UNIT - II (Boolean Algebra & Soitching functions)

→ Boolean algebra is a mathematical System that defines a Series of logical operations (AND, OR, NOT) performed on sets of variables (a, b, c..)

→ fundamental postulates of Boolean Algebra.

1. Result of Each operation is either 0 or 1 $\Rightarrow 1, 0 \in B$

2. Identity Elements

$A+0 = A \qquad A \cdot 1 = A.$

$0+0 = A \qquad 0 \cdot 1 = 0 = A$

$1+0 = A \qquad 1 \cdot 1 = 1 = A.$

3. Commolative law.

a) $(A+B) = (B+A)$

b) $(A \cdot B) = (B \cdot A)$

a) $A=0, \ B=0$

$0+0 = 0+0$

b) $A \overset{A}{=} 0 \quad B=1$

$0 \cdot 1 = 1 \cdot 0$

$$\left.\begin{aligned} &4.\ a)\ A \cdot (B+C) = (A \cdot B) + (A \cdot C) \\ &b)\ A + (B \cdot C) = (A+B) \cdot (A+C) \end{aligned}\right\} \text{Distributed law.}$$

$A=0 \quad B=1 \quad C=0.$

$0 \cdot (1+0) = (0 \cdot 1) + (0 \cdot 0)$

$0 = 0$

b) $A=0$ $B=0$ $C=0$.

$0+(0\cdot 0) = (0+0)(0+0)$

$0=0$.

5) a) $A+\overline{A}=1$ Since $0+\overline{0} = 0+1 = 1$

and $1+\overline{1} = 1+0 = 1$

b) $A\cdot\overline{A}=0$ Since $0\cdot\overline{0} = 0\cdot 1 = 0$.

and $1\cdot\overline{1} = 1\cdot 0 = 0$

Complement .

UNIT-II

Basic theorems and properties

Duality:— The principle of duality theorem Say that, Starting with a Boolean relation, you can derive another boolean relation by.

1. changing Each OR Sign to an AND Sign.
2. changing Each AND Sign to an OR Sign and.
3. complementing any 0 to 1 appearing in the Expression.

Example: Dual of relation $A+\bar{A}=1$ is $A\cdot\bar{A}=0$.

Duality is a very important property of Boolean algebra.

Basic Theorems:— the following theorems using fundamental posulates We can define the of Boolean algebra.

Theorem 1(a) $A+A=A$.

$$\boxed{\begin{matrix}0\\1\end{matrix}}\ \begin{matrix}+\\+\end{matrix}\ \boxed{\begin{matrix}0\\1\end{matrix}}\ \begin{matrix}=\\=\end{matrix}\ \boxed{\begin{matrix}0\\1\end{matrix}} \Rightarrow A+A=A$$

proof:-
$$\begin{aligned}A+A &= (A+A)\cdot 1\\ &= (A+A)(A+\bar{A})\\ &= A+A\bar{A}\\ &= A+0 = A.\end{aligned}$$

Theorem 1(b) $A\cdot A=A$

$$\boxed{\begin{matrix}0\\1\end{matrix}}\ \begin{matrix}+\\+\end{matrix}\ \boxed{\begin{matrix}0\\1\end{matrix}}\ \begin{matrix}=\\=\end{matrix}\ \boxed{\begin{matrix}0\\1\end{matrix}} \Rightarrow A+A=A$$

proof:— $A \cdot A = A \cdot A + 0$

$= AA + AA'$

$= A(A + A')$

$= A \cdot 1 = A$.

Theorem 2(a) : $A+1=1$

$$\boxed{\begin{array}{l} 1 + 0 = 1 \\ 1 + 1 = 1 \end{array}} \Rightarrow 1+A=1 \text{ or } A+1=1$$

proof: $A+1 = 1 \cdot (A+1)$

$= (A+A')(A+1) = AA + A + 0 + A' = A + A' \cdot 1$

$= A + A' \cdot 1 = A + A' = 1$

Theorem 2(b) : $A \cdot 0 = 0$

$$\boxed{\begin{array}{l} 0 \cdot 0 = 0 \\ 0 \cdot 1 = 0 \end{array}} \Rightarrow 0 \cdot A = 0 \text{ or } A \cdot 0 = 0 .$$

proof: $A \cdot 0 = 0$ by duality theorem 2(a)

Theorem (3): $\bar{\bar{A}} = A$

$$\boxed{\bar{\bar{0}}} = \boxed{0}$$

$$\bar{\bar{1}} = 1 \qquad \bar{\bar{A}} = A$$

proof: Complement of $\bar{A}$ is A and also $\bar{\bar{A}}$

Theorem 4(a): $A + AB = A$

proof: $A + AB = A \cdot 1 + AB$

$= A(1+B) = A \cdot 1 = A$.

em 4(b) : $A(A+B) = A$

proof : $A(A+B) = A \cdot A + AB$ By posulate : 4(b)

$= A + AB$ By theorem : 1(b)

$= A(1+B) = A$

Theorem 5(a) : $A + A'B = A + B$

proof : $A + AB + A'B$

$= A + B \cdot (A + A') = A + B \cdot 1 = A + B$

Theorem 5(b) : $A \cdot (\bar{A} + B) = AB$

proof : $A \cdot (\bar{A} + B)$

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

$= AA' + AB + ABB$

$= AB + ABB$

$= AB + AB$

$= AB$

Demorgan's Theorems

Demorgan's Suggested two theorems that form an important Boolean algebra. In the Equation form, they are.

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

The complement of a product is Equal to the Sum of Complements. This illustrated by below table.

Truth Table

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

2) $\overline{A+B} = \overline{A}\overline{B}$

The complement of a Sum is Equal to the product of the Complements. The below truth table illustrate this law.

Truth Table

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

tant

sensus Theorem :

In Simplication of Boolean Expression, an Expression of the form $AB+A'C+BC$ the term BC is Redundant and can be Eliminated to form the Equivalent Expression $AB+A'C$. The theorem used for this Simplification is known as consensus theorem and it is stated as

$$AB+A'C+BC = AB+A'C$$

— The key to recognize the consensus terms is to first find a pair of terms, one of which contains a variable and the other contains its complement.

— Now we have to find the third term which should contain the remaining variables from pair of terms eliminating selected variable and its complement.

proof:

$$\begin{aligned} AB+A'C+BC &= AB+A'C+(A+A')BC \\ &= AB+A'C+ABC+A'BC \\ &= AB+A'C \pm ABC+A'BC \\ &= AB(1+C)+A'C(1+B) \\ &= AB+A'C \end{aligned}$$

Example2.1: Solve the given Expression using Consensus theorem

$$A'B'+AC+BC'+B'C+AB$$

Solution:

$$A'B'+AC+BC'+B'C+AB = A'B'+AC+BC'+AB$$

$$A'B'+AC+BC'+AB = A'B'+AC+BC'$$

$$A'B'+AC+BC'+B'C+AB = A'B'+AC+BC'$$

Note: The brackets indicate how the consensus terms are identified.

Dual of Consensus Theorem

- The dual form of consensus theorem is stated as

$$(A+B)(A'+C)(B+C) = (A+B)(A'+C)$$

proof: $(AA' + AC + A'B + BC)(B+C) = AA' + AC + A'B + BC$

$$(AC + A'B + BC)(B+C) = AC + A'B + BC$$

$$ABC + A'BB + BCC + ACC + A'BC + BCC = AC + A'B + BC$$

$$(A+A')(BC + A'B + BC + AC) = AC + A'B + BC$$

$$(1) BC + A'B + BC + AC = AC + A'B + BC$$

$$BC + BC + A'B + AC = AC + A'B + BC$$

$$BC + A'B + AC = AC + A'B + BC.$$

Example 2.2: Solved the following Boolean Expression using dual of consensus theorem.

$$(A+B)(A'+C)(B+C)(A'+B)(B+D)$$

Solution: $(A+B)(A'+C)(B+C)(A'+B)(B+D)$

$$= (A+B)(A'+C)(A'+D)(B+D)$$

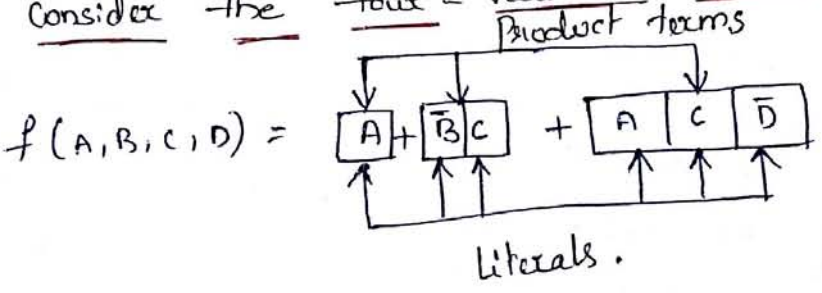
$$= (A+B)(A'+C)(A'+D)$$

witching function / Boolean function

- Boolean Expressions are Constructed by connecting Boolean Constants and Variables with the Boolean operations.
- These Boolean Expressions are also Known as Boolean formulas.
- Boolean Expressions are used to describe the functions f.
- for Example, if the Boolean Expression $(A+B')C$ is used to describe the function f, then Boolean function is written as

$$f(A,B,C) = (A+B')\,C \quad \text{or} \quad f = (A+B')C$$

- Based on the Structure of Boolean Expression, It Can be Categorized in different formulas.
- let us Consider the four - Variable Boolean function



Product terms
$f(A,B,C,D) = \boxed{A} + \boxed{\bar{B}}\boxed{C} + \boxed{A}\boxed{C}\boxed{\bar{D}}$
Literals.

- In this Boolean function the Variables are appeared either in a Complemented or an uncomplemented form.
- Each occurance of a variable in either a Complemented or an uncomplemented form is called a literal.
- The above Boolean function Consists of 6 variables or literals.
- The product term is defined as either a literal or a product of literals.
- The above function Contains three product terms, A, $\bar{B}C$ and ACD'.

Let us consider another four variable Boolean function.

Sum terms

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

literals.

- The above Boolean function consists of seven literals.
- A sum term is defined as either a literal or a sum of literals.
- The above function contains three sum terms, namely $(B+D')$, $(A+B'+C)$ and $(A'+C)$.
- These literals and terms are arranged in one of the two forms.
  - Sum of product form (SOP) and
  - product of sum form (POS)

E

Sum of product form

- The words sum and product are derived from the symbolic representations of the OR and AND functions by + and . (addition and multiplication), respectively.
- A product term is any group of literals that are ANDed together.
- for Example, ABC, xy and so on
- Sum term is any group of literals are ORed together. Such as A+B+C, x+y and so on
- A Sum of products (SOP) is a group of product terms ORed together
- for Example

$$f(A,B,C) = \overset{\text{Sum}}{ABC + AB'C'}$$

product terms

$$f(P,Q,R,S) = \overset{\text{Sum}}{P'Q + QR + RS}$$

product terms

The Sum of products Expressions Consists of two or more product terms (AND) that are ORed together.

- Each product term Consists of one ore more literals appearing in either Complemented or uncomplemented form.

- for Example, in this Expression ABC+ AB'c', the first product term Contains literals A, B and c, in their uncomplemented form.

- The Second product term Contains B and c in their complemented form.

- The Sum of product form is also Known as normal form.

<u>product of sum form</u>

A product of Sums is any groups of Sum terms ANDed together.

Some Examples

product

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

Sum terms.

product

$$f(P,Q,R,S) = (P+Q) \cdot (R+\overline{S}) \cdot (P+S)$$

Sum terms

- There product of Sums Expressions Consists of two or more Sum terms (OR) that are ANDed together.

- Each Sum term consists of one or more literals appearing in either complemented or uncomplemented form.

- The product of Sum form is also Known as conjective normal form or conjunctive normal formula.

Canonical form (Standard SOP and POS forms)

— The Canonical forms are the Special cases of SOP forms. There are also known as standard SOP and POS

Standard SOP form or minterm Canonical form

— for Example, in Expression $AB + AB\bar{C}$ the first product term do not contain literal C.

— If Each term in SOP form Contains all the literals then the SOP form is known as standard or Canonical SOP form.

— Each individual term in the standard SOP form is known as minterm.

— Therefore Canonical SOP form is also known as minterm Canonical form.

— See this Expression $ABC' + ABC + A'BC + AB'C$ all the literals are present in each product term.

— one standard Sum of products Expression is as shown in fig

$$f(A,B,C) = \boxed{A\bar{B}C} + \boxed{ABC} + \boxed{\bar{A}B\bar{C}}.$$

Each product term Consists of all literals in either Complemented form or uncomplemented form.

