



Electrónica



Tema 4 Álgebra de Boole

Circuitos Digitales

- Los circuitos digitales son aquellos en los que sus señales sólo pueden tomar el valor '0' y '1'.
- Los circuitos digitales se dividen en:
 - Combinacionales: Las salidas sólo dependen de las entradas (no tienen memoria)
 - Secuenciales: Las salidas dependen de las entradas actuales y las pasadas (tienen memoria)

Algebra de Boole

- Se usan variable simbólicas, como A,B para representar las señales digitales.
- Convenio:
 - Lógica positiva : 0 LOW, 1 HIGH.
 - Lógica negativa: 1 LOW, 0 HIGH.

LOW, HIGH normalmente se refiere al voltaje de la señal.

Axiomas

- Los siguientes axiomas definen el sistema de la lógica booleana:

Axioma 1 : Abstracción Digital

$$X = 0 \text{ si } X \neq 1$$

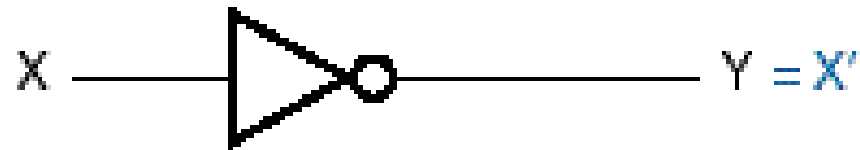
$$X = 1 \text{ si } X \neq 0$$

Axiomas

Axioma 2 : Operador complemento

Si $X = 0$, entonces $X' = 1$

Si $X = 1$, entonces $X' = 0$



Copyright © 2000 by Prentice Hall, Inc.
Digital Design Principles and Practices, 3/e

Axiomas

Axioma 3-5 : Operadores AND,OR

$$(A3) 0 \cdot 0 = 0$$

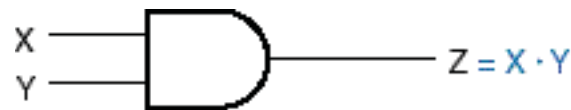
$$(A3') 1 + 1 = 1$$

$$(A4) 1 \cdot 1 = 1$$

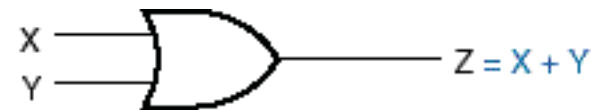
$$(A4') 0 + 0 = 0$$

$$(A5) 0 \cdot 1 = 1 \cdot 0 = 0$$

$$(A5') 1 + 0 = 0 + 1 = 1$$



(a)



(b)

Axiomas

Una Variable

Identidad: (T1) $X+0 = X$; (T1') $X \cdot 1 = X$

Elemento nulo: (T2) $X+1 = 1$; (T2') $X \cdot 0 = 0$

Idempotencia: (T3) $X+X = X$; (T3') $X \cdot X = X$

Involución: (T4) $(X')' = X$

Complemento: (T5) $X+X' = 1$; (T5') $X \cdot X' = 0$

Axiomas

Dos y Tres Variables

Table 4-2 Switching-algebra theorems with two or three variables.

(T6)	$X + Y = Y + X$	(T6')	$X \cdot Y = Y \cdot X$	(Commutativity)
(T7)	$(X + Y) + Z = X + (Y + Z)$	(T7')	$(X \cdot Y) \cdot Z = X \cdot (Y \cdot Z)$	(Associativity)
(T8)	$X \cdot Y + X \cdot Z = X \cdot (Y + Z)$	(T8')	$(X + Y) \cdot (X + Z) = X + Y \cdot Z$	(Distributivity)
(T9)	$X + X \cdot Y = X$	(T9')	$X \cdot (X + Y) = X$	(Covering)
(T10)	$X \cdot Y + X \cdot Y' = X$	(T10')	$(X + Y) \cdot (X + Y') = X$	(Combining)
(T11)	$X \cdot Y + X' \cdot Z + Y \cdot Z = X \cdot Y + X' \cdot Z$			(Consensus)
(T11')	$(X + Y) \cdot (X' + Z) \cdot (Y + Z) = (X + Y) \cdot (X' + Z)$			

Axiomas

n Variables

Idempotencia generalizada :

$$(T12) X+X+\dots+X = X$$

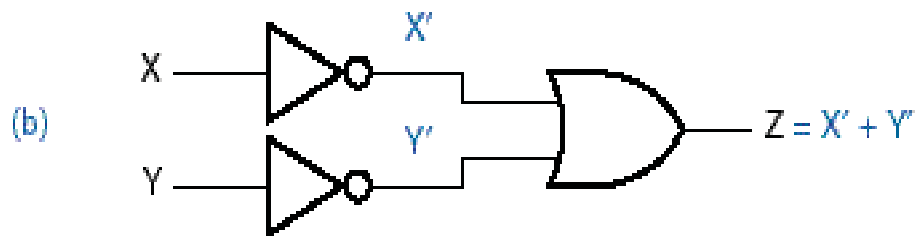
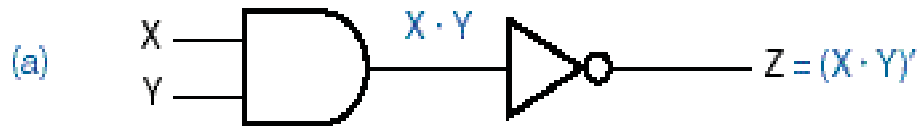
$$(T12') X \cdot X \cdot \dots \cdot X = X$$

Teorema de DeMorgan :

$$(T13) (X_1 \cdot X_2 \cdot \dots \cdot X_N)' = X_1' + X_2' + \dots + X_N'$$

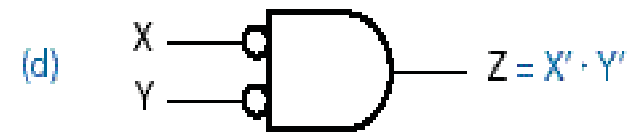
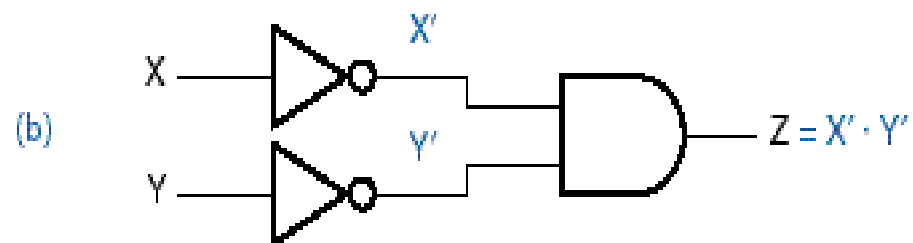
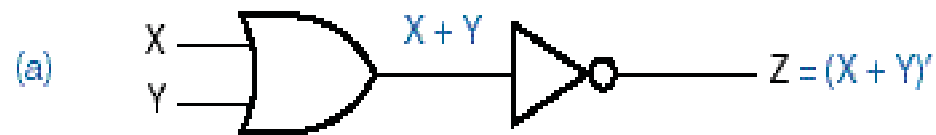
$$(T13') (X_1 + X_2 + \dots + X_N)' = X_1' \cdot X_2' \cdot \dots \cdot X_N'$$

Teorema de DeMorgan (T13)



Copyright © 2000 by Prentice Hall, Inc.
Digital Design Principles and Practices, 3/e

Teorema de DeMorgan(T13')



Representación de funciones lógicas

Tabla de verdad

Table 4-4
General truth table structure for a 3-variable logic function, $F(X, Y, Z)$.

<i>Row</i>	<i>X</i>	<i>Y</i>	<i>Z</i>	<i>F</i>
0	0	0	0	$F(0, 0, 0)$
1	0	0	1	$F(0, 0, 1)$
2	0	1	0	$F(0, 1, 0)$
3	0	1	1	$F(0, 1, 1)$
4	1	0	0	$F(1, 0, 0)$
5	1	0	1	$F(1, 0, 1)$
6	1	1	0	$F(1, 1, 0)$
7	1	1	1	$F(1, 1, 1)$

Representación de funciones lógicas

Definiciones

Literal : Variable o su complemento.

