الجامعة السورية الخاصة
SYRIAN PRIVATE UNIVERSITY

كـلـيـــة هندسة الحاسوب والمعلوماتية والاتصالات
Faculty of Computer & Informatics
and Communications Engineering

Logic Circuits

Dr. Eng.

Hassan M. Ahmad

Hassan.Ahmad@spu.edu.sy, istamo48@mail.ru

Chapter_ 5

Combinational Logic Analysis

(تحليل المنطق التوافقي)

Lecture _08

Combinational Logic Circuits

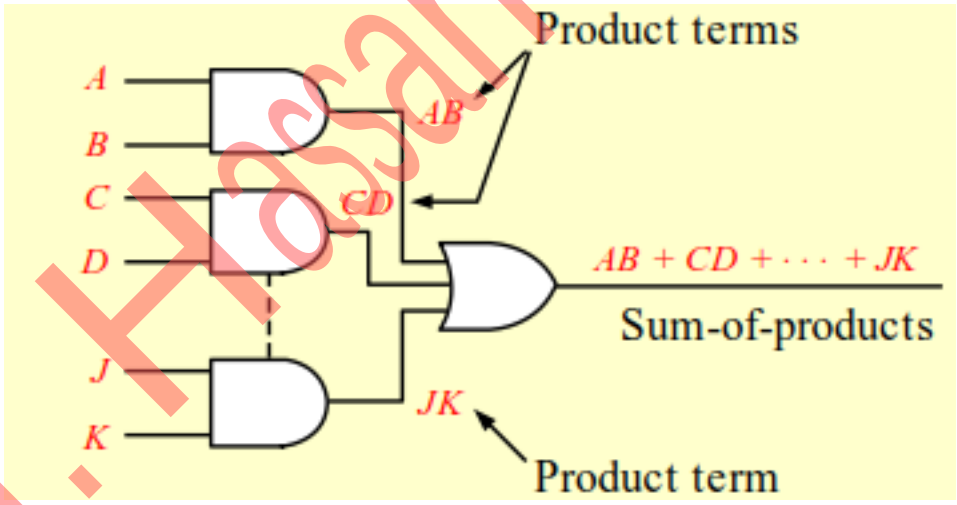
&

its Implementing

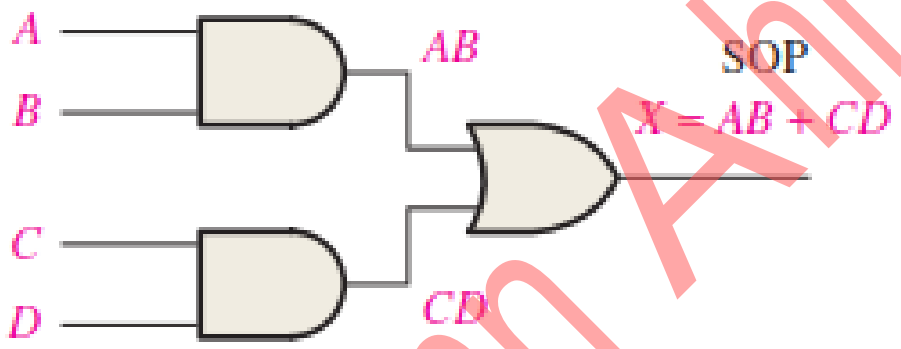
8-1. Basic Combinational Logic Circuits

AND-OR Logic

- In Sum-of-Products (SOP) form, basic combinational circuits can be directly implemented with AND-OR combinations if the necessary complement terms are available.



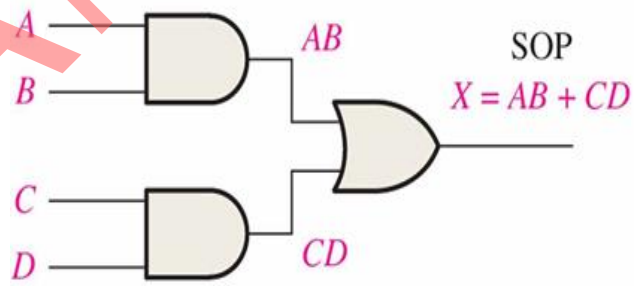
- AND-OR circuit consists of two 2-input AND gates and one 2-input OR gate, as shown in Figure-Logic diagram.



- In general, an AND-OR circuit can have any number of AND gates each with any number of inputs.

- The truth table for a 4-input AND-OR logic circuit is shown.

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

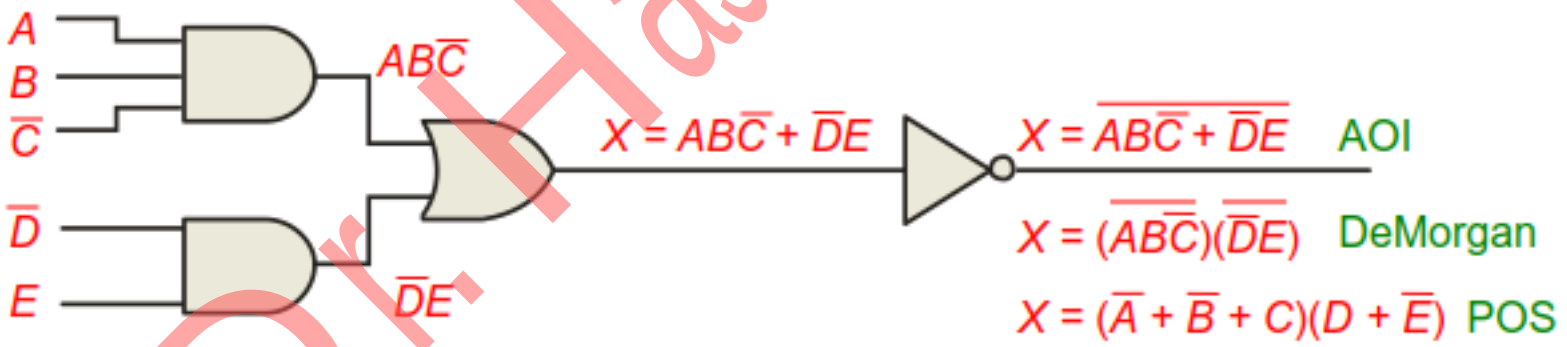


- The operation of the AND-OR circuit in Fig. is stated as follows:

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

AND-OR-Invert Logic

- When the output of an AND-OR is complemented (inverted), it results in an **AND-OR Invert** circuit.
- The logic diagram in Figure shows an AND-OR Invert circuit and development of the POS output expression.
- An example of an **AOI** implementation is shown. The output expression can be changed to a POS expression by applying DeMorgan's theorem twice.



AND-OR-Invert Logic

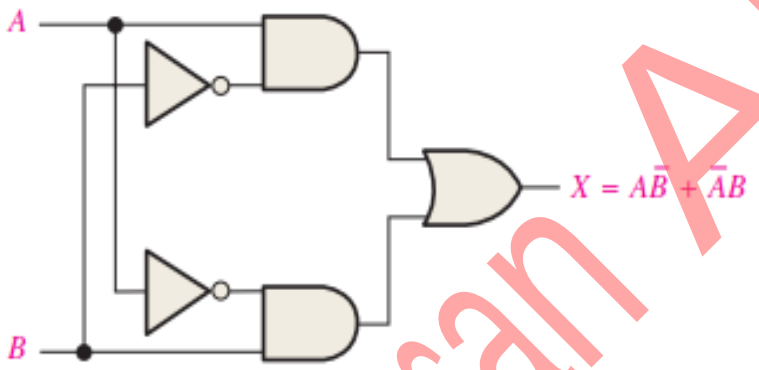
- In general, an **A**ND-**O**R **I**nvert circuit can have any number of AND gates each with any number of inputs.
- The operation of the AND-OR Invert circuit in Fig. is stated as follows:

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

- A truth table can be developed from truth table for a 4-input AND-OR logic circuit by simply changing all 1s to 0s and all 0s to 1s in the output column.

Exclusive-OR Logic

- The exclusive-OR gate is actually a combination of two AND gates, one OR gate, and two invertors, as shown in Figure.



