
Ejercicios prácticos en SIMULINK

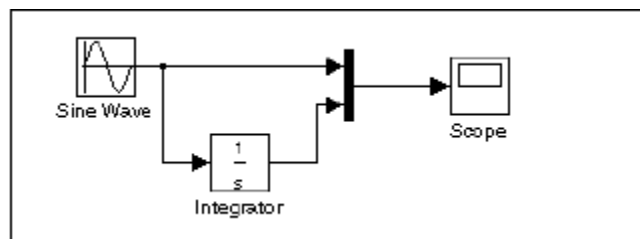
I. OBJETIVO.

Conocer los bloques más comúnmente utilizados que se encuentran dentro de la librería de Simulink.

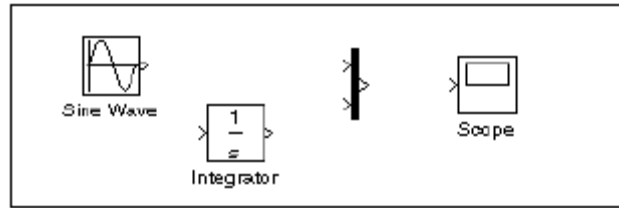


1. Un Modelo Simple

Fabricaremos un simple modelo:



- Para crear este modelo necesitaremos copiar bloques desde las siguientes librerías:
 - Librería **Sources** (el bloque **Sine Wave**)
 - Librería **Sinks** (el bloque **Scope**)
 - Librería **Continuous** (el bloque **Integrator**)
 - Librería **Signals & Systems** (el bloque **Mux**)
 - Con los bloques copiados dentro de la ventana del modelo, deberá verse algo como lo siguiente:
-



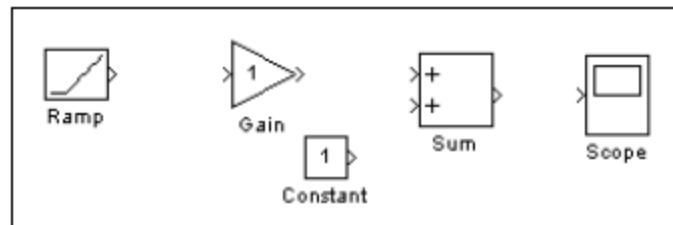
- c. Ahora conecte los bloques como se desea.
- d. Seleccione Start desde el menú Simulation y haga doble click en el bloque **Scope**. Deberá ver la respuesta del sistema debido a una entrada senoidal.

2. Conversión de Celsius a Fahrenheit

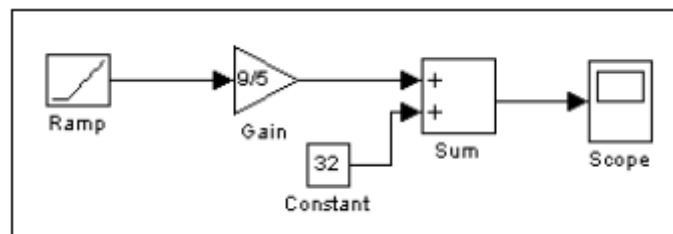
- a. Para modelar la ecuación que convierte temperatura Celsius a Fahrenheit:

$$TF = \frac{9}{5}TC + 32$$

- b. Consideremos los bloques que necesitamos:
 - Bloque **Ramp** (desde la librería **Sources**)
 - Bloque **Constant**, para definir una constante de 32 (desde la librería **Sources**)
 - Bloque **Gain**, para multiplicar la señal de entrada por 9/5 (desde la librería **Math**)
 - Bloque **Sum**, para sumar las dos cantidades (desde la librería **Math**)
 - Bloque **Scope** para visualizar la salida (desde la librería **Sinks**)
- c. Tendremos como lo siguiente:



- d. Asigna valores a los bloques **Gain** y **Constant** haciendo doble click sobre ellos e ingresando el valor apropiado.
- e. Luego conecte los bloques como se muestra.



- f. Ejecute y observe la respuesta en **Scope**.

