2-1 Table 2-1 Truth Tables for the Three Basic Logic Operations

AND

X	Y	Z
0	0	0
0	1	0
1	0	0
1	1	1

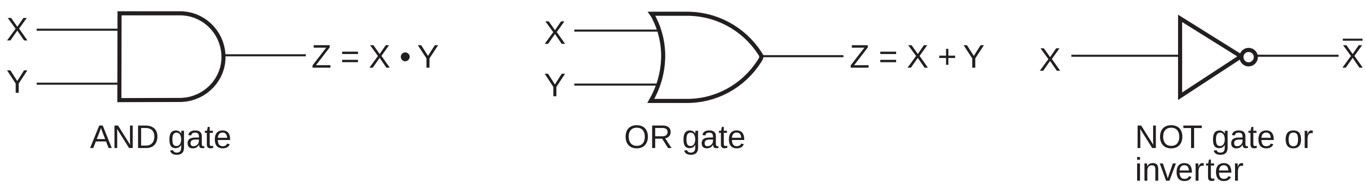
OR

X	Y	Z
0	0	0
0	1	1
1	0	1
1	1	1

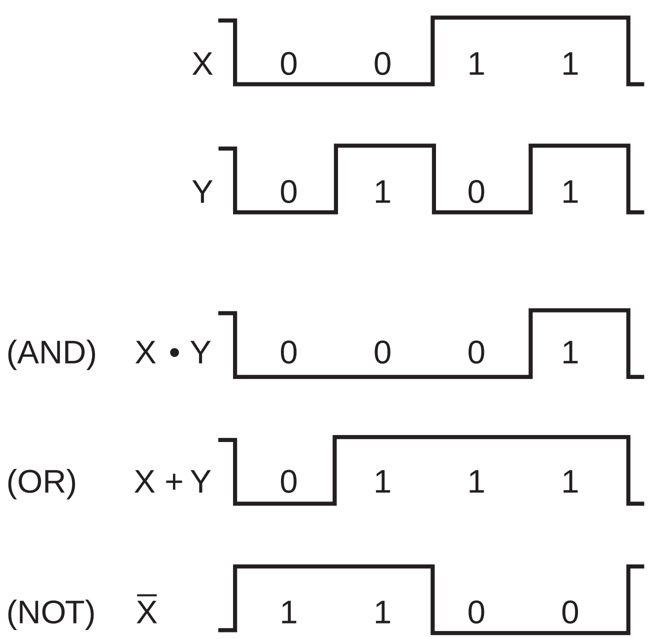
NOT

X	Z
0	1
1	0

Figure 2-1 Digital Logic Gates

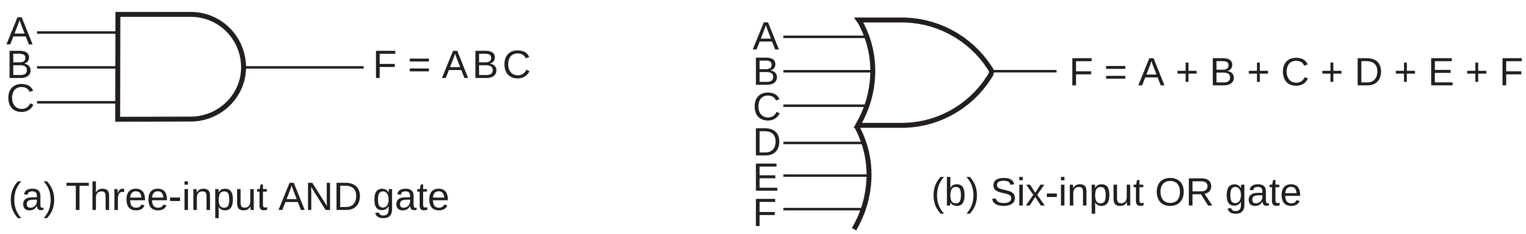


(a) Graphic symbols



(b) Timing diagram

Figure 2-2 Gates with More than Two Inputs



(a) Three-input AND gate

(b) Six-input OR gate

Table 2-2 Truth Table for the Function F = X + Y'Z

X	Y	Z	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

2-5 Figure 2-3 Logic Circuit Diagram for F = X + Y'Z

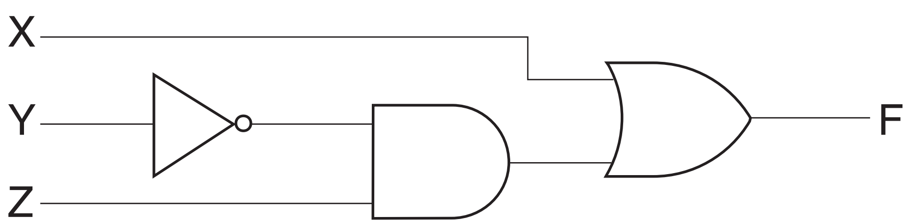


Table 2-3 Basic Identities of Boolean Algebra

1.	$X + 0 = X$	2.	$X \cdot 1 = X$	
3.	$X + 1 = 1$	4.	$X \cdot 0 = 0$	
5.	$X + X = X$	6.	$X \cdot X = X$	
7.	$X + \overline{X} = 1$	8.	$X \cdot \overline{X} = 0$	
9.	$\overline{\overline{X}} = X$			
10.	$X + Y = Y + X$	11.	$XY = YX$	Commutative
12.	$X + (Y + Z) = (X + Y) + Z$	13.	$X(YZ) = (XY)Z$	Associative
14.	$X(Y + Z) = XY + XZ$	15.	$X + YZ = (X + Y)(X + Z)$	Distributive
16.	$\overline{X + Y} = \overline{X} \cdot \overline{Y}$	17.	$\overline{X \cdot Y} = \overline{X} + \overline{Y}$	DeMorgan's

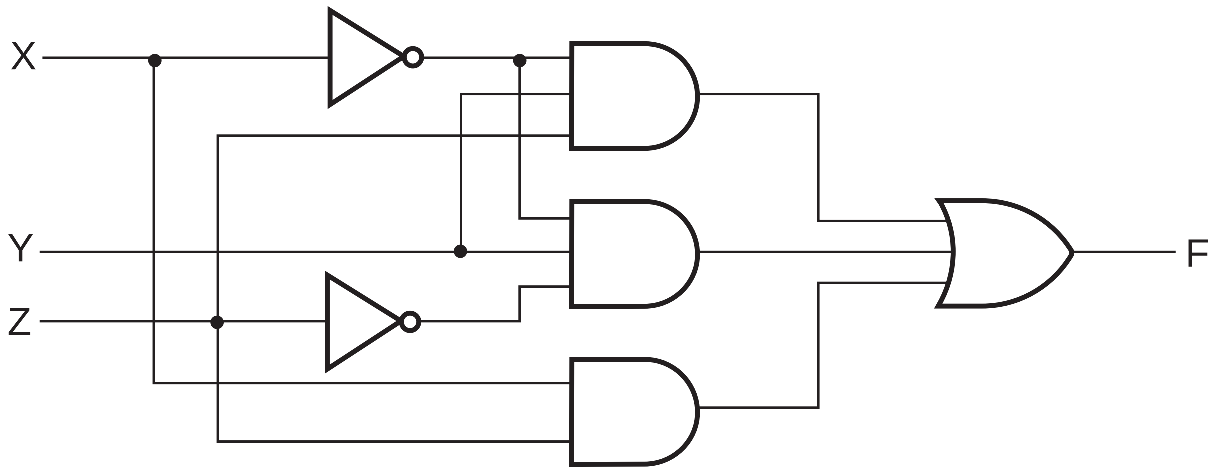
Table 2-4 Truth Tables to Verify DeMorgan's Theorem

A)	X	Y		$\overline{X+Y}$	B	X	Y	$\overline{X}$	$\overline{Y}$	$\overline{X}?\overline{Y}$
	0	0	0	1		0	0	1	1	1
	0	1	1	0		0	1	1	0	0
	1	0	1	0		1	0	0	1	0
	1	1	1	0		1	1	0	0	0

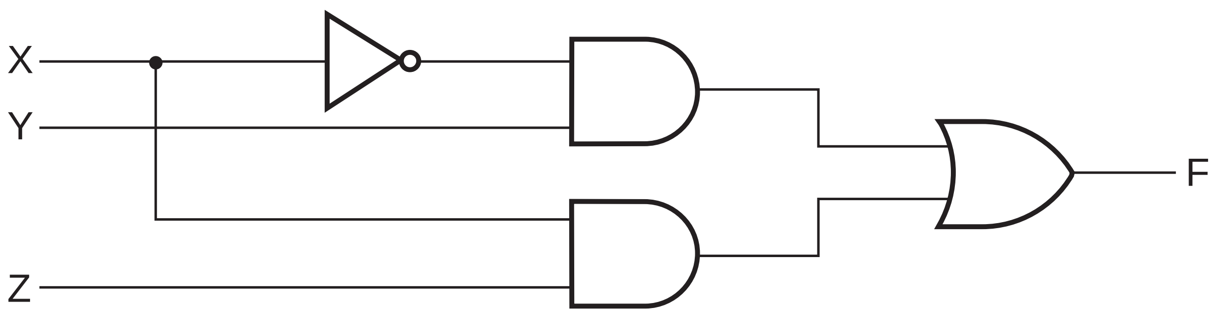
Table 2-5 Truth Table for Boolean Function

X	Y	Z	F
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

Figure 2-4 Implementation of Boolean Function with Gates



(a) $F = \overline{X}YZ + \overline{X}Y\overline{Z} + XZ$



(b) $F = \overline{X}Y + XZ$

Table 2-6 Minterms for Three Variables

X	Y	Z	Product Term	Symbol	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7
0	0	0	$\overline{X}\,\overline{Y}\,\overline{Z}$	m_0	1	0	0	0	0	0	0	0
0	0	1	$\overline{X}\,\overline{Y}Z$	m_1	0	1	0	0	0	0	0	0
0	1	0	$\overline{X}Y\overline{Z}$	m_2	0	0	1	0	0	0	0	0
0	1	1	$\overline{X}YZ$	m_3	0	0	0	1	0	0	0	0
1	0	0	$X\overline{Y}\,\overline{Z}$	m_4	0	0	0	0	1	0	0	0
1	0	1	$X\overline{Y}Z$	m_5	0	0	0	0	0	1	0	0
1	1	0	$XY\overline{Z}$	m_6	0	0	0	0	0	0	1	0
1	1	1	XYZ	m_7	0	0	0	0	0	0	0	1

