Tema 2: **Fundamentos de Sistemas Digitales**

Contenido

1. **Introducción al Álgebra de Boole**
   - Definición. Axiomas
   - Leyes y Teoremas básicos
2. **Puertas Lógicas Digitales**
   - Puertas básicas
3. **Funciones Lógicas**
   - Representación en forma estándar
   - Conversión y equivalencia entre formas
   - Tablas de verdad
4. **Simplificación e Implementación de Funciones**
   - Mediante las propiedades del álgebra de Boole
   - Tablas de Karnaugh de 3, 4 y 5 variables
   - Simplificación en forma de SOP y POS
   - Conjuntos Completos. Implementación
   - Funciones incompletas. Simplificación

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:

$$\mathbf{a + b = b + a} \qquad\qquad \mathbf{a \cdot b = b \cdot a}$$

- Existen dos elementos neutros, uno por operación:

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

- Cada operación es distributiva respecto a la otra:

$$\mathbf{a \cdot (b+c) = (a \cdot b) + (a \cdot c)}$$

$$\mathbf{a + (b \cdot c) = (a+b) \cdot (a+c)}$$

- Para todo elemento a existe un elemento complementario $\overline{\mathbf{a}}$, que cumple:

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

1. Introducción al Álgebra de Boole

Leyes y Teoremas

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

- *Idempotencia*:

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

- *Dominancia de elementos*

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

- 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

1. Introducción al Álgebra de Boole

Leyes y Teoremas

- *Ley de Absorción*:

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

- *Involución*:

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

- *Asociatividad*:

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

- Teorema del consenso

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

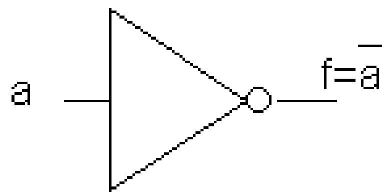
- Teoremas de De Morgan

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

2. Puertas Lógicas Digitales

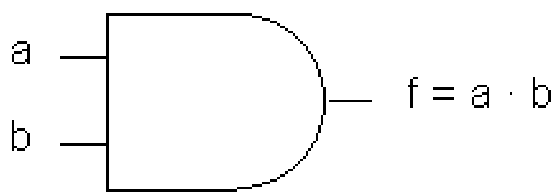
Puertas Básicas

- *Inversor o Negador*



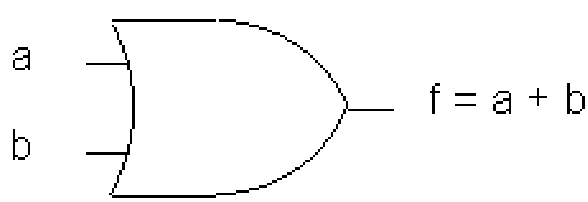
a	f
0	1
1	0

- *Puerta AND*



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

- *Puerta OR*

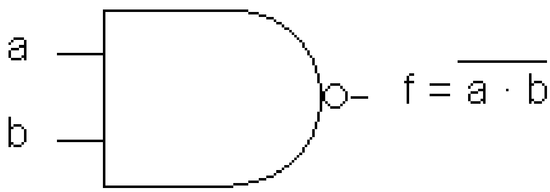


a	b	f
0	0	0
0	1	1
1	0	1
1	1	1

2. Puertas Lógicas Digitales

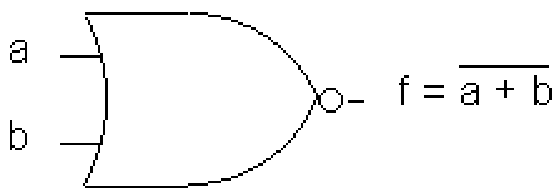
Puertas Básicas

- *Puerta NAND*



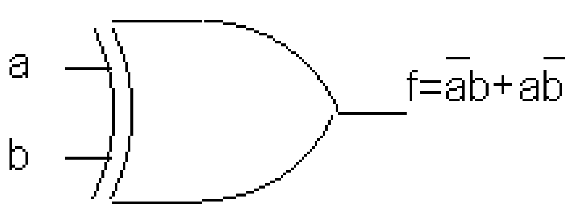
a	b	f
0	0	1
0	1	1
1	0	1
1	1	0

- *Puerta NOR*



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

- *Puerta OR Exclusiva (EXOR o XOR)*



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

3. Funciones Lógicas

Definición

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

$$\mathbf{f} = \mathbf{f}(\mathbf{a},\mathbf{b},\mathbf{c},\ldots)$$

$$\mathbf{f}_1(\mathbf{a},\mathbf{b},\mathbf{c}) = \mathbf{abc} + \overline{\mathbf{a}}\mathbf{b}\overline{\mathbf{c}} + \mathbf{a}\overline{\mathbf{b}}\mathbf{c} + \overline{\mathbf{a}}\mathbf{bc}$$

$$\mathbf{f}_2(\mathbf{a},\mathbf{b},\mathbf{c},\mathbf{d}) = \mathbf{a} + \mathbf{bc} + \mathbf{a}(\overline{\mathbf{b}} + \mathbf{d})(\mathbf{c} + \overline{\mathbf{d}})$$

- 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 en su forma complementada:

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

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

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

3. Funciones Lógicas

Tablas 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.

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

$$\mathbf{f} = \sum_{3}(1,3,4,7)$$

$$\mathbf{f} = \prod_{3}(0, 2, 5, 6)$$

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

3. Funciones Lógicas

Representación en forma estándar (i)

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

$$\mathbf{f}_1(\mathbf{a},\mathbf{b},\mathbf{c}) = \mathbf{abc} + \overline{\mathbf{a}}\mathbf{b}\overline{\mathbf{c}} + \mathbf{a}\overline{\mathbf{b}}\mathbf{c} + \overline{\mathbf{a}}\mathbf{bc}$$

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

$$\mathbf{f}_2(\mathbf{a},\mathbf{b},\mathbf{c},\mathbf{d}) = (\mathbf{a} + \overline{\mathbf{b}} + \mathbf{c} + \mathbf{d})(\mathbf{a} + \mathbf{b} + \mathbf{c} + \overline{\mathbf{d}})$$

- Se llama término canónico 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.

- Ejemplo: en las funciones:

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

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

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

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

3. Funciones Lógicas

Representación en forma estándar (ii)

- Una función formada únicamente por términos canónicos se denomina función canónica o estándar.

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

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

$$\overline{\mathbf{a}}\,\overline{\mathbf{b}}\,\overline{\mathbf{c}} = 000_2 = 0_{10}$$

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

$$\mathbf{a}\overline{\mathbf{b}}\,\overline{\mathbf{c}} = 100_2 = 4_{10}$$

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

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

- De igual forma:

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

$$= \prod_4 (2, 6, 1, 15)$$

Simplificación mediante Álgebra de Boole

- Se basa en la utilización sistemática de los postulados o axiomas, y teoremas y leyes ya comentadas. A partir de la expresión canónica, básicamente se apoya en la propiedad:

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

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

- Ejemplos:

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

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

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

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

4.Simplificación e Implementación de Funciones

Tablas (Mapas) de Karnaugh de 3 y 4 variables

- 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.
- Las celdas están distribuidas de forma que cada una de ellas esta rodeada únicamente por otras en las que difiere en una sola variable (código Gray).
- 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

4.Simplificación e Implementación de Funciones

Tablas (o Mapas) de Karnaugh de 5 variables

- 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 ‘1’ en las casillas de los términos que cumplen la función en forma de suma de productos.

