

## Reference Books

- 1) Morris Mano
- 2) Bhagwati Kumar
- 3) R. P. Jain
- 4) Millman and Halkias

= BC  
✓

## • Digital Electronics

→ Boolean logical Ideas.

x	y	$f_0$	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$	$f_6$	$f_7$	$f_8$	$f_9$	$f_{10}$	$f_{11}$	$f_{12}$	$f_{13}$	$f_{14}$	$f_{15}$
0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	0	0	0	0	1	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

→ Boolean logical Ideas are categorised in three ways.

(i) Producing the constant 0, 1 [NULL, Identity] operation.

(ii) Unary operation: - Transfer, complement. [Buffer,  $\overline{NOT}$ ]

(iii) Binary operation: - AND, OR, NAND, NOR, EX-OR, EX-NOR, Inhibition, Implication.

•  $f_0 = 0 \Rightarrow$  NULL

•  $f_1 = x \cdot y \Rightarrow$  AND  $x \cap y$

•  $f_2 = x \cdot \bar{y} \Rightarrow$  Inhibition  $x/y$  [ $x$  but NOT  $y$ ]

•  $f_3 = x \Rightarrow$  Buffer

•  $f_4 = \bar{x} \cdot y \Rightarrow$  Inhibition  $y/x$  [ $y$  but NOT  $x$ ]

•  $f_5 = y \Rightarrow$  Buffer

•  $f_6 = x \oplus y \Rightarrow$  EX-OR  $= \bar{x}y + x\bar{y}$  "ODD Function"

$\uparrow$   $\downarrow$  also called stair case logic

	$S_2$	$S_1$	Buff? f
$f = 1$	0	0	0
$S_1 = 0$	0	1	1
	1	0	1
	1	1	0

$$f_2 = S_2 \oplus S_1$$

- $f_7 = x+y \Rightarrow \text{OR } x \vee y$
- $f_8 = \overline{x+y} \Rightarrow \text{NOR } x \veebar y$
- $f_9 = x \odot y \Rightarrow \text{EX-NOR } = x'y' + xy \text{ Even function}$

Co-incident logic gate, Equivalence logic gate

- $f_{10} = \bar{y} \Rightarrow \text{NOT}$
- $f_{11} = x+y' \Rightarrow \text{Implication } x \rightarrow y \text{ (if } y \text{ then } x)$
- $f_{12} = \bar{x} \Rightarrow \text{NOT}$
- $f_{13} = x'+y \Rightarrow \text{Implication } x \rightarrow y \text{ (if } x \text{ then } y)$
- $f_{14} = \overline{x \cdot y} \Rightarrow \text{NAND } \overline{x \cdot y}$
- $f_{15} = 1 \Rightarrow \text{Identity}$

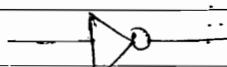
→ FOR  $n$  input variables :-  $2^n$  combination

$2^{2^n}$  possible function

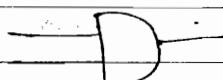
Symbols for the logic gates



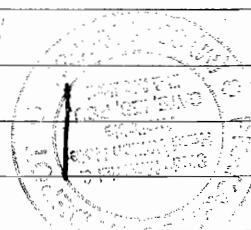
Buffer



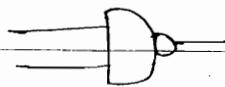
NOT



AND



OR



NAND



NOR



EX-OR



EX-NOR

→ NAND, NOR known as universal logical gates.

Shortcut -

NAND (4) NOR

NOT	1	1
AND	2	3
OR	3	2
EX-OR	4	5
EX-NOR	5	4

• Duality :-

operator  $B = \{\cdot, +\}$ , Digit  $B = \{0, 1\}$

Interchange :- (1) ( $\cdot, +$ )

(2) (0, 1)

Step 1 :- Interchange the operator ( $\cdot, +$ )

Interchange the Identity (0, 1)

AND

$$x \cdot x = x$$

$$x \cdot 0 = 0$$

$$x \cdot 1 = x$$

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

OR :

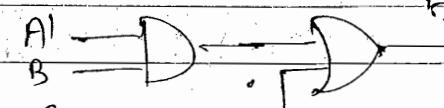
$$x + x = x$$

$$x + 1 = 1$$

$$x + 0 = x$$

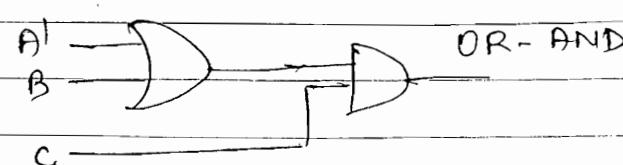
$$x + \bar{x} = 1$$

Ex:-  $f = (A \cdot B) + C$



AND-OR

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

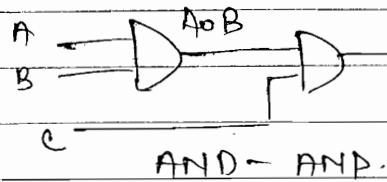


OR- AND

## • De-generative form: (two stage only)

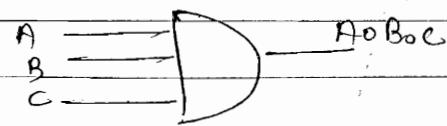
If a two-level logic gate sys. of p can be expressed with a single logic gate. Then the two level logic gate sys. is known as de-generative form for the single logic gate.

ex:- AND-AND is de-generated form. for the AND gate.



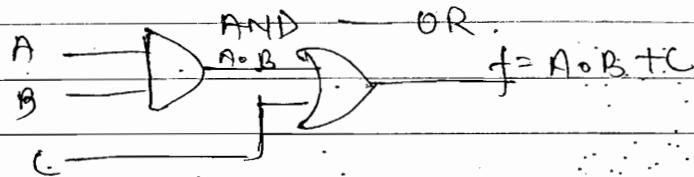
$$f = A \cdot B \cdot C$$

AND.



AND- AND.

## • Non-De-generative form:



AND  $\rightarrow$  OR.

$$f = A \cdot B + C$$

short cut

AND  $\rightarrow$  OR  $\xrightarrow{\text{dual}}$  OR  $\rightarrow$  AND

NAND  $\rightarrow$  NAND  $\xrightarrow{\text{dual}}$  NOR  $\rightarrow$  NOR

NOR  $\rightarrow$  OR  $\xrightarrow{\text{dual}}$  NAND  $\rightarrow$  AND

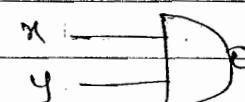
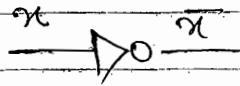
OR  $\rightarrow$  AND  $\xrightarrow{\text{dual}}$  NAND  $\rightarrow$  AND  $\rightarrow$  NOR



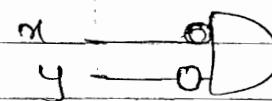
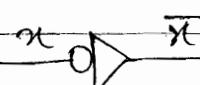
→ Combinations of the same horizontal line are dual forms

ex:- AND- OR  $\rightarrow$  OR- AND.

NOTE:- Inversion Before Binary operation is not same as that of Inversion after the Binary operation



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

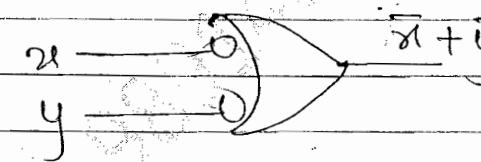
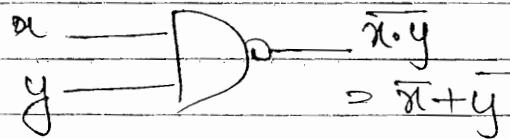


$$\bar{x} \cdot y$$

(Bubbled AND)

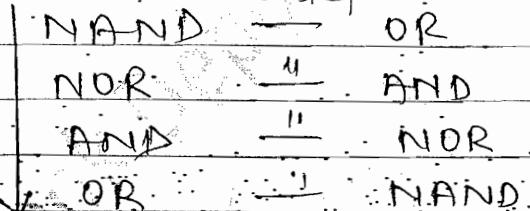
But it is valid for unary operation.

### • Alternative logic Gate:-



Shortcut

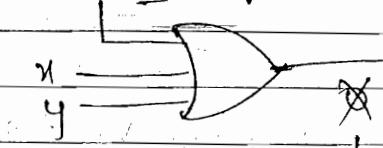
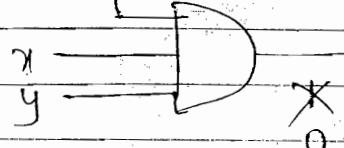
Bubbled



### • Inhibiting the logic Gates:-

$\ominus$  Inhibit

$\ominus$  Inhibit



Shortcut

Inhibit

Low → 0  
High → 1

Disable

Enable

NAND

Low

High

NOR

High

Low

AND

Low

High

OR

High

Low

- Positive & Negative logic: -

+ve logic

High  $\rightarrow 1$

Low  $\rightarrow 0$

-ve logic

High  $\rightarrow 1$

Low  $\rightarrow 0$

+ve.

ex:-  $\neg A \rightarrow 1$

$\neg A \rightarrow 0$

(+ve) AND logic

A	B	$f = A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

(-ve) AND = (+ve) OR

A	B	$f = A + B$
1	0	1
1	0	1
0	1	1
0	1	0

$\rightarrow$  -ve AND equal to +ve OR vice-versa

$\rightarrow$  same case for NAND and NOR.

$\rightarrow$  -ve NAND = +ve NOR vice-versa

- Complementing the Boolean Expression:-

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

Step - (1) Dual form

Step - (2) Complement individual variable

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

$$(2) = (A + B') \cdot C' \equiv f'$$

$\rightarrow$  ORDER TO SOLVE BOOLEAN EXPRESSION:-

(1) [ ] Brackets

(2)  $\neg$  NOT

(3) AND

(4) OR

• Ex-OR Gate Specialities:- Ex-NOR:-

$$\begin{array}{l}
 x \oplus x = 0 \\
 x \oplus \bar{x} = 1 \\
 x \oplus 1 = \bar{x} \\
 x \oplus 0 = x
 \end{array}$$

$$\begin{array}{l}
 x \odot x = 1 \\
 x \odot \bar{x} = 0 \\
 x \odot 1 = x \\
 x \odot 0 = \bar{x}
 \end{array}$$

~~Ex-NOR is even function for even No. of I/P variables.~~ [ ~~Ex-OR = ~~Ex-NOR~~~~ ]

$$\overline{x \oplus y} = x \odot y$$

$$x \oplus y \oplus z = x \odot y \odot z$$

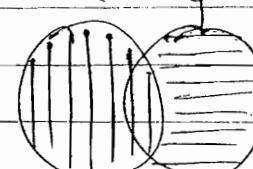
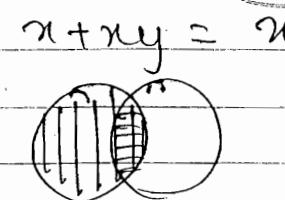
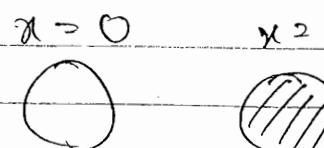
$$x \oplus y \oplus z \oplus w = x \odot y \odot w \odot z$$

Ex-NOR is odd function for odd No. of I/P variables  
[ ~~Ex-OR = ~~Ex-NOR~~~~ ]

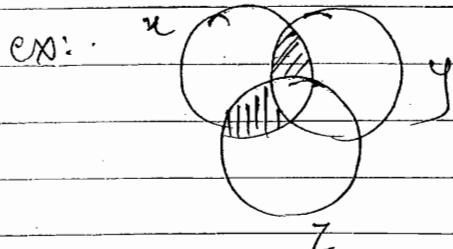
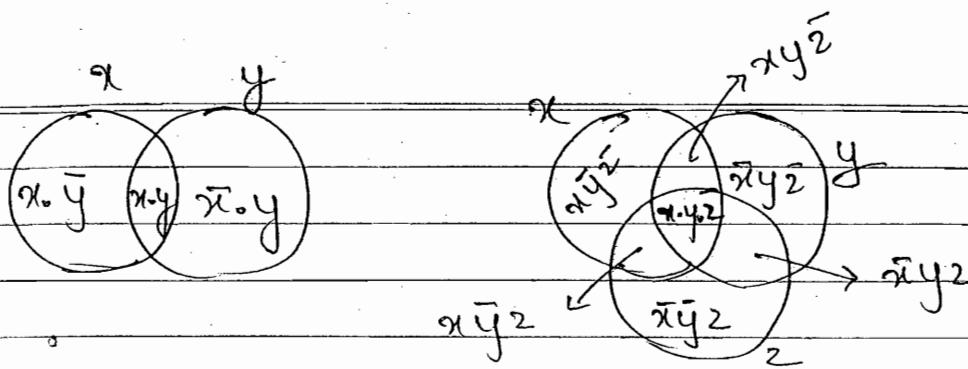
$$\begin{array}{l}
 x \oplus y \\
 \bar{x} \oplus y \\
 x \oplus \bar{y}
 \end{array}
 \left. \right\} = x \odot y$$

ex-  $\bar{x} \oplus y = (\bar{x})y + \bar{x}\bar{y}$   
 $= xy + \bar{x}\bar{y}$   
 $= x \odot y$

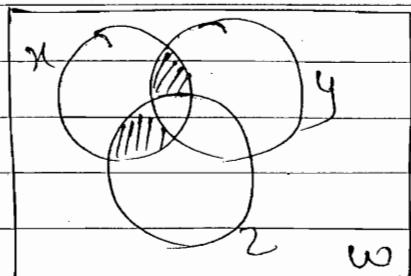
• Venn Diagram :-



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



$$f = xȳz + x̄ȳz$$



$$g = xȳz + x̄ȳz$$

$$f = w(xȳz + x̄ȳz)$$

19/9/2014

### Logic Minimization Technique

NOT if  $x=0$ ,  $\bar{x}=1$ ,  $(\bar{x})' = x$

AND  $x \cdot x = x$ ,  $x \cdot 0 = 0$

$x \cdot 1 = x$ ,  $x \cdot \bar{x} = 0$

OR  $x + x = x$

$x + \bar{x} = 1$

$x + 1 = 1$

$x + 0 = x$

Commutative law

$x \cdot y = y \cdot x$

$x + y = y + x$

Associative law

$x \cdot (y \cdot z) = (x \cdot y) \cdot z$

$x + (y + z) = (x + y) + z$

Distributive law  $x \cdot (y + z) = x \cdot y + x \cdot z$

$$x + (y \cdot z) = (x + y) \cdot (x + z)$$

De Morgan's law  $\overline{x \cdot y} = \overline{x} + \overline{y}$

$$\overline{x+y} = \overline{x} \cdot \overline{y}$$

Consensus Theorem

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

→ Redundant (any no. of variables)

(part of group nor effect the off of sys.).

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

$$\text{ex:- } A \cdot B + \overline{A} \cdot C + B \cdot C \cdot D \cdot E = A \cdot B + \overline{A} \cdot C$$

∴ A, C are complements variable, and can ↑ some other variable upto any no. (D, E ...).

→ A variable is associated with variable, its complement associated with other variable, Next term formed by the left over variables, the term is called redundant

→ Consensus Theorem can be extended to any no. of variables.

Transposition Theorem :-

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

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

Operators and association can be interchange.



Absorption law :- ORing a variable with ANDing of that variable with other variable results in the same variable

$$x + x \cdot y = x \quad , \quad x \cdot (y + x) = x$$

Redundant Literal Rule (RLR) :-

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

ORing a variable with ANDing of its complement with another variable results in the ORing of these two variables.

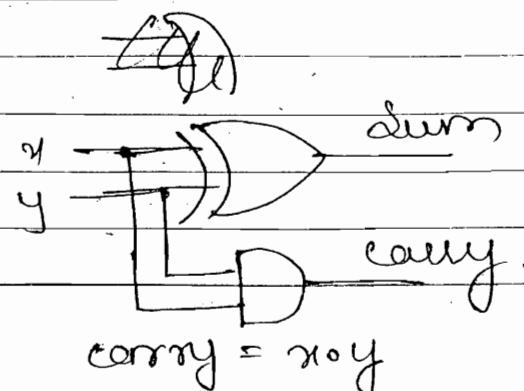
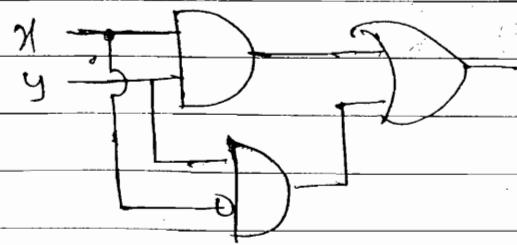
• SOP and POS form :-

SOP (Minterm) (concentrate in 1)

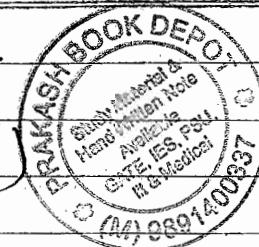
x	y	Minterm f	$x \cdot y$	Carry dum
0	0	$\bar{x} \cdot \bar{y}$	0	0
0	1	$\bar{x} \cdot y$	1	0
1	0	$x \cdot \bar{y}$	0	1
1	1	$x \cdot y$	1	0

$$f = x \cdot y + \bar{x} \cdot y$$

$$\text{Sum} = \bar{x} \cdot y + x \cdot \bar{y} \\ = x \oplus y$$

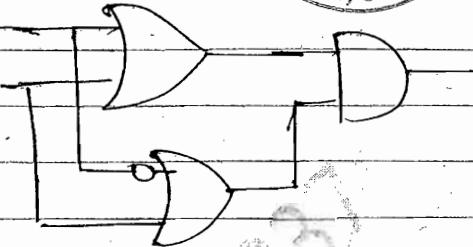


pos (Maxterm) Concentrate in zero.



$x \ y \quad \text{Maxterm. } f = (x+y) \cdot (\bar{x}+y)$

0	0	$x+y$	0
0	1	$x+\bar{y}$	1
1	0	$\bar{x}+y$	0
1	1	$\bar{x}+\bar{y}$	1



• If 0 are less in op table concentrate in 0 (pos) and make circuit, If 1 are less concentrate in 1 (sop) and make circuit.

•  $\sum m \Rightarrow SOP \quad \Pi M \Rightarrow POS$

ex:- (1)  $\sum m(0,2) = \Pi M(1,3)$  ~~to 0.91P~~

(2)  $\sum m(0,2) = \bar{x} \cdot \bar{y} + x \cdot \bar{y}$

(3)  $\Pi M(1,5) = \sum m(0,2,3,4,6,7)$

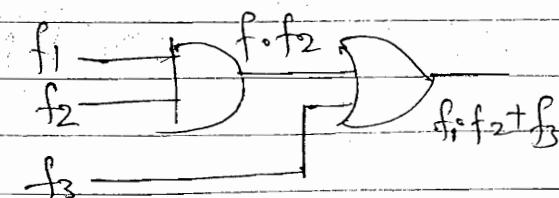
(4)  $\Pi M(0,2) = (x+y)(\bar{x}+y)$

(5)  $\sum m(1,5,9) = \Pi M(0,2,3,4,6,7,8,10,11,12,13,14,15)$

Q  $f_1 = \sum m(0,1,2,4,6)$

$f_2 = \sum m(2,3,4,6,7)$

$f_3 = \sum m(3,4,5,6)$



Solution  $f_1, f_2 \rightarrow$  Present in both

$\Rightarrow \sum m(2,4,6)$

$f_1, f_2 + f_3 \rightarrow \sum m(2,4,6) + \sum m(3,4,5,6)$

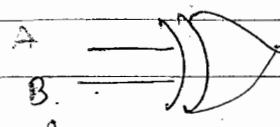
$= \sum m(2,3,4,5,6)$

(842)  $\rightarrow$  code in word

ex:-  $0101 = 5$   
 $0010 = 2$

binary NO.

Q.  $A = \Sigma m(0, 1, 2, 4, 6)$   $B = \Sigma m(1, 3, 5, 6, 7)$



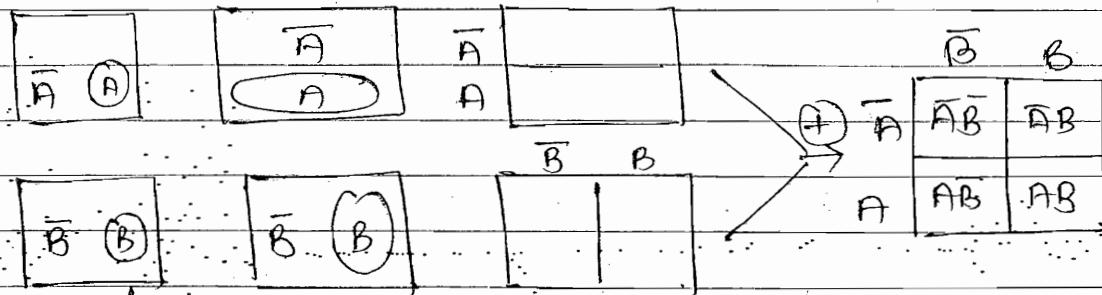
$$f = A \oplus B$$

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

$$\bar{A} \cdot B = \Sigma m(3, 5, 7), \quad \bar{B} \cdot A = \Sigma m(0, 2, 4)$$

$$f = \bar{A} \cdot B + A \cdot \bar{B} \\ = \Sigma m(0, 2, 3, 4, 5, 7)$$

• K-MAP :- It is a modification of Venn diagrams. It is a group of adjacent cells. Each cell is represented by minterm. Minterm is a group of literal.



		B	B
		$\bar{A}$	A
		0	1
$\bar{A}$	0	$\bar{A}\bar{B}$	$\bar{A}B$
	1	0	1
A	0	$A\bar{B}$	AB
	1	2	3

for 2 variables

Must be one literal

change in K-Map  
in adjacent cell).

		$\bar{B}\bar{C}$	$\bar{B}C$	$BC$	$B\bar{C}$	
		$\bar{A}$	$\bar{A}\bar{B}\bar{C}$	$\bar{A}\bar{B}C$	$\bar{A}BC$	$\bar{A}B\bar{C}$
		$\bar{B}$	0	1	3	2
$\bar{A}$	0	$\bar{A}\bar{B}\bar{C}$	1	$\bar{A}BC$	3	$\bar{A}B\bar{C}$
	1	4	$A\bar{B}\bar{C}$	$A\bar{B}C$	7	$ABC$
A	0	5	6	7	6	5
	1	2	1	3	0	1

for 3 variables

$$f = (A B C)$$

MSB LSB

MSB LSB

DO NOT write like  $\bar{C}A\bar{B}$  Because C is LSB  
always comes last

	$\bar{A}\bar{B}$	$\bar{A}B$	$A\bar{B}$	$AB$
0	$\bar{A}\bar{B}\bar{C}$ 0	$\bar{A}B\bar{C}$ 2	$A\bar{B}\bar{C}$ 6	$ABC$ 4
1	$\bar{A}\bar{B}C$ 3	$\bar{A}BC$ 7	$ABC$ 7	$\bar{A}\bar{B}C$ 5

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

$\overline{mP} \quad \overline{LSB} \quad \overline{MSB}$

Q  $f = \sum m(0, 1, 2, 6)$   $f = (ABC)$

$000 \downarrow \quad 001 \downarrow \quad 010 \downarrow \quad 110 \downarrow$

$f = \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + \bar{A}B\bar{C} + AB\bar{C}$  By default.

Solution KMAP

	$\bar{B}\bar{C}$	$\bar{B}C$	$BC$	$B\bar{C}$
case A	1 0	1	3	1 2
A B	4	5	7	16

$\overline{MSB} \quad \overline{LSB}$

case C

	$\bar{A}\bar{B}$	$\bar{A}B$	$A\bar{B}$	$AB$
case C	1	1	1	1
A B	11	3	7	5

$\overline{MSB} \quad \overline{LSB}$

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

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

Mistake (what we do??)

	$\bar{A}\bar{B}$	$\bar{A}B$	$A\bar{B}$	$AB$
case A	1 0	1	3	1 2
A B	4	5	7	16

$\overline{MSB} \quad \overline{LSB}$

$3-1=2$  pair

No. are written wrong

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

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

$f = (ABC)$

$\overline{MSB} \quad \overline{LSB}$

	$\bar{B}\bar{C}$	$\bar{B}C$	$BC$	$B\bar{C}$
case A	0	1 1	1 3	2
A B	4	1 5	1 7	6

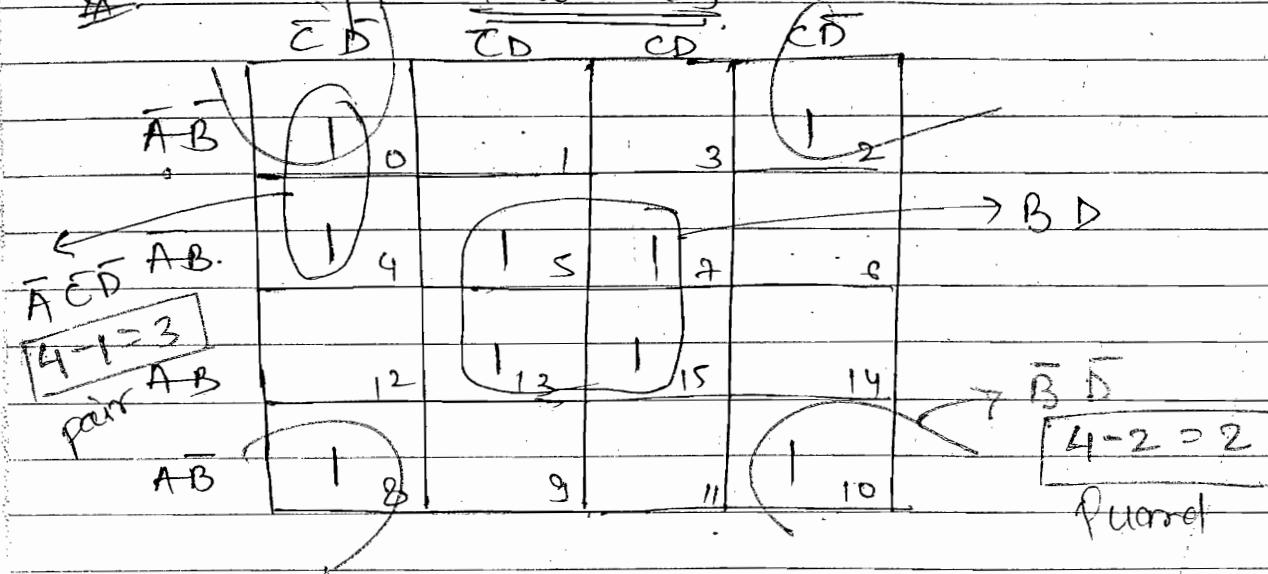
$\overline{MSB} \quad \overline{LSB}$

$f = C$

$3-2=1$

Quesd

Q. for 4 variables



Shortcut      Pair  $\rightarrow$  1      Subpart

Quad  $\rightarrow$  2      Subtract

Octet  $\rightarrow$  3      Subtract

• Variable - Shortcut  $\Rightarrow$  Opp in No. of literals

Ex:- No. of 1p variable = 5

1p pair = 3

Opp variable =  $5 - 3 = 2$

• Redundant Group:- It is that part of circuit which does not effect the circuit result.

→ While solving the K-Map we should follow the technique of higher order grouping to lower order grouping.

on

$$\textcircled{1} f = \sum m(1, 3, 6, 7)$$

	$\bar{B}\bar{C}$	$\bar{B}C$	$BC$	$B\bar{C}$
$\bar{A}$	0	1	1	2
$A$	4	5	1	6

$$f = \underbrace{\bar{A}C + AB}_{\text{E.P.I.}} + \underbrace{BC}_{\text{Redundant}} \xrightarrow{\text{R.P.I.}}$$

$$\textcircled{2} f = \sum m(1, 5, 6, 7, 11, 12, 13, 15)$$

	$\bar{C}\bar{D}$	$\bar{C}D$	$CD$	$C\bar{D}$
$\bar{A}\bar{B}$	0	1	2	3
$\bar{A}B$	4	1	1	6
$A\bar{B}$	1	1	1	5
$AB$	8	9	11	10

$$f = \underbrace{\bar{A}\bar{C}D + AB\bar{C} + \bar{A}Bc}_{\text{R.P.I.}} + \underbrace{ACB}_{\text{Redundant.}} + BD$$

leave "quadrant".

• If  $\textcircled{13} \rightarrow 1$  is not present then we have to consider quadrant. (It will give less # literal)

• Must consider all the 1's in group.

20/9/2014

• Implicant :- It is min-term corresponding to that cell which is having 1 in the K-map.

• Prime Implicant :- All possible grouping of K-map known as prime implicant.

• Non Prime Implicant :- This all those implicant which are not able to get possible grouping.

• Essential Prime Implicant :- R.P.I. is prime implicant (P.I.) only which is having at least a single 1 which is only one time group. (Uniquely grouped)

• Selective Prime Implicant :- These are prime implicant which are under the process of selection.

all G.P.S are P.I. But reverse is not ~~true~~ true

- Redundant P.I.: It is the P.I. which does not affect the circuit result.

Ex: -  $\begin{array}{c} \bar{B}\bar{C} \quad \bar{B}C \quad BC \quad B\bar{C} \\ \hline \bar{A} \quad (1, 1, 1, 1) \quad 3 \quad 2 \\ \hline A \quad 4 \quad 5 \quad (1, 1, 1) \end{array}$

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

E.P.I.      E.P.J.  
also P.I.      also P.I.

Ex: -  $\begin{array}{c} \bar{B}\bar{C} \quad \bar{B}C \quad BC \quad B\bar{C} \\ \hline \bar{A} \quad (1, 1, 1, 1) \quad 2 \quad (1, 1) \\ \hline A \quad 4 \quad 5 \quad (1, 1, 1) \end{array}$

$$f = \Sigma m(0, 1, 4, 6, 7)$$

$$f = \bar{A}\bar{B} (0, 1) \quad E.P.I. \checkmark$$

$$\bar{A}\bar{C} (0, 2) \quad P.I. \quad \checkmark \quad \text{S.P.I.}$$

$$B\bar{C} (2, 6) \quad P.I. \quad \checkmark \quad \text{selectively prime}$$

$$AB (6, 7) \quad E.P.I. \checkmark \quad \text{implicant.}$$

J.M.P  $\rightarrow$  5  
(implicant), P.I.  $\rightarrow$  4, E.P.J  $\rightarrow$  2  
, Non P.I.  $\rightarrow$  0,

Ex: -  $\begin{array}{c} (1, 1, 1, 2) \\ \hline 4 \quad 5 \quad (1, 1, 1) \end{array}$

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

Implicant = 4, Non Prime Implicant  $\Rightarrow$  1  
Prime Implicant = 2      E.P.I. = 2

→ DO NOT CARE Conditions: - The op corresponding to the unspecified J.M.P is known as don't care condition.

Q Design a clk which gives alarm at 5 and 7 o'clock.

Solve four 3/1 P AB CD

~~0000 0 → X~~

$g \rightarrow 0$

$\bar{c}\bar{d}$   $\bar{c}d$   $c\bar{d}$   $cd$

Never  
used  
3/1 P

$1 \rightarrow 0$

$10 \rightarrow 0$

$\bar{a}\bar{b}$

$X_0$   $0_1$   $0_3$   $0_2$

$2 \rightarrow 0$

$11 \rightarrow 0$

$0_4$   $1_5$   $1_7$   $b_6$

$3 \rightarrow 0$

$12 \rightarrow 0$

$\bar{a}b$

$0_{12}$   $X_{13}$   $X_{14}$   $X_{14}$

$4 \rightarrow 0$

$11 \times 13 \rightarrow X$

$0_8$   $0_9$   $0_{11}$   $0_{10}$

$5 \rightarrow 0_1$

$110, 14 \rightarrow X$

$ab$

$6 \rightarrow 0$

$111, 15 \rightarrow X$

$a\bar{b}$

$7 \rightarrow 1$

not uses  
in 3/1 P

$8 \rightarrow 0$

$\bar{a}\bar{b}\bar{d} \rightarrow b\bar{d}$

$X \rightarrow$  blocks are empty so we have a  
chance to take don't care.

$$\text{Ex} - f = \sum m(1, 2) + \phi(3, 5, 7)$$

	$\bar{b}\bar{c}$	$\bar{b}c$	$bc$	$b\bar{c}$
$\bar{a}$	0	1	$\times$	1
$a$	4	$\times$	$\times$	6

$$f = \bar{a}b + c$$

• POS forms in K-Maps

(conversion of SOP into POS)

$$f = \sum m(0, 1, 6, 7)$$

	$\bar{b}\bar{c}$	$\bar{b}c$	$bc$	$b\bar{c}$
$\bar{a}$	1	1	3	2
$a$	0	5	1	6

$$f = \prod M(2, 3, 4, 5)$$

	$\bar{b}\bar{c}$	$\bar{b}c$	$bc$	$b\bar{c}$
$\bar{a}$	0	1	0	2
$a$	0	0	1	0

$$f = \bar{a}\bar{b} + ab$$

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

$$\text{POS} = \text{SOP}$$

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

(2) Compliment variable  $\Rightarrow$

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

Not matching with eq<sup>n</sup> - ① But drawing in K-map Identity must be also change.

① change operator

② complement individual variable.

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

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

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

→ Blue colour 0 convert into red colour 1 and remaining are 0 (2,3,4,5), pair the 0's bcoz pos form.

• Variable Entering Method 3. - (VEM)

4 to 1 P  $\rightarrow$  Using 3 variable K-map.

$$f = \bar{A}\bar{B}C\bar{D} + \bar{A}BCD + \bar{A}B\bar{C}\bar{D} + \bar{A}B\bar{C}D + A\bar{B}CD$$

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

$$f = \bar{A}C\bar{D} + \bar{A}BD + BCD + ABC$$

$$f = \bar{A}\bar{B}C\bar{D} + \bar{A}BCD + \bar{A}B\bar{C}\bar{D} + \bar{A}B\bar{C}D + A\bar{B}CD + A\bar{B}\bar{C}D$$

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

$$f = \bar{A}C\bar{D} + \bar{A}BD + BCD + A\bar{B}\bar{C}D$$

B C

②

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

$$f = BC \quad \text{Ans}$$

~~Conventional~~ Variable Entering Method :-

$\bar{y}\bar{z}$	$\bar{y}z$	$yz$	$y\bar{z}$	
$\bar{x}$	0 <sub>0</sub>	A	1 <sub>3</sub>	$\bar{A}$ <sub>2</sub>
x	B <sub>1</sub>	0 <sub>5</sub>	X <sub>7</sub>	C <sub>6</sub>

Step 1 :- Keep all the variables as zero, and get the simplified ~~and~~ expression.

NOTE :- Check whether the given 1 in K-map is fully covered or not.

~~Step 1~~ - If it fully covered then it becomes redundant.

Step 2 :- Select any variable and keep 1 in its position and other variables should be kept as 0. and given ~~0~~ + replaced by don't care (X).

Step 3 :- Repeat the above step by selecting each variable.

	$\bar{y}\bar{z}$	$\bar{y}z$	$yz$	$y\bar{z}$
<del>Step 1</del>	$\bar{x}$	0 0	0 1	0
x	0 0	0 0	X 1	0

Red colour  $\rightarrow$  change  
Black colour  $\rightarrow$  same

$$f_1 = yz \quad \text{Redundant}$$

$\bar{y}\bar{z}$   $\bar{y}z$   $yz$   $y\bar{z}$

<del>for A</del>	$\bar{x}$	0	1	X	0	
x	0	0	X	0		

$$f_2 = A \bar{x} z$$

$\bar{y}\bar{z}$   $\bar{y}z$   $yz$   $y\bar{z}$

<del>for <math>\bar{A}</math></del>	$\bar{x}$	0	0	X	1	
x	0	0	X	0		

$$f_3 = \bar{A} \bar{x} y$$

	$\bar{y}\bar{z}$	$\bar{y}z$	$y\bar{z}$	$yz$
for B: $\bar{x}$	0	0	x	0
$\bar{x}$	0	0	x	0

$$f_4 = B \bar{x} \bar{y} \bar{z}$$

	$\bar{y}\bar{z}$	$\bar{y}z$	$y\bar{z}$	$yz$
$\bar{y}$	0	0	x	0
$x$	0	0	(x)	0

$$f_5 = \bar{c}xy$$

$$f = (\bar{y}\bar{z}) + A\bar{x}\bar{z} + \bar{A}\bar{x}y + Bx\bar{y}\bar{z} + \bar{c}xy$$

Redundant

If  $A \rightarrow$  cell is not present then consider the redundant in and. It will not be redundant anymore.

cell is fully covered.

obtained

	$\bar{y}\bar{z}$	$\bar{y}z$	$y\bar{z}$	$yz$
$\bar{x}$	0	(A)	$A + \bar{A}y$	$\bar{A}$
$x$	(B)	0	(x)	C

$$f = A\bar{x}\bar{z} + \bar{c}xy + \bar{A}\bar{x}y + Bx\bar{y}\bar{z}$$

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

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

Method

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

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

Ans.

gen Method

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

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

$\bar{A}B$

Not fully covered.

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

<u><math>A, A_1</math> with X</u>	can be grouped together (group)	make group	$\boxed{011110}$
	don't care		

$\bar{B} \quad B \quad \bar{C} + C$  for C

3<sup>rd</sup> Method

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

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

- Cell having 1 must have neighbour cell of variable and its complement for fully covered.  $\bar{y}_1 \bar{y}_2 y_2 \bar{y}_1$

$\bar{x}$	$\boxed{A}$	$\boxed{1}$	$\boxed{\bar{A}}$
$x$			

$\bar{y}_2 \bar{y}_2 y_2 \bar{y}_2$

$\bar{x}$	$\boxed{A}$	$\boxed{1}$	$\boxed{\bar{B}}$
$x$			

Then not fully covered.

- pairing of variable with don't care and 1 only.

$A \cdot A$

$A \cdot B$  X

$A \cdot \bar{X}$

$A \cdot 1$

- All 1's in K-map must check → fully covered or not.

$\bar{y}_2 \rightarrow$  this is fully covered.

$\bar{x}$	$\boxed{A}$	$\boxed{1}$	$\boxed{A}$
$x$	$\boxed{1}$		

→ not fully covered,  $\rightarrow x \bar{y}_2$

- Tabulation Method of Mcclusky:

Step 1: Get the circuit expression in the form of SOP.

Step 2: Make the different groups depending on the number of 1's in the minterms.

Step 3: Compare the successive groups <sup>until we get the</sup> and get the simplified answer. <sup>and</sup> <sub>different group</sub>

→ In don't care Condition :- Consider them. but remove at table.

→ Solving the don't care terms in Tabulation Method  
Consider the don't care no. along with given no. and get the final minterms but don't this no. in E.P.I. Table.

$$f = \sum m(0, 1, 3, 7, 8, 9, 11, 14, 15)$$

Group	Minterm	Variable	Implicant Table
		A B C D	
0	0	0 0 0 0	
1	1	0 0 0 1	
	8	1 0 0 0	
2	3	0 0 1 1	
	9	1 0 0 1	
3	7	0 1 1 1	
	11	1 0 1 1	
	14	1 1 1 0	
4	15	1 1 1 1	

Group	Minterms	Variable	pair Table
		A B C D	
0	0, 1	0 0 0 -	
	0, 8	- 0 0 0	
1	1, 3	0 0 - 1	
	1, 9	- 0 0 1	
	8, 9	1 0 0 -	
2	3, 7	0 - 1 1	
	3, 11	- 0 1 1	
	9, 11	1 0 - 1	

3	7, 15	- 1 1 1
	11, 15	1 - 1 1
	14, 15	1 1 1 - ABC

Group.	Minterm	Variable	
0	0, 1, 8, 9	- 0 0 -	Quads
	0, 8, 1, 9	- 0 0 -	BC Table
1	3, 3, 9, 11	- 0 - 1	
	1 9, 3, 11	- 0 - 0	BD
2	3, 7, 11, 15	- - 1 1	
	3, 11, 7, 15	- - 1 1	CD

$$f = \bar{B}\bar{C} + (\bar{B}D) + CD + ABE$$

→ Redundent

	0	1	3	7	8	9	11	14	15
single in column.									
E.P.I $\bar{B}\bar{C}$	(X)	X				(X)	X		
R.P.I $\bar{B}D$			X	X		X	X		

E.P.I $CD$			X	(X)					
R.P.I $ABC$							X		
								(X)	
									X

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

→ Row having cross become R.P.I

→ Suppose column II

don't have two ones

only  $\bar{B}D$  row having cross them  $\bar{B}D$

Become P.I, not

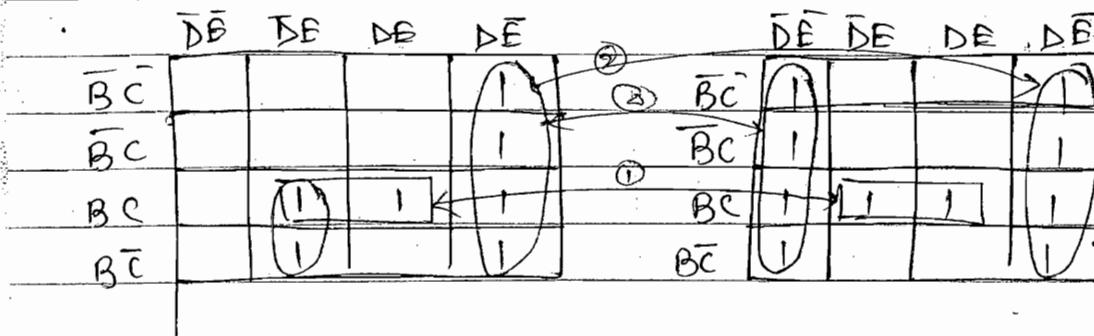
Redundent  
( $\bar{B}D$ )

→ Circle only those X which are single in column

R.P.I.

## 5-Variable K-Map

- For grouping only one literal change required.



$$\begin{aligned}
 1. B'C'AB'C + A'BC' &= B'C'E \quad (\text{grouped together}) \\
 2. A'DE + A'DE &= A'DE \quad (\text{"}) \\
 3. A'DE + A'DE &\Rightarrow \text{Two literal change, NO grouped together} \\
 4. B'C'E + A'B'DE &\Rightarrow \text{grouped together} \\
 &\quad (\text{quad made } \rightarrow \text{two})
 \end{aligned}$$

- standard SOP (ssop):- Must having all literals.
- If any literal is missing it called SOP.

SOP to SSOP:-

$$\rightarrow f = \overline{A}BC + AB + B\bar{C} \quad (\text{SOP})$$

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

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

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

$\bar{A}$	$\bar{B}C$	$\bar{B}C$	$BC$	$B\bar{C}$
$A$	0	1	1 2	1 2
	4	5	7	6
	8	9	11	10

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

Ex:-

	$\bar{C}\bar{D}$	$\bar{C}D$	$CD$	$C\bar{D}$
$\bar{A}\bar{B}$	1 0	1 3	1 2	
$\bar{A}B$	1 4	1 5	1 7	1 6
$AB$	1 12	1 11	1 15	1 14
$A\bar{B}$	1 8	1 9	1 11	1 10

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

## Diagonal Group (EX-NOR, EX-OR)

- Not the circuit simplified Technique → Only to write the answer quickly. (Save time)

- Same Nature → EX-NOR  $(\bar{A}\bar{B})$  {Both bar, Both without bar}
  - Different Nature → EX-OR  $(\bar{A}, B)$  {one bar, other without bar}
- (making grouping in diagonal).

ex:-

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

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

→ direct →  $A \oplus B$  (EX-NOR)  
(same Nature)

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

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

→  $A \oplus B$  (EX-OR)  
(different Nature)

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

→ B constant, A & C different  
nature

B constant,  $(\bar{A}, \bar{C}), (A, C)$  same nature

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

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

→  $B, \bar{C}$  constant,  $(\bar{A}D), (A\bar{D})$  different  
nature

$$B\bar{C} (A \oplus D)$$

→  $\bar{B}\bar{D}$  constant  $(\bar{A}\bar{B}, A, S)$  same NOR  
 $\bar{B}\bar{D} [A \oplus C]$

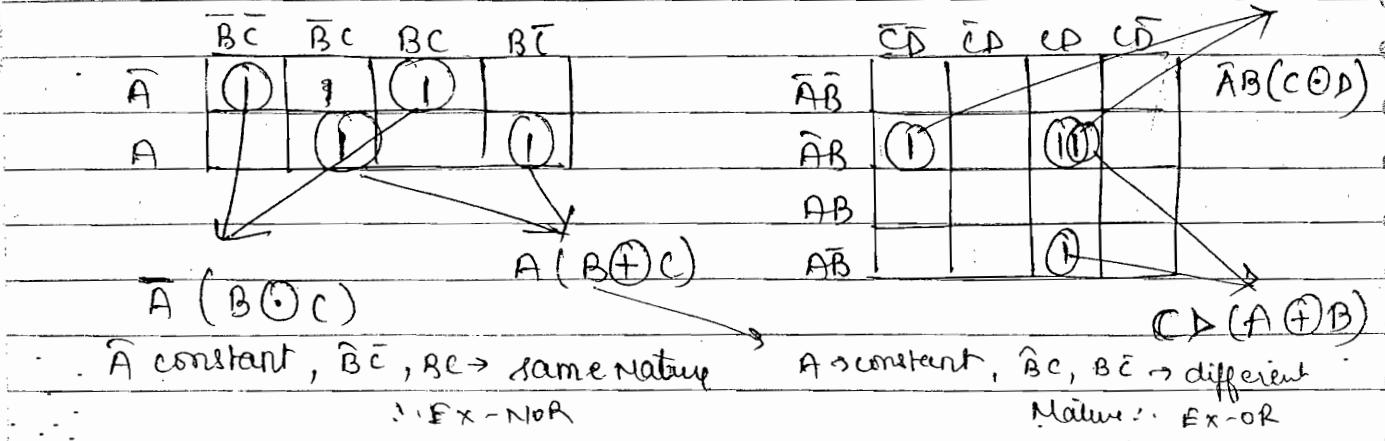
Only apply when given in option :-

- Only apply when simplified is not possible. [only present in diagonal].

$$\Rightarrow \text{If } \begin{array}{|c|c|c|c|c|} \hline \bar{B}\bar{C} & \bar{B}C & BC & B\bar{C} \\ \hline \bar{A} & 1 & 1 & 1 \\ \hline A & 1 & 1 & 1 \\ \hline \end{array} \text{ taking this pairing } \\ AC + \bar{A}\bar{B}C$$

- Other Method for simplifying:-

OFFSET IN K-MAP:-



$\bar{A}$  constant,  $\bar{B}\bar{C}$ ,  $BC \rightarrow$  same nature

$\therefore$  EX-NOR

$A$  constant,  $\bar{B}C$ ,  $B\bar{C} \rightarrow$  different

Nature: EX-OR

### Sequential Circuit:-

present O/p depend upon not only present S/p but also past O/p.

