



SISTEMAS LÓGICOS

UNIDAD 2: Álgebra De Boole

Definición

Comenzaremos definiendo el Álgebra de Boole como el conjunto de elementos "B" que puede asumir dos valores posibles (0 y 1) y que están relacionados por dos operaciones binarias suma(+) y producto (*) lógico y además cumple con los siguientes postulados.

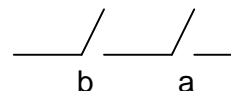
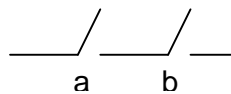
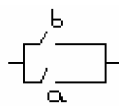
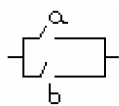
Nota: antes de comenzar a desarrollar los postulados aclararemos que la demostración de los mismos se realiza a través de un circuito eléctrico e interruptores donde el interruptor abierto representa un "0" y un interruptor cerrado representa un "1".

Postulados

Postulado N° 1: ambas operaciones son conmutativas.

$$a + b = b + a$$

$$a * b = b * a$$



Esto nos indica que el orden en que operamos con dos variables es intrascendente y el resultado es el mismo.

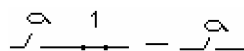
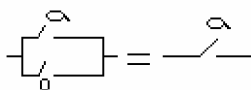
a	B	A+b	
0	0	0	No circula corriente
0	1	1	circula corriente
1	0	1	circula corriente
1	1	1	circula corriente

a	B	A*b	
0	0	0	No circula corriente
0	1	0	No circula corriente
1	0	0	No circula corriente
1	1	1	circula corriente

Postulado N° 2 : Dentro del álgebra de boole existen dos elementos neutros, el cero "0" para la suma y el "1" para el producto.

$$a + 0 = a$$

$$a * 1 = a$$





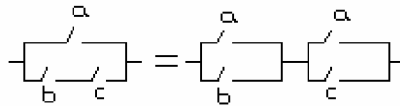
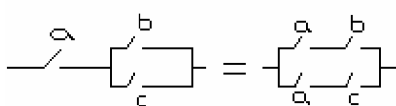
Para el caso de la suma es claro que la corriente nunca circulara por la salida abierta "0" y que la circulación de corriente dependerá exclusivamente del estado de a asimismo si comparamos con la suma ordinaria tenemos que el resultado de sumar cero a cualquier numero es el mismo numero.

Para el caso del producto una llave que esta siempre cerrada (uno) permite el paso de la corriente y el resultado final (paso o no de corriente) solo dependerá del estado de a, asimismo si comparamos con la multiplicación ordinaria tenemos que el resultado de multiplicar uno a cualquier numero el resultado es el mismo numero.

Postulado N° 3: Cada operación es distributiva respecto a la otra.

$$a * (b + c) = a * b + a * c$$

$$a + (b * c) = (a + b) * (a + c)$$



La primera ecuación distributiva es bien conocida en el álgebra Ordinaria y no debe presentar ninguna dificultad, la segunda identidad es igual que la primera.

Este postulado indica que podemos factorizar una expresión, es decir que si tenemos una expresión de dos o mas términos y estos términos tienen una (o mas) variables en común, estas variables pueden expresarse como factor común.

Ejemplos:

$$a * b * c + a * \bar{b} * c + a * b * \bar{c}$$

$$(a + b) * (a + c)$$

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

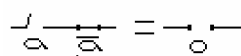
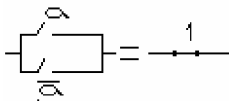
o

$$a + (b * c)$$

Postulado N° 4: para cada elemento **a** del álgebra de boole existe un elemento \bar{a} tal que:

$$a + \bar{a} = 1$$

$$a * \bar{a} = 0$$



Probablemente se observe una similitud entre estos postulados y los del álgebra ordinaria no obstante se notará que la propiedad distributiva sobre la suma no es aplicable en el álgebra ordinaria y además en el álgebra ordinaria no se cuenta con un elemento \bar{a} .

Para que el conjunto de postulados sea útil se debe cumplir que uno no contradiga a otro y que además exista independencia en los postulados es decir que ningún postulado se pueda demostrar a partir de otro.



Dualidad

Observe que los postulados se presentan de a pares si se mira cuidadosamente se observa que en cada caso, un postulado del par puede obtenerse a partir del otro intercambiando "0" por "1" y "+" por "*"

Ejemplo

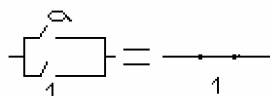
$$\begin{array}{lcl} a + 0 = a & & a + (b * c) = (a + b) * (a + c) \\ \downarrow \downarrow \downarrow & & \\ \bar{a} * 1 = \bar{a} & & \bar{a} * (\bar{b} + \bar{c}) = (\bar{a} * \bar{b}) + (\bar{a} * \bar{c}) \end{array}$$

Teoremas

Además de los postulados mencionados, para el álgebra de boole se pueden demostrar teoremas que son necesarios para el manejo conveniente del álgebra.

Para la demostración de estos teoremas nos valdremos al igual que los postulados de circuito eléctricos donde cada variable es representada por una llave que en estado normal ("1"-cerrado) deja pasar corriente y en su estado complementario ("0"-abierto) no deja circular corriente, además realizaremos las demostraciones en forma algebraica aplicando los postulados.

Teorema1: Para cada elemento del álgebra se verifica que $a + 1 = 1$



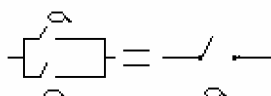
Se ve claramente que independientemente del estado de a, en el circuito siempre habrá circulación de corriente.