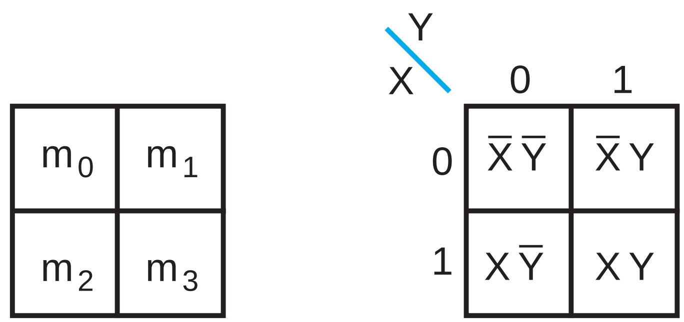
Table 2-7 Maxterms for Three Variables

X	Y	Z	Sum Term	Symbol	M_0	M_1	M_2	M_3	M_4	M_5	M_6	M_7
0	0	0	$X + Y + Z$	M_0	0	1	1	1	1	1	1	1
0	0	1	$X + Y + \overline{Z}$	M_1	1	0	1	1	1	1	1	1
0	1	0	$X + \overline{Y} + Z$	M_2	1	1	0	1	1	1	1	1
0	1	1	$X + \overline{Y} + \overline{Z}$	M_3	1	1	1	0	1	1	1	1
1	0	0	$\overline{X} + Y + Z$	M_4	1	1	1	1	0	1	1	1
1	0	1	$\overline{X} + Y + \overline{Z}$	M_5	1	1	1	1	1	0	1	1
1	1	0	$\overline{X} + \overline{Y} + Z$	M_6	1	1	1	1	1	1	0	1
1	1	1	$\overline{X} + \overline{Y} + \overline{Z}$	M_7	1	1	1	1	1	1	1	0

Table 2-8 Boolean Function of Three Variables

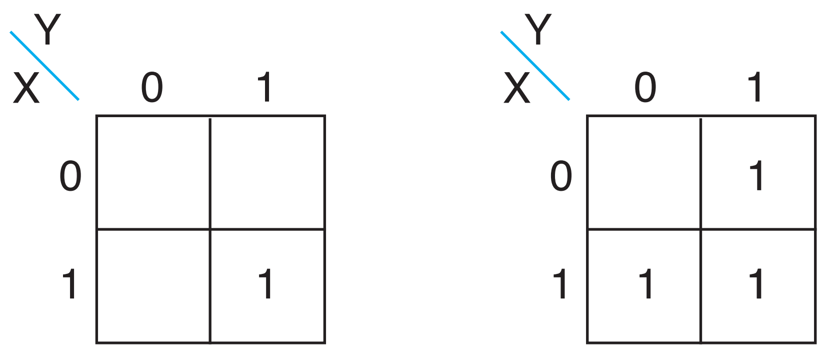
(a) X	Y	Z	F	$\overline{F}$	(b) X	Y	Z	E
0	0	0	1	0	0	0	0	1
0	0	1	0	1	0	0	1	1
0	1	0	1	0	0	1	0	1
0	1	1	0	1	0	1	1	0
1	0	0	0	1	1	0	0	1
1	0	1	1	0	1	0	1	1
1	1	0	0	1	1	1	0	0
1	1	1	1	0	1	1	1	0

Figure 2-8 Two-Variable Map



(a) (b)

Figure 2-9 Representation of Functions in the Map



(a) XY   (b) X + Y

Figure 2-10 Three-Variable Map

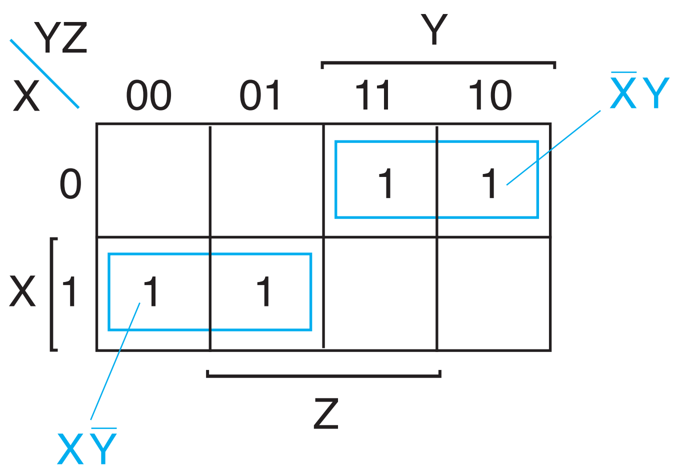
m_0	m_1	m_3	m_2
m_4	m_5	m_7	m_6

(a)

				Y	Y
	X \ YZ	00	01	11	10
	0	$\overline{X}\,\overline{Y}\,\overline{Z}$	$\overline{X}\,\overline{Y}\,Z$	$\overline{X}\,Y\,Z$	$\overline{X}\,Y\,\overline{Z}$
X	1	$X\,\overline{Y}\,\overline{Z}$	$X\,\overline{Y}\,Z$	$X\,Y\,Z$	$X\,Y\,\overline{Z}$
			Z	Z	

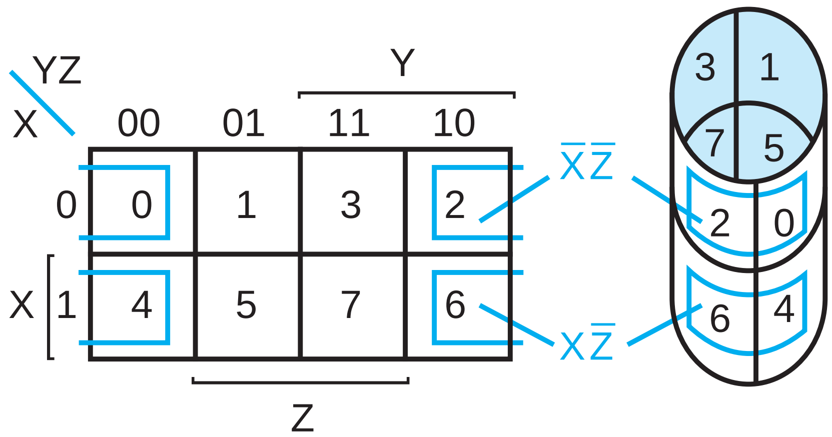
(b)

Figure 2-11 Map for Example 2-3



M. Morris Mano & Charles R. Kime

Figure 2-12 Three-Variable Map: Flat and on a Cylinder to Show Adjacent Squares



(a)

(b)

Figure 2-13 Product Terms Using Four Minterms

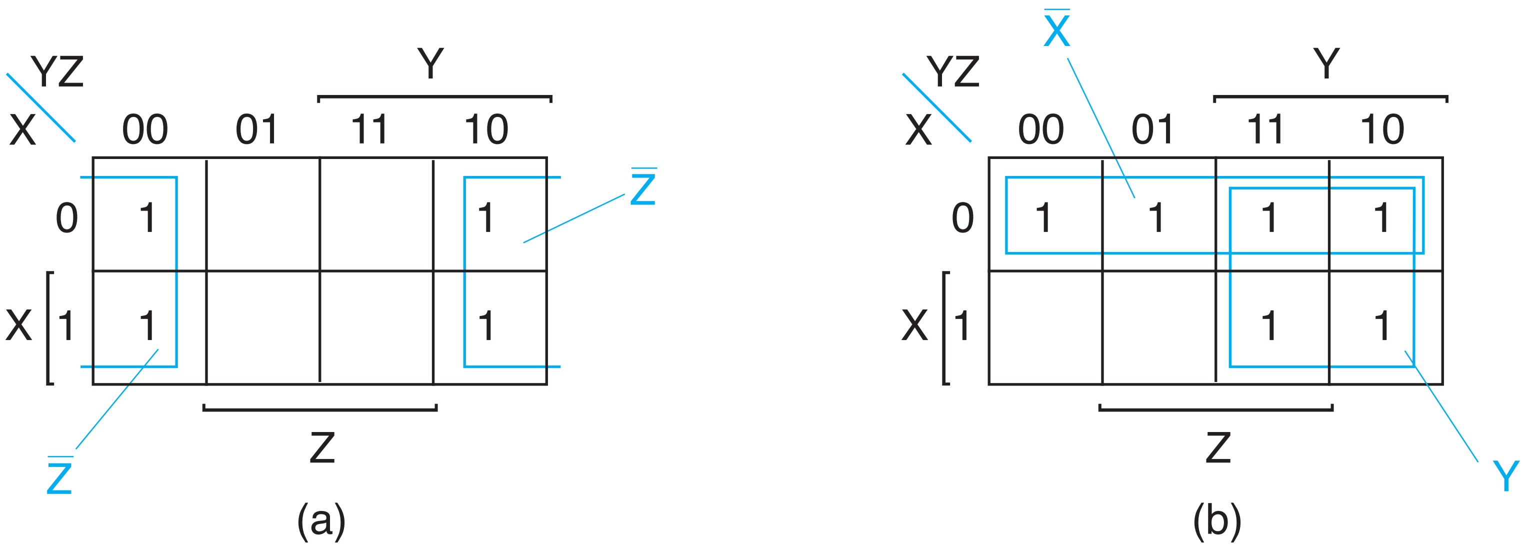
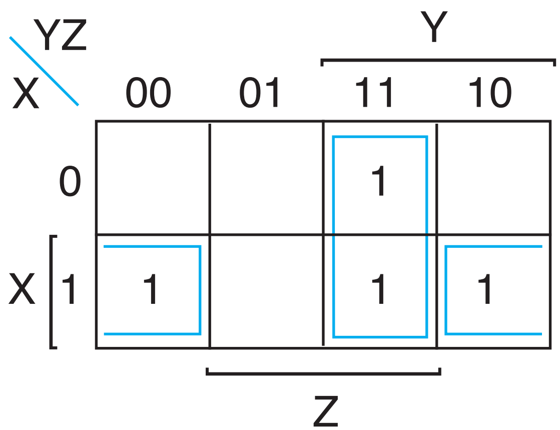
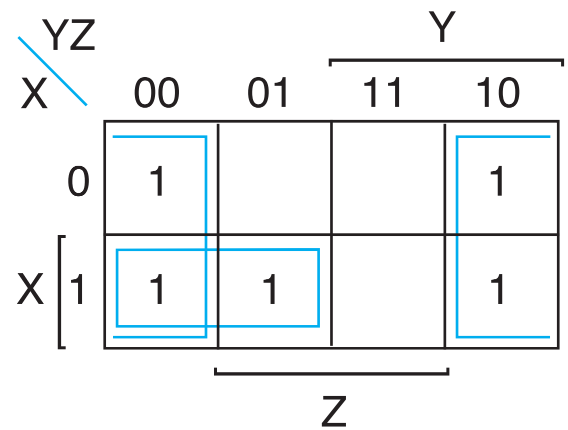


