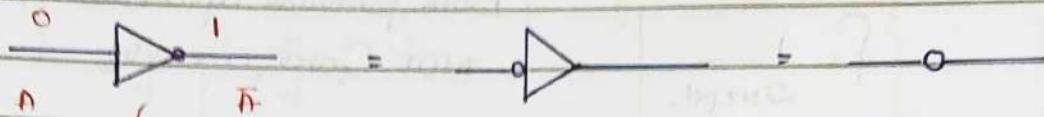


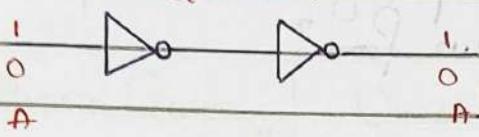
LOGIC GATES

1. NOT Gate [Inverter, Negation, Complement logic].



Propagation delay [τ_{pd}].

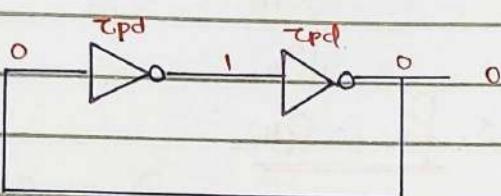
Buffer (for even)



No. of not gates are even - Buffer.

No. of not gate odd - Inverter.

[High.]



$2\tau_{pd} \quad 2\tau_{pd} \quad 2\tau_{pd} \quad 2\tau_{pd}$

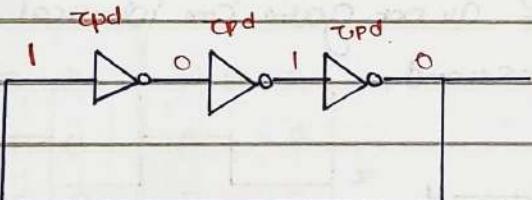
$2\tau_{pd} \quad 2\tau_{pd} \quad 2\tau_{pd}$

$0 \rightarrow 0 \rightarrow 0 \rightarrow$

[Low]

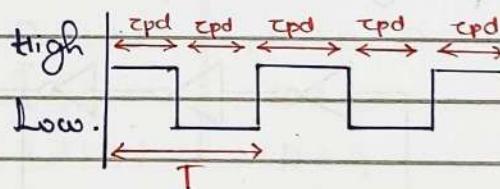
When even no. of 'not' gate in circuit (Loop).

1. Basic memory element
2. Bistable multivibrator
3. DC Generator ($f = 0 Hz$)



$2\tau_{pd} \quad 2\tau_{pd} \quad 2\tau_{pd} \quad 2\tau_{pd} \quad 2\tau_{pd}$

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



When Odd no. of NOT Gate:

1. Astable multivibrator
2. Square wave generator
3. Clock generator
4. Free running Circuit
5. Ring oscillator.

$$T = 2\tau_{pd}$$

$$f = \frac{1}{2\tau_{pd}}$$

$$f = \frac{1}{T}$$

$$T = 2 \times N \times \tau_{pd}$$

N = no. of not gate in loop.

$$f = \frac{1}{2N\tau_{pd}}$$

* Both formulae applicable when number of NOT Gate are odd.

Q1. For the circuit given below, all NOT Gate are identical to each other and having Propagation delay 10 ps . Find the frequency of generated wave form?

$$N = 5.$$

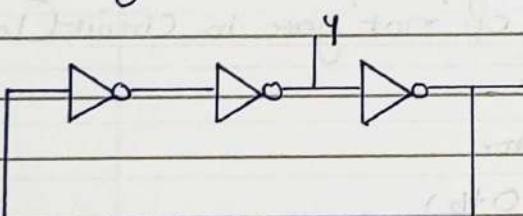
$$\tau_{pd} = 10 \times 10^{-12}$$

$$f = \frac{1}{2N\tau_{pd}} = \frac{1}{2 \times 5 \times 10 \times 10^{-12}} = \frac{10^12}{10 \times 10^9 \text{ Hz}}$$

(odd).

$$\therefore f = 10 \text{ GHz}$$

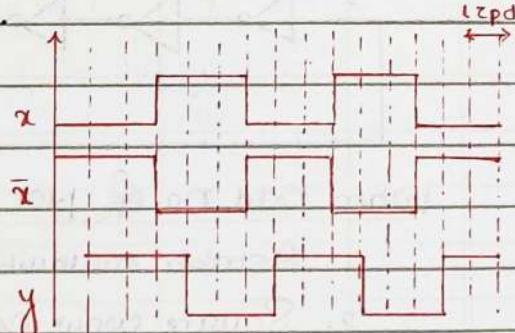
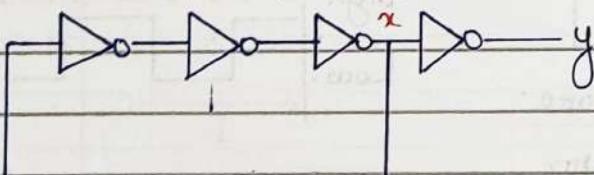
Q2. Circuit given below are called. Astable Multivibrator.



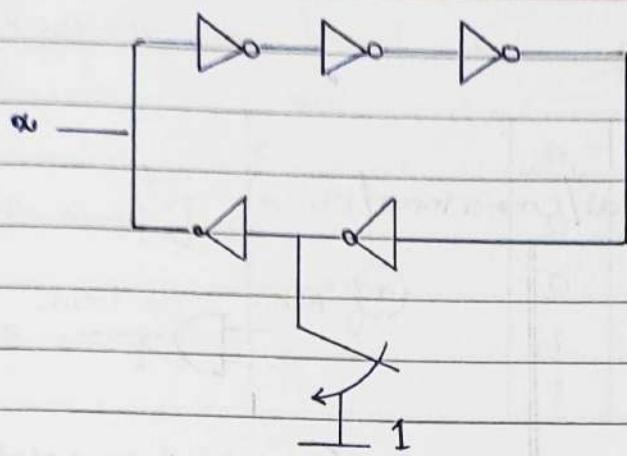
Note: always, For calculation of frequency consider the no. of not gate in loop.

HW

Q3. Sketch the waveform of y if all not gate are identical and having propagation delay of 1 microsecond.



Q4. For the circuit given below 2 only conditions will be - x and y both toggle



Odd no. of Not gate in the loop
 \therefore x and y will toggle

Basic Gate.

NOT

AND

OR

Universal Gate

NAND

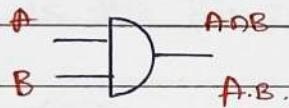
NOR

Exclusive Gate.

