

DESARROLLO DE UN SOFTWARE DE SIMULACIÓN MULTIPLATAFORMA
PARA SISTEMAS TERMODINÁMICOS CÍCLICOS DE POTENCIA Y
REFRIGERACIÓN

JORGE IVÁN PIMIENTA ALVERNIA

UNIVERSIDAD PONTIFICIA BOLIVARIANA
FACULTAD DE INGENIERÍA MECÁNICA
ESCUELA DE INGENIERÍAS
BUCARAMANGA
2015

DESARROLLO DE UN SOFTWARE DE SIMULACIÓN MULTIPLATAFORMA
PARA SISTEMAS TERMODINÁMICOS CÍCLICOS DE POTENCIA Y
REFRIGERACIÓN

JORGE IVÁN PIMIENTA ALVERNIA

Trabajo de grado presentado para optar al título de Ingeniero Mecánico.

Director del proyecto
M.Sc EDWIN JESÚS CÓRDOBA TUTA.
Ingeniero Mecánico

UNIVERSIDAD PONTIFICIA BOLIVARIANA
FACULTAD DE INGENIERÍA MECÁNICA
ESCUELA DE INGENIERÍAS
BUCARAMANGA
2015

NOTA DE ACEPTACIÓN:

Firma del presidente del jurado

Firma del jurado

Firma del jurado

Bucaramanga 16 de Marzo de 2015

*“A mis padres quienes me apoyaron y fueron patrocinadores de mis triunfos.
A mis hermanos Juan Sebastian y Santiago, que hacen parte importante de mi vida.
A mi pequeña Marly, que siempre ha estado para apoyarme y celebrar mis logros.
Y a Juan Manuel, que mas que un profesor es un amigo.”*

Jorge Ivan Pimienta Alvernia

AGRADECIMIENTOS

El autor de este proyecto expresa sus mas sinceros agradecimientos a:

La Facultad de Ingeniería Mecánica de la Universidad Pontificia Bolivariana, por abrirme sus puertas y brindarme maravillosas experiencias e invaluable enseñanzas.

Al ingeniero Edwin Jesús Córdoba Tuta, mi director de proyecto por su apoyo no solo en el desarrollo de este proyecto, sino también en el transcurso de mis estudios en la UPB.

A todos los docentes de ingeniería mecánica, especialmente a Juan Manuel Arguello por su apoyo, confianza y amistad a lo largo de estos años, y a los ingenieros Javier Castellanos, Emíl Hernandez y Miguel Ángel Reyes, por su paciencia y carisma para enseñar.

Finalmente a todos mis compañeros y amigos de la facultad de Ingeniería Mecánica, por compartir tan buenas experiencias.

CONTENIDO

	pag.
INTRODUCCIÓN	14
1. PLANTEAMIENTO DEL PROBLEMA	15
2. JUSTIFICACIÓN	16
3. OBJETIVOS	18
3.1 OBJETIVO GENERAL	18
3.2 OBJETIVOS ESPECÍFICOS	18
4. MARCO TEÓRICO	19
4.1 ESTADO DEL ARTE	19
4.1.1 Software de simulación de sistemas de conversión de energía	19
4.1.1.1 GateCycle™	19
4.1.1.2 Ariane™	20
4.1.1.3 THERMOFLEX™	20
4.1.1.4 Cycle-Tempo™	21
4.1.2 Cálculo de propiedades termodinámicas	21
4.1.2.1 Simulis Thermodynamics®	21
4.1.2.2 REFPROP™	22
4.1.2.3 CoolProp™	23
4.2 SISTEMAS	23
4.2.1 Sistemas cerrados	23
4.2.2 Volúmenes de control	24
4.3 ENERGÍA	24
4.3.1 Trabajo y potencia	24
4.3.2 Calor	25
4.4 PRIMERA LEY DE LA TERMODINÁMICA	25
4.4.1 Energía interna	26
4.4.2 Entalpía	26
4.5 VOLÚMENES DE CONTROL EN ESTADO ESTABLE	27
4.5.1 Ejemplos de Volúmenes de control en estado estable	27

4.5.1.1 Intercambiadores de calor	27
4.5.1.2 Difusores y toberas	27
4.5.1.3 Estranguladores	28
4.5.1.4 Turbinas	29
4.5.1.5 Bombas y compresores	29
4.5.1.6 Cámaras de mezclado	30
4.6 SEGUNDA LEY DE LA TERMODINÁMICA	31
4.7 CICLOS DE POTENCIA Y REFRIGERACIÓN	33
4.7.1 Máquina térmica	33
4.7.2 Ciclo Carnot	33
4.7.3 Ciclo Rankine	34
4.7.4 Ciclo Brayton	34
4.7.5 Ciclo de refrigeración por compresión	35
5. METODOLOGÍA	38
5.1 ENUNCIADO DEL PROBLEMA	38
5.2 LISTA DE REQUERIMIENTOS	38
5.3 INTERACCIÓN ENTRE EL USUARIO Y EL SOFTWARE	38
5.4 DISEÑO DE LA INTERFAZ DE USUARIO	39
5.4.1 Primer diseño	39
5.4.2 Implementaciones	39
5.5 MODELAMIENTO DEL SOFTWARE	40
5.6 DIAGRAMA DE CLASES	40
6. RESULTADOS Y DISCUSIÓN	44
6.1 SIMULACIONES DE CICLO RANKINE	44
6.1.1 Caso 1: Ciclo Rankine simple	44
6.1.1.1 Enunciado	44
6.1.1.2 Resultados	45
6.1.1.3 Análisis de resultados del caso	45
6.1.2 Caso 2: Ciclo Rankine con recalentamiento	45
6.1.2.1 Enunciado	45
6.1.2.2 Resultados	46
6.1.2.3 Análisis de resultados del caso	46

6.1.3 Caso 3: Ciclo Rankine regenerativo con CAA Abierto	47
6.1.3.1 Enunciado	47
6.1.3.2 Resultados	48
6.1.3.3 Análisis de resultados del caso	48
6.1.4 Caso 4: Ciclo Rankine regenerativo con recalentamiento	48
6.1.4.1 Enunciado	48
6.1.4.2 Resultados	49
6.1.4.3 Análisis de resultados del caso	49
6.2 SIMULACIÓN DE CICLOS DE REFRIGERACIÓN	51
6.2.1 Caso 5: Refrigeración por compresión	51
6.2.1.1 Enunciado	51
6.2.1.2 Resultados	51
6.2.1.3 Análisis de de resultados del caso	52
7. CONCLUSIONES Y RECOMENDACIONES	54
APÉNDICE	55

LISTA DE FIGURAS

	pag.
Figura 1. Sistema térmico simple	16
Figura 2. Sistema térmico típico	17
Figura 3. Captura de pantalla de GateCycle™	19
Figura 4. Captura de pantalla de Ariane™	20
Figura 5. Captura de pantalla de THERMOFLEX™	21
Figura 6. Captura de pantalla de Cycle-Tempo™	22
Figura 7. Esquema de un sistema cerrado	24
Figura 8. Esquema de un sistema abierto	25
Figura 9. Esquema de un intercambiador de calor	28
Figura 10. Esquemas representativos de un difusor y una tobera	28
Figura 11. Diagrama ejemplo de un estrangulador	29
Figura 12. Ensamblaje de una turbina de vapor	30
Figura 13. Diagrama de una cámara de mezclado	31
Figura 14. Esquema de una máquina térmica	33
Figura 15. Diagrama T-s del ciclo de Carnot	34
Figura 16. Diagrama T-s de un ciclo Rankine ideal	35
Figura 17. Diagrama P-v de un ciclo Brayton ideal	36
Figura 18. Diagrama P-v de un ciclo Brayton ideal	37
Figura 19. Prototipado de la interfaz	40
Figura 20. Primera implementación	41
Figura 21. Implementación final	41
Figura 22. Modelamiento del proceso del cálculo de un ciclo	42
Figura 23. Modelamiento del proceso de creación del archivo xml	42
Figura 24. Diagrama de clases simplificado	43
Figura 25. Diagrama de la planta para el caso 1	44
Figura 26. Diagrama de la planta para el caso 2	46
Figura 27. Diagrama de la planta para el caso 3	47
Figura 28. Diagrama de la planta para el caso 4	49
Figura 29. Diagrama del refrigerador para el caso 1	52

LISTA DE TABLAS

	pag.
Tabla 1. Resultado de entalpía en cada estado	45
Tabla 2. Eficiencia térmica de la planta	45
Tabla 3. Resultado de entalpía en cada estado	46
Tabla 4. Eficiencia térmica de la planta	47
Tabla 5. Resultado de entalpía en cada estado	48
Tabla 6. Eficiencia térmica del sistema	48
Tabla 7. Resultado de entalpía en cada estado	50
Tabla 8. Eficiencia térmica de la planta	50
Tabla 9. Resultado de flujo en cada estado	51
Tabla 10. Resultado de entalpía en cada estado	52
Tabla 11. Eficiencia térmica de la planta	52

LISTA DE APÉNDICES

	pag.
A. INFORMES DE SIMULACIÓN	56
B. DIAGRAMA DE CLASES UML DEL SOFTWARE	83
C. MANUAL DE USUARIO DEL SOFTWARE	84
D. DOCUMENTACIÓN DEL SOFTWARE	92

RESUMEN GENERAL DE TRABAJO DE GRADO

TITULO: DESARROLLO DE UN SOFTWARE DE SIMULACIÓN MULTIPLATAFORMA PARA SISTEMAS TERMODINÁMICOS CÍCLICOS DE POTENCIA Y REFRIGERACIÓN

AUTOR(ES): JORGE IVAN PIMIENTA ALVERNIA

FACULTAD: Facultad de Ingeniería Mecánica

DIRECTOR(A): EDWIN JESÚS CÓRDOBA TUTA

RESUMEN

Este trabajo se centró en el desarrollo de un software de simulación para sistemas termodinámicos cíclicos de potencia y refrigeración. Durante el avance de este proyecto, se estudiaron los requerimientos del software, se diseñó la interfaz de usuario según parámetros de directivas de interacción humana, se modelaron los procesos principales para el funcionamiento del software para luego codificarlas. Una vez desarrollado, se corrieron simulaciones para validar los cálculos realizados y se cuantificó el error con respecto a la soluciones de diferentes autores. El software fue desarrollado en lenguaje C++ con orientación a objetos, haciendo uso de la biblioteca multiplataforma Qt, lo que garantizó la compatibilidad de la aplicación entre sistemas operativos.

PALABRAS CLAVES:

Termodinámica, Simulación, Ciclos de potencia, Ciclos de refrigeración

V° B° DIRECTOR DE TRABAJO DE GRADO

GENERAL SUMMARY OF WORK OF GRADE

TITLE: DEVELOPMENT OF A MULTI PLATFORM SIMULATION SOFTWARE FOR THERMODYNAMIC POWER SYSTEMS AND REFRIGERATION

AUTHOR(S): JORGE IVAN PIMIENTA ALVERNIA

FACULTY: Facultad de Ingeniería Mecánica

DIRECTOR: EDWIN JESÚS CÓRDOBA TUTA

ABSTRACT

The present work focus on the development of a multiplatform simulation software for power and refrigeration cycles. In the course of this project, the software requirements were studied, the user interface was designed according to human interaction guidelines, the principal software functionalities were modeled and coded after. Once developed, simulations were made for computations validation and the error was quantified against other authors solutions. The software was developed in C++ with object oriented programming, using the cross platform library Qt, ensuring cross compatibility between different operating systems.

KEYWORDS:

Thermodynamics, Simulation, Power cycles, Refrigeration cycles, Modeling, Software development.

V° B° DIRECTOR OF GRADUATE WORK

INTRODUCCIÓN

Los seres humanos, la mayoría sin siquiera notarlo, se encuentran en contacto con sistemas termodinámicos cíclicos en su diario vivir. Los sistemas de aire acondicionado, los refrigeradores en los hogares y los motores de combustión interna de los automóviles son ejemplos de esta situación. Además de estos, existen gran cantidad de ejemplos donde este tipo de sistemas encuentran aplicaciones, tanto en ámbitos domésticos como industriales. Es principalmente este gran espectro de aplicaciones, lo que convierte a estos sistemas en elementos primordiales para el cotidiano desarrollo de la industria y la sociedad. Dicho de otra forma, sin ellos la vida del ser humano no sería tan cómoda o tan productiva como lo es hoy en día.

Es evidente entonces, que la importancia en el estudio de este tipo de sistemas, como de los ciclos termodinámicos que constituyen los principios fundamentales de operación de estos, reside en sus múltiples aplicaciones. Pese a esta utilidad, los sistemas termodinámicos se caracterizan por operar con baja eficiencia energética, encontrándose esta alrededor del 40% para sistemas en general¹.

Lo anteriormente expuesto, pretende dejar en evidencia la importancia que la eficiencia tiene en estos sistemas, que desperdician o pierden alrededor del 60% de la energía que se les suministra. Como prescindir total o parcialmente de ellos es en muchos casos imposible y en otros poco viable, se hace necesario desarrollar medios para aumentar la eficiencia en estos, buscando así reducir la energía desperdiciada.

Es gracias a esto, que se toma la iniciativa de desarrollar el proyecto en esta área, buscando aportar en la búsqueda por sistemas termodinámicos más eficientes y tratando de sentar un precedente en la Facultad de Ingeniería Mecánica para futuros proyectos con objetivos similares o en la misma área.

Además de desarrollar el proyecto en esta área, se tiene conciencia de la gran cantidad de servicios que prestan las ciencias de la computación y la informática al resto de disciplinas, entre ellas incluida la ingeniería mecánica. Lo anterior sumado al gusto personal por la computación, impulsa el deseo de combinar ambas áreas del conocimiento en este proyecto, desarrollando un software que beneficie a la ingeniería mecánica en el campo de la termodinámica.

¹BORGNAKKE, Claus y SONNTAG, Richard E. Fundamentals of Thermodynamics. 8.^a ed. John Wiley & Sons Inc, 2013.

1. PLANTEAMIENTO DEL PROBLEMA

Dada la importancia de los sistemas termodinámicos cíclicos para el desarrollo de la vida del ser humano y teniendo en cuenta que la optimización de estos genera de igual manera optimización en la vida misma de este, es evidente la necesidad de impulsar la investigación y el desarrollo en esta área, buscando producir sistemas más eficientes.

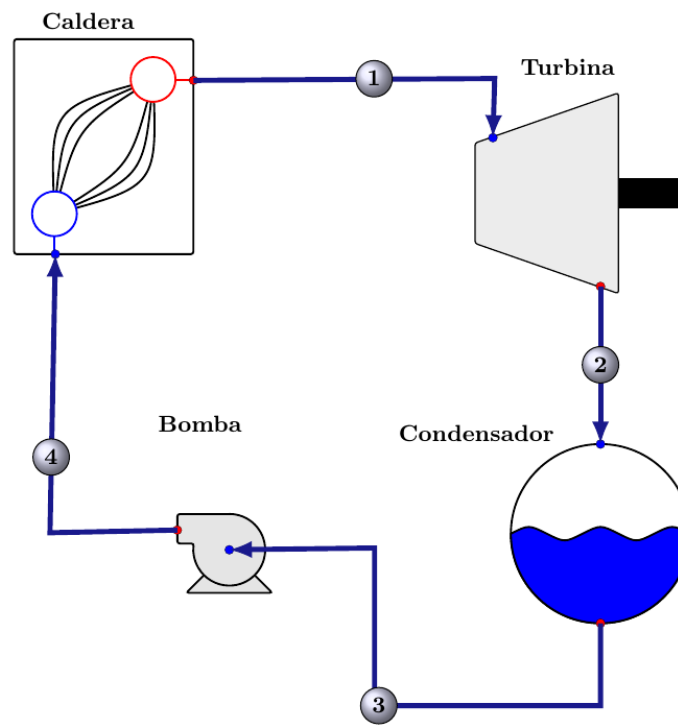
Cabe señalar que sin importar el componente específico a optimizar, todo proceso de optimización parte inicialmente del análisis y estudio del sistema como un todo y de la interacción entre sus componentes. Si bien estos sistemas al estar constituidos por un número reducido de componentes (entre cuatro y ocho componentes aproximadamente) su análisis resulta matemáticamente sencillo y temporalmente aceptable sin la necesidad de utilizar herramientas computacionales, al aumentar el número de componentes, el análisis será cada vez mas complejo y demandará mas tiempo, esto sin tener en cuenta el número de iteraciones que en ocasiones son necesarias para estudiar correctamente un sistema.

Como respuesta al problema anterior, específicamente en el proceso inicial de análisis y estudio del sistema como un todo y la interacción entre sus componentes, se llevará a cabo el desarrollo de un software de simulación de sistemas termodinámicos cíclicos de potencia y refrigeración.

2. JUSTIFICACIÓN

Como se había mencionado con anterioridad, el análisis de sistemas térmicos con un número reducido de componentes es matemáticamente sencillo. La figura 1 muestra un sistema térmico que opera bajo ciclo Rankine y se encuentra formado por cuatro componentes, una caldera, una bomba centrífuga, una turbina y un condensador. Este sistema se encuentra caracterizado termodinámicamente por cuatro ecuaciones, una por cada componente, siendo estas función de las entalpías del ciclo.

Figura 1: Sistema térmico simple.

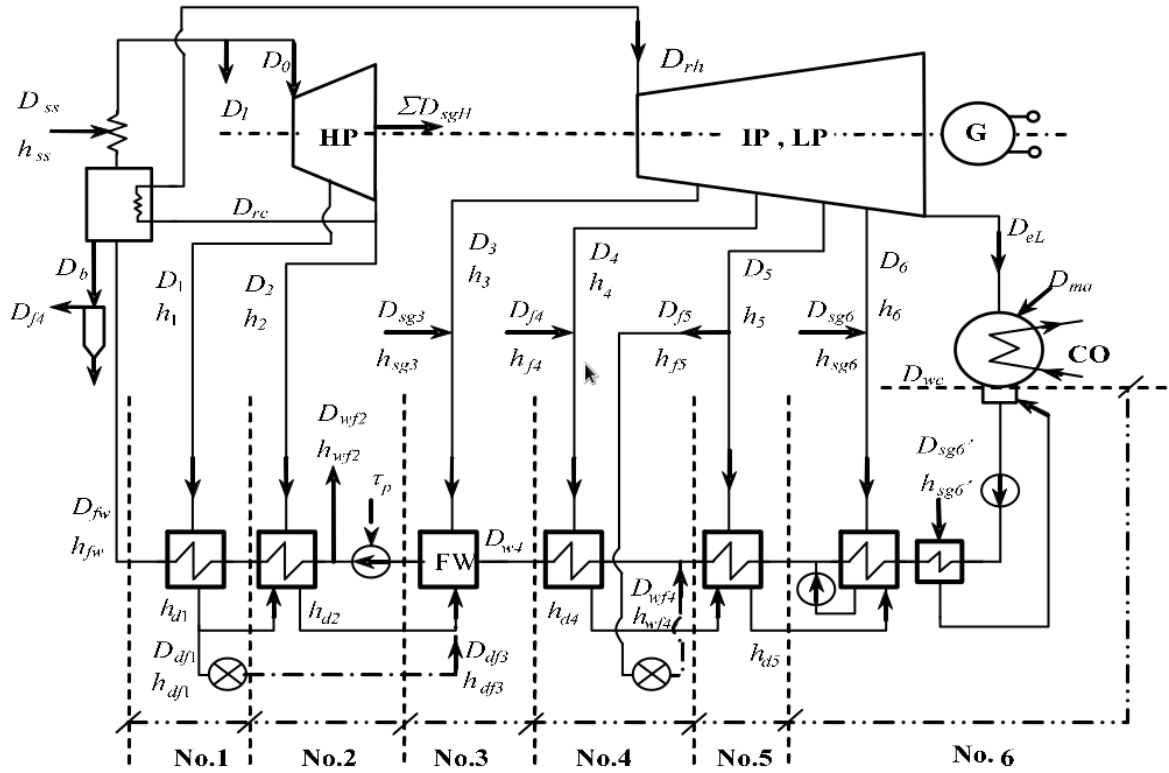


Fuente: El autor

A pesar de la aparente sencillez en el estudio de sistemas similares al presentado en la figura 1, el estudio de estos crece en complejidad al mismo tiempo en que el sistema mismo crece en número de componentes. Lo anterior se puede evidenciar con el esquema enseñado en la figura 2, el cual representa un sistema térmico típico de una aplicación real y se encuentra formado por más de veinte componentes.

La situación anterior es el principal estímulo para el desarrollo de este proyecto, puesto que al acelerar el proceso de cálculo de sistemas termodinámicos cíclicos se reduce el tiempo de estudio de estos o se aumenta en número de iteraciones posibles, permitiendo analizar sistemas de mayor complejidad.

Figura 2: Sistema térmico real.



Fuente: Liping Li (2012). Process Performance Monitoring and Degradation Analysis, Thermal Power Plants, Dr. Mohammad Rasul (Ed.), ISBN: 978-953-307-952-3, InTech, DOI: 10.5772/27335. Disponible en: <http://www.intechopen.com/books/thermal-power-plants/process-performance-monitoring-and-degradation-analysis>

3. OBJETIVOS

3.1. OBJETIVO GENERAL

Desarrollar un software de simulación de sistemas termodinámicos cíclicos de potencia y refrigeración para el sistema operativo Windows[®] y para distribuciones Linux[®] basadas en el sistema operativo Debian[®] .

3.2. OBJETIVOS ESPECÍFICOS

1. Determinar los algoritmos para la caracterización del comportamiento termodinámico de los dispositivos que componen los sistemas.

Resultado: Diagramas de flujo de los principales procesos a implementar en el software.

Indicador de calidad: Diagramas de flujo de los principales procesos a implementar en el software según el estándar de simbología ISO 5807.

2. Definir la estructura y la interacción entre los componentes del software.

Resultado: Diagrama de clases del software en lenguaje unificado de modelamiento (UML).

Indicador de calidad: Diagrama de clases en lenguaje unificado de modelamiento (UML) según el estándar para UML 2 del Object Management Group (OMG).

3. Describir el funcionamiento y la operación del software.

Resultado: Documentación técnica y manual de usuario del software. Indicador de calidad: Guía de usuario con las instrucciones necesarias para la manipulación del software y documentación técnica detallando las características técnicas de este.

4. MARCO TEÓRICO

A continuación se presentarán el estado del arte y los conceptos teóricos más importantes, relativos a la temática de este proyecto.

4.1. ESTADO DEL ARTE

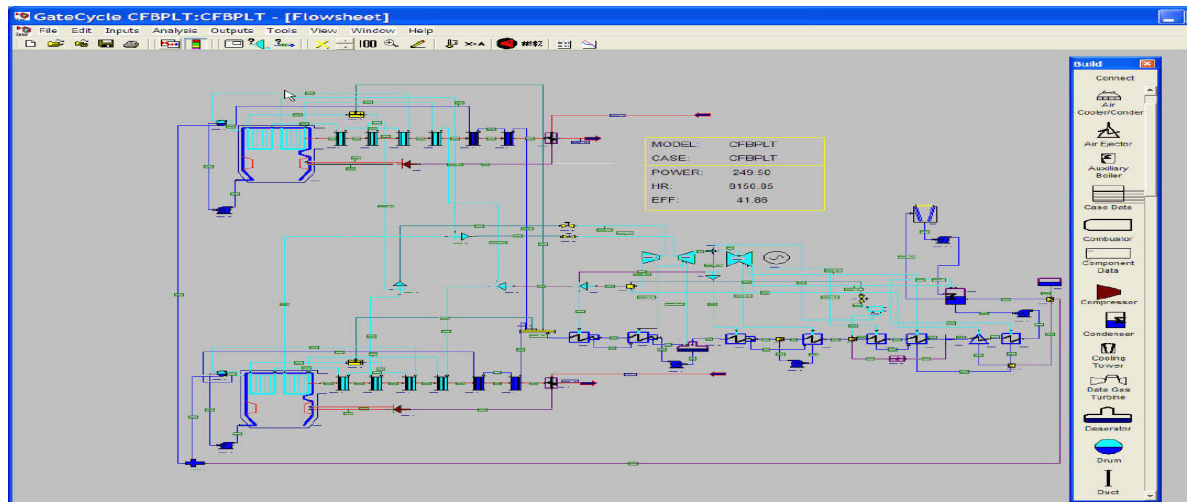
Se comienza por presentar y explicar brevemente las características de cuatro paquetes informáticos de vanguardia, en el modelamiento y simulación de sistemas térmicos

4.1.1. Software de simulación de sistemas de conversión de energía

4.1.1.1. GateCycle™

GateCycle™ es un software de monitoreo de plantas desarrollado por General Electric™ que predice el rendimiento de sistemas energéticos, tanto en etapa diseño como fuera de esta. Es utilizado para evaluación rápida, ingeniería de detalle, diseño, reequipamiento, repotenciación, y pruebas de aceptación. Gracias a un enfoque componente por componente y funciones de macro avanzadas, permite modelar prácticamente cualquier tipo de sistema¹. En la figura 3 se presenta una captura de pantalla de GateCycle™.

Figura 3: Captura de pantalla de GateCycle™



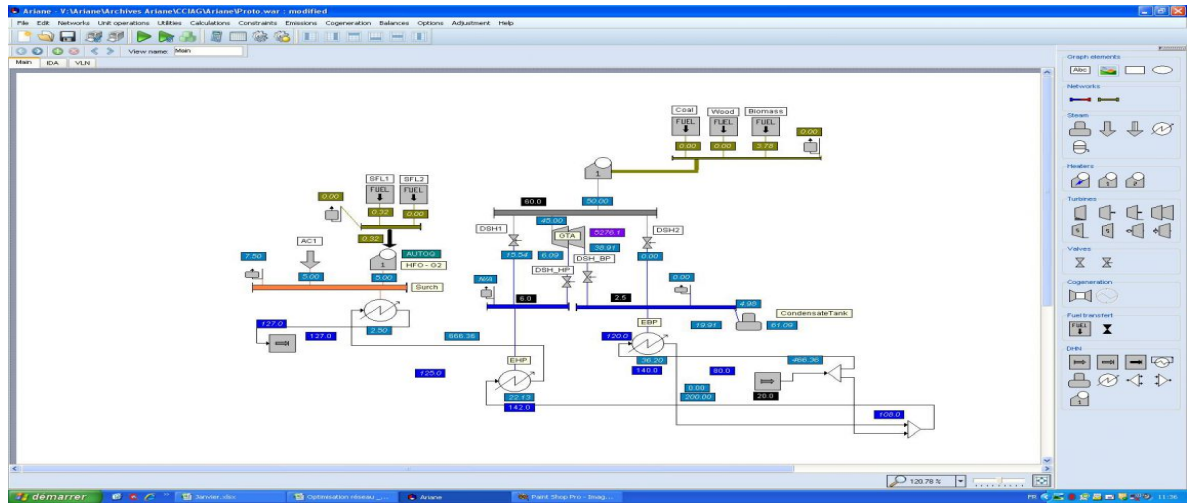
Fuente: <http://www.wyattllc.com/GateCycle/GateCycle.jpg>

¹GATECYCLE®. Software. GE Enter Software LLC. 1490 Drew Avenue, Suite 180, Davis, California 95616.

4.1.1.2. Ariane™

Ariane™ es una herramienta computacional de gerencia, modelamiento y optimización, diseñada específicamente para plantas de producción de energía (vapor, electricidad, agua caliente, etc), desarrollada por ProSim™ . Permite la configuración de cada equipo de la planta, teniendo en cuenta sus restricciones técnicas, su modo de operación, características de diseño, y costos de mantenimiento. Además, se utilizan modelos no lineales que hacen posible una representación realista de la evolución del rendimiento y de los balances de entalpía, bajo cualquier carga que se encuentre la planta². En la figura 4 se presenta una captura de pantalla de este software.

Figura 4: Captura de pantalla de Ariane™



Fuente: http://www.prosim.net/bibliotheque/Image/logiciels-softwares/Captures-Ariane/Image3_1150x719.jpg

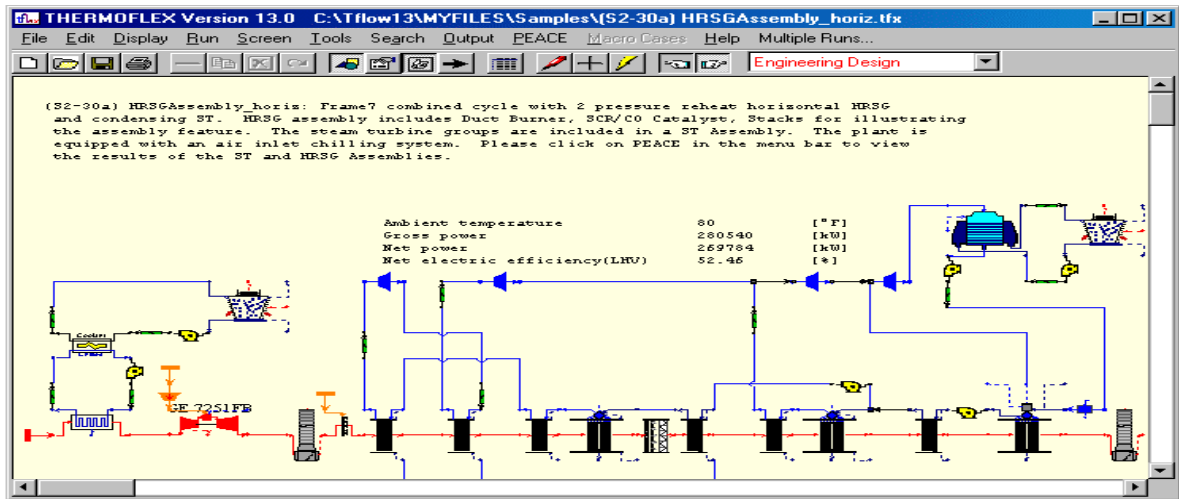
4.1.1.3. THERMOFLEX™

THERMOFLEX™ es un software desarrollado por Thermoflow™ para modelamiento de sistemas térmicos, con énfasis en generación de energía y cogeneración. Cuenta con modos para diseño y fuera de diseño, además de permitir el modelamiento de una gran variedad de sistemas de centrales eléctricas. Adicional a esto, cuenta con más de 180 modelos de componentes que ya han sido probados en aplicaciones específicas en la industria³. En la figura 5 se presenta una captura de pantalla de este software.

