Gatenotes.in

GATE

Computer Science

Ravindrababu Ravula GATE CSE

Hand Written Notes

-: SUBJECT :-

Digital Logic

Standard : Division : Roll :

Subject : DIGITAL LOGIC

INDEX

Serial Number	Date	Title	Page Number	Teacher's Sign/Remarks
1.		Boolean Algebra + Logic gates.		
2.		Minimization.		
3.		Combinational Circuit.		
4.		Sequential circuits.		
5.		Number Systems.		
6.		Computer Arithmetic (fixed and floating point)		

Rakesh Nama

Boolean Algebra + Logic gates

Boolean algebra is an algebric structure defined on a set of elements together with two binary operators (+) and (·) :

→ A variable is a symbol for example 'A' used to represent a logical quantity, whose value can be '0' or '1'.

→ The complement of a variable is the inverse of variable and is represented by an over bare, for example '$\bar{A}$'.

→ A literal is a variable or the complement of variable.

- Closure: For any x and y in the alphabet A, x+y and x.y are also in A.

- Boolean value: The value of Boolean variable can be either '1' or '0'.

- Boolean Operators: There are '4' boolean Operators –

(i) AND (·) Operator (A.B)
(ii) OR (+) Operator (A+B)
(iii) NOT ($\bar{A}$/A') operator.
(iv) XOR (⊕) operator ($A \oplus B = \bar{A}B + B\bar{A}$)

- Operator precedence: The operator for evaluating Boolean expression is –

(i) parenthesis (ii) NOT (iii) AND (iv) OR.

- **Duality:** If an expression contains only the operations AND, OR and NOT. Then the dual of that expression is by replacing each AND by OR, each OR by AND, all occurrences of '1' by '0' and all occurrences of '0' by 1. principle of duality is useful in determining the complement of a function.

logic expression: $(x.y'.z)+(x.y.z')+(y.z)+0$

Duality of above logic expression is:
$(x+y'+z).(x+y+z').(y+z).1$.

- **Boolean function:**

→ Any Boolean functions can be from binary variables and the boolean operators $\cdot$, $+$ and $'$ (for AND, OR and NOT respectively.

→ for a given value of variable, the function can take only one value either '0' or '1'.

→ A Boolean function can be shown by a ~~that~~ truth table. The shown a function in a truth ~~at~~ table we need a list of the 2^n combinations of 0's and 1's of the 'n' binary variables - and a column showing the combination for which the function is equal to 1 or 0. So, the table will have 2^n rows and column for each input variable and the final output.

- A function can be specified or represented —

    1. A truth table.
    2. A circuit.
    3. A Boolean expression.
    4. SOP (sum of products).
    5. POS (product of sum).
    6. Canonical SOP.
    7. Canonical POS.

- Important Boolean operations over boolean values —

AND Operation	OR Operation
$0 \cdot 0 = 0$	$1 + 1 = 1$
$1 \cdot 1 = 1$	$0 + 0 = 0$
$0 \cdot 1 = 1 \cdot 0 = 0$	$1 + 0 = 0 + 1 = 1$
$0' = 1$	$1' = 0$

- Table of some basic theorems —

law/theorem	law of addition	law of multiplication
Identity law	$X + 0 = X$	$X \cdot 1 = X$
complement law	$X + \bar{X} = 1$	$X \cdot \bar{X} = 0$
Idempotent law	$X + X = X$	$X \cdot X = X$
Dominant law	$X + 1 = 1$	$X \cdot 0 = 0$
Involution law	$(X')' = X$	
Commutative law	$X + Y = Y + X$	$X \cdot Y = Y \cdot X$

Associative law	$x+(y+z)=(x+y)+z$	$x\cdot(y\cdot z)=(x\cdot y)\cdot z$
Distributive law	$x+yz=(x+y)(x+z)$	$A(B+C)=AB+AC$
Demorgan's law	$(x+y)'=x'\cdot y'$	$(x\cdot y)'=x'+y'$.
Absorption law	$x+xy=x$	$x(x+y)=x$.

- Important theorem used in simplification =

→ NOT - Operation theorem : $\bar{\bar{A}} = A$.

→ AND - operation theorem:

$A\cdot A = A$

$A\cdot 1 = A$

$A\cdot 0 = 0$

$A\cdot \bar{A} = 0$

→ OR - operation theorem:

$A+A = A$

$A+1 = 1$

$A+0 = A$

$A+\bar{A} = 1$

- Distribution theorem =

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

Note-

$(A+\bar{A}B) \rightarrow (A+B)$

$(A+\bar{A}\bar{B}) \rightarrow (A+\bar{B})$

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

$(\bar{A}+A\bar{B}) \rightarrow (\bar{A}+\bar{B})$

- Demorgan's theorem :

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

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

- Transposition theorem :

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

- Consensus Theorem : This theorum is used to eliminate redundant term. It is applicable only when if a boolean function contain three variable. Each variable are used to times. only one variable is complemented or uncomplimented. Then the related terms so that complemented or uncomplemented variable is the answer -

Ex -

(I) $AB+\bar{B}C+AC = AB+\bar{B}C.$

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

(III) $AB+\bar{A}C+BC = AB+\bar{A}C.$

(IV) $A\bar{B}+AC+BC = A\bar{B}+BC.$

• Problems on Boolean algebra —

• Simplify the following Boolean expressions to a minimum number of literals —

LEVEL-1

(a) $xy' + xy'$

$\Rightarrow x(y+y')$

$\Rightarrow x \cdot 1 \Rightarrow 1$

(b) $(x+y)(x+\bar{y})$

$\Rightarrow x + y\bar{y}$ (transposition theorem)

$\Rightarrow x + 0 \Rightarrow x.$

(c) $xyz + \bar{x}y + xy\bar{z}$

$\Rightarrow xy(z+\bar{z}) + \bar{x}y$

$\Rightarrow xy + \bar{x}y$

$\Rightarrow y(x+\bar{x})$

$\Rightarrow y \cdot 1 \Rightarrow y$

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

$\Rightarrow (\bar{A} \cdot \bar{B})(A \cdot B)$

$\Rightarrow 0 \cdot 0 \Rightarrow 0.$

LEVEL-2

(1) $\bar{A}\bar{C} + ABC + A\bar{C} + A\bar{B}$ (reduce to two literals)

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

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

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

$\Rightarrow \bar{C} + A\bar{B} + AC$

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

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

$\Rightarrow A(1+\bar{B}) + \bar{C}$

$\Rightarrow A + \bar{C}.$

Page 7

(2) $\bar{A}B(\bar{D}+CD) + B(\bar{A}+\bar{A}CD)$ → reduce to one literals.

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

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

⟹ $B(\bar{A}\bar{D}+\bar{A}+C+CD)$

⟹ $B(\bar{A}+\bar{D}+C+CD)$

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

⟹ $B(\bar{A}+C+\bar{D})$

⟹ $B(\bar{\bar{A}}+\bar{\bar{C}}+\bar{D})$ (Demorgan's law)

⟹ $B\,\overline{(\bar{A}.\bar{C}.D)}$

$\bar{a}\cdot\bar{b} = \overline{(a+b)}$

- Find the complement –

(a) $F = x\bar{y} + \bar{x}y$

→ $F_{dual} = (x+\bar{y})(\bar{x}+y)$

$F_c = (\bar{x}+y)\cdot(x+\bar{y})$

OR

$\bar{F} = \overline{(x\bar{y}+\bar{x}y)}$

⟹ $\overline{x\bar{y}} \cdot \overline{\bar{x}y}$

⟹ $(\bar{x}+y)(x+\bar{y})$

(b) $F = (\bar{A}B+CD)\bar{E}+E$

→ $F_{dual} = \{(\bar{A}+B)\cdot(C+D)+\bar{E}\}\cdot E$ $\boxed{E\cdot\bar{E}=0}$

$F_{com} = \{(A+\bar{B})\cdot(\bar{C}+\bar{D})+E\}\bar{E}$

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

(c) $(\bar{x}+y+\bar{z})\cdot(x+\bar{y})\;(x+z)$

$\Rightarrow$

$$f_d = \bar{x}\cdot y\bar{z} + x\bar{y} + xz$$

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

- List the truth table of the function $\Rightarrow$

(a) $F = xy + \bar{x}\bar{y} + \bar{y}z$

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

$= y + \bar{y}z$ $\qquad \boxed{A+BC = (A+B)(A+C)}$

$F = y + z$

table -

x	y	z	F = y+z
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

(b) $F = \bar{x}\bar{z} + yz$.

$\Rightarrow$

x	y	z	$f = \bar{x}\bar{z} + yz$
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

$1\cdot1 + 0\cdot0 = 1$

$1\cdot0 + 0\cdot1 = 0$

$1\cdot0 + 1$

Q) - Reduce the expression, $f = \overline{(A+\overline{B}C)}\ (A\overline{B}+ABC)$

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

$\Rightarrow (\overline{A}\cdot\overline{\overline{B}C})\ (A(\overline{B}+BC))$

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

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

$\Rightarrow 0+0+0+0 \Rightarrow 0.$

Q) Reduce the function to one literal.

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

$\rightarrow$ $f = (B+BC)\ (B+\overline{B}C)\ (B+D)$

$\Rightarrow B(1+C)\ ((B+\overline{B})(B+C))\ (B+D)$

$\Rightarrow B(1+C)\ (B+C)\ (B+D)$

$\Rightarrow B\ (B+C)\ (B+D)$

$\Rightarrow B\ (B+BD+BC+CD)$

$\Rightarrow B\ (B+CD)$

$\Rightarrow B+BCD$

$\Rightarrow B(1+CD)$

$\Rightarrow B.$

Q) Reduce the Boolean expression -

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

$\Rightarrow AB+AB+AC+\underbrace{\overline{B}B}_{0}+\overline{B}D$

$\Rightarrow AB+AC+B\overline{D}$

Q. Reduce the boolean expression -

$$F = \bar{A}B + \bar{A}B\bar{C} + \bar{A}BCD + \bar{A}B\bar{C}\bar{D}E.$$

$\Rightarrow \bar{A}B\,(1+\bar{C}+CD+\bar{C}\bar{D}E)$

$\Rightarrow \bar{A}B \cdot 1$

$\Rightarrow \bar{A}B.$

Q. Reduce the boolean expression -

$$F = AB + \overline{AC} + A\bar{B}C\,(AB+C)$$

$\Rightarrow AB + \bar{A} + \bar{C} + 0 + A\bar{B}C$

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

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

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

$\Rightarrow 1 + B + \bar{C}$

$\Rightarrow 1.$

Q. Reduce the given boolean ex -

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

$\Rightarrow (\overline{\bar{A}B}) \cdot (\overline{\bar{A}+AB})$ $\qquad$ $A+BC = (A+B)(A+C)$

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

$\Rightarrow (AB)\ (A\bar{B})$ $\qquad$ $\overline{\bar{A}+B} = A.\bar{B}$

$= 0.$

Q If $x=1$ in the logic function –

$$[x+z\{\bar{y}+(\bar{z}+x\bar{y})\}]\ \{\bar{x}+\bar{z}(x+y)\}=1$$

(a) $y=z$ (b) $y=\bar{z}$ (c) $z=1$ ✓(d) $z=0$.

$$\rightarrow \boxed{[1+z\{\bar{y}+(\bar{z}+x\bar{y})\}]}\ \{0+\bar{z}(1+y)\}=1$$

$\Rightarrow 1.\bar{z}=1$

$\Rightarrow \bar{z}=1$

$z=0$

Q The boolean expression –

$$y=\bar{A}\bar{B}\bar{C}D+\bar{A}BC\bar{D}+A\bar{B}\bar{C}D+AB\bar{C}\bar{D}$$

reduce to

(a) $y=\bar{A}\bar{B}\bar{C}D \quad \bar{A}B\bar{C} \quad A\bar{C}D$

(b) $y=\overline{A}\overline{B}\overline{C}D \quad BC\bar{D} \quad A\bar{B}\bar{C}D$

(c) $y=\bar{A}BC\bar{D} \quad \bar{B}\bar{C}D \quad A\bar{B}\bar{C}D$

✓(d) $y=\bar{A}BC\bar{D} \quad \bar{B}\bar{C}D \quad AB\bar{C}\bar{D}$

$\rightarrow$ Solu $y=\bar{A}\underline{\bar{B}\bar{C}D}+\bar{A}BC\bar{D}+A\underline{\bar{B}\bar{C}D}+AB\bar{C}\bar{D}$

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

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

Q $W=R+\bar{P}Q+\bar{R}S$.

$X=PQ\bar{R}\bar{S}+\bar{P}\bar{Q}\bar{R}\bar{S}+P\bar{Q}\bar{R}\bar{S}$

$Y=\overline{RS+PR+\bar{P}\bar{Q}+\bar{P}\bar{Q}}$

$Z=\overline{R+S+PQ+\bar{P}\bar{Q}\bar{R}+P\bar{Q}\bar{S}}$

then

✓(a) $W=Z,\ X=\bar{Z}$ (b) $W=Z,\ X=Y$

(c) $W=Y$ (d) $W=Y=\bar{Z}$

→

$$W = R + \bar{P}Q + \bar{R}S$$

$$= (R+\bar{R})(R+S) + \bar{P}Q \qquad \boxed{A+BC = (A+B)(A+C)}$$

$$= R + S + \bar{P}Q \text{ ——— (I)}$$

$$X = PQ\bar{R}\bar{S} + \bar{P}\bar{Q}\bar{R}\bar{S} + P\bar{Q}\bar{R}\bar{S}$$

$$= PQ\bar{R}\bar{S} + \bar{Q}\bar{R}\bar{S}(\bar{P}+P)$$

$$= \bar{R}\bar{S}(PQ+\bar{Q})$$

$$= \bar{R}\bar{S}(P+\bar{Q})$$

$$= \bar{R}\bar{S}P + \bar{R}\bar{S}\bar{Q} \text{ ——— (II)}$$

$$Y = RS + \overline{PR + \bar{P}\bar{Q} + \bar{P}Q}$$

$$= RS + \overline{PR + \bar{P}}$$

$$= RS + \overline{PR}\ \overline{\bar{P}}$$

$$= RS + (\bar{P}+\bar{R})(P)$$

$$= RS + \bar{P}Q + P\bar{R} + \bar{R}Q \text{ ——— (III)}$$

$$Z = R + S + \overline{PQ + \bar{P}\bar{Q}\bar{R} + P\bar{Q}\bar{S}}$$

$$= R + S + \overline{P(Q+\bar{Q}\bar{S}) + \bar{P}\bar{Q}\bar{R}}$$

$$= R + S + \overline{P(Q+\bar{S}) + \bar{P}\bar{Q}\bar{R}}$$

$$= R + S + \overline{PQ + P\bar{S} + \bar{P}\bar{Q}\bar{R}}$$

$$= R + S + \overline{PQ} \cdot \overline{P\bar{S}} \cdot \overline{\bar{P}\bar{Q}\bar{R}}$$

$$= R + S + (\bar{P}+\bar{Q})(\bar{P}+S)(P+Q+R)$$

$$= R + S + (\bar{P} + \bar{P}S + \bar{Q}\bar{P} + \bar{Q}S)(P+Q+R)$$

$$= R + S + (\bar{P} + \bar{Q}\bar{P} + \bar{Q}S)(P+Q+R)$$

$= R+S+\bar{P}Q+\bar{P}R+\bar{Q}\bar{P}R+P\bar{Q}S+\bar{Q}SR$

$= R+S+\bar{P}Q+\bar{P}R+P\bar{Q}S+\bar{Q}RS$

$= R+\bar{P}R+\bar{Q}RS + S+P\bar{Q}S+\bar{P}Q$

$= R[1+\bar{P}+\bar{Q}S]+S[1+P\bar{Q}]+\bar{P}Q.$

$Z = R+S+\bar{P}Q.$ ——— (IV)

here, equ (I) and (IV) same equation.

$\boxed{W=Z}$,

$Z_{dual} = RS(\bar{P}+Q)$

$X = \bar{R}\bar{S}(P+\bar{Q}).$

$Z = R+S+\bar{P}Q.$

$Z_c = \bar{R}\bar{S}(P+\bar{Q})$

$\bar{Z} = \overline{R+S+\bar{P}Q}$

$= \bar{R}\bar{S}(P+\bar{Q})$

$\boxed{X=\bar{Z}}$

- **Logic gates** — A logic gate is an idealised or physical device implementing a Boolean function, that is, it performs a logical operation in one or more ~~log~~ logic i/p and produce a single logic o/p.

Logic gates can be classified as —

(I) NOT, AND, OR are basic gates.
(II) NAND, NOR are universal gates.
(III) EXOR, EXNOR are arithmetic circuit or code convertor or comparators.

- **Not gate (Inverter):**

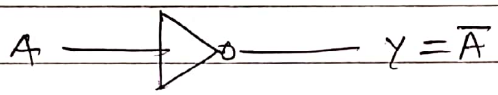


table —

i/p A	o/p $Y = \overline{A}$
0	1
1	0

- **AND gate** —

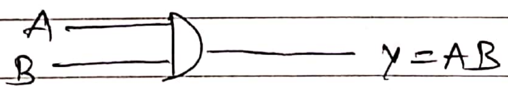


table —

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

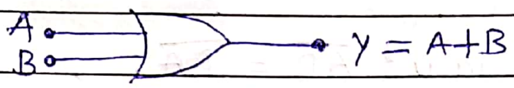
Properties of AND gate —

→ follow Commutative and Associative law.

$AB = BA$

$ABC = A(BC) = (AC)B.$

- **OR Gate –**



(OR gate symbol)

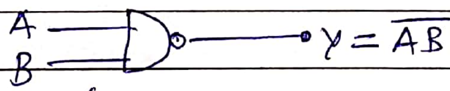
Truth table of OR Gate:

Inputs		output
A	B	$Y = A+B$
0	0	0
0	1	1
1	0	1
1	1	1

properties of OR logic –

(I) commutative law $= (A+B) = (B+A)$.
(II) Associative law $= (A+B+C) = (A+B)+C = B+(A+C)$.

- **NAND Gate:**



(NAND gate symbol)

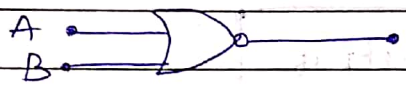
Truth table of NAND gate –

input		output
A	B	$Y = \overline{AB}$
0	0	1
0	1	1
1	0	1
1	1	0

properties of NAND gate –

(1) commutative law : $\overline{AB} = \overline{BA}$.
(2) Associative law : $\overline{ABC} \neq \overline{ABC}$.

- NOR gate –



$Y = \overline{A+B}$

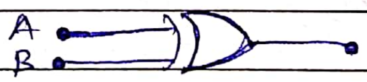
(NOR gate symbol)

Truth table of NOR gate –

Input		output
A	B	$Y = \overline{A+B}$
0	0	1
0	1	0
1	0	0
1	1	0

→ NOR Gate follows commutative law but not follow associative law.

- EXOR Gate :



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

(Ex-OR gate symbol)

Truth table of Ex-OR Gate –

Input		output
A	B	$Y = A \oplus B = \bar{A}B + A\bar{B}$
0	0	0
0	1	1
1	0	1
1	1	0

properties of Ex-OR Gate –

- Enable Input = 0
- Disable input = 1
- It is called stair case switch.
- When both input are different, then output become high or logic '1'.
- When both Input are same, then output become low or logic '0'.
- Arithmetic gate. • Inequality detector.

Note: $A \oplus A = 0$

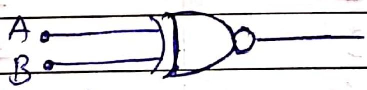
$A \oplus \bar{A} = 1$

$A \oplus 0 = A$

$A \oplus 1 = \bar{A}$.

→ follow Associative and Commutative both law.

- **Ex-NOR Gate:**



$Y = A \odot B = AB + \bar{A}\,\bar{B}$

(Ex-NOR gate symbol).

Truth table of Ex-NOR Gate –

Input		Output
A	B	$Y = A \odot B = AB + \bar{A}\bar{B}$.
0	0	1
0	1	0
1	0	0
1	1	1

properties of EXNOR Gate –

- Enable Input = 1
- Disable Input = 0
- when both the input are same, then output become high or logic '1'.
- when both the Input are diff, then output become Low or logic '0'.
- equality detector.

- NAND and NOR Gate universal gate –

logic gate	Required no. of NAND Gate	Required no. of NOR Gate.
NOT	1	1
AND	2	3
OR	3	2
EXOR	4	5
EXNOR	5	4

• Implementation of NAND Gate –

procedure to Implement Boolean functions –

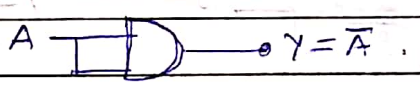
- → **Step-1:** Take complement of given function & apply DeMorgan's theorem.
- → **Step-2:** Take one more time complement to get original function.
  And implement using NAND gates.

• NAND Gate as universal gate –

(i) **NOT gate:**

$$Y = \overline{A}$$

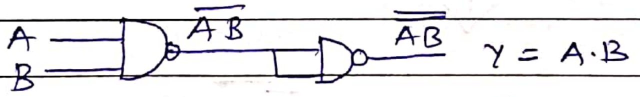
$$\text{NAND}, \ Y = \overline{A \cdot B} = \overline{A \cdot A} = \overline{A}$$



(ii) **AND gate:**

$$Y = A \cdot B$$

$$\text{NAND}, \ Y = \overline{A \cdot B}$$

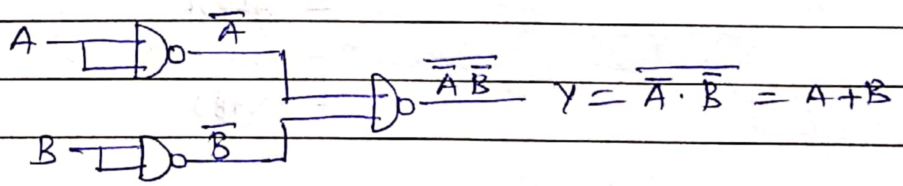


(iii) **OR gate:**

$$Y = A + B$$

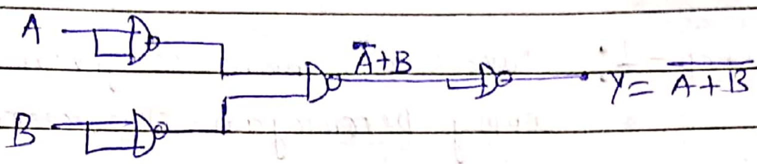
$$\text{Step-I}, \ \overline{Y} = \overline{A+B} = \overline{A} \cdot \overline{B}$$

$$\text{Step-II} \ \overline{\overline{Y}} = \overline{\overline{A} \cdot \overline{B}}$$



(iv) NOR gate:

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



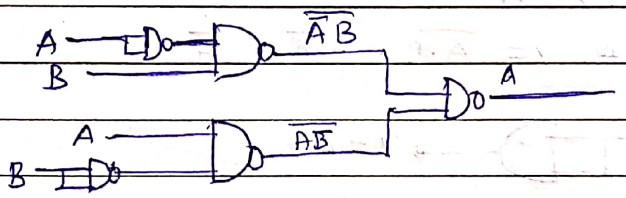
(v) EX-OR gate:

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

S-1: $\overline{Y} = \overline{\overline{A}B + A\overline{B}}$

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

S-2: $\overline{\overline{Y}} = \overline{\overline{\overline{A}B} \cdot \overline{A\overline{B}}}$

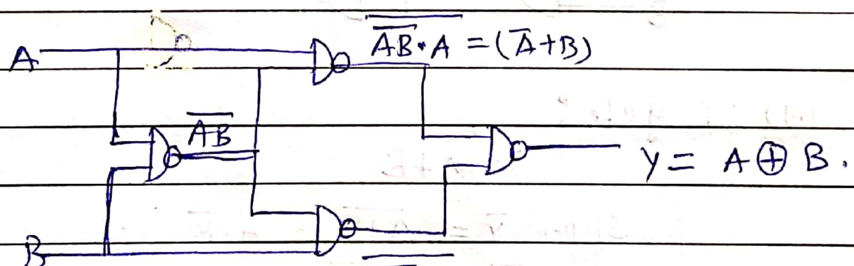


$Y = \overline{\overline{\overline{A}B} \cdot \overline{A\overline{B}}}$

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

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

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



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

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

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

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

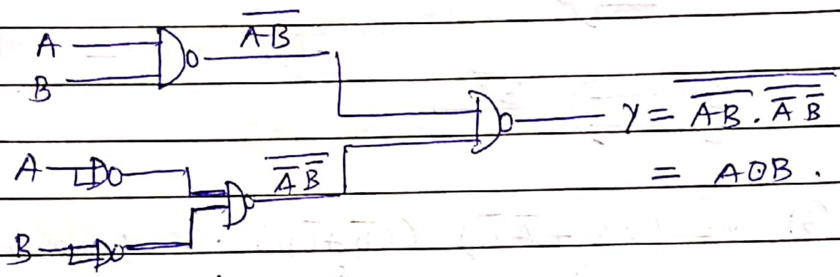
(VI) Ex-NOR gate :

$$Y = A \odot B$$
$$= AB + \bar{A}\bar{B}$$

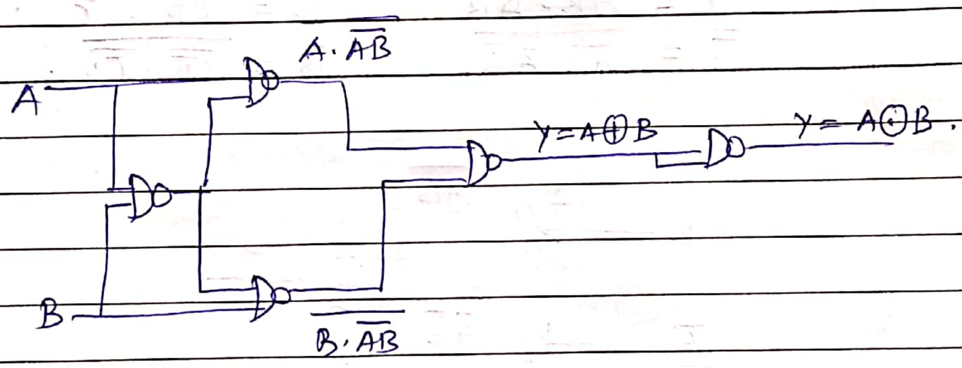
Step-1 : $\bar{Y} = \overline{AB + \bar{A}\bar{B}}$

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

Step-II : $\bar{\bar{Y}} = \overline{\overline{AB} \cdot \overline{\bar{A}\bar{B}}}$.



