Síntese de Circuitos Combinacionais

- Projecto Descrição informal dum circuito =>
  => Descrição formal duma Função
  => Diagrama lógico

- Nessa descrição, muitas vezes utiliza-se os termos “e”, “ou” e “não”.
- Por exemplo:

A saída ALARME fica a “1” se a entrada PÂNICO estiver a “1” *ou* se a entrada ACTIVADO estiver a “1” *e* a entrada SAINDO estiver a “0” *e* a casa *não* estiver SEGURA. A casa está segura se as entradas JANELA *e* PORTA *e* GARAGEM estiverem a “1”.

ALARME = PÂNICO + ACTIVADO . SAINDO’ . SEGURA’
SEGURA = JANELA . PORTA . GARAGEM

Síntese de Circuitos Combinacionais

- Implementação da Função Lógica

ALARME = PÂNICO + ACTIVADO . SAINDO' . (JANELA . PORTA . GARAGEM)'

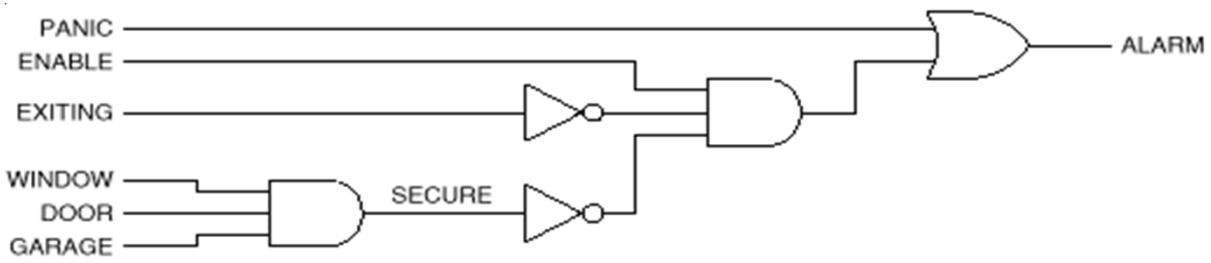


Figure 4–19 Alarm circuit derived from logic expression.

- A mesma função pode ser reescrita, sob a forma de soma de produtos, como sendo:

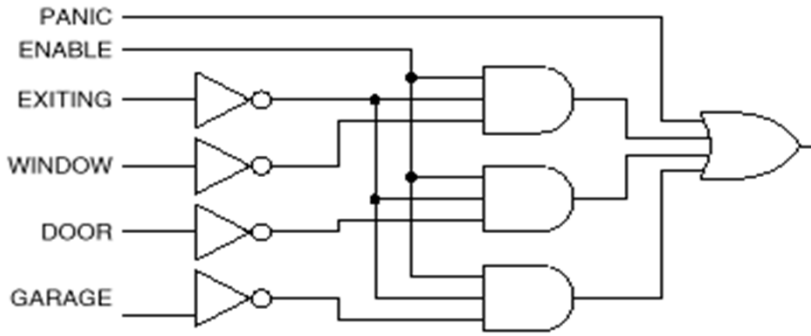


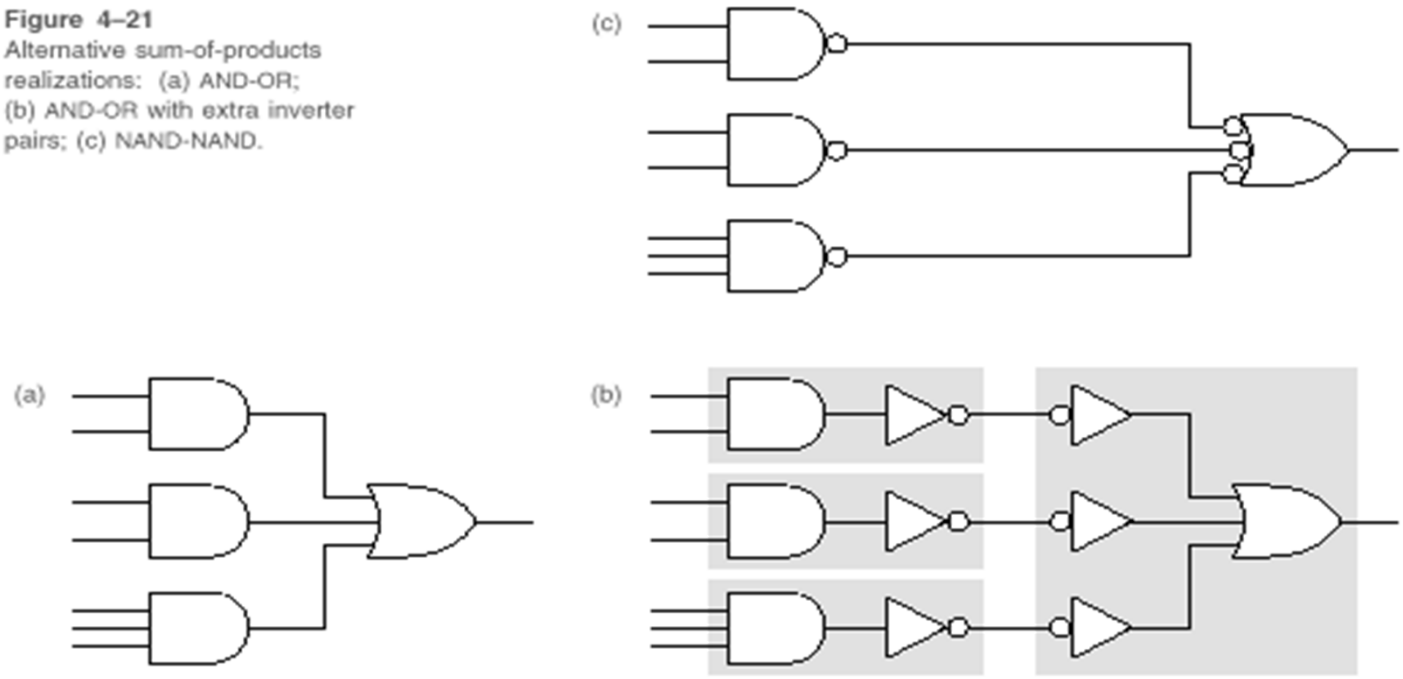
Figure 4–20
Sum-of-products version of alarm circuit.

ALARM = PANIC
+ ENABLE · EXITING' · WINDOW'
+ ENABLE · EXITING' · DOOR'
+ ENABLE · EXITING' · GARAGE'

Manipulações de circuitos

- Expressões lógicas obtidas a partir da descrição informal utilizam portas AND, OR e NOT.
- Existem diversas razões para usar portas NAND e NOR (são portas mais rápidas na maioria das tecnologias e implementam facilmente um inversor).

