

**CONTROL DE MOTORES DC UTILIZANDO VARIAS TARJETAS
DE ADQUISICIÓN DE DATOS NI 6008/6009**

ABELARDO ANDRÉS VESGA SEPÚLVEDA

**UNIVERSIDAD PONTIFICIA BOLIVARIANA
ESCUELA INGENIERÍA
FACULTAD DE INGENIERÍA ELECTRÓNICA
BUCARAMANGA
2013**

**CONTROL DE MOTORES DC UTILIZANDO VARIAS TARJETAS
DE ADQUISICIÓN DE DATOS NI 6008/6009**

ABELARDO ANDRÉS VESGA SEPÚLVEDA
Estudiante de Ingeniería Electrónica

**Trabajo de Grado presentado como requisito para optar al título de
Ingeniero Electrónico**

Director:
EDGAR BARRIOS URUEÑA
Msc. Ingeniería Eléctrica

UNIVERSIDAD PONTIFICIA BOLIVARIANA
ESCUELA INGENIERÍA
FACULTAD DE INGENIERÍA ELECTRÓNICA
BUCARAMANGA
2013

Nota de Aceptación

Firma del Presidente del Jurado

Firma del Jurado

Firma del Jurado

Bucaramanga, 06 de Marzo de 2013

DEDICATORIA

Como primera medida, a papito Dios; ya que sin su presencia nada de esto fuera posible.

A mis padres y a mi hermano, que con su apoyo constante, amor, dedicación, me dieron las fuerzas necesarias para seguir adelante y culminar éste trabajo.

A mi novia, que siempre estuvo ahí pendiente dándome fuerzas y alentándome día a día. Gracias.

AGRADECIMIENTOS

A papito Dios, por proveer todo lo necesario para mi vida.

Al Ingeniero Edgar Barrios, por su apoyo, confianza, paciencia y colaboración en el desarrollo de éste trabajo.

A los Ingenieros: Fabio Guzmán, Alex Monclou, Juan Carlos Villamizar, Carlos Gerardo Hernández y Luis Ángel Silva por su apoyo moral y motivación.

A mis amigos que siempre estuvieron cerca motivándome, alentándome y brindándome su apoyo.

A mi familia por la paciencia y confianza.

CONTENIDO

	Pág.
INTRODUCCIÓN	13
OBJETIVOS	14
OBJETIVO GENERAL	14
OBJETIVOS ESPECIFICOS	14
1. MARCO TEÓRICO	15
1.1. ENTORNO DE TRABAJO MATLAB	15
1.2. TOOLBOX DE ADQUISICION DE DATOS	15
1.3. SISTEMA DE ADQUISICIÓN DE DATOS	16
1.4. TARJETA NI USB-6008/6009	17
1.5. MOTORES DC CON ENCODER	19
1.6. ENCODER	20
2. METODOLOGIA	22
2.1. INVESTIGACIÓN Y DOCUMENTACIÓN	22
2.2. RECONOCIMIENTO DE LOS INSTRUMENTOS	22
2.3. AMPLIFICACIÓN DE SEÑALES	23
2.4. ACONDICIONAMIENTO DE SEÑALES	23
2.5. MUESTREO Y TRANSMISIÓN DE DATOS	23
2.6. PROGRAMACIÓN EN MATLAB	23
2.7. MANUFACTURA DEL CIRCUITO IMPRESO	23
2.8. DESARROLLO DE PRUEBAS	23
2.9. EVALUACIÓN DE TIEMPOS DE RESPUESTA	24
2.10. REDACCIÓN DEL INFORME FINAL	24
3. RESULTADOS Y DISCUSIÓN	25
3.1. Lectura de Posición (Entrada)	26
3.2. Control de posición del motor DC (Salida)	26
3.3. Control del Motor DC	30
3.4. External Reference (+2.5 V)	30
3.5. Diseños en simulink para medir tiempos de respuesta con la tarjeta de adquisición de datos NI-USB 6008/6009	32
3.5.1. Modificación de los parámetros de configuración de simulación en Simulink	32
3.5.2. Pruebas con varias entradas y varias salidas análogas de la tarjeta de adquisición de datos NI-USB 6008/6009	38
3.5.2.1. Una salida análoga	40
3.5.2.2. Dos salidas análogas	41
3.5.2.3. Una salida y una entrada análogas	42
3.5.2.4. Una salida y ocho entradas análogas	43
3.5.2.5. Dos salidas y una entrada análogas	44
3.5.2.6. Dos salidas y ocho entradas análogas	45

3.6. Evaluación del comportamiento del control de 2 motores DC con 2 tarjetas de adquisición de datos NI-USB 6008/6009	47
3.6.1. Evaluación de la velocidad de respuesta de control utilizando una tarjeta NI-USB 6008/6009 y un motor DC (una entrada y una salida de la tarjeta)	47
3.6.2. Evaluación de la velocidad de respuesta de control utilizando una tarjeta NI-USB 6008/6009 y dos motores DC (dos entradas y dos salidas de la tarjeta)	57
3.6.3. Evaluación de la velocidad de respuesta de control utilizando dos tarjetas NI-USB 6008/6009 y dos motores DC (una entrada y una salida de cada tarjeta)	64
4. CONCLUSIONES Y RECOMENDACIONES	74
BIBLIOGRAFÍA	76

LISTA DE FIGURAS

	Pág.
Figura 1. Relación entre los componentes de un sistema de adquisición de datos	17
Figura 2. Hardware de la tarjeta	19
Figura 3. Motor DC	200
Figura 4. Variedades de encoder	21
Figura 5. Funcionamiento del encoder	251
Figura 6. Diseño esquemático general implementado en el proyecto	25
Figura 7. Circuito impreso del driver de los motores	27
Figura 8. Entrada de señal análoga de la tarjeta NI-USB 6008/6009	29
Figura 9. Salida A0 de la tarjeta NI-USB 6008/6009	310
Figura 10. Diseño circuital del control y sus componentes	321
Figura 11. Diseño de nuestro sistema de control	32
Figura 12. Señal diente de sierra medida en la entrada análoga ai0 de la tarjeta NI-USB 6008/6009	33
Figura 13. Modificación de los parámetros de configuración de la simulación en Simulink paso 1	34
Figura 14. Modificación de los parámetros de configuración de la simulación en Simulink paso 2	35
Figura 15. Modificación de los parámetros de configuración de la simulación en Simulink paso 3	36
Figura 16. Señal diente de sierra medida en la entrada análoga ai0 de la tarjeta NI-USB 6008/6009 en tiempo discreto	37
Figura 17. Diseño en Simulink. Utilización de varias entradas y varias salidas análogas de las tarjetas NI USB 6008/6009	39
Figura 18. Utilización de una salida análoga de la tarjeta NI-USB 6008/6009 para evaluar el periodo de la señal	40
Figura 19. Utilización de dos salidas análogas de la tarjeta NI-USB 6008/6009 para evaluar el periodo de la señal	41
Figura 20. Utilización de una salida y una entrada análogas de la tarjeta NI-USB 6008/6009 para evaluar el periodo de la señal	42
Figura 21. Utilización de una salida y ocho entradas análogas de la tarjeta NI-USB 6008/6009 para evaluar el periodo de la señal	43
Figura 22. Utilización de dos salidas y una entrada análogas de la tarjeta NI-USB 6008/6009 para evaluar el periodo de la señal	44
Figura 23. Utilización de dos salidas y ocho entrada análogas de la tarjeta NI-USB 6008/6009 para evaluar el periodo de la señal	45
Figura 24. Diseño en Simulink del control PID de velocidad de un motor DC por medio de una tarjeta NI-USB 6008/6009	48
Figura 25. Configuración de los parámetros de las entradas análogas de la tarjeta NI-USB 6008/6009	49

Figura 26. Configuración de los parámetros de las salidas análogas de la tarjeta NI-USB 6008/6009	50
Figura 27. Evaluación del tiempo de respuesta para un set-point positivo	51
Figura 28. Evaluación del tiempo de respuesta para un set-point negativo	52
Figura 29. Evaluación del tiempo de respuesta para el paso de un set-point positivo a un set-point negativo	53
Figura 30. Evaluación del tiempo de respuesta para el paso de un set-point negativo a un set-point positivo	54
Figura 31. Evaluación del tiempo de respuesta para el paso de un set-point positivo alto a uno más bajo	55
Figura 32. Evaluación del tiempo de respuesta para el paso de un set-point negativo a uno menos negativo	56
Figura 33. Diseño en Simulink del control PID de velocidad de dos motores DC por medio de una tarjeta NI-USB 6008/6009	57
Figura 34. Evaluación del tiempo de respuesta para un set-point positivo	58
Figura 35. Evaluación del tiempo de respuesta para un set-point negativo	59
Figura 36. Evaluación del tiempo de respuesta para el paso de un set-point positivo a un set-point negativo	60
Figura 37. Evaluación del tiempo de respuesta para el paso de un set-point negativo a un set-point positivo	61
Figura 38. Evaluación del tiempo de respuesta para el paso de un set-point positivo alto a uno más bajo	62
Figura 39. Evaluación del tiempo de respuesta para el paso de un set-point negativo a uno menos negativo	63
Figura 40. Diseño en Simulink del control PID de velocidad de dos motores DC por medio de dos tarjetas NI-USB 6008/6009	65
Figura 41. Evaluación del tiempo de respuesta para un set-point positivo	66
Figura 42. Evaluación del tiempo de respuesta para un set-point negativo	67
Figura 43. Evaluación del tiempo de respuesta para el paso de un set-point positivo a un set-point negativo	68
Figura 44. Evaluación del tiempo de respuesta para el paso de un set-point negativo a un set-point positivo	69
Figura 45. Evaluación del tiempo de respuesta para el paso de un set-point positivo alto a uno más bajo	70
Figura 46. Evaluación del tiempo de respuesta para el paso de un set-point negativo a uno menos negativo	71

LISTA DE TABLAS

	Pág.
Tabla 1. Diferencias entre las tarjetas NI-USB 6008 y 6009	18
Tabla 2. Comparación del periodo de una señal en modo continuo y discreto	38
Tabla 3. Evaluación del periodo de las señales de salida para varias entradas y para varias salidas de la tarjeta de adquisición de datos NI-USB 6008/6009	46
Tabla 4. Resultado de las pruebas realizadas para evaluar la velocidad de respuesta de control utilizando una tarjeta NI-USB 6008/6009y un motor DC (una entrada y una salida de la tarjeta)	57
Tabla 5. Resultado de las pruebas realizadas para evaluar la velocidad de respuesta de control utilizando una tarjeta NI-USB 6008/6009 y dos motores DC (dos entradas y dos salidas de la tarjeta)	64
Tabla 6. Resultado de las pruebas realizadas para evaluar la velocidad de comportamiento del control PID de los motores	72

RESUMEN GENERAL DEL TRABAJO DE GRADO

TITULO: CONTROL DE MOTORES DC UTILIZANDO VARIAS TARJETAS DE ADQUISICIÓN DE DATOS NI 6008/6009
AUTOR: Abelardo Andrés Vesga Sepúlveda
FACULTAD: Facultad de Ingeniería Electrónica
DIRECTOR: Msc. Edgar Barrios Urueña

RESUMEN

Esta tesis tuvo como objetivo general diseñar e implementar un sistema que permita el control de posición de varios motores DC, empleando la Toolbox de adquisición de datos de Matlab y varias tarjetas NI-USB 6008/6009.

Se controló 2 motores DC por medio de 2 tarjetas de adquisición de datos NI-USB 6008/6009. Se estudió la Toolbox de adquisición de datos de Matlab por medio de 2 proyectos de grado realizados en la Universidad Pontificia Bolivariana seccional Bucaramanga, tales como [1] y [2], así como la información original de la Toolbox de adquisición de datos de Matlab. Se diseñó un sistema en Simulink con el cual se evaluó los tiempos de respuesta de las tarjetas NI-USB 6008/6009 probando varias entradas y varias salidas de las tarjetas. Se diseñó y construyó los drivers de los motores que incluyó amplificación y acondicionamiento de las señales provenientes de las tarjetas NI-USB 6008/6009. Se diseñó un control PID en Simulink con el cual se realizó el control de posición de los motores DC. Se evaluó los tiempos de respuesta utilizando varios set-points, donde se observó que el software utilizado para realizar el control (Matlab) responde bastante lento, por lo mismo se concluyó que no es un software óptimo para realizar este tipo de control.

PALABRAS CLAVES: Toolbox, simulink.

GENERAL SUMMARY OF WORK OF DEGREE

TITLE: CONTROL OF DC MOTORS USING MULTIPLE DATA ACQUISITION CARDS NI 6008/6009
AUTHOR: Abelardo Andrés Vesga Sepúlveda
FACULTY: Facultad de Ingeniería Electrónica
DIRECTOR: Msc. Edgar Barrios Urueña

ABSTRACT

This thesis aimed to overall design and implements a system to control the position of multiple DC motors, using the data acquisition toolbox of Matlab and several cards NI-USB 6008/6009.

Two DC motors were controlled by two data acquisition cards NI-USB 6008/6009 cards. Data acquisition toolbox of Matlab was studied through two grade projects conducted at the University Pontificia Bolivariana Bucaramanga, such as [5] and [6] as well as the original information of the data acquisition toolbox of Matlab. A system in Simulink was designed to test response times of the NI-USB 6008/6009 cards testing several inputs and several outputs of the cards. The engine drivers were designed and built which included amplification and signal conditioning coming from the NI-USB 6008/6009 cards. PID control was designed in Simulink which was conducted with the control position of DC motors. Response times were evaluated using several set-points, where it was observed that the software used to make control (Matlab) responded quite slow, so it was concluded that Matlab is not an optimal software for this type of control.

KEYWORDS: Toolbox, simulink

INTRODUCCIÓN

En los últimos años en la Universidad Pontificia Bolivariana se ha venido implementando el uso de Matlab como software de adquisición de datos dado que éste es un software conocido por los estudiantes de Ingeniería Electrónica desde los primeros niveles de estudio, haciendo su utilización y aprendizaje algo más sencillo y más cómodo; así mismo el uso de las tarjetas de adquisición de datos NI USB 6008/6009 de National Instruments como interfaz física en la adquisición de señales eléctricas a estudiar. En este trabajo se propuso inicialmente evaluar el comportamiento del control de posición de 2 motores DC por medio de la Toolbox de adquisición de datos de Matlab y la utilización de 2 tarjetas NI-USB 6008/6009. Se realizó el estudio teórico y práctico de la Toolbox de adquisición de datos de Matlab por medio de [1] y [2], para entender su funcionamiento con miras a controlar el arranque, el frenado, la velocidad y la posición de varios motores DC. Se implementó 2 tarjetas NI-USB 6008/6009 para la adquisición de señales eléctricas provenientes de 2 motores con encoder y se adecuó las señales de salida para evaluar el comportamiento del control de los motores. Se sugiere cambiar el control de posición propuesto inicialmente por el control de velocidad, ya que el control de posición no se logró debido a que al momento de entregar datos, el sistema es muy lento y los motores giran a su máxima velocidad para alcanzar el set-point deseado.

OBJETIVOS

OBJETIVO GENERAL

- Diseñar e implementar un sistema que permita el control de posición de varios motores dc, empleando la Toolbox de adquisición de datos de Matlab y utilizando varias tarjetas NI-USB 6008/6009.

OBJETIVOS ESPECÍFICOS

- Realizar un estudio teórico y práctico del funcionamiento de la Toolbox de Adquisición de Datos de Matlab.
- Realizar procedimientos para controlar el arranque, el frenado, la velocidad y la posición de varios motores, implementados a través de Matlab.
- Implementar varias tarjetas NI-USB 6008/6009 para realizar la adquisición de señales eléctricas de dos o más motores dc con encoder.
- Acondicionamiento de la salida de 0 a 5 Voltios de la tarjeta NI-USB 6008/6009 para convertirla en una salida de -24 a 24 Voltios.
- Evaluar el comportamiento del control de los motores dc.

1. MARCO TEÓRICO

A continuación se exponen los conceptos y elementos más significativos para el desarrollo del proyecto.

1.1. ENTORNO DE TRABAJO MATLAB

MATLAB es un entorno de computación y desarrollo de aplicaciones totalmente integrado, orientado para el desarrollo de proyectos con elevados cálculos matemáticos y la visualización gráfica de estos. MATLAB integra análisis numérico, cálculo matricial, procesado de señal, todo ello en un entorno fácil para el usuario.

MATLAB consta de diferentes aplicaciones o toolboxes especializados orientados a ingenieros, científicos y todo tipo de profesionales técnicos. Entre ellos destacan: Sistemas de Control, Adquisición de Datos, Tiempo Real, Lógica Fuzzy, Procesamiento de Imágenes, Redes Neuronales, Optimización, Procesamiento de Señal, etc [3].

1.2. TOOLBOX DE ADQUISICION DE DATOS

La Toolbox de Adquisición de Datos es una colección de funciones de archivos .m y archivos .mex construido sobre el entorno de MATLAB. La Toolbox también incluye varios vínculos de librerías dinámicas (DLL's) llamados adaptadores, los cuales permiten interactuar con un hardware específico. La Toolbox tiene las siguientes características:

Un marco para traer datos reales, medidos en el espacio de trabajo de MATLAB usando un hardware de adquisición de datos compatible con el PC.

Soporte para entradas análogas (AI), salidas análogas (AO) y entradas y salidas digitales (DIO). Incluyendo conversiones simultáneas para entradas y salidas análogas.

Soporte para estos tipos de hardware más populares:

- Advantech®. Tarjetas que usen el administrador de dispositivos de Advantech.
- Módulos VXI E1432A/33A/34A de Agilent Technologies®.
- Keithley®. Tarjetas que usen DriverLINX.
- Tarjetas de Measurement Computing™ Corporation.

- National Instruments®. Tarjetas que usan el software tradicional NI-DAQ o NI-DAQmx.
- Puertos paralelos LPT1 y LPT3.
- Microsoft® Windows®. Tarjetas de sonido.

Adicionalmente, se puede usar un Kit de adaptador de interface para dispositivos que no soportan la Toolbox.

Eventos de adquisiciones dirigidas [4].

1.3. SISTEMA DE ADQUISICIÓN DE DATOS

El objetivo de cualquier sistema de adquisición de datos es proporcionar las herramientas y recursos necesarios para tomar señales físicas y convertirlas en datos que posteriormente se puedan procesar y mostrar.

Un sistema de adquisición de datos se podría tomar como un grupo de hardware y software que permiten interactuar con el mundo real, consta de estos componentes:

Hardware de Adquisición: Es el corazón de cualquier sistema de adquisición de datos. La función principal es hacer la conversión de señales análogas a señales digitales y señales digitales a análogas. Conversión A/D y D/A.

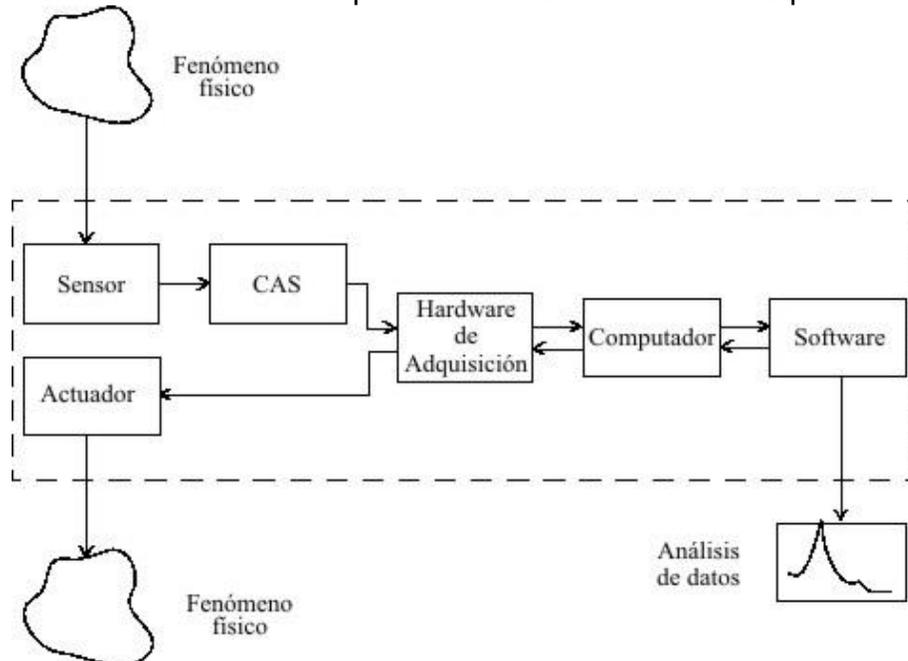
Sensores y Actuadores (Transductores): Un transductor es un dispositivo que convierte un tipo de energía de entrada en una energía de salida de otra forma.

Acondicionador de señal: Las señales de los sensores a menudo son incompatibles con el hardware de adquisición de datos, y para superar esto las señales deben ser acondicionadas. Por ejemplo, las señales podrían ser amplificadas o volverlas en señales sin componentes de frecuencias indeseadas. Las señales de salida también pueden ser acondicionadas.

Computador: Proporciona un procesador, un sistema de reloj, un bus de datos, memoria y espacio en el disco para almacenar datos.

Software: Permite el intercambio de información entre el computador y el hardware [5].

Figura 1. Relación entre los componentes de un sistema de adquisición de datos



Fuente: Data Acquisition Toolbox User's Guide. Copyright 1999-2007 by the Mathworks, Inc. Disponible en: <http://www.mathworks.com>

1.4. TARJETA NI USB-6008/6009

La tarjeta NI USB-6008/6009 brinda una conexión de 8 canales de entradas análogas (AI), dos canales de salidas análogas (AO), 12 canales de entrada/salida digitales y un contador de 32-bit con una interfaz USB de alta velocidad.

Tabla 1. Diferencias entre las tarjetas NI-USB 6008 y 6009

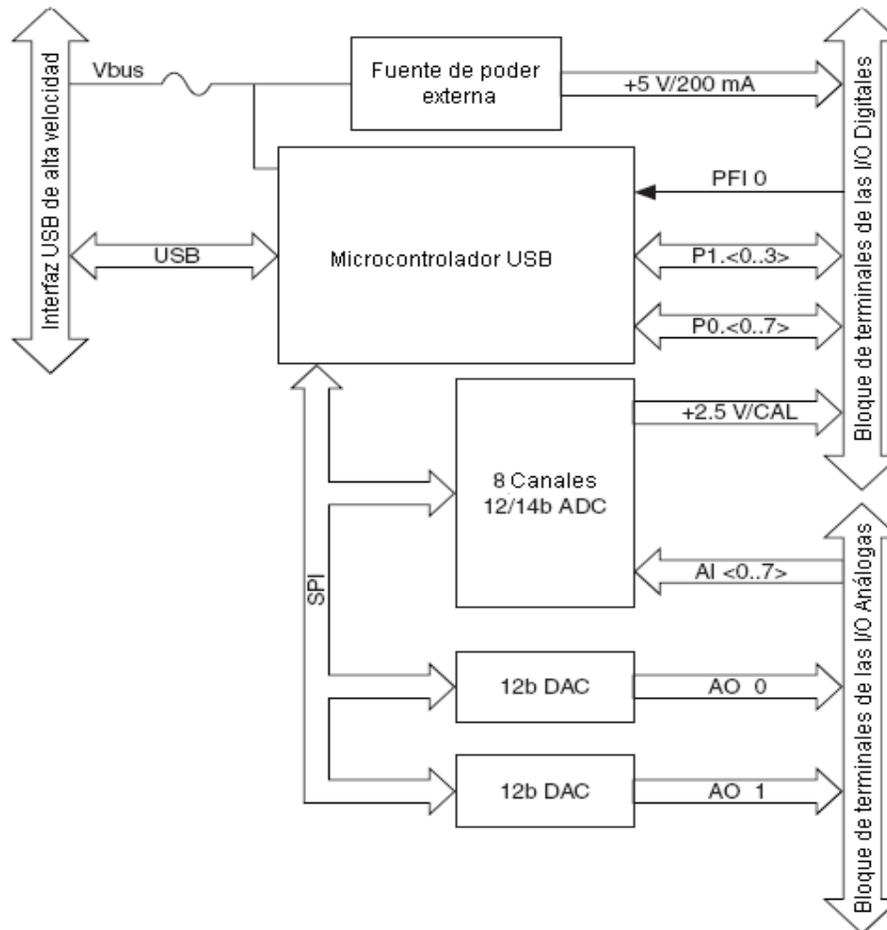
Característica	USB 6008	USB 6009
Resolución en las entradas análogas	12 Bits en modo diferencial, 11 Bits para modo de nodo simple	14 Bits en modo diferencial, 13 Bits para modo de nodo simple
Máximo frecuencia de muestreo en las entradas análogas Canal individual*	10 kS/s	48 kS/s
Máxima frecuencia de muestreo en las entradas análogas Múltiples canales (Total*)	10 kS/s	42 kS/s
Configuración DIO	Colector abierto	Colector abierto o control activo
*Pueden depender del sistema		

Fuente: User guide and specifications USB-6008/6009. Disponible en:
http://www.tau.ac.il/~electro/pdf_files/computer/ni_6008_ADC_manual.pdf

Software de la Tarjeta: El software que soporta USB-6008 para Windows 2000/XP es NI-DAQmx.

