جامعة فلسطين التقنية - خضوري
Palestine Technical University - Kadoorie

Digital Electronics and Logic Design

Combinational Logic Analysis

Dr. Jafar Saifeddin Jallad

Dept. of Electrical Engineering

Palestine Technical University

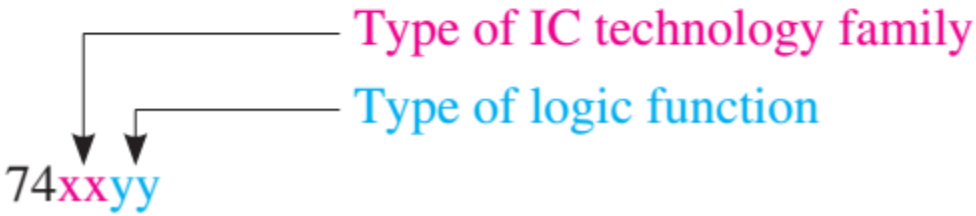
Tulkaram, Palestine

Fixed-Function Logic Gates

All of the various fixed-function logic devices currently available are implemented in two major categories of circuit technology: **CMOS** (complementary metal-oxide semiconductor) and **bipolar** (also known as **TTL**, transistor-transistor logic). A type of bipolar technology that is available in very limited devices is ECL (emitter-coupled logic). BiCMOS is another integrated circuit technology that combines both bipolar and CMOS. CMOS is the most dominant circuit technology.

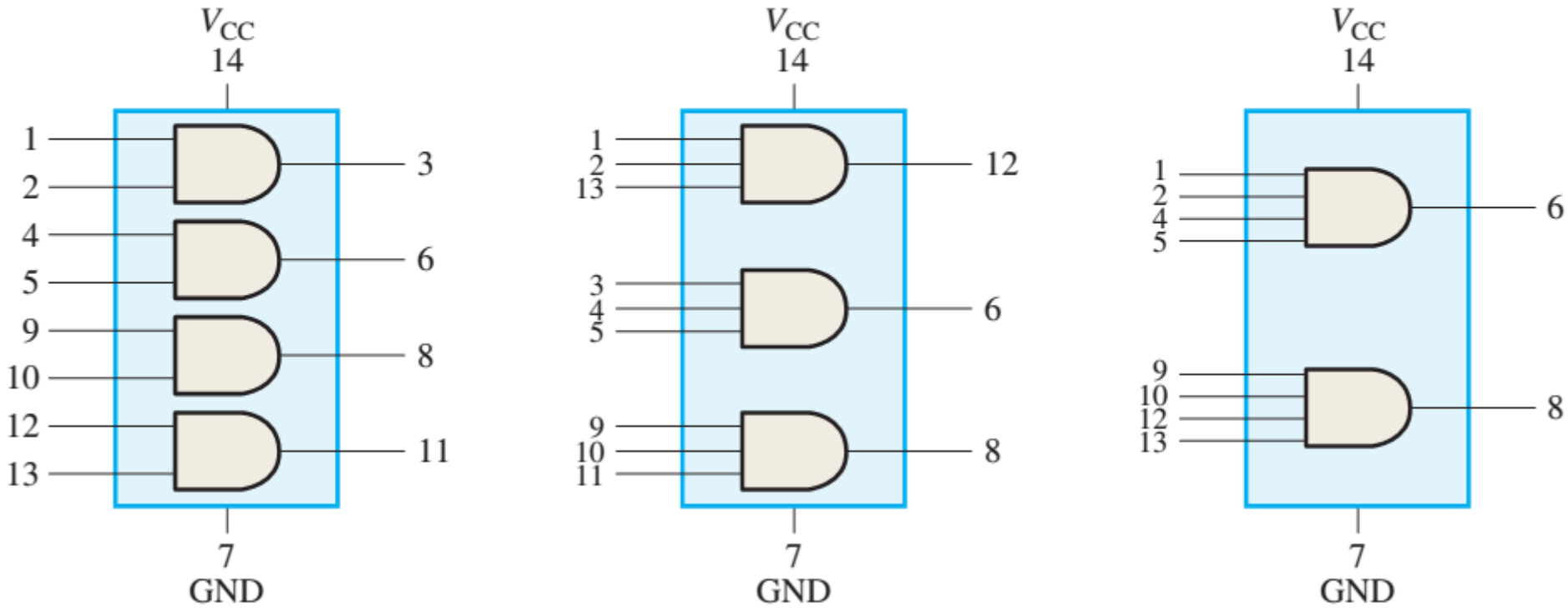
Series Logic Gate Functions

The 74 series is the standard fixed-function logic devices. The device label format includes one or more letters that indentify the type of logic circuit technology family in the IC package and two or more digits that identify the type of logic function. For example, 74HC04 is a fixed-function IC that has six inverters in a package as indicated by 04. The letters, HC, following the prefix 74 identify the circuit technology family as a type of CMOS logic.



AND Gate

Figure shows three configurations of fixed-function AND gates in the 74 series. The 74xx08 is a quad 2-input AND gate device, the 74xx11 is a triple 3-input AND gate device,



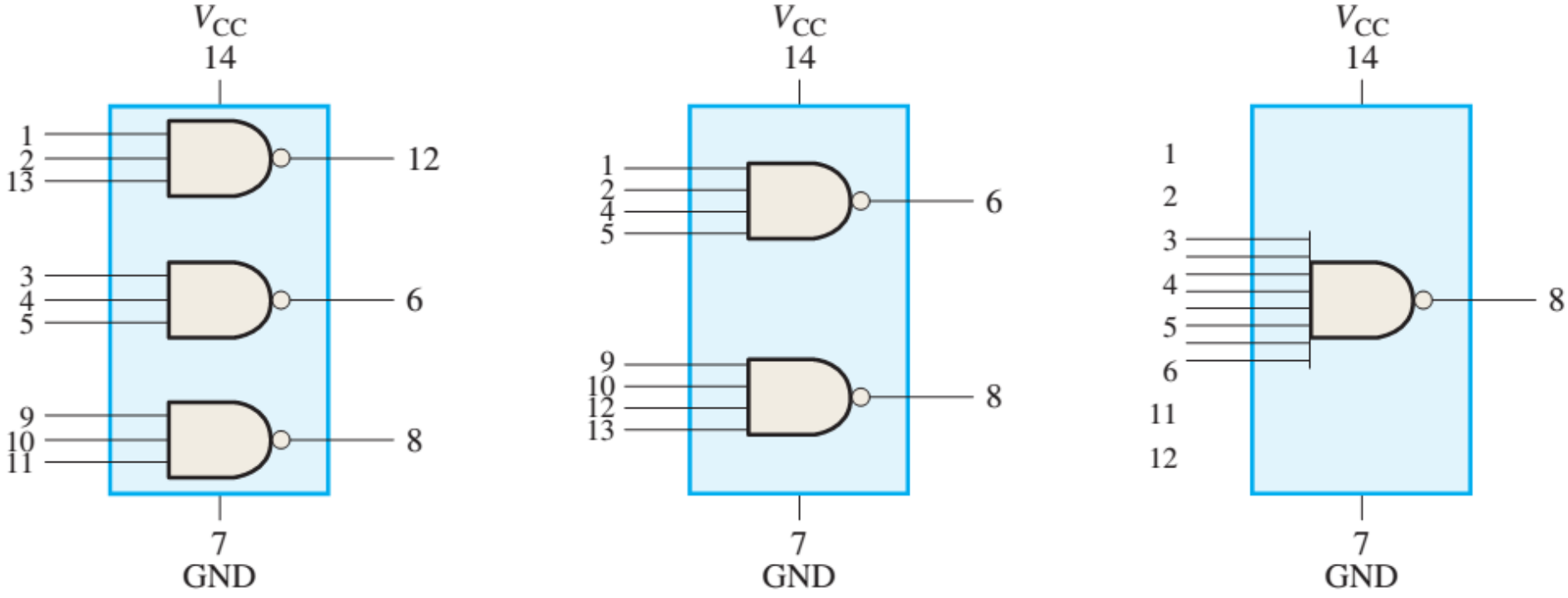
(a) 74xx08 (b) 74xx11 (c) 74xx21

74 series AND gate devices with pin numbers.

and the 74xx21 is a dual 4-input AND gate device. The label xx can represent any of the integrated circuit technology families such as HC or LS. The numbers on the inputs and outputs are the IC package pin numbers.

NAND Gate

Figure shows four configurations of fixed-function NAND gates in the 74 series. The 74xx00 is a quad 2-input NAND gate device, the 74xx10 is a triple 3-input NAND gate device, the 74xx20 is a dual 4-input NAND gate device, and the 74xx30 is a single 8-input NAND gate device.

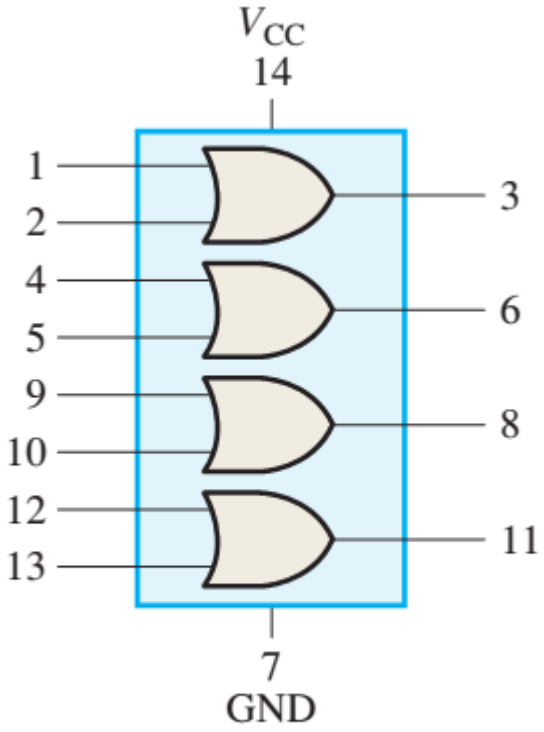


(b) 74xx10 (c) 74xx20 (d) 74xx30

74 series NAND gate devices with package pin numbers.

OR Gate

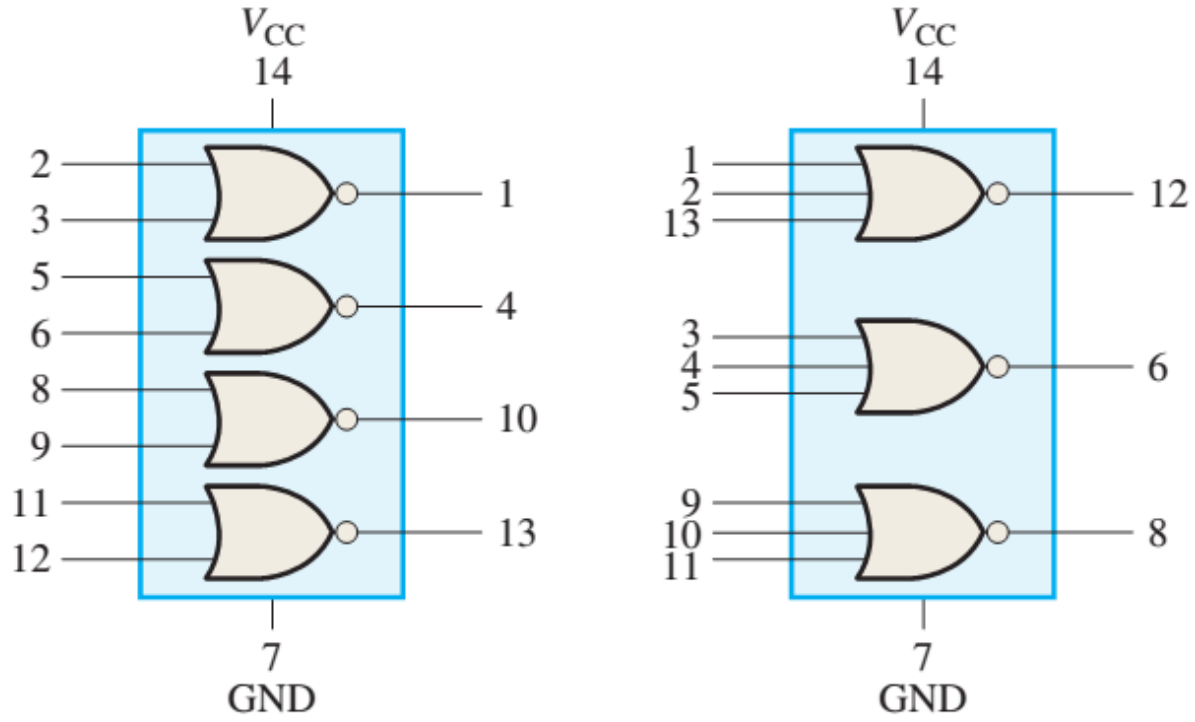
Figure shows a fixed-function OR gate in the 74 series. The 74xx32 is a quad 2-input OR gate device.



74xx32

NOR Gate

Figure shows two configurations of fixed-function NOR gates in the 74 series. The 74xx02 is a quad 2-input NOR gate device, and the 74xx27 is a triple 3-input NOR gate device.

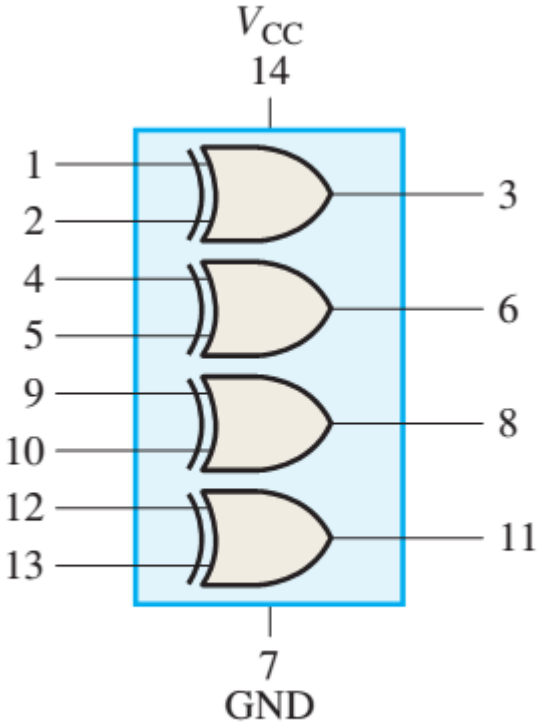


(a) 74xx02 (b) 74xx27

74 series NOR gate devices.

XOR Gate

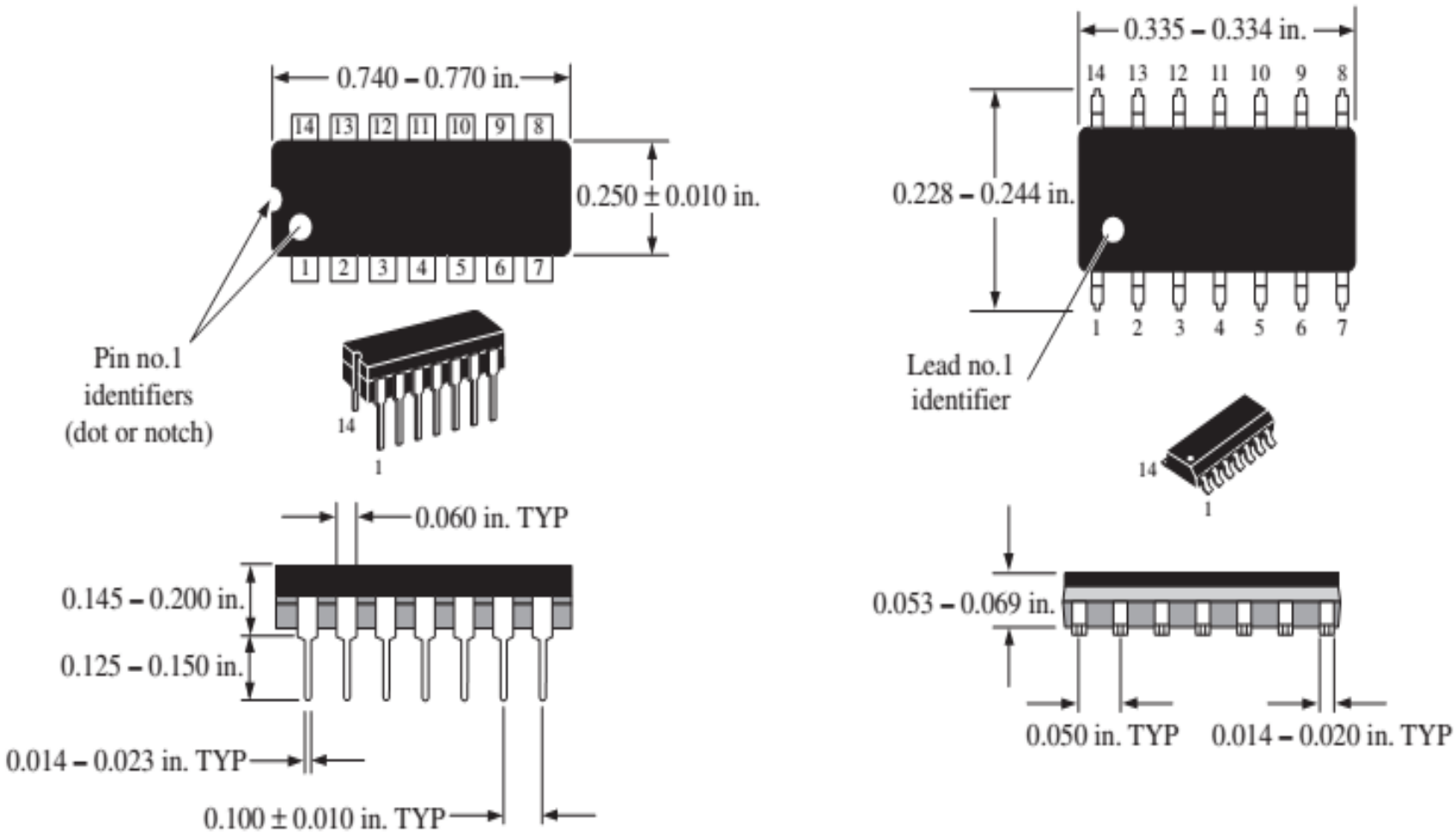
Figure shows a fixed-function XOR (exclusive-OR) gate in the 74 series. The 74xx86 is a quad 2-input XOR gate.



74xx86

Two major fixed function logic families are TTL and CMOS. A third technology is BiCMOS, which combines the first two. Packaging for fixed function logic is shown.

IC Packages

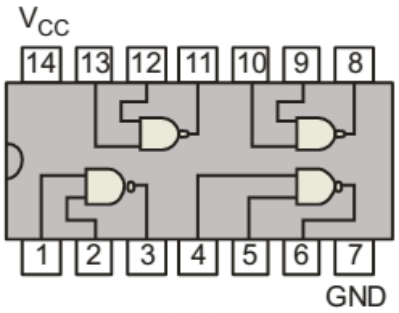


(a) 14-pin dual in-line package (DIP) for feedthrough mounting

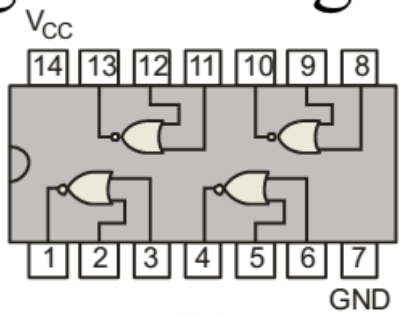
(b) 14-pin small outline package (SOIC) for surface mounting

Typical dual in-line (DIP) and small-outline (SOIC) packages showing pin numbers and basic dimensions.

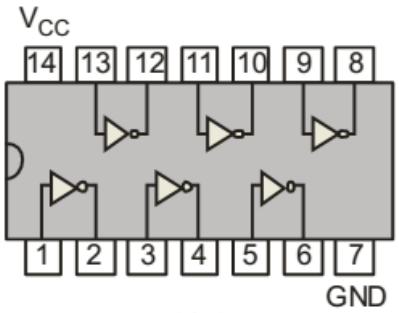
Some common gate configurations are shown.



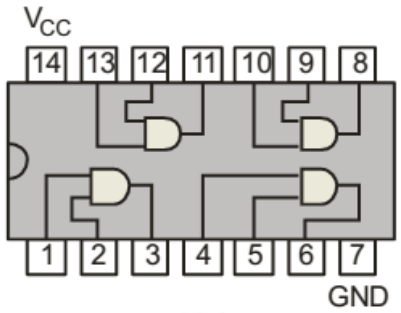
'00



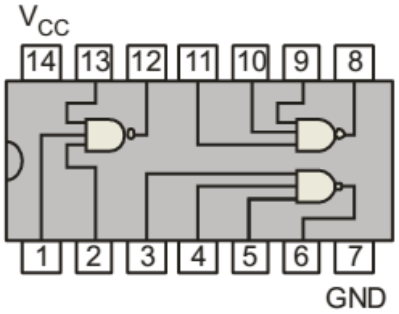
'02



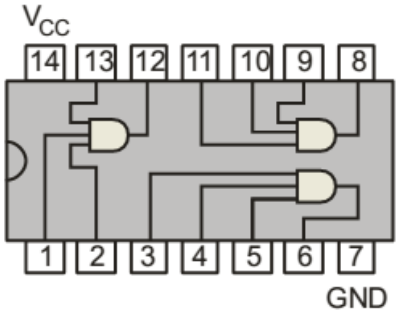
'04



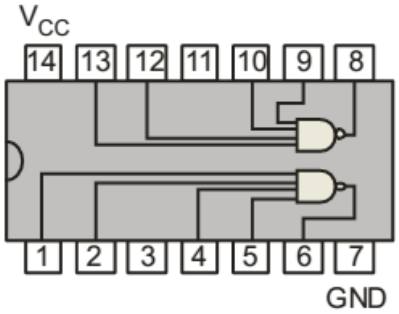
'08



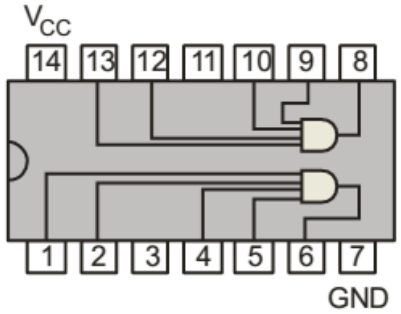
'10



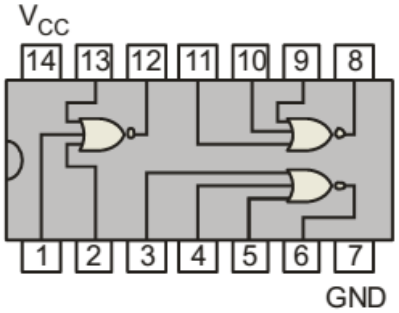
'11



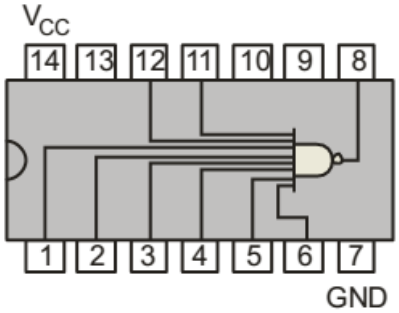
'20



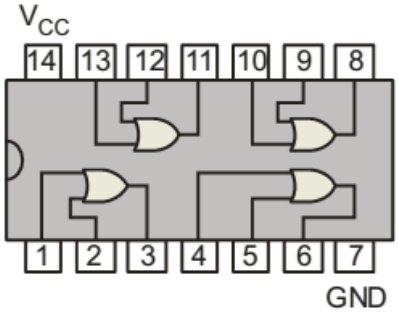
'21



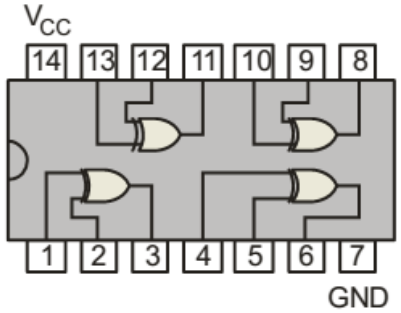
'27



'30



'32

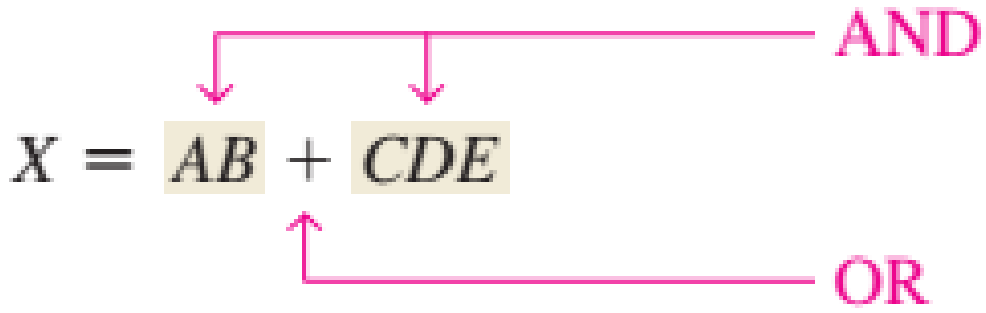


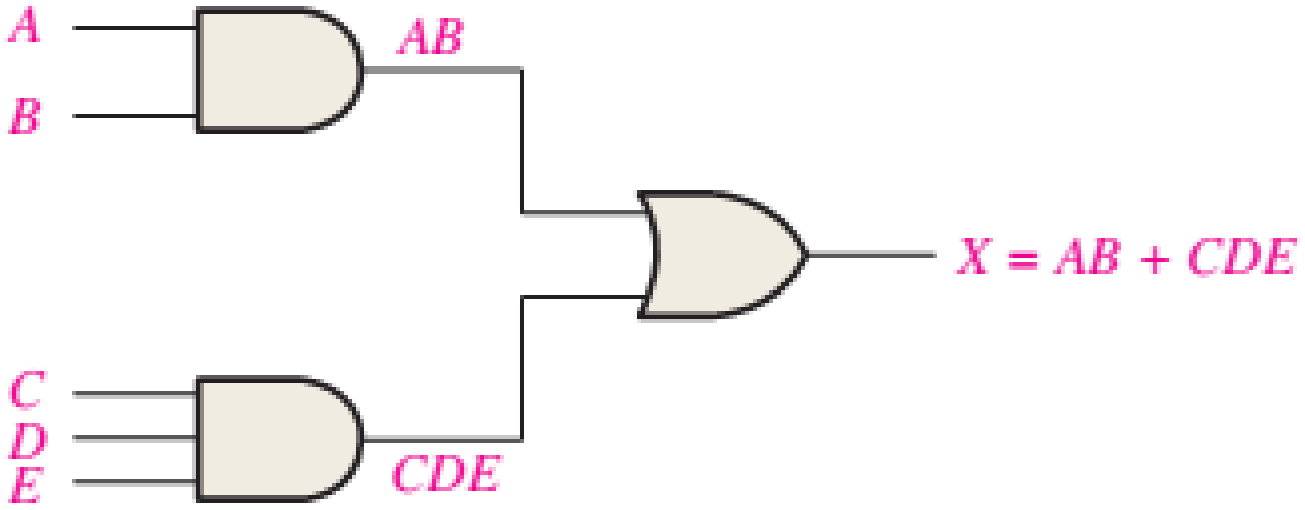
'86

From a Boolean Expression to a Logic Circuit

Let's examine the following Boolean expression:

$$X = AB + CDE$$





Logic circuit for $X = AB + CDE$.

As another example, let's implement the following expression:

$$X = AB(C\overline{D} + EF)$$

AND

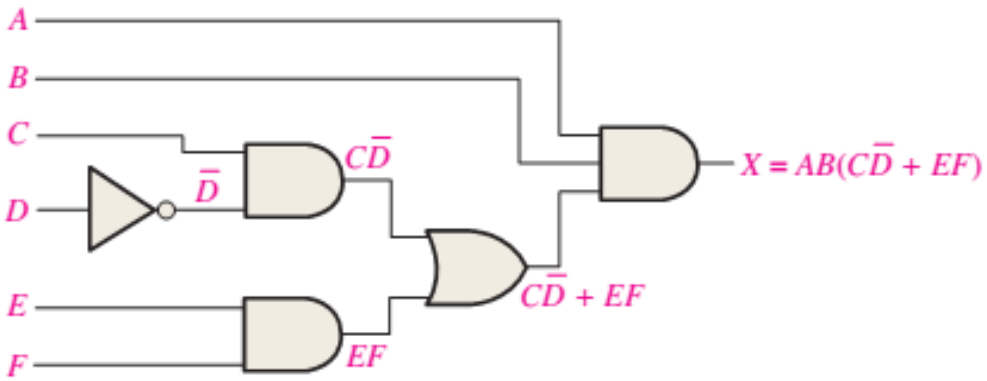
NOT

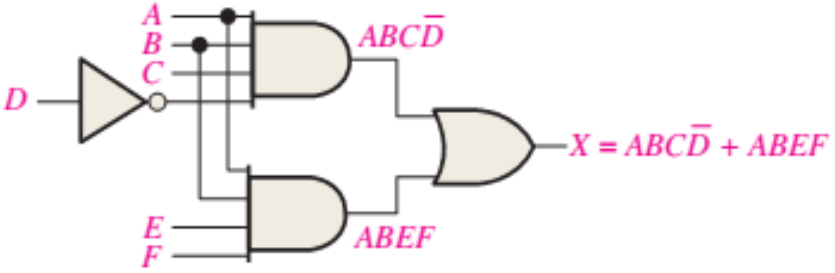
OR

$$X = AB(C\overline{D} + EF)$$

AND

$$AB(C\overline{D} + EF) = ABC\overline{D} + ABEF$$

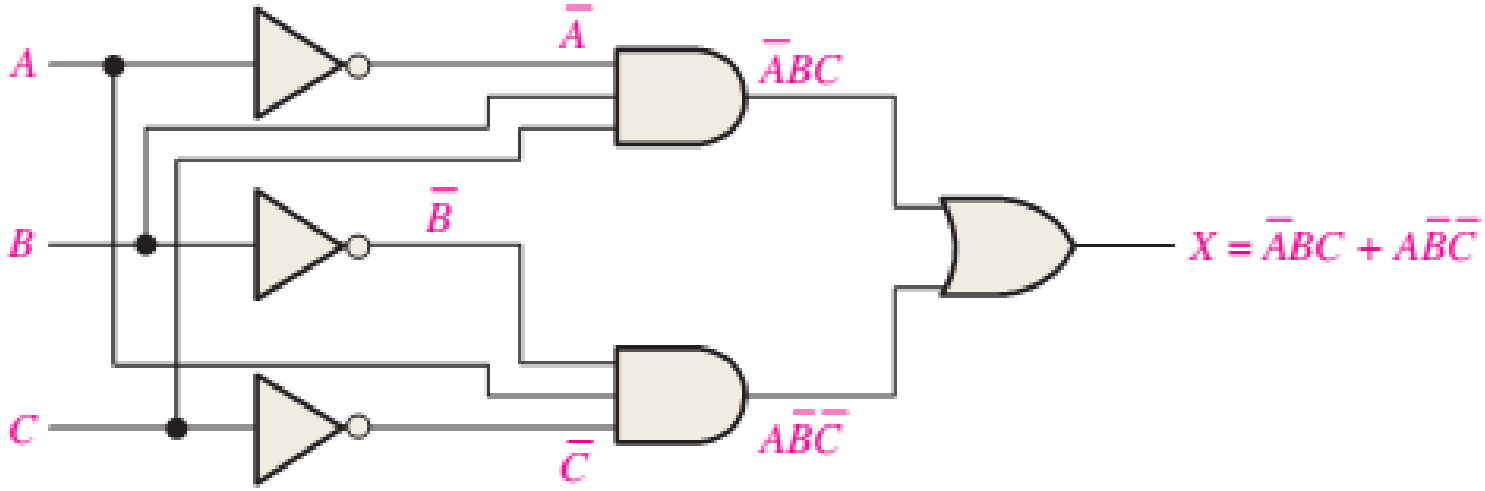




From a Truth Table to a Logic Circuit

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	

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



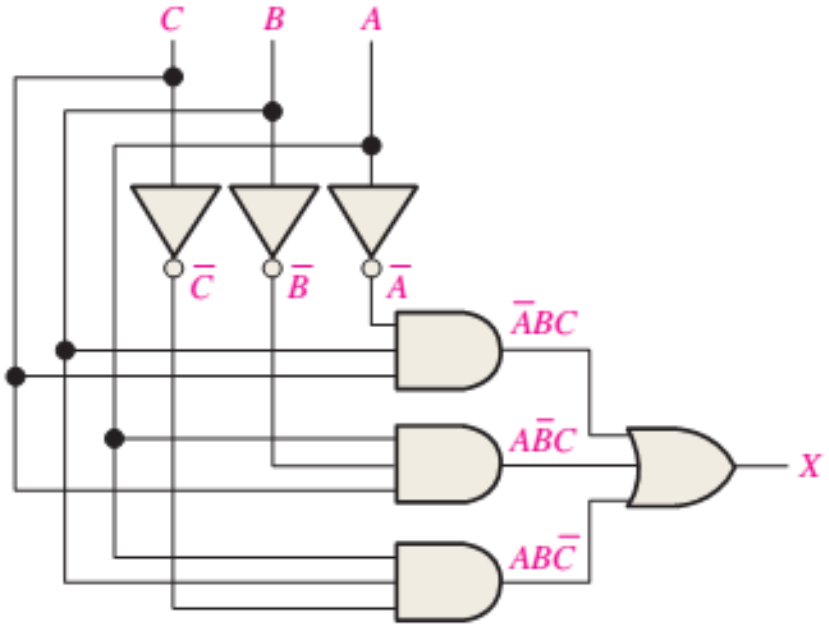
Design a logic circuit to implement the operation specified in the truth table of Table

TABL

Inputs			Output
A	*B*	*C*	*X*
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

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	

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

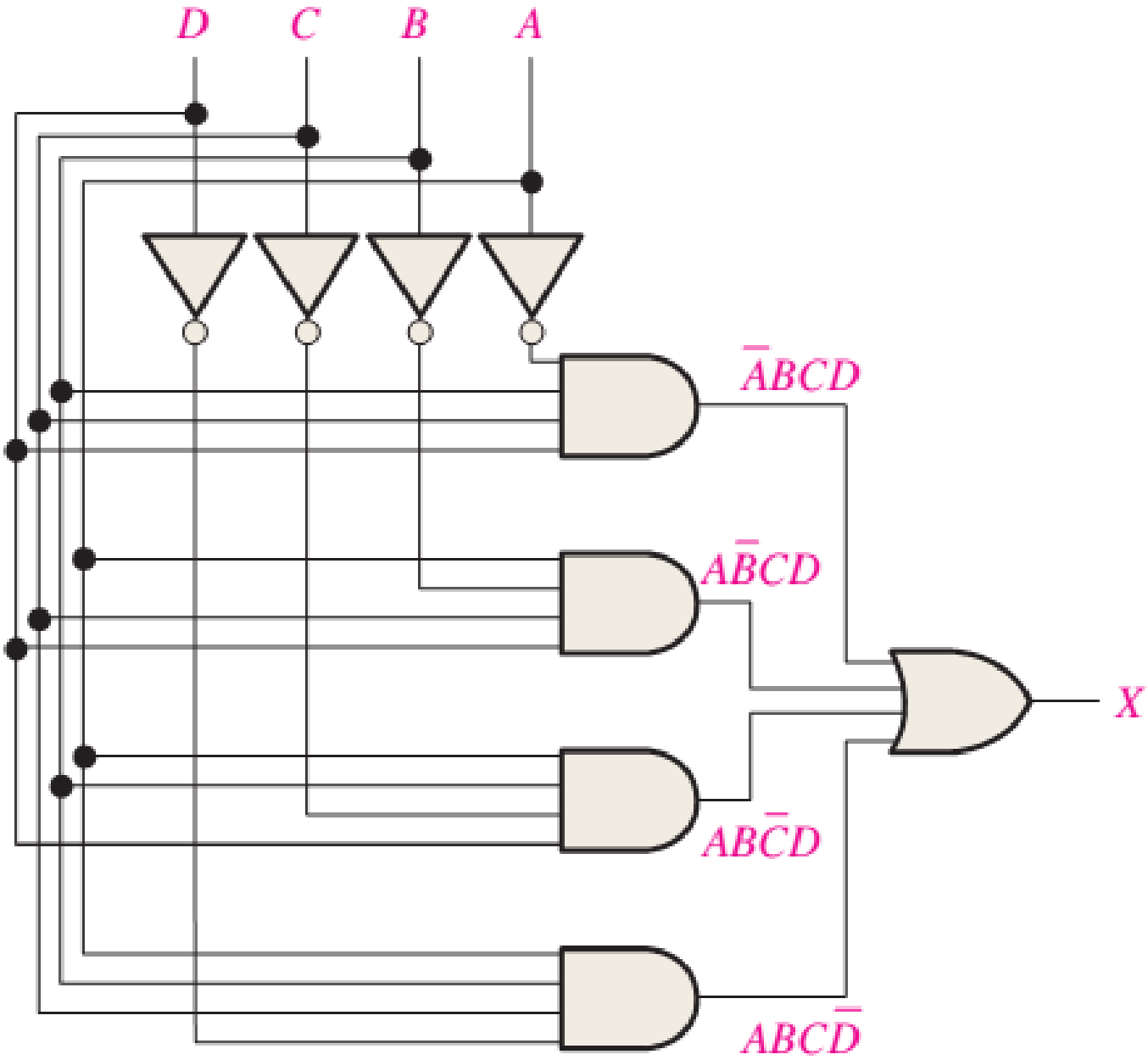


Develop a logic circuit with four input variables that will only produce a 1 output when exactly three input variables are 1s.