Término producto: Un literal o un producto de literales.

Término suma: Un literal o una suma de literales.

Suma de Productos : Una suma de términos producto.

Producto de Sumas : Un producto de términos suma.

Representación de Funciones lógicas

Definiciones

Término normalizado: Un término suma o producto, en el cual cada variable no aparece más de una vez.

Un término no normalizado siempre puede simplificarse a término normalizado o a una constante.

$$X+X, X+X', X \cdot X, X \cdot X'$$

Representación de Funciones lógicas

Definiciones

Mintérmino de n variables: Un término producto normalizado con n literales.

Maxtérmino de n variables : Un término suma normalizado con n literales.

$$X+Y'+Z, X \cdot Y' \cdot Z'$$

Representación de Funciones lógicas

Tabla de verdad

Table 4-6
Minterms and maxterms for a 3-variable logic function, $F(X, Y, Z)$.

<i>Row</i>	<i>X</i>	<i>Y</i>	<i>Z</i>	<i>F</i>	<i>Minterm</i>	<i>Maxterm</i>
0	0	0	0	$F(0,0,0)$	$X' \cdot Y' \cdot Z'$	$X + Y + Z$
1	0	0	1	$F(0,0,1)$	$X' \cdot Y' \cdot Z$	$X + Y + Z'$
2	0	1	0	$F(0,1,0)$	$X' \cdot Y \cdot Z'$	$X + Y' + Z$
3	0	1	1	$F(0,1,1)$	$X' \cdot Y \cdot Z$	$X + Y' + Z'$
4	1	0	0	$F(1,0,0)$	$X \cdot Y' \cdot Z'$	$X' + Y + Z$
5	1	0	1	$F(1,0,1)$	$X \cdot Y' \cdot Z$	$X' + Y + Z'$
6	1	1	0	$F(1,1,0)$	$X \cdot Y \cdot Z'$	$X' + Y' + Z$
7	1	1	1	$F(1,1,1)$	$X \cdot Y \cdot Z$	$X' + Y' + Z'$

Representación de Funciones lógicas

Definiciones

Mintérmino i : El mintérmino para la fila i de la tabla de verdad.

Maxtérmino i : El maxtérmino para la fila i de la tabla de verdad.

Representación de Funciones lógicas

Formas canónicas

Suma canónica: La suma de los mintérminos de la tabla de verdad.

$$\Sigma_{A,B,C}(1,2,3)$$

Producto canónico: El producto de los maxtérminos de la tabla de verdad.

$$\Pi_{A,B,C}(0,4,5,6,7)$$

Análisis de circuitos combinacionales

Dado un circuito, obtener una descripción formal de su función lógica.

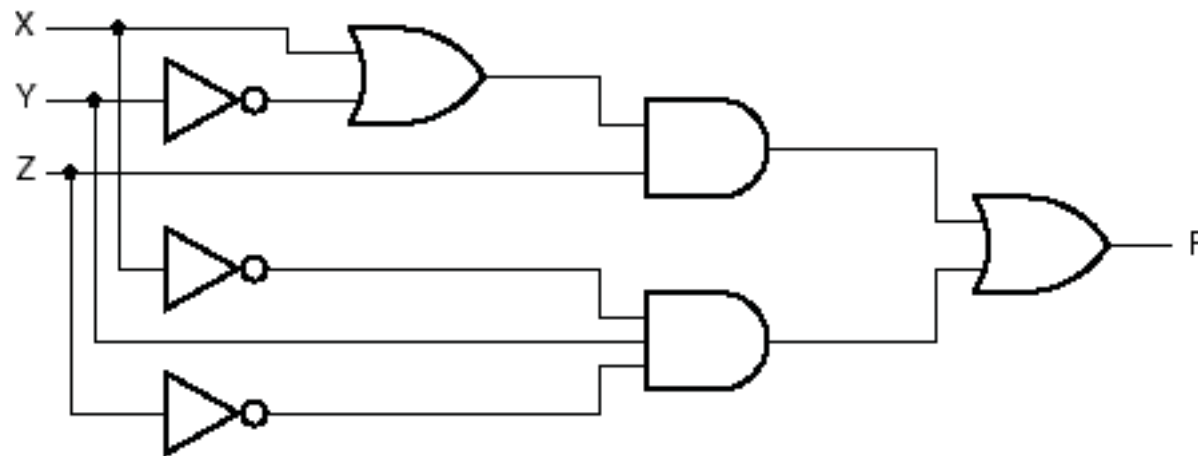
Posibilita:

- Determinar el comportamiento de varias combinaciones de entradas.
- Manipular la descripción algebraica para:
 - Sugerir estructuras de circuitos alternativas.
 - Mapearlo a un dispositivo en especial (CPLDs).
- Modelar el circuito para su uso en un sistema más grande

Análisis de circuitos combinacionales

Construir la tabla de verdad

- Aplicar todas las posibles combinaciones de entrada y computar las salidas aplicando los axiomas del álgebra de conmutación.



Análisis de circuitos combinacionales

Tabla de verdad

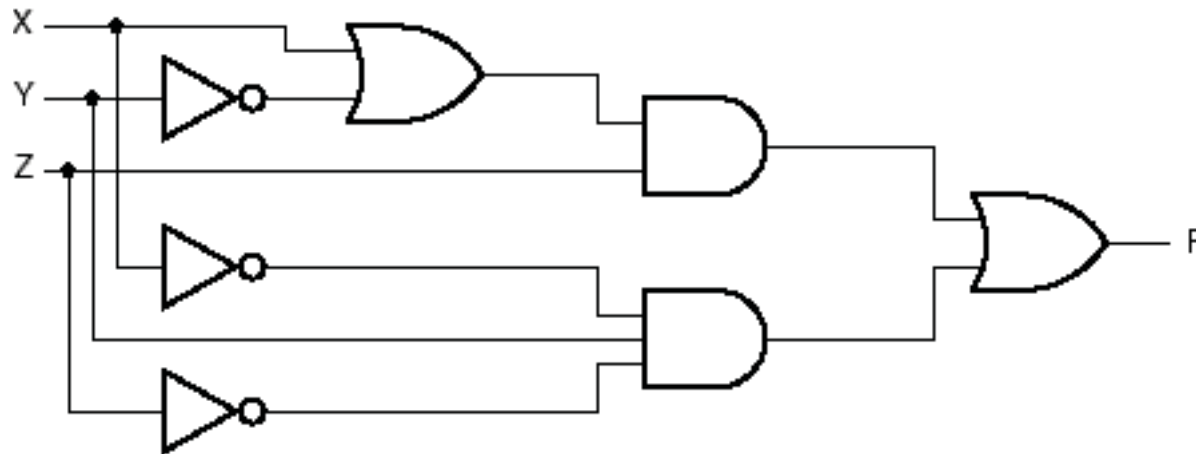
Row	X	Y	Z	F
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	0
5	1	0	1	1
6	1	1	0	0
7	1	1	1	1

Table 4-7
Truth table for the
logic circuit of
Figure 4-9.