  - The truth table for an exclusive-OR gate is

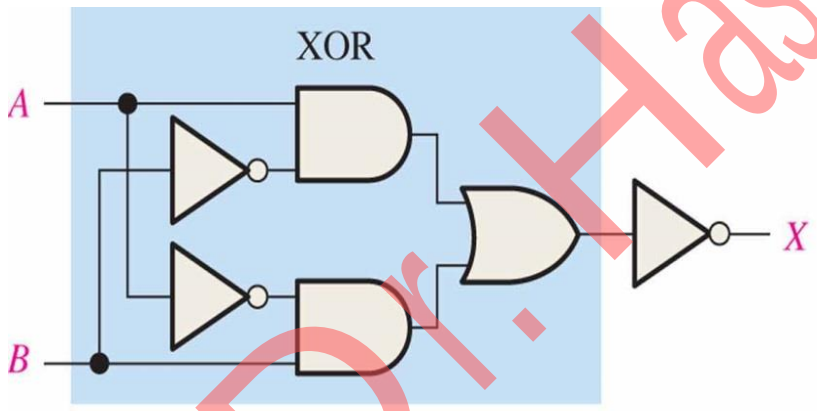
A	B	X
0	0	0
0	1	1
1	0	1
1	1	0

  - Notice that the output is HIGH whenever *A* and *B* disagree (opposite levels).
  - A special exclusive OR operator $\oplus$ is often used, so the output expression can be stated as $X = A\overline{B} + \overline{A}B$
    - ***X*** is equal to *A* exclusive-OR *B*" and can be written as $X = A \oplus B$

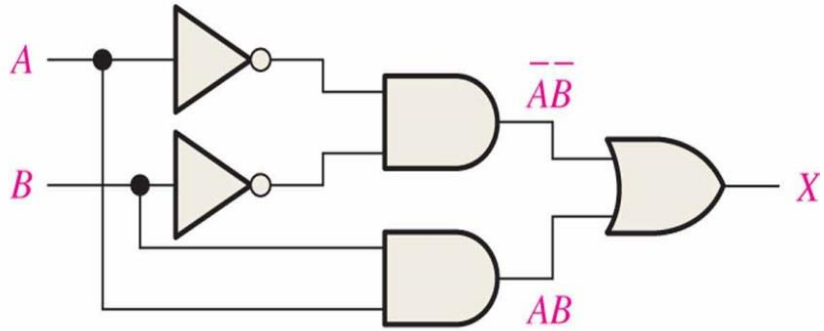
Exclusive-NOR Logic

- The complement of exclusive-OR function is exclusive-NOR, which is derived as follows: $X = \overline{A\overline{B} + \overline{A}B} = \overline{(A\overline{B})}\ \overline{(\overline{A}B)} = (\overline{A} + B)(A + \overline{B}) = \overline{A}\,\overline{B} + AB$

  - The exclusive-NOR can be implemented by simply inverting the output of an exclusive-OR, as shown in Fig. (a), or by directly implementing the expression $\overline{A}\,\overline{B} + AB$, as show in Fig. (b).



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



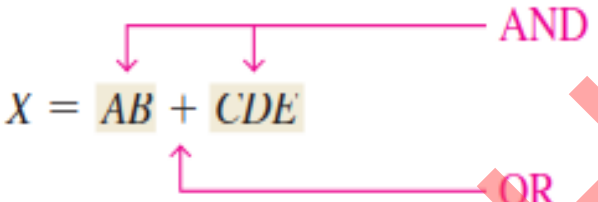
(b) $X = \overline{A}\,\overline{B} + AB$

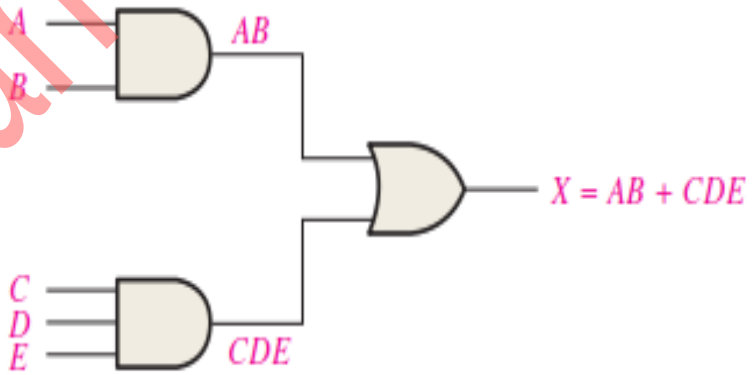
8-2. Implementing Combinational Logic

From a Boolean Expression to a Logic Circuit

Let's examine the following Boolean expression: $X = AB + CDE$

- This expression is composed of two terms, *AB* and *CDE*, with a domain of five variables.
- The first term is formed by ANDing *A* with *B*, and the second term is formed by ANDing *C*, *D* and *E*.





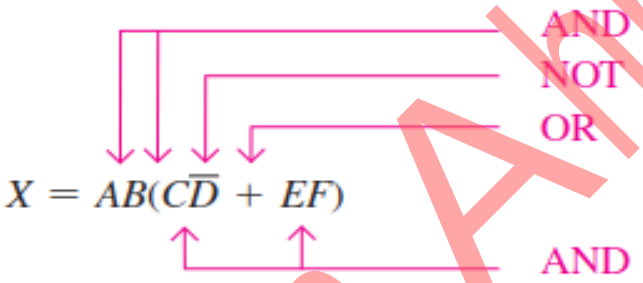
Example 8-1

Given the Boolean expression $X = AB(C\overline{D} + EF)$

Implement the corresponding logic circuit.

Solution

The structure that indicated in relation to the expression is as follows:

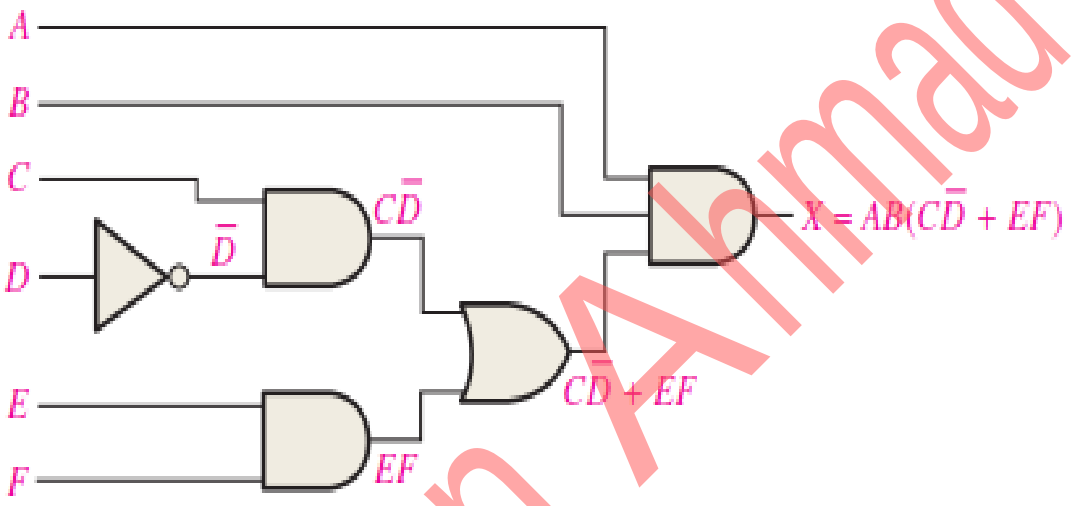


Before we can implement the final expression, we must create the sum term $C\overline{D} + EF$;
but before we can get this term, we must create the product terms $C\overline{D}$ and EF ;
but before we can get the term $C\overline{D}$, we must create $\overline{D}$.
So, as we can see, the logic operations must be done in the proper order.

The logic gates required to implement $X = AB(C\overline{D} + EF)$ are as follows:

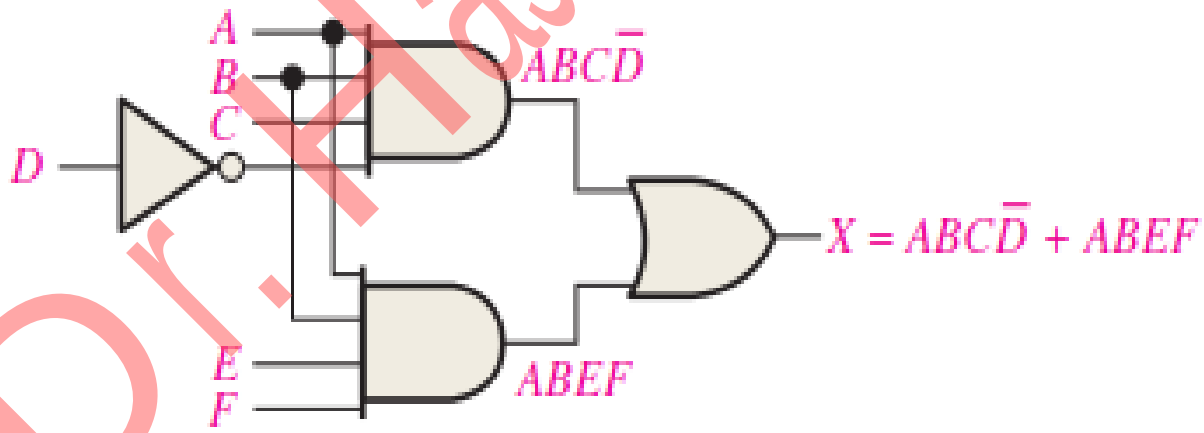
1. **One** invertor to form $\overline{D}$.
2. **Two** 2-input AND gates to form $C\overline{D}$ and EF .
3. **One** 2-input OR gate to form $C\overline{D} + EF$.
4. **One** 3-input AND gate to form X.

The resulting logic circuit for given expression is shown in Fig.



The expression is converted to SOP as follows: $X = AB(C\overline{D} + EF) = ABC\overline{D} + ABEF$

and the resulting circuit is shown in Fig.