²ARIANE®. software. ProSim PLC. URL: <http://www.prosim.net/en/software-ariane-10.php>.

³THERMOFLEX™. Software. Thermoflow Inc. URL: http://www.thermoflow.com/products_fullyflexible.html#.

Figura 5: Captura de pantalla de THERMOFLEX™



Fuente: http://www.thermoflow.com/Images/tfx_001.gif

4.1.1.4. Cycle-Tempo™

Cycle-Tempo™ desarrollado por Asimptote™ para el análisis termodinámico y la optimización de sistemas de conversión de energía. Es apto para centrales eléctricas convencionales, sistemas de enfriamiento y refrigeración por compresión y sistemas energéticos no convencionales como las centrales eléctricas solares de ORC, centrales eléctricas de ciclo kalina, etc. Su característica principal es el cálculo de todas las masas y los flujos de energía relevantes en el sistema, y además, permite análisis de exergía⁴. En la figura 6 se presenta una captura de pantalla de este software.

4.1.2. Cálculo de propiedades termodinámicas

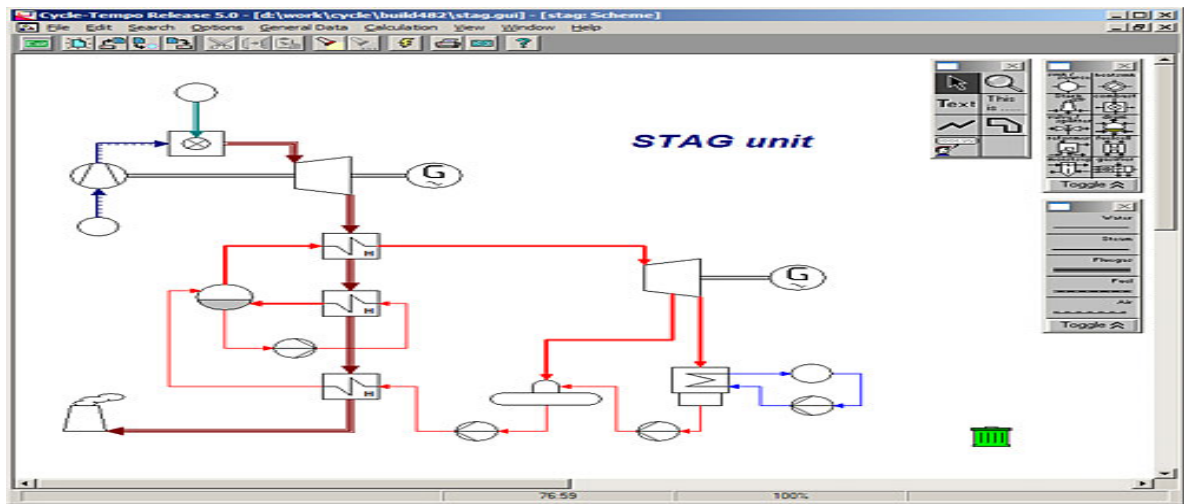
Ya presentados los diferentes programas de modelamiento y simulación de sistemas térmicos, se enuncian y comentan a continuación, dos programas y una librería informática para el cálculo de propiedades termodinámicas de fluidos.

4.1.2.1. Simulis Thermodynamics®

Simulis® Thermodynamics es un servidor de cálculo termofísico que genera con alta precisión propiedades de sustancias puras y mezclas y equilibrio de fases de fluido. La versión standard de Simulis® Thermodynamics es suministrada como un complemento de Microsoft® Excel o como un toolbox en MATLAB® y permite al usuario correr cálculos termodinámicos completos en estas aplicaciones, pero también puede

⁴CYCLE-TEMPO. Software. ASIMPTOTE. URL: <http://www.asimptote.nl/software/cycle-tempo>.

Figura 6: Captura de pantalla de Cycle-Tempo™



Fuente: http://img.informer.com/screenshots/196/196311_1.jpg

ser conectar con cualquier código heredado⁵.

4.1.2.2. REFPROP™

REFPROP™ es un acrónimo para REFerence fluid PROPERTIES. Este programa, desarrollado por el Instituto Nacional de Estándares y Tecnología (NIST por sus siglas en inglés), calcula las propiedades termodinámicas y de transporte de fluidos importantes para la industria y sus mezclas. Estas propiedades pueden ser presentadas en tablas y gráficos a través de la interfaz gráfica de usuario; son también accesibles a través de hojas de cálculo o aplicaciones escritas por el usuario accediendo el REFPROP.dll.

REFPROP™ esta basado en los modelos de fluidos puros y mezclas mas precisos actualmente disponibles. Implementa tres modelos para las propiedades termodinámicas de fluidos puros: ecuaciones de estado explicitas en la energía de Helmholtz, la ecuación de estado modificada de Benedict-Webb-Rubin, y un modelo extendido de estados correspondientes (ECS). Los cálculos de mezclas emplean un modelo que aplica reglas de mezcla a la energía de Helmholtz de componentes de la mezcla; utiliza una función de partida para tener en cuenta la partida de mezclas ideales. La viscosidad y la conductividad térmica son modeladas ya sea con correlaciones especificas de fluidos, un método ECS, o en algunos casos el método de fricción teórica⁶.

⁵BAUDOIN, Olivier, VACHER, Alain y D'ECHELOTTE, Stéphane. Simulis® Thermodynamics: an open framework for users and developers. Visitado: 04-03-2015. ProSim SA. URL: http://www.prosim.net/bibliotheque/File/Publications/Events/32_ESAT_E_2008-05_SimulisThermodynamics_anopenframework%20forusersanddevelopers.pdf.

⁶LEMMON, Eric W., HUBER, M. L. y MCLINDEN, M. O. NIST Standard Reference Database

4.1.2.3. CoolProp™

CoolProp™ es una librería de código abierto para el cálculo de propiedades termofísicas escrita en C++. Cuenta con interfaces disponibles para la mayoría de lenguajes de programación y plataformas de interés técnico. La librería incluye 110 fluidos puros y pseudo-puros, como también propiedades de 40 fluidos incompresibles y aire húmedo.

Las propiedades termodinámicas más precisas son obtenidas a partir de las formulaciones multiparámetro de tipo explícitas de energía de Helmholtz. Para las propiedades de transporte, se emplean una amplia gama de métodos, incluyendo el método extendido de estados correspondientes⁷.

4.2. SISTEMAS

Tener una descripción precisa del objeto de estudio es primordial en todo análisis de ingeniería. En termodinámica, un *sistema* es el objeto de estudio y puede ser tan simple como un cuerpo libre o tan complejo como una refinería química completa. Una vez se ha definido el sistema y se identifican las interacciones relevantes con otros sistemas, se aplican leyes físicas o relaciones al análisis⁸.

Las fronteras y los alrededores del sistema, son dos elementos a tener siempre en cuenta para todo análisis termodinámico. Los alrededores es el ambiente que envuelve el sistema, todo aquello que es externo a este. El o los elementos que permiten distinguir entre un sistema y sus alrededores se les denomina fronteras y pueden estar en reposo o en movimiento. Dependiendo de la interacción con sus alrededores y el comportamiento de sus fronteras, un sistema puede ser analizado como cerrado o como volumen de control.

4.2.1. Sistemas cerrados

Se considera a un sistema cerrado si sus fronteras no permiten el tránsito de masa a través de ellas. Este tipo de sistema, puede interactuar con su exterior mediante el intercambio de energía. El sistema cilindro-pistón que se esquematiza en la figura 7, es un ejemplo de sistema cerrado, ya que al no haber intercambio de masa con los alrededores del sistema, la cantidad de esta permanece constante en todo momento.

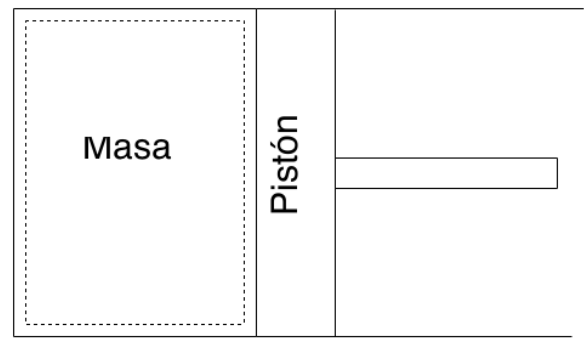
Un tipo especial de sistema cerrado que no interactúa con sus alrededores en ninguna forma se le denomina sistema aislado. En el sistema cerrado de la figura 7, no

23: Reference Fluid Thermodynamic and Transport Properties - REFPROP. 9.0. National Institute of Standards and Technology, Standard Reference Data Program. Gaithersburg, 2010. URL: <http://www.nist.gov/srd/nist23.cfm>.

⁷BELL, Ian H. y col. Pure and Pseudo-pure Fluid Thermophysical Property Evaluation and the Open-Source Thermophysical Property Library CoolProp. En: *Industrial & Engineering Chemistry Research* 53.6 (2014), págs. 2498-2508. DOI: 10.1021/ie4033999. eprint: <http://pubs.acs.org/doi/pdf/10.1021/ie4033999>. URL: <http://pubs.acs.org/doi/abs/10.1021/ie4033999>.

⁸MORAN, Michael J. y col. *Fundamentals of Engineering Thermodynamics*. 7.^a ed. John Wiley & Sons Inc, 2011, p. 4-5.

Figura 7: Esquema de un sistema cerrado



Fuente: El autor.

se evidencia ningún tipo de intercambio del sistema con sus alrededores, por lo que en últimas se puede catalogar a este como aislado.

4.2.2. Volúmenes de control

En el volumen de control al contrario del sistema cerrado, la masa puede fluir a través de sus fronteras. La figura 8 presenta un sistema cilindro-pistón que permite el flujo de masa tanto hacia dentro como hacia fuera de este, por lo que se considera un volumen de control. Cabe agregar que la mayoría de sistemas en ingeniería son analizados como volumen de control.

4.3. ENERGÍA

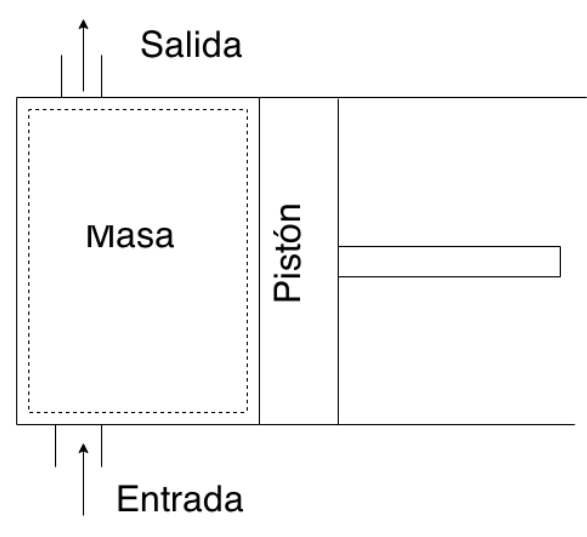
Si bien existen diferentes tipos de energía, el trabajo y el calor son comúnmente empleadas en la caracterización de los sistemas termodinámicos.

4.3.1. Trabajo y potencia

La mecánica clásica define el trabajo como el producto de una fuerza por la distancia que esta recorre. Este concepto puede ser aplicado también a la termodinámica; si el pistón del sistema presentando en la figura 7 se desplazase a causa de una presión aplicada sobre este, el trabajo resultante del cambio de posición sería igual al producto de la presión aplicada sobre el émbolo por la distancia recorrida por este. Cabe aclarar, que el trabajo es una cantidad transitoria que solo se presenta mientras el sistema sufre un cambio de estado, siempre y cuando este involucre el desplazamiento de alguna frontera⁹. La potencia no es otra cosa que el trabajo por unidad de tiempo.

⁹RAJPUT, R. K. Engineering Thermodynamics. 3.^a ed. Laxmi Publications LTD, 2007, P. 46.

Figura 8: Esquema de un sistema abierto



Fuente: El autor.

4.3.2. Calor

Al igual que el trabajo, el calor es una cantidad transitoria, ya que es solo se evidencia durante la transición entre un estado y otro. La principal diferencia entre el trabajo y el calor, radica en aquello que los produce; mientras que el trabajo es producto del desplazamiento de una frontera, el calor el producto de la diferencia de temperatura entre el sistema y sus alrededores.

4.4. PRIMERA LEY DE LA TERMODINÁMICA

El concepto de energía y la hipótesis que esta no puede ser creada ni destruida, es fruto de las investigaciones científicas del siglo XVIII. A esta hipótesis se le conoce como la ley de conservación de la energía. La primera ley de la termodinámica es una declaración de este mismo principio con referencia particular a la energía térmica y la energía mecánica.

La primera ley de la termodinámica enuncia:

Quando un sistema experimenta un ciclo termodinámico, el trabajo neto suministrado a este desde los alrededores es igual al trabajo neto realizado por el sistema en sus alrededores¹⁰.

La primera ley de la termodinámica puede ser expresada matemáticamente con la ecua-

¹⁰RAJPUT, Engineering Thermodynamics, p. 101-102.

ción 4.1, donde \oint representa la suma para un ciclo completo.

$$\oint dQ = \oint dW \quad (4.1)$$

A pesar de que la primera ley de la termodinámica no puede ser probada analíticamente, evidencia experimental ha confirmado su validez. Además, ante la falta de fenómenos que la contradigan, ha sido aceptada como una ley de la naturaleza. Cabe anotar que no se impusieron restricciones que limitaran la aplicación de esta ley a transformaciones reversibles de energía, por lo que puede ser también aplicada a transformaciones irreversibles. Sin embargo, para procesos no cíclicos se necesita una formulación más general de la primera ley de la termodinámica, formulación que involucra el concepto de energía interna¹¹.

4.4.1. Energía interna

“La energía interna se define como la suma de todas las formas microscópicas de energía en un sistema. Se relaciona con la estructura molecular y el grado de actividad molecular y se puede considerar como la suma de las energías cinética y potencial de las moléculas¹²”.

A escala microscópica existen cuatro formas de energía que pueden aportar a la energía interna, ellas son:

- Energía latente. Se relaciona con la fase de un sistema.
- Energía sensible. Se relaciona directamente con la energía cinética de la molécula.
- Energía química. Se relaciona con los enlaces atómicos de las moléculas.
- Energía nuclear. Se relaciona con los fuertes enlaces dentro del núcleo del átomo.

En general para aplicaciones reales, solo se tienen en cuenta las energías en forma latente y sensible¹³.

4.4.2. Entalpía

Para el análisis de volúmenes de control resulta insuficiente tener en cuenta solo la energía interna, debido al flujo de masa a través de las fronteras del sistema. Es debido a esto, que es frecuente para estos casos la combinación de energía interna, presión y volumen. A esta combinación de propiedades se le conoce como entalpía¹⁴.

$$h = u + Pv \quad (4.2)$$

$$H = U + PV \quad (4.3)$$

¹¹RAJPUT, Engineering Thermodynamics, p. 102.

¹²ÇENGEL, Yunus A. y BOLES, Michael A. Termodinámica. 7.ª ed. McGRAW-HILL, 2012, p. 55.

¹³Ibíd., p. 55.

¹⁴Ibíd., p. 126 - 127.

Las ecuaciones 4.2 y 4.3, representan de forma matemática la entalpía específica h y la entalpía total H .

4.5. VOLUMENES DE CONTROL EN ESTADO ESTABLE

Se dice que un volumen de control se encuentra en estado estable, cuando tanto el flujo másico a través de sus fronteras como las funciones de estado (presión, temperatura, energía interna, etc) se mantienen constantes con respecto al tiempo. Las ecuaciones 4.4 y 4.5 son respectivamente el balance de masa y el balance de energía para un volumen de control en estado estable¹⁵.

$$\sum \dot{m}_e h_e = \sum \dot{m}_s h_s \quad (4.4)$$

$$\dot{Q}_{v.c.} + \sum \dot{m}_e h_e = \dot{W}_{v.c.} + \sum \dot{m}_s h_s \quad (4.5)$$

4.5.1. Ejemplos de Volúmenes de control en estado estable

A continuación se presentaran y explicaran, diferentes ejemplos de sistemas estudiados como volúmenes de control, cuya operación es en estado estable.

4.5.1.1. Intercambiadores de calor

Intercambiador de calor es el nombre general dado a cualquier dispositivo cuya función principal sea fomentar el transporte de energía térmica entre dos fluidos, sin que estos dos se mezclen. Tienen diferentes características dependiendo de su aplicación y los fluidos que intercambiarán calor en este. Pueden ser modelados como un par de sistemas en estado estable con tasas de transferencia de calor iguales y opuestas, o pueden ser modelados como un sistema en estado estable con dos entradas y dos salidas que no intercambia calor con su entorno¹⁶.

En la figura 9 es posible apreciar las características anteriormente mencionadas de un intercambiador de calor, donde dos fluidos intercambian energía sin entrar en contacto físico.

4.5.1.2. Difusores y toberas

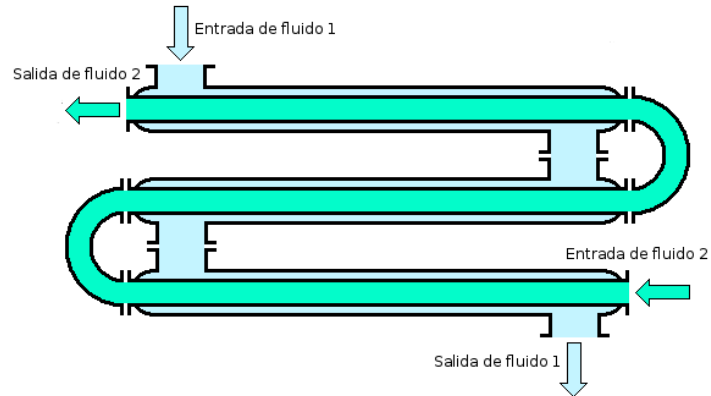
Los difusores son dispositivos utilizados para convertir parte de la energía cinética de un flujo en energía de presión. Por otro lado, las toberas tiene la función totalmente contraria, ya que convierten parte de la energía de presión de un flujo en energía cinética. Dicho en otras palabras, los difusores disminuyen la velocidad del flujo que los atraviesa, mientras que las toberas la aumentan. Cabe agregar, que al modelar toberas y difusores se suelen asumir adiabáticas. Adicional a esto, su simplicidad mecánica no les permite realizar o absorber trabajo¹⁷.

¹⁵BORGNAKKE y SONNTAG, Fundamentals of Thermodynamics, p. 165-167.

¹⁶BALMER, Robert T. Modern Engineering Thermodynamics. Elsevier Inc, 2011, p. 184.

¹⁷Ibíd., p. 174 - 175.

Figura 9: Esquema de un intercambiador de calor



Fuente: Por Turbojet (Trabajo Propio), usado bajo CC BY-SA 4.0-3.0-2.5-2.0-1.0 (<http://creativecommons.org/licenses/by-sa/4.0-3.0-2.5-2.0-1.0>) / modificado a partir del original.

Figura 10: Esquemas representativos de un difusor y una tobera



(a) Esquema de un difusor

(b) Esquema de una tobera

Fuente: El autor

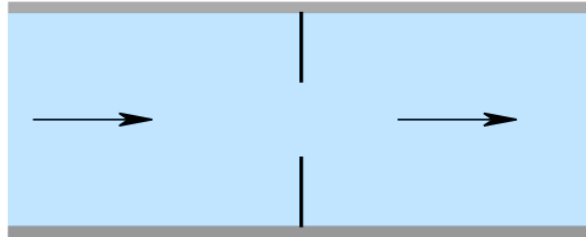
Tal como se evidencia en la figura 10a, un difusor en su forma más simple, es para el flujo incompresible un ducto divergente. De la misma manera, la figura 10b muestra como en su forma más elemental, una tobera para el flujo incompresible no es más que un ducto convergente.

4.5.1.3. Estranguladores

Un proceso de estrangulamiento se produce, cuando un fluido circulando por una línea encuentra una restricción a su flujo. Esta restricción puede ser un plato con un círculo en él como se esquematiza en la figura 11, un cambio drástico de diámetro entre dos tuberías o una válvula de estrangulamiento. Ya que el estrangulamiento es un proceso considerablemente más agresivo que el realizado por una tobera, los cambios

en la energía cinética realizados por este pueden ser despreciados. De la misma manera, la transferencia de calor por efecto del estrangulamiento también es despreciada, por lo que la presión es la única variación que produce el estrangulador en el flujo, mientras que la entalpía se mantiene constante. De esto se concluye entonces, que un proceso de estrangulamiento en estado estable, es aproximadamente una disminución en la presión del flujo manteniendo la entalpía constante¹⁸.

Figura 11: Diagrama ejemplo de un estrangulador



Fuente: El autor

4.5.1.4. Turbinas

Son máquinas rotativas que producen trabajo a partir de la presión de un fluido que pasa a través de ellas. Dos clases generales son la turbina de vapor, donde el fluido de trabajo que sale de la turbina pasa a un condensador, y la turbina de gas, donde los gases de escape de esta son liberados a la atmósfera. Para ambas clases el principio de funcionamiento consta de las mismas dos fases. En la primera fase, el fluido de trabajo a alta presión pasa por una tobera, donde se disminuye la presión del flujo y se aumenta su velocidad. En la segunda fase, el flujo a alta velocidad se dirige hacia los alabes de la turbina, donde la velocidad de este es reducida antes de ser expulsado. Esta pérdida de velocidad en el flujo produce torque en el eje rotativo de la turbina, lo que resulta en una salida de trabajo¹⁹.

La figura 12 presenta el ensamblaje de una turbina de vapor, donde se aprecian los alabes de esta junto con parte de su carcasa.

4.5.1.5. Bombas y compresores

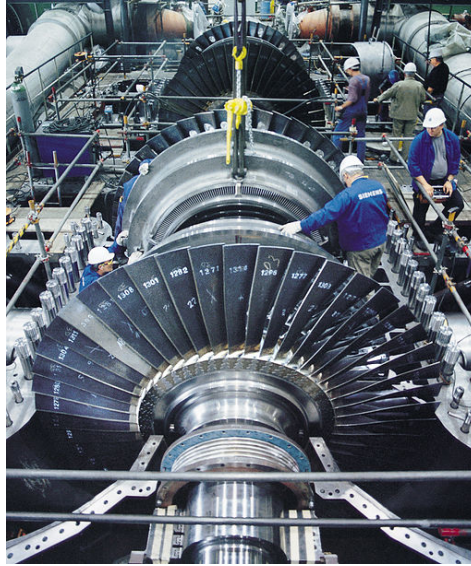
A pesar de ser dos máquinas diferentes, las bombas y los compresores comparten la misma función, aumentar la presión de un fluido imprimiéndole trabajo de eje. La diferencia principal entre ambas máquinas es el fluido, ya que los compresores se utilizan para aumentar la presión de los gases y las bombas de los líquidos. En ambos casos son generalmente modelados como adiabáticos²⁰.

¹⁸BORGNAKKE y SONNTAG, Fundamentals of Thermodynamics, p. 172 - 173.

¹⁹Ibíd., p. 173.

²⁰Ibíd., p. 174.

Figura 12: Ensamblaje de una turbina de vapor



Fuente: Por Siemens Pressebild CC-BY-SA-3.0 (<http://creativecommons.org/licenses/by-sa/3.0/>), vía Wikimedia Commons

4.5.1.6. Cámaras de mezclado

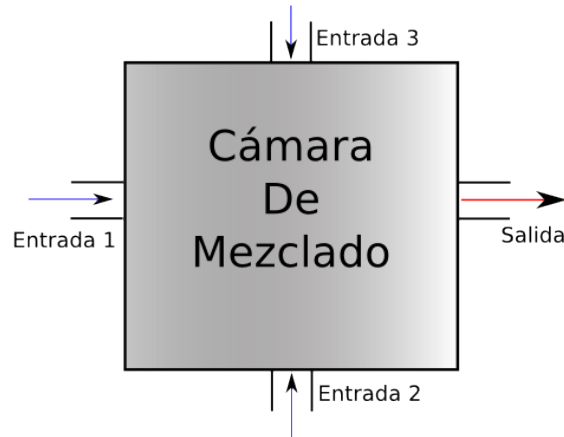
Se conoce como cámara de mezclado el espacio o la sección donde tiene lugar el proceso de mezclado de dos o más flujos. Si bien su nombre lo indica, no necesariamente tienen que ser una cámara, como por ejemplo, la tubería en forma de T de una regadera, donde se mezclan un flujo de agua caliente con uno frío. Una cámara de mezclado puede tener desde dos entradas de flujo hasta las necesarias para una aplicación particular, como por ejemplo, la cámara de mezclado esquematizada en la figura 13 que cuenta con tres entradas. A pesar de no tener restricciones en el número de entradas, la cámara de mezclado opera a una presión constante, la misma a la que los flujos debe entrar y salir de esta²¹.

Cabe agregar, que el principio de conservación de la masa requiere que la cantidad de fluido que entra sea la misma que salga, dicho de otra forma, el flujo másico a la salida de la cámara de mezclado, debe ser igual a la suma de los flujos másicos de las entradas, tal como lo muestra la ecuación 4.6

$$\dot{m}_{salida} = \sum \dot{m}_{entradas} \quad (4.6)$$

²¹ÇENGEL y BOLES, Termodinámica, p. 240.

Figura 13: Diagrama de una cámara de mezclado



Fuente: El autor

4.6. SEGUNDA LEY DE LA TERMODINÁMICA

Como se mencionó anteriormente, la primera ley de la termodinámica no impone restricciones en cuanto a la dirección de un procesos se refiere, dicho en otras palabras, permite cualquier tipo de irreversibilidades en los procesos. Sin embargo, la experimentación ha demostrado que si bien un proceso cumple con la primera ley de la termodinámica, no necesariamente es posible en la realidad. Es gracias a lo anterior que se necesita de la segunda ley de la termodinámica para establecer si un proceso es posible en la realidad o no.

A causa de la naturaleza experimental de la segunda ley de la termodinámica, no existe para esta un enunciado definitivo. Dos de los enunciados más tradicionales son el de Clausius y el de Kelvin-Planck, sin embargo el enunciado de entropía es la forma más efectiva de la segunda ley de la termodinámica para un amplio abanico de aplicaciones de ingeniería²².

El enunciado de Clausius afirma que:

Es imposible para cualquier sistema operar de manera tal, que el único resultado sea una transferencia de energía como calor desde un cuerpo más frío a uno más caliente²³.

Dicho en otras palabras, el calor no puede espontáneamente circular de un elemento frío a uno caliente sin que se suministre trabajo al sistema. Un ejemplo de esta situación

²²MORAN y col., Fundamentals of Engineering Thermodynamics, p. 238.

²³Ibíd., p. 239.

son los sistemas de refrigeración, en los cuales existe una transferencia de calor de un cuerpo frío a uno caliente, gracias a un motor eléctrico al que se le suministra energía proveniente de los alrededores del sistema.

El enunciado de Kelvin-Planck afirma que:

Es imposible para cualquier sistema operar en un ciclo termodinámico y entregar una cantidad neta de trabajo a sus alrededores, mientras recibe energía por medio de transferencia de calor de un único reservorio térmico²⁴.

El enunciado de Kelvin-Planck introduce el término reservorio a la definición de la segunda ley de la termodinámica. Un reservorio es un tipo especial de sistema que se mantiene a temperatura constante. Cabe aclarar que un reservorio es una idealización, pero puede ser aproximando si el calor suministrado a este proviene del sistema, no es suficiente para cambiar su temperatura. Es debido a esto que en la mayoría de los casos, el reservorio posee un tamaño considerablemente superior al del sistema. Por ejemplo, para un sistema de refrigeración, un motor de combustión interna o una planta eléctrica el reservorio puede ser la atmósfera de la tierra o un gran cuerpo de agua.

El enunciado de entropía afirma que:

Es imposible para cualquier sistema operar de forma tal que destruya entropía²⁵.

De la misma manera en la que se tienen en cuenta los balances de masa y de energía en el análisis de un sistema, se tienen en cuenta también los balances de entropía.

$$\Delta S = S_i + S_{gen} \quad (4.7)$$

La ecuación 4.7 presenta de manera simplificada un balance de entropía, donde ΔS es el cambio en la cantidad de entropía en un sistema durante un intervalo de tiempo, S_i es la cantidad neta de entropía transferida al sistema a través de las fronteras en el mismo intervalo de tiempo y S_{gen} es la cantidad de entropía producida por el sistema también en el mismo intervalo de tiempo.