Figure 2-14 Maps for Example 2-4



(a) $F_1(X, Y, Z) = \ m(3, 4, 6, 7)$

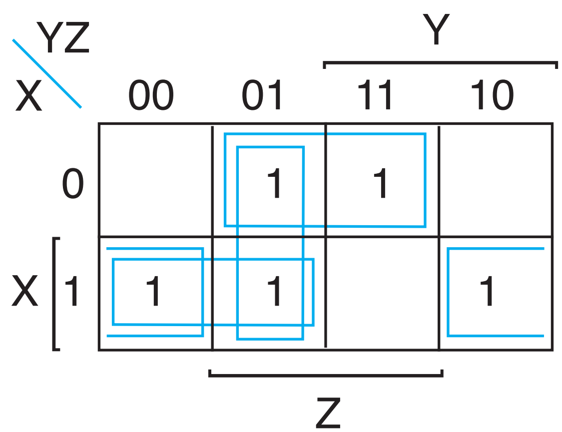
$= YZ + X\overline{Z}$



(b) $F_2(X, Y, Z) = \ m(0, 2, 4, 5, 6)$

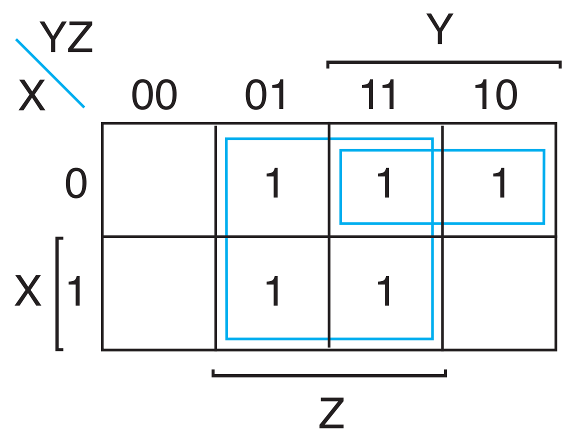
$= \overline{Z} + X\overline{Y}$

Figure 2-15 F(X, Y, Z) = SUM m(1, 3, 4, 5, 6)



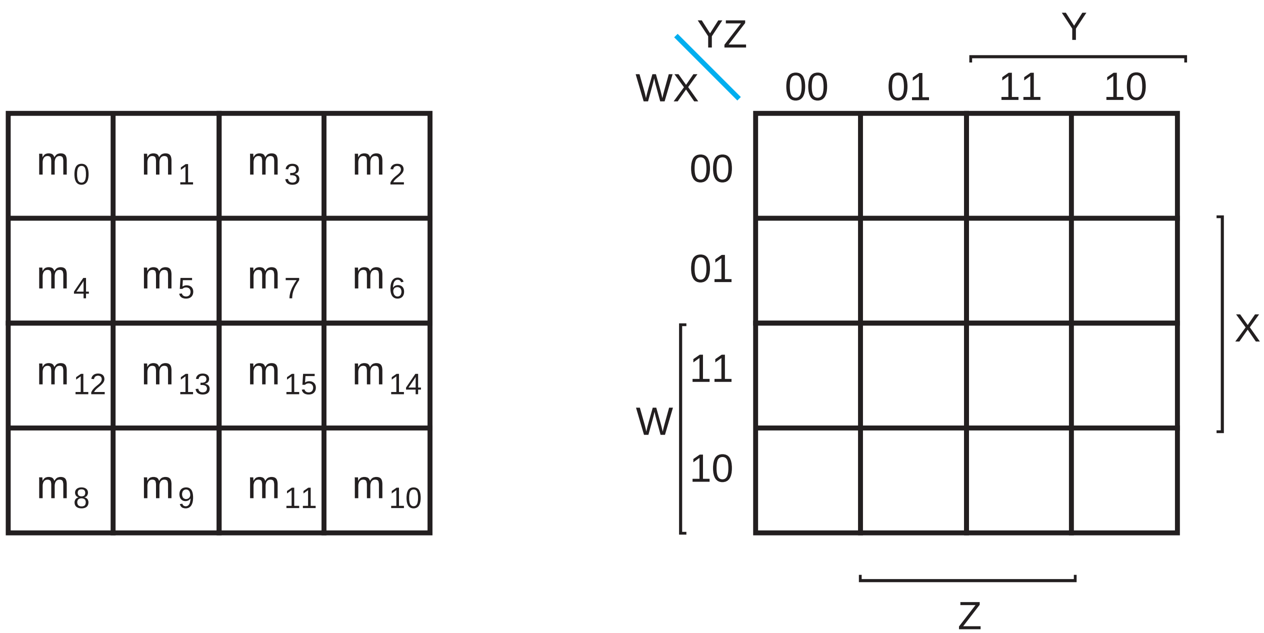
M. Morris Mano & Charles R. Kime

Figure 2-16 F(X, Y, Z) = SUM m (1, 2, 3, 5, 7)



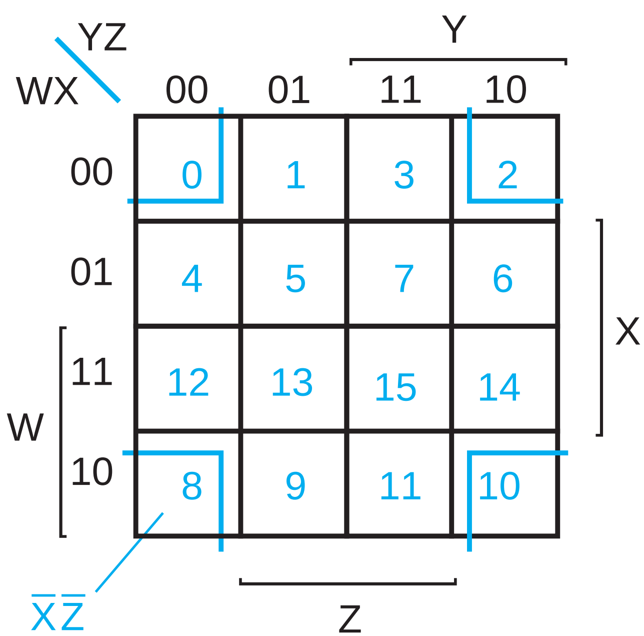
M. Morris Mano & Charles R. Kime
LOGIC AND COMPUTER DESIGN FUNDAMENTALS

Figure 2-17 Four-Variable Map

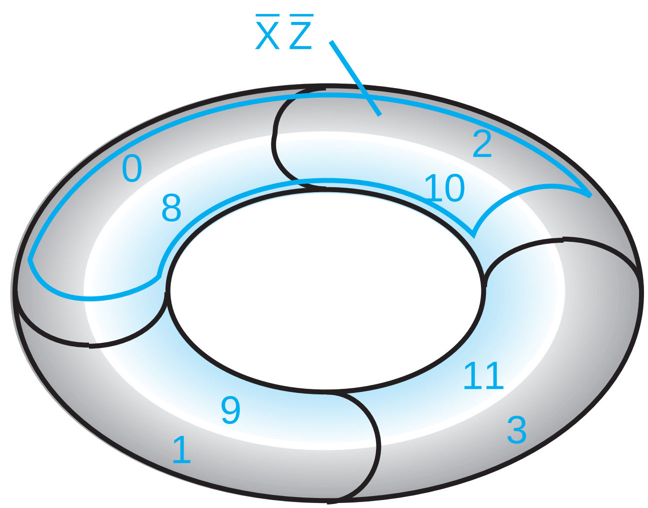


(a) (b)

Figure 2-18 Four-Variable Map: Flat and on a Torus to Show Adjacencies



(a)



(b)

M. Morris Mano & Charles R. Kime
LOGIC AND COMPUTER DESIGN FUNDAMENTALS

Figure 2-19 Map for Example 2-5: F = Y' + W'Z' + XZ'

