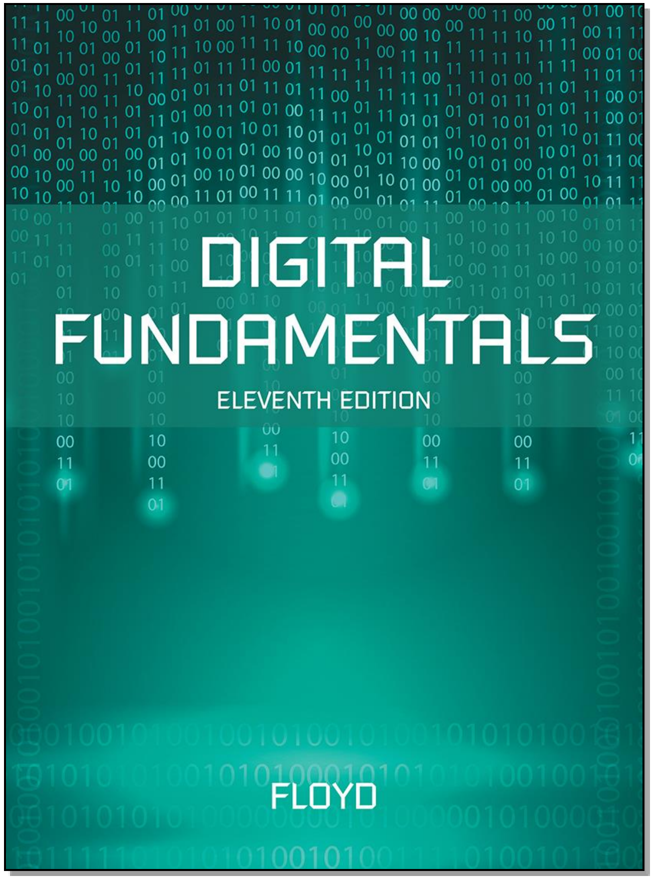
Digital Fundamentals

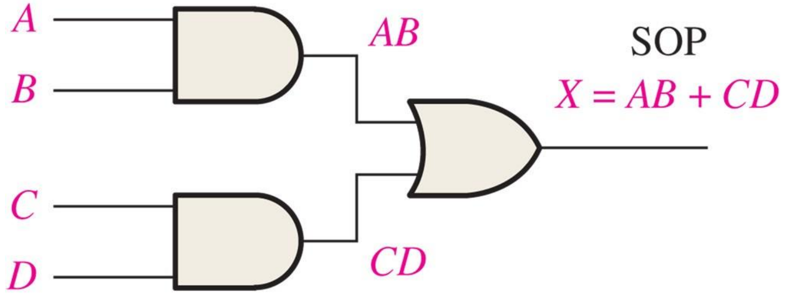
ELEVENTH EDITION



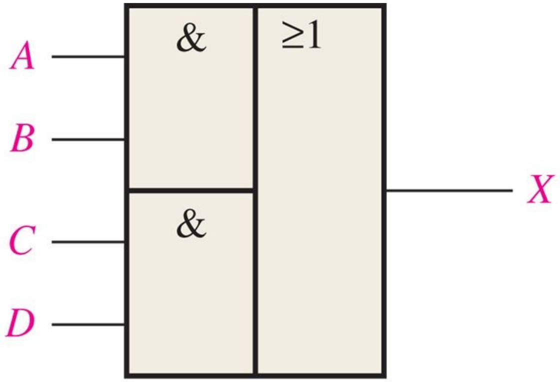
CHAPTER 5

Combinational Logic Analysis

An example of AND-OR logic



(a) Logic diagram (ANSI standard distinctive shape symbols)



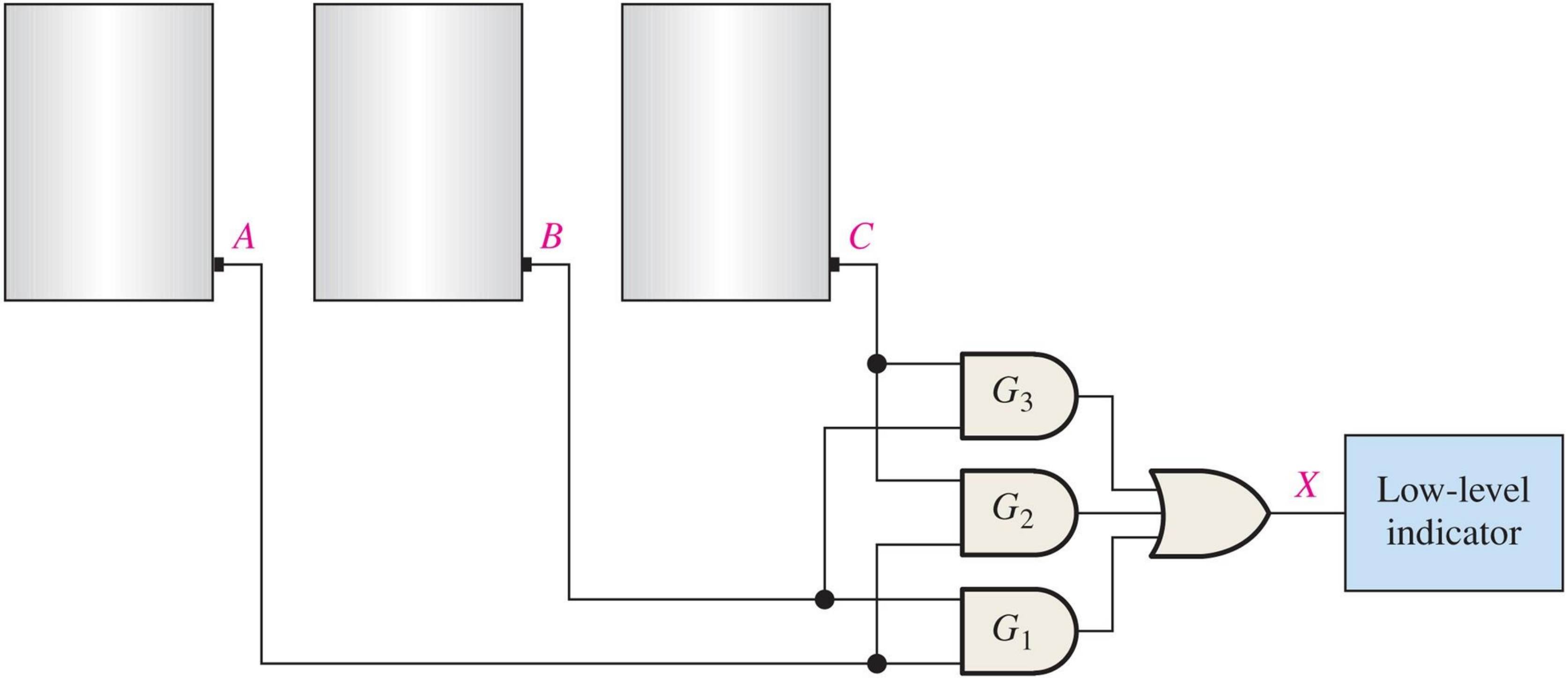
(b) ANSI standard rectangular outline symbol

TABLE 5–1

Truth table for the AND-OR logic in Figure 5–1.

Inputs						Output
A	*B*	*C*	*D*	*AB*	*CD*	*X*
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	0	1	0	0	0	0
0	0	1	1	0	1	1
0	1	0	0	0	0	0
0	1	0	1	0	0	0
0	1	1	0	0	0	0
0	1	1	1	0	1	1
1	0	0	0	0	0	0
1	0	0	1	0	0	0
1	0	1	0	0	0	0
1	0	1	1	0	1	1
1	1	0	0	1	0	1
1	1	0	1	1	0	1
1	1	1	0	1	0	1
1	1	1	1	1	1	1

In a certain chemical-processing plant, a liquid chemical is used in a manufacturing process. The chemical is stored in three different tanks. A level sensor in each tank produces a HIGH voltage when the level of chemical in the tank drops below a specified point. Design a circuit that monitors the chemical level in each tank and indicates when the level in any two of the tanks drops below the specified point.



An AND-OR-Invert circuit produces a POS output

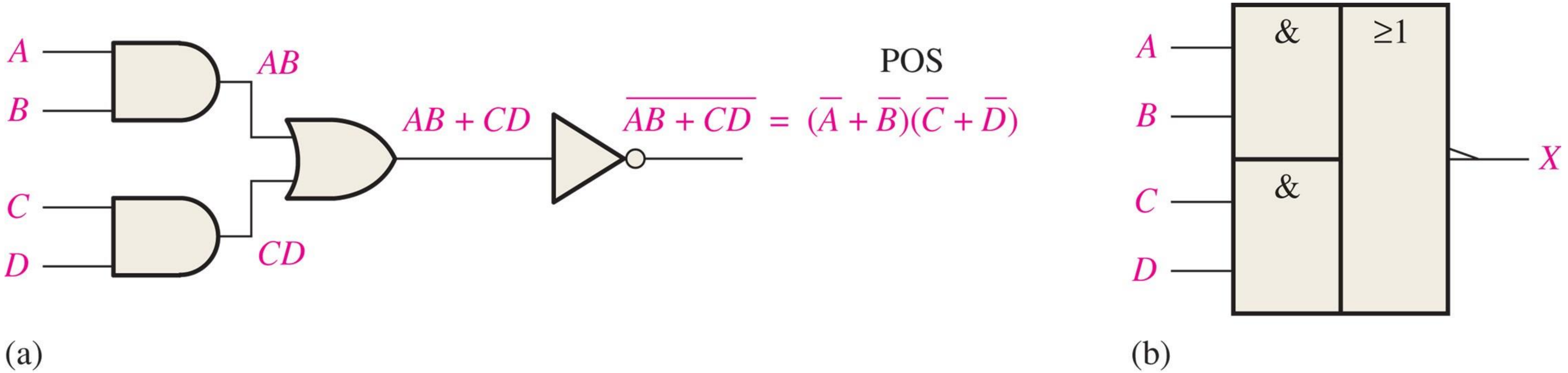
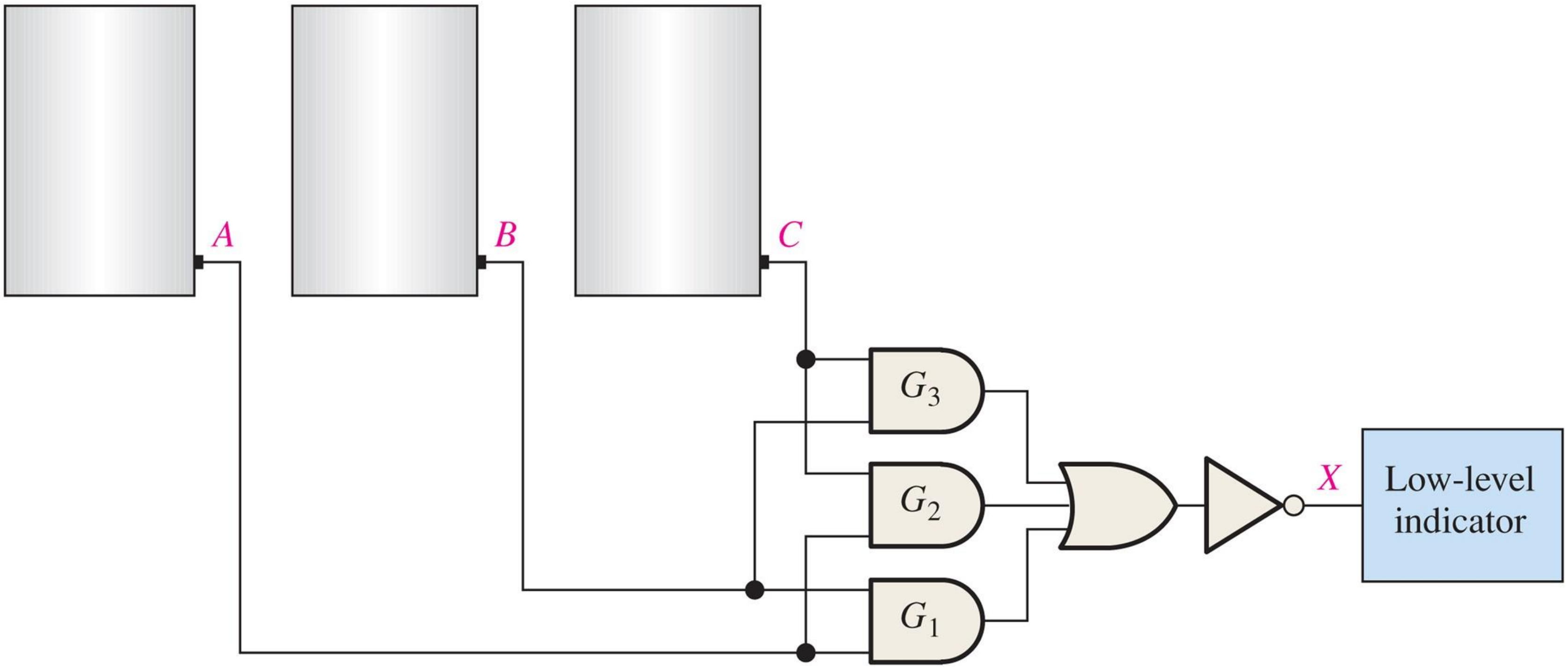
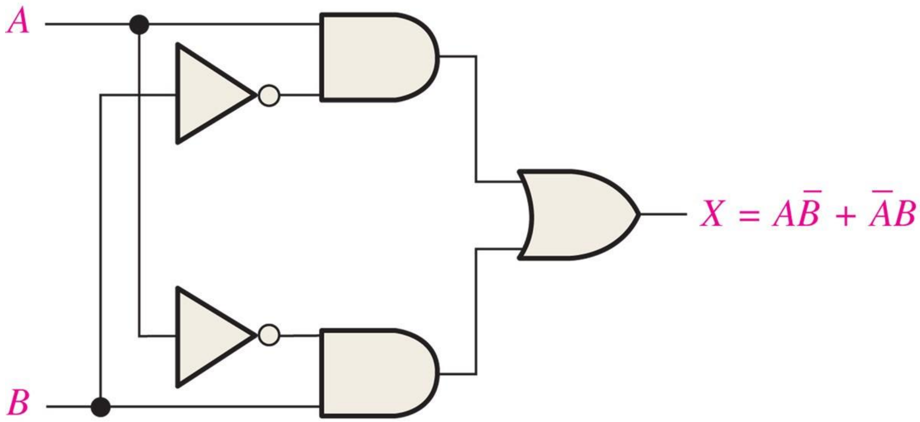

A
B
C
D
AB
CD
$AB + CD$
POS
$\overline{AB + CD} = (\overline{A} + \overline{B})(\overline{C} + \overline{D})$
(a)
A
B
C
D
&
&
≥1
X
(b)

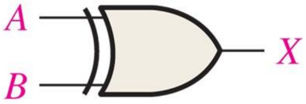
FIGURE 5-4



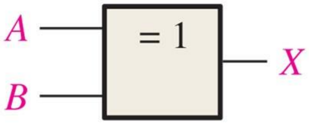
Exclusive-OR logic diagram and symbols



(a) Logic diagram



(b) ANSI distinctive shape symbol



(c) ANSI rectangular outline symbol

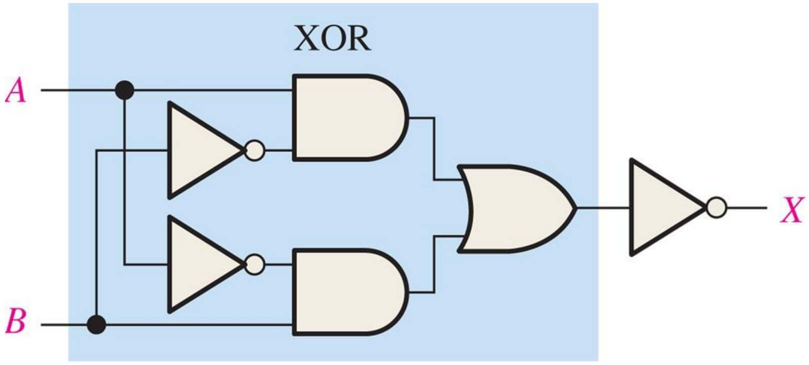
TABLE 5–2

Truth table for an exclusive-OR.