Análisis de circuitos combinacionales

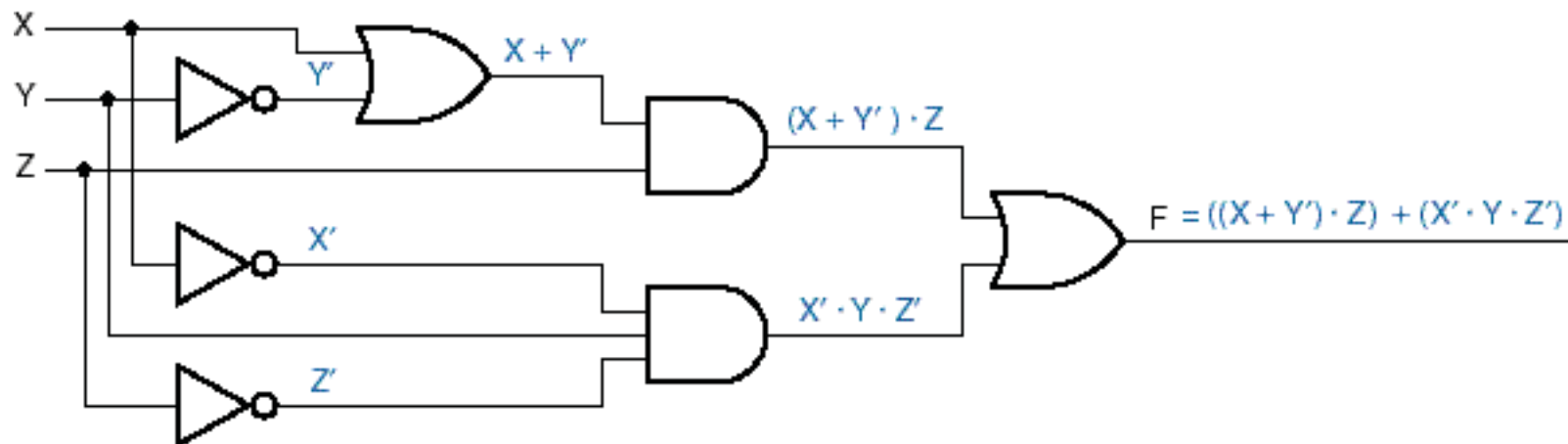
Aproximación algebraica

- Empezar en las entradas, construyendo las ecuaciones para los nodos intermedios, hasta llegar a las salidas.



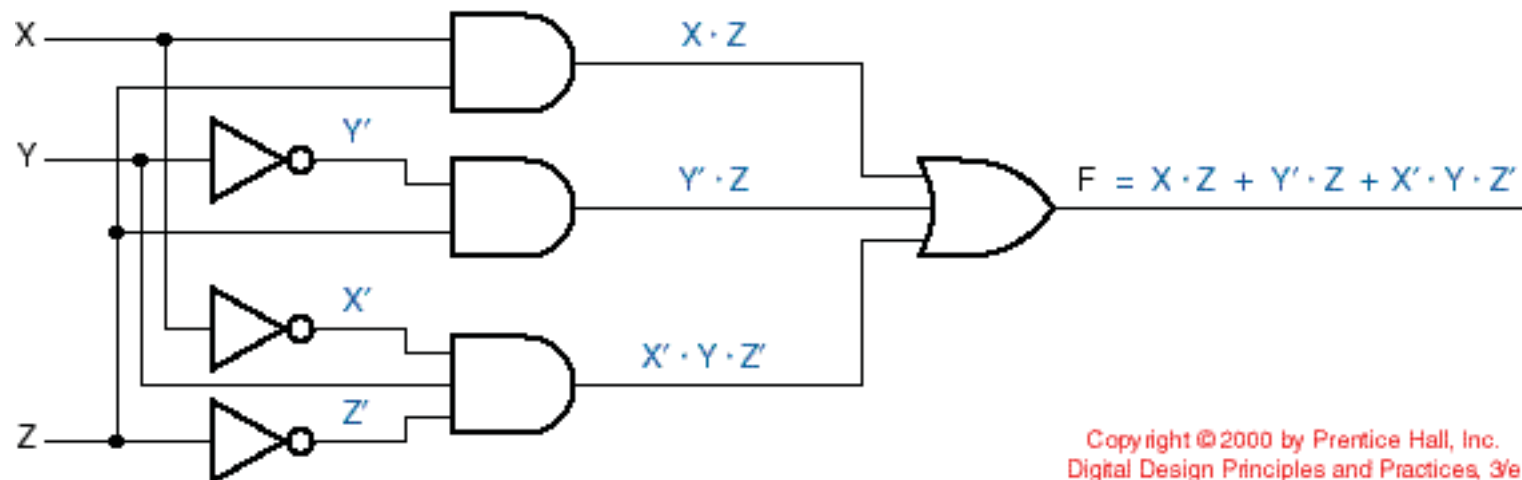
Análisis de circuitos combinacionales

Ecuación algebraica



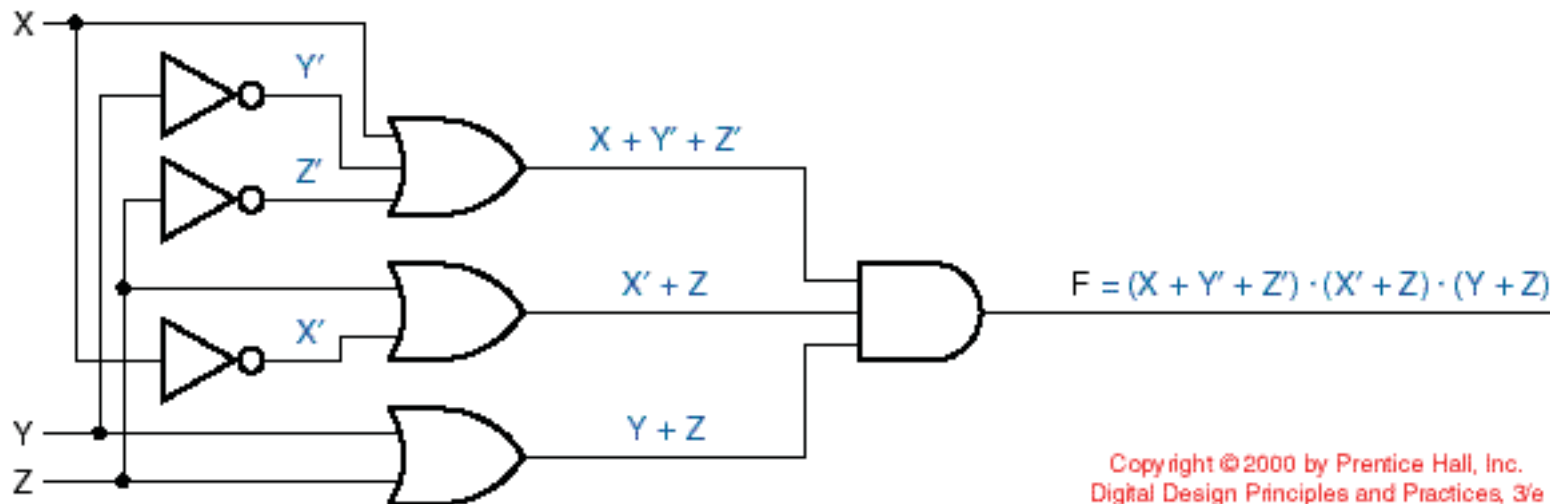
Análisis de circuitos combinacionales

Suma canónica (AND-OR)



Análisis de circuitos combinacionales

Producto canónico (OR-AND)



Síntesis de circuitos combinacionales

Dada una descripción del diseño, sintetizar un
circuito para implementarlo.

Descripción

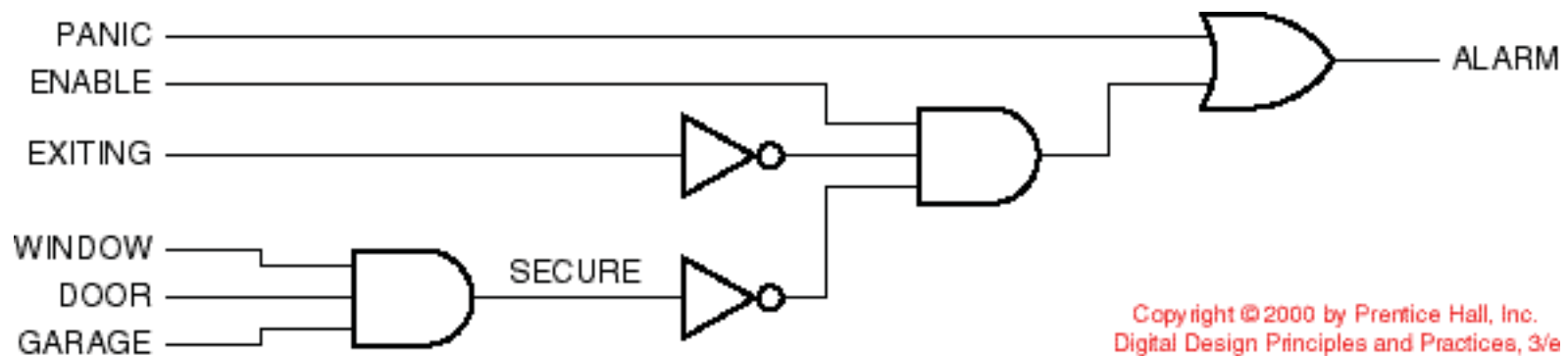
- Texto.
- Tabla de verdad.