Para el principio de dualidad se demuestra que: $a * 0 = 0$

En forma algebraica

$$\begin{array}{ll} 1 = a + \bar{a} & \text{por postulado 4} \\ 1 = a + (\bar{a} * 1) & \text{por postulado 2} \\ 1 = (a + \bar{a}) * (a + 1) & \text{por postulado 3} \\ 1 = 1 * (a + 1) & \text{por postulado 4} \\ 1 = a + 1 & \text{por postulado 2} \end{array}$$

Teorema 2: Para cada elemento del álgebra se verifica que: $a + a = a$





Por dualidad $a * a = a$

Algebraicamente:

$$a = a + 0 \quad \text{Postulado 2}$$

$$a = a + (a * \bar{a}) \quad \text{Postulado 4}$$

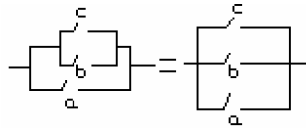
$$a = (a + a) * (a + \bar{a}) \quad \text{Postulado 3}$$

$$a = (a + a) * 1 \quad \text{Postulado 4}$$

$$a = a + a \quad \text{Postulado 2}$$

Teorema 3: Las operaciones suma y producto son asociativas

$$a + (b + c) = (a + b) + c = a + b + c$$



Algebraicamente se demuestra:

$$Z = [(a+b) + c] * [a + (b+c)]$$

$$Z = [(a+b) + c] a + [(a+b)+c] * (b+c)$$

$$Z = a + [(a+b) + c] * (b+c) \quad \text{por teorema 4}$$

$$Z = a + \{ [(a+b) + c] * b + [(a+b)+c] * c \} \quad \text{por postulado 3}$$

$$Z = a + \{ b + [(a+b) + c] * c \} \quad \text{por teorema 4 y postulado 3}$$

$$Z = a + (b + c) \quad (1)$$

También podríamos escribir:

$$Z = (a+b) * [a+(b+c)] + c*[a+(b+c)]$$

$$Z = (a+b)*[a+(b+c)] + c$$

$$Z = \{ a*[a+(b+c)] + b[a + (b+c)] \} + c$$

$$Z = \{ a*[a+(b+c)] + b \} + c$$

$$Z = (a + b) + c \quad (2)$$

Y por carácter transitivo

$$A + (b + c) = (a + b) + c$$

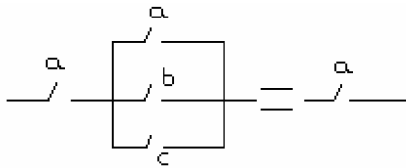
Y por dualidad

$$(a * b) * c = a * (b * c)$$



Teorema 4: Para cada elemento del álgebra de boole se demuestra que

$$a [(a+b) + c] = a$$



Como se puede ver existe circulación de corriente por el circuito solo si la llave a esta cerrada, independientemente del estado de las llaves b y c.

$$\begin{aligned} a [(a+b) + c] &= a (a+c) + ac && \text{por postulado 3} \\ &= a + ac && \text{por teorema 3} \\ &= a \end{aligned}$$

Por dualidad se cumple que $a + [(ab) c] = a$

Teorema 5: Para cualquier par de elementos del álgebra de boole se demuestra que

$$a +(ab) = a \quad \text{y} \quad a (a+b) = a$$

Algebraicamente:

$$\begin{aligned} a &= 1 * a && \text{por postulado 2} \\ a &= (a + b) * a && \text{por postulado 2} \\ a &= (1 * a) + (b * a) && \text{por postulado 3} \\ a &= a + (b * a) && \text{por postulado 2} \\ a &= a + (a * b) && \text{por postulado 1} \end{aligned}$$

La segunda igualdad se demuestra por dualidad

Teorema 6: Para todo complemento de a se verifica que

$$\overline{\overline{a}} = a$$

Teorema 7 : Para cada par de elementos del álgebra de boole se demuestra que

$$\overline{a+b} = \overline{a} * \overline{b} \quad \text{y} \quad \overline{a*b} = \overline{a} + \overline{b}$$



Estas expresiones corresponden a las leyes de Morgan, como se vera mas adelante son leyes fundamentales para la transformación de funciones.

Su demostración algebraica:

$$(a + b) + \overline{a} * \overline{b} = [(a + b) + \overline{a}] * [(a + b) + \overline{b}] \quad \text{Propiedad Distributiva}$$

$$= [a + (\overline{a} + b)] * [b + (\overline{b} + a)] \quad \text{Propiedad Conmutativa}$$

$$= [a + b + \overline{a}] * [a + b + \overline{b}] \quad \text{Propiedad asociativa}$$

$$= (1 + b) * (1 + a) \quad \text{Postulado 4}$$

$$= 1 * 1 \quad \text{Teorema 1}$$

$$(a + b) \overline{a} \overline{b} = 1 \quad (3)$$

Si consideramos a la expresión (3) de la forma: $z + \overline{z} = 1$

Donde $z = a + b$ y $\overline{z} = \overline{a} * \overline{b}$

Podemos decir que $\overline{a} * \overline{b}$ es el complemento de $a + b$

Las leyes de Morgan, se aplican fácilmente a mas de dos variables, se deja para el lector la

demostración de: $\overline{a + b + c} = \overline{a} * \overline{b} * \overline{c}$

Ejemplos:

1) Simplifique la expresión: $\overline{a} \overline{b} c + a \overline{b} \overline{c}$

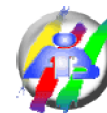
solucion:

$$\begin{aligned} \overline{a} \overline{b} c + a \overline{b} \overline{c} &= \overline{a} \overline{b} (c + \overline{c}) \\ &= \overline{a} \overline{b} * 1 \\ &= \overline{a} \overline{b} \end{aligned}$$

2) Simplifique: $(\overline{a} + b) (a + b)$

solucion:

$$\begin{aligned} (\overline{a} + b) (a + b) &= \overline{a} a + \overline{a} b + b a + b b \\ &= 0 + \overline{a} b + b a + b \\ &= \overline{a} b + a b + b \\ &= b (\overline{a} + a + 1) \\ &= b \end{aligned}$$



3) Simplifique la expresión: $\overline{(a+c)(b+d)}$

solucion:

$$\begin{aligned} &= \overline{(a+c)(b+d)} \\ &= \overline{a+c} + \overline{b+d} \\ &= a\overline{c} + \overline{b}d \end{aligned}$$

Funciones En El Algebra De Boole

Una función es una variable BINARIA cuyo valor depende de una expresión algebraica, en la que se relacionan entre si, variables binarias por medio de operaciones básicas de suma y producto lógico.

La representación es de la forma $F(a,b,c, \dots)$ y donde a, b, c, son las variables binarias que intervienen en la función y que cumplen con los teoremas y postulados del álgebra de Boole.

