**Unidad IV: Álgebra de Boole.**
Introducción a la lógica. Postulados. Teoremas. Funciones booleanas.
Simplificación de funciones booleanas. Operaciones y símbolos: And, Or, Not, Nand, Nor, Or exclusivos. Formas canónicas. Mintérminos y Máxterminos.

## Lógica:

Lógica es la parte del razonamiento humano que nos dice que una determinada proposición o sentencia de asignación es cierta (clasificándose entonces como VERDADERA) si se cumplen determinadas condiciones (si ellas no se cumplen, será FALSA). Cuando se combinan varias proposiciones se forman funciones lógicas o proposicionales. Los procesos que realizan los circuitos digitales pueden expresarse como funciones proposicionales o lógicas, teniendo en cuenta sus dos estados lógicos.

Ejemplo: *Iré al restaurante sólo si tengo tiempo y tengo dinero.*

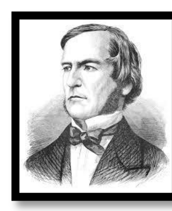
proposición básica: *Iré al restaurante*

condiciones de las que depende la proposición: *si tengo tiempo y tengo dinero*

La proposición básica será VERDADERA sólo si son VERDADERAS las otras dos.

## ¿Qué es el Álgebra de BOOLE?

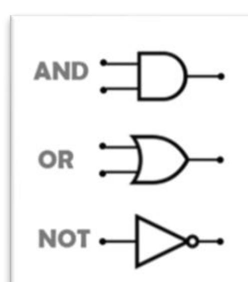
George Boole, matemático y lógico irlandés desarrolló (hacia 1850) un sistema matemático para formular proposiciones lógicas con símbolos, de manera que los problemas puedan formularse y resolverse de forma similar a como se hace en el álgebra ordinaria. El álgebra de Boole se aplica en el diseño y el análisis de los sistemas digitales.



## Circuitos y Puertas Lógicas

Existen varios tipos de **circuitos** lógicos que son los elementos básicos que constituyen los bloques sobre los que se construyen los sistemas digitales más complejos, como por ejemplo una computadora. Podemos afirmar que un microprocesador consta de cientos de miles de puertas lógicas.

Un circuito que realiza una operación lógica determinada se denomina **puerta lógica (**o también, **compuerta lógica)**.




Las tres operaciones lógicas básicas son: NOT, AND y OR. En la figura de la derecha observamos los símbolos lógicos correspondientes a dichas operaciones. (Los símbolos lógicos usados para representar las puertas lógicas en el presente material de estudio se corresponden con el estándar ANSI/IEEE 91-1984.)

Las líneas a la izquierda de la compuerta representan las **entradas** del circuito y la línea a su derecha, su **salida**. Las puertas AND y OR pueden tener cualquier número de entradas.

## Operación NOT:

La operación ***NOT*** (operación de inversión o complementación) cambia de un nivel lógico al nivel lógico opuesto, esto quiere decir, en términos de bits, cambia un 1 por un 0, y un 0 por 1. La operación NOT se implementa mediante un circuito lógico conocido como ***inversor***.
El comportamiento de un inversor puede representarse mediante la siguiente **tabla de verdad** (tabla que muestra la salida para cada posible entrada en términos de niveles y bits correspondientes):

| Entrada | Salida |
|---|---|
| 0 | 1 |
| 1 | 0 |

Expresión booleana: si a la entrada tenemos la variable $\mathbf{A}$ **,** a la salida tenemos $\overline{\mathbf{A}}$ **,(***A invertido* o *negado*, o bien, *complemento de A*).

## Operación AND:

La operación AND realiza la operación que se conoce como multiplicación lógica y se implementa mediante un circuito lógico conocido como *puerta AND.*
La operación ***AND*** genera un nivel ALTO sólo cuando todas las entradas están a nivel ALTO, es decir, que a nivel de bits, tendremos un 1 a la salida sólo cuando todas sus entradas estén en 1.

El comportamiento de una AND puede representarse mediante la siguiente **tabla de verdad** (redefinimos a la tabla de verdad como una tabla en la que se enumeran todas las combinaciones de entrada con las correspondientes salidas)**:**

| Entradas | | Salida |
|---|---|---|
| **A** | **B** | **X** |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

El número total de posibles combinaciones de entradas binarias a una puerta está determinado por la siguiente fórmula:
$N = 2^n$

(*N* = número de posibles combinaciones de entrada; *n=* número de variables de entrada.)

En este caso la expresión booleana es: $X=AB$

## Operación OR:

La operación OR realiza la operación que se conoce como **suma lógica o booleana** y se implementa mediante un circuito lógico conocido como *puerta OR.*

La operación ***OR*** genera un nivel ALTO cuando una o más entradas están a nivel ALTO.

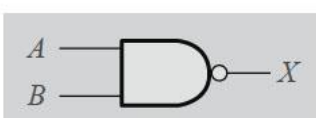
El comportamiento de una OR puede representarse mediante la siguiente **tabla de verdad**:

| Entradas | | Salida |
|---|---|---|
| **A** | **B** | **X** |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

En este caso la expresión booleana es: $X = A+B$

## Otras compuertas: NAND, NOR, XOR:

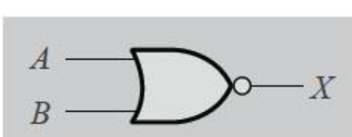
La puerta **NAND** es un elemento lógico popular, debido a que se puede utilizar como una puerta universal, es decir, las puertas NAND se pueden combinar para implementar las operaciones de las puertas AND, OR y del inversor.




| Entradas | | Salida |
|---|---|---|
| *A* | *B* | *X* |
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

1 = ALTO, 0 = BAJO

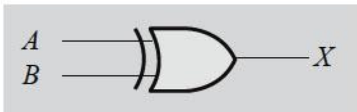
La puerta **NOR**, al igual que la puerta NAND, es un útil elemento lógico porque también se puede emplear como una puerta universal.




| Entradas | | Salida |
|---|---|---|
| *A* | *B* | *X* |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

1 = ALTO, 0 = BAJO

La salida de una puerta **OR−exclusiva** se pone a nivel ALTO sólo cuando las dos entradas están a niveles lógicos opuestos. La salida de una XOR equivale a la expresión booleana = $\mathbf{\overline{A}B + A\overline{B}}$




| Entradas *A* | *B* | Salida *X* |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

## Álgebra de BOOLE Y Simplificación Lógica

**VARIABLE:** es un símbolo (normalmente una letra mayúscula en cursiva) que se utiliza para representar magnitudes lógicas. Cualquier variable puede tener un valor de 0 o de 1.

**COMPLEMENTO:** es el inverso de la variable y se indica mediante una barra encima de la misma. Por ejemplo, el complemento de la variable $\boldsymbol{A}$ es $\overline{\mathbf{A}}$ . Si $\boldsymbol{A = 1}$, entonces $\overline{\boldsymbol{A}} \boldsymbol{= 0}$. Si $\boldsymbol{A = 0}$, entonces $\overline{\boldsymbol{A}} \boldsymbol{= 1}$. El complemento de la variable *A* se lee “no *A*” o “*A* barra”. En ocasiones, se emplea un apóstrofe en lugar de la barra para indicar el complemento de una variable; por ejemplo *B*’ indica el complemento de *B*.

**LITERAL:** es una variable o el complemento de una variable.

### Reglas básicas de la suma lógica o booleana (equivalente a la operación OR):

0 + 0 = 0

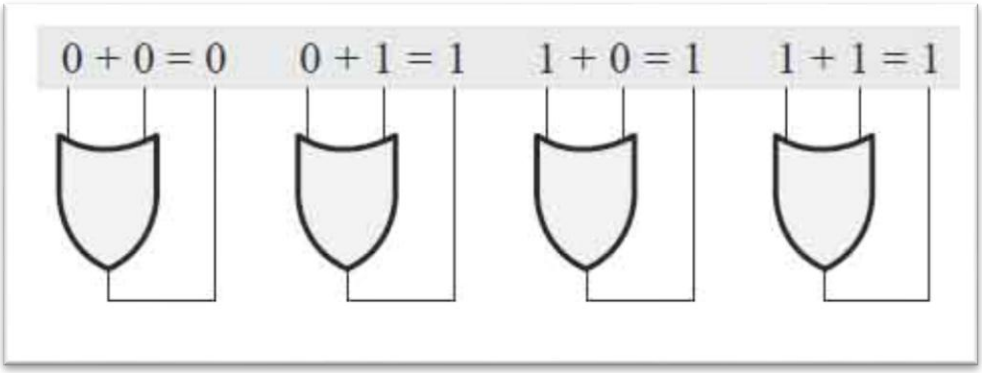
0 + 1 = 1

1 + 0 = 1

1 + 1 = 1

La suma booleana difiere de la suma binaria en el caso en que se suman dos 1s. En la suma booleana no existe acarreo.

Relación de la **suma lógica o booleana** con la puerta OR:




**Término suma:**

En el álgebra de Boole, un ***término suma*** es una suma de literales. En los circuitos lógicos, un término suma se obtiene mediante una operación OR, sin que exista ninguna operación AND en la expresión. Algunos ejemplos de términos suma son: $\mathbf{A+B}$, $\mathbf{A+\overline{B}}$, $\mathbf{A+B+\overline{C}}$, $\mathbf{A+B+C+\overline{D}}$, etc.
Un término suma es igual a 1 cuando uno o más de los literales del término es 1. Un término suma es igual a 0 sólo si cada uno de los literales es igual a 0.

**Reglas básicas de la multiplicación lógica o booleana (equivalente a la operación AND):**

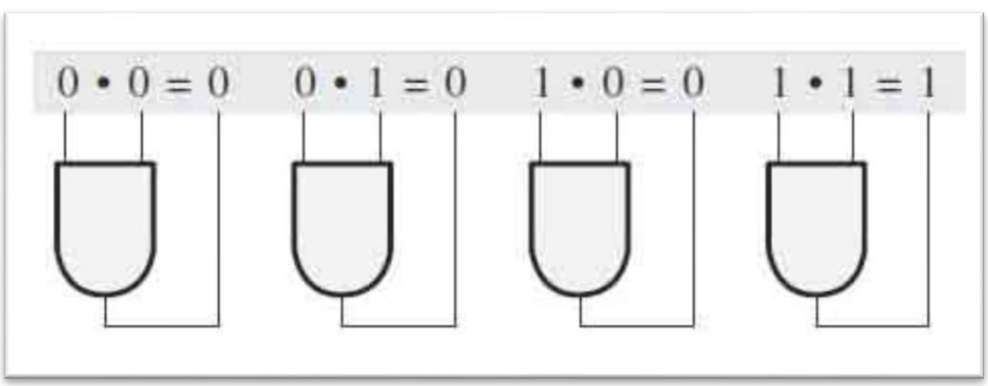
0 . 0 = 0

0 . 1 = 0

1 . 0 = 0

1 . 1 = 1

Relación de la **multiplicación lógica o booleana** con la puerta AND:




**Término producto:**

En el álgebra de Boole, un ***término producto*** es un producto de literales. En los circuitos lógicos, un término suma se obtiene mediante una operación AND, sin que existe ninguna operación OR en la expresión.
Algunos ejemplos de términos suma son $\mathbf{AB}$, $\mathbf{A\overline{B}}$, $\mathbf{ABC\overline{D}}$ , etc.

Un término producto es igual a 1 sólo si cada uno de los literales del término es 1. Un término producto es igual a 0 cuando uno o más de los literales son iguales a 0.

## Leyes y Reglas del Álgebra de Boole

Mediante ellas es posible modificar y simplificar expresiones booleanas.

## Leyes del Álgebra de Boole

Las leyes básicas del álgebra de Boole (las leyes conmutativas de la suma y la multiplicación, y las leyes asociativas de la suma y la multiplicación y la ley distributiva) son las mismas que las del álgebra ordinaria.
Cada una de las leyes se ilustra con dos o tres variables, pero el número de variables puede ser mayor.

| | |
|---|---|
| Ley **CONMUTATIVA**: | A + B = B + A<br>AB = BA |
| Ley **ASOCIATIVA**: | ( A + B ) + C = A + ( B + C )<br>(AB) C = A (BC) |
| Ley **DISTRIBUTIVA**: | A (B+C) = AB + AC<br>También es posible sacar factor común, o sea: AB + AC = A (B+C) |

## Las 12 reglas del Álgebra de Boole

| | | | |
|---|---|---|---|
| 1 | $\mathbf{A+0=A}$ | 7 | $\mathbf{A.A=A}$ |
| 2 | $\mathbf{A+1=1}$ | 8 | $\mathbf{A.\overline{A}=0}$ |
| 3 | $\mathbf{A.0=0}$ | 9 | $\mathbf{\overline{\overline{A}}=A}$ |
| 4 | $\mathbf{A.1=A}$ | 10 | $\mathbf{A+AB=A}$ |
| 5 | $\mathbf{A+A=A}$ | 11 | $\mathbf{A+\overline{A}B=A+B}$ |
| 6 | $\mathbf{A+\overline{A}=1}$ | 12 | $\mathbf{(A+B)(A+C)=A+BC}$ |

Vale tener en cuenta que cada A, B y C puede representar una variable o una combinación de variables.
Las 9 primeras reglas pueden demostrarse a través de su ejemplificación con compuertas y las 3 últimas (10 a 12) a través de la aplicación de las 9 primeras reglas y las leyes del álgebra de Boole.

<u>Demostración de Regla 10</u>: $\mathbf{A+AB=A}$

$\mathbf{A+AB=A\,(1+B)}$ Factor común (distributiva)
$\mathbf{A+AB=A.1}$ Regla 2 $(\boldsymbol{1+B=1})$
$\mathbf{A+AB=A}$ Regla 4 $(\boldsymbol{A.1=A})$

<u>Demostración de Regla 11</u>: $\mathbf{A+\overline{A}B=A+B}$

$\mathbf{A+\overline{A}B=(A+AB)+\overline{A}B}$ Regla 10 $(\boldsymbol{A=A+AB})$
$\mathbf{A+\overline{A}B=AA+AB+\overline{A}B}$ Regla 7 $(\boldsymbol{A=A.A})$
$\mathbf{A+\overline{A}B=AA+AB+\overline{A}A+\overline{A}B}$ Regla 8 $(\boldsymbol{A.\overline{A}=0})$ (sumamos $\mathbf{\overline{A}A}$)
$\mathbf{A+\overline{A}B=(A+\overline{A})(A+B)}$ Factor común (distributiva)
$\mathbf{A+\overline{A}B=1.(A+B)}$ Regla 6 $(\boldsymbol{A+\overline{A}=1})$
$\mathbf{A+\overline{A}B=A+B}$ Regla 4 $(\boldsymbol{(A+B).1=A+B})$

<u>Demostración de Regla 12</u>: $\mathbf{(A+B)(A+C)=A+BC}$

$\mathbf{(A+B)(A+C)=AA+AC+AB+BC}$ Distributiva
$\mathbf{(A+B)(A+C)=A+AC+AB+BC}$ Regla 7 $(\boldsymbol{A.A=A})$
$\mathbf{(A+B)(A+C)=A(1+C)+AB+BC}$ Factor común (distributiva)
$\mathbf{(A+B)(A+C)=A.1+AB+BC}$ Regla 2 $(\boldsymbol{1+C=1})$
$\mathbf{(A+B)(A+C)=A+AB+BC}$ Regla 4 $(\boldsymbol{A.1=A})$
$\mathbf{(A+B)(A+C)=A(1+B)+BC}$ Factor común (distributiva)
$\mathbf{(A+B)(A+C)=A.1+BC}$ Regla 2 $(\boldsymbol{1+B=1})$
$\mathbf{(A+B)(A+C)=A+BC}$ Regla 4 $(\boldsymbol{A.1=A})$

## Teoremas de DeMORGAN:

<u>Primer teorema de DeMorgan</u>:

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

**Expresión lógica** correspondiente**:** $\mathbf{\overline{XY}=\overline{X}+\overline{Y}}$

Dicho de otra manera:
**El complemento de dos o más variables a las que se aplica la operación AND es equivalente a aplicar la operación OR a los complementos de cada variable.**

Segundo teorema de DeMorgan:

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

**Expresión lógica** correspondiente: $\overline{X+Y} = \overline{X}\,\overline{Y}$

Dicho de otra manera:
**El complemento de dos o más variables a las que se aplica la operación OR es equivalente a aplicar la operación AND a los complementos de cada variable.**

Estos teoremas también pueden aplicarse a expresiones en las que intervienen más de dos variables.

Ejemplos de aplicación de Morgan

a_ $\overline{\mathbf{ABCD}} = \overline{\mathbf{A}} + \overline{\mathbf{B}} + \overline{\mathbf{C}} + \overline{\mathbf{D}}$

b_ $\overline{\mathbf{A+B+C+D}} = \overline{\mathbf{A}}\,\overline{\mathbf{B}}\,\overline{\mathbf{C}}\,\overline{\mathbf{D}}$

c_ $\overline{\overline{\mathbf{A}}\,\overline{\mathbf{B}}\,\overline{\mathbf{C}}\,\overline{\mathbf{D}}} = \overline{\overline{\mathbf{A}}} + \overline{\overline{\mathbf{B}}} + \overline{\overline{\mathbf{C}}} + \overline{\overline{\mathbf{D}}} = \mathbf{A+B+C+D}$

d_ $\overline{\mathbf{(AB+C)(A+BC)}} = \overline{\mathbf{(AB+C)}} + \overline{\mathbf{(A+BC)}} = (\overline{\mathbf{AB}})\overline{\mathbf{C}} + \overline{\mathbf{A}}(\overline{\mathbf{BC}}) = (\overline{\boldsymbol{A}} + \overline{\boldsymbol{B}})\boldsymbol{C} + \boldsymbol{A}(\overline{\boldsymbol{B}} + \overline{\boldsymbol{C}})$

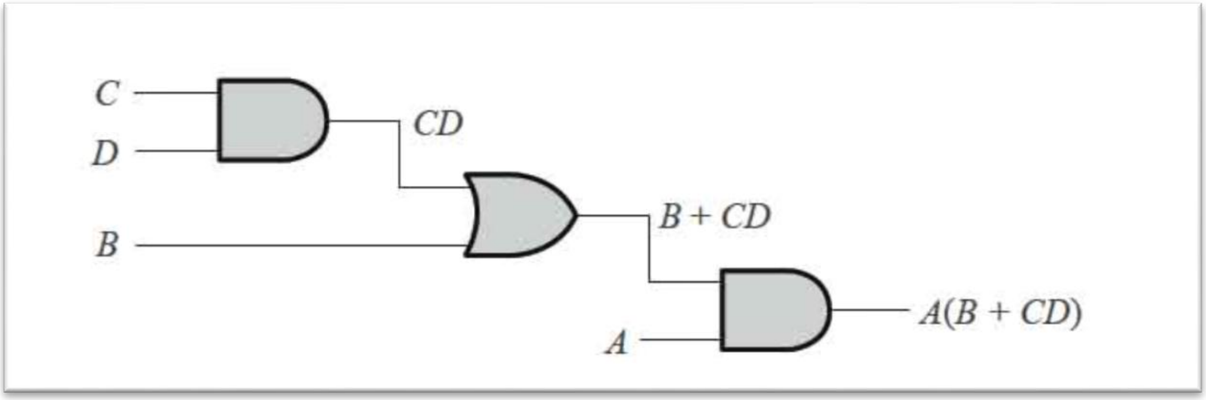
e_ $\overline{\overline{\mathbf{A+B\overline{C}}} + \mathbf{D}(\overline{\mathbf{E}+\overline{\mathbf{F}}})} = \overline{\overline{(\mathbf{A+B\overline{C}})}}\ \overline{(\mathbf{D}(\overline{\mathbf{E}+\overline{\mathbf{F}}}))} = (\mathbf{A+B\overline{C}})\overline{(\mathbf{D}(\overline{\mathbf{E}+\overline{\mathbf{F}}}))} =$

$= (\mathbf{A+B\overline{C}})(\overline{\mathbf{D}} + \overline{\overline{\boldsymbol{E}+\overline{\boldsymbol{F}}}}) = (\mathbf{A+B\overline{C}})(\overline{\mathbf{D}} + \mathbf{E} + \overline{\mathbf{F}})$

*(Teniendo en cuenta que se considera a cada término como una única variable y aplicando la regla 9 para simplificar doble barra)*

## Expresión booleana de un circuito lógico

Al observar un circuito, se comienza con las entradas situadas más a la izquierda y se va escribiendo la expresión correspondiente a cada compuerta.




La expresión lógica para el circuito anterior es: A(B+CD)

## Tabla de verdad de un circuito lógico

Se puede decir que un circuito lógico puede describirse mediante una **tabla de verdad**.
Una vez determinada la expresión booleana de un determinado circuito, puede desarrollarse una tabla de verdad que represente la salida del circuito lógico para todos los valores posibles de las variables de entrada.
El procedimiento requiere que se **evalúe** la expresión booleana para todas las posibles combinaciones de valores de las variables de entrada. Por ejemplo, para el caso de un circuito con cuatro variables de entrada, se tendrán dieciséis ($2^4$=16) posibles combinaciones de valores.

**Evaluación de la expresión:** Para evaluar la expresión **A(B + CD),** en primer lugar hallamos los valores de las variables que hacen que la expresión sea igual a 1, utilizando las reglas de la suma y la multiplicación booleanas.

En este caso, la expresión es igual a 1 sólo si A = 1 y B + CD=1, ya que:
A(B + CD) = 1 . 1 = 1
Ahora hay que determinar cuándo el término B + CD es igual a 1. El término B + CD=1 siempre que B =1 , o bien si C =1 y a la vez D=1 , ya que:
B + CD = 1 + 0 = 1
B + CD = 0 + 1 = 1
B + CD = 1 + 1 = 1
(El término CD =1 sólo si se cumple a la vez que C =1 y D=1).

Resumiendo, la expresión $A\,(B + CD) = 1$ cuando $A = 1$ y $B = 1$, independientemente de los valores de $C$ y $D$, o cuando $A = 1$ y $C = 1$ o cuando $A = 1$ y $C = 1$ y $D = 1$, independientemente del valor de $B$.
La expresión $A\,(B + CD) = 0$ para todas las restantes combinaciones de valores de las variables.

**Representación de los resultados en una tabla de verdad:**

- Enumerar las dieciséis combinaciones de unos y ceros de las variables de entrada en una secuencia binaria.
- Poner un 1 en la columna de salida para las combinaciones de variables de entrada que se han determinado en la evaluación de la expresión.

- Se escribe un 0 en la columna de salida para el resto de las combinaciones de las variables de entrada.

| *Entradas* | | | | *Salidas* |
|---|---|---|---|---|
| **A** | **B** | **C** | **D** | **A(B+CD)** |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

## Ejemplos de simplificación mediante Boole

Cómo utilizar las reglas, leyes y teoremas del álgebra de Boole para manipular y simplificar una expresión.

*Ejemplo 1- simplificar la siguiente expresión:*

$$\mathbf{AB + A(B + C) + B(B + C)}$$

$\mathbf{AB + AB + AC + BB + BC}$ Distributiva
$\mathbf{AB + AB + AC + B + BC}$ Regla 7 $(\mathbf{BB = B})$
$\mathbf{AB + AC + B + BC}$ Regla 5 $(\mathbf{AB + AB = AB})$
$\mathbf{AB + AC + B}$ Regla 10 $(\mathbf{B + BC = B})$
$\mathbf{AC + B}$ Regla 10 $(\mathbf{AB + B = B})$

La expresión más simple que se puede obtener es: $\mathbf{AC + B}$

*Ejemplo 2- simplificar la siguiente expresión:*

$$\overline{\mathbf{AB + AC}} + \overline{\mathbf{A}}\,\overline{\mathbf{B}}\,\mathbf{C}$$

| | |
|---|---|
| $\left(\overline{\mathbf{AB}}\right)\left(\overline{\mathbf{AC}}\right) + \overline{\mathbf{A}}\,\overline{\mathbf{B}}\,\mathbf{C}$ | De Morgan |
| $\left(\overline{\mathbf{A}} + \overline{\mathbf{B}}\right)\left(\overline{\mathbf{A}} + \overline{\mathbf{C}}\right) + \overline{\mathbf{A}}\,\overline{\mathbf{B}}\,\mathbf{C}$ | De Morgan |
| $\left(\overline{\mathbf{A}}\,\overline{\mathbf{A}} + \overline{\mathbf{A}}\,\overline{\mathbf{C}} + \overline{\mathbf{A}}\,\overline{\mathbf{B}} + \overline{\mathbf{B}}\,\overline{\mathbf{C}}\right) + \overline{\mathbf{A}}\,\overline{\mathbf{B}}\,\mathbf{C}$ | Distributiva |
| $\overline{\mathbf{A}} + \overline{\mathbf{A}}\,\overline{\mathbf{C}} + \overline{\mathbf{A}}\,\overline{\mathbf{B}} + \overline{\mathbf{B}}\,\overline{\mathbf{C}} + \overline{\mathbf{A}}\,\overline{\mathbf{B}}\,\mathbf{C}$ | Regla 7 $(\overline{\mathbf{A}}\,\overline{\mathbf{A}} = \overline{\mathbf{A}})$ |
| $\overline{\mathbf{A}} + \overline{\mathbf{A}}\,\overline{\mathbf{C}} + \overline{\mathbf{A}}\,\overline{\mathbf{B}} + \overline{\mathbf{B}}\,\overline{\mathbf{C}}$ | Regla 10 $(\overline{\mathbf{A}}\,\overline{\mathbf{B}} + \overline{\mathbf{A}}\,\overline{\mathbf{B}}\,\mathbf{C}) = \overline{\mathbf{A}}\,\overline{\mathbf{B}}$ |
| $\overline{\mathbf{A}} + \overline{\mathbf{A}}\,\overline{\mathbf{B}} + \overline{\mathbf{B}}\,\overline{\mathbf{C}}$ | Regla 10 $(\overline{\mathbf{A}} + \overline{\mathbf{A}}\,\overline{\mathbf{C}}) = \overline{\mathbf{A}}$ |
| $\overline{\mathbf{A}} + \overline{\mathbf{B}}\,\overline{\mathbf{C}}$ | Regla 10 $(\overline{\mathbf{A}} + \overline{\mathbf{A}}\,\overline{\mathbf{B}}) = \overline{\mathbf{A}}$ |

La expresión más simple que se puede obtener es: $\overline{\mathbf{A}} + \overline{\mathbf{B}}\,\overline{\mathbf{C}}$

## Formas estándar o canónicas:

Todas las expresiones booleanas pueden convertirse en cualquiera de las dos formas estándar: **suma de productos (o suma de mintérminos)** o **producto de sumas (o producto de maxtérminos)**.
La estandarización posibilita que la evaluación, simplificación e implementación de las expresiones booleanas sea mucho más sistemática y sencilla.
Una ***suma de productos estándar*** es aquella en la que ***todas*** las variables del dominio aparecen en cada uno de los términos de la expresión.
Un ***producto de sumas estándar*** es aquel en el que ***todas*** las variables del dominio o sus complementos aparecen en cada uno de los términos de la expresión.

### Suma de productos:

Consiste en 2 o más términos productos sumados. Ejemplos:

$$\mathbf{AB} + \mathbf{ABC}$$
$$\mathbf{ABC} + \mathbf{C\overline{D}E} + \mathbf{\overline{B}CD}$$
$$\mathbf{\overline{A}B} + \mathbf{\overline{A}B\overline{C}} + \mathbf{AC}$$

En una expresión con formato de suma de productos, una barra no puede extenderse sobre más de una variable; sin embargo, más de una variable puede tener una barra encima. Por ejemplo, una suma de productos puede contener el término $\overline{\mathbf{A}}\,\overline{\mathbf{B}}\,\overline{\mathbf{C}}$ , pero no el término $\overline{\mathbf{ABC}}$ **.**

### Conversión de una expresión general a formato suma de productos

Cualquier expresión lógica puede ser transformada a una expresión suma de productos aplicando el álgebra de Boole. Por ejemplo, la expresión A(B+CD) puede convertirse en una suma de productos aplicando la ley distributiva:

A (B + CD) = AB + ACD

**Dominio de una expresión booleana:**
Es el conjunto de variables contenido en la expresión (bien en su forma complementada o no complementada).

**Conversión de una suma de productos a su forma estándar.**
Una suma de productos no estándar se convierte a su forma estándar utilizando la regla 6:
$\mathbf{A + \overline{A} = 1}$ (la suma de una variable y su complemento es igual a 1). Se debe:

- Multiplicar cada término producto no estándar por un término formado por la suma de la variable que falta y su complemento. Con esto se obtienen dos términos producto. Como se sabe, se puede multiplicar por 1 cualquier expresión sin que se altere su valor.
- Repetir el paso 1 hasta que todos los términos de la expresión contengan todas las variables o sus complementos del dominio. (Resulta que el número de términos producto se duplica por cada variable que falta).

Ejemplo: teniendo
$\mathbf{A\overline{B}C + \overline{A}\,\overline{B} + AB\overline{C}D}$

Dominio: A, B, C, D. Por lo tanto al primer término le falta la variable D y al segundo las variables C y D. Multiplicamos al primer término por $\mathbf{(D + \overline{D})}$**:**
$\mathbf{A\overline{B}C\,(D + \overline{D}) + \overline{A}\,\overline{B} + AB\overline{C}D}$
$\mathbf{A\overline{B}CD + A\overline{B}C\overline{D} + \overline{A}\,\overline{B} + AB\overline{C}D}$
Multiplicamos al segundo término original ($\mathbf{\overline{A}\,\overline{B}}$)por $\mathbf{(C + \overline{C})}$**:**
$\mathbf{A\,\overline{B}\,C\,D + A\,\overline{B}\,C\overline{D} + \overline{A}\,\overline{B}\left(C + \overline{C}\right) + A\,B\,\overline{C}\,D}$
$\mathbf{A\,\overline{B}\,C\,D + A\,\overline{B}\,C\,\overline{D} + \overline{A}\,\overline{B}\,C + \overline{A}\,\overline{B}\,\overline{C} + A\,B\,\overline{C}\,D}$

Multiplicamos a los nuevos términos generados por $\mathbf{(D + \overline{D})}$**:**

$\mathbf{A\,\overline{B}\,C\,D + A\,\overline{B}\,C\,\overline{D} + \overline{A}\,\overline{B}\,C\left(D + \overline{D}\right) + \overline{A}\,\overline{B}\,\overline{C}\,(D + \overline{D}) + A\,B\,\overline{C}\,D}$

$$\boxed{\mathbf{A\,\overline{B}\,C\,D + A\,\overline{B}\,C\,\overline{D} + \overline{A}\,\overline{B}\,C\,D + \overline{A}\,\overline{B}\,C\,\overline{D} + \overline{A}\,\overline{B}\,\overline{C}\,D + \overline{A}\,\overline{B}\,\overline{C}\,\overline{D} + A\,B\,\overline{C}\,D}}$$

**Una expresión suma de productos es igual a 1 si y sólo si uno o más de los términos productos que forman la expresión es igual a 1.**

**Producto de sumas**:
Consiste en una expresión en la que dos o más términos suma se multiplican. Ejemplo:

$$\left(\overline{\mathbf{A}}+\mathbf{B}\right)(\mathbf{A}+\overline{\mathbf{B}}+\mathbf{C})$$

NOTA: en este material de estudio no se trata nada más sobre productos de sumas.

## Conversión de una suma de productos a tabla de verdad

Se prepara la tabla con la lista de las posibles combinaciones de valores de las variables de entrada (Para una expresión cuyo dominio es de dos variables, existen cuatro combinaciones distintas de estas variables: $2^2$ = 4. En caso de un dominio de tres variables, ocho: $2^3$ = 8 combinaciones posibles de dichas variables. Con un dominio de cuatro variables, existen dieciséis combinaciones $2^4$ = 16.) y sus correspondientes valores de salida.
Luego, hay que pasar la suma de productos a su formato estándar, si no lo está ya. Por último, se escribe un 1 en la columna de salida para cada valor binario que hace que la suma de productos estándar sea 1, y se escribe un 0 para los restantes valores.

Ejemplo: para $\overline{\mathbf{A}}\,\overline{\mathbf{B}}\,\mathbf{C} + \mathbf{A}\overline{\mathbf{B}}\,\overline{\mathbf{C}} + \mathbf{A}\,\mathbf{B}\,\mathbf{C}$ (son 3 variables, 8 combinaciones)

| *Entradas* | | | *Salida* | *Término producto* |
|---|---|---|---|---|
| **A** | **B** | **C** | **X** | |
| 0 | 0 | 0 | 0 | |
| 0 | 0 | 1 | 1 | $\overline{\mathbf{A}}\,\overline{\mathbf{B}}\,\mathbf{C}$ |
| 0 | 1 | 0 | 0 | |
| 0 | 1 | 1 | 0 | |
| 1 | 0 | 0 | 1 | $\mathbf{A}\overline{\mathbf{B}}\,\overline{\mathbf{C}}$ |
| 1 | 0 | 1 | 0 | |
| 1 | 1 | 0 | 0 | |
| 1 | 1 | 1 | 1 | $\mathbf{A}\,\mathbf{B}\,\mathbf{C}$ |

Los valores binarios que hacen que los términos producto de la expresión sean igual a 1 son
$\overline{\mathbf{A}}\,\overline{\mathbf{B}}\,\mathbf{C}\quad \mathbf{001}$
$\mathbf{A}\overline{\mathbf{B}}\,\overline{\mathbf{C}}\quad \mathbf{100}$
$\mathbf{A}\,\mathbf{B}\,\mathbf{C}\quad \mathbf{111}$

## Determinación de las expresiones estándar a partir de una tabla de verdad

Se debe:

- enumerar todos los valores de las variables de entrada para los que la salida es 1.
- reemplazar cada 1 por la variable y cada 0 por la variable complementada.

*Ejemplo*: a partir de la siguiente tabla de verdad determinar la expresión estándar.

| *Entradas* | | | *Salida* | *Término producto* |
|---|---|---|---|---|
| **A** | **B** | **C** | **X** | |
| 0 | 0 | 0 | 0 | |
| 0 | 0 | 1 | 0 | |
| 0 | 1 | 0 | 0 | |
| 0 | 1 | 1 | 1 | $\mathbf{\overline{A}\,B\,C}$ |
| 1 | 0 | 0 | 1 | $\mathbf{A\,\overline{B}\,\overline{C}}$ |
| 1 | 0 | 1 | 0 | |
| 1 | 1 | 0 | 1 | $\mathbf{A\,B\,\overline{C}}$ |
| 1 | 1 | 1 | 1 | $\mathbf{A\,B\,C}$ |

La expresión suma de productos estándar resultante para la salida X es:

$$\mathbf{X = \overline{A}\,B\,C + A\,\overline{B}\,\overline{C} + A\,B\,\overline{C} + A\,B\,C}$$

## Mapas de Karnaugh

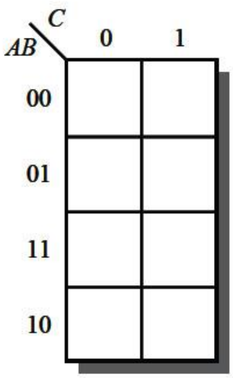
<u>Importante</u>: antes de poder utilizar un mapa de Karnaugh, las expresiones booleanas **deben estar en su forma estándar**. Si una expresión no lo está, se pasará al formato estándar mediante el procedimiento de *desarrollo numérico de un producto no estándar* no descripto en este material de estudio.

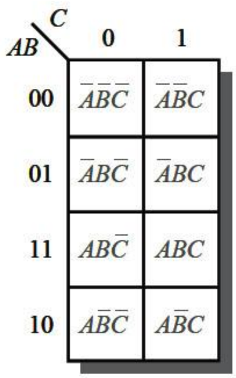
Un mapa de Karnaugh es un método sistemático de simplificación de expresiones booleanas y, si se aplica adecuadamente, genera las expresiones suma de productos y producto de sumas más simples posibles, conocidas como **expresiones mínimas**.

Consiste en una matriz de celdas en la que cada celda representa un valor binario de las variables de entrada. Las celdas se organizan de manera que la simplificación de una determinada expresión consiste en agrupar adecuadamente las celdas.
Los mapas de Karnaugh se pueden utilizar para expresiones de dos, tres, cuatro y cinco variables, pero sólo nos trataremos los casos de tres y cuatro variables.

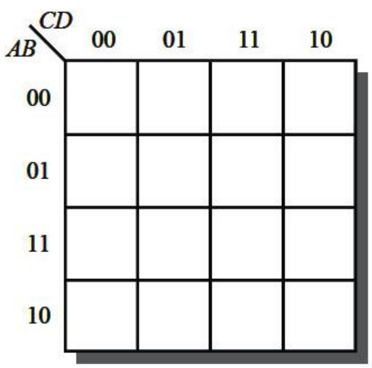
El número de celdas de un mapa de Karnaugh es igual al número total de posibles combinaciones de las variables de entrada, al igual que el número de filas de una tabla de verdad.
Para tres variables, el número de celdas necesarias es de $2^3 = 8$. Para cuatro variables, el número de celdas es de $2^4 = 16$.

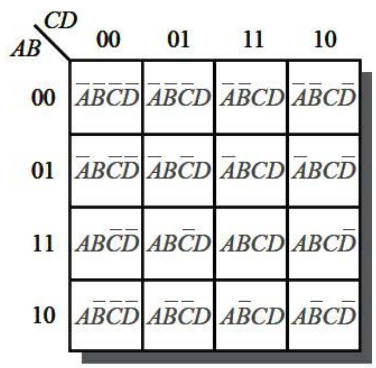
Mapa de Karnaugh de 3 variables:




Mapa de Karnaugh de 4 variables:




## Adyacencia de celdas en un mapa de Karnaugh

- Las celdas que **difieren en una única variable son adyacentes**.
- Las celdas con valores que **difieren en más de una variable no son adyacentes**.

Por ejemplo, en el mapa de tres variables, la celda 010 es adyacente a las celdas 000, 011 y 110. La celda 010 no es adyacente a la celda 001, ni a la celda 111, ni a la celda 100 ni a la celda 101.
Físicamente, cada celda es adyacente a las celdas que están situadas inmediatas a ella por cualquiera de sus cuatro lados. Una celda no es adyacente a aquellas celdas que tocan diagonalmente alguna de sus esquinas. Además, las celdas de la fila superior son adyacentes a las de la fila inferior y las celdas de la columna izquierda son adyacentes a las situadas en la columna de la derecha. Esto se denomina adyacencia cíclica, ya que podemos pensar que el mapa de Karnaugh se dobla de forma que se toquen los extremos superior e inferior como si fuera un cilindro o los extremos de la derecha e izquierda para formar la misma figura.

## Minimización de una suma de productos mediante el mapa de Karnaugh

Una expresión suma de productos minimizada está formada por el mínimo número de términos producto posibles con el mínimo número de variables por término.

Generalmente, una expresión suma de productos minimizada puede implementarse mediante un número de puertas menor que su expresión estándar, lo cual constituye la finalidad del proceso de simplificación.

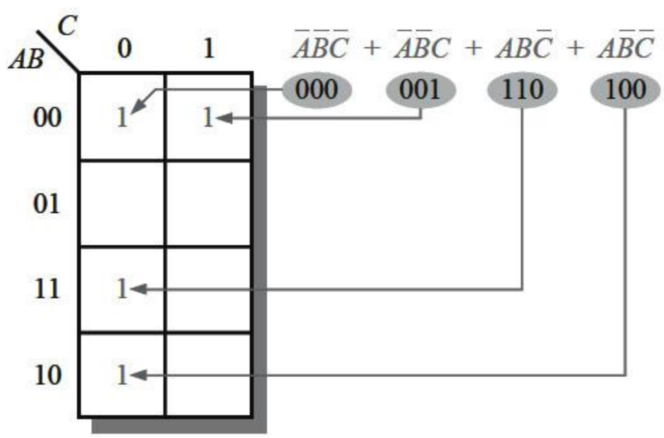
Por cada término de la expresión suma de productos, se coloca un 1 en el mapa de Kanaugh en la celda correspondiente al valor del producto.

Para completar entonces los mapas de Karnaugh, se debe:

Paso 1: determinar el valor binario de cada término producto de la suma de productos estándar.
Paso 2: colocar un **1** en el mapa de Karnaugh en la celda que tiene el mismo valor que dicho término producto.
Por ejemplo, para el término $\mathbf{A\,\overline{B}\,C}$, se escribiría un 1 en la celda 101 de un mapa de Karnaugh de tres variables.

Normalmente, cuando se trabaja con una expresión suma de productos, los 0s no se incluyen en el mapa (las celdas que no contienen un 1 son aquellas para las que la expresión es igual a 0).

Cuando una expresión suma de productos se ha reflejado por completo en el mapa de Karnaugh, en dicho mapa habrá tantos 1s como términos producto tenga la suma de productos estándar.

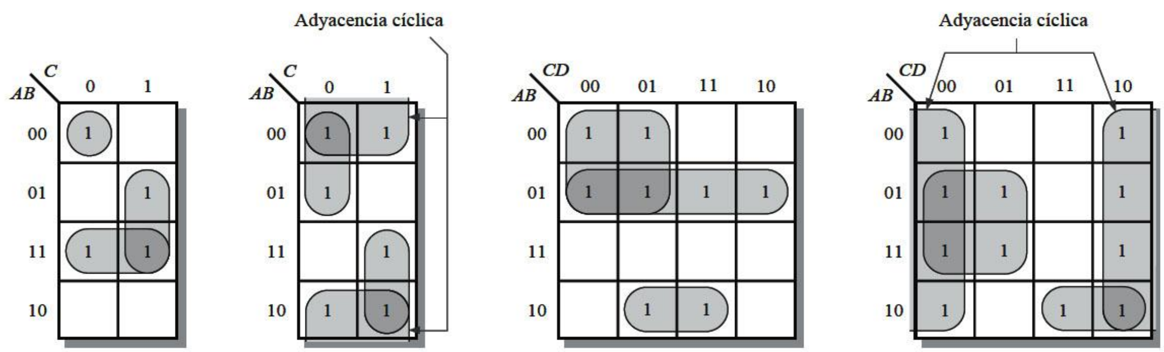



Después de haber obtenido el mapa de Karnaugh de una suma de productos, **la expresión suma de productos mínima se obtiene agrupando los 1s y determinando la expresión suma de productos mínima a partir del mapa**.

**Agrupación de unos**:
Podemos agrupar los unos del mapa de Karnaugh de acuerdo con las reglas siguientes, rodeando las celdas adyacentes que contengan unos. La finalidad es maximizar el tamaño de los grupos y minimizar el número de estos grupos.

1. Un grupo tiene que contener 1, 2, 4, 8 ó 16 celdas, valores que se corresponden con las potencias de 2. En el caso de un mapa de Karnaugh de 3 variables, el grupo máximo puede contener $2^3 = 8$ celdas.

2. Cada celda de un grupo tiene que ser adyacente a una o más celdas del mismo grupo, pero no todas las celdas del grupo tienen que ser adyacentes entre sí.
3. Incluir siempre en cada grupo el mayor número posible de 1s de acuerdo a la regla número 1.
4. Cada 1 del mapa tiene que estar incluido en al menos un grupo. Los 1s que ya pertenezcan a un grupo pueden estar incluidos en otro, siempre que los grupos que se solapen contengan 1s no comunes.




En algunos casos, puede existir más de una forma de agrupar los 1s para formar grupos máximos.

**<u>Determinación de la expresión suma de productos mínima a partir del mapa.</u>**

Cuando todos los 1s que representan los términos productos estándar de una expresión se han trasladado al mapa y se han agrupado adecuadamente, comienza el proceso de obtención de la suma de productos mínima. Para encontrar los términos mínimos y la expresión suma de productos mínima se aplican las siguientes reglas:

1. Agrupar las celdas que contienen 1s. Cada grupo de celdas que contiene 1s da lugar a un término producto compuesto por todas las variables que aparecen en el grupo en sólo una forma (no complementada o complementada). Las variables que aparecen complementadas y sin complementar dentro del mismo grupo se eliminan. A éstas se les denomina **variables contradictorias**.
2. Determinar la operación producto mínima para cada grupo.

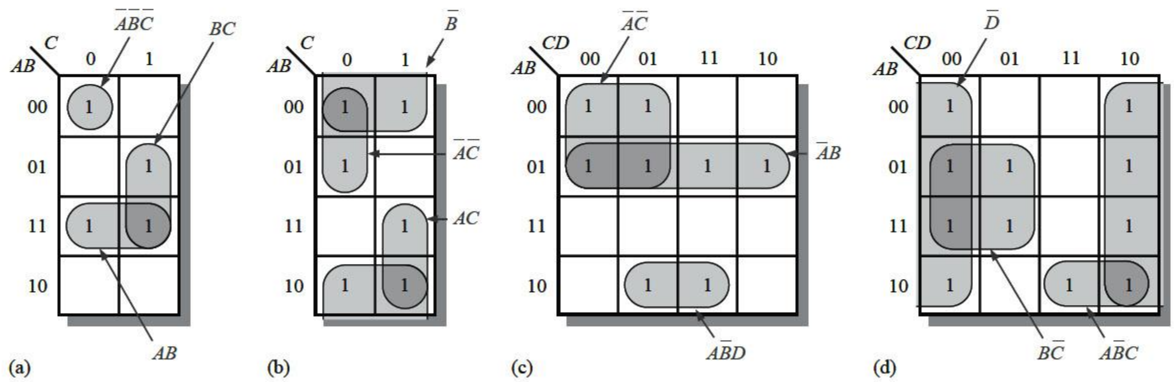
(a) Para un mapa de 3 variables:

(1) Un grupo formado por 1 celda da lugar a un término producto de 3 variables.
(2) Un grupo formado por 2 celdas da lugar a un término producto de 2 variables.
(3) Un grupo formado por 4 celdas da lugar a un término de 1 variable.
(4) Un grupo formado por 8 celdas indica que la expresión vale 1.

(b) Para un mapa de 4 variables:

(1) Un grupo formado por 1 celda da lugar a un término producto de 4 variables.
(2) Un grupo formado por 2 celdas da lugar a un término producto de 3 variables.
(3) Un grupo formado por 4 celdas da lugar a un término producto de 2 variables.
(4) Un grupo formado por 8 celdas da lugar a un término de 1 variable.
(5) Un grupo formado por 16 celdas indica que la expresión vale 1.

**3.** Cuando se han obtenido todos los términos producto mínimos a partir del mapa de Karnaugh, se suman para obtener la expresión suma de productos mínima.




Expresiones suma de productos mínima para cada uno de los mapas de Karnaugh anteriores:

**a )** $\mathbf{A\,B + B\,C + \overline{A}\,\overline{B}\,\overline{C}}$

**b)** $\mathbf{\overline{B} + \overline{A}\,\overline{C} + A\,C}$

**c)** $\mathbf{\overline{A}\,B + \overline{A}\,\overline{C} + A\,\overline{B}\,D}$

**d)** $\mathbf{\overline{D} + A\,\overline{B}\,C + B\,\overline{C}}$