$\Downarrow$



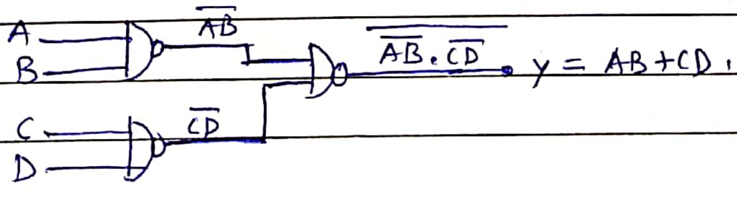
5 NAND gate required to implement one EX-NOR gate.

- Implementation of Boolean functions using only NAND gates –

(i) $Y = AB + CD$

Step-1 : $\bar{Y} = \overline{AB + CD}$
$= \overline{AB} \cdot \overline{CD}$

Step-2 : $\bar{\bar{Y}} = \overline{\overline{AB} \cdot \overline{CD}}$.



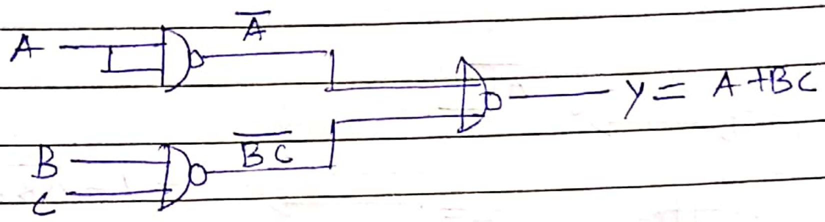
(2) $Y = A + BC$

→

S-1: $\overline{Y} = \overline{A+BC}$

$= \overline{A} \cdot \overline{BC}$

S-2: $\overline{\overline{Y}} = \overline{\overline{A} \cdot \overline{BC}}$



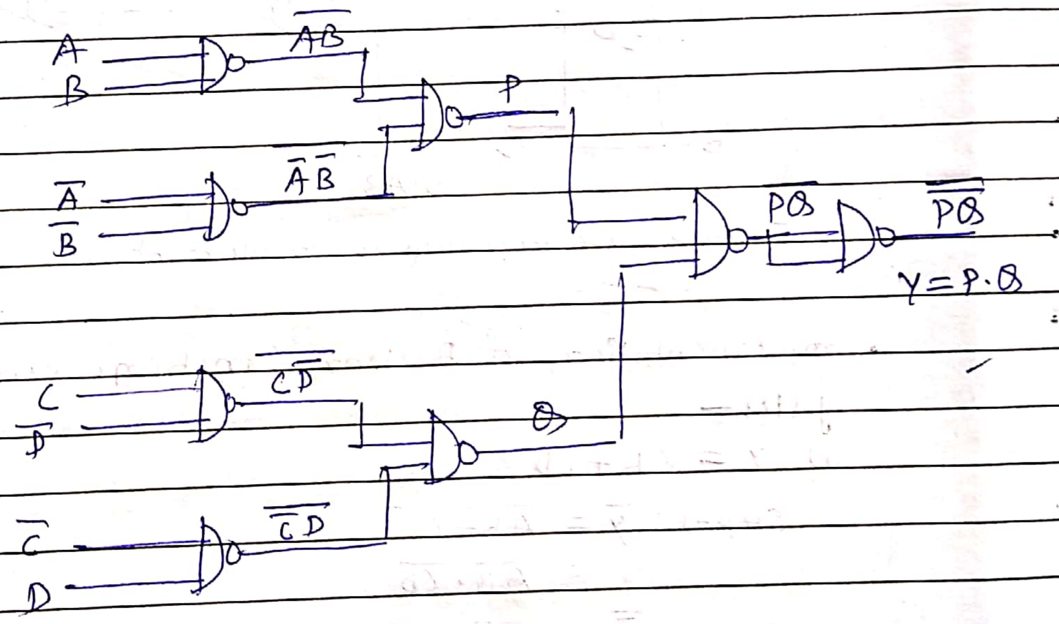
(3) $Y = \underbrace{(AB + \overline{A}\,\overline{B})}_{P}\ \underbrace{(C\overline{D} + \overline{C}D)}_{Q}$ · (complement are available)

→

$P = AB + \overline{A}\,\overline{B}$ $\qquad Q = C\overline{D} + \overline{C}D$

$\overline{P} = \overline{AB} \cdot \overline{\overline{A}\,\overline{B}}$ $\qquad \overline{Q} = \overline{C\overline{D}} \cdot \overline{\overline{C}D}$

$\overline{\overline{P}} = \overline{\overline{AB}\ \overline{\overline{A}\,\overline{B}}}$ $\qquad \overline{\overline{Q}} = \overline{\overline{C\overline{D}} \cdot \overline{\overline{C}D}}$



(4) $Y = (A+B)\,(C+D)$

(i) using only two I/p NAND gates.

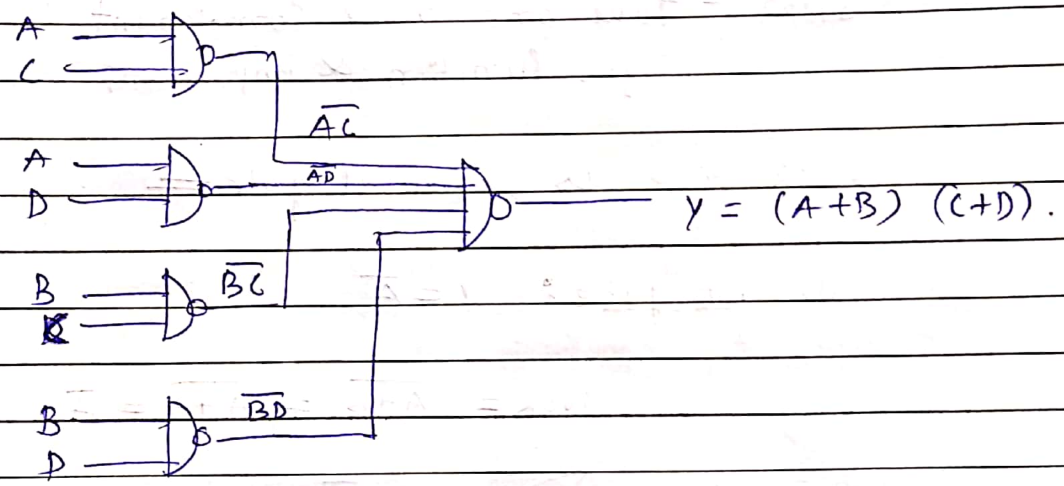
(ii) using only NAND gates.

→

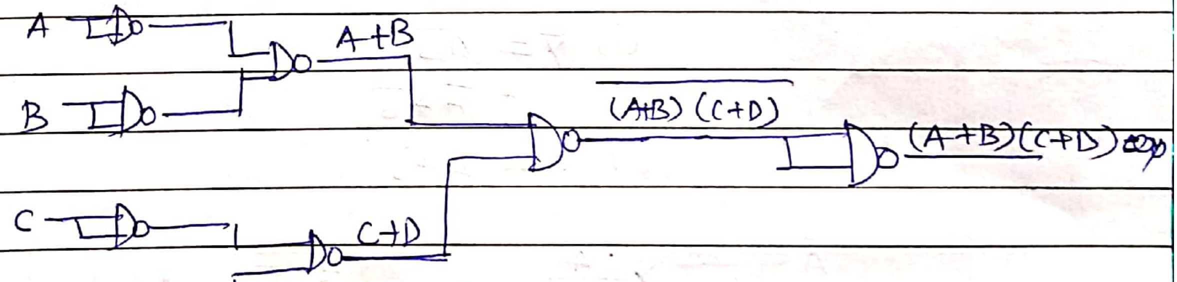
(ii) $Y = AC + AD + BC + BD$

S-1: $\overline{Y} = \overline{AC}\;\overline{AD}\;\overline{BC}\;\overline{BD}$

S-2: $\overline{\overline{Y}} = \overline{\overline{AC}\;\overline{AD}\;\overline{BC}\;\overline{BD}}$



(i) $y = (A+B)\;(C+D)$



(8 NAND gate required).

- ## Implementation of NOR Gate :

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

Procedure :-

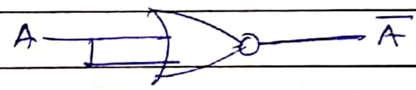
Step1:- Apply complement for the given logic function and apply De-morgan's Theorem.
(If it is not in the required form)

Step2 :- Take one more complement to get original logic function & implement using NOR gates.

- NOR Gate as universal gate :

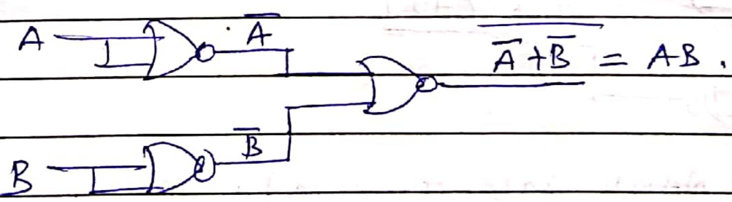
(i) NOT gate : $Y = \overline{A}$

$$NOR = \overline{A+B} = \overline{A+A} = \overline{A}$$

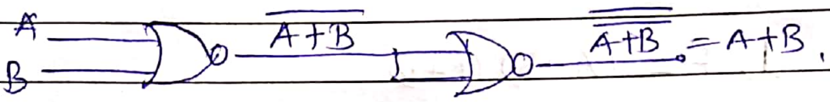


(ii) AND gate : $Y = A \cdot B$

(i) $\overline{Y} = \overline{AB}$
$= \overline{A} + \overline{B}$
(ii) $\overline{\overline{Y}} = \overline{\overline{A}+\overline{B}}$

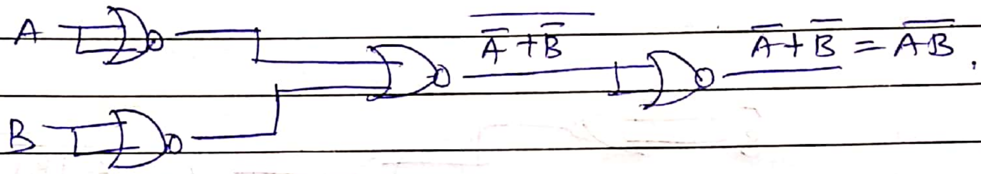


(iii) OR gate: $Y = A + B$



(iv) NAND gate: $Y = \overline{AB}$

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



(V) EX-NOR: $Y = A \odot B = \underbrace{AB}_{P} + \underbrace{\overline{A}\overline{B}}_{Q}$

$P = AB$

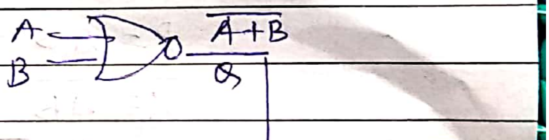
(i) $\overline{P} = \overline{AB}$

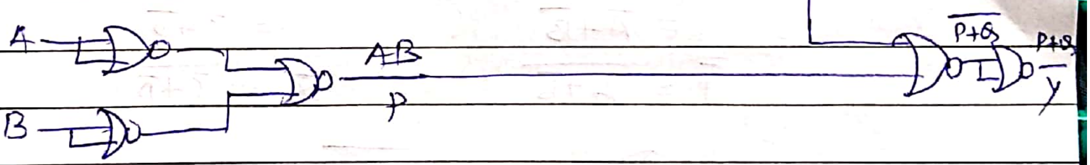
$= \overline{A} + \overline{B}$

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

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

$= \overline{A+B}$

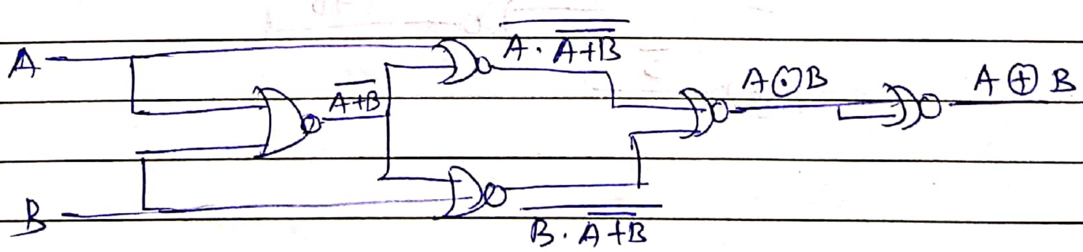




minimum no. of NOR gate require EX-NOR gate is '4'.