Una función Booleanas se puede escribir en diversas formas, cuando se expresa la forma algebraica, sin embargo la escritura en la forma estándar, facilita los procedimientos de simplificación de expresiones Booleanas. La forma estándar, contiene términos que se conocen, como términos de producto y términos de suma, en término de producto, sería $a * b * c$ y en términos de suma, sería $a + b + c$.

Se llama términos canónicos, a todo término de producto o de suma que contenga todas las variables que intervienen en la función, así en la función:

$$F(a,b,c) = a\overline{b}\overline{c} + abc + a\overline{b}c$$

Vemos que el 1° término y el 2° término, son canónicos, no así el 3° término, donde falta la variable c.

El número máximo de suma o productos canónicos será 2^n siendo n el número de variables.

Una función que posee todos los términos canónicos, se denomina Función Canónica.

Para una mayor facilidad de representación, en la función canónica, cada término puede representarse mediante el número decimal equivalente al binario, obtenida de reemplazar las variables por "0" y "1" según algún criterio. De esta manera la función lógica:

$$F(a,b,c) = a\overline{b}\overline{c} + abc + a\overline{b}c$$

Puede expresarse como:

$$F(a,b,c) = \Sigma(2,3,5)$$



c	b	a	Decimal
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

y la función lógica $F(a, b, c) = \overline{a+b+c} * \overline{a+b+c} * a+b+c$

Se puede expresar: $F_{(abc)} = \Pi(4, 2, 7)$

Los símbolos Π y Σ representan suma de productos en el primer caso, y producto de suma en el segundo.

Tabla De Verdad De Una Función Lógica

La Tabla de Verdad de una Función Lógica, es una forma de representación de la misma en la cual se indican los valores que toma la función por cada una de las combinaciones posibles de las variables que la integran.

Ejemplo:

$$f(a, b) = \overline{a} * \overline{b} + \overline{a} * b \quad \text{o} \quad F_{(a, b)} = \Sigma(0, 2) \quad (4)$$

$$f(a, b) = \overline{a} + b * a + b = 0 \quad F_{(a, b)} = \Pi(1, 3) \quad (5)$$

Decimal	B	A	Σ	Π
0	0	0	1	1
1	0	1	0	0
2	1	0	1	1
3	1	1	0	0



Veamos la ecuación (4), para ella y de acuerdo al Teorema 1 ($a + 1 = 1$) se tendrá que el valor de la función será "1", siempre que al menos un término de la función sea "1"; de la ecuación (5) podemos deducir que la función valdrá "0", si al menos un término de la función vale "0".

Una Tabla de Verdad, define una función Booleana. Una expresión algebraica que representa la función, se determina a partir de la tabla obteniendo la suma lógica de todos los términos de productos, para los cuales la función toma el valor "1".

Un término de producto, en el cual todas las variables figuran solo una vez, se denominan mini término. Los cuatro mini términos posibles, para una función de dos variables son

$$ab, a\bar{b}, \bar{a}b, \bar{a}\bar{b}$$

Por la Propiedad de Dualidad de las expresiones Booleanas podemos decir que un término de una suma que contiene todas las variables (una sola vez), se denominan Maxi términos, lo cual Maxi términos para una función de dos variables, son:

$$a + b, a + \bar{b}, \bar{a} + b, \bar{a} + \bar{b}$$

En la siguiente tabla se representan los mini términos y los maxi términos.

Para 3 variables:

<i>cba</i>	<i>miniterminos</i>		<i>maxiterminos</i>	
	<i>exp.</i>	<i>Simb.</i>	<i>exp.</i>	<i>simb.</i>
000	$\bar{a}\bar{b}\bar{c}$	m_0	$a+b+c$	M_0
001	$\bar{a}\bar{b}c$	m_1	$\bar{a}+b+c$	M_1
010	$\bar{a}b\bar{c}$	m_2	$a+\bar{b}+c$	M_2
011	$\bar{a}bc$	m_3	$\bar{a}+\bar{b}+c$	M_3
100	$a\bar{b}\bar{c}$	m_4	$a+b+\bar{c}$	M_4
101	$a\bar{b}c$	m_5	$\bar{a}+b+\bar{c}$	M_5
110	$a\bar{b}\bar{c}$	m_6	$a+\bar{b}+\bar{c}$	M_6
111	abc	m_7	$\bar{a}+\bar{b}+c$	M_7

Si a una función Booleana la expresamos a partir de una tabla de Verdad:

<i>cba</i>	F	F
000	1	0
001	0	1
010	1	0
011	0	1
100	0	1
101	1	0
110	0	1
111	1	0



Si consideramos ahora el complemento de la función F:

$$f(a, b, c) = \bar{a}\bar{b}\bar{c} + \bar{a}b\bar{c} + a\bar{b}c + abc$$

$$= \Sigma(0, 2, 5, 7) = m_0 + m_2 + m_5 + m_7$$

Si consideramos ahora el complemento de la función (\bar{F})

$$\bar{f}(a, b, c) = a\bar{b}\bar{c} + ab\bar{c} + \bar{a}b\bar{c} + \bar{a}b\bar{c}$$

$$= \Pi(1, 3, 4, 6) = M_1 + M_3 + M_4 + M_6$$

Si a partir de este complemento tenemos la función F:

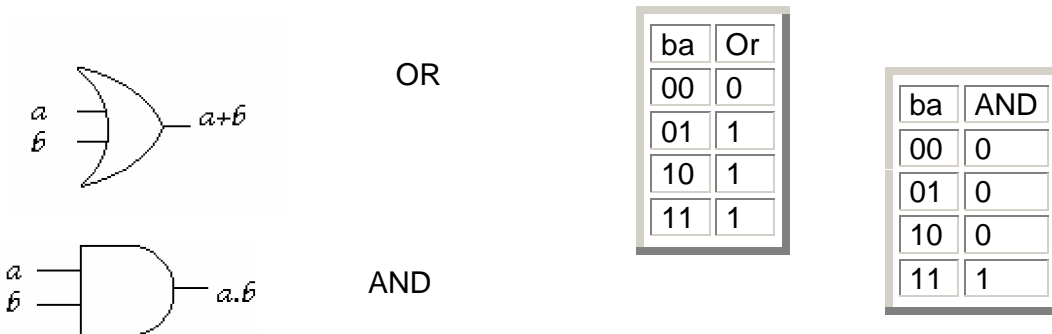
$$F = \overline{m_1 + m_3 + m_4 + m_6} = \overline{m_1 + m_3 + m_4 + m_6} = M_1 + M_3 + M_4 + M_6$$

ya que $m_j = M_j$

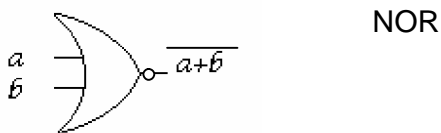
$$F = \bar{a} + \bar{b} + c \cdot \bar{a} + \bar{b} + c \cdot a + \bar{b} + c \cdot a + \bar{b} + c$$

