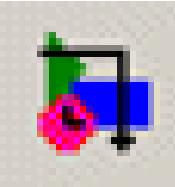


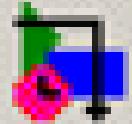
Índice

- Creación de modelos
- Solución de ecuaciones diferenciales
- Subsistemas
- Ejemplos en ingeniería



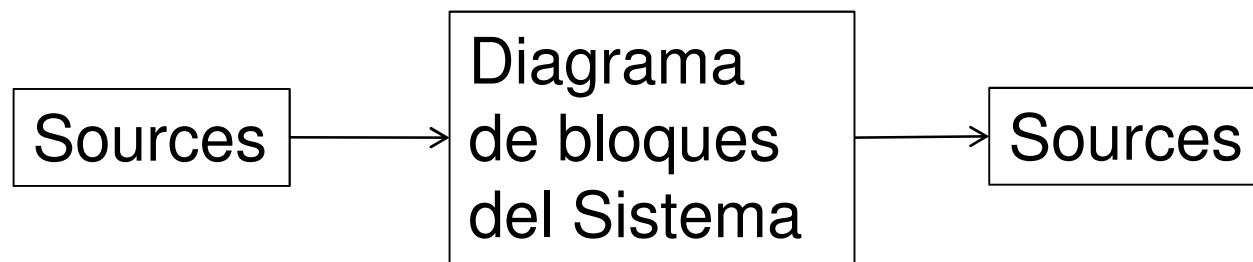
Creación de nuevo modelo en Simulink

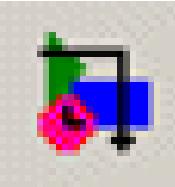
- Para editar un modelo
 - Abrir la ventana de un nuevo modelo
 - Conectar los bloques
 - Añadir bloques
 - Cambiar el tamaño de los bloques
 - Modificar etiquetas y añadir anotaciones
 - Parametrizar los bloques y la simulación
 - Ejecutar y modificar
- Guardar un modelo (formato Simulink y .m)
- Abrir un modelo desde Matlab



Elementos de un modelo Simulink

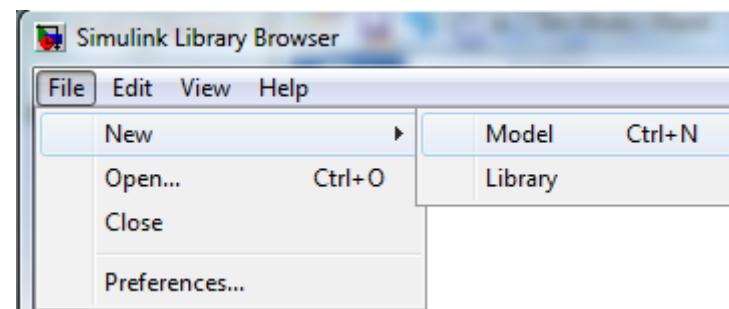
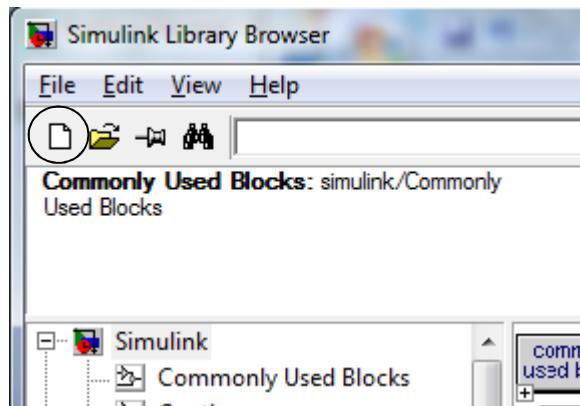
- Un modelo Simulink típico consiste de tres elementos:
 - Entradas, Sources o inputs
 - Constantes, generadores de funciones (ondas senoidales, escalón o señales creadas en Matlab)
 - Sistema modelado, representado por el diagrama de bloques
 - Salidas, Sinks u outputs
 - Gráficos, osciloscopios, ficheros

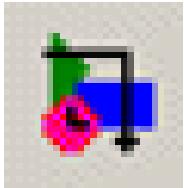




Abrir ventana de nuevo modelo

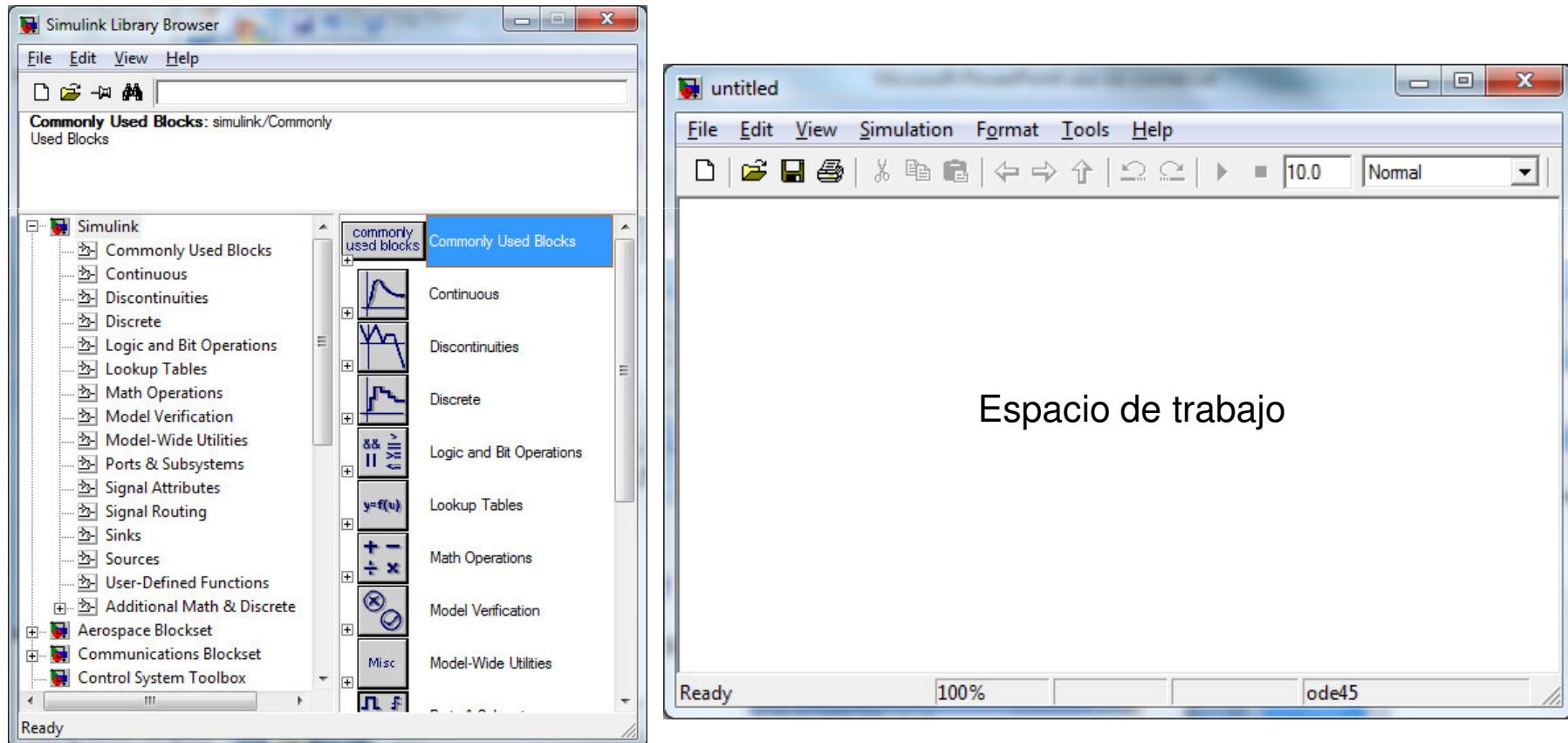
- Para crear un modelo en Simulink se pulsa sobre el icono New model  del Simulink Library Browser o se selecciona File → New → Model

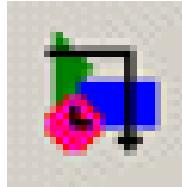




Espacio de trabajo

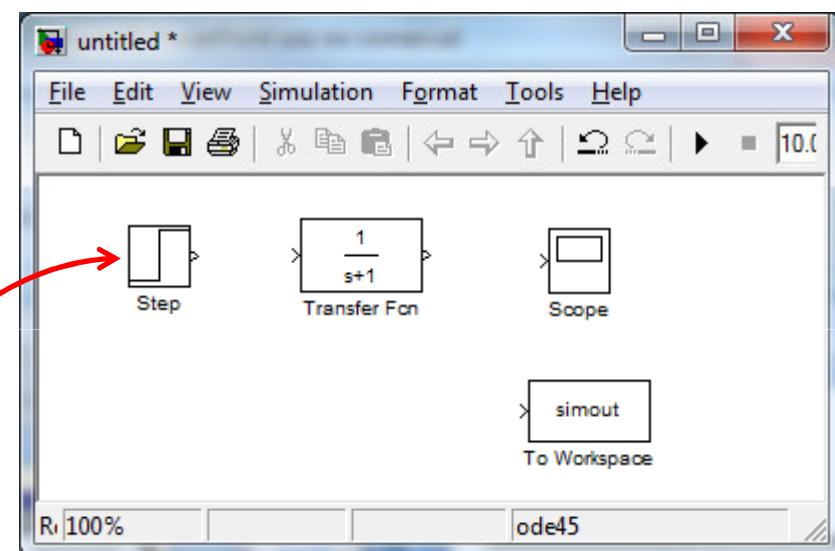
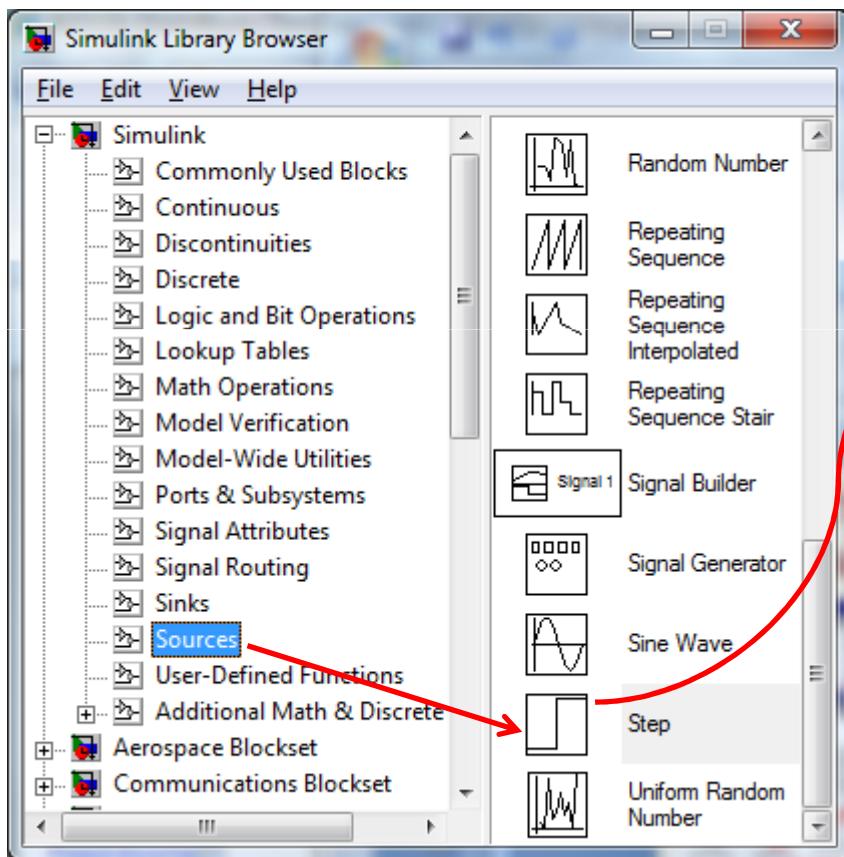
- En el espacio de trabajo se colocarán los diagramas de bloque del modelo



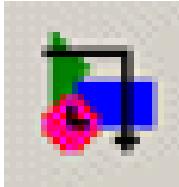


Añadir bloques

Hacer **click** sobre una librería para desplegar los bloques

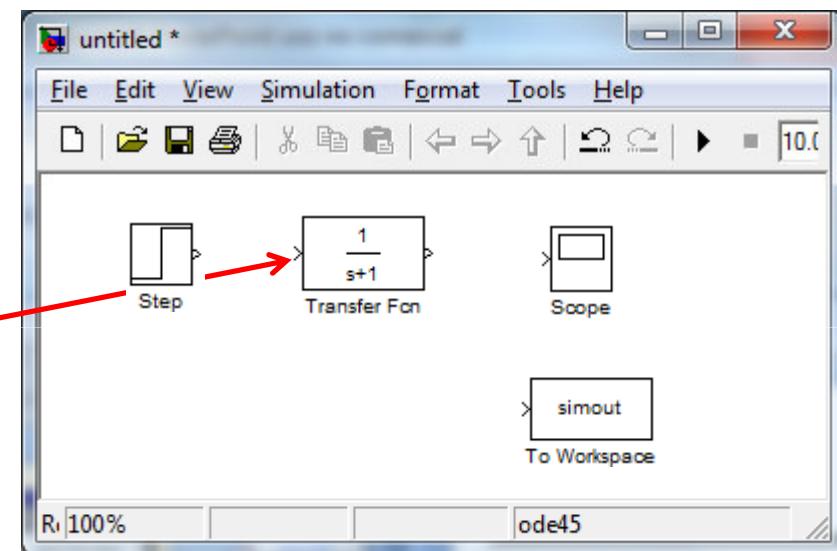
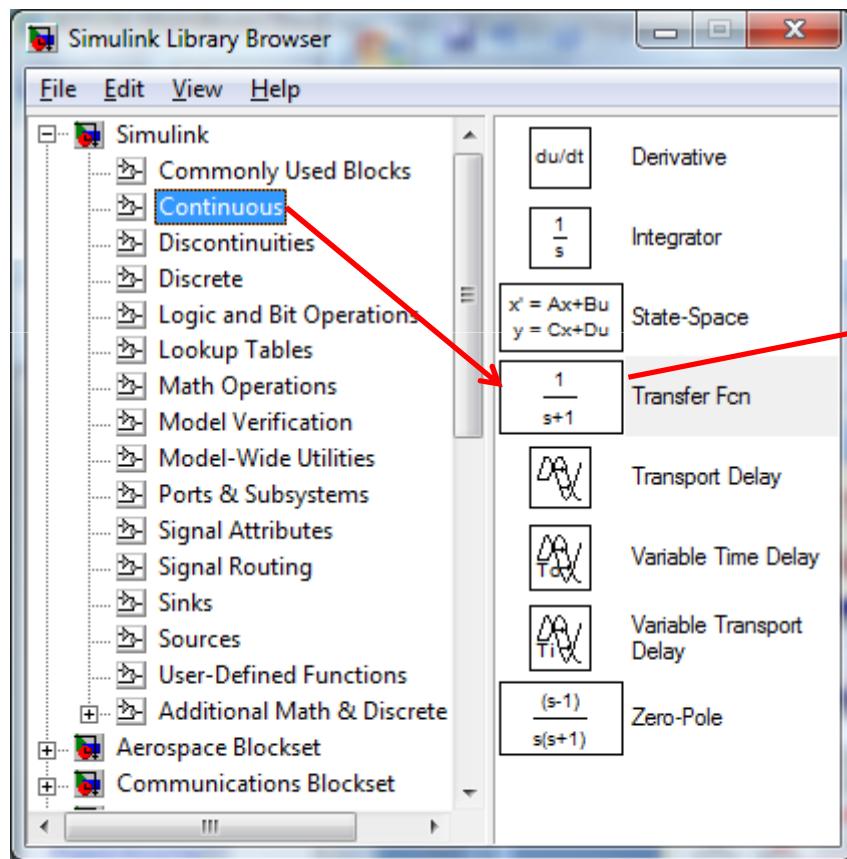


Arrastrar los bloques deseados dentro de la **ventana del modelo Simulink**

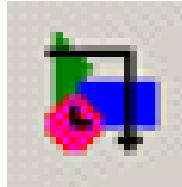


Añadir bloques

Hacer **click** sobre una librería para desplegar los bloques

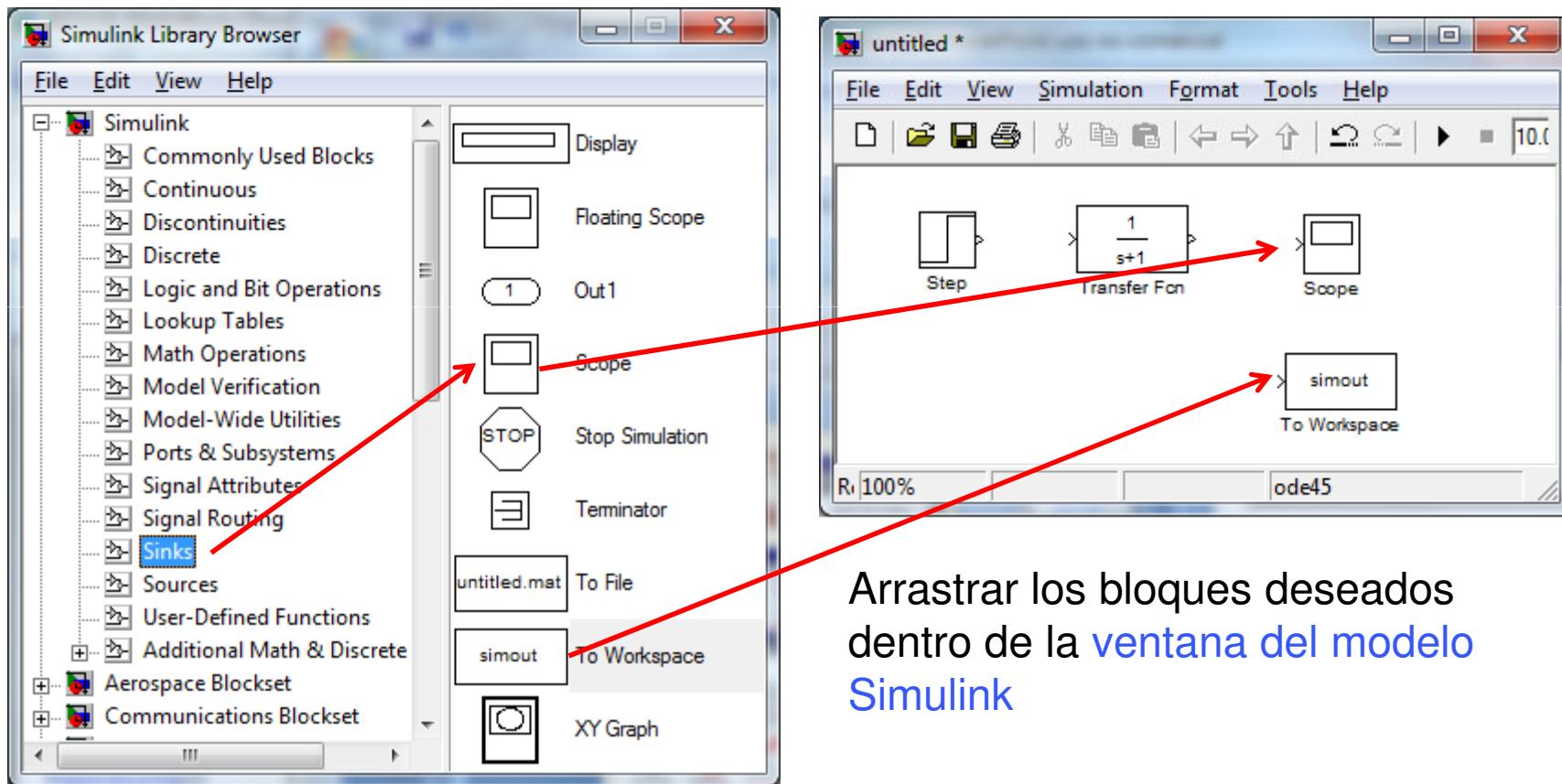


Arrastrar los bloques deseados dentro de la **ventana del modelo Simulink**

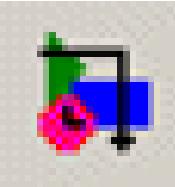


Añadir bloques

Hacer **click** sobre una librería para desplegar los bloques



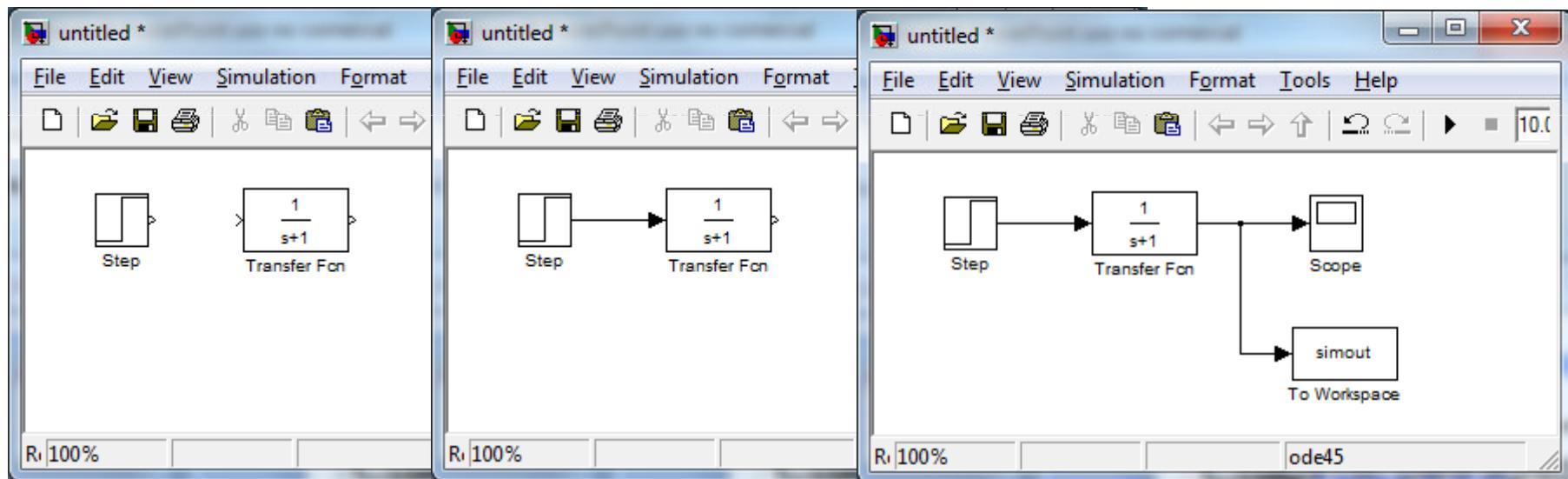
Arrastrar los bloques deseados dentro de la **ventana del modelo Simulink**

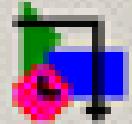


Conectar los bloques

Para añadir un conector:

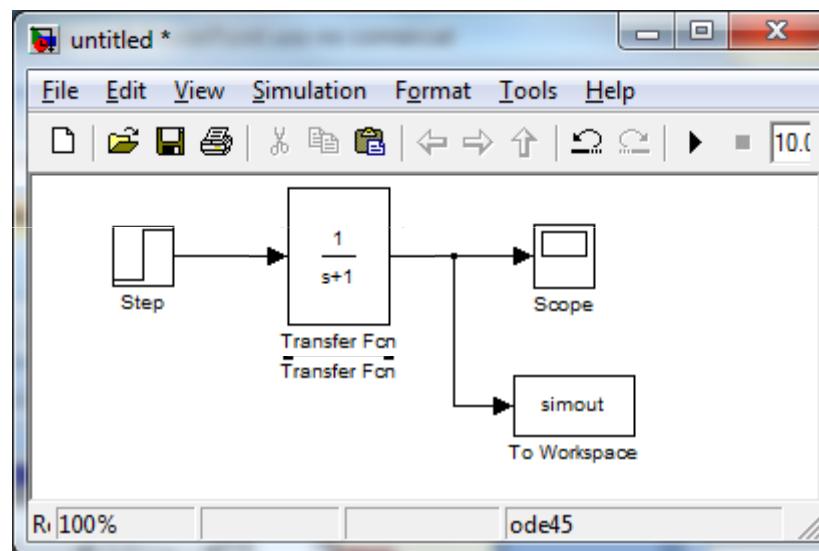
Arrastrar, pulsando el botón derecho del ratón y la tecla **Ctrl**, desde una salida, o desde una entrada, de alguno de los bloques al otro bloque



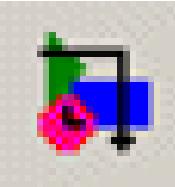


Cambiar el tamaño y/o mover los bloques

Tras seleccionar el bloque, aparecen en él los puntos, desde los cuales se puede arrastrar para cambiar el tamaño del bloque

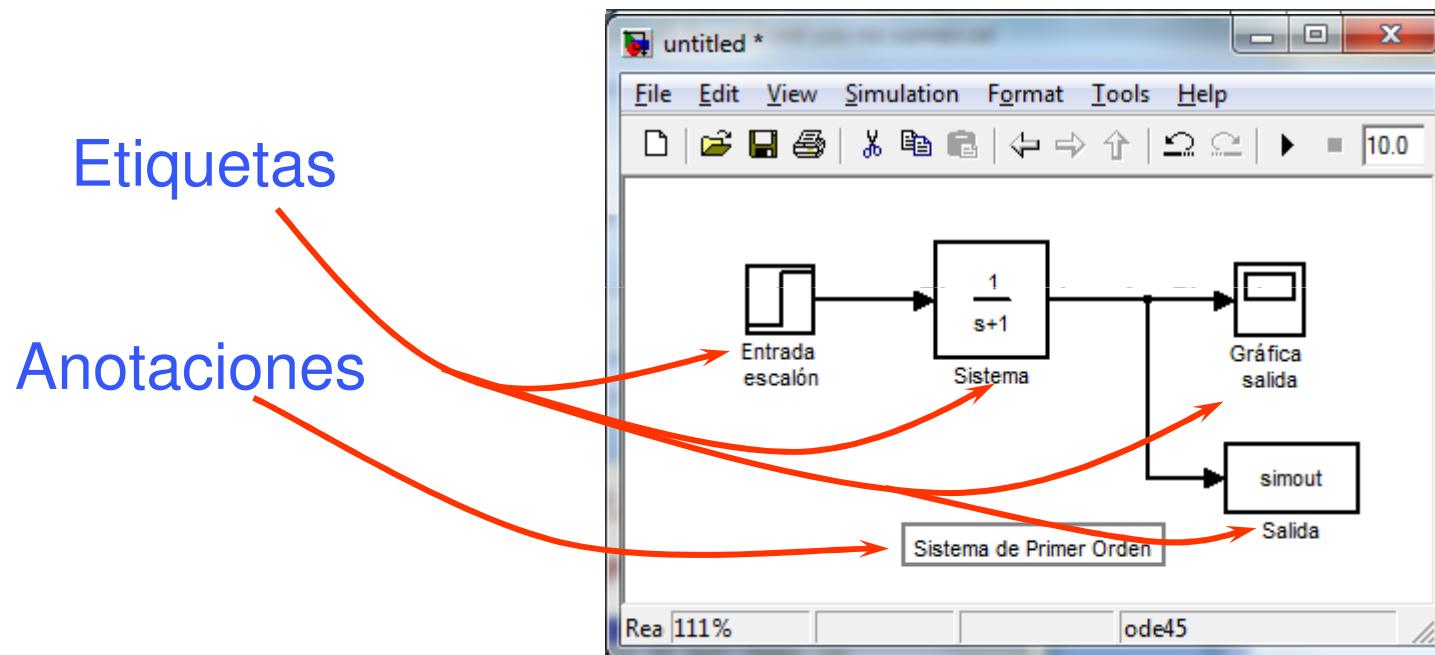


Posteriormente se pueden mover el bloque para que las líneas de conexión queden rectas

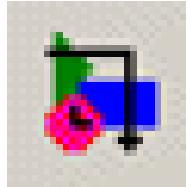


Modificar etiquetas y añadir anotaciones

Hacer click en la etiqueta y editarla



Hacer dobleclick en el fondo y escribir el texto



Parametrizar los bloques

The image shows a Simulink model window titled "untitled *". The model consists of a "Source Block Parameters: Entrada escalón" block, a "Sistema" block (containing a transfer function $\frac{1}{s+1}$), a "Gráfica salida" block, and a "simout" block. A red arrow points from the "Entrada escalón" block in the model to the "Source Block Parameters" dialog window on the right.

Source Block Parameters: Entrada escalón

Step
Output a step.

Parameters

Step time: 1

Initial value: 0

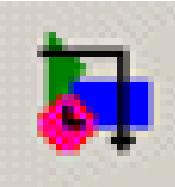
Final value: 1

Sample time: 0

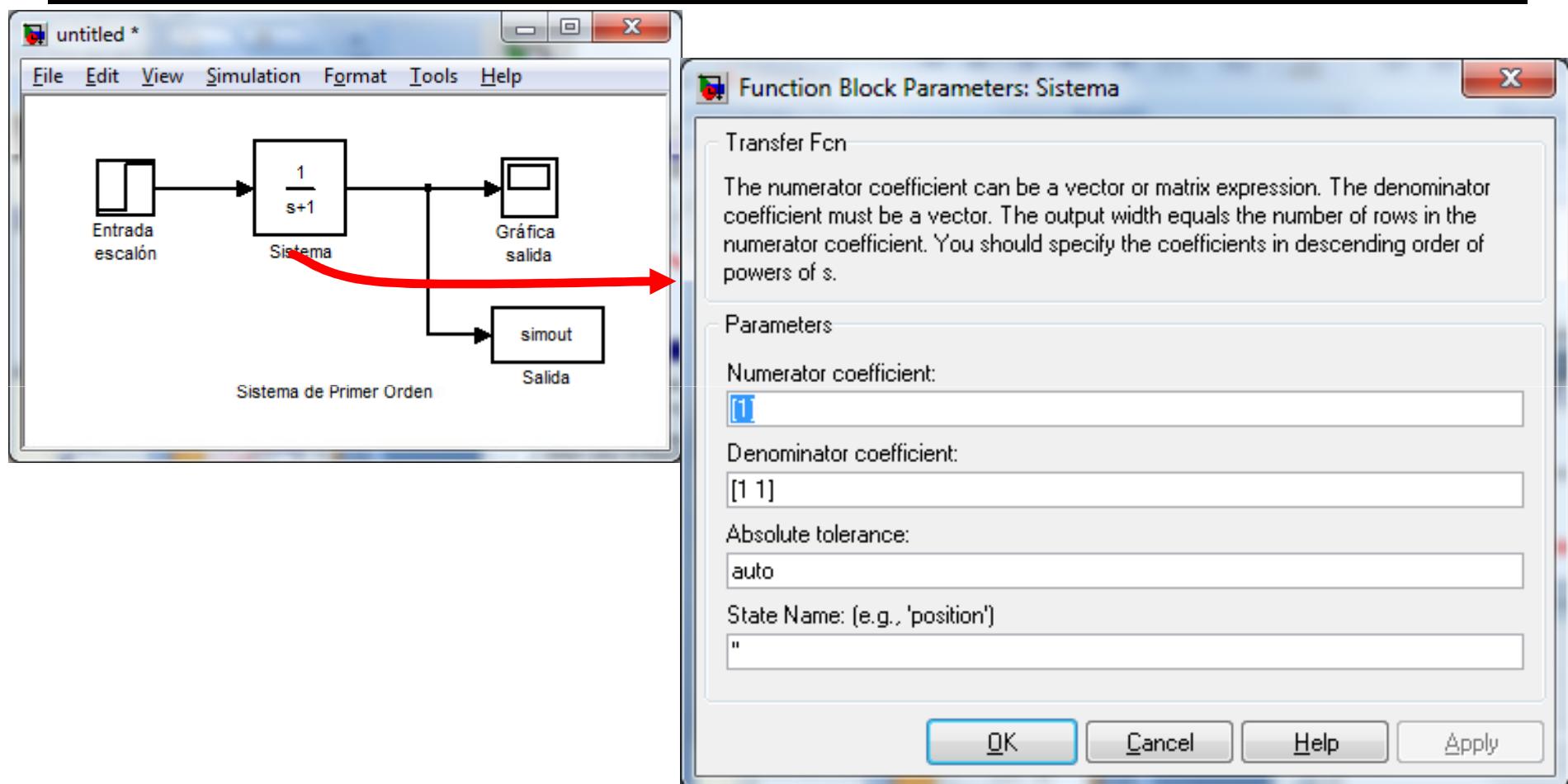
Interpret vector parameters as 1-D

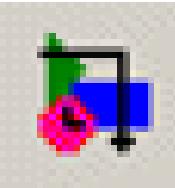
Enable zero crossing detection

OK Cancel Help

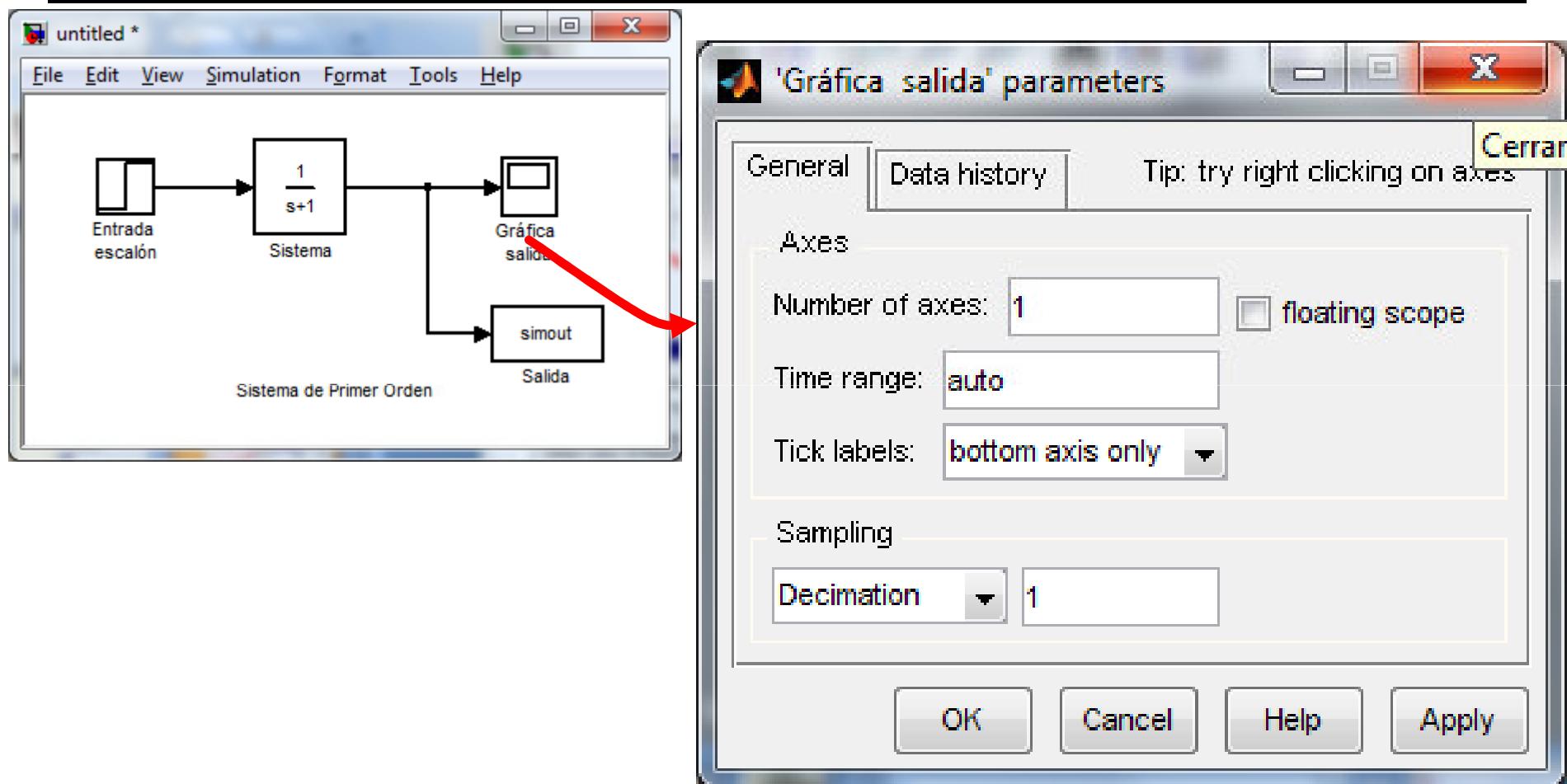


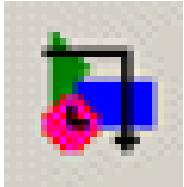
Parametrizar los bloques





Parametrizar los bloques





Parametrizar los bloques

The image shows a Simulink model window titled "untitled *". The model contains a Step input block labeled "Entrada escalón", a Transfer Function block labeled "Sistema" with the value $\frac{1}{s+1}$, a Scope block labeled "Gráfica salida", and a Sink block labeled "Salida" with the identifier "simout". A red arrow points from the "Salida" block in the model to the "Sink Block Parameters" dialog box.

