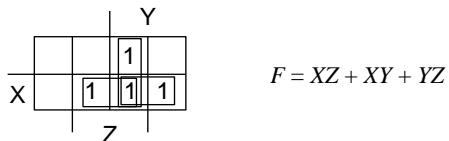

CHAPTER 3

© 2016 Pearson Education, Inc.

3-1.

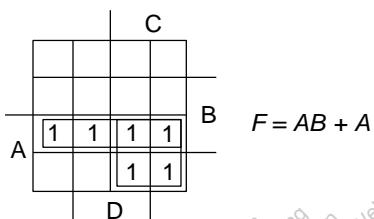
Place a 1 in each K-map cell where 2 or more inputs are equal to 1.



$$F = XZ + XY + YZ$$

This is the same function as the carry for the full adder.

3-2.*



$$F = AB + A$$

3-3.

Assuming inputs G3, G2, G1, G0 and outputs B3, B2, B1, B0, with G3 and B3 being the most significant bits, and treating the invalid input combinations as don't cares:

		G ₃ G ₀	00	01	11	10
		G ₃ G ₂	00	01	11	10
		G ₃ G ₁	X	X	X	X
00	00					
01	01					
11	11	1	1	1	1	
10	10	X	X	X	X	X

		G ₃ G ₀	00	01	11	10
		G ₃ G ₂	00	01	11	10
		G ₃ G ₁	X	X	X	X
00	00					
01	01					
11	11	1	1	1	1	
10	10	X	X	X	X	X

		G ₃ G ₀	00	01	11	10
		G ₃ G ₂	00	01	11	10
		G ₃ G ₁	X	X	X	X
00	00					
01	01					
11	11					
10	10	X	X	X	X	X

		G ₃ G ₀	00	01	11	10
		G ₃ G ₂	00	01	11	10
		G ₃ G ₁	X	X	X	X
00	00					
01	01					
11	11					
10	10	X	X	X	X	X

$$B_3 = G_3$$

$$B_2 = \overline{G_3}G_2$$

$$B_1 = \overline{G_2}G_1 + \overline{G_3}G_2\overline{G_1}$$

$$B_0 = G_3G_0 + G_2G_1G_0 + \overline{G_2}G_1\overline{G_0}$$

$$+ \overline{G_2}\overline{G_1}G_0 + \overline{G_3}G_2\overline{G_1}\overline{G_0}$$

3-4. a) For the 3 x 3 pattern, there are exactly three row, three column and two diagonal combinations that represent a win for the X player: $W = X_1 X_2 X_3 + X_4 X_5 X_6 + X_7 X_8 X_9 + X_1 X_4 X_7 + X_2 X_5 X_8 + X_3 X_6 X_9 + X_1 X_5 X_9 + X_3 X_5 X_7$ Gate Input cost = 32

b) $W = X_5 (X_1 X_9 + X_2 X_8 + X_3 X_7 + X_4 X_6) + X_1 X_2 X_3 + X_1 X_4 X_7 + X_7 X_8 X_9 + X_3 X_6 X_9$ Gate Input Cost = 30

3-5. a) For the 4 x 4 pattern, there are exactly four row, four column and two diagonal combinations that represent a win for the X player: $W = X_1 X_2 X_3 X_4 + X_5 X_6 X_7 X_8 + X_9 X_{10} X_{11} X_{12} + X_{13} X_{14} X_{15} X_{16} + X_1 X_5 X_9 X_{13} X_2 X_6 X_{10} X_{14} + X_3 X_7 X_{11} X_{15} + X_4 X_8 X_{12} X_{16} + X_1 X_6 X_{11} X_{16} + X_4 X_7 X_{10} X_{13}$ Gate Input cost = 50

b) $W = X_1(X_2 X_3 X_4 + X_5 X_9 X_{13} + X_6 X_{11} X_{15}) + X_7(X_5 X_6 X_8 + X_3 X_{11} X_{15} + X_4 X_{10} X_{13}) + X_9 X_{10} X_{11} X_{12}$
+ $X_{13} X_{14} X_{15} X_{16} + X_2 X_6 X_{10} X_{14} + X_4 X_8 X_{12} X_{16}$ Gate Input Cost = 48

3-6.

a) Detecting a change in one-out-of-three inputs can be done using a parity function as Z. The truth table shown is for even parity. For this case,
 $Z = X_1 \oplus X_2 \oplus X_3$

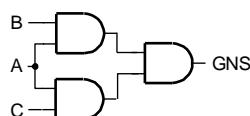
If odd parity is chosen, then an alternative result for Z is:
 $Z = \overline{X_1 \oplus X_2 \oplus X_3}$

X1	X2	X3	Z
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

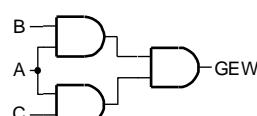
 3-7.⁺

ABCD	GNS	YNS	RNS	GEW	YEW	REW
0000	1	0	0	0	0	1
0001	1	0	0	0	0	1
0011	1	0	0	0	0	1
0010	1	0	0	0	0	1
0110	1	0	0	0	0	1
0111	1	0	0	0	0	1
0101	0	1	0	0	0	1
0100	0	0	1	0	0	1
1100	0	0	1	1	0	0
1101	0	0	1	1	0	0
1111	0	0	1	1	0	0
1110	0	0	1	1	0	0
1010	0	0	1	1	0	0
1011	0	0	1	1	0	0
1001	0	0	1	0	0	0
1000	0	0	1	0	0	1

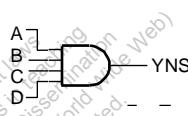
This work is protected by United States copyright law
 and is provided solely for the use of instructors in teaching their courses and assessing student learning.
 Dissemination or sale of any part of this work (including on the World Wide Web)
 in whole or in part without the express written permission of Pearson Education, Inc.
 is illegal.



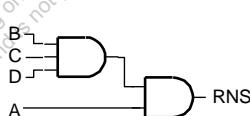
$$GNS = \bar{A}C + \bar{A}\bar{B}$$



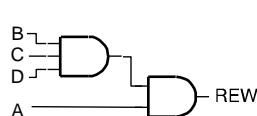
$$GEW = AB + AC$$



$$YEW = A\bar{B}\bar{C}\bar{D}$$



$$RNS = A + B\bar{C}\bar{D}$$



$$REW = \bar{A} + \bar{B}\bar{C}\bar{D}$$

3-8.

A	B	C	S5	S4	S3	S2	S1	S0	S0 = C
0	0	0	0	0	0	0	0	0	S1 = 0
0	0	1	0	0	0	0	0	1	$S2 = \bar{A}B\bar{C} + AB\bar{C}$
0	1	0	0	0	0	1	0	0	$S3 = \bar{A}BC + A\bar{B}C$
0	1	1	0	0	1	0	0	1	$S4 = A\bar{B} + AC$
1	0	0	0	1	0	0	0	0	$S5 = AB$
1	0	1	0	1	1	0	0	1	
1	1	0	1	0	0	1	0	0	
1	1	1	1	1	0	0	0	1	

3-9.⁺

A	B	C	D	S2	S1	S0	
0	0	0	0	0	0	0	$S_0 = \bar{B}\bar{C}D + \bar{B}C\bar{D} + A\bar{B} + A\bar{C}\bar{D} + \bar{A}BCD$
0	0	0	1	0	0	1	$S_1 = \bar{A}\bar{B} + A\bar{B} + \bar{A}CD + B\bar{C}\bar{D}$
0	0	1	0	0	0	1	$S_2 = ABC + ABD$
0	0	1	1	0	1	0	
0	1	0	0	0	1	0	
0	1	0	1	0	1	0	
0	1	1	0	0	1	0	
0	1	1	1	0	1	1	
1	0	0	0	0	1	1	
1	0	0	1	0	1	1	
1	0	1	0	0	1	1	
1	0	1	1	0	1	1	
1	1	0	0	0	1	1	
1	1	0	1	1	0	0	
1	1	1	0	1	0	0	
1	1	1	1	1	0	0	

3-10.

A	B	C	D	W	X	Y	Z	
0	0	0	0	0	0	1	1	$W = A\bar{C} + BD + BD$
0	0	0	1	0	1	0	0	$X = B\bar{C}\bar{D} + \bar{B}C+\bar{B}D$
0	0	1	0	0	1	0	1	$Y = \bar{C}\bar{D} + CD$
0	0	1	1	0	1	1	0	$Z = \bar{D}$
0	1	0	0	0	1	1	1	
0	1	0	1	1	0	0	0	
0	1	1	0	0	1	0	1	
0	1	1	1	1	0	1	0	
1	0	0	0	0	1	0	1	
1	0	0	1	1	0	1	1	
1	0	1	0	1	1	0	0	
1	0	1	1	1	1	0	0	
				1010 to				
				1111				
					XXXX			

3-11.

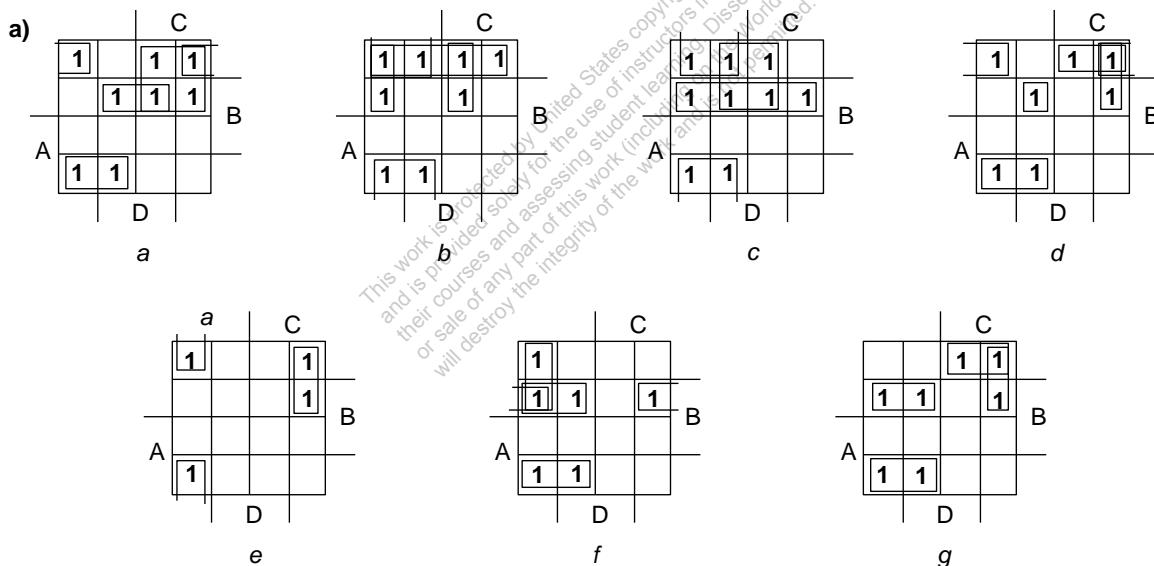
a)

