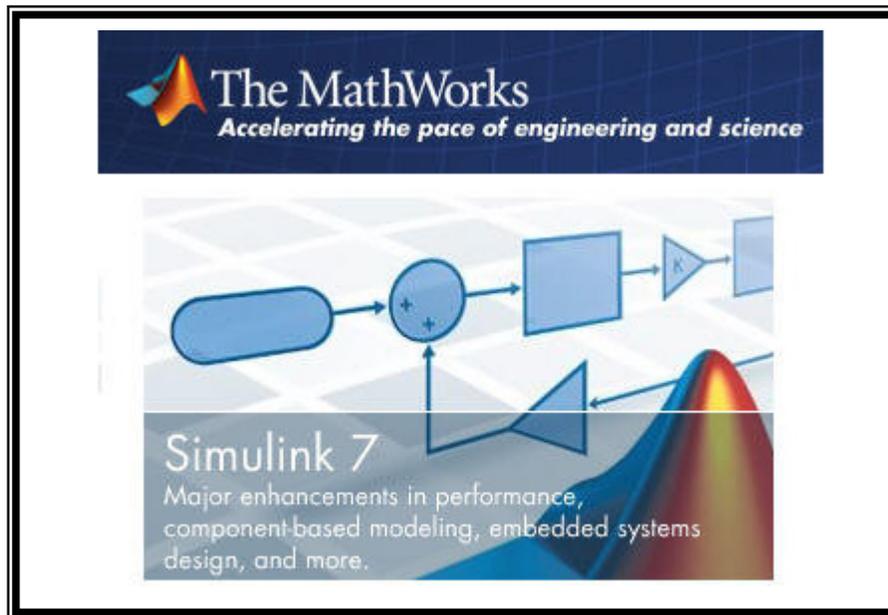

APUNTES

INTRODUCCION A SIMULINK



DESARROLLADO POR: FRANCISCO CUBILLOS M.

2013

INTRODUCCION

El programa de MathWorks para simulación (modelación y análisis) de sistemas dinámicos no lineales fue presentado en 1990, con el nombre de **SIMULAB** para computadoras personales y con el nombre de **SIMULINK** para estaciones de trabajo. Su aparición estuvo unida a la primera versión de MATLAB para Windows. Desde mayo de 1994, que está disponible la versión 1.3, **SIMULINK** tiene un tratamiento similar a los otros Toolboxes de **MATLAB**, en el sentido que se instala de forma separada, pero sigue siendo la mejor herramienta para aprovechar toda la potencia de **MATLAB** y de los otros "**Toolboxes**".

Simulink es una herramienta interactiva para modelar, simular y analizar sistemas dinámicos. Nos permite construir diagramas de bloque gráficos, evaluar el rendimiento del sistema y refinar sus diseños. Simulink está firmemente integrado con Stateflow para modelar comportamiento even-driven. Simulink es la herramienta a escoger para el diseño de sistemas de control, diseños DSP, diseños de sistemas de comunicaciones y otras aplicaciones de simulación. Como una extensión de Matlab, Simulink adiciona muchas características específicas a los sistemas dinámicos, mientras conserva toda la funcionalidad de propósito general de Matlab. Así Simulink no es completamente un programa separado de Matlab, sino un anexo a él. El ambiente de Matlab está siempre disponible mientras se ejecuta una simulación en Simulink.

Simulink tiene dos fases de uso: la definición del modelo y el análisis del modelo. La definición del modelo significa construir el modelo a partir de elementos básicos construidos previamente, tal como, integradores, bloques de ganancia o servomotores. El análisis del modelo significa realizar la simulación, linealización y determinar el punto de equilibrio de un modelo previamente definido.

Para simplificar la definición del modelo Simulink usa diferentes clases de ventanas llamadas ventanas de diagramas de bloques. En estas ventanas se puede crear y editar un modelo gráficamente usando el mouse. Simulink usa un ambiente gráfico lo que hace sencillo la creación

de los modelos de sistemas. Después de definir un modelo este puede ser analizado seleccionando una opción desde los menús de Simulink o entrando comandos desde la línea de comandos de Matlab.

Simulink puede simular cualquier sistema que pueda ser definido por ecuaciones diferenciales continuas y ecuaciones diferenciales discretas. Esto significa que se puede modelar sistemas continuos en el tiempo, discretos en el tiempo o sistemas híbridos.

Simulink usa diagramas de bloques para representar sistemas dinámicos. Mediante una interfase gráfica con el usuario se pueden arrastrar los componentes desde una librería de bloques existentes y luego interconectarlos mediante conectores.

Algunas Características Principales:

Facilidad de Uso:

- Una biblioteca extensa de bloques predefinidos para construir modelos gráficos de su sistema.
- El Debugger gráfico de Simulink para localizar y diagnosticar fallos en el modelo.
- El Visualizador de modelos para navegar por las jerarquías del modelo.
- Utilidad gráfica para buscar modelos y bibliotecas.
- Bloques personalizables que pueden incorporar código ya existente del usuario en C, Ada, MATLAB, y Fortran.

Ayuda para aplicaciones más grandes:

- Sistemas lineales, no lineales, tiempo continuo, tiempo discreto, multirate, ejecutados condicionalmente, e híbridos.
- Los modelos pueden agruparse en jerarquías para crear una vista simplificada de los componentes o los subsistemas.

- Los objetos de datos de Simulink le permiten crear tipos de datos de MATLAB específicos de la aplicación para los modelos de Simulink .
- Simulink Explorer GUI para visualizar y editar objetos de datos de Simulink
- Visualizador de la biblioteca para seleccionar los bloques en una manera conveniente.
- Protección de propiedad intelectual utilizando funciones S (requiere Real-Time Workshop 4.0)
- Puede ejecutar las simulaciones desde la línea de comando de MATLAB, de forma interactiva o en modo batch.

Ayuda Computacional:

- Ayuda para las señales y las operaciones de matrices.
- El bloque Bitwise Logical Operator enmascara, invierte, o desplaza los bits de una señal entero y sin signo en una manera lógica

Acelerador de Simulink

Para incrementar la velocidad de Simulink se debe instalar el acelerador "Accelerator". Este permite automáticamente generar una versión mejorada de los modelos los cuales correrán diez veces más rápido que el original. El acelerador puede ser usado sobre modelos continuos, discretos en el tiempo y híbridos.

El acelerador trabaja generando y compilando un código-C para un modelo dado. Una vez se completa la compilación, la simulación es ejecutada en la ventana de modelos de Simulink exactamente igual que antes sólo que más rápidamente. El propósito del acelerador es aumentar la velocidad de simulación.

Si el programa MatLab posee instalado el "Accelerator" podrá iniciarse la acción aceleradora seleccionando la opción simulation en el menú principal del Simulink y dentro de esta seleccionando la opción Accelerate. Esta acción es totalmente transparente en el sentido de que el incremento de la velocidad se presenta sin ningún otro requerimiento por parte del usuario.

Generador de código-C en Simulink

Una vez se ha creado un modelo dinámico en Simulink, se puede invocar el generador de código-C que permite convertir el diagrama de bloques implementado en un código C. Este puede ser útil para varios propósitos: puede ser usado para control en tiempo real, simulación en tiempo real o simulación acelerada en tiempo no real. Sus aplicaciones pueden ser control de movimiento, control de procesos, sistemas automotores, equipos médicos, robótica, etc.

El código-C es diseñado tal que puede ser ejecutado en tiempo real. No requiere ser escrito manualmente por un programador pues es creado a nivel de diagramas de bloques en Simulink.

El código generado puede correr sobre un amplio rango de hardware ubicado en estaciones de trabajo, PC o microprocesadores. Este código es la forma en la que puede usarse el Simulink para adquisición de datos.

Construcción de modelos

Simulink le permite crear modelos y simulaciones de prácticamente cualquier tipo de sistema dinámico del mundo real. Usando las potentes funciones de simulación de Simulink, puede crear modelos, evaluar diseños y corregir imperfecciones en éstos antes de contruir los prototipos.

Algunas recomendaciones para construir modelos:

- **Use sistemas jerárquicos:** Los modelos complejos se benefician de la incorporación de jerarquías usando subsistemas. La agrupación de bloques simplifica el nivel superior del modelo haciéndolo fácil de entender.
- **Cree modelos claros:** Modelos bien organizados y documentados son fáciles de entender e interpretar. Para lo anterior, utilice etiquetas para las señales y anotaciones sobre el modelo.

- **Diseñe el modelo en un papel y después use el ordenador**, coloque primero los bloques que se van a utilizar y luego realice las conexiones, esto último reduce el tiempo de diseño a través de sucesivas aperturas de las bibliotecas de bloques.

Los elementos fundamentales para construir modelos con Simulink son: Líneas, Bloques y Subsistemas, estos serán explicados en secciones posteriores.

Qué hace el Simulink

SIMULINK permite definir de una forma lógica las diferentes etapas del proceso de cálculo y el flujo de datos entre ellas. Cada bloque puede representar un solo elemento del proceso o bien un subsistema, además de ser fácilmente modificable para reflejar un cambio en el algoritmo o el enfoque del diseño. Estos diagramas de bloques permiten una descripción en alto nivel del sistema además de ser fácilmente modificables con la finalidad de conseguir el comportamiento deseado, es decir, proporcionan una estructura jerárquica para evaluar el comportamiento de algoritmos alternativos bajo diferentes condiciones de funcionamiento.

En los grupos de investigación y desarrollo, SIMULINK permite que cada persona integrante del mismo pueda entender y modificar de una forma sencilla los detalles del algoritmo y el progreso hacia la fase de implementación.

Simulink incluye muchas características para el análisis detallado de sistemas. Las capacidades principales incluyen: linealización, determinación de puntos de equilibrio, animación, optimización de los parámetros, y análisis paramétrico.

- **Extracción de Modelos Lineales**

Las dinámicas de diagramas de bloques no lineales pueden ser aproximadas por linealización, permitiendo aplicar técnicas de diseño que requieran representaciones de modelos lineales. Puede usar la función `linmod` de Simulink para conseguir modelos lineales del espacio de estados de su diagrama de bloques.

- **Animación**

Simulink proporciona acceso inmediato a los potentes gráficos 2-D y 3-D y capacidades de animación de MATLAB. Puede usar MATLAB para mejorar las visualizaciones y entender mejor el comportamiento de su sistema durante el progreso de la simulación. Para acceder a las capacidades de animación de MATLAB, usted puede usar las funciones S de Simulink para incorporar comandos gráficos estándar de MATLAB.

➤ **Análisis Paramétrico**

Con Simulink, el proceso de diseño puede ser automatizado con el uso de archivos con comandos de MATLAB (M-files) para ejecutar simulaciones múltiples con parámetros variables. Los gráficos resultantes muestran familias de curvas que ilustran la respuesta temporal en función de los parámetros variados. Puede también mejorar el rendimiento del sistema con el uso de funciones contenidas en el Toolbox de Optimización o el Blockset de Diseño de Control no Lineal para sintonizar los parámetros del modelo.