Sink Block Parameters: Salida

To Workspace

Write input to specified array or structure in MATLAB's main workspace. Data is not available until the simulation is stopped or paused.

Parameters

Variable name: `simour`

Limit data points to last: `inf`

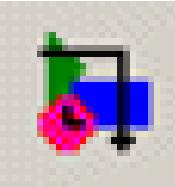
Decimation: `1`

Sample time (-1 for inherited): `-1`

Save format: Structure

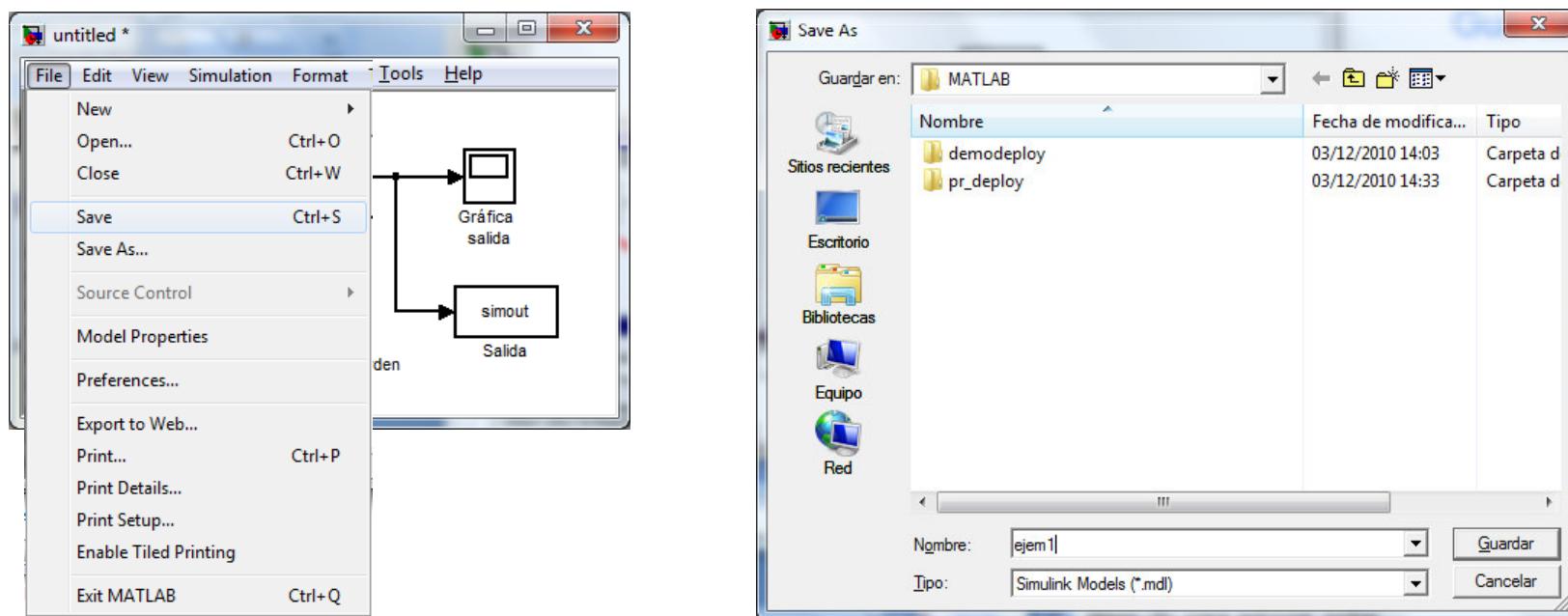
Log fixed-point data as an fi object

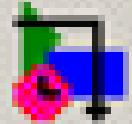
OK Cancel Help Apply



Guardar el modelo

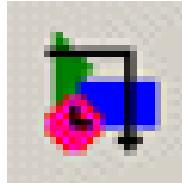
- Para guardar el modelo seleccionar File → Save
- El sufijo de los modelos Simulink es .mdl
- Desde la ventana de comandos de Matlab se puede abrir el modelo escribiendo el nombre del fichero



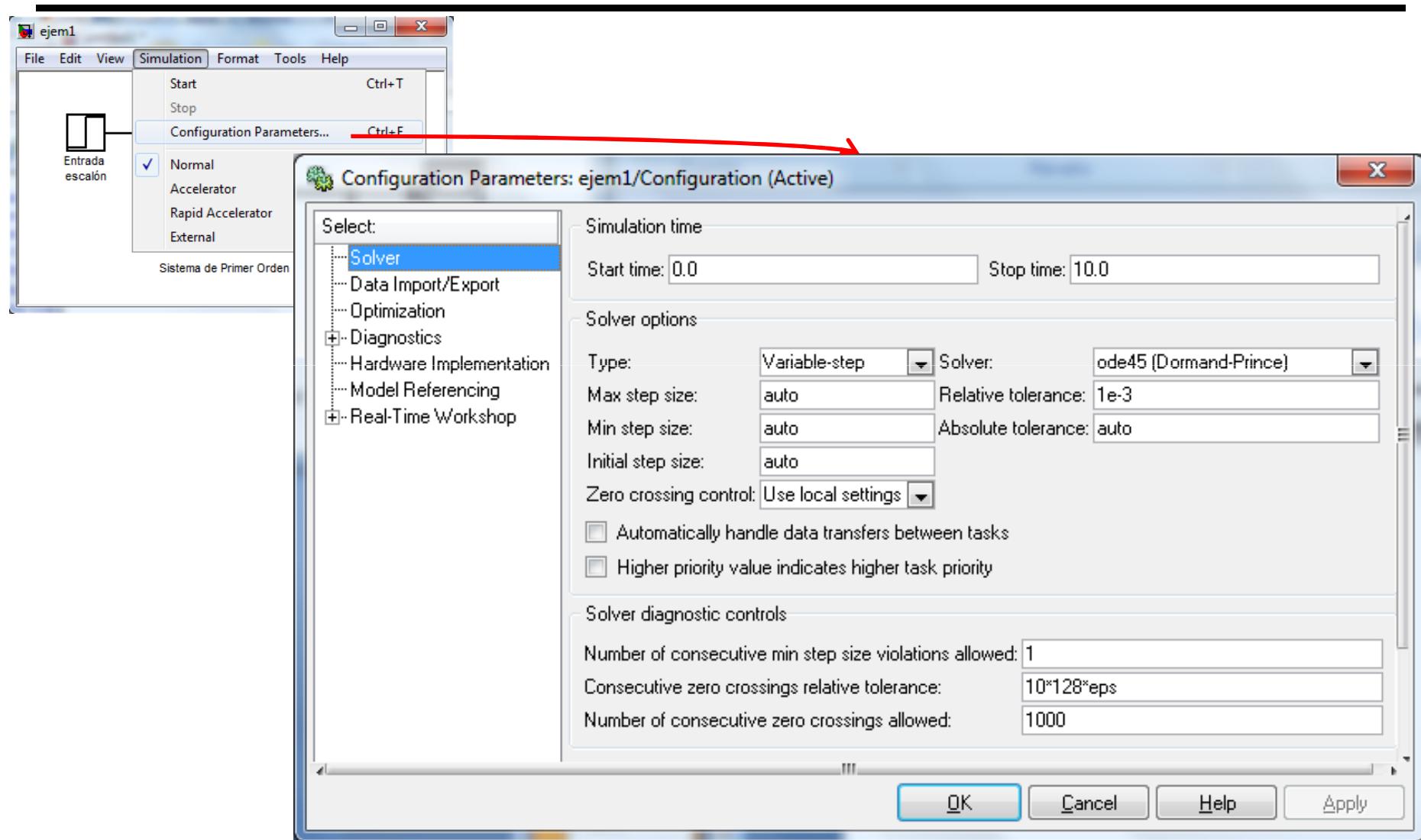


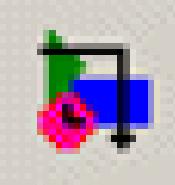
Ejecución de la simulación del modelo

- Asignar los parámetros de la simulación
- Ejecutar una simulación desde la ventana del modelo
- Poner y sacar valores en/desde los modelos
 - Utilizar en Matlab los valores obtenidos en la simulación
 - Variables definidas en Matlab y Simulink
- Simular desde la línea de comandos



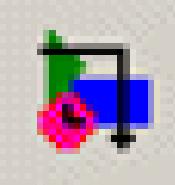
Asignar parámetros de la simulación





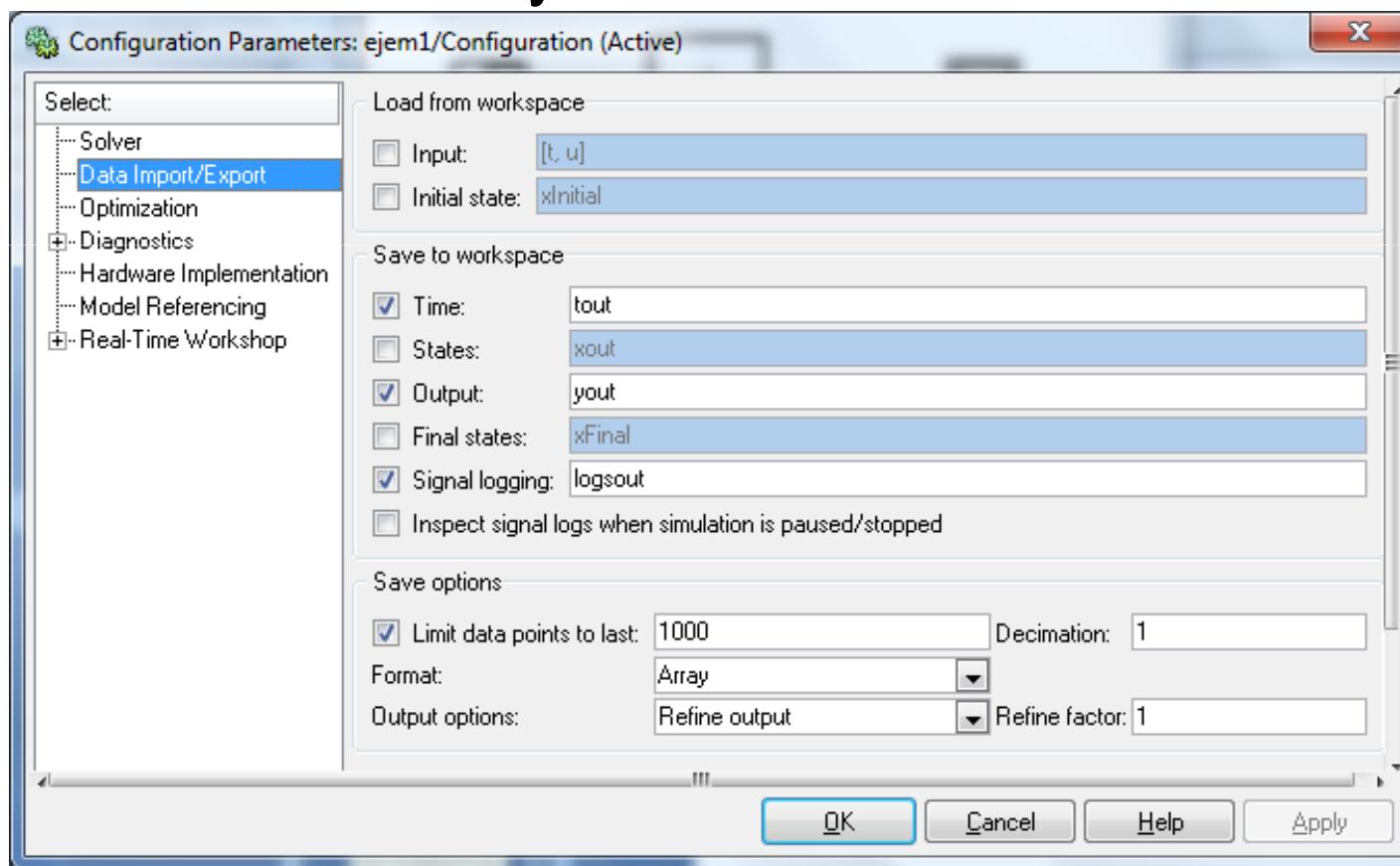
Parámetros de la simulación

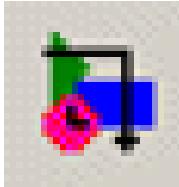
- Solvers proporcionados:
 - ode45: Método basado en Dormand - Prince , un paso Runge – Kutta y es recomendado como un primer método
 - ode23: Método basado en Bogacki – Shampine, un paso un paso Runge – Kutta y puede ser más eficiente que ode45 cuando la tolerancia es amplia
 - ode113: Este es un multipaso , de orden variable Adams – Bashforth – Moulton PECE. Es recomendable cuando la función evaluación consume tiempo y la tolerancia es poca
 - ode15s: Es un multipaso , de orden variable basado en la fórmula de diferenciación “ backward”
 - ode23: un paso basado en la fórmula de Rosembrock de orden 2.
-



Parámetros de la simulación

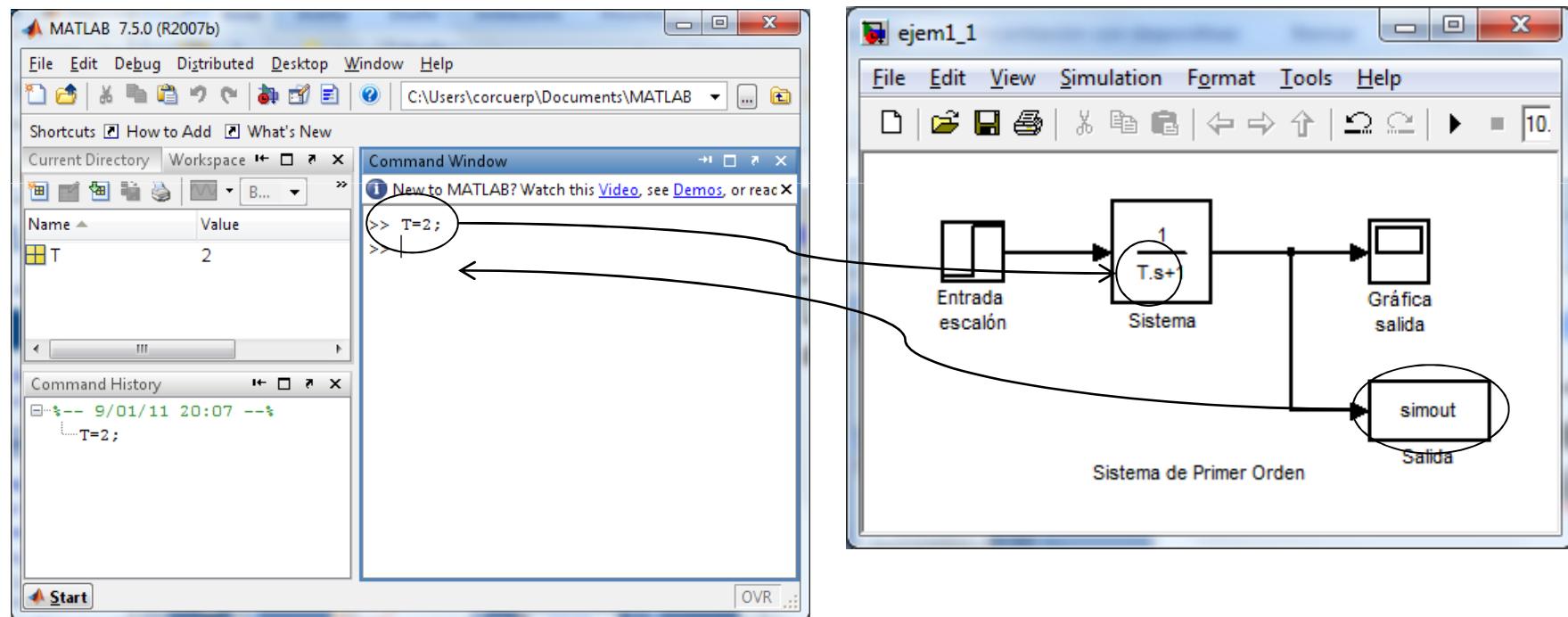
- Otros parámetros tiene que ver con la entrada/salida de datos al modelo y desde el modelo

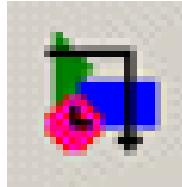




Parámetros de la simulación

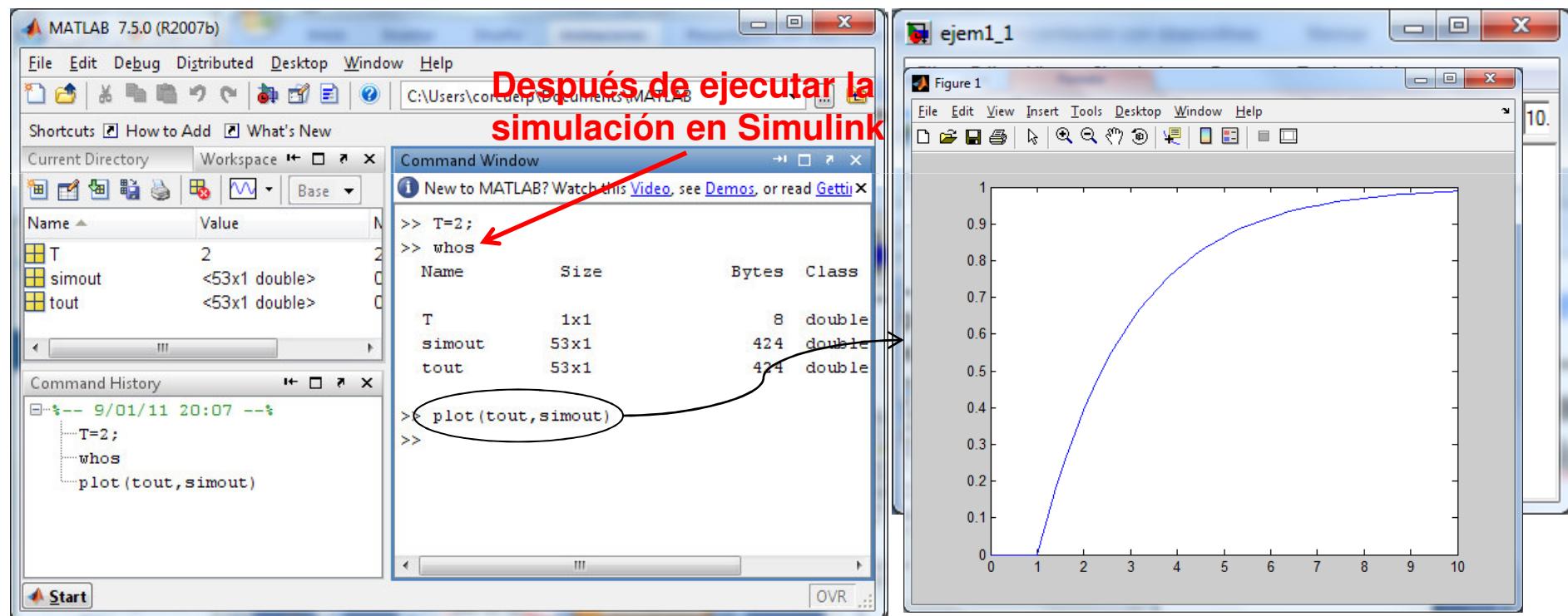
- Los parámetros y variables de los modelos se pueden acceder desde la ventana de Comandos de Matlab

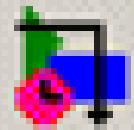




Parámetros de la simulación

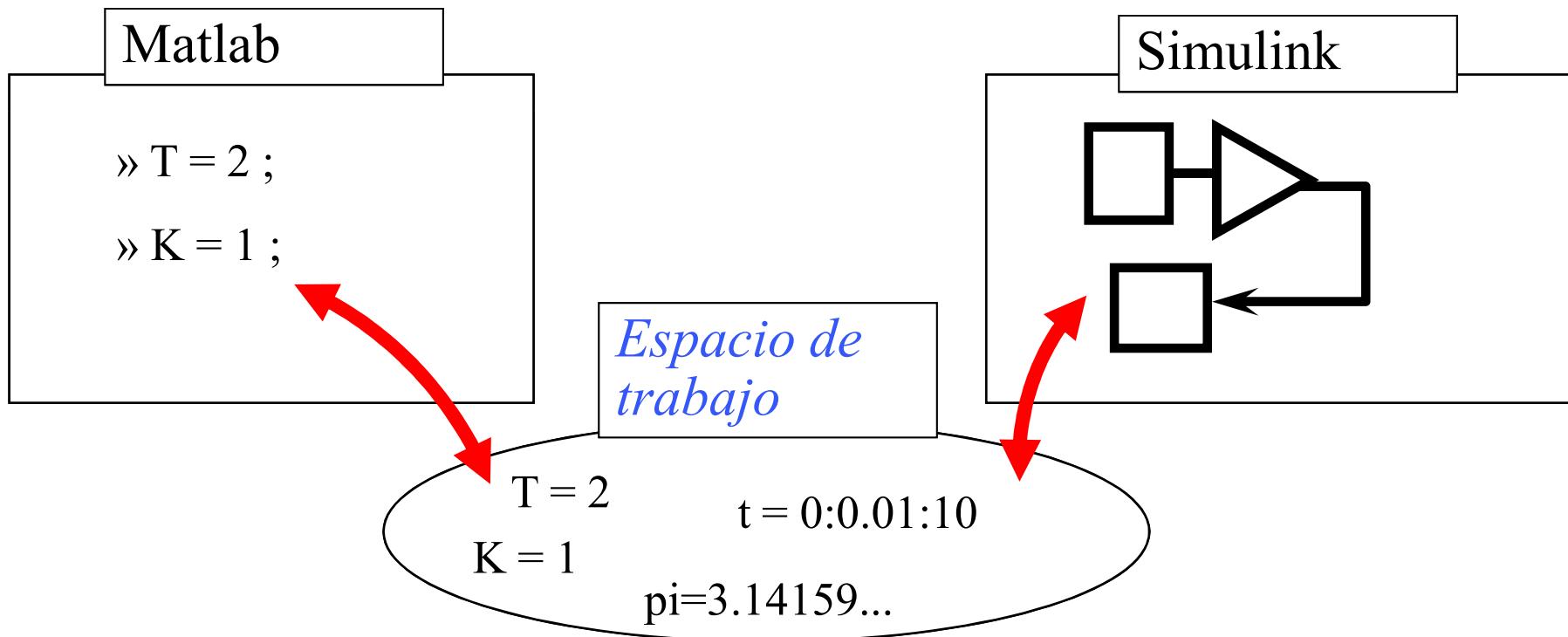
- Los parámetros y variables de los modelos se pueden acceder desde la ventana de Comandos de Matlab

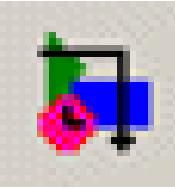




Variables definidas en Matlab y Simulink

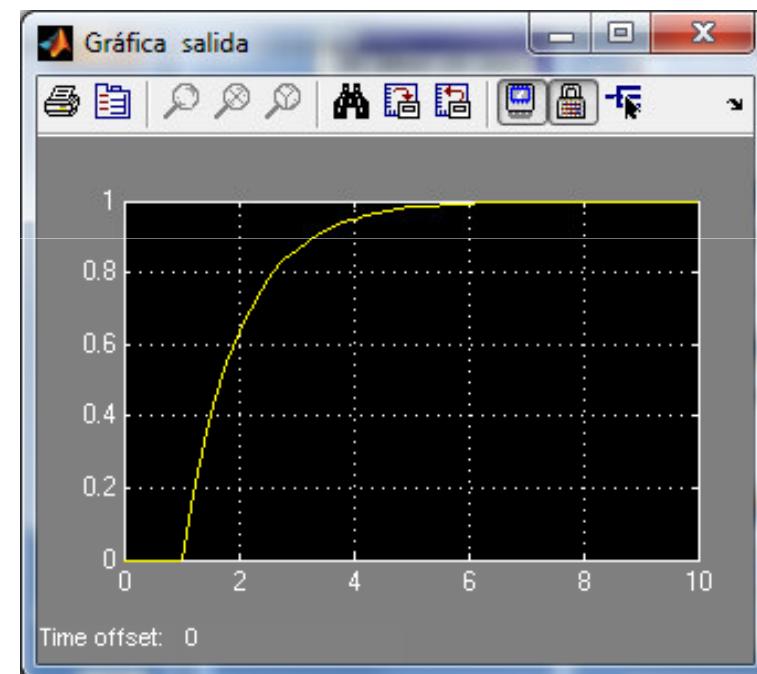
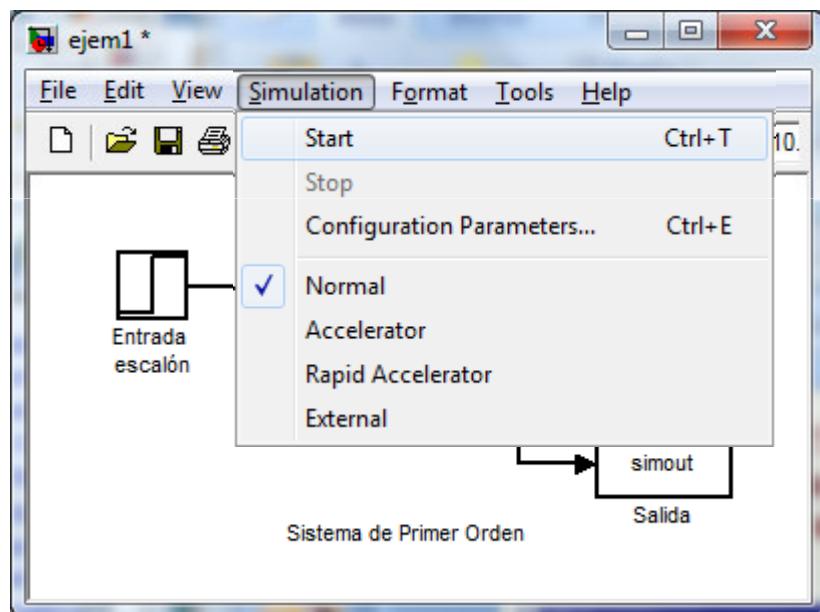
- Tanto desde la ventana de Matlab como la de Simulink se “ve” el mismo Workspace o Espacio de trabajo

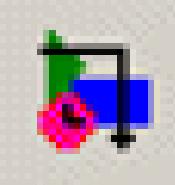




Ejecución de la simulación

- Se pulsa el ícono Start ➤ o en el menú Simulation → Start





Ejecución de la simulación

- Comando Matlab **sim**

`[t, x, y] = sim('model', Timespan, Opciones, ut)`

donde *model* es el nombre del diagrama de bloques.