PS	LS	RS	RR	PL	LL	RL
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	0	1	0	0	0	1
0	0	1	1	0	0	1
0	1	0	0	0	1	0
0	1	0	1	0	1	0
0	1	1	0	0	0	1
0	1	1	1	0	1	0
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	1	0	0
1	0	1	1	1	0	0
1	1	0	0	1	0	0
1	1	0	1	1	0	0
1	1	1	0	1	0	0
1	1	1	1	1	0	0

b)

PL = PS
 LL = $\overline{PS} LS \overline{RS} + \overline{PS} LS RR$
 RL = $\overline{PS} LS RS + \overline{PS} RS \overline{RR}$

3-12.



b)

$$a = \bar{A}\bar{C} + \bar{A}\bar{B}\bar{D} + \bar{A}B\bar{D} + A\bar{B}\bar{C}$$

$$b = \bar{A}\bar{B} + \bar{B}\bar{C} + \bar{A}\bar{C}\bar{D} + \bar{A}\bar{C}D$$

$$c = \bar{A}\bar{B} + \bar{B}\bar{C} + \bar{A}\bar{D}$$

$$d = \bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}\bar{D} + \bar{A}\bar{B}C + \bar{A}\bar{C}\bar{D}$$

$$e = \bar{B}\bar{C}\bar{D} + \bar{A}\bar{C}\bar{D}$$

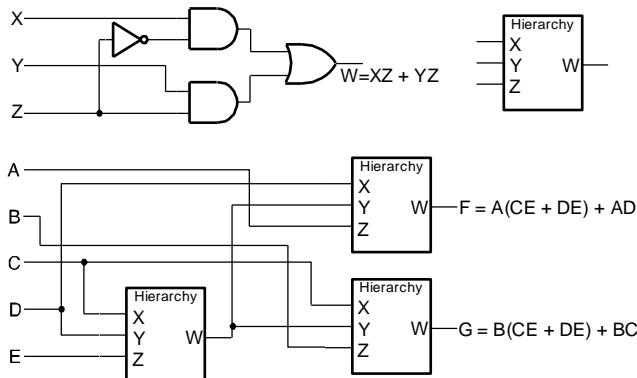
$$f = A\bar{B}\bar{C} + \bar{A}\bar{B}\bar{D} + \bar{A}\bar{B}C + \bar{A}\bar{C}\bar{D}$$

$$g = A\bar{B}\bar{C} + \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + \bar{A}\bar{C}\bar{D}$$

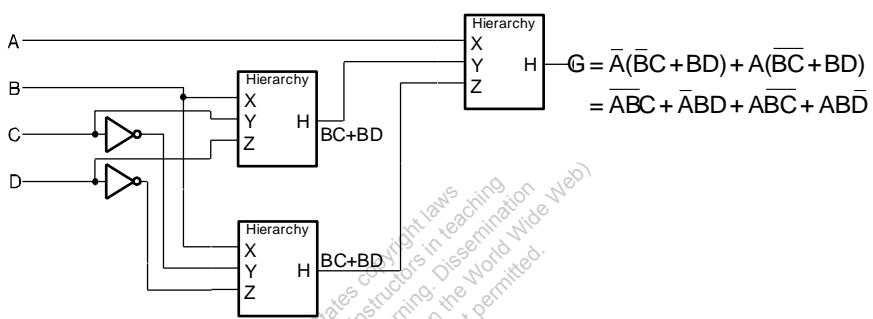
c) The following gate input counts include input inverters and share AND gates.

Total gate inputs for this solution = 74. Total gate inputs for book solution is 70. The book solution is better by 4 gate inputs.

3-13.

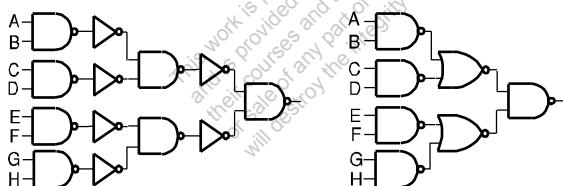


3-14.



3-15.⁺

a)

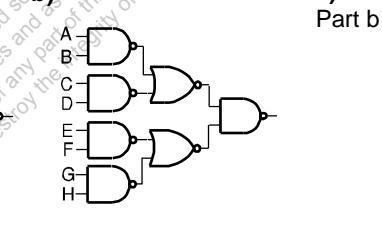


b)

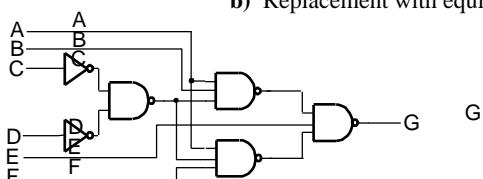
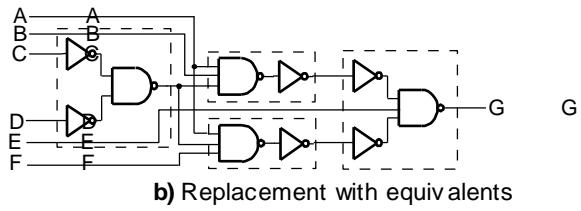
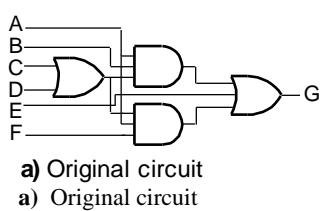
*This work is protected by United States copyright laws
and is intended solely for the use of instructors in teaching
their courses and assessing student learning. Dissemination
of any part of this work (including on the World Wide Web)
will destroy the integrity of the work and is not permitted.*

c)

Part b requires 6 fewer gates.

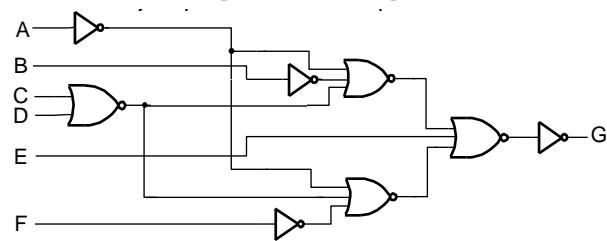
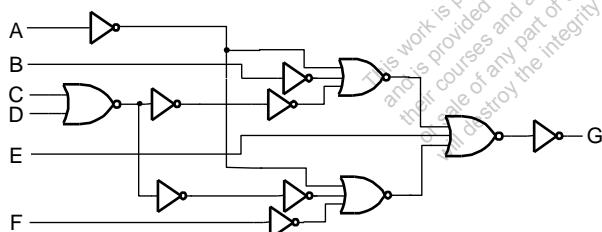
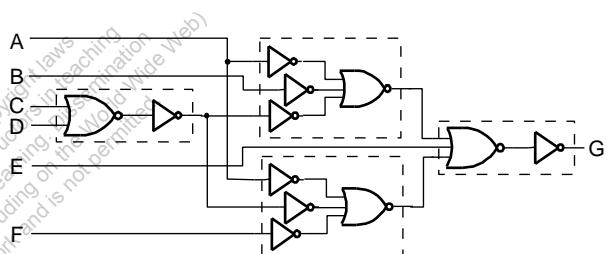
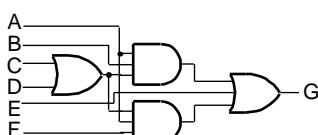


3-16.



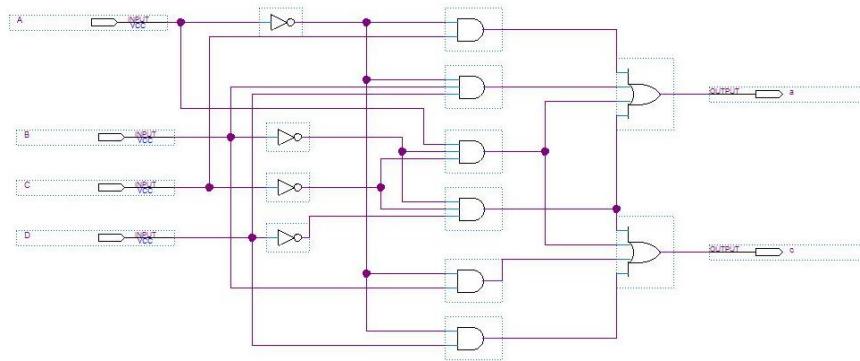
c) Cancel inverters

3-17.

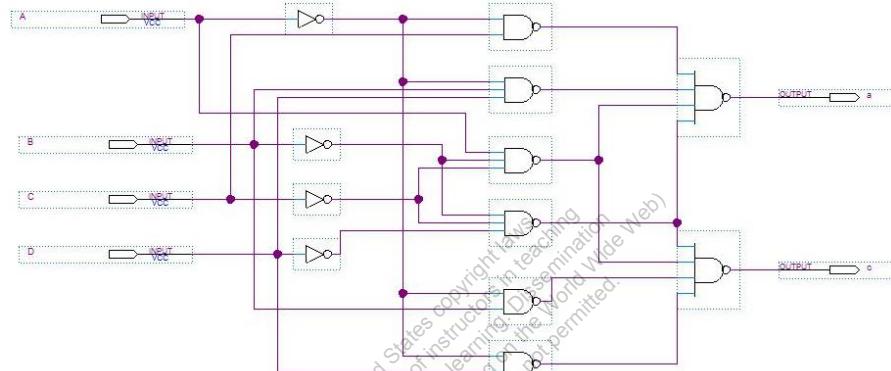


3-18.

a) Using Inverter, 2NAND, 3NAND and 4NAND gates.