A diferencia de la masa y la energía que se conservan en un sistema, la entropía es producida cuando existe irreversibilidades como la fricción, en un sistema. Es por esto que el término S_{gen} en la ecuación 4.7 puede ser positivo o cero pero no negativo. En consecuencia, la producción de entropía es un claro indicador de la posibilidad de un proceso en la realidad.

²⁴MORAN y col., Fundamentals of Engineering Thermodynamics, p. 240.

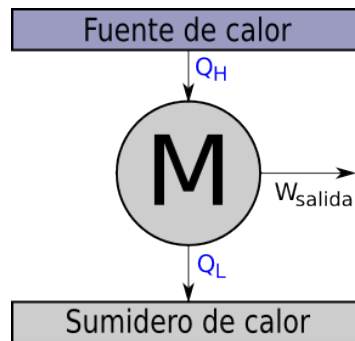
²⁵Ibíd., p. 242.

4.7. CICLOS DE POTENCIA Y REFRIGERACIÓN

4.7.1. Máquina térmica

Las máquinas creadas a lo largo del tiempo, han sido clasificados tradicionalmente de forma general según su fuente de energía y de forma específica según su función. Partiendo de esto, es posible definir un motor térmico como una máquina que produce trabajo a partir de una fuente de energía calórica (fuego, vapor, solar, etc)²⁶.

Figura 14: Esquema de una máquina térmica



Fuente: El autor

Como se dijo con anterioridad, para que una máquina térmica pueda producir trabajo, según la segunda ley de la termodinámica, esta debe operar entre una fuente de energía y un sumidero. El diagrama de una máquina térmica presentado en la figura 14, esquematiza esta restricción impuesta a los sistemas térmicos por parte de la segunda ley de la termodinámica.

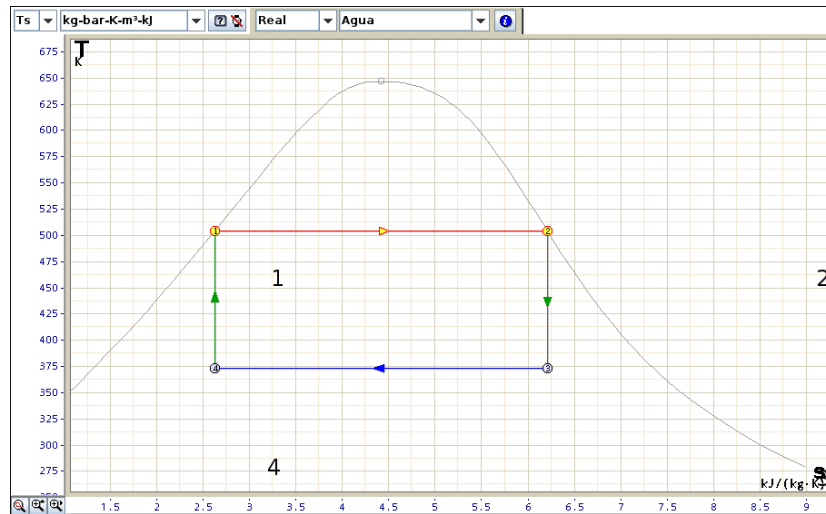
4.7.2. Ciclo Carnot

Es el ciclo termodinámico idealmente más eficiente de todos aquellos que operan entre dos límites específicos de temperatura. El ciclo de Carnot se compone de cuatro procesos termodinámicos, como se observa en el diagrama T-s del ciclo de Carnot mostrado en la figura 15.

A pesar de ser idealmente el ciclo más eficiente, se asocian algunas situaciones imprácticas estos cuatro procesos del ciclo de Carnot en la realidad. El proceso 1-2 es una transferencia isotérmica de calor hacia el fluido de trabajo. El proceso 2-3 es una expansión isentrópica que puede ser aproximado con una turbina bien diseñada, sin embargo según el proceso de diagrama, la turbina deberá operar con vapor de baja calidad, lo que produce desgaste en sus alabes. El proceso 3-4 es una transferencia de calor isotérmica desde el fluido de trabajo. El proceso 4-1 es una compresión isentrópica

²⁶BALMER, Modern Engineering Thermodynamics, p. 449.

Figura 15: Diagrama T-s del ciclo de Carnot



Fuente: El autor

que involucra comprimir una mezcla de líquido y vapor hasta un líquido saturado, lo que representa dificultades en la aplicación real del proceso, ya que resulta impráctico el diseño de un compresor para el manejo de dos fases. Es gracias a esto que se concluye que el ciclo de Carnot no puede llevarse a cabo con dispositivos reales²⁷.

4.7.3. Ciclo Rankine

Los inconvenientes del ciclo de Carnot enunciados anteriormente, pueden ser eliminados al sobrecalentar el vapor en la caldera y condensar por completo el fluido en el condensador. Al resultado de estas modificaciones se le conoce como ciclo Rankine, el cual se esquematiza en el diagrama T-s mostrado en la figura 16. Cabe señalar, que el diagrama corresponde al ciclo Rankine ideal, en donde no se tienen en cuenta irreversibilidades internas y al igual que el ciclo de Carnot, se compone de cuatro procesos. El proceso 1-2 es una adición de calor a presión constante en una caldera. El proceso 2-3 es una expansión isentrópica en una turbina. El proceso 3-4 es un rechazo de presión a temperatura constante en un condensador. El proceso 4-1 es una compresión isentrópica en una bomba²⁸.

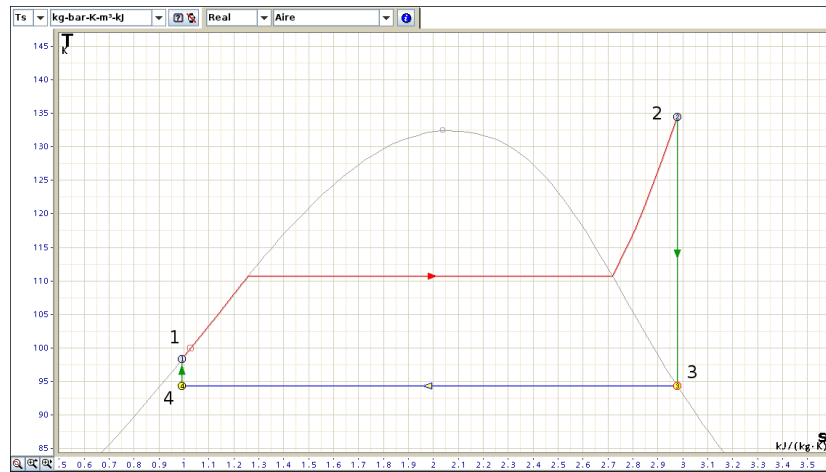
4.7.4. Ciclo Brayton

A diferencia del ciclo de Carnot y el ciclo Rankine, el ciclo Brayton opera con gases de combustión y no con vapor. En la figura 17, se esquematiza el montaje de una planta que opera bajo ciclo Brayton. En ella se evidencian los tres componentes primordiales

²⁷ ÇENGEL y BOLES, Termodinámica, p. 560 - 561.

²⁸ Ibid., p. 562.

Figura 16: Diagrama T-s de un ciclo Rankine ideal



Fuente: El autor

para la operación del ciclo, el compresor, que toma aire del ambiente, lo comprime y lo descarga a mayor presión hacia la cámara de combustión, en la cual se mezcla combustible con aire a presión, provocando así una reacción de combustión. Los gases de combustión posteriormente fluyen a través de la turbina, produciendo trabajo²⁹.

A pesar al proceso anterior, es común estudiar el ciclo empleando un análisis estándar de aire, es decir, asumir como fluido de trabajo el aire, el cual se comporta como gas ideal, y el aumento de temperatura producto de la combustión, es ocasionada por una transferencia de calor externa al sistema. De esta manera, se evitan las complejidades del proceso de combustión y los cambios de composición durante esta³⁰.

4.7.5. Ciclo de refrigeración por compresión

Si bien existen diferentes ciclos de refrigeración, los sistemas de refrigeración por compresión son los mas importantes desde el punto de vista de aplicaciones domésticas y comerciales. Utilizan vapor como fluido de trabajo y representan la forma mas practica para refrigeración. La figura 18 se presenta el esquema de un sistema de refrigeración por compresión. En él se observan los cuatro procesos, compresión, condensación, expansión y evaporación, fundamentales para la operación del ciclo de refrigeración por compresión³¹.

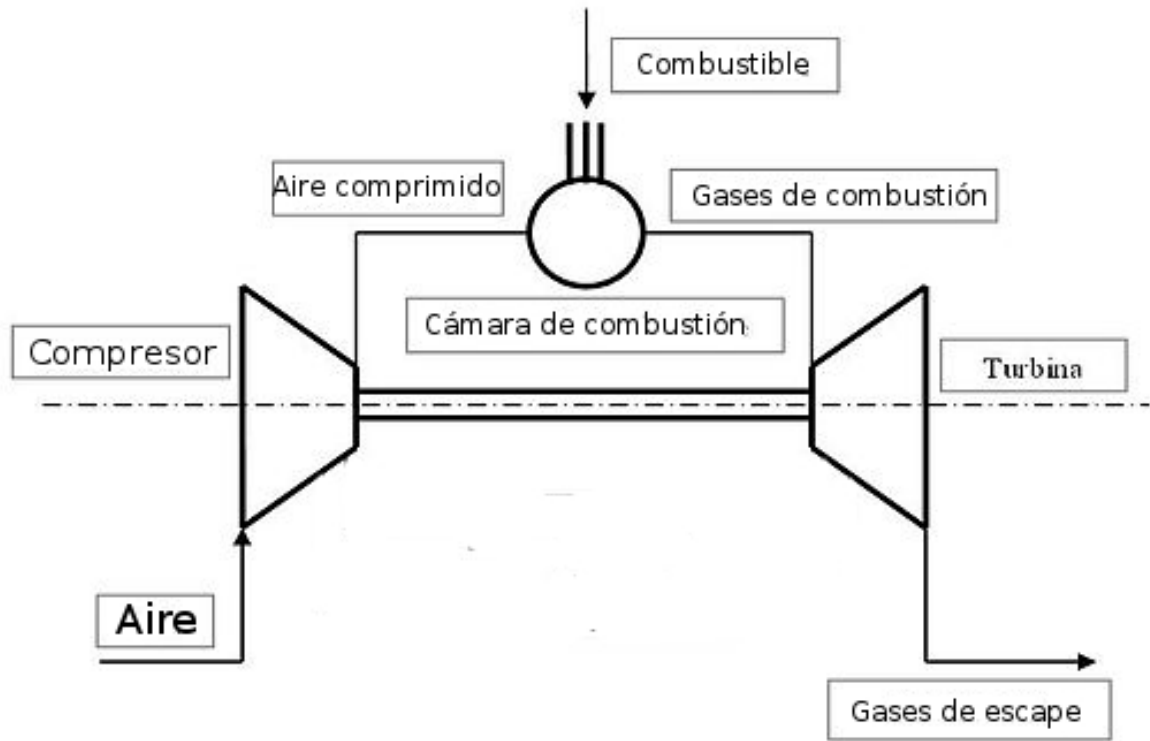
Tener un coeficiente de rendimiento alto, un costo de operación considerablemente más bajo que los sistemas de refrigeración por aire a nivel del mar, un evaporador mas pequeño para el mismo efecto refrigerante que los demás ciclos de refrigeración, son

²⁹MORAN y col., Fundamentals of Engineering Thermodynamics, p. 509.

³⁰Ibíd., p. 510.

³¹RAJPUT, Engineering Thermodynamics, p. 730.

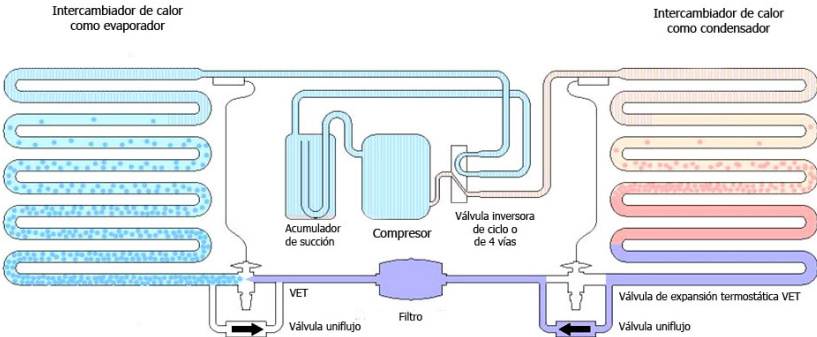
Figura 17: Diagrama P-v de un ciclo Brayton ideal



Fuente: Por Dott. Ing. Sebastian Brusca (Presentación de power point) [CC BY-SA 3.0 (<http://creativecommons.org/licenses/by-sa/3.0>)], vía Wikimedia Commons / Modificado a partir del original

algunas de las ventajas de los ciclos de refrigeración por compresión. Por otro lado, tener un costo inicial alto, probabilidad de filtración de vapores, toxicidad e inflamabilidad, son algunas de las desventajas de estos sistemas. Sin embargo, sus desventajas han sido disminuidas con el tiempo, gracias a optimización en el diseño de los sistemas.

Figura 18: Diagrama P-v de un ciclo Brayton ideal



Fuente: Por Neurotronix (Trabajo propio) CC BY-SA 4.0-3.0-2.5-2.0-1.0 (<http://creativecommons.org/licenses/by-sa/4.0-3.0-2.5-2.0-1.0>), vía Wikimedia Commons / Modificado a partir del original

5. METODOLOGÍA

Este proyecto fue llevado a cabo siguiendo un modelo iterativo de prototipado, buscando así un desarrollo incremental y retroalimentado de todos los elementos que lo componen.

5.1. ENUNCIADO DEL PROBLEMA

Para tener un punto de partida en el desarrollo de este proyecto, se comenzó por redactar un pequeño enunciado que permitiera visualizar el problema a solucionar.

Se desea desarrollar una herramienta informática que permita simular sistemas térmicos de potencia y refrigeración. Los usuarios de dicha aplicación serán docentes, profesionales o estudiantes, los cuales si bien tendrán conocimiento del tema, pueden no tener un conocimiento avanzado de computación, por lo que la interfaz gráfica del software debe ser lo más interactiva y sencilla posible

5.2. LISTA DE REQUERIMIENTOS

Ya con el enunciado del problema, se crea una lista con todas las funciones que el software idealmente debe tener, buscando de esta manera dar un mejor sentido a los requerimientos del proyecto.

- Analizar componentes tanto individualmente, como en grupo sin necesariamente formar un ciclo cerrado con estos.
- Crear el sistema a simular según la necesidad del caso, con el número de componentes necesarios y en el orden necesario.
- Presentar los diagramas termodinámicos del sistema estudiado.
- Generar un informe que contenga el diagrama del sistema, los diagramas termodinámicos del proceso, y las propiedades termodinámicas de cada componente involucrado en el proceso.
- Guardar y/o reabrir el sistema trabajado.

5.3. INTERACCIÓN ENTRE EL USUARIO Y EL SOFTWARE

Con la lista de requerimientos ya creada, se busca apoyar al posterior desarrollo de una interfaz de usuario amigable, por lo que se crea una lista del posible flujo de trabajo de un usuario al interactuar con el software.

- El usuario ejecuta el software.
- El usuario crea un nuevo proyecto o abre uno existente.
- El usuario agrega dispositivos al sistema.
- El usuario conecta los dispositivos.
- El usuario configura los dispositivos del sistema.
- El usuario verifica la validez del sistema.
- Si el sistema es válido se emite una señal positiva, sino se emite una señal negativa con una advertencia.
- El usuario simula el sistema.
- Se emite una señal informando que la simulación finalizó sin problemas.
- El usuario visualiza los resultados de la simulación.
- El usuario genera el informe del proyecto.
- El usuario guarda el proyecto.

Con la información de esta lista, se siguen las bases de un proceso de diseño interactivo, identificando los procesos repetitivos y buscando estrategias para hacer estos procesos obvios y sencillos de utilizar por parte del usuario.

5.4. DISEÑO DE LA INTERFAZ DE USUARIO

Ya que un buen diseño de la interfaz gráfica es importante en el desarrollo de un software, si se quiere que llegue a la mayor cantidad de usuarios posibles. Ya que Apple Inc.[®] es ampliamente reconocida a nivel mundial, por desarrollar productos con los que sus usuarios interactúan de forma muy natural, se decide tomar como guía para el diseño de la interfaz las directrices de interacción humana de Apple Inc.

5.4.1. Primer diseño

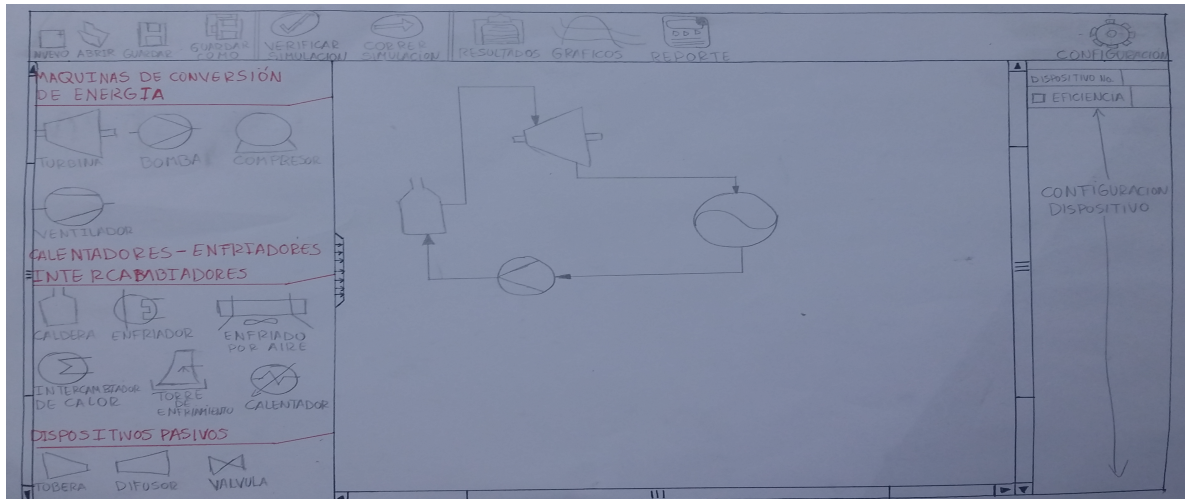
Para facilitar y acelerar el prototipado de la interfaz, se diseña inicialmente en papel y lápiz. La figura 19 presenta el dibujo original del primer diseño de la interfaz gráfica.

5.4.2. Implementaciones

A continuación, la figura 20 presenta la primera implementación de la interfaz de usuario, en el software desarrollado.

La figura 21 presenta el diseño final de la interfaz de usuario, en la implementación definitiva del software.

Figura 19: Prototipado de la interfaz



Fuente: El autor

5.5. MODELAMIENTO DEL SOFTWARE

El desarrollo de este software tiene dos elementos primordiales para su funcionamiento, el modelamiento de un ciclo de potencia o refrigeración, y el modelamiento del proceso de construcción del archivo con extensión .xml que permite guardar y reabrir un proyecto creado.

La figura 22 presenta el diagrama de flujo que modela el proceso de cálculo de un ciclo dentro del software.

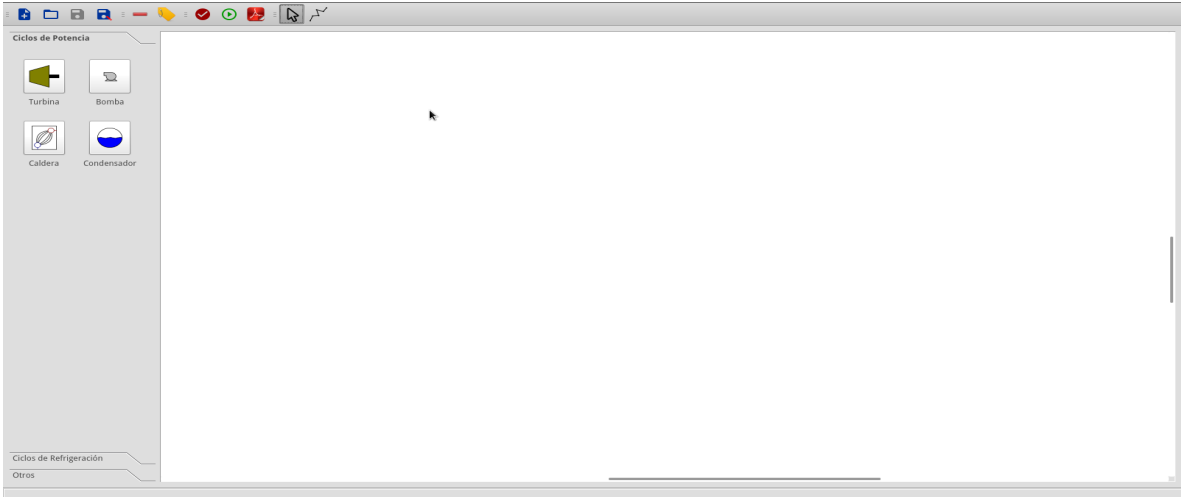
La figura 23 presenta el diagrama de flujo que modela el proceso de creación del documento xml, que permitira guardar y reabrir los proyectos creados.

5.6. DIAGRAMA DE CLASES

Una vez modelados estos dos procesos y la interfaz de usuario diseñada, se procede a modelar la estructura general del software con un diagrama de clases en lenguaje UML. La figura 24 muestra un diagrama de clases simplificado del software a codificar. Para observar el diagrama de clases con total detalle, referirse al apéndice B de este documento.

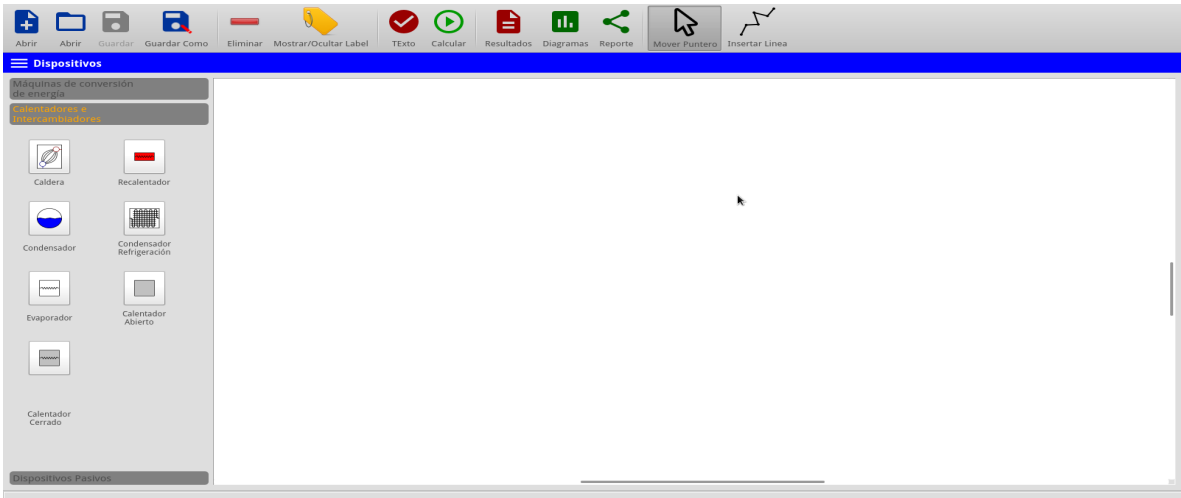
A continuación, se procede a la codificación de todos los esquemas y modelos presentados con anterioridad, llevando así el software a una etapa funcional.

Figura 20: Primera implementación



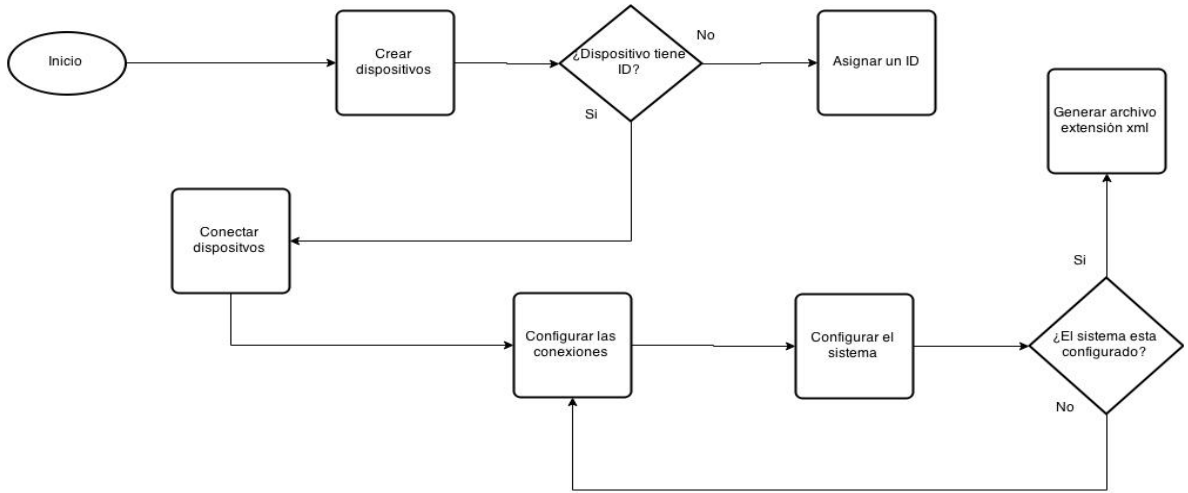
Fuente: El autor

Figura 21: Implementación final



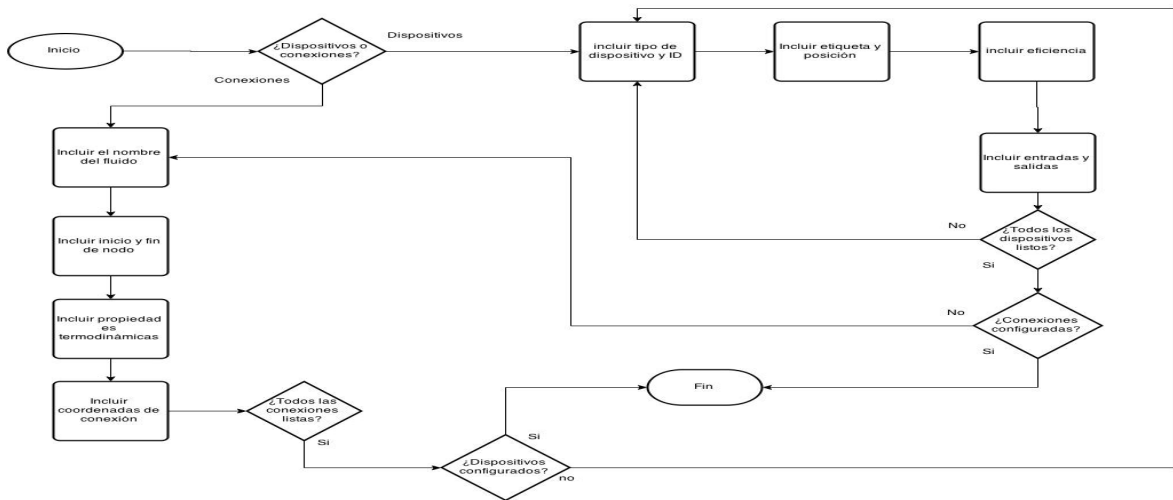
Fuente: El autor

Figura 22: Modelamiento del proceso del cálculo de un ciclo



Fuente: El autor

Figura 23: Modelamiento del proceso de creación del archivo xml



Fuente: El autor

6. RESULTADOS Y DISCUSIÓN

Para validar los resultados del software, se simuló en los tres ejemplos resueltos, dos de ellos de ciclos de vapor y uno de ellos de ciclos de refrigeración. Una vez simulado el ejemplo, se comparan resultados, se analizan y comentan.

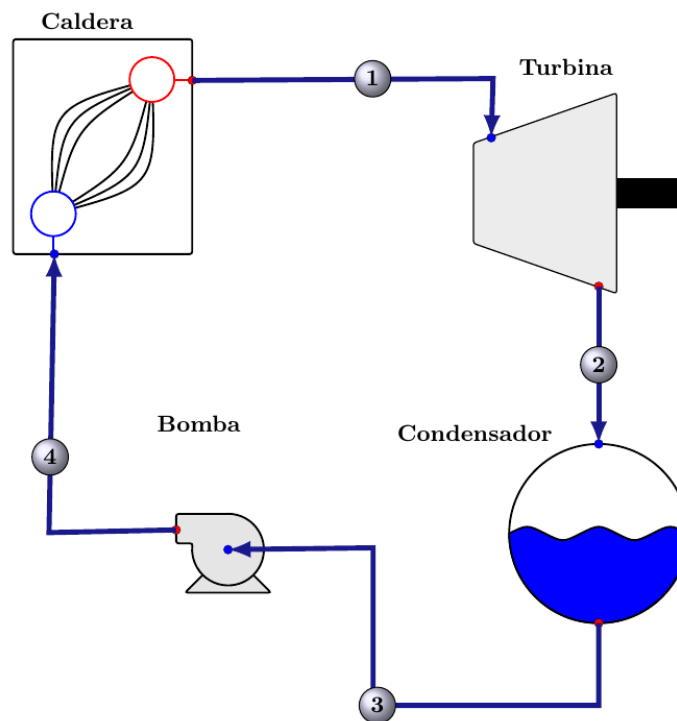
6.1. SIMULACIONES DE CICLO RANKINE

6.1.1. Caso 1: Ciclo Rankine simple

6.1.1.1. Enunciado

Ejemplo 10-1¹. Considere una central eléctrica de vapor que opera en el ciclo Rankine ideal simple. El vapor de agua entra a la turbina a 3 MPa y 350 °C y es condensado en el condensador a una presión de 75 kPa. Determine la eficiencia térmica de este ciclo. La figura 25 muestra el diagrama de la planta correspondiente a este problema.