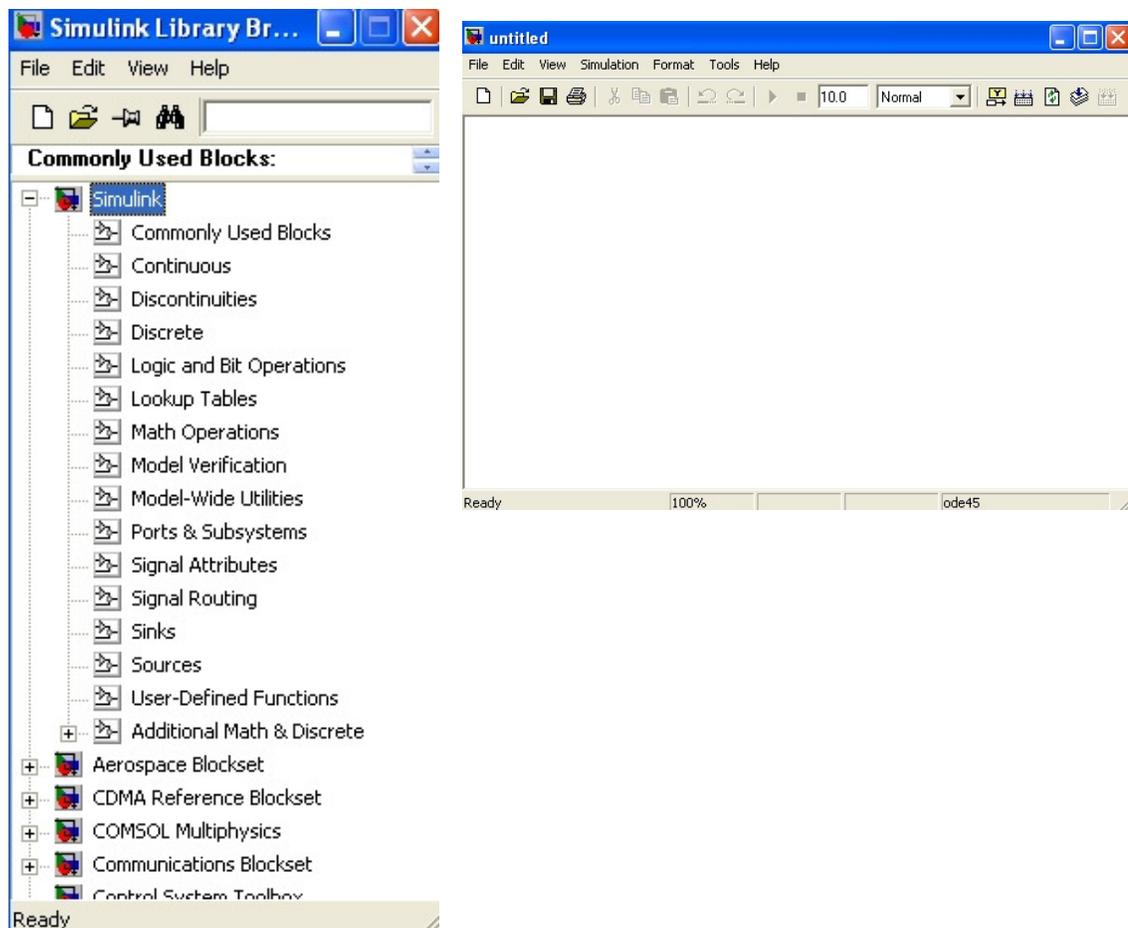
➤ **Integración con MATLAB**

Dado que Simulink está construido a partir de MATLAB, proporciona un entorno único de desarrollo. Este sistema permite ejecutar las simulaciones de manera interactiva, con el uso de las interfaces gráficas de Simulink, o de manera sistemática, con la ejecución de conjuntos de experimentos en el modo batch desde la línea de comandos de MATLAB. Entonces puede generar vectores de prueba y analizar los resultados colectivamente.

1. CREACIÓN DE UN MODELO

1.1. Ventanas y Menús en Simulink

Para comenzar SIMULINK, se debe arrancar primero MATLAB. En el indicador de MATLAB, se introduce la orden simulink. El computador incluirá la ventana de comandos de MATLAB y la ventana SIMULINK Block Library.



La ventana SIMULINK Block Library visualiza los íconos de sus bibliotecas de bloque. Construye modelos copiando bloques de la biblioteca de bloques en una ventana de modelo.

Cuando se ejecuta una simulación y se analizan sus resultados, se puede utilizar comandos de MATLAB que se introducen desde la ventana de comandos de MATLAB.

SIMULINK utiliza ventanas separadas para visualizar una biblioteca de bloques, un modelo o la salida de una simulación gráfica. Estas ventanas no son ventanas de figura de MATHALAB y no se pueden manipular utilizando los comandos del entorno gráfico de MATLAB.

Los menús de SIMULINK aparecen cerca de la parte superior de cada ventana SIMULINK. Los comandos del menú se aplican a los contenidos de esa ventana.

Un error frecuente de los nuevos usuarios de SIMULINK es comenzar una simulación mientras el SIMULINK Block Library es la ventana activa. Hay que asegurarse de que la ventana del modelo es la ventana activa antes de comenzar una simulación.

Para finalizar una sesión de SIMULINK se escoge Exit MATLAB en el menú File.

1.2. Construcción de un modelo

Esta sección analiza las tareas que se llevan a cabo durante la construcción de un modelo. Para crear un nuevo modelo, se escoge la orden **New** en el menú **File**. Simulink crea una nueva ventana. Se puede mover la ventana de la misma forma que se hace con otras ventanas.

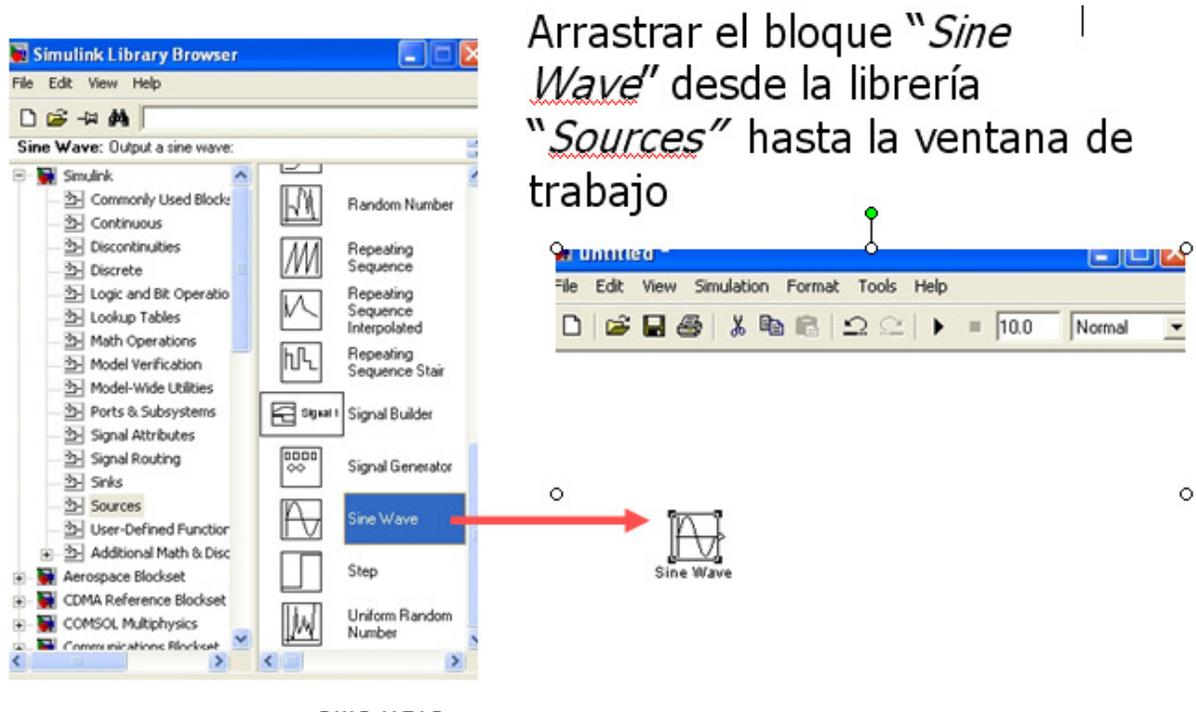
Para editar el diagrama de un modelo existente, se debe hacer una de las dos acciones siguientes:

- Escoger la orden **Open** en el menú **File** y especificar el archivo-M que describe el modelo que se desea editar.
- Introducir el nombre del modelo en la ventana de orden de Matlab.

Simulink crea una nueva ventana y visualiza ese modelo en la ventana.

1.2.1. SELECCIONAR OBJETOS

Muchas acciones de edición y construcción de modelos requieren que primero se seleccione uno o más bloques y líneas(objetos).



Cuando selecciona un objeto pulsando encima de él, cualquier otro que lo estuviera deja de estarlo.

1.2.1.2. Selección de más de un objeto

Puede seleccionar más de un objeto seleccionándolos uno a uno o todos a la vez si están próximos utilizando un recuadro que los englobe.

1.2.1.2.1. Selección de objetos de uno en uno

Para seleccionar más de un objeto, haciéndolo de uno en uno, se debe mantener pulsada la tecla Shift y pulsar sobre cada objeto que se desea seleccionar. Para desactivar la selección de un objeto que está seleccionado, se pulsa otra vez sobre el mismo mientras se mantiene pulsada la tecla Shift.

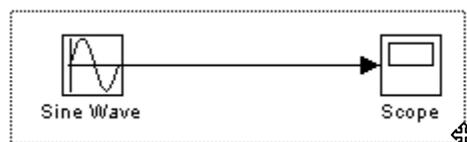
1.2.1.2.2. Selección de objetos utilizando un cuadro de delimitación

Una manera fácil de seleccionar más de un objeto en la misma área de la ventana es dibujar un cuadro de delimitación alrededor de los objetos. Para definir el cuadro de delimitación se debe hacer lo siguiente:

1. Se define la esquina de comienzo de un cuadro de delimitación posicionando el puntero en un ángulo del cuadro, a continuación se pulsa el botón del mouse.



2. Luego se arrastra el puntero al ángulo opuesto del cuadro.



3. Seguido se debe liberar el botón del mouse. Quedan seleccionados todos los bloques y líneas que están parcialmente encerrados por el cuadro de delimitación.



1.2.2. MANIPULACIÓN DE BLOQUES

Se presentará cómo efectuar acciones útiles para construir modelos en los que intervienen bloques.