From a Truth Table to a Logic Circuit

- **Let** the following table specifies a logic function.

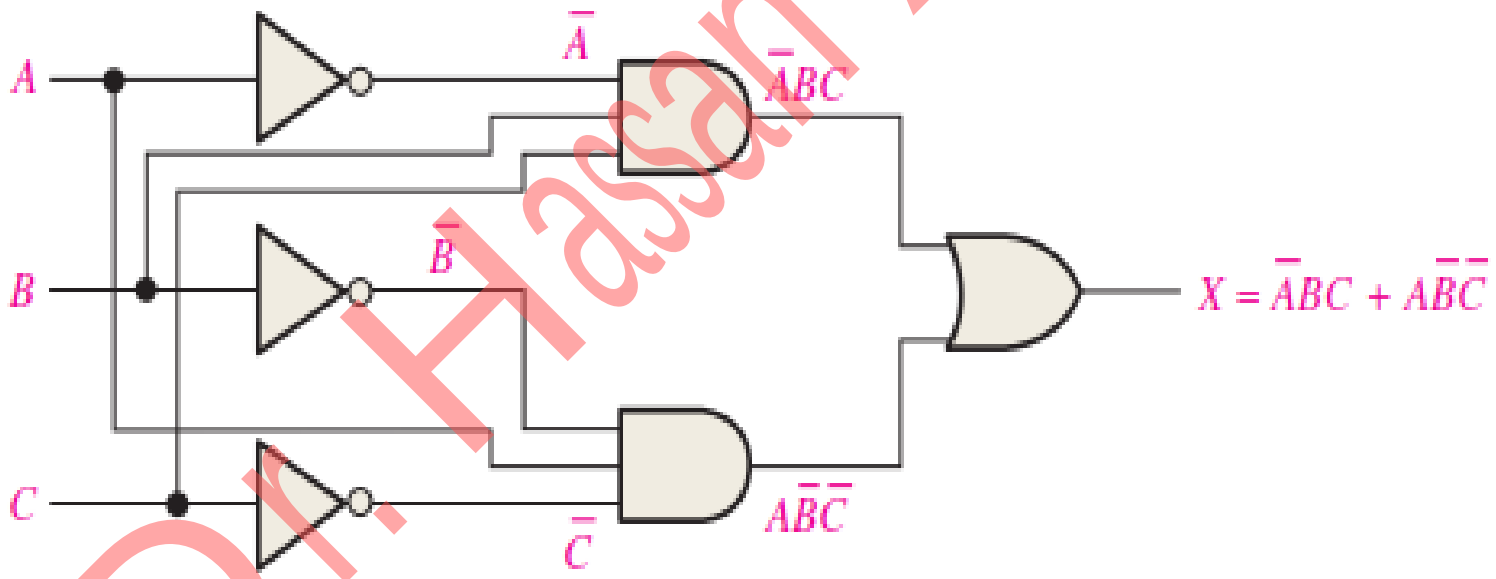
Inputs			Output	
A	*B*	*C*	*X*	Product Term
0	0	0	0	
0	0	1	0	
0	1	0	0	
0	1	1	1	$\overline{A}BC$
1	0	0	1	$A\overline{B}\,\overline{C}$
1	0	1	0	
1	1	0	0	
1	1	1	0	

- The Boolean SOP expression obtained from the truth table **OR**ing the product terms for which $X = 1$ is: $X = \overline{A}BC + A\overline{B}\,\overline{C}$
- The first term in the expression is formed by **AND**ing the three variables $\overline{A}$, B, and C
- The second term is formed by **AND**ing the three variables A, $\overline{B}$, and $\overline{C}$

- The logic gates required to implement this expression are as follows:
  - **Three** invertors to form $\overline{A}$, $\overline{B}$, and $\overline{C}$ variables;
  - **Two** 3-inputs AND gates to form terms $\overline{A}BC$ and $A\overline{B}\overline{C}$;
  - **One** 2-input OR gate to form the final output function, $\overline{A}BC + A\overline{B}\overline{C}$.

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

- The implementation of logic function $X = \overline{A}BC + A\overline{B}\overline{C}$ is illustrated in Fig.



Example 8-2

Design a logic circuit to implement the operation specified in following given truth table.

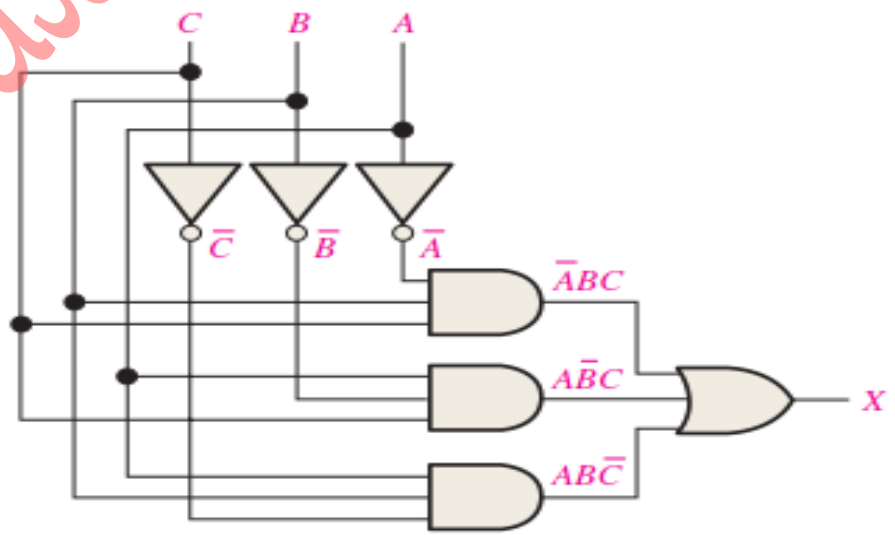
Inputs			Output	
A	B	C	X	Product Term
0	0	0	0	
0	0	1	0	
0	1	0	0	
0	1	1	1	$\overline{A}BC$
1	0	0	0	
1	0	1	1	$A\overline{B}C$
1	1	0	1	$AB\overline{C}$
1	1	1	0	

Solution

Notice that $X = 1$ for only three of the input conditions. Therefore, the logic expression is

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

The logic gates required are **three** inverters, **three** 3-input AND gates and **one** 3-input OR gate. The logic circuit is shown in Fig.



Example 8-3

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

Solution

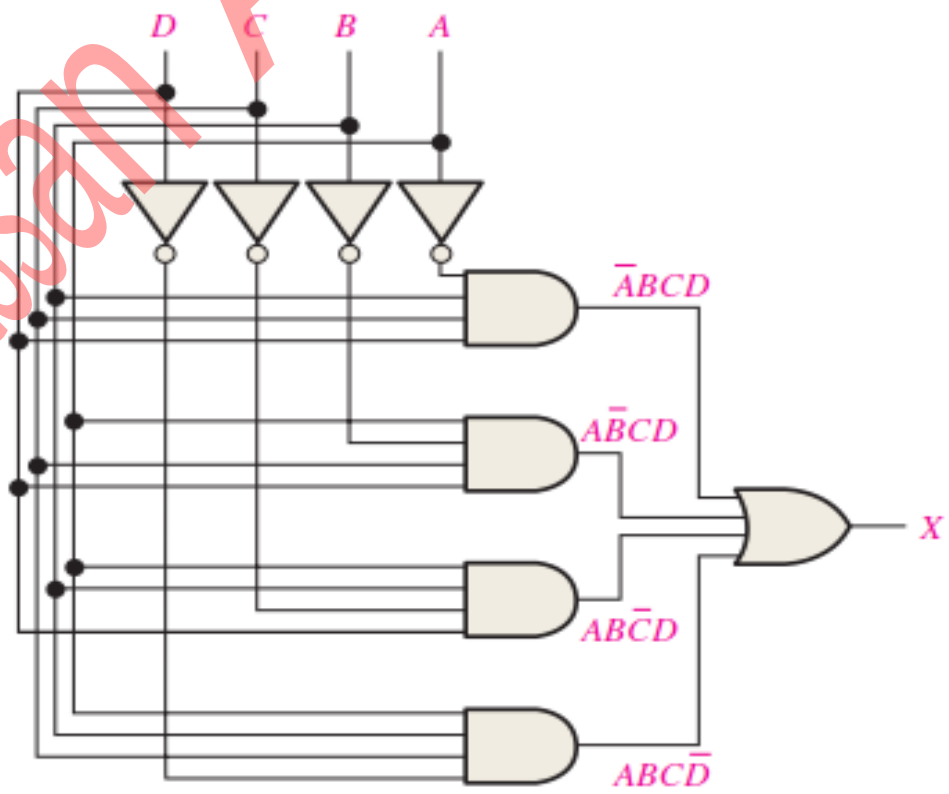
The combinations in which there are exactly three 1s are listed in Table.

