

FUNDAMENTOS Y ESTRUCTURA DE COMPUTADORES

1º Graduado en Ingeniería Informática

PRÁCTICA S3. Simulación. SOLUCIÓN

SIMPLIFICACIÓN DE FUNCIONES LÓGICAS

1. Objetivos

Los objetivos que se persiguen con la realización de esta práctica pueden resumirse en los siguientes puntos:

- Comprender la metodología de diseño y simulación de sistemas digitales combinacionales mediante OrCAD 17.2 Lite.
- Asimilar los conceptos básicos de la herramienta de captura de esquemáticos, CAPTURE.
- Comprender el proceso de simulación de los sistemas digitales mediante la herramienta PSPICE 17.2 Lite.
- Asimilar la forma de describir los estímulos con buses para poder realizar la simulación.
- Mostrar el proceso para crear las trazas que visualizan el resultado de la simulación.
- Practicar con los conjuntos funcionalmente completos de puertas lógicas

2. Desarrollo de la práctica

En esta práctica vamos a diseñar un comparador de magnitud. Diseñaremos un circuito digital que tiene como entradas los 4 bits de un dígito **BCD** natural. Posee dos salidas (**x**, **y**) cuyo valor debe responder a la tabla que se adjunta. Realizad el diseño con el menor número posible de puertas lógicas básicas.

x	y	Condición
0	0	Valor menor que 5
0	1	Valor igual a 5
1	0	Valor mayor que 5
1	1	Código erróneo

Podrán realizar el ejercicio de las cuatro formas estudiadas siguientes:

- ✓ Como suma de productos, utilizando puertas NOT, AND y OR.
- ✓ Como producto de sumas, utilizando puertas NOT, OR y AND.
- ✓ Solamente con puertas NAND de cualquier número de entradas.
- ✓ Solamente con puertas NOR de cualquier número de entradas.

2.1. Capturar los diseños mediante la herramienta CAPTURE de OrCAD.

El programa OrCAD 17.2-2016 Lite está instalado en el servidor de la UCO <https://citrix.uco.es/Citrix/UCOFARMWeb/> donde el alumnado deberá ingresar con su usuario corporativo y contraseña.



Se podrán usar los circuitos integrados indicados en la siguiente tabla:

Circuito integrado	Descripción
74LS00	4 Puertas NAND de 2 entradas
74LS02	4 Puertas NOR de 2 entradas
74LS04	6 Puertas NOT (inversores)
74LS08	4 Puertas AND de 2 entradas
74LS10	3 Puertas NAND de 3 entradas
74LS11	3 Puertas AND de 3 entradas
74LS27	3 Puertas NOR de 3 entradas
74LS32	4 Puertas OR de 2 entradas

2.2. Realizar la simulación del diseño mediante la herramienta PSPICE 17.2.

- **Estímulos:**

Tiempo (ns)	BCD[3:0]			
0	0	0	0	0
200	0	0	0	1
400	0	0	1	0
600	0	0	1	1
800	0	1	0	0
1000	0	1	0	1
1200	0	1	1	0
1400	0	1	1	1

1600	1	0	0	0
1800	1	0	0	1
2000	1	0	1	0
2200	1	0	1	1
2400	1	1	0	0
2600	1	1	0	1
2800	1	1	1	0
3000	1	1	1	1

- **Señales a representar (trazas):**

✓ **Entrada BCD[3:0]:** Carácter BCD de 4 bits.