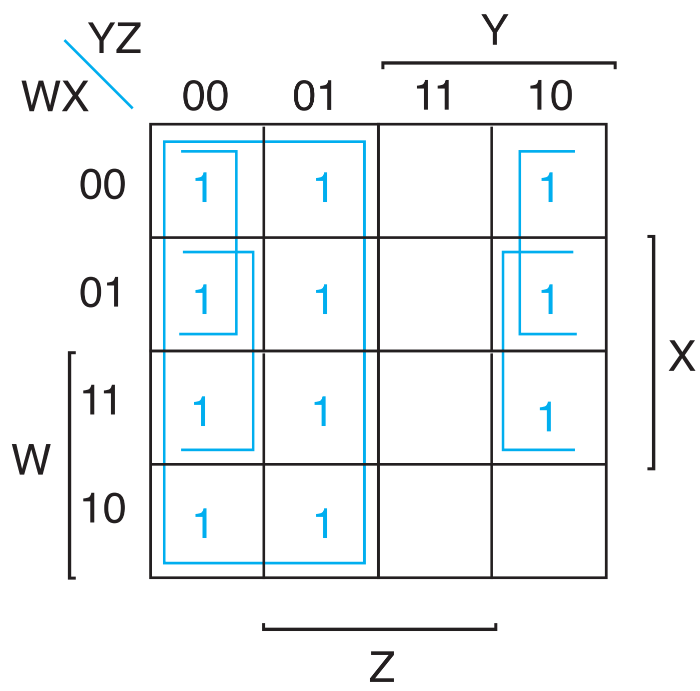
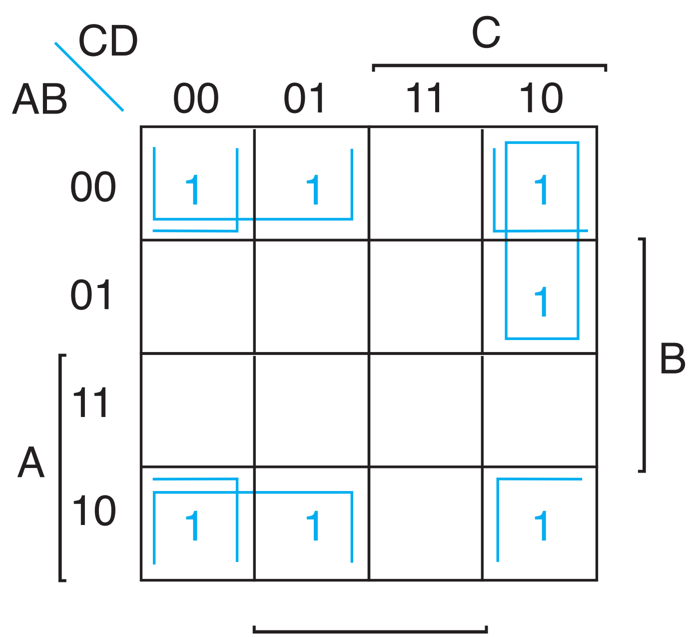
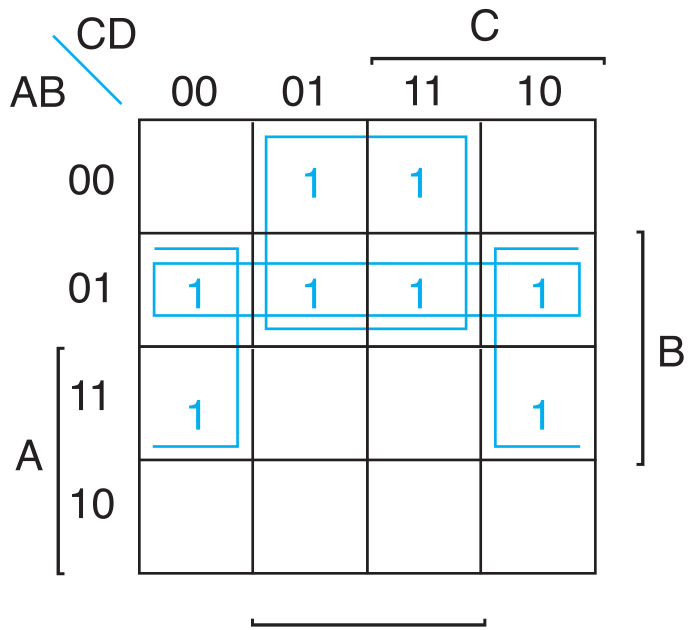


Figure 2-20 Map for Example 2-6: F = B'D' + B'C + A'CD'



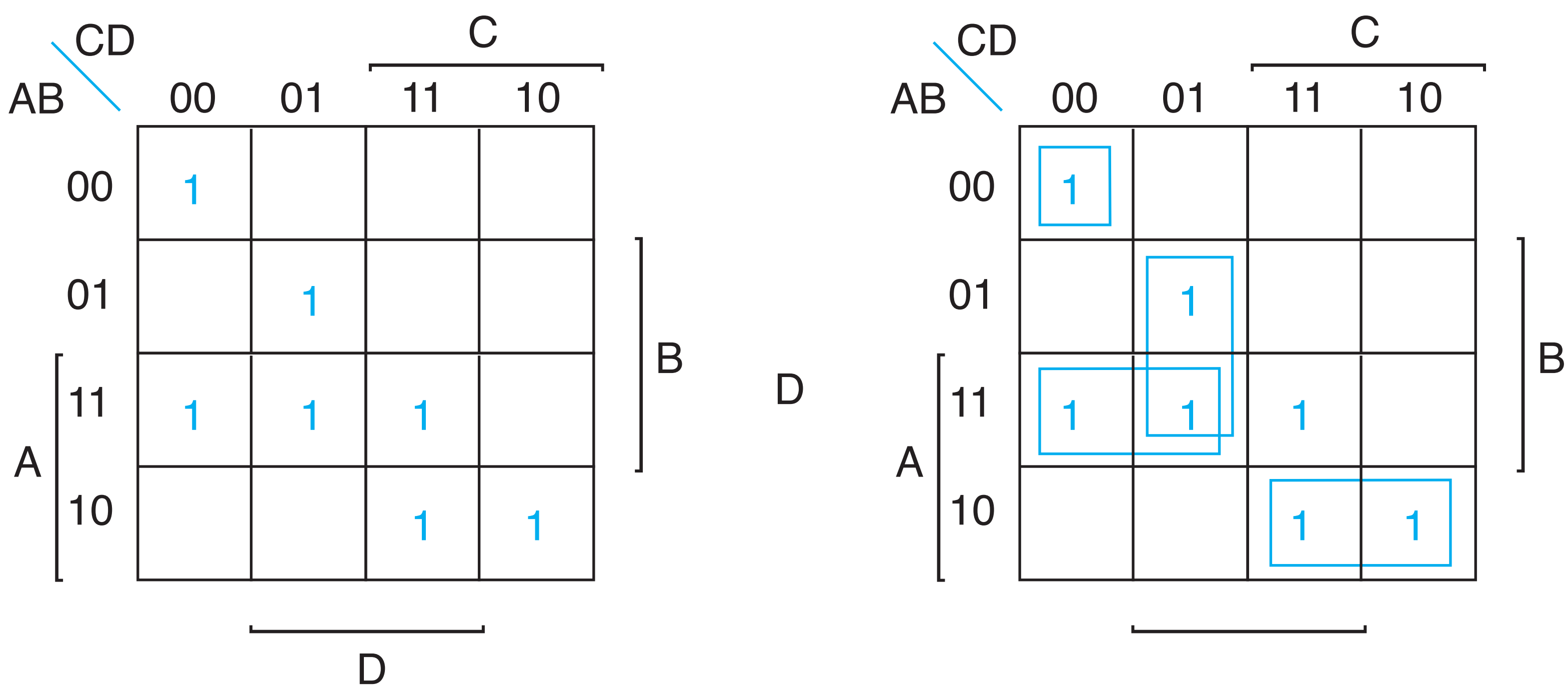
M. Morris Mano & Charles R. Kime
LOGIC AND COMPUTER DESIGN FUNDAMENTALS

Figure 2-21 Prime Implicants for Example 2-7: A'D, BD', and A'B



M. Morris Mano & Charles R. Kime
LOGIC AND COMPUTER DESIGN FUNDAMENTALS

Figure 2-22 Simplification with Prime Implicants in Example 2-8



(a) Plotting the minterms (b) Essential prime implicants

Figure 2-23 Map for Example 2-9

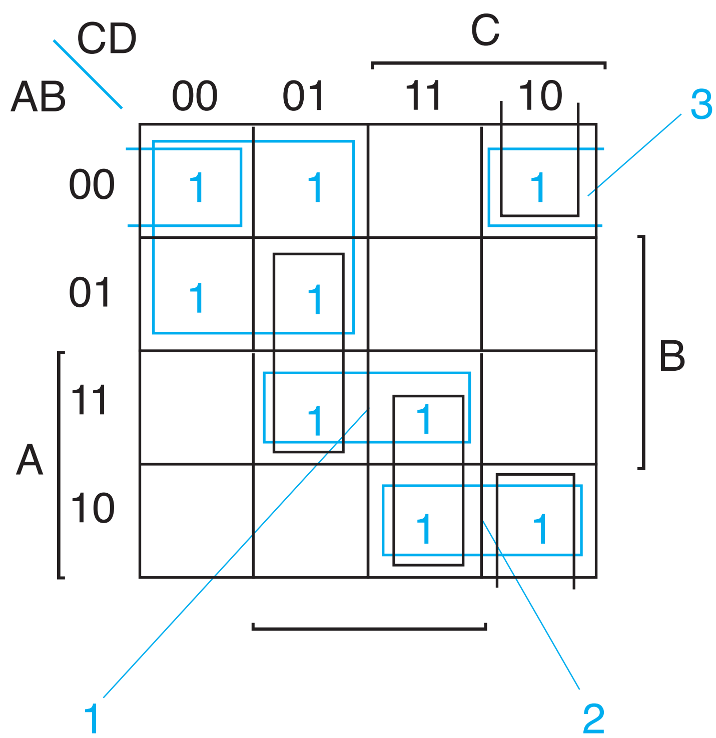


Figure 2-24 Map for Example 2-10: F = (A' + B')(C' + D')(B' + D)