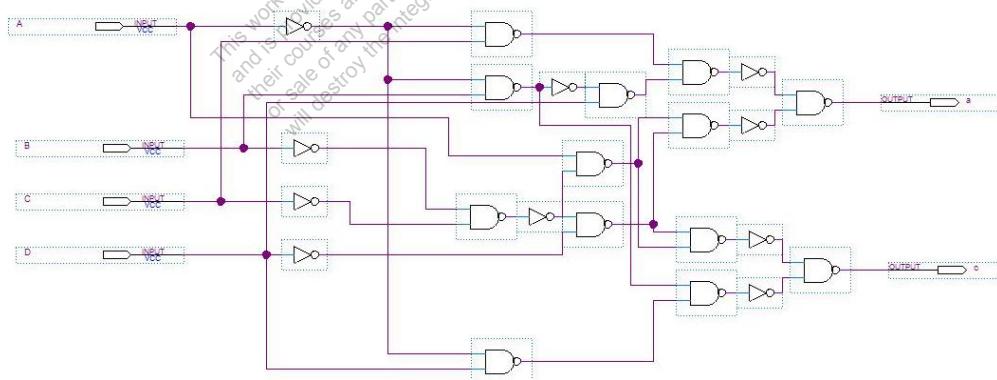


1) Original circuit using AND, OR, Inverter



2) Mapped to Inverter, 2NAND, 3NAND, and 4NAND

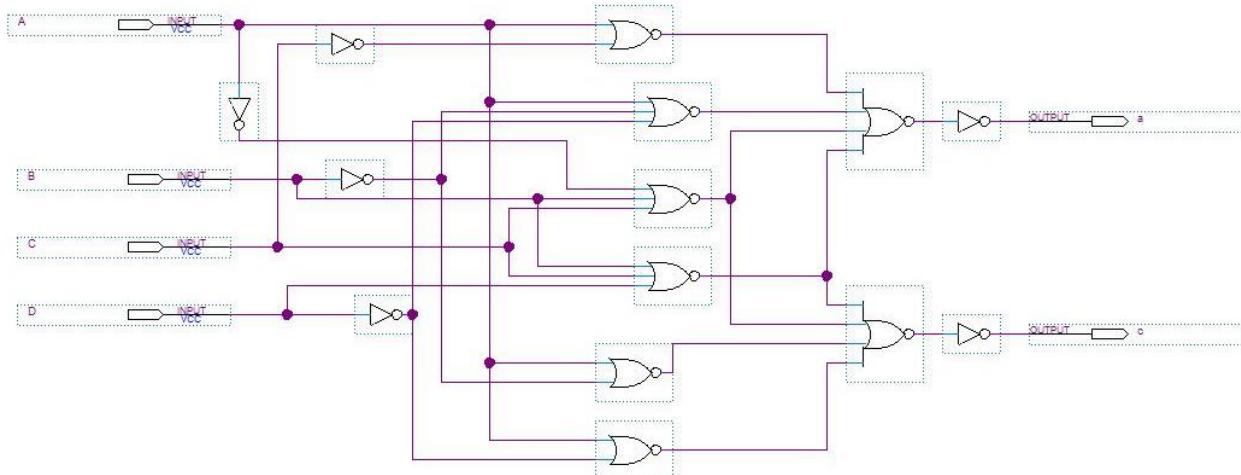
b) Using Inverter and 2NAND gates. There is not a one to one correspondence to the above schematics because common terms with only two literals become available (\overline{AB} , \overline{BC})



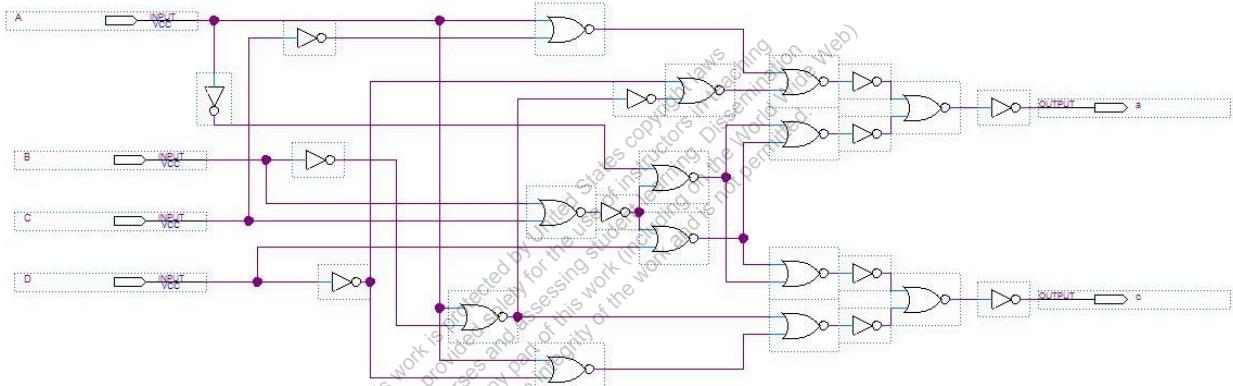
3-19.

For original circuit, see 3-18 part (a) above.

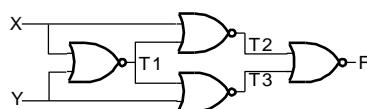
a) Mapped to Inverter, 2NOR, 3NOR, and 4NOR



b) Mapped to 2NOR and Inverter



This work is protected by United States copyright laws and is provided for the use of individual students and institutions for their courses and assessments only. Any other use, including sale or distribution of this material, without the author's consent, is illegal.

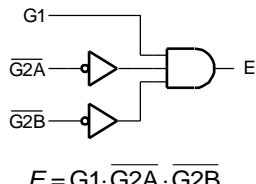
3-20.


$$T1 = \bar{X}\bar{Y}$$

$$T2 = \bar{X}Y$$

$$T3 = X\bar{Y}$$

$$F = XY + \bar{X}Y$$

3-21.


$$Y0 = \overline{\overline{A}\overline{B}\overline{C}E}$$

$$Y1 = \overline{A}\overline{B}\overline{C}E$$

$$Y2 = \overline{A}\overline{B}\overline{C}E$$

$$Y3 = \overline{A}\overline{B}CE$$

$$Y4 = \overline{A}\overline{B}CE$$

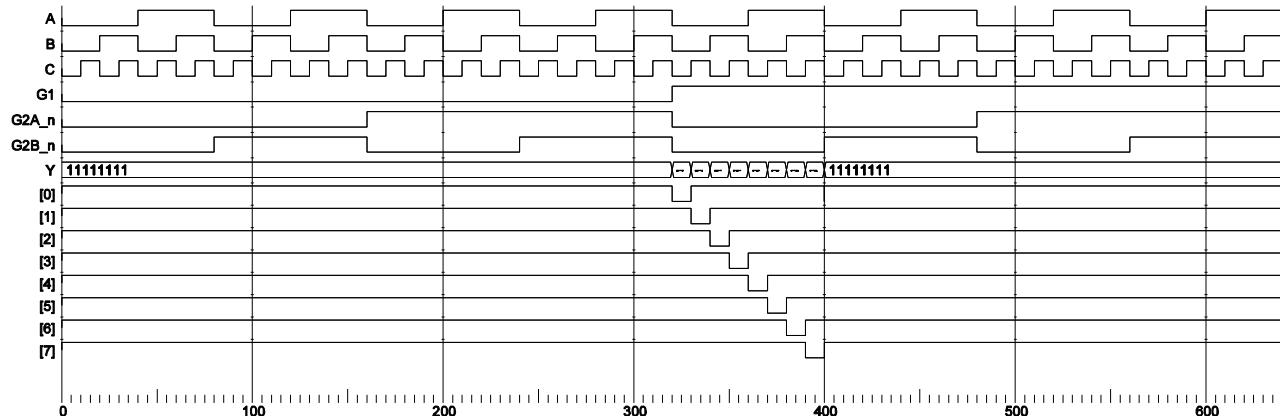
$$Y5 = A\overline{B}CE$$

$$Y6 = \overline{A}BCE$$

$$Y7 = ABC\bar{E}$$

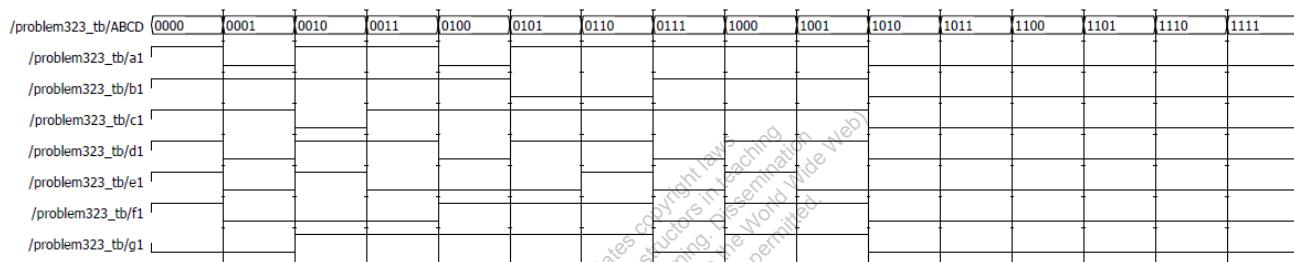
Except for $G1 = 1$ and $G2A$ and $G2B = 0$, the outputs $Y0$ through $Y7$ are all 1's. Otherwise, one of $Y0$ through $Y7$ is equal to 0 with all others equal to 1. The output that is equal to 0 has index i = decimal value of the values of (A,B,C) in binary. E.g., if $(A,B,C) = (1,1,0)$, then $Y6 = 0$.

3-22.

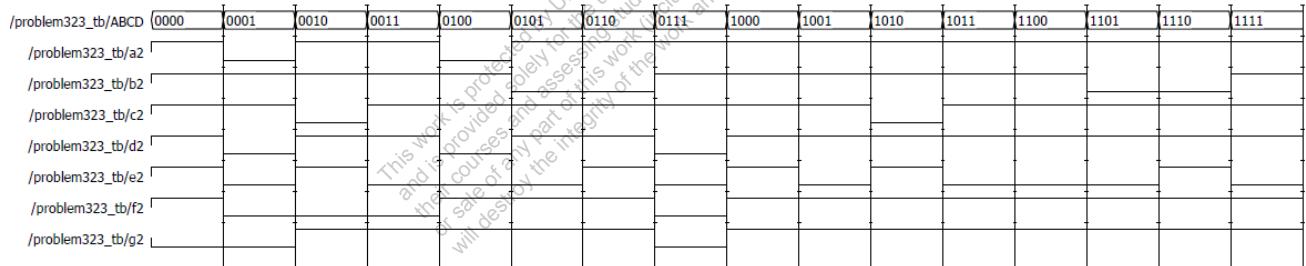


3-23.

a)

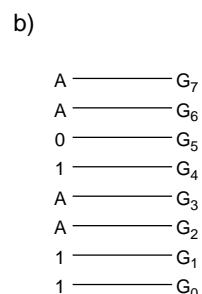
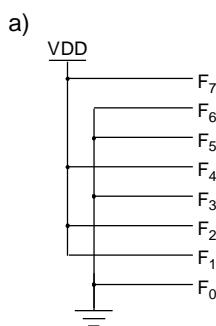


b) Treating the input values 1010-1111 as don't cares changes the behavior for those values:

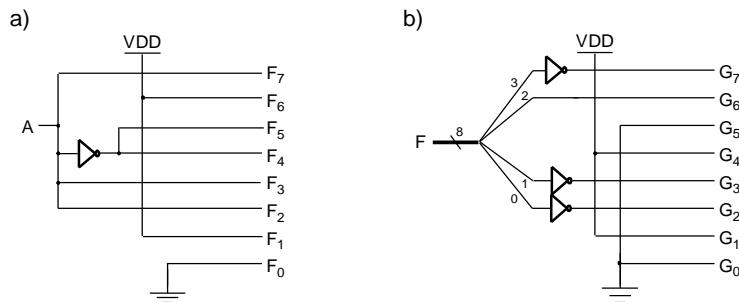


The equations in this case are: $a = A + C + \overline{BD} + BD, b = \overline{B} + \overline{CD} + CD, c = B + \overline{C} + D, d = A + \overline{CD} + \overline{BC} + \overline{BD} + \overline{BCD}, e = \overline{BD} + \overline{CD}, f = A + \overline{CD} + \overline{BC} + \overline{BD}, g = A + \overline{BC} + \overline{BC} + \overline{CD}$

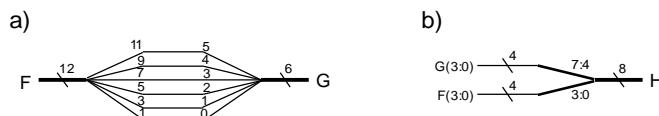
3-24. *



3-25.

