

8

180



-: HAND WRITTEN NOTES:-

OF

4

ELECTRONICS & COMMUNICATION ENGINEERING

4

-: SUBJECT:-

DIGITAL ELECTRONICS

8

(2)

(2)

⇒ Boolean Algebra :-

(i) When no. of variable are less. (1, 2, 3)

(ii) It is preferred when output is 0 or 1.

(3)

⇒ K-map :-

(i) When no. of variables are 2, 3, 4, 5 (upto 5 variable)

(ii) Output is 0, 1 or ∞ .

⇒ Tabulation method.

(i) It is used when no. of variables are more.

Boolean Algebra :-

(3)

⇒ A complement $\rightarrow \bar{A}$ or, A'

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

⇒ NOT :-

$$0 = 1$$

$$1 = 0$$

⇒ AND :-

$$0 \cdot 0 = 0$$

$$A \cdot A = A$$

$$0 \cdot 1 = 0$$

$$A \cdot 1 = A$$

$$1 \cdot 0 = 0$$

$$A \cdot 0 = 0$$

$$1 \cdot 1 = 1$$

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

⇒ OR :-

$$0+0 = 0$$

$$A+A = A$$

$$0+1 = 1$$

$$A+1 = 1$$

$$1+0 = 1$$

$$A+0 = A$$

$$1+1 = 1$$

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

Problem:- $AB + A\bar{B}$

Sol:- $A(B + \bar{B})$

$$= A$$

$$(\because B + \bar{B} = 1)$$

problem:- $AB + A\bar{B}C + A\bar{B}\bar{C}$, find the min. no. of NAND Gate.

option. (a) 0

(b) 1

(c) 2

(d) 3.

Sol:-

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

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

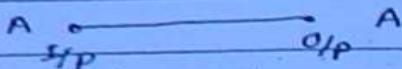
$$= AB + A\bar{B}$$

$$(\because C + \bar{C} = 1)$$

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

$$= A$$

NO NAND gate required.



4

Advantage of Minimization :-

⇒ No. of logic gate ↓

⇒ Speed ↑

⇒ Power dissipation ↓

⇒ complexity of circuit less

⇒ fan in ↓ (no. of input ↓)

⇒ Cost ↓.

Problem:- Simplify :-

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

Sol:- $A\bar{B}\bar{C} + A\bar{B} (1 + \bar{C}D)$

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

$$(1 + \lambda = 1)$$

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

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

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

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

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

Sol:- $A \cdot A + A \cdot C + AB + BC$

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

$$= A(1+B+C) + BC$$

$$= A + BC$$

Transposition Theorem

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

Similarly :

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

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

Sol:- take $A+B = X$

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

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

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

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

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

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

$$= A + BC$$

(S)

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

Sol:- $\overbrace{(A+B)} \quad \overbrace{(A+\bar{B})} \quad \overbrace{(\bar{A}+B)} \quad \overbrace{(\bar{A}+\bar{B})}$

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

$$= (A)(\bar{A})$$

$$= 0$$

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

Distribution theorem.

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

Sol:- $(A+\bar{A})(A+B)$

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

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

Sol:- $(A+\bar{A})(A+\bar{B})$

$$= 1(A+\bar{B}) = A+\bar{B}$$

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

Sol:- $A(B+\bar{B}) + \bar{A}\bar{B}$

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

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

$$= A + \bar{B} \text{ Ans.}$$

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

$$\begin{aligned} \text{Sol: } & B(A + \bar{A}) + A\bar{B} \\ &= B + A\bar{B} \\ &= (B + A)(B + \bar{B}) \\ &= A + B \quad \text{Ans.} \end{aligned}$$

(6)

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

$$\begin{aligned} \text{Sol: } & ABC + A\bar{B}C + A\bar{B}C + \bar{A}BC \quad (\because A + A = A) \\ &= AB(C + \bar{C}) + (A + \bar{A})BC \\ &= AB + BC \\ &= B(A + C) \end{aligned}$$

$$(iii) AB + \bar{A}C + \underline{BC} \rightarrow \text{redundant term.}$$

$$\begin{aligned} \text{Sol: } & AB + \bar{A}C + BC(A + \bar{A}) \\ &= AB + \bar{A}C + BCA + \bar{A}BC \\ &= AB(1 + C) + \bar{A}C(1 + B) \\ &= AB + \bar{A}C \end{aligned}$$

Note: In this case BC is known as redundant term i.e. not used or not compulsory term.

$\Rightarrow AB + \bar{A}C + BC = AB + \bar{A}C$, called consensus theorem or redundancy theorem.

⇒ Shortcut method :-

- (a) Three variable.
- (b) each variable comes twice.
- (c) One variable is complemented.

$$(iv) AB + B\bar{C} + AC$$

$$\text{Sol: } B\bar{C} + AC \quad \left\{ \begin{array}{l} \text{The term which is complemented} \\ \text{is taken.} \end{array} \right.$$

$$(v) A\bar{B} + BC + AC$$

$$\text{Sol: } A\bar{B} + BC$$

$$(m) (\bar{A}+B) (\bar{A}+\underline{C}) (B+C)$$

Sol: $(A+B) (\bar{A}+C)$, $\because (B+C)$ is redundant term.

$$(n) (\bar{A}+\underline{B}) (\bar{\underline{B}}+C) (A+C)$$

Sol: $(A+B) (\bar{B}+C)$

(o) $\bar{A}\bar{B} + \underline{A}\bar{C} + \bar{B}\bar{C}$

Sol: In this case all the variable are complemented only one are uncomplemented. then.

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

(\because The term which is uncomplemented is taken)

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

Sol: $\bar{B}C + \bar{A}\bar{B} \bar{A}\bar{C}$

(q) $(\bar{A}+\bar{B}) (\bar{B}+\underline{C}) (\bar{A}+\underline{C})$

Sol: $(\bar{B}+\bar{C}) (\bar{A}+C)$

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

Demorgan's theorem.

Boolean Algebra :-

↳ Minimization

⇒ SOP $\begin{cases} \rightarrow \text{minimal} \\ \rightarrow \text{canonical} \end{cases}$

⇒ POS $\begin{cases} \rightarrow \text{minimal} \\ \rightarrow \text{canonical} \end{cases}$

⇒ Dual

⇒ Complement Expression

⇒ Truth table

⇒ Venn Diagram

⇒ Switching circuit

⇒ Statement

(A) Minimization :-

(a) $XY + \bar{X}YZ$

Sol: $A = XY$ and $B = YZ$

Then,

= $A + \bar{A}B$

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

= $A + B$

= $XY + YZ$

8

(b) let $f(A+B) = \bar{A}+B$ Then the value of

$f[f(x+y), y], z$ is

(a) $XY + Z$

(c) $\bar{X}Y + Z$

(b) $X\bar{Y} + Z$

(d) X

Sol: $f[f\{(x+y), y\}, z]$

= $F[\bar{x+y} + BY, z]$

= $F[\bar{x}\bar{y} + Y, z]$

= $\bar{\bar{x}}\bar{y} + Y + Z$

= $\bar{\bar{x}}\bar{y} \cdot \bar{Y} + Z$

= $(\bar{\bar{x}} + \bar{\bar{y}})\bar{Y} + Z$

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

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

(c) let $x * y = \bar{x} + y$ and $z = x * y$

Then the value of $z * x$ is

(a) X

(c) 0

(b) 1

(d) \bar{X}

(B) SOP (Sum of Product Form)

$$\underbrace{ABC}_{\text{minterm}} + \underbrace{\bar{A}BC}_{\text{minterm}} + \underbrace{AB\bar{C}}_{\text{minterm}}$$

(9)

⇒ In SOP Form, each product term is known as Minterm or Implicant.

⇒ SOP Form is used when O/P of logical expression is 1.
(means $1 \rightarrow A$ and $0 \rightarrow \bar{A}$)

$$\text{Ex : } S \rightarrow 101 \rightarrow A\bar{B}C$$

$$g \rightarrow 1001 \rightarrow A\bar{B}\bar{C}D$$

Ques: For the given truth table, minimize SOP expression.

A	B	Y
$\bar{A}\bar{B}$	0	1 ✓
0	1	0
$A\bar{B}$	1	1 ✓
1	1	0

Sol: In SOP form only 1 taken.

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

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

$$= \bar{B}$$

⇒ Y can written as :-

$$Y(A, B) = \sum m(0, 2)$$

Ques: Simplified the expression for

$$Y(A, B) = \sum m(0, 2, 3)$$

Sol: $\begin{array}{ccc} 00 & 10 & 11 \end{array}$

logical expression in SOP form :-

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

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

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

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

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

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

SOP can be of two form.

a) Minimal form

b) Canonical form.

(10)

$\Rightarrow A + \bar{A}B + A\bar{B}$ (It is a minimal form)

\Rightarrow In canonical form, each term must have all variable.
e.g. $A + \bar{A}B + A\bar{B}$

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

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

Thus each min term will contain all variable.

ES-2003

Problem:- In canonical SOP form, no. of min term presenting the logical expression $A + \bar{B}C$ is.

(a) 4

(c) 6

(b) 5

(d) 7

Sol:-

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

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

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

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

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

i.e. 5 terms.

(C) POS Form (Product of Sum) :-

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

↳ max term

11

⇒ POS form are used when o/p is logic '0'.

$$0 \rightarrow A$$

$$1 \rightarrow \bar{A}$$

$$\text{Ex :- } 5 \rightarrow 101 \rightarrow \bar{A}BC$$

$$9 \rightarrow 1001 \rightarrow \bar{A}BC\bar{D}$$

Ques:- For a given truth table minimize POS expression.

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

Sol:- we take only that value at which o/p is '0'.

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

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

$$= \bar{B}$$

⇒ Y can be written in POS form as,

$$Y(A, B) = \prod M(1, 3) = \bar{B}$$

and for SOP :-

$$Y(A, B) = \sum m(0, 2) = \bar{B}$$

i.e.

$$\sum m(0, 2) = \prod M(1, 3)$$

$$\Rightarrow \text{If } F(A, B, C) = \sum m(0, 1, 4, 7)$$

There are 3 variable then 8 combination then max. term are, 2, 3, 5, 6.

$$F(A, B, C) = \sum m(0, 1, 4, 7) = \prod M(2, 3, 5, 6)$$

⇒ with 'n' variable, maximum possible minterms or
maxterms are 2^n . eq.

(i) for, $n = 2$, i.e. (A, B)

Total no. of min or max terms are $2^2 = 4$.

(ii) for, $n = 3$ i.e. (A, B, C)

Total no. of min or max terms are $2^3 = 8$

⇒ For $n = 2$, (A, B) total 16 logical expression i.e.

I	A	$A\bar{B}$	AB
0	\bar{A}	$\bar{A}B$	$A+\bar{B}$
	$\bar{A}B + A\bar{B}$	B	$A+\bar{B}$
	$AB + \bar{A}\bar{B}$	\bar{B}	$\bar{A}+\bar{B}$

Note:- With n variable maximum possible logical expression are 2^{2^n} .

eg. for $n = 2$, logical expression = $2^2 = 16$

for $n = 3$ = $2^3 = 256$

DES-2004
EATTE-2003
JTO-2001
JTO-2002

Problem:- For $n = 4$, what is the total no. of logical expression.

Sol:- logical expression = 2^4

$$= 2^{16} = 65536.$$

(P) Dual Form :-

(13)

+ive logic

-ive logic

\Rightarrow +ive logic means higher voltage corresponds to logic '1'.

\Rightarrow -ive logic means higher voltage corresponds to logic '0'.

\Rightarrow logic '0' \rightarrow 0V
logic '1' \rightarrow +5V

\Rightarrow logic 0 = +5V
logic 1 = 0V

Ques:- logic 0 \rightarrow -5V

logic 1 \rightarrow 0V

Sol:- Higher value of voltage (0V) for logic 1. then +ive logic.

Ques ECL :

logic 0 \rightarrow -1.7V

logic 1 \rightarrow -0.8V

Sol:- -0.8V is larger value than -1.7V then it is +ive logic.

+ive logic AND

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

-ive logic AND

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

+ive logic OR

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

-ive logic OR

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

\Rightarrow For -ive logic or gate, convert 1 to 0 and 0 to 1.

\Rightarrow we can say that +ive logic AND gate is equal to -ive logic OR gate and -ive logic AND gate is equal to +ive logic OR gate.

- ⇒ Dual expression is used to convert +ive logic into -ive logic or, -ive logic to +ive logic.
- ⇒ $AB \xrightarrow{\text{dual}} A+B$
- ⇒ dual is nothing but -ive logic.

(14)

- ⇒ AND $\xrightarrow[\text{Dual}]{\text{-ive logic}} \text{OR}$
- ⇒ OR $\xrightarrow[\text{Dual}]{\text{-ive logic}} \text{AND}$

- (1) AND \longleftrightarrow OR
 - (2) $\cdot \longleftrightarrow +$
 - (3) 1 $\longleftrightarrow 0$
 - (4) Keep variable as it is
- } Dual.

Ex:- Find Dual.

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

Dual :-

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

If we find again dual then,

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

- ⇒ For any logical expression , if two times dual is used resulting same expression.

Self Dual :-

$$AB + BC + AC$$

Dual :-

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

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

$$= BA + BC + AC + AC$$

$$= AB + BC + AC \quad (\text{again same expression})$$

- ⇒ In some of the logical expression not all its dual gives the same expression.

⇒ In self dual expression, if one time dual is used result in same expression.

$$n \text{ variable} \rightarrow \text{self dual} = 2^{2^{n-1}}$$

(15)

i.e. If there are n variables then total no. of self dual expression is $2^{2^{n-1}}$.

eg :-

(i) For $n=1 \Rightarrow 2^2 = 2$.

Then 2 dual expression.

$$\left. \begin{array}{l} A \rightarrow \text{self dual} \rightarrow A \\ \bar{A} \rightarrow \bar{A} \end{array} \right\} \text{Total self dual expression are 2.}$$

(ii) For $n=2 \Rightarrow 2^2 = 4$.

Then 4 dual expression.

$$\left. \begin{array}{ll} A \rightarrow A & , B \rightarrow B \\ \bar{A} \rightarrow \bar{A} & , \bar{B} \rightarrow \bar{B} \end{array} \right.$$

(iii) For $n=3 \Rightarrow 2^{2^{3-1}} = 16$.

Then 16 dual expression.

$$A, \bar{A}, B, \bar{B}, C, \bar{C}, \bar{AB} + \bar{BC} + \bar{CA}, AB + BC + CA, \dots$$

(E) Complement :-

$$\text{if } Y = ABC + \bar{A}BC + A\bar{B}C$$

complement is,

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

(•) AND \leftrightarrow OR

(•) . \leftrightarrow +

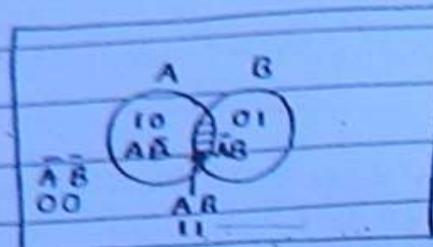
(•) 1 \leftrightarrow 0

(•) complement of each variable.

complement..

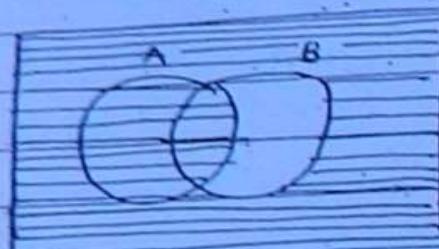
(F) Venn Diagram :-

For two variable (A, B).



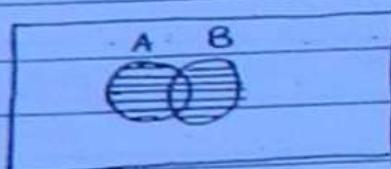
(16)

Ques:- For a given venn diagram, minimize the SOP expression for shaded region.



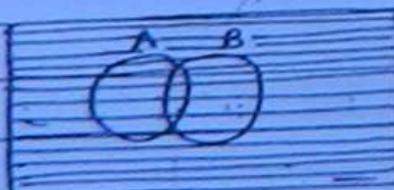
$$\begin{aligned}
 \text{Sol:-- } Y &= \bar{A}\bar{B} + A\bar{B} + AB \\
 &= \bar{B}(\bar{A}+A) + AB \\
 &= \bar{B} + AB \\
 &= (\bar{B}+A)(\bar{B}+B) \\
 &= A+\bar{B} \\
 &\quad \uparrow \downarrow \text{ for POS form.}
 \end{aligned}$$

Ques:- SOP expression for shaded region.



$$\begin{aligned}
 \text{Sol:-- } Y &= AB + A\bar{B} + \bar{A}\bar{B} \\
 &= A(B+\bar{B}) + \bar{A}\bar{B} \\
 &= A + \bar{A}\bar{B} \\
 &= (A+\bar{A})(A+\bar{B}) \\
 &= A+B \\
 &\quad \uparrow \downarrow \text{ (0,0) } \rightarrow \text{ (in POS form) }
 \end{aligned}$$

Ques:- SOP expression



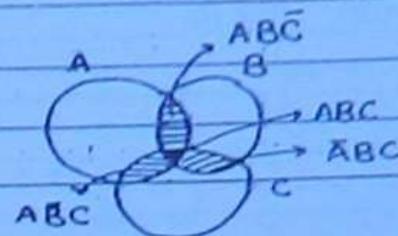
$$\begin{aligned}
 \text{Sol:-} \quad & \bar{A}\bar{B} + A\bar{B} + AB + \bar{A}B \\
 = & B(A + \bar{A}) + \bar{B}(A + \bar{A}) \\
 = & B + \bar{B} \\
 = & 1.
 \end{aligned}$$

(17)

⇒ For 3-variable :-

SOP form for shaded portion

$$\begin{aligned}
 \rightarrow \text{SOP form for shaded portion} &= ABC + \bar{A}BC + AB\bar{C} + A\bar{B}C + ABC + A\bar{B}\bar{C}, \\
 &= BC(A + \bar{A}) + AB(\bar{C} + C) + AC(B + \bar{B}) \quad \text{→ extradded.} \\
 &= AB + BC + CA
 \end{aligned}$$



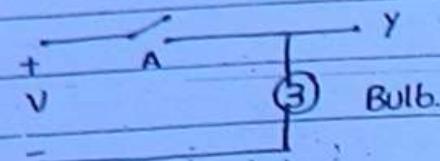
(a) Switching Circuit :-

(18)

For Series :-

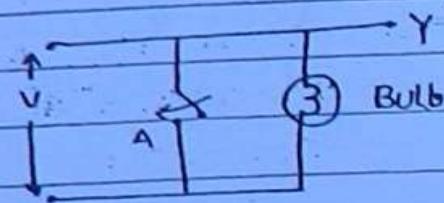
Truth table :-

A	Y
0	0
1	1



For Parallel :-

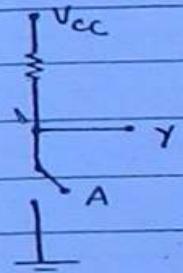
A	Y
0	1
1	0



⇒ In place of bulb if there is resistor then answer remains the same but some drop.

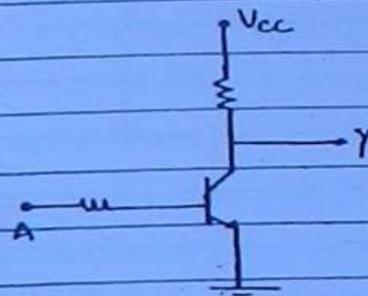
Truth table :-

A	Y
0	1
1	0



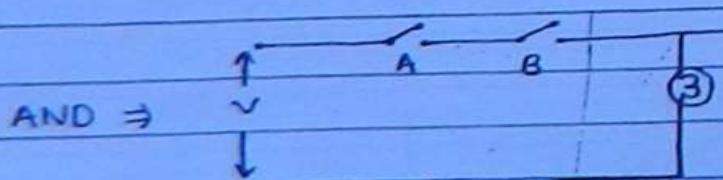
⇒ In place of switch if there is a transistor.

A	Y
0	1
1	0

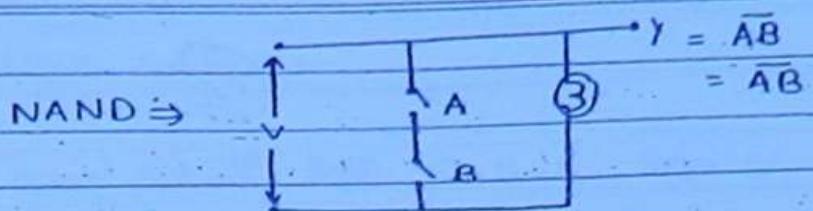


(a) For $A=L$, transistor becomes short circuit.

For two switch A and B :-

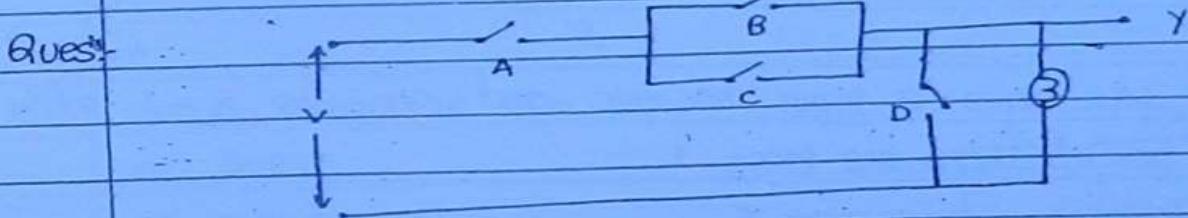
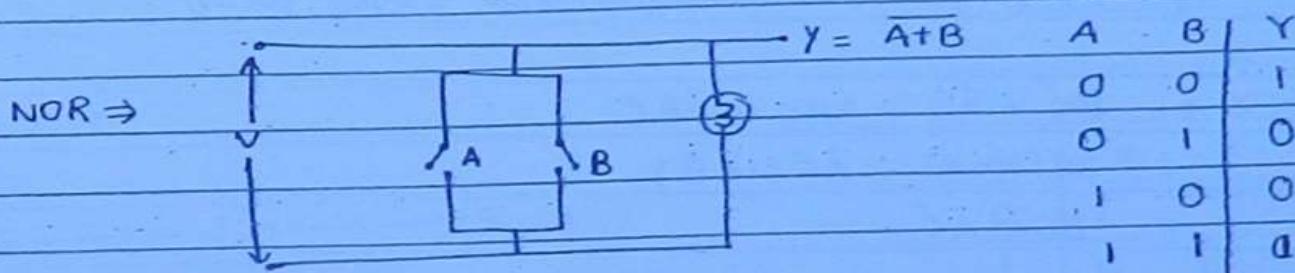
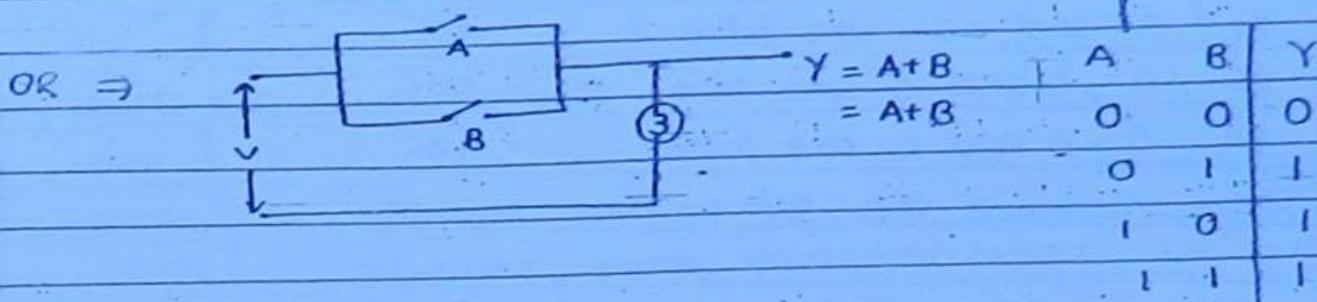


A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1



(19)

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



Sol:

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

$$= (AB + AC) \bar{D}$$

$$= AB\bar{D} + AC\bar{D}$$

(H) STATEMENT :-

Ques

A logic circuit have 3 input A, B, C and o/p is Y.
 Ques:- o/p Y is 1. for the following combination.

- (i) B and C are true = BC
- (ii) A and C are false = $\bar{A}\bar{C}$
- (iii) A, B and C are true = ABC
- (iv) A, B and C are false = $\bar{A}\bar{B}\bar{C}$

then minimize the o/p for Y.

Sol:- o/p Y=1. (take min term = SOP form)

$$\begin{aligned} Y &= BC + \bar{A}\bar{C} + ABC + \bar{A}\bar{B}\bar{C} \\ &= BC(1+A) + \bar{A}\bar{C}(1+B) \\ &= BC + \bar{A}\bar{C} \end{aligned}$$

If o/p Y=0, then take max term (POS form).

Ques:- A logic ckt have 3 input A, B, C and o/p is F = 1. when majority no. of I/Ps are logic 1.

- > (i) minimizing expression F
- (ii) Implement logic ckt

Sol:-	A	B	C	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

$$\begin{aligned} F &= \bar{A}\bar{B}\bar{C} + A\bar{B}\bar{C} + AB\bar{C} + ABC \\ &= \bar{A}\bar{B}\bar{C} + ABC + ABC + A\bar{B}\bar{C} + AB\bar{C} + ABC \\ &= BC(A+\bar{A}) + AC(B+\bar{B}) + AB(\bar{C}+C) \\ &= AB + BC + CA. \end{aligned}$$

(I) LOGIC GATES :-

⇒ Basic Building Blocks

(21)

NOT
AND } → Basic gate |

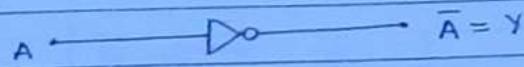
OR }
NAND } → universal gate

NOR }

EXOR } → Arithmetic ckt.

EXNOR } comparator, parity generator/checker, code converter
(Binary to gray, Gray to Binary)

NOT :-



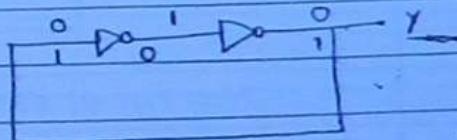
A	Y
0	1
1	0



IES-2010
GAZP-2010

Ques:- Circuit shown in the fig are

- (a) Buffer
- (b) Astable MV
- (c) Bistable MV
- (d) square wave generator.



Sol:- If there is no feedback then it is buffer. In Buffer if we apply 0 then get 0

" 1 " " 1 "

" no I/P " " no I/P "

Buffer means whatever the I/P ie. the O/P.

⇒ But there is a feedback and the O/P is stable if we give 1 as I/P, O/P is also 1 and if gives 0 then O/P is 0 that two stable state.

⇒ Hence it is Bistable multivibrator.

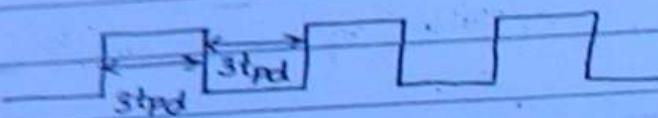
Ques: CKT shown is



Sol: tpd = Propagation delay.

$$0^{\circ} \text{ for } = s tpd$$

$$1^{\circ} \text{ for } = s tpd$$



It is called

- (i) Square wave generator.
- (ii) As o/p is not stable sometime 1 and sometime 0 hence it is also called astable multivibrator.
- (iii) Clock generator
- (iv) Ring oscillator.

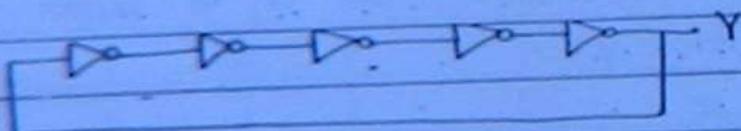
$$\text{Total time period (T)} = 6 tpd$$

then,

$$T = 2N tpd$$

N = no. of inverters in feedback.

Ques:- In a ckt shown in fig. the propagation delay of each NOT gate is 100 Psec. Then frequency of generator square wave is



(A) 10 GHz

~~(B)~~ 1 GHz

(C) 100 MHz

(D) 10 MHz

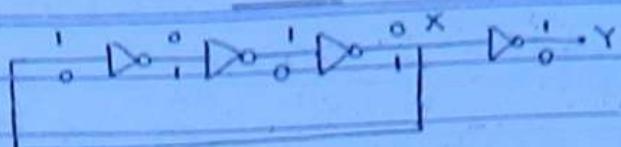
Sol: $T = 2N tpd$

$$= 2 \times 5 \times 100 \text{ Psec} = 1000 \text{ Psec}$$

$$f = \frac{1}{T} = \frac{1}{1000 \times 10^{12} \text{ sec}} = 10^9 \text{ Hz}$$

classmate = 1 GHz

Ques:-



(23)

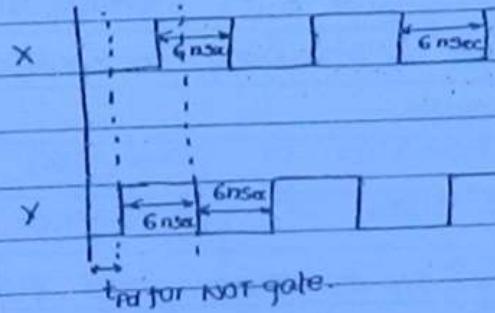
Sol:- The CKT shown in the fig. the propagation delay of each NOT gate is 2nsec. Then time period of generated square wave is.

- (a) 6ns (c) 14ns
 (b) 12ns (d) 16ns

Sol:- Astable Multivibrator, square wave generator.

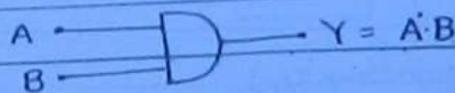
$$T = 2 \times t_{pd}$$

$$= 2 \times 3 \times 2 \text{ nsec} = 12 \text{ nsec.}$$



Thus time period at x and y is same.

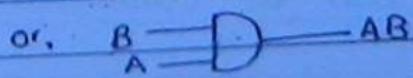
AND GATE :-



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

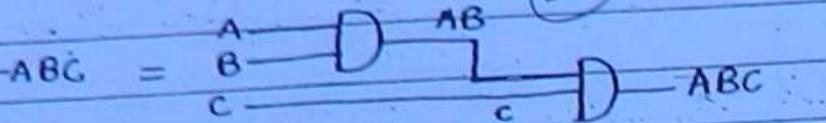
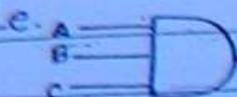
⇒ o/p is low if any of the i/p is low i.e. logic '0'.

⇒ AND gate follow both commutative law and associative law.
 (i) $AB \equiv BA$

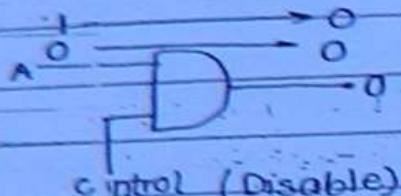


$$(c) ABC = (AB) \cdot C = A(BC)$$

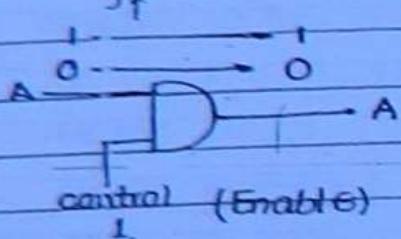
i.e.



\Rightarrow Disable & Enables :-



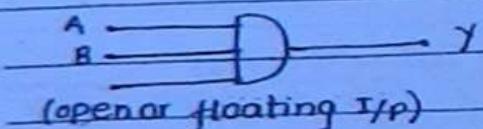
\Rightarrow Thus o/p remains in '0' due to control I/P disable. AND gate is not in working state.



\Rightarrow AND gate is in working state o/p is changing in Enabled state.

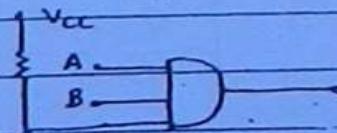
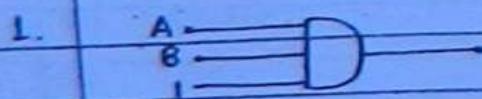
\Rightarrow In TTL logic family, If any I/P is open and float then it will act as '1'.

\Rightarrow In ECL logic family, floating input will act as logic '0'.



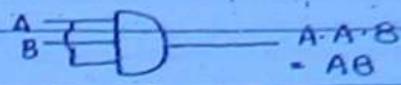
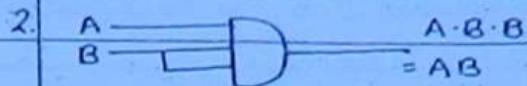
* Question occurs mostly from ECL and TTL in Exam.

Unused I/P's :-



\Rightarrow In Multipin (I/P) AND gate unused I/P can be connected to logic 1. or "pull off" up".

\Rightarrow unused I/P can be connected to logic '0' or "pull down".



(25)

⇒ unused I/P can be connected to one of the used I/P.



⇒ If it is TTL logic family, then unused I/P can be open or floated. (unconnected)

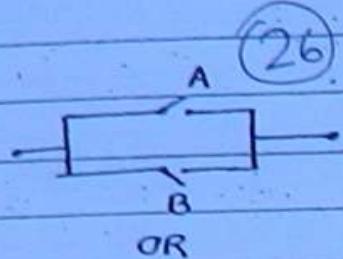
Note:- Because of unnecessary I/P attached to B, fan-in will be down.



⇒ Best way to connecting unused pin (I/P) in AND gate is connecting to logic '1'.



OR Gate :- (Inclusive OR)



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

⇒ When any of the I/P is High in OR gate then o/p is High.

⇒ OR gate follows both commutative and Associative law.

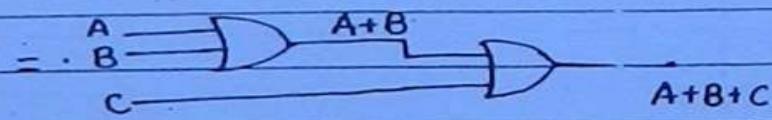
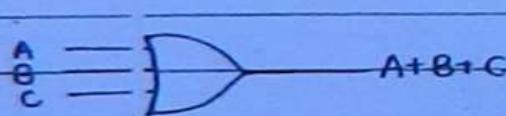
i) Commutative law :-

$$A+B = B+A$$

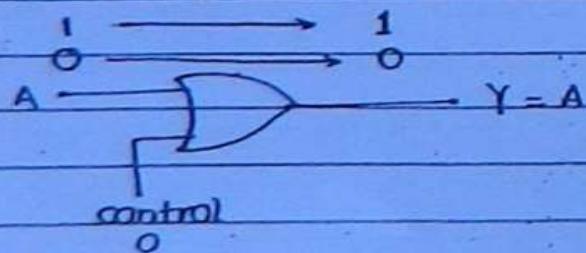


ii) Associative law :-

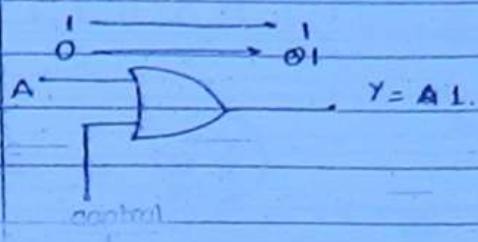
$$A+B+C = (A+B)+C = A+(B+C)$$



⇒ Enable and Disable :-



⇒ O/P is changing as I/P is changing or we say the gate is enabled.



\Rightarrow O/P is fixed or not changed.
it is said to be disable.

(27)

Unused I/P's :-

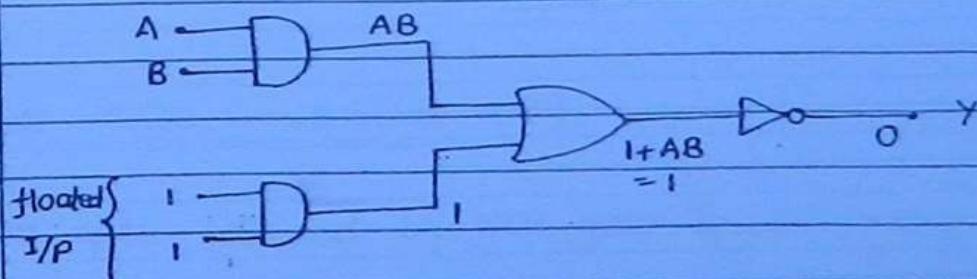
1. In OR gate, unused I/P is connected to logic '0' - "pull down."
2. Connect to one of the used I/P.
3. If it is ECL then unused I/P can be open or floated.

\Rightarrow In OR gate, Best way of connecting the unused I/P is to connect to logic '0'.



GATE-2004.

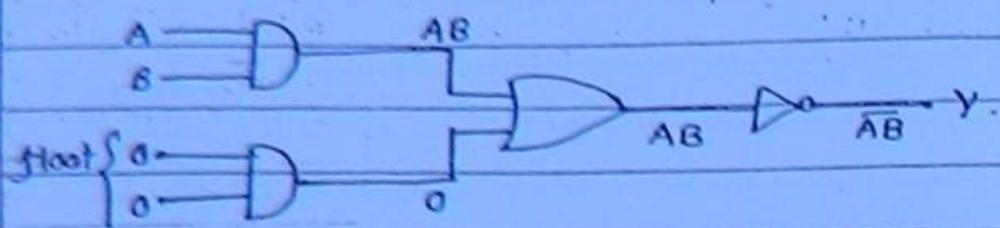
Problem:- In the CKT shown in fig. in TTL, AND, OR, INVERTER CKT for the given I/P O/P is



- (A) 0
 (B) 1
 (C) AB
 (D) \bar{AB}

Sol:- In TTL, all I/P's are float then it is logic L

Problem: For ECL AND, OR, INVERTER.



(26)

- (A) 0
- (B) 1
- (C) AB
- (D) \bar{AB}

Sol:- If all I/P are floating in ECL then it is '0'
and o/p $Y = \bar{AB}$ Ans.

NAND GATE :- (Bubbled OR)

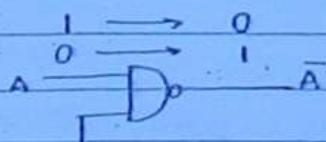
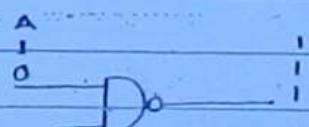
$$\begin{array}{c} A \\ \text{---} \\ B \end{array} \text{---} \text{D}_o \text{---} \bar{A}\bar{B} = \bar{A} + \bar{B}$$

$$\begin{array}{c} A \\ \text{---} \\ B \end{array} \text{---} \text{D}_o \text{---} \bar{A} + \bar{B}$$

(29)

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

\Rightarrow When both I/P high the O/P is low.



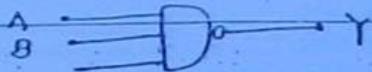
0 disable (not changing if one I/P is zero)
1 enable.

\Rightarrow NAND gate follow commutative law but not follow associative law

$$\begin{array}{c} A \\ \text{---} \\ B \\ \text{---} \\ C \end{array} \text{---} \text{D}_o \text{---} \bar{ABC} \neq \begin{array}{c} A \\ \text{---} \\ B \\ \text{---} \\ C \end{array} \text{---} \text{D}_o \text{---} \bar{AB} \text{---} \text{D}_o \text{---} \bar{(AB)}C = AB + C$$

\Rightarrow The only two gate not follow associative law ie universal gate NAND or NOR gate.

\Rightarrow Unused I/P in NAND gate can be connected similar to unused I/P in AND gate.



NOR GATE :- (Bubbled AND)

\Rightarrow OR gate followed by NOT gate.

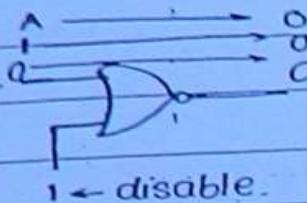
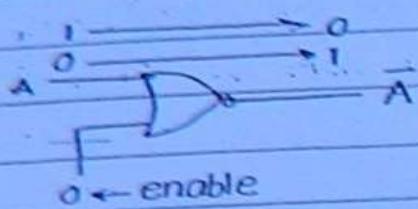
$$\begin{array}{c} A \\ \text{---} \\ B \end{array} \text{---} \text{D}_o \text{---} \bar{A} + \bar{B} = \bar{A} \cdot \bar{B}$$

$$\begin{array}{c} A \\ \text{---} \\ B \end{array} \text{---} \text{D}_o \text{---} Y = \bar{A} \cdot \bar{B}$$

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

⇒ when both I/P is low the O/P is High.

(30)



- ⇒ enable and disable both are same as OR gate.
- ⇒ NOR gate follow commutative law and not follow associative law.

$$\text{i.e. } \text{A} \overline{\text{A+B}} = \overline{\text{B+A}}$$

$$\text{ii) } \overline{\text{A+B+C}} \neq \overline{\text{A+B}} \text{ C}$$

- ⇒ unused I/P in NOR gate can be connected similar to OR gate.

EXOR or XOR :-

- ⇒ Exclusive OR gate.

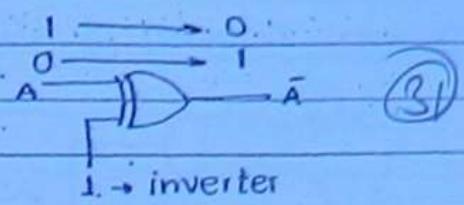
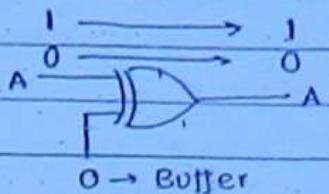
- ⇒ OR gate is also called as inclusive OR gate.



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

⇒ when A = B, O/P is low i.e '0'.

⇒ when A ≠ B, O/P is High i.e. logic '1'.



⇒ It is also called controlled inverter.

Note:-

$$A \oplus A = 0$$

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

$$A \oplus 0 = A$$

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

⇒ If $A \oplus B = C$ then,

- (i) $A \oplus C = B$
- (ii) $B \oplus C = A$
- (iii) $A \oplus B \oplus C = 0$

⇒ Since, $A \oplus A = 0$ } Then we say.
 $A \oplus A \oplus A = A$ } odd no. of same I/P gives same O/P
 $A \oplus A \oplus A \oplus A = 0$ } and even no. same O/I/P gives 0/P.
 and so on --- } as O/P.

⇒ $B \oplus B \oplus B \oplus \dots n = B$, if n is odd
 $= 0$, if n is even.

Problem:- The okt shown in fig. contains cascoding of 20 EXOR gate.