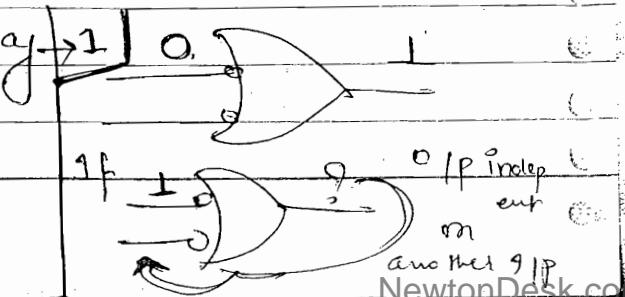
latch = O/p store the value (0 or 1) until the S/p change  
 $\therefore$  called latch.

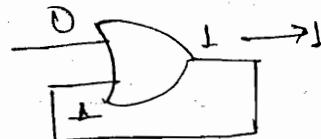
Active low- Input ( $S=0$ ), affect the O/p.

$\rightarrow$  If S/p is 0  $\rightarrow$  O/p is away  $\rightarrow$  1 O.

$S$	$R$
0 0	$\rightarrow$ don't care.

If  $S \rightarrow 0$   $O \rightarrow 1$



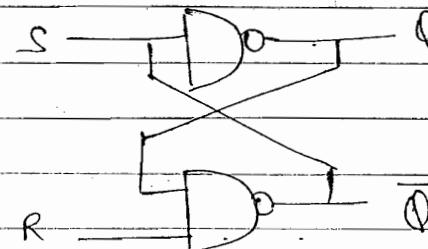


NAND  $\rightarrow$  low  
NOR  $\rightarrow$  high

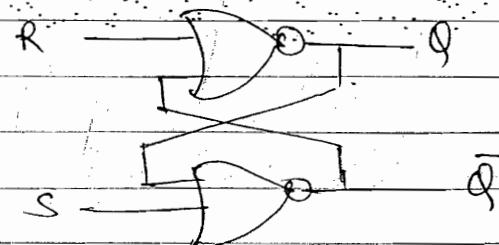
- $R \rightarrow 0, \bar{Q} \rightarrow 1, Q \rightarrow 0$  So Reset,  $S \rightarrow 1$
- $R \rightarrow 1, \bar{Q} \rightarrow 0, Q \rightarrow 1$  So Set,  $S \rightarrow 0$
- Apply op again-again to get stable op (microscopic method).
- $S-R \rightarrow$  NAND gate  $\rightarrow$  Active low  
 $\rightarrow$  NOR Gate  $\rightarrow$  Active high

- Logic gate with feedback connection known as sequential circuits. ex- latches.

Latches are two type: ① NAND gate latch (active-low)

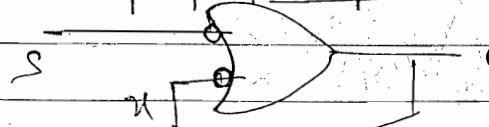


② NOR gate latch: (active-high)

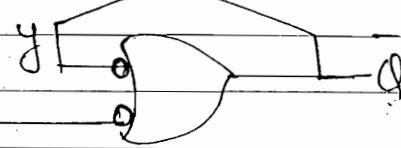


$\left. \begin{array}{l} \text{S, R, position} \\ \text{change to} \\ \text{get complete} \\ \text{diff. table} \\ \text{from S-R(NAND) above,} \end{array} \right\}$

• S-R Flip-flop operation (Active-low)



Case 1:  $S=1, R=1$



Initially if  $Q=0$   
when the feedback SFF will be  $x=1, y=0$ , Then we get the next stage  $Q^+=1$

$$Q^+ = 0.$$

✓ Initially if  $Q = 1$  :- Then the feedback  $S_{fp}$  will be  $x = 0, y = 1$ . Then the next stage will be  $Q^+ = 1$

So By observation we can say that when  $S = 1, R = 1$  we get  $Q^+ = \text{No change}$ .

Case 2 :-  $S = 1, R = 0$

✓ Initially  $Q = 0$  :- Then the feedback  $S_{fp}$  will be  $x = 1, y = 0$  so we get the next stage  $Q^+ = 0$

✓ Initially  $Q = 1$  :- Then the feedback  $S_{fp}$  will be  $x = 0, y = 1$  so we get the next stage  $Q = \bar{Q} = 1$  only. which is an intermediate stage. It is not the stable stage. So that the internal feedback will operate until we get stable Q/p.

By the observation we can say that when  $S = 1, R = 0$  we get  $Q^+ = 0$  "Reset"

case 3 :-  $S = 0, R = 1$

The same above process is conducted for initially  $Q = 0$  and for initially  $Q = 1$  By.

By observation we can say when  $S = 0, R = 1$  we get  $Q^+ = 1$  "Set"

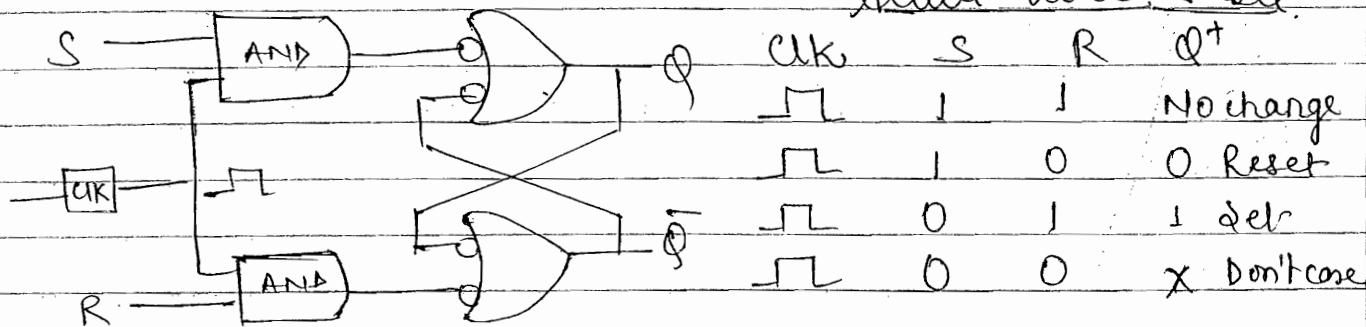
case 4 :- When  $S = 0, R = 0$

By the observation we got  $Q^+ = \bar{Q}^+ = 1$  which is an useless condition known as don't care condition.

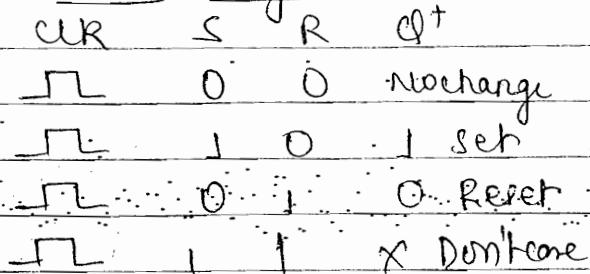
NOTE: (1) The S-R latch can be controlled with the help of a switch known as gated SR latch

(2) The controlling can also be done by using clock generator known as clocked SR flip-flop.

Active low Table



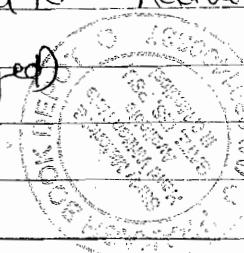
Active high Table



• Difference b/w flip-flop and latch is clock generator.  
no CLK  $\rightarrow$  latch. Controlling done  $\rightarrow$  flip-flop.

• S/p gives, changed internally. By NOT gate and S/p comes  $\rightarrow$  Active high (Mean. flow changes into high). S/p. in Active low  $\rightarrow$  equal to Active high (opposite internally changed).

$\rightarrow$  Modified SR  $\rightarrow$  JK



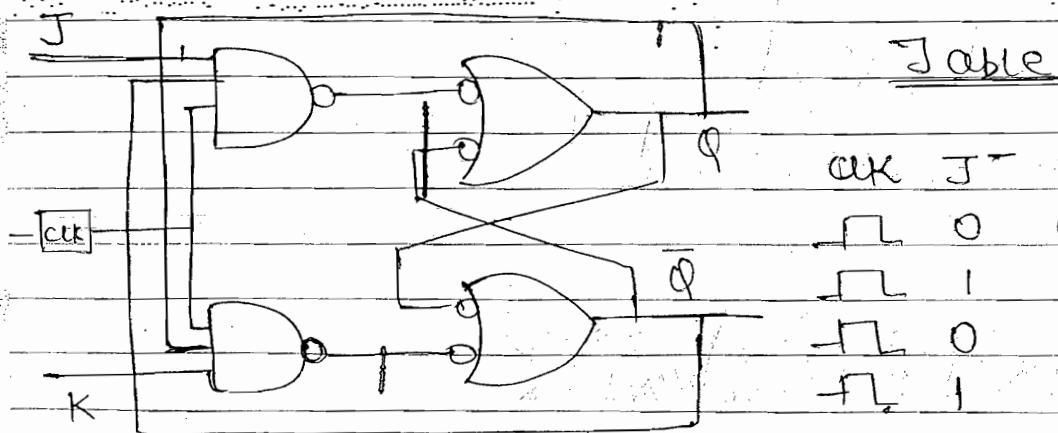
• J-K Flip-Flop :- The modification of SR flip-flop with external feedback connection is known as J-K flip-flop.

When  $J=1$ ,  $K=1$ , and clock pulse is applied. By observation we can say that  $Q^+ = \bar{Q}$  toggle condition.

NOTE : ① Internal feedback should be operated until we get the stable O/p. This operation doesn't depend on clock pulse.

② External feedback operated only one time for the clock pulse.

If output  $[Q, \bar{Q}]$  <sup>applied to</sup> feedback to S/p, it will not effect the S/p because clock is now OFF, it will wait until the next clock pulse will applied]



$S \quad R \rightarrow 99\% \rightarrow$  High Active High  
(By default)

	J	K	
active high	0	0	N.C
	0	1	Set
	1	0	Reset
	1	1	

1 1  $\rightarrow$  X 1 1 Toggle.

feedback

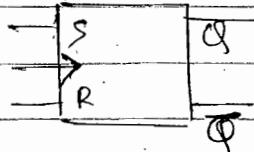
$\rightarrow$  To make J = K

R  $\bar{Q}$

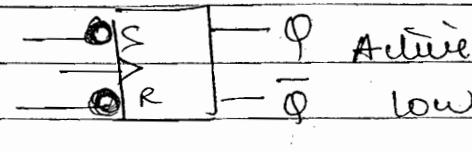
[R  $\bar{Q}$ ]

J = K

[S  $\bar{Q}$ ]

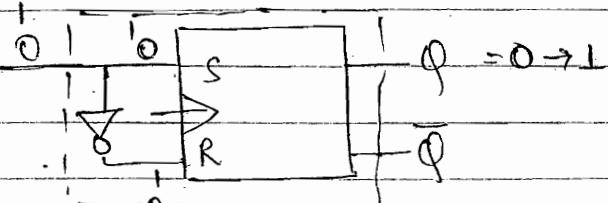


Active high



Active low

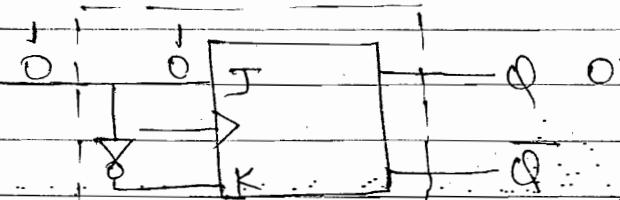
- D-Flip Flop and T-Flip Flop:-



Table

clk	D	Q <sup>+</sup>
0	0	0
1	1	1

$\rightarrow$  By J-K flip-flop  $\rightarrow$  not use this one, Because of external feedback connect, size  $\uparrow$ , But asked in exam



- J - flip-flop (Toggle)

T = high always (constant)

Table

clk	J	Q <sup>+</sup>
0	0	0
1	1	1

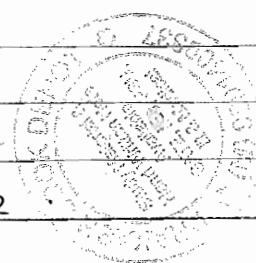
NO change  
Toggle.

$t_{c+tc} = 2tc$

$t_c$   $t_c$

1/p  $\propto$  Hz.

1/p  $\propto$  Hz



→ If  $Q_p$  is always set at high, for 1 clock pulse it will toggle  $0 \rightarrow 1$  and for 2nd clock pulse it will toggle  $1 \rightarrow 0$

→ 2  $Q_p$  pulse, 1  $Q_p$  pulse with  $\frac{1}{2}$  freq.

22/9/2014

### • Characteristic Equations of Flip-Flop:-

$Q \quad J \quad K \quad Q^+$

0 0 0 0 N.C

0 0 1 0 Reset

0 1 0 1 Set

0 1 1 1 Toggle  $\bar{Q}$

1 0 0 1 N.C

1 0 1 0 Reset

1 1 0 1 Set

1 1 1 0 Toggle  $\bar{Q}$

$\bar{J} \bar{K} \quad \bar{J} \bar{K} \quad J \bar{K} \quad J \bar{K}$

0	1	1	1
1	0	0	0

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

### • Characteristic Equations of S-R Flip-Flop:-

$Q \quad S \quad R \quad Q^+$

0 0 0 0 N.C

0 0 1 0 Reset

0 1 0 1 Set

0 1 1 X Don't care

1 0 0 1 N.C.

1 0 1 0 Reset

1 1 0 1 Set

1 1 1 X Don't care

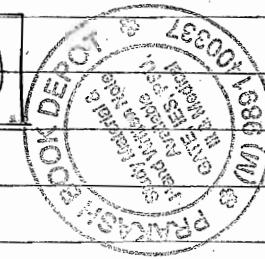
$\bar{S} \bar{R} \quad \bar{S} \bar{R} \quad S \bar{R} \quad S \bar{R}$

0	1	X	1
1	0	X	0

$$Q^+ = S + \bar{R}Q$$

• Characteristic eqn of D-Flip Flop: -

Q	D	Q <sup>+</sup>	D	Q <sup>+</sup>
0	0	0	0	0
0	1	1	0	1
1	0	0	1	1
1	1	1	1	1



• Characteristic eqn of T-Flip Flop

Q	T	Q <sup>+</sup>	T	T
0	0	0	N.C.	0
0	1	1	Toggle Q	1
1	0	1	N.C.	
1	1	0	Toggle	Q <sup>+</sup> = Q + T

→ Excitation Table: To get particular O/p with possible ways of I/p.

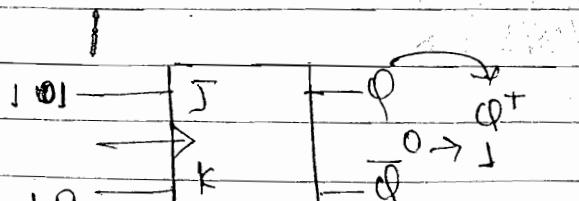
→ consist of possible ways of I/p to get particular O/p

For JK flip-flop:

For S.R flip-flop:

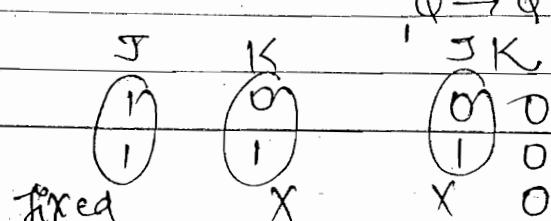
Q	Q <sup>+</sup>	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

Q	Q <sup>+</sup>	S	R
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0



For D flip-flop:

Q	Q <sup>+</sup>	D
0	0	0
0	1	1

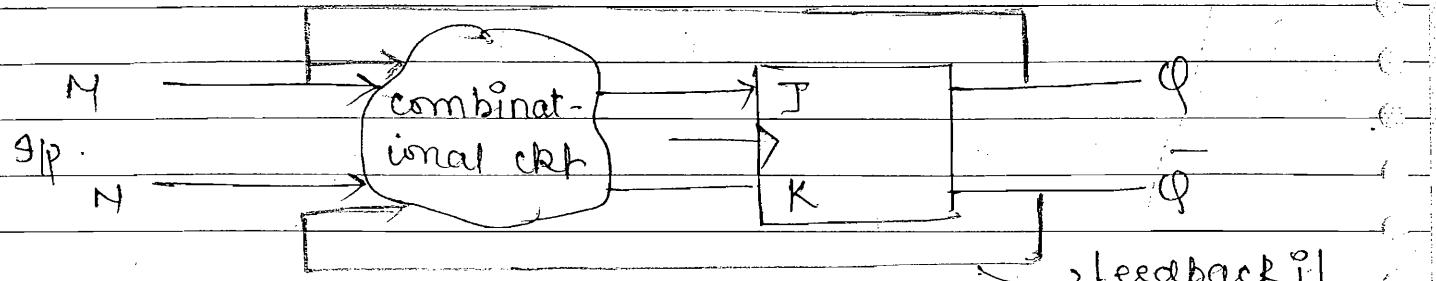


Truth Table :- O/p with desire o/p.

For T flip flop.

Q	Q <sup>+</sup>	T
0	0	0
0	1	1
1	0	1
1	1	0

• Designing of flip flop :- (By using JK)



client requirement			table results		
Q	M	N	Q <sup>+</sup>	J	K
0	0	0	1	1	X
0	0	1	0	0	X
0	1	0	1	1	X
0	1	1	1	1	X
1	0	0	X	X	X
1	0	1	0	X	1
1	1	0	1	X	0
1	1	1	X	X	X

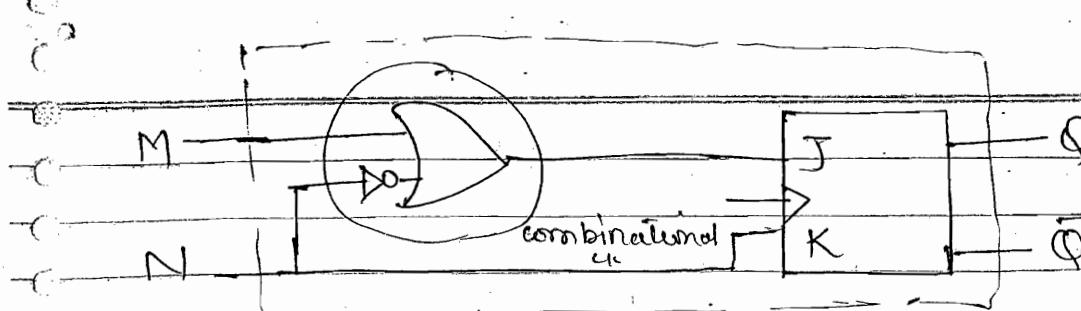
• If we don't get X from table, then draw K-map of J, K

Excitation Table of JK

Q	Q <sup>+</sup>	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

for J		for K			
Q	Q <sup>+</sup>	M̄N̄	M̄N	MN	MN̄
0	1	1	1	1	1
0	X	X	X	X	X

for J		for K			
Q	Q <sup>+</sup>	M̄N̄	M̄N	MN	MN̄
0	X	X	X	X	X
1	X	1	X		



Designing of flip-flop using SR: -

With Table of JK  $\rightarrow$  Client Requirement

$Q$   $A \oplus J$   $B \oplus K$   $Q^+$   $S$   $R$

$Q$	$A \oplus J$	$B \oplus K$	$Q^+$	$S$	$R$
0	0	0	0	0	X
0	0	1	0	0	X
0	1	0	1	J	0
0	1	1	1	J	0
1	0	0	1	X	0
1	0	1	0	0	1
1	1	0	1	X	0
1	1	1	0	0	1

SR excitation Table.

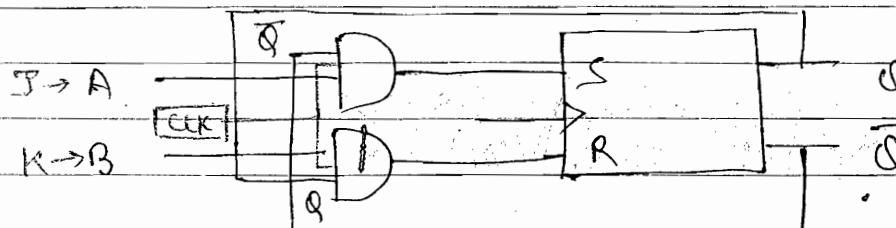
$Q$	$Q^+$	$S$	$R$
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0

$AB$	$\bar{AB}$	$AB$	$\bar{AB}$
Q	0	0	1
Q	X	S	1

$AB$	$\bar{AB}$	$AB$	$\bar{AB}$
Q	X	X	1
Q	4	1	1

$$S = \bar{Q}A$$

$$R = QB$$



S-R flip-flop convert into JK flip-flop = CLR  
is externally applied.

Requirement  $\rightarrow$  Truth Table  
By using  $\rightarrow$  Excitation Table.

- All excitation Table are for  $\rightarrow$  Active High
- This JK flip-flop and previous JK are same.  
because previously having NAND gate  $S R \rightarrow$  Active low connected into high Active. But in  
this AND are used Because  $S R \rightarrow$  is already Active high (Excitation Table).

~~Q Design~~  
~~Ans~~  $\rightarrow$  D-F-F By using S-R:-

Delay  $\rightarrow$  Truth Table

S-R  $\rightarrow$  Excitation Table.

~~Q Hows~~

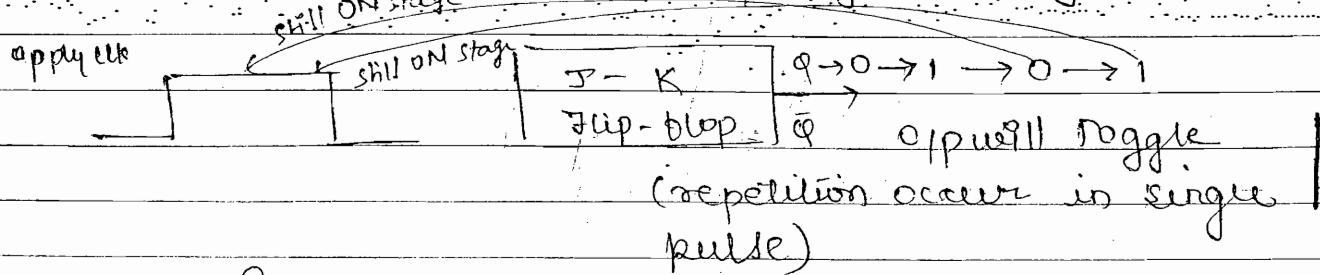
① S-R  $\Rightarrow$  J-K, T, D

② D  $\Rightarrow$  S-R, J-K, T

③ J-K  $\Rightarrow$  S-R, T, D

④ T.  $\Rightarrow$  J-K, S-R, D...

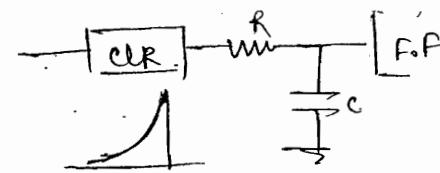
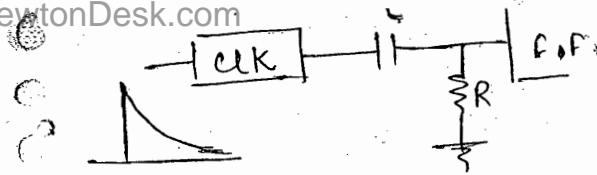
- Clk pulse width  $\uparrow \rightarrow$  propagation delay  $\downarrow$ .



Because after changing Q from initially  $0 \rightarrow 1$  Clk is still ON stage so again change  $1 \rightarrow 0$ ,  $0 \rightarrow 1$ ,

$\Rightarrow$  problem called Race Around problem.

- $\Rightarrow$  To Avoid is ① Reduce pulse width, OR  
② Use filter (RC).



- Race Around Problem :- When  $J=1, K=1$  and clock pulse is applied. Then we get the toggle condition. But when the op is occurring at the instant of time when the S/p pulse is ON state then there is repetition of Toggle at the output for the single pulse of S/p known as race around problem.

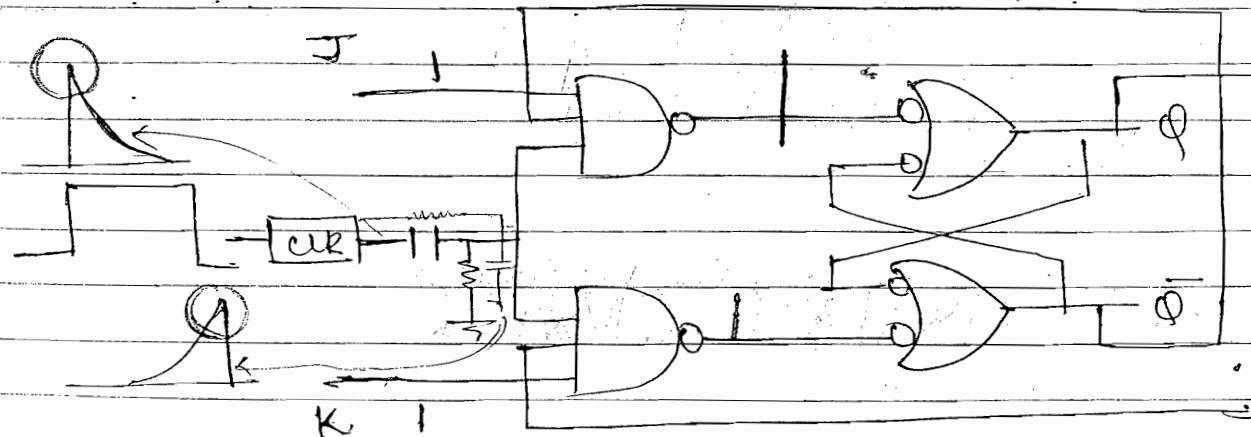
- Methods to avoid Race Around Problem :-

- Reducing the pulse width.
- By using the edge triggering.
- By using MASTER SLAVE J-K Flip-Flop.

NOTE Edge triggering two types:-

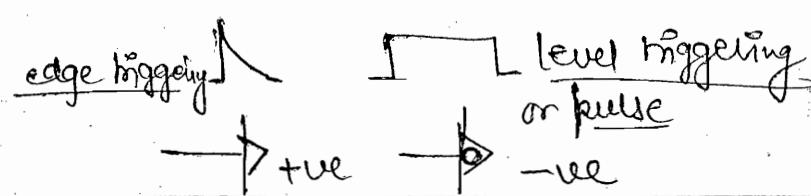
(i) POSITIVE EDGE TRIGGERING (leading edge triggering) (rising edge)

(ii) NEGATIVE EDGE TRIGGERING (trailing edge triggering) (falling edge)



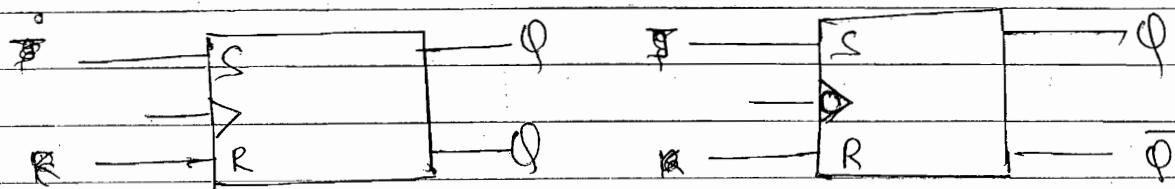
- Green position having sufficient magnitude to trigger remaining NOT having

so.  $clk \rightarrow 0$



• Timing Diagram :-

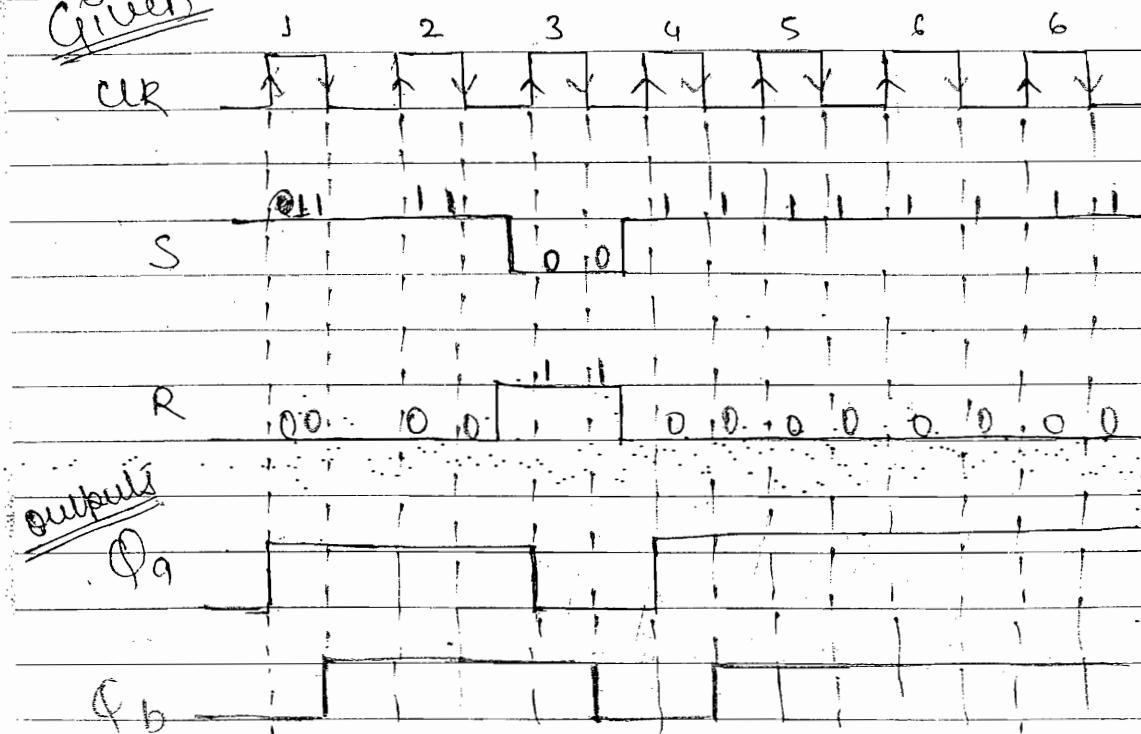
Edge Triggering



-ve edge triggering.

-ve E.T.

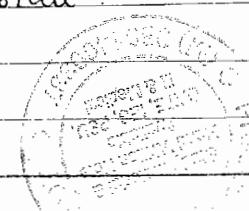
Given



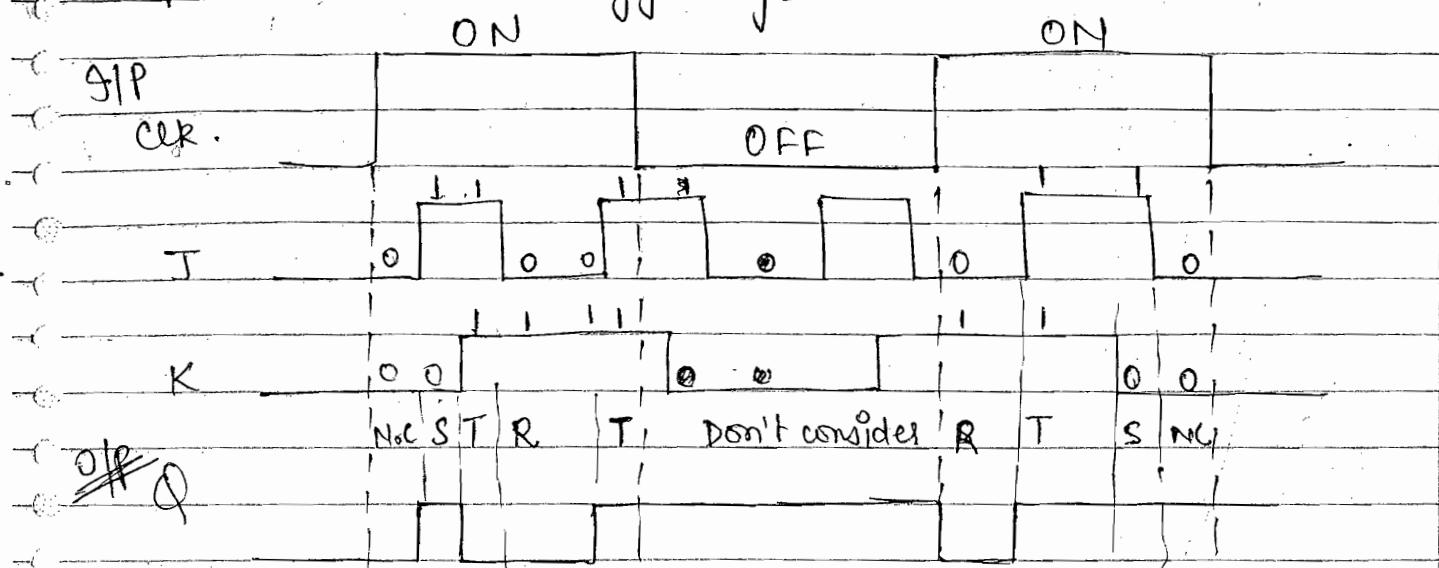
→ By using the result of S-R flip-flop, 0 0 → NC  
 $0 \downarrow \rightarrow$  Reset,  $1 \downarrow \rightarrow$  Set and  $1 \uparrow \rightarrow x$  we draw the timing op.

→ Before drawing, first write the S, R state

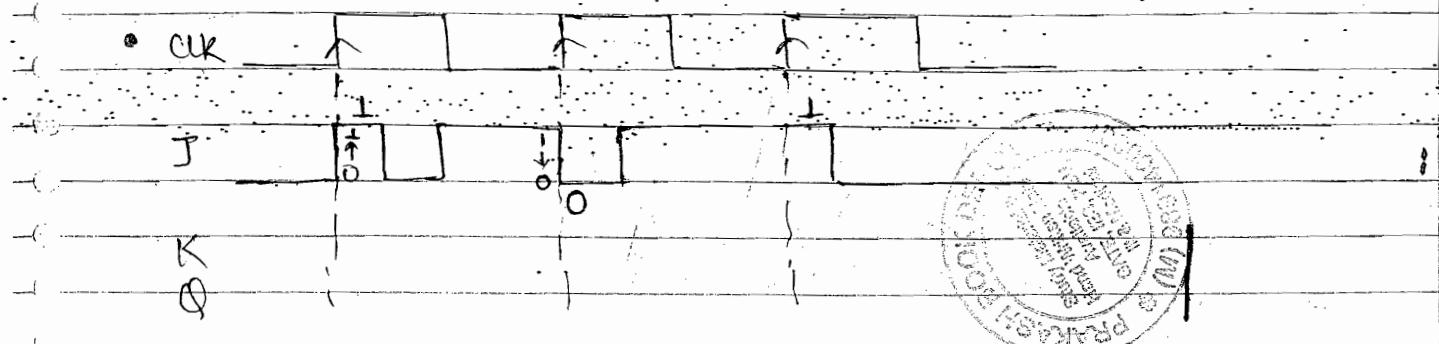
- → +ve triggering
- → -ve triggering



## Positive OR. level Triggering:-



→ OFF level triggering period don't consider, the ON state remains continuous upto next ON stage, In toggle period stage will change ( $0 \rightarrow 1, 1 \rightarrow 0$ ).

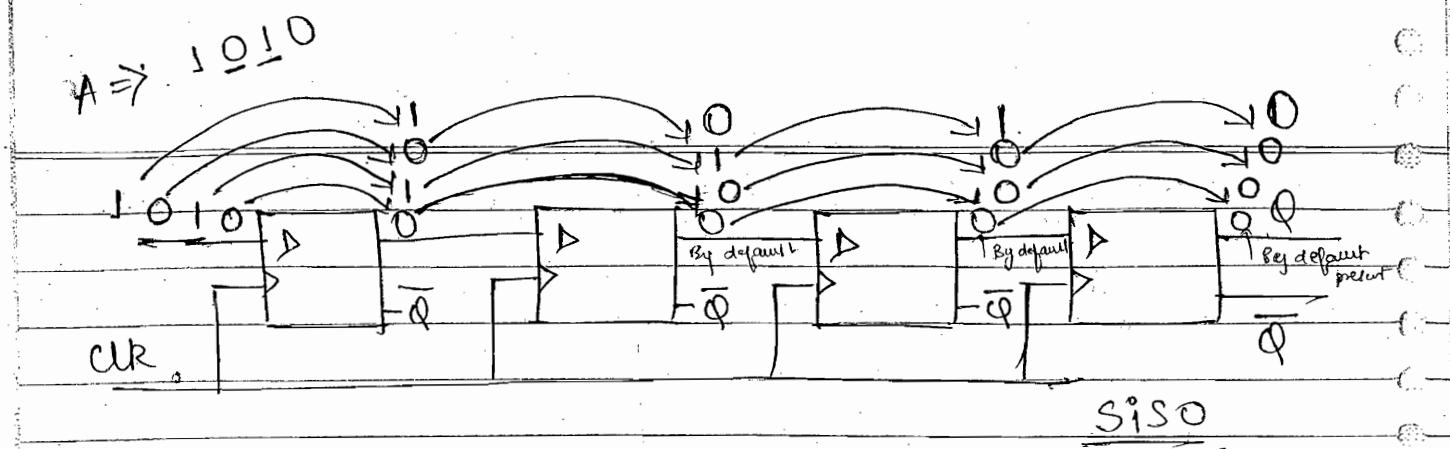


## Parallel Binary Shift Registers:-

Flip flop connected in cascade manner is known as Binary Shift Registers.

### Types

- (1) Serial In Serial Out (SISO)
- (2) Serial In parallel Out (SIPO)
- (3) Parallel In Serial Out (PiSO)
- (4) Parallel InParallel Out (PiPO)



	<u>S<sub>i</sub> S<sub>0</sub></u>	<u>n</u>	<u>Q<sub>i</sub> P<sub>0</sub></u>	<u>(n-1)</u>
(1)	S <sub>i</sub> S <sub>0</sub>	n	0	0
(2)	S <sub>i</sub> P <sub>0</sub>	n	1	0
(3)	P <sub>i</sub> S <sub>0</sub>	1	1	(n-1)
(4)	P <sub>i</sub> P <sub>0</sub>	1	0	0

- No. of S<sub>i</sub> Bit  $\Rightarrow$  No. of clock pulse required to store these Bit in S<sub>i</sub>, and only one clk pulse to input the bit in F.F. (Data is already present at Q<sub>i</sub>. So (n-1) Bit to come out data for S<sub>0</sub> and 0 for P<sub>0</sub>).

### • Application of Binary Shift Register:

- They are used for temporary data storage.

- These will provide the time delay.

$$\text{Time Delay } \Delta t = N \cdot T_c \quad N \rightarrow \text{No. of clk pulse.}$$

$$= N \cdot \frac{1}{f_c}$$

- These are used for data conversion ex: Sipo convert serial form of data into parallel form and piso convert parallel data into serial form.

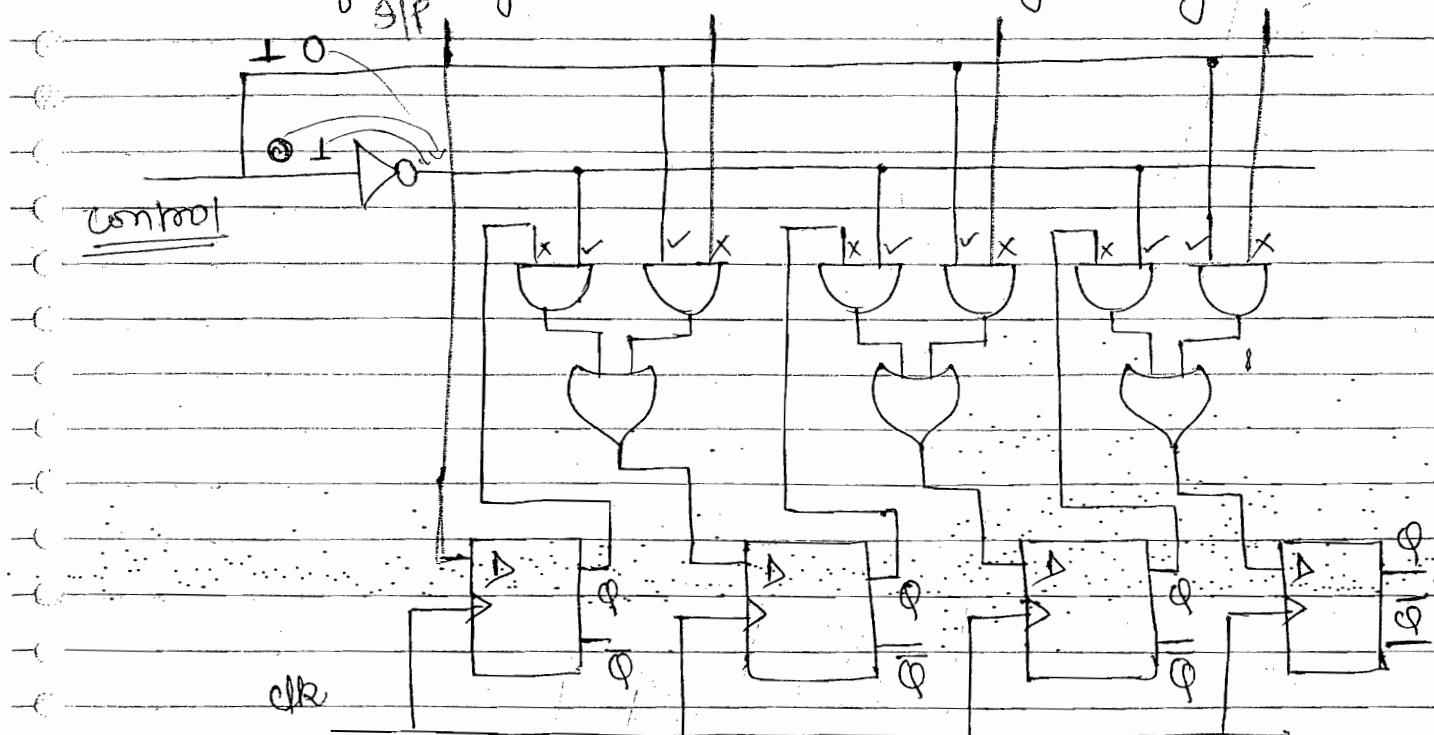
→ Left Shift  $\Rightarrow$  Multiplication by 2.

(4) These are used for arithmetic operation.

Ex:- left shift register is multiplication by 2  
right shift register is  $\rightarrow$  division by 2

NOTE:- Right shift Register having error of 0.5 for odd numbers

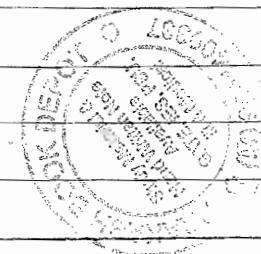
(5) Shift Registers are used to design Ring Counters.



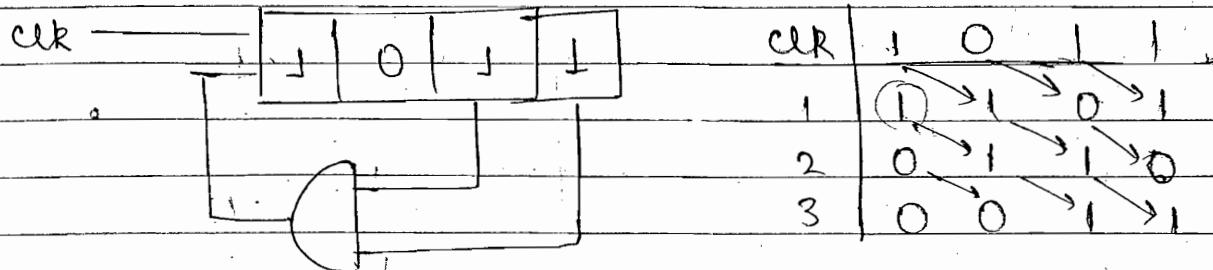
• When control pulse is 0  $\rightarrow$  SISO

• When control pulse is 1  $\rightarrow$  PISO

To make Serial output op. control will change to 0.



Q After 3. pulse what's O/p?



10011 → o/p.

23/9/2013

- Special type of Syn. Counter  $\rightarrow$  Ring counter.
  - No. of possible state  $\rightarrow$  Mod value.

ex: 8  $\rightarrow$      $\rightarrow$  Mod 8

No. of flip flops used

Georgian 18th century

Ex: Mod 10  $\rightarrow$  No. of flip flop used  $\rightarrow 5$

- Timing diagram  $\rightarrow$  Take columns and draw in  $\Sigma$

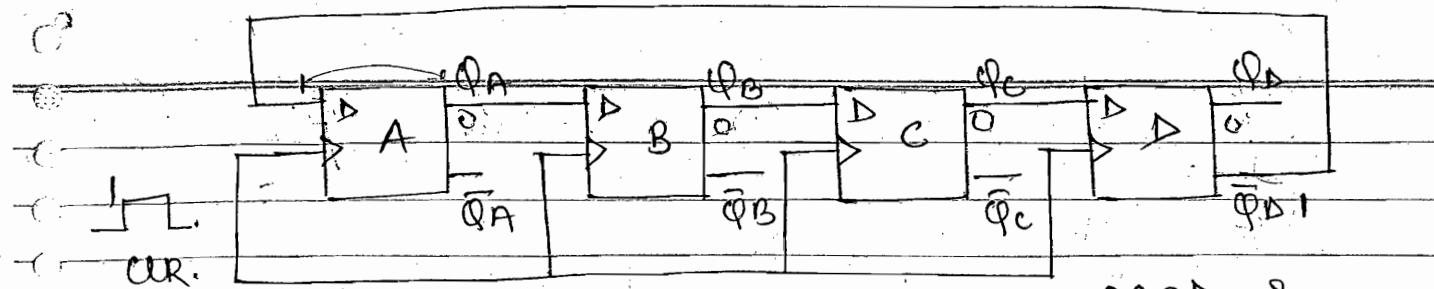
11

Counter: It counts the no. of clk pulses applied to it. Counter are of two type

- (i) Synchronous Counter (Parallel Counter)
  - (ii) Asynchronous Counter (Ripple Counter, Wave)

→ Ring Counters are special category of syn. counters.  
Categories in two ways.

