



EXAMEN DE FUNDAMENTOS DE COMPUTADORES
CURSO 2012-13, PRIMER PARCIAL, 5 DE SEPTIEMBRE DE 2013

1.- (1 puntos) Dadas las siguientes secuencias de 12 bits, considerado como número binario entero en C_2 .

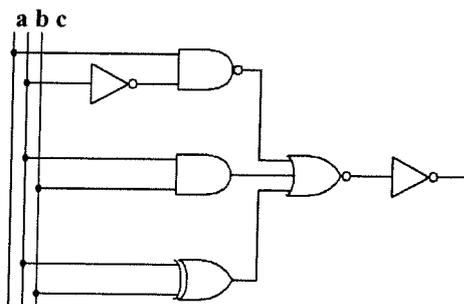
A=0110 0000 0111
B=0000 1101 0110
C=1100 1111 0011
D=1001 0000 1010

Hallar:

- El correspondiente valor en decimal.
- La representación en octal de A y B.
- La representación en hexadecimal de C y D.
- las sumas A+B, y A-D indicando si se produce ACARREO y OVERFLOW o DESBORDAMIENTO.

2.- (1 puntos): Dado el circuito de la figura

- Encontrar la expresión simplificada de la función lógica $F(a,b,c)$ que implementa.
- Mostrar la tabla de verdad correspondiente a dicha función y la expresión de la función en forma de minterms.
- Expresar con puertas NAND una función equivalente a la del circuito.



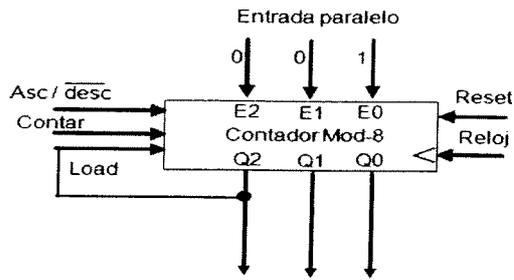
3. (3 puntos).- Sea un sistema con las siguientes características: 2 entradas (in_1, in_2) y 2 salidas (A y B). Tanto las entradas como la salida A representan números enteros sin signos de 2 bits; la salida B es de un bit. El comportamiento del sistema es el siguiente:

- En la salida A aparece siempre el número mayor de los 2 presentes en las entradas.
- Si hay dos números iguales en las entradas en la salida A aparece cualquiera de ellos.
- La salida B se pone a 1 si las dos entradas son iguales y a 0 en caso contrario.

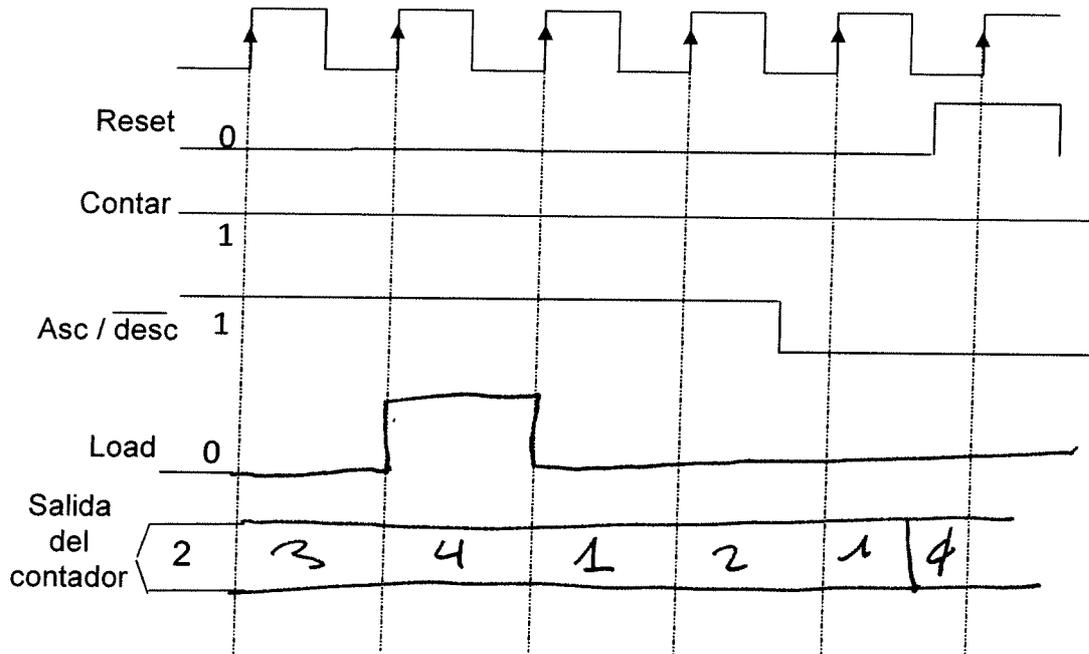
Se desea hacer una implementación modular del sistema, para ello se pide:

- Implementar un comparador de números de dos bits con dos salidas, S y B, de un bit cada una. La salida B se pondrá a 1 si y sólo si los dos números de entrada son iguales (como en la descripción de la señal B del sistema completo). La señal S tomará el valor 0 si el primer número es mayor que el segundo, el valor 1 si el segundo número es mayor que el primero. Finalmente, si los dos números son iguales S puede tomar cualquier valor.
- Utilizar el circuito del apartado a) y un multiplexor de tamaño mínimo para implementar todo el sistema.

4.- (2 puntos) Dado el contador módulo-8 de la figura

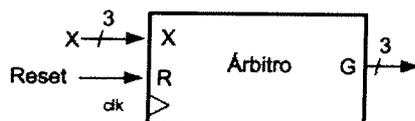


Rellenar el siguiente cronograma:



5 -(3 puntos) Se quiere diseñar un circuito digital para arbitrar el acceso de tres dispositivos a un bus, con el interfaz indicado en la figura. El árbitro recibirá una señal de petición por cada canal, $X=(x_2x_1x_0)$: si $x_i=1$ hay petición por el canal i , si es 0 no la hay. El canal 0 es más prioritario que el 1, que a su vez es más prioritario que el 2. El árbitro concederá el bus activando la correspondiente señal de grant: $G=(g_2g_1g_0)$: si $g_i=1$ se concede el uso del bus al dispositivo conectado al canal i , si es 0 el dispositivo no puede hacer uso del bus. El comportamiento del árbitro debe ser el siguiente:

- Mientras no haya petición por ninguno de los canales el árbitro permanece Inactivo, dejando las tres señales de grant a 0.
- Si estando Inactivo se recibe petición por alguno de los canales (puede haber varias peticiones simultáneas), el árbitro debe conceder el uso del bus al canal más prioritario, activando su señal de grant.
- Una vez concedido el bus a un dispositivo, la señal de grant correspondiente debe permanecer activa hasta que el dispositivo que está siendo atendido desactive la petición (incluso si otro dispositivo más prioritario solicita el uso del bus). Por ejemplo, si se ha concedido el bus al dispositivo del canal 1, la señal g_1 debe permanecer activa hasta que x_1 se ponga a 0. Cuando el dispositivo desactiva la petición el árbitro pasará a estado Inactivo para iniciar un nuevo ciclo de bus.



Se pide:

- (1,5 puntos) Especificar el sistema mediante un diagrama de estados como máquina de Moore.
- (0,5 puntos) Indicar las tablas de verdad que especifican las funciones de salida y transición de estados del sistema.
- (1 punto) Implementar el sistema mediante un registro del tamaño mínimo necesario, una ROM de tamaño mínimo para la transición de estados y un decodificador de tamaño mínimo para las salidas.

1a

A = 0110 0000 0111₂

B = 0000 1101 0110₂

C = 1100 1111 0011₂

D = 1001 0000 1010₂

(A) al ser positivo se aplica directamente la sustitución en serie

$$A = \begin{matrix} 11 & 00 & 0 & 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{matrix} = 2^{10} + 2^9 + 2^8 + 2^7 + 2^6 = 1543_{10}$$

(B) Al ser positivo se aplica directamente la sustitución en serie

B = 214₁₀

(C) al ser negativo se le cambia el signo y se le aplica la sustitución en serie

C = 1100 1111 0011

-C = 0011 0000 1100
+ 1

-C = 0011 0000 1101 \leftarrow Le aplico sustitución en serie

-C = 784₁₀ \Rightarrow C = -784₁₀

(D) al ser negativo le aplico la sustitución en serie después de cambiar el signo

D = 1001 0000 1010

-D = 0110 1111 0101
+ 1

-D = 0110 1111 0110 \leftarrow aplico la sustitución en serie
-D = 1782 D = -1782

1b) como $8 = 2^3$ se aplica la conversión entre potencias de la misma base. Se agrupan los bits de 3 en 3 empezando por la derecha y se sustituyen por su dígito octal correspondiente.

$$A = \underbrace{0110}_3 \underbrace{0000}_0 \underbrace{0111}_7 = 3007_8$$

$$B = \underbrace{0000}_3 \underbrace{1101}_2 \underbrace{0110}_6 = 0326_8$$

1c) como $16 = 2^4$ se aplica la conversión entre potencias de la misma base. Se agrupan los bits de 4 en 4 empezando por la derecha y se sustituyen por su dígito hexadecimal correspondiente.

$$C = \underbrace{1100}_C \underbrace{1111}_F \underbrace{0011}_3 = CF3_{16}$$

$$D = \underbrace{1001}_9 \underbrace{0000}_0 \underbrace{1010}_A = 90A_{16}$$

1d) A+B

$$\text{como } A = 01100000111$$

$$\text{como } B = 000011010110$$

$$A+B = 011011011101$$

~~3~~ acarreo

~~3~~ desbordamiento.