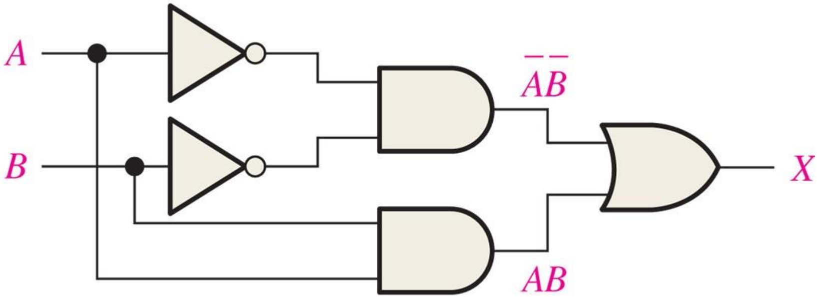
A	*B*	*X*
0	0	0
0	1	1
1	0	1
1	1	0

Exclusive-NOR Logic

Two equivalent ways of implementing the exclusive-NOR

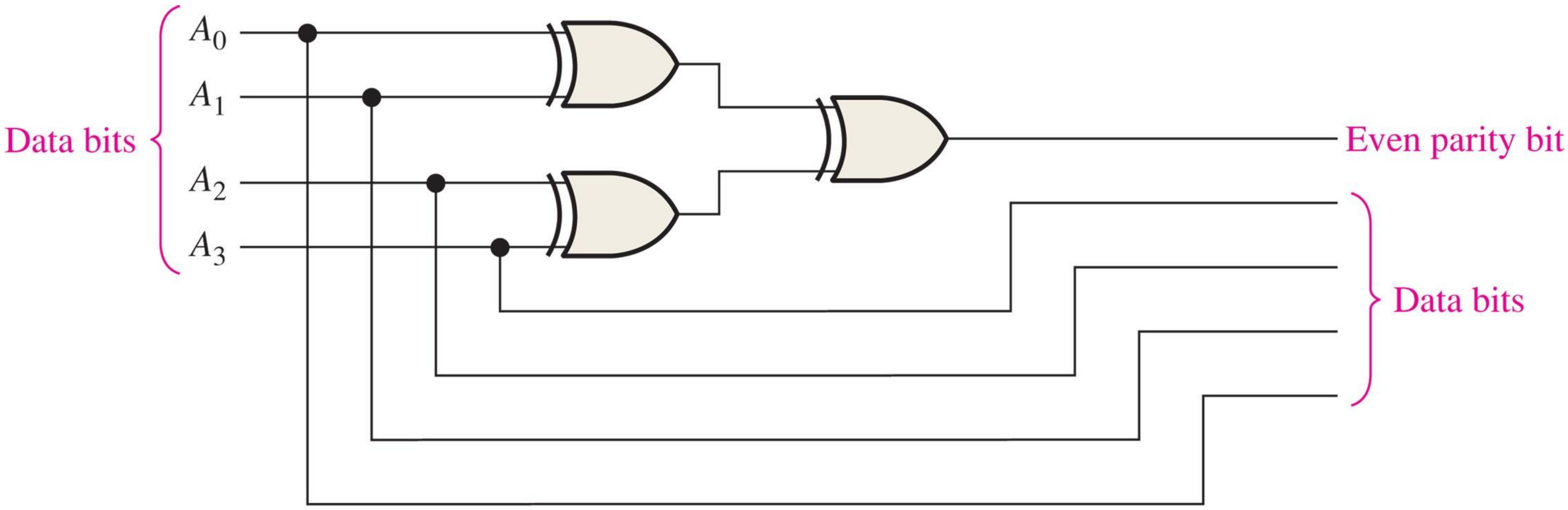


(a) $X = \overline{A\overline{B} + \overline{A}B}$



(b) $X = \overline{A}\,\overline{B} + AB$

Even-parity generator



Logic circuit for $X = AB + CDE$

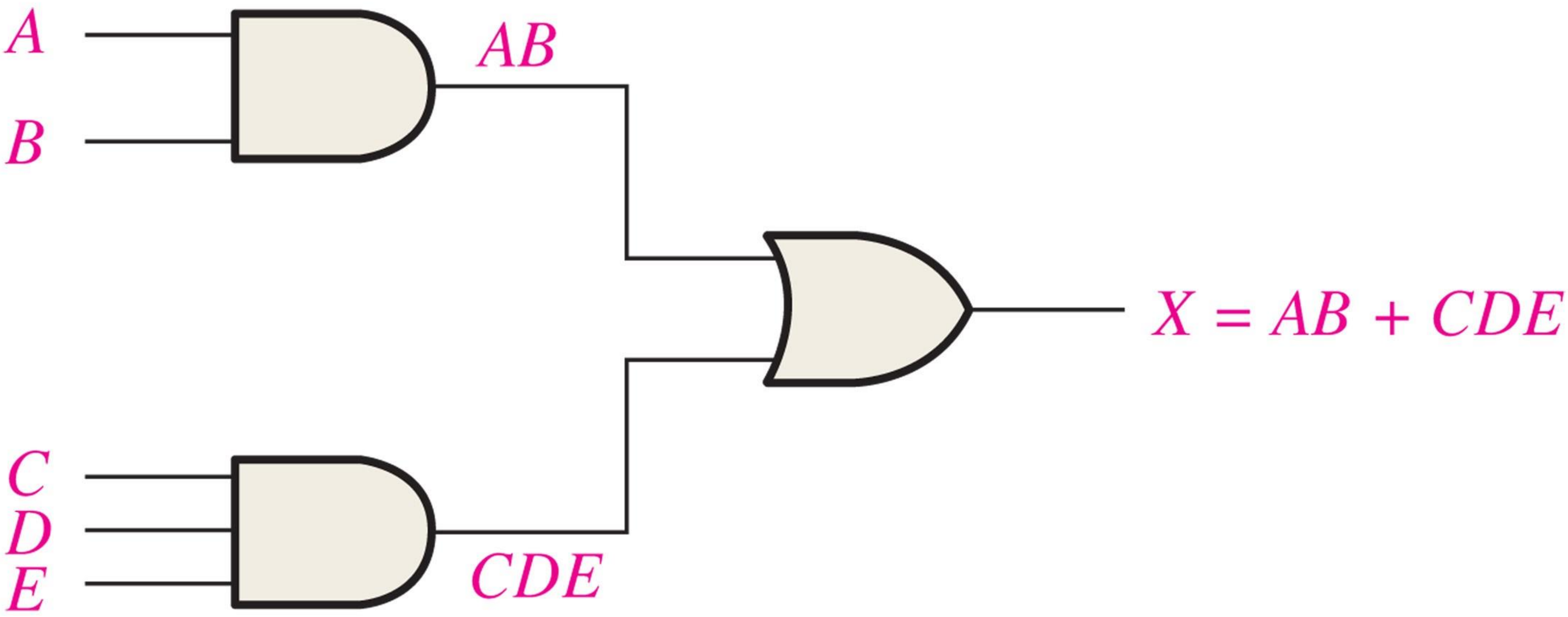
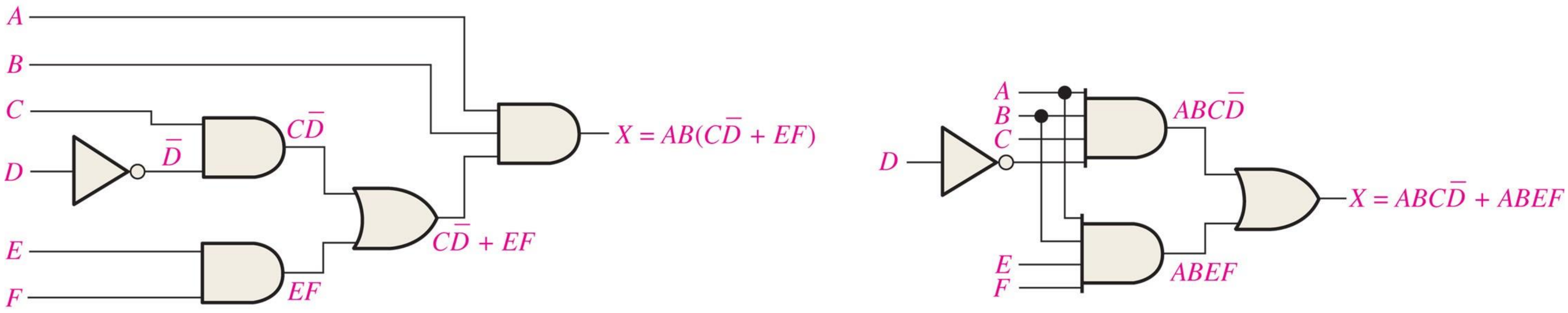


FIGURE 5-10 Logic circuits for $X = AB(C\overline{D} + EF) = ABC\overline{D} + ABEF$.



(a)

(b) Sum-of-products implementation of the circuit in part (a)

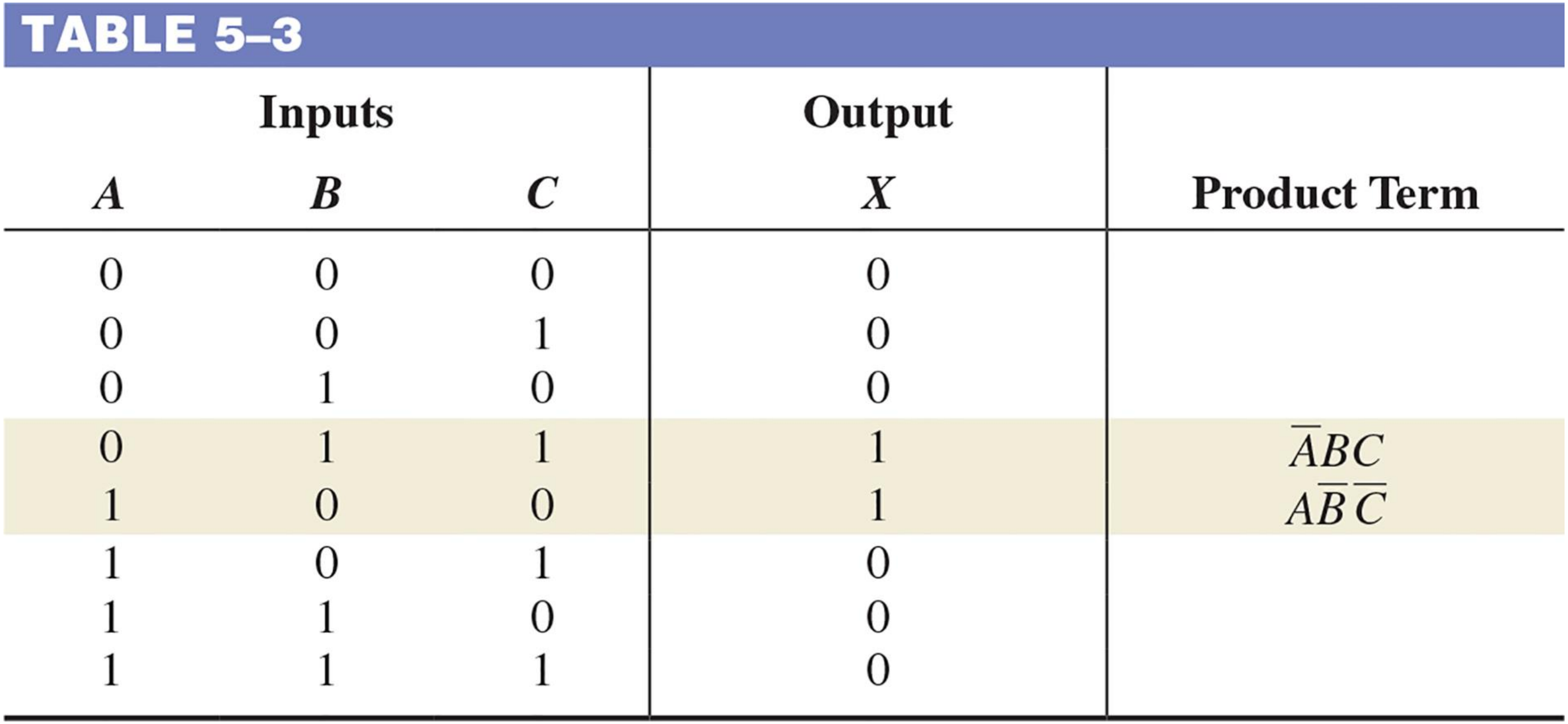


TABLE 5–3

Inputs			Output	
A	B	C	X	Product Term
0	0	0	0	
0	0	1	0	
0	1	0	0	
0	1	1	1	$\overline{A}BC$
1	0	0	1	$A\overline{B}\,\overline{C}$
1	0	1	0	
1	1	0	0	
1	1	1	0	

FIGURE 5-11 Logic circuit for $X = \bar{A}BC + A\bar{B}\bar{C}$.

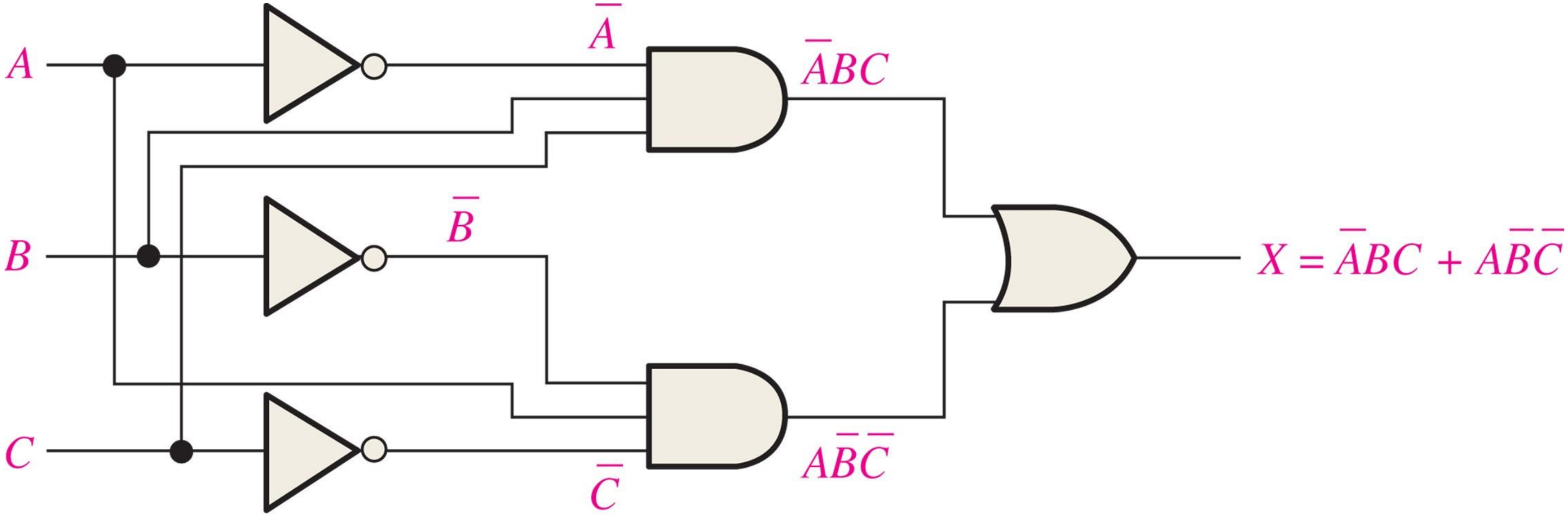
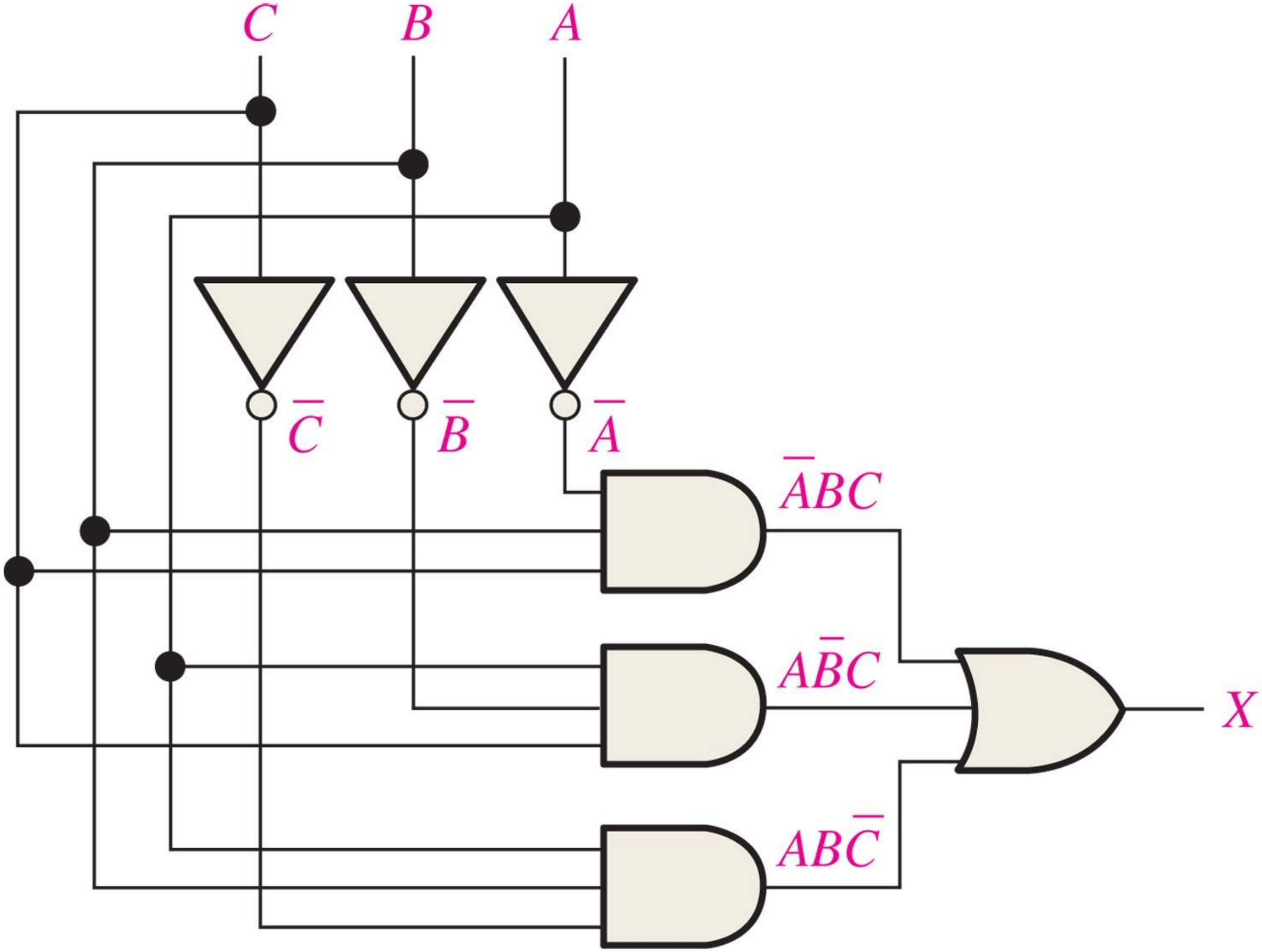

A
B
C
$\overline{A}$
$\overline{B}$
$\overline{C}$
$\overline{A}BC$
$A\overline{B}\,\overline{C}$
$X = \overline{A}BC + A\overline{B}\,\overline{C}$

TABLE 5–4

Inputs			Output	
A	*B*	*C*	*X*	**Product Term**
0	0	0	0	
0	0	1	0	
0	1	0	0	
0	1	1	1	$\overline{A}BC$
1	0	0	0	
1	0	1	1	$A\overline{B}C$
1	1	0	1	$AB\overline{C}$
1	1	1	0	

FIGURE 5-12


C
B
A
$\overline{C}$
$\overline{B}$
$\overline{A}$
$\overline{A}BC$
$A\overline{B}C$
$AB\overline{C}$
X

Develop a logic circuit with four input variables that will only produce a 1 output when exactly three input variables are 1s.

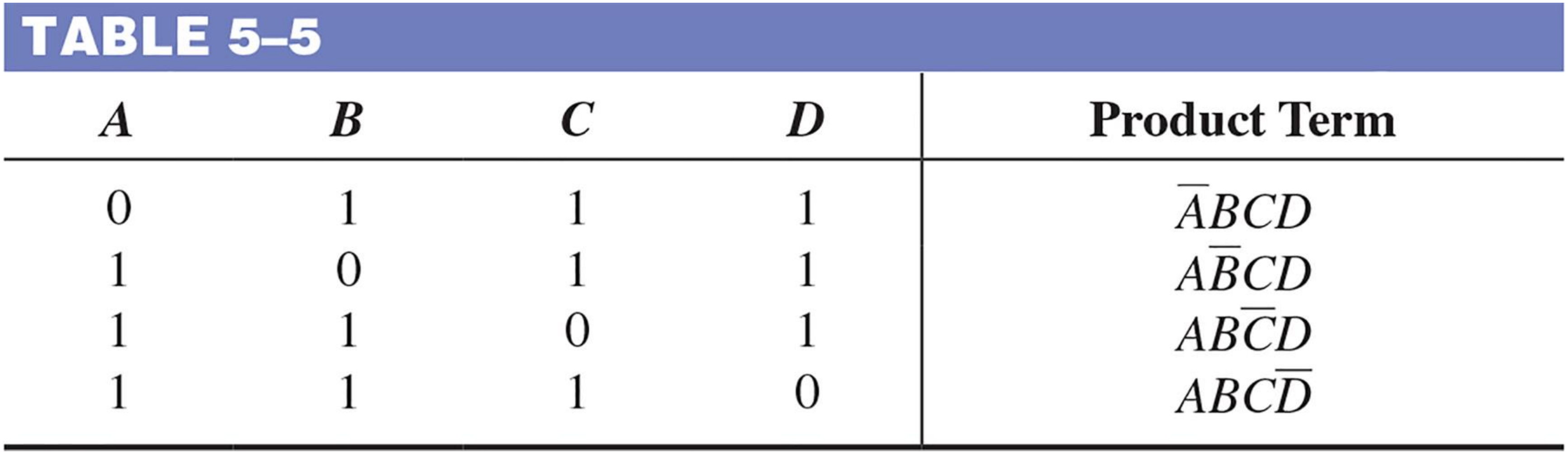


TABLE 5–5

A	B	C	D	Product Term
0	1	1	1	$\overline{A}BCD$
1	0	1	1	$A\overline{B}CD$
1	1	0	1	$AB\overline{C}D$
1	1	1	0	$ABC\overline{D}$

FIGURE 5-13

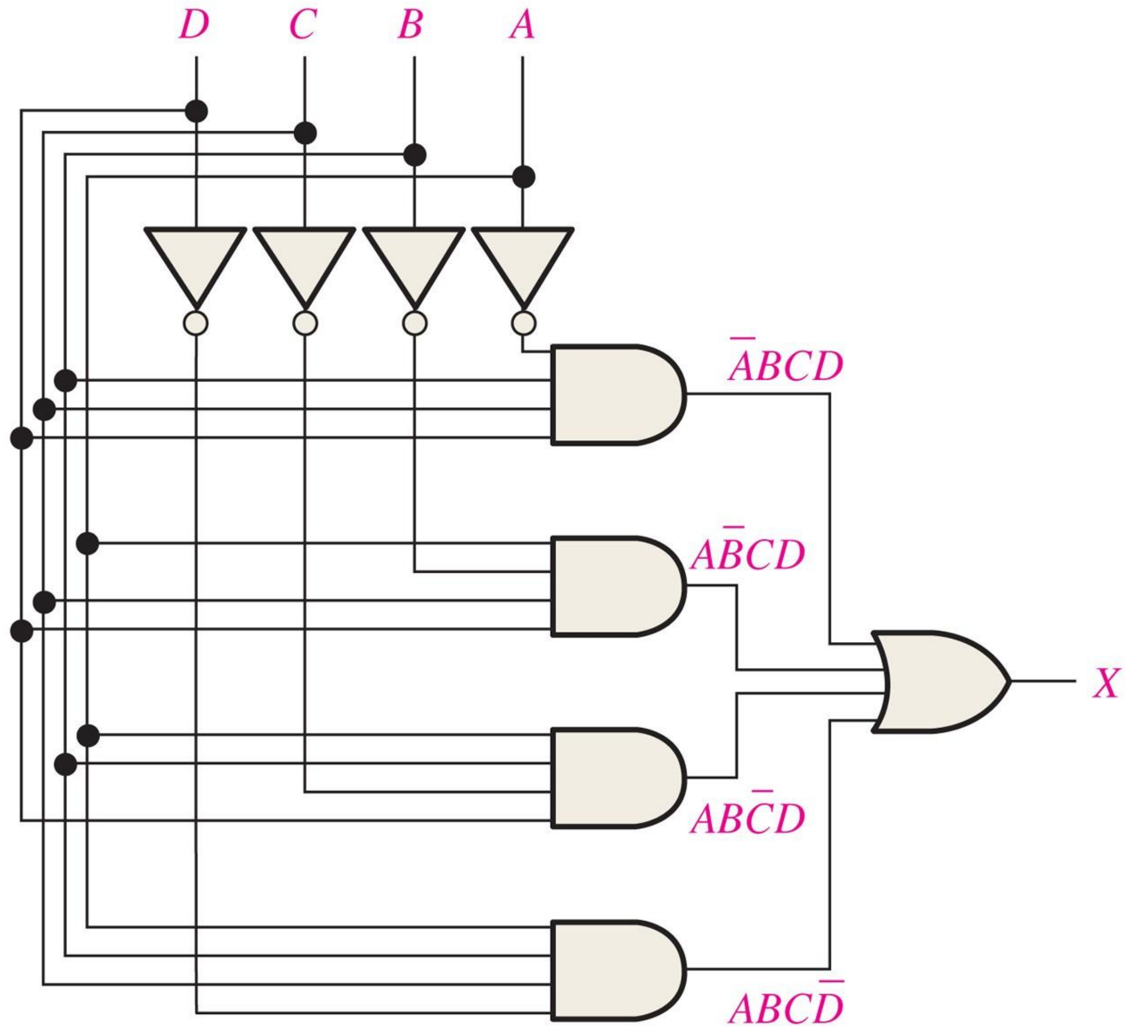
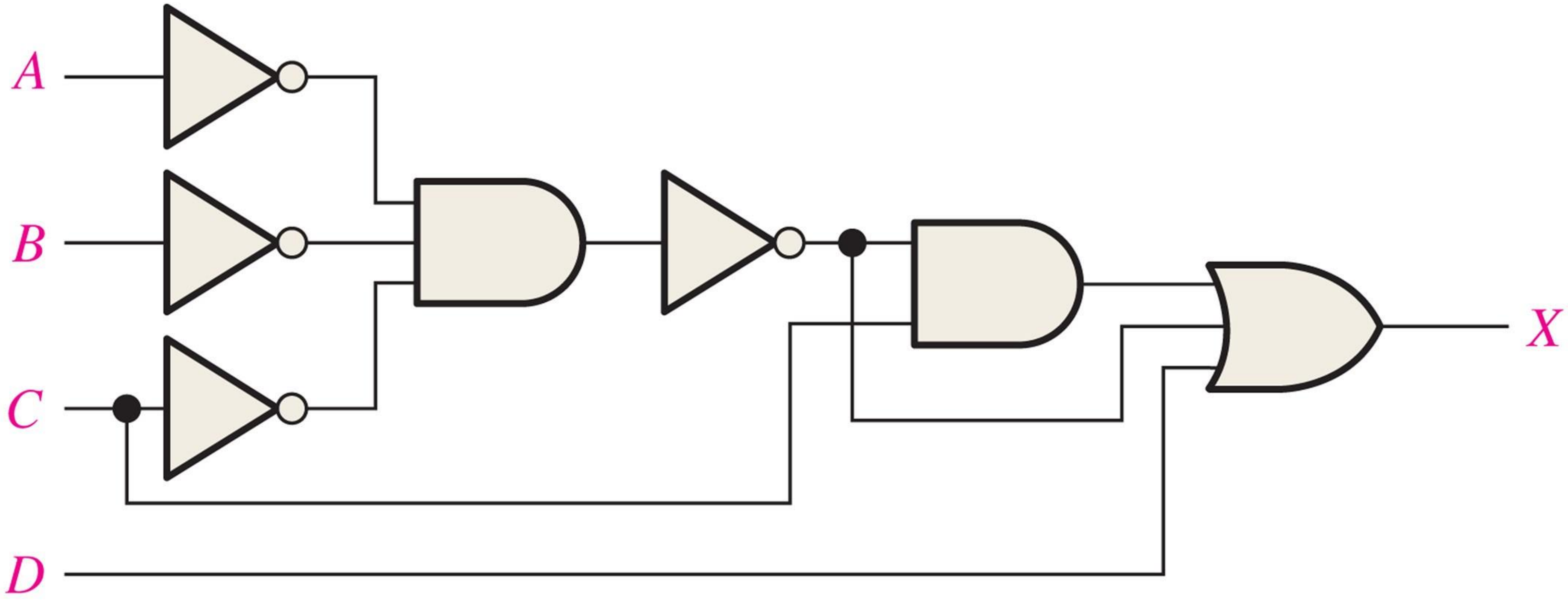

D
C
B
A
$\overline{A}BCD$
$A\overline{B}CD$
$AB\overline{C}D$
$ABC\overline{D}$
X

FIGURE 5-14

Reduce the combinational logic circuit in Figure 5–14 to a minimum form.



Solution

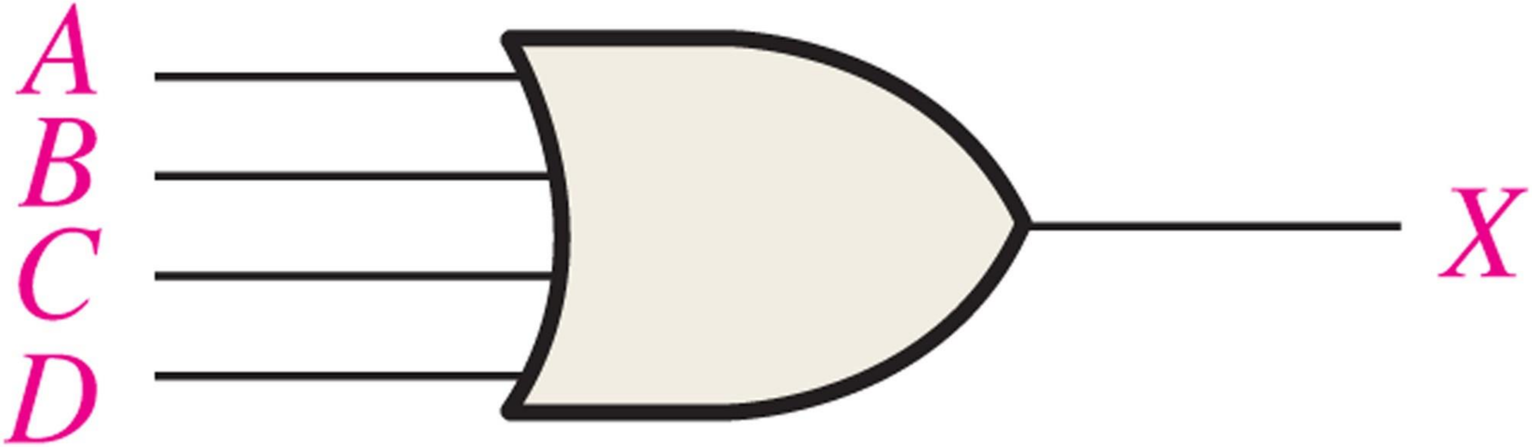
The expression for the output of the circuit is

$$X = (\overline{\overline{A}\,\overline{B}\,\overline{C}})C + \overline{\overline{A}\,\overline{B}\,\overline{C}} + D$$

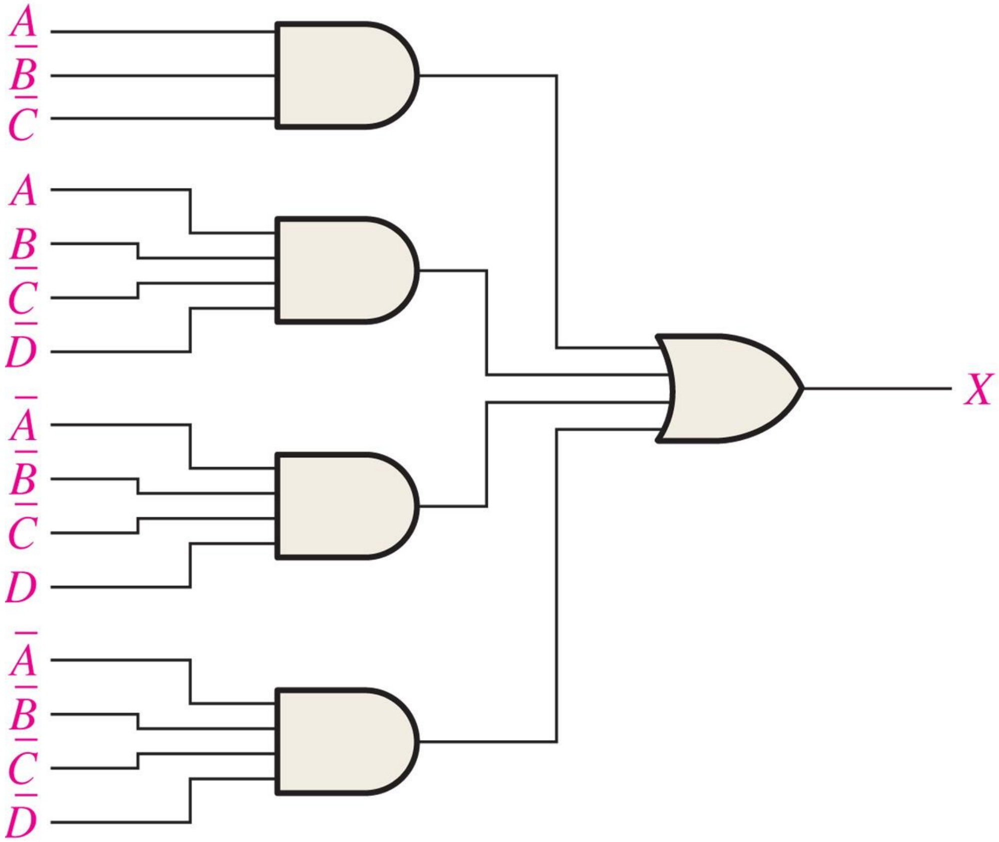
Applying DeMorgan’s theorem and Boolean algebra,

$$\begin{aligned} X &= (\overline{\overline{A}} + \overline{\overline{B}} + \overline{\overline{C}})C + \overline{\overline{A}} + \overline{\overline{B}} + \overline{\overline{C}} + D \\ &= AC + BC + CC + A + B + C + D \\ &= AC + BC + C + A + B + \cancel{C} + D \\ &= C(A + B + 1) + A + B + D \\ X &= A + B + C + D \end{aligned}$$

FIGURE 5-15



Minimize the combinational logic circuit in Figure 5–16. Inverters for the complemented variables are not shown.

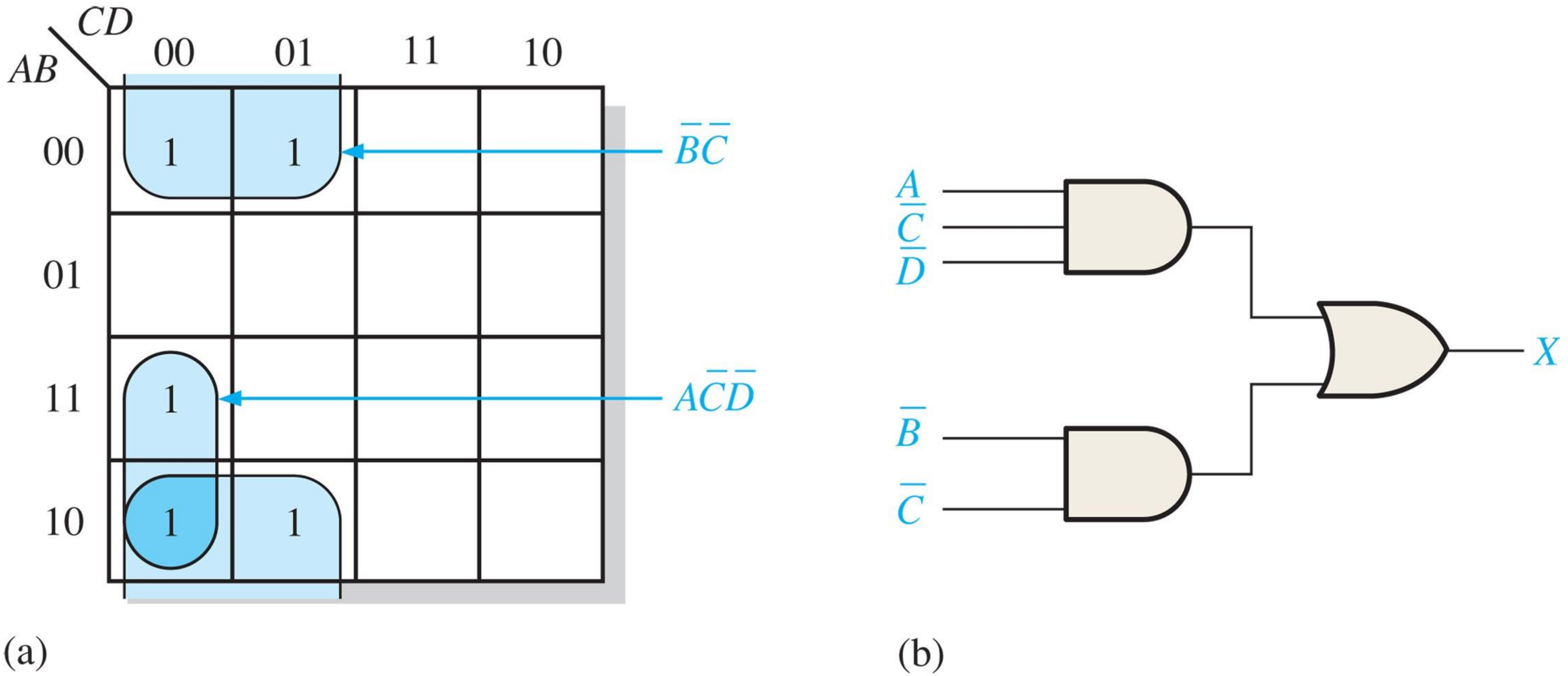


The output expression is

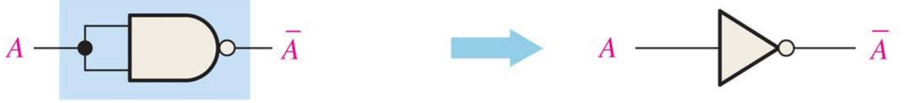
$$X = A\overline{B}\,\overline{C} + AB\overline{C}\,\overline{D} + \overline{A}\,\overline{B}\,\overline{C}D + \overline{A}\,\overline{B}\,\overline{C}\,\overline{D}$$

Expanding the first term to include the missing variables D and $\overline{D}$,

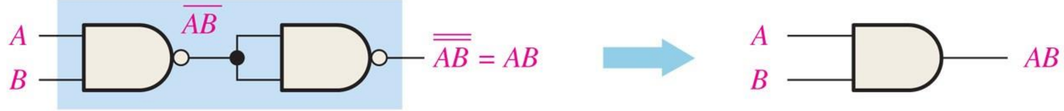
$$\begin{aligned} X &= A\overline{B}\,\overline{C}(D + \overline{D}) + AB\overline{C}\,\overline{D} + \overline{A}\,\overline{B}\,\overline{C}D + \overline{A}\,\overline{B}\,\overline{C}\,\overline{D} \\ &= A\overline{B}\,\overline{C}D + A\overline{B}\,\overline{C}\,\overline{D} + AB\overline{C}\,\overline{D} + \overline{A}\,\overline{B}\,\overline{C}D + \overline{A}\,\overline{B}\,\overline{C}\,\overline{D} \end{aligned}$$



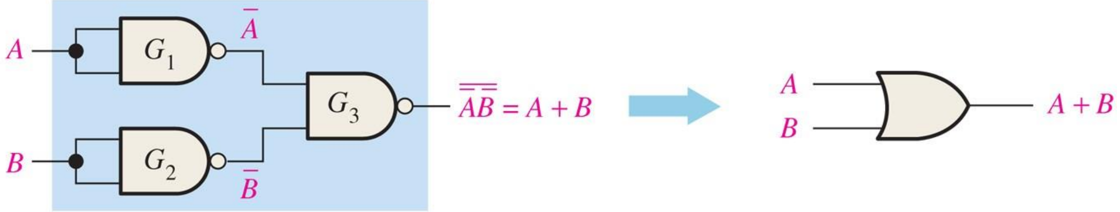
Universal application of NAND gates



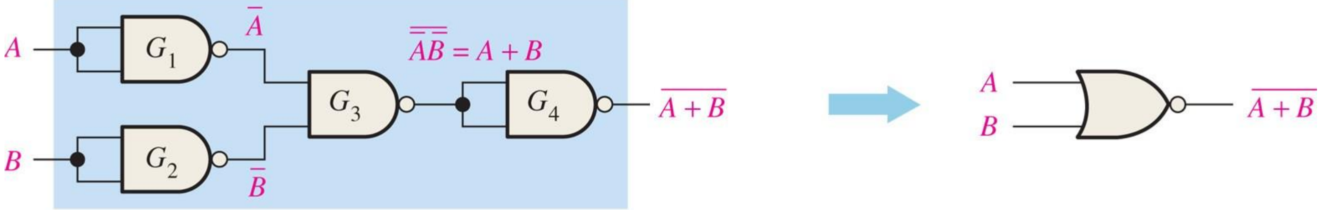
(a) One NAND gate used as an inverter



(b) Two NAND gates used as an AND gate

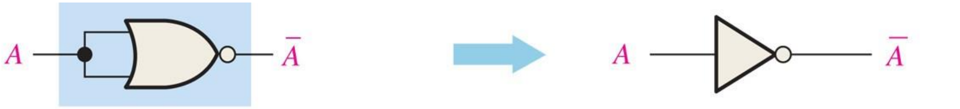


(c) Three NAND gates used as an OR gate

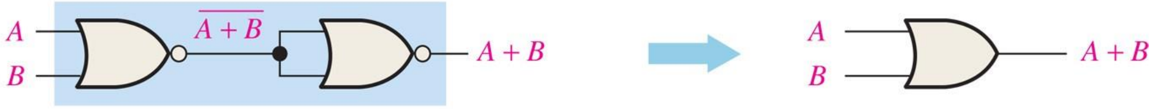


(d) Four NAND gates used as a NOR gate

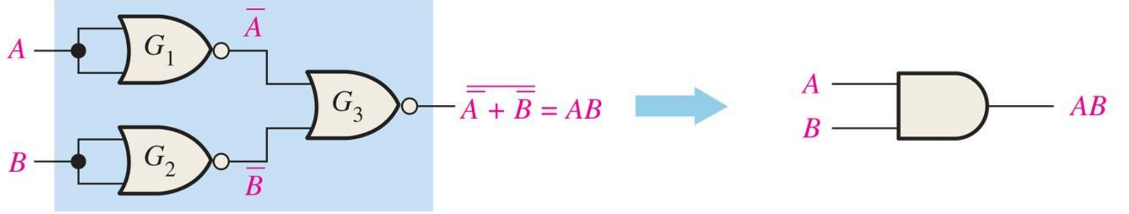
Universal application of NOR gates



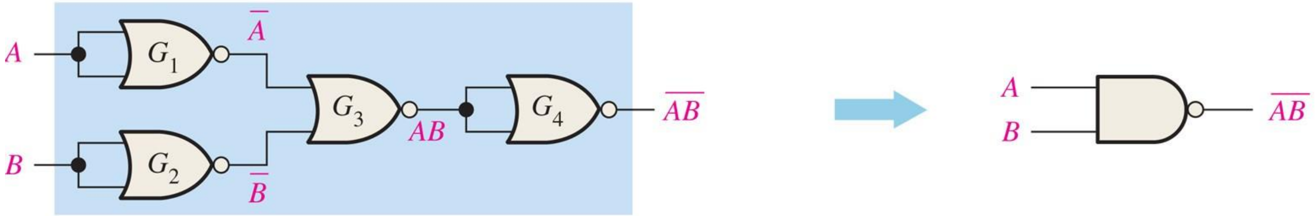
(a) One NOR gate used as an inverter



(b) Two NOR gates used as an OR gate



(c) Three NOR gates used as an AND gate

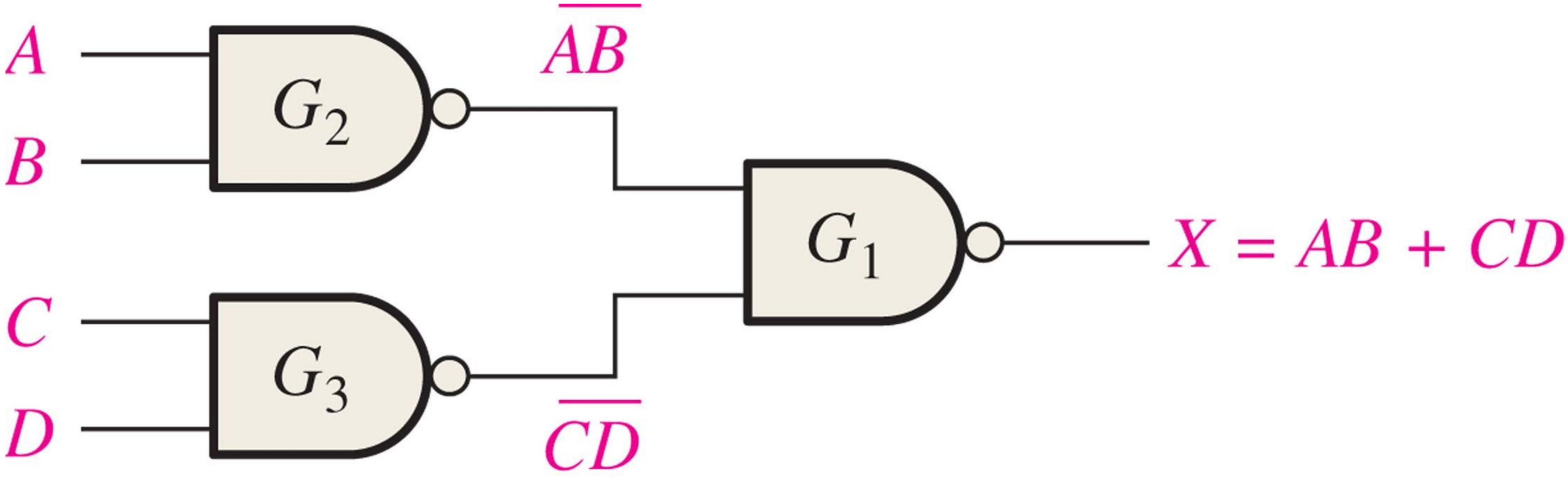


(d) Four NOR gates used as a NAND gate

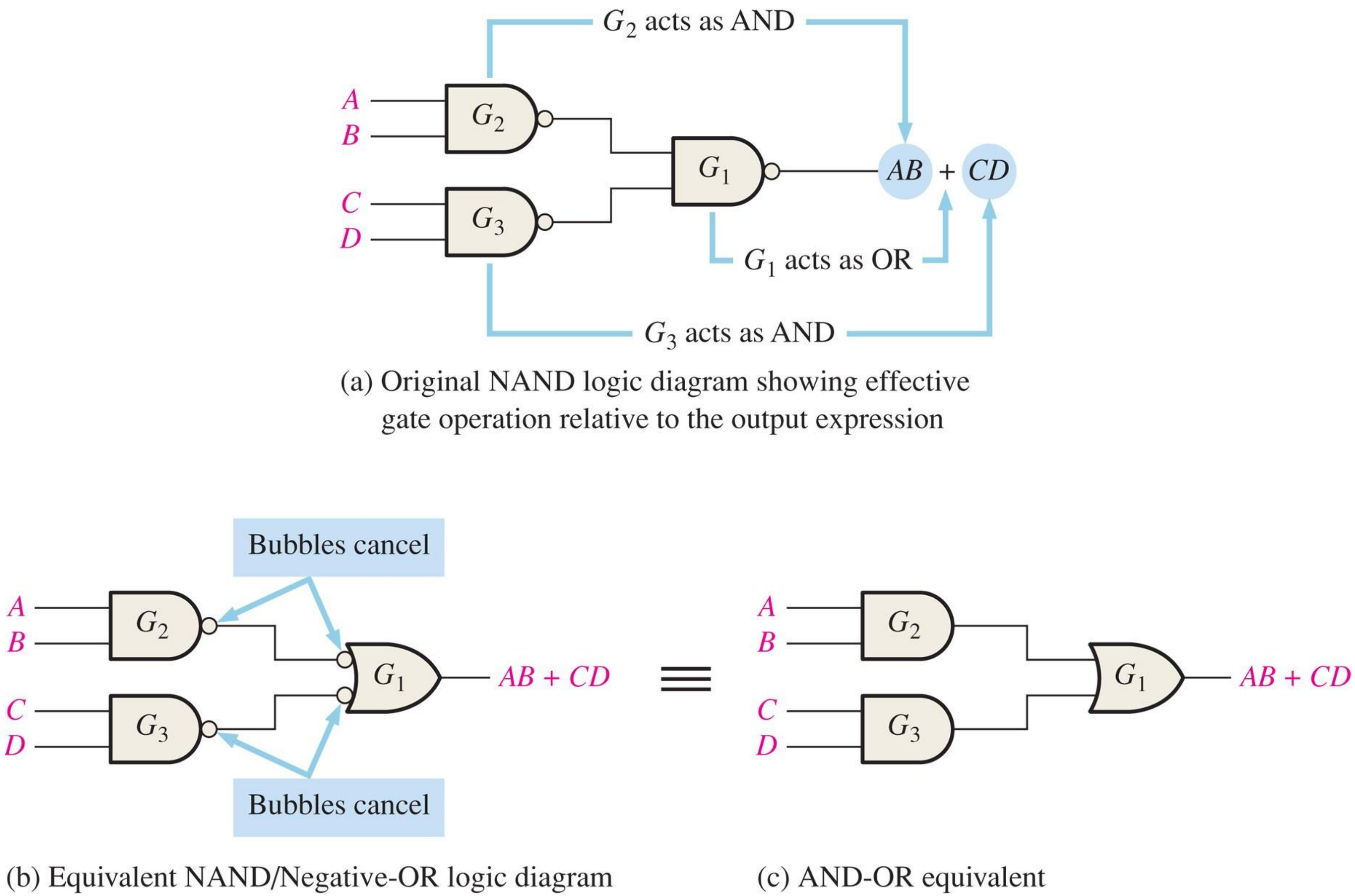
$$\underset{\text{NAND}}{\overline{AB}} = \underset{\text{negative-OR}}{\overline{A} + \overline{B}}$$

NAND logic for *X* = *AB* + *CD*

$$
\begin{aligned}
X &= \overline{(\overline{AB})(\overline{CD})} \\
&= \overline{(\overline{A} + \overline{B})(\overline{C} + \overline{D})} \\
&= (\overline{\overline{A} + \overline{B}}) + (\overline{\overline{C} + \overline{D}}) \\
&= \overline{\overline{A}}\,\overline{\overline{B}} + \overline{\overline{C}}\,\overline{\overline{D}} \\
&= AB + CD
\end{aligned}
$$



Development of the AND-OR equivalent of the circuit in Figure 5-20.

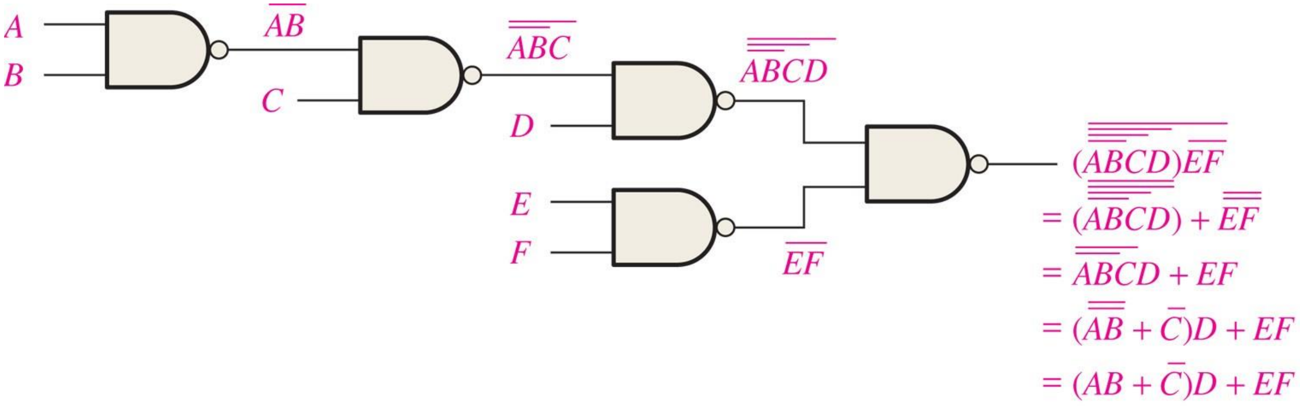


(a) Original NAND logic diagram showing effective gate operation relative to the output expression