Simplificación en forma de SOP y POS (i)

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

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**

$\bar{a}\bar{d}$

$\bar{a}c$

$\bar{b}\,\bar{d}$

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

$(c + \bar{d})$

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**

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

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

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

Simplificación en forma de SOP y POS (ii)

$$f = \sum_5 (0,2,4,6,9,13,16,18,20,22,26,29,30,31)$$

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**

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

$abce$

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

$ad\bar{e}$

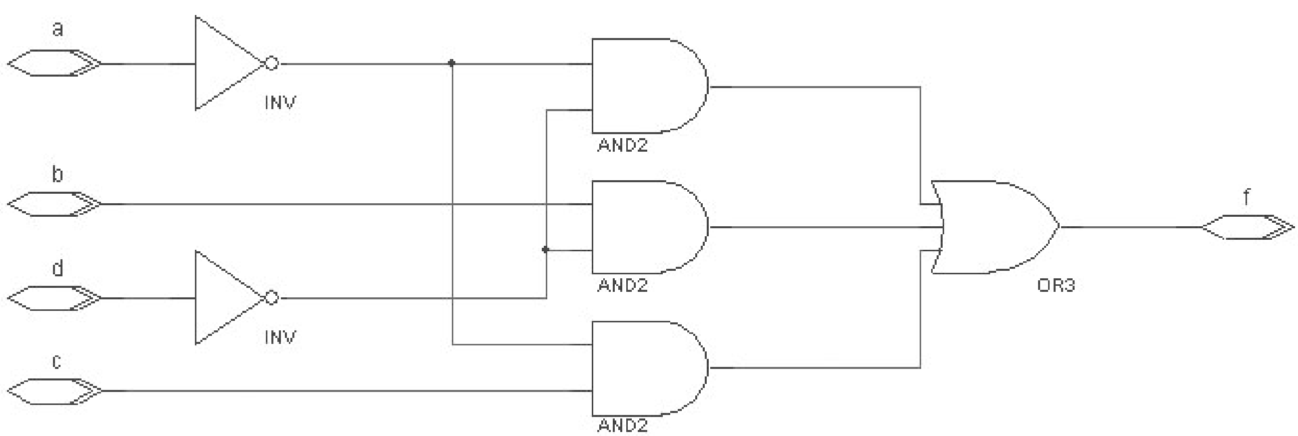
$$f = (\bar{a}b\bar{d}e + abce + \bar{b}\bar{e} + ad\bar{e})$$

4.Simplificación e Implementación de Funciones

Conjuntos Completos. Implementación

- Un conjunto completo esta compuesto por un tipo de puertas mínimo que permita implementar cualquier función:
- Ejemplos:
  - Inversor + puerta AND + puerta OR
  - Puerta NAND
  - Puerta NOR

