

PRÁCTICO 2 - Lógica Combinacional

TABLA 2-3 Mintérminos y maxtérminos para tres variables binarias.

| Mintérminos | | | | | Maxtérminos | |
|-------------|---|---|----------|-------------|----------------|-------------|
| x | y | z | Término | Designación | Término | Designación |
| 0 | 0 | 0 | $x'y'z'$ | m_0 | $x + y + z$ | M_0 |
| 0 | 0 | 1 | $x'y'z$ | m_1 | $x + y + z'$ | M_1 |
| 0 | 1 | 0 | $x'yz'$ | m_2 | $x + y' + z$ | M_2 |
| 0 | 1 | 1 | $x'yz$ | m_3 | $x + y' + z'$ | M_3 |
| 1 | 0 | 0 | $xy'z'$ | m_4 | $x' + y + z$ | M_4 |
| 1 | 0 | 1 | $xy'z$ | m_5 | $x' + y + z'$ | M_5 |
| 1 | 1 | 0 | xyz' | m_6 | $x' + y' + z$ | M_6 |
| 1 | 1 | 1 | xyz | m_7 | $x' + y' + z'$ | M_7 |

Mapas de Karnaugh:

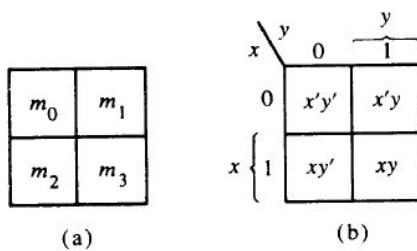


Figura 3-1 Mapa de dos variables.

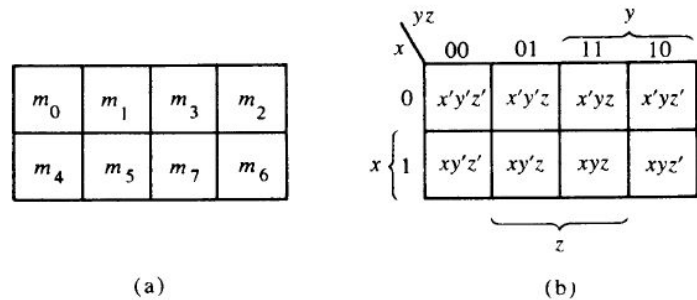


Figura 3-3 Mapa de tres variables.