Timespan especifica la salida de los puntos de tiempo

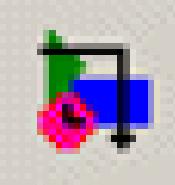
Opciones es una estructura que permite asignar los valores de los parámetros en la ventana de diálogo Simulation:Parameters

ut asigna la parte Load de la página Workspace I/O de la ventana Simulation:Parameters

Ejemplo:

> `[t, y] = sim('ejem1_1',5);`

> `plot(t, y)`



Solución de ecuaciones diferenciales que modelan Sistemas Continuos

- Modelo Simulink que resuelve la ecuación diferencial:

$$\frac{dx}{dt} = 5 \sin(4t)$$

- Condición inicial:

$$x(0) = -2.$$

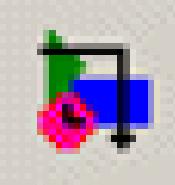
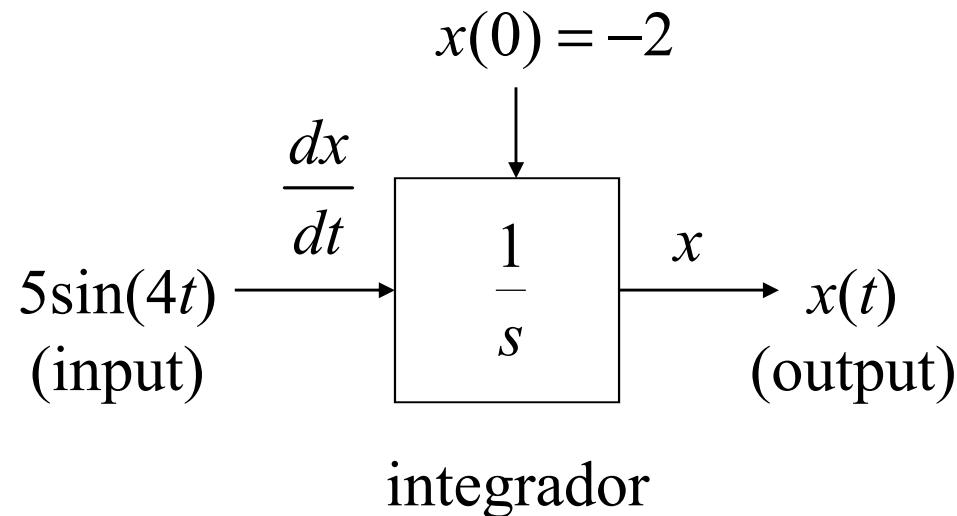
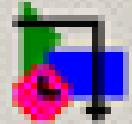


Diagrama del modelo

- Input: función $5\sin(4t)$
- Output: $x(t)$ que es la solución de la ecuación diferencial



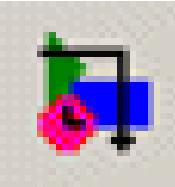
- A continuación, se construye el modelo con Simulink



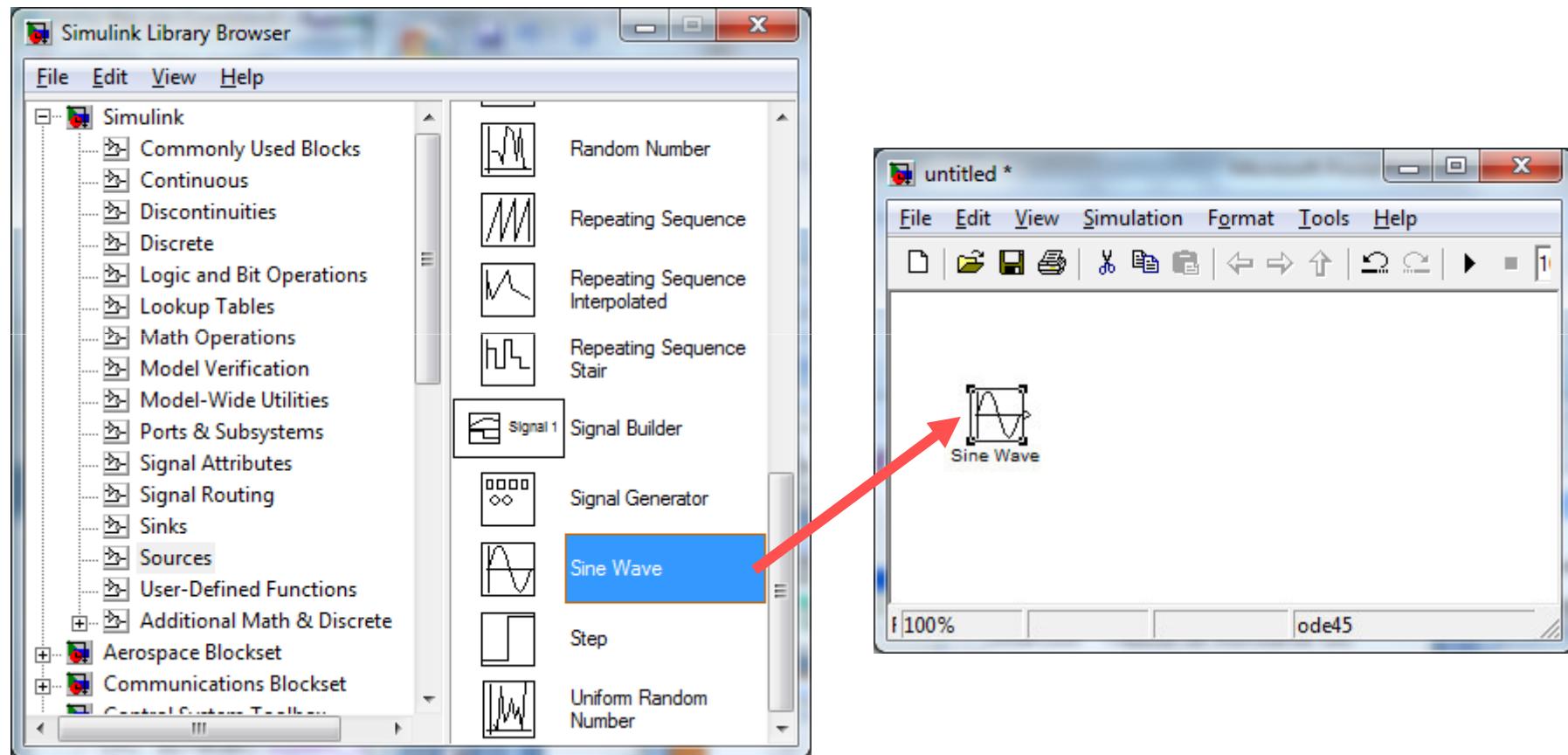
Selección de bloques para el modelo

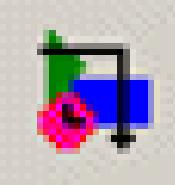
- La siguiente tabla resume el bloque y la librería donde se encuentra para ser incluido en el modelo
 - Se arrastra el bloque de la librería hasta la ventana de trabajo

Modelo	Librería	Bloque
Input	Sources	Sink
Integrador	Continuous	Integrator
Output	Sink	Scope

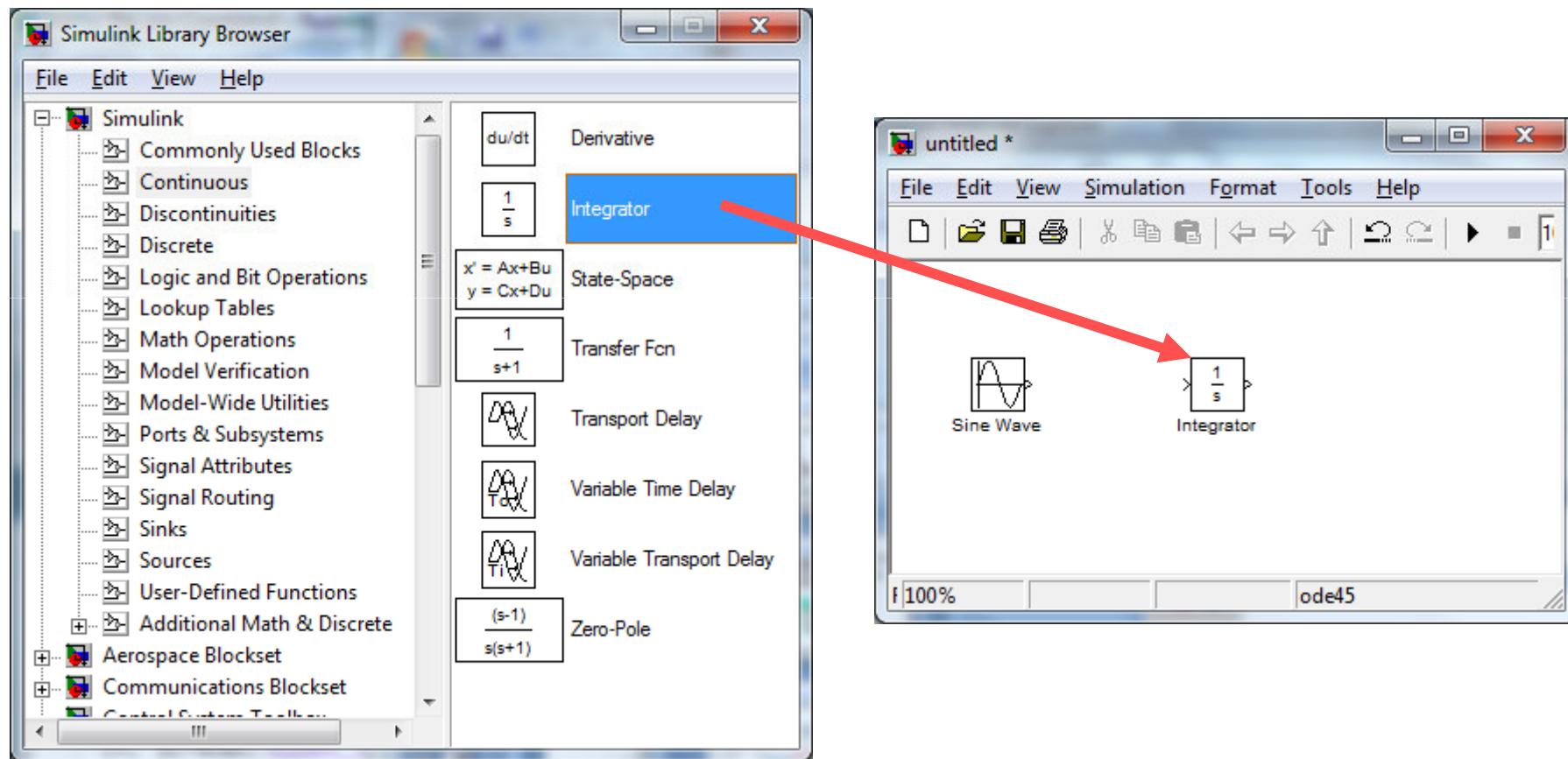


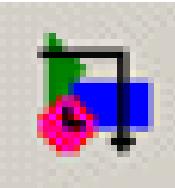
Selección de bloques para el modelo



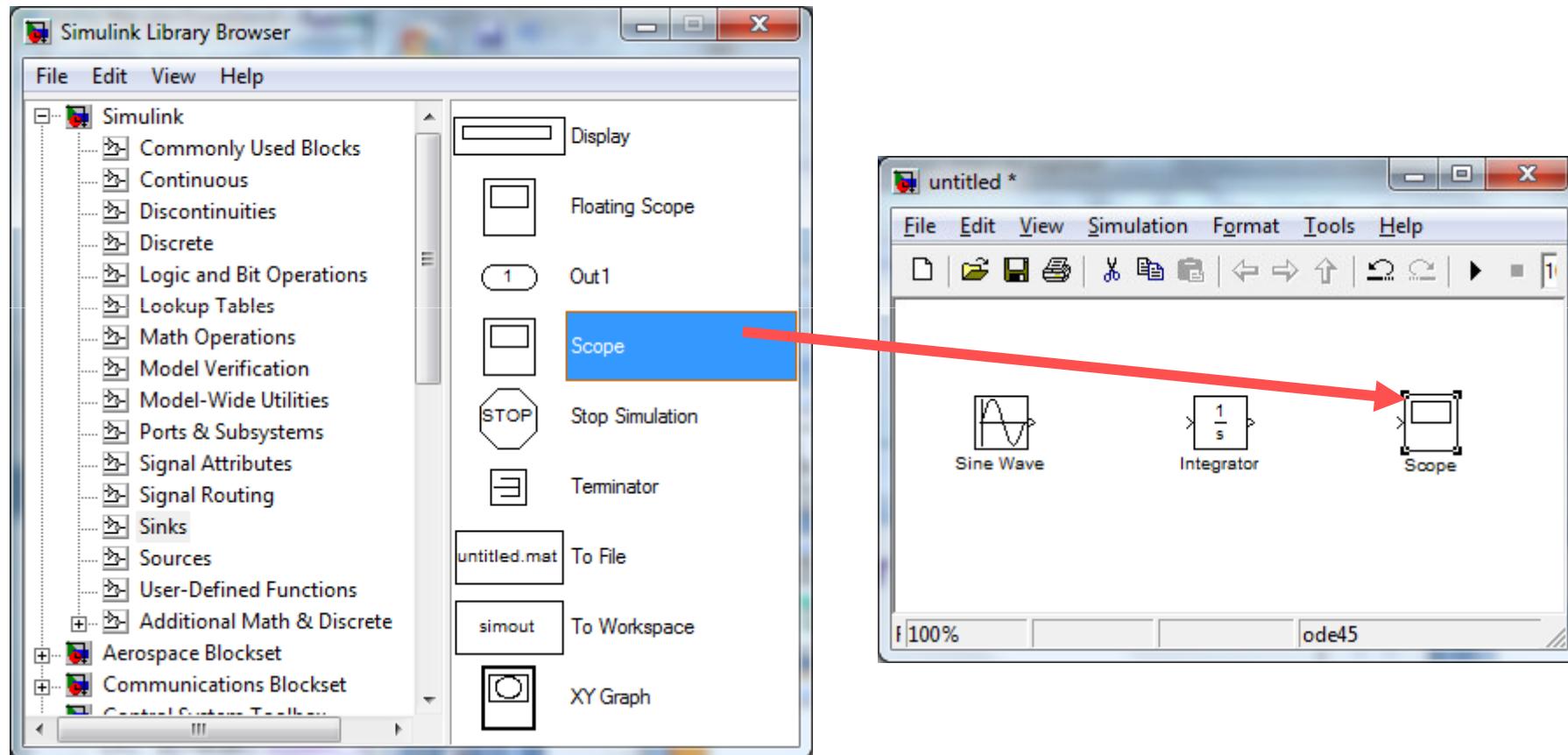


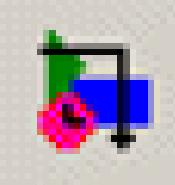
Selección de bloques para el modelo





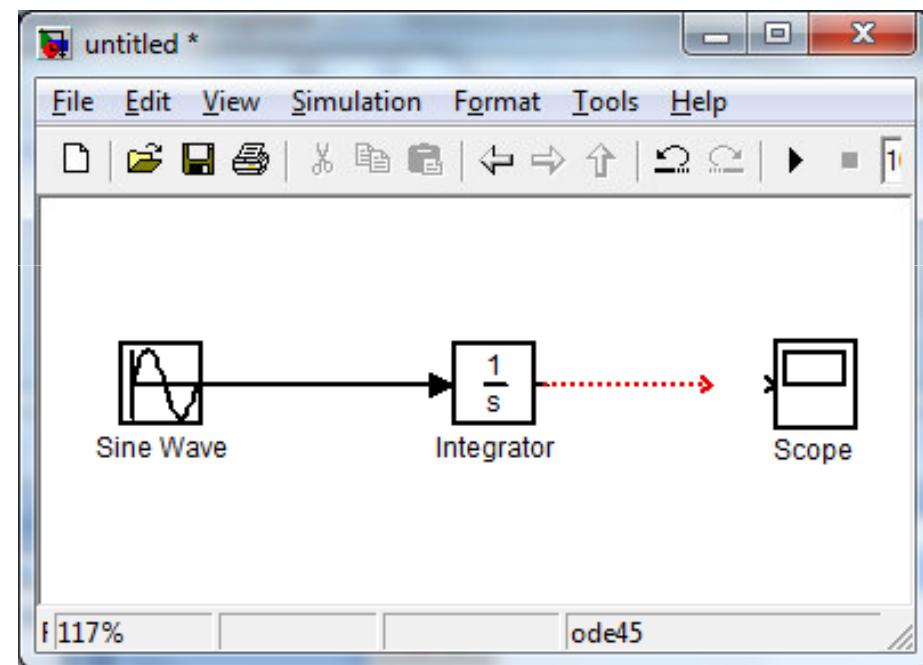
Selección de bloques para el modelo



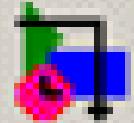


Conexión de los bloques con líneas de señal

- Colocar el cursor en el puerto de salida (> a la derecha) del bloque “Sine Wave”. El cursor cambia de forma a cruz
- Arrastrar desde el puerto de salida del bloque “Sine Wave” hasta el puerto de entrada (> a la izquierda) del bloque “Integrator”. Cuando el cursor se encuentra sobre el puerto de entrada cambia de forma a cruz doble
- Arrastrar desde la salida del bloque “Integrator” hasta la entrada del bloque “Scope”



Las flechas indican la dirección de la señal.

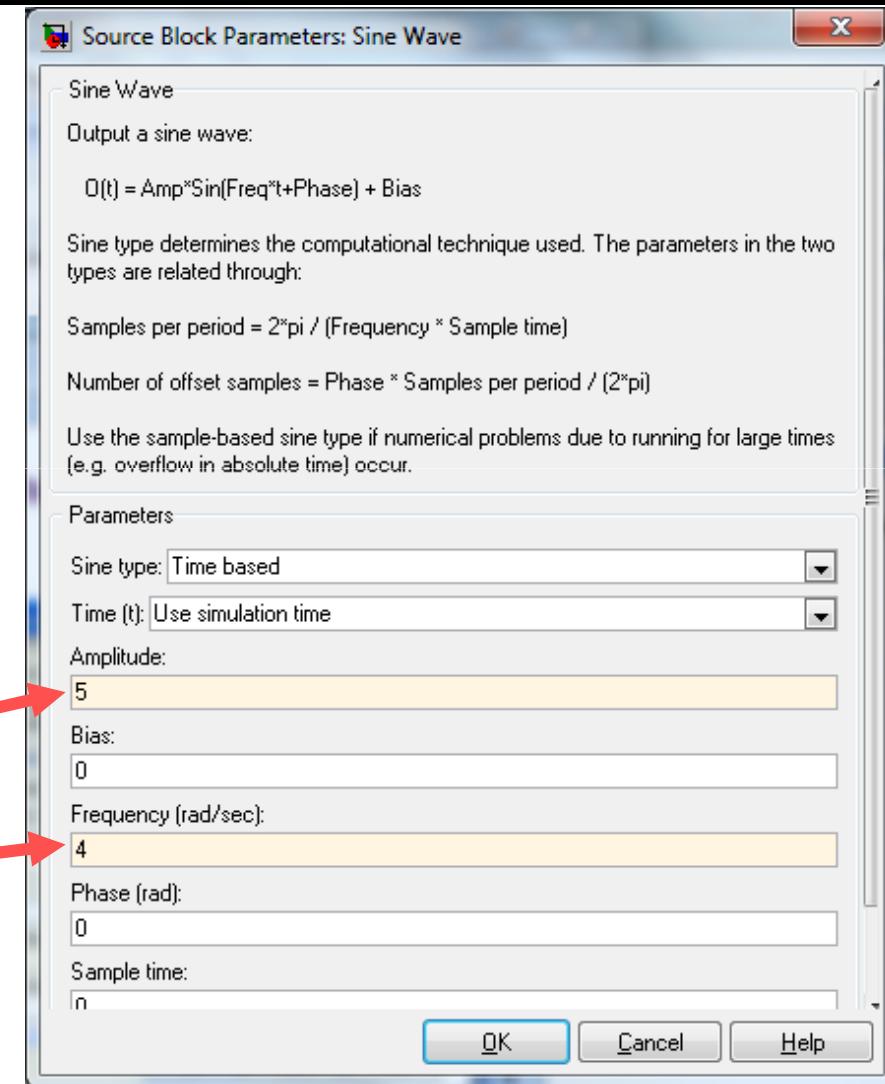


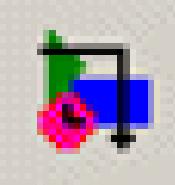
Configurar bloques con datos del modelo

- El input del modelo es:
 $5\sin(4t)$
- Para ello se hace doble click en el bloque “Sine Wave” y en la ventana de diálogo de los parámetros del bloque ingresar:

Amplitude = 5

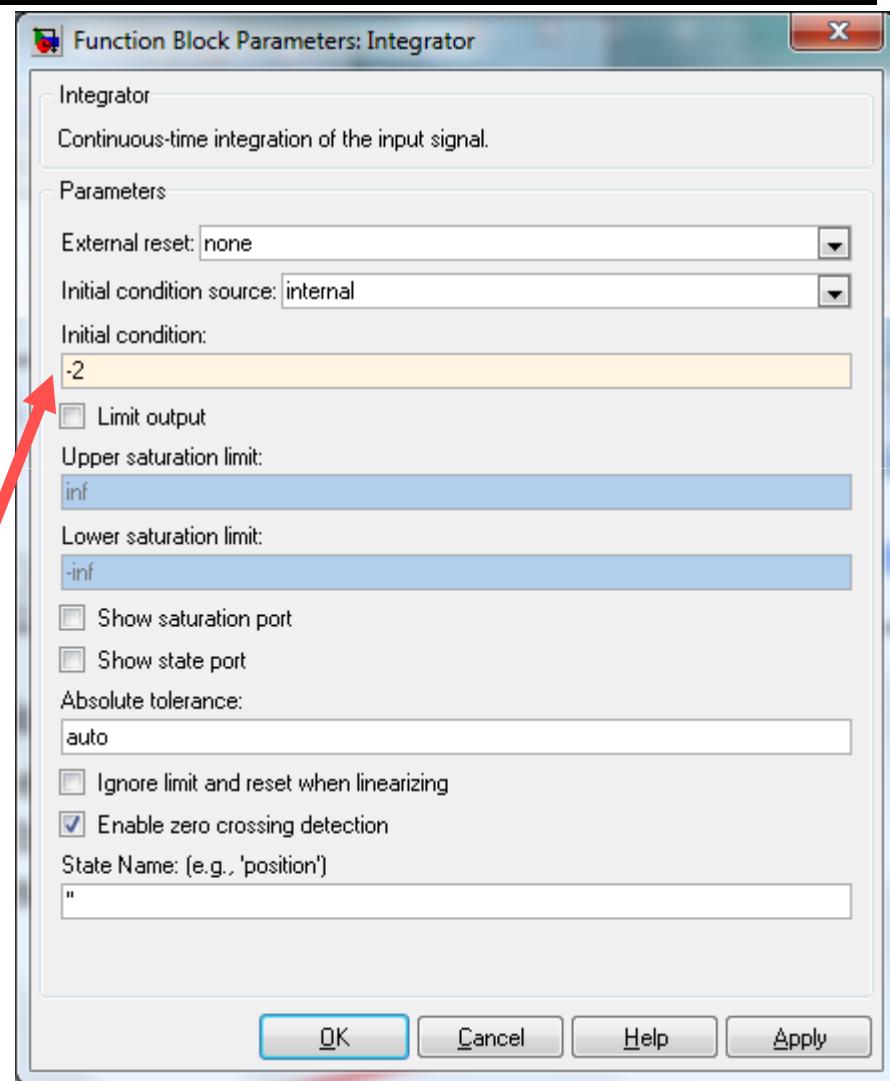
Frequency = 4

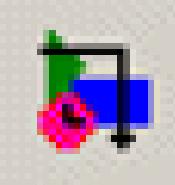




Configurar bloques con datos del modelo

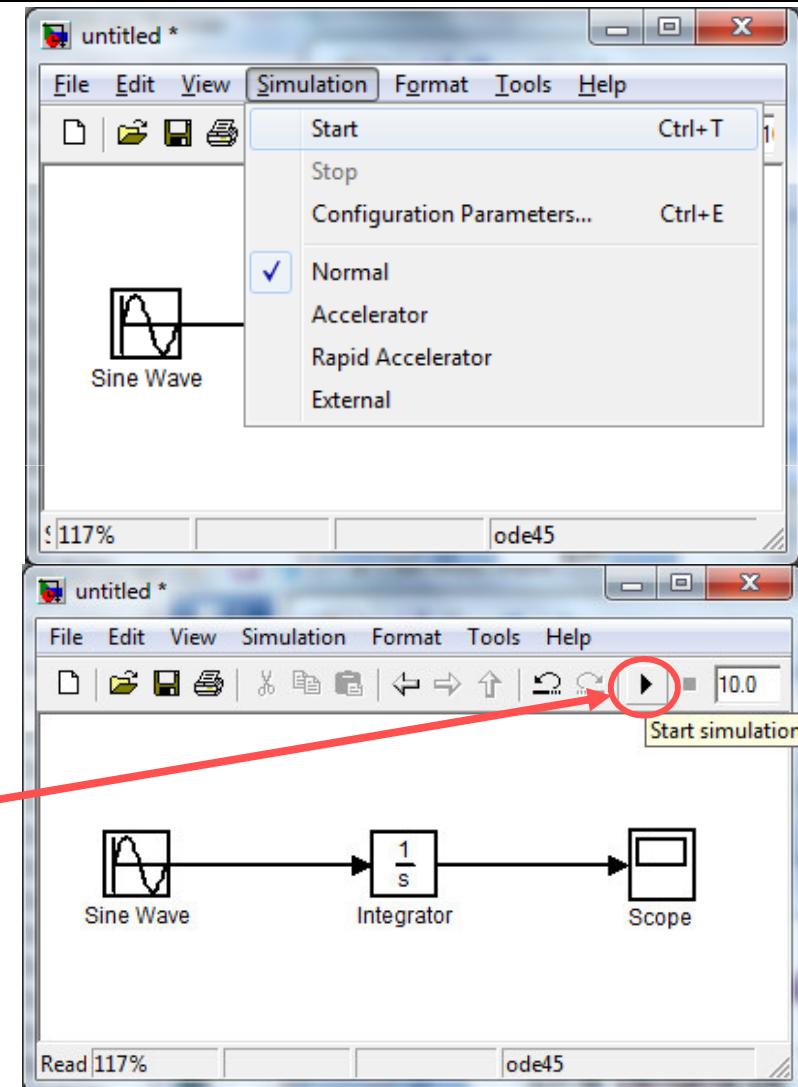
- El valor inicial es: -2
- Para ello se hace doble click en el bloque “Integrator” y se ingresa la condición inicial = -2

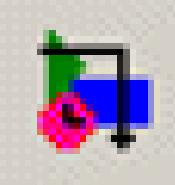




Ejecutar la simulación

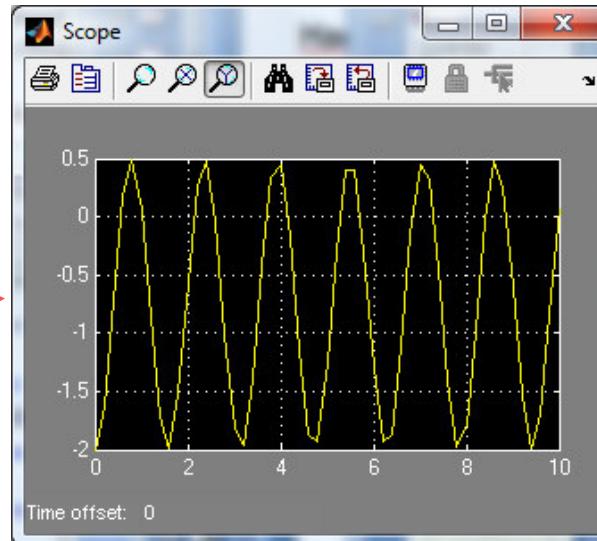
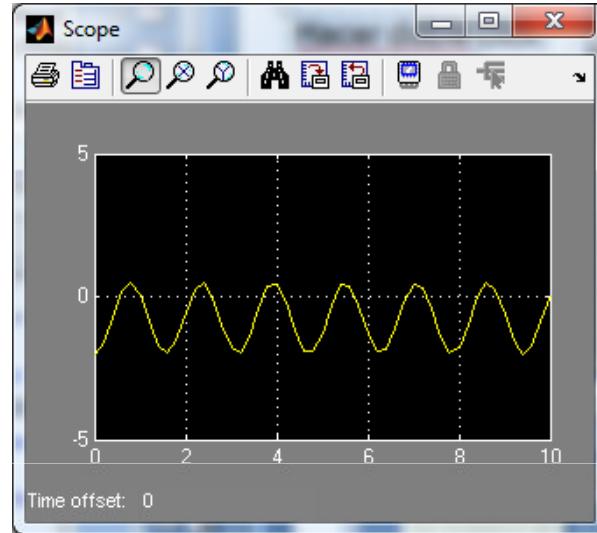
- En la ventana de trabajo, click en “Simulation” y seleccionar “Start”
- Otra forma es hacer click en el icono Start ➤

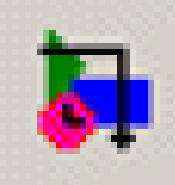




Visualizar resultados de la simulación

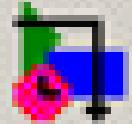
- Hacer doble click en el bloque “Scope”
- Se visualiza el output $x(t)$ en la ventana Scope
- Se puede mejorar la visualización utilizando los iconos de la ventana. Ej.: Autoscale y Tick labels all





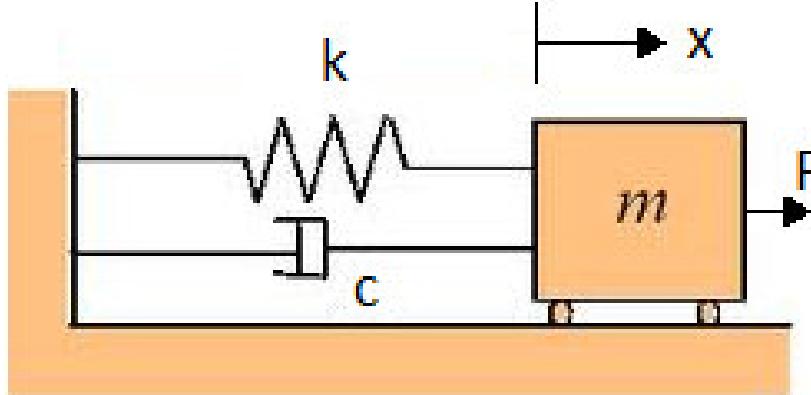
Visualizar resultados de la simulación

- Ejercicio: Poner en la misma gráfica la señal de entrada y la de salida



Sistema masa-muelle-amortiguador

- Descripción:



- Modelo:

$$m \frac{d^2x}{dt^2} + c \frac{dx}{dt} + kx = f(t)$$

Notación simplificada:

$$m\ddot{x} + c\dot{x} + kx = F$$

- Condiciones iniciales, sistema en equilibrio:

– función de la fuerza es un escalón con magnitud 3

– valores de los parámetros: $m = 0.25$, $c = 0.5$, $k = 1$

Notación Laplace:

$$ms^2X(s) + csX(s) + kX(s) = F(s)$$

$$G(s) = \frac{X(s)}{F(s)} = \frac{1/m}{s^2 + \frac{c}{m}s + \frac{k}{m}}$$

FT

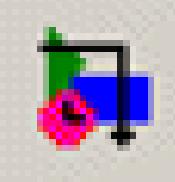
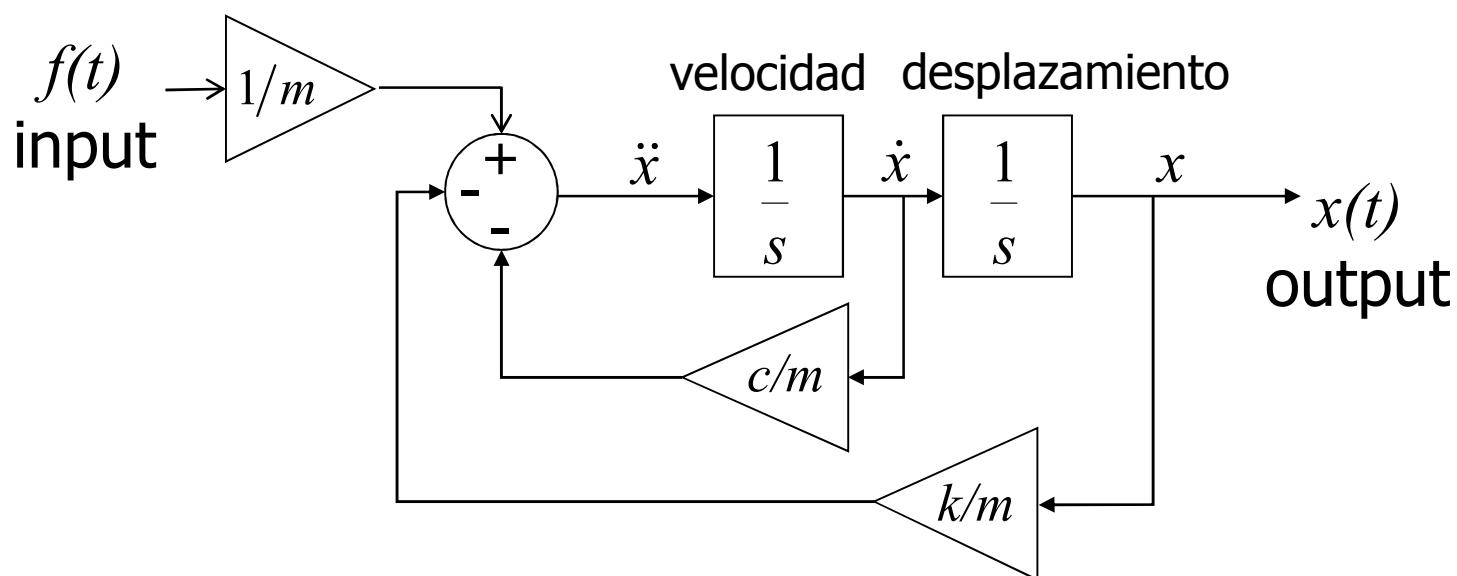


Diagrama de simulación del sistema continuo de segundo orden

- Si se expresa la ecuación en términos de la derivada de mayor orden

$$m\ddot{x} + c\dot{x} + kx = f(t) \rightarrow \ddot{x} = \frac{1}{m}f(t) - \frac{k}{m}x - \frac{c}{m}\dot{x}$$

- Diagrama de bloques asociado:



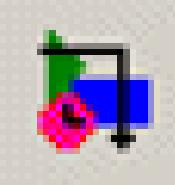
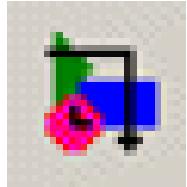


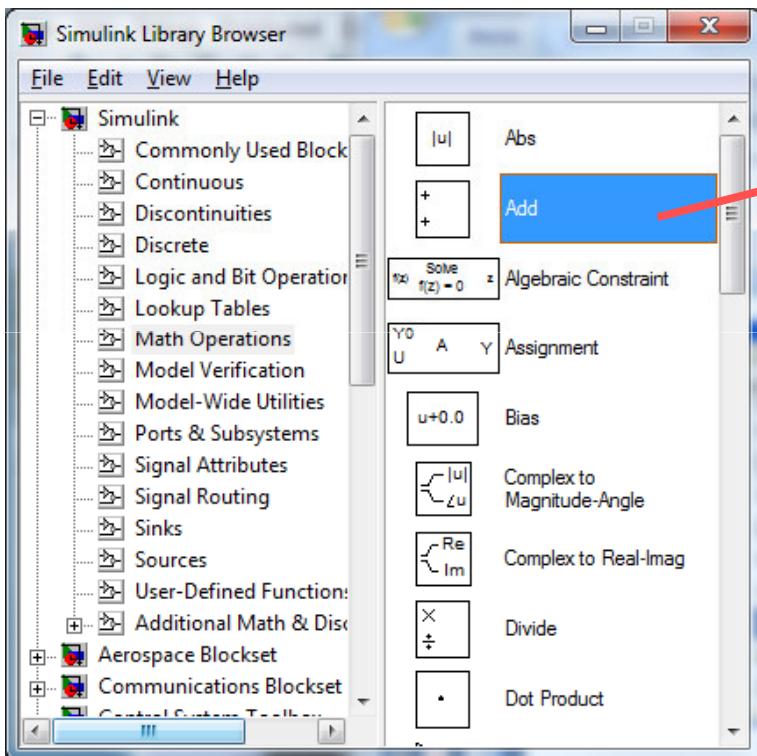
Diagrama de bloques Simulink del sistema continuo de segundo orden

- A partir del modelo se identifican los bloques necesarios en las librerías Simulink:
 - Bloque Step (1) - Librería Sources
 - Bloque Sum (1) - Librería Math Operations
 - Bloque Gain (3) - Librería Math Operations
 - Bloques Integrator (2) - Librería Continuos
 - Bloque Scope (1) - Librería Sink
- Cada bloque requiere ser configurado con las ICs:
 - el sistema inicialmente está en equilibrio: $\dot{x} = 0, x = 0$
 - input: escalón con magnitud 3
 - valores de los parámetros: $m = 0.25, c = 0.5, k = 1$

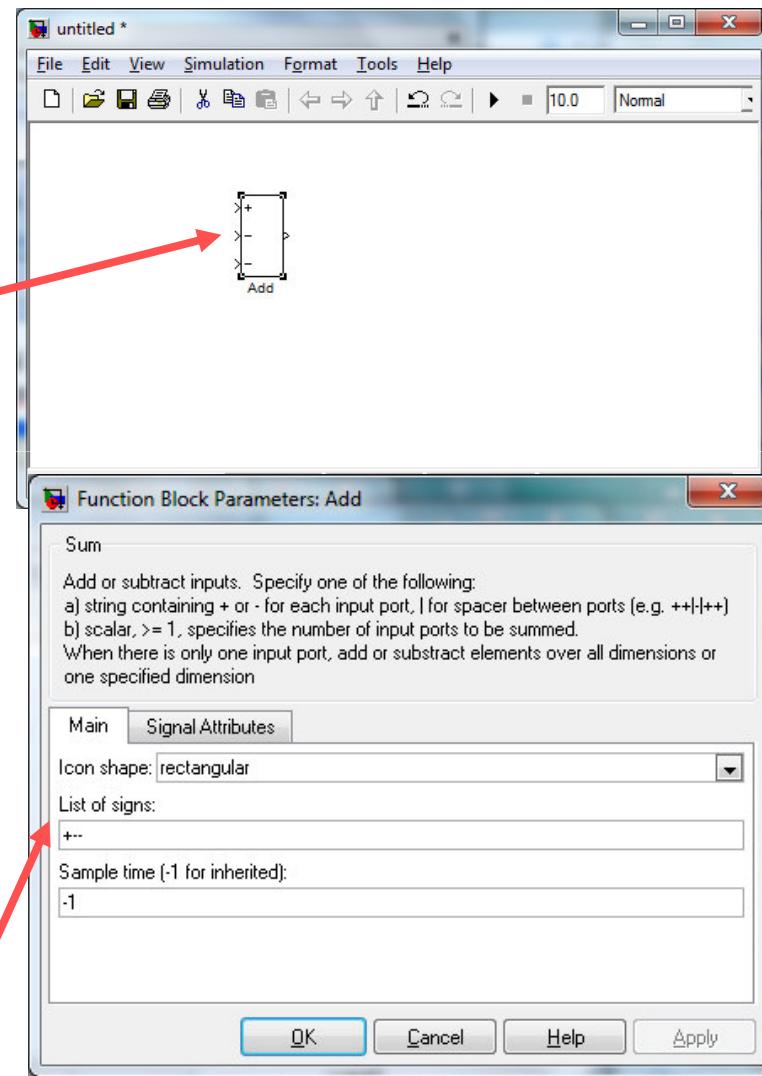


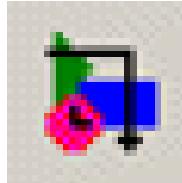
Creación del diagrama de bloques en Simulink (scso)

