

Simulación de sistemas con *Simulink*

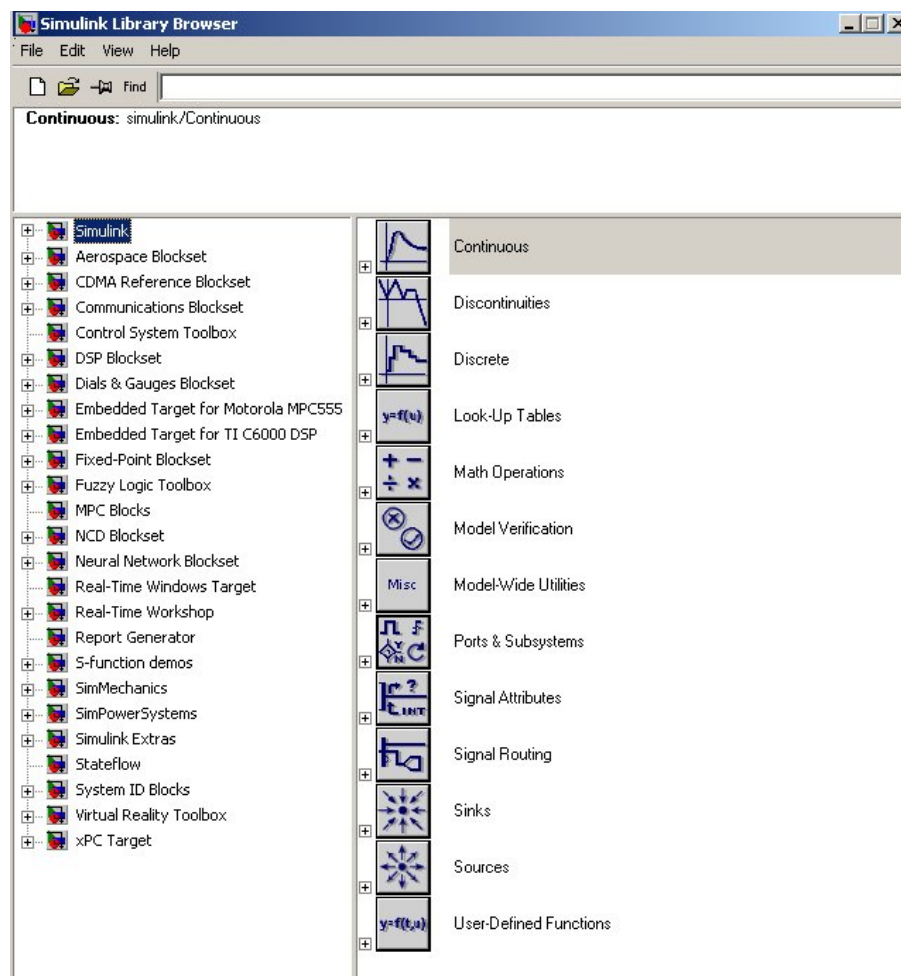
1.1	INTRODUCCIÓN A SIMULINK	2
1.1.1	CREACIÓN DE UN MODELO	3
1.2	SIMULACIÓN DE UN SISTEMA CONTINUO	4
1.3	SIMULACIÓN DE UN SISTEMA DISCRETO	5
1.4	<i>SIMULINK</i> EN TIEMPO REAL	6
1.5	DESCRIPCIÓN DE LOS EQUIPOS UTILIZADOS	6
1.5.1	SISTEMA TÉRMICO	6
1.5.2	DESCRIPCIÓN TARJETA E/S	7
1.6	ENSAYO DEL SISTEMA TÉRMICO EN BUCLE ABIERTO	7
1.6.1	EJERCICIO 1	8
1.7	ENSAYO DEL SISTEMA TÉRMICO EN BUCLE CERRADO.....	8
1.7.1	EJERCICIO 2	9
1.8	IMPLEMENTACIÓN DE UN REGULADOR CONTINUO	11
1.8.1	EJERCICIO 3	11

1.1 Introducción a Simulink


Simulink es la interfase gráfica de simulación de *MATLAB*. Permite el análisis y estudio de sistemas (de distintas disciplinas de la técnica) mediante la simulación de los modelos construidos en *Simulink*. La creación de estos modelos es sencilla e intuitiva, ya que se forman mediante la interconexión gráfica de distintos bloques. Dentro del editor de modelos de *Simulink* se insertan bloques, se conectan y se parametrizan para su posterior simulación.