Ejemplo : 'Alarm es 1 si Panic es 1 ó (Enable es
1 y Exiting es 0 y Secure es 0)'

Secure es 1 si Window, Door y Garage son 1.

Síntesis de circuitos combinacionales

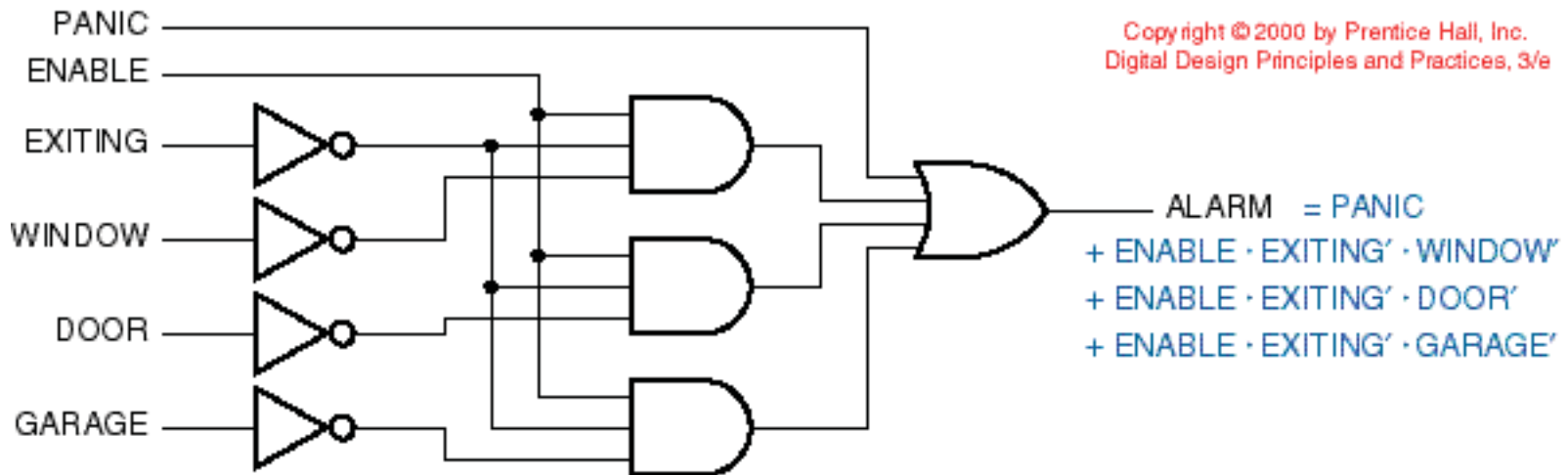
Solución para el Ejemplo



Síntesis de circuitos combinacionales

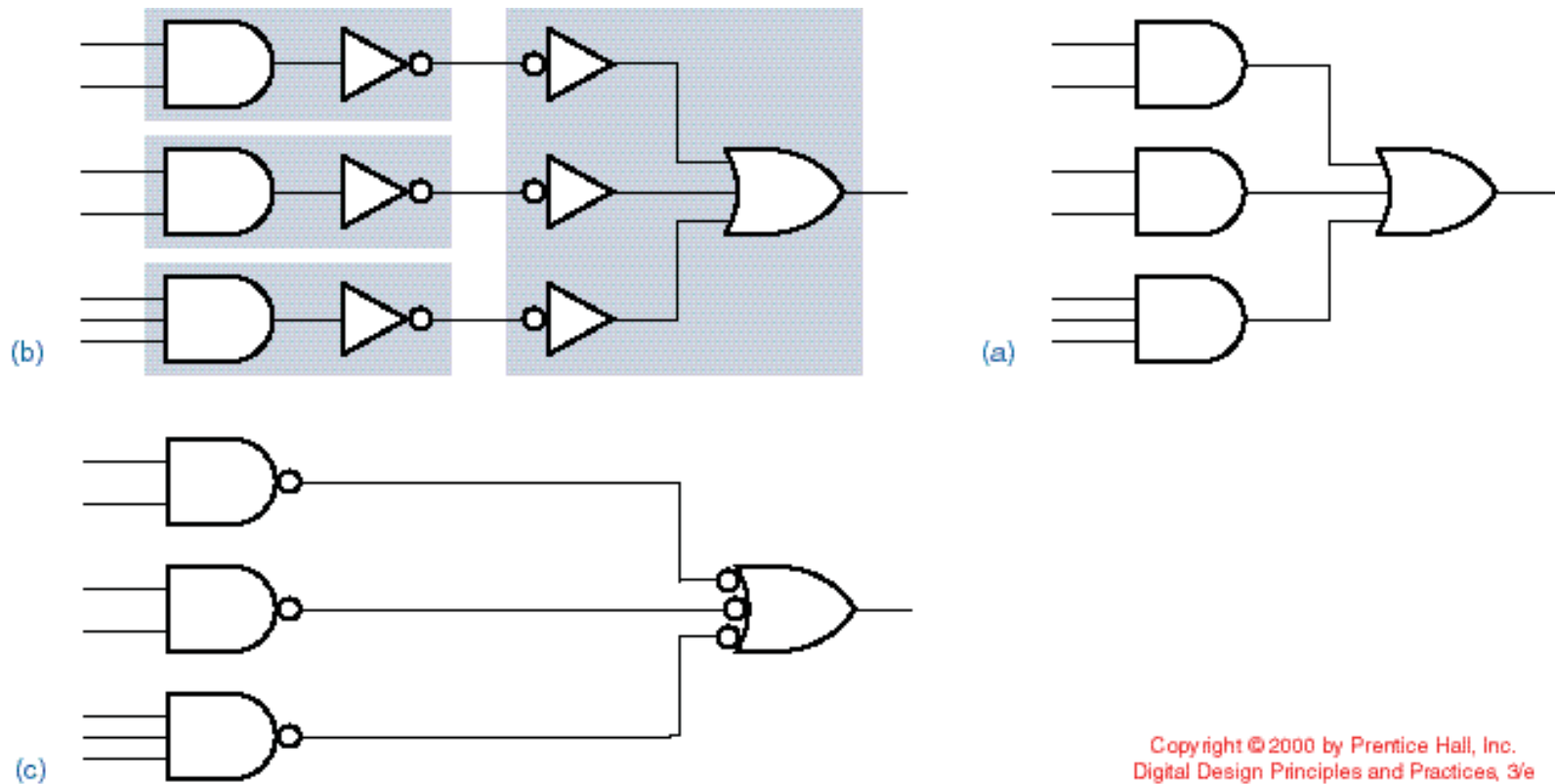
Solución en Suma de Productos para el Ejemplo

$$\text{ALARM} = \text{PANIC} + \text{ENABLE} \cdot \text{EXITING}' \cdot (\text{WINDOW} \cdot \text{DOOR} \cdot \text{GARAGE})'$$



Manipulación de Circuitos

AND-OR \Rightarrow NAND-NAND (CMOS)