POS

Standard POS form or maxterm Canonical form

- If Each term in POS form Contains all the literals then the POS form is known as standard or canonical POS form.
- Each individual term in the standard POS form is called maxterm.
- ∴ Canonical POS form is also known as maxterm Canonical form.
- one standard product of sums Expression is as shown in fig.

$$f(A,B,C) = \underbrace{\boxed{A+B+C} + \boxed{A+B'+C}}$$

Each sum term consists of all literals in either complemented form or uncomplemented form.

fig: Standard POS form.

Converting Expressions in standard SOP or POS form

- Sum of products form can be converted to standard ~~form~~ sum of products by ANDing the terms in the Expression with terms formed by ORing literals and its complement which are not present in the form.
- for Example, a three literal Expression with literals A, B and C, where C is missing, then
- if there is a term AB,
- we form term $(C+\bar{C})$ it with AB.
- ∴ we get $AB(C+\bar{C}) = ABC + AB\bar{C}$

Steps to convert SOP to standard SOP form.

Step1 :- find the missing literal in Each product ter

Step2 :- AND Each product term having missing literal/s terms form by ORing the literal and its Complement form.

Step3 :- Expand terms by applying distributive law and reorder the literals in the product terms.

Step4 :- Reduce the Expression by omitting repeated product terms if any. Because A+A = A .

Example 2.9 : Convert the given Expression in standard SOP form

$$f(A,B,C) = AC + AB + BC$$

Solution : Step1 : find the missing literal/s in Each product term

$$f(A,B,C)\ \boxed{AC} + \boxed{AB} + \boxed{BC}$$

literal A is missing

literal C is missing

literal B is missing

Step2 : AND product term with (missing literal + its complement)

original product term

$$f(A,B,C) = AC \cdot (B+\bar{B}) + AB \cdot (C+\bar{C}) + BC\ (A+\bar{A})$$

missing literals and their complements

3: Expand the terms and reorder literals

Expand: $f(A,B,C) = ACB + ACB' + ABC + ABC' + BCA + BCA'$

Reorder: $f(A,B,C) = ABC + AB'C + ABC + ABC' + ABC + A'BC$

Note: After having sufficient practice student should expand product term and reorder literals in it in a single step.

Step 4: omit repeated product terms

$$f(A,B,C) = ABC + AB'C + ABC + ABC' + ABC + A'BC$$

$$= ABC + AB'C + ABC' + A'BC$$

Example 2.10: Convert the given Expression in standard SOP form

Solution: Step 1: find the missing literal/s in Each product term

$$f(A,B,C) = \boxed{A} + ABC$$

literals B and C are missing.

Step 2: AND product term with (missing literal + its complement)

$$f(A,B,C) = A \cdot (B+B') \cdot (C+C') + ABC$$

$$= (AB + AB') \cdot (C+C') + ABC$$

$$= ABC + AB'C' + AB'C + AB'C'$$

$$= ABC + ABC' + AB'C + AB'C'$$

Steps to Convert POS to standard POS form

omit

× (A,B,C)

Step1: find the missing literals in Each Sum term if any

Step2: OR Each Sum term having missing literal/s with form by ANDing the literal and its Complement.

Step3: Expand the terms by applying distributive law and reorder the literals in the Sum terms.

Step4: Reduce the Expression by omitting repeated sum terms if any. Because A+A=A.

Example: Convert the given Expression in standard POS form.

$$f(A,B,C) = (A+B)(B+C)(A+C)$$

Solution: Step1: find the missing literal/s in Each Sum term

$$f(A,B,C) = \boxed{A+B}\ \boxed{B+C}\ \boxed{A+C}$$

literal B is missing

literal A is missing

literal C is missing!

Step2: OR Sum term with (missing literal + its complement.

·original products

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

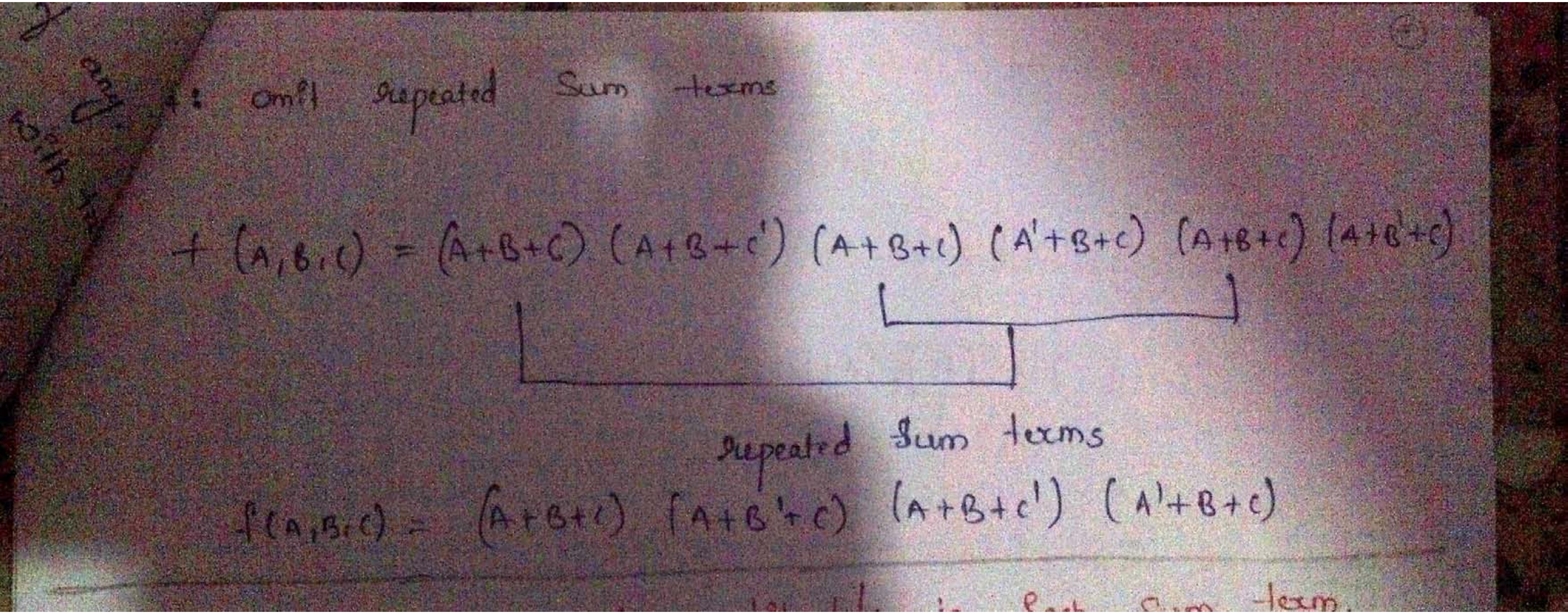
missing literals and their Complements

Step3: Expand the terms and reorder literals

Expand: Since $A+BC = (A+B)(A+C)$ we have

$$f(A,B,C) = (A+B+C)(A+B+C')(B+C+A)(B+C+A')(A+C+B)(A+C+B')$$

reorder: $f(A,B,C) = (A+B+C)(A+B+C')(A+B+C)(A'+B+C)(A+B+C)(A+B'+C)$



4: omit repeated sum terms

$$f(A,B,C) = \underbrace{(A+B+C)(A+B+C')\underbrace{(A+B+C)(A'+B+C)(A+B+C)}}_{\text{repeated sum terms}}(A+B'+C)$$

$$f(A,B,C) = (A+B+C)(A+B'+C)(A+B+C')(A'+B+C)$$

in each sum term.

Example : Convert the given Expression in standard Pos fo

$$Y = A \cdot (A+B+C)$$

Solution: Step 1: find the missing literal/s in Each Sum term

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

literals B and C are missing.

Step 2: OR Sum term with (missing literal . its complement.

$$f(A,B,C) = (A + B \cdot B' + C \cdot C')\ (A+B+C)$$

Step 3: Expand the terms and reorder literals

Since $A + BC = (A+B)(A+C)$ we have,

$$f(A,B,C) = (A + B \cdot B' + C \cdot C')\ (A + B \cdot B' + C')\ (A+B+C)$$

$$= (A+B+C)\ (A+B'+C)\ (A+B+C')\ (A+B'+C')\ (A+B+C)$$

Step 4: Omit repeated Sum terms

Repeated Sum term.

$$f(A,B,C) = (A+B+C)\ (A+B'+C')\ (A+B+C')\ (A+B'+C')\ (A+B+C)$$

$$f(A,B,C) = (A+B+C)\ (A+B'+C)\ (A+B+C')\ (A+B'+C')$$

M-Notations : Minterms and maxterms

- Each individual term in standard SOP form is called minterm and Each individual term in standard POS form is called maxterm.
- The concept of minterms and maxterms allow us to introduce a very convenient shorthand notations to Express logical functions.
- Below table gives the minterms and max-terms for a three literal/ Variable logical function
- where the number of minterms as well as maxterms is $2^3 = 8$.
- for an n-variable logical function there are 2^n minterms and an Equal number.

9

mple 12: $A+A'B+AB' = A+B$

L.H.S $= A+A'B+AB'$

$= A(1+B') + A'B$

$= A(1+B') + A'B \quad = A+AB'+A'B$

$= (A+B) + AB'$

$= A+B+AB' \quad = A+(B+B'A) = A+(A+B)$

$= \cancel{A(1+B')+B} \quad A+A+B$

$= A+B$

Example 13: $\overline{\overline{AB}+\overline{A}+AB}$

$= \overline{\overline{A}+\overline{B}+\overline{A}+AB}$ Demorgan's theorem 1: $[\overline{AB} = \overline{A}+\overline{B}]$

$= \overline{\overline{A}+\overline{B}+AB} \quad = \overline{A'+B+B'}$ $\boxed{A+A'=1}$

$= \overline{A'+1}$ $\boxed{A+1} = 1$

$= \overline{1}\cdot 0$

$= 0$

$\overline{A'}\cdot 1'$

$A\cdot 1$

$A\cdot 0 = 0$

Example 14: $AB+\overline{AC}+AB'C\ (AB+C)$

$= AB+\cancel{\overline{AC}} + AA\overline{B}BC + A\overline{B}CC$

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

$= AB+\overline{AC}+AB'C$

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

$= \cancel{A(B+C)+\overline{AC}} \quad = \overline{A}+B+\overline{C}+AB'C$

$= A'+AB'C+B+C'$

$= A'+B'C+B+C'$

$= A'+B+C'+B'C$

$= A'+B+C'+B'$

$= A'+C'+1 \quad = 1$

Example : Simplify the Expression $Z = AB + AB' + (\overline{\bar{A}\,\bar{C}})$

Solution:-

Step1:- Apply the Demorgan's theorem and multiply out all to get Expression in Sum of products form.

$$Z = AB + AB' + (\overline{\bar{A} \cdot \bar{C}})$$
$$= AB + AB' + \bar{\bar{A}} + \bar{\bar{C}}$$
$$= AB + AB' + (A + C)$$
$$= (AB + AB')(A + C)$$
$$= AAB + ABC + AAB' + AB'C$$
$$= AB + ABC + AB' + AB'C$$
$$= AB(1+C) + AB'(1+C)$$
$$= AB + AB' = A(1+B') = A.$$

Example : Simply the following three Variable Expression using Boolean algebra. $Y = \Sigma m(1,3,5,7)$

Solution: Step1: from the minterms we can write Expression in Sum of products form as follows

$$Y = A'B'C + A'BC + AB'C + ABC$$

Step2: Search for common terms for factorization and apply boolean rules.

$$Y = A'B'C + A'BC + AB'C + ABC$$
$$= A'C(B+B') + AB'C + ABC$$
$$= A'C + AC(B+B')$$
$$= A'C + AC$$
$$= C(A+A') = C.$$

: Convert the Expression given in the previous Example into (11) minterms using Complementary property and Simply the Expression.

Solution:

$Y = \pi M(3,5,7)$

$Y = \Sigma m(0,1,2,4,6)$

Step1: from the minterms write Expression in SOP form

$$Y = A'B'C' + A'B'C + A'BC' + AB'C' + ABC'$$

$$= A'B'C' + AB'C' + A'BC' + ABC' + A'B'C \text{ rearranging terms}$$

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

$$= B'C' + BC' + A'B'C$$

$$= C'(B+B') + A'B'C$$

$$= C' + A'\underline{B'C}$$

$$= \bar{C} + \bar{A}\bar{B} \qquad \boxed{A + A'B = A+B}$$

Digital Logic Gates

Logic operators: To represent and Solve arithmetic Expressions such as +, -, X and ÷, we use arithmetic operators

- Similarly, we can use logical operators to represent and Solve logical Expressions.
- There are basic logical operators: NOT /INVERT, AND and OR.
- Logical operator NOT /INVERT

Variables			minterms	maxterms
A	B	C	m_i	M_i
0	0	0	$\bar{A}\bar{B}\bar{C} = m_0$	$A+B+C = m_0$
0	0	1	$\bar{A}\bar{B}C = m_1$	$A+B+C' = m_1$
0	1	0	$\bar{A}B\bar{C} = m_2$	$A+B'+C = m_2$
0	1	1	$\bar{A}BC = m_3$	$A+B'+C' = m_3$
1	0	0	$AB'C' = m_4$	$A'+B+C = M_4$
1	0	1	$AB'C = m_5$	$A'+B+C' = M_5$
1	1	0	$ABC' = m_6$	$A'+B'+C = M_6$
1	1	1	$ABC = m_7$	$A'+B'+C' = M_7$

Table: minterms and maxterms ~~for~~ three Variables

— As shown in table (2.5) Each minterm is represented by m_i and Each maxterm is represented by m_i, where subscript i is the decimal number Equivalent of the natural binary number.