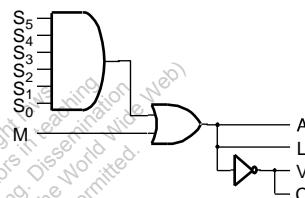


3-26.

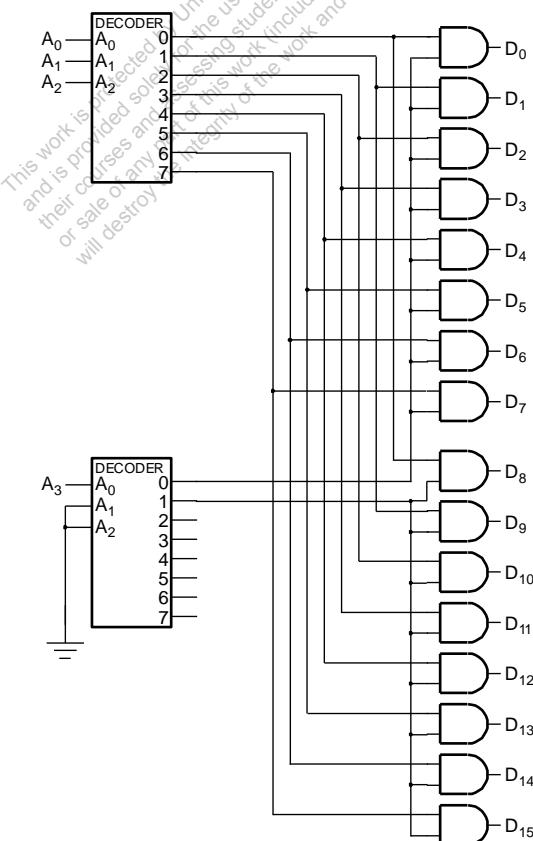


3-27.

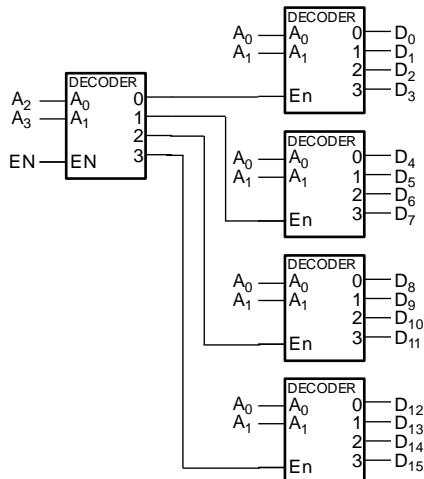
$$A = (S_0 \cdot S_1 \cdot S_2 \cdot S_3 \cdot S_4 \cdot S_5 + M \\ L = A \\ V = \bar{A} = (\overline{S_0 \cdot S_1 \cdot S_2 \cdot S_3 \cdot S_4 \cdot S_5} + M \\ C = V$$



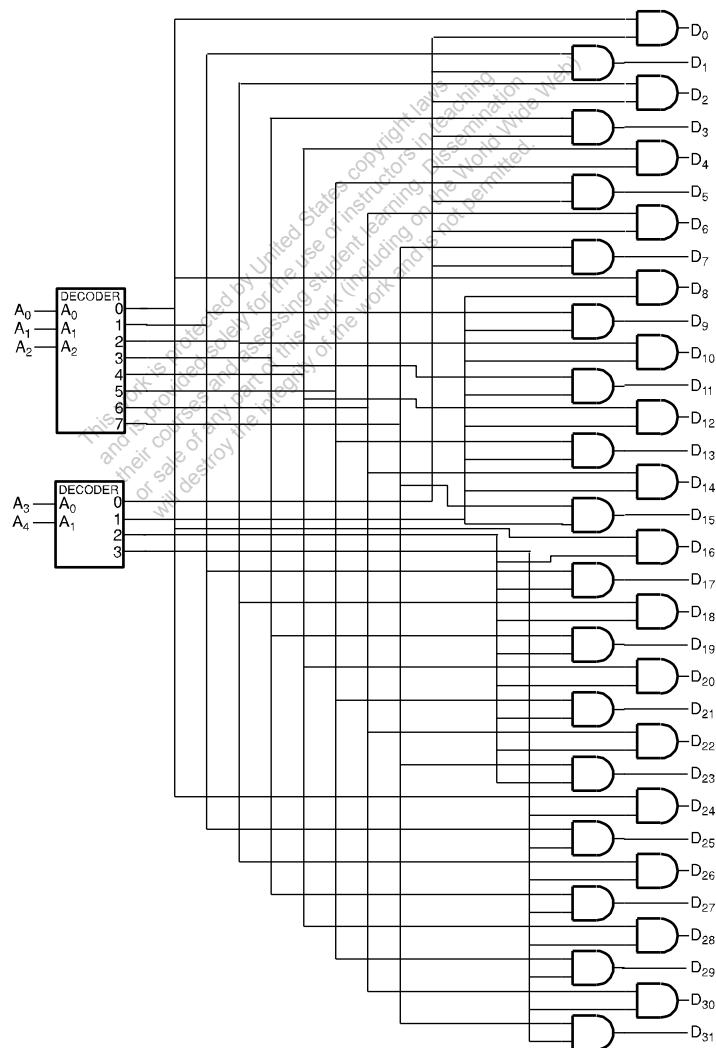
3-28.



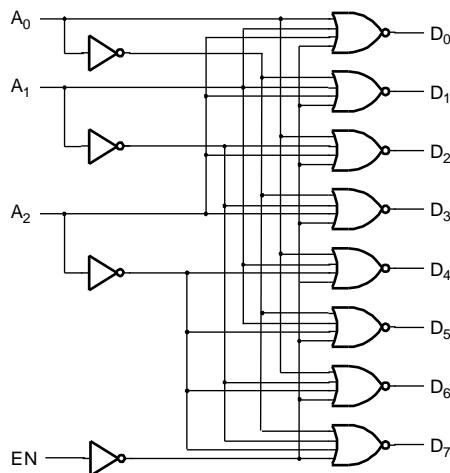
3-29.



3-30.*

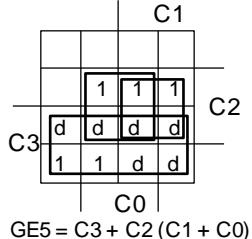


3-33.

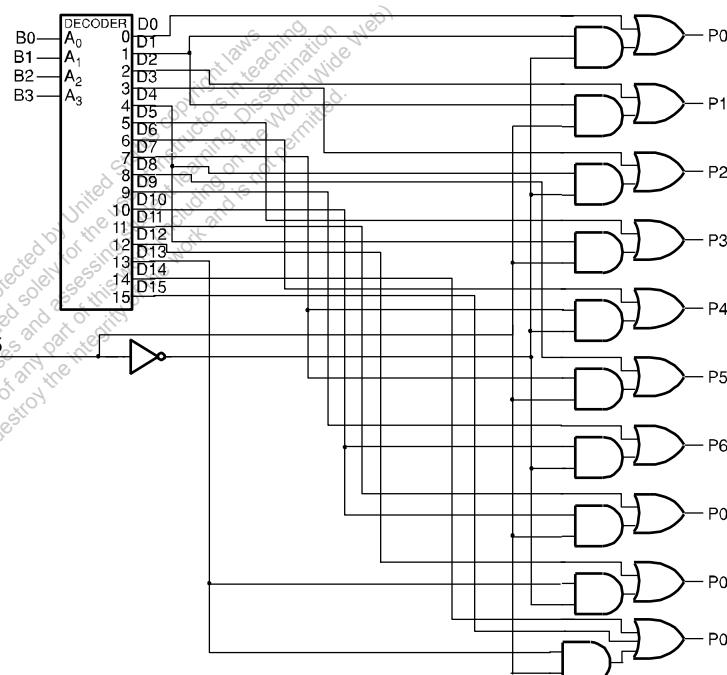


3-34.

K-Map for GE5: BCD = (C3, C2, C1, C0)



Equations for output logic:
 $P_0 = D_0 + GE5 \cdot D_1$
 $P_1 = D_2 + GE5 \cdot D_1$
 $P_2 = D_3 + GE5 \cdot D_4$
 $P_3 = D_5 + GE5 \cdot D_4$
 $P_4 = D_6 + GE5 \cdot D_7$
 $P_5 = D_8 + GE5 \cdot D_7$
 $P_6 = D_9 + GE5 \cdot D_{10}$
 $P_7 = D_{11} + GE5 \cdot D_{10}$
 $P_8 = D_{12} + GE5 \cdot D_{13}$
 $P_9 = D_{14} + D_{15} + GE5 \cdot D_{13}$



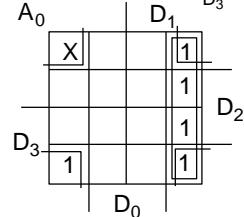
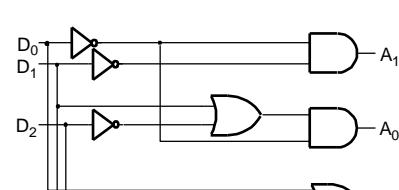
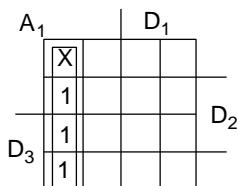
3-35.*

D_3	D_2	D_1	D_0	A_1	A_0	V
0	0	0	0	X	X	0
X	X	X	1	0	0	1
X	X	1	0	0	1	1
X	1	0	0	1	0	1
1	0	0	0	1	1	1

$$V = D_0 + D_1 + D_2 + D_3$$

$$A_0 = \overline{D_0}(D_1 + \overline{D}_2)$$

$$A_1 = \overline{D_0}\overline{D_1}$$

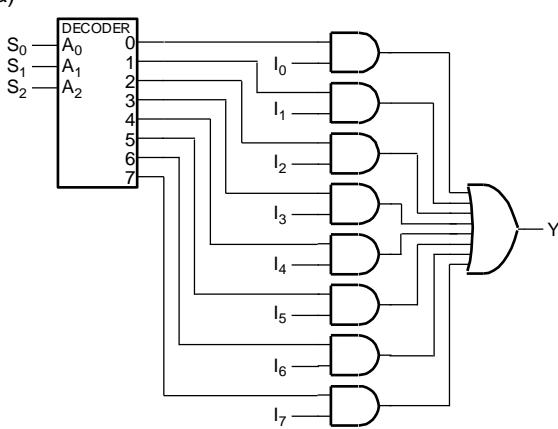

3-36.