Hardware: El siguiente diagrama de bloques muestra los principales componentes de las NI-USB 6008/6009 [6].

Figura 2. Hardware de la tarjeta



Fuente: User guide and specifications USB-6008/6009. Disponible en: http://www.tau.ac.il/~electro/pdf_files/computer/ni_6008_ADC_manual.pdf

1.5. MOTORES DC CON ENCODER

El motor de corriente continua es una máquina que convierte la energía eléctrica en mecánica, principalmente mediante el movimiento rotatorio. En la actualidad existen nuevas aplicaciones con motores eléctricos que no producen movimiento rotatorio, sino que con algunas modificaciones, ejercen tracción sobre un riel. Estos motores se conocen como motores lineales.

Esta máquina de corriente continua es una de las más versátiles en la industria. Su fácil control de posición, paro y velocidad la han convertido en una de las mejores opciones en aplicaciones de control y automatización de procesos. Pero con la llegada de la electrónica su uso ha disminuido en gran medida, pues los motores de corriente alterna, del tipo asíncrono, pueden ser controlados de

igual forma a precios más accesibles para el consumidor medio de la industria. A pesar de esto los motores de corriente continua se siguen utilizando en muchas aplicaciones de potencia (trenes y tranvías) o de precisión (máquinas, micro motores, etc.)

La principal característica del motor de corriente continua es la posibilidad de regular la velocidad desde vacío a plena carga [7].

Figura 3. Motor DC

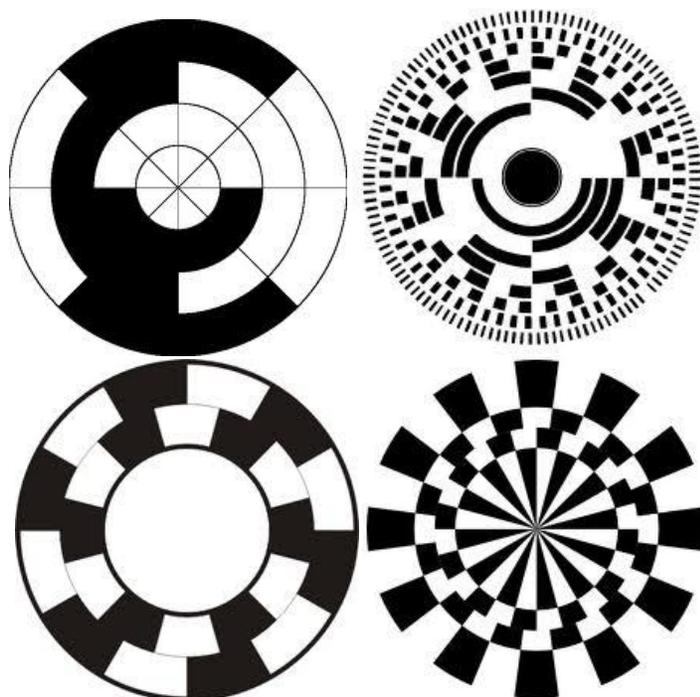


Fuente: FITZGERALD, A. E., CHARLES KINGSLEY, STEPHEN D. UMANS.
Máquinas Eléctricas. 6ª Edición. Editorial MC GRAW HILL

1.6. ENCODER

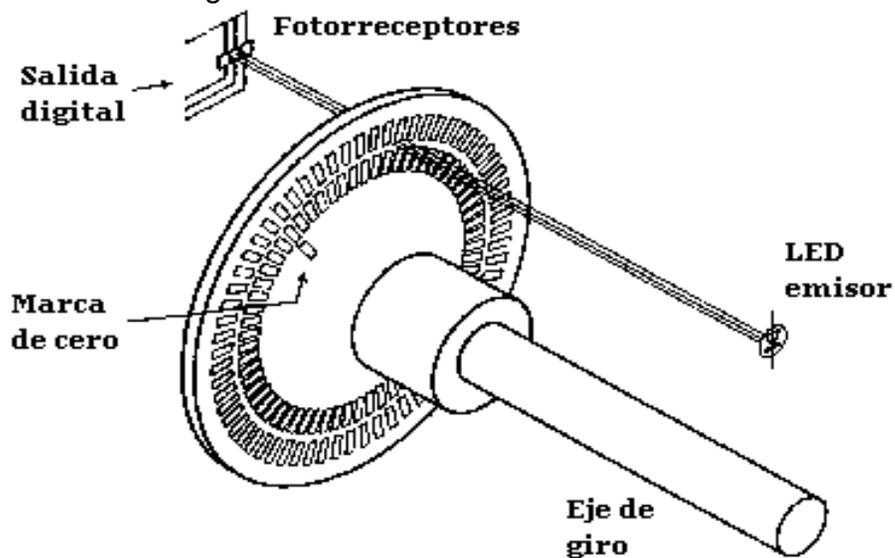
Un encoder es un codificador rotatorio, también llamado codificador del eje, suele ser un dispositivo electromecánico usado para convertir la posición angular de un eje a un código digital, lo que lo convierte en una clase de transductor. Estos dispositivos se utilizan en robótica, en lentes fotográficas de última generación, en dispositivos de entrada de ordenador (tales como el ratón y el TrackBall), y en plataformas de radar rotatorias. Hay dos tipos principales: absoluto e incremental [8].

Figura 4. Variedades de encoder



Fuente: CUENCA, ANDRÉS. Encoders, Tipos. Disponible en: <http://www.forosdeelectronica.com/f16/encoders-informacion-tecnica-25>

Figura 5. Funcionamiento del encoder



Fuente: CUENCA, ANDRÉS. Encoders, Tipos. Disponible en: <http://www.forosdeelectronica.com/f16/encoders-informacion-tecnica-25>

2. METODOLOGIA

Se realizaron las siguientes actividades con el fin de realizar el Proyecto:

Se estudió la Toolbox de Adquisición de Datos de Matlab por medio de algunos proyectos realizados en la Universidad Pontificia Bolivariana seccional Bucaramanga, tales como [1] y [2], y se complementó el estudio utilizando la información original de la Toolbox de Adquisición de Datos Matlab.

Se complementó el hardware de control, se diseñó y se construyó los acondicionadores de potencia que reciben la señal de las tarjetas de adquisición y hacen el control de los motores.

Por último se evaluó el comportamiento del control de los motores midiendo su tiempo de respuesta para llegar al set-point deseado.

Para cumplir lo anterior, se desarrolló las siguientes actividades:

2.1. INVESTIGACIÓN Y DOCUMENTACIÓN

Se recopiló y analizó la información sobre control de velocidad y posición en los motores; los cuales se implementaron en el sistema final.

Se diseñó los controladores de los motores.

Se adecuó el hardware necesario para realizar el control de los motores DC.

Obtención y compra de materiales

Retomando los resultados obtenidos por los diseños y bajo criterios de la investigación preliminar, se compró los componentes teniendo en cuenta el mercado disponible en Colombia.

2.2. RECONOCIMIENTO DE LOS INSTRUMENTOS

Se familiarizó con los elementos que se adquirieron para el módulo de adquisición de datos. Se estudió las características respectivas de cada uno de los componentes comprados.

2.3. AMPLIFICACIÓN DE SEÑALES

Se realizó los circuitos. A partir de una señal de 0 a 5 Voltios obtenida de una de las salidas de la tarjeta de adquisición de datos, se convirtió en una señal de -12 Voltios a 12 Voltios con la potencia necesaria para controlar el motor DC.

2.4. ACONDICIONAMIENTO DE SEÑALES

Se realizó los circuitos acondicionadores de señal para cada una de las tarjetas. Se tuvo en cuenta cada una de sus características y hojas de datos del fabricante.

2.5. MUESTREO Y TRANSMISIÓN DE DATOS

Estas señales cuantizadas se enviaron a través de una transmisión serial con la cual se ingresó los respectivos datos al PC.

2.6. PROGRAMACIÓN EN MATLAB

Se programó el software, orientado a la adquisición de datos transmitidos de forma serial. Se analizó el tratamiento de datos correspondientes a las diferentes características inherentes de las señales.

2.7. MANUFACTURA DEL CIRCUITO IMPRESO

Se diseñó el circuito impreso. Se tuvo en cuenta las respectivas necesidades y opciones de conexión.

2.8. DESARROLLO DE PRUEBAS

Se evaluó y analizó las respuestas del sistema a diferentes tipos de señales de estimulación.

2.9. EVALUACIÓN DE TIEMPOS DE RESPUESTA

Se evaluó los tiempos de respuesta utilizando varios set-points y las características generales del sistema de control.

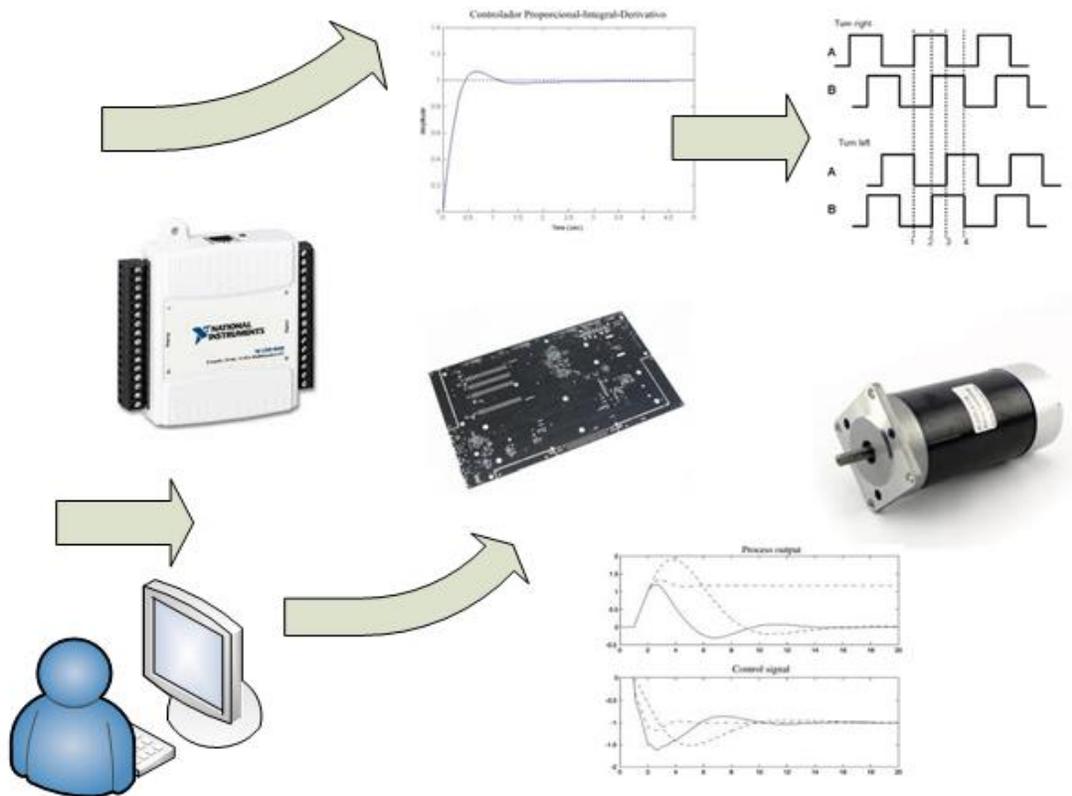
2.10. REDACCIÓN DEL INFORME FINAL

Se redactó el texto con los temas base para el estudio de la ciencia.

3. RESULTADOS Y DISCUSIÓN

Este proyecto consta de los siguientes módulos: un PC, donde tenemos la parte de software conformada por Matlab, de donde se utilizó Simulink y la toolbox de adquisición de datos por medio de la cual se realizó el control de los motores; el siguiente modulo consta de la tarjeta de adquisición de datos NI-USB 6008/6009 que se utilizó como nuestro puente entre software y las señales físicas a adquirir y como último los motores DC con sus respectivos drivers, como se pudo observar en la figura 6:

Figura 6. Diseño esquemático general implementado en el proyecto



Donde las tarjetas de adquisición de datos NI-USB 6008/6009 leen las señales eléctricas provenientes de los motores, y posteriormente se envían al PC para ser analizadas por medio de Simulink y así poder nuevamente enviar respuesta a las tarjetas de adquisición de datos NI-USB 6008/6009 para que éstas envíen señales a los motores para su control.

Luego de estudiar las características primordiales de la Tarjeta de adquisición de datos NI-USB 6008/6009 y tomando en cuenta el tipo de motor DC a emplear el diseño implementado consta de:

3.1. Lectura de Posición (Entrada):

La señal digital de posición suministrada por el encoder que se encuentra montado sobre el eje del motor, se convirtió a una señal análoga en un rango comprendido entre -10 y +10 VDC. De esta forma la polaridad positiva o negativa de la señal indicará si se encuentra adelante o atrás del cero de referencia del sistema de posición.

La señal análoga de posición se ingresó a la Tarjeta de adquisición de datos NI-USB 6008/6009 y a través de comunicación USB entrega el valor de posición digital al software de control (simulink).

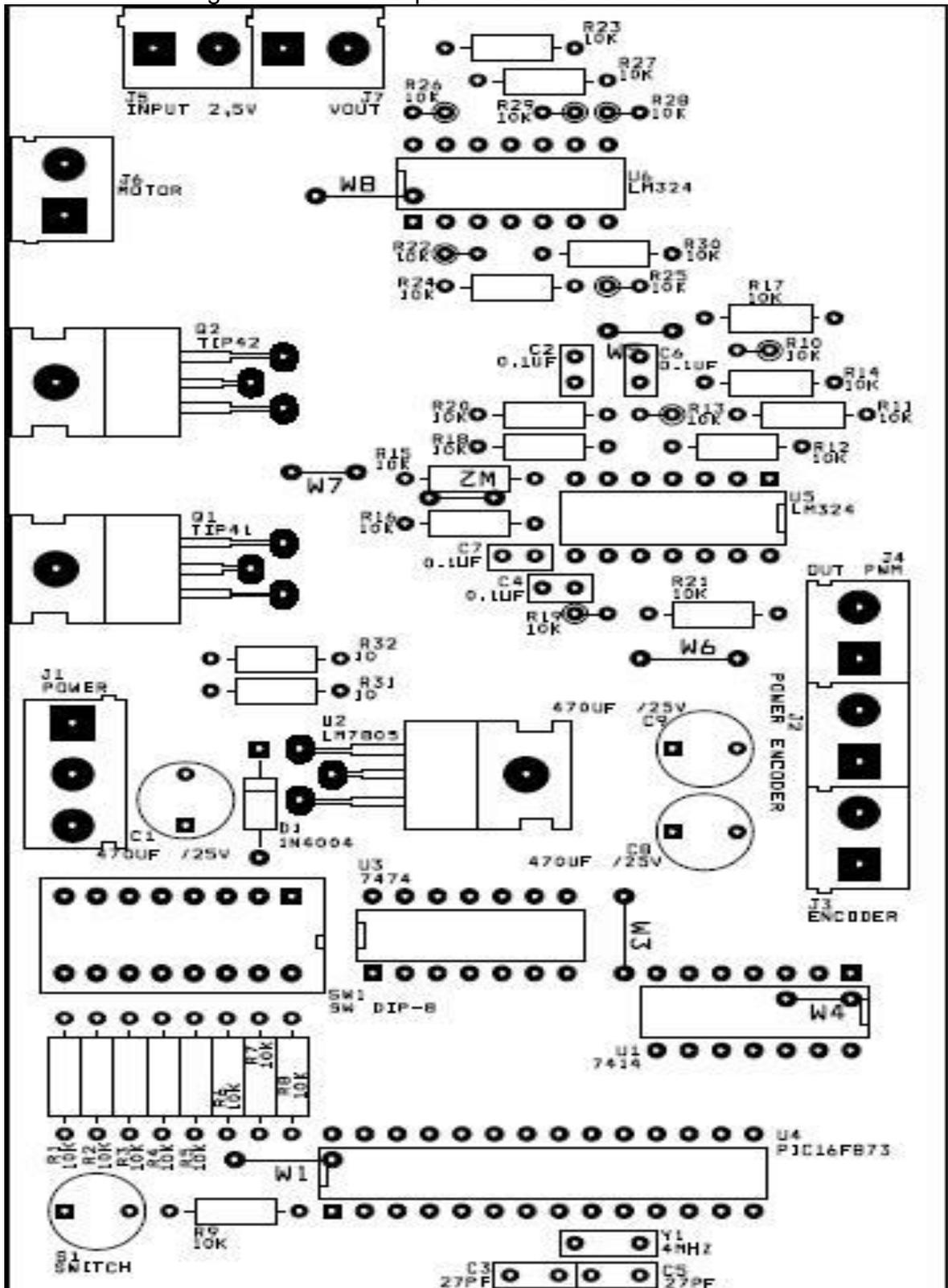
3.2. Control de posición del motor DC (Salida):

El software de control (simulink) suministra el valor de tensión que debe aplicarse al motor DC, este valor se encuentra en el rango de salida 0-5 V y el valor de cero corresponderá a 2.5 V. Este valor es transmitido a través del puerto USB hacia la Tarjeta de adquisición de datos NI-USB 6008/6009.

La Tarjeta de adquisición de datos NI-USB 6008/6009 entrega una señal análoga 0-5 V correspondiente a la tensión que debe aplicarse al motor. Adicionalmente la Tarjeta de adquisición de datos NI-USB 6008/6009 tiene una salida de tensión de referencia de 2.5 V que serán utilizados como referencia cero.

La tensión entregada por la Tarjeta de adquisición de datos NI-USB 6008/6009 se debe escalar de forma que 2.5 V sean transformados en cero, 0 V en -12 VDC y 5 V en +12 VDC. Esto se logra a través de un circuito construido a partir de amplificadores operacionales.

Figura 7. Circuito impreso del driver de los motores



¿Por qué no se emplea la entrada de pulsos de la NI-6008 para la lectura de los pulsos del encoder del motor?

Las características de la entrada de pulsos son:

Event Counter

Number of counters	1
Resolution	32 bits
Counter measurements.....	Edge counting (falling-edge)
Counter direction	Count up
Pull-up resistor.....	4.7 kohms to 5 V
Maximum input frequency.....	5 MHz
Minimum high pulse width.....	100 ns
Minimum low pulse width.....	100 ns
Input high voltage	2.0 V
Input low voltage.....	0.8 V

Se resaltó en rojo las características en contra de su implementación: la Tarjeta de adquisición de datos NI-USB 6008/6009 solamente cuenta pulsos en forma ascendente y aunque la frecuencia de conteo es de 5MHz la respuesta a un cambio de giro sería demasiado lenta al realizar el control, por cuanto dependería de la velocidad que el software de control (Simulink) pueda detectar los cambios de giro. En este caso la detección de la posición se realiza a través de un circuito dedicado a esa labor. Se empleó un micro controlador PIC16F873A, el cual recibe dos señales de un circuito flip-flop que corresponden a los pulsos y el sentido de giro. Esta información será usada para incrementar o decrementar un registro de posición que puede ser escalado externamente a través de un grupo de microswitch's de forma que su rango pueda ser extendido para mejorar la resolución en la medida de la posición del sistema a controlar.

Una vez obtenida la posición se debe convertir en una señal análoga, para esto se emplearon 2 señales de PWM: una para cada signo de la posición. Las señales PWM se restan y se filtran para obtener el valor promedio de voltaje en forma análoga entre -10 y 10 VDC, cómo se puede observar en el diseño circuitual de la figura 9.

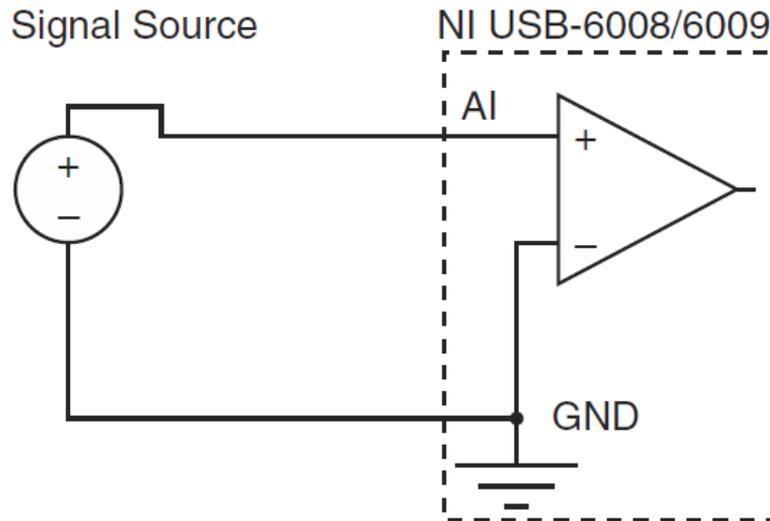
La señal análoga de posición se lee en la NI-6008 en modo "single", a través del canal 0, cómo podemos observar a continuación:

Analog Input

Analog inputs Differential	4
Single-ended	8, software-selectable

Input resolution	
NI USB-6008	
Differential	12 bits
Single-ended	11 bits
Max sample rate (aggregate) ¹	
NI USB-6008	10 kS/s
Converter type.....	Successive approximation
AI FIFO.....	512 bytes
Timing resolution.....	41.67 ns (24 MHz timebase)
Timing accuracy.....	100 ppm of actual sample rate
Input range	
Differential	± 20 V, ± 10 V, ± 5 V, ± 4 V, ± 2.5 V, ± 2 V, ± 1.25 V, ± 1 V
Single-ended.....	± 10 V

Figura 8. Entrada de señal analógica de la tarjeta NI-USB 6008/6009



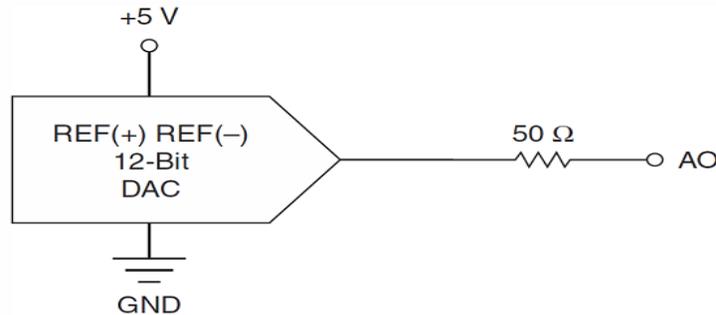
Fuente: User guide and specifications USB-6008/6009. Disponible en:
http://www.tau.ac.il/~electro/pdf_files/computer/ni_6008_ADC_manual.pdf

Se tomó en cuenta que la velocidad de adquisición no corresponde a la formulada en la tabla de características ya que está limitada enormemente por la velocidad del software de control (Simulink) como se demostrará en las pruebas realizadas.

3.3. Control del Motor DC

El motor DC requiere de una tensión y una corriente adecuadas para funcionar en el rango -12 y 12VDC @1A. La Tarjeta de adquisición de datos NI-USB 6008/6009 tiene 2 salidas análogas. En este diseño se empleó la salida 0.