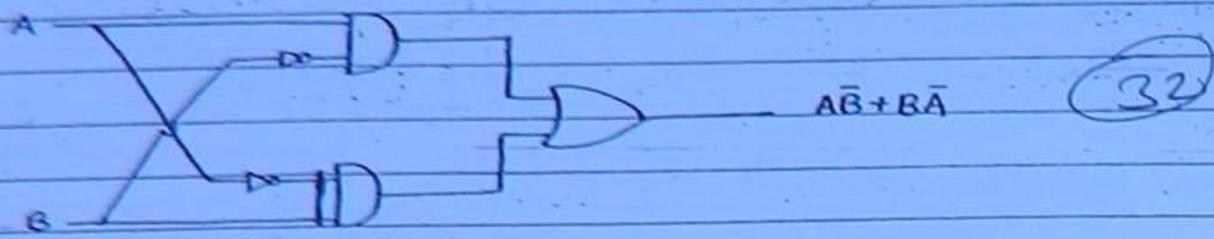
If x is the I/P then O/P is.

- (a) 0
 - (b) 1
 - (c) x
 - (d) \bar{x}
-

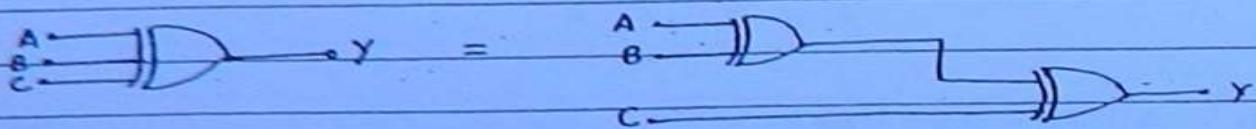
Sol: O/P of even EXOR gate have same O/P.

= x Ans.

Internal diagram of EXOR gate :-



- ⇒ EXOR gate follows both commutative and associative law.
- ⇒ EXOR gate is fully available with two I/P's only.



Truth table :-

A	B	C	Y ($A \oplus B \oplus C$)
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

- ⇒ The O/P of EXOR gate is 1. when no. of 1's at the I/P is odd no.

⇒ logical expression :-

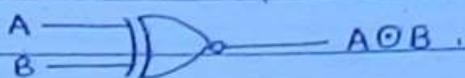
$$\begin{aligned}
 Y &= \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} + ABC \\
 &= 001 + 010 + 100 + 111 \rightarrow \text{odd no. of 1's.}
 \end{aligned}$$

$$Y = \sum m(1, 2, 4, 7)$$

- ⇒ The reduced form of this expression is,

$$A \oplus B \oplus C$$

EXNOR or XNOR :-



(35)

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

⇒ whenever the I/P is same O/P is High.

$$SOP \text{ expression} = \bar{A}\bar{B} + AB$$

$$POS \text{ expression} = (A+\bar{B})(\bar{A}+B)$$

$$\begin{array}{l} \text{EXOR} :- \bar{A}\bar{B} + A\bar{B} \\ (A+B)(\bar{A}+\bar{B}) \end{array}$$

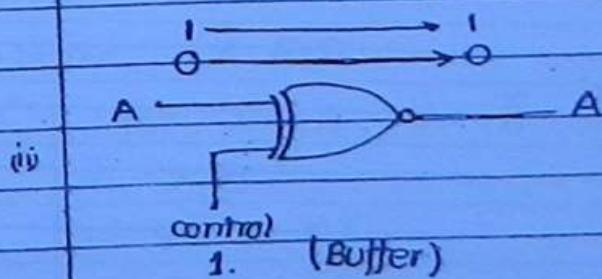
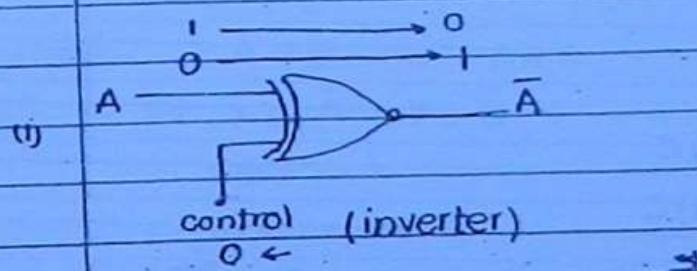
$$\begin{array}{l} \text{EXNOR} :- \bar{A}\bar{B} + AB \\ (A+\bar{B})(\bar{A}+B) \end{array}$$

⇒ When $A = B$, then O/P is High.

therefore coincidence logic ckt and also called as equivalent detector.

⇒ When $A \neq B$, The O/P is low.

⇒ Enable and Disable :-



$$AOA = 1$$

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

$$AO\bar{O} = \bar{A}$$

$$AOI = A$$

(34)

$$\text{since, } AOA = 1$$

$$AOAOA = A$$

$$AOAOAOA = 1.$$

and so on.

$$B \oplus B \oplus B \oplus \dots n = \begin{cases} 1 & \text{if } n = \text{even} \\ B & \text{if } n = \text{odd} \end{cases}$$

\Rightarrow EXOR and EXNOR is not always complement, it is complement only when the no. of I/P is even. and if I/P is odd then EXOR and EXNOR are same.

$$\text{i.e. } A \oplus B \oplus C = A \odot B \odot C \Rightarrow \text{same.}$$

$$\text{and, } A \oplus B \oplus C \oplus D = \overline{A \odot B \odot C \odot D} \Rightarrow \text{complement}$$

Ques:- Find expression of $A \odot B \odot C$.

$$\text{Sol:- } A \odot B \odot C$$

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

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

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

Since,

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

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

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

$$= A \oplus B \oplus C.$$

ques - Minimize.

A	B	C	Y	
→ 0	0	0	1	(A) $A \oplus B \oplus C$
0	0	1	0	(B) $A \ominus B \odot C$
0	1	0	0	(C) $\overline{A \oplus B \odot C}$
→ 0	1	1	1	(D) $AB + BC + AC$
1	0	0	0	
→ 1	0	1	1	
→ 1	1	0	1	
1	1	1	0	

(35)

sol:- for EXOR \rightarrow O/P is 1 when odd no. of 1's at I/P.

In this case.,

$$\begin{aligned} Y &= \overline{A \oplus B \oplus C} \\ &= \overline{A \ominus B \odot C} \quad \text{Ans.} \end{aligned}$$

\Rightarrow EXOR and EXNOR are never always complemented, It is complement only when even variable occurs.

\Rightarrow EXNOR gate is even no. of 1's detector when no. of I/P's are even.

\Rightarrow EXNOR gate is odd no. of 1's detector when no. of I/P's are odd.

Problem:- $\bar{A} \oplus B = A \ominus B$.Sol:- Put $x = \bar{A}$, $y = B$

$$\begin{aligned} &x \oplus y \\ &= \cancel{A}x\bar{y} + \bar{x}y \\ &= \bar{A}\bar{B} + AB = A \ominus B. \end{aligned}$$

Problem:- $\hat{A} \oplus \bar{B}$ Sol:- Put $x = A$, $y = \bar{B}$

$$\begin{aligned} &x \oplus y \\ &= x\bar{y} + y\bar{x} \\ &= AB + \bar{A}\bar{B} \\ &= A \ominus B. \end{aligned}$$

Problem:-

$$A \oplus B \oplus AB$$

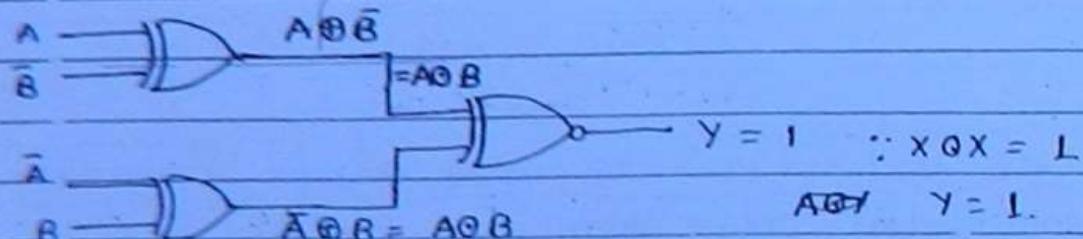
Sol:-

$$\begin{aligned}
 & (A\bar{B} + \bar{A}\bar{B}) \oplus AB \\
 &= (A\bar{B} + \bar{A}\bar{B}) \bar{A}\bar{B} + (A\bar{B} + \bar{A}\bar{B}) AB \\
 &= A\bar{B}(\bar{A} + \bar{B}) + \bar{A}\bar{B}(\bar{A} + \bar{B}) + (\bar{A}\bar{B} \cdot \bar{A}\bar{B}) AB \\
 &= A\bar{B} + \bar{A}\bar{B} + [(A + B)(\bar{A} + \bar{B})]AB \\
 &= A\bar{B} + \bar{A}\bar{B} + [\bar{A}\bar{B} + AB]AB \\
 &= A\bar{B} + \bar{A}\bar{B} + AB \\
 &= A(\bar{B} + B) + \bar{A}\bar{B} = A + \bar{A}\bar{B} \\
 &= (A + \bar{A})(A + \bar{B}) = A + B \quad \text{Ans}_C
 \end{aligned}$$

(36)

$$A \oplus B \oplus AB = A + B$$

Problem:-



(O) 0

(E) 1

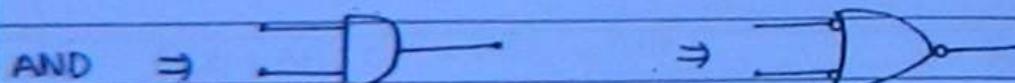
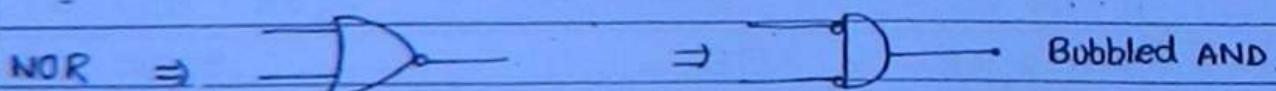
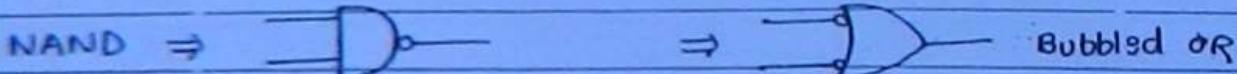
(G) $A \oplus B$

TOT $A \oplus B$

Sol:-

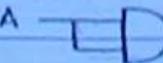
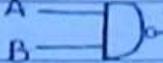
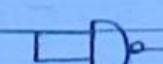
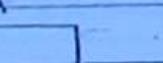
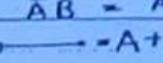
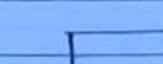
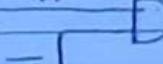
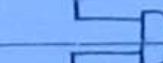
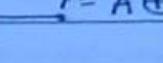
$$Y = 1 \quad \text{Ans}_C$$

SYMBOLS :-

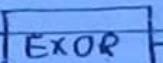
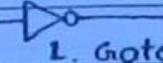
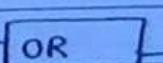
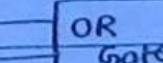


NAND as Universal :-

(37)

(i) NOT :- A  \bar{A} \Rightarrow 1 gate required(ii) AND :- A  \bar{AB}  AB \Rightarrow 2 gate(iii) OR :- A  \bar{A}  \bar{B} $\bar{AB} = \bar{A} + \bar{B} \Rightarrow 3$ gate.  $A + B$ (iv) EXOR :- A  \bar{AB}  $A \cdot \bar{AB}$  B  $B \cdot \bar{AB}$  $Y = A \oplus B = \bar{A}B + A\bar{B}$ $\Rightarrow 4$ Gate.

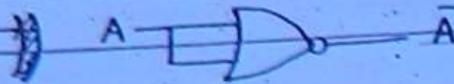
$$\begin{aligned}
 Y &= (\bar{A} \cdot \bar{B} \cdot B \cdot \bar{A}) \\
 &= (\bar{A} \cdot \bar{B} + B \cdot \bar{A}) \\
 &= (A(\bar{A} + \bar{B}) + B(\bar{A} + \bar{B})) \\
 &= A\bar{B} + B\bar{A} = A \oplus B
 \end{aligned}$$

(v) EXNOR :- A  \bar{A} \Rightarrow 5 Gate
B  1 Gate
4 Gate(vi) NOR :- A  \bar{A} \Rightarrow 4 Gate
B  \bar{B} \Rightarrow 3 Gate
1 Gate

NOR AS universal :-

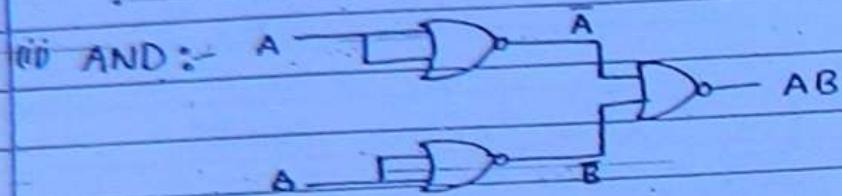
(38)

(i) NOT :-



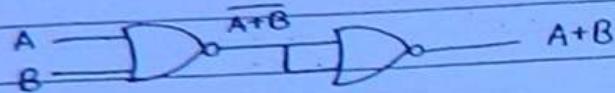
$\Rightarrow 1$ gate

(ii) AND :-



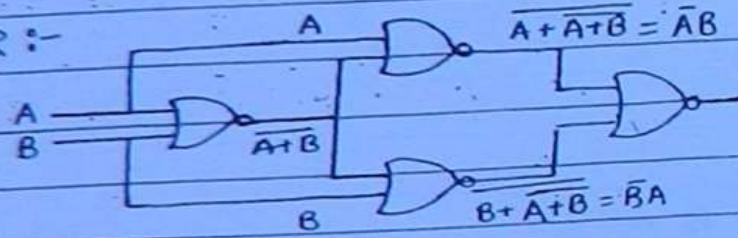
$\Rightarrow 3$ gate

(iii) OR :-



$\Rightarrow 2$ gate

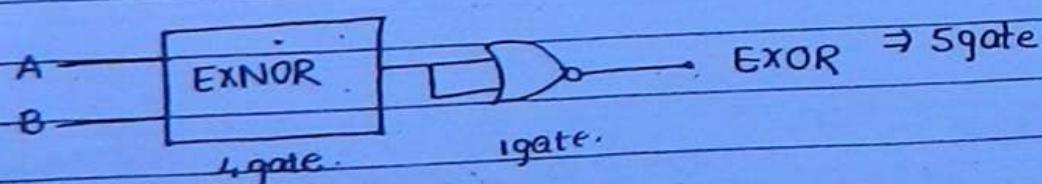
(iv) EXNOR :-



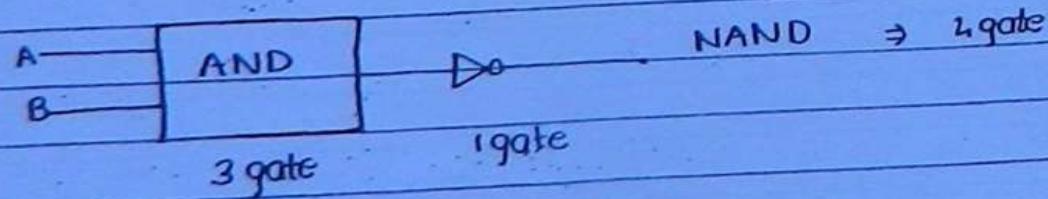
$\Rightarrow 4$ gate

$$\begin{aligned} y &= (\overline{\bar{A} + \bar{B}}) + (\overline{\bar{B} + \bar{A}}) = \overline{\bar{A}(A+B)} + \overline{\bar{B}(A+B)} \\ &= \overline{\bar{A}\bar{B}} + \overline{\bar{B}\bar{A}} = \overline{AB} \\ &= \overline{(A \oplus B)(B \oplus A)} \\ &= \overline{AB} + \overline{A(\bar{A}\bar{B})} + \overline{B(A\bar{B})} + \overline{AB} = A \oplus B \end{aligned}$$

(v) EXOR :-



(vi) NAND

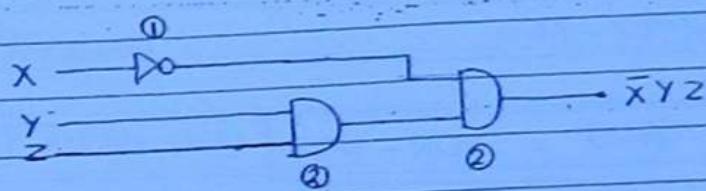


Note:-	Logic gate	No. of NAND	No. of NOR
	NOT	1	1
	AND	2	3
	OR	3	2
	EXOR	4	5
	EXNOR	5	4

(39)

Problem:- To implement $\bar{X}YZ$. The min no. of two I/P NAND gate required.

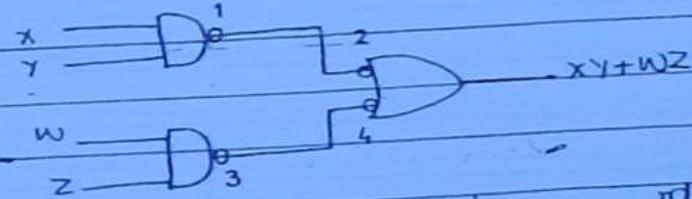
Sol:-



Total no. of NAND gate = $2+2+1 = 5$. Ans.

Problem:- To implement $XY + WZ$, the min no. of 2 input NAND gate required.

Sol:-



\Rightarrow 1st inverter cancelled 2nd and 3rd cancelled 4th.

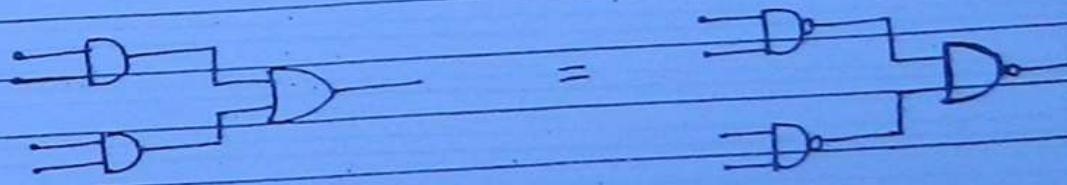
\Rightarrow Now the total no. of NAND \oplus gate is.

$$= 2 + \text{Bubbled OR} (= \text{NAND})$$

$$= 2 + 1 = 3.$$

= 3 NAND gate required.

Note:-



Two level AND-OR = Two level NAND-NAND

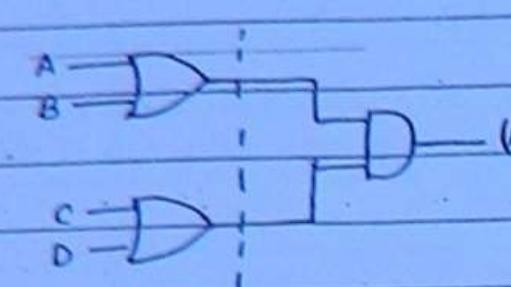
AND-OR = NAND-NAND

- ⇒ To implement SOP form, only NAND gate alone.
 ⇒ To implement POS form, only NOR gate alone.

(40)

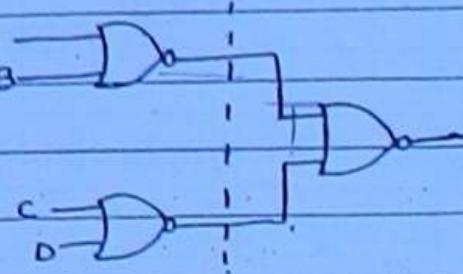
Q:- if $(A+B)(C+D)$, then min no. of Gate.

Sol:-



Twolevel OR-AND

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



Twolevel NOR-NOR

⇒ OR-AND = NOR-NOR

Digital circuits

(G)

↓
combination ckt

↓
Sequential ckt

⇒ Present O/P is only depend
on present I/P.

⇒ Present O/P [Present I/P
Previous O/P]

⇒ No feedback

⇒ feedback.

⇒ No memory

⇒ Memory.

⇒ e.g. Half Adder (HA)

⇒ e.g:- FlipFlop (FF)

FA

Register

MUX

Counter.

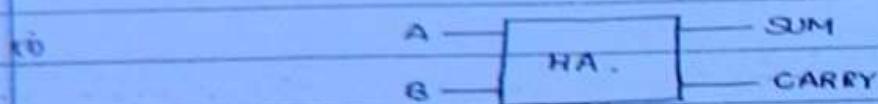
DEMUX

Procedure to Design :-

- (i) Identify I/P and O/P.
- (ii) Construct truth table
- (iii) Write logical expression in SOP or POS form.
- (iv) Minimize logical expression if possible
- (v) Implement logic circuits.

42

(A) HALF ADDER (HA) :-



Truth table :-

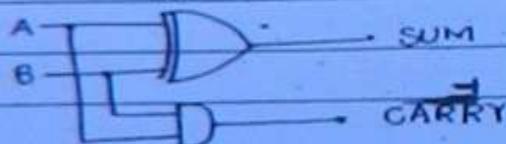
A	B	SUM	CARRY
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Logical expression:-

$$\text{SUM} = \bar{A}\bar{B} + A\bar{B} = A \oplus B$$

$$\text{CARRY} = AB$$

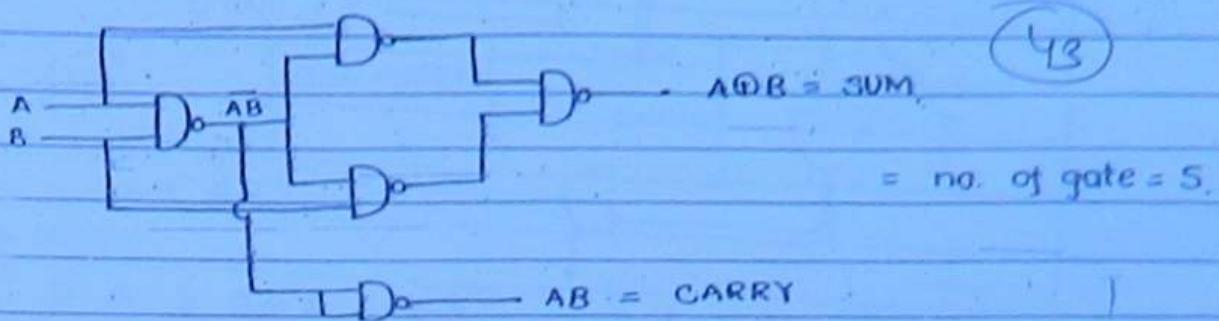
Implement :-



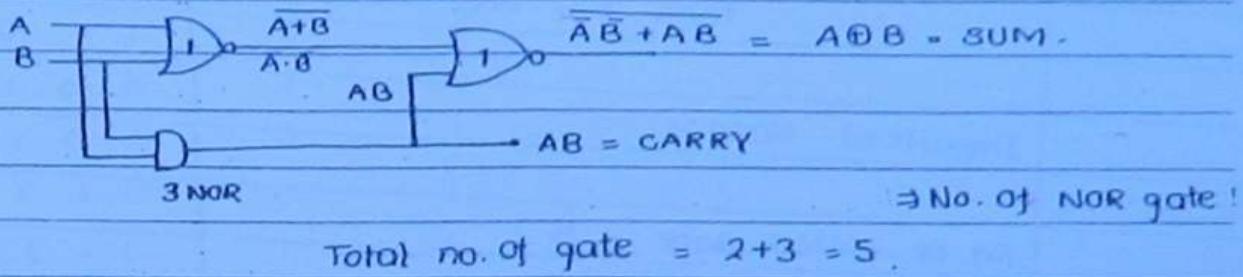
Important Ques :-

- (i) logical Expression for SUM :- $A \oplus B$ CARRY: AB
- (ii) Min. no. of NAND Gate : 5
- (iii) Min. no. of NOR Gate : 5
- (iv) No. of MUX : 3
- (v) No. of DECODER: 1, 2x4 Decoder and 1 OR Gate

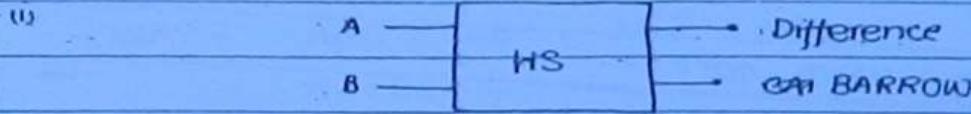
HA USING NAND gate :-



HA using NOR Gate :-



(B) HALF SUBTRACTOR :-



(ii) Truth table :-

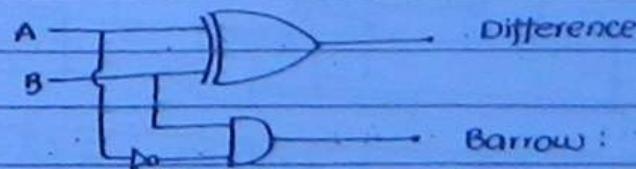
A	B	Difff.	Barrow
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

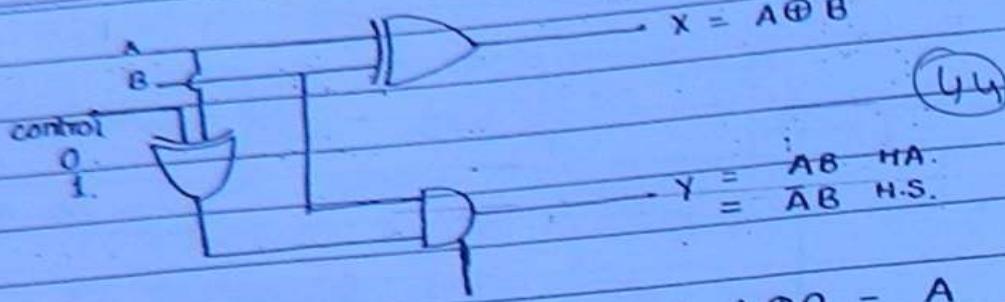
(iii) logical expression :-

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

$$\text{Barrow} = \bar{A}B$$

(iv) Implement :-





Note :- i) if control = 0, then $A \oplus 0 = A$
 then $Y = AB$ and ckt is HA.
 ii) if control = 1 then, $A \oplus 1 = \bar{A}$
 then $Y = \bar{AB}$ and ckt is HS.

Important Ques :-

No. of NAND Gate = 5

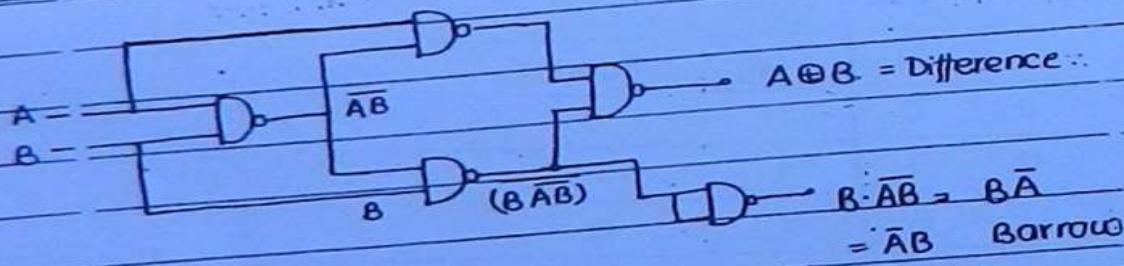
No. of NOR Gate = 5

No. of MUX :- 3 : (2x1 MUX)

No. of DECODER :- 1 (2x4) Decoder and 1 OR Gate.

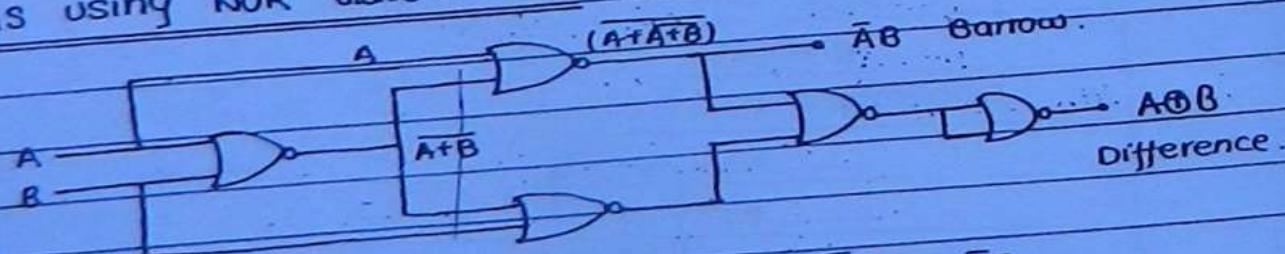
logical expression for Difference = $A\bar{B} + \bar{A}B$, Borrow = $\bar{A}\bar{B}$.

HS using NAND gate :-



No. of NAND Gate = 5

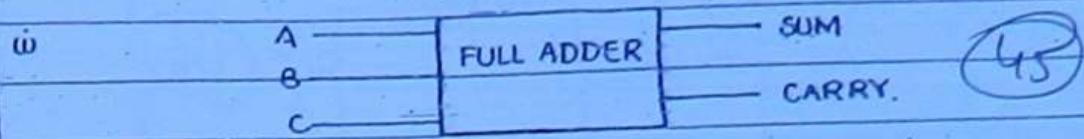
HS using NOR Gate :-



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

classmate \Rightarrow No of NOR gate = 5

(C) FULL ADDER :-



vii Truth table:-

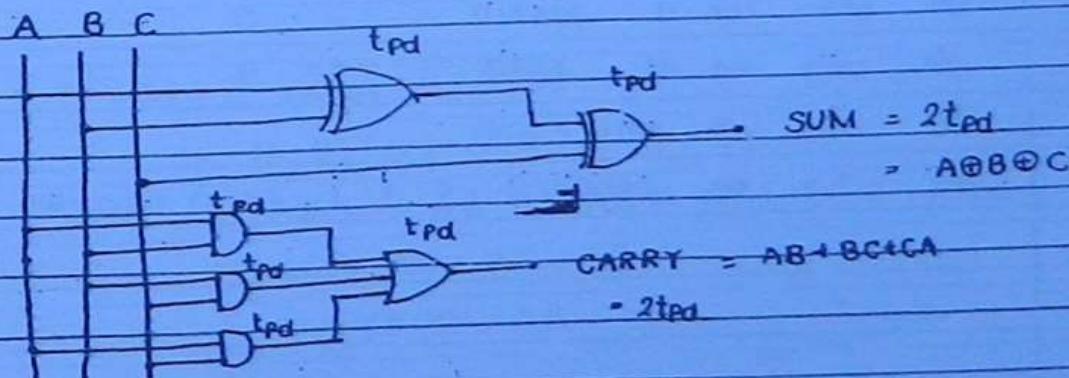
A	B	C	SUM	CARRY
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

viii logical expression :-

$$\text{SUM} = \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} + ABC = \sum m(1, 2, 4, 7) = A \oplus B \oplus C$$

$$\text{CARRY} = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC = AB + BC + AC.$$

⇒ The truth table of carry shows the majority of 1's function.
 $= \sum m(3, 5, 6, 7)$



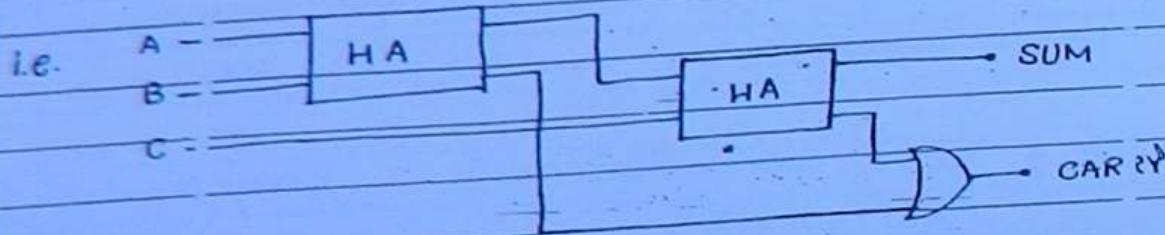
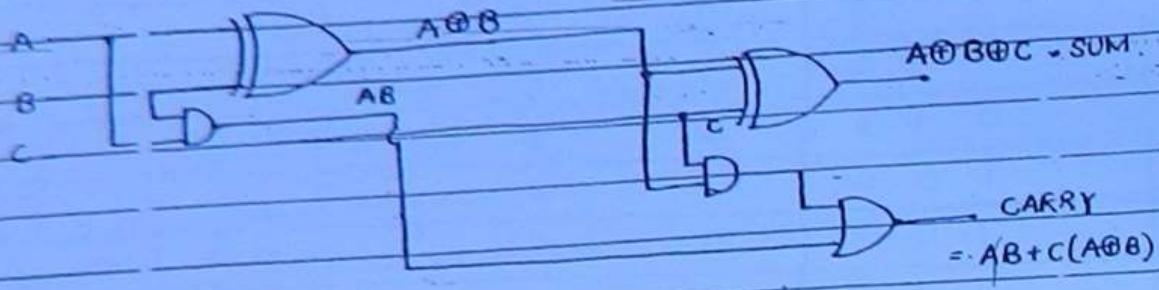
⇒ In full adder each logic gate have propagation delay of t_{pd} to provid sum or carry o/p, it requires to $2t_{pd}$ delay.

$$\begin{aligned}
 \text{Now, } \text{CARRY}_{\text{SUM}} &= \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC \\
 &= AB(C + \bar{C}) + C(\bar{A}B + AB) \\
 &= AB + C(A \oplus B)
 \end{aligned}$$

(46)

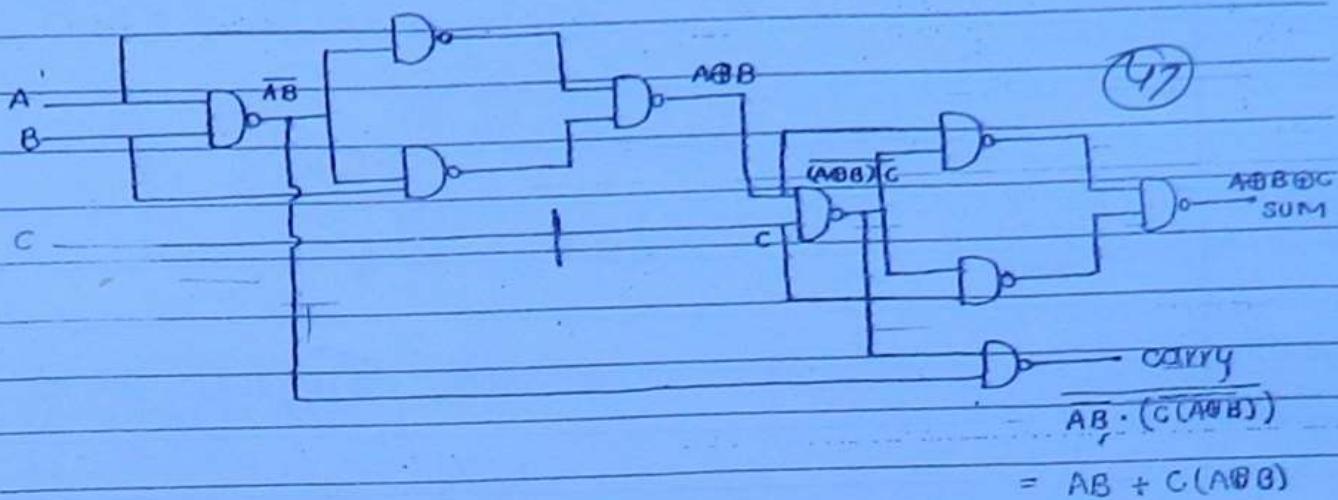
$$\boxed{\text{CARRY} = AB + C(A \oplus B)}$$

Implementation:-

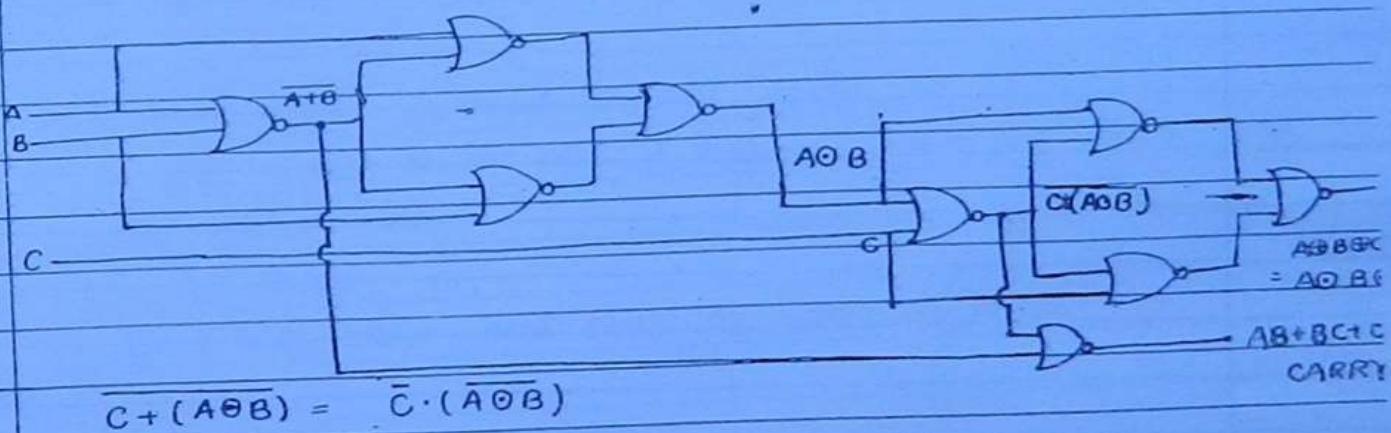


Important ques:-

- (i) logical expression for SUM = $A \oplus B \oplus C$ CARRY = $AB + BC + AC$
- (ii) No. of HA and OR gate = 2HA, 1-OR
- (iii) min. no. of NAND = 9
- (iv) min. no. of NOR = 9
- (v) No. of MUX = 1
- (vi) No. of DECODER = 1, (3x8) Decoder and 2-OR gate

vii) Implementation of Full adder using NAND gate :-viii) Implementation of Full adder using NOR gate :-

since $A \oplus B \oplus C = A \oplus B \oplus C$. The the ckt is same only
NAND is replaced by NOR.



and,

$$\begin{aligned} \overline{A+B} + \overline{C} \cdot (\overline{A \oplus B}) &= (\overline{A+B}) \cdot \overline{\overline{C}} \cdot (\overline{A \oplus B}) \\ &= (A+B) \cdot (C + \overline{A \oplus B}) \\ &= AC + A\bar{A}\bar{B} + AAB + BC + B\bar{A}\bar{B} + BAB \\ &= AC + AB + BC + AB \\ &= C(A+B) + AB \\ \text{CARRY} &= AB + BC + CA \end{aligned}$$

48

PARALLEL ADDER :-

There are three type of adder

1. Serial adder (we write with sequential ckt)
2. Parallel adder.
3. Look ahead carry adder.

(49)

⇒ In serial adder only one Full adder (FA) is used to add group of bits.

⇒ It is slowest adder.

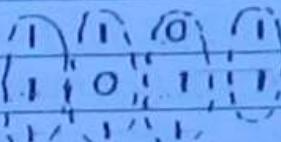
Parallel adder :-

⇒ For 4 bit adder

:- 3 FA and 1 HA required.

:- or 4 FA is required.

FA FA FA HA



1 1 0 0 0 0

⇒ Parallel adder is used to add group of bits.

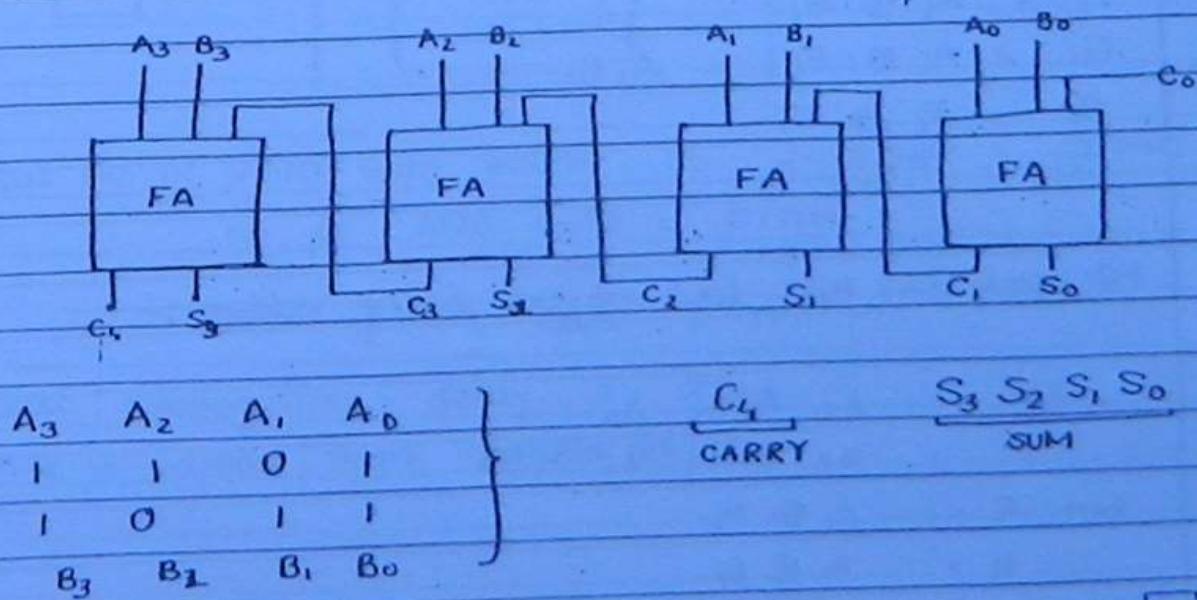
⇒ To add two N bit no. it requires $(N-1)$ Full adder and 1 Half adder. or,

N Full adder or,

$(2N-1)$ Half adder and $N-1$ OR gates required.

Now,

Diagram of Parallel adder :-

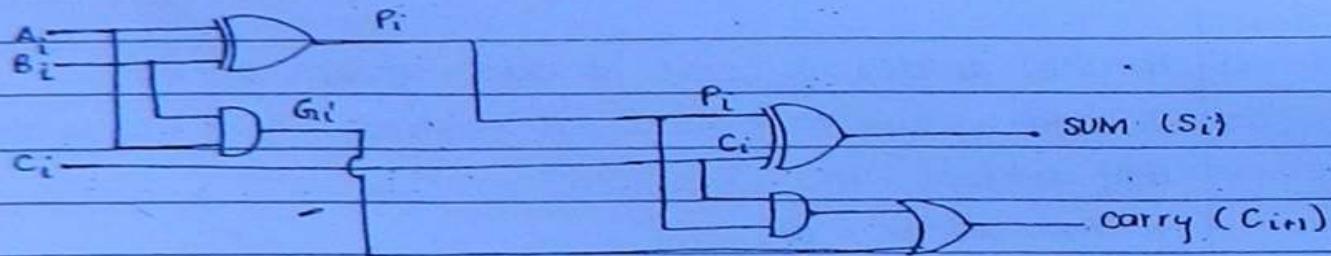


- ⇒ Parallel adder is also called Ripple carry adder.
- ⇒ Propagation delay from I/P array to O/P array. Hence it is also known as Ripple carry adder. (50)
- ⇒ In parallel adder each FA will provide 2 logic gate delay
- In n bit parallel adder provide total delay of.

$$T_{\text{delay}} = 2ntpd$$

LOOK AHEAD CARRY CIRCUIT :-

- ⇒ Disadvantage of parallel adder is carry propagation delay present.
- ⇒ As no of bit increases - speed of operation reduced.
- ⇒ To avoid this look ahead carry adder is used.