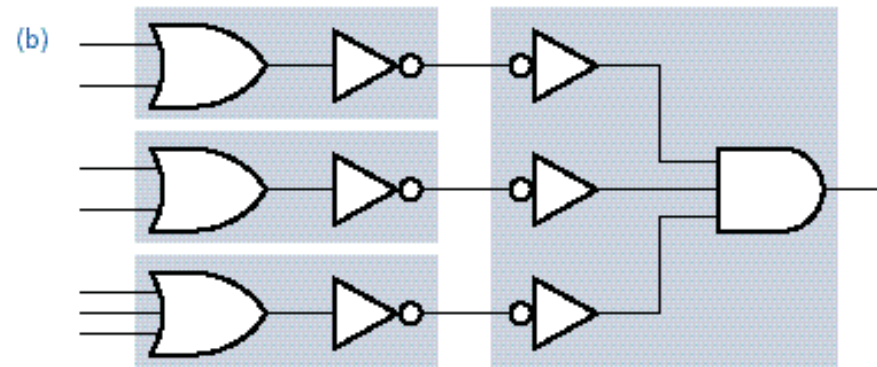
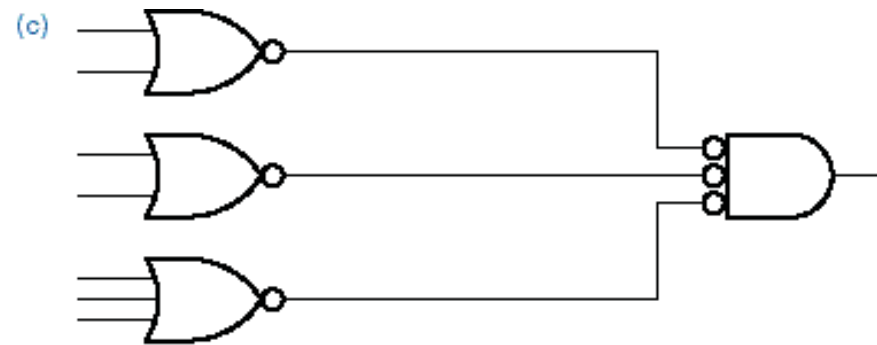
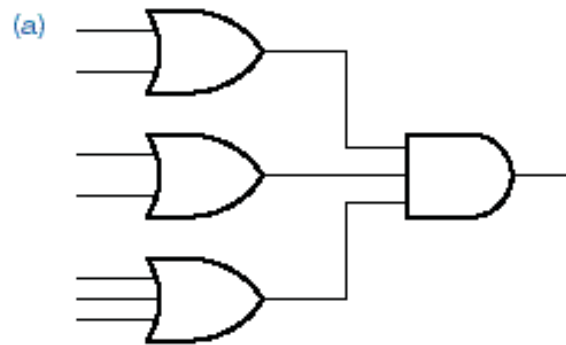


Copyright © 2000 by Prentice Hall, Inc.
Digital Design Principles and Practices, 3/e

Manipulación de Circuitos

OR- AND \Rightarrow NOR-NOR (CMOS)

Copyright © 2000 by Prentice Hall, Inc.
Digital Design Principles and Practices, 3/e



Minimización de Circuitos

Reducir el número de puertas y su tamaño para un circuito dado.

- El punto de partida normalmente es la tabla de verdad, la lista de mintérminos o la de maxtérminos.
- Se puede disminuir:
 - Número de puertas del primer nivel.
 - Número de entradas a puertas del primer nivel.
 - Número de entradas a puertas del segundo nivel.

Minimización de Circuitos

Basado principalmente en la combinación de los Teoremas (T10) y (T10)' :

$$\text{Término_Producto} \cdot Y + \text{Término_Producto} \cdot Y' = \text{Término_Producto}$$

$$(\text{Término_Suma} + Y) \cdot (\text{Término_Suma} + Y') = \text{Término_Suma}$$

Mapas de Karnaugh

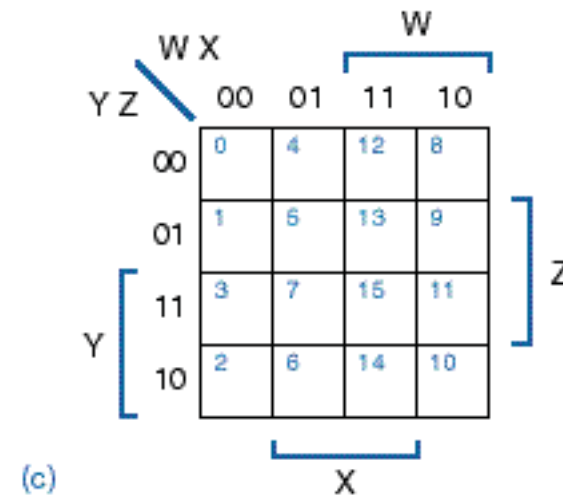
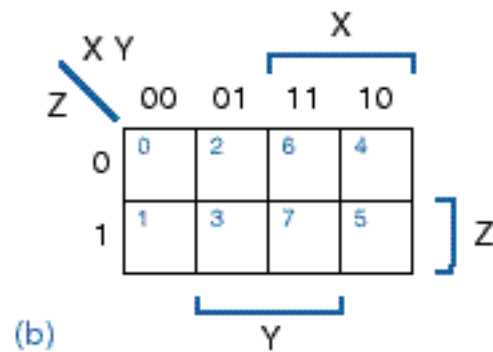
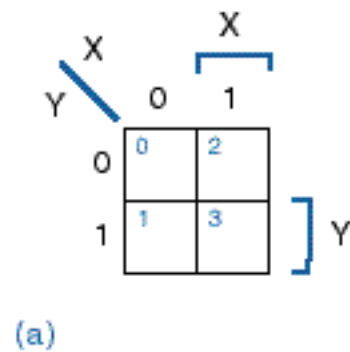
Representación gráfica de una función lógica usada para minimizar circuitos.

- Útil para funciones pequeñas de 2,3,4 variables.
- Se ordenan los valores en filas y columnas de tal manera que las celdas adyacentes difieran en una única variable.

Mapas de Karnaugh

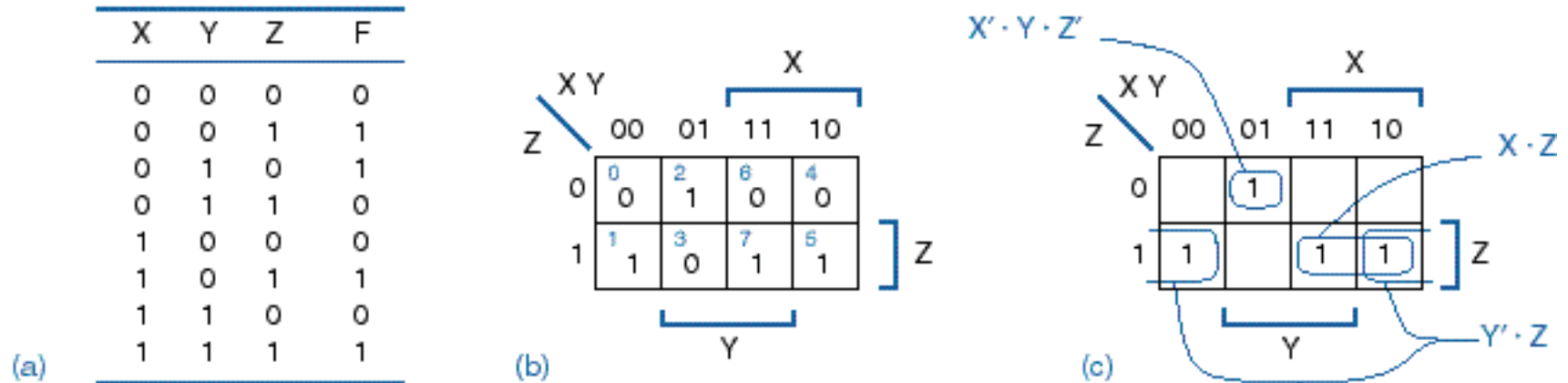
Ejemplos

Copyright © 2000 by Prentice Hall, Inc.
Digital Design Principles and Practices, 3/e