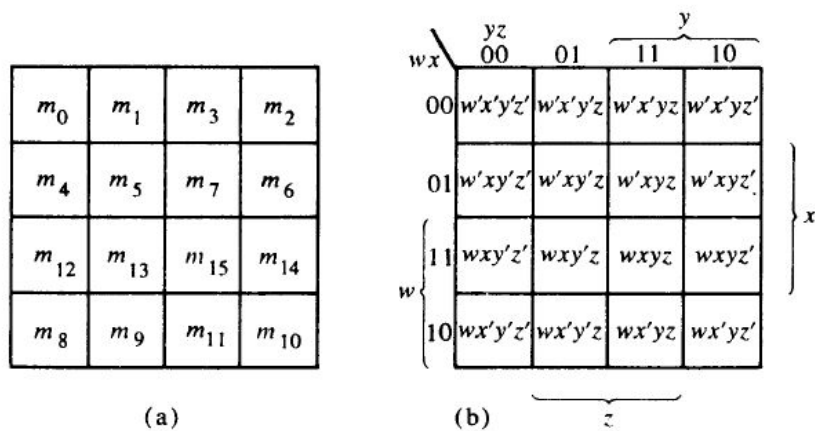
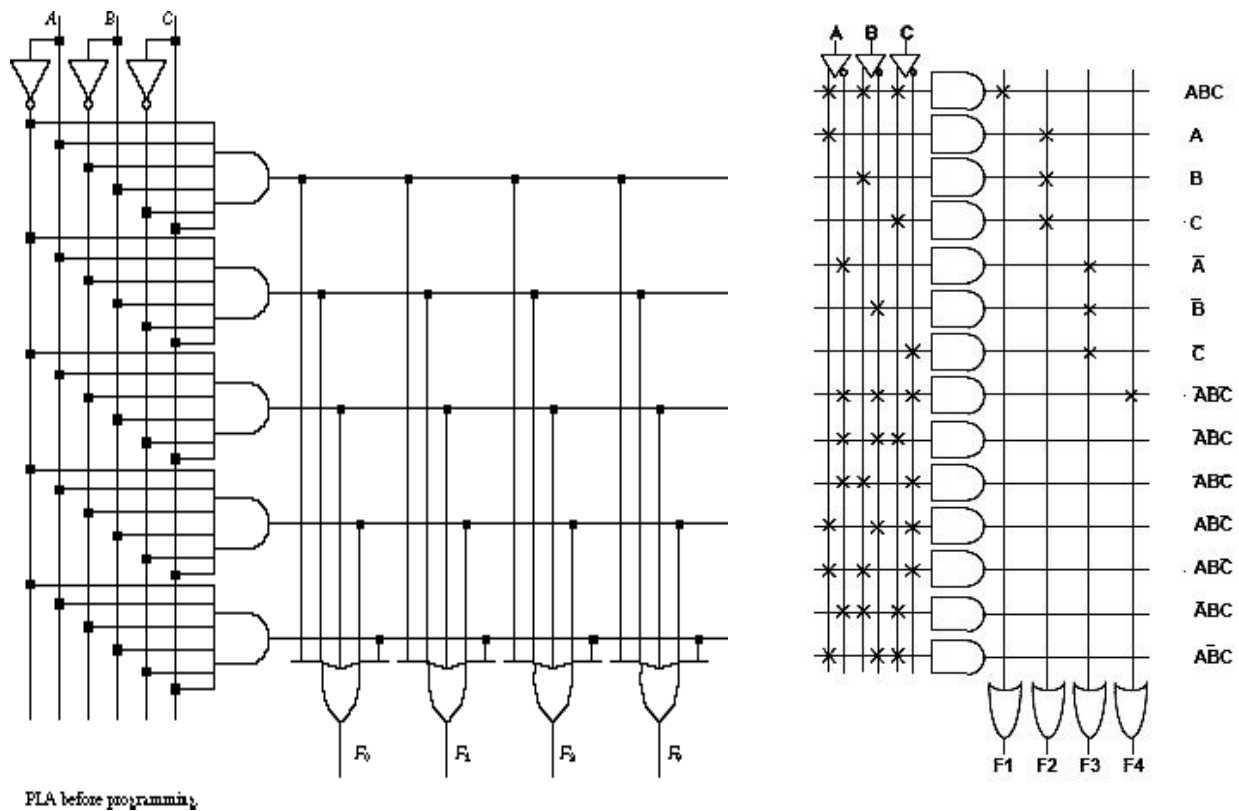


Figura 3-8 Mapa de cuatro variables.

Programmable Logic Array (PLA):**Ejercicio 1:**

Un detector de paridad impar de 4 entradas y una salida funciona de la siguiente manera: Si la cantidad de entradas con valor '1' es impar la salida se pone en '1', en el resto de los casos la salida es falsa y toma valor '0'.

- Construir la tabla de verdad para dicho sistema.
- Obtener la ecuación lógica como suma de minitérminos y producto de maxitérminos.
- Implementar el sistema con compuertas NAND.
- Implementar el sistema con una PLA.

Ejercicio 2:

Un sistema digital recibe información en forma de palabras de 5 bits (**ABCDE**) en un código protegido contra errores, de tal forma que cualquier dato que se reciba debe contener 3 y sólo 3 bits en '1'. Diseñar un circuito con dichas entradas y una salida **err** que se activa por bajo cuando se recibe un dato incorrecto.