En *Simulink* es posible crear y simular modelos mecánicos, eléctricos, electrónicos, aeronáuticos, etc. gracias a la gran variedad de bloques (*blocksets*) de los que dispone. Estos conjuntos de bloques se encuentran agrupados en la *Simulink library browser*, que se despliega al ejecutar *Simulink*, y presenta el aspecto de la figura

La librería principal de bloques se encuentra bajo la carpeta llamada *Simulink* y en ella aparecen los bloques agrupados en las siguientes categorías: continuos, no lineales (*Discontinuities*), discretos, tablas, operaciones matemáticas, verificación de modelos, puertos y subsistemas, señales, dispositivos de salida (*Sinks*), generadores (*Sources*).



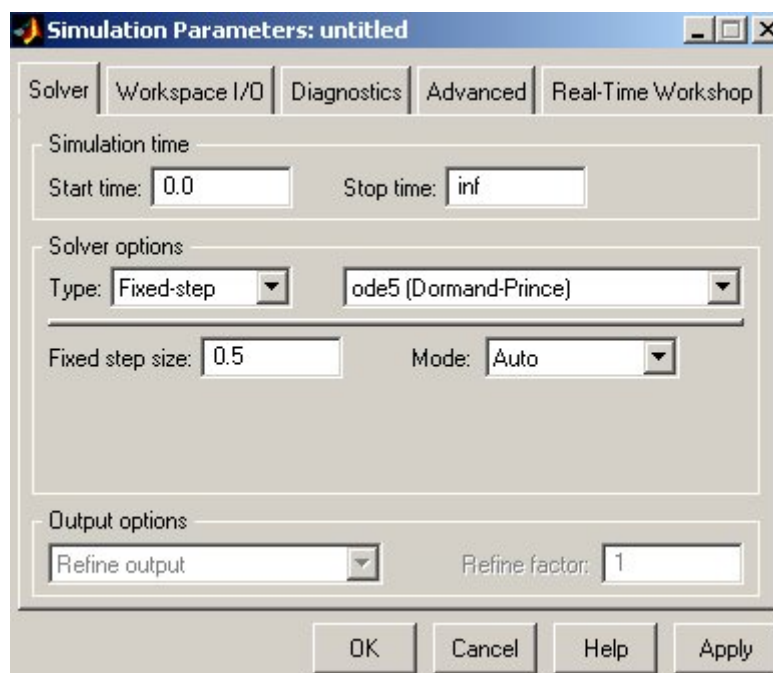
1.1.1 Creación de un modelo

Para ejecutar *Simulink* pulsar sobre el botón  en el menú principal, o bien teclear **Simulink** desde el editor de comandos de *matlab* (aparecerá una nueva ventana con todas las librerías disponibles). A continuación, y desde el menú principal, se seleccionará *File*→*New*→*Model* que abrirá la ventana de edición donde se creará el modelo *Simulink* para su posterior simulación. Los distintos bloques del modelo a crear se han de seleccionar primero en las correspondientes librerías, después arrastrar y soltar en la ventana de edición; por último, interconectar entre si.

Pulsando dos veces sobre cada bloque se despliega la ventana de parámetros correspondiente a dicho bloque; cada campo que aparece en ella se rellenará con los datos requeridos para el modelo que se va a simular.

Una vez creado el modelo y parametrizados todos sus bloques se procede a la simulación seleccionando en el menú *Simulation*→*Start*. Para detener la simulación seleccionar *Simulation*→*Stop*.

