameters are also cleared, which instructs the drive to use the rotary scaling parameters. When the bit is set, the corresponding bits in the SERCOS Position Data Scaling, Velocity Data Scaling, and Acceleration Data Scaling parameters are also set, which instructs the drive to use the linear scaling parameters. ***Continued on next page***

Attribute	Axis Type	Data Type	Access	Description
Drive Scaling Bits (continued)				**Linear Scaling Unit** When the Scaling Unit is set to linear, the Linear Scaling bit attribute is used to determine whether the controller scales position, velocity, and acceleration attributes based on Metric or English Drive Units as defined by the SERCOS Interface standard. When the bit is clear (default), the corresponding bits in the SERCOS Position Data Scaling, Velocity Data Scaling, and Acceleration Data Scaling parameters are also cleared, which instructs the drive to use the Metric scaling parameters. When the bit is set, the corresponding bits in the SERCOS Position Data Scaling, Velocity Data Scaling, and Acceleration Data Scaling parameters are also set, which instructs the drive to scale in English units. If the Scaling Unit is set to rotary, the Linear Scaling Unit bit has no affect. When interfacing to Rockwell SERCOS drive products, the Standard Drive Units based on the Scaling Unit and Linear Scaling Unit bit selections are shown in the following table: **Standard Drive Units** Rotary – Metric: Rev, English: Rev Linear – Metric: Millimeter, English: Inch **Data Reference** The Data Reference bit determines which side of the mechanical transmission to reference position, velocity, acceleration, and torque data. If motor is selected then position, velocity, acceleration, and torque data is referenced to the motor side of the transmission. If load is selected then position, velocity, acceleration, and torque data is referenced to the load-side of the transmission. This is only applicable when using an auxiliary feedback device.

Standard Drive Units

	Metric	English
Rotary	Rev	Rev
Linear	Millimeter	Inch

<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Drive Status Bits</td><td>AXIS_SERVO_DRIVE</td><td>DINT</td><td>GSV</td><td>Lets you access all the drive status bits in one 32-bit word. This attribute is the same as the Drive Status tag.
<table>
<tr><th>Tag</th><th>Bit</th></tr>
<tr><td>Servo Action Status</td><td>0</td></tr>
<tr><td>Drive Enable Status</td><td>1</td></tr>
<tr><td>Shutdown Status</td><td>2</td></tr>
<tr><td>Process Status</td><td>3</td></tr>
<tr><td>Bus Ready Status</td><td>4</td></tr>
<tr><td><i>Reserved</i></td><td>5</td></tr>
<tr><td>Home Input Status</td><td>6</td></tr>
<tr><td>Reg 1 Input Status</td><td>7</td></tr>
<tr><td>Reg 2 Input Status</td><td>8</td></tr>
<tr><td>Pos Overtravel Input Status</td><td>9</td></tr>
<tr><td>Neg Overtravel Input Status</td><td>10</td></tr>
<tr><td>Enable Input Status</td><td>11</td></tr>
<tr><td>Accel Limit Status</td><td>12</td></tr>
<tr><td>Absolute Reference Status</td><td>13</td></tr>
<tr><td>Reserved</td><td>14</td></tr>
<tr><td>Reserved</td><td>15</td></tr>
<tr><td>Velocity Lock Status</td><td>16</td></tr>
<tr><td>Velocity Standstill Status</td><td>17</td></tr>
<tr><td>Velocity Threshold Status</td><td>18</td></tr>
<tr><td>Torque Threshold Status</td><td>19</td></tr>
<tr><td>Torque Limit Status</td><td>20</td></tr>
<tr><td>Velocity Limit Status</td><td>21</td></tr>
<tr><td>Position Lock Status</td><td>22</td></tr>
<tr><td>Power Limit Status</td><td>23</td></tr>
<tr><td>Reserved</td><td>24</td></tr>
<tr><td>Low Velocity Threshold Status</td><td>25</td></tr>
<tr><td>High Velocity Threshold Status</td><td>26</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Drive Status</td><td>AXIS_SERVO_DRIVE</td><td>DINT</td><td>Tag</td><td>Lets you access all the drive status bits in one 32-bit word. This tag is the same as the Drive Status Bits attribute.
<table>
<tr><th>Tag</th><th>Bit</th></tr>
<tr><td>Servo Action Status</td><td>0</td></tr>
<tr><td>Drive Enable Status</td><td>1</td></tr>
<tr><td>Shutdown Status</td><td>2</td></tr>
<tr><td>Process Status</td><td>3</td></tr>
<tr><td>Bus Ready Status</td><td>4</td></tr>
<tr><td>Reserved</td><td>5</td></tr>
<tr><td>Home Input Status</td><td>6</td></tr>
<tr><td>Reg 1 Input Status</td><td>7</td></tr>
<tr><td>Reg 2 Input Status</td><td>8</td></tr>
<tr><td>Pos Overtravel Input Status</td><td>9</td></tr>
<tr><td>Neg Overtravel Input Status</td><td>10</td></tr>
<tr><td>Enable Input Status</td><td>11</td></tr>
<tr><td>Accel Limit Status</td><td>12</td></tr>
<tr><td>Absolute Reference Status</td><td>13</td></tr>
<tr><td>Reserved</td><td>14</td></tr>
<tr><td>Reserved</td><td>15</td></tr>
<tr><td>Velocity Lock Status</td><td>16</td></tr>
<tr><td>Velocity Standstill Status</td><td>17</td></tr>
<tr><td>Velocity Threshold Status</td><td>18</td></tr>
<tr><td>Torque Threshold Status</td><td>19</td></tr>
<tr><td>Torque Limit Status</td><td>20</td></tr>
<tr><td>Velocity Limit Status</td><td>21</td></tr>
<tr><td>Position Lock Status</td><td>22</td></tr>
<tr><td>Power Limit Status</td><td>23</td></tr>
<tr><td>Reserved</td><td>24</td></tr>
<tr><td>Low Velocity Threshold Status</td><td>25</td></tr>
<tr><td>High Velocity Threshold Status</td><td>26</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Drive Status</td><td>AXIS_SERVO_DRIVE</td><td>BOOL</td><td>Tag</td><td>
<table>
<tr><th>Tag</th><th>Bit</th></tr>
<tr><td>Servo Action Status</td><td>0</td></tr>
<tr><td>Drive Enable Status</td><td>1</td></tr>
<tr><td>Shutdown Status</td><td>2</td></tr>
<tr><td>Process Status</td><td>3</td></tr>
<tr><td>Bus Ready Status</td><td>4</td></tr>
<tr><td>Reserved</td><td>5</td></tr>
<tr><td>Home Input Status</td><td>6</td></tr>
<tr><td>Reg 1 Input Status</td><td>7</td></tr>
<tr><td>Reg 2 Input Status</td><td>8</td></tr>
<tr><td>Pos Overtravel Input Status</td><td>9</td></tr>
<tr><td>Neg Overtravel Input Status</td><td>10</td></tr>
<tr><td>Enable Input Status</td><td>11</td></tr>
<tr><td>Accel Limit Status</td><td>12</td></tr>
<tr><td>Absolute Reference Status</td><td>13</td></tr>
<tr><td>Safe-Off Mode Active Status</td><td>14 (requires Drive firmware revision 1.85 or higher)</td></tr>
</table>
</td></tr>
<tr><td>Drive Thermal Fault Action</td><td>AXIS_SERVO_DRIVE</td><td>SINT</td><td>GSV
SSV</td><td>
<table>
<tr><th>Fault Action</th><th>Value</th></tr>
<tr><td>Shutdown</td><td>0</td></tr>
<tr><td>Disable Drive</td><td>1</td></tr>
<tr><td>Stop Motion</td><td>2</td></tr>
<tr><td>Status Only</td><td>3</td></tr>
</table>
</td></tr>
<tr><td>Drive Undervoltage Fault</td><td>AXIS_SERVO_DRIVE</td><td>BOOL</td><td>Tag</td><td>Set when drive DC bus voltage is below the predefined operating limits for the bus.</td></tr>
<tr><td>Drive Unit</td><td>AXIS_SERVO_DRIVE</td><td>INT</td><td>GSV</td><td>The Drive Unit attribute establishes the unit of measure that is applied to the Drive Resolution attribute value. Units appearing in the enumerated list may be linear or rotary, english or metric. Further discrimination is provided in the enumerated list to specify whether the Drive Unit is referenced directly to the motor or to the external, or auxiliary feedback.
0 = motor revs
1 = aux revs
2 = motor inches
3 = aux inches
4 = motor mm
5 = aux mm</td></tr>
</table>

Attribute	Axis Type	Data Type	Access	Description
Drive Warning Bits	AXIS_SERVO_DRIVE	DINT	GSV*	**Warning** / **Bit**: Drive Overload Warning — 0 Drive Overtemperature Warning — 1 Motor Overtemperature Warning — 2 Cooling Error Warning — 3 **Drive Overload Warning** When the load limit of the motor is exceeded, the Overload Warning bit is set. If the condition persists, an Overload Fault occurs. This warning bit gives the control program an opportunity to reduce motor loading to avoid a future shutdown situation. **Drive Overtemperature Warning** When the over-temperature limit of the drive is exceeded, the Drive Overtemperature Warning bit is set. If the condition persists, a Drive Overtemperature Fault occurs. This warning bit gives the control program an opportunity to reduce motor loading, or increasing drive cooling, to avoid a future shutdown situation. **Motor Overtemperature Warning** When the over-temperature limit of the motor is exceeded, the Motor Overtemperature Warning bit is set. If the condition persists, a Motor Overtemperature Fault occurs. This warning bit gives the control program an opportunity to reduce motor loading, or increasing motor cooling, to avoid a future shutdown situation. **Cooling Error Warning** When the ambient temperature limit inside the drive enclosure is exceeded, the Cooling Error Warning bit sets. If the condition persists, a Cooling Error Fault occurs. This warning bit gives the control program an opportunity to increase drive cooling to avoid a future shutdown situation.

<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Dynamics Configuration Bits</td><td>AXIS_CONSUMED
AXIS_GENERIC
AXIS_SERVO
AXIS_SERVO_DRIVE
AXIS_VIRTUAL</td><td>DINT</td><td>GSV
SSV</td><td>Revision 16 improved how the controller handles changes to an S-curve profile.
Do you want to return to revision 15 or earlier behavior for S-curves?
• NO — Leave these bits ON (default).
• YES — Turn OFF one or more of these bits.
<table>
<tr><th>To turn off this change</th><th>Turn off this bit</th></tr>
<tr><td>Reduced S-curve Stop Delay
This change applies to the Motion Axis Stop (MAS) instruction. It lets you use a higher deceleration jerk to stop an <i>accelerating</i> axis more quickly.
The controller uses the deceleration jerk of the stopping instruction if it is more than the current acceleration jerk.</td><td>0</td></tr>
<tr><td>Reduced S-curve Velocity Reversals
Before revision 16, you could cause an axis to momentarily reverse direction if you decreased the deceleration jerk while the axis was decelerating. This typically happened if you tried to restart a jog or move with a lower deceleration rate while the axis was stopping. This change prevents the axis from reversing in those situations.</td><td>1</td></tr>
<tr><td>Reduced S-curve Velocity Overshoots
You can cause an axis to overshoot its programmed speed if you decrease the acceleration jerk while the axis is accelerating. This change keeps to overshoot to no more than 50% of the programmed speed.</td><td>2</td></tr>
</table></td></tr>
<tr><td>Enable Input Status</td><td>AXIS_SERVO_DRIVE</td><td>BOOL</td><td>Tag</td><td>If this bit is:
• ON — The Enable input is active.
• OFF — The Enable input is inactive.</td></tr>
<tr><td>External Drive Type</td><td>AXIS_SERVO_DRIVE</td><td>DINT</td><td>GSV
SSV</td><td>0 = torque servo
1 = velocity servo
2 = hydraulic servo
When the application requires the servo module axis to interface with an external velocity servo drive, the External Drive Type should be configured for velocity servo. This disables the servo module’s internal digital velocity loop. If the External Drive Type attribute is set to torque servo, the servo module’s internal digital velocity loop is active. This configuration is the required configuration for interfacing to a torque loop servo drive. If the External Drive Type attribute is set to hydraulic servo, the object will enable certain features specific to hydraulic servo applications. In general, selecting the hydraulic External Drive Type configures the servo loop the same as selecting the velocity servo External Drive Type.</td></tr>
</table>

<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr>
<td>Fault Configuration Bits</td>
<td>AXIS_SERVO
AXIS_SERVO_DRIVE</td>
<td>DINT</td>
<td>GSV
SSV</td>
<td>
<table>
<tr><th>Axis Type</th><th>Fault Configuration</th><th>Bit</th></tr>
<tr><td>AXIS_SERVO</td><td>Soft Overtravel Checking</td><td>0</td></tr>
<tr><td></td><td>Reserved</td><td>1</td></tr>
<tr><td></td><td>Drive Fault Checking</td><td>2</td></tr>
<tr><td></td><td>Drive Fault Normally Closed</td><td>3</td></tr>
<tr><td>AXIS_SERVO_DRIVE</td><td>Soft Overtravel Checking</td><td>0</td></tr>
<tr><td></td><td>Hard Overtravel Checking</td><td>1</td></tr>
<tr><td></td><td>Reserved</td><td>2</td></tr>
<tr><td></td><td>Reserved</td><td>3</td></tr>
<tr><td></td><td>Drive Enable Input Fault Handling</td><td>4</td></tr>
<tr><td></td><td>Drive Enable Input Checking</td><td>5</td></tr>
</table>

Change to rotary or Overtravel Checking requires Home range checks.

Soft Overtravel Checking

Soft overtravel checking is only available for a linear axis.

Do you want a Positive Soft Overtravel Fault or Negative Soft Overtravel Fault to happen if the axis goes outside the configured travel limits?

- YES — Set this bit.
- NO — Clear this bit.

The Maximum Positive Travel and Maximum Negative Travel attributes set the travel limits. This check supplements but does not replace hardware overtravel fault protection that uses hardware limit switches to directly stop axis motion at the drive and deactivate power to the system.

Hard Overtravel Checking

Hard overtravel checking is only available for a linear axis.

Do you want a Positive Hard Overtravel Fault or Negative Hard Overtravel Fault to happen if the axis activates the positive or negative overtravel limit switch inputs?

- YES — Set this bit.
- NO — Clear this bit.

Drive Fault Checking

The motion module provides a dedicated drive fault input for each axis. These inputs may be connected to fault outputs on the external drive (if provided) to notify the servo module of a fault in the drive itself. Set the Drive Fault Checking bit if you are using the servo module’s drive fault input, and then specify the drive fault contact configuration of the amplifier’s drive fault output as described below.

Continued on next page
</td>
</tr>
</table>

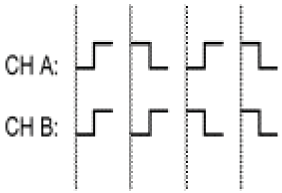
Attribute	Axis Type	Data Type	Access	Description
Fault Configuration Bits (continued)		DINT	GSV SSV	**Drive Fault Normally Closed** The Drive Fault Normally Closed bit attribute controls the sense of the Drive Fault input to the servo module. If this bit is set (true) then during normal (fault-free) operation of the drive, the Drive Fault input should be active, that is, 24 Volts. If a drive fault occurs, the drive will open its drive fault output contacts and remove 24 Volts from the servo module's Drive Fault input generating an axis Drive Fault condition. This is the default 'fail-safe' configuration. In some cases it may be necessary to clear the Drive Fault Normally Closed bit to interface with a drive system that closes its contacts when faulted. This is generally not recommended for 'fail-safe' operation. **Drive Enable Input Fault Handling** When the Drive Enable Input Fault Handling bit is set, it lets the drive post a fault based on the condition of the Drive Enable Input. If an attempt is made to enable the drive axis without an active Drive Enable Input, the drive sets a Drive Enable Input Fault. If the Drive Enable Input ever goes from active to inactive while the drive axis is enabled, the drive also sets a Drive Enable Input Fault. If the Drive Enable Input Fault Handling bit is clear (default), then the drive does not generate a Drive Enable Input Fault. **Drive Enable Input Checking** When the Drive Enable Input Checking bit is set (the default) the drive regularly checks the current state of the Drive Enable Input. This dedicated input serves as a permissive to enable the drive's power structure and servo loop. Once the drive is enabled, a transition of the Drive Enable Input from active to inactive results in a drive initiated axis stop where the axis is decelerated to a stop using the configured Stopping Torque and then disabled. If the drive enable Input Checking bit is clear, then no Drive Enable Input checking is done, hence the state of the input is irrelevant to drive operation. The state of the switch is still reported as part of the Drive Status bits attribute.

<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Feedback Fault</td><td>AXIS_SERVO
AXIS_SERVO_DRIVE</td><td>BOOL</td><td>Tag</td><td>AXIS_SERVO

Set for a specific feedback source when one of the following conditions occurs:
The differential electrical signals for one or more of the feedback channels (for example, A+ and A-, B+ and B-, or Z+ and Z-) are at the same level (both high or both low). Under normal operation, the differential signals are always at opposite levels. The most common cause of this situation is a broken wire between the feedback transducer and the servo module or drive;
Loss of feedback ‘power’ or feedback ‘common’ electrical connection between the servo module or drive and the feedback device. The controller latches this fault. Use a Motion Axis Fault Reset (MAFR) or Motion Axis Shutdown Reset (MASR) instruction to clear the fault.
AXIS_SERVO_DRIVE

Set when one of the feedback sources associated with the drive axis has a problem that prevents the drive from receiving accurate or reliable position information from the feedback device.
Set when one of the feedback sources for the axis cannot send accurate or reliable position information because there is a problem.

For AXIS_SERVO axis, possible problems are:
The differential electrical signals for one or more of the feedback channels (for example, A+ and A-, B+ and B-, or Z+ and Z-) are at the same level (both high or both low). Under normal operation, the differential signals are always at opposite levels. The most common cause of this situation is a broken wire between the feedback transducer and the servo module or drive;
Loss of feedback power or common electrical connection between the servo module or drive and the feedback device.
The controller latches this fault. Use a Motion Axis Fault Reset (MAFR) or Motion Axis Shutdown Reset (MASR) instruction to clear the fault.</td></tr>
<tr><td>Feedback Fault Action</td><td>AXIS_SERVO
AXIS_SERVO_DRIVE</td><td>SINT</td><td>GSV
SSV</td><td><table><tr><th>Fault Action</th><th>Value</th></tr><tr><td>Shutdown</td><td>0</td></tr><tr><td>Disable Drive</td><td>1</td></tr><tr><td>Stop Motion</td><td>2</td></tr><tr><td>Status Only</td><td>3</td></tr></table>This controller attribute is also replicated in the motion module.</td></tr>
</table>

<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Feedback Noise Fault</td><td>AXIS_SERVO</td><td>BOOL</td><td>Tag</td><td>Set when there is noise on the feedback device’s signal lines.
• For example, simultaneous transitions of the feedback A and B channels of an A Quad B is referred to generally as feedback noise.
• Feedback noise (shown below) is most often caused by loss of quadrature in the feedback device itself or radiated common-mode noise signals being picked up by the feedback device wiring. You can see both of these on an oscilloscope.


• To troubleshoot the loss of channel quadrature, look for:
> physical misalignment of the feedback transducer components
> excessive capacitance (or other delays) on the encoder signals
• Proper grounding and shielding usually cures radiated noise problems.
The controller latches this fault. Use a Motion Axis Fault Reset (MAFR) or Motion Axis Shutdown Reset (MASR) instruction to clear the fault.</td></tr>
<tr><td>Feedback Noise Fault Action</td><td>AXIS_SERVO
AXIS_SERVO_DRIVE</td><td>SINT</td><td>GSV
SSV</td><td><table><tr><th>Fault Action</th><th>Value</th></tr><tr><td>Shutdown</td><td>0</td></tr><tr><td>Disable Drive</td><td>1</td></tr><tr><td>Stop Motion</td><td>2</td></tr><tr><td>Status Only</td><td>3</td></tr></table>This controller attribute is also replicated in the motion module.</td></tr>
<tr><td>Backlash Compensation</td><td>AXIS_SERVO
AXIS_SERVO_DRIVE</td><td>REAL</td><td>GSV
SSV</td><td>0…100%

It is not unusual for an axis to have enough static friction (sticktion) that even with a significant position error it won’t move. Integral gain can be used to generate enough output to the drive to correct the error, but this approach may not be responsive enough for the application. An alternative is to use Backlash Compensation to break sticktion in the presence of a non-zero position error. This is done by adding, or subtracting, a fixed output level, called Backlash Compensation, to the Servo Output value based on its current sign.

The Backlash Compensation value should be just under the value that would break the sticktion. A larger value causes the axis to dither. Dither is when the axis moves rapidly back and forth centered on the commanded position.

This controller attribute is replicated in the motion module.</td></tr>
</table>

Attribute	Axis Type	Data Type	Access	Description
Backlash Compensation Window	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV	Position Units To address the issue of dither when applying Backlash Compensation and hunting from the integral gain, a Backlash Compensation Window is applied around the current command position when the axis is not being commanded to move. If the actual position is within the Backlash Compensation Window the Backlash Compensation value is applied to the Servo Output but scaled by the ratio of the position error to the Backlash Compensation Window. Within the window, the servo integrators are also disabled. Thus, once the position error reaches or exceeds the value of the Backlash Compensation Window attribute, the full Backlash Compensation value is applied. Of course, should the Backlash Compensation Window be set to zero, this feature is effectively disabled. A non-zero Backlash Compensation Window has the effect of softening the Backlash Compensation as its applied to the Servo Output and reducing the dithering effect that it can create. This generally allows higher values of Backlash Compensation to be applied. Hunting is also eliminated at the cost of a small steady-state error.

<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Guard Status</td><td>AXIS_SERVO_DRIVE</td><td>DINT</td><td>Tag
GSV</td><td>Important: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won’t see the right value as the axis runs. See Axis Info Select 1.
<table>
<tr><th>Tag</th><th>Bit</th></tr>
<tr><td>Guard OK</td><td>0</td></tr>
<tr><td>Guard Config Locked</td><td>1</td></tr>
<tr><td>Guard Gate Drive Output</td><td>2</td></tr>
<tr><td>Guard Stop Input</td><td>3</td></tr>
<tr><td>Guard Stop Request</td><td>4</td></tr>
<tr><td>Guard Stop in Progress</td><td>5</td></tr>
<tr><td>Guard Stop Decel</td><td>6</td></tr>
<tr><td>Guard Stop Standstill</td><td>7</td></tr>
<tr><td>Guard Stop Output</td><td>8</td></tr>
<tr><td>Guard Limited Speed Input</td><td>9</td></tr>
<tr><td>Guard Limited Speed Request</td><td>10</td></tr>
<tr><td>Guard Limited Speed Monitor in Progress</td><td>11</td></tr>
<tr><td>Guard Limited Speed Output</td><td>12</td></tr>
<tr><td>Guard Max Speed Monitor in Progress</td><td>13</td></tr>
<tr><td>Guard Max Accel Monitor in Progress</td><td>14</td></tr>
<tr><td>Guard Direction Monitor in Progress</td><td>15</td></tr>
<tr><td>Guard Door Control Lock</td><td>16</td></tr>
<tr><td>Guard Door Control Output</td><td>17</td></tr>
<tr><td>Guard Door Monitor Input</td><td>18</td></tr>
<tr><td>Guard Door Monitor In Progress</td><td>19</td></tr>
<tr><td>Guard Lock Monitor Input</td><td>20</td></tr>
<tr><td>Guard Enabling Switch Input</td><td>21</td></tr>
<tr><td>Guard Enabling Switch in Progress</td><td>22</td></tr>
<tr><td>Guard Reset Input</td><td>23</td></tr>
<tr><td>Guard Reset Required</td><td>24</td></tr>
<tr><td>Guard Stop Input Cycle Required</td><td>25</td></tr>
<tr><td>Reserved</td><td>26…31</td></tr>
</table>
</td></tr>
</table>

Attribute	Axis Type	Data Type	Access	Description
Guard Faults	AXIS_SERVO_DRIVE	DINT	Tag GSV	**Important:** To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won't see the right value as the axis runs. See Axis Info Select 1.

Tag	Bit
Reserved	0
Guard Internal Fault	1
Guard Configuration Fault	2
Guard Gate Drive Fault	3
Guard Reset Fault	4
Guard Feedback 1 Fault	5
Guard Feedback 2 Fault	6
Guard Feedback Speed Compare Fault	7
Guard Feedback Position Compare Fault	8
Guard Stop Input Fault	9
Guard Stop Output Fault	10
Guard Stop Decel Fault	11
Guard Stop Standstill Fault	12
Guard Stop Motion Fault	13
Guard Limited Speed Input Fault	14
Guard Limited Speed Output Fault	15
Guard Limited Speed Monitor Fault	16
Guard Max Speed Monitor Fault	17
Guard Max Accel Monitor Fault	18
Guard Direction Monitor Fault	19
Guard Door Monitor Input Fault	20
Guard Door Monitor Fault	21
Guard Door Control Output Fault	22
Guard Lock Monitor Input Fault	23
Guard Lock Monitor Fault	24
Guard Enabling Switch Monitor Input Fault	25
Guard Enabling Switch Monitor Fault	26
Guard Feedback 1 Voltage Monitor Fault	27
Guard Feedback 2 Voltage Monitor Fault	28
Reserved	29…31

Do you want any of these faults to give the controller a major fault?

- YES - Set the General Fault Type of the motion group = Major Fault.
- NO - You must write code to handle these faults.

For more information about error codes displayed on drives and/or multi-axis motion control systems, see page 344.

Attribute	Axis Type	Data Type	Access	Description
Gearing Lock Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set whenever the slave axis is locked to the master axis in a gearing relationship according to the specified gear ratio. The clutch function of the gearing planner is used to ramp an axis up, or down, to speed in a gearing process (MAG with Clutch selected). This bit is cleared during the intervals where the axis is clutching.
Gearing Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set if the axis is a slave that is currently gearing to another axis. Cleared when the gearing operation is stopped or is superseded by some other motion operation.
Ground Short Fault	AXIS_SERVO_DRIVE	BOOL	Tag	When the drive detects an imbalance in the DC bus supply current, the Ground Short Fault bit is set, indicating that current is flowing through an improper ground connection.
Group Instance	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	DINT	GSV	Instance Number of Group assigned to Axis The Group Instance attribute is used to determine what motion group object instance this axis is assigned to.
Hard Overtravel Fault Action	AXIS_SERVO_DRIVE	SINT	GSV SSV	<table><tr><th>Fault Action</th><th>Value</th></tr><tr><td>Shutdown</td><td>0</td></tr><tr><td>Disable Drive</td><td>1</td></tr><tr><td>Stop Motion</td><td>2</td></tr><tr><td>Status Only</td><td>3</td></tr></table>
Home Configuration Bits	AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	DINT	GSV SSV	0 = (Reserved) 1 = Home Switch Normally Closed 2 = Marker Edge Negative **Home Switch Normally Closed** The Home Switch Normally Closed bit attribute determines the normal state of the home limit switch used by the homing sequence. The normal state of the switch is its state prior to being engaged by the axis during the homing sequence. For example, if the Home Switch Normally Closed bit is set (true) then the condition of the switch prior to homing is closed. When the switch is engaged by the axis during the homing sequence, the switch is opened, which constitutes a homing event.
Home Direction	AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	SINT	GSV SSV	0 = unidirectional forward 1 = bidirectional forward 2 = unidirectional reverse 3 = bidirectional reverse

Attribute	Axis Type	Data Type	Access	Description
Home Event Armed Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set when a home event has been armed through execution of the MAH (Motion Axis Home) instruction. Cleared when a home event occurs.
Home Event Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set when a home event has occurred. Cleared when another MAH (Motion Axis Home) instruction is executed.
Home Event Task	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	DINT	MSG	User Event Task that is triggered to execute when a Home event occurs. An instance value of 0 indicates that no event task has been configured to be triggered by the Home Event. This attribute indicates which user Task is triggered when a home event occurs. The user Task is triggered at the same time that the Process Complete bit is set for the instruction that armed the home event. This attribute is set through internal communication from the user Task object to the Axis object when the Task trigger attribute is set to select the Home Event Task Instance attribute of the Axis. This attribute should not be set directly by an external device. This attribute is available to be read externally (Get attributes List) for diagnostic information.
Home Input Status	AXIS_SERVO AXIS_SERVO_DRIVE	BOOL	Tag	If this bit is: • ON — The home input is active. • OFF — The home input is inactive.
Home Mode	AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	SINT	GSV SSV	0 = passive 1 = active (default) 2 = absolute
Home Offset	AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV SSV	Position Units When applied to an active or passive Homing Mode, using a non-immediate Home Sequence, the Home Offset is the desired position offset of the axis Home Position from the position at which the home event occurred. The Home Offset is applied at the end of the specified homing sequence before the axis moves to the Home Position. In most cases, Home Offset is set to zero. After an active bidirectional homing sequence has completed, the axis is left at the specified Home Position. If the Home Offset is non-zero, the axis will then be offset from the marker or home switch event point by the Home Offset value. If the Home Offset is zero, the axis will sit right ‘on top of’ the marker or home switch point.