P_i = Propagation

G_i = Generation term

$$P_i = A_i \oplus B_i$$

$$G_i = A_i \otimes B_i = A_i B_i$$

$$S_i = P_i \oplus C_i$$

$$C_{i+1} = P_i C_i + G_i$$

For four (4) bit look ahead carry adder :-

I/P.	A ₃	A ₂	A ₁	A ₀
	B ₃	B ₂	B ₁	B ₀

Then P₀ = A₀ ⊕ B₀

P₁ = A₁ ⊕ B₁

P₂ = A₂ ⊕ B₂

P₃ = A₃ ⊕ B₃

$$G_0 = A_0 B_0$$

$$S_0 = P_0 \oplus G_0$$

$$G_1 = A_1 B_1$$

$$S_1 = P_1 \oplus G_1$$

$$G_2 = A_2 B_2$$

$$S_2 = P_2 \oplus G_2$$

$$G_3 = A_3 B_3$$

$$S_3 = P_3 \oplus G_3$$

(57)

$$C_{i+1} = P_i C_i + G_i$$

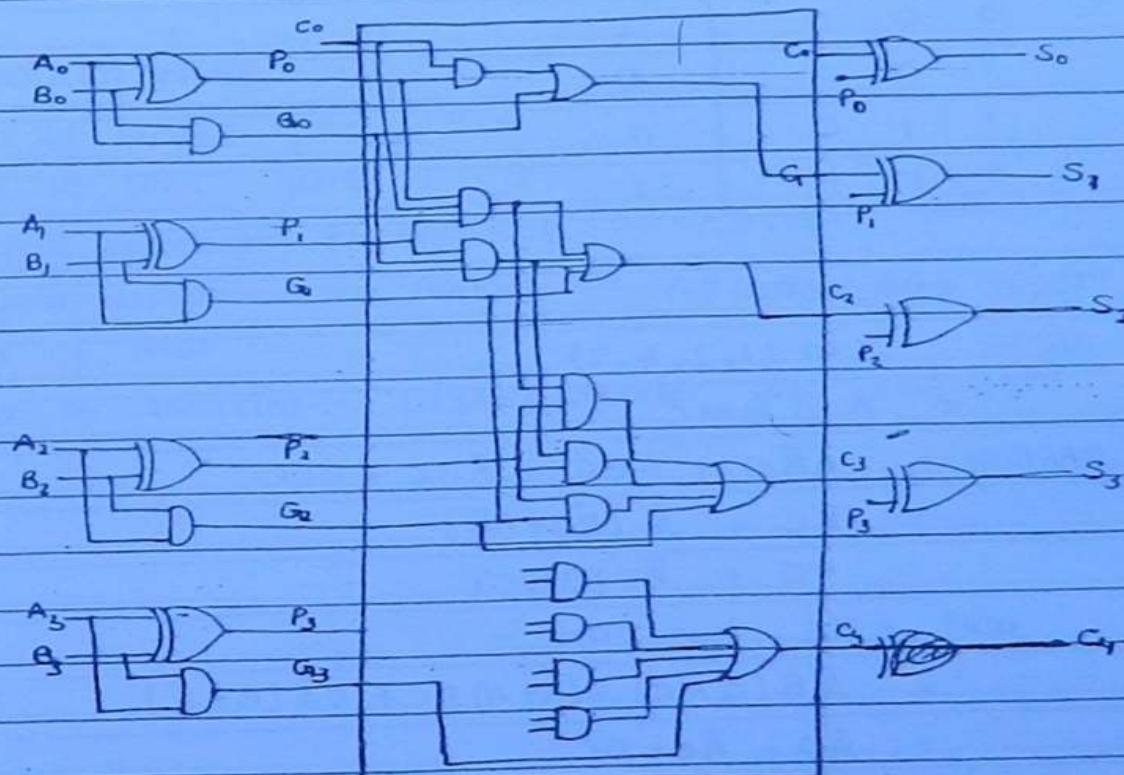
look ahead carry generator expression.

$$C_1 = P_0 C_0 + G_0$$

$$C_2 = P_1 C_1 + G_1 = P_1 (P_0 C_0 + G_0) + G_1 = P_1 P_0 C_0 + P_1 G_0 + G_1$$

$$C_3 = P_2 C_2 + G_2 = P_2 P_1 P_0 C_0 + P_2 P_1 G_0 + P_2 G_1 + G_2$$

$$C_4 = P_3 C_3 + G_3 = P_3 P_2 P_1 P_0 C_0 + P_3 P_2 P_1 G_0 + P_3 P_2 G_1 + P_3 G_2 + G_3$$



$$\Rightarrow \text{Total no. of AND Gate inside} = 1+2+3+4 = \frac{n(n+1)}{2} = \frac{4 \times 5}{2}$$

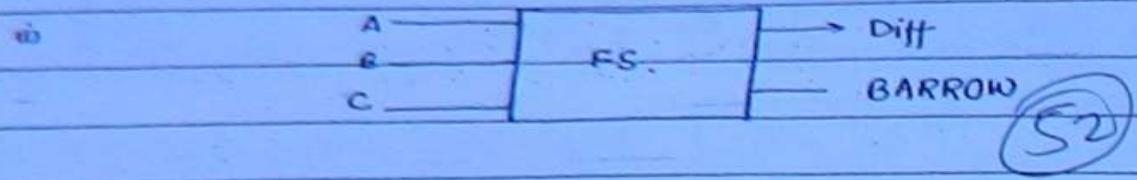
$$\text{no. of AND Gate} = \frac{n(n+1)}{2}$$

$$\text{no. of OR Gate} = n.$$

$$\Rightarrow \text{Total propagation delay} = 2t_{pd}$$

\Rightarrow This is faster than parallel adder.

FULL SUBTRACTOR :-



(v) Truth table:-

A	B	C	Diff. (A-B-C)	BARROW
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

(vi) logic expression :-

$$\text{Diff.} := \sum m(1, 2, 4, 7)$$

$$= A \oplus B \oplus C$$

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

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

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

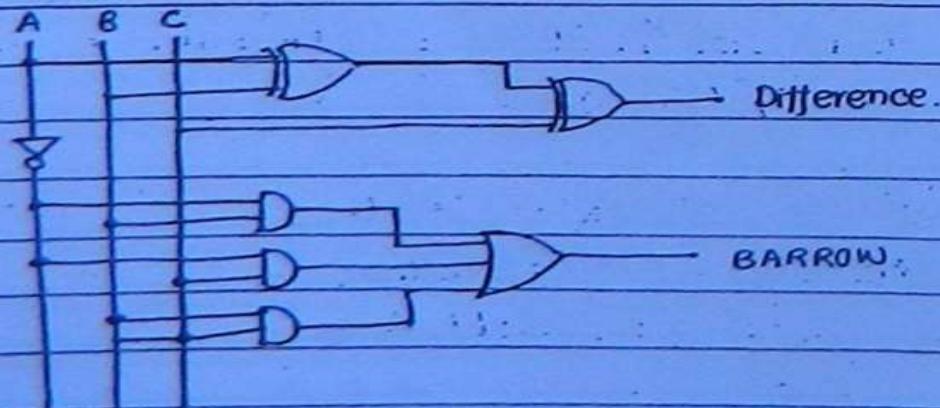
add $\bar{A}BC$ two more :-

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

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

$$= \sum m(1, 2, 3, 7)$$

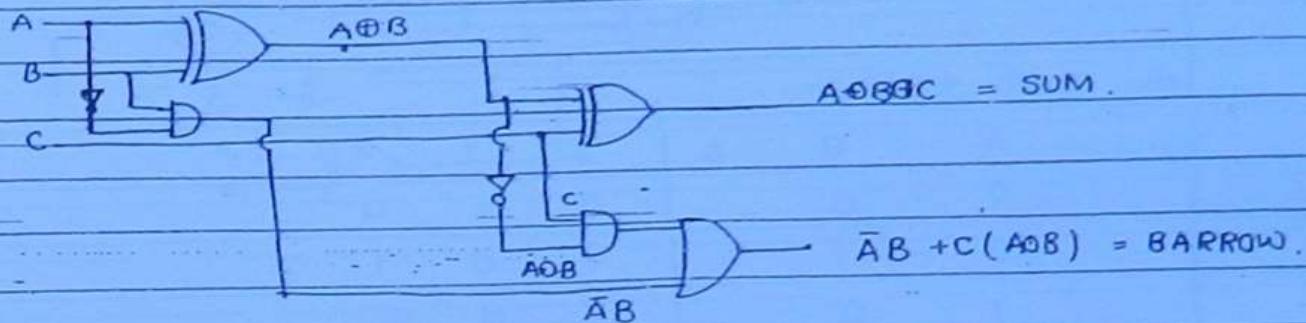
(vii) Implementation :-



Barrow expression :-

$$\begin{aligned} & \bar{A}\bar{B}C + \bar{A}B\bar{C} + \bar{A}\bar{B}\bar{C} + ABC \\ &= \bar{A}B(C + \bar{C}) + C(\bar{A}\bar{B} + AB) \\ &= \bar{A}B + C(A\oplus B) \end{aligned}$$

(S3)



⇒ Full subtractor will be implemented with 2-HS and 1 OR Gate.

Important Ques:-

no. of NAND gate = 9

no. of NOR gate = 9

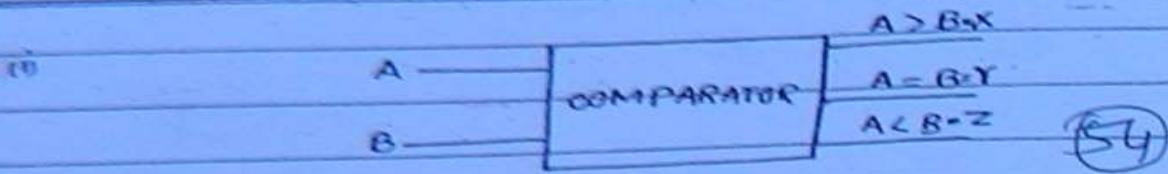
logical expression for Difference = $A \oplus B \oplus C$

logical expression for Barrow = $\bar{A}B + \bar{A}C + BC$ or, $\bar{A}B + C(A \oplus B)$

no. of MUX :-

no. of Decoder :- 1 (3x8) Decoder and 2 OR gate

COMPARATOR :-



(ii) Truth table :-

A	B	C	X	Y	Z
0	0	0	0	1	0
0	1	0	0	0	1
1	0	0	1	0	0
1	1	1	0	1	0

X	Y	Z
$= AB$	$= A \oplus B$	$= \bar{A}B$

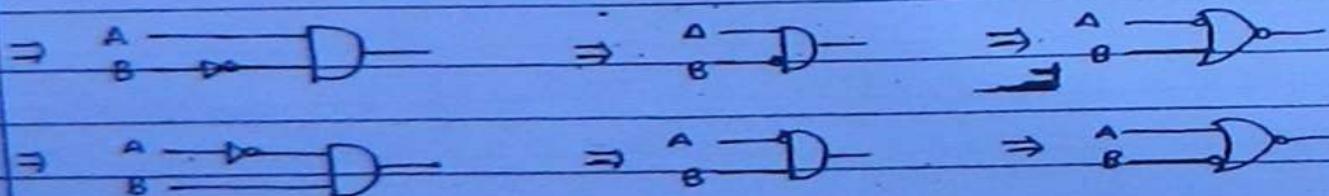
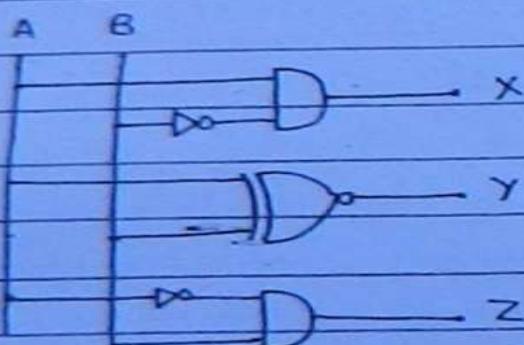
(iii) Expression :-

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

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

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

(iv) Implementation :-



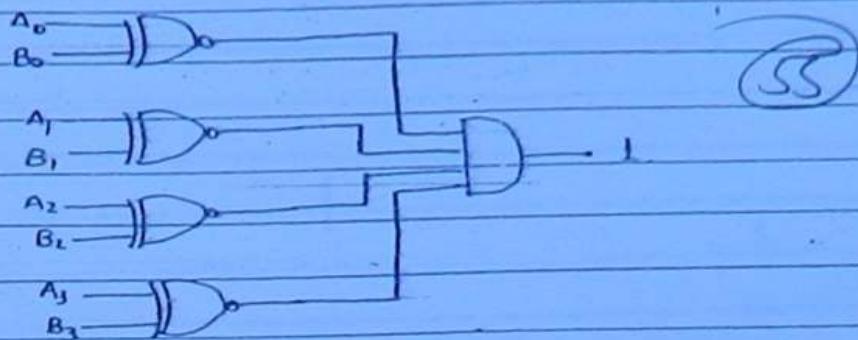
Note: for equality condition $A \oplus B$. condition holds.

\Rightarrow If $A_3 A_2 A_1 A_0$ are equal to $B_3 B_2 B_1 B_0$

Then the equality condition is.

$$(A_3 \oplus B_3) (A_2 \oplus B_2) (A_1 \oplus B_1) (A_0 \oplus B_0)$$

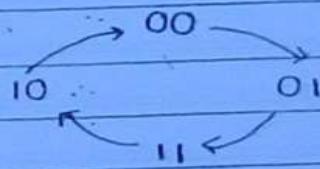
Then the CKt is :-



K-MAP :-

- It is used when o/p is 0, 1, and X (don't care)
- In K-MAP gray code representation is used.
- K-map is graphical representation

⇒ Gray code representation



⇒ each successive term is changed by only one bit.

Two variable:- A		MSB	LSB	
		B		
0	0	0	0	0
0	1	0	1	1
1	0	1	0	2
1	1	1	1	3
		10	11	

(For Two variable)

For Three variable:-

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

(A)

(57)

Four variable:-

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

 \Rightarrow Minimize :-

Q:- $f(A, B) = \sum m(0, 2, 3)$

(S8)

Sol:-

A\B	0	1
0	1	
1	1	1

$f(A, B) = A + \bar{B}$ (+ is put due to SOP form)

Q:- $f(A, B) = \sum m(1, 2, 3)$

A\B	0	1
0	1	1
1	1	1

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

Q:- $f(A, B) = \sum m(0, 1, 2, 3)$

Sol:- $f(A, B) = 1$.

\Rightarrow In K-map if all are one means the function is 1.

Q:- ~~$f(A, B) = \sum m(1, 3) + \sum d(2)$~~

A\B	X	111
X	X	111

$f(A, B) = B$ (no need of any gate)

Q:- $f(A, B) = \sum m(0, 3) + \sum d(2)$

Sol:-

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

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

(59)

Q:- $f(A, B) = \sum (0, 3) + \sum d(2, 1)$

Sol:-

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

$$f(A, B) = 1.$$

⇒ In SOP form if all are 1's means o/p is 1.

⇒ All are don't care means don't care.

Three variable :-

Q:- $f(A, B, C) = \sum m(1, 3, 5, 7)$

Sol:-

A	BC	00	01	11	10
0		1 1	1 1	1 1	
1		1 1	1 1	1 1	

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

Q:- $f(A, B, C) = \sum m(0, 1, 3, 6)$

Sol:-

A	BC	$\bar{B}\bar{C}$	$\bar{B}C$	$B\bar{C}$	BC
0	\bar{A}	1 1	1 1	1 1	1 1
1	A				1 1

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

Q:- $f(A, B, C) = \sum m(1, 3, 6, 7)$

Sol:-

A	BC	$\bar{B}\bar{C}$	$\bar{B}C$	$B\bar{C}$	BC
0	\bar{A}	1 1	1 1	1 1	1 1
1	A				1 1

⇒ if we take BC then it is redundant term and it must be removed.

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

Procedure :-

- i) Octets
- ii) Quads
- iii) Pairs
- iv) Single term
- v) Remove redundant

Priority.

Q:- $f(A, B, C) = \Sigma m(0, 1, 2, 4, 7)$

(60)

Sol:-

A	BC	$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$
\bar{A}	1	1	1	1	1
A	1	1	1	1	1

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

Q:- $f(A, B, C) = \Sigma m(0, 1, 5, 6, 7)$

Sol:-

A	BC	$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$
\bar{A}	1	1	1	1	1
A	1	1	1	1	1

$$\begin{aligned} f(A, B, C) &= \bar{A}\bar{B} + \bar{B}\bar{C} + AB \\ &= \bar{A}\bar{B} + AB + AC \end{aligned} \quad \left. \begin{array}{l} \text{two sol'} \\ - \end{array} \right.$$

\Rightarrow K-map provide minimize expression but not necessarily unique. i.e two sol' also.

Q:- $f(A, B, C) = \Sigma m(0, 1, 2, 5, 7) + \Sigma d(3, 6)$

Sol:-

A	BC	$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$
\bar{A}	1	1	1	X	1
A	1	1	1	1	X

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

Q:- $f(A, B, C) = \Sigma m(0, 1, 6, 7) + \Sigma d(3, 5)$

Sol:-

	$\bar{B}C$	$B\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$
\bar{A}	1	1	1	X	
A		X	1	1	1

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

(6)

Q:- $f(A, B, C) = \sum m(0, 1, 6, 7) + \sum d(3, 4, 5)$

Sol:-

	$\bar{B}C$	$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$
\bar{A}	1	1	1	X	
A	*	*	1	1	1

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

Four Variable :-

Q:- $f(A, B, C, D) = \sum m(0, 1, 3, 5, 7, 8, 9, 11, 13, 15)$

Sol:-

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

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

Q:- $f(A, B, C, D) = \sum m(0, 1, 4, 5, 8, 9, 13, 15)$

Sol:-

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

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

Q:- $f(A, B, C, D) = \sum m(0, 2, 8, 10, 14) + \sum d(5, 15)$

Sol:-

$\bar{A}B$	$\bar{C}\bar{D}$	$\bar{C}D$	$C\bar{D}$	CD
1				1
	*		*	
		*		1
1	.	.	1	1

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

(62)

POS (Product of Sum) :-

Simplify :-

Q:- $f(A, B) = \pi M(0, 2, 3)$

$A \backslash B$	$\bar{B}, 0$	$\bar{B}, 1$
0, A	1	
1, A	1	0

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

Q:- $f(A, B) = \pi M(0, 3) + \pi d(1)$

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

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

Q:- $f(A, B, C) = \pi M(0, 1, 3, 5, 7)$

$A \backslash B+C$	$B+C$	$B+\bar{C}$	$\bar{B}+\bar{C}$	$\bar{B}+C$	$B+C$
A 0	1	1	1	1	
\bar{A} 1		1	1	1	

$$\begin{aligned} f(A, B, C) &= \bar{C} + (A+B) \\ &= \bar{C}(A+B) \end{aligned}$$

Date: 2003

Q:- for the K-map minimize POS expression is.

Sol:-

		$B+C$	$B+\bar{C}$	$\bar{B}+\bar{C}$	$\bar{B}+C$	
		A	'0'	x	'x'	1
		\bar{A}	'1'	1	'0'	x

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

(63)

=> The two function are same if the position of 1's are 0's are same in K-map. and if the 1's place 0 are placed and at 0's place 1's are placed then the function is complementary to each other.

=> Problem - 26 - page - 13

Q:- $w = R + \bar{P}a + \bar{R}s$ $x = P\bar{a}\bar{R}\bar{s} + \bar{P}\bar{a}\bar{R}s + P\bar{a}Rs$

		$\bar{R}\bar{s}$	$\bar{R}s$	Rs	$R\bar{s}$	
		$\bar{P}\bar{a}$	-	1	1	1
		$\bar{P}a$	1	1	1	1
		$P\bar{a}$	-	1	1	1

		$\bar{R}\bar{s}$	$\bar{R}s$	Rs	$R\bar{s}$	
		$\bar{P}\bar{a}$	1	-	-	-
		$\bar{P}a$	-	1	-	-
		$P\bar{a}$	-	-	1	-

$$Y = RS + PR + P\bar{E} + \bar{P}Q \quad Z = R + S + P\bar{E} + \bar{P}\bar{E}\bar{R}' + P\bar{Q}\bar{S}$$

		$\bar{R}\bar{s}$	$\bar{R}s$	Rs	$R\bar{s}$	
		$\bar{P}\bar{a}$	0	0	0	1
		$\bar{P}a$	1	1	1	1
		$P\bar{a}$	-	1	0	0

		$\bar{R}\bar{s}$	$\bar{R}s$	Rs	$R\bar{s}$	
		$\bar{P}\bar{a}$	0	0	1	1
		$\bar{P}a$	1	1	1	1
		$P\bar{a}$	-	1	0	0

$$\Rightarrow \text{Then } w = z = \bar{x}$$

Note

⇒ If Truth table is :-

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

(The method for writing
the expression from
Truth table).

(64)

Then,

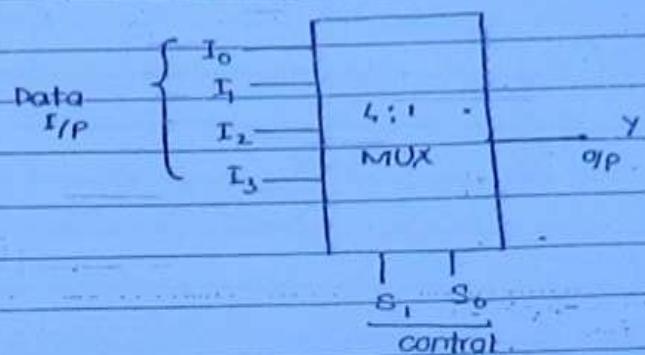
$$Y = \bar{A}\bar{B} \cdot 0 + \bar{A}B \cdot 1 + A\bar{B} \cdot 0 + AB \cdot C$$

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

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

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

→ Many I/P and one O/P.



→ It is combinational circuits.

→ Depending on control or select I/P, one of the I/P is transferred to the o/p line.

→ It is select I/P then also called as Data selector, or.

Many to one ckt or, universal logic ckt or, parallel to serial sckt.

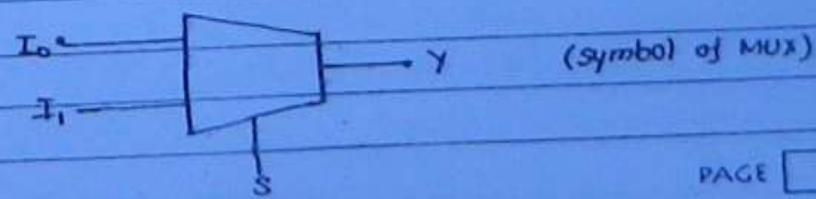
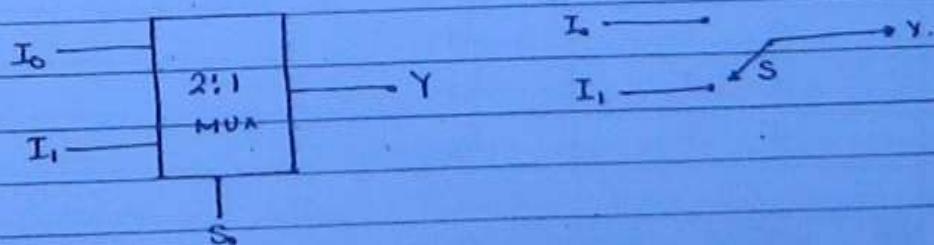
$$\rightarrow \boxed{m = 2^n}$$

or, $n = \log_2 m$

where m = no. of data I/P.

n = no. of select I/P. (control I/P).

2:1. MUX :-



iii) Truth table :-

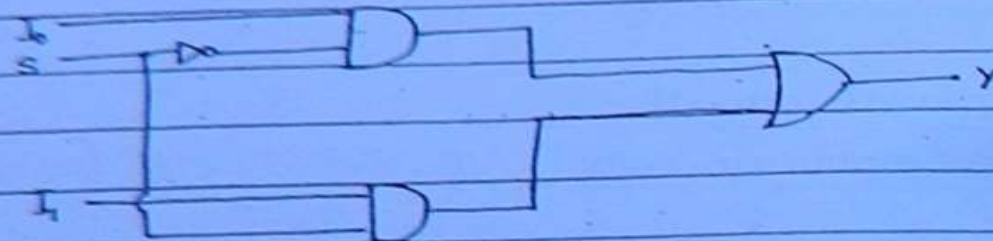
S	Y
0	I_0
1	I_1

iv) logical expression :-

$$Y = \bar{S}I_0 + S I_1$$

(66)

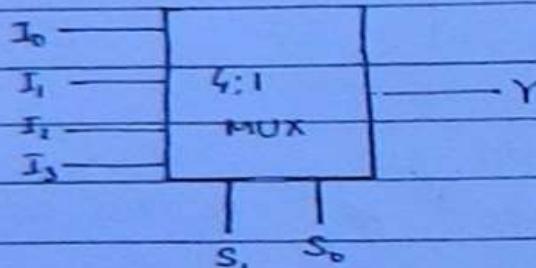
v) Implementation :-



⇒ In MUX, generally AND gate followed by OR gate.

4:1 MUX :-

vi)

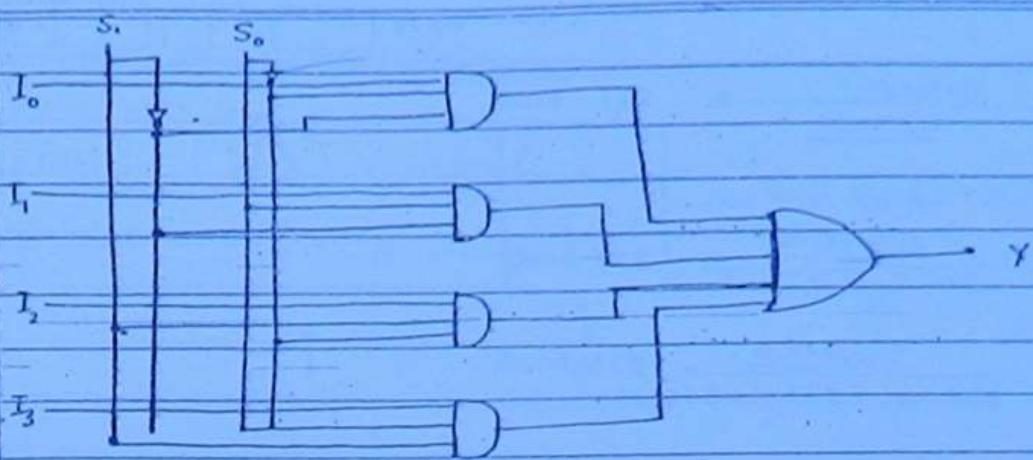


vii) Truth table :-

S_{01}	S_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

viii) logical expression :-

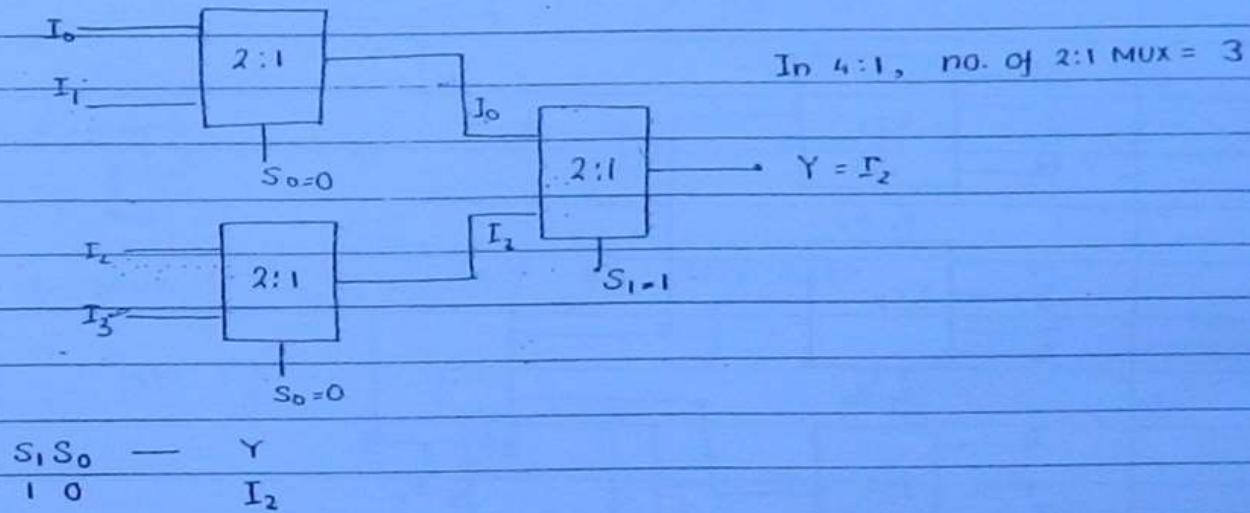
$$Y = S_1 S_0 I_0 + S_0 S_1 I_1 + S_1 S_0 I_2 + S_0 S_1 I_3$$



(67)

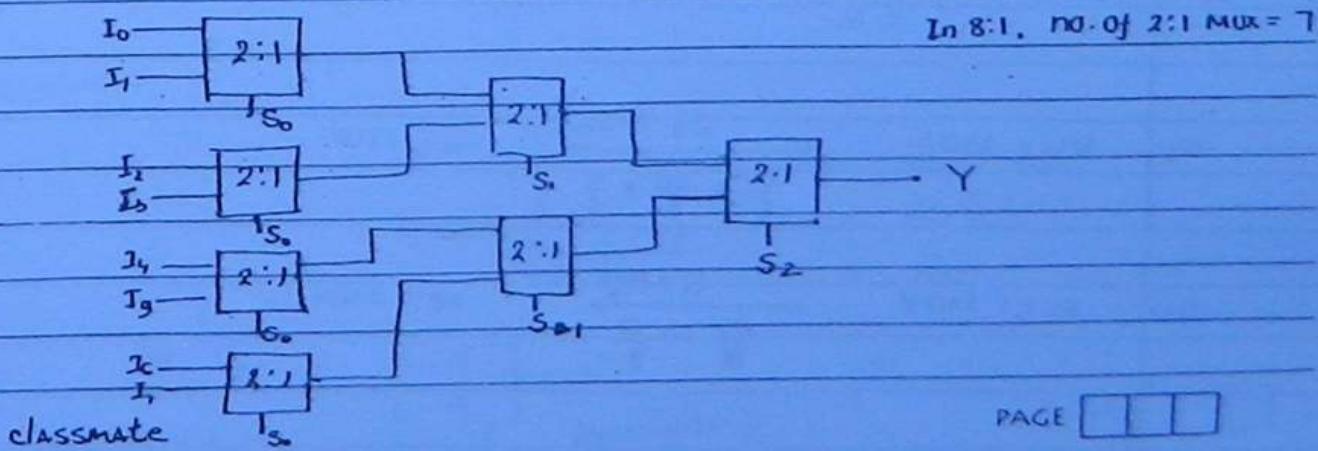
I. Implementation of higher order MUX with lower order:-

(A) Implement 4:1 MUX using 2:1 mux:-



(B) Implement 8:1 MUX using 2:1 MUX:-

$$\text{no. of MUX} = \frac{8}{2} + \frac{4}{2} + \frac{2}{2} = 4 + 2 + 1 = 7$$



(C) 16×1 8×1 MUX $\rightarrow 2 \times 1$ MUX

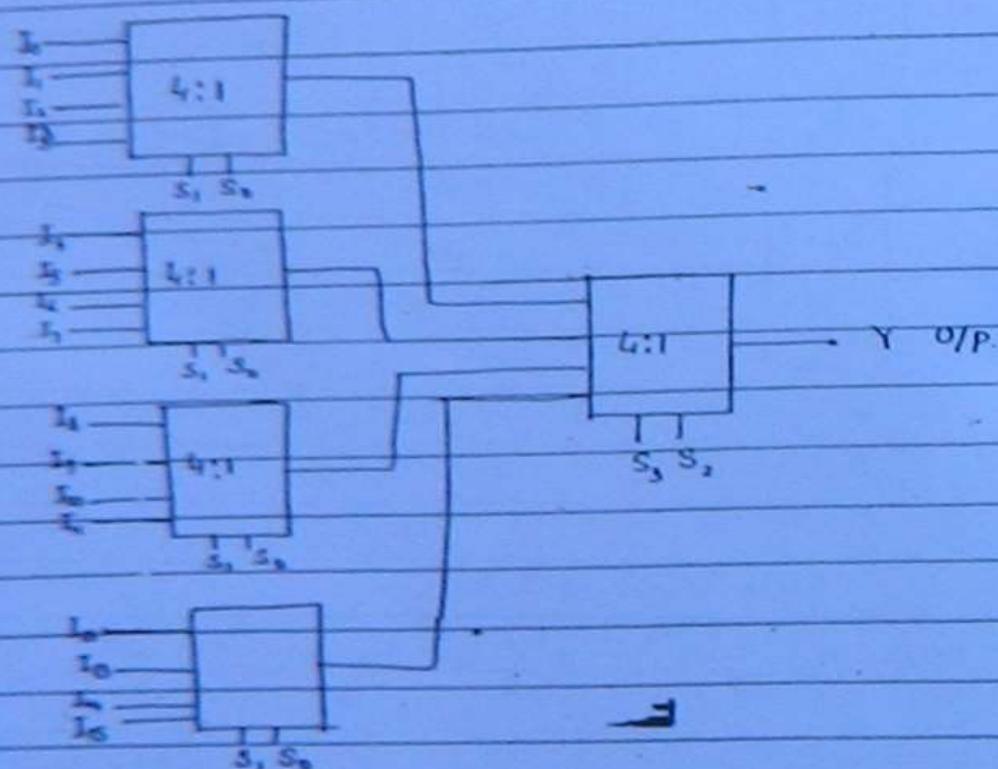
68

(D) 64×1 8×8 MUX $\rightarrow 2 \times 1$ MUX(E) 256×1 $(2 \times 8) \times (2 \times 8)$ MUX $\rightarrow 2 \times 1$ MUX

\Rightarrow Therefore for $2^n \times 1$ MUX the no. of $2^1 \times 1$ mux is $2^n - 1$.
MUX required.

(F) 16×1 MUX from 4×1 MUX :-

$$\text{no. of MUX} = \frac{16}{4} + \frac{4}{4} = 4 + 1 = 5$$

(G) 8×1 MUX $\rightarrow 2 \times 1$ MUX $\rightarrow 4 \times 1$ MUX

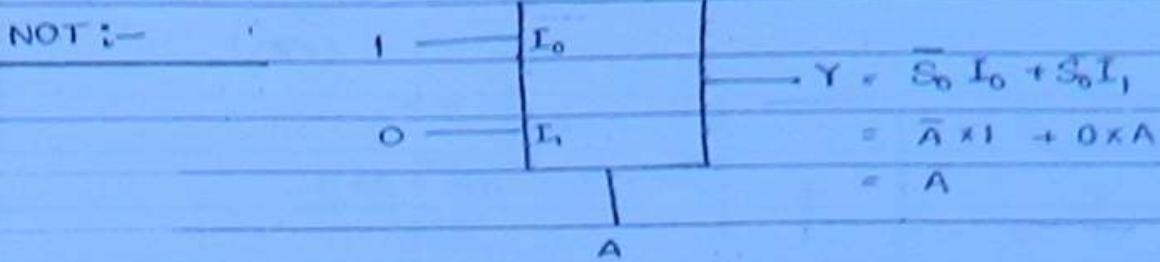
$$\frac{64}{4} + \frac{16}{4} + \frac{4}{1}$$

(H) 64×1 MUX $\leftarrow 8 \times 1$ MUX $\rightarrow 64:1$ MUX

$$\frac{64}{8} + \frac{8}{8}$$

II MUX AS UNIVERSAL :-

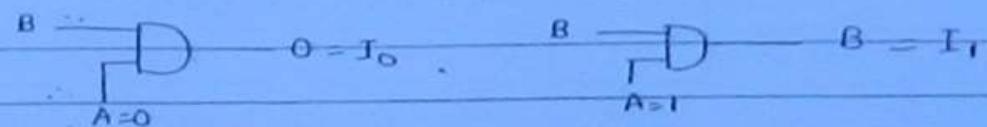
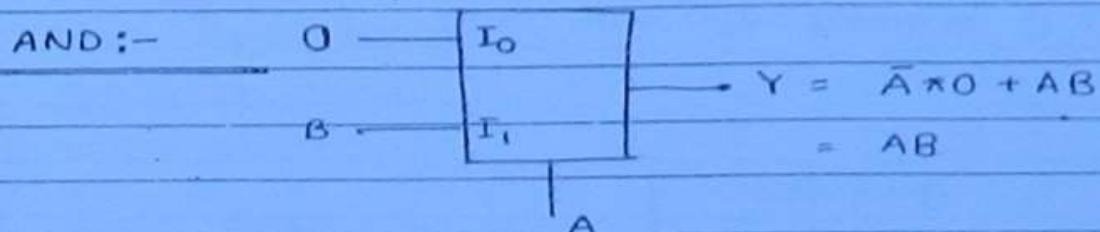
2:1 MUX :-



(69)

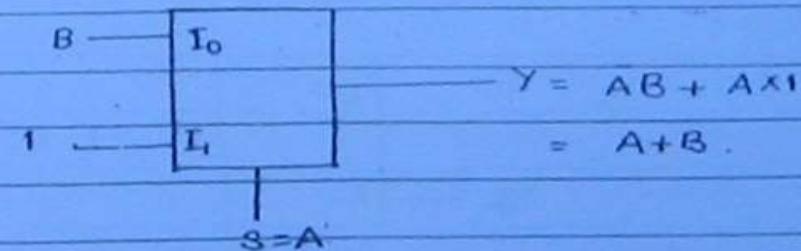
A	Y
0	1
1	0

\Rightarrow 1-MUX is required for NOT Gate.



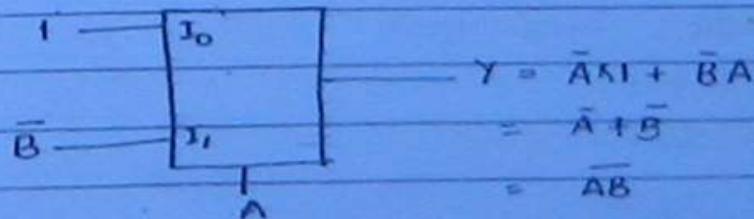
\Rightarrow 1-MUX (2:1) is required for AND gate.

OR :-



\Rightarrow 1-MUX (2:1) is required for OR gate.

NAND :-



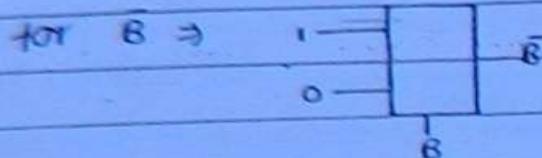
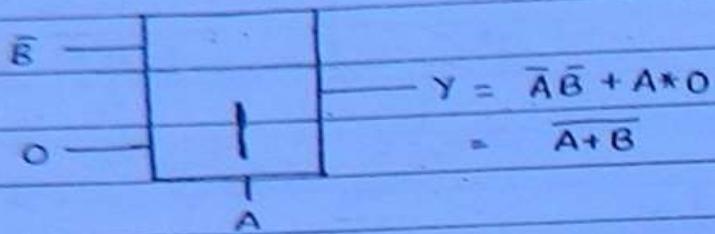
for \bar{B} :-



→ 2-MUX required for NAND gate.

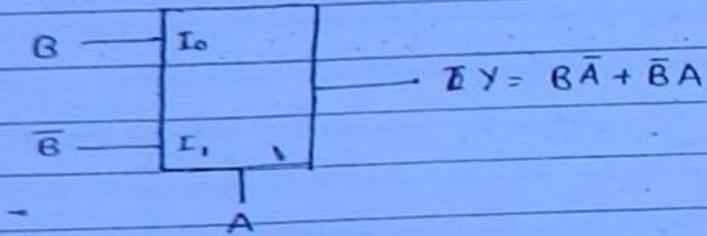
(70)

NOR :-



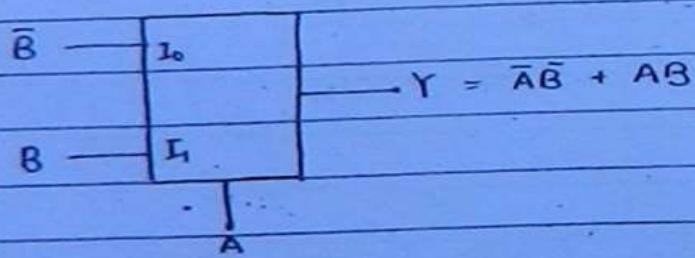
⇒ 2 MUX (2:1) required for NOR gate.

EXOR :-



⇒ 2 MUX (2:1) required for EXOR gate.

EXNOR :-



⇒ 2 MUX (2:1) required for EXNOR gate.

Ques: EXOR, AND gate required 2x1 MUX.

(a) 1,1

~~(b)~~ 1,1

(c) 1, 2

(d) 2,2

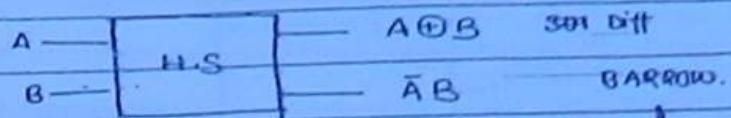
Sol:-

EXOR = 2 , AND = 1

⇒ for HA - 3 MUX required (2:1)

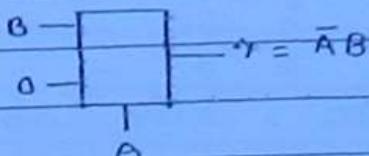
⇒ for HS - 3 MUX required (2:1)

(77)



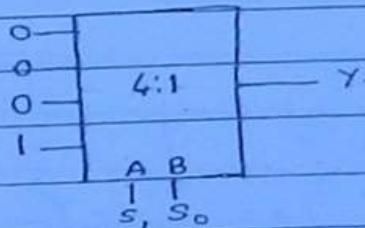
$A \oplus B = 2$ MUX required qnd.

$\bar{A}B = 1$ MUX.