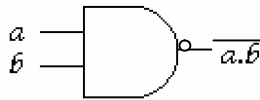
$$F = \Pi(1, 3, 4, 6)$$

Las funciones básicas del álgebra de Boole, suma lógica y producto lógico, son llamados Función OR y Función AND respectivamente y se simboliza de la forma:



Para la aplicación del Teorema de Morgan, se definen dos nuevas funciones, llamadas NOR (OR negada) y NAND (AND Negada) y se representan:





NAND

ba	NOR
00	1
01	0
10	0
11	0

ba	NAND
00	1
01	1
10	1
11	0

TABLAS DE VERDAD: TABLA DONDE SE REPRESENTA EL FUNCIONAMIENTO DE UN CIRCUITO COMBINACIONAL, DE TAL MANERA QUE SE VAN ESCRIBIENDO LOS VALORES QUE TOMA LA SALIDA DEL SISTEMA PARA CADA UNA DE LAS COMBINACIONES DE LAS VARIABLES DE ENTRADA.

Reducción De Funciones

La complejidad de un circuito digital, que ejecuta una función Booleana está relacionada directamente con la expresión algebraica que la representa.

Las expresiones Booleanas pueden ser simplificadas por manipulación algebraicas, como se ha visto en varios ejercicios de este capítulo.

El método de mapa de Karnaugh, ofrece un procedimiento directo, para simplificar expresiones Booleanas, de hasta cinco variables, se puede trazar mapas por mayor cantidad de variables pero son mas difíciles de usar.

El mapa de Karnaugh es una herramienta muy útil para el diseñador lógico ya que permite reducir en forma sencilla una función lógica.

Este método tabular se basa en el cuarto postulado. $a + \bar{a} = 1$ por lo que se puede escribir :

$$\begin{aligned}
 F(a,b,c) &= a \bar{b} c + a \bar{b} \bar{c} + a b c + a \bar{b} \bar{c} \\
 &= a \bar{b} (c + \bar{c}) + a \bar{b} c + a \bar{b} \bar{c} \\
 &= a \bar{b} 1 + a \bar{b} c + a \bar{b} \bar{c} \\
 &= a \bar{b} + a \bar{b} c + a \bar{b} \bar{c}
 \end{aligned}$$

Aplicando Propiedad Asociativa.

Nótese que el término c que se encontraba en su forma directa, e inversa se puede eliminar.

Se deja al lector que continúe con la reducción de esta función.

El método tabular de Karnaugh, para la reducción de funciones consiste en representar la tabla de verdad de la función (*deberá estar escrita en forma canónica*), en un gráfico como la figura siguiente:



Si suponemos una función de cuatro variables a, b, c, d . Se indicarán dos de ellas en las columnas (a, b) y dos de ellas (c, d) en las filas, de tal forma que al pasar de una columna a la siguiente y de una fila a la siguiente solo debe cambiar un bit.

ab \ cd	00	01	11	10
00→			↓
01				
11				
10				

Al desplazarnos, solo cambia el bit d

Al desplazarnos, solo cambia el bit b

Para función de dos o tres variables, la tabla tomará la siguiente forma:

a \ b	0	1
0		
1		

a \ bc	0	1
00		
01		
11		
10		

En la celda de la tabla se colocará un 1 para cada término de la función, que tome el valor uno, (si la función está expresada como suma de producto), o que tome el valor cero, (si la función está expresada como producto de sumas)

Para una mejor comprensión de lo expresado veamos los siguientes ejemplos:

Ejemplo

$$F = a.\bar{b}.c + a.b.c + a.b.\bar{c} + \bar{a}.b.c$$

Reducir la función lógica :

Tabla de Verdad de la función

abc	F
000	0
001	0
101	0
011	1
100	0
101	1
110	1
111	1

Tabla de Karnaugh

a \ bc	0	1
00		
01		1
11	1	
10	1	1



El siguiente paso para lograr la reducción de la función es formar grupos de unos, cada grupo debe tener 32, 16, 8, 4, 2, etc., cantidad de unos y se comenzará por formar los grupos con mayor cantidad de unos posibles. Cada grupo debe cumplir la condición de que para pasar de un uno a otro continuo (a derecha e izquierda o arriba y abajo) solo debe cambiar un bit.

En nuestro ejemplo.

a \ bc	0	1
00		
01		1
11	1	
10	1	1

Este grupo no corresponde, nótese que al pasar de un 1 a otro 1, cambian dos bit:
La variable b cambia de 0 a 1
La variable c cambia de 1 a 0

Los grupos que aquí se pueden formar son dos (rojo y azul), no se pueden formar grupos de 4 ya que no se estaría cumpliendo con la condición de que varía solo un bit al desplazarnos de un uno a otro, tanto en forma horizontal como vertical.

A los efectos de aclarar un poco más, veamos como sería un grupo de cuatro elementos.

a \ bc	0	1
00	1	1
01		1
11	1	1
10	1	1

Como se puede apreciar aquí hemos formado dos grupos de 4 unos nótese que el grupo azul está formado por los elementos de la primera y última fila de la tabla existe la posibilidad de formar un tercer grupo de cuatro elementos, esta tarea se deja para el lector.

Ya formamos los grupos ahora corresponde escribir la nueva función reducida. Esto se hace de una manera sencilla donde cada grupo o término independiente será uno de los términos de la función y dentro de cada término se escribirá la variable que no varíe y se desecha lo que cambia.