(b) Equivalent NAND/Negative-OR logic diagram

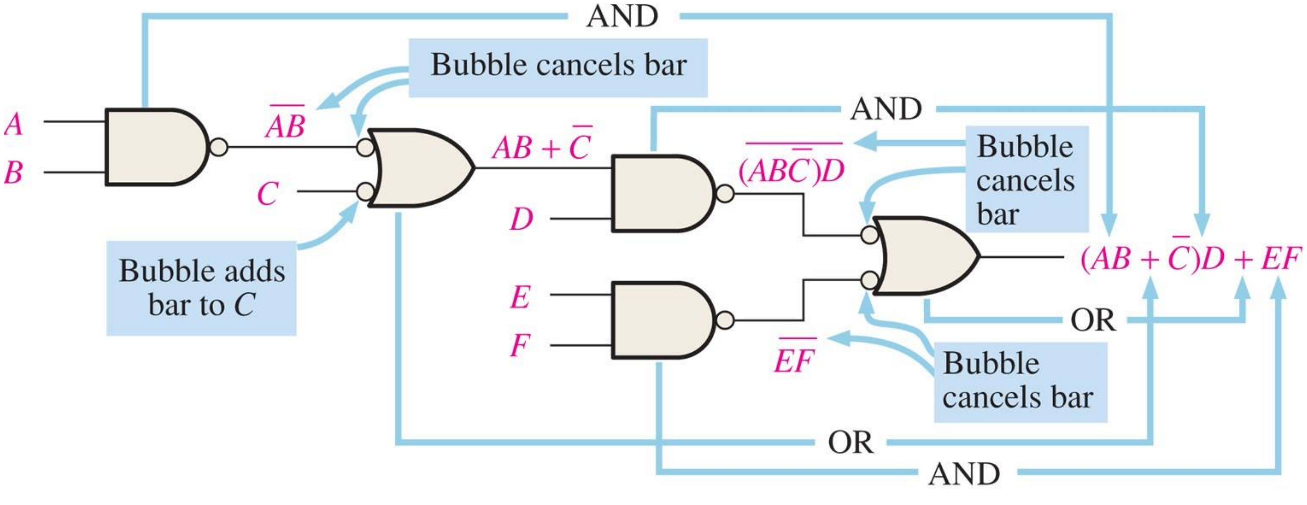
(c) AND-OR equivalent

Illustration of the use of the appropriate dual symbols in a NAND logic diagram



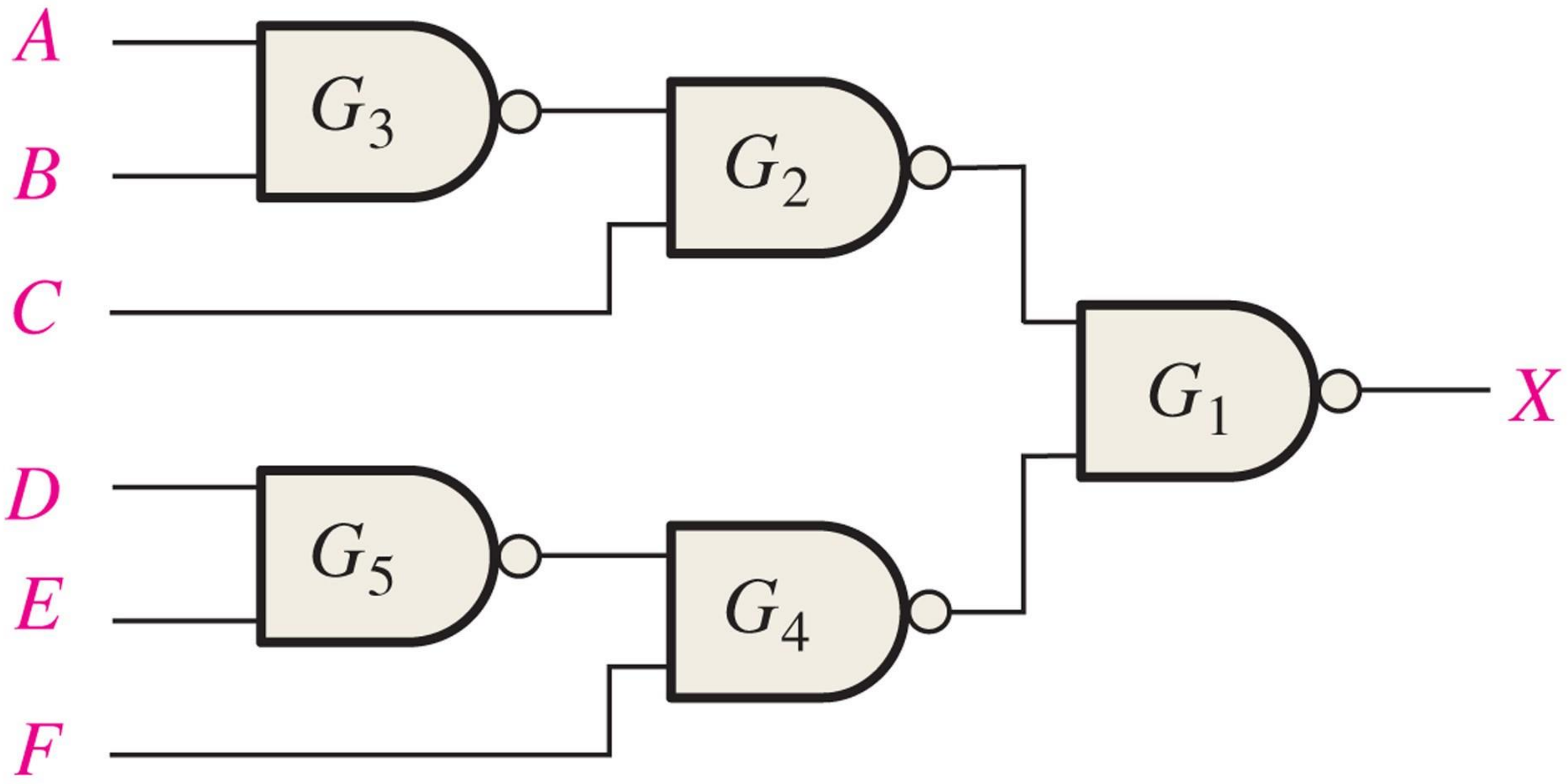
A
B
$\overline{AB}$
C
$\overline{\overline{AB}C}$
D
$\overline{\overline{\overline{AB}C}D}$
E
F
$\overline{EF}$
$\overline{(\overline{\overline{\overline{AB}C}D})\overline{EF}}$
$= (\overline{\overline{\overline{\overline{AB}C}D}}) + \overline{\overline{EF}}$
$= \overline{\overline{\overline{AB}C}}D + EF$
$= (\overline{\overline{AB}} + \overline{C})D + EF$
$= (AB + \overline{C})D + EF$

(a) Several Boolean steps are required to arrive at final output expression.

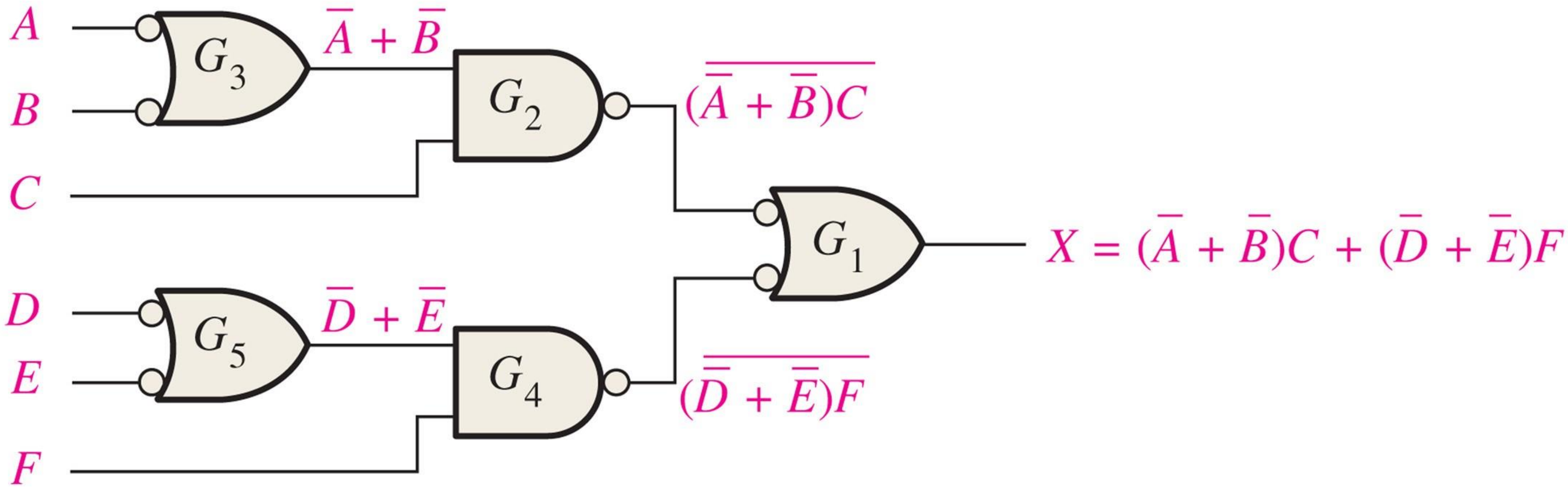


(b) Output expression can be obtained directly from the function of each gate symbol in the diagram.

FIGURE 5-23

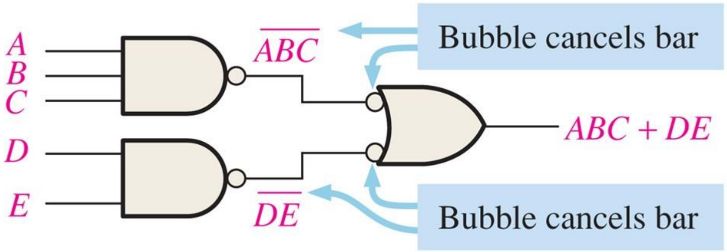


Redraw the logic diagram and develop the output expression for the circuit in Figure 5–23 using the appropriate dual symbols.


A
B
C
D
E
F
G_3
G_2
G_1
G_5
G_4
$\bar{A} + \bar{B}$
$\overline{(\bar{A} + \bar{B})C}$
$X = (\bar{A} + \bar{B})C + (\bar{D} + \bar{E})F$
$\bar{D} + \bar{E}$
$\overline{(\bar{D} + \bar{E})F}$

Implement each expression with NAND logic using appropriate dual symbols:

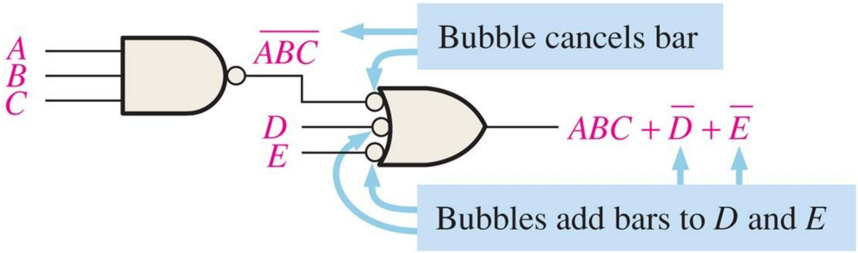
(a) $ABC + DE$ **(b)** $ABC + \overline{D} + \overline{E}$



Bubble cancels bar

Bubble cancels bar

(a)



Bubble cancels bar

Bubbles add bars to D and E

(b)

NOR Logic

A NOR gate can function as either a NOR or a **negative-AND**, as shown by DeMorgan's theorem.

$$\underset{\uparrow\ \text{NOR}}{\overline{A + B}} = \underset{\uparrow\ \text{negative-AND}}{\overline{A}\,\overline{B}}$$

FIGURE 5-26 NOR logic for $X = (A + B)(C + D)$.