Figura 25: Diagrama de la planta para el caso 1



El autor

¹ÇENGEL y BOLES, Termodinámica, p. 563 - 565.

6.1.1.2. Resultados

La tabla 1 presenta las entalpías calculadas en la simulación, las entalpías obtenidas del ejemplo y el error calculado entre estas dos.

Tabla 1: Resultado de entalpía en cada estado

Estado	Entalpía Simulación [kJ/kg]	Entalpía Ejemplo[kJ/kg]	Error [%]
1	3116,063	3115	0,03412520064
2	2403,004	2663	-9,763274502
3	384,432	384,4	0,008324661811
4	387,471	387,5	-0,007483870968

La tabla 2 presenta la eficiencia térmica calculada en la simulación, la eficiencia dada por el ejemplo y el error entre estas dos.

Tabla 2: Eficiencia térmica de la planta

$\eta_{simulación}$ [%]	$\eta_{ejemplo}$ [%]	Error [%]
26,021	26	0,08076923077

6.1.1.3. Análisis de resultados del caso

Como se evidencia en las tablas 1 y 2, el software desarrollado permite simular sistemas que operan en ciclo Rankine simple, con un error en la eficiencia calculada por la simulación inferior al 0,1 % y un el error en las entalpías calculadas por la simulación que no supera el 10 % en su punto máximo.

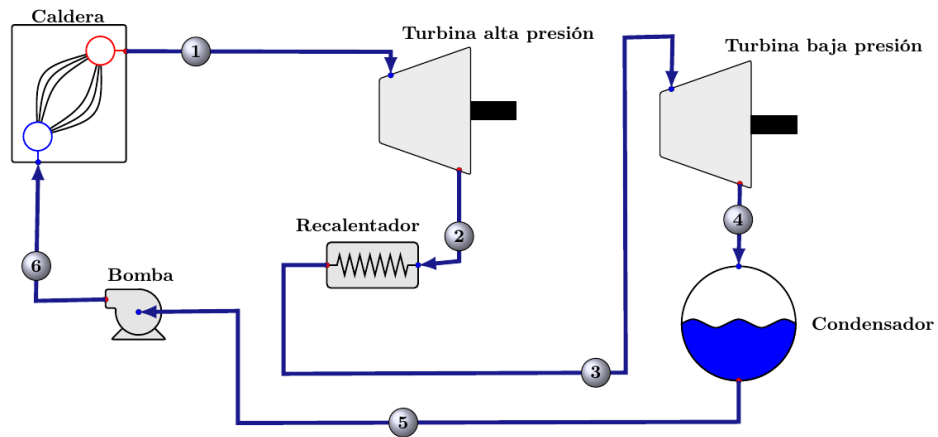
6.1.2. Caso 2: Ciclo Rankine con recalentamiento

6.1.2.1. Enunciado

EJEMPLO 9.3². Considérese un ciclo de recalentamiento que utiliza vapor de agua. El vapor sale de la caldera y entra a la turbina a 4 MPa y 400°C. Luego de ser expandido en la turbina hasta 400 kPa, el valor es recalentado a 400°C y luego expandido hasta 10 kPa en la turbina de baja presión. Determinar la eficiencia del ciclo. La figura 26 muestra el diagrama de la planta correspondiente a este problema.

²BORGNAKKE y SONNTAG, Fundamentals of Thermodynamics, p. 415 - 417.

Figura 26: Diagrama de la planta para el caso 2



El autor

6.1.2.2. Resultados

La tabla 3 presenta las entalpías calculadas en la simulación, las entalpías obtenidas del ejemplo y el error calculado entre estas dos. Como se puede ver en este caso, el error entre las entalpías calculadas y las propias del ejemplo, es inferior al 0,1 %.

Tabla 3: Resultado de entalpía en cada estado

Estado	Entalpía Simulación [kJ/kg]	Entalpía Ejemplo[kJ/kg]	Error [%]
1	3214,473	3213,6	0,02716579537
2	2686,309	2685,6	0,02640005958
3	3273,91	3273,4	0,01558013075
4	2504,572	2504,3	0,01086131853
5	191,803	191,8	0,001564129301
6	195,834	195,8	0,01736465781

La tabla 4 presenta la eficiencia térmica calculada en la simulación, la eficiencia obtenida del ejemplo y el error entre estas dos. En este caso, el error entre la eficiencia propia del ejemplo y la calculada mediante la simulación, es inferior al 0,1 %.

6.1.2.3. Análisis de resultados del caso

Como se evidencia en las tablas 3 y 4, el software desarrollado permite simular sistemas que operan en ciclo Rankine con recalentamiento, con un error tanto en la

Tabla 4: Eficiencia térmica de la planta

$\eta_{simulación} [\%]$	$\eta_{ejemplo} [\%]$	Error [%]
35,868	35,9	-0,08913649025

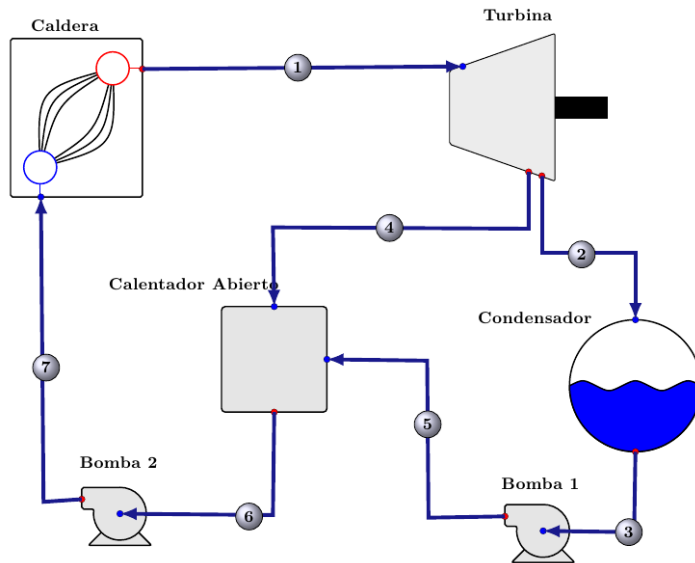
eficiencia calculada por la simulación como en las entalpías calculadas por la simulación inferior al 0,1 %.

6.1.3. Caso 3: Ciclo Rankine regenerativo con CAA Abierto

6.1.3.1. Enunciado

Ejemplo 10-5³. Considere una central eléctrica de vapor que opera en un ciclo Rankine ideal regenerativo con un calentador abierto de agua de alimentación. El vapor entra a la turbina a 15 MPa y 600 °C, y se condensa en el condensador a una presión de 10 kPa. Una parte de vapor sale de la turbina a una presión de 1.2 MPa y entra al calentador abierto de agua de alimentación. Determine la fracción de vapor extraído de la turbina y la eficiencia térmica del ciclo. La figura 27 muestra el diagrama de la planta correspondiente a este problema.

Figura 27: Diagrama de la planta para el caso 3



El autor

³ÇENGEL y BOLES, Termodinámica, p. 580 - 582.

6.1.3.2. Resultados

La tabla 5 presenta las entalpías calculada en la simulación, las entalpías dadas por el ejemplo y el error entre estas dos. En el caso de este problema, los errores en el cálculo de propiedades no superan el 6 %.

Tabla 5: Resultado de entalpía en cada estado

Estado	Entalpía Simulación [kJ/kg]	Entalpía Ejemplo[kJ/kg]	Error [%]
1	3583,132	3581	0,05953644233
2	2115,239	2114	0,05860927152
3	191,803	191,8	0,001564129301
4	203,889	193	5,641968912
5	2858,779	2858	0,02725682295
6	798,332	798,7	-0,04607487167
7	813,97	814,3	-0,04052560481

La tabla 6 presenta la eficiencia térmica calculada en la simulación, la eficiencia dada por el ejemplo y el error entre estas dos. En este caso, el error en las eficiencias entre la simulación y el ejemplo es inferior al 1 %

Tabla 6: Eficiencia térmica del sistema

$\eta_{simulación}$ [%]	$\eta_{ejemplo}$ [%]	Error [%]
46,093	46,3	-0,4470842333

6.1.3.3. Análisis de resultados del caso

Como se evidencia en las tablas 5 y 6, el software desarrollado permite simular sistemas que operan en ciclo Rankine regenerativo con un calentador de agua de alimentación abierto, obteniendo en los resultados calculados por la simulación, un error en la eficiencia inferior al 1 % y un error en las entalpías que no supera el 6 % en su punto máximo.

6.1.4. Caso 4: Ciclo Rankine regenerativo con recalentamiento

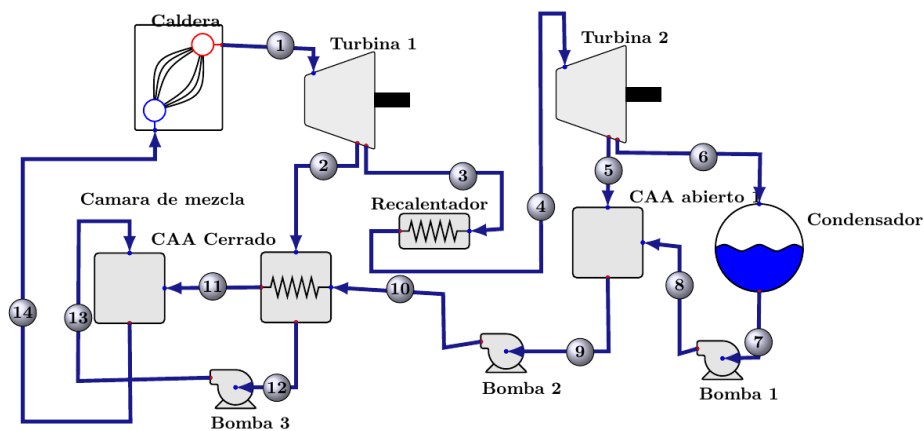
6.1.4.1. Enunciado

EJEMPLO 10-6⁴. Considere una central eléctrica de vapor que opera en un ciclo Rankine ideal regenerativo con recalentamiento, con dos calentadores de agua de alimentación, uno abierto y otro cerrado, además de un recalentador. El vapor entra a la

⁴ÇENGEL y BOLES, Termodinámica, p. 582 - 584.

turbina a 15 MPa y 600°C y se condensa a una presión de 10 kPa. Una parte de vapor se extrae de la turbina a 4 MPa para el calentador cerrado, mientras que el resto se recalienta a la misma presión hasta 600°C. El vapor extraído se condensa por completo en el calentador y se bombea hasta 15 MPa antes de mezclarse con el agua de alimentación a la misma presión. El vapor para el calentador abierto se extrae de la turbina de baja presión a una presión de 0.5 MPa. Determine las fracciones de vapor extraído de la turbina, así como la eficiencia térmica del ciclo. La figura 28 muestra el diagrama de la planta correspondiente a este problema.

Figura 28: Diagrama de la planta para el caso 4



El autor

6.1.4.2. Resultados

La tabla 7 presenta las entalpías calculadas en la simulación, las entalpías obtenidas del ejemplo y el error calculado entre estas dos.

La tabla 8 presenta la eficiencia térmica calculada en la simulación, la eficiencia obtenida del ejemplo y el error entre estas dos.

La tabla 9 presenta los índices de flujo calculados en la simulación, los índices de flujo obtenidos del ejemplo y el error calculado entre estos dos.

6.1.4.3. Análisis de resultados del caso

Para este caso, el error en la entalpía calculada por la simulación no supera el 2%, tal como se muestra en la tabla 7. Sin embargo, el error en la eficiencia calculada por la simulación es de 9,3%, tal como se observa en la tabla 8. Si bien este error es inferior al 10%, es considerable a causa de las proporciones del sistema.

Los resultados presentados en la tabla 9, proporcionan una idea sobre la fuente de error, ya que en ella, se observan dos inconsistencias en el cálculo del flujo en la

Tabla 7: Resultado de entalpía en cada estado

Estado	Entalpía Simulación [kJ/kg]	Entalpía Ejemplo[kJ/kg]	Error [%]
1	3583,132	3583,1	0,00089308141
2	3153,875	3155	-0,03565768621
3	3153,875	3155	-0,03565768621
4	3674,868	3674,9	-0,0008707719938
5	3013,59	3014,8	-0,04013533236
6	2335,618	2335,7	-0,003510724836
7	191,803	191,81	-0,003649444763
8	192,301	192,3	0,0005200208008
9	640,087	640,09	-0,0004686840913
10	655,868	643,92	1,855510001
11	1087,798	1087,4	0,03660106676
12	1087,489	1087,4	0,008184660658
13	1101,098	1101,2	-0,009262622594
14	1090,098	1098,8	-0,7919548598

Tabla 8: Eficiencia térmica de la planta

$\eta_{simulación}$ [%]	$\eta_{ejemplo}$ [%]	Error [%]
44,624	49,2	-9,300813008

simulación. La primera inconsistencia, se apoya en el hecho que el valor de la fracción de flujo nunca puede ser mayor a la unidad, que es precisamente lo que se observa en los estados 1 y 14 del sistema. La segunda inconsistencia se encuentra entre los estados 2 y 3, donde la suma de los flujos de estos debe ser igual a la unidad y no a 0,604, ya que en el sistema no hay derivaciones de flujo antes de estos.

Posterior a la simulación, los datos de entrada del sistema fueron revisados estado a estado, sin encontrar errores. Adicional a esto, el calculo de flujos fue consistente con los resultados del sistema, tanto en el caso 3 como en simulaciones preliminares, viéndose inconsistencias solo en este caso. A causa de esto, se considera que la fuente de error es un “bug” que se manifiesta bajo las condiciones particulares de este caso.

Para solucionar este problema, el debido proceso consistirá en replicar el “bug” en diferentes simulaciones, identificar la fuente precisa del error, repararlo e intentar replicarlo de nuevo, verificando así que se encuentra solucionado.

Tabla 9: Resultado de flujo en cada estado

Estado	Flujo Simulación	Flujo Ejemplo	Error [%]
1	1,083	1	8,3
2	0,187	0,1766	5,889014723
3	0,417	0,8214	-49,2330168
4	0,417	0,8214	-49,2330168
5	0,142	0,1306	8,728943338
6	0,753	0,6908	9,004053272
7	0,753	0,6908	9,004053272
8	0,753	0,6908	9,004053272
9	0,895	0,8214	8,960311663
10	0,895	0,8214	8,960311663
11	0,895	0,8214	8,960311663
12	0,187	0,1766	5,889014723
13	0,187	0,1766	5,889014723
14	1,083	1	8,3

6.2. SIMULACIÓN DE CICLOS DE REFRIGERACIÓN

6.2.1. Caso 5: Refrigeración por compresión

6.2.1.1. Enunciado

EJEMPLO 11-1⁵. En un refrigerador se utiliza refrigerante 134a como fluido de trabajo, y opera en un ciclo ideal de refrigeración por compresión de vapor entre 0.14 y 0.8 MPa. Si el flujo másico del refrigerante es de 0.05 kg/s, determine a) la tasa de eliminación de calor del espacio refrigerado y la entrada de potencia al compresor, b) la tasa de rechazo de calor al ambiente y c) el COP del refrigerador.

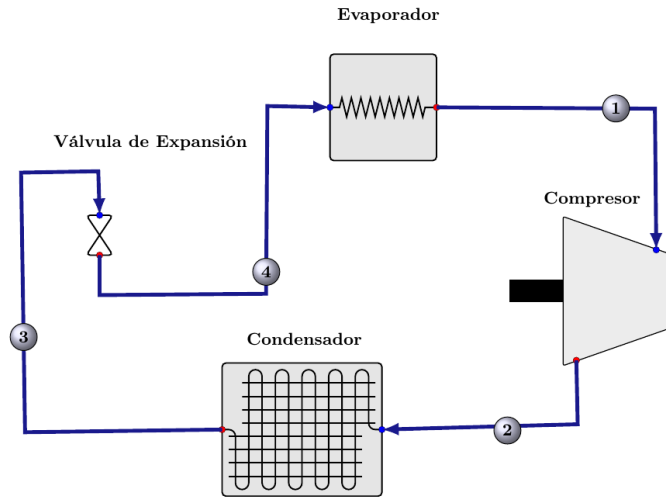
6.2.1.2. Resultados

La tabla 10 presenta los resultados del cálculo de entalpía obtenidos en la simulación, los resultados de entalpía propios del ejemplo y el error entre estos dos. En el caso de este problema, el error entre las entalpías calculadas y el ejemplo, se encuentra entre el 54% y el 155% aproximadamente. La explicación correspondiente de la razón de este error, será presentada mas adelante.

La tabla 11 presenta la eficiencia térmica calculada en la simulación, la eficiencia dada por el ejemplo y el error entre estas dos. En este caso, el error es inferior al presentando en las entalpías, sin embargo, sigue siendo considerablemente alto. Al igual

⁵ÇENGEL y BOLES, Termodinámica, p. 621 - 622.

Figura 29: Diagrama del refrigerador para el caso 1



El autor

Tabla 10: Resultado de entalpía en cada estado

Estado	Entalpía Simulación [kJ/kg]	Entalpía Ejemplo[kJ/kg]	Error [%]
1	387,32	239,16	61,95015889
2	423,532	275,39	53,79352918
3	243,645	95,47	155,2058238
4	243,645	95,47	155,2058238

que para las entalpías, la explicación correspondiente de esta situación, será presentada más adelante.

Tabla 11: Eficiencia térmica de la planta

$\eta_{simulación}$ [%]	$\eta_{ejemplo}$ [%]	Error [%]
3,968	3,064	29,50391645

6.2.1.3. Análisis de de resultados del caso

Para los refrigerantes como el R-134a, utilizado como flujo de trabajo en el problema 3, las entalpías son calculadas a partir de un punto de referencia. Según REFPROP™ y CoolProp™, se toma como referencia para el cálculo de entalpías, el valor de la entalpía

para líquido saturado a 0°C , que en el caso del refrigerante R-134a es de $200[\text{Kj}/\text{Kg}]^6$. Por otro lado, el ejemplo y sus resultados fueron tomados del libro de termodinámica de Çengel, en el cual usan como referencia para el cálculo de entalpías del refrigerante R-134a, el valor de la entalpía para líquido saturado a -40°C , que en este caso es 0^7 . Es a causa de esta diferencia entre las dos referencias anteriores, que los valores de entalpía calculada difieren de los del ejemplo.

⁶DuPont Corporation. Thermodynamic Properties of HFC-134a Refrigerant (1,1,1,2-tetrafluoroethane) SI Units. URL: http://www2.dupont.com/Refrigerants/en_US/assets/downloads/h47752_hfc134a_thermo_prop_si.pdf, p. 1.

⁷ÇENGEL y BOLES, Termodinámica, p. 926 - 939.

7. CONCLUSIONES Y RECOMENDACIONES

Con este proyecto se busco sentar un precedente para futuros proyectos que involucren el desarrollo de aplicaciones informáticas, relacionadas con la ingeniería mecánica. Se desarrollo el software de simulación cumpliendo de manera satisfactoria con los objetivos propuestos por el proyecto. Como se mostró con anterioridad, el software permite calcular de manera rápida e interactiva, los sistemas térmicos que cumplan con las siguientes condiciones:

1. El sistema debe operar en un ciclo cerrado.
2. Cada dispositivo debe tener uno de sus estados definido termodinámicamente, y conocerse una propiedad termodinámica en los demás.

Sin embargo, se busca que el software creado sirva como base para el desarrollo de proyectos de mayor envergadura o para el desarrollo incremental del software mismo, ya que teniéndolo como base, no es necesario empezar de cero.

Para aquellos futuros proyectos que busquen complementar el software desarrollado, se recomienda una ampliación de la librería de dispositivos, buscando así posibilitar mayores oportunidades de aplicación. Además, si bien la interfaz de usuario fue diseñada lo mas sencilla y fácil de usar posible, hay muchas maneras en las que el aplicativo puede mejorarse, como incluyendo mejores opciones para la configuración de los dispositivos y del sistema mismo.

Por ultimo, se espera que el software desarrollado en este proyecto sea, usado tanto por estudiantes en las asignaturas de termodinámica como por investigadores en esta área, buscando fomentar el uso de más herramientas informáticas en la Facultad de Ingeniería Mecánica de la UPB seccional Bucaramanga, que apoyen los procesos académicos. Además, se espera con esto despertar conciencia en los estudiantes, sobre la importancia de las herramientas computacionales, la programación y el modelamiento, en la ingeniería de hoy en día.

BIBLIOGRAFÍA

- ARIANE[®]. software. ProSim PLC. URL: <http://www.prosim.net/en/software-ariane-10.php>.
- BALMER, Robert T. Modern Engineering Thermodynamics. Elsevier Inc, 2011.
- BAUDOIN, Olivier, VACHER, Alain y D'ECHELOTTE, Stéphane. Simulis[®] Thermodynamics: an open framework for users and developers. Visitado: 04-03-2015. ProSim SA. URL: http://www.prosim.net/bibliotheque/File/Publications/Events/32_ESAT_E_2008-05_SimulisThermodynamics_anopenframework%20forusersanddevelopers.pdf.
- BELL, Ian H. y col. Pure and Pseudo-pure Fluid Thermophysical Property Evaluation and the Open-Source Thermophysical Property Library CoolProp. En: *Industrial & Engineering Chemistry Research* 53.6 (2014), págs. 2498-2508. DOI: 10.1021/ie4033999. eprint: <http://pubs.acs.org/doi/pdf/10.1021/ie4033999>. URL: <http://pubs.acs.org/doi/abs/10.1021/ie4033999>.
- BORGNAKKE, Claus y SONNTAG, Richard E. Fundamentals of Thermodynamics. 8.^a ed. John Wiley & Sons Inc, 2013.
- ÇENGEL, Yunus A. y BOLES, Michael A. Termodinámica. 7.^a ed. MCGRAW-HILL, 2012.
- CYCLE-TEMPO. Software. ASIMPTOTE. URL: <http://www.asimptote.nl/software/cycle-tempo>.
- DuPont Corporation. Thermodynamic Properties of HFC-134a Refrigerant (1,1,1,2-tetrafluoroethane) SI Units. URL: http://www2.dupont.com/Refrigerants/en_US/assets/downloads/h47752_hfc134a_thermo_prop_si.pdf.
- GATECYCLE[®]. Software. GE Enter Software LLC. 1490 Drew Avenue, Suite 180, Davis, California 95616.
- LEMMON, Eric W., HUBER, M. L. y MCLINDEN, M. O. NIST Standard Reference Database 23: Reference Fluid Thermodynamic and Transport Properties - REFPROP. 9.0. National Institute of Standards and Technology, Standard Reference Data Program. Gaithersburg, 2010. URL: <http://www.nist.gov/srd/nist23.cfm>.
- MORAN, Michael J. y col. Fundamentals of Engineering Thermodynamics. 7.^a ed. John Wiley & Sons Inc, 2011.
- RAJPUT, R. K. Engineering Thermodynamics. 3.^a ed. Laxmi Publications LTD, 2007.
- THERMOFLEX[™]. Software. Thermoflow Inc. URL: http://www.thermoflow.com/products_fullyflexible.html#.