- (i) Johnson's Ring Counter or (Twisted Ring Counter)  
"Feedback" is given from complementary output  
of last flip flop.



MOD - 8

CLR	$Q_A$	$Q_B$	$Q_C$	$Q_D$	1	2	3	4	5	6	7	8
1	1	0	0	0	0	1	0	0	0	0	0	0
2	1	1	0	0	0	1	0	0	0	0	0	0
3	1	1	1	0	0	1	1	0	0	0	0	0
4	1	1	1	1	1	1	1	1	0	0	0	0
5	0	1	1	1	1	0	1	1	1	0	0	0
6	0	0	1	1	1	0	0	1	1	1	0	0
7	0	0	0	1	1	0	0	0	1	1	1	0
8	0	0	0	0	0	0	0	0	0	0	0	1
9	1	0	0	0	X							

Repeated. Not used.

formula -  $n \rightarrow 2^n$   $\rightarrow$  possible states  
 $\rightarrow$  CLR pulse.

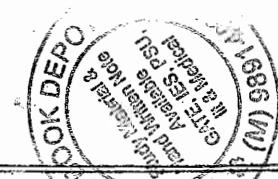
no. of FF. Mod value

Initially  $Q_A, Q_B, Q_C, Q_D$  are at 0 states. When applied CLR pulse.  $Q_A = 1$ , and its binary shifted shown in table.

Q. What are the used and un-used stages of 4-bit Johnson's ring counter and what will be the next stages for un-used states and check whether it is self corrected counter or not.

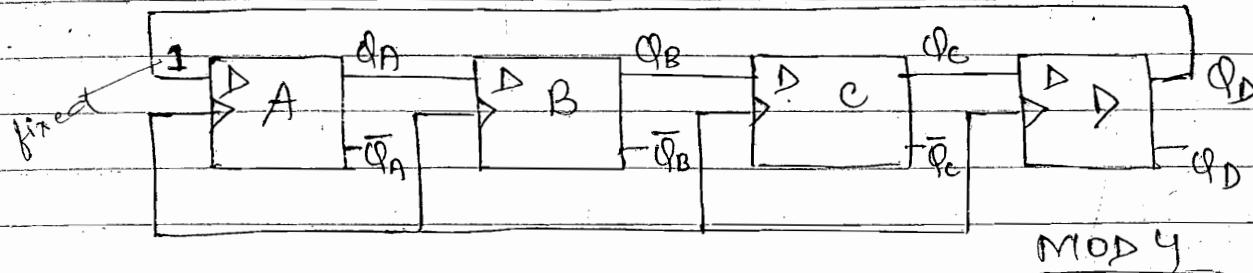
Ans possible states are  $2^n \rightarrow 16$  but getting 8 because of such connection. So used other counter.



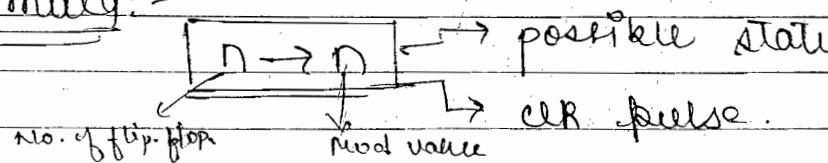


## (ii) Ordinary Ring Counter or Ring Counter:

- Feedback is given from an complementary O/p of last flip-flop.



formula:-



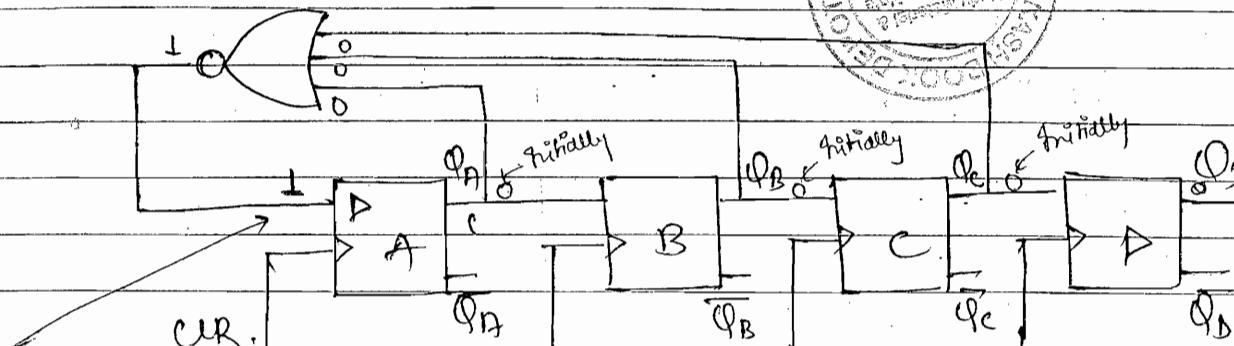
→ first F.O.F. set always to 1 to start it. If 0 is applied always get 0000, NO op..

- To Make it Self. Started Add. NOR Gate

we get same Table. So called

## Self Started Ring Counter

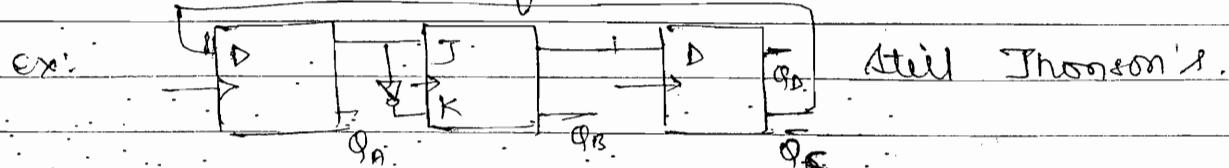
- Set started ring counter



→ It is ready for CLR.

- unused state for Johnson's  $\Rightarrow 2^n - 2^n$

- Un-used state for ring counter  $\Rightarrow 2^n - n$

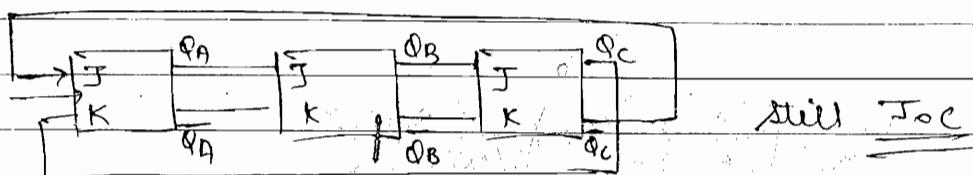
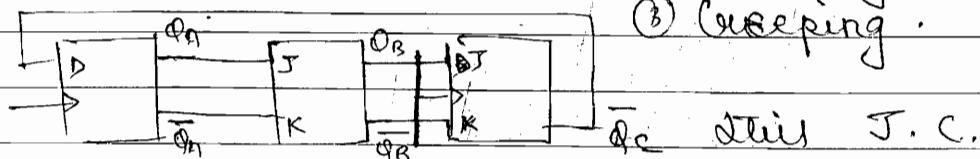


Other names of Johnson's :-

⑥ Switch Tail

⑥ Walking

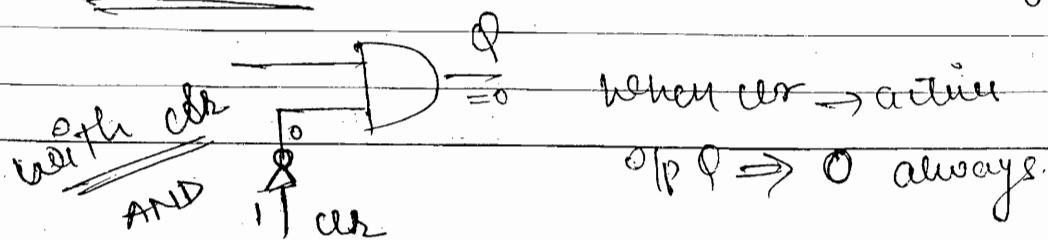
⑥ Counting



F.P. Q.P.

Asynchronous:-  $0/p Q \rightarrow$  takes with comb of  $\rightarrow CLR + Q$

$= Q$   
final.

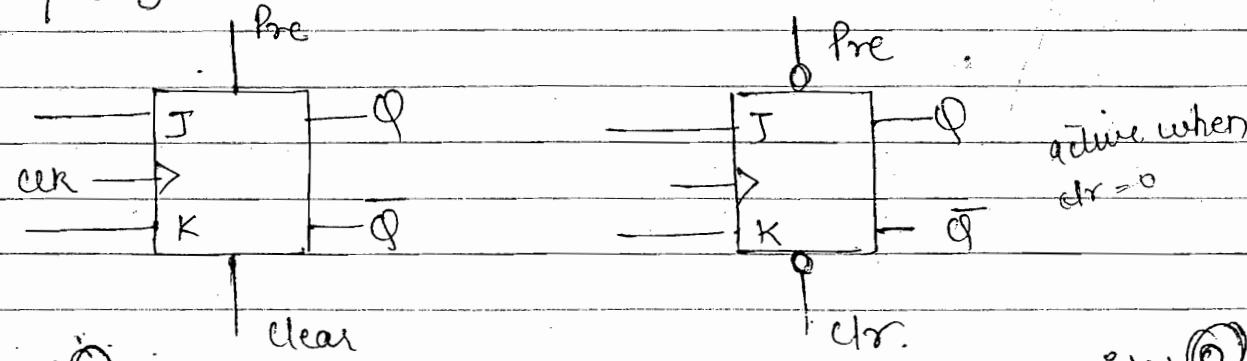


Pre

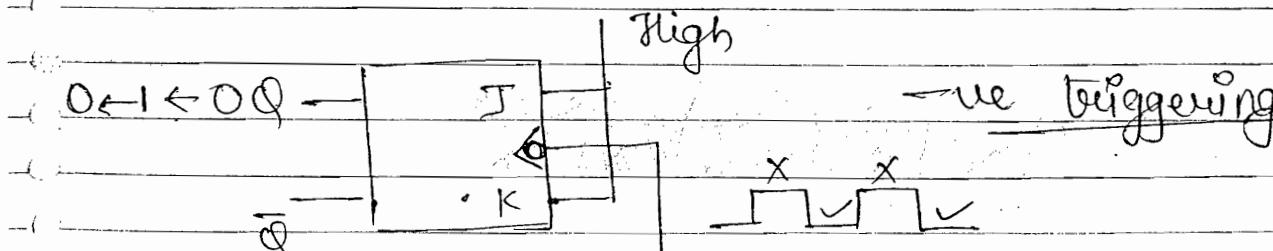
with Pre  
orIf pre is active ( $Q = 1$ )  
always.

- If clr or Pre active  $\rightarrow$  T & K inputs are not considered
- If both are 0 and 0  $\rightarrow$  T & K inputs  $\rightarrow$  0 output  $\rightarrow$  which is 0.
- If Both are 1 and 1  $\rightarrow$  0 output  $\rightarrow$  which having fast speed.

### Asynchronous Counter :- (Preset AND Clear)



Pre	clr	Q <sup>+</sup>	Pre	clr	Q <sup>+</sup>
0	0	flip-flop		1	f.f.
0	1	0	1	0	0
1	0	1	0	1	1
1	1	X	X	X	don't care



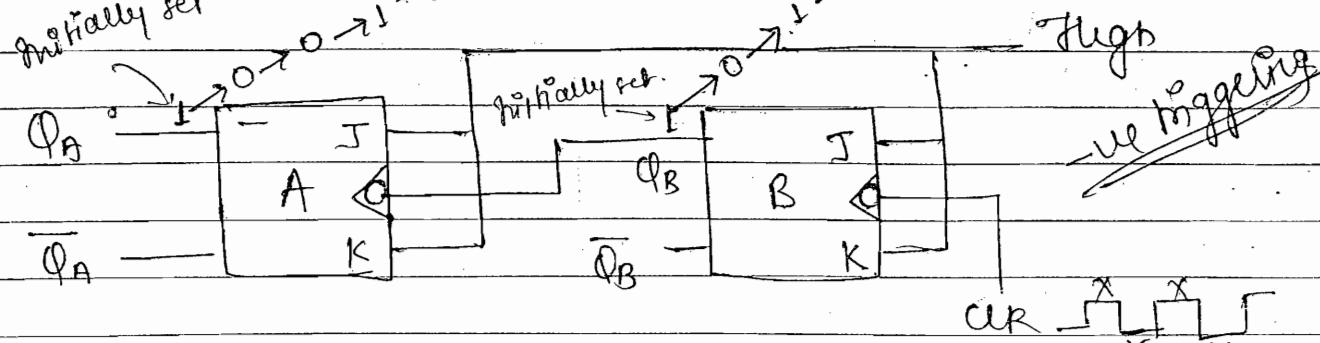
$\Rightarrow$  Give 0p in -ve pulse. T flip flop

Conventional + Objective

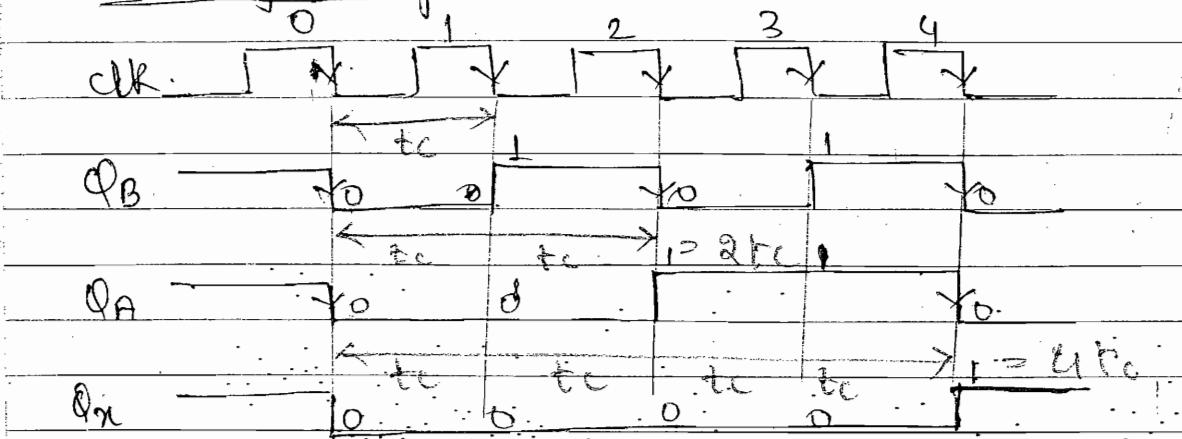
Asynch  $\rightarrow$  F. F. are not triggered simultaneously  
F. F. triggered by previous F. F. output.

### Asynch (Ripple) :-

Initially set



### Timing diagram



### Table

clk	Q_A	Q_B
0	0	0
1	0	1
2	1	0
3	1	1
4	0	0

Mod 4

formula

$$n \rightarrow 2^n$$

possible state

No. of F.F.

Mod value

clr.

- If there are  $n$  no. of flip-flop, first draw the timing diagram then Truth Table

- In 3 F.F. when 3 F.F. trigger flip-flop get next state pulse so consider 1 F.F. previous state.

- Consider all 1st state of all F.F.

# Ripple $\rightarrow$ frequency divider

$Q_B \rightarrow$  trigger by  $\text{clk} \overset{\text{edge}}{\text{edge}} \Phi_A \rightarrow$  trigger by  $-\text{ve edge of}$   
 $Q_B \rightarrow$  be triggered by  $-\text{ve edge of } \Phi_A$

- Also called frequency Dividers

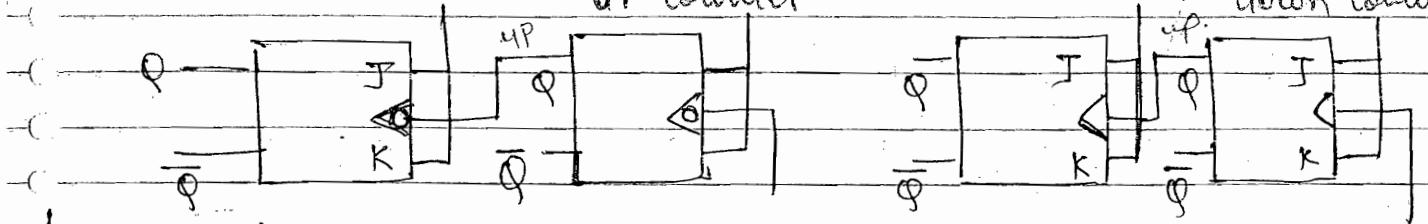
Because Time period multiple by 2

$\downarrow$   
frequency. Divided by 2.

- (-ve) triggering  $Q$       (+ve) triggering  $Q$ .

UP counter

down counter

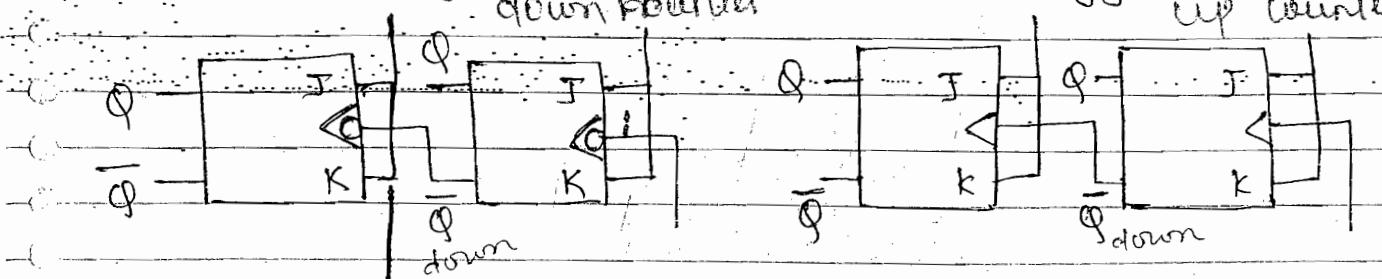


- (-ve) trigger  $\bar{Q}$

down counter

- (+ve) trigger  $\bar{Q}$

up counter



Shortcut

(-ve) triggering  $\rightarrow$  up connecting up counter  
 $\rightarrow$  down connected down counter

the triggering  $\rightarrow$  up connecting down counter  
down connected up counter.

$$\begin{array}{ccccccc} 11000 & & & & & & \\ 0 & 0 & 2 & 2 & 2 & 0 & \\ & & & & & & \\ 16 & & & & & & \\ & & & & & & \end{array}$$

## • Designing of Asynchronous MOD Counter :-

~~shorted~~ Right Trigger  $\rightarrow$  Right Code

Left trigger  $\rightarrow$  Opposite code.

Ex:- 1101  $\rightarrow$

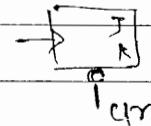
1011

• If  $clr$  is triggered by  $\rightarrow 1$

active

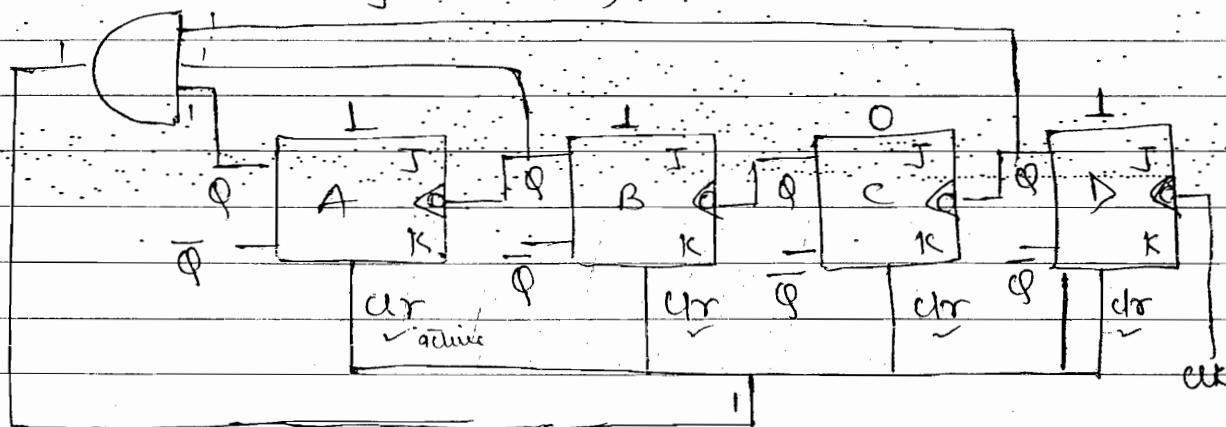
AND Gate Used.

• If  $clr$  is active by  $\rightarrow 0$  NAND Gate Used



$\overline{clr}$

• AND OR NAND gate are connected by 1  
(those (F.F.) having 1 value)



for MOD 13

Right Trigger

1101 (Code)

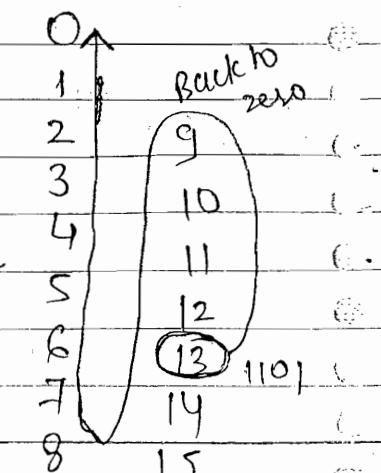
Steps

Given Design MOD 13 First write the code of 13. 1101, This code write in F.F. Those flip flop having 1 at O/p.  $\rightarrow$  connect those F.F. with AND

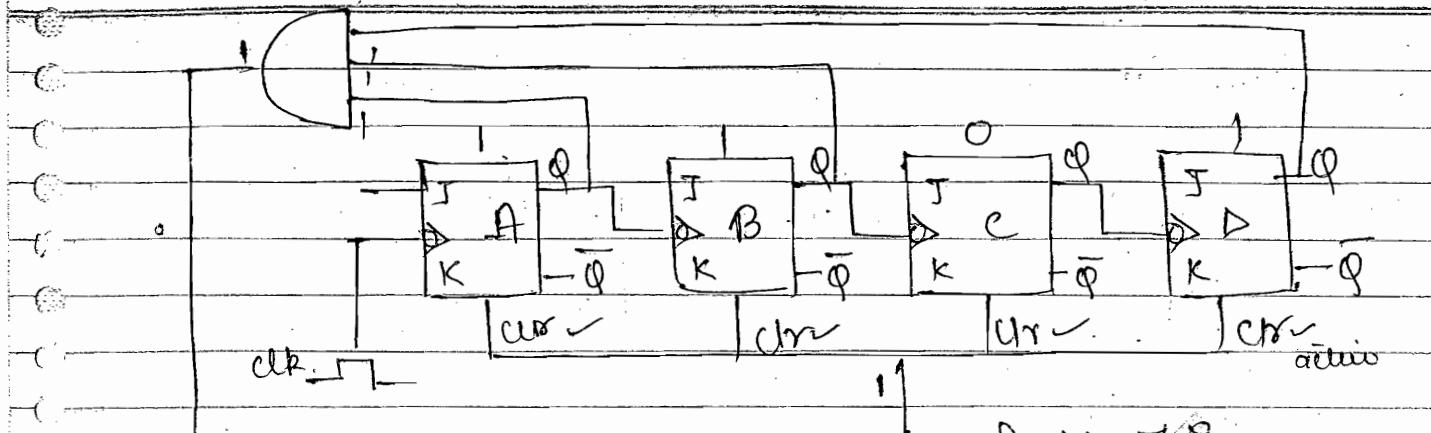
get to get 1 at O/p AND gate so that  $clr$

will get active and counter O/p become 0000 (reset)

Back to zero



for MOD 13 1101



left trigger

(code will reverse)

MOD 13  $\Rightarrow$  1101

1011

MOD 11

Right trigger

A B C D A B C D

0 1 0 0 0 0 0 0 left trigger.

1 0 0 0 1 0 0 0

2 0 0 1 0 0 1 0 0

3 0 0 1 1 1 1 0 0

the unstable

4 0 1 0 0 0 0 1 0 state from where

F.F. Reset to

5 0 1 0 1 0 1 0 0

0000 don't

6 0 1 1 0 1 0 1 0

count is counted

7 0 1 1 1 1 1 1 0

and called

8 1 0 0 0 0 0 0 1

Quasi state

9 1 0 0 1 1 0 0 1

10 1 0 1 0 1 0 1 0 1

11 1 0 1 1 1 1 1 0 1

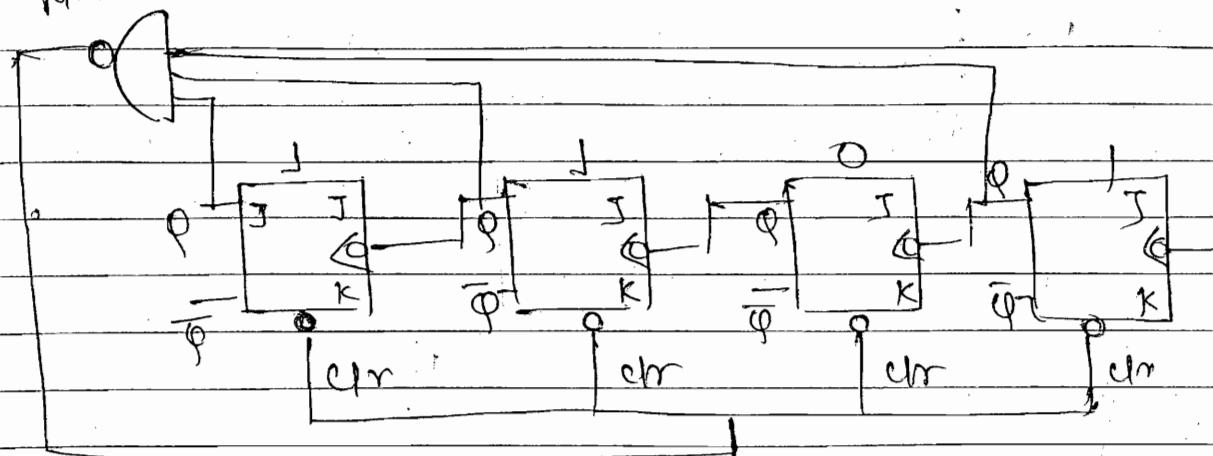
12 1 1 0 0 0 0 1 1

13 1 1 0 1 1 0 1 1

14 1 1 1 0 0 1 1 1

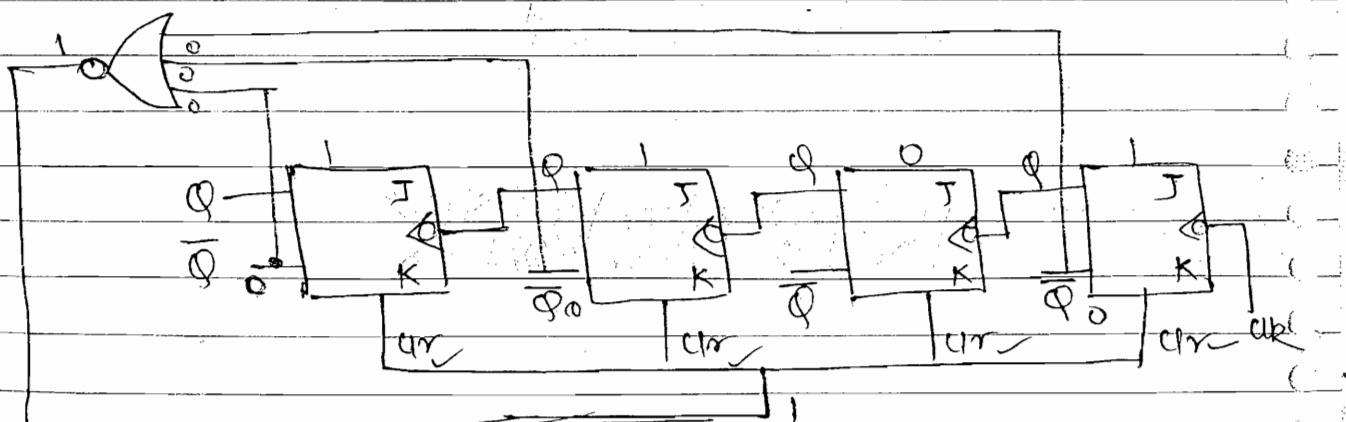
15 1 1 1 1 1 1 1 1

if  $\bar{Q}$  having Bubble  
NAND Gate



- As we are using AND gate, NO Need of CLR pulse is required to make F.O.F. in Reset position after the undesirable state comes.  
ex:- Mod 13  $\rightarrow$  1101. When this comes, F.P. O/p reset to 0000 automatically.
- But if we don't use AND gate counter O/p will Reset only when CLR pulse will applied. automatic action will not occur.

By using NOR Gate :- Mod 13 - 1101 24/9/2014



- Connect with  $\bar{Q}$ . don't connect with Q O/p F.F. because before coming of 13, Q will comes in some Before state, so it will return from that state Not comes upto 13.

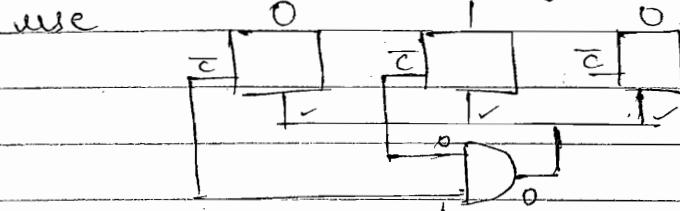
Shortcut:-

NAND	Q	$\overline{C_{tr}}$
NOR	$\overline{Q}$	$C_{tr}$
AND	Q	$C_{tr}$
OR	$\overline{Q}$	$\overline{C_{tr}}$

Page No. 61

Q. 39  $\overline{C_{tr}}$ ,  $\overline{110} \rightarrow Q$  So OR gate. MOD 6 0005

Don't use AND gate because if we



0005  
001  
010  
011  
100  
101

Return Back  
 from 010 and become MOD 2

### • Down Counter :-

Connect the the gate  $g_{ff}$  with 1 with the 1 in that position, and take the code, that code ~~is not~~ subtracted by Maximum value.

No. of F.F.

Ex:- 101  $\rightarrow 7 - 5 = 2$  MOD 2 (3 flip-flop)

Ex:- 101  $\rightarrow 15 - 5 = 10$  MOD 10 (4 flip-flop).

- Right Trigger  $\rightarrow$  Right code (value)
- Left Trigger  $\rightarrow$  Reverse value & subtract  
By maximum value (state).

### • Procedure To find Mod Value for A syn. Up Counter

Step 1:- Keep the one's (1's) at those flip-flops which are having connection to the logic gate of other F.F. should be kept at zero]

Step 2:- Check whether it is right trigger or left trigger.

Right Trigger  $\rightarrow$  Right code  
 Left Trigger  $\rightarrow$  Reverse code is the Mod value

• Procedure for Down Counter:

Step 1 Keep the 1's at those position from where logic gate is having connection. Other position should be kept at zero.

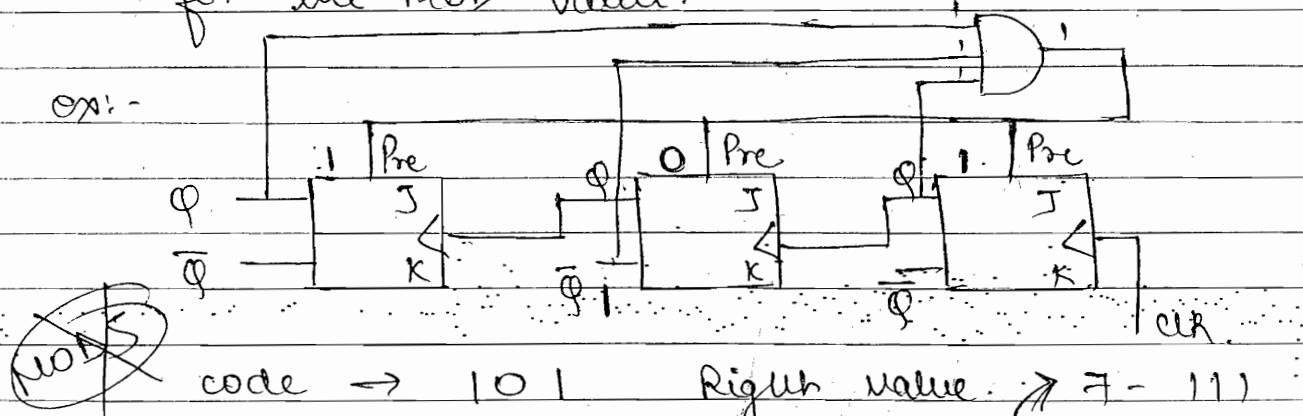
Step 2 check whether it is right trigger or left trigger.

Right Trigger  $\rightarrow$  Right value

Left Trigger  $\rightarrow$  Reverse value

Should be subtract from the maximum state for the MOD value.

Ex:-



code  $\rightarrow$  101

Right value  $\rightarrow 7 - 111$

Maximum state  $\rightarrow 7$

$$\Rightarrow 7 - 5 = 2$$

$$b = 110$$

$$S \ 101X$$

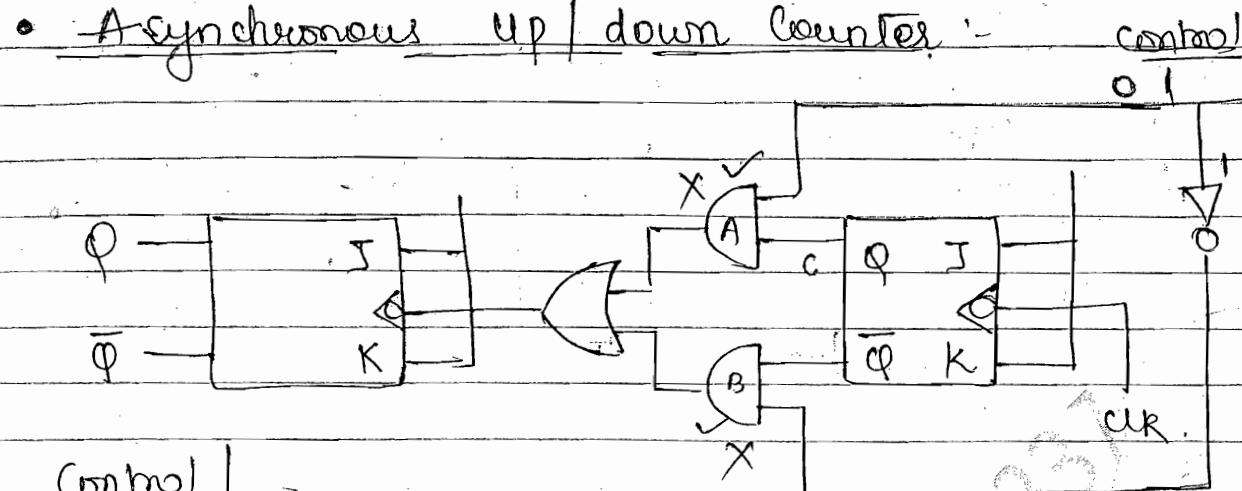
$$= \underline{\text{MOD 2}}$$

Ex: Left trigger. Suppose code is 100

Reverse value is 001

$$\begin{aligned} \text{Maximum state} &\rightarrow 7 \rightarrow 7 - 1 = 6 \\ &= \underline{\text{MOD 6}} \end{aligned}$$

- Asynchronous Up/Down Counter



Control

0 B operate  $\rightarrow$  down counter (pass the value present in  $\bar{Q}$ )

1 A operate  $\rightarrow$  up counter (pass the value present in  $Q$ )

Conventional

- Asynchronous Counter :- [ORDerly Sequence]

~~NO J/K~~ (NO check for Right OR left Trigger). Simultaneously Trig.

Synchronous Up Co.

Syn. Down Counter

A B C D

A B C D

0 0 0 0

1 1 1 1

0 0 0 1

1 1 1 0

0 0 1 0

1 1 0 1

0 1 0 0

1 0 1 1

0 1 1 0

1 0 0 1

0 1 0 1

1 0 1 0

0 1 1 1

1 0 0 0

1 0 0 1

0 1 1 0

1 0 1 0

0 1 0 1

1 0 1 1

0 1 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

1 1 0 1

0 0 1 0

1 1 1 0

0 0 0 1

1 1 1 1

0 0 0 0

1 1 0 0

0 0 1 1

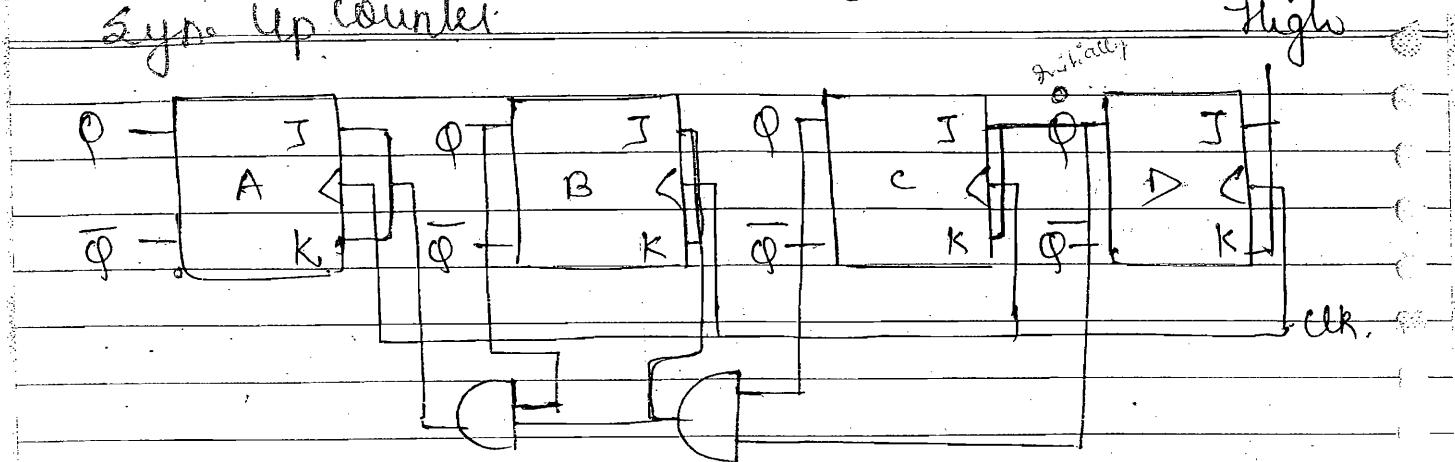
1 1 0 1

0 0 1 0

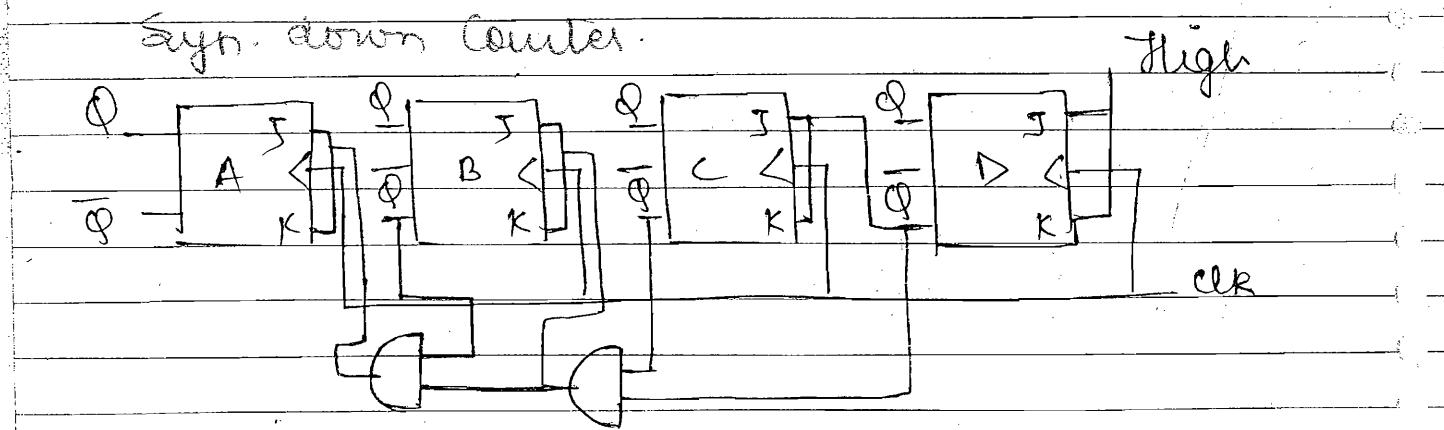
1 1 1 0

# Called Synchronous Series Carry

Syn. Up Counter



Syn. down Counter



- When  $Q_n = 1$   $C \rightarrow$  toggle  $\therefore Q_n = Q_C = 1 \rightarrow Q$  (connected)

$Q_C = 1 \rightarrow B \rightarrow$  toggle  $\therefore Q_B = 1$   $\therefore Q_B = Q_D = 1$  so we AND gate, together give 91p to  $J-K$  (B),  $A \rightarrow$  toggle when  $Q_D = Q_C = Q_B = 1$  so we AND gate, together give 91p. to  $J-K$  (A),

objection

• Synchronous

Time delay

$N \cdot T_C$

• Synchronous

$T_C$

• Synchronous Series  
Carry

$T_C + (N-2)T_G$

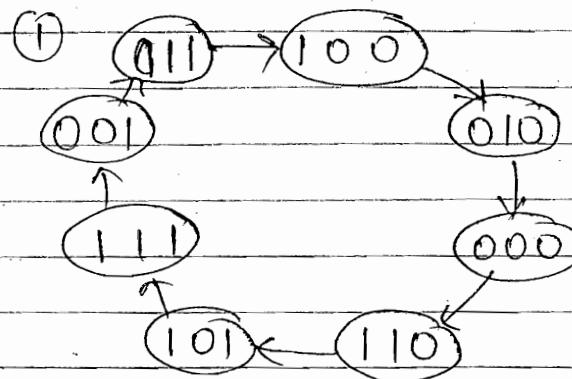
$T_G \rightarrow$  Gate delay.

- If Mention Syn. Series Carry or Orderly Manner go for this Method otherwise Next coming Method Any Order

## Synchronous Design (No Need of Rde)

S N E K B  $\rightarrow$   
 State Next Excitati K-Map Boolean Design  
 Diagram. state Table equation

Client requirement: - 4 2 0 6 5 7 1 3. (By JK)



③ K-Map for JK

$Q_1$	$Q_0$	$J$	$K$
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

② Present state | Next state |  $J_0\ K_0$  |  $J_1\ K_1$  |  $J_2\ K_2$

	$Q_2\ Q_1\ Q_0$	$Q_2^+\ Q_1^+\ Q_0^+$			
4	1 0 0	0 1 0	0 X	1 X	X 1
2	0 1 0	0 0 0	0 X	X 1	0 X
0	0 0 0	1 1 0	0 X	1 X	1 X
6	1 1 0	1 0 1	1 X	X 1	X 0
5	1 0 1	1 1 1	X 0	1 X	X 0
7	1 1 1	0 0 1	X 0	X 1	X 1
1	0 0 1	0 1 1	X 0	1 X	0 X
3	0 1 1	1 0 0	X 1	1 X	1 X

	$Q_1\bar{Q}_0\ \bar{Q}_1\bar{Q}_0\ Q_1\bar{Q}_0\ Q_1\bar{Q}_0$	$\bar{Q}_1\bar{Q}_0\ \bar{Q}_1\bar{Q}_0\ Q_1\bar{Q}_0\ Q_1\bar{Q}_0$
$\bar{Q}_2$	0 X 1 X 3 2	$\bar{Q}_2$ X 0 1 X 3 X 2
$Q_2$	4 X 1 X 2 D	$Q_2$ X 4 5 7 X 6

$$J_0 = Q_2 Q_1$$

$$K_0 = \bar{Q}_2 Q_1$$

$\bar{Q}_1, \bar{Q}_0, \bar{Q}_1, Q_0, Q_1, Q_0, Q_1, \bar{Q}_0$ 

$\bar{Q}_2$	1	0	1	$X_3$	$X_2$
$Q_2$	1	1	0	$X_2$	$X_1$

 $\bar{Q}_1, \bar{Q}_0, \bar{Q}_1, Q_0, Q_1, Q_0, Q_1, \bar{Q}_0$ 

$\bar{Q}_2$	$X_0$	$X_1$	$X_2$	$X_3$	$X_4$
$Q_2$	$X_1$	$X_2$	$X_3$	$X_4$	$X_5$

$J_1 = \bar{Q}_1$

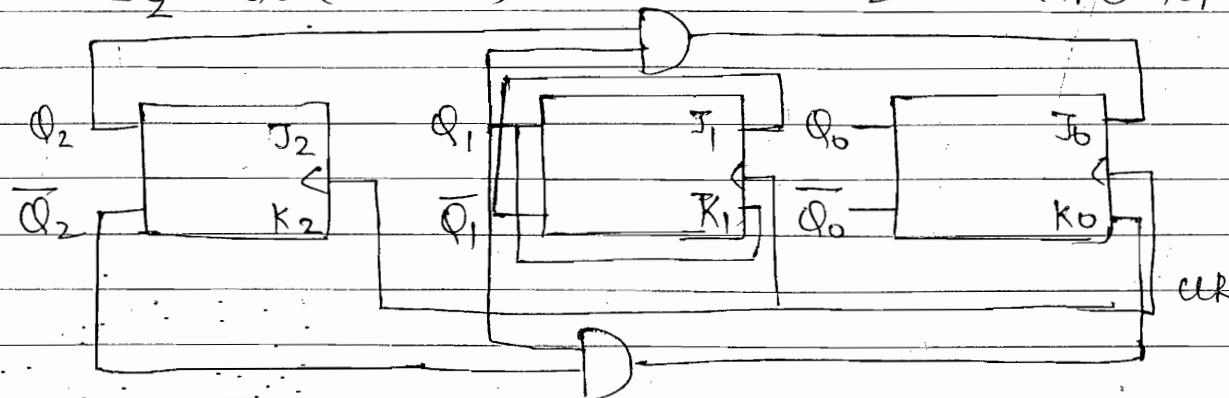
$\bar{Q}_2$	1	0	1	0
$Q_2$	X	X	X	X

$K_1 = Q_1$

$\bar{Q}_2$	$X_1$	$X_1$	$X$	$X$
$Q_2$	1	0	1	0

$J_2 = \bar{Q}_2 (Q_1 \odot Q_0)$

$K_2 = Q_2 (Q_1 \odot Q_0)$



T flip flop. 0 2 3 1 0

Present state | Next state |  $T_1 | T_2$

$Q$	$Q^+$	$T$	$Q_2$	$Q_1$	$Q_2^+$	$Q_1^+$	$T_1$	$T_2$
0	0	0	0	0	0	0	0	1
0	1	1	2	1	0	1	1	0
1	0	1	3	1	1	0	0	1
1	1	1	3	0	1	0	1	0

$\bar{Q}_1$	1
$Q_2$	1

$T_1 = Q_1 \oplus Q_2$

$\bar{Q}_1$	1
$Q_2$	1

$T_2 = Q_1 \odot Q_2$

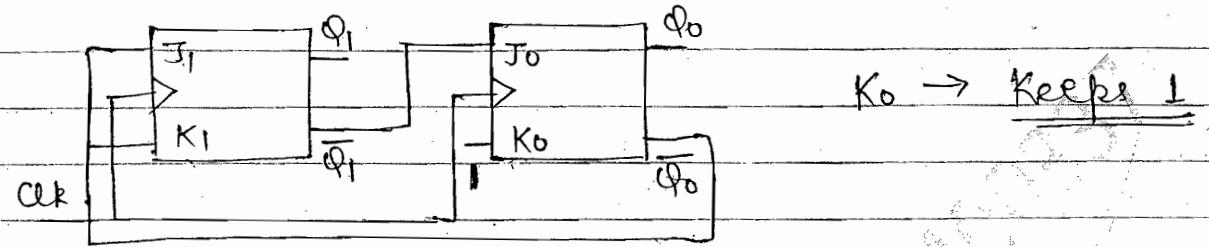
(a)

• Procedure to find MOD Value: For Syn. Counter:-

Step 1:- Write J/P and Q/P in form of Table

Step 2:- Write the J/P connection on their head.