Arrastrar el bloque “*Sum*” desde la librería “*Math Oper*”



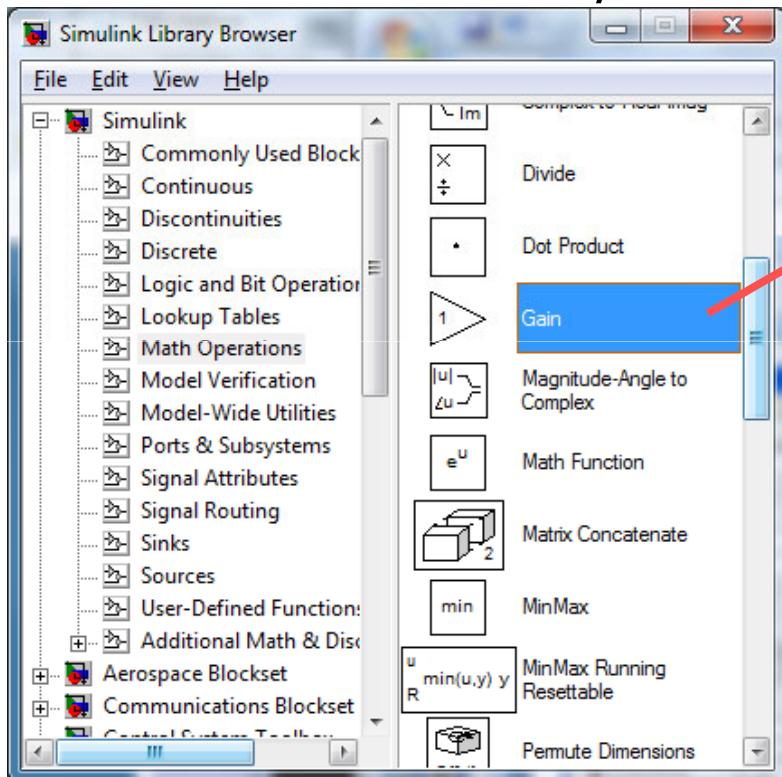
Doble click en *Sum* para cambiar los parámetros a *rectangular* y *+ - -*



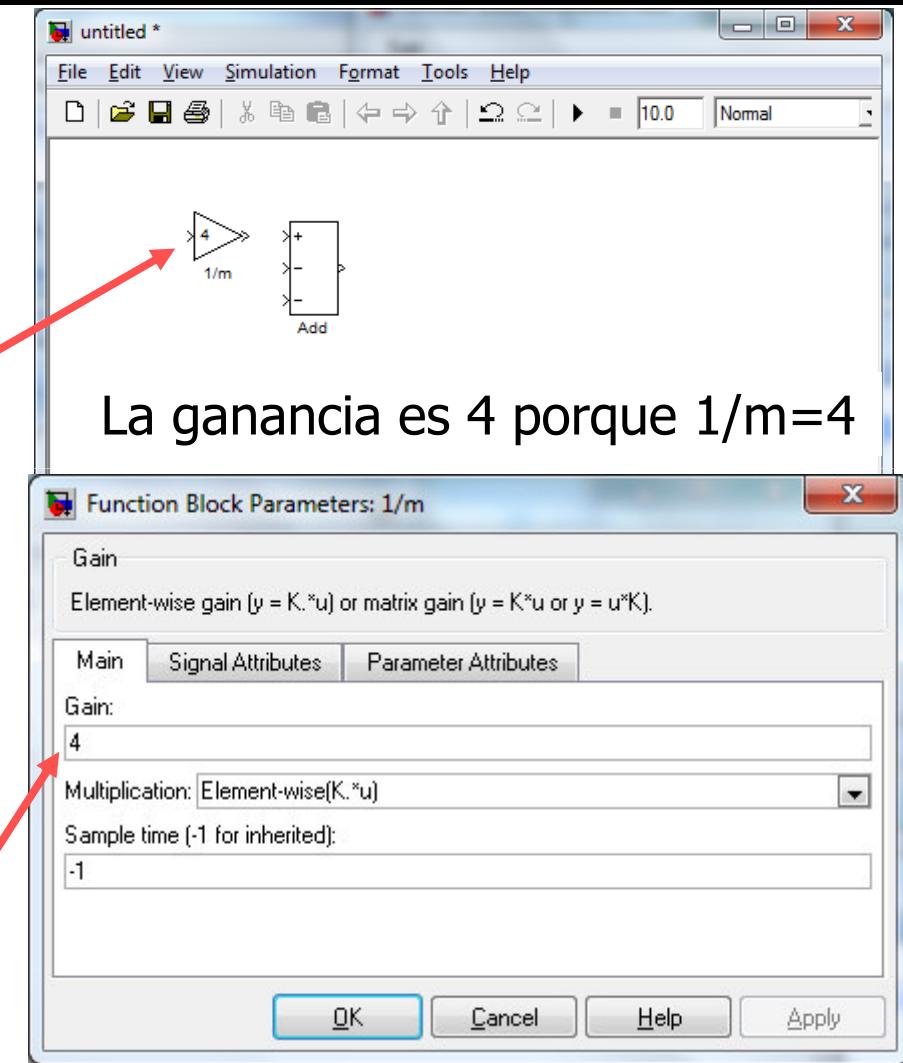


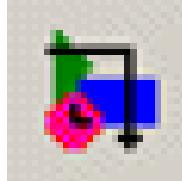
Creación del diagrama de bloques en Simulink (scso)

Arrastrar el bloque “*Gain*” desde la librería “*Math Oper*”



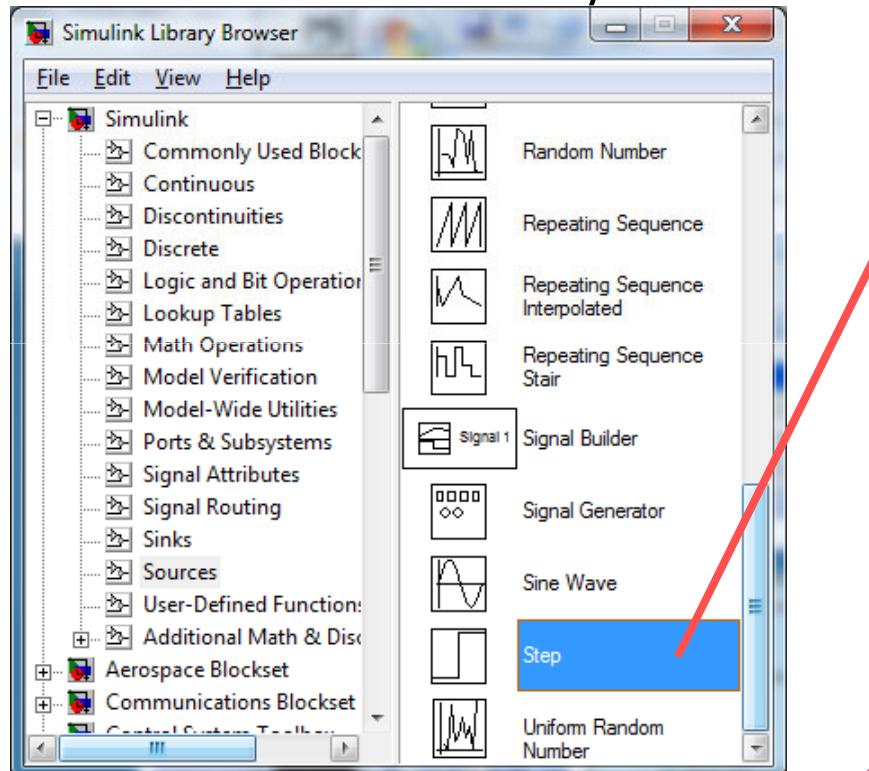
Doble click en *Gain* para cambiar los parámetros del bloque
Cambiar etiqueta



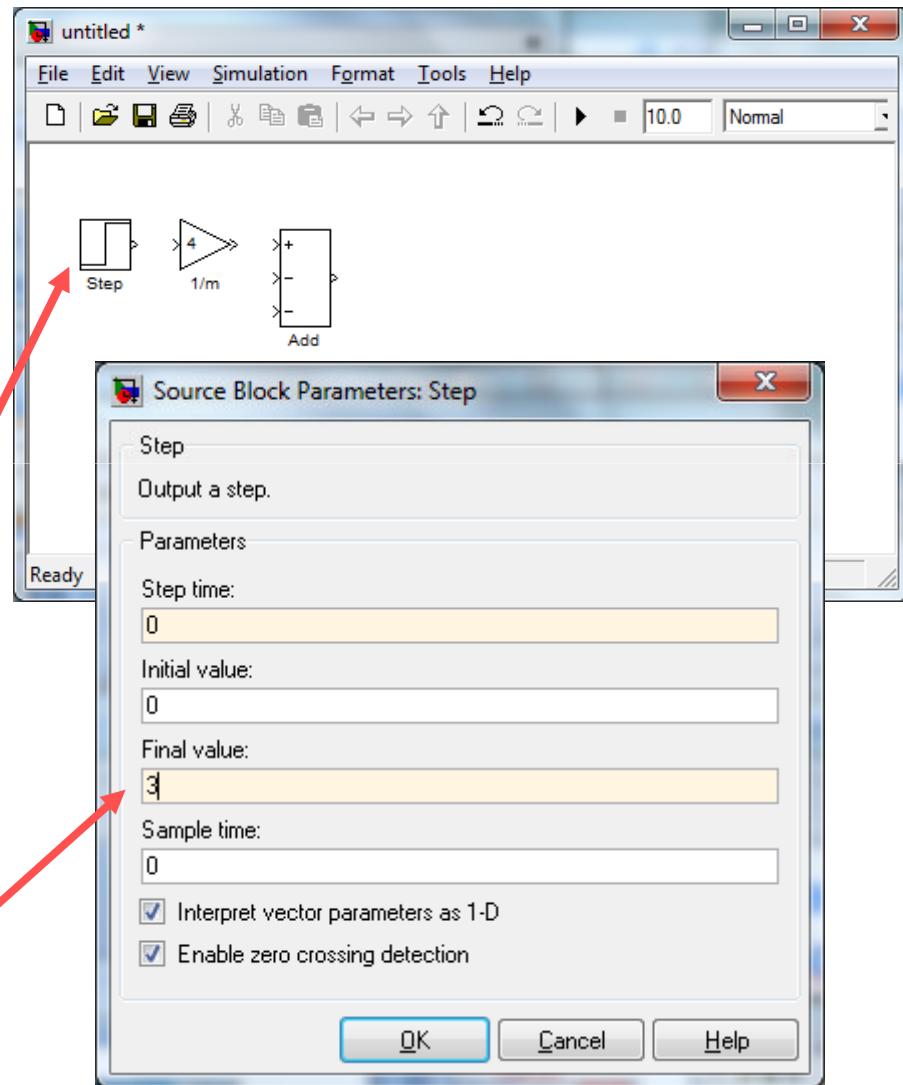


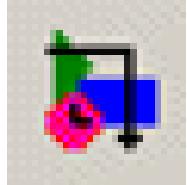
Creación del diagrama de bloques en Simulink (scso)

Arrastrar el bloque "Step"
desde la librería "Math Oper"



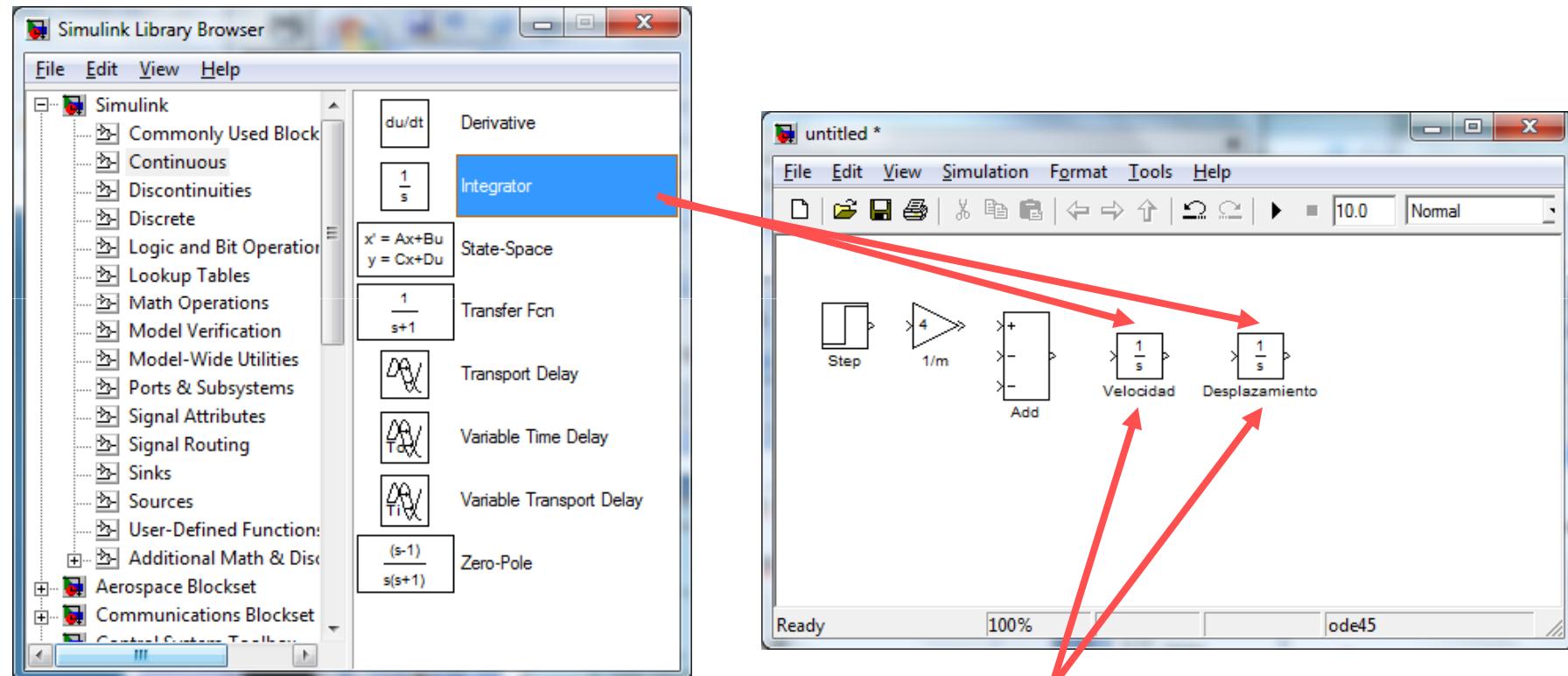
Doble click en *Step* para cambiar
los parámetros *Step time* a 0 y
Final value a 3



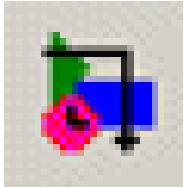


Creación del diagrama de bloques en Simulink (scso)

Arrastrar los bloques “*Integrator*” desde la librería “*Continuous*”

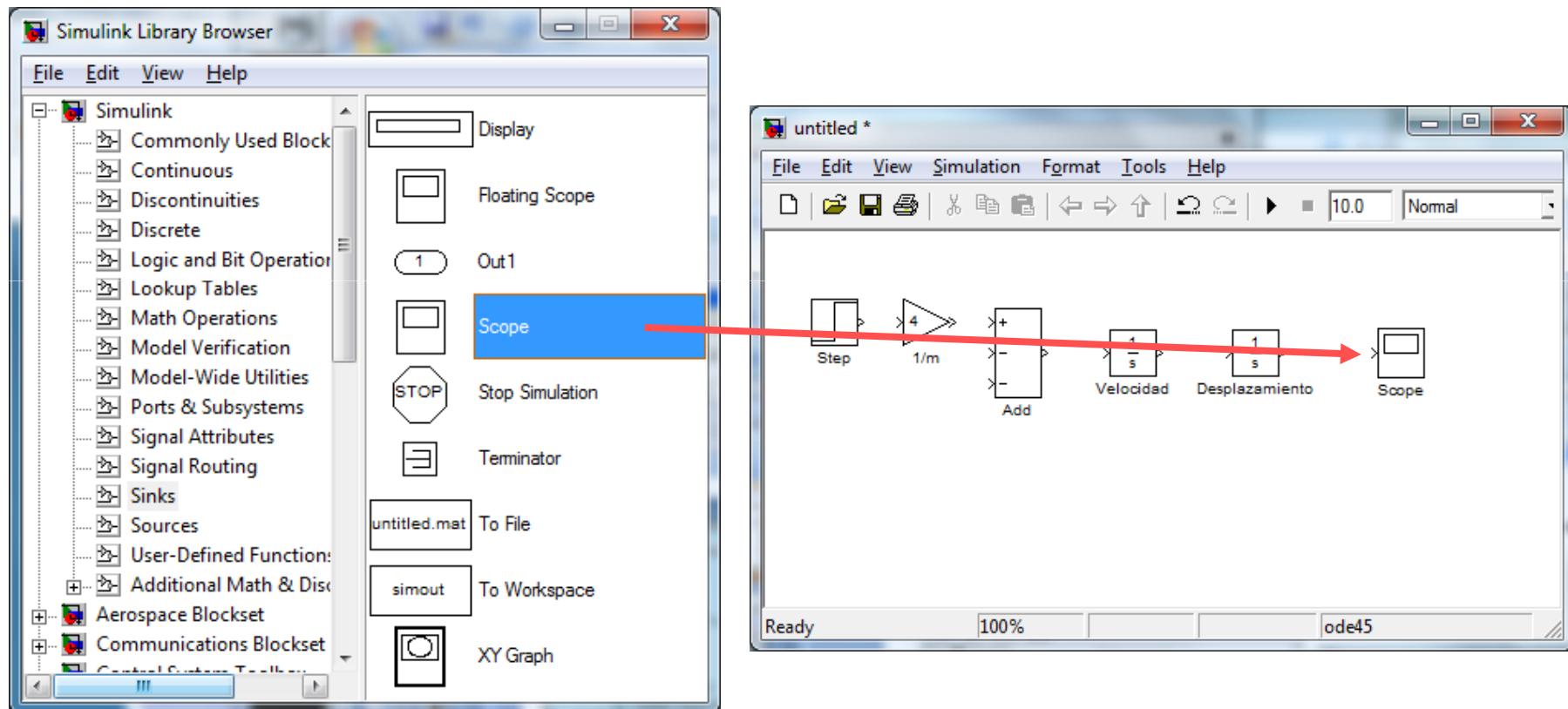


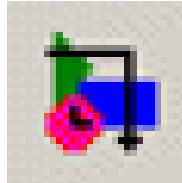
El bloque “integrator” tiene por defecto valor inicial 0. No es necesario modificarlo.
Se cambian los títulos



Creación del diagrama de bloques en Simulink (scso)

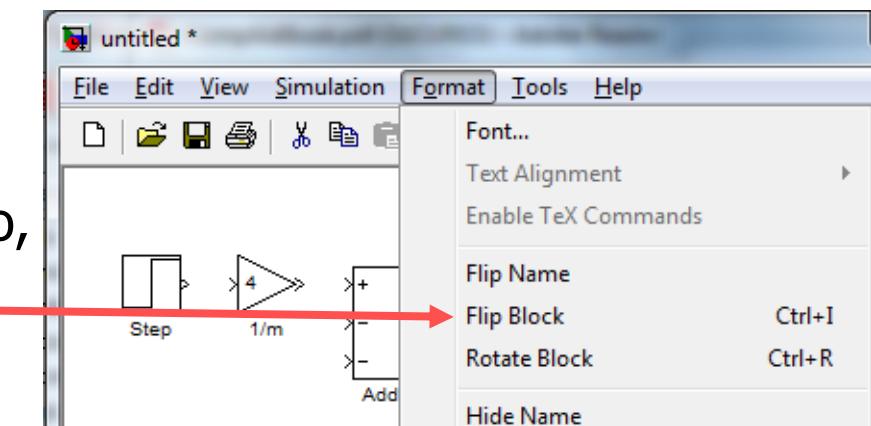
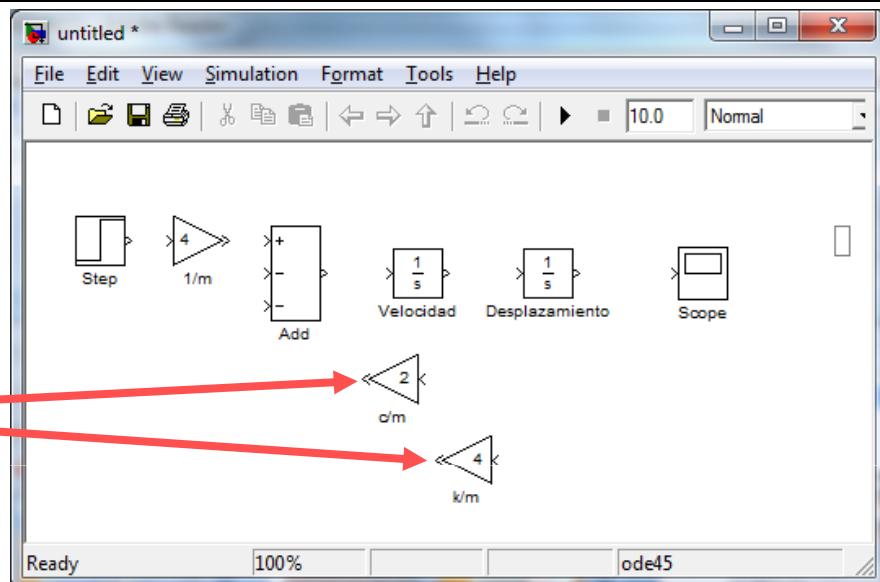
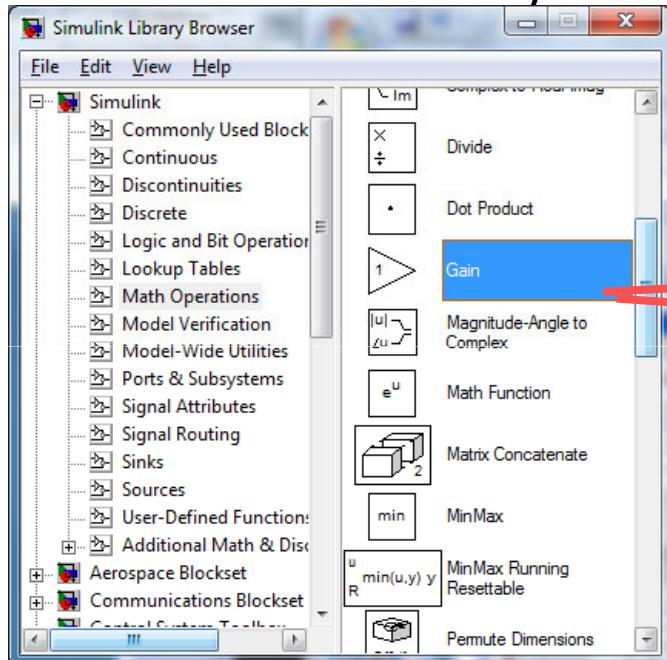
Arrastrar el bloque "Scope" desde la librería "Sink"



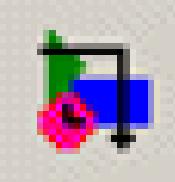


Creación del diagrama de bloques en Simulink (scso)

Crear dos nuevos bloques “Gain” desde la librería “Math Oper”



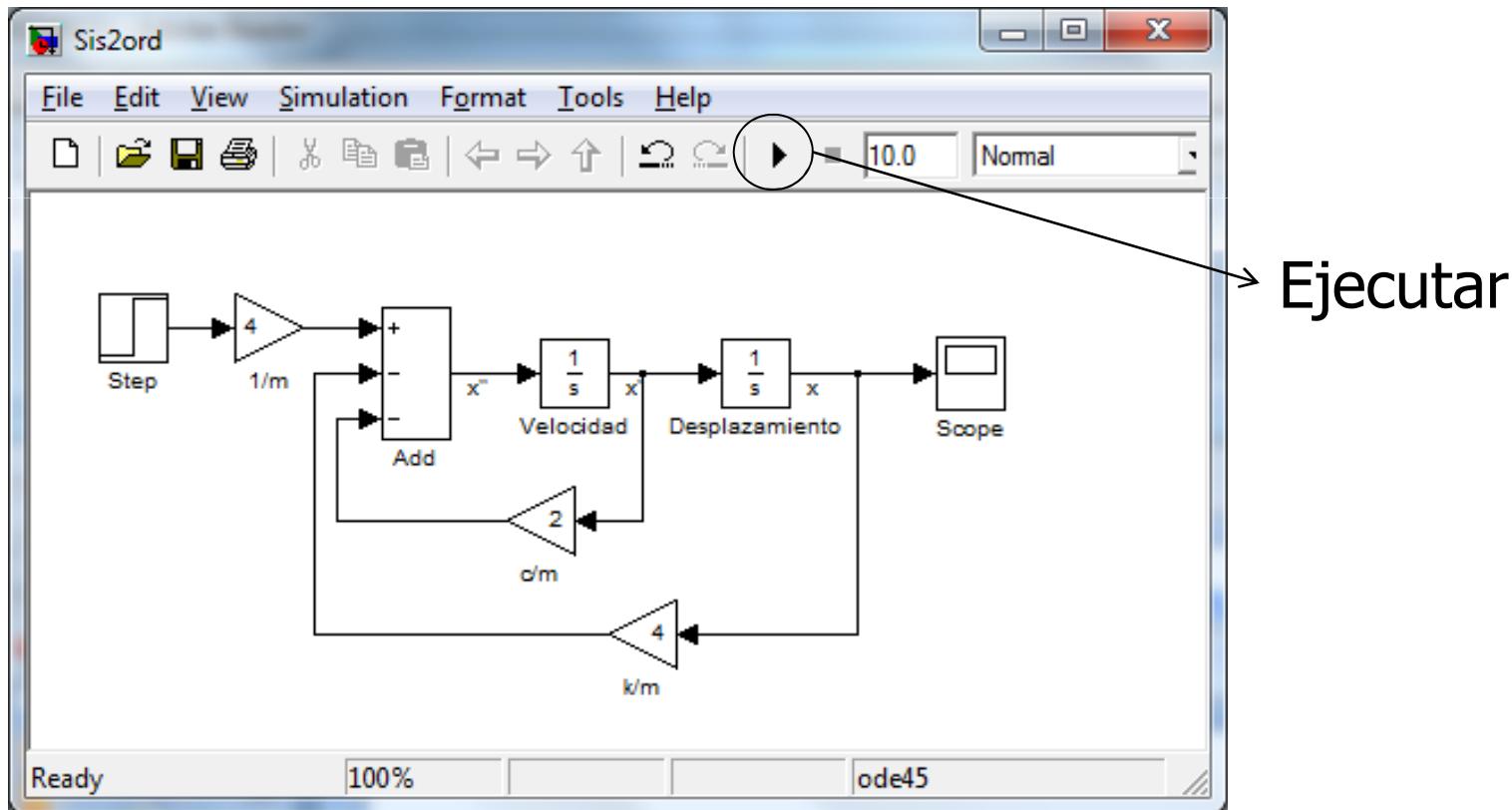
Para girar el bloque gain, seleccionarlo, y elegir Flip Block en el menú Format. Cambiar etiqueta. Doble click en los bloques Gain para modificar los parámetros. $c/m = 2$, $k/m = 4$

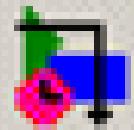


Creación del diagrama de bloques en Simulink (scso)

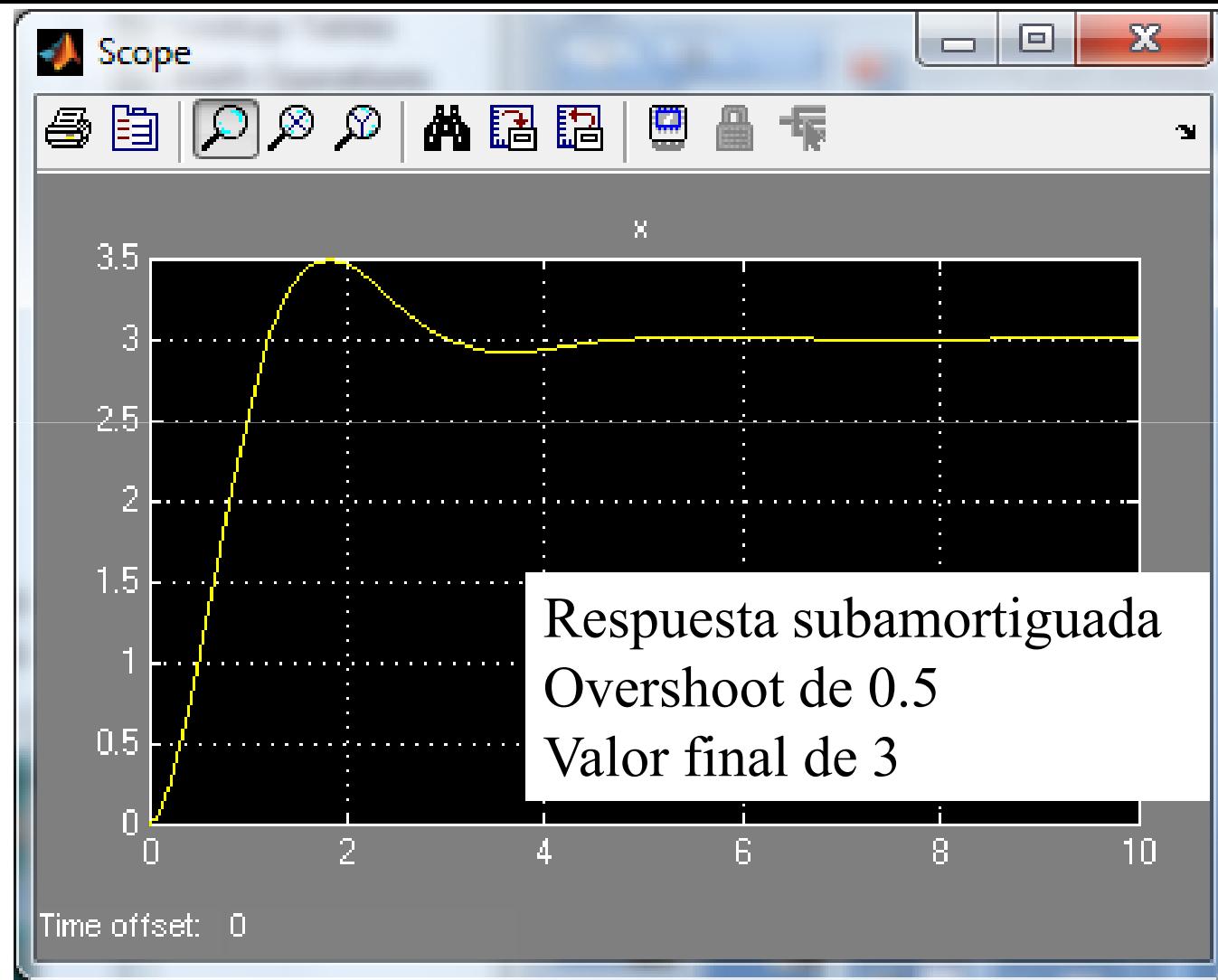
Conectar todos los bloques. Etiquetar las señales haciendo doble click sobre las líneas.

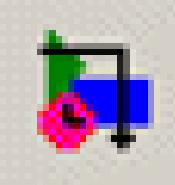
Para hacer derivaciones de una línea se coloca el cursor sobre la línea y se mantiene pulsada la tecla Ctrl hasta conectar al otro bloque.