Para nuestro caso:



a \ bc	00	01	11	10
00				
01			1	
11			1	
10			1	1

$\overline{a} \cdot \overline{b} \cdot c$
 $\overline{a} \cdot b$
 $b \cdot c$

$$f(a,b,c) = \overline{a} \cdot b + b \cdot \overline{c} + a \cdot \overline{b} \cdot c$$

En resumen, los pasos a seguir para reducir una función, por el método tabular de Karnaugh son:

- Confeccionar una tabla según la cantidad de variables de la función.
- Marcar dentro de la tabla aquellas celdas para las que la función toma el valor 1.
- Formar grupos con las celdas marcadas.
- Escribir la nueva función reducida.

Cabe aclarar que este método es muy sencillo para funciones de hasta cuatro variables, para funciones de cinco variables, se deben formar dos tablas de cuatro variables, una para cuando la quinta variable es '0', y otra para cuando la quinta variable es '1'. Para armar la función reducida, se tomarán los grupos y elementos aislados de ambas tablas.

Ejemplo

Para $e = 0$

ab \ cd	00	01	11	10
00				
01		1	1	1
11			1	1
10				

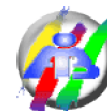
$$f(a,b,c) = \overline{a} \cdot \overline{b} \cdot \overline{c} \cdot \overline{d} \cdot \overline{e} + a \cdot \overline{d} \cdot \overline{e} + \overline{a} \cdot \overline{b} \cdot c \cdot e$$

Para $e = 1$

ab \ cd	00	01	11	10
00				
01				
11	1			
10	1			

Funciones Incompletamente Especificadas

La tabla de verdad de una función traduce una descripción verbal de un sistema para luego encontrarla función aritmética (modelo matemático) que realice ese sistema. Hasta ahora se ha supuesto que la descripción verbal contempla todas las situaciones posibles, es decir, contempla las 2^n posibles combinaciones de la entrada (n = cantidad de variables).



En algunas ocasiones no se detallan (describen) todas las posibles combinaciones de la entrada sino que ciertas combinaciones directamente no son analizadas porque no interesan al sistema sencillamente porque no se van a dar en la práctica.

Siempre que estemos seguros de que cierta condición de las entradas no se darán nunca es evidente que poco importa definir la salida como un uno o como un cero.

Cuando se presentan estas circunstancias se dice que la función está incompletamente especificada y en la tabla de verdad se indica colocando una X en lugar de un 0 o un 1.

Esta X podrá ser tomada como 0 o como 1 en la tabla de Karnaugh, según convenga para lograr una mayor reducción de la función.

ab \ cd	00	01	11	10
00				
01		1	1	1
11			1	x
10				

Para este caso conviene que la X asuma el valor 1 ya que así podremos formar un grupo de cuatro elementos y una Mayor simplificación de la función.

ab \ cd	00	01	11	10
00		x		
01				
11	1			
10	1			

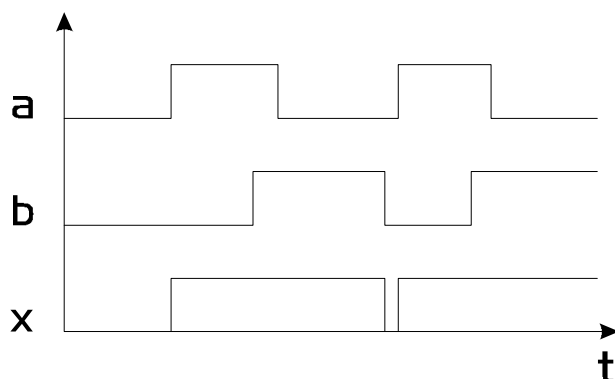
Para este caso donde la X no puede formar grupos con otros unos, conviene que la X asuma el valor 0

Compuertas Lógicas:

Son circuitos que implementan las diferentes operaciones que existen en el álgebra de Boole.

- ü suma lógica à función OR
- ü producto lógico à función AND
- ü negación à función INV

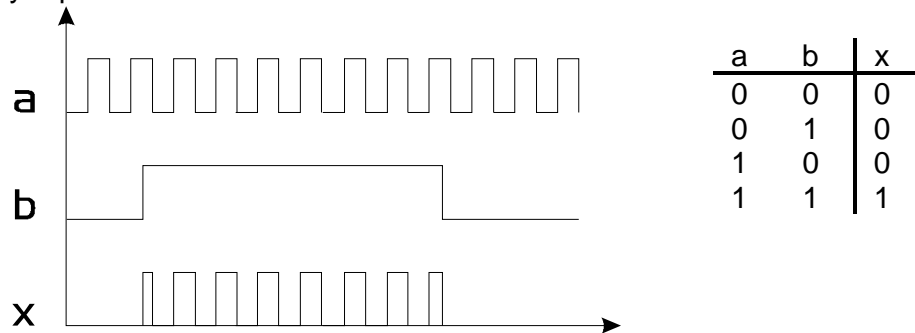
Problema: aplicar las formas de onda siguientes a las entradas 'a' y 'b' de una puerta OR y representar la forma de onda de su salida 'x'.



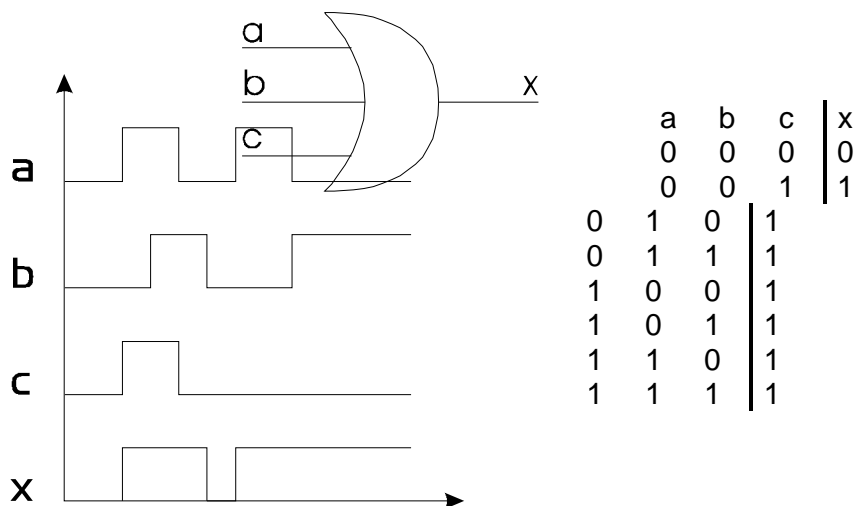
A	b	x
0	0	0
0	1	1
1	0	1
1	1	1



Problema: aplicar las formas de onda siguientes a las entradas 'a' y 'b' de una puerta AND y representar la forma de onda de su salida 'x'.