Attribute	**Axis Type**	**Data Type**	**Access**	**Description**
Home Position	AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV SSV	Position Units The Home Position is the desired absolute position for the axis after the specified homing sequence has been completed. After an active homing sequence has completed, the axis is left at the specified Home Position. In most cases, Home Position is set to zero, although any value, within the Maximum Positive and Negative Travel limits of the axis (if enabled), may also be used. (A description of the Maximum Positive and Negative Travel configuration attributes may be found in the Servo and Drive Axis Object specifications). For a rotary axis, the Home Position is constrained to be a positive number less than the Position Unwind value divided by the Conversion Constant. When configured for absolute Homing Mode, the Home Position value is applied directly to the absolute feedback device to establish an absolute position reference for the system.
Home Return Speed	AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV	Position Units / Seconds The Home Return Speed attribute controls the speed of the jog profile used after the first leg of an active bidirectional homing sequence.
Home Sequence	AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	SINT	GSV SSV	0 = immediate (default) 1 = switch 2 = marker 3 = switch then marker 4 = torque limit 5 = torque limit then marker
Home Speed	AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV	Position Units / Seconds The Home Speed attribute controls the speed of the jog profile used in the first leg of an active homing sequence as described in the above discussion of the Home Sequence Type attribute.
Axis Homed Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	The HomedStatus bit is set by the MAH instruction upon successful completion of the configured homing sequence. This bit indicates that an absolute machine reference position has been established. When this bit is set, operations that require a machine reference, such as Software Overtravel checking can be meaningfully enabled. For non-CIP Drive axis data types, the HomedStatus bit is cleared under the following conditions: • Download • Control power cycle • Reconnection to Motion Module • Feedback Loss Fault • Shutdown
Homing Status	AXIS_CONSUMED AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set if a Home motion profile is currently in progress. Cleared when the homing operation is stopped or is superseded by some other motion operation.

<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Inhibit Status</td><td>AXIS_SERVO
AXIS_SERVO_DRIVE</td><td>BOOL</td><td>Tag</td><td>Use the InhibitStatus bit of an axis to see if the axis is inhibited or uninhibited. If the bit is:
• ON — The axis is inhibited.
• OFF — The axis is uninhibited.
The controller changes the InhibitStatus bit only after all of these have happened:
• The axis has changed to inhibited or uninhibited.
• All uninhibited axes are ready.
• The connections to the motion module are running again.</td></tr>
<tr><td>InhibitAxis</td><td>AXIS_SERVO
AXIS_SERVO_DRIVE</td><td>INT</td><td>GSV
SSV</td><td><table><tr><th>To</th><th>Set the attribute to</th></tr><tr><td>Block the controller from using the axis. This inhibits the axis.</td><td>1 or any non-zero value</td></tr><tr><td>Let the controller use the axis. This uninhibits the axis.</td><td>0</td></tr></table></td></tr>
<tr><td>Integrator Hold Enable</td><td>AXIS_SERVO
AXIS_SERVO_DRIVE</td><td>SINT</td><td>GSV
SSV</td><td>When the Integrator Hold Enable attribute value is configured TRUE, the servo loop temporarily disables any enabled integrators while the command position is changing. This feature is used by point-to-point moves to minimize the integrator wind-up during motion. When the Integrator Hold Enable attribute value is FALSE, all active integrators are always enabled.
0 = disabled
1 = enabled</td></tr>
<tr><td>Inter Module Sync Fault</td><td>AXIS_SERVO</td><td>BOOL</td><td>Tag</td><td>If this bit is on, the analog servo cards of a SoftLogix5800 controller aren’t synchronized. The hardware or vbfirmware of the card causes this fault. For example, the cable between 2 cards isn’t connected.</td></tr>
<tr><td>Interpolated Actual Position</td><td>AXIS_CONSUMED
AXIS_GENERIC
AXIS_SERVO
AXIS_SERVO_DRIVE
AXIS_VIRTUAL</td><td>REAL</td><td>GSV
Tag</td><td>Interpolated Actual Position in Position Units
Interpolated Actual Position is the interpolation of the actual position, based on past axis trajectory history, at the time specified by the ‘Interpolated Time’ attribute.</td></tr>
<tr><td>Interpolated Command Position</td><td>AXIS_CONSUMED
AXIS_GENERIC
AXIS_SERVO
AXIS_SERVO_DRIVE
AXIS_VIRTUAL</td><td>REAL</td><td>GSV
Tag</td><td>Interpolated Command Position in Position Units
Interpolated Command Position is the interpolation of the commanded position, based on past axis trajectory history, at the time specified by the ‘Interpolated Time’ attribute.</td></tr>
<tr><td>Interpolation Time</td><td>AXIS_CONSUMED
AXIS_GENERIC
AXIS_SERVO
AXIS_SERVO_DRIVE
AXIS_VIRTUAL</td><td>DINT</td><td>GSV/SSV
Tag</td><td>CST time to interpolate to
Interpolated Time is the 32-bit CST time used to calculate the interpolated positions. When this attribute is updated with a valid CST value, the Interpolated Actual Position and Interpolated Command Position values are automatically calculated.</td></tr>
</table>

Attribute	Axis Type	Data Type	Access	Description
Jog Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set if a Jog motion profile is currently in progress. Cleared when the Jog is complete or is superseded by some other motion operation.
LDT Calibration Constant	AXIS_SERVO	REAL	GSV	This attribute provides for setting a calibration constant for LDT devices. This attribute is only active if the Transducer Type is set to LDT.
LDT Calibration Constant Units	AXIS_SERVO	SINT	GSV	0 = m/sec 1 = Usec/in This attribute provides a selection for the units of the LDT calibration constant attribute. This attribute is only active if the Transducer Type is set to LDT.
LDT Length	AXIS_SERVO	REAL	GSV	This attribute provides for setting the length of an LDT device. This attribute is only active if the Transducer Type is set to LDT.
LDT Length Units	AXIS_SERVO	SINT	GSV	0 = m 1 = in This attribute provides a selection for the units of the LDT length attribute. This attribute is only active if the Transducer Type is set to LDT.
LDT Recirculations	AXIS_SERVO	SINT	GSV	This attribute provides the number of recirculations. This attribute is only active if the Transducer Type is set to LDT and LDT Type is set to PWM.
LDT Scaling	AXIS_SERVO	REAL	GSV	This attribute provides for setting the scaling factor for LDT devices. This attribute is only active if the Transducer Type is set to LDT.
LDT Scaling Units	AXIS_SERVO	SINT	GSV	0 = Position Units/m 1 = Position Units/in This attribute provides a selection for the units of the LDT scaling attribute. This attribute is only active if the Transducer Type is set to LDT.
LDT Type	AXIS_SERVO	SINT	GSV	0 = PWM 1 = Start/Stop Rising 2 = Start/Stop Falling This attribute provides a selection for the LDT Type. It provides the following enumerated values: PWM, Start/Stop Rising, and Start/Stop Falling. This attribute is only active if the Transducer Type is set to LDT.
Linear Motor Mass	AXIS_SERVO_DRIVE	REAL	SSV	The Linear Motor Mass attribute is a float that specifies the unloaded moving mass of a linear motor.
Linear Motor Rated Speed	AXIS_SERVO_DRIVE	REAL	GSV	The Linear Motor Rated Speed attribute is a float that specifies the nameplate rated speed of a linear motor. For PM motors, this is generally specified at rated voltage based on either rated current, rated force, or rated power. For induction motors this value is the speed of the motor driven at rated frequency under rated force load. This value is synonymous with the term base speed.

Attribute	Axis Type	Data Type	Access	Description
Load Inertia Ratio	AXIS_SERVO_DRIVE	REAL	GSV SSV	%Rated / Pos Units per Sec^2 The Motor Inertia value represents the inertia of the motor without any load attached to the motor shaft in Torque Scaling units of %Rated / Pos Units per Sec^2. The Load Inertia Ratio attribute's value represents the ratio of the load inertia to the motor inertia. Auto-tuning uses the Motor Inertia value to calculate the Load Inertia Ratio based on the following equation. Load Inertia Ratio = (Total Inertia - Motor Inertia) / Motor Inertia. Total Inertia is directly measured by the auto-tuning algorithm and applied to the Torque Scaling attribute in units of %Rated / Pos Units per Sec^2. If the Load Inertia Ratio value is known, the Motor Inertia value can also be used to calculate a suitable Torque Scaling value for the fully loaded motor without performing an auto-tune. The equation used by RSLogix5000 to calculate the Torque Scaling value is as follows: Torque Scaling = (1 + Load Inertia Ratio) * Motor Inertia. The value for Load Inertia may be automatically calculated using Rockwell's MotionBook program while the value for Motor Inertia is derived from the Motion database file based on the motor selection.
Map Instance	AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	DINT	GSV	I/O Map Instance Number. This is 0 for virtual and consumed Data Types. The axis is associated to a specific motion compatible module by specifying the instance of the map entry representing the module.
Marker Distance	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV Tag	**Important**: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won't see the right value as the axis runs. See Axis Info Select 1. Marker Distance in Position Units Marker Distance is the distance between the axis position at which a home switch input was detected and the axis position at which the marker event was detected. This value is useful in aligning a home limit switch relative to a feedback marker pulse to provide repeatable homing operation.

Attribute	Axis Type	Data Type	Access	Description
Master Input Configuration Bits	AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	DINT	GSV SSV	Bits 0 = Master Delay Compensation 1 = Master Position Filter **Master Delay Compensation** By default, both the Position Camming and Gearing functions, when applied to a slave axis, perform Master Delay Compensation to compensate for the delay time between reading the master axis command position and applying the associated slave command position to the input of the slave's servo loop. When the master axis is running at a fixed speed, this compensation technique insures that the slave axis command position accurately tracks the actual position of the master axis; in other words, Master Delay Compensation allows for zero tracking error when gearing or camming to the actual position of a master axis. The Master Delay Compensation algorithm extrapolates the position of the master axis at the predicted time when the command position is applied to the slave's servo loop. Since master axis position is measured in discrete feedback counts and is inherently noisy, the extrapolation process amplifies that noise according to the total position update delay. The total position update delay is proportional to the Coarse Update Period of the motion group, and, if the master or the slave involves an AXIS_SERVO_DRIVE data type, it also includes the delay term that is proportional to the SERCOS Update Period. The greater the delay, the greater the noise introduced by the extrapolator. The Master Delay Compensation feature also has an extrapolation filter to filter the noise introduced by the extrapolation process. The time constant of the filter is fixed at 4x the total position update delay (independent of the Master Position Filter Bandwidth), which again is a function of the Coarse Update Period (and the SERCOS Update Period, if a AXIS_SERVO_DRIVE data type). The controller uses a 1^{st} order extrapolation algorithm that results in zero tracking error while the master axis is moving at constant velocity. If the master axis accelerates or decelerates the tracking error is non-zero and proportional to the acceleration or deceleration rate and also proportional to the square of the total position update delay time. From both a noise and acceleration error perspective, minimizing the coarse update period is vital. Some applications do not need zero tracking error between the master and the slave axis. In these cases, it may be beneficial to disable the Master Delay Compensation feature to eliminate the disturbances the extrapolation algorithm introduces to the slave axis. When the Master Delay Compensation feature is disabled (bit cleared), the slave axis will appear to be more responsive to movements of the master and run generally smoother than when Master Delay Compensation feature is enabled (bit set). However, when the master axis is running at a constant velocity, the slave will lag the master by a tracking error that is proportional to the speed of the master. This function does not get applied when the Master is a "Virtual Axis". Master Delay Compensation, even if explicitly enabled, is not applied in cases where a slave axis is gearing or camming to the master axis' command position. Since the controller generates the command position directly, there is no intrinsic master position delay to compensate for. ***Continued on next page***

Attribute	Axis Type	Data Type	Access	Description
Master Input Configuration Bits (continued)				**Master Position Filter** The Master Position Filter bit controls the activity of an independent single-pole low-pass filter that effectively filters the specified master axis position input to the slave's gearing or position camming operation. When enabled (bit set), this filter has the effect of smoothing out the actual position signal from the master axis, and thus smoothing out the corresponding motion of the slave axis. The trade-off for smoothness is an increase in lag time between the response of the slave axis to changes in motion of the master. Note that the Master Position Filter also provides filtering to the extrapolation noise introduced by the Master Delay Compensation algorithm, if enabled. When the Master Position Filter bit is set, the bandwidth of the Master Position Filter is controlled by the Master Position Filter Bandwidth attribute, see below. This can be done by setting the Master Position Filter bit and controlling the Master Position Filter Bandwidth directly. Setting the Master Position Filter Bandwidth to zero can be used to effectively disable the filter.
Master Offset	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV Tag	**Important**: To use this attribute, make sure Auto Tag Update is Enabled for the motion group (default setting). Otherwise, you won't see the right value as the axis runs. Master Offset in Master Position Units The Master Offset is the position offset that is currently applied to the master side of the position cam. The Master Offset is returned in master position units. The Master Offset will show the same unwind characteristic as the position of a linear axis.
Master Offset Move Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set if a Master Offset Move motion profile is currently in progress. This bit is cleared when the Master Offset Move is complete or is superseded by some other motion operation.
Master Position Filter Bandwidth	AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV SSV	Hertz The Master Position Filter Bandwidth attribute controls the activity of the single-pole low-pass filter that filters the specified master axis position input to the slave's gearing or position camming operation. When enabled, this filter has the effect of smoothing out the actual position signal from the master axis, and thus smoothing out the corresponding motion of the slave axis. The trade-off for smoothness is an increase in lag time between the response of the slave axis to changes in motion of the master. If the Master Position Filter is disabled, the Master Position Filter Bandwidth has no effect.

Attribute	Axis Type	Data Type	Access	Description
Maximum Acceleration	AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV SSV	Position Units / Sec^2 The Maximum Acceleration and Deceleration attribute values are frequently used by motion instructions such as MAJ, MAM, MCD, and so on, to determine the acceleration and deceleration rates to apply to the axis. These instructions all have the option of specifying acceleration and deceleration as a percent of the Maximum Acceleration and Maximum Deceleration attributes for the axis. The Maximum Acceleration and Maximum Deceleration values for the axis are automatically set to ~ 85% of the measured Tune Acceleration and Tune Deceleration by the MAAT (Motion Apply Axis Tune) instruction. If set manually, these values should typically be set to ~85% of the maximum acceleration and maximum deceleration rate of the axis. This provides sufficient 'head-room' for the axis to operate at all times within the acceleration and deceleration limits of the drive and motor.
Maximum Deceleration	AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV SSV	Position Units / Sec^2 The Maximum Acceleration and Deceleration attribute values are frequently used by motion instructions such as MAJ, MAM, MCD, and so on, to determine the acceleration and deceleration rates to apply to the axis. These instructions all have the option of specifying acceleration and deceleration as a percent of the Maximum Acceleration and Maximum Deceleration attributes for the axis. The Maximum Acceleration and Maximum Deceleration values for the axis are automatically set to ~ 85% of the measured Tune Acceleration and Tune Deceleration by the MAAT (Motion Apply Axis Tune) instruction. If set manually, these values should typically be set to ~85% of the maximum acceleration and maximum deceleration rate of the axis. This provides sufficient 'head-room' for the axis to operate at all times within the acceleration and deceleration limits of the drive and motor.
Maximum Negative Travel	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV	Position Units The Axis Object provides configurable software travel limits via the Maximum Positive and Negative Travel attributes. If the axis is configured for software overtravel limit checking by setting the Soft Overtravel Bit and the axis passes outside these maximum travel limits, a Software Overtravel Fault is issued. When software overtravel checking is enabled, appropriate values for the maximum travel in both the Maximum Positive and Maximum Negative Travel attributes need to be established with Maximum Positive Travel always greater than Maximum Negative Travel. Both of these values are specified in the configured Position Units of the axis. Soft Travel limits are always checked if the Soft Travel Limit enable attribute is true. This controller attribute is replicated in the motion module.

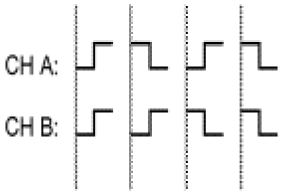
Attribute	Axis Type	Data Type	Access	Description
Maximum Positive Travel	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV	Position Units The Axis Object provides configurable software travel limits via the Maximum Positive and Negative Travel attributes. If the axis is configured for software overtravel limit checking by setting the Soft Overtravel Bit and the axis passes outside these maximum travel limits, a Software Overtravel Fault is issued. When software overtravel checking is enabled, appropriate values for the maximum travel in both the Maximum Positive and Maximum Negative Travel attributes need to be established with Maximum Positive Travel always greater than Maximum Negative Travel. Both of these values are specified in the configured Position Units of the axis. Soft Travel limits are always checked if the Soft Travel Limit enable attribute is true. This controller attribute is replicated in the motion module.
Maximum Speed	AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV SSV	Position Units / Sec The value of the Maximum Speed attribute is used by various motion instructions (for example, MAJ, MAM, MCD, and so on) to determine the steady-state speed of the axis. These instructions all have the option of specifying speed as a percent of the Maximum Speed attribute value for the axis. The Maximum Speed value for the axis is automatically set to the Tuning Speed by the MAAT (Motion Apply Axis Tune) instruction. This value is typically set to ~90% of the maximum speed rating of the motor. This provides sufficient 'head-room' for the axis to operate at all times within the speed limitations of the motor.
Memory Usage	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	DINT	MSG	Amount of memory consumed for this instance (in bytes)
Memory Use	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	INT	GSV	Controller memory space where instance exists. 105 (0x69) = I/O space 106 (0x6a) = Data Table space RSLogix 5000 software uses this attribute to create axis instances in I/O memory for axes that are either to be produced or consumed. The Memory Use attribute can only be set as part of an axis create service and is used to control which controller memory the object instance is created in.
Module Channel	AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE	SINT	GSV	Zero based channel number of the module. 0xff, indicates unassigned. The axis is associated to a specific channel on a motion module by specifying the Module Channel attribute.

Attribute	Axis Type	Data Type	Access	Description
Module Class Code	AXIS_SERVO AXIS_SERVO_DRIVE	DINT	GSV	ASA Object class code of the motion engine in the module; for example, 0xAF for the M02AE module. The ASA class code of the object in the motion module which is supporting motion; for example, 0xAF is the ASA object ID of the 'Servo Module Axis Object' residing in the 1756-M02AE module.
Module Fault	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set when a serious fault has occurred with the motion module associated with the selected axis. Usually a module fault affects all axes associated with the motion module. A module fault generally results in the shutdown of all associated axes. Reconfiguration of the motion module is required to recover from a module fault condition. Do you want this fault to give the controller a major fault? • YES — Set the General Fault Type of the motion group = Major Fault. • NO — You must write code to handle these faults.
Module Fault Bits	AXIS_CONSUMED AXIS_SERVO AXIS_SERVO_DRIVE	DINT	GSV*	Lets you access the module fault bits in one 32-bit word. This attribute is the same as the Module Faults tag. **Module Fault** / **Bit**: Control Sync Fault — 0 Module Sync Fault — 1 Timer Event Fault — 2 Module Hardware Fault — 3 SERCOS Ring Fault — 4 Inter Module Sync Fault — 5 These faults have module scope instead of axis scope. • These faults show up in all the axes that are connected to the motion module. • The motion planner updates these fault bits every coarse update period. Do you want any of these faults to give the controller a major fault? • YES — Set the General Fault Type of the motion group = Major Fault. • NO — You must write code to handle these faults.

Module Fault	Bit
Control Sync Fault	0
Module Sync Fault	1
Timer Event Fault	2
Module Hardware Fault	3
SERCOS Ring Fault	4
Inter Module Sync Fault	5

Attribute	Axis Type	Data Type	Access	Description
Module Faults	AXIS_SERVO AXIS_SERVO_DRIVE	DINT	Tag	Lets you access the module fault bits in one 32-bit word. This tag is the same as the Module Fault Bits attribute. **Module Fault** / **Bit**: Control Sync Fault – 0 Module Sync Fault – 1 Timer Event Fault – 2 Module Hardware Fault – 3 SERCOS Ring Fault – 4 Inter Module Sync Fault – 5 These faults have module scope instead of axis scope. • These faults show up in all the axes that are connected to the motion module. • The motion planner updates these fault bits every coarse update period. Do you want any of these faults to give the controller a major fault? • YES — Set the General Fault Type of the motion group = Major Fault. • NO — You must write code to handle these faults.
Module Hardware Fault	AXIS_SERVO AXIS_SERVO_DRIVE	BOOL	Tag	If this bit is set, the motion module has a hardware problem that, in general, is going to require replacement of the module.
Module Sync Fault	AXIS_SERVO AXIS_SERVO_DRIVE	BOOL	Tag	If this bit is set, the motion module lost communication with the controller and missed several position updates in a row. • The motion module can miss up to 4 position updates. After that, the motion module shuts down. • This bit clears when communication is reestablished.
Mot Feedback Fault	AXIS_SERVO_DRIVE	BOOL	Tag	Set for the A Quad B feedback device when one of these happens: • The differential electrical signals for one or more of the feedback channels (for example, A+ and A-, B+ and B-, or Z+ and Z-) are at the same level (both high or both low). Under normal operation, the differential signals are always at opposite levels. The most common cause of this situation is a broken wire between the feedback transducer and the servo module or drive. • Loss of feedback ‘power’ or feedback ‘common’ electrical connection between the servo module or drive and the feedback device. The controller latches this fault. Use a Motion Axis Fault Reset (MAFR) or Motion Axis Shutdown Reset (MASR) instruction to clear the fault.

Attribute	Axis Type	Data Type	Access	Description

Motor Feedback Noise Fault	AXIS_SERVO_DRIVE	BOOL	Tag	Set when there is noise on the feedback device's signal lines.
				• For example, simultaneous transitions of the feedback A and B channels of an A Quad B is referred to generally as feedback noise.
				• Feedback noise (shown below) is most often caused by loss of quadrature in the feedback device itself or radiated common-mode noise signals being picked up by the feedback device wiring. You can see both of these on an oscilloscope.
				
				• To troubleshoot the loss of channel quadrature, look for:
				  ▪ physical misalignment of the feedback transducer components
				  ▪ excessive capacitance (or other delays) on the encoder signals
				• Proper grounding and shielding usually cures radiated noise problems.
				The controller latches this fault. Use a Motion Axis Fault Reset (MAFR) or Motion Axis Shutdown Reset (MASR) instruction to clear the fault.

Attribute	Axis Type	Data Type	Access	Description
Motion Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	DINT	Tag	Lets you access all the motion status bits in one 32-bit word. This tag is the same as the Motion Status Bits attribute.

Motion Status	Bit
Accel Status	0
Decel Status	1
Move Status	2
Jog Status	3
Gearing Status	4
Homing Status	5
Stopping Status	6
Axis Homed Status	7
Position Cam Status	8
Time Cam Status	9
Position Cam Pending Status	10
Time Cam Pending Status	11
Gearing Lock Status	12
Position Cam Lock Status	13
Reserved	14
Master Offset Move Status	15
Coordinated Motion Status	16

<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Motion Status Bits</td><td>AXIS_CONSUMED
AXIS_GENERIC
AXIS_SERVO
AXIS_SERVO_DRIVE
AXIS_VIRTUAL</td><td>DINT</td><td>GSV</td><td>Lets you access all the motion status bits in one 32-bit word. This attribute is the same as the Motion Status tag.
<table>
<tr><th>Motion Status</th><th>Bit</th></tr>
<tr><td>Accel Status</td><td>0</td></tr>
<tr><td>Decel Status</td><td>1</td></tr>
<tr><td>Move Status</td><td>2</td></tr>
<tr><td>Jog Status</td><td>3</td></tr>
<tr><td>Gearing Status</td><td>4</td></tr>
<tr><td>Homing Status</td><td>5</td></tr>
<tr><td>Stopping Status</td><td>6</td></tr>
<tr><td>Homed Status</td><td>7</td></tr>
<tr><td>Position Cam Status</td><td>8</td></tr>
<tr><td>Time Cam Status</td><td>9</td></tr>
<tr><td>Position Cam Pending Status</td><td>10</td></tr>
<tr><td>Time Cam Pending Status</td><td>11</td></tr>
<tr><td>Gearing Lock Status</td><td>12</td></tr>
<tr><td>Position Cam Lock Status</td><td>13</td></tr>
<tr><td>Reserved</td><td>14</td></tr>
<tr><td>Master Offset Move Status</td><td>15</td></tr>
<tr><td>Coordinated Motion Status</td><td>16</td></tr>
</table>
</td></tr>
<tr><td>Motor Capacity</td><td>AXIS_SERVO_DRIVE</td><td>REAL</td><td>GSV
Tag</td><td>Important: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won’t see the right value as the axis runs. See Axis Info Select 1.
The present utilization of motor capacity as a percent of rated capacity.</td></tr>
<tr><td>Motor Data</td><td>AXIS_SERVO_DRIVE</td><td>Struct {
INT;
SINT
[256]}</td><td>MSG</td><td>Struct {length; data[ ]}
The Motor Data attribute is a structure with a length element and an array of bytes that contains important motor configuration information needed by an A-B SERCOS drive to operate the motor. The length element represents the number of valid data elements in the data array. The meaning of data within the data array is understood only by the drive. The block of data stored in the Motor Data attribute is derived at configuration time from an RSLogix 5000 software motion database file.</td></tr>
<tr><td>Motor Electrical Angle</td><td>AXIS_SERVO_DRIVE</td><td>REAL</td><td>GSV
Tag</td><td>Important: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won’t see the right value as the axis runs. See Axis Info Select 1.
Degrees
The present electrical angle of the motor shaft.</td></tr>
</table>

<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Motor Feedback Configuration</td><td>AXIS_SERVO_DRIVE</td><td>INT</td><td>GSV</td><td>The controller and drive use this for scaling the feedback device counts. These attributes are derived from the corresponding Motor and Auxiliary Feedback Unit attributes.
Bit
0 = Feedback type
• 0 — rotary (default)
• 1 — linear
1 = (reserved)
2 = Linear feedback unit
• 0 — metric
• 1 — english
3 = Feedback Polarity (Aux Only)
• 0 — not inverted
• 1 — inverted
<table>
<tr><th colspan="3">If the bits are</th><th rowspan="2">Then Feedback Resolution is scaled to</th></tr>
<tr><th>2</th><th>1</th><th>0</th></tr>
<tr><td>0</td><td></td><td>0</td><td>Feedback Cycles per Feedback Rev</td></tr>
<tr><td>1</td><td></td><td>0</td><td>Feedback Cycles per Feedback Rev</td></tr>
<tr><td>0</td><td></td><td>1</td><td>Feedback Cycles per mm</td></tr>
<tr><td>1</td><td></td><td>1</td><td>Feedback Cycles per inch</td></tr>
</table>
Feedback Polarity
The Feedback Polarity bit attribute can be used to change the sense of direction of the feedback device. This bit is only valid for auxiliary feedback devices. When performing motor/feedback hookup diagnostics on an auxiliary feedback device using the MRHD and MAHD instructions, the Feedback Polarity bit is configured for the auxiliary feedback device to insure negative feedback into the servo loop. Motor feedback devices must be wired properly for negative feedback since the Feedback Polarity bit is forced to 0, or non-inverted.</td></tr>
<tr><td>Motor Feedback Interpolation Factor</td><td>AXIS_SERVO_DRIVE</td><td>DINT</td><td>GSV</td><td>Feedback Counts per Cycle
The Feedback Interpolation attributes establish how many Feedback Counts there are in one Feedback Cycle. The Feedback Interpolation Factor depends on both the feedback device and the drive feedback circuitry. Quadrature encoder feedback devices and the associated drive feedback interface typically support 4x interpolation, so the Interpolation Factor for these devices would be set to 4 Feedback Counts per Cycle (Cycles are sometimes called Lines). High Resolution Sin/Cosine feedback device types can have interpolation factors as high as 2048 Counts per Cycle. The product to the Feedback Resolution and the corresponding Feedback Interpolation Factor is the overall resolution of the feedback channel in Feedback Counts per Feedback Unit. In our example, a Quadrature encoder with a 2000 line/rev resolution and 4x interpolation factor would have an overall resolution of 8000 counts/rev.</td></tr>
</table>

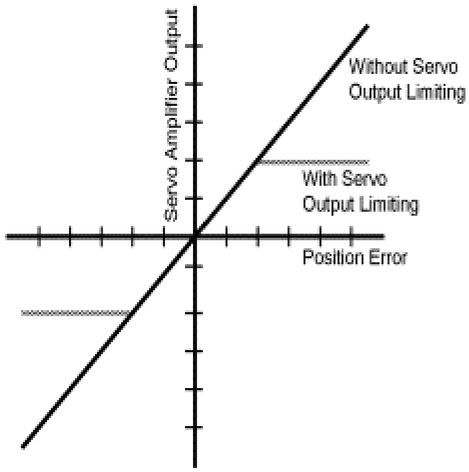
<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Motor Feedback Resolution</td><td>AXIS_SERVO_DRIVE</td><td>DINT</td><td>GSV</td><td>Cycles per Motor Feedback Unit
The Motor and Aux Feedback Resolution attributes are used to provide the A-B drive with the resolution of the associated feedback device in cycles per feedback unit. These parameters provide the SERCOS drive with critical information needed to compute scaling factors used to convert Drive Counts to Feedback counts.</td></tr>
<tr><td>Motor Feedback Type</td><td>AXIS_SERVO_DRIVE</td><td>INT</td><td>GSV</td><td>The Motor and Aux Feedback Type attributes are used to identify the motor mounted or auxiliary feedback device connected to the drive.
<table>
<tr><th>Feedback Type</th><th>Code</th><th>Rotary Only</th><th>Linear Only</th><th>Rotary or Linear</th></tr>
<tr><td><None></td><td>0x0000</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>SRS</td><td>0x0001</td><td>X</td><td></td><td></td></tr>
<tr><td>SRM</td><td>0x0002</td><td>X</td><td></td><td></td></tr>
<tr><td>SCS</td><td>0x0003</td><td>X</td><td></td><td></td></tr>
<tr><td>SCM</td><td>0x0004</td><td>X</td><td></td><td></td></tr>
<tr><td>SNS</td><td>0x0005</td><td>X</td><td></td><td></td></tr>
<tr><td>MHG</td><td>0x0006</td><td>X</td><td></td><td></td></tr>
<tr><td>Resolver</td><td>0x0007</td><td>X</td><td></td><td></td></tr>
<tr><td>Analog Reference</td><td>0x0008</td><td>X</td><td></td><td></td></tr>
<tr><td>Sin/Cos</td><td>0x0009</td><td></td><td></td><td>X</td></tr>
<tr><td>TTL</td><td>0x000A</td><td></td><td></td><td>X</td></tr>
<tr><td>UVW</td><td>0x000B</td><td></td><td></td><td>X</td></tr>
<tr><td>Unknown Stegmann</td><td>0x000C</td><td></td><td></td><td>X</td></tr>
<tr><td>Endat</td><td>0x000D</td><td></td><td></td><td>X</td></tr>
<tr><td>RCM21S-4</td><td>0x000E</td><td>X</td><td></td><td></td></tr>
<tr><td>RCM21S-6</td><td>0x000F</td><td>X</td><td></td><td></td></tr>
<tr><td>RCM21S-8</td><td>0x0010</td><td>X</td><td></td><td></td></tr>
<tr><td>LINCODER</td><td>0x0011</td><td></td><td>X</td><td></td></tr>
<tr><td>Sin/Cos with Hall</td><td>0x0012</td><td></td><td></td><td>X</td></tr>
<tr><td>TTL with Hall</td><td>0x0013</td><td></td><td></td><td>X</td></tr>
</table></td></tr>
<tr><td>Motor Feedback Units</td><td>AXIS_SERVO_DRIVE</td><td>INT</td><td>GSV</td><td>The Motor Feedback Units attribute establishes the unit of measure that is applied to the Motor Feedback Resolution attribute value. The Aux Feedback Units attribute establishes the unit of measure that is applied to the Aux Feedback Resolution attribute value. Units appearing in the enumerated list cover linear or rotary, english or metric feedback devices.
0 = revs
1 = inches
2 = mm</td></tr>
</table>

