Ingeniería Técnica de Sistemas Electrónicos.
Escuela Técnica Superior de Ingeniería Industrial y de Telecomunicación.
Laboratorio de Electrónica Digital.

Práctica nº 6: Diseño de circuitos digitales secuenciales.

El objetivo de esta práctica es realizar el diseño de circuitos digitales secuenciales. El proceso recomendado para realizar y verificar los circuitos propuestos debe seguir los mismos puntos que para el diseño de circuitos combinacionales: realizar el diseño del circuito, simularlo, realizar su implementación y verificar su operación real.

Todos los ficheros de simulación y verificación del circuito deben guardarse en la carpeta Pr6 dentro de la carpeta de trabajo de cada grupo, creando subcarpetas para cada uno de los ejercicios de la práctica.

C). Diseño de un circuito secuencial con herramientas CAD.

Se desea realizar el diseño de un circuito secuencial mediante una descripción VHDL sobre el programa Quartus II para su implementación en la tarjeta DE2. La descripción del circuito como una máquina de estados puede tener muchos estados que se pueden reducir mediante algoritmos a un número finito de estados. La versión de Quartus II disponible no reduce los estados de descripción VHDL de una máquina de estados y la implementa como tal, sin embargo se dispone de herramientas de minimización de estados en Sis, por ejemplo *stamina*. Quartus II si tiene herramientas de asignación secundaria por lo que no es necesario utilizar las disponibles en Sis (Nova, Jedi, etc).

En esta práctica se debe realizar un circuito secuencial síncrono que lea los cuatro bits (E3E2E1E0) de un dígito decimal en código BCD de forma serial (un bit por ciclo de reloj) por una entrada X, comenzando por el más significativo E3, seguido de E2, E1 y el menos significativo E0 de forma que al leer el cuarto bit se muestre en la salida Z del circuito el resultado del cumplimiento de una condición (1 cumple, 0 no cumple) sobre el dígito del código leído, volviéndose a comenzar el proceso de lectura de un nuevo dígito BCD. Durante el proceso de lectura de los tres primeros bits la salida debe mantenerse a 0. La resolución lógica de este problema puede plantearse en un diagrama de estados sin minimizar en el que desde el estado inicial al llegar el primer bit se puede ir a dos estados distintos en función del valor en el primer árbol, desde cada uno de estos dos estados se pasa a otros dos estados al llegar el segundo bit, y lo mismo al llegar el tercer bit. Al llegar el cuarto bit se genera la salida y se vuelve al estado inicial. También hay que tener en cuenta que hay transiciones que no son necesarias al corresponder a combinaciones que no son dígitos del código BCD (generan don't cares en la tabla de estados). El código BCD y la condición son:

(VER HOJAS DE PRACTICA POR GRUPO)

El problema debe realizarse manualmente siguiendo los siguientes pasos:

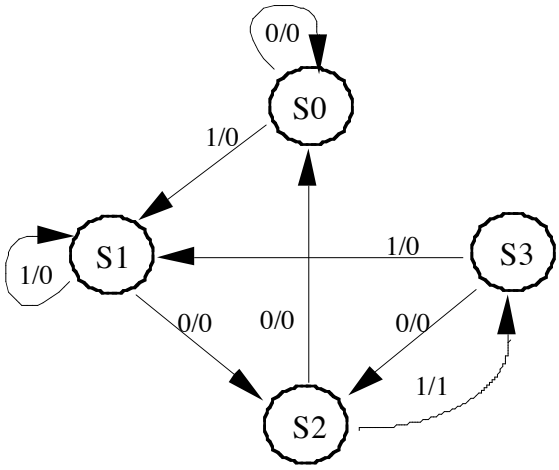
• Diseñar el diagrama de estados o la tabla de estados correspondiente al problema lógico.

• Minimizar la tabla de estados utilizando el minimizador *stamina* de Sis. Para realiza la minimización hay que seguir estos pasos:

- Abrir una ventana MS-DOS haciendo doble-click sobre el icono de nombre Sis que se encuentra en el escritorio y moverse al directorio de trabajo de cada grupo (comando MS-DOS **cd c:\Edigital\EDXXX\Pr6\...**, donde XXX es el nombre del grupo, seguir … hasta la subcarpeta de la práctica).