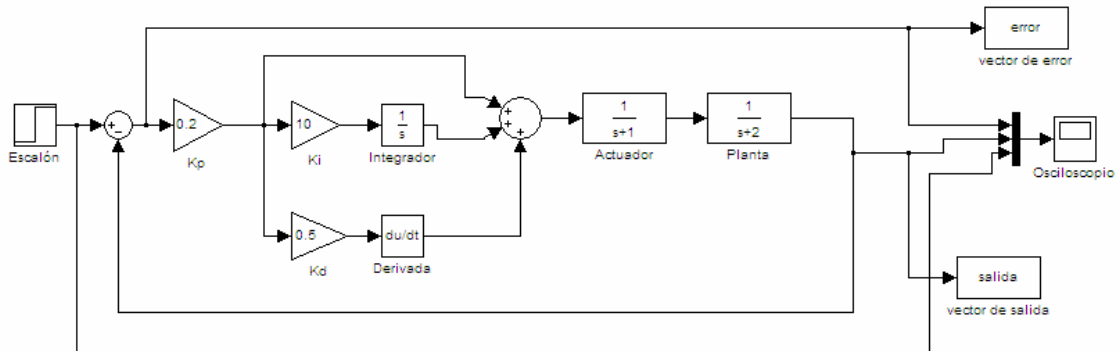
Los tiempos de la simulación del modelo, tiempo de inicio (*start time*) y tiempo de parada (*stop time*), se especifican en la ventana mostrada abajo a la que se accede desde *Simulation parameters*.



Ajuste Parámetros de la Simulación

1.2 Simulación de un sistema continuo

- Construir el diagrama de bloques de la figura adjunta:



Los bloques que necesitas están en:

Escalón	→ Librería <i>Sources</i> : Step
Integrador, Derivada	→ Librería <i>Continuous</i>
Actuador, Planta	→ Librería <i>Continuous</i> : Transfer Fcn
Kp, Ki, Kd	→ Librería <i>Math Operations</i> : Gain
Sumadores	→ Librería <i>Math Operations</i> : Sum
Osciloscopio	→ Librería <i>Sinks</i> : Scope
Mux	→ Librería <i>Signal Routing</i>
Vectores	→ Librería <i>Sinks</i> : To workspace

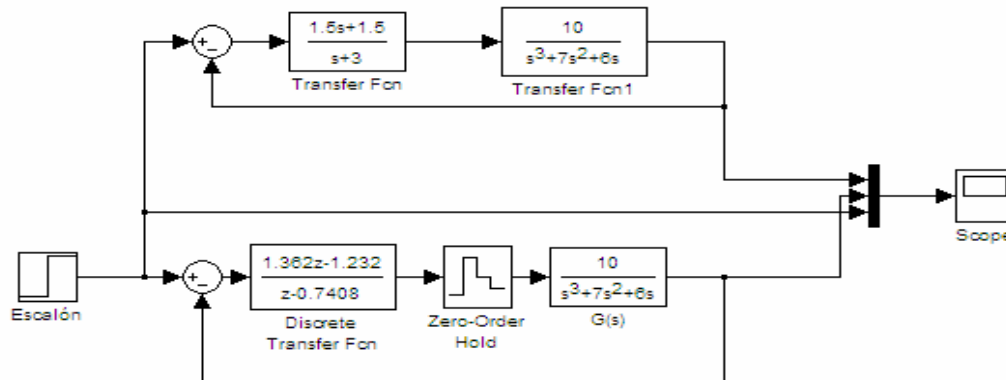
Parametrización de bloques:

Escalón, final value=1, step time=0
 Kp, Ki, Kd, los valores de la figura.
 Integrador y derivada sin ajuste de parámetros.
 Actuador y Planta, los valores de la figura.
 Mux, 3 entradas.
 Vectores, nombre y tipo array.
 Sumadores, signos.

- Simular y visualizar las respuestas del osciloscopio.
- Obtener los valores de:
 - Valor final y_{∞} :
 - Pico sobreoscilación M_p :
 - Tiempo pico t_p :
 - Tiempo establecimiento ($\pm 5\%$) t_s :
 - Error en régimen permanente e_{∞} :
- Consultar nuevas variables creadas en el editor de comandos de *Matlab* y representar gráficamente los valores del error y de la salida en función del tiempo.

1.3 Simulación de un sistema discreto

- Construir el diagrama de bloques de la figura adjunta:



Los bloques que necesitas están en:

Escalón	→ Librería <i>Sources</i> : Step
Transfer Fcn	→ Librería <i>Continuous</i>
Discrete Transfer F.	→ Librería <i>Discrete</i>
Zero-Order Hold	→ Librería <i>Discrete</i>
Sum	→ Librería <i>Math Operations</i>
Scope	→ Librería <i>Sinks</i>
Mux	→ Librería <i>Signal Routing</i>

Parametrización de bloques:

Escalón, final value=1, step time=0
 Transfer Fcn., los valores de la figura.
 Discrete Transfer Fcn., los valores de la figura con sample time=0.1
 Zero-Order Hold, sample time=0.1
 Mux, 3 entradas.
 Sumadores, signos.

- Simular y comparar las respuestas continua y discreta equivalente. Comentar resultados.
- Subir tiempo de muestreo a 0.5 s. (tanto en el bloqueador como en la función de transferencia discreta del regulador). Simular y visualizar nueva respuesta. Comentar resultados.
- Indicar el valor de T que inestabiliza el sistema.

1.4 *Simulink* en tiempo real

Simulink permite crear modelos para la realización de ensayos en tiempo real sobre un ordenador con sistema operativo Windows, permitiendo así la adquisición de señales desde un sistema real mediante una tarjeta de entradas/salidas. De esta manera, *Simulink* se convierte en una interfase gráfica que muestra los resultados de los ensayos realizados y permite realizar cambios en el funcionamiento del sistema real.

Para ello se dispone del correspondiente modelo en *Simulink* (con extensión .mdl) que incluye los bloques específicos (*S-Function*) de la tarjeta de entradas-salidas ubicada en el ordenador. A través de ellos *Simulink* se conecta con el sistema físico real permitiendo la transferencia de señales entre ambos.

El modelo de *Simulink* se ha de compilar para crear lo que se conoce como modelo de tiempo real, el programa que se va a ejecutar bajo el sistema operativo Windows. El ajuste de los parámetros de compilación para la creación de este modelo de tiempo real se efectúa dentro del menú *Simulation Parameters*. Puesto que se dispone de Windows como sistema operativo, es necesario especificar que el modelo a construir (llamado *target*) se va a ejecutar bajo dicho sistema operativo. Los ensayos en tiempo real con *Simulink* deben funcionar en modo externo en vez de modo normal.

1.5 Descripción de los equipos utilizados

1.5.1 Sistema térmico

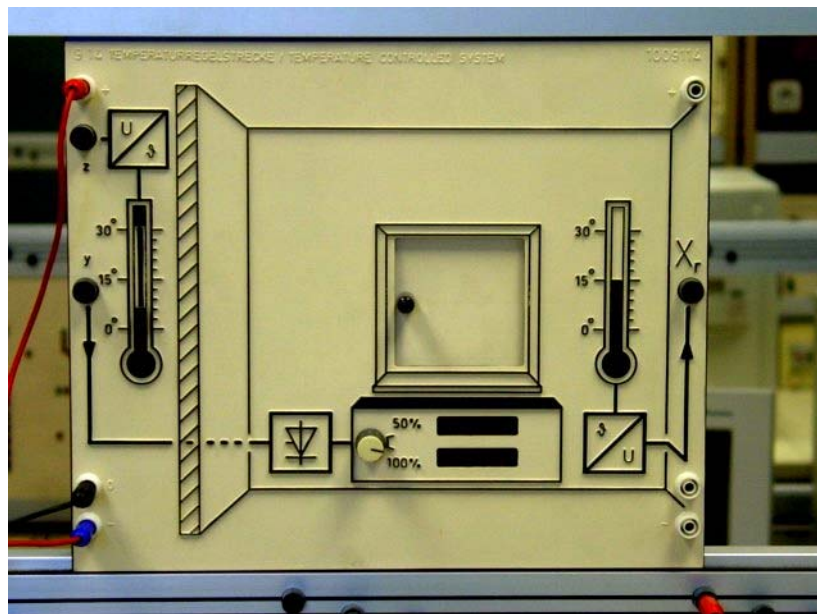


Fig. 1. Maqueta del sistema térmico.