Figura 9. Salida A0 de la tarjeta NI-USB 6008/6009



Las características de las salidas de la Tarjeta de adquisición de datos NI-USB 6008/6009 son:

Analog Output

Analog outputs	2
Output resolution	12 bits
Maximum update rate	150 Hz, software-timed
Output range	0 to +5 V
Output impedance	50 ohms
Output current drive	5 mA
Power-on state.....	0 V

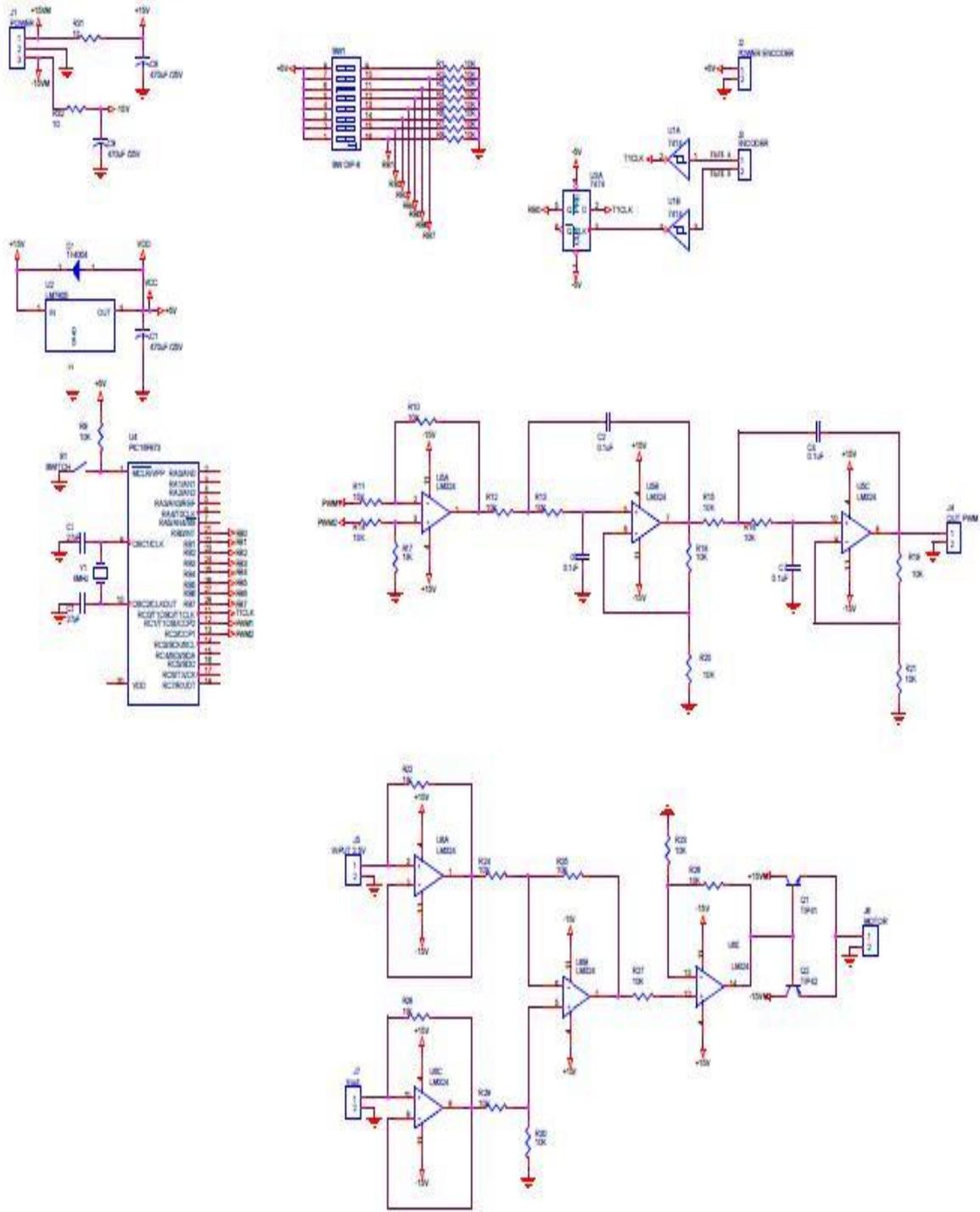
Como se puede observar las salidas no poseen la corriente ni el rango de voltaje necesario para poder manejar directamente el motor. Por lo tanto la señal se amplifica de 0-5VDC @5mA hasta convertirla en -12 a 12VDC @1A de forma que puede atacar directamente al bobinado de armadura del motor.

3.4. External Reference (+2.5 V).

“The NI USB-6008/6009 creates a high-purity reference voltage supply for the ADC using a multi-state regulator, amplifier, and filter circuit. You can use the resulting +2.5 V reference voltage as a signal for self-test” [6].

La salida de voltaje de referencia de la Tarjeta de adquisición de datos NI-USB 6008/6009 se emplea como valor cero para la salida análoga.

Figura 10. Diseño circuitual del control y sus componentes

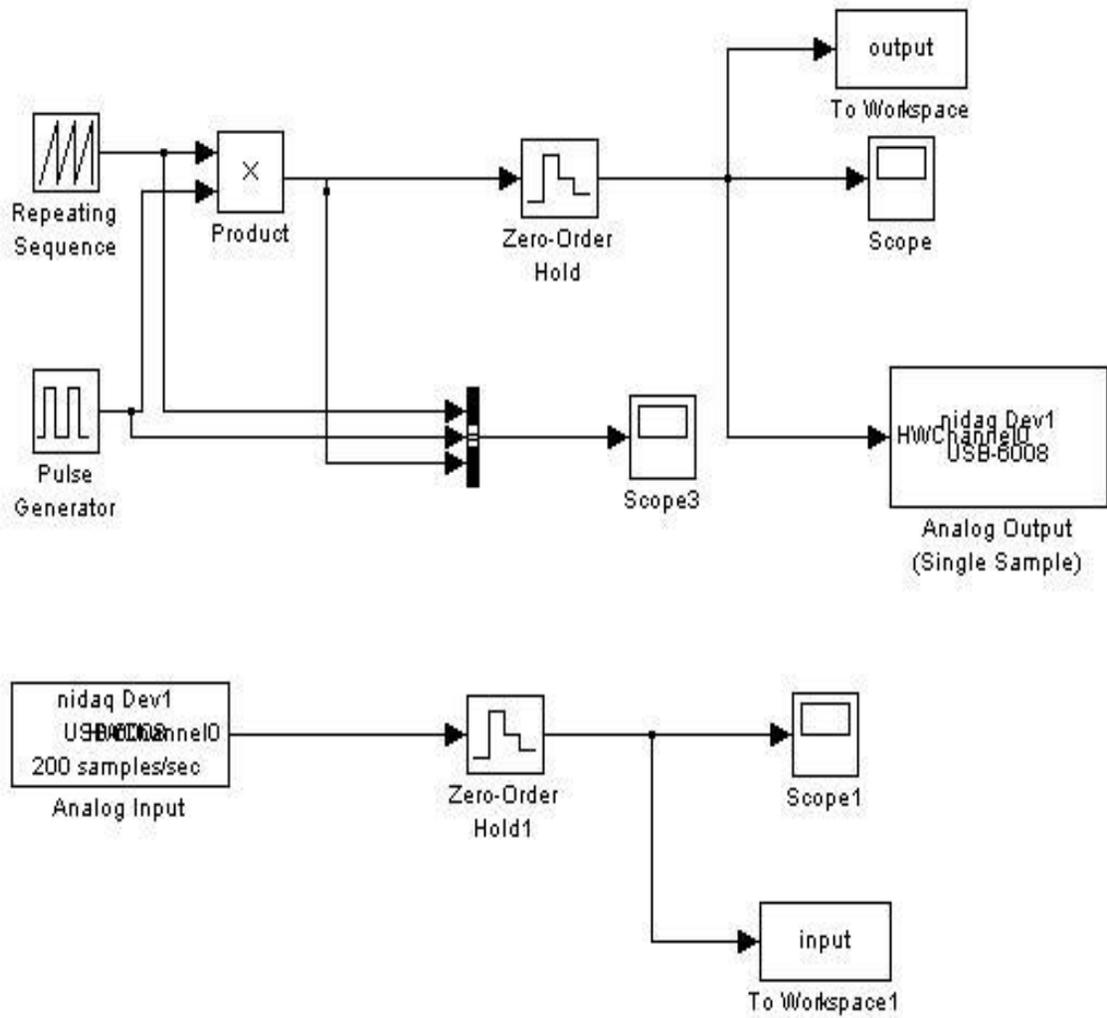


3.5. Diseños en simulink para medir tiempos de respuesta con la tarjeta de adquisición de datos NI-USB 6008/6009

3.5.1. Modificación de los parámetros de configuración de simulación en Simulink

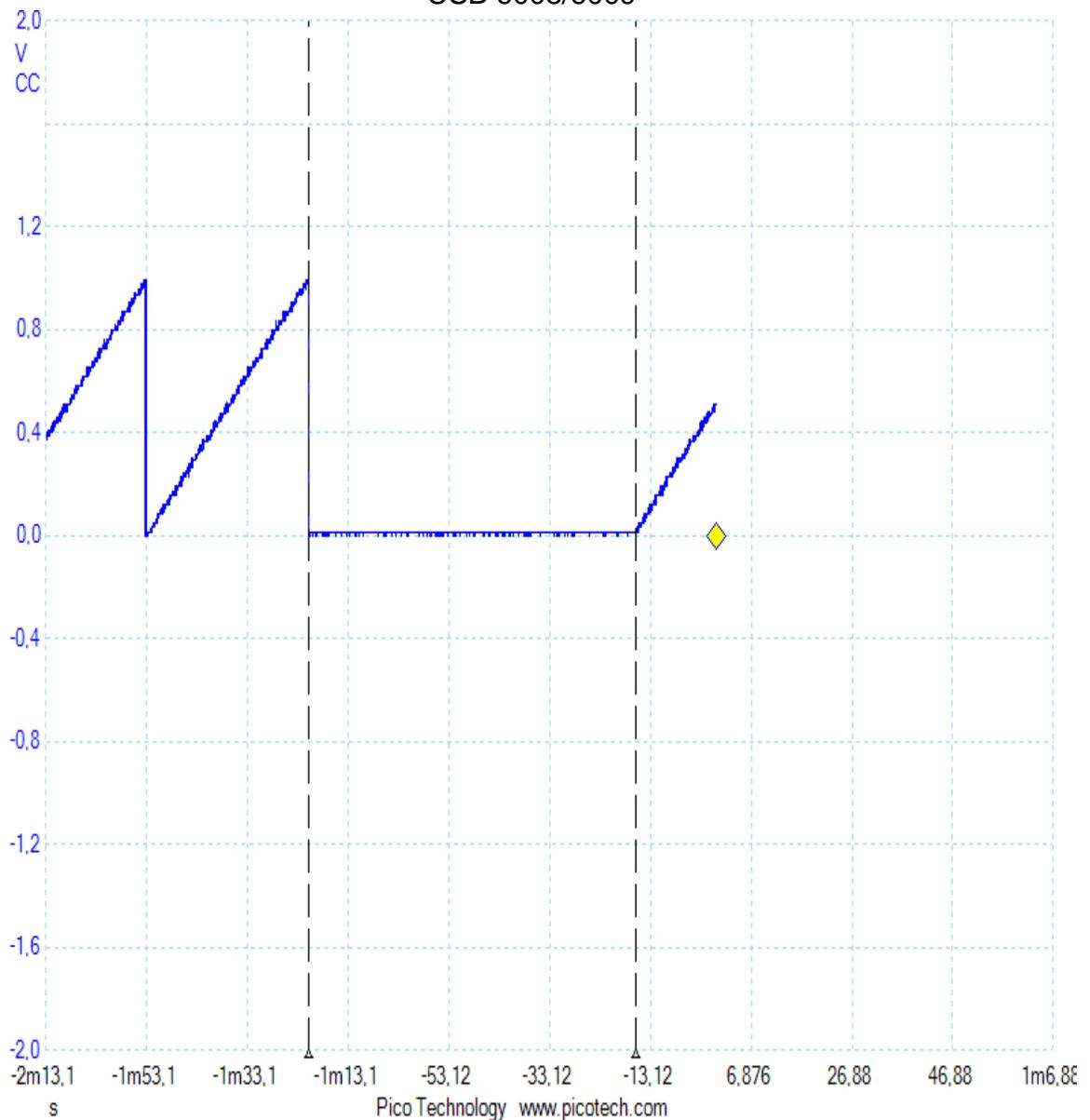
Inicialmente se diseñó un sistema de control en Simulink (Matlab), donde se utilizó la tarjeta de adquisición de datos NI-USB 6008/6009. Ésta leyó una señal cualquiera por una de sus salidas análogas (ao0), y después entregó la señal por una de sus entradas análogas (ai0); así se evaluó la señal, como se pudo observar en la figura 11:

Figura 11. Diseño de nuestro sistema de control



Se utilizó una señal diente de sierra con un tiempo de muestreo de 0.5 seg, y se multiplicó con una señal de pulsos para que cuando la señal de pulsos se encuentre en 1 V, muestre la señal diente de sierra, y cuando la señal de pulsos se encuentre en 0 V, muestre una señal en 0 V resultado de multiplicar la señal diente de sierra por 0 V, y así poder observar detalladamente el periodo la señal diente de sierra de una manera más simple, como se pudo observar en la figura 12, donde se vio la señal diente de sierra después de pasar por la tarjeta de adquisición de datos NI-USB 6008/6009:

Figura 12. Señal diente de sierra medida en la entrada analógica ai0 de la tarjeta NI-USB 6008/6009



En la figura anterior se pudo observar que el periodo de la señal diente de sierra es de 65.2 seg después de hacer pasar la señal diente de sierra por la tarjeta de adquisición de datos NI-USB 6008/6009 con un tiempo de muestreo de 0.5 seg. El periodo de nuestra señal diente de sierra es muy lento para realizar un control, por lo tanto se realizó un ajuste en la configuración de los parámetros de Simulink, cómo se muestra en las siguientes figuras:

Figura 13. Modificación de los parámetros de configuración de la simulación en Simulink paso 1

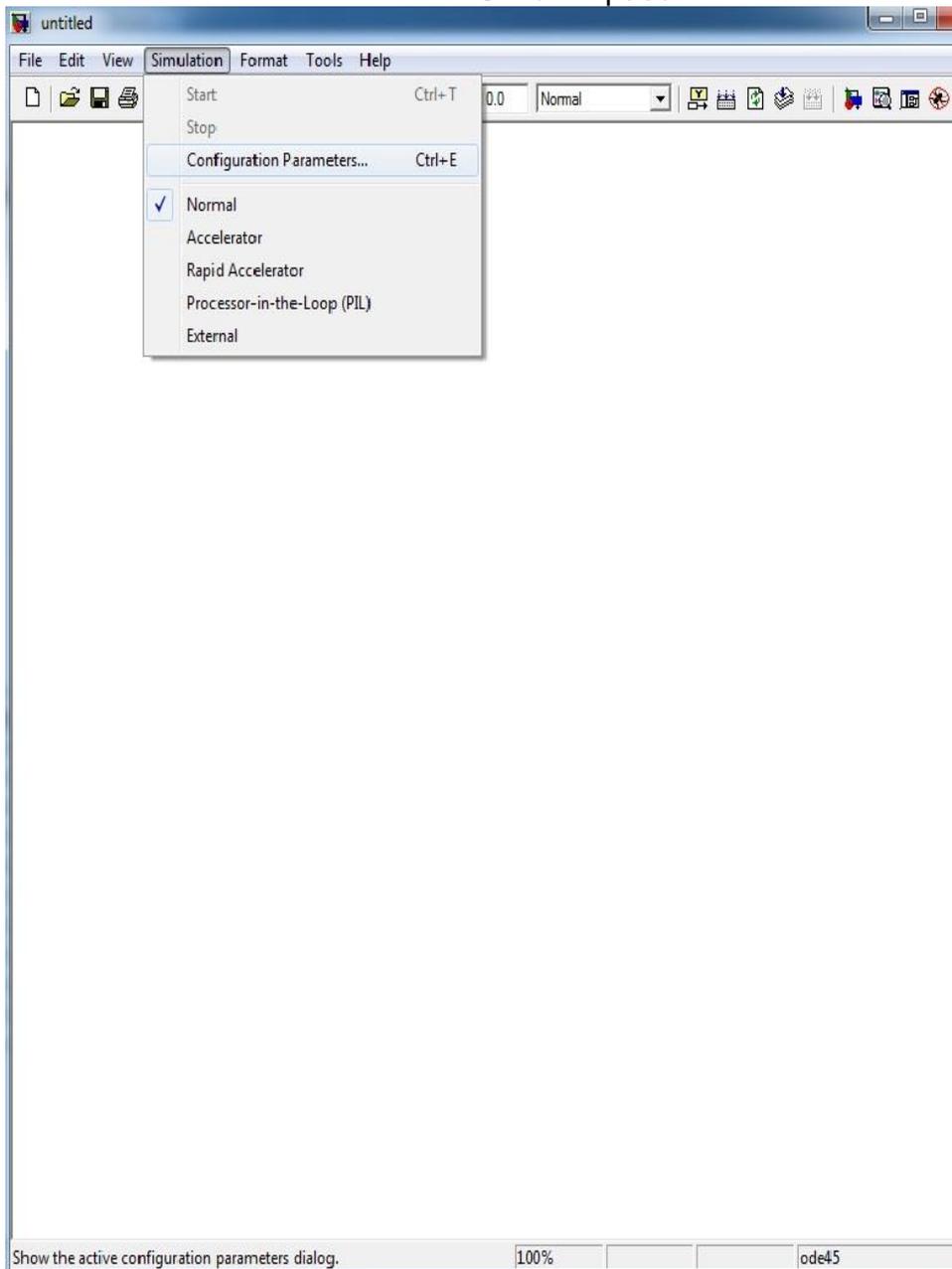


Figura 14. Modificación de los parámetros de configuración de la simulación en Simulink paso 2

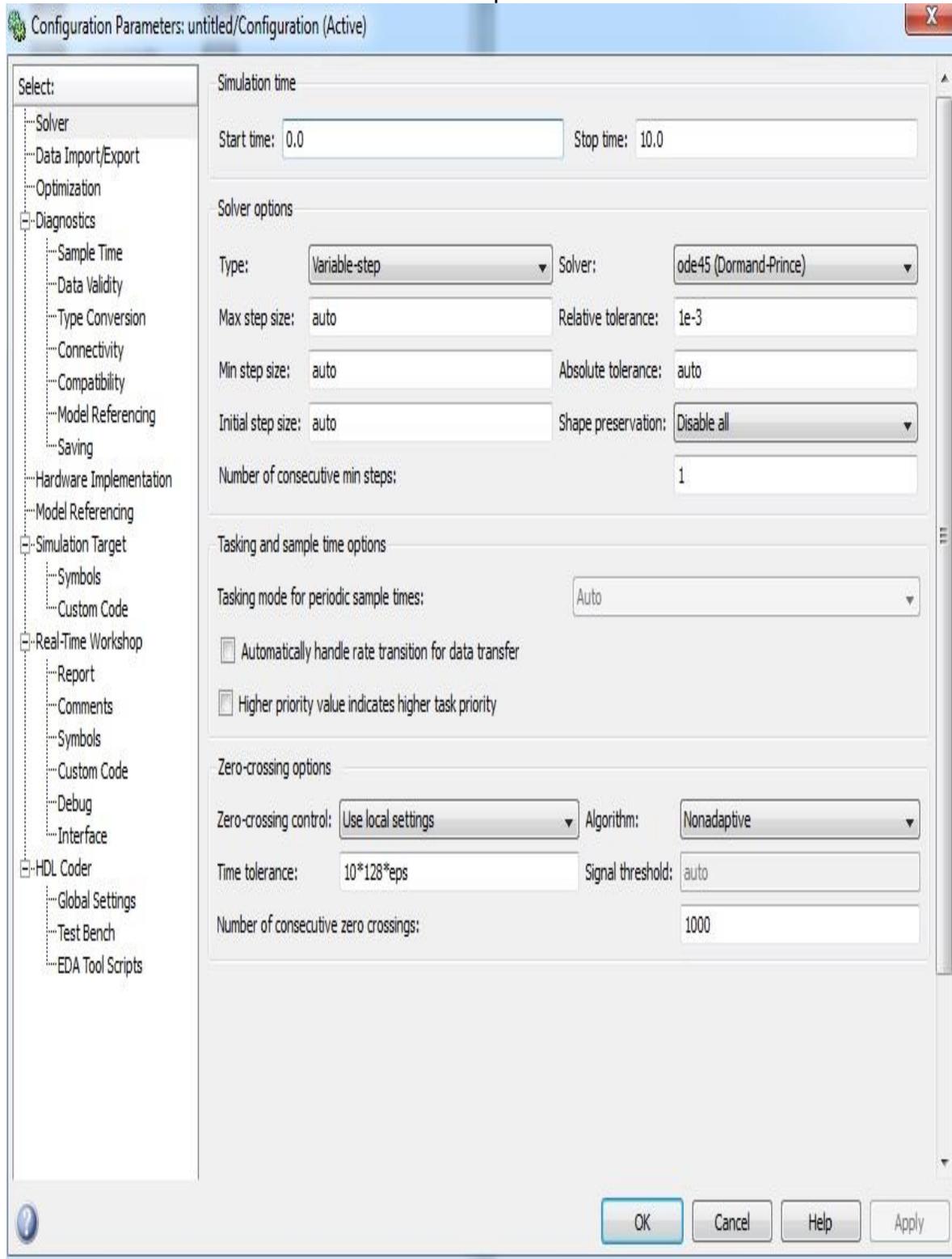
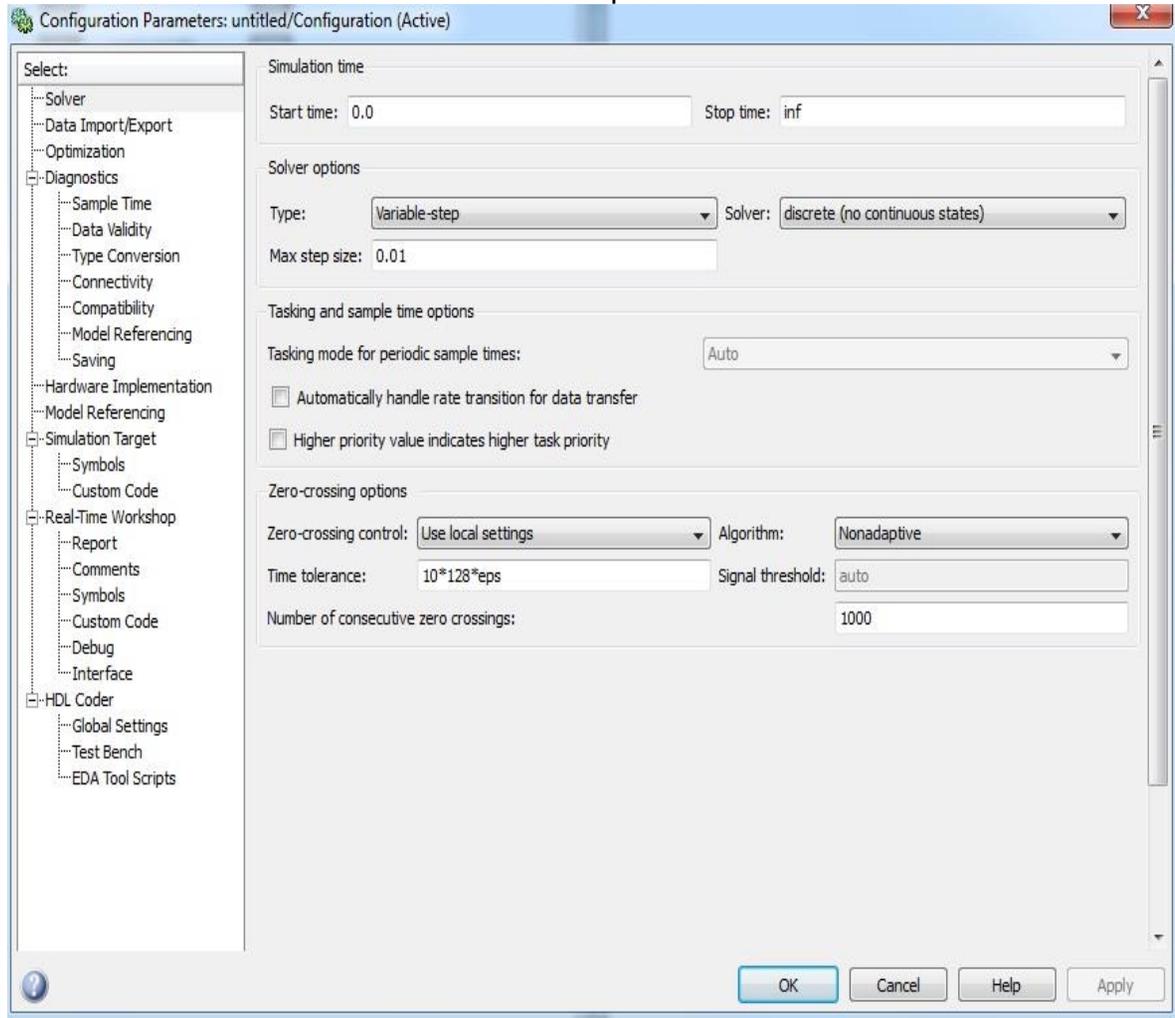