X-OR

X-NOR

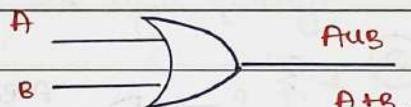
AND Gate.



$$A = \{1, 2, 3, 4\} \quad B = \{2, 3, 4\}$$

$$A \cap B = \{2, 3, 4\}$$

OR Gate



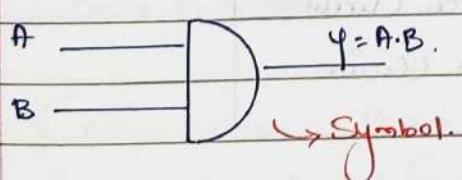
$$A = \{1, 2\} \quad B = \{3, 4\}$$

$$A \cup B = \{1, 2, 3, 4\}$$

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

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

AND Gate. [Intersection]

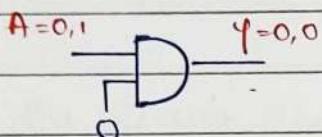


A	B	$Y = A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

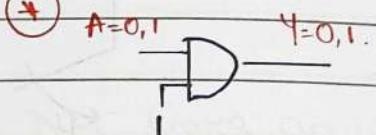
Truth Table



floating terminal/Control Input/Strobe



Control '0' disabled.

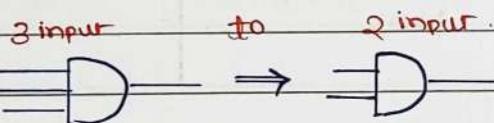


Control '1' enabled.

* Commutative Law : $A \cdot B = B \cdot A$

* Associative Law : $(A \cdot B) \cdot C = A \cdot (B \cdot C)$

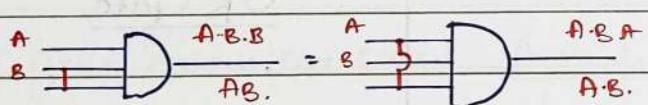
} AND is both commutative and associative.



TTL = Transistor Transistor

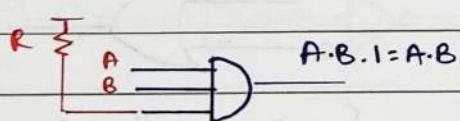
Logic.

1.



{ Fanout ↓ }

2.



pull up arrangement.

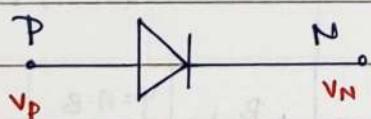
{ Costly }

3.



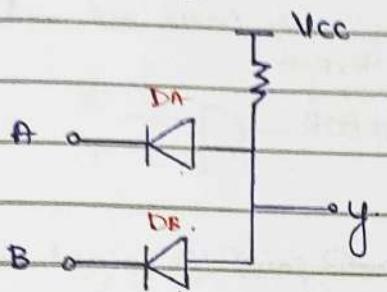
{ Noise margin ↓ }

"Diode".



$V_p > V_N \rightarrow$ Forward bias \rightarrow Shorted Circuit.

$V_p < V_N \rightarrow$ Reverse bias \rightarrow Open circuit.

Circuit diagram: [AND]

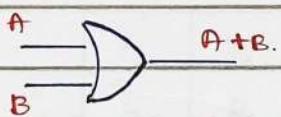
A	B	D _A	D _B	y
0	0	F _B	F _B	0
0	1	F _B	R _B	0
1	0	R _B	F _B	0
1	1	R _B	R _B	1

Note:

- * Whenever logic are designed by TTL the floating terminal always works as high.
- * Whenever logic are designed by ECL (Emitter Coupled Logic) then floating terminal always works as low.
- * Noise Margin: Maximum noise added to the input which will not affect the output are called Noise margin.
- * Fan out: The number of logic drivers by the logic are called fan out.
- * Fan in: Number of inputs of a logic are called fan in.
- * ECL is the fastest logic among all the logic family.

OR Gate.

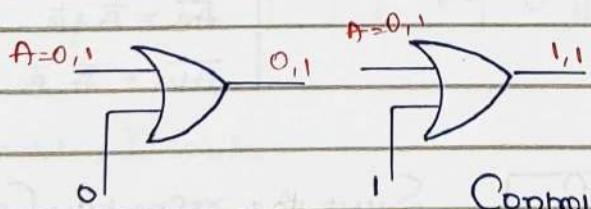
1. Symbol.



2. Truth Table.

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

3. Enable/Disable



Control '1' disabled.

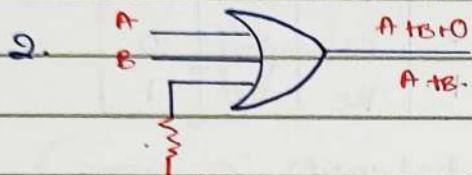
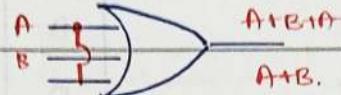
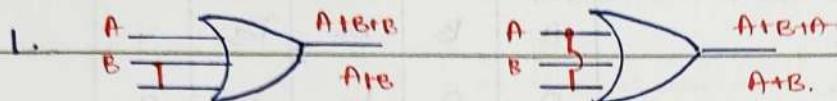
Control '0' enabled.

4. Commutativity: $A + B = B + A$

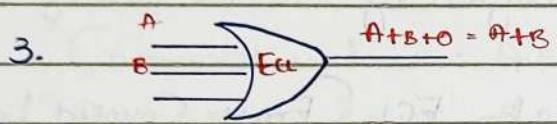
5. Association: $A + (B+C) = (A+B)+C$



ECL - Emitter Coupled Logic.

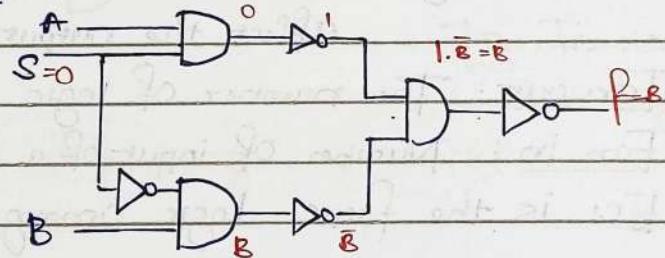


Pull down arrangement.

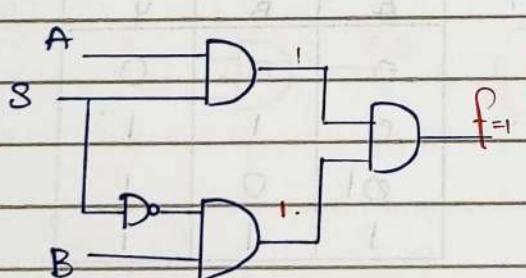


Q1. Output at $S=0$ is \underline{B}

$$\rightarrow f = \underline{B}$$



Q2. $f=1$ for combination of A, B, S



[None of the options are correct.]

De Morgan's Law

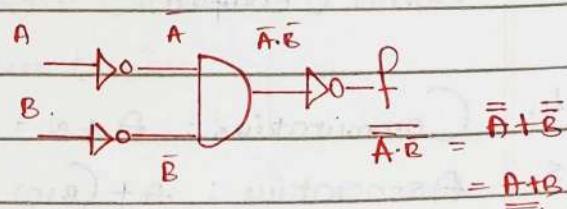
$$\overline{AB} = \overline{A} + \overline{B}$$

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

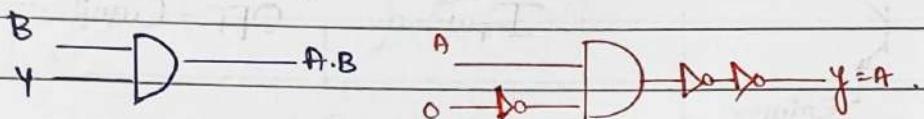
Q3. For the given truth table.