1.2.2.1. Copiar y mover bloques de una ventana a otra

Para copiar y mover bloques de una ventana a otra se deben hacer los siguientes pasos:

1. Abrir la biblioteca de bloques apropiada o la ventana del modelo fuente.
2. Arrastrar el bloque que se desea copiar en la ventana del modelo.

También se puede copiar bloques utilizando **Copy** y **Paste** en el menú **Edit** mediante los siguientes pasos:

1. Seleccionar el bloque que se desea copiar.
2. Escoger Copy en el menú Edit.
3. Seleccionar la ventana en el modelo de forma que sea la ventana activa.
4. Escoger Paste en el menú Edit.

Otra forma de copiar o mover bloques entre aplicaciones que sean compatibles, es empleando los comandos **Copy**, **Cut** y **Paste**.

1.2.2.2. Mover bloques en un modelo

Para mover un solo bloque de un lugar a otro en una ventana de modelo, se debe seleccionar y arrastrar a una nueva posición.

Para mover más de un bloque, incluyendo las líneas de conexión, se debe hacer lo siguiente:

1. Seleccionar los bloques y las líneas.
2. Arrastrar los bloques y las líneas seleccionadas a sus nuevas posiciones y liberar el botón del mouse.

1.2.2.3. Duplicar bloques en un modelo

La forma de duplicar bloques en un modelo depende del tipo de computador que se esté utilizando:

- ✓ **Windows:** mantener pulsada la tecla **Ctrl**, con el botón izquierdo del mouse seleccionar el bloque y arrastrarlo a una nueva localización. También se puede seleccionar el bloque pulsando el botón derecho del mouse mientras el puntero se encuentra sobre el bloque y lo arrastra a una nueva posición.
- ✓ **Macintosh:** mantener pulsada la tecla **Ctrl**, seleccionar el bloque y arrastrarlo a una nueva localización.

1.2.2.4. Especificación de los parámetros del bloque

Ciertos aspectos de una función de bloque se definen a través de sus parámetros. Se puede asignar valores a los parámetros de un bloque accediendo a su cuadro de diálogo. Haciendo doble clic, al bloque, se visualiza el cuadro de diálogo del bloque, que lista los parámetros y sus valores actuales. Se puede cambiar estos valores o aceptar los valores visualizados.

1.2.2.5. Suprimir bloques

Para suprimir uno o más bloques, se debe seleccionar y pulsar la tecla **Delete** o escoger del menú **Edit** la opción **Clear o Cut**. La orden **Cut** escribe el bloque o los bloques en el portapapeles dejándolos disponibles para que se puedan pegar en un modelo. La utilización de la tecla **Delete** o la orden **Clear** no afecta a los contenidos del portapapeles.

1.2.2.6. Desconectar bloques

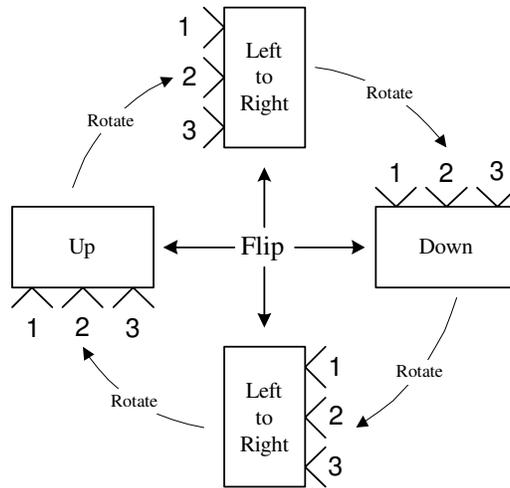
Para desconectar un bloque del modelo sin suprimirlo, se debe mantener pulsada la tecla **Shift** y seleccionar y arrastrar el bloque desde su posición original en el modelo.

1.2.2.7 Cambiar la orientación de los bloques

Hay varias comandos que permiten cambiar la orientación de un bloque, éstas son:

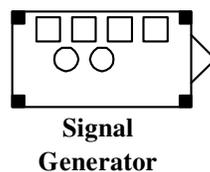
- ✓ La orden **Rotate** en el menú **Options** gira un bloque 90° en el sentido de las agujas de un reloj.
- ✓ La orden **Flip Horizontal** (sistema Windows) o la orden **Flip** (sistema Macintosh) en el menú **Options** gira el bloque 180°.
- ✓ La orden **Orientation** en el menú **Style** le permite seleccionar la orientación del bloque **Left to Right, Right to Left, Up o Down**.

En la siguiente figura SIMULINK ordena los puertos después de cambiar la orientación de un bloque utilizando los comandos **Rotate** y **Flip**. Los textos en los bloques indican su orientación.



1.2.2.8. Redimensionar los bloques

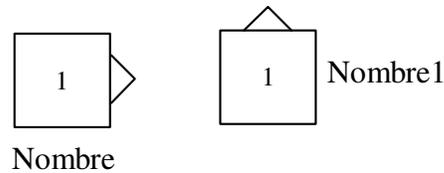
Para cambiar el tamaño de un bloque, se debe seleccionar y arrastrar cualquiera de sus asaderas de selección. El tamaño mínimo de un bloque es de cinco por cinco pixels. El tamaño máximo está limitado por el tamaño de la ventana. La forma del cursor refleja la esquina y la dirección en la que se está siendo arrastrada. Mientras el bloque está siendo redimensionado, un rectángulo punteado muestra el tamaño propuesto.



1.2.2.9. Manipulación de los nombres de los bloques

Los nombres de bloques en un modelo deben ser únicos y deben contener al menos un carácter. Estos nombres aparecen debajo de los bloques si los puertos están en los laterales

y a la derecha de los bloques si están en la parte superior o inferior. Puede cambiar los nombres de los bloques y sus localizaciones.



1.2.2.9.1. Cambiar los nombres de los bloques

Se puede editar los nombres de los bloques de una de las tres formas:

1. Seleccionando el recuadro en el que se visualiza el nombre e introduciendo el nuevo.
2. Colocando el punto de inserción en el nombre e introduciendo el nuevo texto.
3. Arrastrando el mouse para seleccionar el trozo de texto a reemplazar e introduciendo el nuevo texto.

Cuando se pulsa el puntero sobre otro bloque o se realiza cualquier otra acción, el nombre se acepta o se rechaza. Si intenta modificar el nombre de un bloque a un nombre que ya existe o a uno que no tiene caracteres, se visualiza un mensaje de error.

También se puede modificar las fuentes utilizadas en los nombres de los bloques seleccionando el bloque o los bloques y escogiendo un tipo fuente del submenú **Fonts** que está en el menú **Style**.

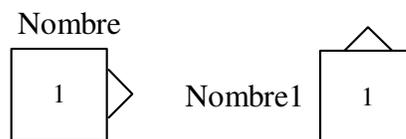
1.2.2.9.2. Cambiar la localización de un nombre de bloque

Se puede cambiar la localización del nombre de un bloque eligiendo la opción **Title** que está en el menú **Style**:

- ✓ **Displayed**, visualiza el nombre.
- ✓ **Hidden**, no visualiza el nombre.

- ✓ **Top/Left** coloca el nombre encima del bloque cuando su orientación es *Left to Right* o *Right to Left* o a la izquierda del bloque cuando su orientación es *Up o Down*.
- ✓ **Bottom/Right**, coloca el nombre debajo del bloque cuando su orientación es *Left to Right* o *Right to Left* o a la derecha del bloque cuando su orientación es *Up o Down*.

Por ejemplo en la figura se muestra la posición de los nombres de bloque Top/Left.



1.2.2.10. Vectorización de los bloques

Casi todos los bloques incorporados aceptan entradas escalares o vectoriales y le permiten especificar parámetros de uno u otro tipo.

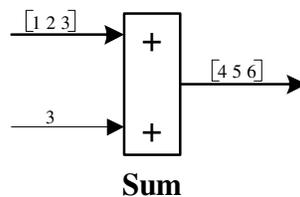
Para determinar qué líneas en un modelo llevan señales vectoriales, se elige la opción **Wide Vector Lines** en el menú **Style**, luego se dibuja las líneas que llevan vectores más gruesas que las que contienen escalares.

Después de elegir esta opción y si ha cambiado el modelo, se debe actualizar explícitamente la visualización seleccionando la opción **Update Diagram** en el menú **Style**. Para comenzar la simulación también actualiza la visualización.

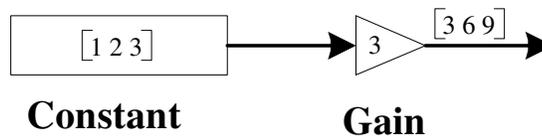
1.2.2.11. Expansión escalar de entradas y parámetros

La expansión escalar es la conversión de un valor escalar en un vector de elementos idénticos. Se puede aplicar la expansión escalar a las entradas y parámetros de bloques.

- ✓ **Entradas:** cuando se utilizan bloques que poseen más de un puerto de entrada, puede mezclar entradas vectoriales y escalares. Las entradas escalares se expanden en vectores que tienen la misma longitud que las entradas vectoriales. Por ejemplo, el bloque Sum muestra el resultado de expandir una entrada escalar para que concuerde el tamaño de una entrada vectorial al bloque.



- ✓ **Parámetros:** se puede especificar los parámetros para bloques vectorizados como vectores o como escalares. Cuando se especifica parámetros, cada elemento se asocia con el vector de entrada correspondiente. Cuando se especifica parámetros escalares, se aplica la expansión escalar para convertirlos automáticamente en vectores del tamaño adecuado. Por ejemplo, la figura muestra el parámetro escalar (Gain) se expande para coincidir con el tamaño de la entrada al bloque, un vector de tres elementos.



1.2.3. MANIPULACION DE LINEAS

Las líneas conectan la salida de un bloque a la entrada de otro bloque. Las líneas también conectan otras líneas a la entrada de un bloque. Pueden conectarse cualquier número de líneas a un puerto de salida, pero solamente se puede conectar una línea a cada puerto de entrada. (El bloque MUX es útil para combinar algunas líneas en una única línea vectorial).

1.2.3.1. Dibujar líneas entre bloques

1. Posicionar el cursor sobre el puerto de salida del primer bloque. No es necesario posicionar el cursor de forma precisa sobre el puerto.



2. Presionar y mantener pulsado el botón del mouse. El cursor cambia a una forma de cruz.



3. Arrastrar el puntero al puerto de entrada del segundo bloque. Se puede posicionar el cursor sobre o cerca del puerto o dentro del bloque. Si se coloca el cursor en el bloque, la línea se conecta al primer puerto de entrada disponible. Para conectar la línea a un puerto específico, se debe posicionar el cursor sobre ese puerto antes de soltar el botón del mouse.