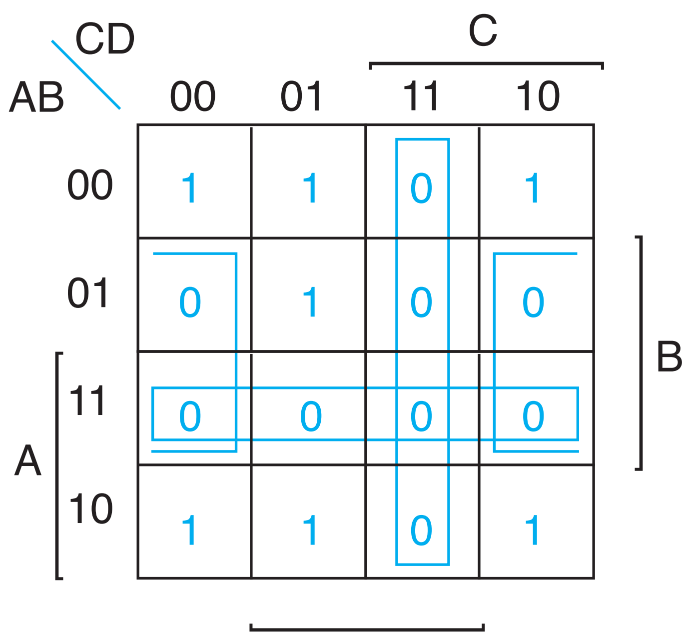
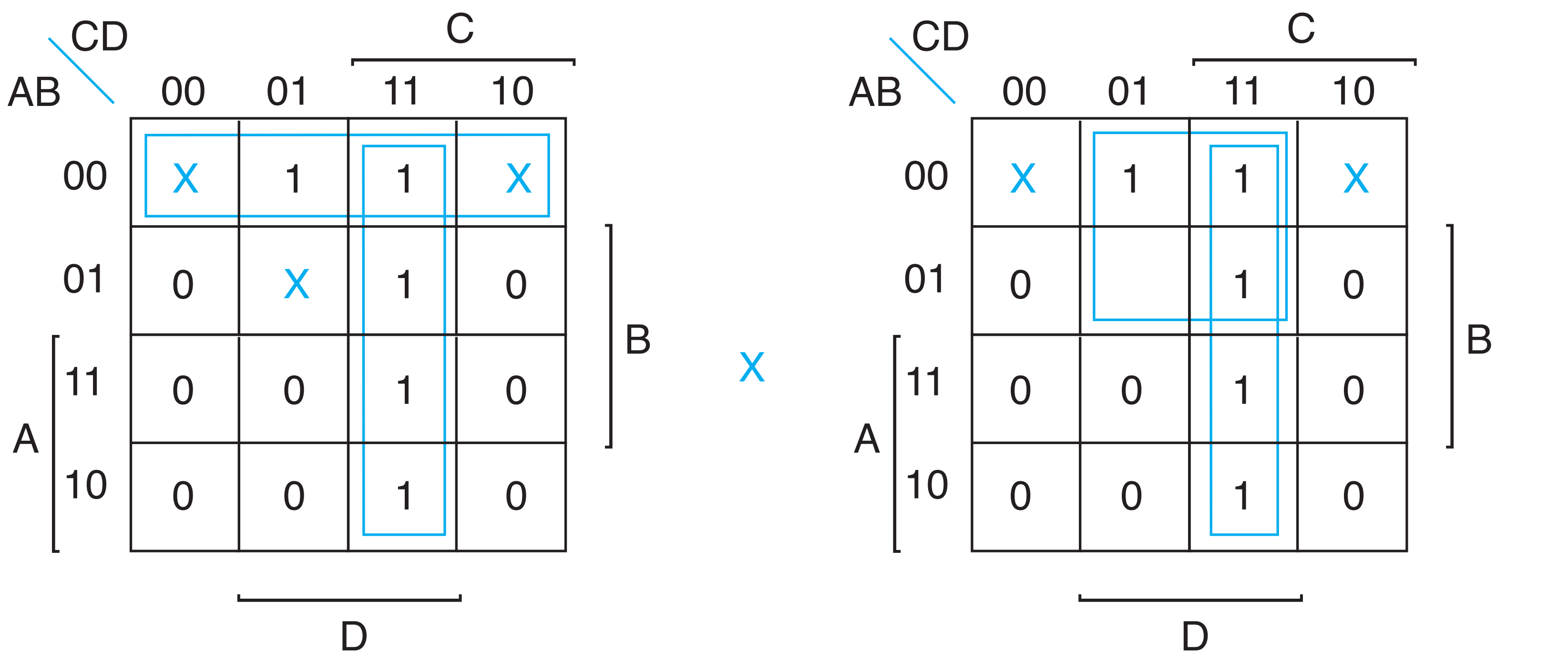


Figure 2-25 Example with Don't Care Conditions

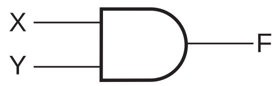
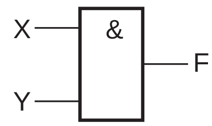
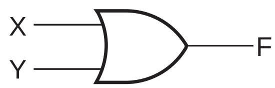
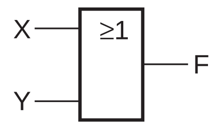
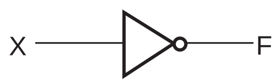
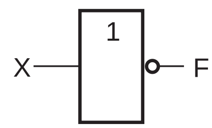
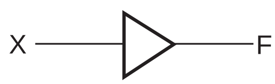
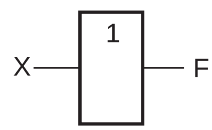
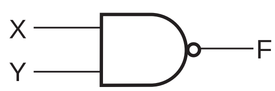
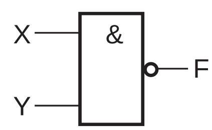
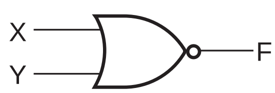
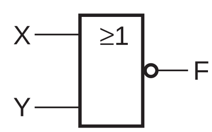
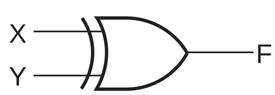
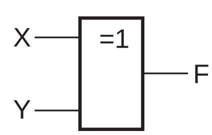
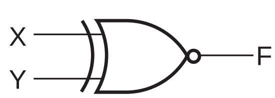
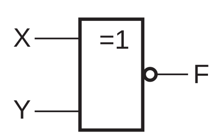


(a) $F = CD + \overline{A}\,\overline{B}$

(b) $F = CD + \overline{A}D$

Figure 2-26 Digital Logic Gates

Graphics Symbols

Name	Distinctive shape	Rectangular shape	Algebraic equation	Truth table
AND			$F = XY$	X Y | F 0 0 | 0 0 1 | 0 1 0 | 0 1 1 | 1
OR			$F = X + Y$	X Y | F 0 0 | 0 0 1 | 1 1 0 | 1 1 1 | 1
NOT (inverter)			$F = \overline{X}$	X | F 0 | 1 1 | 0
Buffer			$F = X$	X | F 0 | 0 1 | 1
NAND			$F = \overline{X \bullet Y}$	X Y | F 0 0 | 1 0 1 | 1 1 0 | 1 1 1 | 0
NOR			$F = \overline{X + Y}$	X Y | F 0 0 | 1 0 1 | 0 1 0 | 0 1 1 | 0
Exclusive–OR (XOR)			$F = X\overline{Y} + \overline{X}Y$ $= X \oplus Y$	X Y | F 0 0 | 0 0 1 | 1 1 0 | 1 1 1 | 0
Exclusive–NOR (XNOR)			$F = XY + \overline{X}\,\overline{Y}$ $= \overline{X \oplus Y}$	X Y | F 0 0 | 1 0 1 | 0 1 0 | 0 1 1 | 1