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

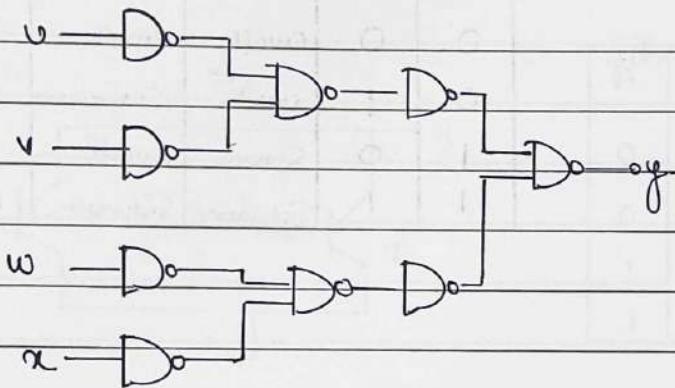
Since the respective Combinational



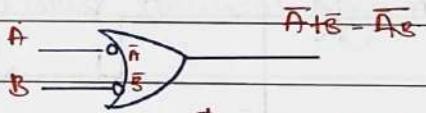
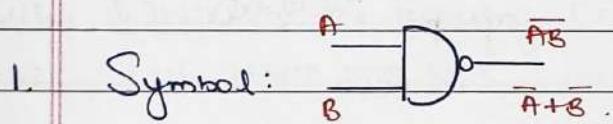
~~HW~~ Q4. A logical circuit is as shown below, which of the following circuit can be used to get the desired expression.



Q5. The logic circuit shown below, is equivalent to.



NAND Gate.



Bubbled OR = NAND

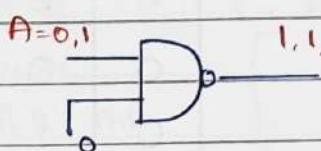
2. Truth Table.

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

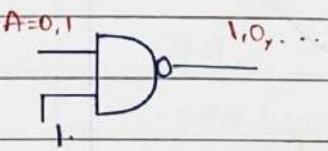
4. Commutative law. ✓

5. Association Law. ✗

3. Enable / Disable.



$A=0,1$ 1,1,1...



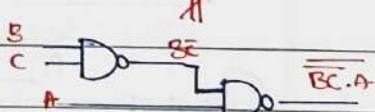
$A=0,1$ 1,0,...

Control '0' disabled.

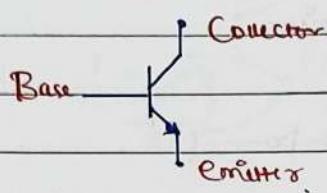
Control '1' enabled



Association ✗

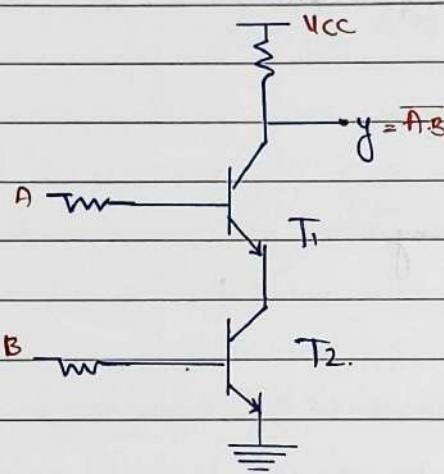


NAND follows commutative law but does not follow association law.

Transistor

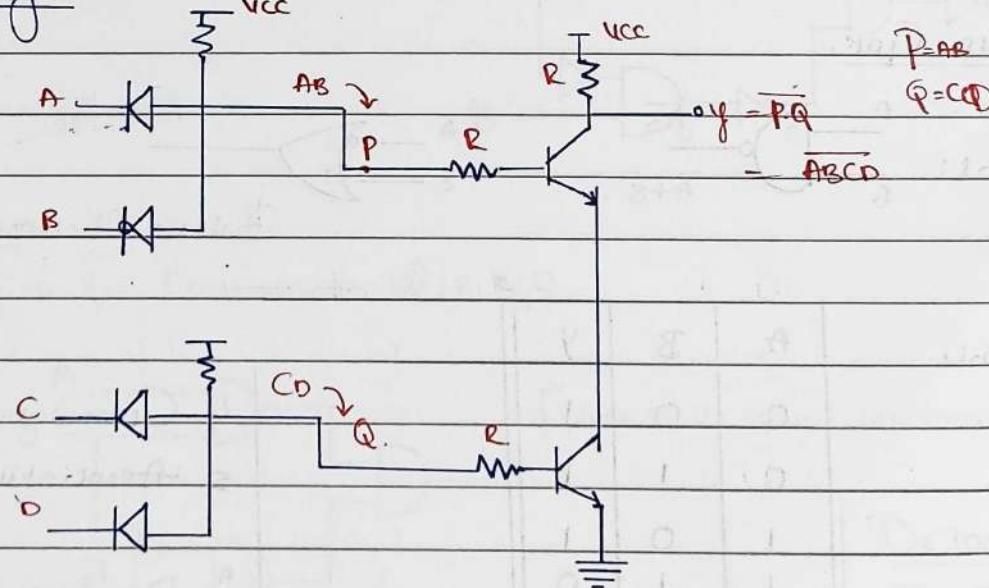
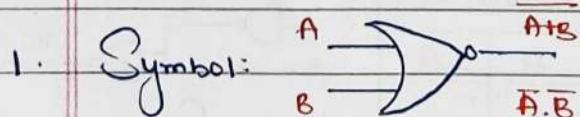
Input = 1

Input = 0

ON = Saturation \rightarrow Short CircuitOFF = Cutoff \rightarrow Open Circuit

A	B	T ₁	T ₂	y
0	0	Cutoff	Cutoff	1
0	1	Cutoff	Saturation	1
1	0	Saturation	Cutoff	1
1	1	Saturation	Saturation	0

eg::

NOR Gate.