(VI) EX-OR gate: $Y = A \oplus B$

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



'5' min no. of NOR required to implement EX-OR gate.

- Implementation of Boolean function using only NOR gates –

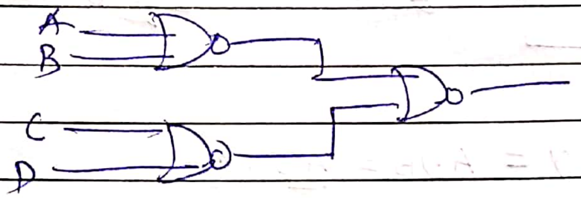
(1) $F = (A+B)(C+D)$

$\rightarrow$

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

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

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



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

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

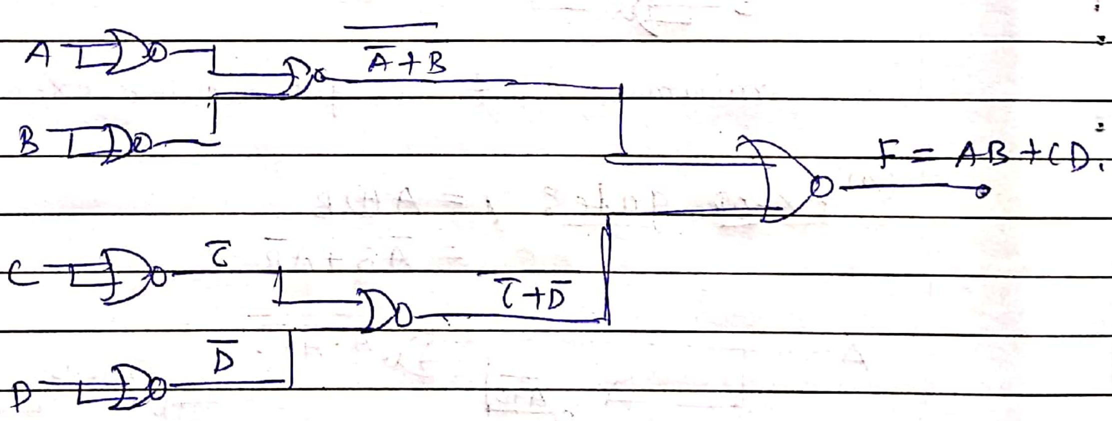
(2) $F = \underbrace{AB}_{P} + \underbrace{CD}_{Q}$.

$P = AB$ $\qquad$ $Q = CD$

$\overline{P} = \overline{AB}$ $\qquad$ $\overline{Q} = \overline{CD}$

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

$\overline{\overline{P}} = \overline{\overline{A}+\overline{B}}$ $\qquad$ $\overline{\overline{Q}} = \overline{\overline{C}+\overline{D}}$



③ $F = \underbrace{a(b+cd)}_{X} + \underbrace{b\bar{c}}_{Y}$

$b\bar{c}$

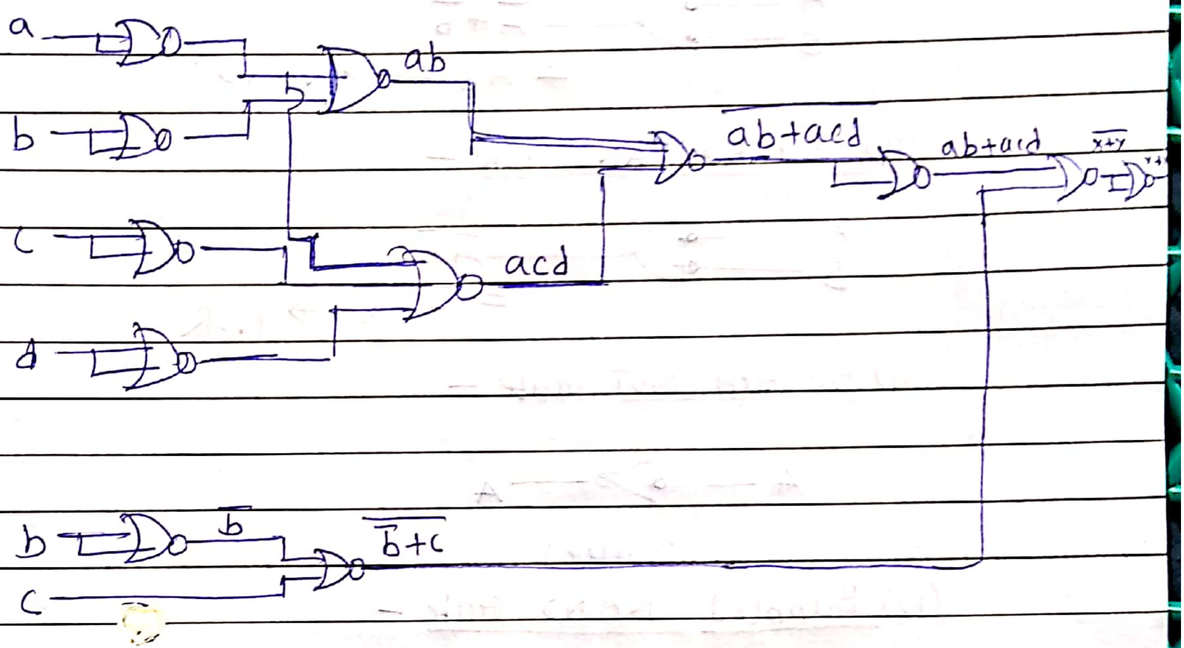
$\bar{b}+c$

$\overline{\bar{b}+c}$

→

$X = a(b+cd)$

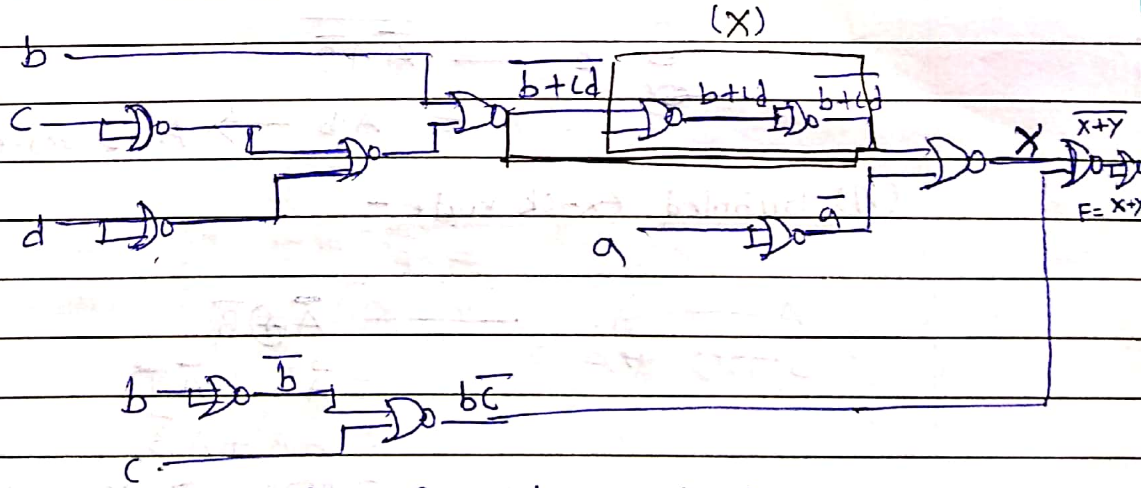
$= ab + acd$



④ Implement the logic function using only two input NOR gates —

$F = \underbrace{a(b+cd)}_{X} + \underbrace{b\bar{c}}_{Y}$

$X = a(\underline{b+cd})$

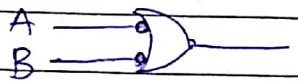


(10 NOR gate required)

• Alternate logic gate (Bubbled gate):

"Bubble indicates Not operation".

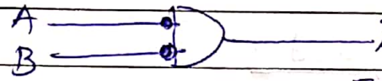
(i) Bubbled OR Gate –



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

$= \overline{AB} \longrightarrow$ NAND

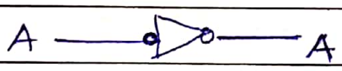
(ii) Bubbled AND Gate –



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

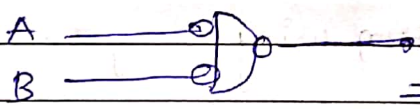
$= \overline{(A+B)} \longrightarrow$ NOR.

(iii) Bubbled NOT Gate –



(Buffer)

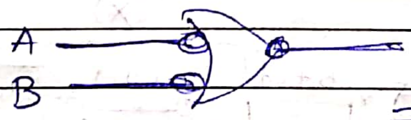
(iv) Bubbled NAND Gate –



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

$= \overline{\overline{(A+B)}} = A+B \Rightarrow$ OR Gate.

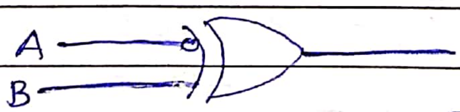
(v) Bubbled NOR Gate –



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

$= A\cdot B \longrightarrow$ AND Gate.

(vi) Bubbled ex-OR Gate –



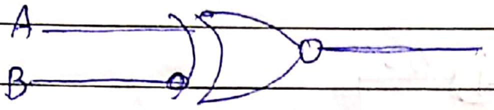
$\overline{A} \oplus B$

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

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

$= A \odot B \longrightarrow$ X-NOR gate.

(VII) Bubbled X-NOR gate —



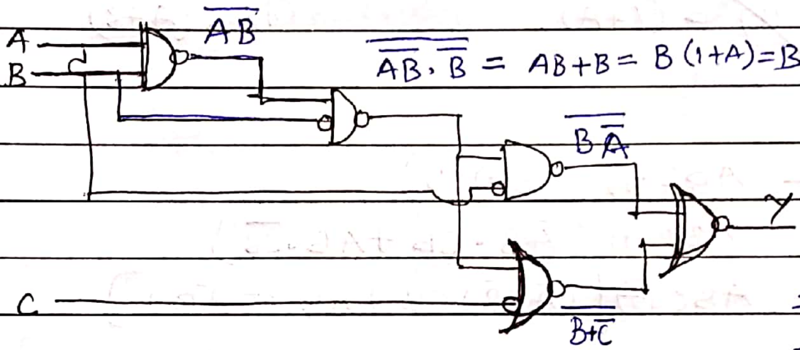
$A \odot \bar B$

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

$= A\bar B + \bar A B$. → X-OR gate.

gate - 2000

$\boxed{Q-1}$ The simplified Boolean expression for the O/P Y is,



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

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

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

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

$= B\bar A + B\bar A\bar C$

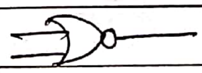
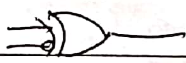
$= B\bar A(1+\bar C)$

$= \bar A B$.

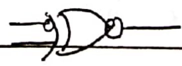
gate - 2010

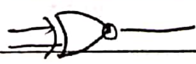
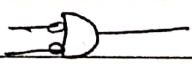
$\boxed{Q-2}$ Match the logic gates in column 'A' with their equivalents in column B.