Attribute	Axis Type	Data Type	Access	Description
Motor ID	AXIS_SERVO_DRIVE	INT	GSV	The Motor ID attribute contains the enumeration of the specific A-B motor catalog number associated with the axis. If the Motor ID does not match that of the actual motor, an error is generated during the drive configuration process.
Motor Inertia	AXIS_SERVO_DRIVE	REAL	GSV SSV	%Rated / Pos Units per Sec^2 The Motor Inertia value represents the inertia of the motor without any load attached to the motor shaft in Torque Scaling units of %Rated / Pos Units per Sec^2. The Load Inertia Ratio attribute's value represents the ratio of the load inertia to the motor inertia. Auto-tuning uses the Motor Inertia value to calculate the Load Inertia Ratio based on the following equation. Load Inertia Ratio = (Total Inertia - Motor Inertia) / Motor Inertia. Total Inertia is directly measured by the auto-tuning algorithm and applied to the Torque Scaling attribute in units of %Rated / Pos Units per Sec^2. If the Load Inertia Ratio value is known, the Motor Inertia value can also be used to calculate a suitable Torque Scaling value for the fully loaded motor without performing an auto-tune. The equation used by RSLogix5000 to calculate the Torque Scaling value is as follows: Torque Scaling = (1 + Load Inertia Ratio) * Motor Inertia. The value for Load Inertia may be automatically calculated using Rockwell's MotionBook program while the value for Motor Inertia is derived from the Motion database file based on the motor selection.
Motor Overtemp Fault	AXIS_SERVO_DRIVE	BOOL	Tag	Set when the motor's temperature exceeds the motor shutdown temperature.
Motor Rated Continuous Current	AXIS_SERVO_DRIVE	REAL	GSV	The Motor Rated Continuous Current attribute is a float that specifies the nameplate AC continuous current rating of the motor. This represents the current applied to the motor under full load conditions at rated speed and voltage. Any positive number. This is a database number and should not be changed.
Motor Rated Peak Current	AXIS_SERVO_DRIVE	REAL	GSV	The Motor Rated Peak Current attribute is a float that specifies the peak or intermittent current rating of the motor. The peak current rating of the motor is often determined by either the thermal constraints of the stator winding or the saturation limits of PM motor magnetic material. Any positive number. This is a database number and should not be changed.
Motor Thermal Fault Action	AXIS_SERVO_DRIVE	SINT	GSV SSV	<table><tr><th>Fault Action</th><th>Value</th></tr><tr><td>Shutdown</td><td>0</td></tr><tr><td>Disable Drive</td><td>1</td></tr><tr><td>Stop Motion</td><td>2</td></tr><tr><td>Status Only</td><td>3</td></tr></table>

Attribute	Axis Type	Data Type	Access	Description
Move Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set if a Move motion profile is currently in progress. Cleared when the Move is complete or is superseded by some other motion operation.
Neg Dynamic Torque Limit	AXIS_SERVO_DRIVE	REAL	Tag	The currently operative negative positive torque/current limit magnitude. It should be the lowest value of all torque/current limits in the drive at a given time, including: amplifier peak limit, motor peak limit, user current limit, amplifier thermal limit, and motor thermal limit.
Neg Hard Overtravel Fault	AXIS_SERVO_DRIVE	BOOL	Tag	Set if the axis moves beyond the negative direction position limits as established by hardware overtravel limit switches mounted on the equipment. This fault can only occur when the drive is in the enabled state and the Hard Overtravel Checking bit is set in the Fault Configuration Bits attribute. If the Hard Overtravel Fault Action is set for Stop Command, the faulted axis can be moved or jogged back inside the soft overtravel limits. Any attempt, however, to move the axis further beyond the hard overtravel limit switch using a motion instruction results in an instruction error. To recover from this fault, the axis must be moved back within normal operation limits of the equipment and the limit switch closed. This fault condition is latched and requires execution of an Motion Axis Fault Reset (MAFR) or Motion Axis Shutdown Reset (MASR) instruction to clear. Any attempt to clear the fault while the overtravel limit switch is still open and the drive is enabled is unsuccessful.
Neg Overtravel Input Status	AXIS_SERVO AXIS_SERVO_DRIVE	BOOL	Tag	If this bit is: • ON — The Negative Overtravel input is active. • OFF — The Negative Overtravel input is inactive.
Neg Soft Overtravel Fault	AXIS_SERVO AXIS_SERVO_DRIVE	BOOL	Tag	If this bit is: • ON — The axis moved or tried to move past the Maximum Negative travel limit. • OFF — The axis moved back within the Maximum Negative travel limit This fault can only happen when the drive is enabled and you configure the axis for Soft Travel Limits. If the Soft Overtravel Fault Action is set for Stop Command, the faulted axis can be moved or jogged back inside the soft overtravel limits. Any attempt, however, to move the axis further beyond the soft overtravel limit using a motion instruction results in an instruction error. As soon as the axis is moved back within the specified soft overtravel limits, the corresponding soft overtravel fault bit is automatically cleared. However the soft overtravel fault stays through any attempt to clear it while the axis position is still beyond the specified travel limits while the axis is enabled.

Attribute	Axis Type	Data Type	Access	Description
Negative Dynamic Torque Limit	AXIS_SERVO_DRIVE	REAL	GSV Tag	**Important**: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won't see the right value as the axis runs. See Axis Info Select 1. %Rated The currently operative maximum negative torque/current limit magnitude. The value should be the lowest value of all torque/current limits in the drive at a given time. This limit includes the amplifier peak limit, motor peak limit, user current limit, amplifier thermal limit, and the motor thermal limit.
Output Cam Execution Targets	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	DINT	GSV	Represents the number of Output Cam nodes attached to this axis. Valid range = 0-8 with default of 0. The Output Cam Execution Targets attribute is used to specify the number of Output Cam nodes attached to the axis. This attribute can only be set as part of an axis create service and dictates how many Output Cam Nodes are created and associated to that axis. Each Output Cam Execution Target requires approximately 5.4k bytes of data table memory to store persistent data. With four Output Cam Execution Targets per axis, an additional 21.6k bytes of memory is required for each axis. The ability to configure the number of Output Cam Execution Targets for a specific axis reduces the memory required per axis for users who do not need Output Cam functionality, or only need 1 or 2 Output Cam Execution Targets for a specific axis. Each axis can be configured differently.
Output Cam Lock Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	DINT	GSV Tag	Set of Output Cam Lock Status bits The Output Cam Lock Status bit is set when an Output Cam has been armed. This would be initiated by executing an MAOC instruction with Immediate execution selected, when a pending output cam changes to armed, or when the axis approaches or passes through the specified axis arm position. As soon as this output cam current position moves beyond the cam start or cam stop position, the Output Cam Lock bit is cleared. This bit is also cleared if the Output Cam is terminated by a MDOC instruction.
Output Cam Lock Status	AXIS_CONSUMED AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	DINT	Tag	A set of bits that are set when an Output Cam is locked to the Master Axis. The bit number corresponds with the execution target number. One bit per execution target.
Output Cam Pending Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	DINT	GSV Tag	A set of bits that are set when an Output Cam is waiting for an armed Output Cam to move beyond its cam start/cam end position. The bit number corresponds with the execution target number. One bit per execution target. The Output Cam Pending Status bit is set if an Output Cam is currently pending the completion of another Output Cam. This would be initiated by executing an MAOC instruction with Pending execution selected. As soon as this output cam is armed, being triggered when the currently executing Output Cam has completed, the Output Cam Pending bit is cleared. This bit is also cleared if the Output Cam is terminated by a MDOC instruction.

Attribute	Axis Type	Data Type	Access	Description
Output Cam Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	DINT	GSV Tag	A set of bits that are set when the Output Cam has been initiated. The bit number corresponds with the execution target number. One bit per execution target. The Output Cam Status bit is set when an Output Cam has been initiated. The Output Cam Status bit is reset when the cam position moves beyond the cam start or cam end position in 'Once' execution mode with no Output Cam pending or when the Output Cam is terminated by a MDOC instruction.
Output Cam Transition Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	DINT	GSV Tag	A set of bits that are set when the transition from the current armed Output Cam to the pending Output Cam is in process. The bit number corresponds with the execution target number. One bit per execution target. The Output Cam Transition Status bit is set when a transition between the currently armed and the pending Output Cam is in process. Therefore, each Output Cam controls a subset of Output Bits. The Output Cam Transition Status bit is reset, when the transition to the pending Output Cam is complete or when the Output Cam is terminated by a MDOC instruction.

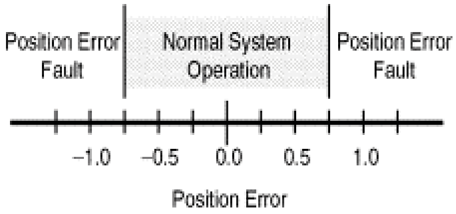
Attribute	Axis Type	Data Type	Access	Description
Output Limit	AXIS_SERVO	REAL	GSV SSV	0.0…10.0V This controller attribute is replicated in the motion module. The Output Limit attribute provides a method of limiting the maximum servo output voltage of a physical axis to a specified level. The servo output for the axis as a function of position servo error, both with and without servo output limiting, is shown below.  The servo output limit may be used as a software current or torque limit if you are using a servo drive in torque (current) loop mode. The percentage of the drive's maximum current that the servo controller commands is equal to the specified servo output limit. For example, if the drive is capable of 30 Amps of current for a 10 Volt input, setting the servo output limit to 5V limits the maximum drive current to 15 Amps. The servo output limit may also be used if the drive cannot accept the full ±10 Volt range of the servo output. In this case, the servo output limit value effectively limits the maximum command sent to the amplifier. For example, if the drive can only accept command signals up to ±7.5 Volts, set the servo output limit value to 7.5 volts.
Output Limit Status	AXIS_SERVO	BOOL	Tag	If this bit is: • ON — The servo output is at or past the Output Limit value. • OFF — The servo output is within the Output Limit value

Attribute	Axis Type	Data Type	Access	Description
Output LP Filter Bandwidth	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV	Hertz The Output LP (Low Pass) Filter Bandwidth controls the bandwidth of the drive's low-pass digital output filter. The programmable low-pass output filter is bypassed if the configured Output LP Filter Bandwidth for this filter is set to zero (the default). This output filter can be used to filter out, or reduce, high frequency variation of the drive output to the motor. The lower the Output LP Filter Bandwidth, the greater the attenuation of these high frequency components of the output signal. Unfortunately, since the low-pass filter adds lag to the servo loop which pushes the system towards instability, decreasing the Output LP Filter Bandwidth usually requires lowering the Position or Velocity Proportional Gain of the system to maintain stability. The output filter is particularly useful in high inertia applications where resonance behavior can severely restrict the maximum bandwidth capability of the servo loop. This controller attribute is replicated in the motion module.
Output Notch Filter Frequency	AXIS_SERVO_DRIVE	REAL	GSV SSV	Hertz The Output Notch Filter Frequency attribute controls the center frequency of the drive's digital notch filter. Currently implemented as a 2nd order digital filter with a fixed Q, the Notch Filter provides approximately 40DB of output attenuation at the Notch Filter Frequency. The programmable notch filter is bypassed if the configured Output Notch Filter Frequency for this filter is set to zero (the default). This output notch filter is particularly useful in attenuating mechanical resonance phenomena. The output filter is particularly useful in high inertia applications where mechanical resonance behavior can severely restrict the maximum bandwidth capability of the servo loop.
Output Offset	AXIS_SERVO	REAL	GSV SSV	+/-10V Another common situation when interfacing an external Servo Drive, particularly for velocity servo drives, is the effect of drive offset. Cumulative offsets of the servo module's DAC output and the Servo Drive Input result in a situation where a zero commanded Servo Output value causes the axis to 'drift'. If the drift is excessive it can play havoc on the Hookup Diagnostic and Tuning procedures as well as result in a steady-state non-zero position error when the servo loop is closed. Output offset compensation can be used to correct this problem by adding a fixed value, called Output Offset, to the Servo Output. This value is chosen to achieve near zero drive velocity when the uncompensated Servo Output value is zero. This controller attribute is replicated in the motion module.
Overload Fault	AXIS_SERVO_DRIVE	BOOL	Tag	When the load limit of the motor/drive is first exceeded, the Overload warning bit is set. If the condition persists, the Overload fault is set. Often this bit is tied into the IT limit of the drive.

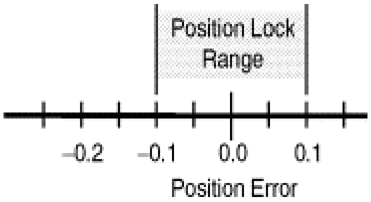
Attribute	Axis Type	Data Type	Access	Description
Overspeed Fault	AXIS_SERVO_DRIVE	BOOL	Tag	Set when the speed of the axis as determined from the feedback has exceeded the overspeed limit which is typically set to 150% of configured velocity limit for the motor.
Physical Axis Fault	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	If this bit is set, the physical axis has one or more faults. The specific faults can then be determined through access to the fault attributes of the associated physical axis. Do you want this fault to give the controller a major fault? • YES — Set the General Fault Type of the motion group = Major Fault. • NO — You must write code to handle these faults.
Pos Dynamic Torque Limit	AXIS_SERVO_DRIVE	REAL	Tag	The currently operative maximum positive torque/current limit magnitude. It should be the lowest value of all torque/current limits in the drive at a given time, including: amplifier peak limit, motor peak limit, user current limit, amplifier thermal limit, and motor thermal limit.
Pos Hard Overtravel Fault	AXIS_SERVO_DRIVE	BOOL	Tag	Set if the axis moves beyond the current position limits as established by hardware overtravel limit switches mounted on the equipment. This fault can only occur when the drive is in the enabled state and the Hard Overtravel Checking bit is set in the Fault Configuration Bits attribute. If the Hard Overtravel Fault Action is set for Stop Command, the faulted axis can be moved or jogged back inside the soft overtravel limits. Any attempt, however, to move the axis further beyond the hard overtravel limit switch using a motion instruction results in an instruction error. To recover from this fault, the axis must be moved back within normal operation limits of the equipment and the limit switch closed. This fault condition is latched and requires execution of an Motion Axis Fault Reset (MAFR) or Motion Axis Shutdown Reset (MASR) instruction to clear. Any attempt to clear the fault while the overtravel limit switch is still open and the drive is enabled is unsuccessful.
Pos Lock Status	AXIS_SERVO AXIS_SERVO_DRIVE	DINT	Tag	Set when the magnitude of the axis position error has become less than or equal to the configured Position Lock Tolerance value for the associated physical axis.
Pos Overtravel Input Status	AXIS_SERVO AXIS_SERVO_DRIVE	BOOL	Tag	If this bit is: • ON — The Positive Overtravel input is active. • OFF — The Positive Overtravel input is inactive.

Attribute	Axis Type	Data Type	Access	Description
Pos Soft Overtravel Fault	AXIS_SERVO AXIS_SERVO_DRIVE	BOOL	Tag	If this bit is: • ON — The axis moved or tried to move past the Maximum Positive travel limit. • OFF — The axis moved back within the Maximum Positive travel limit This fault can only happen when the drive is enabled and you configure the axis for Soft Travel Limits. If the Soft Overtravel Fault Action is set for Stop Command, the faulted axis can be moved or jogged back inside the soft overtravel limits. Any attempt, however, to move the axis further beyond the soft overtravel limit using a motion instruction results in an instruction error. As soon as the axis is moved back within the specified soft overtravel limits, the corresponding soft overtravel fault bit is automatically cleared. However the soft overtravel fault stays through any attempt to clear it while the axis position is still beyond the specified travel limits while the axis is enabled.
Position Cam Lock Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set whenever the master axis satisfies the starting condition of a currently active Position Cam motion profile. The starting condition is established by the Start Control and Start Position parameters of the MAPC instruction. This bit is bit is cleared when the current position cam profile completes, or is superseded by some other motion operation. In unidirectional master direction mode, the Position Cam Lock Status bit is cleared when moving in the 'wrong' direction and sets when moving in the 'correct' direction.
Position Cam Pending Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set if a Position Cam motion profile is currently pending the completion of a currently executing cam profile. This would be initiated by executing an MAPC instruction with Pending execution selected. This bit is cleared when the current position cam profile completes, initiating the start of the pending cam profile. This bit is also cleared if the position cam profile completes, or is superseded by some other motion operation.
Position Cam Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set if a Position Cam motion profile is currently in progress. Cleared when the Position Cam is complete or is superseded by some other motion operation.
Position Command	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV Tag	Position Command in Position Units **Important**: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won't see the right value as the axis runs. See Axis Info Select 1. Position Command is the current value of the Fine Command Position into the position loop summing junction, in configured axis Position Units. Within the active servo loop, the Position Command value is used to control the position of the axis.
Position Data Scaling	AXIS_SERVO_DRIVE	INT	GSV	This attribute is derived from the Drive Units attribute. See IDN 76 in IEC 1491.

Attribute	Axis Type	Data Type	Access	Description
Position Data Scaling Exp	AXIS_SERVO_DRIVE	INT	GSV	This attribute is derived from the Drive Units attribute. See IDN 78 in IEC 1491.
Position Data Scaling Factor	AXIS_SERVO_DRIVE	DINT	GSV	This attribute is derived from the Drive Units attribute. See IDN 77 in IEC 1491.
Position Differential Gain	AXIS_SERVO	REAL	GSV SSV	In some External Velocity Servo Drive applications where the level of damping provided by the external drive is insufficient for good position servo loop performance, additional damping may be achieved via the Position Loop Differential Gain. Assuming a non-zero Position Loop Differential Gain value, the difference between the current Position Error value and the last Position Error value is computed. This value is then multiplied by the Position Loop Differential Gain to produce a component to the Servo Output or Velocity Command that attempts to correct for the change in position error, creating a 'damping' effect. Increasing this gain value results in greater 'damping' of the axis.
Position Error	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV Tag	**Important**: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won't see the right value as the axis runs. See Axis Info Select 1. Position Error in Position Units Position Error is the difference, in configured axis Position Units, between the command and actual positions of an axis. For an axis with an active servo loop, position error is used, along with other error terms, to drive the motor to the condition where the actual position is equal to the command position.
Position Error Fault	AXIS_SERVO AXIS_SERVO_DRIVE	BOOL	Tag	Set when the axis position error exceeds the Position Error Tolerance. This fault can only occur when the drive is in the enabled state. The controller latches this fault. Use a Motion Axis Fault Reset (MAFR) or Motion Axis Shutdown Reset (MASR) instruction to clear the fault.
Position Error Fault Action	AXIS_SERVO AXIS_SERVO_DRIVE	SINT	GSV SSV	<table><tr><th>Fault Action</th><th>Value</th></tr><tr><td>Shutdown</td><td>0</td></tr><tr><td>Disable Drive</td><td>1</td></tr><tr><td>Stop Motion</td><td>2</td></tr><tr><td>Status Only</td><td>3</td></tr></table> This controller attribute is replicated in the motion module.

Attribute	Axis Type	Data Type	Access	Description
Position Error Tolerance	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV	Position Units The Position Error Tolerance parameter specifies how much position error the servo or drive tolerates before issuing a Position Error Fault. Like the position lock tolerance, the position error tolerance is interpreted as a ± quantity. For example, specifying a position error tolerance of 0.75 Position Units means that a Position Error Fault is generated whenever the position error of the axis is greater than 0.75 or less than -0.75 Position Units, as shown.  The self tuning routine sets the position error tolerance to twice the following error at maximum speed based on the measured response of the axis. In most applications, this value provides reasonable protection in case of an axis fault or stall condition without nuisance faults during normal operation. If you need to change the calculated position error tolerance value, the recommended setting is 150% to 200% of the position error while the axis is running at its maximum speed. This controller attribute is replicated in the motion module.
Position Feedback	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV Tag	**Important**: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won't see the right value as the axis runs. See Axis Info Select 1. Position Feedback in Position Units Position Feedback is the current value of the Fine Actual Position into the position loop summing junction, in configured axis Position Units. Within the servo loop, the Position Feedback represents the current position of the axis.

Attribute	Axis Type	Data Type	Access	Description
Position Integral Gain	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV	1/mSec-Sec This controller attribute is replicated in the motion module. Position Integral Gain (Pos I Gain) improves the steady-state positioning performance of the system. By using Position Integral Gain, it is possible to achieve accurate axis positioning despite the presence of such disturbances as static friction or gravity. Increasing the integral gain generally increases the ultimate positioning accuracy of the system. Excessive integral gain, however, results in system instability. Every servo update, the current Position Error is accumulated in a variable called the Position Integral Error. This value is multiplied by the Position Integral Gain to produce a component to the Velocity Command that attempts to correct for the position error. The characteristic of Pos I Gain correction, however, is that any non-zero Position Error accumulates in time to generate enough force to make the correction. This attribute of Pos I Gain makes it invaluable in applications where positioning accuracy or tracking accuracy is critical. The higher the Pos I Gain value the faster the axis is driven to the zero Position Error condition. Unfortunately, Pos I Gain control is intrinsically unstable. Too much Pos I Gain results in axis oscillation and servo instability. If the axis is configured for an external velocity loop servo drive, the Pos I Gain should be zero–most analog velocity loop servo amplifiers have integral gain of their own and do not tolerate *any* amount of Pos I Gain in the position loop without producing severe oscillations. If Pos I Gain is necessary for the application, the velocity integrator in the drive must be disabled. In certain cases, Pos I Gain control is disabled. One such case is when the servo output to the axis' drive is saturated. Continuing integral control behavior in this case would only exacerbate the situation. Another common case is when performing certain motion. When the Integrator Hold Enable attribute is set, the servo loop automatically disables the integrator during commanded motion. While the Pos I Gain, if employed, is typically established by the automatic servo tuning procedure, the Pos I Gain value may also be set manually. You can compute the Pos I Gain based on the current or computed value for the Pos P Gain using the following formula: Pos I Gain = 0.25 * 0.001 Sec/mSec * (Pos P Gain)^2 Assuming a Pos P Gain value of 100 Sec^{-1} this results in a Pos I Gain value of 2.5 ~0.1 mSec^{-1}-Sec^{-1}

Attribute	Axis Type	Data Type	Access	Description
Position Integrator Error	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV Tag	**Important**: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won't see the right value as the axis runs. See Axis Info Select 1. Position Integrator Error in Position Units - mSec Position Integrator Error is the running sum of the Position Error, in the configured axis Position Units, for the specified axis. For an axis with an active servo loop, the position integrator error is used, along with other error terms, to drive the motor to the condition where the actual position is equal to the command position.
Position Lock Status	AXIS_SERVO AXIS_SERVO_DRIVE	BOOL	Tag	If this bit is: • ON — The axis position error is less than or equal to the Position Lock Tolerance value of the axis. • OFF — The axis position error is greater than the Position Lock Tolerance value of the axis.
Position Lock Tolerance	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV	Position Units The Position Lock Tolerance attribute value specifies how much position error the motion module tolerates when giving a true Position Locked Status indication. When used in conjunction with the Position Locked Status bit, it is a useful parameter to control positioning accuracy. The Position Lock Tolerance value should be set, in Position Units, to the desired positioning accuracy of the axis. Note that the position lock tolerance value is interpreted as a ± quantity. For example, if your position units are Inches, specifying a position lock tolerance of 0.01 provides a minimum positioning accuracy of ±0.01 inches as shown below. 
Position Polarity	AXIS_SERVO_DRIVE	INT	GSV	This attribute is derived from the Drive Polarity attribute. See IDN 55 in IEC 1491.

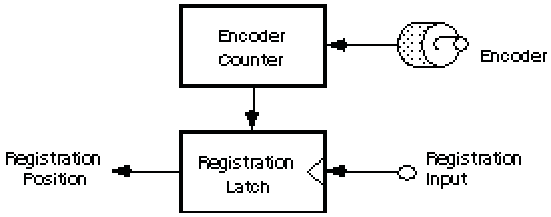
Attribute	Axis Type	Data Type	Access	Description
Position Proportional Gain	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV	1/Seconds This controller attribute is replicated in the motion module. The Position Error is multiplied by the Position Proportional Gain (Pos P Gain) to produce a component to the Velocity Command that tries to correct for the position error. Increasing this gain increases the bandwidth of the position servo loop and results in greater static stiffness of the axis, which is a measure of the corrective force that is applied to an axis for a given position error. Too little Pos P Gain results in excessively compliant, or mushy, axis behavior. Too large a Pos P Gain results in axis oscillation due to servo instability. A well-tuned system moves and stops quickly and shows little or no ringing during constant velocity or when the axis stops. If the response time is poor, or the motion sloppy or slow, you may need to increase the proportional gain. If excessive ringing or overshoot is observed when the motor stops, you may need to decrease the proportional gain. While the tuning procedure sets the Pos P Gain, you can also set it manually. You can compute the Pos P Gain based on either the desired loop gain or the desired bandwidth of the position servo system. **Loop Gain Method** If you know the desired loop gain in Inches per Minute per mil or millimeters per minute per mil, use the following formula to calculate the corresponding P gain. Pos P Gain = 16.667 * Desired Loop Gain (IPM/mil) A loop gain of 1 IPM/mil (Pos P gain = 16.7 Sec^{-1}) gives stable positioning for most axes. However, position servo systems typically run much tighter than this. The typical value for the Position Proportional Gain is ~100 Sec^{-1}. **Bandwidth Method** If you know the desired unity gain bandwidth of the position servo in Hertz, use the following formula to calculate the corresponding P gain. Pos P Gain = Bandwidth (Hertz) / 6.28 Position servo systems typically run with at least a unity gain bandwidth of ~16 Hertz. The typical value for the Position Proportional Gain is ~100 Sec^{-1}. **Maximum Bandwidth** There are limitations to the maximum bandwidth that can be achieved for the position loop based on the dynamics of the inner velocity and torque loops of the system and the desired damping of the system, Z. These limitations may be expressed as follows: Bandwidth (Pos) = 0.25 * 1/Z^2 * Bandwidth (Vel) = 0.25 * 1/Z^2 * Bandwidth (Torque) For example, if the bandwidth of the drive's torque loop is 100 Hz and the damping factor, Z, is 0.8, the velocity bandwidth is approximately 40 Hz and the position bandwidth is 16 Hz. Based on these numbers the corresponding proportional gains for the loops can be computed. Note that the bandwidth of the torque loop includes feedback sampling delay and filter time constant.

Attribute	Axis Type	Data Type	Access	Description
Position Servo Bandwidth	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV	Hertz The value for the Position Servo Bandwidth represents the unity gain bandwidth that is to be used to calculate the gains for a subsequent MAAT (Motion Apply Axis Tune) instruction. The unity gain bandwidth is the frequency beyond which the position servo is unable to provide any significant position disturbance correction. In general, within the constraints of a stable servo system, the higher the Position Servo Bandwidth is the better the dynamic performance of the system. A maximum value for the Position Servo Bandwidth is generated by the MRAT (Motion Run Axis Tune) instruction. Computing gains based on this maximum value via the MAAT instruction results in dynamic response in keeping with the current value of the Damping Factor described above. Alternatively, the responsiveness of the system can be 'softened' by reducing the value of the Position Servo Bandwidth before executing the MAAT instruction.. There are limitations to the maximum bandwidth that can be achieved for the position loop based on the dynamics of the inner velocity and current loops of the servo system and the desired damping of the system, Z. Exceeding these limits could result in an unstable system. These bandwidth limitations may be expressed as follows: Max Position Bandwidth (Hz) = $0.25 * 1/Z^2 *$ Velocity Bandwidth (Hz) For example, if the maximum bandwidth of the velocity servo loop is 40 Hz and the damping factor, Z, is 0.8, the maximum the maximum position bandwidth is 16 Hz. Based on these numbers the corresponding proportional gains for the loops can be computed.
Position Units	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	STRING	MSG	Fixed length string of 32 characters The Position Units attribute can support an ASCII text string of up to 32 characters. This string is used by RSLogix 5000 software in the axis configuration dialog boxes to request values for motion-related parameters in the specified Position Units.