Mapas de Karnaugh

Ejemplo



Simplificación: Término $\cdot Y$ + Término $\cdot Y'$ =
Término

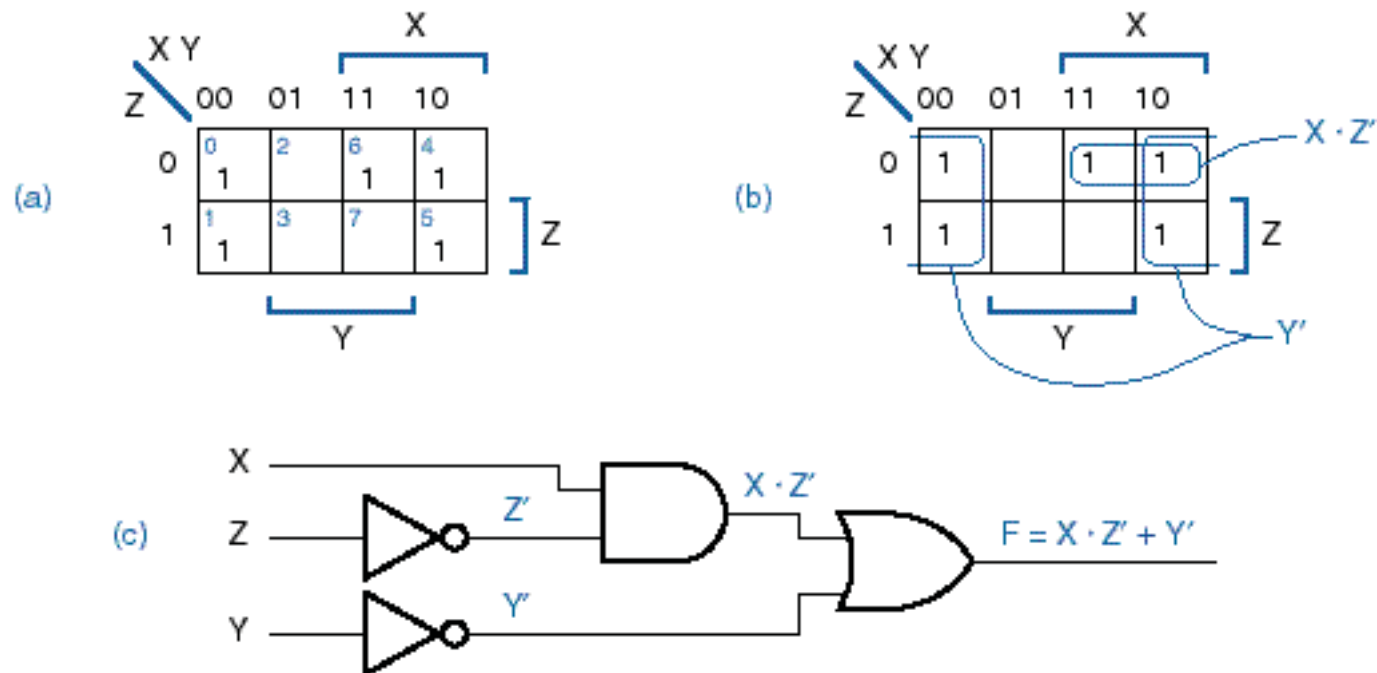
Mapas de Karnaugh

Combinación de celdas : Un conjunto de 2^i celdas puede ser combinado si hay i variables de la función lógica que toman todas las 2^i posibles combinaciones, y el resto de las $n-i$ variables tienen el mismo valor en esas celdas.

Gráficamente: Rectángulos que contienen todo '1'.

Mapas de Karnaugh

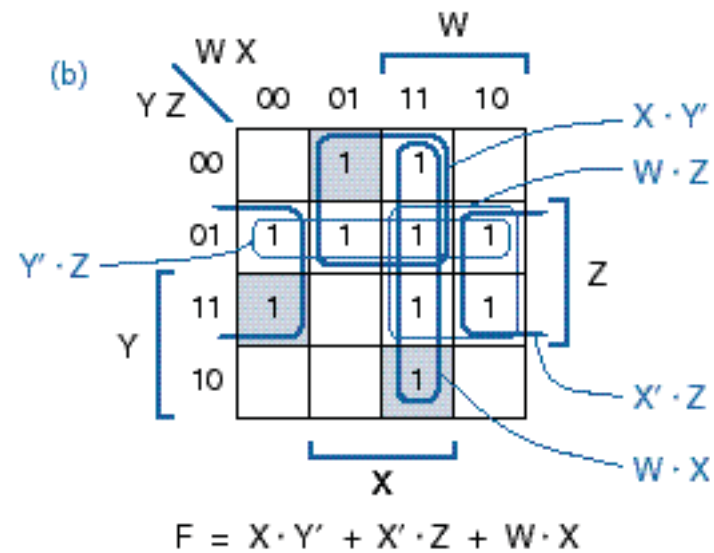
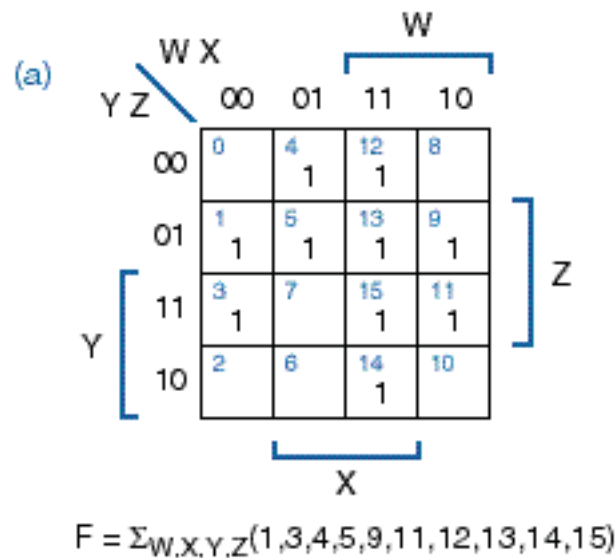
Ejemplos



Mapas de Karnaugh

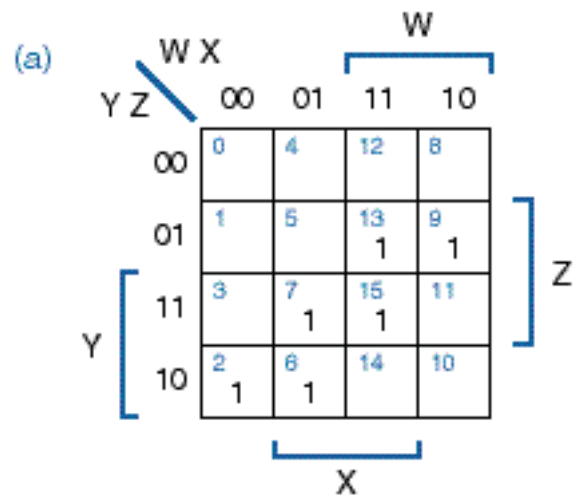
Ejemplo

Copyright © 2000 by Prentice Hall, Inc.
Digital Design Principles and Practices, 3/e

