

UNIVERSIDAD DE ALMERIA

ESCUELA SUPERIOR DE INGENIERÍA

Desarrollo de una herramienta software
para simulación y control del demostrador
de una planta termosolar híbrida de
energías renovables

Curso 2015/2016

Alumno/a:

Gustavo José Martín de Dios

Director/es:

Dr. D. José Domingo Álvarez Hervás

Dr. D. Javier Bonilla Cruz



Índice de contenidos

1. Introducción	9
2. Descripción proyecto HYSOL	12
2.1. Introducción	12
2.2. Tecnología HYSOL	12
2.3. Demostrador de la tecnología HYSOL	14
2.4. Herramienta software para operación y control del demostrador HYSOL	16
3. Materiales y métodos	20
3.1. Fases del proyecto	20
3.2. LabVIEW	21
3.3. Importación de modelos mediante complemento FMI	26
3.4. Módulo MathScript	26
3.5. Módulo de Control y Simulación	27
3.6. Módulo de Supervisión y Registro de datos	29
3.7. Teoría de Control	29
3.7.1. Control PID	30
3.7.2. Esquema anti-windup	32
3.7.3. Transferencia sin saltos	33
3.7.4. Control anticipativo (<i>feedforward</i>)	34
3.7.5. Control en cascada	35
4. Sistema SCADA desarrollado	37
4.1. Interfaz gráfica	37
4.2. Variables compartidas	43
4.3. Inicialización del sistema	48
4.4. Visualización de gráficas	52
4.5. Lazos de control	55
4.5.1. Lazo de control FIC-103	56
4.5.2. Lazo de control FIC-104	65
4.5.3. Lazo de control de la turbina de gas virtual	74
4.6. Alarmas	81

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

5. Resultados y discusión. Conclusiones	83
5.1. Resultados y discusión.....	83
5.2. Conclusiones.....	93
6. Bibliografía	95

Índice de Figuras

Figura 1a. Configuración CSP + TES de torre.....	13
Figura 1b. Configuración HYSOL.....	13
Figura 2. Demostrador de la tecnología HYSOL en Manchasol.....	14
Figura 3. Configuración del demostrador del proyecto HYSOL.....	16
Figura 4. Esquema de la herramienta software del demostrador HYSOL.....	16
Figura 5a. Turbina de gas virtual.....	18
Figura 5b. HRS.....	18
Figura 5c. Tanque de sales calientes.....	18
Figura 5d. Enfriador de aire.....	18
Figura 6. Panel frontal.....	22
Figura 7. Diagrama de Bloques.....	23
Figura 8. Paleta de Controles.....	24
Figura 9. Paleta de Funciones.....	25
Figura 10. Ejemplo de Nodo MathScript usado en la herramienta.....	27
Figura 11. Bloque que representa el Lazo de Simulación y Control.....	28
Figura 12. Paleta de funciones del módulo de Control y Simulación.....	28
Figura 13. Paleta de funciones del módulo DSC.....	29
Figura 14. Esquema clásico algoritmo PID.....	30
Figura 15. Esquema anti-windup.....	32
Figura 16. Esquema transferencia sin saltos.....	33
Figura 17. Perturbación “D(s)” en esquema de control.....	34
Figura 18. Esquema de control con <i>feedforward</i>	35
Figura 19. Esquema de control en cascada.....	36
Figura 20. Sinóptico de la planta.....	37
Figura 21. Ventana de explorador de proyecto.....	38
Figura 22. Visualización general del SCADA-pestaña “INIT”.....	39
Figura 23. Detalle de las pestañas del SCADA.....	39
Figura 24. Ejemplo de <i>Stacked Sequence</i>	39
Figura 25. Inicialización de las pestañas.....	40
Figura 26. Estructura bucle <i>For</i> más el <i>Case</i> acoplado.....	40
Figura 27a. Invocación del VI “MAIN”.....	41
Figura 27b. Invocación del VI “GRAPHS”.....	41
Figura 28. Módulo del modelo importado a través del FMI.....	41
Figura 29. Variables compartidas en librería de proyecto.....	43
Figura 30. Ventana de configuración de variables compartidas.....	44
Figura 31a. Modo lectura.....	44

Figura 31b. Modo escritura.....	44
Figura 32. Enlace de datos con variable compartida (<i>Data Binding</i>).....	45
Figura 33. Indicador verde de conexión satisfactoria con variable compartida	45
Figura 34. Pestaña de inicialización del sistema SCADA	48
Figura 35. Ventana de inicialización en el diagrama de bloques.....	50
Figura 36a. Icono de variables globales: modo lectura.....	50
Figura 36b. Icono de variables globales: modo escritura.....	50
Figura 37. VI de variables globales	51
Figura 38. Traspaso de información de las variables globales a las variables compartidas	51
Figura 39. Introducción de los valores de inicialización al modelo como entradas.....	52
Figura 40. Visualización de una de las cuatro gráficas en tiempo real, habilitando dos de las cuatro variables disponibles	53
Figura 41. Diagrama de bloques del VI de las gráficas.....	53
Figura 42. Bucle principal del VI de las gráficas	54
Figura 43. Salidas del modelo que se mostrarán en las gráficas en tiempo de simulación	54
Figura 44. Gráficas en tiempo de simulación.....	55
Figura 45. Esquema de control FIC-103.....	56
Figura 46. Ventana de elección de diferentes modos de operación y controles de consignas para el lazo FIC-103.....	58
Figura 47. Esquema de la bomba P01	59
Figura 48. Esquema general del lazo FIC-103 en el diagrama de bloques	59
Figura 49. Bloque PID Advanced de LabVIEW.....	60
Figura 50. Lazo secundario del FIC-103 en el diagrama de bloques	60
Figura 51. Inicialización de la válvula FV-01	62
Figura 52. Lazo primario del FIC-103 en el diagrama de bloques.....	62
Figura 53. Acción <i>feedforward</i> en el lazo primario del FIC-103 en el diagrama de bloques	63
Figura 54. Evento que ejecuta el valor inicial del <i>feedforward</i> en el FIC-103.....	64
Figura 55. Transmisión de la información a la variable de apoyo de la señal de control en el FIC-103	64
Figura 56. Variables del sistema informando al resto de variables para que estén actualizadas en el FIC-103.....	65
Figura 57. Esquema de control FIC-104.....	65
Figura 58. Ventana de elección de diferentes modos de operación y controles de consignas para el lazo FIC-104	67
Figura 59. Esquema general del lazo FIC-104 en el diagrama de bloques	68
Figura 60. Lazo secundario del FIC-104 en el diagrama de bloques	69
Figura 61. Lazo primario del FIC-104 en el diagrama de bloques.....	70
Figura 62. Acción <i>feedforward</i> en el lazo primario del FIC-104 en el diagrama de bloques	71
Figura 63. Evento que ejecuta el valor inicial del <i>feedforward</i> en el FIC-104.....	72
Figura 64. Transmisión de la información a la variable de apoyo de la señal de control del FIC-104	72

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

Figura 65. Variables del sistema informando al resto de variables para que estén actualizadas en el FIC-104.....	73
Figura 66. Evento que no permite la confluencia de los dos lazos de control	73
Figura 67. Esquema de control de la turbina de gas virtual	74
Figura 68. Virtual Gas Turbine en el SCADA.....	74
Figura 69. Ventana emergente de la Turbina	76
Figura 70. Diagrama de bloques de la ventana emergente de la Turbina.....	77
Figura 71. Ventana de finalización de la ventana emergente de la Turbina.....	77
Figura 72. Desindexación del vector de datos de la curva de temperatura	78
Figura 73. Desindexación del vector de datos de la curva de flujo másico	78
Figura 74. Diagrama de bloques del esquema de control de la turbina de gas virtual	79
Figura 75. Diagrama de bloques del primer lazo de control del esquema de control de la turbina de gas virtual	79
Figura 76. Diagrama de bloques del segundo lazo de control del esquema de control de la turbina de gas virtual	80
Figura 77. Visión de las alarmas en el SCADA.....	81
Figura 78. Diagrama de bloques de las alarmas	82
Figura 79. Configuración de las alarmas en las variables compartidas.....	82
Figura 80. Esquema del demostrador de la planta en su configuración inicial	83
Figura 81. Sinóptico de la planta usando el control de flujo másico del FIC-103	84
Figura 82. Gráficas que enseñan la dinámica de la temperatura de sales fundidas TT106 y del flujo másico de sales FT103	84
Figura 83. Permuta al control de temperatura con <i>feedforward</i> en el FIC-103.....	85
Figura 84. Gráficas que muestran la dinámica de la temperatura de sales TT106 y del flujo másico de sales FT103	86
Figura 85. Activación del lazo de control FIC-104.....	86
Figura 86. Control de temperatura activado en el FIC-104	87
Figura 87. Gráficas que enseñan la dinámica de la temperatura de sales TT106 y del flujo másico de gases FT104.....	88
Figura 88. Cambio al control de flujo másico de gases en el lazo FIC-104.....	88
Figura 89. Gráficas que muestran la dinámica de la temperatura de sales TT106 y del flujo másico de gases FT104.....	89
Figura 90. Ventana donde aparece la opción de la turbina de gas virtual.....	90
Figura 91. Sinóptico de la planta donde se está utilizando el lazo de control de la turbina de gas virtual.....	90
Figura 92. Gráfica donde se enseña la dinámica de la temperatura del gas a la entrada del HRS TY134.....	91
Figura 93. Ventana donde aparece la opción de la turbina de gas virtual.....	91
Figura 94. Sinóptico de la planta usando el lazo de control de la turbina de gas virtual...	92
Figura 95. Gráficas donde se representa la dinámica de la temperatura del gas a la entrada del HRS TY134 y del flujo másico de gases FT104.....	92
Figura 96. Cuadro de alarmas en su ventana correspondiente del SCADA.....	93

Índice de Tablas

Tabla 1. Sensores y variables del sistema.....	15
Tabla 2. Cronograma del trabajo de fin de grado.....	20
Tabla 3. Entradas y salidas del modelo del demostrador.....	42
Tabla 4. Variables compartidas utilizadas en el proyecto	47
Tabla 5. Parámetros de inicialización del sistema	49
Tabla 6. Variables utilizadas para las alarmas.....	81

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

1. Introducción

Hoy en día en cualquier proceso industrial es habitual usar sistemas autónomos sin la intervención permanente del ser humano. Estos sistemas son capaces de controlar y gobernar el proceso.

Existen muchos y muy variados sistemas de control y todos, si se aplican bien, ofrecen soluciones óptimas en entornos industriales. Uno de los más utilizados es el sistema SCADA (del inglés: *Supervisory Control And Data Acquisition*) que es una aplicación software a través de la cual se puede supervisar y controlar las distintas variables características de la planta, además de registrar y tener la capacidad de poder tratar dichos datos con el fin de retroalimentar al sistema. También dispone de un sistema de alarmas con el cual se podrá avisar al operario de valores fuera de rangos comunes o de situaciones extrañas en la planta.

Lo que hace de los sistemas SCADA una herramienta diferenciada es su característica de control supervisado. De hecho, la parte de control está definida y supeditada por el proceso que se desea controlar y, en última instancia, por el hardware e instrumental de control utilizados en la planta, que pueden existir previamente a la implantación del SCADA, que se instalará sobre y en función de estos sistemas de control.

En consecuencia, el operador supervisa el control de la planta y no solo monitorea las variables que en un momento determinado están actuando sobre la planta; esto significa que puede actuar y modificar las variables de control en tiempo real, algo que pocos sistemas permiten con la facilidad intuitiva que ofrecen los sistemas SCADA.

La labor del supervisor representa una tarea delicada y esencial desde el punto de vista normativo y operativo. De esta acción depende en gran medida el poder de garantizar la calidad y eficiencia del proceso. En el supervisor descansa la responsabilidad de orientar o corregir las acciones que se desarrollan. Por lo tanto, la toma de decisiones sobre las acciones de control está en manos del supervisor, que en el caso de SCADA es el operario [1].

Esto diferencia notablemente a sistemas SCADA de los sistemas clásicos de automatización, en los que las variables de control están distribuidas sobre los controladores de la planta. Eso dificulta mucho las variaciones en el proceso, ya que una vez implementados, estos sistemas no permiten un control óptimo en tiempo real.

Entre las funciones de un sistema SCADA, se pueden destacar las siguientes:

- ✓ Posibilidad de crear paneles de alarma, que exigen la presencia del operador para reconocer una parada o situación de alarma, con registro de incidencias.
- ✓ Generación de datos históricos de las señales del sistema y almacenamiento de los mismos, que pueden ser volcados para su proceso sobre una hoja de cálculo.
- ✓ Ejecución de programas que modifican la ley de control o incluso el programa total sobre el autómatas (bajo ciertas condiciones).
- ✓ Posibilidad de programación numérica, que permite realizar cálculos aritméticos de elevada resolución sobre la CPU del ordenador.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

- ✓ Supervisión remota de instalaciones: el usuario es capaz de conocer el estado de las instalaciones bajo su responsabilidad y coordinar eficientemente las labores de producción y mantenimiento en la planta.
- ✓ Control remoto de instalaciones: los sistemas SCADA permiten activar o desactivar equipos remotos de manera automática o a solicitud del operador. Igualmente es posible realizar ajustes remotamente a parámetros del proceso.
- ✓ Procesamiento de información: en algunos casos, los datos capturados requieren procesamiento adicional.
- ✓ Presentación de gráficos dinámicos: despliegue de pantallas con el diagrama del proceso conteniendo información instantánea del comportamiento del mismo y poder compararlo con datos históricos.

En definitiva, los sistemas SCADA ofrecen una perspectiva integrada de todos los recursos de control e información de la planta. De esta manera, los ingenieros, supervisores y operarios pueden visualizar e interactuar con los procesos mediante sus representaciones gráficas [2].

Una premisa fundamental en la automatización industrial es que las máquinas o software hagan lo que les corresponde, y que el ser humano no haga lo que una máquina, software o robot puede realizar mecánicamente. De esta forma, el ser humano dedicará su tiempo a las tareas que demandan “pensar” y no a trabajos repetitivos que fácilmente una máquina o un software pueden realizar o supervisar.

En este trabajo de fin de grado se abordará la realización de un sistema SCADA del demostrador de una planta termosolar híbrida de energías renovables, el cual se explicará más adelante. Este sistema está enfocado a la simulación del proceso con el fin de entrenar a futuros operarios de la planta, para probar nuevas estrategias de control y estudiar las dinámicas del sistema.

La simulación de procesos es una de las herramientas más potentes en el ámbito de la ingeniería, la cual se utiliza para representar un proceso mediante otro que lo hace mucho más simple y entendible. Es una de las herramientas más importantes y más interdisciplinarias. El usuario define la estructura del sistema que quiere simular.

Una ejecución del programa de simulación correspondiente informa de cual será el comportamiento dinámico de su proceso a través del modelo que se ha diseñado. Así podemos ver los pronósticos para la demanda y utilidad de nuestro producto, o ver cuando un sistema puede fallar bajo condiciones adversas del entorno donde funcionará.

Cabe mencionar la creciente importancia de la simulación en la investigación de operaciones y en sus aplicaciones industriales. En los países altamente desarrollados la simulación es una herramienta principal en los procesos de toma de decisiones, en el manejo de empresas y la planificación de la producción.

Los modelos a simular se convierten en la plataforma mínima a desarrollar por las empresas para poder tener una valoración sobre el producto o planta que desea llevar a cabo. Para una industria, el invertir en el desarrollo e implementación de un sistema de simulación es

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

sumamente beneficioso, ya que los efectos que se generan, permiten manejar el proceso dentro de un ambiente permisible a modificaciones y cambios, y sin incurrir en cuantiosos gastos económicos y de tiempo.

La simulación hoy en día es cada vez más amigable para el usuario, que no tiene que ser un especialista en informática para poder hacer uso de ella y poder tener un pronóstico sobre un tema determinado.

En este trabajo de fin de grado se va a simular el comportamiento del demostrador de la planta termosolar híbrida, creando un sistema SCADA que represente fidedignamente el proceso y con una interfaz gráfica accesible e intuitiva para cualquier persona sin necesidad de tener una alta cualificación.

➤ Objetivos:

Los objetivos principales que se plantean alcanzar en este trabajo de fin de grado son los siguientes:

- ✓ Estudiar y simular las dinámicas del sistema.
- ✓ Diseñar, probar y validar nuevas estrategias de control en simulación antes de implantarlas en el sistema real.
- ✓ Diseñar un sistema SCADA para entrenar operadores de planta.

Una vez cumplidos estos objetivos, se podrá tener una idea general del funcionamiento de la planta y así poder tratar de optimizar el diseño en simulación para mejorar la eficiencia del proceso sin tener que llevar a cabo la construcción del sistema real, ni realizar campañas experimentales sobre dicha instalación.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

2. Descripción proyecto HYSOL

2.1. Introducción

Las energías renovables son una de las fuentes de energía con más futuro que existen. Uno de los problemas que concierne a las energías renovables es la dificultad para satisfacer las demandas energéticas especialmente bajo condiciones meteorológicas adversas. Por ello, una de las tecnologías más usadas para paliar esta carencia es la energía termosolar de concentración –también conocida como **CSP**, del inglés: *Concentrating Solar Power*-.

Las plantas CSP están preparadas para sobreponerse a este inconveniente gracias al eficiente almacenamiento térmico de la energía solar –**TES**, del inglés: *Thermal Energy Storage*-. Sin embargo, los sistemas de almacenamiento están limitados en sus capacidades de almacenamiento, principalmente por cuestiones económicas, actualmente los disponibles no tienen la capacidad para responder a la demanda solicitada especialmente bajo largos periodos de condiciones meteorológicas desfavorables.

La solución a este problema hoy en día consiste en la hibridación de la energía solar con combustibles fósiles, lo que ha dado lugar a plantas de ciclo combinado integrado con tecnología CSP, donde la contribución de la energía solar es relativamente baja (<20%) y plantas CSP con apoyo de combustibles fósiles. Una gran alternativa a usar combustibles fósiles, ya que no son sostenibles y su tratamiento es dañino para el medio ambiente, es la hibridación de energía solar con biogás. El biogás presenta la ventaja de que puede ser más eficientemente transportado que la biomasa a través de una red de distribución o licuado en tanques presurizados.

2.2. Tecnología HYSOL

Dentro de este contexto nace el proyecto HYSOL [3], cuyo principal objetivo es el estudio, diseño, optimización y construcción a escala preindustrial de un demostrador basado en una configuración innovadora de hibridación entre una planta CSP con sistemas TES y turbina de vapor (ciclo Rankine) junto con una turbina de gas (ciclo Brayton abierto) y un recuperador de calor entre el fluido de almacenamiento y los gases exhaustos de la turbina de gas –**HRS**, del inglés: *Heat Recovery System*-, donde el gas de combustión puede ser biogás, siendo su naturaleza totalmente renovable. Hay múltiples tipos de configuraciones para HYSOL en función del tipo de planta CSP y del fluido del sistema TES, pero la más prometedora se corresponde a un sistema de torre en planta CSP con un sistema TES de almacenamiento directo donde tanto el fluido de trabajo como el de almacenamiento son sales fundidas, como se puede observar en la figura 1b.

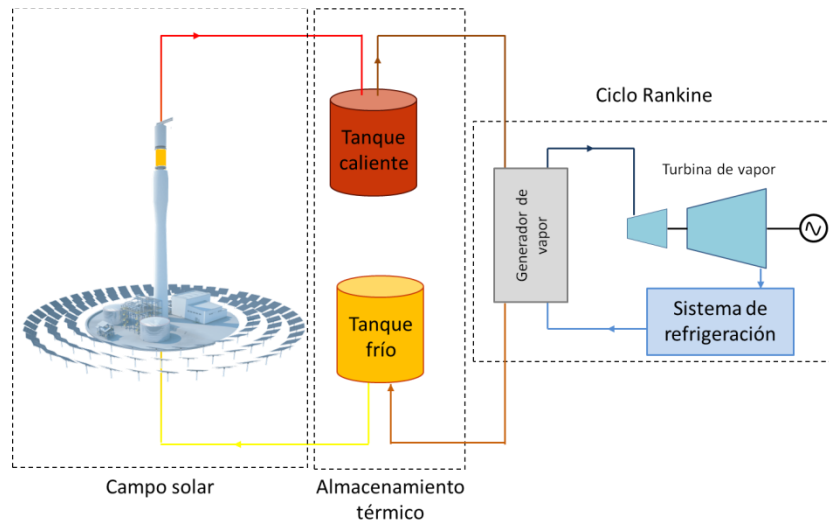


Figura 1a. Configuración CSP + TES de torre

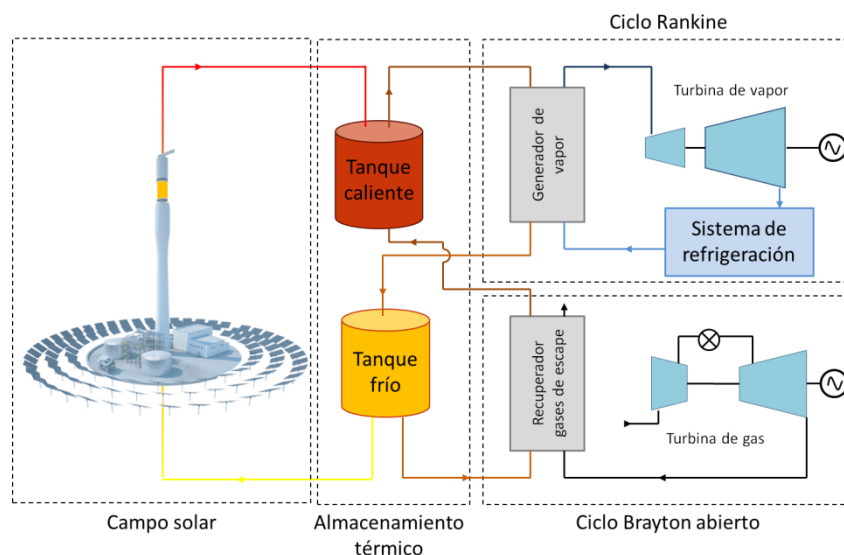


Figura 1b. Configuración HYSOL

En la figura 1 se puede observar la diferencia entre una planta CSP convencional (figura 1a) y la configuración del proyecto HYSOL (figura 1b), que es la incorporación de un ciclo Brayton abierto, con una turbina de gas y el recuperador de gases (HRS).

La tecnología HYSOL tiene principalmente tres modos de operación:

1. Solar con turbina de vapor:

La energía térmica de las sales fundidas que provienen del campo solar se almacena en el tanque caliente; esta energía es usada en el ciclo Rankine para producir electricidad gracias a un generador de vapor y una turbina. Una vez las sales fundidas hayan circulado por el generador de vapor cediendo su energía térmica, son almacenadas en el tanque frío.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

2. Descarga con turbina de vapor:

Cuando el campo solar no pueda proporcionar suficiente energía para mantener la turbina de vapor en condiciones nominales de operación y el sistema TES tenga suficiente energía térmica, las sales fundidas pueden continuar fluyendo desde el tanque caliente al generador de vapor como en el modo 1.

3. Turbina de gas y vapor:

Cuando tanto el campo solar como el sistema TES no puedan proporcionar energía térmica suficiente al ciclo Rankine, la turbina de gas en el ciclo de Brayton abierto producirá electricidad con biogás como gas de combustión. Además, el flujo de gases de escape de la turbina será usado para calentar las sales fundidas en el HRS, que serán circuladas desde el tanque frío y almacenadas en el tanque caliente, lo que permitirá producir electricidad como se indicó en el punto 2. Los gases de salida del recuperador también podrían ser utilizados para optimizar el rendimiento de la planta.

2.3. Demostrador de la tecnología HYSOL

Con el fin de reproducir y estudiar la configuración previamente descrita, se ha construido en el ámbito del proyecto HYSOL un demostrador de esta tecnología en la planta termosolar Manchasol perteneciente al grupo COBRA (ver figura 2). Los principales elementos del demostrador son los siguientes (ver figura 3):

- ✓ Turbina de gas virtual: este componente, con su adecuado sistema de control, simula una turbina de gas con postcombustor. Está compuesto por un sistema de entrada de aire, un tanque de gas y un calentador de aire (*Air Heater*).
- ✓ Sistema de Recuperación de Calor (HRS): permite recuperar energía térmica del flujo de gases de la turbina de gas virtual, calentando así las sales fundidas.
- ✓ Tanque de sales fundidas: una vez que las sales fundidas son calentadas en el HRS, son almacenadas en este tanque.
- ✓ Enfriador de aire: emula el proceso de descarga de las sales fundidas; la transferencia de energía desde el tanque caliente de sales fundidas en el generador de vapor.



Figura 2. Demostrador de la tecnología HYSOL en Manchasol

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

Con el fin de saber en cada momento cuál es el valor de las variables más significativas del demostrador, así como el valor de los actuadores y las consignas que controlan el proceso, se ha creado una serie de variables a observar y controlar, que también están incluidas en el sistema SCADA.

Estos sensores y variables están etiquetados de la siguiente manera:

Nombre Variable	Tipo	Descripción
F-01-SC2	Consigna	Indica la temperatura (en °C) deseada a la entrada del HRS
F-01-SI	Actuador	Indica el porcentaje de apertura deseado de la válvula de apertura del Calentador de Aire (<i>Air Heater</i>)
FT103	Sensor	Muestra el caudal de sales fundidas circulante (en kg/h) a la entrada del HRS
FT103_SP	Consigna	Indica el caudal circulante de sales fundidas (en kg/h) deseado a la entrada del HRS / salida de la válvula FV-01-ZL
FT104	Sensor	Muestra el caudal circulante de gas (en kg/h) a la salida del Calentador de Aire (<i>Air Heater</i>)
FT104_SP	Consigna	Indica el caudal de gas circulante (en kg/h) deseado a la salida del Calentador de Aire (<i>Air Heater</i>) / entrada del HRS
FV-01-ZL	Sensor	Muestra el valor en porcentaje de apertura de la válvula que controla el paso de sales fundidas
LT129	Sensor	Muestra el valor de altura (en metros) del tanque de almacenamiento de sales fundidas calientes
P-01-SC	Actuador	Indica el valor en hertzios a la que deseamos que funcione la bomba
TT106	Sensor	Muestra la temperatura (en °C) que tienen las sales fundidas a la salida del HRS
TT106_SP	Consigna	Indica el valor deseado de temperatura (en °C) de las sales fundidas a la salida del HRS
TT112	Sensor	Muestra la temperatura (en °C) del tanque de almacenamiento de sales fundidas calientes
TT140	Sensor	Muestra la temperatura (en °C) de las sales fundidas a la entrada del HRS
TY103	Sensor	Muestra la temperatura (en °C) de los gases a la salida del HRS
TY134	Sensor	Muestra la temperatura (en °C) de los gases a la entrada del HRS
ZI-V01	Actuador	Indica el valor en porcentaje de apertura deseado de la válvula que controla el paso de sales fundidas

Tabla 1. Sensores y variables del sistema

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

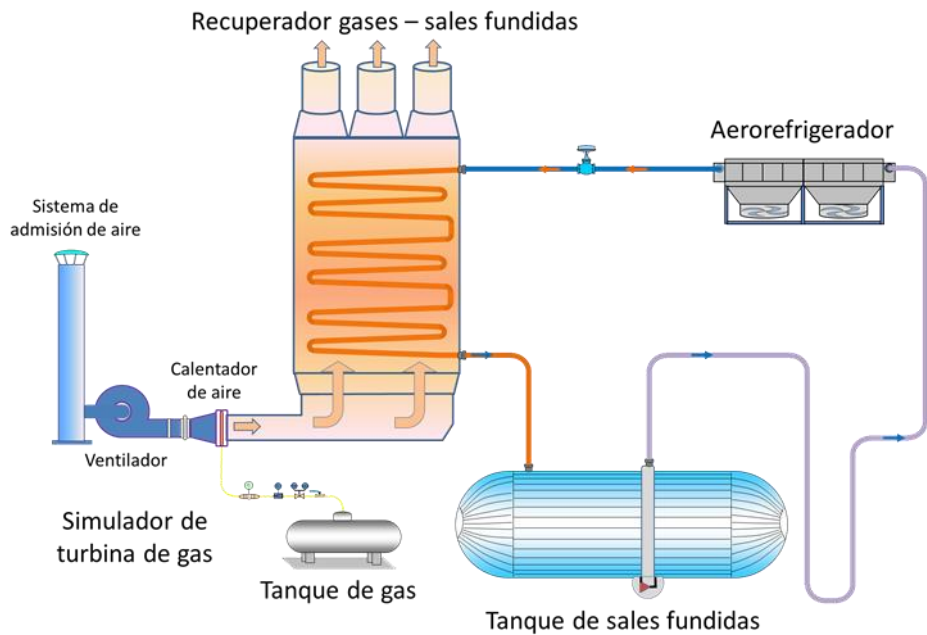


Figura 3. Configuración del demostrador del proyecto HYSOL

2.4. Herramienta software para operación y control del demostrador HYSOL

El centro de Investigaciones Energéticas, Medioambientales y tecnológicas – Plataforma Solar de Almería (a partir de ahora CIEMAT-PSA) es el coordinador del paquete de trabajo de modelado dinámico y control avanzado del proyecto HYSOL, una de las tareas es el desarrollo de una herramienta software para estudiar el comportamiento del demostrador [4]. El esquema de la herramienta software se muestra en la figura 4.

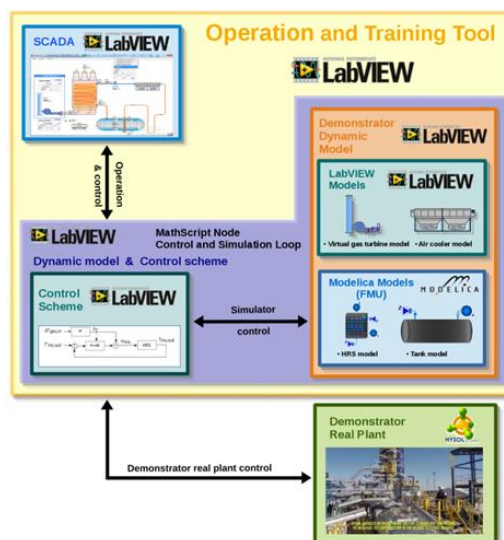


Figura 4. Esquema de la herramienta software del demostrador HYSOL

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

Los principales componentes de la herramienta son los siguientes:

- Modelo dinámico del demostrador: el modelo simula el comportamiento del demostrador real. Los diferentes modelos que lo componen están explicados posteriormente.
- Estrategias de control: los esquemas de control controlan las entradas del modelo del demostrador o la planta real de acuerdo con las consignas establecidas en el sistema SCADA; recibe la realimentación del modelo dinámico o la planta real dependiendo del modo de operación. Las estrategias de control se explicarán en el capítulo 3.
- Sistema SCADA: el modo de operación puede ser establecido en el sistema SCADA: modo simulación, modo planta real o modo híbrido. El SCADA también permite fijar las consignas del esquema de control y muestra los resultados del demostrador, tanto en su modo de simulación, en su modo real, como en ambos. El sistema SCADA al completo se explicará en el capítulo 4.

El modelo dinámico del demostrador está formado por los principales elementos anteriormente descritos en el demostrador [5]:

- Modelo de la turbina de gas virtual: se obtendrá teniendo en cuenta valores medidos de entrada y salida. Las entradas son la temperatura ambiente, la modulación de calor y la consigna del flujo másico de aire. Las salidas del modelo son la temperatura del aire de salida y el flujo másico (figura 5a).
- Modelo del recuperador de calor (HRS): para hacer este modelo se ha usado el lenguaje Modelica y la herramienta Dymola. Está basado en primeros principios, en balances de masas y de energías. Las entradas al modelo son las temperaturas de entrada de las sales fundidas y de los gases y los flujos másicos. Por su parte, las salidas del modelo son la temperatura de las sales fundidas y la de los gases (figura 5b).
- Modelo del tanque de sales fundidas: también se ha modelado usando el lenguaje Modelica y considerando primeros principios. El modelo tiene dos canales de sales fundidas, una de entrada y otra de salida. Se han considerado balances de masa y energía de las sales fundidas. El modelo también calcula el nivel del tanque. Las sales fundidas son extraídas por una bomba considerada ideal (figura 5c).
- Modelo del enfriador de aire: el enfriador de aire (*Air Cooler*), ha sido considerado ideal en el simulador (figura 5d).

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

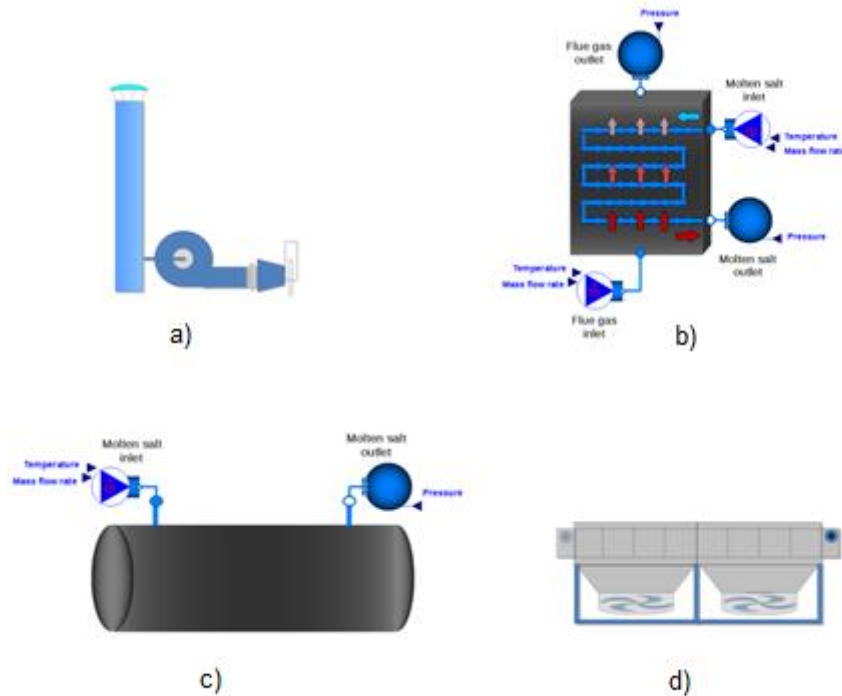


Figura 5:
a) turbina de gas virtual; b) HRS;

c) tanque de sales calientes; d) enfriador de aire

Existen tres modos de trabajo en la herramienta: modo simulación, modo planta real y modo híbrido.

- ✓ **Modo simulación:** en este modo, el sistema SCADA se comunica con el esquema de control que rige el modelo dinámico del demostrador. Este modo es muy útil para estudiar dinámicas del sistema, para probar nuevas estrategias de control en simulación [6] y en el contexto de simulación en tiempo real para entrenar y enseñar a futuros operadores de planta.
- ✓ **Modo planta real:** el sistema SCADA también se comunica con el esquema de control en este modo, pero controla la planta real del demostrador en lugar del modelo dinámico. El propósito de este modo es probar y validar nuevas estrategias de control que ya han sido previamente probadas en el modo de simulación.
- ✓ **Modo híbrido:** en este modo, el esquema de control controla simultáneamente la planta real del demostrador y el modelo de simulación, donde la realimentación que llega al esquema de control proviene de la planta real. Este modo es muy útil para estimar y comparar errores entre el modelo y la planta real. También es útil para calibrar y validar modelos dinámicos y estrategias de control. Una vez el modelo

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

dinámico se encuentre completamente validado, el modo híbrido también será útil para la detección de fallos en la planta real.