Resultados del modelo en Simulink (scso)





Comprobación de resultados

- Forma Standard

$$\frac{\ddot{x}}{m} + \frac{c}{k}\dot{x} + x = \frac{1}{k}f(t)$$

- Frecuencia natural

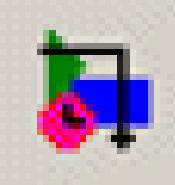
$$\omega_n = \sqrt{\frac{k}{m}} = 2.0$$

- Tasa de amortiguamiento

$$\frac{2\zeta}{\omega_n} = \frac{c}{k} \rightarrow \zeta = 0.5$$

- Ganancia estática

$$K = \frac{1}{k} = 1$$



Uso del bloque de función de transferencia en un scso

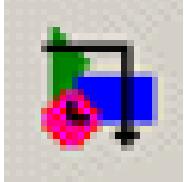
- El siguiente sistema muelle-masa-amortiguador se resuelve según la ec. de movimiento (sin fricción):
$$m\ddot{x} + c\dot{x} + kx = f(t)$$
- Tomando la transformada de Laplace e ignorando las condiciones iniciales

$$ms^2 X(s) + csX(s) + kX(s) = F(s)$$

- La función de transferencia (output/input $X(s)/F(s)$) es:

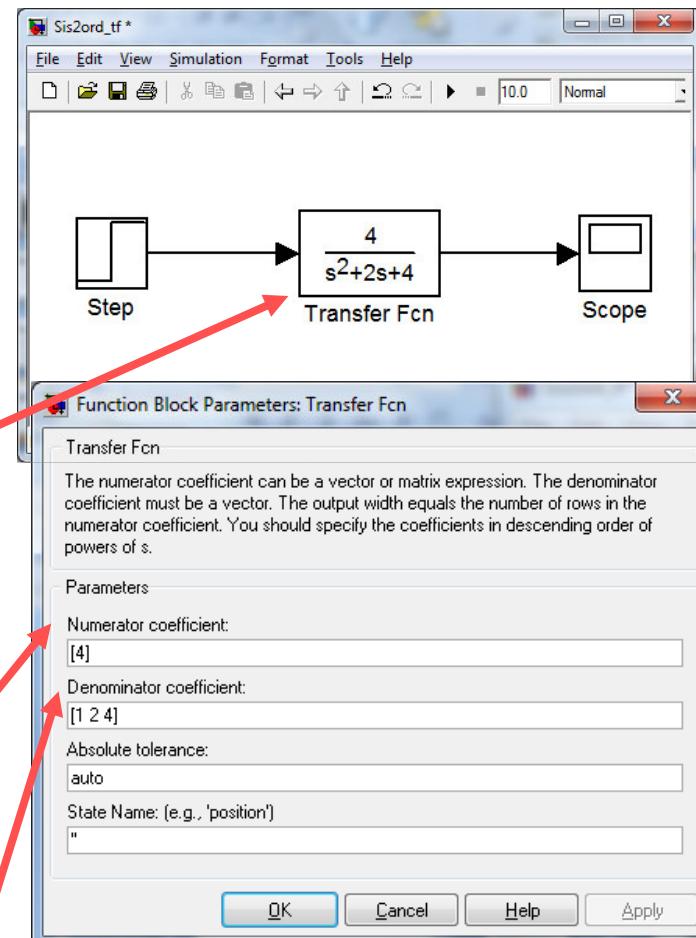
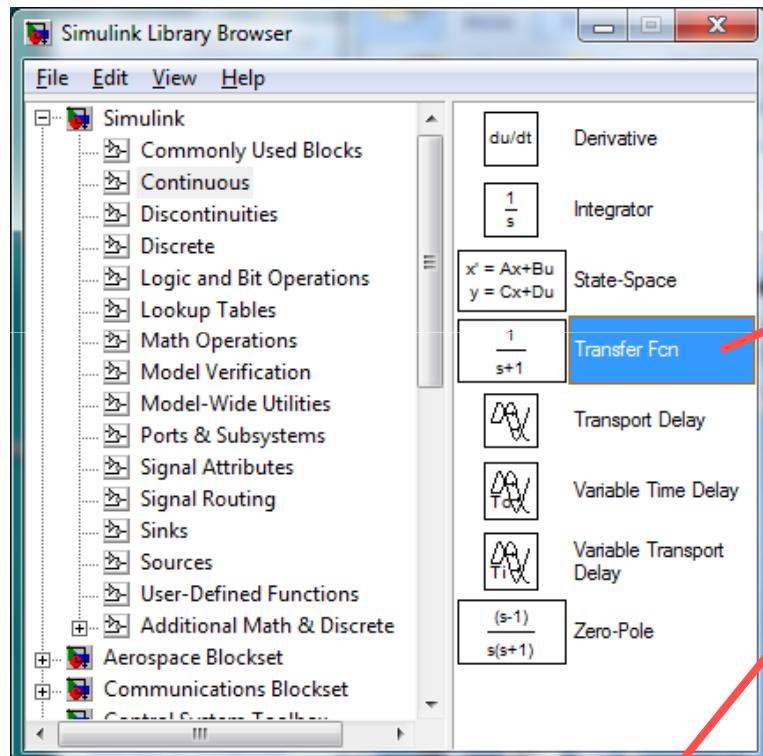
$$G(s) = \frac{X(s)}{F(s)} = \frac{1/m}{s^2 + \frac{c}{m}s + \frac{k}{m}}$$

$$G(s) = \frac{4}{s^2 + 2s + 4}$$

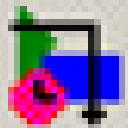


Uso del bloque de función de transferencia en Simulink para un scso

Arrastrar los bloques “*Transfer Fcn*” desde la librería “*Continuous*”



Ajustar el Numerator y Denominator coefficient a [4] y [1 2 4] del bloque “Transfer Fcn”



Bloque de espacio de estado

- El bloque espacio de estado (*state-space*) es una alternativa al bloque función de transferencia
- Las variables de estado para el sistema mma son posición y velocidad

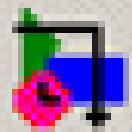
$$x_1 = x$$

$$x_2 = \dot{x}$$

- Las derivadas de las variables de estado son

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -\frac{k}{m}x_1 - \frac{c}{m}x_2 - \frac{1}{m}F$$



Bloque de espacio de estado

- En notación matricial

$$\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{Bu}$$

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix}$$

$$\mathbf{u} = F$$

$$\mathbf{B} = \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix}$$

$$\mathbf{y} = \mathbf{Cx} + \mathbf{Du}$$

$$y = x_1$$

$$\mathbf{C} = [1 \ 0]$$

$$\mathbf{D} = 0$$

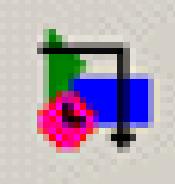
donde:

A = matriz del sistema

B = matriz de entrada

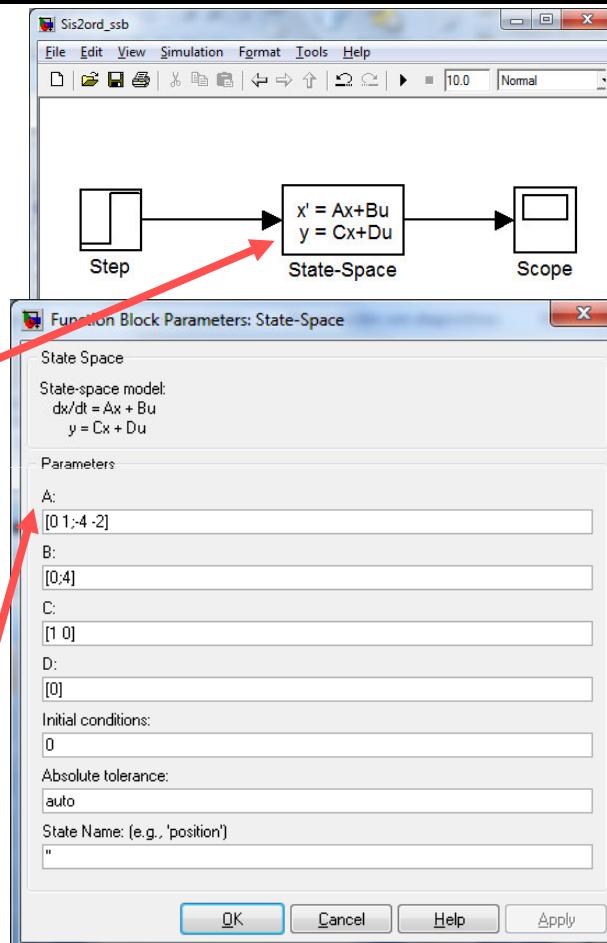
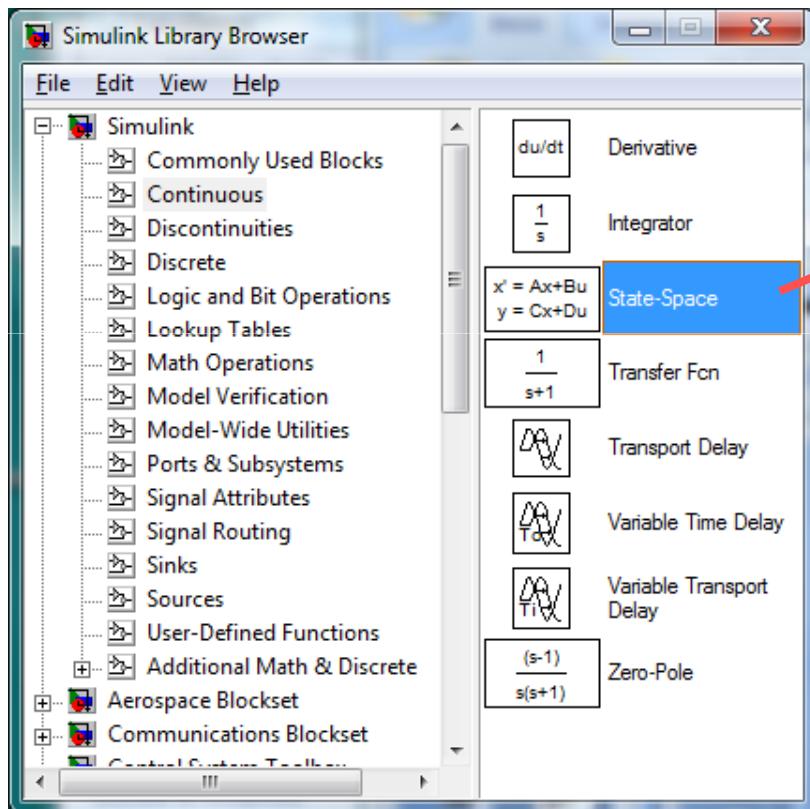
C = matriz de salida

D = matriz de transmitancia directa

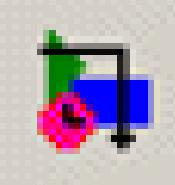


Bloque de espacio de estado en Simulink para un scso

Arrastrar los bloques “*State Space*” desde la librería “*Continuos*”



Ajustar los valores de las matrices $A=[0 \ 1; -4 \ -2]$, $B=[0; \ 4]$, $C=[1 \ 0]$ y $D=[0]$ del bloque “*State Space*”



Sistema discreto Simulink

- Se modela la amortización de un préstamo de dinero:

$$b(k) = rb(k - 1) - p(k)$$

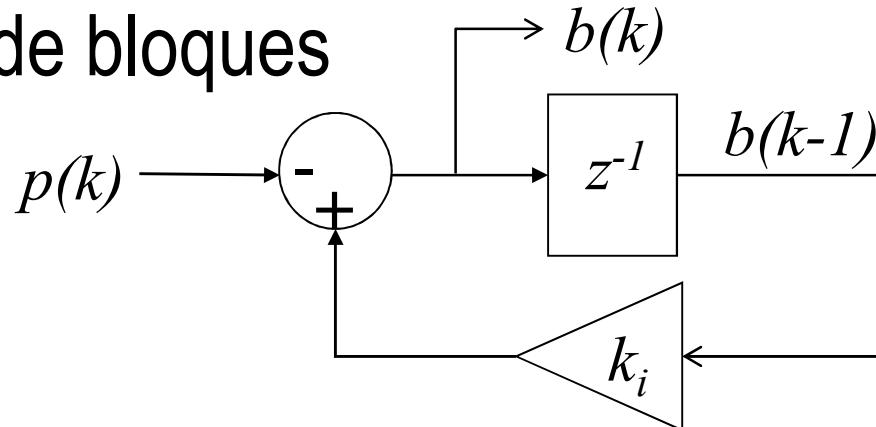
donde:

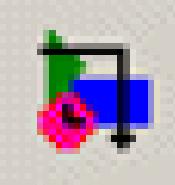
$b(k)$ pago mensual

$r = i+1$ siendo i el interés mensual, y

$p(k)$ pago de final de mes.

- Modelo de bloques





Modelo de bloques de sistema discreto

Simulink

- Si el balance de un préstamo es:

$$b(k) = rb(k - 1) - p(k)$$

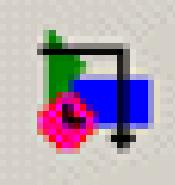
Considerar como condiciones iniciales:

Balance inicial: 15000

Interés: 1%

Pago mensual = 200

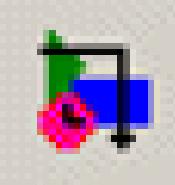
- Calcular el balance del préstamo despues de 100 pagos



Modelo de bloques de sistema discreto

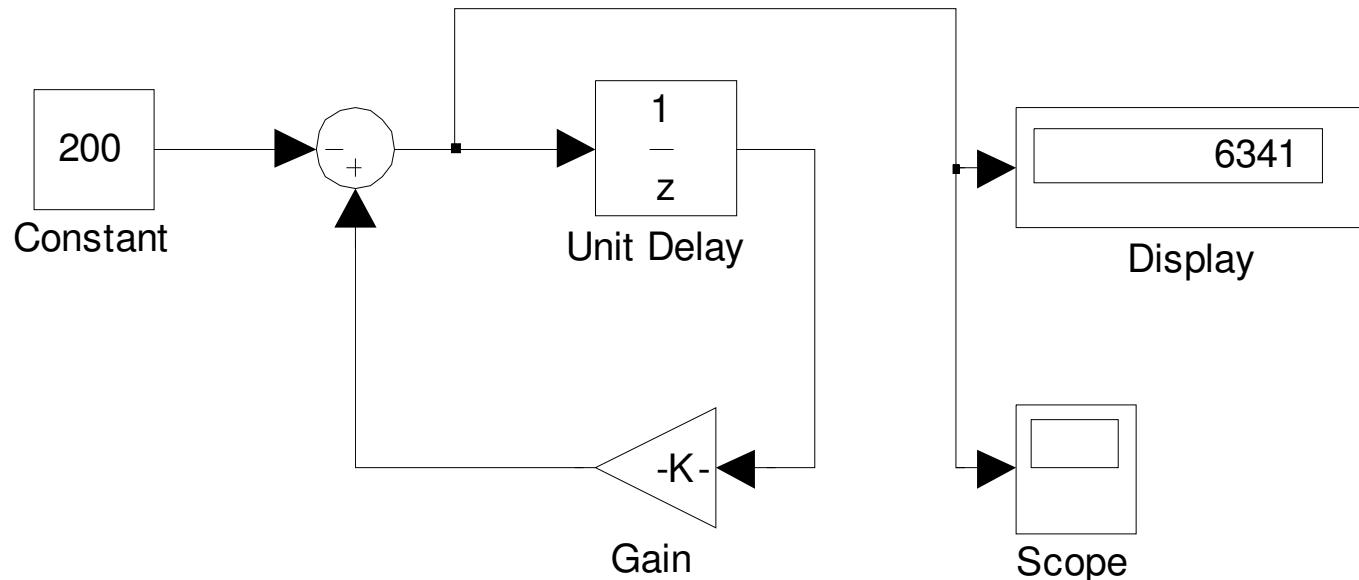
Simulink

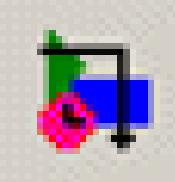
- A partir del modelo se identifican los bloques necesarios en las librerías Simulink:
 - Bloque Constant (1) - Librería Sources
 - Bloque Sum (1) - Librería Math Operations
 - Bloque Gain (3) - Librería Math Operations
 - Bloques Unit delay (1) - Librería Discrete
 - Bloque Scope y Display (1) - Librería Sink
- Cada bloque requiere ser configurado con las ICs:
 - Bloque Unit Delay, Initial Condition: 15000, Sample Time:
1
 - Bloque Gain: 1.01



Modelo de bloques de sistema discreto Simulink

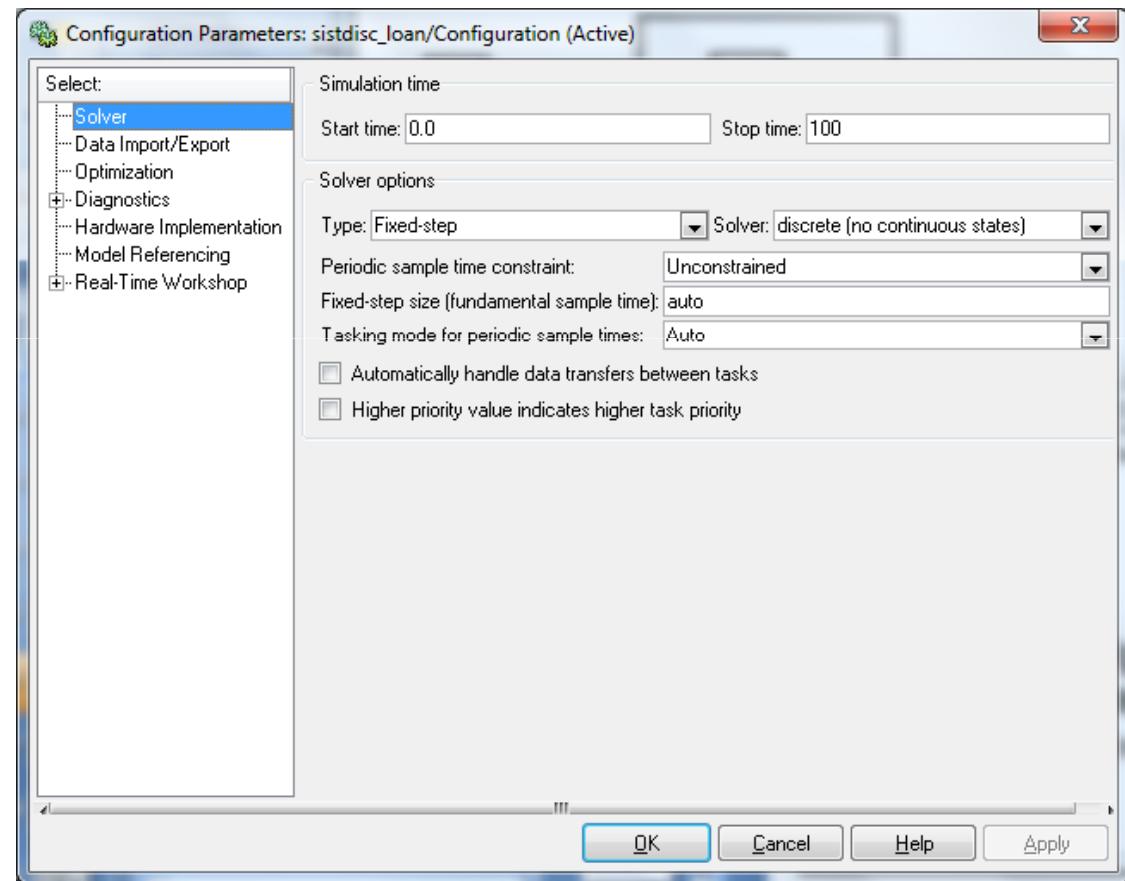
- Modelo Simulink:

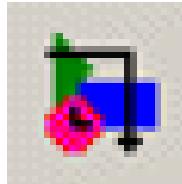




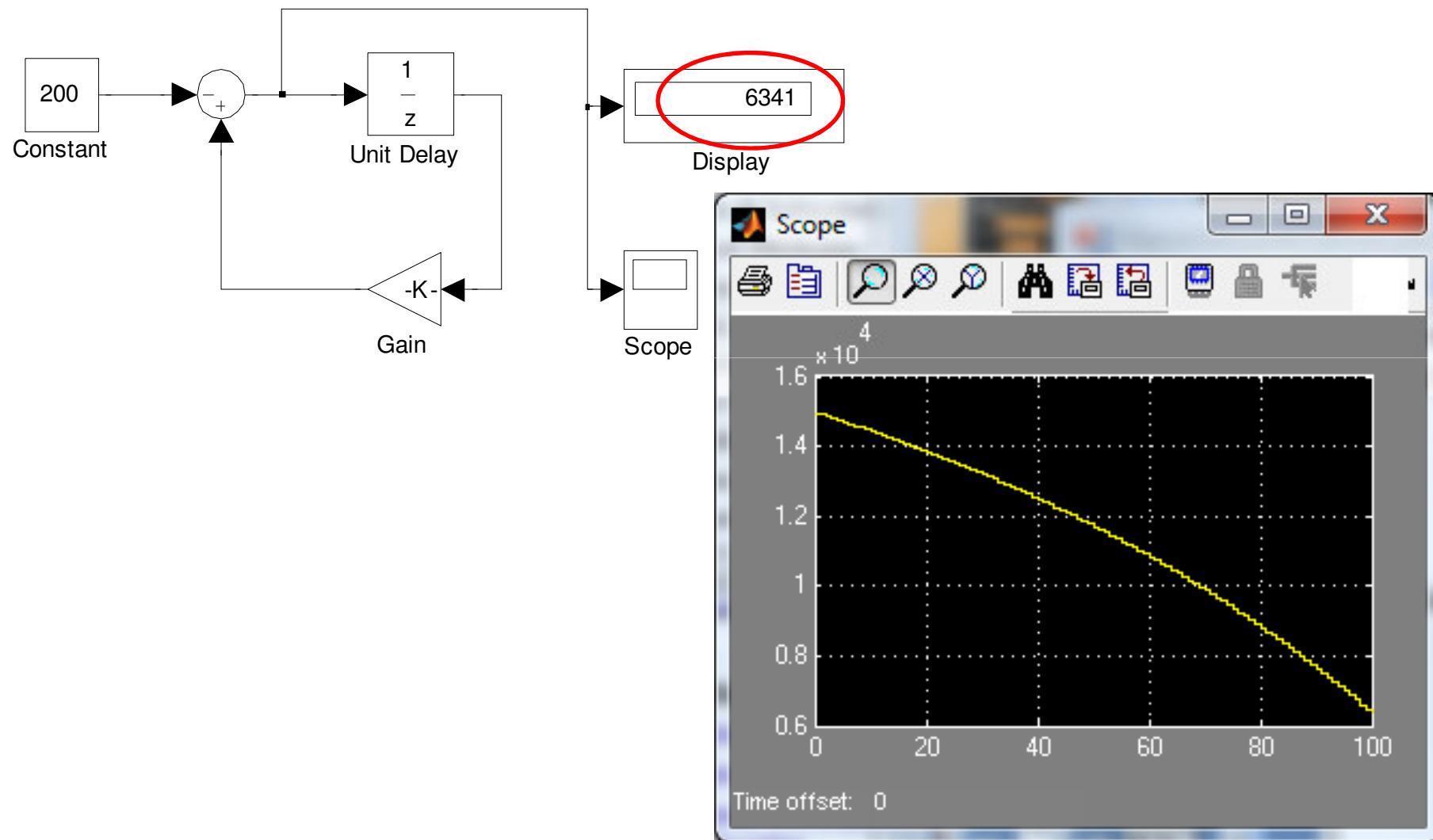
Modelo de bloques de sistema discreto Simulink

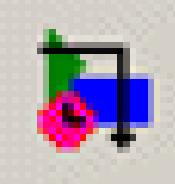
En Parameter Configuration
de Simulation ajustar a
Solver type: Fixed-step
Solver: discrete
Start time: 0 Stop
time: 100





Resultado de la simulación de sistema discreto Simulink



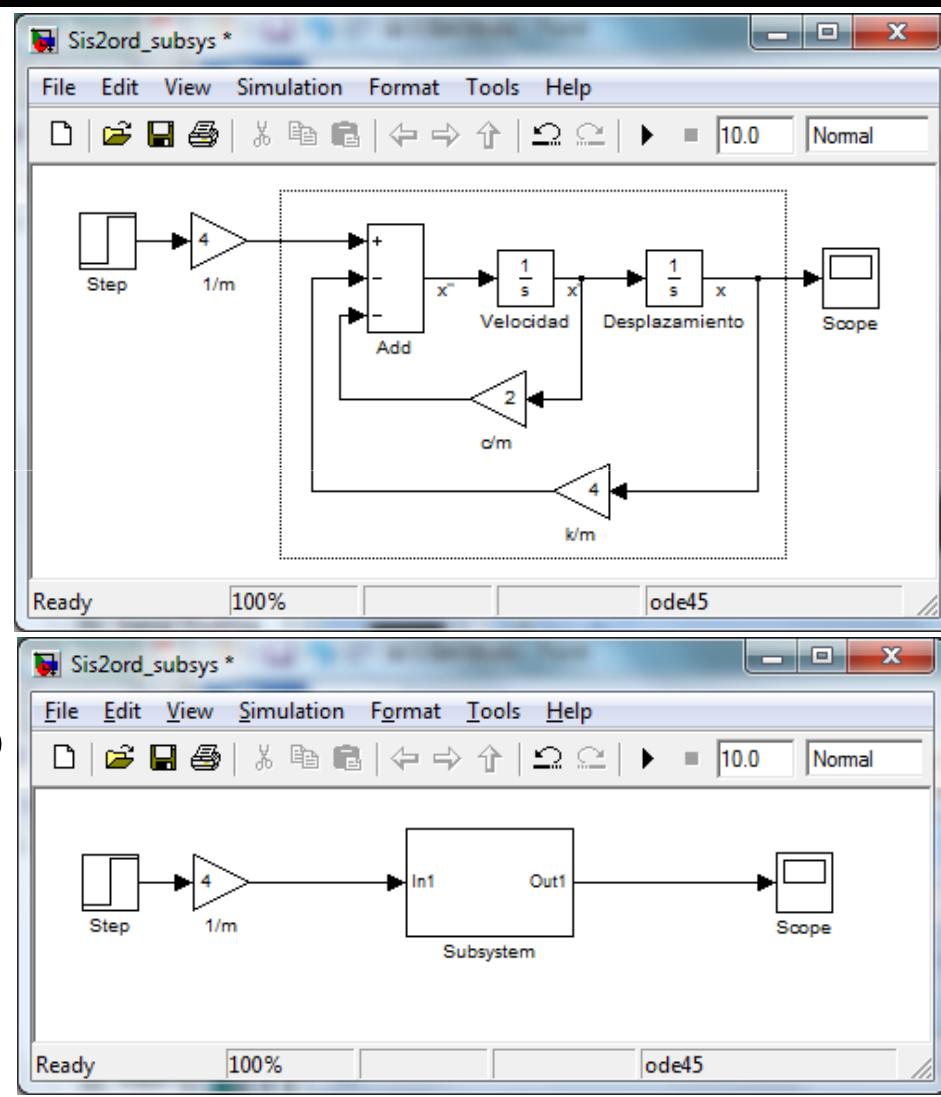


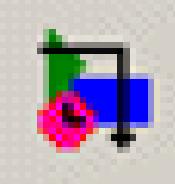
Creación de Subsistemas: Encapsulado

Seleccionar todos los bloques y señales a ser incluídos en el subsistema usando una caja envolvente

Seleccionar Edit>Create Subsystem del menu del modelo. Simulink reemplaza los bloques seleccionados por un bloque Subsystem con puertos de entrada y salida según las líneas de señales y les asigna un nombre por defecto

Cambiar el tamaño del bloque Subsystem para que las etiquetas de puertas sean legibles y mover para alinear



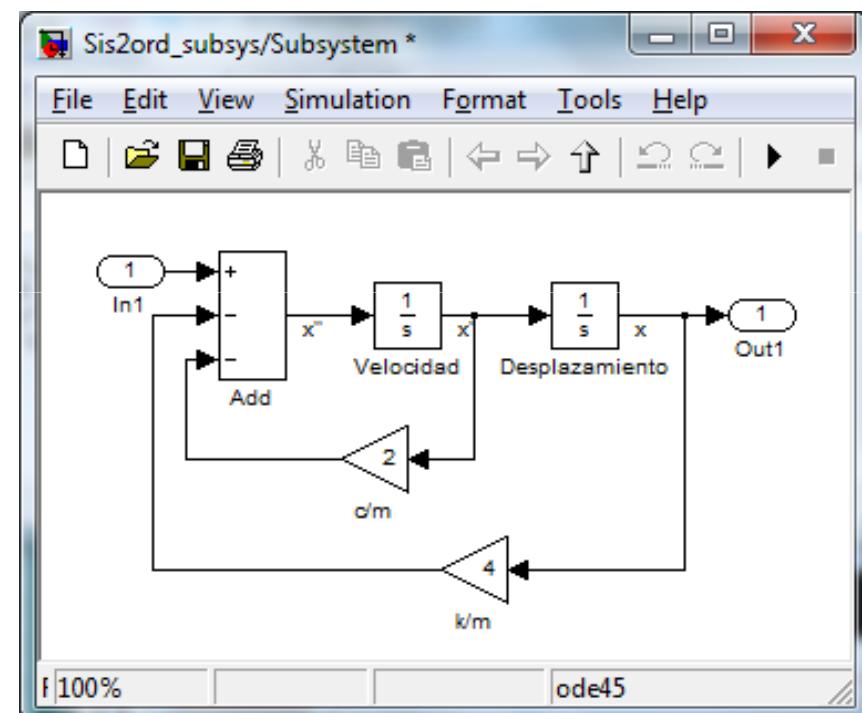


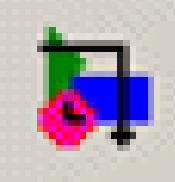
Creación de Subsistemas: Encapsulado

Para ver o editar el subsistema, hacer doble-click sobre el bloque. Aparecerá una nueva ventana conteniendo el subsistema.

Además del los bloques, se añade un bloque Import para la señal que entra al subsistema y un bloque Outport se añade para la señal que sale del subsistema. Si se cambia las etiquetas de esos puertos, cambia las etiquetas en el ícono del nuevo bloque.

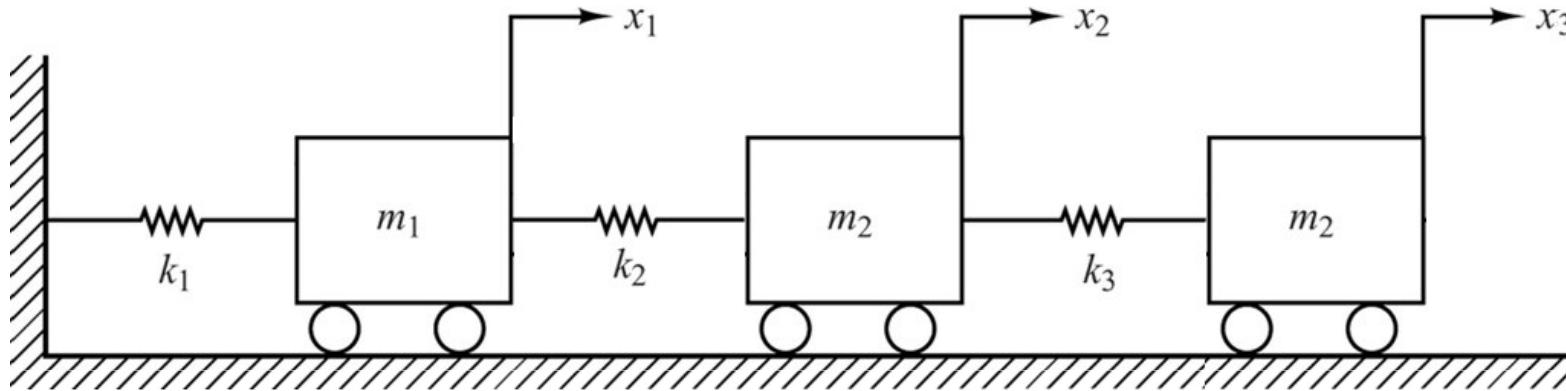
Nota: Una vez se crea el subsistema no hay operación inversa, por lo que se sugiere guardar el modelo antes de crear el subsistema.



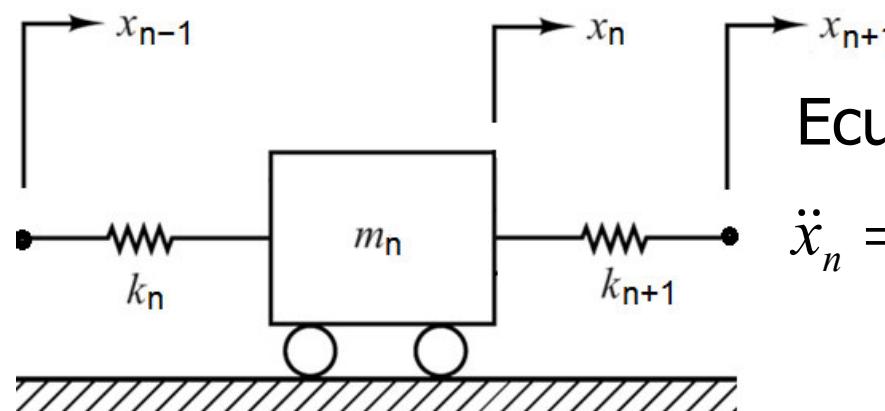


Creación de Subsistemas: bloque subsystem

- Se desea modelar un sistema de masa-resorte:

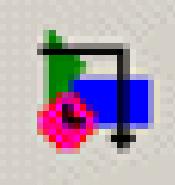


- El modelo de cada móvil es:



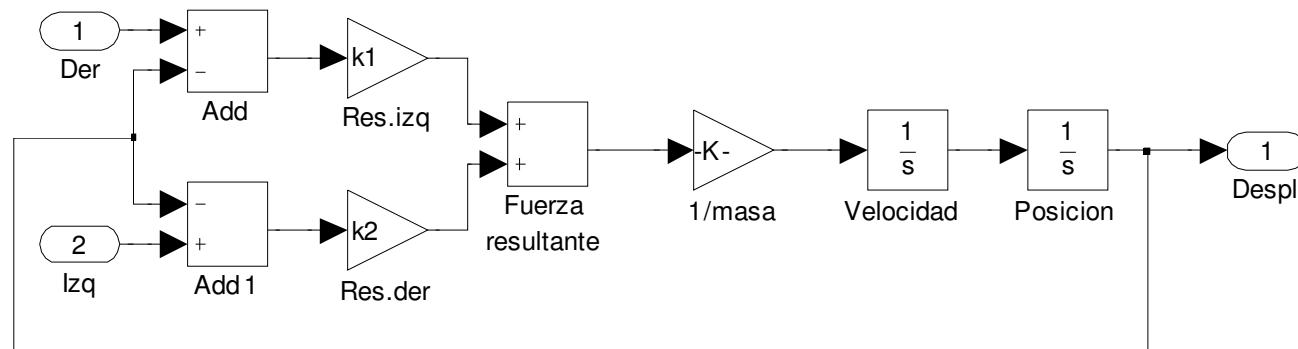
Ecuación para un móvil:

$$\ddot{x}_n = \frac{1}{m_n} [k_n(x_{n-1} - x_n) - k_{n+1}(x_n - x_{n+1})]$$

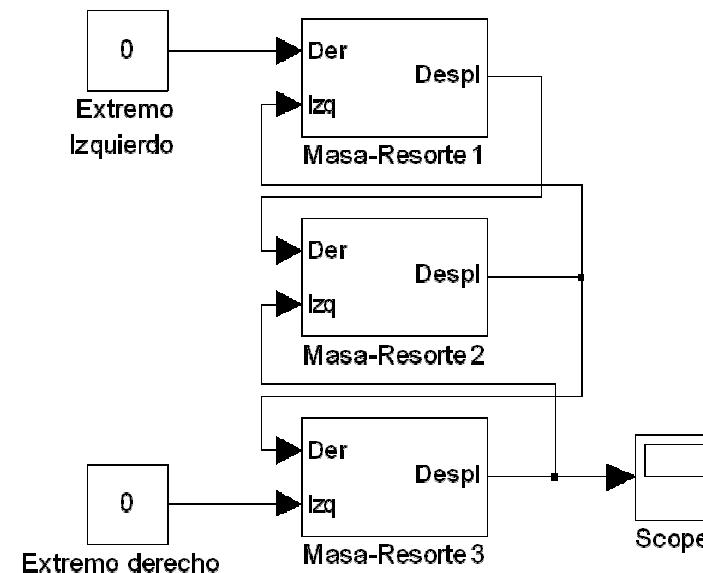


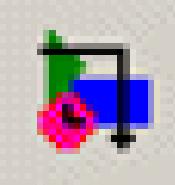
Creación de Subsistemas: bloque subsystem

Se construye el subsistema de un móvil según el modelo. La ganancia del bloque 1/masa se asigna 1/m1.