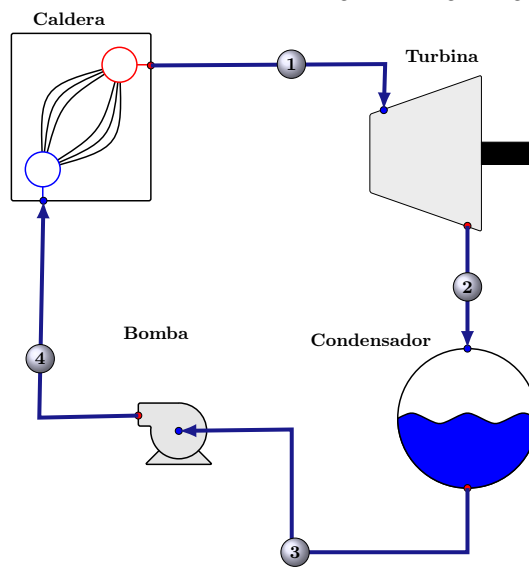
A. INFORMES DE SIMULACIÓN

A.1. CASO 1: RANKINE SIMPLE

1. Información de la Planta

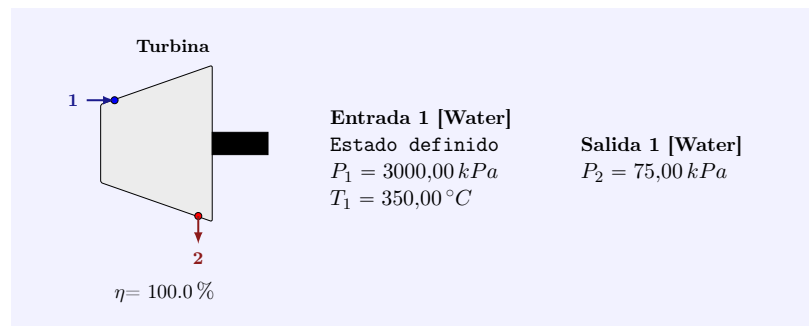
1.1. Diagrama de la Planta

Figura 1: Diagrama general de la planta

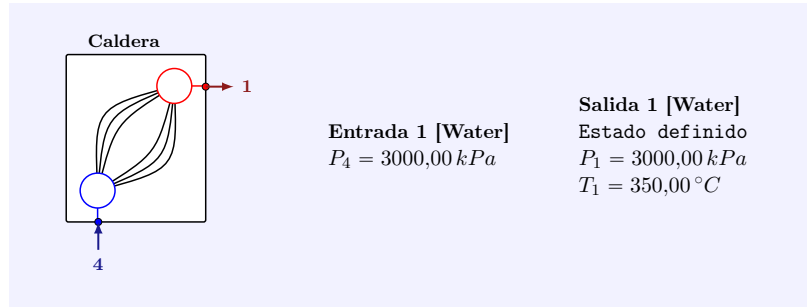


1.2. Análisis de los dispositivos

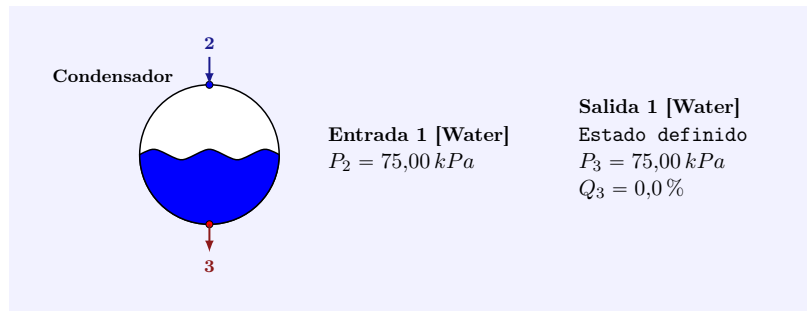
1.2.1. Volúmen de control: Turbina



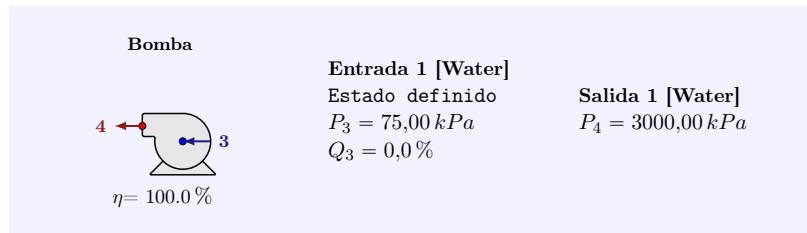
1.2.2. Volúmen de control: Caldera



1.2.3. Volúmen de control: Condensador



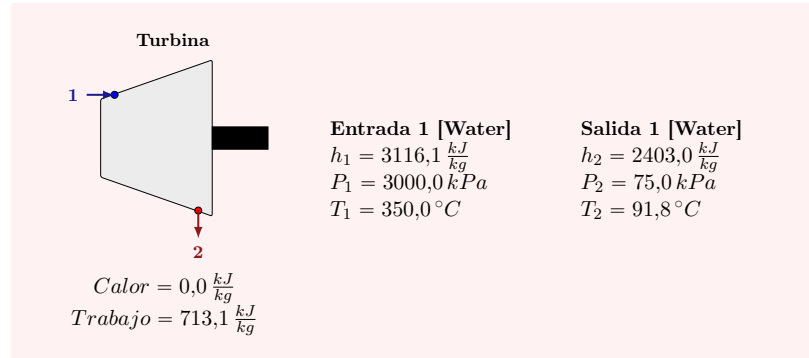
1.2.4. Volúmen de control: Bomba



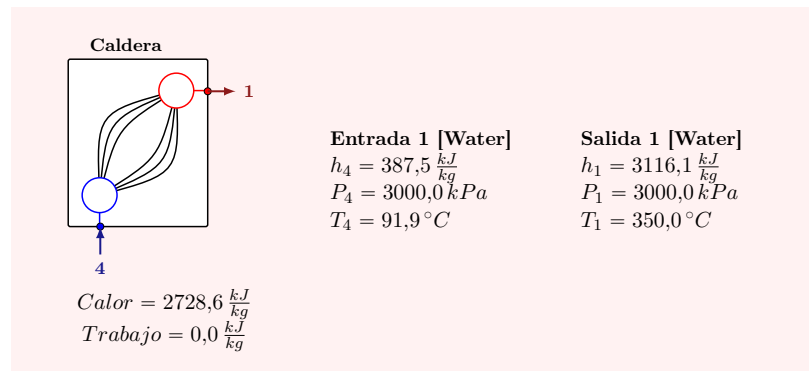
2. Resultados

2.1. Dispositivos

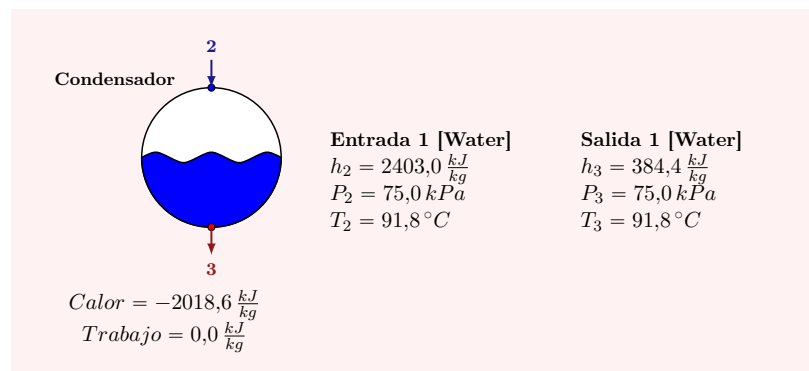
2.1.1. Turbina



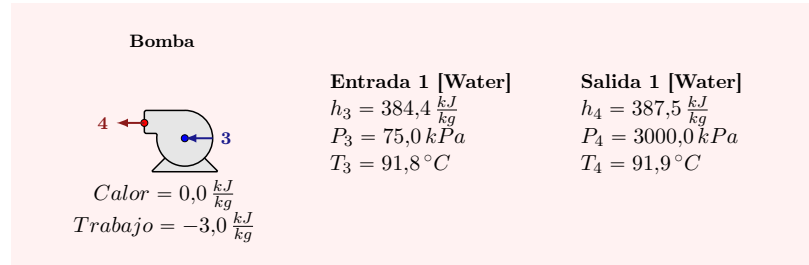
2.1.2. Caldera



2.1.3. Condensador



2.1.4. Bomba



2.2. Eficiencia de la Planta

$$\eta_{termica} = 26,021\%$$

2.3. Resultados de cada estado

Tabla 1: Resultados de cada estado

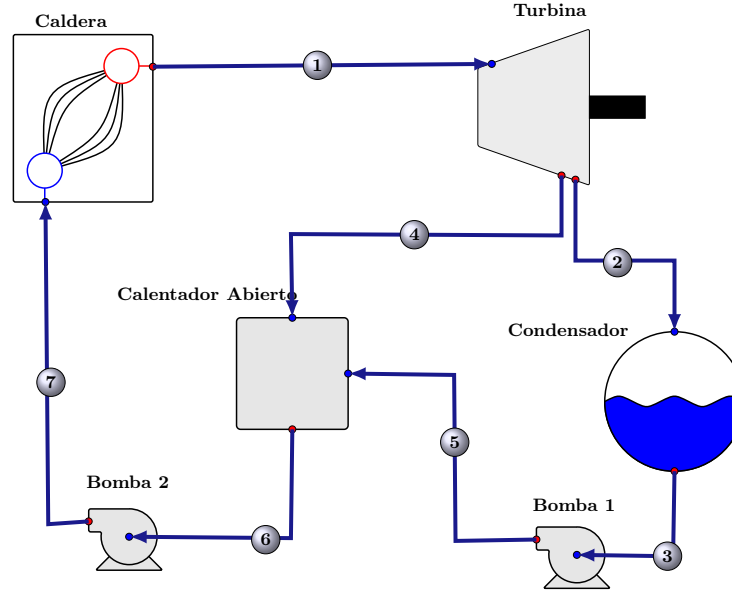
Estado	Fluido	y[flujo]	Presión [kPa]	Temperatura [K]	Entalpía [kJ/kg]
1	Water	1.000	3000.000	623.150	3116.063
2	Water	1.000	75.000	364.908	2403.004
3	Water	1.000	75.000	364.908	384.432
4	Water	1.000	3000.000	365.094	387.471

A.2. CASO 2: RANKINE RECALENTADO

1. Información de la Planta

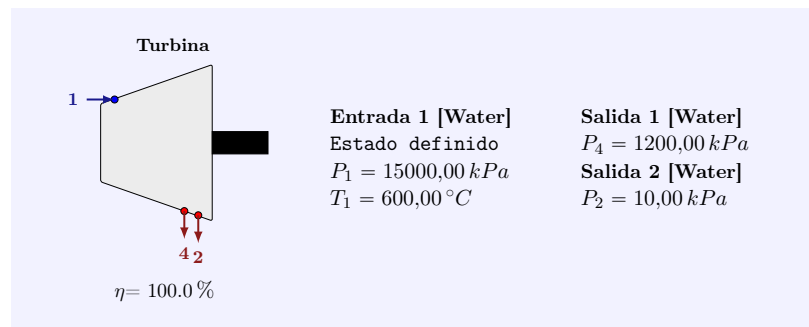
1.1. Diagrama de la Planta

Figura 1: Diagrama general de la planta

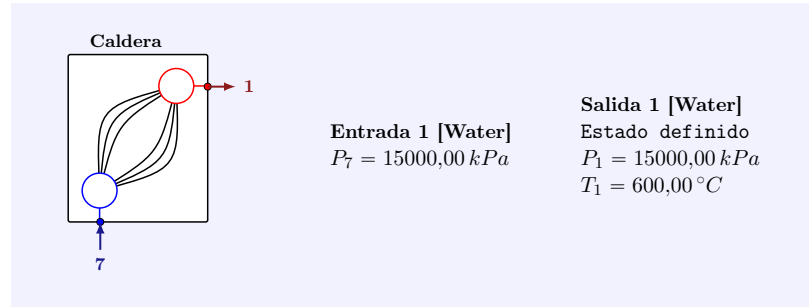


1.2. Análisis de los dispositivos

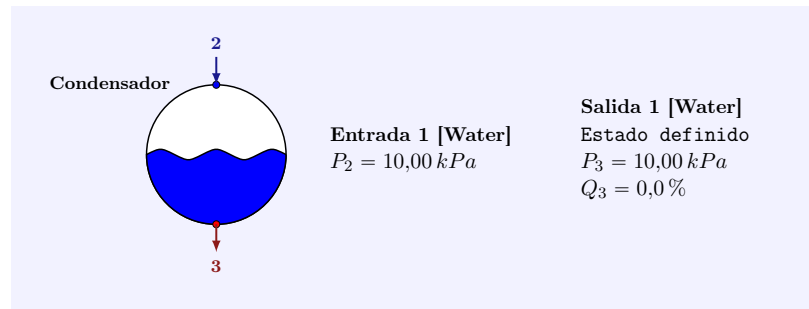
1.2.1. Volúmen de control: Turbina



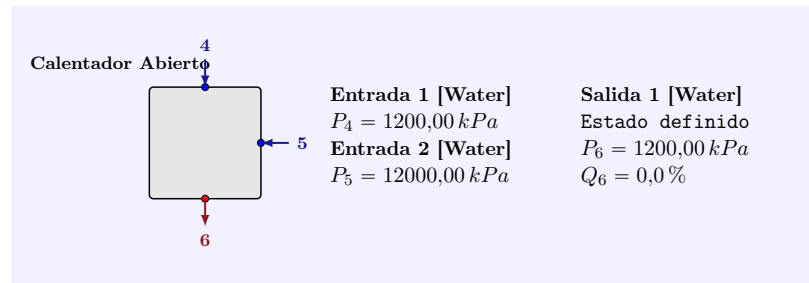
1.2.2. Volúmen de control: Caldera



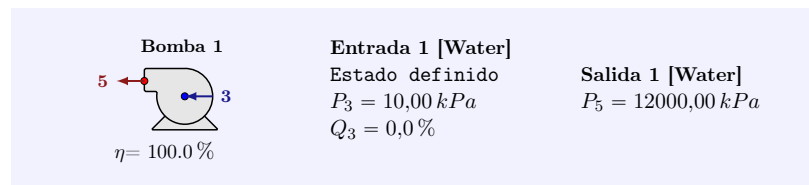
1.2.3. Volúmen de control: Condensador



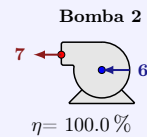
1.2.4. Volúmen de control: Calentador Abierto



1.2.5. Volúmen de control: Bomba 1



1.2.6. Volúmen de control: Bomba 2



Bomba 2

$\eta = 100.0\%$

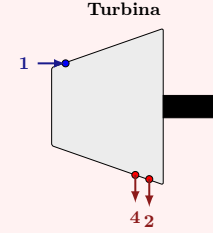
Entrada 1 [Water]
 Estado definido
 $P_6 = 1200,00 \text{ kPa}$
 $Q_6 = 0,0\%$

Salida 1 [Water]
 $P_7 = 15000,00 \text{ kPa}$

2. Resultados

2.1. Dispositivos

2.1.1. Turbina



Turbina

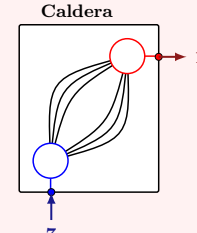
$Calor = 0,0 \frac{\text{kJ}}{\text{kg}}$
 $Trabajo = 1301,4 \frac{\text{kJ}}{\text{kg}}$

Entrada 1 [Water]
 $h_1 = 3583,1 \frac{\text{kJ}}{\text{kg}}$
 $P_1 = 15000,0 \text{ kPa}$
 $T_1 = 600,0^\circ\text{C}$

Salida 1 [Water]
 $h_4 = 2858,8 \frac{\text{kJ}}{\text{kg}}$
 $P_4 = 1200,0 \text{ kPa}$
 $T_4 = 217,1^\circ\text{C}$

Salida 2 [Water]
 $h_2 = 2115,2 \frac{\text{kJ}}{\text{kg}}$
 $P_2 = 10,0 \text{ kPa}$
 $T_2 = 45,8^\circ\text{C}$

2.1.2. Caldera



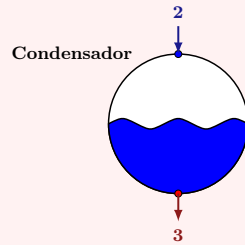
Caldera

$Calor = 2769,2 \frac{\text{kJ}}{\text{kg}}$
 $Trabajo = 0,0 \frac{\text{kJ}}{\text{kg}}$

Entrada 1 [Water]
 $h_7 = 814,0 \frac{\text{kJ}}{\text{kg}}$
 $P_7 = 15000,0 \text{ kPa}$
 $T_7 = 190,0^\circ\text{C}$

Salida 1 [Water]
 $h_1 = 3583,1 \frac{\text{kJ}}{\text{kg}}$
 $P_1 = 15000,0 \text{ kPa}$
 $T_1 = 600,0^\circ\text{C}$

2.1.3. Condensador

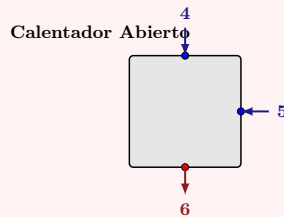


Entrada 1 [Water]
 $h_2 = 2115,2 \frac{kJ}{kg}$
 $P_2 = 10,0 kPa$
 $T_2 = 45,8^\circ C$

Salida 1 [Water]
 $h_3 = 191,8 \frac{kJ}{kg}$
 $P_3 = 10,0 kPa$
 $T_3 = 45,8^\circ C$

$Calor = -1492,8 \frac{kJ}{kg}$
 $Trabajo = 0,0 \frac{kJ}{kg}$

2.1.4. Calentador Abierto



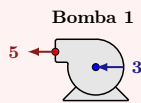
Entrada 1 [Water]
 $h_4 = 2858,8 \frac{kJ}{kg}$
 $P_4 = 1200,0 kPa$
 $T_4 = 217,1^\circ C$

Salida 1 [Water]
 $h_6 = 798,3 \frac{kJ}{kg}$
 $P_6 = 1200,0 kPa$
 $T_6 = 188,0^\circ C$

Entrada 2 [Water]
 $h_5 = 203,9 \frac{kJ}{kg}$
 $P_5 = 12000,0 kPa$
 $T_5 = 46,2^\circ C$

$Calor = 0,0 \frac{kJ}{kg}$
 $Trabajo = 0,0 \frac{kJ}{kg}$

2.1.5. Bomba 1

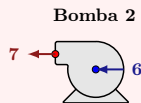


Entrada 1 [Water]
 $h_3 = 191,8 \frac{kJ}{kg}$
 $P_3 = 10,0 kPa$
 $T_3 = 45,8^\circ C$

Salida 1 [Water]
 $h_5 = 203,9 \frac{kJ}{kg}$
 $P_5 = 12000,0 kPa$
 $T_5 = 46,2^\circ C$

$Calor = 0,0 \frac{kJ}{kg}$
 $Trabajo = -9,4 \frac{kJ}{kg}$

2.1.6. Bomba 2



Entrada 1 [Water]
 $h_6 = 798,3 \frac{kJ}{kg}$
 $P_6 = 1200,0 kPa$
 $T_6 = 188,0^\circ C$

Salida 1 [Water]
 $h_7 = 814,0 \frac{kJ}{kg}$
 $P_7 = 15000,0 kPa$
 $T_7 = 190,0^\circ C$

$Calor = 0,0 \frac{kJ}{kg}$
 $Trabajo = -15,6 \frac{kJ}{kg}$

2.2. Eficiencia de la Planta

$$\eta_{termica} = 46,093\%$$

2.3. Resultados de cada estado

Tabla 1: Resultados de cada estado

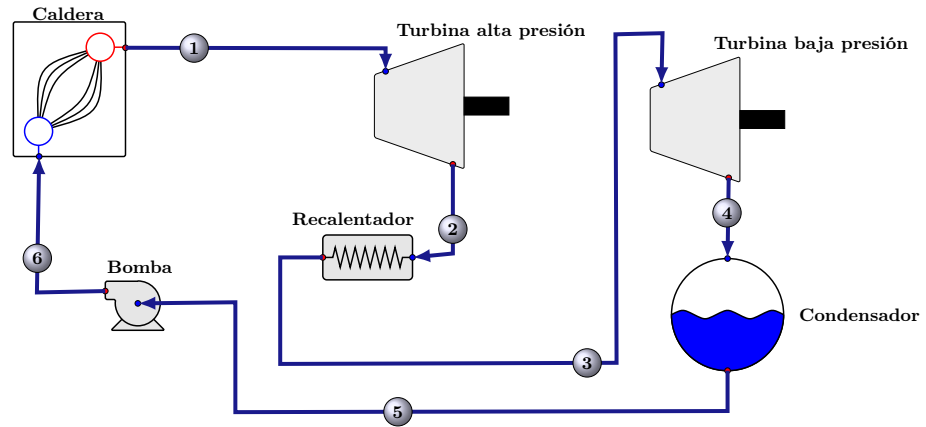
Estado	Fluido	y[flujo]	Presión [kPa]	Temperatura [K]	Entalpía [kJ/kg]
1	Water	1.000	15000.000	873.150	3583.132
2	Water	0.776	10.000	318.956	2115.239
3	Water	0.776	10.000	318.956	191.803
5	Water	0.776	12000.000	319.354	203.889
4	Water	0.224	1200.000	490.276	2858.779
6	Water	1.000	1200.000	461.107	798.332
7	Water	1.000	15000.000	463.151	813.970

A.3. CASO 3: CICLO RANKINE REGENERATIVO CON CAA ABIERTO

1. Información de la Planta

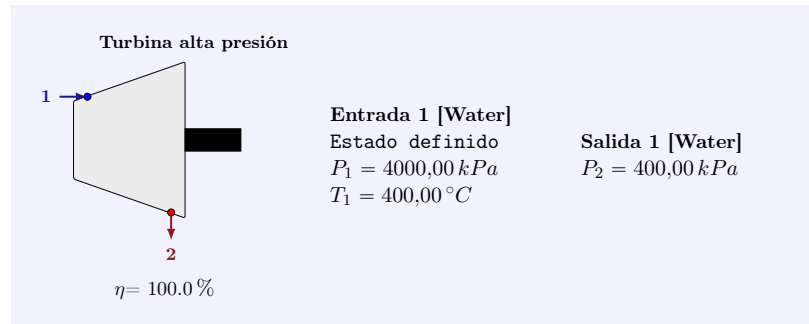
1.1. Diagrama de la Planta

Figura 1: Diagrama general de la planta

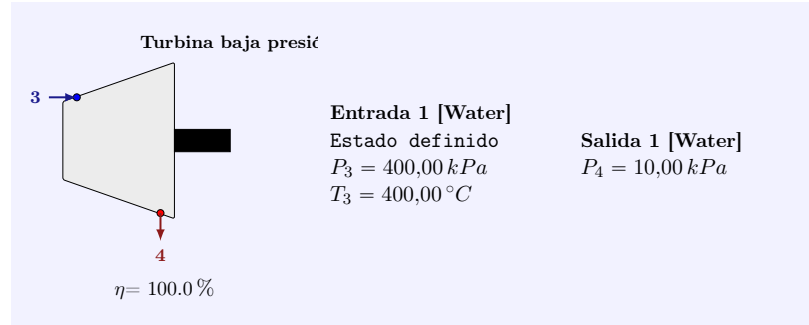


1.2. Análisis de los dispositivos

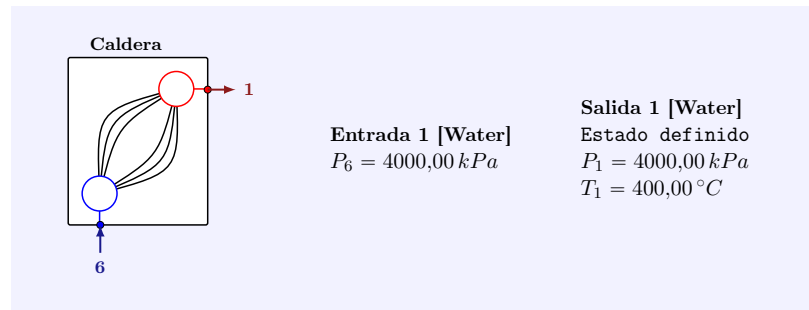
1.2.1. Volúmen de control: Turbina alta presión



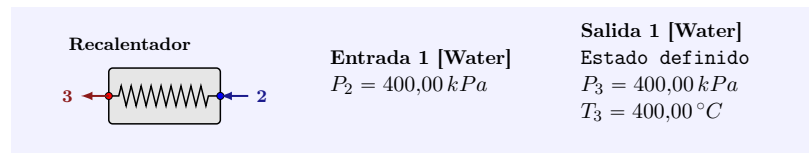
1.2.2. Volúmen de control: Turbina baja presión



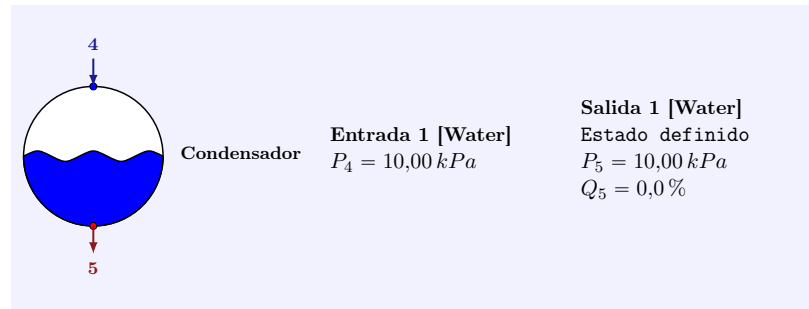
1.2.3. Volúmen de control: Caldera



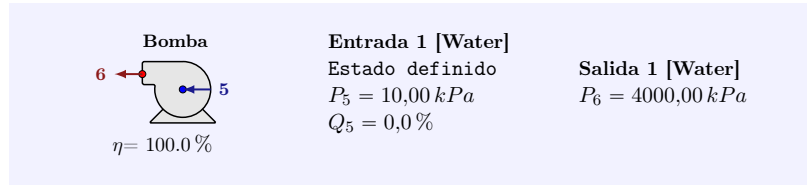
1.2.4. Volúmen de control: Recalentador



1.2.5. Volúmen de control: Condensador



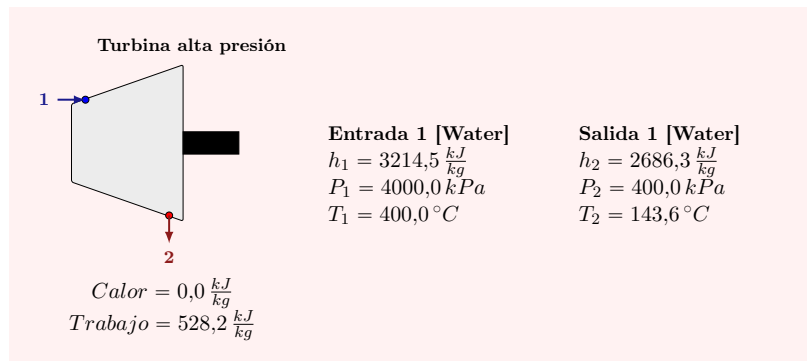
1.2.6. Volúmen de control: Bomba



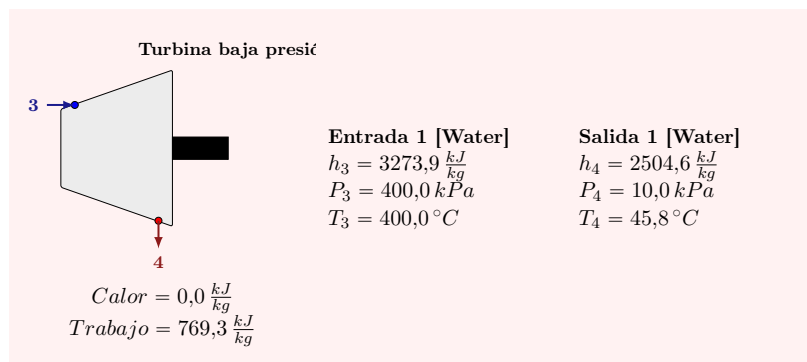
2. Resultados

2.1. Dispositivos

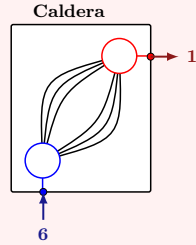
2.1.1. Turbina alta presión



2.1.2. Turbina baja presión



2.1.3. Caldera



Entrada 1 [Water]
 $h_6 = 195,8 \frac{kJ}{kg}$
 $P_6 = 4000,0 kPa$
 $T_6 = 45,9^\circ C$

Salida 1 [Water]
 $h_1 = 3214,5 \frac{kJ}{kg}$
 $P_1 = 4000,0 kPa$
 $T_1 = 400,0^\circ C$

$Calor = 3018,6 \frac{kJ}{kg}$
 $Trabajo = 0,0 \frac{kJ}{kg}$

2.1.4. Recalentador

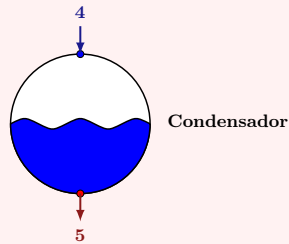


Entrada 1 [Water]
 $h_2 = 2686,3 \frac{kJ}{kg}$
 $P_2 = 400,0 kPa$
 $T_2 = 143,6^\circ C$

Salida 1 [Water]
 $h_3 = 3273,9 \frac{kJ}{kg}$
 $P_3 = 400,0 kPa$
 $T_3 = 400,0^\circ C$

$Calor = 587,6 \frac{kJ}{kg}$
 $Trabajo = 0,0 \frac{kJ}{kg}$

2.1.5. Condensador

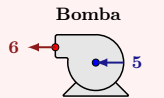


Entrada 1 [Water]
 $h_4 = 2504,6 \frac{kJ}{kg}$
 $P_4 = 10,0 kPa$
 $T_4 = 45,8^\circ C$

Salida 1 [Water]
 $h_5 = 191,8 \frac{kJ}{kg}$
 $P_5 = 10,0 kPa$
 $T_5 = 45,8^\circ C$

$Calor = -2312,8 \frac{kJ}{kg}$
 $Trabajo = 0,0 \frac{kJ}{kg}$

2.1.6. Bomba



Entrada 1 [Water]
 $h_5 = 191,8 \frac{kJ}{kg}$
 $P_5 = 10,0 kPa$
 $T_5 = 45,8^\circ C$

Salida 1 [Water]
 $h_6 = 195,8 \frac{kJ}{kg}$
 $P_6 = 4000,0 kPa$
 $T_6 = 45,9^\circ C$

$Calor = 0,0 \frac{kJ}{kg}$
 $Trabajo = -4,0 \frac{kJ}{kg}$

2.2. Eficiencia de la Planta

$$\eta_{termica} = 35,868\%$$

2.3. Resultados de cada estado

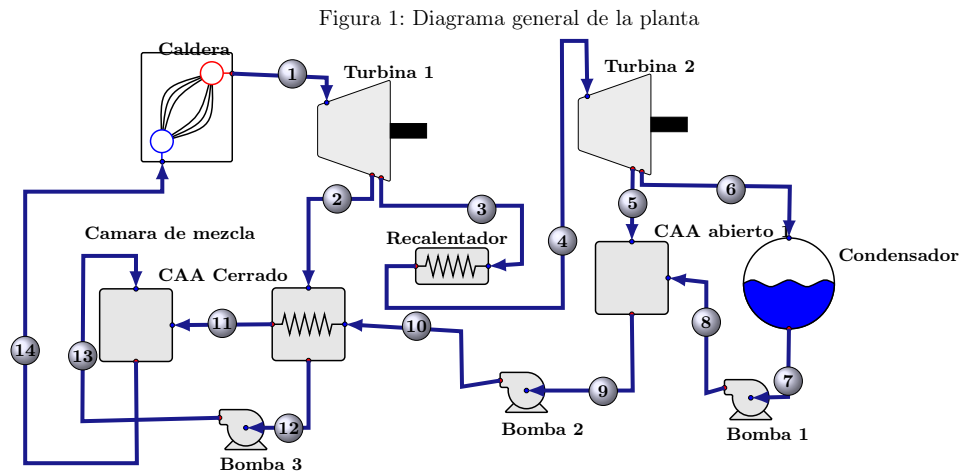
Tabla 1: Resultados de cada estado

Estado	Fluido	y[flujo]	Presión [kPa]	Temperatura [K]	Entalpía [kJ/kg]
1	Water	1.000	4000.000	673.150	3214.473
2	Water	1.000	400.000	416.758	2686.309
3	Water	1.000	400.000	673.150	3273.910
4	Water	1.000	10.000	318.956	2504.572
5	Water	1.000	10.000	318.956	191.803
6	Water	1.000	4000.000	319.088	195.834