Este trabajo de fin de grado está enfocado en la realización del modo simulación de la herramienta software para operación y control del demostrador del proyecto HYSOL.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

3. Materiales y Métodos

3.1. Fases del proyecto

Este trabajo de fin de grado ha tenido una duración aproximada de realización en un periodo de 6 meses, en los cuales se ha trabajado a una media de 3 horas al día aproximadamente.

Para tener una visión gráfica de la duración en el tiempo y las diferentes actividades llevadas a cabo, se ha construido el siguiente cronograma:

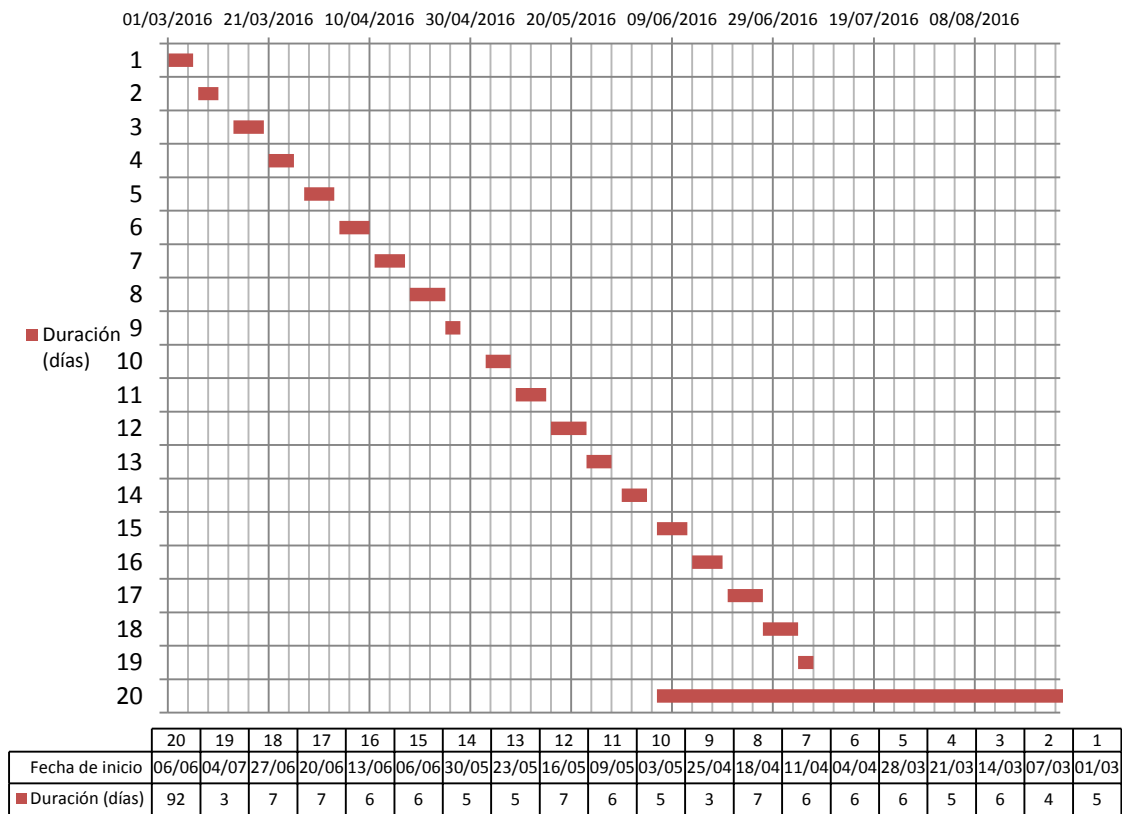


Tabla 2. Cronograma del trabajo de fin de grado

Las actividades realizadas son las siguientes:

- Actividad 1 (01/03/2016 a 05/03/2016): integración del módulo FMI y familiarización con el sistema.
- Actividad 2 (07/03/2016 a 10/03/2016): creación de la interfaz de usuario.
- Actividad 3 (14/03/2016 a 19/03/2016): creación de la lista de variables compartidas y estudio de su tecnología.
- Actividad 4 (21/03/2016 a 25/03/2016): creación de la inicialización del sistema (marcha/paro).
- Actividad 5 (28/03/2016 a 02/04/2016): creación de las gráficas en tiempo real y de las gráficas en tiempo de simulación.

- Actividad 6 (04/04/2016 a 09/04/2016): comprobación de la funcionalidad de la herramienta hasta la fecha, retroalimentación con los tutores y corrección de fallos.
- Actividad 7, 8 y 9 (11/04/2016 a 27/04/2016): implantación del lazo de control FIC-103. Verificación, validación y revisión por parte de los tutores.
- Actividad 10 y 11 (03/05/2016 a 14/05/2016): implantación del lazo de control FIC-104. Verificación, validación y revisión por parte de los tutores.
- Actividad 12, 13 y 14 (16/05/2016 a 03/06/2016): implantación del lazo de control de la Turbina de Gas Virtual. Verificación, validación y revisión por parte de los tutores.
- Actividad 15 (06/06/2016 a 11/06/2016): comprobación de la funcionalidad de la herramienta al completo hasta la fecha, retroalimentación con los tutores y corrección de fallos.
- Actividad 16 (13/06/2016 a 18/06/2016): inclusión de mejoras en la inicialización de todas las variables del sistema.
- Actividad 17 (20/03/2016 a 26/03/2016): inclusión de mejoras en la marcha y el paro de la herramienta.
- Actividad 18 (27/06/2016 a 03/07/2016): implantación del sistema de alarmas.
- Actividad 19 (04/07/2016 a 06/07/2016): revisión y aprobación final por parte de los tutores.
- Actividad 20 (06/06/2016 a 05/09/2016): redacción de la memoria del trabajo de fin de grado.

3.2. LabVIEW

El elemento principal que se ha usado para desarrollar esta herramienta ha sido el software LabVIEW. LabVIEW (acrónimo de Laboratory Virtual Instrumentation Engineering Workbench), creado por National Instruments, es una plataforma y entorno de desarrollo para diseñar sistemas y aplicaciones con la tarea de monitorear, adquirir, controlar, presentar y analizar datos y procesos en un modo real o de simulación.

LabVIEW [7] es una herramienta de programación gráfica, puesto que, a diferencia de otros sistemas de desarrollo comerciales que utilizan C o BASIC, utiliza un lenguaje de programación gráfico como el lenguaje G para crear programas basados en diagramas de bloques. Las principales ventajas que aporta el uso de LabVIEW son:

- ✓ Se reduce el tiempo de desarrollo de aplicaciones, ya que es más intuitivo.
- ✓ Proporciona gran flexibilidad al sistema.
- ✓ Posibilidad de incorporar aplicaciones escritas en otros lenguajes.
- ✓ Da la posibilidad al usuario de crear soluciones a problemas complejos de una manera más sencilla gracias a su programación gráfica.
- ✓ Generación de código automático.
- ✓ Existen múltiples módulos que se pueden incorporar para mejorar y potenciar la capacidad de LabVIEW.
- ✓ Programación multihilo.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

- ✓ Interfaz amigable.
- ✓ Comunicación rápida y eficaz entre diferentes componentes del sistema.

Originalmente, este software estaba orientado a aplicaciones de control de instrumentos electrónicos usadas en el desarrollo de sistemas de instrumentación, lo que se conoce como instrumentación virtual. Por esta razón, los programas creados en LabVIEW se conocen como VI (Virtual Instruments) y con esa misma extensión.

Los VIs tienen una parte interactiva con el usuario y otra parte de código fuente, donde se aceptan parámetros procedentes de otros VIs. Esas son sus dos ventanas principales: el panel frontal y el diagrama de bloques.

➤ Panel frontal: es la interfaz gráfica con el usuario, se utiliza para interactuar con el usuario. Recoge las entradas establecidas por el usuario y muestra las salidas proporcionadas por el programa. En esta interfaz se definen los controles que son valores de entrada (botones, marcadores, etc.) y los indicadores que son los valores de salida (gráficas, por ejemplo).

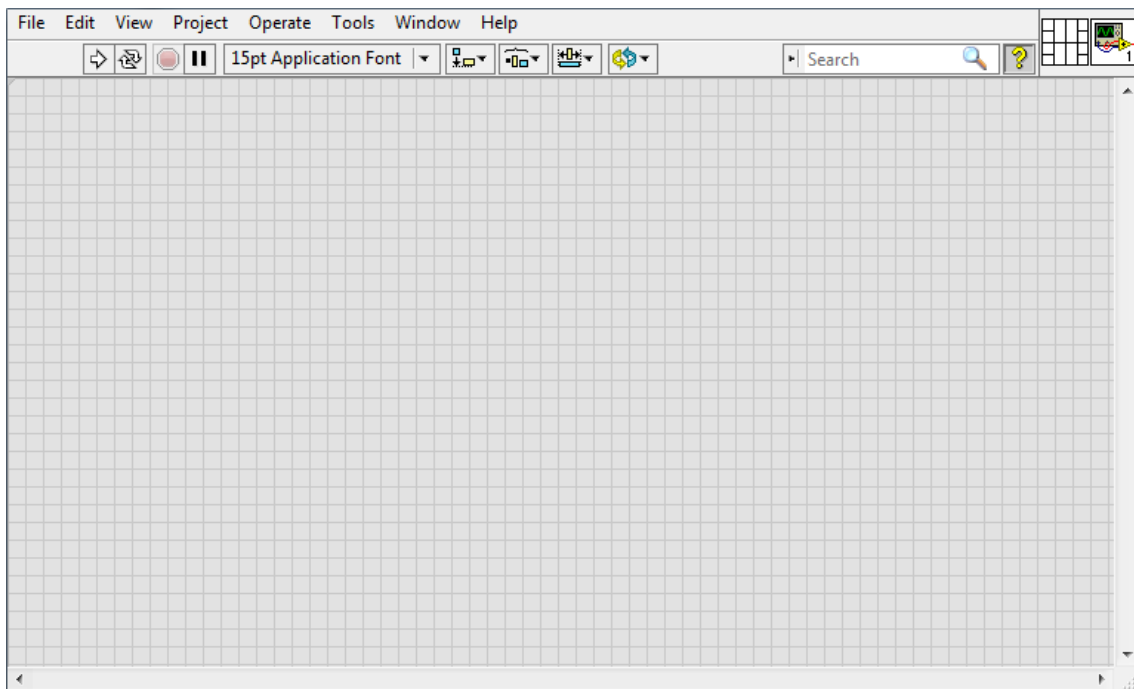


Figura 6. Panel Frontal

➤ Diagrama de bloques: el diagrama de bloques constituye el código fuente del VI. Es donde se realiza la implementación del programa VI para controlar o realizar cualquier procesamiento de las entradas y salidas que se crearon en el panel frontal. El diagrama de bloques incluye funciones y estructuras integradas en las librerías que incorpora LabVIEW. Los controles e indicadores que se colocaron previamente en el panel frontal se materializan en el diagrama de bloques mediante los terminales. Estos terminales son análogos a variables en lenguajes de programación basados en texto. El diagrama de bloques se

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

construye conectando los distintos objetos entre sí, como si de un circuito se tratara; por esos “cables” fluyen los datos del sistema.

LabVIEW tiene una amplia biblioteca de funciones, entre ellas, aritméticas, comparaciones, conversiones, funciones de entrada/salida, analíticas, etc.

Las estructuras, similares a las declaraciones causales y a los bucles en lenguajes de programación convencionales, ejecutan el código que tienen de forma condicional o repetitiva (bucle for, while, case, etc).

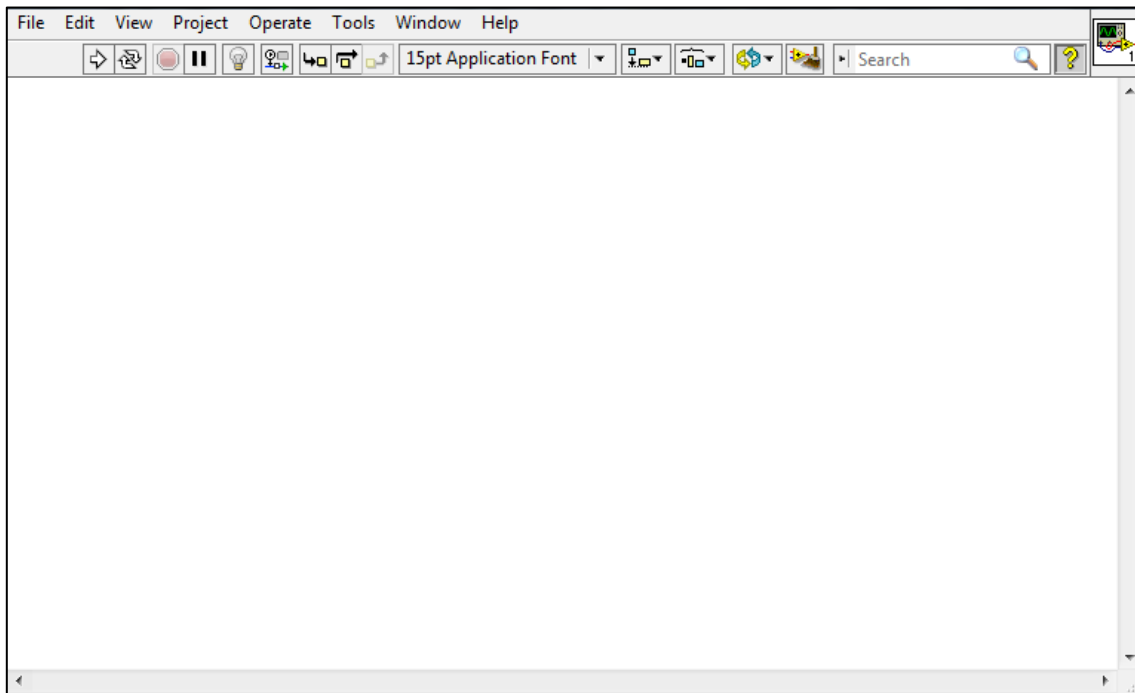


Figura 7. Diagrama de Bloques

- **Paletas:** las paletas de LabVIEW son las herramientas que se necesitan para crear y modificar tanto el panel frontal como el diagrama de bloques.
 - *Paleta de controles:* contiene los controles e indicadores que se utilizan para crear el panel frontal. Está dividida en varias categorías y según su estilo en varios submenús. Dentro de cada submenú hay otros menús que clasifican los controles e indicadores según el tipo de dato. Dentro de ellos se pueden encontrar, por ejemplo, los que sirven para dibujar gráficas y los que corresponden a datos de tipo numérico, booleano, textos, matrices, vectores y diferentes tipos compuestos:

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

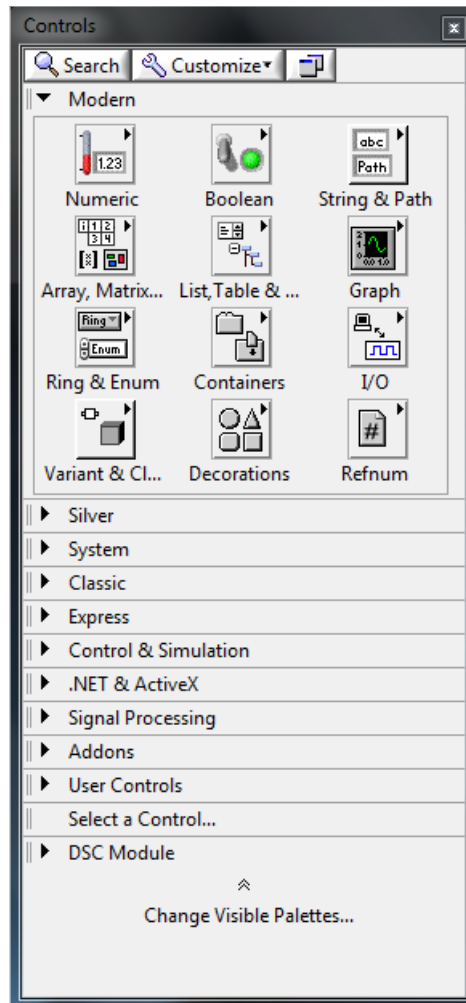


Figura 8. Paleta de Controles

- *Paleta de funciones:* esta paleta se utiliza en el diagrama de bloques para implementar el programa. La paleta de funciones contiene todos los objetos necesarios para la construcción del programa, ya sean funciones aritméticas, vectoriales, matriciales, de tipos compuestos de datos, estructuras de programación, funciones de entrada/salida de datos, adquisición de señales, subVIs, etc. Esta paleta también se divide en submenús:

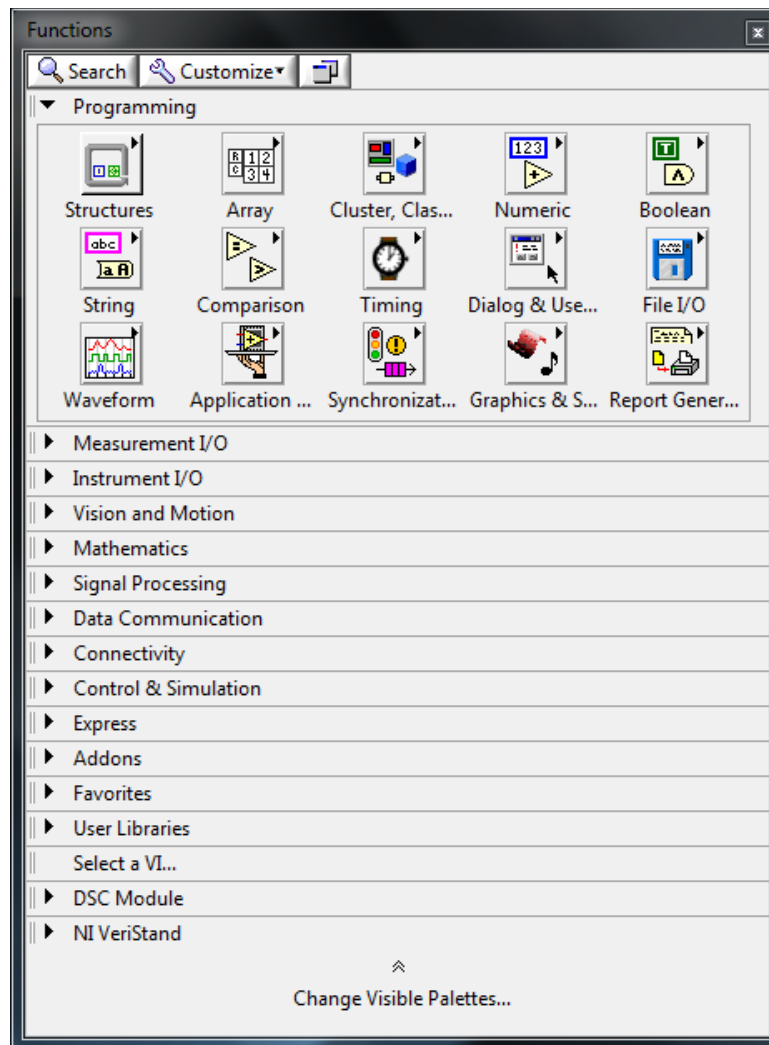


Figura 9. Paleta de Funciones

A su vez, en LabVIEW se pueden incluir diferentes módulos y complementos para hacer más completa y potenciar su capacidad de trabajo. Los módulos usados para la construcción de la herramienta desarrollada en el ámbito de este trabajo de fin de grado se explican a continuación en las siguientes secciones.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

3.3. Importación de modelos mediante el complemento FMI

Una vez que tenemos los modelos que simulan el comportamiento del demostrador real, hay que incluirlos en LabVIEW para poder trabajar con ellos.

El Complemento FMI (Functional Mock-up Interface) para VeriStand y LabVIEW de Dofware s.r.l. [8] permite la integración de modelos desarrollados con herramientas de modelado y simulación compatibles con FMI con software de National Instruments, que se ejecuten en el SO Windows en VeriStand y LabVIEW.

Como se ha explicado en el apartado 2.4., los modelos que se han de integrar con el complemento FMI son los del HRS y del tanque de sales calientes, que son los que se han creado en una herramienta externa a LabVIEW.

FMI es una herramienta estándar independiente que sirve para el intercambio de modelos y co-simulación de modelos dinámicos. La propuesta de uso de FMI es la siguiente; primero en un entorno de modelado de sistemas se crea el modelo dinámico; a continuación, se genera y se exporta el modelo en un archivo FMU (Functional Mock-up Unit). Entonces ya tenemos el modelo listo para poder ser usado en otro entorno de simulación.

Así se han importado los modelos para este simulador. Se obtuvieron los modelos en la herramienta Dymola usando el lenguaje Modelica y se obtuvo el ejecutable FMU que incluye los modelos diseñados.

3.4. Módulo MathScript

El módulo LabVIEW MathScript [9] añade matemática textual al entorno de desarrollo LabVIEW con un compilador original para los archivos “.m” que se hayan desarrollado en el software MATLAB. Se pueden combinar los enfoques textual y gráfico para desarrollo de algoritmos, procesamiento de señales, diseño de control y tareas de análisis de datos.

Este módulo permite integrar archivos “.m” personalizados al entorno gráfico de LabVIEW. Combinar los beneficios de matemática basada en texto con programación gráfica representa un nuevo enfoque híbrido de programación que ofrece la libertad de escoger la sintaxis más apropiada. En el centro de MathScript está un lenguaje de programación, muy similar a MATLAB, basado en texto de alto nivel con sintaxis y funcionalidad que abstrae la complejidad de las tareas de procesamiento de señales, análisis y matemáticas. Con más de 750 funciones integradas para ese trabajo, también se pueden crear nuevas funciones definidas por el usuario.

MathScript proporciona dos metodologías para programación: una interfaz interactiva y una programática. Diseñado para el desarrollo de scripts, la Ventana Interactiva de MathScript ofrece una interfaz de línea de comandos en la cual se puede cargar, almacenar, diseñar y ejecutar scripts de archivos “.m”. Diseñado para el desarrollo de scripts, el Nodo MathScript es una característica embebida de LabVIEW que conecta las variables de E/S basadas en texto con las entradas y salidas de LabVIEW.

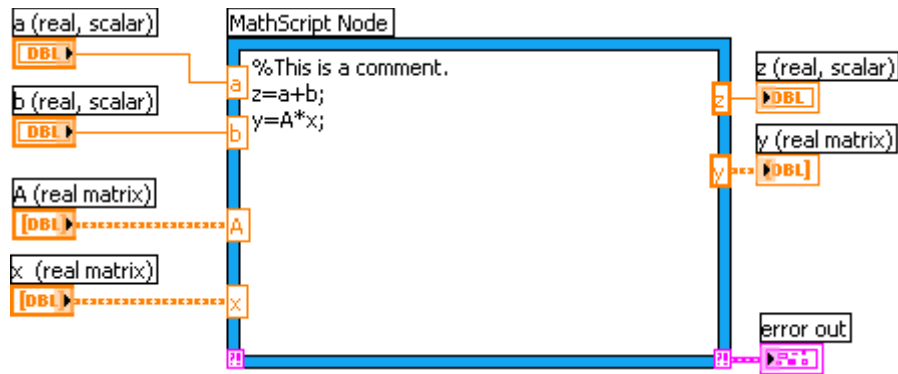


Figura 10. Ejemplo de Nodo MathScript usado en la herramienta

Diseñado para rendimiento óptimo en un sistema operativo en tiempo real, el motor MathScript toma los sinónimos de lenguaje escrito indirectamente con los entornos de archivos basados en texto y lo escribe estrictamente en el contexto del entorno gráfico de LabVIEW para una eficiente propagación de escritura de datos a través de código esencial. Esta escritura asegura que LabVIEW puede compilar de manera eficiente el código MathScript basado en texto para manejo de errores semántico y sintaxis al momento de la edición así como ayuda para contexto de nodo.

3.5. Módulo de Control y Simulación

El módulo LabVIEW Control Design and Simulation [9] permite analizar el comportamiento de modelos en lazo abierto, diseñar controladores de lazo cerrado, simular sistemas y realizar implementaciones físicas.

Se pueden crear modelos desde cero usando representaciones tipo función de transferencia o espacio de estado. Con las herramientas de análisis de tiempo y frecuencia, como respuesta al escalón en tiempo o gráficas de Bode, se pueden analizar de manera interactiva el comportamiento de lazos abiertos y cerrados. También hay herramientas integradas para sistemas de entrada múltiple, salida múltiple (MIMO) y de una sola entrada, una sola salida (SISO) y se pueden aplicar habilidades de simulación para verificar las dinámicas de sistemas lineales y no lineales.

Este módulo puede simular modelos implementados en el módulo LabVIEW MathScript basados en sintaxis MATLAB.

En la siguiente imagen (figura 11) se puede observar cómo el bucle se representa en LabVIEW. En el bloque superior izquierdo se configuran los parámetros que queremos a la hora de simular, como pueden ser el tiempo de simulación, el integrador numérico, el tamaño del paso en el caso de integradores numéricos de paso fijo, etc. En el bloque superior derecho solo se obtiene una salida en forma de reporte de errores.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

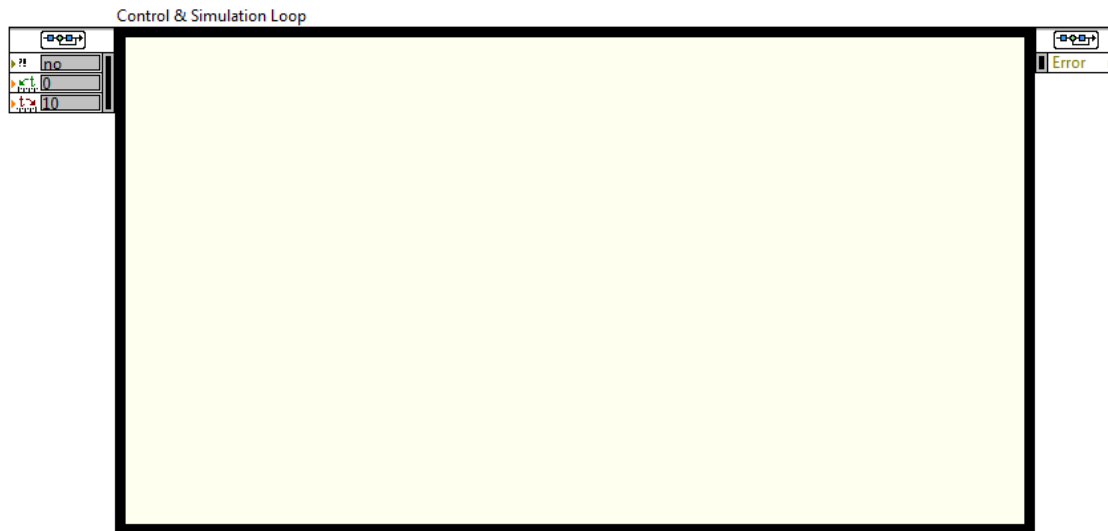


Figura 11. Bloque que representa el Lazo de Simulación y Control

Los menús que se utilizan para el desarrollo del control y simulación se pueden dividir en PID, Lógica Difusa, Simulación, Diseño de Control e Identificación de Sistemas. Dentro de cada menú hay diferentes submenús que nos permite escoger las funciones más adecuadas de entre un amplio abanico de posibilidades.

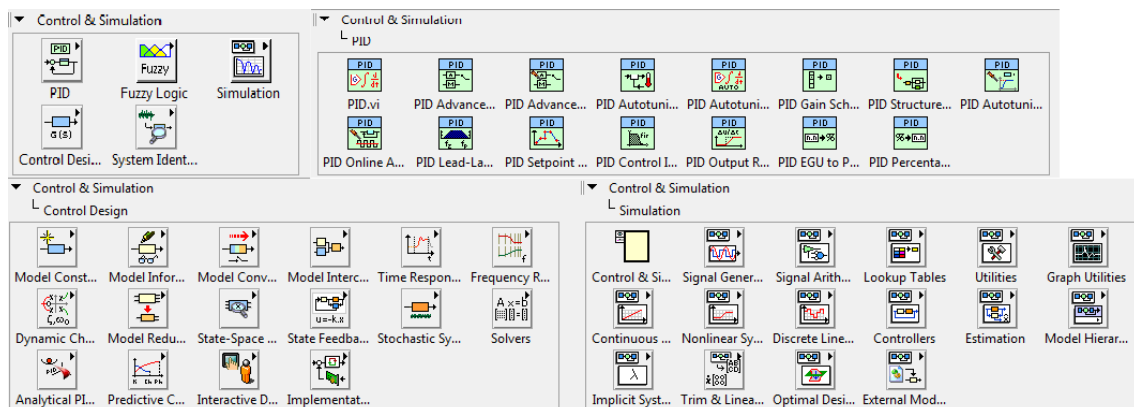


Figura 12. Paleta de funciones del módulo de Control y Simulación

3.6. Módulo de Supervisión y registro de datos

El módulo LabVIEW Datalogging and Supervisory Control (DSC) [9] es el complemento ideal para desarrollar un SCADA o aplicaciones de registro de datos de muchos canales. Con LabVIEW DSC, se pueden desarrollar de manera interactiva un sistema de monitoreo y control distribuido que va desde docenas hasta decenas de miles de etiquetas. Incluye herramientas para registrar datos a una base de datos histórica en red, rastrear tendencias de datos históricos y en tiempo real, administrar alarmas y eventos y añadir seguridad a interfaces de usuarios.

El módulo DSC también mejora el rendimiento de las variables compartidas en LabVIEW. Las variables compartidas se pueden usar para acceder e intercambiar datos a través de VIs en un proyecto de LabVIEW o a través de una red. Con DSC, se pueden registrar los datos automáticamente; añadir alarmas y seguridad a las variables compartidas y configurar la programación de las variables compartidas.

El módulo DSC, al igual que otros módulos, tiene diferentes funciones disponibles para la hora de programar en LabVIEW. Básicamente tiene funciones para el manejo de las alarmas, trabajar con datos históricos, variables compartidas y seguridad.

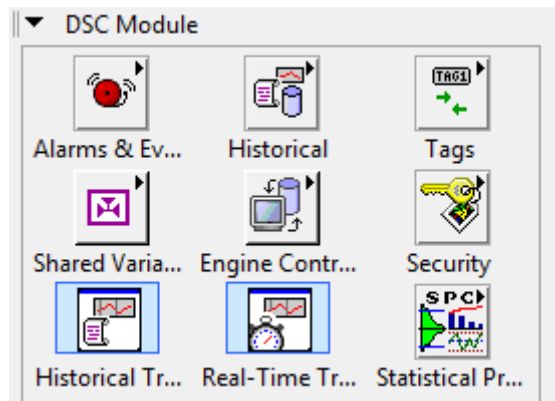


Figura 13. Paleta de funciones del módulo DSC

3.7. Teoría de control

A la hora de llevar a cabo la implementación de la herramienta para la simulación, se ha tenido en cuenta, el demostrador de la planta real y todos sus componentes. Una de las partes más importantes son los esquemas de control.

En esta herramienta hay tres lazos de control principales:

1. Control de la temperatura de las sales fundidas a la salida del HRS con el flujo másico de las sales fundidas como variable de control (también llamado **FIC-103**).
2. Control de la temperatura de las sales fundidas a la salida del HRS con el flujo másico de gases como variable de control (también llamado **FIC-104**).

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

3. Lazo de control de la turbina de gas. Mantiene la temperatura de gas a la entrada del HRS a una temperatura deseada a pesar de las perturbaciones de los gases de escape de la turbina de gas.

Estos lazos de control se explicarán en detalle en el capítulo 4.

Para poder conseguir que estos lazos de control actúen de manera apropiada, se han de tener en cuenta las diferentes teorías de control que se conocen y que se han utilizado para el diseño de estos bucles [10], [11] y [12].

3.7.1. Control PID

Un controlador PID es un mecanismo de control por realimentación ampliamente usado en sistemas de control industrial. Calcula la desviación o error entre un valor medido y un valor deseado. Tiene numerosas ventajas, entre las que destacan que es sencillo, potente y robusto.

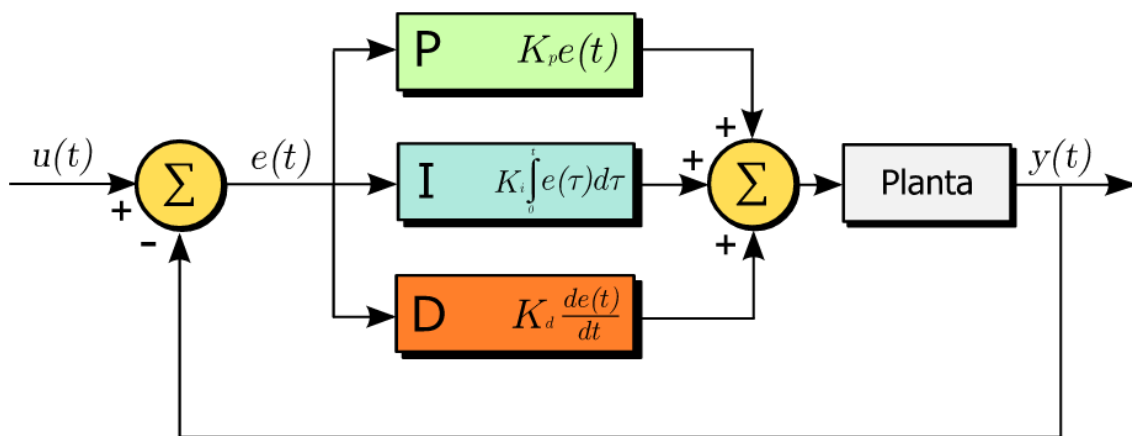


Figura 14. Esquema clásico algoritmo PID

El algoritmo del control PID consta de tres parámetros distintos: el proporcional, el integral, y el derivativo. El valor Proporcional depende del error actual; el control proporcional incrementa el valor de la salida cuanto mayor sea el error. El Integral depende de los errores pasados; el control integral incrementa el valor de la salida cuanto más tiempo perdure el error y el Derivativo es una predicción de los errores futuros; el control derivativo incrementa la salida cuanto más rápidamente se produce el error (adelanto del error). La suma de estas tres acciones es usada para ajustar el proceso por medio de un elemento de control como puede ser una válvula o cualquier actuador.

Este algoritmo se puede considerar como un control proporcional, que dispone de un control integral para eliminar el error en estado estacionario y un control derivativo para mejorar la estabilidad y aumentar la rapidez de la respuesta.

Ajustando estas tres variables en el algoritmo de control del PID, el controlador puede proveer una acción de control diseñada para los requerimientos del proceso específico. La respuesta del controlador puede describirse en términos de la respuesta del control ante un

error, el grado el cual el controlador sobrepasa el punto de ajuste, y el grado de oscilación del sistema.

Combinando estas tres variables, se pueden obtener diferentes configuraciones básicas, como por ejemplo las siguientes:

- Control Proporcional (P): da una salida del controlador que es proporcional al error, es decir:

$$u(t) = K_p * e(t)$$

con lo que el controlador es solo la K_p que es una ganancia proporcional ajustable $C(s) = K_p$

- Control Proporcional-Integral (PI): se define mediante:

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(\tau) d\tau$$

donde T_i se denomina tiempo integral y es el cual ajusta la acción integral. Entonces, el controlador es de la siguiente forma:

$$C(s) = K_p \left(1 + \frac{1}{T_i s} \right)$$

Con un control proporcional, es necesario que exista error para tener una acción de control distinta de cero. Con acción integral, un error siempre nos dará una acción de control, por lo tanto el error en régimen permanente será siempre cero. Muchos procesos industriales utilizan este tipo de controlador.