Step 3:- Keep some default state and make a horizontal line.



Step 1  
 $\overline{Q_1}$  1  $\overline{Q_0}$   $\overline{Q_0}$  step 2

CLR |  $\overline{J_1}$   $\overline{K_0}$   $\overline{J_0}$   $\overline{K_1}$   $\overline{Q_0}$   $\overline{Q_1}$

0 0

$K_0 \rightarrow \text{Keeps 1}$

Step 2

1 1 1 1 1 1

0 1 0 0 0 1

0 1 1 1 0 0 X repeated

Step 3 } MOD 93.

~~An will be~~

$Q_0 Q_1 \Rightarrow 00, 11, 01, 00, \dots$

or 11, 01, 00, 11

$Q_1 Q_0 \Rightarrow 00, 11, 10, 00, \dots$

~~Q~~  $4(Q_1) + 2(Q_0) = ? \Rightarrow 4(0) + 2(0) = 0$

$4(1) + 2(1) = 6$

~~Ans~~  $\Rightarrow 0, 6, 4, \dots$   $4(1) + 2(0) = 4$

~~Q8.~~ Pg No 63.

~~Q8.~~ CLR |  $\overline{J_A}$   $\overline{K_A}$   $\overline{J_B}$   $\overline{K_B}$   $\overline{J_C}$   $\overline{K_C}$   $\overline{Q_A}$   $\overline{Q_B}$   $\overline{Q_C}$

1 1 0 2

1 1 0 1 1 1 0 0 0 1

0 1 1 0

↓

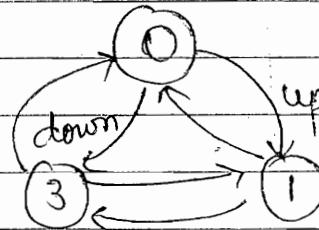
Repeat

~~Q8.~~ 110 will repeat after 2 pulse

MOD 2

- Synchronous (Up / Down)  $\rightarrow$  Externally control must
- K-Map for up counter  $\rightarrow$  Take both value of  $J_0$  (up & down)

Given.



	up	down
0	0	3
1	1	1
3	3	0

~~Up~~ Additional s/p for up.

<del>Up</del> $y=0$	Present state	Next state	$J_0$	$K_0$	$J_1$	$K_1$
$y=0$	$q_1\ q_0$	$q_1^+\ q_0^+$				
0	0 0	0 1	0	1	1 X	0 X
0	0 1	1 1	X	0	1 X	
1	1 1	0 0	X	1	X 1	

~~down~~ Additional s/p for down.

<del>down</del> $y=1$	P.S.	N.S.	$J_0$	$K_0$	$J_1$	$K_1$
1	1 1	0 1	0 1	X 0	X 1	
1	0 1	0 0	0 0	X 1	0 X	
1	0 0	1 1	1 1	1 X	1 X	

$q_0$	$q_1^+$	$J_1$	$K_1$	$q_0$	$q_1^+$	$J_1$	$K_1$
0	0	0	X	0	0	0	0
0	1	1	X	1	X	X	0
1	0	X		1	1	1	X
1	1	X	0				

$J_0 = 1$

$K_0 = \bar{y} \oplus q_1$

- When the empty space left keep their don't care.
- Both  $J_0$  of up/down taken in K-Map.

Both  $K_0$

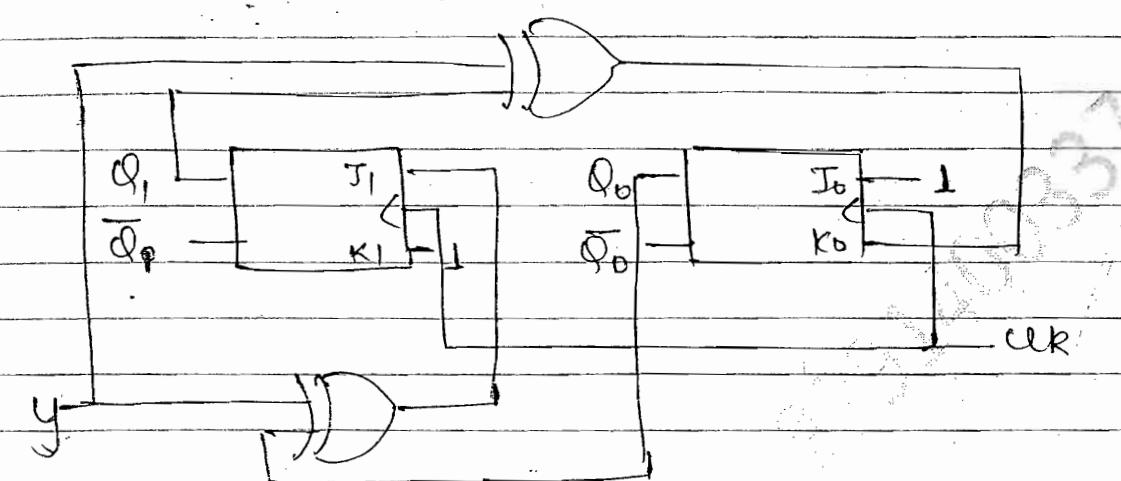
$\bar{Q}_1 \bar{Q}_0$	$\bar{Q}_1 Q_0$	$Q_1 \bar{Q}_0$	$Q_1 Q_0$
$y$	$0_1$	$1_1$	$x_1$
$y$	$1_0$	$0_0$	$x_0$

$\bar{Q}_1 \bar{Q}_0$	$\bar{Q}_1 Q_0$	$Q_1 \bar{Q}_0$	$Q_1 Q_0$
$y$	$x_0$	$x_1$	$1_1$
$y$	$x_0$	$x_1$	$1_1$

$$J_1 = \bar{y}Q_0 + y\bar{Q}_0$$

$$= y \oplus Q_0$$

$$K_1 = 1$$



Q Client Requirement :- 0 3 5 6 (up).

P. S.	N. S.	T <sub>0</sub>	T <sub>1</sub>	T <sub>2</sub>
$Q_2 \ Q_1 \ Q_0$	$Q_2^+ \ Q_1^+ \ Q_0$	$T_0 \ K_0$	$T_1 \ K_1$	$T_2 \ K_2$
0 0 0	0 1 1	1	1	0
0 1 1	1 0 1	0	1	1
1 0 1	1 1 0	1	1	0
1 1 0	0 0 0	0	1	1

$Q \ Q^+$	$T$	$\bar{Q}_1 \bar{Q}_0 \ \bar{Q}_1 Q_0 \ Q_1 \bar{Q}_0 \ Q_1 Q_0$	$\bar{Q}_1 \bar{Q}_0 \ \bar{Q}_1 Q_0 \ Q_1 \bar{Q}_0 \ Q_1 Q_0$
0 0	0	$Q_2 \ \bar{1} \ x \ 0_3 \ x_2$	$\bar{Q}_2 \ \bar{1} \ x_1 \ 1_3 \ x_2$
0 1	1	$Q_2 \ x_4 \ 1_5 \ x_7 \ 0_6$	$Q_2 \ x_4 \ 1_5 \ x_7 \ 1_6$
1 0	1		
1 1	0		

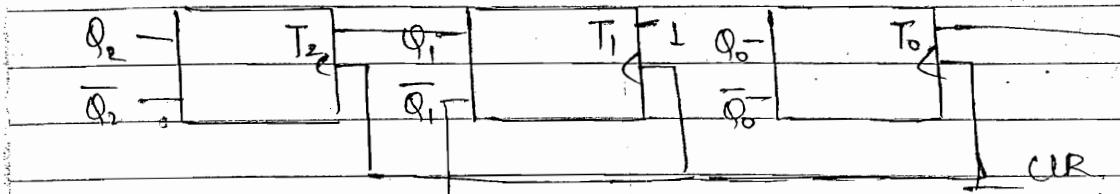
$$T_0 = \bar{Q}_1$$

$$T_1 = 1$$

$\bar{Q}_2$	0	x	1	x
$Q_2$	x	0	x	1

$$T_2 = Q_1$$

Wanted  $\Rightarrow 0 \ 3 \ 5 \ 6$       Unwanted  $\Rightarrow 1 \ 2 \ 4 \ 7$

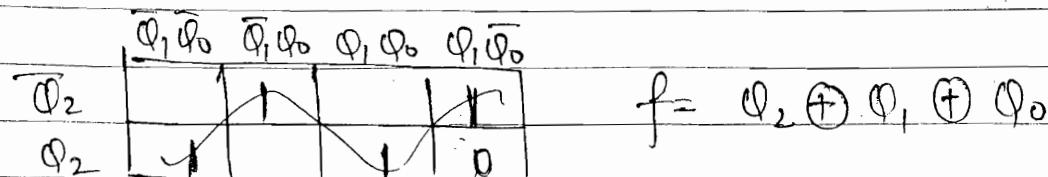


CLR	$Q_1$	$1$	$\bar{Q}_1$	$T_2$	$T_1$	$T_0$	$Q_2$	$Q_1$	$Q_0$	Suppose unwanted state $\rightarrow 4$
	1	0	1				1	0	0	
1	0	1	1	1	1	1	1	1	1	$1 \rightarrow 7$
2	1	1	0	0	0	0	0	1	1	$1 \rightarrow 1$
3	0	1	1	0	1	0	1	0	1	$0 \rightarrow 2$
4	1	1	0	1	0	1	0	0	0	$0 \rightarrow 4$

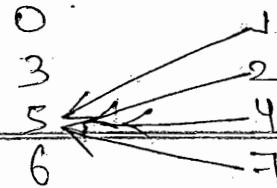
lock out

Self Corrected Counter: - Because of voltage fluctuation, the counter go to unwanted state and if it is coming back to original state after one or more pulse, it is known as self corrected counter if it is not coming back known as lock out condition.

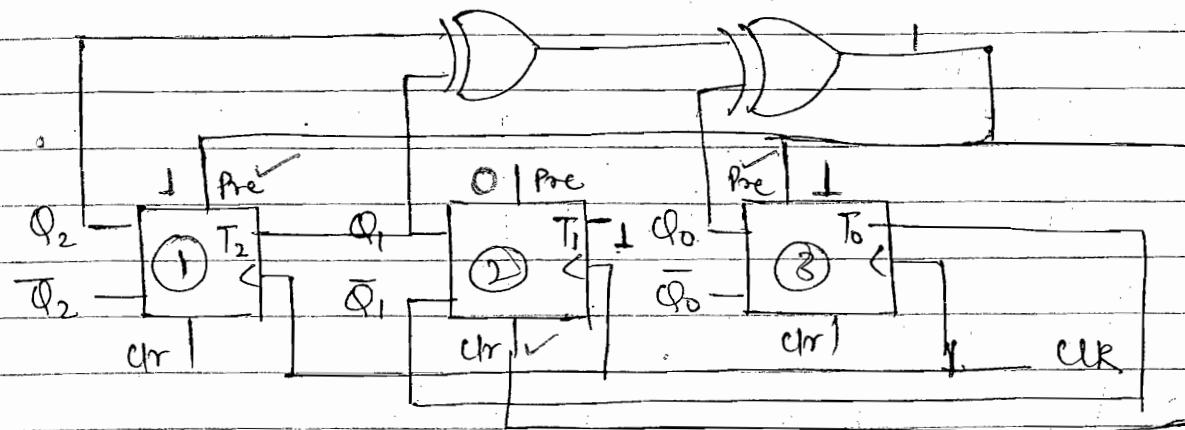
Method of Avoid lock out Condition. Design an additional combinational ckt with the help of unwanted state whose opp. should be connected to desire preset and clear S/p. as per the required wanted state.



Wanted      Unwanted



Suppose want to come back original state (5) after any unwanted state.



5  $\rightarrow$  1 0 1      Connect 1 with preset value and connect 0 " clear value.

Suppose 3  $\rightarrow$  0 1 1      1  $\rightarrow$  clear  
 2, 3  $\rightarrow$  preset.

25/9/2014

### • State Reduction Method :-

State Diagram :- Graphical representation of a ckt performance consist of a parameters present state, S/p, Next state, O/p.

Step 1:- Draw the state diagram for the given ckt

Step 2:- Draw the state table for the state diagram.

Step 3:- Identify the equivalent state

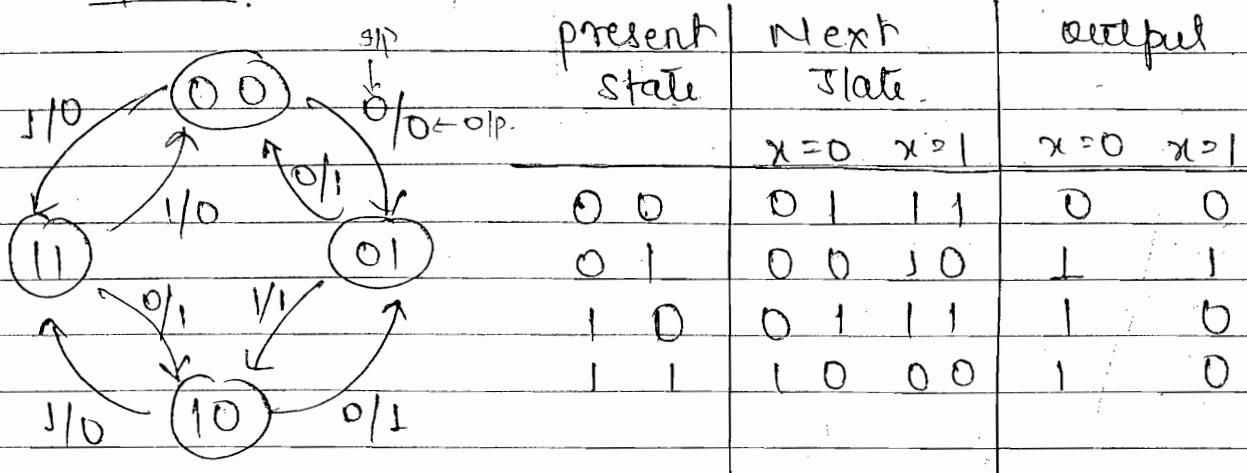
Equivalent state:- Two state are said to be equivalent if their Next state and O/p are equal.

Step 4:- If two state are equivalent one of the state should be eliminated

Steps: Repeat the above process until we get all different state.

Step :-

Given



$x/y$   $x \rightarrow \text{SIP}$   $00 \rightarrow \text{SIP}$  applied get. 01 which gives  
 $y \rightarrow \text{O/P}$   $\text{O/P} \rightarrow 0$ .

Pg No. 62

Present state	Next state	Output
a	c	0 0
b	d	0 0
c	ge	1 1
d	f	1 0
e	f'd	0 1
f	g	1 0 x
g	h	0 1 x

① equivalent state  
 ② state to eliminate  
 ③ equivalent state  
 ④ state to eliminate

Ans. 5 state

eliminate.

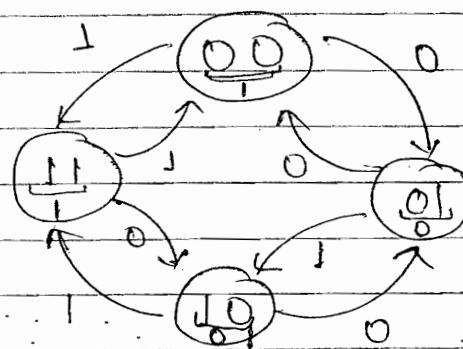
- Types
  - (1) Meile Method
  - (2) Moore Method

→ Present O/p depends upon present S/p and present state.

Mile Method: - The present O/p not only depends on the present state it also depends upon the present S/p.

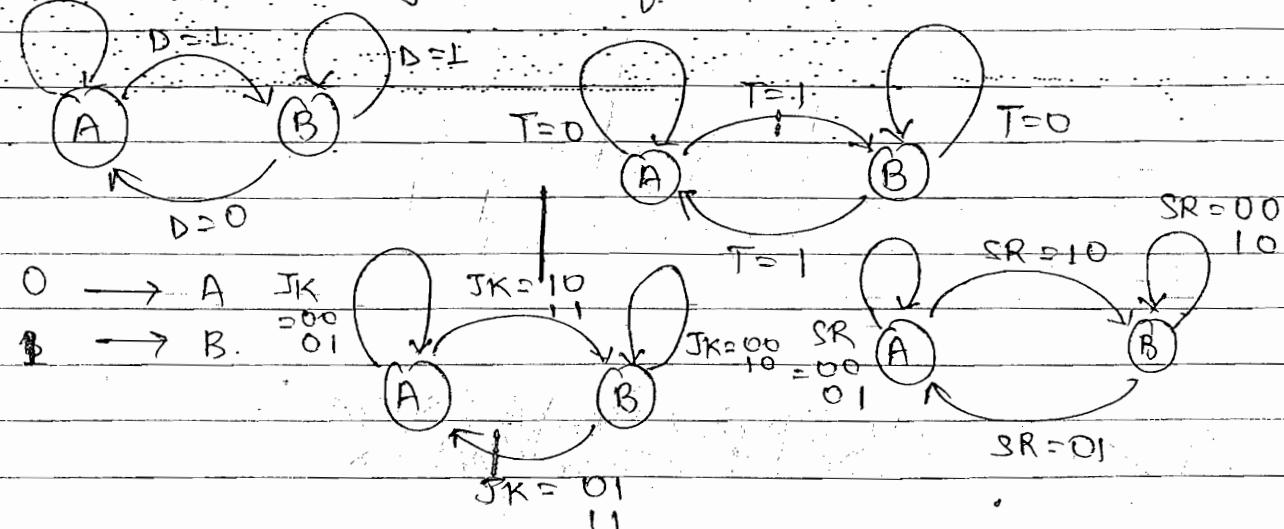
NOTE: - The above example is Mile Method only.

Moore Method: - In this method O/p The present O/p depends on only present state.



whatever will be the S/p, 0, 1  
O/p will depend only present state.

State Diagram of flip-flop:-



NAND

0	0	1
0	1	1
1	0	1
1	1	0

~~AB=00~~

AB=11

AB=00

AB=01

AB=10

AB=11

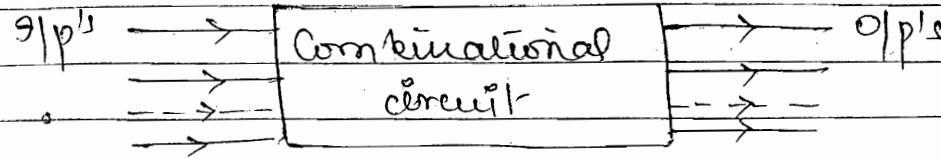
AB=00

AB=01

AB=10

AB=11

# • Combinational Circuit

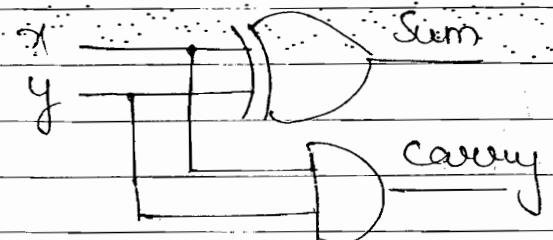


## Half Adder

x	y	Carry	Sum
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

$$\begin{aligned} \text{Sum} &= \bar{x}y + x\bar{y} \\ &= x \oplus y \end{aligned}$$

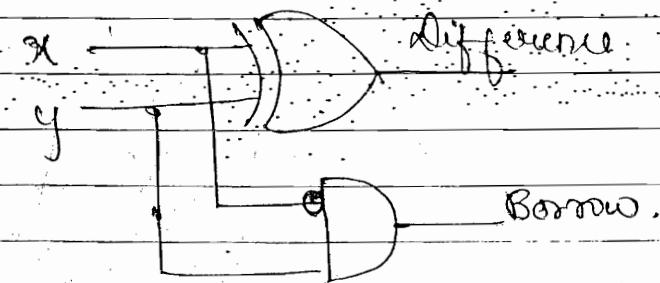
$$\text{Carry} = x \cdot y$$



## Half Subtractor

x	y	Borrow	Diff.
0	0	0	0
0	1	1	1
1	0	0	1
1	1	0	0

$$\begin{aligned} \text{Difference} &= x \oplus y \\ \text{Borrow} &= \bar{x} \cdot y \end{aligned}$$



## Full Adder

x	y	z	Carry	Sum
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

## Full Subtractor

x	y	z	Borrow	Difference
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	0
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

NewtonDesk.com

Sum and difference have same eq  $= x \oplus y \oplus z$   
Carry and Borrow having also same eq. carry  $= x_1 y_1 + x_2 y_2 + y_3$   
only Borrow having  $\bar{x}$  Borrow  $= \bar{x}_1 y_1 + \bar{x}_2 y_2 + y_3$

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

$$= x \oplus y \oplus z$$

$$\text{Difference} = x \oplus y \oplus z$$

K-Map. of Both sum & Diff.

$$\bar{y} \bar{z} \quad \bar{y} z \quad y \bar{z} \quad y z$$

$\bar{x}$	0	1	3	2
$x$	1	0	2	3

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

$$= x y + x z + y z$$

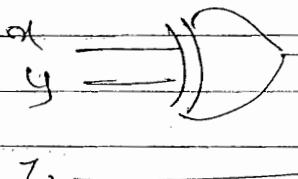
$\bar{x}$	0	1	1	1
$x$	1	0	1	1

Carry

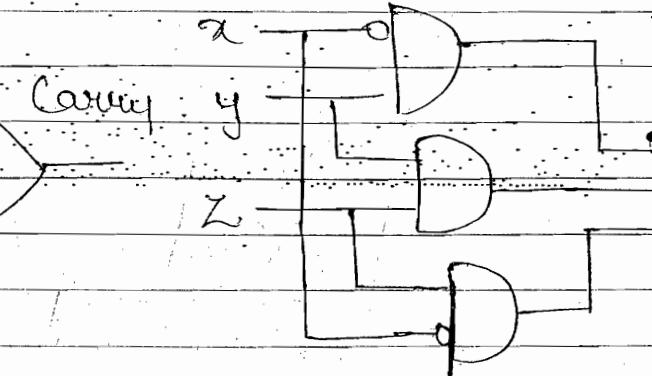
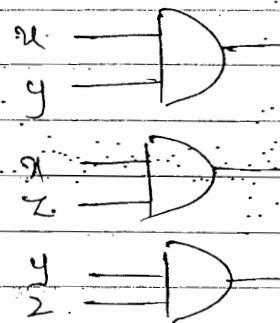
$\bar{x}$	0	1	1	1
$x$	1	0	1	1

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

Borrow



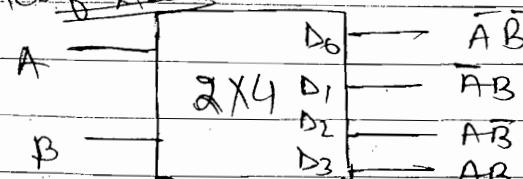
Sum OR Difference



• Decoders :- these are used in memory system.

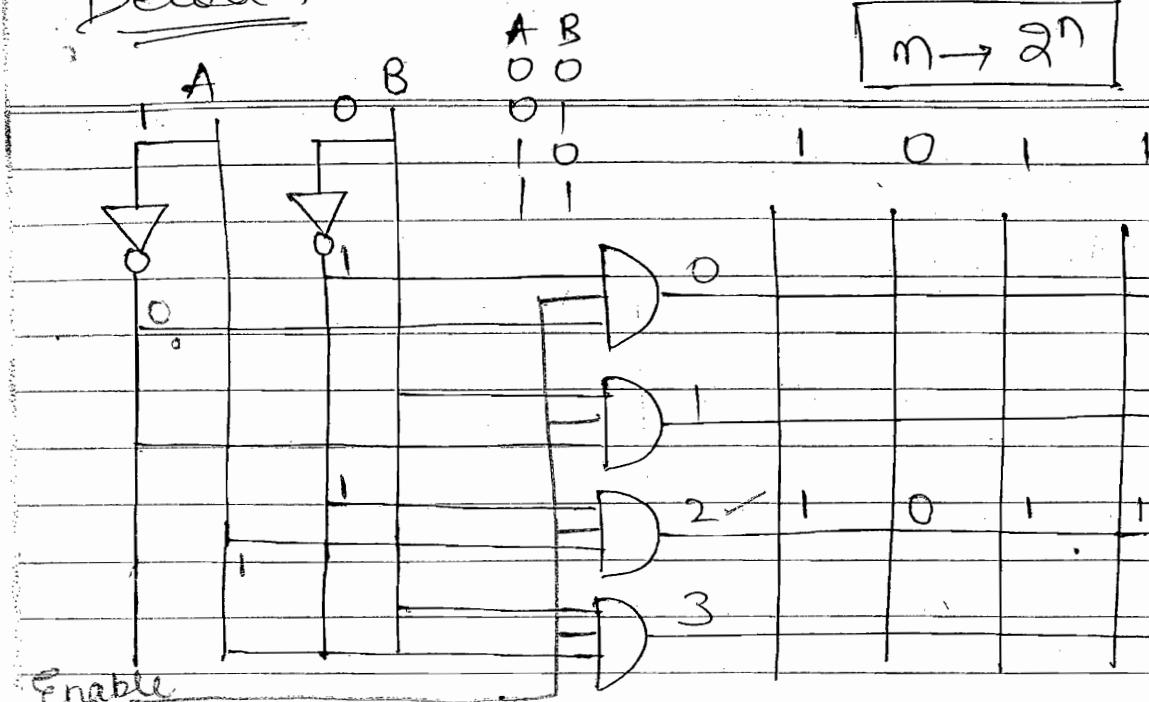
For  $n$ -select lines  $2^n$  location activated

Half Adder



	A	B	Sum	Carry
0	0	0	0	0
1	0	1	1	0
2	1	0	1	0
3	1	1	0	1

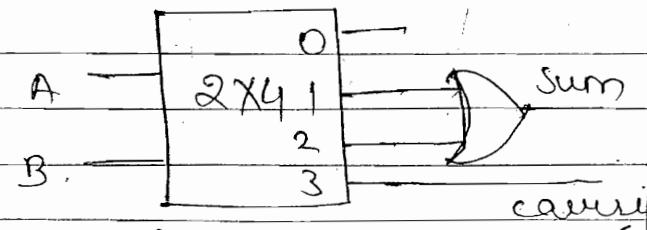
Deedee



Hoa

$$\text{Sum} = \sum m(j, 2)$$

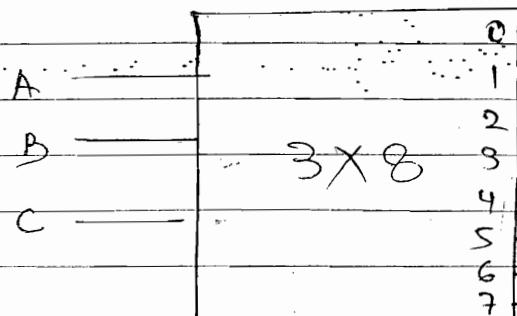
Carry = 0 m(3)



## full Adder

$$\text{Sum} = \Sigma m (1, 2, 4, 7)$$

Carry:  $\Sigma m (3, 5, 6, 9)$



Sum  $\rightarrow \Sigma m\{1, 2, 4, 7\}$

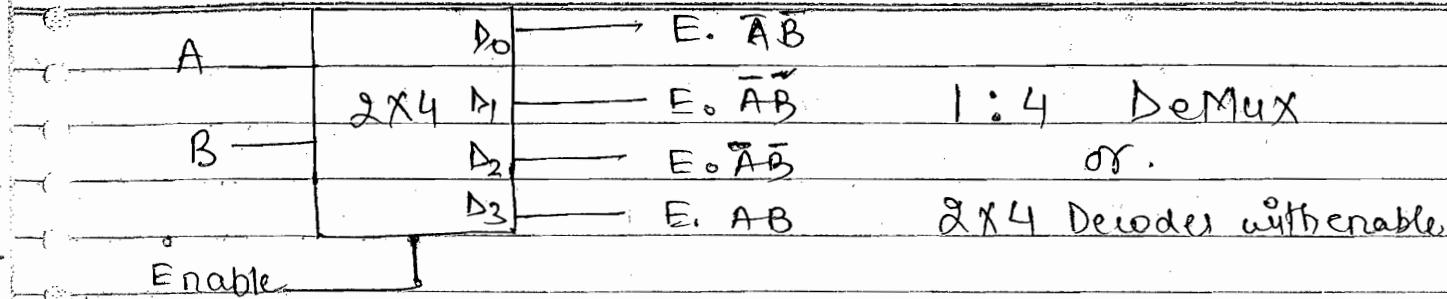
Carry =  $\text{Em}\{3,5,6,7\}$

De Mux :- Decoder with enable pin is DEMUX  
It is 1 to Many circuit.

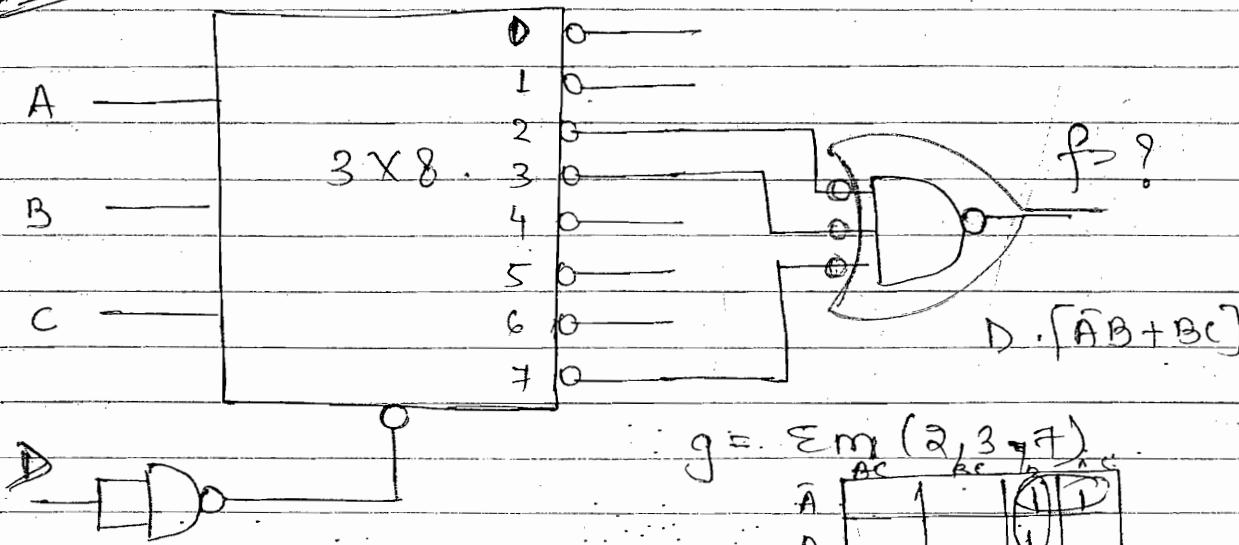
called as

✓ Decoder with Enable = Demux

- If Enable (1) activated then only O/p comes
  - If Enable (0) deactivated O/p will always be 0

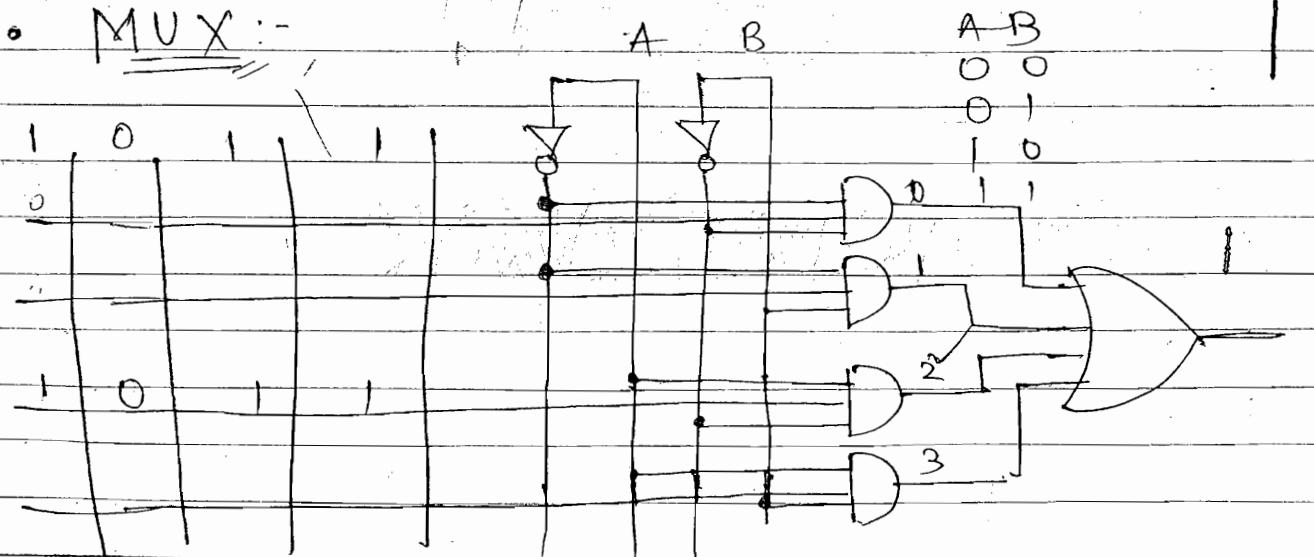


~~Q TIES~~



- Demux or Decoder MUST have OR & NOR gate only otherwise change into OR & NOR ex. NAND given with equivalent to bubbled NOR.

MUX :-



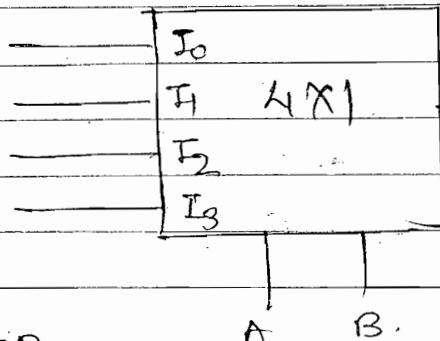
Names

→ Many to 1 circuit, waveform generator or parallel to serial converter, Data Selector

• Enables can also be used if  $E=1$  Output  
 $E=0$  Output is 0.

• If given like that:

already find  $I_o$  and write with  $\bar{A} \bar{B}$



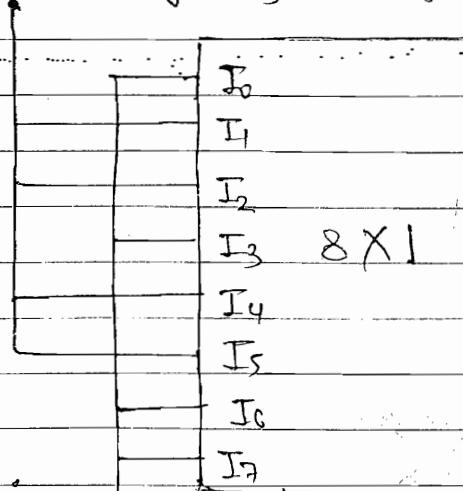
$$f = (I_0) \bar{A} \bar{B} + (I_1) \bar{A} B +$$

$$(I_2) A \bar{B} + (I_3) A B.$$

~~Jo, Vo, gmp~~

• Designing Bif. MUX

Vcc [logic 1]  $f = \sum m(1, 2, 4, 5)$



$$f = \sum m(1, 2, 4, 5)$$

Maximum. Value  $f$ . (3 bit)

$f = 8 \times 1$

	A	B	C
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

Ground [logic 0]

Design By using  $4 \times 1$  MOX1 -  $f = \sum m(1, 2, 4, 5)$

	A	B	C	f
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	1
5	1	0	1	1
6	1	1	0	0
7	1	1	1	0

MSB (missing)

A used in Input

Writing  $\oplus$ p f in form  
of A

LSB (missing)

C used in Input

Writing  $\oplus$ p f in form  
of C

A	$J_0$
1	$J_1$
$\bar{A}$	$J_2$
0	$J_3$

$$f = \sum m(1, 2, 4, 5)$$

C	$J_0$
$\bar{C}$	$J_1$
1	$J_2$
0	$J_3$

$$f = \sum m(1, 2, 4, 5)$$

B C

A B

Shortcut

लाला लाला

011

011

	$J_0$	$J_1$	$J_2$	$J_3$
$\bar{A}$	0	1	2	3
A	4	5	6	7
C	A	1	$\bar{A}$	0

	$J_0$	$J_1$	$J_2$	$J_3$
$\bar{C}$	0	2	4	6
C	1	3	5	7
C	$\bar{C}$	1	0	

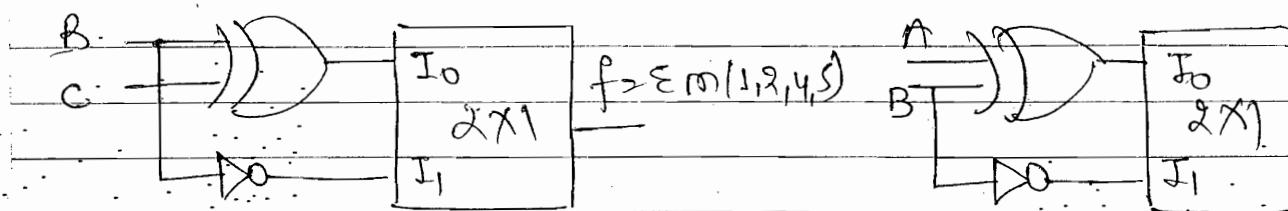
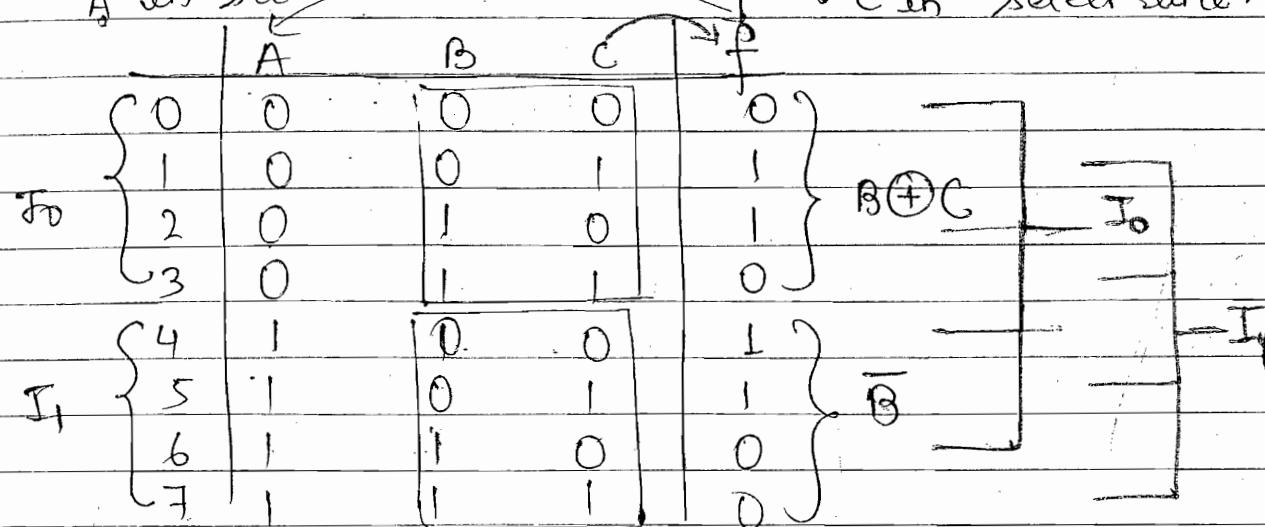
By 2x1 MUX

- MSB is Select line

A is Selection

- LSB is Select line.

C is Selection



Show why

	f0	I1			f0	I1
$\bar{B}C$	0	4			$\bar{A}\bar{B}$	0
$\bar{B}C$	1	5			$\bar{A}B$	2
$B\bar{C}$	2	6			$A\bar{B}$	4
$BC$	3	7			$AB$	6
$\bar{B}C + B\bar{C}$	$\bar{B}C + \bar{B}C$				$A(\bar{B})B$	$\bar{A}\bar{B} + \bar{B}B$
$B(\bar{B})C$	$\bar{B}$					$= \bar{B}$

$\begin{smallmatrix} \text{C1} \\ \text{C1} \\ \text{C1} \\ \text{C1} \end{smallmatrix}$

$\begin{smallmatrix} \text{C1} \\ \text{C1} \end{smallmatrix}$

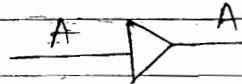
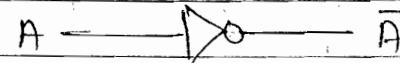
~~TOPS~~

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

$$f = ? \quad f = \Sigma m (1, 2, 4, 6)$$

Solution

	$I_0$	$I_1$	$I_2$	$I_3$	write this column then circle the No. and write the function.
$\bar{A}$	0	1	2	3	
A	4	5	6	7	
	A	$\bar{A}$	1	0	

BufferNOT

0	$I_0 \otimes I_1$	$f = \bar{A}(I_0) + A(I_1)$
1	$I_1$	$= \bar{A}(0) + A(1)$
		$= A$

1	$I_0 \otimes I_1$	$f = \bar{A}(I_0) + A(I_1)$
0	$I_1$	$= \bar{A}(1) + A(0)$
		$= \bar{A}$

AND

	A	B	f
0	0	0	0
1	0	1	0
2	1	0	0
3	1	1	1

0	$I_0$	$I_1$	$I_2$	$I_3$	$f = \bar{B}(I_0) + B(I_1)$
1	$I_1$				$= \bar{B}(0) + BA$
2		$I_2$			$= AB$
3			$I_3$		

$\bar{A}$	0	1	2	3	0	1	2	3	0	1	2	3
A	1	0	1	0	1	0	1	0	1	0	1	0

EXOR

	A	B	f
0	0	0	0
1	0	1	1
2	1	0	1
3	1	1	0

$\bar{A}$	0	1	2	3	$\bar{A}$	0	1	2	3	$\bar{A}$	0	1	2	3
A	1	0	1	0	A	1	0	1	0	A	1	0	1	0

This gate is p'p'p'p' from a NOT.

Sheetlet

2x1 MUX

NOI	1
AND	1
OR	1
EX-OR	2
EX-NOR	2
NAND	2
NOR	2

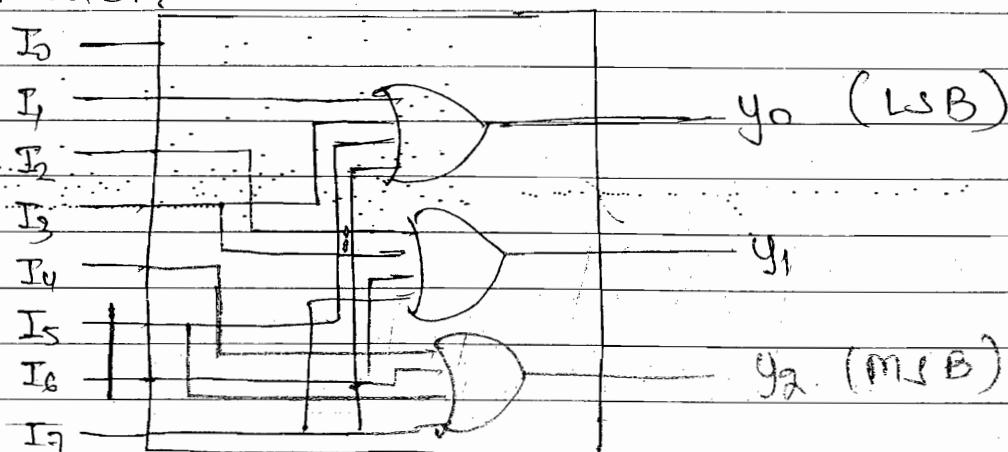
Q H.o.A. No. of MUX.

Sum  $\rightarrow$  EX-OR  $\rightarrow$  2Carry  $\rightarrow$  AND  $\rightarrow$  1Total 3 Ans

26/9/2014

- Design A circuit :-  $9/10 \rightarrow$  generate corresponding code will generate in 0/1.

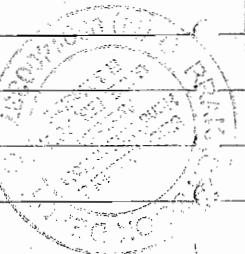
- Encoder :-



8 X 3 Encoder.

Octal to Binary converter.

