TEMA 2: ÁLGEBRA DE BOOLE

Índice:

TEMA 2: ÁLGEBRA DE BOOLE

Bibliografía:

- T.L.Floyd. Fundamentos de Sistemas Digitales.
  - Cap. 3: Puertas Lógicas
  - Cap. 4: Álgebra de Boole y Simplificación Lógica
  - Cap. 5: Lógica Combinacional
- C.Blanco. Fundamentos de Electrónica Digital.
  - Cap. 2: Álgebra de Boole y Funciones Lógicas
- J.Mª Angulo y J. García. Sistemas Digitales y Tecnología de Computadores.
  - Cap 3. Álgebra de Boole
- E. Mandado. Sistemas Electrónicos Digitales.
  - Cap 2. Álgebra de Boole
  - Cap. 3 Sistemas Combinacionales

1. Introducción al Álgebra de Boole. Definición. Axiomas

Es un conjunto de elementos que pueden tomar dos valores perfectamente diferenciados (que representaremos con **0** y **1**) relacionados por los operadores **+** (suma lógica) y · (producto lógico), que cumplen los siguientes axiomas:

- Ambas operaciones son conmutativas:

$$a + b = b + a \qquad\qquad a \cdot b = b \cdot a$$

- Existen dos elementos neutros, uno por operación:

$$0 + a = a \qquad\qquad a \cdot 1 = a$$

- Cada operación es distributiva respecto a la otra:

$$a\cdot (b+c) = (a\cdot b) + (a\cdot c)$$
$$a+(b\cdot c) = (a+b)\cdot (a+c)$$

- Para todo elemento a existe un elemento complementario **ā**, que cumple:

$$a+\overline{a}=1 \qquad\qquad a\cdot\overline{a}=0$$

Leyes y Teoremas

Principio de dualidad: Cada identidad deducida de los anteriores axiomas permanece válida si las operaciones de suma lógica y producto lógico y los elementos 0 y 1 se intercambian entre si.

Idempotencia:

$$\left.\begin{array}{l} a+a=a \\ a\cdot a=a \end{array}\right\}\forall a\in B \qquad \left.\begin{array}{l} a+1=1 \\ a\cdot 0=0 \end{array}\right\}\forall a\in B$$

Estas dos leyes, junto con el postulado que establece la existencia del elemento neutro, definen la suma y el producto lógico:

a + b	S
0 + 0	0
0 + 1	1
1 + 0	1
1 + 1	1

a · b	P
0 · 0	0
0 · 1	0
1 · 0	0
1 · 1	1

Ley de Absorción:

$$\left.\begin{array}{r} a + a \cdot b = a \\ a \cdot (a + b) = a \end{array}\right\} \forall a, b \in B$$

Involución:

$$\overline{\overline{a}} = a \qquad \forall a \in B$$

Asociatividad:

$$\left.\begin{array}{l} a + (b + c) = (a + b) + c \\ a \cdot (b \cdot c) = (a \cdot b) \cdot c \end{array}\right\} \forall a, b, c \in B$$

Teorema del consenso:

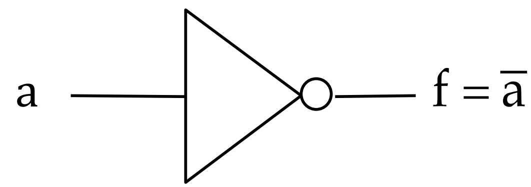
$$ab + \overline{a}c = ab + \overline{a}c + bc$$

Teoremas de DeMorgan:

$$\left.\begin{array}{l} \overline{a + b} = \overline{a} \cdot \overline{b} \\ \overline{a \cdot b} = \overline{a} + \overline{b} \end{array}\right\} \forall a, b \in B$$

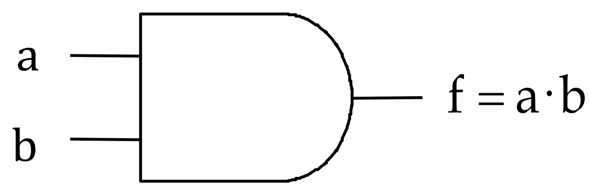
Puertas Lógicas Digitales: Puertas Básicas

Inversor o Negador:



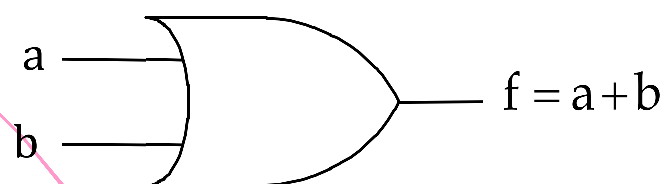
a	f
0	1
1	0

Puerta AND:



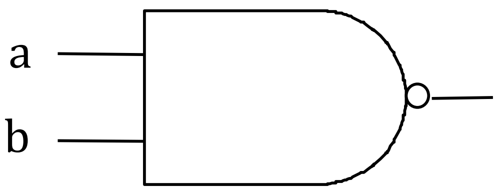
a	b	f
0	0	0
0	1	0
1	0	0
1	1	1

Puerta OR:



a	b	f
0	0	0
0	1	1
1	0	1
1	1	1

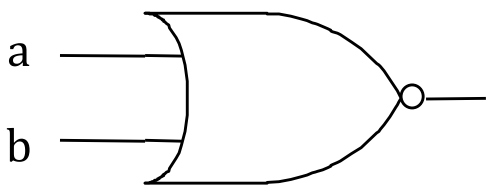
Puerta NAND:



$$f = \overline{a \cdot b}$$

a	b	f
0	0	1
0	1	1
1	0	1
1	1	0

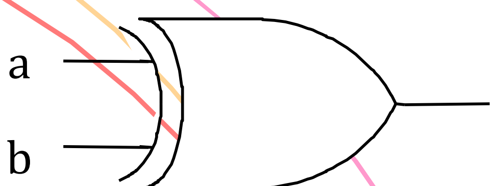
Puerta NOR:



$$f = \overline{a + b}$$

a	b	f
0	0	1
0	1	0
1	0	0
1	1	0

Puerta OR Exclusiva (EXOR o XOR):



$$f = \overline{a}b + a\overline{b} = a \oplus b$$

a	b	f
0	0	0
0	1	1
1	0	1
1	1	0

Funciones Lógicas. Definición

Una función lógica es un conjunto de variables relacionadas entre si por las operaciones básicas definidas de suma lógica, producto lógico y negación:

$$f = f(a,b,c,\ldots)$$

$$f_1(a,b,c) = abc + \overline{a}b\overline{c} + a\overline{b}c + \overline{a}bc$$

$$f_2(a,b,c,d) = a + bc + a(\overline{b} + d)(c + \overline{d})$$

Toda función booleana se comporta como una variable del sistema.
Definimos un término suma como una suma de variables bien en su forma directa o complementada:

$$\overline{a} + b + \overline{c}; \qquad a + \overline{b} + c;$$

Si, por el contrario, dichas variables están relacionadas mediante productos lógicos diremos que se trata de un término producto:

$$\overline{a}\cdot b\cdot \overline{c}; \qquad a\cdot \overline{b}\cdot c;$$

Funciones Lógicas. Representación estándar

Cuando relacionamos dos o más términos producto mediante la suma lógica, la expresión resultante diremos que queda expresada en forma de Suma de Productos:

$$f_1(a,b,c) = abc + \overline{a}b\overline{c} + a\overline{b}c + \overline{a}bc$$

Cuando relacionamos dos o más términos suma mediante el producto lógico, la expresión resultante estará expresada en forma de Producto de Sumas:

$$f_2(a,b,c,d) = (a+\overline{b}+c+d)(a+b+c+\overline{d})$$

Se llama término canónico o estándar de una función lógica a todo producto o suma en la cual aparecen todas las variables que forman parte de la función, bien sea en su forma directa o inversa. En las funciones:

$$f(a,b,c) = ab\overline{c} + a\overline{b}\overline{c} + ac + \overline{b}c$$

$ab\overline{c}$ y $a\overline{b}\overline{c}$ son productos canónicos

$$f(a,b,c) = (b+\overline{c})(a+\overline{b}+\overline{c})(a+c)(\overline{a}+\overline{b}+\overline{c})$$

$(a+\overline{b}+\overline{c})$ y $(\overline{a}+\overline{b}+\overline{c})$ son sumas canónicas

Una función formada únicamente por términos canónicos diremos que es una función canónica o estándar.