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

The product terms are Ored to get the following expression:

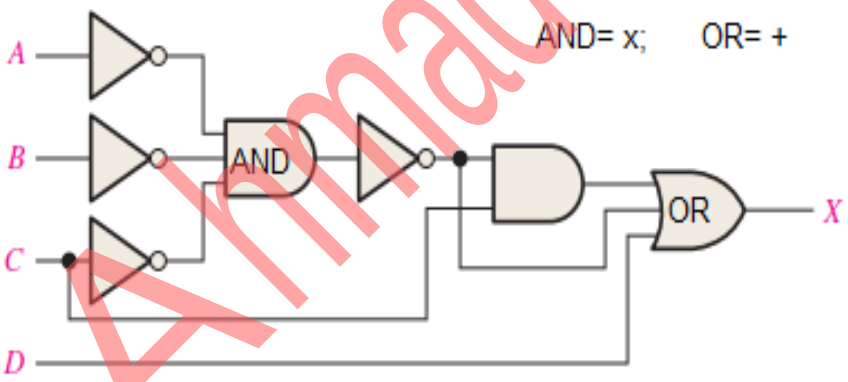
$$X = \bar{A}BCD + A\bar{B}CD + AB\bar{C}D + ABC\bar{D}$$

This expression is implemented in Fig. with AND-OR logic.



Example 8-4

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



Solution

The expression for the output of the circuit is:

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

Applying DeMorgan's theorem and Boolean algebra,

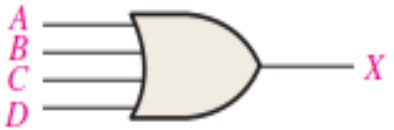
$$\begin{aligned}
X &= (\overline{\overline{A}} + \overline{\overline{B}} + \overline{\overline{C}})C + \overline{\overline{A}} + \overline{\overline{B}} + \overline{\overline{C}} + D && \text{(rul_9)}\\
&= AC + BC + CC + A + B + C + D\\
&= AC + BC + C + A + B + C + D && \text{(rul_7)}\\
&= AC + BC + C + A + B + D && \text{(rul_5)}\\
&= C(A + B + 1) + A + B + D && \text{(rul_2)}\\
&= C(A + 1) + A + B + D && \text{(rul_2)}\\
&= A + B + C + D && \text{(rul_4)}
\end{aligned}$$

Basic rules of Boolean algebra.

1. $A + 0 = A$	7. $A \cdot A = A$
2. $A + 1 = 1$	8. $A \cdot \overline{A} = 0$
3. $A \cdot 0 = 0$	9. $\overline{\overline{A}} = A$
4. $A \cdot 1 = A$	10. $A + AB = A$
5. $A + A = A$	11. $A + \overline{A}B = A + B$
6. $A + \overline{A} = 1$	12. $(A + B)(A + C) = A + BC$

A, *B*, or *C* can represent a single variable or a combination of variables.

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

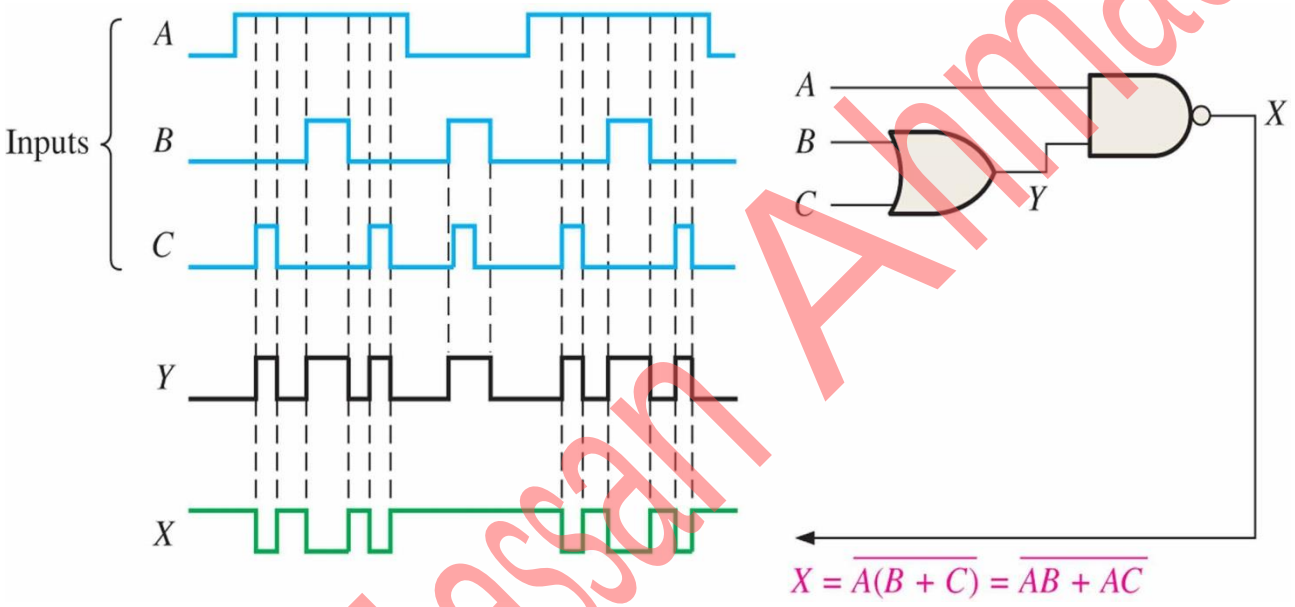


8-3. Pulse Waveform Operation

- The operation of any gate is the same regardless (بغض النظر) of whether its inputs are pulsed or constant levels.
- The nature of the inputs (pulsed or constant levels) does not alter (يغيّر) the truth table of a circuit.
- The following is a review of the operation of individual gates for use in analyzing combinational circuits with pulse waveform inputs:
  1. The output of an AND gate is HIGH only when all inputs are HIGH at the same time.
  2. The output of an OR gate is HIGH only when at least one of its inputs is HIGH.
  3. The output of a NAND gate is LOW only when all inputs are HIGH at the same time.
  4. The output of a NOR gate is LOW only when at least one of its inputs is HIGH.

Example 8-5

Determine the final output waveform X for the circuit in following Fig., with input waveforms A, B, and C as shown.

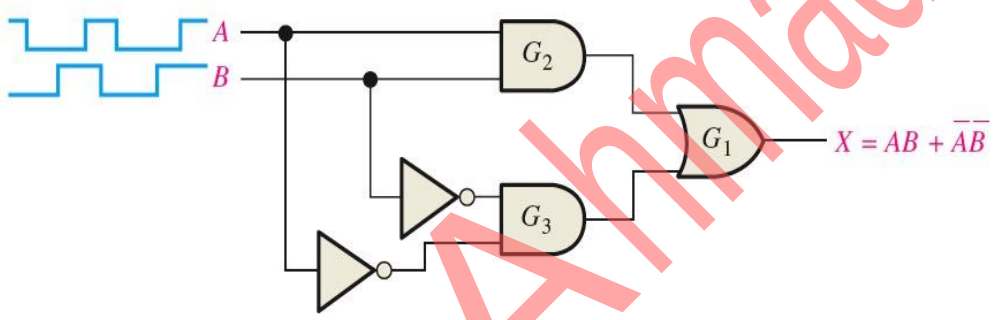


Solution

- ✓ The output expression, $\overline{AB + AC}$, indicates that the output X is LOW when:
  - both A and B are HIGH **or**
  - both A and C are HIGH **or**
  - all inputs are HIGH.
- ✓ The output waveform X is shown in the timing diagram of given Fig.
- ✓ The intermediate waveform Y at the output of the OR gate is also shown.

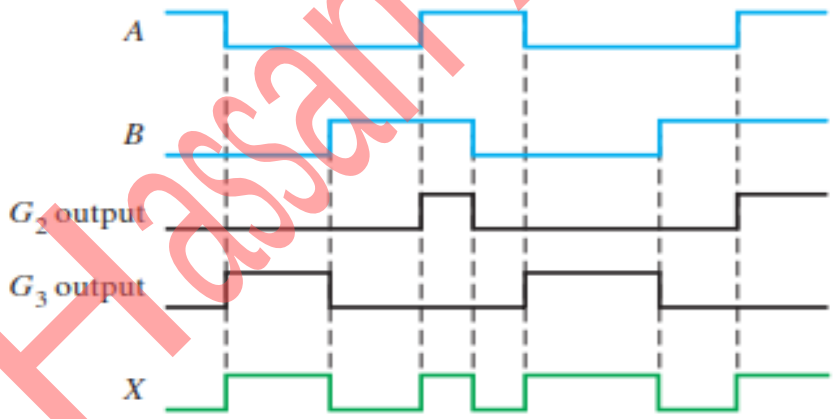
Example 8-6

Draw the timing diagram for the circuit in following Fig. showing the outputs of G_1, G_2, and G_3 with the input waveforms, A, and B, as indicated.



Solution

When both inputs are HIGH or when both inputs are LOW, the output X is HIGH as shown in following Fig.

