Tema 3. Álgebra de Conmutación

Objetivos

- Indicar los postulados o axiomas del Álgebra de Boole.
- Teoremas del Álgebra de Boole.
- Definir el concepto de función lógica o función de conmutación.
- Definir el concepto de puerta lógica

Tema 3. Álgebra de Conmutación

Contenido

- Postulados del Algebra de Boole
- Funciones Lógicas
- Funciones Lógicas básicas derivadas del Álgebra de Conmutación: NOT, OR, AND
- Otras Funciones Lógicas importantes: NAND, NOR, XOR, XNOR

Postulados o axiomas del Álgebra de Boole

- Un Álgebra es una estructura matemática que comprende un conjunto de elementos y un conjunto de operaciones u operadores, que actúan sobre dichos elementos.
- Los postulados o axiomas determinan como se realizan dichas operaciones.
- Los postulados no se demuestran y permiten deducir los teoremas y propiedades de dicha estructura.
- Se suele usar los axiomas propuestos por Huntington en 1904 para definir el Álgebra de Boole.
- Sea un conjunto de elementos llamado B en el que se puede establecer una relación de equivalencia que denotaremos con el símbolo = y para la que se verifica el principio de sustitución.
  - $\exists\, x,y \in B \,/\, x = y$
  - Es decir, en cualquier expresión en la que aparezca ***x*** se podrá sustituir por ***y***, y viceversa.

Postulados o axiomas del Álgebra de Boole

- Postulados.

1. **Leyes de composición interna**.

   + (operador OR o suma lógica) • (operador AND o producto lógico)

   $\forall x,y \in B \quad \Rightarrow x + y \in B \qquad x \bullet y \in B$

2. **Conmutatividad de las leyes de composición interna.**

   $\forall x,y \in B \quad \Rightarrow x + y = y + x \qquad x \bullet y = y \bullet x$

3. **Elementos neutros**.

   $\exists 0 \in B \,/\, \forall x \in B,\ x + 0 = 0 + x = x$

   $\exists 1 \in B \,/\, \forall x \in B,\ x \bullet 1 = 1 \bullet x = x$

4. **Distributividad de las leyes de composición interna.**

   $\forall x,y,z \in B \quad x + (y \bullet z) = (x+y) \bullet (x+z)$

   $\forall x,y,z \in B \quad x \bullet (y + z) = (x \bullet y) + (x \bullet z)$

5. **Elemento opuesto**.

   $\forall x \in B \quad \exists \overline{x} \in B \,/\, x + \overline{x} = 1 \qquad x \bullet \overline{x} = 0$

   $\overline{x}$ *se denomina complemento de x*

6. **Número de elementos.**

   $\exists x,y \in B \,/\, x \neq y$

   En el conjunto B existen por lo menos 2 elementos.

Teoremas del Álgebra de Boole.

- ***Para cada teorema existen dos enunciados.***
  1. No es necesario demostrar ambos enunciados.
  2. Se demuestra un enunciado y el otro queda probado por el principio de dualidad (Ver principio de dualidad en la siguiente transparencia para B_2).

Teorema 1. $\forall x \in B_2$ $\quad x + 1 = 1 \quad x \bullet 0 = 0$

Teorema 2. Idempotencia. $\forall x \in B_2$ $\quad x + x = x \quad x \bullet x = x$

Teorema 3. Involución. $\forall X \in B_2$ $\quad \overline{\overline{X}} = X$

Teorema 4. Absorción. $\forall x,y \in B_2$ $\quad x + x \bullet y = x \quad x \bullet (x+y) = x$

Teorema 5. $\forall x,y,z \in B_2$ $\quad x + [(x \bullet y) \bullet z] = x \quad x \bullet [(x + y) + z] = x$

Teorema 6. Asociativa.

$$\forall x,y,z \in B_2 \qquad x + (y + z) = (x + y) + z \qquad x \bullet (y \bullet z) = (x \bullet y) \bullet z$$

Teorema 7. $\forall x, y \in B_2$ $\quad x + \overline{x} \cdot y = x + y \quad x \cdot (\overline{x} + y) = x \cdot y$

Teorema 8. Leyes de De Morgan.

$$\forall x, y \in B_2 \qquad \overline{x + y} = \overline{x} \cdot \overline{y} \qquad \overline{x \cdot y} = \overline{x} + \overline{y}$$

Álgebra de Conmutación. Principio de Dualidad.

- El Álgebra de Conmutación (A. C.) es un Álgebra de Boole que emplea solamente dos elementos. B = {0, 1}. lo representamos B_2
- Las operaciones quedan definidas de la siguiente forma:

OR

+	0 1
0	0 1
1	1 1

AND

•	0 1
0	0 0
1	0 1

NOT

-	
0	1
1	0

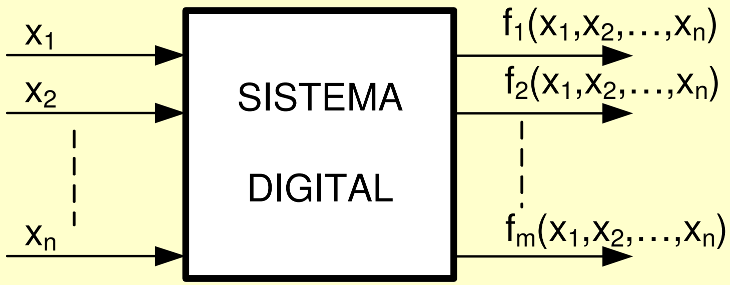
- Se comprueba fácilmente que los operadores cumplen los postulados de Huntington.
- Los elementos 0 y 1 de B_2 corresponde a los dos valores binarios usados en los sistemas digitales.
- ***Principio de dualidad.***
  - *Si en una igualdad se sustituye* ***0*** *por* ***1***, ***+*** *por* *******, *y viceversa, en todos los lugares que aparezcan, se obtiene otra igualdad.*
  - Si se parte del enunciado a) de un postulado y se aplica el principio de dualidad se obtiene el enunciado b), y viceversa.

Funciones Lógicas o de Conmutación
Definición

Una función lógica f, que representaremos $f(x_1, x_2, ..., x_n)$, donde $x_1, x_2, ..., x_n$ son las variables de entrada y f la de salida se define:

> *Toda función cuyos valores de entrada y salida solamente pueden ser los elementos del álgebra de conmutación, es decir 0 y 1, y están relacionados mediante los operadores del Álgebra de Conmutación {+, •, ¯}.*
>
> *Las variables de entrada y salida se denominan variables lógicas.*

- Las señales de entrada y salida de un sistema digital solamente pueden tomar los valores 0 y 1, por lo que:
  - Se podrán representar mediante variables lógicas.
  - Las señales de salida se podrán expresar matemáticamente a partir de las de entrada mediante una función lógica.
  - El álgebra de conmutación permitirá el análisis y diseño de los sistemas digitales.



Funciones Lógicas básicas

- Las derivadas de los operadores del Álgebra de Conmutación:
  - ✓ Función lógica AND.
  - ✓ Función lógica OR.
  - ✓ Función lógica NOT.
- Otras funciones básicas:
  - ✓ Función lógica NAND. (puerta universal)
  - ✓ Función lógica NOR. (puerta universal)
  - ✓ Función lógica XOR.
  - ✓ Función lógica XNOR
- **Puerta lógica.**
  - ✓Circuito digital que implementa un operador del álgebra de conmutación o una función lógica sencilla.
  - ✓Actualmente se implementan mediante Circuitos Integrados Digitales.

Función lógica NOT

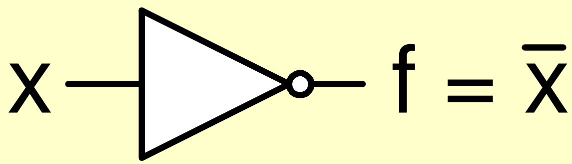
Función lógica NOT.

- Corresponde al operador complemento o NOT (¯) del A.C.
- Tiene una única entrada.
- **Expresión lógica**: $\mathbf{f(x) = \overline{x}}$
  - ✓Donde ***x*** es la variable de entrada y ***f*** la de salida
- **Tabla de verdad**:

x	*f*
0	*1*
1	*0*

✓ El valor de la función lógica NOT es el opuesto del de la variable de entrada.

- **Puerta lógica**:



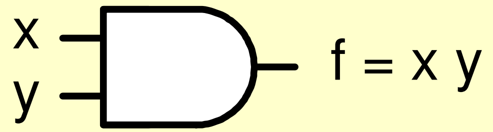
Función lógica AND

Función lógica AND.

- Corresponde al operador Producto Lógico o AND (•) del A.C.
- **Expresión lógica**: $f(x, y) = x \bullet y$
  - ✓Donde ***x*** e ***y*** son las variables de entrada y ***f*** la de salida
- **Tabla de verdad**:

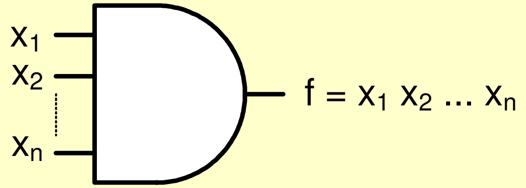
x	*y*	*f*
0	*0*	*0*
0	*1*	*0*
1	*0*	*0*
1	*1*	*1*

- **Puerta lógica**:



Extensión a más de dos entradas.

- Por ser conmutativa y asociativa según los postulados y teoremas del Álgebra de Boole, se puede hacer la AND lógica de más de dos entradas.



- $f(x_1, x_2, \ldots, x_n) = x_1\, x_2 \ldots x_n$

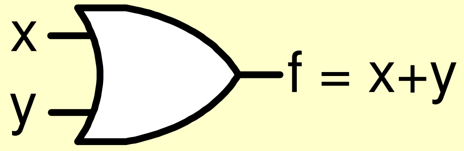
Función lógica OR

Función lógica OR

- Corresponde al operador Suma Lógica u OR (+) del A.C.
- **Expresión lógica**: $f(x, y) = x + y$
  - ✓Donde ***x*** e ***y*** son las variables de entrada y ***f*** la de salida
- **Tabla de verdad**:

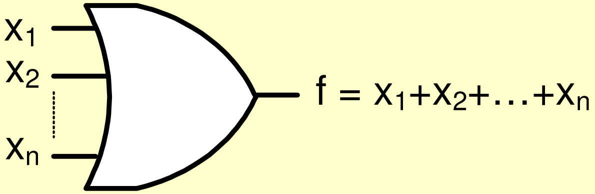
x	*y*	*f*
0	*0*	*0*
0	*1*	*1*
1	*0*	*1*
1	*1*	*1*

- **Puerta lógica**:



Extensión a más de dos entradas.

- Por ser conmutativa y asociativa según los postulados y teoremas del Álgebra de Boole, se puede hacer la OR lógica de más de dos entradas.
- $f(x_1, x_2,..., x_n) = x_1 + x_2 +...+ x_n$



Función lógica NAND

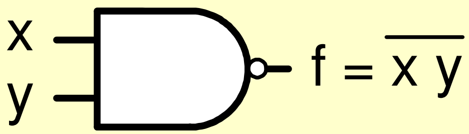
Las restantes funciones lógicas básicas se obtienen combinando varios operadores del Álgebra de Boole.

Función lógica NAND.

- Corresponde a las siglas de NOT AND. Por tanto, es el complemento de la AND.
- **Expresión lógica**: $f(x,y) = \overline{x\,y}$
  - ✓Donde ***x*** e ***y*** son las variables de entrada y ***f*** la de salida
- **Tabla de verdad**:

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

- **Puerta lógica**:



Función lógica NAND

Extensión a más de dos entradas.

- La función lógica NAND es conmutativa, pero no es asociativa:

$$\overline{x\,y} = \overline{y\,x} \qquad ; \qquad \overline{\overline{(x\,y)}\,z} \neq \overline{x\,\overline{(y\,z)}} \neq \overline{x\,y\,z}$$

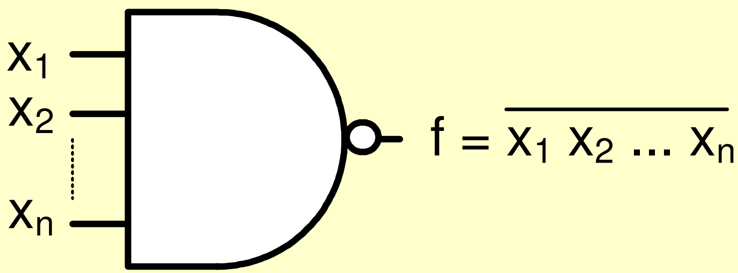
- No obstante, por ser la AND complementada se puede obtener la función lógica NAND de más de 2 variables de entrada:

1. Se hace la AND lógica de todas las variables de entrada.
2. Se complementa la salida de la función lógica AND.

$$f(x_1, x_2, \ldots, x_n) = x_1 x_2 \ldots x_n$$

✓**La función lógica NAND de n entradas es 0 si todas las variables de entrada son 1.**

✓**De lo contrario, si alguna de las variables de entrada es 0 el valor de salida es 1.**



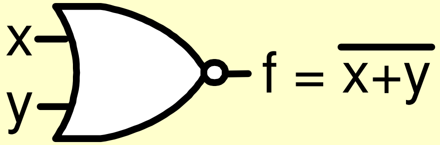
Función lógica NOR

Función lógica NOR.

- Corresponde a las siglas de NOT OR. Por tanto, es el complemento de la OR.
- **Expresión lógica**: $\mathbf{f(x,y) = \overline{x + y}}$
  - ✓Donde ***x*** e ***y*** son las variables de entrada y ***f*** la de salida
- **Tabla de verdad**:

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

- **Puerta lógica**:



Función lógica NOR

Extensión a más de dos entradas.

- La función lógica NOR es conmutativa, pero no es asociativa:

$$\overline{x+y} = \overline{y+x} \quad ; \quad \overline{\overline{(x+y)}+z} \neq \overline{x+\overline{(y+z)}} \neq \overline{x+y+z}$$

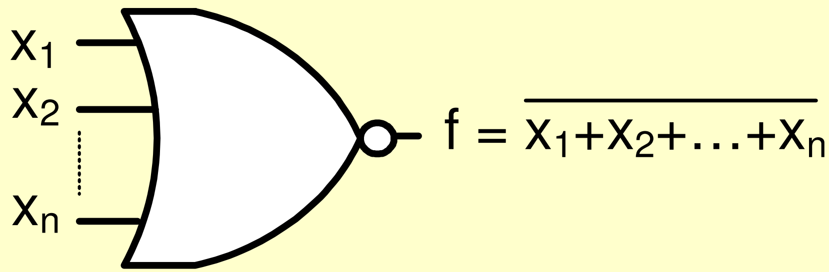
- No obstante, por ser la OR complementada se puede obtener la función lógica NOR de más de 2 variables de entrada:

  1. Se hace la OR lógica de todas las variables de entrada.
  2. Se complementa la salida de la función lógica OR.

$$f(x_1, x_2, \ldots, x_n) = \overline{x_1 + x_2 + \ldots + x_n}$$

✓**La función lógica NOR de n entradas es 1 si todas las variables de entrada son 0.**

✓**De lo contrario, si alguna de las variables de entrada es 1 el valor de salida es 0.**



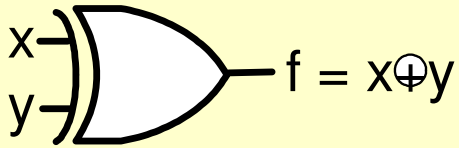
Función lógica XOR

Función lógica XOR u OR EXCLUSIVA

- El valor de salida es 1 si hay un número impar de 1 en sus variables de entrada.
- Se representa mediante el símbolo $\oplus$
- **Expresión lógica**: $\mathbf{f(x,y) = x \oplus y = \bar{x}\,y + x\,\bar{y}}$
  - ✓Donde ***x*** e ***y*** son las variables de entrada y ***f*** la de salida
- **Tabla de verdad**:

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

- **Puerta lógica**:



Extensión a más de dos entradas.

- La función lógica XOR es conmutativa y asociativa, por lo que se puede hacer directamente la XOR de más de dos entradas.
- $F(x,y,z) = x \oplus y \oplus z$

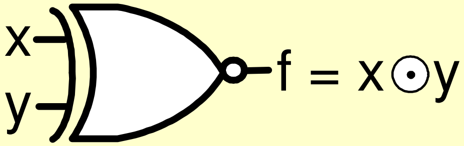
Función lógica XNOR

Función lógica XNOR o NOR EXCLUSIVA

- Se definirá solamente para dos variables de entrada.
- Se representa mediante el símbolo $\odot$
- **Expresión lógica**: $f(x,y) = x \odot y = \overline{x}\,\overline{y} + x\,y$
  - ✓Donde ***x*** e ***y*** son las variables de entrada y ***f*** la de salida
- **Tabla de verdad**:

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

- **Puerta lógica**:



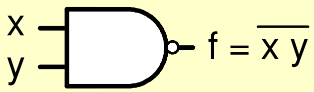
Puertas universales: NAND y NOR

Las puertas NAND y NOR se les añade el adjetivo de universales porque es posible implementar cualquier función lógica utilizando solamente puertas NAND o puertas NOR:

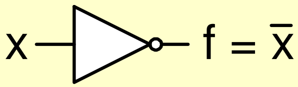
- La manera más fácil de demostrarlo es consiguiendo cualquier puerta NOT, OR y AND utilizando puertas NAND o puertas NOR.
- Se verá el caso para puertas NAND

Puerta universal NAND

Implementación de las funciones lógicas básicas mediante puertas NAND.



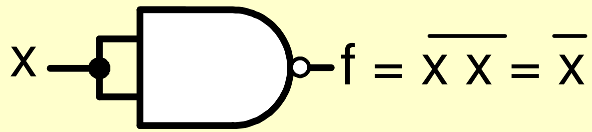
NOT



✓Dos opciones:

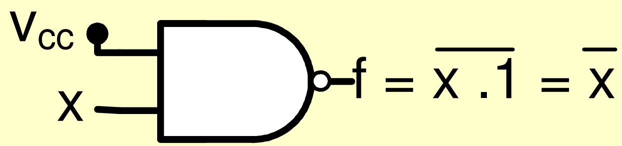
a) Por el Teorema de Idempotencia **x • x = x**

$$\mathbf{f = \overline{x} = \overline{x \cdot x}}$$



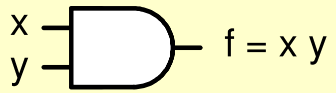
b) Por el Postulado del Elemento neutro **x • 1 = x**

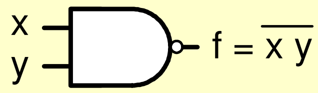
$$\mathbf{f = \overline{x} = \overline{x \cdot 1}}$$



Puerta universal NAND

AND

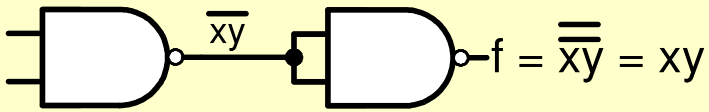




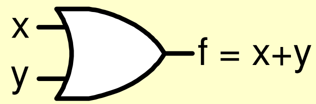
- Una es el complemento de la otra.
- Por el Teorema de Involución

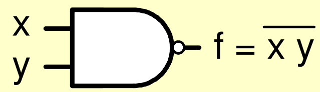
$$\overline{\overline{\mathbf{x}}} = \mathbf{x}$$

$$\mathbf{f = x \cdot y = \overline{\overline{x \cdot y}}}$$

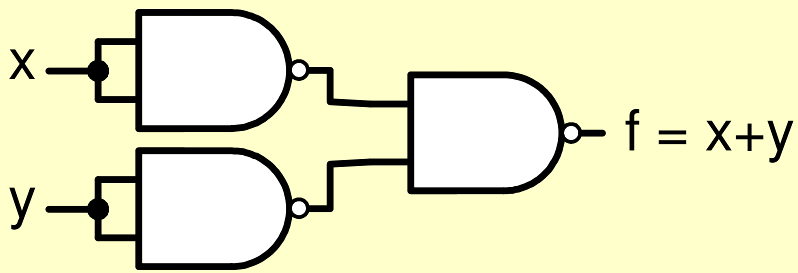


OR





$$\mathbf{f = x + y = \overline{\overline{x + y}} = \overline{\overline{x} \cdot \overline{y}}}$$



Bibliografía detallada
Álgebra de Conmutación

- **Las diapositivas se han confeccionado utilizando como fuente:**
  - "Diseño Lógico". A. Lloris, A. Prieto. Mc-Graw Hill. 1996.
    Apartados: 1.2, 1.3, 1.4, 1.5, 1.7