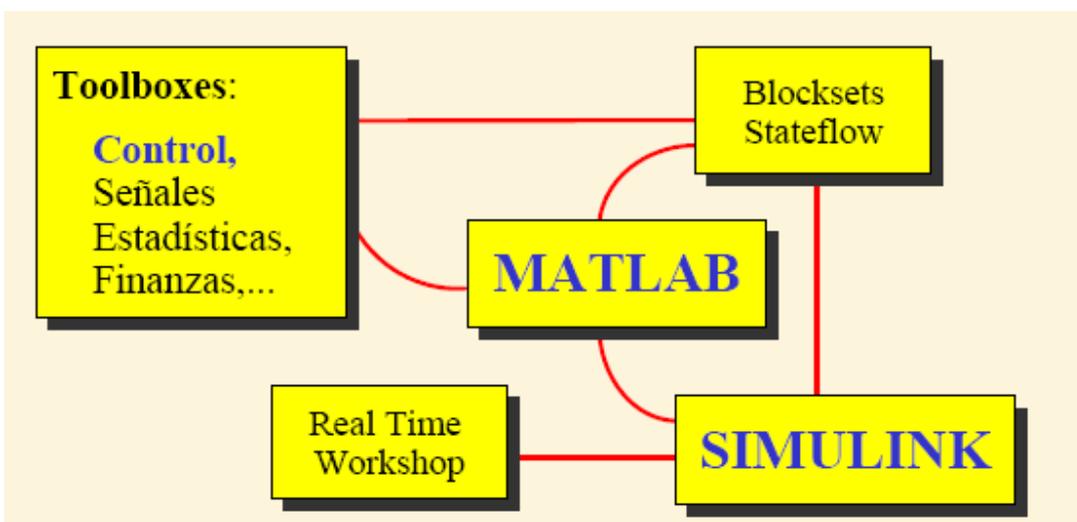


Curso basico de MATLAB

26. Simulink

Es una herramienta interactiva, para modelar y analizar sistemas dinámicos, basada en diagramas de bloques. Está fuertemente acoplada con *Matlab*.

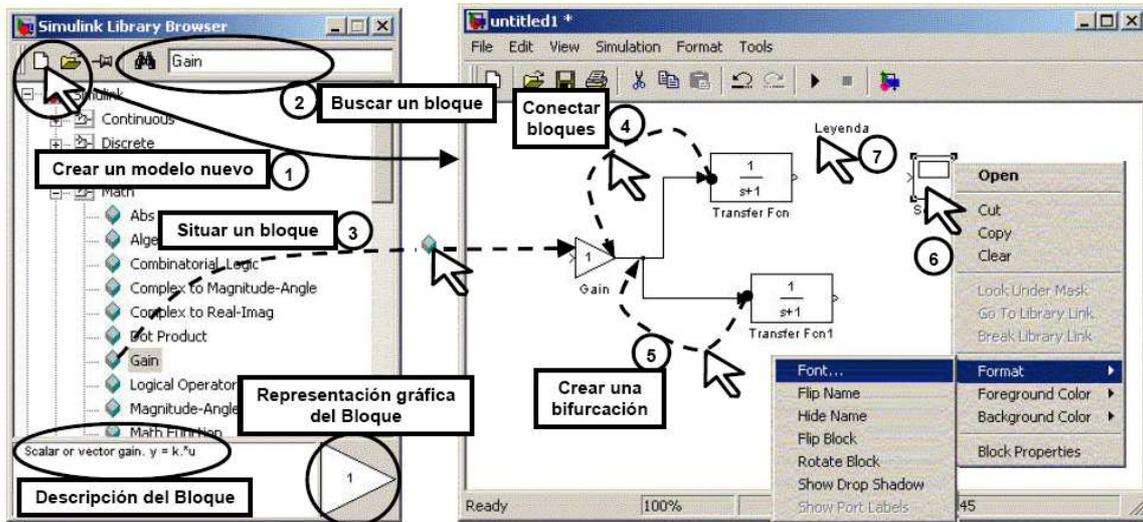


Los sistemas a modelar pueden ser lineales, no lineales, de tiempo continuo, de datos muestreados (con distintas frecuencias de muestreo) o una combinación de ambos.

