

Tecnología de Computadores
Práctica 6
Sistemas combinacionales:
Sumadores

Universidad Miguel Hernández ¹

8 de abril de 2016

¹Copyright (c) 2016 P. Pablo Garrido Abenza. Todos los derechos reservados.



Resumen

Este documento describe la práctica número 6 de la asignatura Tecnología de Computadores de 2º del Grado en Ingeniería Informática en Tecnologías de la Información de la Universidad Miguel Hernández. Este material docente ha sido desarrollado enteramente, a menos que se indique lo contrario, por el profesor P. Pablo Garrido Abenza.

Índice general

1. Introducción	3
1.1. Objetivos	3
1.2. Software necesario	3
2. Descripción	4
2.1. Inserción de componentes con Simulín	6
3. Entrega de la práctica	8
Glosario de Acrónimos	9

Capítulo 1

Introducción

1.1. Objetivos

Tras completar esta práctica el alumno deberá ser capaz de:

- Diseñar y comprobar circuitos combinacionales de forma jerárquica, utilizando puertas lógicas básicas y otros circuitos combinacionales más elementales (componentes), que habrán sido creados previamente.
- Conocer el funcionamiento de unos circuitos combinacionales: los sumadores.

1.2. Software necesario

El software necesario para la realización de las siguientes prácticas es *Simulín* v5.61 o posterior, el cual se encuentra ya instalado en los ordenadores del aula de informática. También puede se puede instalar en cualquier otro ordenador personal, ya que este software es de libre distribución. Está disponible para Windows, Linux y Mac OS X, y se puede descargar desde el material de la asignatura, o también, desde el siguiente enlace.

Capítulo 2

Descripción

Vamos a diseñar e implementar un circuito **sumador** / **restador** para valores binarios de 4 bits. Para ello procederemos construyendo componentes de forma jerárquica de la siguiente forma:

1. Crear el componente **semisumador** (archivo: `semisum.sim`). El alumno puede utilizar cualquier diseño, de entre los propuestos en la bibliografía, o cualquier otro.
2. Utilizando el componente *semisum* anterior, crear un **sumador completo** (archivo: `sumador.sim`). El sumador completo utilizará dos componentes semisumadores. Para comprender cómo se insertan componentes previamente implementados con la herramienta de simulación *Simulín* ver la sección 2.1.
3. Utilizando sumadores completos, implementar un **sumador binario de 4 bits** (archivo: `sumador4.sim`) que sume dos valores binarios de 4 bits $A = a_3a_2a_1a_0$ y $B = b_3b_2b_1b_0$, y, teniendo en cuenta un acarreo de entrada C_{in} , proporcione un resultado $R = r_3r_2r_1r_0$ con un bit de acarreo C_{out} . El circuito tendrá, por tanto, 9 entradas (los 4 bits a sumar de cada valor: $a_3, a_2, a_1, a_0, b_3, b_2, b_1, b_0$, y el acarreo de entrada), y 5 salidas (4 para el resultado r_3, r_2, r_1, r_0 , y la del acarreo de salida). Además, necesitaremos 4 componentes sumadores completos conectados en cascada (ver los consejos de la página siguiente).



Se aconseja comprobar el correcto funcionamiento de cada componente antes de insertarlo en circuitos mayores, para evitar arrastrar errores de niveles inferiores.



A la hora de insertar los módulos de entrada o de salida en el sumador de 4-bit con *Simulín*, es necesario introducirlos según queramos que nos aparezcan en la Tabla de Verdad. El orden correcto de los módulos E/S debe ser de mayor a menor peso. Asegúrate con la herramienta 'Cambiar orden I/O' del menú 'Simulación' de *Simulín* si fuese necesario (Fig. 2.1).



Figura 2.1: Simulín - Ventana de cambio de orden módulos E/S

4. Para comprobar el correcto funcionamiento del circuito anterior haremos un nuevo circuito de test que incluya un componente `sumador4`, las entradas necesarias, y el resultado de la suma (salidas) se visualizará en un display de 7 segmentos (archivo: `sumador4_test.sim`). Notar que al tener tantos módulos de entrada no podremos obtener la Tabla de Verdad, sino que deberemos asignar valores manualmente a las entradas (click-dcho > Valor lógico inverso) y comprobar el valor obtenido.
5. Partiendo del sumador de 4 bits anterior, realizar las modificaciones oportunas para disponer de un **sumador-subtractor** dependiendo de una señal S/R, tal que cuando sea S/R=0 el circuito suma los valores de las entradas, y cuando S/R=1 los reste utilizando complemento a la base C2 (archivo: `sumsubs4.sim`).
6. Finalmente, y al igual que hemos hecho para el caso anterior, haremos un nuevo circuito de test que incluya un componente `sumsubs4`, las entradas necesarias, y el resultado de la suma/resta (salidas) se visualizará en un display de 7 segmentos (archivo: `sumsubs4_test.sim`).