$I_7$	$I_6$	$I_5$	$I_4$	$I_3$	$I_2$	$I_1$	$I_0$	$y_2$	$y_1$	$y_0$
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	0	1	0	0	1	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	0
0	1	0	0	0	0	0	0	1	0	1
1	0	0	0	0	0	0	0	1	1	1



for Hexg to Binary conversion  $16 \times 4$

$$y_0 = I_1 + I_2 + I_5 + I_7$$

$$y_1 = I_2 + I_3 + I_6 + I_7$$

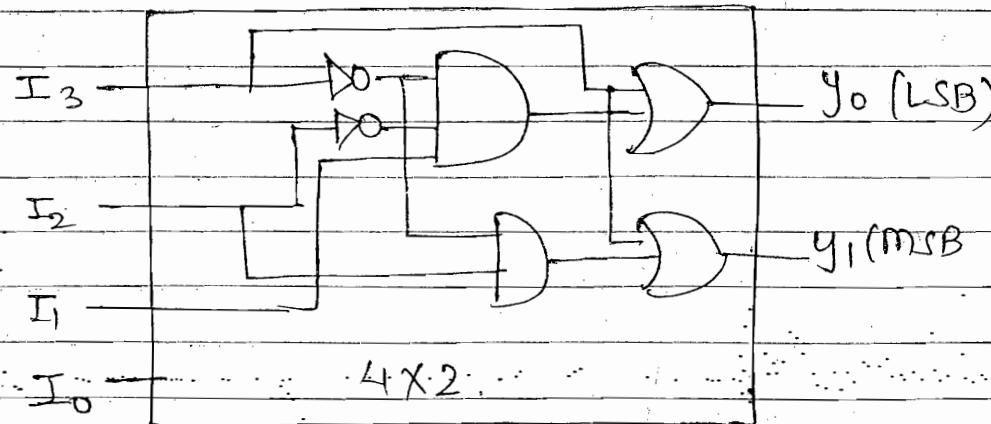
$$y_2 = I_4 + I_5 + I_6 + I_7$$

- Priority Encoder: Code will generate only when its higher no. having 0 value.

$I_3$	$I_5$	$I_4$	$I_6$	$y_1$	$y_0$
0	0	0	1	0	0
0	0	1	X	0	1
0	1	X	X	1	0
1	X	X	X	1	1

$$y_0 = I_1 I_2' I_3' + I_3$$

$$y_1 = I_2 I_3' + I_3$$



code for L  
only generate  
when  $I_2, I_3$   
are 0,  $I_6$   
is whatever  
X.

- Higher Order Circuits By using Lower Order Circuits

$4 \times 16$  concentrate in 01P NO.

By using  $2 \times 4$

Shortest

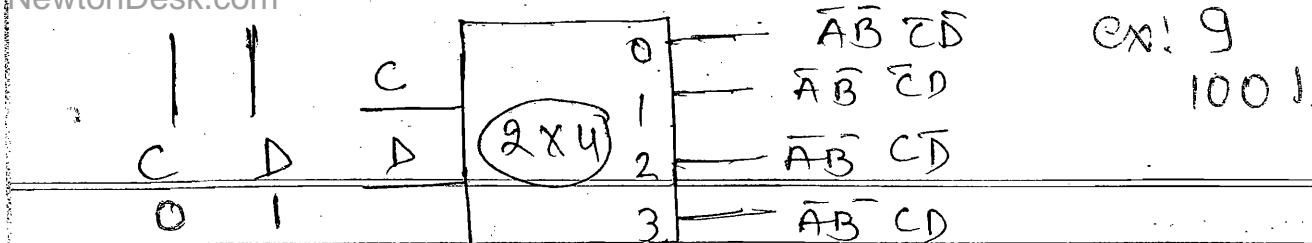
$$\frac{16}{4} = 4$$

$$\frac{4}{4} = 1$$

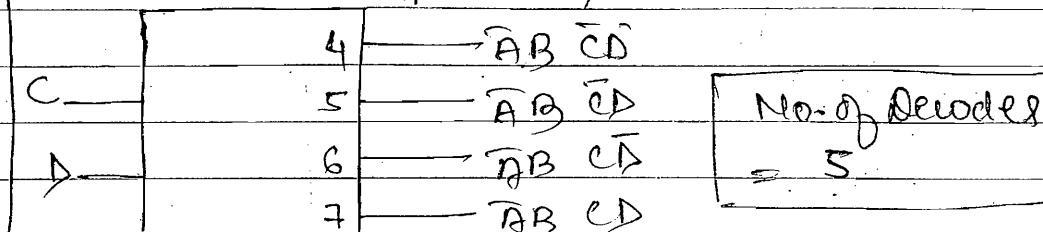
Decade

Total. Required

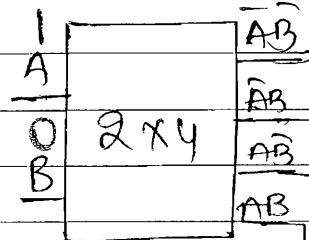
(5)



enable 9IP  $\rightarrow$  DEMUX



No. of Decoders  
= 5

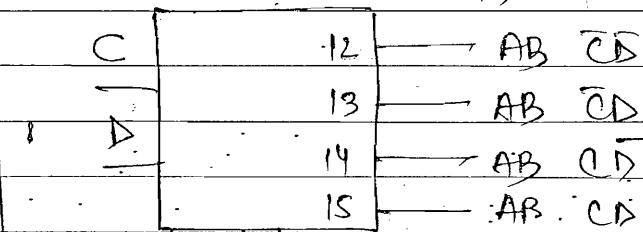


E. 9IP. DEMUX

No. of Enable  
Decoder  
= 4

Q. 30

E 9IP. (Demux)



E 9IP. (Demux)

of even NO.

Ex:-  $16 \times 256$   $256 = 16$   
 $H \times 16$

No external  
ckt required

$$\frac{16}{16} = 1$$

$17$  required.

Ex:-  $3 \times 8$  By using  $2 \times 4$

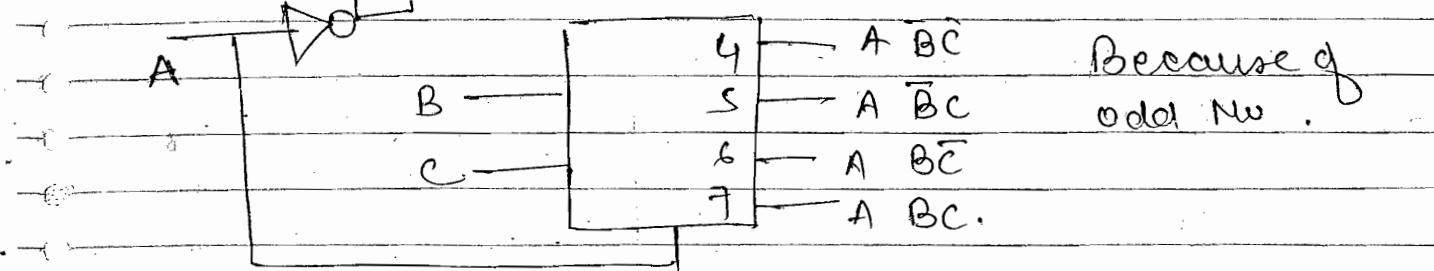
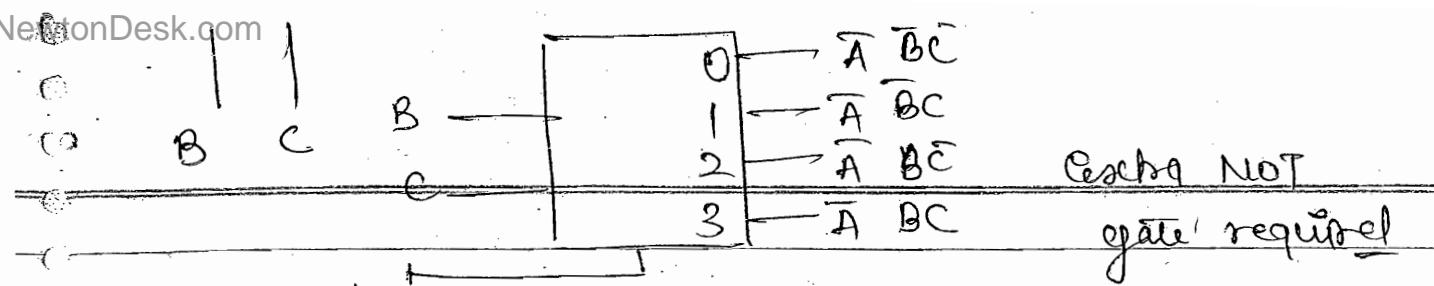
If odd  
NO,

$$\frac{3}{4} = \frac{3}{4}$$

$$\frac{2}{4} = \frac{2}{4}$$

Not possible

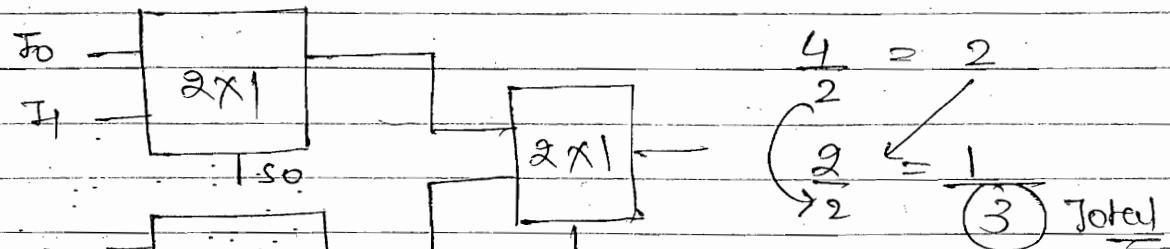
External ckt circuit required "NOT gate".



Total 3ct required. & with NOT gate.

If NOT gate not given in option  $\Rightarrow$  Then 3ct Required

MUX :-  $2 \times 1$  using  $2 \times 1$   $\rightarrow$  Concentrate In I/p No.

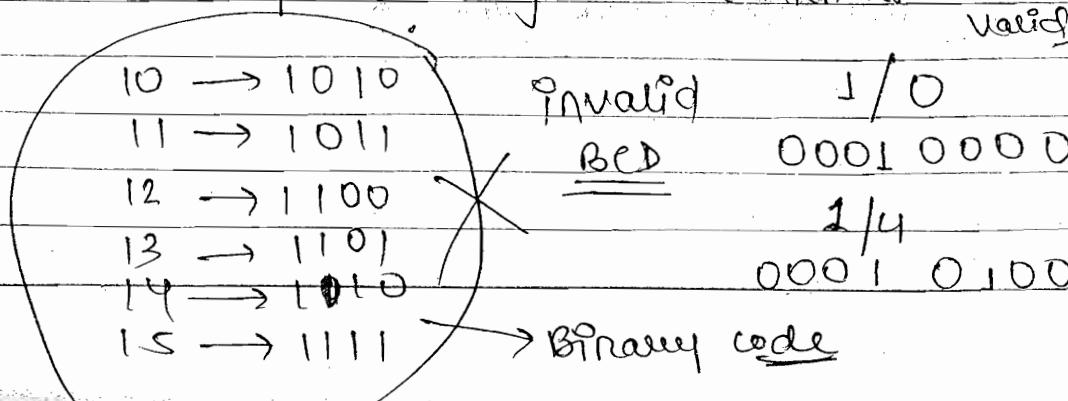


So. will select from both  $(T_0, T_1)$   $(T_2, T_3)$  But by  $S_1$  only 1 will select.

### Codes & Code Converters :-

BCD :- Valid only upto 4 digit, "8 4 2 1"

upto 9. only BCD written same as Binary. Valid



BCD  $\rightarrow$  not a self complimentary code

"2421" Code.

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	1	0	1	1
6	1	1	0	0
7	1	1	0	1
8	1	1	1	0
9	1	1	1	1

Self complimentary  
code

"5421" same as "2421" except that, half MSB is 0 and half LSB is 1

Excess-3 (Ex-3 code)  $\rightarrow$  Addition of 3 in BCD

$$0 \rightarrow 0000 \rightarrow \text{BCD}$$

$$+ \quad \quad \quad \quad$$

$$0011 \rightarrow \text{Ex-3}$$

$$\rightarrow 0001 \rightarrow \text{BCD}$$

$$+ 0011$$

$$0100 \rightarrow \text{Ex-3}$$

BCD to Ex-3 Code Conversion:

	I/p BCD				O/p Ex-3			
	A	B	C	D	w	x	y	z
0	0	0	0	0	0	0	1	1
1	0	0	0	1	0	1	0	0
2	0	0	1	0	0	1	0	1
3	0	0	1	1	0	1	1	0
4	0	1	0	0	0	1	1	1
5	0	1	0	1	1	0	0	0

NewtonDesk.com **EX-OR** → Arithmetic logic gate  
Used in add & subtractor

6	0	1	1	0	1	0	0	1
7	0	1	1	1	1	0	1	0
8	1	0	0	0	1	0	1	1
9	1	0	0	1	1	1	0	0

$\bar{C}D$   $\bar{C}D$   $CD$   $CD$

$\bar{A}B$	0	1	2	3
$\bar{A}B$	4	1s	1s	1c
$AB$	$X_{12}$	$X_{13}$	$X_{15}$	$X_{14}$
$A\bar{B}$	1s	1s	$X_{11}$	$X_{10}$

$\bar{A}D$   $\bar{B}D$   $CD$   $CD$

$\bar{A}B$	0	1	2	3
$\bar{A}B$	4	1s	1s	1c
$AB$	$X_{12}$	$X_{13}$	$X_{15}$	$X_{14}$
$A\bar{B}$	1s	1s	$X_{11}$	$X_{10}$

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

$$x = B\bar{C}\bar{D} + \bar{B}\bar{D} + \bar{B}C$$

$\bar{A}B$	0	1	2	3
$\bar{A}B$	4	1s	1s	1c
$AB$	$X_{12}$	$X_{13}$	$X_{15}$	$X_{14}$
$A\bar{B}$	1s	1s	$X_{11}$	$X_{10}$

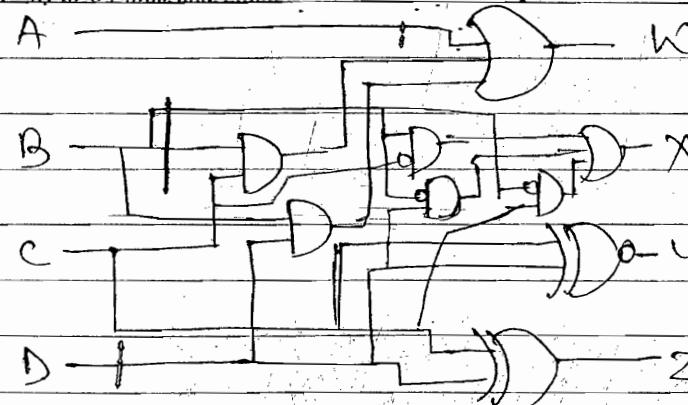
$\bar{A}B$	0	1	2	3
$\bar{A}B$	4	1s	1s	1c
$AB$	$X_{12}$	$X_{13}$	$X_{15}$	$X_{14}$
$A\bar{B}$	1s	1s	$X_{11}$	$X_{10}$

$$y = \bar{C}D + CD$$

$$z = \bar{C}D + \bar{C}D$$

9/p. B. o. CD

0/p. EX-3



• Binary To Gray :- msB will not change.

$B_3 \ B_2 \ B_1 \ B_0$

Binary  $1 \oplus 1 \oplus 0 \oplus 1$

$\downarrow \quad \downarrow \quad \downarrow \quad \downarrow$

Gray  $\Rightarrow 1_{q_3} \ 0_{q_2} \ 1_{q_1} \ 1_{q_0} = 1011$

msB.

• Binary      Gray

$$B_3 \longrightarrow G_3$$

$$G_3 = B_3$$

$$B_2 \longrightarrow G_2$$

$$G_2 = B_3 \oplus B_2$$

$$B_1 \longrightarrow G_1$$

$$G_1 = B_2 \oplus B_1$$

$$B_0 \longrightarrow G_0$$

$$G_0 = B_1 \oplus B_0$$

• Gray To Binary Code :-

$$\begin{array}{cccc} G_3 & G_2 & G_1 & G_0 \\ \text{Gray} \Rightarrow & 1 & 0 & 1 \\ & \downarrow & \downarrow & \downarrow \\ \text{Binary} \Rightarrow & 1 & 1 & 0 \\ & \text{MSB } (B_3) & B_2 & B_1 & B_0 \end{array}$$

Gray code :-

- ✓ unit distance code.
- ✓ Hamming distance.

Gray      Binary

$$G_3 \longrightarrow B_3 \quad B_3 = G_3$$

$$G_2 \longrightarrow B_2 \quad B_2 = B_3 \oplus G_2$$

$$G_1 \longrightarrow B_1 \quad B_1 = B_2 \oplus G_1$$

$$G_0 \longrightarrow B_0 \quad B_0 = B_1 \oplus G_0$$

Binary Code      Gray Code

0	0      0	0      0
1	0      1	0      1
2	1      0	1      1
3	1      1	1      0

Gray Code

Hamming Distance is '1' in Gray Code,

⇒ Gray code is "unit distance code"

⇒ Gray → K-Maps.

• B.C.D To Seven Segment Code Conversion:-

		I/P.				O/P.						
		A	B	C	D	a	b	c	d	e	f	g
$f \overline{I} \overline{b}$	$\overline{g}$	0	0	0	0	1	1	1	1	1	1	0
$e \overline{I} \overline{c}$	0	0	0	0	1	0	1	1	0	0	0	0
$d$	1	0	0	0	1	0	1	1	0	0	0	0
2	0	0	1	0	1	1	1	0	1	0	1	1
3	0	0	1	1	1	1	1	1	1	0	0	1
4	0	1	0	0	0	1	1	1	0	0	1	1
5	0	1	0	1	1	0	1	1	0	1	1	1
6	0	1	1	0	0	0	1	1	1	1	1	1
7	0	1	1	1	1	1	1	1	0	0	0	0
8	1	0	0	0	1	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	1	0	0	1	1

Draw K-map. A B C D  $\rightarrow$  I/P.

(a, b, c, d, e, f, g  $\rightarrow$  O/P. individually).

And write the eqn.

• Number Representation:-

Binary  $\rightarrow$  4 digit 8 4 2 1

5 digit 16 8 4 2 1

6 digit 32 16 8 4 2 1

1) Sign Magnitude

2) 1's Complement

3) 2's Complement

Case (i) +ve Number (sign, 1's, 2's complement)

+51  $\rightarrow$  0 1 0 0 1 1  
 ↓ ← →  
 sign bit magnitude

same

Case (ii) -ve Number ex: -51

$\begin{array}{|c|c|c|c|c|c|c|} \hline 1 & 1 & 1 & 0 & 0 & 1 & 1 \\ \hline \end{array} \rightarrow \text{sign magnitude}$   
 sign bit magnitude

$\begin{array}{|c|c|c|c|c|c|c|} \hline 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ \hline \end{array} \rightarrow 1\text{'s complement}$

$0 \ 0 \ 1 \ 1 \ 0 \ 0$

$+ \ 1$

$\begin{array}{|c|c|c|c|c|c|c|} \hline 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ \hline \end{array} \rightarrow 2\text{'s complement}$

Range for sign magnitude and 1's complement.

$$-(2^{n-1} - 1) \text{ to } + (2^{n-1} - 1)$$

ex: -  $n = 3$   $n = \text{No. of Sign Bit}$

$$-(2^{3-1} - 1) \text{ to } + (2^{3-1} - 1)$$

$$-3 \text{ to } +3$$

	Sign	Magnitude	1's Complement	2's Complement					
	x	y	z	x	y	z	x	y	z
	0	0	0	0	0	0	0	0	0
	0	0	1	0	0	1	1	0	1
(+ve)	0	1	0	1	0	1	0	1	0
Sign	1	0	1	1	0	1	1	1	1
Bit value	1	0	0	-0	1	1	-0	1	0
Sign	1	0	1	-1	1	1	-1	1	1
Bit value	1	1	0	-2	1	0	-2	1	1
Sign	1	1	1	-3	1	0	-3	1	0

(-ve) Sign Bit

Range for 2's complement

$$-(2^{n-1}) \text{ to } + (2^{n-1} - 1)$$

Changes comes only in -ve no

Special case for 2's complement - Carry enters into sign bit.

Ex: ① 1 0 0  $\Rightarrow -(2^{n-1})$  1 0 0  
 $\Rightarrow -4$  1 1 1's comp.

② 1 0 0 0  $\Rightarrow -8$  1 0 0 2's comp.

③ 1 0 0 0 0  $\Rightarrow -16$ .

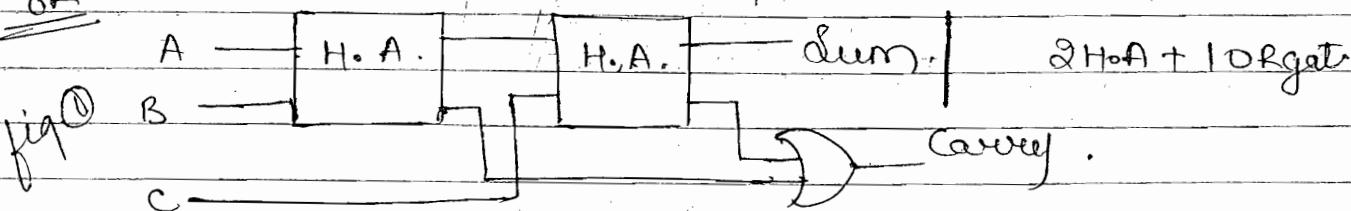
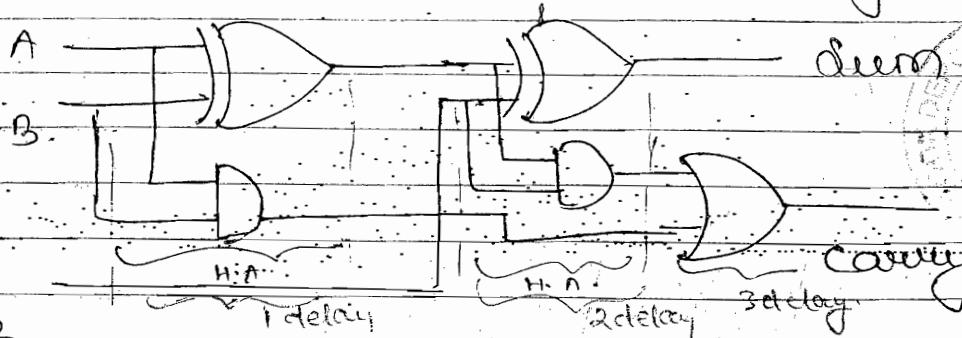
NO representation for -0 in 2's complement

Ex:

	True	Sign. Mg.	1's comp.	2's comp.
① 1 0 1	+5	+5	+5	+5
② 1 1 0 1	-13	-2	-3	-3

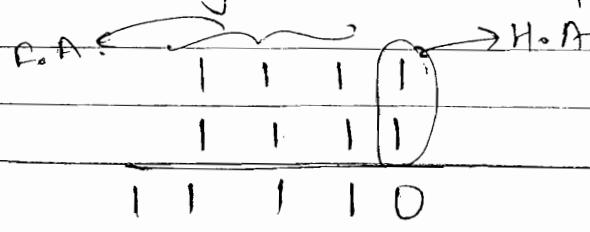
28/9/2012

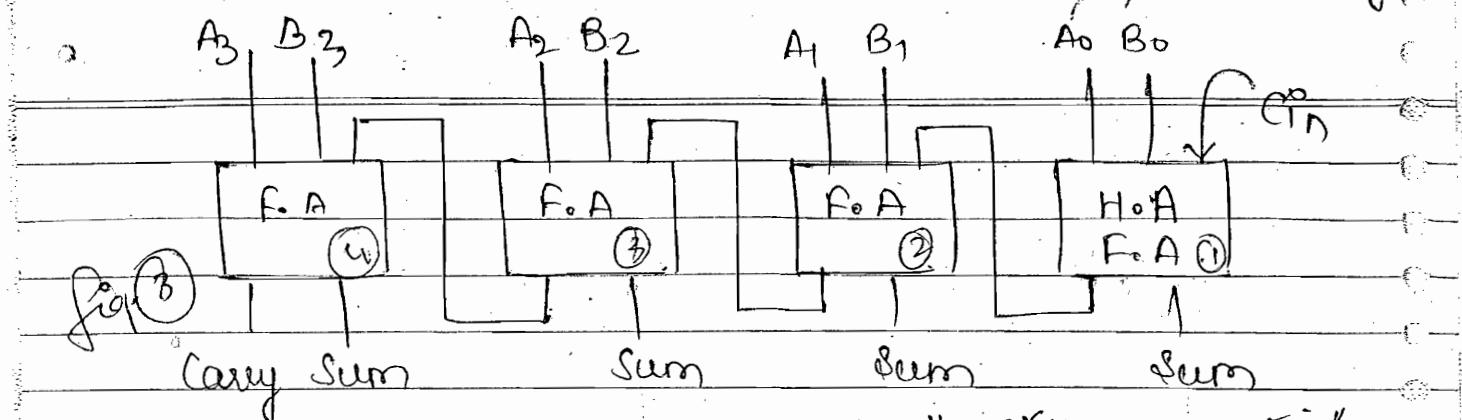
• Full Adder & Parallel Carry Adder



$\Rightarrow A \rightarrow A_3 A_2 A_1 A_0$   $B \rightarrow B_3 B_2 B_1 B_0$

• Ripple Carry Adder OR Parallel Carry Adder





①  $(N-1) F.o.A + 1 H.o.A.$

If  $C_{in}$  present :  $\checkmark N F.o.A.$   
(added with other ck)

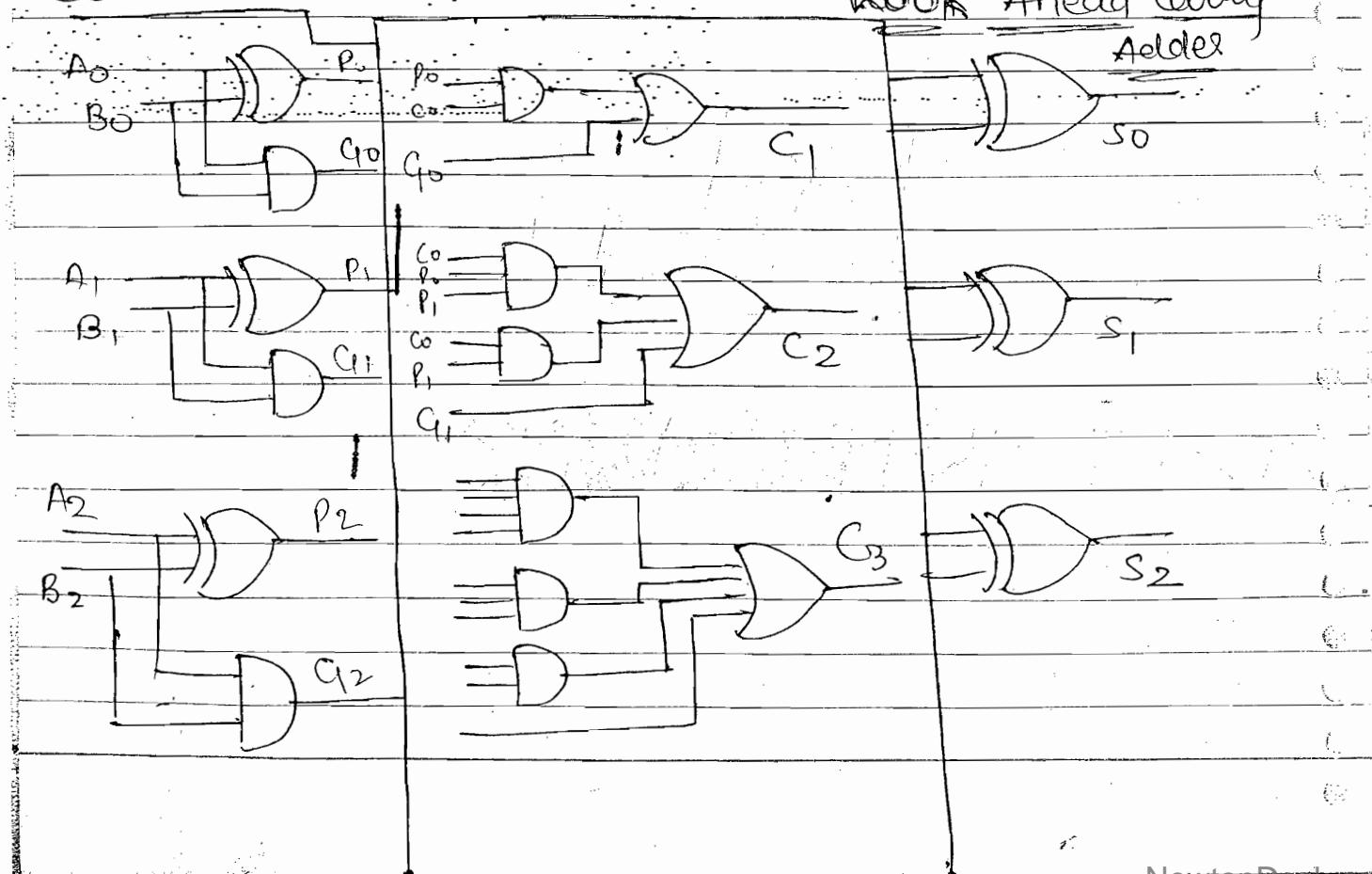
~~also with~~  $(N-1) [2H.o.A + 1 \text{ OR gate}] + 1 H.o.A$ . from fig ①

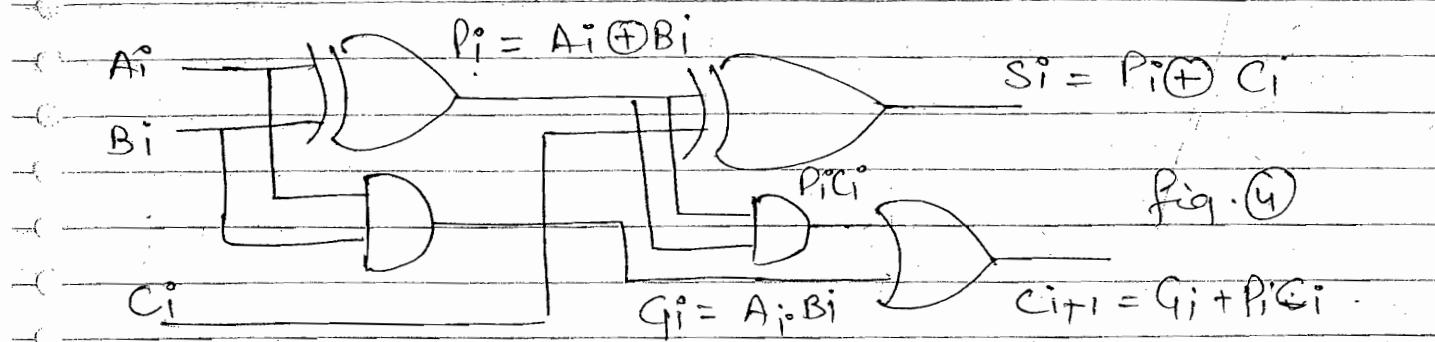
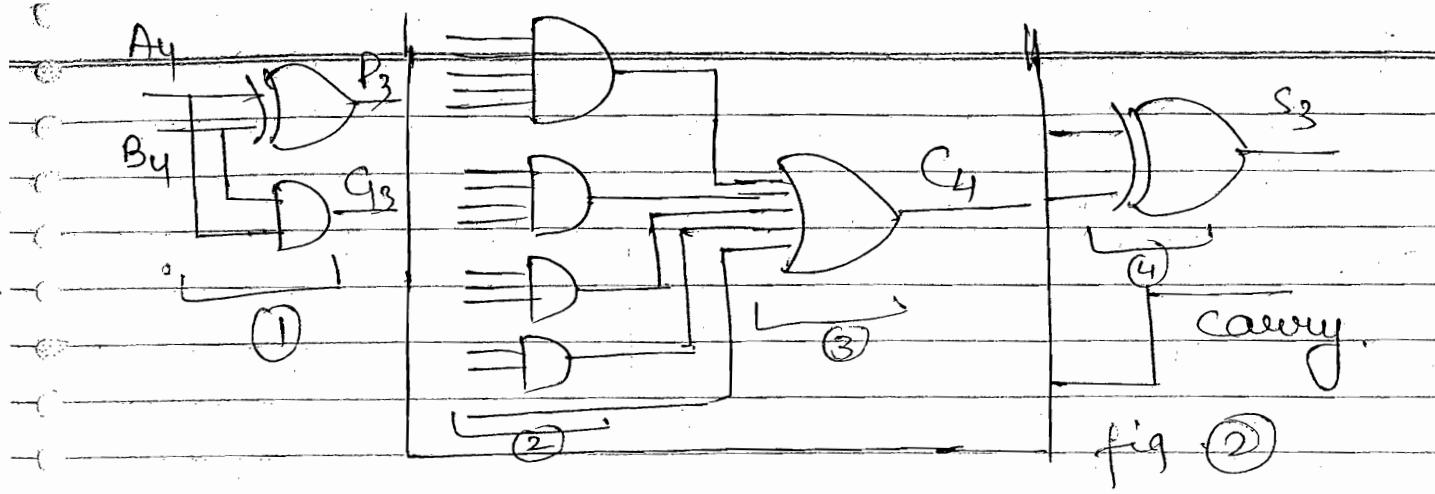
$$\Rightarrow (2N-1) H.o.A. + (N-1) \text{ OR gate}$$

Ex: N = 4.

$\checkmark H.o.A + 3 \text{ OR gate}$

~~hook~~ Ahead Carry Adder





$$C_0 + 1 = C_1 = C_0 + P_0 C_0$$

$$C_1 + 1 = C_2 = C_1 + P_1 C_1 = C_1 + P_1 [C_0 + P_0 C_0] = C_1 + P_1 C_0 + P_1 P_0 C_0$$

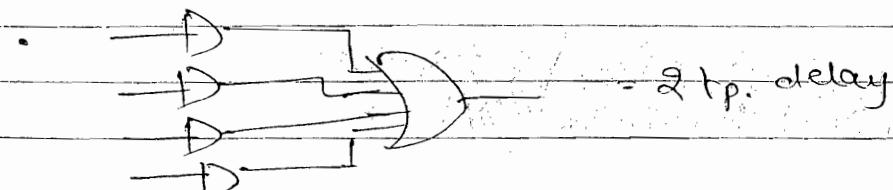
$$C_2 + 1 = C_3 = C_2 + P_2 C_2 = C_2 + P_2 [C_1 + P_1 C_0 + P_1 P_0 C_0]$$

$$= C_2 + P_2 C_1 + P_1 P_2 C_0 + P_1 P_0 C_0 P_2$$

$$C_3 + 1 = C_4 = C_3 + P_3 C_3 = C_3 + P_3 [C_2 + P_2 C_1 + P_1 P_2 C_0 + P_1 P_0 C_0]$$

$$= C_3 + P_3 C_2 + P_3 P_2 C_1 + P_3 P_2 P_1 C_0 + P_3 P_2 P_1 P_0 C_0$$

•  $\rightarrow D - D - D - D - D$  = 5tp. delay.



• fig. ③ Has 3 delay. But all three F.o.D. having 2 delay. A<sub>i</sub>, B<sub>i</sub> are already operated in case of F.o.A.

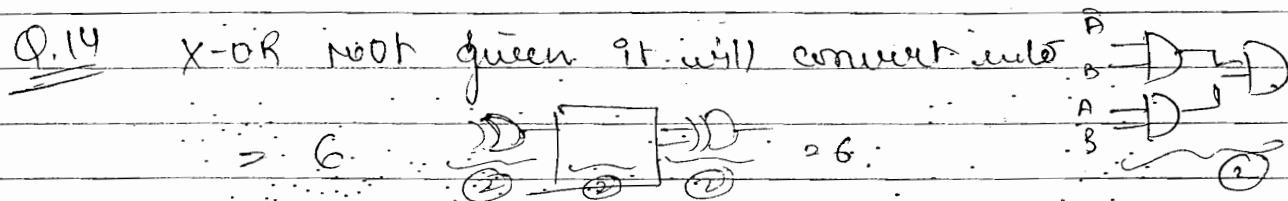
So total Time delay  $\Rightarrow 3 + 2 + 2 + 2 = 9$  bit delay.

But in fig ② only 4 bit delay  $\textcircled{1} \textcircled{2} \textcircled{3} \textcircled{4}$

$\approx$  ex: fig ③ for 10 bit  $= 2 \times 20 \text{ (F.A)} = 20$   
 $= 20 + 1 = 21$  bit delay.  
 H.A.

$\approx$  But in fig ② only vertical CLK↑, horizontal remain same even in 10 bit  $\rightarrow$  4 bit delay.

$\approx$  CLK are individually having carry, all CLK carry individually and add finally by OR gate.



Q.15 only Box. Total OR gate  $\Rightarrow 4$  and AND = 10 gate

Shortcut

$$\text{AND gate} = \frac{n(n+1)}{2}$$

$$\text{OR Gate} = n$$

• Complement Methods -

Dec'	Bin'	Oct'	Hexa'
7's $\rightarrow$	10's	2's	8's

$(r-1)$ 's $\rightarrow$	9's	1's	7's	15's
--------------------------	-----	-----	-----	------

$r \rightarrow$  Radix (or) Base

shortalt for 2's complement:

ex:-  $\begin{array}{r} 10100 \\ \swarrow \\ 01100 \end{array}$  2's.

comes upto Non-zero, write same as it and remaining we get inverted

ex:-  $\begin{array}{r} 10101 \\ \swarrow \\ 01011 \end{array}$  2's.

ex:-  $\begin{array}{r} 1000 \\ \swarrow \\ 1000 \end{array}$  2's.

ex:- 9's complement.

$$\begin{array}{r} 1) 9 \\ 9's \\ \hline \end{array}$$

$$\begin{array}{r} 2) 99 \\ 17 \\ \hline 82 \end{array}$$

$$\begin{array}{r} 3) 99.9 \\ 17.3 \\ \hline 82.6 \end{array}$$

ex:- 10's complement.

$$\begin{array}{r} 1) 2 \\ +1 \\ \hline 3 \end{array}$$

$$\begin{array}{r} 2) 82 \\ +1 \\ \hline 83 \end{array}$$

$$\begin{array}{r} 3) 82.6 \\ +1 \\ \hline 82.7 \end{array}$$

{ 1. must be to  
LSB added

2's Complement Adder/ Subtractor

$B_3 | B_2 | B_1 | B_0$        $A_3 | A_2 | A_1 | A_0$

4 Bit parallel

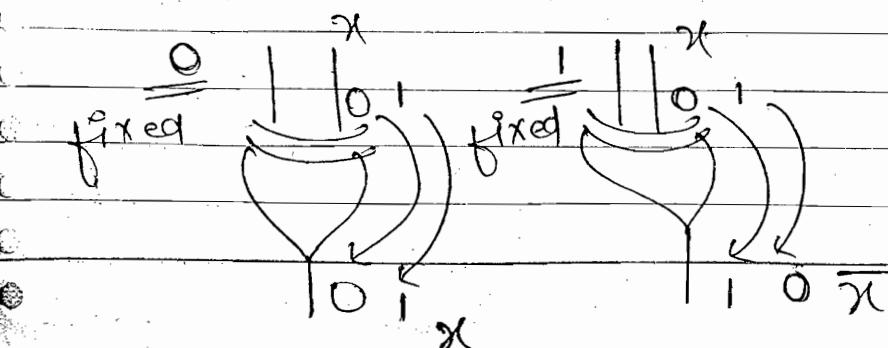
Binary Adder  $\leftarrow$  Cin

carry out

Sum

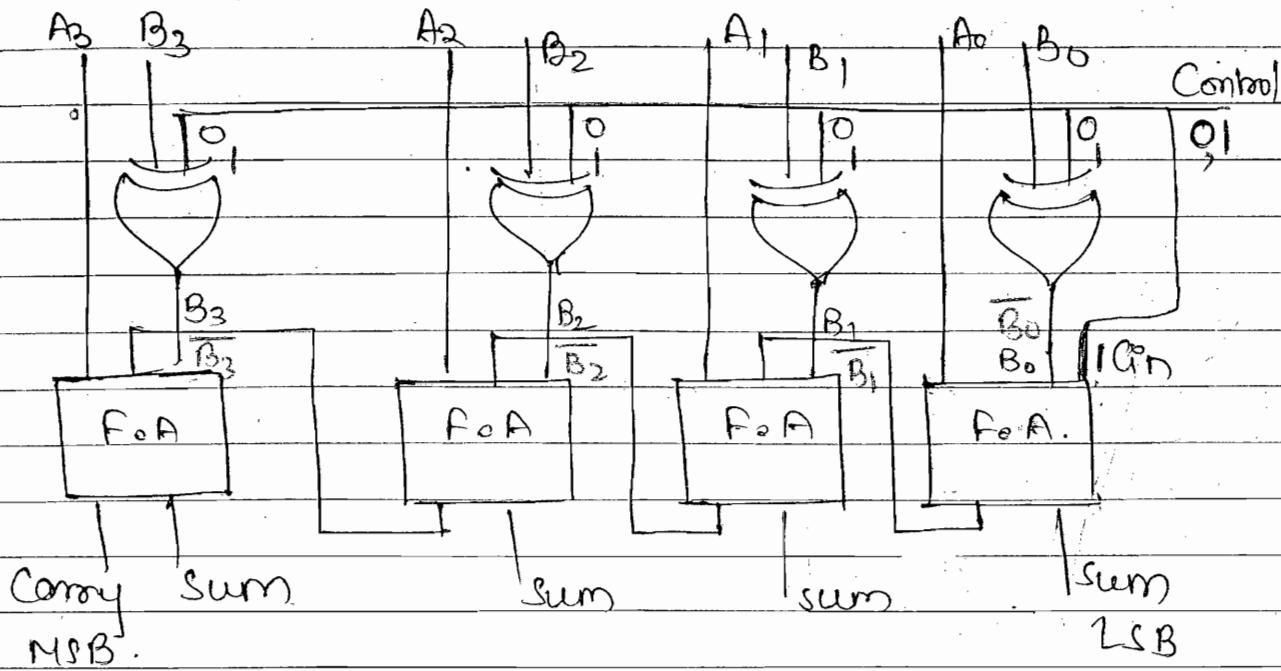
$A \rightarrow A_3 A_2 A_1 A_0$

$B \rightarrow B_3 B_2 B_1 B_0$



$x \leftarrow y$   
 $x \rightarrow \text{minuend}$   
 $y \rightarrow \text{subtrahend}$

$$x - y \rightarrow x + \bar{y}'$$



Control  $\Rightarrow 0 \Rightarrow A + B$

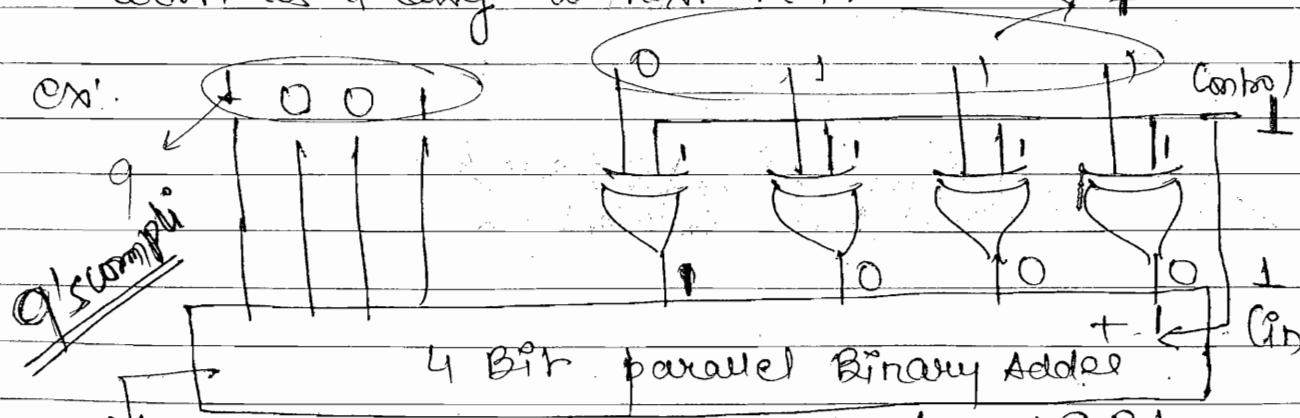
$\Rightarrow 1 \Rightarrow A - B$

$B_0 \rightarrow 1$ 's complement

$+ 1 \rightarrow \bar{y}'$ 's complement

when control = 1,  $B_0 \rightarrow \bar{B}_0$  ( $\bar{y}'$ 's complement) by adding

$Cin = 1$   $B_0 \rightarrow \bar{B}_0$  ( $\bar{y}'$ 's complement) which will added with  $A_0 \rightarrow A + \bar{y}'$ 's complement. This result will work as a carry to next FoA.



Carry:

Ignore:  $\cancel{\frac{1}{2}}$

$$0 \quad 0 \quad 1 \quad 0 + 1001 \quad \underline{0010} \Rightarrow 2$$

sum

Ignore = 2

## • BCD Addition:-

Case (i)  $5 \rightarrow 0101$

$4 \rightarrow 0100$

1001 valid (no carry).

6 Invalid BCD

$10 - 1010$  add 6  
 $11 - 1011$   
 $12 - 1100$   
 $13 - 1101$   
 $14 - 1110$   
 $15 - 1111$

• When two valid BCD No. are added the result in invalid or carry is generated. The 6 should be added to get valid BCD No.

Case (ii)

$5 \rightarrow 0101$

$7 \rightarrow 0111$

1100 invalid

$+ 0110$

10001 / 0010

1 2  $\Rightarrow 12$

(Invalid)

Case (iii)

$8 \rightarrow 1000$

(Valid But carry generate)

$9 \rightarrow 1001$

carry

1 0001  $\rightarrow$  valid

$\downarrow + 110$

0001 / 0111

$\Rightarrow 17$

Ex:-

10111011 10111111

Carry out

4 Bit Adder

8421

11XX

1X1X

Designed

fixed 0

0/1 Bit Adder

10101110

- BCD Subtraction -

- Case(i) +ve Result

ex:  $6 \rightarrow 0110$

$-4 \rightarrow 0101$  [9's compliment]  $\Rightarrow$  9's complement of

1011 Invalid subtractend should be

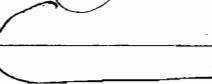
added to minuend if

we get valid op and

No carry which indicated

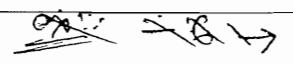
result is +ve and

9's complement should be taken for final answer

end 

around carry +1

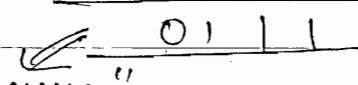
$$\begin{array}{r} 0010 \\ +2 \\ \hline \end{array}$$

$\Rightarrow$  9's complement of subtractend should be added to the minuend 

If the result is invalid or carry is generated then '9's' should be added with "end around carry".

ex:- -ve Result  $-6 \rightarrow 0011$  (9's complement)

$$+4 \rightarrow 0100$$

 0111 Valid

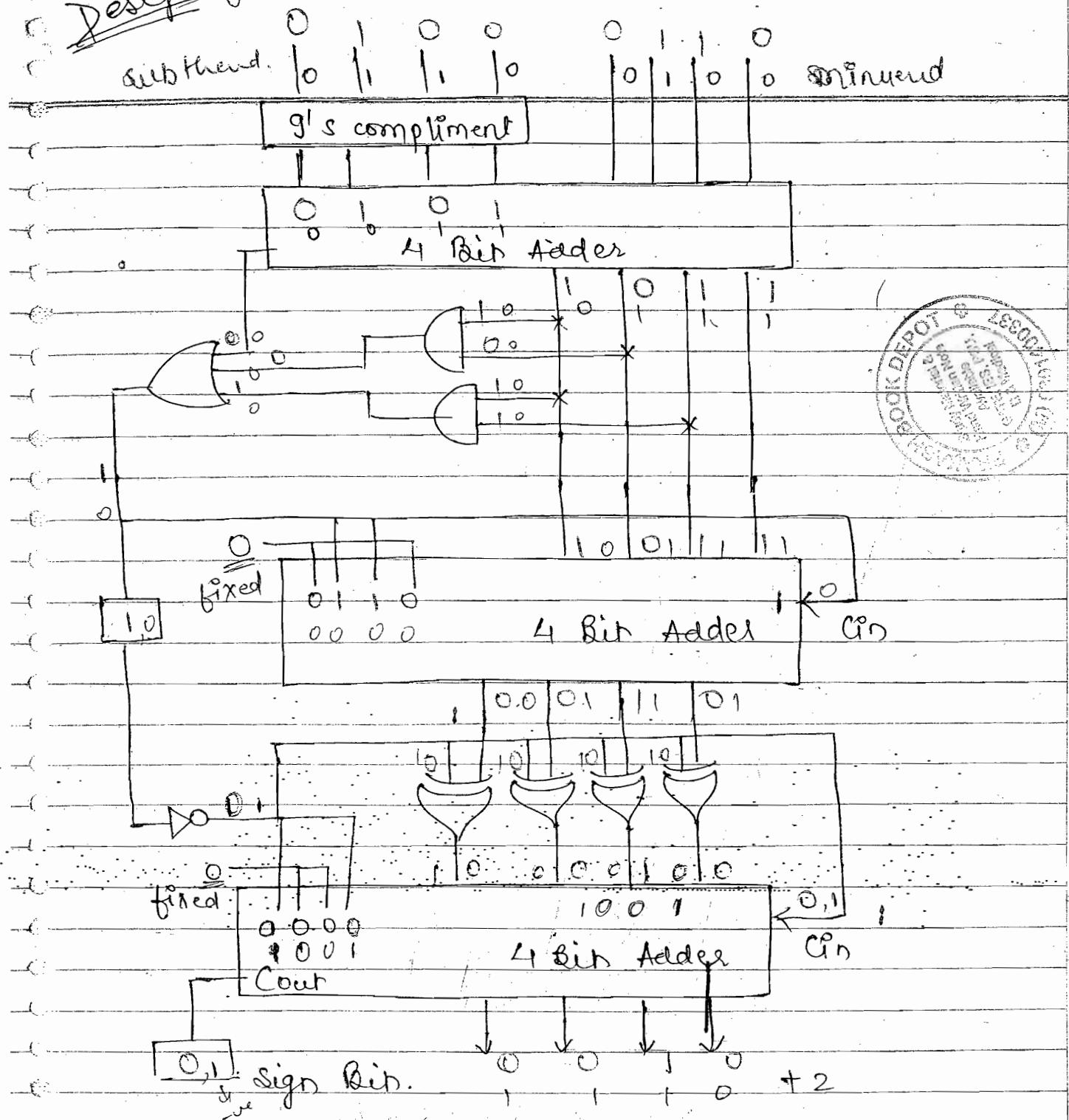
"No carry"

1) valid 2) Result -ve

3) No carry }

Ans Take 9's

$$\begin{array}{r} 9 \\ -7 \\ \hline -2 \end{array}$$

Designing

### 2's Complement Arithmetic

- Step 1: 2's complement of subtrahend is add to the minuend.
- Step 2: Check the msb Bit. (sign Bit)
- Step 3: If the sign bit is one take 2's complement

For the final answer, if the sign bit is zero  
answer is already in the true binary.

Step 9: Ignore the carry.

case (i)  $+9 \rightarrow 01001$   
 $+4 \rightarrow \begin{array}{r} 00100 \\ 01101 \end{array} \Rightarrow +13$

sign bit

case (ii)  $+9 \rightarrow \begin{array}{r} 0 \\ 1001 \end{array}$   
 $-4 \rightarrow \begin{array}{r} 1 \\ 10100 \\ 10101 \end{array} \Rightarrow +5$

carry (Ignore)

case (iii)  $-9 \rightarrow \begin{array}{r} 1 \\ 10111 \end{array} \quad (2\text{'s compliment of } 9)$   
 $+4 \rightarrow \begin{array}{r} 0 \\ 10000 \\ 10111 \\ 10101 \end{array} \Rightarrow -5$

sign bit 2's compliment

case (iv)  $-9 \rightarrow \begin{array}{r} 1 \\ 10111 \end{array}$   
 $-4 \rightarrow \begin{array}{r} 1 \\ 11000 \\ 11011 \end{array}$   

carry sign bit

 $11101 \Rightarrow -13$

case (v)  $-9 \rightarrow \begin{array}{r} 1 \\ 10111 \end{array}$   
 $+9 \rightarrow \begin{array}{r} 01001 \\ 10000 \end{array} \Rightarrow +0$

carry sign bit

- 1's complement : Arithmetical P.C.

Step 1:- Add the 1's complement subtrahend to the minuend

If the carry occurs make it end around carry.

Step 2:- Check the MSB Bit (Sign Bit). If it is 0

Answer is True Binary. If it is 1 take 1's complement for the final answer.

- Overflow Condition:- The carry enter to sign bit if it is known as overflow condition. Then sys. introduces one more bit for no. representation.

$$5 \rightarrow 101$$

$$4 \rightarrow 100$$

$$\begin{array}{r} 101 \\ + 100 \\ \hline \text{Carry} \end{array}$$

$$5 \rightarrow 00101$$

$$4 \rightarrow 00100$$

$$\begin{array}{r} 00101 \\ + 00100 \\ \hline \text{Carry} \end{array}$$

overflow condition:  $f = \bar{x}\bar{y}z + x\bar{y}z$

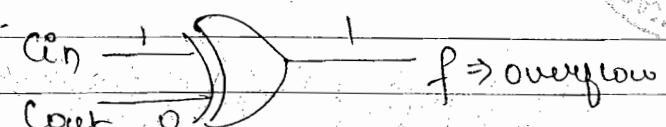
$$= \bar{0} \bar{0} 1 + 0 \bar{0} 1 = 1 + 0$$

$$= 1 \quad (\text{Overflow})$$

if  $f = 0$  No overflow

if  $f = 1$  overflow.

~~2nd~~ 2nd Method:-



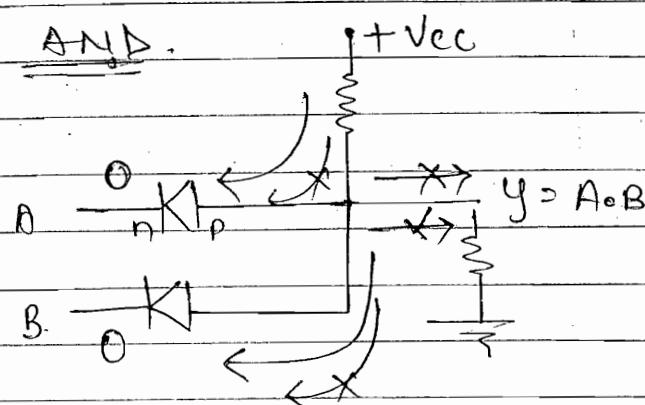
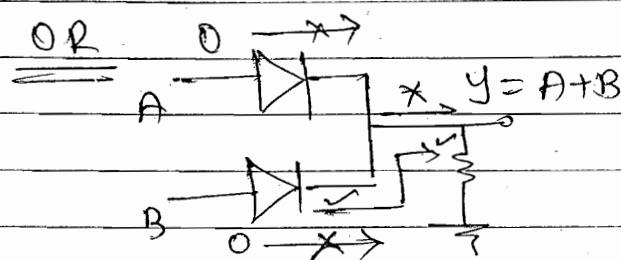
if  $f = 0 \rightarrow \text{No overflow}$   
 $f = 1 \rightarrow \text{overflow.}$

$Cin \rightarrow \text{Carry } 1+1 \Rightarrow 10$

$Cout \rightarrow \text{Carry coming out } 1 \Rightarrow 0.$

# Logic Family (Theory Book Parameter Table Imp)

- Diode has logic gate.

ANDOR

A	B	D <sub>1</sub>	D <sub>2</sub>	y
0	0	0	0	0 GND
0	1	1	0	1
1	0	0	1	1
1	1	1	1	1

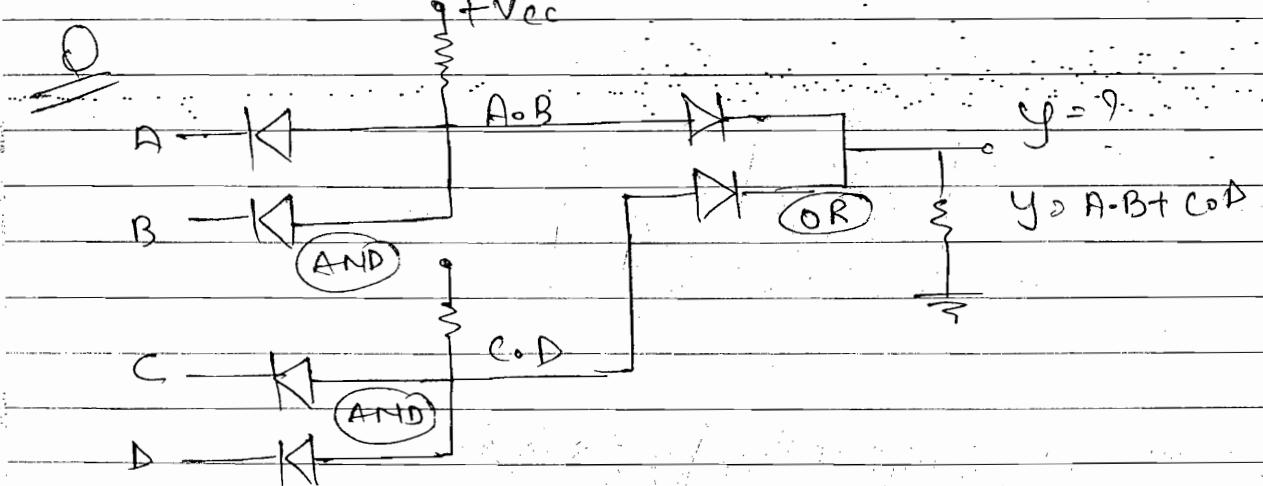
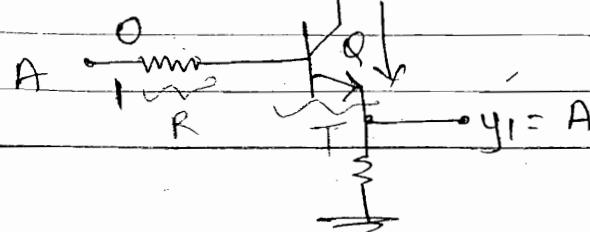
A	B	D <sub>1</sub>	D <sub>2</sub>	y
0	0	✓ <sup>ON</sup>	✓	0
0	1	✓	X	0
1	0	X <sup>OFF</sup>	✓	0
1	1	X	X	1

+Vcc

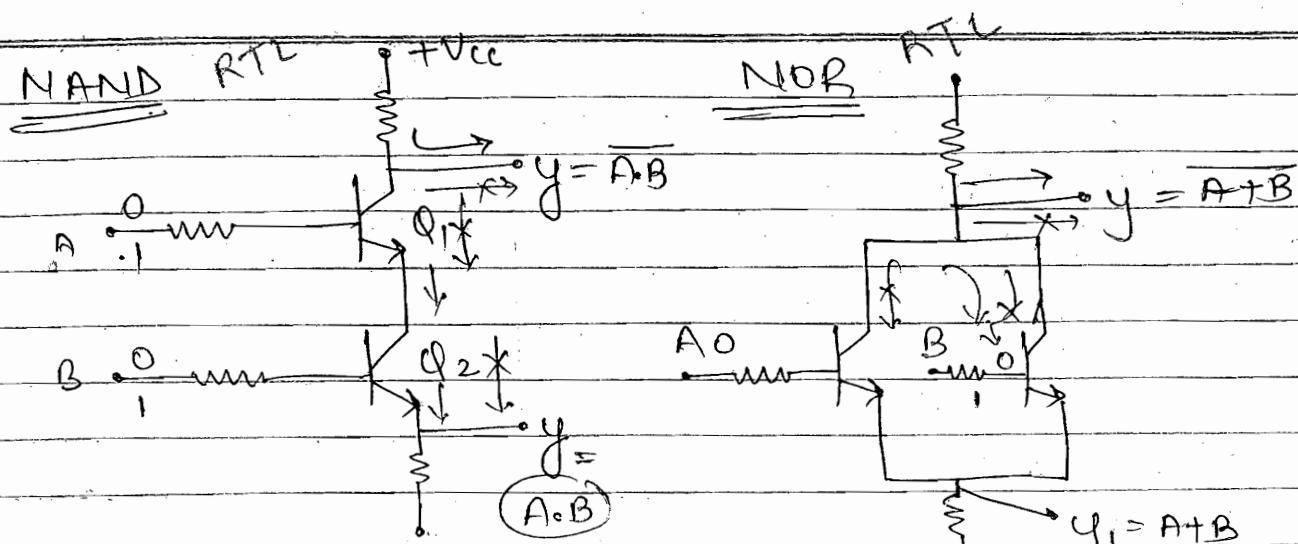
A · B

y = 9

y = A · B + C · D

NOT (RTL)

A	y
0	1
1	0



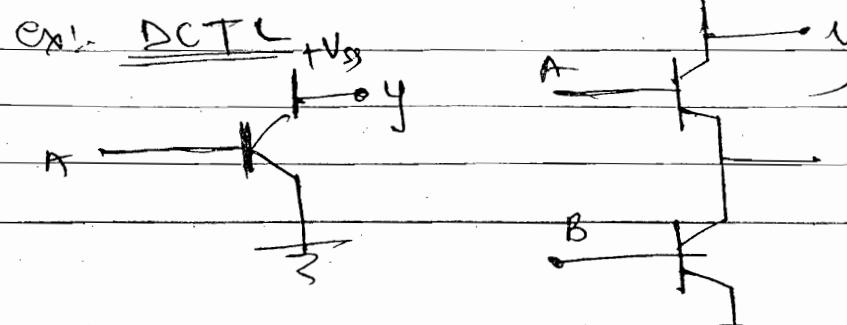
A	B	y
0	0	1
0	1	1
1	0	1
1	1	0

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

• DCTL :- Direct Coupled Transistor logic  
It is same as that of RTL except the input resistors are removed.

NOTE :- DCTL suffers with current hogging problem

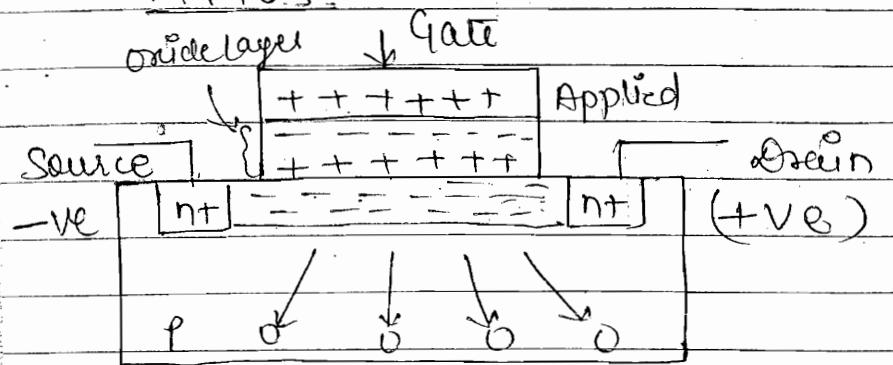
Current Hogging :- Because of different saturation level of loading gate current will be hogged in certain nodes only, and rest of remaining loads are starvation of current, known as current hogging



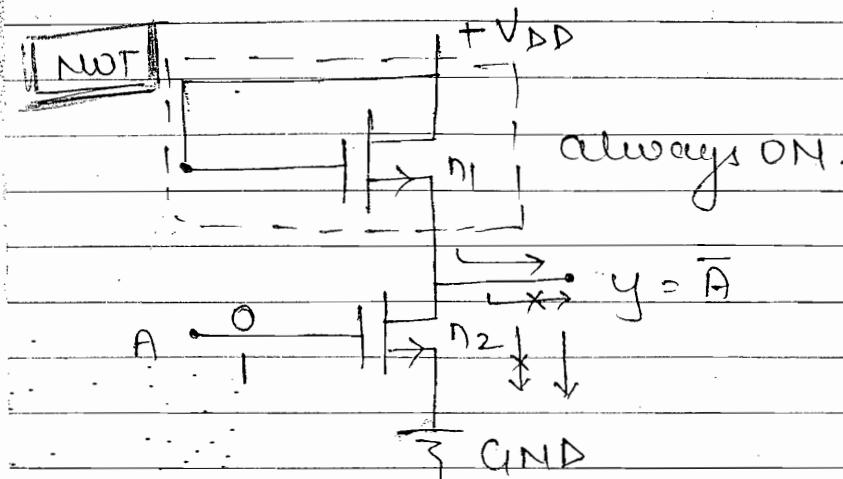
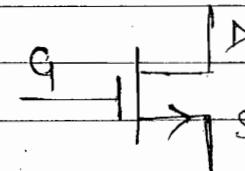
Conventional Diagr. Truth Table 15 Marks

MOS Technology

• N MOS:

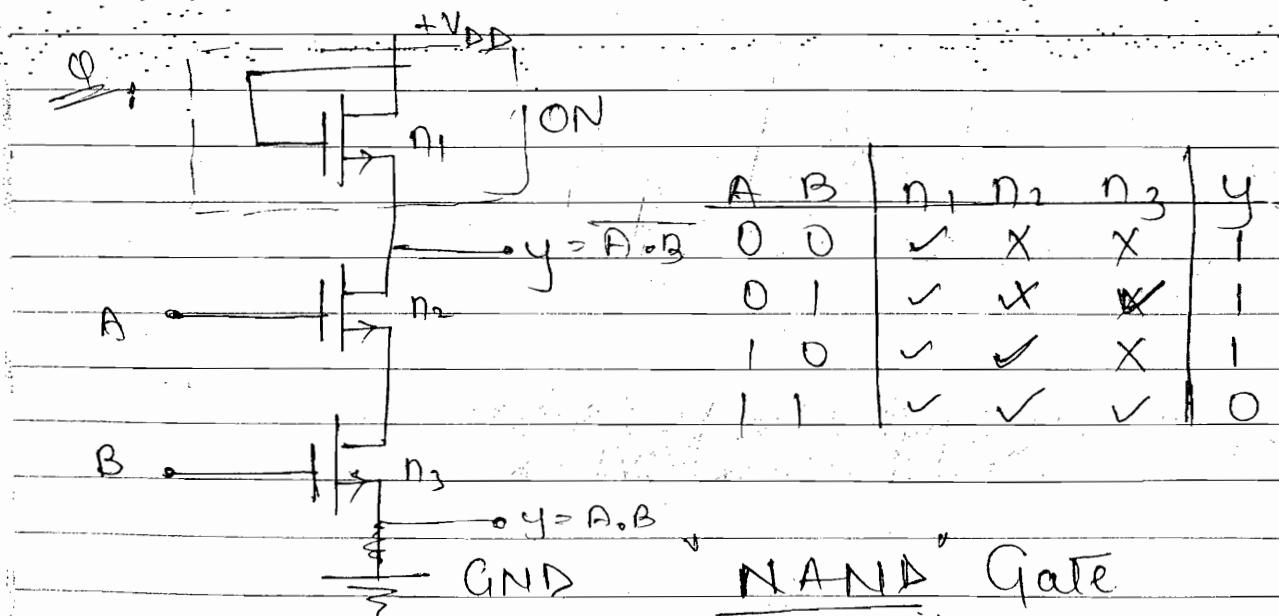


NMOS:  $\rightarrow$  Gate  
+ve  $\rightarrow$  ON  
-ve  $\rightarrow$  OFF



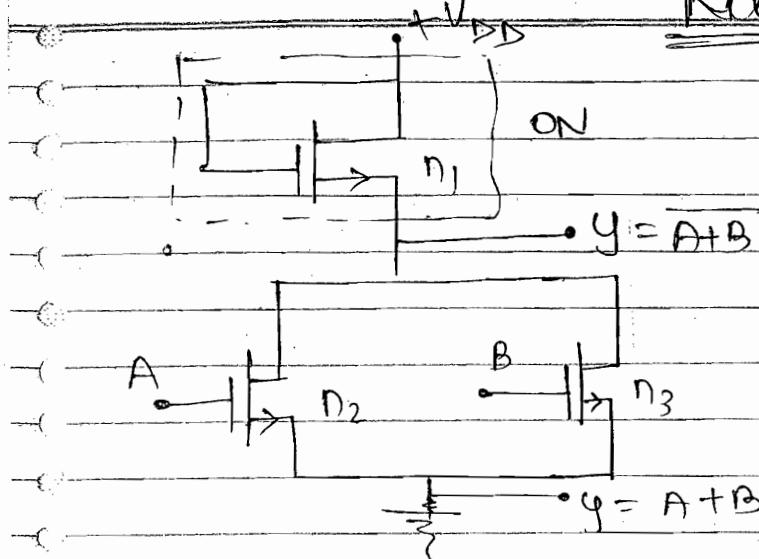
Symbol

A	n <sub>1</sub>	n <sub>2</sub>	y
0	✓	x	1
1	✓	✓	0



A	B	n <sub>1</sub> , n <sub>2</sub>	n <sub>3</sub>	y
0	0	✓	x	1
0	1	✓	x	0
1	0	✓	✓	1
1	1	✓	✓	0

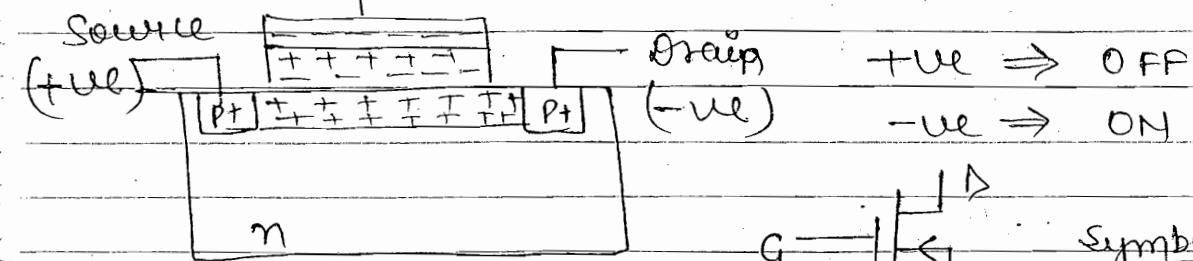
NAND Gate

"NOR" Gate

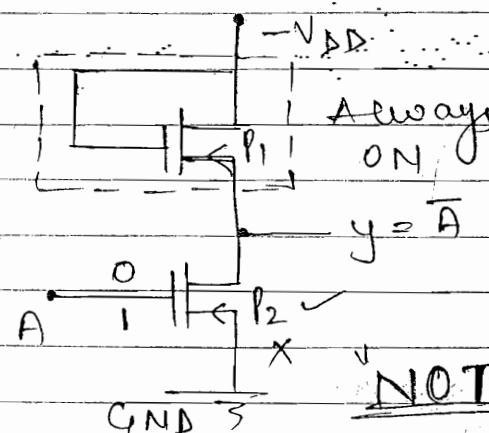
A	B	n <sub>1</sub> n <sub>2</sub> n <sub>3</sub>	y
0	0	✓ X X	1
0	1	✓ X ✓	0
1	0	✓ ✓ X	0
1	1	✓ ✓ ✓	0

PMOS :-

Gate (-ve)

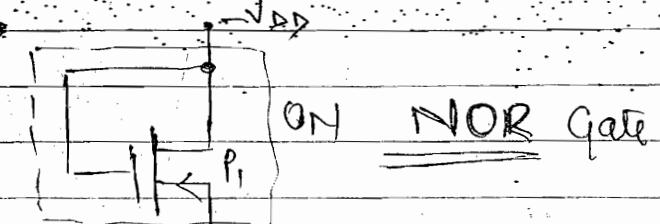
PMOS :- Gate

Symbol



NOT

y = A



NOR Gate

y = A + B

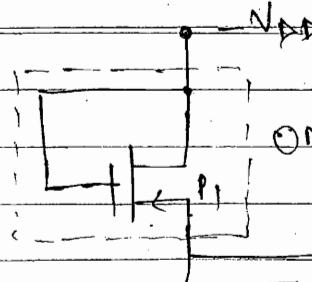
A	P <sub>1</sub>	P <sub>2</sub>	y
0	✓	✓	1
1	✓	X	0

A	B	P <sub>1</sub>	P <sub>2</sub>	P <sub>3</sub>	y
0	0	✓	✓	X	1
0	1	✓	X	✓	0

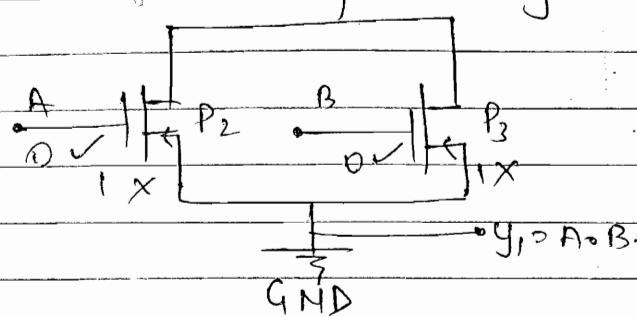
GND. A + B

NOR Gate Table

A	B	P <sub>1</sub>	P <sub>2</sub>	P <sub>3</sub>	Y
0	0	✓	✓	✓	1
0	1	✓	✓	X	0
1	0	✓	X	✓	0
1	1	✓	X	X	0



ON

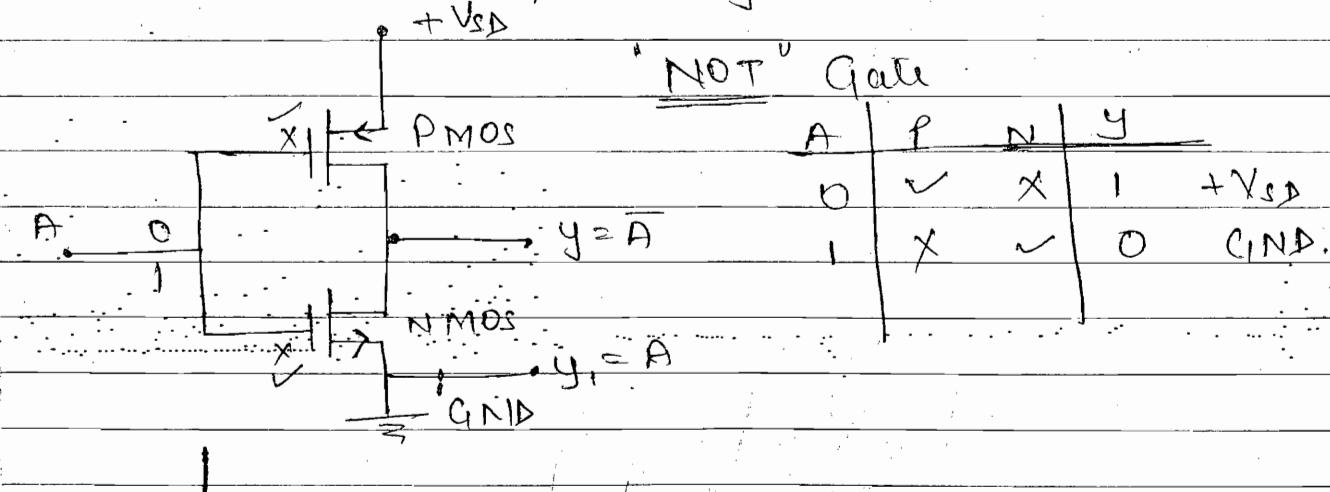
"NAND" Gate

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

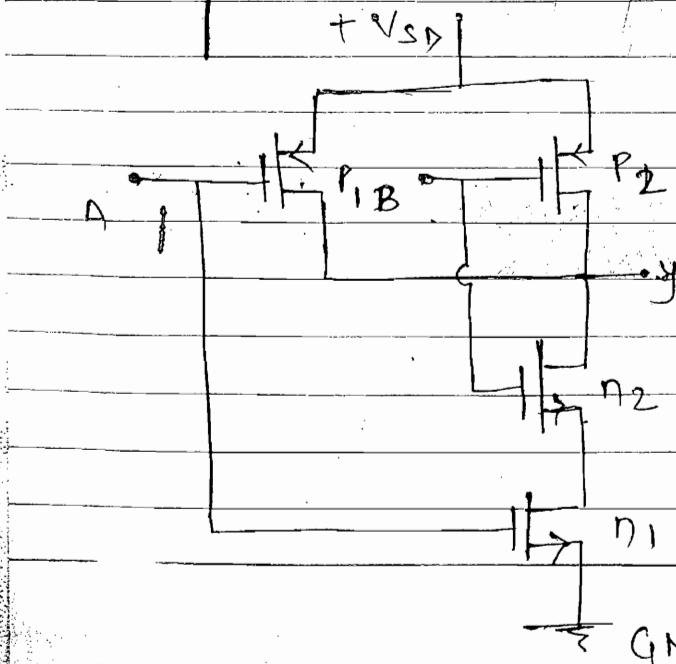
$$y = \overline{A \cdot B}$$

GND

A	B	P <sub>1</sub>	P <sub>2</sub>	P <sub>3</sub>	Y
0	0	✓	✓	✓	1 GND
0	1	✓	✓	X	1 GND
1	0	✓	X	✓	1 GND
1	1	✓	X	X	0 -V <sub>DD</sub>

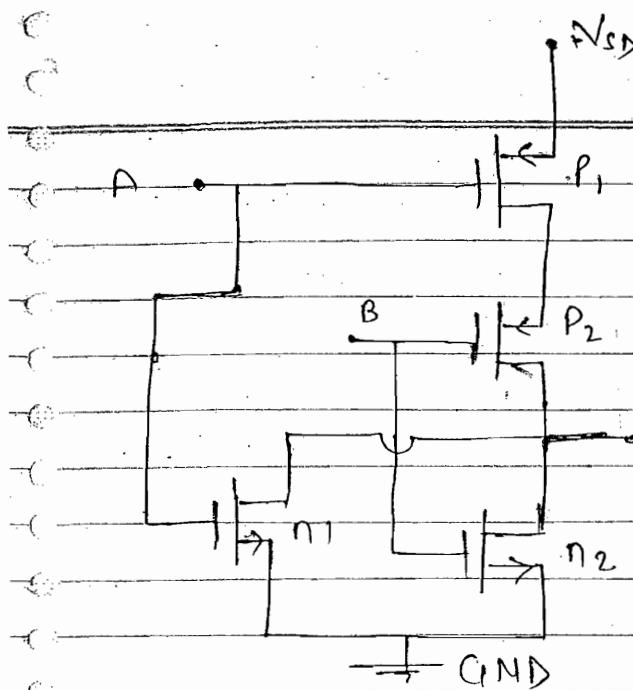
• C MOS :- Complementary MOS :-+V<sub>DD</sub>"NOT" Gate

A	P	N	Y
0	✓	X	1 +V <sub>DD</sub>
1	X	✓	0 GND

"N AND" Gate

A	B	P <sub>1</sub>	P <sub>2</sub>	N <sub>1</sub>	N <sub>2</sub>	Y
0	0	✓	X	X	X	1 +V <sub>DD</sub>
0	1	✓	X	X	✓	1 +V <sub>DD</sub>
1	0	X	✓	✓	X	1 +V <sub>DD</sub>
1	1	X	X	✓	✓	0 GND

GND



A	B	P <sub>1</sub>	P <sub>2</sub>	N <sub>1</sub>	N <sub>2</sub>	Y	
0	0	✓	✓	✗	✗	1	V <sub>DD</sub>
0	1	✓	✗	✗	✓	0	GND
1	0	✗	✓	✓	✗	0	GND
1	1	✗	✗	✓	✓	0	GND

Shortcut:-

• (N-p-n) OR NMOS:-

(i) column  $\rightarrow$  ~~AND~~ Multiplication  $\rightarrow$  AND operation

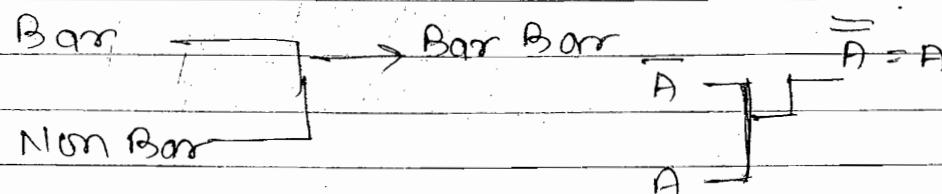
(ii) Row  $\rightarrow$  OR operation  $\rightarrow$  Addition.

• (p-n-p) OR PMOS:-

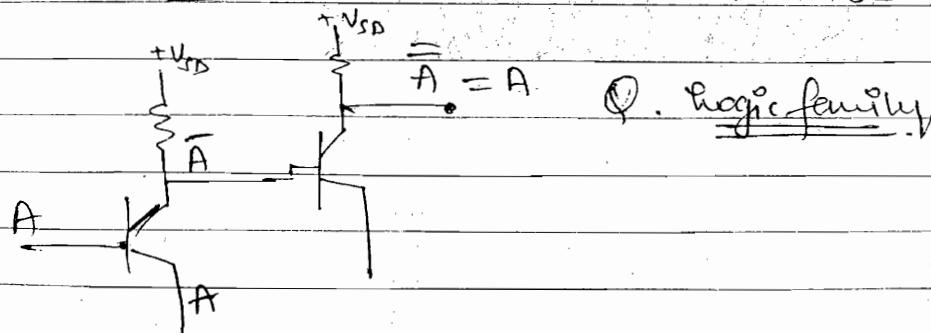
(i) column  $\rightarrow$  OR operation  $\rightarrow$  Addition

(ii) Row  $\rightarrow$  AND  $\rightarrow$  Multiplication

For Both:-

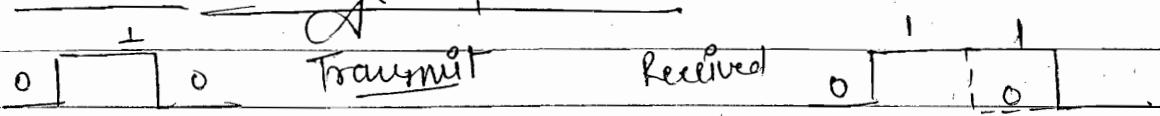


• CMOS :- Concentrate on N-mos (and and)



Q. Logic family

- Even Parity Generator

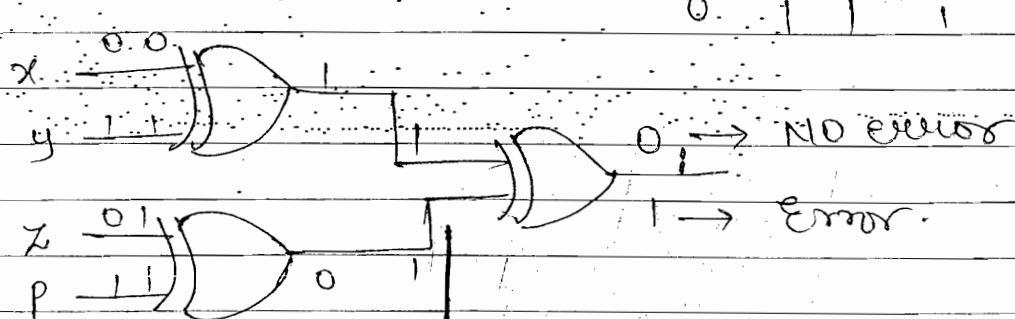


- Parity Even parity.  $0 \ 1 \ 0 \ \boxed{1} \ P$

x	y	z	p
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

$\rightarrow$  No. of 1's including parity must be even No.  $\Rightarrow$  No error.

- Even Parity Checker



- Hamming Code :- (7 Bit)

Data  $\rightarrow$  1 1 0 1

Tx  $\rightarrow$  P<sub>1</sub> P<sub>2</sub> P<sub>3</sub> P<sub>4</sub> P<sub>5</sub> P<sub>6</sub> P<sub>7</sub>  
 1 0 1 0 1 0 1

for 7 Bit  $\rightarrow$  Parity Bits are  $\Rightarrow$  P<sub>1</sub> P<sub>2</sub> P<sub>4</sub>

12 Bit  $\rightarrow$  u u  $\Rightarrow$  P<sub>1</sub> P<sub>2</sub> P<sub>4</sub> P<sub>8</sub>

15 Bit  $\rightarrow$  u u  $\Rightarrow$  P<sub>1</sub> P<sub>2</sub> P<sub>4</sub> P<sub>8</sub>

thus are data bits.

	$P_4$	$P_2$	$P_1$	
	$\uparrow$	$\uparrow$	$\uparrow$	
	$2^2$	$2^1$	$2^0$	To make even parity $\Rightarrow$ I's NO. even
	$x$	$y$	$z$	
0	0	0	0	$P_1 (1, 3, 5, 7) \Rightarrow (P_1 1 1 1) \Rightarrow P_1 = 1$
1	0	0	1	$P_2 (2, 3, 6, 7) \Rightarrow (P_2 1 0 1) \Rightarrow P_2 = 0$
2	0	1	0	$P_3 (4, 5, 6, 7) \Rightarrow (P_3 1 0 1) \Rightarrow P_3 = 0$
3	0	1	1	
4	1	0	0	$R_x (\text{Received})$ : $1 \ 0 \ (\underline{0}) \ 0 \ 1 \ 0 \ 1$
5	1	0	1	$P_1 \ P_2 \ P_3 \ P_4 \ P_5 \ P_6 \ P_7$
6	1	1	0	$P_1 (1, 3, 5, 7) \Rightarrow (1 0 1 1) \Rightarrow V = 1$
7	1	1	1	$P_2 (2, 3, 6, 7) \Rightarrow (0 0 0 1) \Rightarrow P = 1$
				$P_4 (4, 5, 6, 7) \Rightarrow (0 1 0 1) \Rightarrow x = 0$

 $\alpha \beta \gamma$ 

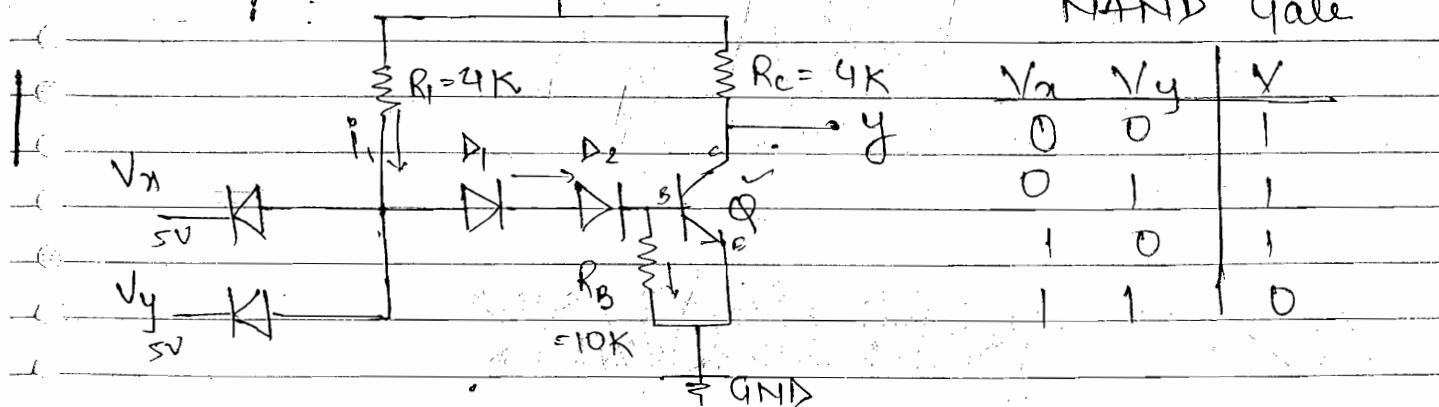
0 1 1

 $\Rightarrow (3) \rightarrow 3 \text{ Bit having error}$ If No error  $\alpha \beta \gamma = 0 0 0$ 

29/9/2014

DTL :- (Diode Transistor logic) $+V_{CC} = 5V$ 

NAND Gate

Ckt Analysis :-Given  $V_y = 0.7V$  Cut off voltage of Diode $V_{BE}(\text{ON}) = 0.7V$  $V_{BE}(\text{sat}) = 0.8V$  $V_{CE}(\text{sat}) = 0.1V$  $\beta = 25$

case(i)  $V_x = 0.1 \text{ V}$   $V_y = 0.1 \text{ V}$

$$i_1 = \frac{V_{cc} - V_1}{R_1} = \frac{5 - 0.8}{4K} = 1.05 \text{ mA}$$

$$i_2 = i_B = i_R = 0 \Rightarrow Q \text{ is off}$$

$\Rightarrow y = \text{High (5V)}$

case(ii)  $V_x = 0.1 \text{ V}$   $V_y = 5 \text{ V}$

Same as the of case(i)  $\Rightarrow y = \text{High (5V)}$

case(iii)  $V_x = 5 \text{ V}$   $V_y = 0.1 \text{ V}$

Same as the case(ii)  $\Rightarrow y = \text{High (5V)}$

case(iv)  $V_x = 5 \text{ V}$   $V_y = 5 \text{ V}$

$$i_1 = \frac{5 - 2.2}{4K} = 0.7 \text{ mA}$$

$$i_2 = 0.7 \text{ mA} \quad i_R = \frac{V_{BE}(\text{sat})}{10} = 0.8 \times 0.08 \text{ mA}$$

$$i_R = 0.08 \text{ mA}$$

$$i_B = i_1 - i_R = 0.7 - 0.08 = 0.62 \text{ mA}$$

$$i_C = \frac{V_{cc} - V_{CE}(\text{sat})}{R_C} = \frac{5 - 0.1}{4K} = 1.2 \text{ mA}$$

$$i_B = 0.62 \text{ mA} \quad i_C = 1.2 \text{ mA}$$

Saturation condition:  $\frac{i_C}{i_B} \leq \beta$ .  $\frac{1.2}{0.62} < 25, 1.9 < 25$

$\Rightarrow Q$  is in saturation region.