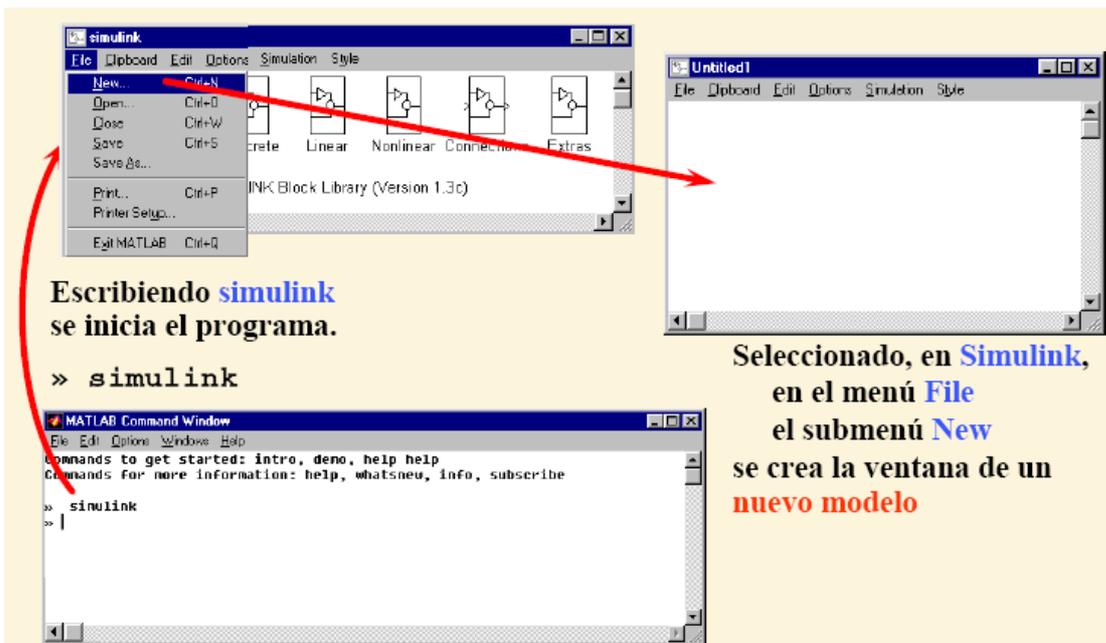
Los modelos a simular pueden ser mecánicos, neumáticos, hidráulicos, eléctricos y electrónicos.

Para modelar sistemas, *Simulink* posee una interfaz grafica que sirve para construir los modelos como diagramas de bloques. Para esto, incluye una muy completa biblioteca de componentes y además de permitirle al usuario crear sus propios componentes e integrarlos a la biblioteca. Los archivos generados tienen extensión *.mdl*

Para iniciar *Simulink* se puede llamar desde el icono de la barra de tareas de *Matlab* o escribiendo e comando *Simulink* en el espacio de trabajo de *Matlab*.