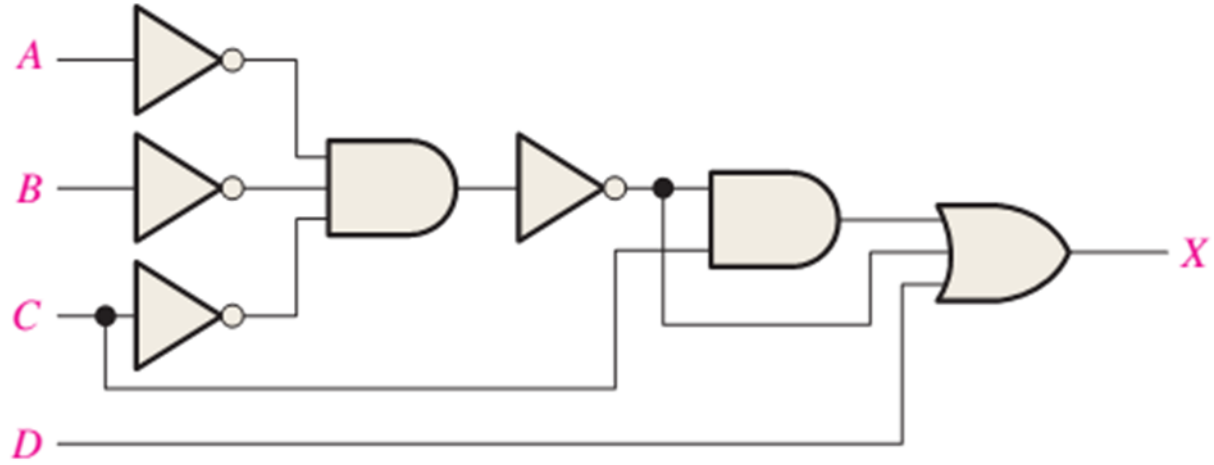
Solution

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	$ABC\overline{D}$

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


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

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



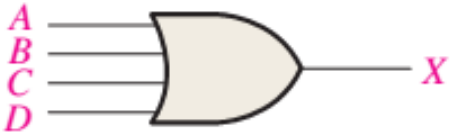
Solution The expression for the output of the circuit is

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

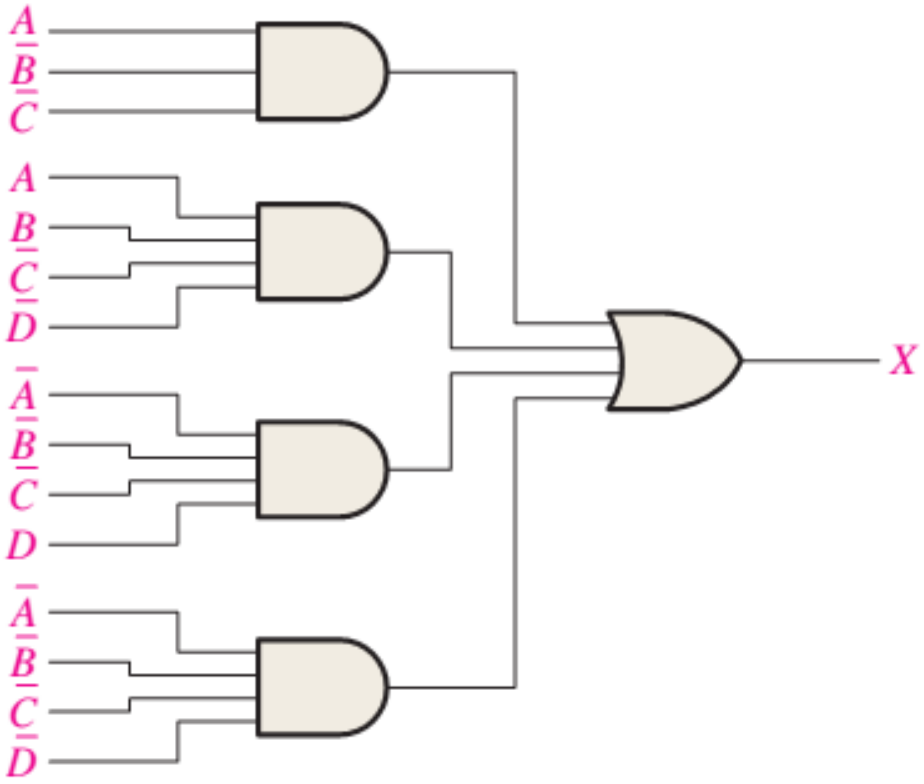
Applying DeMorgan's theorem and Boolean algebra,

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

The simplified circuit is a 4-input OR gate as shown in Figure



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



Solution

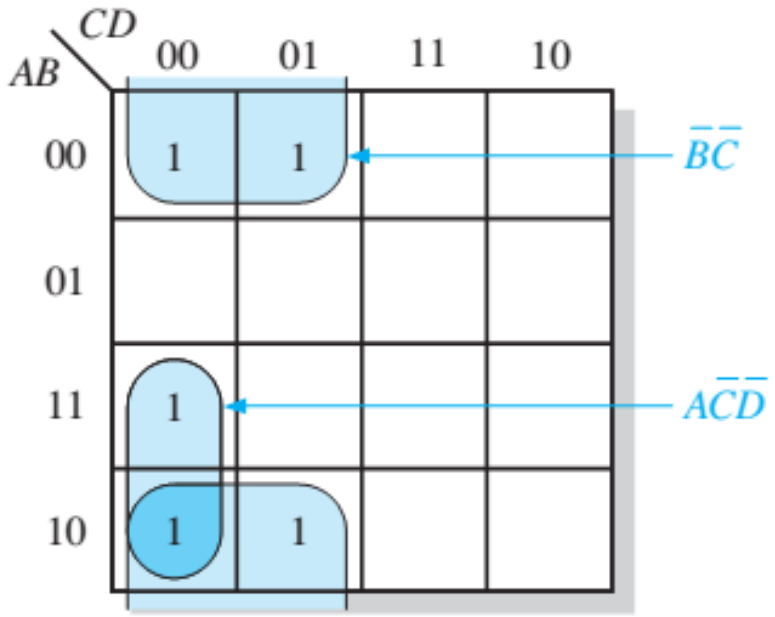
The output expression is

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

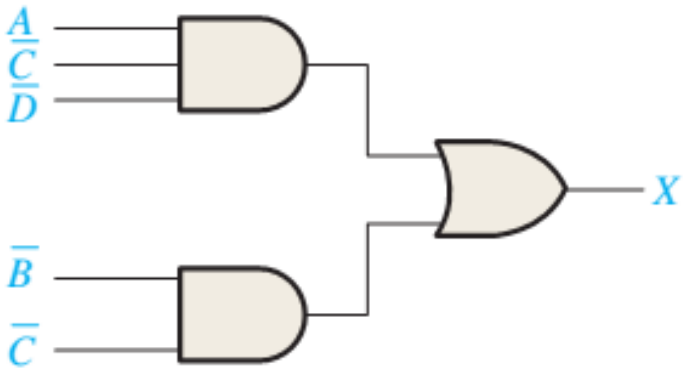
Expanding the first term to include the missing variables D and $\overline{D}$,

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

This expanded SOP expression is mapped and simplified on the Karnaugh map in Figure (a). The simplified implementation is shown in part (b). Inverters are not shown.



(a)

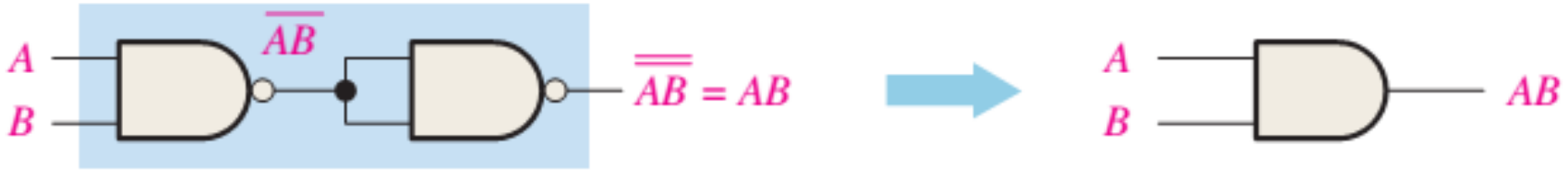


(b)

The Universal Property of NAND and NOR Gates

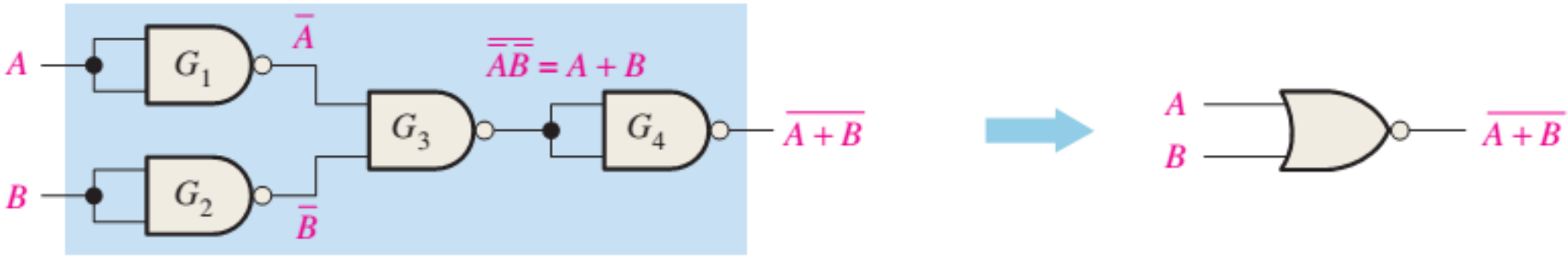
The NAND Gate as a Universal Logic Element

(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



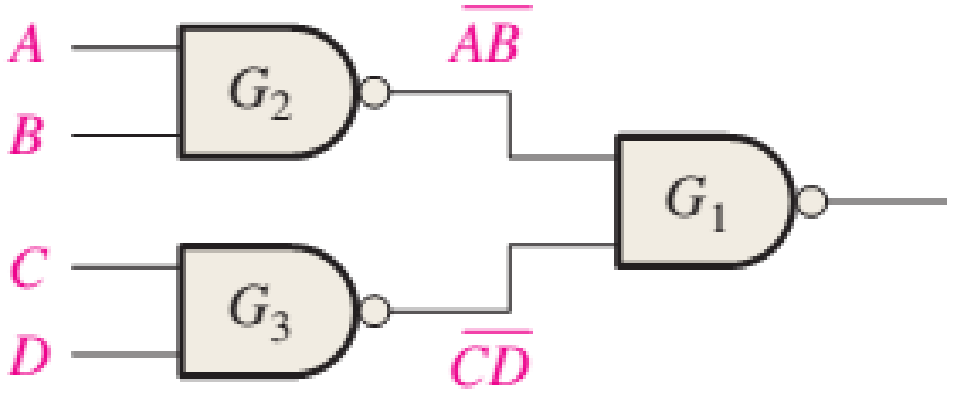
(d) Four NAND gates used as a NOR gate

by DeMorgan's theorem,

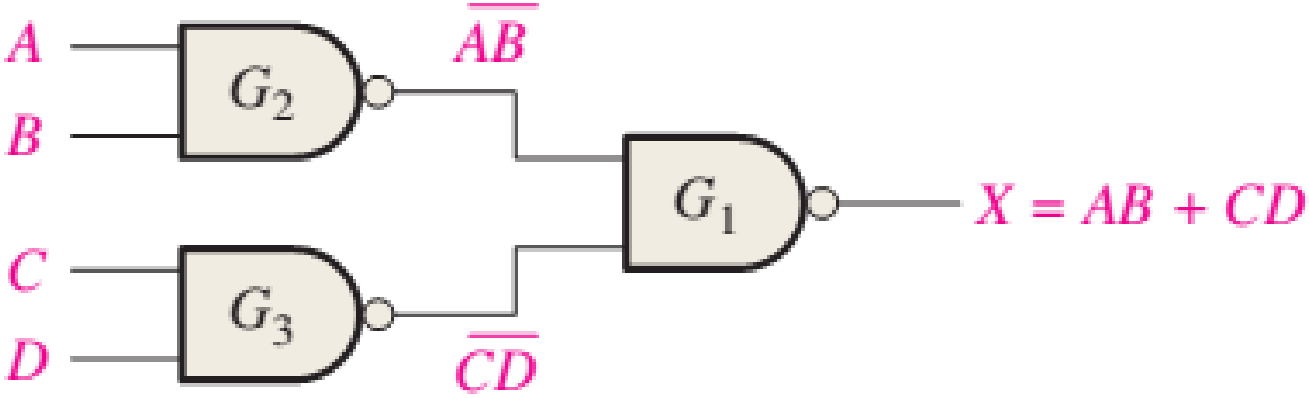
$$\overline{AB} = \overline{A} + \overline{B}$$