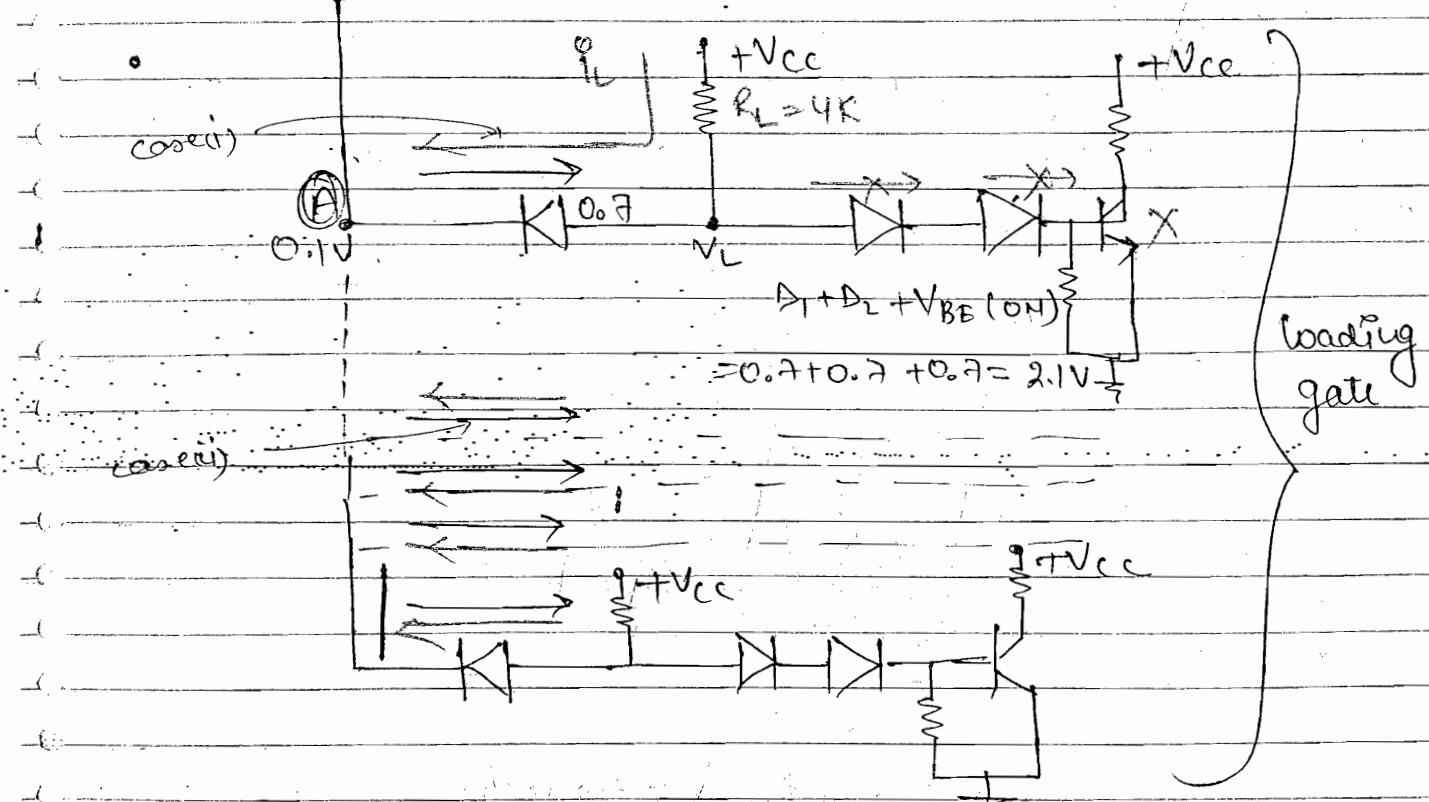
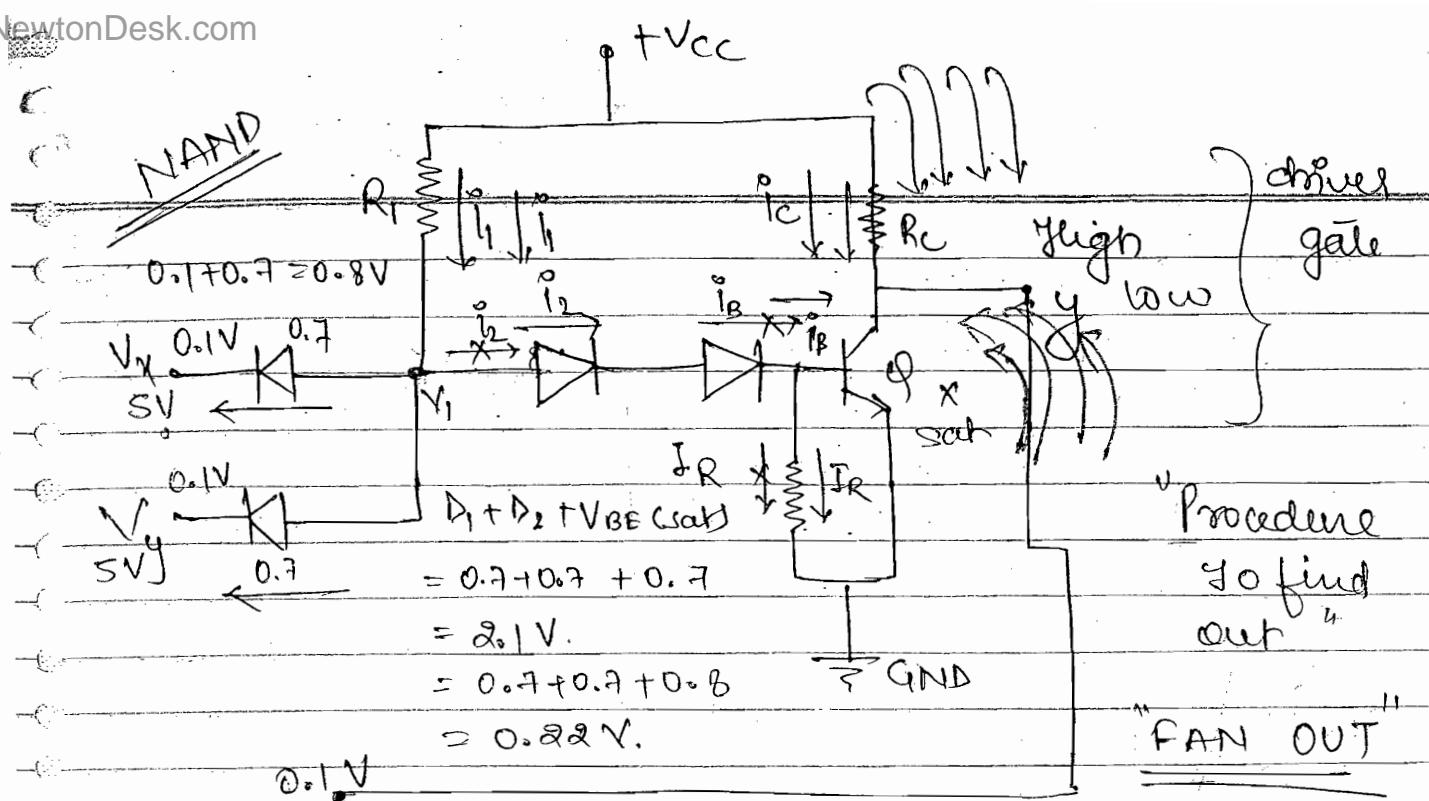
At the saturation condition if necessary because

the more voltage will occur at  $R_C = 4K$

by large value of  $i_C$ , so low voltage at  $y$ .

- Drive  $1K \rightarrow$  connected are loaded ckt.

Find the no. of load ckt can drive



$$I_L = I \frac{V_{CC} - V_L}{R_L} = \frac{5 - 0.8}{4K} = 1.05 \text{ mA}$$

$$i_c^e = i_c + NL$$

Saturation condition  $i_c / i_B \leq \beta$

$$\Rightarrow N=13$$

Valid for this values only

NewtonDesk.com  
Opp low case of driving  $\rightarrow$  Sinking  
Opp high case of driving  $\rightarrow$  Sourcing.

$$i_c + N i_l \rightarrow 25$$

$$N \approx 13$$

$$1.2 + N(1.05) = 25$$

$$0.62$$

- When the  $V_g$  is low  $\rightarrow$  all load ckt gate get on. So all the ~~the~~ load gate draw the current which is coming from  $V_E + V_{ce} = 5V$  (driving ckt)

- Procedure:-

case (i) Opp of driving gate is low :- In this case the voltage drop  $V_L$  for the loading gate when becomes as 0.8V then exist a load current which passes through the saturated Transistor of driving gate. Then for  $N$  No. of loads we get  $\frac{i_c}{i_B} = \frac{i_c + N i_l}{i_B}$  and the saturation condition will be  $\frac{i_c}{i_B} \leq \beta$ . i.e.  $\frac{i_c + N i_l}{i_B} \leq \beta$ .

where  $N$  indicates the fan out value.

Case (ii) : Opp of driving gate is high:- In this pnp diode of loading gate is reverse Biased and there exist a reverse saturation current which passed through the  $R_c$  of driving gate. And by  $N$  No. of loaded it causes more voltage drop across  $R_c$  and Opp can alter from high voltage which leads to improper operation.

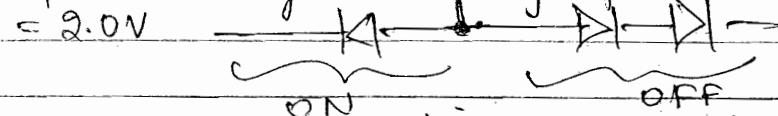
- So there is a limit on the fan out even in this case also.

NOTE :- From the among two cases the least value is preferable for overall fan out.

So O/p low case of driving gate is preferable.

- No. of load gate can be connected more in case high because minority current is very low compare to low case.

- At p/n junction voltage can vary upto  $1.3V$  bcoz  $1.3 + 0.7$



This section will ON. (proper function)

⇒ This called Noise margin.

- Procedure to find Noise Margin:-

case(i) O/p of driving gate is low:- In this case the other section of loading gate to become ON the voltage drop required at V<sub>in</sub> is  $D_1 + D_2 + V_{BE}(\text{ON}) = 0.7$

So the difference of voltage  $= 2.1 - 0.8 = 1.3V$   $+ 0.7 + 0.7 = 2.1V$

is the maximum acceptable floating voltage known as "NOISE MARGIN".

case(ii) O/p of driving gate is high:- In this case also fan out should be measured.

Noise Margin

NOTE:- Among above two case noise margin should be measured & least value is preferable of noise margin.

- Procedure to find Power dissipation :-

$$P_D = V_{cc} \cdot I_{cc} = \frac{P_D(\text{low}) + P_D(\text{high})}{2}$$

For high - OFF Transistor.  $i_1 = i_{cc}$

for low - ON - Transistor  $i_{ct} + i_2 = i_{cc}$

- $D_1 \& D_2$  If only one diode are removed, fanout & Noise Margin effects.

FAN OUT value will  $\uparrow$ , Noise Margin  $\downarrow$

- If add one More diode  $D_3$ ,  $D_1 \& D_2$  fanout & Noise Margin value effect. Fanout value  $\downarrow$  Noise Margin  $\uparrow$

- As per client Requirement of fanout and noise, we select the diode value and No. of diodes as per circuit value.

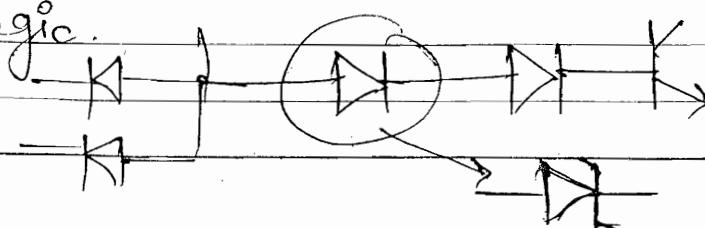
Q What happen when one of diode is Removed in DTI ckt:

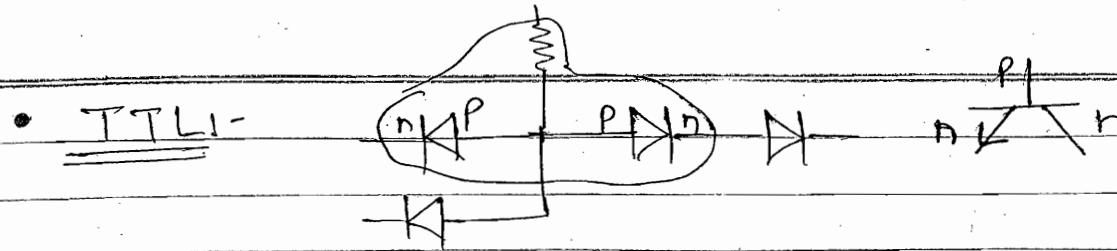
Ans FAN OUT  $\uparrow$ , NOISE MARGIN  $\downarrow$

Q What happen when one more diode is Added.

Ans FAN OUT  $\downarrow$ , NOISE MARGIN  $\uparrow$

- TTL: High Threshold logic: When one of diode is replaced by zener diode, Noise Margin at levels can be  $\uparrow$  known as High threshold logic.



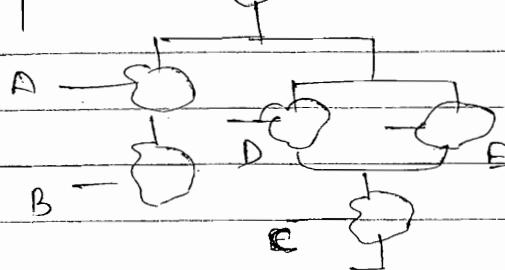


procedure will remain same only values get change

Q1 IES Q2 Design the <sup>ckt</sup> MOSFET By using NMOS:-  
~~IS Marks~~

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

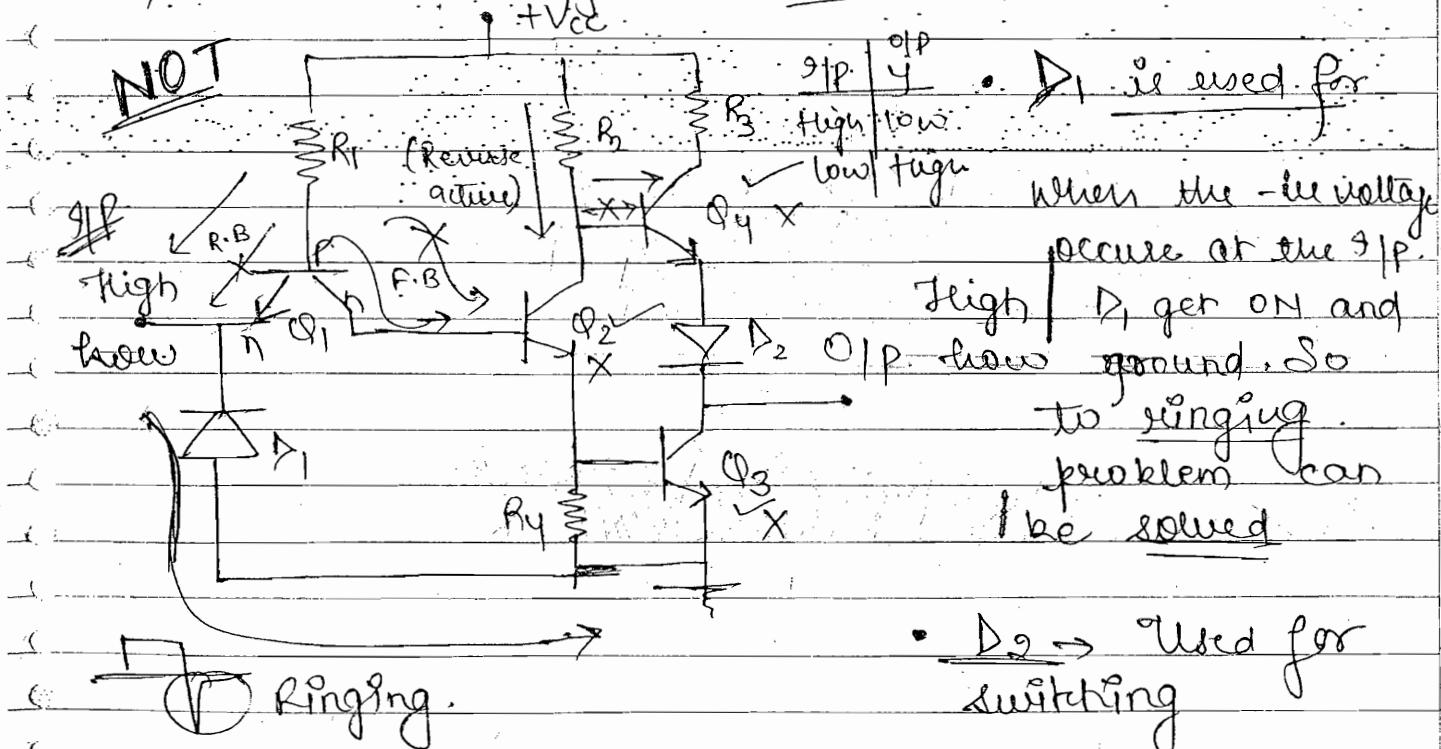
By NMOS



By CMOS. NMOS designs

remain same, for PMOS  
 again draw the ckt above NMOS  
 by now  $\Rightarrow$  column 2

Q3 Draw the diagram: ITL

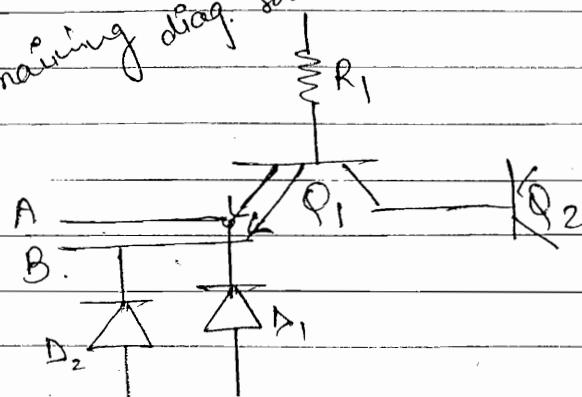


$\rightarrow$  When the  $\overline{Q1}$  is high  $\overline{Q1}$  is  $R_{B1}$ , any fluctuation occurs NOT problem.

- Purpose of  $D_1$  :- It avoids the ringing i.e. the -ve voltage fluctuation can be grounded by the J/P diode  $D_1$ , so that the J/P. transistor is saved.
- Purpose of  $D_2$  :- For proper switching.

- TTL NAND Gate

Remaining diag. save



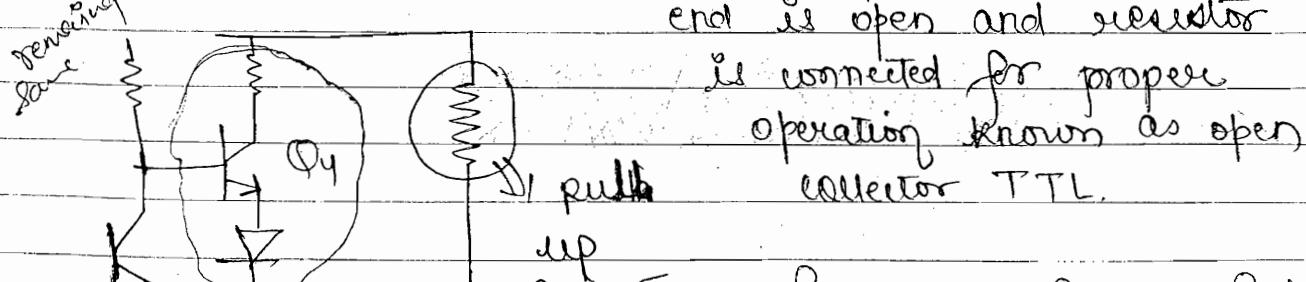
More J/P. It becomes NAND, when at least J/P is low, O/P is high.

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

- TTL logic are three types :-

(i) Totem pole TTL :- The above discuss TTL is known as totem pole TTL. Bcoz Both O/P transistor cannot be ON simultaneously ( $Q_3, Q_4$ )

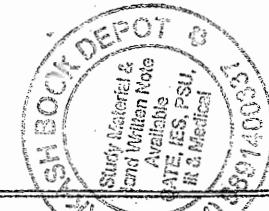
(ii) Open Collector TTL :- The O/P transistor collector end is open and resistor is connected for proper



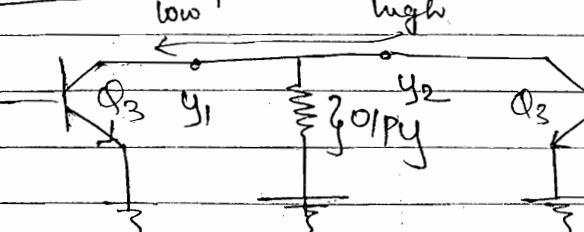
Purpose of Pull up Resistor

① for proper O/P switching voltage.

② It eliminates power supply fluctuation.



\* NOTE Open collector TTL allows wired logic.



→ This  $Q_3$  is same (another  $Q_3$  taken).

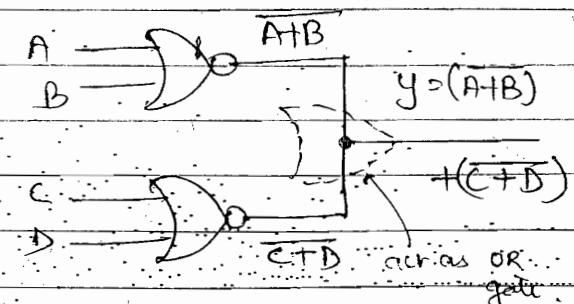
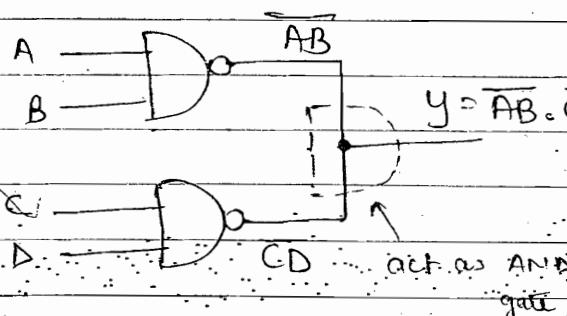
→  $y_2$  will give to  $y_1$ , NOT  
left (current) for  $y$ . so  
 $y > 0$ .

→ When both  $y_1 = y_2 = 1$

$0 \parallel y > 1$   
So called wired AND logic

wired AND logic

wired OR logic

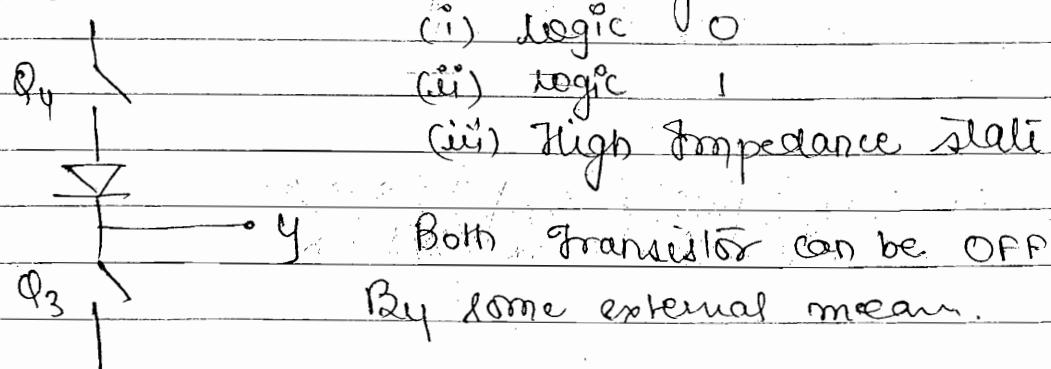


(iii) Ter state TTL: Having three states

(i) logic 0

(ii) logic 1

(iii) High Impedance state



Both transistors can be OFF together  
By some external means.

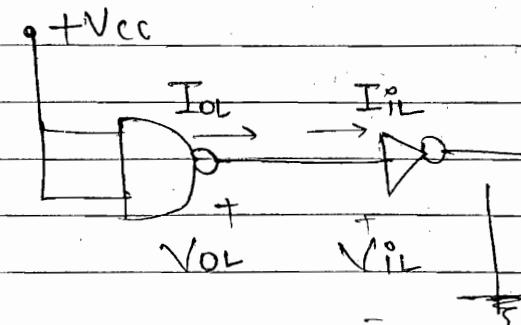
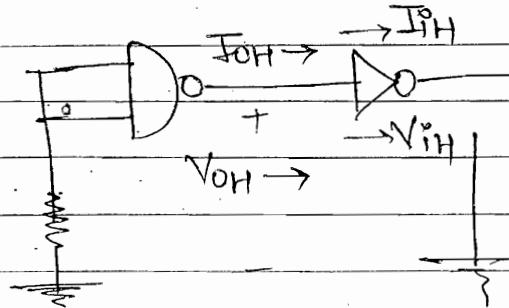
→ When current driving  $\rightarrow$  load  $\Rightarrow$  Sourcing  
→ "  $\rightarrow$  load  $\rightarrow$  driving  $\Rightarrow$  driving

Q Question comes to find fanout value.

~~I<sub>ES</sub>~~ Voltage and Current parameters:-

High

Low



graph

$$\left. \begin{array}{l} V_{OH} \\ V_{IH} \end{array} \right\} \text{Noise Margin}_H = V_{OH} - V_{IH}$$

$$\text{Fanout} = \frac{I_{OH}}{I_{IH}}$$

Take  
Refer  
value.

$$\left. \begin{array}{l} V_{IL} \\ V_{OL} \end{array} \right\} \text{Noise Margin}_L = V_{IL} - V_{OL}$$

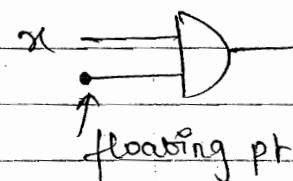
$$\text{Fanout} = \frac{I_{OL}}{I_{IL}}$$

- $V_{OH} \rightarrow$  O/p high voltage    $I_{OH} \rightarrow$  O/p high current
- $I_{IH} \rightarrow$  S/p high current    $V_{IH} \rightarrow$  S/p high voltage
- $I_{OL} \rightarrow$  O/p low current    $I_{PL} \rightarrow$  S/p low current
- $V_{OL} \rightarrow$  O/p low voltage    $V_{IL} \rightarrow$  S/p low voltage

- Suppose  $V_{OH} \rightarrow 5V$  But due to voltage fluctuation or by some other means  $V_{IH} \neq 5V$ , it is  $4.9V, 4.8V, 4.7V$  upto it acceptable. But below these values high voltage  $5V$  no any more high voltage become low voltage.

- $V_{OL} =$  low voltage  $= 0V$  (ground voltage)  $V_{IL}$  can be accepted upto  $0.1, 0.2, 0.3V$  but above this value  $V_{IL}$  is no any more low voltage it becomes high voltage.

NOTE: The floating  $\Sigma p$  in TTL technique is taken as



$\Sigma p$  is not connected to any logic

floating pt.

Q

No. of NAND Gate = ?

SOP (Given)  $\Rightarrow$  AND-OR.

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

Implementation of

logic gate By NAND

Shortcut:- Bar Bar, split einer Bar.

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

= 3 NAND Gate

NAND

• all terms should be product terms.

• To get this  $\Rightarrow$  wherever be the + make it double compliment ( $\overline{\overline{+}}$ )

for NOR gate :

• all terms should be sum terms.

• To get this  $\Rightarrow$  wherever be the + make it double compliment ( $\overline{\overline{+}}$ )

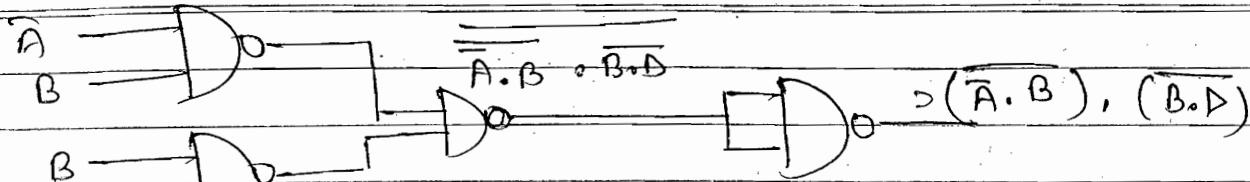
In Pos form (NAND): -

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

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

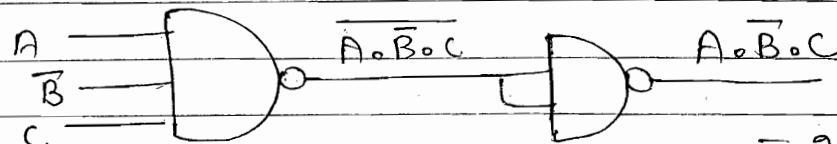
$$= (\overline{\overline{A}} \cdot \overline{\overline{\bar{B}}}) \cdot (\overline{\overline{\bar{B}}} \cdot \overline{\overline{\bar{D}}})$$

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



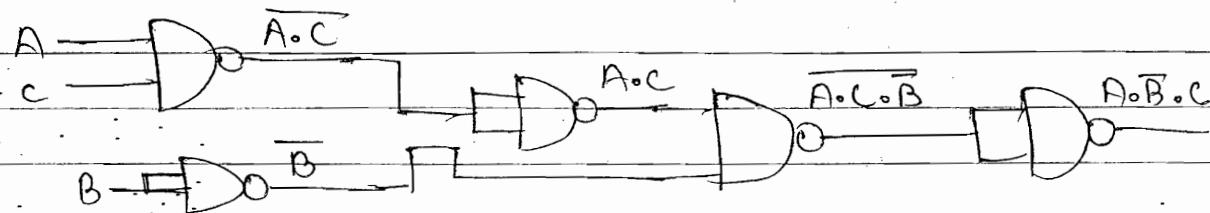
one additional NAND gate used

If there is 1 p. f.  $f = A \cdot \overline{B} \cdot C$



By using 2 1 p. terminal

→ Take complementary term separately



(NOR gate) OR-AND Implementation of logic gate By using NOR gate

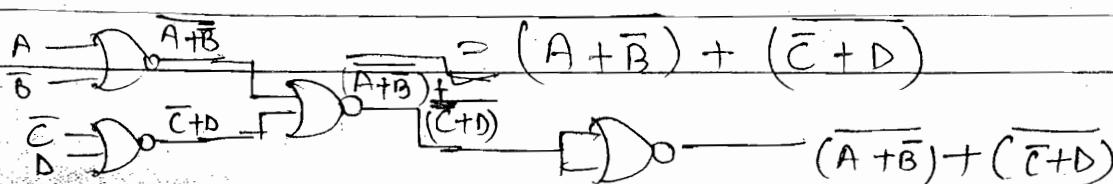
$$\text{POS} \Rightarrow f = (\overline{A} + B) \cdot (C + \overline{D})$$

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

3 NOR gate

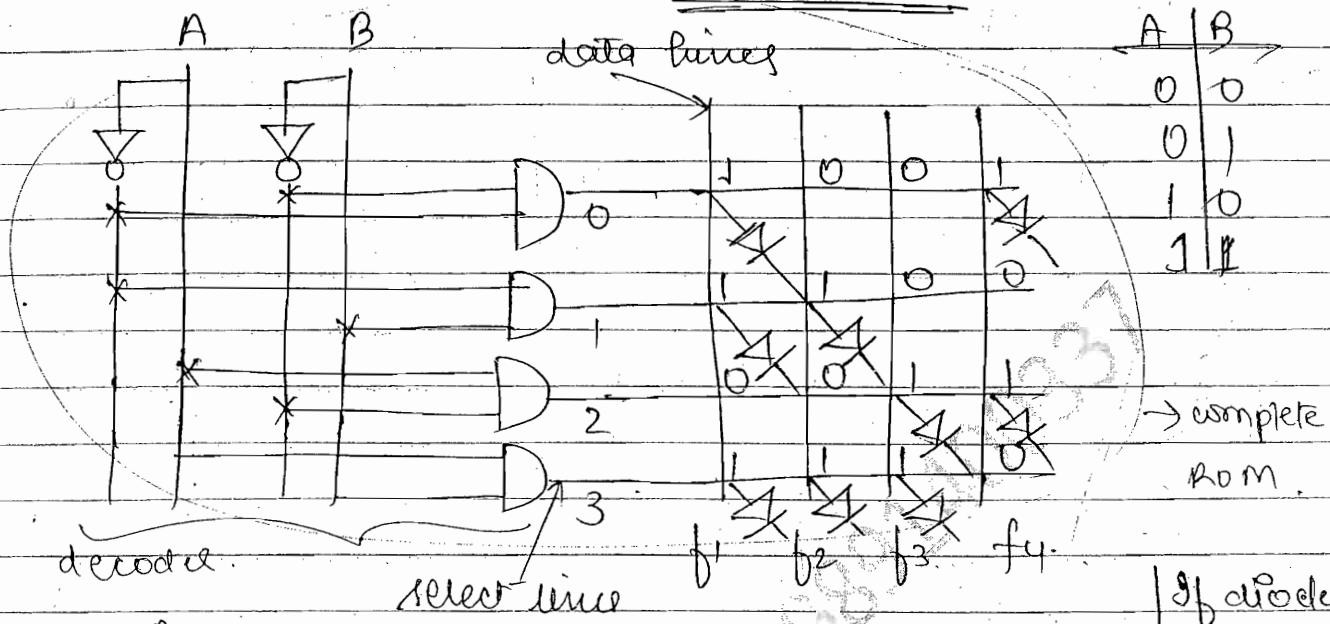
$$\text{In SOP form: } f = \overline{A} \cdot \overline{B} + \overline{C} \cdot \overline{D}$$

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



Memory :-

30/8/2014

ROM :- (Designing): $2^n \times m$  ROM

- Diodes are putting there where 1' having. If diode is connected it call DIODE
- Blocks are memory allocation

$$16 \rightarrow 4 \times 4 \rightarrow \text{data lines} = 2^n \times 4$$

$$\rightarrow 2^n \times m$$

where  $n = \text{select lines}$        $m = \text{data lines}$  place in diodes.

Given  $f_1 = \sum m (0, 1, 3)$        $f_2 = \sum m (1, 3)$   
 $f_3 = \sum m (2, 3)$        $f_4 = (0, 2)$

(above circuit)

- For 3 Bit Square :-

$$9 \times 9 \rightarrow 0$$

1

2

3

4

5

6

7

$$7 \times 7 = 49$$

32	16	8	4	2	1
----	----	---	---	---	---

$$7 \rightarrow 7^2 = 49$$

1	1	0	0	0	1
---	---	---	---	---	---

$f_6$	$f_5$	$f_4$	$f_3$	$f_2$	$f_1 \rightarrow m$
-------	-------	-------	-------	-------	---------------------

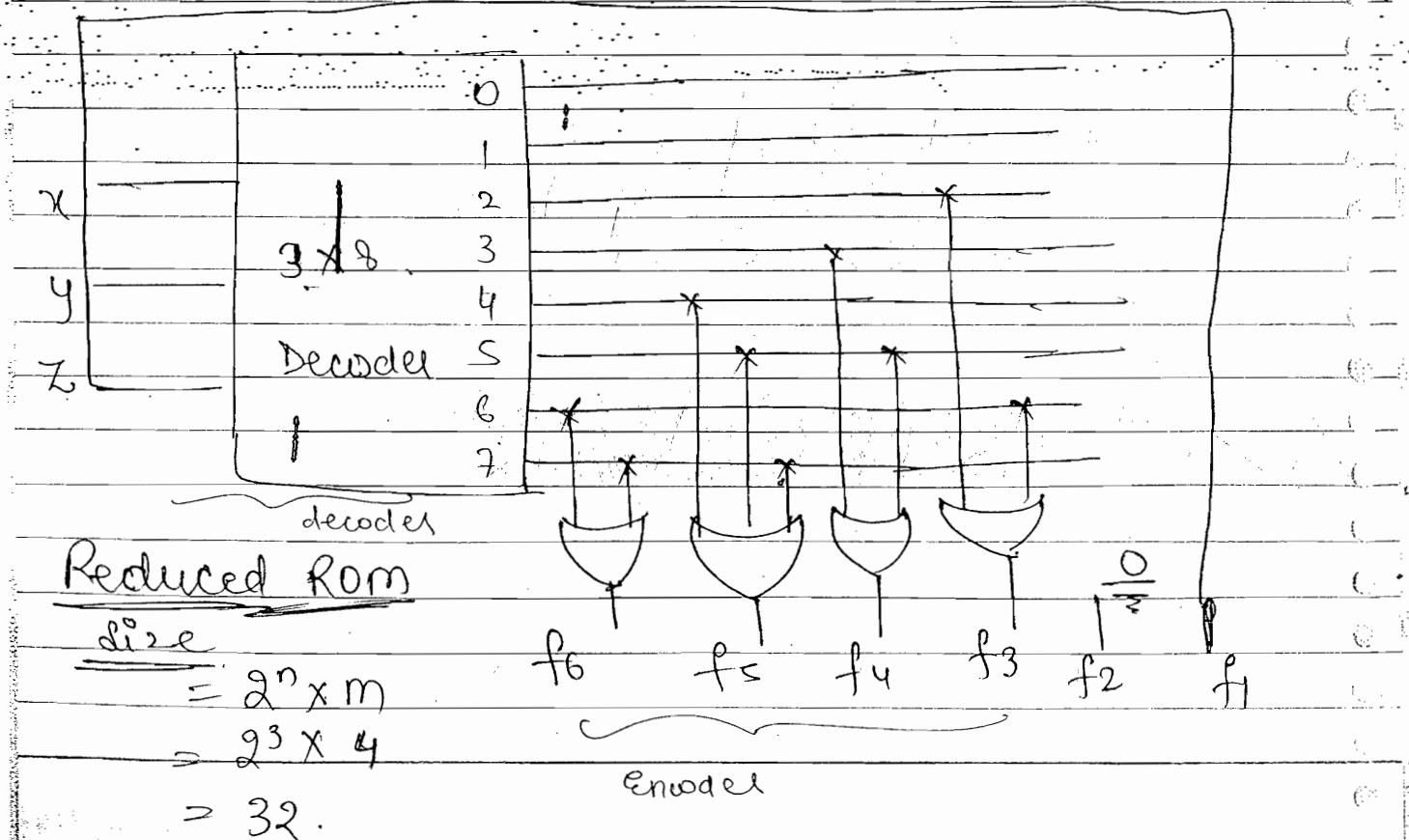
No. of data lines required = 8  
 $\Rightarrow 2^3 \times 8 = 48 \rightarrow$  Memory allocation

No. of data lines are  $m \rightarrow 6$   
 No select  $\rightarrow n \rightarrow 3$ .

$$\Rightarrow 2^n \times m \Rightarrow 2^3 \times 6 \\ \Rightarrow 48.$$

	x	y	z	$f_6$	$f_5$	$f_4$	$f_3$	$f_2$	$f_1$	
0	0	0	0	0	0	0	0	0	0	'0'
1	0	0	1	0	0	0	0	0	1	'1'
2	0	1	0	0	0	0	1	0	0	'4'
3	0	1	1	0	0	1	0	0	1	'9'
4	1	0	0	0	1	0	0	0	0	'16'
5	1	0	1	0	1	1	0	0	1	'25'
6	1	1	0	1	0	0	1	0	0	'36'
7	1	1	1	1	1	0	0	0	1	'49'

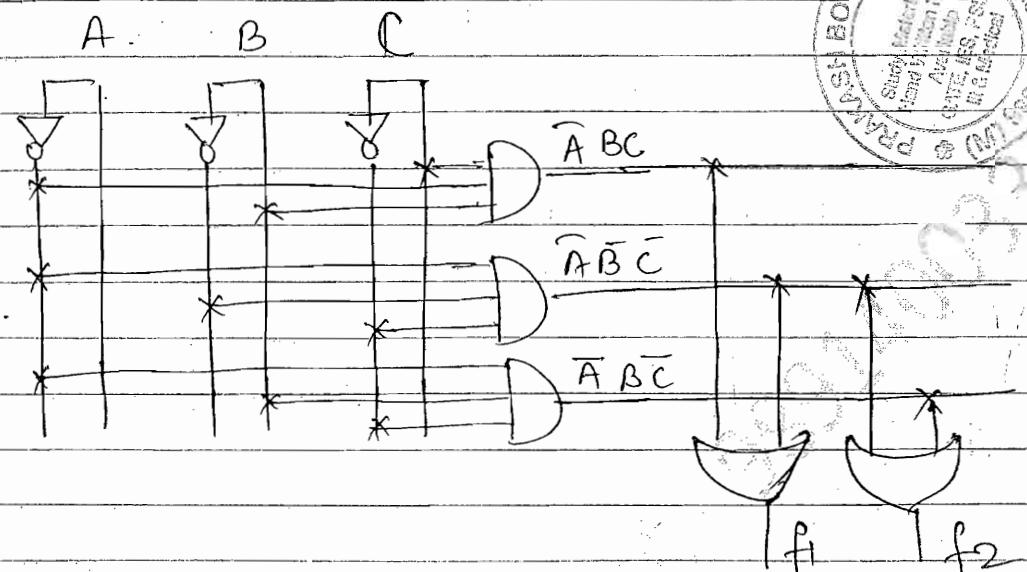
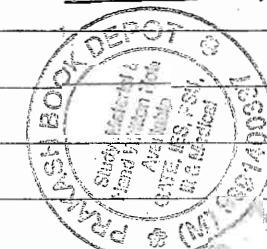
ROM  $\rightarrow$  Combination of Decoder and Encoder:



- It will generate the No. square code So encoder.

- Given.  $f_1 = \bar{A}BC + \bar{A}\bar{B}\bar{C}$   
 $f_2 = \bar{A}\bar{B}\bar{C} + \bar{A}B\bar{C}$

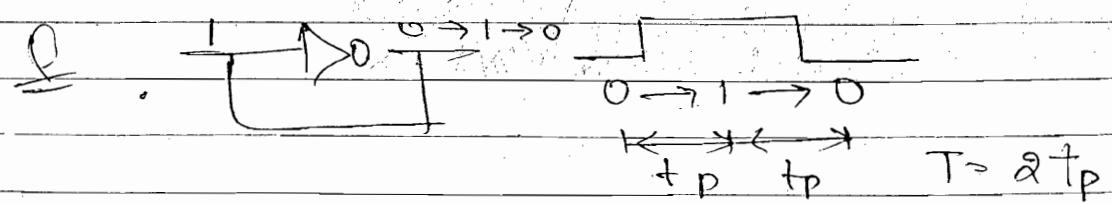
"PLA"



No. of Minterms = No. of AND gates

	AND Array	OR Array
ROM	fixed	programmable
PLA	programmable	programmable
PAL	programmable	fixed

- In PAL - OR array is fixed and AND array is variable.



$$\therefore f = \frac{1}{2tp}$$

for n gates.

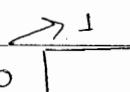
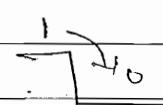
for  $N$  gates  $\Rightarrow T = N(2tp)$

$$f = \frac{1}{N(2tp)}$$

$\therefore n \rightarrow \text{odd}$  always

$n \rightarrow \text{even} \times \rightarrow \text{then is } \boxed{0} \rightarrow \boxed{0} \rightarrow \boxed{0} \rightarrow \boxed{0} \rightarrow \boxed{0} \rightarrow \boxed{0}$

### Hazard:-

- Static '0' Hazard:-  should come 0 but comes 1
- Static '1' Hazard:-  should come 1 but comes 0

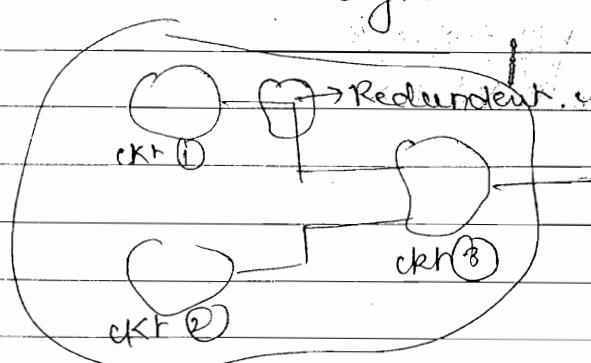
Because of different propagation delay the different part of the ckt, the o/p of the ckt is not accurate for temporary interval of time known hazard in the circuit.

For a temporary interval we are getting wrong result that result is called "Glitch".

Types:- (1) static '0' Hazard

(2) static '1' Hazard

(3) dynamic Hazard



Redundant ckt ckt ① o/p comes quickly

ckt ② o/p -- slowly

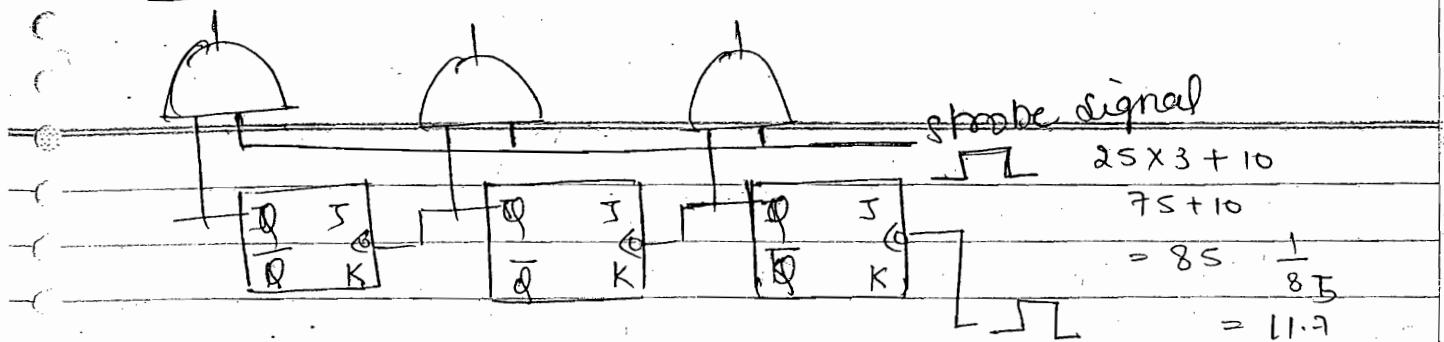
so o/p will generate by ckt ①

③ By ckt ① o/p and previous o/p of ckt ② for temporary period

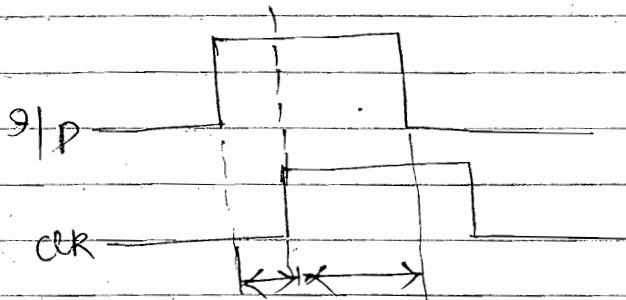
So add Redundant ckt for delay the rising of ckt ①

Now o/p of ckt ① and ckt ② comes at same timing give accurate result by ckt ②

Strobe signal is used to avoid error.



- To avoid decoding every strobe signal used.
- When the O/p is available, then strobe signal is applied to get the display.



set up. hold time  
Time

• Setup time should be added not hold time for delay.

• SOP should be applied first then CLK pulse applied.

- Mod value divided to get frequency value.

Q. as page no. 65  $\rightarrow$  If initial value 101 not given  
Ans: f/16.

Page no. 57.