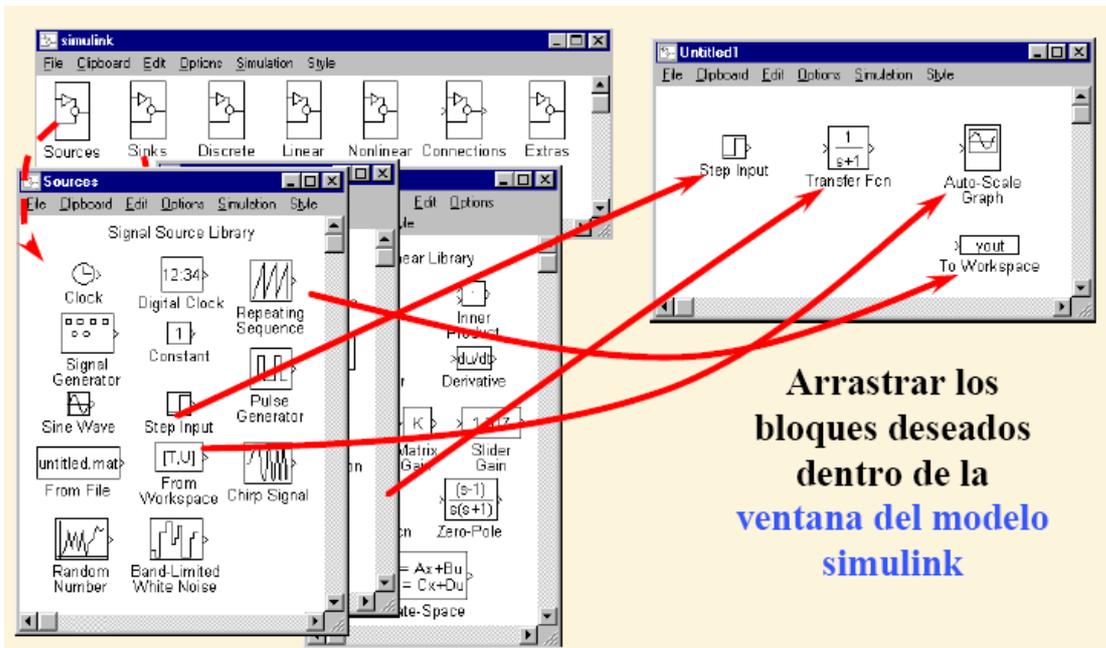
26.1 Crear un modelo nuevo



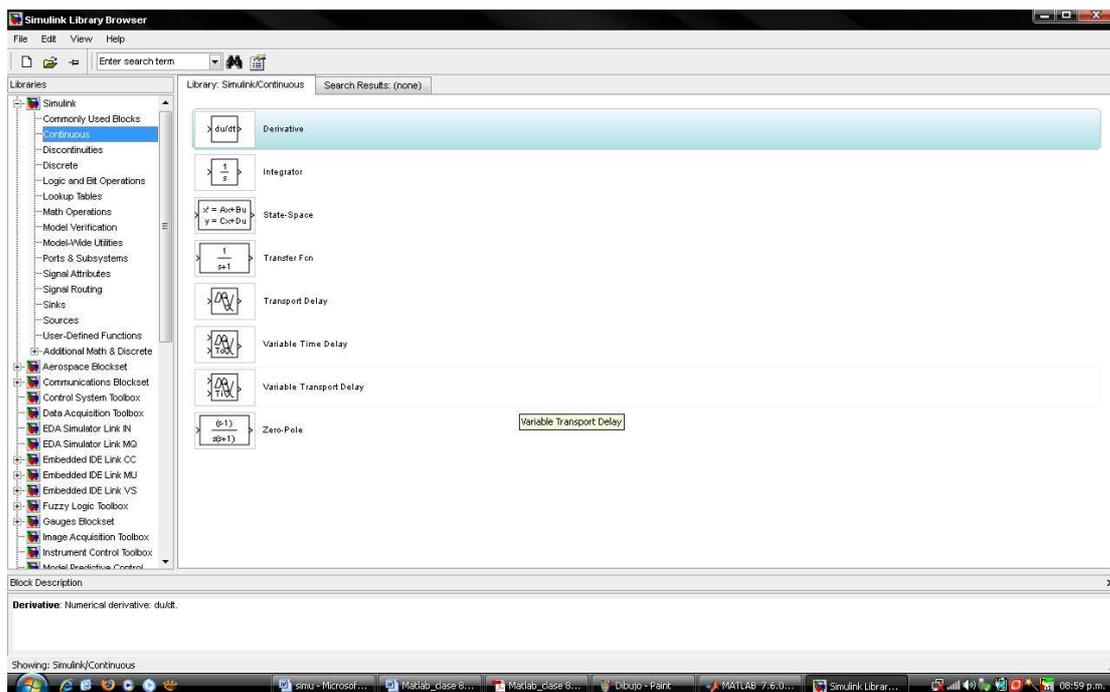
También haciendo clic en el icono.

26.2 Añadir bloques en el modelo

En las primeras versiones de *Simulink*, la biblioteca de funciones comprendía una cantidad acotada de bloques.



En las últimas versiones de *Simulink*, la biblioteca incorporo una gran cantidad de bloques.



<p>Continuous: Bloques que pueden ser representados como una función continua en el tiempo. Derivative: La salida del bloque se corresponde con la derivada de la entrada. Integrator: La salida del bloque se corresponde con la integral de la entrada. Los parámetros del bloque permiten controlar el valor inicial de la salida, así como la existencia de límites superiores e inferiores en la salida. Transfer Fcn: Permite expresar una función de transferencia mediante su expresión en la variable compleja s. Sus parámetros son los polinomios del numerador y del denominador de la función, expresados como vectores fila. Transport Delay: La salida del bloque se corresponde con la entrada al mismo retrasada una cantidad de tiempo, que se fija como parámetro en el bloque. Zero-Pole: Función de transferencia expresada en función de la ganancia en régimen permanente, y la situación de los polos y ceros del sistema.</p>	<p>Sources: Fuentes de señales. Chirp Signal: Genera una señal senoidal, modulada en frecuencia, entre un valor inicial y un valor final. Clock: Tiempo que se lleva de simulación. Constant: Proporciona una señal de valor constante. From Workspace: Proporciona una secuencia de datos tomadas del entorno de trabajo del <i>Matlab</i>. La variable elegida debe contener una matriz indicando los valores de la señal, y los instantes en los que la señal toma estos valores. From File: Proporciona datos tomados de un fichero ".mat", en el que debe estar el valor de la variable, junto a los instantes de tiempo en que toma cada valor. Pulse Generator: Genera una onda cuadrada, de la que se puede controlar la amplitud, el periodo y el tiempo de <i>duty</i> (relación entre el tiempo que la onda toma su valor máximo y el tiempo que toma el valor mínimo). Ramp: Genera una señal de tipo <i>Rampa</i>. Random Number: Genera números aleatorios distribuidos según una función normal. Signal Generator: Simula un generador de señales electrónico, permitiendo generar ondas dientes de sierra, ondas cuadradas y senoidales. Sine Wave: Generador de ondas senoidales. Step: Genera una señal de tipo escalón. Uniform Random Number: Genera números aleatorios distribuidos según una función uniforme.</p>
<p>Math: Bloques que realizan operaciones matemáticas sobre sus entradas. Abs: Calcula el valor absoluto de su entrada. Gain: Aplica una ganancia constante a la entrada. Math Function: Este bloque incluye la mayor parte de las funciones matemáticas típicas, con la excepción de las funciones trigonométricas. Product: Calcula el producto escalar de sus entradas. Un parámetro del bloque permite regular el número de entradas del mismo. Sign: Calcula el signo de la entrada. +1 indica positivo, -1 negativo, y 0 un valor nulo. Sum: Calcula la suma de todas sus entradas. Un parámetro permite indicar el número de entradas, y si estas deben ser invertidas antes de la suma. Ejemplo: un valor para el parámetro "++-+" indicaría que el bloque tiene 4 entradas, y que la tercera de ellas debe ser invertida antes de sumarla. Trigonometric Function: En este bloque se incluyen todas las funciones trigonométricas típicas.</p>	<p>Sinks: Sumideros de señales. Display: Representa numéricamente el valor de una variable. Scope: Representa gráficamente la evolución en el tiempo de una variable. To Workspace: Guarda el valor de la señal indicada en una variable del entorno de trabajo del <i>Matlab</i>. Se puede escoger el nombre de la misma, y limitar su tamaño. To File: Guarda en un fichero de tipo ".mat" los datos de la señal de entrada al bloque. Stop Simulation: Detiene la simulación si el valor de la entrada es distinto de 0.</p>