$A - D \rightarrow$ las restas se convierten en sumas (3)

$$A - D = A + (-D)$$

conocemos $A = 01100000111$

conocemos $D = 100100001010$

NO conocemos $-D \Rightarrow$ hay q. calcularlo cambiando el signo a D

$$-D = 011011110101$$

$$\begin{array}{r} 011011110101 \\ + 1 \\ \hline 011011110110 \end{array} \Rightarrow$$

$$\begin{array}{r} 011000001111 \\ 011011110110 \\ \hline \end{array}$$

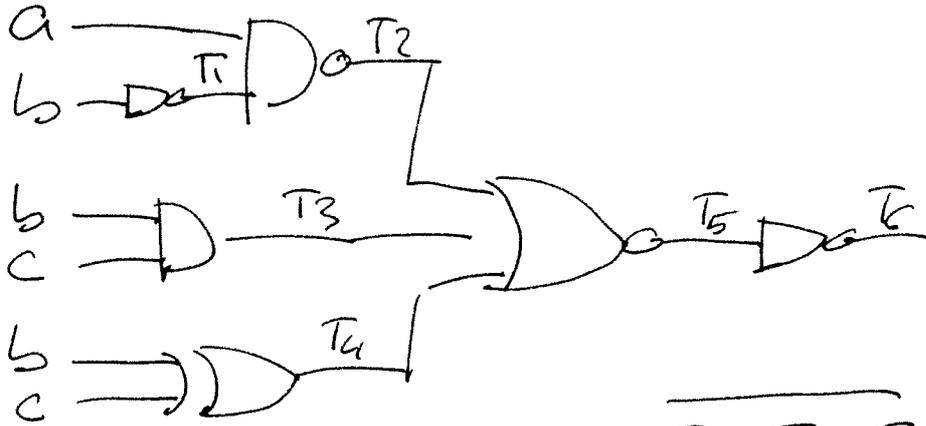
$$11001111101$$

\exists desbordamiento.

\nexists acarreo

12

11



$$T_1 = \bar{b}$$

$$T_2 = \overline{aT_1} = \overline{ab}$$

$$T_3 = bc$$

$$T_4 = b\bar{c} + \bar{b}c$$

$$T_5 = \overline{T_2 + T_3 + T_4} = \overline{\bar{b} \cdot \overline{bc} \cdot (b\bar{c} + \bar{b}c)} = \overline{\bar{b}(bc)(b\bar{c} + \bar{b}c)} = \overline{\bar{b}(\bar{b} + \bar{c})(b\bar{c} + bc)} = a\bar{b}\bar{c}$$

$$T_6 = \overline{T_5} \Rightarrow T_6 = a + b + c$$

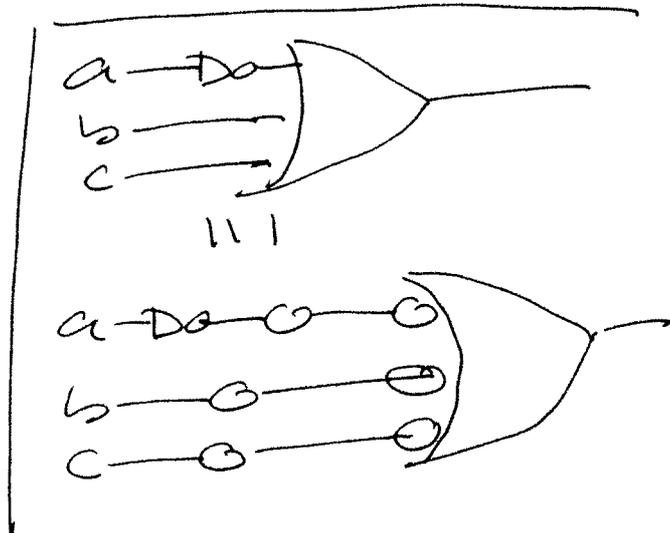
para obtener la tabla de verdad multiplico cada sumando por la variable q. le falta mas la completitud

$$\bar{a}(b + \bar{b})(c + \bar{c}) + (\bar{a} + a)b(\bar{c} + c) + (\bar{a} + a)(\bar{b} + b)c$$