Column A | Column B

P.  $\overline{A+B}$ (NOR ~~NAND~~) | 1.  $A\oplus\bar B = \bar A\bar B + AB$ (EXNOR)

Q.  $\overline{AB}$ (NAND) | 2.  $\bar A + \bar B = \overline{AB}$ (NAND)

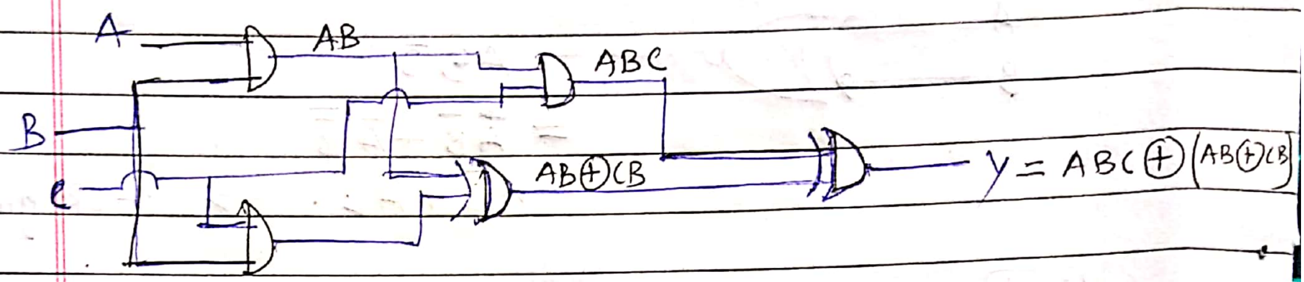
R.  $A\oplus B$ (EXOR) | 3.  $\bar A\odot B = A\bar B + \bar A B$ (EXOR)

S.  $A\odot B$ (EXNOR) | 4.  $\bar A\bar B = \overline{A+B}$ = (NOR)

$\boxed{P\to 4,\ Q\to 2,\ R\to 3,\ S\to 1}$

gate - 2016

$\boxed{Q-3}$ The Output of the combinational circuit given below is:



(a) $A+B+C$ (b) $A(B+C)$

(c) $B(C+A)$ ✓ (d) $C(A+B)$

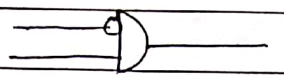
→

$$Y = ABC \oplus (AB \oplus CB)$$
$$= ABC \oplus (\overline{AB} \cdot CB + AB \cdot \overline{CB})$$
$$= ABC \oplus [(\bar{A}+\bar{B})CB + AB(\bar{C}+\bar{B})]$$
$$= ABC \oplus (\bar{A}CB + AB\bar{C})$$
$$= \overline{ABC}(\bar{A}CB + AB\bar{C}) + ABC\,(\overline{\bar{A}CB + AB\bar{C}})$$
$$= (\bar{A}+\bar{B}+\bar{C})(\bar{A}CB + AB\bar{C}) + ABC\,(A+\bar{C}+\bar{B}) \cdot (\bar{A}+\bar{B}+C)$$
$$= \bar{A}CB + AB\bar{C} + (A+\bar{C}+\bar{B})\,ABC$$
$$= \bar{A}BC + AB\bar{C} + ABC$$
$$= BC + AB\bar{C}$$
$$= B(C + A\bar{C}) = B(C+A)$$

Page 32

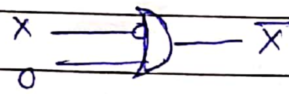
Gate-2015

$\boxed{Q-4}$ Check given gate are universal or not –

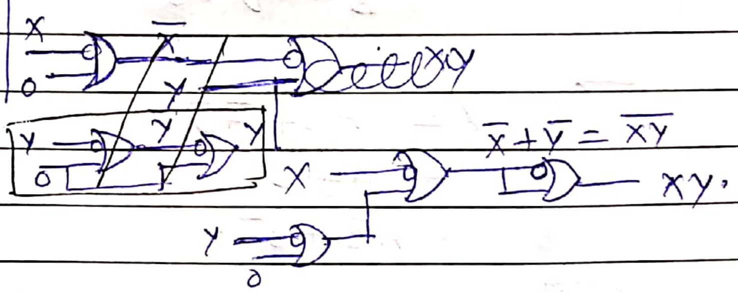


→ If a gate are universal, then we can Implement NOT, AND, OR, NAND, NOR.

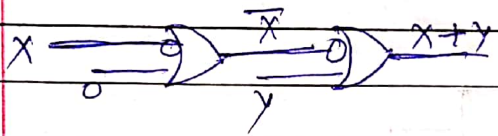
✓ NOT



✓ AND



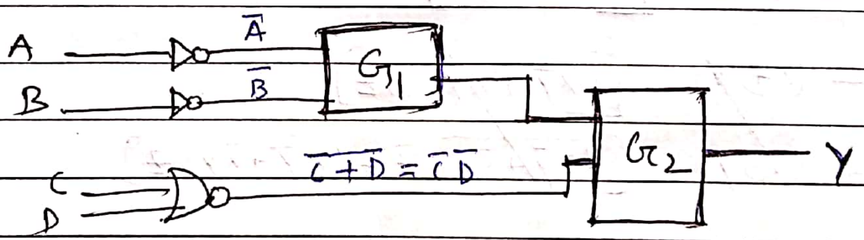
✓ OR



so, given gate are universal gate.

gate-15

$\boxed{Q-5}$ In the figer shown, the O/P y is required to be $y = AB + \overline{C}\overline{D}$. The gate G_1 and G_2 must be –



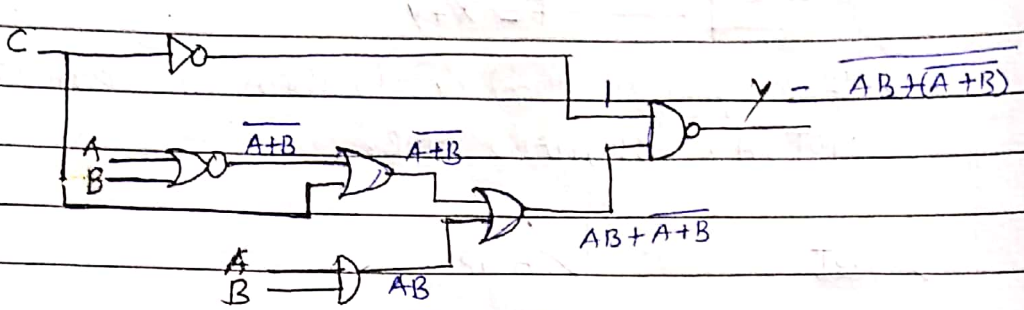
(a) NOR, OR (b) OR, NAND (c) NAND, OR (d) AND, NAND

→

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

Gate-14

$\boxed{Q-6}$ If $C=0$, $Y=?$

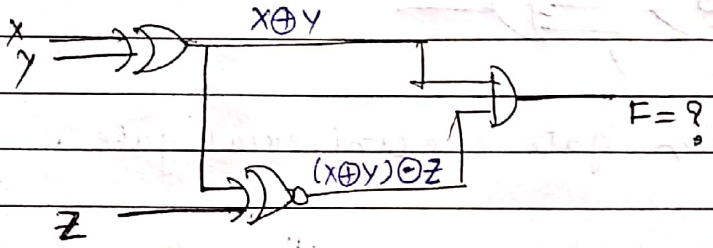


→ $Y = \overline{AB + \bar{A}\bar{B}}$

$= A \oplus B$

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

Gate-14 $\boxed{Q-7}$



→ $F = (X\oplus Y)\,\boxed{(X\oplus Y)\odot Z}$

$= (X\oplus Y)\,[(X\bar{Y} + \bar{X}Y)\odot Z]$

$= (X\oplus Y)\,[\overline{(X\bar{Y} + \bar{X}Y)}\,\bar{Z} + (X\bar{Y} + \bar{X}Y)\,Z]$

$= (X\oplus Y)\,[(X\oplus Y)\,Z + \overline{(X\oplus Y)}\cdot\bar{Z}]$

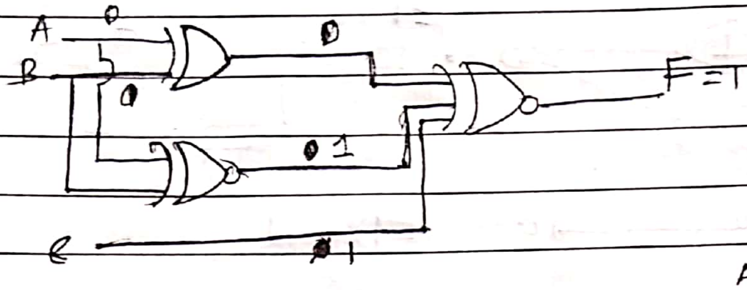
$= (X\oplus Y)\,Z + 0$

$= (\bar{X}Y + X\bar{Y})\,Z$

$= \bar{X}YZ + X\bar{Y}Z$

atlantis
Date
Page 34

Gate-10 [Q-8] for the output F to be '1' in the logic circuit shown, the i/p combination should be



A	B	C	F
0	0	0	0
1	1	0	0

011 = 0→1

× (a) A=1, B=1, C=0

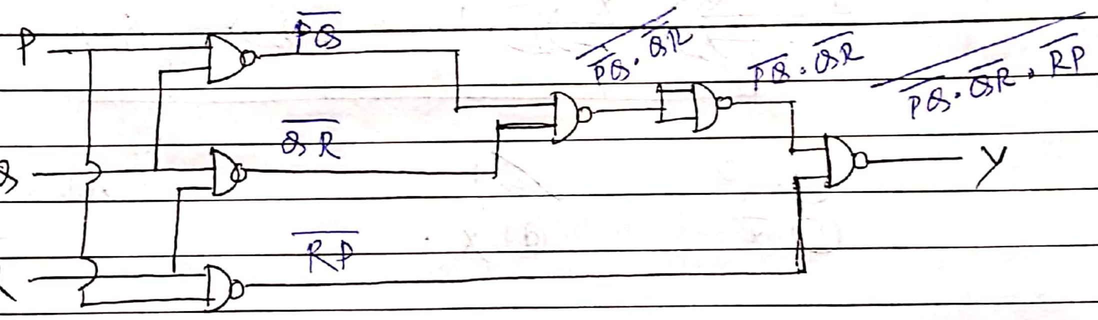
× (b) A=1, B=0, C=0

× (c) A=0, B=1, C=0

✓ (d) A=0, B=0, C=1.

010=1
100=1=0

G-11 [Q-9] The o/p y in the circuit below is always '1' when,



× (a) two or more of the i/ps P, Q, R are '0'.

✓ (b) two or more " " " are '1'.

(c) any odd no. of the i/ps P, Q, R is '0'.

✓ (d) " " " " is '1'.

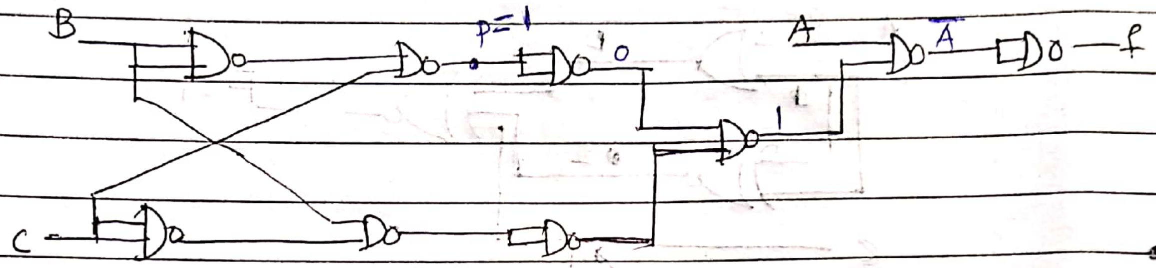
→

$y = \overline{\overline{PQ}\ \overline{QR}\ \overline{PR}}$

$P, Q = 1$

$\boxed{y = PQ + QR + PR}$

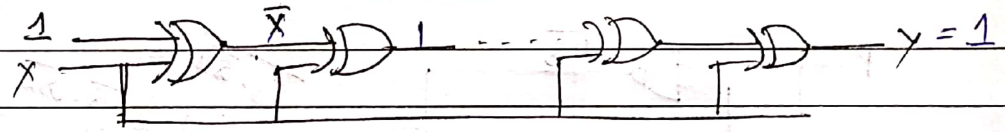
G-6 $\boxed{Q-10}$ The point p in the following fig is stuck at 1, the o/p f will be –



ⓐ $\overline{AB\bar{C}}$ ⓑ $\bar{A}$

ⓒ $AB\bar{C}$ ⓓ A ✓

G-2 $\boxed{Q-11}$ If the I/p to the dig ckt consisting of a cascade of 20 X-OR gates is X. Then the O/p Y is equal to –



ⓐ 0 ⓑ 1 ✓

ⓒ $\bar{X}$ ⓓ X.