<p>Signals&Systems: Manejo de sistemas y señales.</p> <p>Subsystem: Permite la realización de sistemas jerárquicos. Al abrir el subsistema, nos permite incluir en su interior, nuevos bloques constructivos, e incluso anidar nuevos subsistemas.</p> <p>In1: Por defecto un subsistema no contiene entradas. Por cada entrada que se desee añadir se le debe incluir uno de estos bloques.</p> <p>Out1: Por defecto un subsistema no contiene salidas. Por cada entrada que se desee añadir se le debe incluir uno de estos bloques.</p> <p>Mux: Permite la inclusión de un conjunto de señales en una única línea de transmisión (que transmite datos vectoriales), lo que facilita la representación en el dibujo. Parámetros: número de entradas. Admite tanto entradas escalares como vectoriales.</p> <p>Demux: Permite la descomposición de los datos puesto en forma vectorial en una línea mediante un multiplexador. Parámetros: número de salidas.</p> <p>Data Store Memory: Define una variable del entorno de trabajo que se va a usar como lugar de almacenamiento de datos útil para evitar tener que hacer conexiones complejas que compliquen el diagrama de bloque que se está usando.</p> <p>Data Store Read: Lee el valor actual de una variable de almacenamiento, que debe estar previamente definida mediante un bloque Data Store Memory</p> <p>Data Store Write: Cambia el valor actual de una variable de almacenamiento, que debe estar previamente definida mediante un bloque Data Store Memory</p>	<p>Nonlinear: Bloques no lineales.</p> <p>Dead Zone: Incluye una <i>zona muerta</i> en el sistema, centrada en torno a cero. El sistema no responde ante estos valores. La magnitud de la zona muerta puede ser modificada, y echa asimétrica por medio de los parámetros del sistema.</p> <p>Relay: La salida pasa al estado on=1 cuando la entrada supera un valor umbral, y a estado off=0 cuando cae por debajo de un umbral distinto. El estado inicial es off.</p> <p>Saturation: La señal de salida no sobrepasa un valor umbral, configurable con los parámetros del bloque.</p> <p>Switch: Una entrada del sistema permite escoger cual de las otras dos entradas se presenta en la salida.</p>
--	---

26.3 Conectar los bloques

Para añadir una línea:
Arrastrar, pulsando el botón derecho del ratón, desde una salida, de alguno de los bloques

The figure consists of four sequential screenshots of the Simulink workspace, each titled 'Untitled'.
1. Top-left: A 'Step Input' block is connected to a 'Transfer Fcn' block (containing $\frac{1}{s+1}$). The 'Transfer Fcn' block is connected to an 'Auto-Scale Graph' block. A 'To Workspace' block (containing 'yout') is present but not yet connected.
2. Top-right: A red dashed arrow shows the 'To Workspace' block being dragged from its input port towards the output of the 'Transfer Fcn' block.
3. Bottom-left: The 'To Workspace' block is now connected to the output of the 'Transfer Fcn' block. A red dashed arrow shows the 'To Workspace' block being dragged from its output port towards the 'Auto-Scale Graph' block.
4. Bottom-right: The 'To Workspace' block is now connected to the output of the 'Auto-Scale Graph' block, completing the connection chain.