Decimal Inputs										Binary Outputs				
9	8	7	6	5	4	3	2	1	0	A ₃	A ₂	A ₁	A ₀	V
0	0	0	0	0	0	0	0	0	0	X	X	X	X	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0	0	0	0	0	0	0	0	0	0	0	1	0	0	1
0	0	0	0	0	0	0	0	0	0	0	1	0	1	1
0	0	0	0	0	0	0	0	0	0	0	1	0	1	1
0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
0	1	X	X	X	X	X	X	X	X	1	0	0	0	1
1	X	X	X	X	X	X	X	X	X	1	0	0	1	1

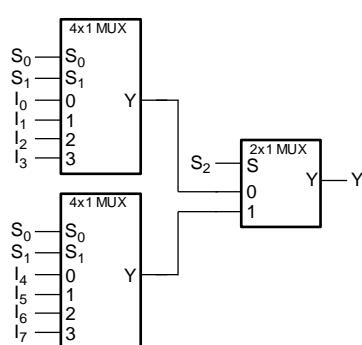
This work is protected by United States copyright laws and is provided solely for the use of instructors in teaching their courses and assessing student learning. Dissemination or sale of any part of this work is illegal. To purchase additional copies or electronic versions for classroom purposes, please contact Pearson Education, Inc. www.pearsonhighered.com

3-37.

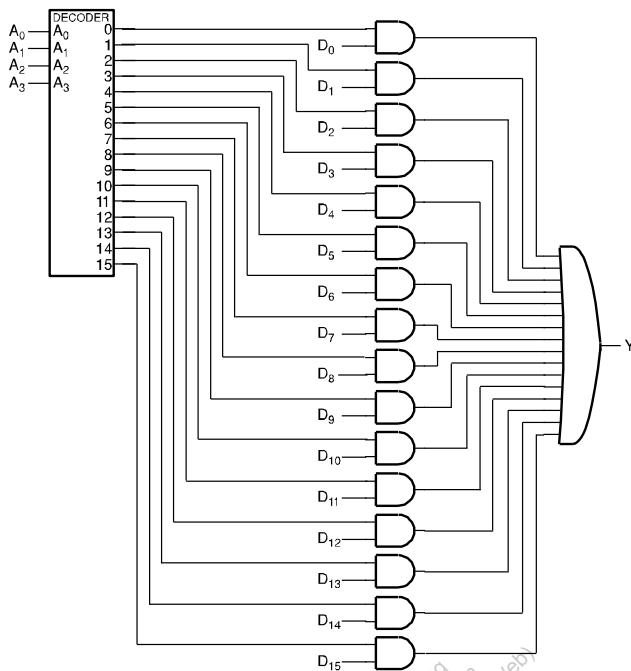
a)



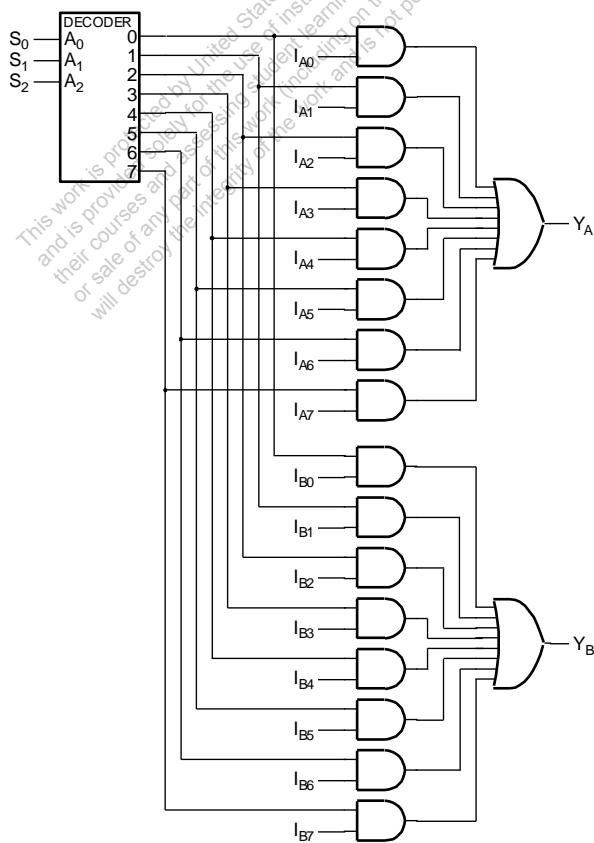
b)



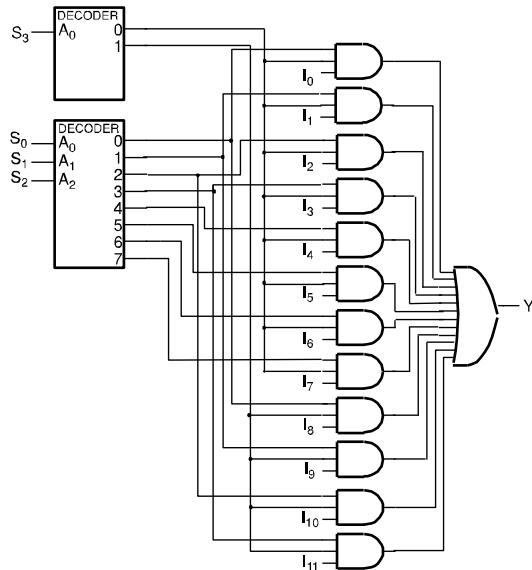
3-38.



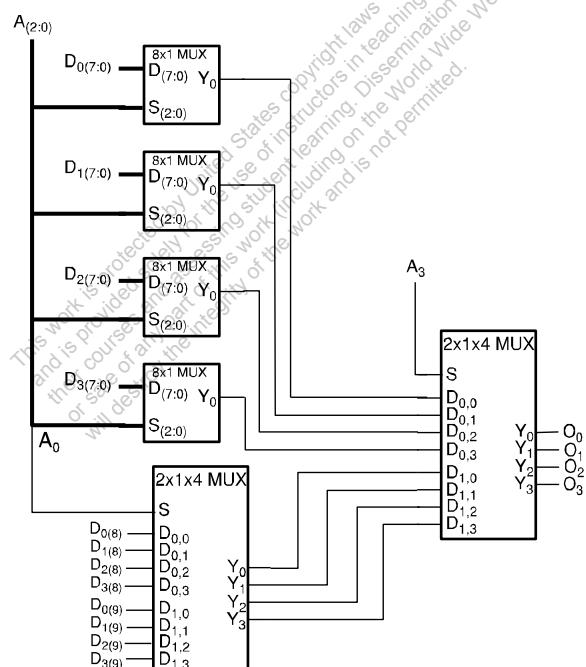
3-39.



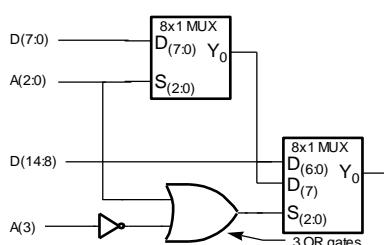
3-40.



3-41.



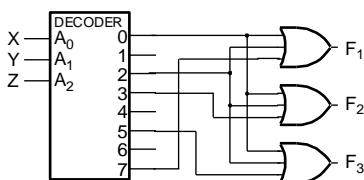
3-42.*



3-43.*

A ₁	A ₀	E	D ₀	D ₁	D ₂	D ₃
0	0	0	0	0	0	0
0	0	1	1	0	0	0
0	1	0	0	0	0	0
0	1	1	0	1	0	0
1	0	0	0	0	0	0
1	0	1	0	0	1	0
1	1	0	0	0	0	0
1	1	1	0	0	0	1

Consider E as the data input and A₀, A₁ as the select lines. For a given combination on (A₁, A₀), the value of E is distributed to the corresponding D output. For example for (A₁, A₀) = (10), the value of E appears on D₂, while all other outputs have value 0.

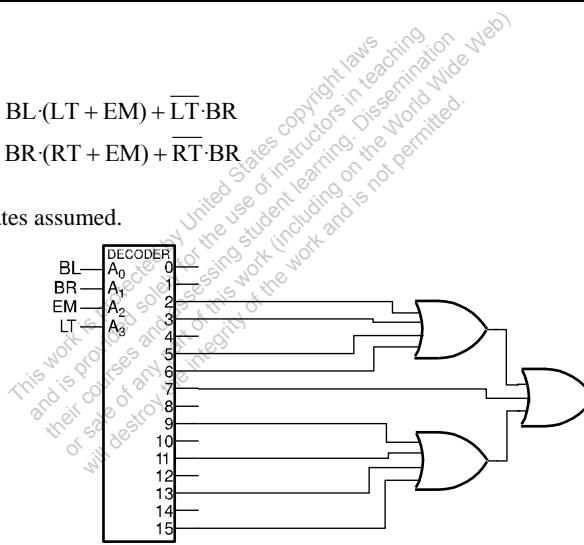
3-44.

3-45.

a) LR = LT·BL + $\overline{LT} \cdot BR + EM \cdot BL = BL \cdot (LT + EM) + \overline{LT} \cdot BR$

RR = RT·BL + $\overline{RT} \cdot BR + EM \cdot BL = BR \cdot (RT + EM) + \overline{RT} \cdot BR$

b) Maximum of four inputs on OR gates assumed.