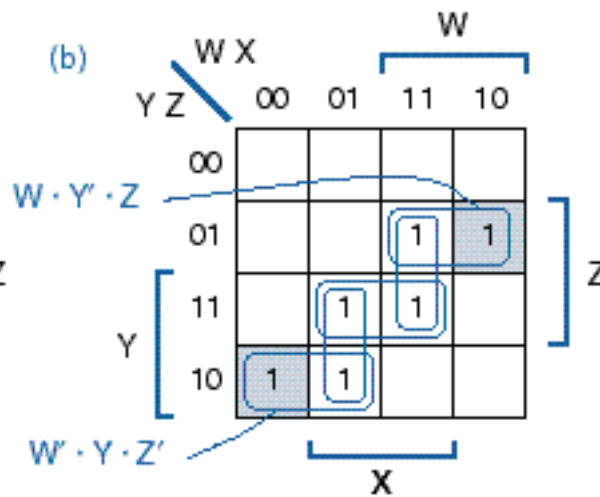


Mapas de Karnaugh

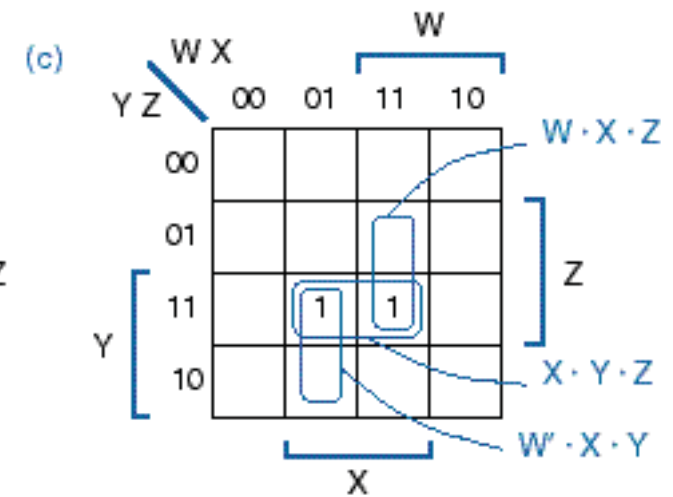
Ejemplos



$$F = \sum_{W,X,Y,Z}(2,6,7,9,13,15)$$

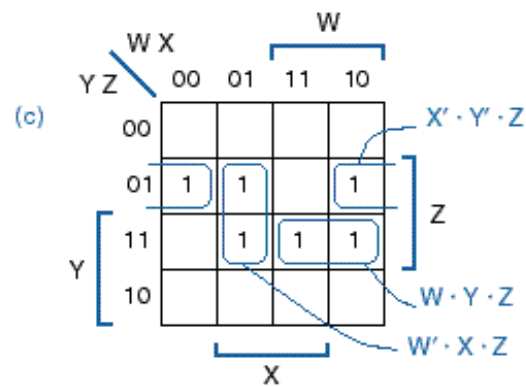
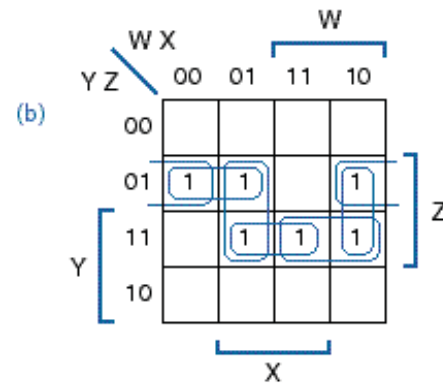
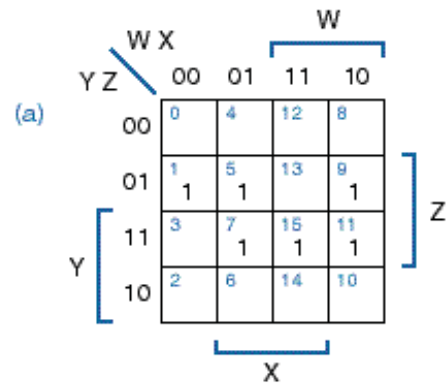