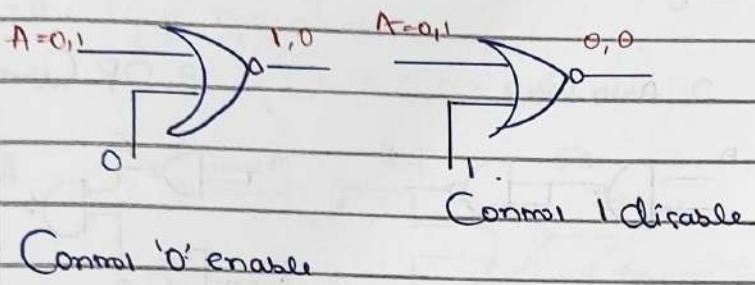
$$\begin{array}{ccc} A & \text{---} & \overline{A} \\ B & \text{---} & \overline{B} \end{array} \quad \overline{A} \overline{B} = \overline{A+B}$$

Bubbled AND = NOR

2. Truth Table:

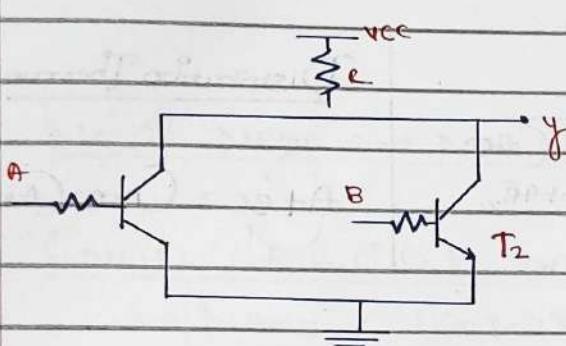
A	B	y
0	0	1
0	1	0
1	0	0
1	1	0

3. Enable/Disable



4. Commutation Law ✓

5. Association Law. ✗



A	B	T ₁	T ₂	y
0	0	Cutoff	Cutoff	1
0	1	Cutoff	Saturation	0
1	0	Saturation	Cutoff	0
1	1	Saturation	Saturation	0

Note: NAND, NOR are called "Universal Logic".

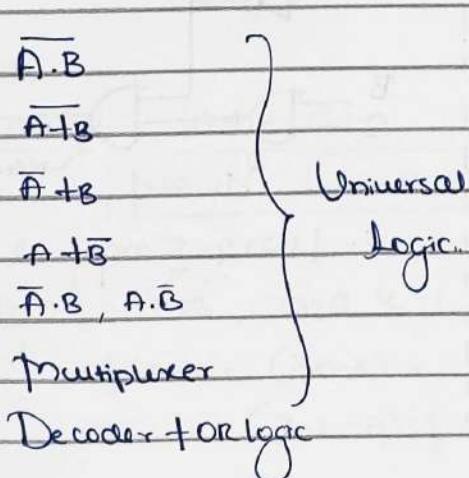
Q1. Which of the following option is called universal logic?

Ans. Both NAND and NOR. (Options given)

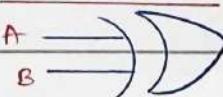
Q2. Which of the following is called Universal logic? [MCQ]

Ans. NAND (if both are given choose NAND)

Q3. Which of the following Option(s) is/are called universal logic? [MCQ]

Ans. $(\bar{A}B)$, $(\bar{A}\cdot B)$, $(A+\bar{B})$, $A\cdot \bar{B}$.X-OR Gate

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

X-NOR Gate

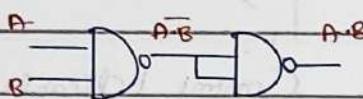
$A \odot B = \overline{A\overline{B} + \overline{A}B}$

NAND as universal logic

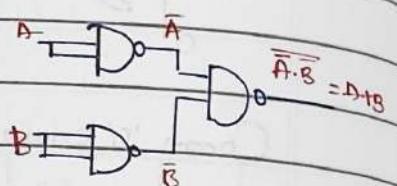
1. NOT Gate.



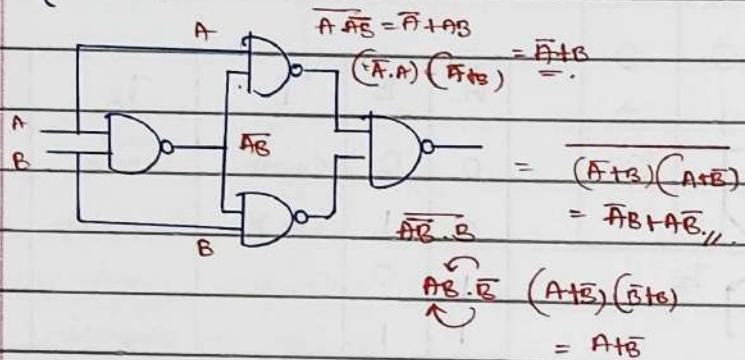
2. AND Gate



3. OR Gate



4. XOR Gate

Distribution Theorem

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

5. X-NOR Gate

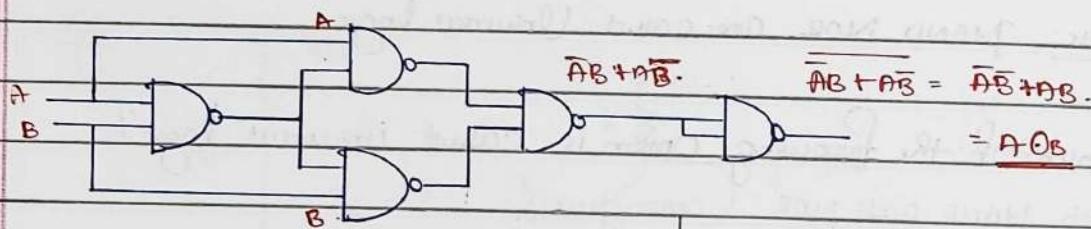


Table of

no. of

NAND and

NOR gate

required to

make the

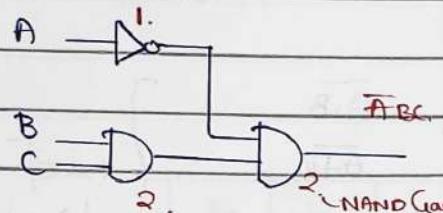
given Gate.

		NAND	NOR
NOT	1	1	
AND	2	3	
OR	3	2	
X-OR	4	5	
X-NOR	5	4	
NAND	1	4	
NOR	4	1	

Q1. Find the minimum no. of two

input NAND Gate required to
implement the logic given below.

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

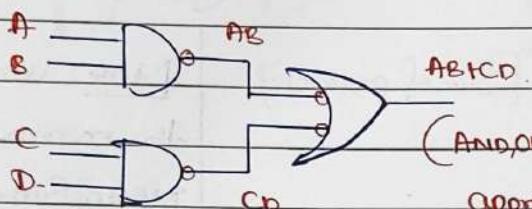


$$\therefore 1+2+2 = 5 \text{ required}$$

NAND Gate required

Q2 Find the minimum no. of two input NAND Gate required to implement the logic given below.

$$f(A, B, C, D) = AB + CD.$$



Create using AOT approach and make it with NAND by adding inverter (' $\bar{}$ ') to balance out!

$\therefore 3$ NAND Gate required.