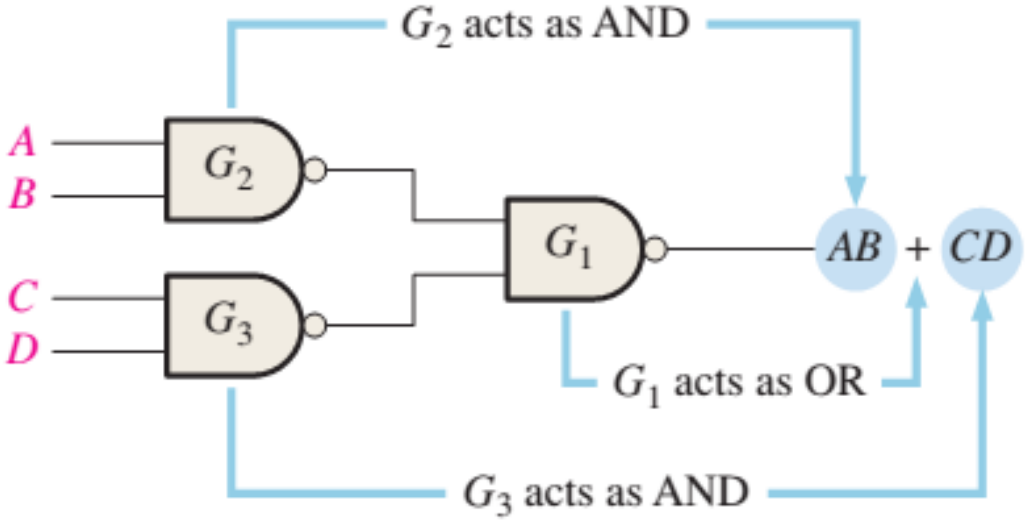
NAND — negative-OR

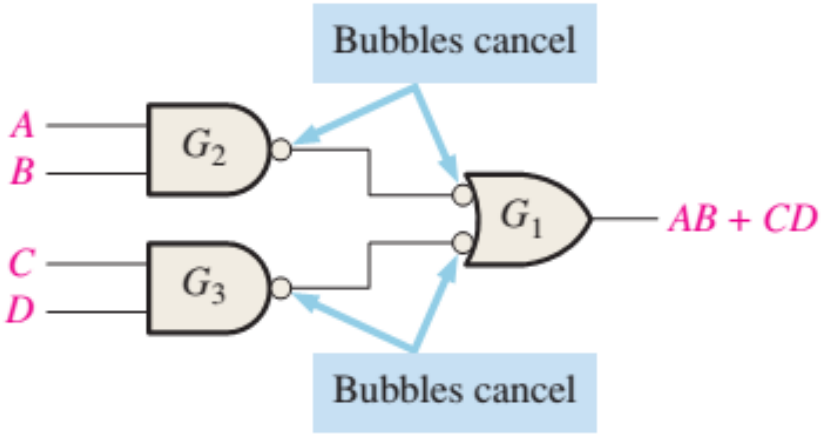
Consider the NAND logic in Figure



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





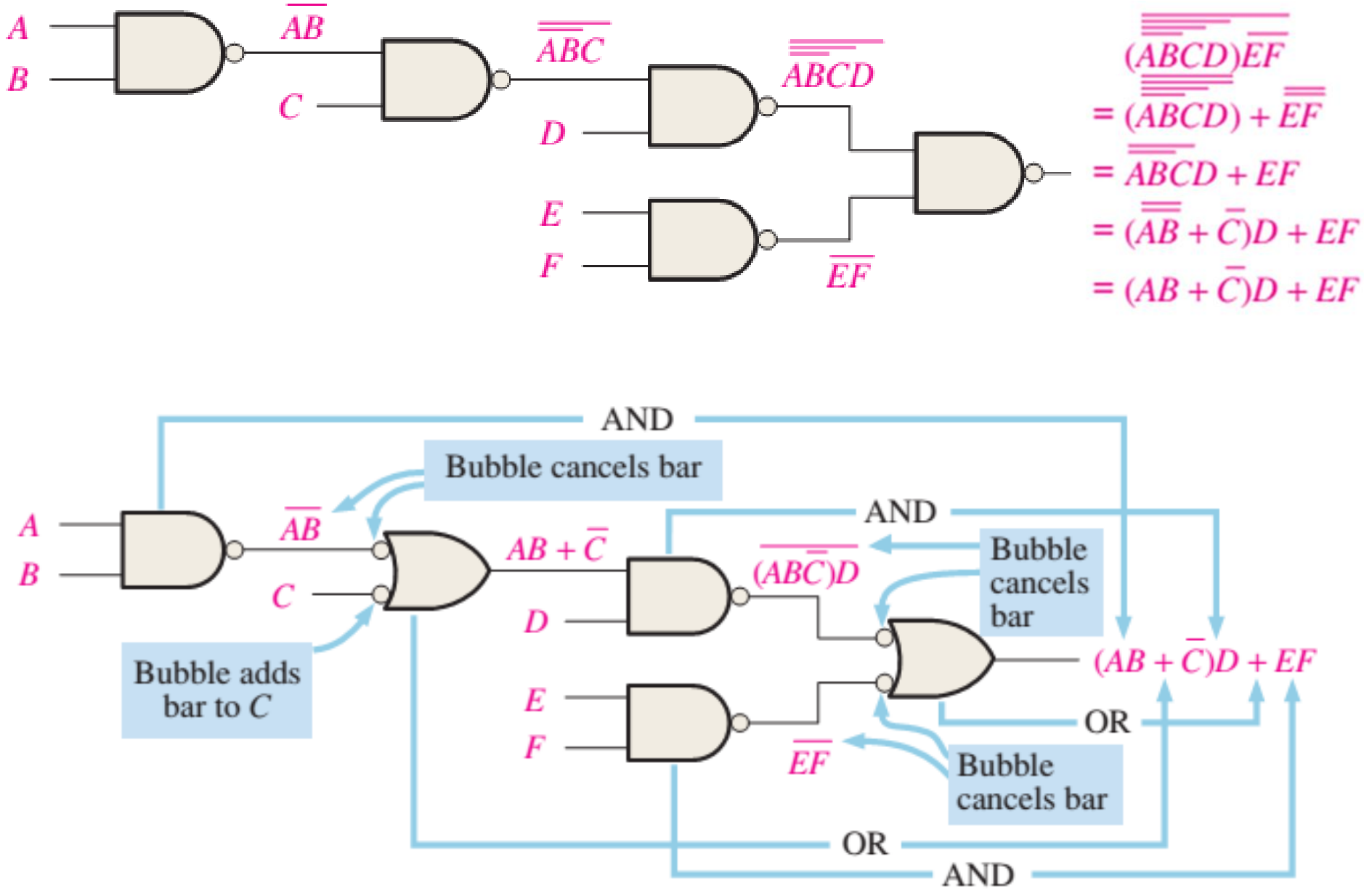


(b) Equivalent NAND/Negative-OR logic diagram

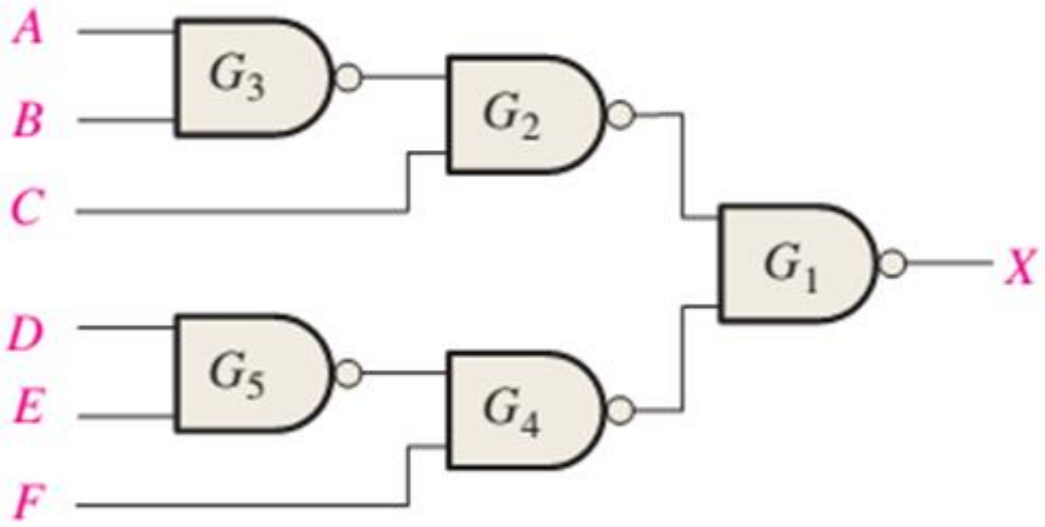
≡

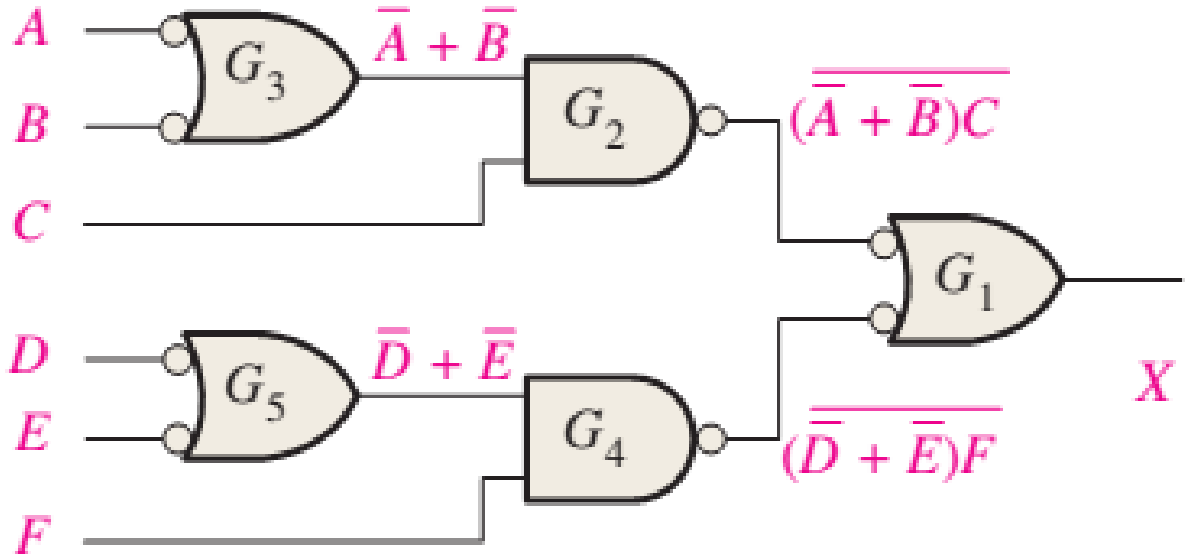
(c) AND-OR equivalent

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



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

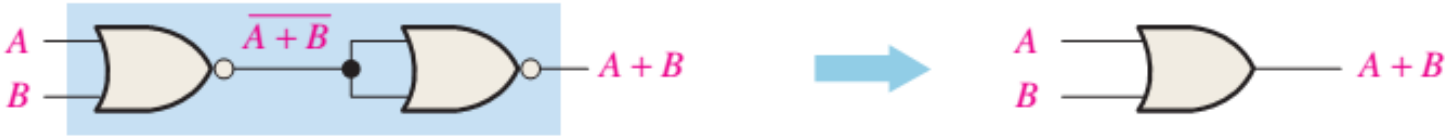




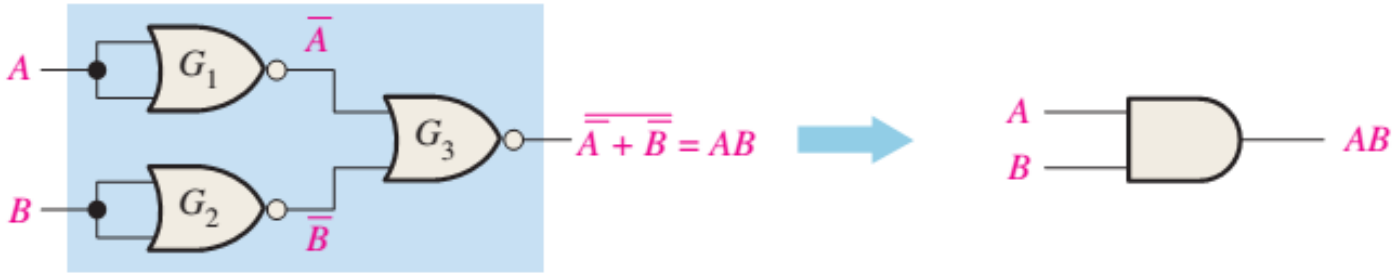
$$X = (\overline{A} + \overline{B})C + (\overline{D} + \overline{E})F$$