Attribute	Axis Type	Data Type	Access	Description
Position Unwind	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	DINT	GSV SSV	Counts per Revolution This controller attribute is replicated in the motion module. If the axis is configured as a rotary axis by setting the corresponding Rotary Axis bit Servo Configuration Bit word, a value for the Position Unwind attribute is required. This is the value used to perform automatic electronic unwind of the rotary axis. Electronic unwind allows infinite position range for rotary axes by subtracting the unwind value from both the actual and command position every time the axis makes a complete revolution. To avoid accumulated error due to round-off with irrational conversion constants the unwind value is requested in units feedback counts per axis revolution and must be an integer. For example, suppose that a given axis is configured as a Rotary Axis with Position Units of 'Degrees' and 10 feedback counts per degree. It is desired to unwind the axis position after every revolution. In this case, the Position Unwind attribute should be set to 3600 since there are 3600 feedback counts (10 * 360) per revolution of the axis.
Positive Dynamic Torque Limit	AXIS_SERVO_DRIVE	REAL	GSV Tag	**Important**: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won't see the right value as the axis runs. See Axis Info Select 1. %Rated The currently operative maximum positive torque/current limit magnitude. The value should be the lowest value of all torque/current limits in the drive at a given time. This limit includes the amplifier peak limit, motor peak limit, user current limit, amplifier thermal limit, and the motor thermal limit.
Power Capacity	AXIS_SERVO_DRIVE	REAL	GSV Tag	**Important**: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won't see the right value as the axis runs. See Axis Info Select 1. The present utilization of the axis power supply as a percent of rated capacity.
Power Limit Status	AXIS_SERVO_DRIVE	BOOL	Tag	Set when the magnitude of the actual supplied power is greater than the configured Power Threshold.
Power Phase Loss Fault	AXIS_SERVO_DRIVE	BOOL	Tag	Set when the drive detects that one or more of the three power line phases is lost from the 3 phase power inputs.
Power Supply ID	AXIS_SERVO_DRIVE	INT	GSV	The Power Supply ID attribute contains the enumeration of the specific A-B Power Supply or System Module catalog numbers associated with the axis. If the Power Supply ID does not match that of the actual supply hardware, an error is generated during the drive configuration process.
Precharge Overload Fault	AXIS_SERVO_DRIVE	BOOL	Tag	The drive's pre-charge resistor gets too hot if you cycle 3-phase power too many times. If that happens, this bit turns on.
Primary Operation Mode	AXIS_SERVO_DRIVE	INT	GSV	This attribute is derived from the Servo Loop Configuration attribute. See IDN 32 in IEC 1491.

Attribute	Axis Type	Data Type	Access	Description
Process Status	AXIS_SERVO AXIS_SERVO_DRIVE	BOOL	Tag	Set when there is an axis tuning operation or an axis hookup diagnostic test operation in progress on the axis.
Programmed Stop Mode	AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	SINT	GSV SSV	Determines how a specific axis will stop when the controller has a critical controller mode change or when an MGS (Motion Group Stop) instruction executes with it's stop mode set to Programmed. The modes of the controller are: Program Mode, Run Mode, Test Mode, and Faulted Mode. Any mode change into or out of program mode (prog->run, prog->test, run->prog & test->prog) will initiate a programmed stop for every axis owned by that controller. Each individual axis can have its own Programmed Stop Mode configuration independent of other axes. **Fast Stop (default) = 0** When the Programmed Stop Mode attribute is configured for Fast Stop, the axis is decelerated to a stop using the current configured value for Maximum Deceleration. Servo action is maintained after the axis motion has stopped. **Fast Disable = 1** When the Programmed Stop Mode attribute is configured for Fast Disable, the axis is decelerated to a stop using the current configured value for Maximum Deceleration. Servo action is maintained until the axis motion has stopped at which time the axis is disabled, that is, Drive Enable disabled, and Servo Action disabled **Hard Disable = 2** When configured for Hard Disable, the axis is immediately disabled, that is, Drive Enable disabled, Servo Action disabled, but the OK contact is left closed. Unless the drive is configured to provide some form of dynamic breaking, this results in the axis coasting to a stop. **Fast Shutdown = 3** When configured for Fast Shutdown, the axis is decelerated to a stop as with Fast Stop but, once the axis motion is stopped, the axis is placed in the Shutdown state, that is, Drive Enable disabled, servo action disabled, and the OK contact opened. To recover from the Shutdown state requires execution of one of the axis or group Shutdown Reset instructions (MASR or MGSR). **Hard Shutdown = 4** When configured for Hard Shutdown, the axis is immediately placed in the Shutdown state, that is, Drive Enable disabled, Servo Action disabled, and the OK contact opened. Unless the drive is configured to provide some form of dynamic breaking, this results in the axis coasting to a stop. To recover from the Shutdown state requires execution of one of the axis or group Shutdown Reset instructions (MASR or MGSR).

Attribute	Axis Type	Data Type	Access	Description
PWM Frequency Select	AXIS_SERVO_DRIVE	SINT	GSV	The PWM Frequency Select attribute controls the frequency of the pulse width modulated voltage applied to the motor by the drive's power structure. Higher PWM Frequency values reduce torque ripple and motor noise based on the motor's electrical time constant. Higher PWM frequencies, however, mean higher switching frequencies, which tends to produce more heat in the drive's power structure. So, for applications that have high torque demands, a lower PWM frequency would be more appropriate. 0 = low frequency (default) 1 = high frequency
Reg 1 Input Status	AXIS_SERVO AXIS_SERVO_DRIVE	BOOL	Tag	If this bit is: • ON — Registration 1 input is active. • OFF — Registration 1 input is inactive.
Reg 2 Input Status	AXIS_SERVO AXIS_SERVO_DRIVE	BOOL	Tag	If this bit is: • ON — Registration 2 input is active. • OFF — Registration 2 input is inactive.
Reg Event 1 Armed Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set when a registration checking has been armed for registration input 1 through execution of the MAR (Motion Arm Registration) instruction. Cleared when either a registration event occurs or a MDR (Motion Disarm Registration) instruction is executed for registration input 1.
Reg Event 1 Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set when a registration event has occurred on registration input 1. Cleared when either another MAR (Motion Arm Registration) instruction or a MDR (Motion Disarm Registration) instruction is executed for registration input 1.
Reg Event 2 Armed Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set when a registration checking has been armed for registration input 2 through execution of the MAR (Motion Arm Registration) instruction. Cleared when either a registration event occurs or a MDR (Motion Disarm Registration) instruction is executed for registration input 2.
Reg Event 2 Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set when a registration event has occurred on registration input 2. Cleared when either another MAR (Motion Arm Registration) instruction or a MDR (Motion Disarm Registration) instruction is executed for registration input 2.
Registration 1 Position	AXIS_CONSUMED AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	Tag	Registration 1 Position in Position Units

Attribute	Axis Type	Data Type	Access	Description
Registration 1 Event Task Registration 2 Event Task	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	DINT	MSG	These attributes show which task is triggered when the registration event happens. - An instance of 0 means that no event task is configured to be triggered by the registration event. - The task is triggered at the same time that the Process Complete bit is set for the instruction that armed the watch event. - The controller sets these attributes. do not set them by an external device.
Registration 1 Position Registration 2 Position	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV Tag	Position Units Two registration position attributes are provided to independently store axis position associated with two different registration input events. The Registration Position value is the absolute position of a physical or virtual axis (in the position units of that axis) at the occurrence of the most recent registration event for that axis. The figure below shows how the registration position is latched by the registration input when a registration event occurs. The latching mechanism can be implemented in the controller software (soft registration) or, for greater accuracy, in physical hardware (hard registration).  The Registration Latch mechanism is controlled by two Event Control instructions, MAR (Motion Arm Registration) and MDR (Motion Disarm Registration). The accuracy of the registration position value, saved as a result of a registration event, is a function of the delay in recognizing the specified transition (typically 1 μsec for hardware registration) and the speed of the axis during this time. The uncertainty in the registration position is the distance traveled by the axis during this interval as shown by the equation. $\text{Uncertainty} = \text{Axis Speed}\left[\frac{\text{Position Units}}{\text{Second}}\right] \text{x Delay}$ Use the formula given above to calculate the maximum registration position error for the expected axis speed. Alternatively, you can calculate the maximum axis speed for a specified registration accuracy by re-arranging this formula as shown. $\text{Maximum Speed}\left[\frac{\text{Position Units}}{\text{Second}}\right] = \frac{\text{Desired Accuracy [Position Units]}}{\text{Delay}}$

Attribute	Axis Type	Data Type	Access	Description
Registration 1 Time Registration 2 Time	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	DINT	GSV Tag	Lower 32 bits of CST time The two Registration Time values contain the lower 32-bits of CST time at which their respective registration events occurred. Units for this attribute are in microseconds.

Attribute	Axis Type	Data Type	Access	Description
Resistive Brake Contact Delay	AXIS_SERVO_DRIVE	REAL	GSV SSV	Sec This attribute controls an optional external Resistive Brake Module (RBM). The RBM is between the drive and the motor and uses an internal contactor to switch the motor between the drive and a resistive load. The drive's RBM output controls this contactor. When the drive's RBM output is energized, the RBM contactor is switched from the load resistors to the UVW motor lines connecting the drive to the motor. This switching does not occur instantaneously and enabling the power structure too early can cause electrical arcing across the contactor. The resistive brake contact delay is the time that it takes to fully close the contactor across the UVW motor lines. In order to prevent electrical arcing across the contactor the enabling of the drive's power structure is delayed. The delay time is variable depending on the RBM model. When applying an RBM, you must set the Resistive Brake Contact Delay to the recommended value found in the RBM specification. The following cases outline how the RBM output relates to the normal enable and disable sequences. **Case 1 – Enable Sequence** 1. Enable axis is initiated via MSO or MAH instruction. 2. Turn on RBM output to connect motor to drive. 3. Wait for Resistive Brake Contact Delay while RBM contacts close. 4. Drive power structure enabled (Drive Enable Status bit is set). 5. Turn on motor brake output to release brake. 6. Wait Brake Release Delay Time while motor brake releases. 7. Track Command reference (Servo Action Status bit is set). **Case 2 – Disable - Category 1 Stop** 1. Disable axis is initiated via an MSF instruction or a drive disable fault action. 2. Drive stops tracking command reference (Servo Action Status bit is cleared). 3. Apply Stopping Torque to stop motor. 4. Wait for zero speed or Stopping Time Limit. 5. Turn off brake output to engage motor brake. 6. Wait for Brake Engage delay while motor brake engages. 7. Disable drive power structure (Drive Enable Status bit is cleared). 8. Turn off RBM output to disconnect motor from drive. **Case 3 – Shutdown Category 0 Stop** 1. Drive stops tracking command reference (Servo Action Status bit is cleared). 2. Disable drive power structure (Drive Enable Status bit is cleared). 3. Turn off brake output to engage brake. 4. Turn off RBM output to disconnect motor from drive.

Attribute	Axis Type	Data Type	Access	Description
Rotary Axis	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	SINT	GSV SSV* * Can only be set if axis is not enabled.	0 = Linear 1 = Rotary When the Rotary Axis attribute is set true (1), it lets the axis unwind. This gives infinite position range by unwinding the axis position whenever the axis moves through a complete physical revolution. The number of encoder counts per physical revolution of the axis is specified by the Position Unwind attribute. For Linear operation, the counts do not roll over. They are limited to +/- 2 billion.
Rotary Motor Inertia	AXIS_SERVO_DRIVE	REAL	SSV	The Rotary Motor Inertia attribute is a float that specifies the unloaded inertia of a rotary motor.
Rotary Motor Rated Speed	AXIS_SERVO_DRIVE	REAL	GSV	The Rotary Motor Rated Speed attribute is a float that specifies the nameplate rated speed of a rotary motor. For PM motors, this is generally specified at rated voltage based on either rated current, rated torque, or rated power. For induction motors this value is the speed of the motor driven at rated frequency under rated torque load. This value is synonymous with the term base speed.
Safe-Off Mode Active Status	AXIS_SERVO_DRIVE	BOOL	GSV Tag	This bit is the status indication of the Kinetix Drive's Safe-Off circuitry. If this bit is the following state: • ON - The Drive's Safety monitor circuitry has encountered a loss of signal from Enable_1 or Enable_2. • OFF - The Drive's Safety monitor circuitry has no fault from Enable_1 or Enable_2. For the Kinetix Drive to pass back this status to the controller via this bit, the Drive must have firmware version 1.85 or higher.
SERCOS Error Code	AXIS_SERVO_DRIVE	INT	GSV* Tag	Error code returned by SERCOS module indicating source of drive parameter update failure. The SERCOS Error Code value can be used to identify the source of the drive parameter update failure that resulted in the Axis Configuration Fault.
SERCOS Fault	AXIS_SERVO_DRIVE	BOOL	Tag	Set when either a requested SERCOS procedure fails to execute properly or the associated drive node has detected a SERCOS communication fault.
SERCOS Ring Fault	AXIS_SERVO_DRIVE	BOOL	Tag	If this bit is set, there is a problem on the SERCOS ring; that is, the light has been broken or a drive has been powered down.
Servo Action Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	If this bit is: • ON — The axis is under servo control. • OFF — Servo action is disabled.

<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Servo Fault</td><td>AXIS_SERVO</td><td>DINT</td><td>Tag</td><td>Lets you access all the servo fault bits in one 32-bit word. This tag is the same as the Servo Fault Bits attribute.
<table>
<tr><th>Servo Fault</th><th>Bit</th></tr>
<tr><td>Pos Soft Overtravel Fault</td><td>0</td></tr>
<tr><td>Neg Soft Overtravel Fault</td><td>1</td></tr>
<tr><td>Reserved</td><td>2</td></tr>
<tr><td>Reserved</td><td>3</td></tr>
<tr><td>Feedback Fault</td><td>4</td></tr>
<tr><td>Feedback Noise Fault</td><td>5</td></tr>
<tr><td>Reserved</td><td>6</td></tr>
<tr><td>Reserved</td><td>7</td></tr>
<tr><td>Position Error Fault</td><td>8</td></tr>
<tr><td>Drive Fault</td><td>9</td></tr>
</table>
These fault bits are updated every coarse update period.
Do you want any of these faults to give the controller a major fault?
• YES — Set the General Fault Type of the motion group = Major Fault.
• NO — You must write code to handle these faults.</td></tr>
<tr><td>Servo Fault Bits</td><td>AXIS_SERVO</td><td>DINT</td><td>GSV*</td><td>Lets you access all the servo fault bits in one 32-bit word. This attribute is the same as the Servo Fault tag.
<table>
<tr><th>Servo Fault</th><th>Bit</th></tr>
<tr><td>Pos Soft Overtravel Fault</td><td>0</td></tr>
<tr><td>Neg Soft Overtravel Fault</td><td>1</td></tr>
<tr><td>Reserved</td><td>2</td></tr>
<tr><td>Reserved</td><td>3</td></tr>
<tr><td>Feedback Fault</td><td>4</td></tr>
<tr><td>Feedback Noise Fault</td><td>5</td></tr>
<tr><td>Reserved</td><td>6</td></tr>
<tr><td>Reserved</td><td>7</td></tr>
<tr><td>Position Error Fault</td><td>8</td></tr>
<tr><td>Drive Fault</td><td>9</td></tr>
</table>
These fault bits are updated every coarse update period.
Do you want any of these faults to give the controller a major fault?
• YES — Set the General Fault Type of the motion group = Major Fault.
• NO — You must write code to handle these faults.</td></tr>
</table>

Attribute	Axis Type	Data Type	Access	Description
Servo Feedback Type	AXIS_SERVO	SINT	GSV	This attribute provides a selection for the Feedback Type. 0 = A Quadrature B (AQB) 1 = Synchronous Serial Interface (SSI) 2 = Linear Displacement Transducer (LDT) **A Quadrature B Encoder Interface (AQB)** Servo modules, such as the 175-6M02AE, provide interface hardware to support incremental quadrature encoders equipped with standard 5-Volt differential encoder interface signals. This interface hardware provides a robust differential encoder input interface to condition each of the encoder signals before being applied to an Encoder-to-Digital Converter (EDC) FPGA. The EDC decodes the encoder signals and uses a 16-bit bidirectional counter to accumulate feedback counts. A regular Timer Event signal, applied to the EDC, latches the encoder counters for all axes simultaneously. This same Timer Event signal also triggers the servo interrupt service routine that performs the servo loop computations. One of the first things done by the interrupt service routine is to read the latched encoder counter values from the EDC. The change in the encoder counter value from the last timer event is computed and this delta value is added to a 32-bit signed integer position accumulator, which represents the Actual Position of the axis. The Actual Position value is used as feedback to the position servo loop and as input to the Watch Event Handler. The delta position value represents velocity feedback, which when configured to do so, may be filtered and applied to the inner velocity servo loop. **Synchronous Serial Interface (SSI)** Some servo modules, like the 1756-M02AS, provide an interface to transducers with Synchronous Serial Interface (SSI) outputs. SSI outputs use standard 5V differential signals (RS422) to transmit information from the transducer to the controller. The signals consist of a Clock generated by the controller and Data generated by the transducer. Each transducer with an SSI output provides output data of a specified number of bits of either Binary or Gray code data. The controller must generate a stream of clock pulses with the correct number of bits and a frequency within the range supported by the transducer. The servo module can be configured via the Servo Axis Object to generate any number of clock pulses between 8 and 32, and the frequency can be set to either 208kHz or 650kHz. The clock signal is maintained in the High state between pulse strings. The transducer shifts data out on the Data line MSB first on each rising edge of the clock signal. The transducer also maintains the data signal in specified states before and after the data is shifted out. These states are checked by the controller to detect missing transducers or broken wires. A Field Programmable Gate Array (FPGA) is used to implement a multi-channel SSI Interface on the controller. Each channel is functionally equivalent. ***Continued on next page***

Attribute	Axis Type	Data Type	Access	Description
Servo Feedback Type (continued)				**Linear Displacement Transducer (LDT)** Servo modules like the 1756-HYD02 use the Linear Magnetostrictive Displacement Transducer, or LDT. A Field Programmable Gate Array (FPGA) is used to implement a multi-channel LDT Interface. Each channel is functionally equivalent and is capable of interfacing to an LDT device with a maximum count of 240,000. The LDT interface has transducer failure detection and digital filtering to reduce electrical noise. The FPGA can interface to two types of LDTs: Start/Stop and PWM. Start/Stop transducers accept an input (interrogate) signal to start the measurement cycle and respond with two pulses on the Return line. The time between the pulses is proportional to the position. PWM transducers respond to the interrogate signal with a single long pulse on the Return line. The pulse width is proportional to the position. The FPGA generates the Interrogate signal every Servo Update time and measures the time between the Start/Stop pulses or the PWM pulse width. The resolution of the position measurement is determined by the frequency of the clock used for the time measurement. In the 1756-HYD02 design, a 60 MHz clock is used, and both edges of the clock signal are used for an effective time resolution of 8.3 nanoseconds. This translates into a position resolution better than 0.001 inch. Note: It is possible to achieve higher resolutions with PWM transducers that are configured to perform multiple internal measurements (recirculations) and report the sum of those measurements in the pulse width.
Servo Loop Configuration	AXIS_SERVO AXIS_SERVO_DRIVE	INT	GSV SSV	The Servo Loop Configuration attribute determines the specific configuration of the servo loop topology when the axis is set to 'servo'. 0 = custom 1 = feedback only 2 = aux. feedback only 3 = position servo 4 = aux. position servo 5 = dual position servo 6 = dual command servo 7 = aux. dual command servo 8 = velocity servo 9 = torque servo 10 = dual command/feedback servo
Servo Output Level	AXIS_SERVO	REAL	GSV Tag	**Important**: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won't see the right value as the axis runs. See Axis Info Select 1. Servo Output Level in Volts Servo Output Level is the current voltage level of the servo output of the specified axis. The Servo Output Level can be used in drilling applications, for example, where the servo module is interfaced to an external Torque Loop Servo Drive, to detect when the drill bit has engaged the surface of the work piece.

Attribute	**Axis Type**	**Data Type**	**Access**	**Description**
Servo Polarity Bits	AXIS_SERVO	DINT	GSV	0 = Feedback Polarity Negative 1 = Servo Polarity Negative **Feedback Polarity Negative** This Feedback Polarity Negative bit attribute controls the polarity of the encoder feedback and, when properly configured, insures that when the axis is moved in the user defined positive direction that the axis Actual Position value increases. This bit can be configured automatically using the MRHD and MAHD motion instructions. **Servo Polarity Negative** This Servo Polarity Negative bit attribute controls the polarity of the servo output to the drive. When properly configured along with the Feedback Polarity Negative bit, it insures that when the axis servo loop is closed that it is closed as a negative feedback system and not an unstable positive feedback system. This bit can be configured automatically using the MRHD and MAHD motion instructions.
Servo Status	AXIS_SERVO	DINT	Tag	Lets you access the status bits for your servo loop in one 32-bit word. This tag is the same as the Servo Status Bits attribute.

Servo Status	**Bit**
Servo Action Status	0
Drive Enable Status	1
Shutdown Status	2
Process Status	3
Output Limit Status	4
Position Lock Status	5
Home Input Status	6
Reg 1 Input Status	7
Reg 2 Input Status	8
Resevered	9
Resevered	10
Drive Fault Input Status	11

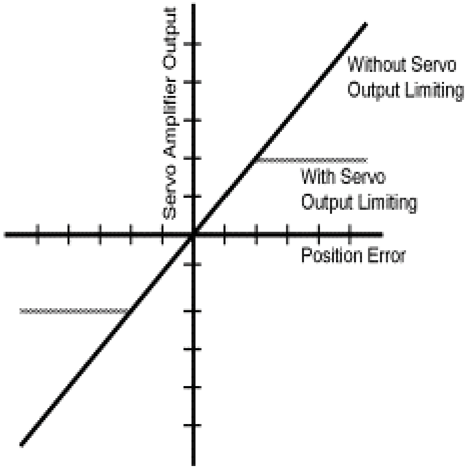
<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Servo Status Bits</td><td>AXIS_SERVO</td><td>DINT</td><td>GSV*</td><td>Lets you access the status bits for your servo loop in one 32-bit word. This attribute is the same as the Servo Status tag.
<table>
<tr><th>Servo Status</th><th>Bit</th></tr>
<tr><td>Servo Action Status</td><td>0</td></tr>
<tr><td>Drive Enable Status</td><td>1</td></tr>
<tr><td>Shutdown Status</td><td>2</td></tr>
<tr><td>Process Status</td><td>3</td></tr>
<tr><td>Output Limit Status</td><td>4</td></tr>
<tr><td>Position Lock Status</td><td>5</td></tr>
<tr><td>Home Input Status</td><td>6</td></tr>
<tr><td>Reg 1 Input Status</td><td>7</td></tr>
<tr><td>Reg 2 Input Status</td><td>8</td></tr>
<tr><td>Resevered</td><td>9</td></tr>
<tr><td>Resevered</td><td>10</td></tr>
<tr><td>Drive Fault Input Status</td><td>11</td></tr>
</table></td></tr>
<tr><td>Shutdown Status</td><td>AXIS_CONSUMED
AXIS_GENERIC
AXIS_SERVO
AXIS_SERVO_DRIVE
AXIS_VIRTUAL</td><td>BOOL</td><td>Tag</td><td>If this bit is:
• ON — The axis is in the Shutdown state.
• OFF — The axis isn’t in the Shutdown state.</td></tr>
<tr><td>Soft Overtravel Fault Action</td><td>AXIS_SERVO
AXIS_SERVO_DRIVE</td><td>SINT</td><td>GSV
SSV</td><td>
<table>
<tr><th>Fault Action</th><th>Value</th></tr>
<tr><td>Shutdown</td><td>0</td></tr>
<tr><td>Disable Drive</td><td>1</td></tr>
<tr><td>Stop Motion</td><td>2</td></tr>
<tr><td>Status Only</td><td>3</td></tr>
</table>
This controller attribute is replicated in the motion module.</td></tr>
<tr><td>SSI Clock Frequency</td><td>AXIS_SERVO</td><td>SINT</td><td>GSV</td><td>0 = 208 kHz
1 = 650 kHz
This attribute provides for setting the Clock Frequency in kHz of the SSI device. This attribute is only active if the Transducer Type is set to SSI.</td></tr>
</table>

Attribute	Axis Type	Data Type	Access	Description
SSI Code Type	AXIS_SERVO	SINT	GSV	0 = Binary 1 = Gray This attribute provides for setting the whether the SSI device is using Binary or Gray code. This attribute is only active if the Transducer Type is set to SSI.
SSI Data Length	AXIS_SERVO	SINT	GSV	This attribute provides for setting the data length of the SSI device. This attribute is only active if the Transducer Type is set to SSI.
Start Actual Position	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV Tag	Start Actual Position in Position Units Whenever a new motion planner instruction starts for an axis (for example, using a MAM instruction), the value of the axis command position and actual position is stored at the precise instant the motion begins. These values are stored as the Start Command Position and Start Actual Position respectively in the configured Position Units of the axis. Start Positions are useful to correct for any motion occurring between the *detection* of an event and the action *initiated* by the event. For instance, in coil winding applications, Start Command Positions can be used in an expression to compensate for overshooting the end of the bobbin before the gearing direction is reversed. If you know the position of the coil when the gearing direction was *supposed* to change, and the position at which it *actually* changed (the Start Command Position), you can calculate the amount of overshoot, and use it to correct the position of the wire guide relative to the bobbin.
Start Command Position	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV Tag	Start Command Position in Position Units Whenever a new motion planner instruction starts for an axis (for example, using a MAM instruction), the value of the axis command position and actual position is stored at the precise instant the motion begins. These values are stored as the Start Command Position and Start Actual Position respectively in the configured Position Units of the axis. Start Positions are useful to correct for any motion occurring between the detection of an event and the action initiated by the event. For instance, in coil winding applications, Start Command Positions can be used in an expression to compensate for overshooting the end of the bobbin before the gearing direction is reversed. If you know the position of the coil when the gearing direction was supposed to change, and the position at which it actually changed (the Start Command Position), you can calculate the amount of overshoot, and use it to correct the position of the wire guide relative to the bobbin.
Start Master Offset	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV Tag	Start Master Offset in Master Position Units The Start Master Offset is the position offset that was applied to the master side of the position cam when the last Motion Axis Move (MAM) instruction with the move type set to 'Absolute Master Offset' or 'Incremental Master Offset' was executed. The Start Master Offset is returned in master position units. The Start Master Offset will show the same unwind characteristic as the position of a linear axis.

Attribute	Axis Type	Data Type	Access	Description
Stopping Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set if there is a stopping process currently in progress. Cleared when the stopping process is complete. The stopping process is used to stop an axis (initiated by an MAS, MGS, Stop Motion fault action, or mode change).
Stopping Time Limit	AXIS_SERVO_DRIVE	REAL	GSV SSV	Sec This attribute maps directly to a SERCOS IDN. See the SERCOS Interface standard for a description. This attribute is automatically set. You usually do not have to change it.
Stopping Torque	AXIS_SERVO_DRIVE	REAL	GSV SSV	% Rated This attribute maps directly to a SERCOS IDN. See the SERCOS Interface standard for a description. This attribute is automatically set. You usually do not have to change it.
Strobe Actual Position	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV Tag	Strobe Actual Position in Position Units Strobe Actual Position, and Strobe Command Position are used to simultaneously store a snap-shot of the actual, command position and master offset position of an axis when the MGSP (Motion Group Strobe Position) instruction is executed. The values are stored in the configured Position Units of the axis. Since the MGSP instruction simultaneously stores the actual and command positions for all axes in the specified group of axes, the resultant Strobe Actual Position and Strobe Command Position values for different axes can be used to perform real time calculations. For example, the Strobe Actual Positions can be compared between two axis to provide a form of 'slip compensation' in web handling applications.
Strobe Command Position	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV Tag	Strobe Command Position in Position Units Strobe Actual Position, and Strobe Command Position are used to simultaneously store a snap-shot of the actual, command position and master offset position of an axis when the MGSP (Motion Group Strobe Position) instruction is executed. The values are stored in the configured Position Units of the axis. Since the MGSP instruction simultaneously stores the actual and command positions for all axes in the specified group of axes, the resultant Strobe Actual Position and Strobe Command Position values for different axes can be used to perform real time calculations. For example, the Strobe Actual Positions can be compared between two axis to provide a form of 'slip compensation' in web handling applications.
Strobe Master Offset	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV Tag	Strobe Master Offset in Master Position Units The Strobe Master Offset is the position offset that was applied to the master side of the position cam when the last Motion Group Strobe Position (MGSP) instruction was executed. The Strobe Master Offset is returned in master position units. The Strobe Master Offset will show the same unwind characteristic as the position of a linear axis.
Telegram Type	AXIS_SERVO_DRIVE	INT	GSV	Set to a value of 7, which means Application Telegram. See IDN 15 in IEC 1491.