4. Soltar el botón del mouse. SIMULINK sustituye los símbolos de los puertos por una línea de conexión con una flecha que muestra la dirección del flujo de señal. Se puede crear líneas de conexión o desde la salida a la entrada o desde la entrada a la salida. En cualquier de los dos casos, la flecha se dibuja en el puerto de entrada apropiado y la señal es la misma.



1.2.3.1.1. Encaminamiento de líneas alrededor de bloques

SIMULINK encamina automáticamente las líneas alrededor de los bloques en lugar de pasar a través de ellos. Sin embargo, se le puede indicar a SIMULINK que dibuje una línea exactamente como se desee, manteniendo pulsada la tecla SHIFT mientras se dibuja la línea, dibujando la línea del puerto de entrada al puerto de salida o dibujando una secuencia de segmentos de líneas.

La opción **Reroute Lines** es útil para limpiar zonas muy pobladas del diagrama de bloques. Utilizando el cuadro de delimitación, se selecciona un área del modelo y se escoge **Reroute Lines** en el menú **Options**.

1.2.3.1.2. Dibujar líneas desde otras líneas

Se puede añadir una línea que comience en cualquier punto de una línea existente. Ambas líneas transportan la misma señal a sus salidas.

Por ejemplo, en la siguiente figura el diagrama de bloque del lado izquierdo muestra una única línea que va desde el bloque **Product** al bloque **Scope**. El diagrama detallado derecho muestra una línea adicional que va desde el bloque **Product** al bloque **To Workspace**. La misma señal va a cada bloque.

Para añadir una línea desde otra línea, se deben seguir estos pasos.

1. Colocar el puntero en la línea donde se desea comenzar la nueva línea.
2. Mientras se mantiene pulsada la tecla **Ctrl.**, pulsar y mantener presionado el botón del mouse.
3. Arrastrar el puntero al puerto destino y soltar el botón del mouse y la tecla **Ctrl.** SIMULINK crea una nueva línea entre los puntos de comienzo y finalización.

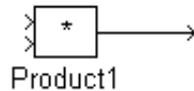
En el sistema Windows, se puede también utilizar el botón derecho del mouse en lugar de mantener pulsada la tecla **Ctrl** mientras se utiliza el botón izquierdo del mouse.

1.2.3.1.3. Dibujar un segmento de línea

Para dibujar un segmento de línea, se dibuja una línea que finaliza en una zona no ocupada del diagrama. Aparece una flecha en el final no conectado de la línea. Para añadir otro segmento de línea, se pulsa el botón del mouse mientras el puntero está sobre la flecha y luego repetir el procedimiento.

Se puede utilizar esta técnica para dibujar una línea con segmentos exactamente donde se desee o para dibujar líneas antes de copiar bloques a los cuales se conectan las líneas. Cuando se comienza una simulación, SIMULINK proporciona un mensaje de aviso si el modelo tiene algunas líneas que no están conectadas.

La figura que se muestra a continuación tiene un segmento de línea sin conectar.



1.2.3.1.4. Ángulos de líneas

SIMULINK dibuja líneas de conexión en múltiplos de 45 grados con estas excepciones:

- Si el puntero se mueve cerca de un puerto que está disponible, la línea se conecta a ese puerto.
- Si se crea una línea mientras se mantiene pulsada la tecla **Shift**, SIMULINK dibuja la línea tal como la crea.
- Si se crea una línea moviendo el puntero sobre el bloque y se suelta, la línea va al puerto no usado en el bloque que está más arriba o más a la izquierda.

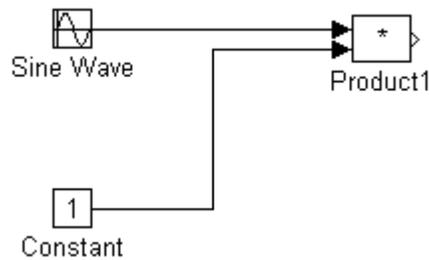
1.2.3.2. Suprimir líneas

Para suprimir una o más líneas, se selecciona la línea o líneas que se van a eliminar y se pulsa la tecla **Delete** o se escoge **Clear** o **Cut** en el menú **Edit**

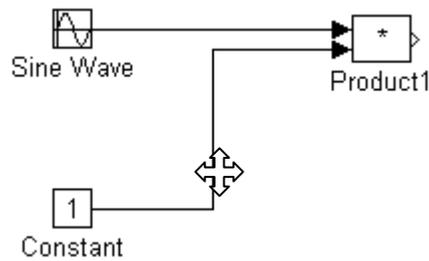
1.2.3.3. Mover segmentos de línea

Para mover un segmento de línea se deben seguir estos pasos:

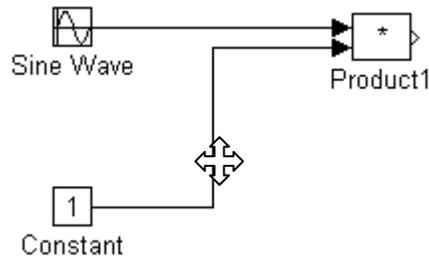
1. Posicionar el puntero sobre el segmento



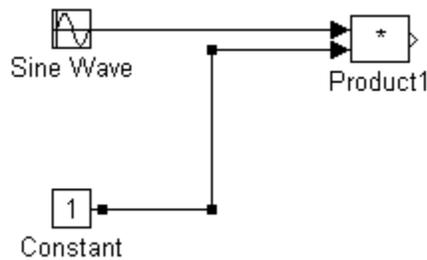
2. Pulsar el botón del mouse y mantenerlo así



3. Arrastrar el puntero a la posición deseada



4. Soltar el botón del mouse

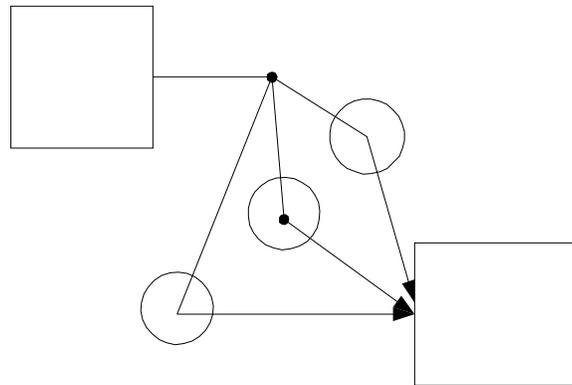


No se pueden mover los segmentos conectados directamente a los puertos de los bloques.

1.2.3.4. Mover vértices

Para mover un vértice de una línea, se debe posicionar el puntero sobre el vértice, pulsar y manteniendo así el botón del mouse, se arrastra el puntero a la posición deseada y se suelta el botón del mouse. No se puede mover los vértices que están en los extremos de la línea.

La figura que sigue muestra la forma del cursor y el movimiento del vértice cuando se arrastra. Se puede arrastrar el vértice en cualquier dirección



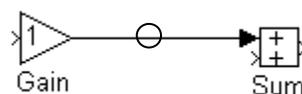
1.2.3.5. Dividir una línea en segmentos

Se puede dividir una línea en dos segmentos (o un segmento de línea en dos segmentos), dejando los extremos de la línea en sus posiciones originales. SIMULINK crea segmentos de línea y un vértice que los une. Para dividir una línea en segmentos, se deben seguir estos pasos:

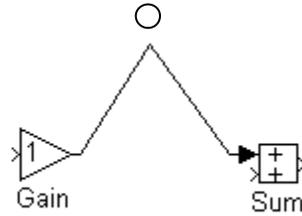
1. Colocar el puntero sobre la línea donde se desea el vértice.



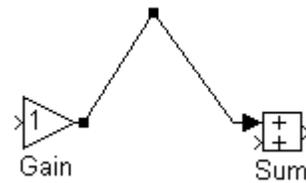
2. Mientras se mantiene pulsada la tecla **Shift**, pulsar y mantener presionado el botón del mouse. .



3. Arrastrar el puntero a la posición deseada.



4. Soltar el botón del mouse y la tecla **Shift**.



1.2.4. RESUMEN DE LAS ACCIONES DEL MOUSE Y EL TECLADO

Acciones de Construcción del modelo	Windows
Seleccionar objetos	Botón izquierdo del mouse
Seleccionar más de un objeto	Shift + botón izquierdo del mouse
Copiar objetos de otras ventanas	Seleccionar el objeto y arrastrar
Duplicar objeto	Option + Botón izquierdo del mouse y arrastrar; o Botón derecho del mouse y arrastrar
Mover objeto	Seleccionar el objeto y arrastrar
Conectar bloques	Botón izquierdo del mouse
Desconectar bloques	Shift + arrastrar bloque
Encaminar líneas desde otra línea	Shift + dibujar líneas
Dibujar líneas desde otra línea	Ctrl + arrastrar línea
Mover segmento de línea	Seleccionar segmento y arrastrar
Mover vértice	Seleccionar vértice y arrastrar
Dividir líneas en segmentos	Shift + arrastrar líneas

1.2.5. AÑADIR ANOTACIONES DE TEXTO AL DIAGRAMA DEL MODELO

- ✓ Posicionando el puntero donde se desea colocar el texto, luego pulsar el botón del mouse y por último escribir el texto.
- ✓ El texto debe ser único en el modelo; y puede ser utilizada para fechar el modelo y etiquetar las líneas.
- ✓ Para modificar las fuentes de texto, se debe seleccionar mediante un cuadro de delimitación y posteriormente, se escoge la fuente del submenú **Fonts**, que está accesible en el menú **Style**.

1.2.6. CREAR SUBSISTEMAS

Los subsistemas son usados cuando nuestro modelo se hace complicado, ya que aumenta de tamaño y complejidad. El agrupamiento es útil por una serie de razones:

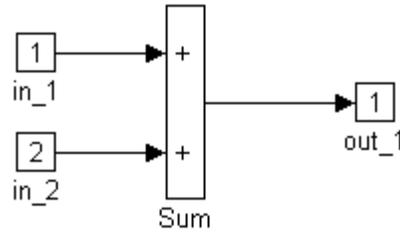
- Ayuda a reducir el número de bloques visualizados.
- Permite mantener juntos los bloques que están funcionalmente relacionados.
- Hace posible establecer un diagrama de bloques jerárquico.

Se pueden crear los subsistemas de dos maneras:

1. Añadiendo un bloque **Subsystem** al modelo, y luego añadiendo los bloques que contiene.
2. Añadiendo los bloques que constituyen el subsistema y posteriormente agrupándolos.