- Sobre la ventana MS-DOS, editar el fichero de nombre entrada (**edit entrada**) que contenga la descripción de tipo Kiss a partir del diagrama de estados propuesto en la figura. Por ejemplo la FSM representada en el diagrama de estados se corresponde con la descripción de tipo kiss.

```
.i 1
.o 1
.s 4
.p 8
.r S0
0 S0 S0 0
1 S0 S1 0
0 S1 S2 0
1 S1 S1 0
0 S2 S0 0
1 S2 S3 1
0 S3 S2 0
1 S3 S1 0
.e
```



La descripción Kiss tiene un formato:

.i N (nº de entradas N)
.o M (nº de salidas M)
.p P (nº de términos P en la descripción de la FSM).
.s S (nº de estados S)
[**.r** R (nombre del estado de reset o reinicio)]
INP PS NS OUT (líneas con los términos de descripción de las transiciones en la FSM)
.e

Las líneas de descripción de la FSM representan las transiciones entre estados asociadas su tabla de estado. Con un formato típico de una máquina de Mealy, dichas líneas están descritas por cuatro campos:

INP: valores lógicos aplicados a las N entradas de la FSM: XXX…, donde X puede tomar el valor 0, 1 o - (don't care) como en Espresso.
PS: nombre (string alfanumérico, S0, SA por ejemplo) del estado presente (o de partida) en la transición.
NS: nombre (string alfanumérico, S0, SA, por ejemplo) del siguiente estado (o de llegada) en la transición.
OUT: valores que toman las salidas de la FSM para el estado actual (PS) y las entradas (INP): YYY…, donde Y puede tomar los valores 0, 1 ó -, como en Espresso.

El estado de Reset del problema de la práctica debe corresponder con el estado en el que se lee el primer bit del dígito BCD.

** La utilización del editor MS-DOS *edit* deshabilita la opción de *scroll* de la pantalla que permite visualizar los resultados de la ejecución de las herramientas de SIS. Para evitar este problema se recomienda usar dos ventanas MS-DOS situadas en el directorio de trabajo (una sólo para el editor *edit* y otra para la ejecución de los demás comandos), o utilizar como editor el *Bloc de Notas* de Windows.

• Reducir si es posible la tabla de estados, para ello utilizar el programa *stamina* mediante la orden **stamina –v 2 -o resul entrada**, en el fichero *resul* queda almacenada en formato kiss la tabla de estados reducida cuyo contenido puede observarse mediante el comando **type resul**. La opción ***–v 2*** de esta orden permite visualizar en pantalla los resultados intermedios que genera el programa durante su ejecución. Para ver esta información hay que arrastrar con el ratón hacia arriba el botón de control de la barra vertical del lado derecho de la pantalla.
Dentro de la información se muestran los máximos compatibles (*Maximal 0, Maximal 1*, …): conjuntos máximos de estados compatibles entre ellos, y sus conjuntos de clases correspondientes (*Class set*): conjuntos de estados que deberían estar agrupados para conseguir una cobertura cerrada en la tabla de estado reducida, si se utiliza en ella el máximo compatible. También se muestran los estados seleccionados para reducir la tabla (state 0, *state 1*, …) con los conjuntos de estados de la tabla original asociados a cada uno de los estados de la tabla reducida; en el fichero **resul** los estados se nombran como S0 (*state 0*), S1 (*state 1*), etc.
Finalmente se muestra la información relativa a las estadísticas de ejecución del programa.

• Realizar una descripción VHDL del diagrama (o tabla) de estados reducido en Quartus II. La descripción debe contar con las entradas de reloj, de reset, la entrada E del dígito BCD y la salida Z del circuito. Crear un proyecto en Quartus II y compilar la descripción del circuito, eliminando los errores que se produzcan. Una vez que la descripción no tenga errores, indicar que el sistema de codificación de los estados en variables de estado mediante el número mínimo de bits posible: para ello utilizar el comando *Assignments->Settings*, seleccionar la ventana *Analysis & Synthesis Settings* y en el campo S*tate Machine Processing* seleccionar el valor Minimal Bits. Volver a compilar.
Obtener la codificación de los estados utilizando el comando *Tools->Netlist Viewers->State Machine Viewer* en la pestaña *Encoding*.

También puede observarse el diseño lógico del circuito usando el comando *Tools->Netlist Viewers->Technologu Map Viewer* (cada bloque se puede expandir a bloque más pequeños o se pueden ver las ecuaciones que realizan pinchando sobre él con el ratón: & -> AND, # -> OR, $ -> XOR, ! -> NOT).
Obtener con la herramienta de análisis temporal la máxima frecuencia de funcionamiento del circuito (comando *Processing->Timing Analyzer Tools*).

• Simular el circuito lógico en Quartus II comprobando el correcto funcionamiento del circuito para todos los dígitos BCD de entrada.

• Programar el circuito en la tarjeta DE2 y comprobar el correcto funcionamiento del mismo para todos los dígitos BCD primero usando switches, botones y leds, y luego usando el LA4240 a través de los pines de expansión del DE2, siguiendo la metodología de señales de la práctica 6.b y el método de disparo similar al de la práctica 5.a2.

IMPORTANTE: Para evitar problemas que puedan estropear la tarjeta, al asignar las entradas y salidas del proyecto hay que dejar los pines sin usar del dispositivo como entradas en alta impedancia. Para ello antes de compilar el dispositivo hay que seleccionar Assignment->Device, pulsar en el botón *Device & Pin Options…*, y en la pestaña *Unused Pins* en su campo *Reserve all unused pins:* seleccionar *As input tri-stated*.