✓ **Salidas: x, y.**

3. Solución de la práctica:

- **Tabla de verdad:**

B ₃	B ₂	B ₁	B ₀	x	y	Valor
0	0	0	0	0	0	< 5
0	0	0	1	0	0	
0	0	1	0	0	0	
0	0	1	1	0	0	
0	1	0	0	0	0	= 5
0	1	0	1	0	1	
0	1	1	0	1	0	> 5
0	1	1	1	1	0	
1	0	0	0	1	0	
1	0	0	1	1	0	
1	0	1	0	1	1	No BCD
1	0	1	1	1	1	
1	1	0	0	1	1	
1	1	0	1	1	1	
1	1	1	0	1	1	
1	1	1	1	1	1	

- **Simplificación como suma de productos:**

B ₃ B ₂ \B ₁ B ₀	00	01	11	10
00				
01			1	1
11	1	1	1	1
10	1	1	1	1

$$x = B_3 + B_2 \cdot B_1$$

B ₃ B ₂ \B ₁ B ₀	00	01	11	10
00				
01		1		
11	1	1	1	1
10			1	1

$$y = B_3 \cdot B_2 + B_3 \cdot B_1 + B_2 \cdot \overline{B_1} \cdot B_0$$

- Simplificación como producto de sumas:**

$B_3B_2 \setminus B_1B_0$	00	01	11	10
00	0	0	0	0
01	0	0		
11				
10				

$$x = (B_3 + B_2) \cdot (B_3 + B_1)$$

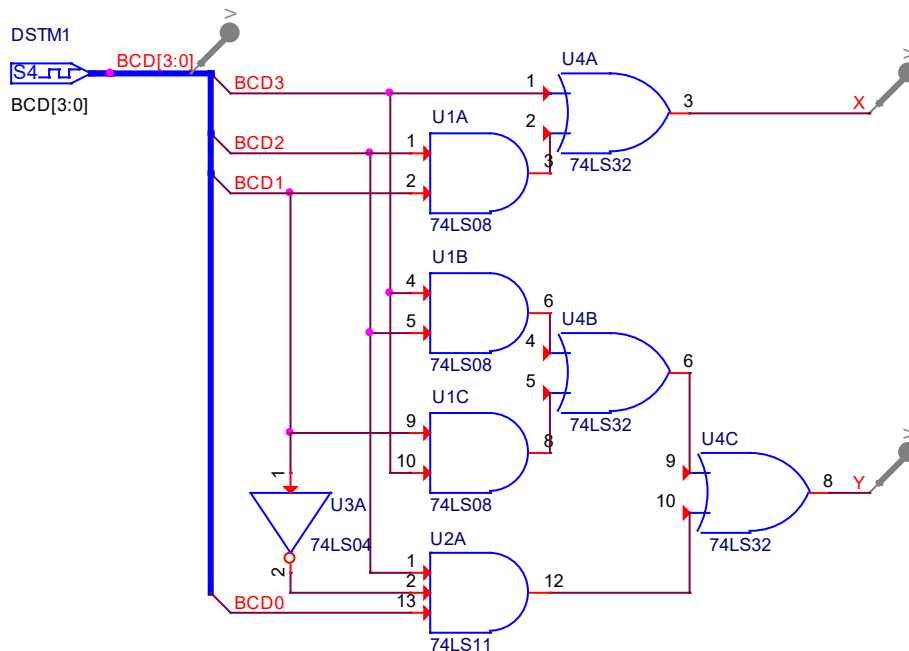
$B_3B_2 \setminus B_1B_0$	00	01	11	10
00	0	0	0	0
01	0		0	0
11				
10	0	0		