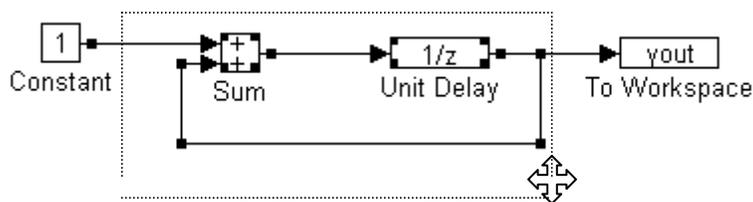
Para crear un subsistema antes de añadir los bloques que contiene, se debe insertar un bloque **Subsystem** y crear los bloques que constituyen el subsistema de la forma siguiente:

- a. Copiar el bloque **Subsystem** de la biblioteca **Connections** del modelo.
- b. Abrir el bloque **subsystem** (doble clic).
- c. En la ventana nueva del bloque **Subsystem**, crear el correspondiente subsistema. Utilizar bloques **Inport** para representar las entradas que vienen desde fuera del subsistema y bloques **Output** para las salidas externas. Por ejemplo, el bloque **Sum** que se muestra a continuación es el único bloque de un subsistema. El diagrama representa el bloque y sus bloques **Inport** y **Outport** asociados:



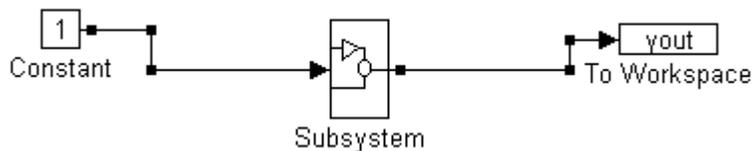
Una vez creados los bloques que se desea convertir en un subsistema, se debe hacerlo siguiente:

- a. Encierre los bloques y líneas de conexión que desea incluir dentro del subsistema mediante un cuadro de delimitación. Por ejemplo, la figura que viene a continuación muestra un modelo que representa un contador. Los bloques Sum y Delay se seleccionan dentro de un cuadro de delimitación:



Cuando se suelta el botón del mouse, se seleccionan los bloques y todas las líneas de conexión.

- b. Escoger **Options**, luego **Group**. SIMULINK reemplaza los bloques que están en el grupo por un único bloque **subsystem**. La figura muestra el modelo después de escoger la orden **Group**:



1.2.7. MODELAR ECUACIONES

Uno de los temas más confusos para los nuevos usuarios de SIMULINK es cómo modelar ecuaciones. Algunos ejemplos ayudarán a comprender cómo modelarlas.

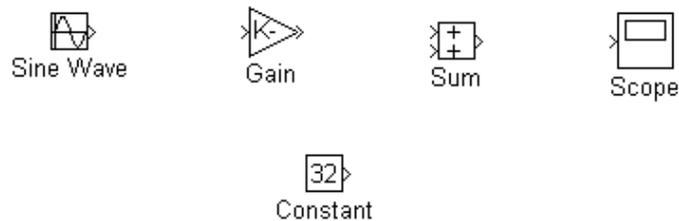
1.2.7.1. Conversión de grados centígrados en Fahrenheit

Modelemos la ecuación que convierte grados centígrados a grados Fahrenheit:

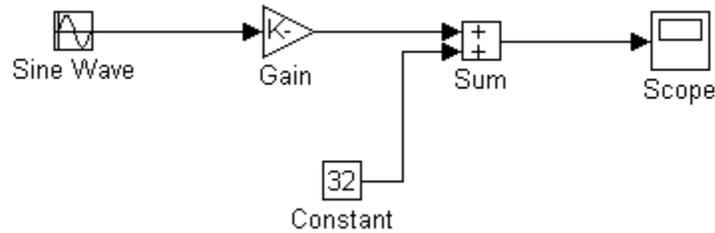
En primer lugar, consideremos los bloques que se necesitan para construir el modelo:

- Un bloque Gain de la biblioteca Linear, para multiplicar la señal de entrada por $9/5$
- Un bloque Constant de la biblioteca Sources, para definir una constante de 32
- Un bloque Sum de la biblioteca Linear, para sumar las dos cantidades
- Un bloque Sine Wave de la biblioteca Sources, para introducir la señal
- Un bloque Scope de la biblioteca Sinks, para visualizar la salida

A continuación, reúna los bloques en la ventana de su modelo:



Se asignan valores a los bloques Gain y Constant abriendo cada uno de ellos (mediante una doble pulsación) e introduciendo los valores apropiados. Luego, pulsar el botón OK. Ajustar la amplitud del bloque Sine Wave a 10 para conseguir una mayor variación de temperatura. Ahora, se debe conectar los bloques.



El bloque Sine Wave representa la temperatura en grados centígrados. El bloque Gain genera $915(T_c)$. Ese valor se suma a la constante 32 mediante el bloque Sum. La salida de ese bloque es la temperatura en grados Fahrenheit. Se debe abrir el bloque Scope para visualizar la salida. Se debe fijar la escala horizontal en algún pequeño valor de tiempo, por ejemplo 10 segundos. Luego, fijar la escala vertical de forma que se pueda visualizar todas las salidas - al menos 50. Manténgase abierto el bloque Scope.

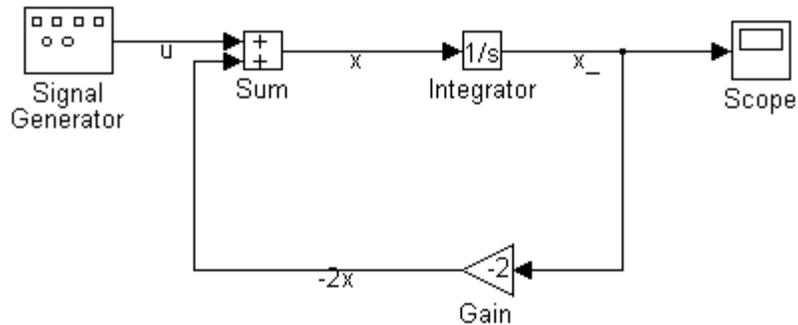
Se debe especificar los parámetros de simulación seleccionando **Parameters** en el menú **Simulation**. Especificar un tiempo de parada de 10 segundos y un tamaño de paso máximo de 0.1. Estos valores deberían ejecutar la simulación rápidamente. Ahora, se debe escoger **Start** en el menú **Simulation** para ejecutar la simulación.

1.2.7.2. Modelar un sistema continuo sencillo

Modelemos la siguiente ecuación diferencial:

$$dx/dt = -2x + u$$

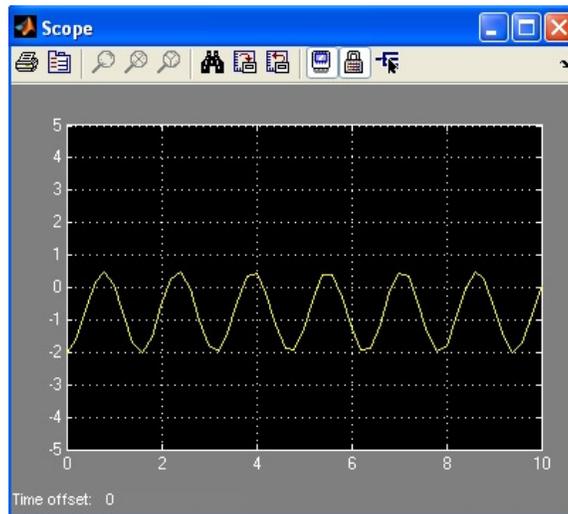
El bloque **Integrator** integra la entrada, dx/dt y produce x . Otros bloques necesitados en este modelo incluyen un bloque Gain y un bloque Sum. Para generar una onda cuadrada, se utiliza un bloque Signal Generator. Otra vez, se visualizará la salida empleando un bloque Scope. Se debe reunir los bloques y definir el valor de la ganancia del bloque Gain.



En este modelo, para invertir la dirección del bloque Gain, se escoge **Flip Horizontal** en el menú **Options**. También, para crear la línea de la salida del bloque Integrator al bloque Gain se debe mantener pulsada la tecla **Ctrl** mientras se dibuja la línea. Posteriormente, conectar todos los bloques.

Un concepto importante en este modelo es el bucle que incluye al bloque Sum, al bloque Integrator y al bloque Gain. En esta ecuación, $x_$ es la salida del bloque Integrator así como la entrada a los bloques que calculan x , sobre el cual se basa. Esta relación se implementa utilizando un bucle.

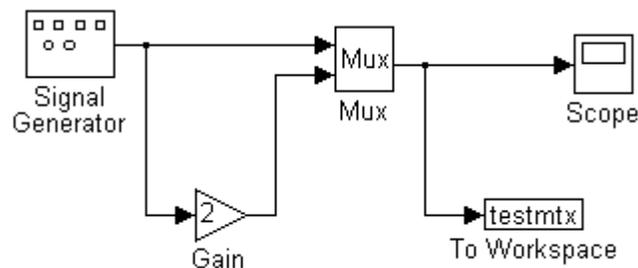
El bloque Scope visualiza $x_$ en cada paso de tiempo. Para una simulación que dura 10 segundos y que el rango vertical del bloque Scope es de 1, la salida que resulta se muestra en la figura.



1.3. Un ejercicio de construcción de un modelo

Este ejemplo muestra cómo construir un modelo utilizando muchas de los comandos y acciones que se utilizarán para desarrollar nuestros propios modelos.

El modelo genera una onda sinusoidal utilizando un bloque Signal Generator. Modifica una copia de la forma de onda pasándola a través de un bloque Gain. Ambas señales, la original y la modificada, se combinan en una señal vectorial empleando un bloque Mux. Esa señal se visualiza utilizando un bloque Scope y enviándola a una variable del espacio de trabajo. El diagrama de bloques del modelo tiene el siguiente aspecto:



Los bloques utilizados en este modelo, se obtienen de las bibliotecas siguientes:

- Biblioteca Sources (para el bloque Signal Generator)
- Biblioteca Linear (para el bloque Gain)
- Biblioteca Connections (para el bloque Mux)
- Biblioteca Sinks (para los bloques Scope y To Workspace)

Primero, generar una nueva ventana de modelo seleccionando **New** en el menú **File**.

Segundo, Abrir la biblioteca Sources para copiar el bloque Signal Generator.

Para copiar un bloque desde la biblioteca de bloques, se arrastra dentro de la ventana del modelo. Para hacer esto, se posiciona el cursor sobre el bloque Signal Generator, a continuación se presiona el botón del mouse, manteniéndolo en esa posición. El cursor cambiará.

Después, se arrastra el bloque dentro de la ventana del modelo. Cuando se mueve el bloque, se puede observar que el recuadro del bloque y su nombre se mueven con el puntero.

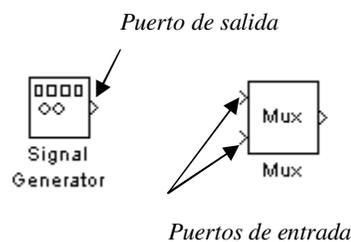
Cuando el puntero se encuentre dentro de la ventana del modelo, se debe soltar el botón del mouse. Una copia del bloque Signal Generator se encontrará en la ventana del modelo.

