






PRÁCTICO 2 - Álgebra de Boole*Postulados y teoremas del álgebra booleana*

Postulado 2	a)	$x + 0 = x$	b)	$x \cdot 1 = x$
Postulado 5	a)	$x + x' = 1$	b)	$x \cdot x' = 0$
Teorema 1	a)	$x + x = x$	b)	$x \cdot x = x$
Teorema 2	a)	$x + 1 = 1$	b)	$x \cdot 0 = 0$
Teorema 3, involución		$(x')' = x$		
Postulado 3, conmutatividad	a)	$x + y = y + x$	b)	$xy = yx$
Teorema 4, asociatividad	a)	$x + (y + z) = (x + y) + z$	b)	$x(yz) = (xy)z$
Postulado 4, distributividad	a)	$x(y + z) = xy + xz$	b)	$x + yz = (x + y)(x + z)$
Teorema 5, DeMorgan	a)	$(x + y)' = x'y'$	b)	$(xy)' = x' + y'$
Teorema 6, absorción	a)	$x + xy = x$	b)	$x(x + y) = x$

Name	Distinctive-Shape Graphics Symbol	Algebraic Equation	Truth Table															
AND		$F = XY$	<table><tr><th>X</th><th>Y</th><th>F</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	X	Y	F	0	0	0	0	1	0	1	0	0	1	1	1
X	Y	F																
0	0	0																
0	1	0																
1	0	0																
1	1	1																
OR		$F = X + Y$	<table><tr><th>X</th><th>Y</th><th>F</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	X	Y	F	0	0	0	0	1	1	1	0	1	1	1	1
X	Y	F																
0	0	0																
0	1	1																
1	0	1																
1	1	1																
NOT (inverter)		$F = \overline{X}$	<table><tr><th>X</th><th>F</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table>	X	F	0	1	1	0									
X	F																	
0	1																	
1	0																	
NAND		$F = \overline{X \cdot Y}$	<table><tr><th>X</th><th>Y</th><th>F</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	X	Y	F	0	0	1	0	1	1	1	0	1	1	1	0
X	Y	F																
0	0	1																
0	1	1																
1	0	1																
1	1	0																
NOR		$F = \overline{X + Y}$	<table><tr><th>X</th><th>Y</th><th>F</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	X	Y	F	0	0	1	0	1	0	1	0	0	1	1	0
X	Y	F																
0	0	1																
0	1	0																
1	0	0																
1	1	0																

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. $y.(w.z' + w.z) + x.y$

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.

Ejercicio 3:

La función OR-exclusiva, denotada por “ \wedge ” 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. Encontrar la expresión equivalente a la función OR-exclusiva utilizando sólo producto de sumas y graficar con compuertas.
- d. Implementar una OR-exclusiva de 3 entradas usando OR-exclusivas de 2 entradas.

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.

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.

Ejercicio 6 (extra):

- a. ¿Cuántas funciones booleanas de n variables hay?
- b. Demostrar que utilizando solo compuertas AND/OR no alcanza para definir todas funciones de n variables.