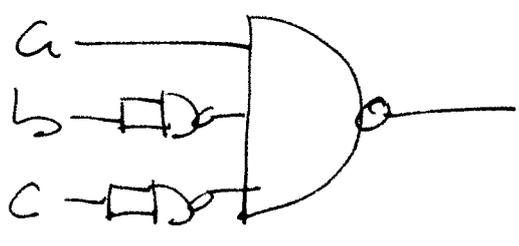
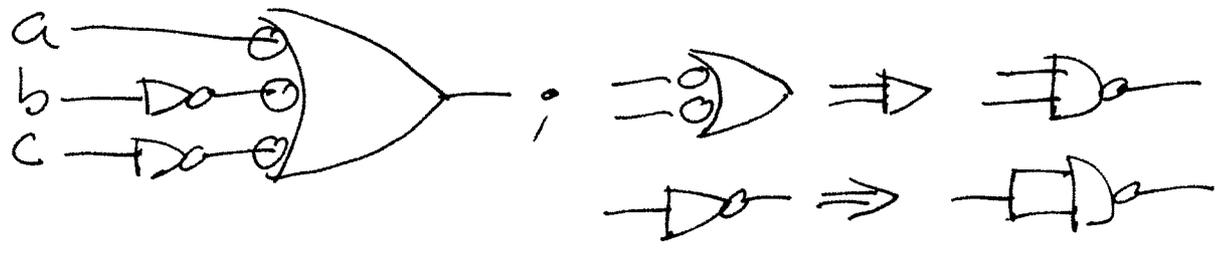
$$= \bar{a}\bar{b}\bar{c} + \bar{a}\bar{b}c + \bar{a}b\bar{c} + \bar{a}bc + a\bar{b}\bar{c} + ab\bar{c} + abc$$

$$= \sum m(0, 1, 2, 3, 4, 5, 6, 7)$$

a	b	c	
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



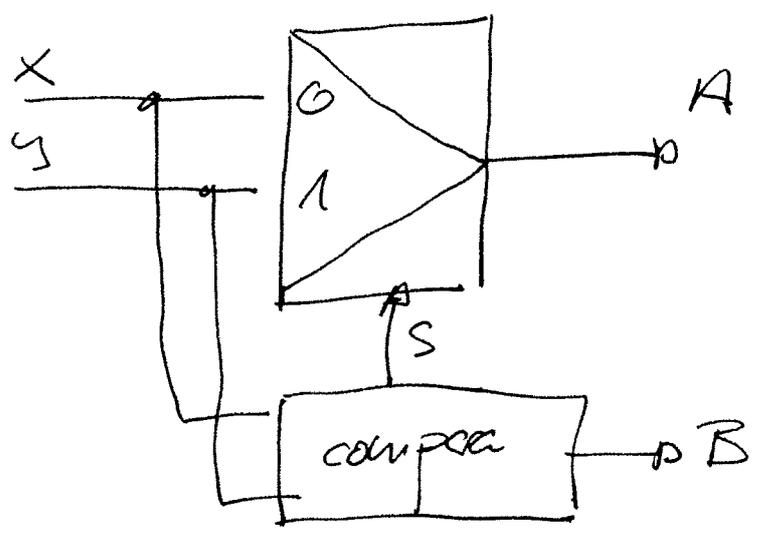
(5)



(3) El compuesto lo implementamos con portas lógicas.

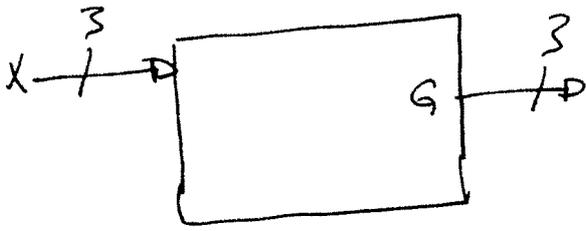
Tabla de verdad

X	Y	Z	A	B
0	0	0	0	1
0	0	1	0	0
0	1	0	1	0
0	1	1	1	0
1	0	0	0	0
1	0	1	0	1
1	1	0	1	0
1	1	1	1	0



(4) resuelto en las hojas de enunciado

(5)



(6)

- $S_0 \rightarrow$ entradas inactivas
- $S_1 \rightarrow$ Bus para canal ϕ
- $S_2 \rightarrow$ Bus para canal 1
- $S_3 \rightarrow$ Bus para canal 2

Entrada: $x_2 x_1 x_0$
 Salida: $g_2 g_1 g_0$