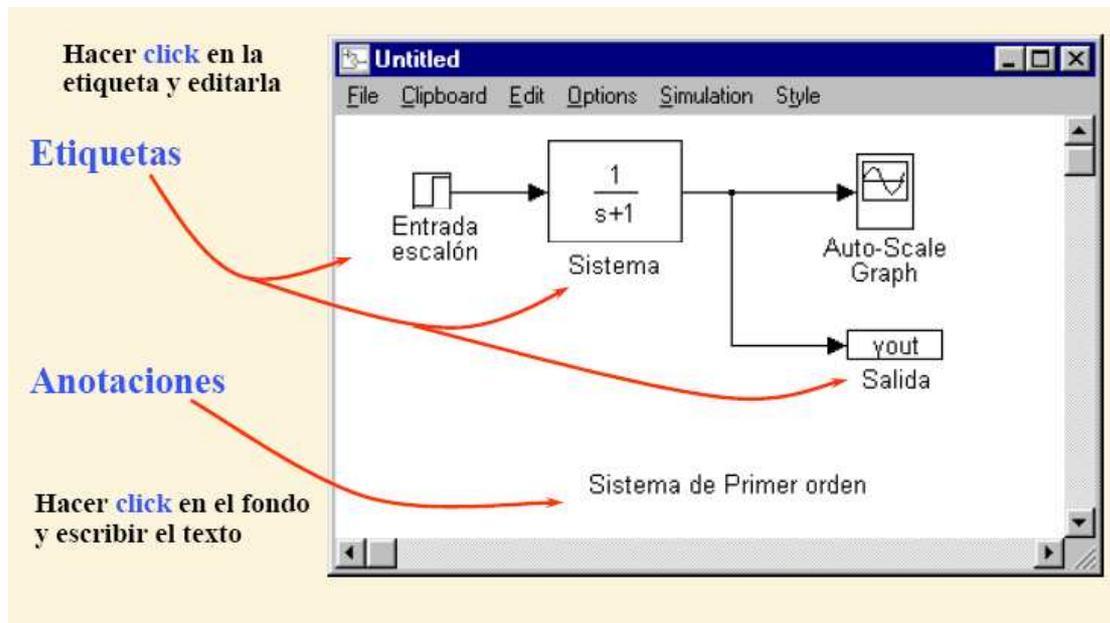
26.4 Cambiar el tamaño de los bloques

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

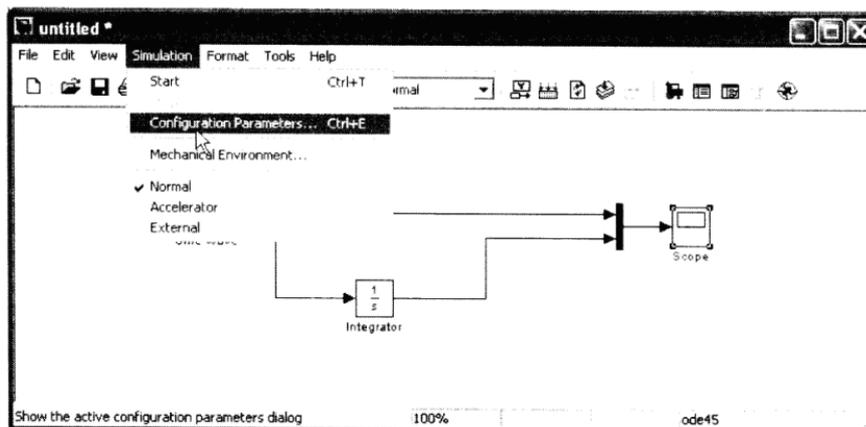
The figure shows a single screenshot of the Simulink workspace titled 'Untitled'. It contains a 'Step Input' block connected to a 'Transfer Fcn' block (containing $\frac{1}{s+1}$). The 'Transfer Fcn' block is connected to an 'Auto-Scale Graph' block and a 'To Workspace' block (containing 'yout'). The 'Transfer Fcn' block is selected, and its four corners are marked with small squares, indicating that it is in a state where its size can be adjusted by dragging these squares.

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

26.5 Modificar etiquetas y añadir anotaciones

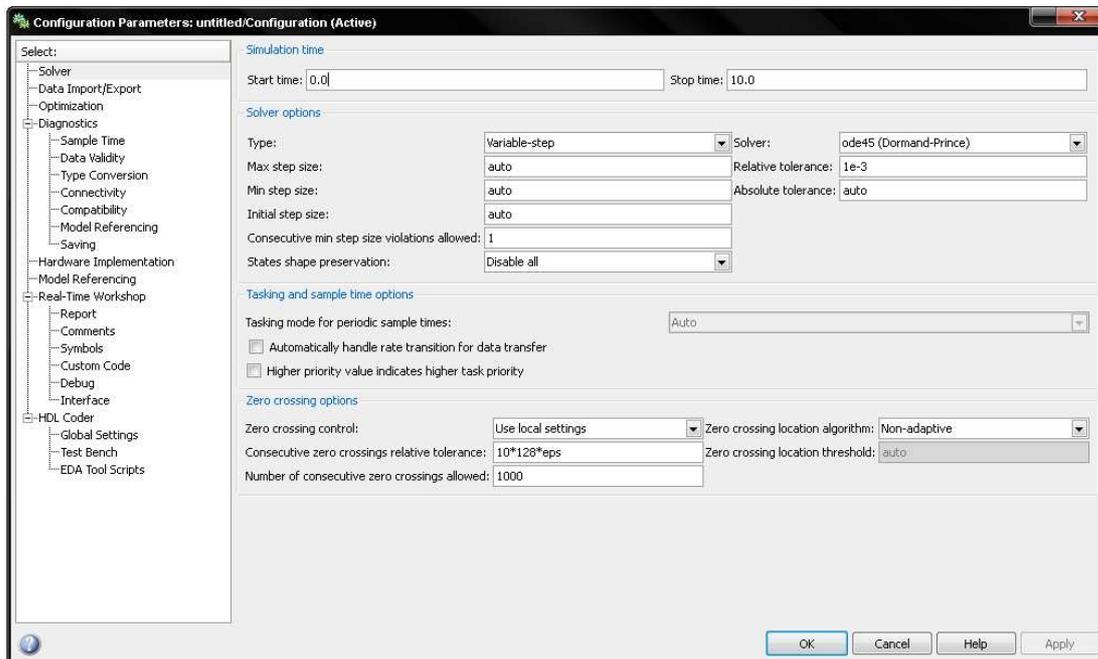


26.6 Configurar los parámetros de la simulación

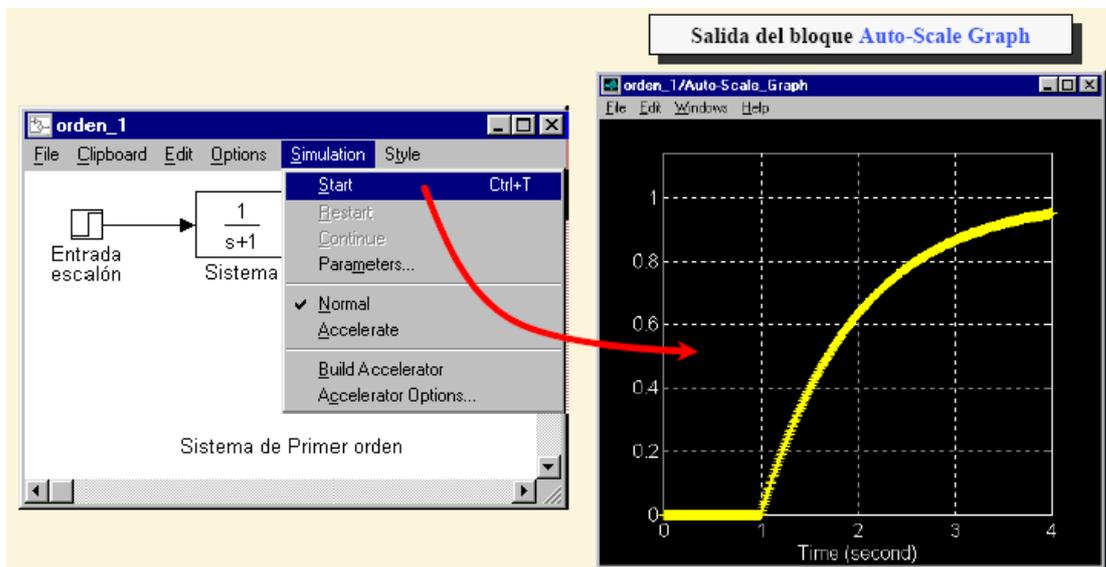


Entre los parámetros más importantes a configurar tenemos:

- *Tiempo inicial de la simulación*
- *Tiempo final de la simulación*
- *Método de resolución numérica*
- *Tipo de paso: fijo o variable*
- *Tolerancia relativa*
- *Tolerancia absoluta*

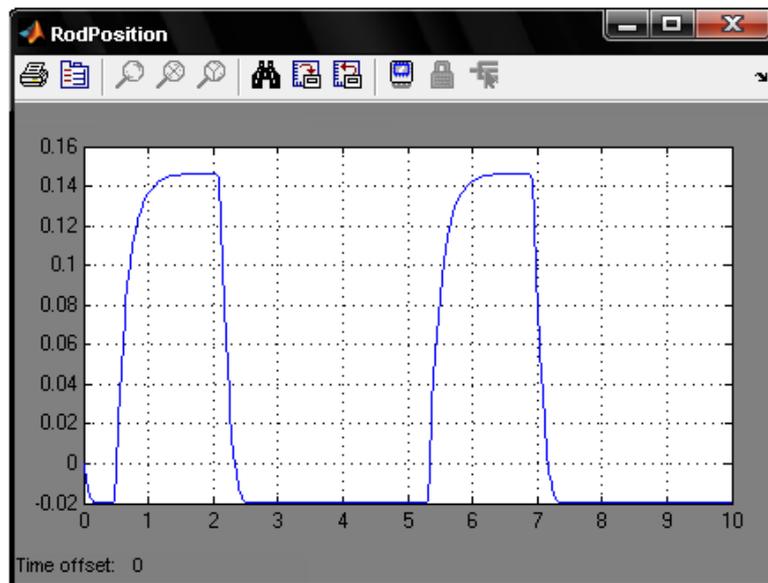
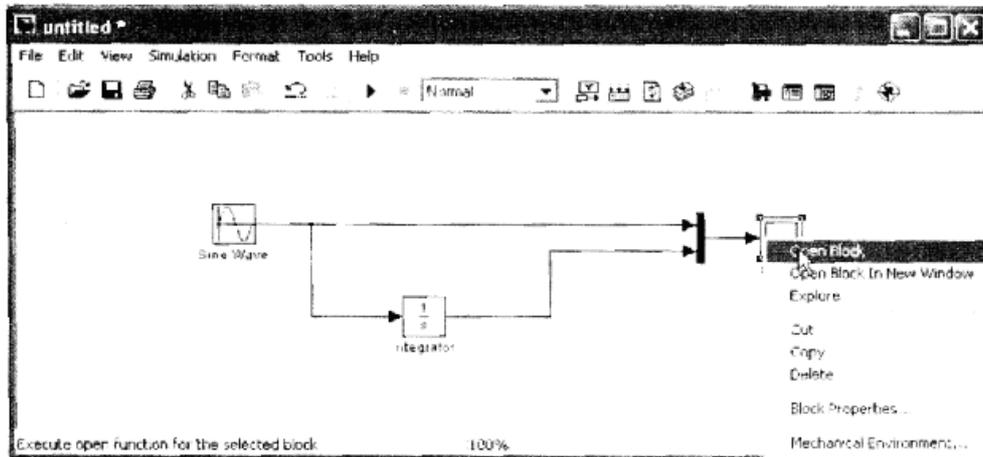


26.7 Ejecutar una simulación



26.8 El elemento SCOPE

Este elemento es muy útil ya que nos permite visualizar los valores de las variables en cada instante y representarlos en una curva.



-  Imprimir
-  Parámetros
-  Magnificar la gráfica
-  Expandir el eje x
-  Expandir el eje y
-  Autoescalar la gráfica
-  Guardar las escalas de los ejes actuales
-  Restaurar las escalas de los ejes anteriores
-  Hacer el Scope flotante
-  Liberar las escalas de los ejes
-  Selección de la señal a graficar

Print	Sirve para imprimir la gráfica que Scope presenta en ese momento.
Parámetros	Abre una ventana para indicar el número de ejes, el rango de tiempo, poner marcadores a las curvas, indicar si es Scope flotante. Además da la opción de enviar los resultados a la ventana de trabajo de MATLAB para su uso en otro programa y de limitar el número de puntos de los datos a graficar.
Autoescalar	Automáticamente fija los ejes dentro de los rangos de la señal graficada.
Selección de señal	Permite seleccionar cualquiera de las señales disponibles en el modelo.

26.9 Creación de subsistemas

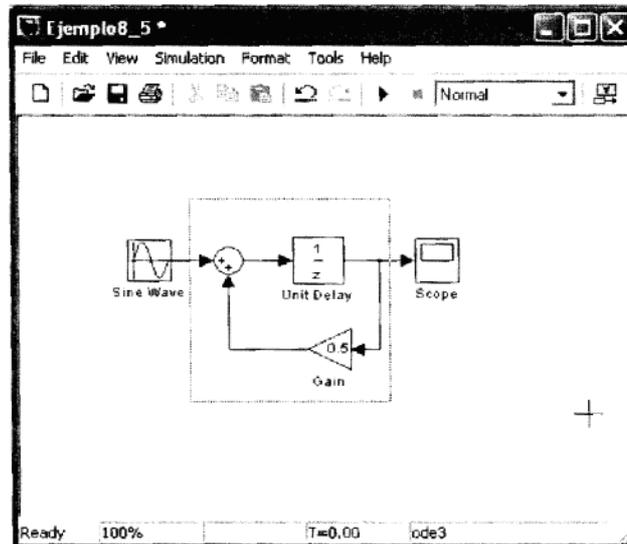
Cuando un modelo aumenta de tamaño y complejidad, resulta de gran utilidad simplificarlo agrupando varios bloques en un subsistema. Usando subsistemas se puede reducir el número de bloques desplegados en el modelo, además de establecer una jerarquía en el diagrama de bloques donde el bloque de un subsistema está en un nivel y los bloques que forman el subsistema están en otro nivel.

Para crear subsistemas hay dos maneras posibles:

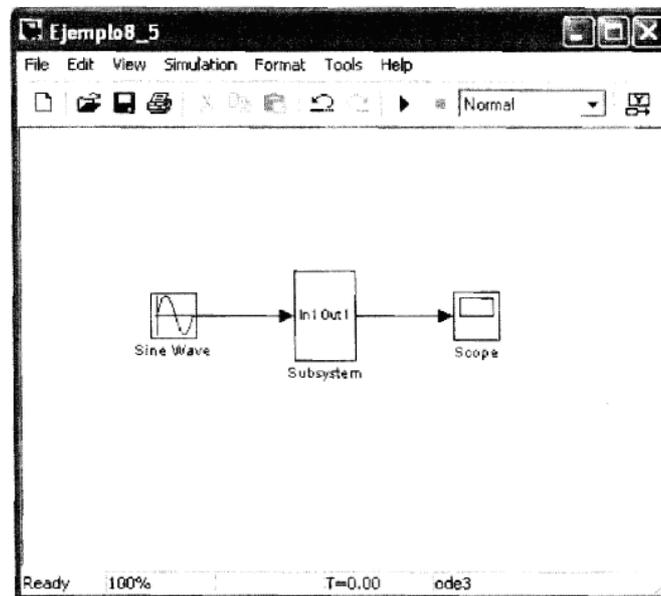
- *Añadir a la ventana de trabajo los bloques que deseamos que contenga el subsistema y agruparlos en un subsistema.*
- *Añadir un bloque de subsistema, abrir dicho bloque y añadirle los bloques del sistema que deseamos que contenga el subsistema.*

a) Añadiendo a la ventana de trabajo los bloques que deseamos que contenga el subsistema y agruparlos en un subsistema.

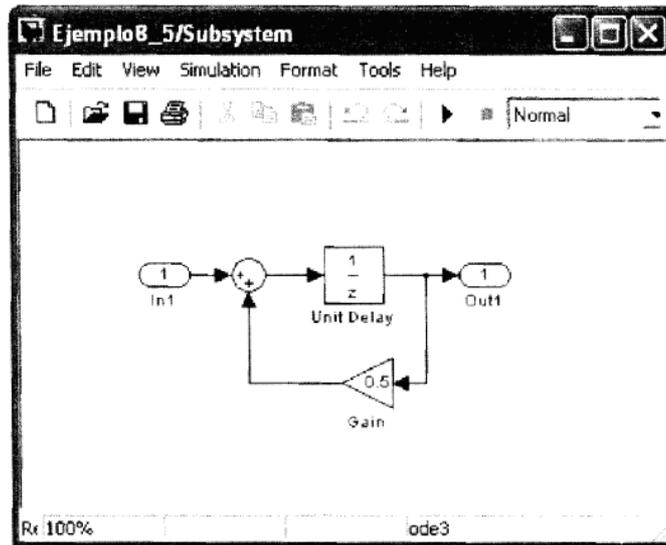
Seleccionamos los bloques y en el menú *EDIT* y luego *CREAR SUBSISTEMA*.



Al hacer esto, *Simulink* reemplaza los electos seleccionados por un bloque de subsistema.

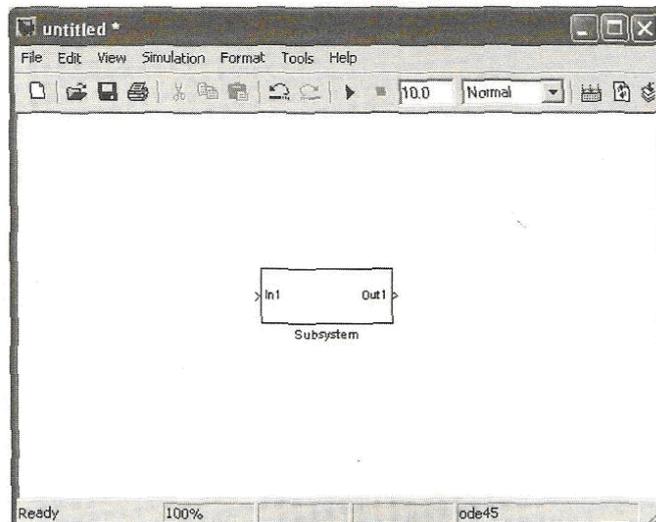


Si se abre el subsistema, puede notarse que *Simulink* añade puertos de entrada y de salida para representar las entradas y salidas del subsistema.



b) Añadiendo un bloque de subsistema, abrir dicho bloque y añadirle los bloques del sistema que deseamos que contenga el subsistema.

Agregamos un bloque de *SUBSISTEMA* de la biblioteca *PUERTOS & SUBSISTEMAS*.



Luego ingresamos al subsistema y agregamos los bloques que deseamos que contenga.

