

Práctica 5. Circuito Combinacional Decodificador 7 Segmentos.

Documentación.

La **documentación** debe ser impresa y llevada el día que se asista al laboratorio. Esta documentación debe conservarse como material de la asignatura para el alumno y en ella se deben anotar las soluciones obtenidas (además de anotarlas en **las hojas de entrega previas**) para que el alumno tenga una copia de los resultados obtenidos.

La **primera parte de la práctica** consiste en el diseño y simulación de los circuitos propuestos mediante la herramienta *Xilinx Foundation 3.1i*. **Los diseños se recogerán en las hojas de resultados previos**, que serán entregadas en el momento de acceder al laboratorio, conservando el alumno el enunciado de la práctica como material docente. Asimismo, antes de la fecha indicada al efecto, deben subirse al Campus Virtual los ficheros del proyecto en un único archivo .zip.

La **segunda parte de la práctica**, que consiste en el **montaje experimental** del circuito, **se revisará en el laboratorio** el último día de prácticas junto al montaje de la práctica 6. En esta ocasión no será necesaria la entrega de hojas de resultados posteriores puesto que serán evaluados en el propio laboratorio.

Página intencionalmente en blanco

Parte 1. Creación de los esquemáticos.

Material Necesario

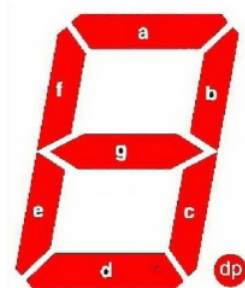
- Ordenador Personal
- Xilinx Foundation 3.1i

Objetivos

- Conocer el manejo del entorno de diseño digital Xilinx Foundation 3.1i para realizar el diseño y la simulación de circuitos combinacionales basados en bloques funcionales.
- Introducir técnicas de diseño modular para la implementación de metodologías de diseño “top-down” y “bottom-up”.
- Poner en práctica los conocimientos teóricos para la transformación de funciones combinacionales aplicando el álgebra de Boole, implementando las mismas con diferentes estrategias.

1.1. Diseño y simulación de un convertidor BCD – 7 Segmentos.

Un display 7-Segmentos es un dispositivo electrónico utilizado para mostrar caracteres alfanuméricos contruidos a partir de 7 segmentos luminosos. En el siguiente dibujo podemos observar el orden de los segmentos del display utilizado en la placa de periféricos.



Son necesarias, por tanto, 8 señales binarias distintas para iluminar cada uno de los 7 segmentos y el punto decimal. Por tanto, el **objetivo** será diseñar un convertidor de una entrada, correspondiente a un dígito BCD, a una salida de 7 segmentos más el punto decimal (8 señales), que permita controlar los displays de la placa de periféricos. Será necesario realizar el diseño utilizando el mínimo número de puertas lógicas y bloques funcionales, disponibles en la librería de la CPLD, así como realizar una macro para incorporarla en el diseño final de la práctica.

Para realizar este diseño se impondrán una serie de condiciones, que se detallan a continuación:

- Las combinaciones a utilizar para la decodificación de BCD a 7 segmentos corresponderán únicamente a las cifras del 0 al 9, el resto no se usarán.
- El punto decimal siempre estará apagado.
- Cada segmento se simplificará y se implementará por separado de distinta forma, dependiendo del DNI de cada alumno: Se sumarán todas las cifras del DNI, se dividirá el resultado por 7, y el resto de dicha división entera (operación “mod”, que dará un número entre 0 y 6) nos servirá para escoger una línea de la tabla que se muestra seguidamente, la cual nos dará la forma de implementar cada segmento del display.

