eman ta zabal zazu
Universidad del País Vasco
Euskal Herriko Unibertsitatea

Ingeniaritza Goi Eskola Teknikoa
Escuela Técnica Superior de Ingeniería
Bilbao

electrónica general

Introducción a la Electrónica Digital

Tema 5

DEPARTAMENTO DE TECNOLOGÍA ELECTRÓNICA

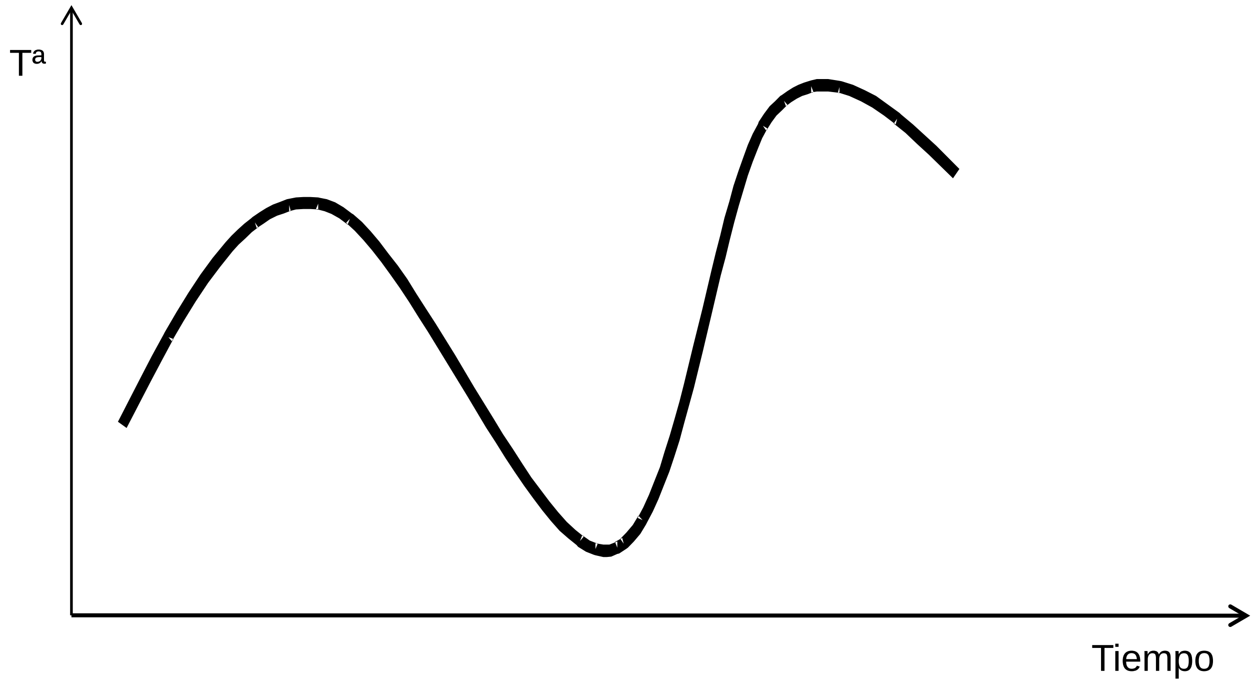
Índice

Introducción a la Electrónica Digital

La **electrónica analógica** utiliza magnitudes con valores continuos

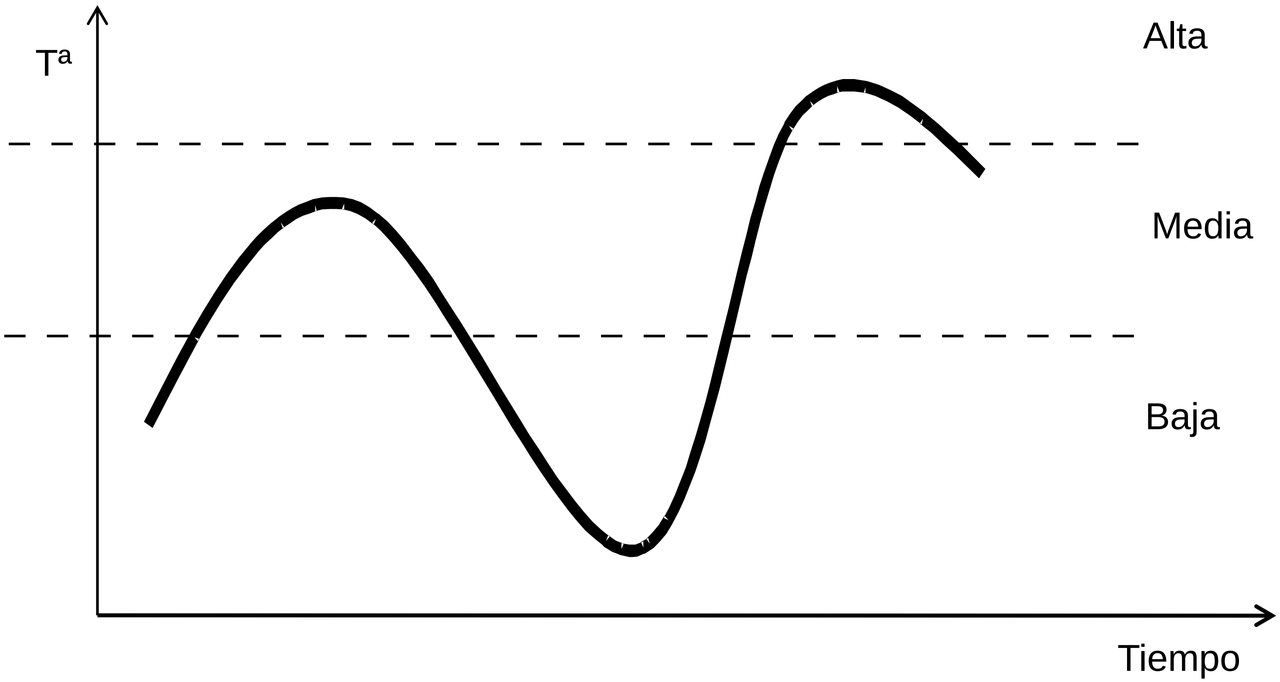
La **electrónica digital** emplea magnitudes con valores discretos.

Una **señal analógica** es aquella cuya magnitud, en cada instante de tiempo, puede tomar cualquiera de los valores infinitos del rango dónde esta definida, pudiendo cambiar de valor en cantidades arbitrariamente pequeñas



La **señales digitales** son aquellas cuyas magnitudes, en cada instante de tiempo, solo pueden tomar un valor entre un conjunto finito de q valores o estados. En el paso de un valor a otro se produce una discontinuidad al no existir valores intermedios

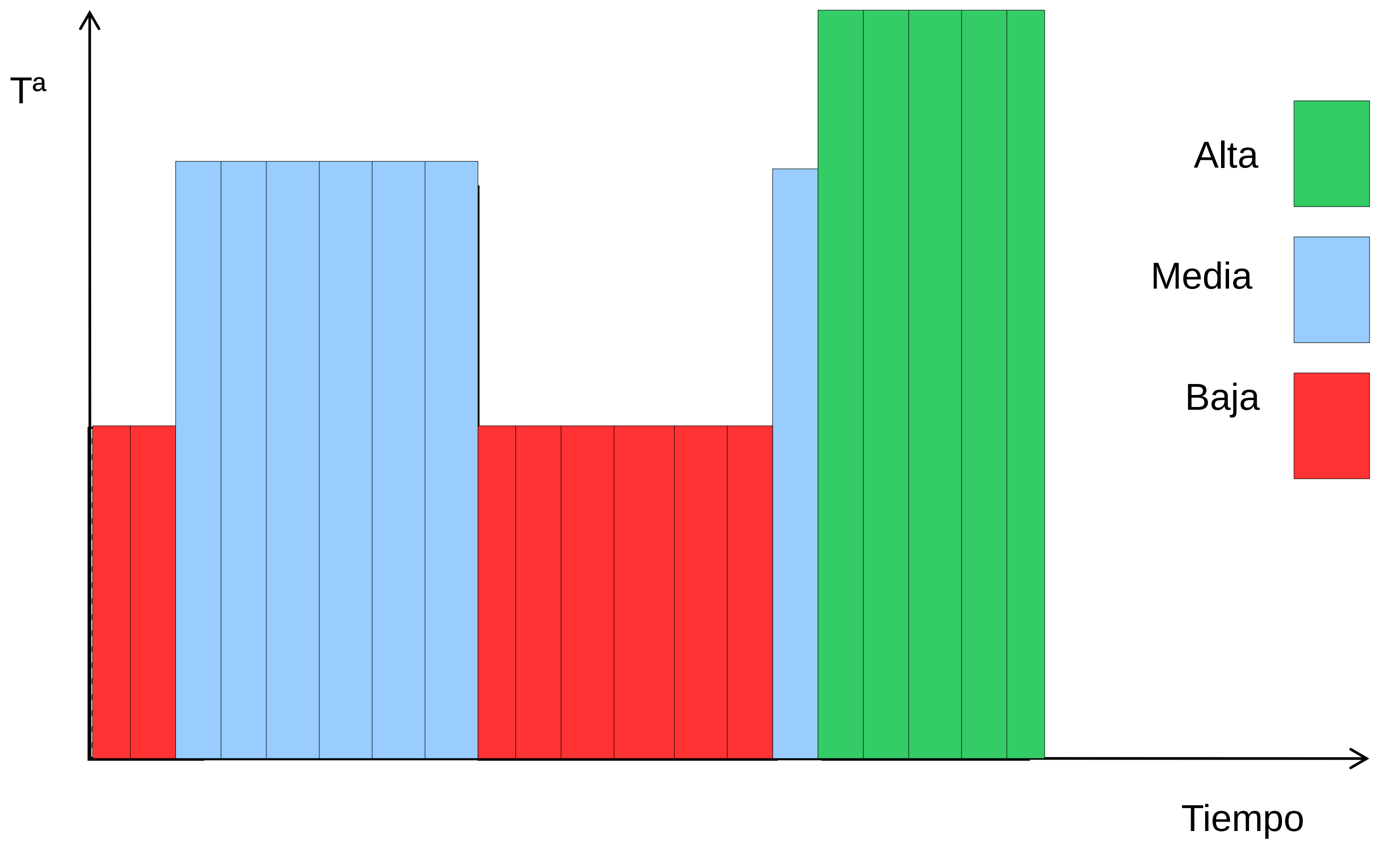
La señales analógicas se pueden digitalizar (muestreo y cuantificación)



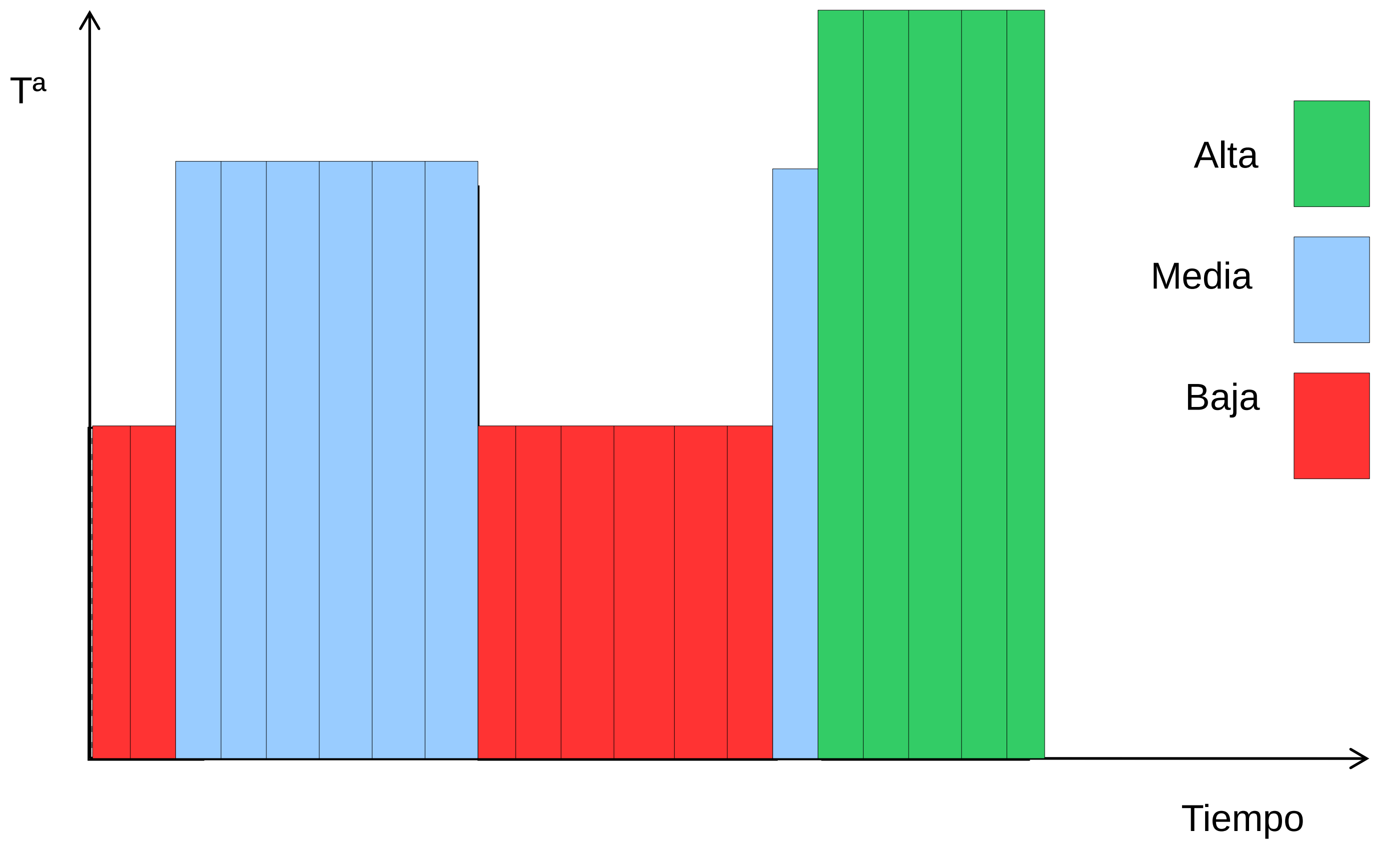
1. Señales analógicas y señales digitales

La figura muestra la señal analógica digitalizada, con q = 3.

En la digitalización de la señal se pierde parte de la información (en tiempo y magnitud), pero hay algunas ventajas.



- Información digitalizada, que se puede procesar por un sistema digital.
- Esta información se recoge fácilmente.
- La precisión y la exactitud se mantiene en todo el sistema fácilmente.
- El ruido y las perturbaciones son menos perjudiciales para el sistema digital.



2. Sistemas de numeración: Sistema Binario

Sistema numérico: conjunto de reglas que se aplican a una forma concreta de representar los números.

En el proceso de digitalización, una vez elegido q, tenemos que elegir una serie de “símbolos” que nos permitan representar los valores posibles de la señal digitalizada.

Un número cualquiera, **N**, tomando una base, b, se puede representar mediante una serie de coeficientes (a_i) o símbolos relativos a la base en que esta representado ($0 \leq a_i < b-1$)

$$N = a_n \, a_{n-1} \cdots a_i \qquad \cdots a_0 \, , a_{-1} \cdots a_{-p}$$

$$N = a_n b^n + a_{n-1} b^{n-1} + \cdots + a_0 b^0 + a_{-1} b^{-1} + \cdots + a_{-p} b^{-p}$$

En el caso del sistema digital:

$$N = a_n 10^n + a_{n-1} 10^{n-1} + \cdots + a_i 10^i + \cdots + a_0 10^0 + a_{-1} 10^{-1} + \cdots + a_{-p} 10^{-p}$$

$$N = 237,4 \Leftrightarrow N = 2 \cdot 10^2 + 3 \cdot 10^1 + 7 \cdot 10^0 + 4 \cdot 10^{-1}$$

2. Sistemas de numeración: Sistema Binario

Señales digitales con dos estados ***: binarias***

Variable Lógica: representación de los valores discretos de una señal digital binaria.

Condiciones que ha de cumplir:

1.- La variable lógica puede adoptar uno de los dos valores posibles.

2.- Los valores se expresan por sentencias declarativas.

3.- Los dos posibles valores expresados por las sentencias declarativas deben ser excluyentes.

Sistema binario $\longrightarrow$ B = 2; $a_i = \{0,1\}$

Interruptor	Transistor	Dirección	Sexo	Afirmación	V. lógica
Abierto	Corte	Izquierda	Mujer	Falsa	0
Cerrado	Saturación	Derecha	Hombre	Verdadera	1

A cada uno de los coeficientes de un número binario se le denomina ***bit***

$N = 1101_2 = 13_{10}$

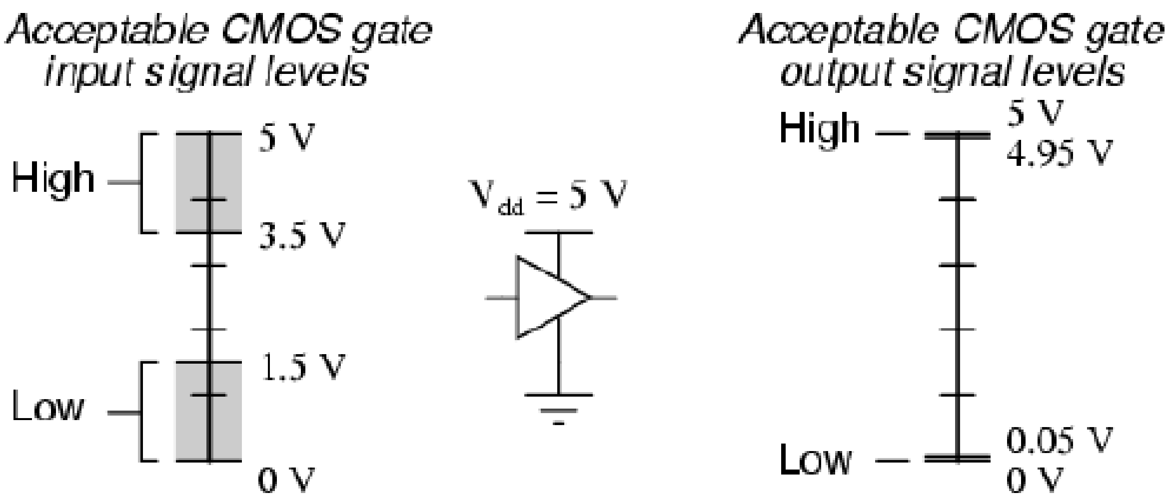
MSB (Most Significant Bit)

LSB (Least Significant Bit)

N = 43,40625

Parte entera				Parte fraccionaria			
División	**Cociente**	**Resto**	**Coeficiente**	**Producto**	**Fracción**	**Entera**	**Coefic.**
43:2	21	1	a_0	0.40625x2	0.8125	0	a_{-1}
21:2	10	1	a_1	0.8125x2	0.625	1	a_{-2}
10:2	5	0	a_2	0.625x2	0.25	1	a_{-3}
5:2	2	1	a_3	0.25x2	0.5	0	a_{-4}
2:2	1	0	a_4	0.5x2	0	1	a_{-5}
1:2	0	1	a_5				

$43{,}40625 = 101011{,}01101_2$


Acceptable CMOS gate input signal levels
High
Low
5 V
3.5 V
1.5 V
0 V
V_{dd} = 5 V
Acceptable CMOS gate output signal levels
High
Low
5 V
4.95 V
0.05 V
0 V

2. Sistemas de numeración: Sistema Binario

En electrónica son utilizados otros sistemas de numeración para acotar la notación binaria:

SISTEMA OCTAL

$B = 8 = 2^3$; $a_i = \{0, 1, 2, 3, 4, 5, 6, 7\}$ Equivale a 3 cifras en binario

SISTEMA HEXADECIMAL:

$B = 16 = 2^4$; $a_i = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F\}$ Equivale a 4 cifras en binario

Es muy fácil pasar de un sistema binario, y viceversa :

$$N = 10100111_2 = A7_{16} = 247_8 = 167_{10}$$

La base matemática de las operaciones entre variables lógicas es el álgebra de Boole.

El álgebra de Boole es todo conjunto (B) de variables lógicas relacionadas por dos operaciones denominadas ***suma y producto lógico***, que cumplen los siguientes postulados:

1. Ambas operaciones son conmutativas:

$$a + b = b + a; \;\; a \cdot b = b \cdot a; \qquad \forall a, b \in \mathbf{B}$$

2. Existe un elemento neutro para cada operación, el 0 para la suma y el 1 para el producto

$$a + 0 = 0 + a = a; \;\; a \cdot 1 = 1 \cdot a = a; \qquad \forall a, b \in \mathbf{B}$$

3. Cada operación es distributiva con respecto a la otra:

$$a + (b \cdot c) = (a + b) \cdot (a + c)\,; \qquad \forall a, b, c \in \mathbf{B}$$

$$a \cdot (b + c) = (a \cdot b) + (a \cdot c)\,; \qquad \forall a, b, c \in \mathbf{B}$$

4. Para cada elemento, existe un elemento, y solo uno, tal que:

$$a + \overline{a} = 1; \quad a \cdot \overline{a} = 0 \quad \forall a \in \mathbf{B}$$

Siendo $\overline{a}$ el complemento de a

+	0	1
0	0	1
1	1	1

*	0	1
0	0	0
1	0	1

Operaciones del álgebra de Boole

A los anteriores postulados, hemos de añadir los siguientes teoremas:

$$(1)\quad a + a = a; \quad a \cdot a = a$$

$$(2)\quad a + 1 = 1; \quad a \cdot 0 = 0$$

$$(3)\quad a = \overline{\overline{a}}$$

$$(4)\quad a + a \cdot b = a; \quad a \cdot (a + b) = a$$

$$(5)\quad a + (b + c) = (a + b) + c; \quad a \cdot (b \cdot c) = (a \cdot b) \cdot c$$

$$(6)\quad \overline{(a + b)} = \overline{a} \cdot \overline{b}; \quad \overline{a \cdot b} = \overline{a} + \overline{b} \quad \textit{Teorema De Morgan}$$

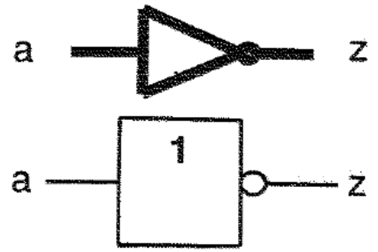
$$(7)\quad \overline{f\,(a, b, \ldots, n, +, \cdot)} = f\left(\overline{a}, \overline{b}, \ldots, \overline{n}, \cdot, +\right) \quad \textit{Teorema de Shannon}$$

Función inversión o NOT (-)

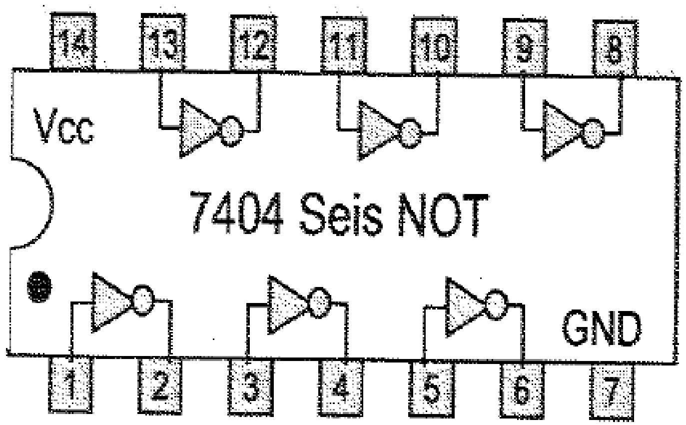
La función NOT corresponde a la operación complemento lógico:

$$z = \overline{a}$$

En electrónica digital se utiliza un círculo para indicar que la entrada o la salida esta complementada



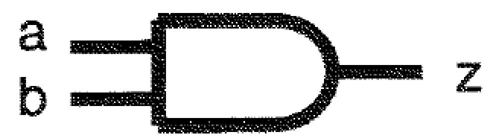
a	z
0	1
1	0

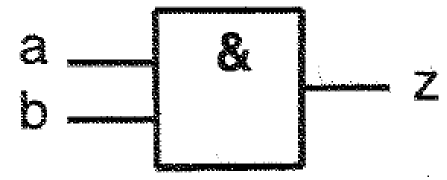


Función AND (·)

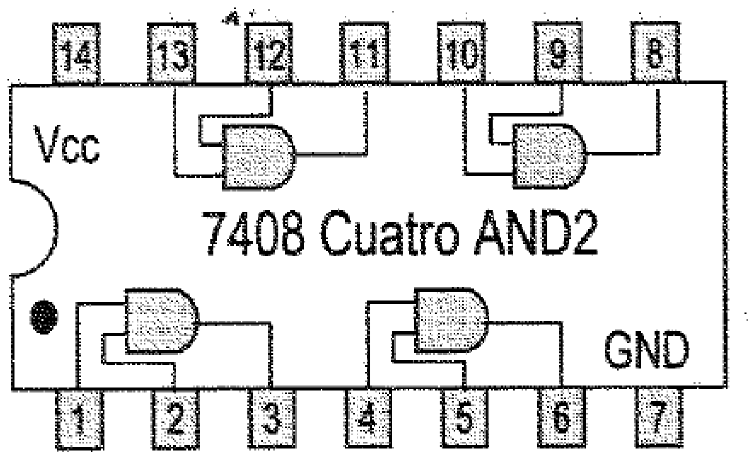
La función AND corresponde a la operación producto lógico:

$$z = a \cdot b$$





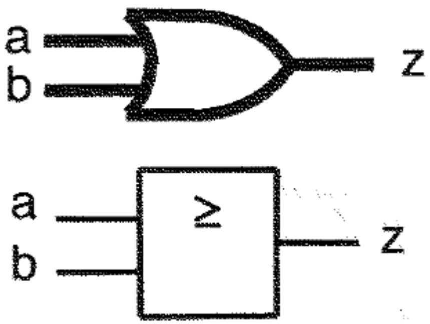
a	b	z
0	0	0
0	1	0
1	0	0
1	1	1



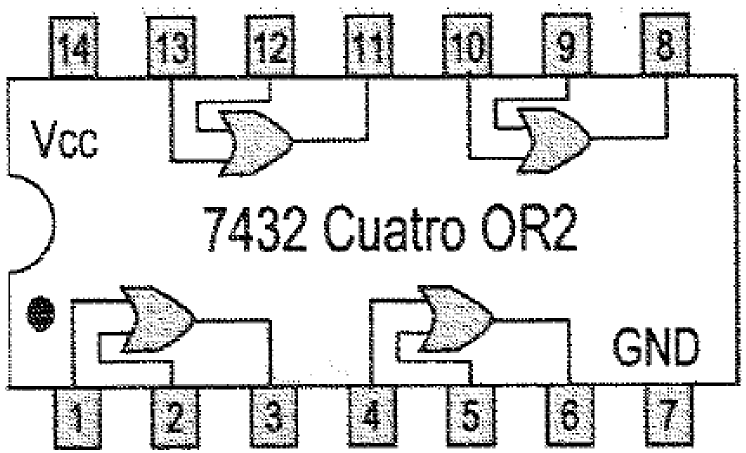
Función OR (+)

La función OR corresponde a la operación suma lógica:

$$z = a + b$$



a	b	z
0	0	0
0	1	1
1	0	1
1	1	1



4. Funciones Lógicas utilizadas en Electrónica Digital

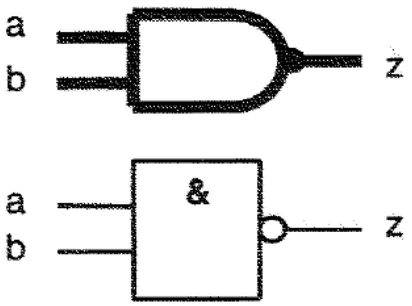
Las funciones anteriores son básicas, de tal forma que se puede obtener cualquier otra función como combinación de ellas.

A continuación se presentan una serie de funciones muy habituales en electrónica digital que combinan las funciones AND, OR y NOT

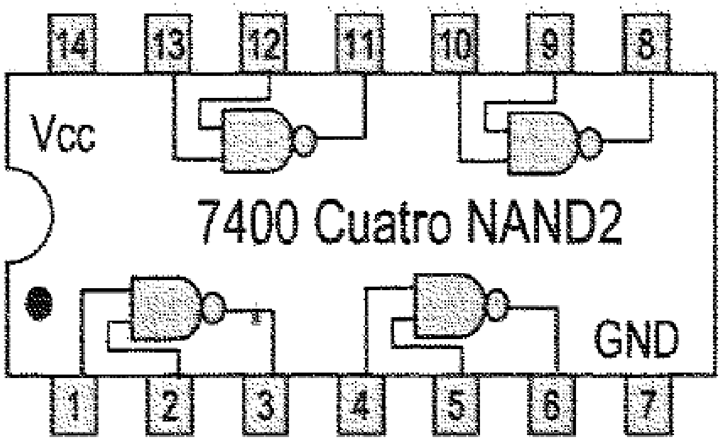
Función NAND

La función NAND corresponde a la operación producto lógico complementado:

$$z = \overline{a \cdot b}$$



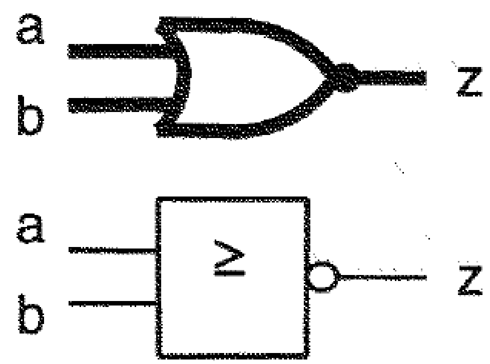
a	b	z
0	0	1
0	1	1
1	0	1
1	1	0



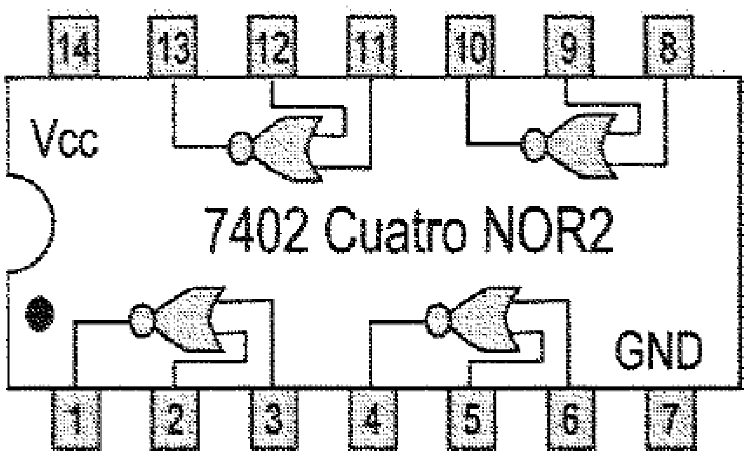
Función NOR

La función NOR corresponde a la operación suma lógica complementada:

$$z = \overline{a + b}$$



a	b	z
0	0	1
0	1	0
1	0	0
1	1	0



Tanto la función NAND como NOR se denominan puertas universales ya que son ellas las que pueden realizar cualquiera de las puertas básicas ya comentadas aplicando los teoremas de De Morgan,:

Función	**NOR** $z = \overline{a + b}$	**NAND** $z = \overline{a \cdot b}$
NOT $z = \overline{a}$	Si a = b $z = \overline{a}$	Si a = b $z = \overline{a}$
AND $z = a \cdot b$	$z = \overline{\overline{a \cdot b}} = \overline{\overline{a} + \overline{b}}$	
OR $z = a + b$	$z = \overline{\overline{a + b}} = \overline{\overline{a} \cdot \overline{b}}$	

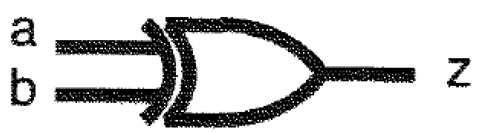
Función OR exclusiva XOR (⊕)

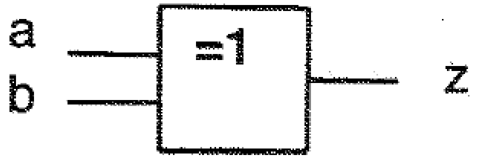
La puerta XOR realiza una función de solo dos variables:

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

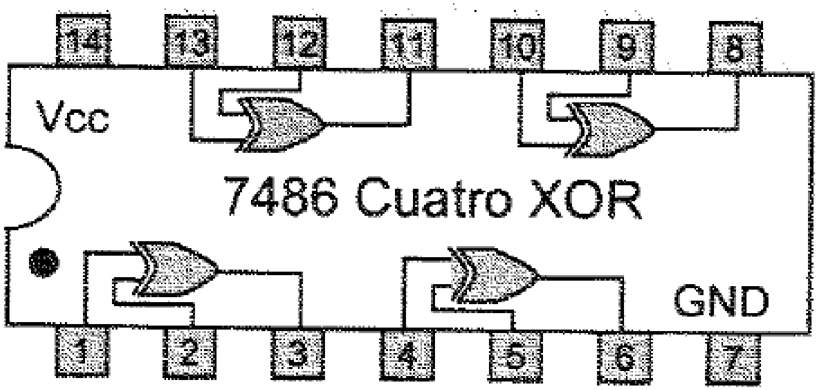
$$z = (a + b) \cdot (\overline{a} + \overline{b})$$

$$z = a \oplus b$$





a	b	z
0	0	0
0	1	1
1	0	1
1	1	0



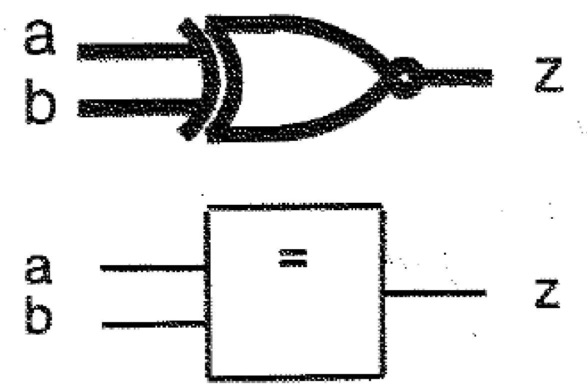
Función NOR exclusiva XNOR

La función XNOR corresponde a la operación XOR complementada:

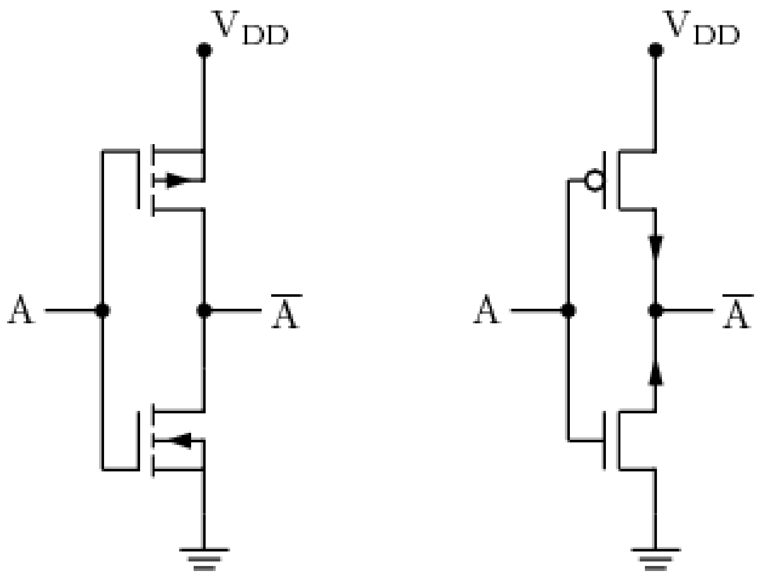
$$z = \overline{\bar{a} \cdot b + a \cdot \bar{b}}$$

$$z = \overline{a \oplus b}$$

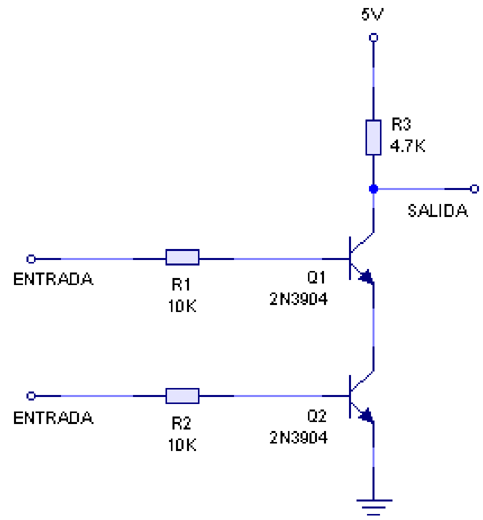
a	b	z
0	0	1
0	1	0
1	0	0
1	1	1

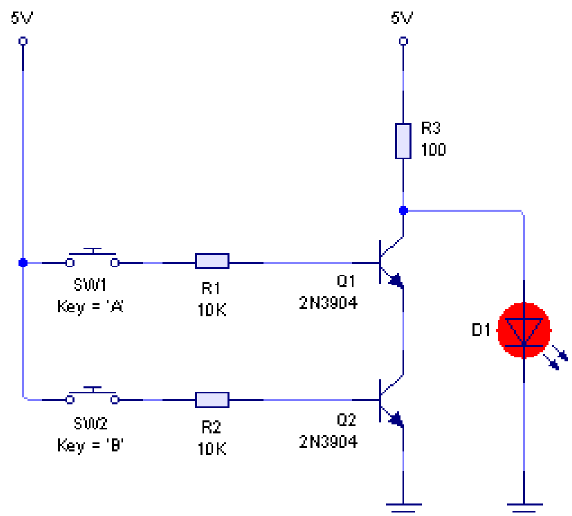


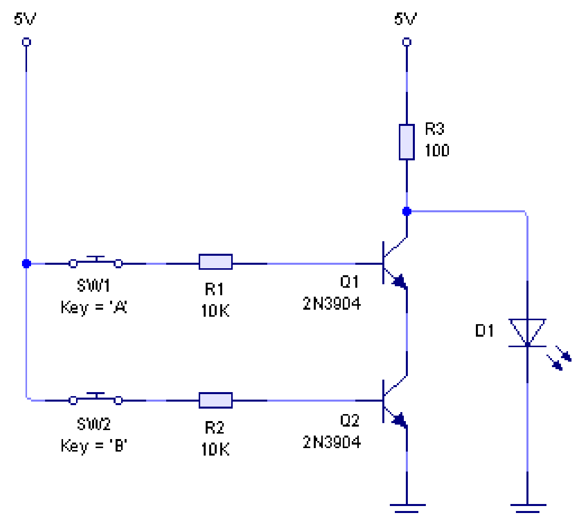
NOT



NAND

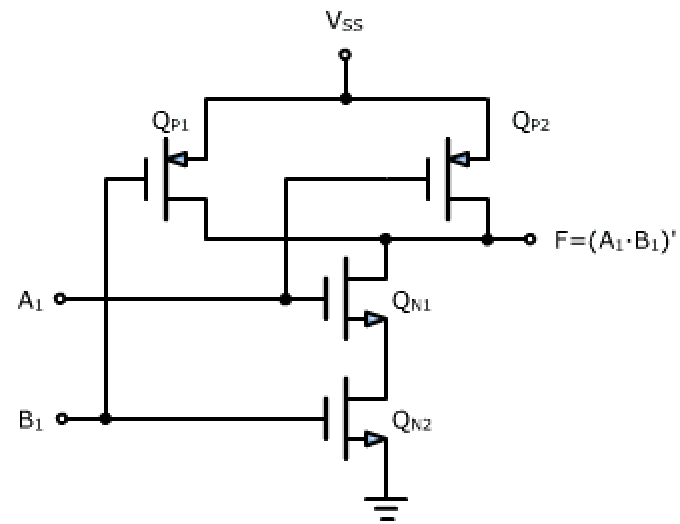

5V
R3
4.7K
SALIDA
ENTRADA
R1
10K
Q1
2N3904
ENTRADA
R2
10K
Q2
2N3904


5V
5V
R3
100
SW1
Key = 'A'
R1
10K
Q1
2N3904
D1
SW2
Key = 'B'
R2
10K
Q2
2N3904


5V
5V
R3
100
SW1
Key = 'A'
R1
10K
Q1
2N3904
D1
SW2
Key = 'B'
R2
10K
Q2
2N3904

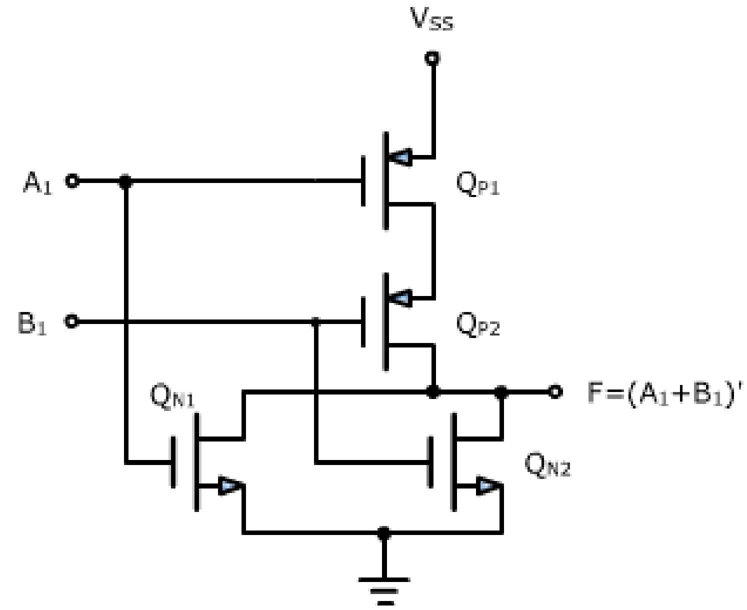
NAND con transistores MOS

A_1	B_1	Q_{P1}	Q_{P2}	Q_{N1}	Q_{N2}	F
0	0	ON	ON	OFF	OFF	1
0	1	ON	OFF	OFF	ON	1
1	0	OFF	ON	ON	OFF	1
1	1	OFF	OFF	ON	ON	0

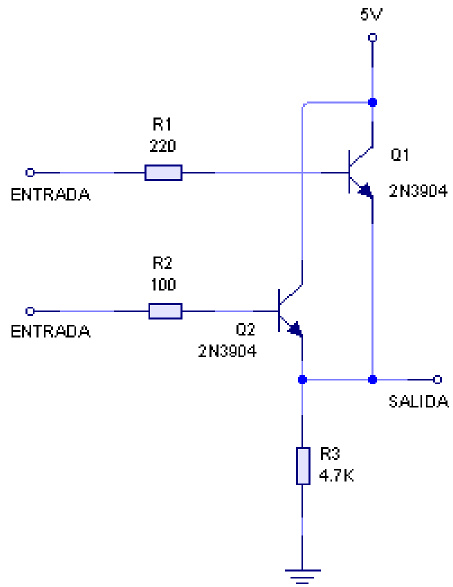


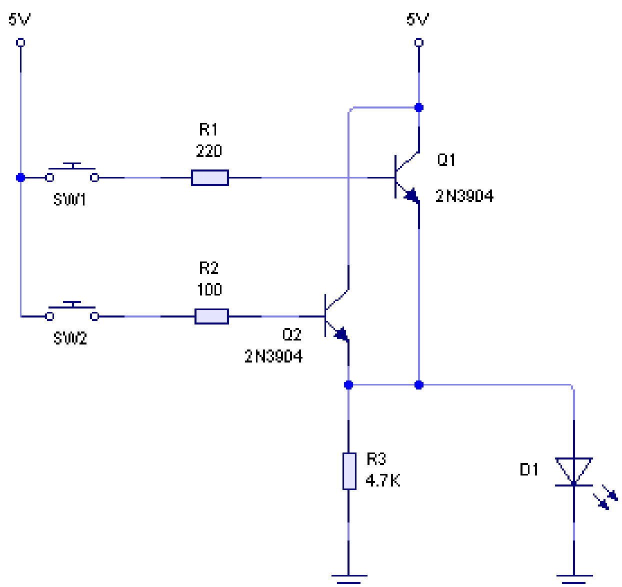
NOR con transistores MOS

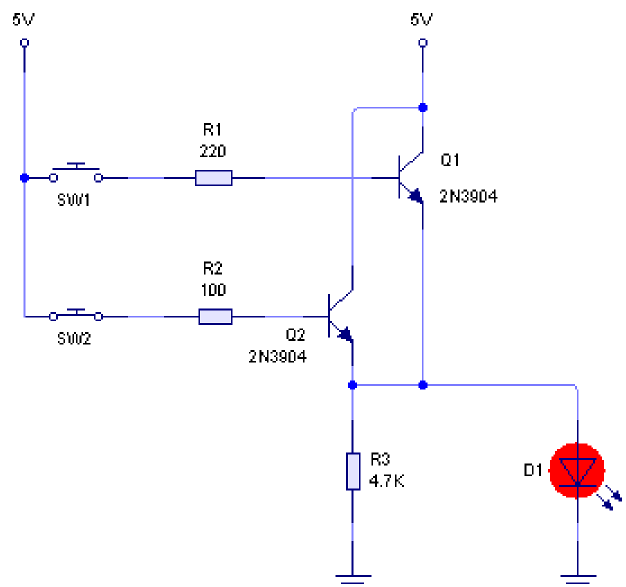
A_1	B_1	Q_{P1}	Q_{P2}	Q_{N1}	Q_{N2}	F
0	0	ON	ON	OFF	OFF	1
0	1	ON	OFF	OFF	ON	0
1	0	OFF	ON	ON	OFF	0
1	1	OFF	OFF<	ON	ON<	0



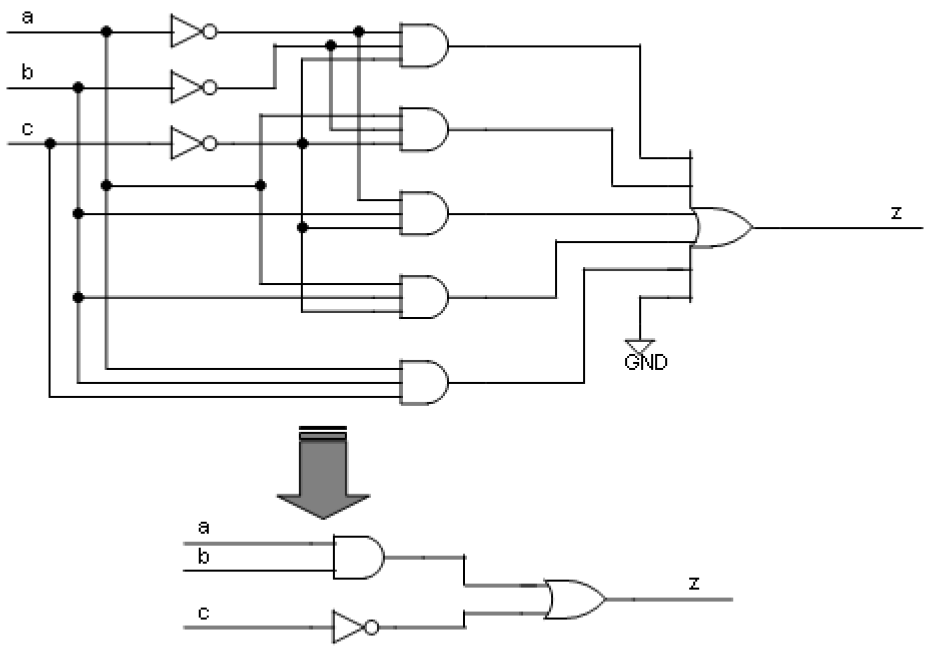
OR


5V
R1
220
Q1
ENTRADA
2N3904
R2
100
ENTRADA
Q2
2N3904
SALIDA
R3
4.7K

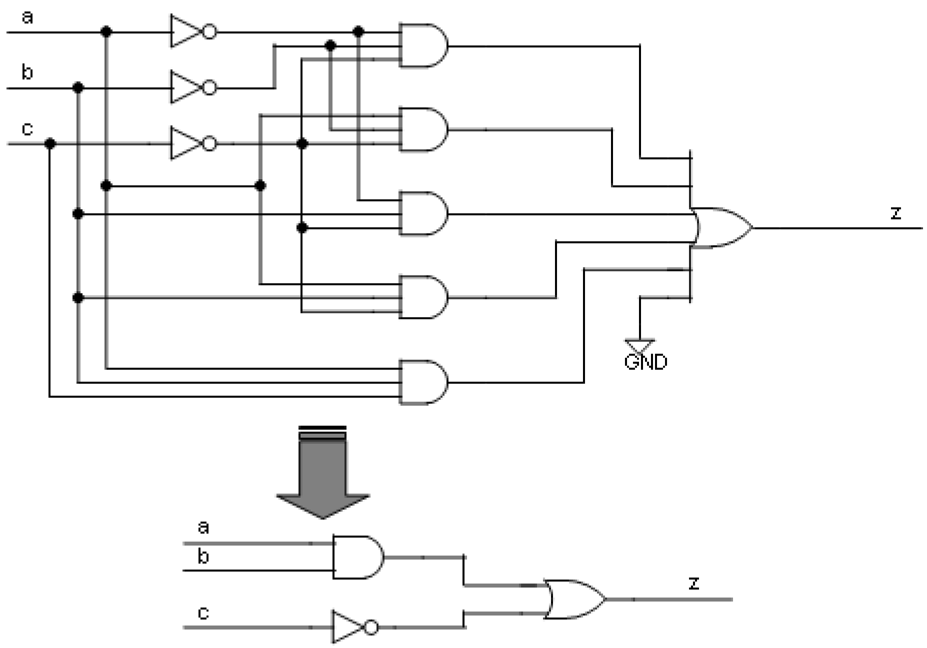

5V
5V
R1
220
Q1
SW1
2N3904
R2
100
SW2
Q2
2N3904
R3
4.7K
D1


5V
5V
R1
220
Q1
SW1
2N3904
R2
100
SW2
Q2
2N3904
R3
4.7K
D1

A menudo es posible realizar una simplificación de las funciones lógicas



A menudo es posible realizar una simplificación de las funciones lógicas



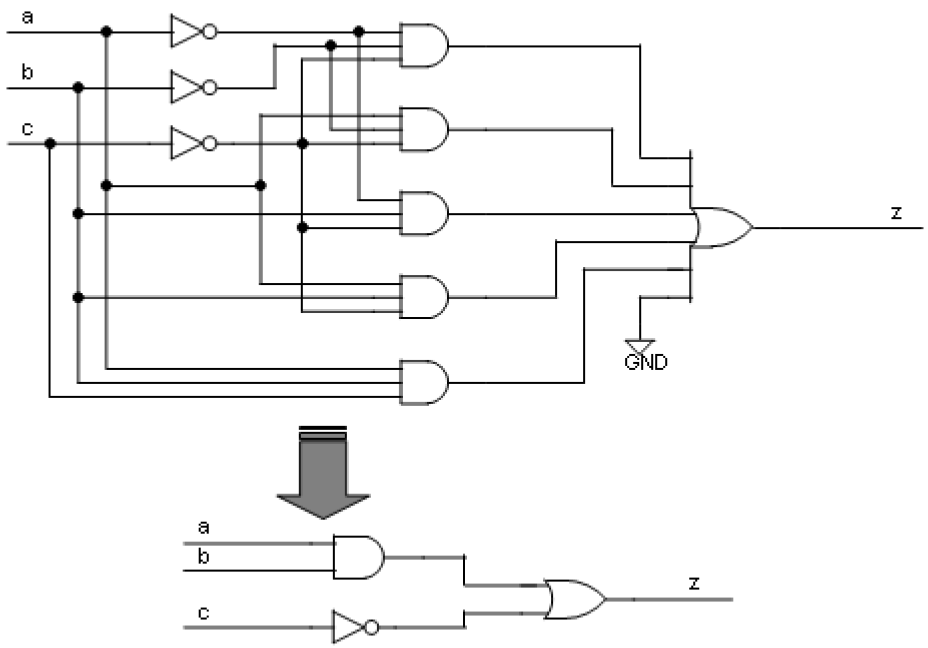
Hay dos opciones:

1.- Simplificación algebraica: utilizando los postulados y teoremas del álgebra de Boole.

$$abc\ldots + \bar{a}bc\ldots = (a+\bar{a})bc\ldots = bc\ldots$$

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

A menudo es posible realizar una simplificación de las funciones lógicas



Hay dos opciones:

1.- Simplificación algebraica: utilizando los postulados y teoremas del álgebra de Boole.
2.- Utilizando las tablas de Karnaugh.

6. Simplificación de funciones lógicas

Las tablas o mapas de Karnaugh es una herramienta gráfica para la representación de la tabla de la verdad de una función.

a \ b	0	1
0	0	1
1	2	3

a \ bc	00	01	11	10
0	0	1	3	2
1	4	5	7	6

ab \ cd	00	01	11	10
00	0	1	3	2
01	4	5	7	6
11	12	13	15	14
10	8	9	11	10

Son rectángulos formados por celdas que corresponden a cada combinación de las variables de la función.

Sus celdas están dispuesta de forma que dos celdas adyacentes únicamente difieren en el estado de una variable lógica.

Proceso de simplificación utilizando las tablas de Karnaugh:

1.- Tabla de la verdad de la función a simplificar.

a	b	c	f
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Proceso de simplificación utilizando las tablas de Karnaugh:

1.- Tabla de la verdad de la función a simplificar.
2.- Rellenamos la tabla de Karnaugh.

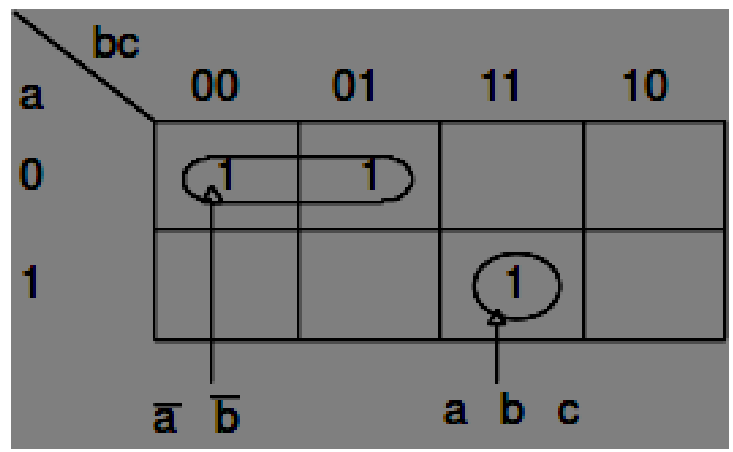
a	*b*	*c*	*f*
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

a \ bc	00	01	11	10
0	1	1		
1			1	

Proceso de simplificación utilizando las tablas de Karnaugh:

1.- Tabla de la verdad de la función a simplificar.
2.- Rellenamos la tabla de Karnaugh.
3.- Agrupamos según celdas de 2^n (0,1,2,4,8…… 2^n).
4.- Se obtiene la expresión para cada grupo.

a	*b*	*c*	*f*
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

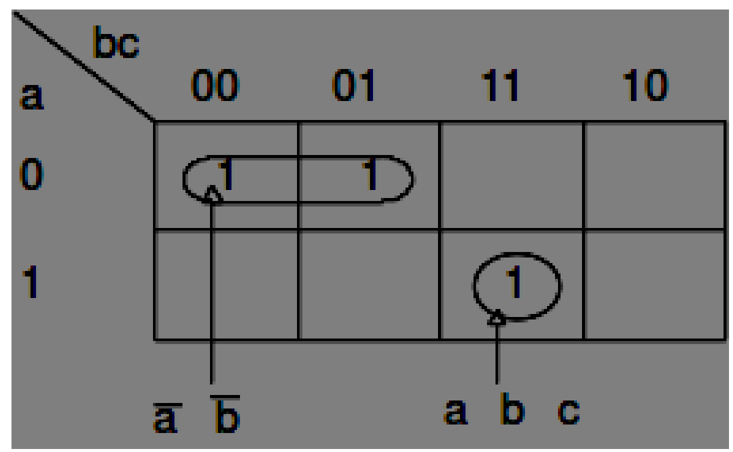


Proceso de simplificación utilizando las tablas de Karnaugh:

1.- Tabla de la verdad de la función a simplificar.
2.- Rellenamos la tabla de Karnaugh.
3.- Agrupamos según celdas de 2^n (0,1,2,4,8…… 2^n).
4.- Se obtiene la expresión para cada grupo.
5.- Crear la función lógica.

$$f(a,b,c) = \bar{a}\bar{b} + abc$$

a	b	c	f
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1



6. Simplificación de funciones lógicas

La expresión anterior se expresar:

1.- En agrupaciones de “1”: ***minterms***

$$f(a,b,c) = \overline{a}\overline{b} + \mathrm{abc} = \overline{a}\overline{b}\,(\mathrm{c} + \overline{c}) + \mathrm{abc} = \overline{a}\overline{b}\,\mathrm{c} + \overline{a}\overline{b}\overline{c} + \mathrm{abc}$$

$$f(a,b,c) = \sum(0,1,7)$$

6. Simplificación de funciones lógicas

La expresión anterior se expresar:

1.- En agrupaciones de “1”: ***minterms***
2.- En agrupaciones de “0”: ***maxterms***

a \ bc	00	01	11	10
0	1	1	0	0
1	0	0	1	0

$$f(a,b,c) = (\overline{a}+b+c)(\overline{a}+b+\overline{c})(a+\overline{b}+\overline{c})(a+\overline{b}+c)(\overline{a}+\overline{b}+c)$$

$$f(a,b,c) = \prod(2,3,4,5,6)$$

Ejemplo: $z = ab + bc$

Núm.	a	b	c	z		
0	0	0	0	0		
1	0	0	1	0		
2	0	1	0	0		
3	0	1	1	1		
4	1	0	0	0		
5	1	0	1	0		
6	1	1	0	1		
7	1	1	1	1		

$\bar{a}+\bar{b}+\bar{c}$

6. Simplificación de funciones lógicas

Ejemplo: $z = ab + bc$

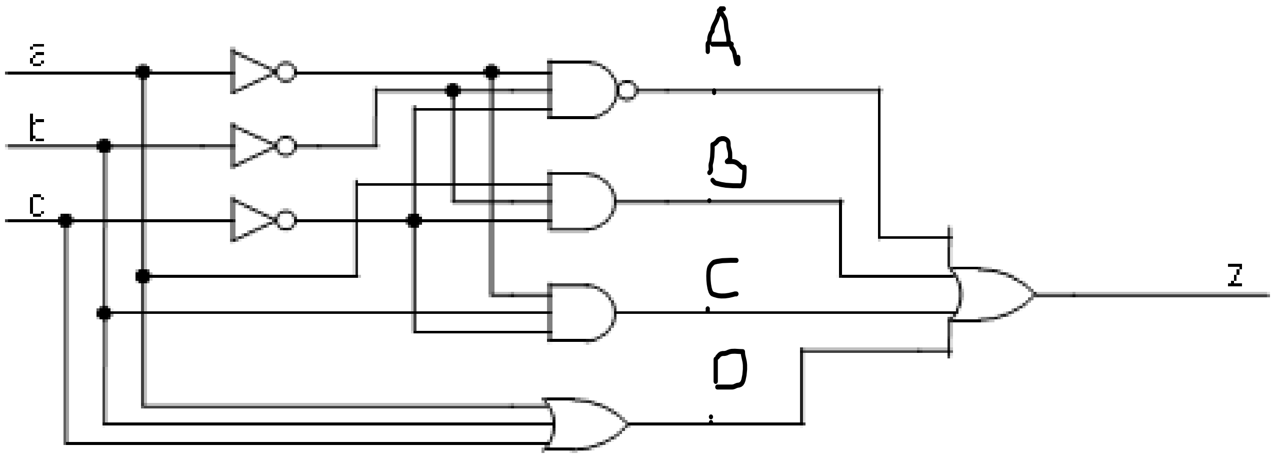
Núm.	a	b	c	z	Minterm	Maxterm
0	0	0	0	0	$\bar{a}\bar{b}\bar{c}$	$a+b+c$
1	0	0	1	0	$\bar{a}\bar{b}c$	$a+b+\bar{c}$
2	0	1	0	0	$\bar{a}b\bar{c}$	$a+\bar{b}+c$
3	0	1	1	1	$\bar{a}bc$	$a+\bar{b}+\bar{c}$
4	1	0	0	0	$a\bar{b}\bar{c}$	$\bar{a}+b+c$
5	1	0	1	0	$a\bar{b}c$	$\bar{a}+b+\bar{c}$
6	1	1	0	1	$ab\bar{c}$	$\bar{a}+\bar{b}+c$
7	1	1	1	1	abc	$\bar{a}+\bar{b}+\bar{c}$

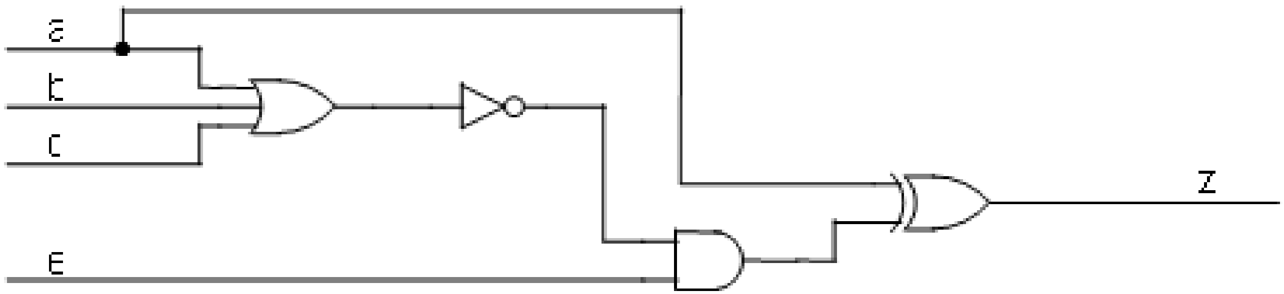
Ejemplo: $z = ab + bc$

Núm.	a	b	c	z	Minterm	Maxterm
0	0	0	0	0	$\bar{a}\bar{b}\bar{c}$	$a + b + c$
1	0	0	1	0	$\bar{a}\bar{b}c$	$a + b + \bar{c}$
2	0	1	0	0	$\bar{a}b\bar{c}$	$a + \bar{b} + c$
3	0	1	1	1	$\bar{a}bc$	$a + \bar{b} + \bar{c}$
4	1	0	0	0	$a\bar{b}\bar{c}$	$\bar{a} + b + c$
5	1	0	1	0	$a\bar{b}c$	$\bar{a} + b + \bar{c}$
6	1	1	0	1	$ab\bar{c}$	$\bar{a} + \bar{b} + c$
7	1	1	1	1	abc	$\bar{a} + \bar{b} + \bar{c}$

$$z = f(a, b, c) = \sum (3,6,7)$$

$$z = f(a, b, c) = \prod (0,1,2,4,5)$$


a
b
c
A
B
C
D
z


a
b
c
e
z