

UNIDAD 2

COMPUERTAS LOGICAS

TABLA DE CONTENIDO.

2.1	Qué es Electrónica Digital.	30
2.2	Álgebra de booleana.	31
2.3	Operación booleana y compuertas lógicas.	31
2.4	Inversión o negación (complemento).	32
2.5	Suma booleana (compuerta OR).	33
2.6	Multiplicación booleana (compuerta AND).	34
2.7	Compuerta OR-EX o XOR.	36
2.8	Compuertas lógicas Combinadas.	36
2.9	Compuerta NOR-EX.	36
2.10	Buffer (YES)	37
2.11	Operaciones booleanas.	38
2.12	Teoremas boléanos.	40
2.13	Teoremas de DeMorgan.	41
2.14	Símbolos lógicos estándares IEEE/ANSI.	43
	Autoevaluación final de la Unidad.	46

OBJETIVOS ESPECÍFICOS.

- ✓ Implementar y simplificar circuitos lógicos empleando diferentes leyes del álgebra de Boole y teoremas de Morgan.

- ✓ Orientar al Estudiante a emplear correctamente la aplicación de compuertas lógicas de un circuito representado por una expresión booleana.

- ✓ Escribir la expresión booleana para las compuertas lógicas y las combinaciones de compuertas lógicas.

INTRODUCCIÓN.

esta unidad esta orientada al estudio de las compuertas lógicas, que son de gran utilidad en el diseño de los circuitos lógicos. En el estudio de las compuertas lógicas será analizada su operación lógica mediante el álgebra booleana.

También veremos como se combinan las compuertas lógicas para producir circuitos lógicos que serán analizados mediante álgebra booleana.

El álgebra booleana se utiliza para expresar los efectos que los diversos circuitos digitales ejercen sobre las entradas lógicas y para manipular variables lógicas con objeto de determinar el mejor método de ejecución de cierta función de un circuito.

Ya que solo pueden haber dos valores, el álgebra booleana es muy sencilla de entender y manejar. De hecho en el álgebra booleana solo existen tres operaciones básicas: OR, AND Y NOT.

Usaremos el álgebra booleana primero para describir y analizar estas compuertas lógicas básicas y mas tarde para analizar combinaciones de compuertas lógicas conectadas como circuitos lógicos.

EVALUACIÓN DIAGNOSTICA.

1. ¿Qué se entiende por compuerta lógica?.
2. ¿qué se entiende por álgebra booleana?.
3. definir la diferencia que existe entre una compuerta AND y OR.
4. enumerar los teoremas de DeMorgan.
5. dibujar una compuerta AND, OR, NOR Y NAND de cuatro entradas y hallar la tabla de verdad de cada una.

2.1 Qué es Electrónica Digital

Es una ciencia que estudia las señales eléctricas, pero en este caso son señales discretas, es decir, están bien identificadas, razón por la cual a un determinado nivel de tensión se lo llama estado alto (**High**) o Uno lógico; y a otro, estado bajo (**Low**) o Cero lógico.

Si suponemos que las señales eléctricas con que trabaja un sistema digital son 0V y 5V. Donde 5V será el estado alto o uno lógico, pero habrá que tener en cuenta que existe la **Lógica Positiva** y la **Lógica Negativa**, veamos cada una de ellas.

Lógica Positiva

En esta notación al 1 lógico le corresponde el nivel más alto de tensión (positivo, si quieres llamarlo así) y al 0 lógico el nivel mas bajo (que bien podría ser negativo), pero que ocurre cuando la señal no está bien definida...?. Entonces habrá que conocer cuales son los límites para cada tipo de señal (conocido como tensión de histéresis), en la figura 2.1 se puede ver con mayor claridad cada estado lógico y su nivel de tensión.

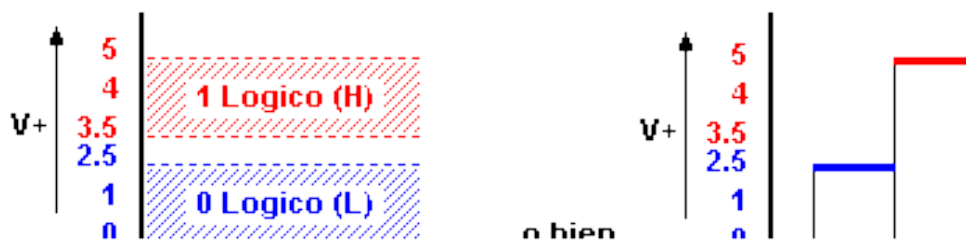


figura 2.1

Lógica Negativa

Aquí ocurre todo lo contrario, es decir, se representa al estado "**1**" con los niveles más bajos de tensión y al "**0**" con los niveles más altos ver figura 2.2.

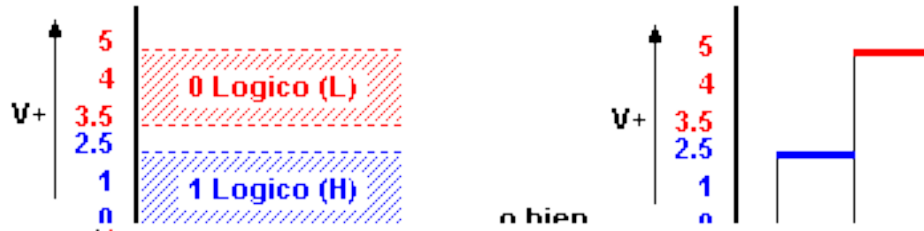


figura 2.2

Por lo general se suele trabajar con lógica positiva, y así lo haremos en este modulo, la forma más sencilla de representar estos estados es como se puede ver en la figura 2.3.

figura 2.3

De ahora en más ya sabrás a que nos referimos con estados lógicos **1** y **0**.

2.2 Álgebra de Boole

El álgebra booleana es la teoría matemática que se aplica en la lógica combinatoria. Las variables booleanas son símbolos utilizados para representar magnitudes lógicas y pueden tener sólo dos valores posibles: 1 (valor alto) ó 0 (valor bajo).

2.3 Operaciones Booleanas y Compuertas Básicas

Las operaciones booleanas son posibles a través de los operadores binarios negación, suma y multiplicación, es decir que estos combinan dos o más variables para conformar funciones lógicas. Una compuerta es un circuito útil para realizar las operaciones anteriormente mencionadas.

Compuertas Lógicas

Las compuertas lógicas son dispositivos que operan con aquellos estados lógicos que funcionan igual que una calculadora, de un lado

ingresas los datos, ésta realiza una operación, y finalmente, te muestra el resultado. Ver figura 2.4



Cada una de las compuertas lógicas se las representa mediante un **Símbolo**, y la operación que realiza (**Operación lógica**) le corresponde una tabla, llamada **Tabla de Verdad**.

2.4 Inversión o negación (complemento) COMPUERTA NOT

Esta operación se indica con una barra sobre la variable o por medio de un apóstrofe en el lado superior derecho de la variable, en esta unidad emplearemos esta última notación (al menos que se indique lo contrario)=. El apóstrofe (') es un operador algebraico que invierte el valor de una variable, es decir, si X denota la señal de entrada de un inversor, entonces X' representa el complemento de tal señal.

Ejemplo

Sí $X = 0$ entonces $X' = 1$.

En la tabla de verdad 2.1. se muestra el resultado de la inversión lógica.

Ecuación	Entrada A	Salida B
$B=A'$	0	1
	1	0

Tabla 2.1. Tabla de verdad del inversor

El símbolo lógico de la negación booleana se representa en la figura 2.5.

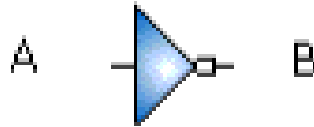


Figura 2.5. Inversor.

2.5 Suma booleana.

La representación matemática de una suma booleana de dos variables se hace por medio un signo más entre las dos variables.

Ejemplo

La suma booleana de las variables A y B se enuncia de la siguiente forma,

$$X = A + B$$

La suma booleana es 1 si alguna de las variables lógicas de la suma es 1 y es 0 cuando todas las variables son 0. Esta operación se asimila a la conexión paralela de contactos.

La tabla de verdad de la suma se muestra en la tabla 2.2.

Entrada A	Entrada B	Salida X
0	0	0
0	1	1
1	0	1
1	1	1

Tabla 2.2.Tabla de Verdad de la función OR

En circuitos digitales, el equivalente de la suma booleana es la operación OR y su símbolo lógico se representa en la figura 2.6.



Figura 2.6. Símbolo lógico para la compuerta OR.

Con la correspondiente ecuación $X = A + B$.

El inverso de la función OR es la función NOR. La tabla de verdad se muestra en la tabla 2.3.

Entrada A	Entrada B	Salida X
0	0	1
0	1	0
1	0	0
1	1	0

Tabla 2.3. Tabla de verdad de la función NOR

El símbolo lógico de la compuerta NOR se representa en la figura 2.7.

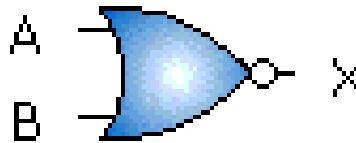


Figura 2.7. Símbolo lógico para la compuerta NOR

Con la correspondiente ecuación $X = (A+B)'$

La suma booleana difiere de la suma binaria cuando se suman dos unos. En la suma booleana no existe acarreo.

2.6 Multiplicación booleana

La representación matemática de una multiplicación booleana de dos variables se hace por medio un signo punto (\cdot) entre las dos variables.

La multiplicación booleana de las variables A y B se enuncia de la siguiente forma,

$$X = A \cdot B$$

La multiplicación booleana es 1 si todas las variables lógicas son 1, pero si alguna es 0, el resultado es 0. La multiplicación booleana se asimila a la conexión serie de contactos.

La tabla de verdad de la multiplicación booleana se muestra en la tabla 2.4.

Entrada A	Entrada B	Salida X
0	0	0
0	1	0
1	0	0
1	1	1

Tabla 2.4. Tabla de verdad de la función AND

En circuitos digitales, el equivalente de la multiplicación booleana es la operación AND y su símbolo se representa en la figura 2.8.

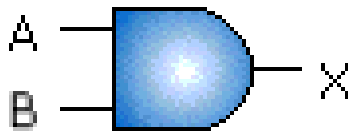


Figura 2.8. Símbolo lógico de la función AND

con la correspondiente ecuación $X = A \cdot B$

El inverso de la función AND es la función NAND. La tabla de verdad se muestra la tabla 2.5.

Entrada A	Entrada B	Salida X
0	0	1
0	1	1
1	0	1
1	1	0

Tabla 2.5. Tabla de verdad de la función NAND

El símbolo lógico de la compuerta NAND se representa en la figura 2.9.

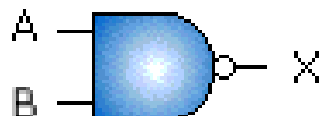


figura 2.9. Símbolo lógico de la función NAND


Con la correspondiente ecuación $X = (A \cdot B)'$

2.7 Compuerta OR-EX o XOR

Es OR EXCLUSIVA en este caso con dos entradas y lo que hará con ellas será una suma lógica entre **a** por **b invertida** y **a invertida** por **b**. Ver tabla 2.6

Al ser O Exclusiva su salida será 1 si una y sólo una de sus entradas es 1

Tabla 2.6. Tabla de verdad de la función XOR



$s = a \cdot \bar{b} + \bar{a} \cdot b$

a	b	s
0	0	0
0	1	1
1	0	1
1	1	0

Estas serían básicamente las compuertas mas sencillas.

2.8 Compuertas Lógicas Combinadas

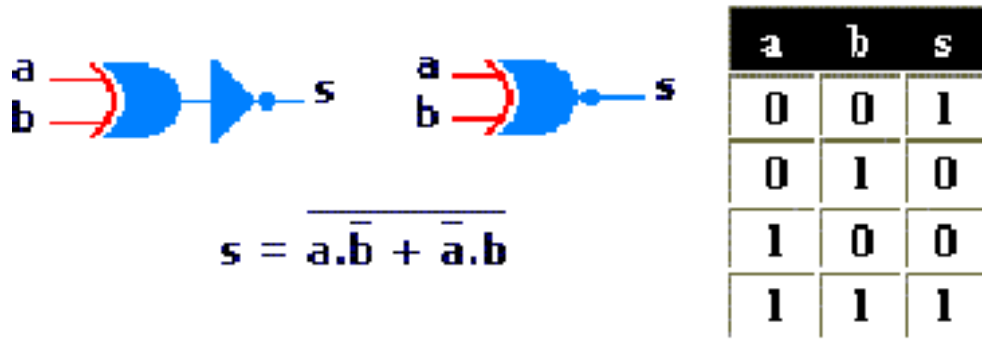
Al agregar una compuerta NOT a cada una de las compuertas anteriores los resultados de sus respectivas tablas de verdad se invierten, y dan origen a tres nuevas compuertas llamadas **NAND**, **NOR** y **NOR-EX**... Veamos ahora como son y cual es el símbolo que las representa...

2.9 Compuerta NOR-EX

Es simplemente la inversión de la compuerta OR-EX, los resultados se pueden apreciar en la tabla 2.7 tabla de verdad, que bien podrías

compararla con la anterior y notar la diferencia, el símbolo que la representa lo tienes en el siguiente gráfico.

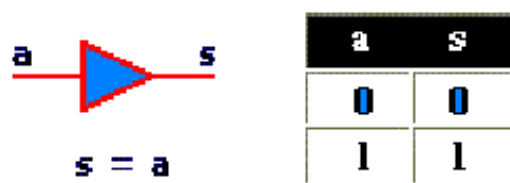
Tabla 2.7. Tabla de verdad de la función NOR



2.10 Buffer's (YES)

Su finalidad es amplificar un poco la señal (o refrescarla si se puede decir). Como puedes ver en el siguiente gráfico la señal de salida es la misma que de entrada. Ver tabla 2.8

Tabla 2.8. Tabla de verdad de la función YES



Hasta aquí de teoría, nos interesa más saber como se hacen evidente estos estados en la práctica, y en qué circuitos integrados se las puede encontrar y más adelante veremos unas cuantas leyes que se pueden aplicar a estas compuertas para obtener los resultados que deseas.

"Deténgase a investigar"

Analizar y presentar en CIPA la siguiente consulta por escrito.

- 1. Dibujar cada una de las compuertas vistas hasta el momento con mas de dos entradas (tres entradas) y hallar la tabla de la verdad correspondiente.***
- 2. investigar la aplicación (practica) y referencia del CI (datos de los integrados) de cada compuerta lógica.***

2.11 Operaciones booleanas

Las operaciones booleanas están regidas por tres leyes similares a las del álgebra convencional. Estas incluyen las leyes conmutativas de la suma y la multiplicación y la ley distributiva.

Leyes conmutativas en dos variables

1. Ley conmutativa de la suma se enuncia como sigue

$$X + Y = Y + X$$

En aplicación a los circuitos digitales, podríamos decir que no importa el orden de conexión de las entradas a una compuerta OR.

2. Ley conmutativa de la multiplicación

$$X \cdot Y = Y \cdot X$$

En aplicación a los circuitos digitales, podríamos decir que no importa el orden de conexión de las entradas a una compuerta AND.

Leyes asociativas en tres variables

3. Ley asociativa de la adición, se escribe en forma algebraica de la siguiente forma

$$A + (B + C) = (A + B) + C$$

En la figura 2.10. se muestra la aplicación de la propiedad a las compuertas OR,

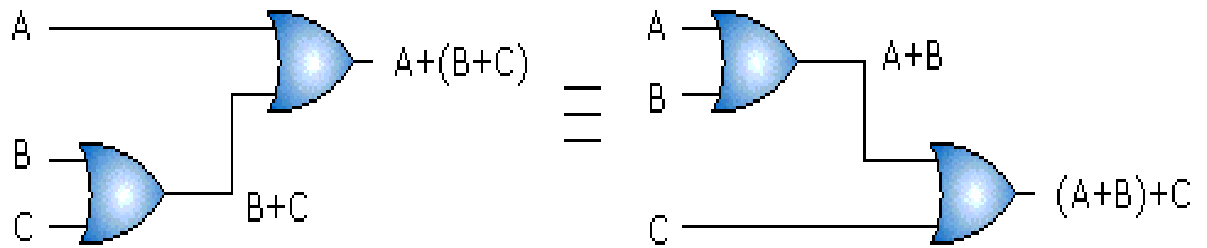


Figura 2.10. Ley asociativa de la adición

4. Ley asociativa de la multiplicación

$$A \cdot (B \cdot C) = (A \cdot B) \cdot C$$

En la figura 2.11. se muestra la aplicación de la propiedad a las compuertas AND,

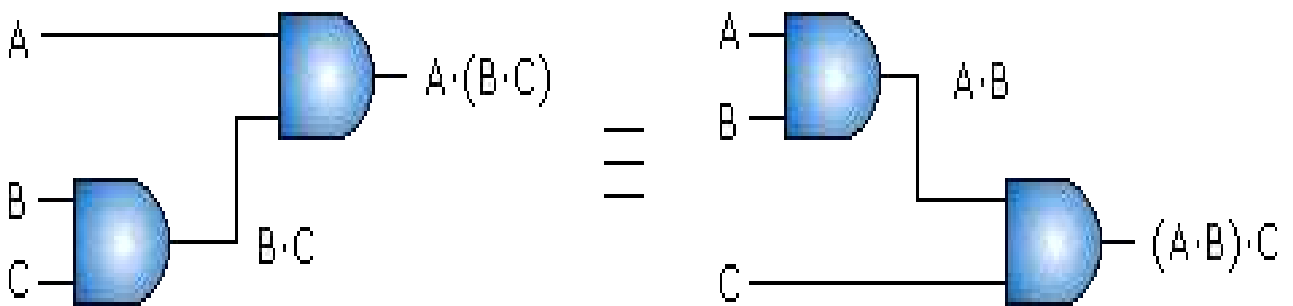


Figura 2.11. Ley asociativa de la multiplicación

Ley distributiva para tres variables

En el álgebra de Boole, la multiplicación lógica se distribuye sobre la suma lógica,

$$A \cdot (B + C) = A \cdot B + A \cdot C$$

En la figura 2.12 se muestra la aplicación de la propiedad a las compuertas AND y OR,

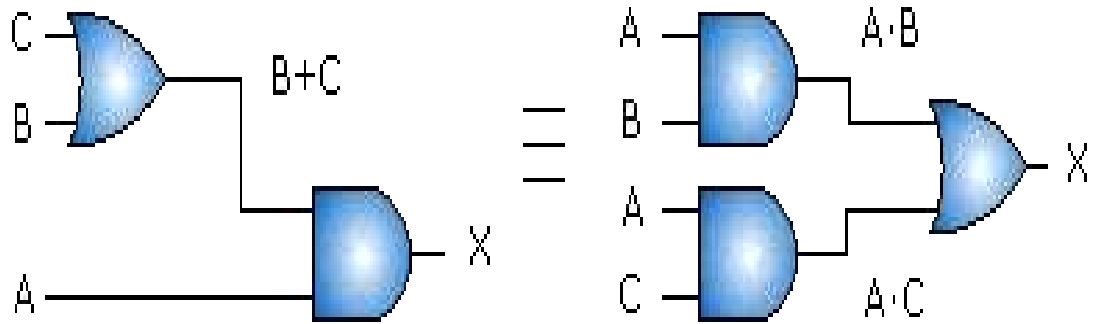


Figura 2.12. Ley distributiva para tres variables

2.12 Teoremas Booleanos

Los teoremas booleanos son enunciados siempre verdaderos, lo que permite la manipulación de expresiones algebraicas, facilitando el análisis ó síntesis de los circuitos digitales. Los teoremas booleanos son los siguientes:

1. $X + 0 = X$
2. $X + 1 = 1$
3. $X \cdot 0 = 0$
4. $X \cdot 1 = X$
5. $(X')' = X$
6. $X + X = X$
7. $X \cdot X = X$
8. $X + X' = 1$
9. $X \cdot X' = 0$
10. $X + XY = X$
11. $X + X' \cdot Y = X + Y$
12. $X \cdot Y + X \cdot Y' = X$ (Teorema de combinación)
13. $(X + Y)(X + Y') = X + X \cdot Y' + X \cdot Y = X$
14. $X \cdot Y + X \cdot Z + Y \cdot Z' = XZ + Y \cdot Z'$ (Consenso)

El teorema 12 se conoce como la ley distributiva para tres variables.

Demostración teorema 12:

$$X \cdot Y + X \cdot Y' = X$$

Utilizando la ley distributiva para tres variables

$$X \cdot Y + X \cdot Y' = X \cdot (Y + Y')$$

Aplicando el teorema 8 se tiene,

$$X \cdot Y + X \cdot Y' = X \cdot 1$$

Dando como resultado,

$$X \cdot Y + X \cdot Y' = X$$

Esta expresión indica que la suma de dos productos canónicos adyacentes, es decir que difieren en una sola de las variables, se reduce al producto de los demás términos suprimiéndose dicha variable. El teorema 13 es otro caso del teorema de combinación. Los teoremas 12 y 13 se utilizarán en las lecciones siguientes de forma sistemática para sintetizar circuitos lógicos con los métodos de mapas de Karnaugh.

Detengámonos y analicemos un poco

Análisis individual.

Tomar dos teoremas y demostrarlos para luego ser socializado en CIPA.

2.13 Teoremas de DeMorgan

Los teoremas de DeMorgan demuestran la equivalencia entre las puertas NAND y negativa - OR, y las puertas NOR y negativa - AND.

El complemento de la suma de variables es igual al producto de los complementos de las variables.

$$(X_1 + X_2 + \dots + X_n)' = X_1' \cdot X_2' \cdot \dots \cdot X_n'$$

En el caso de dos variables se tiene,

$$(X + Y)' = X' \cdot Y'$$

El circuito equivalente a la ecuación anterior se muestra en la figura 2.13.

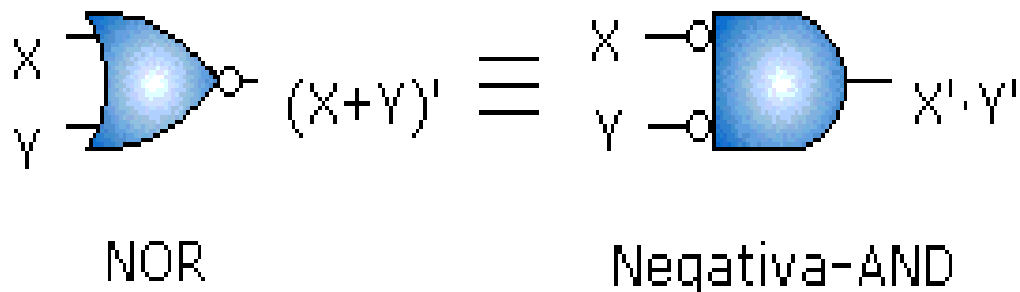


Figura 2.13. Símbolo lógico para la compuerta NOR.

Ejemplo

Obtener una compuerta OR utilizando compuertas NAND.

$$Y = (A + B) = [(A + B)']' = (A' \cdot B')'$$

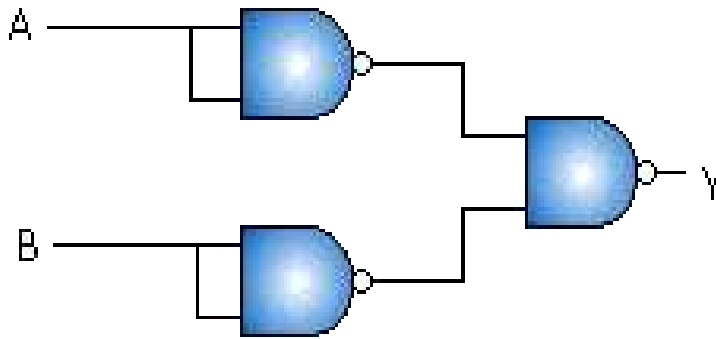


Figura 2.14. Compuerta OR utilizando compuertas NAND

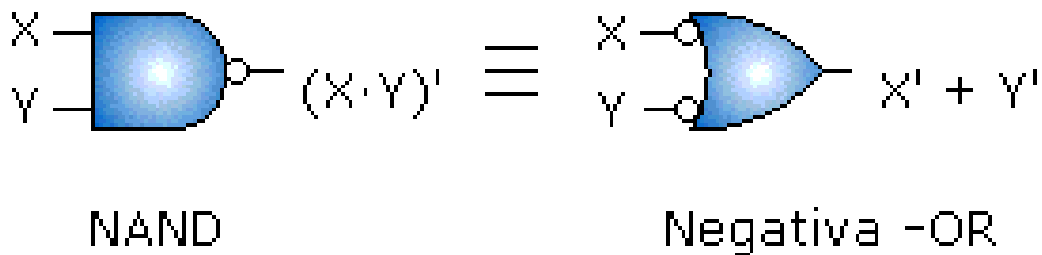
El complemento del producto de variables es igual a la suma de los complementos de las variables.

$$(X_1 \cdot X_2 \cdot \dots \cdot X_n)' = X_1' + X_2' + \dots + X_n'$$

En el caso de dos variables se tiene,

$$(X \cdot Y)' = X' + Y'$$

El circuito equivalente en dos variables a la ecuación se muestra en la figura 2.15.



NAND

Negativa -OR

Figura 2.15. Símbolo lógico para la compuerta NOR.

Ejemplo

Obtener una compuerta AND utilizando compuertas NOR.

$$Y = A \cdot B = [(A \cdot B)']' = (A' + B)'$$

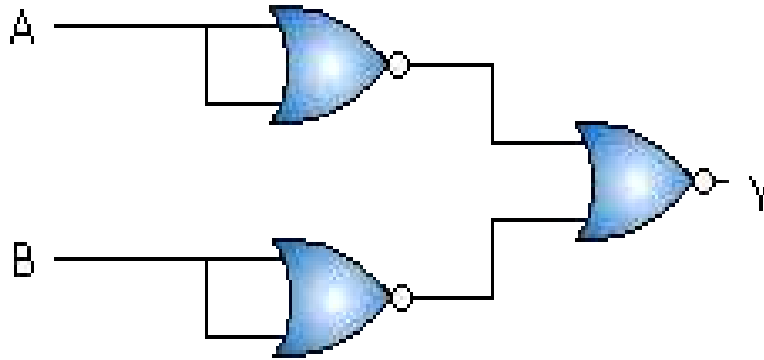


Figura 2.16. Circuito lógico para la compuerta AND.

2.14 Símbolos lógicos estándares IEEE/ANSI.

Los símbolos lógicos empleados en esta unidad son los símbolos estándar que se han utilizado en la industria digital por muchos años. Estos símbolos son bastante adecuados para las compuertas lógicas básicas ya que cada símbolo tiene una forma de distinguirse de los demás, y la entrada de la misma compuerta tiene la misma función, sin embargo no proporcionan información suficiente para dispositivos Lógicos mas complejos que tienen varias entradas y salidas y modos de operación.

En 1984 se introdujo un nuevo conjunto de símbolos estándares que proporcionan información mas útil sobre estos complejos dispositivos.

A continuación se observaran en la figura 2. algunos ejemplos de estos.

Simboos lógico estandar (a) tradicionales.

(b) rectangulares

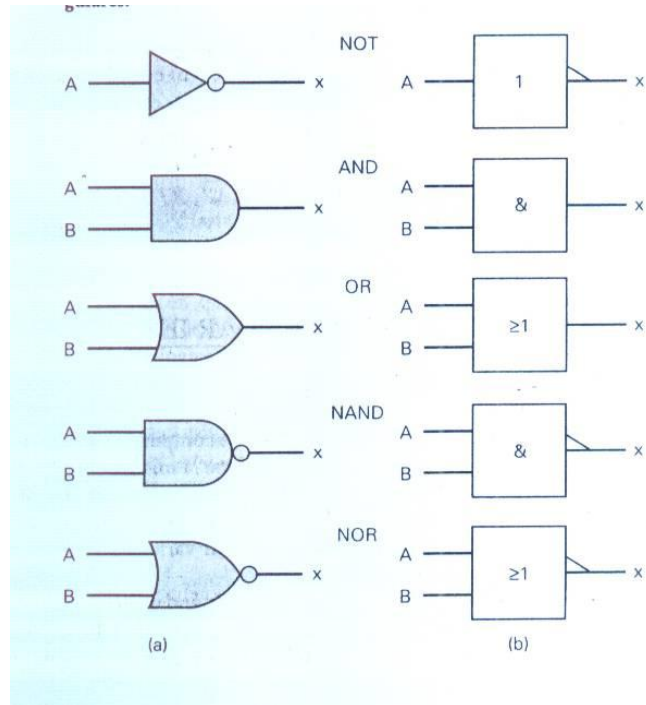
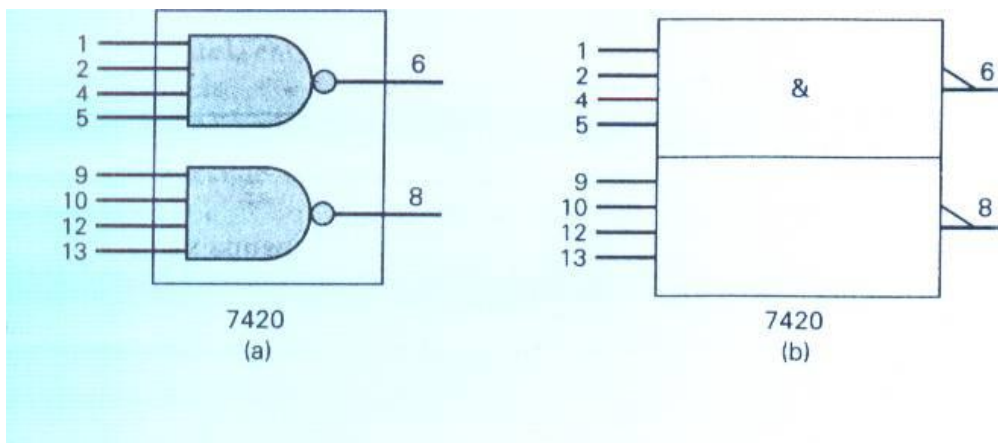


Figura 2.17. Símbolos lógicos estándares IEEE/ANSI.

Figura 2.18.

CI NAND 7420 con compuertas cada una con cuatro entradas: (a) símbolo tradicional (b) símbolo rectangular.



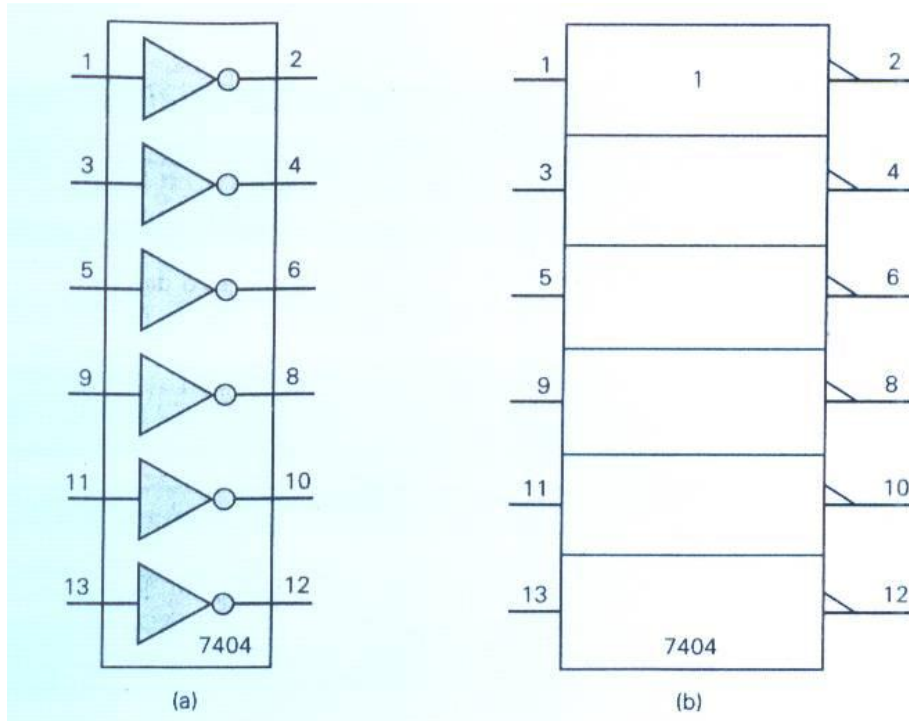
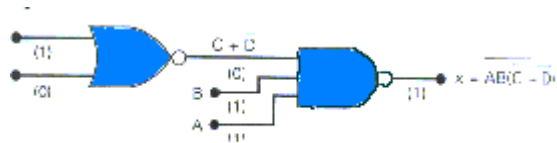


Figura 2.19.
 CI NOT 7404: (a) símbolo tradicional (b) símbolo rectangular.

Autoevaluación final de la Unidad.

2.1 Determine la tabla de la verdad completa para el circuito de la figura.



2.2 Complete cada expresión.

(a) $A+1 =$

(b) $A.A =$

(c) $C+C =$

(d) $X.0 =$

(e) $D.1 =$

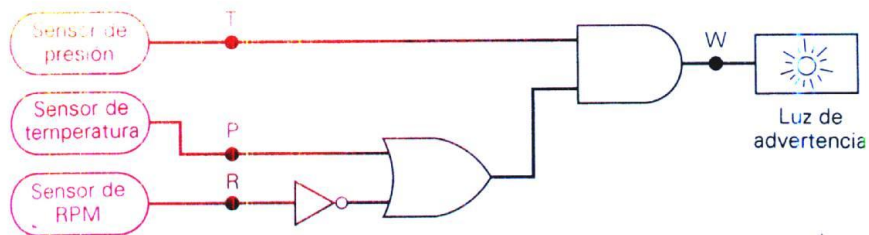
(f) $D+0 =$

2.3 demuestre los teoremas de DeMorgan ensayando todos los casos posibles.

2.4 Un avión del aeropuerto camilo daza dela ciudad de Cúcuta emplea un sistema para vigilar las rpm (revoluciones por minuto), presión y temperatura de motores usando sensores que operan como sigue:

- salida del sensor de RPM = 0 sólo cuando la velocidad es menor que 4800 rpm.
- Salida del sensor de P = 0 solo cuando la presión es menor que 220 psi.
- Salida del sensor T = 0 solo cuando la temperatura es menor que 200 grados F.

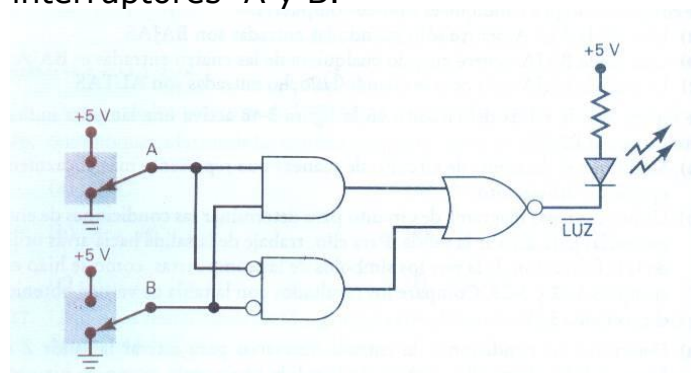
2.5 La siguiente figura muestra el circuito lógico que controla la luz de advertencia de la cabina para ciertas combinaciones de condiciones del motor.



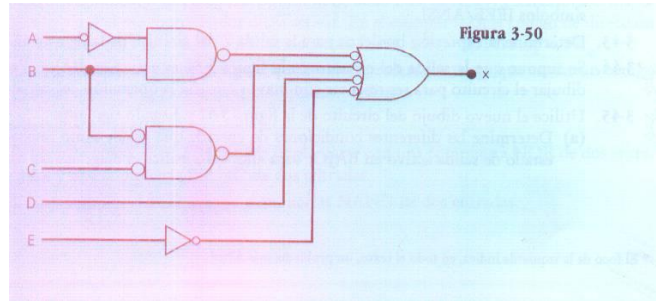
- (a) Determine que condiciones del motor advertirán al piloto.
 (b) Cambie este circuito a otro que solo utilice compuertas NAND.

2.6 En la figura siguiente se muestra una aplicación de las compuertas lógicas que simula un interruptor de dos tiros, similar a los empleados en los hogares para encender o apagar la luz con dos interruptores diferentes. En este caso la luz proviene de un LED, que conduce cuando la entrada de la compuerta NOR es baja. Note que esta salida esta marcada como LUZ .

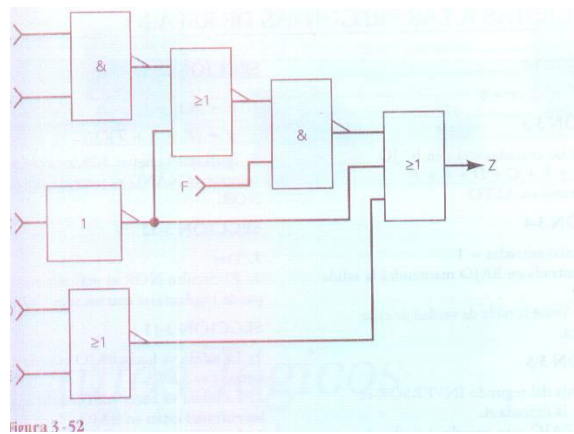
Determine las condiciones necesarias para encender el LED . luego verifique que el circuito opera como un interruptor de dos tiros empleando los interruptores A y B.



2.7 Dibuje el circuito de la siguiente figura utilizando los símbolo IEEE/ANSI.



2.8 según la siguiente figura:



(a) determine las diferentes condiciones de entrada que darán como resultado un estado de salida en Bajo. Para ello, solo utilice el diagrama del circuito, sin escribir la expresión Z y sin generar la tabla de la verdad completa. Los resultados deben ser.

A	B	C	D	E	F
1	1	1	1	1	1
1	1	1	1	0	1
1	1	1	0	1	1

2.9 Implane $y = ABCD$ usando compuertas NAND.