De la misma forma, se deben copiar el resto de los bloques dentro de la ventana del modelo. Se puede mover un bloque, desde un lugar a otro dentro de la ventana del modelo, utilizando la misma técnica de arrastre que se empleó para copiar el bloque.

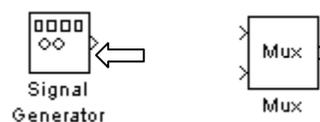
Cabe destacar que el bloque Mux tiene tres puertos de entradas pero sólo dos señales de entrada. Para ajustar el número de puertos de entrada, se debe abrir el bloque Mux dando un doble clic encima de él. SIMULINK visualiza su cuadro de diálogo. Cambia el valor del parámetro **Number of inputs** a 2, a continuación se pulsa el botón OK. SIMULINK ajusta el número de puertos de entrada.

Con todos los bloques ya copiados en la ventana del modelo, el modelo debería tener un aspecto parecido al siguiente:

Ahora se deben conectar los bloques. Si se examinan los íconos de los bloques, se verá un signo en forma de ángulo a la derecha del bloque Signal Generator y dos a la izquierda del bloque Mux. El símbolo > apuntando hacia fuera de un bloque es un puerto de salida; si el símbolo apunta hacia el bloque es un puerto de entrada. Cuando los bloques se conectan, los símbolos del puerto desaparecen.



A continuación se conecta el bloque Signal Generator al bloque Mux. Posicionando el puntero sobre el puerto de salida en el lado derecho del bloque Signal Generator.



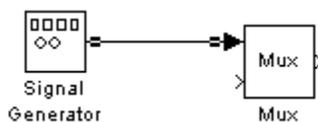
Luego se presiona y se mantiene pulsado el botón del mouse. Se observa que el botón cambia a una forma de una cruz.



Manteniendo pulsado el botón del mouse, se debe mover el cursor al puerto de entrada superior del bloque Mux o sobre el propio bloque Mux. Obsérvese que el cursor mantiene su forma de cruz y que una línea conecta el bloque Signal Generator con el puerto de entrada superior del bloque Mux.

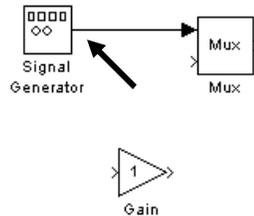


Luego, se libera el botón del mouse. Los bloques se conectan.

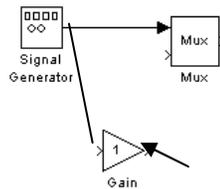


En el modelo que hay al comienzo se observa que la mayoría de las líneas conectan puertos de salida de bloques a puertos de entrada de otros bloques. Sin embargo, dos líneas conectan líneas a puertos de entrada de otros bloques. Estas líneas conectan la salida de Signal Generator al bloque Gain, y la salida Mux al bloque To Workspace y llevan la misma señal que las líneas desde las cuales se originan.

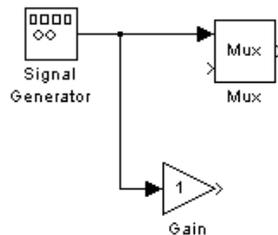
Dibujar esta clase de línea es ligeramente diferente a dibujar la línea que se acaba de dibujar. Para unir una conexión a una línea ya existente, se deben seguir estos pasos. En primer lugar, posicionar el cursor sobre la línea que hay entre los bloques Signal Generator y Mux.



Mientras se mantiene pulsado el botón del mouse, se debe presionar y mantener también pulsada la tecla ctrl. Luego, se arrastra el cursor al puerto de entrada del bloque Gain o sobre el propio bloque Gain.



Ahora, se libera el botón del mouse. SIMULINK dibuja una línea entre el punto de comienzo y el puerto de entrada del bloque gain.



De forma análoga se debe dibujar una línea similar entre la línea de salida del bloque Mux y el bloque To Workspace. Se debe acabar de realizar las conexiones. Cuando se ha finalizado de conectar los bloques, es necesario ajustar algunos de sus parámetros. En primer lugar, se debe abrir el bloque Gain y cambie el parámetro **Gain** a 2.

A continuación, se debe abrir el bloque To Workspace y modificar el parámetro **Variable name** a testmtx. Éste es el nombre de la variable del espacio de trabajo que mantendrá la salida de la simulación.

La salida por defecto del bloque signal generator es una onda sinusoidal con una amplitud de 1. Como esto es aceptable para este ejercicio, no hay necesidad de modificar ningún parámetro para este bloque.

Se debe ejecutar la simulación durante 10 segundos. Primero, se debe ajustar los parámetros de simulación seleccionando **Parameters** del menú **Simulation**. En el cuadro de diálogo que aparece, ajuste el **Stop Time** a 10 y cambie el **Maximum Step Size** a 0.1.

A continuación se debe abrir el bloque Scope para visualizar la salida de la simulación. Antes de que comience la simulación, se debe ajustar los parámetros de forma que se pueda visualizar la simulación completa. Se modifican los parámetros **Horizontal Range** (tiempo) a 10 (segundos) y **Vertical Range** (para esta simulación, corresponde a la amplitud de la onda sinusoidal) a 3.

Manteniendo abierta la ventana del bloque Scope se debe ejecutar la simulación. Se escoja **Start** en el menú Simulation y se observan las trazas del vector de entrada al bloque Scope.

Para guardar este modelo, se selecciona **Save** en el menú **File** y se especifica el nombre y la localización del archivo-M que describe el modelo. Para finalizar SIMULINK y MATLAB se debe escoja **Exit MATLAB** en el menú **File**.

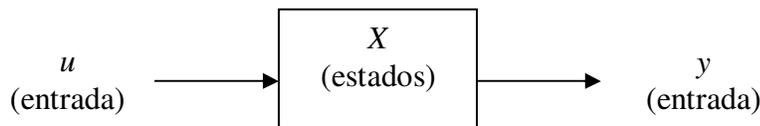
1.4. Guardar el modelo

- ✓ Se escoge la orden **save** o **save as** en el menú **File**. Simulink guarda el modelo generando un *archivo.M* que contiene las ordenes de MATLAB necesarias para recrear el modelo.
- ✓ Si se está guardando por primera vez, se debe asignar un nombre y una localización al archivo.M, y luego pulsar OK para guardar.

2. Simulación y Análisis

2.1. Cómo trabaja Simulink

Cada bloque mediante un modelo de **Simulink** tiene estas características generales: Un conjunto de entradas u , un conjunto de salidas y y un conjunto de estados x .



El vector de estado puede constar de estados continuos, estados discretos o una combinación de ambos.

La simulación consta de dos fases: inicialización y simulación. Algunas acciones tienen lugar durante la fase de inicialización.

En primer lugar, los parámetros del bloque se pasan a MATLAB para su evolución. Los valores numéricos resultantes se utilizan como los parámetros actuales de bloque.

En segundo lugar, la jerarquía del modelo se reduce a su nivel inferior. Es decir, cada subsistema se sustituye por los bloques que contiene.

En tercer lugar, los bloques se disponen en el orden en que se necesita que se actualicen. El algoritmo de ordenación construye una lista tal que cualquier bloque con alimentación directa no se actualiza hasta que se calculan los bloques que excitan sus entradas. Es durante este paso cuando se detectan los lazos algebraicos.

Finalmente, se comprueban las conexiones entre bloques para asegurar que la longitud del vector de salida de cada bloque coincide con la entrada que esperan los bloques a los que se conecta.

Ahora ya la simulación está ya preparada para poderse ejecutar.

2.1.1. Lazos algebraicos

Lazos algebraicos o implícitos ocurren cuando dos o más bloques con alimentación directa de sus entradas forman un lazo de realimentación. Cuando esto ocurre, SIMULINK debe efectuar iteraciones en cada paso para determinar si existe una solución a este problema.

Ejemplos de bloques con alimentación directa son:

- Bloques Gain
- La mayoría de los bloques no lineales
- Bloques Transfer Function, cuando el numerador y el denominador tienen el mismo orden
- Bloques Zero-Pole, cuando hay tantos ceros como polos
- Bloque State-Space, cuando hay una matriz D distinta de cero

SIMULINK informa de un error cuando no puede resolver un lazo algebraico en 200 iteraciones de una rutina de Newton-Raphson.

Para romper lazos algebraicos en lugar de permitir que SIMULINK que los resuelva de forma iterativa, inserte un bloque Memory entre dos bloques cualesquiera incluidos en el lazo.

2.2. Simulación

Se puede ejecutar una simulación seleccionando comandos desde los menús de SIMULINK o introduciéndolas desde la ventana de comandos de MATLAB.

- Seleccionar comandos desde los menús es rápido de aprender. Se puede visualizar gráficamente la conducta del sistema con bloques tipo scope.

- La introducción de comandos de simulación y análisis desde la ventana de comandos de MATLAB o desde los propios programas le permite visualizar los efectos de cambiar los bloques o parámetros de integración.

2.2.1. Parámetros de Simulación

Antes de que se ejecute una simulación, se debe especificar los parámetros de simulación y elegir el método de integración. Los parámetros de simulación incluyen:

- Tiempo de comienzo y finalización
- Tamaño del paso mínimo
- Tamaño del paso máximo
- Tolerancia o error relativo
- Variables de retorno

Cuando se ejecuta la simulación utilizando comandos del menú, se debe asignar los parámetros de simulación seleccionando **Parameters** en el menú **Simulation**, después selecciona un método de integración y rellenar los parámetros en el cuadro de dialogo de **Control Panel** (Windows).

2.2.1.1. Tiempos de comienzo y terminación

Los parámetros **Star Time** y **Stop Time** especifican los valores de t en los que la simulación comienza y termina. El tiempo de simulación y el tiempo del reloj de pared no son iguales.

La cantidad de tiempo que toma ejecutar una simulación depende de muchos factores, entre los que se incluyen la complejidad del modelo, los tamaños de paso mínimo y máximo y la velocidad del reloj del computador.

2.2.1.2. Tamaño de paso mínimo

El parámetro Minimum Step Size es el tamaño de paso utilizado al comienzo de una simulación. Los integradores no emplean un tamaño de paso por debajo de este valor cuando generan un punto de salida a menos que el sistema contenga bloques discretos con períodos de muestreo más pequeños que el tamaño de paso mínimo. Un punto de salida es un punto generado en un bloque tipo sumidero (sink), tales como los bloques Scope o To Workspace o devuelto en una trayectoria de estado o de salida. Un punto de salida se genera después que se han completado las iteraciones del método de integración. Cuando hay discontinuidades en el sistema, si se asigna un valor muy pequeño al tamaño de paso mínimo puede originar que se genere un número enorme de puntos, lo que podría malgastar memoria y recursos.

Para los métodos de integración de Adams y Gear, el tamaño de paso mínimo no afecta la precisión de la solución, pero si el número de puntos de salida generados.