The NOR Gate as a Universal Logic Element

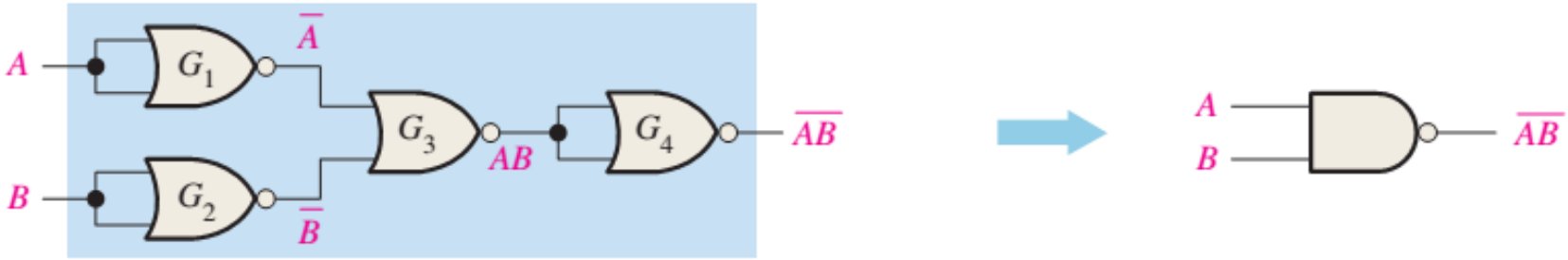
(a) One NOR gate used as an inverter



(b) Two NOR gates used as an OR gate



(c) Three NOR gates used as an AND gate



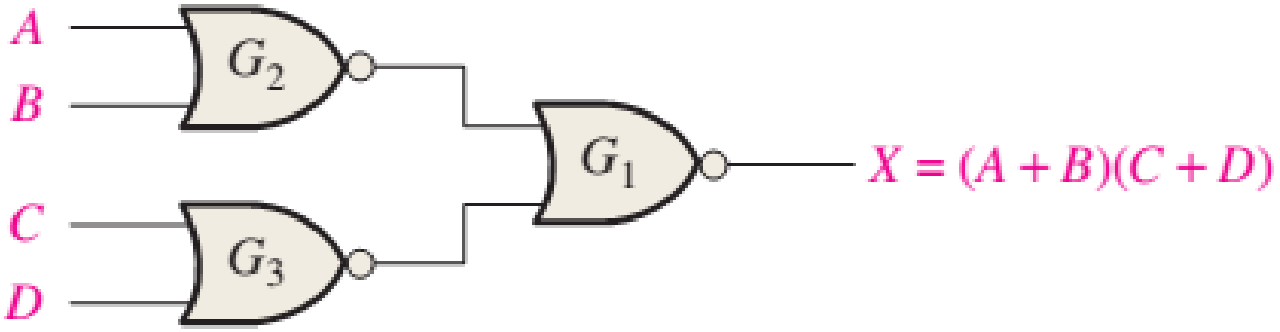
(d) Four NOR gates used as a NAND gate

NOR Logic

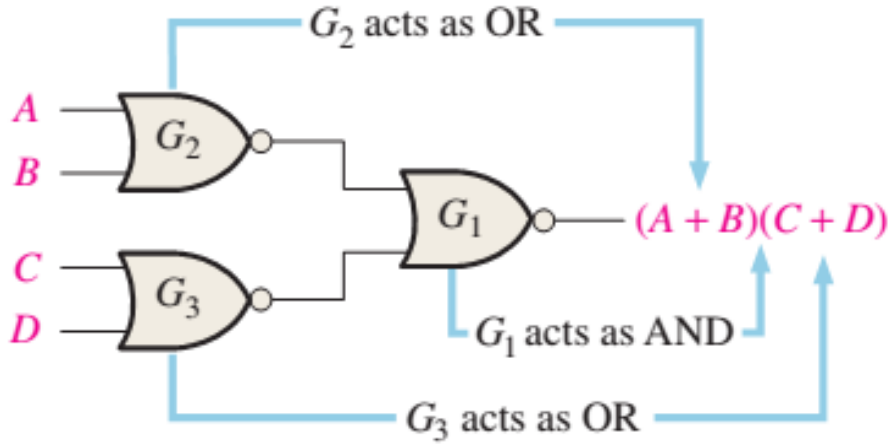
A NOR gate can function as either a NOR or a **negative-AND**, as shown by DeMorgan's theorem.

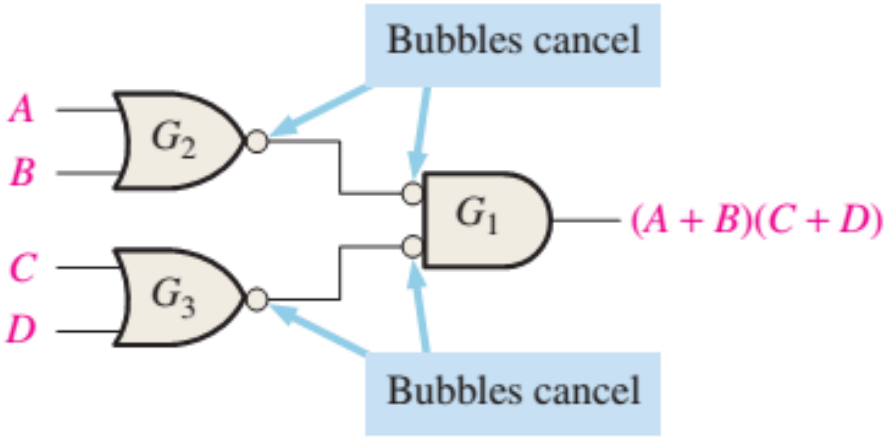
$$\overline{A + B} = \overline{A}\,\overline{B}$$

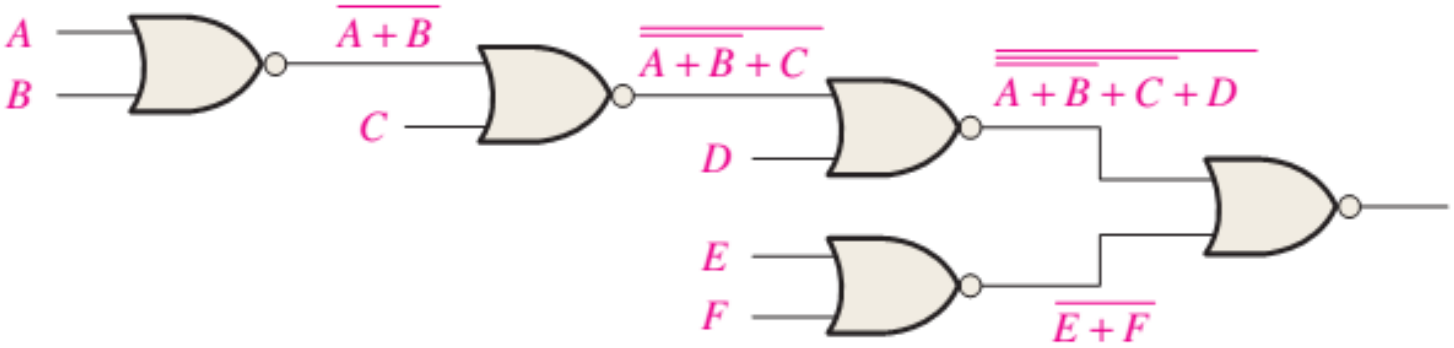
NOR — negative-AND



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







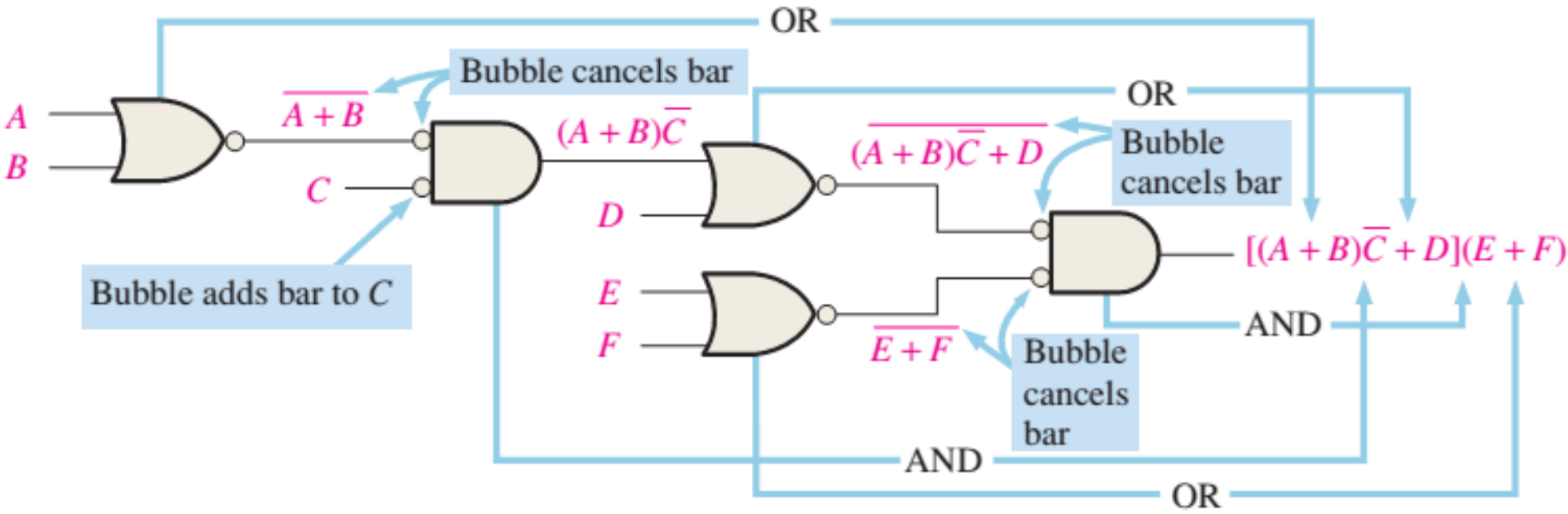
$$\overline{\overline{\overline{A+B}+C}+D+\overline{E+F}}$$

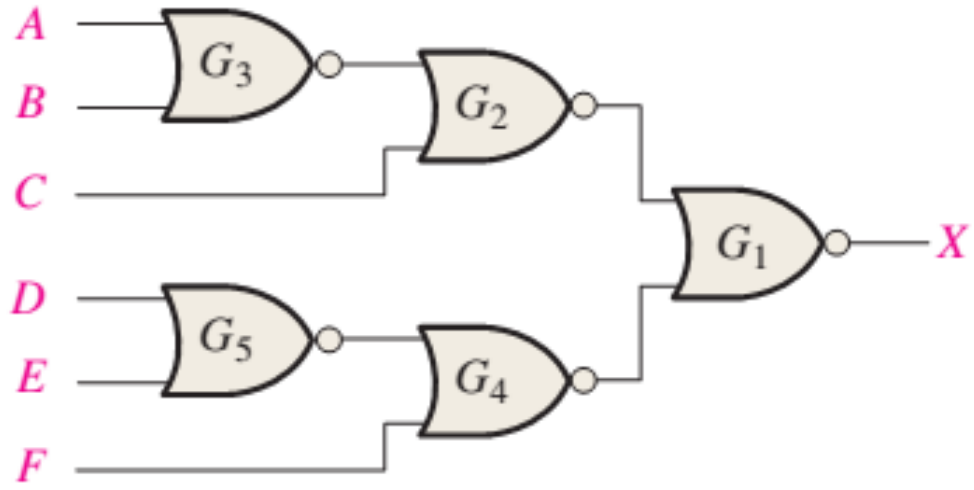
$$= (\overline{\overline{A+B}+C}+D)(\overline{E+F})$$

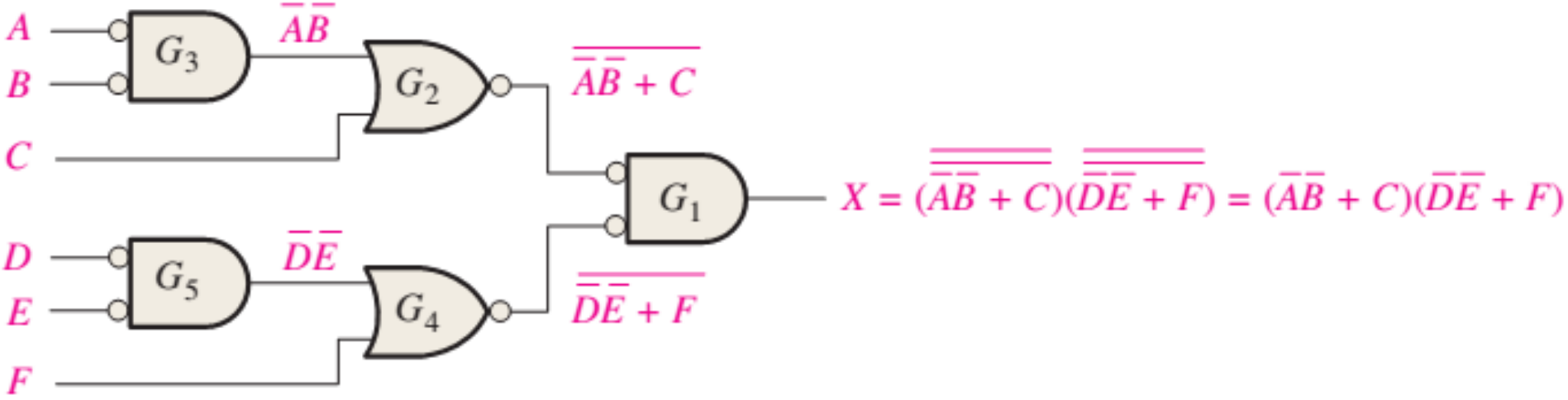
$$= (\overline{\overline{A+B}+C}+D)(E+F)$$