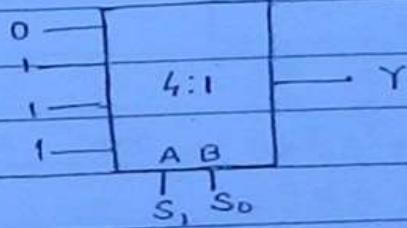


★ 4:1 MUX :-

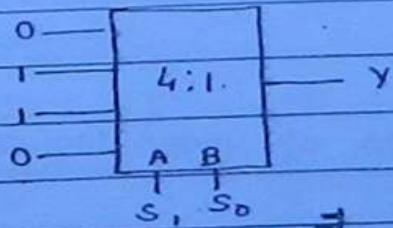
AND :-



OR :-

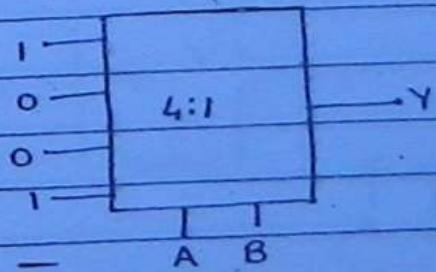


EXOR :-



⇒ Any two variable function is implemented with 4:1 MUX.

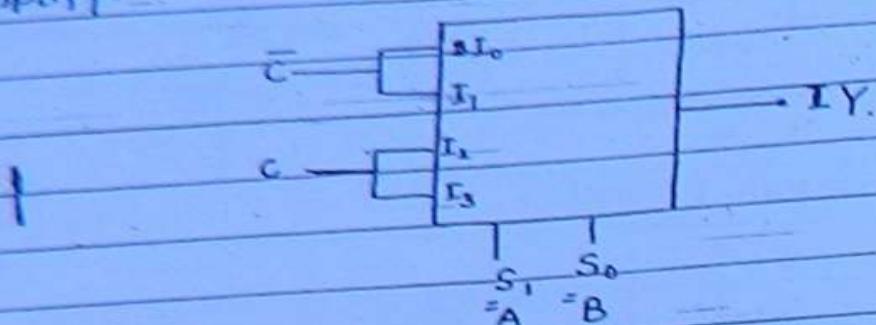
EXNOR :-



III. Determine minimize o/p logical expression:-

Simplify:-

Q:-



72

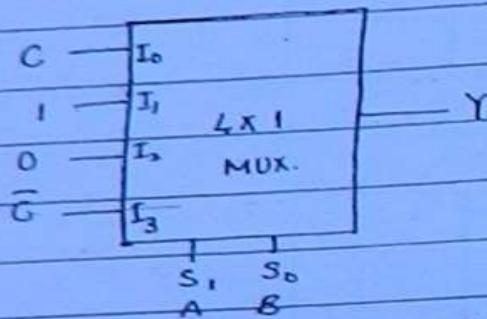
$$\text{Sol: } Y = \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + A\bar{B}C + ABC$$

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

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

$$= A \oplus C$$

Q:-



Sol: -

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

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

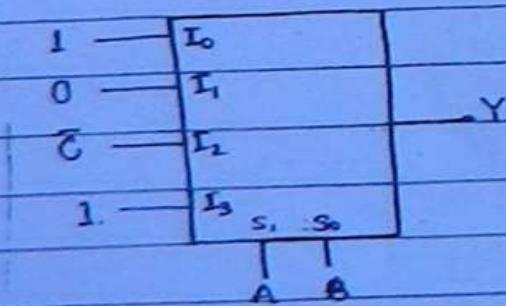
$$= \bar{A}\bar{C} + B\bar{C}$$

A	$\bar{B}C$	$\bar{B}\bar{C}$	$\bar{B}C$	$B\bar{C}$	BC
\bar{A}	1	1	0	0	0
A	0	0	1	1	1

IV. Implementation of given logical expression:-

Q:- $f(A, B, C) = \sum m(0, 1, 4, 6, 7)$

Sol: -



	I_0	I_1	I_2	I_3
\bar{C}	①	2	④	⑥
C	①	3	5	⑦

1. 0 \bar{C} 1.

(23)

A	B	C	
0	0	0	$0 \rightarrow \bar{C} \quad B \bar{A}$
0	0	1	$1 \rightarrow C \quad \bar{B} A$
0	1	0	$0 \rightarrow \bar{C} \quad B \bar{A}$
0	1	1	$1 \rightarrow C \quad B \bar{A}$

1	0	0	$0 \rightarrow \bar{C} \quad \bar{B} A$
1	0	1	$1 \rightarrow C \quad \bar{B} A$
1	1	0	$0 \rightarrow \bar{C} \quad B A$
1	1	1	$1 \rightarrow C \quad B A$

Q:- Implement logical expression.

$$f(A, B, C) = (1, 2, 3, 5, 6, 7) m\bar{3}$$

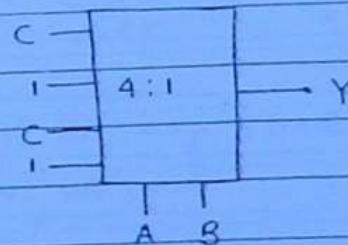
(i) AB as select line

(ii) AC

(iii) BC

Sol:- (i)		I_0	I_1	I_2	I_3
	\bar{C}	0	②	4	⑥
	C	①	-	③	⑤

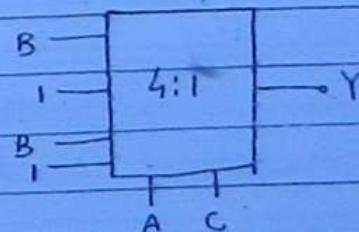
1 C 1 C 1



\Rightarrow 1 - 4:1 MUX required.

(ii)	\bar{AC}	AC	AC	AC
	I_0	I_1	I_2	I_3
	0	①	4	⑤
	②	③	⑥	⑦

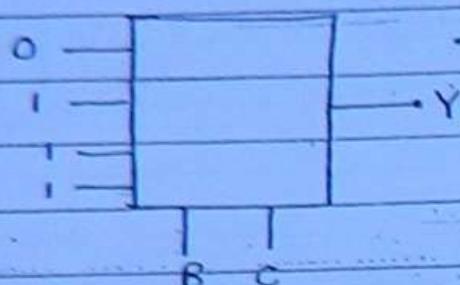
1 B 1 B 1



(Q) EC control :-

	I_0	I_1	I_2	I_3
\bar{A}	0	0	1	1
A	1	0	0	0
	0	1	1	1

74.



⇒ using one 4:1 MUX → Any two variable function implement
→ some of three variable implement.

⇒ one 4x1 MUX and one NOT → All Two → implement
→ All Three

⇒ one 8x1 MUX → all Three
→ Some four.

⇒ one 8x1 MUX and one NOT → All Three are four implement
→ All Four are implemented

JTO-2093

Ques:- $f = \prod M(0, 1, 4, 7)$

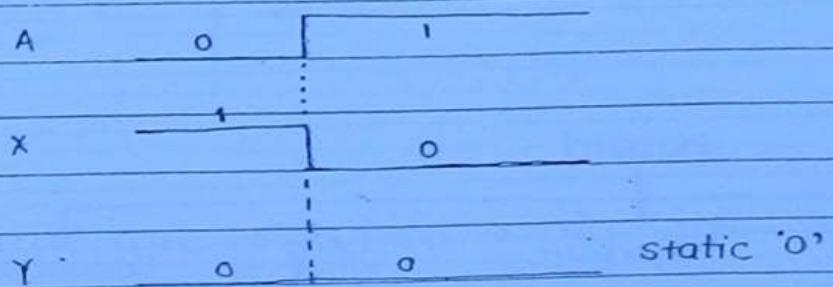
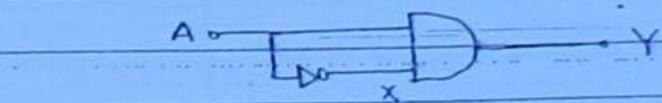
Sol:- first convert it into minterm expression $f = \sum m(2, 3, 5, 6)$

Hazard :-

- ⇒ Hazard occurs due to propagation delay of the logic ckt.
- ⇒ This is unwanted change at the O/P.

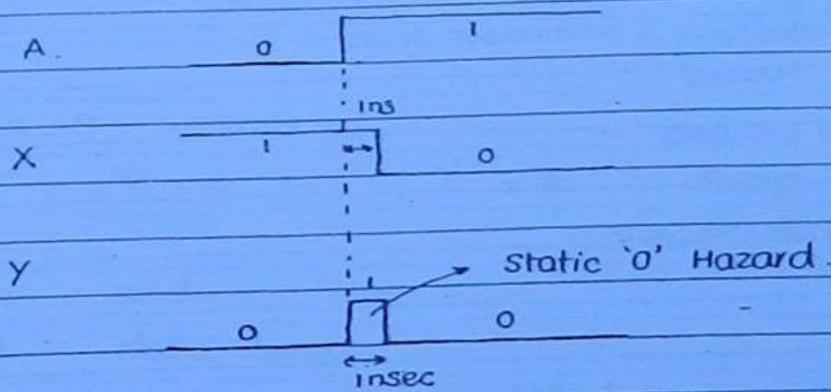
TS

Q) For the given ckt determine O/P waveform when no propagation delay.



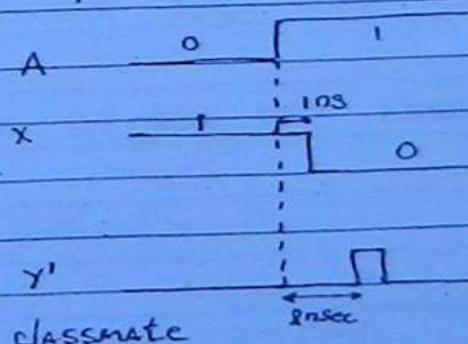
case II :-

If there is propagation delay of 1ns in NOT gate and no delay in AND gate.



case III :-

If $t_{pd}(\text{NOT}) = 1\text{ns}$, $t_{pd}(\text{AND}) = 2\text{ns}$.



Hazard

Static	dynamic	(16) Essential.
⇒ occur in two level ckt	⇒ occur in multilevel ckt	⇒ occurs in Asynchronous sequential ckt
⇒ occur in combinational ckt	⇒ in combinational circuit	
Static Hazard	static Hazard	⇒ Avoid by adding redundant term.
⇒ AND-OR circuit	⇒ OR-AND circuit	
⇒ in SOP FORM	⇒ IN POS FORM	

- ⇒ To avoid static and dynamic Hazard redundant terms are added in combinational ckt.
- ⇒ Essential Hazard :- These Hazards can not be avoided but feels essential.

Memories

Primary

(77)

Secondary

RAM	ROM	Semi random.	Serial access memory
→ Need power.	⇒ Read only	⇒ AT disk	⇒ magnetic tape
⇒ Random access	⇒ Random access	⇒ CD	⇒ Magnetic bubble
⇒ Volatile	⇒ Non volatile	⇒ DVD	⇒ Ferrite core
⇒ Temporary data	⇒ Permanent data	⇒ HD	⇒ CCD (charged couple device)
		BIOS/ System program.	

⇒ Ferrite core → DRO → Discrimitive read only out.

RAM (Random access memory) :-

⇒ Each memory location if m bits are stored then memory capacity = $(2^n \times m)$
with n -bit address - max. no. of memory location required is = 2^n .

⇒ $4K \times 8$ memory.

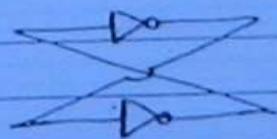
$$= 2^2 \times 2^{10} \times 8 = 2^{12} \times 8$$

= 12 - address line
8 - data lines.

RAM

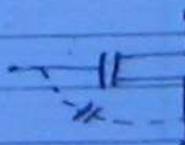
static

1. Stored like FF



Dynamic

1. Data stored in MOS capac.



Memories

Primary

CB

Secondary

RAM	ROM	Semi random.	Serial access memory
⇒ Read /write	⇒ Read only	⇒ All disk	⇒ magnetic tape
⇒ Random access	⇒ Random access	⇒ CD	⇒ Magnetic bubb.
⇒ Volatile	⇒ Non volatile	⇒ DVD	⇒ Ferrite core
⇒ Temporary data	⇒ Permanent data BIOS/ System program.	⇒ HD	⇒ CCD (charged couple device)

⇒ Ferrite core → DRO → Discriptive read only out.

RAM (Random access memory) :-

- ⇒ Each memory location if m bits are stored then memory capacity $(2^n \times m)$
- ⇒ with n -bit address - max. no. of memory location require is $= 2^n$.

⇒ $4K \times 8$ memory.

$$= 2^2 \times 2^{10} \times 8 = 2^{12} \times 8$$

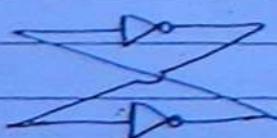
= 12 - address line

8 - data lines.

RAM

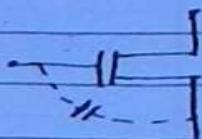
static

1. Stored like FF



Dynamic

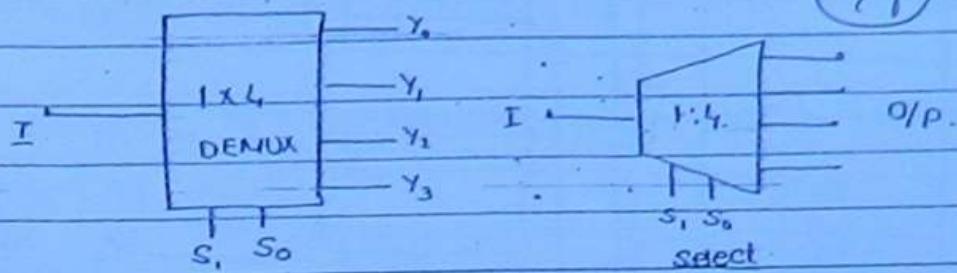
2. Data stored in MOS capac.



DEMUX (DEMULTIPLEXER) :-

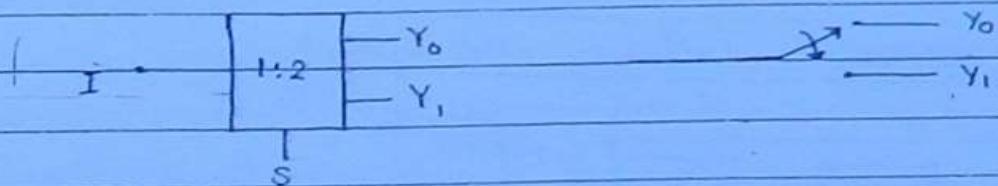
⇒ Single I/P and Many O/P.

(79)



⇒ DEMUX is combinational ckt which have one I/P and many O/P depends on select I/P. I/P is transferred to any of the O/P.

⇒ Also known as 1 to many ckt or, data distributor.

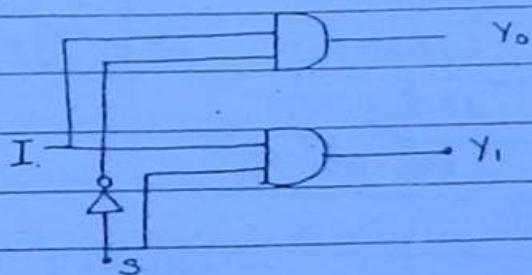


Truth table :-

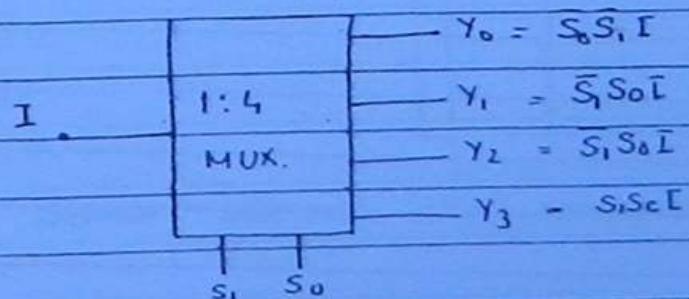
Expression :-

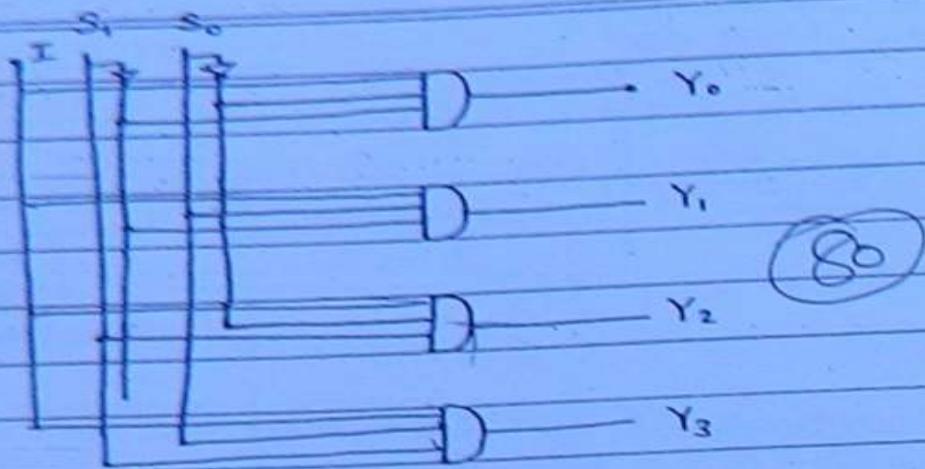
S	Y_1	Y_0	$Y_0 = \bar{S}I$
0	0	I	$Y_1 = SI$
1	I	0	

Implementation :-



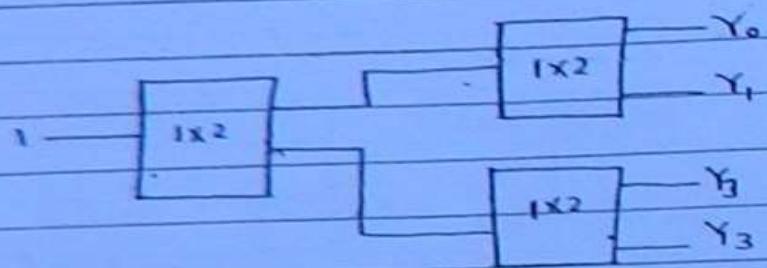
4x4 OR 1:4 DEMUX :-





Implementation of Higher order DEMUX from lower order:

(ii) 1×4 DEMUX \leftrightarrow 1×2 DEMUX



(iii) 1×8 DEMUX \leftrightarrow 1×2 DEMUX

(iv) 1×16 DEMUX \leftrightarrow 1×4 DEMUX

(v) 1×64 DEMUX \leftrightarrow 1×16 DEMUX $\frac{64}{4} + \frac{16}{4} + \frac{4}{4} + 1 \rightarrow 21$

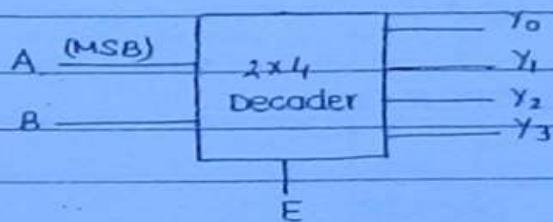
(vi) 1×64 DEMUX \leftrightarrow 1×8 DEMUX

(vii) 1×256 DEMUX \leftrightarrow 1×16 DEMUX

DECODER :-

- ⇒ Decoder is a combinational ckt which have many S/P and m/o/p.
- ⇒ It is used to convert binary data to other code (binary to eg. Binary to octal (3x8)
BCD to Decimal (4x10) (8)
Binary to Hexadecimal
BCD to seven segment
- ⇒ 2 to 4 decoder is minimum possible decoder.

2x4 Decoder :-



Truth table:-

E A B	Y_3	Y_2	Y_1	Y_0
0 x x	0	0	0	0
1 0 0	0	0	0	①
1 0 1	0	0	①	0
1 1 0	0	①	0	0
1 1 1	①	0	0	0

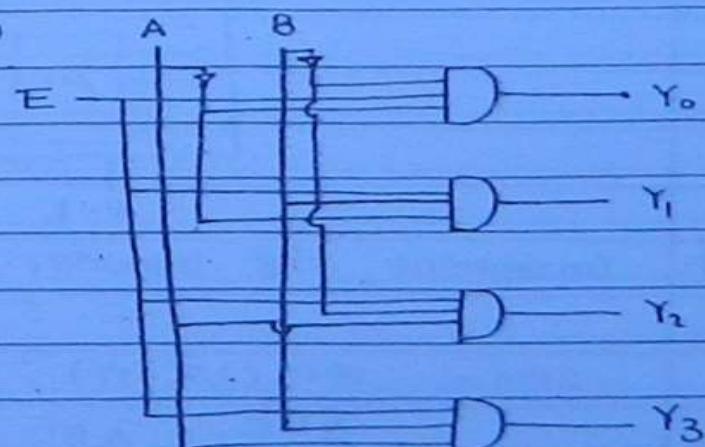
logical Expression :-

$$Y_0 = \bar{A}\bar{B}E$$

$$Y_1 = \bar{A}BE$$

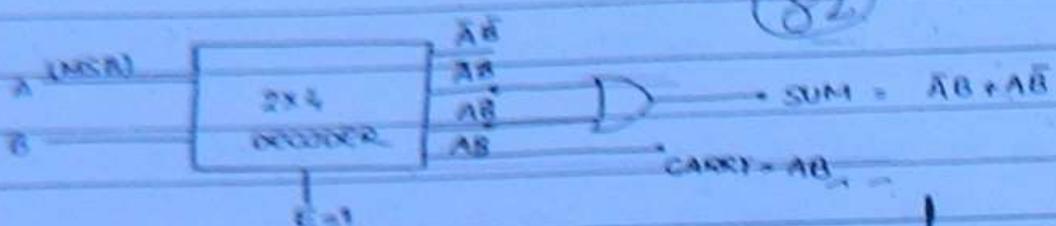
$$Y_2 = A\bar{B}E$$

$$Y_3 = ABE$$

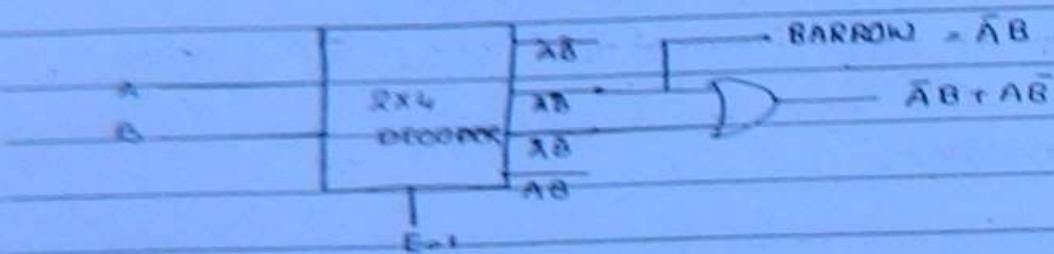


- ⇒ Decoder and DEMUX internal ckt remains sam.

a) Implement Half adder using 2x4 decoder.

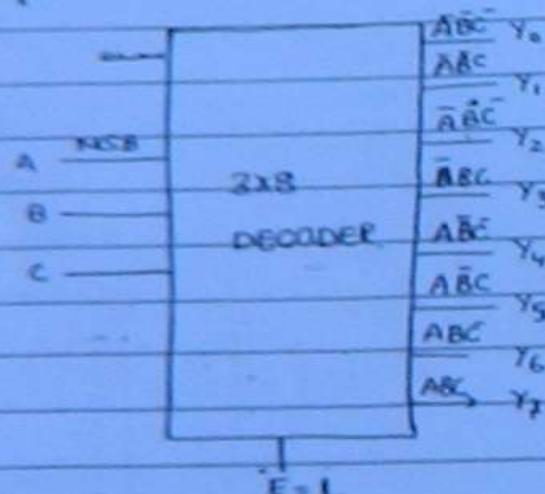


b) we implement HA using 1 - 2x4 decoder and NOR gate and same for MS.



Binary to octal DECODER :-

c) Also called as 3x8 decoder.

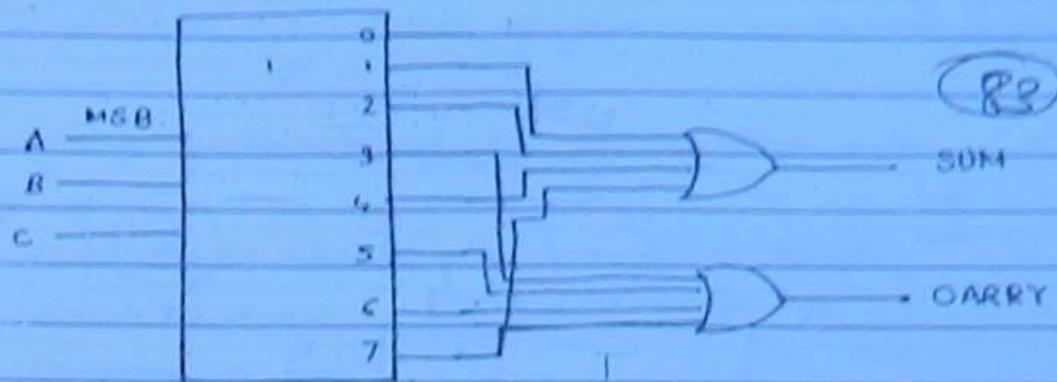


d) Implement 3x8 Decoder make FA.

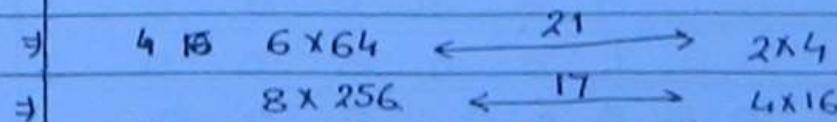
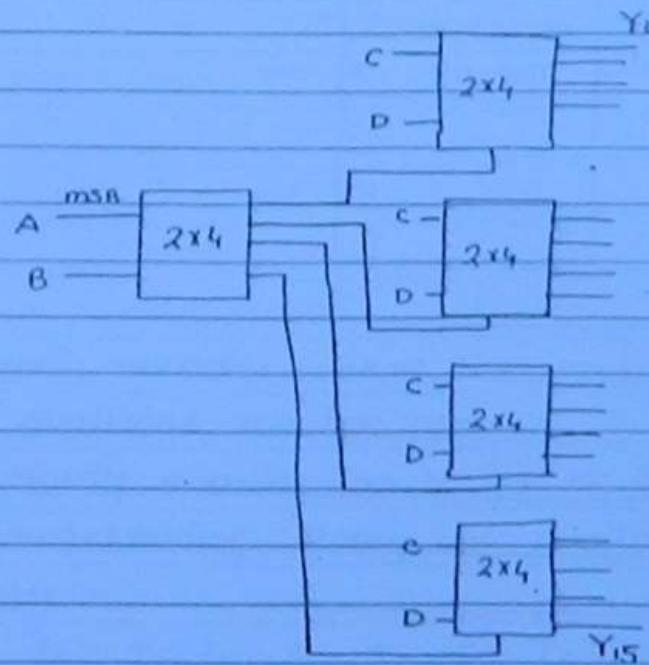
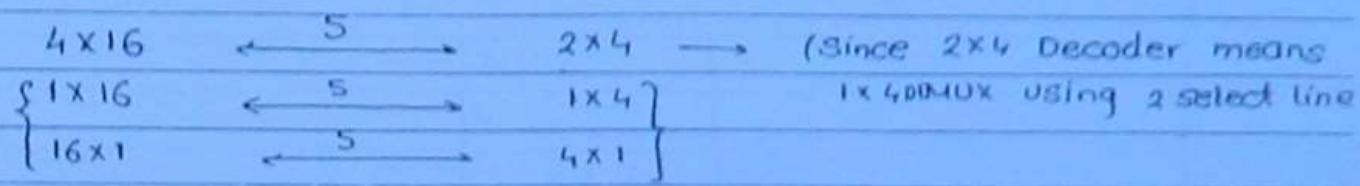
Sol:-

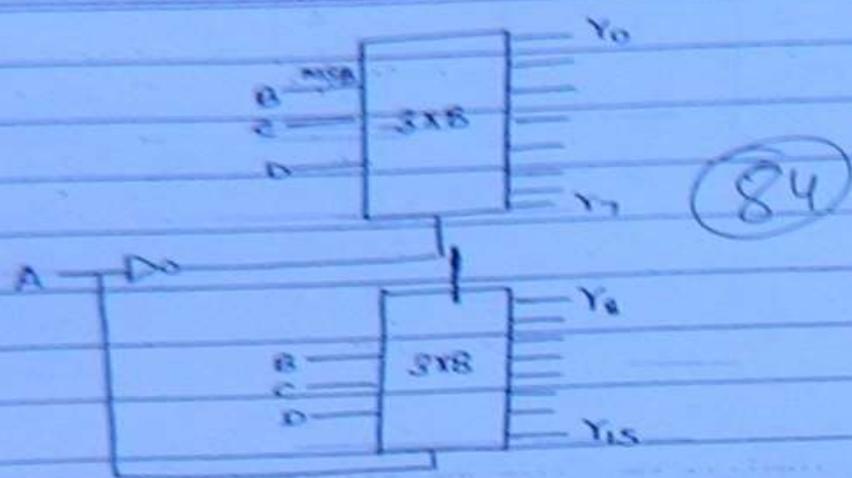
$$\begin{aligned} \text{SUM} &= \Sigma m(1, 2, 4, 7) \\ &= \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + A\bar{B}\bar{C} + ABC \end{aligned}$$

$$\text{CARRY} = \Sigma m(3, 5, 6, 7)$$



Q:- Implementation of Higher order decoder using lower order:-





3

ENCODER :-

⇒ Encoder is the combinational ckt which have many I/P and many O/P.

Encoder is used to convert other code to Binary.

e.g. Octal to Binary

Decimal to BCD

Hexadecimal to Binary

(8)

Octal to Binary Encoder :-

	I_0		
	I_1		
	$I_2 \quad 8 \times 3$	Y_0	
	$I_3 \quad \text{OCTAL to}$	Y_1	
	$I_4 \quad \text{BINARY}$	Y_2	
	I_5		
	I_6		
	I_7		

⇒ In normal encoder one of the I/P line is high and corresponding binary available at the O/P.

⇒ In priority encoder no. of I/P is high. only highest priority no. corresponding binary is available at the O/P.

Truth table :-

I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0	Y_2	Y_1	Y_0
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	1	0	0	1
0	0	0	0	0	0	1	0	0	1	0
0	0	0	0	0	1	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	1	0	0	0	0	1	0	1
0	0	1	0	0	0	0	0	1	0	0
1	0	0	0	0	0	0	0	1	1	1

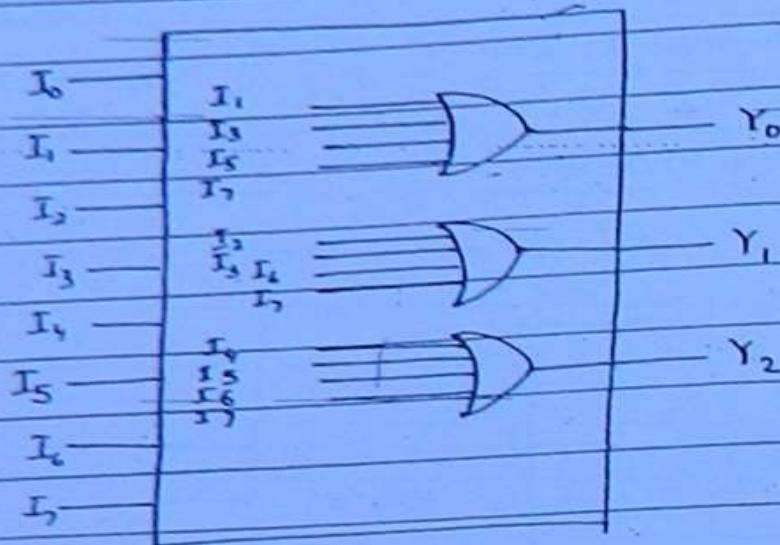
logical expression :-

$$Y_0 = I_1 + I_3 + I_5 + I_7$$

$$Y_1 = I_2 + I_3 + I_6 + I_7$$

$$Y_2 = I_4 + I_5 + I_6 + I_7$$

(86)

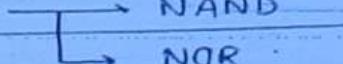


- ⇒ Decoder contains AND Gate.
- ⇒ DEMUX contains AND Gate
- ⇒ ENCODER contains OR Gate.

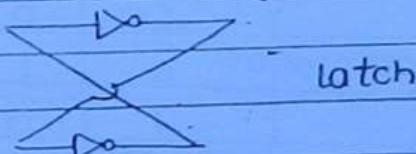
- ⇒ It is basic memory element.
- ⇒ It can store 1 bit.
- ⇒ FF have two o/p which have complemented to each other.
- ⇒ It have two stable state hence it is known as bistable multivibrator.

(87)

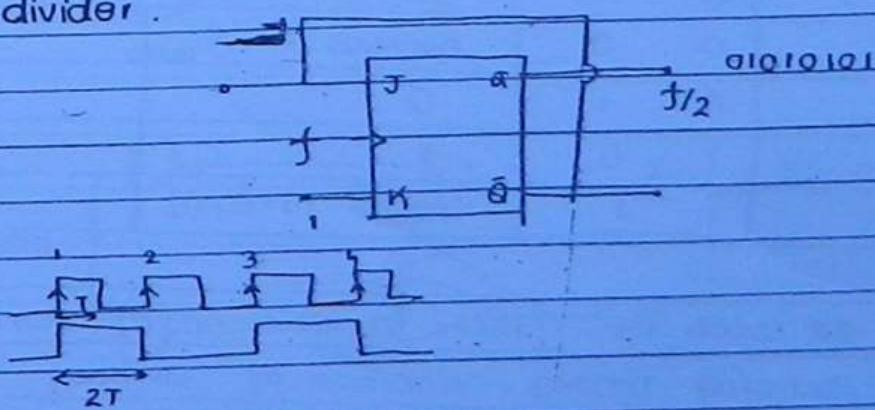
CONTENT :-

0. SR latch 

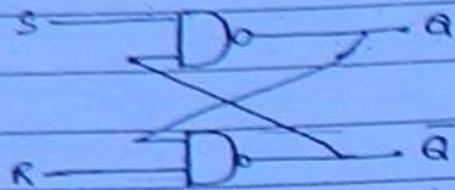
- | | | |
|----------|----------|--|
| 1. SR FF | ⇒ CKts | |
| | 2. JK FF | ⇒ Truth table |
| | 3. D FF | ⇒ characteristic table |
| | 4. T FF | ⇒ characteristic equation
⇒ Excitation table
⇒ conversion from one to another
⇒ simple CKt. |



- ⇒ using Not gate the problem is, it have only one I/P then we use NAND or NOR gate instead of NOT gate.
- ⇒ FF is not only used for storing 1bit but it also used for frequency divider.



SR latch using NAND :-



NAND:-

enable - L

disable - O

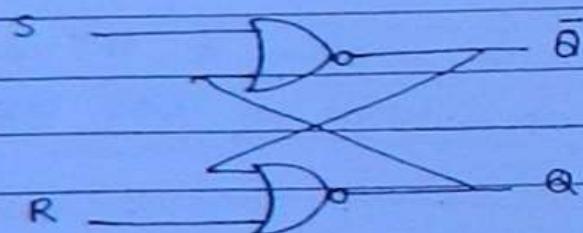
A	B	Y
00	1	
01	1	
10	1	
11	0	

Truth table:-

S	R	Q
0	0	Invalid ($Q = \bar{Q} = 1$)
0	1	1
1	0	0
1	1	Previous state (no change)

⇒ In SR latch if both gates are enabled O/P remains same previous state and both are disable then O/P remains same invalid state.

SR latch using NOR gate:-



∴ NAND enable is 1 and disable is 0

and, in NOR - E = 0

D = 1

⇒ then we change a and \bar{Q} position

Truth table:-

S	R	Q
0	0	Previous state.
0	1	0
1	0	1
1	1	invalid ($Q = \bar{Q} = 0$)

A	B	Y
00	1	
01	0	
10	0	
11	0	

⇒ SR latch is used to eliminate switch bouncing.

⇒ Bouncing means vibration of switches when ON or OFF

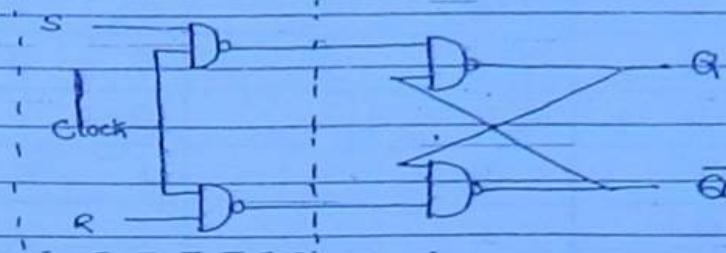
SR Flip Flop :-

89.

S = Set

R = Reset

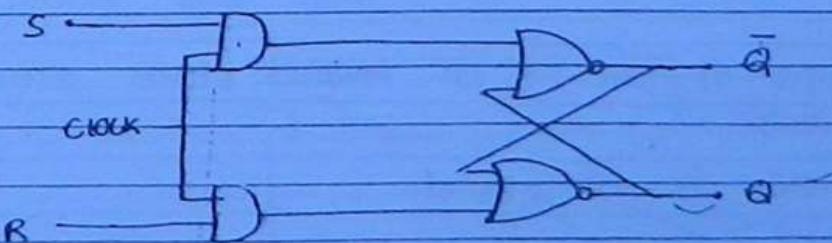
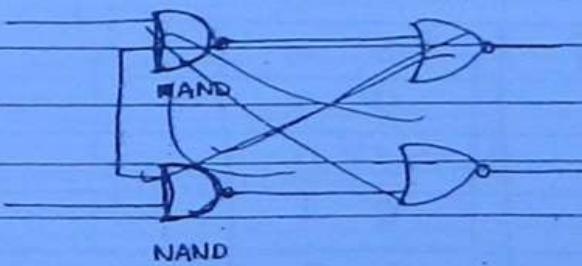
This term is used only as a inverter



Truth table :-

CLOCK	S	R	Q_{n+1}	
0	*	*	Previous state. (Q_n)	
1	0	0	Q_n	Hold state
1	0	1	0	→ Reset
1	1	0	1	→ Set
1	1	1	invalid.	→ unused

S	R	Q_{n+1}	
0	0	Q_n	
0	1	0	< very imp_
1	0	1	
1	1	invalid.	



⇒ Truth table is same as for NAND gate SR FF.

Characteristic table :-

(Q0)

S	R	Q_n	Q_{n+1}	S	R	Q_{n+1}
0	0	0	0	0	0	Q_n
0	0	1	1	0	1	0
0	1	0	0	1	0	1
0	1	1	0	1	1	invalid
1	0	0	1			
1	0	1	1			
1	1	0	X			
1	1	1	X			

S	$R\bar{Q}_n$	$\bar{R}Q_n$	RQ_n	$R\bar{Q}_n$	$\bar{R}\bar{Q}_n$
\bar{S}		[1]			
S	X	X	X	X	X

$$Q_{n+1} = S + \bar{R}Q_n$$

$$Q_{n+1} = S + \bar{R}Q_n \quad \text{and} \quad SR = 0 \quad \text{--- (i)}$$

\Rightarrow since $S=1, R=1$, the o/p is invalid because $S \cdot R = 1$ not satisfy the above condition.

Excitation table :-

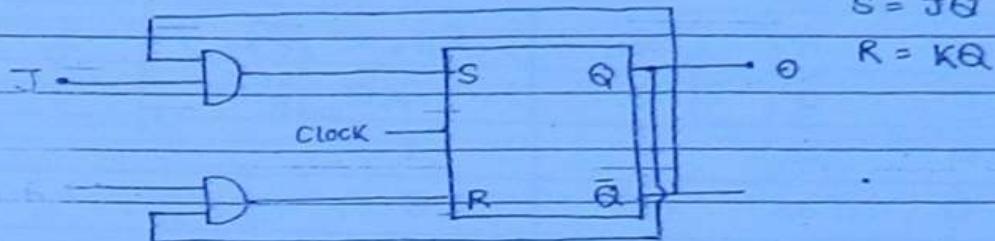
Q_n	Q_{n+1}	S	R
0	0	0	*
0	1	1	0
1	0	0	1
1	1	X	0

\Rightarrow Disadvantage of SR FF is invalid state present when $S=1$ and $R=1$.

\Rightarrow To avoid this JK FF is used.

JK Flip Flop :-

(91)



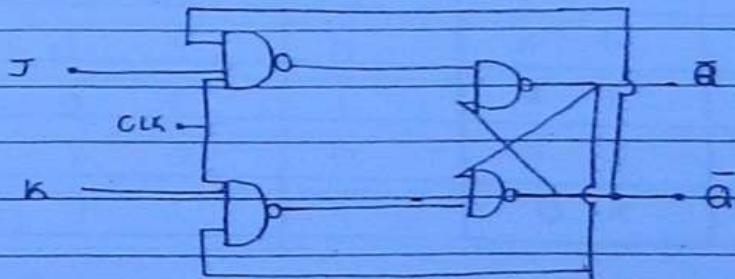
$$S = J\bar{K}$$

$$R = K\bar{Q}$$

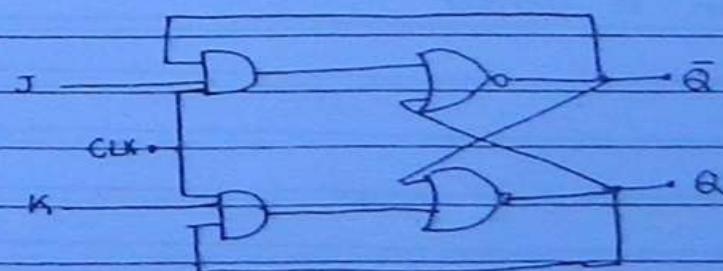
Clock	J	K	Q_{n+1}
0	x	x	Q_n
1	0	0	Q_n
1	0	1	0
1	1	0	1
1	1	1	\bar{Q}_n

J	K	Q_{n+1}	
0	0	Q_n	Hold
0	1	0	Reset
1	0	1	Set
1	1	\bar{Q}_n	Toggle.

J-K FF using NAND gate :-



J-K FF using NOR gate :-



JK FF characteristic table :-

J	K	Q_n	Q_{n+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Logical Expression :-

minimization :-

$\bar{J} \bar{K} \bar{Q}_n$	$\bar{K} \bar{Q}_n$	$\bar{K} Q_n$	$K \bar{Q}_n$	$K Q_n$
0	0	1	1	0
1	0	1	0	1

$$Q_{n+1} = \bar{J} \bar{Q}_n + \bar{K} Q_n$$

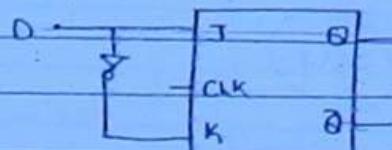
$$Q_{n+1} = J Q_n + K \bar{Q}_n$$

Exitation table :-

Q_n	Q_{n+1}	J	K
0	0	0	x
0	1	1	x
1	0	x	1
1	1	x	0

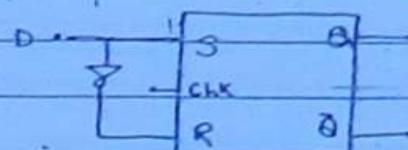
- Drawback in JK ff is Race arround condition which is eliminated in D flipflop.