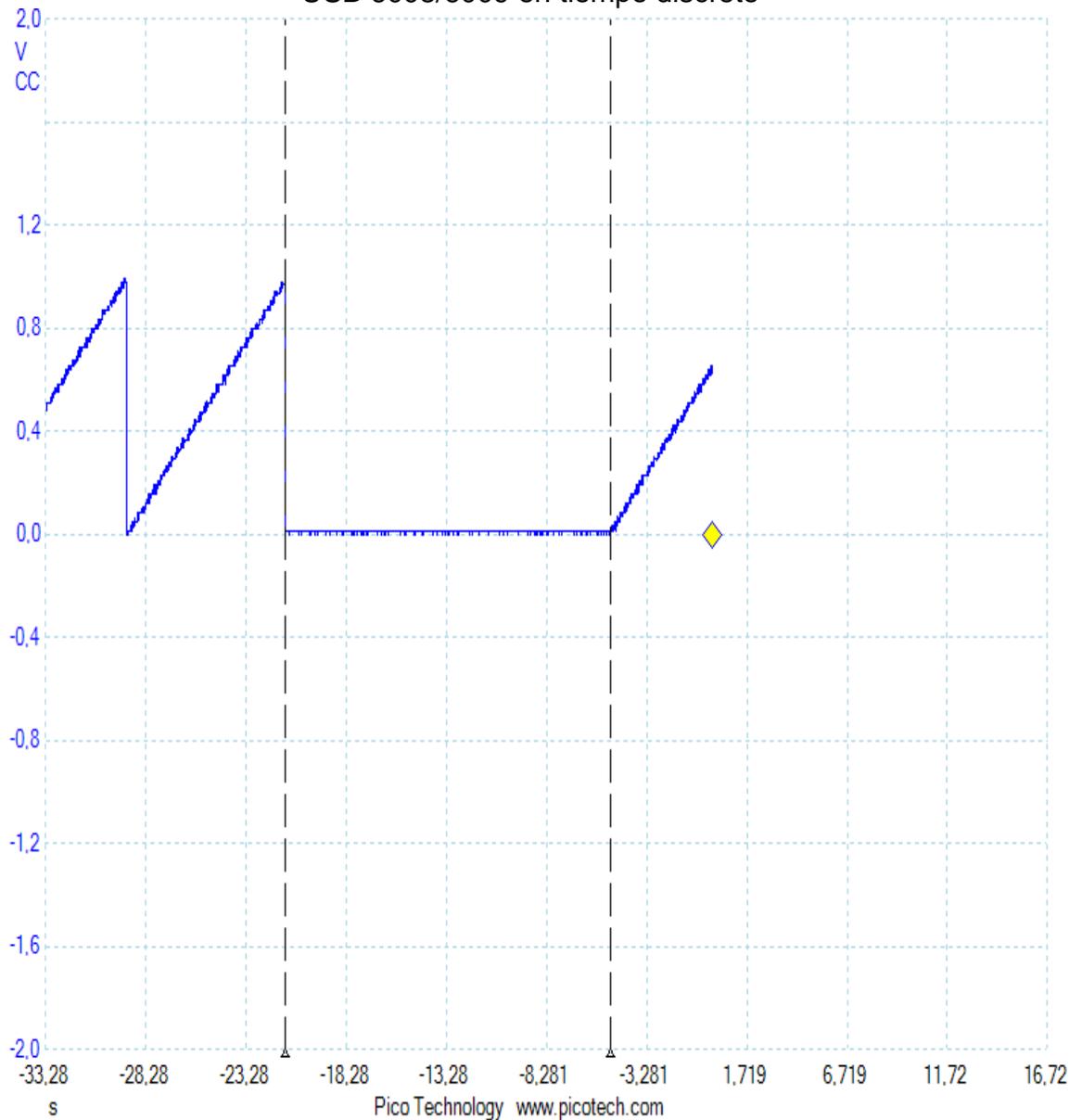
Figura 15. Modificación de los parámetros de configuración de la simulación en Simulink paso 3



En las tres figuras anteriores, se cambió el “Solver” en “Solver options”, de ode45 a estado discreto, ya que el sistema trabaja en estado discreto, y posteriormente se colocó el “Max step size” en 0.01, para así poder trabajar el sistema en estado discreto y con un tiempo de muestreo de 0.01 seg, al igual que los retenedores utilizados en el sistema, para así tener todo el sistema sincronizado en tiempo discreto.

Después de los cambios realizados anteriormente, se procedió a realizar la misma prueba realizada en la figura 10 y el resultado fue una disminución en el retardo de la señal tratada por la tarjeta de adquisición de datos NI-USB 6008/6009. Se observó que el periodo de la señal diente de sierra disminuyó a 16.23 segundos, como se pudo observar en la figura 16:

Figura 16. Señal diente de sierra medida en la entrada analógica ai0 de la tarjeta NI-USB 6008/6009 en tiempo discreto



Éste resultado muestra que modificando el “Solver” en “Solver options”, de ode45 a estado discreto en los parámetros de configuración de Simulink, se pudo disminuir el retardo de la señal de salida, como se pudo observar en la tabla 2, y así mismo se pudo hacer que el sistema no sea tan lento como lo era antes. Ya con este primer paso se pudo seguir con el objetivo principal de este proyecto que es el de controlar 2 motores con 2 tarjetas NI-USB 6008/6009.

Tabla 2. Comparación del periodo de una señal en modo continuo y discreto

	Tiempo continuo	Tiempo discreto
Periodo de la señal diente de sierra (segundos)	65.2	16.23

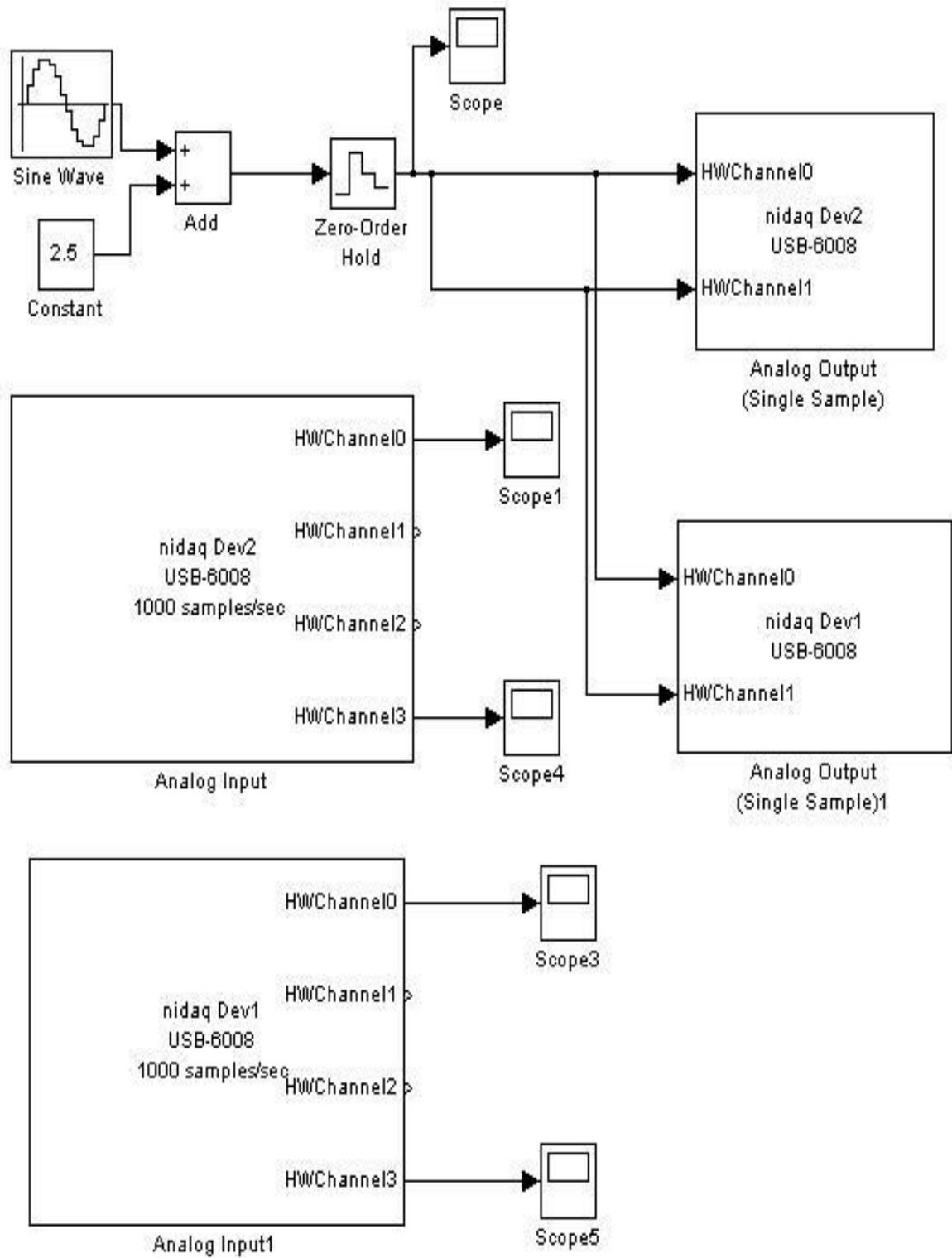
3.5.2. Pruebas con varias entradas y varias salidas análogas de la tarjeta de adquisición de datos NI-USB 6008/6009

Inicialmente, lo que se quiso probar y evaluar con este proyecto es el comportamiento de la velocidad de control de posición de varios motores dc utilizando varias tarjetas NI-USB 6008/6009, y así poder comparar y demostrar, que además de ser un sistema bastante lento para realizar control, si es posible hacerlo con las herramientas que ofrece la Universidad Pontificia Bolivariana en su laboratorio de control.

Se debe tener en cuenta que las tarjetas NI-USB 6008/6009 cuentan con 2 salidas análogas y 8 entradas análogas, así que se empezó primero realizando las pruebas con una salida análoga, después con 2 salidas análogas simultáneamente, después con una salida análoga y una entrada análoga simultáneamente, luego con una salida análoga y las 8 entradas análogas simultáneamente, para así llegar a trabajar con 2 tarjetas simultáneamente pero utilizándolas y haciendo que respondan con todas sus entradas y salidas análogas simultáneamente; así se pudo evaluar la velocidad de respuesta en cada uno de los casos mencionados anteriormente.

Se diseñó un sistema en Simulink donde se evaluó el periodo de respuesta de una onda senoidal con una frecuencia 10 rad/seg y con un tiempo de muestreo de 5 ms y se observó el tiempo de respuesta en cada uno de los casos mencionados anteriormente, como muestra la figura 17 a continuación:

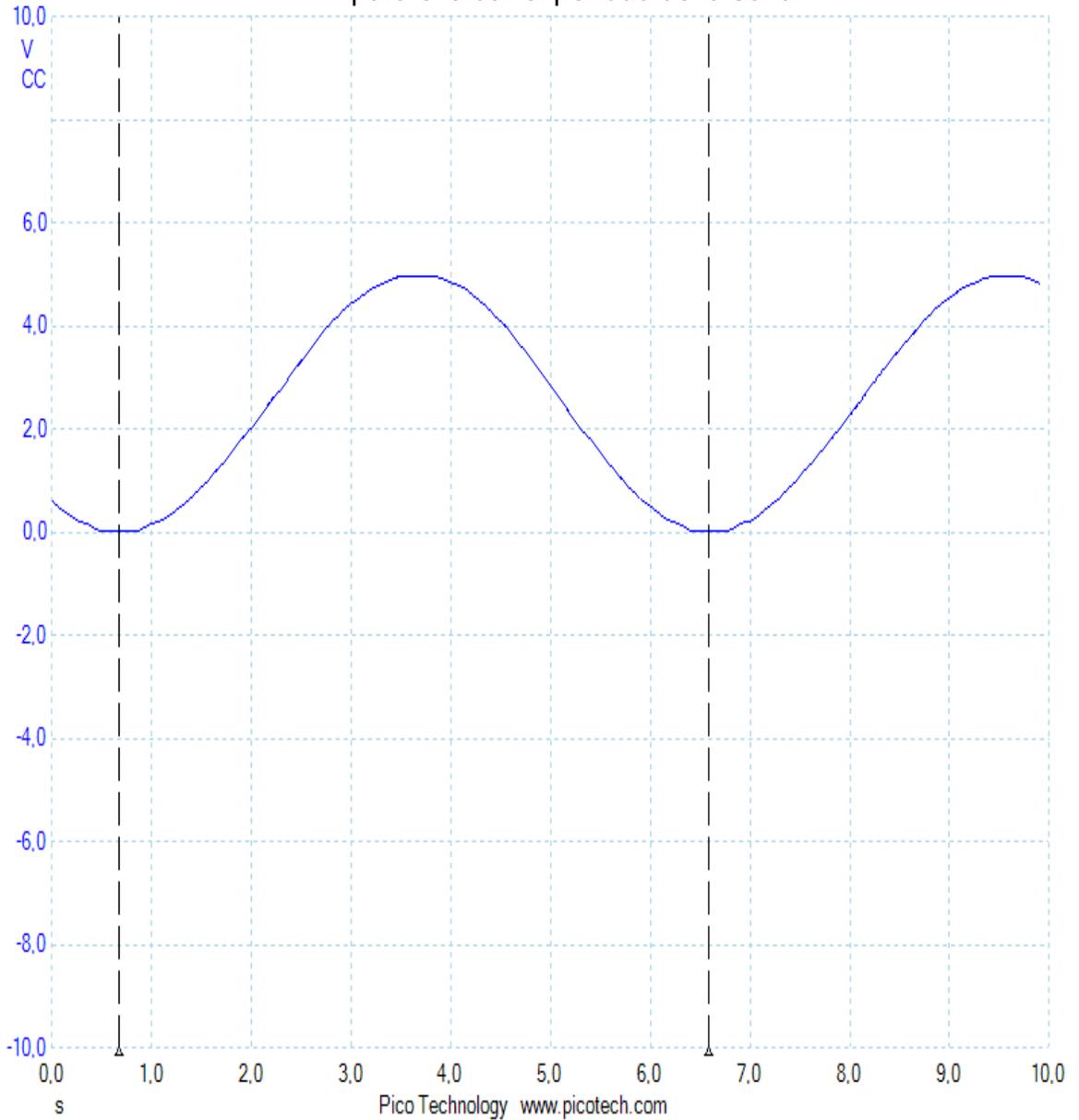
Figura 17. Diseño en Simulink. Utilización de varias entradas y varias salidas análogas de las tarjetas NI USB 6008/6009



3.5.2.1. Una salida análoga

En el primer caso, se evaluó el periodo de la señal utilizando una salida análoga de la tarjeta NI-USB 6008/6009, como se pudo observar en la figura 18:

Figura 18. Utilización de una salida análoga de la tarjeta NI-USB 6008/6009 para evaluar el periodo de la señal

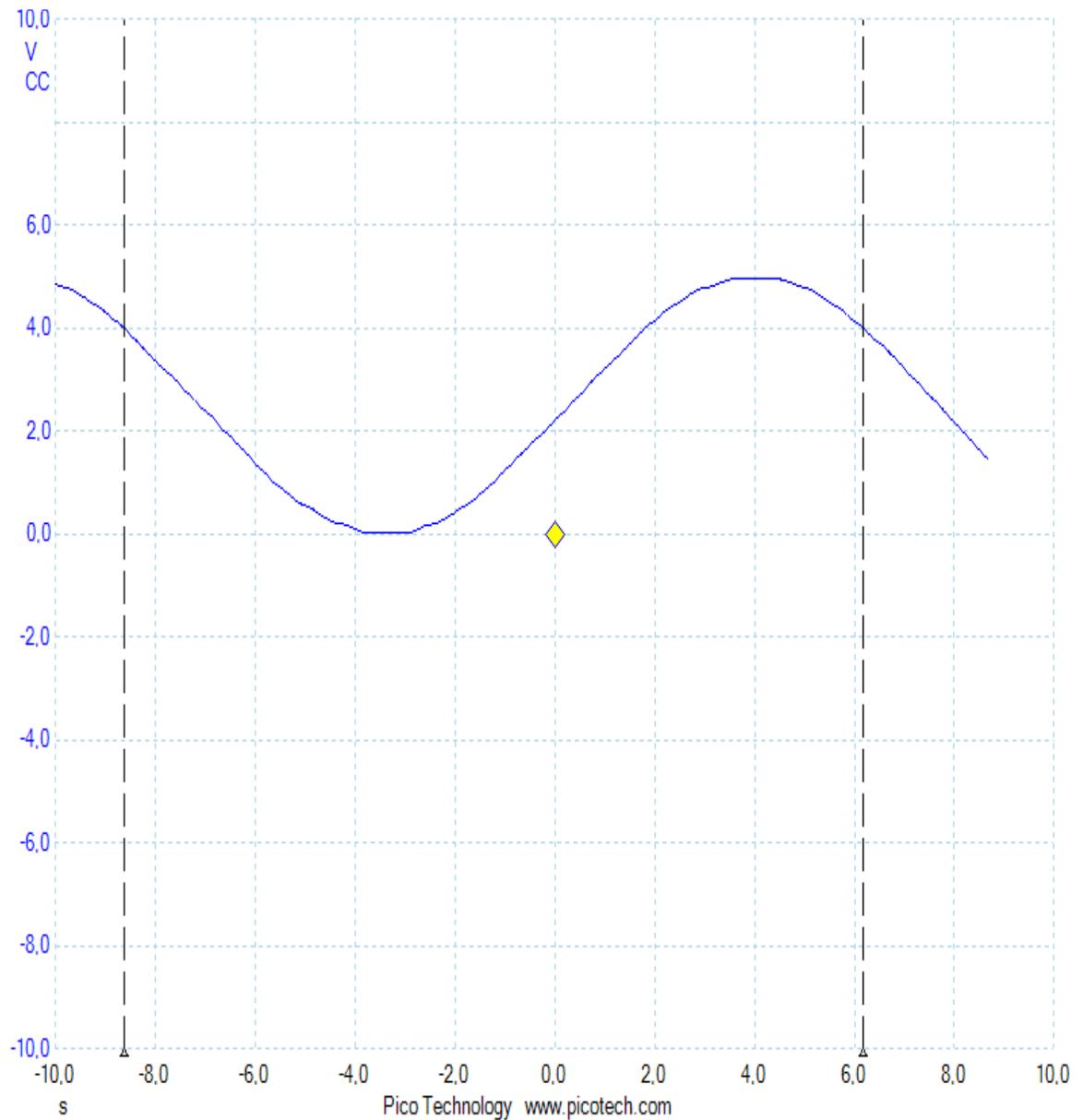


Periodo de la señal de salida: 5.916 segundos

3.5.2.2. Dos salidas análogas

En este caso, se evaluó el periodo de la señal utilizando dos salidas análogas de la tarjeta NI-USB 6008/6009, como se pudo observar en la figura 19:

Figura 19. Utilización de dos salidas análogas de la tarjeta NI-USB 6008/6009 para evaluar el periodo de la señal

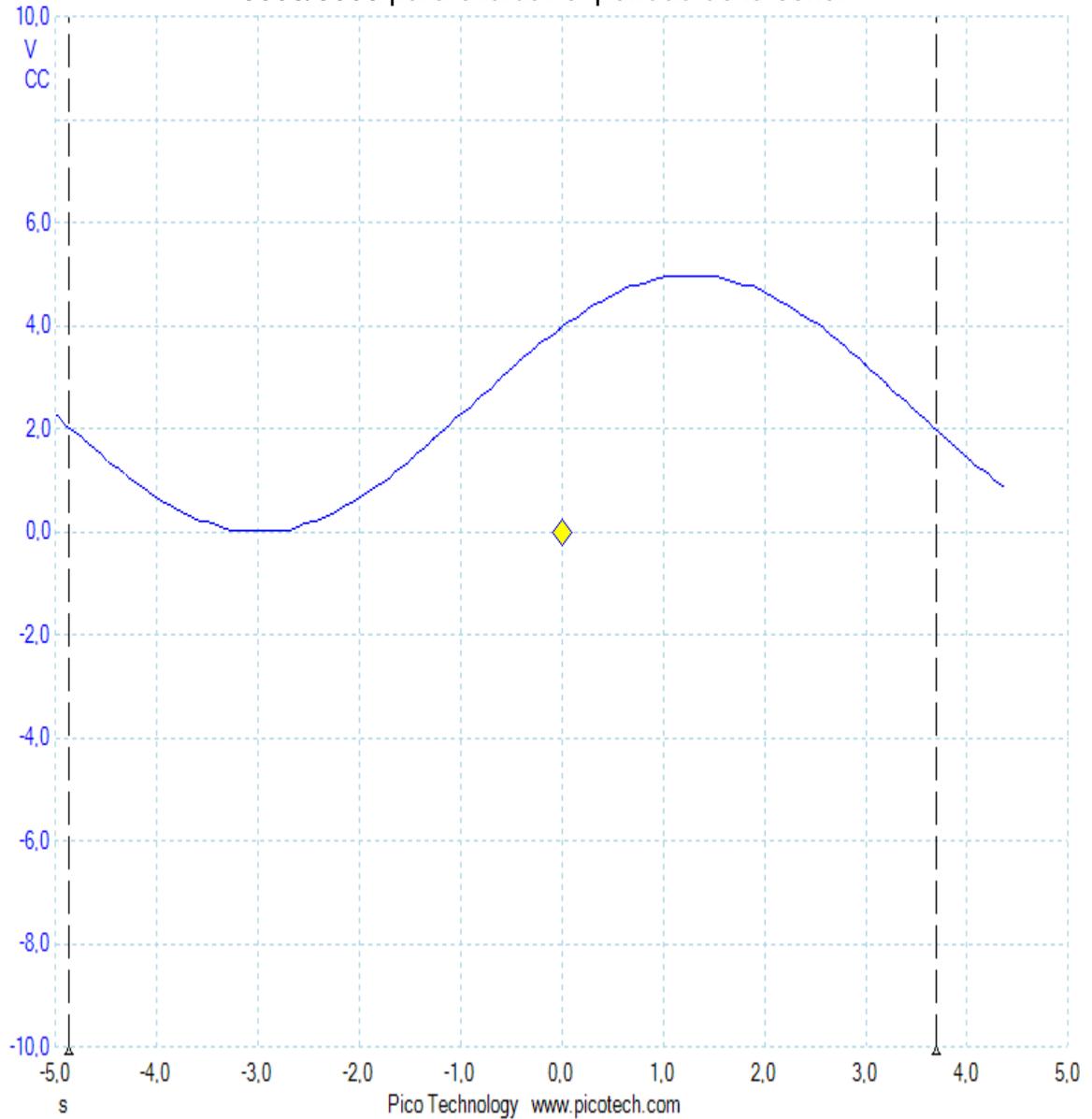


Periodo de la señal de salida: 14.82 segundos

3.5.2.3. Una salida y una entrada análogas

Se evaluó el periodo de la señal utilizando una salida y una entrada análogas de la tarjeta NI-USB 6008/6009, como se pudo observar en la figura 20:

Figura 20. Utilización de una salida y una entrada análogas de la tarjeta NI-USB 6008/6009 para evaluar el periodo de la señal