M. Morris Mano & Charles R. Kime
LOGIC AND COMPUTER DESIGN FUNDAMENTALS

Figure 2-28 Alternative Graphics Symbols for NAND and NOT Gates

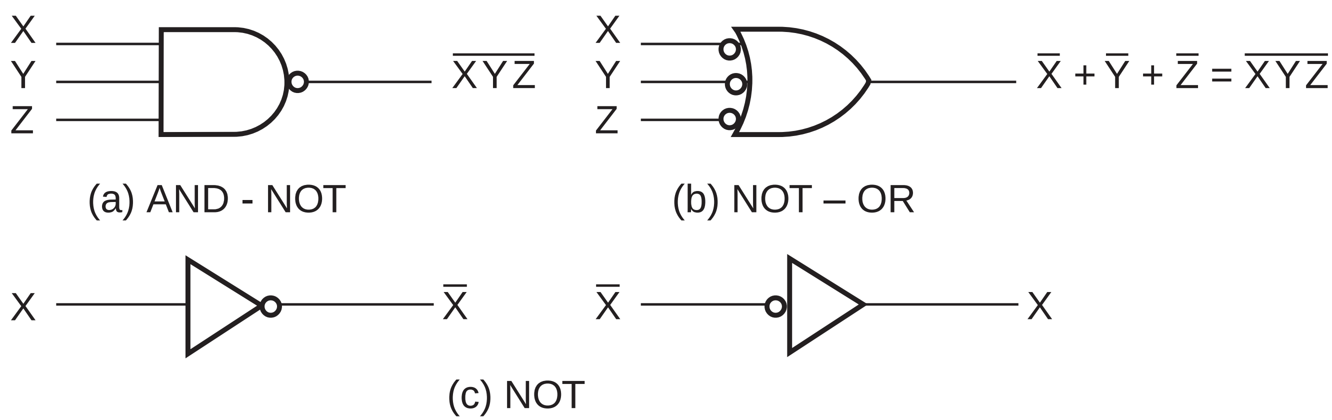


Figure 2-29 Three Ways to Implement F = AB + CD

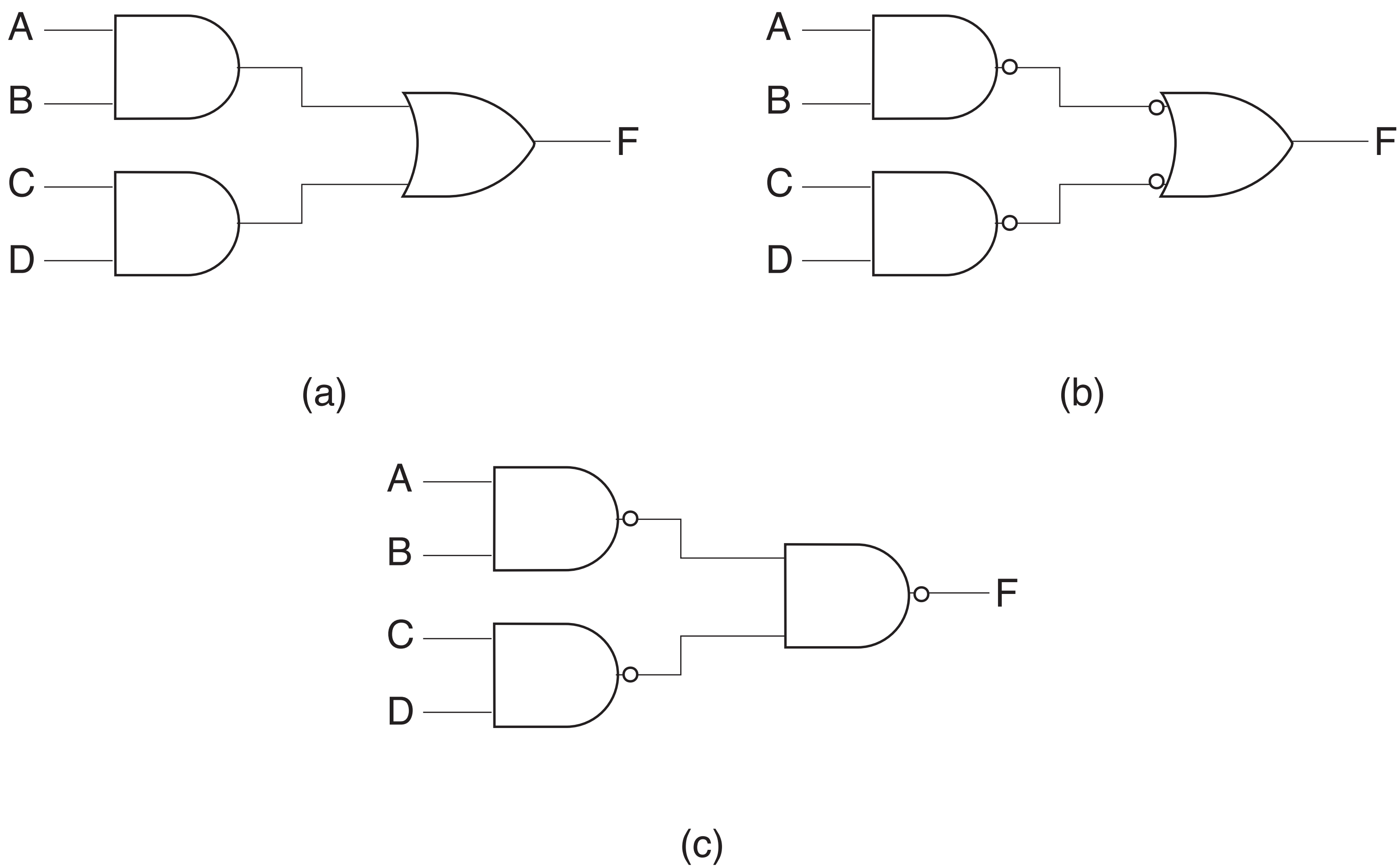


Figure 2-5 Sum-of-Products Implementation

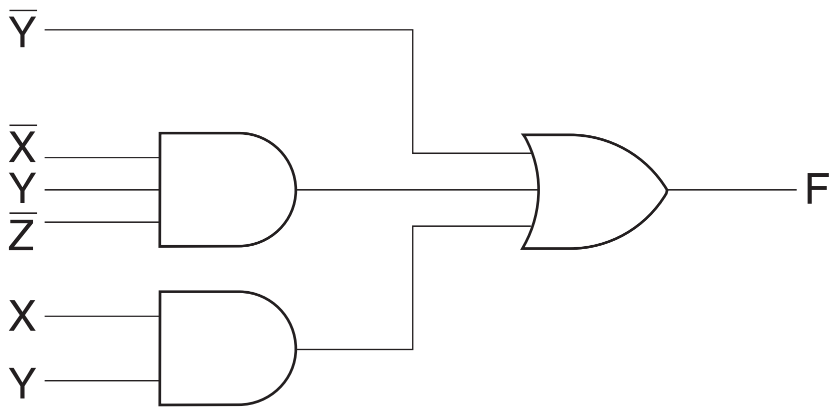
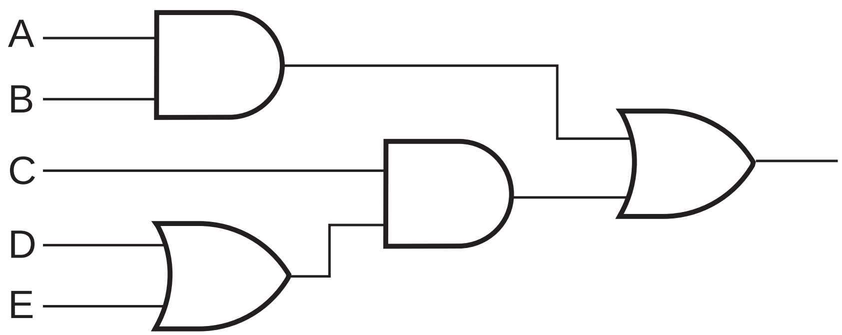
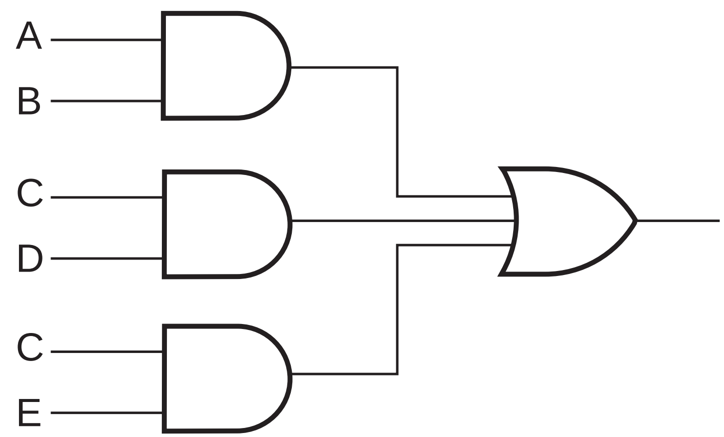


Figure 2-6 Three-Level and Two-Level Implementation



(a) AB + C (D + E)



(b) AB + CD + CE

Figure 2-7 Product-of-Sums Implementation

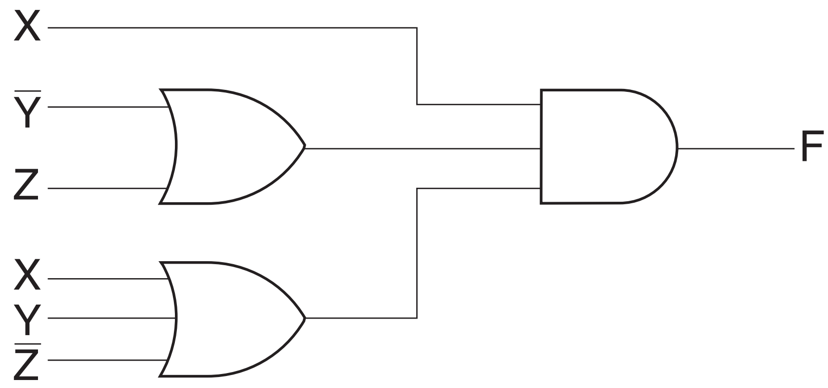
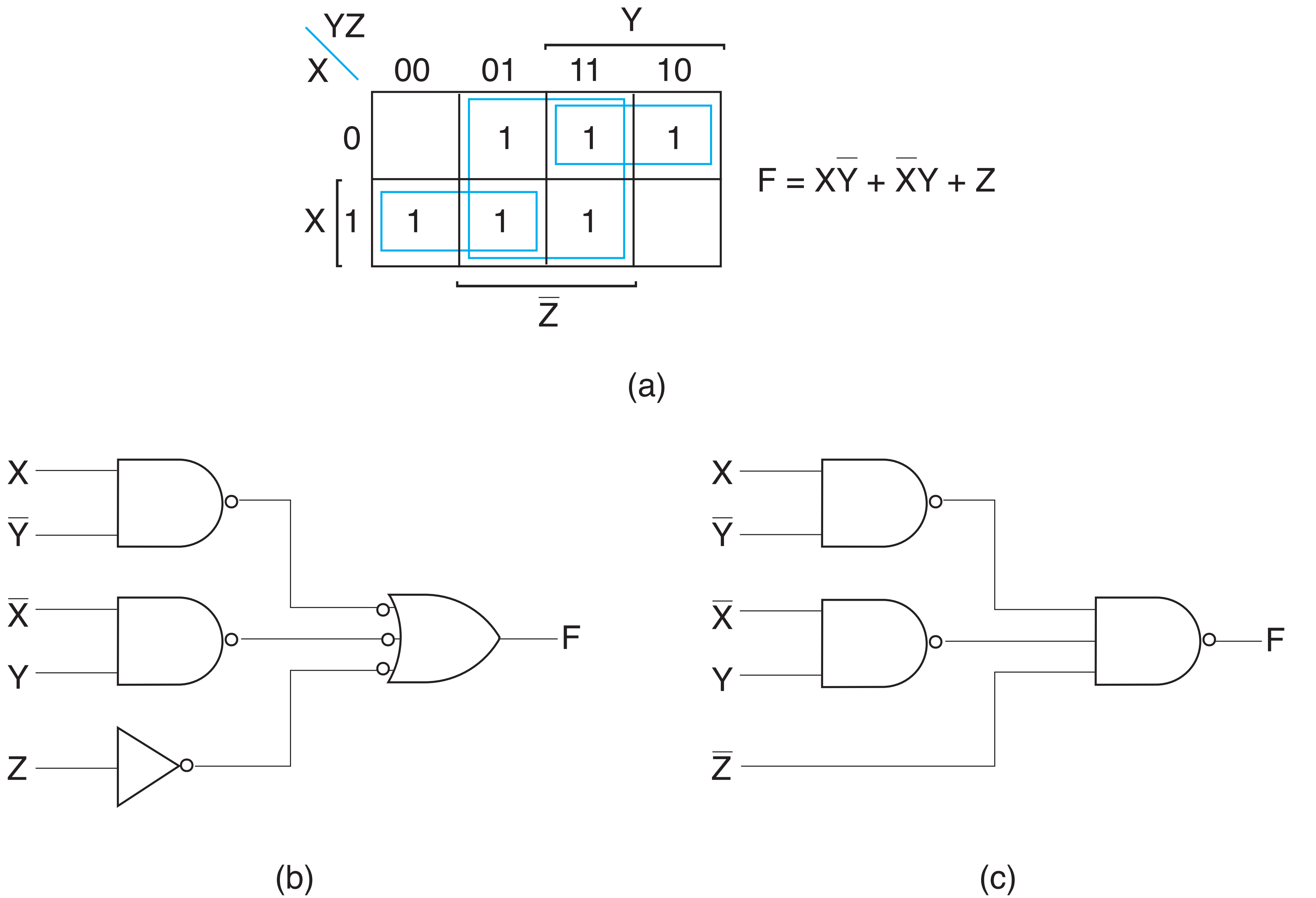
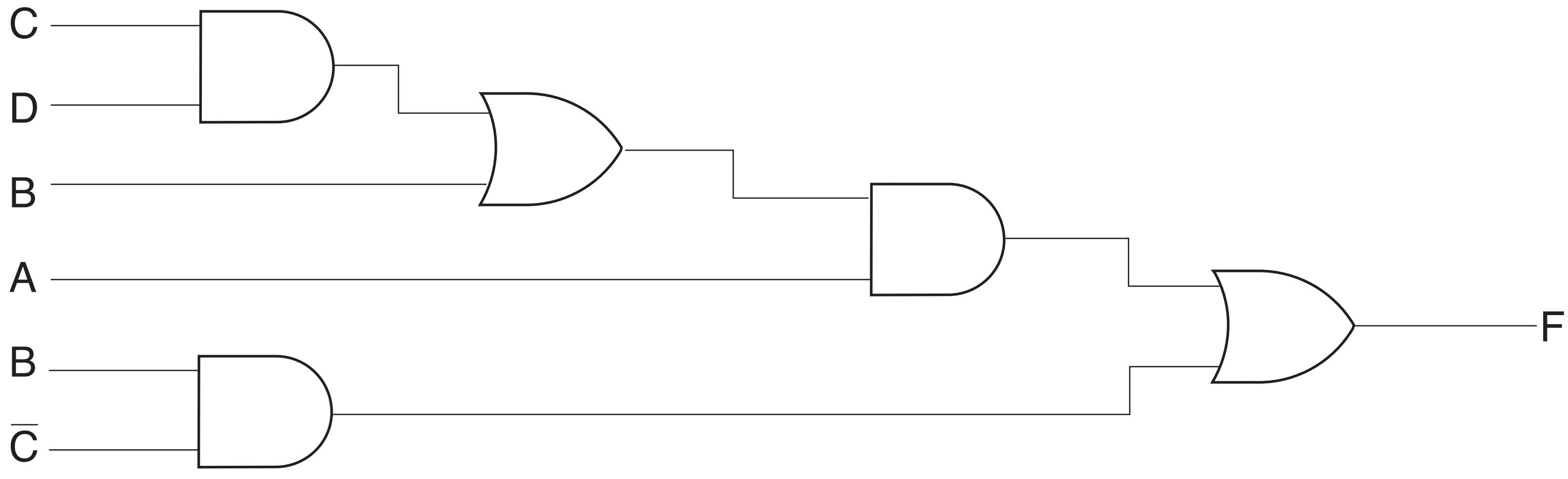


