
Capítulo 13: ALGEBRA BOOLEANA

13.1 Introducción



George Boole

1815-1864

George Boole fue un lógico y matemático británico. Escribió los libros: "The Mathematical Analysis of Logic" (1847) y "An Investigation of the Laws of Thought" (1854).

Desarrolló la lógica Simbólica mediante la cual las proposiciones pueden ser representadas mediante símbolos y la teoría que permite trabajar con estos símbolos, sus entradas (variables o proposiciones) y sus salidas (respuestas). Dicha lógica cuenta con operaciones lógicas que siguen el comportamiento de reglas algebraicas. Consideró que las proposiciones lógicas podían ser tratadas mediante herramientas matemáticas. Las proposiciones lógicas (asertos, frases o predicados de la lógica clásica) son aquellas que únicamente pueden tomar valores Verdadero/Falso, o preguntas cuyas únicas respuestas posibles sean Sí/No.

Según Boole, al conjunto de reglas de la Lógica Simbólica se le denomina Álgebra Booleana. Todas las variables y constantes del Álgebra Booleana, admiten sólo uno de dos valores en sus entradas y salidas: Sí/No, 0/1 o Verdadero/Falso. Estos valores bivalentes y opuestos pueden ser representados por números binarios de un dígito denominado bit, por lo cual el Álgebra Booleana se puede entender cómo el Álgebra del Sistema Binario.

Todas las operaciones pueden representarse mediante elementos físicos de diferentes tipos: mecánicos, eléctricos, neumáticos o electrónicos que admiten entradas binarias o lógicas y que devuelven una respuesta (salida) también binaria o lógica. Sus estados pueden ser: Abierto/Cerrado, en el caso de interruptores, Encendida/Apagada si se refiere a una bombilla, Cargado/Descargado, si se tratase de un condensador, Nivel Lógico 0/Nivel lógico 1, para producir una salida lógica de un circuito semiconductor, entre otras.

Un día en 1864 George Boole recorrió dos millas de su residencia a la universidad, bajo una lluvia torrencial para dar una conferencia que llevó a cabo con sus ropas mojadas. Como resultado, adquirió un fuerte resfriado que afectó sus pulmones y así terminó su carrera a la edad de 49 años. Parece ser que negligentemente su esposa Mary (nieta de Sir George Everest), creía que su remedio podría ser la causa. En efecto, ella puso a Boole en su cama y le arrojó cubos de agua, lo cual aceleró más su enfermedad.

El trabajo de Boole ha llegado a ser como un paso fundamental en la revolución de los computadores hoy en día. El álgebra Booleana tiene una amplia aplicación en el switch telefónico y en el diseño de computadores modernos.

A mediados del siglo XX el Álgebra Booleana se utilizó en el manejo de información digital llamada Lógica Digital. En efecto, Shannon (1930) la pudo formular en su teoría de la codificación y John Von Neumann la pudo enunciar en el modelo de arquitectura que define la estructura interna de los ordenadores desde la primera generación.



Claude Elwood Shannon
(1916 - 2001)

Claude Elwood Shannon (1916 - 2001) ingeniero eléctrico y matemático. Nacido el 30 de abril de 1916, Míchigan-USA. Reconocido como "el padre de la teoría de la información". Falleció el 24 de febrero del año 2001, a la edad de 84 años, después de una larga lucha en contra la enfermedad de Alzheimer. Aplico el álgebra booleana a los circuitos con relés . "A Symbolic Analysis of Relay and Switchin Circuits" Trans. AIEE 1938.

El desarrollo de este capítulo tiene como finalidad iniciar a sus lectores en la comprensión de las funciones lógicas de un circuito digital como una aplicación tecnológica de la lógica de proposiciones. No se desarrollarán sistemas complejos, pero si se realizarán las operaciones básicas de un sistema digital combinatorio.

Se estudiarán aspectos básicos de un circuito digital tales como:

- La descripción simbólica de la lógica digital que mantiene sus bases en el álgebra booleana y otras técnicas matemáticas similares.
- Una introducción a los componentes electrónicos básicos de la lógica digital como lo son las compuertas lógicas.

13.2 Circuitos eléctricos simples o conmutados

13.2.1 Noción de circuito eléctrico simple

Los circuitos eléctricos simples son circuitos conmutados (figura 13.1) con interruptores que están conformados por una conexión de una fuente de voltaje (V), un interruptor o suiche (S) y una bombilla o lámpara (LAMP). La función de este sistema eléctrico consiste en cerrar o abrir el interruptor para que se encienda o se apague la lámpara, respectivamente.

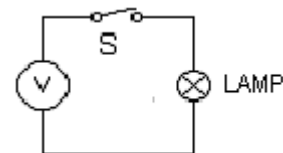


Figura 13.1: Circuito eléctrico simple

Al cerrar o poner el interruptor en estado lógico¹ “1”, se produce el encendido de la lámpara, a esta acción se asignará “1”; al abrir o poner el interruptor en estado en lógico “0”, se produce el apagado de la lámpara, a cuya acción se asignará el estado lógico “0” (vea tabla 13.1).

S	LAMP
0	0
1	1

Tabla 13.1: estado lógica del circuito conmutado

13.2.2 Tipos de circuitos conmutados

Los circuitos conmutados según la distribución de sus interruptores se pueden clasificar así:

- **Circuitos conmutados en serie.** Son aquellos circuitos cuyos interruptores van de manera consecutiva (figura 13.2).

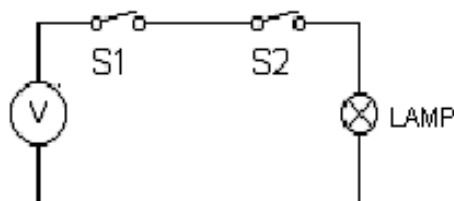


Figura 13.2: circuitos conmutados en serie

- **Circuitos conmutados en paralelo.** Son aquellos circuitos cuyos interruptores van distribuidos en diferentes filas (figura 13.3).

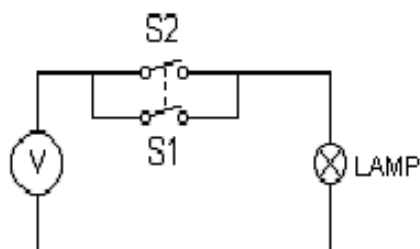


Figura 13.3: circuitos conmutados en paralelo

13.2.3 Funciones lógicas de circuitos conmutados

Una tabla de verdad contiene valores obtenidos de las posibles combinaciones de los valores de los interruptores o pulsadores (o simplemente entradas) también se denomina función lógica.

¹ En circuitos digitales los estados lógicos “1” y “0” también se les denomina nivel “alto” o “high” o H y nivel “bajo” o “low” o L, respectivamente.

Según la distribución de los interruptores o pulsadores en el circuito en serie corresponde a la función lógica denominada función AND y equivale al encendido o apagado de la lámpara.

Ahora, la distribución de los interruptores o pulsadores en el circuito en paralelo se produce la o función OR y corresponde al encendido o apagado de la lámpara.

Función lógica de un circuito conmutado en serie. Un circuito conmutado en serie (figura 13.2) tiene como función, dar encendido a la lámpara, si los interruptores están cerrados o apagarla, si se abre algún interruptor o pulsador. La función lógica de este circuito corresponde a la compuerta AND (tabla 13.2).

Función AND		
S1	S2	LAMP
0	0	0
0	1	0
1	0	0
1	1	1

Tabla 13.2: función lógica AND

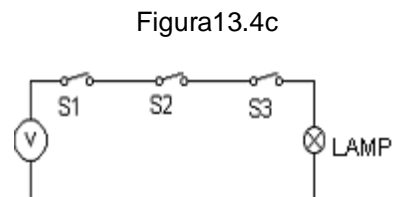
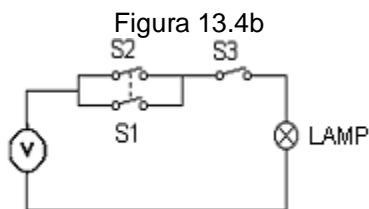
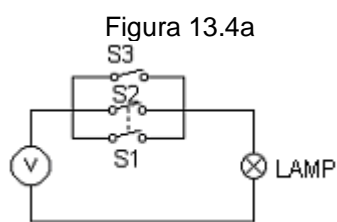
Función lógica de un circuito conmutado en paralelo. Un circuito conmutado en paralelo (figura 13.3) tiene como función darle encendido a la lámpara, siempre que alguno de los interruptores esté cerrado o de apagarla cuando se abren todos los interruptores o pulsadores. La función lógica de este circuito corresponde a la compuerta OR (tabla 13.3).

Función OR		
S1	S2	LAMP
0	0	0
0	1	1
1	0	1
1	1	1

Tabla 13.3: función lógica de OR

Nota: Las funciones lógicas de circuitos conmutados toman valores similares a los de las tablas de verdad de la lógica de proposiciones, solo que en vez de escribir verdadero escriba 1 y en vez de falso escriba 0.

Ejercicio 13.1: Construya la tabla de las funciones lógicas correspondientes a cada uno de los circuitos figuras 13.4a, 13.4b y 13.4c



13.4 Puertas lógicas

13.4.1 Concepto de puerta lógica

Una puerta lógica es un dispositivo físico conformado por elementos eléctricos o electrónicos que realizan una función lógica, cuyo resultado puede hallarse de acuerdo a los distintos valores de las variables de entrada.

Las puertas lógicas básicas son: AND, OR, NOT (o inversor) y las puertas negadas o invertidas de éstas, NAND, NOR, YES o BUFFER. Otro tipo de puerta es XOR y su negación XNOR (o comparadora).

En este texto, las entradas de las puertas se denotarán con letras minúsculas del alfabeto como w, x, y, z; de manera similar se tomará la salida con la letra mayúscula Q. Cada entrada toma los valores lógicos 0 ó 1 e igualmente, en la salida se produce 0 ó 1. Con los valores que toman las entradas y el respectivo resultado de la salida se construyen las tablas de verdad correspondientes a una función lógica.

13.4.2 Función lógica de las puertas

Se denomina función lógica a una tabla de verdad de una puerta o circuito digital. Una función lógica es el resultado obtenido de las posibles combinaciones de los valores de las entradas de un conjunto de puertas lógicas.

Función lógica NOT o Inversor (vea figura 13.5). Esta puerta se utiliza para cambiar el valor de la entrada. En efecto, la salida se obtiene cambiando el valor de la entrada así: cambia 0 por 1 y 1 por 0. La tabla de verdad de esta función lógica esta representada en la tabla 13.4.

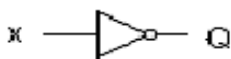


Figura 13.5: símbolo lógico de la puerta Inversor

x	Q
0	1
1	0

Tabla 13.4: función lógica Inversor o NOT

Función lógica de AND (vea figura 13.6). Esta puerta es equivalente a la conjunción vista en lógica de proposiciones y, en efecto, su salida toma el valor 1 si

todas las entradas tienen 1 y toma el valor 0 si alguna de las entradas es 0. La tabla de verdad de esta función lógica esta representada en la tabla 13.5.

x	y	Q
0	0	0
0	1	0
1	0	0
1	1	1

Tabla 13.5: tabla de verdad la compuerta AND

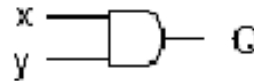


Figura 13.6: símbolo lógico de la compuerta AND

Función lógica de OR (vea figura 13.7). Esta compuerta es equivalente a la disyunción vista en lógica de proposiciones y, en efecto, su salida es 1, si alguna entrada es 1 y toma el valor 0 si todas las entradas son 0. La tabla de verdad de esta función lógica esta representada en la tabla 13.6.

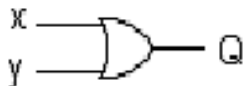


Figura 13.7: símbolo lógico de la compuerta OR

x	y	Q
0	0	0
0	1	1
1	0	1
1	1	1

Tabla 13.6: tabla de verdad la compuerta OR

Función lógica de NAND (vea tabla 13.8). Esta compuerta corresponde a la negación de la compuerta AND; por lo tanto, su función lógica tendría los valores contrarios de la AND. Intuitivamente, se dice que la salida es 1 si alguna entrada es 0 y es 0 si todas las entradas son 1. La tabla de verdad de esta función lógica esta representada en la tabla 13.7.



Figura 13.8: símbolo lógico de la compuerta NAND

x	y	Q
0	0	1
0	1	1
1	0	1
1	1	0

Tabla 13.7: tabla de verdad la compuerta NAND

Función lógica de NOR (vea figura 13.9). Esta compuerta corresponde a la negación de la compuerta OR; por lo tanto, su función lógica tendría los valores contrarios de la OR. Su función lógica se representa en la tabla 13.8. Se puede concluir que la salida es 0, si al menos una entrada es 1 y es 1 si todas las entradas son 0. La tabla de verdad de esta función lógica esta representada en la tabla 13.8.



Figura 13.9: símbolo lógico de la compuerta NOR

x	y	Q
0	0	1
0	1	0
1	0	0
1	1	0

Tabla 13.8: tabla de verdad la compuerta NOR

Función lógica de XOR (vea figura 13.10). La compuerta XOR también se denomina compuerta OR-exclusiva. Su función lógica se representa en la tabla 13.9. Se puede concluir que si las entradas toman valores contrarios la salida es 1 y si son iguales su salida es 0. La tabla de verdad de esta función lógica esta representada en la tabla 13.9.



Figura 13.10: símbolo lógico de la compuerta XOR

x	y	Q
0	0	0
0	1	1
1	0	1
1	1	0

Tabla 13.9: tabla de verdad la compuerta XOR

La compuerta XOR con 3 ó más entradas es 1 si se tiene impar de 1 y es 0 en los demás casos.

Función lógica de XNOR (figura 13.12). La compuerta XNOR también denominada en otros textos como XE. Esta compuerta se utiliza para comparar. Su función lógica se representa en la tabla 12.10, de la cual se puede concluir que la salida es 1 si las entradas son iguales y 0 si las entradas toman valores contrarios.

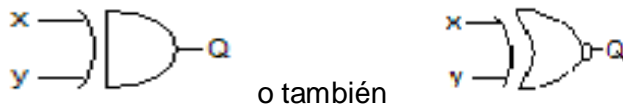


Figura 13.12: símbolo lógico de la compuerta XNOR

x	y	Q
0	0	1
0	1	0
1	0	0
1	1	1

Tabla 13.10: tabla de verdad la compuerta XNOR

La compuerta XNOR con 3 ó más entradas es 0 si se tiene impar de 1 y es 1 en los demás casos.

Ejercicio 13.2: Utilizando las compuertas AND, OR, NAND o NOR dibuje el circuito correspondiente a la compuertas XOR y XNOR.

12.4 Circuitos digitales

Se denomina circuito digital a la interconexión formada por dos o más compuertas lógicas que producen una función lógica. Es decir, es la combinación de varias compuertas lógicas; por eso también le llaman circuitos lógicos combinacionales. Para dibujarlo se utilizan líneas que corresponden a los alambres conductores eléctricos; la conexión de estos alambres se diferenciará del cruce de los cables porque llevarán un punto en dicho enlace (vea figura 13.12).

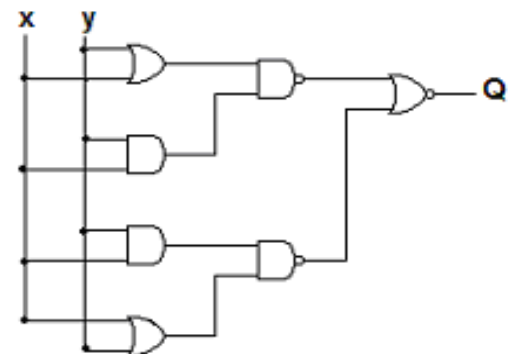


Figura 13.12: Circuito lógico

x	y	Q ₁	Q ₂	Q ₃	Q ₄	Q ₅	Q ₆	Q
0	0	0	0	1	0	0	1	0
0	1	1	0	1	0	1	1	0
1	0	1	0	1	0	1	1	0
1	1	1	1	0	1	1	0	1

Tabla 13.12: Función lógica del circuito lógico de la figura 13.12

13.5 Función lógica de un circuito digital

Función lógica de un circuito digital está formada por las posibles combinaciones de valores de las entradas de la compuerta principal, consignando así la salida final. Las salidas parciales de las diferentes compuertas lógicas conformarán las entradas de las compuertas de un nivel superior.

Se recomienda marcar las salidas parciales de cada compuerta con Q_i (i=1, 2, 3, ...). Los valores obtenidos para cada Q_i servirán como entrada para cada compuerta de nivel superior; luego se evalúa cada compuerta según la función lógica de cada una; siguiendo la secuencia del circuito se continúa hasta obtener el resultado final.

Ejemplo 13.1: Determine la función lógica del circuito de la figura 13.12. Pongamos los Q_i en la salida de cada compuerta como en la figura 13.13 que da como resultado la tabla 13.12.

Nota: Un circuito lógico como la siguiente:

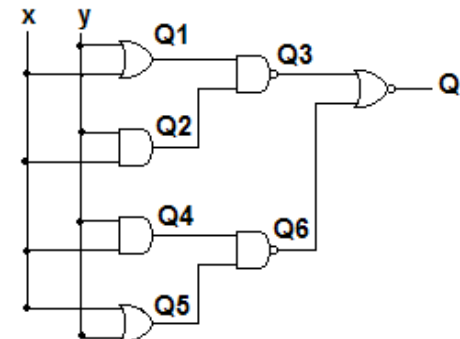


Figura 13.13: Circuito del ejemplo 12.1

Ejercicio 13.3: Determine la función lógica de cada uno de los circuitos digitales de las figuras 13.14a hasta 13.14d:

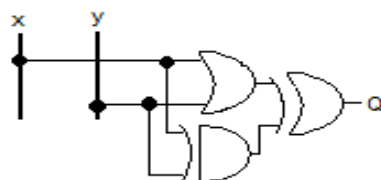


Figura 13.14a

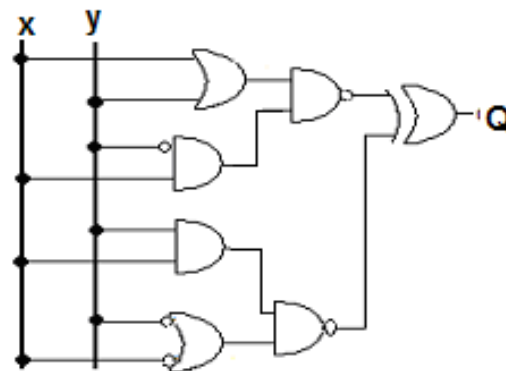


Figura 13.14b

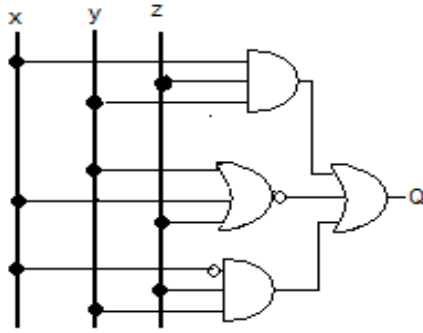


Figura 13.14c

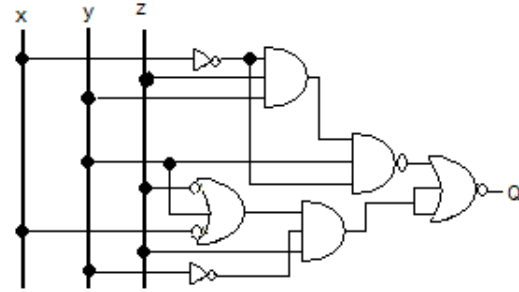


Figura 13.14d

13.7 Concepto de álgebra booleana

El nombre de álgebra booleana se dio en memoria a su descubridor George Boole. Mucho antes que Shannon, E. V. Huntington en 1904 ya había establecido la definición de “álgebra booleana” como un sistema axiomático consistente, completo e independiente que comprende un conjunto B con mínimo dos elementos digamos w, x, y, z (entradas), Q (salidas); dos operaciones binarias (suma +, producto \cdot); una operación unitaria (negación $\bar{}$) y que cumplen los siguientes axiomas:

- Modulativa: $x+0=0+x=x$
 $x \cdot 1=1 \cdot x=x$
- Conmutativa: $x+y=y+x$
 $x \cdot y=y \cdot x$
- Distributiva: $x \cdot (y+z)=x \cdot y+x \cdot z$
 $x+(y \cdot z)=(x+y) \cdot (x+z)$
- Complemento: $\bar{\bar{x}} + x = x + \bar{\bar{x}} = 1$
 $\bar{\bar{x}} \cdot x = x \cdot \bar{\bar{x}} = 0$

Mediante el álgebra booleana se puede diseñar un circuito digital y simplificarlo hasta expresarlo como la mínima expresión booleana que por tanto contendría el menor número de compuertas lógicas posibles al momento de implementarle físicamente, utilizando las entradas del circuito como variables y las cantidades 0 y 1 como sus únicos valores posibles para dichas entradas booleanas.

13.8 Elementos del álgebra booleana

Los siguientes elementos se utilizan para escribir expresiones booleanas que representan simbólicamente un circuito digital. Cada expresión lógica tiene su respectiva función lógica. Los elementos del álgebra booleana son:

Símbolos numéricos. Se representan por 0 y 1; corresponden a los posibles estados de una compuerta lógica (en la entrada o salida). Es 0, cuando el nivel de voltaje es bajo (0 a 0.8V) equivalente a desactivado (interruptor abierto) en lógica positiva y es 1, si el nivel de voltaje es alto (2,5 a 5V) equivalente a activado (interruptor cerrado) en lógica positiva.

Constantes. Es una señal de entrada o salida que no cambia (siempre 0 ó 1).

Variables. Es una señal que cambia con el tiempo, denominadas entradas y salidas; representadas por las letras minúsculas o mayúsculas del alfabeto respectivamente, así: las entradas por w, x, y, z; la salida por Q.

Signos de agrupación. Se utilizan paréntesis izquierdo “(“ y paréntesis derecho “)“.

Símbolos operacionales. Las únicas operaciones del álgebra booleana son: OR (+), AND (•) y la negación o inversor NOT ($\overline{}$ o vínculo superior).

Símbolo relacional. Como único símbolo relacional se utiliza el signo “=”.

13.9 Expresiones booleanas

Al igual que en álgebra tradicional, también se trabaja con letras del alfabeto para denominar variables y formar ecuaciones para obtener el resultado de ciertas operaciones mediante una ecuación o expresión booleana. Efectivamente, los resultados de las correspondientes operaciones también serán binarios.

Ejemplo 13.2: las siguientes expresiones son booleanas con x, y, z entradas y Q la salida

$$\overline{x.y} + x.y = Q$$

$$x + y.z + \overline{y.z} + x.z = Q$$

13.10 Transcripción de circuito digital a expresión booleana

Para transcribir un circuito digital a una expresión booleana escriba la expresión correspondiente a cada compuerta del circuito comenzando siempre por las entradas de las más independientes hasta escribir la operación de la compuerta principal.

Ejemplo 12.3: Escriba la expresión booleana correspondiente a cada uno de los circuitos de las figuras 13.15 y 13.16.

La expresión lógica de la figura 13.15 se podría obtener así: determine las salidas parciales: $Q_1=x+y$, $Q_2=x.y$, $Q_4=x.y$, $Q_5=x+y$

Las salidas Q_1 y Q_2 entran a la compuerta NAND que produce la salida Q_3 y en efecto su expresión booleana parcial será: $Q_3=\overline{Q_1.Q_2}$

Las salidas Q_4 y Q_5 son entradas de la compuerta NAND que produce la salida parcial $Q_6 = \overline{Q_4 \cdot Q_5}$

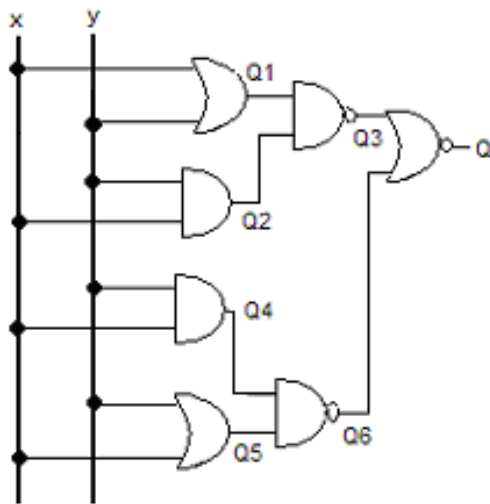


Figura 13.15

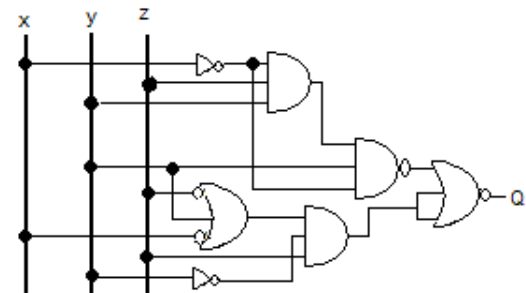


Figura 13.16

Las salidas Q_3 y Q_6 son entradas de la compuerta NOR que es la compuerta principal y en efecto la salida final será: $Q = \overline{Q_3 + Q_6}$

Por lo tanto, la expresión resultante será: $Q = \overline{(\overline{x + y}) \cdot x \cdot y + \overline{x \cdot y} \cdot (x + y)}$

Procediendo de igual manera, la expresión lógica de la figura 13.16 queda:

$$Q = \overline{((\overline{x \cdot y \cdot z}) \cdot x \cdot y) + ((\overline{x + y + z}) \cdot \overline{y} \cdot z)}$$

13.11 Transcripción de expresión booleana a circuito digital

Para transcribir una expresión booleana a circuito digital se comienzan a dibujar las compuertas correspondientes a las operaciones indicadas más interna, siguiendo de manera lógica a dicha expresión.

Ejemplo 13.4: Dibuje el circuito digital correspondiente a la siguiente expresión booleana $Q = \overline{x + y \cdot y + (x + y) \cdot x}$

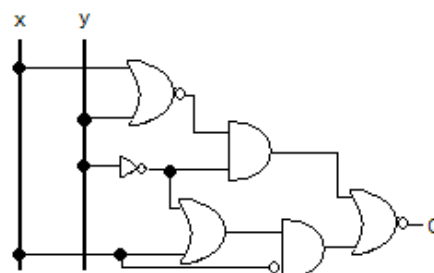


Figura 13.17

Ejercicio 13.4:

1. Dibuje el circuito digital correspondiente a cada una de las siguientes expresiones booleanas utilizando el mínimo de compuertas posible (sin simplificar la expresión):

- a) $Q = (\bar{x} + \bar{y}).(x + \bar{y})$
- b) $Q = (x + \overline{y.x}).(\overline{y + x.y}).x$
- c) $Q = \overline{x.z + y + x.y + z.x}$
- d) $Q = x.(\overline{x + y.x + y + \bar{y} + x}).\bar{z}$
- e) $Q = \overline{x + y}.(x + y)y + (z.(x + y)).x$

2. Escriba la expresión lógica de cada uno de los circuitos dados en las figuras 13.18a hasta 13.18d:

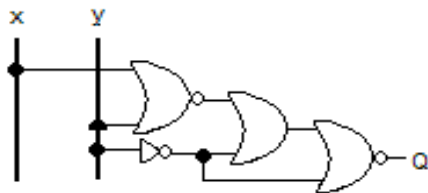


Figura 13.18a

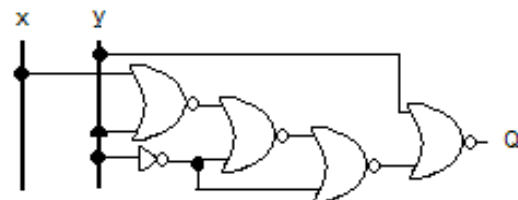


Figura 13.18b

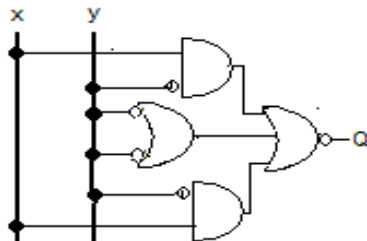


Figura 13.18c

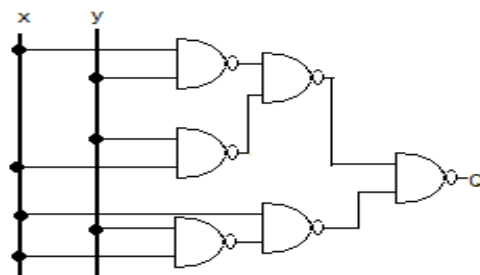


Figura 13.18d

13.12 Velocidad y Retraso de Propagación

La velocidad en la que opera un circuito digital corresponde a la rapidez con que puede completar una tarea. Las limitaciones en velocidad surgen principalmente de dos fuentes:

1. El número de compuertas que una señal encuentra desde el punto de entrada al circuito y hasta la salida (que se conoce como camino lógico)
2. El retraso encontrado por una señal en transitar por una compuerta el cual no solamente proviene del tiempo que requieren los transistores en cambiar de estado, sino también del tiempo que requiere la capacitancia de las compuertas en cargarse y descargarse. Un retraso de propagación es un retardo que existe en una aplicación desde las entradas hasta obtener la salida de un dispositivo o compuerta lógica. Por ejemplo, en las compuertas de tipo NAND y NOR los retrasos de propagación son de 10 ns (nanosegundos, $1 \text{ ns} = 10^{-9}$ segundos).

13.13 Postulados del álgebra booleana

Según la enciclopedia Lexis, “un postulado es una proposición cuya verdad se admite sin pruebas y que es necesaria para servir de base en posteriores razonamientos”. En efecto, el álgebra booleana sugiere los siguientes postulados:

a) $0+0=0$

b) $0.0=0$

c) $0+1=1$

d) $0.1=0$

e) $1+0=1$

f) $1.0=0$

g) $1+1=1$

h) $1.1=1$

i) $\overline{0}=1$

j) $\overline{1}=0$ tabla 13.12

13.14 Leyes del álgebra booleana

Partiendo de la premisa que el álgebra booleana se utiliza para verificar la consistencia de las funciones lógicas de circuitos integrados o chips, se puede asegurar que las leyes de esta álgebra son una herramienta fundamental que se utiliza para simplificar expresiones booleanas complejas y por ende, para simplificar circuitos digitales. Las leyes del álgebra booleana son las mismas utilizadas en el álgebra proposicional, con la diferencia que sus valores constantes ya no serán V y F sino 1 y 0 (vea tabla 13.13).

12.12 Simplificación de expresiones booleanas

Las leyes y los postulados booleanos permiten obtener de una expresión compleja otras expresiones idénticas más simples (generalmente con menos variables); igualmente se obtendrá su correspondiente circuito digital con menos compuertas lo cual permite que los retardos de propagación sean menores debido a que los bits tienen que pasar por el conjunto de compuertas y de tal manera pasaría por menos de las compuertas de las que pasaría originalmente. Esto es muy conveniente no solamente por el aspecto económico en la construcción de un sistema electrónico digital complejo (por ejemplo, un computador), sino además por la velocidad en el transporte de los bits.

El método de simplificar expresiones booleanas será el mismo que se ha utilizado para hacer una demostración; es decir, se pone una columna con las expresiones obtenidas y otra con la justificación (ley o postulado que se utilizó) de cada paso dado hasta alcanzar la mínima expresión.

NOMBRE DE LA LEY	LOGICA DE PROPOSICIONES
1. Idempotencia	<ul style="list-style-type: none"> $x \cdot x = x$ $x + x = x$
2. Identidad	<ul style="list-style-type: none"> $x \cdot 1 = x$ $x + 0 = x$
3. Dominación	<ul style="list-style-type: none"> $x \cdot 0 = 0$ $x + 1 = 1$
4. Conmutativa	<ul style="list-style-type: none"> $x \cdot y = y \cdot x$ $x + y = y + x$
5. Asociativa	<ul style="list-style-type: none"> $x \cdot y \cdot z \Leftrightarrow (x \cdot y) \cdot z \Leftrightarrow x \cdot (y \cdot z)$ $x + y + z = (x + y) + z = x + (y + z)$
6. Distributiva	<ul style="list-style-type: none"> $x \cdot (y + z) = (x \cdot y) + (x \cdot z)$ $x + (y \cdot z) = (x + y) \cdot (x + z)$
7. Complementación: <ul style="list-style-type: none"> Contradicción Tercero excluido 	<ul style="list-style-type: none"> $x \cdot \bar{x} = 0$ $x + \bar{x} = 1$
8. Involución	<ul style="list-style-type: none"> $\bar{\bar{x}} = x$ doble negación $\bar{\bar{\bar{x}}} = \bar{x}$ triple negación
9. D' Morgan	<ul style="list-style-type: none"> $\overline{x \cdot y} = \bar{x} + \bar{y}$ $\overline{x + y} = \bar{x} \cdot \bar{y}$
10. Absorción	<ul style="list-style-type: none"> $x \cdot (x + y) = x$ $x + (x \cdot y) = x$
12. Booleana	<ul style="list-style-type: none"> $x \cdot (\bar{x} + y) = x \cdot y$ $x + (\bar{x} \cdot y) = x + y$

Tabla 13.13 Leyes del álgebra booleana

Ejemplo 13.5: Simplifique la siguiente expresión $Q = (\bar{x} + \bar{y}) \cdot \bar{z} + \overline{x \cdot y}$, hasta obtener la mínima expresión. Determine la cantidad de compuertas que se ahorra.

EXPRESION	JUSTIFICACIÓN
$Q = (\bar{x} + \bar{y}) \cdot \bar{z} + \overline{x \cdot y}$	Expresión original
$Q = \bar{x} \cdot \bar{y} \cdot \bar{z} + \bar{x} \cdot \bar{y}$	Ley de D'Morgan
$Q = \bar{x} \cdot \bar{y} \cdot (\bar{z} + 1)$	Ley distributiva
$Q = \bar{x} \cdot \bar{y}$	Ley de identidad

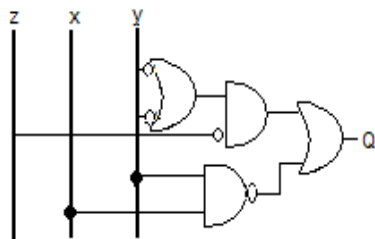


Figura 13.19a: circuito original de la expresión del ejemplo 13.5



Figura 13.19b: circuito resultante de la expresión del ejemplo 13.5

Nótese que el circuito original (figura 13.19) tiene 7 compuertas: 3 NOT, 1 AND, 1 NAND y 2 OR.; en cambio el circuito resultante (figura 10.19b) quedó con solamente una compuerta (1 NAND), lo cual solo produciría el retraso de propagación que técnicamente igual al que haya programado el fabricante al pasar por cada compuerta.

Ejercicio 13.6

1) Simplifique cada una de las siguientes compuertas, justificando cada paso realizado; dibuje ambos circuitos y determine la cantidad de compuertas que se ahorra.

a) $Q = \bar{x} + \bar{y} + (\bar{x} + \bar{y}).z$

b) $Q = \bar{x}.y + x.z + x.z$

c) $Q = y + z.y + \bar{x}.y.w$

d) $Q = \bar{x}.y + \bar{x}$

e) $Q = x + \bar{x}.y + \bar{y}.x$

f) $Q = x.(\bar{x} + y).(\bar{x} + \bar{y})$

g) $Q = x.\bar{y} + \bar{y}.x$

h) $Q = w.(y + w).(w + x + y +)$

i) $Q = (\bar{x} + \bar{y}).\bar{z} + \bar{x}.y$

j) $Q = (\bar{w}.x + y).(\bar{x} + \bar{w})$

k) $Q = (x + y).\bar{z}.z + x).z$

l) $Q = y.(x.(\bar{x}.z + w)) + x.(\bar{y}.z + \bar{x}).y$

13.13 Formas útiles de expresiones booleanas

Las expresiones booleanas en aplicaciones tecnológicas pueden se representadas por dos formas especiales que muy útiles, son ellas: la **forma normal disyuntiva** (FND o expresada en minterms) y la **forma normal conjuntiva** (FNC o expresada en maxterms).

La cantidad de términos de estas formas depende de la cantidad de variables que se van a utilizar y se expresa 2^n con n = cantidad de variables. Es decir, si $n=2$, la FND o FNC constará de 4 posibles combinaciones de términos producto o suma respectivamente y si es de 3 variables será de 8 posibles combinaciones de términos.

13.13.1 Forma normal disyuntiva

Es aquella expresión booleana formada como una suma de productos que involucra todas las variables con o sin negación; es decir, cada término que la compone está escrito como un producto de sus variables. Cada producto se denomina término minimal o miniterm. La expresión se denomina función polinomial minterms o minimales y denota $f(x)$ para expresiones de una entrada, $f(x,y)$ para expresiones de dos entradas o $f(x,y,z)$ para expresiones de tres entradas y así sucesivamente. Por ejemplo, la función $f(x, y, z) = (x.y.z) + (\bar{x}.\bar{y}.z) + (x.y.\bar{z})$ que tiene 3 variables y tres términos minterms.

El complemento de $f(x,y,z)$ son los términos que le faltan a la expresión para que la FND quede completa, en efecto estos son: $\bar{f}(x, y, z) = \bar{x}.\bar{y}.\bar{z} + x.\bar{y}.\bar{z} + x.y.\bar{z} + \bar{x}.y.\bar{z} + \bar{x}.y.z$

Dada una expresión booleana escrita en sumas de productos incompleta, para obtener una FND se procede por simple inspección así:

- Aplique la ley de D'Morgan para quitar los complementos que estén indicados hasta que aparezcan solamente aplicados a las variables individuales.
- Aplique la ley distributiva del producto sobre la suma; de tal manera la expresión puede quedar reducida a un polinomio.
- Observe cada término de la función los cuales deben tener todas las variables que tiene la función. Si en algún término faltase una entrada entonces se multiplica el término por la suma de esta variable con su complemento, ya que por la ley de complementación es 1; por ejemplo, si faltase x se multiplica el término por la suma $x + \bar{x}$.
- Aplique ley de idempotencia para reducir los términos semejantes (o repetidos) y otras leyes reconocidas para simplificar.

Ejemplo 13.6: Escriba en FND la expresión $f(x, y, z) = \overline{x + y.z + x.(y + z)}$.

Aplicando los pasos anteriores dados en la sección 13.13.1 se tiene:

$$\begin{aligned}
 f(x, y, z) &= x + y + z + x + \bar{y}.z && \text{Ley de D'Morgan} \\
 &= x + y + z && \text{Ley identidad booleana e idempotencia} \\
 &= x.(y + \bar{y}).(z + \bar{z}) + (x + \bar{x}).y.(z + \bar{z}) + (x + \bar{x}).(y + \bar{y}).z && \text{Ley identidad y complementación} \\
 &= x.y.z + x.y.\bar{z} + x.\bar{y}.z + x.\bar{y}.\bar{z} + \bar{x}.y.z + \bar{x}.y.\bar{z} + \bar{x}.\bar{y}.z && \text{Ley distributiva e idempotencia}
 \end{aligned}$$

13.13.2 Forma normal conjuntiva

Es aquella expresión booleana formada como un producto de sumas que involucra todas las variables con o sin negación; es decir, cada término que la compone está escrito como una suma de sus variables. Cada suma se denomina término maximal o maxiterm. La expresión se denomina función polinomial maxiterms o maximales. Por

ejemplo, $f(x, y, z) = (x + y + z) \cdot (\bar{x} + \bar{y} + z) \cdot (x + y + \bar{z})$ que es una función de tres variables con tres términos maxiterms

Dada una expresión escrita de manera libre, a partir de ella se puede obtener la forma normal conjuntiva y esta puede usarse para hallar su complemento (vea ejemplo 12.7).

Dada una expresión booleana escrita en productos incompletos de sumas, para obtener una FNC se procede así:

- Aplique la ley de D'Morgan para quitar los complementos que estén indicados hasta que aparezcan solamente aplicados a las variables individuales.
- Aplique la ley distributiva de la suma sobre el producto (factorización).
- Observe cada término de la función los cuales deben tener todas las variables que tiene la función. Si en algún término faltase una entrada entonces se suma al término el producto de esta variable con su complemento, porque por ley de complementación es 0; por ejemplo, si faltase x se agrega al término $x \cdot \bar{x}$.
- Aplique ley de idempotencia para reducir los términos semejantes (o repetidos).

Ejemplo 13.7: Escriba en FNC la expresión $f(x, y, z) = \overline{x + y \cdot z} + \overline{x \cdot (y + z)}$.

Para tal fin, aplique a la expresión los siguientes pasos:

$f(x, y, z) = \bar{x} \cdot \bar{y} \cdot \bar{z} + (x + \bar{y} \cdot z)$	Ley de D'Morgan
$= \bar{x} \cdot \bar{y} \cdot \bar{z} + x \cdot \bar{y} + x \cdot z$	Ley distributiva
$= \bar{x} \cdot \bar{y} \cdot \bar{z} + x \cdot \bar{y} \cdot (z + \bar{z}) + x \cdot (y + \bar{y}) \cdot z$	Ley de complementación
$= \bar{x} \cdot \bar{y} \cdot \bar{z} + x \cdot \bar{y} \cdot z + x \cdot \bar{y} \cdot \bar{z} + x \cdot y \cdot z + x \cdot \bar{y} \cdot z$	Ley distributiva y FND
$= ((\bar{x} \cdot \bar{y} \cdot \bar{z} + x \cdot \bar{y} \cdot z + x \cdot \bar{y} \cdot \bar{z} + x \cdot y \cdot z + x \cdot \bar{y} \cdot z)')'$	Doble complemento
$= ((x + y + z) \cdot (\bar{x} + y + \bar{z}) \cdot (\bar{x} + y + z) \cdot (\bar{x} + \bar{y} + \bar{z}) \cdot (\bar{x} + y + \bar{z}))'$	Ley de D'Morgan
$= (\bar{x} + \bar{y} + \bar{z}) \cdot (x + \bar{y} + z) \cdot (x + \bar{y} + \bar{z}) \cdot (x + y + z) \cdot (x + \bar{y} + z)$	Halla complemento

Observe el primer apóstrofe (complemento) aplica la ley de D'Morgan y el segundo halla el complemento, es decir, busca los términos que faltan para completar la FNC.

Dada una función lógica $f(x, y, z)$ en una tabla de valores para representarla en FNC (maxterms) tome de la tabla los 0 y en FND (minterms) tome los 1.

12.14 Simplificación de expresiones booleanas con compuertas NAND o NOR

Las diferentes empresas fabricantes de circuitos integrados y chips buscan la eficiencia de sus construcciones fabricándolas en serie y de manera automatizada; para tal fin, por consideraciones constructivas, utilizan solo compuertas NAND o NOR, es decir, diseñan los circuitos digitales con únicamente compuertas NAND o NOR, respectivamente.

Con el fin de lograr tales técnicas obtenga primero la expresión óptima; luego, utilice la ley de involución, y por último, aplique la ley de D'Morgan, si es necesario.

Ejemplo 13.8: Escriba la expresión óptima utilizando solamente compuertas NOR que corresponde a la expresión $Q = \overline{x} \cdot \overline{(y + w)} \cdot ((\overline{w + x}) + \overline{x})$.

$$\begin{aligned}
 Q &= \overline{x} \cdot \overline{(y \cdot w)} \cdot ((\overline{w + x}) + \overline{x}) \\
 &= \overline{x} \cdot \overline{(y \cdot w)} \cdot (\overline{w} + \overline{x}) \\
 &= \overline{x} \cdot \overline{y \cdot w} \cdot \overline{w} + \overline{x} \cdot \overline{y \cdot w} \cdot \overline{x} \\
 &= 0 + \overline{x} \cdot \overline{y \cdot w} \\
 &= \overline{x \cdot y \cdot w} \\
 &= \overline{(x + y + w)}
 \end{aligned}$$



Figura 13.20: Circuito óptimo equivalente de la expresión del ejemplo 13.8

El circuito óptimo equivalente a la expresión está dado en la figura 13.20

13.15 Mapas de Karnaugh

El mapa de Karnaugh fue inventado en 1950 por Maurice Karnaugh, un físico y matemático estadounidense. Se Graduó en la universidad de Yale en el 1952, es actualmente gobernador emérito del ICC (International Council for Computer Communication). Ha trabajado como investigador en los Laboratorios Bell desde 1952 a 1966 y en el centro de investigación de IBM de 1966 a 1993. Así mismo, ha impartido de informática en el Politécnico de Nueva York de 1980 a 1999, y desde 1975 es miembro del IEEE (Institute of Electrical and Electronics Engineers), por sus aportaciones sobre la utilización de métodos numéricos en las telecomunicaciones.

Los mapas K aprovechan la capacidad del cerebro humano de trabajar mejor con patrones que con ecuaciones y otras formas de expresión analítica. Externamente, un mapa de Karnaugh consiste de una serie de cuadrados, cada uno de los cuales representa una línea de la tabla de verdad. Puesto que la tabla de verdad de una función de N variables posee 2^N filas, el mapa K correspondiente debe poseer también 2^N cuadrados. Cada cuadrado alberga un 0 ó un 1, dependiendo del valor que toma la función en cada fila. Las tablas de Karnaugh se pueden utilizar para funciones de hasta 6 variables.

Estos mapas han sido creados con el fin de obtener expresiones lógicas más simples y por ende circuitos digitales más simples y más económicos, que producen menos retardos de propagación y por lo tanto, serán de menor tamaño.

13.15.1 Concepto de mapas de Karnaugh

Hasta el momento se pueden simplificar expresiones booleanas utilizando las leyes del álgebra booleana, pero existe una herramienta importante que permite simplificar

expresiones complejas escritas en FND o FNC denominada “**mapa de Karnaugh**”. Dichos mapas se utilizan para simplificar expresiones escritas en estas formas sin tener que recurrir a las leyes del álgebra booleana.

Un mapa de Karnaugh es un arreglo rectangular o cuadrado de celdas adyacentes (comparten un mismo termino) con los cuales se representa de manera gráfica una expresión booleana escrita en FND o en FNC. En síntesis es una tabla de verdad en la que se escribe **1** en la intersección de la fila y la columna para indicar que el término formado por las variables de esa fila y esa columna, existe, o se escribe **0**, si ese término no existe.

Los mapas se clasifican según el número de variables que representan así:

- Mapas de Karnaugh de 2 variables (Tabla 13.14a)
- Mapas de Karnaugh de e 3 variables (Tabla 13.14b)
- Mapas de Karnaugh de e 4 variables (Tabla 13.14c)

En términos prácticos, los mapas de Karnaugh son útiles hasta no más de seis variables caso tal que usa 2 mapas de 4 variables. Para un número mayor de variables, se acude a otras técnicas de simplificación como el método de Quine McCluskey.

Observe, el orden de las negaciones de las variables es estricto, mas no el nombre de las variables, es decir, el resultado no cambia si donde se escribió x se haya escrito y y viceversa, pero sin afectar el orden de las negaciones en cada mapa.

	\bar{y}	y
\bar{x}		
x		

Tabla 13.14a: mapa de Karnaugh de 2 variables

	$\bar{y}.\bar{z}$	$\bar{y}.z$	$y.z$	$y.\bar{z}$
\bar{x}				
x				

Tabla 13.14b: mapa de Karnaugh de 3 variables

	$\bar{y}.\bar{z}$	$\bar{y}.z$	$y.z$	$y.\bar{z}$
$\bar{w}.x$				
$w.x$				
$w.\bar{x}$				
$\bar{w}.\bar{x}$				

Tabla 13.14c: mapa de Karnaugh de 4 variables

	$\bar{x}.\bar{y}.\bar{z}$	$\bar{x}.\bar{y}.z$	$\bar{x}.y.\bar{z}$	$\bar{x}.y.z$	$x.\bar{y}.\bar{z}$	$x.\bar{y}.z$	$x.y.\bar{z}$	$x.y.z$
$\bar{v}.\bar{w}$								
$\bar{v}.w$								
$v.\bar{w}$								
$v.w$								

Tabla 13.14d: mapa de Karnaugh de 5 variables v, w, x, y, z.

También puede utilizar 2 mapas de Karnaugh de 4 entradas para simplificar una expresión de 5 variables; basta con poner una 5° variable afirmativa y también negativa, con las respectivas combinaciones.

13.15.2 Adyacencias en los mapas de Karnaugh

Se dice que dos celdas están adyacentes si dichas celdas están contiguas, mas no de manera diagonal; es decir, si sus direcciones respectivas difieren en un solo dígito. Las adyacencias de 1 se agrupan en potencias de 2 así: de a dos, de a cuatro, de a ocho y de a dieciséis 1 contiguos. También hay adyacencias cuando los 1 están ubicados en los extremos opuestos de las filas o columnas o que comparten las esquinas.

Los mapas de Karnaugh permiten representar expresiones de máximo 6 variables con los cuales se hacen simplificación de variables así:

- Adyacencias de dos 1^s simplifican 1 variable
- Adyacencias cuatro 1^s simplifican 2 variables
- Adyacencias ocho 1^s simplifican 3 variables
- Adyacencias dieciséis 1^s simplifican 4 variables
- Adyacencias de treinta y dos 1^s simplifican 5 variables

Ejemplo 13.9: Las Tabla 13.15a hasta Tabla 13.15f corresponden a mapas de Karnaugh que representan diferentes expresiones booleanas.

	$\bar{y}.\bar{z}$	$\bar{y}.z$	$y.\bar{z}$	$y.z$
$\bar{w}.\bar{x}$	1	1		
$\bar{w}.x$	1	1		
$w.\bar{x}$	1	1		
$w.x$	1	1		

Tabla 13.15a: mapa de Karnaugh que simplifica 3 variables

	$\bar{y}.\bar{z}$	$\bar{y}.z$	$y.\bar{z}$	$y.z$
$\bar{w}.\bar{x}$		1	1	
$\bar{w}.x$				
$w.\bar{x}$				
$w.x$		1	1	

Tabla 13.15b: mapa de Karnaugh que simplifica 2 variables

	$\bar{y}.\bar{z}$	$\bar{y}.z$	$y.\bar{z}$	$y.z$
$\bar{w}.\bar{x}$	1			1
$\bar{w}.x$				
$w.\bar{x}$				
$w.x$	1			1

Tabla 13.15c: mapa de Karnaugh que simplifica 2 variables

	$\bar{y}.\bar{z}$	$\bar{y}.z$	$y.\bar{z}$	$y.z$
$\bar{w}.\bar{x}$				
$\bar{w}.x$	1			1
$w.\bar{x}$	1			1
$w.x$				

Tabla 13.15d: mapa de Karnaugh que simplifica 2 variables

	$\bar{y}.\bar{z}$	$\bar{y}.z$	$y.\bar{z}$	$y.z$
$\bar{w}.\bar{x}$	1	1		
$\bar{w}.x$	1	1		
$w.\bar{x}$	1	1		
$w.x$			1	

Tabla 13.15e: mapa de Karnaugh con varias adyacencias

	$\bar{y}.\bar{z}$	$\bar{y}.z$	$y.\bar{z}$	$y.z$
$\bar{w}.\bar{x}$	1			1
$\bar{w}.x$		1		1
$w.\bar{x}$	1		1	
$w.x$		1		1

Tabla 13.15f: mapa de Karnaugh sin adyacencias

Ejercicio 13.7: Escriba las expresiones booleanas correspondientes a cada uno de los mapas de Karnaugh dados en el ejemplo 13.9.

12.15.3 Simplificación de expresiones booleanas mediante mapas de Karnaugh

En la simplificación con mapas de Karnaugh debe considerarse no realizar muchas agrupaciones, porque a mayor cantidad de agrupaciones mayor cantidad de términos. Por lo tanto, optimizar los circuitos es obtener la menor cantidad de términos con la menor cantidad de variables y en efecto, debe realizar la menor cantidad posible de agrupaciones en el mapa, pero abarcando el máximo de adyacencias posibles.

La optimización de los circuitos es muy importante desde el punto de vista técnico y económico debido a que se mejora la velocidad de respuesta, se disipa menor potencia, se disminuyen los costos (menos compuertas), entre otros conceptos.

Ejemplo 13.10: Determine la expresión resultante de la utilización de los mapas de Karnaugh del ejemplo 13.9.

El mapa de la Tabla 13.15a representa a una expresión con 8 términos; tiene una adyacencia de 8 unos que en efecto simplificará 3 variables. La expresión resultante es:

$$Q = \bar{y}$$

El mapa de la Tabla 13.15b representa a una expresión con 4 términos; tiene una adyacencia de 4 unos que en efecto simplificará 2 variables. La expresión resultante es:

$$Q = \bar{x}.z$$

El mapa de la Tabla 13.15c representa a una expresión con 4 términos; tiene una adyacencia de 4 unos que en efecto simplificará 2 variables. La expresión resultante es:

$$Q = \bar{x}.\bar{z}$$

El mapa de la Tabla 13.15d representa a una expresión con 4 términos; tiene una adyacencia de 4 unos que en efecto simplificará 2 variables. La expresión resultante es:

$$Q = x.\bar{z}$$

El mapa de la Tabla 13.15e representa a una expresión con 8 términos; tiene tres adyacencias repartidas en dos de 4 unos, una de dos unos y un término que no se simplifica. En efecto simplificará las variables así:

- En la primera adyacencia (de 4 unos) que reduce 4 términos de 4 variables, a uno solo con dos variables; en efecto, el queda: $Q = \bar{w}.\bar{y}$
- En la segunda adyacencia (de 4 unos) que reduce 4 términos de 4 variables a uno solo con dos variables; en efecto, el queda: $Q = x.\bar{y}$
- En la tercera adyacencia (de 2 unos) que reduce 2 términos con 4 variables a uno solo con tres variable; en efecto, el término queda: $Q = \bar{w}.x.z$
- El término que no se simplifica es: $Q = w.\bar{x}.y.z$
- Por lo tanto, la expresión simplificada queda: $Q = w.\bar{x}.y.z + \bar{w}.x.z + x.\bar{y} + \bar{w}.\bar{y}$

El mapa de la Tabla 13.15f representa a una expresión con 8 términos, pero ninguno se simplifica, porque no tiene adyacencias.

Ejemplo 13.12: Simplifique la expresión

$$Q = x.\bar{y}.\bar{z} + w.x.\bar{y}.z + w.\bar{x}.z + \bar{w}.\bar{y}.z + w.\bar{x}.\bar{y}.z + w.x.y.z$$

Redúzcala a una expresión óptima. Dibuje el circuito digital óptimo con compuertas básicas. Dibújelo también utilizando solamente compuertas NAND de dos entradas y determine la cantidad de retrasos de propagación del circuito si se sabe que cada compuerta produce un retardo de 10 ns.

La expresión dada está escrita en FND incompleta. Utilicemos los pasos dados en la sección 13.13 para que sea completa; en efecto, la expresión queda así:

$$Q = \bar{w}.x.\bar{y}.z + w.x.\bar{y}.z + w.x.\bar{y}.z + w.\bar{x}.\bar{y}.z + w.\bar{x}.y.z + \bar{w}.\bar{x}.\bar{y}.z + w.\bar{x}.\bar{y}.z + w.x.y.z + w.\bar{x}.y.z$$

Ahora, simplifiquemos la expresión obtenida aplicando mapas de Karnaugh (tabla 13.16) y la expresión simplificada queda como sigue:

$$Q = \bar{y}.\bar{z} + w.\bar{y} + w.z$$

Para convertir la función anterior en otra óptima aplique la ley distributiva y finalmente quedará así: $Q = \bar{y}.\bar{(z + w)} + w.z$

La expresión booleana óptima utilizando solamente NAND es: $Q = \bar{\bar{y}.\bar{(z.w)}}.\bar{w}.z$

El circuito digital óptimo utilizando diferentes compuertas se da en la figura 13.21. El dibujo del circuito óptimo utilizando solamente compuertas NAND se da en la figura 13.22. Por lo tanto, la cantidad de retardos de propagación es 40 ns.

	$\bar{y}.\bar{z}$	$\bar{y}.z$	$y.z$	$y.\bar{z}$
$\bar{w}.\bar{x}$	1			
$\bar{w}.x$	1			
$w.\bar{x}$	1	1	1	
$w.x$	1	1	1	

Tabla 13.16: mapa de Karnaugh del ejemplo 13.12

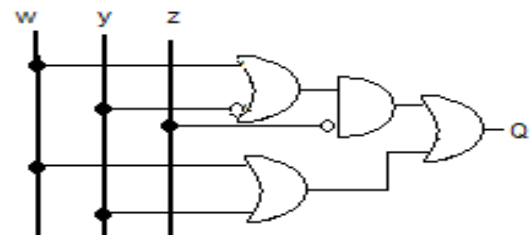


Figura 13.21: circuito lógico de la expresión óptima del ejemplo 13.12

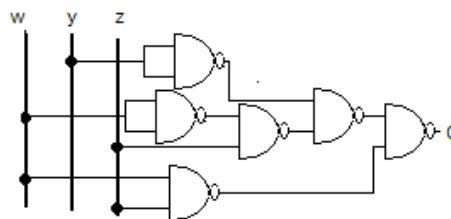


Figura 13.22: circuito digital de la mínima expresión del ejemplo 13.8 con sólo NAND

12.16 Mapas de Karnaugh con términos irrelevantes (condiciones no importa)

Un término irrelevante en una expresión booleana es el resultado de la combinación de valores lógicos de variables no requeridas para la solución de un problema específico. Estos términos se simbolizan en el mapa en vez de 0 ó de 1 por “x”. Dichos símbolos pueden utilizarse para conformar adyacencias con las cuales se ayuda a realizar simplificaciones importantes. En el argot de la electrónica estos términos producen condiciones que se denominan “**don’t care**” o “**no importa**”.

Ejemplo 13.12: Dado el siguiente mapa (Tabla 13.17) con términos irrelevantes marcados con “x”, obtenga la expresión simplificada resultante

La expresión correspondiente

$$f(w, x, y, z) = \overline{w}.x.\overline{y}.z + w.x.y.z$$

La expresión resultante es:

$$f(w, x, y, z) = w.z$$

	$\overline{y}.z$	$\overline{y}.z$	$y.z$	$y.z$
$\overline{w}.x$				
$\overline{w}.x$				
$w.x$		1	1	
$w.x$		x	x	

Tabla 13.17: mapa de Karnaugh del ejemplo 13.12

12.17 Aplicaciones de circuitos digitales

Existe gran variedad de aplicaciones de los circuitos digitales, más aún cuando en la actualidad en casi todas las situaciones del quehacer humano, el hombre quisiera controlar y automatizar desde aparatos para el oficio doméstico, máquinas y herramientas de trabajo hasta un sistema de cómputo.

Con la implementación de funciones booleanas y la construcción de circuitos digitales se han diseñado los denominados circuitos integrados que son construidos con gran cantidad y variedad de compuertas lógicas que se distribuyen y utilizan según la necesidad. En el mercado se encuentran esos componentes electrónicos con su respectivo manual que le permiten al diseñador electrónico construir el circuito, ya sea de automatismo o de control. A continuación veremos algunos ejemplos de la cotidianidad, útiles desde el punto de vista técnico y de la economía.

Ejemplo 13.13: Dada la función lógica de la Tabla 13.18 diseñe el circuito digital para 2 entradas (x, y) y 4 salidas (Q₃, Q₂, Q₁, Q₀), teniendo en cuenta que 0: low y 1: high que utilice solamente compuertas lógicas simples.

x	y	Q ₃	Q ₂	Q ₁	Q ₀
0	0	1	1	1	0
0	1	1	1	0	1
1	0	1	0	1	1
1	1	0	1	1	1

Tabla 13.18

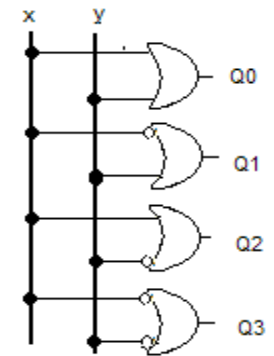


Figura 13.23: circuito del ejemplo 12.13

Llevemos la función lógica de cada Q_i a un mapa de Karnaugh para obtener un resultado simplificado. En efecto, en las tablas 13.19a a 13.19d.

Q ₃		
	\bar{y}	y
\bar{x}	1	1
x	1	0

Tabla13.19a

Q ₂		
	\bar{y}	y
\bar{x}	1	0
x	1	1

Tabla13.19b

Q ₁		
	\bar{y}	y
\bar{x}	1	1
x	0	1

Tabla13.19c

Q ₀		
	\bar{y}	y
\bar{x}	0	1
x	1	1

Tabla13.19d

Los resultados son:

$$Q_3 = \bar{x} + \bar{y}, Q_2 = x + \bar{y}, Q_1 = \bar{x} + y, Q_0 = x + y$$

Ahora el dibujo del circuito con compuertas básicas se ve en la figura 12.23

Ejemplo 13.14: Con el fin de controlar el nivel de agua de una represa se instalan en ella 4 sensores w, x, y, z con los cuales se controlarán: dos compuertas C1 (auxiliar) y C2 (principal) y una alarma, según la figura 13.24, bajo las siguientes condiciones:

- Si el agua está en el cuarto nivel (N4) se activará el cuarto sensor (z); e abrirán ambas compuertas C1 y C2; además, se activará una alarma (A) que indicará posible desbordamiento de la represa.
- Si está por debajo de N4 (entre N3 y N4), entonces se activará el sensor y; abrirán las compuertas 1 (C1) y 2 (C2).
- Si está por debajo de N3 (entre N2 y N3), entonces se activará el sensor x; se abrirá la compuerta 2 (C2), únicamente.
- Si está por debajo de N2 (entre N1 y N2), entonces se activará el sensor w; se abrirá la compuerta 1 (C1), únicamente.
- Si está por debajo de N1 (N0), no se activará ningún sensor; ninguna compuerta se abrirá.

Sensores				Compuertas		
w	x	y	Z	C1	C2	A
0	0	0	0	0	0	0
0	0	0	1	X	X	X
0	0	1	0	X	X	X
0	0	1	1	X	X	X
0	1	0	0	X	X	X
0	1	0	1	X	X	X
0	1	1	0	X	X	X
0	1	1	1	X	X	X
1	0	0	0	1	0	0
1	0	0	1	X	X	X
1	0	1	0	X	X	X
1	0	1	1	X	X	X
1	1	0	0	0	1	0
1	1	0	1	X	X	X
1	1	1	0	1	1	0
1	1	1	1	1	1	1

Tabla 13.21

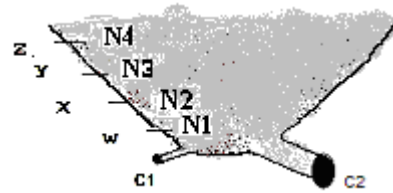


Figura 13.24: Represa con dos compuertas.

	w	x	y	z
N0	0	0	0	0
N1	1	0	0	0
N2	1	1	0	0
N3	1	1	1	0
N4	1	1	1	1

Tabla 13.20:

Diseñe el circuito digital que realice el control para esas compuertas. En efecto, construyamos primero la tabla de verdad correspondiente a los sensores; en efecto, vea la tabla 13.20

La función booleana para las compuertas depende de los sensores y se puede ver la tabla 13.21. Los términos irrelevantes se dan porque no es posible que se activen los sensores de los niveles superiores sin activarse los de los inferiores.

Ahora, construyamos los mapas de Karnaugh y el circuito digital y la alarma (A).

En las tablas 13.2a, 13.22b, 13.22c, respectivamente están las tablas para las compuertas C1, C2 y la alarma A.

C1	$\bar{y}.\bar{z}$	$\bar{y}.z$	$y.\bar{z}$	$y.z$
$\bar{w}.\bar{x}$	0	x	x	x
$\bar{w}.x$	x	x	x	x
$w.\bar{x}$	0	x	1	1
$w.x$	1	x	x	x

Tabla 13.22a

C2	$\bar{y}.\bar{z}$	$\bar{y}.z$	$y.\bar{z}$	$y.z$
$\bar{w}.\bar{x}$	0	x	x	x
$\bar{w}.x$	x	x	x	x
$w.\bar{x}$	1	x	1	1
$w.x$	0	x	x	x

Tabla 13.22b

A	$\bar{y}.\bar{z}$	$\bar{y}.z$	$y.\bar{z}$	$y.z$
$\bar{w}.\bar{x}$	0	x	x	x
$\bar{w}.x$	x	x	x	x
$w.\bar{x}$	0	x	1	0
$w.x$	0	x	x	X

Tabla 13.22c

La $C1 = y + w \cdot \bar{x}$, la $C2 = y + x$ y la alarma $A = z$

Circuito digital se da en la figura 12.25:

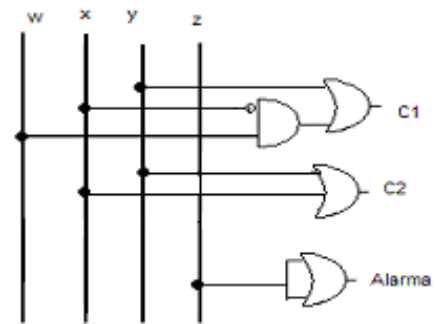
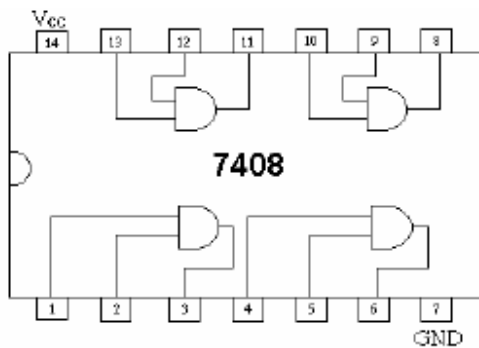
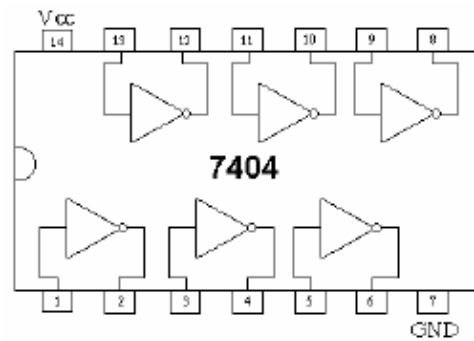


Figura 13.25

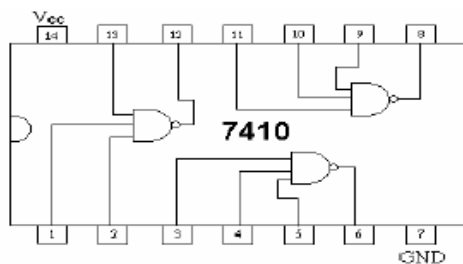
DIAGRAMAS DE COMPUERTAS LÓGICAS TTL



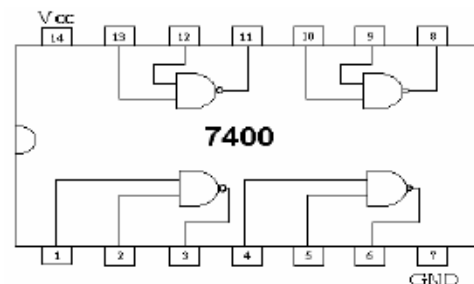
AND de 2 entradas



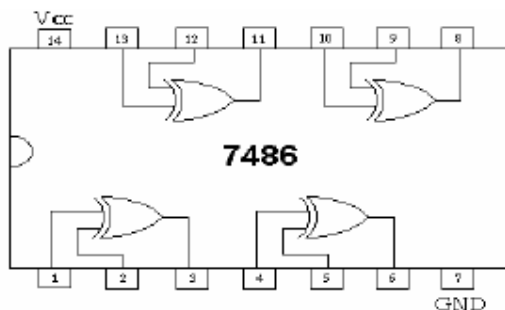
Compuerta INV



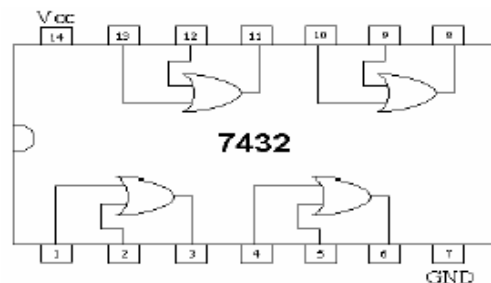
Compuerta NAND de 3 entradas



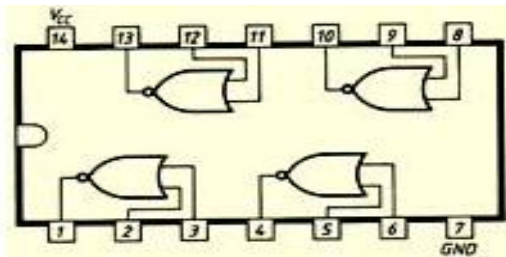
Compuerta NAND



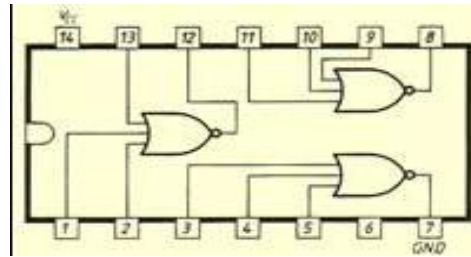
XOR



OR de 2 entradas c/u



NOR(7402) 2 ENTRADAS



NOR (7427) 3 ENTRADAS

AUTOEVALUACIÓN

1. En los problemas 1.1 hasta 1.5 escoja la opción correcta:

- A) Si 1 y 2 son correctas
- B) Si 2 y 3 son correctas
- C) Si 3 y 4 son correctas
- D) Si 2 y 4 son correctas
- E) Si 1 y 3 son correctas

1.1 Dada la expresión booleana siguiente:

$$f(w, x, y, z) = \bar{w}.\bar{x}.y.z + w.x.\bar{y}.z + w.\bar{x}.\bar{y}.z + w.\bar{x}.y.\bar{z} + w.x.y.z + w.\bar{x}.y.z + \bar{w}.x.y.z + w.x.y.\bar{z}$$

Sus circuitos simplificados y óptimos son:

- 1. $y.z + w.z + w.y$
- 2. $z + w.y$
- 3. $z.(w + y) + w.y$
- 4. $w.z + w.\bar{x} + w.y$

Respuesta: _____

1.2 El circuito óptimo obtenido de la tabla 13.23 es:

- 1. $x + \bar{v}.y + v.\bar{y} + z$
- 2. $x + v \oplus y + v \oplus z$
- 3. $x + z + v \oplus y$
- 4. $x + \bar{v}.y + v.\bar{z} + v.\bar{y} + \bar{v}.z$

Respuesta: _____

wx \ vyz	0.0.0	0.0.1	0.1.1	0.1.0	1.1.0	1.1.1	1.0.1	1.0.0
0.0	0	X	X	X	0	X	X	X
0.1	1	X	X	X	1	X	X	X
1.1	1	X	1	1	1	X	1	1
1.0	0	X	X	X	0	X	X	X

Tabla 13.23

Dada la expresión booleana siguiente:

$$f(w, x, y, z) = \bar{w}\bar{x}.y.z + w.\bar{x}.y.z + \bar{w}.x.\bar{y}.z + w.x.\bar{y}.z + \bar{w}.x.y.\bar{z} + w.\bar{x}.y.\bar{z} + \bar{w}.x.y.z + w.x.y.\bar{z}$$

Sus circuitos simplificados y óptimos son:

1. $y.z + w.z + w.y$
2. $z + w.y$
3. $z.(w + y) + w.y$
4. $w.z + \bar{w}.z + w.y$

Respuesta: _____

1.3 El circuito óptimo obtenido de la tabla 13.24 es:

1. $x + \bar{v}.y + v.\bar{y}$
2. $x + v \oplus y + v \oplus z$
3. $x + v \oplus y$
4. $x + \bar{v}.y + v.\bar{z} + v.\bar{y} + v.z$

Respuesta: _____

	$\bar{v}.y.\bar{z}$	$\bar{v}.y.z$	$\bar{v}.y.\bar{z}$	$\bar{v}.y.z$	$v.y.\bar{z}$	$v.y.z$	$v.y.\bar{z}$	$v.y.z$
$\bar{w}.\bar{x}$	0	0	X	X	0	0	x	x
$\bar{w}.x$	1	X	X	X	1	x	x	x
$w.\bar{x}$	1	X	1	1	1	x	1	1
$w.x$	0	X	X	X	0	x	x	x

Tabla 13.24

1.4. La cantidad de compuertas circuito digital óptimo de la expresión $Q = \overline{(x.y + x.\bar{y})}.(x.\bar{y} + x)$ y sus correspondientes compuertas es.

- 1) 3 NAND y 1 NO 2) 1 NOR y 2 NO 3) 1 XNOR 4) 1 AND Respuesta: _____

1.5 El circuito óptimo que representa la tabla 13.25 requiere:

- A) 1 NAND y 2 NOT B) 1 NOR y 1 NOT C) 1 AND y 1 NOT D) 1 NAND y 1 NOT

	$\bar{c}.\bar{d}$	$\bar{c}.d$	$c.d$	$c.\bar{d}$
$\bar{a}.\bar{b}$	1	1	1	1
$\bar{a}.b$	1	1	1	1
$a.b$	1	1	1	1
$a.\bar{b}$	0	0	0	0

Tabla 13.25

2. Resuelva los 2.1 a 2.3 según las tablas 13.26, 13.27 y 13.28.

	$\bar{y}.z$	$\bar{y}.\bar{z}$	$y.z$	$y.\bar{z}$
$\bar{w}.\bar{x}$	1			x
$\bar{w}.x$		1	x	
$w.\bar{x}$		x	1	
$w.x$	x			1

Tabla 13.26

	$\bar{y}.z$	$\bar{y}.\bar{z}$	$y.z$	$y.\bar{z}$
$\bar{w}.\bar{x}$		1	x	
$\bar{w}.x$	1			X
$w.\bar{x}$	x			1
$w.x$		x	1	

Tabla 13.27

	$\bar{y}.z$	$\bar{y}.\bar{z}$	$y.z$	$y.\bar{z}$
$\bar{w}.\bar{x}$				
$\bar{w}.x$	1	x	1	x
$w.\bar{x}$				
$w.x$	x	1	x	1

Tabla 13.28

3.1 El circuito óptimo de la tabla 13.26 es:

- A) una XNOR B) una AND C) una XOR D) una NOR E) una NAND

3.2 El circuito óptimo de la tabla 13.27 es:

- A) una XNOR B) una AND C) una XOR D) una NOR E) una NAND

3.3 El circuito óptimo de la tabla 13.28 es:

- A) una XNOR B) una AND C) una XOR D) una NOR E) una NAND

4. La compuerta del circuito digital óptimo de la expresión $Q = \overline{(x.y + x.y)}(\overline{x.y} + x)$ corresponde a:

- A) NOT B) OR C) XNOR D) AND E) XOR

5. La expresión optima correspondiente a $\overline{\overline{x + \overline{y}.y.x}}$ es:

- A) 1 NOR y 2 NOT B) 1 NAND y 2 NOT C) 1 XNOR D) 1 OR y 2 NOT

TALLER 13

1. Dada la función lógica de la tabla 13.29 diseñe un circuito digital con seis entradas (A0, A1, A2, A3, C1, C0) y una salidas (Q), teniendo en cuenta que 0: low y 1: high que utilice solamente compuertas lógicas.

C1	C0	Q
0	0	A0
0	1	A1
1	0	A2
1	1	A3

Tabla 13.29

2. Dada la función lógica de la tabla 13.30 diseñe un circuito digital con cuatro entradas (D, C, B, A) y una salidas (Q1, Q2), teniendo en cuenta que 0: low y 1: high que utilice solamente compuertas lógicas.

D	C	B	A	Q ₁	Q ₀
1	1	1	0	1	1
1	1	0	x	1	0
1	0	x	x	0	1
0	x	x	x	0	0

Tabla 13.30

3. Determine la expresión booleana óptima del ejemplo 13.12 utilizando la técnica NOR y dibuje su circuito correspondiente.

4. Simplifique las expresiones booleanas dadas desde 4.1 hasta 4.6, utilizando mapas de karnaugh y leyes del álgebra booleana. Conviértalas en expresiones óptimas. Dibuje el circuito digital óptimo con compuertas básicas. Dibújelo también utilizando solamente compuertas NAND y compuertas NOR de dos entradas. Determine la cantidad de retrasos de propagación del circuito dado que en cada compuerta produce un retardo de 10 ns.

Para tal fin tres técnicos se ponen en esta tarea y presentan la siguiente función booleana:

$$\text{Técnico 1: } T1 = x.\bar{y}.\bar{z} + x.y.z + x.y.\bar{z}$$

$$\text{Técnico 2: } T2 = \overline{\bar{x} + (\bar{z} + y)}$$

$$\text{Técnico 3: } T3 = x.(\bar{z} + y)$$

Analice cuál o cuáles cumplen con los requerimientos del propietario. Diseñe el circuito digital óptimo recomendable para el chip y explique brevemente el por qué.

9. El propietario de electrónicas NORNAND & Cia. ha pedido a sus técnicos diseñar un chip para poner en funcionamiento una alarma a partir de tres interruptores. La alarma se activa cerrando simultáneamente dos interruptores; en los demás casos quedará desactivada. Para tal fin, los técnicos se ponen en esta tarea y presentan las siguientes funciones booleanas:

$$A1 = \overline{(\bar{x}.y.z)(x.\bar{y}.z)(x.y.\bar{z})(\bar{x}.\bar{y}.\bar{z})}$$

$$A2 = \bar{x}.y.z + x.\bar{y}.z + x.y.\bar{z} + \bar{x}.\bar{y}.\bar{z}$$

¿Ambas funciones son correctas para las exigencias del propietario? Analícelas. Si ambas funciones son correctas, técnicamente ¿cuál recomienda? ¿Por qué?

10. Se quiere diseñar un dispositivo con tres interruptores (x, y, z): uno para control de alarma de un vehículo, otro para asegurar las puertas y un tercero para activar el mecanismo del elevavidrios. El sistema funciona así:

- Si se cierra el primer interruptor (x) se activa la alarma.
- Si se cierra el segundo interruptor (y) se activan los seguros de las puertas.
- Si se cierran simultáneamente los dos primeros interruptores (x, y), el sistema no funcionará (ni alarma ni elevavidrios ni los seguros).
- Si se cierra el tercer interruptor (z) se activa el elevavidrios (cerrando los vidrios si están abiertos).

Diseñe el circuito digital óptimo adecuado técnicamente para el sistema planteado.

12. Seleccione la respuesta correcta según los siguientes casos, que dé respuestas a las situaciones planteadas en los numerales 12.1 a 12.3:

- A) Si la afirmación y la razón son VERDADERAS y la razón es una explicación CORRECTA de la afirmación
- B) Si la afirmación y la razón son VERDADERAS, pero la razón NO es una explicación CORRECTA de la afirmación
- C) Si la afirmación es VERDADERA, pero la razón es una proposición FALSA
- D) Si la afirmación es FALSA, pero la razón es una proposición VERDADERA
- E) Si tanto la afirmación como la razón son proposiciones FALSAS

Los mapas de Karnaugh se usan como herramienta para simplificar expresiones booleanas expresadas en las formas FNC o FND o circuitos digitales complejos, simplificándolos de manera óptima sin la utilización de leyes del álgebra booleana **PORQUE** los mapas de Karnaugh permiten expresar circuitos digitales con menor valor económico, que producen menos retardos y son más confiables.

Respuesta: ____

13. Determine la cantidad de compuertas básicas que se ahorra entre el circuito original de la expresión $Q = (\bar{x}.y + x.\bar{y})(x.\bar{y} + x)$ y el correspondiente circuito digital óptimo dibujado con compuertas NOR.

14. La Compañía Bell Comunicaciones (CBC) quiere diseñar un artefacto luminoso que tenga 4 interruptores y puede ser **ENCENDIDO** como se indica en la tabla 13.31. Si usted fuese el técnico,

14.1 ¿Cuál función booleana completa presentaría?

14.2 ¿Cuál sería el circuito óptimo que cumpla dicha función utilizando solamente compuertas NAND de 2 entradas?

15. Determine la función lógica de los circuitos de las figuras 13.26 y 13.27:

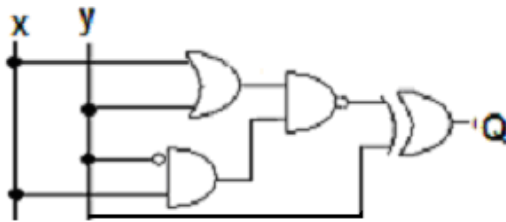


Figura 13.26: problema 15

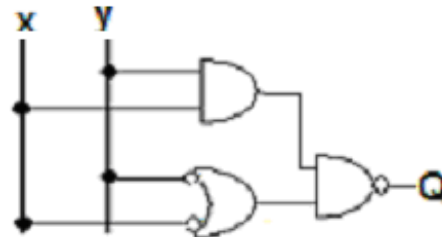
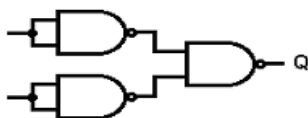


Figura 13.27: problema 15

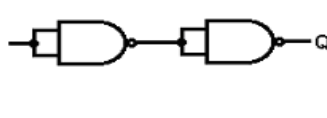
w	x	y	z	f
0	0	0	0	0
0	0	0	1	X
0	0	1	0	0
0	0	1	1	X
0	1	0	0	X
0	1	0	1	0
0	1	1	0	1
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	1
1	1	0	0	1
1	1	0	1	0
1	1	1	0	X
1	1	1	1	0

Tabla 13.31

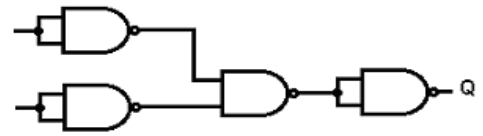
16. Los circuitos lógicos combinacionales de la figura 13.28 tienen su equivalencia con las compuertas básicas. Determine a cuál compuerta corresponde cada una.



a)



b)



c)

Figura 13.28: problema 16