<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Test Direction Forward</td><td>AXIS_SERVO
AXIS_SERVO_DRIVE</td><td>SINT</td><td>GSV</td><td>The direction of axis travel during the last hookup test initiated by a MRHD (Motion Run Hookup Test) instruction.
0 = reverse
1 = forward (positive)
<table>
<tr><th>For this Data type</th><th>Details</th></tr>
<tr><td>AXIS_SERVO</td><td>This value does not depend on the Servo Polarity Bits attribute. The MAHD (Motion Apply Hookup Test) instruction uses the Test Direction Forward attribute and the Test Output Polarity attribute to set the Servo Polarity Bits attribute for negative feedback and correct directional sense.</td></tr>
<tr><td>AXIS_SERVO_DRIVE</td><td>This value does not depend on the Drive Polarity attribute. The MAHD (Motion Apply Hookup Test) instruction uses the Test Direction Forward attribute and the Test Output Polarity attribute to set the Drive Polarity attribute for the correct directional sense.</td></tr>
</table></td></tr>
<tr><td>Test Increment</td><td>AXIS_SERVO
AXIS_SERVO_DRIVE</td><td>REAL</td><td>GSV
SSV</td><td>Position Units
The Motor Feedback Test Increment attribute is used in conjunction with the MRHD (Motion Run Hookup Diagnostic) instruction to determine the amount of motion that is necessary to satisfy the MRHD initiated test process. This value is typically set to approximately a quarter of a revolution of the motor..</td></tr>
<tr><td>Test Status</td><td>AXIS_SERVO
AXIS_SERVO_DRIVE</td><td>INT</td><td>GSV</td><td>0 = test process successful
1 = test in progress
2 = test process aborted by user
3 = test process time-out fault (~2 seconds)
4 = test failed – servo fault
5 = test failed – insufficient test increment
More for AXIS_SERVO_DRIVE data type…
6 = test failed – wrong polarity
7 = test failed – missing signal
8 = test failed – device comm error
9 = test failed – feedback config error
10 = test failed – motor wiring error
This attribute returns the status of the last run MRHD (Motion Run Hookup Diagnostic) instruction that initiates a hookup diagnostic process on the axis. Use this attribute to determine when the MRHD initiated operation has successfully completed. Conditions may occur, however, that make it impossible to properly perform the operation. When that happens, the test process is automatically aborted and a test fault reported that is stored in the Test Status output parameter.</td></tr>
</table>

Attribute	**Axis Type**	**Data Type**	**Access**	**Description**
Time Cam Pending Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set if a Time Cam motion profile is currently pending the completion of a currently executing cam profile. This would be initiated by executing an MATC instruction with Pending execution selected. This bit is cleared when the current time cam profile completes, initiating the start of the pending cam profile. This bit is also cleared if the time cam profile completes, or is superseded by some other motion operation.
Time Cam Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set if a Time Cam motion profile is currently in progress. Cleared when the Time Cam is complete or is superseded by some other motion operation.
Timer Event Fault	AXIS_SERVO AXIS_SERVO_DRIVE	BOOL	Tag	If this bit is set, the motion module has a problem with its timer event that synchronizes the module's servo loop to the master timebase of the chassis (that is, Coordinated System Time). To clear this bit, reconfigure the motion module.
Torque Command	AXIS_SERVO_DRIVE	REAL	GSV Tag	**Important**: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won't see the right value as the axis runs. See Axis Info Select 1. %Rated The command when operating in Torque Mode in terms of % rated.
Torque Data Scaling	AXIS_SERVO_DRIVE	INT	GSV	This 16-bit attribute displays the scaling method to use on torque values (for example, Torque Command Value, and Bipolar torque limit value) with decimal values ranging from 0 to 127. Bit values are: • Bits 2-0: Scaling method - 000 – percentage scaling - 001 – linear scaling (force) - 010 – rotational scaling (torque) • Bit 3: - 0 – preferred scaling - 1 – parameter scaling • Bit 4: Units - 0 – Newton meter (Nm) - 1 – inch pound force (lbf) • Bit 5: (reserved) • Bit 6: Data reference - 0 – at the motor shaft - 1 – at the load All other bits are reserved

Attribute	Axis Type	Data Type	Access	Description
Torque Data Scaling Exp	AXIS_SERVO_DRIVE	INT	GSV	This 16-bit unsigned attribute displays the scaling exponent for all torque data in a drive, with decimal values ranging from -2^15 to (2^15)-1. Bit values are: • Bit 14-0: Exponent value • Bit 15: Exponent sign: - 0 - Positive - 1 - Negative
Torque Data Scaling Factor	AXIS_SERVO_DRIVE	DINT	GSV	This 16-bit unsigned attribute displays the scaling factor for all torque data in a drive, with decimal values ranging from 1 to (2^16)-1.
Torque Feedback	AXIS_SERVO_DRIVE	REAL	GSV Tag	**Important**: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won't see the right value as the axis runs. See Axis Info Select 1. %Rated The torque feedback when operating in Torque Mode in terms of % rated.
Torque Limit Bipolar	AXIS_SERVO_DRIVE	REAL	GSV SSV	%Rated The Torque Limit attribute provides a method of limiting the maximum command current/torque to the motor to a specified level in terms of the motor's continuous current/torque rating. The output of the servo drive to the motor as a function of position servo error, both with and without servo torque limiting, is shown below.  The torque limit specifies the maximum percentage of the motors rated current that the drive can command as either positive or negative torque. For example, a torque limit of 150% shall limit the current delivered to the motor to 1.5 times the continuous current rating of the motor.

Attribute	Axis Type	Data Type	Access	Description
Torque Limit Negative	AXIS_SERVO_DRIVE	REAL	GSV SSV	%Rated This attribute maps directly to a SERCOS IDN. See the SERCOS Interface standard for a description. This attribute is automatically set. You usually do not have to change it.
Torque Limit Positive	AXIS_SERVO_DRIVE	REAL	GSV SSV	%Rated This attribute maps directly to a SERCOS IDN. See the SERCOS Interface standard for a description. This attribute is automatically set. You usually do not have to change it.
Torque Limit Source	AXIS_SERVO_DRIVE	DINT	GSV Tag	**Important**: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won’t see the right value as the axis runs. See Axis Info Select 1. This parameter displays the present source (if any) of any torque limiting for the axis. 0 = Not Limited 1 = Neg.e Torque Limit 2 = Pos. Torque Limit 3 = Amp Peak Limit 4 = Amp I(t) Limit 5 = Bus Regulator Limit 6 = Bipolar Torque Limit 7 = Motor Peak Limit 8 = Motor I(t) Limit 9 = Voltage Limit
Torque Limit Status	AXIS_SERVO_DRIVE	BOOL	Tag	Set when the magnitude of the axis torque command is greater than the configured Torque Limit.
Torque Offset	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV Tag	Torque Offset from –100% to +100% Torque Offset compensation can be used to provide a dynamic torque command correction to the output of the velocity servo loop. Since this value is updated synchronously every Coarse Update Period, the Torque Offset can be tied into custom outer control loop algorithms using Function Block programming.
Torque Polarity	AXIS_SERVO_DRIVE	INT	GSV	G10/Configuration It maps directly to the SERCOS IDN. It is automatically set based on the current Drive Polarity Settings. All command bits are set according to the Command polarity bit value and all feedback bits are set according to the Feedback Polarity bit setting.

Attribute	**Axis Type**	**Data Type**	**Access**	**Description**
Torque Scaling	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV	% / Position Units Per Second^2 This controller attribute is replicated in the motion module. The Torque Scaling attribute is used to convert the acceleration of the servo loop into equivalent % rated torque to the motor. This has the effect of 'normalizing' the units of the servo loop's gain parameters so that their values are not affected by variations in feedback resolution, drive scaling, motor and load inertia, and mechanical gear ratios. In fact, the Torque Scaling value, when properly established, represents the inertia of the system and is related to the Tune Inertia attribute value by a factor of the Conversion Constant. • AXIS_SERVO — The Torque Scaling value is typically established by the MAAT instruction as part of the controller's automatic tuning procedure. • AXIS_SERVO_DRIVE — The Torque Scaling value is typically established by the drive's automatic tuning procedure. The value can be manually calculated, if necessary, using the following guidelines. Torque Scaling = 100% Rated Torque / (Acceleration @ 100% Rated Torque) For example, if this axis is using position units of motor revolutions (revs), and that with 100% rated torque applied to the motor, the motor accelerates at a rate of 3000 Revs/Sec^2, the Torque Scaling attribute value would be calculated as shown below. Torque Scaling = 100% Rated / (3000 RPS^2) = 0.0333% Rated/ Revs Per Second^2 Note that if the Torque Scaling value does not reflect the true torque to acceleration characteristic of the system, the gains also do not reflect the true performance of the system.
Torque Threshold	AXIS_SERVO_DRIVE	REAL	GSV SSV	%Rated This attribute maps directly to a SERCOS IDN. See the SERCOS Interface standard for a description. This attribute is automatically set. You usually do not have to change it.
Torque Threshold Status	AXIS_SERVO_DRIVE	BOOL	Tag	Set when the magnitude of the physical axis Torque Feedback is less than the configured Torque Threshold.
Transform State Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	If the bit is: • ON — The axis is part of an active transform. • OFF — The axis isn't part of an active transform.

Attribute	**Axis Type**	**Data Type**	**Access**	**Description**
Tune Acceleration	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV	Position Units / Sec^2 The Tune Acceleration and Tune Deceleration attributes return the measured acceleration and deceleration values for the last run tuning procedure. These values are used, in the case of an external torque servo drive configuration, to calculate the Tune Inertia value of the axis, and are also typically used by a subsequent MAAT (Motion Apply Axis Tune) to determine the tuned values for the Maximum Acceleration and Maximum Deceleration attributes.
Tune Acceleration Time	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV	Sec The Tune Acceleration Time and Tune Deceleration Time attributes return acceleration and deceleration time in seconds for the last run tuning procedure. These values are used to calculate the Tune Acceleration and Tune Deceleration attributes.
Tune Deceleration	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV	Position Units / Sec^2 The Tune Acceleration and Tune Deceleration attributes return the measured acceleration and deceleration values for the last run tuning procedure. These values are used, in the case of an external torque servo drive configuration, to calculate the Tune Inertia value of the axis, and are also typically used by a subsequent MAAT (Motion Apply Axis Tune) to determine the tuned values for the Maximum Acceleration and Maximum Deceleration attributes.
Tune Deceleration Time	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV	Sec The Tune Acceleration Time and Tune Deceleration Time attributes return acceleration and deceleration time in seconds for the last run tuning procedure. These values are used to calculate the Tune Acceleration and Tune Deceleration attributes.

Attribute	Axis Type	Data Type	Access	Description
Tune Inertia	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV	% / MegaCounts Per Sec^2 The Tune Inertia value represents the total inertia for the axis as calculated from the measurements made during the tuning procedure. In actuality, the units of Tune Inertia are not industry standard inertia units but rather in terms of percent (%) of rated drive output per MegaCounts/Sec^2 of feedback input. In this sense it represents the input gain of torque servo drive. These units represent a more useful description of the inertia of the system as seen by the servo controller. The Tune Inertia value is used by the MAAT (Motion Apply Axis Tune) instruction to calculate the Torque Scaling. If the Tune Inertia value exceeds 100%Rated/MegaCounts Per Second^2, performance of the digital servo loop may be compromised due to excessive digitization noise associated with the velocity estimator. This noise is amplified by the Torque Scaling gain which is related to the Tune Inertia factor and passed on to the torque output of the drive. A high Tune Inertia value can, thus, result in excitation of mechanical resonances and also result in excessive heating of the motor due to high torque ripple. The only solution to this problem is to lower the loop bandwidths and optionally apply some output filtering. Since the Tune Inertia value represents a measure of the true system inertia, this situation can occur when driving a high inertia load relative to the motor, that is, a high inertia mismatch. But it can also occur when working with a drive that is undersized for the motor or with a system having low feedback resolution. In general, the lower the Tune Inertia the better the performance of the digital servo loops approximates that of an analog servo system. The product of the Tune Inertia (% Rated/MCPS) and the Velocity Servo BW (Hertz) can be calculated to directly determine quantization noise levels. Based on this product, the tuning algorithm can take action to limit high frequency noise injection to the motor. For motors with a Tune Inertia BW product of 1000 or more, the LP Filter is applied with a Filter BW of 5x the Velocity Servo Bandwidth in Hertz. This limits the amount of phase lag introduced by the LP filter to ~12 degrees which is relatively small compared to the 30 to 60 degrees of phase margin that we have for a typical tuned servo system. With a typical tuned LP filter BW value of 200 Hz, we can expect the high frequency quantization noise in the 1 KHz range to be attenuated roughly by a factor of 5. When the Tune Inertia BW product reaches 4000 or more, the LP filter alone is not going to be enough to manage the quantization noise level. So the tune algorithm begins to taper the system bandwidth by the ratio of 4000/(Tune Inertia * Vel Servo BW). This holds the quantization noise level at a fixed value, independent of the Tune Inertia BW product. For example, a motor with a Tune Inertia value of 213 and a Vel Servo BW of 41 Hz (8733 Inertia BW product) tunes with a Pos P Gain of 46 and a Vel P Gain of 117 and LP Filter BW of 93. This is a good noise-free gain set.

Attribute	Axis Type	Data Type	Access	Description
Tune Rise Time	AXIS_SERVO	REAL	GSV	Seconds The Tune Rise Time attribute returns the axis rise time as measured during the tuning procedure. This value is only applicable to axes configured for interface to an external velocity servo drive. In this case, the Tune Rise Time attribute value is used to calculate the Tune Velocity Bandwidth.
Tune Speed Scaling	AXIS_SERVO	REAL	GSV	% / KiloCounts Per Seconds The Tune Speed Scaling attribute returns the axis drive scaling factor measured during the tuning procedure. This value is only applicable to axes configured for interface to an external velocity servo drive. In this case, the Tune Speed Scaling attribute value is directly applied to the Velocity Scaling attribute by a subsequent MAAT (Motion Apply Axis Tune) instruction.
Tune Status	AXIS_SERVO AXIS_SERVO_DRIVE	INT	GSV	0 = tune process successful 1 = tune in progress 2 = tune process aborted by user 3 = tune process timed out 4 = • AXIS_SERVO — tune process failed due to servo fault • AXIS_SERVO_DRIVE — tune process failed due to drive fault 5 = axis reached Tuning Travel Limit 6 = axis polarity set incorrectly **More codes for a AXIS_SERVO_DRIVE** 7 = tune measurement fault 8 = tune configuration fault The Tune Status attribute returns status of the last run MRAT (Motion Run Axis Tuning) instruction that initiates a tuning procedure on the targeted axis. Use the attribute to determine when the MRAT initiated operation has successfully completed. Conditions may occur, however, that make it impossible for the control to properly perform the operation. When this is the case, the tune process is automatically aborted and a tune fault reported that is stored in the Tune Status output parameter.

Attribute	Axis Type	Data Type	Access	Description
Tuning Configuration Bits	AXIS_SERVO AXIS_SERVO_DRIVE	DINT	GSV SSV	Bits 0 = Tuning Direction Reverse 1 = Tune Position Error Integrator 2 = Tune Velocity Error Integrator 3 = Tune Velocity Feedforward 4 = Tune Acceleration Feedforward 5 = Tune Output Low-Pass Filter 6 = bidirectional Tuning 7 = Tune Backlash Compensation 8 = Tune Torque Offset **Tuning Direction Reverse** The Tune Direction Reverse bit determines the direction of the tuning procedure. If this bit is set (true), motion is initiated in the reverse (or negative) direction. **Tune Position Error Integrator** If this bit is: • ON — The tuning procedure calculates the Position Integral Gain. • OFF — The tuning procedure sets the Position Integral Gain to 0. **Tune Velocity Error Integrator** If this bit is: • ON — The tuning procedure calculates the Velocity Integral Gain. • OFF — The tuning procedure sets the Velocity Integral Gain to 0. **Tune Velocity Feedforward** If this bit is: • ON — The tuning procedure calculates the Velocity Feedforward Gain. • OFF — The tuning procedure sets the Velocity Feedforward Gain to 0. **Tune Acceleration Feedforward** If this bit is: • ON — The tuning procedure calculates the Acceleration Feedforward Gain. • OFF — The tuning procedure sets the Acceleration Feedforward Gain to 0. **Tune Output Low-Pass Filter** If this bit is: • ON — The tuning procedure calculates the Output Filter Bandwidth. • OFF — The tuning procedure sets the Output Filter Bandwidth to 0, which disables the filter. ***Continued on next page***

Attribute	Axis Type	Data Type	Access	Description
Tuning Configuration Bits (continued)				**Bidirectional Tuning** The Bidirectional Tuning bit determines whether the tuning procedure is unidirectional or bidirectional. If this bit is set (true), the tuning motion profile is first initiated in the specified tuning direction and then is repeated in the opposite direction. Information returned by the Bidirectional Tuning profile can be used to tune Backlash Compensation and Torque Offset. When configured for a hydraulics External Drive Type the bidirectional tuning algorithm also computes the Directional Scaling Ratio. **Tune Backlash Compensation** This tuning configuration is only valid if configured for bidirectional tuning. If this bit is: • ON — The tuning procedure calculates the Backlash Compensation Gain. • OFF — The Backlash Compensation Gain is not affected. **Tune Torque Offset** This tuning configuration is only valid if configured for bidirectional tuning. If this bit is: • ON — The tuning procedure calculates the Torque Offset. • OFF — The Torque Offset is not affected.
Tuning Speed	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV	Position Units / Seconds The Tuning Speed attribute sets the maximum speed of the tuning procedure. This attribute should be set to the desired maximum operating speed of the motor before you run the tuning procedure. The tuning procedure measures maximum acceleration and deceleration rates based on ramps to and from the Tuning Speed. Thus, the accuracy of the measured acceleration and deceleration capability is reduced by tuning at a speed other than the desired operating speed of the system..
Tuning Torque	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV	% The Tuning Torque attribute determines the maximum torque of the tuning procedure. This attribute should be set to the desired maximum safe torque level before you run the tuning procedure. The default value is 100%, which yields the most accurate measure of the acceleration and deceleration capabilities of the system. In some cases a lower tuning torque limit value may be desirable to limit the stress on the mechanics during the tuning procedure. In this case the acceleration and deceleration capabilities of the system are extrapolated based on the ratio of the tuning torque to the maximum torque output of the system. Note that the extrapolation error increases as the Tuning Torque value decreases.

Attribute	Axis Type	Data Type	Access	Description
Tuning Travel Limit	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV	Position Units The Tuning Travel Limit attribute limits the travel of the axis during the tuning procedure. If the axis cannot complete the tuning procedure before exceeding the Tuning Travel Limit, the motion module stops the tuning procedure and reports that the Tuning Travel Limit was exceeded via the Tune Status attribute. This does not mean that the Tuning Travel Limit was actually exceeded, but that had the tuning procedure gone to completion that the limit would have been exceeded.
Velocity Command	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV Tag	**Important**: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won't see the right value as the axis runs. See Axis Info Select 1. Velocity Command in Position Units / Seconds Velocity Command is the current velocity reference to the velocity servo loop, in the configured axis Position Units per Second, for the specified axis. The Velocity Command value, hence, represents the output of the outer position control loop. Velocity Command is not to be confused with Command Velocity, which represents the rate of change of Command Position input to the position servo loop.
Velocity Data Scaling	AXIS_SERVO_DRIVE	INT	GSV	This attribute is derived from the Drive Units attribute. See IDN 44 in IEC 1491.
Velocity Data Scaling Exp	AXIS_SERVO_DRIVE	INT	GSV	This attribute is derived from the Drive Units attribute. See IDN 46 in IEC 1491.
Velocity Data Scaling Factor	AXIS_SERVO_DRIVE	DINT	GSV	This attribute is derived from the Drive Units attribute. See IDN 45 in IEC 1491.
Velocity Droop	AXIS_SERVO_DRIVE	REAL	GSV SSV	Position Units / sec This attribute maps directly to a SERCOS IDN. See the SERCOS Interface standard for a description. This attribute is automatically set. You usually do not have to change it.
Velocity Error	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV Tag	**Important**: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won't see the right value as the axis runs. See Axis Info Select 1. Velocity Error in Position Units / Seconds Velocity Error is the difference, in configured axis Position Units per Second, between the commanded and actual velocity of an axis. For an axis with an active velocity servo loop, velocity error is used, along with other error terms, to drive the motor to the condition where the velocity feedback is equal to the velocity command..

Attribute	Axis Type	Data Type	Access	Description
Velocity Feedback	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV Tag	**Important**: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won't see the right value as the axis runs. See Axis Info Select 1. Velocity Feedback in Position Units / Seconds Velocity Feedback is the actual velocity of the axis as estimated by the motion module, in the configured axis Position Units per second. The estimated velocity is computed by applying a 1 KHz low-pass filter to the change in actual position over the servo update interval. Velocity Feedback is a signed value—the sign (+ or -) depends on which direction the axis is currently moving.

Attribute	Axis Type	Data Type	Access	Description
Velocity Feedforward Gain	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV	% This controller attribute is replicated in the motion module. Servo Drives require non-zero command input to generate steady-state axis acceleration or velocity. To provide the non-zero output from the Servo Module a non-zero position or velocity error needs to be present. We call this dynamic error while moving 'following error'. Well, this non-zero following error condition is a situation we are trying to avoid. We ideally want zero following error -- all the time. This could be achieved through use of the position integral gain controls as described above, but typically the response time of the integrator action is too slow to be effective. An alternative approach that has superior dynamic response is to use Velocity and Acceleration Feedforward. The Velocity Feedforward Gain attribute is used to provide the Velocity Command output necessary to generate the commanded velocity. It does this by scaling the current Command Velocity by the Velocity Feedforward Gain and adding it as an offset to the Velocity Command generated by the position loop control elements. With this done, the position loop control elements do not need to generate much of a contribution to the Velocity Command, hence the Position Error value is significantly reduced. Hence, the Velocity Feedforward Gain allows the following error of the servo system to be reduced to nearly zero when running at a constant speed. This is important in applications such as electronic gearing and synchronization applications where it is necessary that the actual axis position not significantly lag behind the commanded position at any time. The optimal value for Velocity Feedforward Gain is 100% theoretically. In reality, however, the value may need to be tweaked to accommodate velocity loops with non-infinite loop gain and other application considerations. One thing that may force a smaller Velocity Feedforward value is that increasing amounts of feedforward tends to exacerbate axis overshoot. If necessary, the Velocity Feedforward Gain may be 'tweaked' from the 100% value by running a simple user program that jogs the axis in the positive direction and monitor the Position Error of the axis during the jog. Increase the Velocity Feedforward Gain until the Position Error at constant speed is as small as possible, but still positive. If the Position Error at constant speed is negative, the actual position of the axis is *ahead* of the command position. If this occurs, decrease the Velocity Feedforward Gain such that the Position Error is again positive. Note that reasonable maximum velocity, acceleration, and deceleration values must be entered to jog the axis.

Attribute	Axis Type	Data Type	Access	Description
Velocity Integral Gain	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV	1/mSeconds-Seconds This controller attribute is replicated in the motion module. When configured for a torque (current) loop servo drive, every servo update the current Velocity Error is also accumulated in a variable called the Velocity Integral Error. This value is multiplied by the Velocity Integral Gain to produce a component to the Servo Output or Torque Command that attempts to correct for the velocity error. The characteristic of Vel I Gain correction, however, is that any non-zero Velocity Error accumulates in time to generate enough force to make the correction. This attribute of Vel I Gain makes it invaluable in applications where velocity accuracy is critical. The higher the Vel I Gain value the faster the axis is driven to the zero Velocity Error condition. Unfortunately, I Gain control is intrinsically unstable. Too much I Gain results in axis oscillation and servo instability. In certain cases, Vel I Gain control is disabled. One such case is when the servo output to the axis' drive is saturated. Continuing integral control behavior in this case would only exacerbate the situation. Another common case is when performing certain motion. When the Integrator Hold Enable attribute is set, the servo loop automatically disables the integrator during commanded motion. Due to the destabilizing nature of Integral Gain, it is recommended that Position Integral Gain and Velocity Integral Gain be considered mutually exclusive. If Integral Gain is needed for the application use one or the other, but not both. In general, where static positioning accuracy is required, Velocity Integral Gain is the better choice. The typical value for the Velocity Integral Gain is ~15 mSec^{-1}-Sec^{-1}. **If you have an AXIS_SERVO_DRIVE data type…** While the Vel I Gain, if employed, is typically established by the automatic servo tuning procedure, the Pos I Gain value may also be set manually. Before doing this it must be stressed that the Torque Scaling factor for the axis must be established for the drive system. Refer to Torque Scaling attribute description for an explanation of how the Torque Scaling factor can be calculated. Once this is done the Vel I Gain can be computed based on the current or computed value for the Vel P Gain using the following formula: Vel I Gain = 0.25 * 0.001 Sec/mSec * (Vel P Gain)^2 Assuming a Vel P Gain value of 0.25 Sec^{-1} this results in a Vel I Gain value of ~15.6 mSec^{-1}-Sec^{-1-}

Attribute	Axis Type	Data Type	Access	Description
Velocity Integrator Error	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV Tag	**Important**: To use this attribute, choose it as one of the attributes for Real Time Axis Information for the axis. Otherwise, you won't see the right value as the axis runs. See Axis Info Select 1. Velocity Integrator Error in Position Units – mSec / Sec Velocity Integrator Error is the running sum of the Velocity Error, in the configured axis Position Units per Second, for the specified axis. For an axis with an active velocity servo loop, the velocity integrator error is used, along with other error terms, to drive the motor to the condition where the velocity feedback is equal to the velocity command.
Velocity Limit Bipolar	AXIS_SERVO_DRIVE	REAL	GSV SSV	Position Units / sec This attribute maps directly to a SERCOS IDN. See the SERCOS Interface standard for a description. This attribute is automatically set. You usually do not have to change it.
Velocity Limit Negative	AXIS_SERVO_DRIVE	REAL	GSV SSV	Position Units / sec This attribute maps directly to a SERCOS IDN. See the SERCOS Interface standard for a description. This attribute is automatically set. You usually do not have to change it.
Velocity Limit Positive	AXIS_SERVO_DRIVE	REAL	GSV SSV	Position Units / sec This attribute maps directly to a SERCOS IDN. See the SERCOS Interface standard for a description. This attribute is automatically set. You usually do not have to change it.
Velocity Limit Status	AXIS_SERVO_DRIVE	BOOL	Tag	Set when the magnitude of the commanded velocity to the velocity servo loop input is greater than the configured Velocity Limit.
Velocity Lock Status	AXIS_SERVO_DRIVE	BOOL	Tag	Set when the magnitude of the physical axis Velocity Feedback is within the configured Velocity Window of the current velocity command.
Velocity Offset	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV Tag	Velocity Offset in Position Units / Sec Velocity Offset compensation can be used to give a dynamic velocity correction to the output of the position servo loop. Since this value is updated synchronously every Coarse Update Period, the Velocity Offset can be tied into custom outer control loop algorithms using Function Block programming.
Velocity Polarity	AXIS_SERVO_DRIVE	INT	GSV	This attribute is derived from the Drive Polarity attribute. See IDN 42 in IEC 1491.

Attribute	Axis Type	Data Type	Access	Description
Velocity Proportional Gain	AXIS_SERVO AXIS_SERVO_DRIVE	REAL	GSV SSV	1/Seconds This controller attribute is replicated in the motion module. **AXIS_SERVO** When configured for a torque (current) loop servo drive, the servo module's digital velocity loop provides damping without the requirement for an analog tachometer. The Velocity Error is multiplied by the Velocity Proportional Gain to produce a component to the Servo Output or Torque Command that ultimately attempts to correct for the velocity error, creating the damping effect. Thus, increasing the Velocity Proportional Gain results in smoother motion, enhanced acceleration, reduced overshoot, and greater system stability. The velocity loop also allows higher effective position loop gain values to be used, however, too much Velocity Proportional Gain leads to high frequency instability and resonance effects. Note that units for Velocity Proportional Gain are identical to that of the Position Proportional Gain making it easy to perform classic inches/min/mil calculations to determine static stiffness or damping. **Maximum Bandwidth** There are limitations to the maximum bandwidth that can be achieved for the velocity loop based on the dynamics of the torque loop of the servo drive and the desired damping of the system, Z. These limitations may be expressed as follows: Bandwidth (Velocity) = 0.25 * $1/Z^2$ * Bandwidth (Torque) For example, if the bandwidth of the drive's torque loop is 100 Hz and the damping factor, Z, is 0.8, the velocity bandwidth is approximately 40 Hz. Based on this number the corresponding gains for the loop can be computed. Note that the bandwidth of the torque loop includes feedback sampling delay and filter time constant. The velocity loop in the motion controller is not used when the servo module is configured for a velocity loop servo drive, Thus, establishing the Velocity Proportional Gain is not required in this case. The typical value for the Velocity Proportional Gain is ~250 Sec^{-1}.

Attribute	Axis Type	Data Type	Access	Description
Velocity Proportional Gain (count.)				**AXIS_SERVO_DRIVE** The standard RA SERCOS drive's digital velocity loop provides damping without the requirement for an analog tachometer. The Velocity Error is multiplied by the Velocity Proportional Gain to produce a Torque Command that ultimately attempts to correct for the velocity error, creating the damping effect. Thus, increasing the Velocity Proportional Gain results in smoother motion, enhanced acceleration, reduced overshoot, and greater system stability. The velocity loop also allows higher effective position loop gain values to be used, however, too much Velocity Proportional Gain leads to high frequency instability and resonance effects. Note that units for Velocity Proportional Gain are identical to that of the Position Proportional Gain making it easy to perform classic calculations to determine damping and bandwidth. If you know the desired unity gain bandwidth of the velocity servo in Hertz, use the following formula to calculate the corresponding P gain. Vel P Gain = Bandwidth (Hertz) / 6.28 In general, modern velocity servo systems typically run with a unity gain bandwidth of ~40 Hertz. The typical value for the Velocity Proportional Gain is ~250 Sec^{-1}. **Maximum Bandwidth** There are limitations to the maximum bandwidth that can be achieved for the velocity loop based on the dynamics of the inner torque loop of the system and the desired damping of the system, Z. These limitations may be expressed as follows: Bandwidth (Velocity) = 0.25 * $1/Z^2$ * Bandwidth (Torque) For example, if the bandwidth of the drive's torque loop is 100 Hz and the damping factor, Z, is 0.8, the velocity bandwidth is approximately 40 Hz. Based on this number the corresponding gains for the loop can be computed. Note that the bandwidth of the torque loop includes feedback sampling delay and filter time constant.

Attribute	Axis Type	Data Type	Access	Description
Velocity Scaling	AXIS_SERVO	REAL	GSV SSV	% / Position Units Per Second This controller attribute is replicated in the motion module. The Velocity Scaling attribute is used to convert the output of the servo loop into equivalent voltage to an external velocity servo drive. This has the effect of 'normalizing' the units of the servo loop gain parameters so that their values are not affected by variations in feedback resolution, drive scaling, or mechanical gear ratios. The Velocity Scaling value is typically established by servo's automatic tuning procedure but these values can be calculated if necessary using the following guidelines. If the axis is using a velocity servo drive, the software velocity loop in the servo module is disabled. In this case the Velocity Scaling value can be calculated by the following formula: Velocity Scaling = 100% / (Speed @ 100%) For example, if this axis is using position units of motor revolutions (revs), and the servo drive is scaled such that with an input of 100% (for example, 10 Volts) the motor goes 5,000 RPM (or 83.3 RPS), the Torque Scaling attribute value would be calculated as shown below. Velocity Scaling = 100% / (83.3 RPS) = 1.2% / Revs Per Second

<table>
<tr><th>Attribute</th><th>Axis Type</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td>Velocity Servo Bandwidth</td><td>AXIS_SERVO
AXIS_SERVO_DRIVE</td><td>REAL</td><td>GSV
SSV</td><td>Hertz

The value for the Velocity Servo Bandwidth represents the unity gain bandwidth that is to be used to calculate the gains for a subsequent MAAT (Motion Apply Axis Tune) instruction. The unity gain bandwidth is the frequency beyond which the velocity servo is unable to provide any significant position disturbance correction. In general, within the constraints of a stable servo system, the higher the Velocity Servo Bandwidth is the better the dynamic performance of the system. A maximum value for the Velocity Servo Bandwidth is generated by the MRAT (Motion Run Axis Tune) instruction. Computing gains based on this maximum value via the MAAT instruction results in dynamic response in keeping with the current value of the Damping Factor described above. Alternatively, the responsiveness of the system can be ‘softened’ by reducing the value of the Velocity Servo Bandwidth before executing the MAAT instruction..