Se hacen dos copias del bloque subsystem y se conectan los bloques como se muestra





Creación de Subsistemas: bloque subsystem

Se asignan las constantes de los resortes y masas desde Matlab creando un fichero .m (pe setSubsys) y se ejecuta en Matlab

% Asignación de constantes para resortes y masas

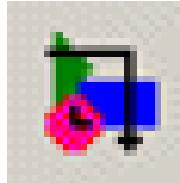
```
k1 = 1; k2 = 2; k3 = 4; m1 = 1; m2 = 3; m3 = 2;
```

Los parámetros de cada bloque subsystem se configura:

Móvil 1: valor del bloque Gain del bloque Resorte Izq a k1 y para el bloque Resorte Der a k2. Valor del bloque Gain del bloque 1/masa a 1/m1. Inicializar el bloque Integrator Velocidad a 0 y el Integrator Posicion a 1.

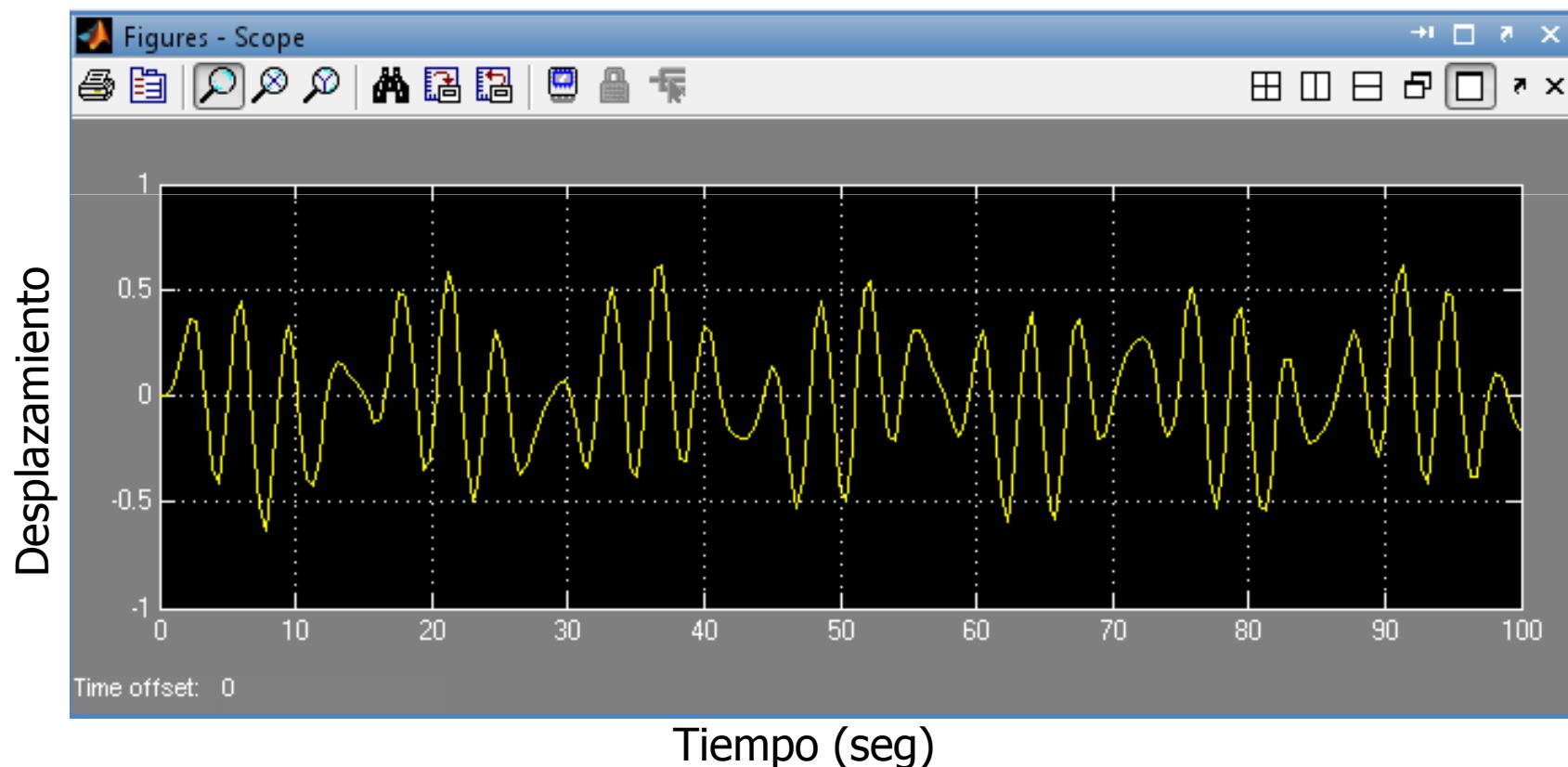
Móvil 2: valor del bloque Gain del bloque Resorte Izq a k2 y para el bloque Resorte Der a k3. Valor del bloque Gain del bloque 1/masa a 1/m2. Inicializar el bloque Integrator Velocidad a 0 y el Integrator Posicion a 0.

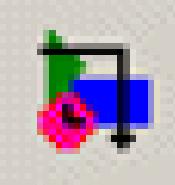
Móvil 3: valor del bloque Gain del bloque Resorte Izq a k3 y para el bloque Resorte Der a 0. Valor del bloque Gain del bloque 1/masa a 1/m3. Inicializar el bloque Integrator Velocidad a 0 y el Integrator Posicion a 0.



Creación de Subsistemas: bloque subsystem

Se configura el bloque Scope para almacenar los datos de salida al workspace. Asignar Start time a 0 y Stop time a 100. Se puede obtener la gráfica desde Matlab con el comando simplot. Después de ejecutar las simulación, se obtiene:

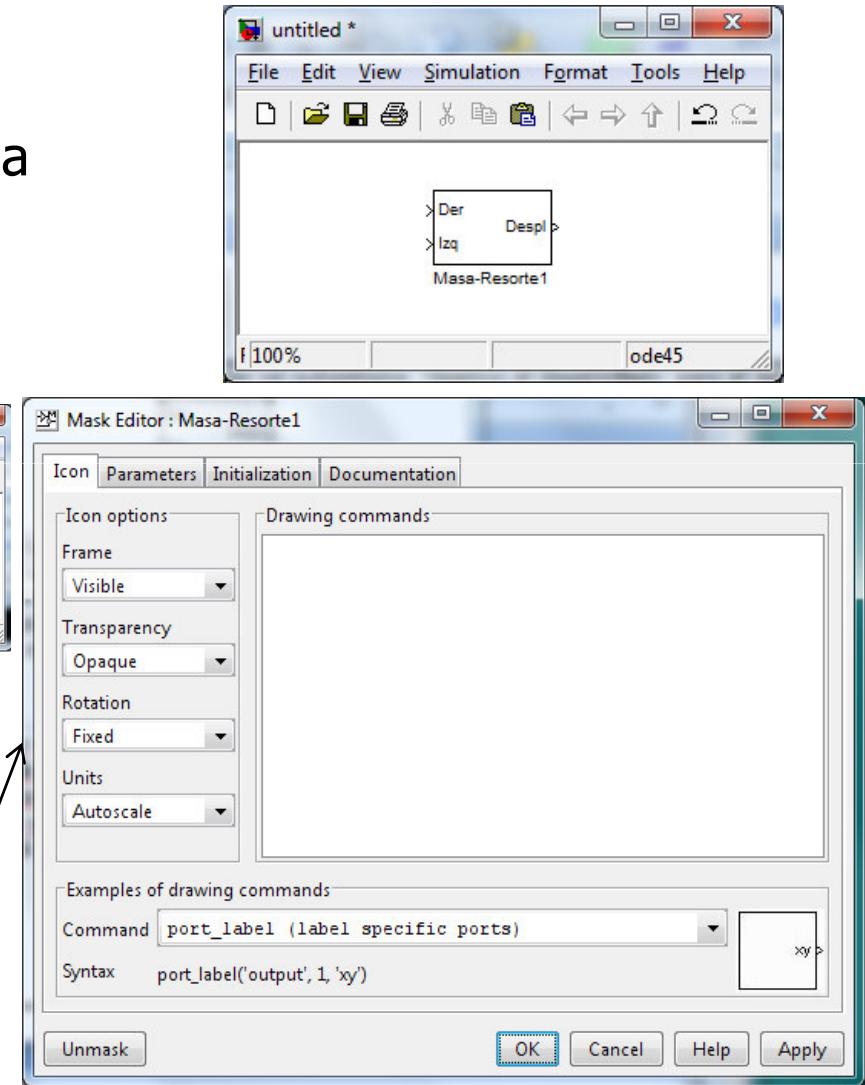
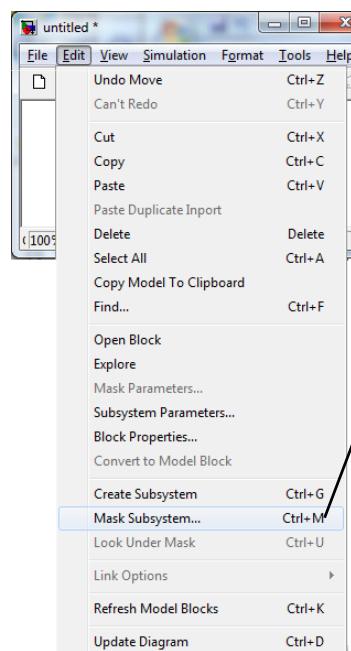


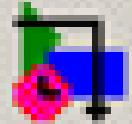


Ejemplo: enmascarado de bloques

Se crea un subsistema. Usamos el desarrollado para el sistema masa-resorte copiando en una nueva ventana de modelo el bloque Masa-Resorte1

Seleccionar el bloque y se escoje **Edit:Mask Subsystem** en el menú de la ventana de modelos para obtener la ventana de diálogo del editor Mask. Guardar el modelo (CartMask)

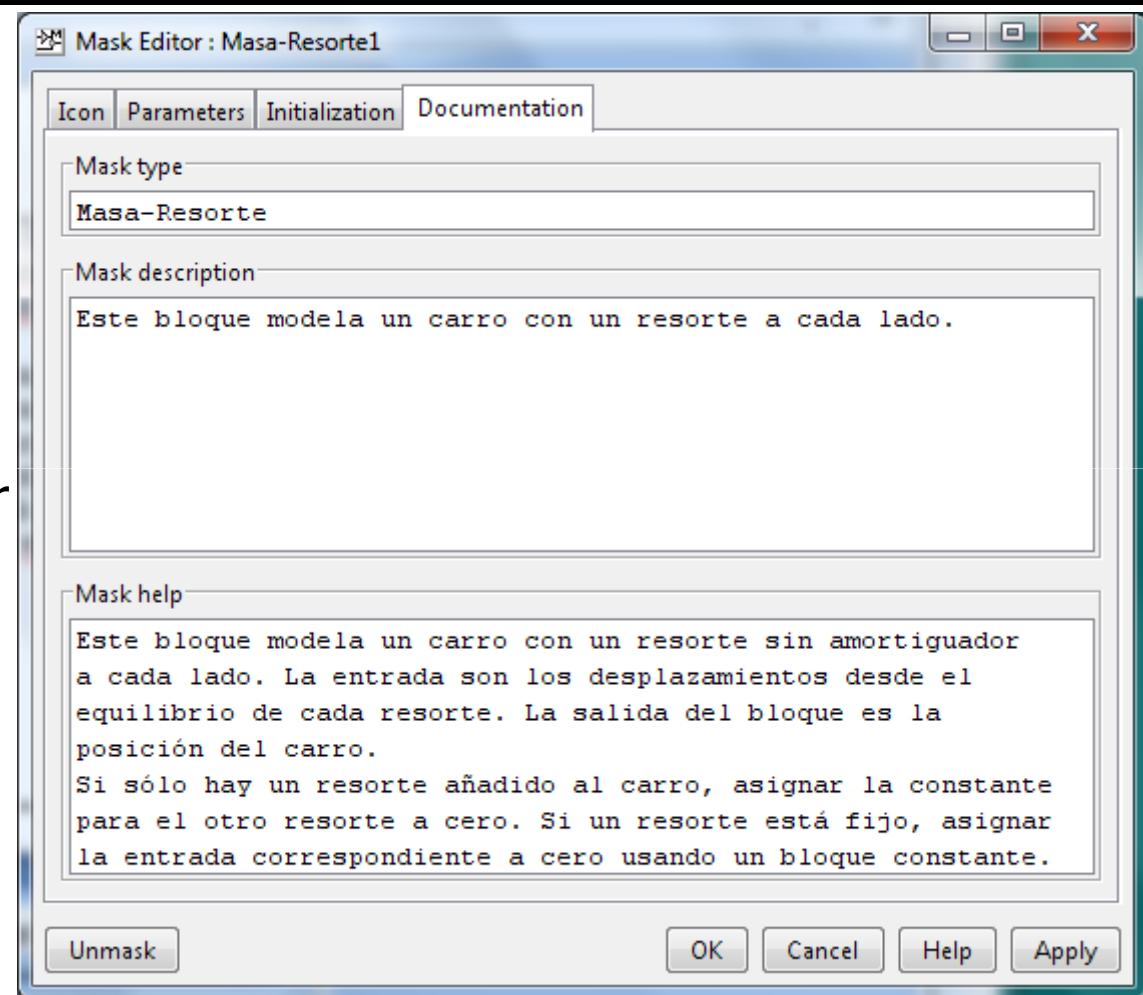


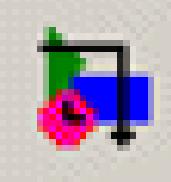


Ejemplo: enmascarado de bloques

El editor Mask tiene cuatro pestañas: Icon, Parameters, Initialization, Documentation.

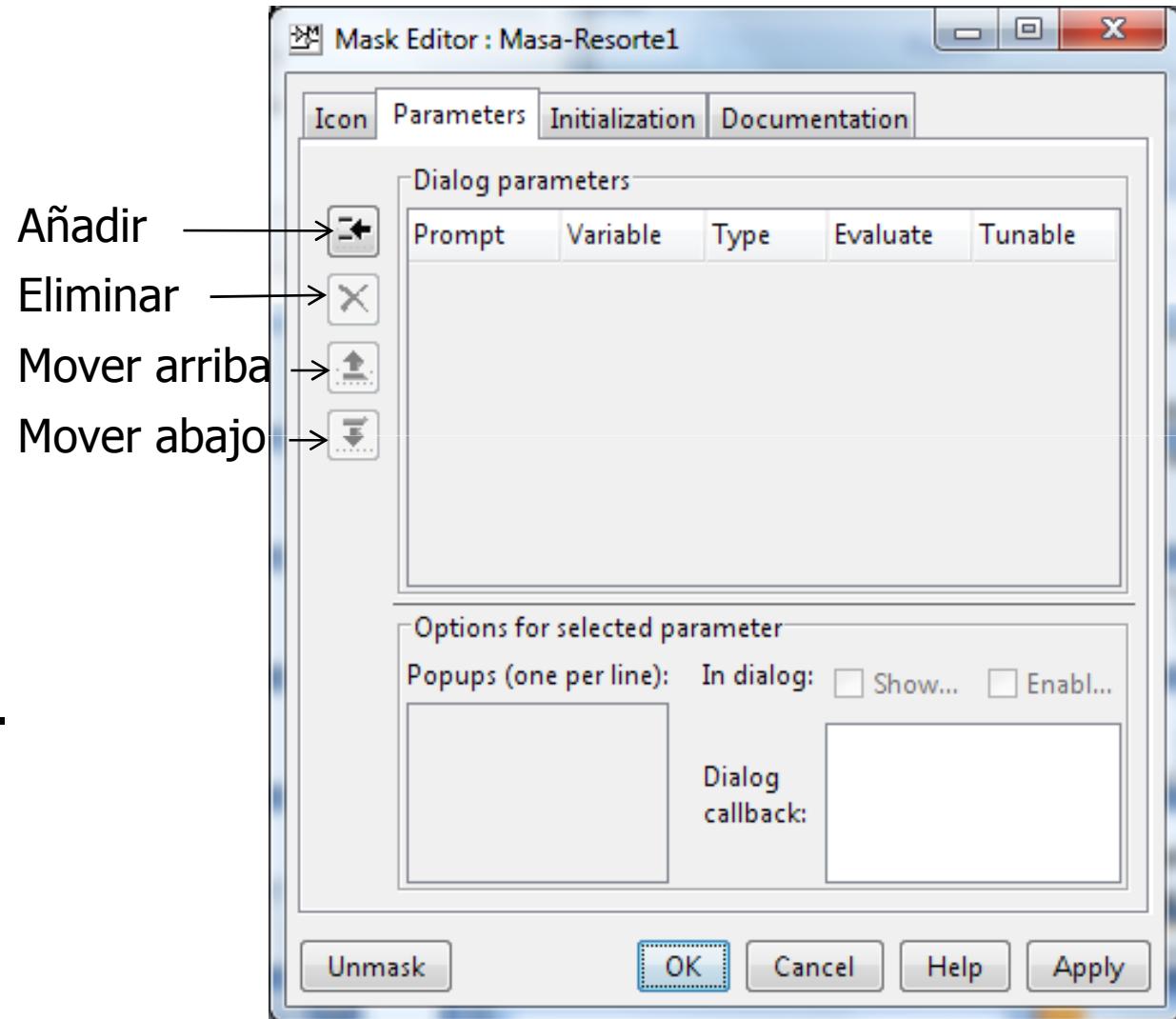
La página Documentation del editor sirve para establecer un tipo , colocar una descripción del bloque y una descripción más detallada que se mostrará cuando se selecciona el sistema Help de Matlab

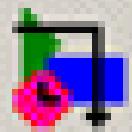




Ejemplo: enmascarado de bloques

La página Parameters del editor Mask se usa para definir parámetros para el bloque del subsistema. Se divide en dos secciones: Una superior en la que se define los campos de diálogo y el orden en que se muestran, asociando una variable Matlab con cada campo. La sección inferior contiene opciones para cada campo definido en la parte superior.

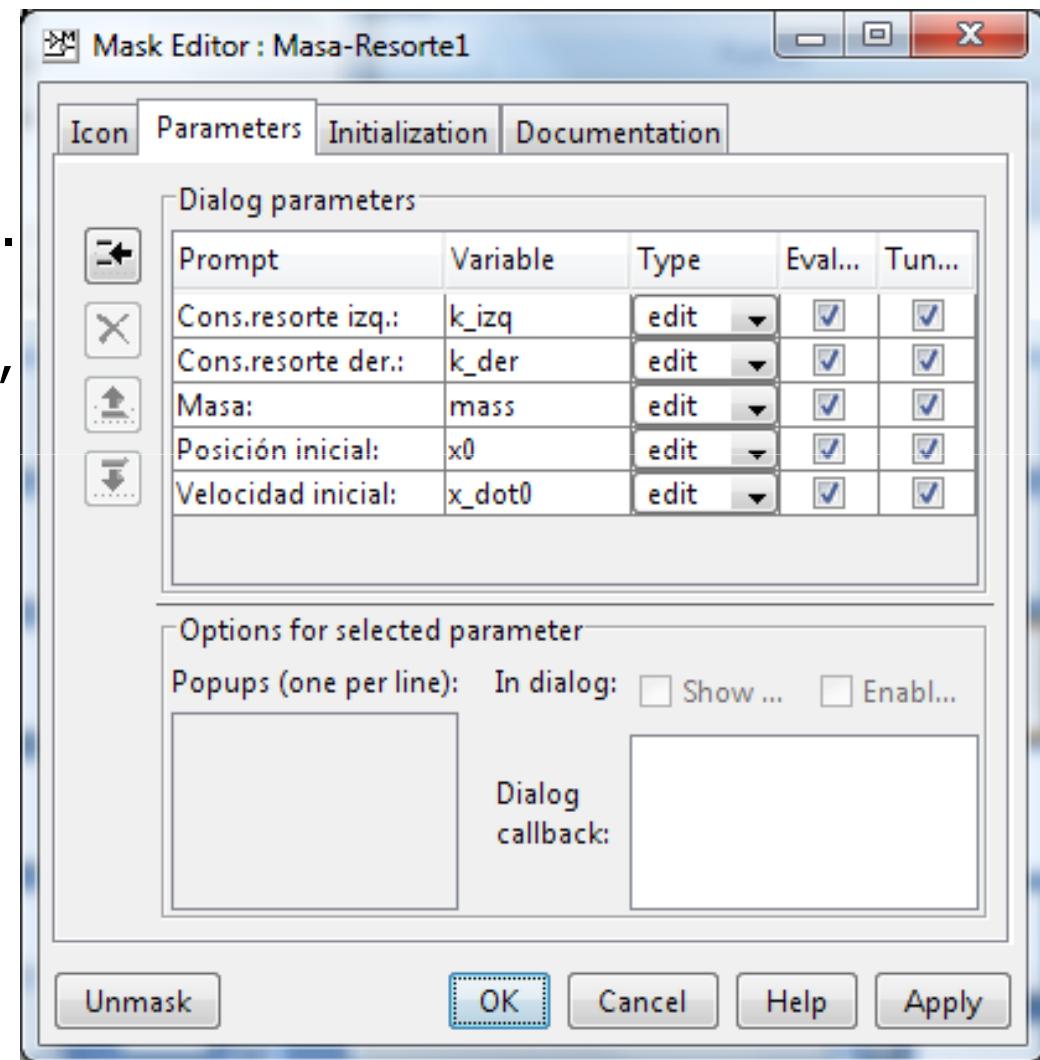
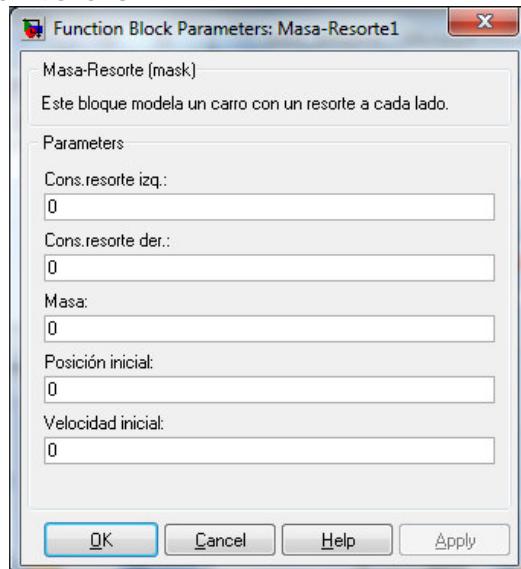


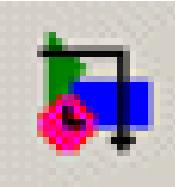


Ejemplo: enmascarado de bloques

La pestaña Parameters se usa para crear, editar y eliminar campos de diálogo asociado a variables Matlab con cada campo. Para cada campo añadido se asocia un Prompt, Variable, Type, Evaluate y Tunable.

Resultado:



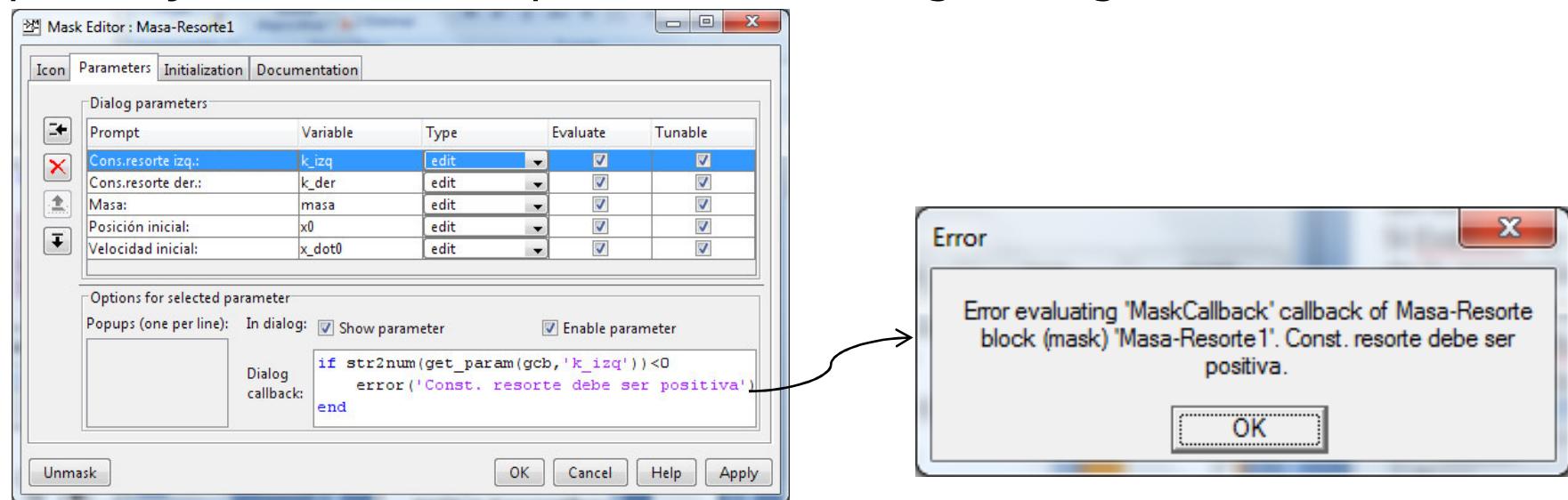


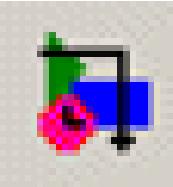
Ejemplo: enmascarado de bloques

El campo Type en Dialog parameters puede ser: edit, Checkbox y Popups . El valor asignado a la variable asociada con un campo de diálogo depende del estado de si selecciona el campo Evaluate.

Si Evaluate se selecciona la variable asociada con el campo tendrá el valor de la expresión del campo. El campo Tunable determina si el parámetro se puede modificar durante la ejecución de la simulación.

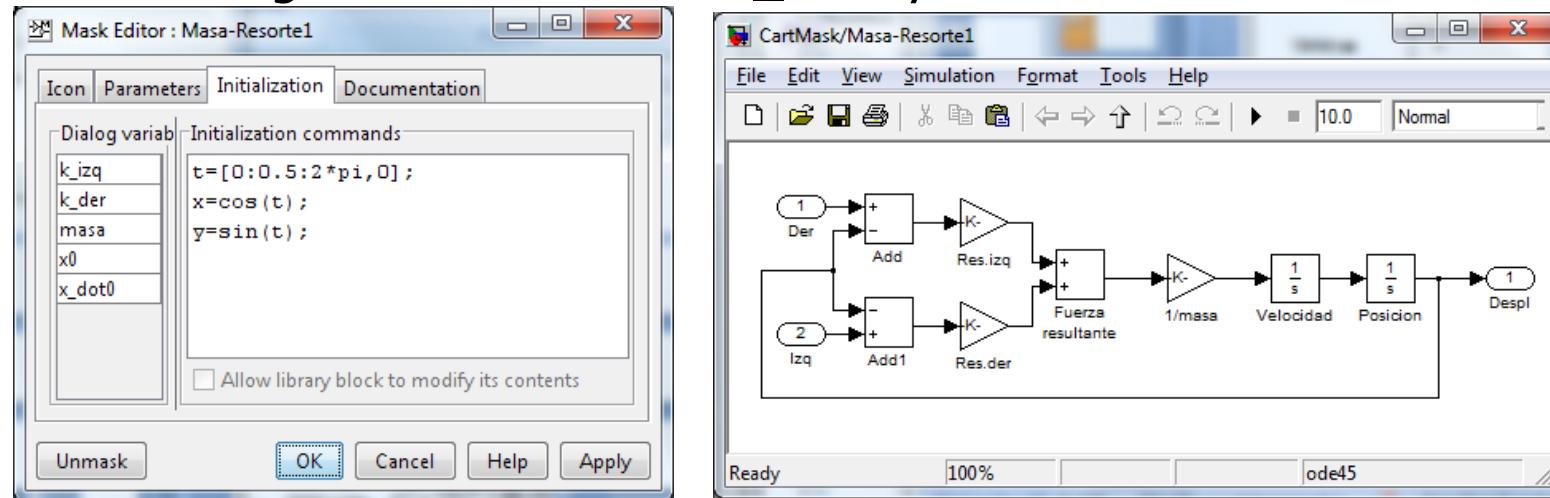
El campo Callback permite asociar con el parámetro un bloque de código que se ejecuta cuando el parámetro de diálogo se ingresa.

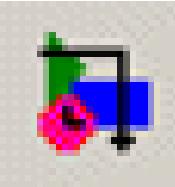




Ejemplo: enmascarado de bloques

La página Initialization proporciona una lista de variables asociados con los parámetros del bloque y un campo de Initialization commands. Los bloques del subsistema se debe configurar para usar las variables definidas en las páginas Initialization y Parameters. Para ello se selecciona el subsistema y se escoge **Edit:Look Under Mask** en el menú de la ventana de modelo. Hacer doble click en el bloque Gain con etiqueta Res.Izq. Y asignar Gain a k_izq. Repetir con Res.Der. Para asignar Gain a k_der. Lo mismo con 1/mass para asignar Gain a masa. Asignar Initial condition del integrador Velocidad a x_dot0 y en Posicion a x0.

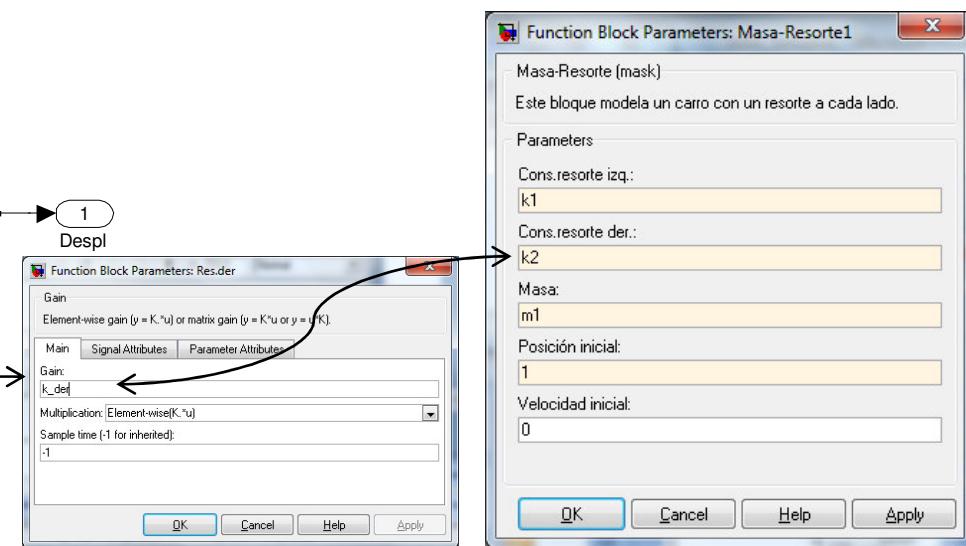
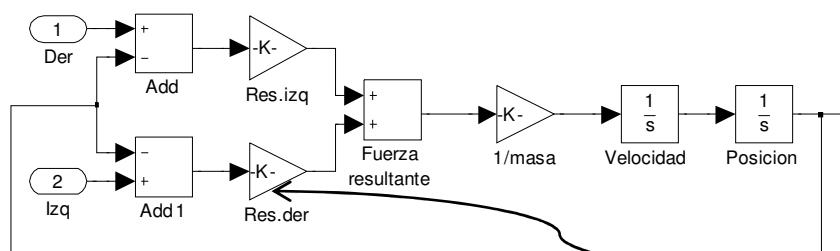


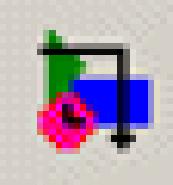


Ejemplo: enmascarado de bloques

Las variables definidas en la máscara de un subsistema son locales y no pueden acceder a las variables del workspace de Matlab. Un campo de entrada en el diálogo de un bloque enmascarado puede contener constantes o expresiones usando variables definidas en el workspace de Matlab. El valor del contenido del campo de entrada se asigna a la variable interna del subsistema enmascarado asociado con el campo de entrada. Esta variable se puede usar para inicializar el bloque o puede usarse para definir otras variables internas definidas en el campo

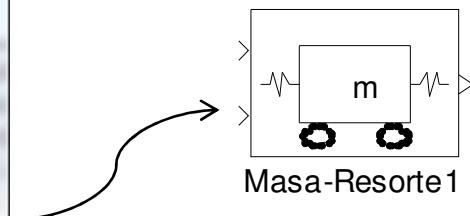
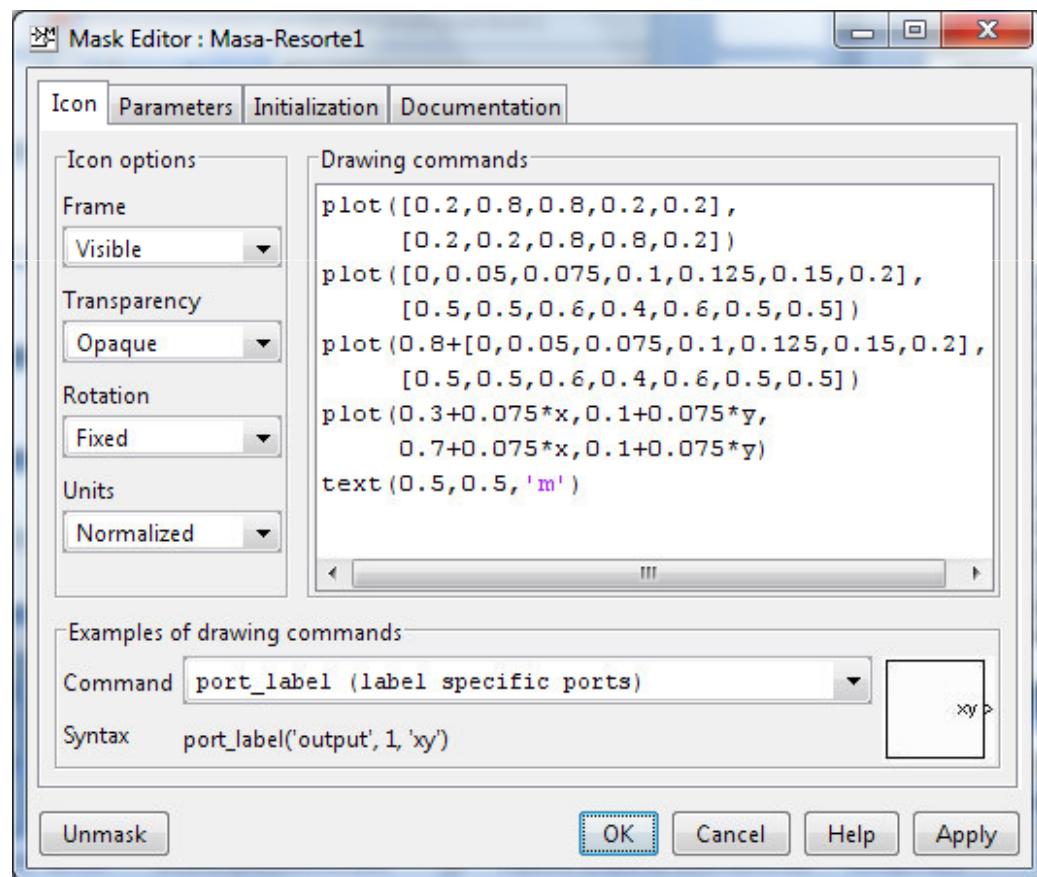
Initialization commands



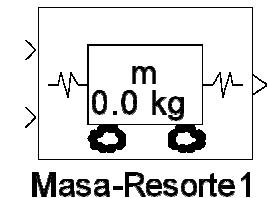


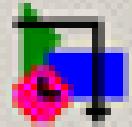
Ejemplo: enmascarado de bloques

La página Icon permite diseñar iconos propios para los bloques enmascarados. Consiste de seis campos. En el campo Drawing commands se puede poner comandos Matlab para dibujar y etiquetar el ícono.



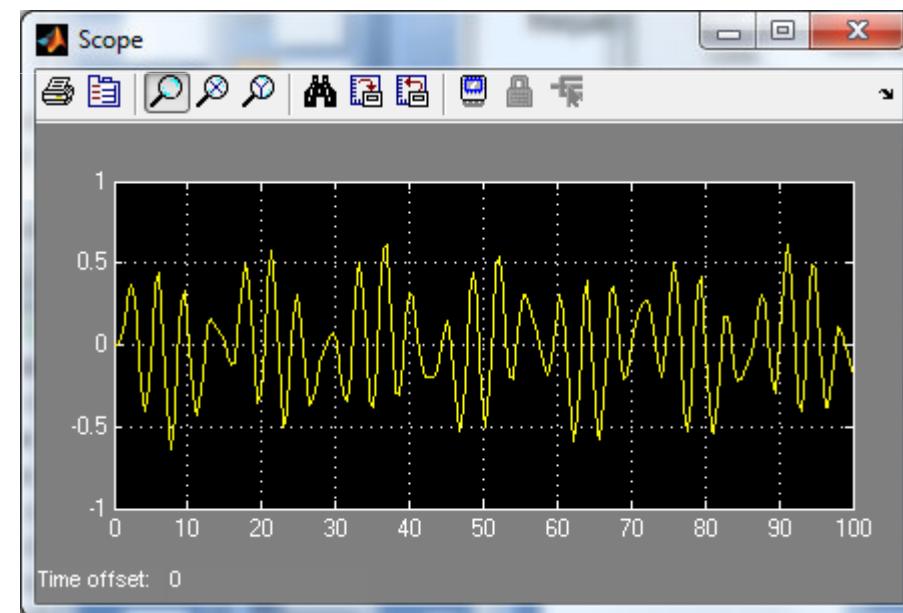
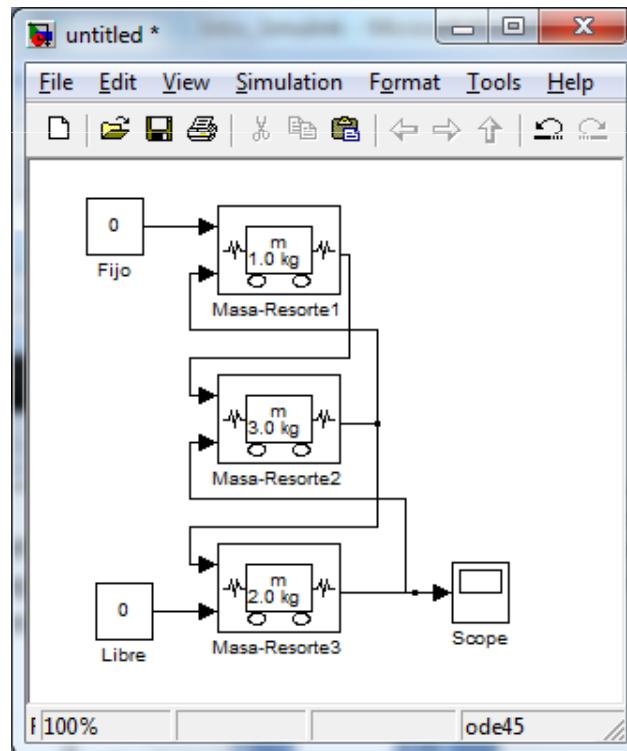
Si en la página Initialization se agrega la línea:
`m_etiq=sprintf('%1.1f kg',mass);`
y en Drawing Commands:
`text(0.45,0.6,'m')`
`text(0.25,0.4,m_etiq)`

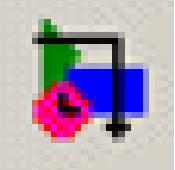




Ejemplo: Uso de bloque enmascarado

Una vez que se crea el bloque enmascarado se puede copiar a una ventana de modelo de forma idéntica cuando se copia de la librería de bloques de Simulink. Para el ejemplo se arrastra tres copias del bloque Masa-Resorte, se conecta y se asignan los parámetros con un fichero .m. El modelo así queda completo.





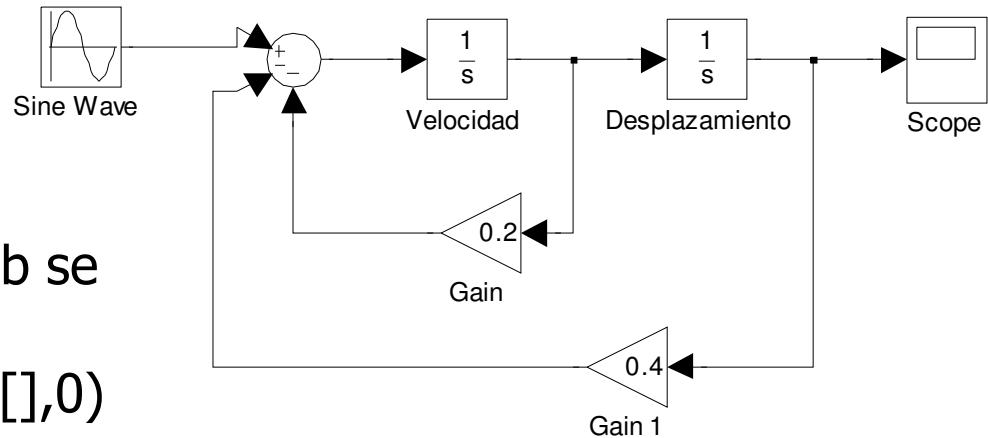
Ejemplo: Vector de estado del modelo

Se tiene el siguiente modelo Simulink guardado en el fichero llamado ejmod.mdl

En la ventana de comandos Matlab se ejecuta el comando:

```
>> [sizes,x0,states]=ejmod([],[],[],0)  
sizes =
```

```
2  
0  
0  
0  
0  
0  
1
```

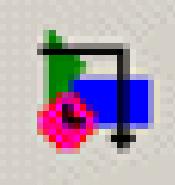


x0 =

```
0  
0
```

states =

```
'ejmod/Desplazamiento'  
'ejmod/Velocidad'
```

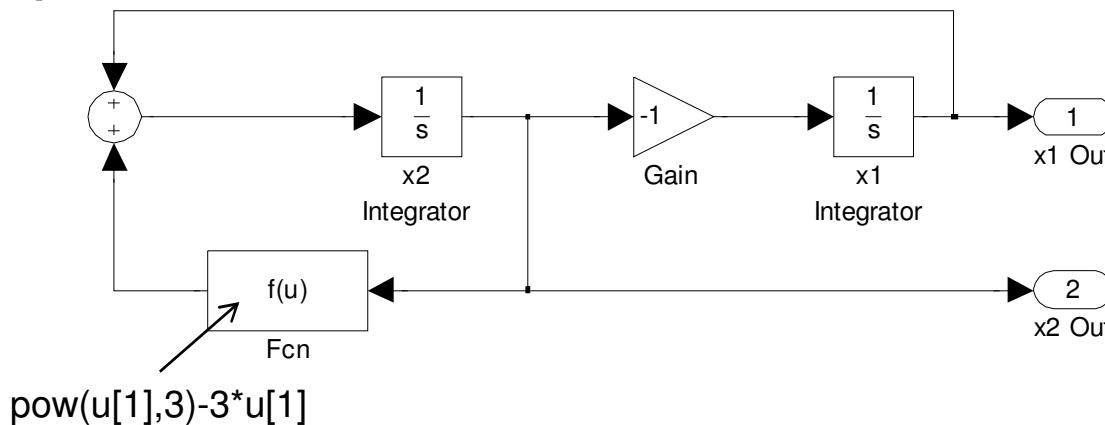


Ejemplo: linealización de un modelo

Se tiene el siguiente sistema no lineal: $\dot{x}_1 = -x_2$

$$\dot{x}_2 = x_1 + x_2^3 - 3x_2$$

cuyo modelo (nolinear_mod.mdl) en Simulink es:



En la ventana de comandos Matlab se ejecutan los comandos:

```
>> [A,B,C,D]=linmod('nolinear_mod');  
>> disp(A)
```

```
0 -1.0000
```

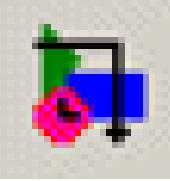
```
1.0000 -3.0000
```

```
>> disp(eig(A))
```

```
-0.3820
```

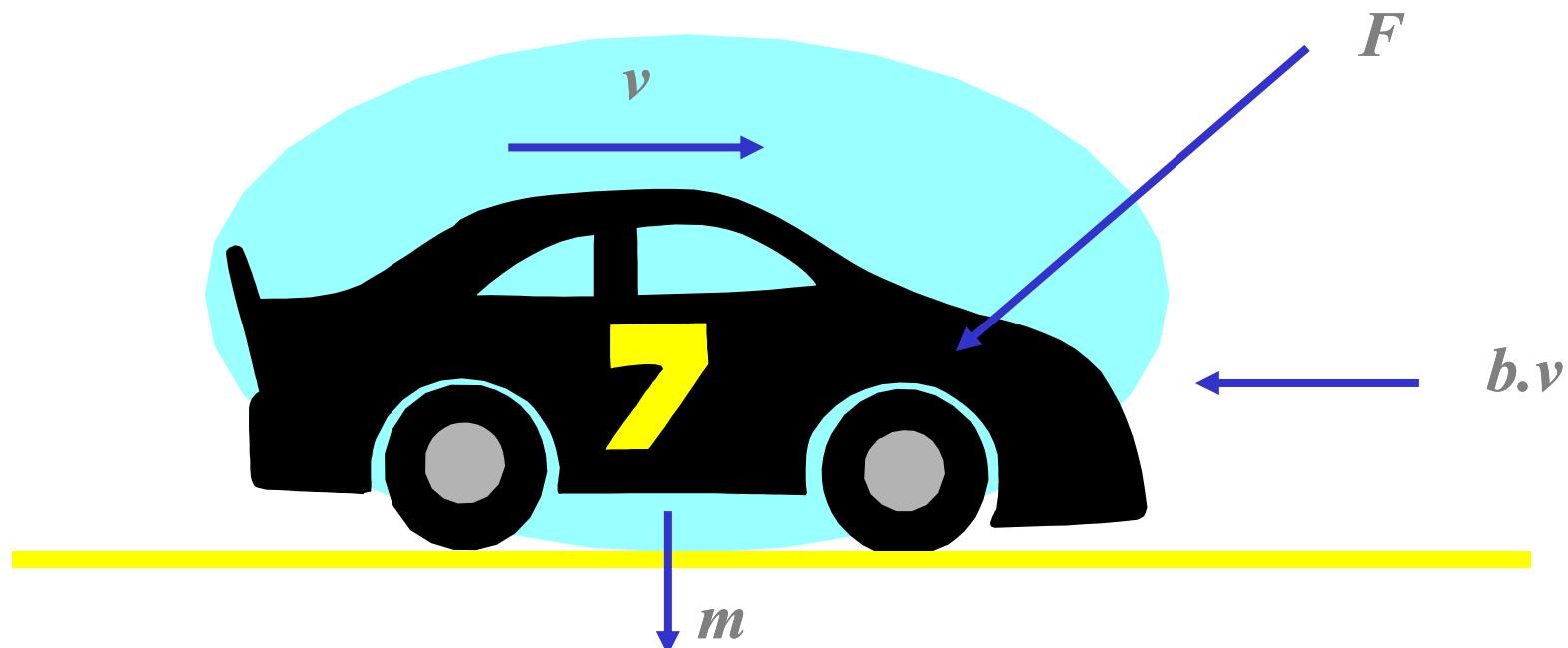
```
-2.6180
```

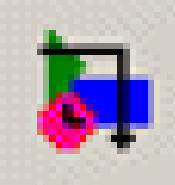
} indica que el sistema es estable



Movimiento de un vehículo

- Considérese un coche conducido en línea recta a lo largo de una carretera plana
- La ecuación de movimiento se deduce a partir de los siguientes factores:





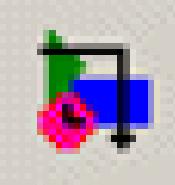
Modelo del vehículo

- Ecuaciones:

$$F = m \cdot dv/dt + b \cdot v$$

donde:

- F = fuerza proporcionada por el motor
- m = masa del vehículo
- dv/dt = tasa de cambio de velocidad (aceleración)
- b = factor de amortiguamiento (resistencia del viento)
- v = velocidad (velocidad del vehículo)



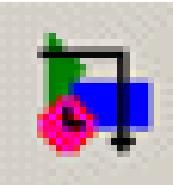
Modelo del vehículo: Simulink

- Ecuaciones:

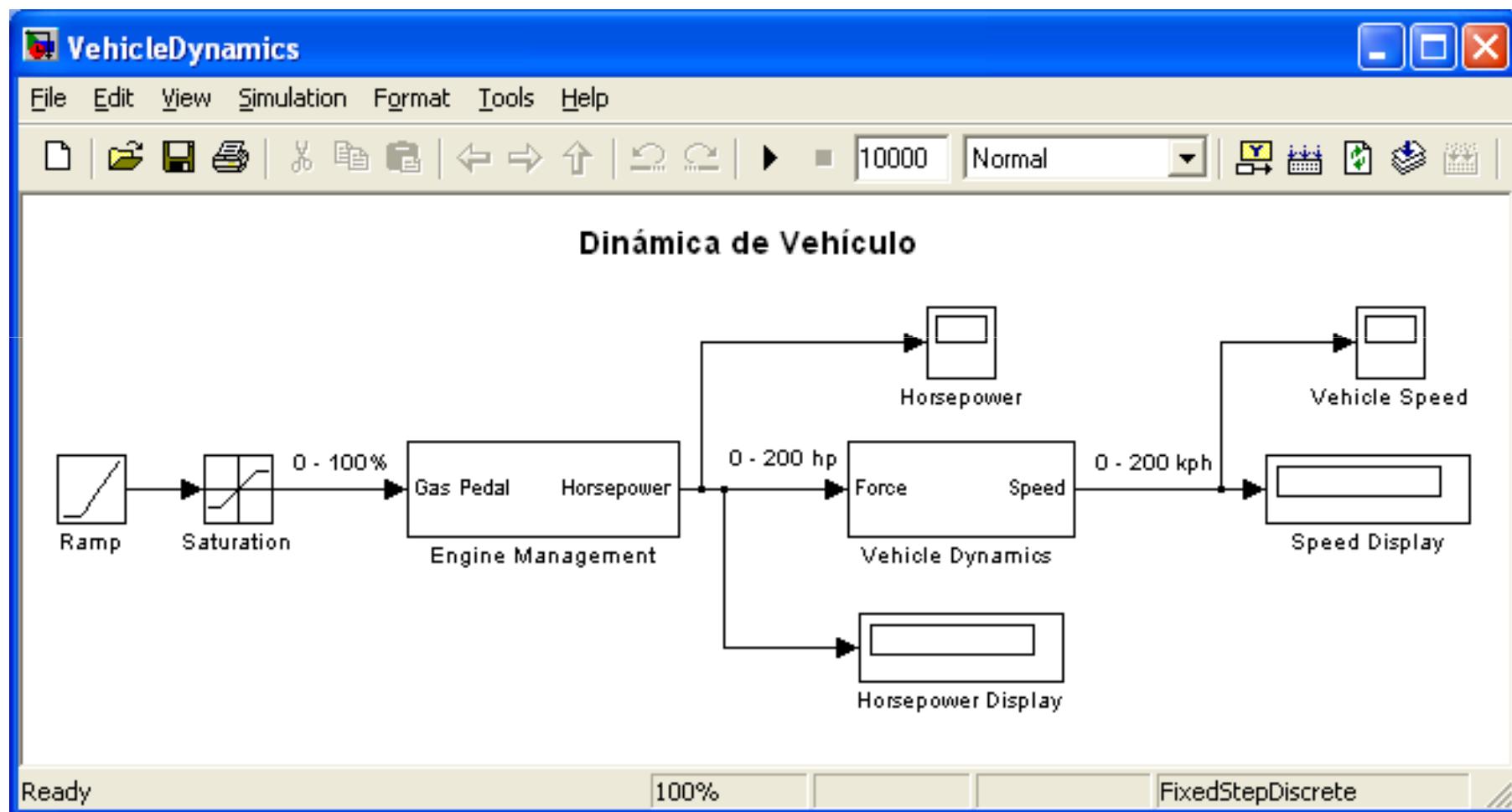
$$F/m - v \cdot b/m = dv/dt$$

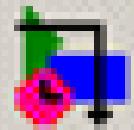
- Subsistemas:

- La entrada del sistema es el control de combustible (pedal de aceleración), bajo control del conductor
- El subsistema “Engine Management” convierte la posición del pedal a potencia del motor
- El subsistema “Vehicle Dynamics” convierte la potencia del motor a velocidad del vehículo
- La salida se suministra en horsepower (hp)



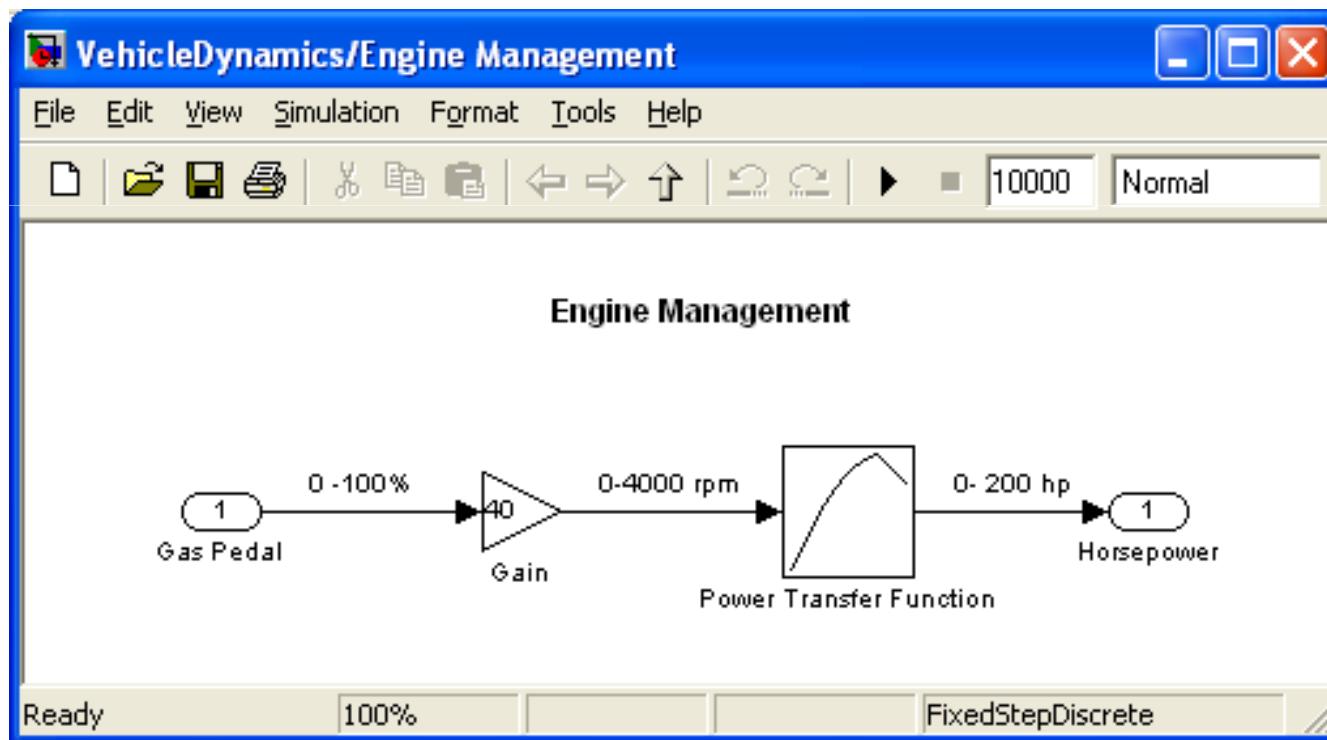
Modelo del vehículo: Simulink

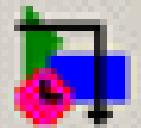




Subsistema Engine Management

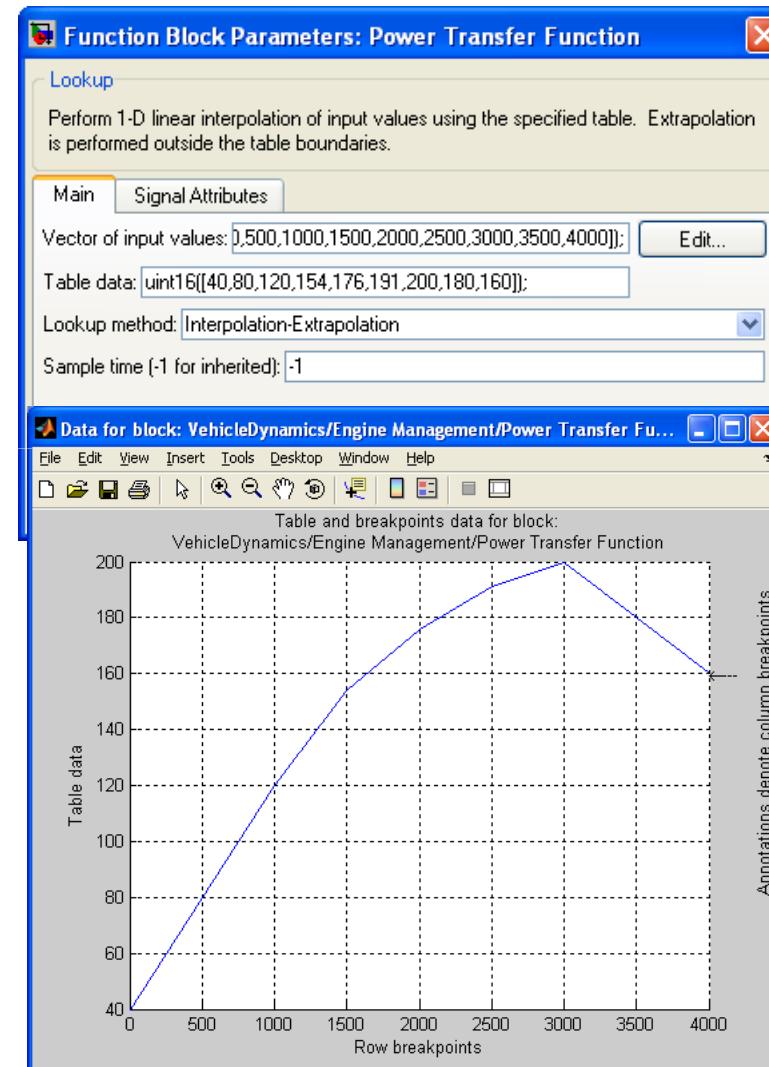
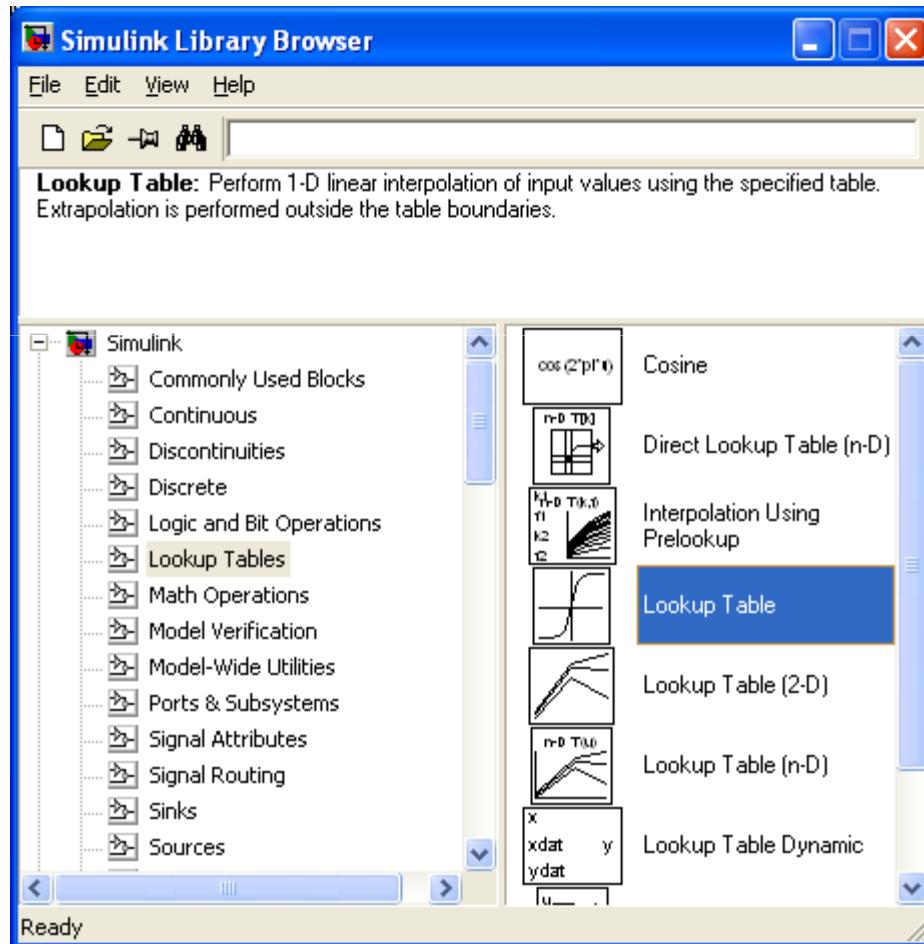
- Convierte la entrada del pedal de combustible (0-100%) a potencia del motor (0 – 200 hp)

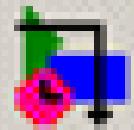




Tablas de interpolación (Lookup Tables)

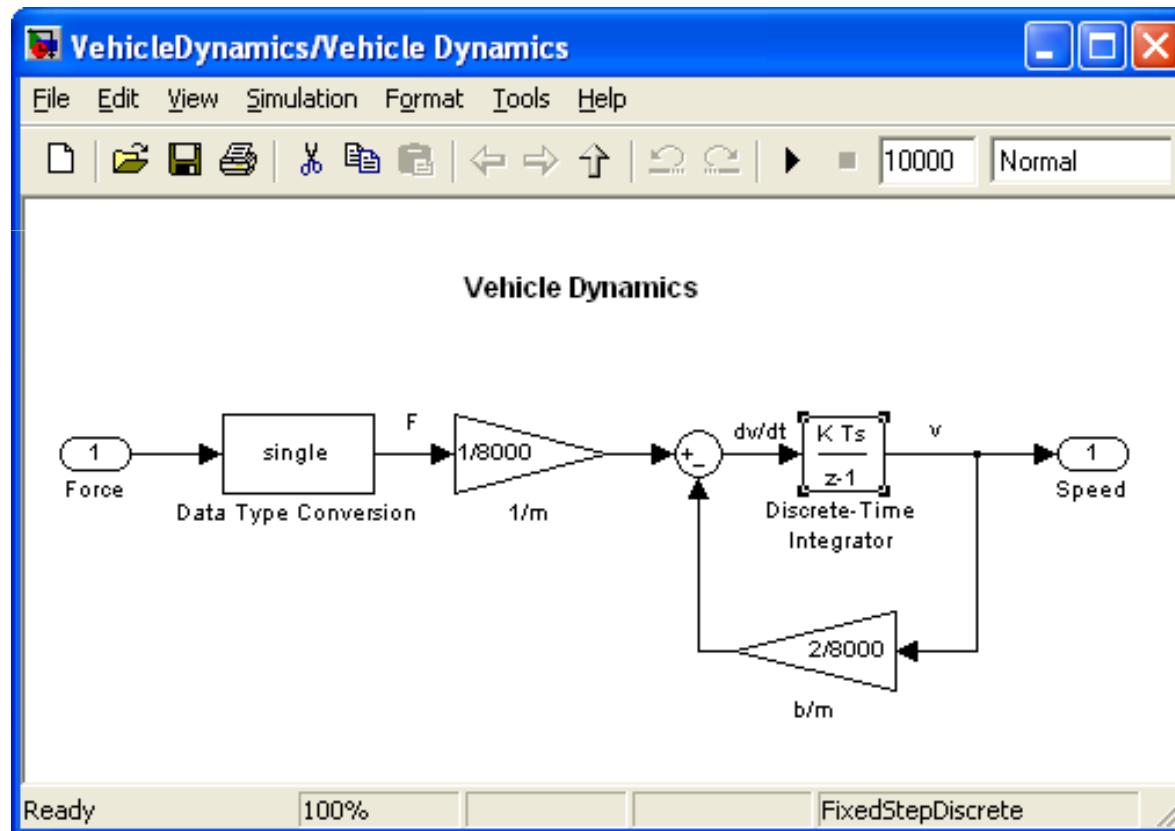
- Convierte rpm a potencia

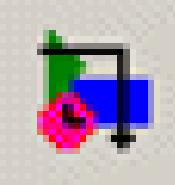




Subsistema Vehicle Dynamics

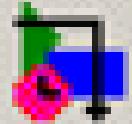
- Convierte la entrada del pedal de combustible (0-100%) a potencia del motor (0 – 200 hp)





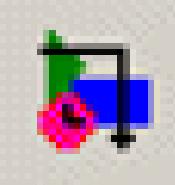
Modelo del vehículo: Ajustes

- Cambiar la masa m del vehículo entre 1 ton (coche pequeño) y 35 tons (camión).
 - Incrementar la resistencia del aire incrementando la variable b .
 - Usando datos reales del fabricante para la Lookup Table. Se puede modelar también motores diesel
 - Reemplazar la entrada Ramp con una entrada Step para simular cambios bruscos en el pedal de aceleración
-



Cancelación de ruido acústico (CRA)

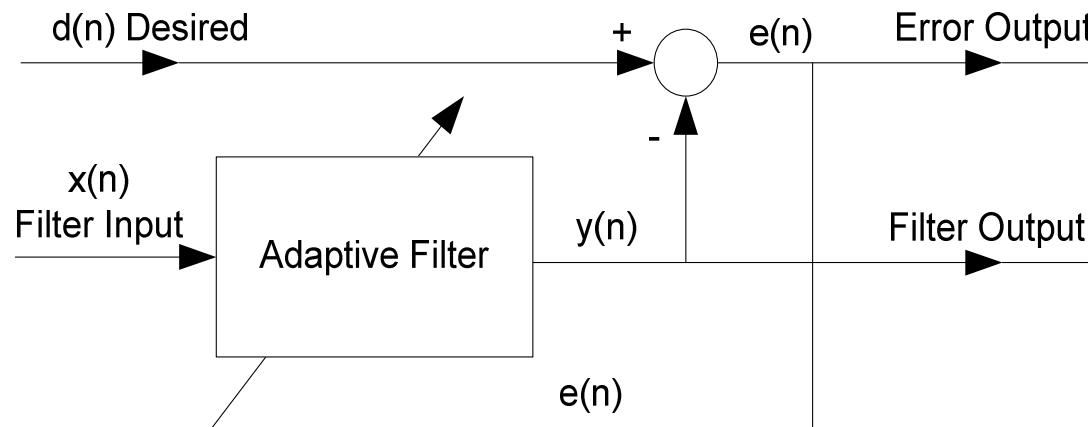
- Desarrollar el modelo Simulink de un sistema de reducción de ruido utilizando el algoritmo de mínimo cuadrados medio (Least Mean Squares, LMS)
- En audio “ruido” se refiere a cualquier señal junto a las que se desean escuchar
- Se usa un filtro adaptativo