A.4. CASO 4: CICLO RANKINE REGENERATIVO CON RECALENTAMIENTO

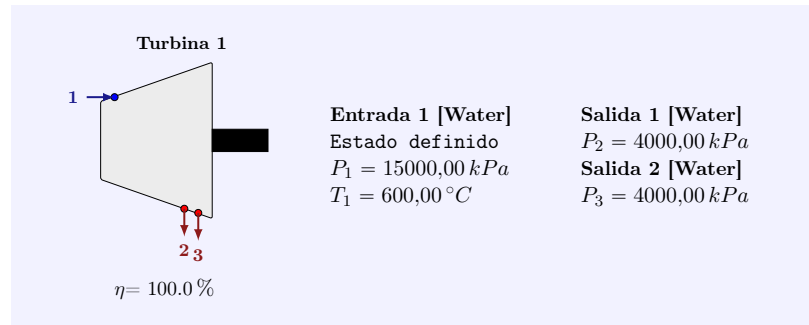
1. Información de la Planta

1.1. Diagrama de la Planta

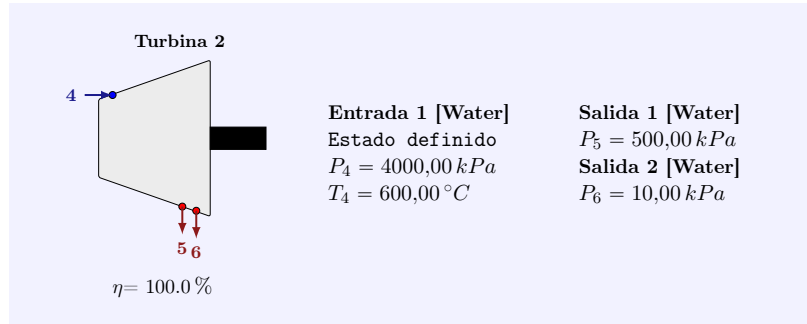


1.2. Análisis de los dispositivos

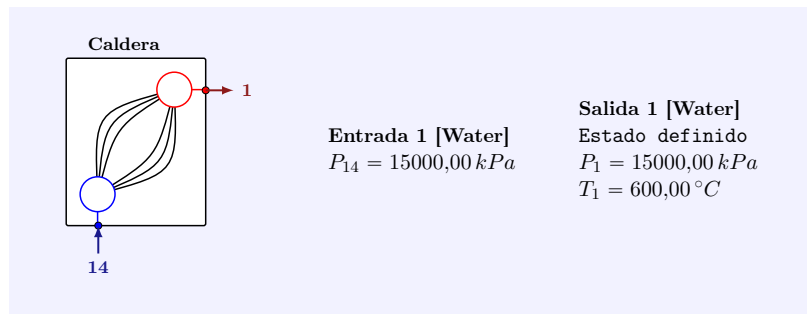
1.2.1. Volúmen de control: Turbina 1



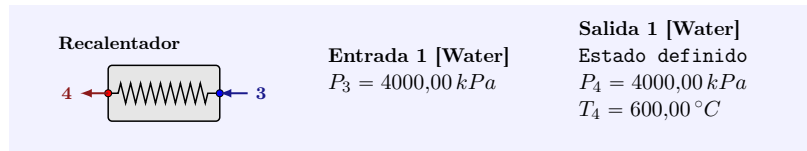
1.2.2. Volúmen de control: Turbina 2



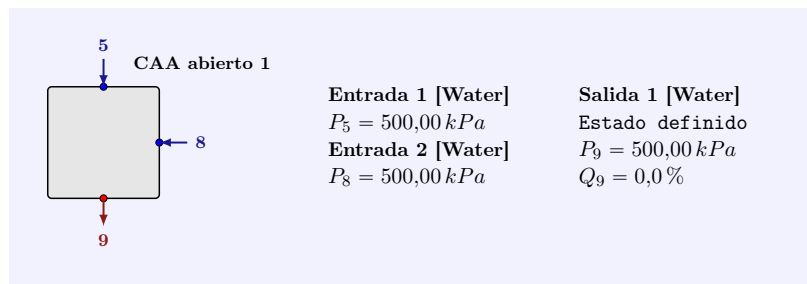
1.2.3. Volúmen de control: Caldera



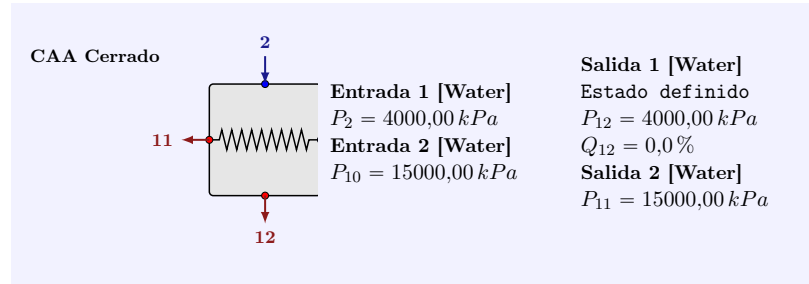
1.2.4. Volúmen de control: Recalentador



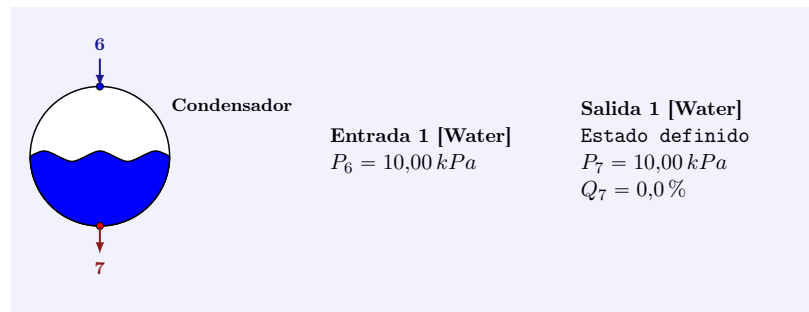
1.2.5. Volúmen de control: CAA abierto 1



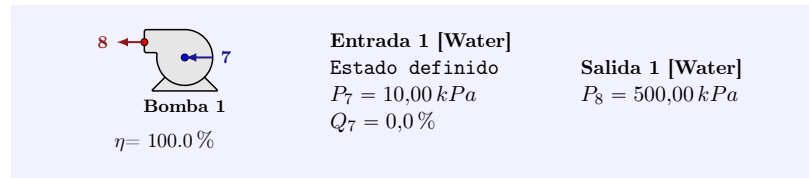
1.2.6. Volúmen de control: CAA Cerrado



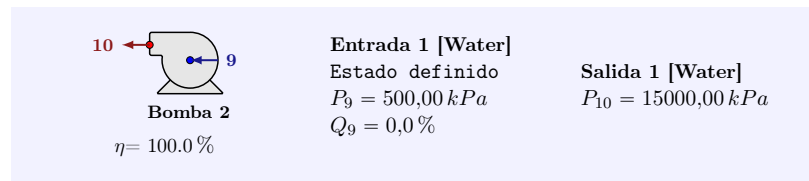
1.2.7. Volúmen de control: Condensador



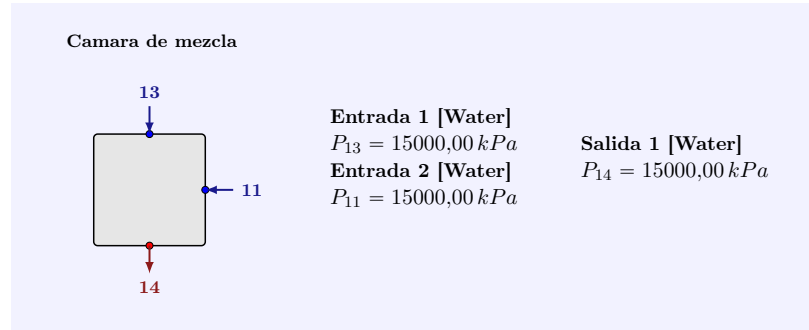
1.2.8. Volúmen de control: Bomba 1



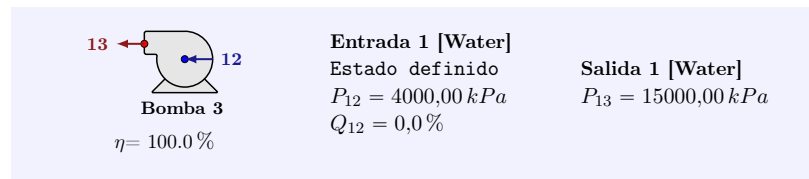
1.2.9. Volúmen de control: Bomba 2



1.2.10. Volúmen de control: Camara de mezcla



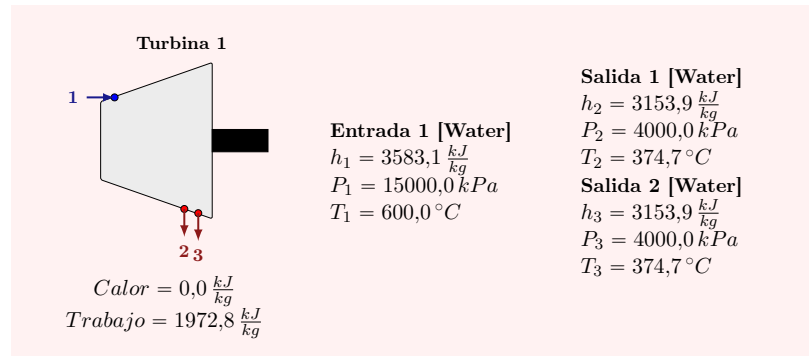
1.2.11. Volúmen de control: Bomba 3



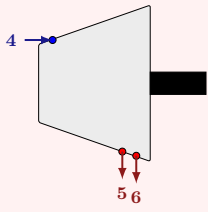
2. Resultados

2.1. Dispositivos

2.1.1. Turbina 1



2.1.2. Turbina 2



Turbina 2

$Calor = 0,0 \frac{kJ}{kg}$
 $Trabajo = -654,2 \frac{kJ}{kg}$

Entrada 1 [Water]

$h_4 = 3674,9 \frac{kJ}{kg}$
 $P_4 = 4000,0 kPa$
 $T_4 = 600,0 ^\circ C$

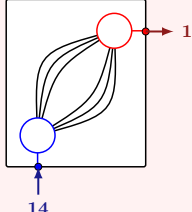
Salida 1 [Water]

$h_5 = 3013,6 \frac{kJ}{kg}$
 $P_5 = 500,0 kPa$
 $T_5 = 275,4 ^\circ C$

Salida 2 [Water]

$h_6 = 2335,6 \frac{kJ}{kg}$
 $P_6 = 10,0 kPa$
 $T_6 = 45,8 ^\circ C$

2.1.3. Caldera



Caldera

$Calor = 2699,1 \frac{kJ}{kg}$
 $Trabajo = 0,0 \frac{kJ}{kg}$


Entrada 1 [Water]

$h_{14} = 1090,1 \frac{kJ}{kg}$
 $P_{14} = 15000,0 kPa$
 $T_{14} = 250,8 ^\circ C$

Salida 1 [Water]

$h_1 = 3583,1 \frac{kJ}{kg}$
 $P_1 = 15000,0 kPa$
 $T_1 = 600,0 ^\circ C$

2.1.4. Recalentador



Recalentador

$Calor = 217,4 \frac{kJ}{kg}$
 $Trabajo = 0,0 \frac{kJ}{kg}$

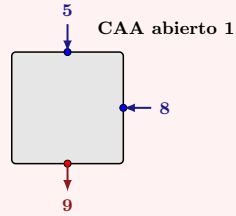
Entrada 1 [Water]

$h_3 = 3153,9 \frac{kJ}{kg}$
 $P_3 = 4000,0 kPa$
 $T_3 = 374,7 ^\circ C$

Salida 1 [Water]

$h_4 = 3674,9 \frac{kJ}{kg}$
 $P_4 = 4000,0 kPa$
 $T_4 = 600,0 ^\circ C$

2.1.5. CAA abierto 1



$$\text{Calor} = 0,0 \frac{\text{kJ}}{\text{kg}}$$

$$\text{Trabajo} = 0,0 \frac{\text{kJ}}{\text{kg}}$$

Entrada 1 [Water]

$$h_5 = 3013,6 \frac{\text{kJ}}{\text{kg}}$$

$$P_5 = 500,0 \text{ kPa}$$

$$T_5 = 275,4^\circ\text{C}$$

Entrada 2 [Water]

$$h_8 = 192,3 \frac{\text{kJ}}{\text{kg}}$$

$$P_8 = 500,0 \text{ kPa}$$

$$T_8 = 45,8^\circ\text{C}$$

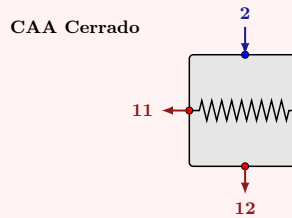
Salida 1 [Water]

$$h_9 = 640,1 \frac{\text{kJ}}{\text{kg}}$$

$$P_9 = 500,0 \text{ kPa}$$

$$T_9 = 151,8^\circ\text{C}$$

2.1.6. CAA Cerrado



$$\text{Calor} = 0,0 \frac{\text{kJ}}{\text{kg}}$$

$$\text{Trabajo} = 0,0 \frac{\text{kJ}}{\text{kg}}$$

Entrada 1 [Water]

$$h_2 = 3153,9 \frac{\text{kJ}}{\text{kg}}$$

$$P_2 = 4000,0 \text{ kPa}$$

$$T_2 = 374,7^\circ\text{C}$$

Entrada 2 [Water]

$$h_{10} = 655,9 \frac{\text{kJ}}{\text{kg}}$$

$$P_{10} = 15000,0 \text{ kPa}$$

$$T_{10} = 153,4^\circ\text{C}$$

Salida 1 [Water]

$$h_{12} = 1087,5 \frac{\text{kJ}}{\text{kg}}$$

$$P_{12} = 4000,0 \text{ kPa}$$

$$T_{12} = 250,4^\circ\text{C}$$

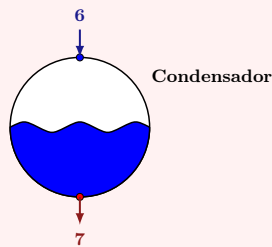
Salida 2 [Water]

$$h_{11} = 1087,8 \frac{\text{kJ}}{\text{kg}}$$

$$P_{11} = 15000,0 \text{ kPa}$$

$$T_{11} = 250,4^\circ\text{C}$$

2.1.7. Condensador



$$\text{Calor} = -1615,1 \frac{\text{kJ}}{\text{kg}}$$

$$\text{Trabajo} = 0,0 \frac{\text{kJ}}{\text{kg}}$$

Entrada 1 [Water]

$$h_6 = 2335,6 \frac{\text{kJ}}{\text{kg}}$$

$$P_6 = 10,0 \text{ kPa}$$

$$T_6 = 45,8^\circ\text{C}$$

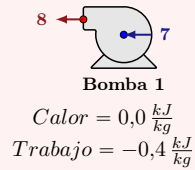
Salida 1 [Water]

$$h_7 = 191,8 \frac{\text{kJ}}{\text{kg}}$$

$$P_7 = 10,0 \text{ kPa}$$

$$T_7 = 45,8^\circ\text{C}$$

2.1.8. Bomba 1



Entrada 1 [Water]

$$h_7 = 191,8 \frac{kJ}{kg}$$

$$P_7 = 10,0 kPa$$

$$T_7 = 45,8^\circ C$$

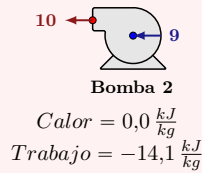
Salida 1 [Water]

$$h_8 = 192,3 \frac{kJ}{kg}$$

$$P_8 = 500,0 kPa$$

$$T_8 = 45,8^\circ C$$

2.1.9. Bomba 2



Entrada 1 [Water]

$$h_9 = 640,1 \frac{kJ}{kg}$$

$$P_9 = 500,0 kPa$$

$$T_9 = 151,8^\circ C$$

Salida 1 [Water]

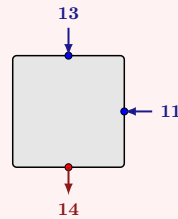
$$h_{10} = 655,9 \frac{kJ}{kg}$$

$$P_{10} = 15000,0 kPa$$

$$T_{10} = 153,4^\circ C$$

2.1.10. Camara de mezcla

Camara de mezcla



$Calor = 0,0 \frac{kJ}{kg}$
 $Trabajo = 0,0 \frac{kJ}{kg}$

Entrada 1 [Water]

$$h_{13} = 1101,1 \frac{kJ}{kg}$$

$$P_{13} = 15000,0 kPa$$

$$T_{13} = 253,2^\circ C$$

Entrada 2 [Water]

$$h_{11} = 1087,8 \frac{kJ}{kg}$$

$$P_{11} = 15000,0 kPa$$

$$T_{11} = 250,4^\circ C$$

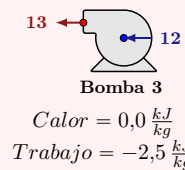
Salida 1 [Water]

$$h_{14} = 1090,1 \frac{kJ}{kg}$$

$$P_{14} = 15000,0 kPa$$

$$T_{14} = 250,8^\circ C$$

2.1.11. Bomba 3



Entrada 1 [Water]

$$h_{12} = 1087,5 \frac{kJ}{kg}$$

$$P_{12} = 4000,0 kPa$$

$$T_{12} = 250,4^\circ C$$

Salida 1 [Water]

$$h_{13} = 1101,1 \frac{kJ}{kg}$$

$$P_{13} = 15000,0 kPa$$

$$T_{13} = 253,2^\circ C$$

2.2. Eficiencia de la Planta

$$\eta_{termica} = 44,624\%$$

2.3. Resultados de cada estado

Tabla 1: Resultados de cada estado

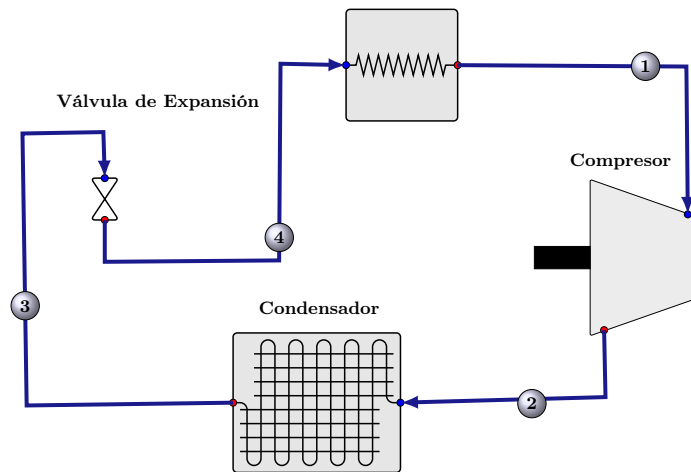
Estado	Fluido	y[flujo]	Presión [kPa]	Temperatura [K]	Entalpía [kJ/kg]
1	Water	1.083	15000.000	873.150	3583.132
2	Water	0.187	4000.000	647.807	3153.875
12	Water	0.187	4000.000	523.504	1087.489
13	Water	0.187	15000.000	526.302	1101.098
11	Water	0.895	15000.000	523.504	1087.798
14	Water	1.083	15000.000	523.989	1090.098
3	Water	0.417	4000.000	647.807	3153.875
4	Water	0.417	4000.000	873.150	3674.868
5	Water	0.142	500.000	548.501	3013.590
8	Water	0.753	500.000	318.973	192.301
9	Water	0.895	500.000	424.981	640.087
10	Water	0.895	15000.000	426.572	655.868
6	Water	0.753	10.000	318.956	2335.618
7	Water	0.753	10.000	318.956	191.803

A.5. CASO 5: REFRIGERACIÓN POR COMPRESIÓN

1. Información de la Planta

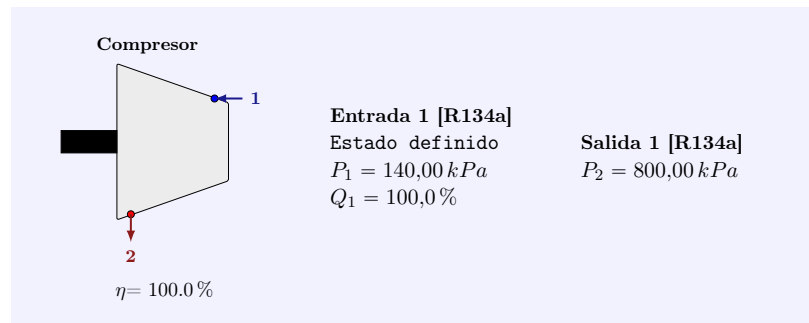
1.1. Diagrama de la Planta

Figura 1: Diagrama general de la planta
Evaporador

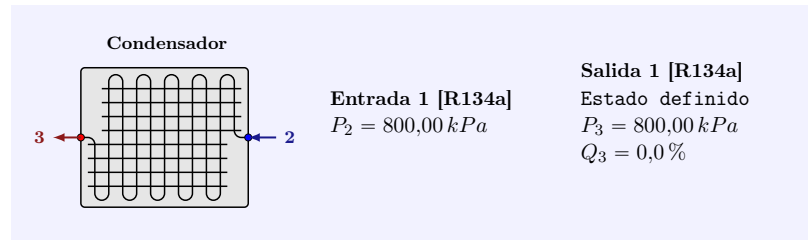


1.2. Análisis de los dispositivos

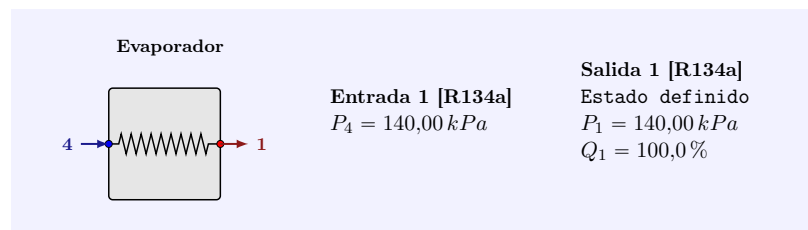
1.2.1. Volúmen de control: Compresor



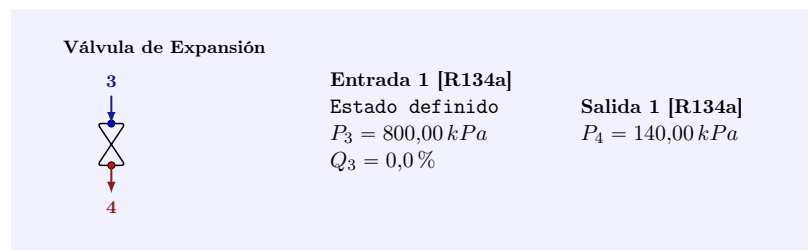
1.2.2. Volúmen de control: Condensador



1.2.3. Volúmen de control: Evaporador



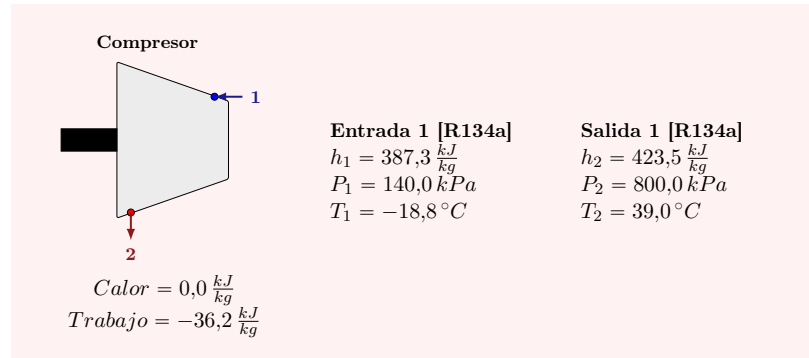
1.2.4. Volúmen de control: Válvula de Expansión



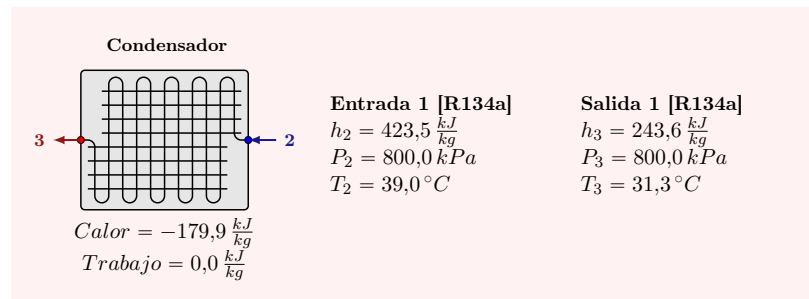
2. Resultados

2.1. Dispositivos

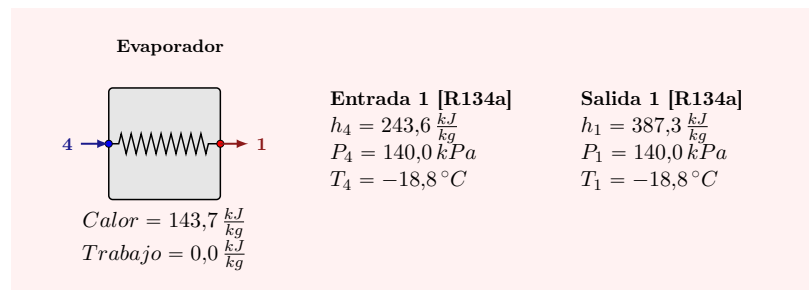
2.1.1. Compresor



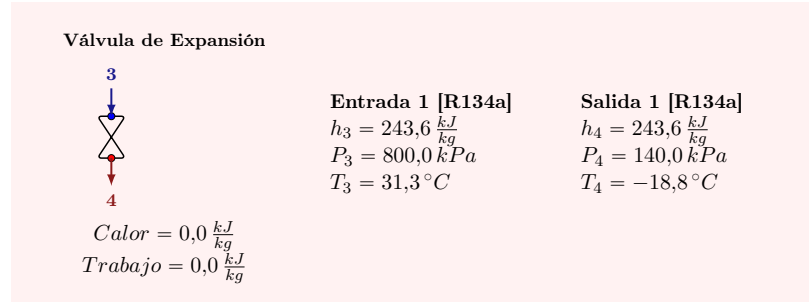
2.1.2. Condensador



2.1.3. Evaporador



2.1.4. Válvula de Expansión



2.2. Eficiencia de la Planta

$$\beta_{Refrigerador} = 3,968$$

$$\beta_{Bomba \text{ Calor}} = 4,968$$

2.3. Resultados de cada estado

Tabla 1: Resultados de cada estado

Estado	Fluido	y[flujo]	Presión [kPa]	Temperatura [K]	Entalpía [kJ/kg]
1	R134a	1.000	140.000	254.390	387.320
2	R134a	1.000	800.000	312.127	423.532
3	R134a	1.000	800.000	304.477	243.645
4	R134a	1.000	140.000	254.390	243.645

C. MANUAL DE USUARIO DEL SOFTWARE

Manual de usuario del software

Jorge Pimienta

Índice

1. Descripción de la interfaz	1
2. Simulación de un sistema	3

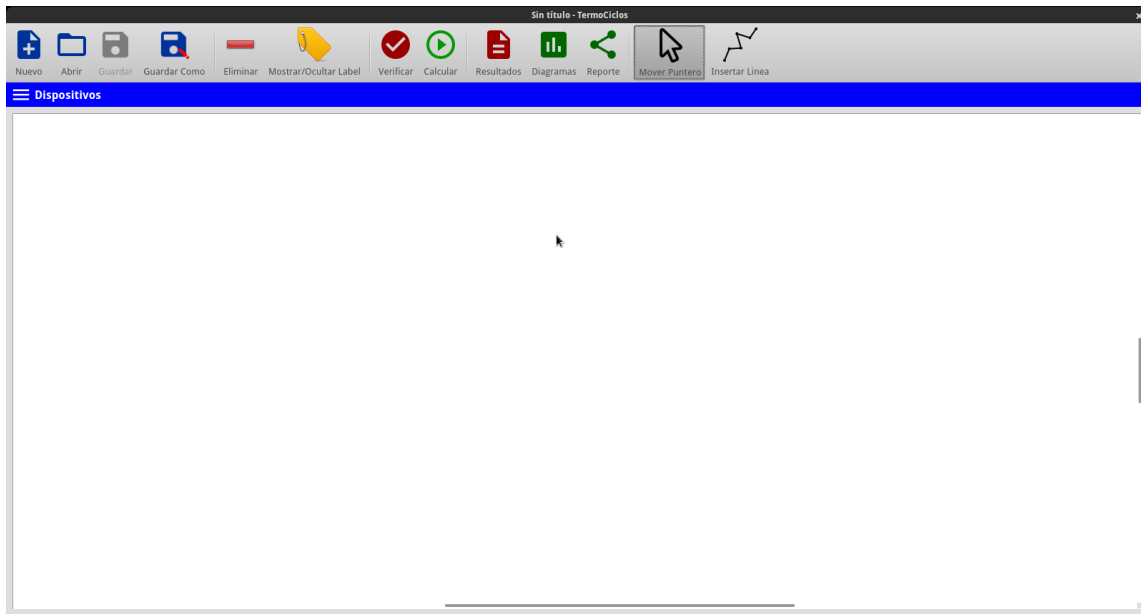
1. Descripción de la interfaz

Al ejecutar el software, emerge la ventana principal de este mostrada en la figura 1. en ella se aprecian en la parte superior, con múltiples opciones, la barra de herramientas. Debajo de esta, se encuentra una barra de color azul con un botón llamado dispositivos, que permite mostrar y ocultar el toolbox de dispositivos termodinámicos.

En la figura 2 se muestra de manera más detallada, la barra de herramientas. A continuación se describen sus botones:

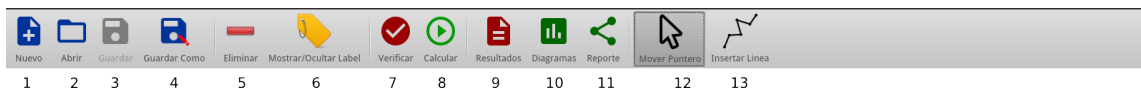
1. Crea un nuevo documento.
2. Abre un documento creado con anterioridad.
3. Guarda el documento que se encuentra abierto.
4. Guarda el documento que se encuentra abierto con otro nombre.
5. Elimina el dispositivo o la conexión seleccionada.
6. Muestra u oculta las etiquetas de los dispositivos.
7. Verifica que el sistema a simular sea coherente.
8. Corre la simulación del sistema.