Las expresiones en forma estándar pueden expresarse de forma mas sencilla a través de su equivalente numérico:

$$f(a,b,c)=\bar{a}\bar{b}\bar{c}+\bar{a}b\bar{c}+a\bar{b}\bar{c}+abc$$

$$\bar{a}\bar{b}\bar{c}=000_2=0_{10}$$

$$\bar{a}b\bar{c}=010_2=2_{10}$$

$$a\bar{b}\bar{c}=100_2=4_{10}$$

$$abc=111_2=7_{10}$$

De forma similar:

$$f(a,b,c)=\bar{a}\bar{b}\bar{c}+\bar{a}b\bar{c}+a\bar{b}\bar{c}+abc=\sum_3(0,2,4,7)$$

$$f=(a+b+\bar{c}+d)(a+\bar{b}+\bar{c}+d)(a+b+c+\bar{d})(\bar{a}+\bar{b}+\bar{c}+\bar{d})=\prod_4(2,6,1,15)$$

Equivalencia entre Formatos.

Sea la función expresada en forma de suma de productos:

$$f(a,b,c) = \bar{a}\bar{b}c + a\bar{b}\bar{c} + a\bar{b}c = \sum_{3}(1,4,5)$$

La expresión negada de esta función estará compuesta por todos los elementos que no la cumplen:

$$\bar{f} = \sum_{3}(0,2,3,6,7)$$

En forma algebraica:

$$\bar{f} = \bar{a}\bar{b}\bar{c} + \bar{a}b\bar{c} + \bar{a}bc + ab\bar{c} + abc$$

Y puesto que

$$f = \bar{\bar{f}} = \overline{\bar{a}\bar{b}\bar{c} + \bar{a}b\bar{c} + \bar{a}bc + ab\bar{c} + abc}$$

Y aplicando ahora los teoremas de DeMorgan:

$$f = \overline{\bar{a}\bar{b}\bar{c} + \bar{a}b\bar{c} + \bar{a}bc + ab\bar{c} + abc} =$$

$$= \overline{\bar{a}\bar{b}\bar{c}} \cdot \overline{\bar{a}b\bar{c}} \cdot \overline{\bar{a}bc} \cdot \overline{ab\bar{c}} \cdot \overline{abc} =$$

$$= (a + b + c)(a + \bar{b} + c)(a + \bar{b} + \bar{c})(\bar{a} + \bar{b} + c)(\bar{a} + \bar{b} + \bar{c})$$

Luego:

$$f = \sum_{3}(1,4,5) = \prod_{3}(0,2,3,6,7)$$

Tabla de Verdad

Una tabla de verdad de una función lógica es una forma de representación de la misma, en la que se indica el valor (0 ó 1) que toma la función para cada una de las combinaciones posibles de las variables de las cuales depende, expresadas en forma de suma de productos.

$$f = \overline{a}\,\overline{b}c + \overline{a}bc + a\overline{b}\,\overline{c} + abc$$

$$f = \sum_3 (1,3,4,7)$$

$$f = \prod_3 (0,2,5,6)$$

abc	f
000	0
001	1
010	0
011	1
100	1
101	0
110	0
111	1

Simplificación mediante Álgebra de Boole

Se basa en la aplicación sistemática de los axiomas, teoremas y leyes ya comentadas. Partiendo de la expresión canónica, básicamente se apoya en la propiedad:

$$abc + \ldots + \overline{a}bc = bc + \ldots$$

$$(a+b+c)(\overline{a}+b+c)\ldots = (b+c)\ldots$$

Ejemplos:

$$abc + ab\overline{c} = ab(c+\overline{c})$$

$$(\overline{a}+b+c)(\overline{a}+\overline{b}+c)\ldots = (\overline{a}+c)+b\overline{b}$$

$$abc + ab\overline{c} + a\overline{b}\overline{c} + a\overline{b}c = ab(c+\overline{c}) + a\overline{b}(c+\overline{c}) =$$

$$= ab + a\overline{b} = a(b+\overline{b}) = a$$

$$(a+b+c)(a+b+\overline{c})(a+\overline{b}+c) =$$

$$= \left[(a+b)+c\overline{c}\right]\left[(a+c)+b\overline{b}\right] = (a+b)(a+c)$$

Simplificación mediante Tablas (o mapas) de Karnaugh

Se basa en sistematizar la aplicación del método algebraico ya descrito mediante la construcción de tablas.