CRA: Filtro adaptativo

- Diagrama de bloques

Adaptive Filter Block Diagram

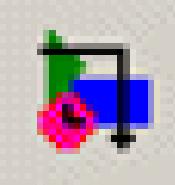


$$e(n) = d(n) - y(n)$$

- Ecuación

– Es un filtro tipo FIR con N coeficientes variables w

$$y(n) = \sum_{k=0}^N w_k(n) \cdot x(n-k)$$

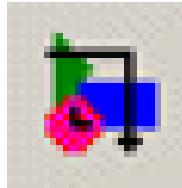


CRA: La ecuación LMS

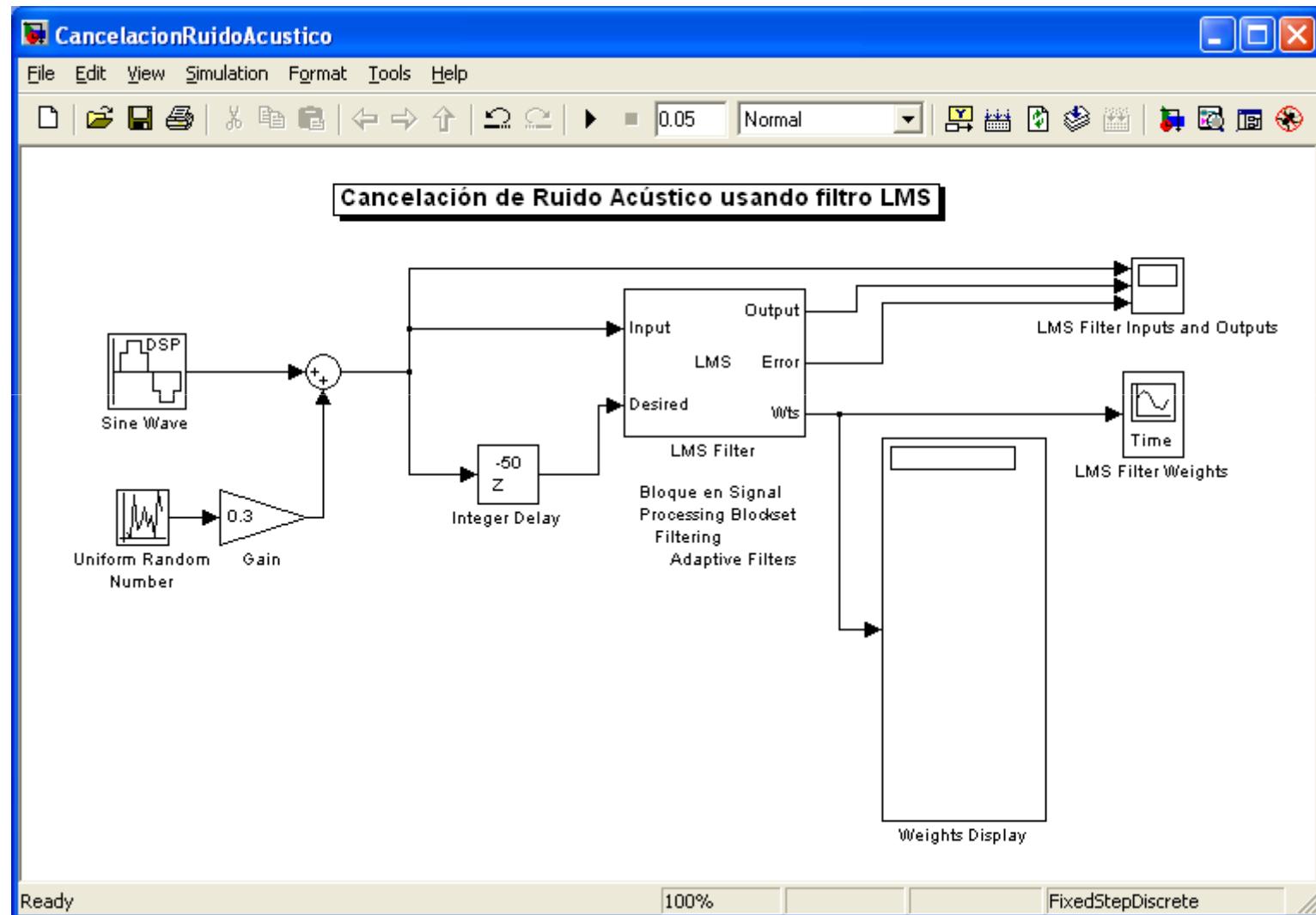
- El algoritmo Least Mean Squares (LMS) actualiza cada coeficiente sobre la base de cada muestra y basado en el error $e(n)$

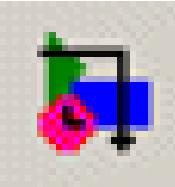
$$w_k(n+1) = w_k(n) + \mu \cdot e(n)x_k(n)$$

- Esta ecuación minimiza la *potencia* del error $e(n)$
- El valor de μ (mu) es crítico (compromiso)
 - Si μ es demasiado bajo, el filtro reacciona lentamente
 - Si μ es muy grande, la resolución del filtro es pobre



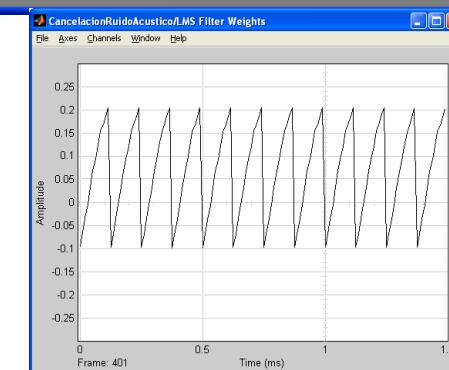
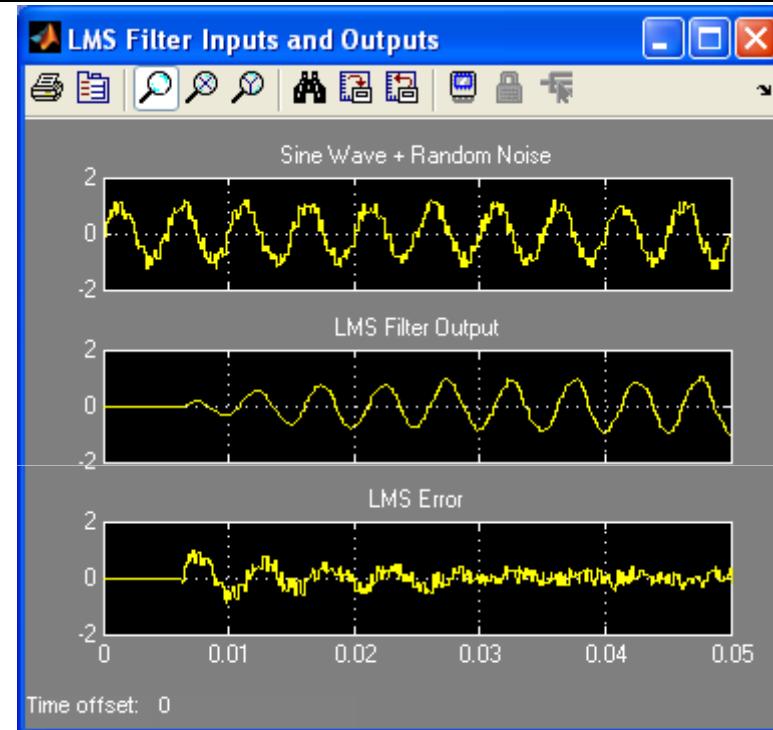
CRA: Modelo Simulink

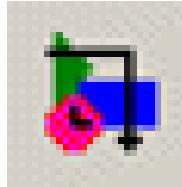




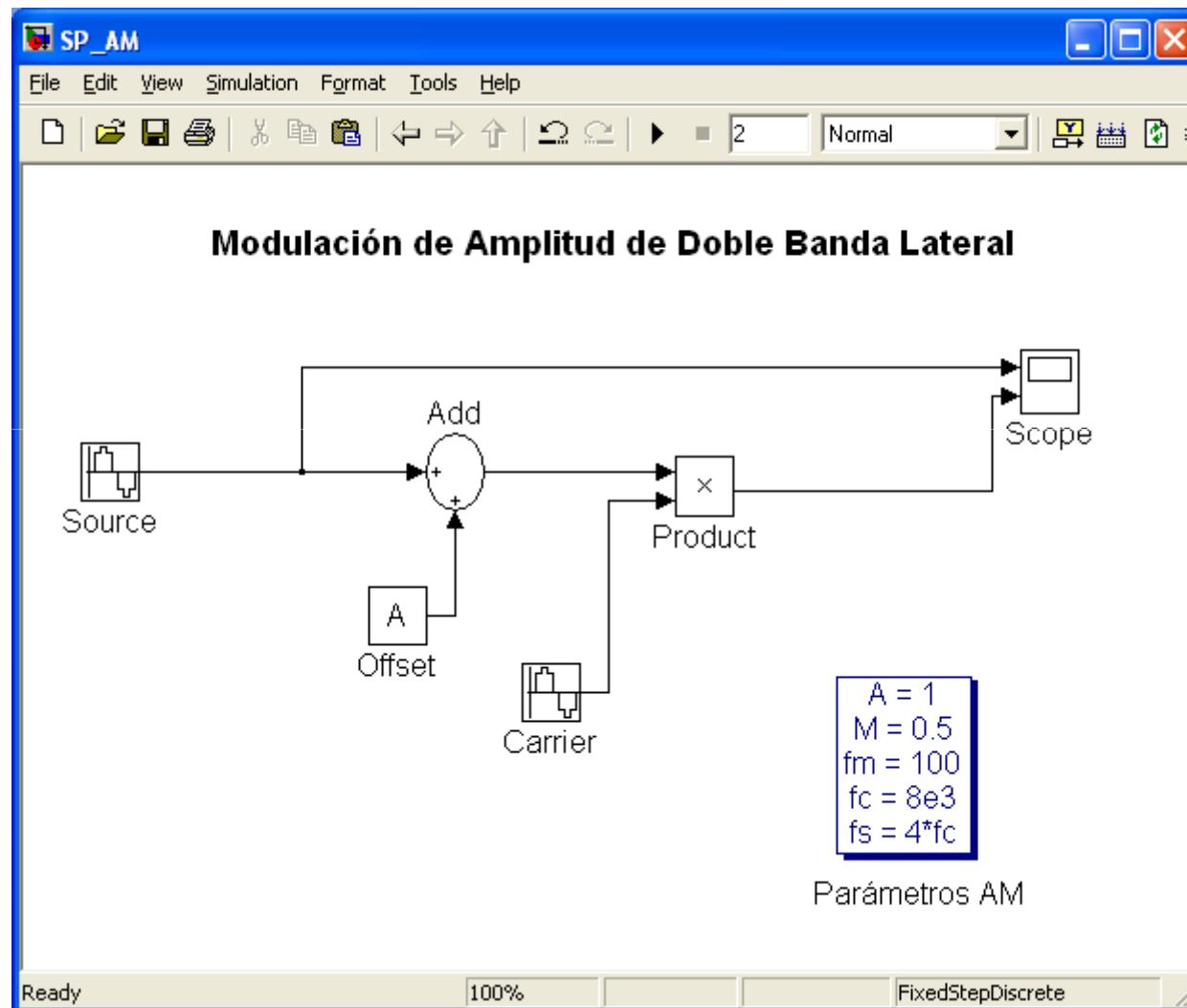
CRA: Entrada/Salidas

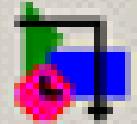
- Entrada: Señal + Ruido
- Salida
- Error
- Evolución de los pesos



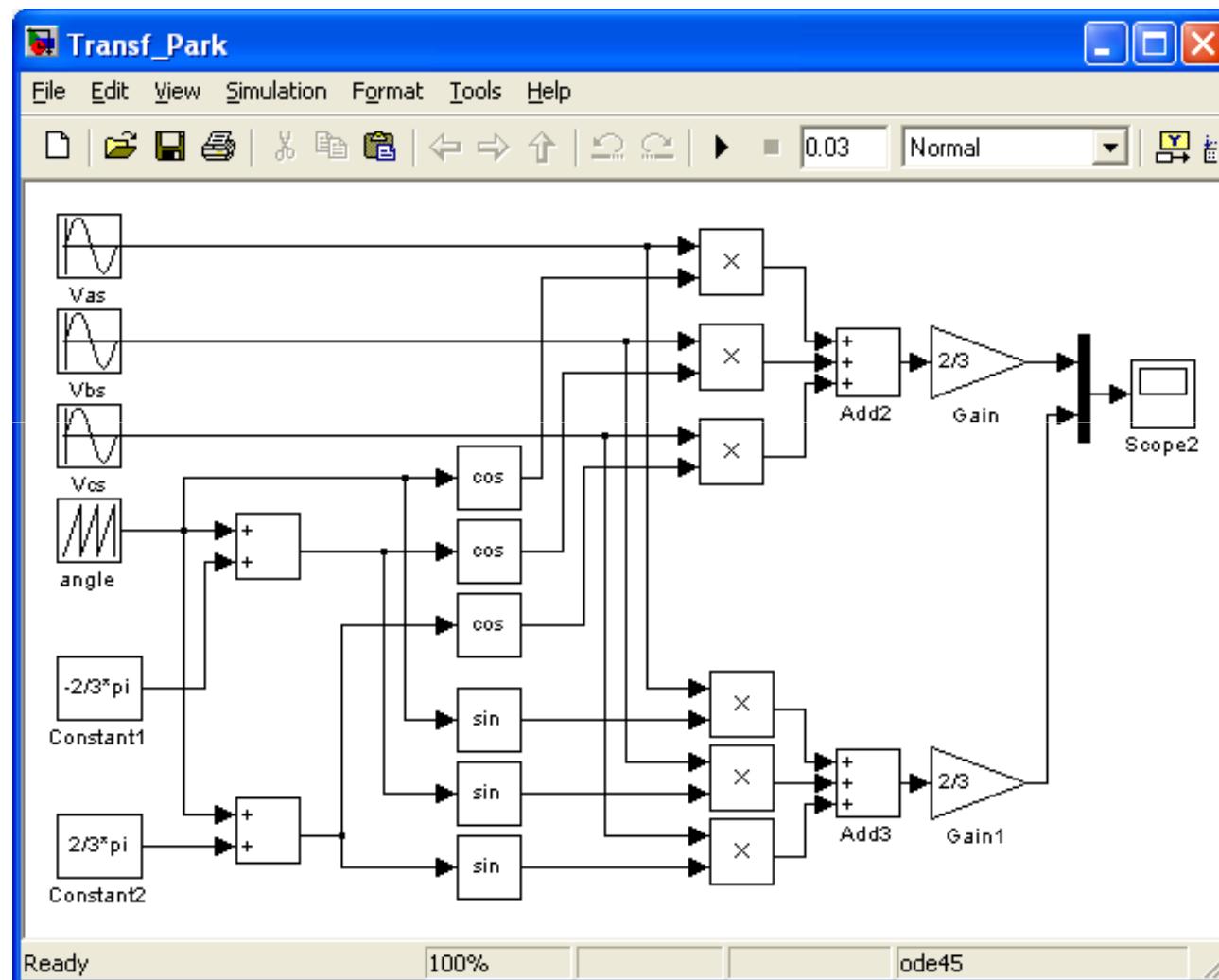


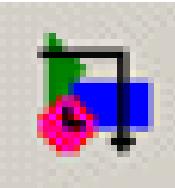
Modulación DSB-AM



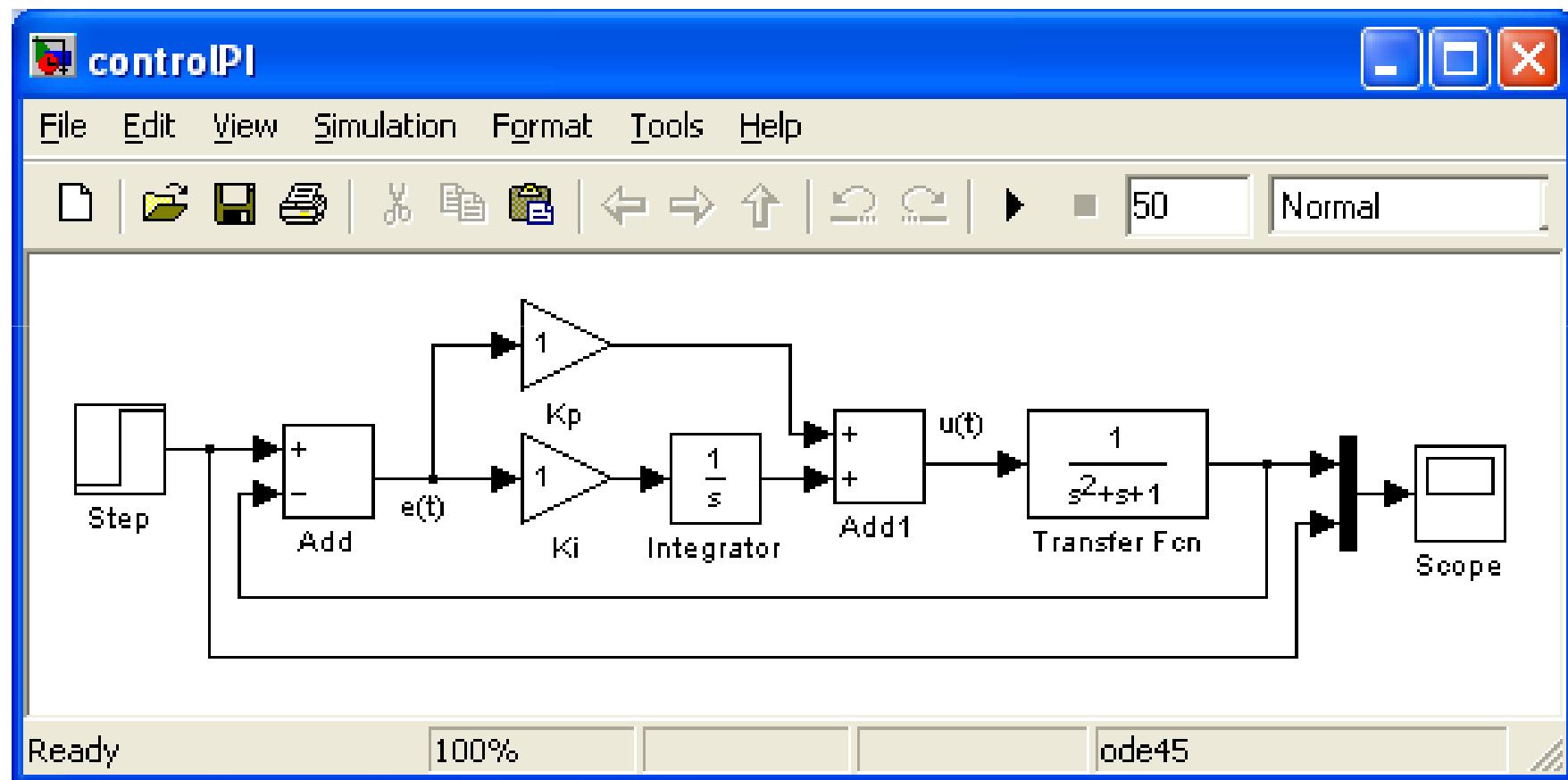


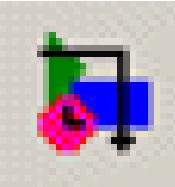
Transformada de Park



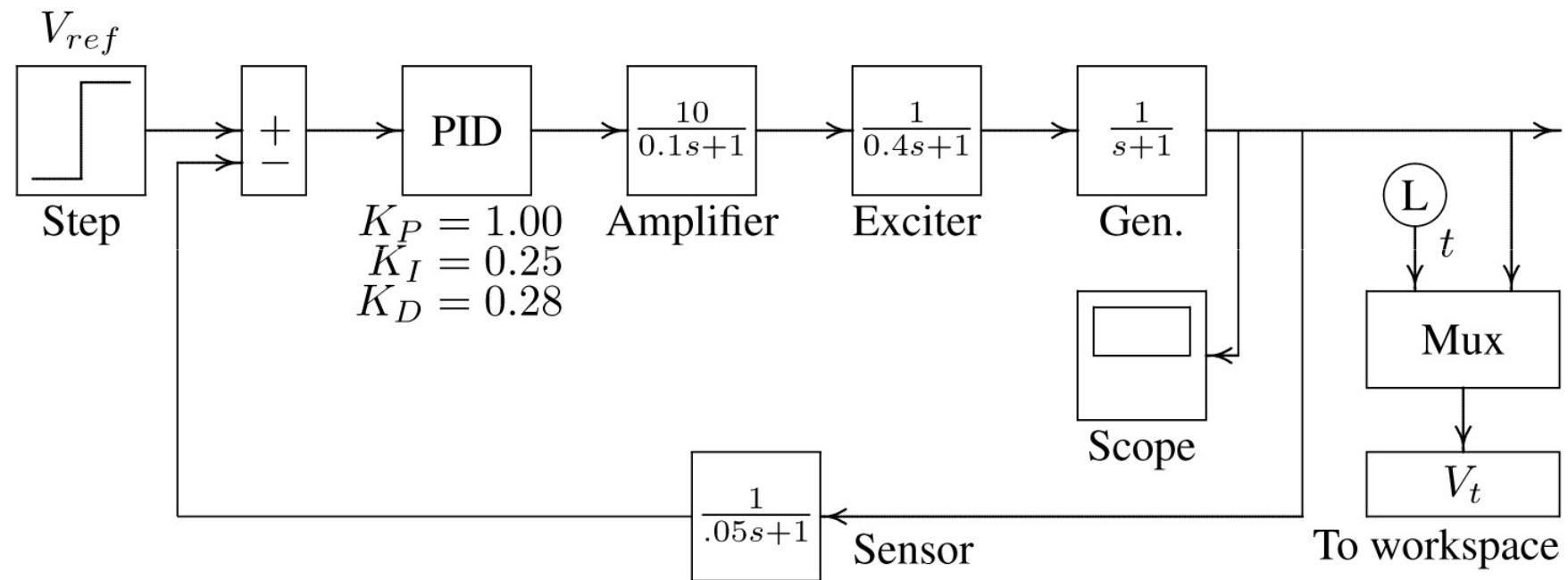


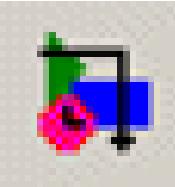
Sistema de Control PI



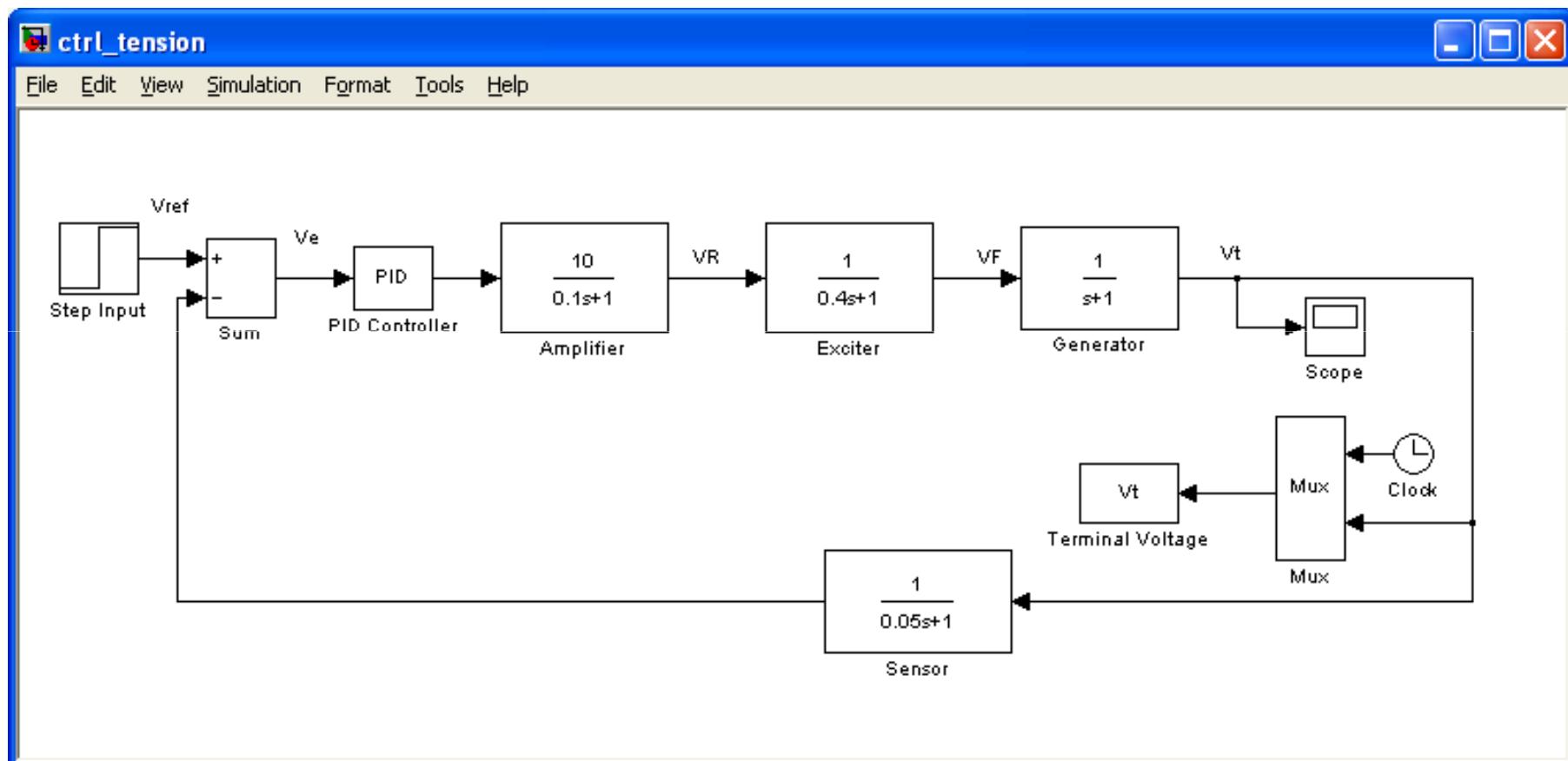


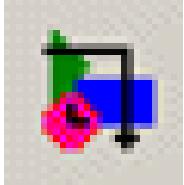
Control de tensión de un generador



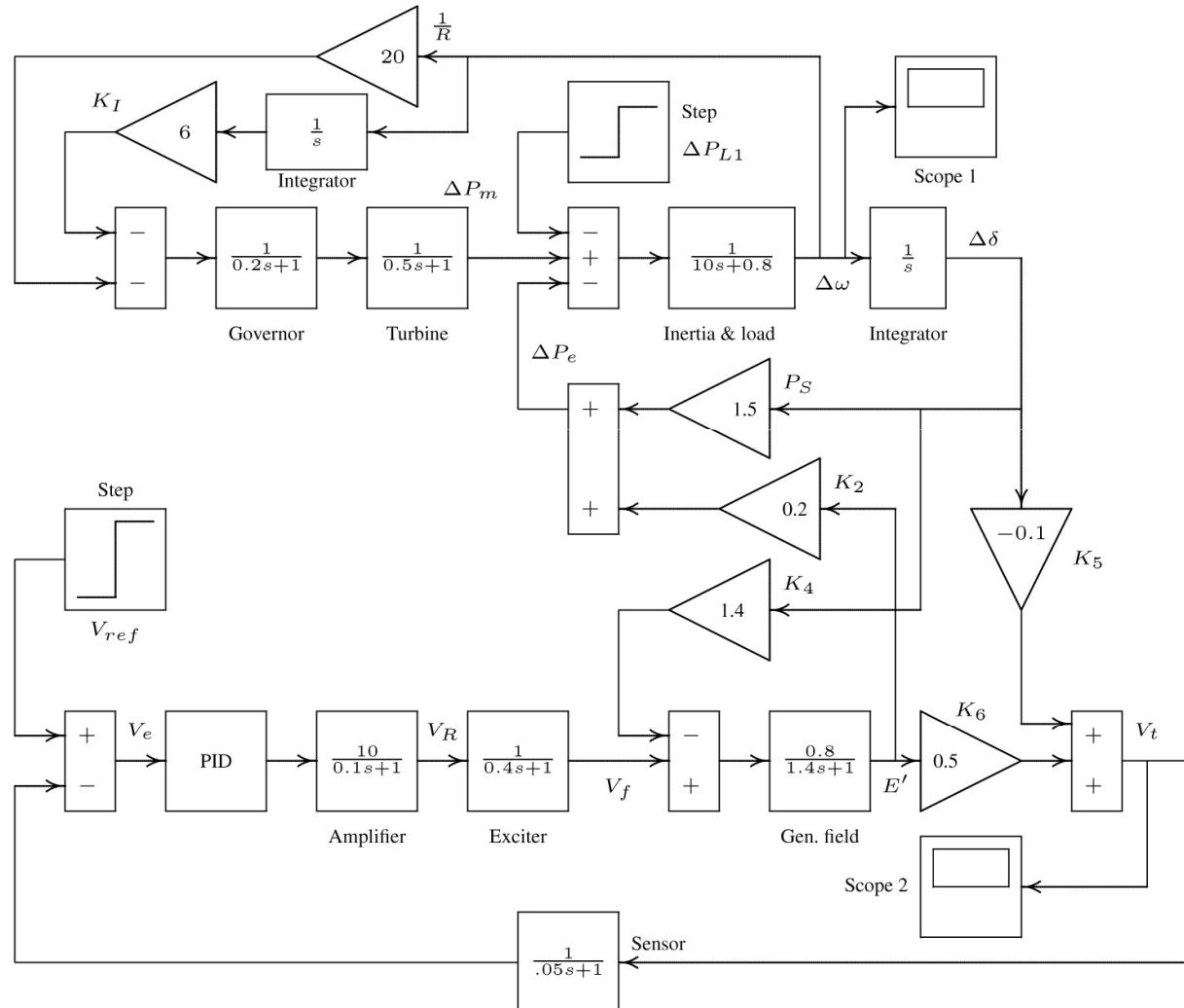


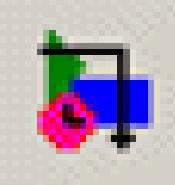
Control de tensión de un generador



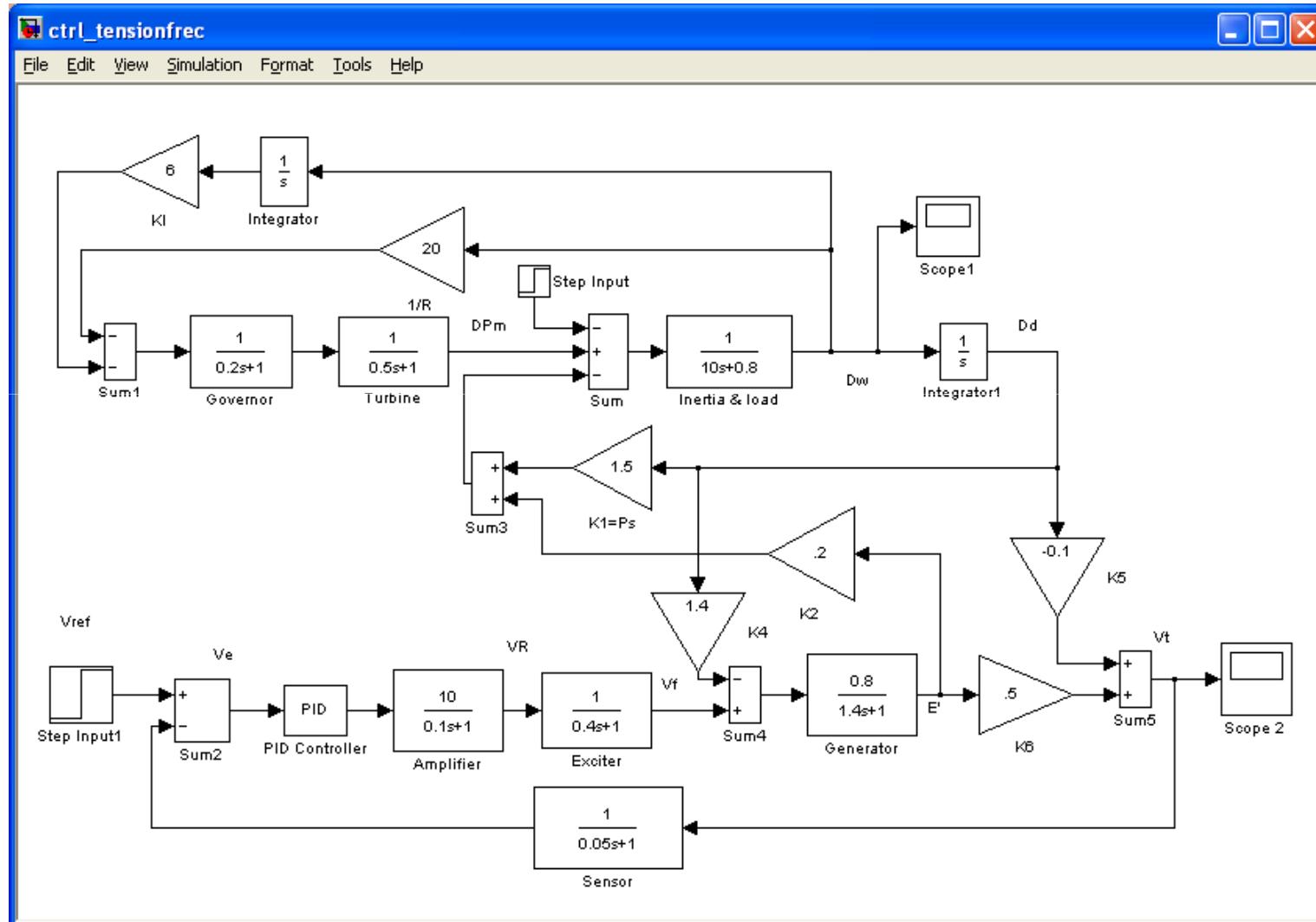


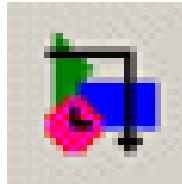
Control tensión/frecuencia de un generador





Control tensión/frecuencia de un generador





Circuito paralelo RLC

