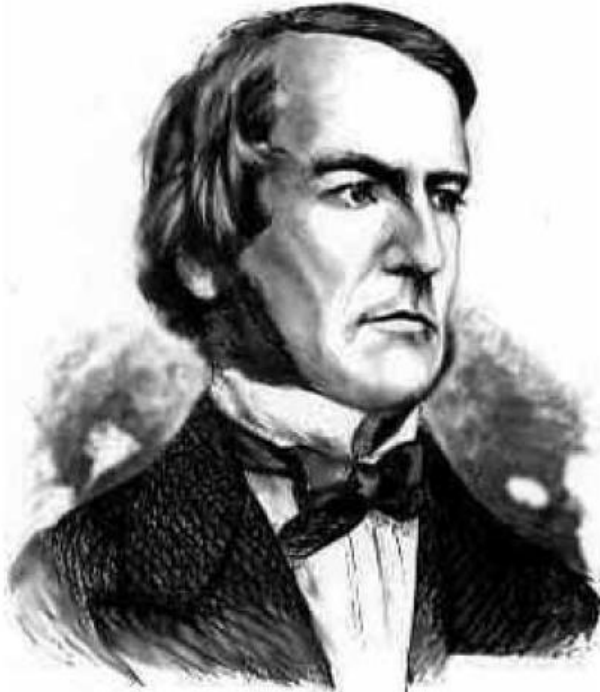


Tema 03

ANÁLISIS DE CIRCUITOS CON PUERTAS LÓGICAS



Boole (1815-1864)

Avelino Rodríguez Núñez



1. ÁLGEBRA DE BOOLE

En 1854 publicó *"An Investigation of the Laws of Thought"*, en el que desarrollaba un sistema de reglas que le permitían expresar, manipular y simplificar problemas lógicos y filosóficos cuyos argumentos admiten dos estados (verdadero o falso) por procedimientos matemáticos. Se podría decir que es el padre de las operaciones lógicas y gracias a su álgebra hoy en día es posible manipular operaciones lógicas.

Álgebra convencional :

- Opera con relaciones aritméticas: (" $+$ ", " $*$ ", " $-$ ", ...)
- Utiliza variables (x, y, z, \dots) para representar cantidades numéricas.

Álgebra de Boole:

- Opera con relaciones lógicas
- Utiliza variables binarias que sólo pueden tomar dos valores : verdadero ó falso. No expresan cantidades numéricas sino estados de esas variables.

1. ÁLGEBRA DE BOOLE

1.1. Tabla de verdad

Es una forma de describir el funcionamiento de un sistema digital. En ella se representa el estado de las entradas (sensores) y salidas (actuadores) para todas las posibles combinaciones que se den en el sistema.

Ejemplo: Diseñar un sistema de aviso de peligro en sillas colgantes en las que no debe subir una sola persona por problemas de desequilibrio.

Para detectar personas se ha instalado dos pulsadores (**a y b**) y una luz (**s**) que **se encenderá** en caso de emergencia.

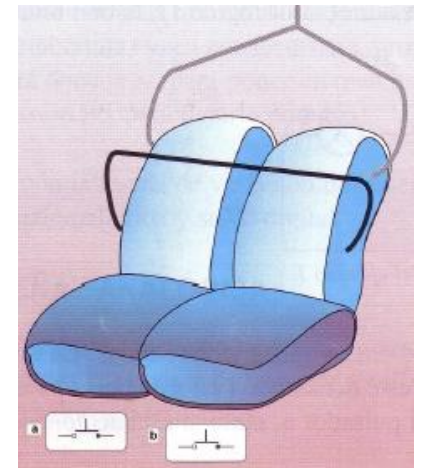


TABLA DE VERDAD

| a | b | s |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

→ No hay ninguna persona en la silla.

→ Hay sólo una persona sobre el pulsador **b**. **Alarma**

→ Hay sólo una persona sobre el pulsador **a**. **Alarma**

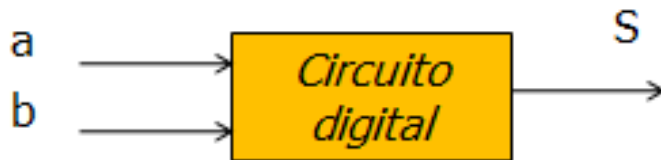
→ Se sienta dos personas.

1. ÁLGEBRA DE BOOLE

1.2. Función lógica

Es una expresión matemática que relaciona las salidas de un sistema con las entradas mediante sumas, productos y negaciones.

A partir de ella se puede diseñar el **circuito lógico**.



$a, b \rightarrow$ variables binarias que pueden tomar los valores "0" o "1"

| a | b | s |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

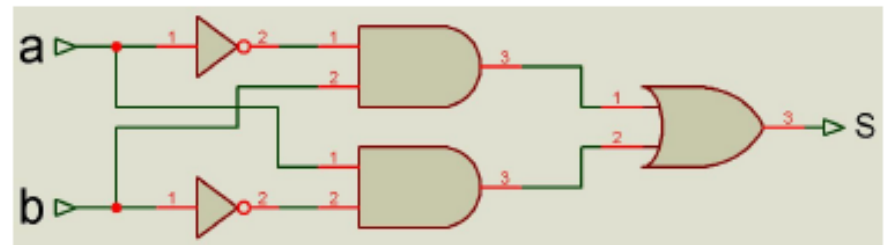
Términos

$\bar{a}.b$

$a.\bar{b}$

Función lógica $\rightarrow S = \bar{a}.b + a.\bar{b}$

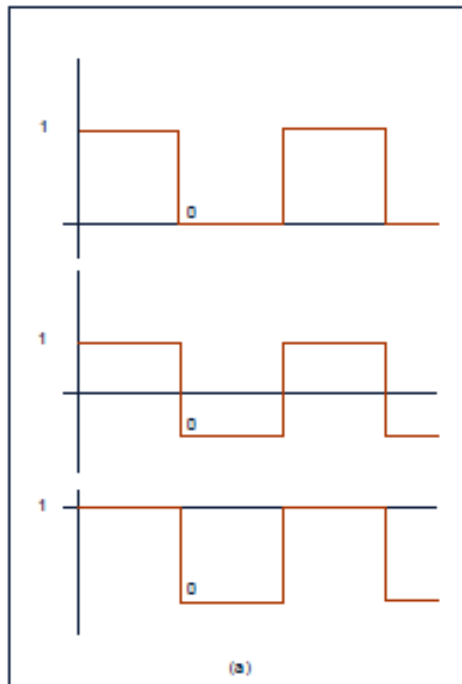
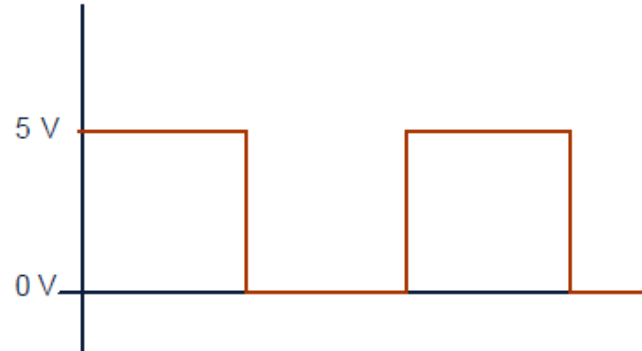
Circuito lógico



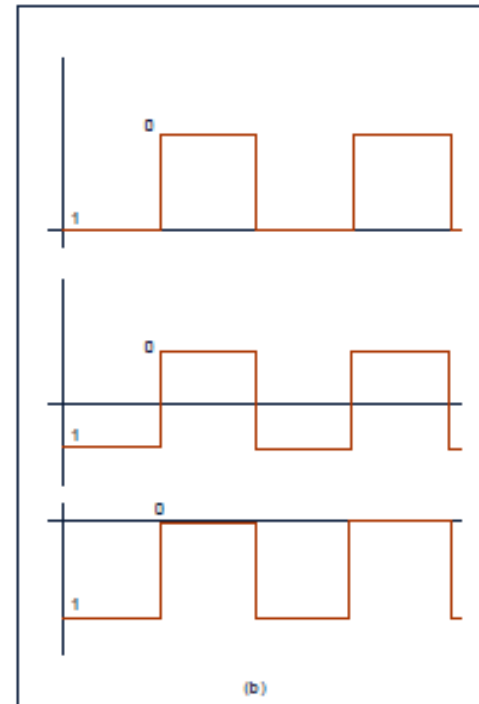
1. ÁLGEBRA DE BOOLE

1.3. Niveles lógicos

Una variable lógica puede tomar dos posibles valores "0" o "1"



Lógica Positiva



Lógica Negativa



2. POSTULADOS, PROPIEDADES Y TEOREMAS

2.1. Operaciones lógicas

- **SUMA LÓGICA (+)**

$$S = a + b$$

| b | a | S |
|----------|----------|----------|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

- **PRODUCTO LÓGICO (.)**

$$S = a \cdot b$$

| b | a | S |
|----------|----------|----------|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

- **INVERSIÓN.** (Negación, complementación)

$$S = \bar{a}$$

| a | S |
|----------|----------|
| 0 | 1 |
| 1 | 0 |



2. POSTULADOS, PROPIEDADES Y TEOREMAS

2.2. Postulados

$$\mathbf{a + 1 = 1 \rightarrow}$$

- $a = 0 \rightarrow 0 + 1 = 1$
- $a = 1 \rightarrow 1 + 1 = 1$

$$\mathbf{a + 0 = a \rightarrow}$$

- $a = 0 \rightarrow 0 + 0 = 0$
- $a = 1 \rightarrow 1 + 0 = 1$

$$\mathbf{a \cdot 1 = a \rightarrow}$$

- $a = 0 \rightarrow 0 \cdot 1 = 0$
- $a = 1 \rightarrow 1 \cdot 1 = 1$

$$\mathbf{a \cdot 0 = 0 \rightarrow}$$

- $a = 1 \rightarrow 1 \cdot 0 = 0$
- $a = 0 \rightarrow 0 \cdot 0 = 0$

$$\mathbf{a + a = a \rightarrow}$$

- $0 + 0 = 0$
- $1 + 1 = 1$

$$\mathbf{a \cdot a = a \rightarrow}$$

- $1 \cdot 1 = 1$
- $0 \cdot 0 = 0$

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

- $a = 1 \rightarrow 1 + 0 = 1$
- $a = 0 \rightarrow 0 + 1 = 1$

$$\mathbf{a \cdot \bar{a} = 0 \rightarrow}$$

- $a = 0 \rightarrow 0 \cdot 1 = 0$
- $a = 1 \rightarrow 1 \cdot 0 = 0$

$$\mathbf{\bar{\bar{a}} = a \rightarrow}$$

- $a = 1 \rightarrow \bar{a} = 0 \rightarrow \bar{\bar{a}} = 1$



2. POSTULADOS, PROPIEDADES Y TEOREMAS

2.3. Propiedades

- Commutativa:
 - $\Rightarrow a + b = b + a$
 - $\Rightarrow a \cdot b = b \cdot a$
- Asociativa:
 - $\Rightarrow (a + b) + c = a + (b + c)$
 - $\Rightarrow (a \cdot b) \cdot c = a \cdot (b \cdot c)$
- Distributiva:
 - $\Rightarrow a \cdot (b + c) = a \cdot b + a \cdot c$
 - $\Rightarrow a + (b \cdot c) = (a + b) \cdot (a + c)$

2.4. Teoremas

1. Ley de absorción

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

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

2. Segunda ley

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

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

3. Leyes de Morgan

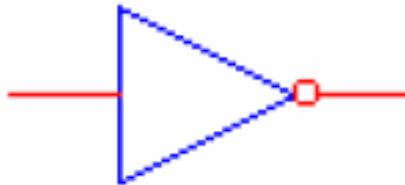
$$\overline{a + b} = \bar{a} \cdot \bar{b}$$

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

3. PUERTAS LOGICAS

3. 1 Puerta NOT o Inversora

Símbolo



ANSI/IEEE 91-1973



ANSI/IEEE 91-1984

Tabla de Verdad

| a | \bar{S} |
|---|-----------|
| 0 | 1 |
| 1 | 0 |

3. PUERTAS LÓGICAS

3. 2 Puerta AND o Multiplicadora

Símbolo



ANSI/IEEE 91-1973



ANSI/IEEE 91-1984

Tabla de Verdad

| a | b | $S = a.b$ |
|---|---|-----------|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

3. PUERTAS LÓGICAS

3. 3 Puerta OR o Sumadora

Símbolo



ANSI/IEEE 91-1973



ANSI/IEEE 91-1984

Tabla de Verdad

| a | b | $S = a + b$ |
|---|---|-------------|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

3. PUERTAS LOGICAS

3. 4 Puerta NOR o Sumadora Inversora

Símbolo



ANSI/IEEE 91-1973



ANSI/IEEE 91-1984

Tabla de Verdad

| a | b | $S = \overline{a + b}$ |
|---|---|------------------------|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

3. PUERTAS LÓGICAS

3. 5 Puerta NAND o Multiplicadora Inversora

Símbolo



ANSI/IEEE 91-1973



ANSI/IEEE 91-1984

Tabla de Verdad

| a | b | $S = \overline{a \cdot b}$ |
|---|---|----------------------------|
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

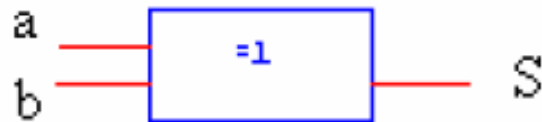
3. PUERTAS LÓGICAS

3. 6 Puerta XOR (OR-EXCLUSIVE)

Símbolo



ANSI/ IEEE 91-1997



ANSI/ IEEE 91-1983

Tabla de Verdad

| a | b | $S = a \oplus b$ |
|---|---|------------------|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

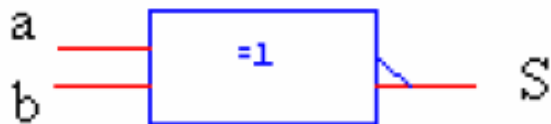
3. PUERTAS LÓGICAS

3. 7 Puerta XNOR (NOR-EXCLUSIVE)

Símbolo



ANSI/ IEEE 91-1997

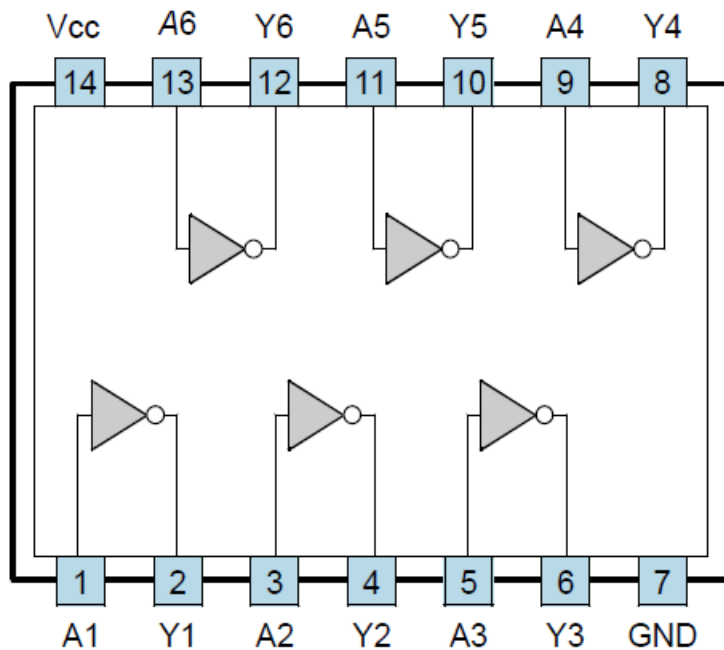


ANSI/ IEEE 91-1983

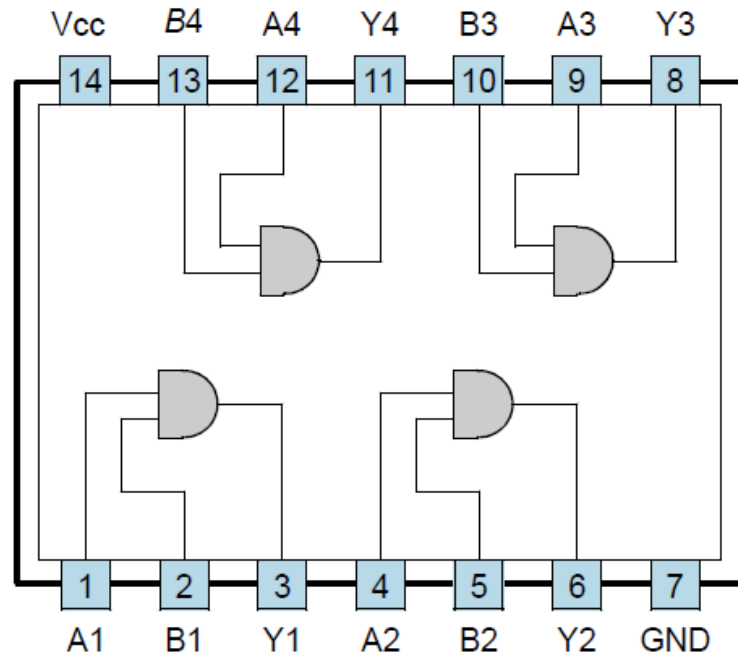
Tabla de Verdad

| a | b | $S = \overline{a \oplus b}$ |
|---|---|-----------------------------|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

3. PUERTAS LÓGICAS

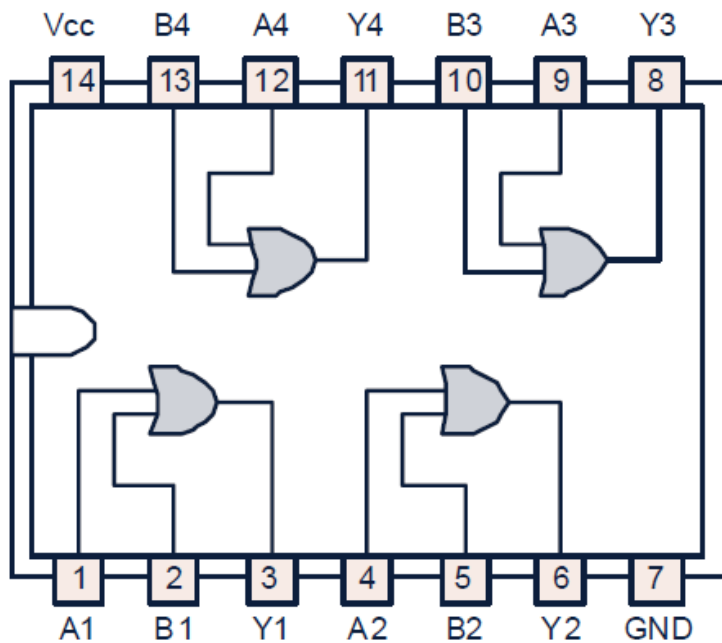


7404

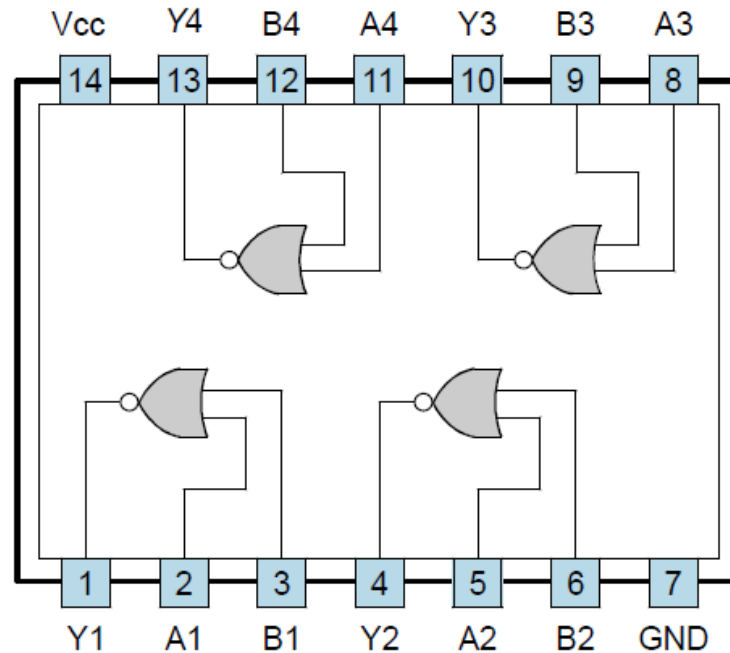


7408

3. PUERTAS LÓGICAS

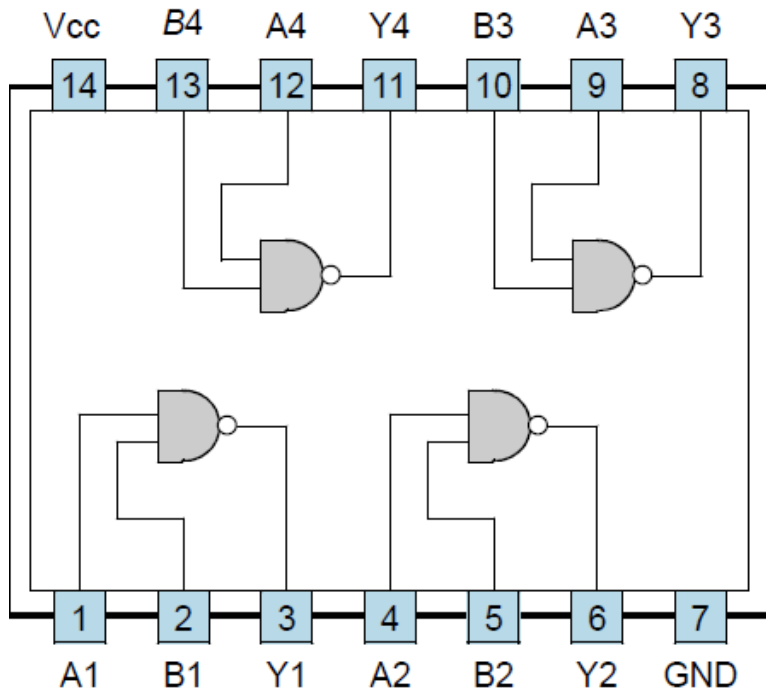


7432

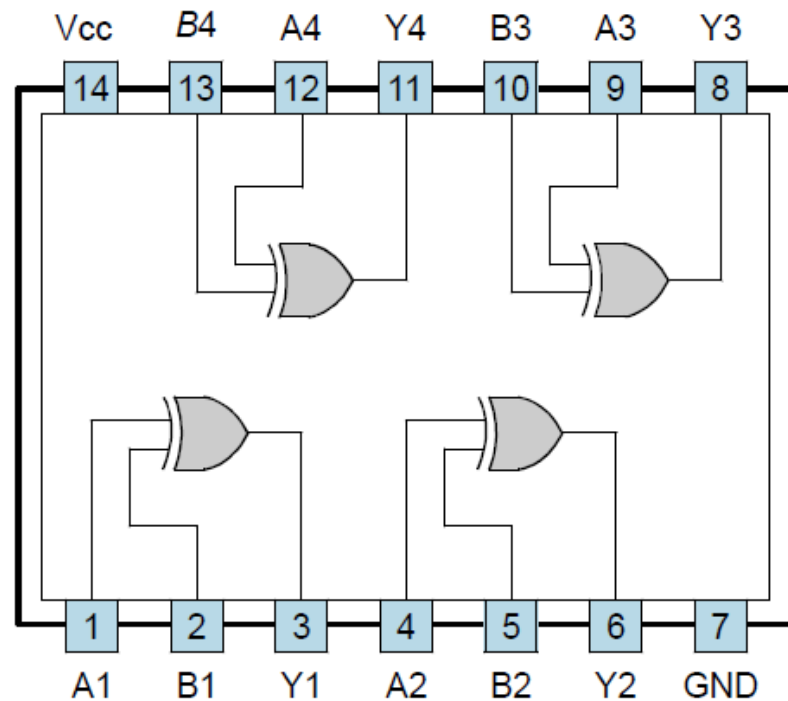


7427

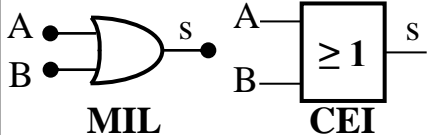
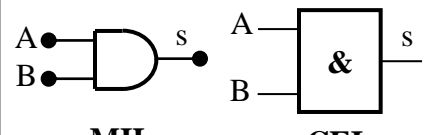
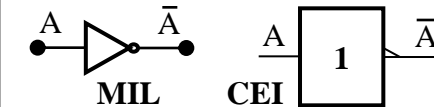
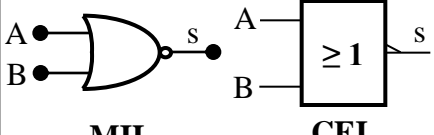
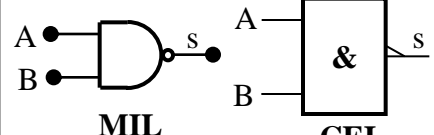
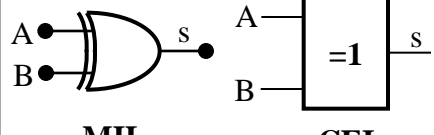
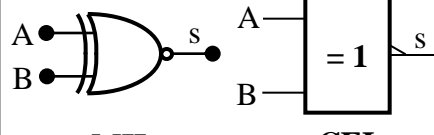
3. PUERTAS LOGICAS



7400



7486

| FUNCION | SIMBOLO | ECUACION | TABLA VERDAD | | | | | | | | | | | | | | | |
|---------|-------------------------------------------------------------------------------------|--------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|------|------|---|---|---|---|---|---|---|---|---|---|---|---|
| OR |  | $F(s) = A + B$ | <table><tr><th>B</th><th>A</th><th>F(s)</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table> | B | A | F(s) | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |
| B | A | F(s) | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | | | | | | | | | | | | | | | | |
| 0 | 1 | 1 | | | | | | | | | | | | | | | | |
| 1 | 0 | 1 | | | | | | | | | | | | | | | | |
| 1 | 1 | 1 | | | | | | | | | | | | | | | | |
| AND |  | $F(s) = A \cdot B$ | <table><tr><th>B</th><th>A</th><th>F(s)</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table> | B | A | F(s) | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 |
| B | A | F(s) | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | | | | | | | | | | | | | | | | |
| 0 | 1 | 0 | | | | | | | | | | | | | | | | |
| 1 | 0 | 0 | | | | | | | | | | | | | | | | |
| 1 | 1 | 1 | | | | | | | | | | | | | | | | |
| NOT |  | $F(A) = \overline{A}$ | <table><tr><th>A</th><th>F(A)</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table> | A | F(A) | 0 | 1 | 1 | 0 | | | | | | | | | |
| A | F(A) | | | | | | | | | | | | | | | | | |
| 0 | 1 | | | | | | | | | | | | | | | | | |
| 1 | 0 | | | | | | | | | | | | | | | | | |
| NOR |  | $F(s) = \overline{A + B}$ | <table><tr><th>B</th><th>A</th><th>F(s)</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table> | B | A | F(s) | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| B | A | F(s) | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | | | | | | | | | | | | | | | | |
| 0 | 1 | 0 | | | | | | | | | | | | | | | | |
| 1 | 0 | 0 | | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | | | | | | | | | | | | | | | | |
| NAND |  | $F(s) = \overline{A \cdot B}$ | <table><tr><th>B</th><th>A</th><th>F(s)</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table> | B | A | F(s) | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 |
| B | A | F(s) | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | | | | | | | | | | | | | | | | |
| 0 | 1 | 1 | | | | | | | | | | | | | | | | |
| 1 | 0 | 1 | | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | | | | | | | | | | | | | | | | |
| XOR |  | $F(s) = A \oplus B$ | <table><tr><th>B</th><th>A</th><th>F(s)</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table> | B | A | F(s) | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 |
| B | A | F(s) | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | | | | | | | | | | | | | | | | |
| 0 | 1 | 1 | | | | | | | | | | | | | | | | |
| 1 | 0 | 1 | | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | | | | | | | | | | | | | | | | |
| XNOR |  | $F(s) = \overline{A \oplus B}$ | <table><tr><th>B</th><th>A</th><th>F(s)</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table> | B | A | F(s) | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 |
| B | A | F(s) | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | | | | | | | | | | | | | | | | |
| 0 | 1 | 0 | | | | | | | | | | | | | | | | |
| 1 | 0 | 0 | | | | | | | | | | | | | | | | |
| 1 | 1 | 1 | | | | | | | | | | | | | | | | |



4. DIAGRAMAS DE CIRCUITOS CON PUERTAS LÓGICAS

4.1 Esquemas a partir de funciones lógicas

Para obtener la ecuación correspondiente a un circuito digital, tenemos en cuenta que:

Cada función (puerta) lógica, nos proporciona un término o variable de la ecuación.

Procedemos del siguiente modo:

Empezando por las entradas del circuito y hacia la salida, obtenemos la ecuación parcial en la salida de cada función (puerta).

La ecuación final es la obtenida en la función (puerta), donde se encuentra la salida del circuito.

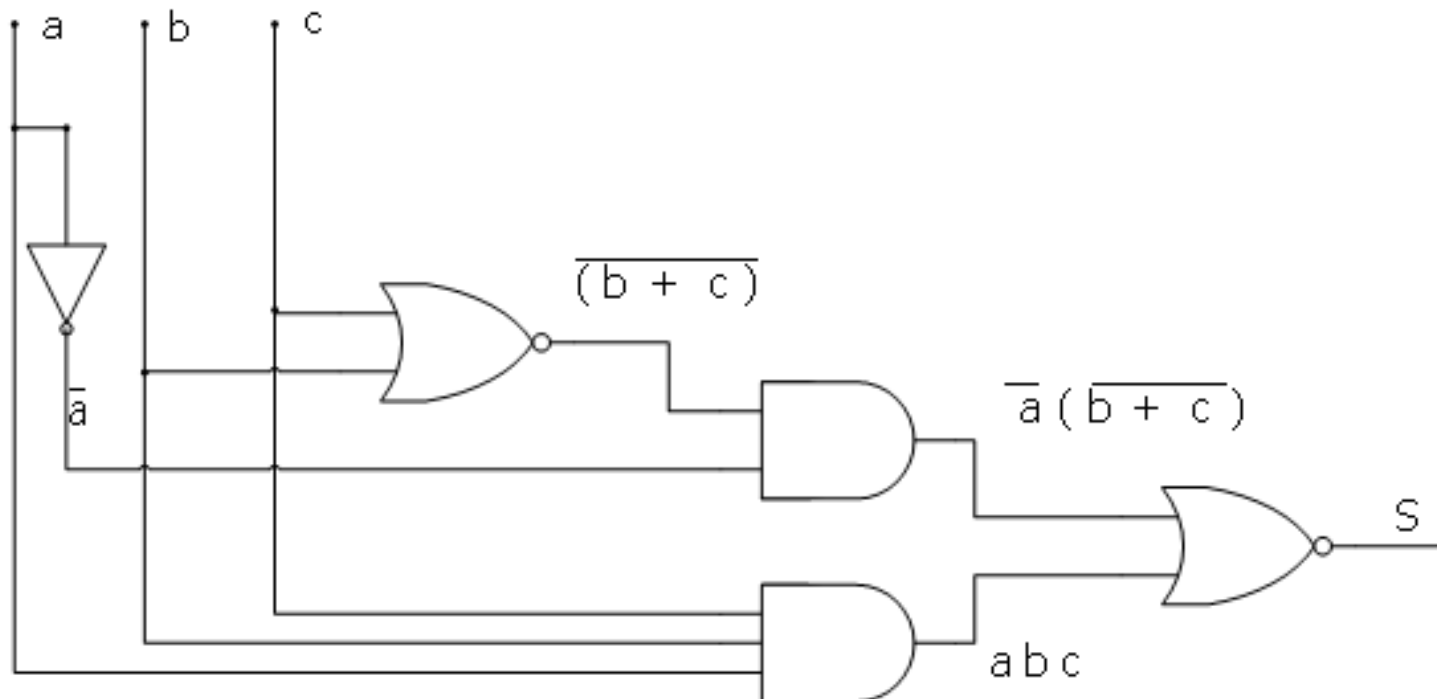
4. DIAGRAMAS DE CIRCUITOS CON PUERTAS LÓGICAS

Dibuja el esquema que resuelve la siguiente ecuación.

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

Pasos a seguir:

- ☐ Resuelvo los paréntesis (las operaciones negadas son como paréntesis).
- ☐ Resuelvo los productos.
- ☐ Resuelvo las sumas.





4. DIAGRAMAS DE CIRCUITOS CON PUERTAS LÓGICAS

4.1 Esquemas a partir de funciones lógicas

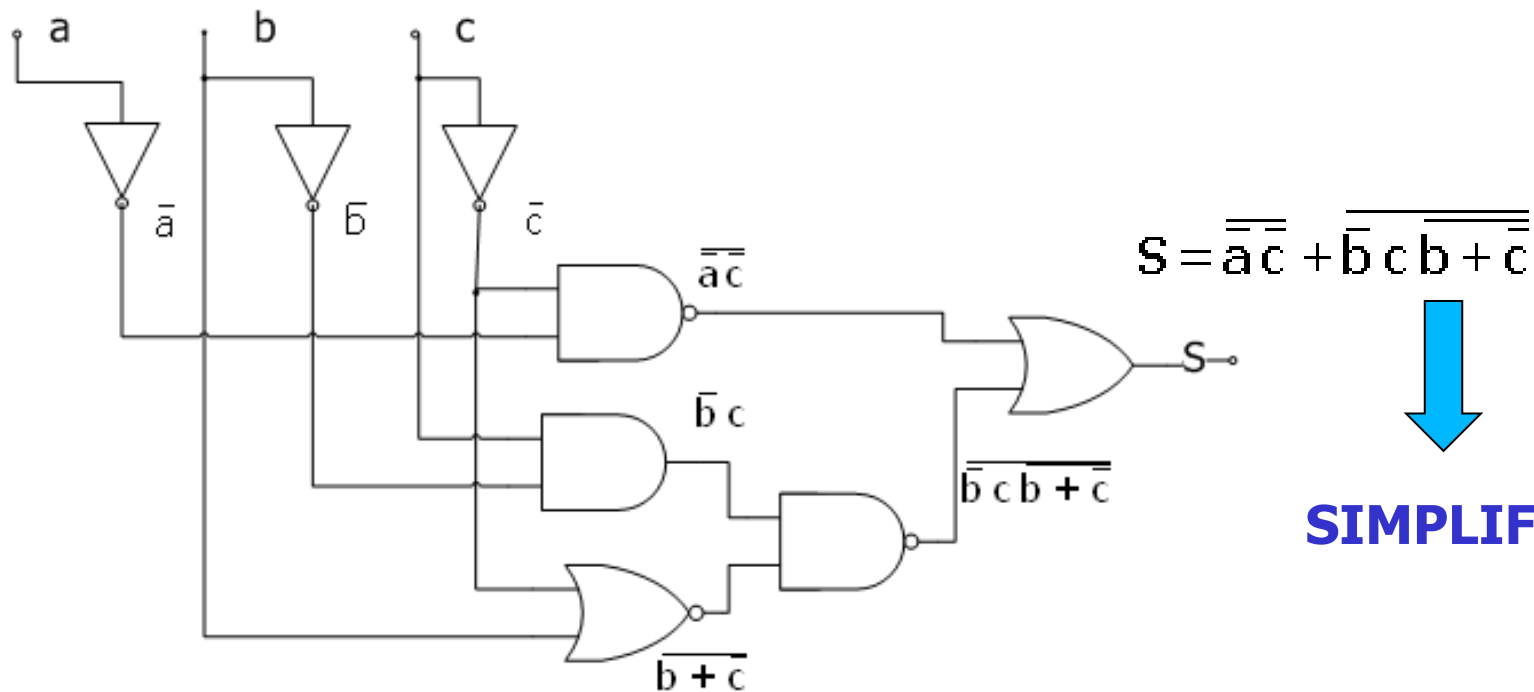
Dibuja el esquema que resuelva las siguientes ecuaciones:

$$S = a(\bar{c} + \bar{d}) + a\bar{b} + b\bar{a} + (\bar{a}\bar{c})d$$

$$S = a\bar{b}c\bar{d} + \bar{a}$$

$$S = \overline{a + b\bar{c}} + a\bar{b}\bar{c}$$

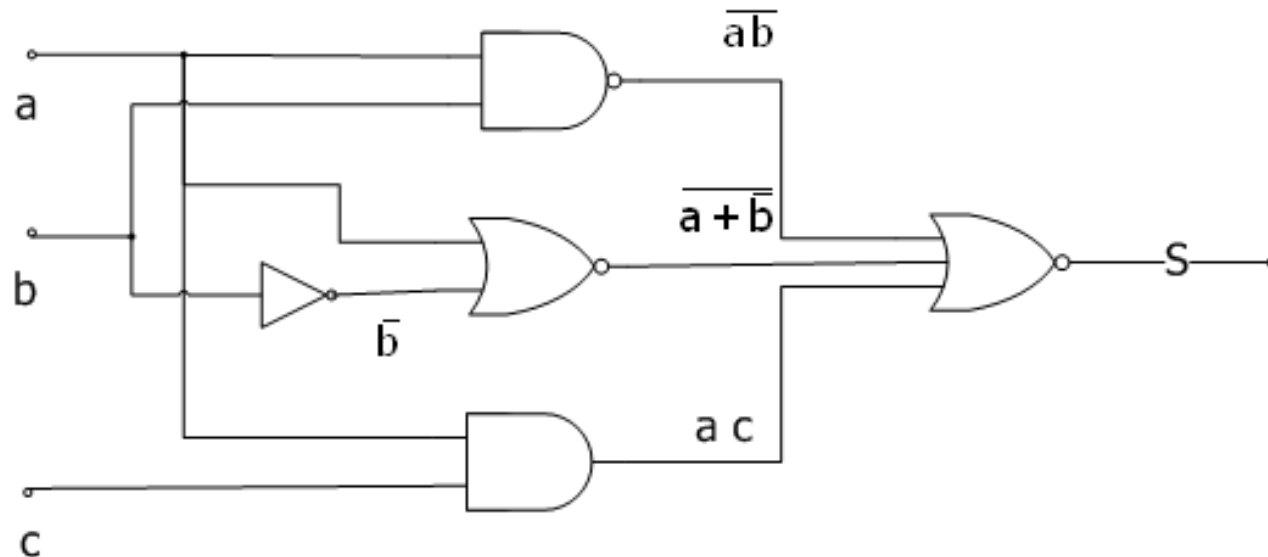
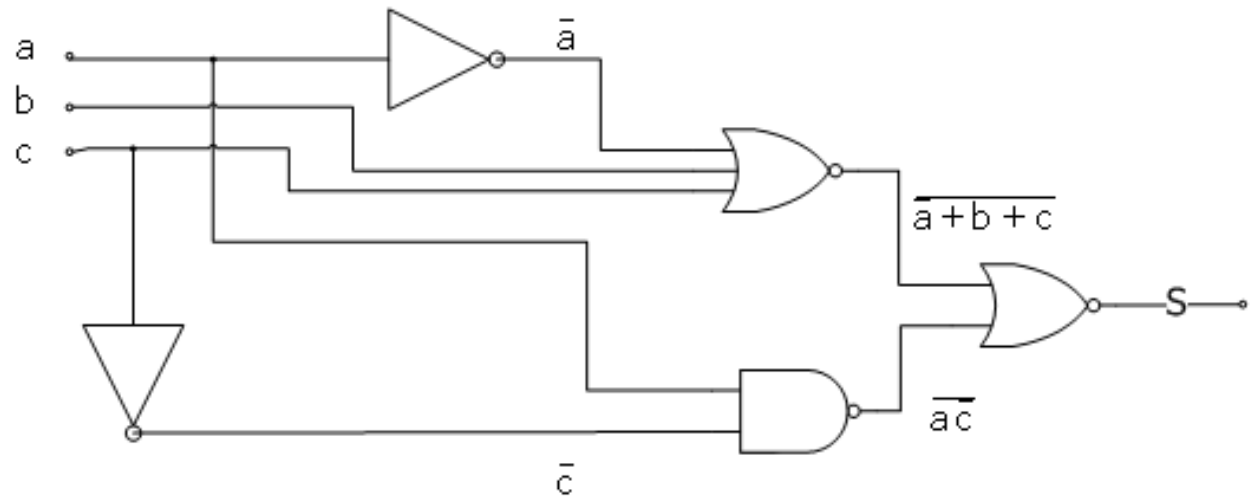
4.2 Funciones lógicas a partir de esquemas



SIMPLIFICAR

4. DIAGRAMAS DE CIRCUITOS CON PUERTAS LÓGICAS

4.3 Obtener la tabla de verdad de los esquemas lógicos:





4. DIAGRAMAS DE CIRCUITOS CON PUERTAS LÓGICAS

TABLA DE VERDAD A PARTIR DE ECUACION

- Analizamos el resultado de cada término, para todas las posibles combinaciones de las variables de entrada.
- El análisis se efectuará siguiendo un orden similar al indicado para obtener el circuito .

Veamos el siguiente ejemplo: $S = [A \cdot (B + \bar{C})] \cdot \bar{D}$

| D | C | B | A | \bar{C} | $B + \bar{C}$ | $A \cdot (B + \bar{C})$ | \bar{D} | $[A \cdot (B + \bar{C})] \cdot \bar{D}$ |
|---|---|---|---|-----------|---------------|-------------------------|-----------|-----------------------------------------|
| 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 |
| 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |



4. DIAGRAMAS DE CIRCUITOS CON PUERTAS LÓGICAS

ECUACION A PARTIR DE LA TABLA DE VERDAD

- A partir de la tabla de verdad podemos obtener la **ecuación canónica**.
- Se define ecuación canónica, a aquella en la que aparecen todas las variables en cada termino de la ecuación.
- Pueden obtenerse de dos modos:
 - Como suma de productos (MINTERMS).
 - Como producto de sumas (MAXTERMS).



4. DIAGRAMAS DE CIRCUITOS CON PUERTAS LÓGICAS

ECUACION DE MINTERM

- Para obtenerla, seleccionamos las combinaciones de la tabla de verdad en las que la salida es "1". Ejemplo:

| C | B | A | S |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 |

Ecuación canónica

$$S = \overline{A}B\overline{C} + A\overline{B}\overline{C} + \overline{A}BC + ABC$$

Ecuación genérica

$$S = \sum_{n=3} (2, 3, 5, 7)$$



4. DIAGRAMAS DE CIRCUITOS CON PUERTAS LÓGICAS

ECUACION DE MAXTERM

- Para obtenerla, seleccionamos las combinaciones de la tabla de verdad en las que la salida es "0". Ejemplo:

| C | B | A | S |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 |

Ecuación canónica

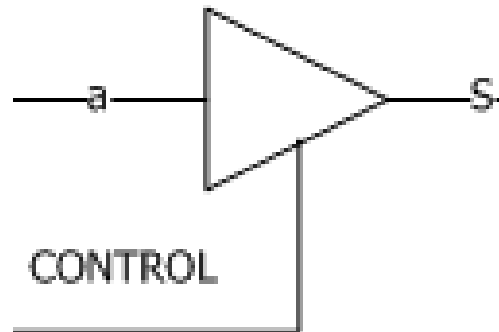
$$S = (A + B + C)(\bar{A} + B + C)(A + B + \bar{C})(A + \bar{B} + C)$$

Ecuación genérica

$$S = \prod_{n=3} (0, 1, 4, 6)$$

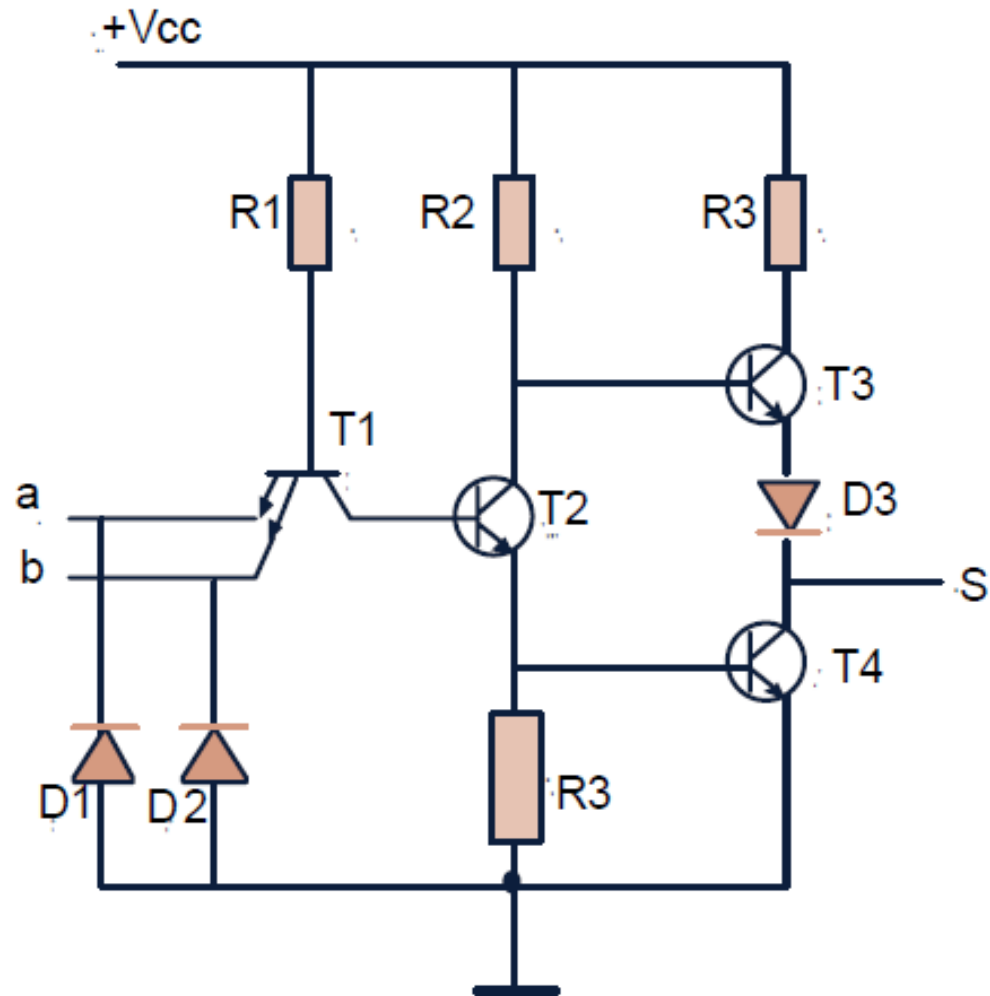
5. PUERTAS LÓGICAS TRIESTADO

Tercer estado también denominado estado de **alta impedancia** o **circuito abierto**.



| a | Control | S |
|---|---------|---------------------|
| 0 | 0 | 0 |
| 0 | 1 | Z (Alta impedancia) |
| 1 | 0 | 1 |
| 1 | 1 | Z |

6. ARQUITECTURA INTERNA DE UNA PUERTA NAND TTL





7. CARACTERÍSTICAS GENERALES DE LAS PUERTAS LÓGICAS INTEGRADAS

- Tensión de alimentación y su tolerancia
- Temperatura de trabajo
- *Fan-out*
- Niveles de tensión de entrada y salida
 - V_{IL} : Es la máxima tensión de entrada considerada como “nivel bajo”.
 - V_{IH} : Es la mínima tensión de entrada considerada como “un nivel alto”.
 - V_{OL} : Es la máxima tensión de salida considerada como un “nivel bajo”.
 - V_{OH} : Es la mínima tensión de salida considerada como un “nivel alto”.



7. CARÁCTERÍSTICAS GENERALES DE LAS PUERTAS LÓGICAS INTEGRADAS

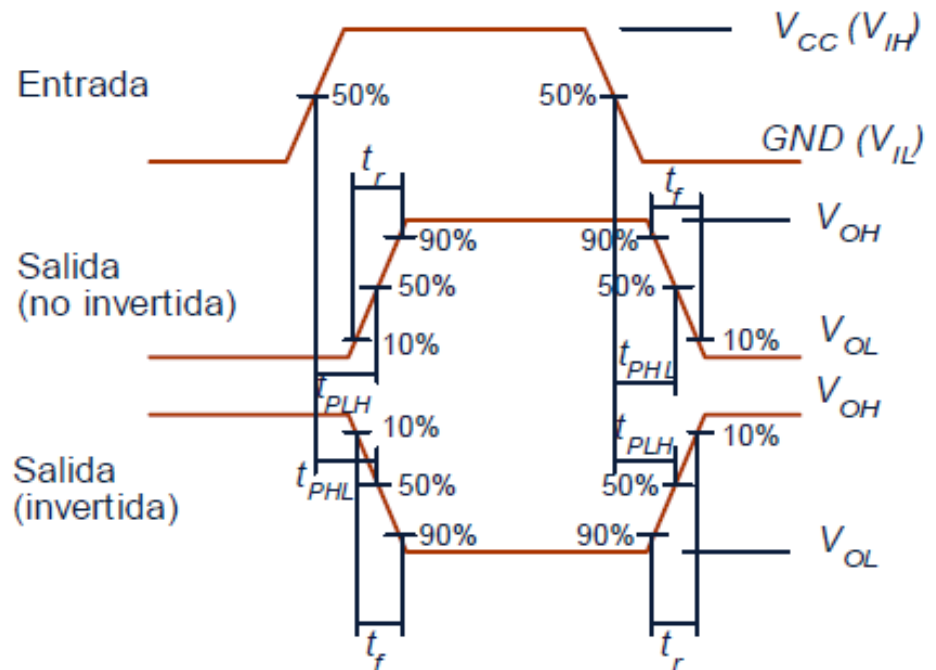
- Margen de ruido: En los estados lógicos alto y bajo, indica las variaciones máximas que se pueden producir a la entrada sin que la salida varié su estado.

$$NI_H = V_{OH} \text{ min} - V_{IH} \text{ min}$$

$$NI_L = V_{IL} \text{ máx} - V_{OL} \text{ máx}$$

7. CARÁCTERÍSTICAS GENERALES DE LAS PUERTAS LÓGICAS INTEGRADAS

- **Tiempo medio de propagación:** Es el retardo que transcurre desde que cambia una entrada y produce un cambio lógico a la salida



t_{PLH} = tiempo de propagación de nivel bajo a nivel alto.

t_{PHL} = tiempo de propagación de nivel alto a nivel bajo.

t_r = tiempo de subida de la señal de salida.

t_f = tiempo de bajada de la señal de salida.



7. CARACTERÍSTICAS GENERALES DE LAS PUERTAS LÓGICAS INTEGRADAS

- **Disipación de potencia** :Se indica la disipación estática por función. O los consumos de corriente de alimentación de cortocircuito y de entrada y salida para lo dos valores lógicos.

CARACTERÍSTICAS IDEALES

- Gran densidad de integración.
- Alta velocidad de propagación.
- Mínimo consumo.
- Máxima inmunidad al ruido y a las variaciones de temperatura.
- Compatibilidad con otras familias lógicas.
- Bajo coste.



8. FAMILIA TTL

- TTL es el acrónimo de *Transistor Transistor Logic*
- La primera serie de dispositivos de esta familia que se creo se denomino *TTL Standar* , que es conocida por la mayoría de los fabricantes como la serie 74 y cuyas principales características son:
 - Tensión de alimentación $5V \pm 10\%$
 - Fan-out = a 10 (numero de salida capaz de hacer)
 - Niveles de tensión:
 - $V_{IH \min} = 2,0V$
 - $V_{IL \max} = 0,8V$
 - $V_{OH \min} = 2,4V$
 - $V_{OL \max} = 0,4V$
 - Margen de ruido en ambos niveles 0,4V
 - Tiempo de propagación 10ns
 - Disipación de potencia 10mW por función



8. FAMILIA TTL

Para mejorar las características Standars se desarrollaron las siguientes series:

- **54/74L** (*Low-power*): con un consumo de **1mW** pero un tiempo de propagación de **33ns**
- **54/74S** (*Schottky*) con un tiempo de progagación de **3ns** y una disipación de potencia de **20mW**
- **54/74LS** (*Low-power Schottky*) con una potencia de **3mW** y un tiempo de propagación de **10ns**
- **54/74AS** (*advanced Schottky*) con una potencia disipada de **7mW** y un tiempo de propagación de **1,5ns**
- **54/74F** (*FAST Fairchid Advanced Schottky*) con una potencia de disipación de **4mW** y un tiempo de propagación de **3ns**



8. FAMILIA TTL

Para mejorar las características Standars se desarrollaron las siguientes series:

- **54/74L** (*Low-power*): con un consumo de **1mW** pero un tiempo de propagación de **33ns**
- **54/74S** (*Schottky*) con un tiempo de progagación de **3ns** y una disipación de potencia de **20mW**
- **54/74LS** (*Low-power Schottky*) con una potencia de **3mW** y un tiempo de propagación de **10ns**
- **54/74AS** (*advanced Schottky*) con una potencia disipada de **7mW** y un tiempo de propagación de **1,5ns**
- **54/74F** (*FAST Fairchid Advanced Schottky*) con una potencia de disipación de **4mW** y un tiempo de propagación de **3ns**



9. FAMILIA CMOS

- CMOS es el acrónimo de *Complementary Metal Oxide Semiconductor*
- La familia CMOS básica aparece en los catálogos como serie 4000 sus características más significativas son:
 - Tensión de alimentación de 3V a 18V
 - Fan-out = 50 numero de puertas que abre
 - Niveles de tensión: para una tensión de 5V
 - $V_{IH\min} = 3,5V$
 - $V_{IL\max} = 1,5V$
 - $V_{OH\min} = 4,95V$
 - $V_{OL\max} = 0,05V$
 - Margen de ruido muy elevado no le afectan impulsos de 30% de la tensión de alimentación
 - Tiempo de propagación varía inversamente a la tensión de alimentación
para : 5V 60ns para: 10V 30ns
 - Disipación de potencia 10nW por función



9. FAMILIA CMOS

Para mejorar las características standards se desarrollaron las siguientes series:

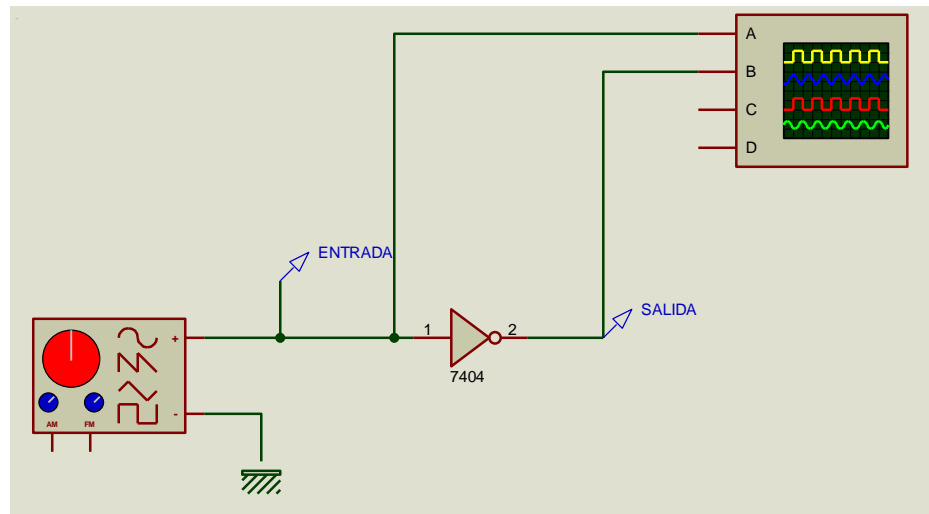
- **54/74C** mantiene las propiedades de la tecnología CMOS y es compatible pin a pin con la serie TTL standard.
- **54/74HC** (*High Speed CMOS*) se alimenta con tensiones comprendidas entre 2 y 6V y tiempo de propagación de 8ns
- **54/74HCT** es similar a la anterior pero compatible eléctricamente con la familia TTL.
- **54/74ACT** (*Advanced CMOS*) con una tensión de alimentación de $5V \pm 10\%$ tienen una potencia disipación de **1mW** y un tiempo de propagación de **3ns**.

10. DIAGRAMAS DE TIEMPOS o CRONOGRAMAS

Un diagrama de tiempos o cronograma, es una gráfica que representa las relaciones entre dos o más formas de onda en función del tiempo.

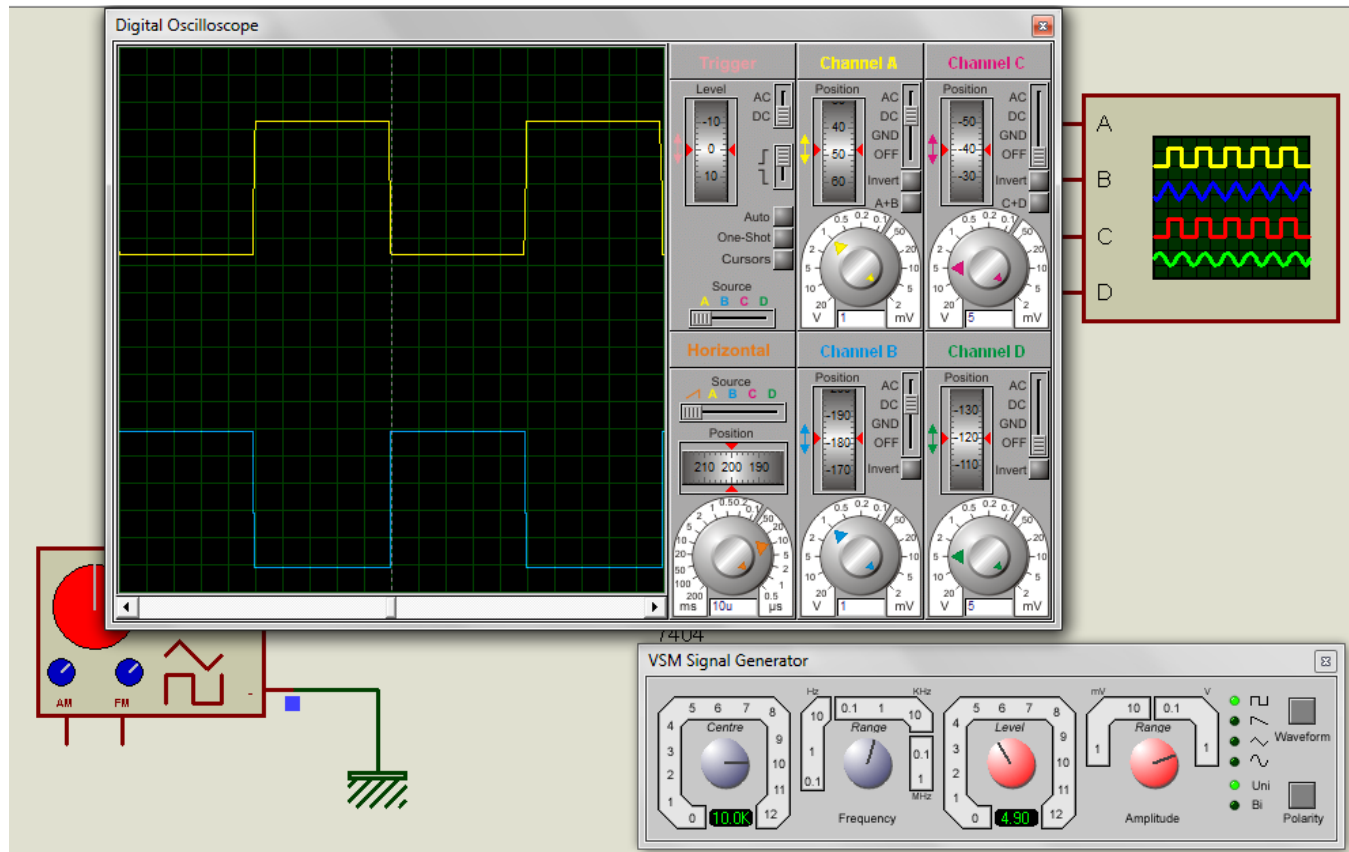
Por ejemplo veamos la respuesta de una puerta inversora ante un tren de pulsos.

El circuito diseñado está simulado en Proteus, utilizamos un generador de funciones y un osciloscopio



10. DIAGRAMAS DE TIEMPOS

Al simular el funcionamiento obtendríamos las señales de entrada y salida visualizadas en el osciloscopio:



10. DIAGRAMAS DE TIEMPOS

Como se puede comprobar cuando la entrada está a nivel alto, la salida está a nivel bajo y viceversa. En el mismo entorno podríamos obtener la relación entre entrada y salida mediante el análisis digital