Figure 2-30 Solution to Example 2-12

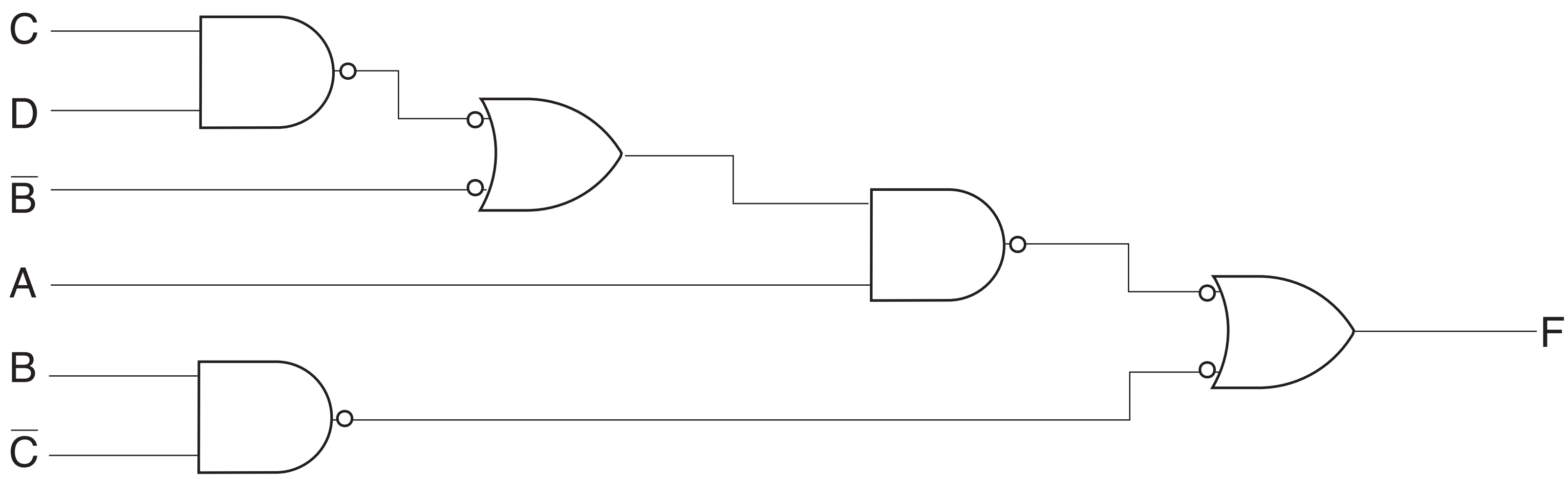


M. Morris Mano & Charles R. Kime
LOGIC AND COMPUTER DESIGN FUNDAMENTALS

Figure 2-31 Implementing F = A(CD + B) + BC'

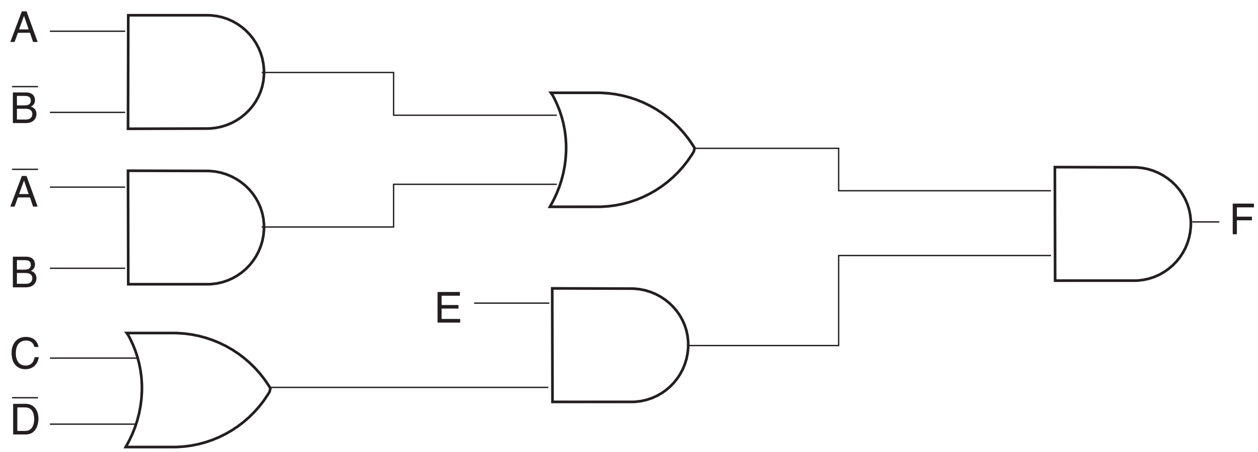


(a) AND – OR gates

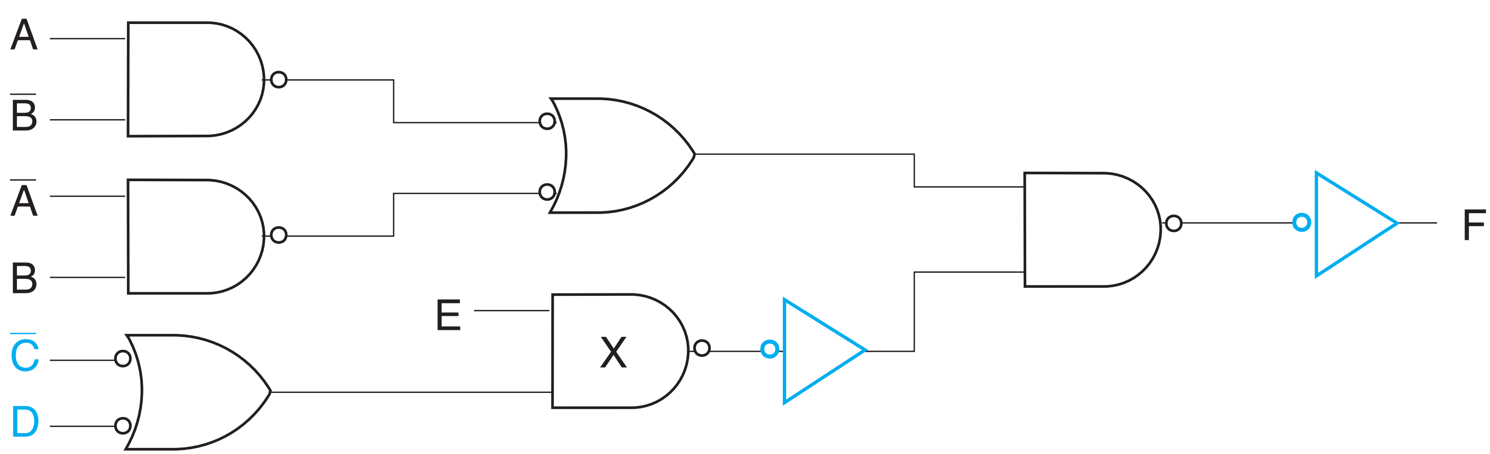


(b) NAND gates

Figure 2-32 Implementing F = (AB' + A'B) E (C + D')