- Construir la tabla de verdad para dicho sistema.
- Obtener la ecuación lógica como suma de minitérminos y producto de maxitérminos.
- Implementar el sistema con compuertas lógicas básicas.
- Implementar el sistema con una PLA.

Ejercicio 3:

Dada la siguiente tabla de verdad para la función F:

| x3 | x2 | x1 | x0 | F(x3,x2,x1,x0) |
|----|----|----|----|----------------|
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 |

- Encontrar la expresión de $F(x_3, x_2, x_1, x_0)$ como suma de minitérminos y como productos de maxitérminos.
- Encontrar la expresión minimizada de $F(x_3, x_2, x_1, x_0)$ utilizando mapas de Karnaugh.

Ejercicio 4:

Dada la siguiente tabla de verdad para la función F:

| x2 | x1 | x0 | F(x2,x1,x0) |
|----|----|----|-------------|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

- Encontrar la expresión de $F(x_2, x_1, x_0)$ como suma de minitérminos y como productos de maxitérminos.
- Encontrar la expresión minimizada de $F(x_2, x_1, x_0)$ utilizando mapas de Karnaugh.

Ejercicio 5:

Dada la siguiente tabla de verdad para la función F:

| x3 | x2 | x1 | x0 | F(x3,x2,x1,x0) |
|----|----|----|----|----------------|
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 |

- Encontrar la expresión de $F(x_3, x_2, x_1, x_0)$ como suma de minterminos y como productos de maxiterminos.
- Encontrar la expresión minimizada de $F(x_3, x_2, x_1, x_0)$ utilizando mapas de Karnaugh.

Ejercicio 6:

Construir un circuito combinacional comparador que tome dos números de 2 bits, $\mathbf{A} = (a_1, a_0)$ y $\mathbf{B} = (b_1, b_0)$ y retorne tres salidas (" $\mathbf{a} > \mathbf{b}$ ", " $\mathbf{a} = \mathbf{b}$ " y " $\mathbf{a} < \mathbf{b}$ ") de 1 bit cada una.

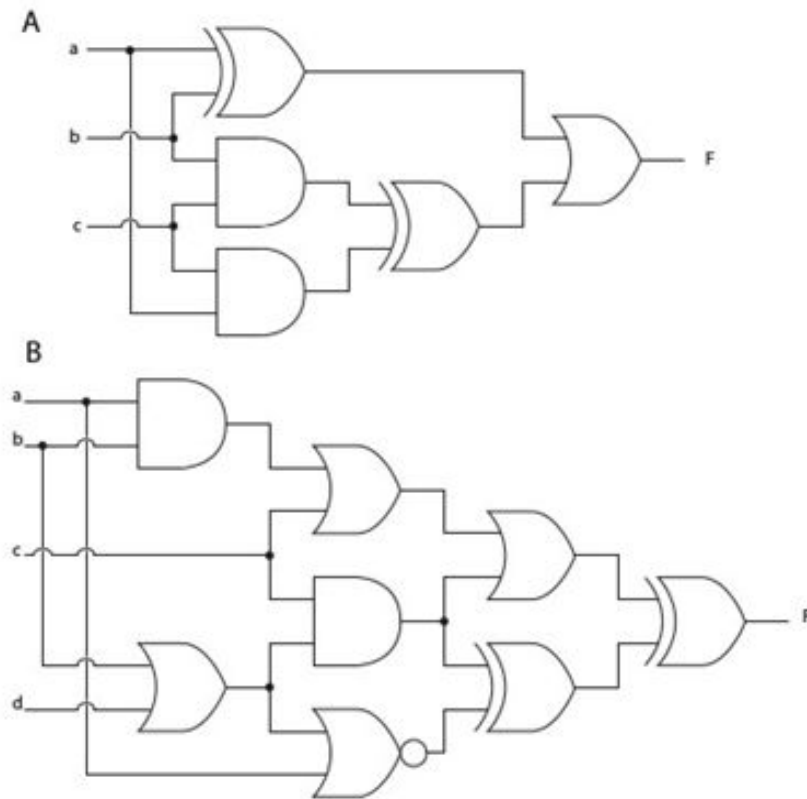
Ej: $\mathbf{A} = (00)$ y $\mathbf{B} = (10)$, entonces " $\mathbf{a} > \mathbf{b}$ " = '0', " $\mathbf{a} = \mathbf{b}$ " = '0' y " $\mathbf{a} < \mathbf{b}$ " = '1'.

