ALGEBRA DE BOOLE – EJERCICIO 1

1) **TABLA NOT**

A	$\overline{A}$
0	**1**
1	**0**

2) **TABLA AND**

A	$\boldsymbol{B}$	$A * B$
0	**0**	**0**
0	**1**	**0**
1	**0**	**0**
1	**1**	**1**

3) **TABLA OR**

A	$\boldsymbol{B}$	$A + B$
0	**0**	**0**
0	**1**	**1**
1	**0**	**1**
1	**1**	**1**

4) **TABLA NAND**

A	$\boldsymbol{B}$	$\overline{A * B}$
0	**0**	**1**
0	**1**	**1**
1	**0**	**1**
1	**1**	**0**

5) **TABLA NOR**

A	$\boldsymbol{B}$	$\overline{A + B}$
0	**0**	**1**
0	**1**	**0**
1	**0**	**0**
1	**1**	**0**

6) **TABLA XOR (EXCLUSIVE OR)**

A	$\boldsymbol{B}$	$A \oplus B$
0	**0**	**0**
0	**1**	**1**
1	**0**	**1**
1	**1**	**0**

7) TABLA XNOR

A	$\boldsymbol{B}$	$\overline{A \oplus B}$
0	**0**	**1**
0	**1**	**0**
1	**0**	**0**
1	**1**	**1**

8) TABLA AND PARA 3 ENTRADAS

A	$\boldsymbol{B}$	C	$A * B * C$
0	**0**	**0**	**0**
0	**0**	**1**	**0**
0	**1**	**0**	**0**
0	**1**	**1**	**0**
1	**0**	**0**	**0**
1	**0**	**1**	**0**
1	**1**	**0**	**0**
1	**1**	**1**	**1**

9) TABLA OR PARA 3 ENTRADAS

A	$\boldsymbol{B}$	C	$A + B + C$
0	**0**	**0**	**0**
0	**0**	**1**	**1**
0	**1**	**0**	**1**
0	**1**	**1**	**1**
1	**0**	**0**	**1**
1	**0**	**1**	**1**
1	**1**	**0**	**1**
1	**1**	**1**	**1**

10) TABLA NAND PARA 3 ENTRADAS

A	$\boldsymbol{B}$	C	$\overline{A * B * C}$
0	**0**	**0**	**1**
0	**0**	**1**	**1**
0	**1**	**0**	**1**
0	**1**	**1**	**1**
1	**0**	**0**	**1**
1	**0**	**1**	**1**
1	**1**	**0**	**1**
1	**1**	**1**	**0**

11) TABLA NOR PARA 3 ENTRADAS

A	$\boldsymbol{B}$	C	$\overline{A+B+C}$
0	**0**	**0**	**1**
0	**0**	**1**	**0**
0	**1**	**0**	**0**
0	**1**	**1**	**0**
1	**0**	**0**	**0**
1	**0**	**1**	**0**
1	**1**	**0**	**0**
1	**1**	**1**	**0**

a) Una salida en nivel ALTO ocurre sólo cuando las tres entradas están en nivel BAJO.
b) Una salida en nivel BAJO ocurre cuando alguna de las cinco entradas están en nivel BAJO.
c) Una salida en nivel BAJO ocurre sólo cuando las tres entradas están en nivel ALTO.

Rta a) Es una compuerta NOR porque solo si NOT((A=0)+(B=0)+(C=0)) = NOT(0) = 1

A	$\boldsymbol{B}$	$F(A,B)$
0	**0**	**1**
0	**1**	**0**
1	**0**	**0**
1	**1**	**0**

Rta b) Es una compuerta AND porque solo si (A=1)x(B=1)x(C=1)x(D=1)x(E=1) = 1

A	$\boldsymbol{B}$	$F(A,B)$
0	**0**	**0**
0	**1**	**0**
1	**0**	**0**
1	**1**	**1**

Rta c) Es una compuerta NAND porque solo si (A=1)x(B=1)x(C=1) = 0 contrario a b)

A	$\boldsymbol{B}$	$F(A,B)$
0	**0**	**1**
0	**1**	**1**
1	**0**	**1**
1	**1**	**0**

a) Si la forma de onda de salida de una compuerta OR de dos entradas es igual que la forma de onda de una de sus entradas, la otra entrada se encuentra necesariamente puesta a nivel BAJO.
b) Si la forma de onda de salida de una compuerta XOR está siempre en nivel ALTO, entonces al menos una de sus entradas se encuentra permanentemente en nivel ALTO.

A	B	$A + B$
0	**0**	**0**
0	**1**	**1**
1	**0**	**1**
1	**1**	**1**

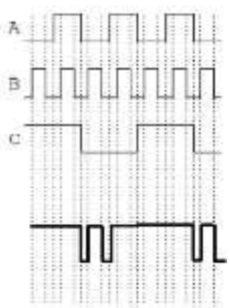
a) **NO SE CUMPLE para (A=1) + (B=1) = 1 porque B no necesamiente debe ser 0**

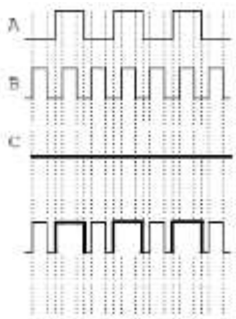
A	B	$A \oplus B$
-	**-**	**-**
0	**1**	**1**
1	**0**	**1**
-	**-**	**-**

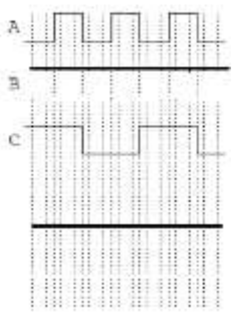
b) **TOTALMENTE CIERTO, en XOR sólo 1 entrada debe ser 1 para que resulte 1**

EJERCICIO 4

- **La compuerta OR tiene salida en ALTO cuando al menos una de sus entradas lo está**
- **Obviamente si una entrada está en ALTO permanente (caso 3), la salida será todo 1**

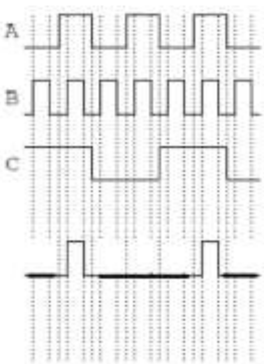


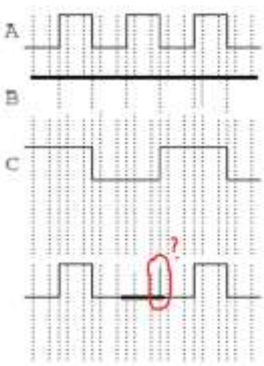


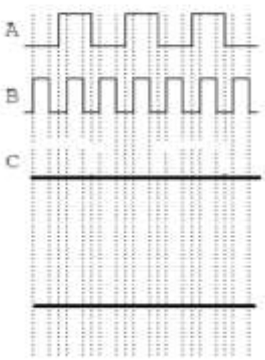


EJERCICIO 5

- **La compuerta AND tiene salida en ALTO sólo si todas las entradas lo están.**
- **Obviamente si una entrada está en BAJO permanente, la salida será todo 0.**

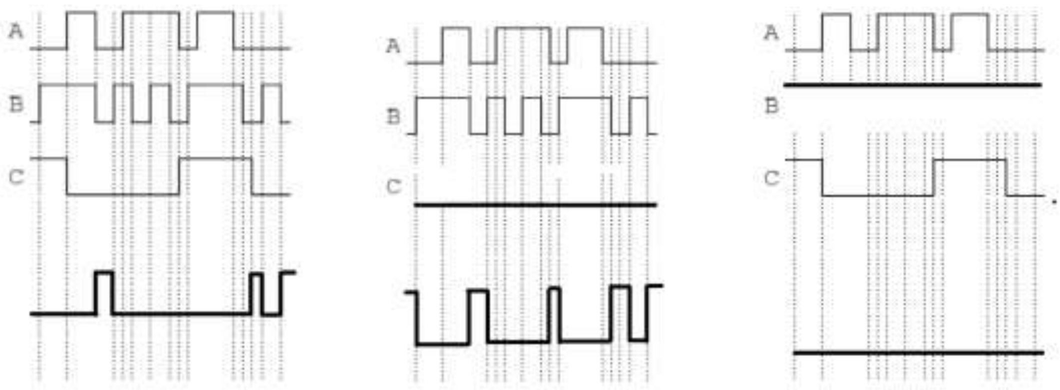






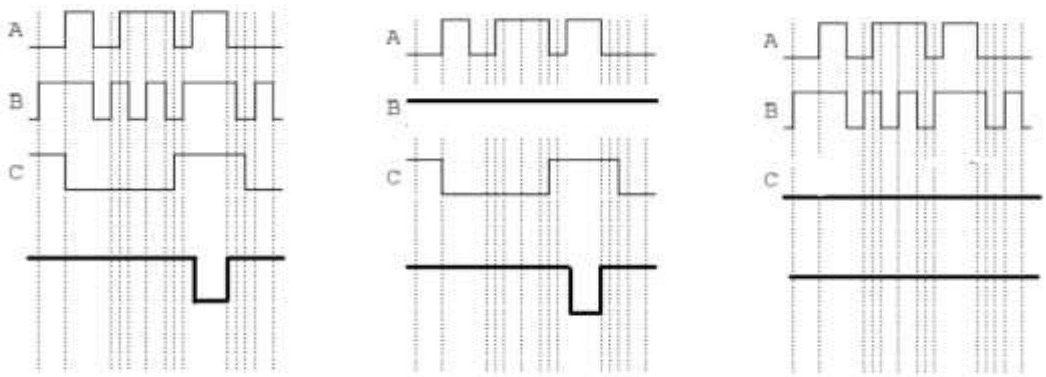
EJERCICIO 6: **La salida de una NOR es ALTO si todas las entradas están en BAJO**

Obviamente si una entrada está en ALTO permanente la salida de NOR será todo en 0



EJERCICIO 7: **La salida de una NAND es ALTO si alguna de las entradas está en BAJO**

Obviamente si alguna entrada está en BAJO permanente la salida de NAND será todo en 1



EJERCICIO 8

Para distinguir entre OR y AND con solo 2 entradas en un único ensayo se puede poner solo una de las dos entradas en ALTO, de modo que si la salida es BAJO es un AND, y si es ALTO es un OR.

A	B	A . B	A + B
0	1	0	1
1	0	0	1

Para distinguir entre OR y NOR con solo 2 entradas en un único ensayo se puede poner cualquier combinación, porque estas compuertas son siempre opuestas entre sí.

A	B	A + B	NOT(A + B)
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

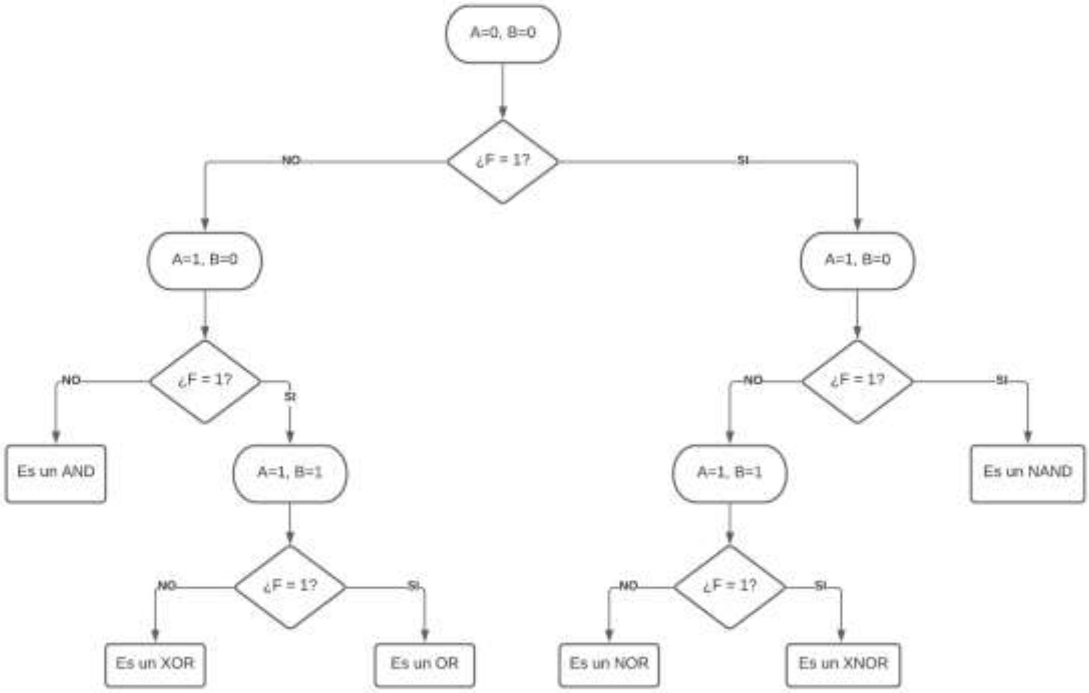
La pregunta anterior estaría mejor hecha si preguntaran distinguir entre OR y NAND, ya que ambas coinciden si las dos entradas son de distinto valor, es decir, difieren en (A=B=0) y (A=B=1)

Para distinguir entre OR y XOR con solo 2 entradas en un único ensayo se debe aplicar una señal de ALTO en ambas entradas, de modo que si es OR la salida es 1, y si es XOR la salida es 0.

A	B	A + B	A (+) B
1	1	1	0

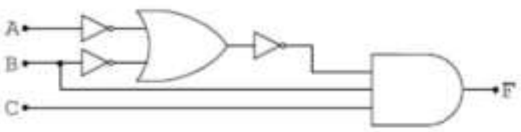
EJERCICIO 9: ¿Cómo distinguir entre OR, AND, NOR, NAND, XOR y XNOR con sólo 2 entradas?

A	B	OR	AND	NOR	NAND	XOR	XNOR
0	**0**	0	0	1	1	0	1
1	**0**	1	0	0	1	1	0
1	**1**	1	1	0	0	0	1



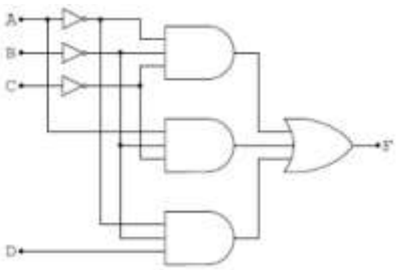
EJERCICIO 10

$$\neg(P \wedge Q) \iff (\neg P) \vee (\neg Q)$$
$$\neg(P \vee Q) \iff (\neg P) \wedge (\neg Q)$$



Expresión lógica: $F = \overline{\overline{A}+\overline{B}} * B * C = \overline{\overline{A * B}} * B * C = A * B * B * C = A * B * C$

A	B	C	F
0	**0**	**0**	**0**
0	**0**	**1**	**0**
0	**1**	**0**	**0**
0	**1**	**1**	**0**
1	**0**	**0**	**0**
1	**0**	**1**	**0**
1	**1**	**0**	**0**
1	**1**	**1**	**1**



Expresión lógica: $F = \overline{A} * \overline{B} * \overline{C} + A * \overline{B} * \overline{C} + \overline{A} * \overline{B} * D$

A	B	C	D	$\overline{A} * \overline{B} * \overline{C}$	$A * \overline{B} * \overline{C}$	$\overline{A} * \overline{B} * D$	F
0	0	0	1	**1**	**0**	**1**	**1**
0	0	0	0	**1**	**0**	**0**	**1**
0	0	1	1	**0**	**0**	**1**	**1**
0	0	1	0	**0**	**0**	**0**	**0**
0	1	0	1	**0**	**0**	**0**	**0**
0	1	0	0	**0**	**0**	**0**	**0**
0	1	1	1	**0**	**0**	**0**	**0**
0	1	1	0	**0**	**0**	**0**	**0**
1	0	0	1	**0**	**1**	**0**	**1**
1	0	0	0	**0**	**1**	**0**	**1**
1	0	1	1	**0**	**0**	**0**	**0**
1	0	1	0	**0**	**0**	**0**	**0**
1	1	0	1	**0**	**0**	**0**	**0**
1	1	0	0	**0**	**0**	**0**	**0**
1	1	1	1	**0**	**0**	**0**	**0**
1	1	1	0	**0**	**0**	**0**	**0**

EJERCICIO 11: Ver circuitos en repositorio de GitHub

EJERCICIO 12: Comprobación de Ley de Morgan para 3 entradas

A	B	C	$\overline{A + B + C}$	$\overline{A} * \overline{B} * \overline{C}$
0	0	0	**NOT(0) = 1**	**1 . 1 . 1 = 1**
0	0	1	**NOT(1) = 0**	**1 . 1 . 0 = 0**
0	1	0	**NOT(1) = 0**	**1 . 0 . 1 = 0**
0	1	1	**NOT(1) = 0**	**1 . 0 . 0 = 0**
1	0	0	**NOT(1) = 0**	**0 . 1 . 1 = 0**
1	0	1	**NOT(1) = 0**	**0 . 1 . 0 = 0**
1	1	0	**NOT(1) = 0**	**0 . 0 . 1 = 0**
1	1	1	**NOT(1) = 0**	**0 . 0 . 0 = 0**

A	B	C	$\overline{A * B * C}$	$\overline{A} + \overline{B} + \overline{C}$
0	0	0	**NOT(0) = 1**	**1 + 1 + 1 = 1**
0	0	1	**NOT(0) = 1**	**1 + 1 + 0 = 1**
0	1	0	**NOT(0) = 1**	**1 + 0 + 1 = 1**
0	1	1	**NOT(0) = 1**	**1 + 0 + 0 = 1**
1	0	0	**NOT(0) = 1**	**0 + 1 + 1 = 1**
1	0	1	**NOT(0) = 1**	**0 + 1 + 0 = 1**
1	1	0	**NOT(0) = 1**	**0 + 0 + 1 = 1**
1	1	1	**NOT(1) = 0**	**0 + 0 + 0 = 0**

EJERCICIO 13 – SUMA DE PRODUCTOS

$$F = \overline{\overline{A} * B * \overline{C}} = \overline{\overline{A} * \overline{\overline{B}} * \overline{C}} = \overline{\overline{A + \overline{B} + C}} = A + \overline{B} + C$$

Verificación con A=0, B=1, C=0 NOT(1.1.1) = 0 debe ser igual a 0 + 0 + 0 = 0

$$F = \overline{\overline{A} + \overline{B} * C} = \overline{\overline{A} + \overline{B} * \overline{\overline{C}}} = \overline{\overline{A} + \overline{B + \overline{C}}} = \overline{\overline{A * (B + \overline{C})}} = A * \left(B + \overline{C}\right) = AB + A\overline{C}$$

Verificación con A=1, B=0, C=1 NOT(0 + 1.1) = 0 debe ser igual a 1.0 + 1.0 = 0

$$F = \overline{A + \overline{B} + C} = \overline{\overline{\overline{A}} + \overline{B} + \overline{\overline{C}}} = \overline{\overline{\overline{A} * B * \overline{C}}} = \overline{A}B\overline{C}$$

Verificación con A=0, B=1, C=0 NOT(0 + 0 + 0) = 1 debe ser igual a 1.1.1 = 1

EJERCICIO 14 – SIMPLIFICAR EXPRESIONES

$$F = A\left(1 + \overline{B} + C\right) + AB\overline{C} = A + A\overline{B} + AC + AB\overline{C} = A\left(1 + \overline{B} + C + B\overline{C}\right) = A$$

A	B	C	$A\overline{B}$	AC	$AB\overline{C}$	F
0	0	0	**0**	**0**	**0**	**0**
0	0	1	**0**	**0**	**0**	**0**
0	1	0	**0**	**0**	**0**	**0**
0	1	1	**0**	**0**	**0**	**0**
1	0	0	**1**	**0**	**0**	**1**
1	0	1	**1**	**1**	**0**	**1**
1	1	0	**0**	**0**	**1**	**1**
1	1	1	**0**	**1**	**0**	**1**

$$F = \overline{\left(\overline{C} + \overline{1 + A}\right)\left(\overline{A} + C\right)} = \overline{\left(\overline{C} + 0\right)\left(\overline{A} + C\right)} = \overline{\overline{A} * \overline{C} + \overline{C}C} = \overline{\overline{A + C}} = A + C$$

A	C	$\left(\overline{C} + \overline{1 + A}\right)$	$\overline{A} + C$	$\left(\overline{C} + \overline{1 + A}\right)\left(\overline{A} + C\right)$	A+C	F
0	0	**1 + 0 = 1**	**1 + 0 = 1**	**1 . 1 = 1**	**0**	**0**
0	1	**0 + 0 = 0**	**1 + 1 = 1**	**0 . 1 = 0**	**1**	**1**
1	0	**1 + 0 = 1**	**0 + 0 = 0**	**1 . 0 = 0**	**1**	**1**
1	1	**0 + 0 = 0**	**0 + 1 = 1**	**0 . 1 = 0**	**1**	**1**

OJO: $\overline{A} * \overline{C} \neq \overline{AC}$

$$F = ABC + A * \left(BCD + B\overline{C}D + BC\overline{D}\right) = AB\left(C * \left(1 + D + \overline{D}\right) + \overline{C}D\right) = AB(C + D)$$

Verificación: $111 + 1 * (110 + 100 + 111) = 1 + 1 * (0 + 0 + 1) = 1 = 1 * 1 * (1 + 0)$

$$F = \overline{\overline{(ABC)} * \overline{\left(A\overline{B}C\right)}} + AB\overline{C} + A * \overline{B} * \overline{C} + \overline{A}B = ABC + A\overline{B}C + A\overline{C} * \left(B + \overline{B}\right) + \overline{A}B$$

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

Verificación: $\overline{\overline{(000)} * \overline{(010)}} + 001 + 011 + 10 = \overline{1 * 1} + 0 + 0 + 0 = 0 = 0 + 0$