- Control Proporcional-Derivativo (PD): se define mediante:

$$u(t) = K_p e(t) + K_p T_d \frac{de(t)}{dt}$$

donde T_d es una constante denominada tiempo derivativo. Esta acción tiene carácter de previsión, lo que hace más rápida la acción de control, aunque tiene la desventaja importante de amplificar las señales de ruido y puede provocar saturación en el actuador. La acción de control derivativa nunca se utiliza por sí sola, debido a que solo es eficaz durante períodos transitorios. El controlador toma la siguiente forma:

$$C(s) = K_p (1 + T_d s)$$

Cuando una acción derivativa se agrega a un controlador proporcional, permite obtener un controlador de alta sensibilidad, es decir, que responde a la velocidad del cambio del error y produce una corrección significativa antes de que la magnitud del error se vuelva demasiado grande. Aunque el control derivativo no afecta en forma directa al error en estado estacionario, añade amortiguamiento al sistema y, por tanto, permite un valor más grande que la ganancia K_p , lo cual provoca una mejora en la precisión en estado estable.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

- Control Proporcional-Integral-Derivativo (PID): esta acción combinada reúne las ventajas de cada una de las tres acciones de control. Se define mediante:

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(\tau) d\tau + K_p T_d \frac{de(t)}{dt}$$

y el controlador toma la siguiente forma:

$$C(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s \right)$$

3.7.2. Esquema anti-windup. Saturación del efecto integral

Todos los actuadores presentan limitaciones físicas y en multitud de ocasiones el sistema de control puede alcanzar dichos límites. Esto da lugar a que el lazo de control se rompa y el sistema pasa a trabajar en lazo abierto ya que el actuador quedará en sus valores límites máximos o mínimos independientemente de lo que ocurra en la salida del proceso.

En caso de que el controlador posea acción integral, este problema se acentúa ya que el término integral puede llegar a tomar valores muy elevados (lo que se conoce como “it winds up”). Esto suele dar lugar a que el término integral tome un valor de signo contrario durante un periodo largo de tiempo cuando el sistema realmente salga de la saturación.

Existen diversos métodos para paliar este efecto, con diferentes esquemas y uno de los más usados es el anti-windup.

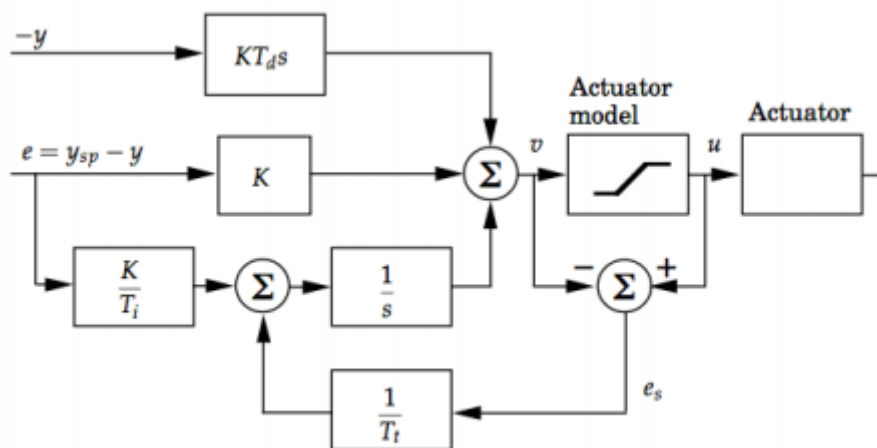


Figura 15. Esquema anti-windup

El objetivo es hacer que la entrada a la integral (integrador) sea nula.

Cuando la señal de control que entra al actuador es superior a la capacidad del propio actuador, el esquema anti-windup actúa restando la salida saturada del actuador con la señal de entrada al actuador. Así no se irá acumulando el valor en la integral y actuará

rápido cuando el sistema deje de entrar en saturación y sin cambios bruscos en la señal de control. Este esquema además adhiere un nuevo parámetro, T_t llamada constante de tracking cuya función es determinar el ratio con el que se pretende que el término integral sea reseteado.

3.7.3. Transferencia sin saltos

La gran mayoría de los controladores pueden trabajar en dos modos: manual y automático. Eso es bastante común en el control de sistemas no lineales, donde el sistema se puede llevar al punto de operación deseado de forma manual y posteriormente se conmuta al modo automático. Debido a que el controlador es un sistema dinámico, cuando se produzca la conmutación de un modo a otro es necesario asegurarse que el estado del sistema es el correcto, para evitar cambios pronunciados en la señal de control.

El principal problema del salto en la señal de control viene dado porque el estado de la componente integral del controlador no es el correcto en el momento del cambio. Un esquema ampliamente utilizado en los controladores PID es el que se basa en el esquema anti-windup anteriormente explicado.

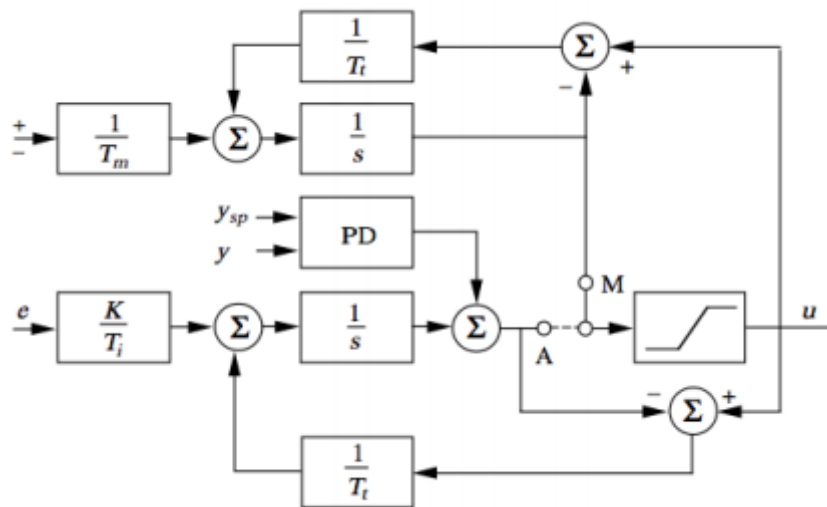


Figura 16. Esquema transferencia sin saltos

El esquema consiste en estar comunicando al sistema continuamente del valor de la señal de control tanto en el modo manual como en el modo automático. Este mecanismo de seguimiento funciona adecuadamente para cambios de modos en estado estacionario, en transitorio, presencia de perturbaciones, etc., ya que realiza un seguimiento continuo de la señal en todo momento.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

3.7.4. Control anticipativo (*feedforward*)

Con un lazo de realimentación, el efecto de las perturbaciones se puede atenuar de forma considerable, pero presenta varias limitaciones:

- ✓ La mejora del rendimiento en la respuesta frente a perturbaciones está limitada y debe estar asociada a disponer de un buen compromiso entre rendimiento y esfuerzo de control.
- ✓ La respuesta frente a un efecto de la perturbación se realiza siempre a posteriori, una vez que dicha perturbación ha afectado a la variable a controlar.

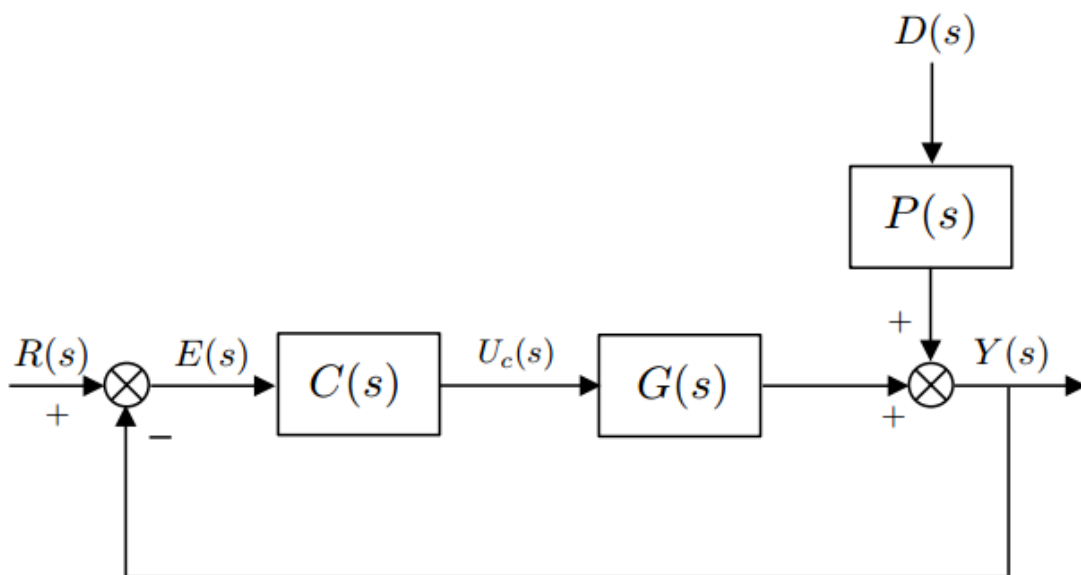


Figura 17. Perturbación “D(s)” en esquema de control

El esquema de control anticipativo o *feedforward* permite solventar las limitaciones anteriormente comentadas de la realimentación para el problema de rechazo a perturbaciones.

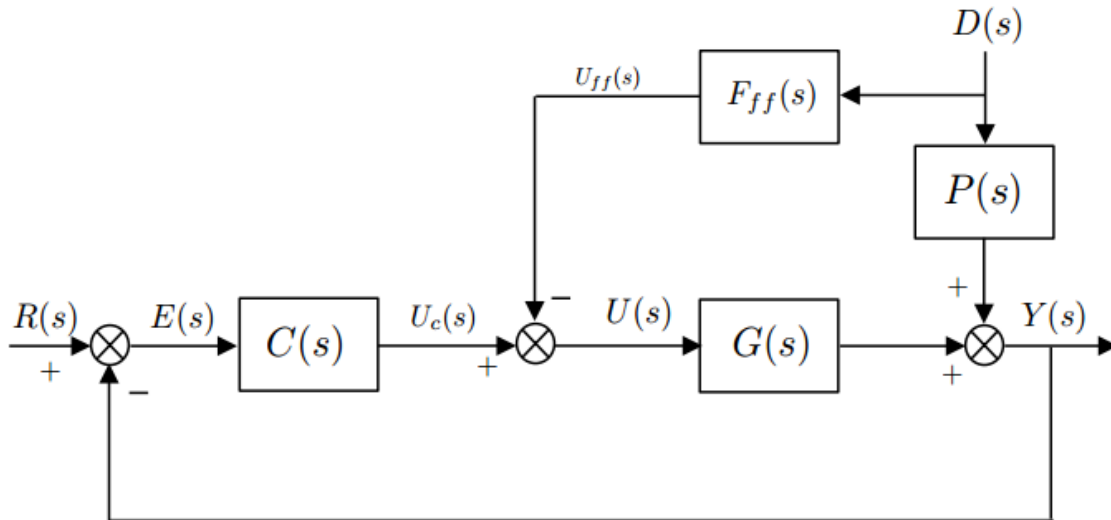


Figura 18. Esquema de control con *feedforward*

El esquema *feedforward* consiste en adelantarse al efecto de la perturbación, conociendo ya su dinámica con anterioridad, gracias a ello se puede calcular el controlador $F_{ff}(s)$. El *feedforward* compensa la señal de control para que la salida del sistema sea la correcta tras el efecto de la perturbación.

3.7.5. Control en cascada

Es una estrategia de control que posee importantes ventajas sobre un lazo de control simple, y se puede utilizar cuando se disponen de varias medidas del proceso y una única variable de control. Es útil cuando existen dinámicas lentas entre la variable de control y la salida a controlar del sistema. Se suele utilizar para:

- ✓ Compensar perturbaciones que afectan a medidas intermedias o secundarias del proceso, con el fin de atenuar su efecto sobre la salida principal del proceso.
- ✓ Reducir el efecto de no linealidades del sistema dadas por variables secundarias (generalmente actuadores) del sistema para atenuar su efecto sobre la salida del sistema a controlar.

La estrategia de control se construye haciendo uso de dos lazos de control, denominados lazo primario, también llamado maestro, y lazo secundario, también llamado esclavo. En definitiva, la salida del controlador maestro envía una consigna al segundo lazo de control, donde el controlador esclavo opera según le dicta el maestro. Este lazo de control interno es usual cuando se quiere controlar la temperatura o el caudal de algún fluido y se necesita tener un control del actuador.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

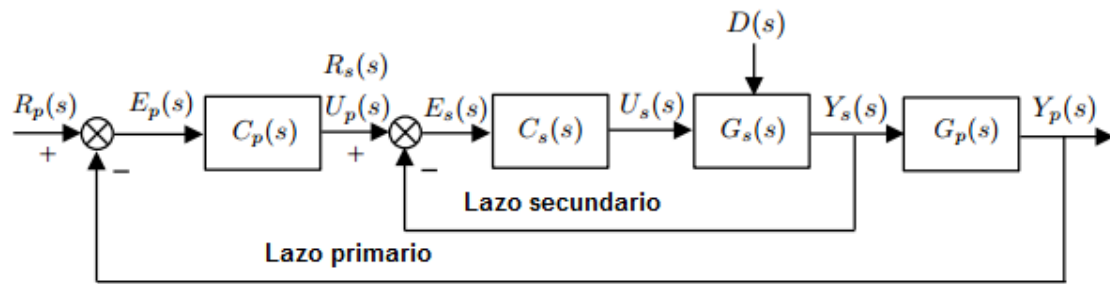


Figura 19. Esquema de control en cascada

Para que el control en cascada sea eficaz, hay que escoger correctamente la variable secundaria. A la hora de su elección, hay que tener en cuenta varios aspectos:

- ✓ Debe existir una correcta relación entre la variable primaria y secundaria.
- ✓ Las principales perturbaciones y no linealidades deben afectar al lazo de control esclavo.
- ✓ El lazo interno debe ser más rápido que el externo.

La puesta en marcha de una estructura en cascada es importante y requiere una serie de pautas para no provocar saltos bruscos e innecesarios en la señal de control. Primero se ponen ambos lazos en modo manual; entonces se establece la referencia del lazo secundario poniendo el valor correspondiente a la salida del controlador y se pone en modo automático. Se repite el procedimiento ahora para el lazo primario, poniendo la salida de su controlador en el valor de referencia.

4. Sistema SCADA desarrollado

4.1. Interfaz gráfica

En este capítulo se va a proceder a explicar en detalle el desarrollo del sistema SCADA al completo, el cual se va a realizar en el entorno LabVIEW como se ha mencionado anteriormente. A la hora de hacer un trabajo de esta envergadura en LabVIEW, es importante saber usar las distintas funcionalidades, tipos de archivo y recursos que tiene.

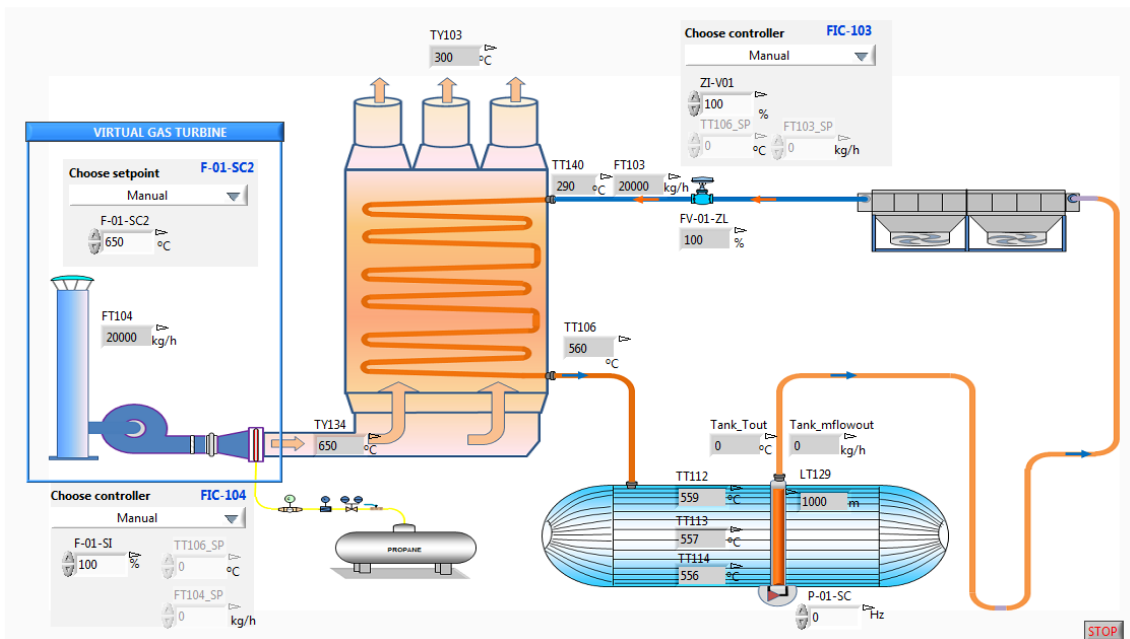


Figura 20. Sinóptico de la planta

Dentro de este ámbito existe la posibilidad de aglomerar todos estos procedimientos en uno solo para así tener un rápido acceso a los distintos VIs y poder ejecutarlos directamente en el mismo directorio y combinarlos entre sí. Esto se consigue gracias a la creación de proyectos en LabVIEW (*project*), con los cuales también se podrán crear librerías y listas de variables compartidas, crear aplicaciones, además de poder crear una jerarquía de VIs dependiendo de nuestras necesidades. Directamente se puede manejar todo el proyecto con todos los archivos requeridos en la misma ventana, y se puede interactuar entre ellos. Se pueden clasificar los archivos del proyecto automáticamente y no se altera la organización del proyecto en sí.

En la ventana de explorador de proyectos se tiene toda la información del proyecto en sí. En dicha ventana existen dos pestañas, una llamada *Items* que muestra los elementos del proyecto en un diagrama con estructura de árbol y otra llamada *Files* que muestra los elementos del proyecto que tienen archivos correspondientes en memoria.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

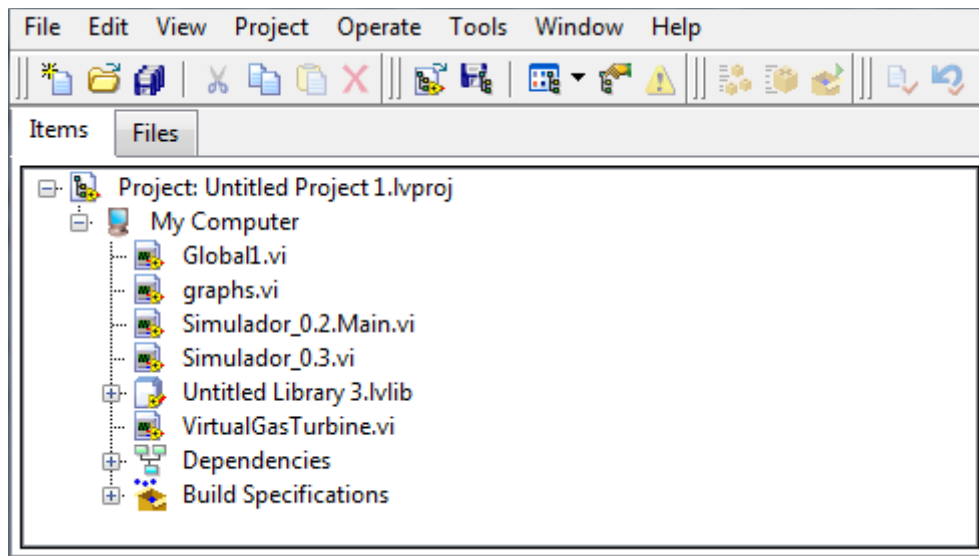


Figura 21. Ventana de explorador de proyecto

A partir de la creación del proyecto, se pueden crear librerías donde se clasificarán todas las variables del proceso y se almacenarán todos los datos relevantes de la planta. Dentro de esta librería se declaran las variables compartidas, las cuales se usarán para almacenar datos, crear históricos, y enlazar datos de un VI a otro.

Asimismo, los VIs serán creados e integrados en el proyecto para su posterior sincronización y utilización de forma adecuada. El VI principal donde está construida la herramienta es el llamado “Simulador_0.3”.

El SCADA creado en LabVIEW se compone de varias pestañas intuitivas que nos mostrarán toda la información necesaria para tener un control completo del sistema. Estas pestañas son las siguientes:

- **“INIT”**: su función es establecer los valores iniciales con los cuales empezará a trabajar la herramienta.
- **“MAIN”**: es la pestaña principal donde se muestra el sinóptico que representa el demostrador de la planta y donde se podrán observar los indicadores y variables características del proceso y dar instrucciones de control.
- **“GRAPHS”**: en esta pestaña se mostrarán las gráficas en tiempo real de todos los parámetros del proceso a elección del usuario.
- **“SIM GRAPHS”**: como en la pestaña anterior, las gráficas, en este caso ejecutándose acorde al tiempo de simulación, serán mostradas.
- **“CONTROL”**: esta importante pestaña sirve para establecer y modificar cuando sea necesario los parámetros de los diferentes lazos de control que actúan en el sistema.
- **“ALARMS”**: el cometido de esta ventana es informar al usuario cuándo se ha producido una anomalía y de qué tipo, avisando en forma de alarma.

Esta interfaz gráfica creada en el panel frontal de LabVIEW tiene su equivalente en el diagrama de bloques.

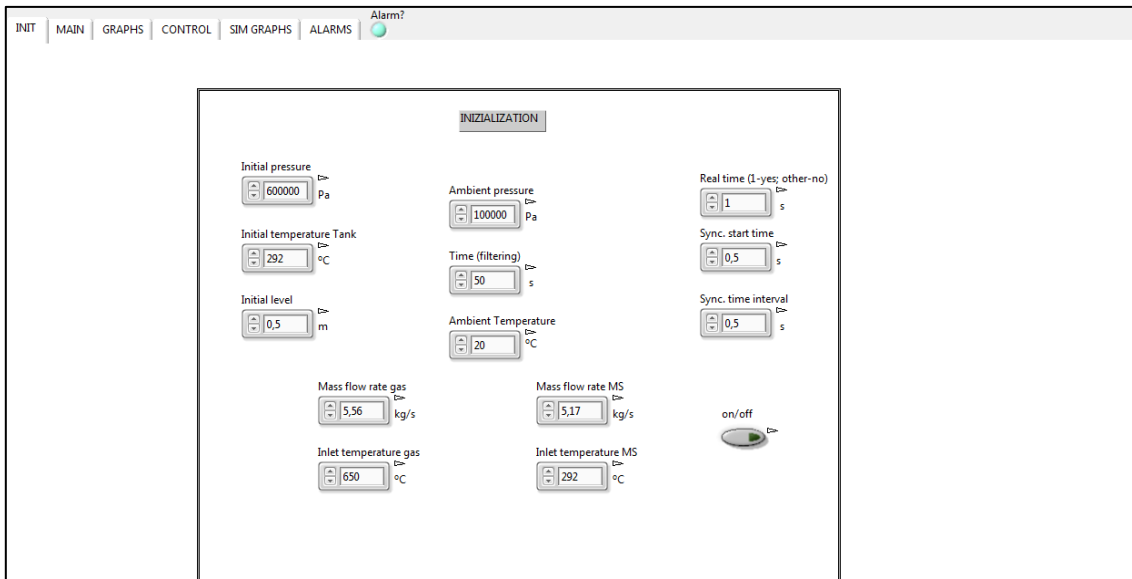


Figura 22. Visualización general del SCADA – pestaña “INIT”

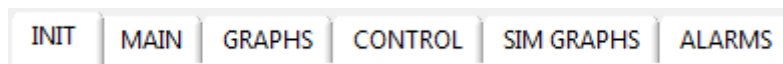


Figura 23. Detalle de las pestañas del SCADA

El sistema SCADA se engloba en una estructura *Stacked Sequence*, esto es, una estructura que ordena la ejecución del código que está en su interior. A diferencia de los lenguajes secuenciales de programación, en los cuales el orden de ejecución del código se corresponde con el orden en el cual están las instrucciones escritas, en LabVIEW se sigue el modelo de flujo de datos; por tanto, este tipo de estructura sirve para ordenar la ejecución según se desee.

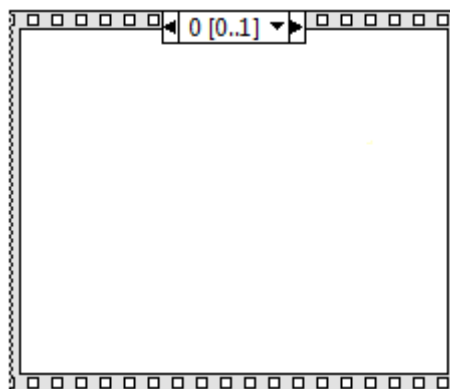


Figura 24. Ejemplo de *Stacked Sequence*

Con el *Stacked Sequence* se consigue que haya diferentes diagramas que se ejecuten en orden, llamados subdiagramas o “frames”; una vez acabado uno, se pasa al siguiente. En la parte superior de la estructura, se aprecia un identificador del subdiagrama, que nos informa qué subdiagrama está seleccionado.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

Dentro de esta estructura se implementa toda la herramienta. Se usarán tres *frames* o subdiagramas:

1. El primer *frame* incluye una instrucción para que se inicialice el SCADA en la primera pestaña, la de inicialización, haciendo uso de una constante para que cambie el valor de la pestaña.

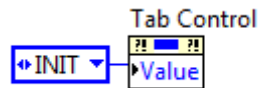


Figura 25. Inicialización de las pestañas

2. El segundo *frame* consiste en un clásico bucle *For* que recorre todas las pestañas del SCADA ejecutando el contenido de cada pestaña dependiendo en la que esté. A su vez, se ejecuta una instrucción dentro de una estructura *Case*, equivalente a un *Switch* en lenguajes clásicos de programación. Las instrucciones dependen de qué valor tome la pestaña.



Figura 26. Estructura bucle *For* más el *Case* acoplado

La pestaña *GRAPHS* y la pestaña *MAIN* invocan a otros VIs creados a parte para que se integren en el VI principal y se ejecuten en conjunto. El resto de pestañas se dedican a inicializar valores de controles.

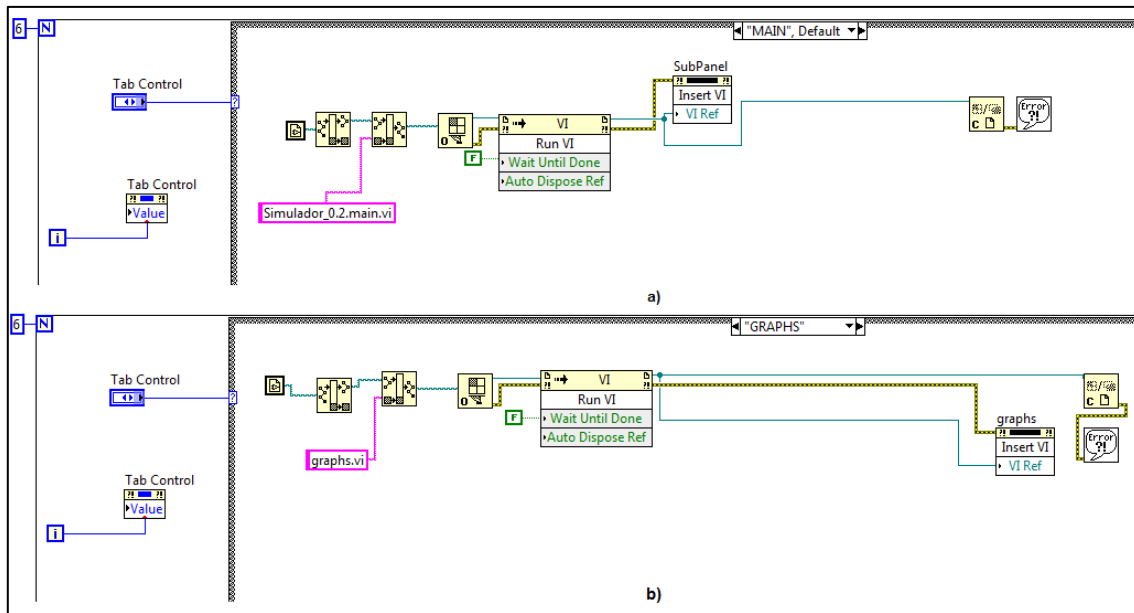


Figura 27. a) invocación del VI “MAIN”; b) invocación del VI “GRAPHS”

3. Una vez ejecutados estos dos *frames*, se procede a ejecutarse el último *frame*, que es el principal, en el cual se incluye el lazo de control y simulación y todos sus accesorios. Dentro de este bucle se van simular todos los modelos de la planta, y uno de los más importantes es el modelo del HRS y del tanque de sales fundidas.

Estos modelos han sido importados a través del complemento FMI y toma la siguiente forma en el diagrama de bloques:

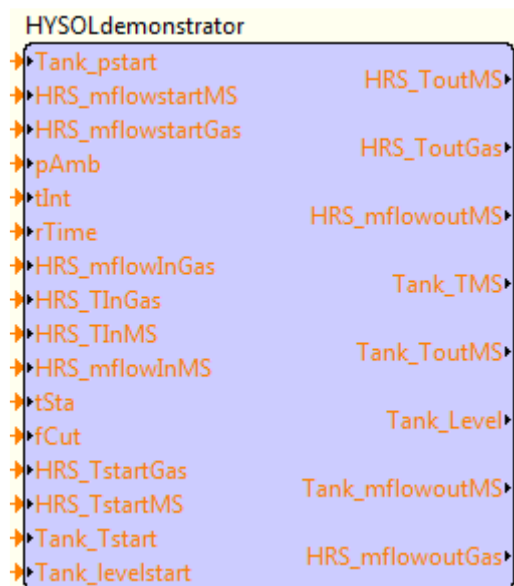


Figura 28. Módulo del modelo importado a través del FMI

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

Donde se pueden observar todas las entradas, condiciones iniciales y salidas que tiene, las cuales se utilizarán a lo largo de la creación de la herramienta:

Entradas y condiciones iniciales	Salidas
Presión inicial del tanque	Temperatura de las sales fundidas a la salida del HRS
Flujo másico inicial de sales fundidas a la entrada del HRS	Temperatura de los gases a la salida del HRS
Flujo másico inicial de gases a la entrada del HRS	Flujo másico de las sales fundidas a la salida del HRS
Presión ambiente	Temperatura de las sales fundida en el tanque
Intervalo de tiempo de sincronización	Temperatura de las sales fundidas a la salida del tanque
Tiempo real	Nivel del tanque
Flujo másico de gases circulante a la entrada del HRS	Flujo másico de las sales fundidas a la salida del tanque
Temperatura de gases a la entrada del HRS	Flujo másico de los gases a la salida del HRS
Temperatura de sales fundidas a la entrada del HRS	
Flujo másico de sales fundidas circulante a la entrada del HRS	
Tiempo de comienzo de la sincronización	
Tiempo de filtrado	
Temperatura inicial de los gases a la entrada del HRS	
Temperatura inicial de las sales fundidas a la entrada del HRS	
Temperatura inicial del tanque	
Nivel inicial del tanque	

Tabla 3. Entradas y salidas del modelo del demostrador

4.2. Variables compartidas

Las variables compartidas son una forma más genérica aún de compartir información. En este caso la comunicación puede ser entre programas de ordenadores distintos, entre lazos de control o VIs de todo un proyecto. A diferencia de la mayoría de métodos para compartir datos en LabVIEW, las variables compartidas se suelen configurar usando diálogos de propiedades. Así, el traspaso de información es transparente para el programador, que no tiene que preocuparse por el protocolo de transmisión y otros detalles. Otra de las características importantes de las variables compartidas es la capacidad que tienen de registrar datos.

Las variables compartidas deben ser creadas en la librería del proyecto. Existen tres tipos de variables compartidas: *single-process*, *network-published* y para valores de entrada salida I/O.

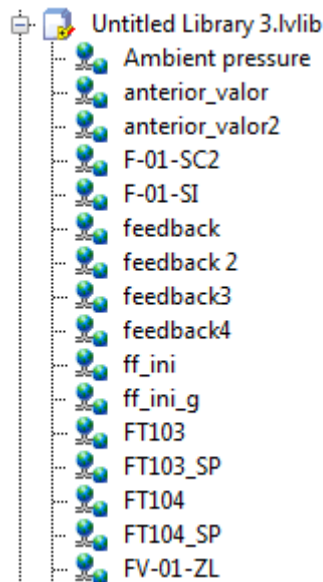


Figura 29. Variables compartidas en librería de proyecto

El tipo de variables usadas en este proyecto son las *network-published* ya que tienen múltiples opciones de configuración. Por tanto, se define el nombre, el tipo de variable y el tipo de dato (doble, entero, lógico o booleano, etc) en la ventana de configuración.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

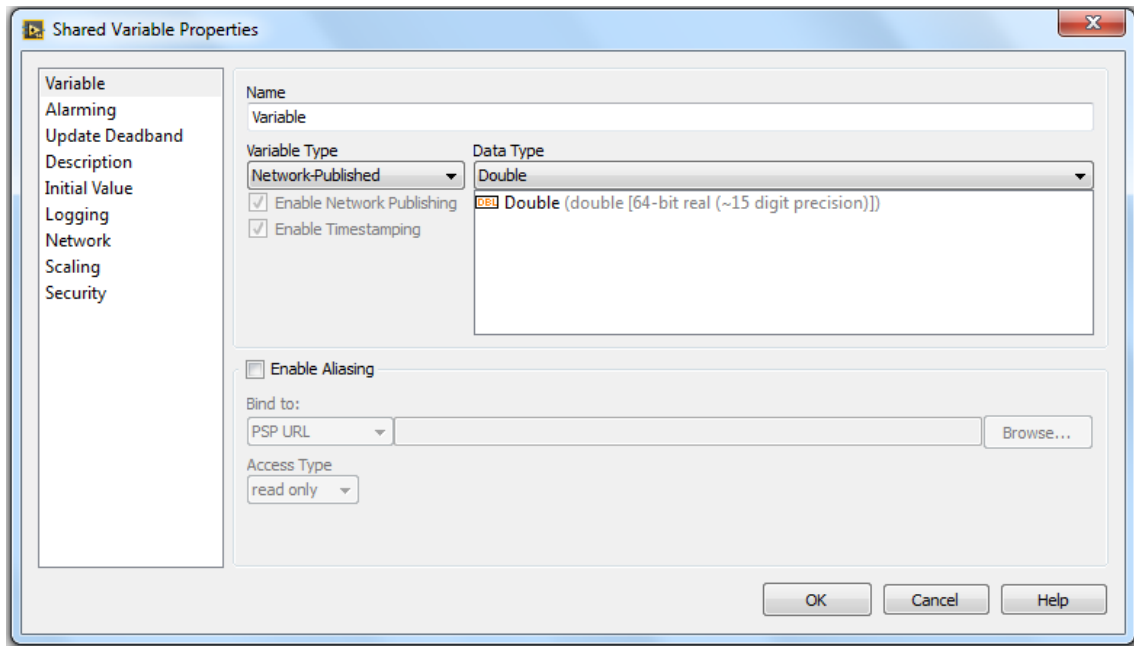


Figura 30. Ventana de configuración de variables compartidas

La opción *Alarming* habilita el uso de alarmas y a su vez, *Logging* permite el registro de las alarmas y los eventos generados por éstas, además de almacenar los valores de los datos para el visualizado de históricos; estas opciones serán usadas más adelante en la sección Alarmas.

A la hora de hacer uso de las variables compartidas en los diagramas de bloques de los VIs del proyecto, se utiliza un nodo llamado *Shared Variable Node* (ver figura 23). La variable tiene dos modos de acceso: de lectura y de escritura. En el modo lectura las variables solo leen los datos que se almacenan en ellas, mientras que en el modo escritura se permite especificar el valor de la variable debido a que un control está actuando sobre ellas. Finalmente el modo utilizado en las variables compartidas del proyecto es el de lectura/escritura, con el cual las variables están constantemente actualizadas tanto como si reciben datos como si los proporcionan.



