

Implementación de funciones Booleanas

Andrés Prieto-Moreno

19th November 2002

1 Introducción

Para nosotros implementar es construir, o lo que técnicamente se llama *síntesis*, por lo tanto en este tema se va a estudiar como a partir de las ecuaciones booleanas se puede construir físicamente un circuito digital. Para ello nos basaremos en las *puertas lógicas*, que son los elementos básicos de la electrónica digital, y que son capaces de realizar físicamente las operaciones básicas del álgebra de Boole (+, *, -, ⊕).

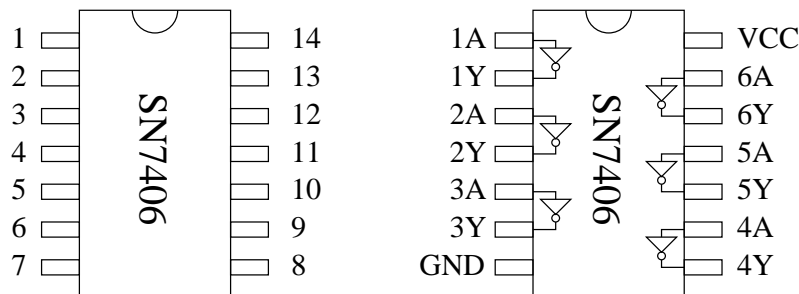
Las puertas lógicas se pueden comprar en las tiendas de electrónica y suelen venir encapsuladas en lo que se suele llamar un circuito integrado o *chip* (ver figura 1). Una puerta lógica esta formada por una serie de transistores y diodos que según como se conecten entre ellos forman una u otra puerta. A su vez los transistores están formados por composiciones semiconductoras como el Si, Ge, etc.

Actualmente, aunque las puertas sean la base de los circuitos digitales, no se suelen utilizar como tal, al contrario, se utilizan otros circuitos más generales que permiten ser configurados para realizar funciones más complicadas. En lugar de tener que unir varios chips básicos, se utiliza sólo uno programable, es lo que se llama implementación con dispositivos lógicos programables o PLDs. De todas formas en las tiendas se siguen encontrando los chips básicos y se pueden consultar en los libros de referencia. Por ejemplo Texas Instruments publica anualmente el Data Book de las puertas TTL que fabrica.

2 Puertas Lógicas. Implementación de funciones

En la figura 3 se puede encontrar una tabla con el nombre, descripción y función que realiza cada puerta lógica. Mediante las puertas anteriores se pueden implementar todas las funciones booleanas que describen teóricamente a los sistemas digitales. No habrá una única forma de implementación, por lo que se recomienda intentar el mínimo número de puertas lógicas para realizar el circuito. Es muy importante fijarse en las equivalencias, como por ejemplo el hecho de que una NAND es una AND+NOT, y muy importante, acordarse de las puertas XOR y NXOR. Veamos un ejemplo de lo anterior (figura 2), supongamos que queremos implementar la función:

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



CHIP 7406: Contiene 6 puertas inversoras

Figure 1: El CHIP 7406 que implementa 6 puertas inversoras.