D-Flip Flop :-



$$J = D$$

$$JK = \bar{D}$$



$$S = D$$

$$R = \bar{D}$$

(Q2)

Truth table :-

CLK	D	Q_{n+1}
0	*	Q_n
1	0	0
1	1	1

D	Q_{n+1}
0	0
1	1

Characteristic table :-

D	Q_n	Q_{n+1}
0	0	0
0	1	0
1	0	1
1	1	1

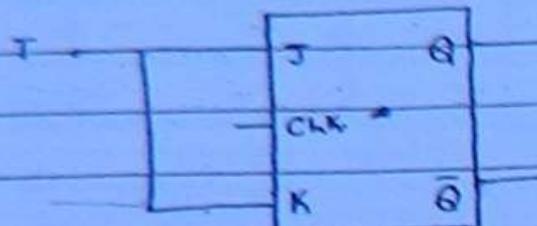
$$Q_{n+1} = D$$

Therefore it is also called transparent latch

Excitation table :-

Q_n	Q_{n+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

T Flip-Flop (Toggle) :-



$$J = K = T$$

Truth table :-

CLK	J	K	T	Q_{n+1}
0	*	x	x	Q_n
1	b	x	0	Q_n
1	x	x	1	\bar{Q}_n

T	Q_{n+1}
0	Q_n
1	\bar{Q}_n

Characteristic table :-

T	Q_n	Q_{n+1}
0	0	0
0	1	1
1	0	1
1	1	0

$$Q_{n+1} = T Q_n + \bar{T} Q_n = T \oplus Q_n$$

Excitation table :-

Q_n	Q_{n+1}	T
0	0	0
0	1	1
1	0	1
1	1	0

Important :-

	J	K	Q_{n+1}	95
SR-FF	0	0	Q_n	
	0	1	0	D-FF
	1	0	1	T-FF
	1	1	Q_n	

\Rightarrow All tables are inside JK FF therefore it is also called as JK FF universal flip flop.

Excitation table :-

Q_n	Q_{n+1}	S	R	J	K	D	T
0	0	0	x	0	x	0	0
0	1	1	0	1	x	1	1
1	0	0	1	x	1	0	1
1	1	x	0	x	0	1	0

FF \rightarrow Flip Flop - one bit storing element

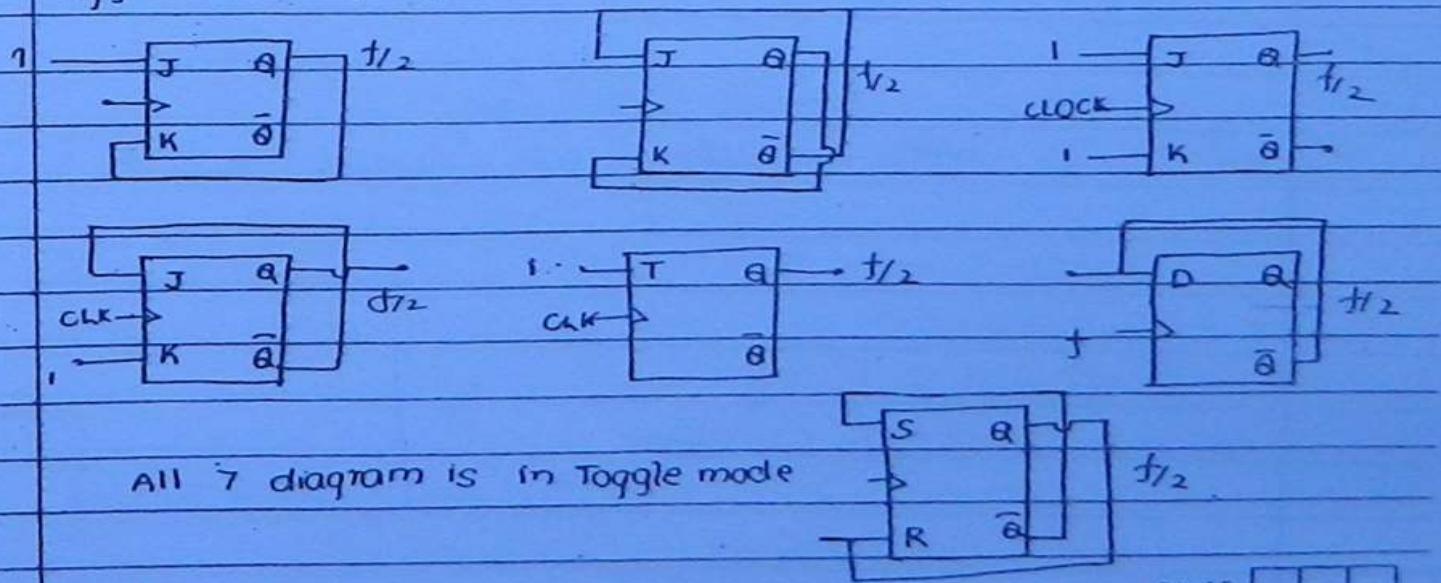
$$Q_{n+1} = S + \bar{R} Q_n \Rightarrow SR = \text{Set Reset}$$

$$Q_{n+1} = J\bar{Q}_n + \bar{K}Q_n \Rightarrow JK = \text{name of person who give the IC}$$

$$Q_{n+1} = D \Rightarrow D = \text{Delay element}$$

$$Q_{n+1} = T \oplus Q_n \Rightarrow T = \text{Toggle}$$

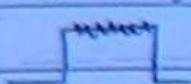
Toggle mode of JK :-



Trigger

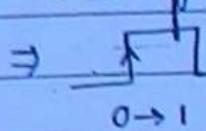
96

Level trigger

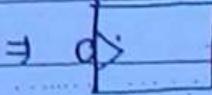
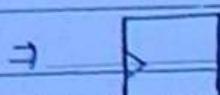
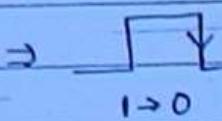


Edge Trigger.

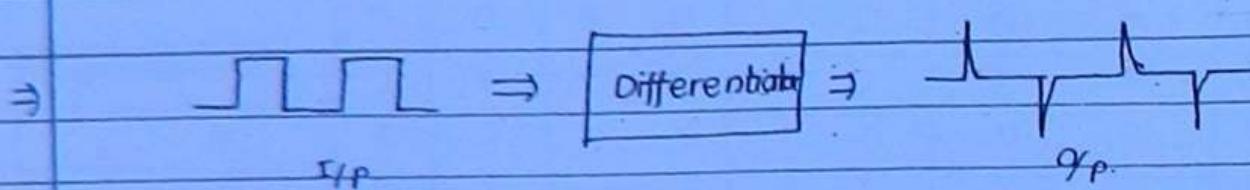
+ive edge trigger



-ive edge trigger



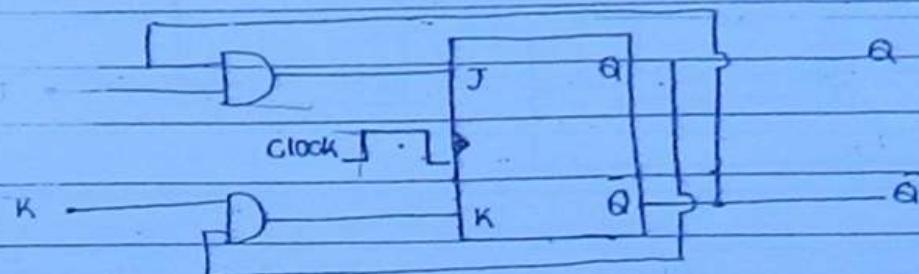
- ⇒ In level trigger ckt, o/p may changes many time in single clock
- ⇒ In edge trigger, o/p may change only ones in single pulse.



Race Around Condition :-

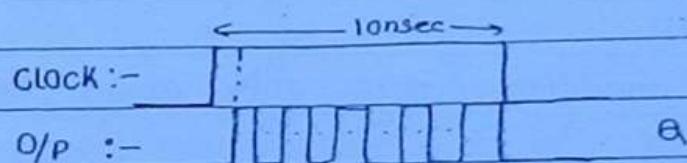
⇒ occurs in JK flip-flop (Draw back)

(91)



if $t_{pw} = 1 \text{ nsec}$

$t_{PFF} = 1 \text{ nsec}$. Then:-



⇒ Then, To remove race around condition:-

$$t_{PFF} < t_{pdclock} \quad t_{pdclock} < < t_{PDEFF}$$

⇒ In JK FF. RAC occurs when $J = K = L$. then t_{pdff} is less than that of $t_{pdclock}$. and therefore the O/P is changes several time in single clock pulse.

Condition to remove Race around condition:-

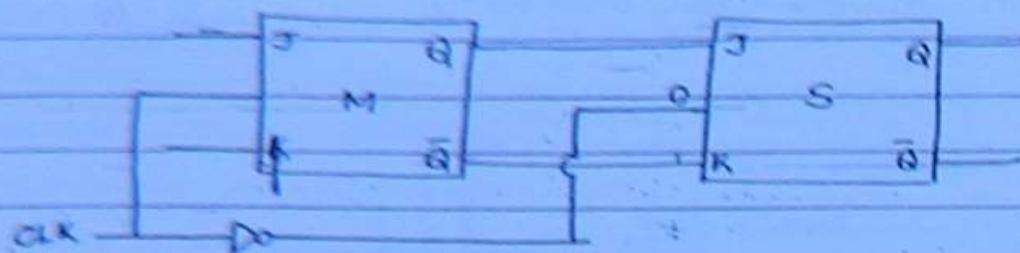
(i) $t_{pdclock} < < t_{PDEFF}$

(ii) Use of Master slave flip flop.

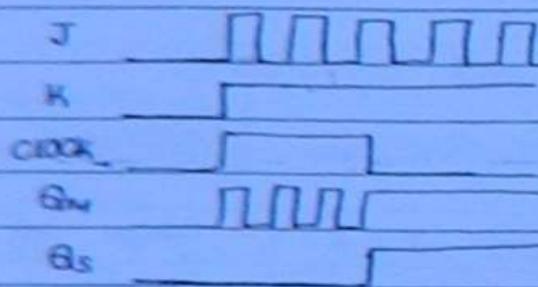
(iii) To increase the propagation delay of JK flip flop.

Master slave Flip Flop :-

(98)



- ⇒ Since the T/P of slave never go to (1,1) therefore in master-slave the race around condition is removed.
- ⇒ Since T/P of slave is $J=Q$ and $K=\bar{Q}$ therefore it is always (1,0) or (0,1).
- ⇒ since race around condition occurs only when the T/P is (1,1).



- ⇒ In M-S FF , o/p is change only when slave o/p is changing
- ⇒ In M-SFF , Master is level triggered and edge is slave is edge triggered .

Conversion of one FF to other FF :-

Procedure :-

⇒ Required FF characteristic table.

⇒ Available FF excitation table.

⇒ Write logical expression for excitation.

(99)

i) JK-Flip Flop to D-Flip-Flop :-

D	Q_n	Q_{n+1}	J	K
0	0	0	0	x
0	1	0	x	1
1	0	1	1	x
1	1	1	x	0

⇒ Write the logical expression for J and K :-

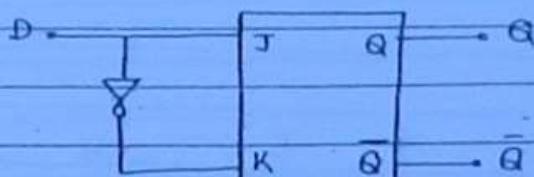
$$J = \begin{cases} \bar{D} & Q_n \\ D & \bar{Q}_n \end{cases}$$

$$K = \begin{cases} \bar{D} & Q_n \\ D & \bar{Q}_n \end{cases}$$

$$J = D$$

$$K = \bar{D}$$

⇒ Implementation :-



Important :-

(A) SR to :-

(10)

i) JK :- $S = J\bar{Q}$
 $R = KQ$

A

ii) D :- $S = D$
 $R = \bar{D}$

iii) T :- $S = T\bar{Q}$
 $R = TQ$

(B) JK to :-

i) SR :- $J = S$
 $K = R$

ii) D :- $J = D$
 $K = \bar{D}$

iii) T :- $J = T$
 $K = \bar{T}$

(C) D to :-

i) SR : $D = S + \bar{R}Q$

ii) JK : $D = J\bar{Q} + \underline{KQ}$

iii) T : $D = T \oplus Q$

(D) T to :-

i) SR : $T = S\bar{Q} + RQ$

ii) JK : $T = J\bar{Q} + KQ$

iii) D : $T = D \oplus Q$

(b) JK FF to SR FF :-

S	R	Q_n	\bar{Q}_{n+1}	J	K
0	0	0	0	0	X
0	0	1	1	X	0
0	1	0	0	0	X
0	1	1	0	X	1
1	0	0	1	1	X
1	0	1	1	X	0
1	1	0	X	X	X
1	1	1	X	X	X

(10)

0	0	0	X
0	1	1	0
1	0	X	1
1	1	X	0

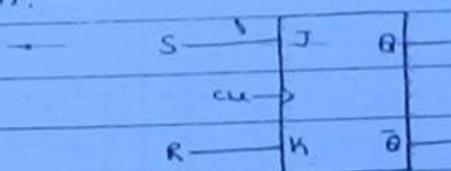
S	$\bar{R}Q_n$				
J \Rightarrow	\bar{S}	X	X		
S	1	X	X	X	

S	$\bar{R}Q_n$	$\bar{R}\bar{Q}_n$	$R\bar{Q}_n$	$R\bar{Q}_n$
K \Rightarrow	X		1	X
S	X		X	X

$$J = S$$

$$K = R$$

implementation:-



(c) JK FF to T FF :-

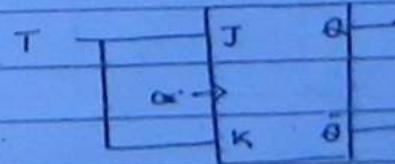
T	Q_n	\bar{Q}_{n+1}	J	K
0	0	0	0	X
0	1	1	X	0
1	0	1	1	X
1	1	0	X	1

T	\bar{Q}_n	\bar{Q}_n	Q_n
J :-	\bar{T}	X	
T	1	X	

T	\bar{Q}_n	Q_n
K :-	X	
T	X	

$$J = T, \quad K = T$$

implementation:-



(IV) SR to JK FF :-

Q_n	Q_{n+1}	S	R
0	0	0	X
0	0	1	0
0	1	0	X
0	1	1	0
1	0	0	1
1	0	1	X
1	1	0	1
1	1	1	0

(102)

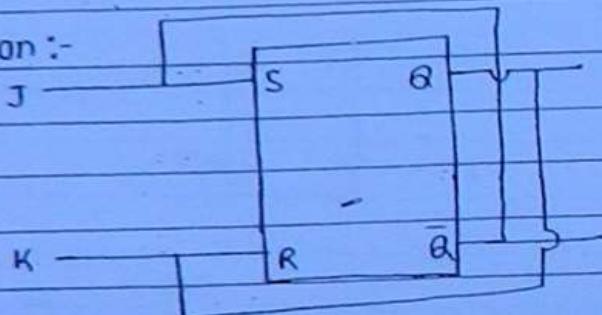
J	S	R	Q_n	Q_{n+1}	S	R
0	0	0	0	0	X	
0	0	1	1		X	0
0	1	0	0		0	X
0	1	1	0		0	1
1	0	0	1		1	0
1	0	1	1		X	0
1	1	0	1		1	0
1	1	1	0		0	1

J	$\bar{R}Q_n$	$R\bar{Q}_n$	$K\bar{Q}_n$		$J\bar{K}Q_n$		KQ_n
\bar{J}		X			J	X	
S :- J	1	X	1		\bar{J}		11

$$S = J\bar{Q}_n$$

$$R = \bar{J}KQ_n$$

Implementation :-



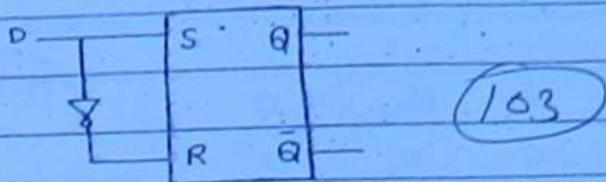
(V) SR FF to D FF :-

D	Q_n	Q_{n+1}	S	R
0	0	0	0	X
0	1	0	0	1
1	0	1	1	0
1	1	1	X	0

D	Q_n	Q_{n+1}	D	
\bar{D}			\bar{D}	
S :- D	0	1	X	
			1	

$$R = \bar{D}$$

Implementation :-



vii) SR to T FF :-

T	Q_n	\bar{Q}_{n+1}	S	R
0	0	0	0	X
0	1	1	X	0
1	0	1	1	0
1	1	0	0	1

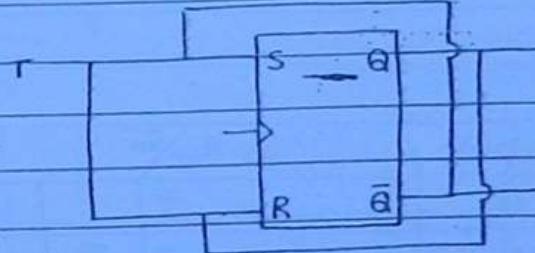
T	Q_n	\bar{Q}_n	\bar{Q}
\bar{T}		X	
T	1		

T	Q_n	\bar{Q}_n	Q_n
\bar{T}	X		
T			1

$$S = T\bar{Q}_n$$

$$R = TQ_n$$

Implementation :-



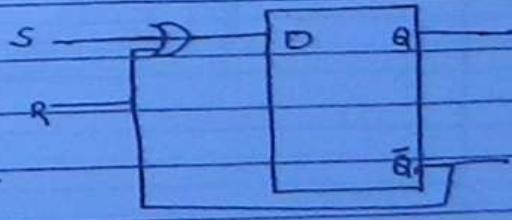
viii) D FF to SR FF :-

S	R	Q_n	\bar{Q}_{n+1}	D
0	0	0	0	0
0	0	1	1	1
0	1	0	0	0
0	1	1	0	0
1	0	0	1	1
1	0	1	1	1
1	1	0	X	X
1	1	1	0	X

Q_n	\bar{Q}_{n+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

S	$\bar{R}Q_n$	D	\bar{Q}_n
\bar{S}		1	
S	1	1	1
1	1	1	X
1	X	X	1

$$D = S + \bar{R}\bar{Q}_n$$



(viii) D FF to JK FF :-

J	K	Q_n	Q_{n+1}	D
0	0	0	0	0

J	K	Q_n	Q_{n+1}	D
0	0	1	1	1

J	K	Q_n	Q_{n+1}	D
0	1	0	0	0

J	K	Q_n	Q_{n+1}	D
0	1	1	0	0

J	K	Q_n	Q_{n+1}	D
1	0	0	1	1

J	K	Q_n	Q_{n+1}	D
1	0	1	1	1

J	K	Q_n	Q_{n+1}	D
1	1	0	1	1

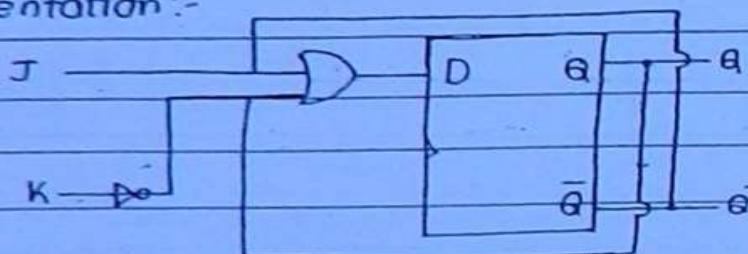
J	K	Q_n	Q_{n+1}	D
1	1	1	0	0

(104)

J	K	Q_n	\bar{Q}_n	$\bar{K}\bar{Q}$	$\bar{K}Q$	$K\bar{Q}$	KQ
0	0	0	1	1	1	1	1
0	0	1	0	1	0	0	1
0	1	0	0	0	1	1	0
0	1	1	0	0	0	1	1

$$D = J\bar{Q} + \bar{K}Q$$

Implementation:-



(ix) D-FF to T FF :-

T	Q_n	Q_{n+1}	D
0	0	0	0

T	Q_n	Q_{n+1}	D
0	1	1	1

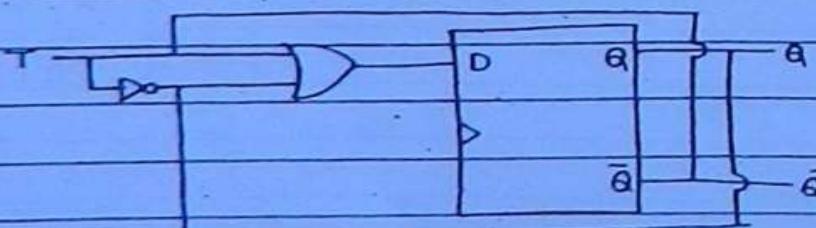
T	Q_n	Q_{n+1}	D
1	0	1	1

T	Q_n	Q_{n+1}	D
1	1	0	0

\bar{Q}_n	\bar{Q}	Q
0	1	1
1	0	0

$$D = \bar{T}Q + T\bar{Q}$$

Implementation:-



(X) T FF to SR FF :-

S	R	Q _n	Q _{n+1}	T
0	0	0	0	0

S	R	Q _n	Q _{n+1}	T
0	0	1	1	0

S	R	Q _n	Q _{n+1}	T
0	1	0	0	0

S	R	Q _n	Q _{n+1}	T
0	1	1	0	1

S	R	Q _n	Q _{n+1}	T
1	0	0	1	1

S	R	Q _n	Q _{n+1}	T
1	0	1	1	0

S	R	Q _n	Q _{n+1}	T
1	1	0	x	x

S	R	Q _n	Q _{n+1}	T
1	1	1	x	x

103

S	R	Q _n	Q _{n+1}	T
0	0	1	1	0

S	R	Q _n	Q _{n+1}	T
0	1	0	0	0

S	R	Q _n	Q _{n+1}	T
0	1	1	0	1

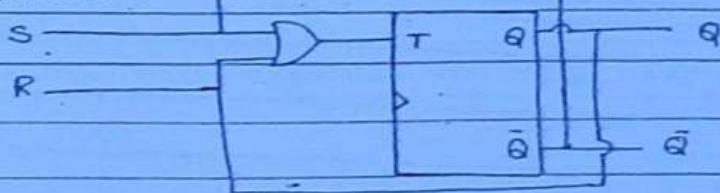
S	R	Q _n	Q _{n+1}	T
1	0	0	1	1

S	R	Q _n	Q _{n+1}	T
1	0	1	1	0

S	R	Q _n	Q _{n+1}	T
1	1	0	x	x

S	R	Q _n	Q _{n+1}	T
1	1	1	x	x

Implementation :-



(X) T FF to JK FF :-

J	K	Q _n	Q _{n+1}	T
0	0	0	0	0

J	K	Q _n	Q _{n+1}	T
0	0	1	1	0

J	K	Q _n	Q _{n+1}	T
0	1	0	0	0

J	K	Q _n	Q _{n+1}	T
0	1	1	0	1

J	K	Q _n	Q _{n+1}	T
1	0	0	1	1

J	K	Q _n	Q _{n+1}	T
1	0	1	0	0

J	K	Q _n	Q _{n+1}	T
1	1	0	1	1

J	K	Q _n	Q _{n+1}	T
1	1	1	0	1

J	K	Q _n	Q _{n+1}	T
0	0	1	1	0

J	K	Q _n	Q _{n+1}	T
0	1	0	0	0

J	K	Q _n	Q _{n+1}	T
0	1	1	0	1

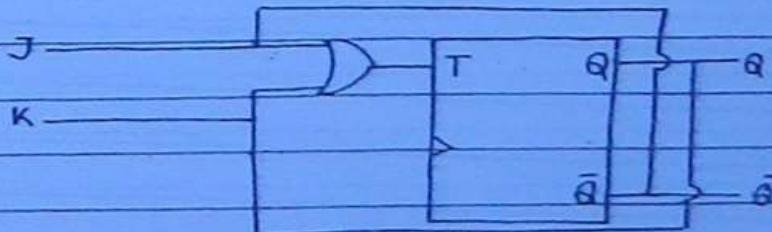
J	K	Q _n	Q _{n+1}	T
1	0	0	1	1

J	K	Q _n	Q _{n+1}	T
1	0	1	0	0

J	K	Q _n	Q _{n+1}	T
1	1	0	1	1

J	K	Q _n	Q _{n+1}	T
1	1	1	0	1

Implementation :-

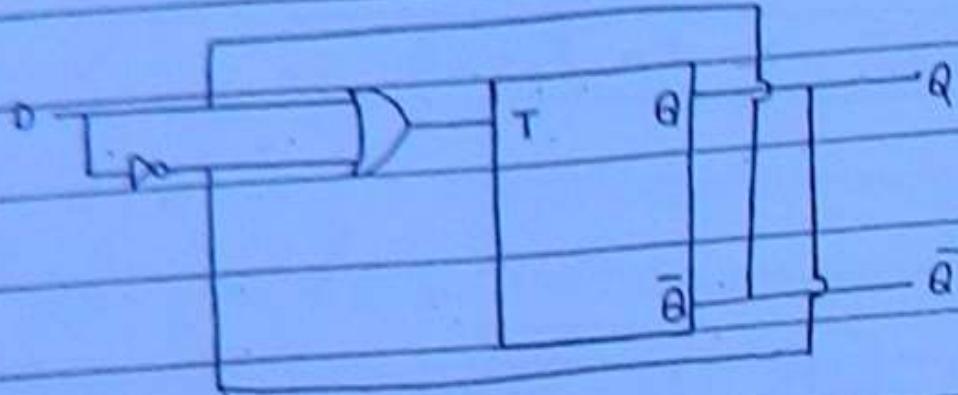


(xi) T-FF to D-FF :-

106

D	Q_n	Q_{n+1}	T	$D\bar{Q}$	$\bar{D}Q$	Q
0	0	0	0	0	0	0
0	1	0	1	0	0	1
1	0	1	1	1	0	1
1	1	1	0	1	1	0

$T = DQ + \bar{D}\bar{Q}$.



* latch.

⇒ level triggered

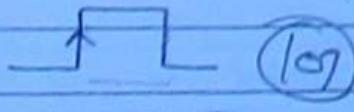
⇒ Asynchronous CKT



Flip Flop.

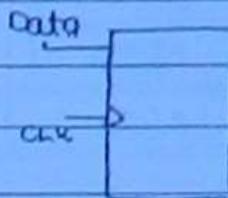
⇒ Edge triggered.

⇒ Synchronous CKT.



⇒ Setup time :-

The min. time required to keep I/P at proper level before applying clock.



Note: - Any if we give Data first then we apply CLK.

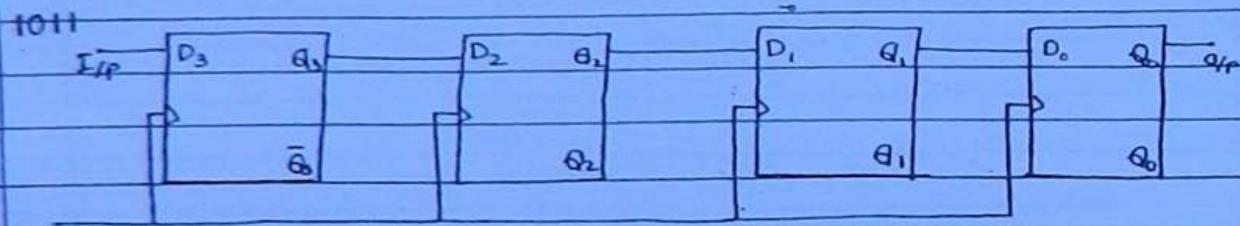
⇒ Hold time:-

The min. time required to keep I/P is same level after applying clock.

★ REGISTER :-

- ⇒ Register are used to store group of bits.
- ⇒ To store n bit n FF are cascaded in register.
- ⇒ Register are four type (Depending on I/P and O/P) :-
 - (i) SISO (serial in serial out).
 - (ii) SIPO (most imp).
 - (iii) PISO
 - (iv) PIPO
- ⇒ Depending on application the register are two type:-
 - (i) Shift register
 - (ii) storage register.

(A) SISO (Serial in Serial out) :-



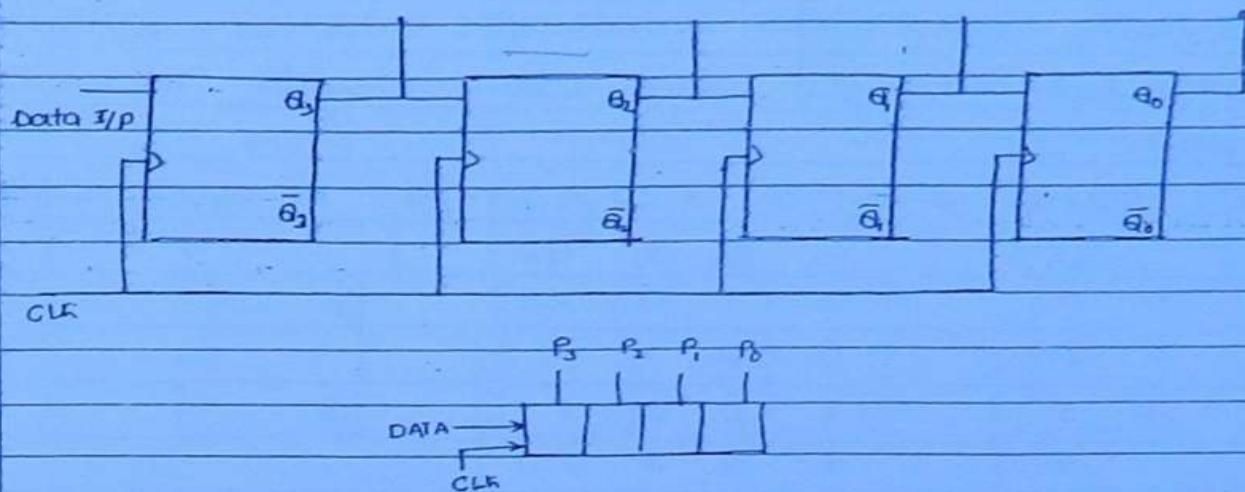
Data	Q ₃ Q ₂ Q ₁ Q ₀	CLK
10101	0 0 0 0	0
	1 0 0 0	1
	1 1 0 0	2
	0 1 1 0	3
	1 0 1 1	4

- ⇒ For serial in register the n bit data storage requires n clock pulse.
- ⇒ In SISO register to store n bit data is require n clock pulse.
- ⇒ SISO register used to provide n clock pulse delay to I/P data.

⇒ To provide n bit data serially out it requires $(n-1)$ clock pulse.

(109)

(A) SIPO (serial in parallel out) :-



⇒ In SIPO register to provide n bit data serially in it requires n clock pulse and provide parallel o/p it requires 0 CLK pulse required.

⇒ It is used to serial to parallel converter.

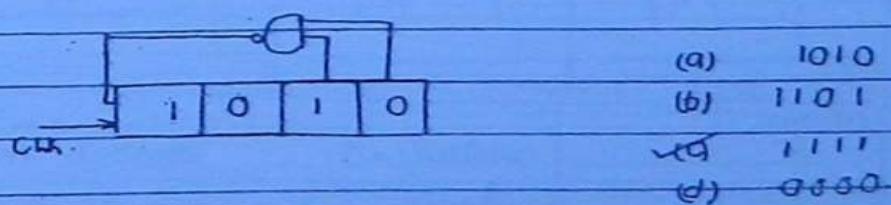
⇒ SIPO is used to convert Temporal code to spacial code.

⇒ Slow to fast converter.

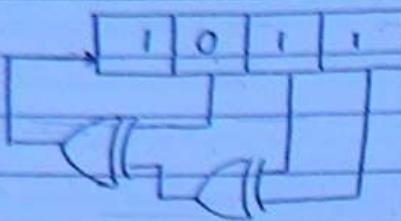
serial. t =

Parallel = Spacial code.

Q:- The ckt shown in the fig. if 4 bit SIPO register which is initially loaded with 1010. If store three clk pulses applied then the data in the system is.



Q:



(a) 4

(b) 7

(c) 11

(d) 15

110

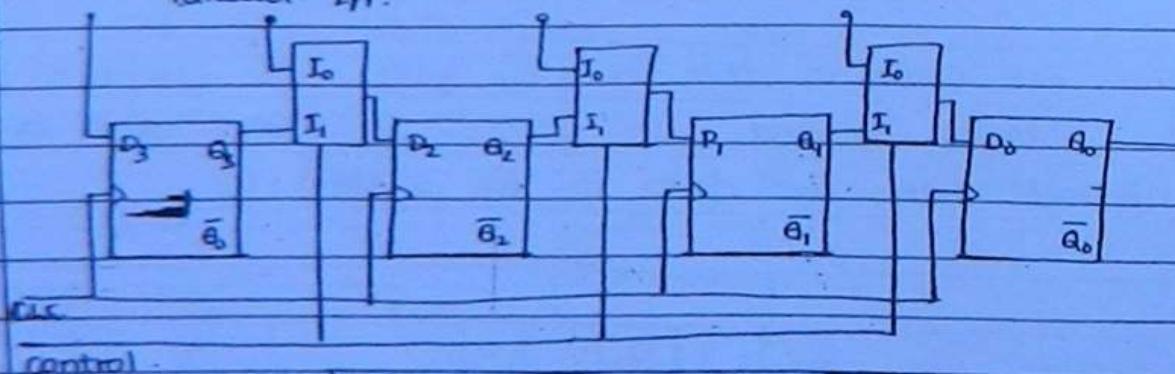
Initially loaded 1011. If CLK pulse applied continuously after how many CLK pulse again the data become 1010.

Sol: o/p of 3 variable XOR is 1 if no. of 1's is odd.

CLK	Q_3	Q_2	Q_1	Q_0
1	0	1	0	1
2	0	0	1	0
3	1	0	0	1
4	1	1	0	0
5	1	1	1	0
6	0	1	1	1
7	0	0	1	1
8	0	0	0	1
9	1	0	0	0
10	0	1	0	0
11	1	0	1	0

(c) PISO (Parallel in serial out) :-

Parallel In.



control = 0 = Parallel In

control = 1 = Serial out

⇒ In PISO register to provide parallel in it require 1 clock pulse and to provide serial out $(n-1)$ clock pulse.

⇒ PISO is also used to convert spacial code to temporal code.

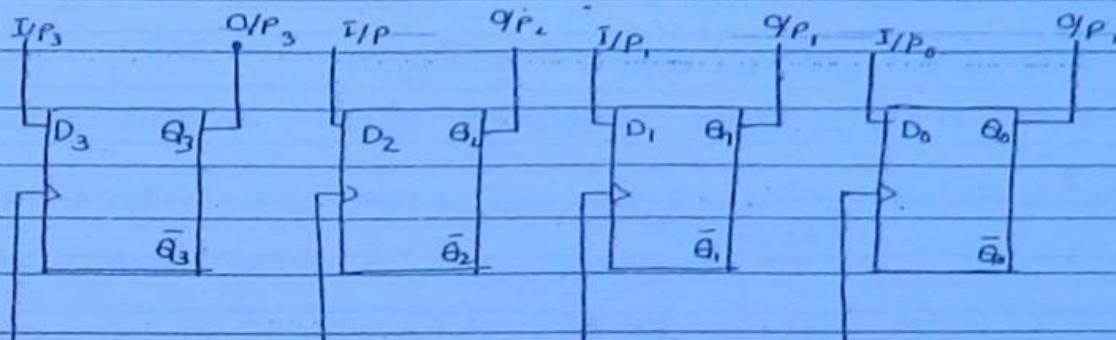
(III)

(D) PIPO (Parallel in parallel out) :-

⇒ PIPO is used as storage register.

⇒ for Parallel in it requires 1 clock pulse.

⇒ for parallel out it requires 0 clock pulse.



Important :-

	I/P	O/P
SISO	n	$n-1$
SIPO	n	0
PISO	1	$n-1$
PIPO	1	0

- ⇒ Each shift left register operation provide multiplication by 2.
- ⇒ If n shift left operation performed then data is multiplied by 2^n .
- ⇒ Each shift right operation performed then data is divide by 2.
- ⇒ If n shift right operation performed then data is divided by 2^n .

★ COUNTERS:-

(112)

- ⇒ Counters are basically used to count no. of clock pulse applied. It can also be used for frequency divider, & time measurement, frequency measurement, range measurement, pulse width.

Pulse.

counter

$$16 \times \text{Pulse width} = \text{Total width.}$$

- ⇒ Also used for waveform generator.

- ⇒ With n -FF, max. possible stage in the counter is 2^n .

$$N \leq 2^n$$

$$\text{or, } n \geq \log_2 N$$

where N = no. of stages.

n = no. of FF.

Depending on clock pulse applied counters of two types:-

(i) Asynchronous

(ii) Synchronous.

Asynchronous

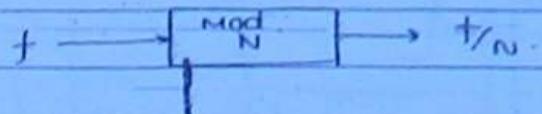
1. Different FF are applied with different clock.
2. It is slower.
3. Fixed count sequence i.e. up or down.
4. Decoding errors will present.
5. Ripple counter

Synchronous

1. All FF are applied same clock.
2. It is faster.
3. Any count sequence is possible.
4. No decoding error will present.
5. Ring counter

⇒ No. of stage use in counter mean modulus of counter
i.e. if MOD 5 counter = 5 stage.
MOD n counter = n stage

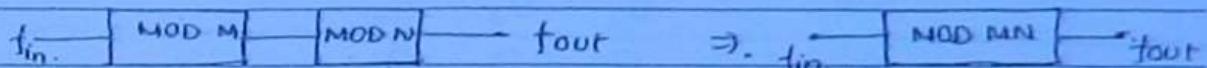
(113)



Q:- A decade counter is applied with frequency of 10MHz then O/P frequency is...

Sol:-
$$\frac{f_{out}}{f_{in}} = \frac{t_{in}}{10} = \frac{10\text{MHz}}{10} = 1\text{MHz}$$

⇒ let MOD M and MOD N are cascaded then it will act as MOD MN counter.



Content :-

Basic

Ripple counter

Non binary ripple counter

Ring counter

Johnson counter

Synchronous series carry

Synchronous parallel carry

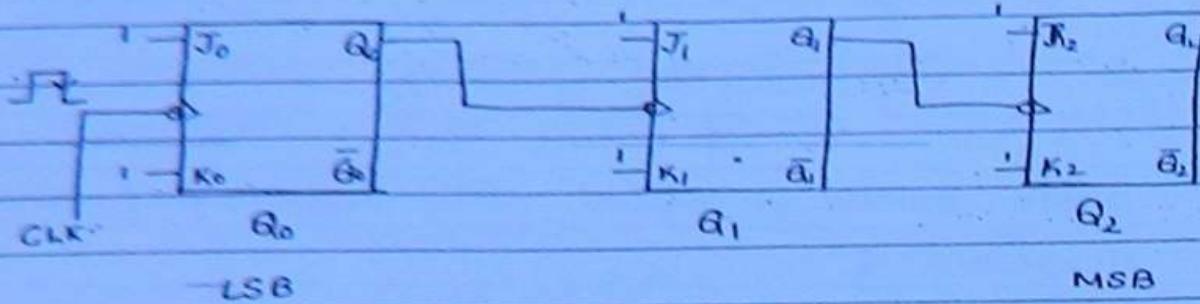
Synchronous counter design and Analysis.

(A) Ripple counter :-

- ⇒ It is a Asynchronous counter.
- ⇒ Different FF used different clock pulse.
- ⇒ Toggle mode.
- ⇒ Only one FF is applied with external CLK and other FF's are CLK is from previous FF o/p. (whether Q or \bar{Q}).
- ⇒ The FF applied with external CLK will acts as LSB bit.

3 bit ripple counter :- (up counter)

(114)



i) Explanation :-

- ⇒ The ckt shown in fig. Q_0 toggle for every CLK pulse
- ⇒ An change when Q_{n-1} change from 1-0. i.e. Q_1 changes when Q_0 changes from 1-0.

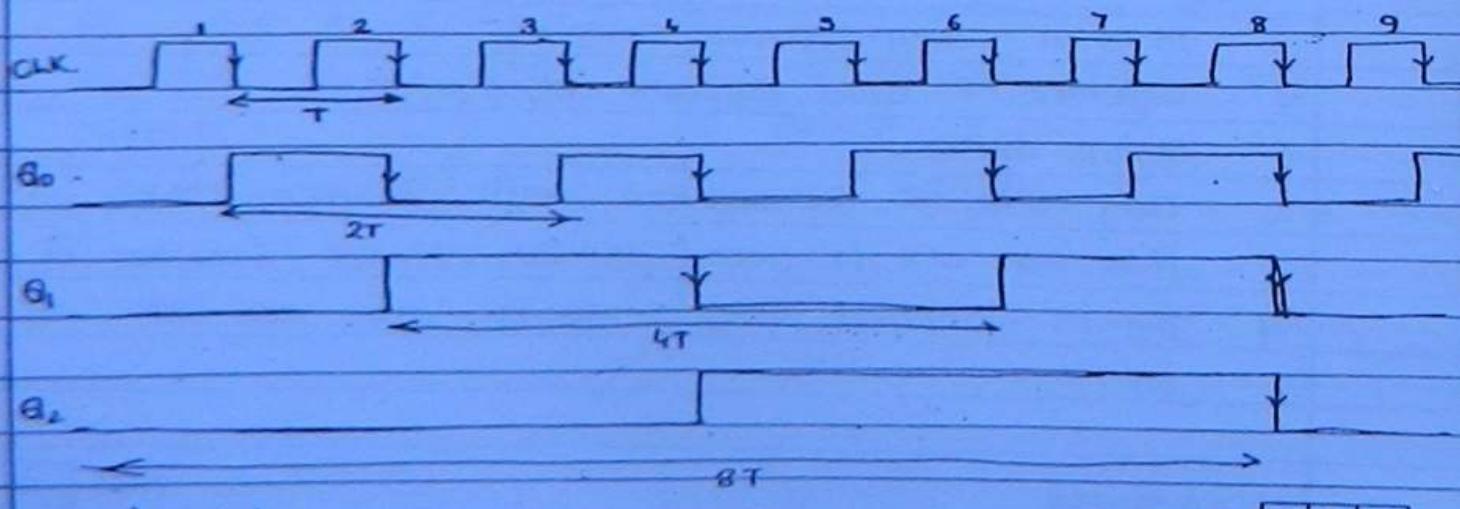
ii) Truth table :-

CLK	Q_2	Q_1	Q_0
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1
8	0	0	0

⇒ This is up counter.