Ejemplo:

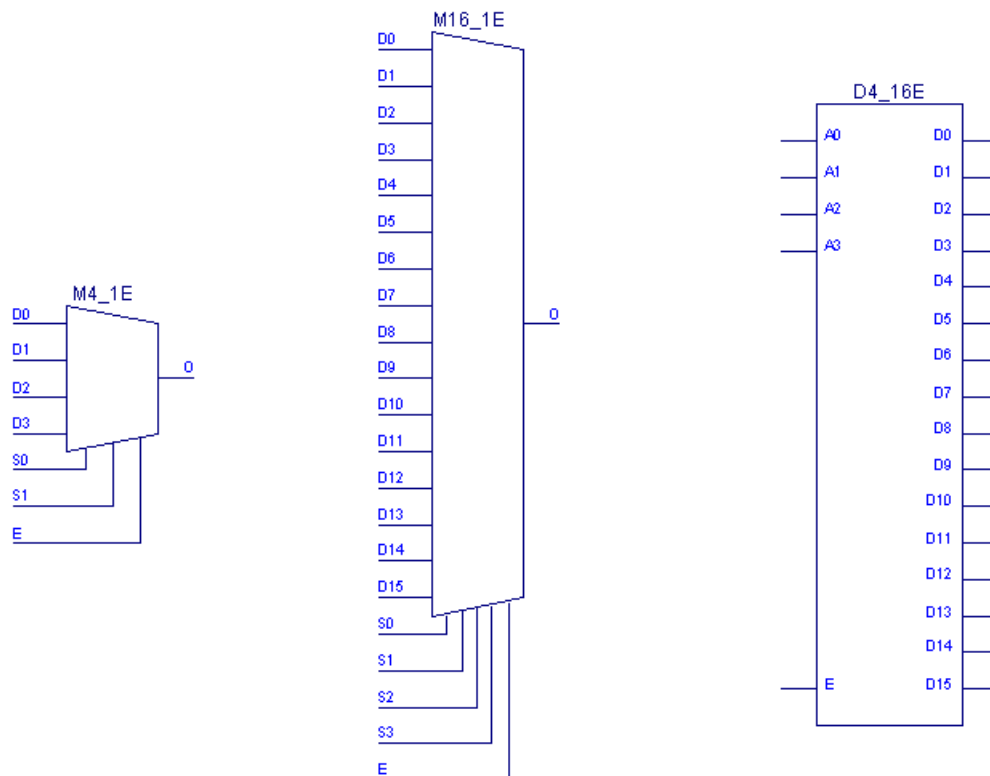
DNI = 12.345.678; Suma = 1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 = 36; $36 \bmod 7 = 1$

Escogemos la fila 1 de las 7 posibilidades expuestas en la siguiente tabla:

SEGMENTO DEL DISPLAY							
	A	B	C	D	E	F	G
DNI MOD 7	0	SM	PM	NAND	NOR	MUX4_1	MUX16_1
	1	PM	NAND	NOR	MUX4_1	MUX16_1	DEC
	2	NAND	NOR	MUX4_1	MUX16_1	DEC	SM
	3	NOR	MUX4_1	MUX16_1	DEC	SM	PM
	4	MUX4_1	MUX16_1	DEC	SM	PM	NAND
	5	MUX16_1	DEC	SM	PM	NAND	NOR
	6	DEC	SM	PM	NAND	NOR	MUX4_1

Leyenda → Forma de implementar el segmento indicado:

- **SM** → Suma Mínima de Productos. El circuito se implementará usando la mínima cantidad de inversores y puertas AND y OR de cualquier número de entradas.
- **PM** → Producto Mínimo de Sumas. El circuito se implementará usando la mínima cantidad de inversores y puertas AND y OR de cualquier número de entradas.
- **NAND** → El circuito se implementará usando la mínima cantidad de puertas NAND de cualquier número de entradas.
- **NOR** → El circuito se implementará usando la mínima cantidad de puertas NOR de cualquier número de entradas.
- **MUX4_1** → El circuito se implementará usando la mínima cantidad de Multiplexores de 4 a 1.
- **MUX16_1** → El circuito se implementará usando únicamente 1 Multiplexor de 16 a 1.
- **DEC** → El circuito se implementará usando únicamente 1 Decodificador de 4 bits a 16 y 1 puerta lógica OR con el número de entradas necesarias (según caso).



(De izquierda a derecha: Multiplexor de 4 a 1, Multiplexor de 16 a 1, y Decodificador de 4 bits a 16)

Nota: La señal “E” de los multiplexores es una señal de habilitación o “Enable”, que deberá estar a “1” lógico para que funcione correctamente el bloque.