Figura 31. a) modo lectura; b) modo escritura

También existe la opción de asignar la tecnología de las variables compartidas a un simple control o indicador que haya en el panel frontal. Esta característica se llama *Data Binding* (enlace de datos) y permite que se comporte como una variable compartida. Mientras el VI está ejecutándose, si la conexión es satisfactoria, un pequeño indicador verde se mostrará en el control o indicador. Así como se usaba en los nodos de variables compartidas, también se pueden poner en modo lectura, en modo escritura o en modo lectura/escritura; siendo éste último el que se ha utilizado.

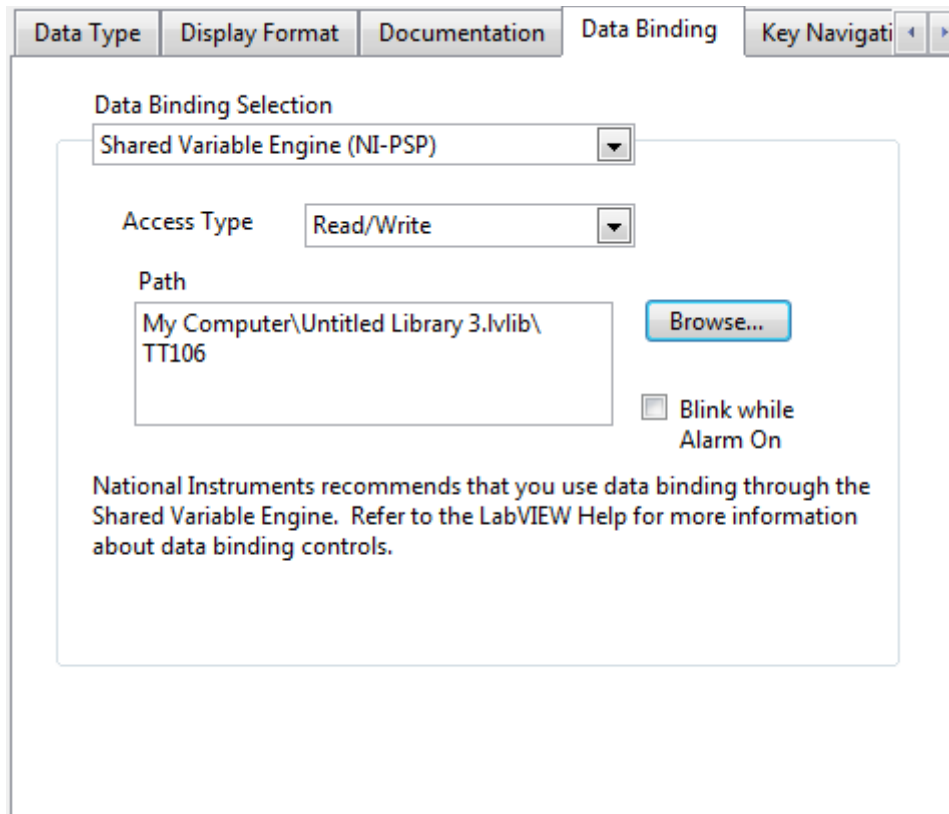


Figura 32. Enlace de datos con variable compartida (*Data Binding*)

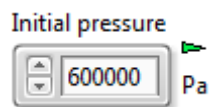


Figura 33. Indicador verde de conexión satisfactoria con variable compartida

Debido a esta amplia potencia y simplicidad que nos proporcionan las variables compartidas, han sido usadas para la realización de este proyecto. Múltiples variables han sido creadas con diferentes fines, unas para el registro de datos y visualización de los parámetros del sistema, y otras con el objetivo de apoyar y ayudar a que la herramienta tenga un funcionamiento optimizado. A continuación se presenta una tabla con la clasificación de todas las variables compartidas creadas:

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

Nombre variable	Tipo de dato	Descripción	Alarming	Logging
Ambient pressure	Doble	Presión (Pa)	No	Sí
anterior_valor	Doble	Variable de apoyo	No	No
anterior_valor2	Doble	Variable de apoyo	No	No
F-01-SC2	Doble	Temperatura (°C)	No	Sí
F-01-SI	Doble	Porcentaje válvula (%)	No	Si
feedback	Doble	Variable de apoyo	No	No
feedback2	Doble	Variable de apoyo	No	No
feedback3	Doble	Variable de apoyo	No	No
feedback4	Doble	Variable de apoyo	No	No
ff_ini	Doble	Variable de apoyo	No	No
ff_ini_g	Doble	Variable de apoyo	No	No
FT103	Doble	Flujo másico (kg/s)	Sí	Sí
FT103_SP	Doble	Flujo másico (kg/s)	No	Sí
FT104	Doble	Flujo másico (kg/s)	No	Sí
FT104_SP	Doble	Flujo másico (kg/s)	No	Sí
FV-01-ZL	Doble	Porcentaje válvula (%)	No	Sí
Initial temperature	Doble	Temperatura (°C)	No	Sí
Initial level	Doble	Nivel (m)	No	Sí
Initial pressure	Doble	Presión (Pa)	No	Sí
Inlet temperatura gas	Doble	Temperatura (°C)	No	Sí
Inlet temperatura MS	Doble	Temperatura (°C)	No	Sí
LT129	Doble	Nivel (m)	Si	Sí
Mass flow rate gas	Doble	Flujo másico (kg/s)	No	Sí
Mass flow rate MS	Doble	Flujo másico (kg/s)	No	Sí
MF_MS_fijo	Doble	Variable de apoyo	No	No
On-Off	Booleano	Verdadero/Falso	No	Sí
P-01-SC	Doble	Frecuencia (Hz)	No	Sí
Real time	Doble	Tiempo (s)	No	Sí
salida_modelo	Doble	Variable de apoyo	No	No
sControl2	Doble	Variable de apoyo	No	No
sControl3	Doble	Variable de apoyo	No	No
sControl4	Doble	Variable de apoyo	No	No
Sinchronization time interval	Doble	Tiempo (s)	No	Sí
sp_vgt	Doble	Variable de apoyo	No	No
tamb	Doble	Temperatura (°C)	No	No
Tank_mflowout	Doble	Flujo másico (kg/s)	No	Sí
Time	Doble	Tiempo (s)	No	Sí
Tout_Tank	Doble	Temperatura (°C)	No	Sí
TT106	Doble	Temperatura (°C)	Sí	Sí

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

TT106_SP	Doble	Temperatura (°C)	No	Sí
TT112	Doble	Temperatura (°C)	No	Sí
TT113	Doble	Temperatura (°C)	No	Sí
TT114	Doble	Temperatura (°C)	No	Sí
TT140	Doble	Temperatura (°C)	No	Sí
TY103	Doble	Temperatura (°C)	No	Sí
TY134	Doble	Temperatura (°C)	No	Sí
valor	Doble	Variable de apoyo	No	No
valor2	Doble	Variable de apoyo	No	No
ZI-V01	Doble	Porcentaje válvula (%)	No	Sí

Tabla 4. Variables compartidas utilizadas en el proyecto

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

4.3. Inicialización del sistema

Una vez creadas las variables compartidas, un aspecto importante del sistema es su inicialización.

La herramienta software trata de representar el sistema de la forma más realista posible, por tanto, la puesta en marcha debe realizarse con los parámetros establecidos para que funcione en torno a su punto de operación. Estos parámetros tienen relación con las condiciones ambientales a las que está sometida la planta, y también a las restricciones del fabricante para garantizar la seguridad del HRS, así como los valores que debe adquirir al inicio de su funcionamiento. A su vez, se introducirán varios parámetros relacionados con la adaptación a tiempo real, tiempo de sincronizado y filtrado necesarios para que la correcta ejecución del modelo.

Una vez que estos parámetros han sido introducidos al modelo, ya se podrá poner en marcha el modelo que simula el comportamiento del demostrador de la planta.

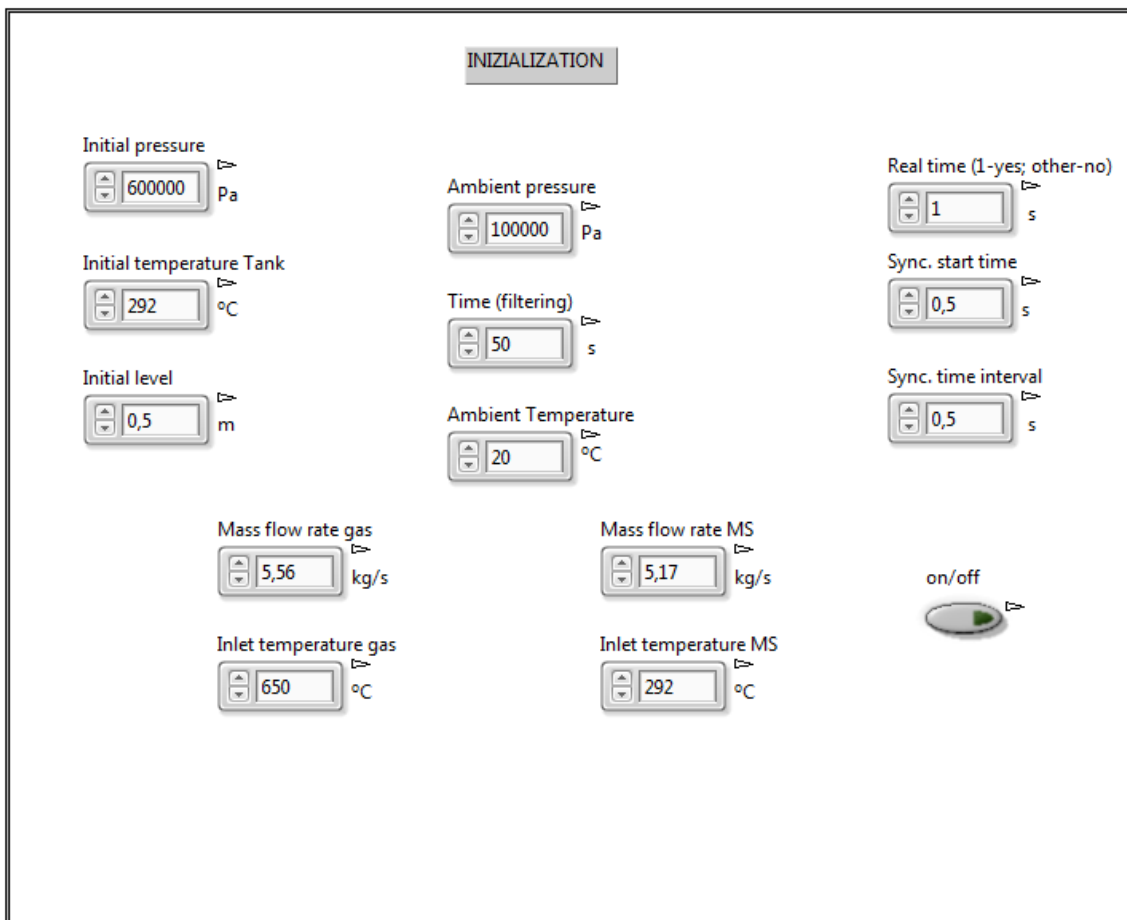


Figura 34. Pestaña de inicialización del sistema en el SCADA

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

Los parámetros que necesitan ser inicializados y que, por lo tanto, aparecen en la ventana de inicialización están descritos en la siguiente tabla. Los valores predeterminados son los que a priori se deben establecer, pero son susceptibles a cambios.

Nombre parámetro	Descripción	Valor	Unidad
Initial Pressure	Presión inicial en el banco de tubos de sales fundidas	600000	Pascales (Pa)
Initial Temperature Tank	Temperatura inicial a la que se desea el tanque	292	Grados Celsius (°C)
Initial Level	Nivel inicial del tanque	0.5	Metros (m)
Ambient Pressure	Presión ambiental atmosférica	100000	Pascales (Pa)
Ambient Temperature	Temperatura ambiente del sistema	20	Grados Celsius (°C)
Mass Flow Rate Gas	Flujo másico de gas que se desea a la hora de iniciar el sistema	5.56	Kilogramos por segundo (kg/s)
Inlet Temperature Gas	Temperatura de gas deseada a la entrada del HRS inicialmente	650	Grados Celsius (°C)
Mass Flow Rate Molten Salt	Flujo másico de sales fundidas que se desea a la hora de iniciar el sistema	5.17	Kilogramos por segundo (kg/s)
Inlet Temperature Molten Salt	Temperatura de sales fundidas deseada a la entrada del HRS inicialmente	292	Grados Celsius (°C)
Real Time	Si está en tiempo real o no (1/0)	1	-
Synchronization Start Time	Tiempo de inicio de la sincronización	0.5	Segundos (s)
Synchronization Time Interval	Intervalo de tiempo de sincronización	0.5	Segundos (s)
Time	Tiempo, en referencia al filtrado	50	Segundos (s)
On/Off	Apagado o encendido del simulador	Off	-

Tabla 5. Parámetros de inicialización del sistema

Estos valores son los que se han establecido de manera predeterminada (por defecto) para el correcto funcionamiento de la herramienta. Por lo tanto, son los valores que toma el sistema a la hora de arrancarse la herramienta.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

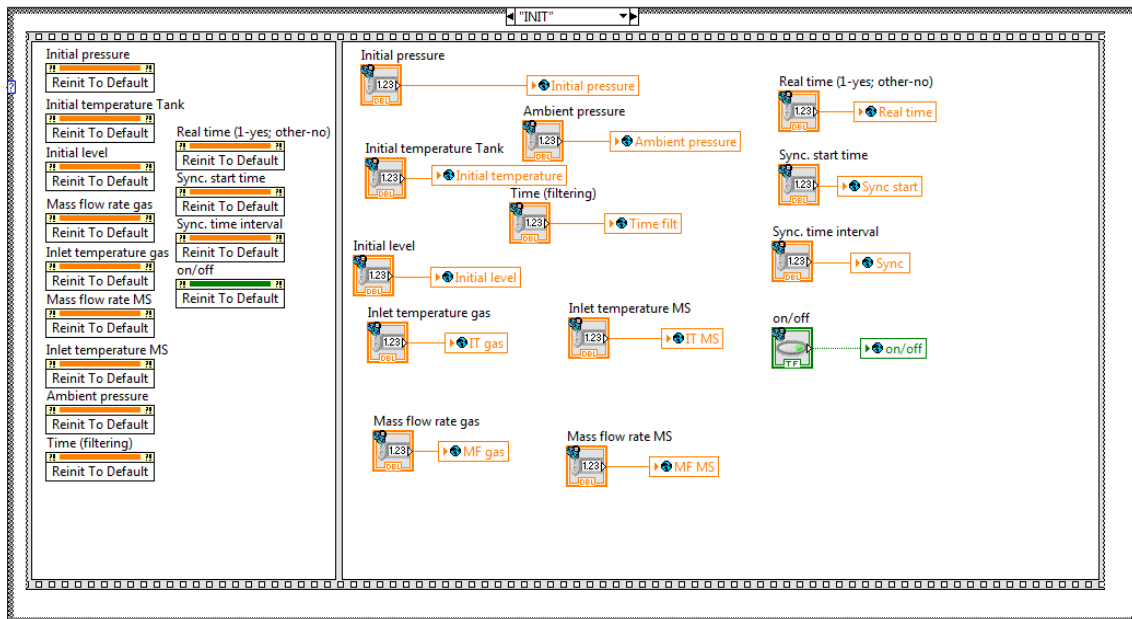


Figura 35. Ventana de inicialización en el diagrama de bloques

La inicialización se realiza de la siguiente manera. En primer lugar, en el segundo *frame* de la estructura principal, cuando el bucle *For* anteriormente explicado está realizando el recorrido por todas las pestañas del sistema, en el caso de inicialización se ejecuta una secuencia en la cual primero se constituyen los valores por defecto programados antes y a continuación se pasa la información a unas variables globales creadas para ayudar a las variables compartidas.

Las variables globales tienen la misma función que las variables compartidas, esto es, compartir información entre varios VIs; pero no tienen la capacidad para registrar y supervisar datos. Las variables globales se almacenan en un VI que se compone únicamente de panel frontal, y se crean de forma homóloga a las variables compartidas creadas. El nombre del VI viene automáticamente asignado como "Global1".



Figura 36. Icono de variables globales: a) modo lectura; b) modo escritura

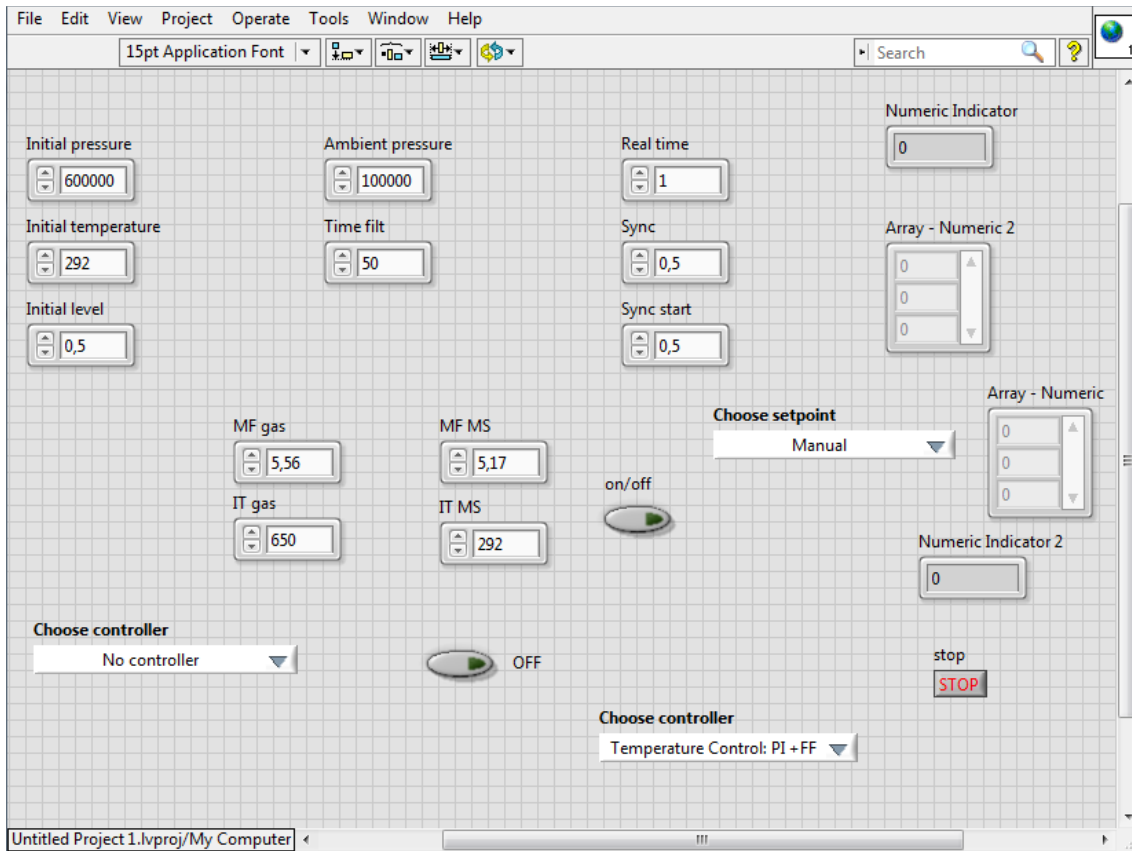


Figura 37. VI de variables globales

Una vez acabada la ejecución de este *frame*, se procede a ejecutarse el último *frame* de la estructura principal. En este subdiagrama se incluye una estructura *Flat Sequence*, que tiene la misma funcionalidad que la *Stacked Sequence* pero apareciendo todos los *frames* a la vez y puestos en orden de ejecución, de izquierda a derecha. En el primer *frame* de esta nueva estructura se traspasan los valores de las variables globales a las variables compartidas para así ya poder tener el control completo sobre estas variables desde los controles en el panel frontal, que están directamente asociados a las variables compartidas.

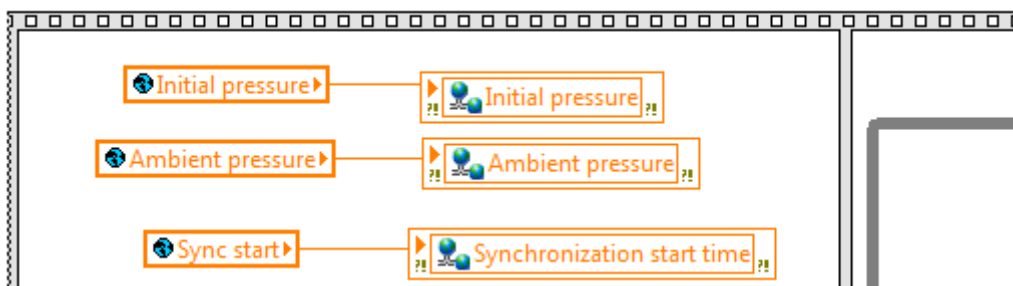


Figura 38. Traspaso de información de las variables globales a las variables compartidas

En el siguiente *frame*, donde se ejecuta el bucle principal de simulación, se transfieren los valores de inicialización finalmente al modelo, y a partir de ahí, ya se puede modificar

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

cuando se desee a través de los controles del panel frontal una vez se esté ejecutando el programa.

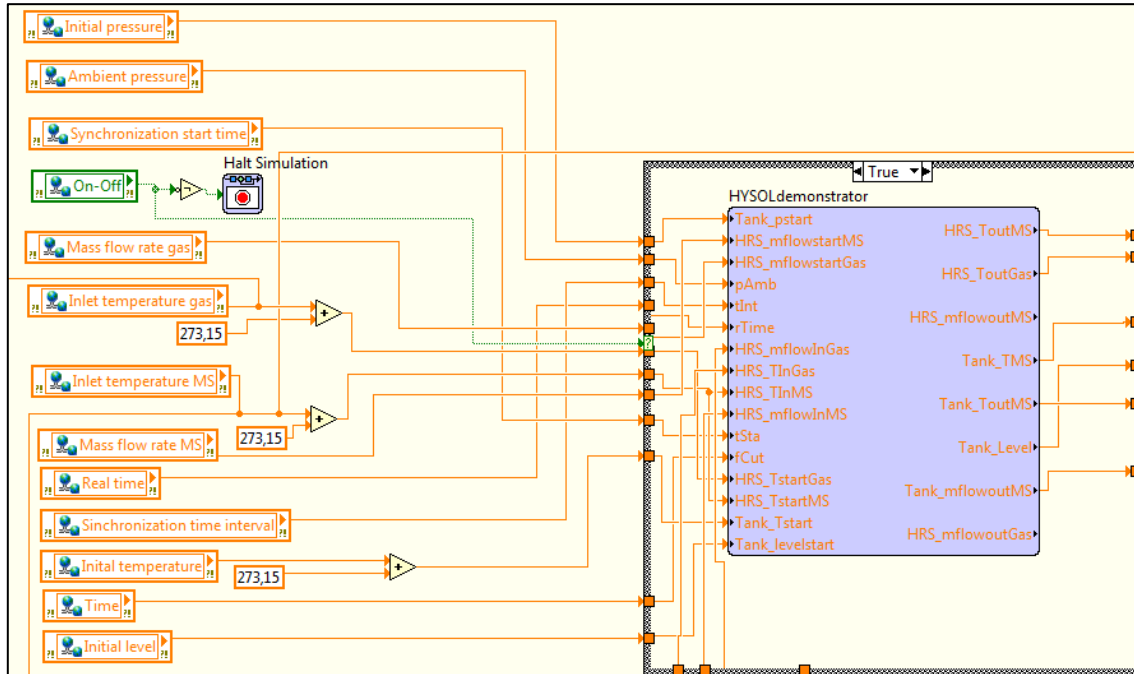


Figura 39. Introducción de los valores de inicialización al modelo como entradas

Hay que tener en cuenta que los valores introducidos en el panel frontal para las temperaturas son en grados Celsius ($^{\circ}\text{C}$), y el modelo trabaja en grados Kelvin ($^{\circ}\text{K}$); así que antes de introducirlos en el modelo, hay que transformarlos en grados Kelvin sumándole la constante 273.15.

4.4. Visualización de Gráficas

Este apartado es muy importante debido a que nos permite hacer un seguimiento de las variables características del sistema en todo momento, tanto en tiempo real, como en tiempo de simulación.

Como ya se ha explicado previamente, las gráficas en tiempo real se han realizado a parte del VI principal, en un VI llamado "graphs.vi". Y para incluirlo en el VI principal, se invoca al VI de las gráficas como se ha mostrado antes.

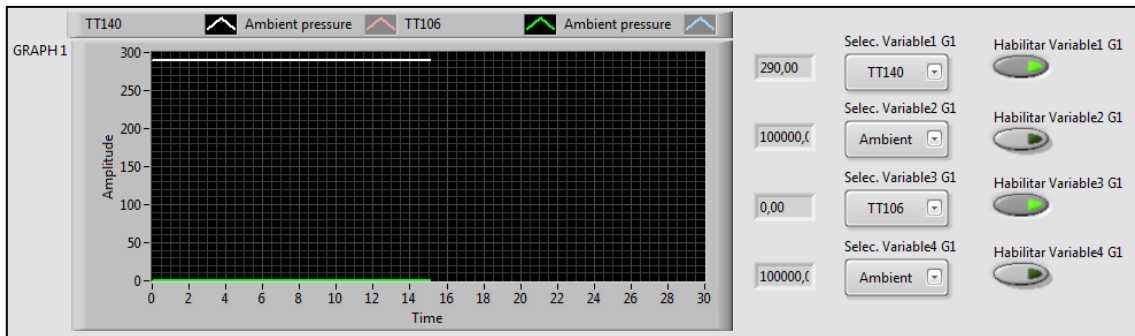


Figura 40. Visualización de una de las cuatro gráficas en tiempo real, habilitando dos de las cuatro variables disponibles

Se han incluido cuatro ventanas de gráficas para poder seguir en detalle todo lo que acontece en el sistema. Dentro de cada ventana se tiene la posibilidad de escoger hasta cuatro variables distintas para ser observadas, siempre y cuando se habilite su visualización. La lista de variables que se puede escoger es la misma que la de las variables compartidas, por tanto se puede escoger todas y cada una de las variables del proceso.

Este VI se compone de una estructura *Flat Sequence* en la cual hay tres subdiagramas. En el primero se inicializan las gráficas borrando cualquier traza de la anterior ejecución. En el siguiente se toman toda la lista de las variables compartidas para conseguir el nombre de cada una de las variables y poder mostrarlas en el selector de variables.

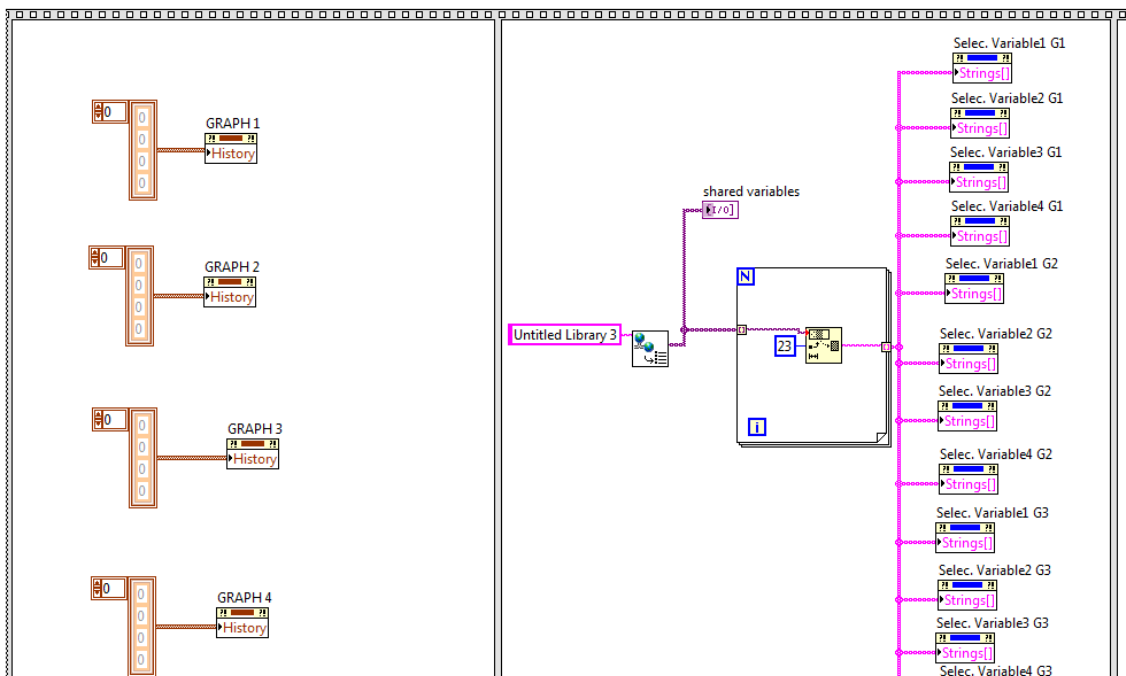


Figura 41. Diagrama de bloques del VI de la gráficas

Finalmente, la última orden de ejecución se compone de un bucle donde para cada una de las cuatro gráficas se leen los valores de las variables compartidas seleccionadas y se grafican en tiempo real, dependiendo si se ha habilitado o no su uso.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

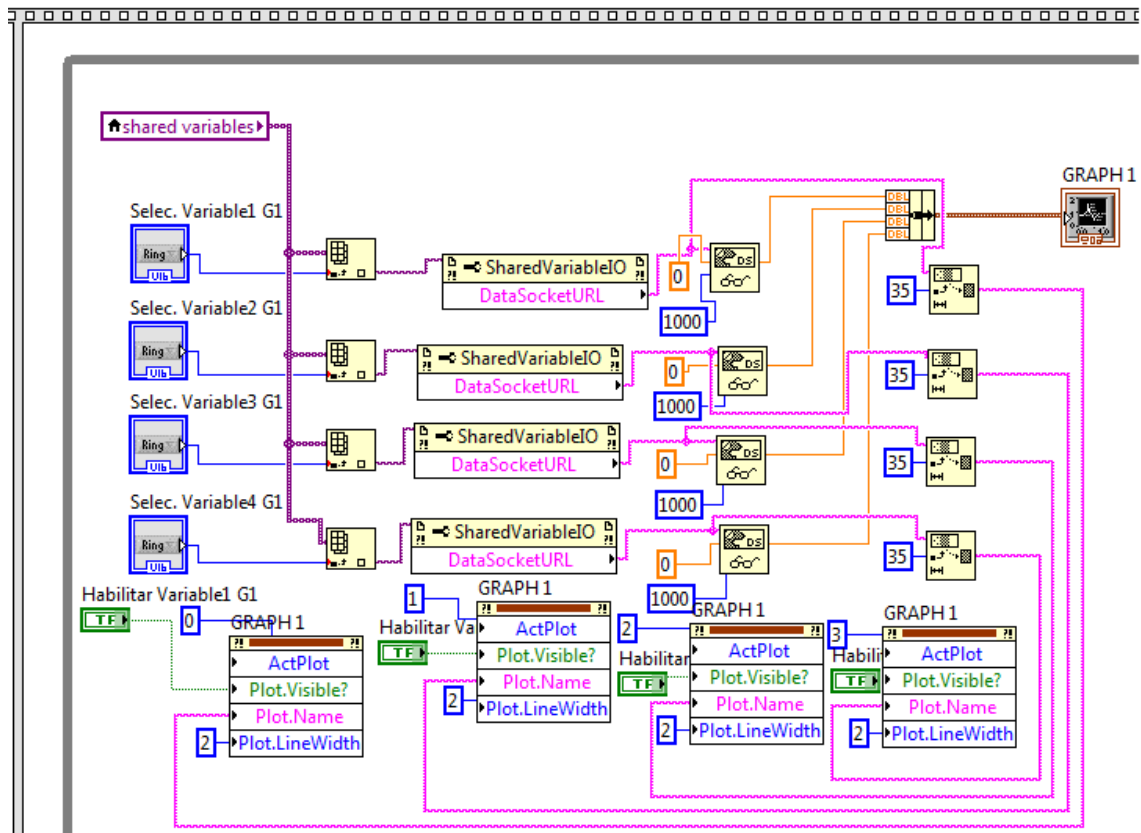


Figura 42. Bucle principal del VI de las gráficas

Por su parte, las gráficas en tiempo de simulación se han creado directamente en el VI principal “Simulador_0.3” ya que simplemente es tomar las salidas del modelo y mostrarlas en unas gráficas especiales de simulación que existen en el lazo de control y simulación.

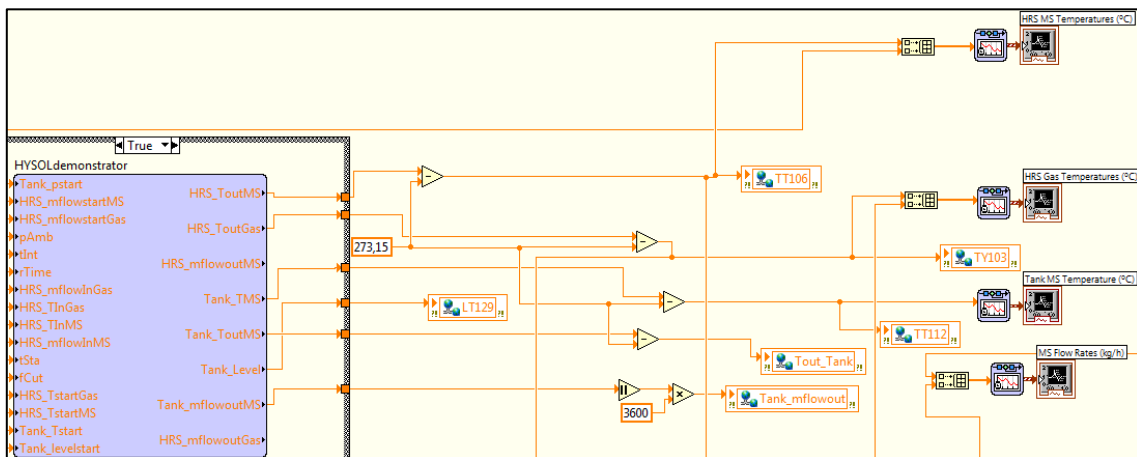


Figura 43. Salidas del modelo que se mostrarán en las gráficas en tiempo de simulación

Las variables escogidas para mostrar son las más significativas del sistema a controlar, que a su vez son las que devuelve el modelo, esto es: i) las temperaturas de las sales fundidas a la salida y a la entrada del HRS, ii) las temperaturas de los gases a la salida y a la entrada del HRS, iii) los flujos másicos tanto de gases como de sales fundidas que circulan por el sistema y, iv) la temperatura del tanque de sales fundidas.

Hay que volver a tener en cuenta que las temperaturas a la salida del modelo están en grados Kelvin y hay que convertirlas a grados Celsius antes de mostrarlas en pantalla.