⇒ It is also called MOD 8 ripple counter.

iii) Timing Diagram :-



⇒ In n bit ripple counter propagation delay of each ff is t_{pdff} . then time period of .CLK is,

115

$$T_{CLK} = n t_{pdff}$$

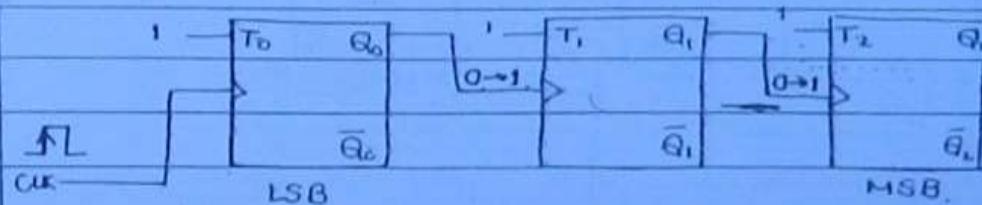
$$f_{CLK} \leq \frac{1}{n t_{pdff}}$$

$$f_{max} = \frac{1}{n t_{pdff}}$$

Note:-

- (i) -ive edge trigger $\rightarrow Q$ as clock \rightarrow up counter
- (ii) +ive $\rightarrow \bar{Q}$ as clock \rightarrow up counter.
- (iii) -ive $\rightarrow \bar{Q}$ as clock \rightarrow down counter.
- (iv) +ive $\rightarrow Q$ as clock \rightarrow down counter.

3-Bit Ripple counter (Down counter):-



(i) Explanation :-

- ⇒ The ckt shown in fig. Q_0 toggles for every clock pulse.
- ⇒ Q_1 toggles when Q_0 changes from 0 to 1.
- ⇒ Q_2 toggles when Q_1 changes from 0 to 1.

(ii) Truth table:-

Clock	Q_2	Q_1	Q_0
0	0	0	0
1	1	1	1
2.	1	1	0
3.	1	0	1
4.	1	0	0
5.	0	1	1
classmate 7	0	1	0
	0	0	1

⇒ This is called ripple counter because the input clock is the o/p of previous FF output. this is just like ripple. then called Ripple counter.

(116)

⇒ if the o/p is (000) and clock is applied then t_{pdff} is delay

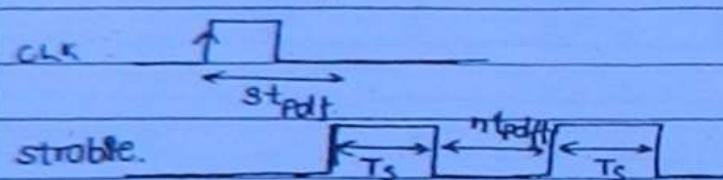
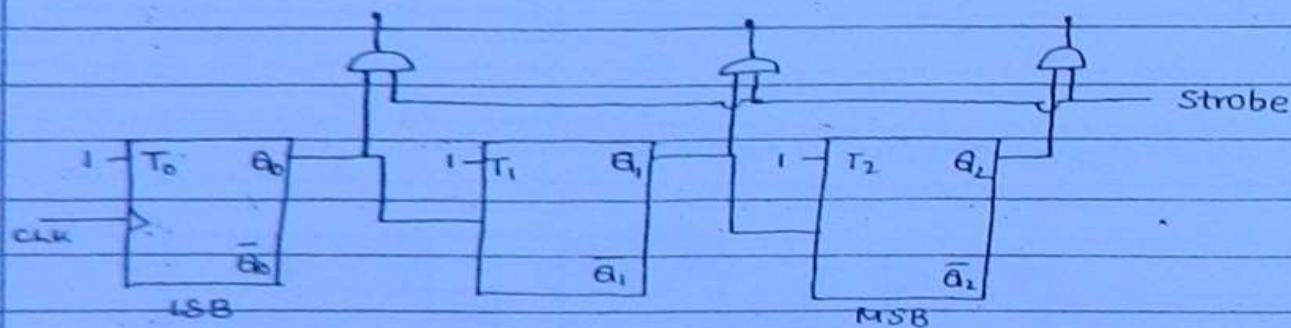
000

$t_{pdff} \rightarrow \{ 001 \}$ unwanted or decoding error.
 $t_{pdff} \rightarrow \{ 011 \}$ also called transient state.

111

⇒ Decoding errors or transient state present in ripple counter due to propagation delay.

⇒ To avoid decoding error strobe signal is used.



⇒ i.e. strobe signal is zero for nt_{pdff} and after that it is one for next clock. then all the o/p is zero for the transient time therefore due to strobe signal we can remove decoding error.

$$T_{CLK} \geq nt_{pdff} + T_s$$

⇒ In ripple counter with n ff. max. possible state is 2^n .

⇒ frequency after n 8FF in the Ripple counter is $f/2^n$. (i.e. for 3-FF o/p is $f/8$)

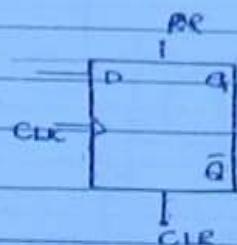
⇒ Clear and preset are known as Asynchronous I/P.

S, R, J, K, D, T, are Synchronous I/P.

(117)

Clear:- clear is use to rest our FF or counter.

Preset:- preset is use to set our FF or counter.



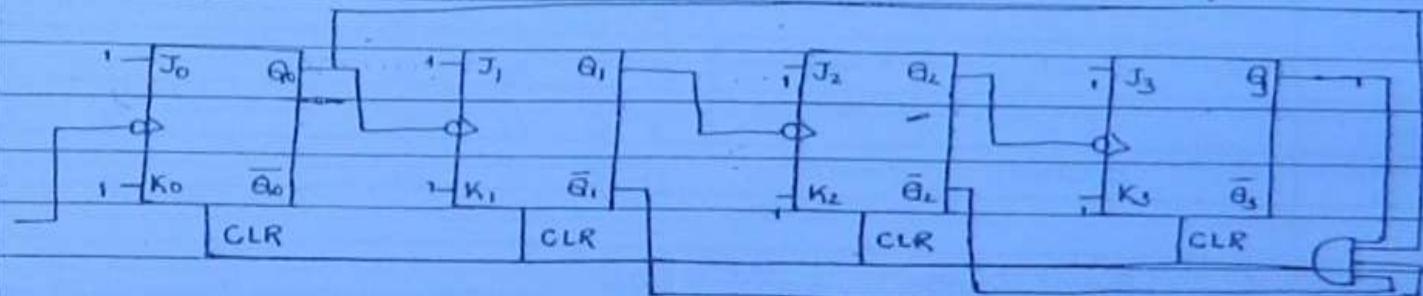
$CLR = 0$, no effect
 $= 1$ → FF is zero

$CLR = 0$ FF is zero
 $= 1$ no effect

(B) Non Binary Ripple counter:-

(B₁) BCD counter :- (Decade counter)

⇒ 4 flip flop used.

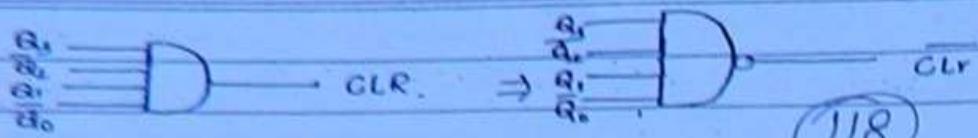


CLK	Q_3	Q_2	Q_1	Q_0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
0	0	0	0	0

classmate

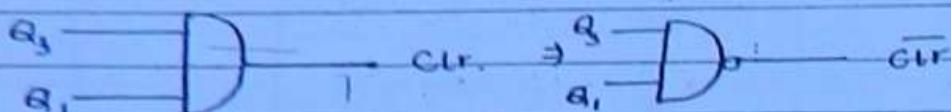
(1010)

କ୍ଷେତ୍ରଫଳ



(118)

if we use only Q_3 and Q_1 , we also use this as clear ckt.



⇒ All BCD counter is decade counter but reverse is not true.

⇒ BCD counter is Asymmetric o/p time Diagram.

⇒ o/p frequency of BCD counter is $f/10$.

⇒ low for 8 clock and high for 2 clock in Q_3 .

⇒ duty cycle is 20%.

⇒ In Asynchronous counter follow steps:-

1. Trigger

2. clock

3. counter

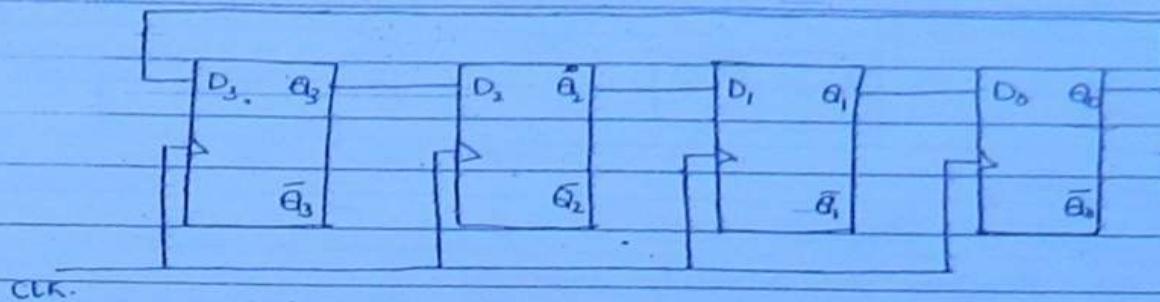
4. Preset /clear

5. Decoding logic (Terminating logic).

(c) Ring Counter :- (Synchronous counter)

119

⇒ The last ff o/p is connected to first ff I/P.



(ii) Explanation:-

⇒ only one FF o/p is high and remaining FF are low.

⇒ In 4 bit ring counter 4 states are there. (i.e. For n FF there is n states).

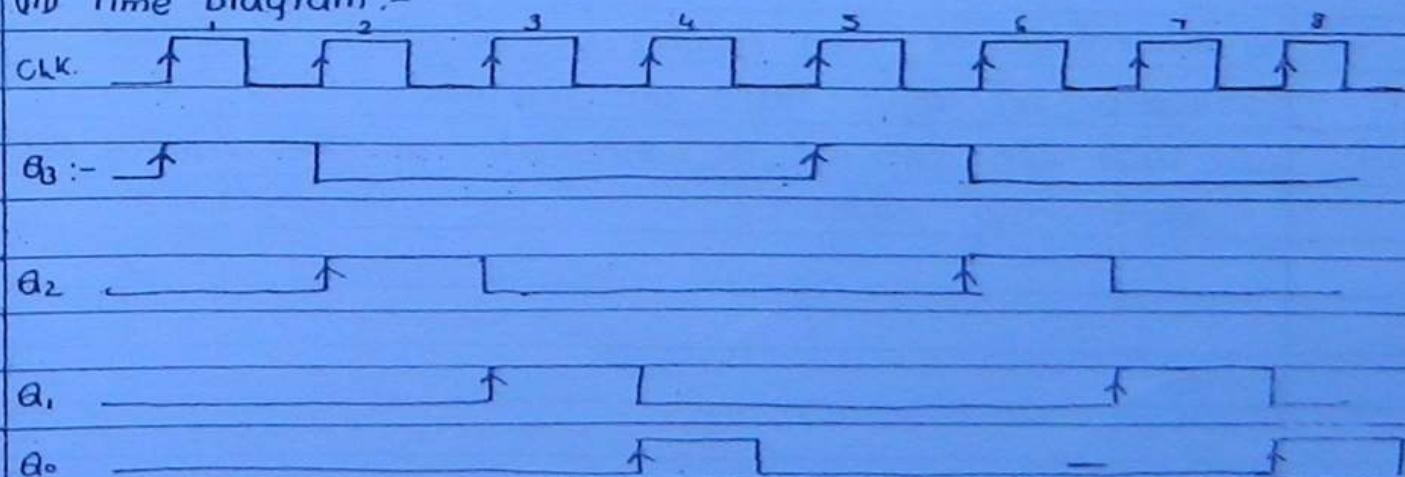
(iii) Truth table :-

CLK	Q ₃	Q ₂	Q ₁	Q ₀
0	0	0	0	0
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1
5	1	0	0	0

↳ 4 states.

→ In synchronous counter the +ive edge or -ive edge, the o/p remains same.

(iv) Time Diagram:-



$$\Rightarrow n \text{ bit} \Rightarrow n \text{ state}$$

$$\Rightarrow t_0 = \frac{t}{n}$$

(T20)

\Rightarrow Phase shift b/w generated waveform is $360/n$.

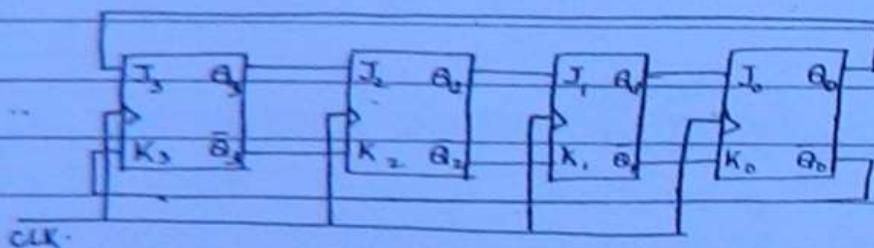
$$\phi = \frac{360}{n}$$

Application :-

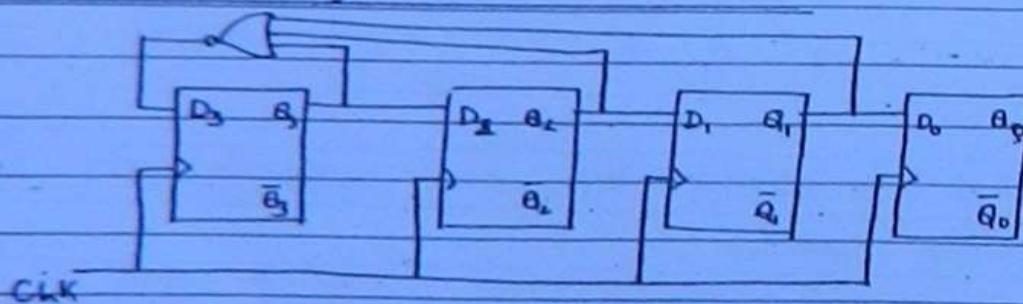
- \Rightarrow used in stepper motor control.
- \Rightarrow in Analog to Digital converter

\Rightarrow No. of unused state in ring counter is $2^n - n$.

ring counter using J-K :-



* self starting Ring counter :-



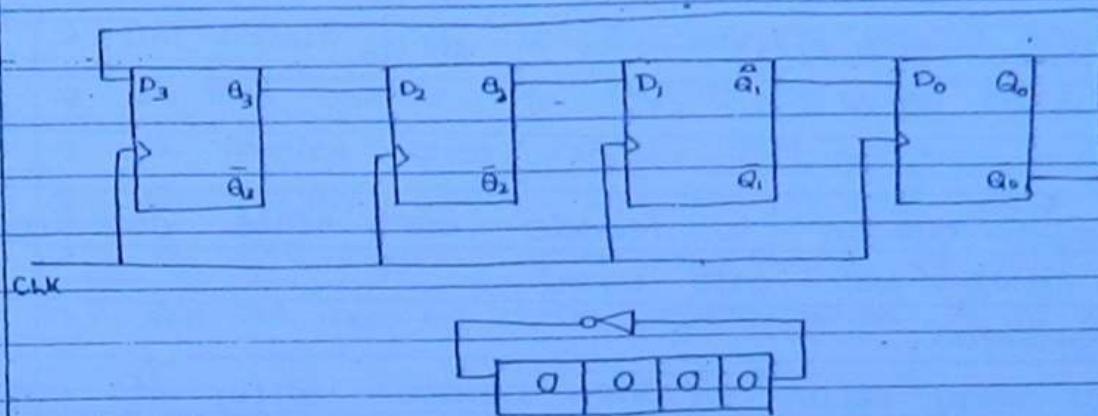
\Rightarrow Advantage of Ring counter is decoding is simple. to decode no logic gate are required.

\Rightarrow last o/p can not be connected in the I/P of self start ring counter.

(D) Johnson Counter :-

(12)

- ⇒ Symmetric o/p waveform.
- ⇒ 8-stages are there for 4 bit counter.
- ⇒ Phase shift = $\frac{360}{4} = 90^\circ$
- ⇒ It is just like SISO register.



(iii) Truth Table :-

CLK	Q_3	Q_2	Q_1	Q_0
0	0	0	0	0
1	1	0	0	0
2	1	1	0	0
3	1	1	01	0
4	1	1	1	1
5	0	1	1	1
6	0	0	1	1
7	0	0	0	1
8	0	0	0	0

8-state

- ⇒ Total no. of used state = 8
- ⇒ Total no. of unused state = $2^n - 8 = 2^4 - 8 = 8$ state
- ⇒ Also called Twisted ring counter, Mobies counter or, creeping counter or, Walking counter or, switch tail counter.

CLOCK	$Q_3\ Q_2\ Q_1\ Q_0$	
0	0 0 0 0	$\rightarrow\ \bar{Q}_3\ \bar{Q}_0$
1	0 0 0 0	$\rightarrow\ Q_3\ \bar{Q}_2$
2	1 0 0 0	$\rightarrow\ Q_2\ \bar{Q}_1$
3	1 1 0 0	$\rightarrow\ Q_1\ \bar{Q}_0$
4	1 1 1 0	$\rightarrow\ \bar{Q}_3\ Q_0$
5	0 1 1 1	$\rightarrow\ \bar{Q}_3\ Q_2$
6	0 0 1 1	$\rightarrow\ \bar{Q}_2\ Q_1$
7	0 0 0 1	$\rightarrow\ \bar{Q}_1\ Q_0$
8	0 0 0 0	

(122)

⇒ In Johnson counter to decode each state one two I/P. AND / NOR gate used.

Disadvantage:

⇒ lock out may occur. (when counter enter into unused state)

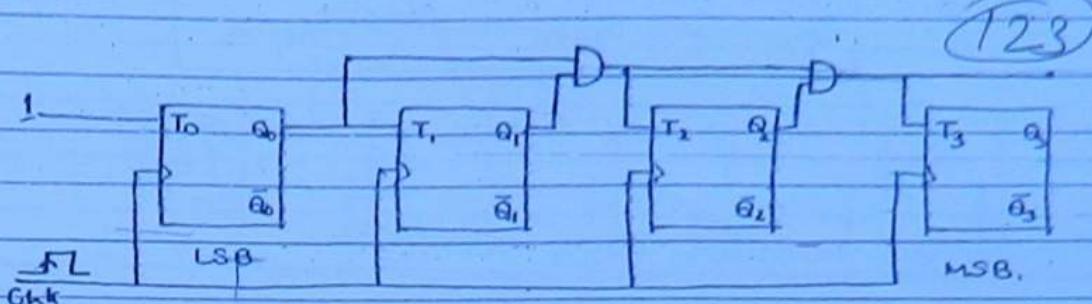
Note:- In synchronous counter propagation delay of each counter is t_{pdff} . then,

$$T_{clk} \geq t_{pdff}$$

$$T_{clk} \leq \frac{1}{t_{pdff}}$$

$$T_{max} = \frac{1}{t_{pdff}}$$

} In synchronous counter.

(A) Synchronous Series carry counter :-

(i) Explanation :-

- ⇒ CKT shown in fig. is Synchronous series carry up counter.
- ⇒ In this counter Q_0 toggles for every clock pulse.
- ⇒ Q_1 toggles when $Q_0 = 1$ and clock is applied.
- ⇒ Q_2 toggles when $Q_1 = Q_0 = 1$ and clock applied.
- ⇒ Q_3 will toggles when $Q_2 = Q_1 = Q_0 = 1$ and clock applied.
- ⇒ This ckt may be down counter when \bar{Q} is connected to T.

(ii) Truth table :-

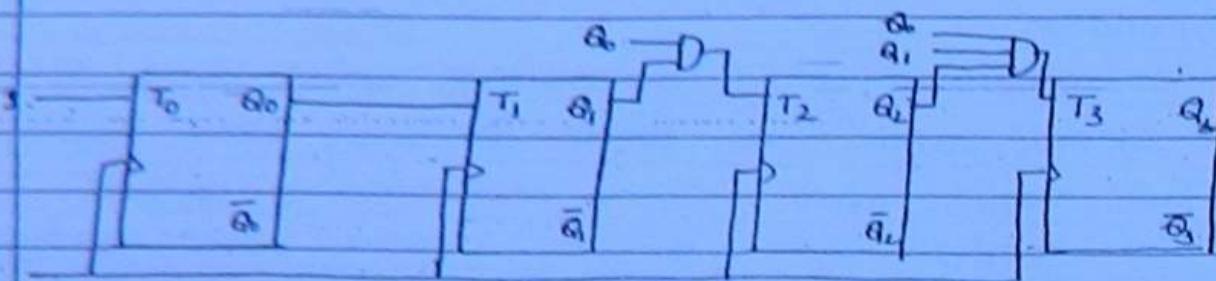
CLOCK	Q_3	Q_2	Q_1	Q_0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

⇒ To provide down counter used \bar{Q} o/p to provide next stage S/I/P.

$$T_{CLK} \geq t_{PDFF} + (n-2) t_{PAND}$$

(124)

Q) Synchronous parallel carry counter :-



⇒ Faster than series carry counter.

⇒ Disadvantage is increased I/P pin of AND Gate.

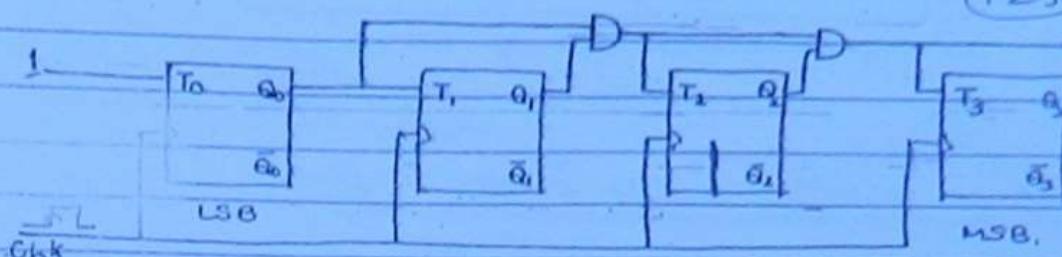
$$T_{CLK} \geq t_{PDFF} + t_{PAND}$$

⇒ Some

⇒ Ripple counter < Synchronous serial carry < Synchronous parallel carry counter for faster logic.

A) Synchronous Series carry counter :-

(125)



ii) Explanation :-

- ⇒ CKT shown in fig. is Synchronous series carry up counter.
- ⇒ In this counter Q_0 toggles for every clock pulse.
- ⇒ Q_1 toggles when $Q_0 = 1$ and clock applied.
- ⇒ Q_2 toggles when $Q_1 = Q_0 = 1$ and clock applied.
- ⇒ Q_3 will toggles when $Q_2 = Q_1 = Q_0 = 1$ and clock applied.
- ⇒ This CKT may be down counter when \bar{Q} is connected to T.

iii) Truth table :-

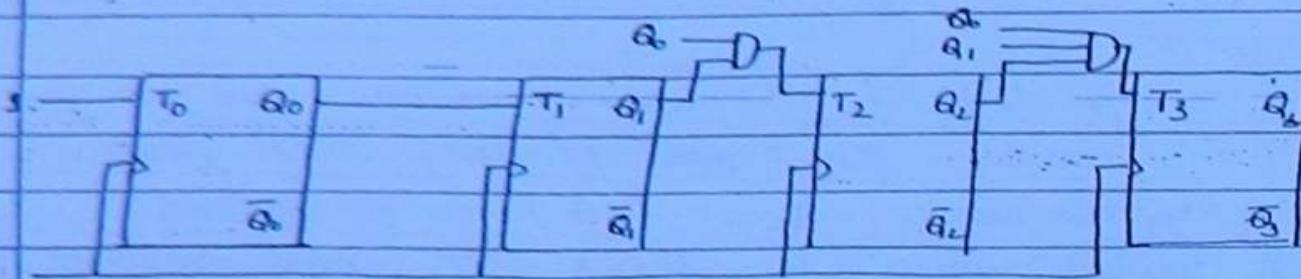
CLOCK	Q_3	Q_2	Q_1	Q_0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

⇒ To provide down counter used \bar{Q} o/p to provide next stage S/P.

$$T_{CLK} \geq t_{PDFF} + (n-2) t_{PAND}$$

126

8) Synchronous parallel carry counter :- .



⇒ Faster than series carry counter.

⇒ Disadvantage is increased I/P pin of AND Gate.

$$T_{CLK} \geq t_{PDFF} + t_{PAND}$$

⇒ Some

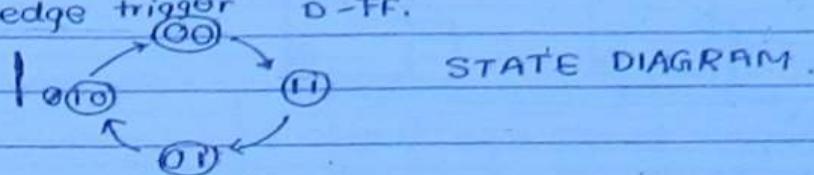
⇒ Ripple counter < Synchronous serial carry < Synchronous parallel carry counter for faster logic.

Synchronous counter design for the given sequence:-

127

Problem: Design a synchronous counter for the count sequence $0 \rightarrow 3 \rightarrow 1 \rightarrow 2 \rightarrow 0$

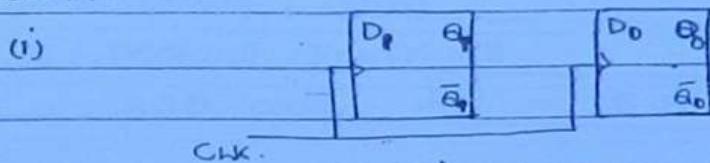
Sol: Using +ive edge trigger D-FF.



Sol:- Procedure :-

- Identify no. of FF and I/P and O/P.
- construct state table.
- logical expression for I/P.
- Minimize.
- Implement the clk.

Now,



(ii) State table :-

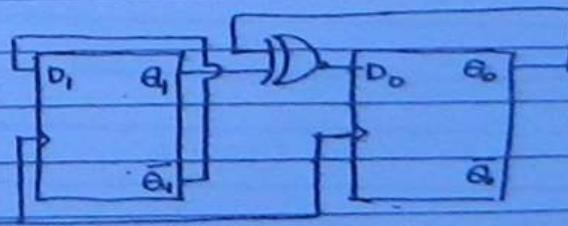
Present state	Next state	D_1	D_0
$Q_1\ Q_0$	$Q_1\ Q_0 +$		
0 0	1 1	1	0 1
1 1	0 1	0	1
0 1	1 0	1	0
1 0	0 0	0	0

(iii) logical expression :-

$$D_1 = \bar{Q}_1 \bar{Q}_0 + \bar{Q}_1 Q_0 = \bar{Q}_1 (Q_0 + \bar{Q}_0) = \bar{Q}_1$$

$$D_0 = \bar{Q}_1 \bar{Q}_0 + Q_1 Q_0 = Q_1 \oplus Q_0 = Q_1 \Theta Q_0$$

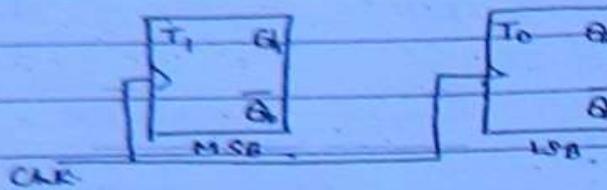
(iv) Implementation :-



Q: Design using T-FF. 0-3-81-2-0

Sol:-

v)



Q1

MSB

128

00
10
01
11

vi) State table:-

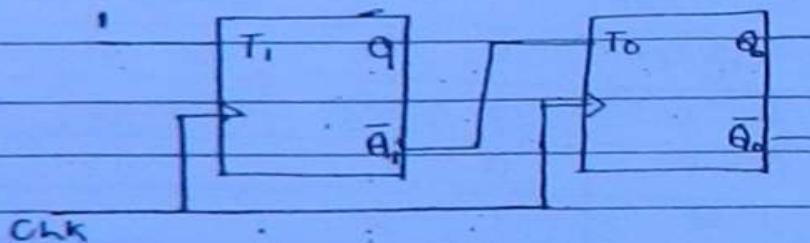
Q_1, Q_0	Q_{1+}, Q_{0+}	T_1	T_0
0, 0	1, 1	1	1
1, 1	0, 1	1	0
0, 1	1, 0	1	1
1, 0	0, 0	1	0

vii) Logical expression:-

$$T_1 = 1$$

$$T_0 = \bar{Q}_1 \bar{Q}_0 + \bar{Q}_1 Q_0 = \bar{Q}_1$$

viii) Implementation:-



Content :-

- ⇒ Various no. system.
- ⇒ Arithmetic operation.
 - complement
 - Add, sub.
- ⇒ Various codes.
- ⇒ Data representation.
 - unsigned
 - signed
 - signed magnitude
 - 1's
 - 2's

(129)

Number system and codes:-

- | | |
|--|---|
| ↓
Weighted
⇒ Positional weighted
e.g. Binary, Octal
Decimal, Hexadecimal
BCD code. | ↓
Unweighted
⇒ Non weighted
e.g. Gray code, Excess-3 code |
|--|---|

- ⇒ A number system with base or radix r contains, r different digit and they are from $(0 - r-1)$.

e.g. $(101)_r$. r = Base or radix.

Base	Different Digit
2	0, 1
8	0, ..., 7
10	0, ..., 9
12	0, ..., 9, A, B
16	0, A, B, C, D, E, F
4	0, ..., 3
6	0, 1, 2, 3, 4, 5

★ Conversion (various number system) :-

1. Decimal to others :-

(130)

⇒ To convert decimal no. into any other base r divide integer part multiply fractional part with r.

e.g.

Q:- convert $(25.625)_{10}$ — $(\text{---})_2$.

Sol:-

$$\begin{array}{r} 2 | 25 \\ 2 | 12 - 1 \\ 2 | 6 - 0 \\ 2 | 3 - 0 \\ 2 | 1 - 1 \\ 0 - 1 \end{array}$$

$$\begin{aligned} 0.625 \times 2 &= 1.25 = 1, \\ 0.25 \times 2 &= 0.50 = 0 \\ 0.5 \times 2 &= 1.0 = 1 \end{aligned}$$

Ans :- $(11001.101)_2$

Q:- Convert $(25.625) — (\text{---})_8$.

Sol:-

$$\begin{array}{r} 8 | 25 \\ 8 | 8 - 1 \\ 0 - 3 \end{array}$$

$$0.625 \times 8 = 5$$

Ans :- $(31.5)_8$.

⇒ When we go from higher to lower base the no. t is increased.

Q:- convert $(25.625)_{10} — (\text{---})_{16}$

Sol:-

$$\begin{array}{r} 16 | 25 \\ 16 | 1 - 9 \\ 0 - 1 \end{array}$$

$$0.625 \times 16 = 10 = A$$

Ans :- $(19.A)_{16}$.

Q:- convert $(254)_{16} — (\text{---})_{16}$.

Sol:-

$$\begin{array}{r} 16 | 254 \\ 16 | 15 \quad 14 = E \\ 0 - 15 = F \end{array}$$

= $(FE)_{16}$

12

Q1 Convert $(27.4)_4 = (?)_4$

Sol:

$$\begin{array}{r} 4 \longdiv{127} \\ 16 \quad \uparrow \\ 16 - 3 \\ \hline 1 \end{array}$$

$$\begin{aligned} 0.4 \times 4 &= 1.6 &= 1 \\ 0.6 \times 4 &= 2.4 &= 2 \\ 0.4 \times 4 &= 1.6 &= 1 \end{aligned}$$

(13)

Ans:- $(123.121)_4$

2 Others to Decimal :-

$$(X_1 X_2 X_3 \dots Y_1 Y_2)_r = (?)_{10}$$

⇒ To convert any other base r to decimal multiply each digit with positional weighted then add.

then,

$$(-)_{10} = X_1 * r^2 + X_2 * r^1 + X_3 * r^0 + Y_1 * r^{-1} + Y_2 * r^{-2}$$

Q1 Convert $(10101.11)_2 = (-)_{2,10}$

Sol:

$$\begin{aligned} &1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} \\ &= 16 + 4 + 1 + \frac{1}{2} + \frac{1}{4} \\ &= \frac{64+16+4+2+1}{4} = \frac{87}{4} = (21.75)_{10} \end{aligned}$$

Q1 convert $(57.4)_8 = (?)_{10}$

Sol:

$$\begin{aligned} &5 \times 8^1 + 7 \times 8^0 + 4 \times 8^{-1} \\ &= 40 + 7 + \frac{4}{8} = \frac{320+56+4}{8} = \frac{380}{8} \\ &= 47.5 = (47.5)_{10} \end{aligned}$$

Q1 convert $(57.4)_{16} = (?)_{10}$

Sol:

$$5 \times 16 + 7 + \frac{4}{16} = 87 + 0.25 = (87.25)_{10}$$

Q1 convert $(BAD)_{16} = (?)_{10}$

Sol:

$$\begin{aligned} &11 \times 16^2 + \cancel{A} \times 10 \times 16 + 13 \\ &= 256 \times 11 + 160 + 13 \\ &= 2816 + 160 + 13 = (2989)_{10} \end{aligned}$$

Q:- convert $(35)_6 = (-)_{10}$

Sol:- $3 \times 6 + 5 \times 1 = 18 + 5 = (23)_{10}$

(132)

3. Octal to Binary & Binary to Octal :-

$(xyz)_8 = (-)_2$

⇒ each no. is represent by its 3 bit binary formate.

Ques:- convert $(37.45)_8 = (-)_2$

Sol:- $(011111.100101)_2$

Binary to octal :-

Ques:- convert $(10110.11)_2$

Sol:- $(010\ 110\ .\ 110)_8$

= $(26.6)_8$.

4. Hexadecimal to Binary and Binary to Hexadecimal :-

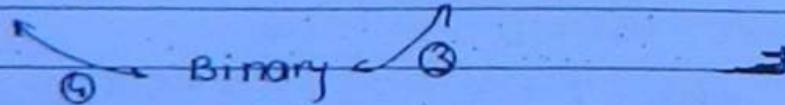
⇒ Each digit is represent by 4 bit binary.

Ques:- convert $(259A)_{16} = (-)_2$

Sol:- $(0010\ 0010\ 1100\ 1010)_2$

5. Hexadecimal to octal or octal to Hexa:-

for Hexa \longleftrightarrow Octadecimel



Q:- convert $(CAD)_{16} = (J8$

Sol:- $(CAD)_{16}$

= $(1100\ 1010\ 1101)_{16,2}$

= $(6255)_8$

★ ARITHMETIC OPERATION :-

(a) Binary Addition, Subtraction, Multiplication :-

(133)

a) Add.

$$\begin{array}{r}
 110110 \\
 + 101101 \\
 \hline
 1100011
 \end{array}$$

b) Sub:-

$$\begin{array}{r}
 11011 \\
 - 10110 \\
 \hline
 00101
 \end{array}$$

c) Multiply:

b)

$$\begin{array}{r}
 1111 \\
 \times 111 \\
 \hline
 1111 \\
 1111 \\
 \hline
 111111 \\
 \hline
 00000 \\
 \hline
 1111 \\
 \hline
 00000 \\
 \hline
 1111 \\
 \hline
 110010
 \end{array}$$

$\rightarrow 4 = \frac{100}{\downarrow}$

$\rightarrow 6 = \frac{110}{\downarrow}$

a)

$$\begin{array}{r}
 1010 \\
 \times 101 \\
 \hline
 1010
 \end{array}$$

$$\begin{array}{r}
 1010 \\
 \times 101 \\
 \hline
 0000 \\
 1010 \\
 \hline
 110010
 \end{array}$$

(B) Octal Addition, subtraction :-

$$0+0 = 0$$

$$7+1 = (8)_{10} =$$

$$\begin{array}{r}
 8 | 8 \\
 8 | 1 \uparrow 0 \uparrow \\
 0 - 1 \uparrow = (10)
 \end{array}$$

$$0+1 = 1$$

$$1+1 = 2$$

$$7+7 = 8 \uparrow 14$$

$$\therefore 1+7 = 10$$

$$8 | 1 \uparrow 6 \uparrow = 16.$$

$$7+2 = 11$$

Sum of 2 octal no!:-

i)

$$\begin{array}{r}
 243 \\
 + 212 \\
 \hline
 455
 \end{array}$$

ii)

$$\begin{array}{r}
 567 \\
 + 243 \\
 \hline
 1032
 \end{array}$$

a)- Subtract :-

i)

$$\begin{array}{r}
 743 \\
 - 564 \\
 \hline
 157
 \end{array}$$

(C) Hexadecimal (Add, Subtraction) :-

(134)

Addition :-

$$1+1 = 2$$

$$1+9 = A$$

$$A+A = (20)_{10} = (14)_{16}$$

$$1+B = C$$

Q:- (i) $\begin{array}{r} 5689 \\ 4574 \\ \hline 9BFD \end{array}$

ii ADD

DAD

$$\begin{array}{r} 188 @ A \\ \hline \end{array}$$

$$\begin{array}{r} 16126 \\ u(1) - 10 \\ \hline 13 \\ 13 \\ \hline 14 \\ 14 \\ \hline 11 \times 124 \end{array}$$

Q: Subtract :-

Sol: (i) $\begin{array}{r} 9748 \\ 587C \\ \hline 3ECEF \end{array}$

ii 9654

$$- 5321$$

$$\underline{4333}$$

(D) Complements :-

$r = \begin{cases} \rightarrow (r-1)'s & \text{complement} \\ \rightarrow r's & \text{complement} \end{cases}$

(135)

Binary $\begin{cases} \rightarrow 1's \\ \rightarrow 2's \end{cases}$

Octal $\begin{cases} \rightarrow 7's \\ \rightarrow 8's \end{cases}$

Decimal $\begin{cases} \rightarrow 9's \\ \rightarrow 10's \end{cases}$

Hexa $\begin{cases} \rightarrow F's \\ \rightarrow 16's \end{cases}$

$(r-1)'s$ complement :-

\Rightarrow Subtract from max. no. to the given no.

e.g. comp. of (1010)

$$= \begin{array}{r} 1111 \\ - 1010 \\ \hline 0101 \end{array}$$

To determine $(r-1)'s$ compliment subtract given no. from max. no. possible in the given base. (max. no. $(r^n - 1)$)

e.g. 1's complement of 101101 is,

Sol:-

$$\begin{array}{r} 111111 \\ - 101101 \\ \hline 010010 \end{array}$$

Q:- determine 7's complement of octal no. 5674.

Sol:-

$$\begin{array}{r} 7777 \\ - 5674 \\ \hline 2103 \end{array}$$

Q:- Determine 9's compliment of decimal 2679.

Sol:-

$$\begin{array}{r} 9999 \\ - 2679 \\ \hline 7320 \end{array}$$

Q:- Det. F's comp. of Hexa. 2689.

Sol:-

$$\begin{array}{r} FFFF \\ - 2689 \\ \hline D976 \end{array}$$

Q:- r's complement :-

To determine r's complement first write (r-1)'s complement then add 1 at LSB. (at right most)

Q:- Det. 2's complement. of 10100.

Sol:

$$\begin{array}{r} 1111 \\ - 10100 \end{array}$$

136

$$\begin{array}{r} 01011 \\ + 1 \\ \hline \end{array}$$

$$\begin{array}{r} 01100 \\ \hline \text{Ans.} \end{array}$$

Q:- Determine 2's complement of 10110.11

Sol: 1's complement = 01001.00

$$\begin{array}{r} + 1 \\ \hline \end{array}$$

$$\begin{array}{r} 01001.01 \\ \hline \text{Ans.} \end{array}$$

Q:- Determine 8's complement of octal 2670.

Sol: T's camp = $\begin{array}{r} 7777 \\ - 2670 \\ \hline 5107 \\ + 1 \\ \hline 5110 \end{array}$

Q:- Determine 10's complement of decimal 5690.

Sol: 9's = $\begin{array}{r} 9999 \\ - 5690 \\ \hline 4309 \\ + 1 \\ \hline 4310 \end{array}$

Q:- Determine 16's complement of Hexadecimal 5289.

Sol: ~~5289~~ = F's = $\begin{array}{r} FF FC \\ - 52 89 \\ \hline AD 76 \\ + 1 \\ \hline AD77 \end{array}$

CODES :-

1. BCD code :-

- ⇒ Binary coded decimal
- ⇒ weighted code.
- ⇒ 4 bit code.
- ⇒ 8421 code.
- ⇒ Each decimal digit with^{is} represented with 4 bit

137

Decimal	BCD	Excess -3 code.
0	0000	0011
1	0001	0100
2	0010	0101
3	0011	0110
4	0100	0111
5	0101	1000
6	0110	1001
7	0111	1010
8	1000	1011
9	1001	1100

1010
1011
1100
1101
1110
1111

} invalid BCD code or. don't care.

- ⇒ During Arithmetic operation if invalid BCD present the add 0110 to get correct result.

A combinational ckrt is applied with 4bit BCD code which is represented as $D_3 D_2 D_1 D_0$, o/p is Y., Y=1 then I/P BCD is divisible by 3. then logical expression for Y is.

	D ₇ D ₆	D ₅ D ₄	D ₃ D ₂	D ₁ D ₀
0000 - 0	1	1	1	1
0011 - 3				
0110 - 6	x	1x	x1	x
1001 - 9	1	x	x	x

(38)

$$Y = \bar{D}_3 \bar{D}_4 \bar{D}_2 \bar{D}_1 + D_8 \bar{D}_6 \bar{D}_2 D_1 \leftrightarrow D_8 D_4 \\ + D_8 D_1 + D_2 D_1 \bar{D}_4 + D_2 \bar{D}_1 D_4$$

$$Y = \bar{D}_8 \bar{D}_4 \bar{D}_2 \bar{D}_1 + D_1 D_8 + D_1 D_2 \bar{D}_4 + \bar{D}_1 D_2 D_4$$

⇒ For write BCD code each digit (decimal) is write separately in BCD.

e.g. $(534)_{10} = (0101\ 0011\ 0100)_{BCD}$

2. Excess - 3 code :-

⇒ Excess - 3 code = BCD + 3

⇒ Unweighted code

⇒ 4 bit code.

Decimal	Excess - 3 code,
0	0011
1	0100
2	0101
3	0110
4	0111
5	1000
6	1001
7	1010
8	1011
9	1100

self complement

⇒ It is self complement code.

⇒ only unweighted code which is self complement is Excess 3-code.