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

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

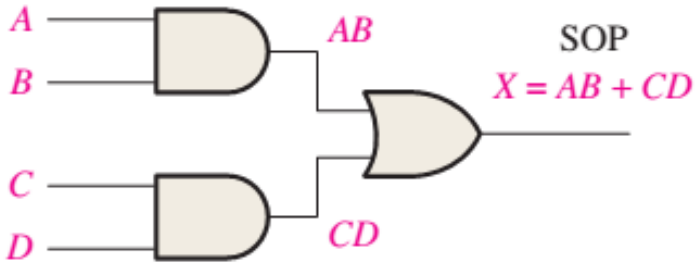



A
B
C
D
E
F
G_3
G_2
G_1
G_5
G_4
X

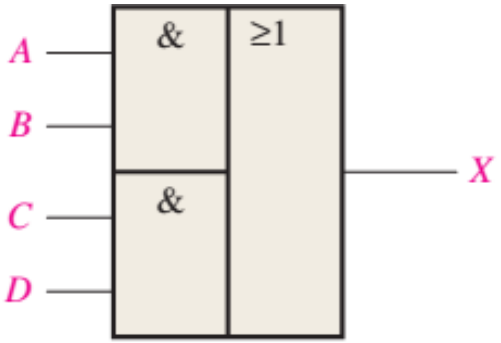

A
B
C
D
E
F
G_3
G_2
G_1
G_5
G_4
$\overline{A}\overline{B}$
$\overline{\overline{A}\overline{B} + C}$
$\overline{D}\overline{E}$
$\overline{\overline{D}\overline{E} + F}$
$X = (\overline{\overline{A}\overline{B} + C})(\overline{\overline{D}\overline{E} + F}) = (\overline{A}\overline{B} + C)(\overline{D}\overline{E}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

AND-OR Logic

AND-OR logic produces an SOP



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



(b) ANSI standard rectangular outline symbol

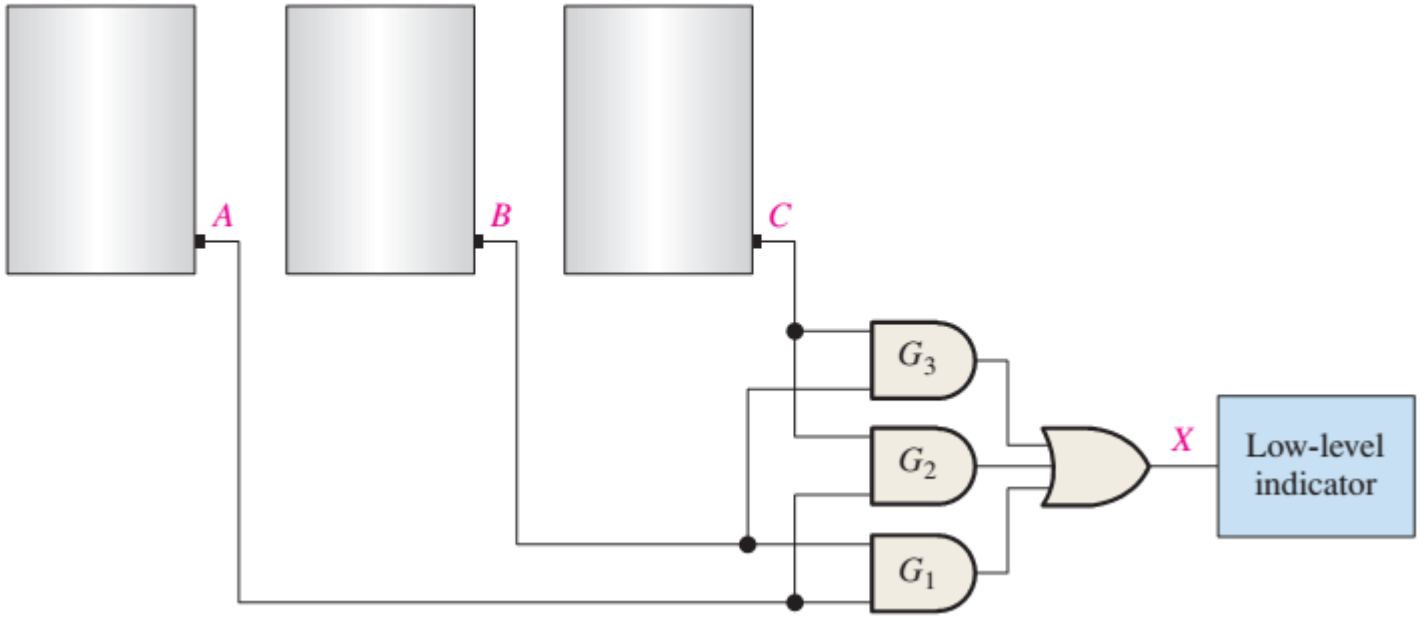
Inputs						Output
A	*B*	*C*	*D*	*AB*	*CD*	*X*
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	0	1	0	0	0	0
0	0	1	1	0	1	1
0	1	0	0	0	0	0
0	1	0	1	0	0	0
0	1	1	0	0	0	0
0	1	1	1	0	1	1
1	0	0	0	0	0	0
1	0	0	1	0	0	0
1	0	1	0	0	0	0
1	0	1	1	0	1	1
1	1	0	0	1	0	1
1	1	0	1	1	0	1
1	1	1	0	1	0	1
1	1	1	1	1	1	1

For a 4-input AND-OR logic circuit, the output *X* is HIGH (1) if both input *A* and input *B* are HIGH (1) or both input *C* and input *D* are HIGH (1).

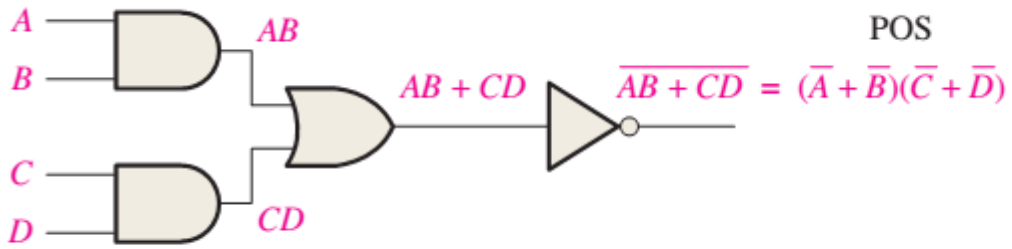
EXAMPLE

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.

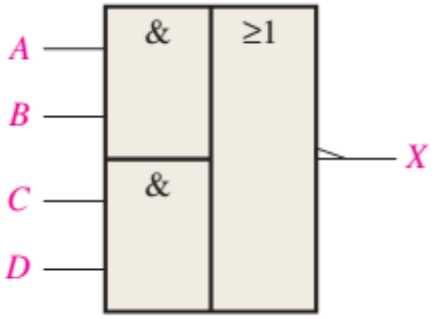
Design a circuit that monitors the chemical level in each tank and indicates when the level in any two of the tanks drops below the specified point.



AND-OR-Invert Logic



(a)



(b)

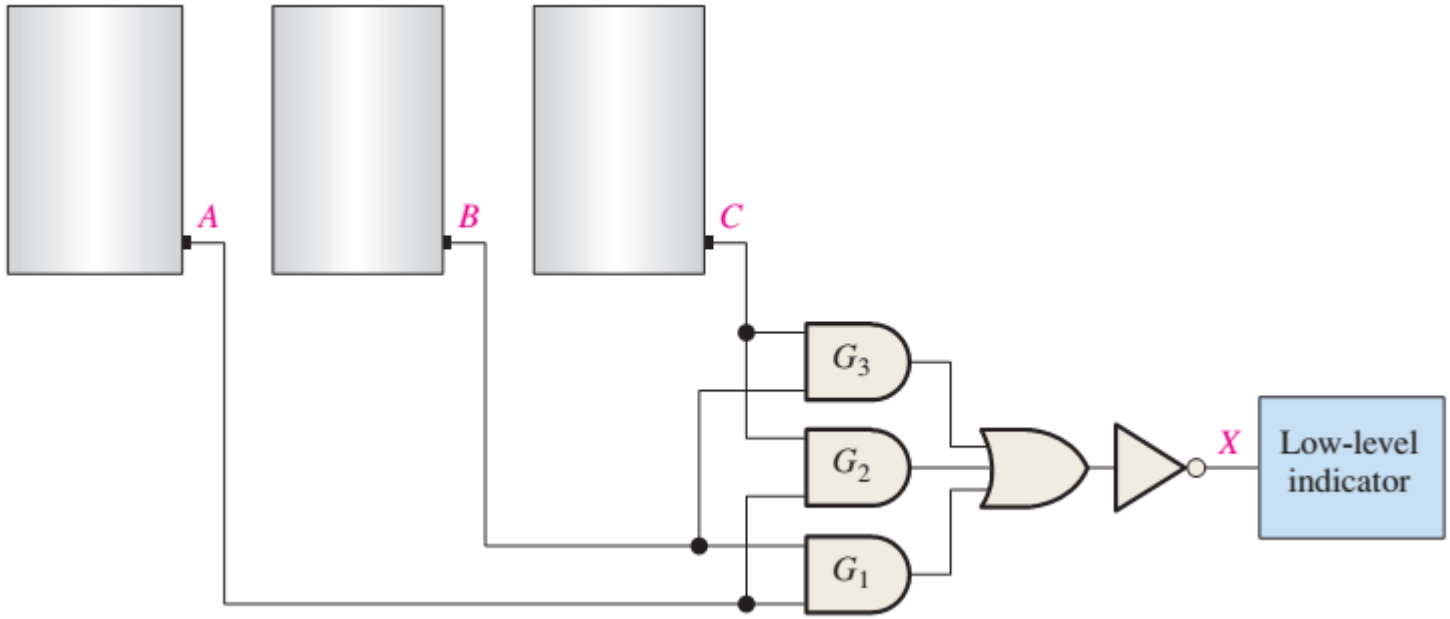
A truth table can be developed from the AND-OR truth table in Table by simply changing all 1s to 0s and all 0s to 1s in the output column.

For a 4-input AND-OR-Invert logic circuit, the output *X* is LOW (0) if both input *A* and input *B* are HIGH (1) or both input *C* and input *D* are HIGH (1).

EXAMPLE

The sensors in the chemical tanks of Example are being replaced by a new model that produces a LOW voltage instead of a HIGH voltage when the level of the chemical in the tank drops below a critical point.

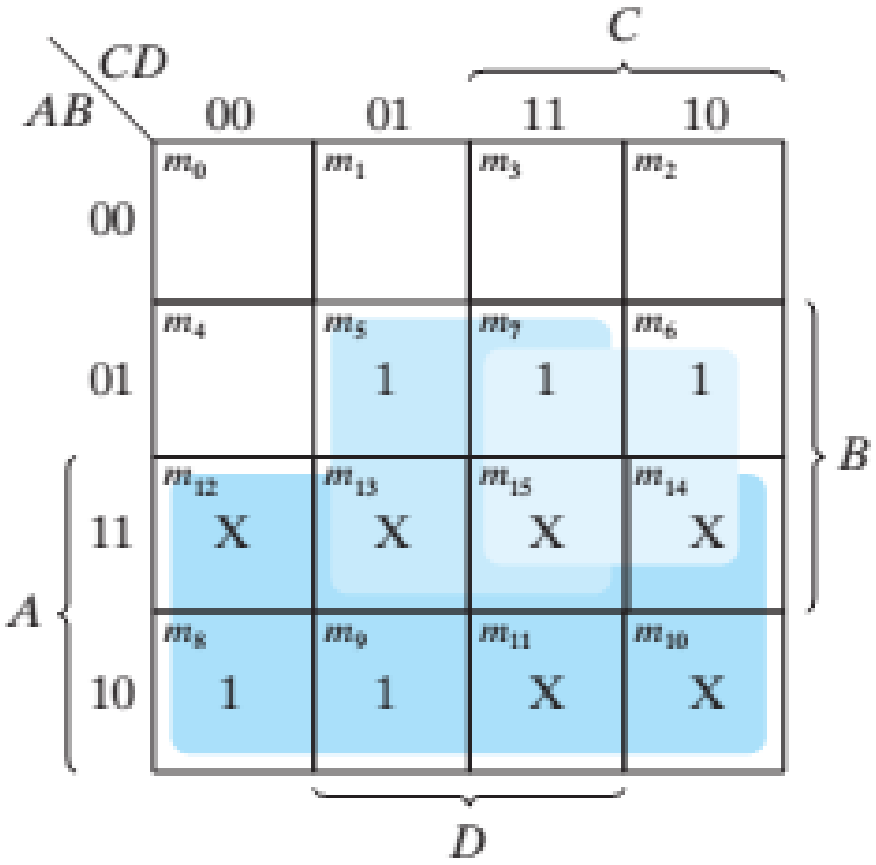
Modify the circuit in Figure to operate with the different input levels and still produce a HIGH output to activate the indicator when the level in any two of the tanks drops below the critical point. Show the logic diagram.