— With these shorthand notations logical function can be represented as follows

1. $f(A,B,C) = A'B'C' + A'B'C + A'BC + ABC'$
   $= m_0 + m_1 + m_3 + m_6 = \Sigma m\,(0,1,3,6)$

2. $f(A,B,C) = (A+B+C')\,(A+B'+C')\,(A'+B'+C)$
   $= m_1 + M_3 + m_6 = \pi M\,(1,3,6)$

Where Σ denotes sum of product while π denotes product of sum.

gebraic Simplification.

Example 1 $A \cdot A'C = 0 \cdot C$

$= 0$

Example 2. $ABCD + ABD$

$ABD(1+C)$

$ABD(1) = ABD.$

Example 3: $ABCD + AB'CD = (ACD)(B+B')$

$= ACD \cdot 1 = ACD.$

Example 4: $A(A+B) = AA + AB$

$= A + AB$

$= A(1+B) = A.$

Example 5: $AB + ABC + AB(D+E)$

$= AB + ABC + ABD + ABE$

$= AB(1+C+D+E)$ $\therefore A+1=1$

$= AB.$

Example 6: $xy + xyz + xyz' + x'yz$

$= xy(1+z) + z'xy + x'yz$

$= xy + xyz' + x'yz$

$= xy(1+z') + x'yz$

$= xy + x'yz$

$= y(x + x'z)$ $\boxed{(A+A'B) = (A+B)}$

$(A+A')(A+B)$

$= y(x+z)$ $1 \cdot (A+B) = A+B$

Example 7: $A'B'C' + A'BC' + A'BC$

$= A'\cdot C'\cdot (B'+B) + A'BC$

$= A'C' + A'BC$

$= A'(C'+BC)$ ∵ $A+A'=1$

∴ $A+A'B = A+B$

$= A'(C'+B)$

Example 8: $ABC + AB'C + ABC' = A(C+B)$

L.H.S $= AC(B+B') + ABC'$

$= AC + ABC'$

$= A(C+BC')$

$= A(C+B)$ ∴ $A+A'B = A+B$.

Example 9: $A'BCD' + BCD' + BC'D' + BC'D$

$= BCD'(A'+1) + BC'D' + BC'D$ $\boxed{(A+1)=1}$

$= BCD' + BC'D' + BC'D$

$= BD'(C+C') + BC'D$

$= BD' + BC'D$

$= B(D'+C'D)$ $A+A'B=(A+B)$

$= B(D'+C')$

Example 10: $AC + C(A+A'B)$

$= AC + AC + A'BC$

$= AC + A'BC$

$= C(A+A'B) = C(A+B)$

Example 11: $A'BC'D + A'BCD + ABD$

$= A'BD(C'+C) + ABD$

$= A'BD + ABD$

$= BD(A+A') = BD$.

c Gates

Logic gates are the basic Elements that makeup a digital System.

The Electronic gate is a circuit that is able to operate on a number of binary i/p's in order to perform a particular logical function.

The types of logical gates are.

NOT, AND, OR, NAND, NOR, Exclusive - OR, and Exclusive-NOR

The gate is a digital circuit with one or more input voltages but only one o/p voltage.

By connecting the different gates in different ways, we can build circuits that perform arithmetic and other functions associated with the human brain because they simulate mental processes.

The operation of a logic gate can be easily understood with the help of truth table.

The truth table is a table that shows all the input-output possibilities of a logic circuit; i.e the truth table indicates the o/p's for different possibilities of the inputs.

INVERTER - NOT Gate

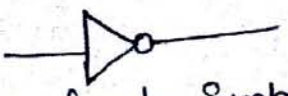


fig: Inverter Symbol

The inverter (NOT circuit) performs a basic logic function called 'inversion' or 'Complementation'.

- The inverter changes one logic level to its opposite level.
- In terms of bits, it changes a logic 1 to a logic 0 and a logic 0 to a logic 1. above fig shows the Symbol for the inverter.

Inverter operation :

When a high level is applied to an inverter input, a low appear on its output.
When a low level is applied to an inverter input, a high will appear on its output.

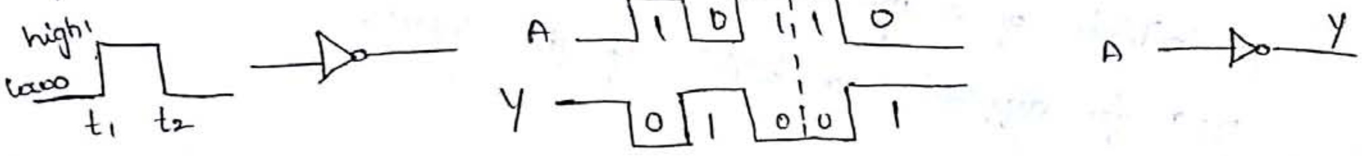
Inverter Truth Table

Input	output	Input	output
low	high	0	1
High	low	1	0

The below fig shows the o/p of an inverter for a pulse input. Here t_1 and t_2 indicate the corresponding points on the i/p and o/p pulse waveforms.

Note: When the input is low, the o/p is high, and when the i/p is high, the o/p is low.

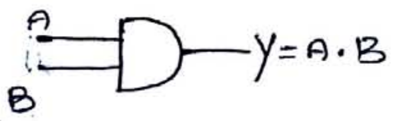
The inverter produces an inverted o/p pulse for a given input pulse



o/p of an inverter for pulse i/p.

& Gate

The AND gate performs logical multiplication, The AND gate have two or more inputs and a Single Output, as indicated by the standard logic Symbol. Shown in the below fig.



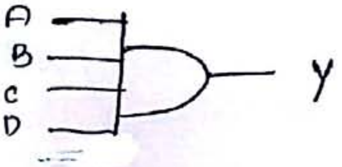


fig ⓐ Two inputs to AND gate fig ⓑ four inputs to AND gate.

The operation of the AND gate is such that the o/p is high only when all of the inputs are high.

When any of the inputs are low, the output is low.

Below fig illustrates a two-input AND gate with all four possibilities of input Combinations, and resulting output for each.

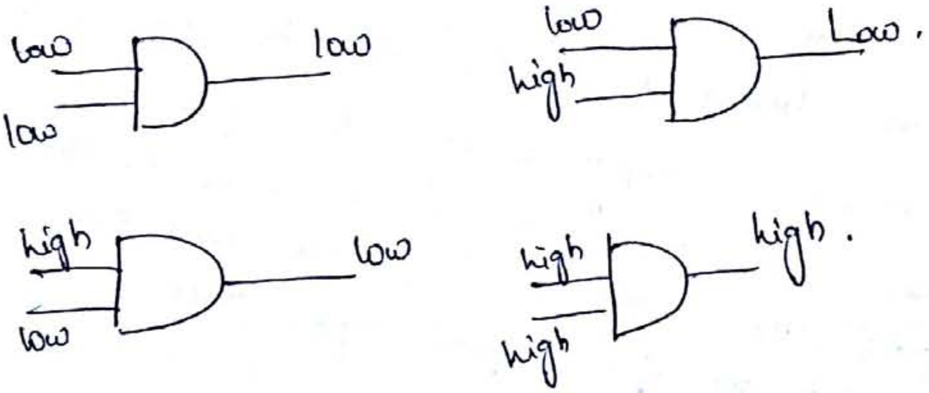


fig: four possible inputs for two input AND gate and resulting outputs.

The truth table for a two-input AND gate is sha

table.

This table can be Expanded for any number of inputs.

Inputs		output
A	B	y
0	0	0
0	1	0
1	0	0
1	1	1

Table: Truth table for 2 input AND gate.

- Below fig shows one way to build a 2-input AND gate,
- The inputs are labeled A and B, while the o/p is y.
- Let us assume a Supply voltage Vcc of +5v.
- Next assume the input voltages are either 0v (low) or 5v (high). with 2 inputs, there are four possible input cases and we will now observe the o/p for all four input cases.

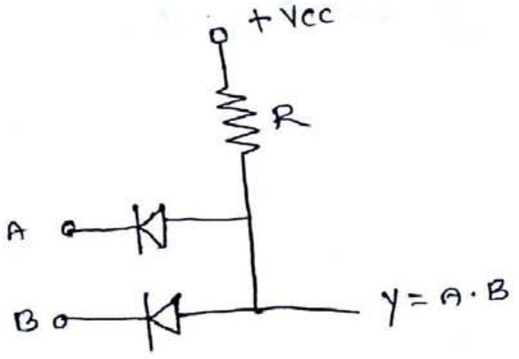


fig: 2-input AND gate.

1: A is low and B is low: When both input voltages are low, the cathode of each diode is grounded. Therefore, the positive supply forward-biases both diodes in parallel. Because of this, the o/p voltage is ideally zero (practically 0.7V for Si). This means y is low.

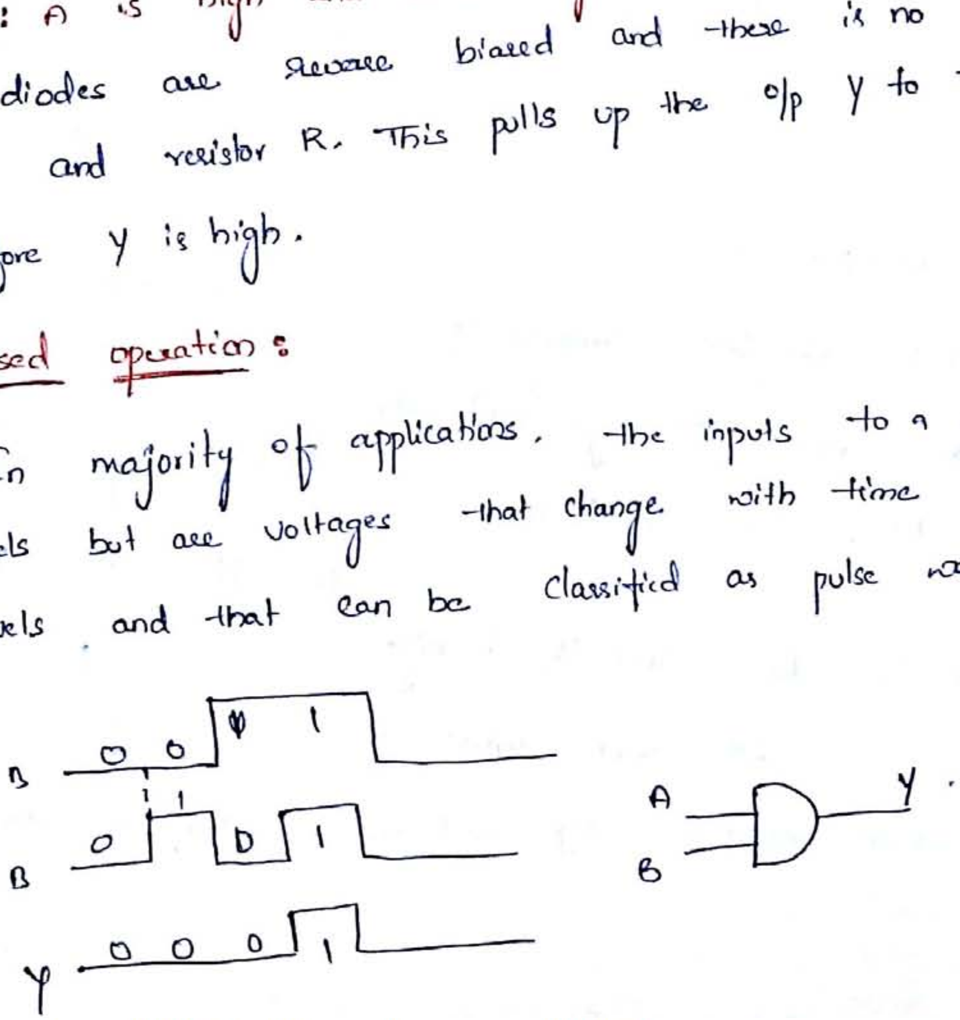
Case 2: A is low and B is high: When A is low, the upper diode is forward-biased (ON), and it pulls the o/p down to a low voltage, i.e y=0. with the B input high, the lower diode goes into reverse bias (OFF).