Estas tablas están constituidas por celdas a las que asignaremos una combinación de tal forma que cada una de ellas esta rodeada únicamente por otras en las que difiere en una sola variable.

De 3 variables:

a \ bc	00	01	11	10
0	0	1	3	2
1	4	5	7	6

De 4 variables:

ab \ cd	00	01	11	10
00	0	1	3	2
01	4	5	7	6
11	12	13	15	14
10	8	9	11	10

De 5 Variables:

ab \ cde	000	001	011	010	110	111	101	100
00	0	1	3	2	6	7	5	4
01	8	9	11	10	14	15	13	12
11	24	25	27	26	30	31	29	28
10	16	17	19	18	22	23	21	20

Para completar las tablas procederemos como si se tratase de una tabla de verdad, colocando un 1 en las casillas de los términos que cumplen la función en forma de suma de productos y un 0 en las que no.

Simplificación en forma de PoS y SoP

$$f = \sum_4 (0,2,3,4,6,7,8,10)$$

ab \ cd	00	01	11	10
00	1	0	1	1
01	1	0	1	1
11	0	0	0	0
10	1	0	0	1

$\bar{a}\bar{d}$ $\bar{a}c$ $\bar{b}\bar{d}$

$$f = \bar{a}\bar{d} + \bar{b}\bar{d} + \bar{a}c$$

ab \ cd	00	01	11	10
00	1	0	1	1
01	1	0	1	1
11	0	0	0	0
10	1	0	0	1

$(c+\bar{d})$ $(\bar{a}+\bar{b})$ $(\bar{a}+\bar{d})$

$$f = (\bar{a}+\bar{d})(\bar{a}+\bar{b})(c+\bar{d})$$

$$f = \sum(0,2,4,6,9,13,16,18,20,22,26,29,30,31)$$

ab \ cde	000	001	011	010	110	111	101	100
00	**1**	**0**	**0**	**1**	**1**	**0**	**0**	**1**
01	**0**	**1**	**0**	**0**	**0**	**0**	**1**	**0**
11	**0**	**0**	**0**	**1**	**1**	**1**	**1**	**0**
10	**1**	**0**	**0**	**1**	**1**	**0**	**0**	**1**

$\bar{a}b\bar{d}e$

$abce$

$\bar{b}\bar{e}$

$ad\bar{e}$

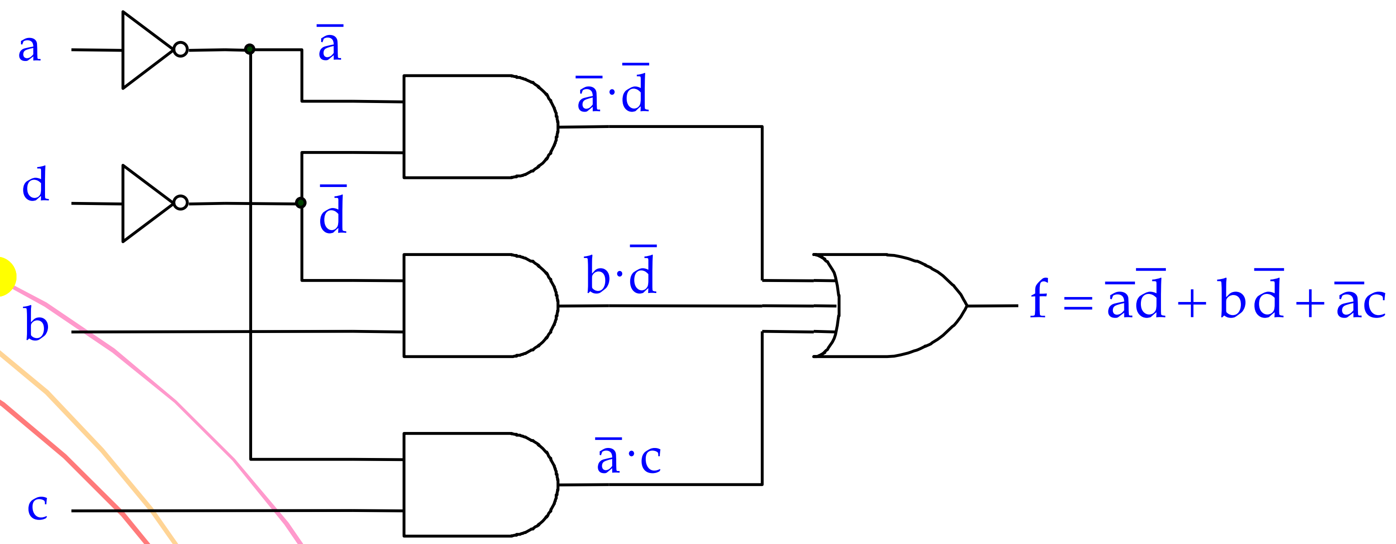
$$f = (\bar{a}b\bar{d}e + abce + \bar{b}\bar{e} + ad\bar{e})$$