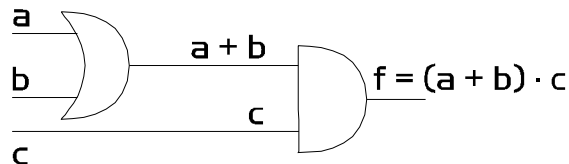
Problema: determinar la forma de onda de una puerta OR de tres entradas y dibujar la salida.



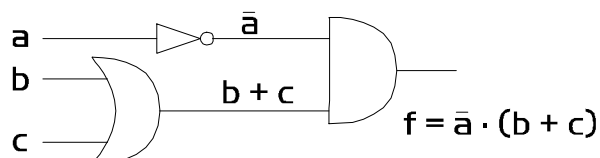
Representación De Funciones Booleanas Con Compuertas Logicas

Son varias variables booleanas relacionadas mediante las operaciones lógicas.

$$f = (a + b) \cdot c$$

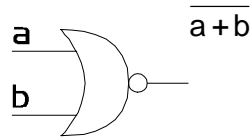


$$f = \bar{a} \cdot (b + c)$$

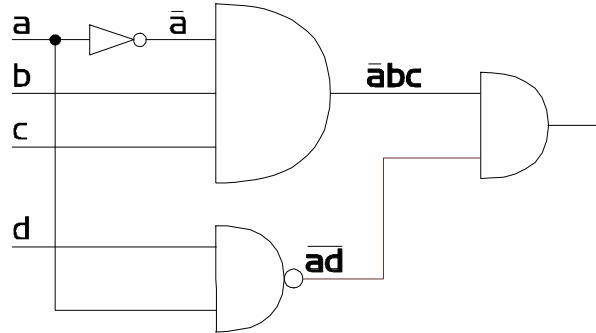




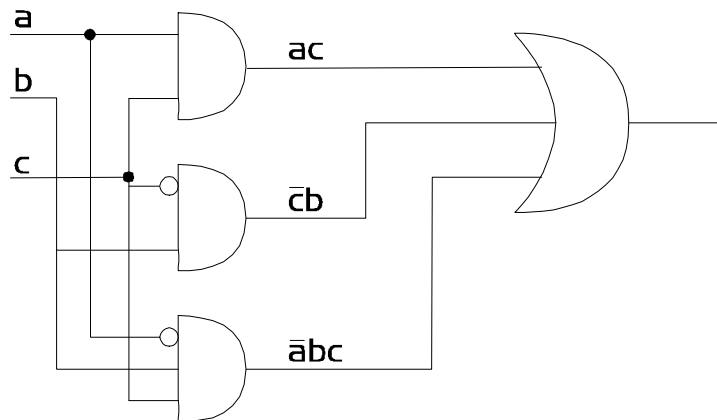
$$f = \overline{a+b}$$



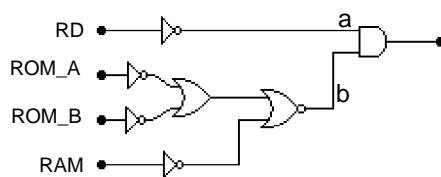
$$f = (\overline{a}bc) \cdot (\overline{a}d)$$



$$f = ac + b\overline{c} + \overline{a}bc$$



El circuito lógico de la figura genera una salida **mem** que se utiliza para activar los circuitos integrados de memoria de un ordenador. determinar las condiciones de entrada necesarias para activar **mem**.

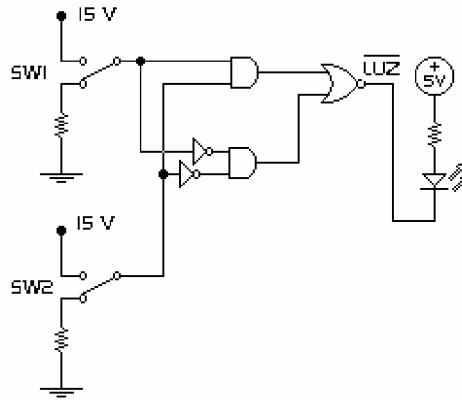


Para conseguir que a y b sea 1,
 RD=0
 RAM=1
 ROM_A=1
 ROM_B=1

La figura muestra una aplicación que simula un circuito con doble interruptor para encender o apagar una luz. En este caso, la luz proviene de un led, que conduce cuando la salida de la puerta NOR está en baja. Nótese que esta salida está marcada como $\overline{\text{LUZ}}$ para indicar que es activa en baja. Determinar las condiciones de entrada necesarias para encender el led.