$$y = (B_3 + B_0) \cdot (B_3 + \overline{B_1}) \cdot (B_2 + B_1)$$

- Circuito como suma de productos:**

$$x = B_3 + B_2 \cdot B_1$$

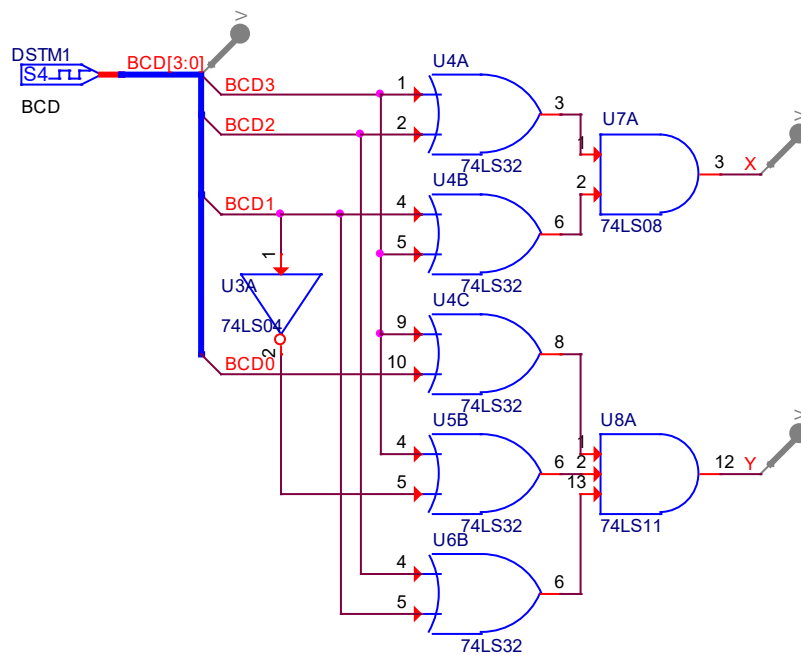
$$y = B_3 \cdot B_2 + B_3 \cdot B_1 + B_2 \cdot \overline{B_1} \cdot B_0$$



- Circuito como producto de sumas:**

$$x = (B_3 + B_2) \cdot (B_3 + B_1)$$

$$y = (B_3 + B_0) \cdot (B_3 + \overline{B_1}) \cdot (B_2 + B_1)$$

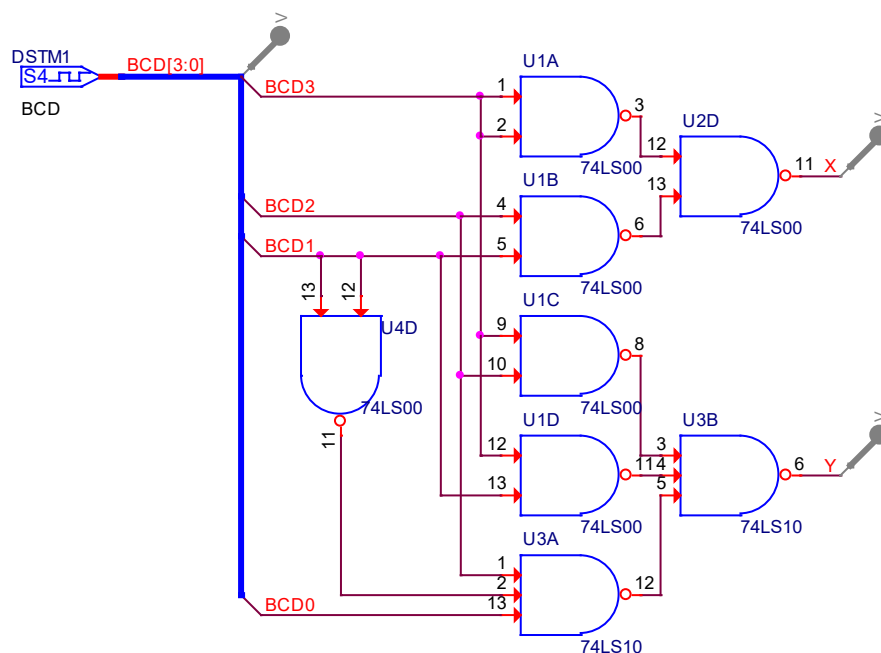


- Implementación con puertas NAND:**

Partimos de la simplificación como suma de productos. Aplicamos el teorema de involución complementando dos veces y aplicamos el teorema de De Morgan al primer complemento:

$$x = B_3 + B_2 \cdot B_1 = \overline{\overline{B_3 + B_2 \cdot B_1}} = \overline{\overline{B_3} \cdot \overline{B_2 \cdot B_1}} = \overline{\overline{B_3} \cdot \overline{B_2} \cdot \overline{B_1}}$$

$$y = B_3 \cdot B_2 + B_3 \cdot B_1 + B_2 \cdot \overline{B_1} \cdot B_0 = \overline{\overline{B_3 \cdot B_2 + B_3 \cdot B_1 + B_2 \cdot \overline{B_1} \cdot B_0}} = \overline{\overline{B_3} \cdot \overline{B_2} \cdot \overline{B_3} \cdot \overline{B_1} \cdot \overline{B_2} \cdot \overline{B_1} \cdot \overline{B_0}}$$

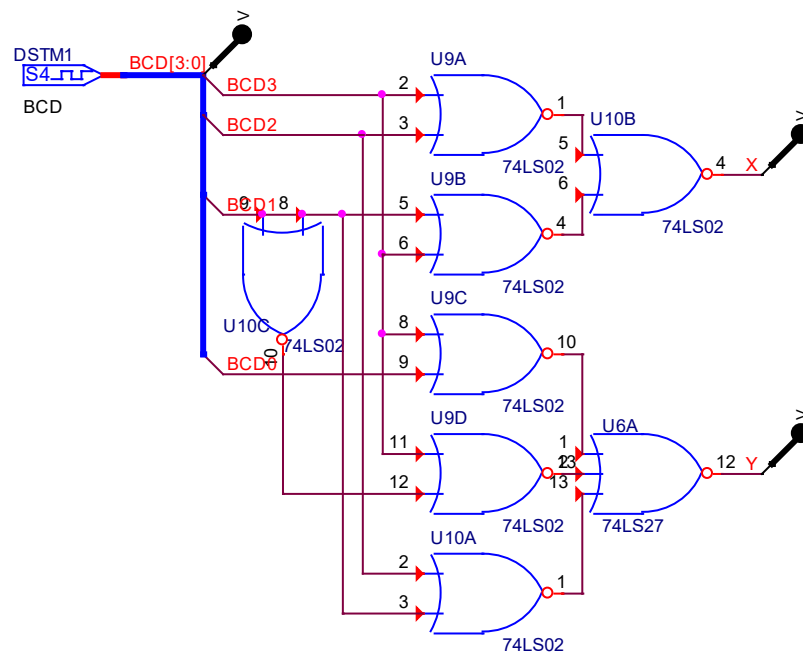


- **Implementación con puertas NOR:**

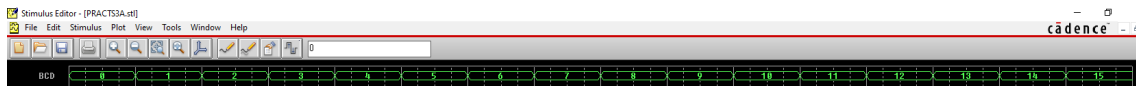
Partimos de la simplificación como producto de sumas. Aplicamos el teorema de involución complementando dos veces y aplicamos el teorema de De Morgan al primer complemento:

$$x = (B_3 + B_2) \cdot (B_3 + B_1) = \overline{\overline{(B_3 + B_2)} \cdot \overline{(B_3 + B_1)}} = \overline{\overline{(B_3 + B_2)} + \overline{(B_3 + B_1)}}$$

$$y = (B_3 + B_0) \cdot (B_3 + \overline{B_1}) \cdot (B_2 + B_1) = \overline{\overline{(B_3 + B_0)} \cdot \overline{(B_3 + \overline{B_1})} \cdot \overline{(B_2 + B_1)}} = \overline{\overline{(B_3 + B_0)} + \overline{(B_3 + \overline{B_1})} + \overline{(B_2 + B_1)}}$$



- **Estímulos de entrada:**



- **Resultado de la simulación:**