3. Modelo de un Sistema Continuo

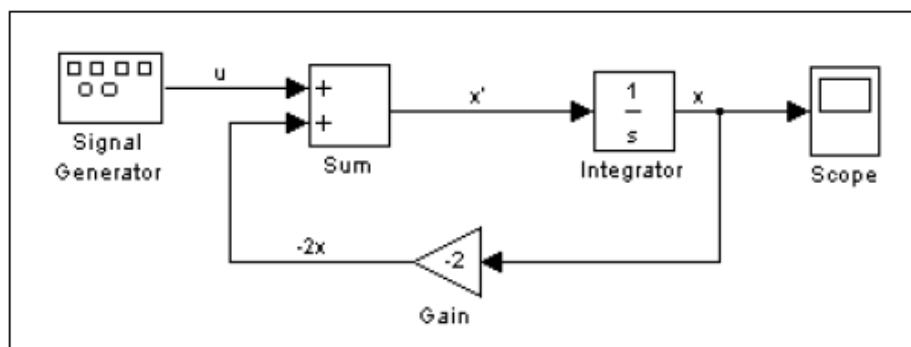
Modelo del sistema en ecuación diferencial

- a. Para modelar la ecuación diferencial

$$\frac{dx(t)}{dt} = -2x(t) + u(t)$$

donde $u(t)$ es una onda cuadrada con amplitud 1 y una frecuencia de 1rad/sec.

- b. El bloque **Integrator**, integra su entrada x' , para producir x .
c. Otros bloques usados son el bloque **Gain** y **Sum**
d. Para generar una señal cuadrada usa el bloque **Generator** y selecciona la forma *Square Wave* pero cambia las unidades por defecto a radians/sec.



- e. Observa la salida con el bloque Scope

Modelo del sistema en Función de Transferencia

- a. La ecuación modelada en este ejemplo, puede también ser expresado en función de transferencia. En el siguiente modelo usa el bloque **Transfer Fcn** el cual acepta u como entrada y x como salida.

Así el bloque **Transfer Fcn** de la ecuación:

$$\frac{dx(t)}{dt} = -2x(t) + u(t)$$

$$sX(s) = -2X(s) + U(s)$$

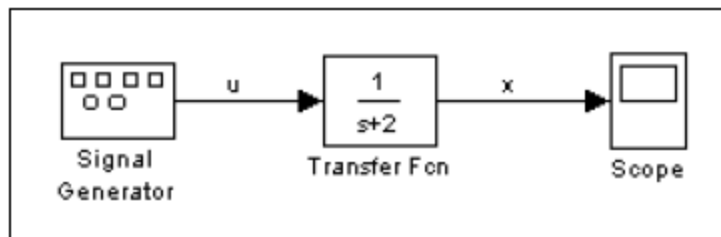
resolviendo:

$$X(s) = \frac{U(s)}{s+2}$$

o

$$\frac{X(s)}{U(s)} = \frac{1}{s+2}$$

- b. El bloque *Transfer Fcn* usa parámetros para especificar los coeficientes del numerador y denominador. En este caso el numerador es **1** y el denominador es: **s+2**. Especifica ambos términos como vectores de coeficiente en potencia decreciente de **s** sucesivamente. En este caso el numerador es 1 o [1] y el denominador es [1 2]. El modelo ahora llega a ser así:



- c. El resultado de la simulación es idéntico al modelo previo realizado

4. Usando el bloque From Workspace

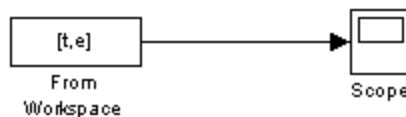
- a. desde la línea de comandos de matlab escriba:

```
» t=0:1:100;
» e=sin(t);
```

deberá tener los vectores en columnas, para ello haga:

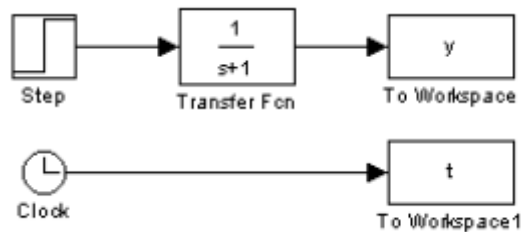
```
» t=t'
» e=e'
```

- b. Luego implemente el siguiente diagrama en simulink. Utilice el bloque *From Workspace* y dentro de sus parámetros coloque la variable **[t,e]**

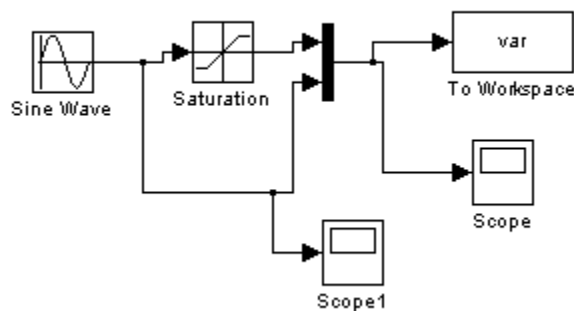


5. Usando el bloque: To Workspace

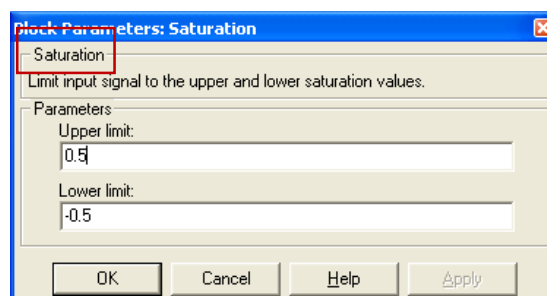
- a. El bloque **Workspace** puede ser usado para entregar trayectorias de salida al workspace de MATLAB. El modelo de abajo ilustra este uso.

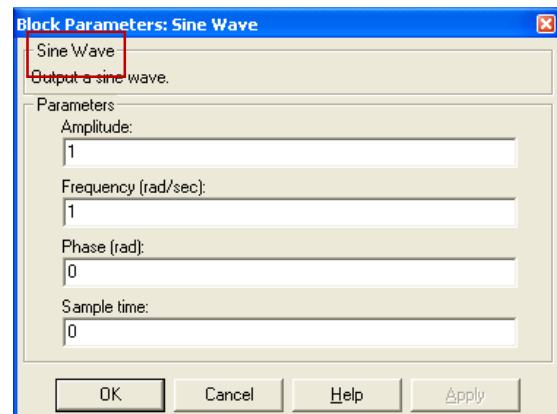
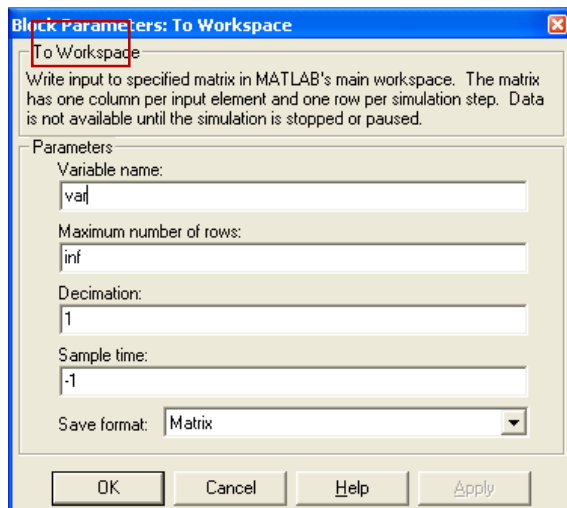


- b. La variable y y t aparece en el workspace cuando la simulación es completada. El vector $time$ es almacenado usando un bloque **Clock** enviando al bloque **To Workspace**. El vector de tiempo también puede ser adquirido ingresando un nombre de variable sobre el Workspace I/O page de los *Simulation Parameters*.
- c. Realice ahora el siguiente ejercicio similar al anterior

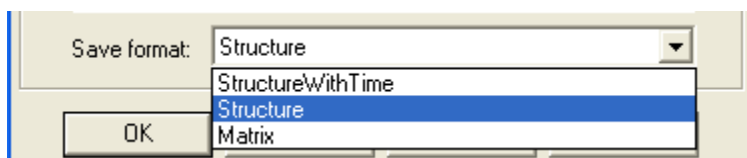


Llene los parámetros de los bloques de **Saturation**, **Sine Wave**, **To Workspace**, como se muestra a continuación:





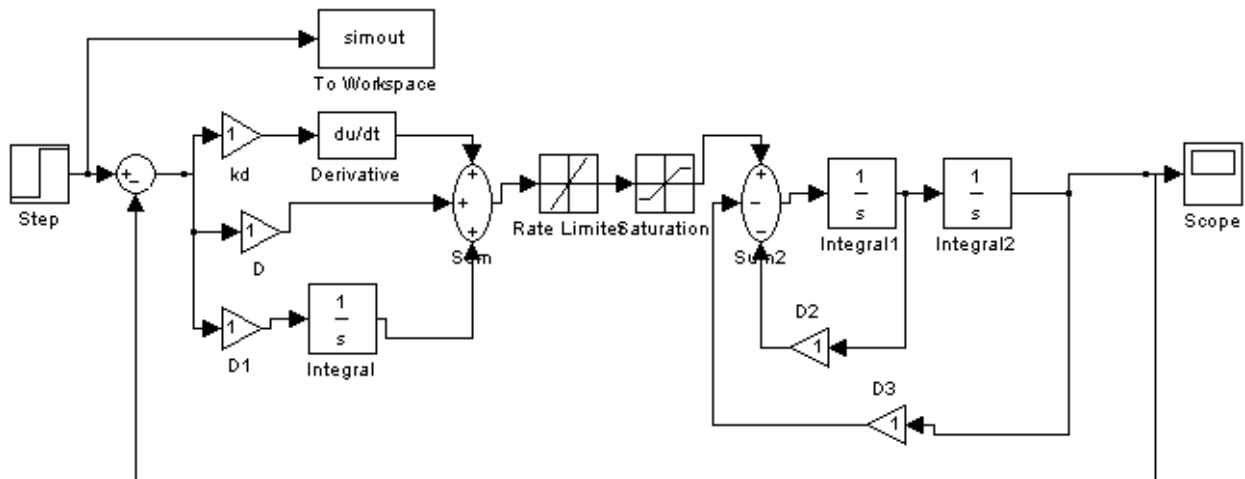
- d. Ir a la línea de comandos de matlab y escriba el comando `whos`
- e. Plotee desde aquí las salidas usando el comando `plot()`, para ello averigüe cual variable contiene dicha información.
- f. Modifique en el bloque **To Workspace** las diferentes maneras de almacenar los valores en el Workspace de matlab como se muestra en la figura:



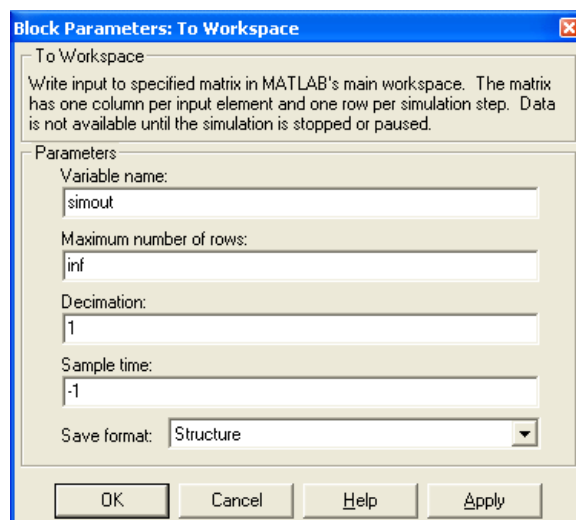
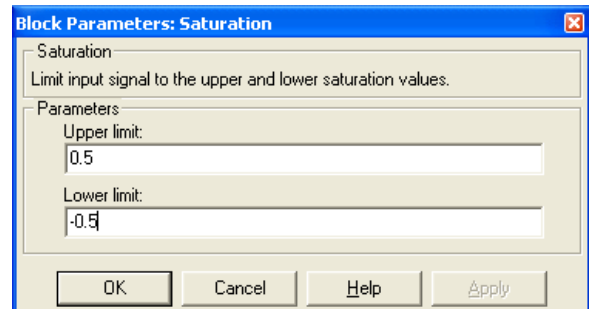
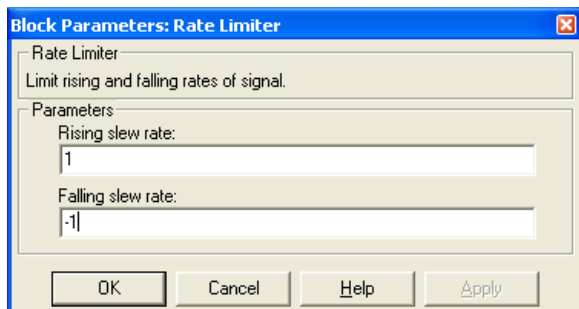
- g. Averigüe de qué manera se guarda la información en las diversas variables cuando es una: *StructureWithTime*, *Structure*, o *Matriz*. Para cada caso plotee las gráficas obtenidas, y para ello utilice el comando `plot(...)`.

6. Uso de varios subsistemas

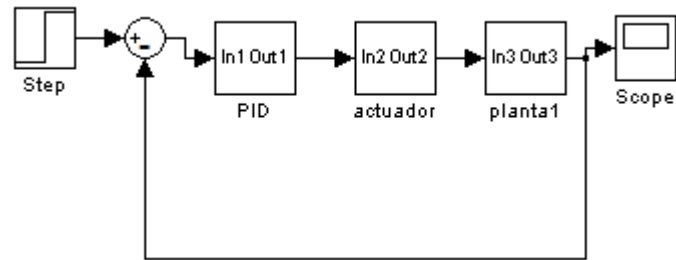
- a. Implemente el siguiente sistema de control PID en lazo cerrado.



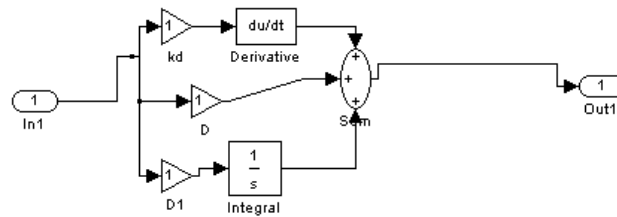
- b. Coloque los valores de los parámetros de las **no linealidades** como se muestra en las siguientes figuras



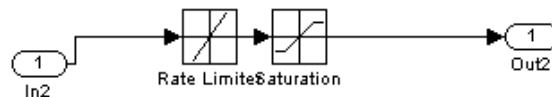
- c. Ejecute y observe la respuesta
- d. Ahora agrupe en pequeños subsistemas el modelo anterior para tenerlo de manera más ordenada, así:



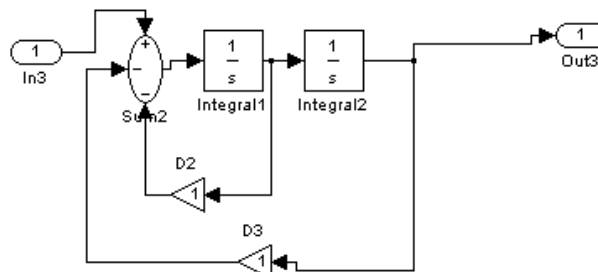
e. El primer subsistema será el siguiente:



f. El segundo subsistema será el siguiente:



g. El tercer subsistema será el siguiente:



con esto completamos el sistema final