No. of NAND and NOR Gate Required:

Case(1) : $A \cdot B \cdot \bar{C} \cdot D \cdot E \dots$

$$\rightarrow \text{NAND} = (2n-2)+k$$

n = total no. of variables

k = no. of Complement

Variables.

$$\rightarrow \text{NOR} = (3n-3)\bar{k}$$

n = no. of variables

k = no. of Complement

Variables.

$$\text{eg.: } f = A \cdot B \cdot C$$

$n=3, k=1$.

$$\text{NAND} = (2n-2)+k$$

$$= (2 \times 3 - 2) + 1 = 5$$

$$\text{NOR} = (3n-3)\bar{k}$$

$$= (3 \times 3 - 3) \bar{1}$$

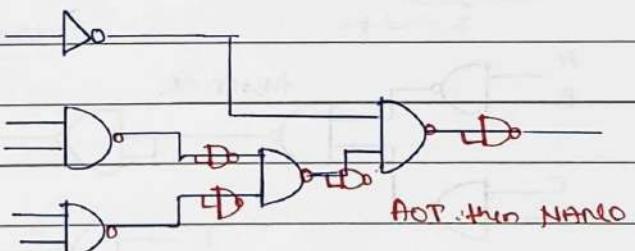
$$= 6 - 1 = 5$$

$$\text{eg. } f = \bar{A} \bar{B} \bar{C} \bar{D} \bar{E} \dots$$

$n=5, k=1$.

$$\text{NAND} = (2n-2)+k$$

$$= 9$$



Case(2) : $A + B + \bar{C} + D \dots$

$$\text{NAND} = (3n-3)-k$$

$$\text{NOR} = (2n-2)+k$$

$$\text{eg. } f = \bar{A} + B, n=2, k=1$$

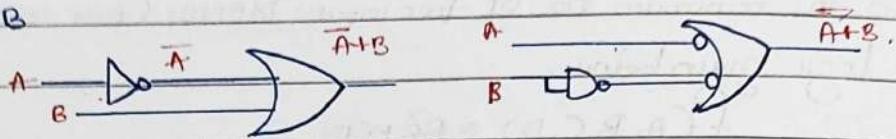
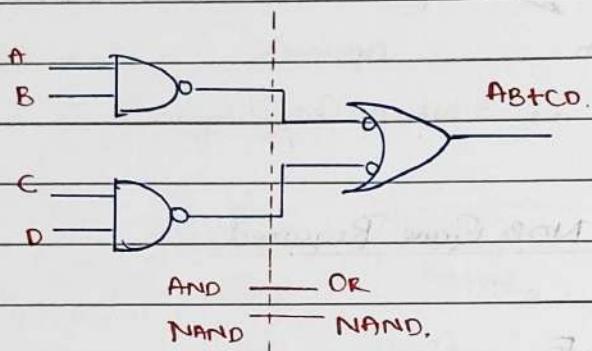
$$\text{NAND} = (3n-3)-k$$

$$= (3 \times 2 - 3) - 1 = 2$$

$$\text{NOR} = (2n-2)+k$$

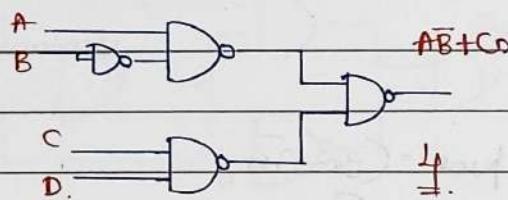
$$= (2 \times 2 - 2) + 1$$

$$= 3$$

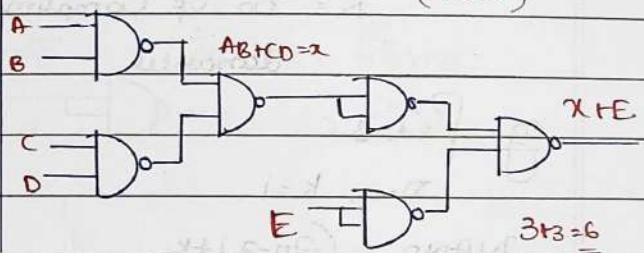
eg: $\overline{A} + B$ Case (3) $f = AB + CD$. [Sum of product]

Note: Whenever in the problem the minimum number of NAND Gate are asked then write in SOP form and implement it by AOI and this is exactly equal to NAND implementation.

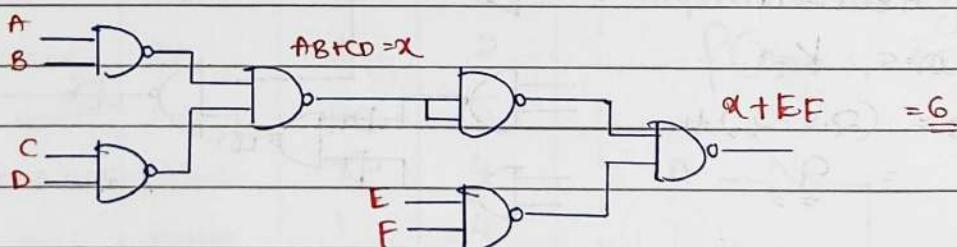
Q. $f = A\bar{B} + CD$.



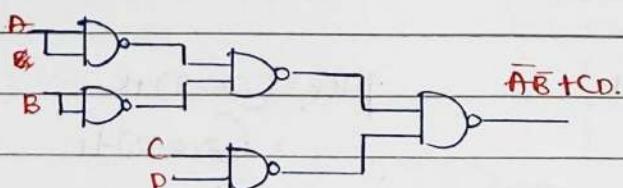
$f = AB + CD + E$. f_{x+E}
($AB + CD$)



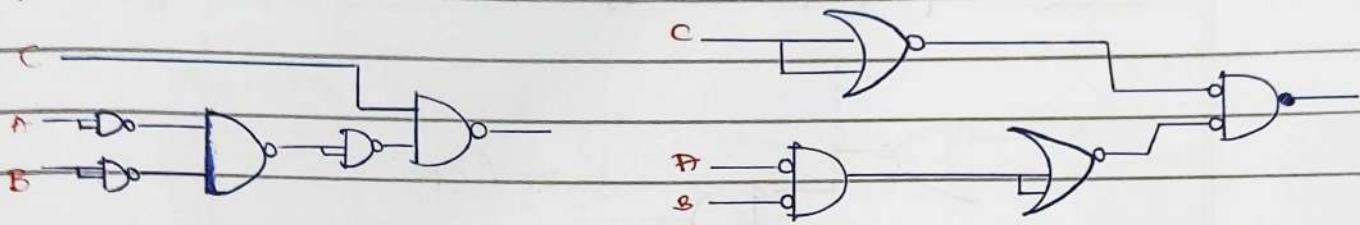
Q. $f = \underbrace{AB + CD}_{x \rightarrow 3+3} + EF$



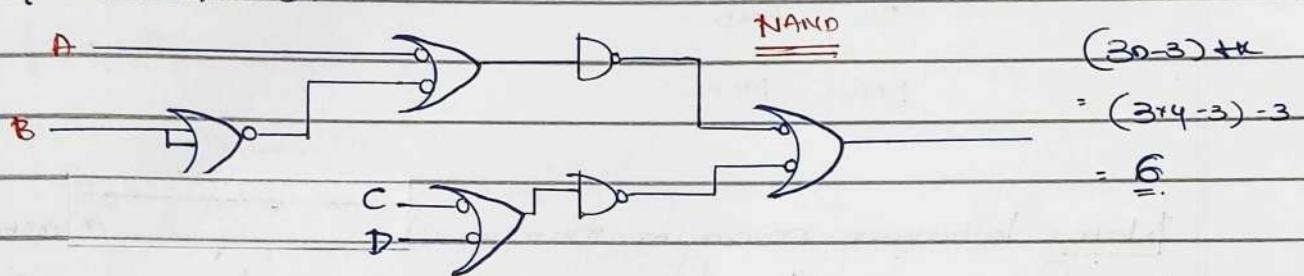
Q. $\overline{A}\bar{B} + CD$



Q1. $f = \overline{ABC}$.