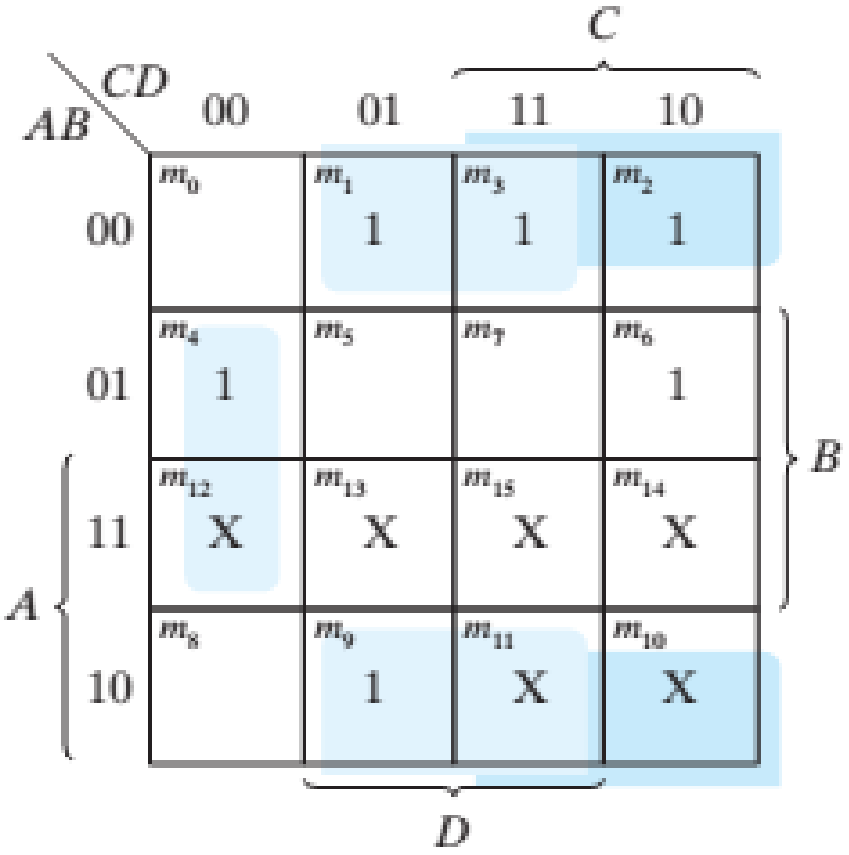
Code Conversion Example

Truth Table for Code Conversion Example

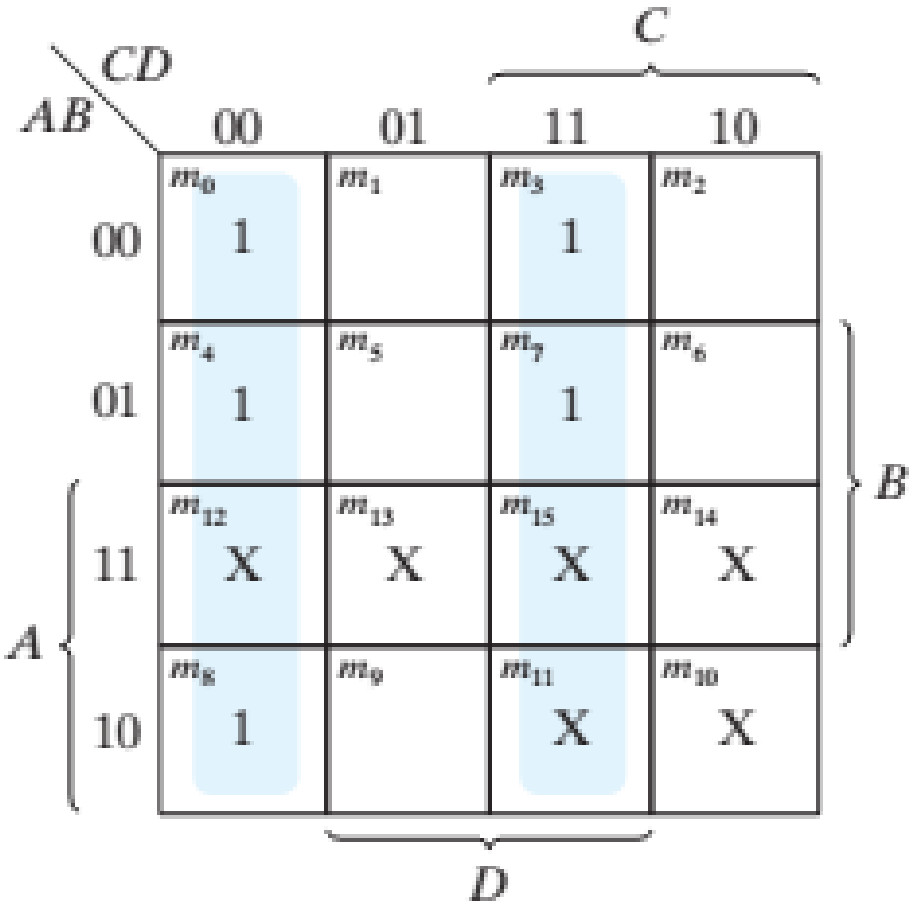
Input BCD				Output Excess-3 Code			
A	***B***	***C***	***D***	***w***	***x***	***y***	***z***
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0



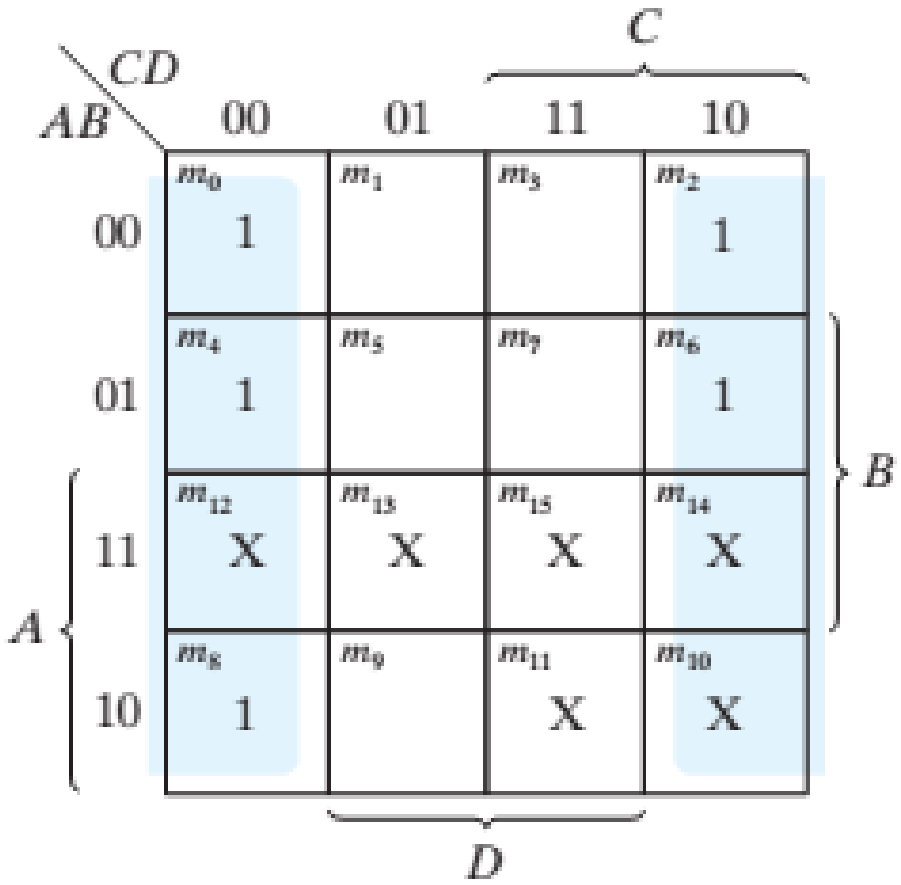
$w = A + BC + BD$



$x = B'C + B'D + BC'D'$


C
CD
AB
00
01
11
10
m_0 1
m_1
m_3 1
m_2
m_4 1
m_5
m_7 1
m_6
m_{12} X
m_{13} X
m_{15} X
m_{14} X
m_8 1
m_9
m_{11} X
m_{10} X
B
A
D

$y = CD + C'D'$


C
CD
AB
00
01
11
10
m_0 1
m_1
m_3
m_2 1
m_4 1
m_5
m_7
m_6 1
m_{12} X
m_{13} X
m_{15} X
m_{14} X
m_8 1
m_9
m_{11} X
m_{10} X
B
A
D

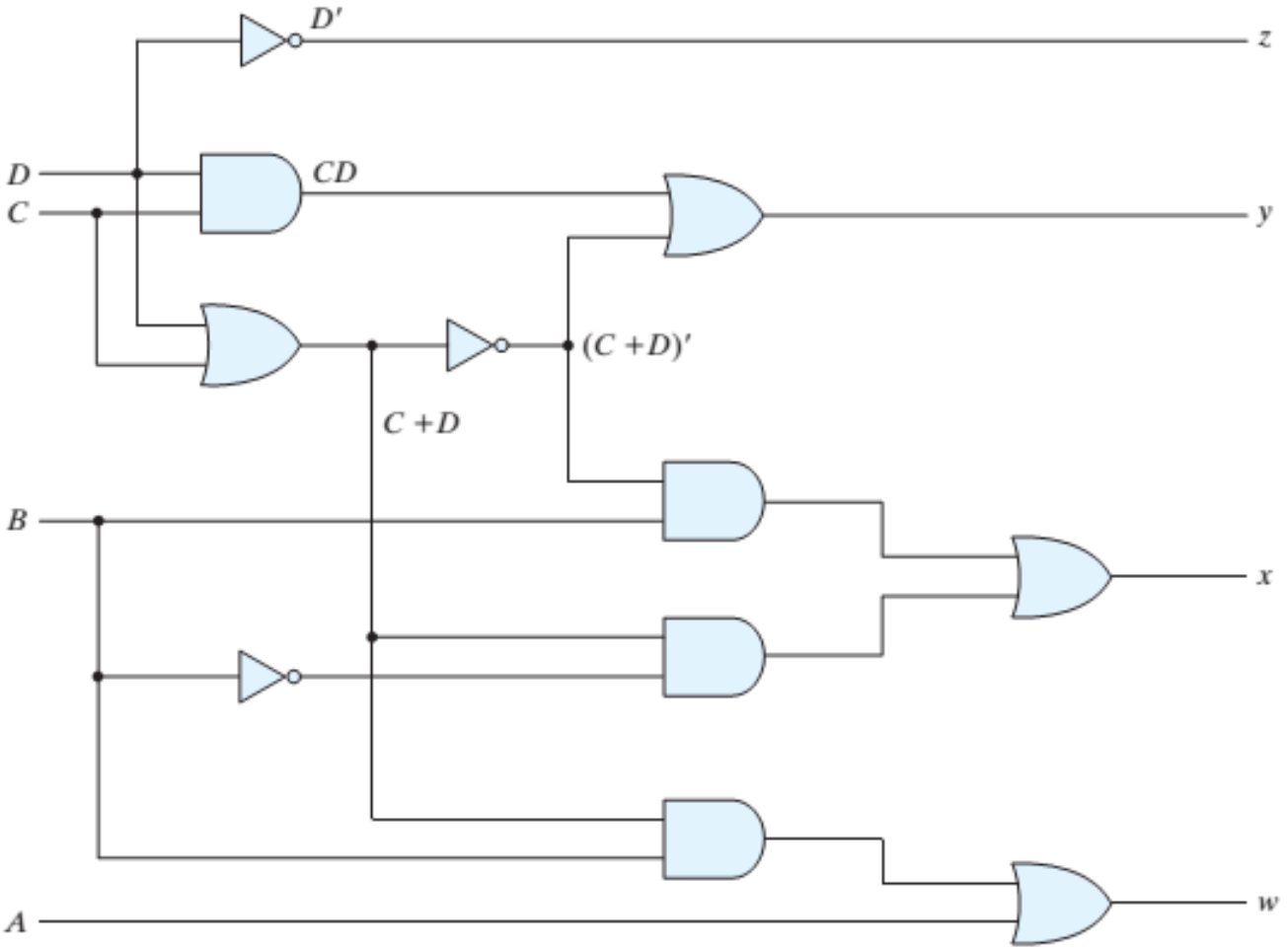
$z = D'$

$$z = D'$$

$$y = CD + C'D' = CD + (C + D)'$$

$$x = B'C + B'D + BC'D' = B'(C + D) + BC'D'$$
$$= B'(C + D) + B(C + D)'$$

$$w = A + BC + BD = A + B(C + D)$$




Thank
you!