There are practical limitations to the maximum Velocity Servo Bandwidth for the velocity servo loop based on the drive system and, in some cases, the desired damping factor of the system, Z. Exceeding these limits could result in an unstable servo operation.
<table>
<tr><th>Data type</th><th>Bandwidth limits</th></tr>
<tr><td>AXIS_SERVO</td><td>For an external velocity loop servo drive,

Max Velocity Servo Bandwidth (Hz) = 0.159 * 2/Tune Rise Time

For an external torque loop servo drive,

Max Velocity Servo Bandwidth (Hz) = 0.159 * 0.25 * $1/Z^2$ * 1/Drive Model Time Constant</td></tr>
<tr><td>AXIS_SERVO_DRIVE</td><td>Max Velocity Servo Bandwidth (Hz) = 0.159 * 0.25 * $1/Z^2$ * 1/Drive Model Time Constant</td></tr>
</table>
The factor of 0.159 represents the 1/2PI factor required to convert Radians per Second units to Hertz.</td></tr>
<tr><td>Velocity Standstill Status</td><td>AXIS_SERVO_DRIVE</td><td>BOOL</td><td>Tag</td><td>Set when the magnitude of the physical axis Velocity Feedback is less than the configured Velocity Standstill Window.</td></tr>
<tr><td>Velocity Standstill Window</td><td>AXIS_SERVO_DRIVE</td><td>REAL</td><td>GSV
SSV</td><td>Position Units / sec

This attribute maps directly to a SERCOS IDN. See the SERCOS Interface standard for a description. This attribute is automatically set. You usually do not have to change it.</td></tr>
<tr><td>Velocity Threshold</td><td>AXIS_SERVO_DRIVE</td><td>REAL</td><td>GSV
SSV</td><td>Position Units / sec

This attribute maps directly to a SERCOS IDN. See the SERCOS Interface standard for a description. This attribute is automatically set. You usually do not have to change it.</td></tr>
</table>

Attribute	Axis Type	Data Type	Access	Description
Velocity Threshold Status	AXIS_SERVO_DRIVE	BOOL	Tag	Set when the magnitude of the physical axis Velocity Feedback is less than the configured Velocity Threshold.
Velocity Window	AXIS_SERVO_DRIVE	REAL	GSV SSV	Position Units / sec This attribute maps directly to a SERCOS IDN. See the SERCOS Interface standard for a description. This attribute is automatically set. You usually do not have to change it.
Watch Event Armed Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set when a watch event has been armed through execution of the MAW (Motion Arm Watch) instruction. Cleared when either a watch event occurs or a MDW (Motion Disarm Watch) instruction is executed.
Watch Event Status	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	BOOL	Tag	Set when a watch event has occurred. Cleared when either another MAW (Motion Arm Watch) instruction or a MDW (Motion Disarm Watch) instruction is executed.
Watch Event Task	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	DINT	MSG	Shows which task is triggered when the watch event happens. • An instance of 0 means that no event task is configured to be triggered by the watch event. • The task is triggered at the same time that the Process Complete bit is set for the instruction that armed the watch event. • The controller sets this attribute. do not set it by an external device.
Watch Position	AXIS_CONSUMED AXIS_GENERIC AXIS_SERVO AXIS_SERVO_DRIVE AXIS_VIRTUAL	REAL	GSV Tag	Watch Position in Position Units Watch Position is the current set-point position of an axis, in the configured axis Position Units, as set up in the last, most recently executed, MAW (Motion Arm Watch) instruction for that axis.

Additional Error Code Information

See these manuals for more information about error codes displayed on drives and/or multi-axis motion control systems.

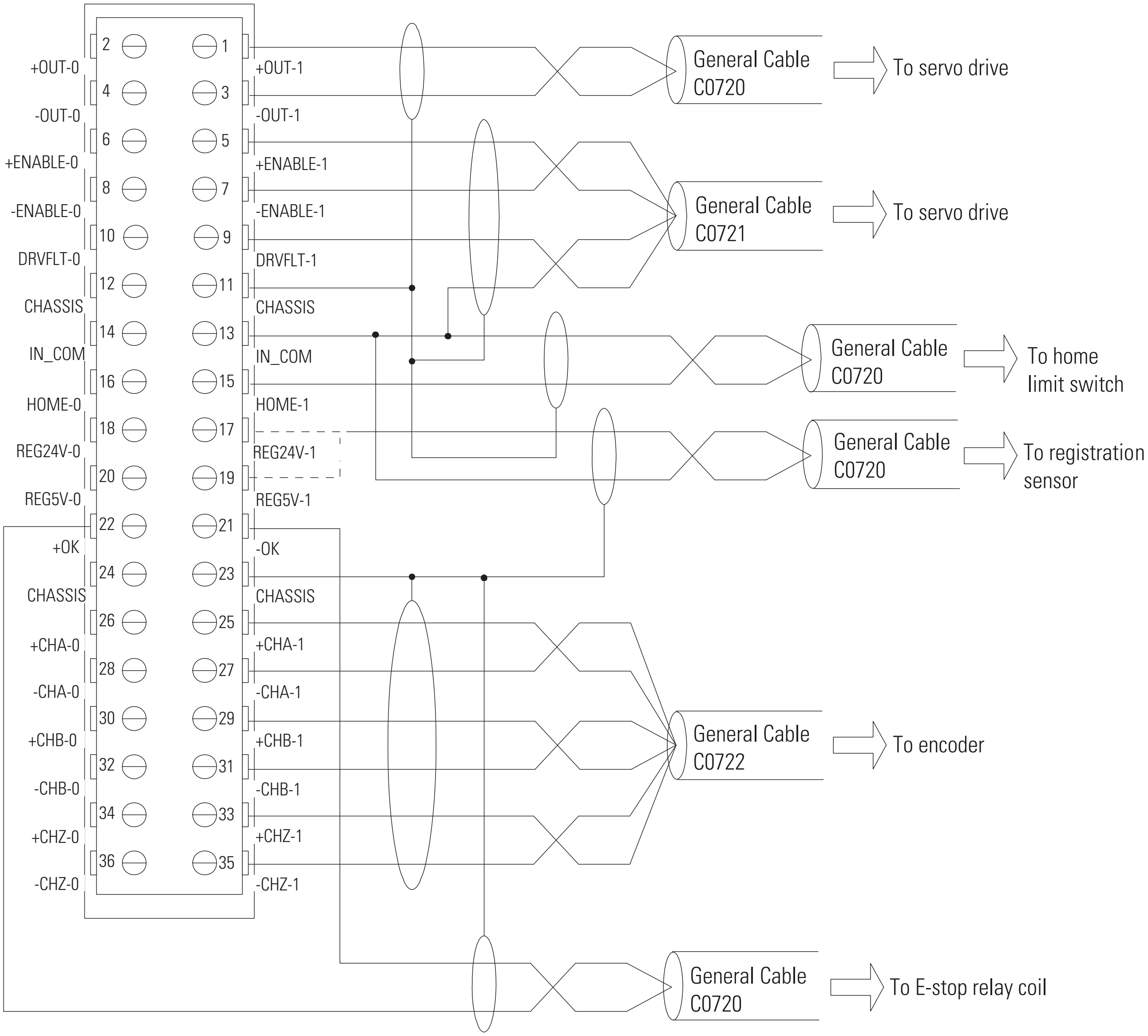
Publication	Description
Kinetix 2000 Multi-Axis Drive User Manual, publication 2093-UM001.	Provides detailed installation instructions for mounting, wiring, and troubleshooting your Kinetix 2000 drive, and system integration for your drive/motor combination with a Logix controller.
Kinetix 6000 Multi-Axis Drive User Manual, publication 2094-UM001.	Provides detailed installation instructions for mounting, wiring, and troubleshooting your Kinetix 6000 drive, and system integration for your drive/motor combination with a Logix controller.
Kinetix 7000 High Power Servo Drive User Manual, publication 2099-UM001.	Provides detailed installation instructions for mounting, wiring, and troubleshooting your Kinetix 7000 drive, and system integration for your drive/motor combination with a Logix controller.
Ultra 3000 Digital Servo Drive Installation Instructions, publication 2098-IN003.	Provides the mounting, wiring, and connecting procedures for the Ultra3000 and standard Rockwell Automation/Allen-Bradley motors recommended for use with the Ultra3000.
8720 High Performance Drive Installation Instructions, publication 8720MC-IN001.	Provides the mounting, wiring, and connecting procedures for the 8720MC and standard Rockwell Automation/ Allen-Bradley motors recommended for use with the 8720MC.

Wiring Diagrams

Introduction

Use the diagrams in this appendix to wire the motion control equipment of your control system.

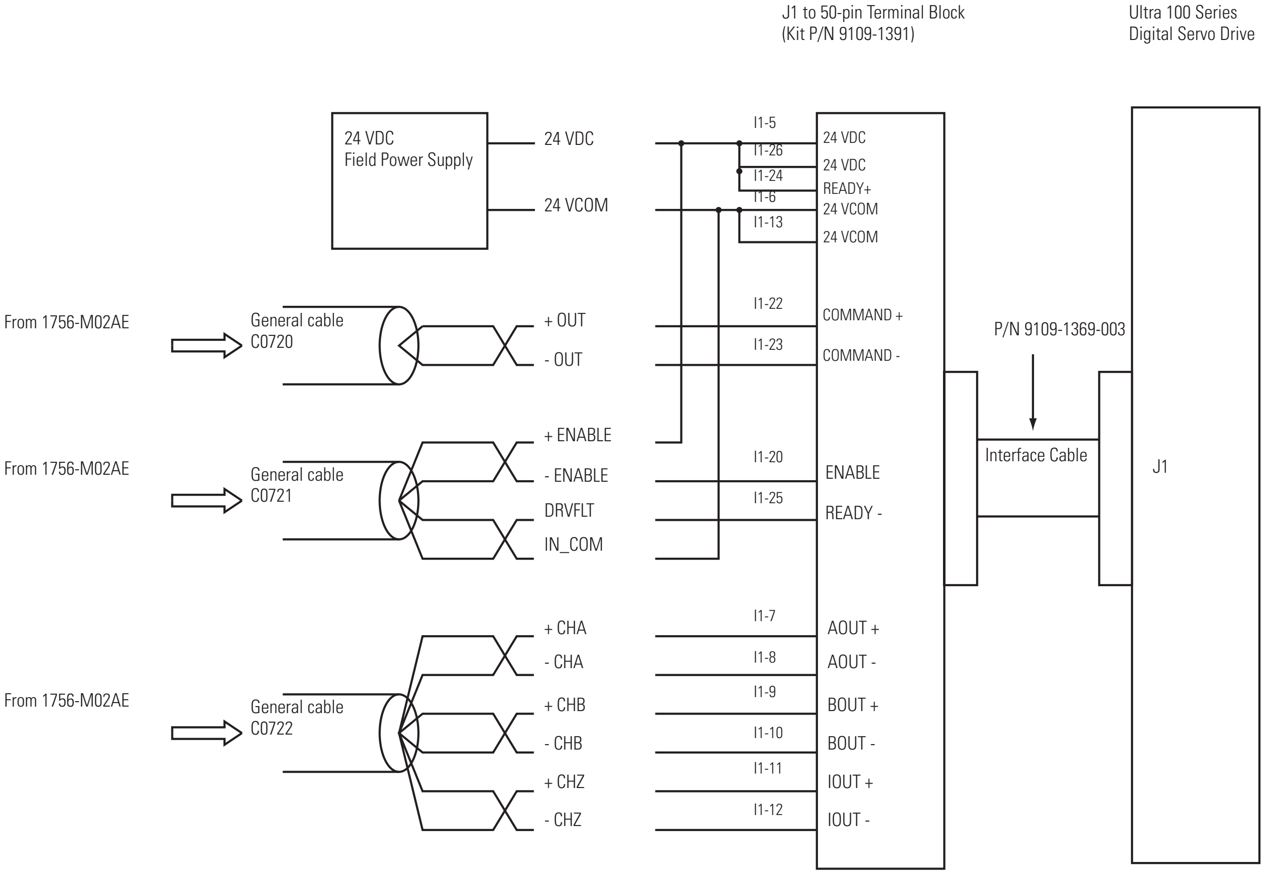
1756-M02AE Module



Notes

This example shows the wiring for Axis 1 Wire Axis 0 the same way.

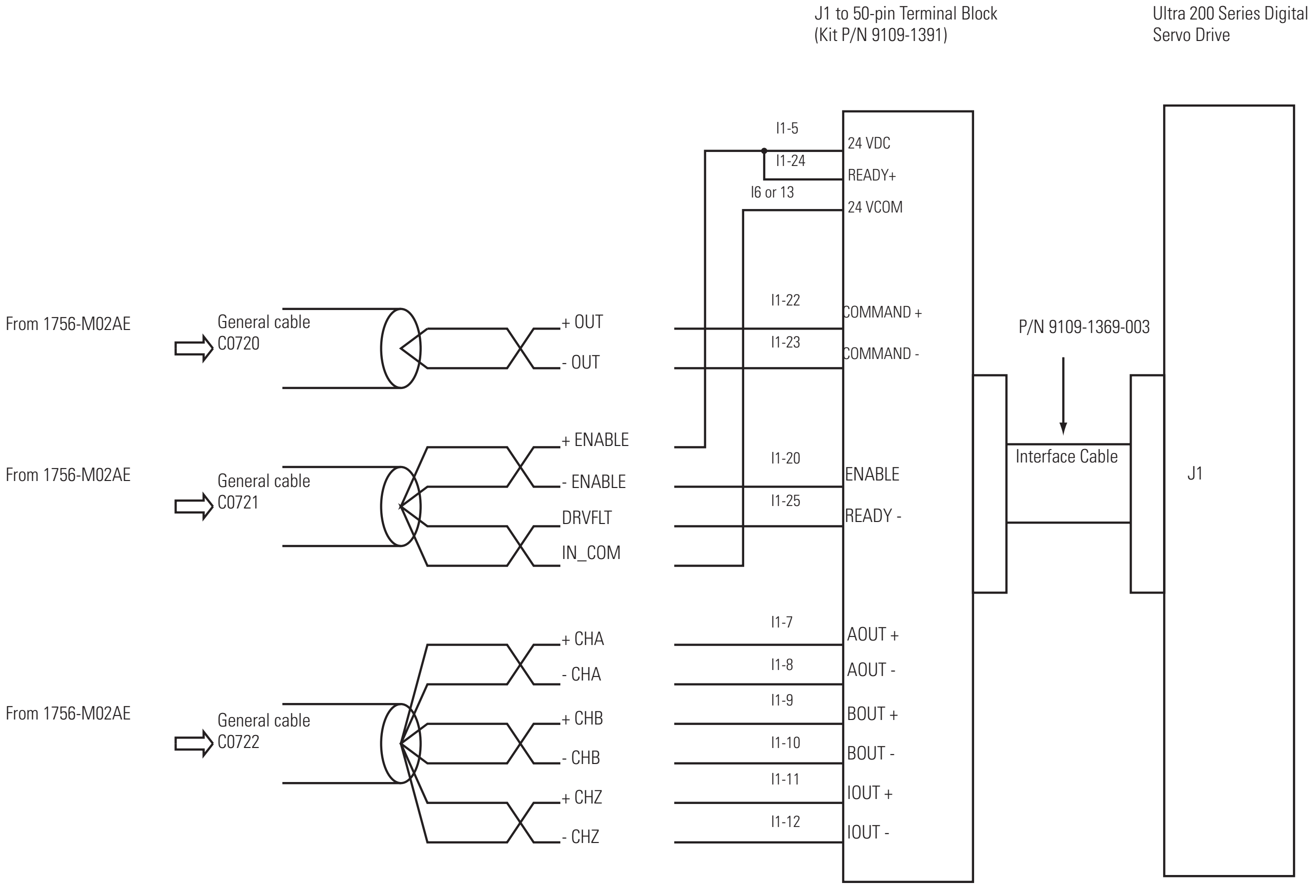
Ultra 100 Series Drive



Notes

- This is an example of one way to wire the drive.
- See Ultra 100 Series Drive Installation Manual, publication number 1398-5.2, for other configurations.

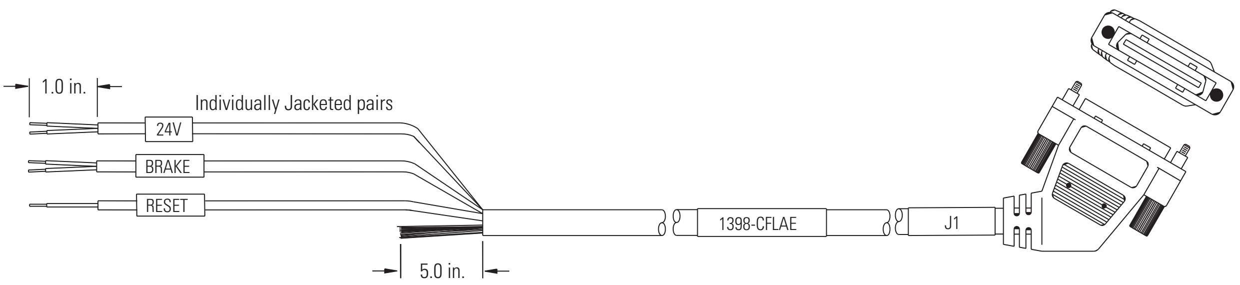
Ultra 200 Series Drive



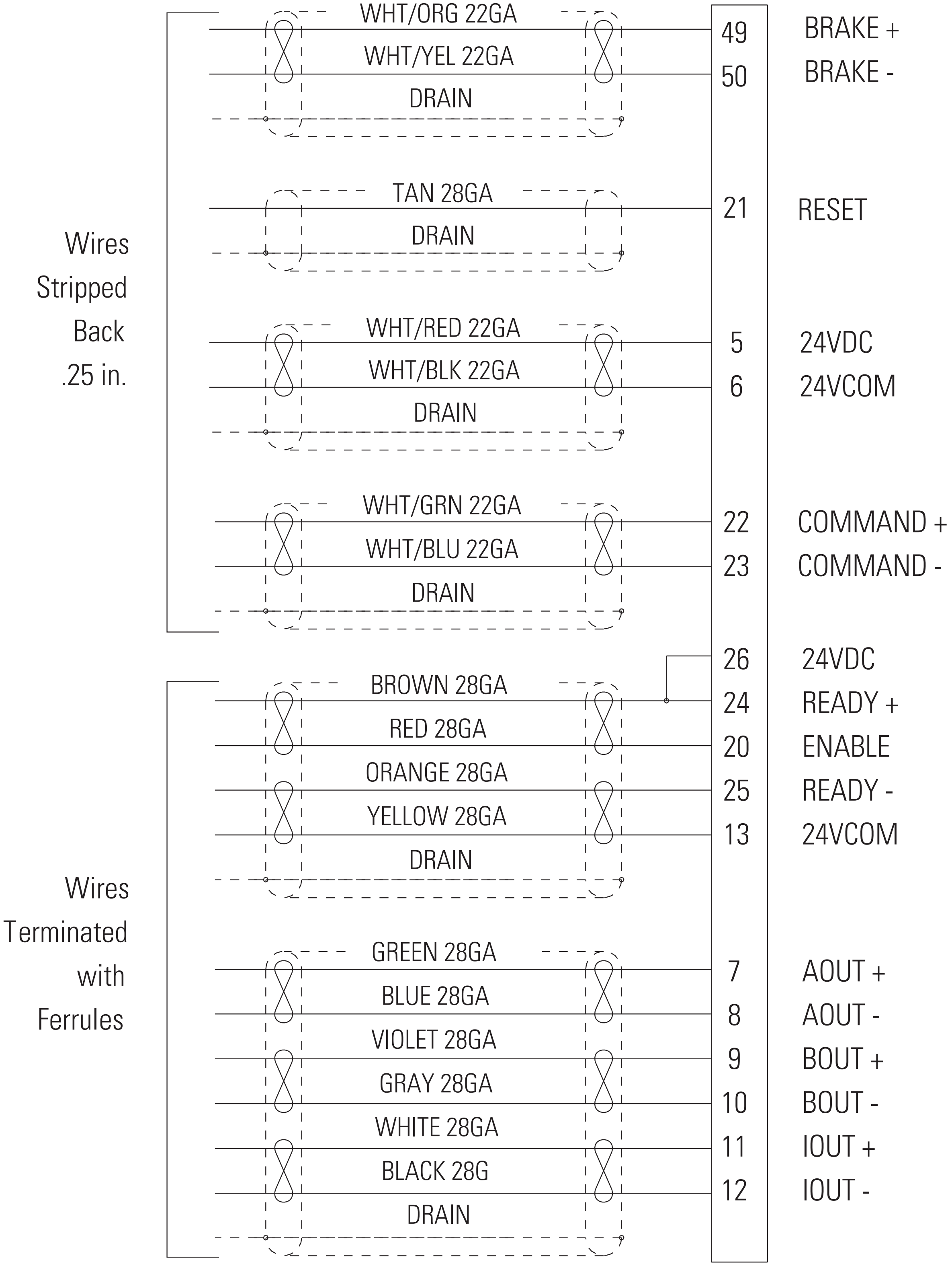
Notes

- This is an example of one way to wire the drive.
- See Ultra 200 Series Drive Installation Manual, publication number 1398-5.0, for other configurations.

1398-CFLAE*xx* Cable

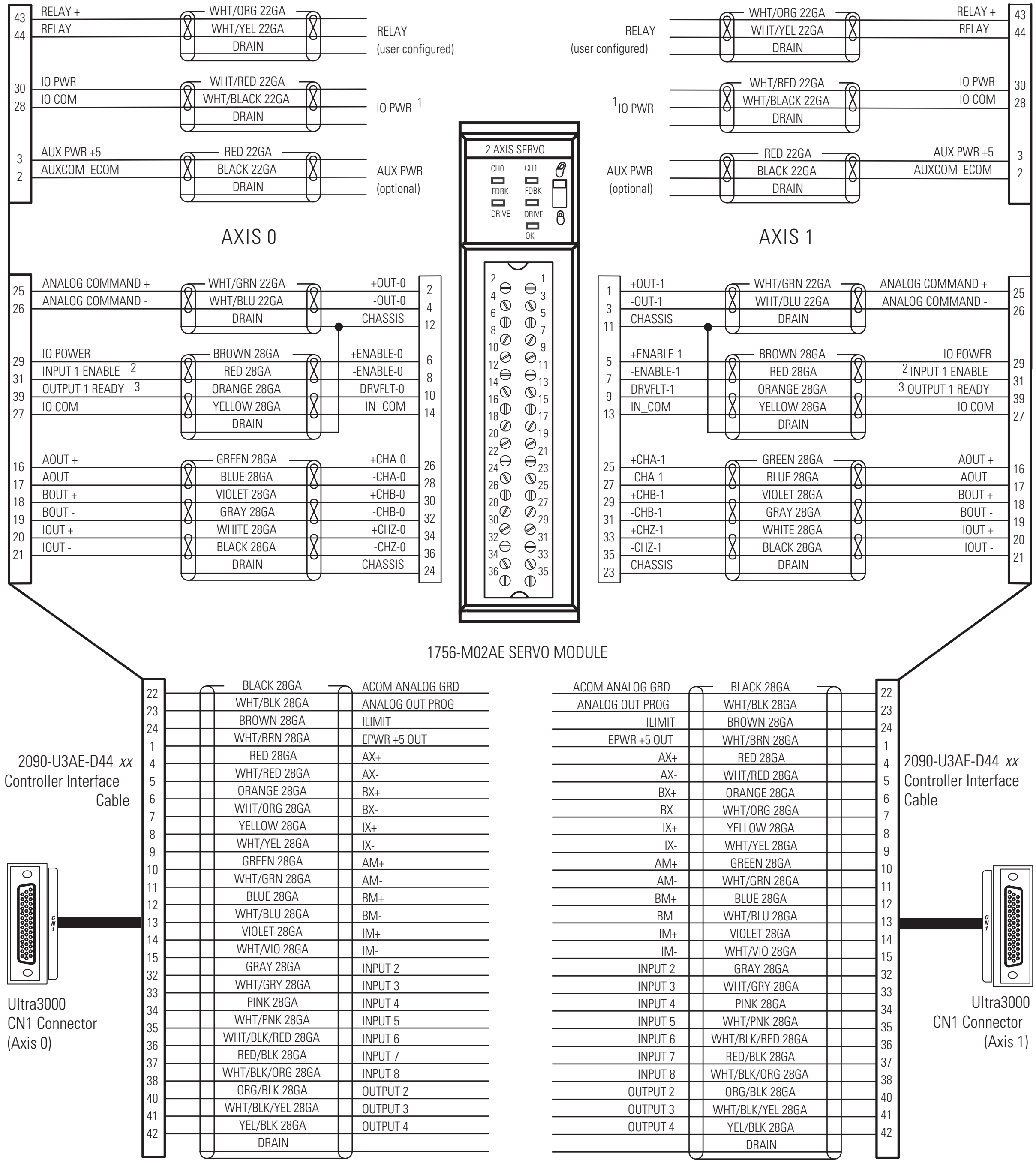


Pinouts for 1398-CFLAE*xx* Cable



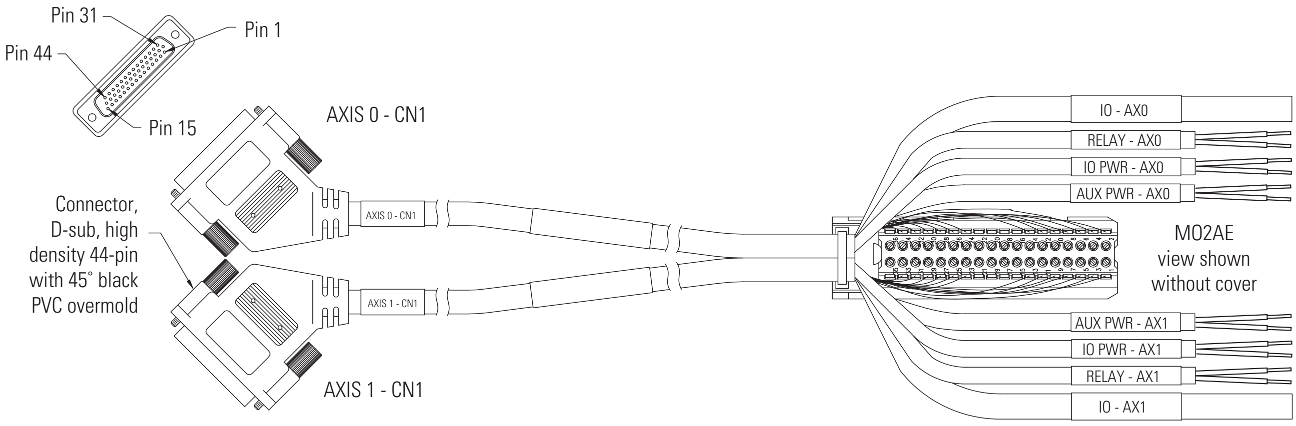
Ultra3000 Drive

Ultra3000 to 1756-M02AE Interconnect diagram.

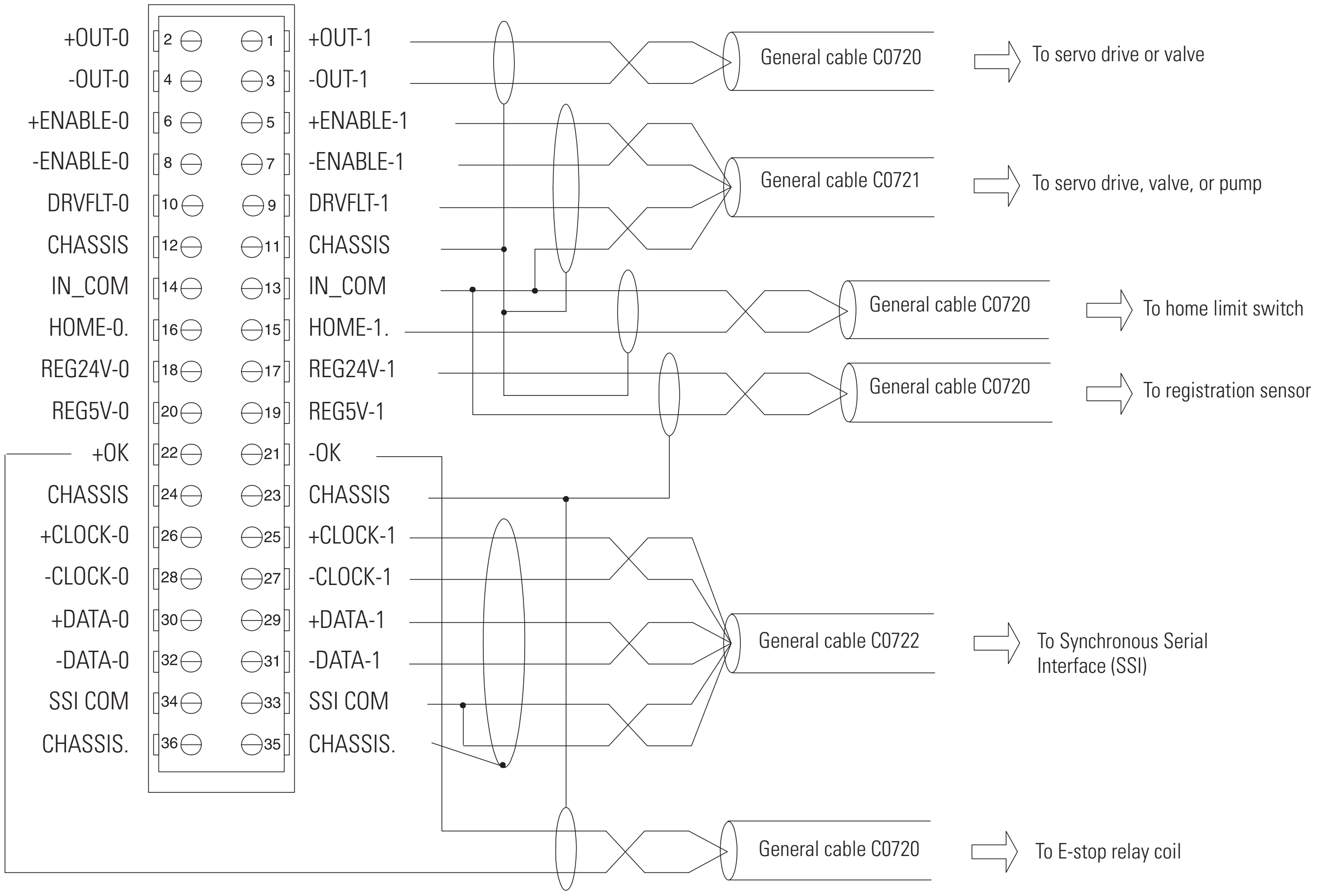


For more information, see Ultra3000 Digital Servo Drives Installation Manual, publication number 2098-IN003.