⇒ The code which addition is 9 is self complement code.

e.g. 2421 } weighted } self complemented
 3321 }
 4311 }
 5211

Q: Write 2421 weighted code.

Sol:

decimal 2421

0 0000

1 0001

2 0010

3 0101

4 0100

5 010011

6 10100

7 01101

8 1110

9 1111

139

Self complemen

3. Binary to Gray code:-

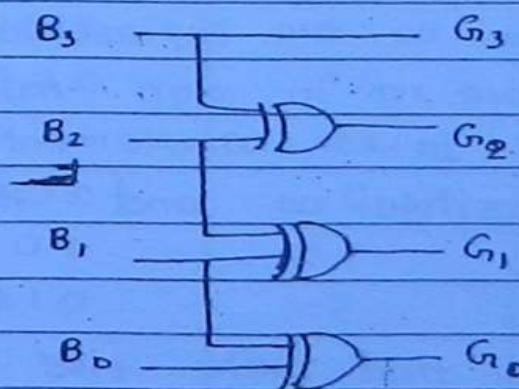
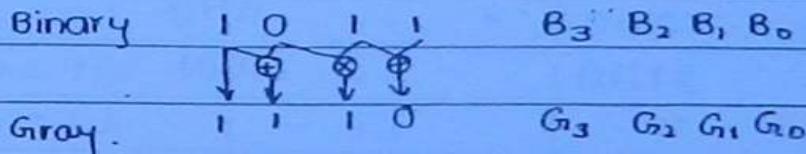
(A) Binary to Gray :-

⇒ Unweighted code .

⇒ Successive no. is differ by 1 bit.

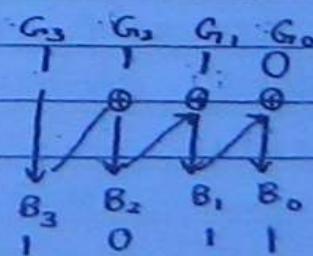
⇒ Also called unit distance code.

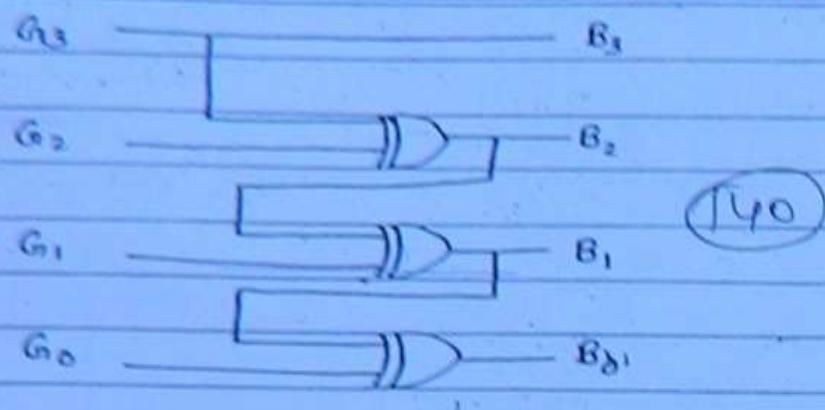
⇒ Also cyclic code, Reflective code. and Minimum error code.



(B) Gray to Binary:-

Bottom: Gray





★ Data Representation :-

Data representation

Magnitude

unsigned

+ive

no sign bit

signed

+ive, -ive

; one extra bit \rightarrow sign.

; sign \rightarrow MSB

; logic - 0 \rightarrow +ive

- 1 \rightarrow -ive.

complement

1's

2's

+ive, -ive

+6	110	0110	-	0110	0110
-6	x	1110	.	1001	1010
	(can't write)				

⇒ In all representation +ive no. are represented in similar way. To represent -ive no. in sign magnitude, only sign bit change. In 1's complement representation of -ive no., first write positive no. and then 1's complement to it.

⇒ And in 2's complement first write +ive no. and then 2's complement to it.

Q. A no. is represent in 1011 the equivalent decimal value.

$$\begin{aligned}
 1011 &= -(0101) = -(0101) \\
 &= -5.
 \end{aligned}$$

Q:- To find $5 - 4$?

Sol:- $5 + (-4)$

$$+5 = 0101$$

$$-4 = 1100$$

$$+5 + (-4) = 0101$$

$$1100$$

(14)

Ans 0001

⇒ In 2's complement addition if any carry present it is discarded.

⇒ In 2's complement to extend no. of bit copy MSB bit.

Q:- Page - (6).

Sol:- $1001 \rightarrow -(0111) = -7$

$$11001 \rightarrow -(001101) = -7$$

$$111001 \rightarrow -(000111) = -7$$

Binary	sign mag	1's	2's
0000	+0	+0	+0
0001	+1	+1	-1
0010	+2	2	-2
0011	+3	3	-3
0100	+4	4	-4
0101	+5	+5	+5
0110	+6	+6	+6
0111	+7	+7	+7
1000	-0	-7	-8 *
1001	-1	-6	-7
1010	-2	-5	-6
1011	-3	-4	-5
1100	-4	-3	-4
1101	-5	-2	-3
1110	-6	-1	-2
1111	-7	-0	-1

4 bit :-

- (i) range of signed mag $-7 \rightarrow +7$
- (ii) " " 1's complement $-7 \rightarrow +7$
- (iii) " " 2's complement $-8 \rightarrow +7$

(142)

⇒ For signed mag. and 1's complement :-

 n bit $\rightarrow -(2^{n-1})$ to $+(2^{n-1} - 1)$

⇒ For 2's complement :-

 n bit $\rightarrow -(2^{n-1})$ to $+(2^{n-1} - 1)$ Q:- Perform $5+4$ using 2's complement.

$$5 = 0101$$

$$4 = \underline{0100}$$

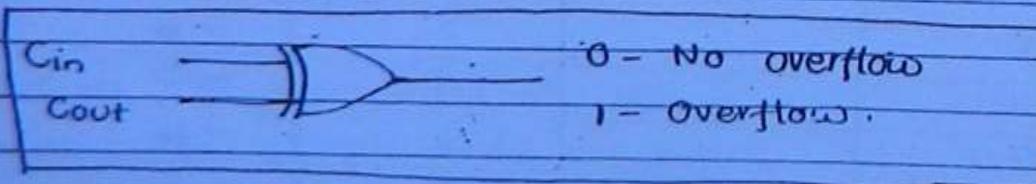
$$\begin{array}{r} 0101 \\ + 0100 \\ \hline 1001 \end{array} *$$

→ overflow may occur when same (two because sign no. are added in signed representation, because for 4 bit we can only represent.

$$-(2^{n-1}) \text{ to } +(2^{n-1} - 1) = (-8 \rightarrow +7)$$

let x and y are sign bit of two no. and z is resultant sign no. then condition for overflow is.

$$Z = \boxed{\bar{X}\bar{Y}z + X\bar{Y}z} \quad \text{condition of overflow.}$$



let Cin = carry into MSB
Cout = carry from MSB

193

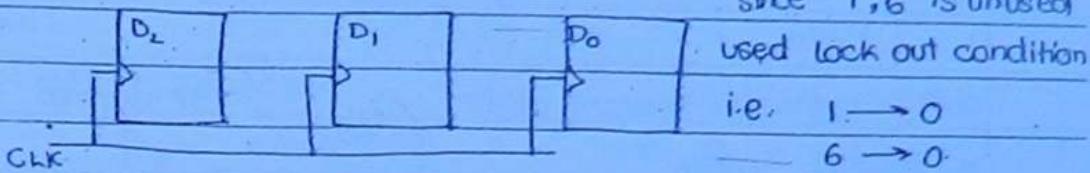
144

Q: Design a synchronous counter using D-FF for the sequence.

$$0 \rightarrow 2 \rightarrow 5 \rightarrow 3 \oplus \rightarrow 4 \rightarrow 7 \rightarrow 0.$$

Sol:

(i)



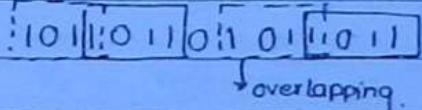
(ii) Truth table:

PS	NS
Q_2, Q_1, Q_0	Q_{2+}, Q_{1+}, Q_{0+}
0 0 0	0 1 0
0 1 0	1 0 1
1 0 1	0 1 1
0 1 1	1 0 0
1 0 0	1 1 1
1 1 1	0 0 0
0 0 1	0 0 0
1 1 0	0 0 0

145

\Rightarrow To avoid lock out change unused states into one of used states in state table.

Ans: 10 or, 31.



\Rightarrow 4 bit \Rightarrow 4 state requires \Rightarrow 2 FF required.

State Machines

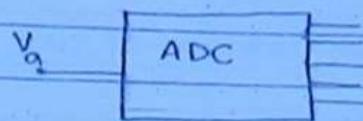
Moore

Mealy

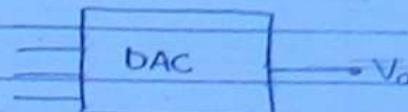
- ⇒ o/p depend on present state
- ⇒ Design easy
- ⇒ More no. of state

(14b)

- ⇒ o/p depend on Present state
- ⇒ Design complex.
- ⇒ less no. of state.



142



- (a) counter type ADC
- (b) SAR type ADC
- (c) parallel comparator type
- (d) dual slope integrating type.

- (i) weighted resistor.
- (ii) R-2R ladder.

(A) Digital to Analog converter (DAC) :-

- (1) Resolution / Step size.
- (2) Analog o/p voltage.
- (3) V_{FS}
- (4) % Resolution
- (5) Error Accuracy

1. Resolution / step size :-

It change in analog voltage corresponding one LSB increment in the I/P.

$$\boxed{\text{Resolution} = \frac{V_r}{2^{n-1}}}$$

where V_r = reference voltage corresponding to logic 1
n = no. of bits

2. Analog o/p voltage :-

$$\boxed{V_{\text{analog}} = \text{Resolution} \times \text{Decimal equivalent of binary data}}$$

Q1. In a 4 bit DAC reference voltage 5V. if binary data 100: is applied then analog voltage is.

Sol:

$$\text{Resolution} = \frac{V_r}{2^{n-1}} = \frac{5}{16-1} = \frac{1}{3}$$

$$V_{\text{analog}} = \frac{1}{3} \times 9 = 3V$$

(3) V_{FS} :-

Full scale voltage is the max. o/p voltage of DAC.

$$V_{FS} = \frac{V_r}{2^n-1} \times 2^n - 1$$

$$V_{FS} = V_r$$

148

(4) % Resolution :-

$$\% \text{ Resolution} = \frac{\text{Resolution}}{V_{FS}} \times 100$$

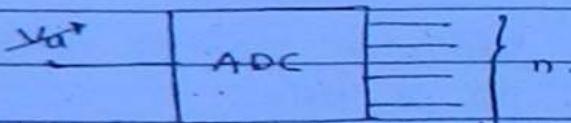
$$\% \text{ Resolution} = \frac{1}{2^n-1} \times 100$$

(5) Errors / Accuracy :-

⇒ error acceptable in ADC's or DAC's is equal to resolution or step size.

(B) Analog to Digital converter :-

Characteristics of ADC's :-



$$(1) \text{ Resolution} = \frac{\text{Range}}{2^n-1}$$

Where,

$$\text{Range} = V_{max} - V_{min}$$

$$(2) \% \text{ Resolution} = \frac{1}{2^n-1} \times 100$$

$$(3) \text{ Dynamic range} = (6n + 1.76) \text{ dB}$$
$$\approx 6ndB$$

Resolution of R-2R ladder type DAC's is :-

$$\boxed{\text{Resolution} = \frac{V_r}{2^n}}$$

(49)

Ques- 5. Page 41.

Sol:-

$$V_{FS} = 10.24$$

$$n = 10$$

$$\text{Resolution} = \frac{10.24}{2^{10}} = 10\text{mV}$$

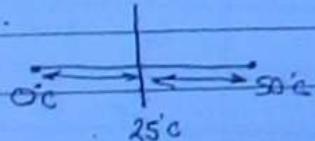
$$\text{error} = \frac{\text{LSB}}{2} = \frac{10\text{mV}}{2} = \pm 5\text{mV}$$

calibrate at 25° i.e. error at 25°C is zero.

$$\Rightarrow \pm 25^\circ\text{C} \Rightarrow 5\text{mV}$$

$$1^\circ\text{C} = \frac{5}{25^\circ\text{C}} \text{mV}$$

$$= 0.2 \text{ mV/C} = 200 \mu\text{V/C}$$

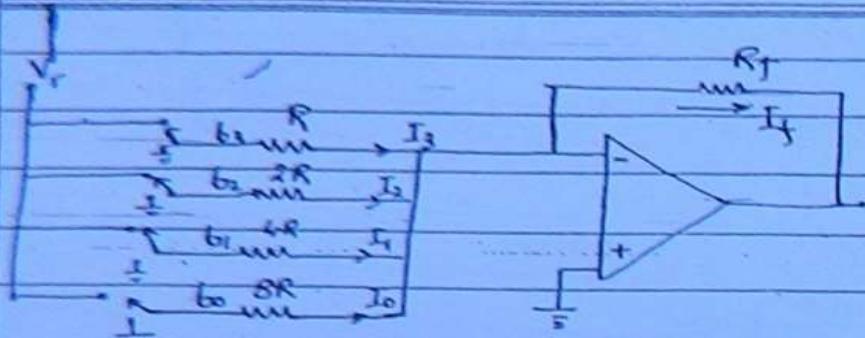


(A) Digital to Analog Circuit :-

Digital to Analog circuits :-

(A) Weighted Resistor DAC :- (4 Bit) :-

150



b_3 = MSB = more current

b_0 = LSB = less current

$$I_3 = \frac{V_r * b_3}{R}$$

$$I_2 = \frac{V_r * b_2}{2R}$$

$$I_1 = \frac{V_r * b_1}{4R}$$

$$I_0 = \frac{V_r * b_0}{8R}$$

$$I_f = I_3 + I_2 + I_1 + I_0$$

$$V_o = -I_f R_f$$

LSB resistance = (2^{n-1}) - MSB resistance.

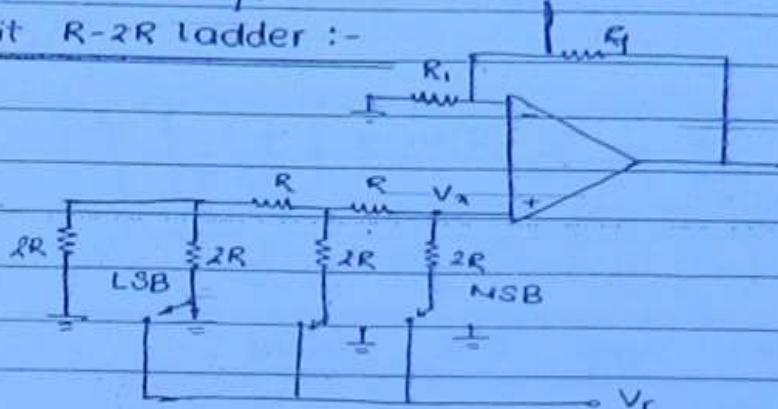
⇒ In weighted resistor DAC the accuracy is less due to use of different resistance.

⇒ To overcome this we use R-2R ladder used.

(B) R-2R ladder :-

→ Normal ladder
→ Inverted ladder.

- * Non inverting
- * Inverting.

(B₁) 3 Bit R-2R ladder :-

(15)

⇒ Adjacent to 2R is LSB.

$$V_o = \left(1 + \frac{R_f}{R_1}\right) V_A$$

V_A = Resolution * Decimal equivalent of binary data.

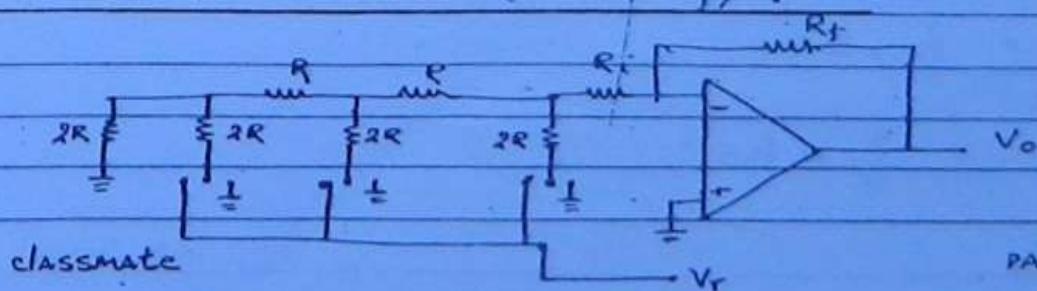
$$= \frac{V_r}{2^n} \times \sum_{i=0}^{n-1} 2^i b_i$$

(decimal equivalent = $b_2 b_1 b_0$ (binary data))

$$\Rightarrow b_2 2^2 + b_1 2^1 + b_0 2^0 = \sum_{i=0}^{n-1} 2^i b_i$$

$$V_o = \frac{V_r}{2^n} \times \sum_{i=0}^{n-1} 2^i b_i \times \left(1 + \frac{R_f}{R_1}\right)$$

V_o = Resolution * Decimal * gain.

(B₂) 3 Bit R-2R Ladder (Inverting) :-

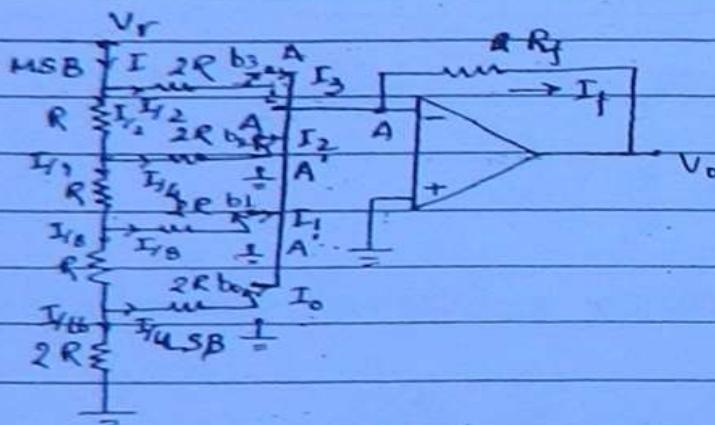
$V_o = \text{Resolution} * \text{decimal} * \text{gain}$.

$$V_o = V_r \times \frac{2^n}{2^n} \sum_{i=0}^{n-1} 2^i b_i \times \left[-\frac{R_f}{R_i + R} \right]$$

(152)

$$I_f = \frac{V_r}{2^n} \times \sum_{i=0}^{n-1} 2^i b_i \times \left[\frac{1}{R_i + R} \right]$$

Q3) Inverted ladder type DAC circuit :-



Since A and A' both are ground then (logical or virtual ground and ground) the switch is at same potential then charging and discharging of switch problem removed in previous ckt.

$$I = \frac{V_r}{R}$$

$$I_3 = \frac{D_0}{b_3} \frac{I}{2} \times b_3$$

$$I_2 = \frac{I}{4} \times b_2$$

$$I_1 = \frac{I}{8} \times b_1$$

$$I_0 = \frac{I}{16} \times b_0$$

$$I_f = I_0 + I_1 + I_2, I_3$$

$$= \frac{I}{16} [8b_3 + 4b_2 + 2b_1 + b_0]$$

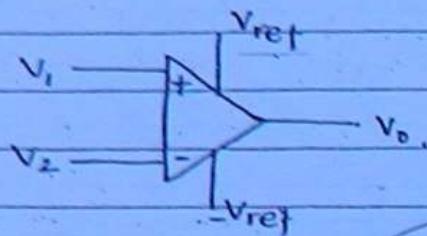
$$= \frac{V_r}{2^n} \left(\sum_{i=0}^{n-1} 2^i b_i \right) \times \frac{1}{R} \quad (153)$$

$$I_f = \frac{V_r}{2^n} \left(\sum_{i=0}^{n-1} 2^i b_i \right) \frac{1}{R}$$

$$V_o = \frac{V_r}{2^n} \left(\sum_{i=0}^{n-1} 2^i b_i \right) \left(-\frac{R_f}{R} \right)$$

Analog to Digital circuit :-

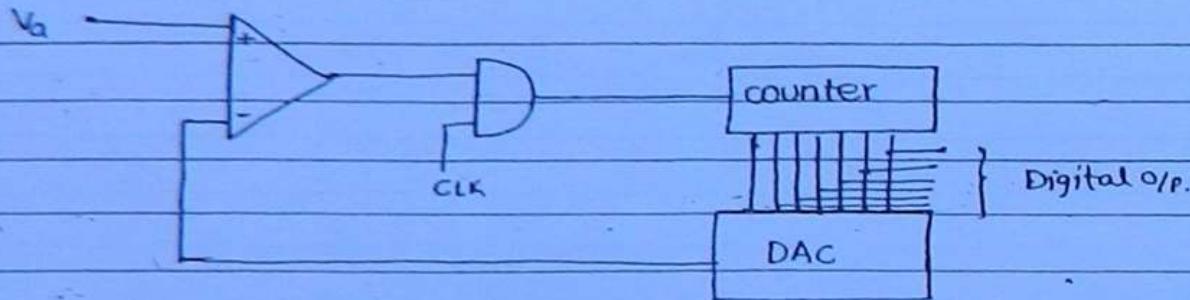
(A) Counter type ADC :-



$$V_1 > V_2 \Rightarrow V_d = V_{ref}$$

$$V_1 < V_2 \Rightarrow V_d = -V_{ref}$$

⇒ It is one bit quantizer.



- ⇒ In counter type ADC a comparator is used in I/P stage to compare - I/P analog voltage with reference voltage provided by DAC feedback.
- ⇒ A counter is used to count no. of clock pulses applied when analog voltage (V_a) is greater than DAC voltage then o/p is 1. Then counter count and if analog voltage (V_a) is less than reference voltage (DAC voltage) then o/p is 0 and counter stops counting and it give the comparative digital o/p.
- ⇒ Max. no. of clock pulses required for N bit conversion is $2^n - 1$.
- ⇒ Max. conversion time = $(2^n - 1) T_{CLK}$.
- ⇒ Conversion time depends on I/P analog voltage.
- ⇒ Also called Ramp type ADC

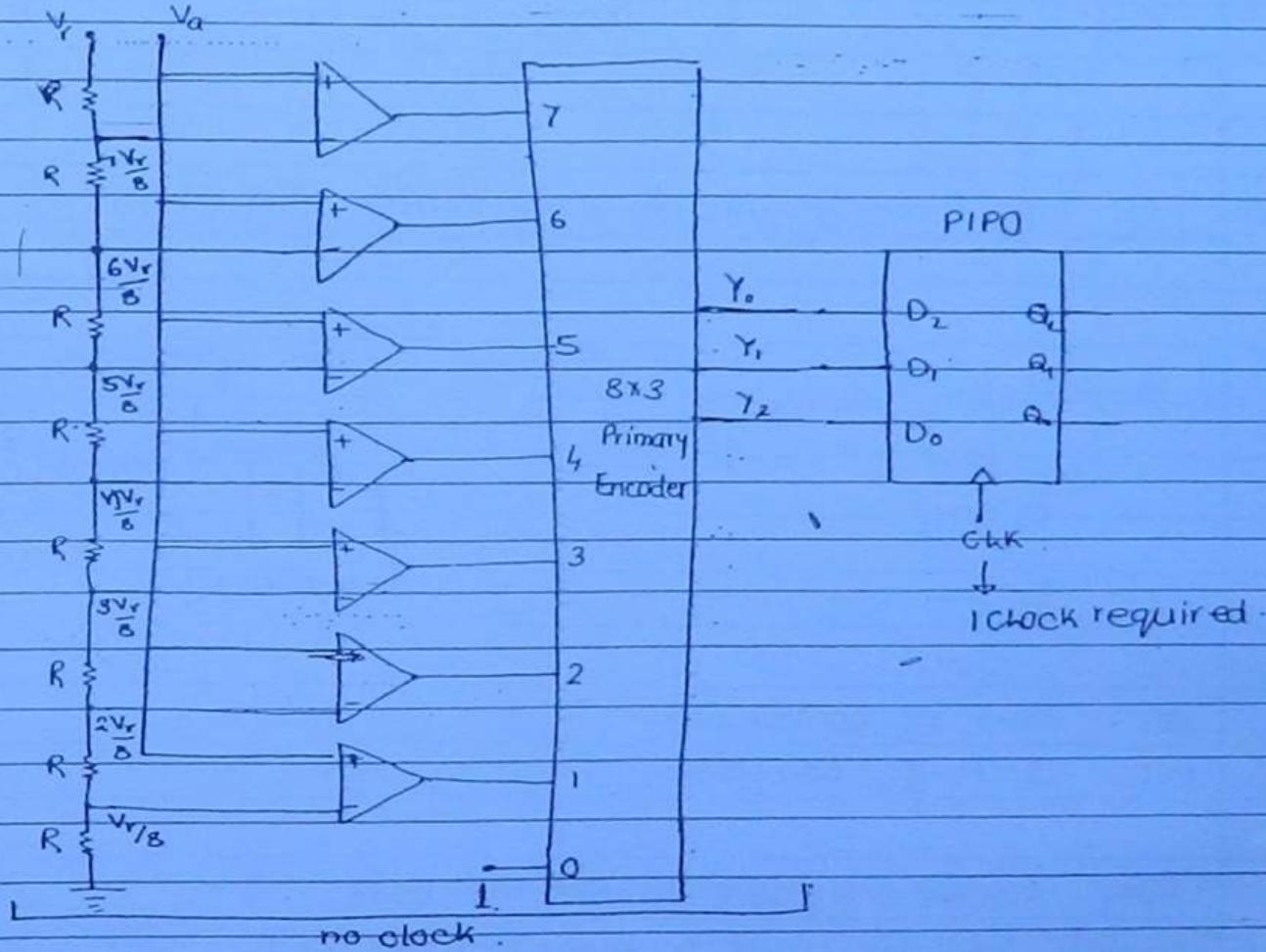
(B) Parallel comparator type :-

- For n bit
 - 2^{n-1} comparators required.
 - 2^n resistor required.
 - $2^n \times n$ priority encoder.

Also called Flash ADC (fastest ADC).

(B.) 3 Bit parallel comparator :-

(15)



⇒ No clock pulse is required.

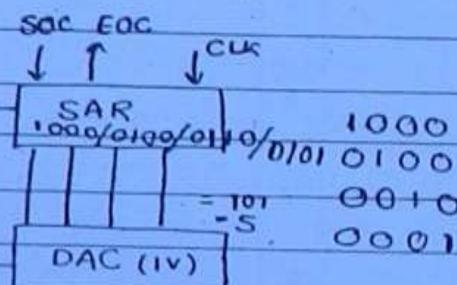
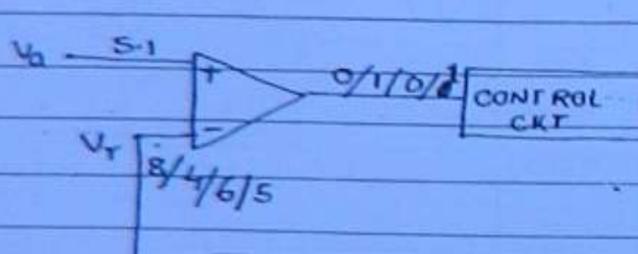
⇒ Therefore it is fastest ADC among all.

⇒ Max no. of clock pulse required for n bit conversion is which is Inside PIPD.

Range of analog	O/P.
$V_a > 7V_r/8$	111
$7V_r/8 > V_a > 6V_r/8$	110
$6V_r/8 > V_a > 5V_r/8$	101
$5V_r/8 > V_a > 4V_r/8$	100
$4V_r/8 > V_a > 3V_r/8$	011
$3V_r/8 > V_a > 2V_r/8$	1010
$2V_r/8 > V_a > V_r/8$	001
$V_r/8 > V_a$	000

(156)

(2) SAR Type (Successive approximation Register) :-



SOC - Start of conversion

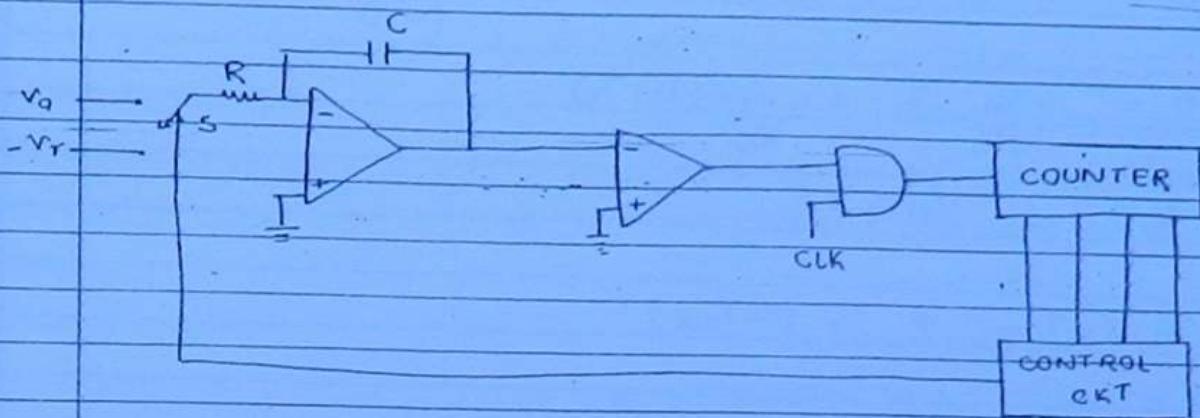
EOC - End of conversion

- ⇒ Ring counter is used to set the base.
- ⇒ Control ckt is used to reset ($V_a < V_r$)
- ⇒ In SAR Type ADC , ring counter will present to successively set the base.
- ⇒ Control ckt is used to reset ; previously set bit when $V_a < V_r$.
- ⇒ In BAR Type ADC , n clock pulse required for n bit conversion.
- ⇒ Conversion time = $n T_{CLK}$
- ⇒ SAR Type, conversion time uniform for any analog voltage (conversion time is independent of PAGE)

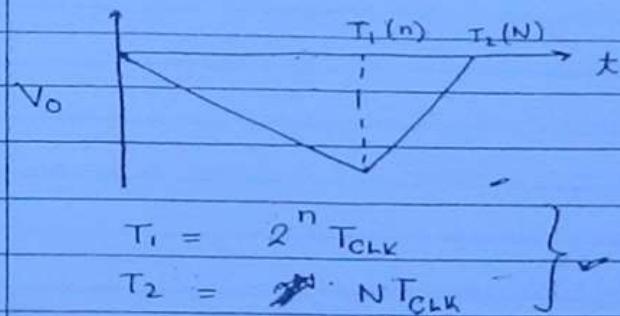
- ⇒ SAR is mostly used in digital ckt to provide interface with microprocessor.

(159)

- (D) Dual slope Integrating type ADC :-



⇒ V_r slope is always greater than V_a slope.



- ⇒ In Dual slope a counter is used to count clock pulse
 ⇒ conversion started initially counter is reset to zero
 - and switch S is connected to V_a (analog voltage) wh integrator is integrating analog voltage o/p of integrator will become -ive voltage due to this comparator o/p i
 1. and counter continues clock pulses after 2^m CLK pulses again counter value became zero at this time t_1 control ckt connect switch S to $-V_r$. During V_r integration upto T_2 time o/p of integrator is -ive. due to this counter again continue clock pulses. at time T_2 o/p of integrator became +ve and comparator o/p became 0 due to this counter

will stops. till N is count when counter stops.

Then,

$$V_o = -\frac{V_a}{RC} \cdot T_1 + \frac{V_r}{RC} (T_2 - T_1)$$

at time $t = T_2$, $V_o = 0$.

158 |

$$\Rightarrow 0 = -\frac{V_a}{RC} \cdot T_1 + \frac{V_r}{RC} (T_2 - T_1)$$

$$\Rightarrow V_a T_1 = V_r (T_2 - T_1)$$

$$\Rightarrow V_a \cdot 2^n \cdot T_{CLK} = V_r (N T_{CLK})$$

$$\Rightarrow N = \frac{V_a \cdot 2^n}{V_r} = \frac{V_a 2^n}{V_r}$$

$$\Rightarrow V_a = \frac{V_r}{2^n} \cdot N$$

$$\boxed{V_a = \frac{V_r}{2^n} \cdot N}$$

If $V_r = 2^n$ then,

$$\boxed{V_a = N}$$

\Rightarrow This is the most accurate ADC among all.

\Rightarrow All ripple and noise is separated or compressed by capacitor. (Therefore this have more accuracy due to integrator).

$$\Rightarrow \text{max. no. of clock pulse} = 2^n + 2^n - 1 \\ \approx 2^n + 2^n = 2^{n+1}.$$

\Rightarrow It is slowest among all.

Application:-

\Rightarrow Mostly used in digital voltmeter.

Clock pulse :-

i) Counter type = $2^n - 1$

ii) Flash = 1

iii) SAR = n

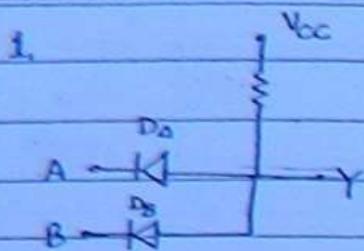
iv) Dual type = 2^{n+1}

DATE

159

classmate

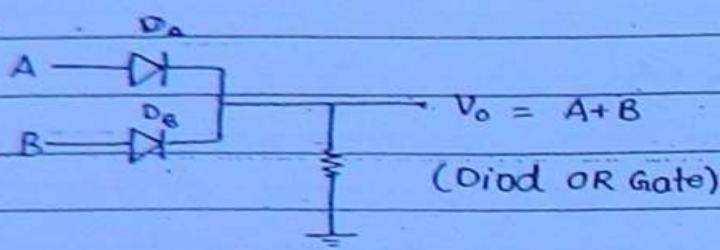
PAGE



(Diode AND Gate)

A	B	D _A	D _B	Y
0	0	ON	ON	0
0	1	ON	OFF	0
1	0	OFF	ON	0
1	1	OFF	OFF	1

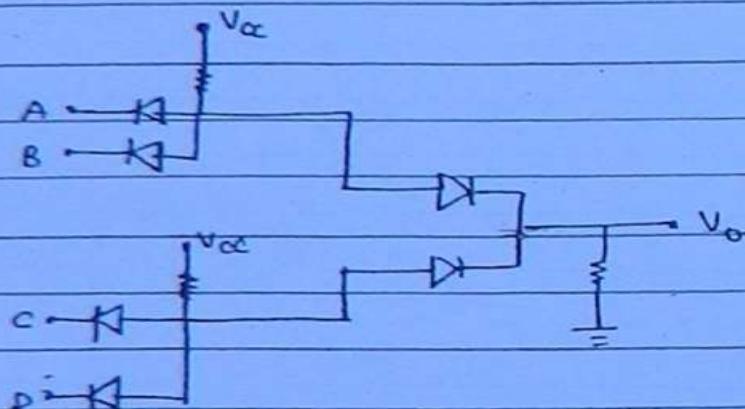
2.



(Diode OR Gate)

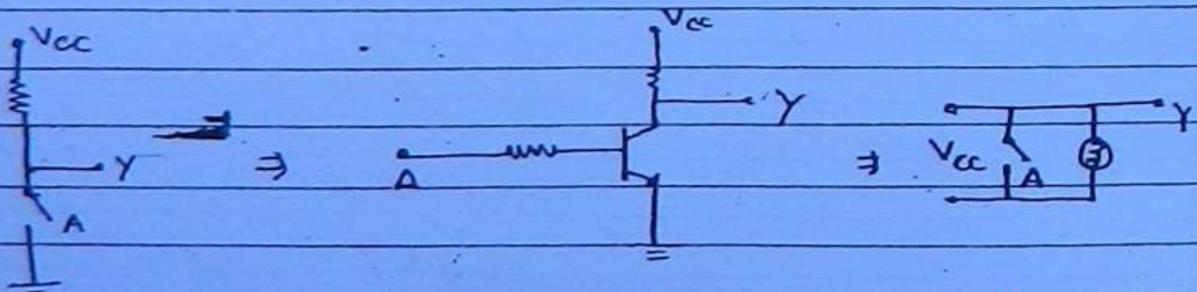
160

⇒ For NOT gate we use transistor.

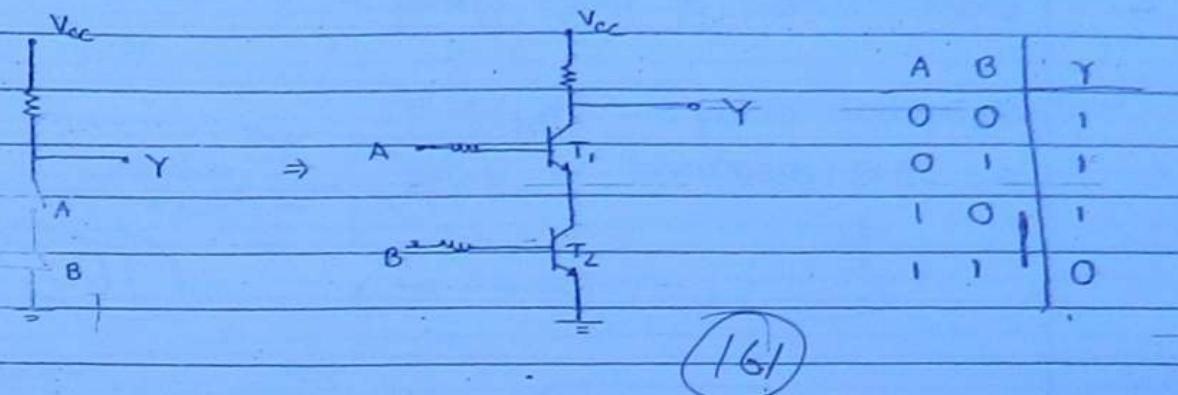


$$V_O = AB + CD.$$

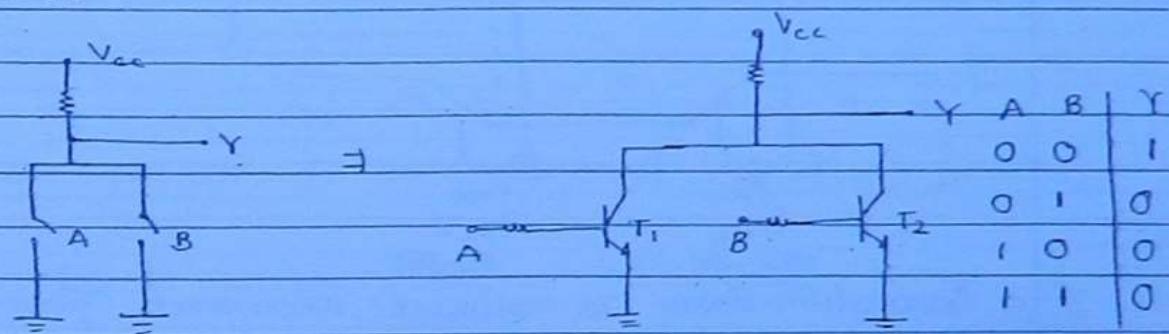
3. NOT :-

if $A = 0$, T_r is cutoff, $Y = L$.if $A = L$, T_r is sat, $Y = 0$

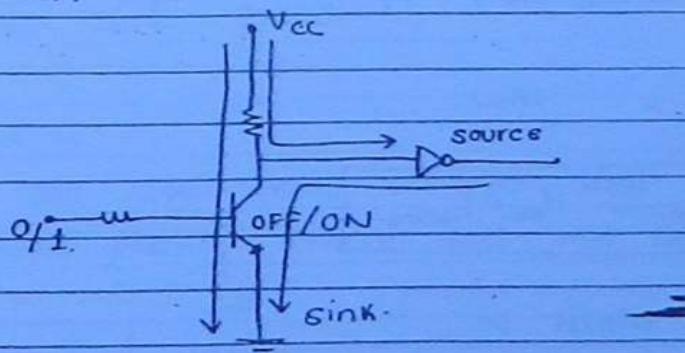
4. NAND Gate :-



5. NOR Gate :-



- ⇒ When logic gate o/p is 1. (Tr. OFF) it will act as current source.
- ⇒ When logic gate o/p is 0 (Tr. ON) it will act as current sink.



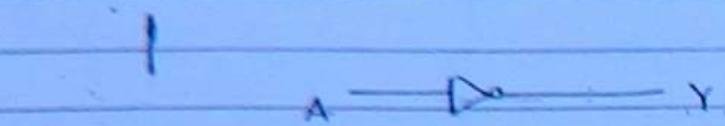
- ⇒ In cutoff and saturation region transistor will act as switch.

J _E	J _C	Region
R _B	R _B	CUTOFF
R _B	F _B	Reverse active
F _B	R _B	Active
F _B	F _B	SATURATION

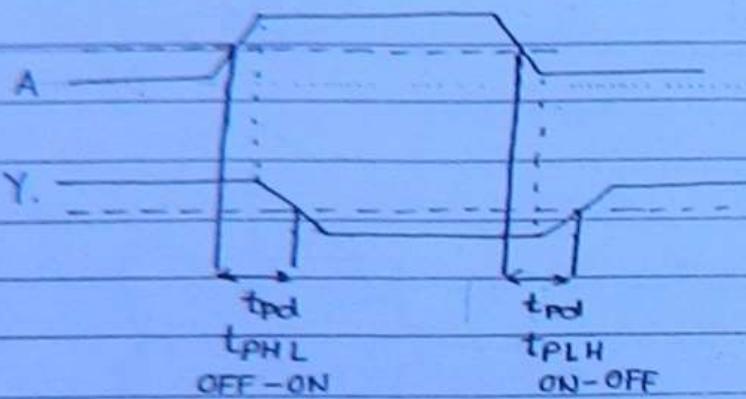
classmate PAGE

Characteristics of logic family :-

- (i) Propagation delay :- (t_{pd}) :-
⇒ It is measured in nsec.



162



$$t_{pd} = \frac{t_{PHL} + t_{PLH}}{2}$$

- ⇒ Propagation delay is always measured from 50% value of the diag.
⇒ In Tr, ON to OFF time is more compare to OFF to ON time due to saturation or storage time.

(ii) Power dissipation :-

- ⇒ Power dissipation by each logic gate.

$$\Rightarrow P_{diss} = mW.$$

$$P_{diss} = V_{cc} \cdot I_{avg}$$

(iii) Figure of Merit :-

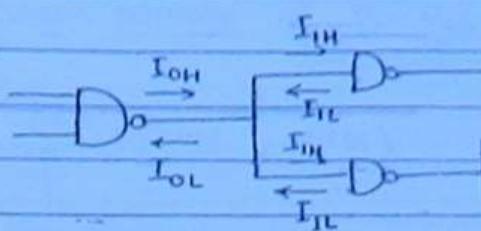
$$FOM = P_{diss} * t_{pd} = P_{diss} \cdot t_{pd}$$

⇒ I^2L have best FOM.