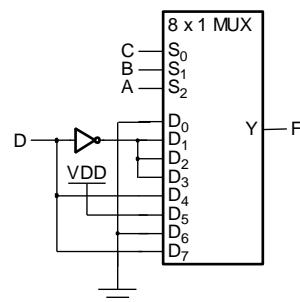
LT	EM	BR	BL	LR
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	1



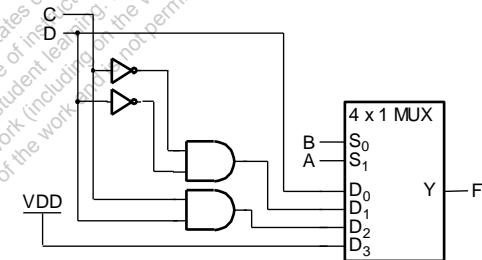
For RR, same circuit with LT replace by RT.

3-46.

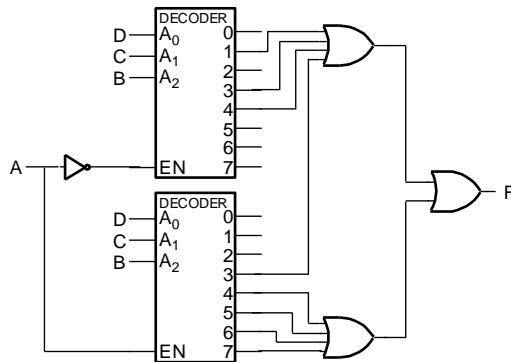
A	B	C	D	F
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1


 3-47.*
 This work is protected by United States copyright laws
 and is provided solely for the use of insititutors in teaching
 their courses and assessing student learning.
 Dissemination or sale of any part of this work (including on the World Wide Web)
 will destroy the integrity of the work and is not permitted.

A	B	C	D	F
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	1
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1



3-48.



3-49.

$$\begin{aligned}
 *S_0 &= C_0 \overline{A_0} \overline{B_0} + \overline{C_0} \overline{A_0} \overline{B_0} + \overline{C_0} \overline{A_0} B_0 + C_0 A_0 B_0 \\
 S_1 &= C_1 \overline{A_1} \overline{B_1} + \overline{C_1} \overline{A_1} \overline{B_1} + \overline{C_1} \overline{A_1} B_1 + C_1 A_1 B_1 \\
 *S_1 &= C_0 A_0 \overline{A_1} \overline{B_1} + A_0 B_0 \overline{A_1} \overline{B_1} + C_0 B_0 \overline{A_1} \overline{B_1} \\
 &\quad + \overline{C_0} \overline{A_0} \overline{A_1} \overline{B_1} + \overline{A_0} \overline{B_0} \overline{A_1} \overline{B_1} + \overline{C_0} \overline{B_0} \overline{A_1} \overline{B_1} \\
 &\quad + \overline{C_0} \overline{A_0} A_1 B_1 + \overline{A_0} \overline{B_0} A_1 B_1 + \overline{C_0} \overline{B_0} A_1 B_1 \\
 &\quad + C_0 A_0 A_1 B_1 + A_0 B_0 A_1 B_1 + C_0 B_0 A_1 B_1
 \end{aligned}
 \quad
 \begin{aligned}
 C_1 &= C_0 A_0 + A_0 B_0 + C_0 B_0 \\
 C_2 &= C_1 A_1 + A_1 B_1 + C_1 B_1 \\
 *C_2 &= C_0 A_0 A_1 + A_0 B_0 A_1 + C_0 B_0 A_1 \\
 &\quad + C_0 A_0 B_1 + A_0 B_0 B_1 + C_0 B_0 B_1 + A_1 B_1
 \end{aligned}$$

This work is protected by United States copyright laws or laws of other countries. Dissemination or unauthorized use of this work is illegal. Under copyright law, only the user's institution may reproduce, display, or store this work in a library, provided that:

1) the user does not further distribute, transmit, or display the work;

2) the user creates no derivative work, such as another version of this work;

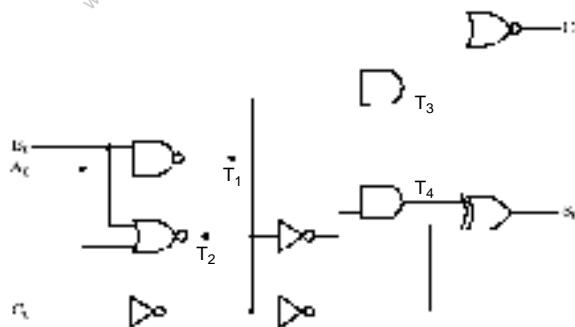
3) the user does not sell this work in any format or medium without the prior permission of the copyright holders;

4) the user complies with all other applicable laws and regulations, including but not limited to fair use.

* These are the three equations for the outputs. The logic diagram consists of a sum-of-products implementation of these equations.

 3-50.*

$$\begin{aligned}
 C_1 &= \overline{T_3 + T_2} = \overline{T_1 C_0 + T_2} = \overline{\overline{A_0} \overline{B_0} C_0 + A_0 + \overline{B_0}} = (\overline{A_0} + \overline{B_0}) \overline{C_0} + \overline{A_0} \overline{B_0} = (A_0 B_0 + C_0)(A_0 + B_0) \\
 C_1 &= A_0 B_0 + A_0 C_0 + B_0 C_0 \\
 S_0 &= C_0 \oplus T_4 = C_0 \oplus \overline{T_1 T_2} = C_0 \oplus \overline{\overline{A_0} \overline{B_0} (A_0 + B_0)} = C_0 \oplus (\overline{A_0} + \overline{B_0})(A_0 + B_0) = C_0 \oplus A_0 \overline{B_0} + \overline{A_0} B_0 \\
 S_0 &= A_0 \oplus B_0 \oplus C_0
 \end{aligned}$$


 3-51.*

	1001 1100	1001 1101	1010 1000	0000 0000	1000 0000
Unsigned					
1's Complement	0110 0011	0110 0010	0101 0111	1111 1111	0111 1111
2's Complement	0110 0100	0110 0011	0101 1000	0000 0000	1000 0000

3-52.

$$\begin{array}{r}
 \text{a) } \begin{array}{r} 11010 \\ + 01111 \\ \hline 01001 \end{array} \\
 \text{b) } \begin{array}{r} 11110 \\ + 10010 \\ \hline 10000 \end{array} \\
 \text{c) } \begin{array}{r} 1111110 \\ + 0000010 \\ \hline 0000000 \end{array} \\
 \text{d) } \begin{array}{r} 101001 \\ + 111011 \\ \hline 100100 \end{array}
 \end{array}$$

3-53.

$$\begin{array}{r}
 \text{a) } \begin{array}{r} 11010 \\ + 01111 \\ \hline 01001 \end{array} \\
 \text{b) } \begin{array}{r} 11110 \\ + 00010 \\ \hline 00000 \end{array} \\
 \text{c) } \begin{array}{r} 1111110 \\ + 0000010 \\ \hline 0000000 \end{array} \\
 \text{d) } \begin{array}{r} 101001 \\ + 000011 \\ \hline 101100 \end{array}
 \end{array}$$

3-54.*

$$\begin{array}{r}
 \begin{array}{r}
 +36 = 0100100 \\
 -24 = 1101000 \\
 -35 = 1011101
 \end{array}
 \quad
 \begin{array}{r}
 36 \\
 +(-24) \\
 = 12
 \end{array}
 \quad
 \begin{array}{r}
 0100100 \\
 +1101000 \\
 = 10001100
 \end{array}
 \\
 \begin{array}{r}
 -35 \\
 -(-24) \\
 = -11
 \end{array}
 \quad
 \begin{array}{r}
 1011101 \\
 +0011000 \\
 = 1110101
 \end{array}
 \end{array}$$

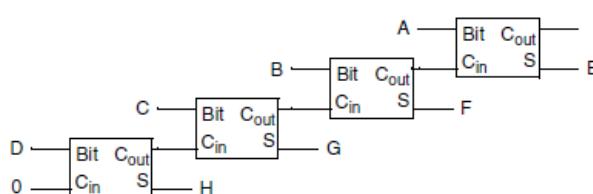
3-55.

$$\begin{array}{r}
 \text{a) } \begin{array}{r}
 100111 \\
 + 111001 \\
 \hline 100000
 \end{array}
 \quad
 \text{b) } \begin{array}{r}
 -25 \\
 -7 \\
 \hline -32
 \end{array}
 \quad
 \text{c) } \begin{array}{r}
 0010011 \\
 + 100110 \\
 \hline 110001
 \end{array}
 \quad
 \text{d) } \begin{array}{r}
 110001 \\
 + 101110 \\
 \hline 011111
 \end{array}
 \quad
 \begin{array}{r}
 -15 \\
 -18 \\
 \hline -33
 \end{array}
 \quad
 \begin{array}{r}
 101110 \\
 + 001001 \\
 \hline 110111
 \end{array}
 \quad
 \begin{array}{r}
 -18 \\
 -9 \\
 \hline -9
 \end{array}
 \quad
 \text{Overflow}
 \end{array}$$

3-56.*

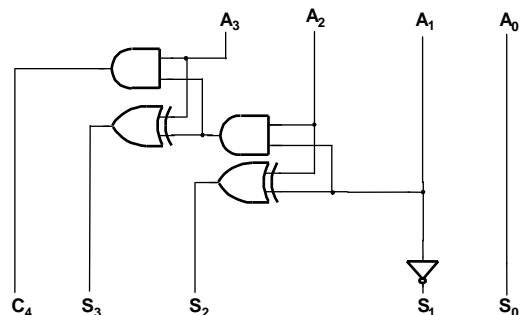
$$\begin{array}{l}
 \text{a) } H = D \\
 G = C \oplus D \\
 F = \bar{B}C + \bar{B}D + B\bar{C}\bar{D} \\
 E = \bar{A}B + A\bar{B}\bar{C}\bar{D} + \bar{A}C + \bar{A}D
 \end{array}$$

$$\begin{array}{ll}
 \text{b) Bit} & \text{Cin} \quad S \quad \text{Cout} \quad S = \text{Bit} \oplus \text{Cin} \\
 0 & 0 \quad 0 \quad 0 \quad 0 \\
 0 & 1 \quad 1 \quad 1 \\
 1 & 0 \quad 1 \quad 1 \\
 1 & 1 \quad 0 \quad 1
 \end{array}
 \quad
 \begin{array}{l}
 \text{Cout} = \text{Bit} + \text{Cin}
 \end{array}$$



A	B	C	D	E	F	G	H
0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	1
0	0	1	0	1	1	1	0
0	0	1	1	1	1	0	1
0	1	0	0	1	0	0	0
0	1	0	1	0	1	1	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	0	1	0	0
1	0	0	1	0	1	1	1
1	0	1	0	1	0	1	0
1	0	1	1	0	1	0	1
1	1	0	0	0	0	1	0
1	1	0	1	0	1	1	0
1	1	1	0	0	1	0	1
1	1	1	1	0	0	0	1
1	1	1	1	1	0	0	0

- c) Using shared inverters, XOR cost = 6. Gate input cost for a = 4 + 0 + 6 + 10 + 14 = 34 Gate input cost for b = 4 x (2 + 6 + 2) = 40 In terms of gate cost, a is the better design.