Figura 1: Ventana Principal del Software



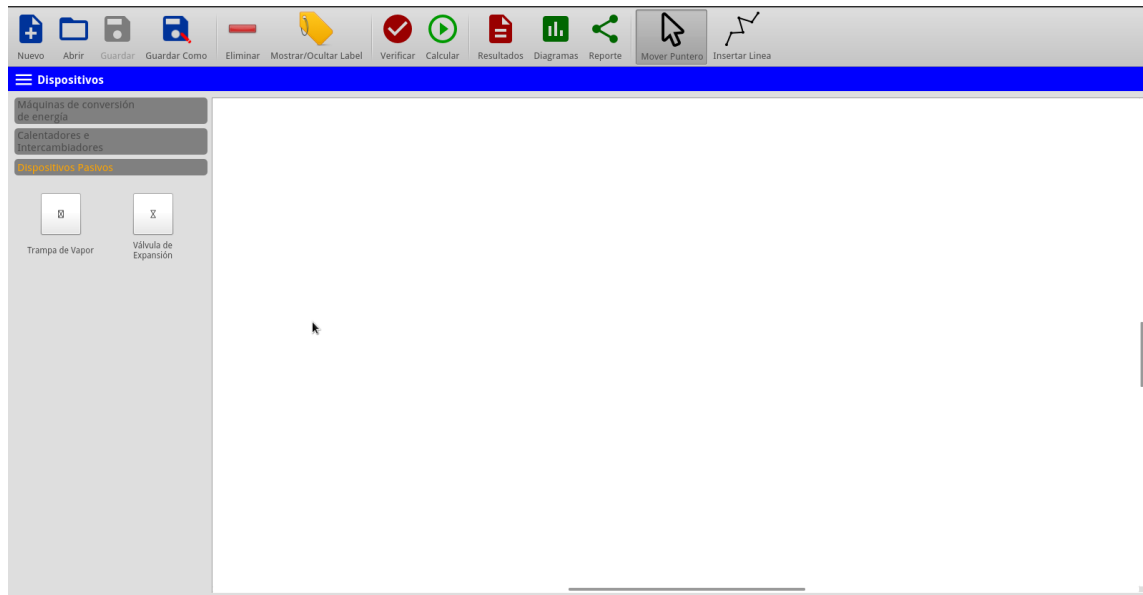
9. Muestra en una ventana emergente, los resultados de la simulación.
10. Dibuja en una ventana emergente, los gráficos del ciclo termodinámico del sistema.
11. Genera y muestra el reporte de la simulación.
12. Activa el modo puntero para editar los dispositivos.
13. Activa el modo linea para editar las conexiones entre dispositivos.

Figura 2: Ventana principal con el toolbox de dispositivos desplegado



La figura 3 muestra la ventana digital con el toolbox de dispositivos desplegado.

Figura 3: ventana digital con el toolbox de dispositivos desplegado



2. Simulación de un sistema

A continuación se presentan en las siguientes figuras la secuencia para insertar dispositivos, conectarlos, asignar propiedades y llegar a un sistema terminado.

Figura 4: Inserción de un dispositivo

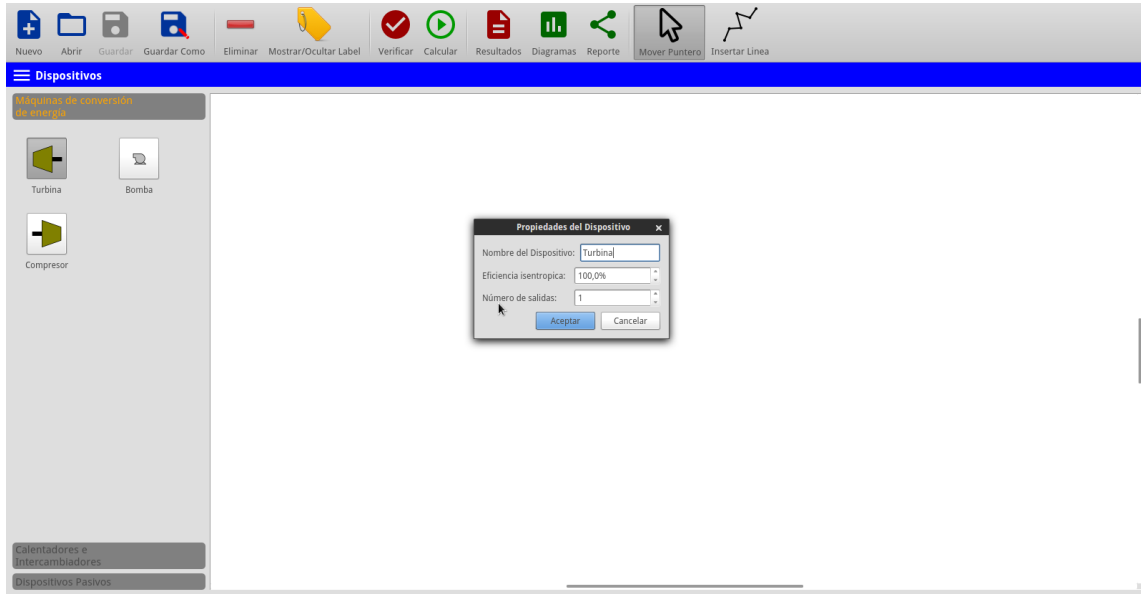


Figura 5: Conexión entre dispositivos

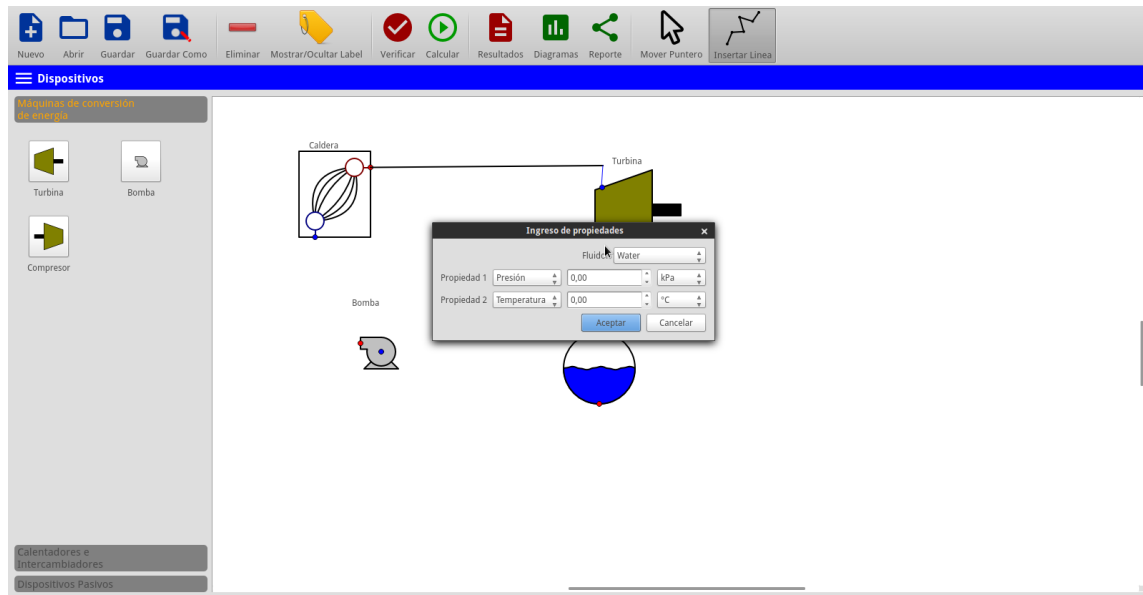


Figura 6: Propiedades al conectar dispositivos

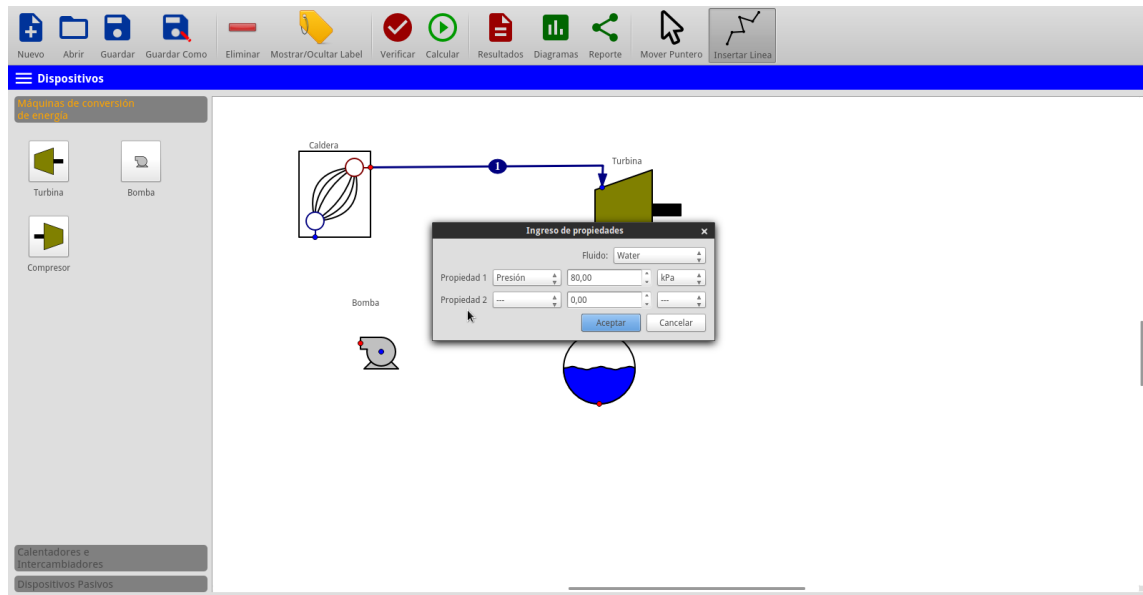
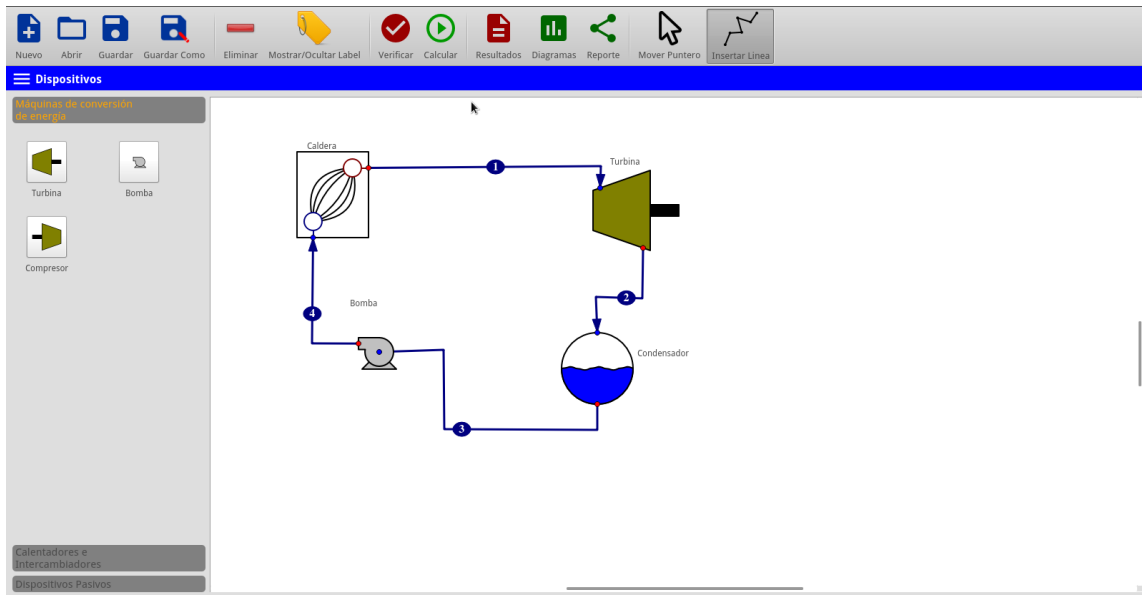


Figura 7: Sistema terminado



D. DOCUMENTACIÓN DEL SOFTWARE

TermoCiclos

Generado por Doxygen 1.8.9.1

Índice general

1	Índice jerárquico	1
1.1	Jerarquía de la clase	1
2	Índice de estructura de datos	3
2.1	Estructura de datos	3
3	Índice de archivos	5
3.1	Lista de archivos	5
4	Documentación de las estructuras de datos	7
4.1	Referencia de la Clase Conector	7
4.1.1	Documentación de las enumeraciones miembro de la clase	8
4.1.1.1	anonymous enum	8
4.1.1.2	TipoConector	8
4.1.2	Documentación del constructor y destructor	8
4.1.2.1	Conector	8
4.1.3	Documentación de las funciones miembro	8
4.1.3.1	ajustarId	8
4.1.3.2	getColorFondo	8
4.1.3.3	getColorLinea	8
4.1.3.4	hoverEnterEvent	8
4.1.3.5	hoverLeaveEvent	8
4.1.3.6	obtenerId	8
4.1.3.7	obtenerTipo	8
4.1.3.8	paint	8
4.1.3.9	setColorFondo	8
4.1.3.10	setColorLinea	8
4.1.3.11	type	8
4.2	Referencia de la Clase ConexionDispositivosItem	8
4.2.1	Documentación de las enumeraciones miembro de la clase	9
4.2.1.1	anonymous enum	9
4.2.2	Documentación del constructor y destructor	10

4.2.2.1	ConexionDispositivosItem	10
4.2.3	Documentación de las funciones miembro	10
4.2.3.1	ajustarDatos	10
4.2.3.2	ajustarId	10
4.2.3.3	ajustarPosEtiqueta	10
4.2.3.4	ajustarVerificado	10
4.2.3.5	boundingRect	10
4.2.3.6	esEtiquetaVisible	10
4.2.3.7	estaVerificado	10
4.2.3.8	itemFinal	10
4.2.3.9	itemInicio	10
4.2.3.10	mostrarEtiqueta	10
4.2.3.11	obtenerDatos	10
4.2.3.12	obtenerFinal	10
4.2.3.13	obtenerId	10
4.2.3.14	obtenerInicio	10
4.2.3.15	obtenerPosEtiqueta	10
4.2.3.16	paint	10
4.2.3.17	shape	10
4.2.3.18	type	10
4.2.3.19	updatePosition	10
4.3	Referencia de la Estructura ConexionDispositivosItem::datosConexion	10
4.3.1	Documentación de los campos	11
4.3.1.1	enableFluido	11
4.3.1.2	enableProp1	11
4.3.1.3	enableProp2	11
4.3.1.4	miFluido	11
4.3.1.5	miProp1	11
4.3.1.6	miProp2	11
4.3.1.7	miTipoProp1	11
4.3.1.8	miTipoProp2	11
4.3.1.9	miUniProp1	11
4.3.1.10	miUniProp2	11
4.4	Referencia de la Estructura DispositivoItem::datosDispositivo	11
4.4.1	Documentación de los campos	12
4.4.1.1	miEficiencia	12
4.4.1.2	miNombre	12
4.4.1.3	miNombreXML	12
4.4.1.4	nPuertosSalida	12
4.4.1.5	tipo	12

4.5	Referencia de la Clase DiagramaScene	12
4.5.1	Documentación de las enumeraciones miembro de la clase	13
4.5.1.1	Modo	13
4.5.2	Documentación del constructor y destructor	13
4.5.2.1	DiagramaScene	13
4.5.3	Documentación de las funciones miembro	13
4.5.3.1	actulizarVinculosIds	13
4.5.3.2	ajustarModo	13
4.5.3.3	ajustarTipoltem	13
4.5.3.4	documentoModificado	13
4.5.3.5	itemInsertado	13
4.5.3.6	itemSeleccionado	13
4.5.3.7	mouseDoubleClickEvent	13
4.5.3.8	mouseMoveEvent	13
4.5.3.9	mousePressEvent	13
4.5.3.10	obtenerMiMenuItemem	13
4.6	Referencia de la Clase DispositivoDialogo	13
4.6.1	Documentación del constructor y destructor	14
4.6.1.1	DispositivoDialogo	14
4.6.2	Documentación de las funciones miembro	14
4.6.2.1	crearWidgets	14
4.6.2.2	deshabilitarNsalidas	14
4.6.2.3	obtenerDatos	14
4.7	Referencia de la Clase DispositivoItem	14
4.7.1	Documentación de las enumeraciones miembro de la clase	15
4.7.1.1	anonymous enum	15
4.7.1.2	TipoDispositivo	15
4.7.2	Documentación del constructor y destructor	16
4.7.2.1	DispositivoItem	16
4.7.3	Documentación de las funciones miembro	16
4.7.3.1	agregarConexion	16
4.7.3.2	ajustarDatos	16
4.7.3.3	ajustarId	16
4.7.3.4	ajustarPosEtiqueta	16
4.7.3.5	ajustarTexto	16
4.7.3.6	contextMenuEvent	16
4.7.3.7	crearImagen	16
4.7.3.8	eliminarConexion	16
4.7.3.9	eliminarConexiones	16
4.7.3.10	esEtiquetaVisible	16

4.7.3.11	imagen	16
4.7.3.12	itemChange	16
4.7.3.13	mostrarEtiqueta	16
4.7.3.14	obtenerConexiones	16
4.7.3.15	obtenerDatos	16
4.7.3.16	obtenerId	16
4.7.3.17	obtenerPolygono	16
4.7.3.18	obtenerPosEtiqueta	16
4.7.3.19	obtenerPtrEntradaCerrado	16
4.7.3.20	obtenerPtrSalidaCerrado	16
4.7.3.21	tipoDispositivo	17
4.7.3.22	type	17
4.8	Referencia de la Clase LabelConexionDispositivosItem	17
4.8.1	Documentación de las enumeraciones miembro de la clase	17
4.8.1.1	anonymous enum	17
4.8.2	Documentación del constructor y destructor	17
4.8.2.1	LabelConexionDispositivosItem	17
4.8.3	Documentación de las funciones miembro	17
4.8.3.1	ajustarTexto	17
4.8.3.2	itemChange	18
4.8.3.3	type	18
4.9	Referencia de la Clase PropiedadesDialogo	18
4.9.1	Documentación del constructor y destructor	18
4.9.1.1	PropiedadesDialogo	18
4.9.2	Documentación de las funciones miembro	18
4.9.2.1	actualizadoCampo1	18
4.9.2.2	actualizadoCampo2	18
4.9.2.3	configurarWidgets	18
4.9.2.4	crearWidgets	18
4.9.2.5	obtenerDatos	18
4.10	Referencia de la Clase TermoCiclos	19
4.10.1	Documentación del constructor y destructor	19
4.10.1.1	TermoCiclos	19
4.10.1.2	~TermoCiclos	19
4.10.2	Documentación de las funciones miembro	19
4.10.2.1	closeEvent	19
5	Documentación de archivos	21
5.1	Referencia del Archivo conector.cpp	21
5.2	Referencia del Archivo conector.h	21

5.3	Referencia del Archivo conexiondispositivositem.cpp	21
5.3.1	Documentación de las variables	22
5.3.1.1	Pi	22
5.4	Referencia del Archivo conexiondispositivositem.h	22
5.5	Referencia del Archivo diagramascene.cpp	22
5.6	Referencia del Archivo diagramascene.h	22
5.7	Referencia del Archivo dispositivodialogo.cpp	22
5.8	Referencia del Archivo dispositivodialogo.h	23
5.9	Referencia del Archivo dispositivoitem.cpp	23
5.10	Referencia del Archivo dispositivoitem.h	23
5.11	Referencia del Archivo main.cpp	23
5.11.1	Documentación de las funciones	24
5.11.1.1	main	24
5.12	Referencia del Archivo propiedadesdialogo.cpp	24
5.13	Referencia del Archivo propiedadesdialogo.h	24
5.14	Referencia del Archivo termociclos.cpp	24
5.15	Referencia del Archivo termociclos.h	25
5.16	Referencia del Archivo test.cpp	25
5.16.1	Documentación de las funciones	25
5.16.1.1	test	25
Índice		27

Capítulo 1

Índice jerárquico

1.1. Jerarquía de la clase

Esta lista de herencias esta ordenada aproximadamente por orden alfabético:

ConexionDispositivosItem::datosConexion	10
DispositivoItem::datosDispositivo	11
QDialog	
DispositivoDialogo	13
PropiedadesDialogo	18
QGraphicsEllipseItem	
Conector	7
LabelConexionDispositivosItem	17
QGraphicsPathItem	
ConexionDispositivosItem	8
QGraphicsPixmapItem	
DispositivoItem	14
QGraphicsScene	
DiagramaScene	12
QMainWindow	
TermoCiclos	19

Capítulo 2

Índice de estructura de datos

2.1. Estructura de datos

Lista de estructuras con una breve descripción:

Conector	7
ConexionDispositivosItem	8
ConexionDispositivosItem::datosConexion	10
DispositivoItem::datosDispositivo	11
DiagramaScene	12
DispositivoDialogo	13
DispositivoItem	14
LabelConexionDispositivosItem	17
PropiedadesDialogo	18
TermoCiclos	19

Capítulo 3

Índice de archivos

3.1. Lista de archivos

Lista de todos los archivos con descripciones breves:

conector.cpp	21
conector.h	21
conexiondispositivositem.cpp	21
conexiondispositivositem.h	22
diagramascene.cpp	22
diagramascene.h	22
dispositivodialogo.cpp	22
dispositivodialogo.h	23
dispositivoitem.cpp	23
dispositivoitem.h	23
main.cpp	23
propiedadesdialogo.cpp	24
propiedadesdialogo.h	24
termociclos.cpp	24
termociclos.h	25
test.cpp	25

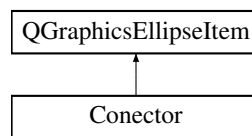
Capítulo 4

Documentación de las estructuras de datos

4.1. Referencia de la Clase Conector

```
#include <conector.h>
```

Diagrama de herencias de Conector



Tipos públicos

- enum { `Type` = UserType + 20 }
- enum `TipoConector` { `entrada`, `salida` }

Métodos públicos

- `Conector` (`DispositivoItem` *item, `TipoConector` tipo, `QPointF` pos, `QGraphicsItem` *parent=0)
- int `type` () const
- void `setColorLinea` (`QPen` color)
- void `setColorFondo` (`QBrush` color)
- `TipoConector` `obtenerTipo` ()
- `QPen` `getColorLinea` ()
- `QBrush` `getColorFondo` ()
- void `ajustarId` (int id)
- int `obtenerId` ()

Métodos protegidos

- void `hoverEnterEvent` (`QGraphicsSceneHoverEvent` *event)
- void `hoverLeaveEvent` (`QGraphicsSceneHoverEvent` *event)
- void `paint` (`QPainter` *painter, const `QStyleOptionGraphicsItem` *option, `QWidget` *widget)

4.1.1. Documentación de las enumeraciones miembro de la clase

4.1.1.1. anonymous enum

Valores de enumeraciones

Type

4.1.1.2. enum Conector::TipoConector

Valores de enumeraciones

entrada

salida

4.1.2. Documentación del constructor y destructor

4.1.2.1. `Conector::Conector (DispositivoItem * item, TipoConector tipo, QPointF pos, QGraphicsItem * parent = 0)`
[explicit]

4.1.3. Documentación de las funciones miembro

4.1.3.1. `void Conector::ajustarId (int id)`

4.1.3.2. `QBrush Conector::getColorFondo ()`

4.1.3.3. `QPen Conector::getColorLinea ()`

4.1.3.4. `void Conector::hoverEnterEvent (QGraphicsSceneHoverEvent * event)` [protected]

4.1.3.5. `void Conector::hoverLeaveEvent (QGraphicsSceneHoverEvent * event)` [protected]

4.1.3.6. `int Conector::obtenerId ()` [inline]

4.1.3.7. `TipoConector Conector::obtenerTipo ()` [inline]

4.1.3.8. `void Conector::paint (QPainter * painter, const QStyleOptionGraphicsItem * option, QWidget * widget)`
[protected]

4.1.3.9. `void Conector::setColorFondo (QBrush color)`

4.1.3.10. `void Conector::setColorLinea (QPen color)`

4.1.3.11. `int Conector::type () const` [inline]

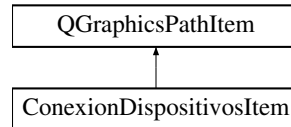
La documentación para esta clase fue generada a partir de los siguientes ficheros:

- [conector.h](#)
- [conector.cpp](#)

4.2. Referencia de la Clase ConexionDispositivosItem

```
#include <conexiondispositivositem.h>
```

Diagrama de herencias de ConexionDispositivosItem



Estructuras de datos

- struct [datosConexion](#)

Tipos públicos

- enum { [Type](#) = UserType + 4 }

Métodos públicos

- [ConexionDispositivosItem](#) ([Conector](#) *itemInicio, [Conector](#) *itemFinal, QPainterPath ruta, [datosConexion](#) datos, QGraphicsItem *parent=0)
- int [type](#) () const
- QRectF [boundingRect](#) () const
- QPainterPath [shape](#) () const
- [Conector](#) * [itemInicio](#) () const
- [Conector](#) * [itemFinal](#) () const
- void [updatePosition](#) ()
- [datosConexion](#) [obtenerDatos](#) ()
- void [ajustarDatos](#) ([datosConexion](#) datos)
- void [ajustarId](#) (int id)
- int [obtenerId](#) ()
- [Conector](#) * [obtenerInicio](#) () const
- [Conector](#) * [obtenerFinal](#) () const
- bool [estaVerificado](#) ()
- void [ajustarVerificado](#) (bool estado)
- bool [esEtiquetaVisible](#) ()
- void [mostrarEtiqueta](#) (bool estado)
- QPointF [obtenerPosEtiqueta](#) ()
- void [ajustarPosEtiqueta](#) (QPointF punto)

Métodos protegidos

- void [paint](#) (QPainter *painter, const QStyleOptionGraphicsItem *option, QWidget *widget=0)

4.2.1. Documentación de las enumeraciones miembro de la clase

4.2.1.1. anonymous enum

Valores de enumeraciones

Type

4.2.2. Documentación del constructor y destructor

4.2.2.1. `ConexionDispositivosItem::ConexionDispositivosItem (Conector * itemInicio, Conector * itemFinal, QPainterPath ruta, datosConexion datos, QGraphicsItem * parent = 0) [explicit]`

4.2.3. Documentación de las funciones miembro

4.2.3.1. `void ConexionDispositivosItem::ajustarDatos (datosConexion datos)`

4.2.3.2. `void ConexionDispositivosItem::ajustarId (int id)`

4.2.3.3. `void ConexionDispositivosItem::ajustarPosEtiqueta (QPointF punto) [inline]`

4.2.3.4. `void ConexionDispositivosItem::ajustarVerificado (bool estado) [inline]`

4.2.3.5. `QRectF ConexionDispositivosItem::boundingRect () const`

4.2.3.6. `bool ConexionDispositivosItem::esEtiquetaVisible () [inline]`

4.2.3.7. `bool ConexionDispositivosItem::estaVerificado () [inline]`

4.2.3.8. `Conector* ConexionDispositivosItem::itemFinal () const [inline]`

4.2.3.9. `Conector* ConexionDispositivosItem::itemInicio () const [inline]`

4.2.3.10. `void ConexionDispositivosItem::mostrarEtiqueta (bool estado) [inline]`

4.2.3.11. `ConexionDispositivosItem::datosConexion ConexionDispositivosItem::obtenerDatos ()`

4.2.3.12. `Conector * ConexionDispositivosItem::obtenerFinal () const`

4.2.3.13. `int ConexionDispositivosItem::obtenerId () [inline]`

4.2.3.14. `Conector * ConexionDispositivosItem::obtenerInicio () const`

4.2.3.15. `QPointF ConexionDispositivosItem::obtenerPosEtiqueta () [inline]`

4.2.3.16. `void ConexionDispositivosItem::paint (QPainter * painter, const QStyleOptionGraphicsItem * option, QWidget * widget = 0) [protected]`

4.2.3.17. `QPainterPath ConexionDispositivosItem::shape () const`

4.2.3.18. `int ConexionDispositivosItem::type () const [inline]`

4.2.3.19. `void ConexionDispositivosItem::updatePosition ()`

La documentación para esta clase fue generada a partir de los siguientes ficheros:

- [conexiondispositivositem.h](#)
- [conexiondispositivositem.cpp](#)

4.3. Referencia de la Estructura ConexionDispositivosItem::datosConexion

```
#include <conexiondispositivositem.h>
```

Campos de datos

- QString [miFluido](#)
- QString [miTipoProp1](#)
- QString [miTipoProp2](#)
- QString [miUniProp1](#)
- QString [miUniProp2](#)
- double [miProp1](#)
- double [miProp2](#)
- bool [enableProp1](#)
- bool [enableProp2](#)
- bool [enableFluido](#)

4.3.1. Documentación de los campos

4.3.1.1. bool ConexionDispositivosItem::datosConexion::enableFluido

4.3.1.2. bool ConexionDispositivosItem::datosConexion::enableProp1

4.3.1.3. bool ConexionDispositivosItem::datosConexion::enableProp2

4.3.1.4. QString ConexionDispositivosItem::datosConexion::miFluido

4.3.1.5. double ConexionDispositivosItem::datosConexion::miProp1

4.3.1.6. double ConexionDispositivosItem::datosConexion::miProp2

4.3.1.7. QString ConexionDispositivosItem::datosConexion::miTipoProp1

4.3.1.8. QString ConexionDispositivosItem::datosConexion::miTipoProp2

4.3.1.9. QString ConexionDispositivosItem::datosConexion::miUniProp1

4.3.1.10. QString ConexionDispositivosItem::datosConexion::miUniProp2

La documentación para esta estructura fue generada a partir del siguiente fichero:

- [conexiondispositivositem.h](#)

4.4. Referencia de la Estructura DispositivoItem::datosDispositivo

```
#include <dispositivoitem.h>
```

Campos de datos

- TipoDispositivo [tipo](#)
- QString [miNombre](#)
- QString [miNombreXML](#)
- qreal [miEficiencia](#)
- int [nPuertosSalida](#)