La maqueta física de la que se dispone para la realización de la práctica corresponde a un sistema térmico cuya variable de proceso es la temperatura del interior de una habitación. El sistema está formado por las siguientes señales:

- Variable manipulada Y: entrada al sistema; suministra la señal de excitación para el sistema calefactor; es una señal analógica con rango de 0 a 10 voltios.
- Variable de proceso Xr: salida del sistema; proporciona el valor de la temperatura del interior de la habitación; es una señal analógica con rango de 0 a 10 voltios.
- Variable de proceso z: proporciona el valor de la temperatura exterior de la habitación; es una señal analógica con rango de 0 a 10 voltios (no utilizarla).

La maqueta necesita para su funcionamiento una alimentación de ± 15 voltios.

1.5.2 Descripción tarjeta E/S

El PC dispone de una tarjeta de E/S con las siguientes características:

- 16 entradas analógicas, bipolares, rango máx.: 10 v; resolución 16 bits. (Denominación: AI0, AI1, AI2,, AI14, AI15)
- 2 salidas analógicas, bipolares, rango máx.: 10 v; resolución 12 bits. (Denominación: AO1, AO2)
- 16 entradas/salidas digitales (TTL).
- Máximo muestreo de las entradas 100 kHz.

1.6 Ensayo del sistema térmico en bucle abierto

El ensayo en bucle abierto del sistema térmico permitirá observar la evolución de la respuesta de cara a obtener una función de transferencia que defina el comportamiento del mismo. Los modelos de *Simulink* para realizar ensayos en tiempo real ya están construidos y compilados por lo que no es necesaria su construcción.

El procedimiento para llevar a cabo el ensayo en bucle abierto:

1. Abrir desde *MATLAB* el fichero de *Simulink*: **termico_ca.mdl**

Aparecerá el diagrama de bloques de la figura 2

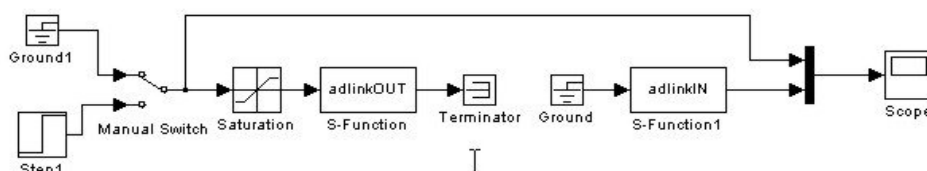


Fig. 2 Modelo para el ensayo en bucle abierto

2. **Conectar** la tarjeta del PC con la maqueta, el cable rojo (salida) a la entrada del sistema (borna Y); el cable azul (entrada) a la salida del sistema (borna Xr).

3. **Ejecutar el modelo de tiempo real** pinchando en:

Simulation → Connect to target

4. **Ejecutar el modelo Simulink** pinchando en:

Simulation → Start real time code.

5. **Activar el escalón** de 5 unidades y registrar la evolución de la temperatura del sistema asegurando que la temperatura exterior es 0° y la ventana está cerrada.

NO OLVIDAR ENCENDER LA FUENTE DE ALIMENTACIÓN DE ±15 VOLTIOS.

1.6.1 Ejercicio 1

Comprobar con MATLAB si la respuesta obtenida del sistema térmico en bucle abierto se ajusta con la siguiente función de transferencia teórica:

$$G(s) = \frac{0.9}{(1 + 7s)^3}$$

1.7 Ensayo del sistema térmico en bucle cerrado

Utilizar el modelo de la figura 3 para ensayar el sistema en bucle cerrado. En este nuevo modelo se añade un comparador para realimentar la salida del sistema a la entrada del mismo. A la salida del comparador se conecta un bloque de ganancia, representado por K, que permitirá variar el valor de la ganancia del conjunto.