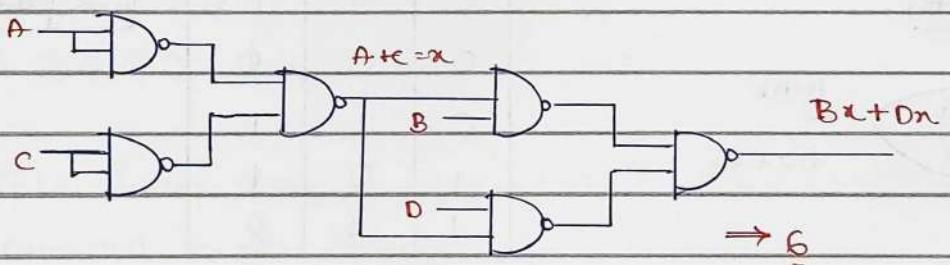
Q2. $f = \overline{A} + B + \overline{C} + \overline{D}$.



Q3. $f = AB + BC + CD + DA$

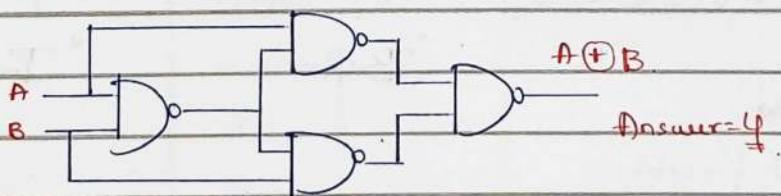
$$f = B(A+C) + D(A+C)$$

$$f = Bx + Dx, \quad A+C=x.$$

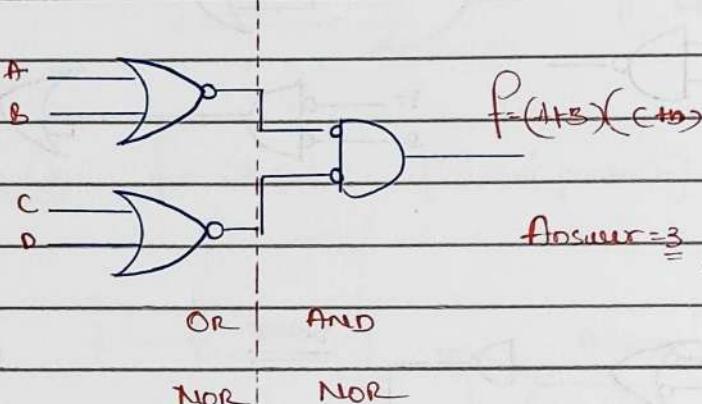


Case (4). Most Important.

Q1. $\overline{AB} + \overline{AB} \rightarrow \text{XOR}$.



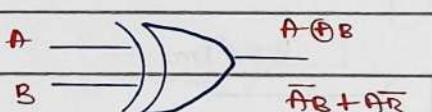
Case(5) $F = (A+B) \cdot (C+D)$ \rightarrow Product of Sum form.



$$\text{Answer} = 3$$

Note: Whenever minimum number of NOR Gate are asked, write the function in Pos form and implement by using -AOI which is AND-OR Implementation which is exactly equal to NOR NOR implementation.

XOR Gate.



A	B	Y	Truth Table
0	0	0	
0	1	1	
1	0	1	
1	1	0	

$$A \oplus A = 0$$

$$A \oplus 0 = A$$

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

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

eg::

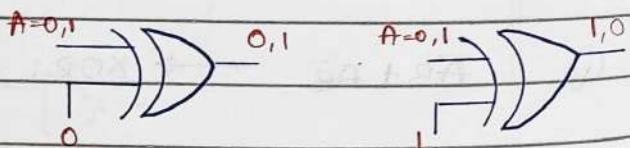
$$A \oplus B = C$$

$$A \oplus C = B$$

$$B \oplus C = A$$

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

Enable/Disable

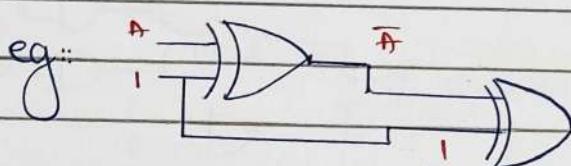


$$\rightarrow A \oplus A = 0 \quad (\text{for even})$$

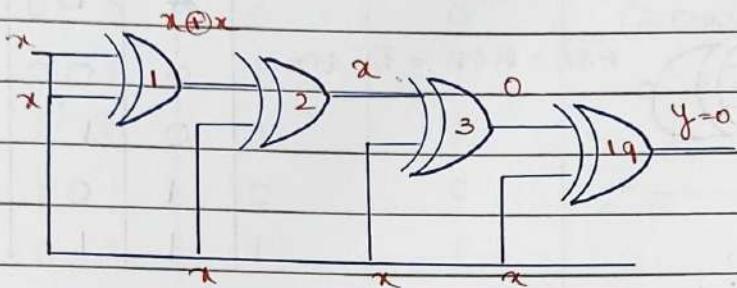
$$A \oplus A \oplus A = A \quad (\text{for odd})$$

"Buffer"

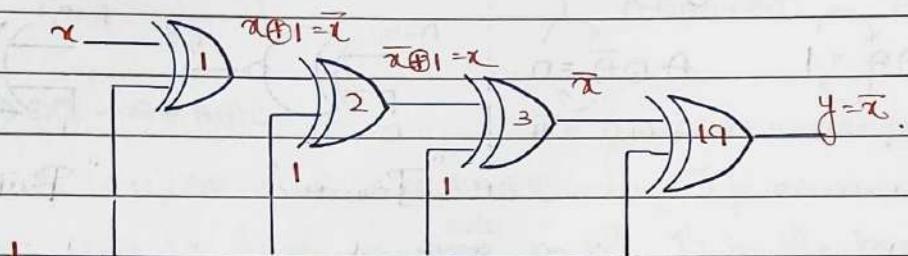
"Inverter".



Q1. Find the output y .



Q2. Find the output y .



→ XOR Gate follows Commutative as well as associative law

$$A \oplus B = B \oplus A \rightarrow \text{Commutative law}$$

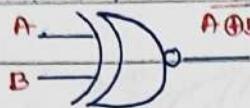
$$A \oplus B \oplus C = (A \oplus B) \oplus C \rightarrow \text{associative law}$$

→ X-OR Gate output will be high when odd number of 1's are connected in the inputs.