$A \oplus B = \bar{A}B + A\bar{B}$
$= 0.x + 1\bar{x}$
$= \bar{x}$

$\bar{x} \oplus x = x.x + \bar{x}\bar{x}$
$= x + \bar{x}$
$= 1$

MINIMIZATION
Rakesh Nama

Representation of Boolean Expressions —

- Canonical form
  - SOP — Sum of product or Sum of midterm
  - POS — product of sum or product of maxterms
- Standard form
  - S SOP → Ex = $F = A + BC$.
  - S POS → $F = (A+B)(\bar{A}+B+C)$

Ex:

SOP	POS
$F(A,B) = AB + \bar{A}\bar{B}$	$F = (A+B)(\bar{A}+\bar{B})$
$F(A,B,C) = A\bar{B}C + \bar{A}\bar{B}C$	$F = (A+\bar{B}+C)(\bar{A}+\bar{B}+C)$
$F(A,B,C) = \Sigma(1,2,3,7)$	$F = \pi(0,2,5)$

- **Minterm** — $0 \rightarrow \bar{A}$, $1 \rightarrow A$

$n = 2$

	A	B	Minterm —
0	0	0	$\bar{A}\,\bar{B}\ (m_0)$ → minterm or standard product.
	0	1	$\bar{A}\,B\ (m_1)$
	1	0	$A\,\bar{B}\ (m_2)$
$2^n - 1$	1	1	$A\,B\ (m_3)$

m_0 to m_3 → 2^n minterms.

- **Maxterm** — $0 \rightarrow A$, $1 \rightarrow \bar{A}$

A	B	Maxterm
0	0	$A+B\ (M_0)$ → maxterm or standard sum
0	1	$A+\bar{B}\ (M_1)$
1	0	$\bar{A}+B\ (M_2)$
1	1	$\bar{A}+\bar{B}\ (M_3)$

• Representation of Minterm and Maxterm –

	A	B	C	Minterm	Maxterm
0	0	0	0	$\bar{A}\bar{B}\bar{C}$ (m_0)	$A+B+C$ (M_0)
1	0	0	1	$\bar{A}\bar{B}C$ (m_1)	$A+B+\bar{C}$ (M_1)
2	0	1	0	$\bar{A}B\bar{C}$ (m_2)	$A+\bar{B}+C$ (M_2)
3	0	1	1	$\bar{A}BC$ (m_3)	$A+\bar{B}+\bar{C}$ (M_3)
4	1	0	0	$A\bar{B}\bar{C}$ (m_4)	$\bar{A}+B+C$ (M_4)
5	1	0	1	$A\bar{B}C$ (m_5)	$\bar{A}+B+\bar{C}$ (M_5)
6	1	1	0	$AB\bar{C}$ (m_6)	$\bar{A}+\bar{B}+C$ (M_6)
7	1	1	1	ABC (m_7)	$\bar{A}+\bar{B}+\bar{C}$ (M_7)

Example –

	A	B	C	Y
0	0	0	0	0
1	0	0	1	1 ✓
2	0	1	0	1 ✓
3	0	1	1	0
4	1	0	0	1 ✓
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

SOP (1's)

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

$$Y = \Sigma(1, 2, 4)$$

POS (0's)

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

$$= \Pi(0, 3, 5, 6, 7).$$

SOP (sum of product)	POS (product of sum)
$F = \overset{000}{m_0} + \overset{001}{m_1} + \overset{011}{m_3}$	$F = \overset{(001)}{M_1} \cdot \overset{(011)}{M_3} \cdot \overset{(100)}{M_4}$
$= \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + \bar{A}BC$	$= (A+B+\bar{C})(A+\bar{B}+\bar{C})(\bar{A}+B+C)$
$= \Sigma(0,1,3)$.	$= \pi(1,3,4)$.

$F = \overset{SSOP}{A+B\bar{C}} = \overset{SOP}{A(B+\bar{B})(C+\bar{C}) + BC(A+\bar{A})}$

$F = \overset{SPOS}{(A+B)} \; (A+\bar{B}+C)$

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

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

- Standard forms

SSOP	SPOS
$Y = A + B\bar{C}$	$Y = (A+B)(A+\bar{B}+C)$

Ex – Simplify the following Boolean functions T_1 & T_2 to a min no. of literals.

A	B	C	T_1	T_2
0	0	0	1	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	0	1
1	0	1	0	1
1	1	0	0	1
1	1	1	0	1

8 4 2 1

→ $T_1 = m_0 + m_1 + m_2$

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

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

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

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

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

$T_2 = m_3 + m_4 + m_5 + m_6 + m_7$

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

= [illegible]

= [illegible]

= [illegible] $(A+B)$

= [illegible]

⇒ [illegible]

⇒ [illegible]

⇒ [illegible]

4 21

$T_2 = m_3 + m_4 + m_5 + m_6 + m_7$

$\Rightarrow \bar{A}BC + A\bar{B}\bar{C} + A\bar{B}C + AB\bar{C} + ABC$

$\Rightarrow BC(A+\bar{A}) + A\bar{B}(\bar{C}+C) + AB\bar{C}$

$\Rightarrow BC + A\bar{B} + AB\bar{C}$

$\Rightarrow \cancel{BC(1+A) + A\bar{B}}$ $\Rightarrow B(C + A\bar{C}) + A\bar{B}$

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

$\Rightarrow \cancel{BC(\bar{B}+A\bar{B})(C+A\bar{B})}$ $\Rightarrow AB + BC + A\bar{B}$

$\Rightarrow \cancel{(A+B)(C+A\bar{B})}$ $\Rightarrow A + BC$.

Ex- convert each of the following to the other canonical form -

(a) $F(x, y, z) = \Sigma(2,4,5,6)$

(b) $F(A,B,C,D) = \pi(0,1,2,4,7,9,12)$

$\rightarrow$

(a) $F(x,y,z) = \Sigma(2,4,5,6)$

$\hookrightarrow F(x,y,z) = \pi(0,1,3,7)$

A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

$\rightarrow$ (b) $F(A,B,C,D) = \pi(0,1,2,4,7,9,12)$

$\hookrightarrow F(A,B,C,D) = \Sigma(3,5,6,8,10,11,13,14,15)$

Q. Express the following function as a sum of minterms & as a product of maxterms –

$F(A,B,C,D) = \bar{B}D + \bar{A}D + BD.$

$\rightarrow$

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

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

$\cancel{= D(1+\bar{A})}$

$\cancel{= D}$

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

$= AC\bar{B}D + A\bar{C}\bar{B}D + \bar{A}C\bar{B}D + \bar{A}\bar{C}\bar{B}D$
$+ \bar{A}BCD + \bar{A}B\bar{C}D + \bar{A}\bar{B}CD + \bar{A}\bar{B}\bar{C}D$
$+ BDCA + BDC\bar{A} + BD\bar{C}A + BD\bar{C}\bar{A}.$

$= AC\bar{B}D + A\bar{C}\bar{B}D + \bar{A}\bar{B}CD + \bar{A}\bar{B}\bar{C}D + \bar{A}BCD$
$+ \bar{A}B\bar{C}D + \cancel{\bar{A}B\bar{C}D} + ABCD + AB\bar{C}D.$

8 4 2 1

1 1 0 0

$= A\bar{B}CD + A\bar{B}\bar{C}D + \bar{A}\bar{B}CD + \bar{A}\bar{B}\bar{C}D + \bar{A}BCD + \bar{A}B\bar{C}D$
$\cancel{\bar{A}B\bar{C}D} + ABCD + AB\bar{C}D.$

$F(A,B,C,D) = \Sigma(11, 9, 3, 1, 7, 5, 15, 13)$

$F(A,B,C,D) = \Sigma(1,3,5,7,9,11,13,15)$ ✓

$F(A,B,C,D) = \pi(0,2,4,6,8,10,12,14)$

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

Q. Convert each of the following expressions into SOP and POS forms –

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

(b) $\bar{x} + x(y+\bar{y})(y+\bar{z})$

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

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

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

$= ABCD + ABC\bar{D} + AB\bar{C}D + AB\bar{C}\bar{D} + A\bar{C}DB + A\bar{C}D\bar{B} + BCAD + BCA\bar{D} + BC\bar{A}D + BC\bar{A}\bar{D}$

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

(b) $\bar{x} + x(x+\bar{y})\ (y+\bar{z})$

$F = (\bar{x}+x)\,(\bar{x}+(x+\bar{y})(y+\bar{z}))$

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

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

$= (\bar{x}+y+\bar{z})$.

4 2 1
0 1 0

$F = \pi(2)$.

Q. Obtain the truth table of following functions and express each function in sop & pos forms–

(a) $(xy+z)\ (y+xz)$ (b) $(x+\bar{y})\ (\bar{y}+z)$

$F = xy + xyz + yz + xz$

$= xy + yz + xz$

$= xy(z+\bar{z}) + yz(x+\bar{x}) + xz(y+\bar{y})$

$= xyz + xy\bar{z} + \bar{x}yz + x\bar{y}z$

x	y	z	F
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

$F(SOP) = \sum(3,5,6,7)$

$F(POS) = \pi(0,1,2,4)$

*. $\boxed{\text{With N-variables possible Boolean function} = 2^{2^n}}$

$n = 2, \quad 2^{2^n} = 2^{2^2} = 2^4 = 16$

$n = 3, \quad 2^{2^3} = 2^8 = 256.$

• ~~INTRODUCTION OF THE KARNAUGH MAP (K-MAP)~~:

• Minimization of Boolean expressions–

→ The following ~~two~~ approaches can be used for simplification of a Boolean expression–

(i) Algebric method (using Boolean algebra method)

(ii) Karnaugh map method (K-map).

• Karnaugh map ~~me~~ (K-map):

→ A K-map is a diagram made up of squares, with each square representing one minterm of the function that is to be minimized.

• Representation of K-map =

→ with n variable K-map there are 2^n cells.

• Example–

→ 2-variable K-map:

A \ B	0	1
0	0	1
1	2	3

(2-variable K-map with cell no.)

→ **3 variable K-map:**

$n=3$, no. of ~~sells~~ cells $= 2^n = 2^3 = 8$

A \ BC	00	01	11	10
0	0	1	3	2
1	4	5	7	6

(Three variable K-map with cell number)

→ **4 variable K-map:**

$n=4$, no. of cells $= 2^n = 16$

AB \ CD	00	01	11	10
00	0	1	3	2
01	4	5	7	6
10	12	13	15	14
11	8	9	11	10

(Four - variable K-map with cell no.)

examples-:

$\boxed{Ex-1}$

WX \ YZ	00	01	11	10
00	1	1	0	1
01	1	0	0	1
11	1	0	0	1
10	1	1	1	1

$F = \bar{Y}\bar{Z} + Y\bar{Z} + W\bar{X} + \bar{W}\bar{X}(\bar{Y}\bar{Z} + \bar{Y}Z)$

$= \bar{Z} + W\bar{X} + \bar{W}\bar{X}\bar{Y}$

Ex-2

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

$F = \bar{C}\bar{A}B + CAB + \bar{A}CD + \bar{C}DA + BD$.

X not need

Q-1

$F(A,B,C,D) = \sum(0,1,2,3,4,5,8,9,10,11,14,15)$

AB \ CD	00	01	11	10
00	1_{0}	1_{1}	0_{3}	1_{2}
01	1_{4}	1_{5}	0_{7}	$_{6}$
11	0_{12}	0_{13}	1_{15}	1_{14}
10	1_{8}	1_{9}	1_{11}	1_{10}

$F = A\bar{B} + \bar{A}\bar{C} + AC + \bar{B}\bar{B}$

Q-2 $F = \bar{A}\bar{B}\bar{C} + \bar{B}C\bar{D} + \bar{A}BC\bar{D} + A\bar{B}\bar{C}$

$= \overset{1}{\bar{A}\bar{B}\bar{C}D} + \overset{0}{\bar{A}\bar{B}\bar{C}\bar{D}} + \overset{10}{A\bar{B}C\bar{D}} + \overset{2}{\bar{A}\bar{B}C\bar{D}} + \overset{6}{\bar{A}BC\bar{D}} + \overset{9}{A\bar{B}\bar{C}D}$

$+ \overset{8}{A\bar{B}\bar{C}\bar{D}}$

8 4 2 1

AB \ CD	00	01	11	10
00	1_{0}	1_{1}	0_{2}	1_{2}
01	0_{4}	0_{5}	0_{7}	1_{6}
11	0_{13}	0_{13}	0_{15}	0_{14}
10	1_{8}	1_{9}	0_{11}	1_{10}

$F = \bar{B}\bar{D} + C\bar{D}\bar{A} + \bar{C}\bar{B}$

~~Ex~~ $\boxed{Q\text{-}3}$

(A) Simplify the Boolean Functions –

(a) $F = \bar A\bar B\bar C\bar D + A\bar C\bar D + \bar B C\bar D + \bar A BCD + B\bar C D$

8421

$$= \underset{0}{\bar A\bar B\bar C\bar D} + \underset{12}{AB\bar C\bar D} + \underset{8}{A\bar B\bar C\bar D} + \underset{10}{A\bar B C\bar D} + \underset{2}{\bar A\bar B C\bar D} + \underset{7}{\bar A BCD} + \underset{13}{AB\bar C D} + \underset{5}{\bar A B\bar C D}$$

K-map-

AB \ CD	00	01	11	10
00	1 (0)	0 (1)	0 (3)	1 (2)
01	0 (4)	1 (5)	1 (7)	0 (6)
11	1 (12)	1 (13)	0 (15)	0 (14)
10	1 (8)	0 (9)	0 (11)	1 (10)

$F = \bar B\bar D + AB\bar C + D\bar A B$

(b) $F = x'z + w'xy' + w(x'y + xy')$

$= x'z + w'xy' + wx'y + wxy'$

w x y z			
0 0 0 1	0 1 0 0	1 0 1 0	1 1 0 0
0 0 1 1	0 1 0 1	1 0 1 1	1 1 0 1
1 0 0 1			
1 0 1 1			

k-map-

8 4 2 1

wx \ yz	00	01	11	10
00	0 (0)	1 (1)	1 (3)	0 (2)
01	1 (4)	1 (5)	0 (7)	0 (6)
11	1 (12)	1 (13)	0 (15)	0 (14)
10	0 (8)	1 (9)	1 (11)	1 (10)

$F = x\bar y + z\bar x + w\bar x y$

Q-4

Simplify the following Boolean functions to POS form—

(a) $F(w,x,y,z) = \Sigma(0,1,2,5,8,10,13)$

→ ~~K-map~~

$F(W,X,Y,Z) = \pi(3,4,6,7,9,11,12,14,15)$

K-map-

WX \ YZ	00	01	11	10
00	(0)	(1)	0 (3)	(2)
01	0 (4)	(5)	0 (7)	0 (6)
11	0 (12)	(13)	0 (15)	0 (14)
10	(8)	0 (9)	0 (11)	(10)

$F(POS) = (\bar{Y}+\bar{Z})\cdot(\bar{X}+\bar{Z})(\bar{W}+X+\bar{Z})$.