2.2.1.3. Tamaño de paso máximo

Se debe asignar el tamaño de paso máximo bastante pequeño para que la simulación no deje de tomar en cuenta detalles importantes. Un tamaño de paso relativamente grande puede originar que algunos modelos se hagan inestables.

Algunas veces una simulación produce resultados que son precisos pero que no son buenos para generar gráficas suaves. En ese caso, puede ser necesario limitar el tamaño del paso máximo de forma que los resultados gráficos tengan una apariencia regular.

2.2.1.4. Tolerancia o error relativo

El parámetro **Tolerance** (Windows) controla el error relativo aceptable de la integración en cada paso. En general, este parámetro se debería fijar en el rango que va desde 0.1 a

1e-6. Cuanto más pequeño sea el valor, más pasos de integración hay que utilizar, lo que origina una simulación más precisa. No obstante, cuando la tolerancia se fija en un valor muy pequeño (1e-10) puede producir un paso tan pequeño que el error de redondeo aumenta muy significativamente.

2.2.1.5. Variables de retorno

Se pueden especificar nombres de variables en este campo para conseguir que SIMULINK escriba valores para el tiempo y las trayectorias de estado y de salida en el espacio de trabajos. La primera variable almacena el tiempo, la segunda el estado y la tercera la salida.

2.2.2. Simulación desde el menú

La ejecución de una simulación desde el menú permite efectuar ciertas operaciones de forma iterativa durante una simulación:

- Cambiar los parámetros de un bloque, a condición de que no origine un cambio en el número de estados, entradas o salidas para ese bloque.
- Cambiar cualquier parámetro de simulación excepto las variables de retorno y el comienzo.
- Cambiar el método de simulación.
- Simular al mismo tiempo otro sistema.
- Pulsar sobre una línea para ver la señal que transporta esa línea sobre un bloque Scope de tipo flotante (no conectado).

Cambios en la estructura del modelo durante una simulación, tales como añadir o suprimir líneas o bloques, originan que se pare la simulación. Se debe seleccionar otra vez **Start** para ver el resultado del cambio.

Se asignan los parámetros de simulación escogiendo **Parameters** en el menú **Simulation**. SIMULINK visualiza el cuadro de diálogo **Control Panel** (Windows) que se muestra a continuación.

Estos cuadros de diálogo permiten escoger el método de integración y definir los parámetros de simulación.

Cuando se han definido los parámetros de simulación, está ya todo preparado para iniciar la simulación. Se ejecuta una simulación seleccionando **Star** del menú de **Simulation**.

Se puede suspender una simulación que está en ejecución escogiendo **Pause** en el menú de **Simulation** y proseguir con una simulación que ha quedado en suspenso seleccionando **Continue**.

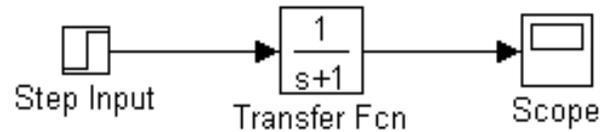
2.2.3. Visualización de las trayectorias de salida

Las trayectorias de salida de SIMULINK se pueden representar gráficamente utilizando uno de los tres métodos siguientes:

- Bloques Scope
- Variables de retorno y los comandos de representación gráfica de MATLAB
- Bloques To Workspace y los comandos de representación gráfica de MATLAB

2.2.3.1. Utilización del bloque Scope

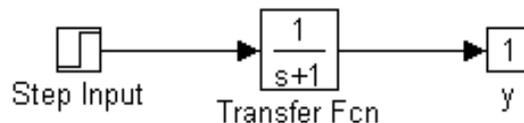
El bloque Scope se puede utilizar para visualizar trayectorias de salida mientras la simulación está en ejecución. El modelo sencillo que se muestra a continuación es un ejemplo del uso del bloque Scope.



La visualización en el bloque Scope es bastante básica; muestra la trayectoria de salida sin ninguna anotación. Los Bloques Graph proporcionan tipos de líneas y colores pero se ejecutan más lentamente que el bloque Scope.

2.2.3.2. Utilización de las variables de retorno

Devolviendo el tiempo y las historias de las salidas a MATLAB, puede utilizar los comandos de representación gráfica de MATLAB para visualizar y poner anotaciones a dichas trayectorias.



El bloque etiquetado `y` es un bloque Outport de la biblioteca Connections. La trayectoria de salida `y` se devuelve por la función de integración. Por ejemplo, llamado al sistema `tfout` e invocando a la simulación desde la línea de orden:

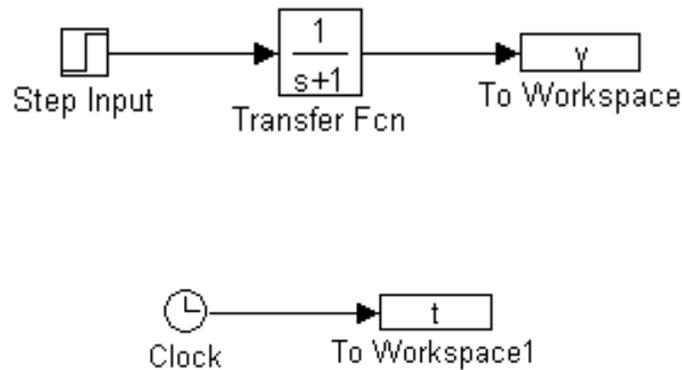
```
[t, x, y] = linsim ('tfout', 2);
```

produce historias temporales. Podría también ejecutarse esta simulación desde el menú **Simulation** especificando `[t, x, y]` como el parámetro **Return Variables**. Puede entonces representar gráficamente estos resultados utilizando:

```
plot (t, y)
```

2.2.3.3. Utilización del bloque Workspace

El bloque To Workspace se puede utilizar para devolver trayectorias de salida al espacio de trabajo de MATLAB. El modelo que se muestra a continuación ilustra este método:



Las variables **y** y **t** aparecen en el espacio de trabajo cuando se completa la simulación.

El vector de tiempos se almacena alimentando un bloque Clock en el bloque To Workspace. El vector de tiempos se puede obtener también introduciendo t en el campo **Return Variables** para simulaciones conducidas por menú o retornándola desde la función de integración:

```
t = linsim ('tfout', 2);
```

El bloque To Workspace puede aceptar una entrada vectorial donde cada trayectoria de los elementos de entrada se almacena como un vector columna en la variable del espacio de trabajo resultante.

2.2.4. Métodos de integración

La simulación de modelos SIMULINK lleva consigo la integración numérica de conjuntos de ecuaciones diferenciales ordinarias. SIMULINK proporciona una serie de métodos de integración para la simulación de tales ecuaciones:

linsim	Método que extrae la dinámica lineal
rk23	Método de Runge-Kutta de tercer orden
rk45	Método de Runge-Kutta de quinto orden
Gear	Método predictor-corrector de Gear para sistemas stiff
adams	Método predictor-corrector de Adams
euler	Método de Euler

Un sistema stiff es aquel en el que coexisten dinámicas lentas y rápidas y estas últimas alcanzan su estado estacionario.

2.2.4.1. Elección de un método

La elección del método apropiado y la selección cuidadosa de los parámetros de simulación son consideraciones importantes para obtener resultados rápidos y precisos.

El comportamiento de la simulación en términos de velocidad y precisión varía para los diferentes modelos y condiciones.

➤ **Linsim**

Linsim utiliza un método que extrae la dinámica lineal de un sistema, dejando solamente la dinámica no lineal del sistema para ser simulado. Este método funciona

muy bien cuando el sistema que se desea simular es relativamente lineal. Los modelos lineales se componen de bloques Transfer Fcn, State-Space, Zero-Pole, Sum y Gain. El método puede tomar tamaños de pasos muy grandes para tales sistemas. Por lo tanto, para obtener puntos de salida razonablemente espaciados, es necesario limitar el tamaño de paso máximo.

Linsim es particularmente bueno comparado con otros métodos cuando los bloques lineales tienen ambas dinámicas, rápidas y lentas.

➤ **rk45, rk23**

Los métodos de Rungr-Kutta, rk23 y rk45 son buenos algoritmos de propósito general que trabajan bien para un abanico amplio de problemas. Normalmente funcionan mejor que los otros métodos cuando el sistema es fuertemente no lineal y/o presenta discontinuidades. Sin embargo, no operan bien cuando el sistema tiene ambas dinámicas, rápidas y lentas. Los métodos rk23 y rk45 dan buen resultado para sistemas híbridos que contienen elementos de tiempo continuo y tiempo discreto.

Aunque rk45 es generalmente más rápido y preciso que rk23, produce menos puntos de salida; por lo tanto, rk23 puede ser la elección preferida para gráficas más regulares.

➤ **adams, gear**

Adams y gear son métodos del tipo predictor-corrector que funcionan bien en problemas donde las trayectorias de estados son regulares. El método de gear está diseñado para sistemas stiff. Es menos eficiente que otros métodos para sistemas no stiff y no da buenos resultados cuando el sistema tiene discontinuidades.

Se utiliza gear para sistemas que sean regulares y no lineales.

Se utiliza adams para sistemas que son regulares y no lineales pero que no tienen constantes de tiempo que varían en un amplio rango.

➤ **euler**

euler es una implementación del método de Euler explícito, un método de primer orden que, en general, utilizará más puntos de tiempo que los métodos de orden superior. Este método no produce resultados tan precisos como otros métodos. Debería evitar utilizarlo a menos que esté investigando la ejecución de su modelo en tiempo real.

2.2.4.2. Comparación de los métodos: Un ejemplo

El comportamiento de la simulación depende de la elección de los métodos y de los valores de los parámetros. Este ejemplo compara los métodos de simulación utilizando la ecuación de Van der Pol, un modelo no lineal simple de segundo orden:

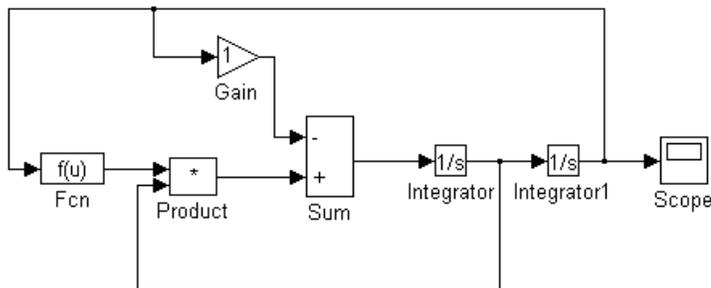
$$\ddot{x} + (x-1)\dot{x} + x = 0$$

Esta ecuación se puede describir como un conjunto de ecuaciones diferenciales de primer orden:

$$Dx_1/Dt = x_1(1-x_2^2) - x_2$$

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

Estas ecuaciones se representan como el sistema vdp, que se proporciona como una demo con SIMULINK.