Case 3: A is high and B is low: Because of the symmetry of the circuit, the circuit operation is similar to case 2. But in this case, upper diode is reverse biased (OFF), lower diode is forward biased (ON) and Y is low.

Case 4: A is high and B is high: When both inputs are at +5V, both diodes are reverse biased and there is no current through diodes and resistor R. This pulls up the o/p Y to the supply voltage. Therefore Y is high.

pulsed operation:

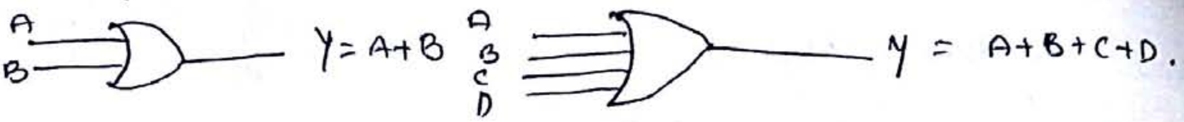
In majority of applications, the inputs to a gate are not constant levels but are voltages that change with time between two logic levels and that can be classified as pulse waveforms.



The OR Gate:

The OR gate performs logical addition.

Symbol

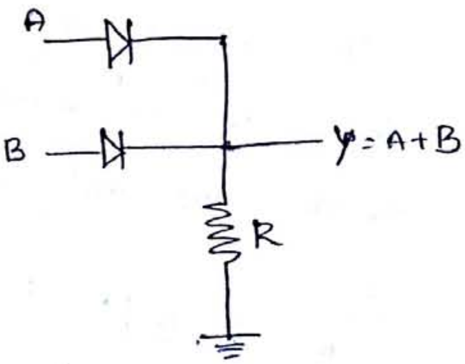


Truth table

A	B	output
0	0	0
0	1	1
1	0	1
1	1	1

Two i/p OR- Gate:

a) When both are low, anodes of both the diodes are grounded with results in RB. So o/p is low.

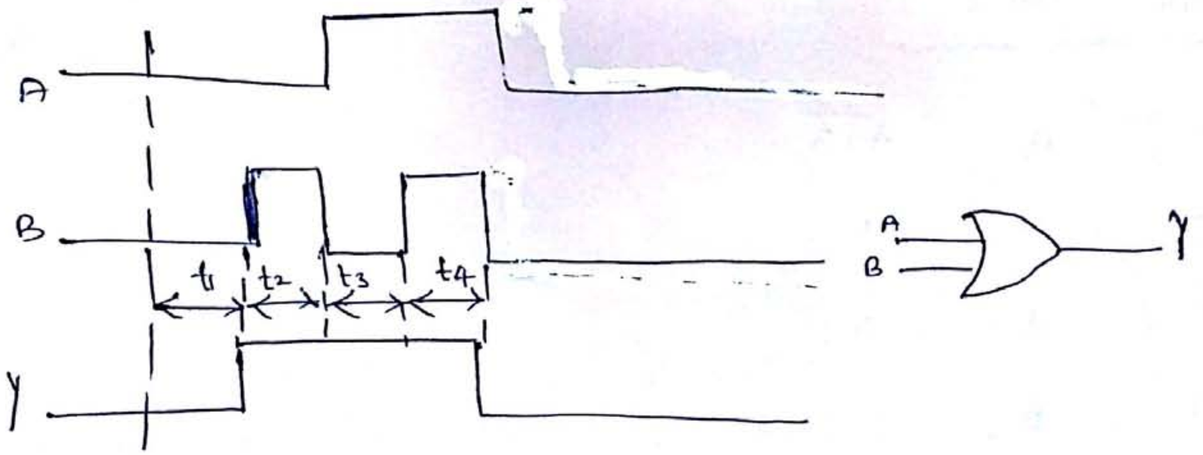


b) When A is low and B is high the diode A is RB and diode B is FB .. which produces o/p voltage of 5V. So the o/p is high

Same as Case b.

4) when both the i/p's are at +5V, both diodes are F.B. Since the i/p voltages, are in parallel, the o/p is ideally +5V, o/p is high.

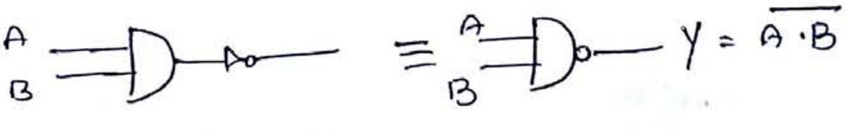
pulsed operation



NAND Gate :(universal)

A NAND gate is a logic gate with two or more inputs are only one output. It's o/p is only when Each any one & all of its input is a 1.

It is AND gate followed by a inverter.

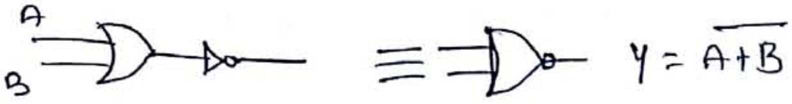


Truth table

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

NOR Gate:

A NOR gate is a logic gate with two or more in only one o/p. its o/p is one only when all inputs a



$Y = \overline{A+B}$

Truth table

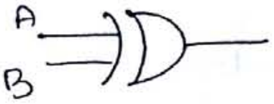
A	B	$\overline{A+B}$
0	0	1
0	1	0
1	0	0
1	1	0

Ex-OR: (anti coincidence gate or inequality detector)

It produces an o/p 1 only when the inputs are not equal i.e if they are odd no. of 1's then o/p is 1 otherwise 0.

Truth table

A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0



$Y = A \oplus B$.

Boolean Expression $Y = A \oplus B$

$= AB' + A'B$.

ecties of XOR Gates

property1: $A \oplus A = 0$: output is logic zero when inputs are same.

property2: $A \oplus \overline{A} = 1$: output is logic 1 when inputs are different.

property 3: $A \oplus 1 = \overline{A}$: Ex-OR as inverter.

When one input of Ex-OR gate is connected to logic one we get the complement of the other input at the o/p of Ex-OR gate.

$0 \oplus 0 = 0$

$0 \oplus 1 = 1$

$$\text{input tied to logic1} \quad \boxed{\begin{matrix}1\\1\end{matrix}} \begin{matrix}\oplus\\\oplus\end{matrix} \underbrace{\boxed{\begin{matrix}0\\1\end{matrix}}}_{\text{other input}} \begin{matrix}=\\=\end{matrix} \boxed{\begin{matrix}1\\0\end{matrix}}$$

out put is complement form of other input.

property4: $A \oplus 0 = A$ Ex-OR as Non-inverter.

When one input of Ex-OR gate is connected to logic 0 we get the un complement of the other input at the o/p of Ex-OR gate.

$$\text{input is Grounded} \quad \boxed{\begin{matrix}0\\0\end{matrix}} \begin{matrix}\oplus\\\oplus\end{matrix} \boxed{\begin{matrix}0\\1\end{matrix}} \begin{matrix}=\\=\end{matrix} \boxed{\begin{matrix}0\\1\end{matrix}}$$

Output is from complement of other input.

$1 \oplus 0 = 1$

$1 \oplus 1 = 0$

property 5: EX-OR as modolo 2 Adder.

The Exclusive-OR gate can be used as a modulo 2 adder its truth table is Same as the truth table of modolo 2 add

$0 + 0 = 0$	$0 \oplus 0 = 0$
$0 + 1 = 1$	$0 \oplus 1 = 1$
$1 \oplus 0 = 1$	$1 \oplus 0 = 1$
$1 + 1 = 0$	$1 \oplus 1 = 0$

property6: $AB \oplus AC = A(B \oplus C)$

A	B	C	$AB \oplus AC$	$A(B \oplus C)$
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	0	0
1	0	0	0	0
1	0	1	1	1
1	1	0	1	1
1	1	1	0	0

property 7: $A \oplus B = C$, then $A \oplus C = B$, $B \oplus C = A$ and $A \oplus B \oplus C = 0$

A	B	$A \oplus B \neq C$	$A \oplus C = B$	$B \oplus C = A$	$A \oplus B \oplus C = 0$
0	0	0	0	0	0
0	1	1	1	0	0
1	0	1	0	1	0
1	1	0	1	1	0

Note: 1. for three input Ex-OR, output is logic 1 only for odd number of logic 1 i/p's

2. No Similar terms for Ex-NOR gate.

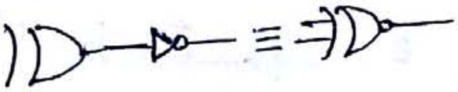
$A \odot B = C$ then $A \odot C = B$ $B \odot C = A$ and $A \odot B \odot C = 1$

-NOR : (Coincidence or Equality detector)

Combination of an X-OR and a NOT gate. It produces an o/p 1 when all the i/p's are '0' or when all i/p's are 1

Truth table

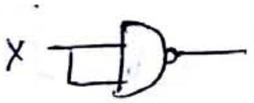
A	B	$A \odot B$
0	0	1
0	1	0
1	0	0
1	1	1



Boolean Expression $Y = A \odot B$
$= AB + A'B'$

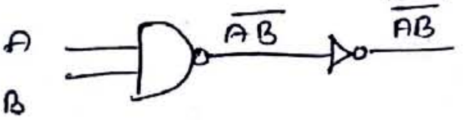
Implementation of NOT, AND, OR and NOR Gates using NAND gate.

NOT function : Considering all the inputs as a single input i.e common input as inverter can be obtained using NAND:



$Y = \overline{X \cdot X} = \overline{X} + \overline{X}$
$= \overline{X}$

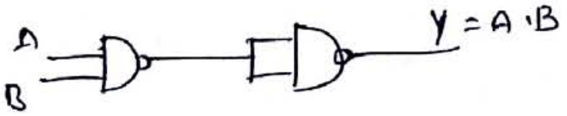
AND function : By connecting an inverter to the NAND we get the AND function.



$\Rightarrow Y = AB$

Truth table

A	B	AB	$\overline{AB}$	$\overline{\overline{AB}}$
0	0	0	1	0
0	1	0	1	0
1	0	0	1	0
1	1	1	0	1

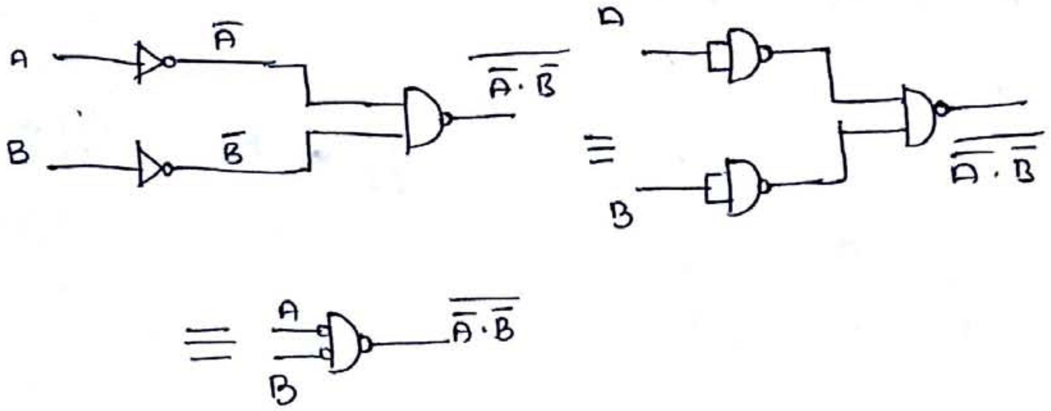


OR function: By connecting an inverters to input of th

Boolean Expression $Y = A+B$ or $\bar{\bar{A}} + \bar{\bar{B}}$

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

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

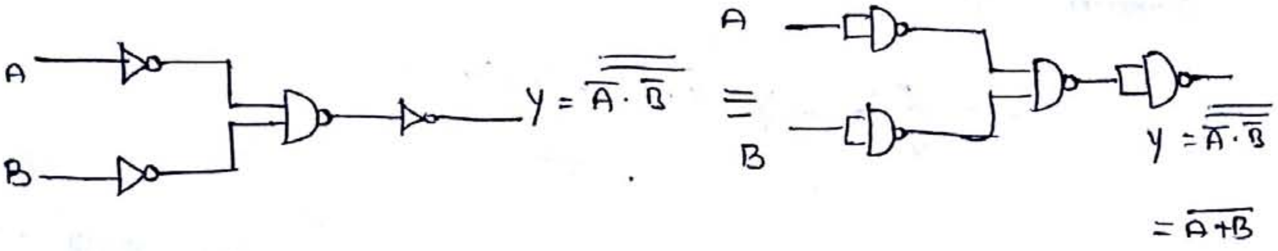

A
B
$\bar{A}$
$\bar{B}$
$\overline{\bar{A}\cdot\bar{B}}$
≡
A
B
$\overline{\bar{A}\cdot\bar{B}}$
≡
A
B
$\bar{A}\cdot\bar{B}$

NOR function

Boolean Expression

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

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


A
B
$Y = \overline{\overline{\bar{A}\cdot\bar{B}}}$
≡
A
B
$Y = \overline{\overline{\bar{A}\cdot\bar{B}}}$
$= \overline{A+B}$

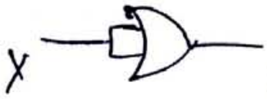
Truth table

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

NOR gate:

NOT Gate:

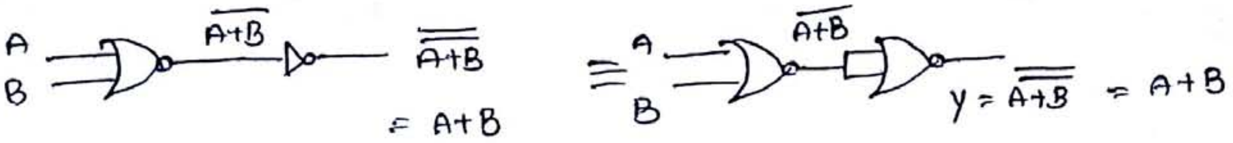
By connecting all inputs to a single common input then NOT gate is made.



$\overline{X+X} = \overline{X}\cdot\overline{X} = \overline{X}$

OR function:

obtained by inverting the o/p of NOR gate.

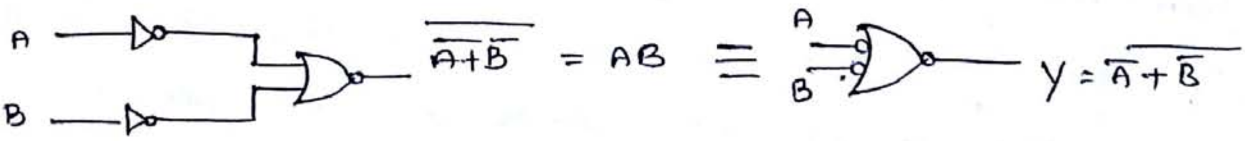

A
B
$\overline{A+B}$
$\overline{\overline{A+B}}$
$= A+B$
≡
A
B
$\overline{A+B}$
$Y = \overline{\overline{A+B}} = A+B$

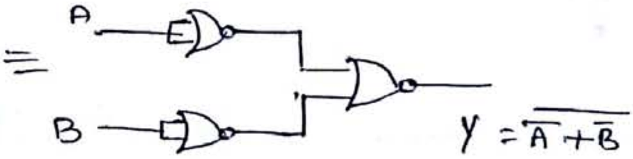
Truth table

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

AND function:

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


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

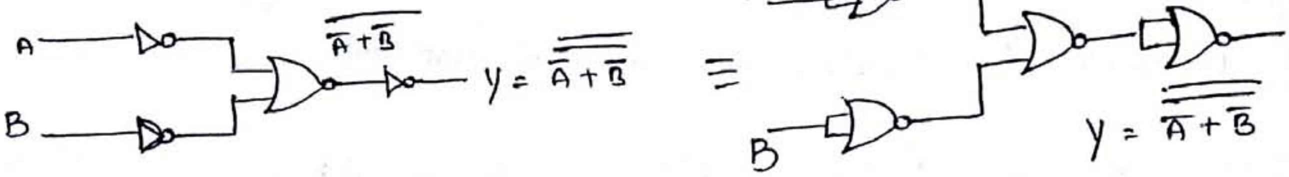


Truth table

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

NAND function

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


A
B
$\overline{\bar{A}+\bar{B}}$
$y = \overline{\overline{\bar{A}+\bar{B}}}$
≡
A
B
$y = \overline{\overline{\bar{A}+\bar{B}}}$

Truth table

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

Conversion of AND | OR | NOT logic to NAND/NOR logic using Graphical procedure

1) Draw AND | OR logic

2) If NAND h/w has been choosen, add bobbles on the o/p of Each AND gate and on input side to all OR gates.

3) If NOR gate h/w has been choosen, add bobbles on the o/p of Each OR gate and bobbles on input side to all AND gates.

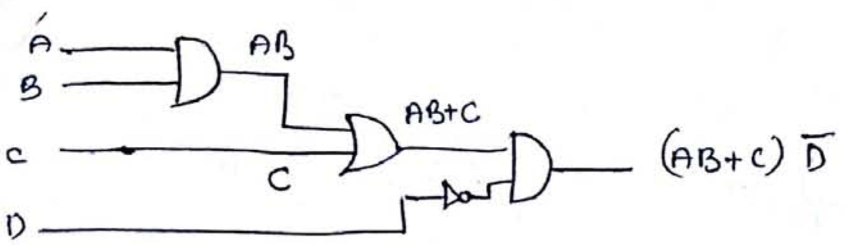
4) Add or Subtract an inverter on Each line that Rred a bobble

Replace bubbled OR by NAND and bubbled AND by NOR

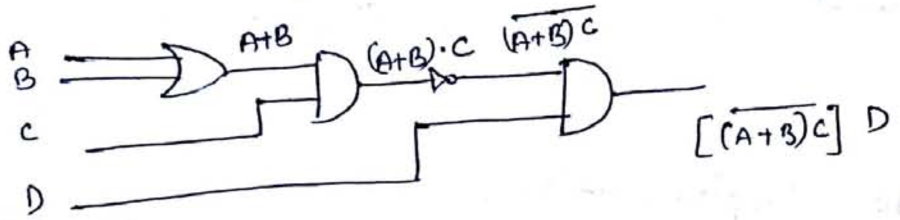
eliminate double inversions.

→ Draw the AOI logic for the Expression $(AB+C)\overline{D}$

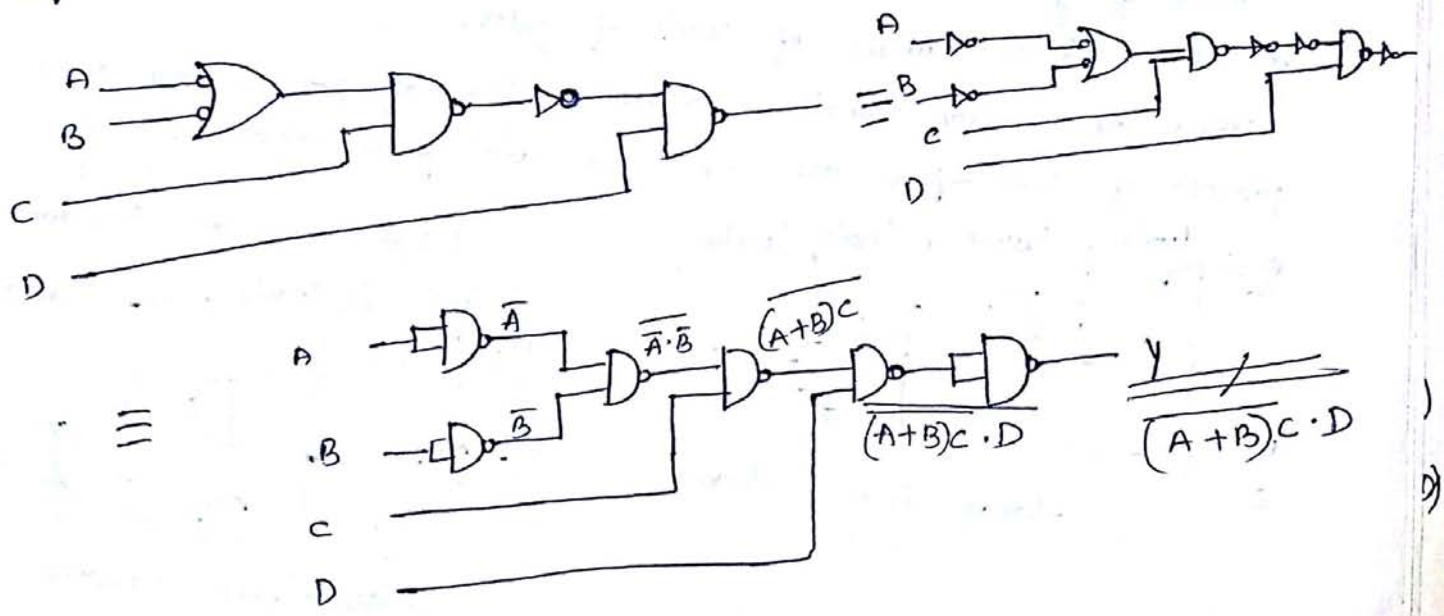
Sol



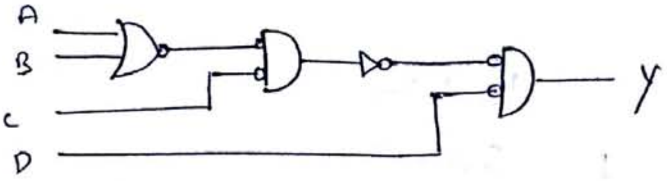
→ Given boolean Expression $y = [\overline{(A+B)C}]\,D$

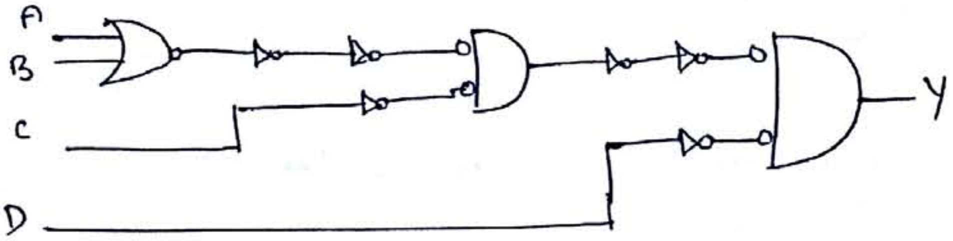


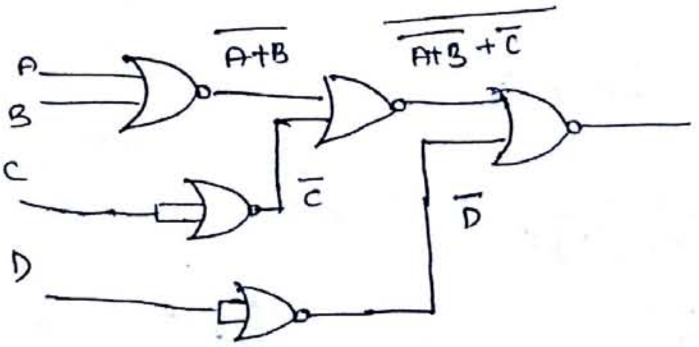
Using NAND


A
B
C
D
≡
A
B
C
D
≡
A
B
C
D
$\overline{A}$
$\overline{B}$
$\overline{A}\cdot\overline{B}$
$\overline{(A+B)C}$
$\overline{\overline{(A+B)C}\cdot D}$
$Y = \overline{(A+B)C}\cdot D$

NOR circuit







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

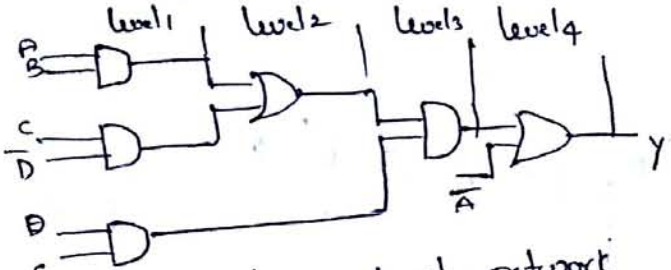
$$= \overline{\overline{(A+B)C} + \overline{D}}$$

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

→ multilevel Gate Implementation.

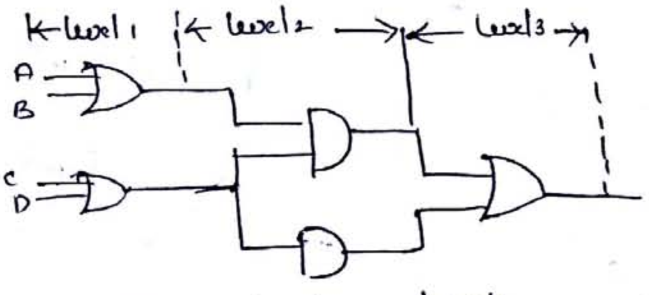
Logic gates are cascaded to get the desired output. The maximum number of gates cascaded in series between a network input and the o/p is refered to as number of levels of gates.

Thus the boolean functions written in sum-of-products form or in product of sum form are the two-level gate networks.



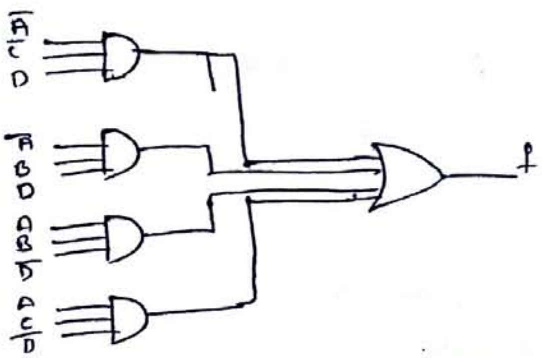
four-level network

let us consider Boolean function



Three-level network.

$$F = \overline{A}\,\overline{C}\,D + A\overline{B}D + AB\overline{D} + AC\overline{D}.$$



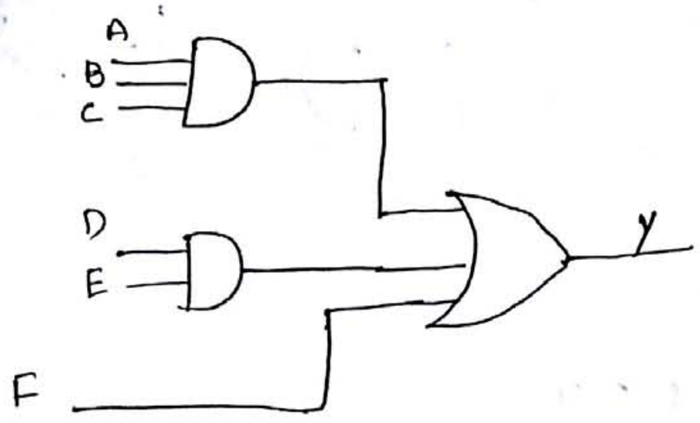
level 2

Gates 5

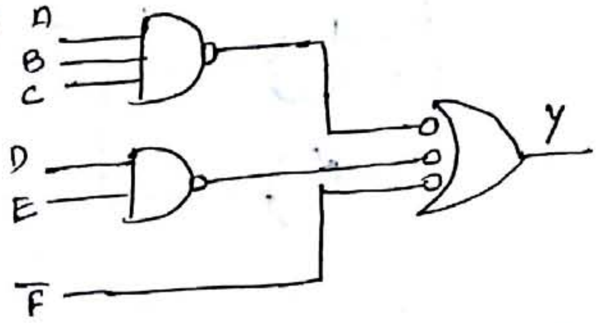
Gate inputs 16

ement the following using NAND - NAND

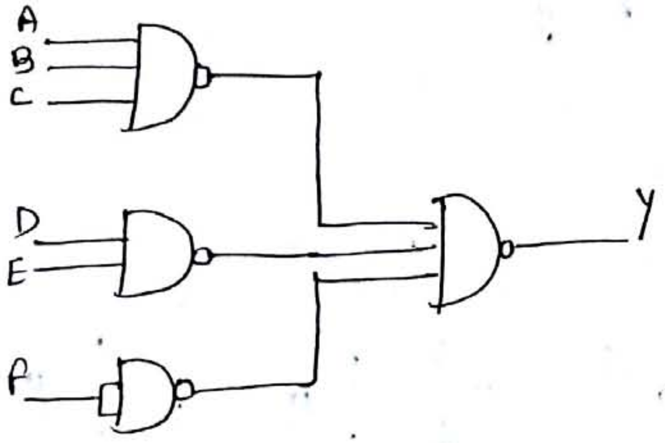
a) $Y = ABC + DE + F$



AND - OR



NAND - Bobbled OR



NAND - NAND.

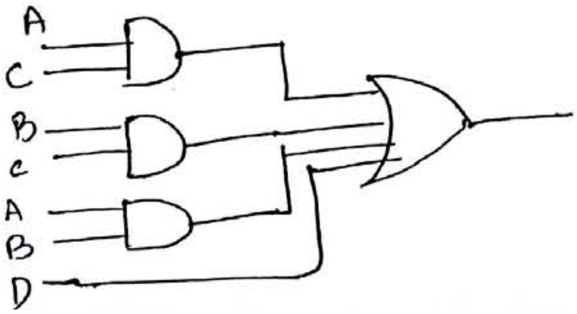
b) $Y = AC + ABC + A'BC + AB + D$

$AC + BC(A + A') + AB + D$

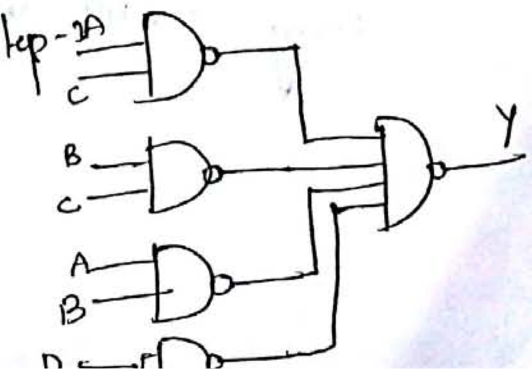
step-1 $AC(1+B) + A'BC + AB + D$. OR $AC + BC + AB + D$.

$AC + A'BC + AB + D$.

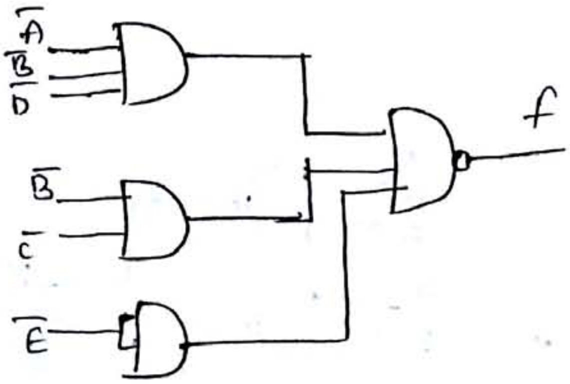
Step-2



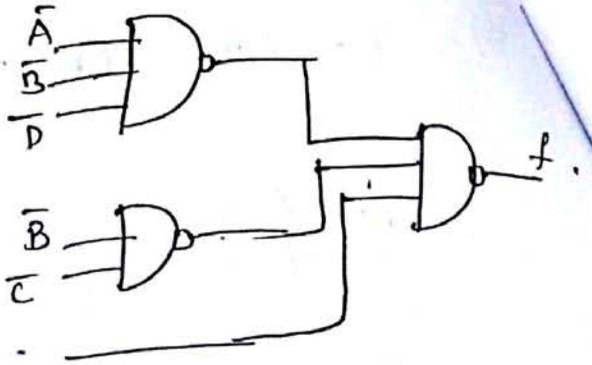
Step-2A



d) $f = A'B'D' + B'C' + E'$



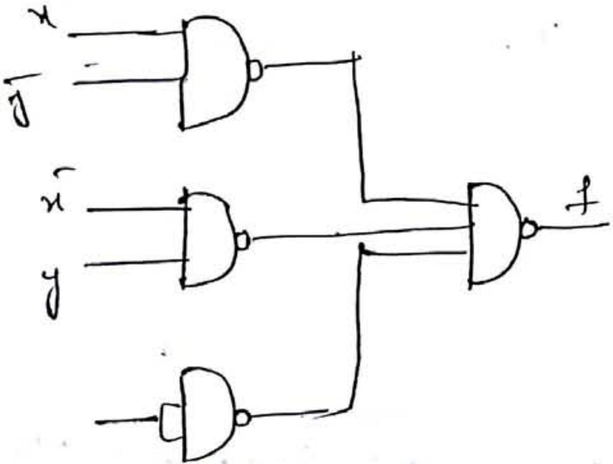
OR



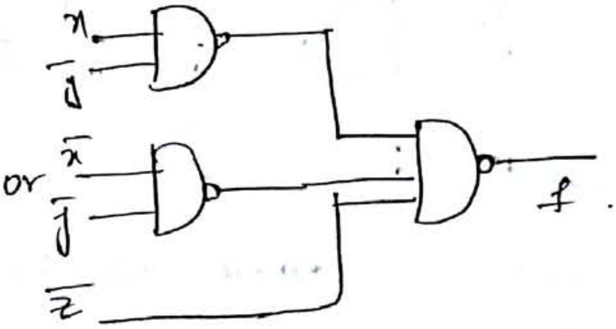
e) $f(x, y, z) = \sum 1, 2, 3, 4, 5, 7$

$x \backslash yz$	00	01	11	10
0		1	1	1
1	1	1	1	

$= z + \bar{x}y + x\bar{y}$



or



f) Find the Reduced pos form of the following Equation $f(A, B, C, D) = \sum m(1, 3, 7, 11, 15) + \sum d(0, 2, 5)$ implement using NAND logic

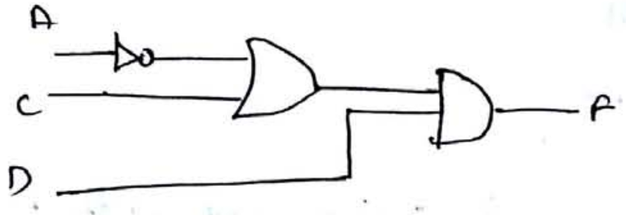
AB \ CD	00	01	11	10
00	X	1	1	X
01	0	X	1	0
11	0	0	1	0
10	0	0	1	0

$$\overline{F} = \overline{D} + A\overline{C}$$

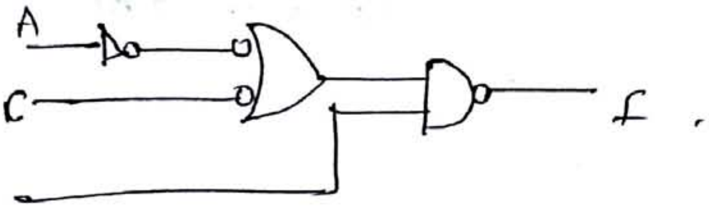
$$F = \overline{\overline{D} + A\overline{C}}$$

$$= \overline{\overline{D}} \left(\overline{A \cdot \overline{C}} \right)$$

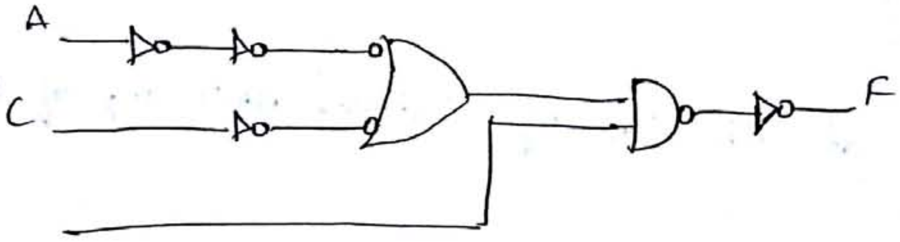
$$= \overline{\overline{D}} \left(\overline{A} + C \right)$$



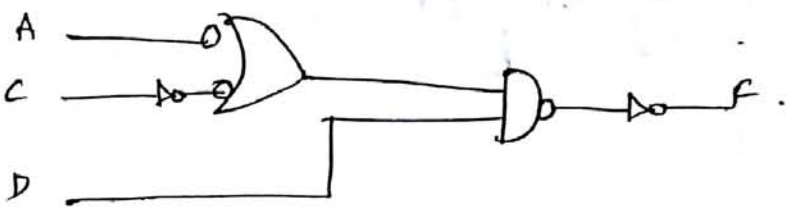
Step1 put bubbles at i/p of OR & o/p of AND.



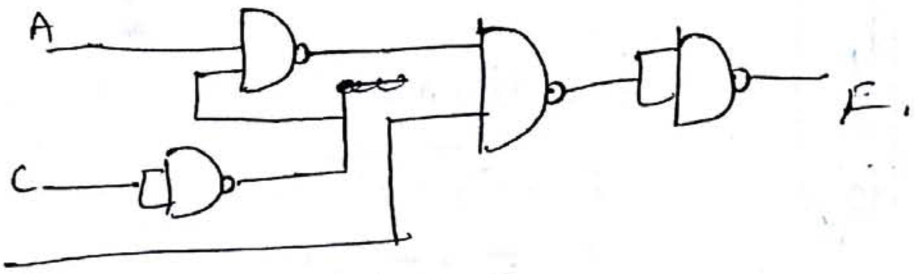
Step2 insert inverter for Each bubble.



Step3 Eliminate double conversion.