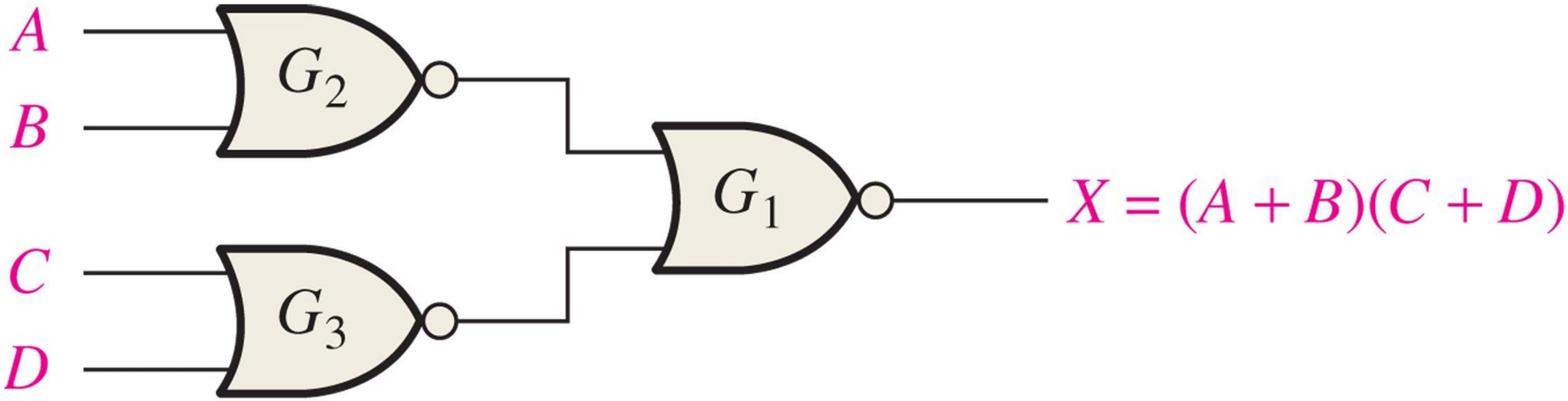


FIGURE 5-27

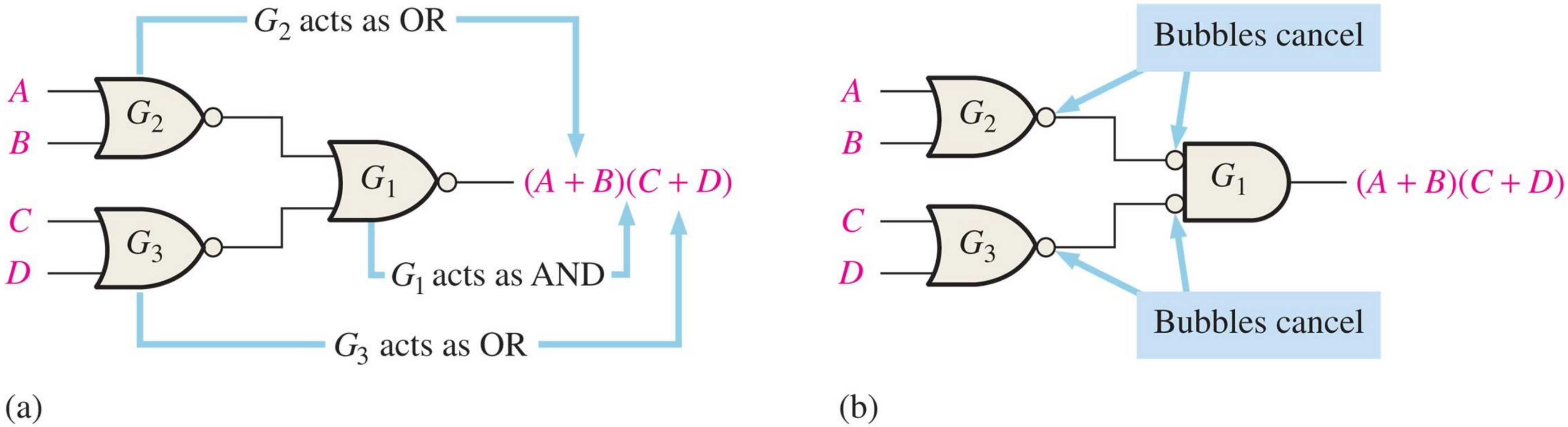
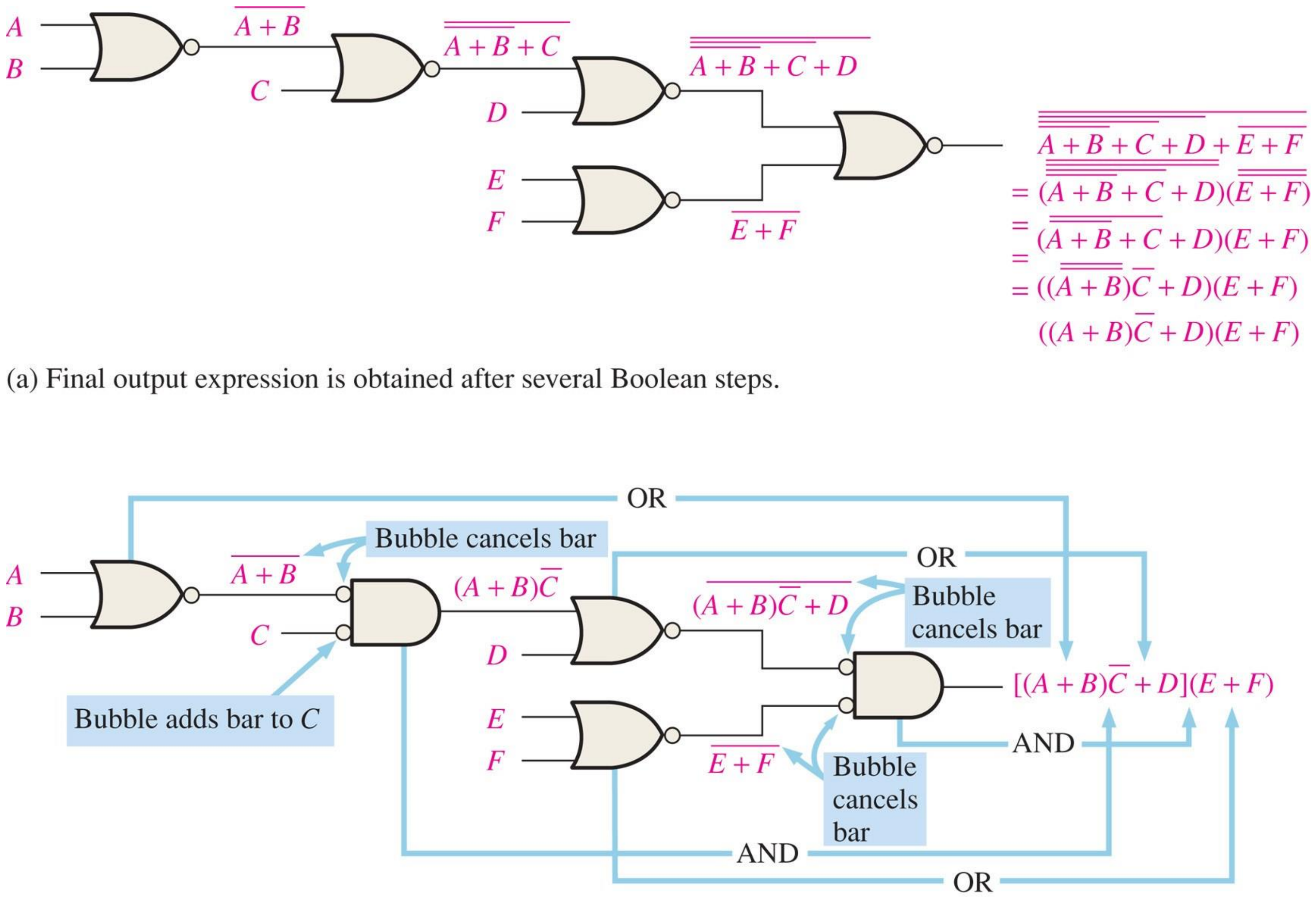

G_2 acts as OR
A
B
G_2
G_1
$(A + B)(C + D)$
C
D
G_3
G_1 acts as AND
G_3 acts as OR
(a)
Bubbles cancel
A
B
G_2
G_1
$(A + B)(C + D)$
C
D
G_3
Bubbles cancel
(b)

FIGURE 5-28 Illustration of the use of the appropriate dual symbols in a NOR logic diagram.


A
B
$\overline{A+B}$
C
$\overline{\overline{A+B}+C}$
D
$\overline{\overline{\overline{A+B}+C}+D}$
E
F
$\overline{E+F}$
$\overline{\overline{\overline{\overline{A+B}+C}+D}+\overline{E+F}}$
$= \overline{\overline{\overline{\overline{A+B}+C}+D}}\,\overline{\overline{E+F}}$
$= (\overline{\overline{A+B}+C}+D)(E+F)$
$= ((\overline{\overline{A+B}})\overline{C}+D)(E+F)$
$((A+B)\overline{C}+D)(E+F)$

(a) Final output expression is obtained after several Boolean steps.

OR
Bubble cancels bar
A
B
$\overline{A+B}$
$(A+B)\overline{C}$
C
Bubble adds bar to C
D
$\overline{(A+B)\overline{C}+D}$
OR
Bubble cancels bar
E
F
$\overline{E+F}$
Bubble cancels bar
$[(A+B)\overline{C}+D](E+F)$
AND
AND
OR

(b) Output expression can be obtained directly from the function of each gate symbol in the diagram.

FIGURE 5-29

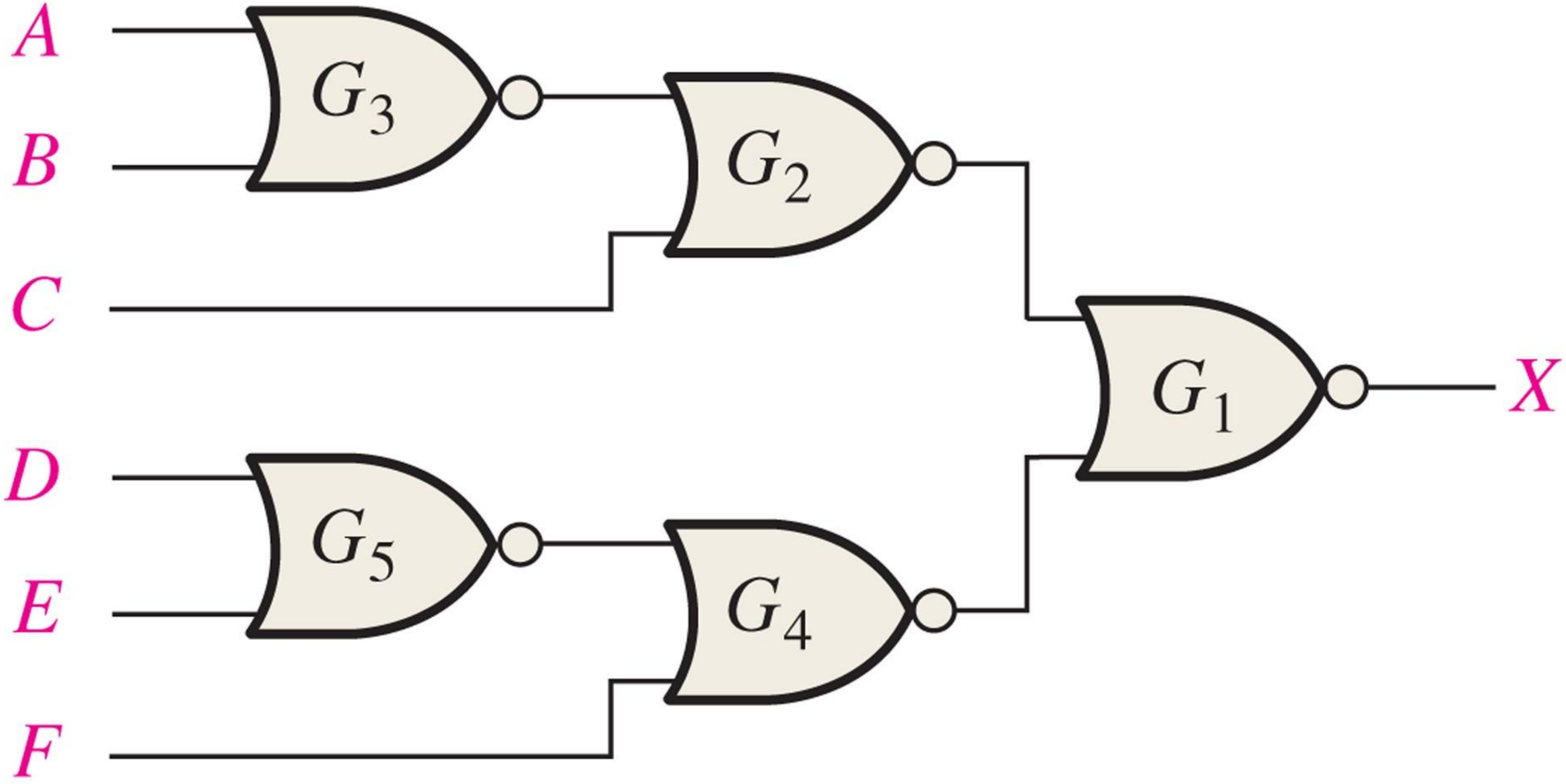


FIGURE 5-30

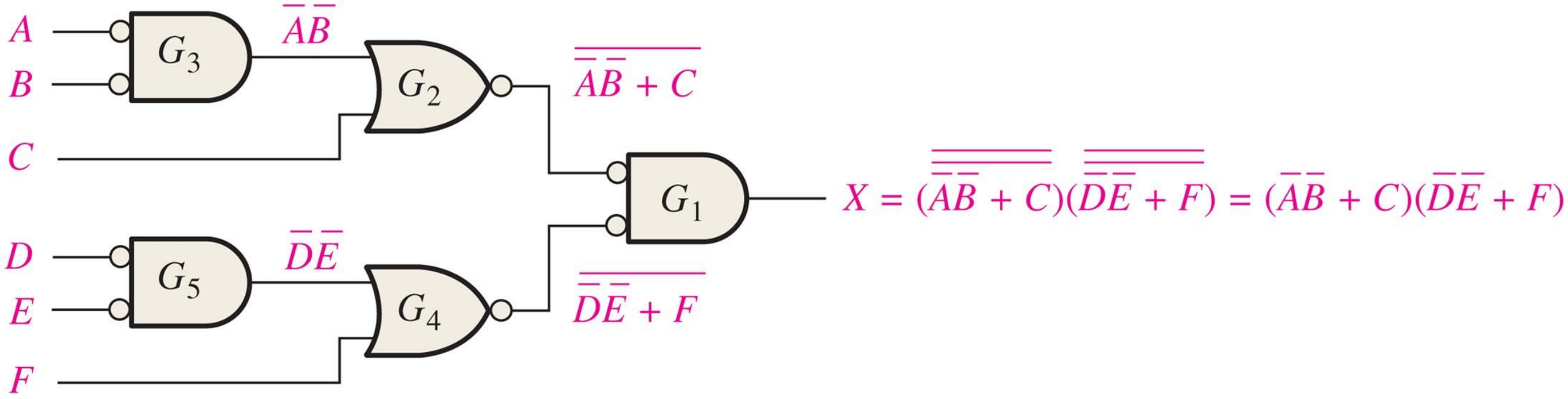
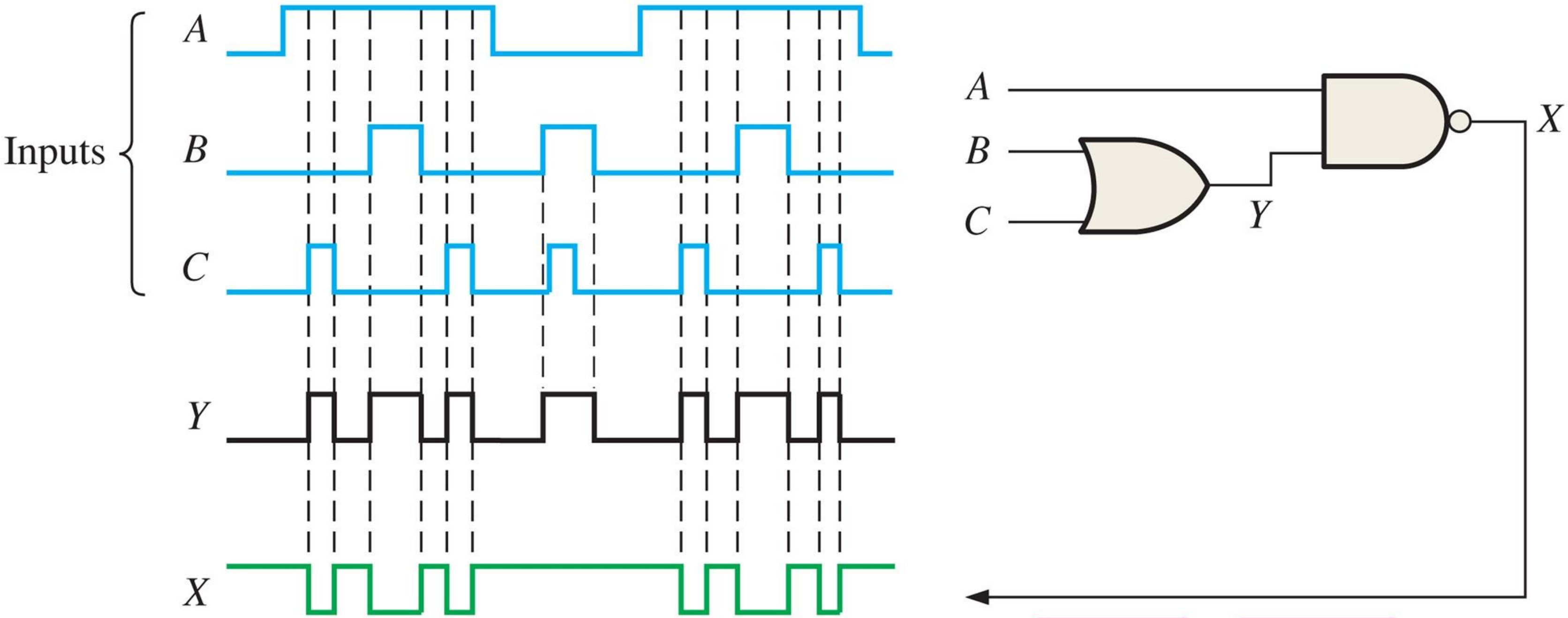

A
B
C
D
E
F
G_3
G_2
G_5
G_4
G_1
$\overline{A}\,\overline{B}$
$\overline{\overline{A}\,\overline{B} + C}$
$\overline{D}\,\overline{E}$
$\overline{\overline{D}\,\overline{E} + F}$
$X = (\overline{\overline{\overline{A}\,\overline{B} + C}})(\overline{\overline{\overline{D}\,\overline{E} + F}}) = (\overline{A}\,\overline{B} + C)(\overline{D}\,\overline{E} + F)$