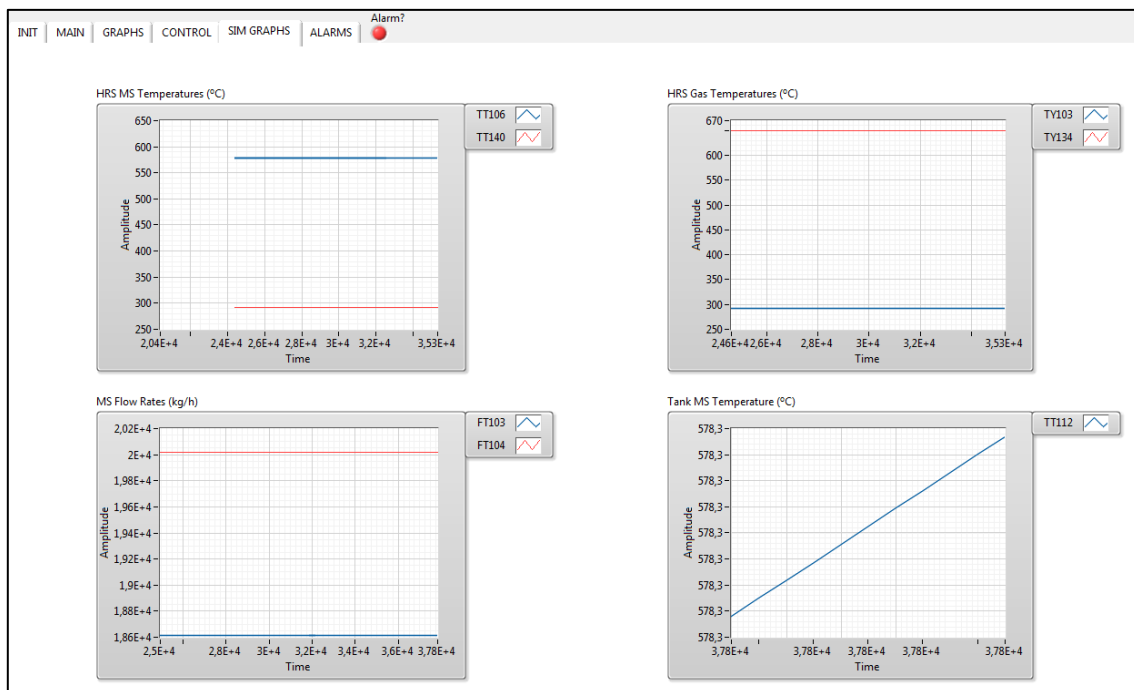


Figura 44. Gráficas en tiempo de simulación

4.5. Lazos de control

La herramienta debe representar de la forma más cercana posible el comportamiento del demostrador de la tecnología HYSOL. Para ello, se han incluido las diferentes estrategias de control aplicadas en el demostrador:

- Control de la temperatura de sales fundidas a la salida del HRS teniendo el flujo másico de sales fundidas como variable de control (**FIG-103**): dentro de este lazo de control se usan dos controladores PI conectados en modo cascada; el PI esclavo actúa sobre la válvula de apertura para controlar el flujo másico de las sales fundidas y el PI maestro controla la temperatura de las sales fundidas usando la consigna del flujo másico como variable de control. Este control maestro está combinado con la acción de un controlador por adelantado (*feedforward*) en configuración paralela.
- Control de la temperatura de sales fundidas a la salida del HRS teniendo el flujo másico de gases como variable de control (**FIG-104**): este lazo de control consiste en

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

dos controladores PID conectados en modo cascada; el PID esclavo actúa sobre la válvula de apertura del calentador de aire para controlar el flujo másico de los gases a la entrada del HRS y el PI maestro controla la temperatura de las sales fundidas usando la consigna del flujo másico como variable de control. Este control maestro está combinado con la acción de un controlador por adelantado (*feedforward*) en configuración paralela. Es un esquema de control muy similar al FIC-103.

- Lazo de control de la turbina de gas: este esquema de control se compone de dos lazos de control con controladores PI. El primer lazo de control depende de si está activado el postcombustor; si es así, se activa el lazo de control virtual y se establece una consigna de temperatura deseada a la salida del postcombustor a partir de los datos de la curva de temperatura de la turbina. Esa salida es la referencia del segundo lazo de control. Por otro lado, si no se activa el postcombustor, los valores de la curva de temperatura de la turbina pasan directamente como consigna al segundo lazo de control, que es el del calentador de aire, para obtener la temperatura del gas deseada a la salida del calentador, que es a su vez la temperatura de entrada del gas al HRS.

4.5.1. Lazo de control FIC-103

El propósito de este lazo de control es el de controlar la temperatura de las sales fundidas a la salida del HRS ($T_{ms,out}$) teniendo como variable de control el flujo másico de sales fundidas.

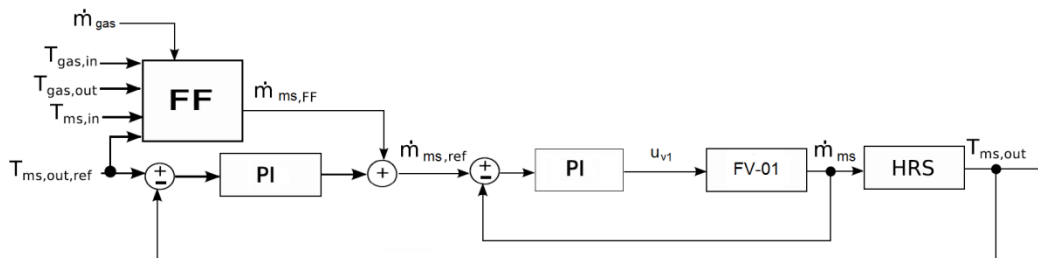


Figura 45. Esquema de control FIC-103

Este esquema de control se puede dividir en tres partes: el PI maestro, el PI esclavo y el *feedforward*. El PI maestro recibe una consigna de temperatura de las sales fundidas a la salida del HRS o referencia deseada introducida por el usuario ($T_{ms,out,ref}$). Este PI actúa mandando una referencia de flujo másico de sales fundidas ($\dot{m}_{ms,ref}$), dentro de los límites posibles de la planta, al segundo lazo de control, que está puesto en modo cascada. Este lazo secundario se compone de un PI al cual le entra la referencia ya citada y actúa enviando una señal de control U_{v1} para mover la válvula FV-01 que regula el paso de las sales fundidas. La señal de control se envía en forma de porcentaje, ya que la válvula se abre en un rango de 0 a 100 %. Una vez ya se ha abierto o cerrado la válvula hasta el punto de permitir que fluya la cantidad de flujo másico requerida, las sales fundidas entran en el HRS para comenzar su intercambio de calor con los gases. A la salida del HRS la temperatura de las sales fundidas se irá incrementando o decrementando hasta conseguir la

temperatura deseada. En este punto se cierra el lazo principal y se aplica la realimentación con el valor de la temperatura de las sales fundidas a la salida del HRS.

Es necesario saber que la temperatura de las sales fundidas a la entrada del HRS ($T_{ms,in}$) viene predeterminada en la inicialización del sistema, ya que a la hora de abordar este trabajo de fin de grado, no se disponían de datos experimentales para el modelado del *Air Cooler* y por lo tanto se ha asumido un modelo de *Air Cooler* ideal.

A su vez, y paralelamente al lazo de control, se diseña un controlador por adelanto (*feedforward*). Esta estrategia de control permite anticiparse y rechazar las perturbaciones medibles que proceden de otros lazos de control, como puede ser el FIC-104. Las perturbaciones se producen debido a la temperatura del gas que entra al HRS ($T_{gas,in}$), a la temperatura del gas a la salida del HRS ($T_{gas,out}$) y al flujo másico de gases que fluctúan por el sistema (\dot{m}_{gas}).

La salida del controlador *feedforward* se obtiene gracias al balance de energía en estado estacionario en el HRS despreciando las pérdidas de calor:

$$c_{p,ms} \cdot (T_{ms,out} - T_{ms,in}) \cdot \dot{m}_{ms} = c_{p,gas} \cdot (T_{gas,in} - T_{gas,out}) \cdot \dot{m}_{gas} \quad (1)$$

$$\dot{m}_{ms} = \dot{m}_{gas} \frac{c_{p,gas} \cdot (T_{gas,in} - T_{gas,out})}{c_{p,ms} \cdot (T_{ms,out} - T_{ms,in})} \quad (2)$$

Siendo $c_{p,gas}$ un valor medio constante de 1224 (J/(kg°K)) y teniendo $c_{p,ms}$ un valor que se representa mediante la siguiente expresión:

$$c_{p,ms} = 1663.43 + 0.446 \frac{(T_{ms,out} + T_{ms,in})}{2} \quad (J/(kg^{\circ}K)) \quad (3)$$

Y así se obtiene la cantidad necesaria de flujo másico de sales fundidas que hay que aportar al lazo de control para compensar el efecto de las perturbaciones (2).

Los PI implementados en este lazo de control tienen incluidos un esquema *anti-windup* y de transferencia sin saltos. Los parámetros de los controladores inicialmente se ajustaron usando el método AMIGO, y finalmente se ajustaron de forma manual. Estos parámetros son los siguientes:

❖ PI maestro:

Constante proporcional: $K_p = -3$

Tiempo Integral: $T_i = 100$ s

❖ PI esclavo:

Constante proporcional: $K_p = 0.1$

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

Tiempo Integral: $T_i = 0.25 \text{ s}$

Dentro de este lazo de control existen diferentes modos de operación, los cuales el usuario puede hacer uso de ellos dependiendo de lo que quiera experimentar. Cuatro modos son los creados: i) control manual, ii) control del flujo másico de sales fundidas, iii) control PI de la temperatura de sales fundidas y, iv) control PI más *feedforward* de la temperatura de sales fundidas.

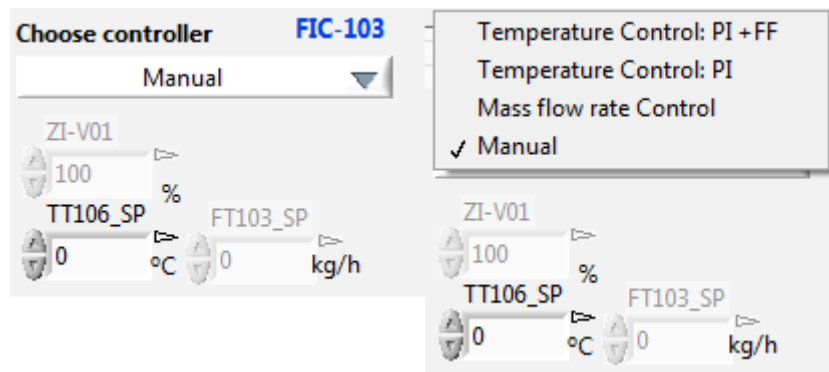


Figura 46. Ventana de elección de diferentes modos de operación y controles de consignas para el lazo FIC-103

El control manual consiste en mover manualmente la válvula FV-01 gracias al control ZI-V01 en un rango de 0 a 100 %; mientras tanto, el resto de controles automáticos permanecen desactivados. Cuando se inicia la herramienta, los lazos de control aparecen desactivados y el control manual es el que está activado.

El control del flujo másico de las sales fundidas se basa en el funcionamiento única y exclusivamente del lazo de control secundario donde actúa el PI esclavo, teniendo el PI maestro desactivado. Así, se tiene un simple bucle en el cual se controla la consigna del flujo másico de las sales fundidas que se desea que entren al HRS utilizando el control FT103_SP.

El control de la temperatura usando el PI, consiste en utilizar el esquema de control (lazo primario y secundario) a excepción de la acción del *feedforward*. Para establecer la consigna de temperatura se usa el control TT106_SP.

El control de la temperatura usando el PI más el *feedforward* utiliza el esquema de control al completo, incluyendo ahora sí la acción del *feedforward*. Al igual que en el anterior modo, la consigna de temperatura se establece con el control TT106_SP.

Cuando se pasa de un modo a otro de operación, los valores de temperatura, flujo másico y porcentaje de válvula se mantienen exactamente igual a los que se tenían en el modo anterior, es decir, no vuelven a 0 o cualquier otro valor de inicio. Esto se consigue gracias a la transferencia sin saltos y al traspaso de información de unas variables a otras para que estén informadas constantemente. Este proceso se explicará posteriormente.

Un aspecto importante que concierne a este lazo de control es tener en cuenta la bomba que impulsa las sales fundidas desde el tanque hasta la válvula FV-01. Esta bomba se representa mediante la siguiente ecuación, modelo P01, en el dominio de Laplace en forma de función de transferencia e incluido en el bucle principal de la herramienta, el de control y simulación:

$$H(s) = \frac{7.6125}{s^2 + 5.8824s + 8.6505} \quad (4)$$

La entrada de este modelo es la frecuencia a la que debe funcionar la bomba y se introduce mediante el control P-01-SC. El valor inicial de frecuencia se ha establecido en 20 Hz en un rango de 0 a 50 Hz. La salida que nos proporciona este modelo es el flujo volumétrico (m^3/h) el cual hay que transformar en flujo másico (kg/h) para que pueda circular por el sistema multiplicándolo por la densidad (kg/m^3). La densidad es calculada mediante la siguiente fórmula:

$$d = 2080 - 0.733 \cdot TT140 \quad (kg/m^3) \quad (5)$$

Siendo TT140 la temperatura de las sales fundidas a la entrada del HRS.

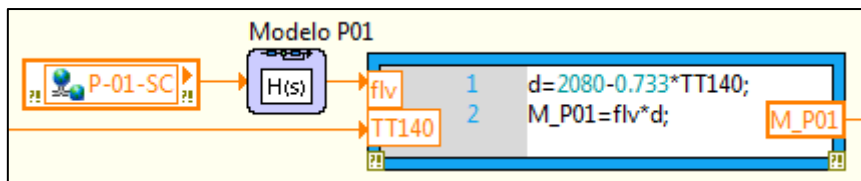


Figura 47. Esquema de la bomba P01

El esquema de control en el diagrama de bloques se presenta de la siguiente manera:

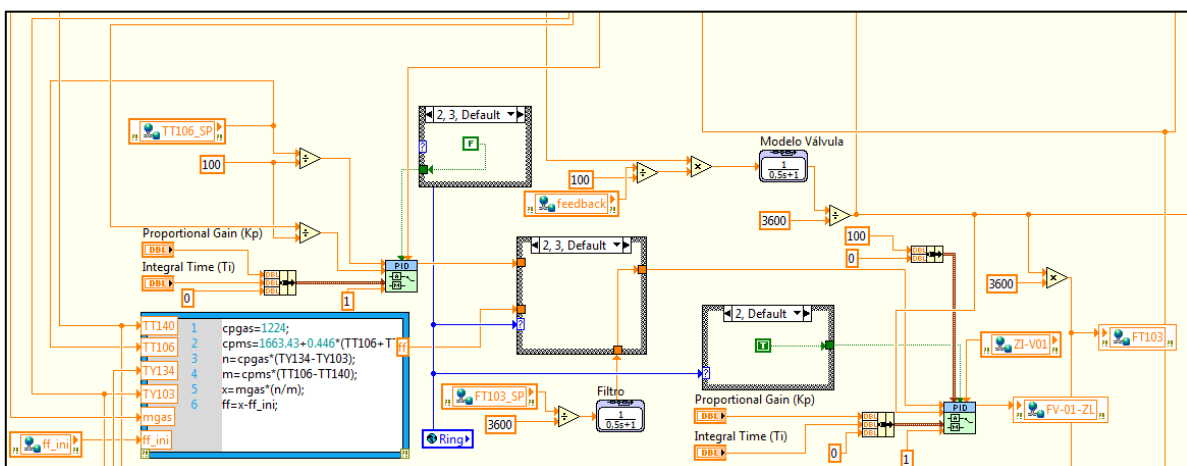


Figura 48. Esquema general del lazo FIC-103 en el diagrama de bloques

Como se puede observar, para el PI se ha usado el bloque de LabVIEW PID Advanced, que proporciona una transferencia sin saltos y con *anti-windup*.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

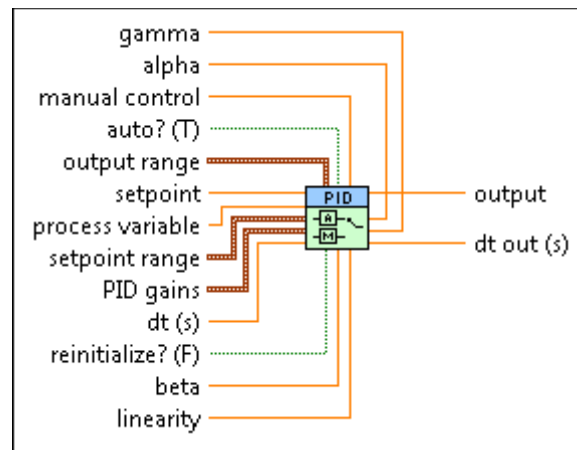


Figura 49. Bloque PID Advanced de LabVIEW

Existen tres importantes estructuras de decisión que se rigen dependiendo en qué posición esté el selector antes comentado de los modos de operación, llamado “Ring”. La posición 0 es el control de temperatura con PI más el *feedforward*, la posición 1 es el control de temperatura con PI, la posición 2 es el control de flujo másico y la posición 3 es el control manual.

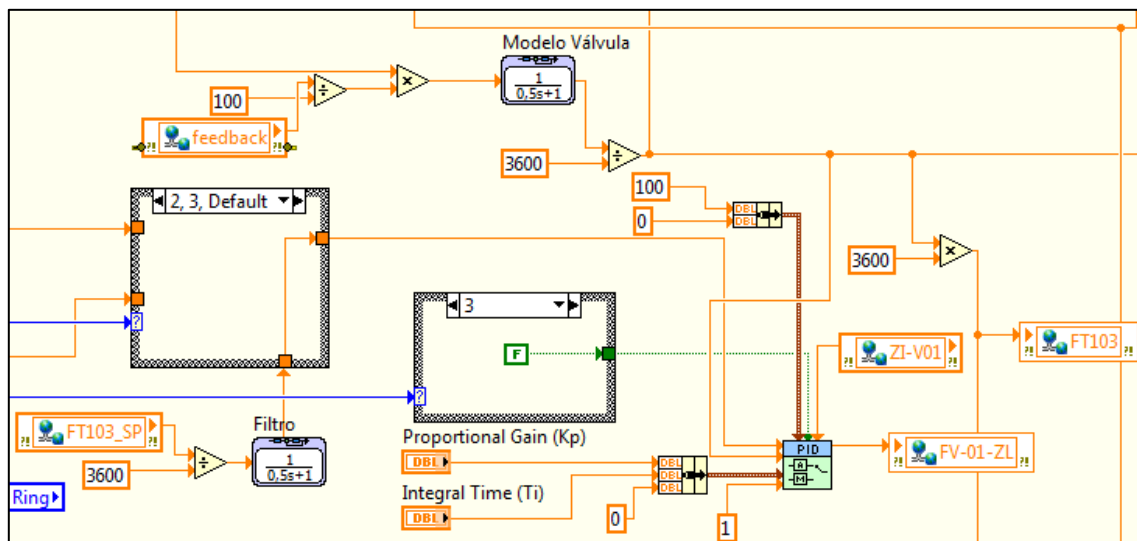


Figura 50. Lazo secundario del FIC-103 en el diagrama de bloques

En el lazo secundario, si estamos en la posición 0, 1 ó 2, se desactiva el control manual y en la posición 3 se activa.

Cuando está funcionando este lazo de control, el PI manda señales de control a la válvula a través de la variable de apoyo “feedback”. Esta válvula se ha representado con el siguiente modelo en forma de función de transferencia:

$$G(s) = \frac{\left(\frac{FV_01_ZL}{100}\right) (M_P01)}{0.5s + 1} \quad (6)$$

Donde M_P01 es el flujo másico proveniente de la bomba y $FV-01-ZL$ es el valor en porcentaje de la bomba. Este modelo devuelve el valor del flujo másico circulante en kg/h y hay que transformarlo en kg/s para que trabaje en el modelo del HRS.

El control manual es el que primero se activa cuando arranca la herramienta. Por lo tanto, debe estar inicializado a un valor predeterminado que el sistema necesite para funcionar correctamente desde su inicio. Este valor ha sido calculado en función de los Hz a los que funcione la bomba, de la temperatura de las sales fundidas a la entrada del HRS (292 °C) y del flujo másico inicial a la entrada del HRS (5.17 kg/s).

El valor inicial de flujo volumétrico que devuelve la bomba usando su modelo (4), haciendo uso de la transformada inversa de Laplace y operando en torno al punto inicial, se representa con la siguiente fórmula:

$$y(0) = u(0) \cdot \frac{7.6125}{8.6505} \text{ (kg/m}^3\text{)} \quad (7)$$

Cuando el valor inicial de la bomba está a 20 Hz, el flujo volumétrico M_P01 es de 17.6 kg/m³, que pasándolo a kg/h haciendo uso de la densidad del fluido (5) es de 32841.22 kg/h, que a su vez es 9.12 kg/s. Por tanto, haciendo el mismo procedimiento con el modelo de la válvula (6), se obtiene el valor inicial al cual debe estar la válvula y está expresado por la siguiente fórmula:

$$u(0) = \frac{y(0)}{M_P01} \cdot 100 \text{ (\%)} \quad (8)$$

Como el valor inicial de flujo másico que entra al HRS se ha establecido en 5.17 kg/s, el valor inicial al que debe estar la válvula es de 56.67 %. Este valor es el que deben adquirir las variables de control que incumben a la válvula y su señal de control.

Todo este procedimiento y fórmulas, al ser de inicialización, se sitúan en el mismo apartado que el resto de la inicialización; es decir, en el *frame* anterior al que incluye el bucle principal de simulación.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

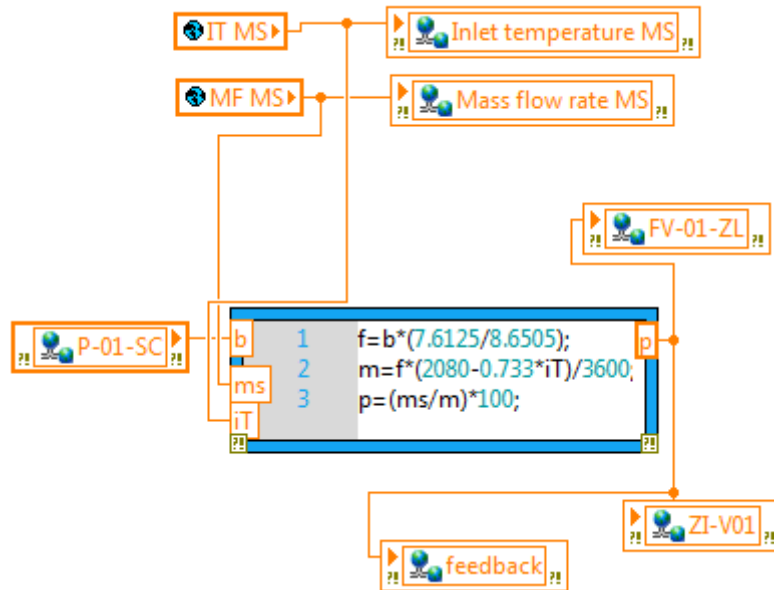


Figura 51. Inicialización de la válvula FV-01

Cuando está activo el control del flujo másico, el valor de la consigna se transforma en kg/s ya que es la unidad a la que trabaja el modelo del HRS; además, ese valor pasa por un filtro que se ha añadido para evitar darle cambios bruscos a la válvula. El modelo del filtro es el siguiente:

$$F(s) = \frac{1}{0.5s + 1} \quad (9)$$

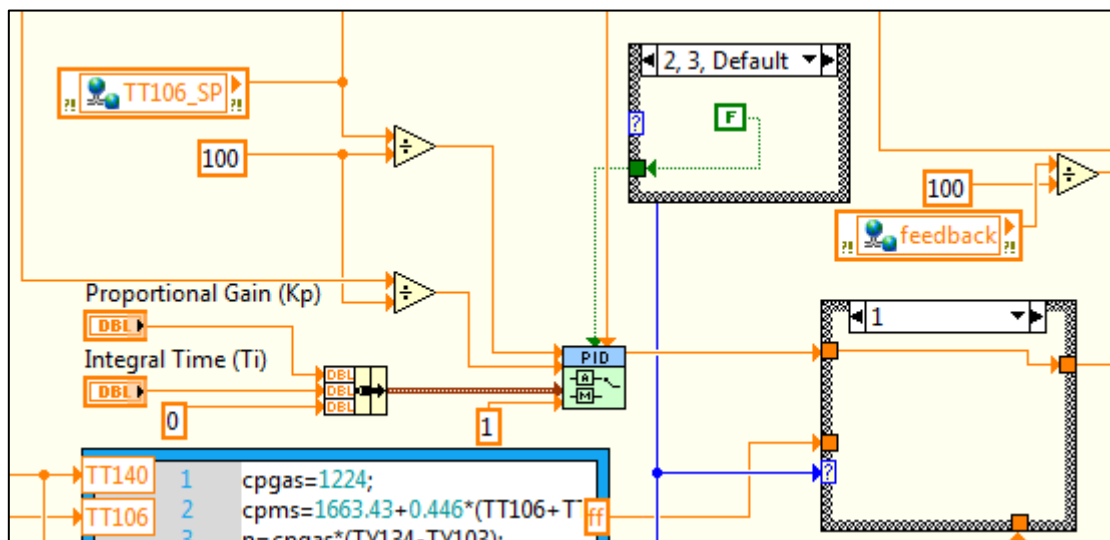


Figura 52. Lazo primario del FIC-103 en el diagrama de bloques

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

En el lazo primario, si el valor del selector es 0 ó 1 el PI está en modo automático y si el valor es 2 ó 3 está en modo manual y el valor que adquiere es el de flujo másico para que la señal de control de este lazo esté permanentemente informada de qué valor toma.

Entre el lazo primario y el secundario, está la estructura condicional principal, en la cual si el selector está en valor 2 ó 3 no se conectan entre sí porque el PI maestro está desactivado; sin embargo, si está en modo 0 ó 1 sí que se conectan entre sí los dos lazos.

Cuando está en modo 0, se activa la acción del *feedforward*, que se ha implementado en forma de matemática textual, que se adhiere a la señal de control del PI maestro para compensar el efecto de las perturbaciones.

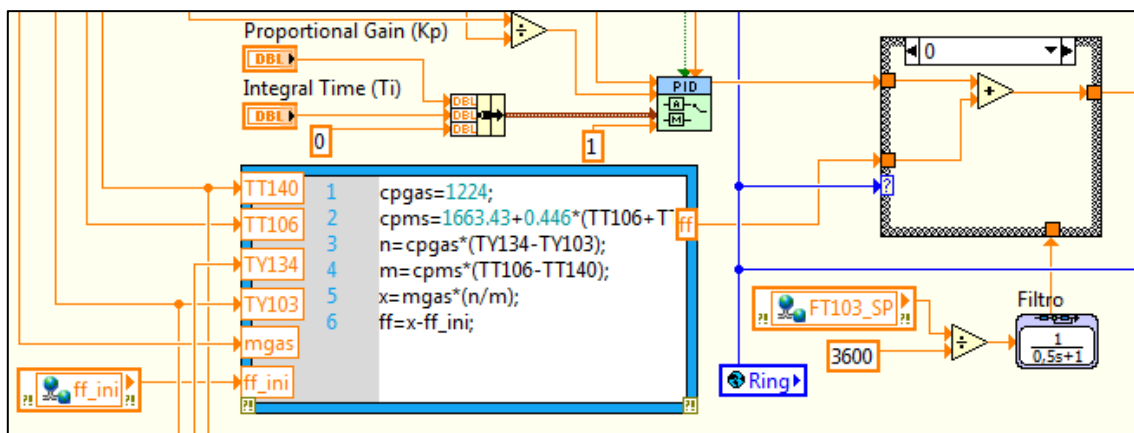


Figura 53. Acción *feedforward* en el lazo primario del FIC-103 en el diagrama de bloques

El *feedforward* necesita tener un valor inicial para que cuando se active y comience a enviar la señal de control, no tenga que dar cambios bruscos y solo proporcione la cantidad necesaria de flujo másico.

El valor inicial, introducido por la variable "ff_ini", se ha calculado de la misma forma que el *feedforward*, simplemente con los valores iniciales de las variables que afectan al *feedforward*. Este valor constante se obtiene cuando se activa la opción PI más *feedforward*; entonces, se ejecuta un evento que está emplazado en el VI "Simulador_0.2.Main" en el cual se calcula el valor inicial del *feedforward* y se pasa esta información al VI principal donde está el bucle de simulación y el *feedforward* principal.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

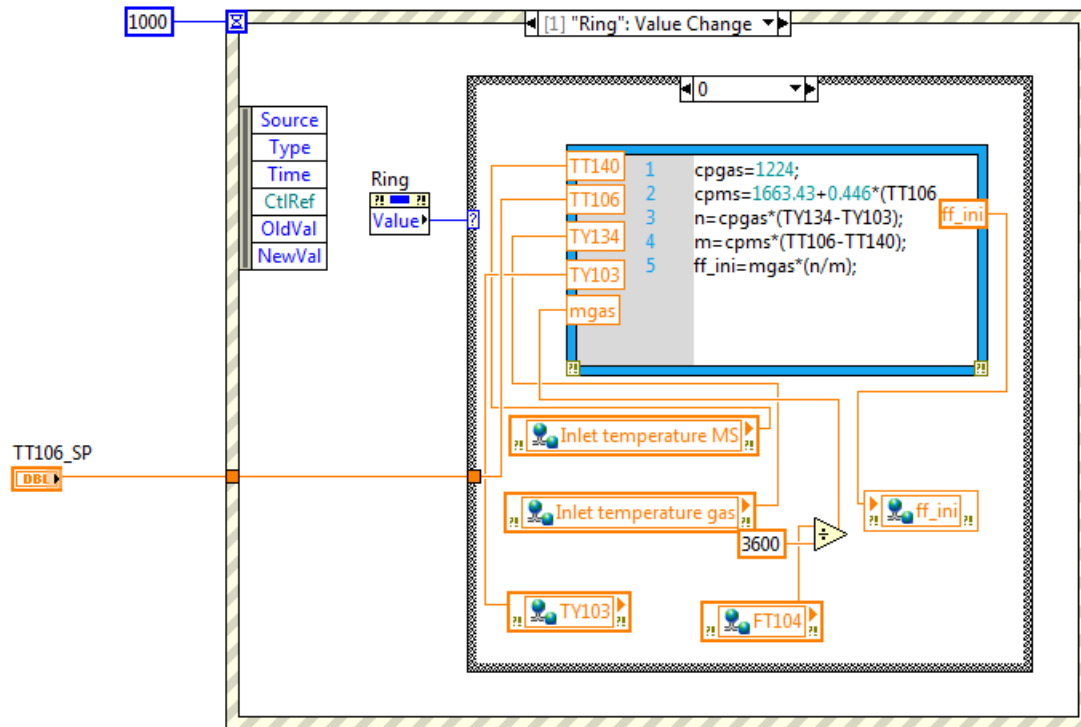


Figura 54. Evento que ejecuta el valor inicial del *feedforward* en FIC-103

Los eventos son estructuras que se ejecutan solo una vez cuando se cumple la condición que tienen asignada, los cuales suelen ser cambios de valor en controles o variables.

En este VI “Simulador_0.2.Main” se han almacenado las variables de apoyo. Estas variables se usan cuando se envían señales de control desde los PI a los modelos, y en vez de conectarlo directamente mediante un cable virtual lo cual produce problemas de inestabilidad, se ha optado por usar otra variable a parte que se conecta al modelo. Esta variable ha sido previamente informada en el “Simulador_0.2.Main” del valor de la señal de control.



Figura 55. Transmisión de la información a la variable de apoyo de la señal de control en el FIC-103

Así mismo, para que los valores de referencia y de control manual estén siempre informados de qué valor está tomando en cada momento, se usa la misma estrategia que con las variables de apoyo, y dependiendo en qué modo de operación esté el lazo de control, se actualizan unas variables u otras.



Figura 56. Variables del sistema informando al resto de variables para que estén actualizadas en el FIC-103

4.5.2. Lazo de control FIC-104

La finalidad de este lazo de control es controlar la temperatura de las sales fundidas a la salida del HRS ($T_{ms,out}$) teniendo como variable de control el flujo másico de gases.

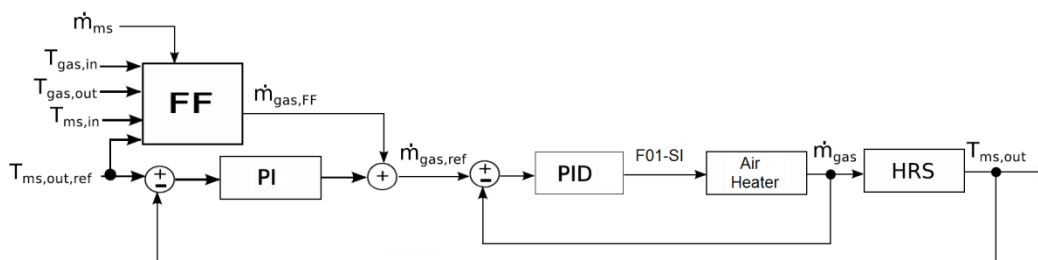


Figura 57. Esquema de control FIC-104

Esta estrategia de control es similar al FIC-103 y también se puede dividir en tres partes: el PI maestro, el PID esclavo y el *feedforward*. El PI maestro recibe una consigna de temperatura de las sales fundidas a la salida del HRS deseada que introduce el usuario ($T_{ms,out,ref}$). Este PI actúa mandando una referencia de flujo másico de gases ($\dot{m}_{gas,ref}$), dentro de los límites posibles que se pueden conseguir en la planta, al lazo de control secundario, que está puesto en modo cascada. Este segundo lazo se compone de un PID el cual recibe la referencia ya citada y actúa enviando una señal de control F01-SI para mover la válvula del *Air Heater* o calentador de aire que regula el paso de los gases. La señal de control se envía en forma de porcentaje, ya que la válvula se abre en un rango de 0 a 100 %. Cuando ya se ha abierto o cerrado la válvula hasta el punto de permitir que fluya la cantidad de flujo másico requerida, los gases entran en el HRS para comenzar su intercambio de calor con las sales fundidas. A la salida del HRS la temperatura de las sales fundidas se irá incrementando o decrementando hasta conseguir la temperatura deseada.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

En este punto se cierra el lazo principal y se aplica la realimentación con el valor de la temperatura de las sales fundidas a la salida del HRS.

Al mismo tiempo, y paralelamente al lazo de control, se diseña un controlador *feedforward*. Esta estrategia de control permite anticiparse y rechazar las perturbaciones medibles que proceden de otros lazos de control, como puede ser el FIC-103. Las perturbaciones se producen debido a la consigna de la temperatura de las sales fundidas que queremos a la salida del HRS ($T_{ms,out,ref}$), a la temperatura de las sales fundidas a la entrada del HRS ($T_{ms,in}$) y al flujo másico de las sales fundidas que fluctúan por el sistema (\dot{m}_{ms}).

La salida del controlador *feedforward* se obtiene gracias al balance de energía en estado estacionario en el HRS despreciando las pérdidas de calor:

$$c_{p,ms} \cdot (T_{ms,out} - T_{ms,in}) \cdot \dot{m}_{ms} = c_{p,gas} \cdot (T_{gas,in} - T_{gas,out}) \cdot \dot{m}_{gas} \quad (10)$$

$$\dot{m}_{gas} = \dot{m}_{ms} \frac{c_{p,ms} \cdot (T_{ms,out} - T_{ms,in})}{c_{p,gas} \cdot (T_{gas,in} - T_{gas,out})} \quad (11)$$

Teniendo $c_{p,gas}$ un valor medio constante de 1224 (J/(kg°K)) y siendo $c_{p,ms}$ un valor que se calcula mediante la siguiente expresión:

$$c_{p,ms} = 1663.43 + 0.446 \frac{(T_{ms,out} + T_{ms,in})}{2} \quad (J/(kg^{\circ}K)) \quad (3)$$

Por lo tanto, se obtiene la cantidad indispensable de flujo másico de gases que hay que aportar al lazo de control para compensar el efecto de las perturbaciones (11).