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

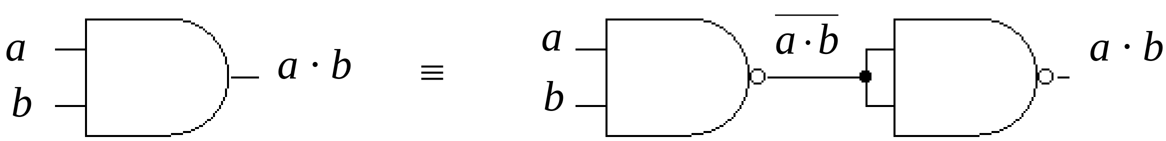


Conjuntos Completos. Implementación

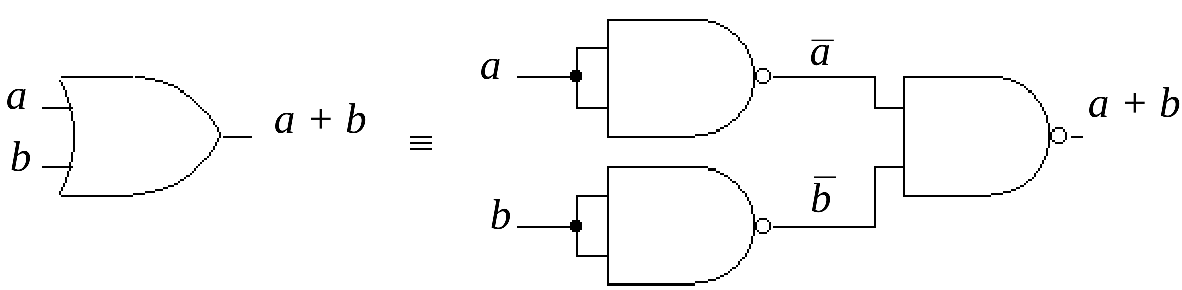
- Puerta NAND como elemento universal

Inversor

Puerta AND



Puerta OR

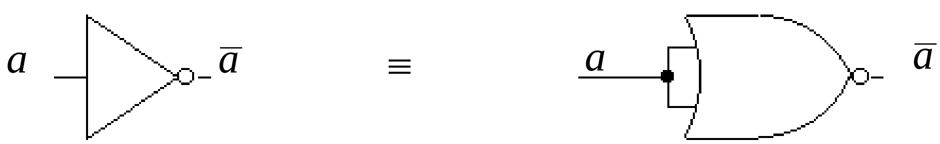


4.Simplificación e Implementación de Funciones

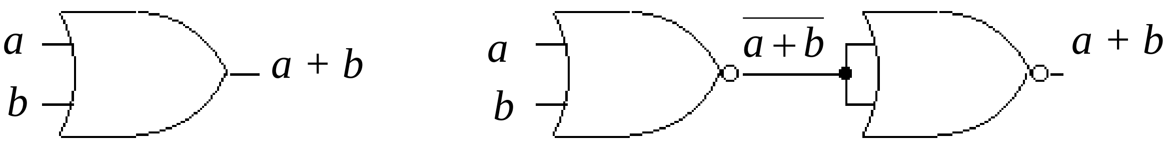
Conjuntos Completos. Implementación

- Puerta NOR como elemento universal

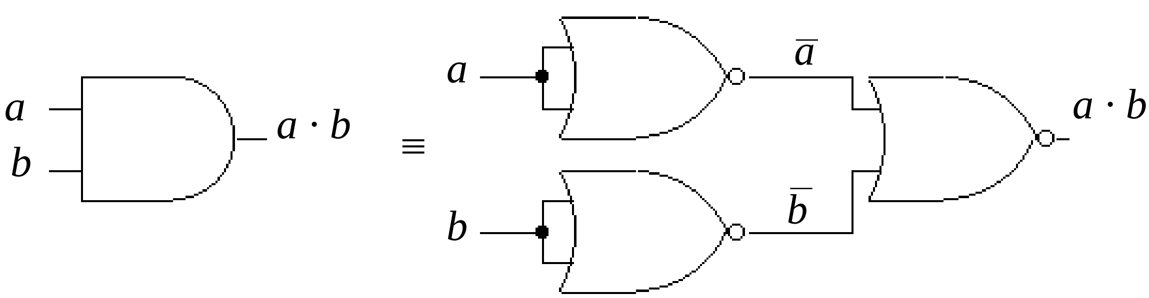
Inversor



Puerta OR



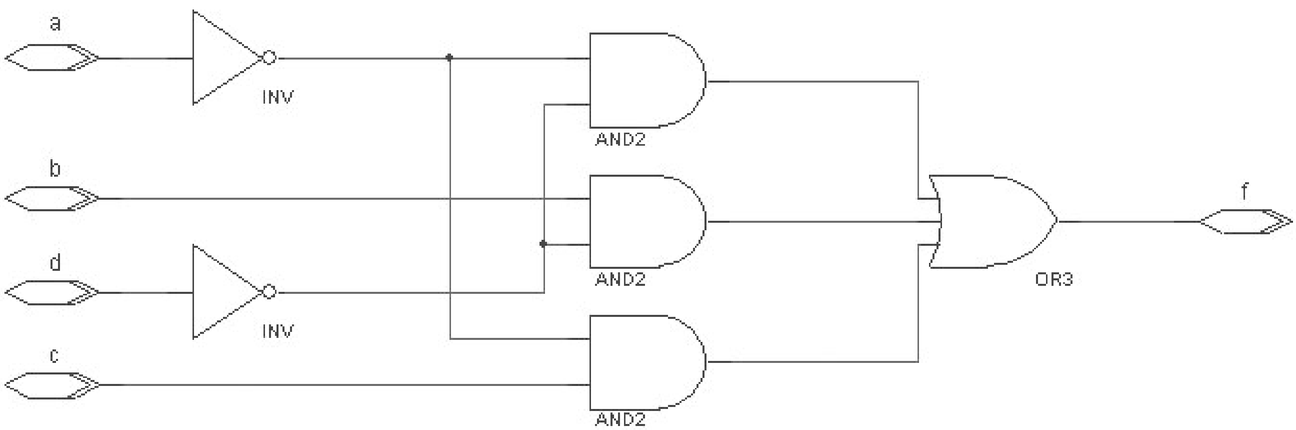
Puerta AND



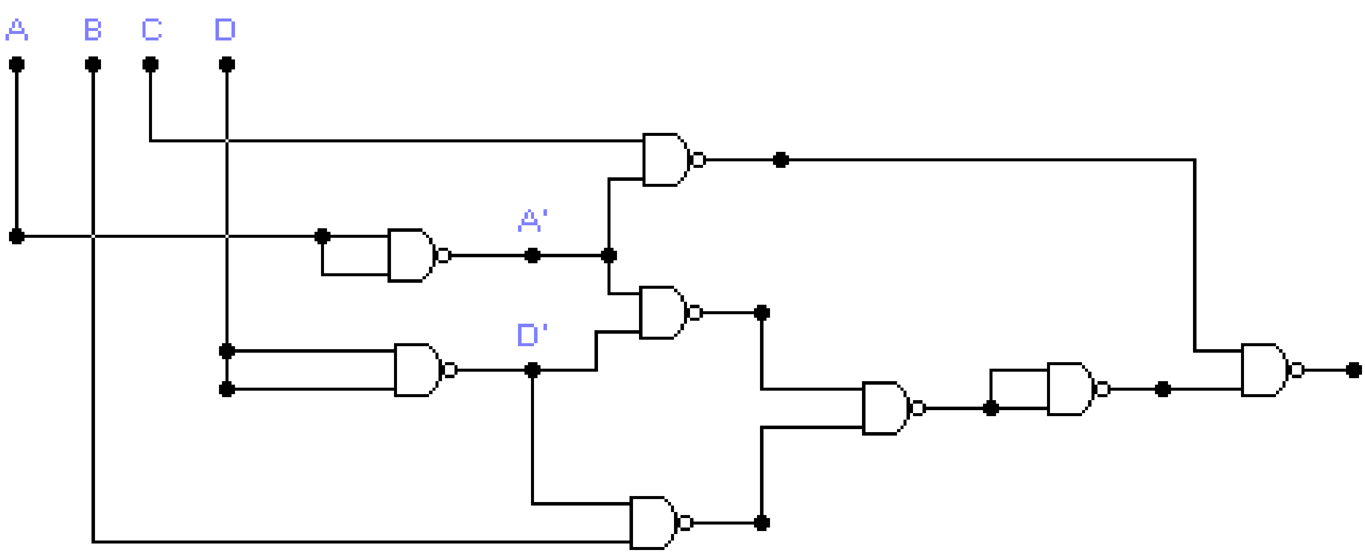
Conjuntos Completos. Implementación

- Implementación de la función anterior solamente con puertas NAND de 2 entradas:

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



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



4.Simplificación e Implementación de Funciones

Funciones Incompletas. Simplificación

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

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

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

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

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