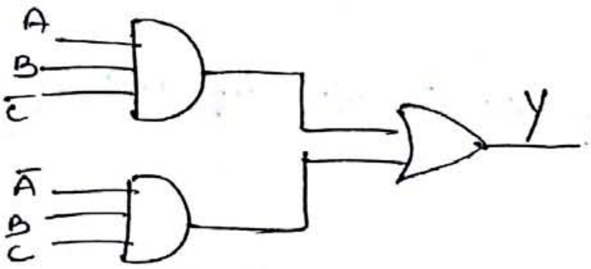


Step4 Replace bobbled OR, and Inverter by NAND place



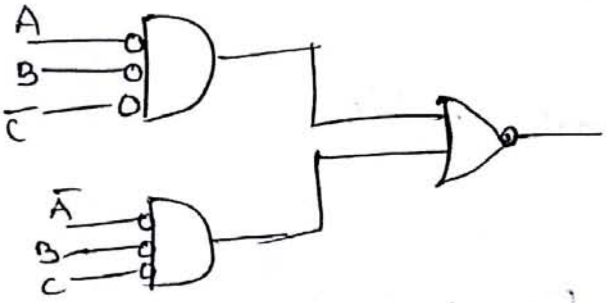
1) $AB\bar{C} + \bar{A}BC$ using a). NOR

b) NAND

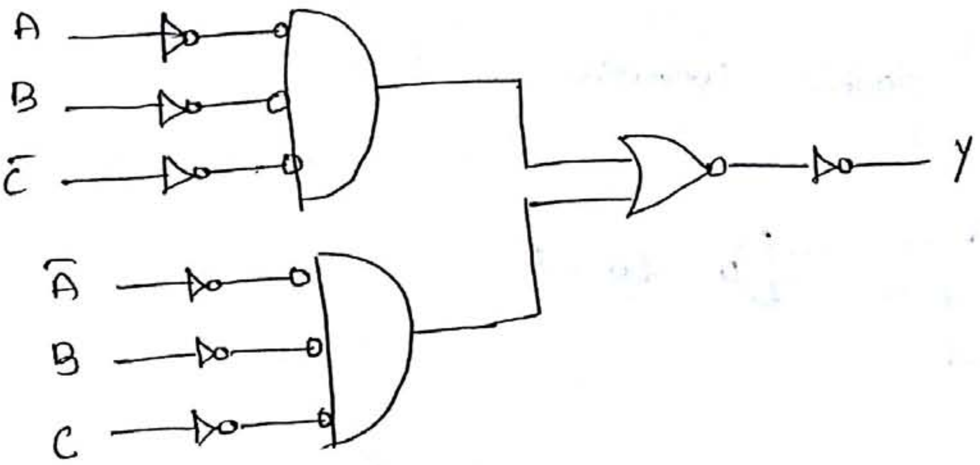


(i) Replace AND with bobbled AND.

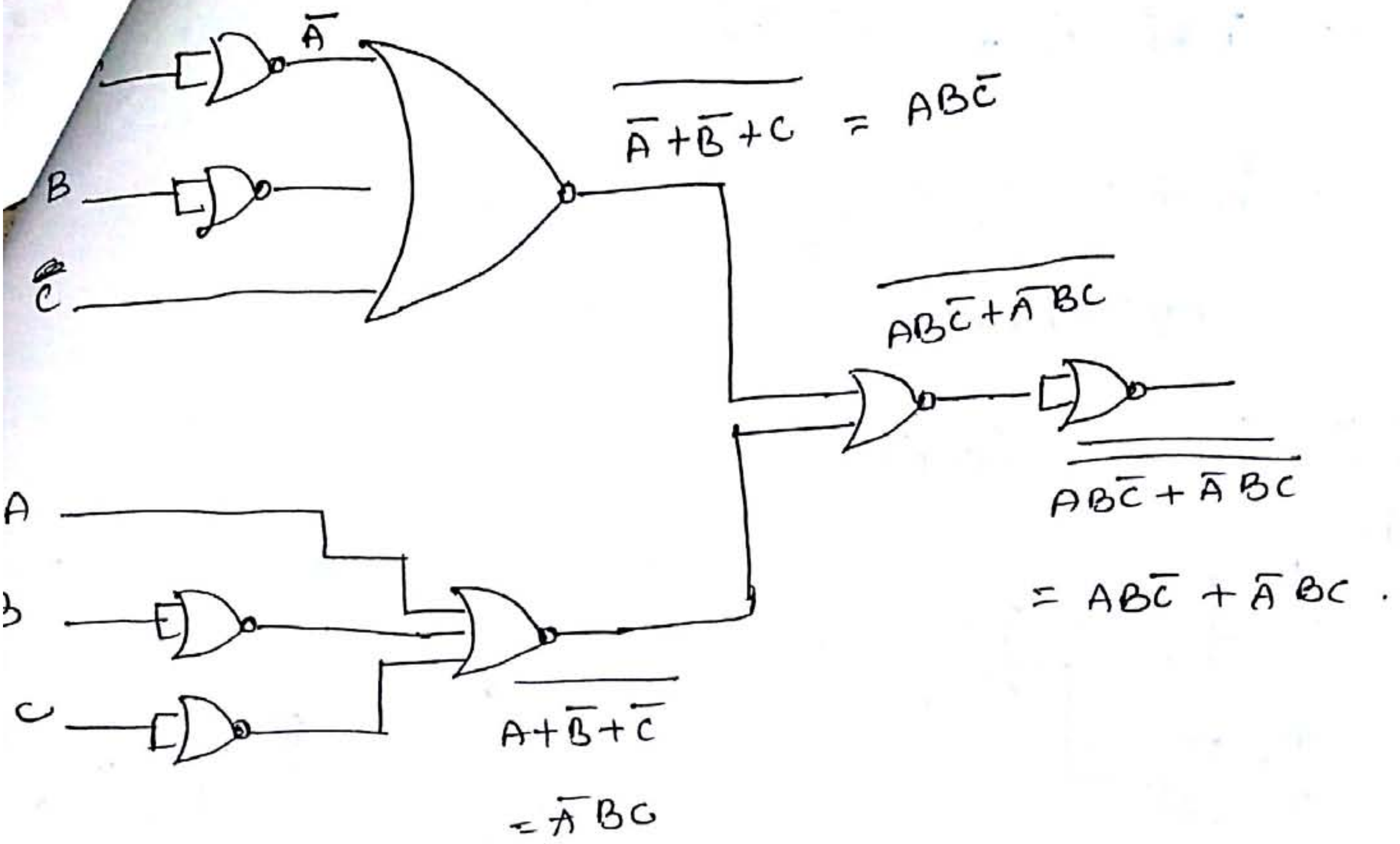
OR with NOR.



ii) Add inverters at the i/p of bobbled AND, o/p of NOR



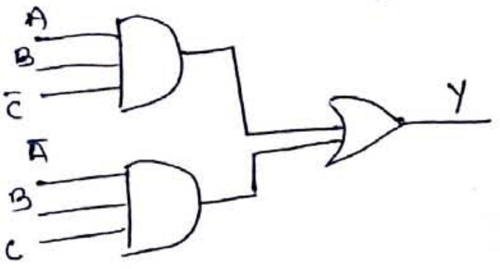
bubbled and, and inverter by NOR.

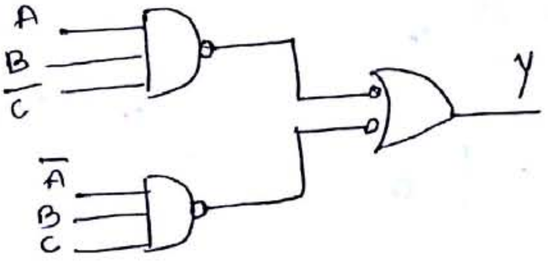


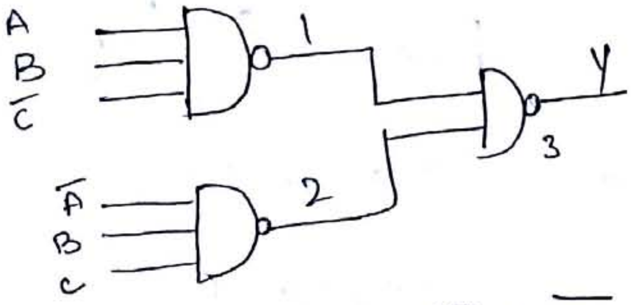
$\bar{A}$
B
C
$\overline{\bar{A}+\bar{B}+C} = AB\bar{C}$
$\overline{AB\bar{C}+\bar{A}BC}$
$\overline{\overline{AB\bar{C}+\bar{A}BC}}$
$= AB\bar{C} + \bar{A}BC$.
A
B
C
$\overline{A+\bar{B}+\bar{C}}$
$= \bar{A}BC$

Using NAND

$Y = AB\bar{C}' + A'BC$







① $\overline{AB\bar{C}} = \bar{A}+\bar{B}+C$ ② $\overline{\bar{A}BC} = A+\bar{B}+\bar{C}$

③ $\overline{(\bar{A}+\bar{B}+C)\ (A+\bar{B}+\bar{C})}$

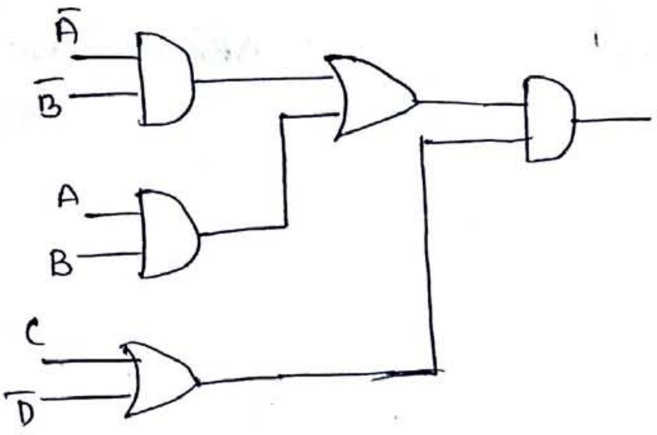
$= \overline{(\bar{A}+\bar{B}+C)} + \overline{(A+\bar{B}+\bar{C})}$

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

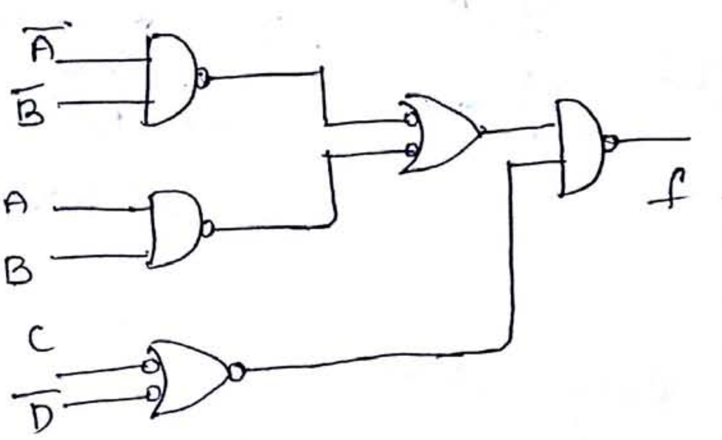
$= AB\bar{C}+\bar{A}BC$

NAND : $(A'B'+AB)\ (C+D')$

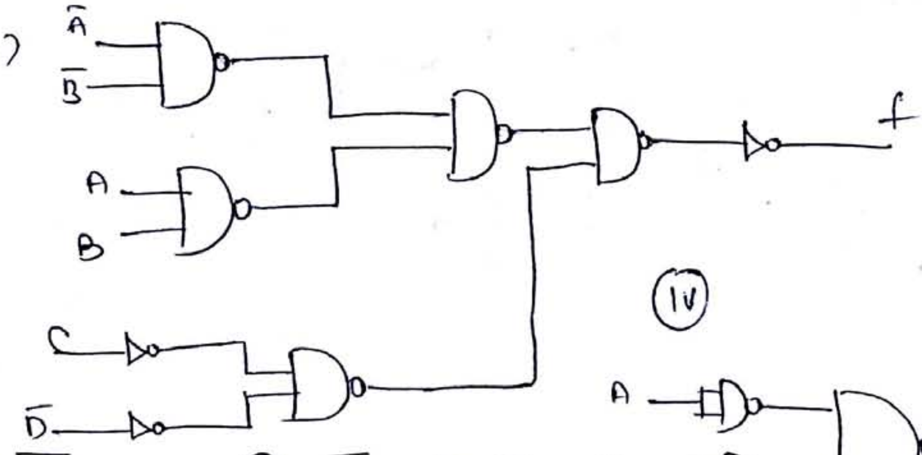
(i)



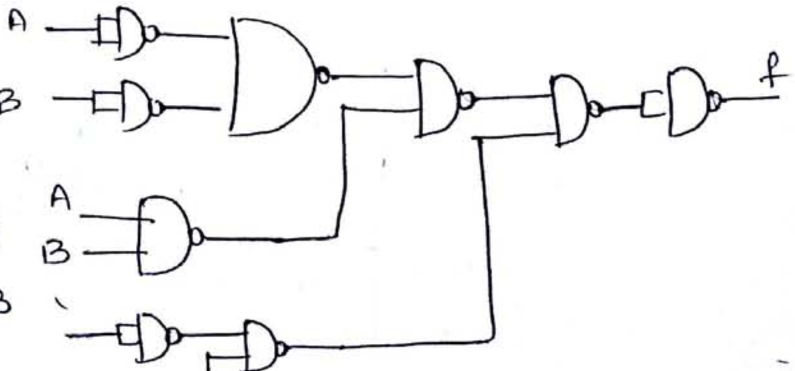
(ii)