⇒ Value of FOM is low the logic family is best

(iv) Fan out :-

It is max. no. of logic gate that can be given by 1 logic gate.



163

$$\text{fanout}_H = \frac{I_{OH}}{I_{OL}}$$

$$\text{fanout}_L = \frac{I_{OL}}{I_{LL}}$$

\Rightarrow Max. fan out is min value of (fanout_H , fanout_L).

Ques:- if $I_{OH} = 400\mu A$, $I_{IH} = 40\mu A$, $I_{OL} = -16mA$, $I_{IL} = 1.6$
find fanout.

$$\text{fanout}_H = \frac{400}{40} = 10$$

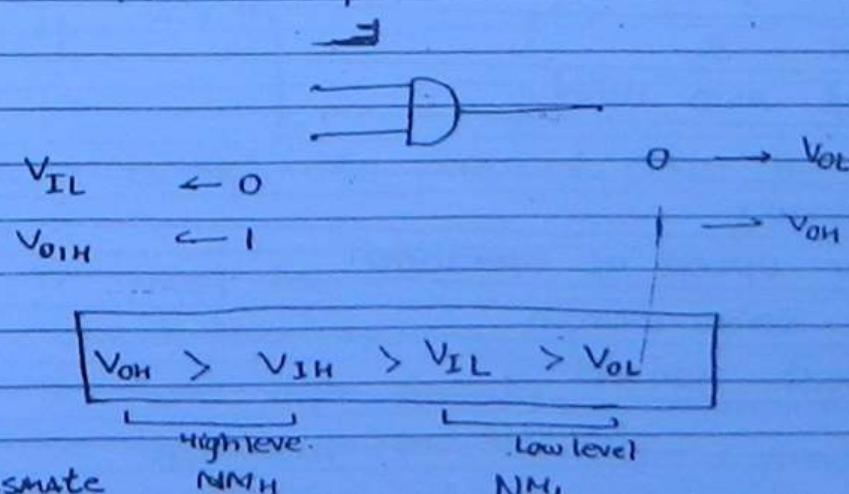
$$\text{fanout}_L = \frac{16}{1.6} = 10$$

$$\text{max. fan out} = (10, 10)_{\min} = 10$$

\Rightarrow TTL have max. fanout.

(v) Noise Margin:-

It is the max. noise voltage that can be added to the logic family which will not affect the o/p.



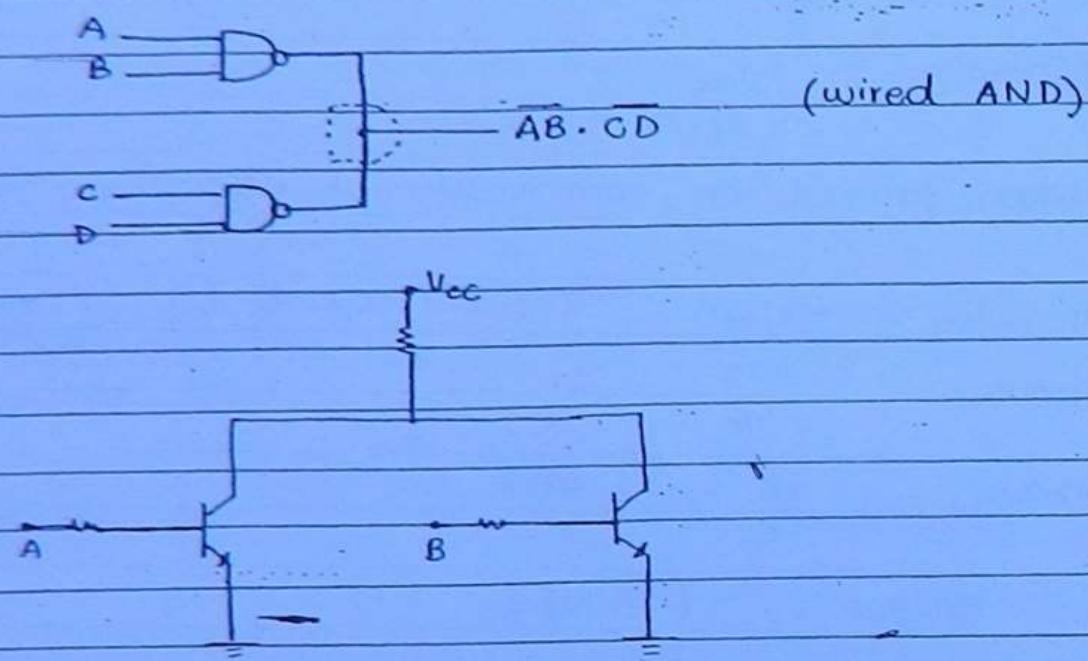
$$NM_H = V_{OH} - V_{IH}$$

$$NM_L = V_{IL} - V_{OL}$$

overall noise margin = $(NM_H, NM_L)_{min.}$

(164)

(A) RTL (Register Transistor logic) family :-



⇒ Basic gate - NOR gate.

⇒ $t_{PD} = 50\text{ns}$

⇒ $P_{diss} = 10\text{ mW}$

⇒ $FOM = 500\text{ PJ}$

⇒ $NM = 0.2\text{ V}$

⇒ Fanout = 3

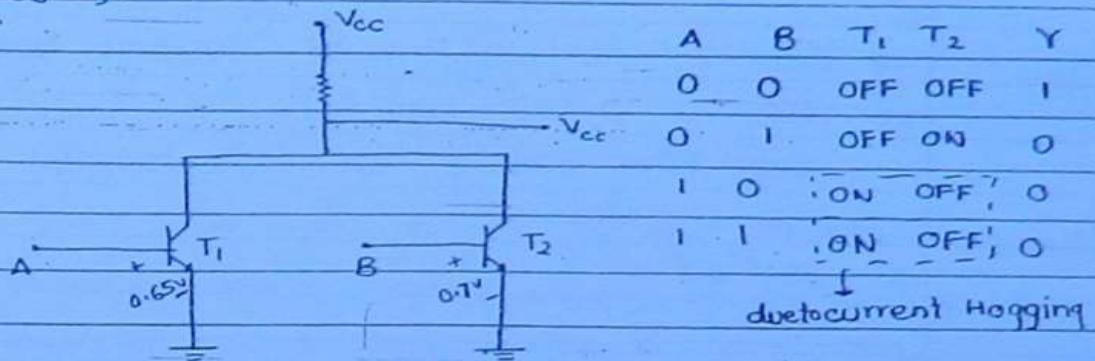
⇒ wired AND used.

Disadvantage :-

1. lower speed of operation.
2. low Noise margin
3. lowest fan out.

(B) DCTL (Direct coupled Transistor logic) :-

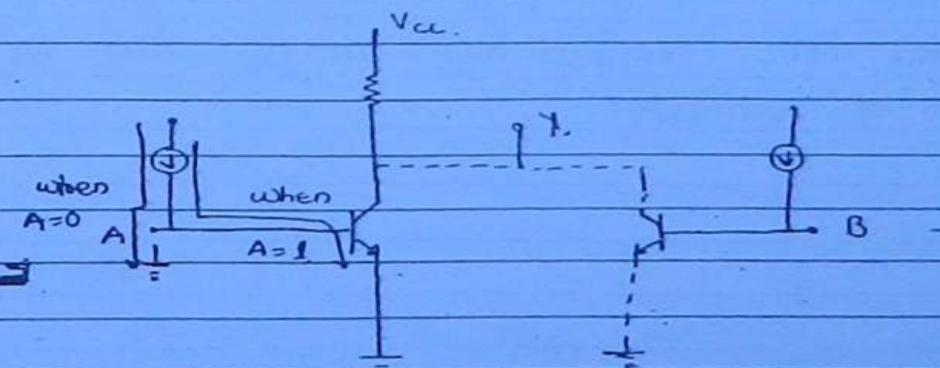
- ⇒ In RTL logic family if V_P resistance removed then result is DCTL.
- ⇒ $B \cdot t_{pd} = 40 \text{ nsec}$.
- Disadvantage :-
- ⇒ Current Hogging.



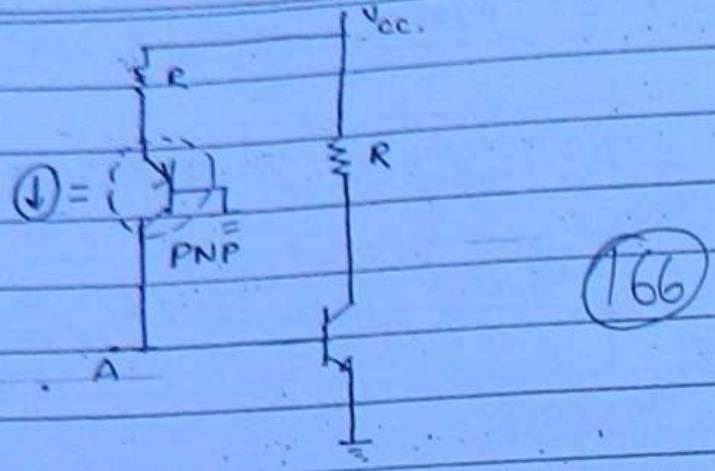
- ⇒ In DCTL logic, If tr. switch different characteristics are used then Tr having lower $V_{BE(SAT)}$ then first ON and it will not allow other Tr to ON. this phenomenon is known as current Hogging.

Integrated Injection logic (I²L) :-

- ⇒ It's injecting the current into Base.



- ⇒ When A is high the current flows through the base of Tr.
- ⇒ i.e. T₁ must be ON.
- ⇒ I²L covers less space.
- ⇒ ie I²L have high density.
- ⇒ It is equivalent to NOT gate.



(166)

- ⇒ There is no problem of current Hogging.
- ⇒ FOM = $0.1 \text{PJ} - 0.7 \text{PJ}$.
- Best FOM among all logic family.
- ⇒ $t_{pd} = 40 \text{ns}$.
- ⇒ Fan out = 8.

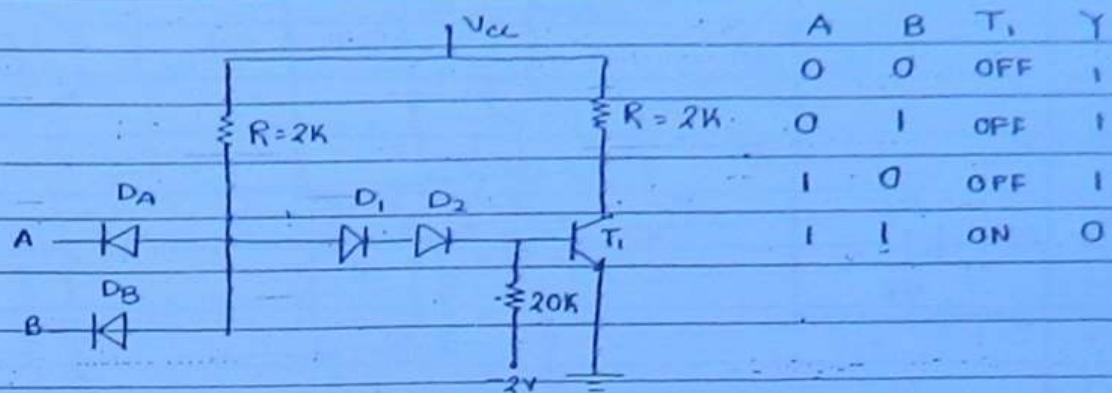
SSI	-	1-12	no of gates used in this integration.
MSI	-	13-99	
LSI	-	100-1000	
VLSI	-	>1000.	

- ⇒ In I^2L logic, due to integration of PNP and NPN tr. it occupies less area hence density are more in I^2BL logic. It is mostly used in MSI and LSI logic family.
- ⇒ Also called MTL (Merged logic family) due to integration of Transistor.

DTL (Diode Transistor logic) family :-

167

- ⇒ AND Gate followed by NOT gate.



- ⇒ 20k resistor used only for discharging the junction capacitance. The capacitance which is discharge is Transition cap. C_c .

⇒ The circuit is called Basic DTL gate.

- ⇒ In this any one of the I/P is low. or all the I/P are low. D_A and D_B will become forward bias whereas D_1 and D_2 will become reverse bias. due to it T_1 is OFF and O/P is 1.

- ⇒ when all the I/P's are high then D_A and D_B become reverse bias and D_1 and D_2 will become forward bias and T_1 is ON and O/P is low.

- ⇒ The basic gate is NAND gate.

⇒ $t_{pd} = 30\text{ns}$.

⇒ $P_{diss} = 8\text{mW}$.

⇒ FOM = 240 PJ

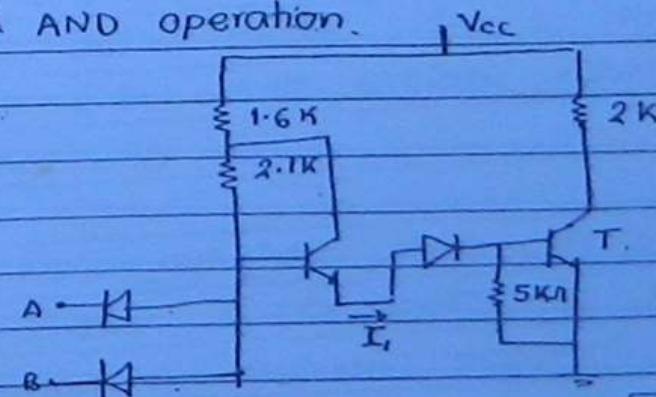
⇒ NM = 0.75 V

⇒ Fanout = 3.

⇒ It provides wired AND operation.

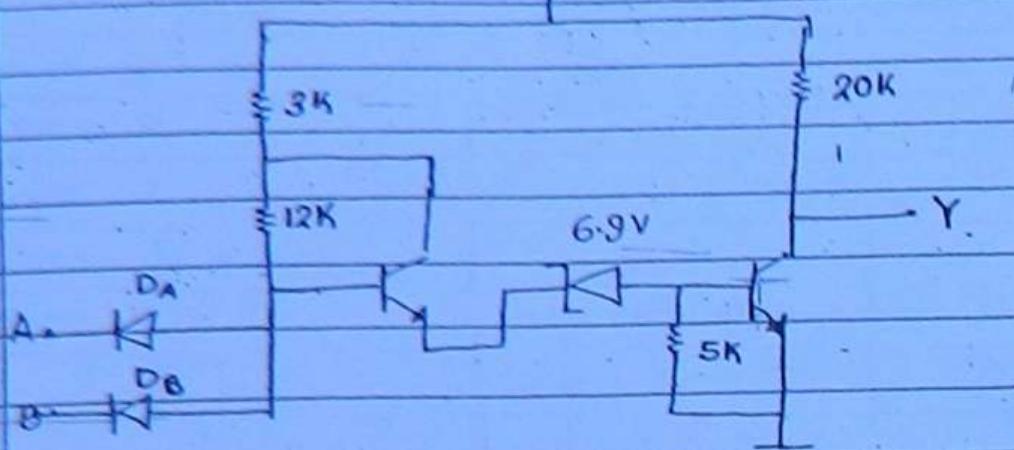
⇒ To increase fanout we introduce T_r in place of Diode.

⇒ 5kΩ resistor used to lower the I_t current.



High Threshold logic (HTL) family :-

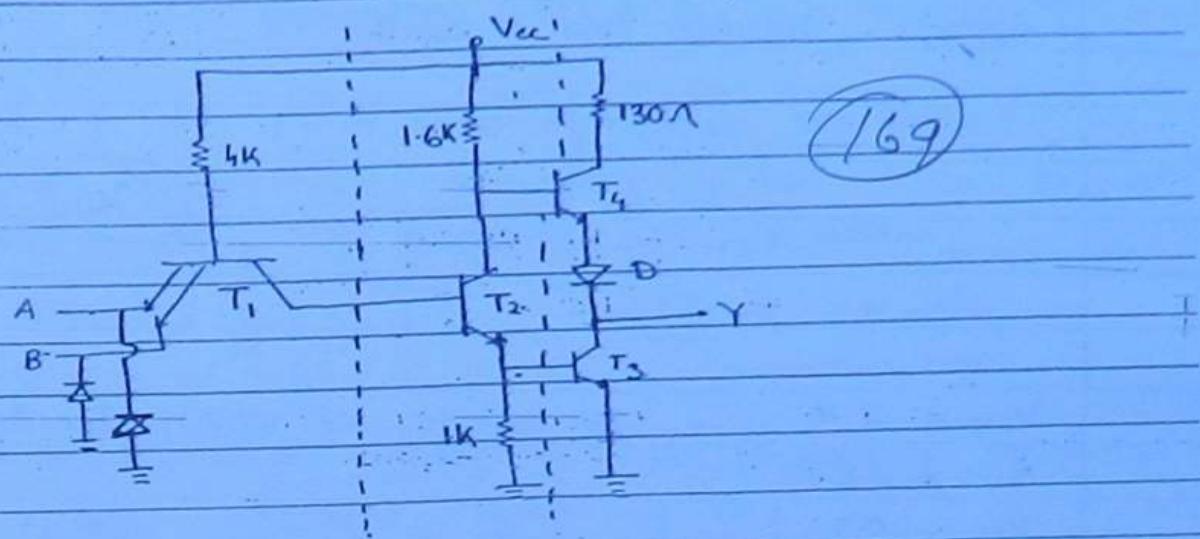
V_{cc} = 15V



(168)

- ⇒ Zener Diode is used in place of D₂
- ⇒ NM = 4 - 5V (Highest noise margin)
- ⇒ since in DTL all diode and Transistor is -ive temp coefficient ($\frac{dV}{dT} = 0 - 2.5 \text{ mV}/^\circ\text{C}$)
- ⇒ logic 0 = 2V } Higher voltage swing
logic 1 = 12V.
- ⇒ t_{pd} = 90 ns
- ⇒ P_{diss} = 55 mW
- ⇒ FOM = 4950 PJ \times 5000 PJ
- ⇒ Fanout = 8
- ⇒ Basic gate = NAND gate
- ⇒ Noise margin = 4V - 5V

TTL (Transistor Transistor logic) family :-



(169)

T_1 = Multiemitter transistor.

A	B	T_1	T_2	T_3	T_4	Y
0	0	A	C	C	S	1
0	1	A	C	C	S	1
1	0	A	C	C	S	1
1	1	R	S	S	C	0

⇒ The ckt shown in fig. is standard TTL logic family. it basically have three stage.

(i) Multiemitter I/P stage

(ii) Phase splitter

(iii) Totem pole or, active pull up. O/P stage.

active = use of T_4

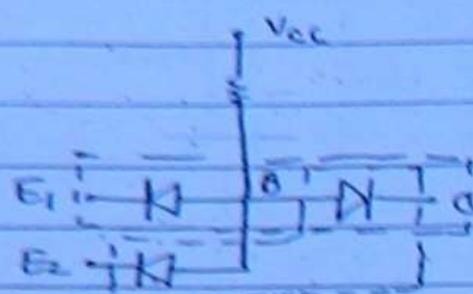
Pullup = T_4 (T_4) connect to V_{cc} .

Operation :-

⇒ Any one of I/P low or all I/P's are low- then EB junction is FB. ($J_E = FB$) and collector base ($J_C = RB$) is RB. T_4 is in active mode. due to this T_4 , T_2 and T_3 are OFF (in cutoff region) where as T_4 is SAT. Hence O/P is 1.

⇒ When all the I/P's are high then J_E (EB junct") of T_1 is RB, and J_C (CB junct") is FB. (The mode of operation is Reverse active.)

DATE _____
T₂ and T₃ are in saturation and T₄ is in cutoff. Hence O/P is zero.



(170)

V_{INH} = 2 V → I/P voltage at which T₄ takes logic

$$V_{OH} = 2.4 \text{ V}$$

$$V_{IL} = 0.8$$

$$V_{IH} = 0.4$$

$$t_{pd} = 10 \text{ ns}$$

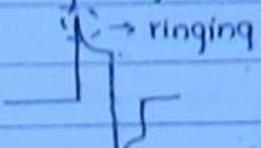
$$P_{diss} = 10 \text{ mW}$$

$$FOM = 100 \text{ PJ}$$

$$\text{Fanout} = 10$$

$$NM = 0.4 \text{ V}$$

- ⇒ Diode D is used to cutoff T₁, T₄ when T₃ is ON.
- ⇒ Advantage of Totem pole :-
 1. lower power dissipation.
 2. Higher speed of operation.
 3. Higher fan out.
- ⇒ Disadvantage of Totem pole :-
It is not used in wired logic
- ⇒ To provide wired AND logic open collector configuration is used.

- ⇒ 130 Ω resistor used in collector in O/P stage to reduce ripple or noise generation. to in high frequency of operation.
- ⇒ In TTL if any I/P is open it behaves as logic L.
- ⇒ Clamping diodes are connected in I/P stage to protect transistor during high frequency of operation.

- ⇒ Clamping D removes ringing of high frequency operation.
- ⇒ There are different type of TTL:-
 - (a) standard TTL
 - (b) High speed
 - (c) Low power
 - (d) Schottky TTL.

(171)

High speed TTL :-

In standard TTL logic family if Resistor value reduce then t_{pd} reduces and known as High speed logic family.

$$t_{pd} = 6 \text{ nsec.}$$

\Rightarrow Power dissipation increases.

(172)

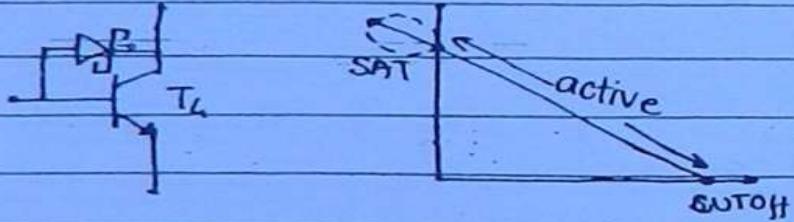
low speed power TTL :-

In TTL logic family if Resistor value increased then power dissipation reduced and resultant is known as low power logic family.

Schottky diode :-

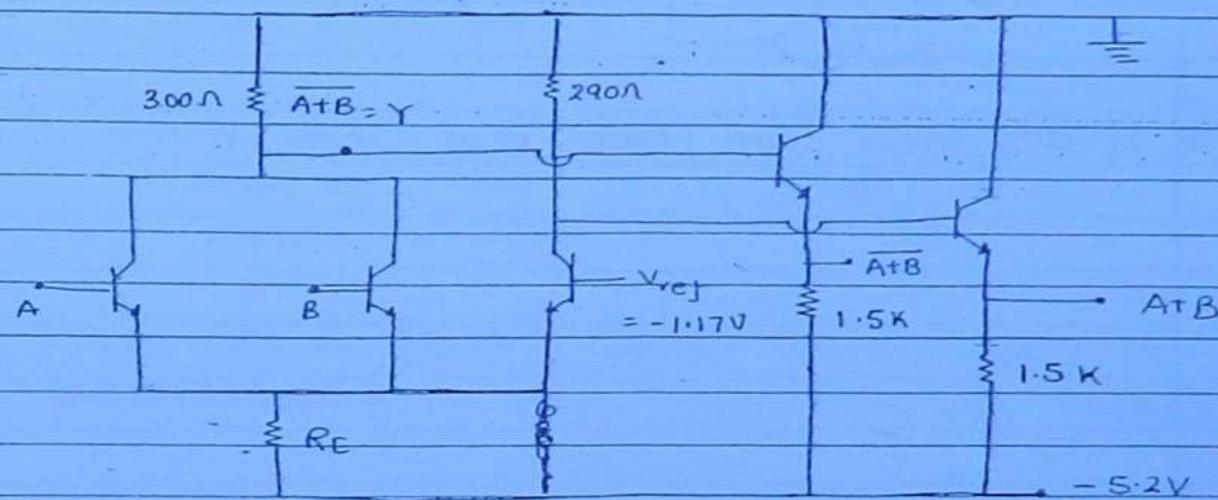
If schottky diode is used b/w collector and base region then it will remove storage time and saturation delay. the family known as schottky diode TTL.

$$t_{pd} = 2 \text{ nsec.}$$



ECL (Emitter coupled logic family) :-

- ⇒ It is never go in saturation region.
- ⇒ work only in cutoff and Active region. (173)
- ⇒ It is fastest logic family due to work in Active and cutoff region. (Because it is non saturated)



$$t_{pd} = 1 \text{ nsec}$$

fanout = 25

- ⇒ It basically contains two stage.
- (1) Differential amp I/P stage.
- (2) CC or Emitter follower O/P stage.
- ⇒ Due to use of D.A. complementary o/p are available in ECL logic family. (NOR/OR) gate.
- ⇒ Due to use of CC stage in the o/p fanout is high
- ⇒ Negative spikes do not affect the transistor due to -ive power supply.
- ⇒ ECL uses -ive power supply. Due to this any spikes or negative voltage not affect operation.

$$t_{pd} = 1 \text{ ns}$$

$$P_{diss} = 55 \text{ mW}$$

$$FOM = 55 \text{ P.J}$$

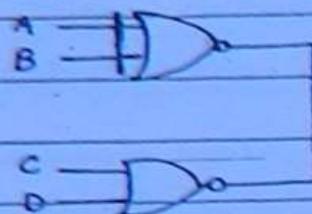
$$\overline{\text{Fanout}} = 25$$

$$N.M. = 0.3V$$

$$\begin{aligned} \text{logic.0} &= -1.7V \\ \text{logic.1} &= -0.85V \end{aligned}$$

It is logic 1 mode only
voltage supply is negative.

⇒ ECL provide wired AND logic

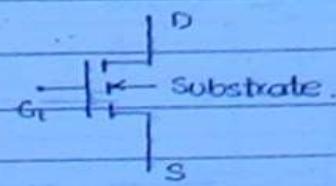
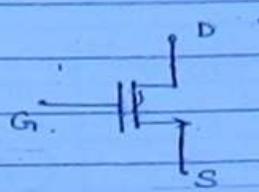


$$(A+B) + (C+D)$$

(174)

⇒ If any I/P is open then it is logic '0'.

NMOS :-

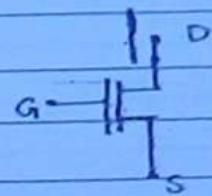


(175)

N-channel :-

logic '0' = OFF

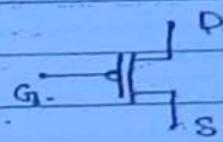
logic 1. = ON.



P-channel MOS :-

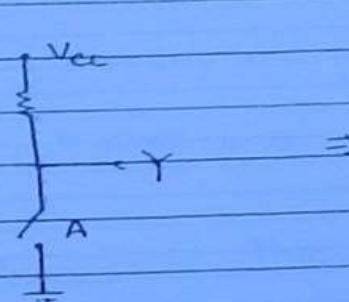
logic '0' = ON

logic 1. = OFF.

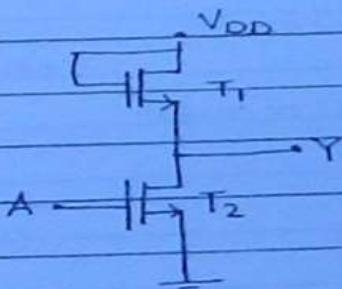


⇒ Since FET is voltage variable resistor hence in MOS circuit in place of reg' resistor we use MOSFET.

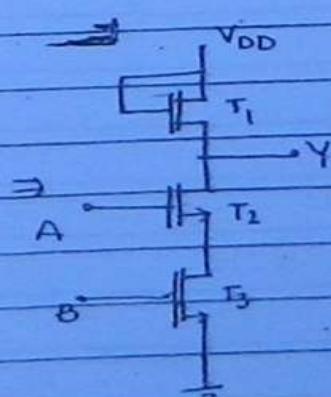
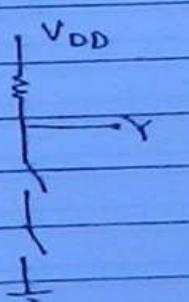
NMOS NOT Gate :-



A	T ₂	Y
0	OFF	1
1	ON	0

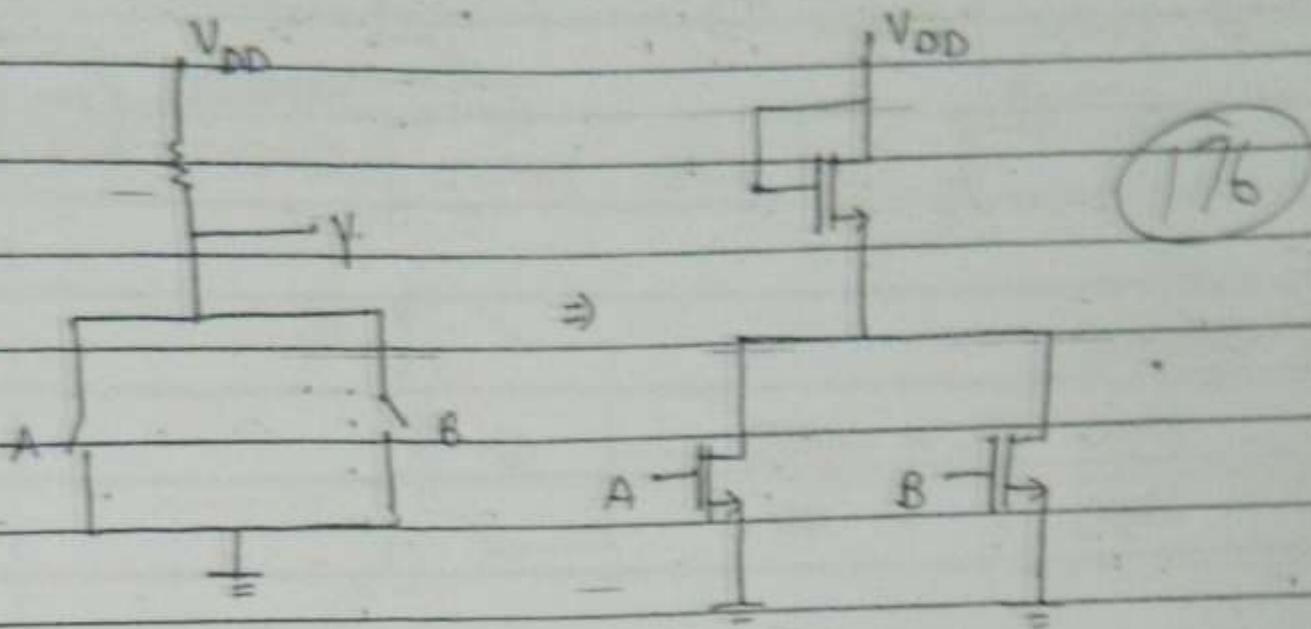


NMOS NAND Gate:-



A	B	T ₂	T ₃	Y
0	0	OFF	OFF	1
0	1	OFF	ON	1
1	0	ON	OFF	1
1	1	ON	ON	0

NMOS NOR Gate:-



$$t_{pd} = 250 \text{ nsec}$$

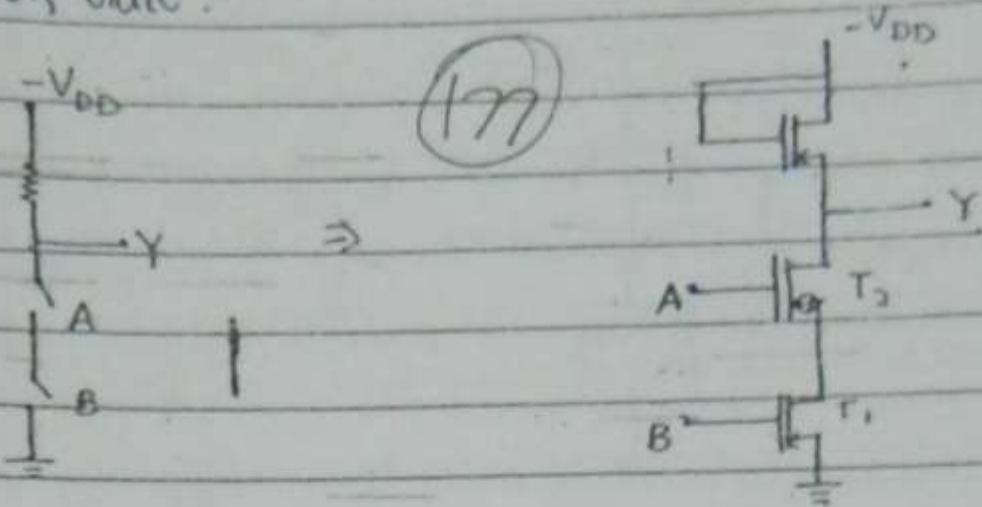
$$P_{diss} = 1 \text{ mW}$$

$$FOM = 250 \text{ PJ}$$

$$\text{fanout} = 5$$

$$NM = 1.5V$$

PMOS NOR Gate :-

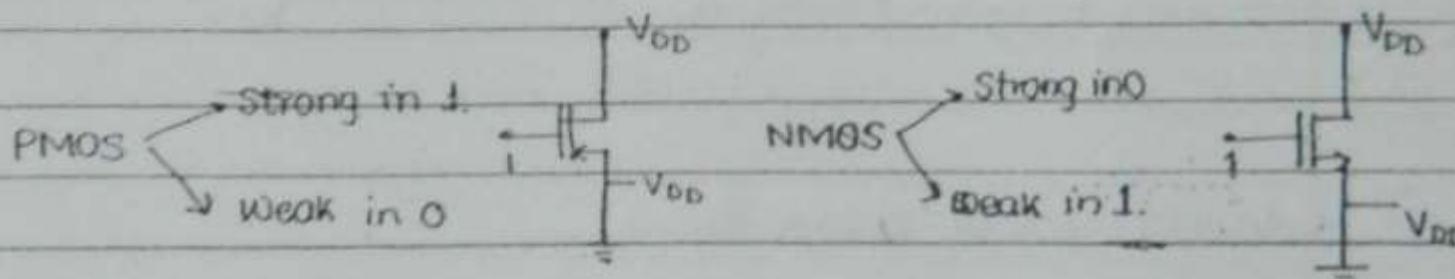


A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

$$t_{PD} = 300 \text{ nsec}$$

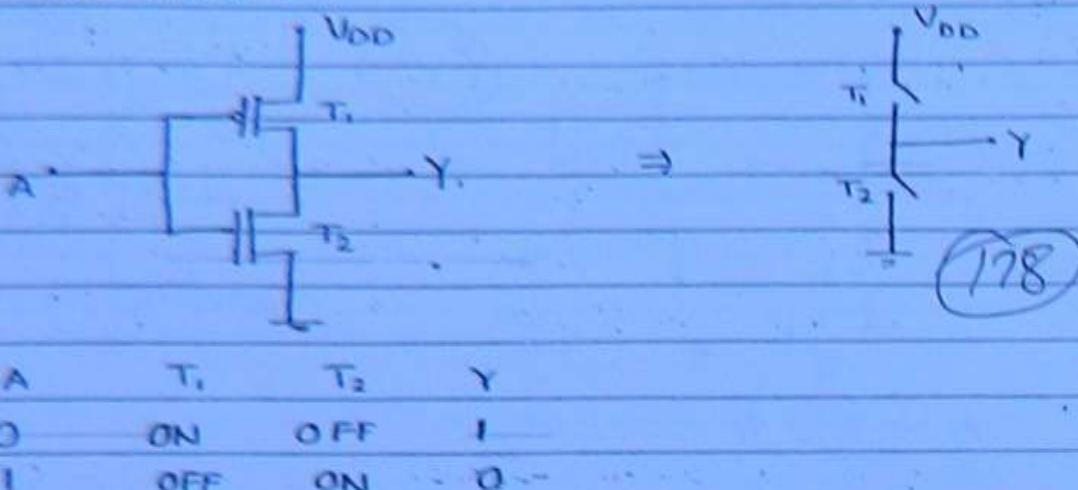
$$P_{diss} = 0.2 \text{ mW}$$

$$FOM = 60 \text{ PJ.}$$



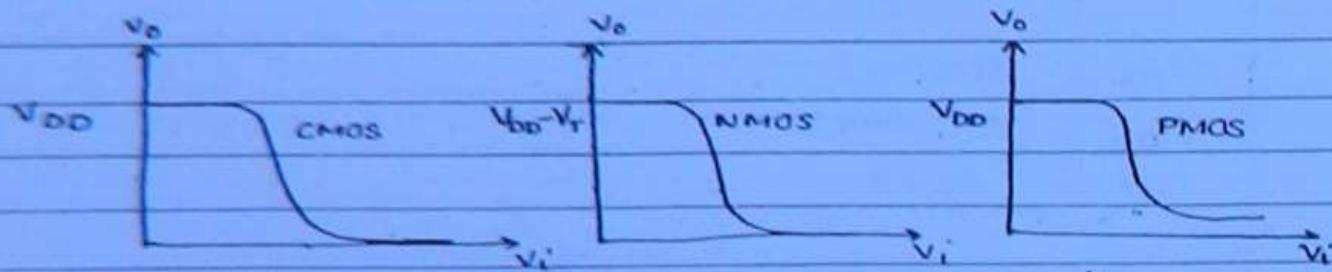
where, V_T = Threshold voltage.

CMOS NOT Gate :-



(178)

Transfer characteristics :-



⇒ lowest power dissipation.

$$P_{diss} = 0.01 \text{ mW}$$

$$t_{pd} = 70 \text{ nsec}$$

$$FOM = 0.7 \text{ pJ}$$

$$\text{Fanout} = 50$$

$$NM = \frac{V_{DD}}{2}$$

Power Dissipation :-

(i) Static PD =

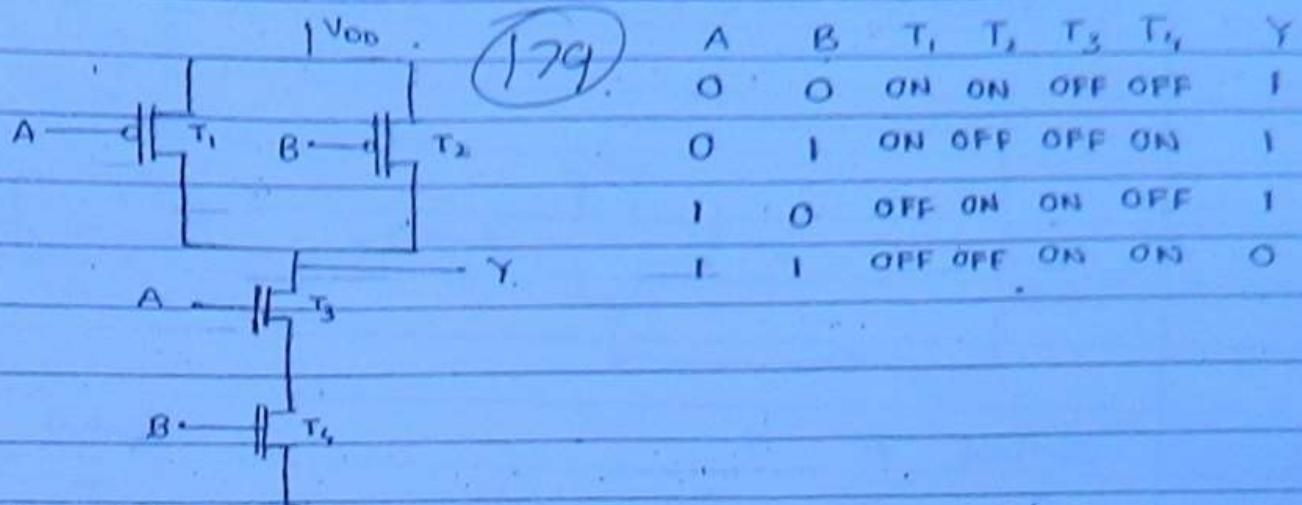
During logic '0' or logic '1'.

(ii) Dynamic PD =

During transition from 0 → 1 or 1 → 0.

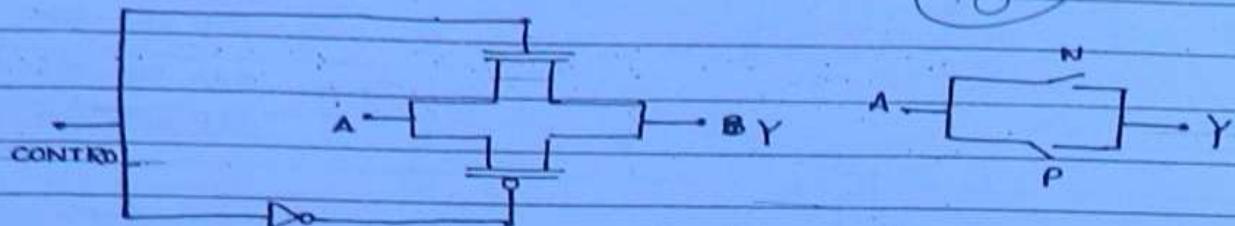
$$PD = C_f V_{DD}^2$$

CMOS NAND Gate:-



Transmission Gate :-

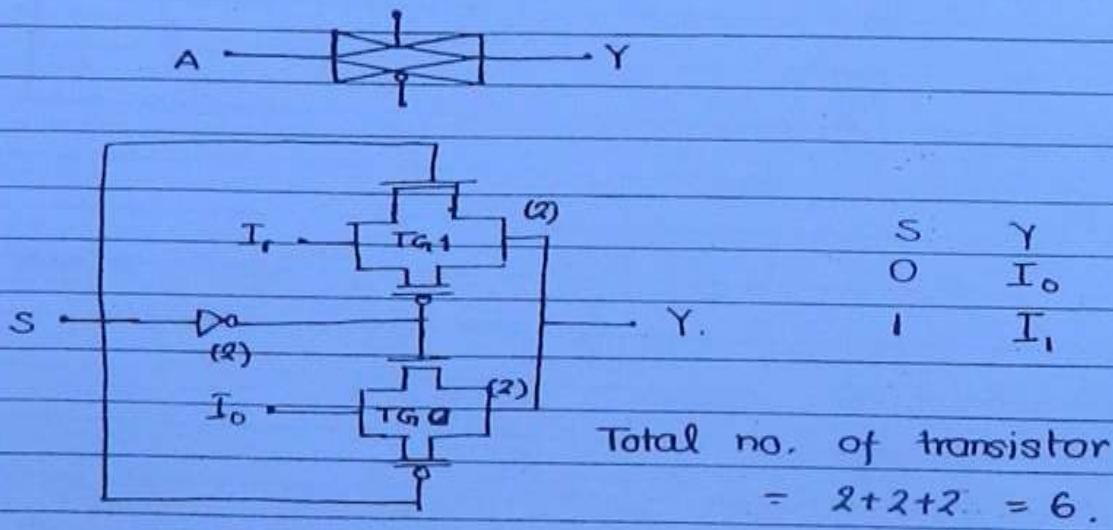
(180)



Control	A	Y
0	*	High impedance.
1	0	0
1	1	1

Control	Y
0	High imped.
1	A

Symbol of Transmission gate :-



CMOS monostable multivibrator :-