(a) AND – OR gates



(b) NAND gates

Figure 2-34 Two Graphic Symbols for NOR Gate

(a) OR – NOT (b) NOT – AND

Figure 2-35 Implementing F = (A + B)(C + D)E with NOR Gates

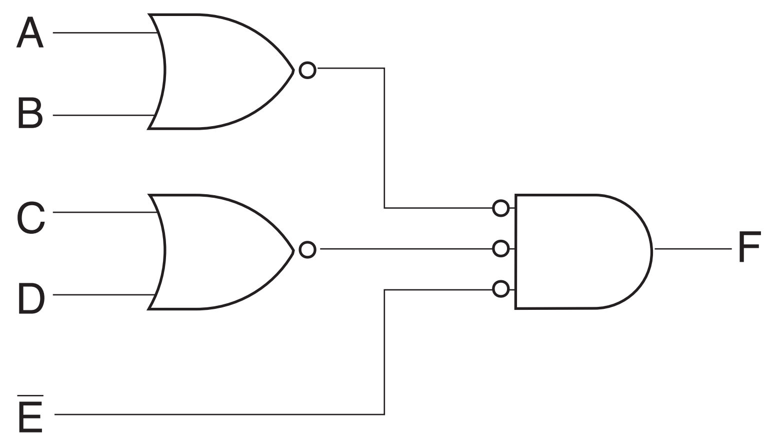


Figure 2-36 Implementing F = (AB' + A'B) E (C + D') with NOR Gates

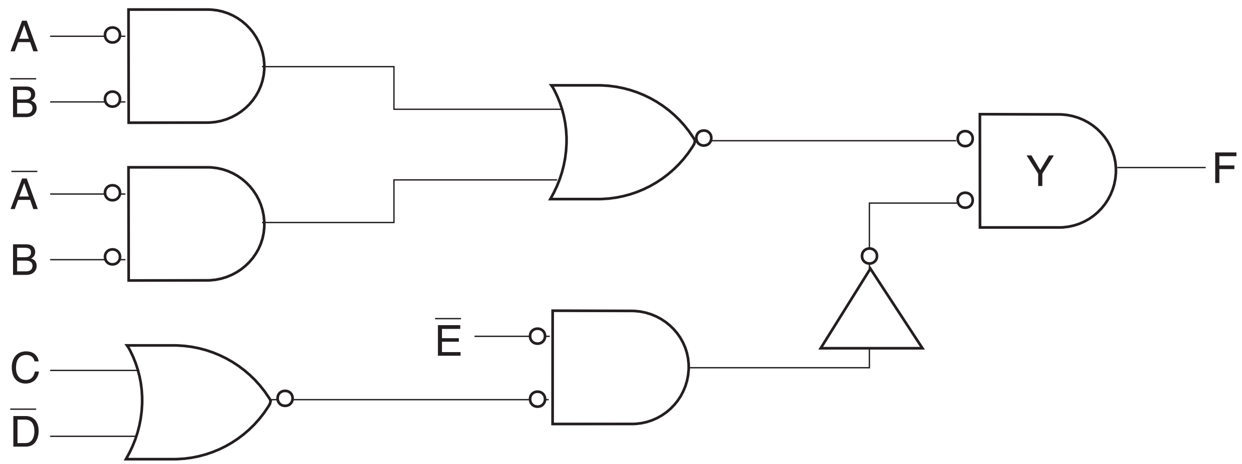


Figure 2-37 Exclusive-OR Constructed with NAND Gates

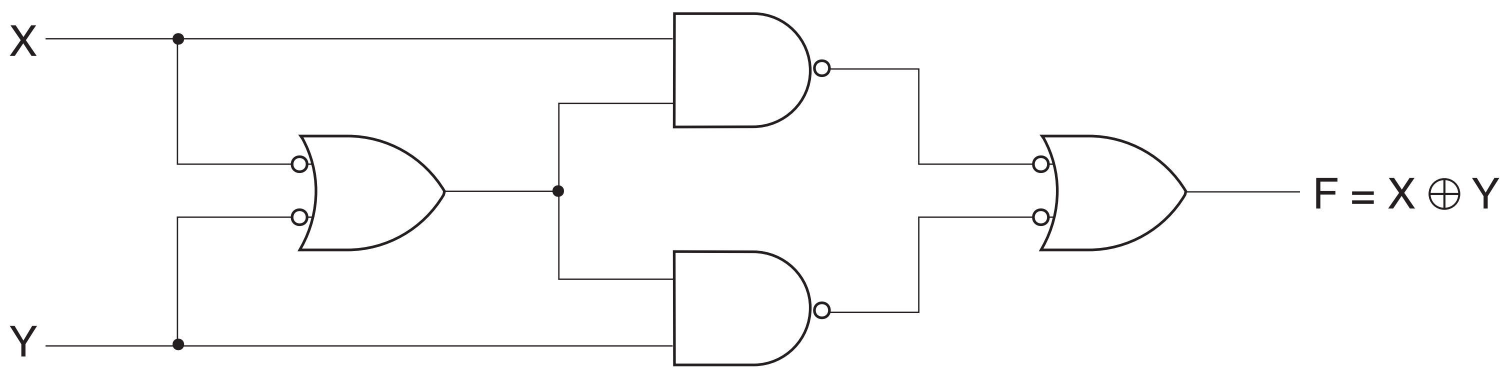
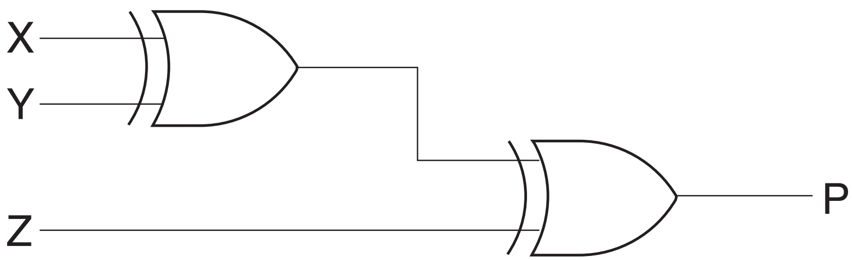
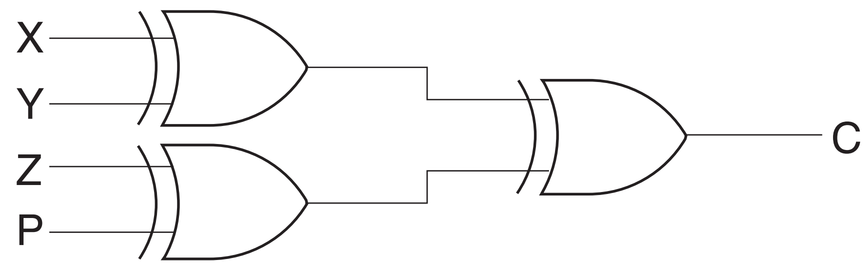


Figure 2-39 Mutiple-Input Odd Functions

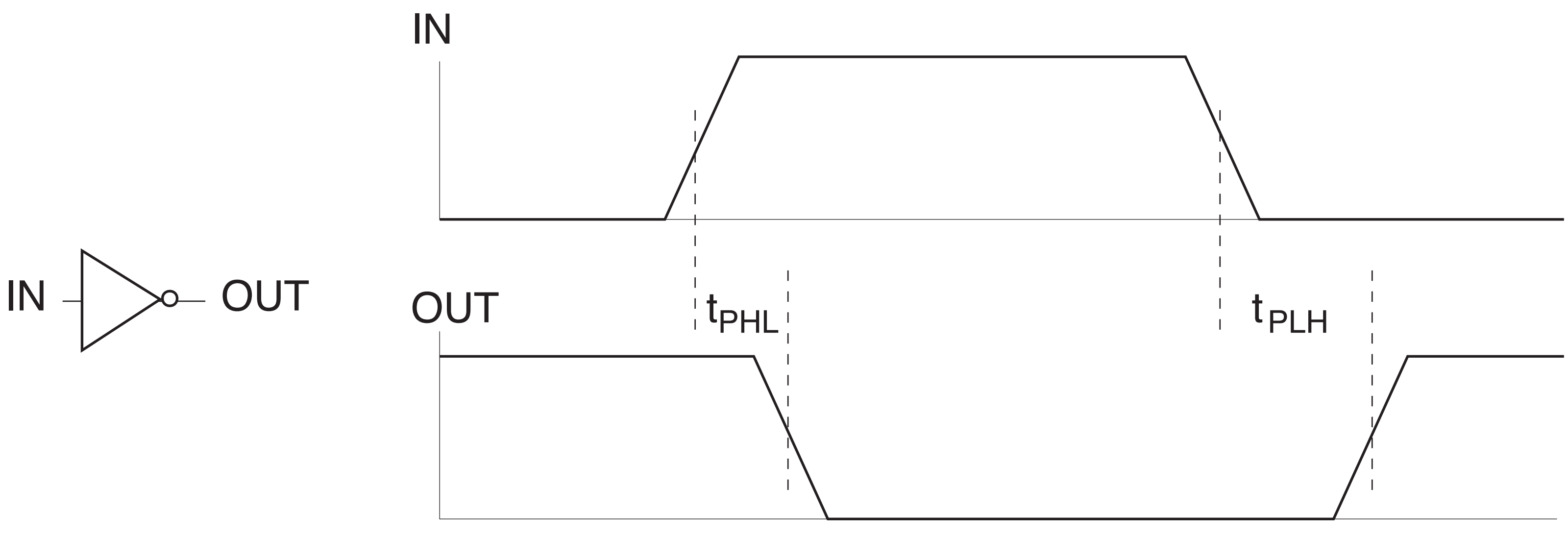


(a) $P = X \oplus Y \oplus Z$



(b) $C = X \oplus Y \oplus Z \oplus P$

Figure 2-40 Propagation Delay for an Inverter



$t_{pd} = \max\,(t_{PHL}, t_{PLH})$

Figure 2-42 Signal Assignment and Logic Polarity

Signal value	Logic value
H	1
L	0

(a) Positive logic

Signal value	Logic value
H	0
L	1

(b) Negative logic