Abrir en *Simulink* el modelo termico_k.mdl

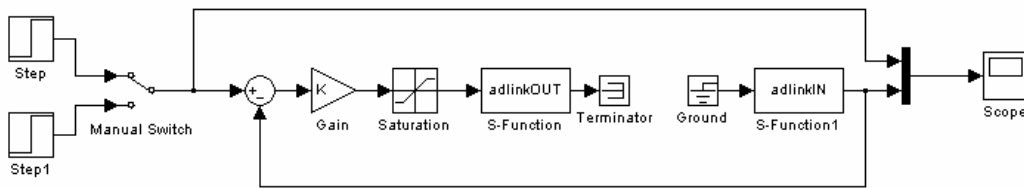


Fig. 3. Modelo para el ensayo en bucle cerrado.

1.7.1 Ejercicio 2

Completar las tablas siguientes. En los recuadros de la derecha dibujar las gráficas de la respuesta obtenida ante la entrada en escalón de amplitud 5, midiendo sobre la misma los parámetros que se indican.

Ajustar un valor de K = 1	
<p>Medir los parámetros:</p> <p>$y(\infty) =$</p> <p>$t_p =$</p> <p>$t_s _{\pm 5\%} =$</p> <p>$M_p =$</p>	<p>Dibujar las respuestas:</p>

Conclusiones sobre la respuesta del sistema:

Ajustar un valor de K = 3	
Medir los parámetros: $y(\infty) =$ $t_p =$ $t_{s \pm 5\%} =$ $M_p =$	Dibujar las respuestas:

Conclusiones sobre la respuesta del sistema:

Ajustar un valor de K = 20	
Medir los parámetros: $y(\infty) =$ $t_p =$ $t_{s \pm 5\%} =$ $M_p =$	Dibujar las respuestas:

Conclusiones sobre la respuesta del sistema:

1.8 Implementación de un regulador continuo

Utilizar el modelo *Simulink* de la figura 4, en el que se ha añadido un regulador PID como suma de las tres acciones de control (proporcional, integral, derivativa).

Abrir el modelo termico_pid.mdl

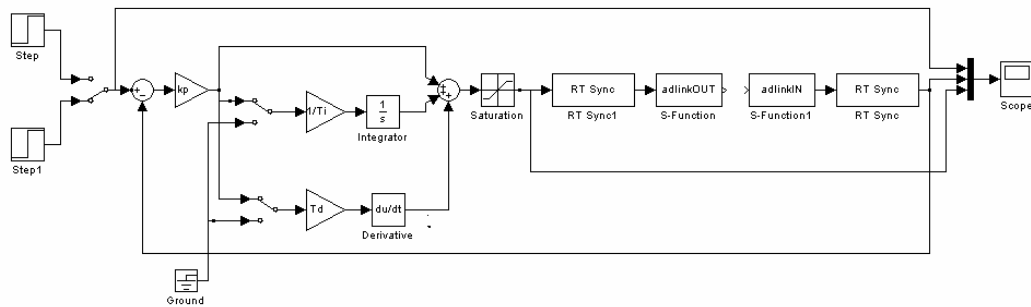


Fig. 4. Modelo para el ensayo con regulador PID continuo

1.8.1 Ejercicio 3

Implementar los siguientes reguladores, midiendo para cada caso las características indicadas y dibujando la gráfica con la respuesta real obtenida en cada caso.

Ajustar un Regulador P ($K_p = 3.5$)	
<p>Medir los parámetros:</p> <p style="margin-left: 40px;">$y(\infty) =$</p> <p style="margin-left: 40px;">$t_p =$</p> <p style="margin-left: 40px;">$M_p =$</p> <p style="margin-left: 40px;">$e_p =$</p>	<p>Dibujar las respuestas:</p>

Conclusiones sobre la respuesta del sistema:

Ajustar un Regulador PI ($K_p = 1, T_i = 15$)	
Medir los parámetros: $y(\infty) =$ $t_p =$ $M_p =$ $e_p =$	Dibujar las respuestas:

Conclusiones sobre la respuesta del sistema:

Ajustar un Regulador PID ($K_p = 1, T_i = 15, T_d = 7$)	
Medir los parámetros: $y(\infty) =$ $t_p =$ $M_p =$ $e_p =$	Dibujar las respuestas:

Conclusiones sobre la respuesta del sistema: