OdC Practico 2

Ejercicio 1:

Simplificar las siguientes funciones booleanas a un número mínimo de literales.

a. $x.y + x.y'$
b. $(x + y).(x + y')$
c. $x.y.z + x'.y + xyz'$
d. $z.x + z.x'.y$
e. $(A + B)'.(A' + B')'$
f. $y.(w.z' + w.z) + x.y$

a) $x.y + x.y' \overset{(1)}{=} x.(y+y') \overset{(2)}{=} x.1 \overset{(3)}{=} x$ | (1) Dist. (2) Suma comp. (3) Elem. neutro

b) $(x+y).(x+y') \overset{(1)}{=} x + y.y' \overset{(2)}{=} x.0 \overset{(3)}{=} 0$ | (1) Dist. (2) Mult. comp. (3) Elem. abs

c) $x.y.z + x'.y + x.y.z' = x.y.z + x.y.z' + x'.y \overset{(1)}{=} x.y(z+z') + x'.y \overset{(2)}{=} x.y.1 + x'.y$
$= y.x + y.x' \overset{(1)}{=} y.(x+x') \overset{(2)}{=} y.1 \overset{(3)}{=} y$ | (1) Dist (2) Suma comp. (3) Elem. neutro.

d) $(A+B)'.(A'+B')' \overset{(1)}{=} (A'.B').(A''.B'')$
$\overset{(2)}{=} A'.B'.A.B = A.A'.B.B' = 0.0 = 0$ | (1) De Morgan (2) Comp. doble (3) Mult. comp.

f) $y.(w.z' + w.z) + xy \overset{(1)}{=} y(w.(z'+z)) + xy$
$= y.(w.1) + xy = y.w + y.x = y.(w+x)$

Ejercicio 2:

Reducir a un número mínimo de literales las siguientes funciones booleanas:

a. *(B.C' + A'.D).(A.B' + C.D')*
b. *B'.D + A'.B.C' + A.C.D + A'.B.C*
c. *[(A.B)'.A].[(A.B)'.B]*
d. *A.B' + C'.D'*

a. Graficar las expresiones encontradas en "b" y "d" mediante cualquier tipo de compuertas del número de entradas necesarias.
b. Encontrar expresiones equivalentes a las funciones "b" y "d", pero utilizando sólo compuertas NAND del número de entradas necesarias.
c. Graficar las expresiones encontradas en el punto anterior.

b) $B'.D + A'.B.C' + A.C.D + A'.B.C$

$\overset{(1)}{=} B'.D + A.C.D + A'.B.(C'+C)$

$\overset{(2)}{=} B'.D + A.C.D + A'.B.1$

$\overset{(3)}{=} B'.D + A.C.D + A'.B$

$\overset{(4)}{=} B'.D + B'.D.A'.C + A'.B + A'.B.C.D + A.C.D$

$= B'.D + B'.D.A'.C + A'.B + A'.B.C.D + A.C.D.(B+B')$

$= B'.D + B'.D.A'.C + A'.B + A'.B.C.D + A.C.D.B + A.C.D.B'$

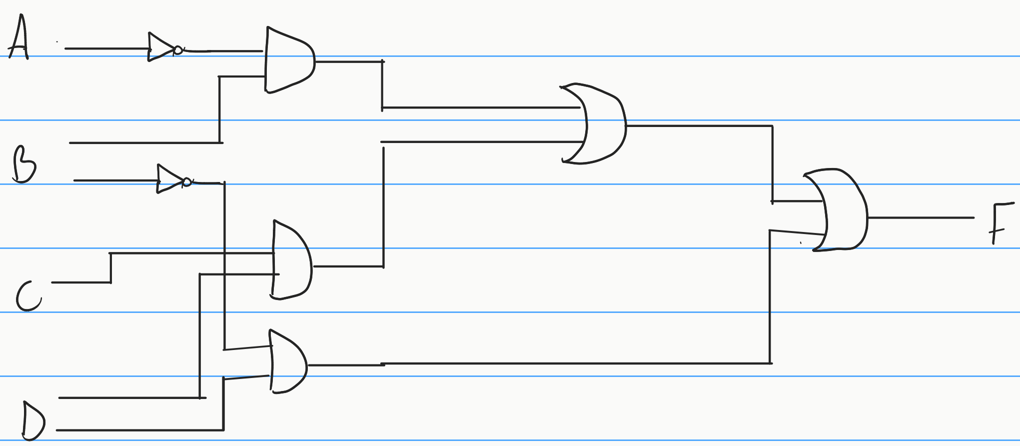
$= B'.D + A'.B + B.C.D(A'+A) + A.B'CD + A'B'CD$

$= B'.D + A'.B + B.C.D + B'.C.D(A+A')$

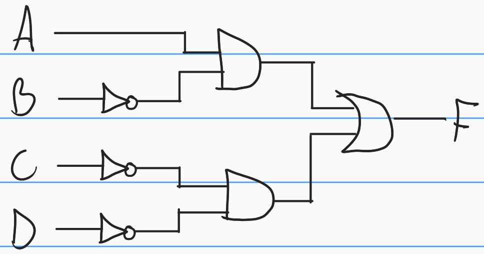
$= B'.D + A'.B + CD.(B+B')$

$= B'.D + A'.B + CD$

A) b) $B'.D + A'.B + CD = F$

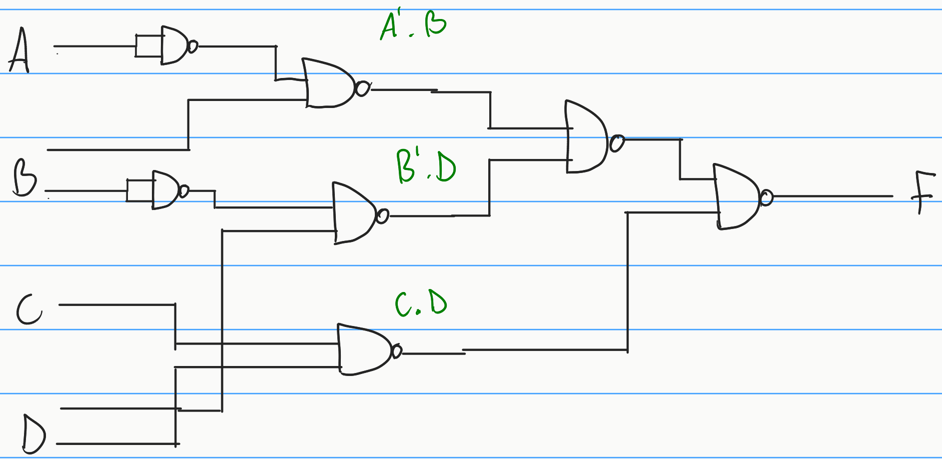


d) $A.B' + C'.D' = F$



B) b) $B'.D + A'.B + CD = (B'.D + A'.B + CD)''$

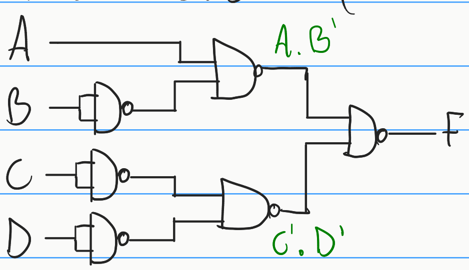
$= ((B'.D)'.(A'.B)'.(CD)')' = F$



NOT = $\overline{X} = (X)' = (X.X)' \longrightarrow$ NAND

NAND = $\overline{X.Y}$

d) $A.B' + C'.D' = (A.B' + C'.D')'' = ((A.B')'.(C'.D')')' = F$



Ejercicio 3:

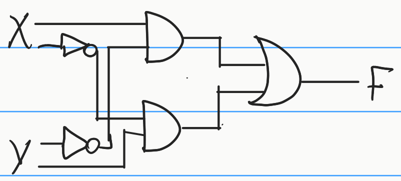
La función OR-exclusiva, denotada por "^" tiene dos entradas y una salida. Si **a** y **b** son las entradas y **c** es la salida, entonces **c** es '1' sólo cuando exactamente una de las entradas vale '1'. En el resto de los casos es '0'.

a. Hacer una tabla de verdad de la función OR-exclusiva.
b. Encontrar la expresión equivalente a la función OR-exclusiva utilizando sólo suma de productos y graficar con compuertas.
c. Implementar una OR-exclusiva de 3 entradas usando OR-exclusivas de 2 entradas.

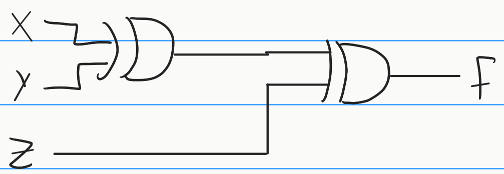
a)

x	y	x ∧ y
0	0	0
0	1	1
1	0	1
1	1	0

b) $x \cdot y' + x' \cdot y$



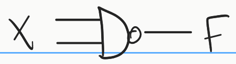
c)



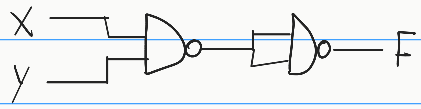
Ejercicio 4:

Mostrar que la función NAND (Not AND) es universal en el sentido de que las funciones NOT, AND, OR y NOR se pueden expresar como productos negados. Graficar las implementaciones de las compuertas NOT, AND, OR y NOR con compuertas NAND.

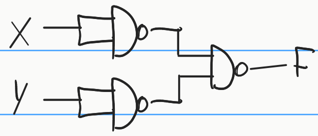
NOT $= x' = (x.x)' = F \Rightarrow$



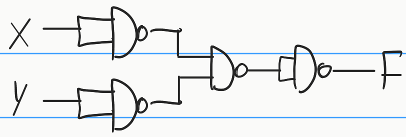
AND $= x.y = (x.y)'' = F \Rightarrow$



OR $= x+y = (x+y)'' = (x'.y')' = F \Rightarrow$

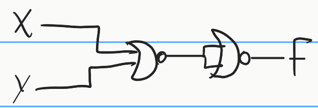


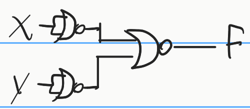
NOR $= (x+y)' = (x+y)''' = (x'.y')'' = F \Rightarrow$



Ejercicio 5:
Mostrar que la función NOR (Not OR) es universal en el sentido de que las funciones NOT, OR, AND y NAND se pueden expresar como sumas negadas. Graficar las implementaciones de las compuertas NOT, OR, AND y NAND con compuertas NOR.

NOT $= \bar{x} = (x+x)' = F \Rightarrow$ 

OR $= x+y = (x+y)'' = F \Rightarrow$ 

AND $= x.y = (x'+y')' = F \Rightarrow$ 

NAND $= (x.y)' = (x'+y')'' = F \Rightarrow$