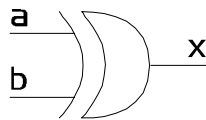
Sw1	Sw2	Led
0	0	1
0	1	0
1	0	0
1	1	1



Otras funciones lógicas:

OR Exclusiva (XOR):

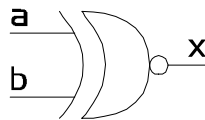
a	b	$a \oplus b$
0	0	0
0	1	1
1	0	1
1	1	0



$$f = a\bar{b} + \bar{a}b$$

NOR Exclusiva (XNOR):

a	b	$\overline{a \oplus b}$
0	0	1
0	1	0
1	0	0
1	1	1



$$f = \bar{a}\bar{b} + ab$$

Cómo implementar un circuito con puertas NAND y NOR

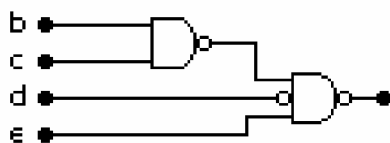
Se utilizan las leyes de Morgan y Doble Negación.

$$f_1 = bc + d + \bar{e}$$

NAND

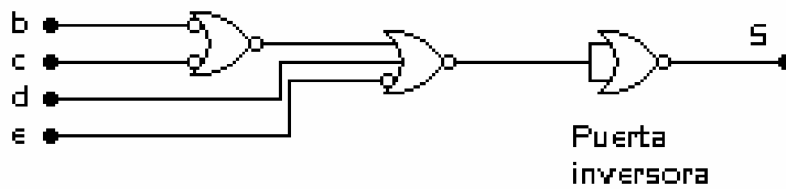
Pasar sumas a productos

$$f_1 = \overline{\overline{bc + d + \bar{e}}} = \overline{\overline{bc} \cdot \overline{d} \cdot \overline{\bar{e}}} = \overline{\overline{bc} \cdot \overline{d} \cdot e}$$

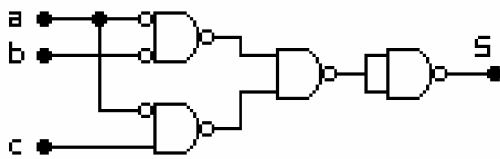


NOR

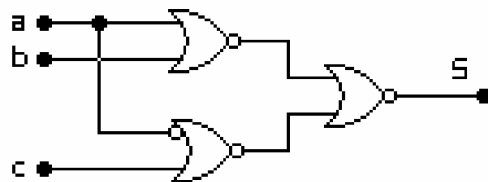
Pasar productos a sumas $f_1 = \overline{\overline{bc+d}} - \overline{\overline{e}} = \overline{\overline{b+c+d}} + \overline{\overline{e}} = \overline{\overline{\overline{\overline{b+c+d+e}}}}$

NAND

$$f_2 = \overline{\overline{(a+b)}} \cdot \overline{\overline{(c+a)}} = (\overline{\overline{a}} \cdot \overline{\overline{b}}) \cdot (\overline{\overline{c}} \cdot \overline{\overline{a}}) = (\overline{\overline{a \cdot b}}) \cdot (\overline{\overline{c \cdot a}})$$

NOR

$$f_2 = \overline{\overline{(a+b)}} \cdot \overline{\overline{(c+a)}} = \overline{\overline{(a+b)}} + \overline{\overline{(c+a)}}$$





Ejercicios Resueltos

Ejercicio 1

Escriba la tabla de verdad de las funciones OR, NOR, AND, NAND, INVERSOR, OR-EXCLUSIVA

Solución

a	INV
0	1
1	0

a	b	OR	NOR	AND	NAND	OR-EX
0	0	0	1	0	1	0
0	1	1	0	0	1	1
1	0	1	0	0	1	1
1	1	1	0	1	0	0

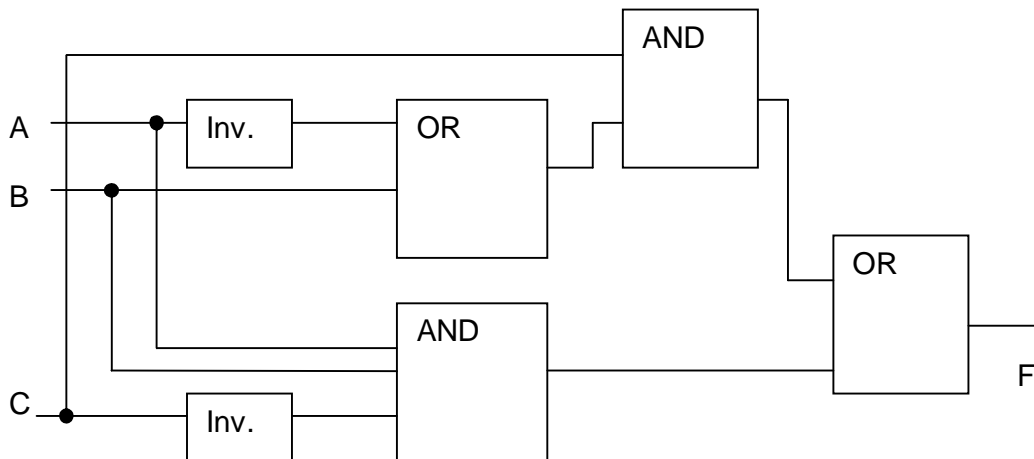
Ejercicio 2

Verifique en forma algebraica el cumplimiento de la propiedad distributiva para $a * (b + c)$

Solución:

Ejercicio 3

Escriba la función de Boole de la salida $F_{(a,b,c)}$ del circuito lógico de la figura. Evalúe si la función obtenida puede ser reducida, en caso afirmativa implemente un nuevo circuito con compuertas NOR y/o NAND



Solución:

$$a. F_{(a,b,s)} = (a' + b) \cdot c + (a \cdot b \cdot c')$$

$$b. F_{(a,b,s)} = (a' \cdot c) + (a \cdot b)$$

$$= [(a' \cdot c) + (a \cdot b)]'' \quad \text{negamos dos veces}$$

$$= [(a' \cdot c)' * (a \cdot b)']' \quad \text{aplicando Morgan a una de las negaciones}$$

**Ejercicio 4**

Simplifique la función F tanto como sea posible a través del método algebraico, compruebe con el método de Karnaugh

a. $F_{(x,y,z)} = xz'y + (xz'y + zx') [y(z+x) + y'z + y'xz']$

b. $F_{(x,y,z)} = (x + y'x')[xz + xz'(y + y')]$

Solución:

Ejercicio 5

Demuestre que

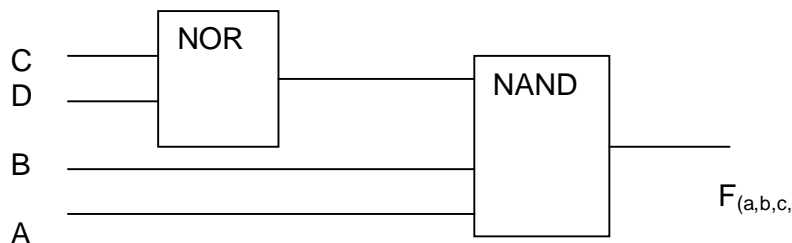
a. $(a + b)(a' + c)(b + c) = (a + b)(a' + c)$

b. $(ab + c + d)(c' + d)(c' + d + e) = abc' + d$

Solución:

Ejercicio 6

Obtenga la expresión Booleana de la figura y determine el valor de la salida si $a = b = c = d = 1$



Solución:

a. $F_{(a,b,s,d)} = [(c + d)' \cdot a \cdot b]'$

b. Si $a = b = c = d = 1 \rightarrow F_{(a,b,s,d)} = 1$

Ejercicio 7

¿Cuál es el único conjunto de condiciones de la entrada que producirá una salida ALTA (1) en una compuerta NOR de tres entradas

Solución:

$a = b = c = 0$

**Ejercicio 8**

En la figura del ejercicio 6 cambie la compuerta NOR por una NAND y la compuerta NAND por una NOR, cual es la nueva expresión de la salida. Que obtiene si $a = b = c = d = 0$

Solución:

- $F_{(a,b,s,d)} = (c + d) \cdot a \cdot b$
- Si $a = b = c = d = 0 \Rightarrow F_{(a,b,s,d)} = 0$

Ejercicio 9

Realice

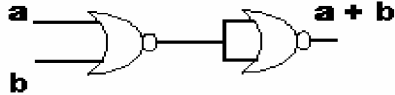
- un inversor a partir de funciones NOR
- una OR a partir de funciones NOR
- una AND a partir de funciones NOR
- una AND a partir de funciones NAND

Solución:

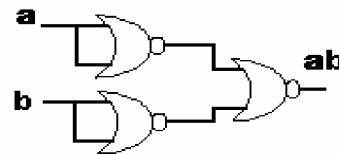
a)



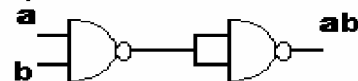
b)



c)



d)

**Ejercicio 10**

Escriba la función en forma de suma de productos para un circuito con cuatro variables de entrada y una salida que sea ALTA (1) solo cuando la entrada 'a' sea BAJA (0) al mismo tiempo en que otras dos entradas también sean BAJAS (0)

Solución:

$$F_{(a,b,c,d)} = a' b' c' d + a' b' c d' + a' b c' d'$$

Ejercicio 11

Realice la función del ejercicio anterior utilizando solo compuertas NAND. ¿Cuántas necesita?

Solución:

Ocho (8)

**Ejercicio 12**

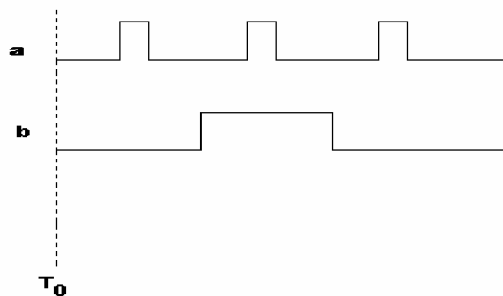
Realice la reducción combinada de las funciones F_1 , F_2 , F_3 de la siguiente tabla de verdad

a	b	c	F_1	F_2	F_3
0	0	0	0	1	1
0	0	1	1	1	0
0	1	0	0	1	0
0	1	1	1	0	1
1	0	0	0	0	1
1	0	1	1	1	1
1	1	0	0	1	0
1	1	1	1	1	1

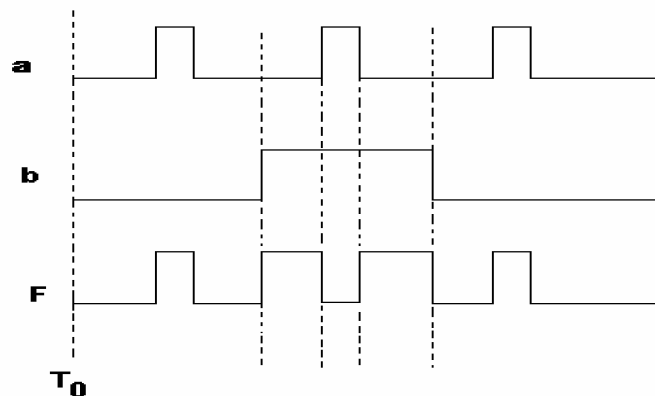
Solución:

Ejercicio 13

En el circuito de la figura determinar la forma de onda de la salida



Solución:



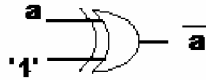
**Ejercicio 14**

Un diseñador necesita un inversor y solo dispone de un integrado 7486 ¿Puede realizar su diseño?

Solución:



Si, conectando una de las entradas siempre en '1'

**Ejercicio 15**

Determine la expresión mínima de $F_{(a,b,c,d)}$ para los siguientes casos

a.

	00	01	11	10
00	1	1	1	1
01	1	1	0	0
11	0	0	0	1
10	0	1	1	0

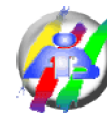
b.

	00	01	11	10
00	1	1	X	1
01	X	X	0	0
11	x	X	0	1
10	0	1	1	0

Solución:

a. $F_{(a,b,c,d)} = c' d' + c' a' + d' b + a b' c d$

b. $F_{(a,b,c,d)} = c' d' + c' a' + d' b + a b' c d$



Bibliografía

- q **Teoría y Practica de los sistemas digitales**
TOCCI RONALD
Tercera Edición
- q **Sistemas electrónicos Digitales**
Rafael Sánchez
Alfaomega 1993
- q **Teoría de Conmutación y diseño lógico**
Hill – Peterson
- q **Digital Design UIT Standart MSI and LSI**
Thomas R. BLAKESLEE
Second Edition
- q **Ingenieria Computacional – Diseño de harware**
Morris Mano
Prentice Hall - 1991
- q **Principios de arquitectura de computadoras**
Miles Murdocca y Vincent Heuring
Prentice Hall - 2000
- q **Sistemas Digitales**
Ruiz, Espinoza, Roure
McGraw Hill
- q **Estructura de computadores y Periféricos**
Martinez Dura, Grau, Solano
Alfaomera