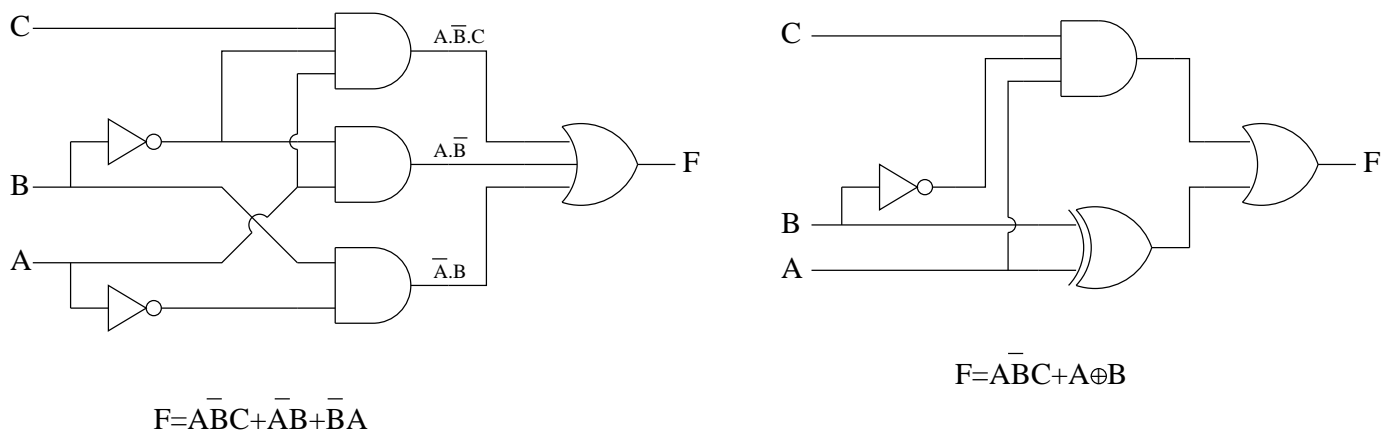


Figure 2: Ejemplo de implementación de funciones.

Podemos observar que la función se puede implementar utilizando una AND triple, dos AND de dos entradas, una puerta OR de tres entradas y dos puertas inversoras, es decir en total 6 puertas. Pero si nos fijamos en que

$$F = A.\overline{B}.C + \overline{A}.B + \overline{B}.A = A.\overline{B}.C + A \oplus B$$

La función se puede implementar con la misma AND triple, con las dos inversoras y con una puerta XOR, es decir hemos utilizado sólo cuatro puertas.

3 Conjuntos completos

Se dice que un conjunto de puertas es completo cuando con las puertas contenidas en él se pueden construir todas las demás. La utilidad de los conjuntos completos es aprovechar al máximo los circuitos integrados. Generalmente en los chips no viene una puerta aislada, sino que suelen venir varias, por eso si utilizando esas puertas podemos implementar otras se podrá aprovechar al máximo la capacidad del chip.

Ejemplos de conjuntos completos son los siguientes:

1. (AND, NOT)
2. (OR, NOT)
3. (NAND)
4. (NOR)

De estos cuatro conjuntos vamos a estudiar los dos últimos, es decir implementar todas las puertas lógicas básicas utilizando, por un lado puertas NAND y por otro puertas NOR. Cumpliéndose los conjuntos NAND y NOR esta claro que los dos primeros también lo serán, ya que $NAND = AND + NOT$ y $NOR = OR + NOT$. En la figura 4 se puede ver la implementación de todas las puertas lógicas a partir de puertas NOR, y en la figura 5 lo mismo pero a partir de puertas NAND. Merece la pena mencionar que las equivalencias se obtienen aplicando la Ley de Morgan. Con lógica NOR aplicamos Morgan hasta obtener términos en sumas complementados, es decir $\overline{(A+B)}$. Con lógica NAND buscamos términos en productos complementados, es decir $\overline{(A.B)}$

Ejemplo: Expresar la función $F = A.B + C$ en lógica NAND y en lógica NORD.

1. Función expresada como NAND:

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

2. Función expresada como NOR:

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

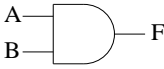
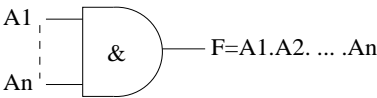

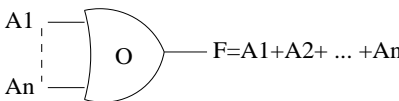
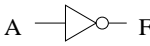
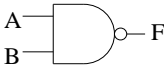
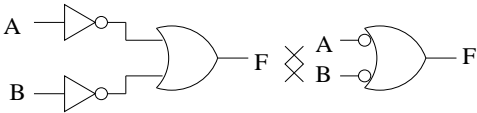
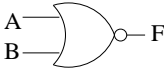
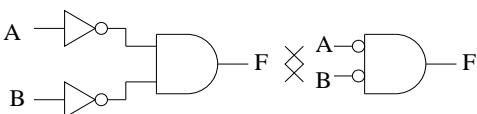

