UNIT :ii LOGIC GATES

BY

BHANUDAS NARWADE

ASST. PROF.

DEGLOOR COLLEGE,DEGLOOR

LOGIC GATES:

Logic Gate: Is an electronic circuit and are the basic building blocks of digital electronics.

It has one or more inputs and only one output.

Out put (0 or 1) depends on certain combinations of input

BASIC LOGIC GATES:

The basic logic gates are

NOT, AND and OR

DERIVED LOGIC GATES:

Derived logic gates are

NAND,NOR,X-OR X-NOR

Universal gates: NAND, NOR

BASIC GATES:

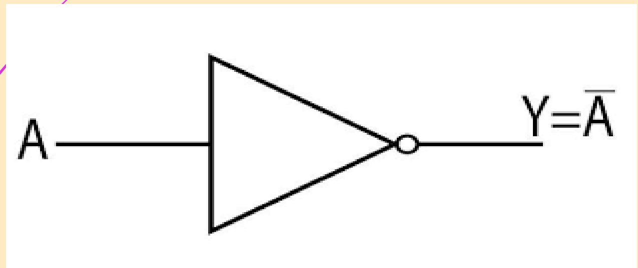
NOT GATE (INVERTER):

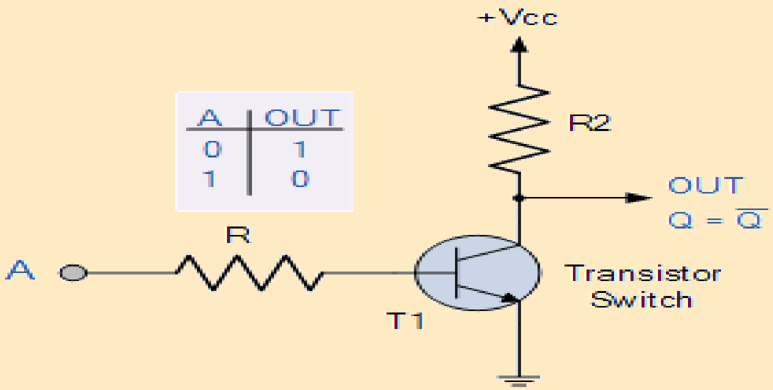
NOT also called Inverter

Used to complement or invert digital signal

To change one logic level to opposite

SYMBOL:





TRUTH TABLE:

Input	Output
A	Y
0	1
1	0

BASIC GATES:

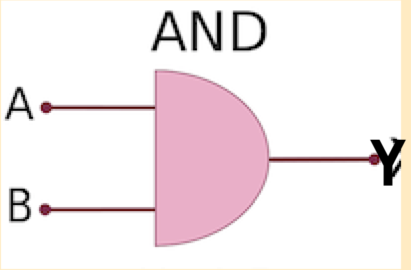
AND GATE :

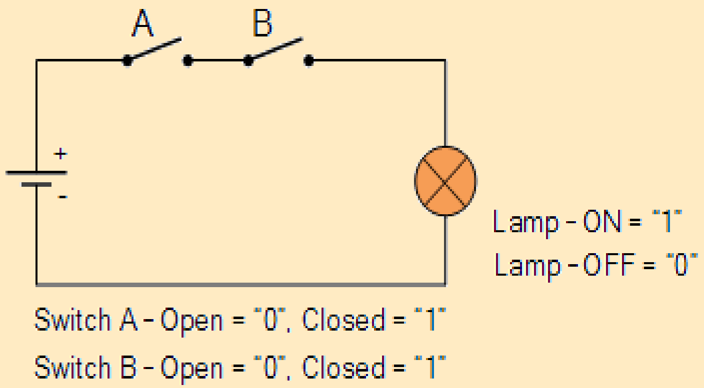
Logical multiplication

Two or more input and only one output

Boolean expression: Y=A.B

SYMBOL:





TRUTH TABLE:

A	B	Y
0	0	0
1	0	0
0	1	0
1	1	1

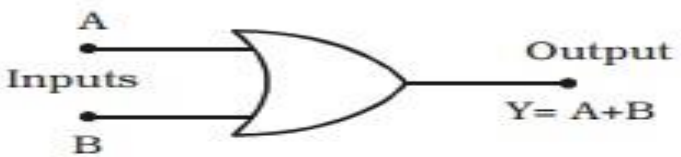
BASIC GATES:

OR GATE :

Logical addition

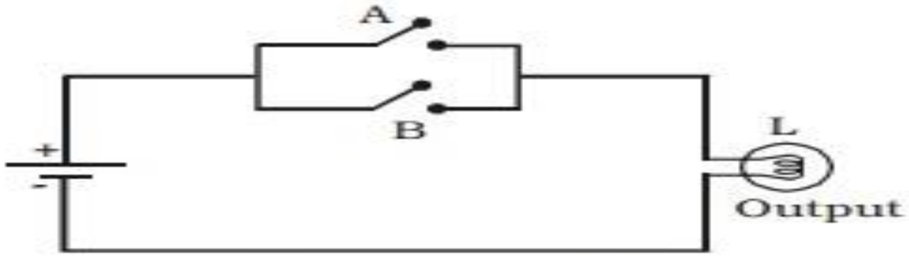
Two or more input and only one output

Boolean expression: Y=A+B



(a) Logic Symbol

Fig



(b)Electrical circuit

OR gate

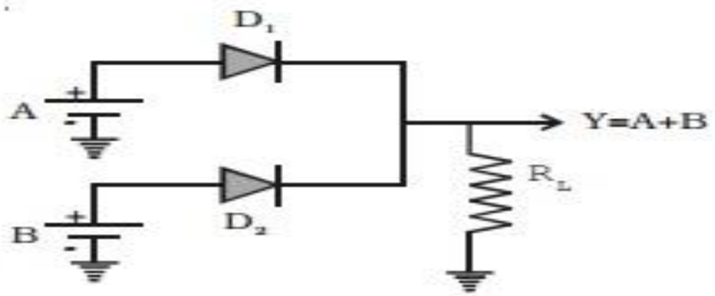


Fig *OR gate using diodes*

Table Truth table of OR gate

Inputs		Output
A	B	Y = A + B
0	0	0
0	1	1
1	0	1
1	1	1

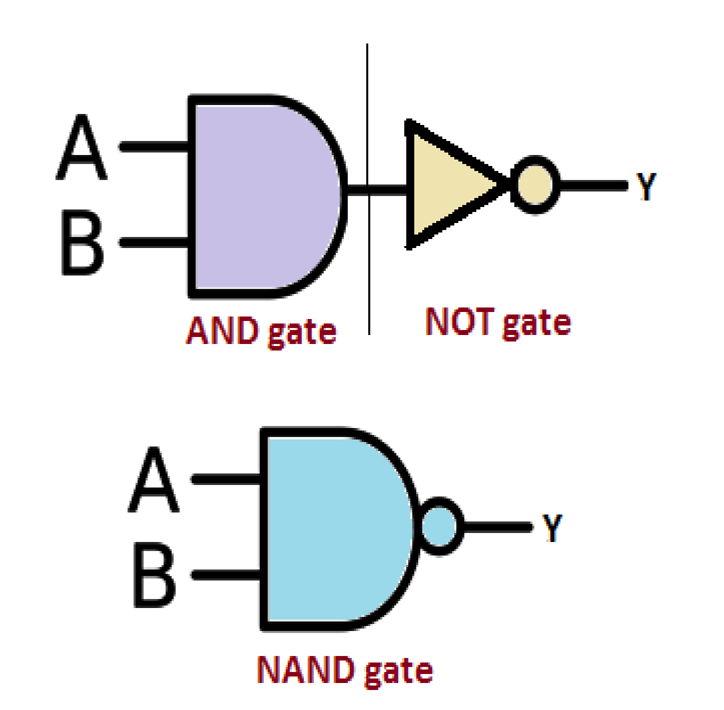
UNIVERSAL GATES:

NAND GATE :

Universal gate due to versatile in nature

Two or more input and only one output

Boolean expression: $Y=\overline{AB}$



Input		Output
A	B	$Y=\overline{A.B}$
0	0	1
0	1	1
1	0	1
1	1	0

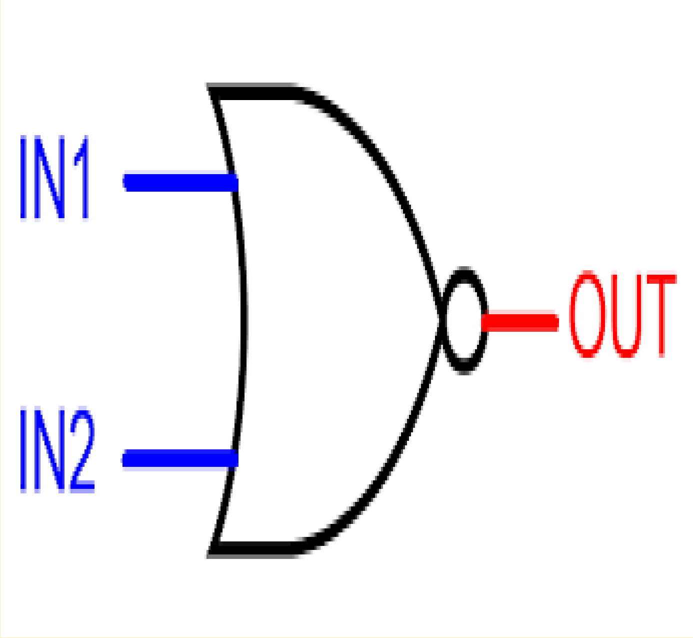
UNIVERSAL GATES:

NOR GATE :

Universal gate due to versatile in nature

Two or more input and only one output

Boolean expression: $Y=\overline{A+B}$



Input		Output
A	B	$\overline{A+B}$
0	0	1
0	1	0
1	0	0
1	1	0

X-OR GATE:

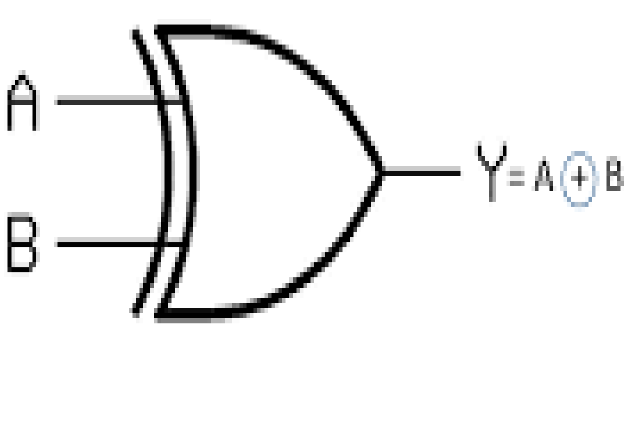
EXCLUSIVE-OR GATE :

Exclusive OR operation widely used in digital circuit

Two or more input and only one output

Can not be performed using basic gates

Boolean expression: $Y=\bar{A}B+A\bar{B}$



EX-OR gate symbol

A	B	$Y=A\oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

EX-OR gate truth table

EX-NOR GATE:

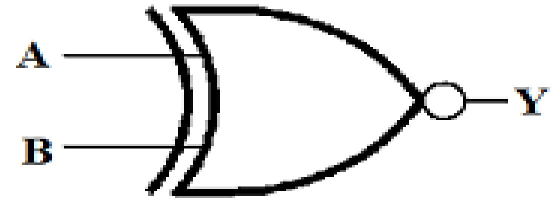
EXCLUSIVE-NOR GATE :

Exclusive NOR is complement of Ex-OR

Two or more input and only one output

Can not be performed using basic gates

Boolean expression: $Y = \bar{A}\,\bar{B} + A\,B$



inputs		output
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	1

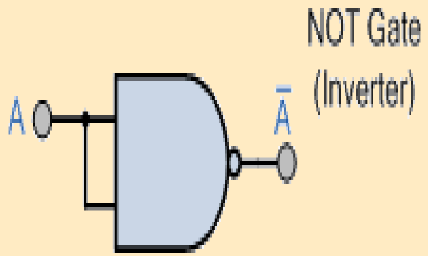
UNIVERSAL PROPERTIES OF NAND:

All digital logic can be performed by using NAND gate

1) NOT from NAND:

BY connecting both or all inputs of NAND together

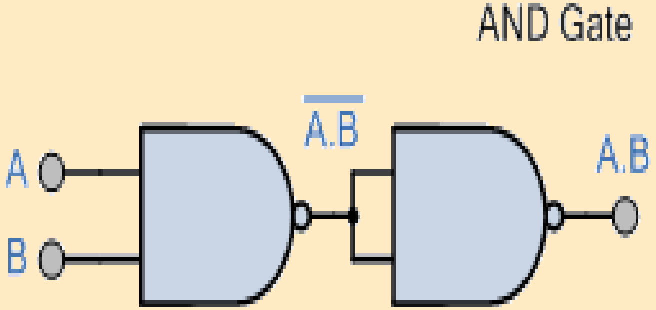
For two input NAND Boolean expression is $Y=\overline{A.A}.=\overline{A}$



2) AND using NAND

Inverting the output of NAND gate AND operation will be formed

$Y=\overline{A.B} = A.B$



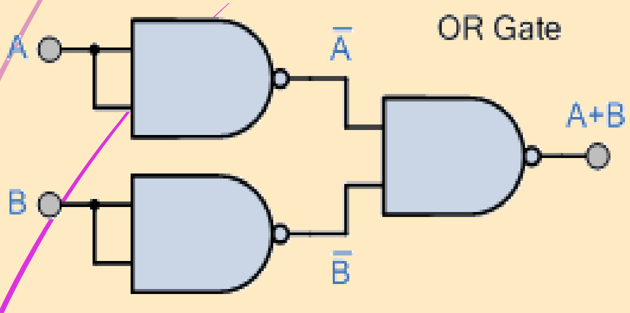
UNIVERSAL PROPERTIES OF NAND:

3) OR Gate from NAND Gate:

Inputs are inverted before applied to another NAND gate

For two input NAND Boolean expression is

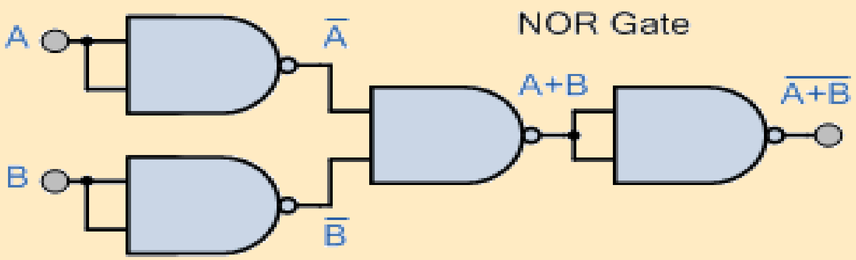
$$Y=\overline{\overline{A}.\overline{B}}.=\overline{\overline{A}}+\overline{\overline{B}}=A+B$$



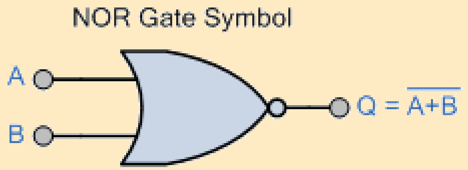
4) NOR using NAND

Out put of OR is inverted to get NOR

$$Y=\overline{\overline{\overline{A}.\overline{B}}}=\overline{\overline{\overline{A}}+\overline{\overline{B}}}=\overline{A+B}$$



UNIVERSAL PROPERTIES OF NOR:



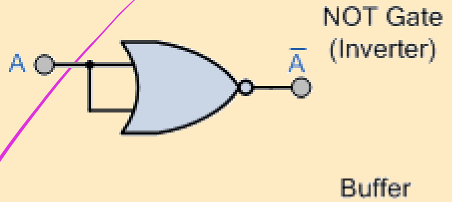
1) NOT Gate from NOR Gate:

Connecting all inputs of NOR gate

For two input NOR Boolean expression is

$$Y=\overline{A+A}=\overline{A}$$

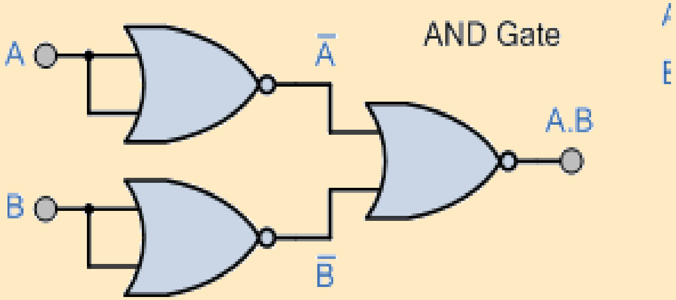
Input		Output
A	B	$\overline{A+B}$
0	0	1
0	1	0
1	0	0
1	1	0



2) AND using NOR

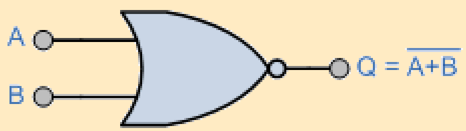
AND gate can be formed using NOR gate with inverted inputs

$$Y=\overline{\overline{A}+\overline{B}}=\overline{\overline{A}}.\overline{\overline{B}}=A.B$$



UNIVERSAL PROPERTIES OF NOR:

NOR Gate Symbol



Input		Output
A	B	$\overline{A+B}$
0	0	1
0	1	0
1	0	0
1	1	0

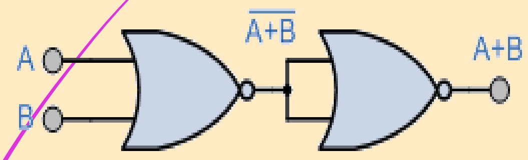
3) OR Gate from NOR Gate:

Out put of NOR is inverted

For two input NOR Boolean expression is

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

OR Gate

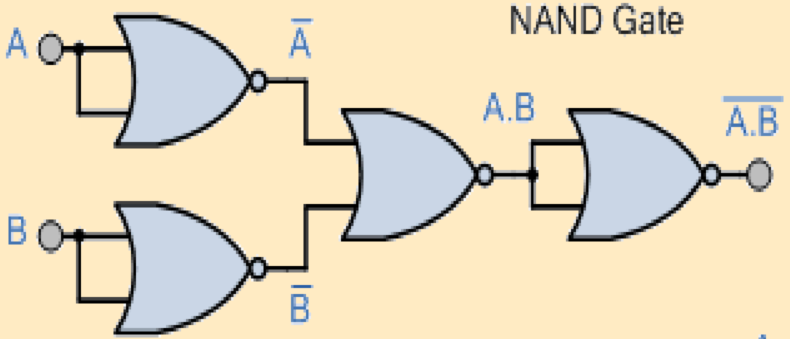


4) NAND using NOR

NAND gate can be formed using NOR gate with expressions

$$Y=\overline{\overline{\overline{A}+\overline{B}}} = \overline{\overline{\overline{A}}.\overline{\overline{B}}} = \overline{A.B}$$

NAND Gate



LOGIC OPERATORS :

1) AND OPERATOR:

In Boolean algebra AND operator is denoted by(.) or no operator symbol at all

Similar to multiplication in ordinary algebra

For example A.B = Y

A	B	Y=A.B
0	0	0
0	1	0
1	0	0
1	1	1

2) OR OPERATOR:

Indicated by + sign

Similar to addition in ordinary algebra

For example A+B =Y

A	B	Y=A+B
0	0	0
0	1	1
1	0	1
1	1	1

LOGIC OPERATORS :

3) NOT OPERATOR:

Used for complement or inversion

Symbol is bar on over variable

For example $A=\bar{A}$

$$\bar{\bar{A}} = A$$

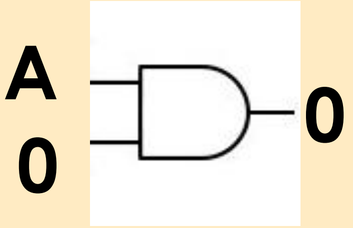
A	$\bar{A}$
0	1
1	0

BOOLEAN ALGEBRA LAWS AND RULES :

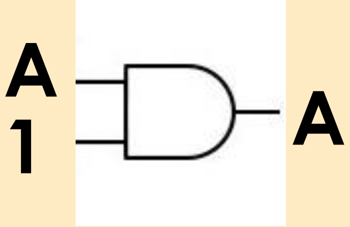
1) AND LAWS:

A

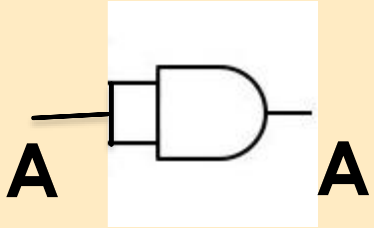
Law 1 : A.0 =0



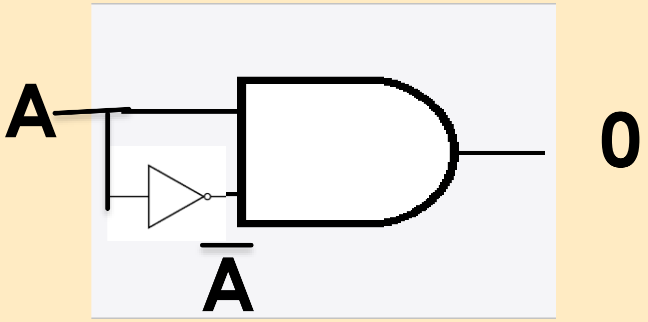
Law 2: A.1=A



Law 3: A.A=A



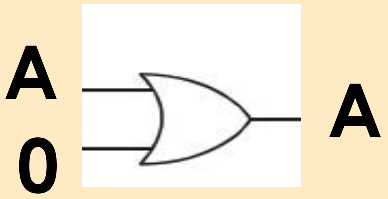
Law 4: $A.\overline{A}=0$



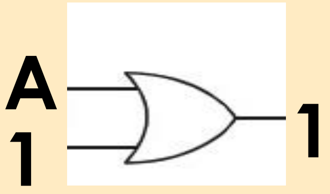
BOOLEAN ALGEBRA LAWS AND RULES :

1) OR LAWS:

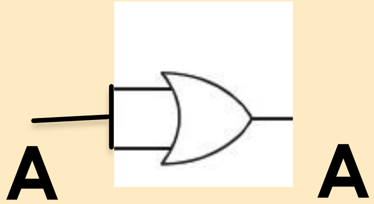
Law 5 : $A+0=A$



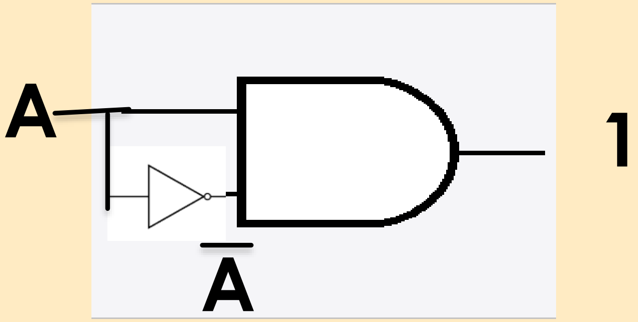
Law 6: $A+1=1$



Law 7: $A+A=A$



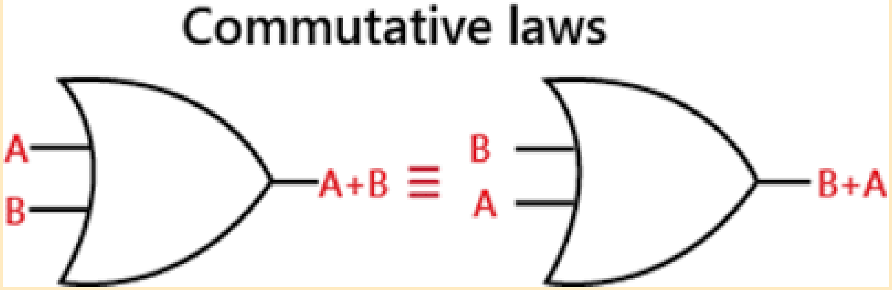
Law 8: $A+\bar{A}=1$



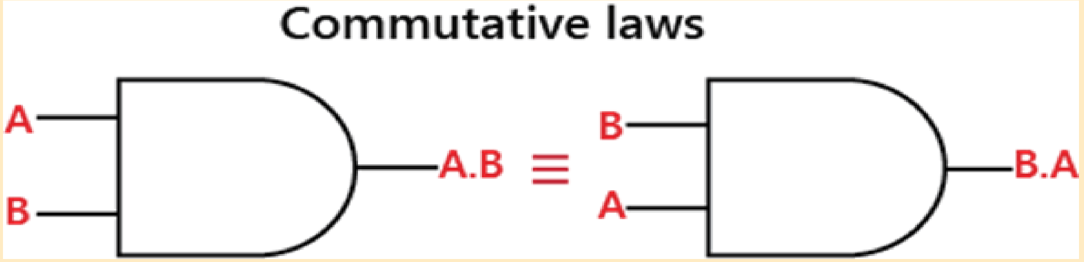
BOOLEAN ALGEBRA LAWS AND RULES :

Commutative Law: order of variables in OR and AND operation is insignificant(allows change of position of variable)

Law 9 (Commutative law of addition) : A + B = B + A

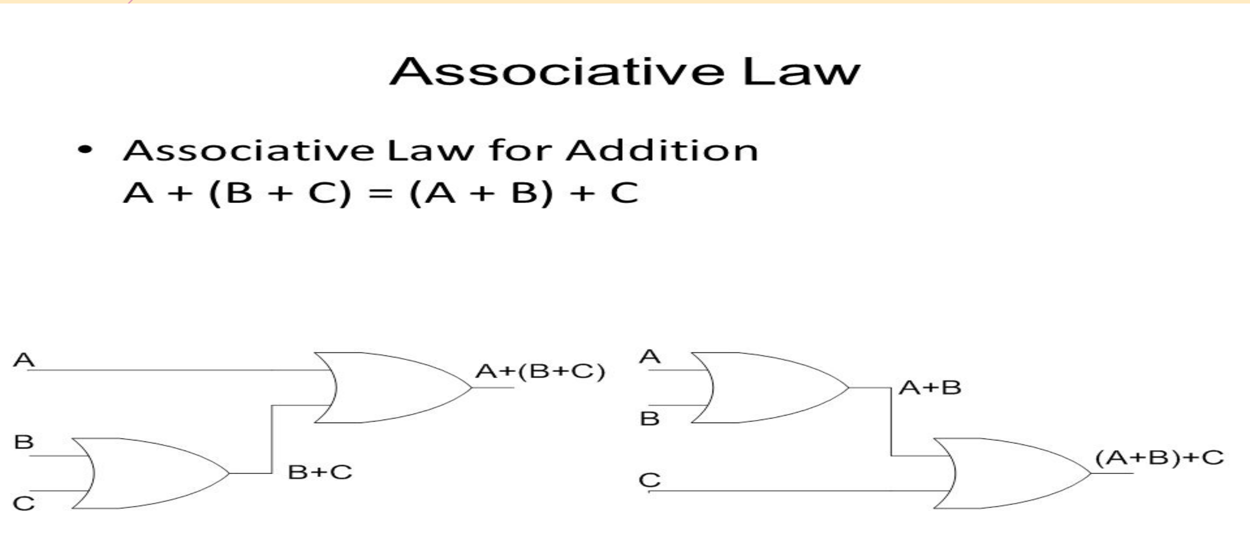


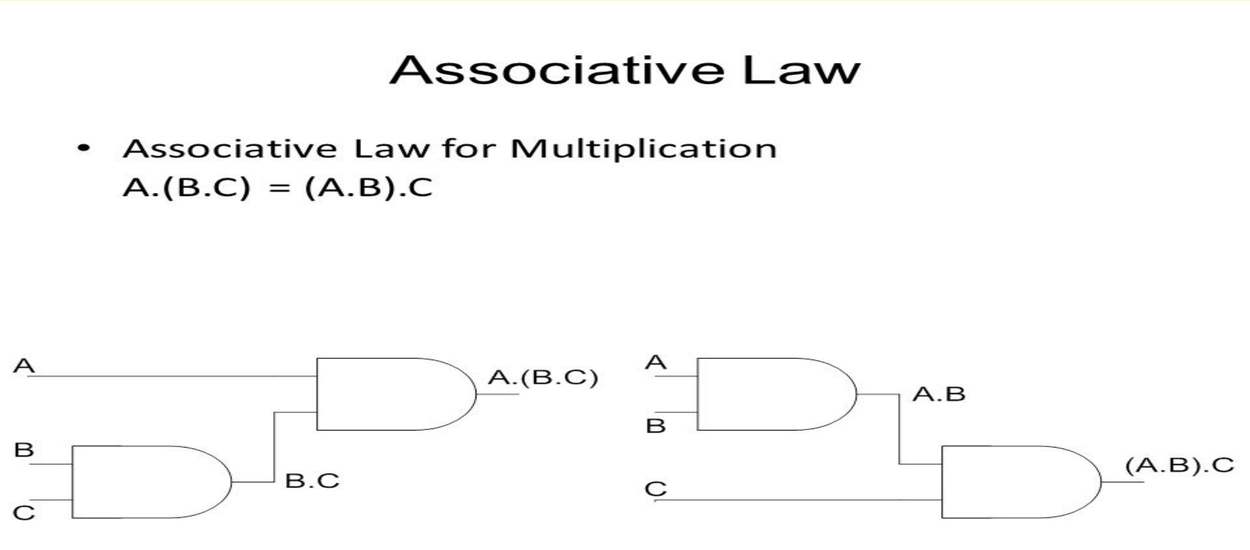
Law 10 (Commutative law of multiplication) : A . B = B . A



BOOLEAN ALGEBRA LAWS AND RULES :

Associative Law: order of grouping in OR and AND operation is insignificant(allows change of grouping)



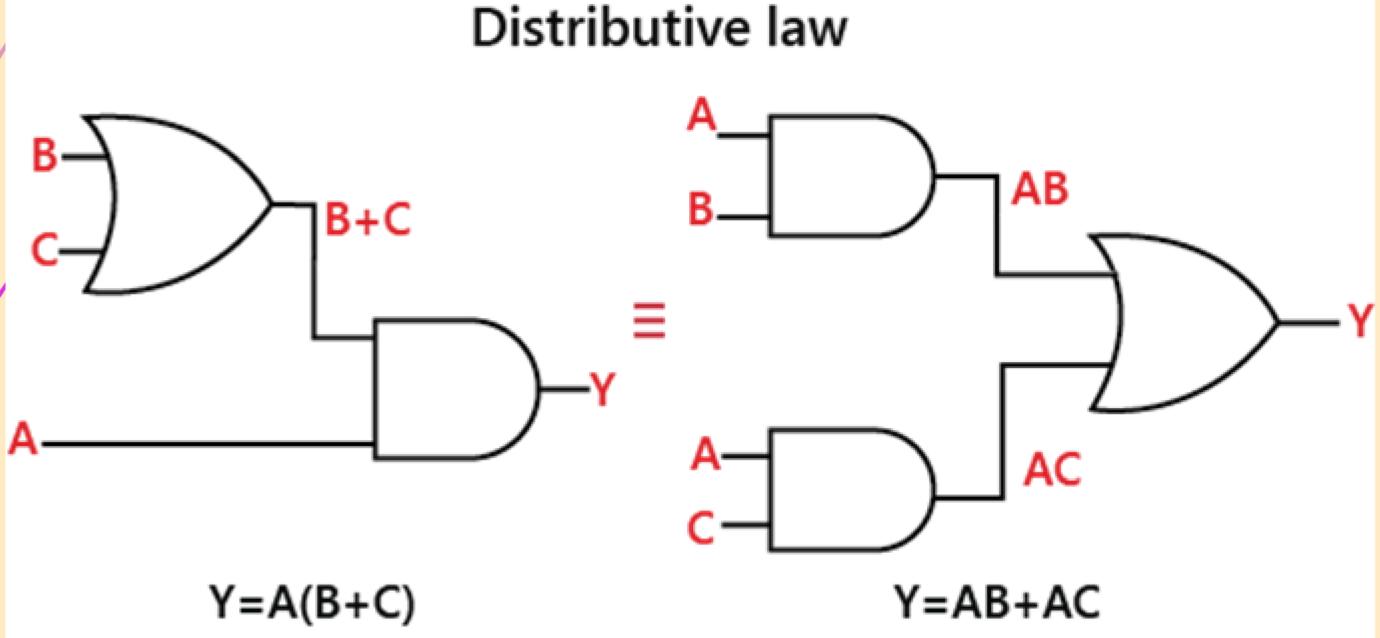


BOOLEAN ALGEBRA LAWS AND RULES :

Distributive Law: factoring or multiplying of different terms in an expression is allowed.

Law 13 : $A . (B + C) = AB + AC$

Law 14 : $A + (B . C) = (A + B). (A + C)$



DE-MORGAN'S THEOREMS :

Demorgan's first Theorem:

Statement: Complement of sum is equal to the product of complement

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

Proof:

A	B	$\bar{A}$	$\bar{B}$	A+B	$\overline{A+B}$	$\bar{A} \cdot \bar{B}$
0	0	1	1	0	1	1
0	1	1	0	1	0	0
1	0	0	1	1	0	0
1	1	0	0	1	0	0

DE-MORGAN'S THEOREMS :

Demorgan's Second Theorem:

Statement: Complement of product is equal to the sum of complement

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

Proof:

A	B	$\overline{A}$	$\overline{B}$	$A \cdot B$	$\overline{A \cdot B}$	$\overline{A} + \overline{B}$
0	0	1	1	0	1	1
0	1	1	0	0	1	1
1	0	0	1	0	1	1
1	1	0	0	1	0	0

SIMPLIFICATION OF BOOLEAN EXPRESSIONS:

Ex.2) $y = \overline{A\bar{B} + C}$

Solution $y = \overline{A\bar{B} + C}$

$$= \overline{A\bar{B}} \cdot \bar{C}$$

$$= (\bar{A} + B) \cdot \bar{C}$$

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

Ex.3) $X = [(A + \bar{B})\,(B + C)] \cdot B$

Solution $X = [(A + \bar{B})\,(B + C)] \cdot B$

$$= (A\,B + A\,C + \bar{B}\,B + \bar{B}\,C) \cdot B$$

$$= (A\,B + A\,C + 0 + \bar{B}\,C) \cdot B$$

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

$$= A\,B + A\,B\,C + 0$$

$$= A\,B\,(1 + C)$$

$$= A\,B\,(1)$$

$$= A\,B$$

Ex.4) $X = \overline{A\,B} \cdot \overline{B + C}$

Solution: $x = \overline{A\,B} \cdot \overline{B + C}$

$$= (\bar{A} + \bar{B}) \cdot (\bar{B} \cdot \bar{C})$$

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

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

$$= \bar{B}\,\bar{C}\,[\bar{A} + 1]$$

$$= \bar{B}\,\bar{C}\,[1]$$

$$= \bar{B}\,\bar{C}$$

SIMPLIFICATION OF BOOLEAN EXPRESSIONS:

Ex.5) $X = \overline{(AB \cdot \overline{C + D}) \cdot AB}$

Solution

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

Ex.(6) $y = \overline{\overline{AB} + \overline{A} + AB}$

Solution

$$
\begin{aligned}
y &= \overline{\overline{AB} + \overline{A} + AB} \\
&= \overline{\overline{A} + \overline{B} + \overline{A} + AB} \\
&= \overline{\overline{A} + \overline{A} + \overline{B} + AB} \\
&= \overline{\overline{A} + \overline{B} + AB} \\
&= \overline{\overline{A} + \overline{B} + A} \\
&= \overline{\overline{A} + A + \overline{B}} \\
&= \overline{1 + \overline{B}} \\
&= \overline{1} \\
&= 0
\end{aligned}
$$

SIMPLIFICATION OF BOOLEAN EXPRESSIONS:

Ex.7) $y = AB + \overline{AC} + A\overline{B}C(AB + C)$

Solution

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

$= AB + \overline{AC} + A\overline{B}CAB + A\overline{B}CC$

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

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

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

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

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

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

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

$= \overline{A} + \overline{C} + 1$

$= 1$

Ex 8) $y = W\overline{X}(W + Y) + WY(\overline{W} + \overline{X})$

Solution:

$y = W\overline{X}(W + Y) + WY(\overline{W} + \overline{X})$

$= W\overline{X}W + W\overline{X}Y + WY\overline{W} + WY\overline{X}$

$= W\overline{X} + W\overline{X}Y + 0 + WY\overline{X}$

$= W\overline{X}(1 + Y + Y)$

$= W\overline{X}(1)$

$= W\overline{X}$

SIMPLIFICATION OF BOOLEAN EXPRESSIONS:

Ex. 9)

$$
\begin{aligned}
y &= A + \bar{B}C(A + \overline{\bar{B}C}) \\
&= A + \bar{B}C(A + \bar{\bar{B}} + \bar{C}) \\
&= A + \bar{B}C(A + B + \bar{C}) \\
&= A + \bar{B}CA + \bar{B}CB + \bar{B}C\bar{C} \\
&= A + \bar{B}CA + 0 + 0 \\
&= A + A\bar{B}C \\
&= A(1 + \bar{B}C) \\
&= A(1) \\
&= A
\end{aligned}
$$

Ex.10) $y = A[B + C(\overline{AB + AC})]$

Solution

$$
\begin{aligned}
y &= A[B + C(\overline{AB + AC})] \\
&= A[B + C(\overline{AB} \cdot \overline{AC})] \\
&= A[B + C(\bar{A} + \bar{B}) \cdot (\bar{A} + \bar{C})] \\
&= A[B + (\bar{A} + \bar{B}) \cdot \bar{A}C] \\
&= A[B + \bar{A}\bar{A}C + \bar{A}\bar{B}C] \\
&= A[B + \bar{A}C + \bar{A}\bar{B}C] \\
&= AB + A\bar{A}C + A\bar{A}\bar{B}C \\
&= AB + 0 + 0 \\
&= AB
\end{aligned}
$$

SUM –OF –PRODUCT FORM(SOP):

Product of two or more variables or their complements is AND function of those variables

Product of two variable can be AB, for three variable ABC, so on

Sum in Boolean algebra is same as OR function

SOP expression is two or more AND fuctions OR ed together

For example AB + CD is SOP expression

AB+ BCD

ABC + DEF

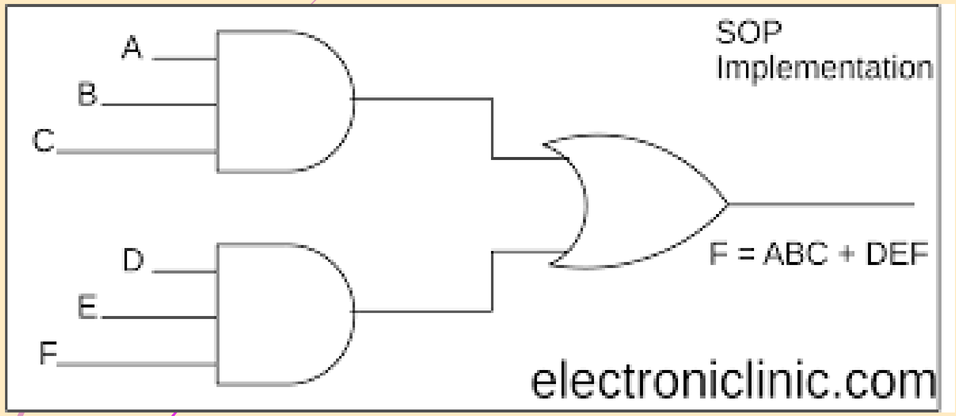
$A\overline{B}C+D\overline{E}F\overline{G}+AEF$

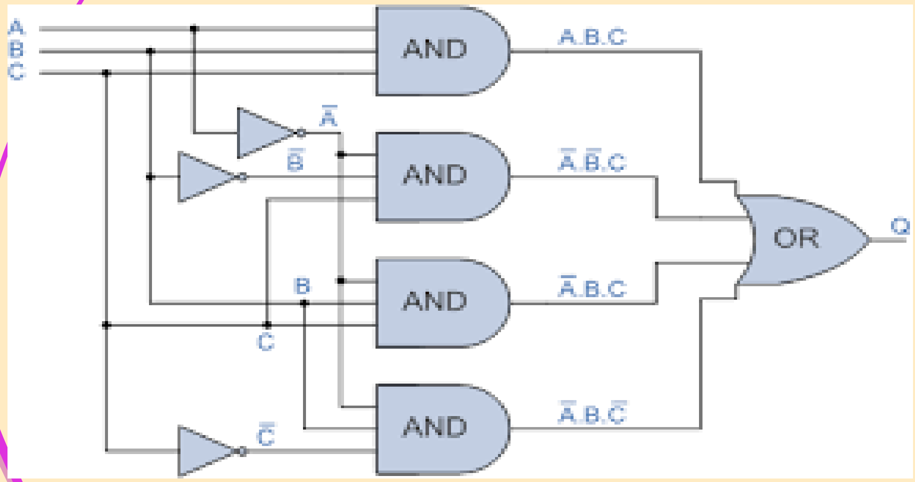
SOP can also contain a term with single variable

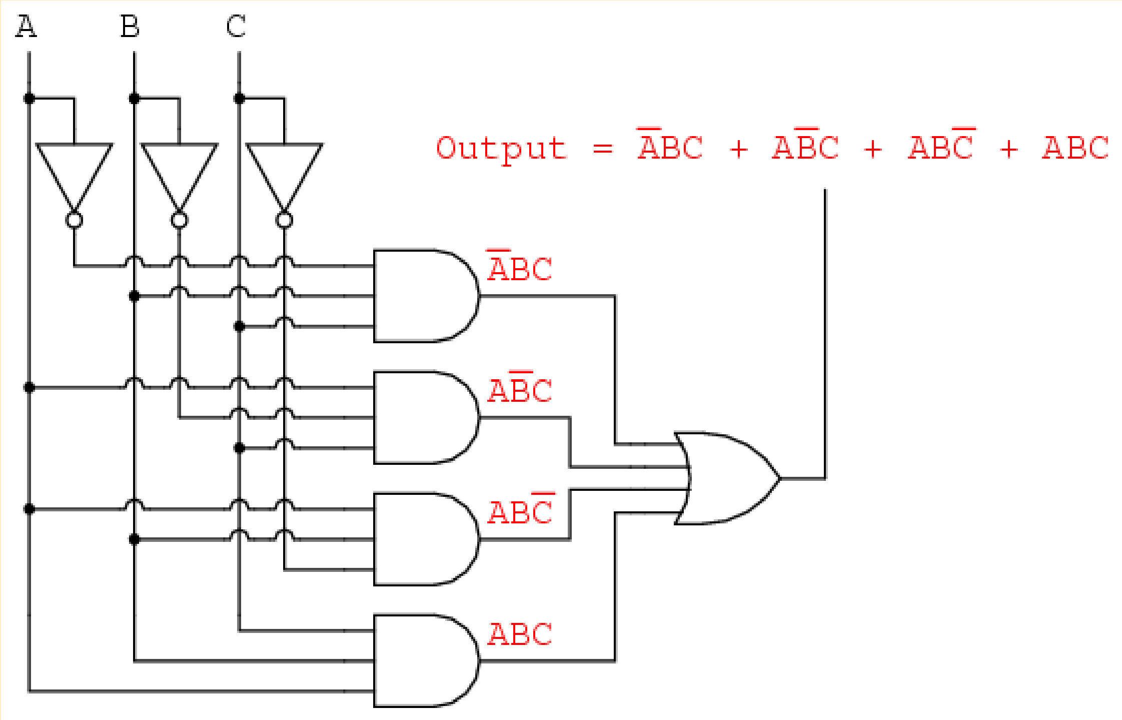
A+ BCD+ACD

SUM –OF –PRODUCT FORM(SOP):

Implement SOP expression using logic gates







PRODUCT OF SUMS FORM (POS):

POS expression is AND of two or more OR functions

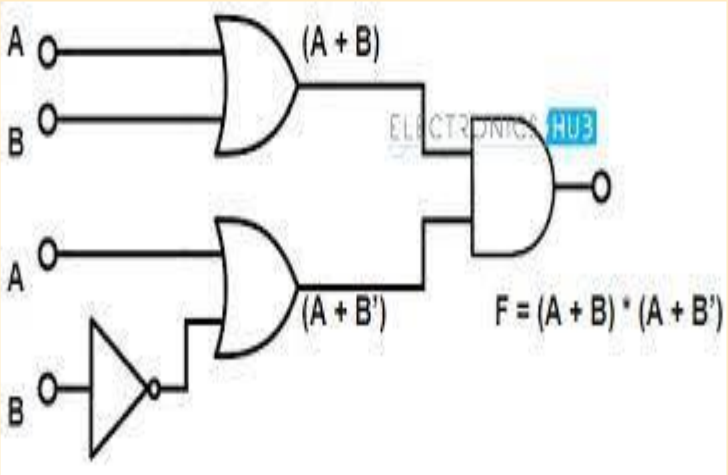
For example (A + B). (C + D) is POS expression

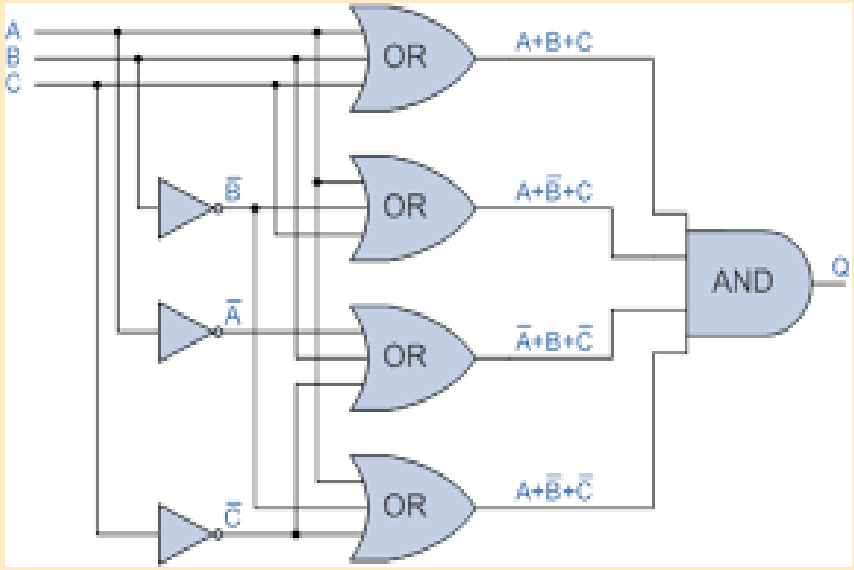
(A+B). (B + C+ D)

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

SOP can also contain a term with single variable

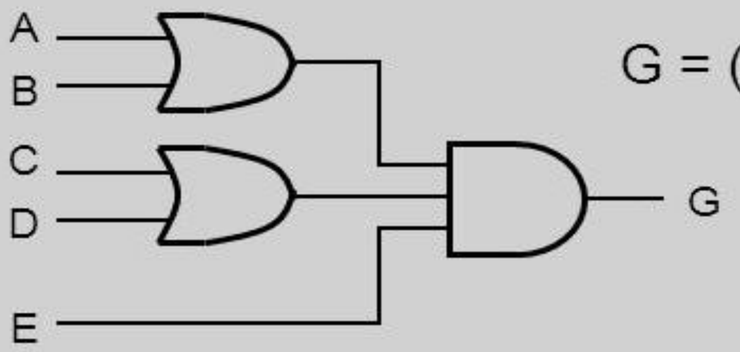
A(B+$\overline{C}$+$\overline{D}$).(A+C+D)





Implementation of POS Expressions

- Product-of-Sums expressions can be implemented using:
  - 2-level OR-AND logic circuits
  - 2-level NOR logic circuits
- OR-AND logic circuit



$G = (A+B).(C+D).E$

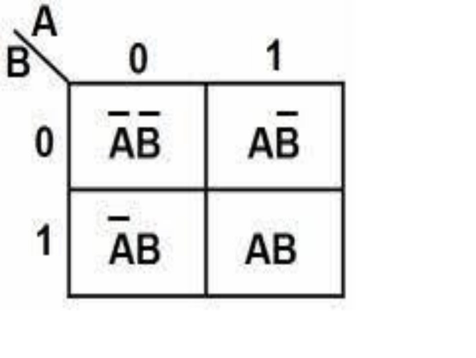
K-MAP (KARNAUGH MAPPING):

Karnaugh mapping (K-map) is another method for reduction in logical function.

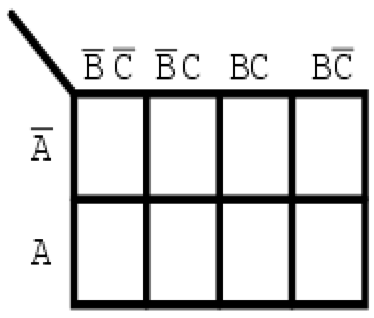
Tool to perform systematic reduction of complex logical circuit into simplified equivalent circuits.

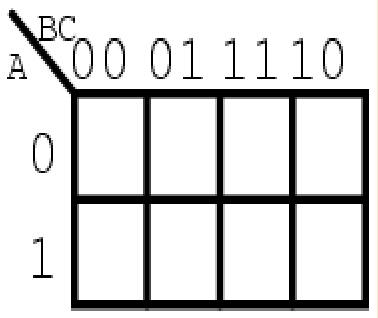
Used to simplify equations having two , three, four , five, six different input variables

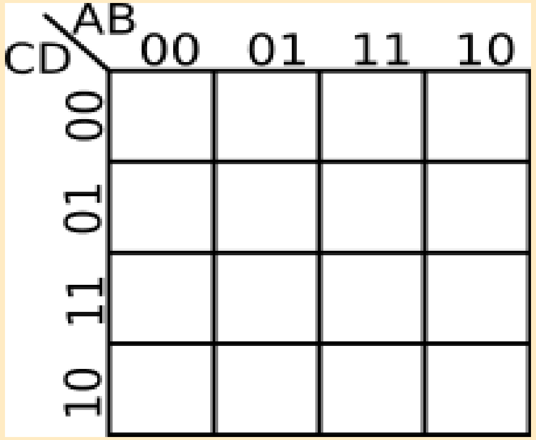
Two variable map will require 2^2 =4 cells , a three variable map require 2^3=8 cells, 4 variable 2^4=16 cells (For N variables 2^N cells required)



兩個變數之四格卡諾圖







K-MAP (KARNAUGH MAPPING):

Plotting a Boolean Expressions: SOP form of expression can be plotted on K map by placing 1 in each cell corresponding to term of SOP

For example

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

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

For three variables $\overline{A}\,B\,\overline{C} + A\,\overline{B}\,C + A\,\overline{B}\,\overline{C} + A\,B\,C$

	$\overline{B}\,\overline{C}$	$\overline{B}\,C$	BC	$B\overline{C}$
$\overline{A}$				1
A	1	1	1	

K-MAP (KARNAUGH MAPPING):

Plotting a Boolean Expressions: SOP form of expression can be plotted on K map by placing 1 in each cell corresponding to term of SOP

For example

$\bar{A}\,\bar{B} + A\bar{B}$

For Four variables $\bar{A}B\bar{C}D + \bar{A}BC\bar{D} + \bar{A}\,\bar{B}\,\bar{C}\,\bar{D} + ABCD$

AB \ CD	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
1 $\bar{A}\bar{B}$	1			
$\bar{A}B$		1		1
AB			1	
$A\bar{B}$				

K-MAP (KARNAUGH MAPPING):

Grouping cells for simplification: Group 1 s that are adjacent as per following rules

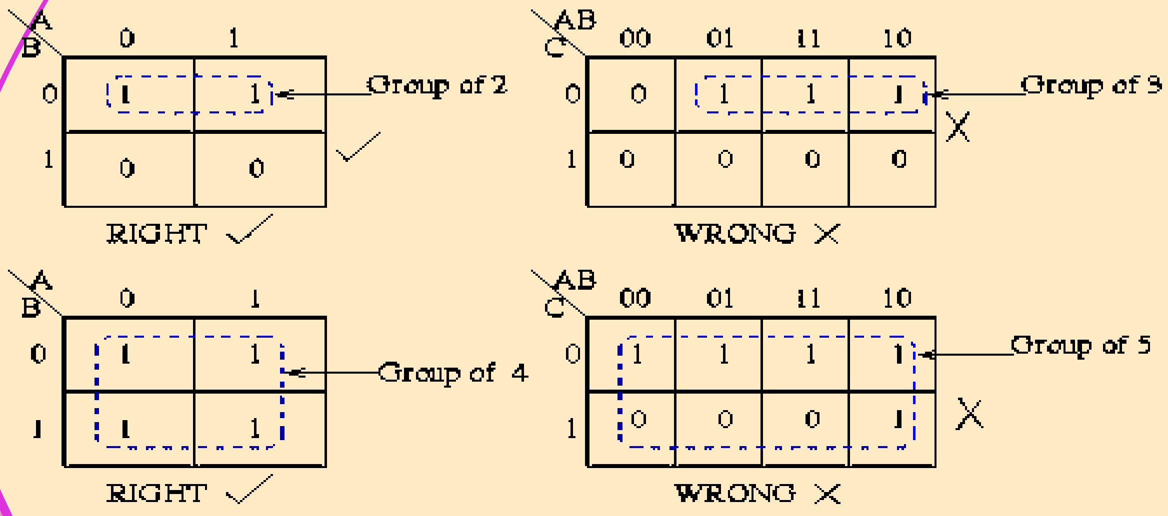
1 Adjacent cells are cells differ by only a single variable for exam(A B C D and A B C$\overline{D}$ are adjacent)

2 The 1 s in adjacent cells must be combined in groups of 1 ,2, 4, 8, 16 and so on

3 Each group of 1 s should be maximized to include largest number of adjacent cells

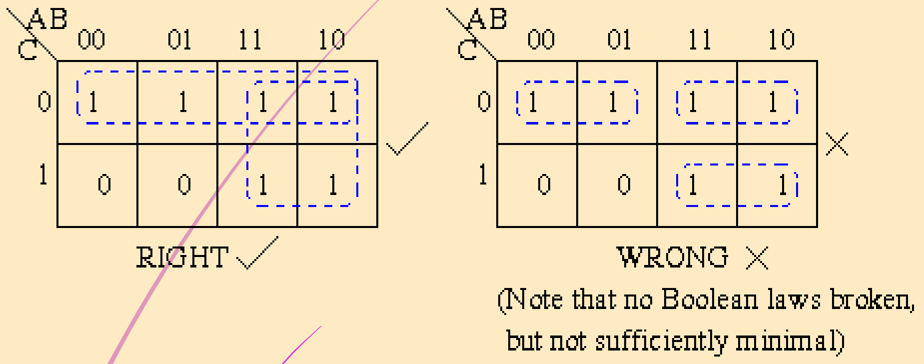
4 Every 1 on map must be included in at least one group

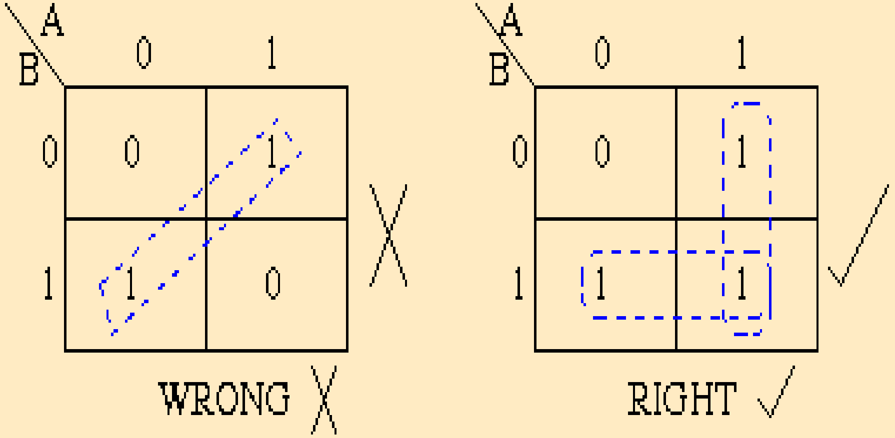
5 There can be overlapping groups if they include noncommon 1 s

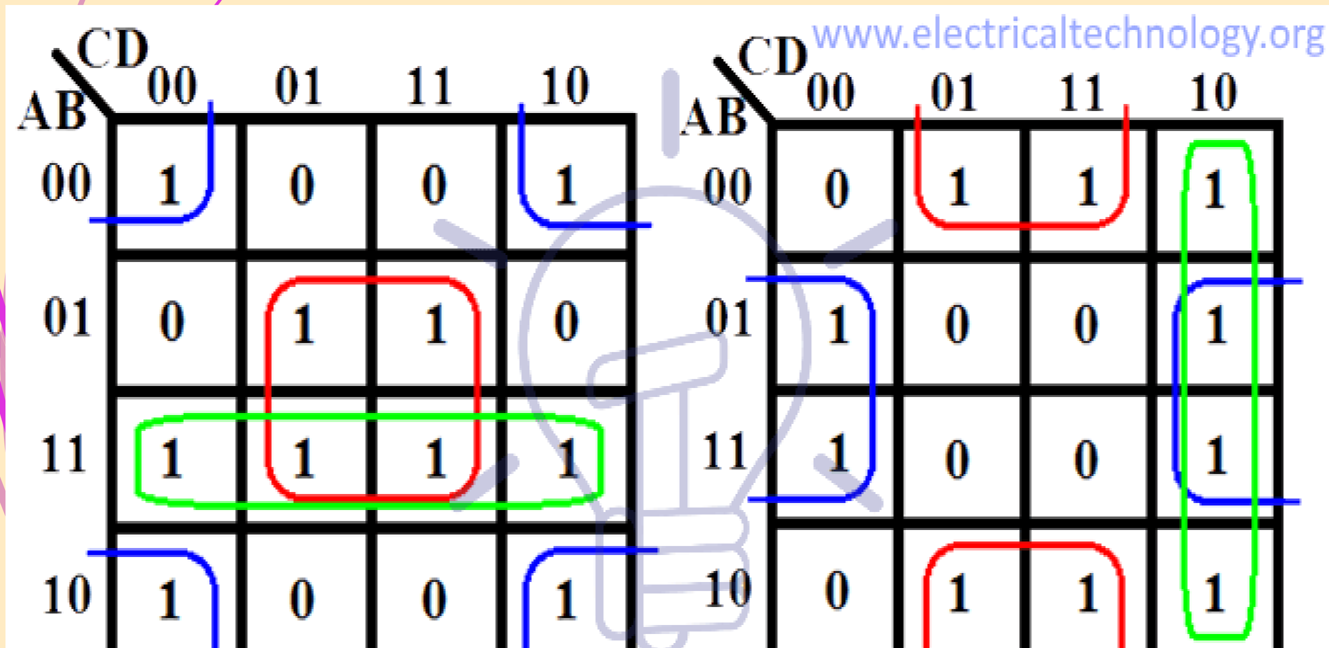


K-MAP (KARNAUGH MAPPING):

Grouping cells for simplification:

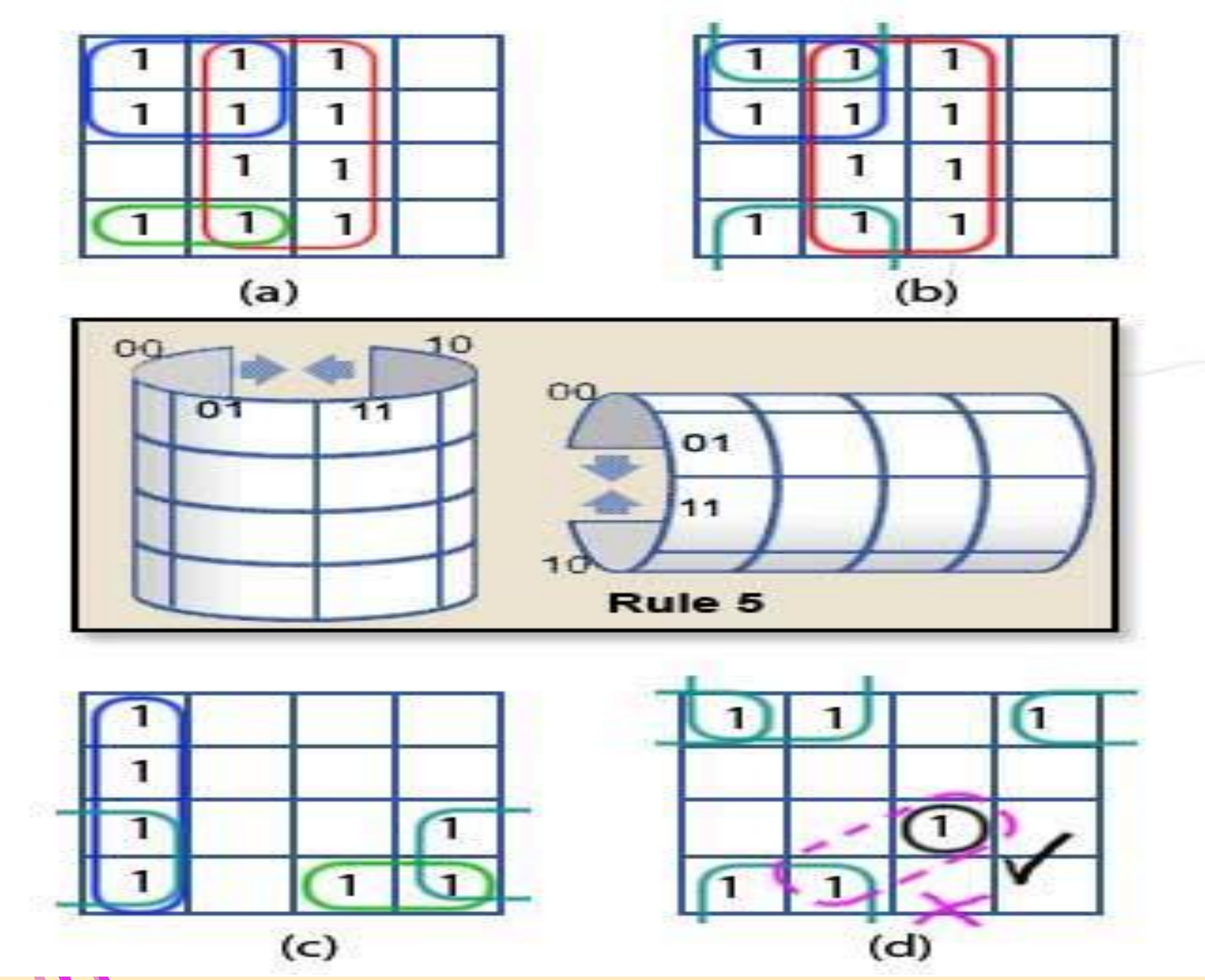


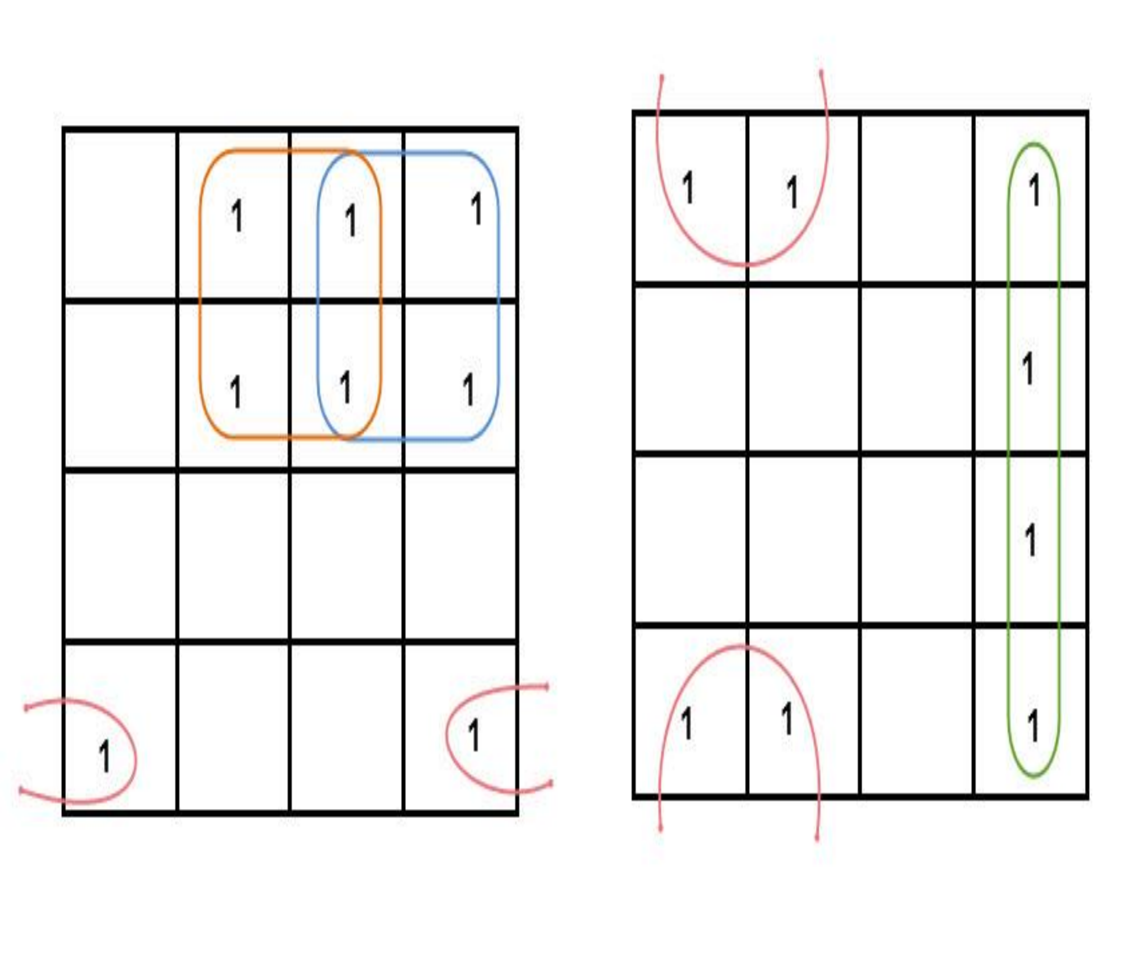




K-MAP (KARNAUGH MAPPING):

Grouping cells for simplification:





K-MAP (KARNAUGH MAPPING):

Simplifying the expression:

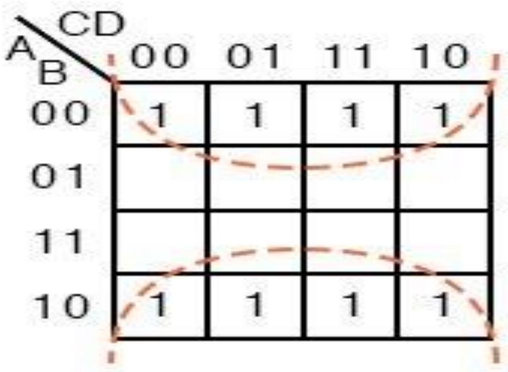
When all 1 s are grouped , the mapped expression is ready for simplification

1 Each group of 1 s creates a product term composed of all variables that appears in only one form (uncomplemented or complemented) within the group

2 Variables that appear uncomplemented and complemented are eliminated

3 The final simplified expression is formed by summing the product terms of all the groups

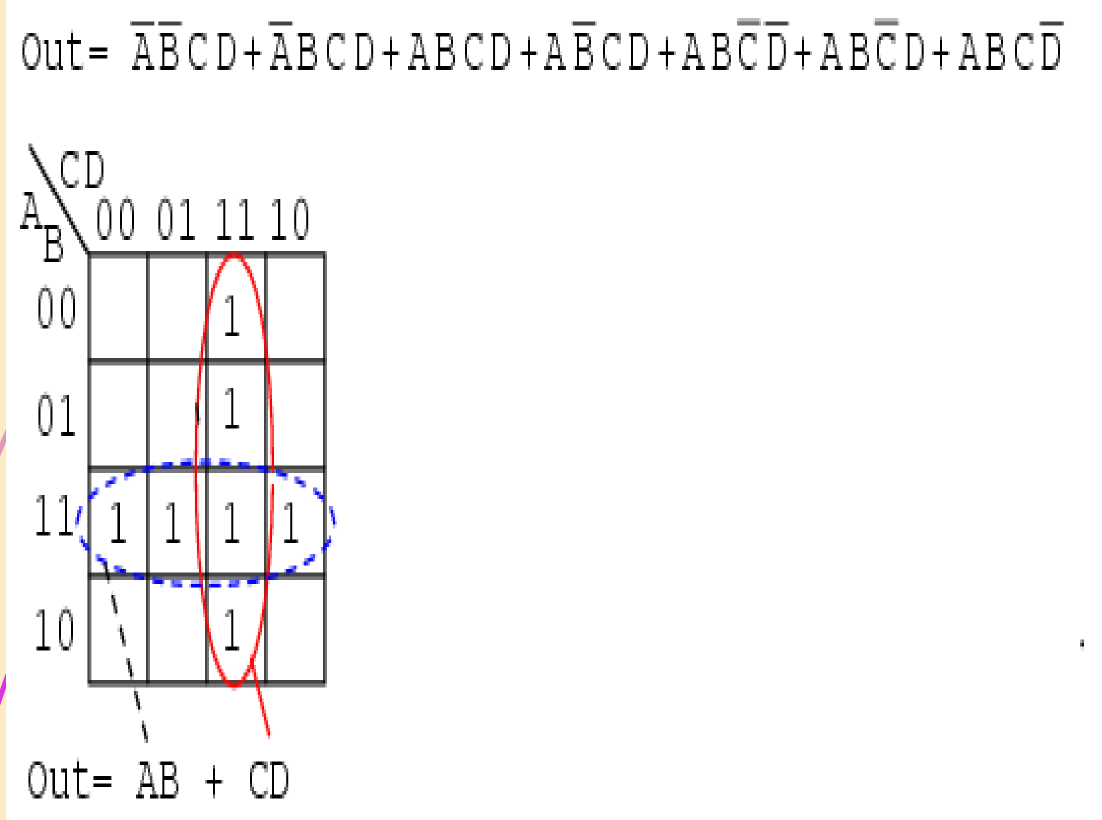
$$Out = \overline{A}\,\overline{B}\,\overline{C}\,\overline{D} + \overline{A}\,\overline{B}\,\overline{C}D + \overline{A}\,\overline{B}CD + \overline{A}\,\overline{B}C\overline{D} + A\overline{B}\,\overline{C}\,\overline{D} + A\overline{B}\,\overline{C}D + A\overline{B}CD + A\overline{B}C\overline{D}$$

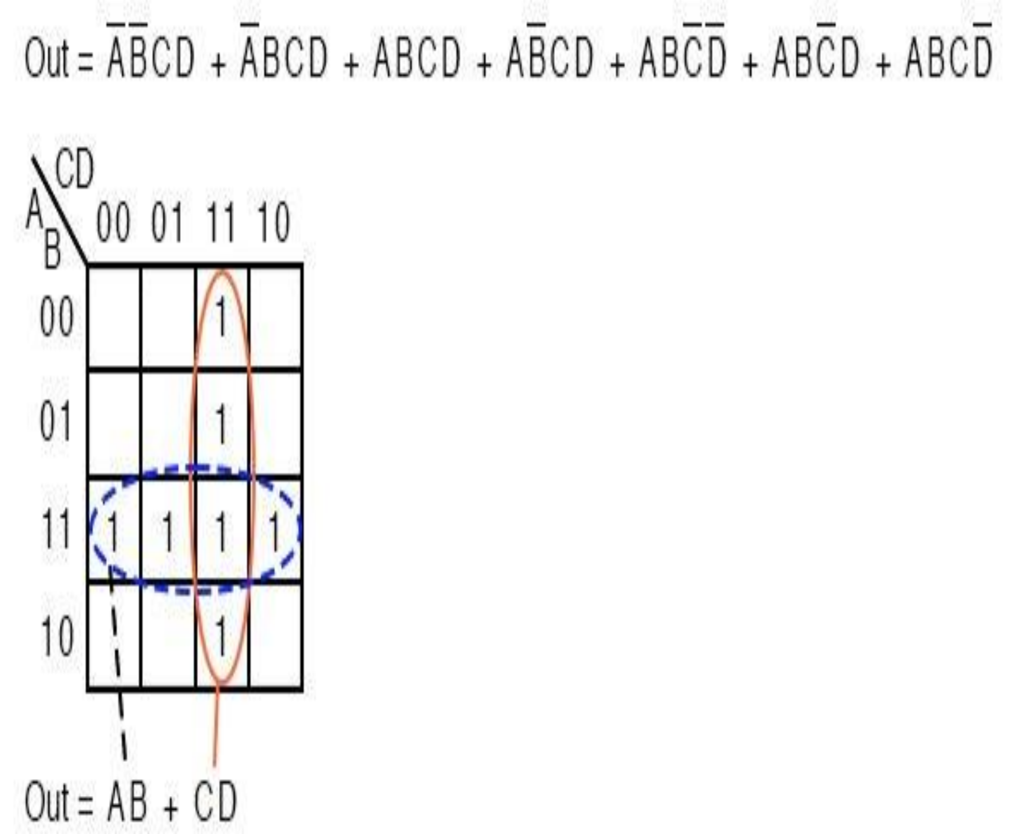


$$Out = \overline{B}$$

K-MAP (KARNAUGH MAPPING):

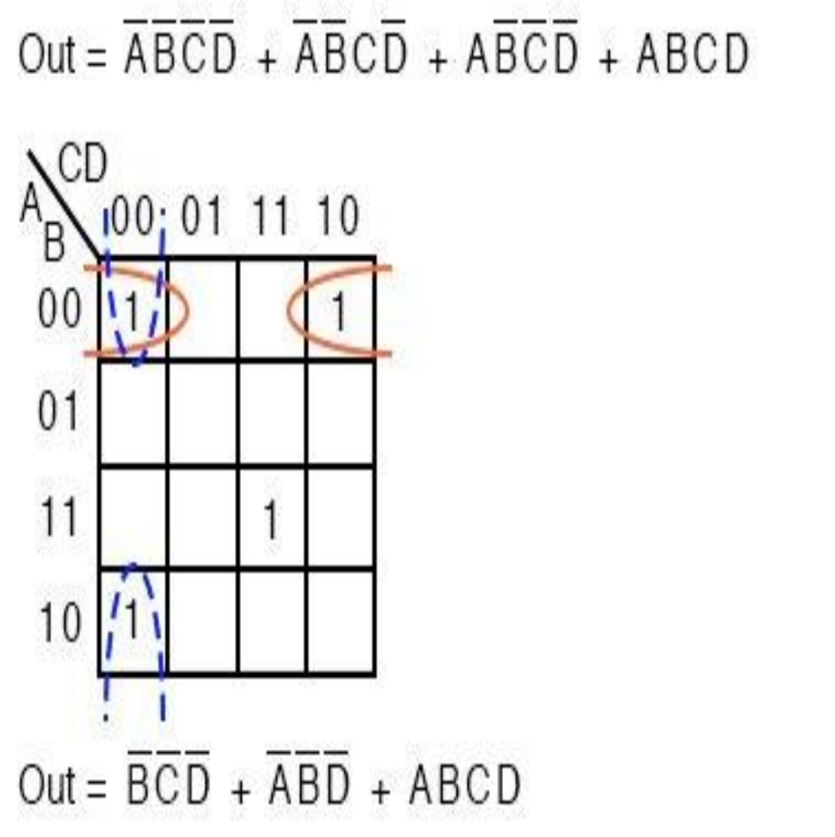
Simplifying the expression:





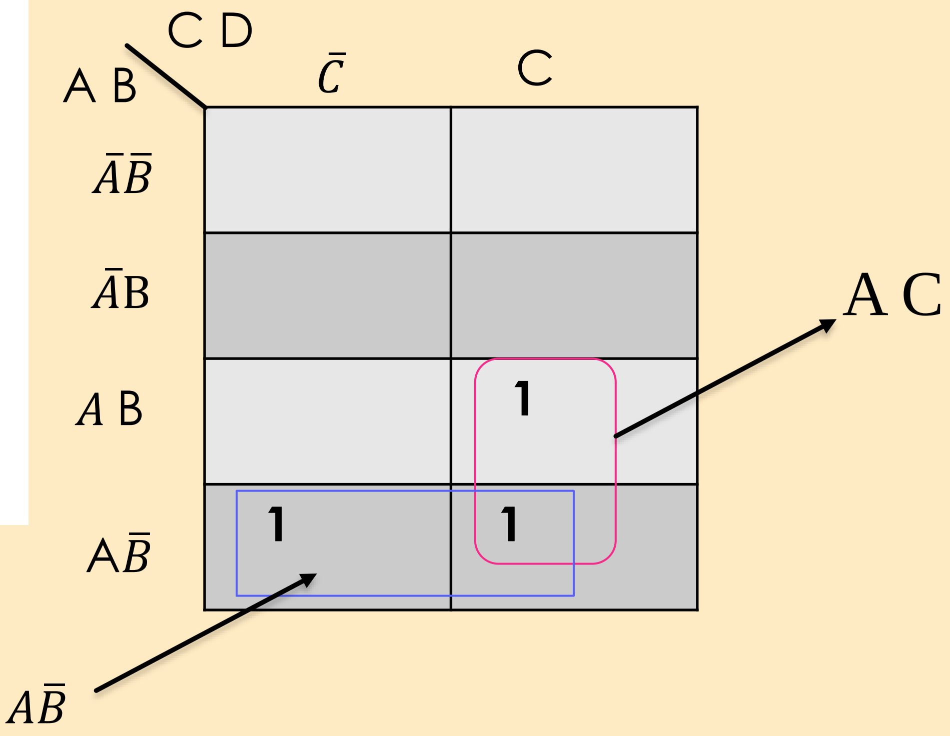
K-MAP (KARNAUGH MAPPING):

Simplifying the expression:



Using K map simplify the following expressions

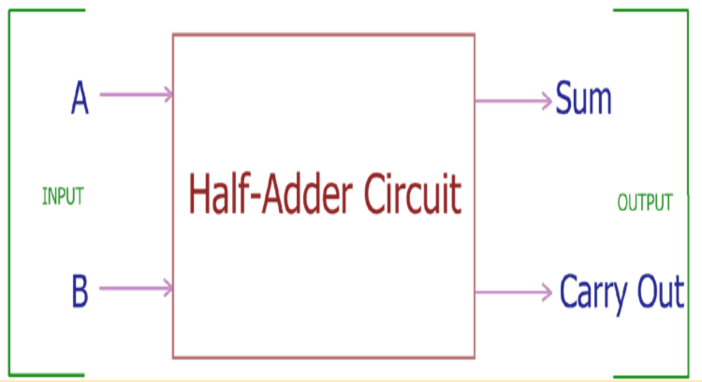
$A\,B\,C + A\bar{B}C + A\bar{B}\bar{C}$



$A\bar{B} + AC$

HALF ADDER:

- Half adder adds two bit binary and produces sum and carry
- Two input (A and B) and two outputs (S=(SUM) and C=CARRY)to half adder

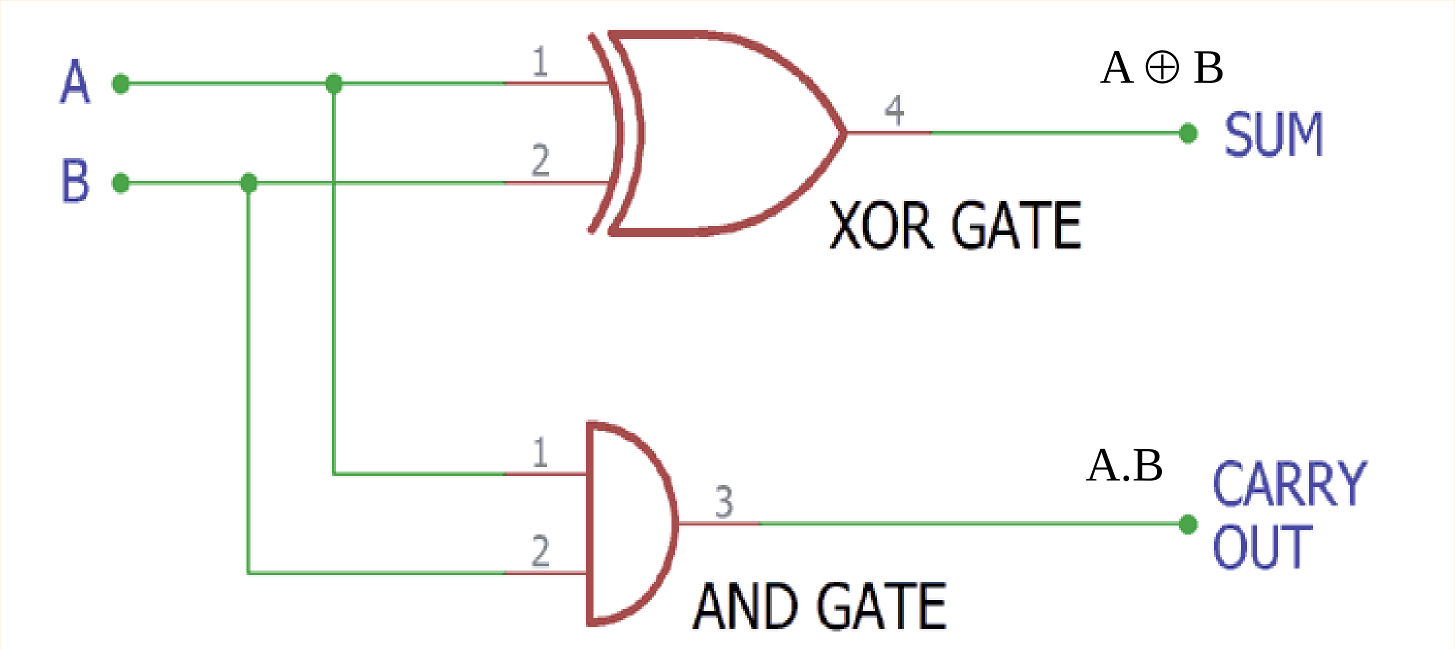


Inputs		Outputs	
A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

From truth table, Sum bit is a 1 only when input variables are not equal.
The sum can be expressed as exclusive OR of the input variable
The output C is a 1 only when both A and B inputs are 1
Carry is expressed as AND of the input variables

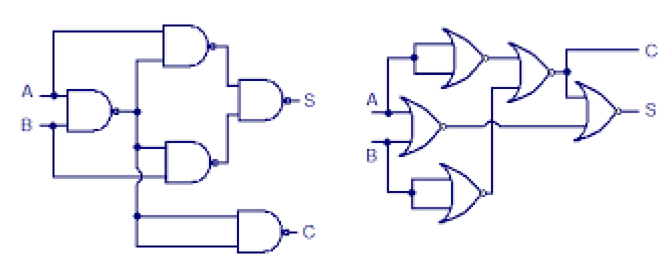
HALF ADDER:

- Logical expression:
- $S = \bar{A}\,B + A\bar{B}$ = A ⊕ B
- C= A.B



HALF ADDER:

- Half adder adds two bit binary and produces sum and carry

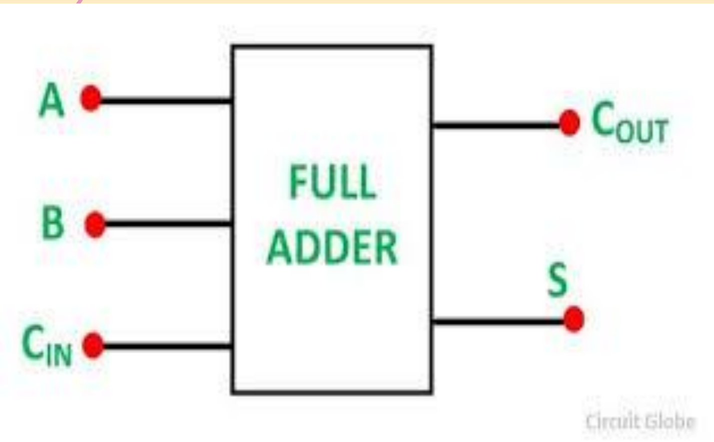


Half adder using NAND logic

Half adder using NOR logic

FULL ADDER:

- Half adder adds three bit binary and produces sum and carry
- Three input (A and B and Carry in) and two outputs (S=(SUM) and C=CARRY out)to full adder



Input			Output	
A	B	C_{in}	Sum	Carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

FULL ADDER:

Input			Output	
A	B	C_{in}	Sum	Carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$$\text{SUM} = \bar{A}\bar{B}C_{in} + \bar{A}B\bar{C}_{in} + A\bar{B}\bar{C}_{in} + ABC_{in}$$

$$= C_{in}(\bar{A}\bar{B} + AB) + \bar{C}_{in}(A\bar{B} + \bar{A}B)$$

$$= C_{in}\left[(\bar{A} + B)\cdot(A + \bar{B})\right] + \bar{C}_{in}(A\bar{B} + \bar{A}B)$$

$$= C_{in}(\overline{A\bar{B}}\cdot\overline{\bar{A}B}) + \bar{C}_{in}(A\bar{B} + \bar{A}B)$$

$$= C_{in}\overline{(A\bar{B} + \bar{A}B)} + \bar{C}_{in}(A\bar{B} + \bar{A}B)$$

$$= C_{in} \oplus (A\bar{B} + \bar{A}B)$$

$$= C_{in} \oplus (A \oplus B)$$

$$\text{Carry} = AB + AC_{in} + BC_{in}$$

$$= AB + AC_{in}(B + \bar{B}) + BC_{in}(A + \bar{A})$$

$$= AB + ABC_{in} + A\bar{B}C_{in} + ABC_{in} + \bar{A}BC_{in}$$

$$= AB(1 + C_{in} + C_{in}) + A\bar{B}C_{in} + \bar{A}BC_{in}$$

$$= AB + A\bar{B}C_{in} + \bar{A}BC_{in}$$

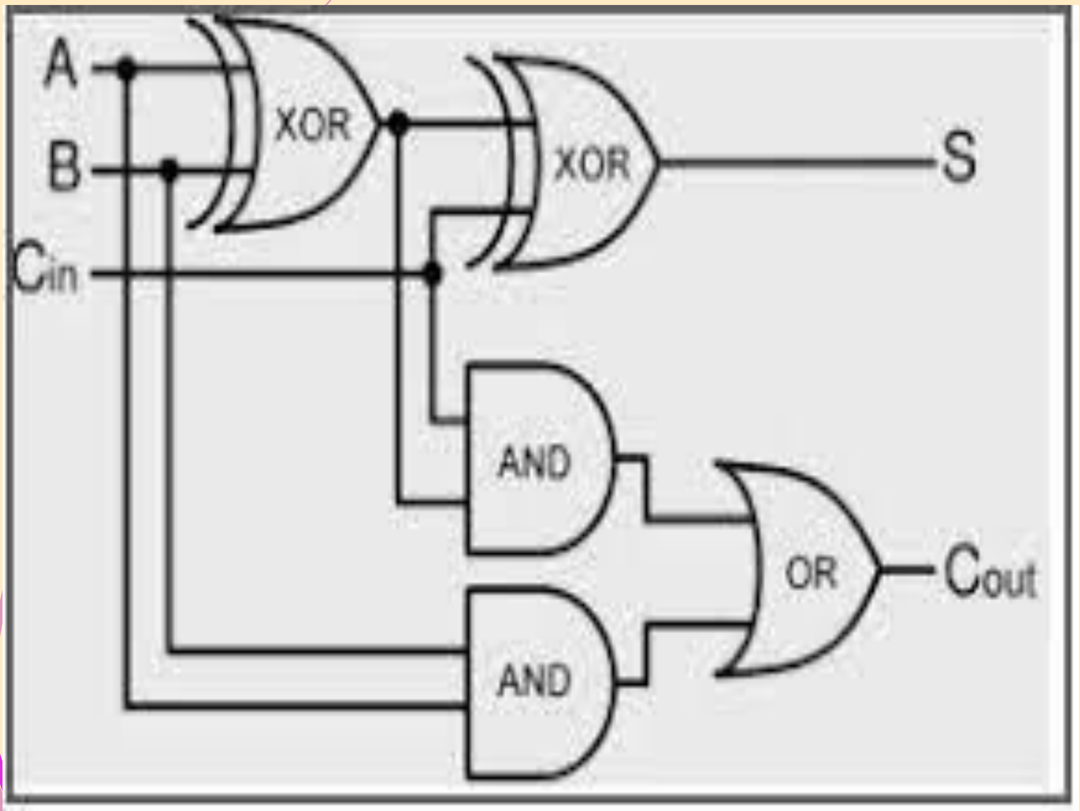
$$= AB + C_{in}(A\bar{B} + \bar{A}B)$$

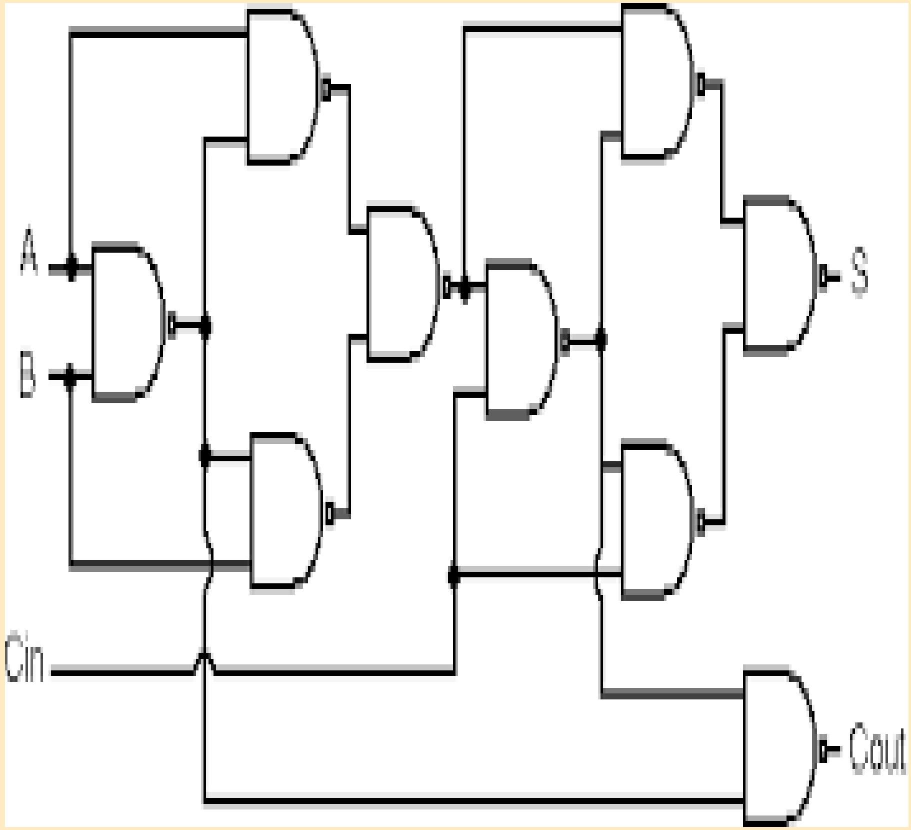
$$= AB + C_{in}(A \oplus B)$$

$A + A = A$

$1 + A = 1$

FULL ADDER:





SELF STUDY:

Simplify the expressions using K map

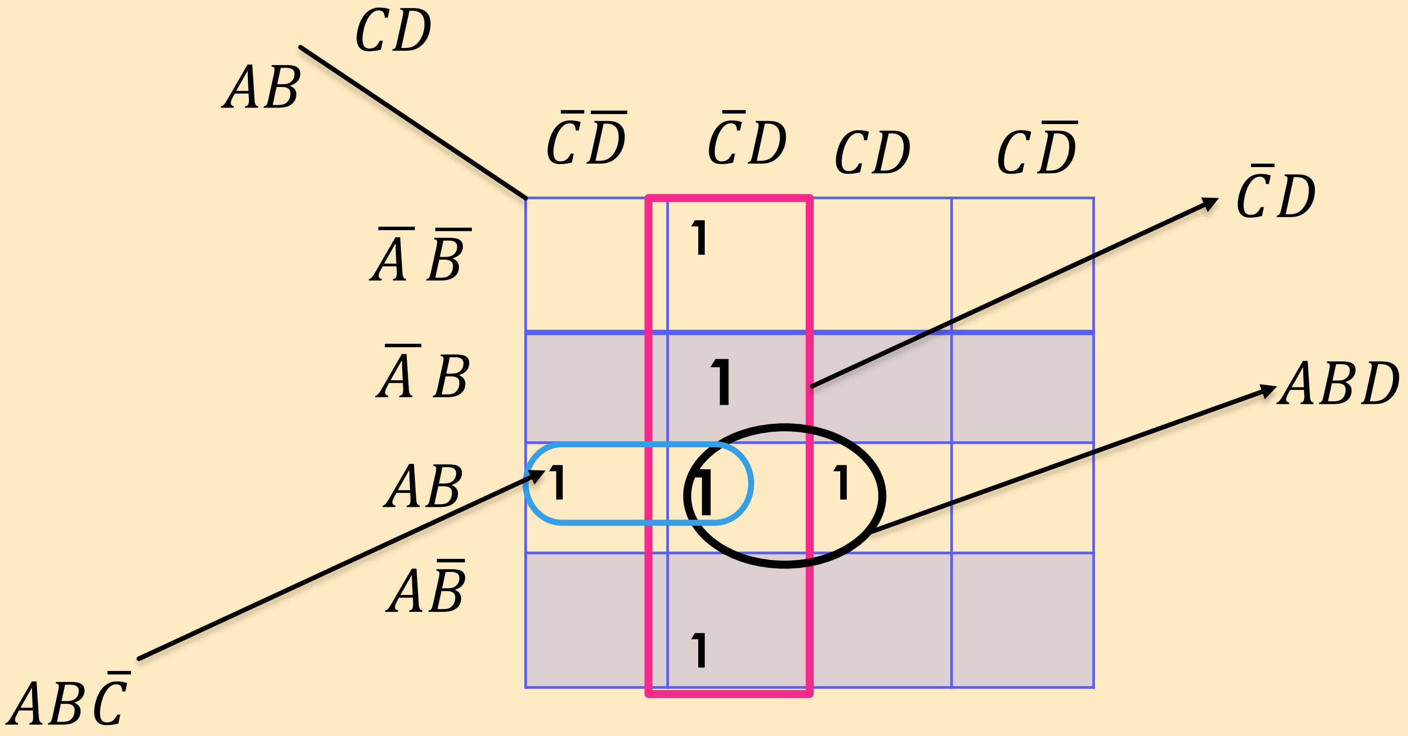
1) $X = \bar{A}B\bar{C}D + A\bar{B}\bar{C}D + \bar{A}\bar{B}\bar{C}D + AB\bar{C}D + AB\bar{C}\bar{D} + ABCD$
2) $X = B\bar{C}\bar{D} + \bar{A}B\bar{C}D + AB\bar{C}D + \bar{A}BCD + ABCD + ABCD$

SOLUTION:

Simplify the expressions using K map

1) $X = \bar{A}B\bar{C}D + A\bar{B}\bar{C}D + \bar{A}\bar{B}\bar{C}D + AB\bar{C}D + AB\bar{C}\bar{D} + ABCD$

$$\boldsymbol{X = \bar{C}D + ABD + AB\bar{C}}$$



SOLUTION:

Simplify the expressions using K map

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

$X = (A + \overline{A}).B\bar{C}\bar{D} + \bar{A}B\bar{C}D + AB\bar{C}D + \bar{A}BCD + ABCD + ABCD$

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

$\boldsymbol{X} = B\bar{C} + \boldsymbol{BD}$