FIGURE 5-31



$X = \overline{A(B + C)} = \overline{AB + AC}$

FIGURE 5-32

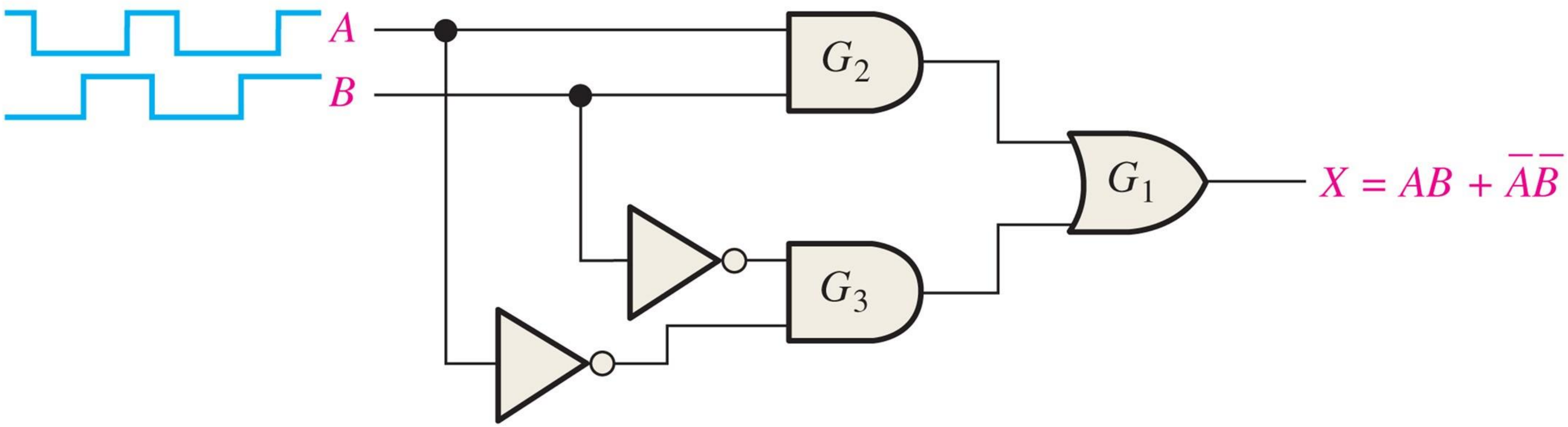


FIGURE 5-33

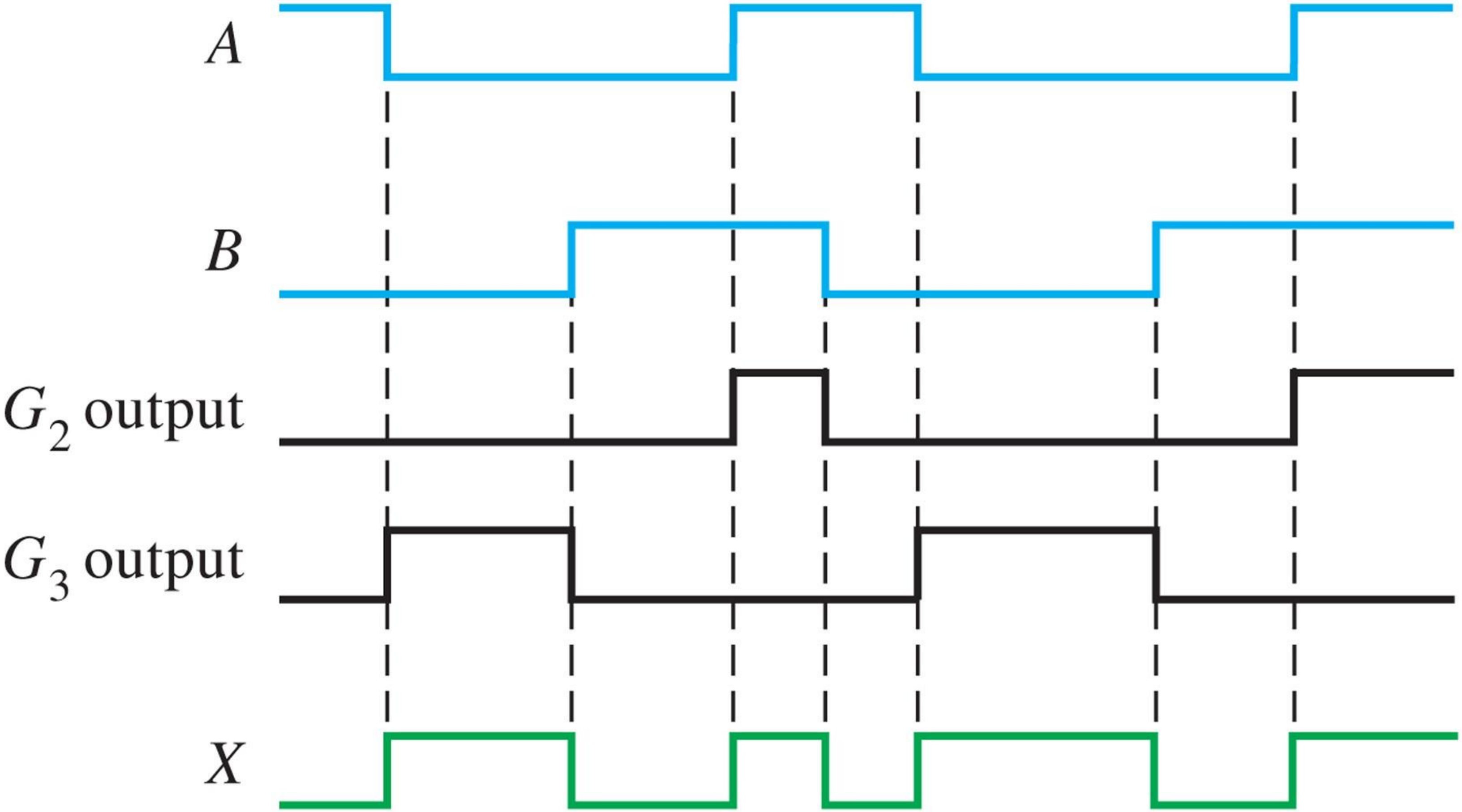


FIGURE 5-34

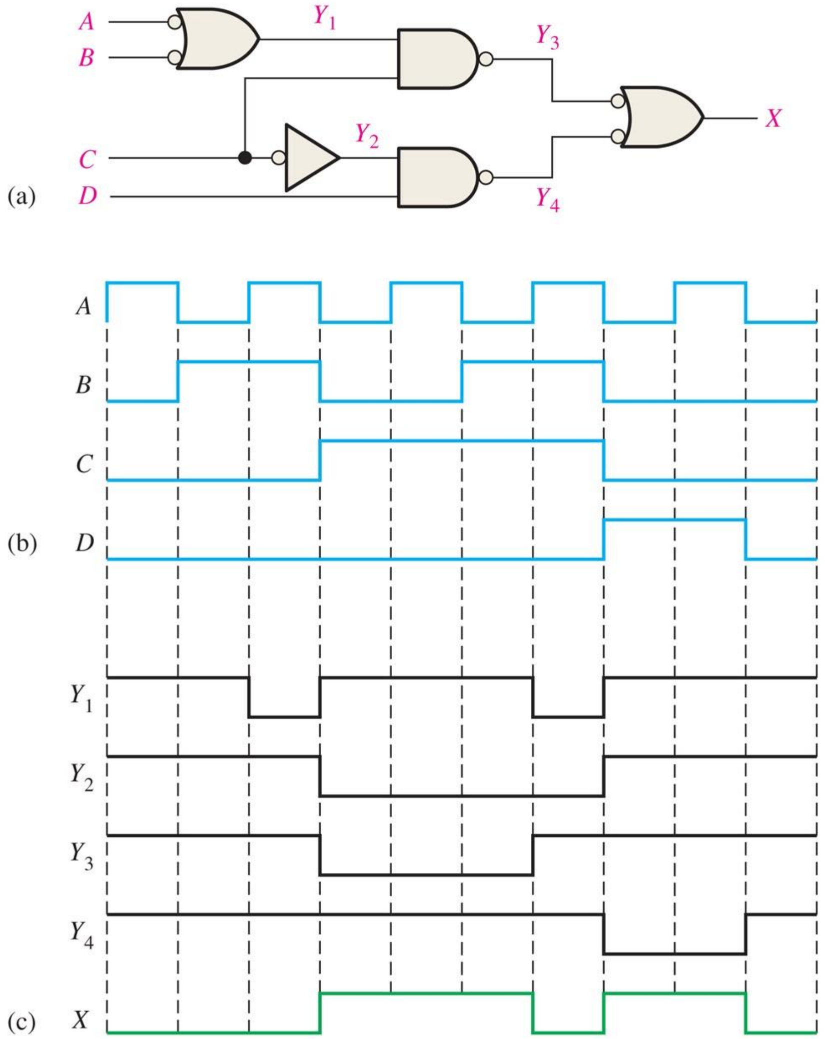


FIGURE 5-35

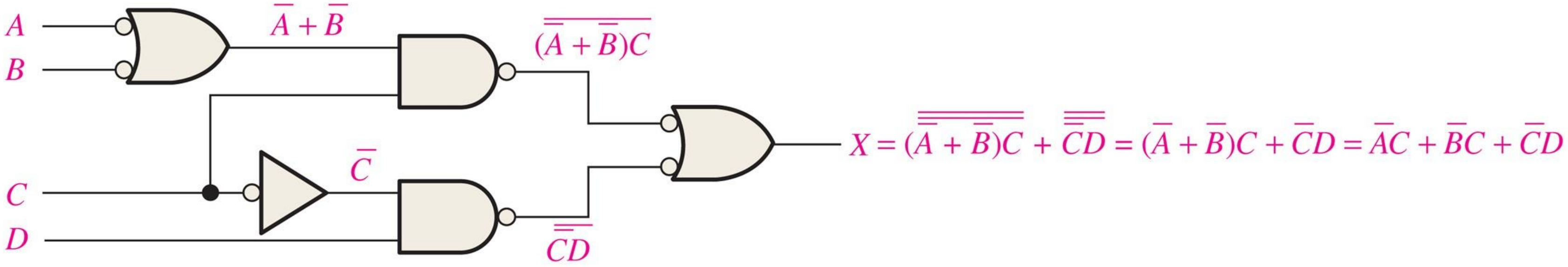
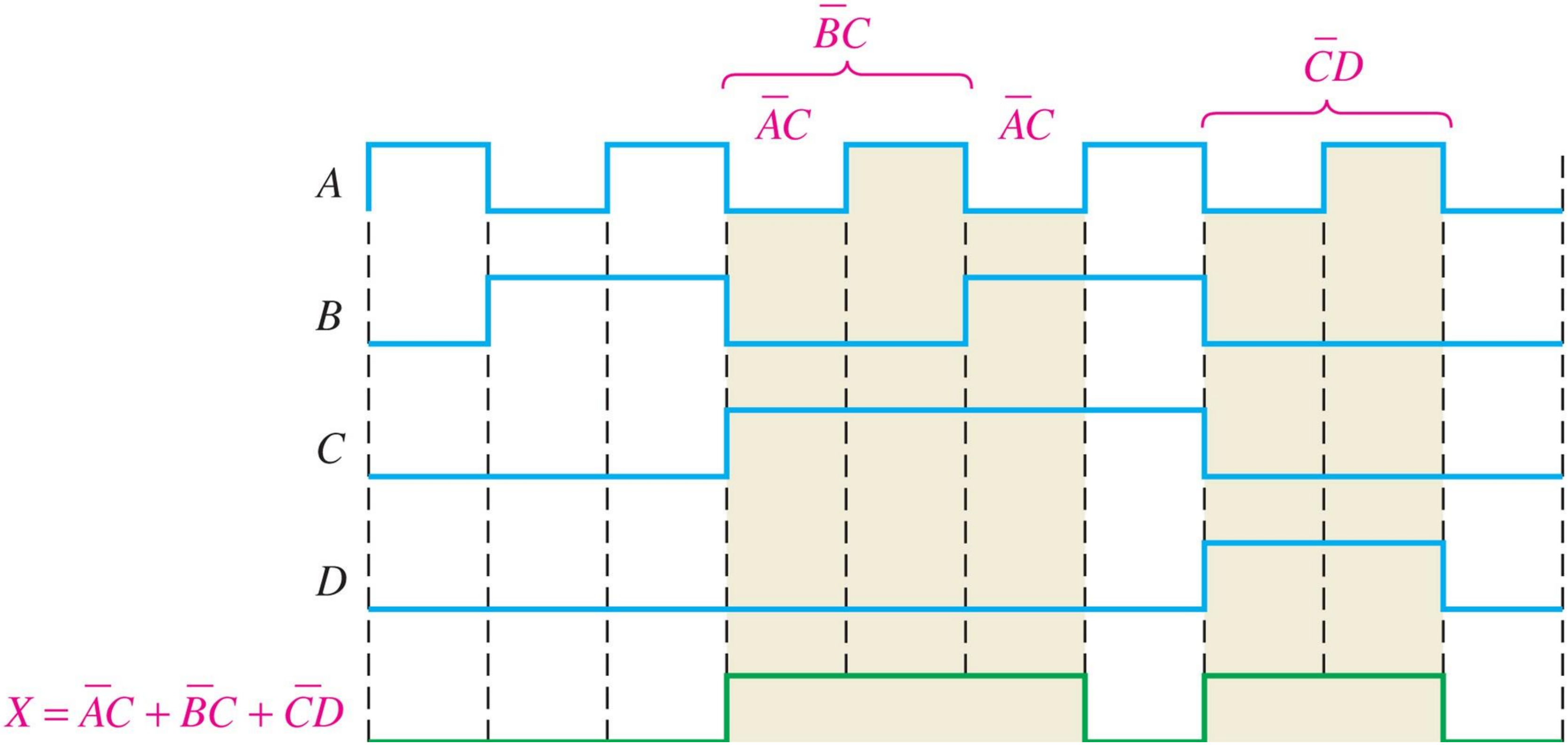

A
B
$\overline{A}+\overline{B}$
$\overline{(\overline{A}+\overline{B})C}$
C
$\overline{C}$
D
$\overline{\overline{C}D}$
$X=\overline{\overline{(\overline{A}+\overline{B})C}}+\overline{\overline{\overline{C}D}}=(\overline{A}+\overline{B})C+\overline{C}D=\overline{A}C+\overline{B}C+\overline{C}D$

FIGURE 5-36


$\overline{B}C$
$\overline{C}D$
$\overline{A}C$
$\overline{A}C$
A
B
C
D
$X = \overline{A}C + \overline{B}C + \overline{C}D$