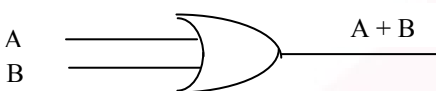





Compuertas lógicas

Un bloque lógico es una representación simbólica gráfica de una o más variables de entrada a un operador lógico, para obtener una señal determinada o resultado. Los símbolos varían de acuerdo con la rama donde se utilizan, o bien del fabricante. Cada bloque lógico representa un dispositivo que permite manipular la señal según el campo de acción: en mecánica se les llama válvulas (paso del aire o aceite); en electricidad apagadores, contactos (paso de corriente eléctrica); y en electrónica puertas o compuertas (paso de pulsos eléctricos). En este libro sólo se abordarán los símbolos usados en electrónica para la representación de las compuertas, ya que son los que interesan al área de la computación, sin embargo el tratamiento teórico por medio del álgebra booleana es válido para todos ellos independientemente del área.

Tabla 5.2 Compuertas básicas

Compuerta	Símbolo
O (Or)	
Y (And)	
No (Not)	
Or-exclusivo (Xor)	

Las compuertas pueden recibir una o más señales de entrada. En la tabla 5.2, A y B son señales que entran a la compuerta y pueden tener un valor de 1 o 0 dependiendo de si existe o no la señal, la cual procede de un sensor o bien de la salida de una compuerta anterior. Esos valores de entrada generan una sola salida, que a su vez también es 0 o 1

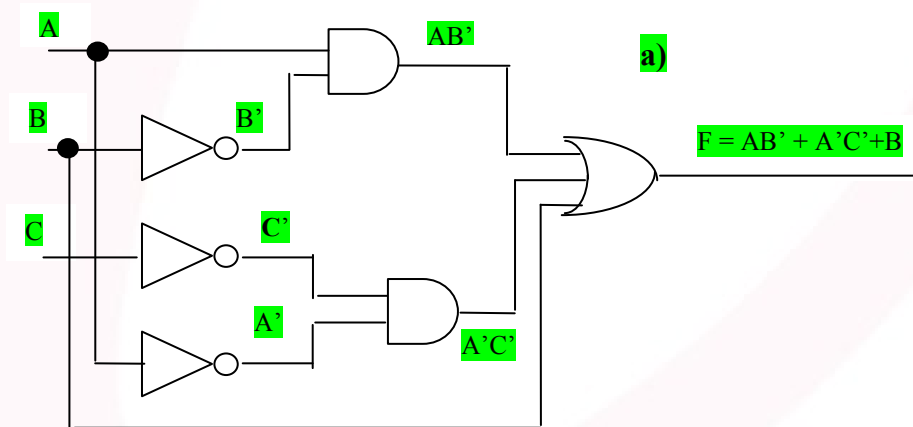
dependiendo de la compuerta de que se trate y de los valores de las señales de entrada.

Para representar expresiones booleanas mediante compuertas lógicas es conveniente tener en cuenta las tablas de verdad de las compuertas básicas (operadores lógicos) Or, And y Not vistas en el capítulo de lógica matemática.

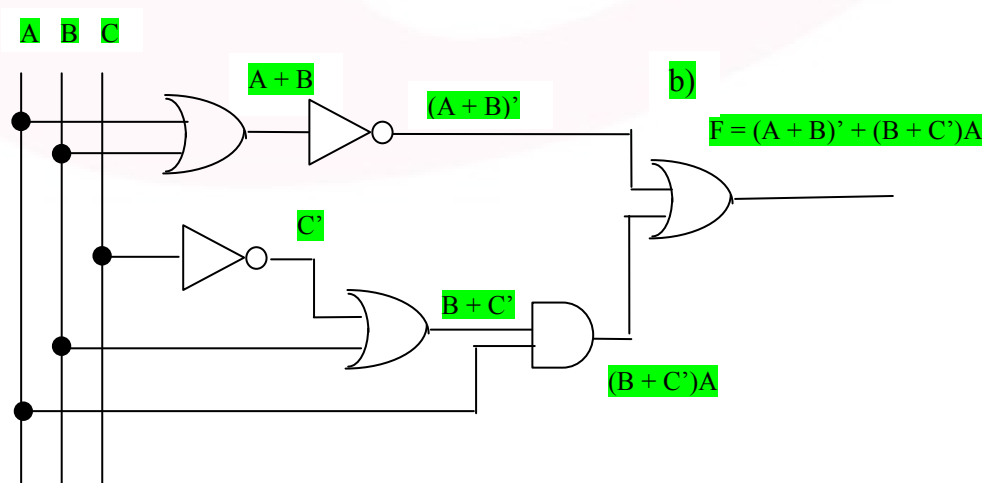
Ejemplo 5.11. Representar las siguientes expresiones booleanas usando compuertas lógicas básicas:

- a) $F = AB' + A'C' + B$
- b) $F = (A + B)' + (B + C')A$

La representación de (a) es:

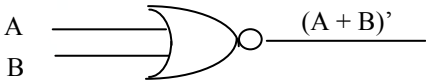
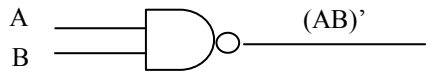
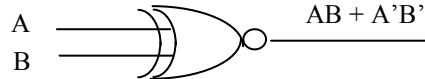


La representación de (b) es:



También existen compuertas lógicas compuestas como Nand y Nor, que son una combinación de los operadores Not y And para la primera y Not y Or para la segunda. En la tabla 5.3 se muestran los símbolos correspondientes.

Tabla 5.3 Compuertas

Compuert	Símbolo
Nor	 $(A + B)'$
Nand	 $(AB)'$
Xnor	 $AB + A'B'$

Generalmente los circuitos digitales se construyen con compuertas Nand y Nor, ya que son más fáciles de encontrar en el mercado, son más comunes desde el punto de vista del hardware y están disponibles en la forma de circuitos integrados. Debido a la preferencia de uso de estas compuertas en el diseño de los circuitos, es importante reconocer la relación que existe entre los circuitos construidos con compuertas And, Or y Not y su diagrama equivalente Nand o Nor.

Cuando se simplifica una función el resultado se puede presentar en "sumas de productos" o en "productos de sumas", y en forma natural la presentación en suma de productos permite una implementación usando compuertas Nand mientras que el producto de sumas se puede representar más fácilmente con compuertas Nor, sin embargo es posible implementar cualquier expresión booleana sólo con compuertas Nand o sólo con compuertas Nor.

Ejemplo 5.12. ¿Cuál es el circuito de la expresión booleana

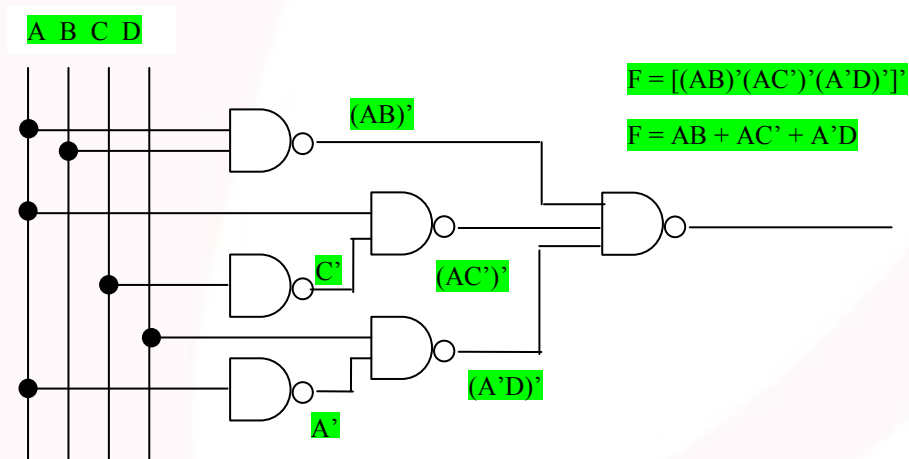
$$F = A(B + C') + A'D$$

hecho sólo con compuertas Nand?

Para obtener el circuito pedido es recomendable llevar la expresión dada a suma de productos:

$$F = AB + AC' + A'D$$

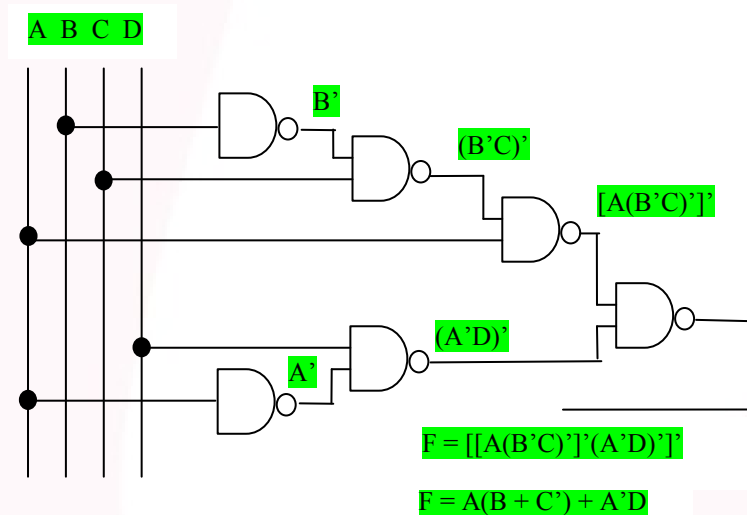
Por lo tanto, el circuito es el siguiente:



Hay que observar que al final se aplicó la ley de De Morgan para quitar la complementación del corchete y obtener el resultado. También se debe destacar que cuando entran dos o más señales a una compuerta Nand primero las multiplica y después complementa dicha multiplicación, pero cuando entra una señal sólo la complementa.

Por otro lado, si no se hubieran hecho las operaciones necesarias para quitar el paréntesis y tener la expresión en sumas de productos, también se podría representar únicamente con compuertas Nand aunque esto algunas veces es un poco más complicado:

$$F = A(B + C') + A'D$$



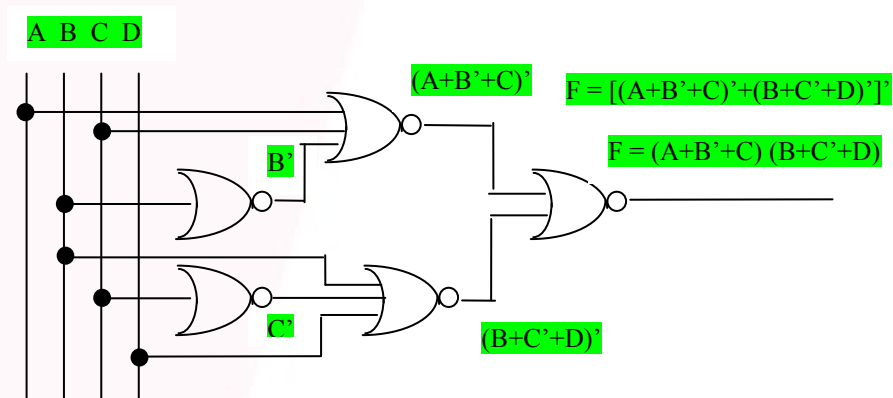
De la misma manera, el bloque lógico Nor facilita su uso cuando la expresión se encuentra dada en productos de sumas.

Ejemplo 5.13. Representar la expresión booleana

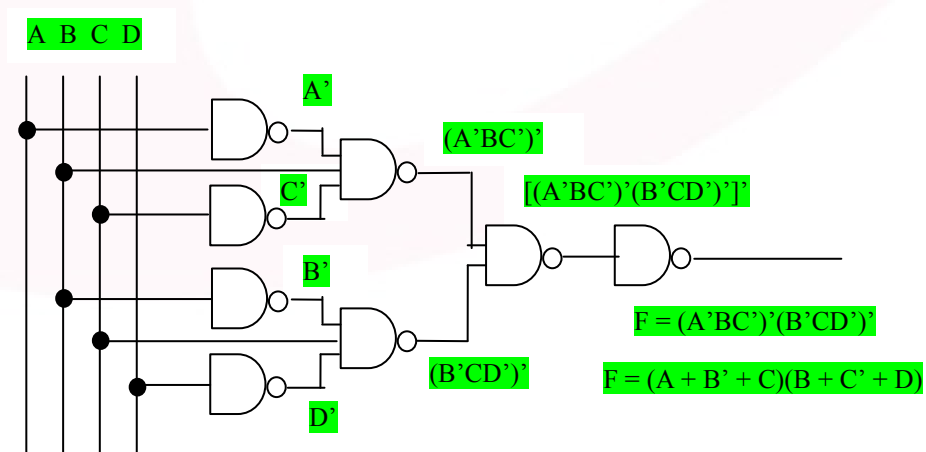
$$F = (A + B' + C)(B + C' + D)$$

usando sólo compuertas Nor.

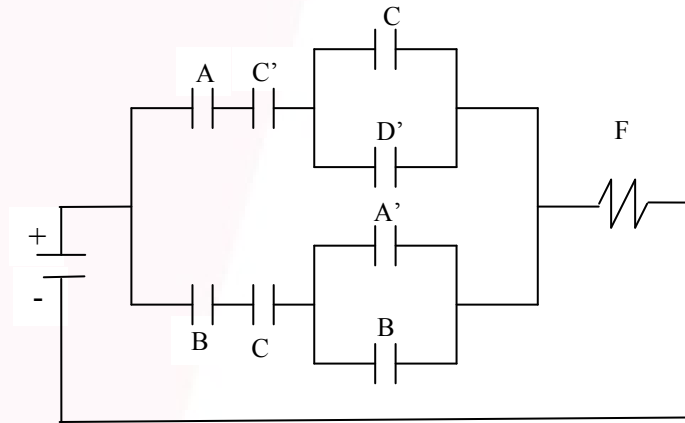
En este caso se tiene el siguiente esquema



La misma expresión booleana representada con compuertas Nand quedaría de la siguiente manera:



Ejemplo 5.14. Considérese el siguiente circuito:



- a) ¿Cuál es la expresión booleana sin simplificar que representa dicho circuito?
- b) Simplificar la expresión booleana usando teoremas del álgebra booleana.
- c) Por medio del mapa de Karnaugh simplificar la expresión del inciso (a) y expresar el resultado en sumas de productos.
- d) ¿Cuál es la expresión simplificada en productos de sumas?
- e) Comprobar, por medio de una tabla de verdad, que la expresión booleana obtenida en el inciso (c) es lógicamente equivalente a la obtenida en el inciso (d).
- f) Representar el resultado del inciso (c) en un circuito lógico, usando para ello compuertas básicas.
- g) ¿Cuál es el circuito del inciso (c) basado en compuertas Nand exclusivamente?
- h) ¿Cuál es el circuito lógico del inciso (c) basado en compuertas Nor exclusivamente?

La solución de cada inciso es la siguiente:

- a) La expresión booleana es

$$F = AC' (C + D') + BC(A' + B)$$

b) Simplificando mediante teoremas resulta que

$$F = AC'C + AC'D' + A'BC + BBC$$

$$F = 0 + AC'D' + A'BC + BC$$

$$F = AC'D' + BC(A' + 1)$$

$$F = AC'D' + BC$$

c) Se sabe que $AC'C = 0$ y $BBC = BC$, y sustituyendo esto en la expresión $F = AC'C + AC'D' + A'BC + BBC$ resulta que la expresión booleana a representar en el mapa es $F = AC'D' + A'BC + BC$. Aplicando la condición de que para representar un minitérmino en el mapa de Karnaugh éste debe contener todas las letras, a continuación se agregan las variables faltantes con sus posibles combinaciones:

$$AC'D' = AB'C'D' + ABC'D'$$

$$A'BC = A'BCD' + A'BCD$$

$$BC = A'BCD' + A'BCD + ABCD' + ABCD$$

A partir de la información se obtiene el siguiente mapa:

		CD			
		00	01	11	10
AB	00				
	01			1	1
	11	1		1	1
	10	1			

Del mapa se obtiene que la expresión booleana en sumas de productos es

$$F = AC'D' + BC$$

lo cual concuerda con el resultado obtenido usando teoremas.

d) Para obtener el producto de sumas se colocan ceros en las casillas vacías y se agrupa la información:

		CD			
AB		00	01	11	10
00	0	0		0	0
01	0	0			
11			0		
10			0	0	0

A partir del mapa se puede leer que:

$$F = A'C' + B'C + C'D$$

Complementado y aplicando leyes de De Morgan resulta que

$$(F)' = (A'C' + B'C + C'D)'$$

$$F = (A + C)(B + C')(C + D')$$

e) Las expresiones booleanas obtenidas en los incisos (c) y (d) son

$$F = AC'D' + BC$$

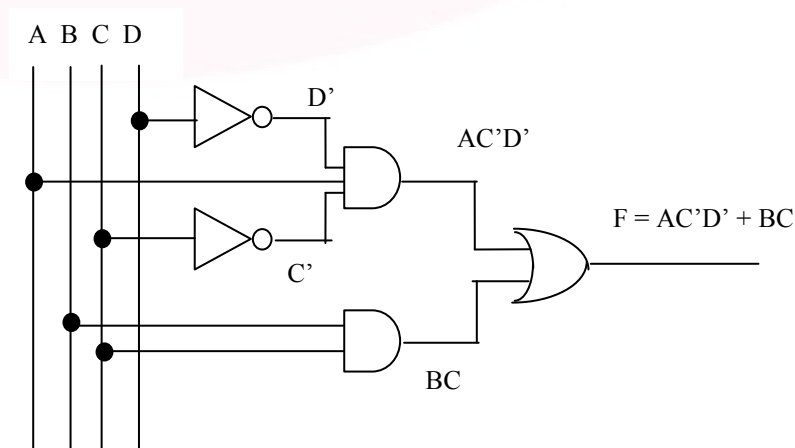
$$F = (A + C)(B + C')(C + D')$$

y a partir de éstas se tiene la siguiente tabla

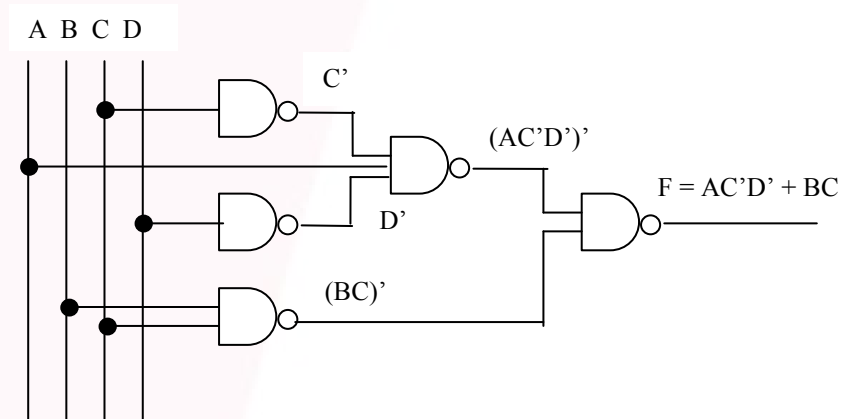
A	B	C	D	A'	B'	C'	D'	AC'	AC'D'	BC	AC'D'+BC	A+C	B+C'	C+D'	F
0	0	0	0	1	1	1	1	0	0	0	0	0	1	1	0
0	0	0	1	1	1	1	0	0	0	0	0	0	1	0	0
0	0	1	0	1	1	0	1	0	0	0	0	1	0	1	0
0	0	1	1	1	1	0	0	0	0	0	0	1	0	1	0
0	1	0	0	1	0	1	1	0	0	0	0	0	1	1	0
0	1	0	1	1	0	1	0	0	0	0	0	0	1	0	0
0	1	1	0	1	0	0	1	0	0	1	1	1	1	1	1
0	1	1	1	1	0	0	0	0	0	1	1	1	1	1	1
1	0	0	0	0	1	1	1	1	1	0	1	1	1	1	1
1	0	0	1	0	1	1	0	1	0	0	0	1	1	0	0
1	0	1	0	0	1	0	1	0	0	0	0	1	0	1	0
1	0	1	1	0	1	0	0	0	0	0	0	1	0	1	0
1	1	0	0	0	0	1	1	1	1	0	1	1	1	1	1
1	1	0	1	0	0	1	0	1	0	0	0	1	1	0	0
1	1	1	0	0	0	0	1	0	0	1	1	1	1	1	1
1	1	1	1	0	0	0	0	0	0	1	1	1	1	1	1

Aquí se observa que las columnas sombreadas concuerdan en todas sus líneas, por lo tanto esto demuestra que $F = AC'D' + BC$ es lógicamente equivalente a $F = (A + C)(B + C')(C + D')$.

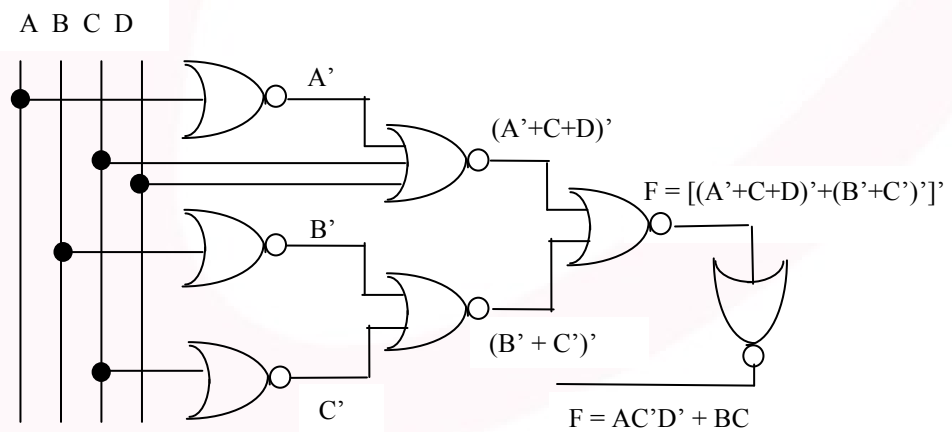
f) La expresión obtenida en el inciso (c) es $F = AC'D' + BC$, y su representación con compuertas básicas es



g) La representación de la expresión booleana $F = AC'D' + BC$, sólo con compuertas Nand es la siguiente:



h) La representación de la expresión booleana $F = AC'D' + BC$, sólo con compuertas Nor es la siguiente:



Problemas

5.7. Representar con compuertas básicas (And, Or y Not), con compuertas Nand (exclusivamente) y con compuertas Nor (exclusivamente), la expresión lógica:

$$F = AB'C + A'B'D' + AD$$

5.8. Representar con compuertas básicas (And, Or y Not), con compuertas Nand (exclusivamente) y con compuertas Nor (exclusivamente), la expresión lógica:

$$F = (B' + C + D')(A + C' + D)B'$$

5.9. Obtener las compuertas Not, And, Or, Nor, X-or y X-nor con base en compuertas Nand exclusivamente.

5.10. Obtener las compuertas Not, And, Or, Nand, X-or y X-nor con base en compuertas Nor exclusivamente.