Tabla de Verdad Decodificador 7 Segmentos:

X_3	X_2	X_1	X_0	A	B	C	D	E	F	G	DP
0	0	0	0								0
0	0	0	1								0
0	0	1	0								0
0	0	1	1								0
0	1	0	0								0
0	1	0	1								0
0	1	1	0								0
0	1	1	1								0
1	0	0	0								0
1	0	0	1								0
1	0	1	0								0
1	0	1	1								0
1	1	0	0								0
1	1	0	1								0
1	1	1	0								0
1	1	1	1								0

- SM (Suma Mínima) → Segmento

X_3, X_2	00	01	11	10
X_1, X_0				
00				
01				
11				
10				

Función Lógica:

.....

.....

.....

.....

.....

Diagrama del Circuito:

• **PM (Producto Mínimo) → Segmento**

X_3, X_2 X_1, X_0	00	01	11	10
00				
01				
11				
10				

Función Lógica:

.....

.....

.....

.....

.....

Diagrama del Circuito:

• **NAND → Segmento**

X_3, X_2 X_1, X_0	00	01	11	10
00				
01				
11				
10				

Función Lógica:

.....

.....

.....

.....

.....

Diagrama del Circuito:

• NOR → Segmento

X_3, X_2 X_1, X_0	00	01	11	10
00				
01				
11				
10				

Función Lógica:

.....

.....

.....

.....

.....

Diagrama del Circuito:

• MUX4_1 → Segmento

X_3, X_2 X_1, X_0	00	01	11	10
00				
01				
11				
10				

Función Lógica:

.....

.....

.....

.....

.....

Diagrama del Circuito:

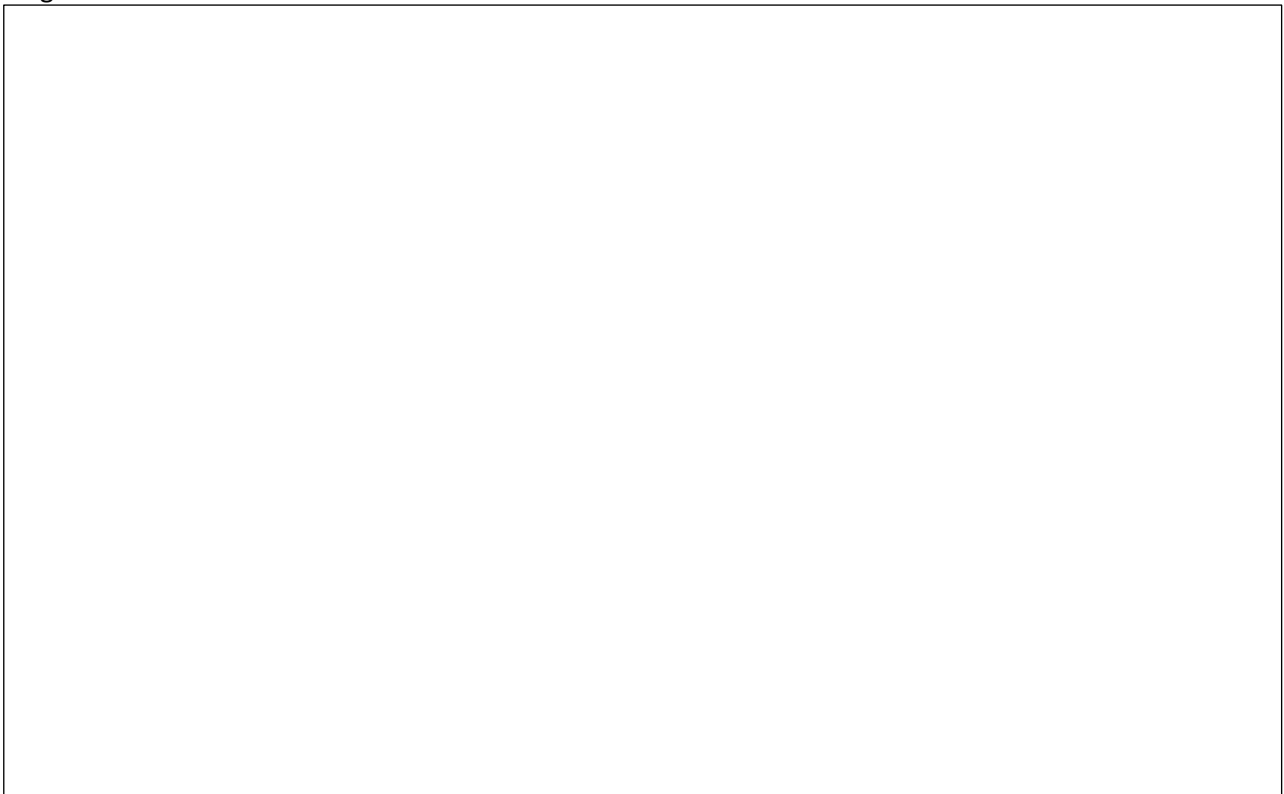
- **MUX16_1 → Segmento**

Diagrama del Circuito:



- **DEC → Segmento**

Diagrama del Circuito:



Parte 2. Montaje Experimental en Laboratorio.

Material Necesario

- Fuente de Alimentación
- Generador de funciones
- Osciloscopio
- Material de la caja del puesto: Cables, sondas, etc.
- Placa de CPLD y de Periféricos
- PC
- Cable Paralelo ↔ JTAG

Objetivos

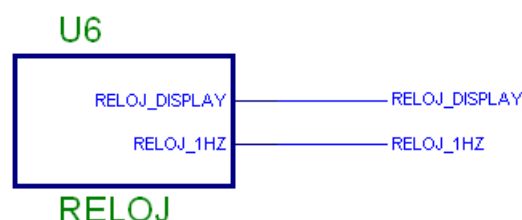
- Testear en las placas de la CPLD y de Periféricos el funcionamiento de las macros creadas en esta práctica a través del uso del software Xilinx Foundation 3.1i

2.1. Implementación del Decodificador BCD – 7 Segmentos

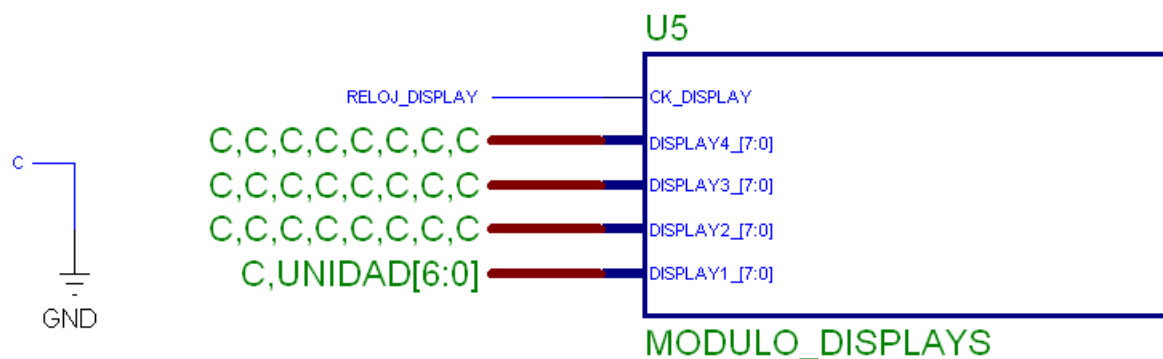
Utilizar las macros previamente diseñadas para construir un circuito que, implementado en la placa de la CPLD, utilice la placa de periféricos para mostrar en su display 1 el código BCD correspondiente a un número en binario de 4 bits del 0 al 9. El número de 4 bits será introducido usando los microinterruptores del 1 al 4 de la placa de la CPLD (siendo el microinterruptor 1 el bit más significativo).