En este ejemplo, comparamos los métodos de integración utilizando un valor de $1e-3$ para el parámetro de tolerancia. Los parámetros de integración se definen como:

```
tol = 1e-3;  
minstep = 1e-5;  
maxstep = 10;  
options = [tol, minstep, maxstep];  
x0 = [1;1];
```

Comparando los métodos de integración linsim, rk23 y rk45:

```
[tls, xls] = linsim ('vdp', 10, x0, options);  
[tr23, xr23] = rk23 ('vdp', 10, x0, options);  
[tr45, xr45] = rk45 ('vdp', 10, x0, options);  
plot (tls, xls, tr23, xr23, 'o', tr45, xr45, '+')
```

La representación gráfica de las tres trayectorias de estados es:

3. La biblioteca de bloques de SIMULINK

SIMULINK organiza sus bloques en bibliotecas de bloques de acuerdo con su conducta.

La ventana **Simulink** visualiza los nombres de las bibliotecas y de los iconos.

La biblioteca Sources (Fuentes) incluye bloques que originan señales. La tabla que sigue a continuación describe los bloques de la biblioteca Sources.

Nombre del bloque	Objetivo
Band-Limited White Noise	Introduce ruido blanco en un sistema continuo
Chirp Signal	Genera una onda sinusoidal de frecuencia creciente
Clock	Visualiza y proporciona el tiempo de simulación
Constant	Genera un valor constante
Digital Clock	Genera tiempo de simulación en el intervalo de muestreo especificado
From File	Lee datos de un archivo
From Workspace	Lee datos de una matriz definida en el espacio de trabajo
Pulse Generator	Genera pulsos a intervalos regulares
Random Number	Genera números aleatorios distribuidos normalmente
Repeating Sequence	Genera una señal arbitraria repetible regularmente
Signal Generator	Genera diferentes formas de ondas
Sine Wave	Genera una onda sinusoidal
Step Input	Genera una función en salto

La biblioteca Sinks (Sumidero) incluye bloques que visualizan o escriben su salida. La tabla que se muestra a continuación describe los bloques de la biblioteca Sinks.

Nombre del bloque	Objetivo
Auto-Scale Graph Scope	Visualiza señales en ventanas de figuras de MATLAB autoescaladas
Graph Scope	Visualiza señales utilizando la ventana de figuras de MATLAB
Hit Crossing	Aumenta el número de pasos de simulación en tomo a un valor especificado.
Scope	Visualiza señales durante la simulación
Stop Simulation	Para la simulación cuando la entrada es distinta de cero

To File	Escribe datos en un archivo
To Workspace	Escribe datos en una matriz en el espacio de trabajo
XY Graph Scope	Visualiza gráficas X-Y de señales en la ventana de figuras de MATLAB

La biblioteca Discrete (Discretos) contiene bloques que describen componentes de tiempo-discreto. La tabla que se muestra a continuación describe los bloques de la biblioteca Discrete.

Nombre del bloque	Objetivo
Discrete-Time Integrator	Realiza la integración en tiempo discreto de una señal
Discrete-Time Limited Integrator	Realiza la integración en tiempo discreto de una señal con límites.
Discrete State-Space	Implementa un sistema discreto en el espacio de estados
Discrete Transfer Fcn	Implementa una función de transferencia discreta
Discrete Zero-Pole	Función de transferencia discreta en términos de polos y ceros
Filter	Implementa filtros IIR y FIR
First-Order Hold	Implementa un muestreador-retenedor de orden uno
Unit Delay	Retarda una señal en un período de muestreo
Zero-Order Hold	Retenedor de orden cero de un período de muestreo

La biblioteca Linear (Lineal) contiene bloques que describen funciones lineales estándar. La tabla que se muestra a continuación describe los bloques que contiene la biblioteca Linear.

Nombre del bloque	Objetivo
Derivative	Genera la derivada respecto al tiempo de la entrada
Gain	Multiplica la entrada al bloque
Inner Product	Genera el producto escalar
Integrator	Integra una señal
Matrix Gain	Multiplica la entrada por una matriz
Slider Gain	Varía una ganancia escalar utilizando una corredera
State-Space	Implementa un sistema lineal en el espacio de estados
Sum	Genera la suma de las entradas
Transfer Fcn	Implementa una función de transferencia lineal
Zero-Pole	Función de transferencia especificada en términos de polos y ceros

La biblioteca Nonlinear (No-lineal) contiene bloques que describen funciones no lineales estándar. La tabla que se muestra a continuación describe los bloques de la biblioteca Nonlinear.

Nombre del bloque	Objetivo
Abs	Genera el valor absoluto de la entrada
Backlash	Modela la conducta de un sistema con huelgo
Combinatorial Logic	Implementa una tabla de verdad

Coulombic Friction	Discontinuidad en cero con cualquier valor de ganancia lineal
Dead Zone	Proporciona una región de salida cero
Fcn	Aplica una expresión especificada a la entrada
Limited Integrator	Integra dentro de límites especificados
Logical Operator	Realiza operaciones lógicas especificadas sobre las entradas
Look-Up Table	Realiza una transformación lineal a tramos de la entrada
MATLAB Fcn	Aplica una función de MATLAB a la entrada
Memory	Saca la entrada al bloque en el paso de integración previo
Product	Multiplifica las entradas
Quantizer	Discretiza la entrada en un intervalo especificado
Rate Limiter	Limita la velocidad de cambio de una señal
Relational Operation	Realiza las operaciones relacionales especificadas sobre la entrada
Relay	Conmuta la salida entre dos valores
Reset Integrator	Reinicializa los estados del integrador durante la simulación
Saturation	Limita el valor de una señal
Sign	Devuelve el signo de la entrada
Switch	Conmuta entre dos entradas
Transpon Delay	Retarda la entrada en una cantidad dada de tiempo
2-D Look-Up Table	Realiza una transformación lineal a tramos de dos entradas
Variable Transpon Delay	Retarda la entrada una cantidad variable de tiempo

La biblioteca Connections (Conexiones) contiene bloques que permiten multiplexación y de multiplexación, implementa Entradas/Salidas externas y crea subsistemas. La tabla que se muestra a continuación describe los bloques de la biblioteca Connections.

Nombre del bloque	Objetivo
Demux	Separa una señal vectorial en sus señales de salida
Inport	Proporciona un enlace a una entrada externa para linealización

Mux	Combina algunas líneas de entrada en una línea vectorial
Outport	Proporciona un enlace a una salida externa para linealización
Subsystem	Representa un sistema dentro de otro sistema

La biblioteca Extras contiene programas de demostración y algunas bibliotecas adicionales: la biblioteca Conversions, la biblioteca de Flip-Flops, la biblioteca PID Controllers, la biblioteca Analyzers, la biblioteca Filter y una biblioteca de bloques que se utilizan con la linealización.

CONCLUSIÓN

Simulink es un paquete para utilizar con MATLAB para modelar, simular y analizar sistemas dinámicos. Su entorno de modelación gráfico usa los familiares diagramas de bloques, de forma que los sistemas ilustrados en el texto se pueden implementar fácilmente. Su facilidad de uso y poderosa capacidad han hecho que Simulink sea la opción elegida por miles de ingenieros, profesores y estudiantes en la industria y en el mundo académico.

La extensa librería de Simulink, las herramientas analíticas y su arquitectura extensible combinan perfectamente con Matlab para proporcionar un entorno de simulación ultimado. La librería de bloques proporciona centenares de funciones predefinidas para la creación de modelos de diagramas-bloque de sistemas lineales, no lineales, de tiempo discreto, tiempo continuo, híbrido, SISO, SIMO y sistemas multitarea, de forma que cada usuario pueda crear sus propios bloques modificando los ya existentes o incorporando código Matlab, C o Fortan.

Las operaciones de modelación y simulación se controlan de forma interactiva mediante menús desplegables o empleando la línea de comando Matlab para simulaciones en modo batch. La librería no lineal incluye un complemento completo de componentes no lineales para modelar un comportamiento como sistema de mundo real. Los modelos Simulink múltiples y las capas establecidas de la jerarquía pueden permanecer abiertos simultáneamente, facilitando así las operaciones de edición de bloques y de corte-pegado.

Los parámetros de bloque se introducen en los cuadros de dialogo como escalares, vectores, o matrices empleando ya sean valores numéricos o variables y expresiones Matlab.

El gráfico ajusta automáticamente su escala a la amplitud de la señal. Estos valores pueden imprimirse desde, Simulink o ser guardados en una amplia gama de formatos de archivos gráficos para su incorporación a informes.

Para simulaciones en vivo, una variedad de bloques gráficos monitorizan la respuesta del sistema mientras avanza la simulación. Una línea gruesa identifica canales múltiples enlazados como un vector único. Las conexiones vectorizadas entre bloques son transparentes. Un bloque Hit Crossing proporciona una precisión mayor para sistemas con discontinuidades. Los gráficos desarrollados en 2D y 3D son accesibles desde Matlab para la ampliación de los resultados de la simulación. La integración con Matlab facilita el acceso inmediato a la potencia matemática, gráfica y de programación de Matlab para analizar datos, automatizar procedimientos y optimizar parámetros.

BIBLIOGRAFÍA

- <http://www.labcontrol.cl>
<http://www.guia.com.uy/44/simu44.htm>
http://www2.uhu.es/omar_sanchez/Simulink_2.htm
<http://www.dia.uned.es/Asignaturas/AutomaticaI/material/simulink.htm>
http://www2.uhu.es/omar_sanchez/Simulink_1.htm
http://216.239.33.100/search?q=cache:EPD9rISfymEC:www.software-shop.com/Productos/Matlab/Simulink/simulink.html+%22Simulink%22&hl=es&lr=lang_es
<http://www.software-shop.com/Productos/Matlab/Simulink/simulink.html>
<http://www.eldish.net/hp/automat/SIMULINK.htm>
<http://www.angelfire.com/la/hmolina/matlab7.html>
http://www.cognivision.es/ESP/ES_simulacion.htm
<http://www.puntolog.com/actual/evolu/addlink/add.htm>
http://www.procad.cl/pagina_n10.htm
<http://www.itam.mx/~dai/dsisdig/Cursos/LabSenales/sysprac0.pdf>
<http://www.esi.unav.es/asignaturas/control1/Practicas/P02/P02.HTM>
<http://www.mathworks.es/products/simulink/description/models.shtml>
<http://www.mathworks.es/products/simulink/description/overview.shtml>
<http://www.mathworks.es/products/simulink/description/simulation.shtml>
<http://www.etse.urv.es/~aoller/tutorial/tutorial.html>
<http://www.esi.unav.es/asignaturas/control1/Simulink/Ej0/ejemplo0.htm>
La Edición de estudiante de Simulink; The Math Works Inc.; Prectice-Hall; 1998; págs. 21-80.