2090-U3AE-D44*xx* Cable.

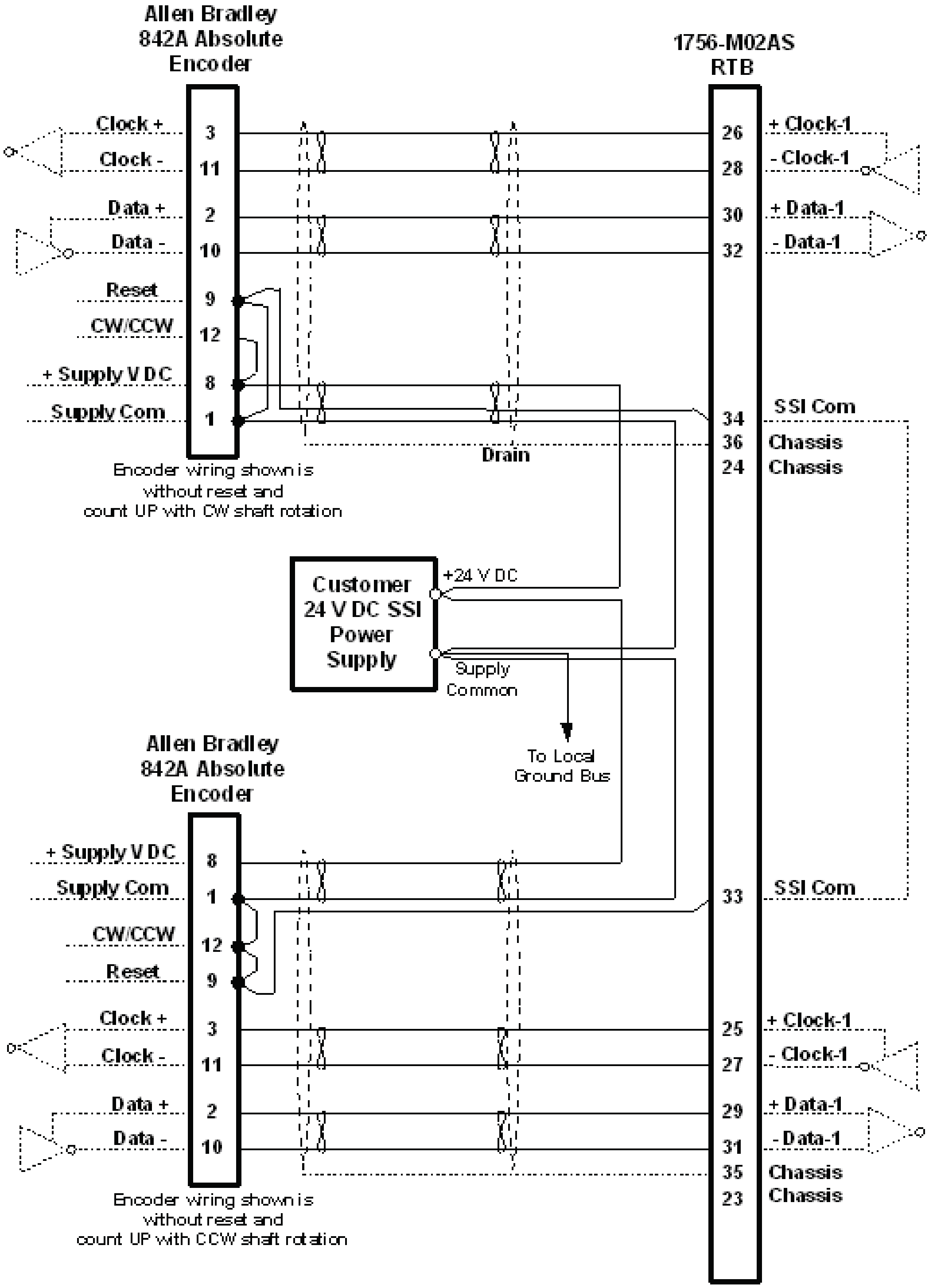


1756-M02AS Module

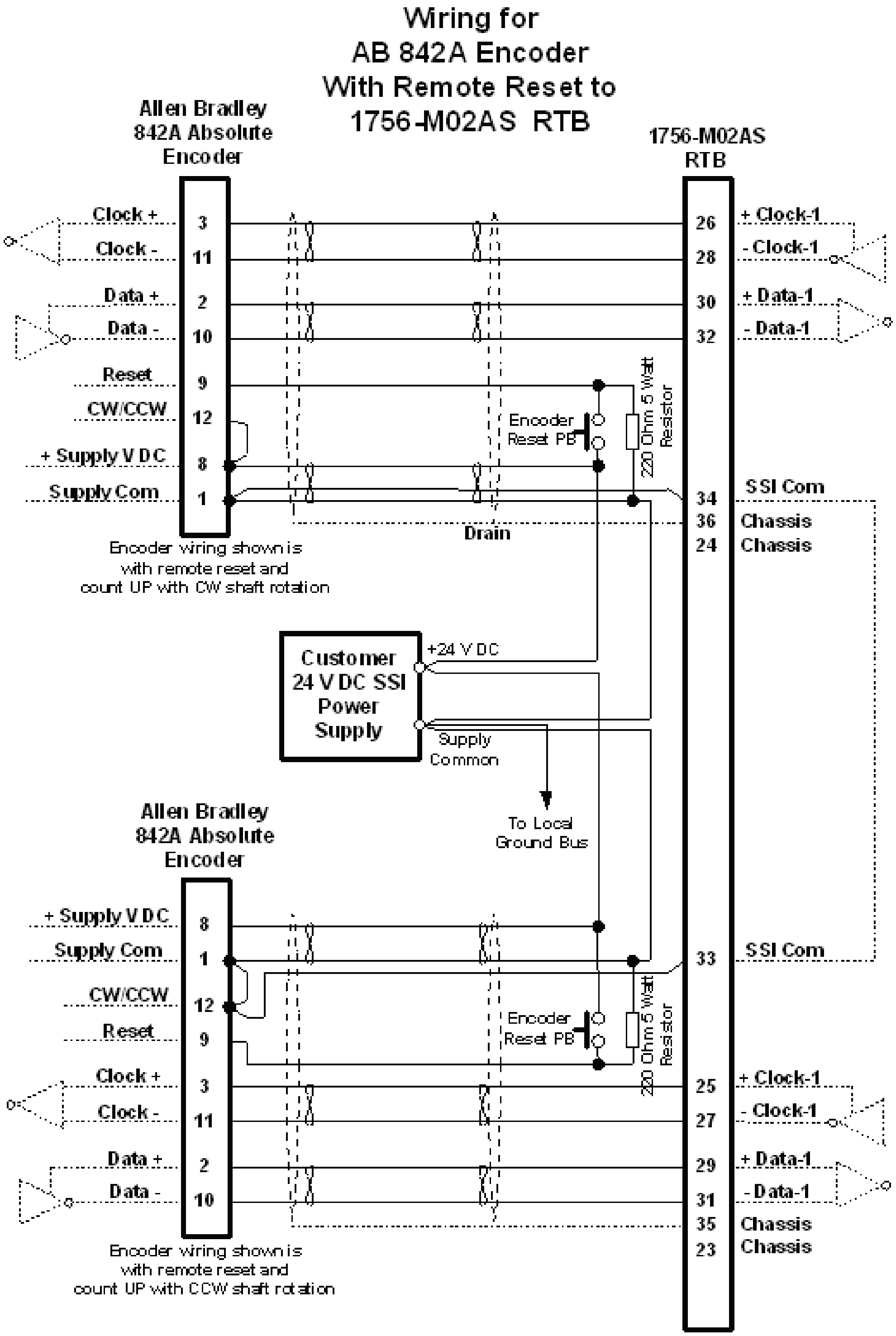


This example shows the wiring for Axis 1 Wire Axis 0 the same way.

Wiring from AB 842A Encoder without Reset to 1756-M02AS RTB

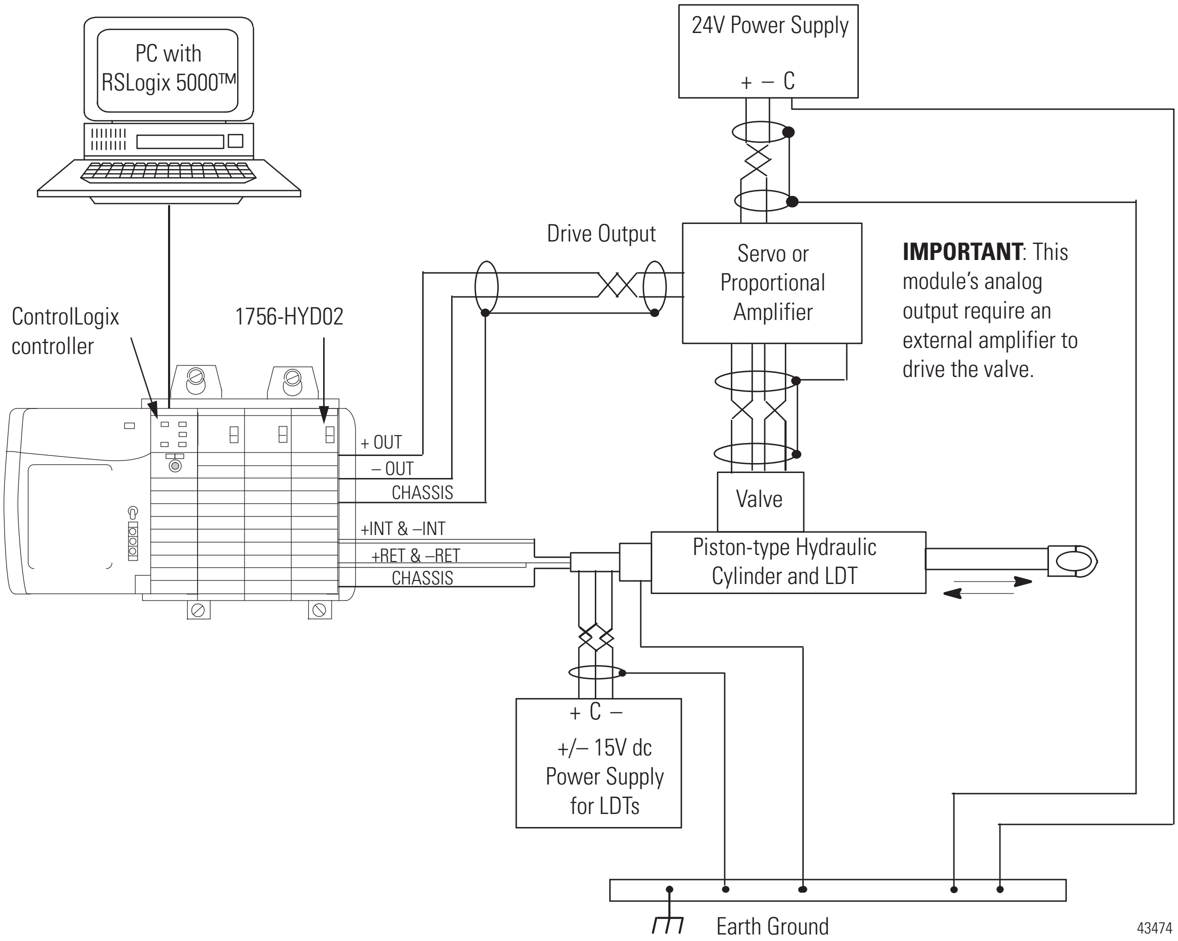


Wiring for AB 842A Encoder with Remote Reset to 1756-M02AS RTB



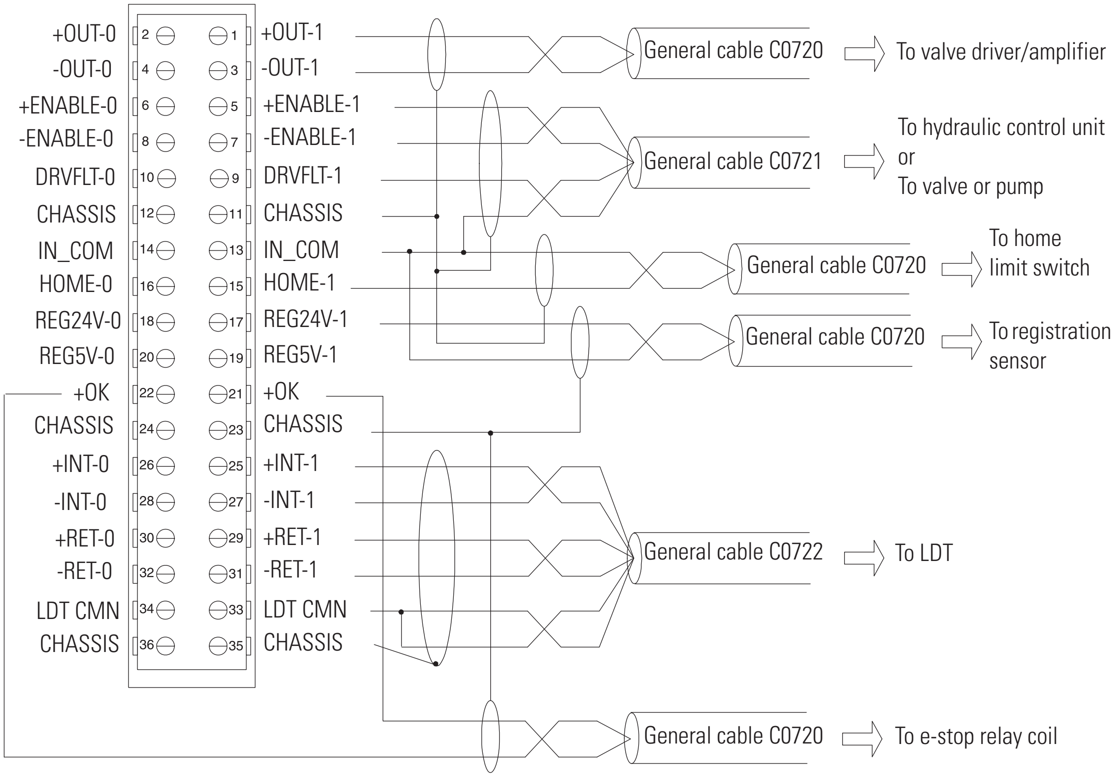
1756-HYD02 Application Example

This example uses a 1-axis loop with a differential LDT input.



IMPORTANT: This module's analog output require an external amplifier to drive the valve.

1756-HYD02 Module



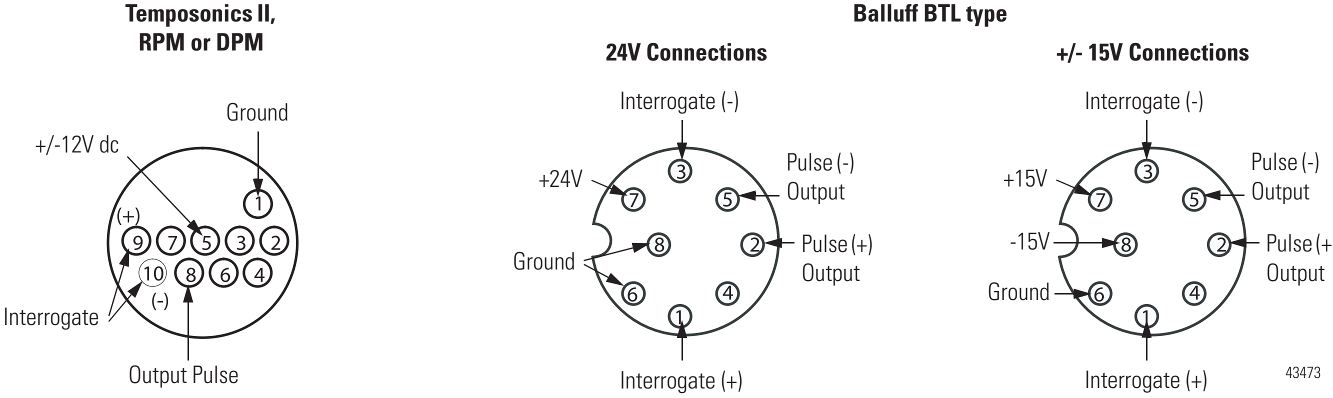
Notes

- This example shows the wiring for Axis 1. Wire Axis 0 the same way.
- Use transducers that use an external interrogation signal.
- Do not exceed the specified isolation voltage between power sources.

LDTs

These diagrams show the connections for Temposonic and Balluff LDTs.

IMPORTANT Other suppliers also have compatible LDTs. Before you connect an LDT to your module, make sure that it is the best LDT for your application.



No shield connections on these examples

This table lists the LDT connections.

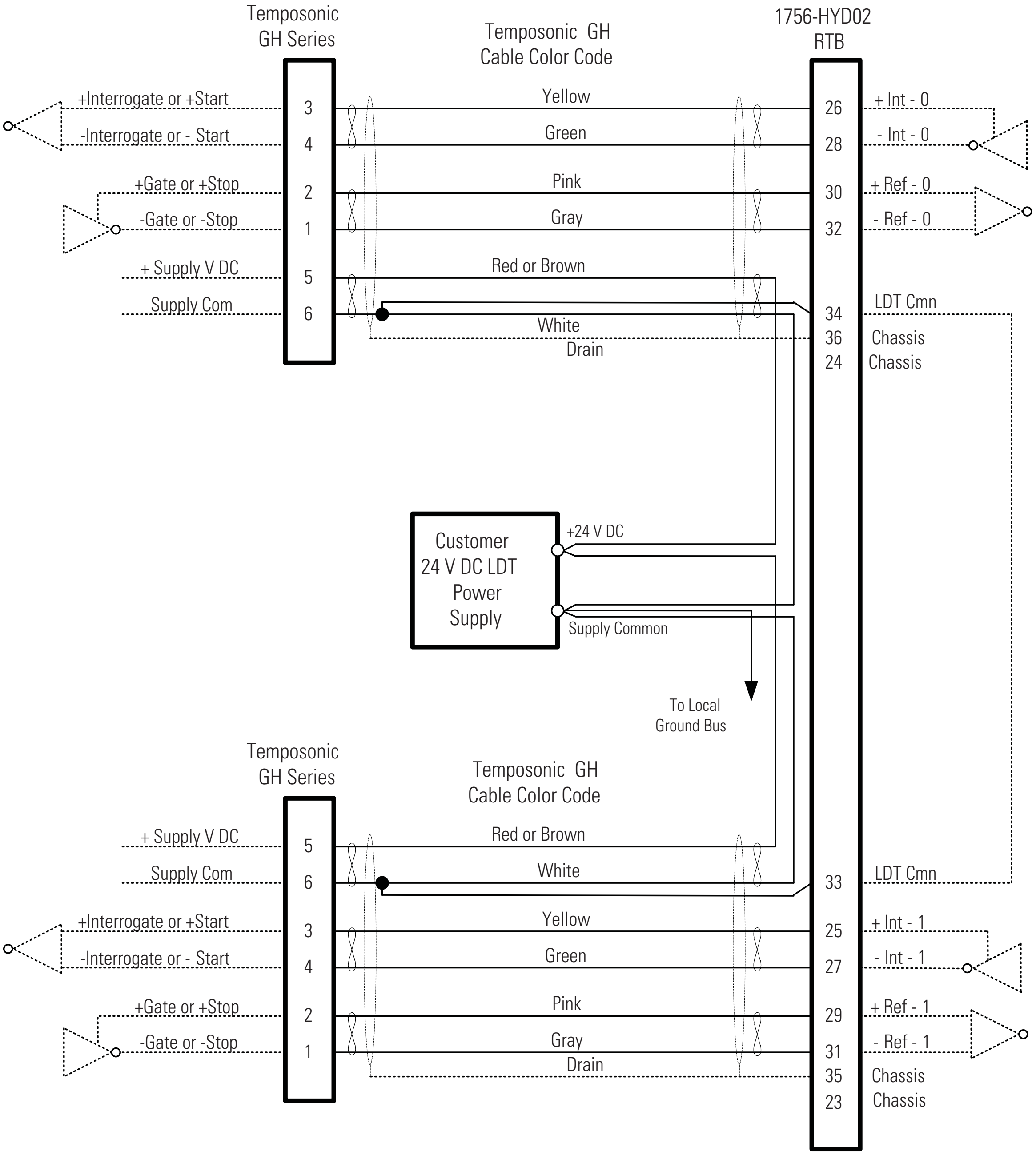
LDT Connections for Fabricating Your Own LDT Cable

Function[1]	1756-HYD02 RTB Wiring (Numbers below represent terminal numbers)		Temposonics II[2] RPM or DPM	Balluff BTL type	
	Channel 0	Channel 1		24V dc	+/- 15V dc
(+) Interrogate	26	25	9 - Yellow	1 - Yellow	1 - Yellow
(-) Interrogate	28	27	10 - Green	3 - Pink	3 - Pink
Power Supply	N/A		5 - Red (+/-12V)	7 - Brown (+24V)	7 - Brown (+15V) 8 - White (-15V)
Ground	34	33	1 - White	6 - Blue 8 - White	6 - Blue
Output Pulse	30 (+) 32 (-)	29 (+) 31 (-)	8 - Purple	2 - Gray (+) 5 - Green (-)	2 - Gray (+) 5 - Green (-)

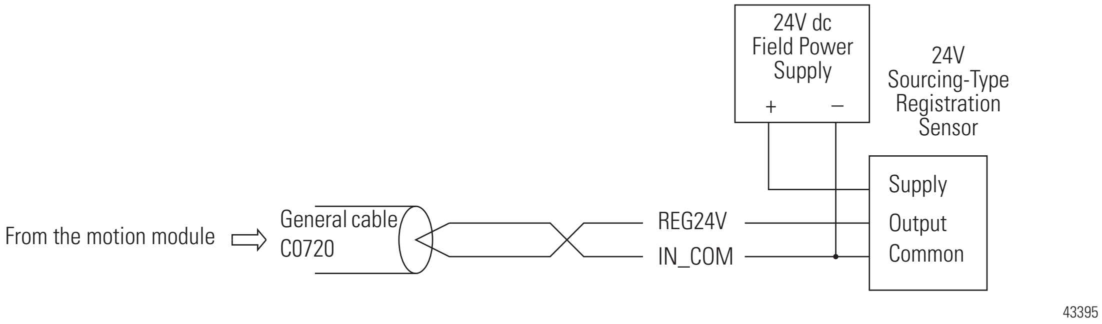
[1] (+) and (-) wires of the same function should be a twisted pair within the cable.

[2] Do not connect to pins 2, 3, 4, 6 or 7

Temposonic GH Feedback Device



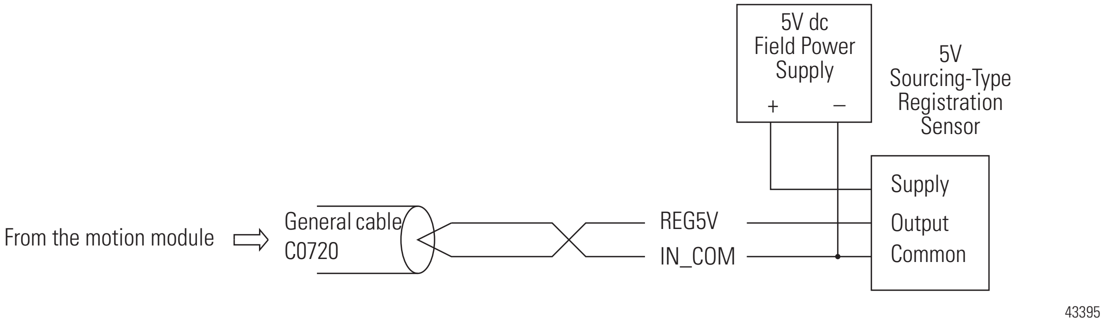
24V Registration Sensor



Notes

- Use sourcing-type registration sensors.
- Wire the inputs so that they get source current from the sensor.
- do not use current sinking sensor configurations because the registration input common (IN_ COM) is shared with the other 24V servo module inputs.

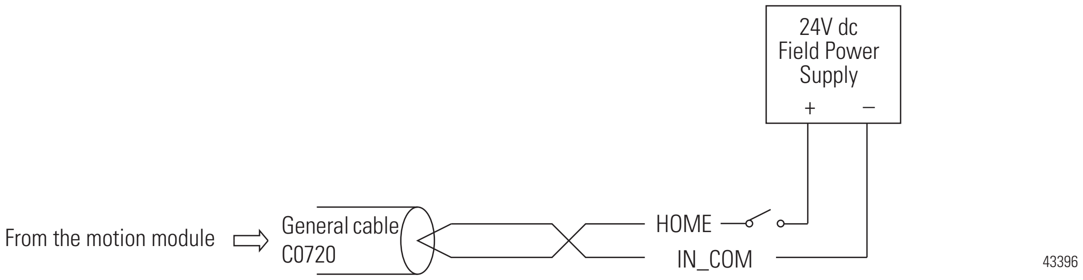
5V Registration Sensor



Notes

- Use sourcing-type registration sensors.
- Wire the inputs so that they get source current from the sensor.
- do not use current sinking sensor configurations because the registration input common (IN_ COM) is shared with the other 24V servo module inputs.

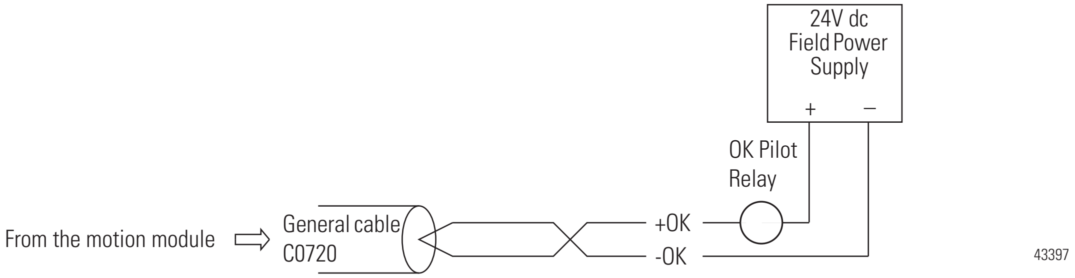
Home Limit Switch Input

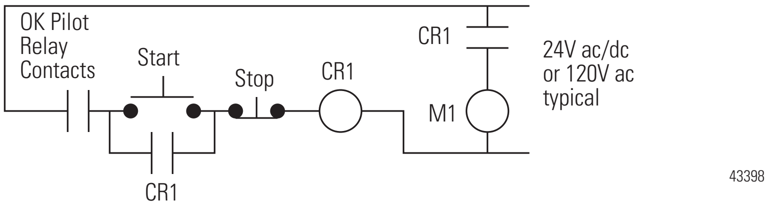


Notes

- The home limit switch inputs to the servo module are designed for 24V dc nominal operation.
- Wire these inputs for current sourcing operation.

OK Contacts





Notes

- Use the OK relay contacts to connect to an E-stop string that controls power to the associated pumps or drives.
- The OK contacts are rated to drive an external 24V dc pilot relay (for example, Allen-Bradley 700-HA32Z24) whose contacts can be incorporated into the E-stop string.

Notes:

Servo Loop Block Diagrams

Introduction

This appendix shows the servo loop block diagrams for common motion configurations.