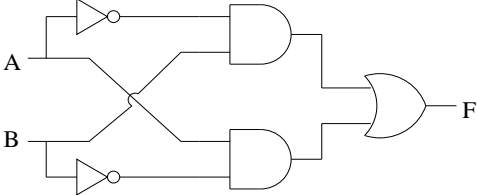

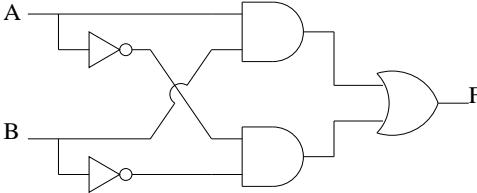
| | SIMBOLO | FUNCION | TABLA | DATOS DE INTERES | | | | | | | | | | | | | | | |
|------|--|--|--|------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|
| AND |  | $F = A.B$ | <table><tr><th>A</th><th>B</th><th>F</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</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> | A | B | F | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 |  |
| A | B | F | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | | | | | | | | | | | | | | | | | |
| 0 | 1 | 0 | | | | | | | | | | | | | | | | | |
| 1 | 0 | 0 | | | | | | | | | | | | | | | | | |
| 1 | 1 | 1 | | | | | | | | | | | | | | | | | |
| OR |  | $F = B+A$ | <table><tr><th>A</th><th>B</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> | A | B | F | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |  |
| A | B | F | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | | | | | | | | | | | | | | | | | |
| 0 | 1 | 1 | | | | | | | | | | | | | | | | | |
| 1 | 0 | 1 | | | | | | | | | | | | | | | | | |
| 1 | 1 | 1 | | | | | | | | | | | | | | | | | |
| NOT |  | $F = \overline{A}$ | <table><tr><th>A</th><th>F</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table> | A | F | 0 | 1 | 1 | 0 | | | | | | | | | | |
| A | F | | | | | | | | | | | | | | | | | | |
| 0 | 1 | | | | | | | | | | | | | | | | | | |
| 1 | 0 | | | | | | | | | | | | | | | | | | |
| NAND |  | $F = \overline{A.B} = \overline{A} + \overline{B}$ | <table><tr><th>A</th><th>B</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> | A | B | F | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 |  |
| A | B | F | | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | | | | | | | | | | | | | | | | | |
| 0 | 1 | 1 | | | | | | | | | | | | | | | | | |
| 1 | 0 | 1 | | | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | | | | | | | | | | | | | | | | | |
| NOR |  | $F = \overline{A+B} = \overline{A}. \overline{B}$ | <table><tr><th>A</th><th>B</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> | A | B | F | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |  |
| A | B | F | | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | | | | | | | | | | | | | | | | | |
| 0 | 1 | 0 | | | | | | | | | | | | | | | | | |
| 1 | 0 | 0 | | | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | | | | | | | | | | | | | | | | | |
| XOR |  OR EXCLUSIVA | $F = A \oplus B = \overline{A}.B + A. \overline{B}$ | <table><tr><th>A</th><th>B</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>0</td></tr></table> | A | B | F | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 |  |
| A | B | F | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | | | | | | | | | | | | | | | | | |
| 0 | 1 | 1 | | | | | | | | | | | | | | | | | |
| 1 | 0 | 1 | | | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | | | | | | | | | | | | | | | | | |
| NXOR |  NOR EXCLUSIVA | $F = \overline{A \oplus B} = \overline{\overline{A}.B + A. \overline{B}} = \overline{\overline{A}.B} + \overline{A. \overline{B}} = (A + \overline{B})(\overline{A} + B) = A.B + \overline{B}. \overline{A}$ | <table><tr><th>A</th><th>B</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>1</td></tr></table> | A | B | F | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 |  |
| A | B | F | | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | | | | | | | | | | | | | | | | | |
| 0 | 1 | 0 | | | | | | | | | | | | | | | | | |
| 1 | 0 | 0 | | | | | | | | | | | | | | | | | |
| 1 | 1 | 1 | | | | | | | | | | | | | | | | | |

Figure 3: Puertas lógicas

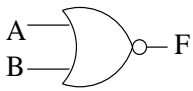
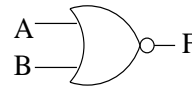
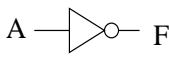

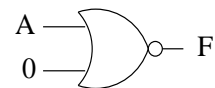
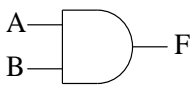
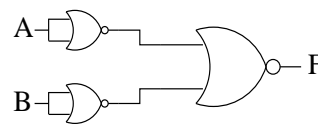
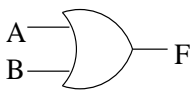
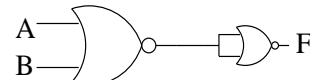
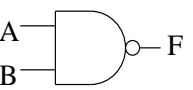
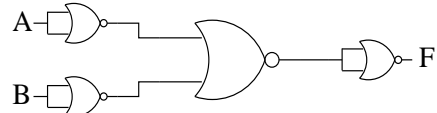

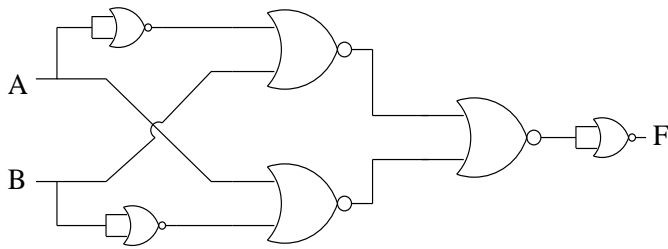
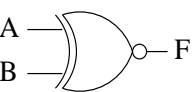
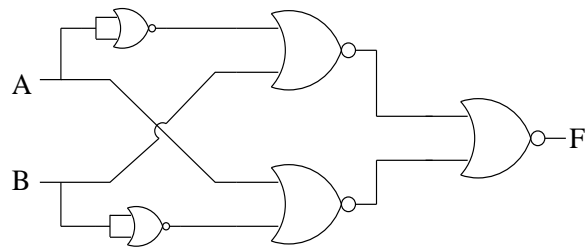
| | SIMBOLO | FUNCION | Circuito Equivalente con puertas NOR | | | | | | | | | | | | | | | |
|------|--|--|--|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| NOR |  | $F = \overline{A+B} = \bar{A}.\bar{B}$ | <table><tr><th>A</th><th>B</th><th>F</th></tr><tr><td>0</td><td>0</td><td>1</td></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>  | A | B | F | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| A | B | F | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | | | | | | | | | | | | | | | | |
| 0 | 1 | 0 | | | | | | | | | | | | | | | | |
| 1 | 0 | 0 | | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | | | | | | | | | | | | | | | | |
| NOT |  | $F = \overline{A+A} = \bar{A}$ |   | | | | | | | | | | | | | | | |
| AND |  | $F = A.B = \overline{\overline{A.B}} = \overline{\overline{A} + \overline{B}}$ |  | | | | | | | | | | | | | | | |
| OR |  | $F = A+B = \overline{\overline{A+B}} = \overline{\overline{A}.\overline{B}}$ |  | | | | | | | | | | | | | | | |
| NAND |  | $F = \overline{A.B} = \overline{\overline{\overline{A.B}}} = \overline{\overline{A} + \overline{B}}$ |  | | | | | | | | | | | | | | | |
| XOR |  OR EXCLUSIVA | $F = A \oplus B = \bar{A}.B + A.\bar{B} = \overline{\overline{\bar{A}.B + A.\bar{B}}} = \overline{\overline{\bar{A}.B} . \overline{A.\bar{B}}} = \overline{(\overline{\bar{A}.B}) . (\overline{A.\bar{B}})} = \overline{(\overline{\bar{A}} + \overline{B}) . (\overline{A} + \overline{\bar{B}})} = \overline{(\overline{\bar{A}} + \overline{B}) + (\overline{A} + \overline{\bar{B}})}$ |  | | | | | | | | | | | | | | | |
| NXOR |  NOR EXCLUSIVA | $F = \overline{A \oplus B} = \overline{\bar{A}.B + A.\bar{B}} = \overline{\bar{A}.B} . \overline{A.\bar{B}} = (\overline{\bar{A}.B}) . (\overline{A.\bar{B}}) = (\overline{\bar{A}} + \overline{B}) . (\overline{A} + \overline{\bar{B}}) = (\overline{\bar{A}} + \overline{B}) + (\overline{A} + \overline{\bar{B}})$ |  | | | | | | | | | | | | | | | |

Figure 4: Obtención de las puertas lógicas mediante NOR.

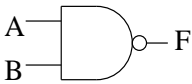
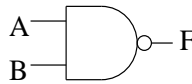
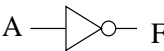

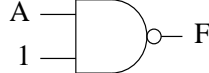
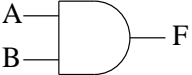
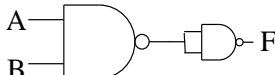

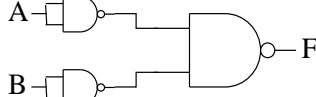
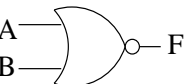
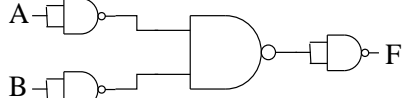
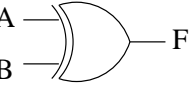
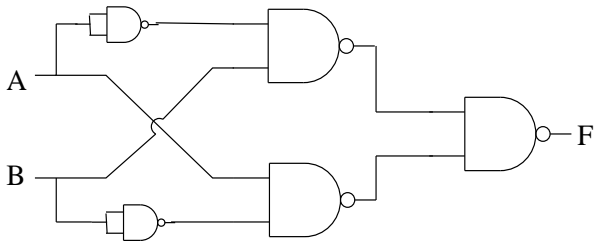
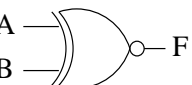
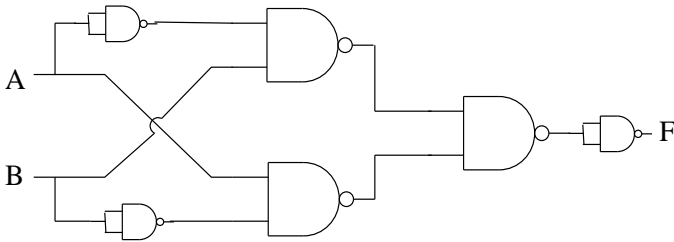
| | SIMBOLO | FUNCION | Circuito Equivalente con puertas NAND | | | | | | | | | | | | | | | |
|------|--|--|--|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| NAND |  | $F = \overline{A \cdot B} = \bar{A} + \bar{B}$ | <table><tr><th>A</th><th>B</th><th>F</th></tr><tr><td>0</td><td>0</td><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>  | A | B | F | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 |
| A | B | F | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | | | | | | | | | | | | | | | | |
| 0 | 1 | 1 | | | | | | | | | | | | | | | | |
| 1 | 0 | 1 | | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | | | | | | | | | | | | | | | | |
| NOT |  | $F = \overline{A} = \bar{A}$ |   | | | | | | | | | | | | | | | |
| AND |  | $F = A \cdot B = \overline{\overline{A \cdot B}}$ |  | | | | | | | | | | | | | | | |
| OR |  | $F = A + B = \overline{\overline{A + B}} = \overline{\bar{A} \cdot \bar{B}}$ |  | | | | | | | | | | | | | | | |
| NOR |  | $F = \overline{A + B} = \overline{\overline{\bar{A} \cdot \bar{B}}} = \bar{A} \cdot \bar{B}$ |  | | | | | | | | | | | | | | | |
| XOR |  OR EXCLUSIVA | |  $F = A \oplus B = \bar{A} \cdot B + A \cdot \bar{B} = \overline{\overline{\bar{A} \cdot B + A \cdot \bar{B}}} = \overline{(\bar{A} \cdot B) \cdot (A \cdot \bar{B})}$ | | | | | | | | | | | | | | | |
| NXOR |  NOR EXCLUSIVA | |  $F = \overline{A \oplus B} = \overline{\bar{A} \cdot B + A \cdot \bar{B}} = \overline{\overline{\bar{A} \cdot B + A \cdot \bar{B}}} = \overline{(\bar{A} \cdot B) \cdot (A \cdot \bar{B})}$ | | | | | | | | | | | | | | | |

Figure 5: Obtención de las puertas lógicas mediante NAND.