CIRCUITOS LÓGICOS

Antes de tratar la noción de circuito lógico, es pertinente referirse a los procesos de transformación de expresiones booleanas. De acuerdo con una necesidad o un propósito específicos, se podrá requerir la transformación de expresiones algebraicas en otras más simples o, tal vez en algunos casos, en unas más complejas.

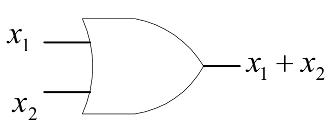
Para el caso de la **simplificación** de expresiones booleanas, pueden emplearse varios métodos y/o técnicas:

- El uso de las leyes de las expresiones booleanas que transforman la expresión original.
- El uso de las leyes de las expresiones booleanas que transforman la expresión *dual* de la expresión original; luego de lo cual debe efectuar, de nuevo, el proceso de dualización.
- Emplear un método gráfico conocido como Mapas de Karnaugh; método útil cuando el problema no involucra demasiadas variables.

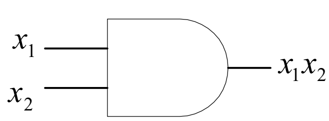
Una expresión booleana se podría "representar" mediante un *circuito lógico*; pero, ¿qué es un circuito lógico? Es un esquema que enseña una expresión booleana a través de una interconexión apropiada de elementos gráficos conocidos como *compuertas lógicas*. Las compuertas lógicas son de varios tipos, y cada una de ellas es un símbolo gráfico que denota a una *operación booleana* en particular, junto con sus operandos y resultado.

Las operaciones booleanas, vistas hasta ahora, son: m.c.s. (+), M.C.I. (·) y complemento (′). Estas operaciones en el contexto de los circuitos de compuertas lógicas adquieren los siguientes nombres: '*o*', '*y*' e '*inversor*', respectivamente[1].

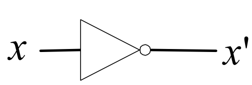
Las compuertas lógicas básicas son:



Compuerta *or*



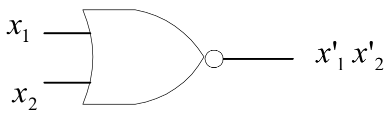
Compuerta *and*



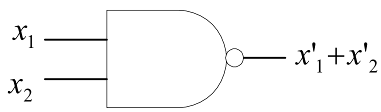
Compuerta *not*

[1] Es muy frecuente emplear los nombres en inglés: *or*, *and* y *not*.

Con base en estas compuertas lógicas podrían configurarse otras, que corresponderían a operaciones booleanas distintas de las estudiadas, observe:



Compuerta *nor*

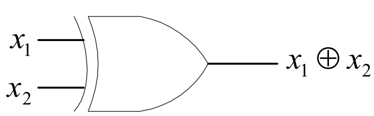


Compuerta *nand*

Otra compuerta bastante común en el ámbito de los circuitos lógicos es la correspondiente a la siguiente expresión booleana: $x'_1x_2 + x_1x'_2$. Esta expresión booleana se la denomina *"o exclusivo"* (*xor*) y el símbolo para su operador booleano es $\oplus$; es decir, $x_1 \oplus x_2 = x'_1x_2 + x_1x'_2$. Esta compuerta puede identificarse mediante una FND resultante de observar la siguiente tabla:

x_1	x_2	$x_1 \oplus x_2$
0	0	0
0	1	1
1	0	1
1	1	0

La compuerta correspondiente a la operación booleana *xor* es:



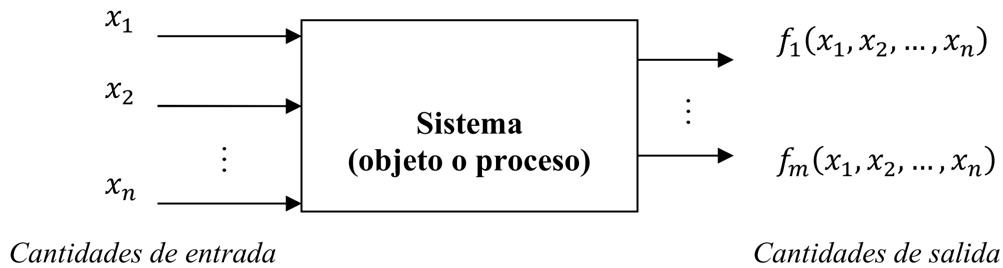
Diseño de circuitos lógicos

El diseño de un circuito lógico es un proceso del que se obtiene una configuración de compuertas lógicas para simbolizar un *sistema* [2]. El diseño comienza con la obtención de una(s) expresión(es) booleana(s) para un sistema (formulado en términos de un problema) para el que sea adecuado ese tipo de estructura matemática[3]; posteriormente, se efectúa la representación de esa(s) expresión(es) a través de un circuito lógico, es decir, integrando compuertas lógicas.

El diseño, como se mencionó, involucra un proceso inicial de identificación de los aspectos, características o atributos fundamentales observables del problema [4]. Varios de estos aspectos observables representan las influencias externas al sistema, las que serán denominadas *cantidades de entrada*; una, o varias más, serán las respuestas del sistema, conocidas como *cantidades de salida*. El

[2] Un sistema puede entenderse como una *abstracción de un tipo de objeto o de proceso*
[3] Una expresión booleana
[4] Se obtiene un sistema muy simple en el entendido que todavía no se conoce cómo se relacionan esos atributos

objetivo es encontrar una expresión booleana que represente el valor de cada cantidad de salida, en términos de todas, o algunas de, las cantidades de entrada.[5]



En un gran número de problemas se posee información adicional del sistema en términos de un conjunto de valores para las cantidades de entrada, y los respectivos de las cantidades de salida. Un avance sustancial en el conocimiento del sistema, sería descubrir una expresión matemática[6] que revele cómo se relacionan las cantidades del sistema, con base en la información recogida de los valores manifestados por ellas. Para este propósito, puede utilizarse el siguiente proceso general de diseño con el fin de obtener un circuito lógico:

- Construya una tabla, referida como *de verificación*, de manera que en ella aparezcan todas las combinaciones posibles de los valores de las cantidades booleanas de entrada y el respectivo resultado, o valor, de cada cantidad booleana de salida.

- Obtenga una expresión booleana, en términos de las variables de entrada, para cada variable de salida empleando alguna de las dos *formas normales*, disyuntiva o conjuntiva.

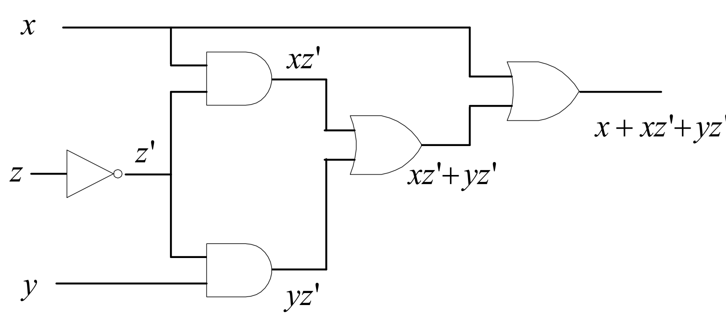
- Aplique algún método para simplificar, si es posible, la expresión producida en el paso anterior.

- Dibuje la expresión simplificada mediante un circuito.

EJEMPLOS.

1) Mediante compuertas *and*, *or* e *inversor* constrúyase una red de compuertas para:

$$f = xz' + yz' + x$$



Simplifique y vuelva a dibujar.

[5] En adelante se usará el término *variable* en lugar de cantidad.

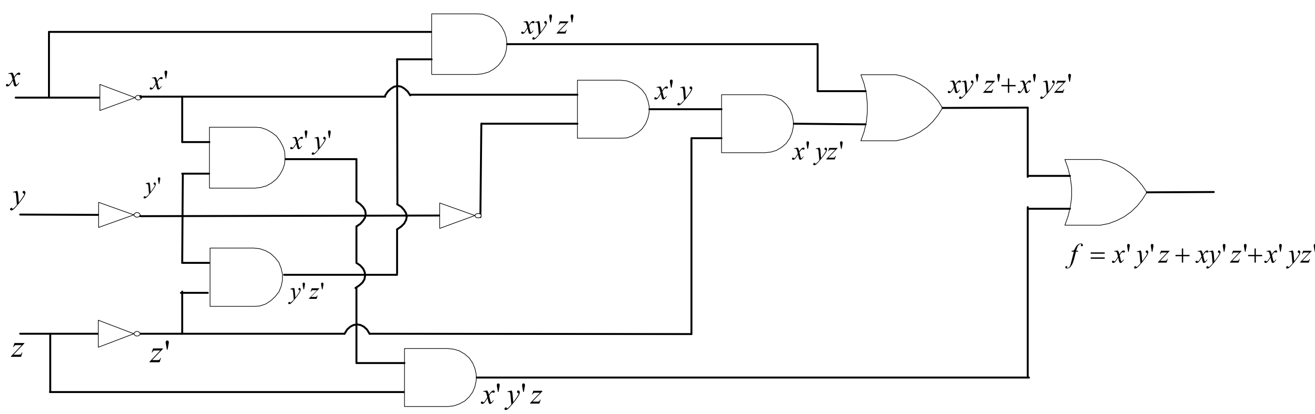
[6] En nuestro caso, una expresión booleana

2) Dibuje una red lógica que tenga salida 1 si: "exactamente una, y sólo una, de las entradas x, y, z tiene valor 1".

x	y	z	f
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

R/. Dado que en la columna correspondiente a la salida (o función) aparecen cantidad menor de unos que de ceros, es preferible representarla a través de la FND, por tanto:

$$f = x'y'z + x'yz' + xy'z'$$



Simplifique y vuelva a dibujar.

$$f = x'y'z + x'yz' + xy'z'$$

3) Supóngase que en cada uno de los 3 accesos a una sala hay un interruptor para el accionamiento del alumbrado central. Los 3 interruptores funcionan de una manera alternativa, es decir, cada uno de ellos puede apagar el alumbrado encendido, y viceversa. Construya la función booleana correspondiente a esta situación y el circuito correspondiente:

El primer paso es construir la tabla de verificación para este sistema, donde las variables de entrada son los estados de encendido-apagado de cada uno de los 3 interruptores, x, y y z, respectivamente. La variable de salida será el estado de encendido-apagado del alumbrado, $f(x, y, z)$, que evidentemente depende de los valores particulares de las variables de entrada.
La solución requiere establecer una configuración inicial de valores entrada-salida y de allí se desprenderán los restantes; por ejemplo, suponga que la configuración $x = 0$, $y = 0$ y $z = 0$, determinan $f(x, y, z) = 0$; de aquí en adelante se puede elegir cualquier secuencia de elección de interruptores y cada uno de los componentes de la secuencia lo que hará es cambiar el estado existente en el anterior. Observe una posibilidad:

x	y	z	$f(x,y,z)$
0	0	0	0
0	0	1	1
0	1	1	0
0	1	0	1
1	1	0	0
1	0	0	1
1	0	1	0
1	1	1	1

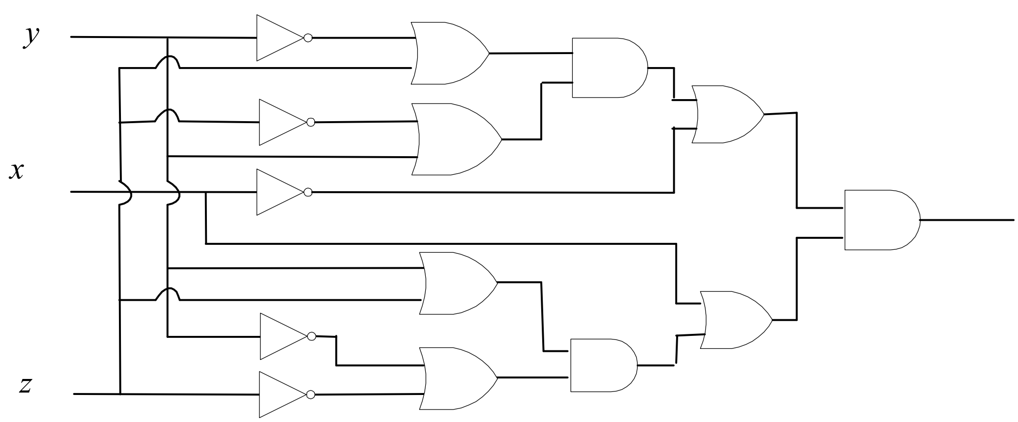
Ahora, debe construirse la FND o la FNC correspondiente a la función $f(x,y,z)$. Puesto que la cantidad de ceros y unos es igual en la columna $f(x,y,z)$, entonces cualquiera de las dos expresiones es igualmente conveniente. Se hará, con la FNC:

$$f(x,y,z) = (x+y+z)(x+y'+z')(x'+y'+z)(x'+y+z')$$

$$f(x,y,z) = x'y'z + x'yz' + xy'z' + xyz = x \oplus y \oplus z$$

Una expresión equivalente es:

$$f(x,y,z) = \big(x + (y+z)(y'+z')\big)\big(x' + (y'+z)(y+z')\big)$$



Compruebe que el circuito presentado corresponde a la última expresión, colocando las sub-expresiones obtenidas luego de cada compuerta.[7]

Continúe simplificando hasta cuando en la expresión se requieran máximo 5 operaciones distintas, y vuelva a dibujar.

4) Un sistema está conformado por dos variables de entrada y una de salida (todas de tipo binario). El comportamiento del sistema es tal que la salida asume el valor de **uno** (1) únicamente cuando ambas entradas tienen el mismo valor.

[7] ¿Se puede obtener una expresión booleana aún más simplificada que la presentada? Si la obtiene, grafíquela.

a) Escriba la tabla de verificación.

x	y	f
0	0	1
0	1	0
1	0	0
1	1	1

b) Obtenga la expresión booleana en FNC que represente el comportamiento de la salida del sistema.

$$f = (x + y')(x' + y)$$

c) Empleando **únicamente** compuertas NOR, o **únicamente** compuertas NAND, dibuje el circuito lógico correspondiente a la expresión que obtuvo en b).

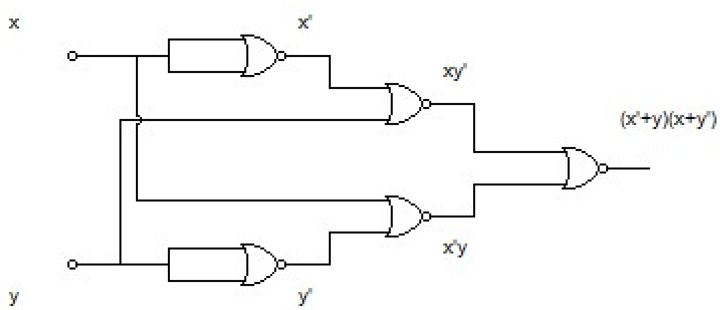


Figura Circuito empleando sólo compuerta *nor* para representar la FNC

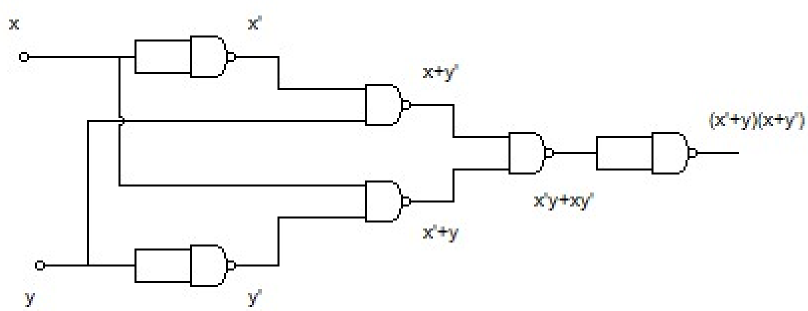


Figura Circuito empleando sólo compuerta *nand* para representar la FNC