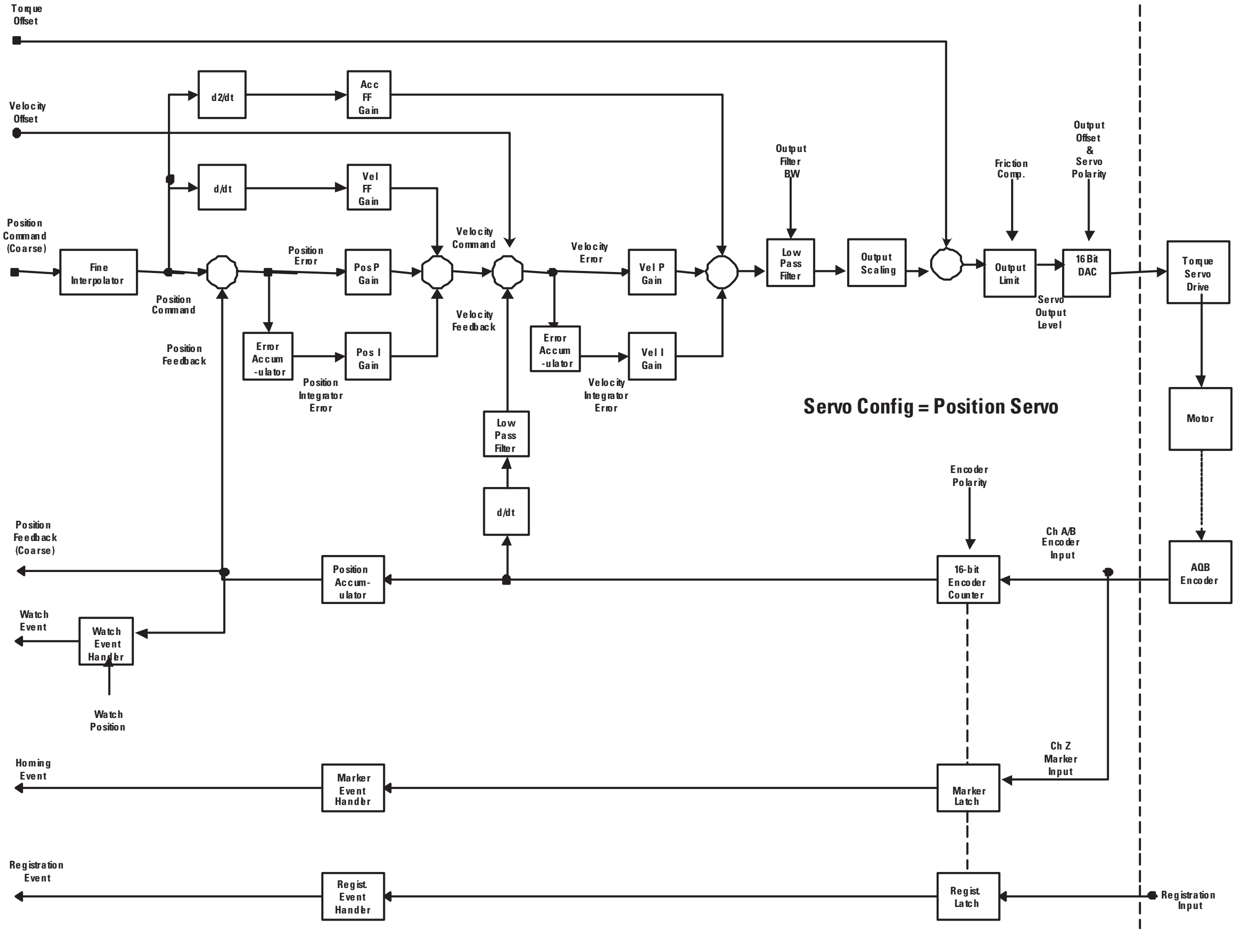
Interpreting the Diagrams

The diagrams use these labels for axes attributes.

Label	AXIS Attribute
Acc FF Gain	AccelerationFeedforwardGain
Friction Comp	FrictionCompensation
Output Filter BW	OutputFilterBandwidth
Output Limit	OutputLimit
Output Offset	OutputOffset
Output Scaling	OutputScaling
Pos I Gain	PositionIntegralGain
Pos P Gain	PositionProportionalGain
Position Error	PositionError
Position Integrator Error	PositionIntegratorError
Registration Position	RegistrationPosition
Servo Output Level	ServoOutputLevel
Vel FF Gain	VelocityFeedforwardGain
Vel I Gain	VelocityIntegralGain
Vel P Gain	VelocityProportionalGain
Velocity Command	VelocityCommand
Velocity Error	VelocityError
Velocity Feedback	VelocityFeedback
Velocity Integrator Error	VelocityIntegratorError
Watch Position	WatchPosition

AXIS_SERVO

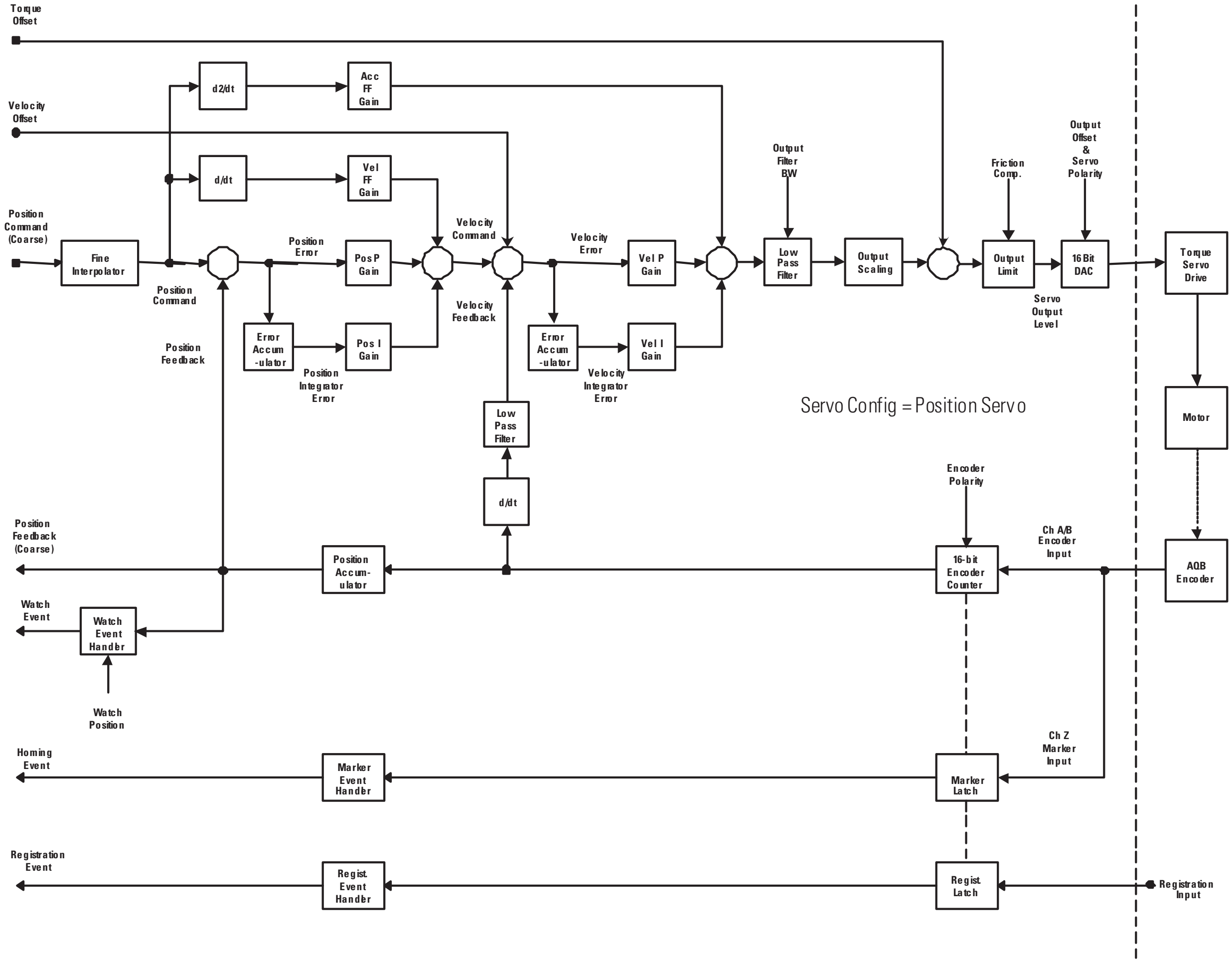
Position Servo with Torque Servo Drive



This configuration gives full position servo control using an external torque loop servo drive. Synchronous input data to the servo loop includes Position Command, Velocity Offset, and Torque Offset. The controller updates these values at the coarse update period of the motion group.

The Position Command value is derived directly from the output of the motion planner, while the Velocity Offset and Torque Offset values are derived from the current value of the corresponding attributes.

Position Servo with Velocity Servo Drive



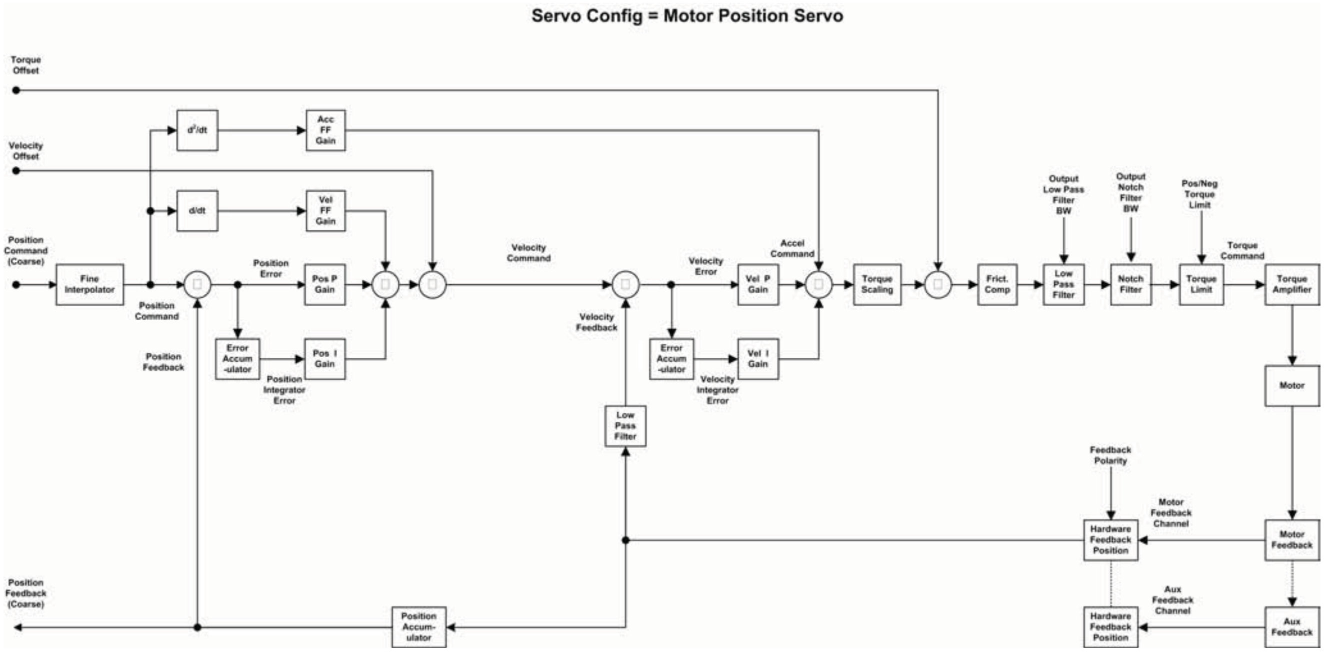
This configuration provides full position servo control using an external velocity loop servo drive.

Note that in this configuration the servo module does not close the velocity loop, but rather the drive does. Synchronous input data to the servo loop includes Position Command and Velocity Offset. (Torque Offset is ignored.) The controller updates these values at the coarse update period of the motion group.

The Position Command value is derived directly from the output of the motion planner, while the Velocity Offset value is derived from the current value of the corresponding attributes.

AXIS_SERVO_DRIVE

Motor Position Servo

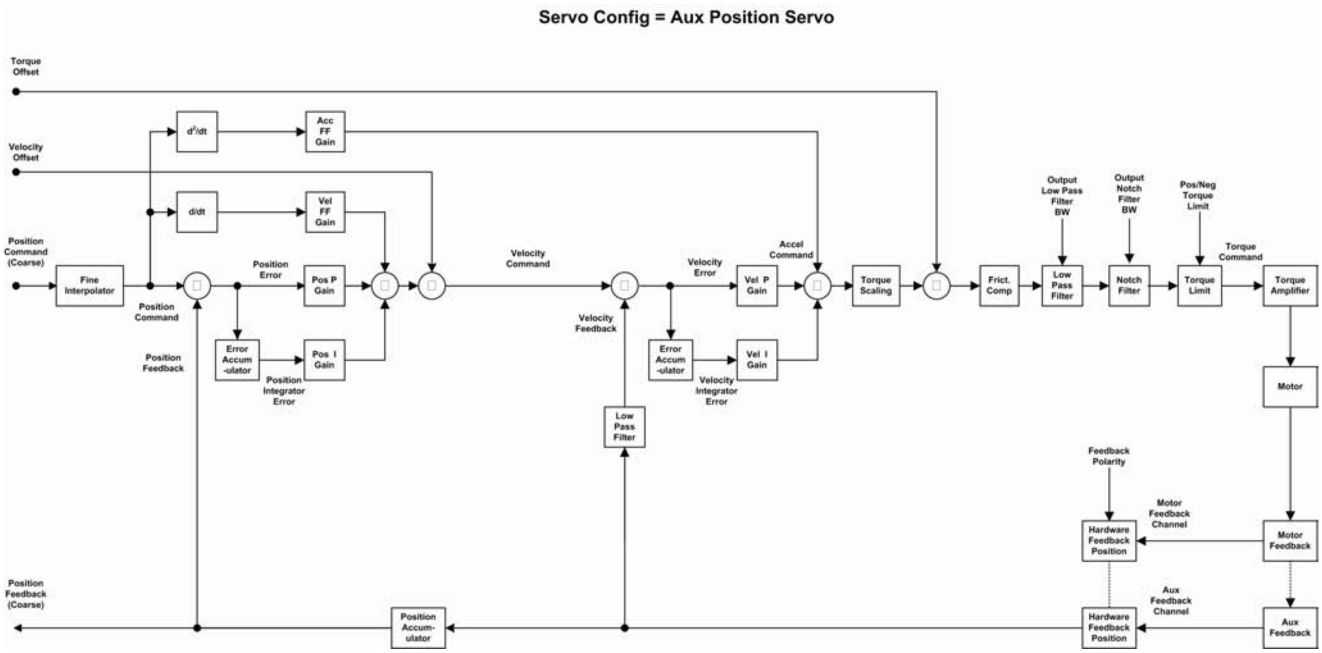


The Motor Position Servo configuration provides full position servo control using only the motor mounted feedback device to provide position and velocity feedback. This servo configuration is a good choice in applications where smoothness and stability are more important that positioning accuracy. Positioning accuracy is limited due to the fact that the controller has no way of compensating for non-linearity in the mechanics external to the motor.

Note that the motor mounted feedback device also provides motor position information necessary for commutation. Synchronous input data to the servo loop includes Position Command, Velocity Offset, and Torque Offset. These values are updated at the coarse update rate of the associated motion group.

The Position Command value is derived directly from the output of the motion planner, while the Velocity Offset and Torque Offset values are derived from the current value of the corresponding attributes. These offset attributes may be changed programmatically via SSV instructions or direct Tag access which, when used in conjunction with future Function Block programs, provides custom 'outer' control loop capability.

Auxiliary Position Servo

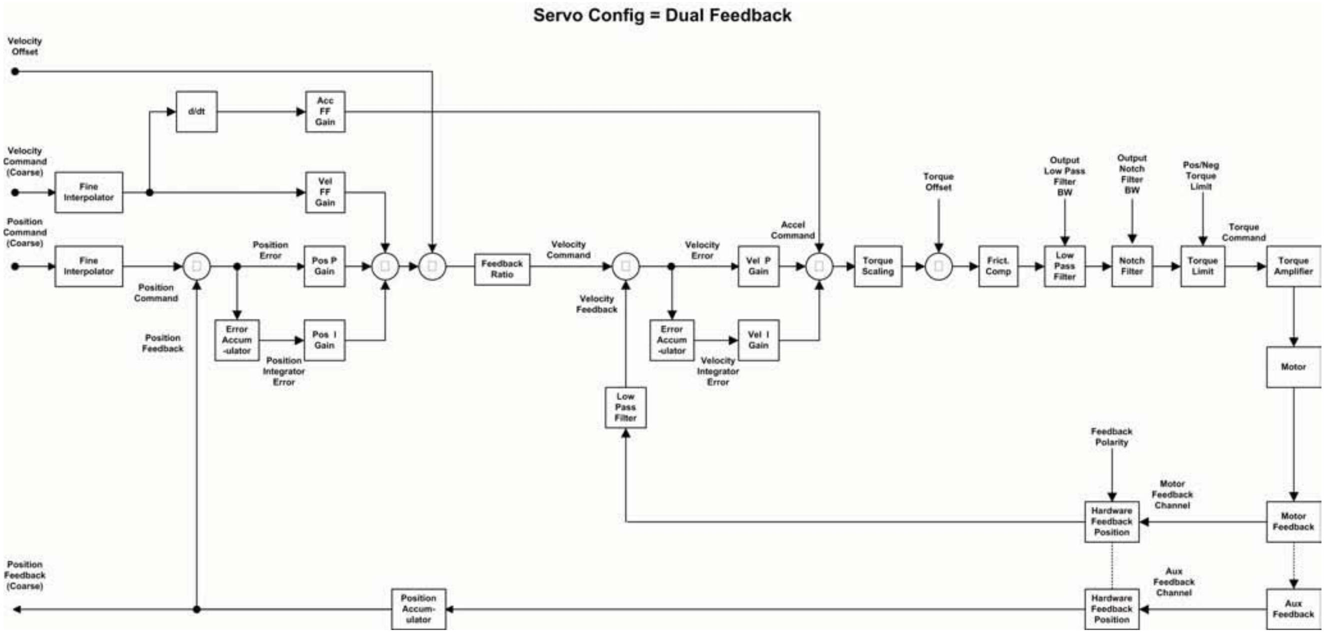


The Auxiliary Position Servo configuration provides full position servo control using an auxiliary (that is, external to the motor) feedback device to provide position and velocity feedback. This servo configuration is a good choice in applications positioning accuracy is important. The smoothness and stability may be limited, however, due to the mechanical non-linearities external to the motor.

Note that the motor mounted feedback device is still required to provide motor position information necessary for commutation. Synchronous input data to the servo loop includes Position Command, Velocity Offset, and Torque Offset. These values are updated at the coarse update rate of the associated motion group.

The Position Command value is derived directly from the output of the motion planner, while the Velocity Offset and Torque Offset values are derived from the current value of the corresponding attributes. These offset attributes may be changed programmatically via SSV instructions or direct Tag access which, when used in conjunction with future Function Block programs, provides custom ‘outer’ control loop capability.

Dual Position Servo

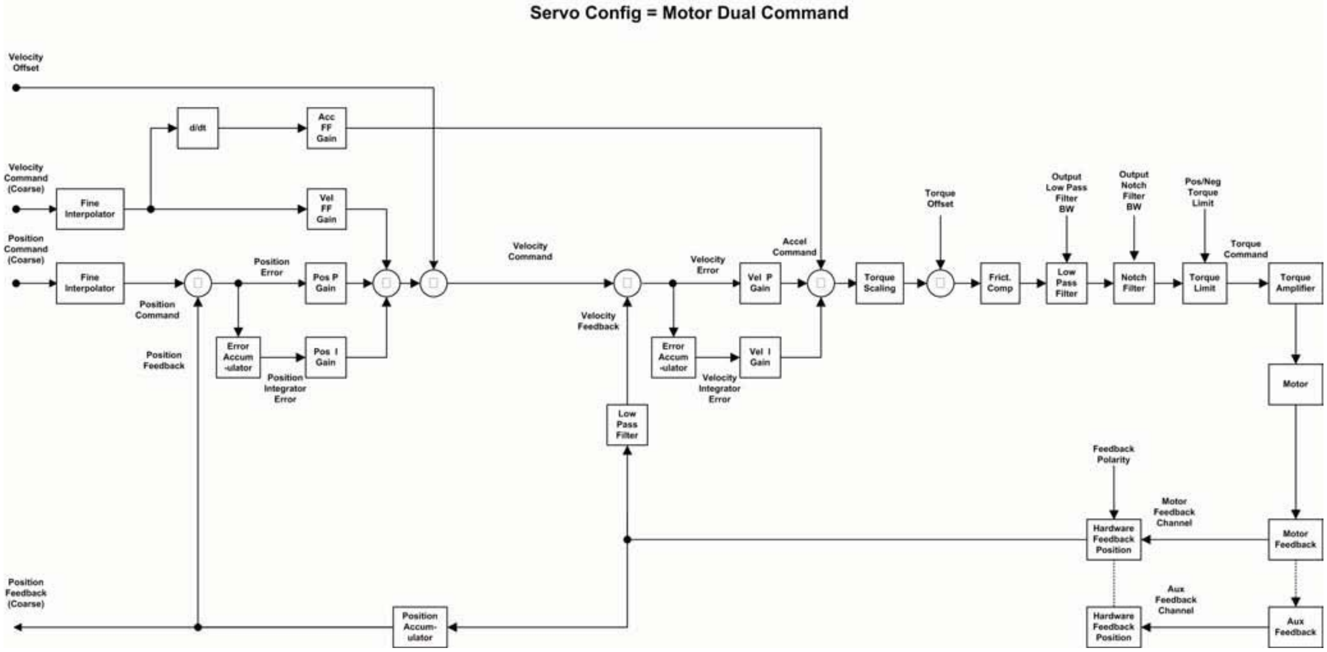


This configuration provides full position servo control using the auxiliary feedback device for position feedback and the motor mounted feedback device to provide velocity feedback. This servo configuration combines the advantages of accurate positioning associated with the auxiliary position servo with the smoothness and stability of the motor position servo configuration.

Note that the motor mounted feedback device also provides motor position information necessary for commutation. Synchronous input data to the servo loop includes Position Command, Velocity Offset, and Torque Offset. These values are updated at the coarse update rate of the associated motion group.

The Position Command value is derived directly from the output of the motion planner, while the Velocity Offset and Torque Offset values are derived from the current value of the corresponding attributes. These offset attributes may be changed programmatically via SSV instructions or direct Tag access which, when used in conjunction with future Function Block programs, provides custom 'outer' control loop capability.

Motor Dual Command Servo

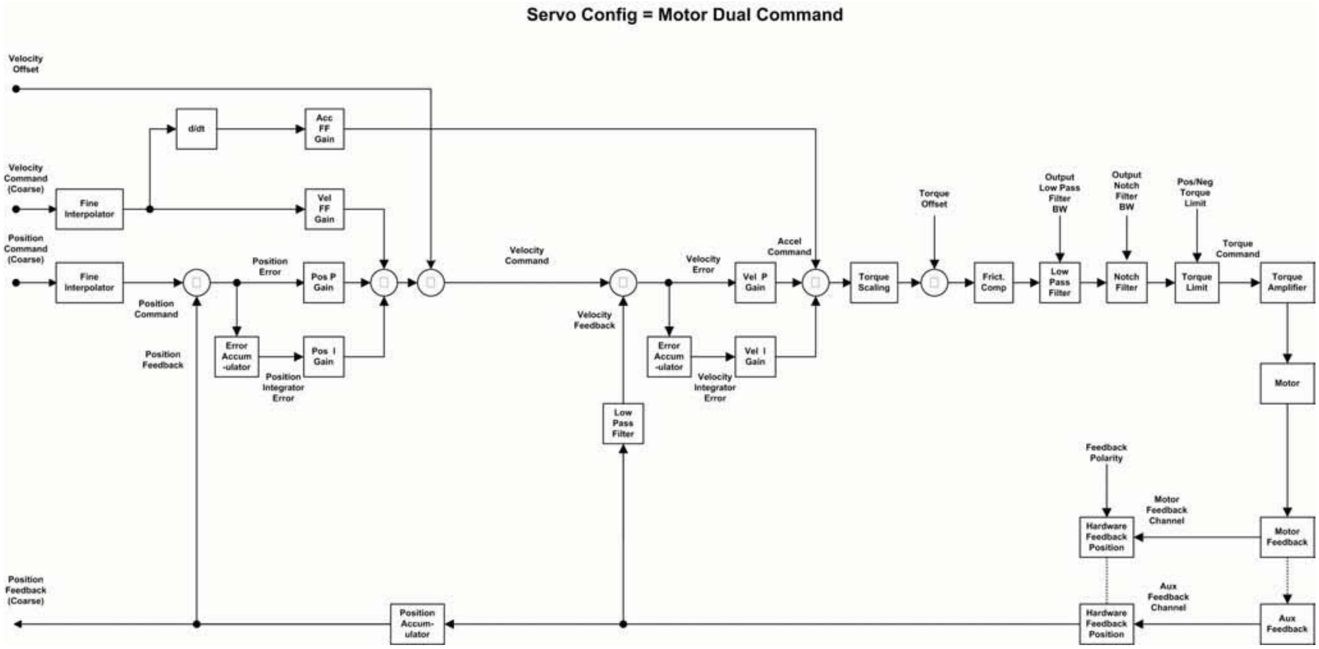


The Motor Dual Command Servo configuration provides full position servo control using only the motor mounted feedback device to provide position and velocity feedback. Unlike the Motor Position Servo configuration, however, both command position and command velocity are applied to the loop to provide smoother feedforward behavior. This servo configuration is a good choice in applications where smoothness and stability are important. Positioning accuracy is limited due to the fact that the controller has no way of compensating for non-linearities in the mechanics external to the motor.

Note that the motor mounted feedback device also provides motor position information necessary for commutation. Synchronous input data to the servo loop includes Position Command, Velocity Command, and Velocity Offset. These values are updated at the coarse update rate of the associated motion group.

The Position and Velocity Command values are derived directly from the output of the motion planner, while the Velocity Offset value is derived from the current value of the corresponding attributes. The velocity offset attribute may be changed programmatically via SSV instructions or direct Tag access which, when used in conjunction with future Function Block programs, provides custom 'outer' control loop capability.

Auxiliary Dual Command Servo



The Auxiliary Dual Command Servo configuration provides full position servo control using only the auxiliary mounted feedback device to provide position and velocity feedback. Unlike the Auxiliary Position Servo configuration, however, both command position and command velocity are applied to the loop to provide smoother feedforward behavior. This servo configuration is a good choice in applications where positioning accuracy and good feedforward performance is important. The smoothness and stability may be limited, however, due to the mechanical non-linearities external to the motor.

Note, that the motor mounted feedback device is still required to provide motor position information necessary for commutation. Synchronous input data to the servo loop includes Position Command, Velocity Command, and Velocity Offset. These values are updated at the coarse update rate of the associated motion group.

The Position and Velocity Command values are derived directly from the output of the motion planner, while the Velocity Offset value is derived from the current value of the corresponding attributes. The velocity offset attribute may be changed programmatically via SSV instructions or direct Tag access which, when used in conjunction with future Function Block programs, provides custom 'outer' control loop capability.

Dual Command Feedback Servo

The Motor Dual Command Feedback Servo configuration provides full position servo control using the auxiliary feedback device for position feedback and the motor mounted feedback device to provide velocity feedback. Unlike the Dual Feedback Servo configuration, however, both command position and command velocity are also applied to the loop to provide smoother feedforward behavior. This servo configuration is a good choice in applications where smoothness and stability are important as well as positioning accuracy.

Note, that the motor mounted feedback device is still required to provide motor position information necessary for commutation. Synchronous input data to the servo loop includes Position Command, Velocity Command, and Velocity Offset. These values are updated at the coarse update rate of the associated motion group.

The Position and Velocity Command values are derived directly from the output of the motion planner, while the Velocity Offset value is derived from the current value of the corresponding attributes. The velocity offset attribute may be changed programmatically via SSV instructions or direct Tag access which, when used in conjunction with future Function Block programs, provides custom ‘outer’ control loop capability.

Velocity Servo

The Velocity Servo configuration provides velocity servo control using the motor mounted feedback device. Synchronous input data to the servo loop includes Velocity Command, Velocity Offset, and Torque Offset. These values are updated at the coarse update rate of the associated motion group. The Velocity Command value is derived directly from the output of the motion planner, while the Velocity Offset and Torque Offset values are derived from the current value of the corresponding attributes. These offset attributes may be changed programmatically via SSV instructions or direct Tag access which, when used in conjunction with future Function Block programs, provides custom ‘outer’ control loop capability.

Torque Servo

The Torque Servo configuration provides torque servo control using only the motor mounted feedback device for commutation. Synchronous input data to the servo loop includes only the Torque Offset. This values are updated at the coarse update rate of the associated motion group.

The Torque Offset value is derived from the current value of the corresponding attribute. This offset attribute may be changed programmatically via SSV instructions or direct Tag access which, when used in conjunction with future Function Block programs, provides custom 'outer' control loop capability.

Drive Gains

Rockwell Automation servo drives use Nested Digital Servo Control Loop such as shown in the block diagrams above, consisting typically of a position loop with proportional, integral and feed-forward gains around a digitally synthesized inner velocity loop, again with proportional and integral gains for each axis.

These gains provide software control over the servo dynamics, and allow the servo system to be completely stabilized. Unlike analog servo controllers, these digitally set gains do not drift. Furthermore, once these gains are set for a particular system, another SERCOS module programmed with these gain values will operate identically to the original one.

Notes:

Notes:

Rockwell Automation Support

Rockwell Automation provides technical information on the Web to assist you in using its products. At http://www.rockwellautomation.com/support/, you can find technical manuals, a knowledge base of FAQs, technical and application notes, sample code and links to software service packs, and a MySupport feature that you can customize to make the best use of these tools.

For an additional level of technical phone support for installation, configuration, and troubleshooting, we offer TechConnect support programs. For more information, contact your local distributor or Rockwell Automation representative, or visit http://www.rockwellautomation.com/support/.

Installation Assistance

If you experience an anomoly within the first 24 hours of installation, review the information that is contained in this manual. You can contact Customer Support for initial help in getting your product up and running.

United States or Canada	1.440.646.3434
Outside United States or Canada	Use the Worldwide Locator at http://www.rockwellautomation.com/support/americas/phone_en.html, or contact your local Rockwell Automation representative.

New Product Satisfaction Return

Rockwell Automation tests all of its products to ensure that they are fully operational when shipped from the manufacturing facility. However, if your product is not functioning and needs to be returned, follow these procedures.

United States	Contact your distributor. You must provide a Customer Support case number (call the phone number above to obtain one) to your distributor to complete the return process.
Outside United States	Please contact your local Rockwell Automation representative for the return procedure.

Documentation Feedback

Your comments will help us serve your documentation needs better. If you have any suggestions on how to improve this document, complete this form, publication RA-DU002, available at http://www.rockwellautomation.com/literature/.

www.rockwellautomation.com

Power, Control and Information Solutions Headquarters

Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
Europe/Middle East/Africa: Rockwell Automation, Vorstlaan/Boulevard du Souverain 36, 1170 Brussels, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia Pacific: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Supersedes Publication LOGIX-UM002D-EN-P - July 2008