3-57.

3-58.

$$B_0 = 1 \quad C_{in} = 0$$

$$S_0 = A_0 \oplus 1 \oplus 0 = \overline{A_0}$$

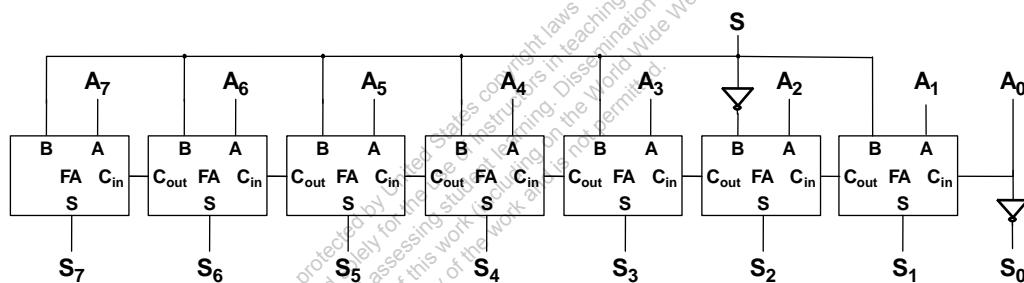
$$C_0 = A_0 \bullet 1 + 0(A_0 + 1) = A_0$$

$$B_1 = S$$

$$B_2 = \overline{S}$$

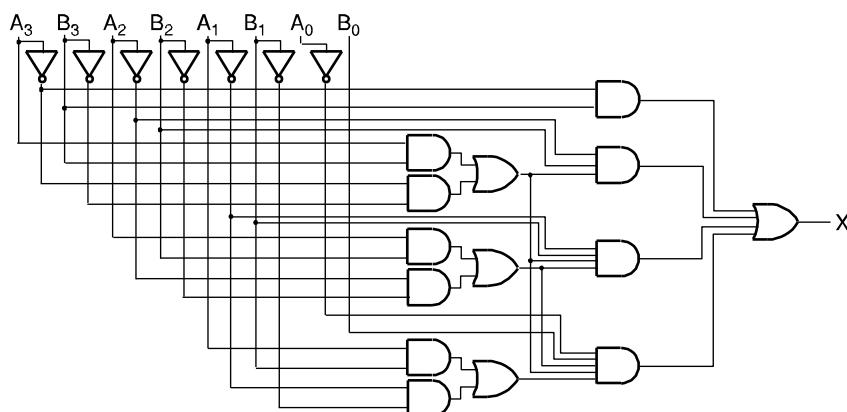
$$B_{3-7} = S$$

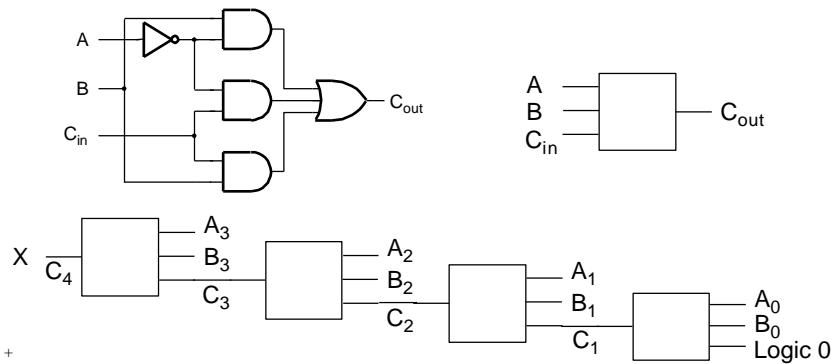
Bits 1-6 use regular full adder/subtractor logic. For bit 7, the carry logic is omitted.


3-59.

Proceeding from MSB to LSB: $A < B$ if $A_i < B_i$ ($\bar{A}_i B_i = 1$) and for all $j > i$, $A_j = B_j$ ($A_j B_j + \bar{A}_j \bar{B}_j = 1$) Based on the above,

$$\begin{aligned} X = & \bar{A}_3 B_3 + (A_3 B_3 + \bar{A}_3 \bar{B}_3) \bar{A}_2 B_2 + (A_3 B_3 + \bar{A}_3 \bar{B}_3)(A_2 B_2 + \bar{A}_2 \bar{B}_2) \bar{A}_1 B_1 \\ & + (A_3 B_3 + \bar{A}_3 \bar{B}_3)(A_2 B_2 + \bar{A}_2 \bar{B}_2)(A_1 B_1 + \bar{A}_1 \bar{B}_1) \bar{A}_0 B_0 \end{aligned}$$



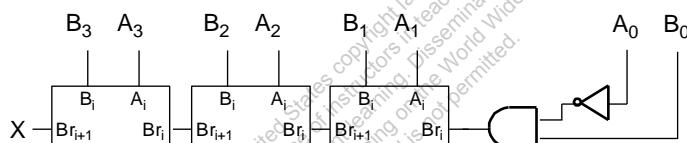
3-60.⁺


3-61.

In a subtractor, the sum is replaced by the difference and the carry is replaced by the borrow. The borrow at any given point is a 1 only if in the LSB direction from that point, $A < B$.

Only borrow logic is needed to produce X , so the difference logic is discarded in contraction. The remaining equation for borrow into the $i + 1$ position is: $Br_{i+1} = \bar{A}_i B_i + \bar{A}_{i-1} Br_i + B_i Br_i$ for $i = 0, 1, 2, 3$. $Br_0 = 0$ giving

$Br_1 = \bar{A}_0 B_0$. When the borrow $Br_4 = 1$, then $A < B$. Thus, $X = Br_4$. The resulting circuit using the borrow logic is:


 3-62.⁺

This problem requires two decisions; Is $A > B$? Is $A = B$? Two “carry” lines are required to build an iterative circuit, G_i and E_i . These carries are assumed to pass through the circuit from right to left with $G_0 = 0$ and $E_0 = 1$. Each cell has inputs A_i , B_i , G_i , and E_i and outputs G_{i+1} and E_{i+1} . Using K-maps, cell equations are:

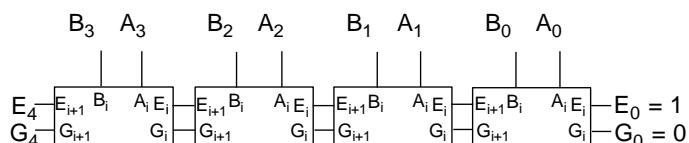
$$E_{i+1} = \bar{A}_i \bar{B}_i E_i + A_i B_i E_i$$

$$G_{i+1} = A_i \bar{B}_i E_i + (A_i + \bar{B}_i) G_i$$

Using multilevel circuit techniques, the cost can be reduced by sharing terms:

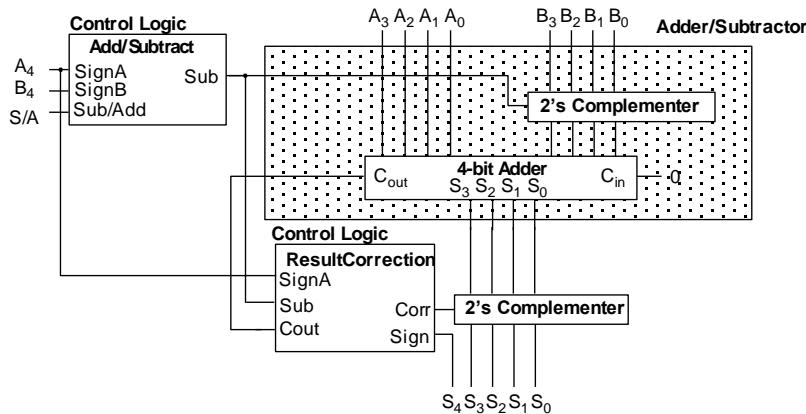
$$E_{i+1} = (\bar{A}_i \bar{B}_i + \bar{A}_i B_i) E_i$$

$$G_{i+1} = (A_i \bar{B}_i + (\bar{A}_i B_i)) G_i$$



3-63.⁺

Circuit Diagram:



Control Logic Truth Tables

S/A	Inputs			Inputs			Corr	Sign	Overflow
	Sign A	SignB	Sub	Sub	SignA	Cout			
0	0	0	0	0	0	0	0	0	0
0	0	1	1	0	0	1	0	0	1
0	1	0	1	0	1	0	0	1	0
0	1	1	0	0	1	1	0	1	1
1	0	0	1	1	0	0	1	1	0
1	0	1	0	1	0	1	0	0	0
1	1	0	0	1	1	0	1	0	0
1	1	1	1	1	1	1	0	1	0

$$\text{Sub} = \overline{S/A} \oplus \text{SignA} \oplus \text{SignB}$$

$$\text{Corr} = \text{Sub} \ \overline{\text{Cout}}$$

$$\text{Sign} = \text{SignA} \oplus \text{Corr}$$

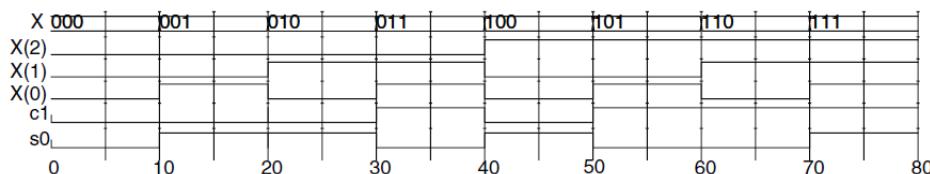
$$\text{Overflow} = \overline{\text{Sub}} \ \text{Cout}$$

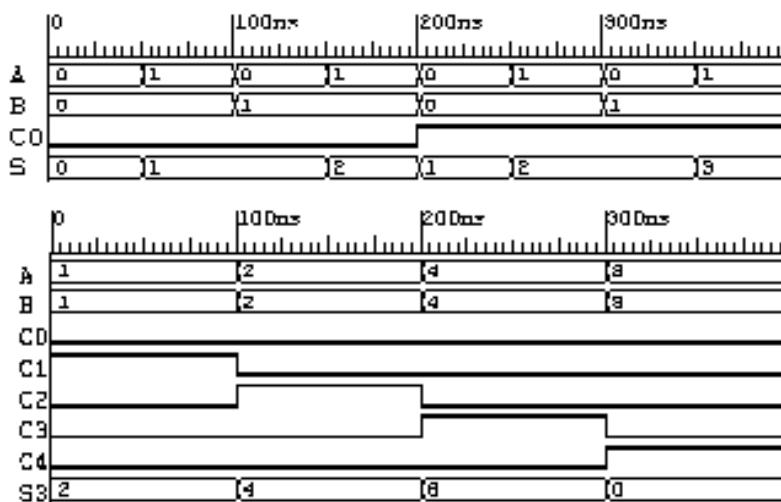
3-64.*

	S	A	B	C ₄	S ₃	S ₂	S ₁	S ₀
a)	0	0111	0111	0	1	1	1	0
b)	1	0100	0111	0	1	1	0	1
c)	1	1101	1010	1	0	0	1	1
d)	0	0111	1010	1	0	0	0	1
e)	1	0001	1000	0	1	0	0	1