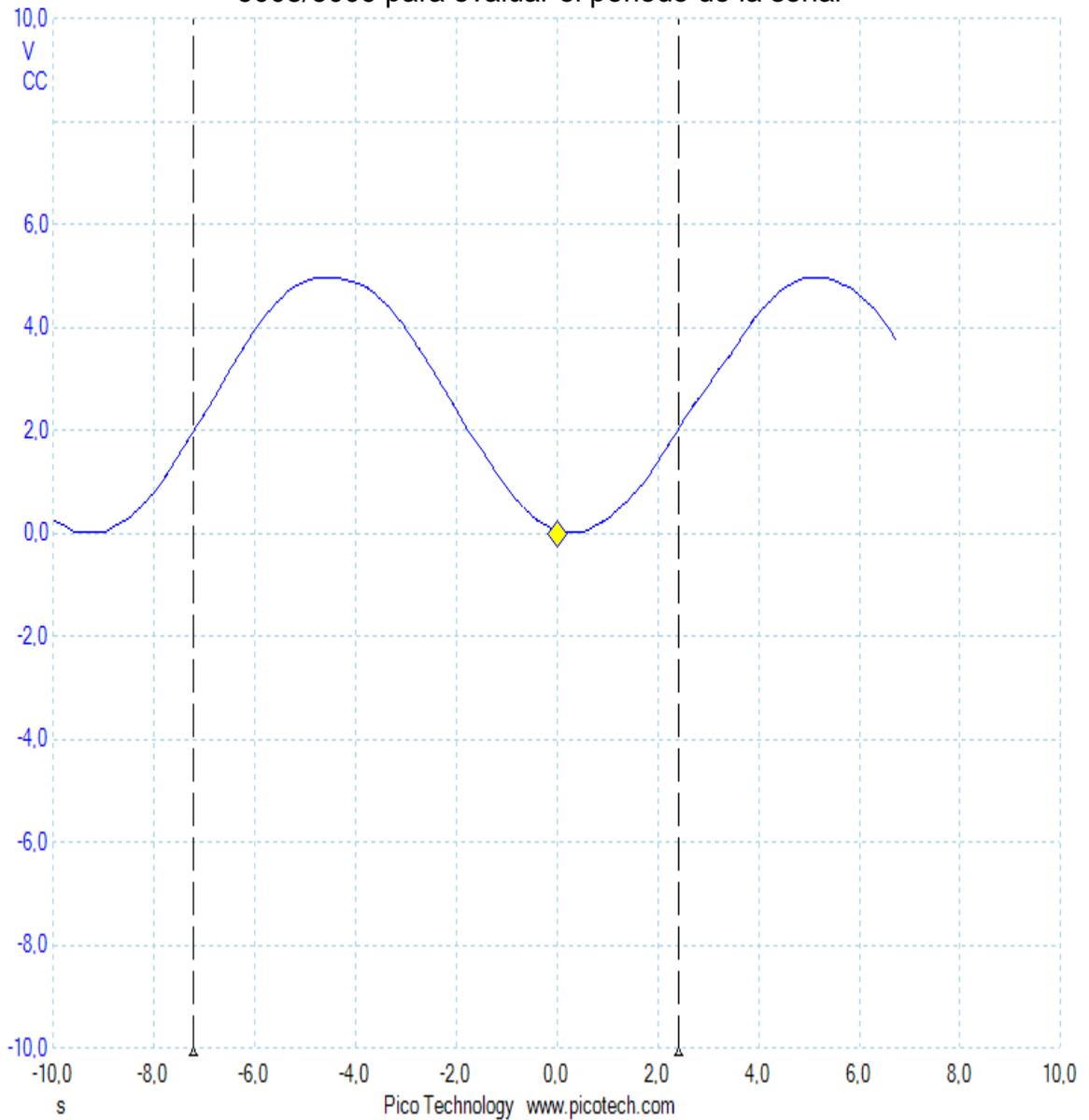


Periodo de la señal de salida: 8.556 segundos

3.5.2.4. Una salida y ocho entradas análogas

Se evaluó el periodo de la señal utilizando una salida y ocho entradas análogas de la tarjeta NI-USB 6008/6009, como se pudo observar en la figura 21:

Figura 21. Utilización de una salida y ocho entradas análogas de la tarjeta NI-USB 6008/6009 para evaluar el periodo de la señal

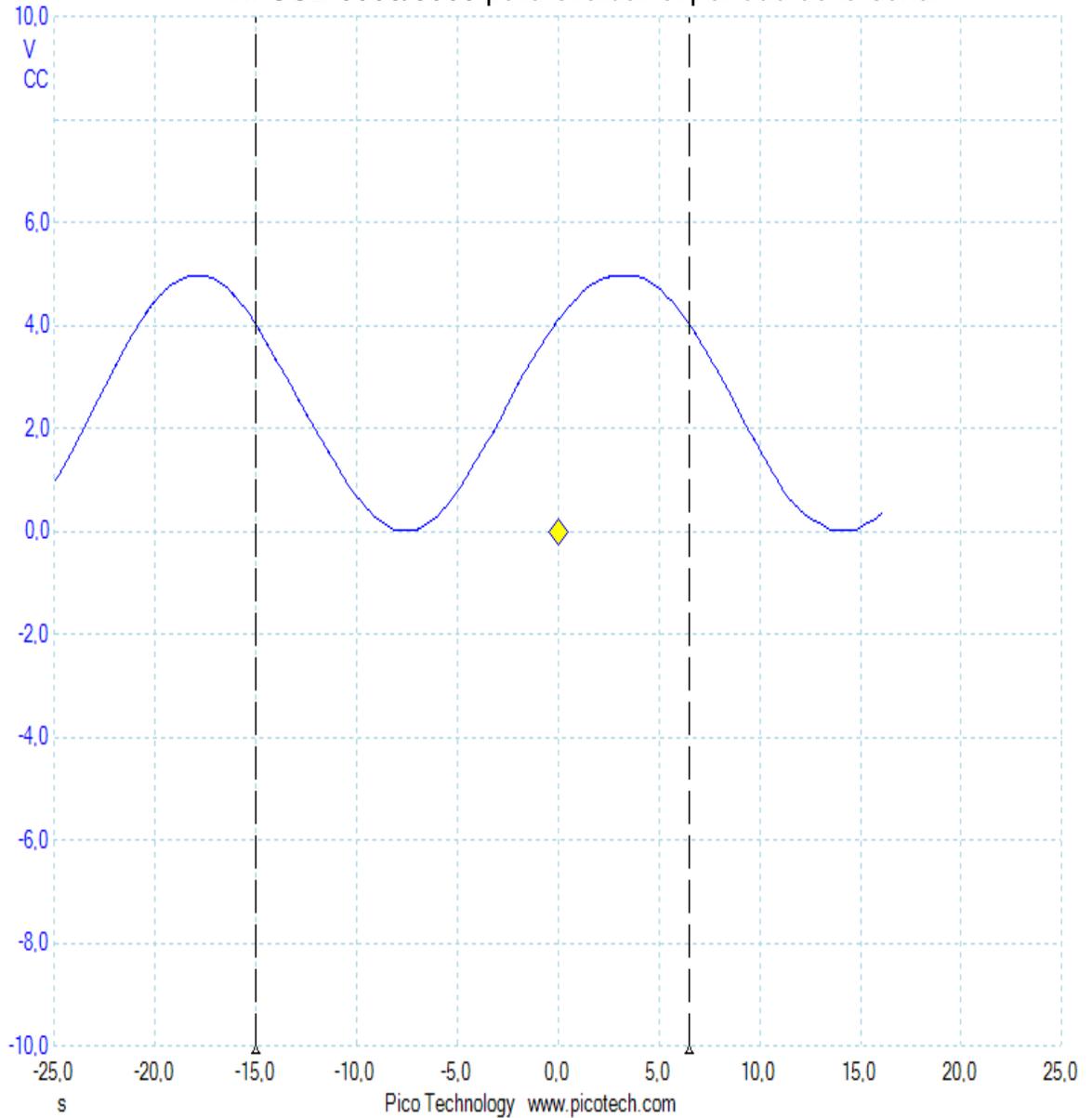


Periodo de la señal de salida: 9.64 segundos

3.5.2.5. Dos salidas y una entrada análogas

Se evaluó el periodo de la señal utilizando dos salidas y una entrada análogas de la tarjeta NI-USB 6008/6009, como se pudo observar en la figura 22:

Figura 22. Utilización de dos salidas y una entrada análogas de la tarjeta NI-USB 6008/6009 para evaluar el periodo de la señal

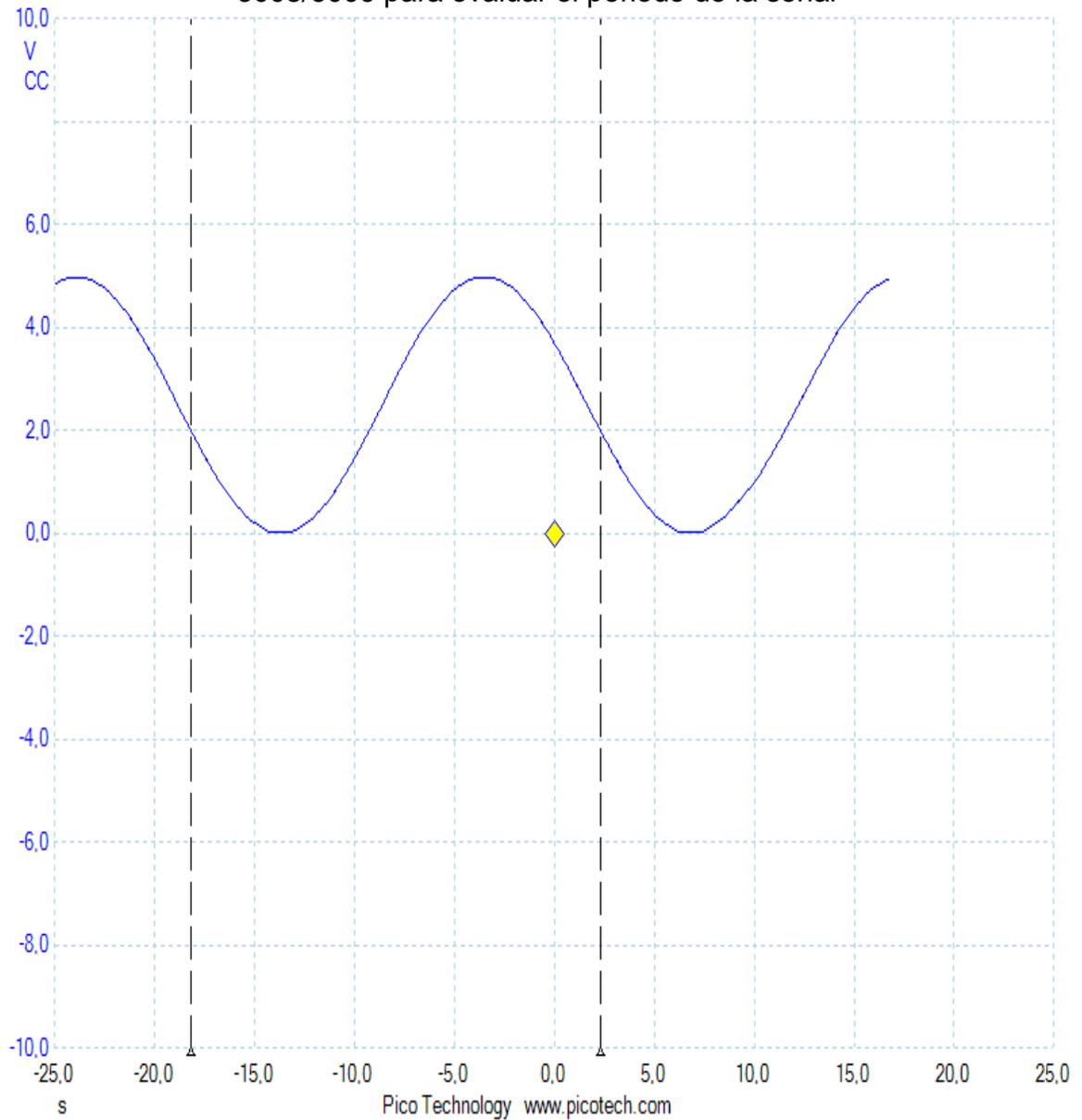


Periodo de la señal de salida: 20.48 segundos

3.5.2.6. Dos salidas y ocho entradas análogas

Se evaluó el periodo de la señal utilizando dos salidas y una entrada análogas de la tarjeta NI-USB 6008/6009, como se pudo observar en la figura 23:

Figura 23. Utilización de dos salidas y ocho entrada análogas de la tarjeta NI-USB 6008/6009 para evaluar el periodo de la señal



Periodo de la señal de salida: 21.56 segundos

Tabla 3. Evaluación del periodo de las señales de salida para varias entradas y para varias salidas de la tarjeta de adquisición de datos NI-USB 6008/6009

Una tarjeta	Una salida	Dos salidas	Una salida y una entrada	Una salida y ocho entradas	Dos salidas y una entrada	Dos salidas y ocho entradas
Periodo de la señal de salida (segundos)	5.916	14.82	8.556	9.64	20.48	21.56

Como se pudo observar en los gráficos y en los resultados de la evaluación del comportamiento del control de los motores DC en la tabla 3, a medida que se utilizó más entradas o más salidas análogas de la tarjeta NI-USB 6008/6009 simultáneamente, el sistema se comportó cada vez más lento. Se pudo notar que el rango de tiempos cuando se utilizó las salidas análogas de la tarjeta NI-USB 6008/6009 es mucho mayor que cuando se utilizó las entradas análogas de la tarjeta NI-USB 6008/6009. Esto quiere decir, que si se está utilizando solamente una salida análoga de la tarjeta NI-USB 6008/6009, y después se pasó a utilizar 2 salidas análogas, el sistema se comporta muchísimo más lento a comparación si estuviera utilizando una salida análoga y después pasara a utilizar una salida y una entrada, en ese caso el sistema se comporta un poco más lento, pero no tanto como si estuviera utilizando 2 salidas análogas.

Entre más entradas análogas utilice, el sistema se comporta más lento, pero entre más salidas utilice, el sistema se comporta mucho más lento; y se debe tener en cuenta que se va a trabajar con 2 tarjetas simultáneamente, es decir, el sistema se va a comportar aún más lento.

Como bien se ha venido mencionando a lo largo del desarrollo de éste proyecto, estamos utilizando la Toolbox de adquisición de datos de Matlab, que como su nombre lo indica, ella es bastante aceptable a la hora de adquirir datos, pero el problema se presenta a la hora de entregar datos, ahí es cuando el sistema no responde como se quiere.

En general, estos tiempos de respuesta no son para nada aceptables a la hora de realizar un control y mucho menos a la hora de realizar control en Robótica.

3.6. Evaluación del comportamiento del control de 2 motores DC con 2 tarjetas de adquisición de datos NI-USB 6008/6009

Teniendo en cuenta los resultados obtenidos en la tabla 2 del capítulo anterior, se diseñó varios sistemas en Simulink con el fin de evaluar la velocidad de respuesta del control de 2 motores por medio de 2 tarjetas de adquisición de datos NI-USB 6008/6009 donde se realizó varias pruebas para demostrar que a medida que se utilizó más entradas o más salidas análogas de las tarjetas, el sistema de control respondió cada vez más lento.

El sistema de control que se utilizó para este proyecto fue un sistema de control PID en Simulink con la ayuda de la toolbox de adquisición de datos de Matlab, para así lograr realizar el control de los motores DC.

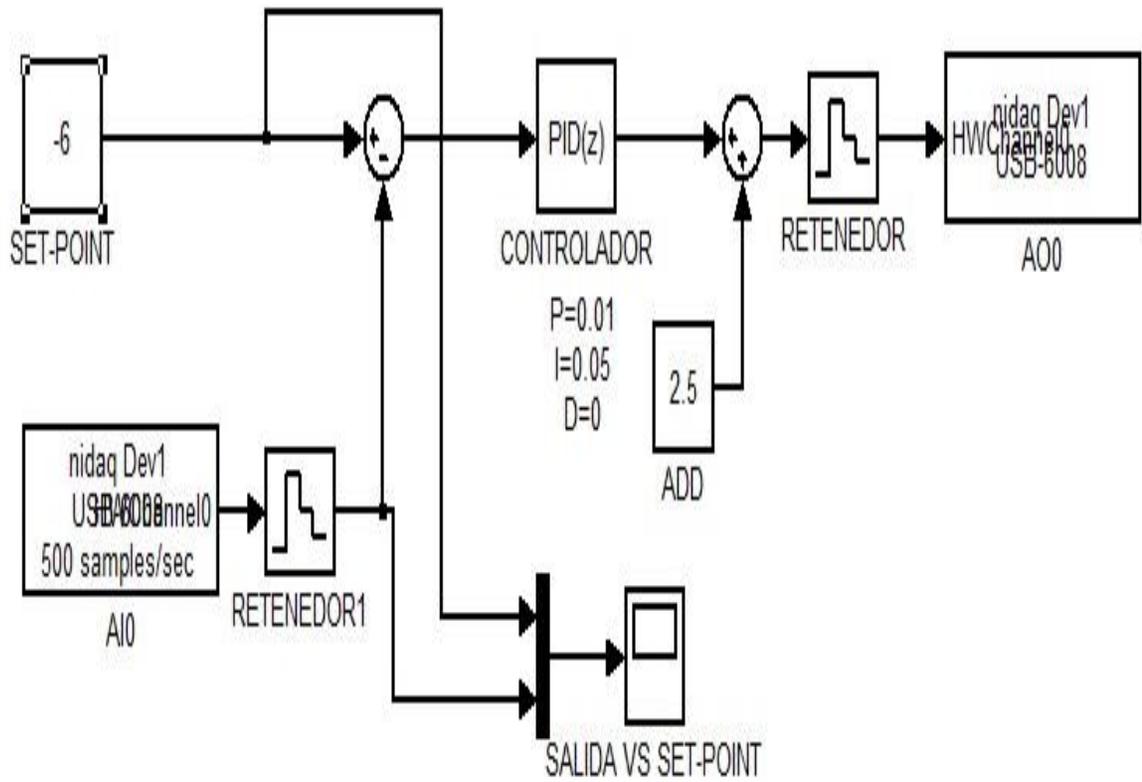
Ya que se contó solo con dos tarjetas NI-USB 6008/6009 y dos motores DC, se realizó las siguientes pruebas para evaluar la velocidad de comportamiento del control PID de los motores:

- Se evaluó el comportamiento de control utilizando una tarjeta NI-USB 6008/6009 y un motor DC.
- Se evaluó el comportamiento de control utilizando una tarjeta NI-USB 6008/6009 y dos motores DC.
- Se evaluó el comportamiento del control utilizando dos tarjetas NI-USB 6008/6009 y un motor DC por cada tarjeta NI-USB 6008/6009.

3.6.1. Evaluación de la velocidad de respuesta de control utilizando una tarjeta NI-USB 6008/6009 y un motor DC (una entrada y una salida de la tarjeta)

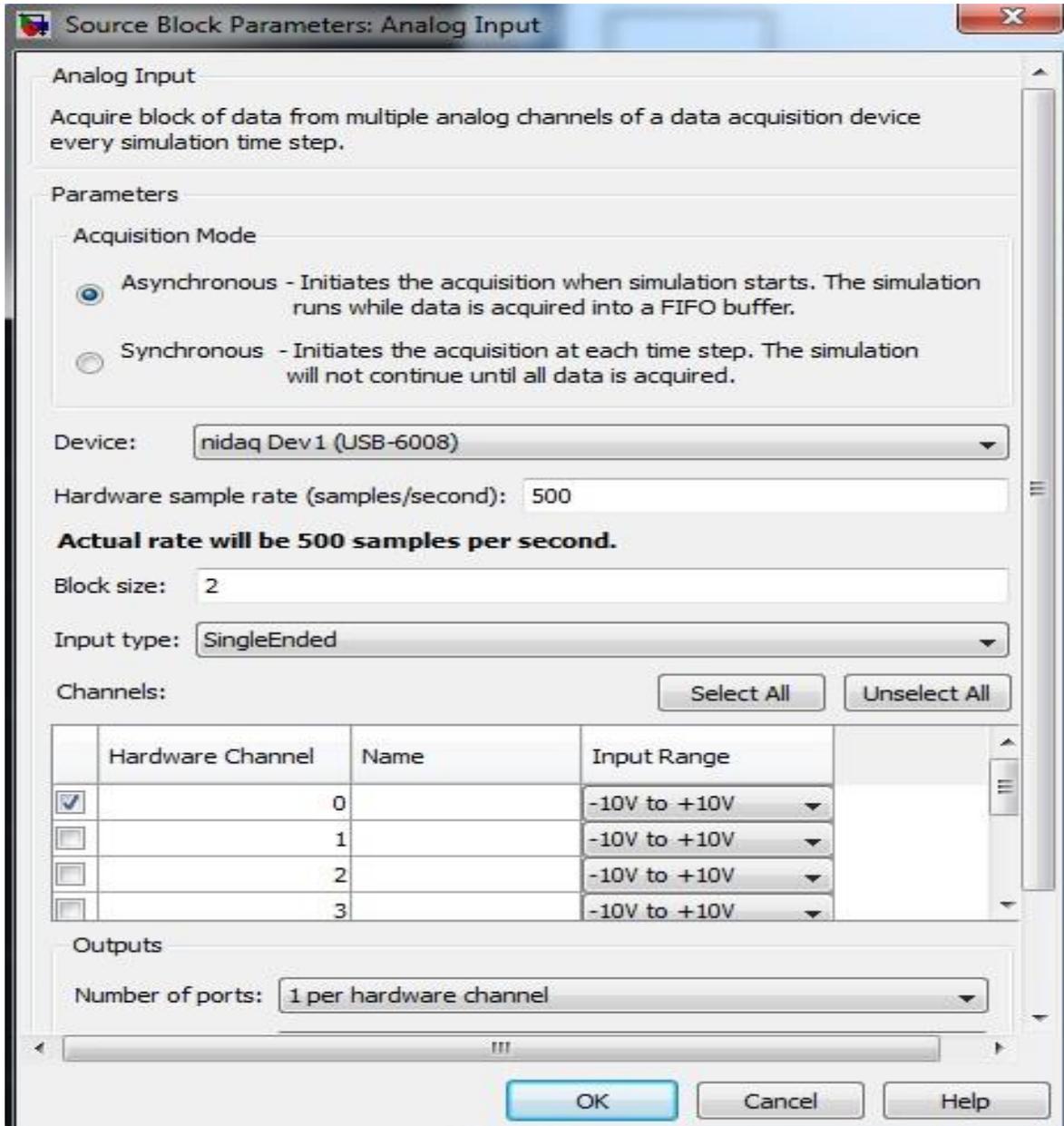
Para este caso, se diseñó un sistema de control PID en Simulink para el control de velocidad de un motor, en el cual se utilizó la toolbox de adquisición de datos para la entrada y la salida análogas de la tarjeta NI-USB 6008/6009, igualmente se utilizó el control PID, cómo se muestra en la figura 24:

Figura 24. Diseño en Simulink del control PID de velocidad de un motor DC por medio de una tarjeta NI-USB 6008/6009



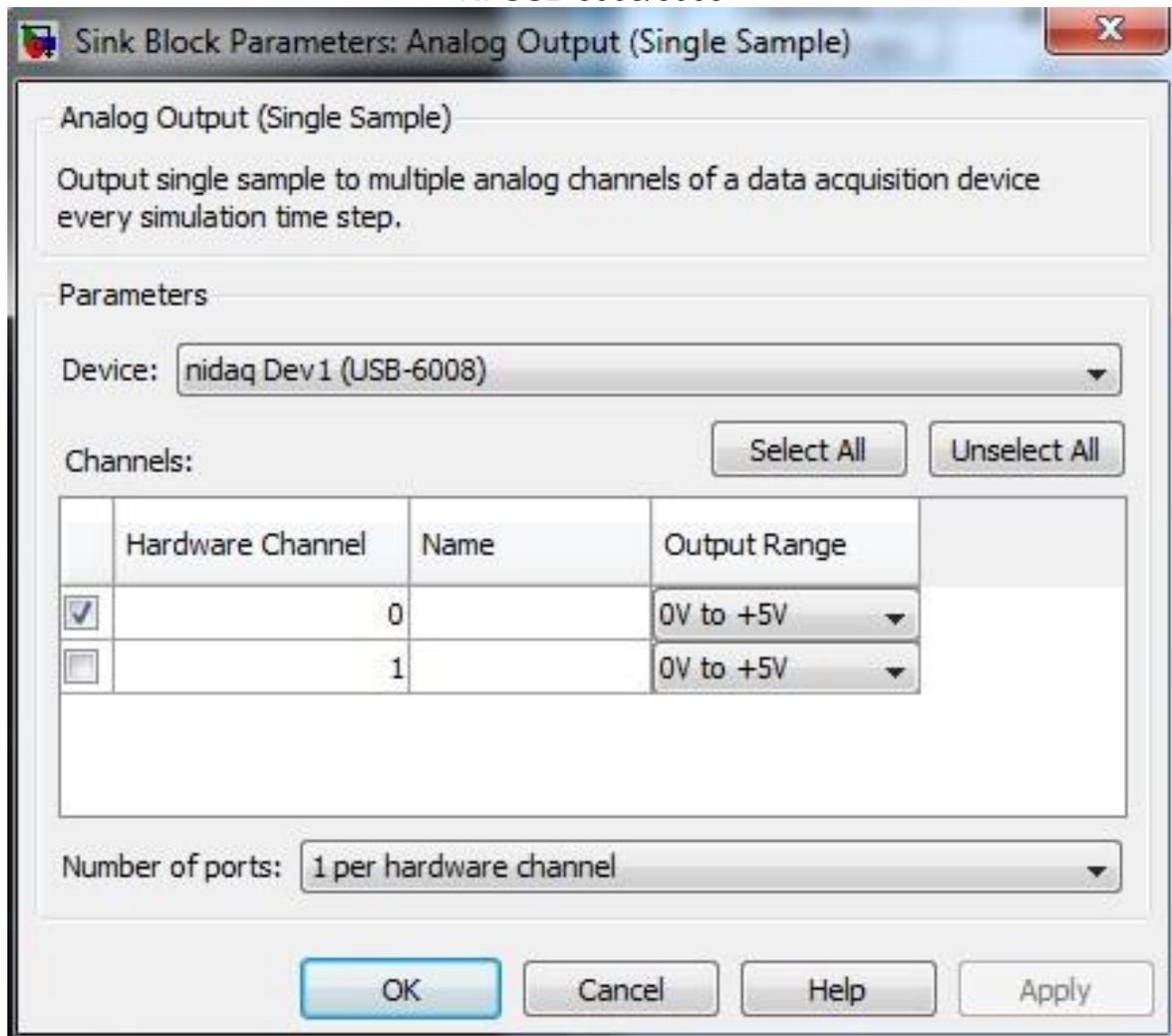
Como se pudo observar en la figura anterior, por medio de la toolbox de adquisición de datos de Matlab se configuró los parámetros de las entradas y las salidas análogas de la tarjeta NI-USB 6008/6009 de la siguiente manera:

Figura 25. Configuración de los parámetros de las entradas análogas de la tarjeta NI-USB 6008/6009



En la figura anterior se pueden observar los parámetros a configurar en el bloque de la entrada análoga de la tarjeta NI-USB 6008/6009, tales como el modo de adquisición, el dispositivo, el tiempo de muestreo, el tipo de entrada y cuales canales de entrada serán utilizados. En éste caso se utilizó un tiempo de muestreo de 500 muestras por segundo y se utilizó el canal 0 de la entrada análoga.