(a) $F(A,B,C,D) = \pi(1,3,5,7,12,13,14,15)$

→

AB \ CD	00	01	11	10
00	1 (0)	0 (1)	0 (3)	1 (2)
01	1 (4)	0 (5)	0 (7)	1 (6)
11	0 (12)	0 (12)	0 (15)	0 (14)
10	1 (8)	1 (9)	(11)	1 (10)

$F = (\bar{A}+\bar{B})\cdot(A+\bar{D})$ (Product of sum).

(x or d)

- $\boxed{\text{Don't care Conditions In K-map}}$:

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

$X = $ '0' or '1'.

→ When simplifying the function, we can choose to include each don't-care minterm with either the 1's or 0's, depending on which combination gives the simplest expression.

Q.

$F(w,x,y,z) = \Sigma(1,3,7,11,15)$

$d(w,x,y,z) = \Sigma(0,2,5)$

→

AB \ CD	00	01	11	10
00	X 0	1 1	1 3	X 2
01	0 4	X 5	1 7	0 6
11	0 12	0 13	1 15	0 14
10	0 8	0 9	1 11	0 10

$F = CD + \bar{A}\bar{B}$.

Prime Implicant and Essential prime implicant:

Implicants: (Number of 1's present in the K-map.)

prime Implicant: It is a product term obtained by combining the maximum possible (no. of adjacent squares in the K-map.)

Essential prime Implicant: If a minterm in a square is covered by only one prime Implicant, that prime implicant is said to be essential.

Examples—

①

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

no. Implicant = (5) (no. of 1's)

PI = 1+1+1+1 = (4) → $\bar{x}\bar{y} + \bar{y}z + xz + xy$.

EPI = $\bar{x}\bar{y} + xy$. (2)

no. NEPI = $\bar{x}\bar{y} + xy + xz$ (2)

or

$\bar{x}\bar{y} + xy + \bar{y}z$.

②

$A \backslash BC$	00	01	11	10
0	1	1	0	0
1	1	1	1	1

no. of Implicant = 6

no. of PI = 1+1 = (2) $(\bar{B}, A)$

no. of EPI = $A + \bar{B}$ = (2)

EX-2

4-variable K-(map) —

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

find,

$I = 10$. (implicants)

$PI = A\bar{B}, AD, CD, \bar{B}C, \bar{B}\bar{D} = (5)$

$EPI = AD, \bar{B}\bar{D}$

$NEPI = A\bar{B}, \bar{B}C$.

EX-4

WX \ YZ	00	01	11	10
00	1	0	0	1
01	1	0	1	1
11	1	0	1	1
10	1	0	0	1

find,

$I = 10$

$PI = \bar{Z}, XY$ (2)

$EPI = \bar{Z}, XY$ (2)

$NEPI = 0$

$F = \bar{Z} + XY$

Q-1 simplify the following Boolean functions by first finding the essential prime implicants –

(a) $F(A, B, C, D)$
$= \Sigma(1, 3, 4, 5, 10, 11, 12, 13, 14, 15)$.

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

I = 10 (implication)

PI = AB, $B\bar{C}$, AC, $\bar{A}\bar{B}D$, $\bar{A}\bar{C}D$, $\bar{B}CD$. (6).

EPI = AB, $\underline{B\bar{C}}$, $\underline{AC}$ (2)

NEPI = AB, $\bar{A}\bar{B}D$, $\bar{B}CD$

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

$\boxed{Q-2}$ The no. of essential prime implicants for the following Boolean function from the K-map.

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

I = 7

PI = 4

$\boxed{EPI = 4}$

- Two-Level & Multi-Level Implementations:-

  - Two-level Implementation of Boolean functions:-

→ If the boolean function in sum-of-product or product-of-sum form then, the function can be implemented in Two levels.

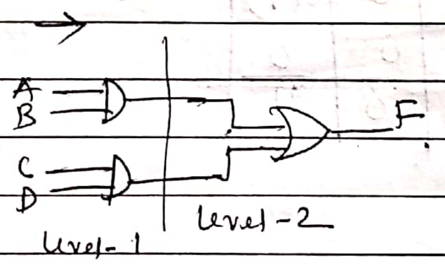
Level 1	Level 2
AND	OR
OR	AND
NAND	NAND
NOR	NOR

(Sum of products form)

EX- ① $F = AB + CD$.

ⓐ using AND, OR gates.

→

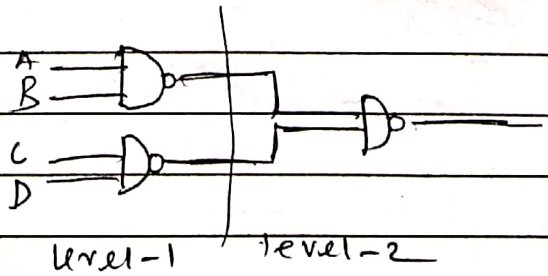


ⓑ using only NAND gates.

→ $F = AB + CD$

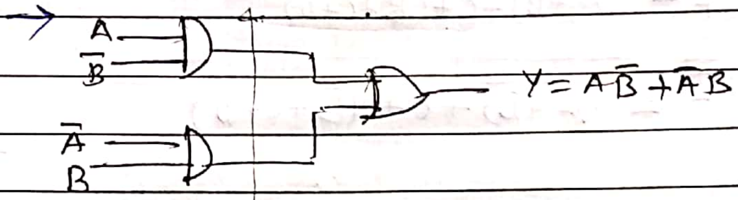
$\overline{F} = \overline{AB} \cdot \overline{CD}$

$\overline{\overline{F}} = \overline{\overline{AB} \cdot \overline{CD}}$



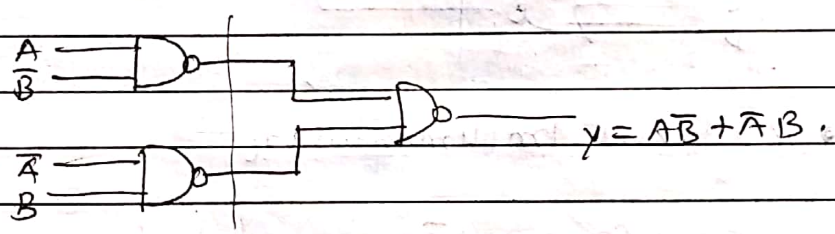
② $F = A\overline{B} + \overline{A}B$

ⓐ using AND & OR gates.



ⓑ using NAND gates only.

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

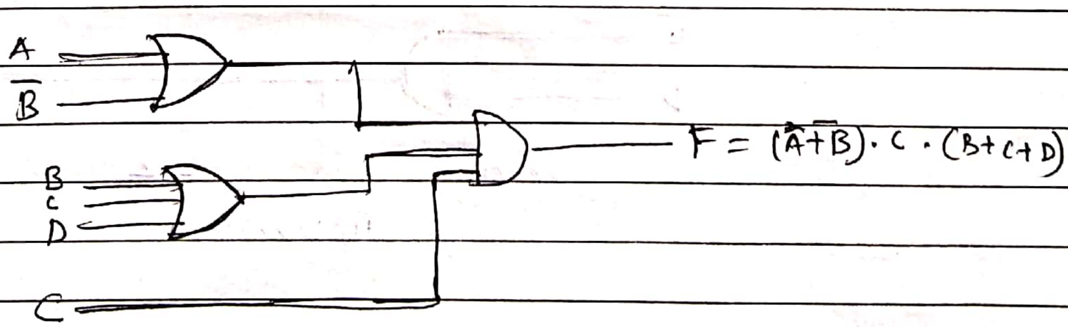


(products-of-sums form)

Ex-①

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

ⓐ using OR, AND gates —

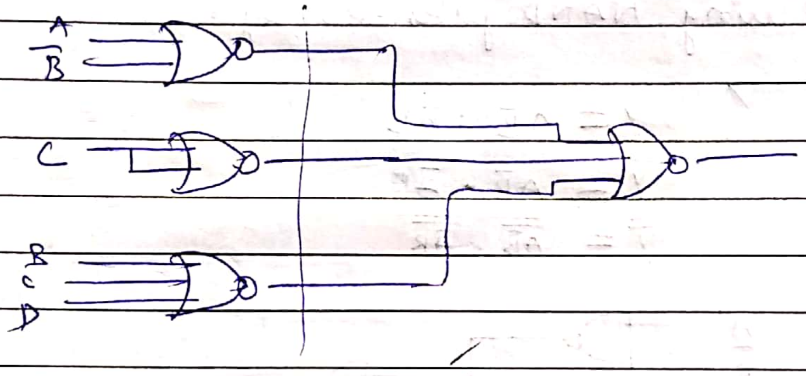


(b) using NOR gates–

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

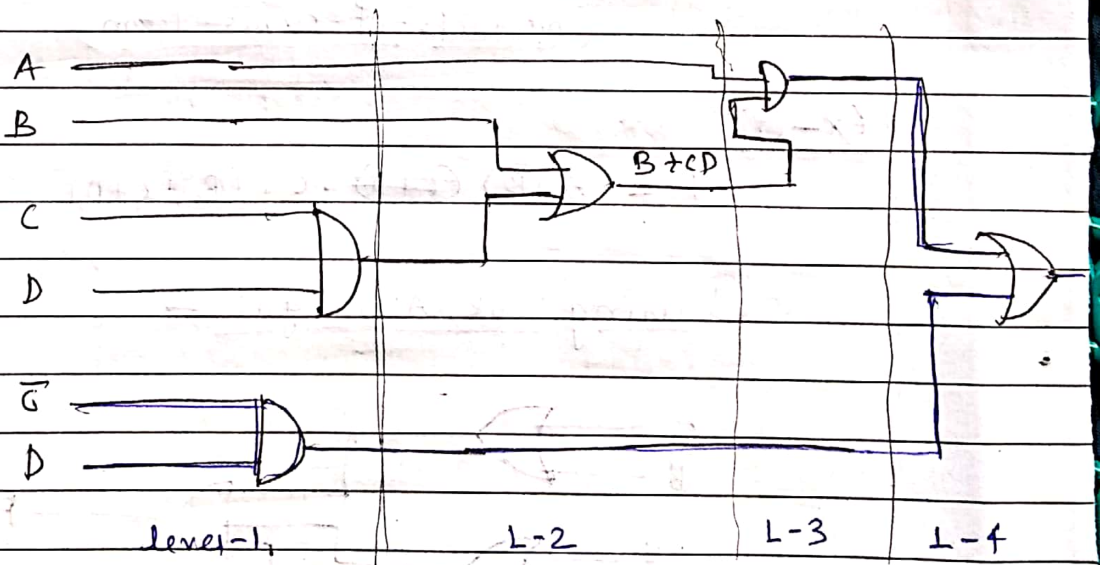
$$\bar{F} = \overline{(A+\bar{B})} + \bar{C} + \overline{(B+C+D)}$$

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



- Multilevel Impumentation –

(1) $F = A(B+CD) + \bar{C}D$.