$A B C$	$A \oplus B \oplus C$	$(A \oplus B) \oplus C$
000	0	0
001	1	1
010	1	1
011	0	0
100	1	1
101	0	0
110	0	0
111	1	1

no. of 1's pairing

Odd number of 1's directed
 ↘ odd parity direction

X-NOR Gate:Symbol.

$$A \oplus B = A \text{OB} = \bar{A}\bar{B} + AB.$$

$$A = B \quad y = 1$$

$$A = \bar{B} \quad y = 0$$

* It is also called as "even parity detector / Equivalence logic / Equal detector / Coincidence logic".

$$\begin{aligned} A \odot A &= 1 \\ A \odot 1 &= A \end{aligned}$$

$$\begin{aligned} A \odot \bar{A} &= 0 \\ A \odot 0 &= \bar{A} \end{aligned}$$

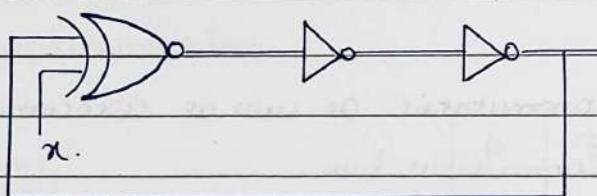
$$\begin{aligned} A = 0, 1 & \Rightarrow \text{Inverter} \\ 0 & \end{aligned}$$

$$\begin{aligned} A = 0, 1 & \Rightarrow \text{Buffer} \\ 1, 0 & \end{aligned}$$

"Inverter".

"Buffer".

eg::



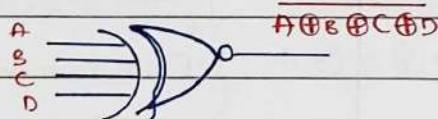
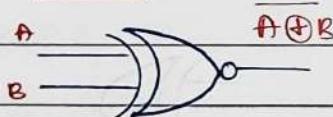
$x=0$ - Astable multivibrator

$x=1$ - Bistable multivibrator.

* X-NOR Gate follows commutative as well as associative law.

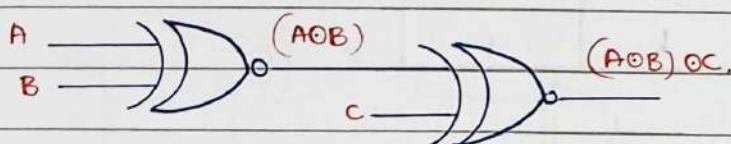
* For even number of variable and even number of inputs.

Case 1:



} Output will be high for even no of inputs are high.

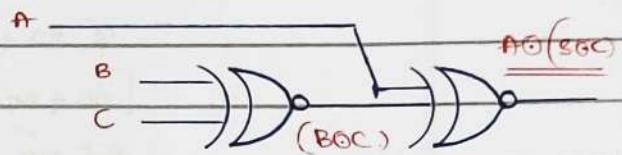
Case 2:



A	B	C	$(A \oplus B) \oplus C$	$A \oplus (B \oplus C)$
1.	0	0	0	0
2.	0	0	1	1
3.	0	1	1	1
4.	0	1	0	0
5.	1	0	1	1
6.	1	0	0	0
7.	1	1	0	0
8.	1	1	1	1

$$(A \oplus B) \oplus C = A \oplus (B \oplus C)$$

↪ Associative Law.



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

$$A \oplus B \oplus C \oplus D = \overline{A \oplus B \oplus C \oplus D}$$

{ for even!
variables }

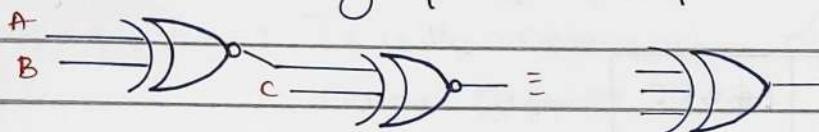
$$* (A \oplus B) \oplus C = A \oplus B \oplus C$$

{ for odd variables

$$((A \oplus B) \oplus C) \oplus D = A \oplus B \oplus C \oplus D \oplus E$$

{ for this combi. }

* Output will be high for odd no of 1's in the input

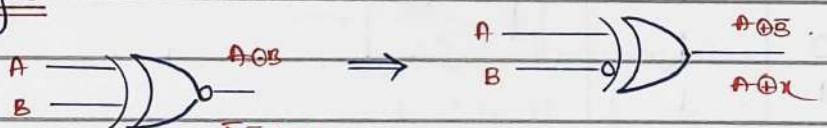


* Output will

be high for
even no. of
1's in input



Magic

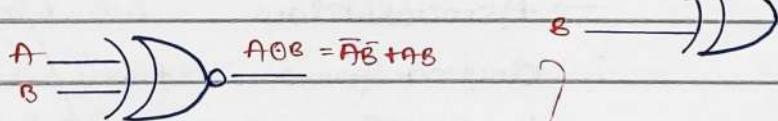


$$A \oplus A = \overline{A} \oplus A$$

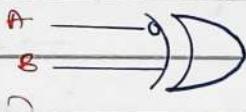
$$\overline{A} \oplus A = A \oplus \overline{A}$$

$$\overline{A} \oplus A = A \oplus \overline{A}$$

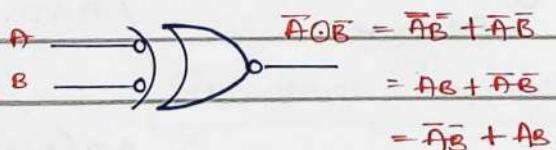
X-NOR



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



Same!



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

$$= A \oplus \overline{A}$$

$$= \overline{A} \oplus A$$

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

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

Q1 The boolean function given: $f(A, B) = A \oplus B \oplus AB$
 Which statement is/are correct?

$$f(A, B) = (A \oplus B) \oplus AB$$

$$= \bar{X} \oplus AB$$

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

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

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

$$= [\bar{A}B + AB] \cdot \bar{A}B + [\bar{A}B + A\bar{B}] \cdot [\bar{A} + \bar{B}]$$

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

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

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

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

$$= A \cdot 1 + \bar{A}B \Rightarrow A + \bar{A}B = (A + \bar{A})(A + B) = A + B \text{ (OR)}$$

Ans: → It is a OR Gate

→ It requires 3 NAND Gate to implement the function.