(iii)



(IV)



① $\overline{\bar{A}\cdot\bar{B}} = A+B$ ② $\overline{AB} = \bar{A}+\bar{B}$

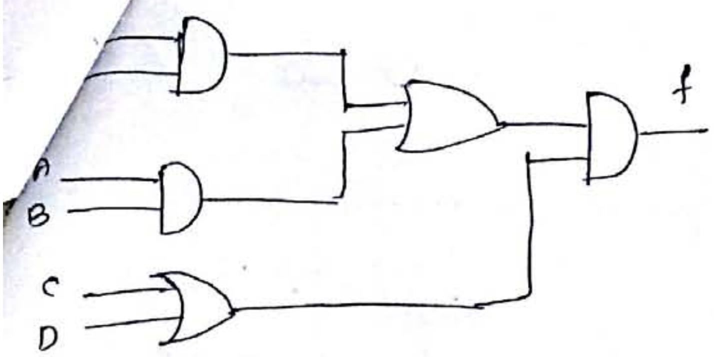
③ $\overline{\bar{C}\cdot D} = C+\bar{D}$

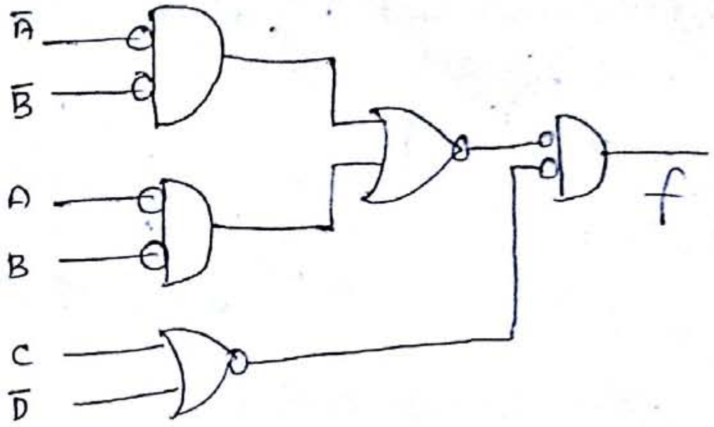
④ $\overline{(A+B)\ (\bar{A}+\bar{B})} = \left(\overline{A+B}\right)+\left(\overline{\bar{A}+\bar{B}}\right)$

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

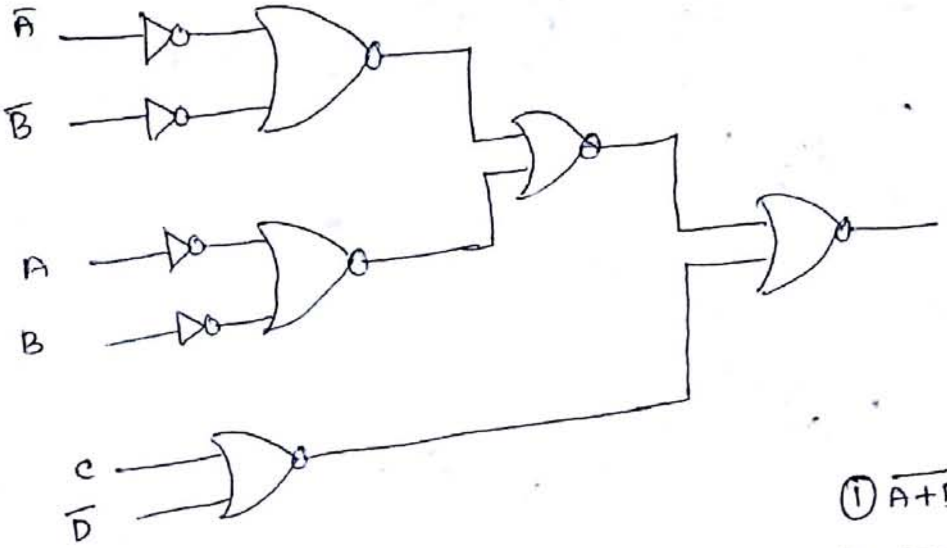
⑤ $(\bar{A}\cdot\bar{B}+A\cdot B)\ (C+\bar{D})$

⑥ $\overline{\overline{(\bar{A}\cdot\bar{B}+A\cdot B)\ (C+\bar{D})}} = (\bar{A}\cdot\bar{B}+AB)\ \overline{(C+\bar{D})}$

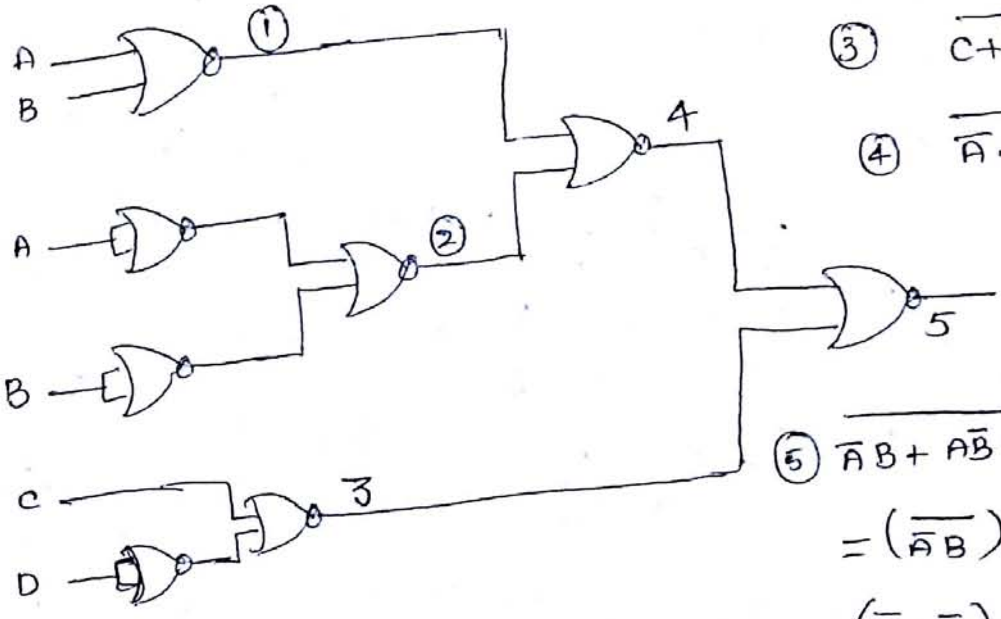




(iii)



(iv)



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

② $\overline{\bar{A}+\bar{B}} = AB$

③ $\overline{C+\bar{D}} = \bar{C}\cdot D$

④ $\overline{\bar{A}\cdot\bar{B}+AB} = (\overline{\bar{A}\cdot\bar{B}})(\overline{AB})$

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

$\Rightarrow (A+B)(\bar{A}+\bar{B})\ (\cancel{\overline{\bar{C}AD}})$

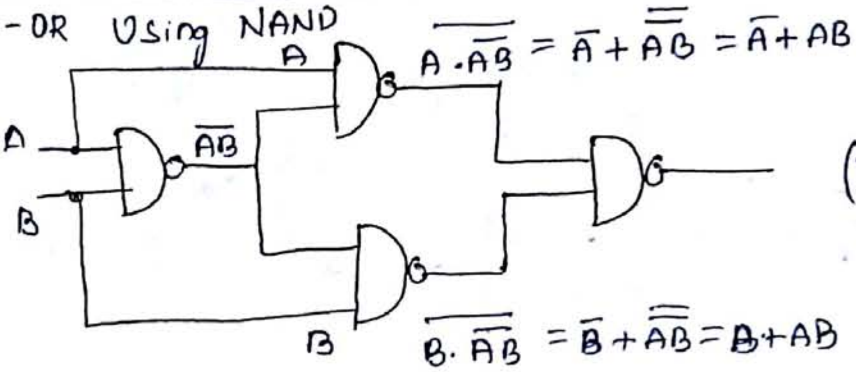
⑤ $\overline{\bar{A}B+A\bar{B}+\bar{C}\cdot D}$

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

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

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

Ex-OR Using NAND


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

$$\overline{(\overline{A}+AB)\ (\overline{B}+AB)}$$

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

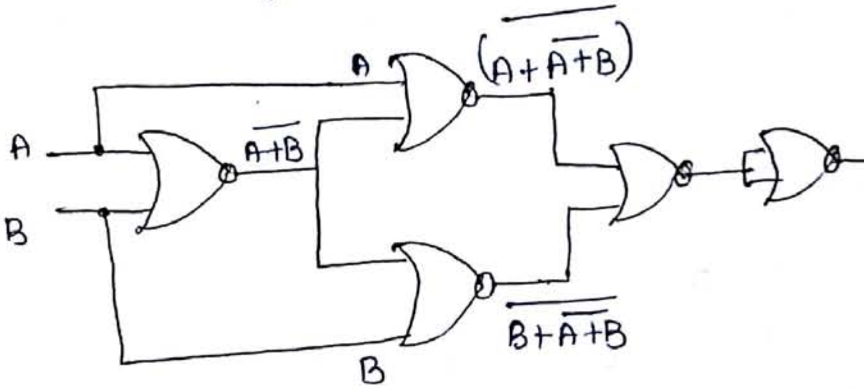
$$\overline{\overline{A}}\cdot(\overline{AB})+(\overline{\overline{B}}\cdot\overline{AB})$$

$$A(\overline{A}+\overline{B})+(B\cdot(\overline{A}+\overline{B})$$

$$A\cdot\overline{A}+AB'+BA'+BB'$$

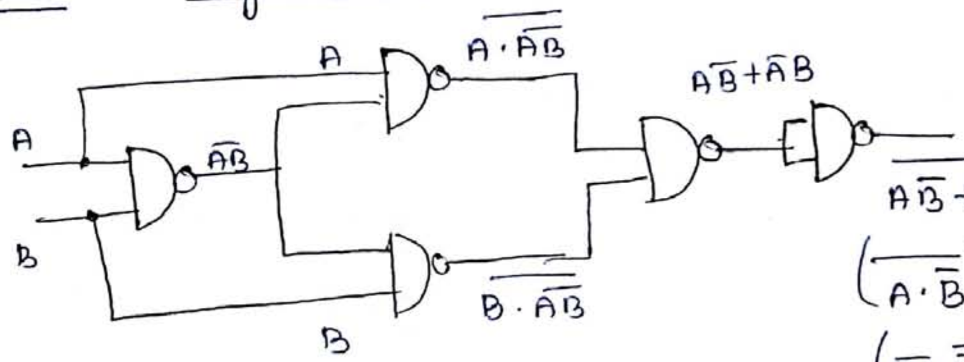
$$= A\cdot B'+A'B$$

Ex-OR Using NOR


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

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

Ex-NOR Using NAND


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

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

$$(\overline{A\cdot\overline{B}})\ (\overline{\overline{A}B})$$

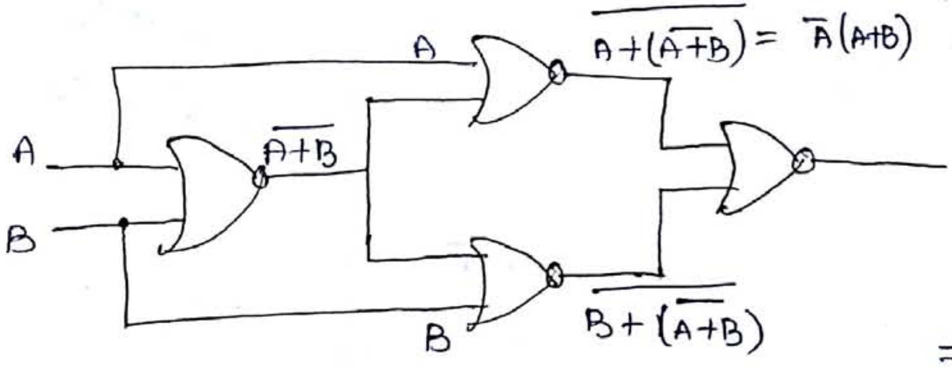
$$(\overline{A}+\overline{\overline{B}})\ (\overline{\overline{A}}+\overline{B})$$

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

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

$$= A'B'+AB$$

Ex-NOR Using NOR


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

$$\overline{B}\cdot(A+B)$$

$$\overline{\overline{A}(A+B)+(\overline{B}\cdot(A+B))}$$

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

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

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

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

$$(B+\overline{B})$$

$$\Rightarrow (A+\overline{B})(\overline{A}+B)$$

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

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