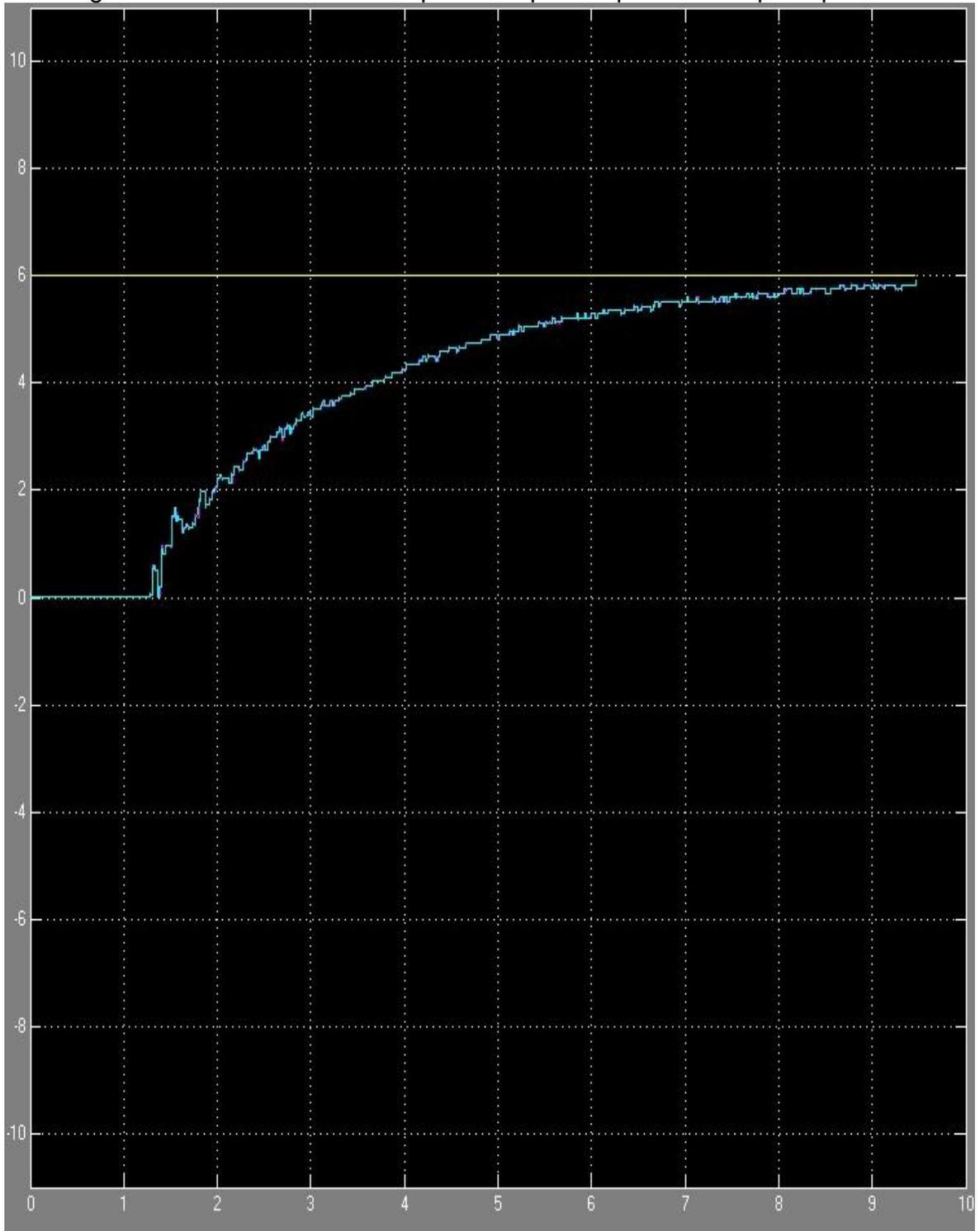
Figura 26. Configuración de los parámetros de las salidas análogas de la tarjeta NI-USB 6008/6009



En la figura anterior se pueden observar los parámetros a configurar en el bloque de la salida análoga de la tarjeta NI-USB 6008/6009. En éste caso se utilizó el canal 0 de la salida análoga.

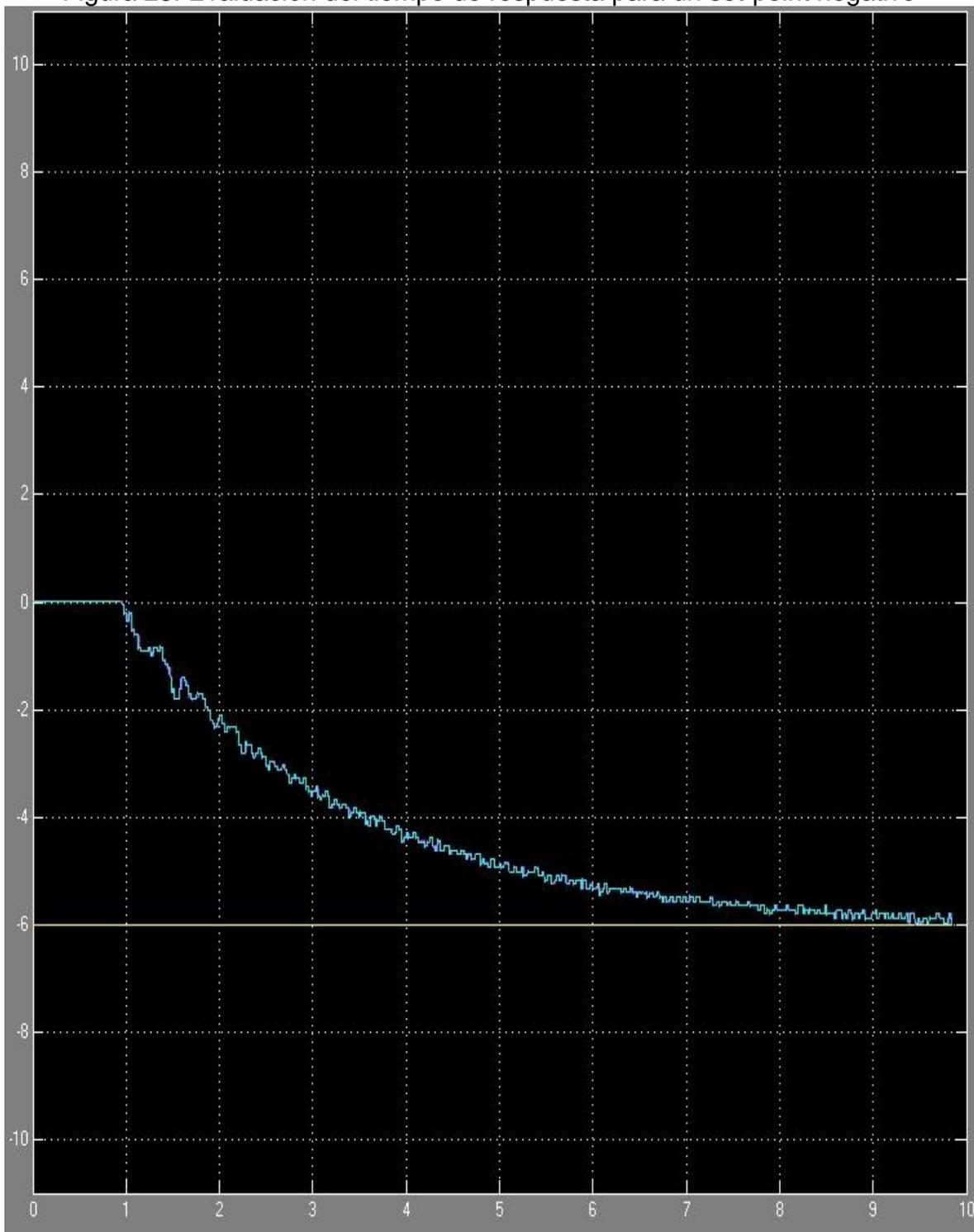
Se utilizó varios set-points para evaluar el tiempo de respuesta del control. Se evaluó el tiempo de respuesta con un set-point positivo, con un set-point negativo, pasando de set-point positivo a negativo, pasando de set-point negativo a positivo, pasando de un set-point positivo alto a uno más bajo y pasando de un set-point negativo alto a uno más bajo, donde se obtuvo los siguientes resultados:

Figura 27. Evaluación del tiempo de respuesta para un set-point positivo



Retardo de la señal de salida: 1.25 segundos

Figura 28. Evaluación del tiempo de respuesta para un set-point negativo



Retardo de la señal de salida: 1.1 segundos

Figura 29. Evaluación del tiempo de respuesta para el paso de un set-point positivo a un set-point negativo



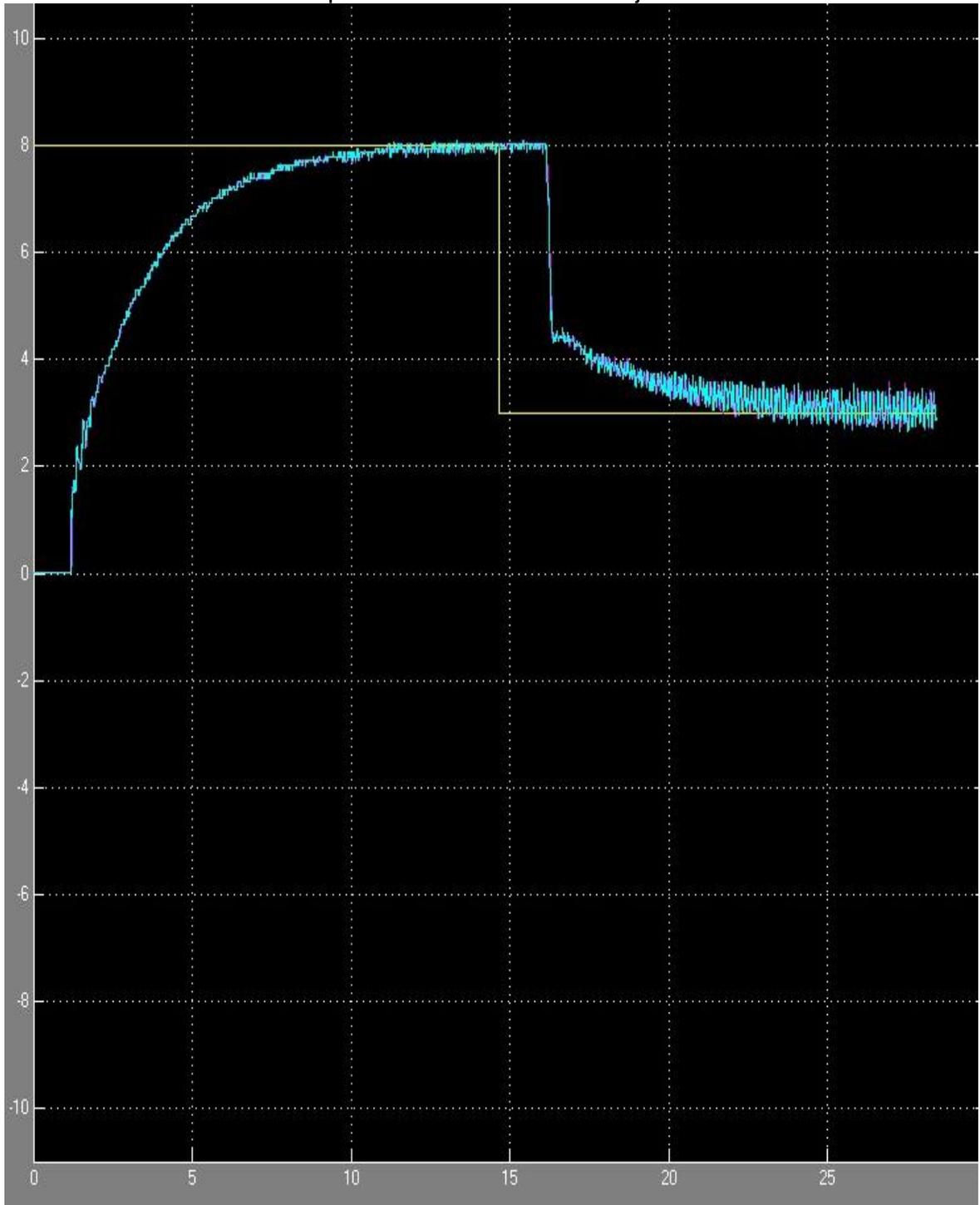
Retardo de la señal de salida: 1.8 segundos

Figura 30. Evaluación del tiempo de respuesta para el paso de un set-point negativo a un set-point positivo



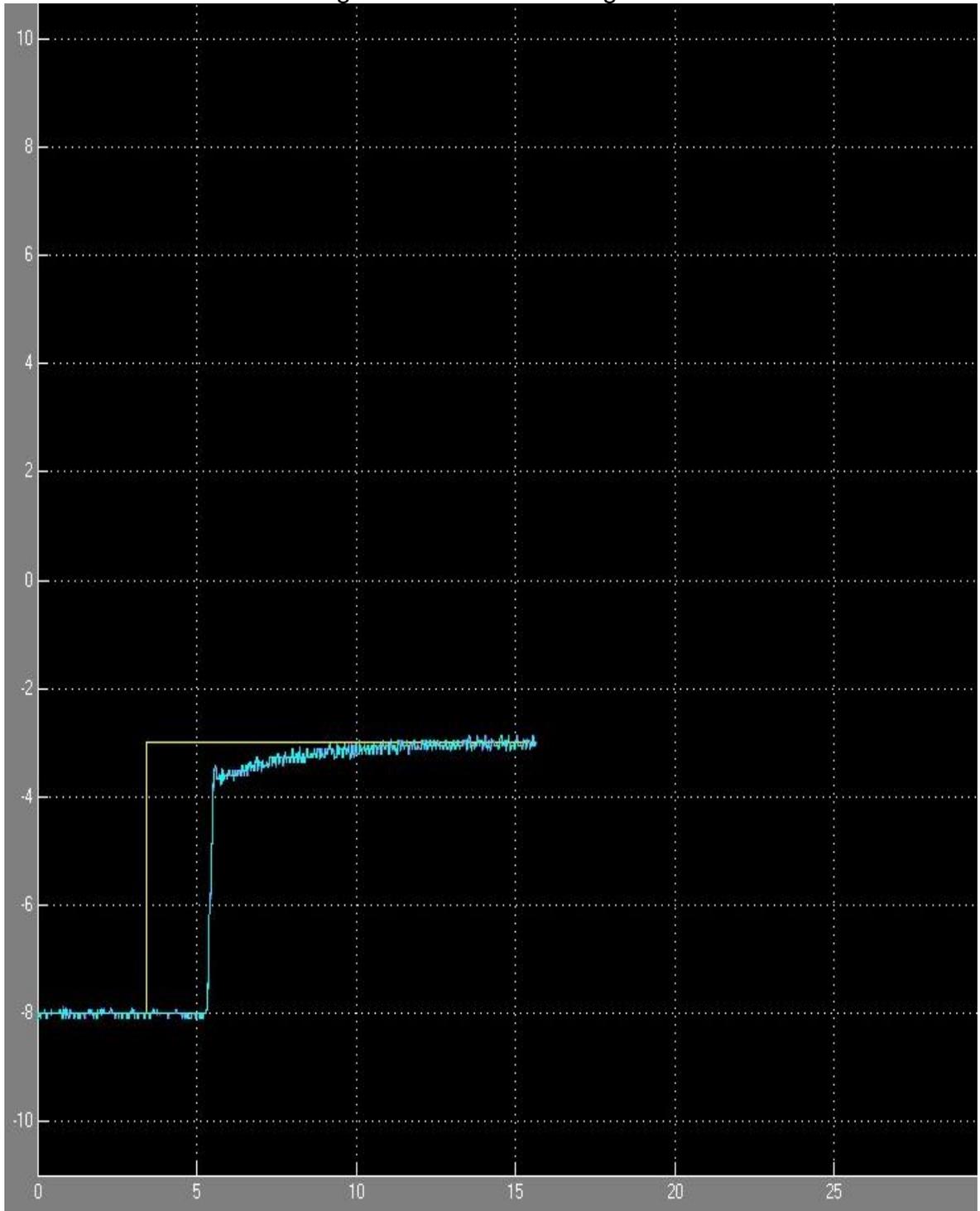
Retardo de la señal de salida: 1.4 segundos

Figura 31. Evaluación del tiempo de respuesta para el paso de un set-point positivo alto a uno más bajo



Retardo de la señal de salida: 1.4 segundos

Figura 32. Evaluación del tiempo de respuesta para el paso de un set-point negativo a uno menos negativo



Retardo de la señal de salida: 1.8 segundos

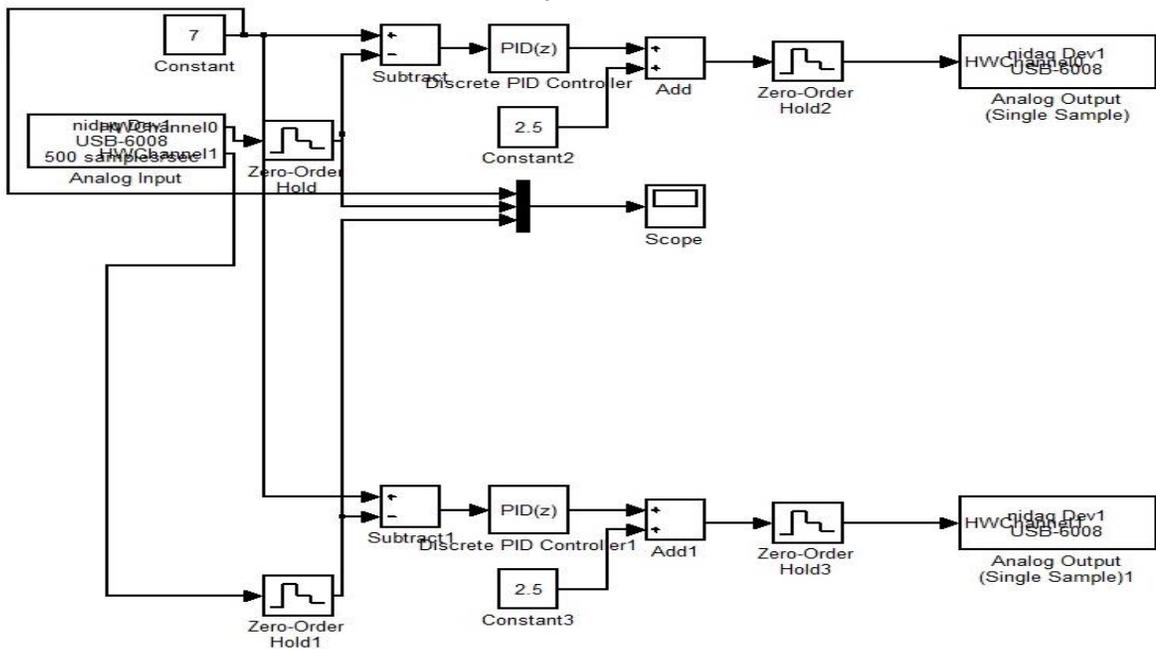
Tabla 4. Resultado de las pruebas realizadas para evaluar la velocidad de respuesta de control utilizando una tarjeta NI-USB 6008/6009 y un motor DC (una entrada y una salida de la tarjeta)

Tiempos (segundos)	UNA TARJETA Y UN MOTOR
Arranque (+)	1.25
Arranque (-)	1.1
De (+) a (-)	1.8
De (-) a (+)	1.4
Voltajes positivos	1.4
Voltajes negativos	1.8
PROMEDIO (segundos)	1.46

3.6.2. Evaluación de la velocidad de respuesta de control utilizando una tarjeta NI-USB 6008/6009 y dos motores DC (dos entradas y dos salidas de la tarjeta)

Para este caso, se diseñó un sistema de control PID en Simulink para el control de velocidad de un motor, en el cual se utilizó la toolbox de adquisición de datos para las dos entradas y las dos salidas análogas de la tarjeta NI-USB 6008/6009, igualmente se utilizó el control PID, cómo se muestra en la figura 33:

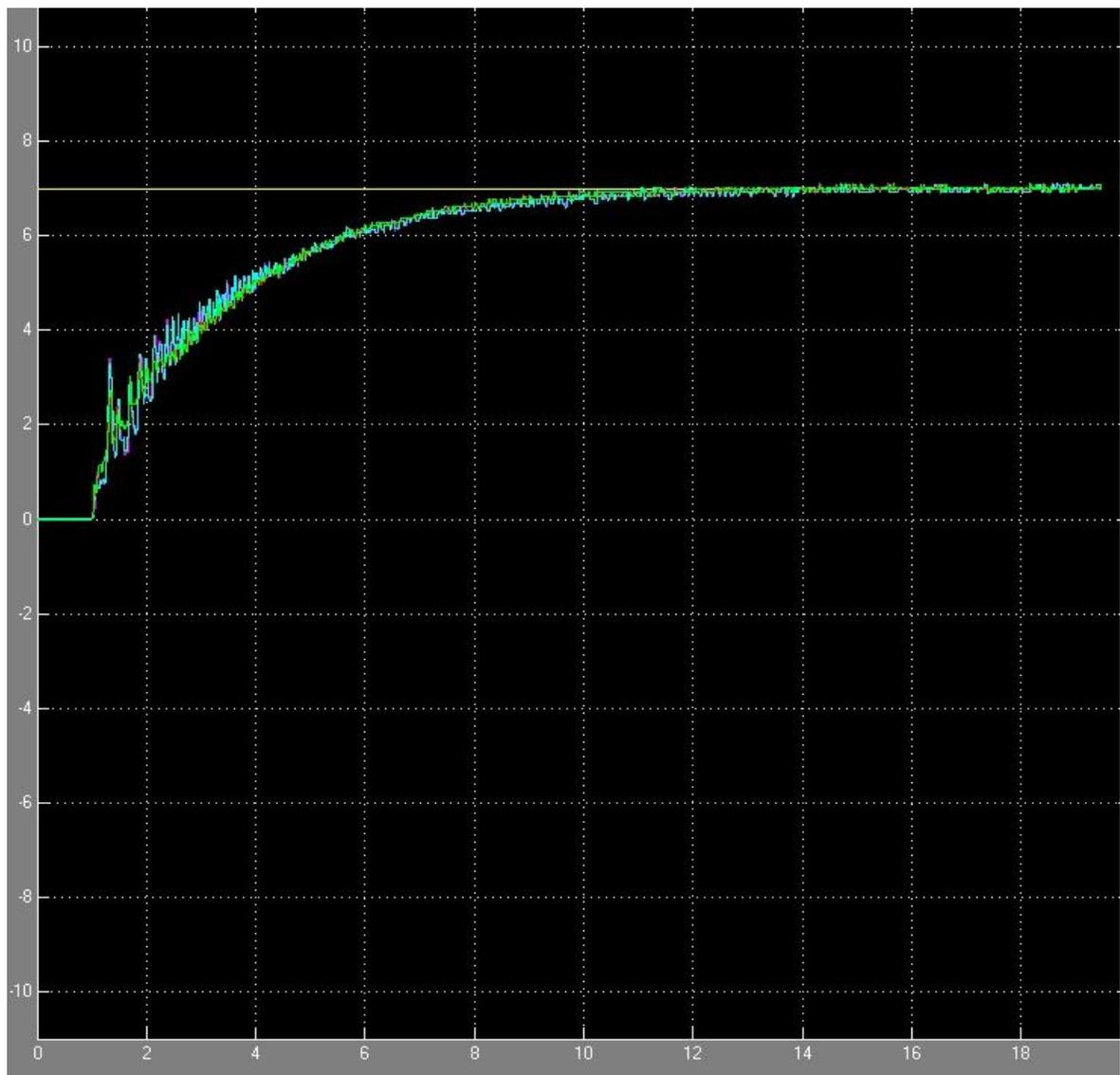
Figura 33. Diseño en Simulink del control PID de velocidad de dos motores DC por medio de una tarjeta NI-USB 6008/6009



Se configuró los parámetros de los bloques de entrada y salida de la tarjeta NI-USB 6008/6009 de acuerdo con el ejemplo anterior, solo que para este caso se utilizaron dos entradas análogas y dos salidas análogas.