Figure 4–21
Alternative sum-of-products realizations: (a) AND-OR; (b) AND-OR with extra inverter pairs; (c) NAND-NAND.



Manipulações de circuitos

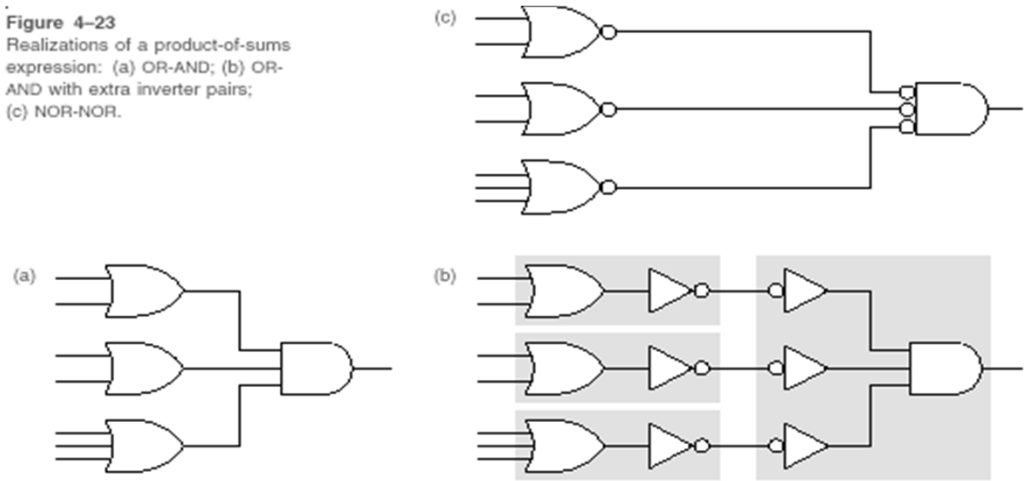


Figure 4–23
Realizations of a product-of-sums expression: (a) OR-AND; (b) OR-AND with extra inverter pairs; (c) NOR-NOR.

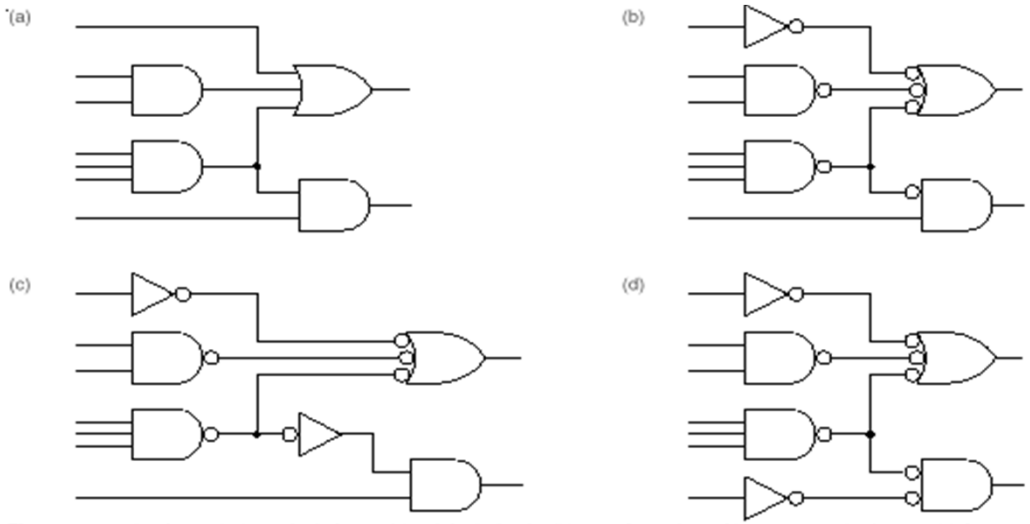


Figure 4–24 Logic-symbol manipulations: (a) original circuit; (b) transformation with a nonstandard gate; (c) inverter used to eliminate nonstandard gate; (d) preferred inverter placement.

Minimização de circuitos combinacionais

- Não é económico construir um circuito lógico a partir da primeira expressão lógica
- A maioria dos métodos de minimização têm por base os teoremas T10 e T10':

T10: Termo.Y + Termo.Y' = Termo

T10': (Termo+Y) . (Termo+Y') = Termo

Mapas de Karnaugh

- Representação gráfica das tabelas de verdade e servem como ponto de partida para minimização de circuitos combinacionais.
- O mapa, para uma função lógica de n-entradas, é um conjunto array de 2^n células, uma para cada possível combinação das entradas.

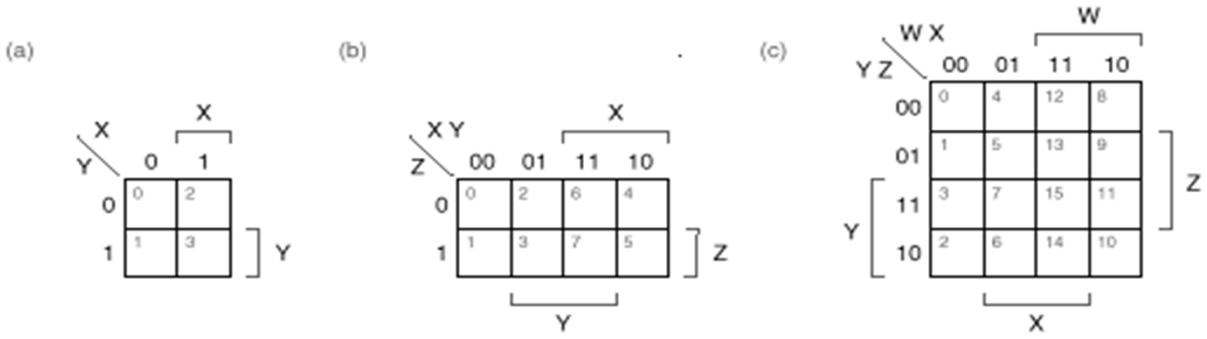
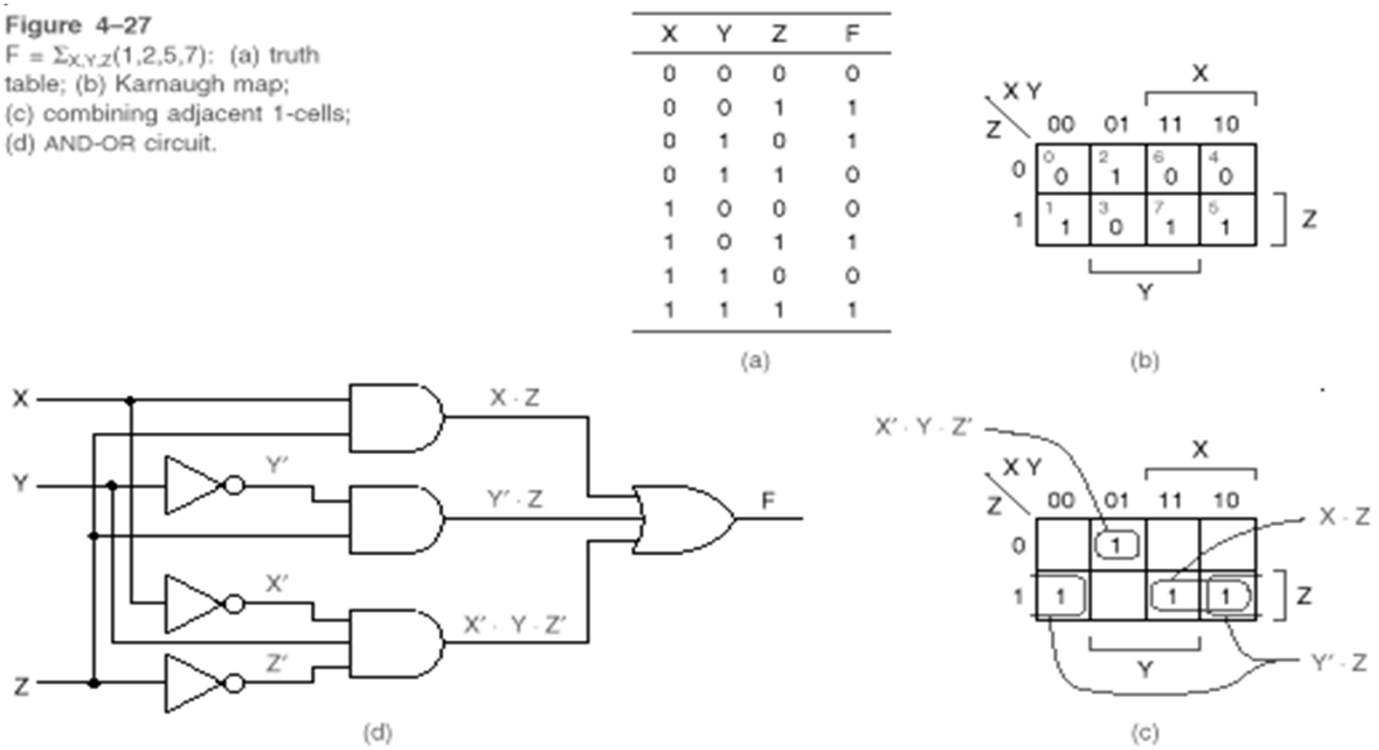


Figure 4–26 Karnaugh maps: (a) 2-variable; (b) 3-variable; (c) 4-variable.

- Cada célula do mapa de Karnaugh contém informação sobre a função. A célula contém “0” ou “1”, se ao termo mínimo correspondente na tabela de verdade da função, for respectivamente “0” ou “1”.
- Cada célula corresponde a uma combinação das entradas que difere das células adjacentes numa só variável.

Mapas de Karnaugh

$$F = \sum_{XYZ}(1,2,5,7)$$



- Células 5 e 7
  - F=...+ X . Y'. Z + X . Y . Z
  - F=...+ (X . Z) . Y' + (X . Z) . Y
  - F=...+ X . Z (porquê?)

Mapas de Karnaugh

$$F = \sum_{XYZ} (0,1,4,5,6)$$

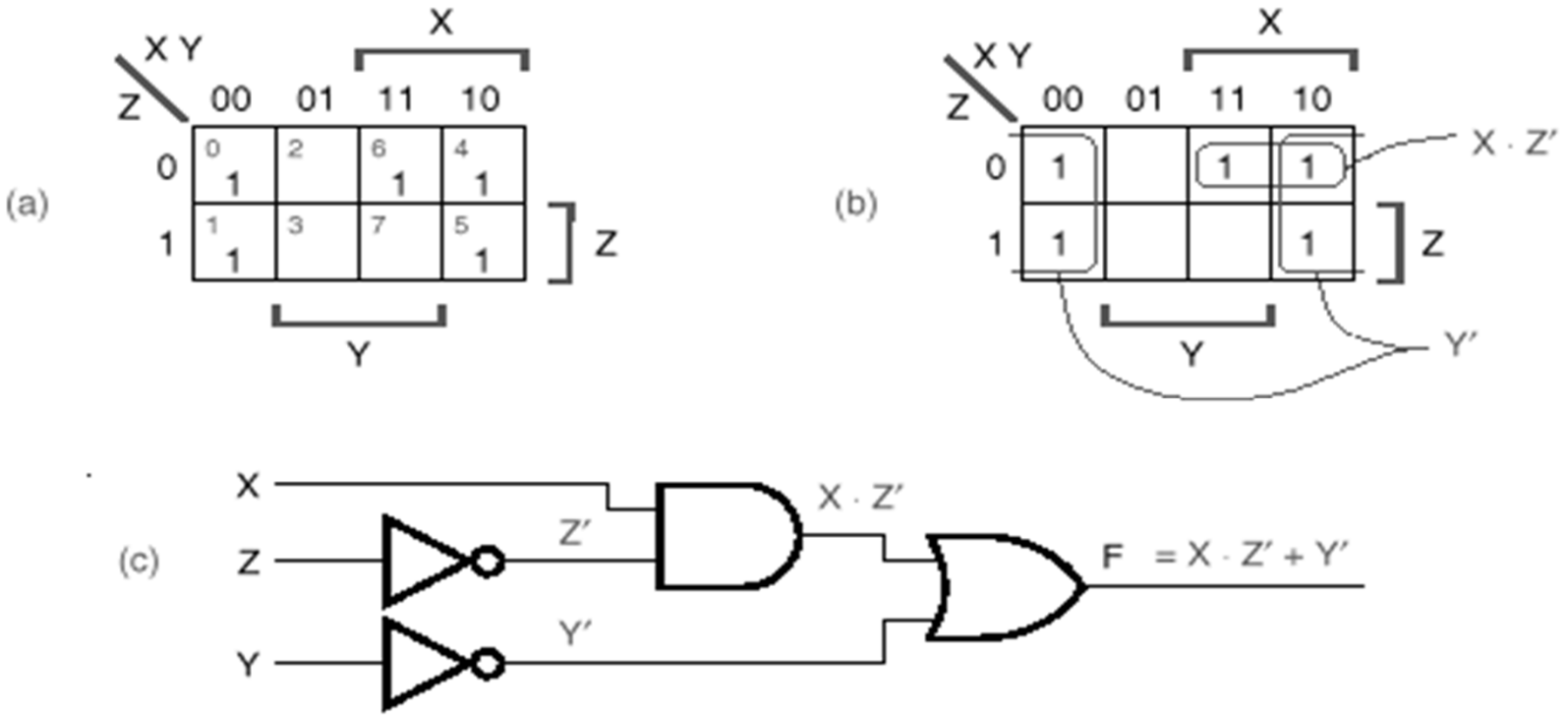
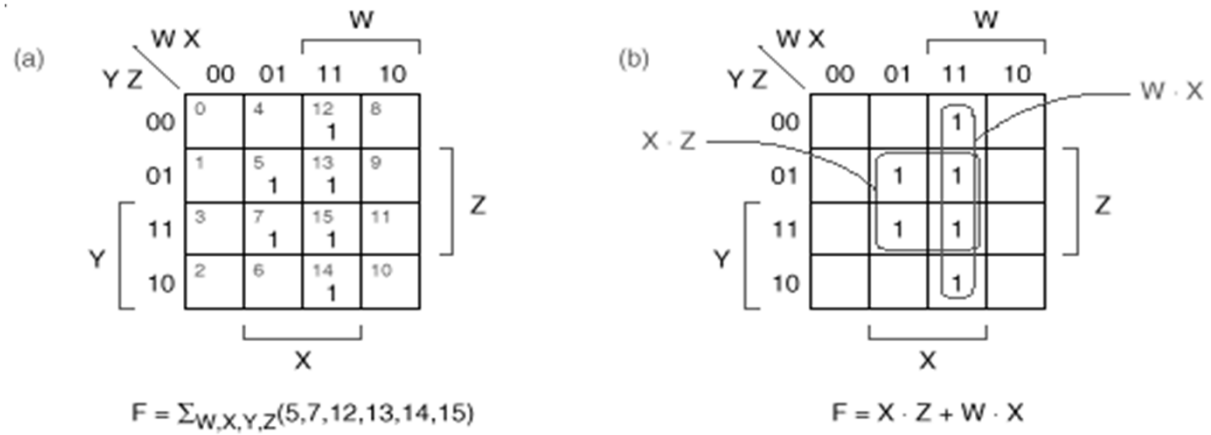


Figure 4–28 $F = \Sigma_{X,Y,Z}(0,1,4,5,6)$: (a) initial Karnaugh map; (b) Karnaugh map with circled product terms; (c) AND/OR circuit.

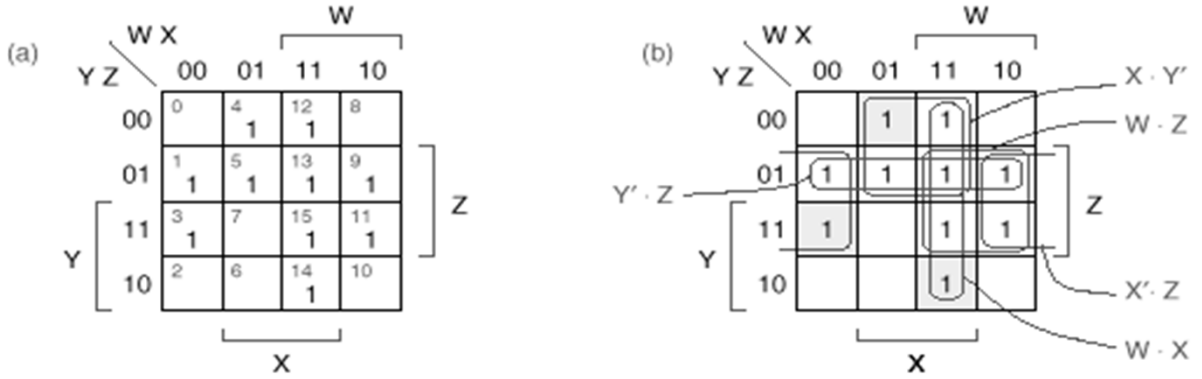
Mapas de Karnaugh

- Como seleccionar células por forma a obter a maior simplificação possível?



$F = \Sigma_{W,X,Y,Z}(5,7,12,13,14,15)$ $\quad F = X \cdot Z + W \cdot X$

Figure 4–30 $F = \Sigma_{W,X,Y,Z}(5,7,12,13,14,15)$: (a) Karnaugh map; (b) prime implicants.



$F = \Sigma_{W,X,Y,Z}(1,3,4,5,9,11,12,13,14,15)$ $\quad F = X \cdot Y' + X' \cdot Z + W \cdot X$

Figure 4–31 $F = \Sigma_{W,X,Y,Z}(1,3,4,5,9,11,12,13,14,15)$: (a) Karnaugh map; (b) prime implicants and distinguished 1-cells.

Mapas de Karnaugh

- Como seleccionar células por forma a obter a maior simplificação possível?

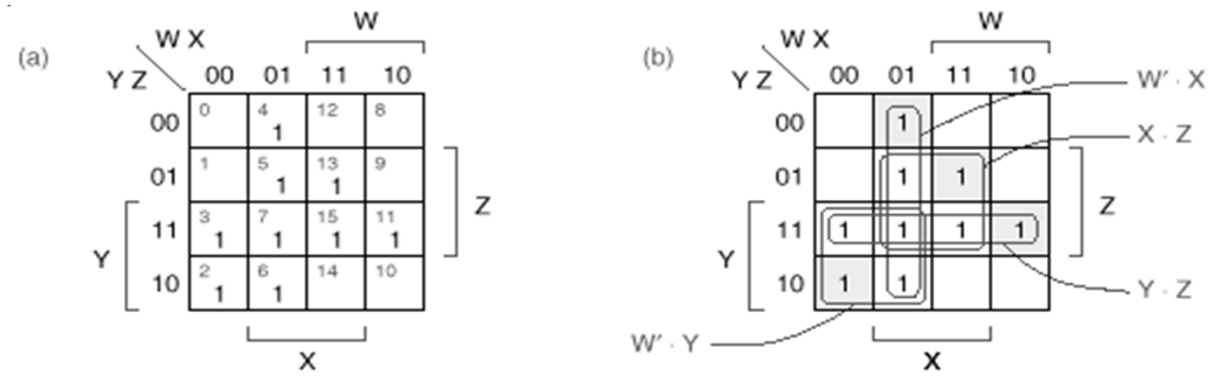


Figure 4–32 F = $\Sigma_{W,X,Y,Z}$(2,3,4,5,6,7,11,13,15): (a) Karnaugh map; (b) prime implicants and distinguished 1-cells.

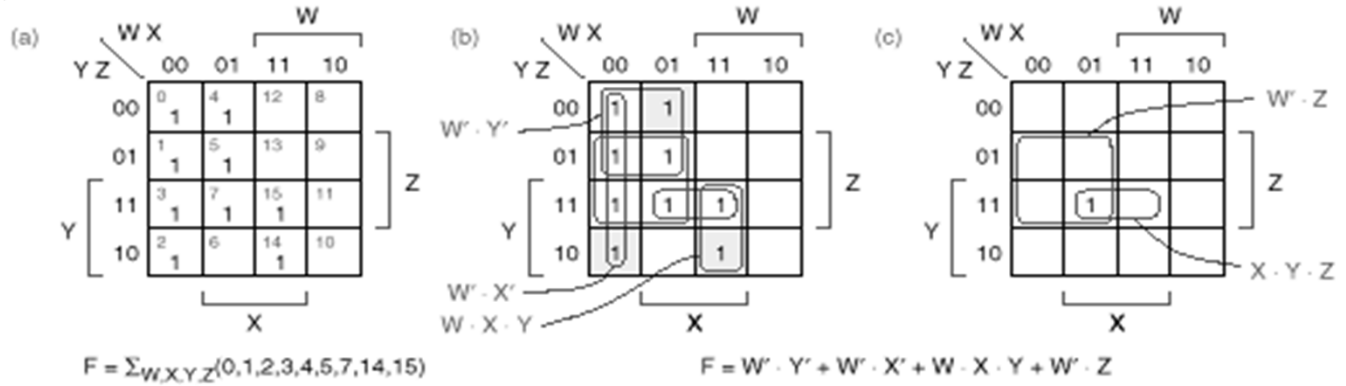
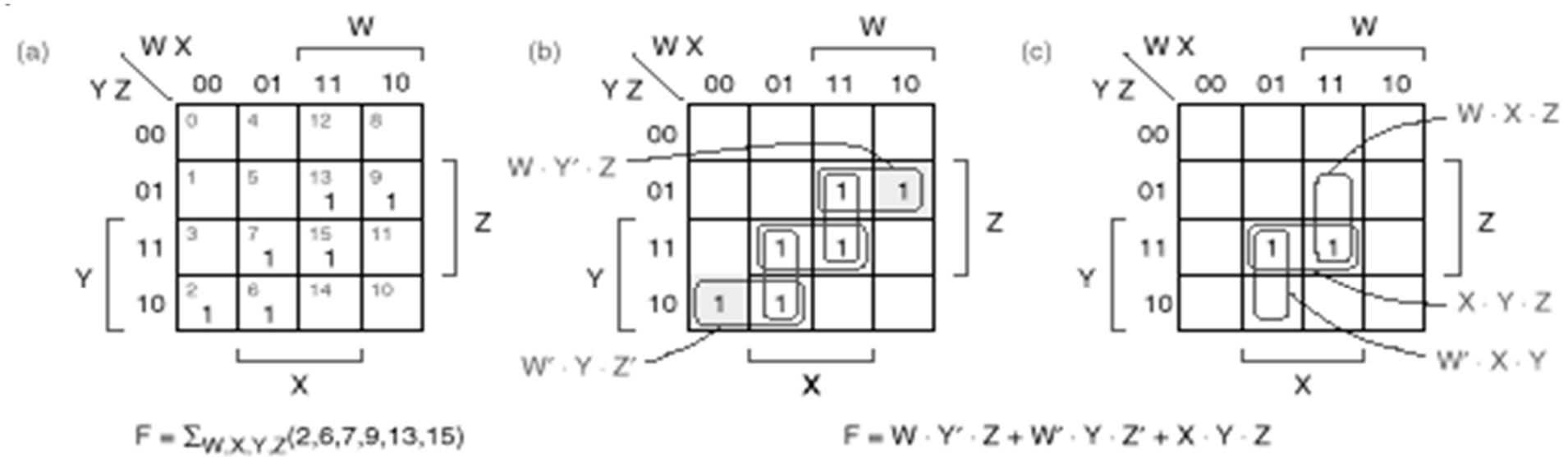


Figure 4–33 F = $\Sigma_{W,X,Y,Z}$(0,1,2,3,4,5,7,14,15): (a) Karnaugh map; (b) prime implicants and distinguished 1-cells; (c) reduced map after removal of essential prime implicants and covered 1-cells.

Mapas de Karnaugh

- Como seleccionar células por forma a obter a maior simplificação possível?



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

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

Figure 4–34 $F = \Sigma_{W,X,Y,Z}(2,6,7,9,13,15)$: (a) Karnaugh map; (b) prime implicants and distinguished 1-cells; (c) reduced map after removal of essential prime implicants and covered 1-cells.

Mapas de Karnaugh

- Como seleccionar células por forma a obter a maior simplificação possível?

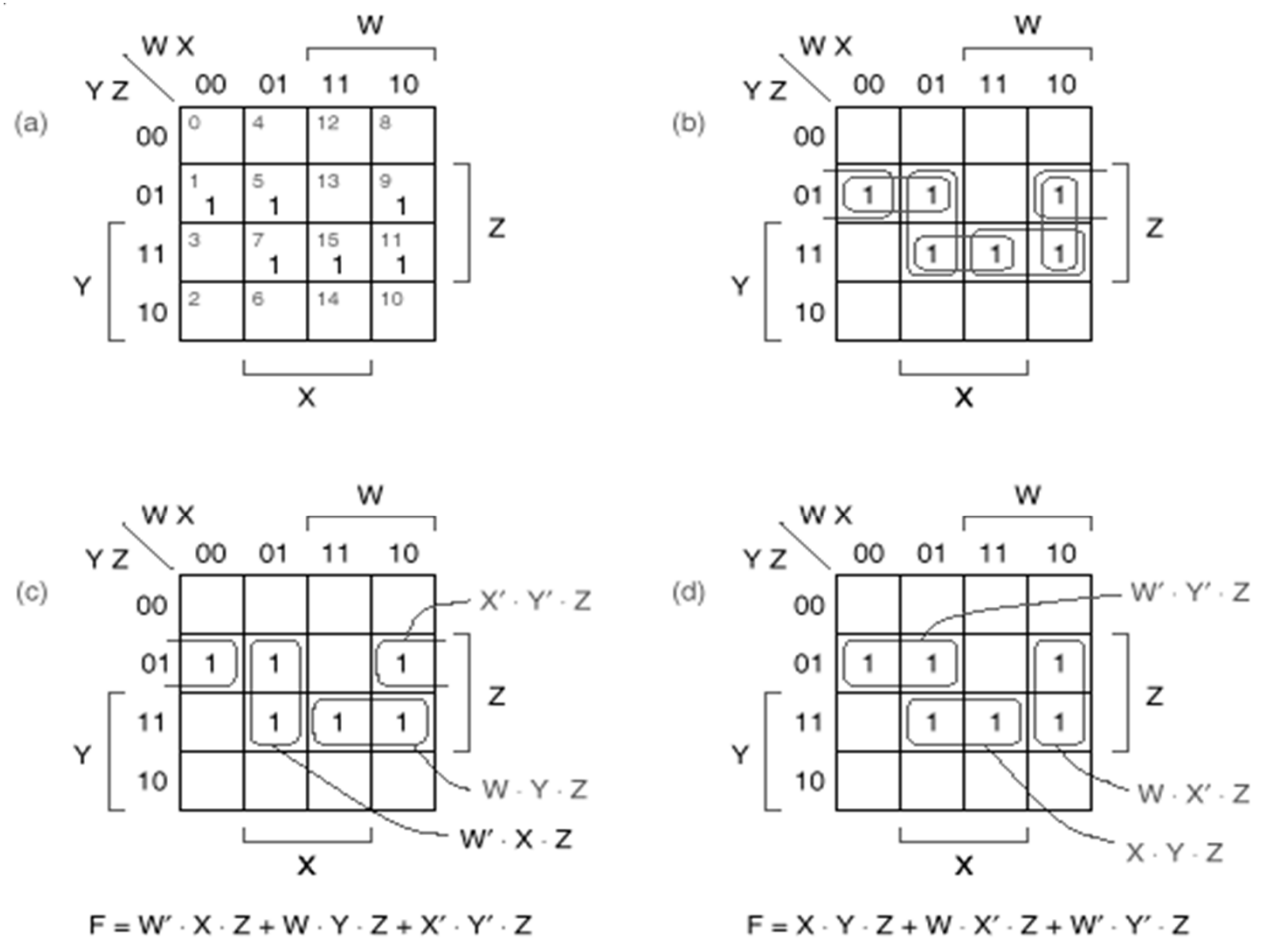


Figure 4–35 $F = \Sigma_{W,X,Y,Z}(1,5,7,9,11,15)$: (a) Karnaugh map; (b) prime implicants; (c) a minimal sum; (d) another minimal sum.

Mapas de Karnaugh

- Como seleccionar células por forma a obter a maior simplificação possível?

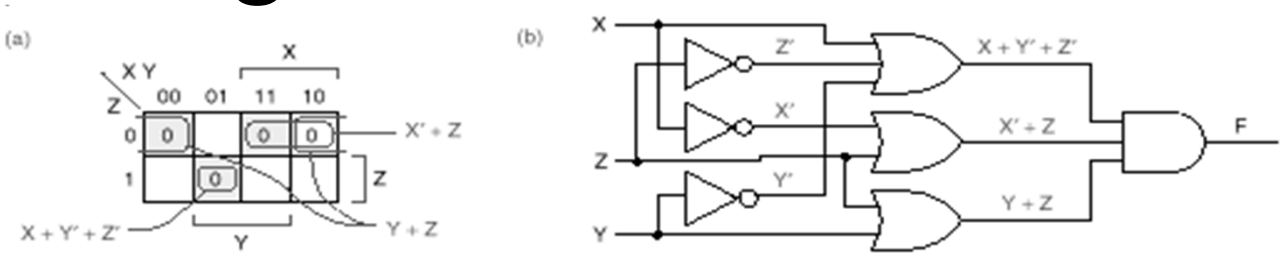


Figure 4–36 $F = \Sigma_{X,Y,Z}(1,2,5,7) = \Pi_{X,Y,Z}(0,3,4,6)$: (a) Karnaugh map; (b) OR/AND circuit.

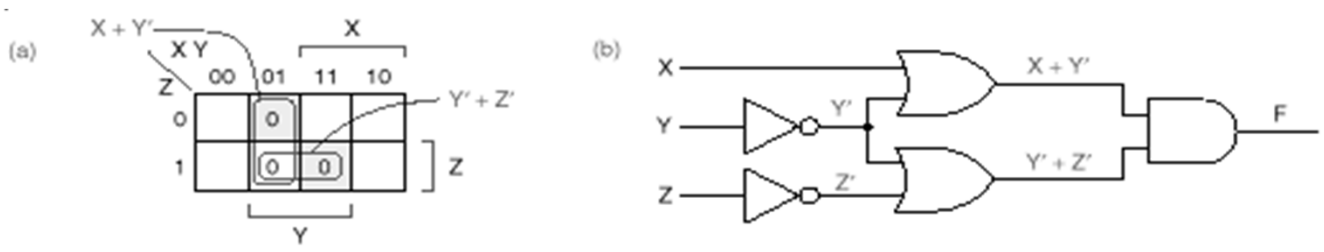


Figure 4–37 $F = \Sigma_{X,Y,Z}(0,1,4,5,6) = \Pi_{X,Y,Z}(2,3,7)$: (a) Karnaugh map; (b) OR/AND circuit.