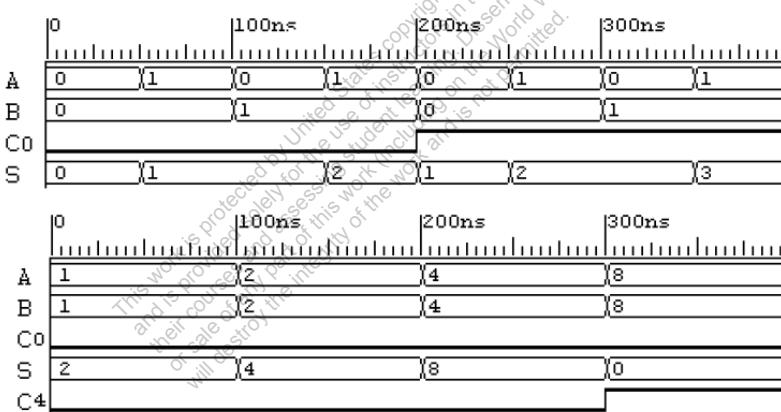
3-65.

```
-- Full Adder: Structural VHDL Description
-- (See Figure 4-28 for logic diagram)
library ieee, lcdf_vhdl;
use ieee.std_logic_1164.all, lcdf_vhdl.func_prims.all; --X is input vector (A0, B0, C0).
entity full_adder_st is
port(X: in std_logic_vector(0 to 2);
C1, S0: out std_logic);
end;
architecture logic of full_adder_st is
component NOT1
port(in1: in std_logic;
out1: out std_logic);
end component;
component NAND2
port(in1, in2: in std_logic;
out1: out std_logic);
end component;
component NOR2
port(in1, in2: in std_logic;
out1: out std_logic);
end component;
component AND2
port(in1, in2: in std_logic;
out1: out std_logic);
end component;
component XOR2
port(in1, in2: in std_logic;
out1: out std_logic);
end component;
signal S: std_logic_vector(0 to 6); -- S(0 to 6) is the vector of six gate output signals
-- from upper left to lower right.
--The following is the circuit netlist.
begin
g0: NAND2 port map (X(1), X(0), S(0));
g1: NOR2 port map (X(1), X(0), S(1));
g2: NOT1 port map (X(2), S(2));
g3: AND2 port map (S(0), S(2), S(3));
g4: NOT1 port map (S(1), S(4));
g5: NOT1 port map (S(2), S(5));
g6: AND2 port map (S(0), S(4), S(6));
g7: NOR2 port map (S(1), S(3), Z(1));
g8: XOR2 port map (S(6), S(5), Z(0));
end logic;
```



3-66.

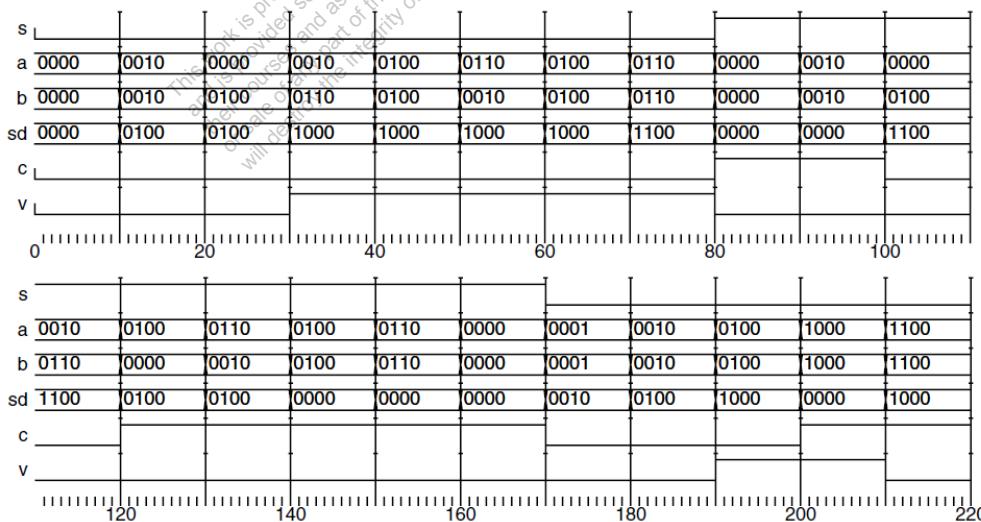
The solution given is very thorough since it checks each of the carry connections between adjacent 0 and 1. In contrast applying C0 = 1 and A = 15 with B = 0 would allow a whole variety of incorrect connections between a variety of incorrect connections between cells that would not be detected.

3-67.*

The solution given is very thorough since it checks each of the carry connections between adjacent cells transferring 0 and 1. In contrast a test applying C0 = 1 and A = 15 with B = 0 would allow a whole variety of incorrect connections between cells that would not be detected.

3-68.⁺

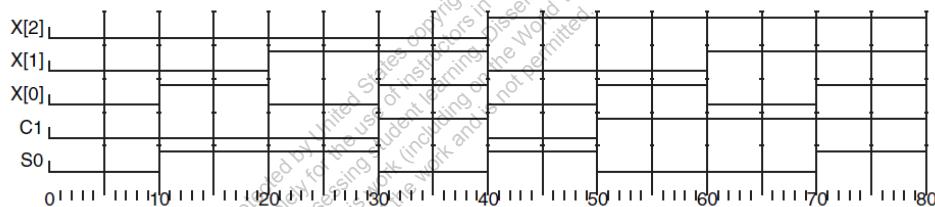
```
-- prob_4-30 adder-subtractor with overflow detection
library ieee;
use ieee.std_logic_1164.all, ieee.std_logic_unsigned.all;
entity addsubov is
port(A,B: in std_logic_vector(3 downto 0);
S: in std_logic;
SD: out std_logic_vector(3 downto 0);
C, V: out std_logic);
end addsubov;
architecture behavior of addsubov is
signal Y: std_logic_vector(3 downto 0);
signal temp4: std_logic_vector(3 downto 0);
signal temp2: std_logic_vector(4 downto 3);
signal CI: std_logic_vector(4 downto 3);
begin
begin
with S select
Y(2 downto 0) <= B(2 downto 0) when '0',
not B(2 downto 0) when '1',
"XXX" when others;
with S select
Y(3) <= B(3) when '0',
not B(3) when '1',
'X' when others;
temp4 <= ('0' & A(2 downto 0)) + ('0' & Y(2 downto 0)) + ("000" & S);
CI(3) <= temp4(3);
temp2 <= ('0' & A(3)) + ('0' & Y(3)) + ('0' & CI(3));
CI(4) <= temp2(4);
SD <= (temp2(3) & temp4(2 downto 0));
C <= CI(4);
V <= CI(4) xor CI(3);
end behavior;
```



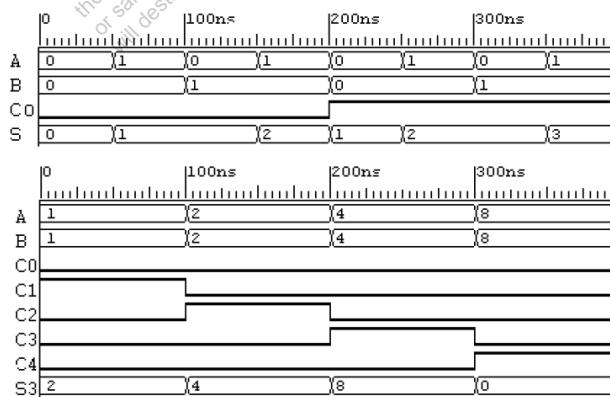
The solution given is very thorough since it checks each of the carry connections between adjacent cells transferring 0 and 1. In contrast a test applying $C_0 = 1$ and $A = 15$ with $B = 0$ would allow a whole variety of incorrect connections between cells that would not be detected.

3-69.

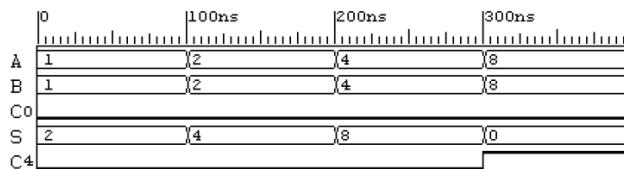
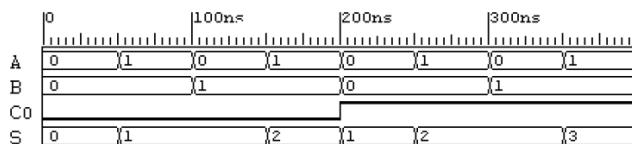
```
// Full Adder: Structural Verilog Description
// (See Figure 4-28 for logic diagram)
module full_adder_st(C1, S0, X);
    input [2:0] X; //X is the vector of inputs (A0, B0, C0).
    output C1, S0;
    wire [0:6] N; //N[0:6] is the six bit vector of gate
                    //outputs from upper left to lower right.
    //The netlist for the gate types:
    nand
        gna(N[0],X[1],X[0]);
    nor
        gno1(N[1],X[1],X[0]),
        gno2(C1,N[1],N[3]);
    not
        gn0(N[2], X[2]),
        gn1(N[4], N[1]),
        gn2(N[5], N[2]);
    and
        ga0(N[3], N[0], N[2]),
        ga1(N[6], N[0], N[4]);
    xor
        gx(S0, N[5], N[6]);
endmodule
```



3-70.



3-71.*



The solution given is very thorough since it checks each of the carry connections between adjacent cells transferring 0 and 1. In contrast a test applying $C_0 = 1$ and $A = 15$ with $B = 0$ would allow a whole variety of incorrect connections between cells that would not be detected.

3-72.

```
// Adder-Subtractor Behavioral Model
module addsub_4b_v (S, A, B, SD, C4);
    input[3:0] A, B;
    input S;
    output[3:0] SD;
    output C4;

    assign {C4, SD} = S?(A - B):(A + B);
endmodule
```