- Construir la tabla de verdad para dicho sistema.
- Obtener la ecuación lógica como suma de minterminos y producto de maxiterminos.
- Encontrar la función minimizada de cada salida como suma de productos usando mapas de Karnaugh.
- Implementar el sistema con compuertas lógicas básicas.

Ejercicio 7:

Analizar los circuitos de lógica combinacional de la figura. Para cada uno:

- Escribir la función booleana correspondiente.
- Encontrar la tabla de verdad para dicha función.
- Escribir el mapa de Karnaugh utilizando esa tabla.
- Obtener la función minimizada como suma de productos a partir del mapa.
- Dibujar el circuito de lógica combinacional resultante.

**Ejercicio 8:**

Un DECODIFICADOR es un circuito combinacional que convierte información binaria de 'n' entradas codificadas (**A**), a un máximo de '2n' salidas únicas (**X**). Esto quiere decir que sólo una salida está activa y representa el valor de las señales de entrada.

Considere un Decodificador activo por bajo (salida activa = '0') con $n=2$ y $2n=4$ (deco 2 x 4).

- Expresar las tablas de verdad de las cuatro salidas X_0 , X_1 , X_2 y X_3 .
- Encontrar las expresiones de X_0 , X_1 , X_2 y X_3 como suma de minitérminos y como producto de maxitérminos.
- Encontrar expresiones minimizadas de X_0 , X_1 , X_2 y X_3 utilizando el método de Karnaugh o un método algebraico.
- Implementar las expresiones anteriores a través del uso de compuertas lógicas.
- Repetir el punto "d" agregando una entrada de HABILITACIÓN (**E**) activa por bajo, de tal forma que cuando **E**=1 ninguna señal de salida permanezca habilitada.

Ejercicio 9:

- Diseñe un circuito SUMADOR COMPLETO (3 entradas: **Cin**, **X**, **Y**; 2 salidas: **S**, **Cout**) mediante el uso de un Decodificador y compuertas OR. Tip: comenzar expresando las salidas **S** y **Cout** como suma de minitérminos.
- Diseñe un sumador completo usando dos semisumadores y una compuerta.

Ejercicio 10:

Implementar un decodificador 3 x 8 y otro de 4 x 16 a partir de decodificadores 2 x 4 con entrada de habilitación (**E**) activa por bajo y compuertas lógicas.

Ejercicio 11:

Considerando que un CODIFICADOR es un circuito combinacional que realiza la operación inversa de un decodificador:

- a. Expresar la tabla de verdad de un codificador de octal a binario (8 entradas x 3 salidas).
- b. Mostrar su implementación con compuertas OR a partir de la expresión del Codificador como suma de minitérminos.

Ejercicio 12:

Un MULTIPLEXOR (MUX) es un circuito combinacional que selecciona información binaria de muchas entradas y la dirige a una única salida, conforme al estado de las señales de selección. De esta forma si un MUX posee 'n' señales de selección (**S**), tendrá '2n' entradas de información (**D**).

- a. Expresar la tabla de verdad de un MUX de 2 entradas (y una salida) y su implementación mediante el uso de compuertas lógicas (AND, OR, NOT, NOR, NAND, etc.)
- b. Mostrar cómo se puede usar un MUX para obtener una compuerta NOT.
- c. ¿Cómo obtener un MUX de 4 entradas (y una salida) en base a multiplexores de 2 entradas?
- d. ¿Cómo obtener un multiplexor de 'n' entradas con multiplexores de 2 entradas?