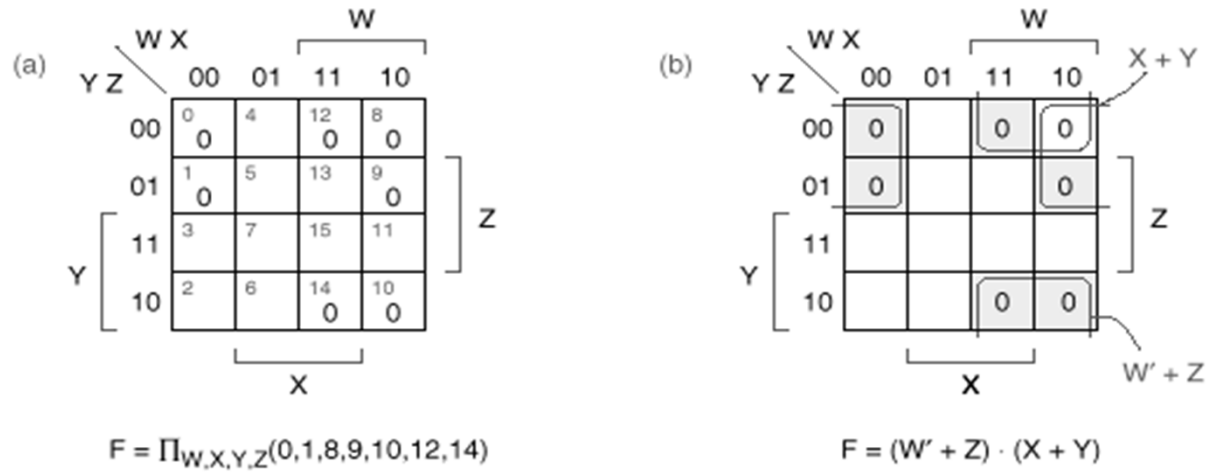


Figure 4–38 $F = \Sigma_{W,X,Y,Z}(2,3,4,5,6,7,11,13,15) = \Pi_{W,X,Y,Z}(0,1,8,9,10,12,14)$: (a) Karnaugh map; (b) prime implicates and distinguished 0-cells.

Mapas de Karnaugh

- Como seleccionar células por forma a obter a maior simplificação possível?
- Levando em conta as saídas indiferentes, d, (don't care conditions)?

$$F = \sum_{N3N2N1N0}(1,2,3,5,7) + d(10,11,12,13,14,15)$$

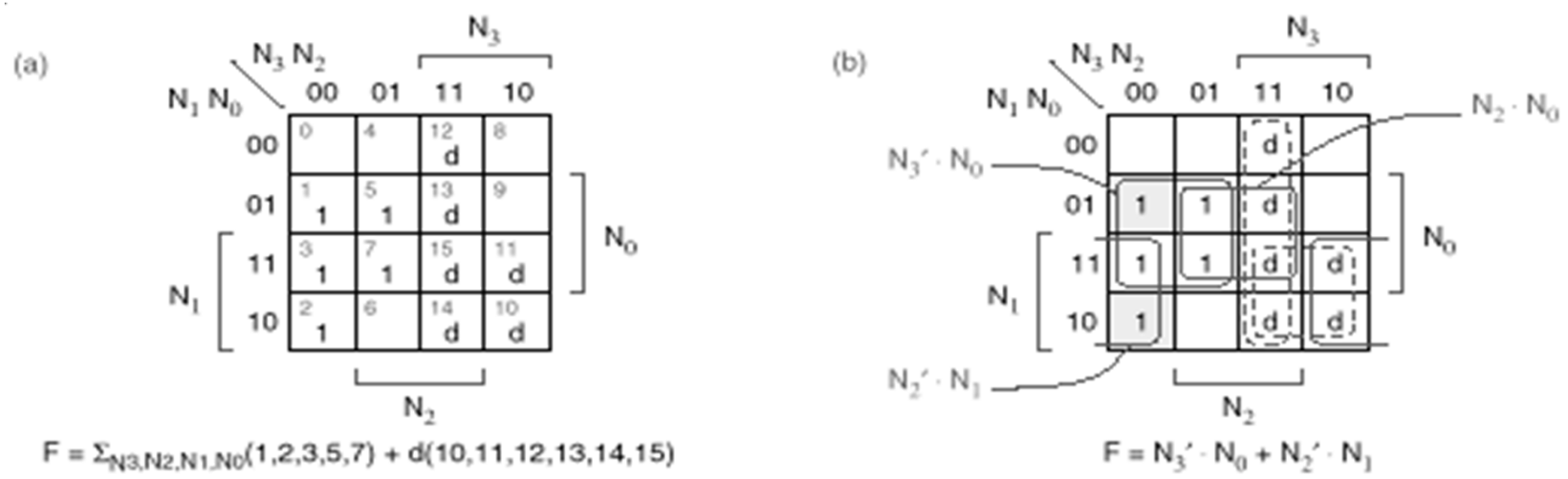


Figure 4–39 Prime BCD-digit detector: (a) initial Karnaugh map; (b) Karnaugh map with prime implicants and distinguished 1-cells.

Mapas de Karnaugh

- Como seleccionar células por forma a obter a maior simplificação possível?

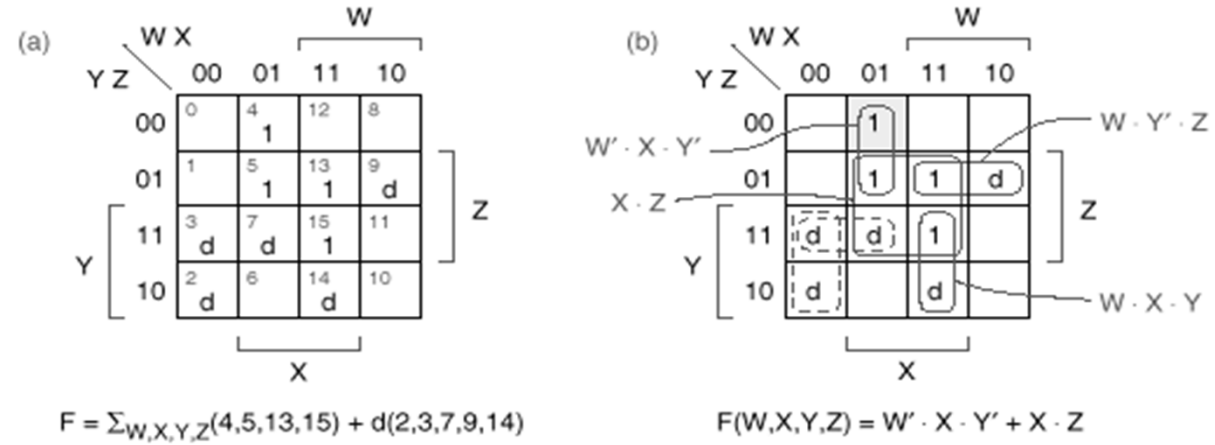


Figure 4–40 $F = \Sigma_{W,X,Y,Z}(4,5,13,15) + d(2,3,7,9,14)$: (a) Karnaugh map; (b) prime implicants and distinguished 1-cells.