(multi level)

(2) $F = A(B+CD) + \bar{C}D.$

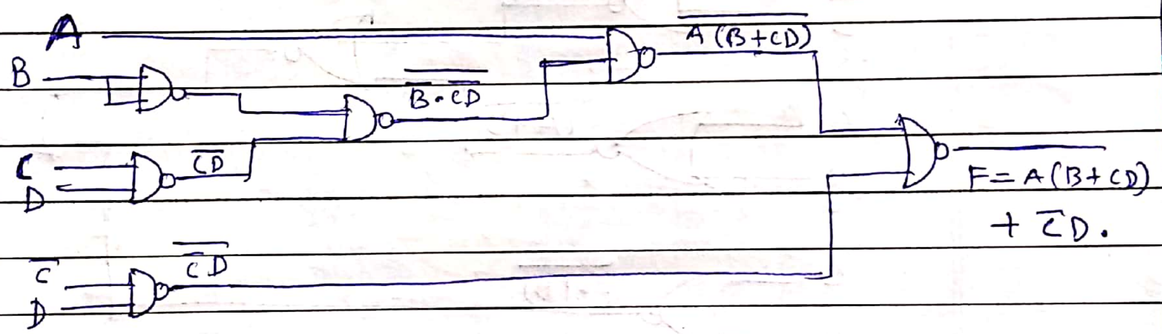
$\bar{F} = \overline{A(B+CD)} \cdot \overline{\bar{C}D}$

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

$f_1 = B + CD$

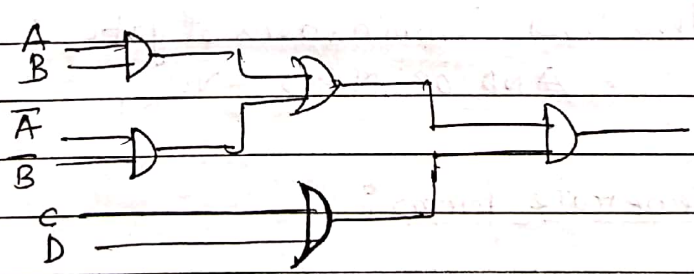
$\bar{f_1} = \bar{B} \cdot \overline{CD}$

$f_1 = \overline{\bar{B} \cdot \overline{CD}}$



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

(a) using AND, OR gates →



(b) using only NOR gate –

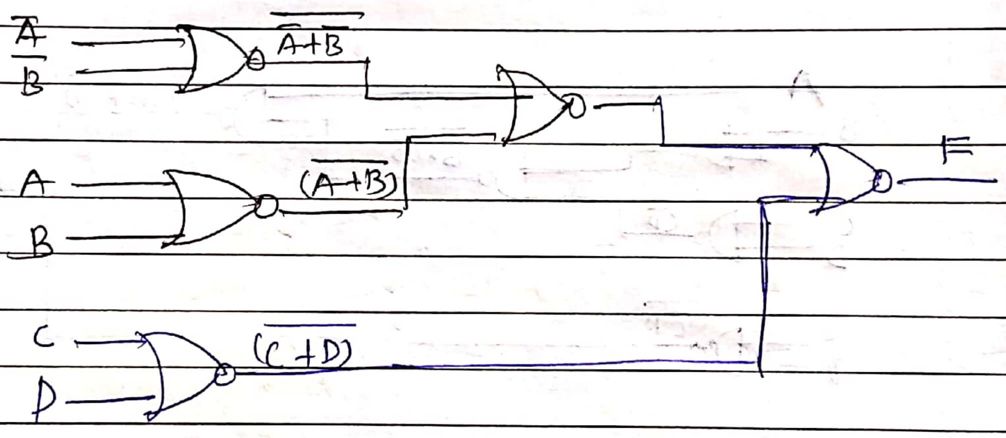
$$F = (AB + \bar{A}\bar{B})(C+D)$$

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

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

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

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



• Degenerate and Non-degenerate forms of two-level Implementation:

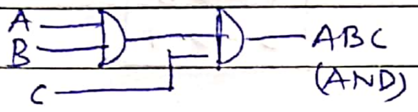
→ Two level Combinations of gates : (16-combination) (In two level)
AND, OR, NAND, NOR,

• Degenerate forms :

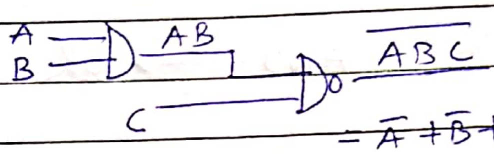
→ If two level Implementation is degenerate to single operation.
Eight combinations are degenerate forms.

AND – AND	OR – OR	NAND – OR	NOR – AND
AND – NAND	OR – NOR	NAND – NOR	NOR – NAND.

$\boxed{Ex}$ - AND - AND

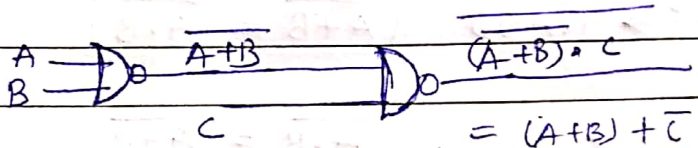


$\boxed{Ex}$ AND - NAND



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

$\boxed{Ex}$ NOR - NAND



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

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

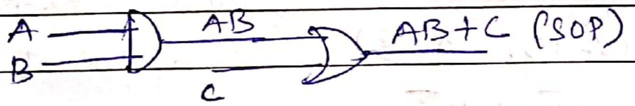
- **Non-degenerate forms :-**

→ These forms produce an Implementation in SOP ~~or~~ form or POS form.

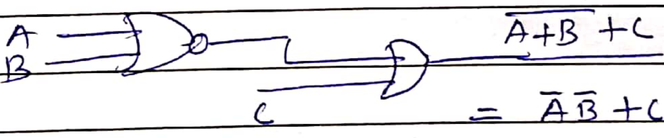
AND - OR	OR - AND	NAND - NAND	NOR - OR
AND - NOR	OR - NAND	NAND - AND	NOR - NOR

$\boxed{Ex}$ -

AND - OR



NOR - OR



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

- Duality between gates -

$\cdot \rightarrow +$	
$+ \rightarrow \cdot$	
$1 \rightarrow 0$	
$0 \rightarrow 1$	

$$AND \xrightarrow{D} OR$$

$$OR \xrightarrow{D} AND$$

$$NOR \xrightarrow{D} NAND \quad (\overline{A+B} = \overline{A \cdot B})$$

$$NAND \xrightarrow{D} NOR. \quad (\overline{A \cdot B} = \overline{A+B})$$

$$\cancel{EX\text{-}OR \xrightarrow{D} EX\text{-}NOR}$$

$$\cancel{EX\text{-}NOR \xrightarrow{D}}$$

$\boxed{EX\text{-}OR}$

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

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

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

(Dual of EX-OR gate = complement of EX-NOR)

$\boxed{EX\text{-}NOR}$

$$Y = AB + \bar{A}\bar{B}$$

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

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

(Dual of EX-NOR gate = complement of EX-OR)

Combinational circuit

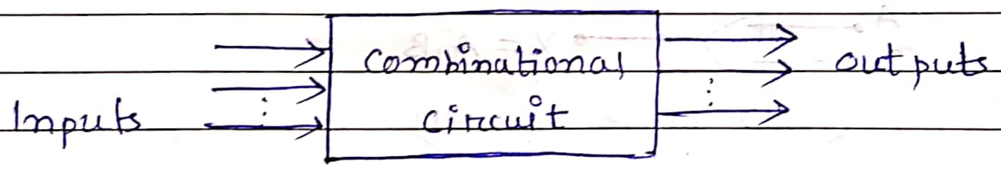
Ravesh Nama.

- Logic circuits can be devided into two types –
  - → Combinational logic circuit.
  - → sequential logic circuit.

- **Combinational logic circuit:**

A combinational logic circuit consists a logic gates whose output is determined by the combination of current inputs.

→ It consists of input variables, logic gate and output variable.
→ No feedback is required.
→ NO memory is required.



Ex- Multiplexer, Encoder, Decoder, parallel adders, etc.

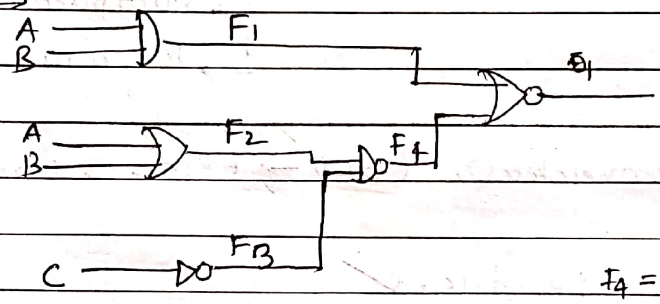
Combinational circuit

• Anlysis procedure —

→ The Analysis of combinational circuit requires that we determine the function that the circuit implements.

Logic diagram
↓
Boolean functions
↓
truth table.
↓
circuit operation.

Ex-1



$F = \overline{F_1 + F_4}$

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

$F = \overline{F_4 + F_1}$

$F_4 = \overline{F_2 \cdot F_3}$

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

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

→

A	B	C	F_1 ($A\cdot B$)	F_2 ($A+B$)	F_3 ($\bar{C}$)	F_4	F
0	0	0	0	0	1	1	0
0	0	1	0	1	0	1	0
0	1	0	0	1	1	0	1 ✓
0	1	1	0	1	0	1	0
1	0	0	0	1	1	0	1 ✓
1	0	1	0	1	0	1	0
1	1	0	0	1	1	0	1 ✓
1	1	1	1	1	0	1	0

$F = \bar{A}B\bar{C} + A\bar{B}\bar{C} + AB\bar{C}$

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

$\boxed{F = \bar{C}\,[(A \oplus B) + AB]}$. $= \bar{C}\,(A+B)$ (final Analysis procedure)

Design procedure :

Specification

↓

Truth table

↓

Boolean function

↓

Logic Diagram

Combinational circuits —

(I) Arithmetic circuits —
- → Adders ✓
- → Subtractors ✓
- → Multiplier. ✓
- → Magnitude comparator. ✓

(II) Code converters. (BCD → Excess-3 code)

(III) Encoder & Decoder. ✓

(IV) Multiplexer & Demultiplexer.

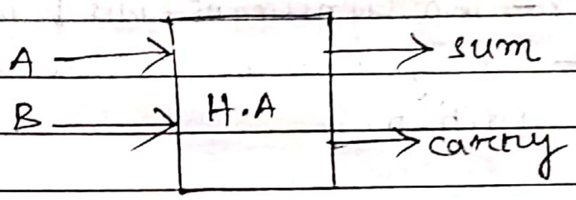
(V) programmable Logic Devices (PLDs)
- → PLA (programmable array logic)
- → PAL (programmable logic array)

0 + 0 = 0 ; 0 + 1 = 1 ; 1 + 0 = 1 ; 1 + 1 = 2 (10)

ADDERS:

→ Half - adder.
→ Full - adder.
→ Binary parallel Adder
(Ripple Carry adder)
→ Carry Look-Ahead Adder.

Half adder:



→ Half adder is a combinational circuit that performs arithmetic sum of Two bits.

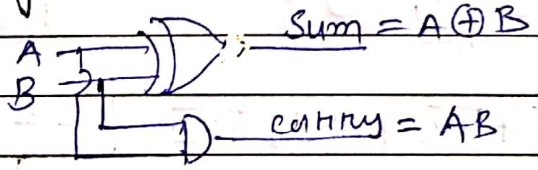
Truth table:

A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$$Sum = A \oplus B = \bar{A}B + A\bar{B}$$

$$Carry = A \cdot B.$$

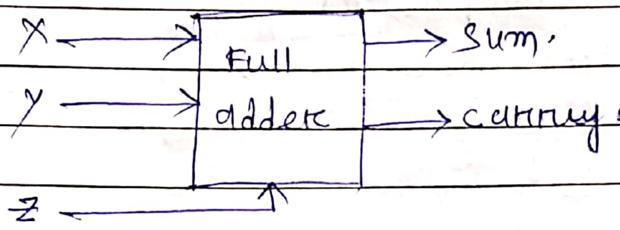
Logic diagram:



*→ required minimum no. of NAND and NOR gate to implement H.A = 5.

• Full Adder:

→ A full adder is a combinational circuit that performs the arithmetic sum of three bits



here, x, y = Two significant bits to be added.
and z = carry from the previous lower significant position.

Truth Table:

x	y	z	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

S

x \ yz	00	01	11	10
0	0_0	1_1	0_3	1_2
1	1_4	0_5	1_7	0_6

C

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

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

$\boxed{C = yz + xz + xy}$

$= \bar{z}(xy + \bar{x}\bar{y}) + z(x\bar{y} + y\bar{x})$

$= \bar{z}(x \odot y) + z(x \oplus y)$.