2.1. Inserción de componentes con Simulín

Con *Simulín* es posible insertar componentes previamente diseñados, permitiendo realizar diseños de forma descendente o *top-down*. Los componentes son circuitos .sim estándar creados previamente, con la particularidad de que se insertan en otros circuitos mayores como bloque funcional, esto es, tan solo aparecen como una caja con las entradas y las salidas de aquel, pero sin mostrar su contenido (circuito).

Un componente se inserta con la opción 'Insertar > Circuitos combinacionales > Componente personalizado', o simplemente, pulsaremos el botón correspondiente en la barra de herramientas. Se abre una ventana en la que a la izquierda nos aparece la lista componentes que podemos insertar, que son los archivos .sim del directorio, exceptuando el actual (Fig. 2.2).

Para que un circuito pueda insertar un componente es necesario que el archivo .sim correspondiente al componente esté en el mismo directorio que el circuito con el que se está trabajando. Si lo hubiésemos grabado con cualquier en otro directorio, bastará con copiar el archivo .sim al directorio actual antes de abrir la ventana anterior.

Al seleccionar un elemento de la lista nos aparece una vista previa del componente a la derecha, como símbolo de bloque o 'caja negra'. Si queremos conocer la función lógica de las distintas salidas, podremos hacerlo pulsando el botón 'Función lógica / Nivel'. Una vez seleccionado el componente, pulsaremos 'Aceptar' para insertarlo de la forma habitual, fijándolo con el ratón en la posición deseada del circuito. Para abortar pulsaremos 'Cancelar' en esta ventana, o botón derecho del ratón si estamos colocando el componente en nuestro circuito.

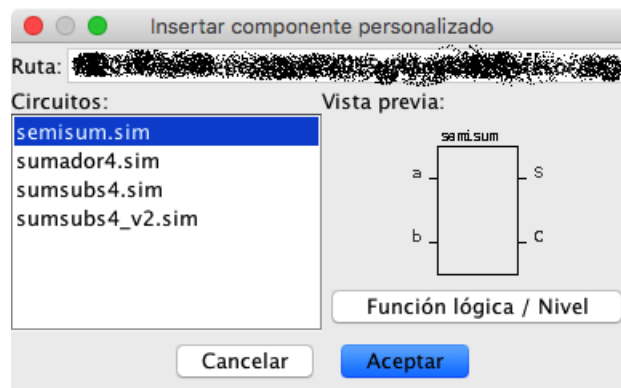


Figura 2.2: Simulín - Ventana de inserción de componentes

Una vez insertado un componente, si quisiéramos ver el circuito que implementa podemos hacerlo simplemente con click-dcho. sobre la caja del componente y elegir 'Abrir circuito' del menú contextual.

Si por algún motivo renombramos el nombre de un circuito que ya se ha insertado en otros circuitos, al intentar abrir éstos se obtendrá un error, ya que no encontrará el archivo. Por ello, si necesitamos cambiar el nombre de un componente seguiremos los siguientes pasos:

1. Hacer una copia del archivo del componente con el nuevo nombre.
2. Abrir el circuito en el que está insertado el componente.
3. Hacer click-dcho. sobre el componente insertado, y desde el menú contextual elegir 'Reasignar circuito'. En la ventana que se abrirá seleccionaremos el circuito correspondiente que tiene el nuevo nombre.
4. Finalmente, ya podremos eliminar el circuito original del componente.

Capítulo 3

Entrega de la práctica

La forma de entrega de esta práctica se realizará de la siguiente forma:

1. Comprimir en un único archivo todos los circuitos *.sim implementados. Para ello se puede utilizar programas como WinZip o 7z, admitiéndose los formatos: .zip, .7z, o .rar.
2. Subir el archivo comprimido a través del apartado *Tareas* de la página de la UMH, en su práctica correspondiente antes del plazo fijado.

Los archivos a entregar para esta práctica se resumen en la tabla 3.

#	Descripción	Archivo
1	Semisumador	<code>semisum.sim</code>
2	Sumador completo	<code>sumador.sim</code>
3	Sumador binario de 4-bits ...	<code>sumador4.sim</code>
4	... y circuito de test	<code>sumador4_test.sim</code>
5	Sumador-subtractor 4-bits	<code>sumsubs4.sim</code>
6	... y circuito de test	<code>sumsubs4_test.sim</code>

Cuadro 3.1: Archivos para los circuitos a implementar



RECUERDA: las prácticas serán corregidas por un programa informático, por lo que se insiste en que los archivos de los circuitos tengan el nombre que se especifica en las tablas, respetando incluso mayúsculas/minúsculas; el nombre del archivo comprimido que los contiene no importa.

Glosario de Acrónimos

Siglas	Significado
IC	Integrated Circuit
LSB	Least Significant Bit
MSB	Most Significant Bit