Q. 19 n = 3

$$\begin{array}{c}
 \text{4x1} \\
 \text{IA, } \left( \begin{array}{c} J_0 \\ S_1 \\ S_2 \\ S_3 \end{array} \right) = 2^2 \times 1 = 2^{3-1} = (2^{n-1} \times 1) \\
 \text{B, C}
 \end{array}$$

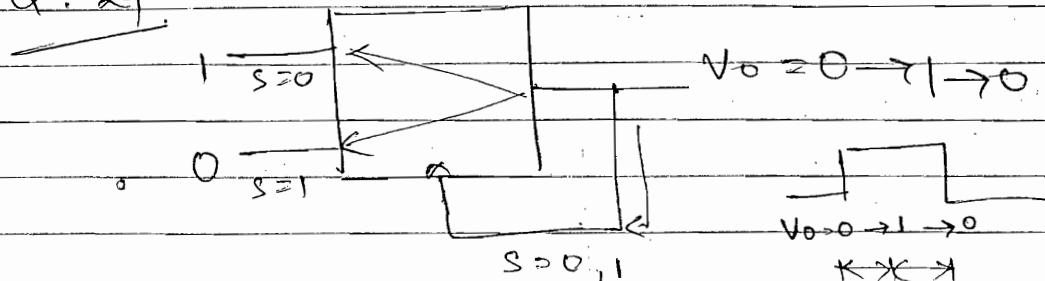
If use A, f  $\rightarrow$   ~~$\left( \begin{array}{c} J_0 \\ S_1 \\ S_2 \end{array} \right)$~~   
B, C  $\rightarrow$   ~~$\left( \begin{array}{c} J_0 \\ S_1 \\ S_2 \end{array} \right)$~~   
two multiplexers required

Q. 20 After 4 stage we get O/p = 1

do ans 11 4 correction in ansheet

MUX → waveform generator

Q. 29



$$1 \text{ msec } 1 \text{ msec.} = 2 \text{ msec.} = T_p$$

$$f = \frac{1}{2} \approx 0.5 \text{ MHz},$$

Logic families parameters [ propagation delay, N.M. (fan out etc) table in theory book is mp (Q). CMOS, TTL..

less delay  $\rightarrow$  More Speed ✓

# Number System

## Conversion

Decimal to any  $\Rightarrow$  Successive Division

## (i) Decimal to Binary

$$10^2 \quad 10^1 \quad 10^0 \quad \cdot \quad 10^{-1} \quad 10^{-2}$$

→ Defined pt

$$\boxed{73} = 7 \times 10^1 + 3 \times 10^0 = 70 + 3 = \boxed{73}$$

$$\begin{array}{r}
 \cancel{2} \mid \cancel{1} \cancel{2} \rightarrow 0 \\
 \cancel{2} \mid 6 \rightarrow 0 \\
 \cancel{2} \mid 3 \rightarrow 1 \\
 \hline
 \end{array}$$

Im. fractional

$$(0.75)_{10}$$

$$0.75 \times 2 = 1.50$$

$$p_{SO} \times 2 = 1.00$$

$$\begin{pmatrix} 1 & 2 \end{pmatrix}_{10} = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}_2$$

$$(0.75)_{10} = (0.11)_2$$

(ii) Decimal to Oct :-  $(253)_{10}$  (ii)  $\rightarrow$  To hexa

$$\begin{array}{r|rr}
 8 & 253 \\
 \hline
 0 & 31 & \rightarrow 5 \uparrow \\
 & 3 & \rightarrow 7
 \end{array}$$

$$\begin{array}{r} (351)_{10} \\ \hline 16 \mid 351 \\ \hline 10 \mid 21 \rightarrow F_1 \\ \hline 1 \rightarrow 5 \end{array}$$

$$(253)_{10} = (375)_8$$

$$(35)_{10} = (15F)_{16}$$

## Any to Decimal:

$$\text{Ans: } \textcircled{1} \quad (1101 \cdot 011)_2$$

$$\begin{array}{ccccccccc}
 1 & 1 & 0 & 1 & . & 0 & 1 & 1 & 1 \\
 2^3 & + & 2^2 & + & 2^1 & + & 2^0 & 2^{-1} & 2^{-2} & 2^{-3} \\
 8 & + & 4 & + & 0 & + & 1 & 0 & + 0.15 & + 0.15 \\
 (13.375)_{10} & & & & & & & & 
 \end{array}$$

(ii) Oct to Dec.

(753) 2

$$= 7 \times 8^2 + 5 \times 8^1 + 3 \times 8^0$$

2 (491) 10

(ii) Hexa to Dec.

(A 3 C) 16

$$= Ax^{16^2} + 3x^{16^1} + Cx^{16^0}$$

$$= (2620)$$

## • Binary to oct

$$(11101, 0111)_2$$

(75. 34) 8

## Binary to Hexa

(101101.11011)<sub>2</sub>

(2770)

• Buy to Biscay:-

(i) Och zu Bins auf

(475.4)

(100, 11101, 100),

(40) Hera to Biscay

(D A G C - 7) 16

1101101010011100.0111

Oct 10 Hex 9: -

$$\text{Ex:- } (654.3)_8 = \underline{\underline{110101}} \underline{\underline{100.011}} \\ = (1A8.6)_{16} \text{ Ans}$$

- Hexa to Octal :-

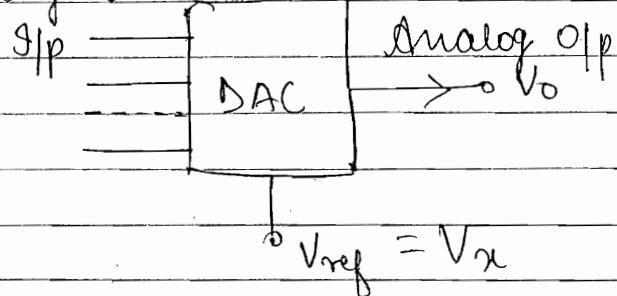
Ex:- CA93.  $\Delta$

$$\begin{array}{r} 1100 \ 1010 \ 1001 \ 0011 \cdot 110 \\ \hline (14522 \ 3 \cdot 64)_8 \text{ Ans} \end{array}$$

11/10/2014

- DAC :- parameters of DAC

Digital



$$V_o = K \left[ 2^{n-1} b_{n-1} + 2^{n-2} b_{n-2} + \dots + 2^1 b_1 + 2^0 b_0 \right]$$

Decimal eqn. Binary data :

$V_x$  full scale voltage

$V_{x6}$

$V_{x5}$

$V_{x4}$

$V_{x3}$

$V_{x2}$

$V_{x1}$

$V_{x0}$

No. of steps =  $2^n - 1$

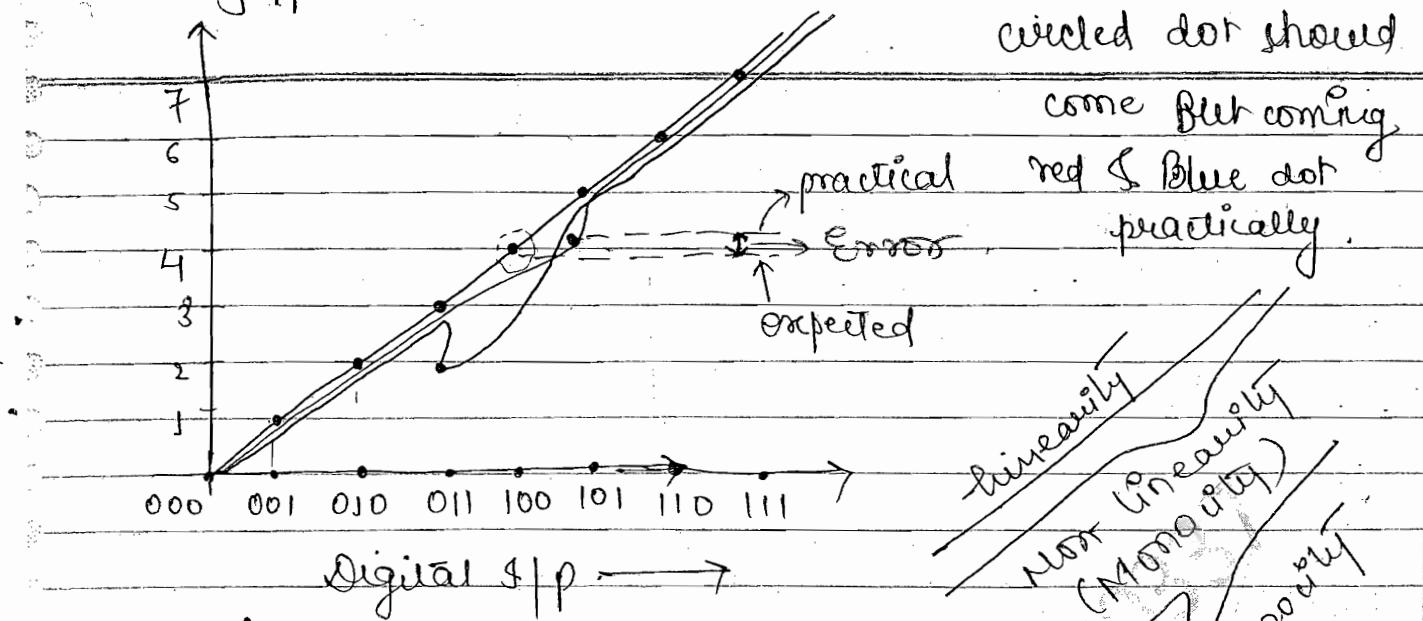
D value don't require step  
so neglected

$$\begin{aligned} \text{Step Size} &= \frac{V_x}{2^n - 1} \\ \text{or} \\ (\text{LSB value}) &= \frac{V_x}{2^n - 1} = \frac{V_{F.S.}}{2^n - 1} = \frac{V_{F.S.}}{\text{No. of steps.}} \end{aligned}$$

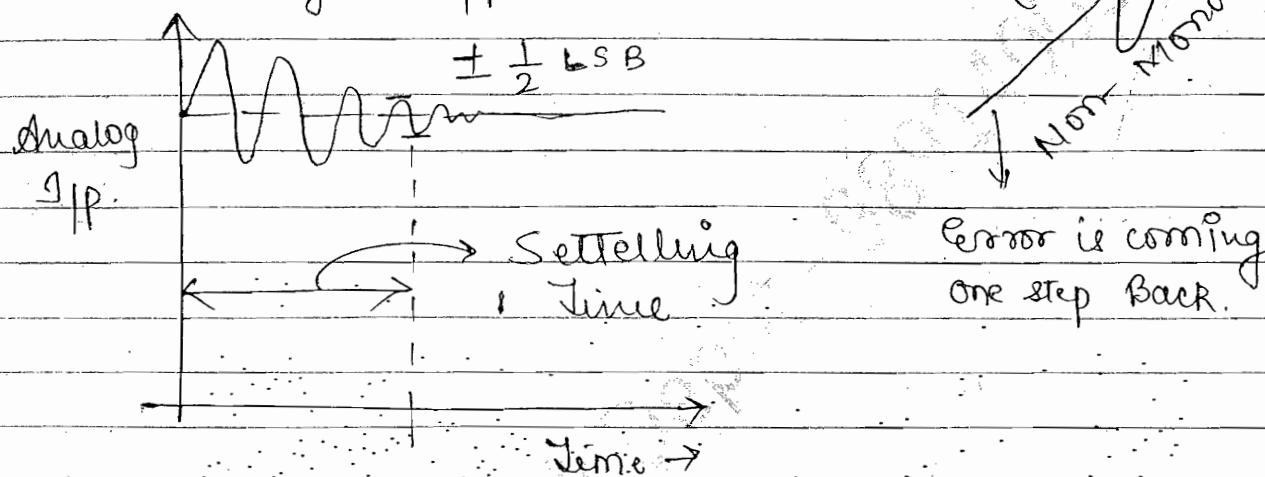
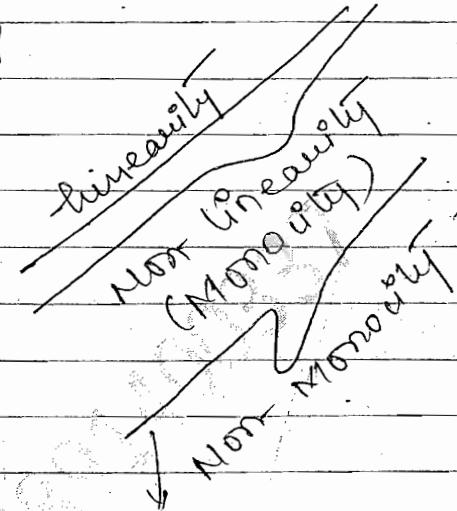
$$\% \text{ Resolution} = \frac{1}{2^n - 1} \times 100$$

Resolution:- Smallest variation it can measure having high resolution

## Analog O/p.

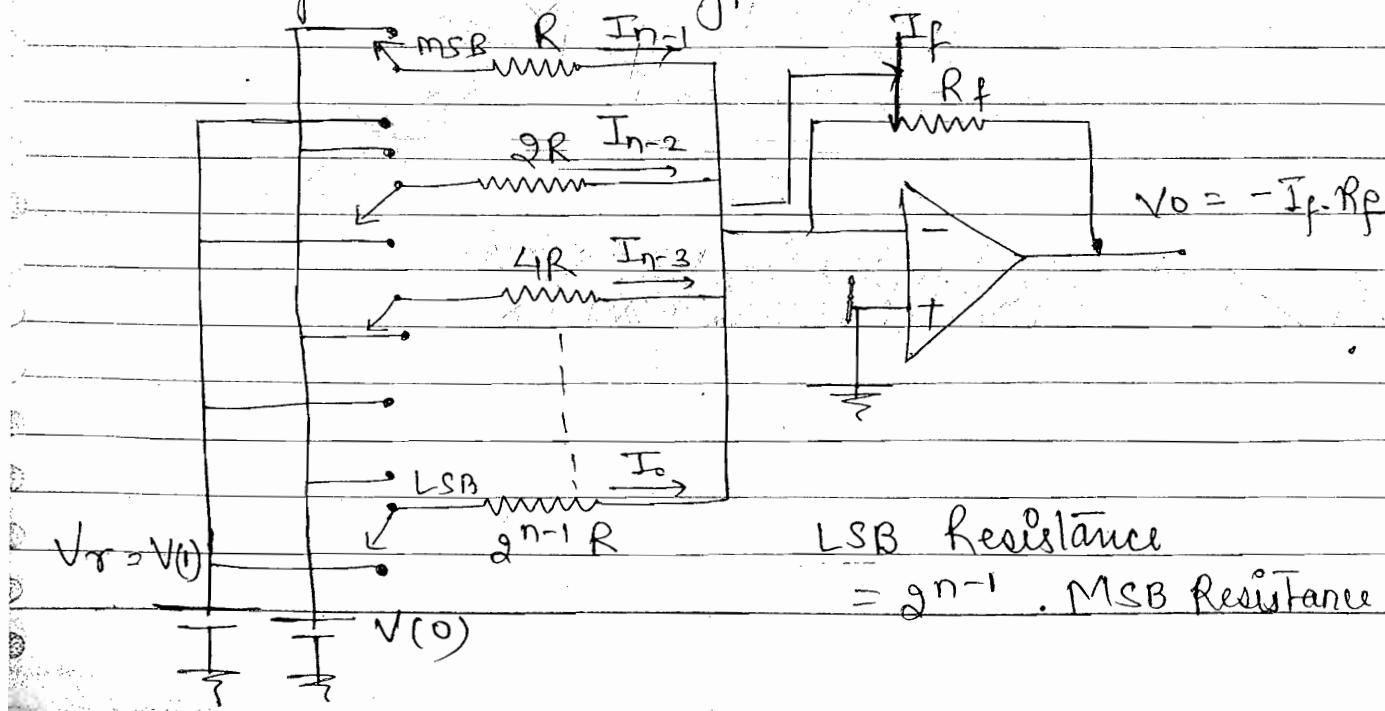


circled dot should come first coming red & blue dot practically.



Error is coming one step back.

## Weighted Resistor Type DAC



LSB Resistance

$$= 2^{n-1} \cdot \text{MSB Resistance}$$

(Assume 3 Bit DAC)

$$V_o = -I_f \cdot R_f$$

$$= -R_f [I_0 + I_1 + I_2]$$

$$= -R_f \left[ \frac{V_r}{2^2 R} + \frac{V_r}{2^1 R} + \frac{V_r}{2^0 R} \right]$$

$$= -\frac{R_f \cdot V_r}{R} \left[ \frac{1}{2^2} + \frac{1}{2^1} + \frac{1}{2^0} \right]$$

$$= -\frac{R_f \cdot V_r}{R} \left[ \frac{1}{2^{n-1}} + \frac{1}{2^{n-2}} + \dots + \frac{1}{2^1} + \frac{1}{2^0} \right]$$

$$= -\frac{R_f \cdot V_r}{R \cdot 2^{n-1}} [1 + 2^1 + 2^2 + \dots + 2^{n-2} + 2^{n-1}]$$

$$V_o = K \left[ 2^{n-1} b_{n-1} + 2^{n-2} b_{n-2} + \dots + 2^1 b_1 + 2^0 b_0 \right]$$

$$\text{where } K = \left| -\frac{R_f \cdot V_r}{R \cdot 2^{n-1}} \right|$$

- According to corresponding digital values, arrows will connect with  $V_o$ ,  $V(1)$  line and/or  $V(0)$  line. Branches will give current. According to that analog voltage comes.

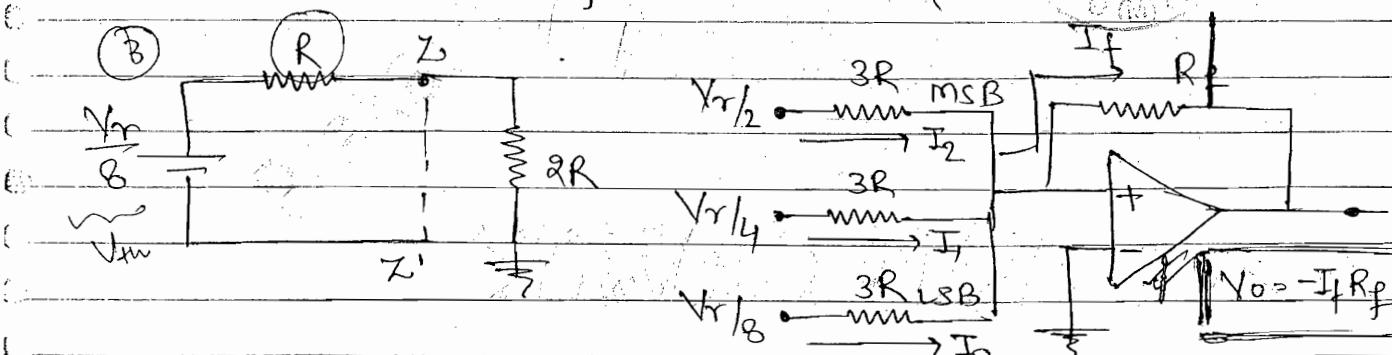
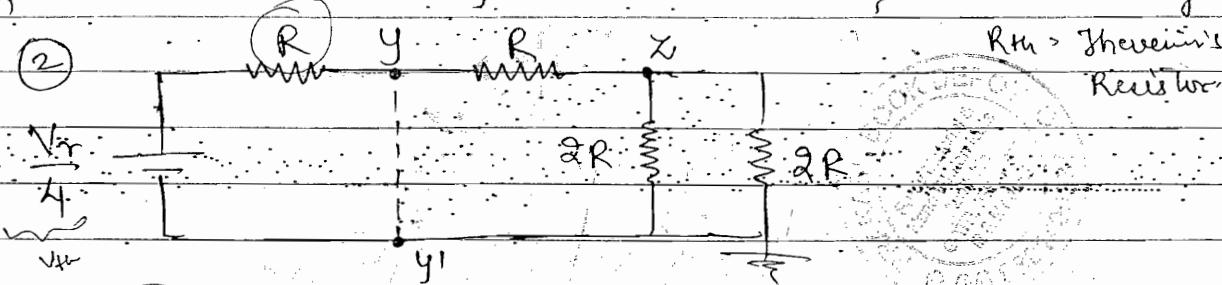
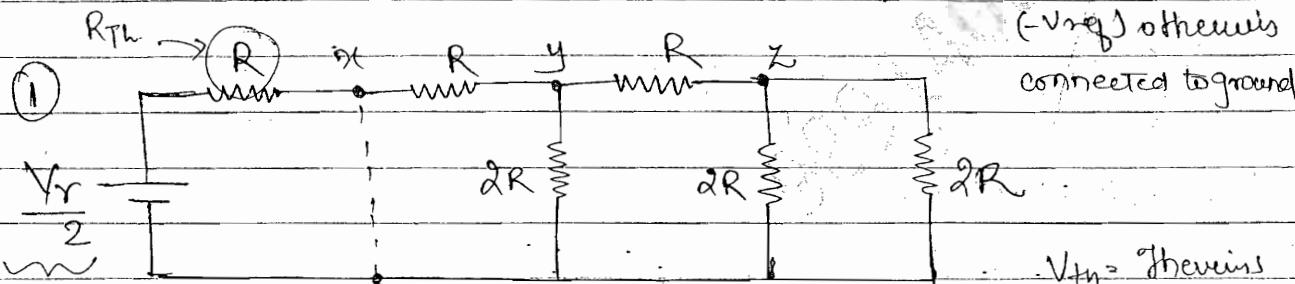
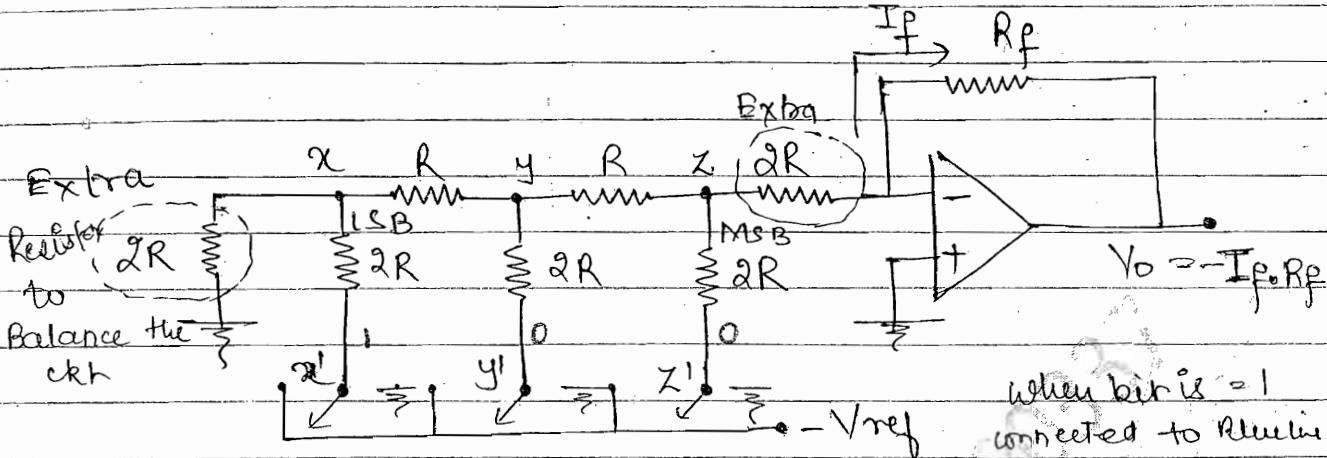
- $V(0)$  connected arrow (grounded) Branch current = 0
- $b_{n-1} \dots b_0$  - If bit(1) is present, term is present otherwise bit(0) term absent.

Q. { • LSB Resistance =  $2^{n-1}$  . MSB Resistance

•  $V_o = K \left[ 2^{n-1} b_{n-1} + 2^{n-2} b_{n-2} + \dots + 2^1 b_1 + 2^0 b_0 \right]$

•  $K = \left| -\frac{R_f \cdot V_r}{R \cdot 2^{n-1}} \right|$

## R-2R ladder Type DAC [Inverting Type].



$$Vo = -If \cdot R_f$$

$$= -R_f [I_0 + I_1 + I_2]$$

$$= -R_f \left[ \frac{V_r}{8(3R)} + \frac{V_r}{4(3R)} + \frac{V_r}{2(3R)} \right]$$

$$\begin{aligned}
 V_o &= \frac{-R_f V_r}{3R} \left[ \frac{1}{2^3} + \frac{1}{2^2} + \frac{1}{2^1} \right] \\
 &= \left[ \frac{-R_f}{3R} \right] \cdot V_r \left[ \frac{1}{2^n} + \frac{1}{2^{n-1}} + \dots + \frac{1}{2^1} \right] \\
 &= \left[ \frac{-R_f}{3R} \right] \left[ \frac{V_r}{2^n} \right] \left[ 1 + 2^1 + \dots + 2^{n-1} \right]
 \end{aligned}$$

$$V_o = K \left[ 2^{n-1} b_{n-1} + 2^{n-2} b_{n-2} + \dots + 2^1 b_1 + 2^0 b_0 \right]$$

where  $K = \left( \frac{-R_f}{3R} \right) \left( \frac{V_r}{2^n} \right)$

Suppose code is 100

$$\begin{aligned}
 V_r &\xrightarrow{\text{2R}} V_{th} \xrightarrow{\text{2R}} V_o \\
 V_{th} &= V_r \cdot \frac{2R}{2R+2R} = \frac{V_r}{2} \\
 R_{th} &= R
 \end{aligned}$$

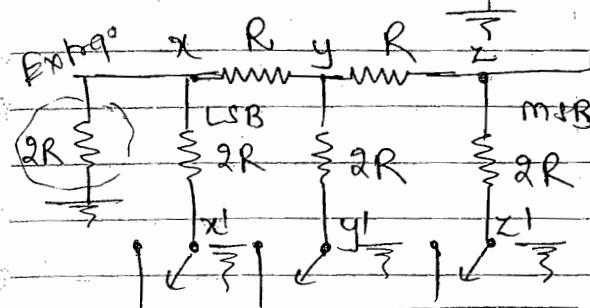
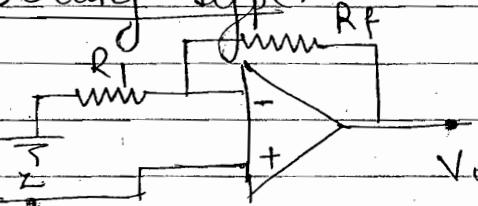
If we apply to second we get  $V_r/4$ ,  $R$  and again apply we get  $V_r/8$ ,  $R$ .

Suppose code 010 find upto  $\frac{V_r}{4}$ .

Suppose code 001 find upto  $\frac{V_r}{2}$

$$V_o = \left[ \frac{R_f}{3R} \right] \left( \frac{V_r}{2^n} \right) \left[ \sum_{i=0}^{n-1} 2^i b_i \right]$$

• For Non-inverting Type:



$$V_0 = \left[ 1 + \frac{R_f}{R_1} \right] \left[ \frac{V_r}{2^n} \right] \left[ \sum_{i=0}^{n-1} 2^i b_i \right]$$

Q.5  $V_0 = \left[ 1 + \frac{7}{1} \right] \left[ \frac{1}{2^4} \right] \left[ 1010_{(2)} = 10 \right]$

$\rightarrow 8 \cdot \frac{1}{16} \cdot 10 = 5 \text{ V}$

01010  
1010

Q.2  $V_0 = K \{ \text{Dec. equivalent Binary data} \}$

$$V_0 = K \cdot 11011011_{(2)} = 219$$

$$V_0 = K \{ 219 \}$$

$$20 = K \{ 11111111 \} = 255 \text{ full scale value}$$

$$20 = K \{ 255 \}$$

$$K = 20/255 \text{ de.}$$

Q.8

$Q_2$	$Q_1$	$Q_0$	$D_3$	$D_2$	$D_1$	$D_0$
-------	-------	-------	-------	-------	-------	-------

(b)

$$0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \rightarrow 0$$

$$0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 1 \rightarrow 1$$

$$0 \ 1 \ 0 \ 0 \ 0 \ 1 \ 0 \rightarrow 2$$

$$0 \ 1 \ 1 \ 0 \ 0 \ 1 \ 1 \rightarrow 3$$

$$1 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \rightarrow 8$$

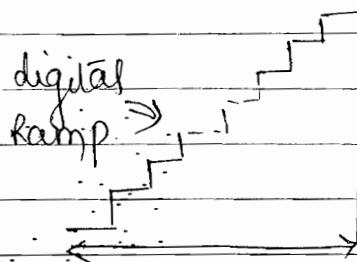
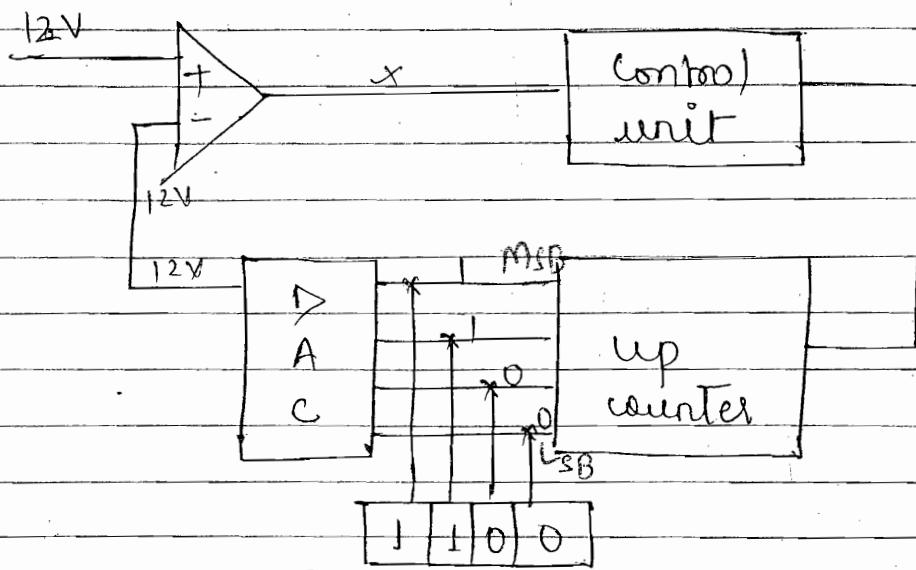
$$1 \ 0 \ 1 \ 1 \ 0 \ 0 \ 1 \rightarrow 9$$

$$1 \ 1 \ 0 \ 1 \ 0 \ 1 \ 0 \rightarrow 10$$

$$1 \ 1 \ 1 \ 1 \ 0 \ 1 \ 1 \rightarrow 11$$

## • ADC Digital Ramp Type ADC OR

### Counter Type ADC



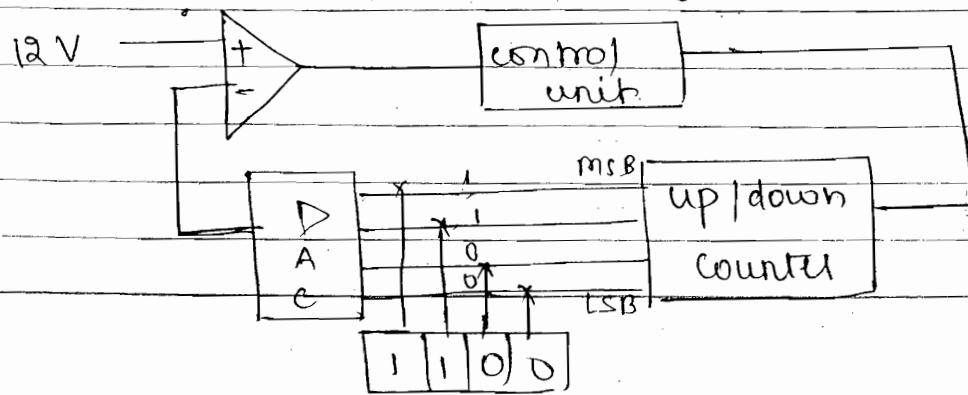
$$t_c(\max) = (2^n - 1) \text{ clk cycle}$$

conversion time ( $t_c$ )

When 12V is applied, it will display 1100.  
ex: for 4 Bit  $\rightarrow$  15 clk cycle.

\*\* Operation:- Theory Book

## • Successive Apparant type ADC



$$t_c(\max) = N. \text{ clk cycle}$$

It will compare every bit. when the voltage is -ve (greater than applied) op-amp control unit convert again  $1 \rightarrow 0$ , when voltage is +ve (smaller than applied) op-amp control unit remain  $1 \rightarrow 1$

4 Bit  $\rightarrow$  4 times comparison.

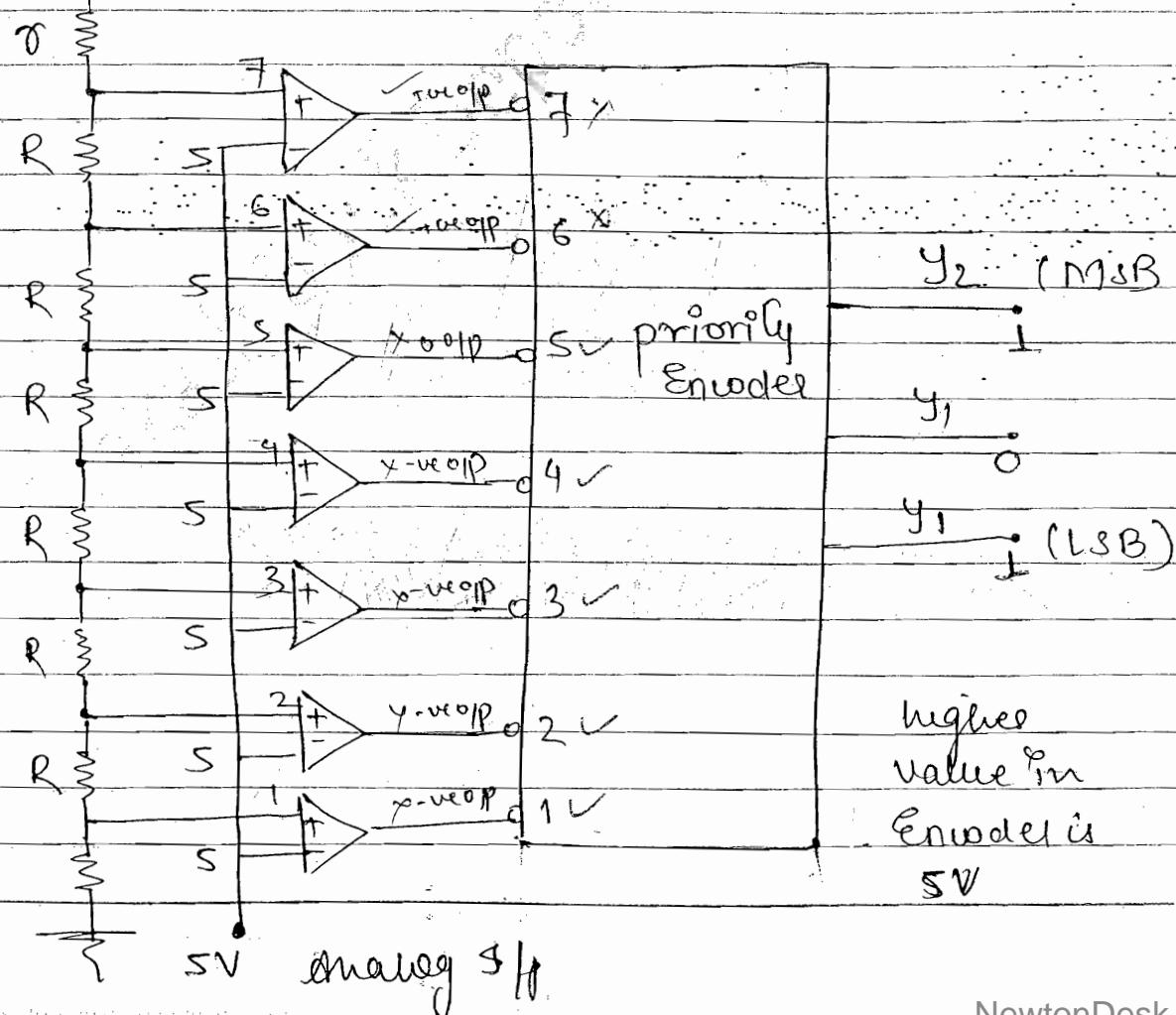
operation:- Theory Book.

Initially 1 msb  
0  
0 vrb.

• Flash Type ADC (OR) Simultaneously ADC (OR)

Parallel Type ADC

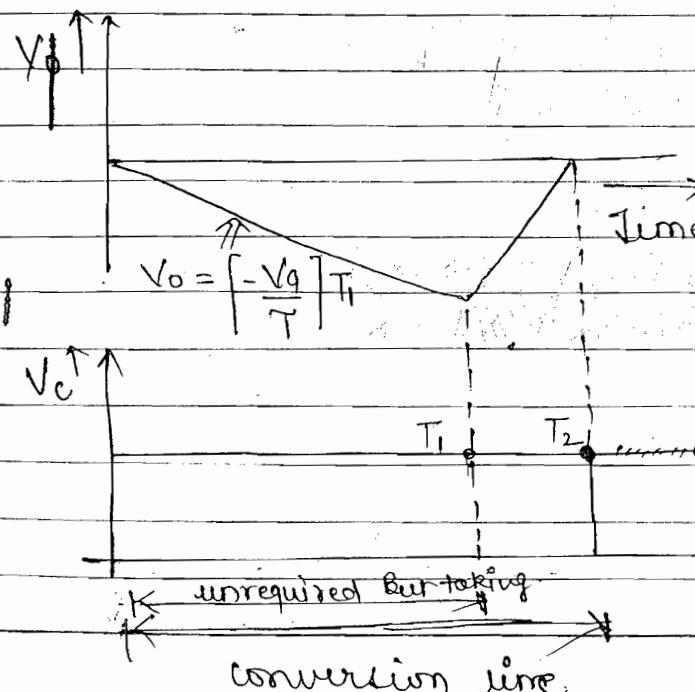
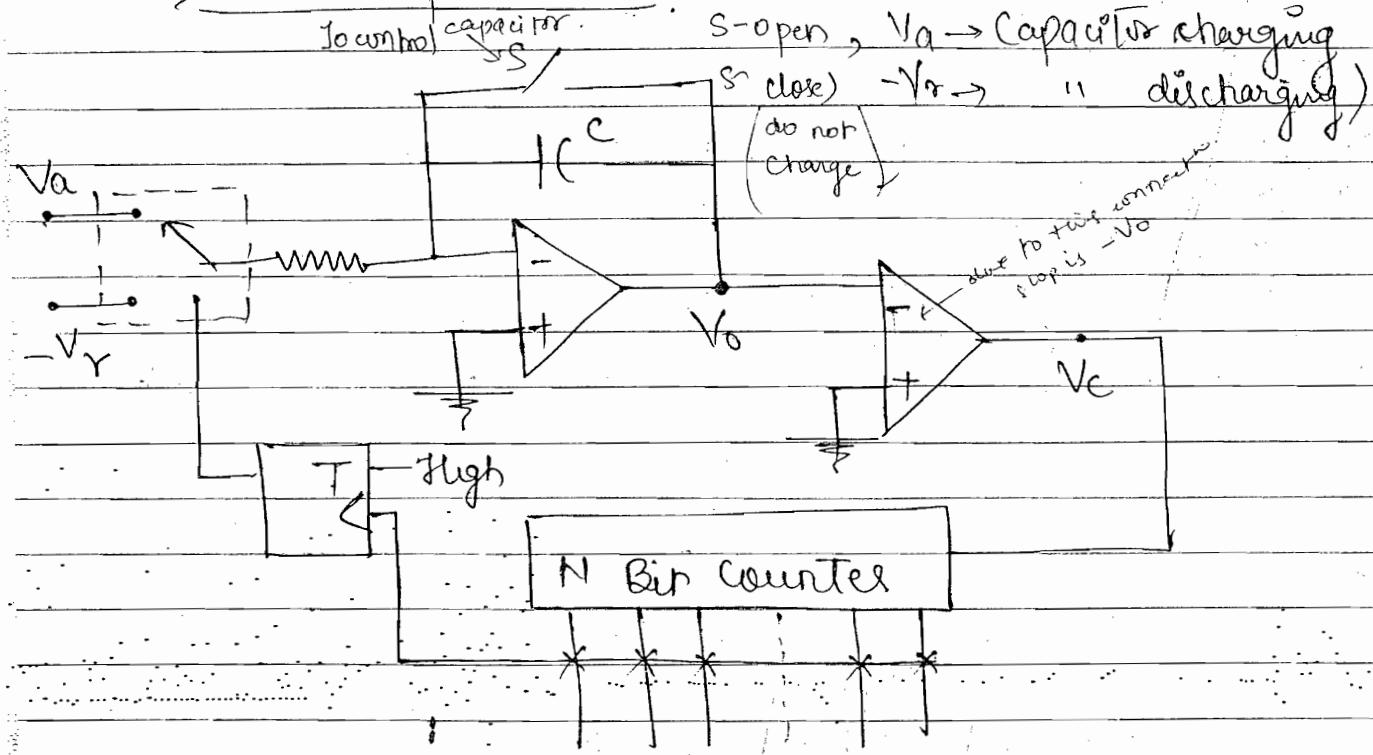
$$+V_r = 7V$$



$$t_c(\max) = 1 \text{ clk cycle}$$

$$\text{No. of comparators} = 2^n - 1$$

### Dual Slope ADC



$$T_1 = 2^N \cdot T_c$$

$$T_2 - T_1 = n \cdot T_c$$

$N \Rightarrow$  No. of Bits

$n \Rightarrow$  No. of counts

$$V_o = -\frac{1}{T} \int_0^{T_1} (V_a) dt$$

$$= - \left[ \frac{V_a}{T} \right]_{T_1}^T \quad \text{--- ①}$$

$$V_o = - \left[ \frac{V_a}{T} \right] T_1 + \left[ -\frac{1}{T} \int_{T_1}^t (-V_r) dt \right]$$

$$= - \left[ \frac{V_a}{T} \right] T_1 + \left[ \frac{V_r}{T} \right] [t - T_1]$$

$$V_o = 0 \text{ at } t = T_2$$

$$0 = \left[ -\frac{V_a}{T} \right] T_1 + \left[ \frac{V_r}{T} \right] [T_2 - T_1]$$

$$\left[ \frac{V_a}{T} \right] T_1 = \left[ \frac{V_r}{T} \right] [T_2 - T_1]$$

$$V_a \cdot 2^N \cdot T_c = V_r \cdot n \cdot T_c$$

$$V_a = V_r \cdot \frac{n}{2^N}$$

$$\text{if } V_r = 2^N \quad | V_a = n$$

$$t_c(\max) = T_2$$

$$t_c(\max) = T_1 + n T_c$$

$$= 2^N T_c + n T_c$$

$$t_c(\max) = (2^N + n) T_c$$

$$t_c(\max) = (2^N + 2^N) T_c$$

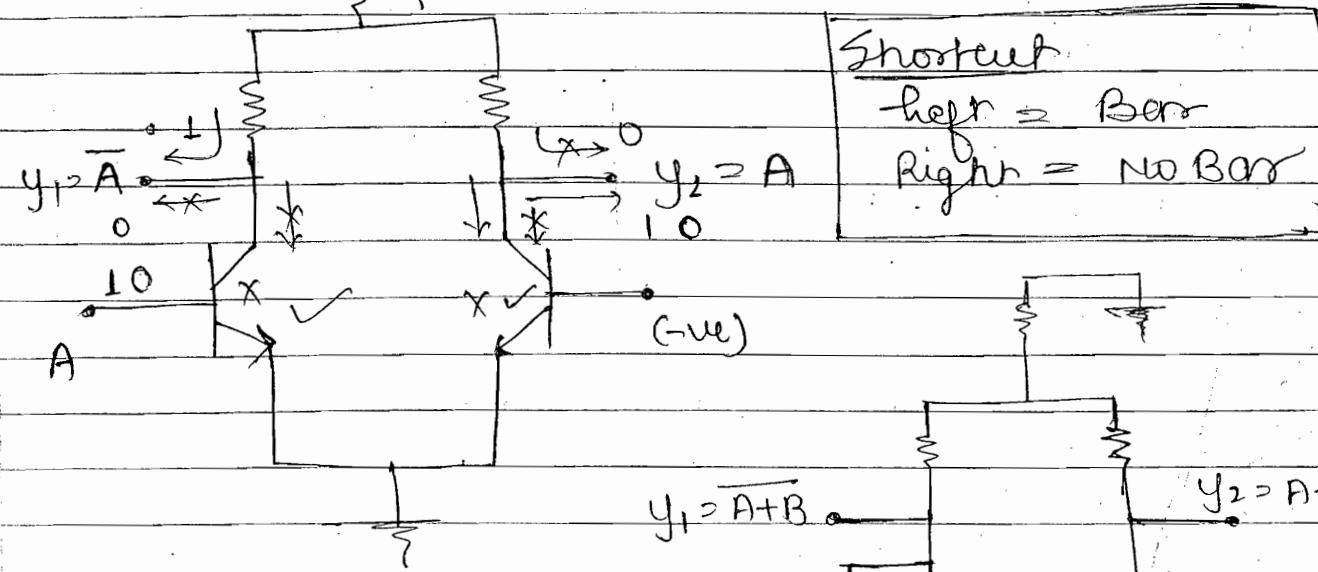
$$t_c(\max) = (2^{N+1}) T_c$$

$$t_c(\max) = (2^{N+1}) T_c$$

Disadvantage :- Even when the capacitor get charged  
counted <sup>still</sup> counts upto Max. value 1111

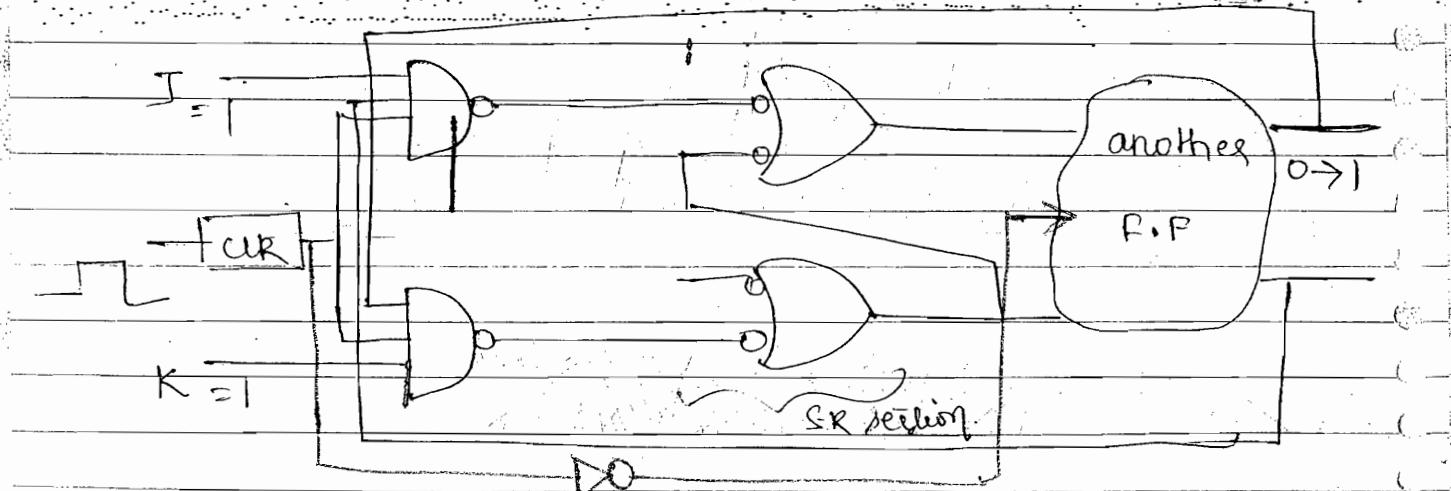
• After 9th T FF bigger and arrow get connected to  $V_r$

• ECL :- (Emitter coupled logic)



✓ A and B are Reel  
so OR operation.

• MASTER SLAVE :-



It consists of two section MASTER and slave  
connected from same clock generator But  
the second section is operated by NOT  
CLOCK

So both the section cannot be trigger simultaneously and the final o/p occurrence time - time CLR pulse is in OFF state So there is No toggle repetition and race around problem is eliminated  
external

NOTE :- feedbacks are taken from the final o/p only.

End

