

Guía de ejercicios # 1 - Lógica Digital

Organización de Computadoras 2015

UNQ

1. Escribir la tabla de verdad de las siguientes expresiones booleanas.

- a) $p \bullet q + \bar{p} \bullet \bar{q}$
- b) $(\bar{p} + q) \bullet (p + \bar{q})$
- c) $(p \bullet q \bullet w) + \bar{p} \bullet \bar{q} \bullet \bar{w}$
- d) $(p \bullet q \bullet w) + p \bullet \bar{q} \bullet \bar{w} + \bar{p} \bullet q \bullet \bar{w}$
- e) $p \bullet (q \bullet \bar{w} + \bar{q} \bullet w)$
- f) $(p + q) \bullet (p + w) \bullet (\bar{p} + \bar{q})$

2. Dadas las siguientes tablas de verdad, escribir la expresión booleana subyacente:

a)

p	q	$F(p, q)$
0	0	1
0	1	1
1	0	1
1	1	0

b)

p	q	$G(p, q)$
0	0	0
0	1	1
1	0	1
1	1	0

c)

A	B	C	$F(A, B, C)$
1	1	0	0
1	0	0	1
1	0	1	1
0	1	0	0
0	0	1	0
0	1	1	1
1	1	1	1
0	0	0	0

3. Escribir la fórmula que se obtiene con suma de productos para la siguiente tabla:

A	B	C	$F(A, B, C)$
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

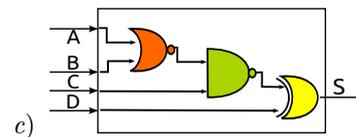
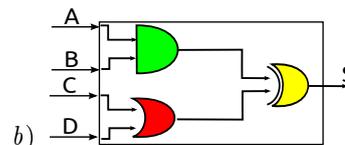
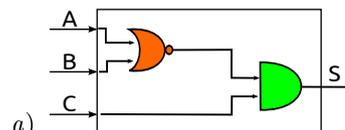
4. Escribir la fórmula que se obtiene con producto de sumas para la siguiente tabla:

A	B	C	$F(A, B, C)$
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

5. Dada la expresión booleana de verdad del ejercicio 2c:

- a) Calcular la cantidad de compuertas que requiere la implementación literal de la expresión.
- b) ¿Se puede simplificar la expresión usando propiedades del álgebra booleana? Dibujar el circuito correspondiente utilizando la menor cantidad de compuertas que pueda.

6. Construir la fórmula de verdad para los siguientes circuitos:



7. Especifique la tabla de verdad de cada circuito del ejercicio anterior.

8. Diseñe un circuito de 3 entradas y una salida que compute la función "mayoría". Es decir, que si dos o más entradas valen 1, la salida debe valer 1, y un 0 en caso contrario. Plantee la solución como una tabla de verdad y derive el circuito de esta última.

9. Diseñe un circuito de 3 entradas y una salida, cuya salida sea tal que la cantidad de unos entre las entradas y las salidas sea par (por ejemplo, para la entrada 101 la salida es 0 y para la entrada 111 la salida es 1). Plantee la solución como una tabla de verdad y derive el circuito de esta última.

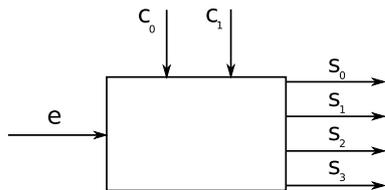
10. Haga un circuito con 4 entradas y una salida tal que la salida sea 1 si y sólo si hay exactamente 2 entradas en 1.

11. Haga el circuito de un comparador de 1 bit. Debe tener 2 entradas (a y b) y 5 salidas (S_1 , S_2 , S_3 , S_4 , S_5), tales que:

- $S_1 = 1 \leftrightarrow a > b$
- $S_2 = 1 \leftrightarrow a \geq b$
- $S_3 = 1 \leftrightarrow a = b$
- $S_4 = 1 \leftrightarrow a \leq b$
- $S_5 = 1 \leftrightarrow a < b$

12. Dibujar el circuito de un demultiplexor de 2 líneas de control, 1 línea de entrada y 4 líneas de salida. Este circuito dirige la única línea de entrada a una de las cuatro líneas de salida, dependiendo del estado de las dos líneas de control. Es decir:

- Si $c_1 = c_0 = 0$ entonces $s_0 = e$ y las restantes salidas valen 0
- Si $c_1 = 0$ y $c_0 = 1$ entonces $s_1 = e$ y las restantes salidas valen 0
- Si $c_1 = 1$ y $c_0 = 0$ entonces $s_2 = e$ y las restantes salidas valen 0
- Si $c_1 = c_0 = 1$ entonces $s_3 = e$ y las restantes salidas valen 0



13. Dibujar el circuito de un decodificador de 2 líneas de entrada (e_i) y 4 líneas de salida (s_i), cuya tabla de verdad es la siguiente:

e_1	e_0	s_3	s_2	s_1	s_0
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0

14. Usando el circuito anterior, dibujar un demultiplexor de 1 línea de entrada, 2 líneas de control y 4 líneas salidas.

15. Diseñar un *full adder* de 1 bit

16. Diseñar un restador de 1 bit:



17. Diseñar un restador con carry de 1 bit:

El mismo tiene 3 entradas, los números a restar mas el borrow de entrada (el anterior pidió 1) y tiene 2 salidas, el resultado de la resta y el borrow

18. Diseñar un *full adder* de 4 bits reusando circuitos que conozca.

19. Diseñar un *restador* de 4 bits reusando circuitos que conozca.

20. Construir un circuito que a partir de un número de 3 bits x_0, x_1, x_2 tenga 4 salidas que computen el resultado de hacer $3 * x$. Si el resultado no entra en 4 bits, debe devolver 1 en sus cuatro salidas.

Ejemplos:

- $0, 0, 0 \rightarrow 0, 0, 0, 0$ porque $3 * 0 = 0$
- $0, 0, 1 \rightarrow 3$ porque $3 * 1 = 3$
- $1, 1, 1 \rightarrow$ porque $3 * 7 = 21$ y 21 no se puede representar con 4 bits

21. Si en el ejercicio anterior no utilizó sumadores, rehacerlo nuevamente usando sumadores.

22. Un número de 4 bits es Miti-Miti si sus dos bits más significativos son iguales a los 2 menos significativos. 0101 es Miti-Miti, pero 0110 no. Se pide hacer un circuito que devuelva 1 si la cadena de entrada es Miti-Miti.