Los PI implementados en este lazo de control tienen incluidos un esquema *anti-windup* y la transferencia sin saltos. Los parámetros de los controladores inicialmente se ajustaron usando el método AMIGO, y finalmente se ajustaron de forma manual. Estos parámetros son los siguientes:

❖ PI maestro:

Constante proporcional: $K_p = 0.05$

Tiempo Integral: $T_i = 100$ s

❖ PI esclavo:

Constante proporcional: $K_p = 0.003$

Tiempo Integral: $T_i = 40$ s

Tiempo Derivativo: $T_d = 0$ s

Dentro de este lazo de control existen diferentes modos de operación, los cuales el usuario puede hacer uso de ellos dependiendo de lo que quiera experimentar. Se han creado cuatro modos de operación: control manual, control del flujo másico de gases, control PI de la temperatura de sales fundidas, control PI más *feedforward* de la temperatura de sales fundidas.

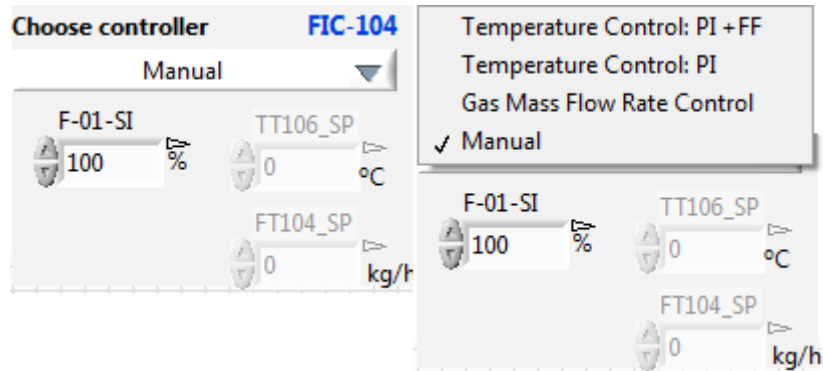


Figura 58. Ventana de elección de diferentes modos de operación y controles de consignas para el lazo FIC-104

El control manual consiste en mover manualmente la válvula que regula el paso de los gases en el calentador de aire gracias al control F01-SI en un rango de 0 a 100 %; mientras tanto, el resto de controles automáticos permanecen desactivados. Cuando se pone en marcha la herramienta, los lazos de control aparecen desactivados y el control manual es el que está activado.

El control del flujo másico de los gases se basa en que actúe solamente el lazo de control secundario donde trabaja el PID esclavo, teniendo el PI maestro desactivado. De esta forma solo funciona un bucle en el cual se controla el flujo másico de los gases que se desea que entren al HRS utilizando el control FT104_SP, para asignar la consigna.

El control de la temperatura usando el PI, consiste en utilizar el esquema de control (lazo primario y secundario) salvo la acción del *feedforward*. Para establecer la consigna de temperatura se usa el control TT106_SP.

El control de la temperatura usando el PI más el *feedforward* utiliza la estrategia de control al completo, incorporando ahora sí la acción del *feedforward*. Al igual que en el anterior modo, la consigna de temperatura se establece con el control TT106_SP.

Cuando se pasa de un modo a otro de operación, los valores de temperatura, flujo másico y porcentaje de válvula se mantienen exactamente igual a los que se tenían en el modo anterior, es decir, no tienen fluctuaciones. Esto se consigue gracias a la transferencia sin saltos y al traspaso de información de unas variables a otras para que estén informadas constantemente.

El esquema de control en el diagrama de bloques se presenta de la siguiente manera:

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

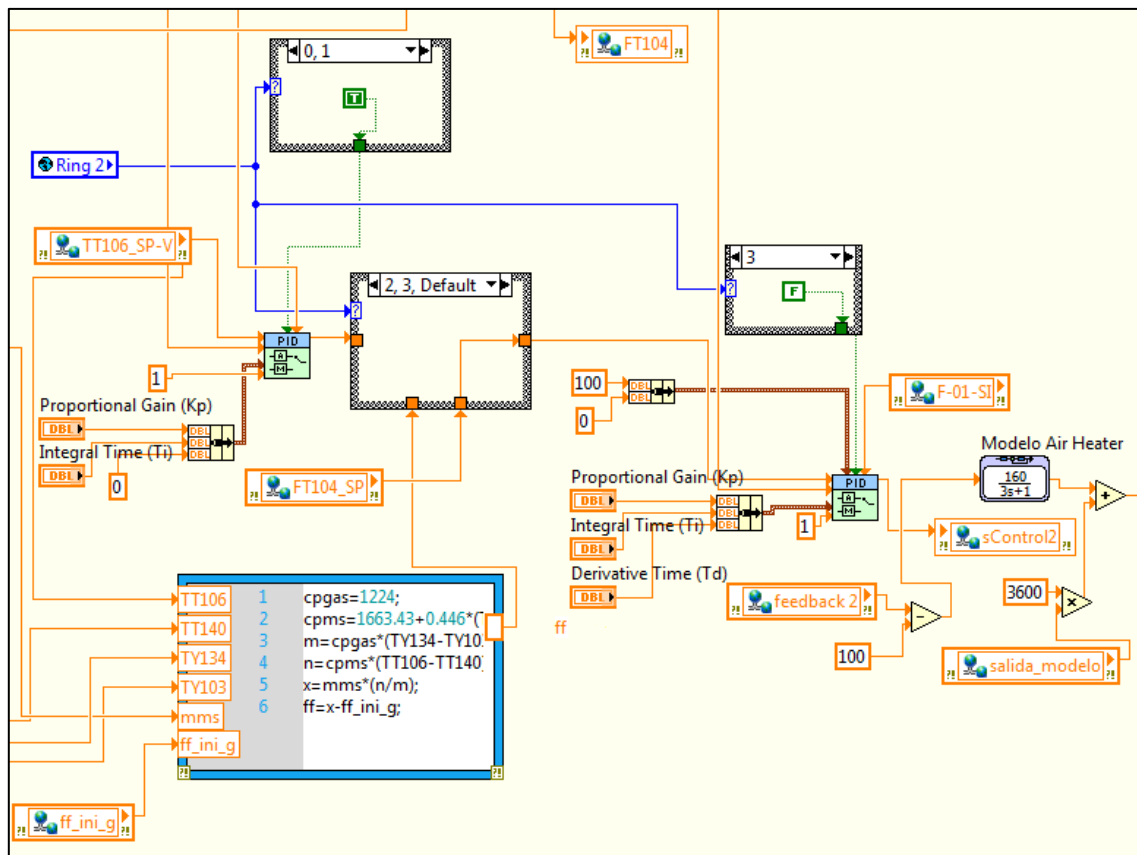


Figura 59. Esquema general del lazo FIC-104 en el diagrama de bloques

Al igual que en el lazo FIC-103, para el PID se ha usado el bloque de LabVIEW PID Advanced, que proporciona una transferencia sin saltos y con *anti-windup*.

También existen tres importantes estructuras de decisión que se rigen dependiendo en qué posición esté el selector antes descrito de los modos de operación, llamado "Ring 2". La posición 0 es el control de temperatura con PI más el *feedforward*, la posición 1 es el control de temperatura con PI, la posición 2 es el control de flujo másico de gases y la posición 3 es el control manual.

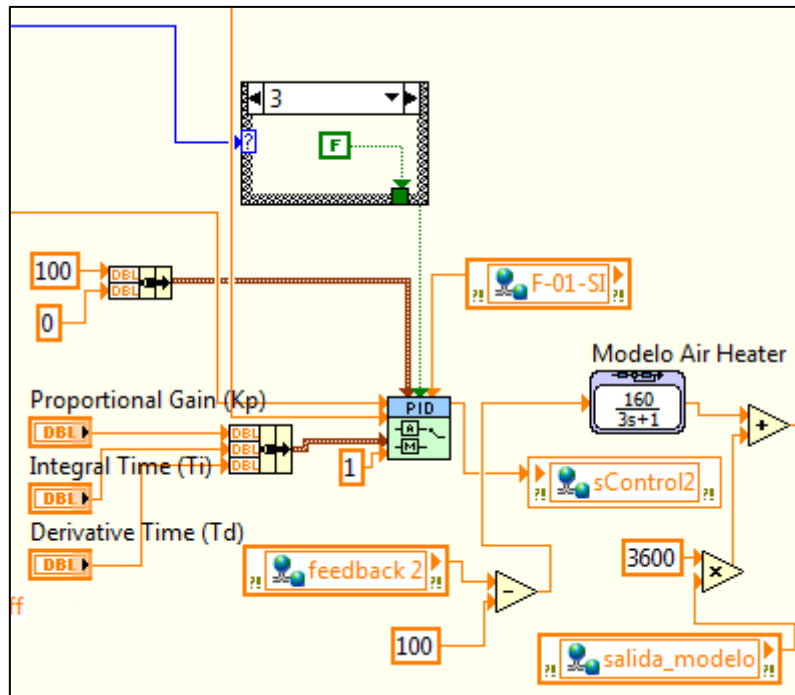


Figura 60. Lazo secundario del FIC-104 en el diagrama de bloques

En el lazo secundario, si estamos en la posición 0, 1 ó 2, se desactiva el control manual y en la posición 3 se activa.

Cuando se está ejecutando este lazo de control, el PID manda señales de control a la válvula del calentador de aire (*Air Heater*) a través de las variables de apoyo “sControl2” y “feedback 2”. Esta válvula del calentador se ha representado con el siguiente modelo en forma de función de transferencia:

$$G(s) = \frac{160}{3s + 1} \quad (12)$$

Este modelo devuelve el valor del flujo másico de gases circulante en kg/h y hay que transformarlo en kg/s para que trabaje en el modelo del HRS.

Como pasa en el lazo FIC-103, el control manual es el primero que se activa cuando arranca la herramienta. Por lo tanto, debe estar inicializado a un valor predeterminado que el sistema necesite para funcionar correctamente desde su inicio. El valor inicial de la válvula debe ser del 100 % para que a la salida del modelo devuelva el máximo de flujo másico de gases, esto es, 20016 kg/h que es el valor que se introduce a través de la variable de inicialización del sistema “Mass flow rate gas” (5.56 kg/s); este valor pasa a la variable de apoyo “salida_modelo” en dicha inicialización. El valor mínimo al 0 % es de 4000 kg/h. Estos valores son los parámetros de diseño del sistema.

El valor inicial de válvula se comparte con las variables de apoyo “sControl2” y “feedback 2” con el fin de no llevar a la inestabilidad ni tener efectos no deseados en el bucle de control.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

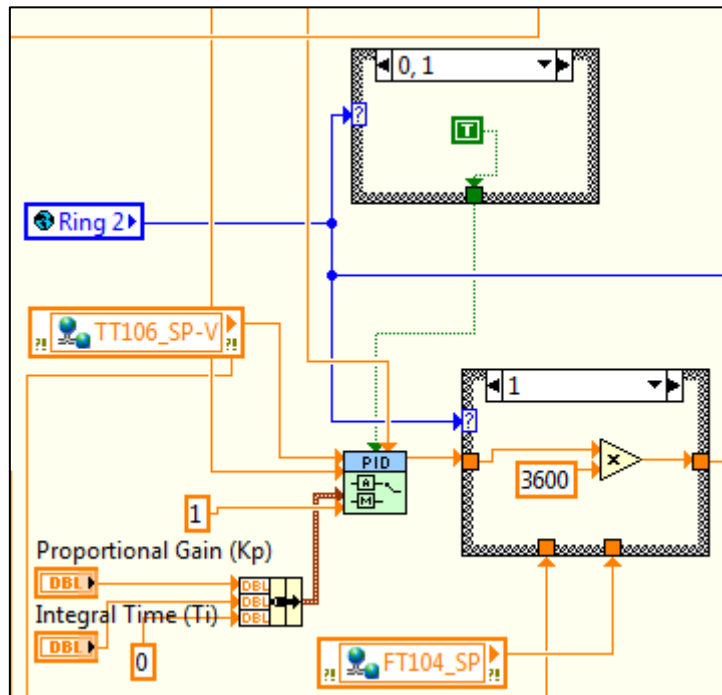


Figura 61. Lazo primario del FIC-104 en el diagrama de bloques

En el lazo primario, si el valor del selector es 0 ó 1 el PI está en modo automático y si el valor es 2 ó 3 está en modo manual y el valor que adquiere es el de flujo másico de gas para que la señal de control de este lazo esté permanentemente informada de qué valor toma.

Entre el lazo primario y el secundario, está la estructura condicional principal, en la cual si el selector está en valor 2 ó 3 no se conectan entre sí porque el PI maestro está desactivado; sin embargo, si está en modo 0 ó 1 sí que se conectan entre sí los dos lazos.

Cuando está en modo 0, se activa la acción del *feedforward*, que se ha implementado en forma de matemática textual, que se adhiere a la señal de control del PI maestro para compensar el efecto de las perturbaciones.

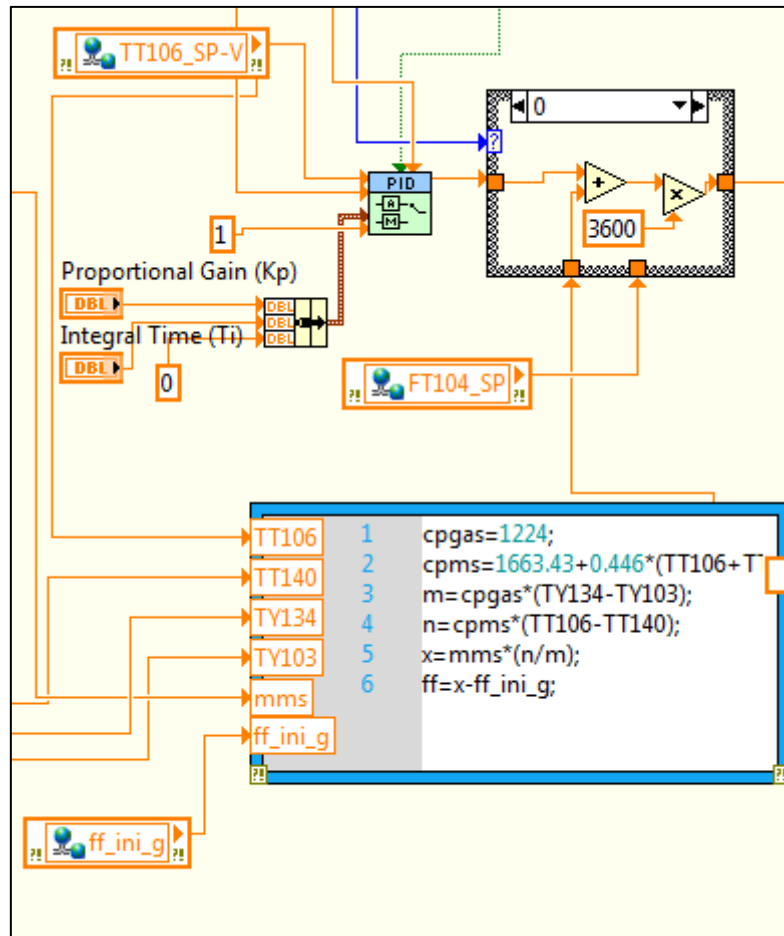


Figura 62. Acción *feedforward* en el lazo primario del FIC-104 en el diagrama de bloques

Exactamente igual que en el lazo FIC-103, el *feedforward* necesita tener un valor inicial para que cuando se active y comience a enviar la señal de control, no tenga que dar cambios bruscos y solo proporcione la cantidad necesaria de flujo másico.

El valor inicial, introducido por la variable “*ff_ini_g*”, se ha calculado de la misma forma que el *feedforward*, simplemente con los valores iniciales de las variables que afectan al *feedforward*. Este valor constante se obtiene cuando se activa la opción PI más *feedforward*; entonces, se ejecuta un evento que está emplazado igualmente en el VI “Simulador_0.2.Main” en el cual se calcula el valor inicial del *feedforward* y se pasa esta información al VI principal donde está el bucle de simulación y el *feedforward* principal.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

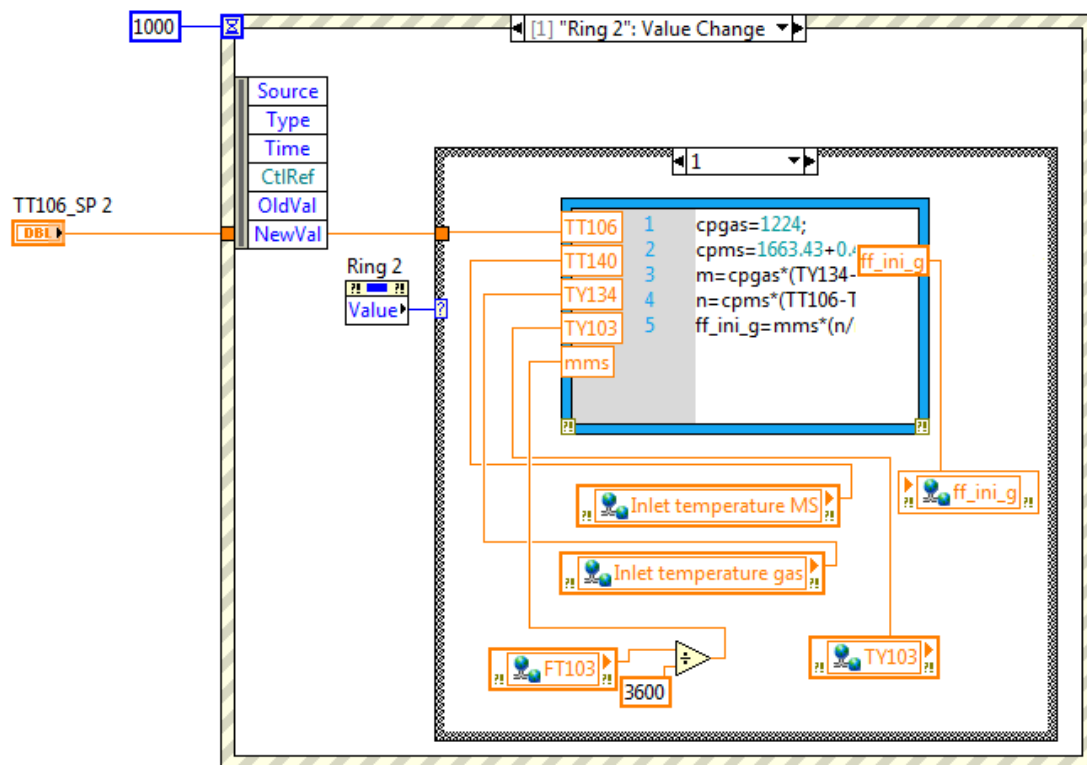


Figura 63. Evento que ejecuta el valor inicial del *feedforward* en el FIC-104

En este VI “Simulador_0.2.Main” también se han almacenado las variables de apoyo de este lazo de control. Estas variables se usan cuando se envían señales de control desde los PID a los modelos, y en vez de conectarlo directamente mediante un cable virtual lo cual produce problemas de inestabilidad, se ha optado por usar otra variable a parte que se conecta al modelo. Estas variables han sido previamente informadas en el “Simulador_0.2.Main” del valor de la señal de control.



Figura 64. Transmisión de la información a la variable de apoyo de la señal de control del FIC-104

Así mismo, para que los valores de referencia y de control manual estén siempre informados de qué valor se está tomando en cada momento, se usa la misma estrategia que con las variables de apoyo, y dependiendo en qué modo de operación esté el lazo de control, se actualizan unas variables u otras.

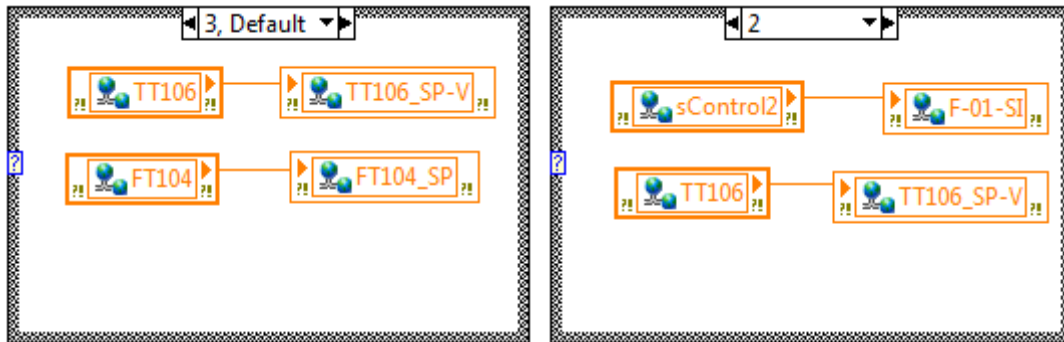


Figura 65. Variables del sistema informando al resto de variables para que estén actualizadas en FIC-104

Estos dos lazos de control FIC-103 y FIC-104 no deben trabajar a la vez, ya que el sistema se inestabilizaría y podría llegar a quedar inutilizable. Para que esto no ocurra, se ha creado una estrategia con la cual cuando uno de los dos lazos esté usando el control PI o PI más *feedforward* de la temperatura de las sales fundidas a la salida del HRS, el otro lazo de control sólo puede estar en modo control manual o control de su flujo másico correspondiente. Así solo se podrá controlar la temperatura de las sales fundidas a la salida del HRS con un solo lazo de control.

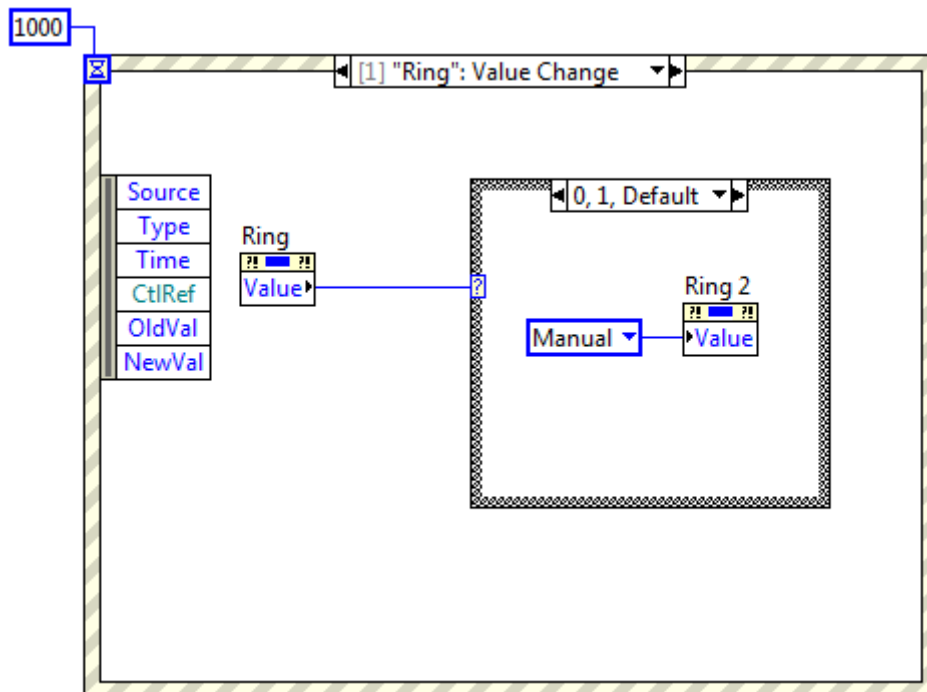


Figura 66. Evento que no permite la confluencia de los dos lazos de control

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

Para ello se ha creado un evento en el VI “Simulador_0.2.Main” que se ejecuta cuando uno de los selectores (“Ring” o “Ring2”) de los lazos de control se cambia a control con PI o con PI más *feedforward*, y obligando al lazo de control opuesto a cambiarse a control manual.

4.5.3. Lazo de control de la turbina de gas virtual

Este lazo de control nos permite controlar la temperatura del gas a la entrada del HRS haciendo uso de curvas de turbinas que simulan el comportamiento de turbinas de gas.

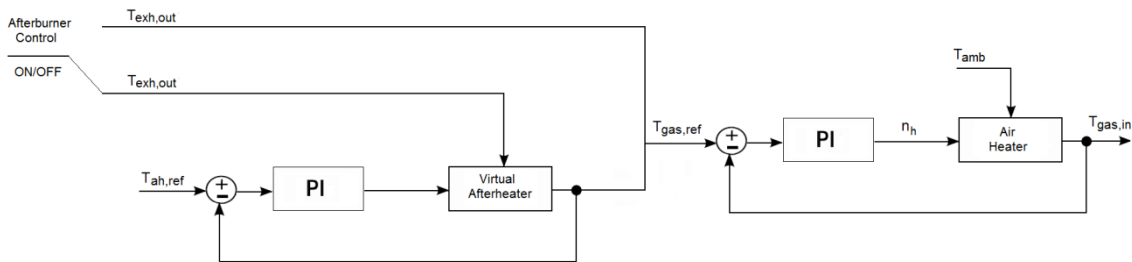


Figura 67. Esquema de control de la turbina de gas virtual

Inicialmente esta estrategia de control está desactivada y la temperatura de los gases a la entrada del HRS es la indicada manualmente en el control F-01-SC2. Una vez que se activa haciendo uso de la opción “Virtual Gas Turbine” del selector, ya se ejecuta el lazo de control.

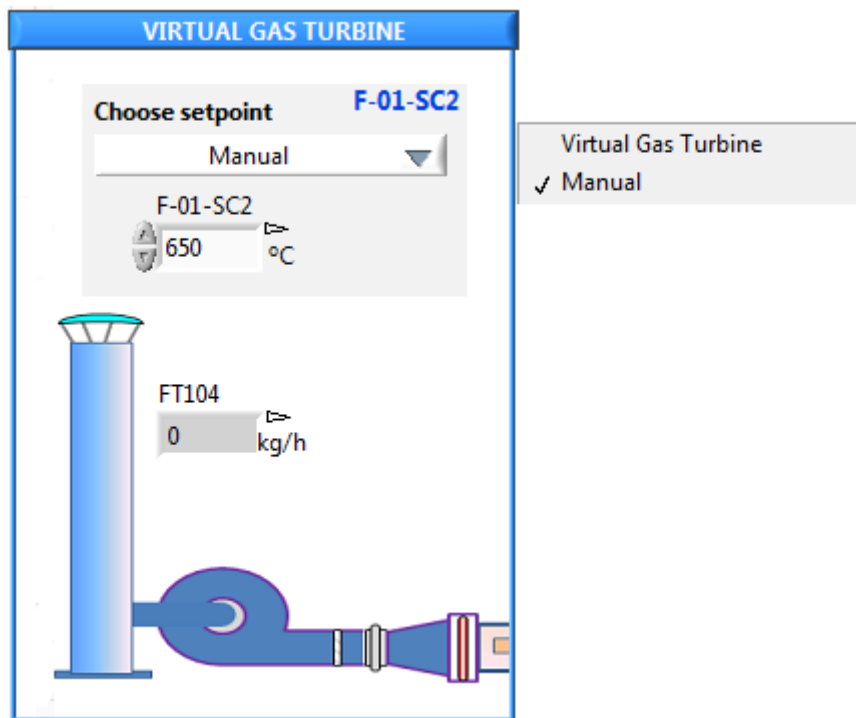


Figura 68. Virtual Gas Turbine en el SCADA

Este esquema de control se compone de dos lazos consecutivos, pero no en modo cascada. El primer lazo de control es el del *Virtual Afterheater* o postcombustor, que consiste en un PI que controla el efecto del postcombustor para incrementar la temperatura de los gases de escape ($T_{\text{exh,out}}$) que provienen de la curva de la turbina hasta la temperatura que se desea a la entrada del HRS que se ha establecido en la consigna ($T_{\text{ah,ref}}$). Todo esto ocurre siempre y cuando se haya activado el control del postcombustor, porque si no se activa, la curva de temperatura o de flujo másico de los gases pasa directamente a la consigna del segundo lazo de control sin tener que ser tratada por el lazo de control del postcombustor. El segundo lazo de control es el del *Air Heater* o calentador de aire, que recibe como referencia el valor ($T_{\text{gas,ref}}$) obtenido del lazo anterior, sea usando postcombustor o no. Si se ha introducido anteriormente una curva de flujo másico en la turbina virtual, el control que se activa es el de flujo másico del lazo de control FIC-104, y a partir de ahí ya controla la cantidad de flujo másico establecido por la curva. Sin embargo, si la curva que se ha introducido es de temperatura de gases, el segundo lazo de control de la turbina es el utilizado. Este bucle consiste en un controlador PI que tiene como referencia los valores de la curva de temperatura directamente si está el postcombustor desactivado o el valor que devuelve el primer lazo de control. La señal de control producida por el PI, que está comprendida entre 0 y 50 % ya que es la carga del calentador (n_h), pasa al modelo del calentador de aire, cuya función es calentar la temperatura ambiente (T_{amb}) mediante combustión para devolver el valor de la temperatura de gases deseada a la entrada del HRS ($T_{\text{gas,in}}$) que se ha establecido en la consigna.

Los PI implementados en este lazo de control tienen incluidos un esquema *anti-windup* y la transferencia sin saltos. Los parámetros de los controladores inicialmente se ajustaron usando el método AMIGO, y finalmente se ajustaron de forma manual. Estos parámetros son los siguientes:

❖ PI del *Virtual Afterheater*:

Constante proporcional: $K_p = 0.4$

Tiempo Integral: $T_i = 30 \text{ s}$

❖ PI del *Air Heater*:

Constante proporcional: $K_p = 0.01$

Tiempo Integral: $T_i = 50 \text{ s}$

Para introducir las curvas de temperatura o de flujo másico, en el SCADA, hay que pulsar el botón "VIRTUAL GAS TURBINE" y entonces aparecerá el VI de la turbina en forma de ventana emergente.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

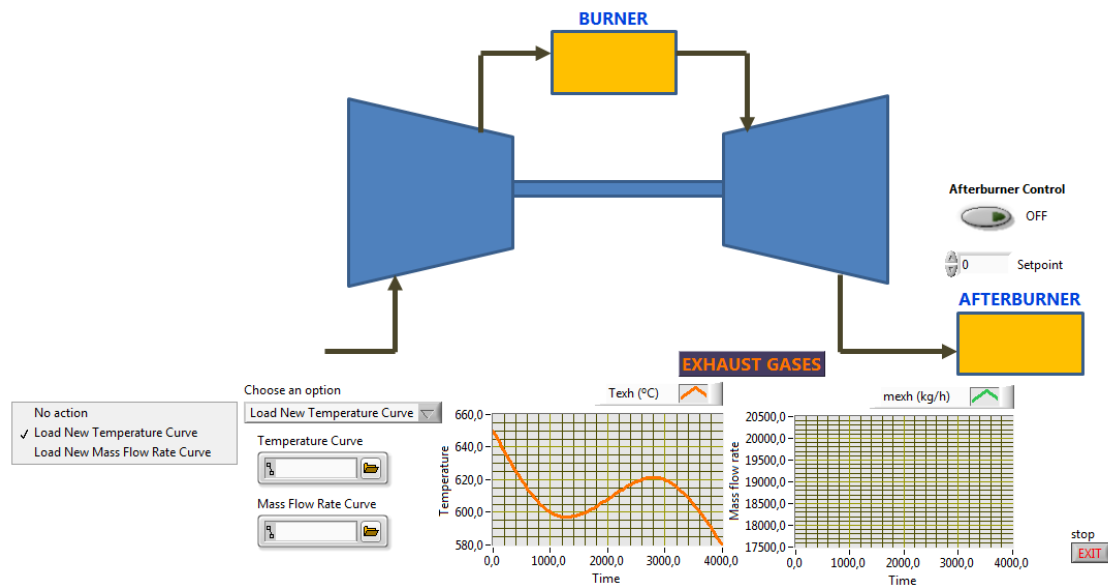


Figura 69. Ventana emergente de la Turbina

Dentro de este VI se pueden cargar y observar las curvas de temperatura o de flujo másico que se quieren utilizar para el funcionamiento del sistema. Estos archivos deben ser archivos de texto (.txt) y tener como máximo 20 datos. Además, se puede establecer la consigna de temperatura de entrada de los gases en el HRS y activar el postcombustor.

El diagrama de bloques de este VI tiene una estructura simple, donde dentro del bucle clásico principal existe un evento con el cual cada vez que se seleccione una opción de cargar curva, se ejecutará una instrucción de lectura de los datos de la gráfica. Esta lectura de datos se hace como si fuera un vector, y a continuación, se interpolan los datos obtenidos para tener una curva lo más continua posible.

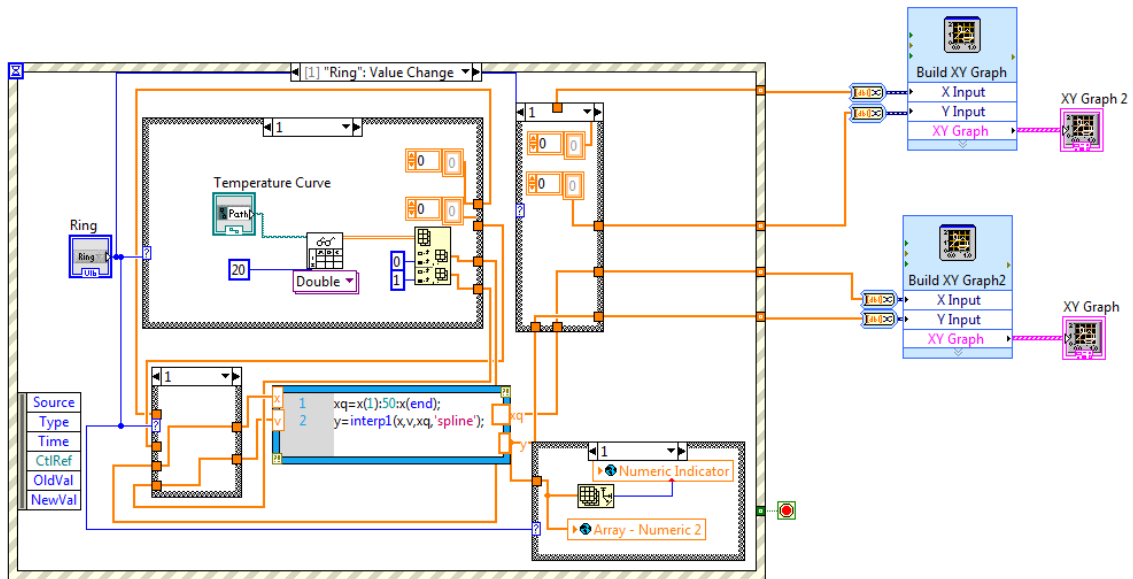


Figura 70. Diagrama de bloques de la ventana emergente de la Turbina

Primero, se amplía el vector de datos añadiendo 50 datos entre cada cifra del archivo, para que se puedan utilizar una gran cantidad de datos de temperatura o flujo másico en el sistema. La interpolación usada ha sido *splines*, ya que proporciona una interpolación continua y suave entre los puntos de dicha curva. Después, una vez interpolados, se muestran los datos en la gráfica y también se guardan en forma de vector en una variable global para poder seguir trabajando con ellos más adelante junto con el número de elementos que hay en el vector.

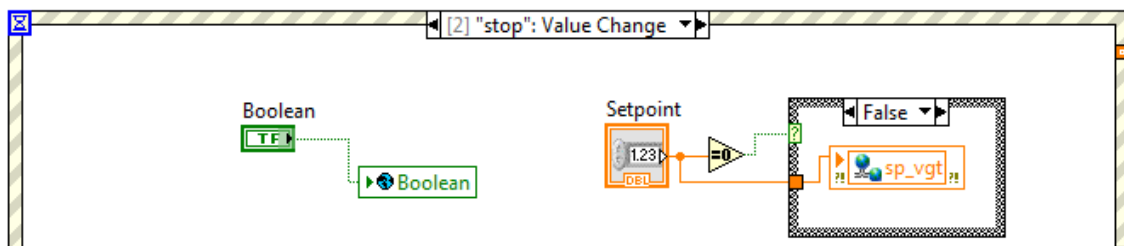


Figura 71. Ventana de finalización de la ventana emergente de la Turbina

Así mismo, cuando se vaya a abandonar la ventana emergente pulsando el botón de stop, se guarda en variables globales y compartidas tanto si se ha activado el postcombustor como si se ha introducido un valor de referencia para la temperatura de gases a la entrada del HRS con el objetivo de transmitir esta información al VI principal de la herramienta.

A la hora de transmitir los valores de la curva de temperatura introducida al VI principal de la herramienta, se ha utilizado el VI "Simulador_0.2.Main". Estos valores se extraen del vector uno a uno gracias al bucle principal de este VI, que por cada iteración, se extrae un valor del vector hasta que se vacíe. Estos valores se almacenan en la variable F-01-SC2 y se transmiten al VI principal de la herramienta.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

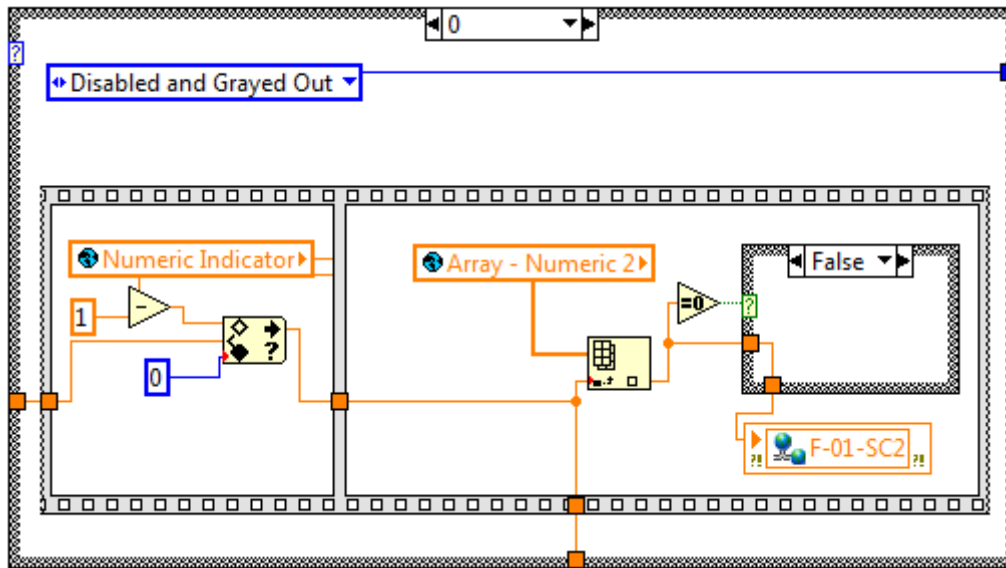


Figura 72. Desindexación del vector de datos de la curva de temperatura

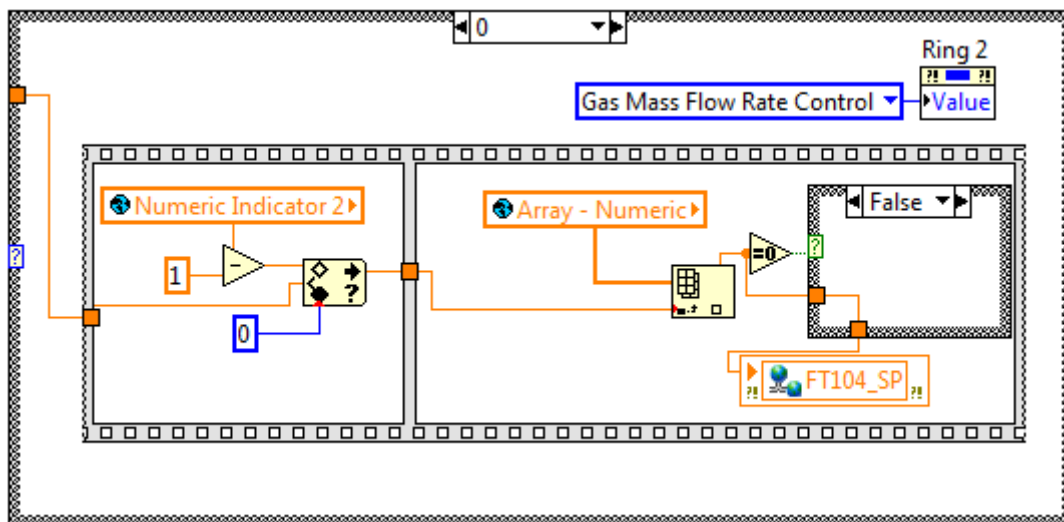


Figura 73. Desindexación del vector de datos de la curva de flujo másico

Cuando introducimos una curva de flujo másico, el procedimiento es el mismo, a excepción de que en vez de pasar los valores a la variable F-01-SC2, se almacenan en la variable FT104_SP para que se consiga el valor de flujo másico deseado utilizando el lazo de control FIC-104.

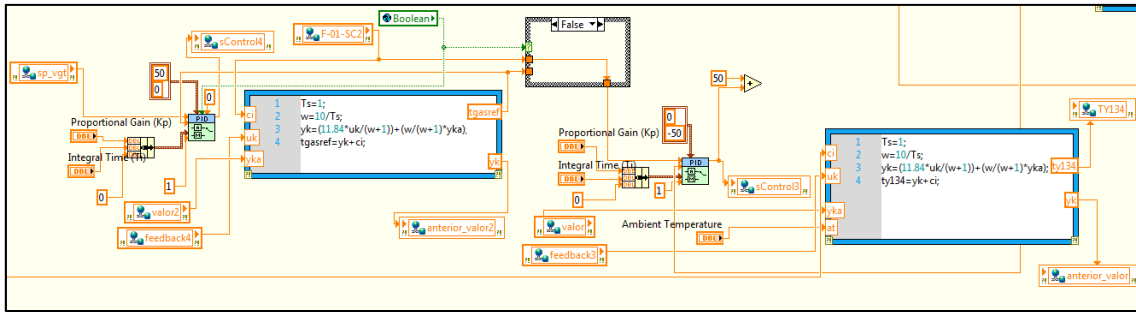


Figura 74. Diagrama de bloques del esquema de control de la turbina de gas virtual

En el VI principal de la herramienta (“Simulador_0.3”), se han implementado los dos lazos de control que regulan el comportamiento de la turbina de gas virtual. En primer lugar se tiene el lazo de control del postcombustor, en el cual se ha usado el bloque PID Advanced de LabVIEW como en el resto de esquemas de control. Este controlador tiene como referencia el valor que se ha establecido anteriormente en la ventana emergente representado en la variable “sp_vgt”. El PI envía señales de control al modelo del postcombustor haciendo uso de las variables de apoyo “sControl4” y “feedback4” para evitar comportamientos no deseados como se ha explicado anteriormente antes en los otros lazos de control. Existe una estructura condicional que está gobernada por la variable booleana que controla la activación o desactivación del postcombustor que también activa o desactiva este lazo de control. Cuando está activado, el valor de entrada a la referencia del segundo lazo de control es la salida de este lazo de control; sin embargo cuando está desactivado, el valor que entra al segundo lazo de control es el de la curva de temperatura de gases de escape de la turbina directamente representada por la variable F-01-SC2.

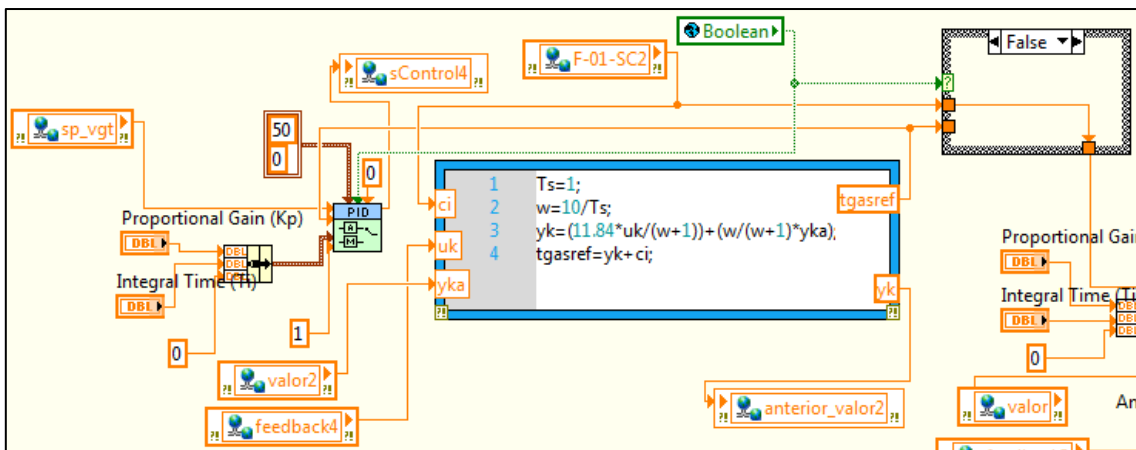


Figura 75. Diagrama de bloques del primer lazo de control del esquema de control de la turbina de gas virtual

Para el modelo del postcombustor se ha utilizado el mismo modelo que se usa en el calentador de aire para obtener la temperatura de entrada de los gases al HRS, representado mediante la siguiente función de transferencia:

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

$$G(s) = \frac{11.84}{10s + 1} \quad (13)$$

Este modelo se ha incluido en la herramienta en forma de ecuaciones en diferencias para así optimizar su comportamiento y conseguir unos resultados satisfactorios. Por ello es necesario usar las variables de apoyo “valor2” y “anterior_valor2” para ir informando al propio modelo de cada paso producido por las ecuaciones.

La salida de este modelo nos devuelve el valor de la temperatura de gas de referencia que cierra el bucle y que entra al segundo lazo de control siempre y cuando esté activado el postcombustor.

El segundo lazo de control, el del calentador de aire, también usa el bloque PID Advanced. Una vez recibe la referencia ($T_{gas,ref}$), el PI envía señales de control al modelo del calentador de aire mediante las variables de apoyo “sControl3” y “feedback3” para evitar comportamientos no deseados como se ha explicado anteriormente antes en los otros lazos de control. El modelo de calentador de aire que se usa para obtener temperatura es el anteriormente citado:

$$G(s) = \frac{11.84}{10s + 1} \quad (14)$$

E igualmente se ha implementado en forma de ecuaciones en diferencias y también se han usado las variables de apoyo “valor” y “anterior_valor” con el mismo fin que en el primer lazo de control. La salida de este modelo es la temperatura de los gases a la entrada del HRS ($T_{gas,in}$) que se almacena en la variable TY134.

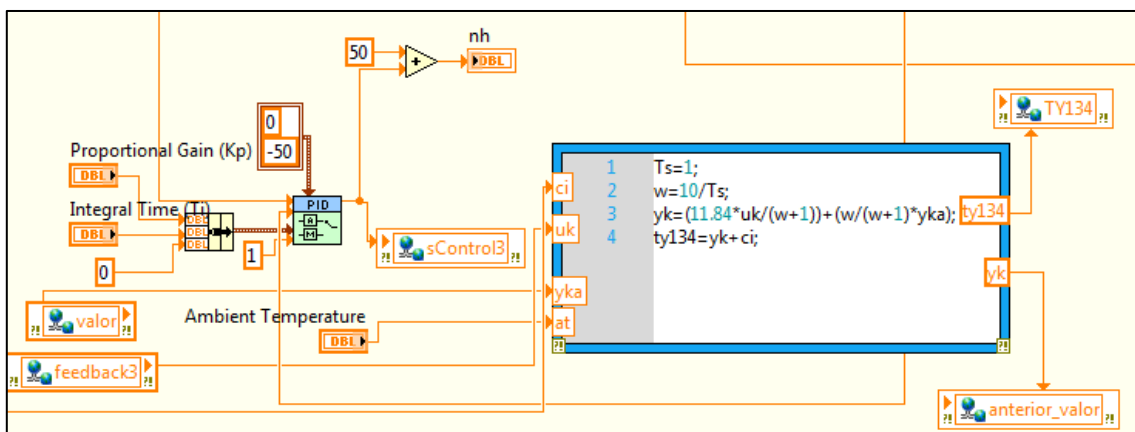


Figura 76. Diagrama de bloques del segundo lazo de control del esquema de control de la turbina de gas virtual

4.6. Alarmas

Las alarmas son una parte importante del sistema, ya que avisan al usuario constantemente de fenómenos extraños en variables relevantes del sistema.

Las variables que se han tomado en cuenta para hacer las alarmas son las siguientes:

Nombre variable	Descripción	Alarma	Valor	Unidad
LT129	Nivel del tanque de sales fundidas	HH	900	mm
		H	800	mm
		L	240	mm
		LL	140	mm
TT106	Temperatura de las sales fundidas a la salida del HRS	H	565	°C
		L	290	°C
FT103	Flujo másico de sales fundidas en el sistema	L	5462	kg/h
		LL	5159	kg/h

Tabla 6. Variables utilizadas para las alarmas

Las alarmas se muestran en el panel frontal de la herramienta en la pestaña “ALARMS” en la siguiente tabla:

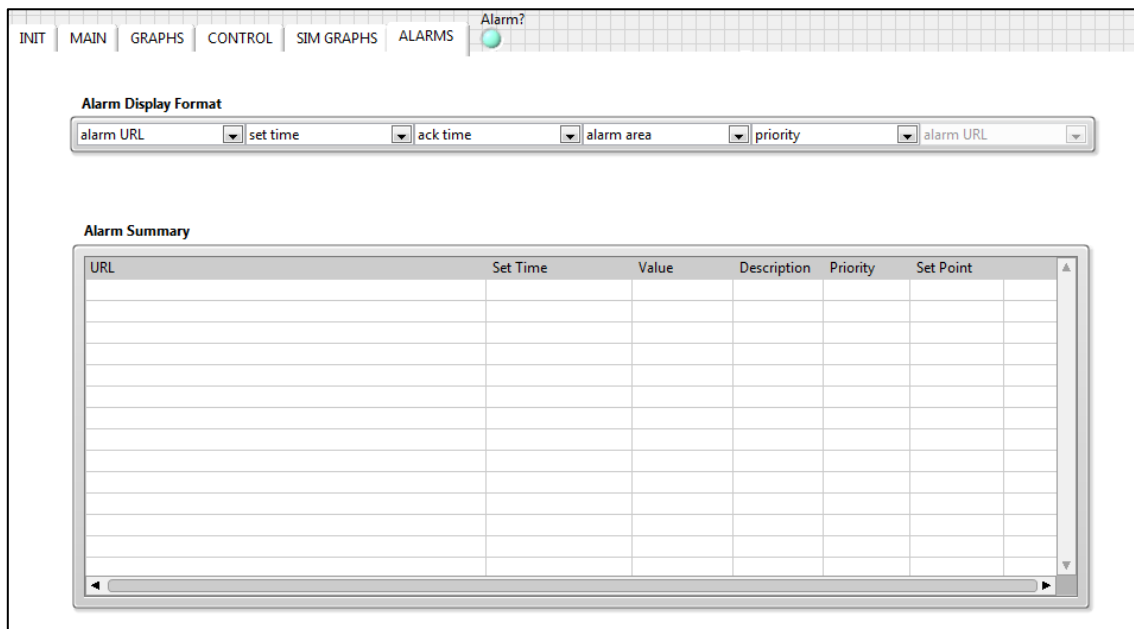


Figura 77. Visión de las alarmas en el SCADA

En esta tabla se muestran todos los eventos de las alarmas donde se describe qué alarma ha saltado, el tiempo en el cual se ha ejecutado, el valor que toma, la descripción, la prioridad y el valor que debería tomar como máximo para no saltar. De todas formas estos

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

parámetros se pueden cambiar en el selector que aparece encima de la tabla. A su vez, hay un indicador en forma de led que se torna en color rojo para avisarnos si ha aparecido alguna alarma.

En el diagrama de bloques las alarmas se actualizan en cada iteración del bucle en el que están implementadas. Este bucle está en paralelo con el bucle principal de la herramienta en el VI "Simulador_0.3". Lo primero que se hace es leer la lista de variables compartidas y limpiar la tabla de las alarmas que se quedaron mostradas en la última utilización de la herramienta. A continuación, se van leyendo las variables para comprobar en todo momento que no ocurre ningún evento anormal.

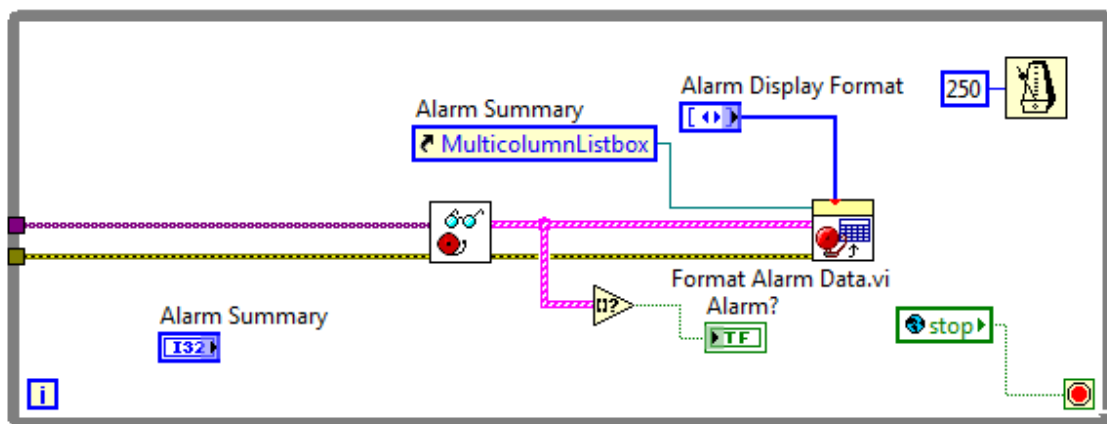


Figura 78. Diagrama de bloques de las alarmas

Las variables compartidas, a la hora de su configuración tienen una opción que es la de *Alarming*, que se habilita y permite establecer el tipo de alarma y su valor límite dependiendo de dicho tipo.

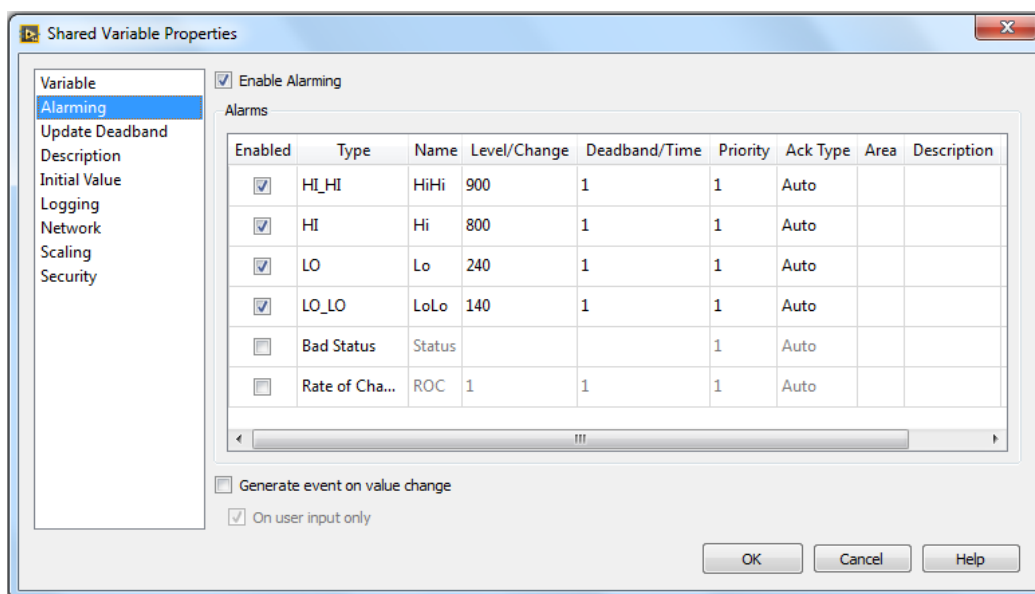


Figura 79. Configuración de las alarmas en las variables compartidas

5. Resultados y discusión. Conclusiones

5.1. Resultados y discusión

Una vez iniciada la herramienta y el simulador, los parámetros del sistema alcanzan sus valores iniciales (ver figura 80).

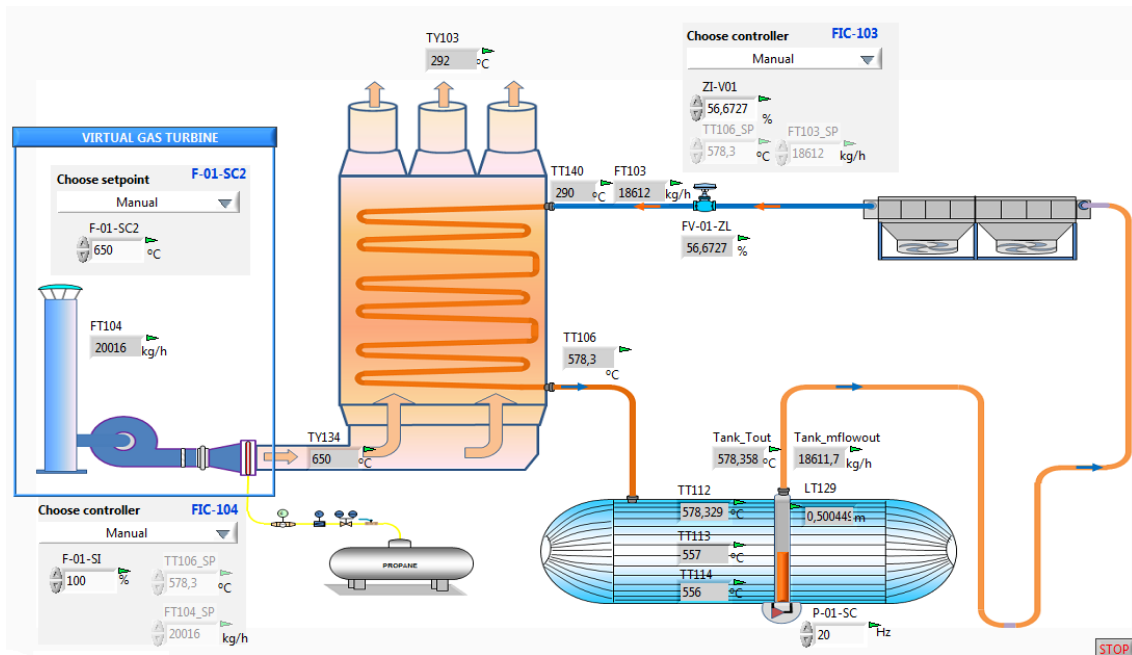


Figura 80. Esquema del demostrador de la planta en su configuración inicial

Para comprobar el lazo de control FIC-103, primero se establece una consigna de flujo másico de sales fundidas en 20000 kg/h para probar el modo de control del flujo másico.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

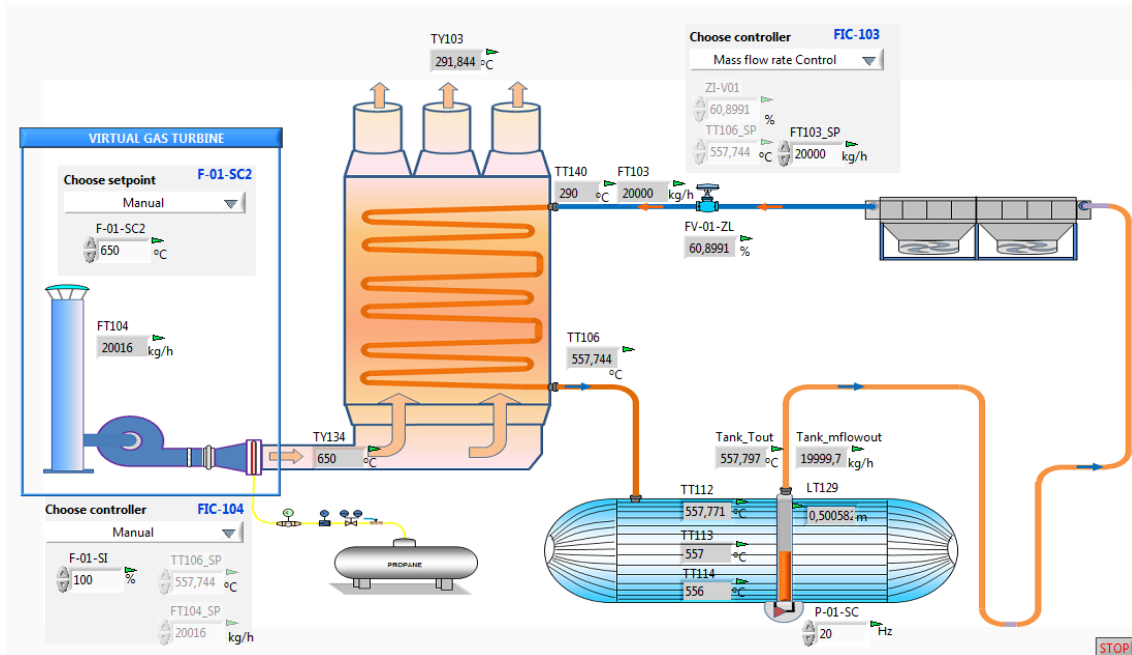


Figura 81. Sinóptico de la planta usando el control de flujo másico del FIC-103

A continuación (figura 82) se muestran las gráficas que representan las dinámicas de la temperatura de las sales fundidas a la salida del HRS (TT106), y la del flujo másico de sales fundidas a la entrada del HRS (FT103). Se puede observar que el flujo másico alcanza el valor predeterminado en la consigna.

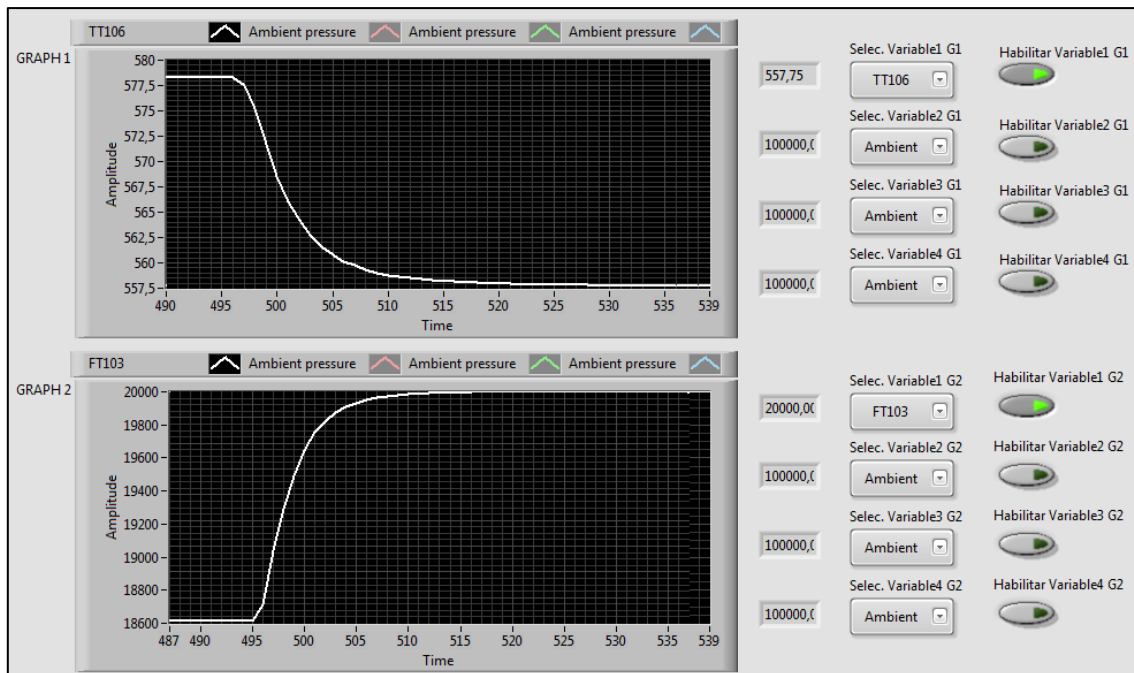


Figura 82. Gráficas que enseñan la dinámica de la temperatura de sales fundidas TT106 y del flujo másico de sales FT103

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

Ahora se analiza el control de temperatura con el flujo másico como variable de control. Primero se cambia de modo de operación para ponerlo en el modo de control de temperatura con PI, y esta vez se ha optado además por usar el *feedforward*. Se establece la consigna a 565 °C.

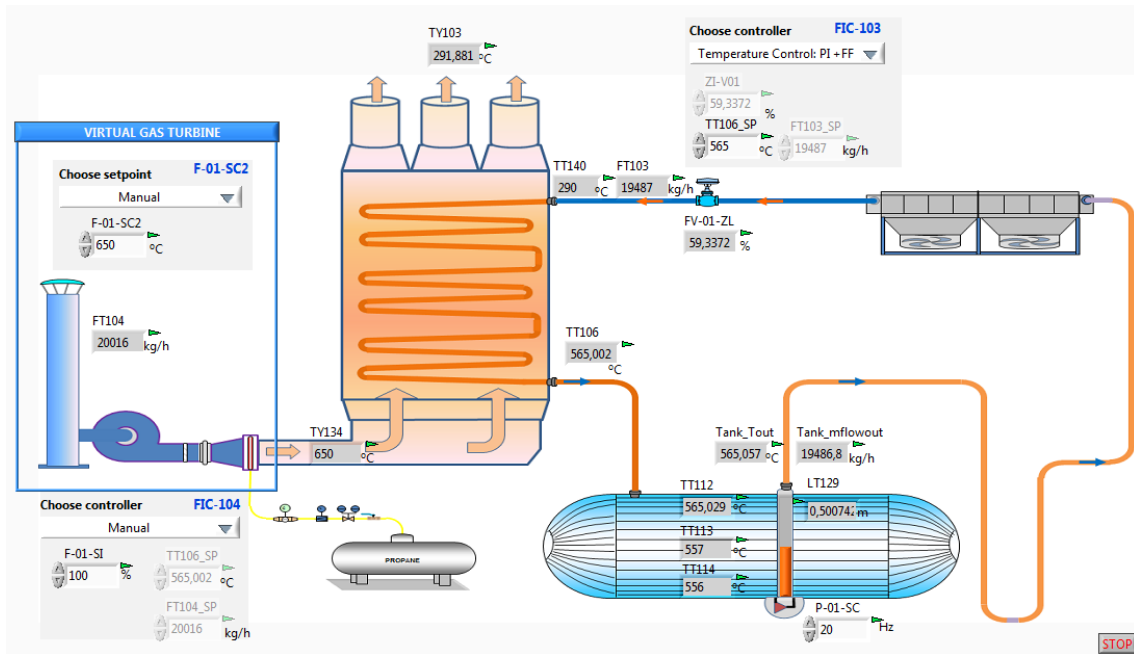


Figura 83. Permuta al control de temperatura con *feedforward* en el FIC-103

En la siguiente imagen (figura 84) se puede ver cómo la temperatura consigue llegar a su nuevo valor de consigna y a su vez la cantidad de flujo másico de sales fundidas se modifica a la que ha sido necesaria para obtener dicha temperatura.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

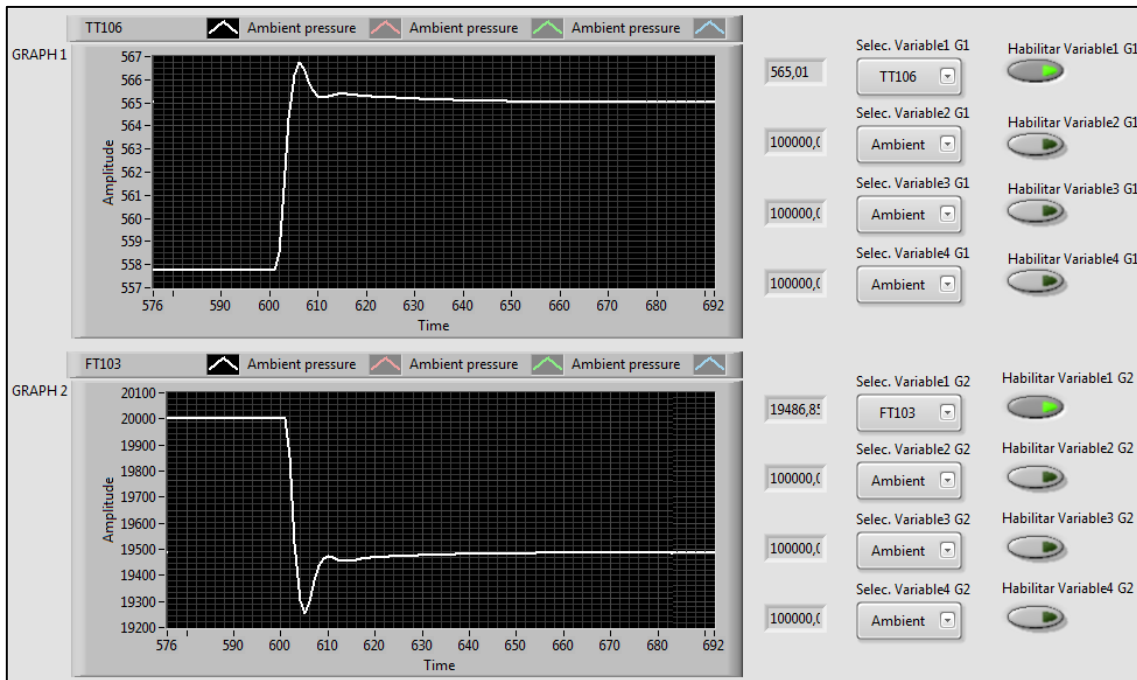


Figura 84. Gráficas que muestran la dinámica de la temperatura de sales TT106 y del flujo másico de sales FT103

Ahora se va a comprobar el lazo de control FIC-104, y para ello se cambia de un lazo de control a otro, seleccionándose el control de temperatura con PI y teniendo las consignas a los valores que tenía el sistema en ese momento.

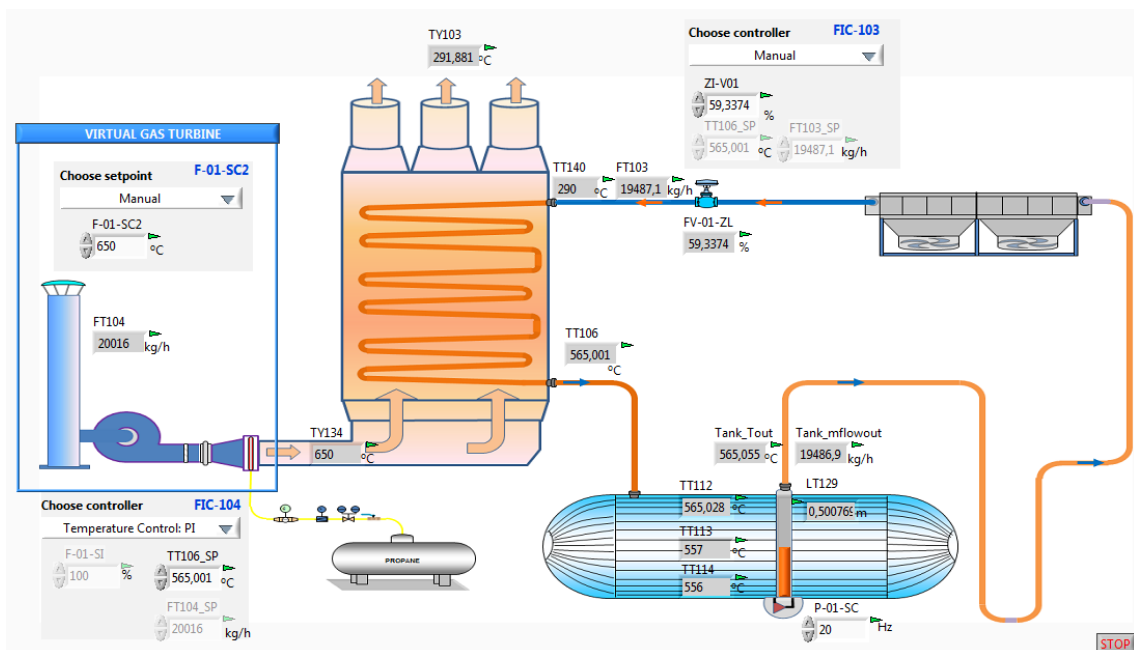


Figura 85. Activación del lazo de control FIC-104

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

Se introduce una nueva consigna para la temperatura de 560 °C en el modo control de temperatura con flujo másico de gases como variable de control para que se aprecie el funcionamiento del lazo de control.

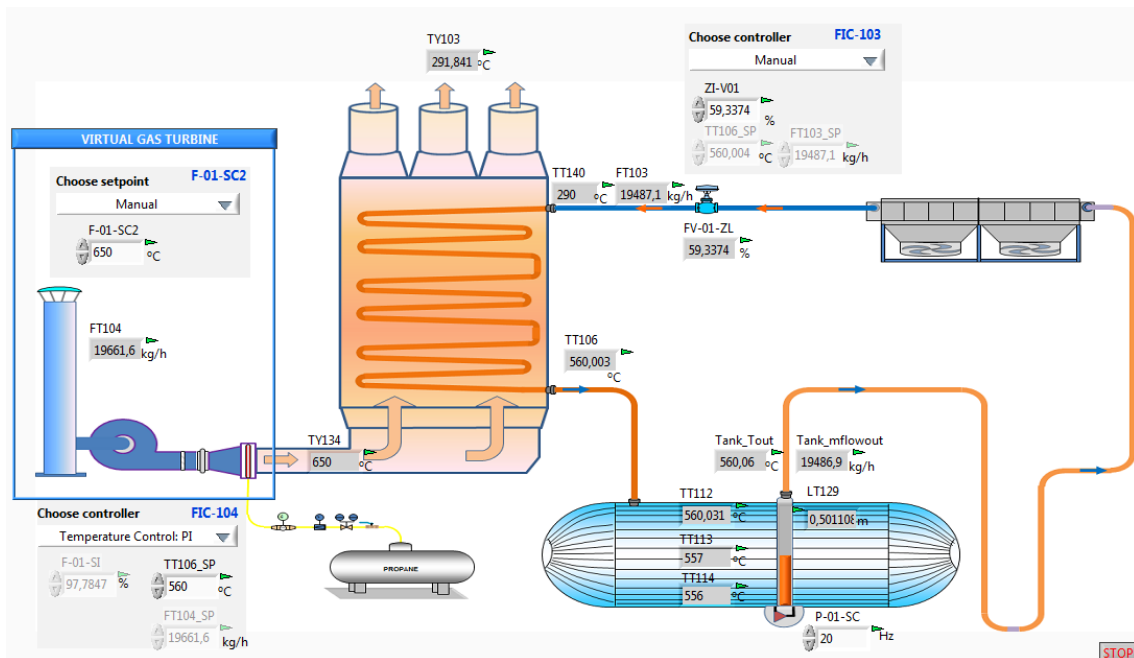


Figura 86. Control de temperatura activado en el FIC-104

A continuación (ver figura 87) se muestran las dinámicas representativas tanto de la variable a controlar (TT106) que alcanza su consigna satisfactoriamente como la del flujo másico de gases (FT104) que es la variable de control, la cual se modifica para obtener dicha temperatura de sales fundidas. También se puede apreciar cómo el flujo másico de sales fundidas (FT103) ya no se ve modificado, ya que se ha dejado de usar el lazo de control FIC-103 y está en modo manual.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

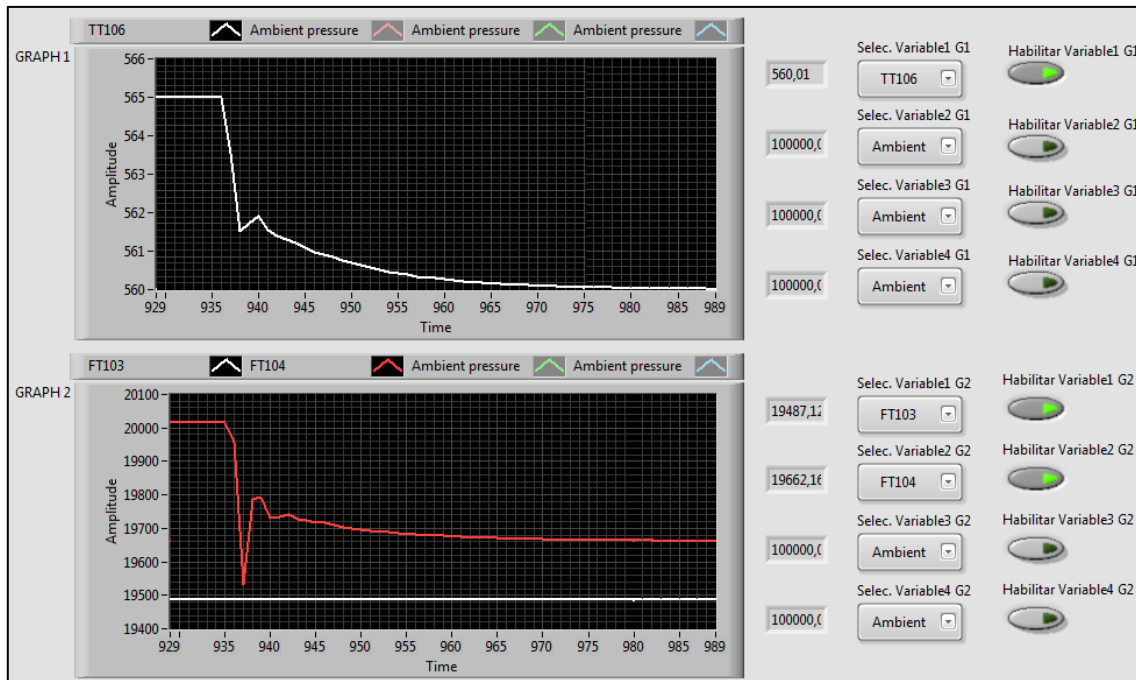


Figura 87. Gráficas que enseñan la dinámica de la temperatura de sales TT106 y del flujo másico de gases FT104

Ahora se cambia de modo para usar el modo de control de flujo másico de los gases para su comprobación y se establece una consigna en 18000 kg/h.

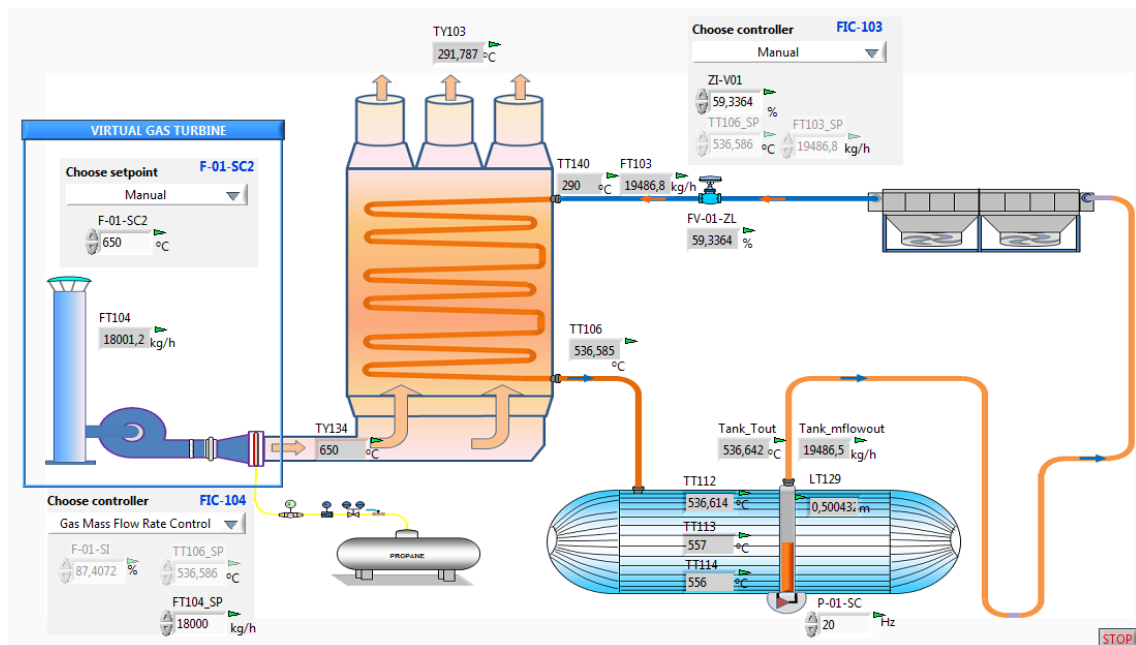


Figura 88. Cambio al control de flujo másico de gases en el lazo FIC-104

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

En esta imagen (figura 89) se puede observar cómo el valor del flujo másico de gases alcanza el deseado establecido en la consigna y a su vez, la temperatura de las sales fundidas se ven afectadas y disminuye.

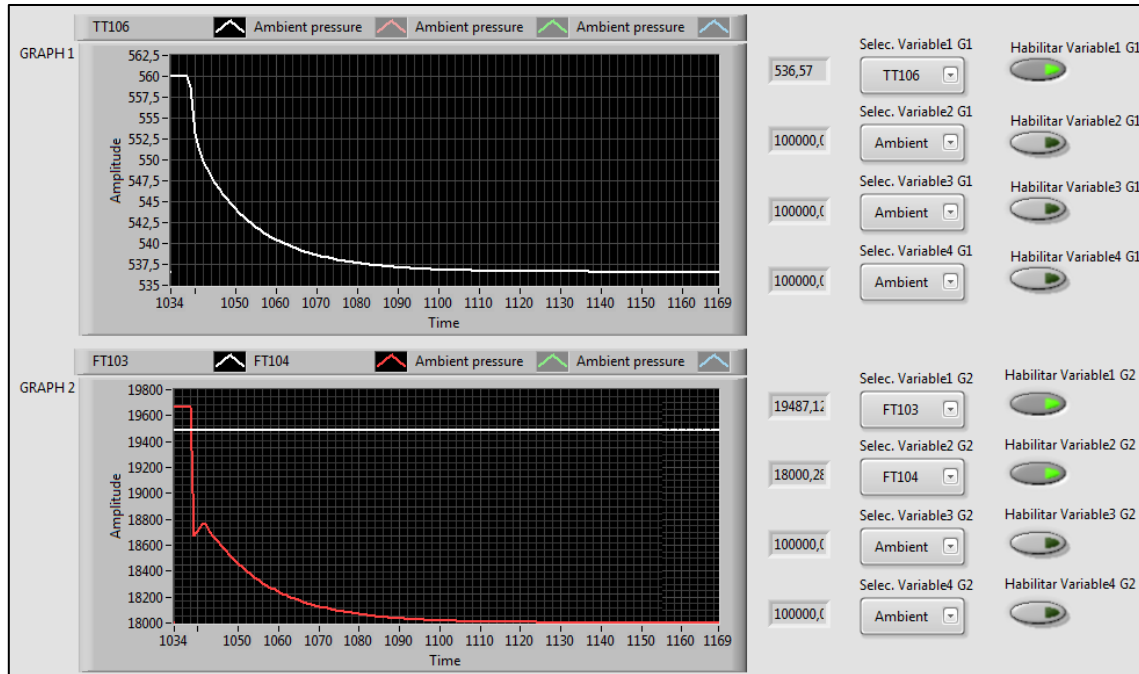


Figura 89. Gráficas que muestran la dinámica de la temperatura de sales TT106 y del flujo másico de gases FT104

Una vez comprobados los dos lazos de control que conciernen a la temperatura de las sales fundidas a la salida del HRS, ahora se comprobará el lazo de control de la turbina de gas virtual. Para ello, se pulsa el botón de “*Virtual Gas Turbine*” para que aparezca la ventana emergente (ver figura 90).

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

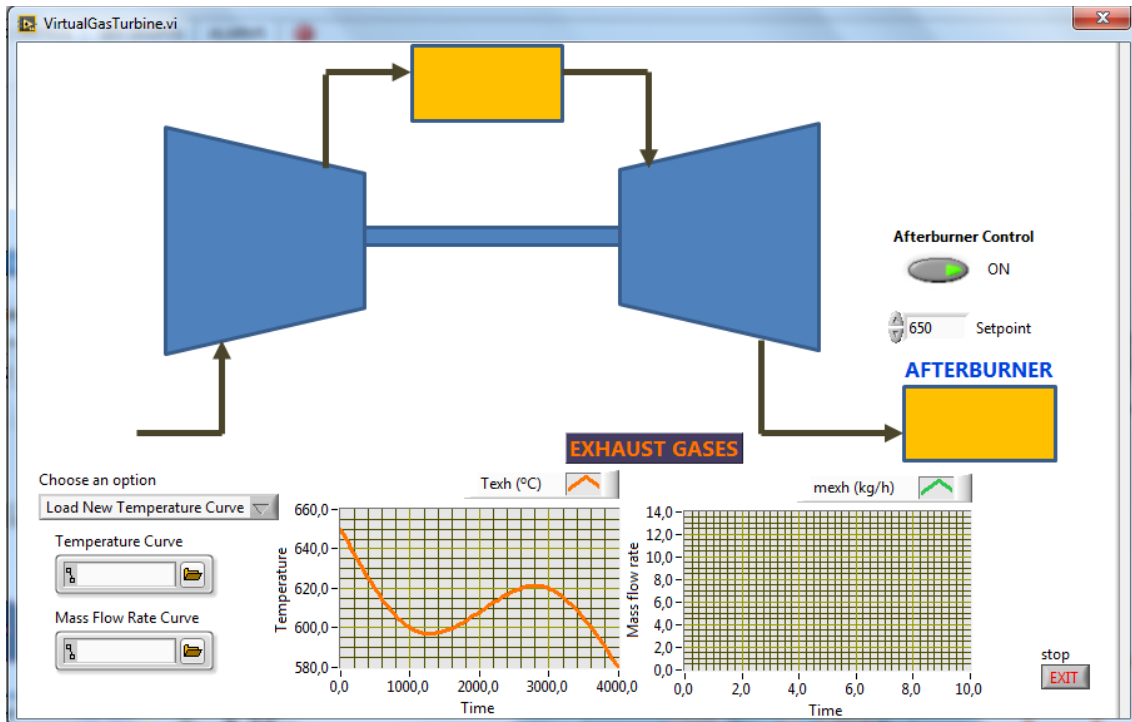


Figura 90. Ventana donde aparece la opción de la turbina de gas virtual

Primero, se carga una curva de temperatura y se activa el postcombustor incluyéndole una referencia de temperatura del gas de 650 °C a la entrada del HRS.

En la figura 91 se puede observar cómo, una vez introducidos los datos anteriores, en el sinóptico principal se van actualizando todos los valores del sistema.

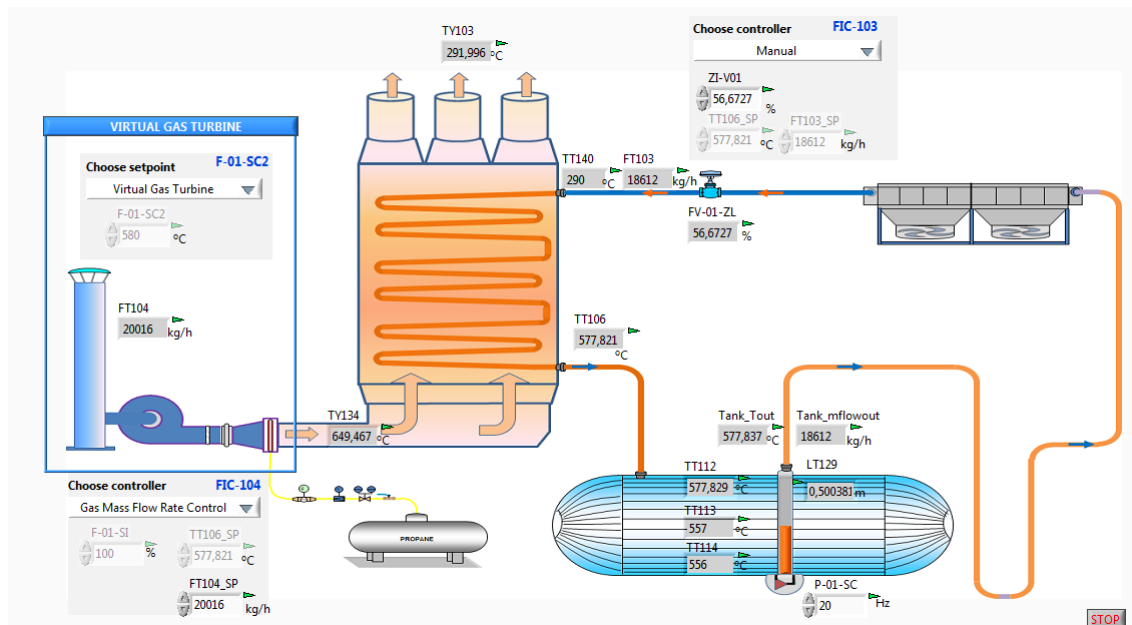


Figura 91. Sinóptico de la planta donde se está utilizando el lazo de control de la turbina de gas virtual

En la siguiente imagen (figura 92) se puede observar el comportamiento de la temperatura del gas a la entrada del HRS (TY134) que alcanza la consigna deseada.

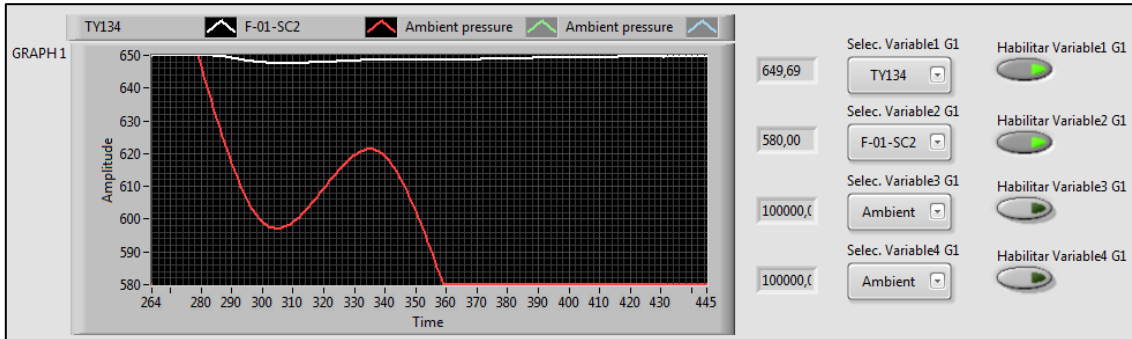


Figura 92. Gráfica donde se enseña la dinámica de la temperatura del gas a la entrada del HRS TY134

Ahora se va a repetir el procedimiento pero esta vez introduciendo una curva de flujo másico de gases con el postcombustor desactivado ya que no es necesario.

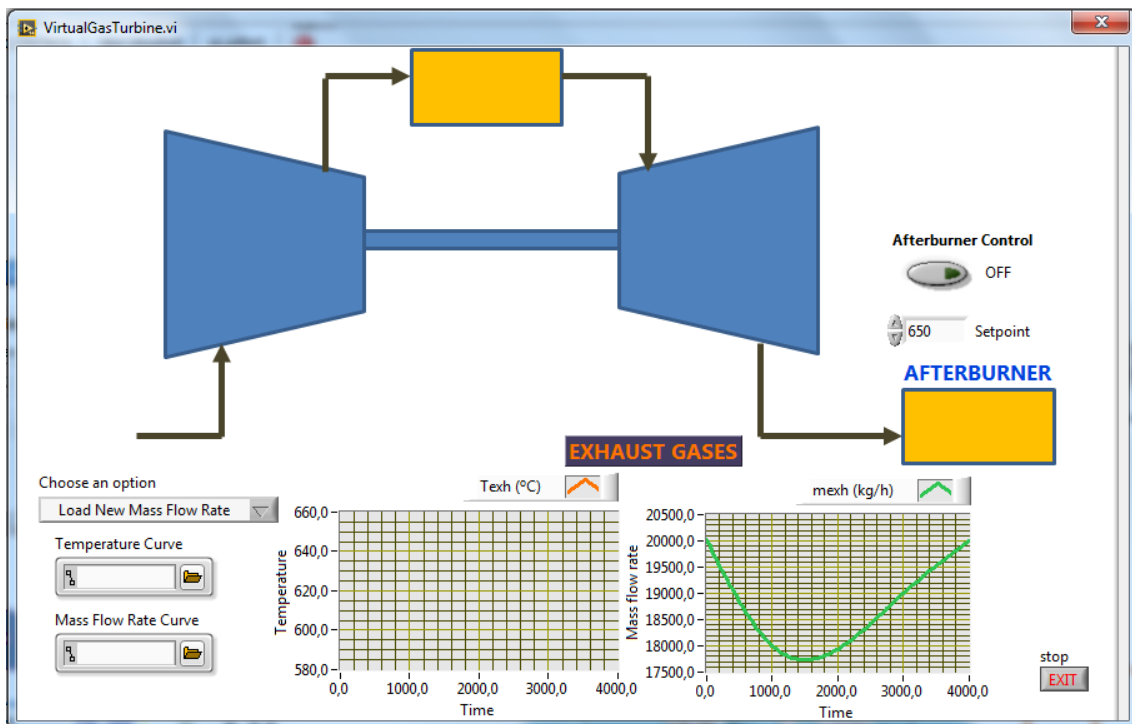


Figura 93. Ventana donde aparece la opción de la turbina de gas virtual

Se puede comprobar como en el sinóptico de la planta se ha activado el control automático de flujo de gases, ya que al cargar una curva de gases, es el control que se debe activar.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

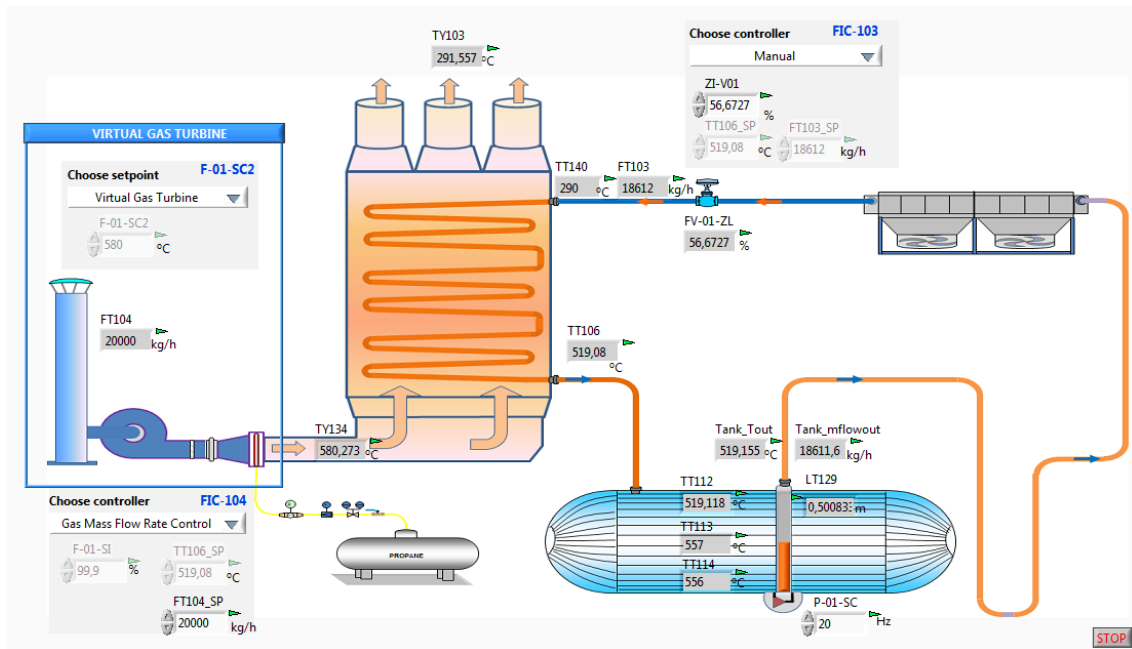


Figura 94. Sinóptico de la planta usando el lazo de control de la turbina de gas virtual

Se observa en la siguiente imagen (figura 95) cómo el flujo másico de gases (FT104) llega a su punto de consigna y a su vez se ve afectada la temperatura de los gases a la entrada del HRS (TY134) disminuyéndose.

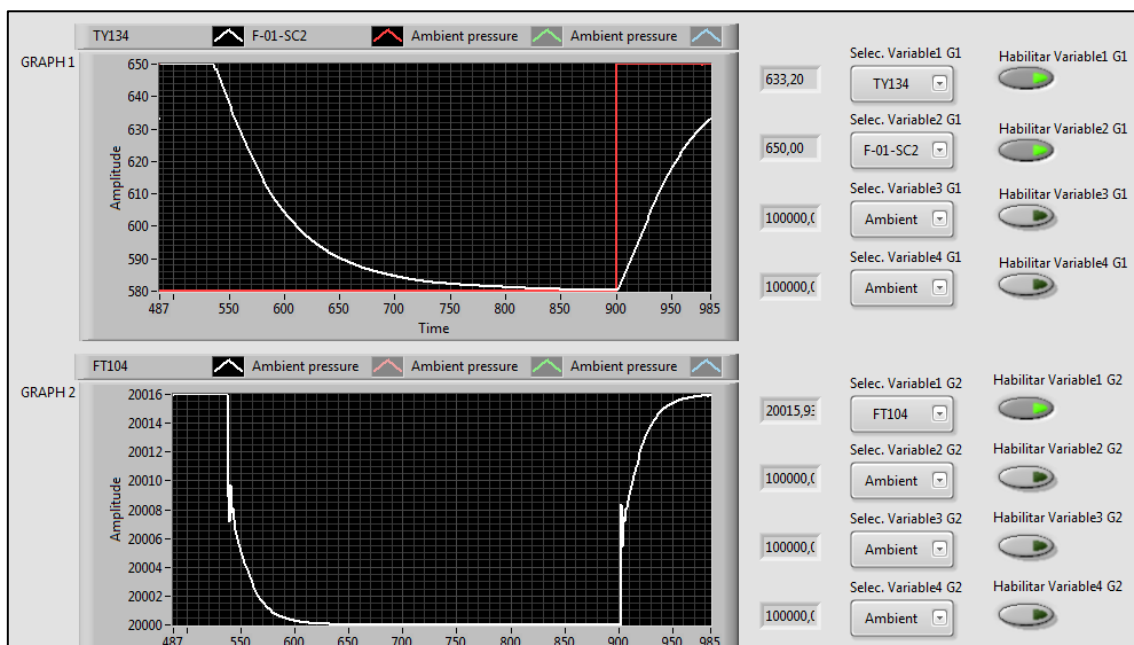


Figura 95. Gráficas donde se representa la dinámica de la temperatura del gas a la entrada del HRS TY134 y del flujo másico de gases FT104

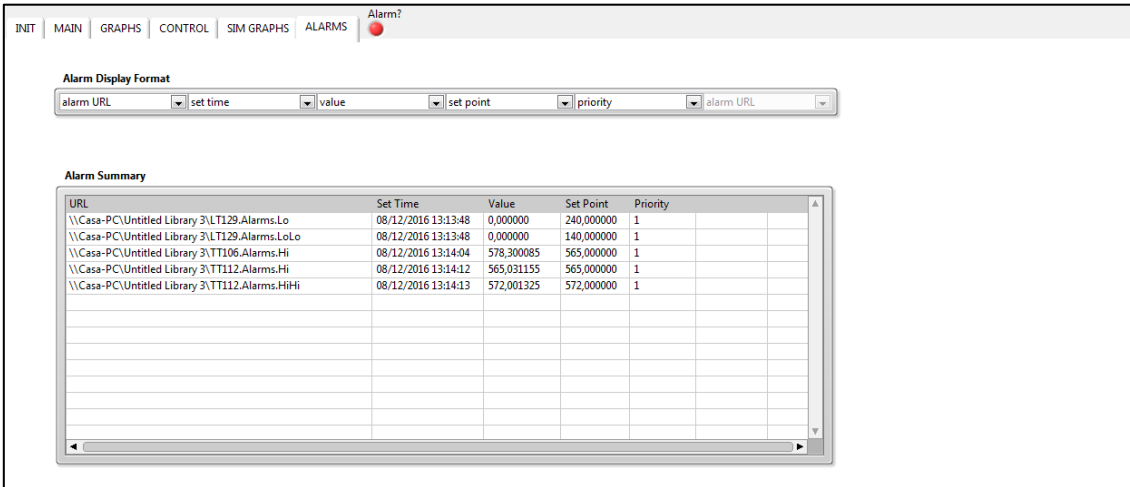
A su vez se ha hecho el cambio de "Virtual Gas Turbine" a "Manual" poniendo la consigna en 650 °C, por lo que el sistema de control nos lleva la temperatura de gases a la entrada

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

del HRS (TY134) a su consigna inicial y ahora, es el flujo másico de gases (FT104) el que se ve afectado incrementando su valor.

Por último, se va a proceder a probar las alarmas. Este procedimiento es muy sencillo: cuando el sistema detecta que el valor está por encima o por debajo del predeterminado anteriormente, salta la alarma en la tabla que aparece en el apartado de alarmas de la herramienta, donde se puede observar el tipo de alarma que ha saltado, su valor de referencia y la hora a la que ha saltado. Para saber cuándo ha saltado una alarma, existe un indicador en forma de led que torna en rojo cuando una alarma se ha activado.



URL	Set Time	Value	Set Point	Priority
\\Casa-PC\Untitled Library 3\LT129.Alarms.Lo	08/12/2016 13:13:48	0,000000	240,000000	1
\\Casa-PC\Untitled Library 3\LT129.Alarms.LoLo	08/12/2016 13:13:48	0,000000	140,000000	1
\\Casa-PC\Untitled Library 3\TT106.Alarms.Hi	08/12/2016 13:14:04	578,300085	565,000000	1
\\Casa-PC\Untitled Library 3\TT112.Alarms.Hi	08/12/2016 13:14:12	565,031155	565,000000	1
\\Casa-PC\Untitled Library 3\TT112.Alarms.HiHi	08/12/2016 13:14:13	572,001325	572,000000	1

Figura 96. Cuadro de alarmas en su ventana correspondiente del SCADA

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

5.2. Conclusiones

En este trabajo de fin de grado se ha presentado el proyecto HYSOL, proyecto que aporta configuraciones novedosas para plantas termosolares basadas completamente en energías renovables, junto con la herramienta software que incluye el modelo dinámico de un recuperador de calor entre gases de combustión de biogases y sales fundidas para la simulación del demostrador de la planta.

El objetivo principal que se ha cumplido en este trabajo de fin de grado ha sido el desarrollo de la herramienta para operación y entrenamiento de operadores del demostrador del proyecto HYSOL. Dicha herramienta software incluye: el modelo dinámico del demostrador (recuperador de gases, simulador de turbina de gas, etc.) con el fin de estudiar su comportamiento; distintas estrategias de control automático para poder testear y validar nuevas estrategias de control; y un sistema SCADA.

Gracias a la herramienta se ha podido estudiar la dinámica del sistema y se han podido desarrollar estrategias de control avanzadas que han sido analizadas y probadas en simulación antes de aplicarse en el demostrador real. Lo que ha permitido ajustar los controladores sin poner en riesgo la instalación real y sin tener que llevar a cabo costosos experimentos.

Dentro de los futuros trabajos que se pueden llevar a cabo, uno de los más importantes es desarrollar otros dos modos de funcionamiento: real e híbrido. El modo real permite aplicar las estrategias de control sobre la planta real, mientras que el modo híbrido permite la comparación en tiempo real de medidas experimentales de planta respecto a resultados de simulación y podría ser aplicado, una vez el modelo haya sido validado, para la detección de fallos en la planta real.

Finalmente, otros objetivos futuros significativos es el diseño, prueba y validación de nuevas estrategias de control avanzadas para optimizar el rendimiento de la planta, además de la inclusión de modelos para los componentes del sistema que hasta ahora no se han podido realizar, como el *Air Cooler*, etc.

La realización de este trabajo de fin de grado ha proporcionado nuevos conocimientos en el contexto de la ingeniería que se dedica a la realización de herramientas y software específico para la simulación de procesos industriales, como pueden ser plantas fundamentadas en nuevas tecnologías basadas en energías renovables. Además, es importante percatarse de que este ámbito es realmente necesario y primordial ya que a partir de un exitoso desarrollo de entornos de simulación, en muchas ocasiones se procederá a la construcción de sistemas realmente costosos y complejos.

6. Bibliografía

- [1] Pérez-López, E. (2015): “Los sistemas SCADA en la automatización industrial”. *Tecnología en Marcha*, 28(4), 3-14.
- [2] Rodríguez Penin, Aquilino (2006): *Sistemas SCADA*, 1.ª ed. Barcelona: Marcombo.
- [3] Consorcio Proyecto HYSOL (2013). [En línea]. Disponible: <http://www.hysolproject.eu>. [Último acceso: 3 de agosto de 2016].
- [4] Bonilla, Javier, Lidia Roca, Eduardo Cerrajero, Luis E. Díez, Lucía González, Diego López, Santiago Mirabal, Silvia Padilla, y Alberto R. Rocha (2016): “Operation and Training Tool for a Gas - Molten Salt Heat Recovery Demonstrator Facility”, *Procedia Computer Science*.
- [5] Bonilla, Javier, Alberto de la Calle, Sebastián Dormido y Lidia Roca (2016): “Modelo Dinámico de un Recuperador de Gases - Sales Fundidas para una Planta Termosolar Híbrida de Energías Renovables”, *Revista Iberoamericana de Automática e Informática Industrial* (En revisión).
- [6] Roca, Lidia, Bonilla, Javier, Berenguel, Manuel, González, Lucía y R. Rocha, Alberto (2016): “Control Strategies Applied in HYSOL Demonstrator: a Simulation-Based Evaluation”. The 24th Mediterranean Conference on Control and Automation (MED), Athens, Greece, June 21-24.
- [7] Lajara Vizcaíno, José Rafael y José Pelegrí Sebastià (2011): *LabVIEW: Entorno de programación*, 2.ª ed. Barcelona: Marcombo.
- [8] Modelisar Consortium (2014). Functional Mock-up Interface (FMI) for Model Exchange and Co-Simulation - Version 2.0. [En línea]. Disponible: <https://www.fmi-standard.org>. [Último acceso: 13 de mayo de 2016].
- [9] National Instruments (2016). [En línea]. Disponible: <http://www.ni.com/es-es.html>. [Último acceso: 25 de julio de 2016].
- [10] Coughanowr, Donald R. y LeBlanc, Steven E. (2008): *Process Systems Analysis and Control*, 3.ª ed. McGraw-Hill Education.
- [11] Ogata, Katsuhiko (2010): *Ingeniería de control moderna*, 5.ª ed. Pearson.
- [12] Ogunnaike, Babatunde A. y Ray, W. Harmon (1994): *Process Dynamics, Modeling, and Control*, 1.ª ed. Oxford University Press.

Desarrollo de una herramienta software para simulación y control del demostrador de una planta termosolar híbrida de energías renovables

Gustavo José Martín de Dios

En el ámbito de la ingeniería, una de las disciplinas más importantes es la de simular el comportamiento de futuras instalaciones con el fin de comprobar y analizar diferentes aspectos. Dentro de ese contexto, en este trabajo de fin de grado se ha desarrollado una herramienta software cuyo objetivo es estudiar y simular comportamientos de un demostrador de una planta termosolar híbrida de energías renovables.