Se utilizó varios set-points para evaluar el tiempo de respuesta del control al igual que en el ejemplo anterior, y se obtuvo los siguientes resultados:

Figura 34. Evaluación del tiempo de respuesta para un set-point positivo

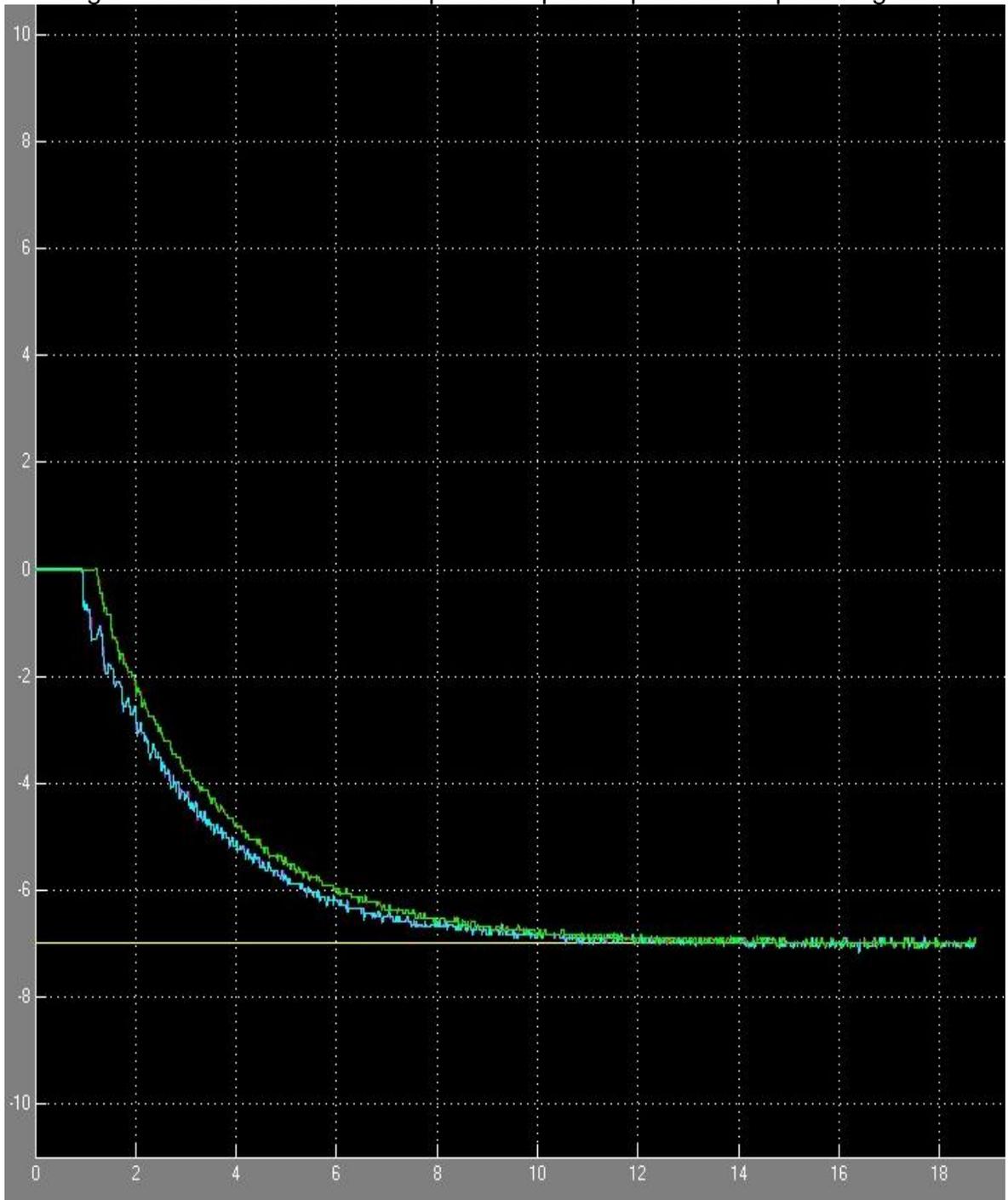


Retardo de la señal de salida del primer motor: 1.4 segundo

Retardo de la señal de salida del segundo motor: 1.5 segundos

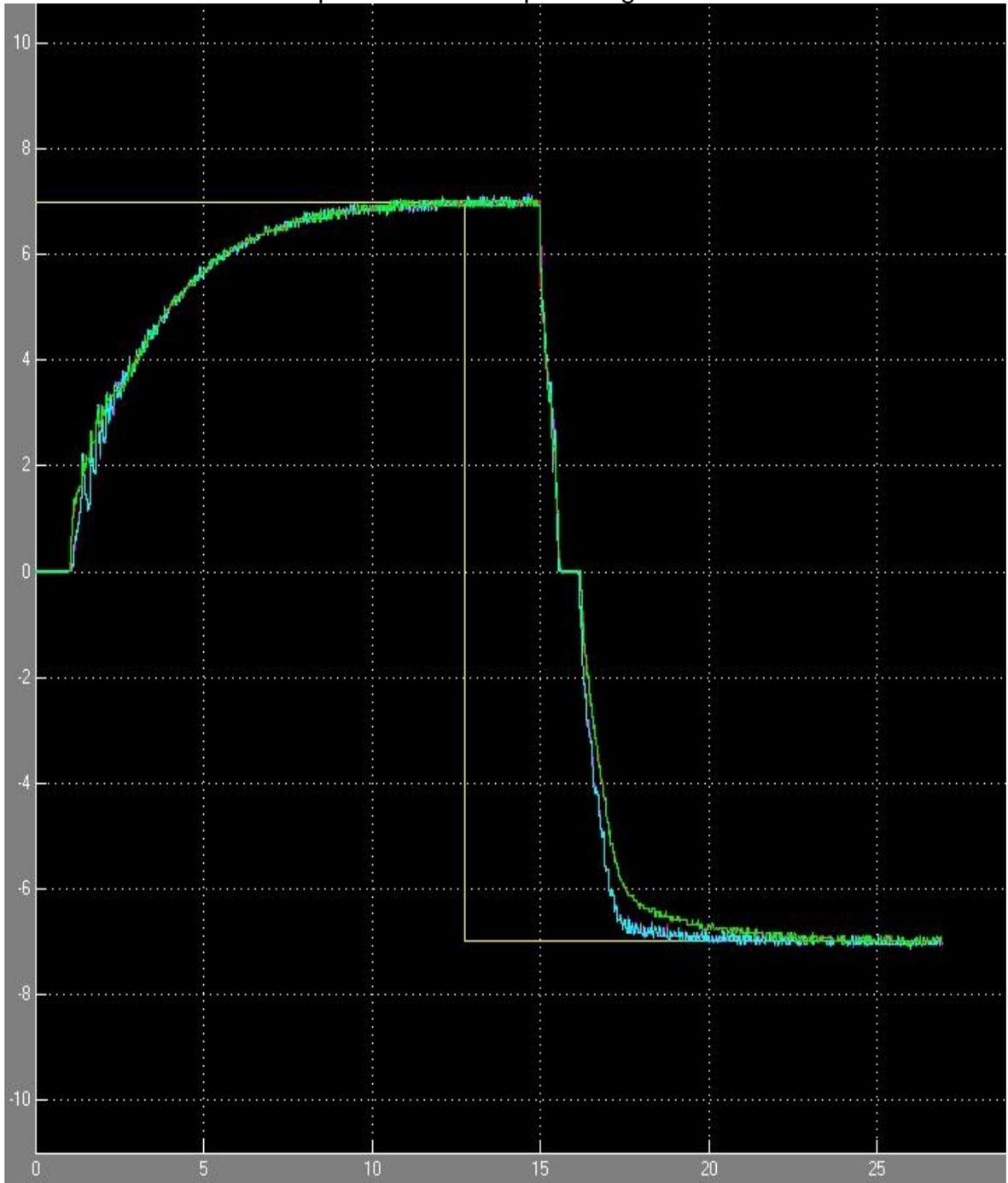
Retardo de la señal del segundo motor con respecto del primero: 0.1 segundos

Figura 35. Evaluación del tiempo de respuesta para un set-point negativo



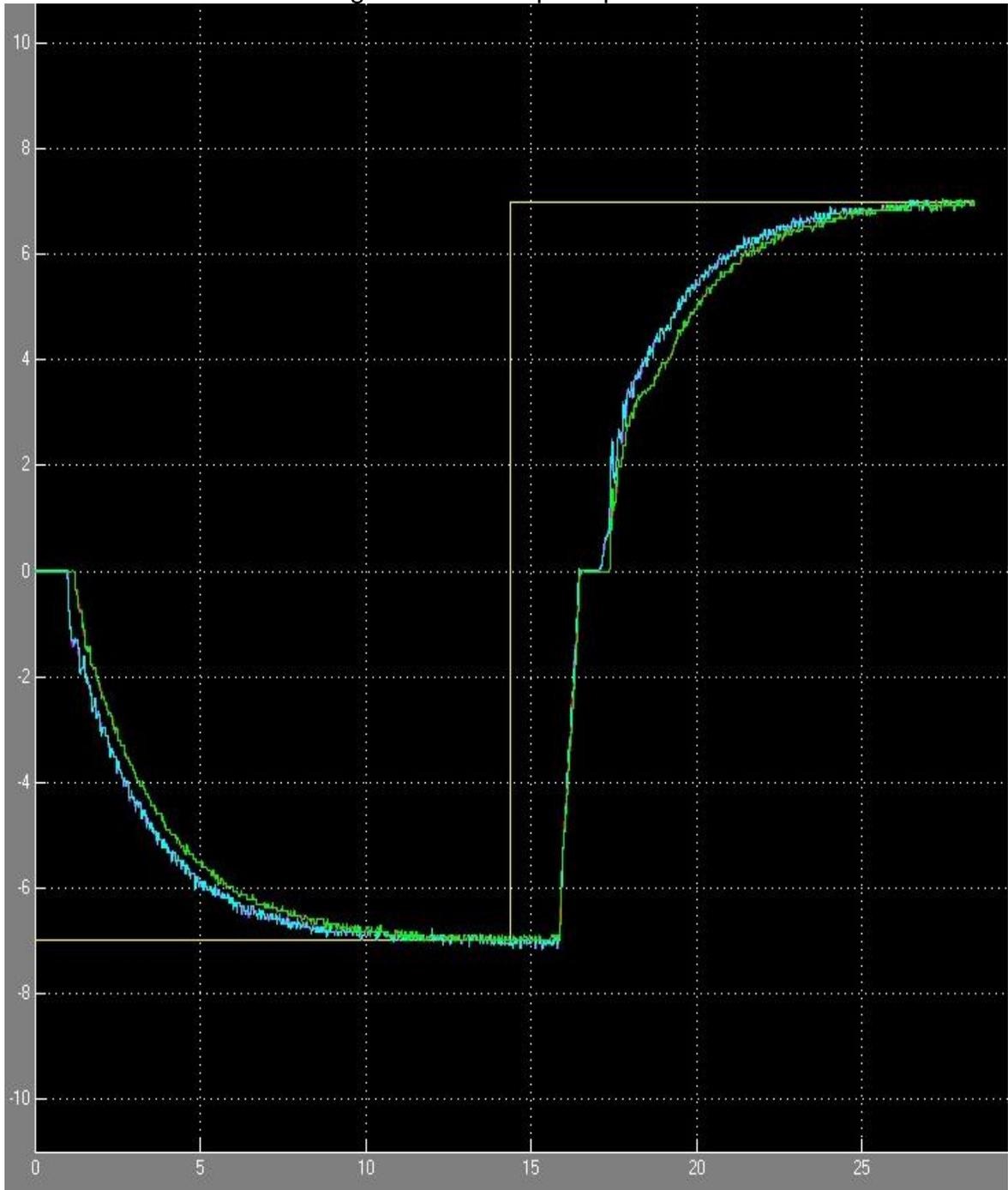
Retardo de la señal de salida del primer motor: 1.1 segundos
Retardo de la señal de salida del segundo motor: 1.4 segundos
Retardo de la señal del segundo motor con respecto del primero: 0.3 segundos

Figura 36. Evaluación del tiempo de respuesta para el paso de un set-point positivo a un set-point negativo



Retardo de la señal de salida del primer motor: 2.0 segundos
Retardo de la señal de salida del segundo motor: 2.2 segundos
Retardo de la señal del segundo motor con respecto del primero: 0.2 segundos

Figura 37. Evaluación del tiempo de respuesta para el paso de un set-point negativo a un set-point positivo



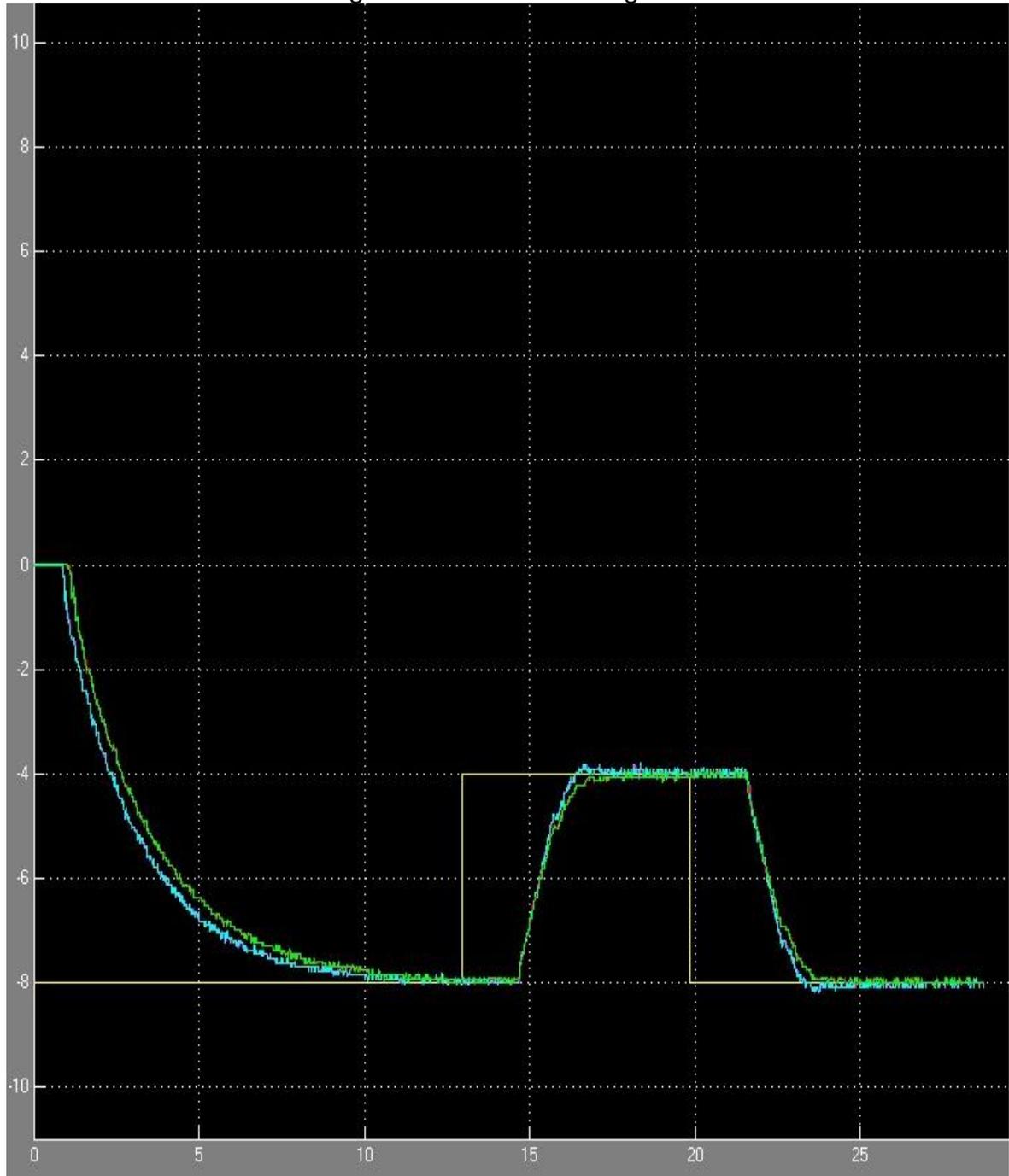
Retardo de la señal de salida del primer motor: 1.6 segundos
Retardo de la señal de salida del segundo motor: 1.8 segundos
Retardo de la señal del segundo motor con respecto del primero: 0.2 segundos

Figura 38. Evaluación del tiempo de respuesta para el paso de un set-point positivo alto a uno más bajo



Retardo de la señal de salida del primer motor: 1.5 segundos
Retardo de la señal de salida del segundo motor: 1.6 segundos
Retardo de la señal del segundo motor con respecto del primero: 0.1 segundos

Figura 39. Evaluación del tiempo de respuesta para el paso de un set-point negativo a uno menos negativo



Retardo de la señal de salida del primer motor: 1.95 segundos
Retardo de la señal de salida del segundo motor: 2.1 segundos
Retardo de la señal del segundo motor con respecto del primero: 0.15 segundos

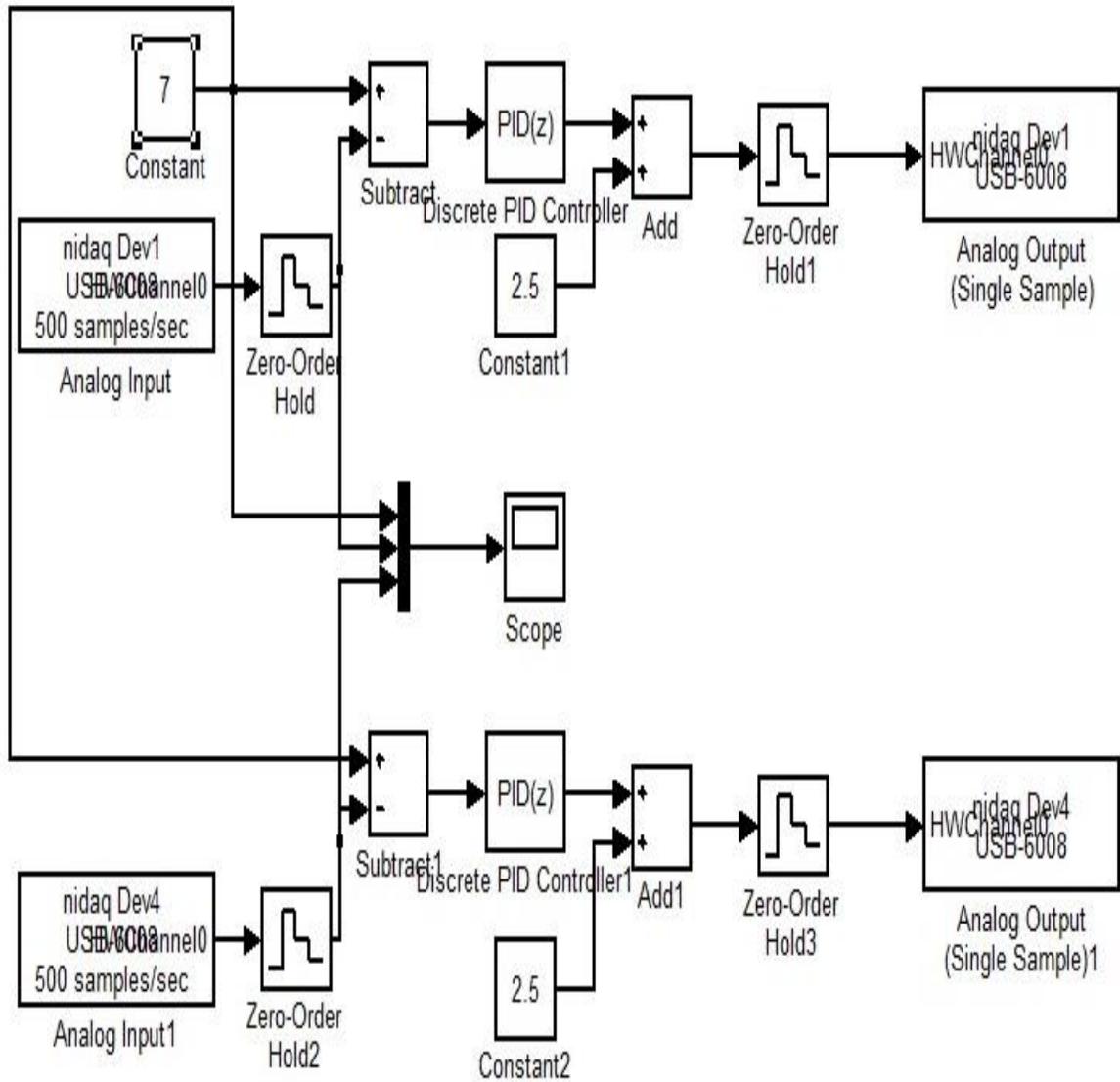
Tabla 5. Resultado de las pruebas realizadas para evaluar la velocidad de respuesta de control utilizando una tarjeta NI-USB 6008/6009 y dos motores DC (dos entradas y dos salidas de la tarjeta)

Tiempo (segundos)		UNA TARJETA Y DOS MOTORES
Arranque (+)	Motor 1	1.4
	Motor 2	1.5
	Retardo	0.1
Arranque (-)	Motor 1	1.1
	Motor 2	1.4
	Retardo	0.3
De (+) a (-)	Motor 1	2.0
	Motor 2	2.2
	Retardo	0.2
De (-) a (+)	Motor 1	1.6
	Motor 2	1.8
	Retardo	0.2
Voltajes positivos	Motor 1	1.5
	Motor 2	1.6
	Retardo	0.1
Voltajes negativos	Motor 1	1.95
	Motor 2	2.1
	Retardo	0.15
PROMEDIO (segundos)	Respuesta	1.68
	Retardo	0.175

3.6.3. Evaluación de la velocidad de respuesta de control utilizando dos tarjetas NI-USB 6008/6009 y dos motores DC (una entrada y una salida de cada tarjeta)

Para este caso, se diseñó un sistema de control PID en Simulink para el control de velocidad de un motor, en el cual se utilizó La toolbox de adquisición de datos para la entrada y la salida análogas de las dos tarjetas NI-USB 6008/6009, igualmente se utilizó el control PID, cómo se muestra en la figura 40:

Figura 40. Diseño en Simulink del control PID de velocidad de dos motores DC por medio de dos tarjetas NI-USB 6008/6009



Se configuró los parámetros de los bloques de entrada y salida de las tarjetas NI-USB 6008/6009 de acuerdo con los ejemplos anteriores, solo que para este caso se utilizaron una entrada y una salida análogas por cada tarjeta.