$$F = W \cdot Y' \cdot Z + W' \cdot Y \cdot Z' + X \cdot Y \cdot Z$$

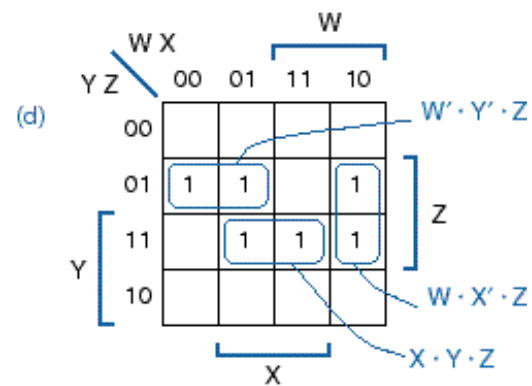


Mapas de Karnaugh

Ejemplos



$$F = W' \cdot X \cdot Z + W \cdot Y \cdot Z + X' \cdot Y' \cdot Z$$

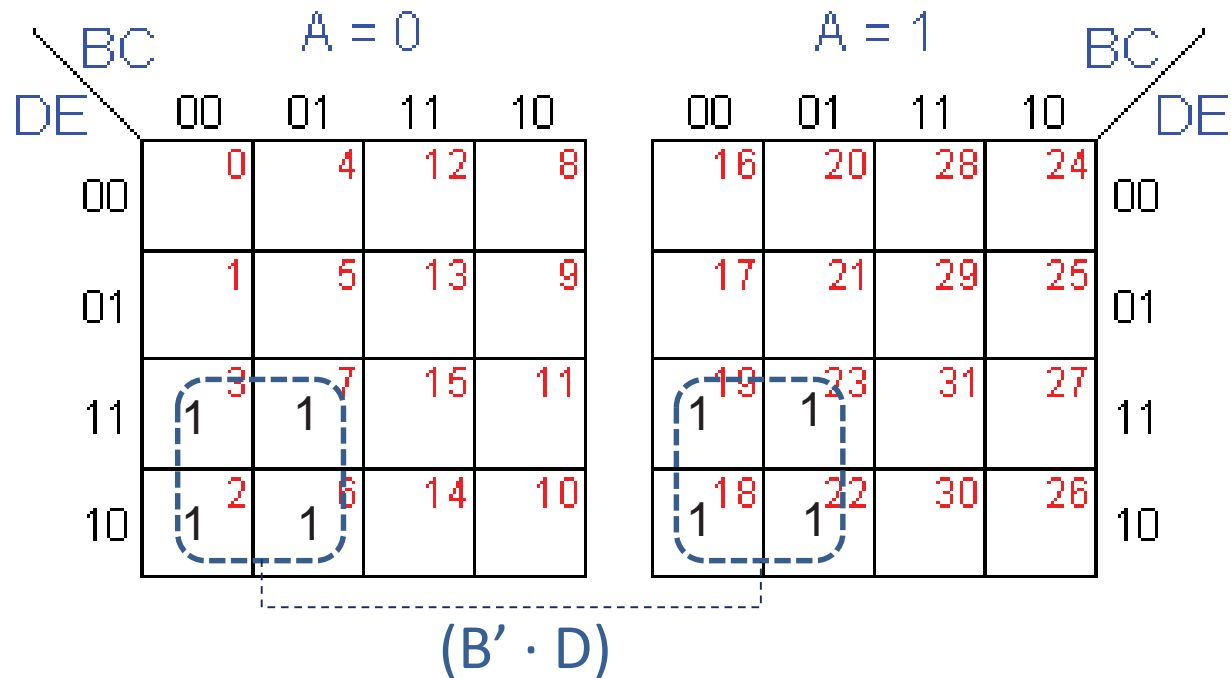


$$F = X \cdot Y \cdot Z + W \cdot X' \cdot Z + W' \cdot Y' \cdot Z$$

Mapas de Karnaugh

5 variables

$F(A,B,C,D,E)$



Mapas de Karnaugh

Simplificación de Productos de Sumas

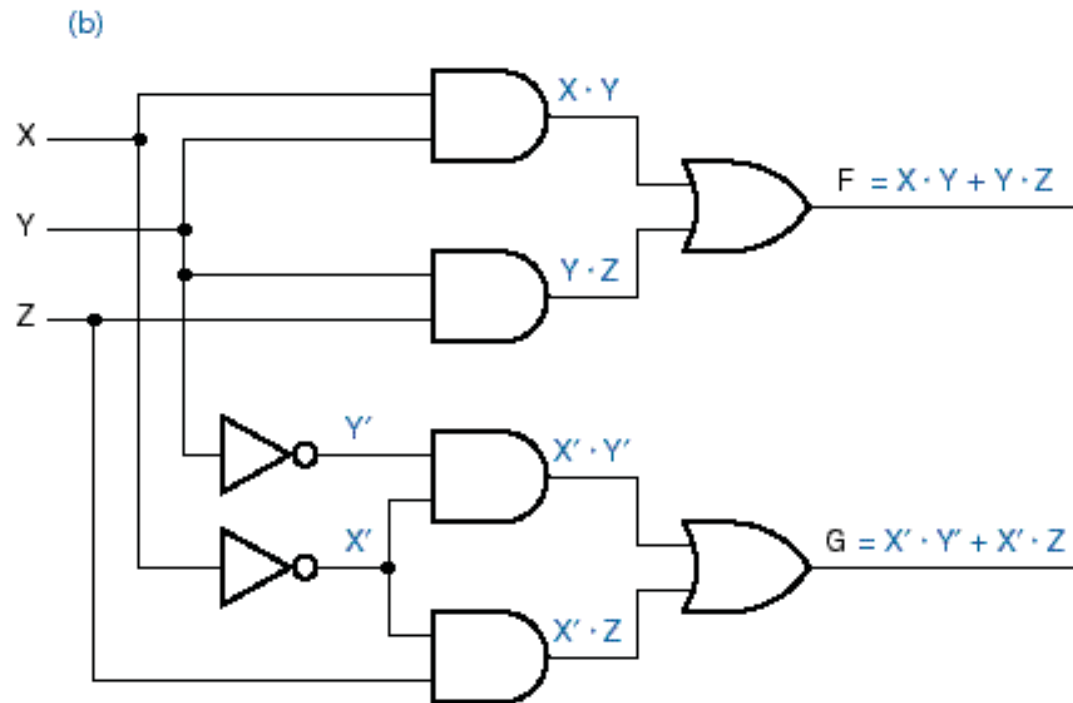
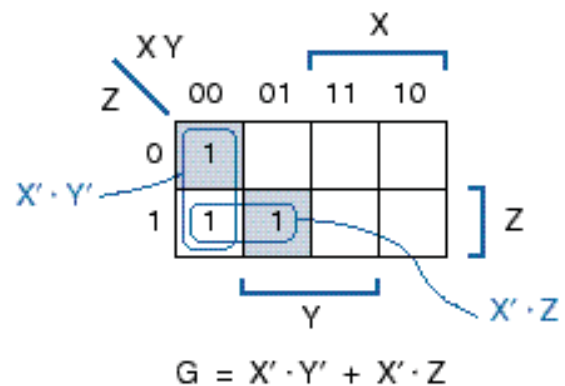
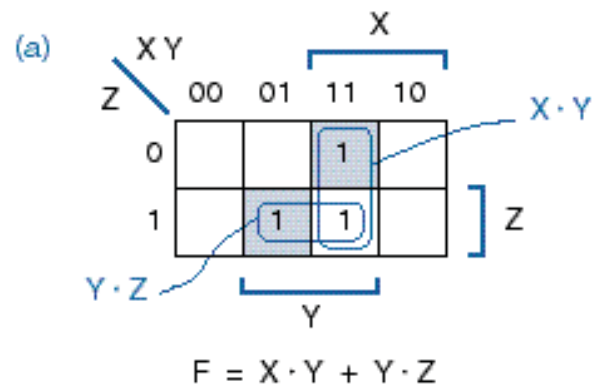
- En esta caso, se consideran los 0s del Mapa de Karnaugh.
- Cada celda es un término suma del producto

Finalmente, se construye un producto mínimo.

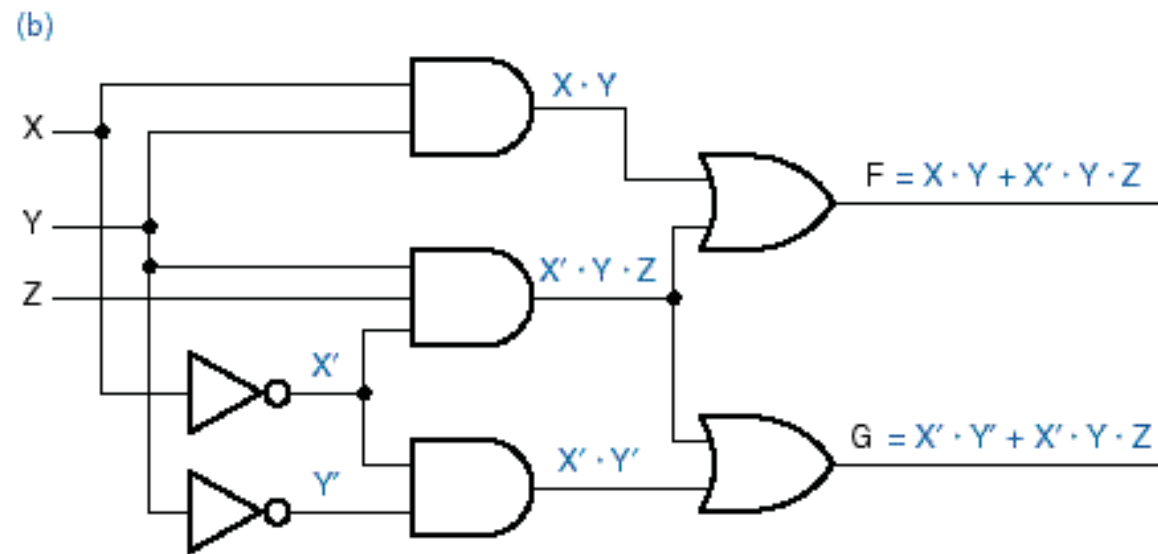
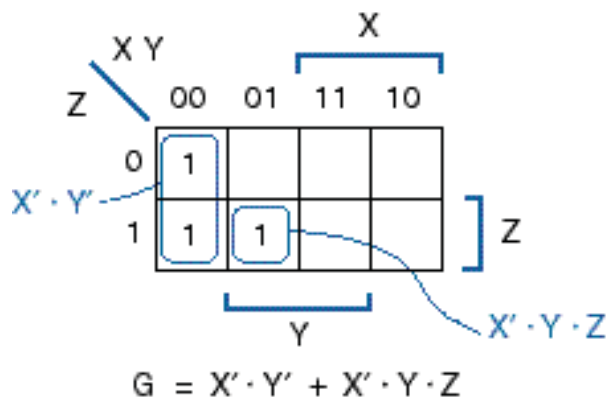
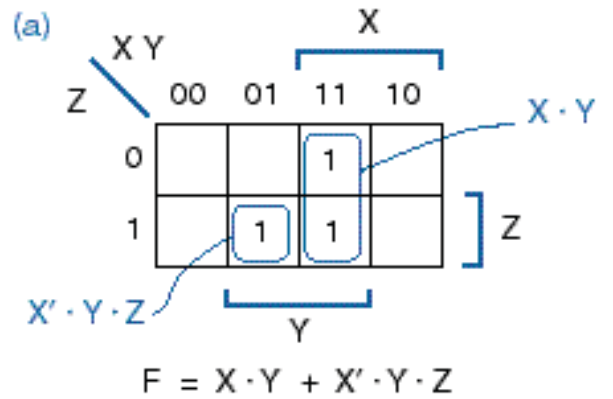
Una vez se han obtenido la suma y el producto mínimo, se escoge el más simple.

Mapas de Karnaugh

- Combinaciones de entrada “Don’t care”
- Minimización para múltiples salidas



Mapas de Karnaugh



Ejercicios

- Encontrar una suma mínima para:

1. $F = X' \cdot Z + X \cdot Y + X \cdot Y' \cdot Z$

2. $F = A' \cdot C' \cdot D + B' \cdot C \cdot D + A \cdot C' \cdot D + B \cdot C \cdot D$

3. $F = W \cdot X \cdot Z' + W \cdot X' \cdot Y \cdot Z + X \cdot Z$

Soluciones

A) $X \cdot Y + Z$

B) D

C) $X \cdot Z + W \cdot X + W \cdot Y \cdot Z$