4.4.1. Documentación de los campos

- 4.4.1.1. qreal DispositivoItem::datosDispositivo::miEficiencia
- 4.4.1.2. QString DispositivoItem::datosDispositivo::miNombre
- 4.4.1.3. QString DispositivoItem::datosDispositivo::miNombreXML
- 4.4.1.4. int DispositivoItem::datosDispositivo::nPuertosSalida
- 4.4.1.5. TipoDispositivo DispositivoItem::datosDispositivo::tipo

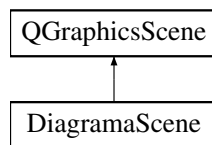
La documentación para esta estructura fue generada a partir del siguiente fichero:

- [dispositivoitem.h](#)

4.5. Referencia de la Clase DiagramaScene

```
#include <diagramascene.h>
```

Diagrama de herencias de DiagramaScene



Tipos públicos

- enum [Modo](#) { [InsertarItem](#), [InsertarLinea](#), [MoverItem](#) }

Slots públicos

- void [ajustarModo](#) ([Modo](#) modo)
- void [ajustarTipoItem](#) ([DispositivoItem::TipoDispositivo](#) tipo)

Señales

- void [itemInsertado](#) ([DispositivoItem](#) *item)
- void [itemSeleccionado](#) ([QGraphicsItem](#) *item)
- void [documentoModificado](#) ()

Métodos públicos

- [DiagramaScene](#) ([QMenu](#) *itemMenu, [QObject](#) *parent=0)
- [QMenu](#) * [obtenerMiMenuItem](#) () const
- void [actualizarVinculosIds](#) ()

Métodos protegidos

- void [mousePressEvent](#) (QGraphicsSceneMouseEvent *mouseEvent)
- void [mouseMoveEvent](#) (QGraphicsSceneMouseEvent *mouseEvent)
- void [mouseDoubleClickEvent](#) (QGraphicsSceneMouseEvent *mouseEvent)

4.5.1. Documentación de las enumeraciones miembro de la clase

4.5.1.1. enum DiagramaScene::Modo

Valores de enumeraciones

InsertarItem

InsertarLinea

MoverItem

4.5.2. Documentación del constructor y destructor

4.5.2.1. DiagramaScene::DiagramaScene (QMenu * *itemMenu*, QObject * *parent* = 0) [explicit]

4.5.3. Documentación de las funciones miembro

4.5.3.1. void DiagramaScene::actualizarVinculosIds ()

4.5.3.2. void DiagramaScene::ajustarModo (Modo *modo*) [slot]

4.5.3.3. void DiagramaScene::ajustarTipItem (DispositivoItem::TipoDispositivo *tipo*) [slot]

4.5.3.4. void DiagramaScene::documentoModificado () [signal]

4.5.3.5. void DiagramaScene::itemInsertado (DispositivoItem * *item*) [signal]

4.5.3.6. void DiagramaScene::itemSeleccionado (QGraphicsItem * *item*) [signal]

4.5.3.7. void DiagramaScene::mouseDoubleClickEvent (QGraphicsSceneMouseEvent * *mouseEvent*) [protected]

4.5.3.8. void DiagramaScene::mouseMoveEvent (QGraphicsSceneMouseEvent * *mouseEvent*) [protected]

4.5.3.9. void DiagramaScene::mousePressEvent (QGraphicsSceneMouseEvent * *mouseEvent*) [protected]

4.5.3.10. QMenu* DiagramaScene::obtenerMiMenuItem () const [inline]

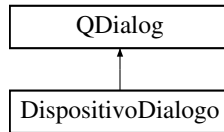
La documentación para esta clase fue generada a partir de los siguientes ficheros:

- [diagramascene.h](#)
- [diagramascene.cpp](#)

4.6. Referencia de la Clase DispositivoDialogo

```
#include <dispositivodialogo.h>
```

Diagrama de herencias de DispositivoDialogo



Métodos públicos

- [DispositivoDialogo](#) ([DispositivoItem::datosDispositivo](#) datos, `QWidget *parent=0`)
- `void crearWidgets ()`
- [DispositivoItem::datosDispositivo](#) `obtenerDatos ()`
- `void deshabilitarNsalidas ()`

4.6.1. Documentación del constructor y destructor

4.6.1.1. `DispositivoDialogo::DispositivoDialogo (DispositivoItem::datosDispositivo datos, QWidget * parent = 0)`
`[explicit]`

4.6.2. Documentación de las funciones miembro

4.6.2.1. `void DispositivoDialogo::crearWidgets ()`

4.6.2.2. `void DispositivoDialogo::deshabilitarNsalidas ()`

4.6.2.3. `DispositivoItem::datosDispositivo DispositivoDialogo::obtenerDatos ()`

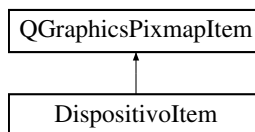
La documentación para esta clase fue generada a partir de los siguientes ficheros:

- [dispositivodialogo.h](#)
- [dispositivodialogo.cpp](#)

4.7. Referencia de la Clase DispositivoItem

```
#include <dispositivoitem.h>
```

Diagrama de herencias de DispositivoItem



Estructuras de datos

- `struct datosDispositivo`

Tipos públicos

- `enum { Type = UserType + 15 }`

- enum `TipoDispositivo` {
`turbina`, `bomba`, `compresor`, `caldera`,
`condensador`, `calentadorCerrado`, `calentadorAbierto`, `recalentador`,
`evaporador`, `valvulaExpansion`, `trampaVapor`, `refCondensador` }

Métodos públicos

- `DispositivoItem` (`TipoDispositivo tipoDispositivo`, `QMenu *menuContextual`, `DispositivoItem::datosDispositivo` `datos`, `QGraphicsItem *parent=0`)
- void `eliminarConexion` (`ConexionDispositivosItem *conexion`)
- void `eliminarConexiones` ()
- `TipoDispositivo tipoDispositivo` () const
- `QPolygonF obtenerPolygono` () const
- void `agregarConexion` (`ConexionDispositivosItem *conexion`)
- `QPixmap imagen` () const
- `QPainter * crearImagen` (int ancho, int alto)
- int `type` () const
- `datosDispositivo obtenerDatos` ()
- void `ajustarDatos` (`datosDispositivo datos`)
- void `ajustarTexto` (const `QString nombre`)
- void `ajustarId` (int id)
- int `obtenerId` ()
- `QList< ConexionDispositivosItem * > obtenerConexiones` ()
- `Conector * obtenerPtrEntradaCerrado` ()
- `Conector * obtenerPtrSalidaCerrado` ()
- bool `esEtiquetaVisible` ()
- void `mostrarEtiqueta` (bool estado)
- `QPointF obtenerPosEtiqueta` ()
- void `ajustarPosEtiqueta` (`QPointF punto`)

Métodos protegidos

- void `contextMenuEvent` (`QGraphicsSceneContextMenuEvent *event`)
- `QVariant itemChange` (`GraphicsItemChange change`, const `QVariant &value`)

4.7.1. Documentación de las enumeraciones miembro de la clase

4.7.1.1. anonymous enum

Valores de enumeraciones

Type

4.7.1.2. enum `DispositivoItem::TipoDispositivo`

Valores de enumeraciones

turbina

bomba

compresor

caldera

condensador

calentadorCerrado
calentadorAbierto
recalentador
evaporador
valvulaExpansion
trampaVapor
refCondensador

4.7.2. Documentación del constructor y destructor

4.7.2.1. `DispositivoItem::DispositivoItem (TipoDispositivo tipoDispositivo, QMenu * menuContextual, DispositivoItem::datosDispositivo datos, QGraphicsItem * parent = 0)`

4.7.3. Documentación de las funciones miembro

4.7.3.1. `void DispositivoItem::agregarConexion (ConexionDispositivosItem * conexion)`

4.7.3.2. `void DispositivoItem::ajustarDatos (datosDispositivo datos)`

4.7.3.3. `void DispositivoItem::ajustarId (int id)`

4.7.3.4. `void DispositivoItem::ajustarPosEtiqueta (QPointF punto) [inline]`

4.7.3.5. `void DispositivoItem::ajustarTexto (const QString nombre)`

4.7.3.6. `void DispositivoItem::contextMenuEvent (QGraphicsSceneContextMenuEvent * event) [protected]`

4.7.3.7. `QPainter * DispositivoItem::crearImagen (int ancho, int alto)`

4.7.3.8. `void DispositivoItem::eliminarConexion (ConexionDispositivosItem * conexion)`

4.7.3.9. `void DispositivoItem::eliminarConexiones ()`

4.7.3.10. `bool DispositivoItem::esEtiquetaVisible () [inline]`

4.7.3.11. `QPixmap DispositivoItem::imagen () const`

4.7.3.12. `QVariant DispositivoItem::itemChange (GraphicsItemChange change, const QVariant & value) [protected]`

4.7.3.13. `void DispositivoItem::mostrarEtiqueta (bool estado) [inline]`

4.7.3.14. `QList<ConexionDispositivosItem *> DispositivoItem::obtenerConexiones () [inline]`

4.7.3.15. `DispositivoItem::datosDispositivo DispositivoItem::obtenerDatos ()`

4.7.3.16. `int DispositivoItem::obtenerId () [inline]`

4.7.3.17. `QPolygonF DispositivoItem::obtenerPoligono () const [inline]`

4.7.3.18. `QPointF DispositivoItem::obtenerPosEtiqueta () [inline]`

4.7.3.19. `Conector* DispositivoItem::obtenerPtrEntradaCerrado () [inline]`

4.7.3.20. `Conector* DispositivoItem::obtenerPtrSalidaCerrado () [inline]`

4.7.3.21. `TipoDispositivo DispositivoItem::tipoDispositivo () const [inline]`

4.7.3.22. `int DispositivoItem::type () const [inline]`

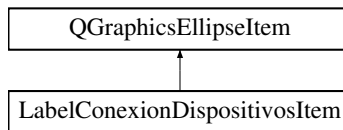
La documentación para esta clase fue generada a partir de los siguientes archivos:

- [dispositivoitem.h](#)
- [dispositivoitem.cpp](#)

4.8. Referencia de la Clase LabelConexionDispositivosItem

```
#include <conexiondispositivositem.h>
```

Diagrama de herencias de LabelConexionDispositivosItem



Tipos públicos

- enum { `Type` = UserType + 5 }

Métodos públicos

- `LabelConexionDispositivosItem` (`QGraphicsItem *parent=0`)
- `int type () const`
- `void ajustarTexto` (`QString dato`)

Métodos protegidos

- `QVariant itemChange` (`GraphicsItemChange change, const QVariant &value`)

4.8.1. Documentación de las enumeraciones miembro de la clase

4.8.1.1. anonymous enum

Valores de enumeraciones

Type

4.8.2. Documentación del constructor y destructor

4.8.2.1. `LabelConexionDispositivosItem::LabelConexionDispositivosItem (QGraphicsItem *parent = 0) [explicit]`

4.8.3. Documentación de las funciones miembro

4.8.3.1. `void LabelConexionDispositivosItem::ajustarTexto (QString dato)`

4.8.3.2. `QVariant LabelConexionDispositivosItem::itemChange (GraphicsItemChange change, const QVariant & value)`
`[protected]`

4.8.3.3. `int LabelConexionDispositivosItem::type () const` `[inline]`

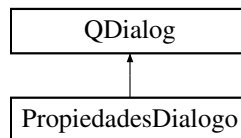
La documentación para esta clase fue generada a partir de los siguientes ficheros:

- [conexiondispositivositem.h](#)
- [conexiondispositivositem.cpp](#)

4.9. Referencia de la Clase PropiedadesDialogo

```
#include <propiedadesdialogo.h>
```

Diagrama de herencias de PropiedadesDialogo



Slots públicos

- void [actualizadoCampo1](#) (const QString &text)
- void [actualizadoCampo2](#) (const QString &text)

Métodos públicos

- [PropiedadesDialogo](#) ([ConexionDispositivosItem::datosConexion](#) info, QWidget *parent=0)
- void [crearWidgets](#) ()
- void [configurarWidgets](#) ()
- [ConexionDispositivosItem::datosConexion](#) [obtenerDatos](#) ()

4.9.1. Documentación del constructor y destructor

4.9.1.1. `PropiedadesDialogo::PropiedadesDialogo (ConexionDispositivosItem::datosConexion info, QWidget *parent = 0)` `[explicit]`

4.9.2. Documentación de las funciones miembro

4.9.2.1. `void PropiedadesDialogo::actualizadoCampo1 (const QString & text)` `[slot]`

4.9.2.2. `void PropiedadesDialogo::actualizadoCampo2 (const QString & text)` `[slot]`

4.9.2.3. `void PropiedadesDialogo::configurarWidgets ()`

4.9.2.4. `void PropiedadesDialogo::crearWidgets ()`

4.9.2.5. `ConexionDispositivosItem::datosConexion PropiedadesDialogo::obtenerDatos ()`

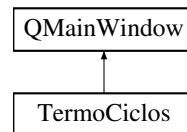
La documentación para esta clase fue generada a partir de los siguientes ficheros:

- [propiedadesdialogo.h](#)
- [propiedadesdialogo.cpp](#)

4.10. Referencia de la Clase TermoCiclos

```
#include <termociclos.h>
```

Diagrama de herencias de TermoCiclos



Métodos públicos

- [TermoCiclos](#) (QWidget *parent=0)
- [~TermoCiclos](#) ()

Métodos protegidos

- void [closeEvent](#) (QCloseEvent *event)

4.10.1. Documentación del constructor y destructor

4.10.1.1. `TermoCiclos::TermoCiclos (QWidget * parent = 0) [explicit]`

4.10.1.2. `TermoCiclos::~~TermoCiclos ()`

4.10.2. Documentación de las funciones miembro

4.10.2.1. `void TermoCiclos::closeEvent (QCloseEvent * event) [protected]`

La documentación para esta clase fue generada a partir de los siguientes ficheros:

- [termociclos.h](#)
- [termociclos.cpp](#)

Capítulo 5

Documentación de archivos

5.1. Referencia del Archivo conector.cpp

```
#include "conector.h"  
#include <QGraphicsSceneHoverEvent>
```

5.2. Referencia del Archivo conector.h

```
#include <QGraphicsItem>  
#include <QGraphicsEllipseItem>  
#include <QPointF>  
#include <QPen>  
#include <QBrush>  
#include <QPainter>  
#include "dispositivoitem.h"
```

Estructuras de datos

- class [Conector](#)

5.3. Referencia del Archivo conexiendispositivositem.cpp

```
#include "conexiendispositivositem.h"  
#include "dispositivoitem.h"  
#include "diagramascene.h"  
#include <math.h>  
#include <QPen>  
#include <QPainter>
```

Variables

- const qreal [Pi](#) = 3.1415927

5.3.1. Documentación de las variables

5.3.1.1. `const qreal Pi = 3.1415927`

5.4. Referencia del Archivo `conexiondispositivositem.h`

```
#include <QGraphicsPathItem>
#include "dispositivoitem.h"
#include "conector.h"
```

Estructuras de datos

- class [LabelConexionDispositivosItem](#)
- class [ConexionDispositivosItem](#)
- struct [ConexionDispositivosItem::datosConexion](#)

5.5. Referencia del Archivo `diagramascene.cpp`

```
#include "diagramascene.h"
#include "conexiondispositivositem.h"
#include "conector.h"
#include "propiedadesdialogo.h"
#include "dispositivodialogo.h"
#include <QGraphicsSceneMouseEvent>
```

5.6. Referencia del Archivo `diagramascene.h`

```
#include <QGraphicsScene>
#include "dispositivoitem.h"
#include "conector.h"
```

Estructuras de datos

- class [DiagramaScene](#)

5.7. Referencia del Archivo `dispositivodialogo.cpp`

```
#include <QHBoxLayout>
#include <QVBoxLayout>
#include "dispositivodialogo.h"
```

5.8. Referencia del Archivo dispositivodialogo.h

```
#include <QDialog>
#include <QDoubleSpinBox>
#include <QSpinBox>
#include <QPushButton>
#include <QLabel>
#include <QLineEdit>
#include "dispositivoitem.h"
```

Estructuras de datos

- class [DispositivoDialogo](#)

5.9. Referencia del Archivo dispositivoitem.cpp

```
#include "dispositivoitem.h"
#include "conexiondispositivositem.h"
#include "conector.h"
#include "diagramascene.h"
#include <QGraphicsScene>
#include <QGraphicsSceneContextMenuEvent>
#include <QMenu>
#include <QPainter>
```

5.10. Referencia del Archivo dispositivoitem.h

```
#include <QGraphicsPolygonItem>
#include <QGraphicsPixmapItem>
#include <QList>
#include "conexiondispositivositem.h"
```

Estructuras de datos

- class [DispositivoItem](#)
- struct [DispositivoItem::datosDispositivo](#)

5.11. Referencia del Archivo main.cpp

```
#include "termociclos.h"
#include <QApplication>
```

Funciones

- int [main](#) (int argc, char *argv[])

5.11.1. Documentación de las funciones

5.11.1.1. `int main (int argc, char * argv[])`

5.12. Referencia del Archivo propiedadesdialogo.cpp

```
#include <QHBoxLayout>
#include <QVBoxLayout>
#include "propiedadesdialogo.h"
```

5.13. Referencia del Archivo propiedadesdialogo.h

```
#include <QDialog>
#include <QDoubleSpinBox>
#include <QComboBox>
#include <QPushButton>
#include <QLabel>
#include <QStringListModel>
#include "CoolProp.h"
#include "HumidAirProp.h"
#include "conexiondispositivositem.h"
```

Estructuras de datos

- class [PropiedadesDialogo](#)

5.14. Referencia del Archivo termociclos.cpp

```
#include "termociclos.h"
#include "conexiondispositivositem.h"
#include "dispositivoitem.h"
#include "diagramascene.h"
#include <QFileDialog>
#include <QtWidgets>
```

5.15. Referencia del Archivo termociclos.h

```
#include "ciclosxml.h"
#include "unidades.h"
#include "CoolProp.h"
#include "HumidAirProp.h"
#include "planta.h"
#include "sle.h"
#include "dispositivoitem.h"
#include <vector>
#include <iostream>
#include <fstream>
#include <QMainWindow>
#include <QLabel>
#include <QProcess>
#include <QDesktopServices>
#include <QCloseEvent>
```

Estructuras de datos

- class [TermoCiclos](#)

5.16. Referencia del Archivo test.cpp

```
#include "ciclosxml.h"
#include "unidades.h"
#include "CoolProp.h"
#include "HumidAirProp.h"
#include "planta.h"
#include "sle.h"
#include <vector>
#include <iostream>
#include <fstream>
```

Funciones

- void [test](#) ()

5.16.1. Documentación de las funciones

5.16.1.1. void [test](#) ()

Índice alfabético

- ~TermoCiclos
 - TermoCiclos, 19
- actualizadoCampo1
 - PropiedadesDialogo, 18
- actualizadoCampo2
 - PropiedadesDialogo, 18
- actulizarVinculosIds
 - DiagramaScene, 13
- agregarConexion
 - DispositivoItem, 16
- ajustarDatos
 - ConexionDispositivosItem, 10
 - DispositivoItem, 16
- ajustarId
 - Conector, 8
 - ConexionDispositivosItem, 10
 - DispositivoItem, 16
- ajustarModo
 - DiagramaScene, 13
- ajustarPosEtiqueta
 - ConexionDispositivosItem, 10
 - DispositivoItem, 16
- ajustarTexto
 - DispositivoItem, 16
 - LabelConexionDispositivosItem, 17
- ajustarTipoItem
 - DiagramaScene, 13
- ajustarVerificado
 - ConexionDispositivosItem, 10
- bomba
 - DispositivoItem, 15
- boundingRect
 - ConexionDispositivosItem, 10
- caldera
 - DispositivoItem, 15
- calentadorAbierto
 - DispositivoItem, 16
- calentadorCerrado
 - DispositivoItem, 15
- closeEvent
 - TermoCiclos, 19
- compresor
 - DispositivoItem, 15
- condensador
 - DispositivoItem, 15
- Conector, 7
 - ajustarId, 8
- Conector, 8
 - entrada, 8
 - getColorFondo, 8
 - getColorLinea, 8
 - hoverEnterEvent, 8
 - hoverLeaveEvent, 8
 - obtenerId, 8
 - obtenerTipo, 8
 - paint, 8
 - salida, 8
 - setColorFondo, 8
 - setColorLinea, 8
 - TipoConector, 8
 - Type, 8
 - type, 8
- conector.cpp, 21
- conector.h, 21
- ConexionDispositivosItem, 8
 - ajustarDatos, 10
 - ajustarId, 10
 - ajustarPosEtiqueta, 10
 - ajustarVerificado, 10
 - boundingRect, 10
 - ConexionDispositivosItem, 10
 - esEtiquetaVisible, 10
 - estaVerificado, 10
 - itemFinal, 10
 - itemInicio, 10
 - mostrarEtiqueta, 10
 - obtenerDatos, 10
 - obtenerFinal, 10
 - obtenerId, 10
 - obtenerInicio, 10
 - obtenerPosEtiqueta, 10
 - paint, 10
 - shape, 10
 - Type, 9
 - type, 10
 - updatePosition, 10
- ConexionDispositivosItem::datosConexion, 10
 - enableFluido, 11
 - enableProp1, 11
 - enableProp2, 11
 - miFluido, 11
 - miProp1, 11
 - miProp2, 11
 - miTipoProp1, 11
 - miTipoProp2, 11
 - miUniProp1, 11

- miUniProp2, 11
- conexiondispositivositem.cpp, 21
 - Pi, 22
- conexiondispositivositem.h, 22
- configurarWidgets
 - PropiedadesDialogo, 18
- contextMenuEvent
 - DispositivoItem, 16
- crearImagen
 - DispositivoItem, 16
- crearWidgets
 - DispositivoDialogo, 14
 - PropiedadesDialogo, 18
- deshabilitarNsalidas
 - DispositivoDialogo, 14
- DiagramaScene, 12
 - actualizarVinculosIds, 13
 - ajustarModo, 13
 - ajustarTipoItem, 13
 - DiagramaScene, 13
 - documentoModificado, 13
 - InsertarItem, 13
 - InsertarLinea, 13
 - itemInsertado, 13
 - itemSeleccionado, 13
 - Modo, 13
 - mouseDoubleClickEvent, 13
 - mouseMoveEvent, 13
 - mousePressEvent, 13
 - MoverItem, 13
 - obtenerMiMenuItem, 13
- diagramascene.cpp, 22
- diagramascene.h, 22
- DispositivoDialogo, 13
 - crearWidgets, 14
 - deshabilitarNsalidas, 14
 - DispositivoDialogo, 14
 - obtenerDatos, 14
- DispositivoItem, 14
 - agregarConexion, 16
 - ajustarDatos, 16
 - ajustarId, 16
 - ajustarPosEtiqueta, 16
 - ajustarTexto, 16
 - bomba, 15
 - caldera, 15
 - calentadorAbierto, 16
 - calentadorCerrado, 15
 - compresor, 15
 - condensador, 15
 - contextMenuEvent, 16
 - crearImagen, 16
 - DispositivoItem, 16
 - eliminarConexion, 16
 - eliminarConexiones, 16
 - esEtiquetaVisible, 16
 - evaporador, 16
 - imagen, 16
 - itemChange, 16
 - mostrarEtiqueta, 16
 - obtenerConexiones, 16
 - obtenerDatos, 16
 - obtenerId, 16
 - obtenerPoligono, 16
 - obtenerPosEtiqueta, 16
 - obtenerPtrEntradaCerrado, 16
 - obtenerPtrSalidaCerrado, 16
 - recalentador, 16
 - refCondensador, 16
 - TipoDispositivo, 15
 - tipoDispositivo, 16
 - trampaVapor, 16
 - turbina, 15
 - Type, 15
 - type, 17
 - valvulaExpansion, 16
- DispositivoItem::datosDispositivo, 11
 - miEficiencia, 12
 - miNombre, 12
 - miNombreXML, 12
 - nPuertosSalida, 12
 - tipo, 12
- dispositivodialogo.cpp, 22
- dispositivodialogo.h, 23
- dispositivoitem.cpp, 23
- dispositivoitem.h, 23
- documentoModificado
 - DiagramaScene, 13
- eliminarConexion
 - DispositivoItem, 16
- eliminarConexiones
 - DispositivoItem, 16
- enableFluido
 - ConexionDispositivosItem::datosConexion, 11
- enableProp1
 - ConexionDispositivosItem::datosConexion, 11
- enableProp2
 - ConexionDispositivosItem::datosConexion, 11
- entrada
 - Conector, 8
- esEtiquetaVisible
 - ConexionDispositivosItem, 10
 - DispositivoItem, 16
- estaVerificado
 - ConexionDispositivosItem, 10
- evaporador
 - DispositivoItem, 16
- getColorFondo
 - Conector, 8
- getColorLinea
 - Conector, 8
- hoverEnterEvent
 - Conector, 8
- hoverLeaveEvent

- Conector, 8
- imagen
 - DispositivoItem, 16
- InsertarItem
 - DiagramaScene, 13
- InsertarLinea
 - DiagramaScene, 13
- itemChange
 - DispositivoItem, 16
 - LabelConexionDispositivosItem, 17
- itemFinal
 - ConexionDispositivosItem, 10
- itemInicio
 - ConexionDispositivosItem, 10
- itemInsertado
 - DiagramaScene, 13
- itemSeleccionado
 - DiagramaScene, 13
- LabelConexionDispositivosItem, 17
 - ajustarTexto, 17
 - itemChange, 17
 - LabelConexionDispositivosItem, 17
 - Type, 17
 - type, 18
- main
 - main.cpp, 24
- main.cpp, 23
 - main, 24
- miEficiencia
 - DispositivoItem::datosDispositivo, 12
- miFluido
 - ConexionDispositivosItem::datosConexion, 11
- miNombre
 - DispositivoItem::datosDispositivo, 12
- miNombreXML
 - DispositivoItem::datosDispositivo, 12
- miProp1
 - ConexionDispositivosItem::datosConexion, 11
- miProp2
 - ConexionDispositivosItem::datosConexion, 11
- miTipoProp1
 - ConexionDispositivosItem::datosConexion, 11
- miTipoProp2
 - ConexionDispositivosItem::datosConexion, 11
- miUniProp1
 - ConexionDispositivosItem::datosConexion, 11
- miUniProp2
 - ConexionDispositivosItem::datosConexion, 11
- Modo
 - DiagramaScene, 13
- mostrarEtiqueta
 - ConexionDispositivosItem, 10
 - DispositivoItem, 16
- mouseDoubleClickEvent
 - DiagramaScene, 13
- mouseMoveEvent
 - DiagramaScene, 13
- mousePressEvent
 - DiagramaScene, 13
- MoverItem
 - DiagramaScene, 13
- nPuertosSalida
 - DispositivoItem::datosDispositivo, 12
- obtenerConexiones
 - DispositivoItem, 16
- obtenerDatos
 - ConexionDispositivosItem, 10
 - DispositivoDialogo, 14
 - DispositivoItem, 16
 - PropiedadesDialogo, 18
- obtenerFinal
 - ConexionDispositivosItem, 10
- obtenerId
 - Conector, 8
 - ConexionDispositivosItem, 10
 - DispositivoItem, 16
- obtenerInicio
 - ConexionDispositivosItem, 10
- obtenerMiMenuitem
 - DiagramaScene, 13
- obtenerPoligono
 - DispositivoItem, 16
- obtenerPosEtiqueta
 - ConexionDispositivosItem, 10
 - DispositivoItem, 16
- obtenerTipo
 - Conector, 8
- obtenerPtrEntradaCerrado
 - DispositivoItem, 16
- obtenerPtrSalidaCerrado
 - DispositivoItem, 16
- paint
 - Conector, 8
 - ConexionDispositivosItem, 10
- Pi
 - conexiondispositivositem.cpp, 22
- PropiedadesDialogo, 18
 - actualizadoCampo1, 18
 - actualizadoCampo2, 18
 - configurarWidgets, 18
 - crearWidgets, 18
 - obtenerDatos, 18
 - PropiedadesDialogo, 18
- propiedadesdialogo.cpp, 24
- propiedadesdialogo.h, 24
- recalentador
 - DispositivoItem, 16
- refCondensador
 - DispositivoItem, 16
- salida

- Conector, 8
- setColorFondo
 - Conector, 8
- setColorLinea
 - Conector, 8
- shape
 - ConexionDispositivosItem, 10
- TermoCiclos, 19
 - ~TermoCiclos, 19
 - closeEvent, 19
 - TermoCiclos, 19
- termociclos.cpp, 24
- termociclos.h, 25
- test
 - test.cpp, 25
- test.cpp, 25
 - test, 25
- tipo
 - DispositivoItem::datosDispositivo, 12
- TipoConector
 - Conector, 8
- TipoDispositivo
 - DispositivoItem, 15
- tipoDispositivo
 - DispositivoItem, 16
- trampaVapor
 - DispositivoItem, 16
- turbina
 - DispositivoItem, 15
- Type
 - Conector, 8
 - ConexionDispositivosItem, 9
 - DispositivoItem, 15
 - LabelConexionDispositivosItem, 17
- type
 - Conector, 8
 - ConexionDispositivosItem, 10
 - DispositivoItem, 17
 - LabelConexionDispositivosItem, 18
- updatePosition
 - ConexionDispositivosItem, 10
- valvulaExpansion
 - DispositivoItem, 16