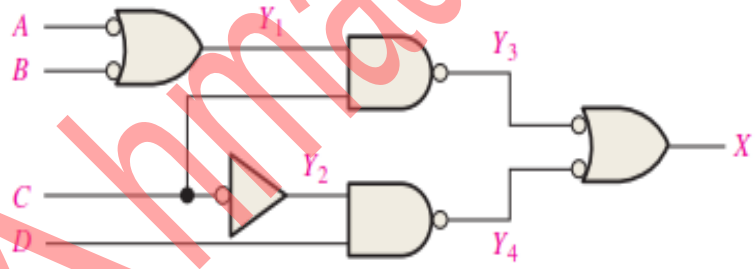


Notice that this is an exclusive-NOR circuit.

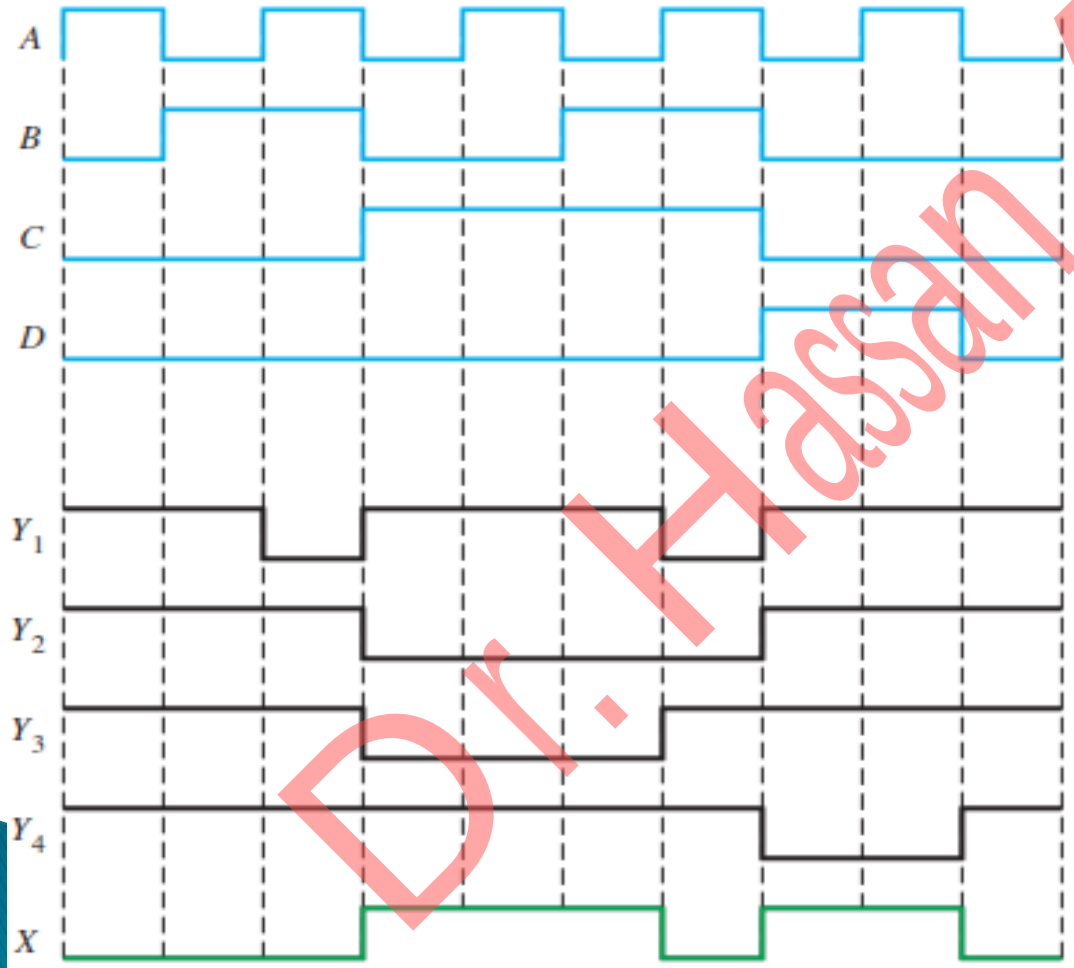
The intermediate outputs of gates G_2, and G_3 are also shown in Fig.

Example 8-7

Determine the output waveform X for the logic circuit with input waveforms in Figure by first finding the intermediate waveform at each of points Y_1, Y_2, Y_3, and Y_4.



Solution



Selected Key Terms

Universal gate	Either a NAND or a NOR gate. The term universal refers to a property of a gate that permits any logic function to be implemented by that gate or by a combination of gates of that kind.
Negative-OR	The dual operation of a NAND gate when the inputs are active-LOW.
Negative-AND	The dual operation of a NOR gate when the inputs are active-LOW.
Node	A common connection point in a circuit in which a gate output is connected to one or more gate inputs.

True/False Quiz

1. AND-OR logic can have only two 2-input AND gates.
2. AOI is an acronym for AND-OR-Invert.
3. If the inputs of an exclusive-OR gate are the same, the output is LOW (0).
4. If the inputs of an exclusive-NOR gate are different, the output is HIGH (1).
5. A parity generator cannot be implemented using exclusive-OR gates.
6. NAND gates can be used to produce the AND functions.
7. NOR gates cannot be used to produce the OR functions.
8. Any SOP expression can be implemented using only NAND gates.
9. Negative-OR is equivalent to NAND.

1. F 2. T 3. T 4. F 5. F
6. T 7. F 8. T 9. T

SELF-TEST

1. Assume an AOI expression is $\overline{AB + CD}$. The equivalent POS expression is

a. $(A + B)(C + D)$

b. $(\overline{A} + \overline{B})(\overline{C} + \overline{D})$

c. $\overline{(A + B)(C + D)}$

d. none of the above

2. The truth table shown is for

a. a NAND gate

b. a NOR gate

c. an exclusive-OR gate

d. an exclusive-NOR gate

Inputs		Output
A	B	X
0	0	1
0	1	0
1	0	0
1	1	1

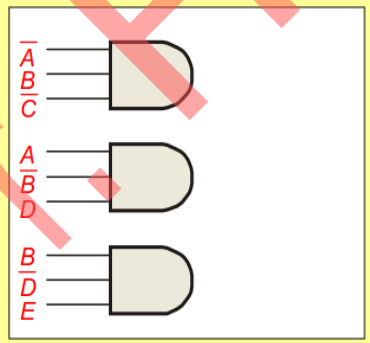
3. To implement the SOP expression $X = \overline{A}B\overline{C} + A\overline{B}D + B\overline{D}E$, the type of gate that is needed is a

a. 3-input AND gate

b. 3-input NAND gate

c. 3-input OR gate

d. 3-input NOR gate



4. Reading the Karnaugh map, the logic expression is

a. $\overline{A}\,\overline{C} + \overline{A}\,B$

b. $\overline{A}\,B + A\,\overline{C}$

c. $A\,\overline{B} + B\,\overline{C}$

d. $\overline{A}\,B + \overline{A}\,\overline{C}$

	$\overline{C}$	C
$\overline{A}\overline{B}$	1	
$\overline{A}B$	1	1
AB		
$A\overline{B}$		

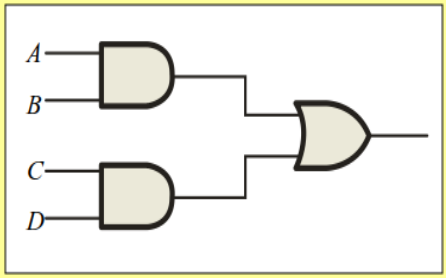
5. The circuit shown will have identical logic out if all gates are changed to

a. AND gates

b. OR gates

c. NAND gates

d. NOR gates



6. The two types of gates which are called *universal gates* are

a. AND/OR

b. NAND/NOR

c. AND/NAND

d. OR/NOR

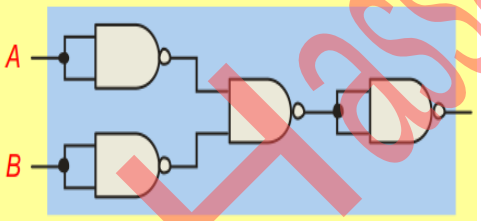
7. The circuit shown is equivalent to an

a. AND gate

b. XOR gate

c. OR gate

d. none of the above



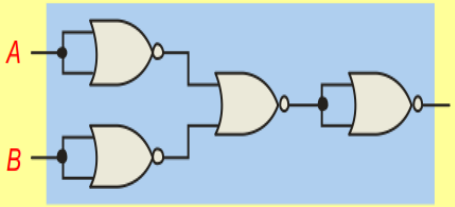
8. The circuit shown is equivalent to

a. an AND gate

b. an XOR gate

c. an OR gate

d. none of the above



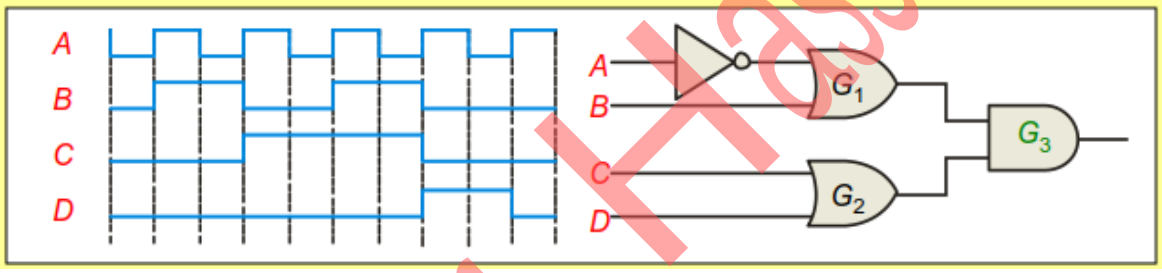
9. During the first *three* intervals for the pulsed circuit shown, the output of

a. G_1 is LOW and G_2 is LOW

b. G_1 is LOW and G_2 is HIGH

c. G_1 is HIGH and G_2 is LOW

d. G_1 is HIGH and G_2 is HIGH



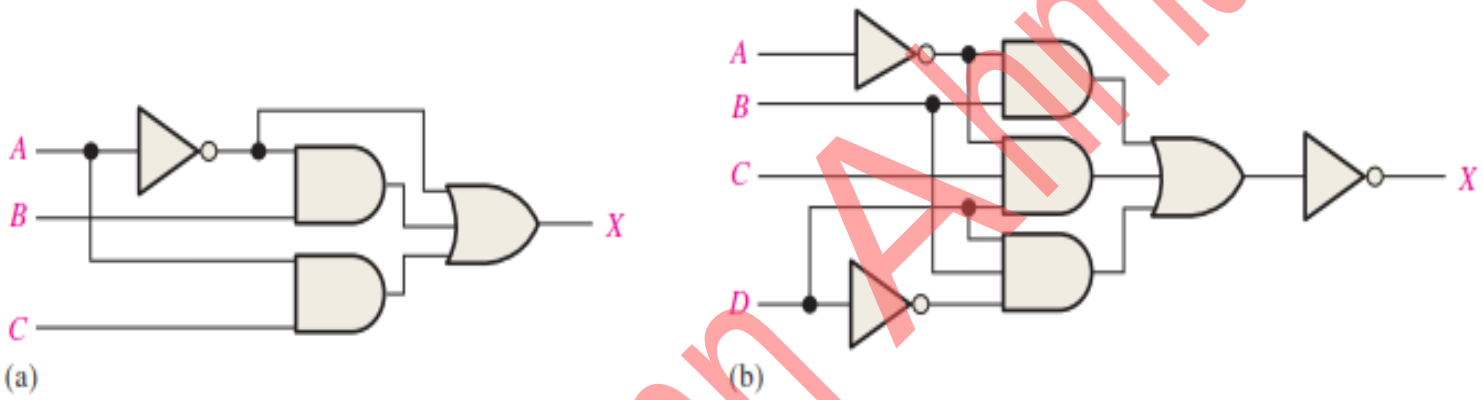
Answers:

1. b
2. d
3. c
4. d
5. c
6. b
7. c
8. a
9. c

Problems & Solutions

Prob. 5-1

Write the output expression for each circuit in Figure.



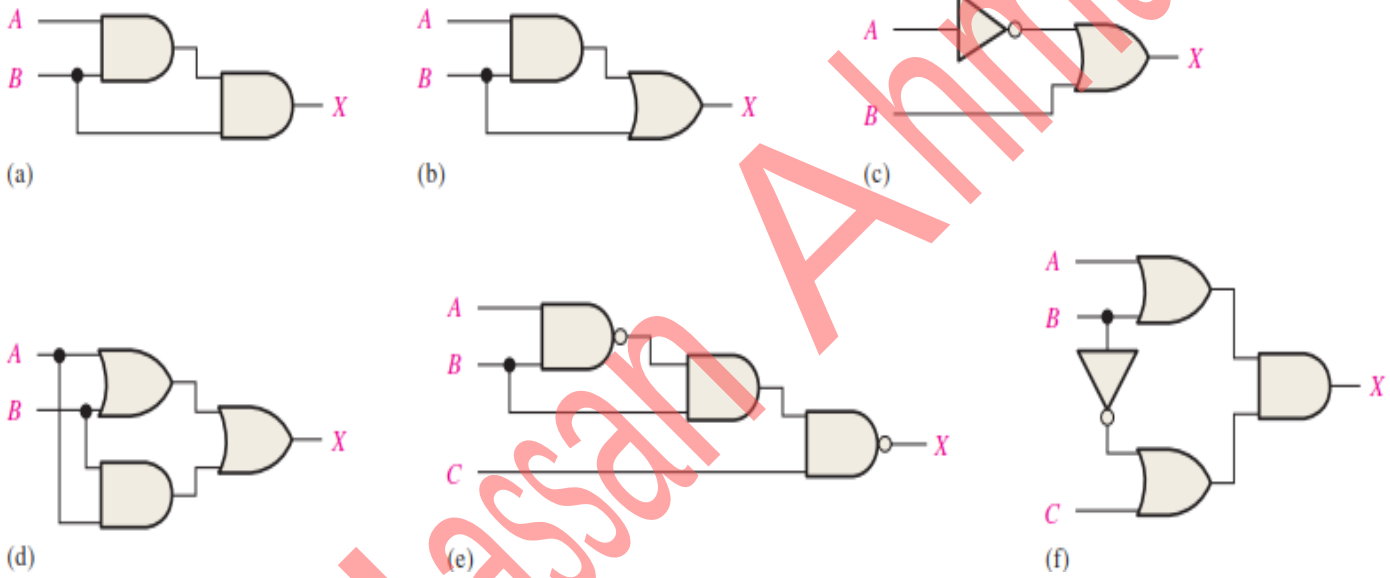
Sol.

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

(b) $X = \overline{\overline{A}\overline{B} + \overline{A}CD + DB\overline{D}}$

Prob. 5-2

Write the output expression for each circuit in Figure.



Sol.

(a) $X = ABB$

(b) $X = AB + B$

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

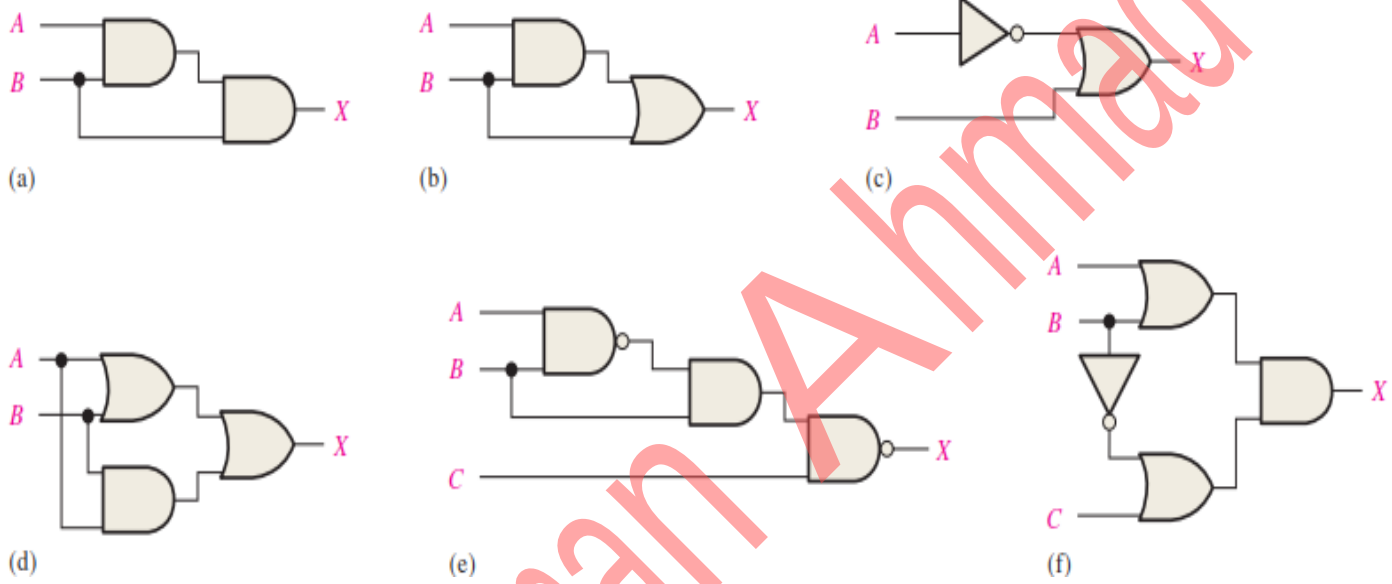
(d) $X = (A + B) + AB$

(e) $X = \overline{\overline{A}BC}$

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

Prob. 5-3

Develop the truth table for each circuit in Figure.



Sol.

(a) $X = ABB$

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

(b) $X = AB + B$

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

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

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

(d) $X = (A + B) + AB$

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

(e) $X = \overline{\overline{A}BC}$

A	B	C	X
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

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

A	B	C	X
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

Prob. 5-4 Use AND gates, OR gates, and inverters as needed to implement the following logic expressions as stated

Sol.

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

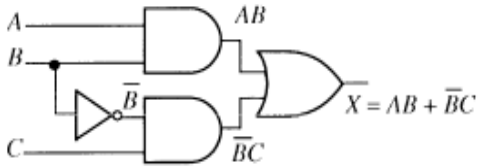
(b) $X = A(B + \overline{C})$

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

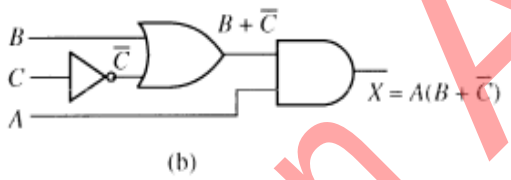
(d) $X = \overline{ABC} + B(EF + \overline{G})$

(e) $X = A[BC(A + B + C + D)]$

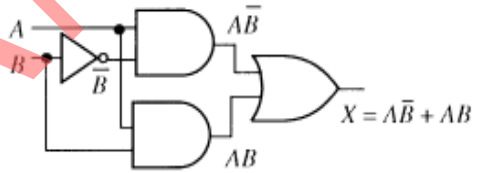
(f) $X = B(C\overline{D}E + \overline{E}FG)(\overline{AB} + C)$



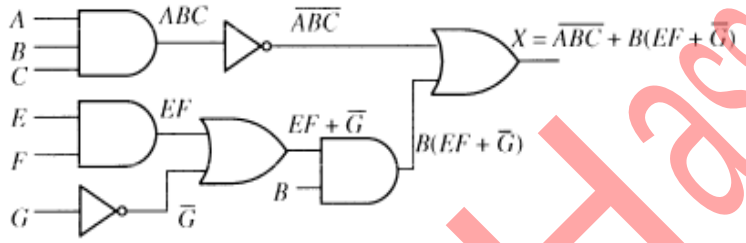
(a)



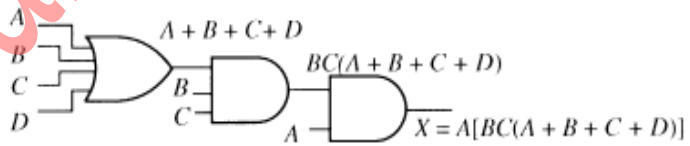
(b)



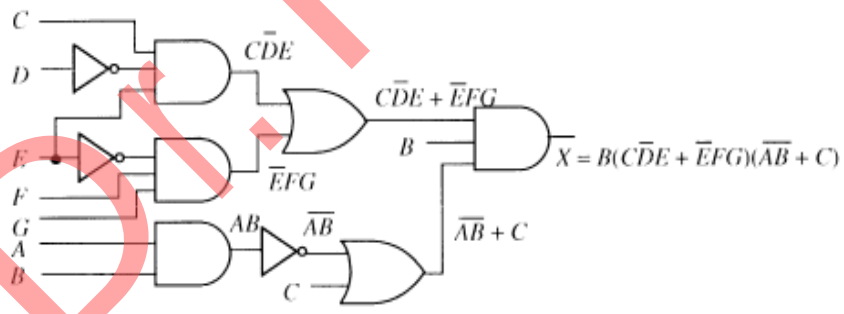
(c)



(d)



(e)



(f)

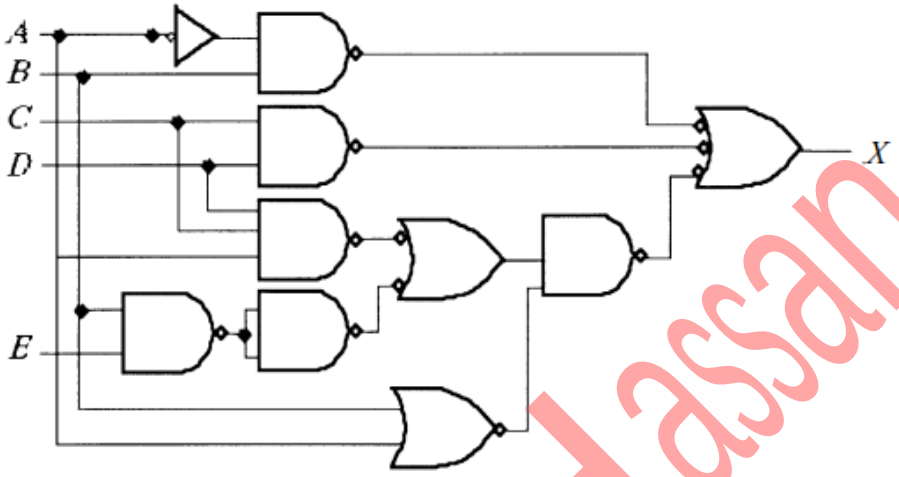
Prob. 5-5 Use NAND gates, NOR gates, or combinations of both to implement the following logic expressions as stated:

Sol.

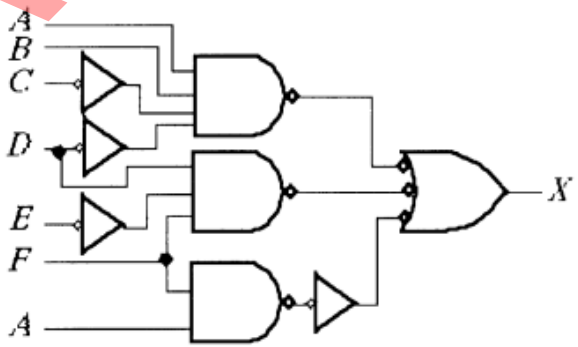
(a) $X = \overline{A}B + CD + \overline{(A + B)}(ACD + \overline{BE})$

(b) $X = AB\overline{C}\,\overline{D} + D\overline{E}F + \overline{AF}$

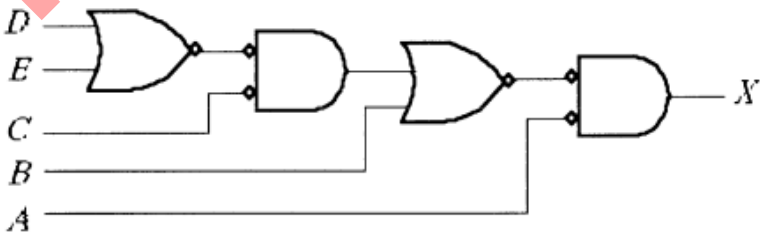
(c) $X = \overline{A}[B + \overline{C}(D + E)]$



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



(b) $X = AB\overline{C}\,\overline{D} + D\overline{E}F + \overline{AF}$



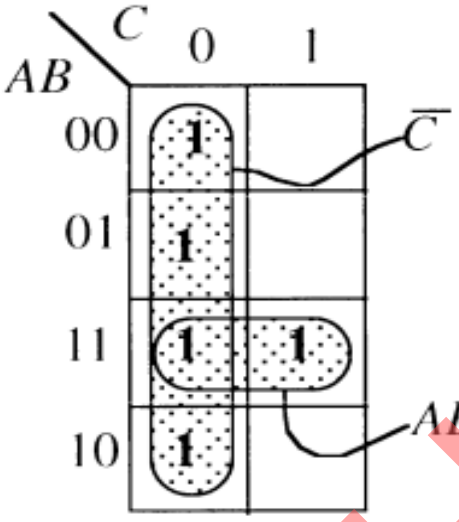
(c) $X = \overline{A}[B + \overline{C}(D + E)]$

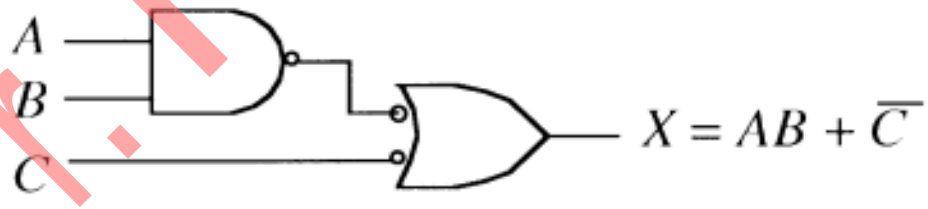
Prob. 5-6 Using Karnough map, implement a logic circuit for the truth table in Table.

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

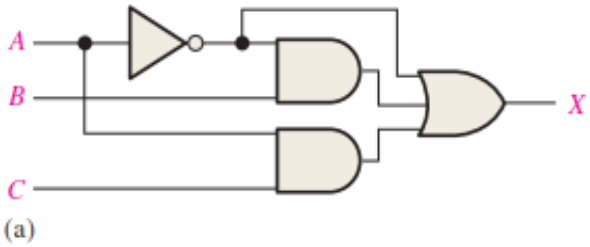
Sol.

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

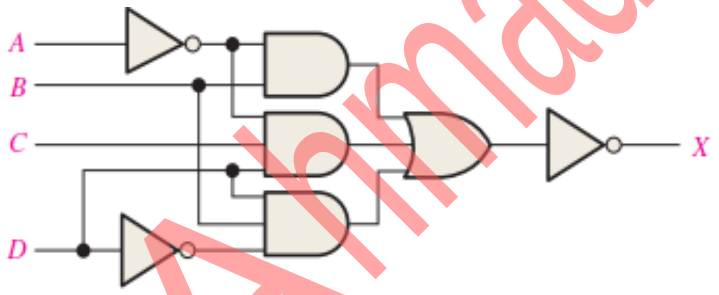




Prob. 5-6 Implement the logic circuits in Figure using only NAND gates, and also using only NOR gates.



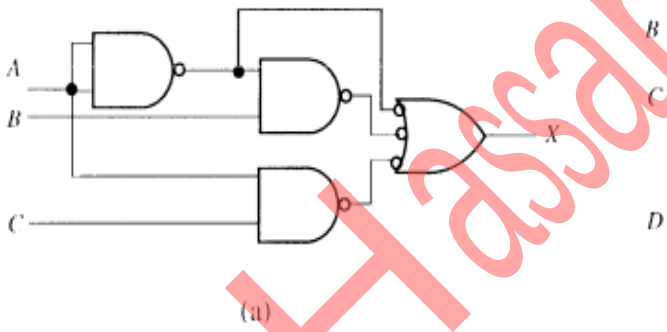
(a)



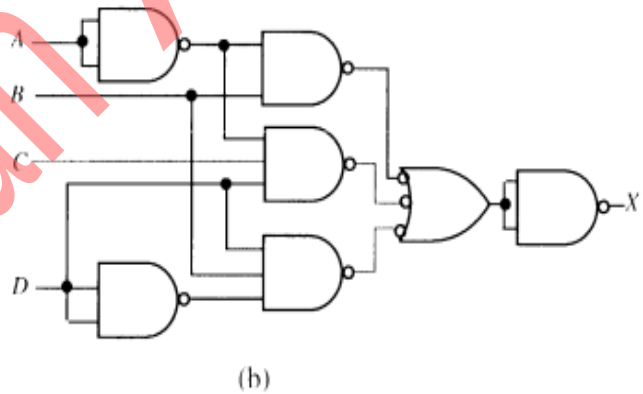
(b)

Sol.

NAND:

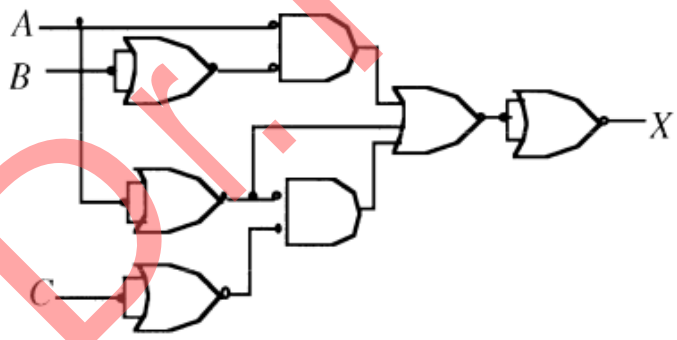


(a)

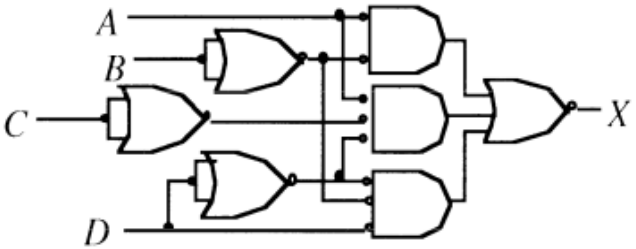


(b)

NOR:

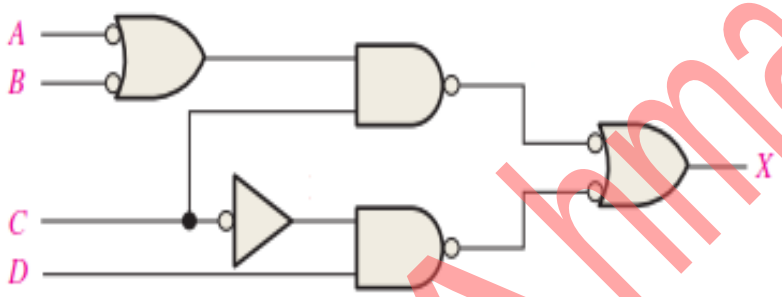


(a)



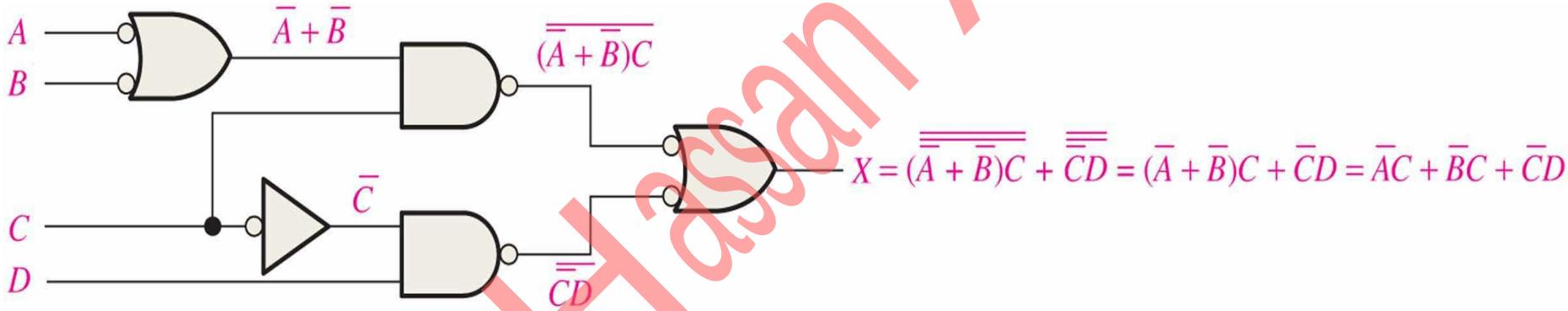
(b)

Prob. 5-7 Determine the output waveform X for the circuit in Fig., directly from the output expression.



Sol.

- The output expression for the circuit is developed in following Fig.

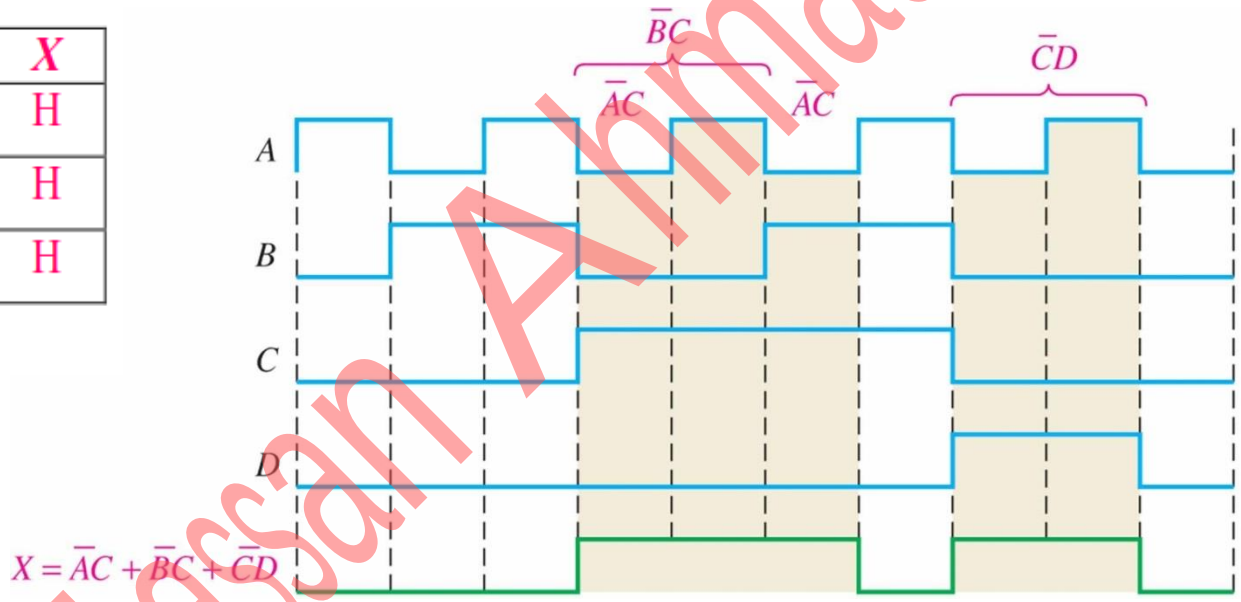


- **Result**: The SOP form indicates that the output is HIGH when A is LOW and C is HIGH **or** when B is LOW and C is HIGH **or** when C is LOW and D is HIGH, regardless (بغض النظر) of values of another variables in each state.

- The result is shown in following Fig. and is the same as the one obtained by the intermediate-waveform method in Example 8-7.

A	B	C	D	Product	X
L	0	H	0	$\overline{A}C = 1$	H
0	L	H	0	$\overline{B}C = 1$	H
0	0	L	H	$\overline{C}D = 1$	H

Each state is may be repeated.



- The corresponding product terms for each waveform condition that results in a HIGH output are indicated.

The end of Lecture_08, chapter 5