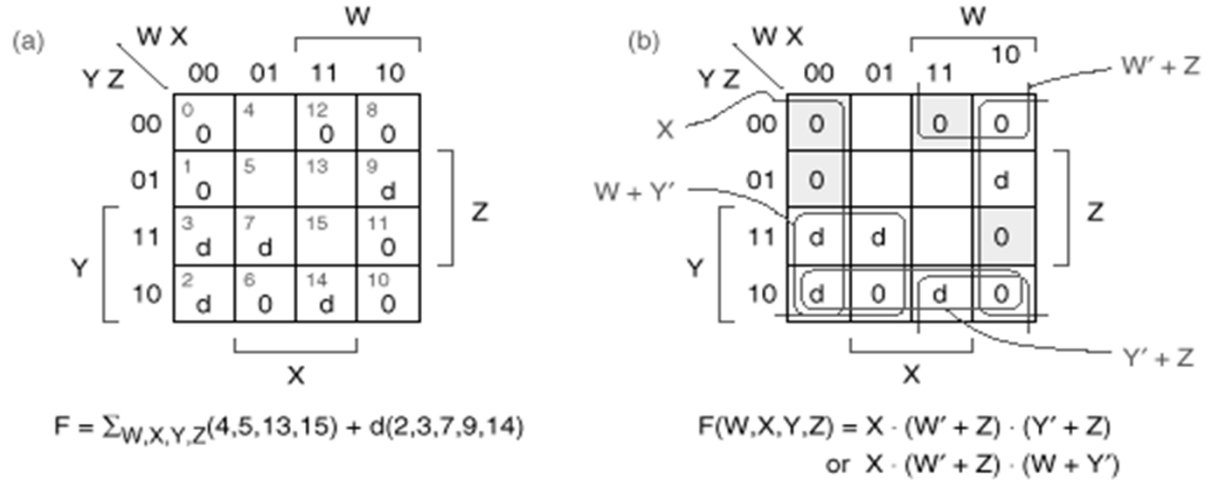


Figure 4–41 $F = \Sigma_{W,X,Y,Z}(4,5,13,15) + d(2,3,7,9,14)$: (a) Karnaugh map; (b) prime implicates and distinguished 0-cells.

Mapas de Karnaugh

- Minimização de circuitos com múltipla saída
- Importante para PLDs

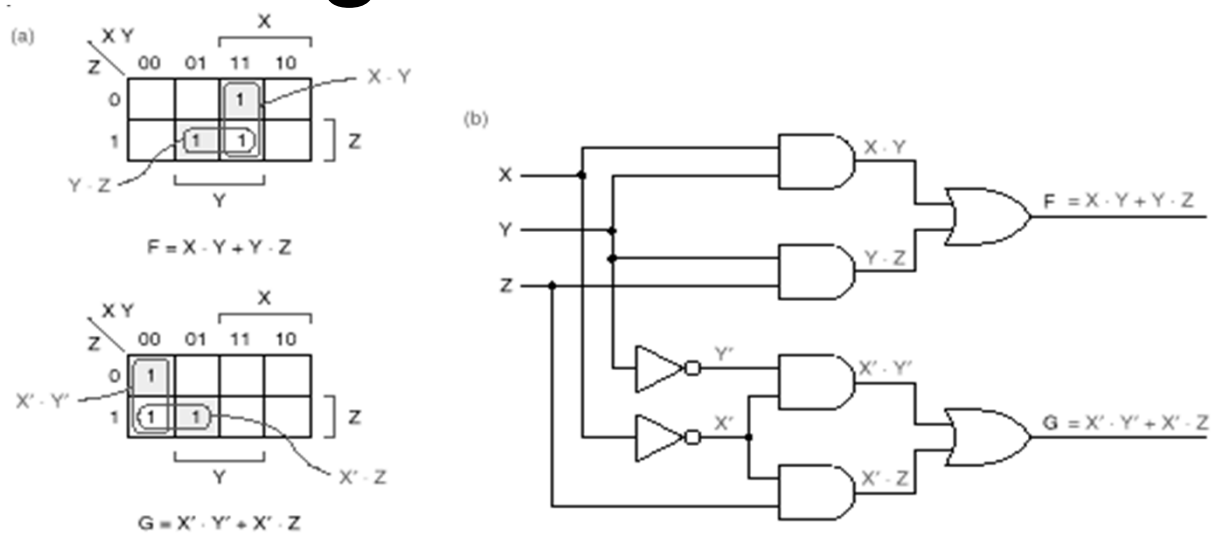


Figure 4–42 Treating a 2-output design as two independent single-output designs: (a) Karnaugh maps; (b) "minimal" circuit.

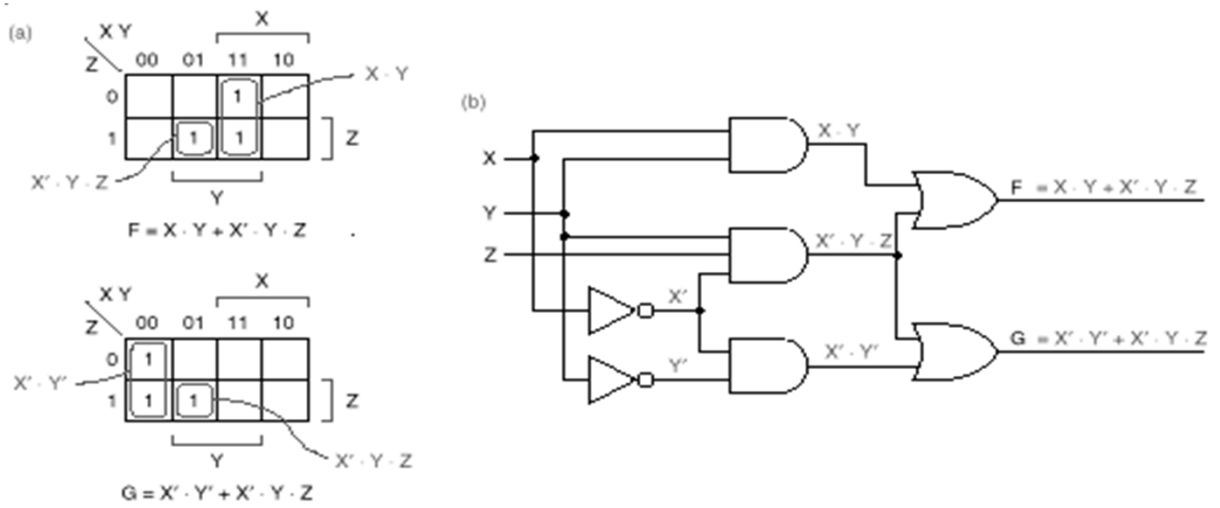


Figure 4–43 Multiple-output minimization for a 2-output circuit: (a) minimized maps including a shared term; (b) minimal multiple-output circuit.