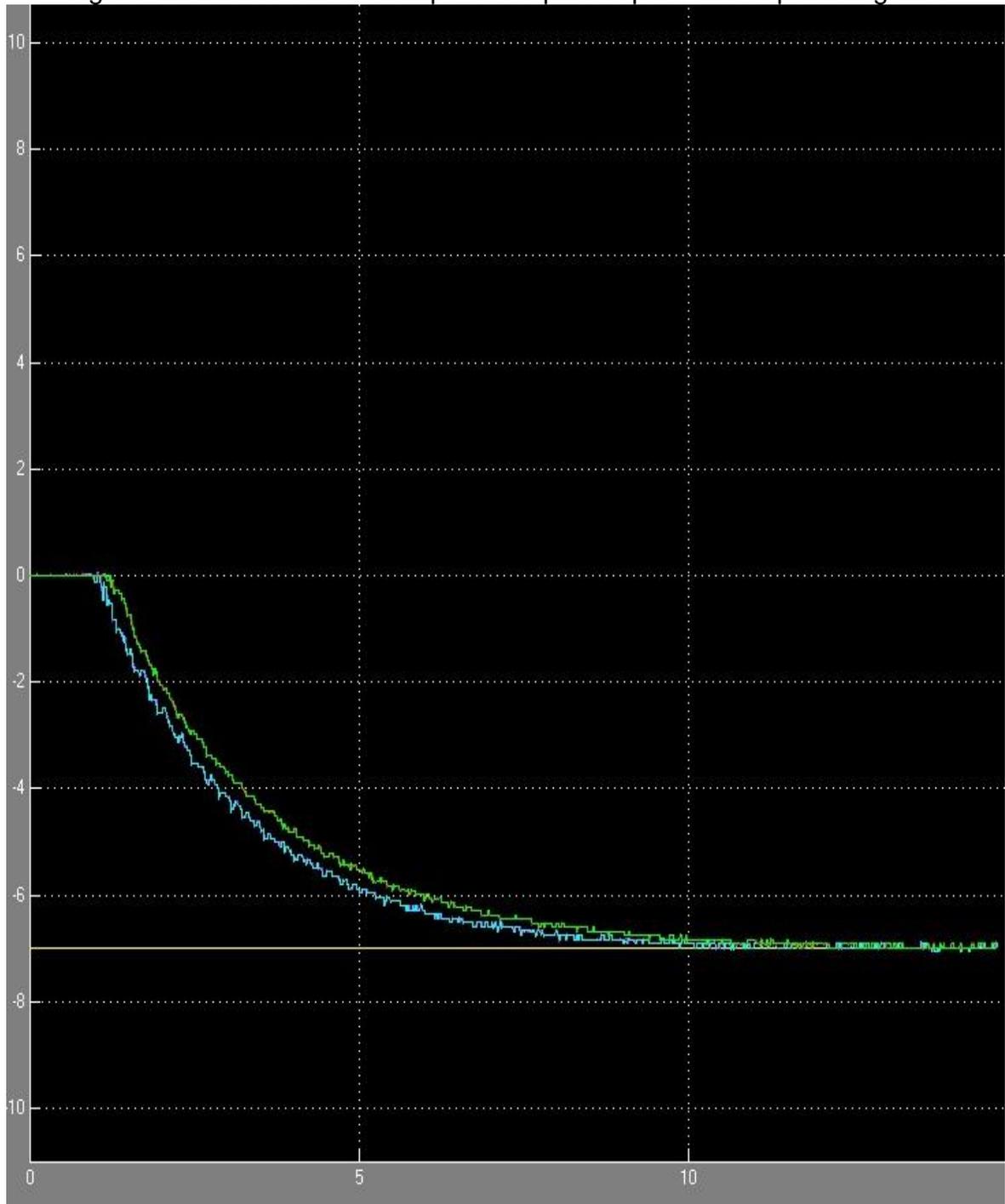
Se utilizó varios set-points para evaluar el tiempo de respuesta del control al igual que en los ejemplos anteriores, y se obtuvo los siguientes resultados:

Figura 41. Evaluación del tiempo de respuesta para un set-point positivo



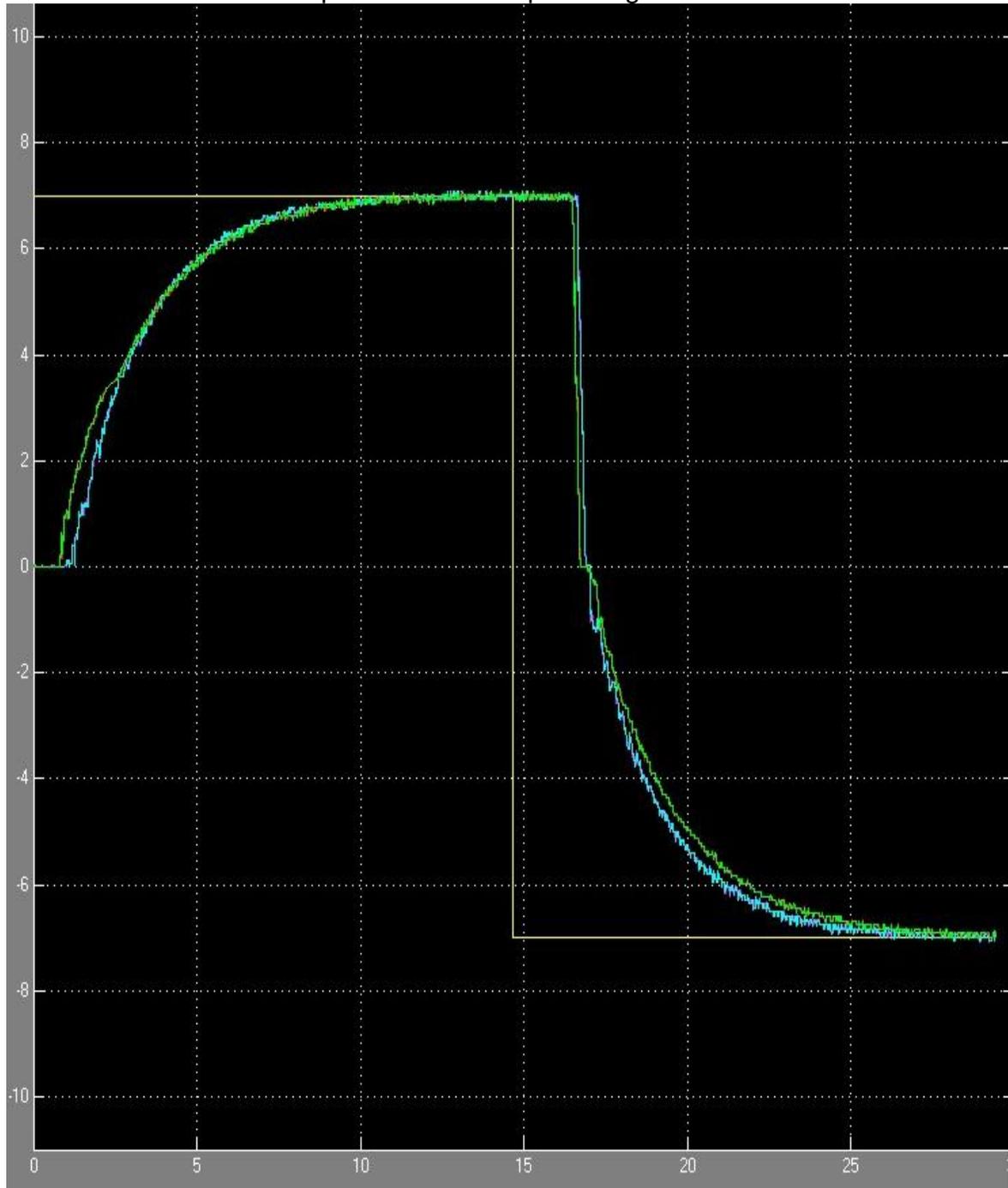
Retardo de la señal de salida del primer motor: 1.0 segundos
Retardo de la señal de salida del segundo motor: 1.3 segundos
Retardo de la señal del segundo motor con respecto del primero: 0.3 segundos

Figura 42. Evaluación del tiempo de respuesta para un set-point negativo



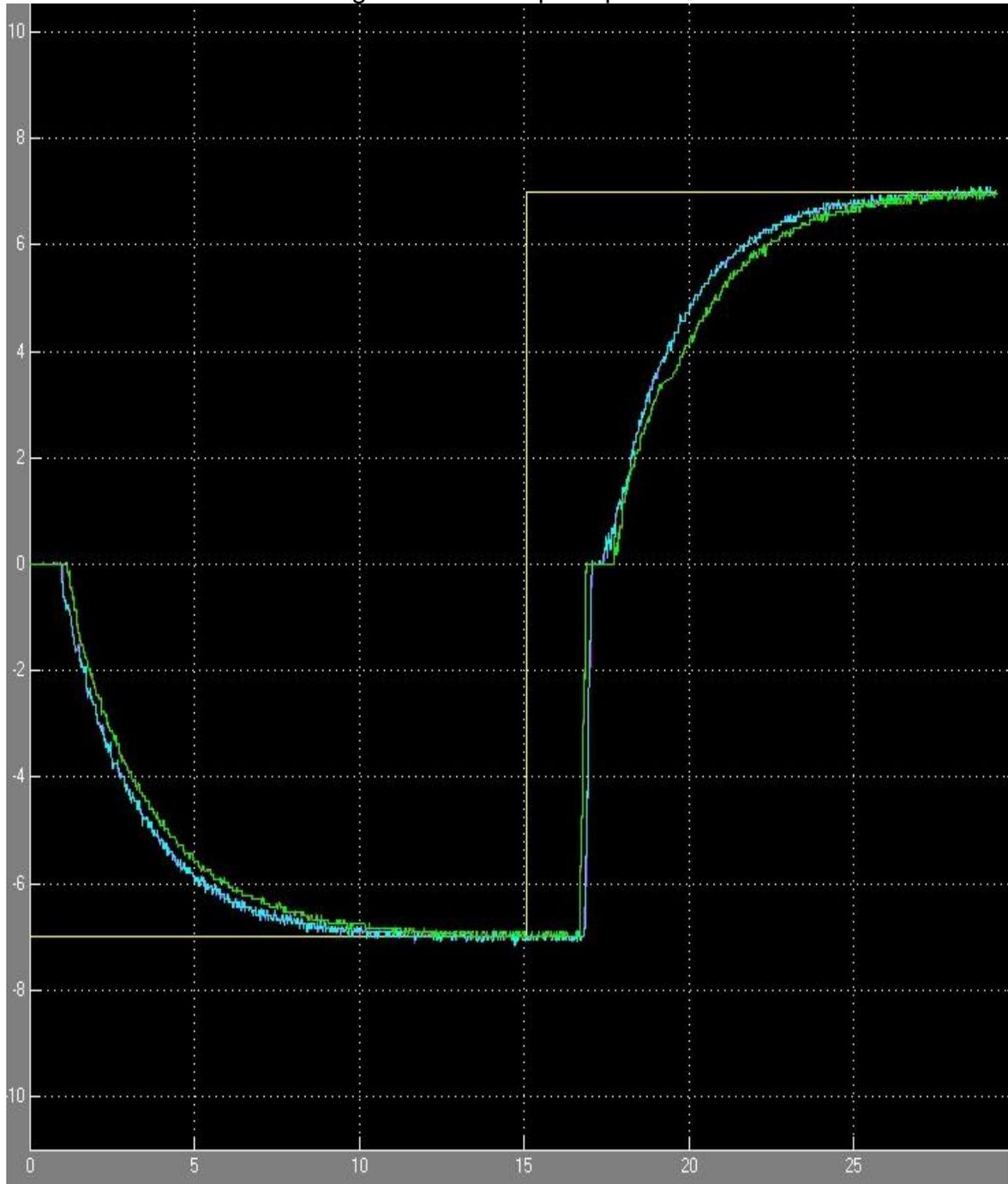
Retardo de la señal de salida del primer motor: 1.2 segundos
Retardo de la señal de salida del segundo motor: 1.4 segundos
Retardo de la señal del segundo motor con respecto del primero: 0.2 segundos

Figura 43. Evaluación del tiempo de respuesta para el paso de un set-point positivo a un set-point negativo



Retardo de la señal de salida del primer motor: 1.8 segundos
Retardo de la señal de salida del segundo motor: 2.1 segundos
Retardo de la señal del segundo motor con respecto del primero: 0.3 segundos

Figura 44. Evaluación del tiempo de respuesta para el paso de un set-point negativo a un set-point positivo



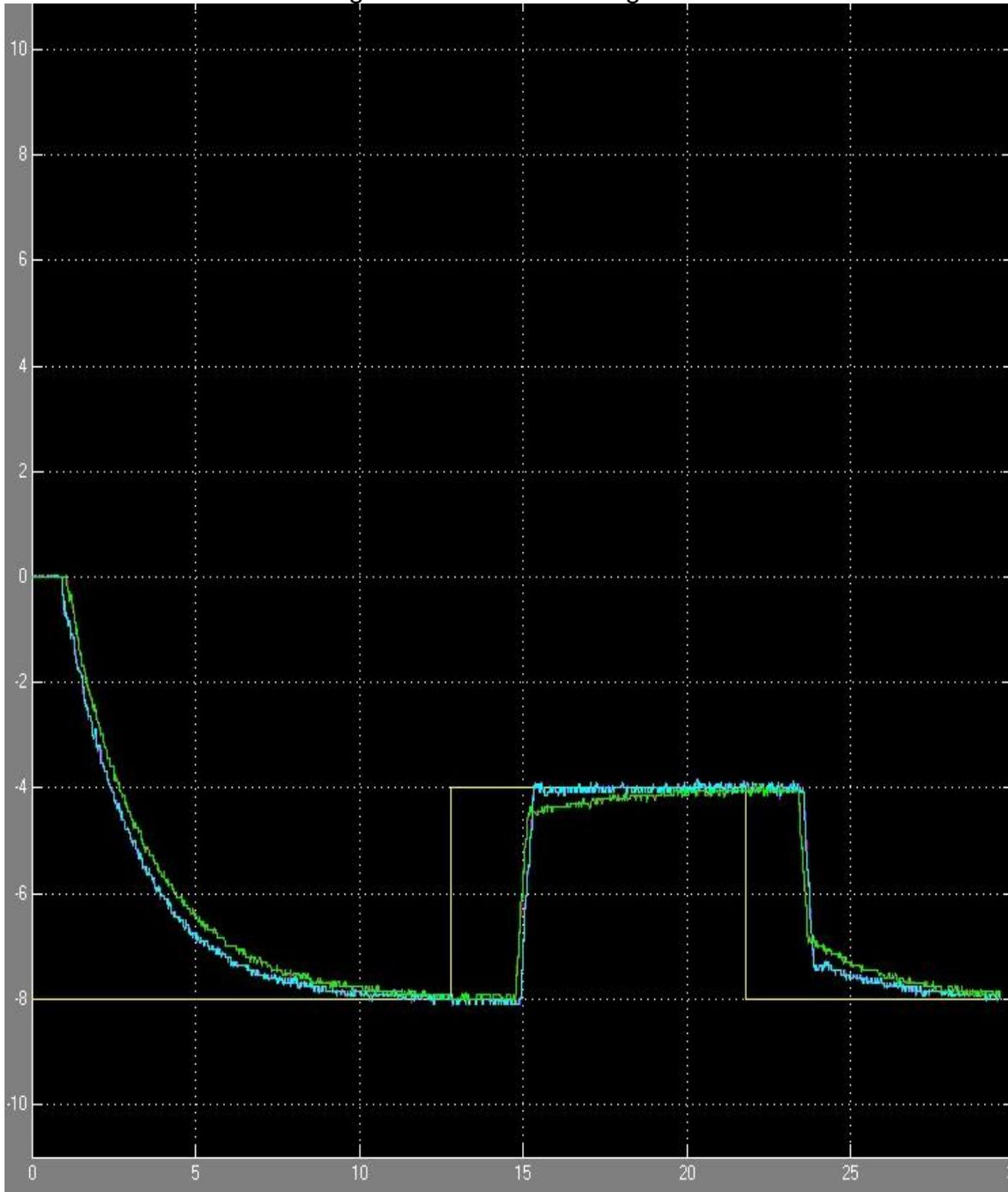
Retardo de la señal de salida del primer motor: 1.9 segundos
Retardo de la señal de salida del segundo motor: 2.2 segundos
Retardo de la señal del segundo motor con respecto del primero: 0.3 segundos

Figura 45. Evaluación del tiempo de respuesta para el paso de un set-point positivo alto a uno más bajo



Retardo de la señal de salida del primer motor: 1.9 segundos
Retardo de la señal de salida del segundo motor: 2.1 segundos
Retardo de la señal del segundo motor con respecto del primero: 0.2 segundos

Figura 46. Evaluación del tiempo de respuesta para el paso de un set-point negativo a uno menos negativo



Retardo de la señal de salida del primer motor: 1.9 segundos
Retardo de la señal de salida del segundo motor: 2.0 segundos
Retardo de la señal del segundo motor con respecto del primero: 0.1 segundos

Tabla 6. Resultado de las pruebas realizadas para evaluar la velocidad de comportamiento del control PID de los motores

Tiempo (segundos)		UNA TARJETA Y UN MOTOR	UNA TARJETA Y DOS MOTORES	DOS TARJETAS Y DOS MOTORES
Arranque (+)	Motor 1	1.25	1.4	1
	Motor 2		1.5	1.3
	Retardo		0.1	0.3
Arranque (-)	Motor 1	1.1	1.1	1.2
	Motor 2		1.4	1.4
	Retardo		0.3	0.2
De (+) a (-)	Motor 1	1.8	2.0	1.8
	Motor 2		2.2	2.1
	Retardo		0.2	0.3
De (-) a (+)	Motor 1	1.4	1.6	1.9
	Motor 2		1.8	2.2
	Retardo		0.2	0.3
Voltajes positivos	Motor 1	1.4	1.5	1.9
	Motor 2		1.6	2.1
	Retardo		0.1	0.2
Voltajes negativos	Motor 1	1.8	1.95	1.9
	Motor 2		2.1	2.0
	Retardo		0.15	0.1
PROMEDIO (segundos)	Respuesta	1.46	1.68	1.73
	Retardo		0.175	0.23

Al realizar el control de 2 motores con 2 tarjetas NI 6008/6009, se observó que cuando se corrió la simulación en Simulink, el sistema presentó retardos significativos al momento de realizar el control, llegando a la conclusión que para realizar control de posición, este sistema no sirve, porque él se demora mucho tiempo en responder y en lograr la posición (set-point) deseada, ya que los motores en el momento que se ejecuta el programa, comienzan a girar a su máxima velocidad mientras se encuentran en su retardo, comenzando así el control de estos motores, logrando llegar a la posición deseada, pero antes de lograr esta posición, el motor ya ha girado demasiadas veces y eso no es permitido para este tipo de control de posición, por lo tanto, se tomó la decisión junto con el director del proyecto, de cambiar el tipo de control, realizando así mejor un control de velocidad, donde estos retardos y la cantidad de giros que realiza el motor antes de llegar al set-point deseado, no son tan determinantes como en el caso de control de posición.

Al realizar el control de velocidad de éste sistema, se observó que en el momento de correr la simulación, el sistema se demoró un tiempo en comenzar a realizar el control mientras los motores se encuentran girando, y después de este tiempo (retardos) empieza a realizar el control de velocidad de los motores, llegando así a la velocidad (set-point) deseadas para cada motor. Lo que se hizo fue darle al sistema un valor (set-point) en Simulink, donde éste va a ser el valor de velocidad deseado, y al sistema le llega un valor en voltaje inyectado al motor para así incrementar o decrementar su velocidad, dependiendo de los valores de set-point a los que se quiera llegar.

4. CONCLUSIONES Y RECOMENDACIONES

La Toolbox de adquisición de datos de Matlab brinda el soporte de librerías y controladores de varios dispositivos, haciendo más sencillo el intercambio de datos de información entre Matlab y el Hardware.

Una de las ventajas que se tiene al utilizar Matlab, es que permite guardar datos como archivos a la hora de extraerlos y llevarlos al entorno de Matlab, para que se pueda posteriormente visualizarlos o manipularlos con el fin de ejecutar cualquier acción que se quiera realizar.

Para garantizar un correcto funcionamiento de la Toolbox de adquisición de datos de Matlab, se debe instalar y realizar una correcta configuración de las tarjetas NI USB 6008/6009.

Simulink permite una descripción de alto nivel del sistema de adquisición, ya que es una herramienta muy versátil para sacarle el mayor provecho a la Toolbox de adquisición de datos.

Se debe tener en cuenta que los datos que obtenemos con la Toolbox de adquisición de datos de Matlab a la hora de realizar una adquisición, no son en tiempo real, ya que se cuenta con varios factores que afectan nuestros tiempos de respuesta, como la memoria del PC y la velocidad del procesador.

Otra ventaja que se obtiene al utilizar Matlab, es su robustez, aplicabilidad y la práctica que se adquiere en el uso de este software en el transcurso de la vida universitaria.

Al evaluar el comportamiento del control de los motores DC con varias tarjetas NI USB 6008/6009 utilizando la toolbox de adquisición de datos de Matlab, se observó que el sistema a controlar responde demasiado lento en comparación con otros sistemas o programas, ya que a medida que se utiliza más entradas y más salidas análogas, se pudo observar cómo responde cada vez más lento el sistema, y al incluir el uso de más tarjetas NI USB 6008/6009, más lento todavía se comporta el sistema.

El sistema utilizado en este proyecto para realizar el control de posición de 2 motores por medio de 2 tarjetas NI USB 6008/6009 no sirve y no es recomendable para este tipo de control, por lo cual se optó por realizar mejor un control de velocidad de 2 motores por medio de 2 tarjetas NI USB 6008/6009.

El control de velocidad utilizado en este proyecto, sigue respondiendo de una manera lenta al sistema, pero la cantidad de oscilaciones que realizan los motores antes de hacer el control no son tan significativas como en el control de posición, siendo así un control lento en respuesta, pero se sabe que llega a la velocidad deseada.

En el momento de realizar el control de velocidad, se encontró con una dificultad bastante importante, y es que el software de control, en este caso Simulink de Matlab, se estaba bloqueando al momento de realizar el cambio de set-point para lograr que el motor que ya está en un set-point, llegara a otro set-point deseado. Esto se debe a que en el momento de ingresar otro set-point, el sistema no está corriendo en tiempo real y se escapan voltajes altos que van a ingresar a la tarjeta NI USB 6008/6009, y es de conocimiento que la tarjeta NI USB 6008/6009 solo trabaja voltajes de 0 a 5 Voltios, así que cuando le entran más de 5 Voltios, o voltajes negativos, la tarjeta se bloquea y no realiza el control que se quiere.

Se recomienda limitar los voltajes que le van a entrar a la tarjeta NI USB 6008/6009 entre 0 y 5 Voltios, con el fin de que si llegan voltajes mayores a 5 Voltios o en su defecto voltajes negativos, éstos no ingresen a la tarjeta NI USB 6008/6009 y así se evita bloqueos en el sistema.

Se recomienda la utilización de un motor reductor eléctrico de velocidad con el fin de realizar el control de posición de 2 motores por medio de 2 tarjetas NI USB 6008/6009.

Se recomienda realizar proyectos donde se logre desarrollar el control de posición de varios motores DC por medio de varias tarjetas NI USB 6008/6009.

BIBLIOGRAFÍA

[1] CARREÑO JAIMES KARINA, GABRIEL JIMENEZ NATALIA. Tutorial para la programación de las tarjetas NI-USB 6008/6009 con la Toolbox de adquisición de datos de Matlab.

[2] RODRIGUEZ SARMIENTO CARLOS, RODRIGUEZ SARMIENTO WILLIAM. Implementación de la Toolbox de adquisición de datos de Matlab y prácticas de laboratorio con el uso de las tarjetas NI-USB 6008/6009.

[3] Data Acquisition Toolbox User's Guide. Copyright 1999-2007 by the Mathworks, Inc. Disponible en: <http://www.mathworks.com>

[4] HENDER MOLINA, LISBETH ROMAN. Tutorial de Matlab Simulink. Disponible en: <http://www.angelfire.com/la/hmolina/matlab7.html>

[5] RAMÓN PALLÁS ARENY. Adquisición y Distribución de Señales. Editorial MARCOMBO.

[6] User guide and specifications USB-6008/6009. Disponible en: http://www.tau.ac.il/~electro/pdf_files/computer/ni_6008_ADC_manual.pdf

[7] FITZGERALD, A. E., CHARLES KINGSLEY, STEPHEN D. UMANS. Máquinas Eléctricas. 6ª Edición. Editorial MC GRAW HILL

[8] CUENCA, ANDRÉS. Encoders, Tipos. Disponible en: <http://www.forosdeelectronica.com/f16/encoders-informacion-tecnica-25/>