Para poder hacer uso de la placa de Periféricos desde la placa CPLD se han realizado unas macros que deben ser copiadas a la librería local del proyecto para posteriormente ser usadas según se detalla en el tutorial de Xilinx (descargar el proyecto “MODULOS.ZIP” del Campus Virtual). Recordando un poco cuáles son estas macros, tendremos que añadir las siguientes:

1.- Módulo del reloj. Genera los relojes necesarios para las prácticas:

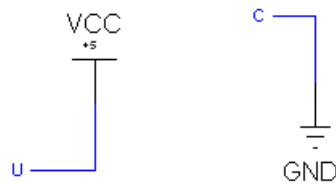


2.- Módulo de Displays 7 segmentos. En el ejemplo mostrado se apagarían todos los displays salvo el nº 1, en el cual su punto decimal (bit 8 del bus) se pone a “0” lógico de manera constante y el resto de los datos provienen del bus UNIDAD[6:0] (por ejemplo), siendo el bit 0 el segmento “A” y el bit 6 el segmento “G”:

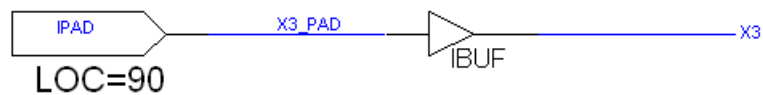


Además de estos dos módulos, será necesario añadir al circuito lo siguiente:

1.- Constantes a “0” y “1” lógico. A través del uso de los símbolos de la librería “GND” y “VCC” podemos generar estos valores lógicos, respectivamente, necesarios para construir diferentes partes de nuestro circuito. Pondremos una etiqueta a cada cable para indicar el valor lógico que llevan, C (CERO) ó U (UNO), y podremos usarlas donde se necesite:



2.- PADS de entrada para los microinterruptores 1 al 4. Será necesario añadir los correspondientes PADS e IBUFs para las cuatro señales de entrada, X_3 , X_2 , X_1 y X_0 , así como ponerle las localizaciones adecuadas a los mismos (ver tutorial de *Xilinx*). Por ejemplo, la señal X_3 debe situarse en el microinterruptor 1, cuya localización corresponde al pin 90, por lo que debería quedar en el circuito así:



Alumno: Grupo: Puesto de trabajo:

Hojas de entrega previa.

Las **hojas de entrega previa** deben ser impresas, rellenas con sus soluciones y entregadas a la entrada de la sesión de prácticas.

Parte 1. Creación de los esquemáticos.

1.1. Diseño y simulación de un convertidor BCD – 7 Segmentos.

Tabla de Verdad Decodificador 7 Segmentos:

X ₃	X ₂	X ₁	X ₀	A	B	C	D	E	F	G	DP
0	0	0	0								0
0	0	0	1								0
0	0	1	0								0
0	0	1	1								0
0	1	0	0								0
0	1	0	1								0
0	1	1	0								0
0	1	1	1								0
1	0	0	0								0
1	0	0	1								0
1	0	1	0								0
1	0	1	1								0
1	1	0	0								0
1	1	0	1								0
1	1	1	0								0
1	1	1	1								0

Alumno: Grupo: Puesto de trabajo:

• **SM (Suma Mínima) → Segmento**

X_3, X_2 X_1, X_0	00	01	11	10
00				
01				
11				
10				

Función Lógica:

.....

.....

.....

.....

.....

Diagrama del Circuito:

• **PM (Producto Mínimo) → Segmento**

X_3, X_2 X_1, X_0	00	01	11	10
00				
01				
11				
10				

Función Lógica:

.....

.....

.....

.....

.....

Diagrama del Circuito:

Alumno: Grupo: Puesto de trabajo:

• **NAND → Segmento**

X_3, X_2 X_1, X_0	00	01	11	10
00				
01				
11				
10				

Función Lógica:

.....

.....

.....

.....

.....

Diagrama del Circuito:

• **NOR → Segmento**

X_3, X_2 X_1, X_0	00	01	11	10
00				
01				
11				
10				

Función Lógica:

.....

.....

.....

.....

.....

Diagrama del Circuito:

Alumno: Grupo: Puesto de trabajo:

• MUX4_1 → Segmento

X_3, X_2 X_1, X_0	X_3, X_2			
	00	01	11	10
00				
01				
11				
10				

Función Lógica:

.....

.....

.....

.....

.....

Diagrama del Circuito:

• MUX16_1 → Segmento

Diagrama del Circuito:

Alumno: Grupo: Puesto de trabajo:

- DEC → Segmento

Diagrama del Circuito:

