**CIRCUITOS COMBINACIONALES**

**Implementación de funciones con compuertas N-OR (NOR), N-AND (NAND).**

Los operadores Booleanos +, - (OR, AND y NOT), pueden ser reemplazados por un dispositivo eléctrico, mecánico, etc.

Para construir un circuito que represente una función booleana, de la cual se debe haber obtenido previamente su mínima expresión, se reemplazan los operadores booleanos por su circuito análogo.

Por ejemplo:





Si quisiéramos implementarlos como “Producto de Sumas”. Al aplicar las Leyes de Morgan obtenemos.

|  |  |
| --- | --- |
|  |  |

Aplicando los postulados podemos convertir el producto de productos en una suma de productos.

Para poder realizar esto, como se muestra en la figura, es necesaria la siguiente conversión a partir de la función dada.

|  |  |
| --- | --- |
| 🡪 |  |

Con la misma función inicial del ejemplo, pero expresado como producción de sumas y transformándola para ser construida exclusivamente con compuertas NAND tenemos.

Su circuito equivalente será:

Está representación es más compleja que las anteriores y se puede desarrollar combinando otro tipo de compuertas.



Por ejemplo con las compuertas NOR y NAND, como se muestra en la siguiente figura.

**Función O-exclusiva.**

La función O-exclusiva de dos variables a y b es aquella que toma el valor uno cuando una de las variables toma el valor uno y la otra el cero.

La función O-exclusiva se representa mediante el símbolo 

El símbolo del circuito, la función y la tabla de la verdad serán:



Donde

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Tabla de la Verdad   |  |  |  | | --- | --- | --- | | a | b | F | | 0 | 0 | 0 | | 0 | 1 | 1 | | 1 | 0 | 1 | | 1 | 1 | 0 | |

**Circuitos Combinacionales**

A la representación de funciones Booleanas por medio de compuertas lógicas, se las denomina “Circuitos Computacionales” debido a que la salida es una combinación de las entradas.

Es importante representar la función con el menor número de compuertas, pero se debe tener en cuenta la cantidad de niveles recorridos por cada señal, debe ser similar para evitar retrasos o desencuentro de las señales.

En la figura que se muestra a continuación se da un ejemplo.



La señal “C” tiene dos niveles, mientras que la señal “A” tiene cinco.

Esto origina un retraso en la señal de llegada “C” con respecto a la señal “A” en la compuerta “C1” y por lo tanto dará un error.

**Procedimiento para el diseño lógico.**

Se plantea la tabla de verdad del problema propuesto.

De la tabla de verdad se obtiene la función.

Se minimiza la función.

Para la minimización de la función se procede agrupar de 2n términos. Si tuviéramos

Trataríamos de agrupar de a 8.

Trataríamos de agrupar de a 4

Trataríamos de agrupar de 2.

Agrupamos de a 1.

Se traza el diagrama (circuito combinacional) reemplazando los operadores Booleanos por compuertas.

Con este procedimiento se logra transferir el problema planteado a la resolución de problemas a través de circuitos lógicos.

Sin embargo, la minimización de una función o de un problema planteado de este tipo, es útil, sin llegar a plantear el circuito lógico, ya que puede utilizarse esta minimización para la realización de algoritmos, dentro de rutinas de programación.