Conjuntos Completos. Implementación

Un conjunto completo esta compuesto por un grupo de puertas mínimo necesario que permita implementar cualquier función:

Ejemplos: Inversor, puerta AND y puerta OR. Puerta NAND. Puerta NOR

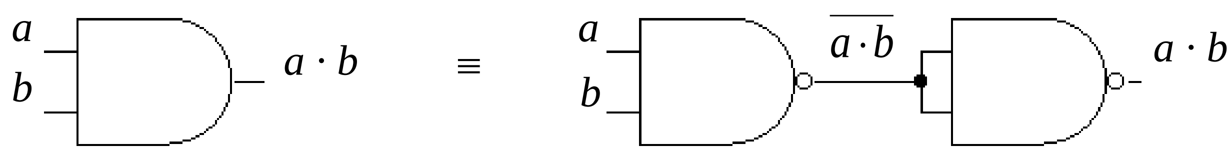
$$f = \overline{a}\,\overline{d} + b\,\overline{d} + \overline{a}c$$



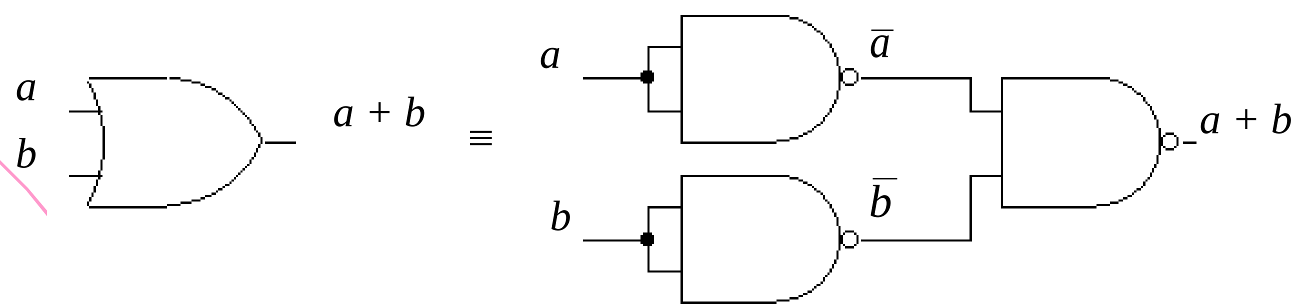
Puerta NAND como Elemento Universal

Inversor:

Puerta AND:

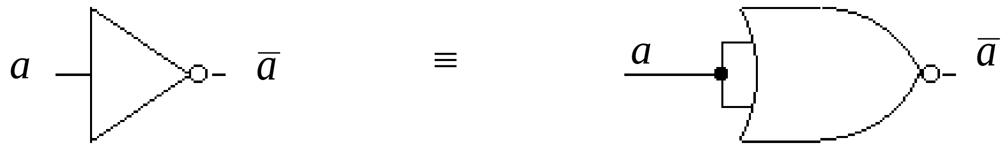


Puerta OR:

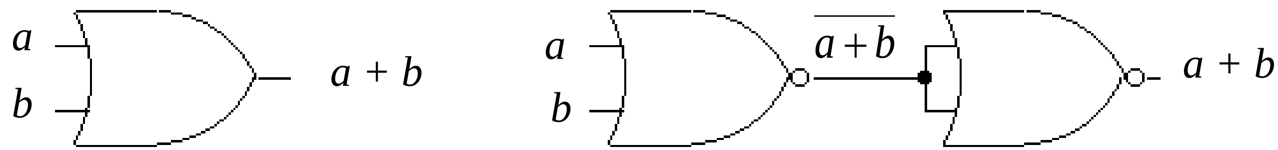


Puerta NOR como Elemento Universal

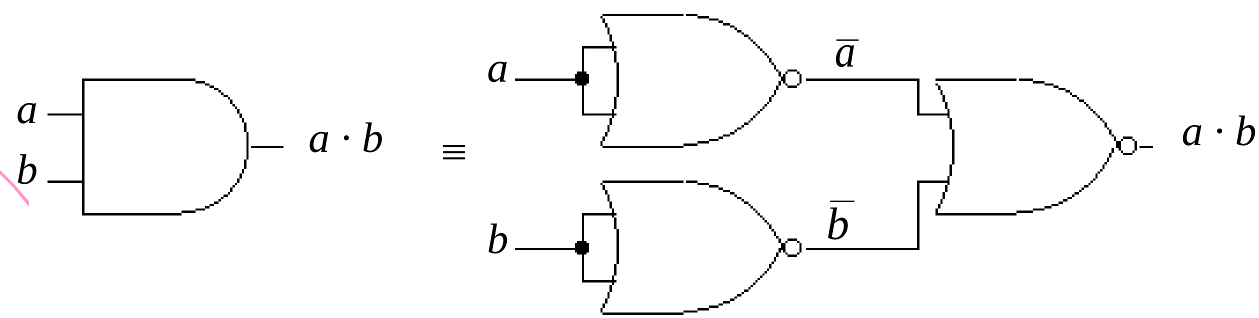
Inversor:



Puerta OR:

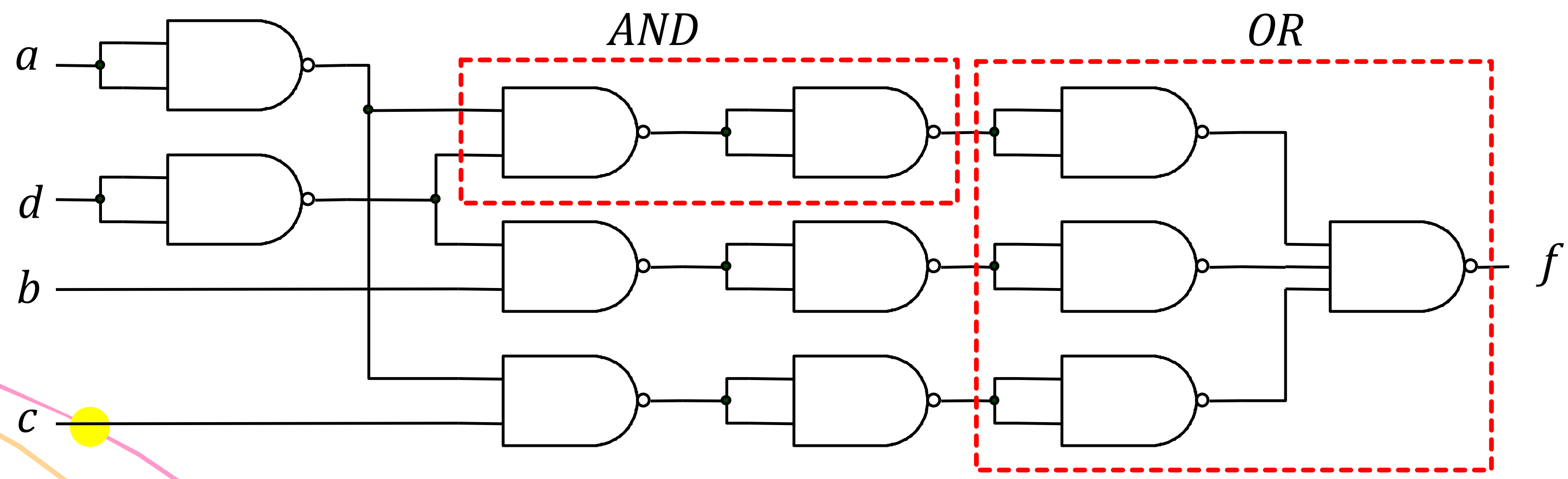


Puerta AND:



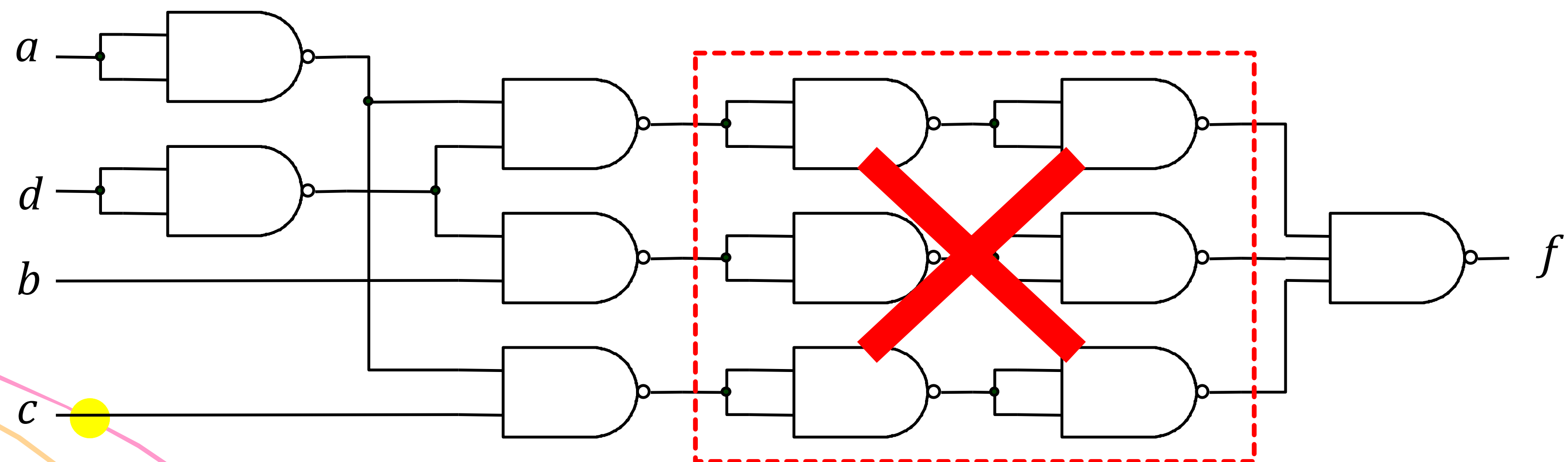
Conjuntos completos. Implementación

$$f = \overline{a}\overline{d} + b\overline{d} + \overline{a}c$$



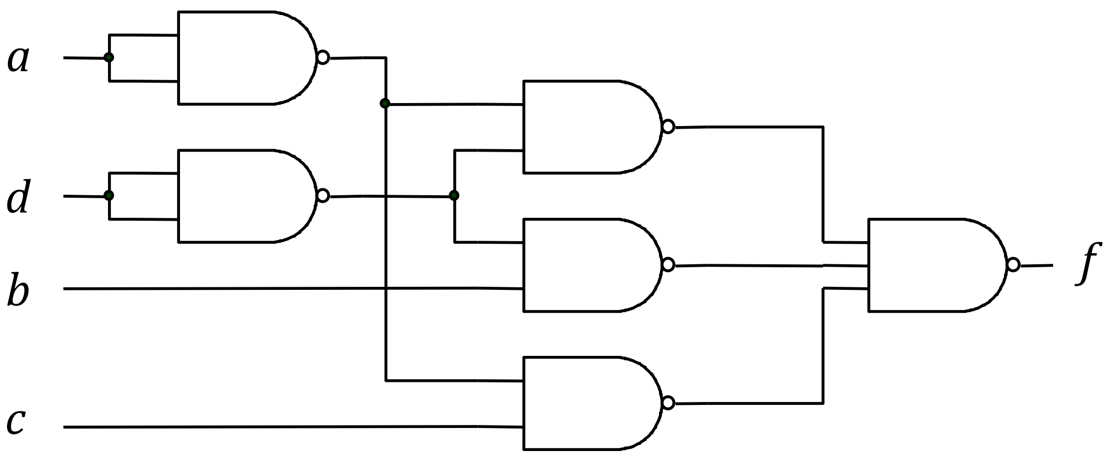
Conjuntos completos. Implementación

$$f = \overline{a}\overline{d} + b\overline{d} + \overline{a}c$$



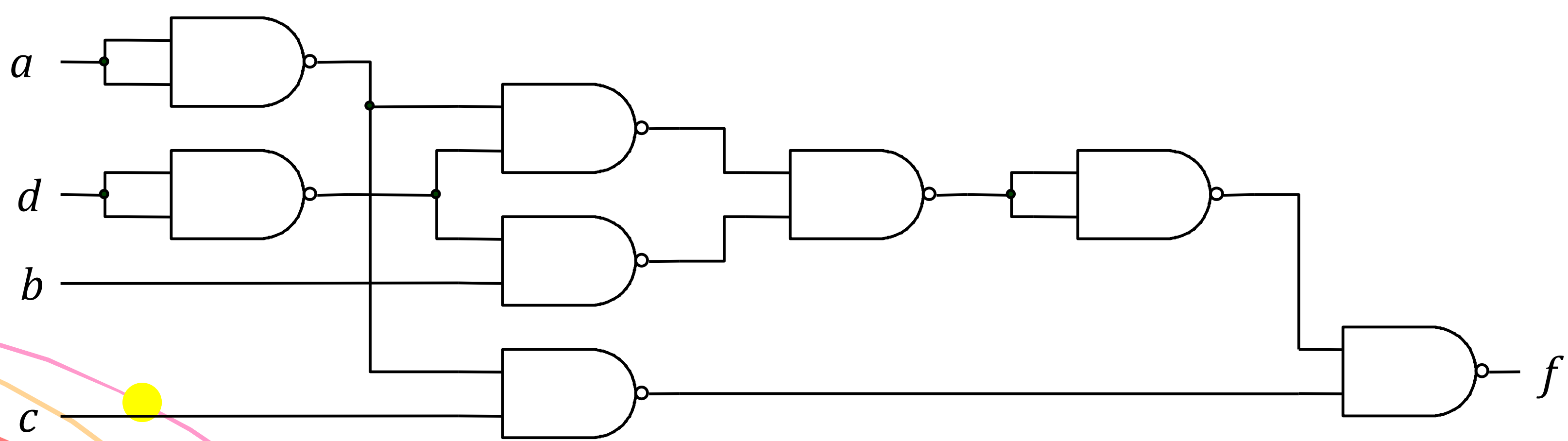
Conjuntos completos. Implementación

$$f = \overline{a}\,\overline{d} + b\overline{d} + \overline{a}c = \overline{\overline{\overline{a}\,\overline{d} + b\overline{d} + \overline{a}c}} = \overline{\overline{\overline{a}\,\overline{d}} \cdot \overline{b\overline{d}} \cdot \overline{\overline{a}c}}$$



Conjuntos completos. Implementación

$$f = \bar{a}\bar{d} + b\bar{d} + \bar{a}c = \overline{\overline{\bar{a}\bar{d} + b\bar{d} + \bar{a}c}} = \overline{\overline{\bar{a}\bar{d}}\cdot\overline{b\bar{d}}\cdot\overline{\bar{a}c}} = \overline{\overline{\overline{\overline{\bar{a}\bar{d}}\cdot\overline{b\bar{d}}}}\cdot\overline{\bar{a}c}}$$



Funciones Incompletas. Simplificación

Son aquellas que no tienen un valor definido para todas las posibles combinaciones de las variables de las que dependen.

abc	f
000	0
001	X
010	X
011	1
100	1
101	0
110	X
111	0

$$f = \sum_{3}(3,4) + \sum_{\varnothing}(1,2,6)$$

a \ bc	00	01	11	10
0	0	X	1	X
1	1	0	0	X

$$f = \overline{a}b + a\overline{c}$$