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

$$f = A + B$$

Q2 Minimized expression will be $y = A \oplus (A + B)$

$$A + B = X$$

→ Associative law

$$A \oplus (A + B)$$

$$A \oplus X$$

does not follow

$$(A \oplus A) + (A \oplus B)$$

$$\bar{A} + \bar{A}X$$

by X-OR Gate and

$$0 + A \oplus B$$

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

OR Gate.

$$A \oplus B$$

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

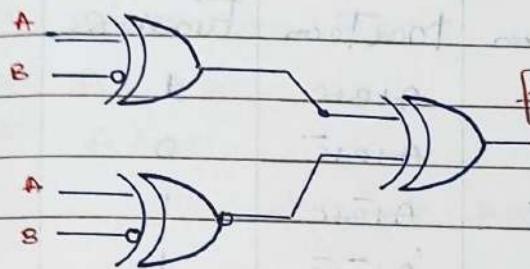
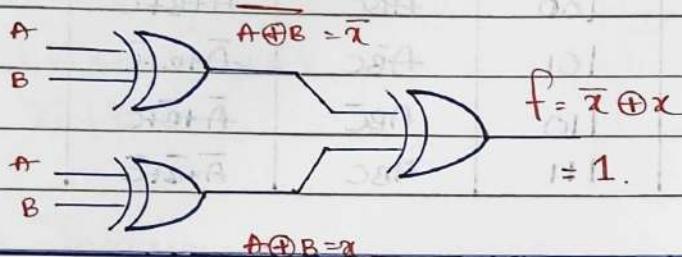
$$= \underline{\bar{A}B}$$

$$A \oplus (A + B)$$

$$A \oplus X$$

A	B	X	$(A + B)$	$A \oplus X$
0	0	0	0	0
0	1	1	1	1
1	0	1	1	0
1	1	1	1	0

eg:

 $f \text{ will be } 1.$ 

Law of Boolean Algebra

Boolean function: It is the combination of inputs on which Output depends.

$$\text{eg.: } f(A, B) = \underbrace{AB}_{\text{No. of variables = 2}} + \underbrace{\overline{A}\overline{B}}_{\text{No. of terms = 2, Literals = 4}}$$

Function.

SOP

POS

Sum of Product
(Minterm)

Product of Sum
(Maxterm)

Standard Canonical Form

Each term should contain all the variables

$$\text{eg.: } f(A, B) = \overline{A}B + A\overline{B}$$

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

Decimal	A.B.C.	Min Term	Max Term	Function
0	000	$\bar{A}\bar{B}\bar{C}$	$A+B+C$	1
1	001	$\bar{A}\bar{B}C$	$A+B+C$	0
2	010	$\bar{A}B\bar{C}$	$A+\bar{B}+C$	1
3	100	$\bar{A}BC$	$A+\bar{B}+\bar{C}$	1
4	100	$A\bar{B}\bar{C}$	$\bar{A}+B+C$	0
5	101	$A\bar{B}C$	$\bar{A}+\bar{B}+C$	0
6	110	$AB\bar{C}$	$\bar{A}+\bar{B}+C$	0
7	111	ABC	$\bar{A}+\bar{B}+\bar{C}$	1

Standard Canonical Sop form:

$$\begin{aligned}
 f(A, B, C) &= \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + \bar{A}B\bar{C} + ABC \\
 &= m_0 + m_2 + m_3 + m_7 \\
 &= \sum m(0, 2, 3, 7) \\
 &= \sum (0, 2, 3, 7)
 \end{aligned}$$

Standard Canonical Pos form:

$$\begin{aligned}
 F(A, B, C) &= (A+B+C)(\bar{A}+B+C)(\bar{A}+\bar{B}+\bar{C})(\bar{A}+\bar{B}+C) \\
 &= m_1 \cdot m_9 \cdot m_2 \cdot m_6 \\
 &= \prod M(1, 4, 5, 6) \\
 &= \prod (1, 4, 5, 6).
 \end{aligned}$$

Q1. No. of terms present in Standard Canonical form (SOP)?

Ans.

$$\begin{aligned}
 f(A, B, C) &= A+B+C \\
 &= A(\bar{B}+B)(\bar{C}+C) + (\bar{A}+A)BC \\
 &= A\bar{B}\bar{C} + A\bar{B}C + A\bar{B}\bar{C} + ABC + \bar{A}BC \\
 \therefore 5 \text{ terms are present.}
 \end{aligned}$$

1. Distribution Theorem.

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

$$\begin{aligned}
 &= A \cdot A + A \cdot C + AB + BC \\
 &= A + AC + AB + BC
 \end{aligned}$$

$$\begin{aligned}
 &= A[1+C+B] + BC \\
 &= A1 + BC \\
 &= A + BC
 \end{aligned}$$

eg: $A + \overline{AB}$
 $(A+\overline{A})(A+B)$
 $= 1(A+B)$
 $= A+B.$

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

Q2. $f(A,B) = \overline{A} + A\overline{B}$.

$$\begin{aligned}
 &= (\overline{A}+A)(\overline{A}+\overline{B}) \\
 &= \overline{A} + \overline{B}.
 \end{aligned}$$

2. Concise Theorem.

$$\begin{aligned}
 f(A,B,C) &= AB + \overline{A}C + BC \\
 &= AB + \overline{A}C + (\overline{A}\overline{B} + A)BC \\
 &= AB + \overline{A}C + \overline{A}BC + ABC \\
 &= AB[1+C] + \overline{A}C[1+B] \\
 &= AB + \overline{A}C
 \end{aligned}$$

$BC \rightarrow$ Redundant term.

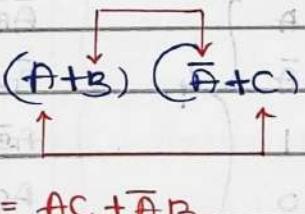
- * 3 variable function
- * each term consists of 2 variables.
- * each variable repeated twice except one
- * One variable repeated in form of complement.

eg: $AB + \overline{B}C + AC$

$$\Rightarrow AB + \overline{B}C \quad AC \rightarrow \text{redundant.}$$

3. Transpose Theorem.

$$\begin{aligned}
 &(A+B)(\overline{A}+C) \\
 &= A\cdot\overline{A} + A\cdot C + \overline{A}\cdot B + BC \\
 &= AC + \overline{A}B + BC \\
 &= AC + \overline{A}B
 \end{aligned}$$



4. DeMorgan's Law

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

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

Q1. Find min. no. of NAND Gate required to implement $f(A,B,C) = A + ABC + A\overline{B}\overline{C}$.

$$A[1+BC+\overline{B}\overline{C}] = A$$

$\therefore 0$ NAND Gate required.

Q2. Find the minimum no. of NAND gate required to implement the Boolean function

Ans

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

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

$$= A$$

∴ 0 NAND Gate required

Q3. Minimize the expression $f(A, B) = A + A\bar{B}$

Ans

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

$$= A$$

Q4. Minimize the expression $f(A, B) = \bar{A}\bar{B} + \bar{A}B + AB$

Ans

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

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

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

Q5. Minimize the expression $f(A, B) = \bar{A}\bar{B} + \bar{A}B + A\bar{B} + AB$

Ans

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

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

$n=1$

A } 4.
 \bar{A}
1
0

$n=2$

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

A 0
 $\bar{A} + 1$

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

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

16 different terms

term

"n" variables

2ⁿ

2 distinct expressions can be formed

2 minterms

2 minterms

Q6. Minimize the expression $f(A, B) = \bar{A}B + A\bar{B} = A \oplus B$.

Ans. Already minimized

Q7. Minimize the expression : $f(A, B) = AB + \bar{A}C + BC$

↳ function written in wrong format.

$$\text{eg.: } f(A, B, C) = \bar{A}\bar{B}C + \bar{A}BC + A\bar{B}C + ABC$$

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

$= A \oplus B \oplus C$ (Already Minimized)

$$\text{eg.: } f(B, A) = \begin{matrix} 0 \\ \uparrow \\ \bar{A}B \end{matrix} + \begin{matrix} 1 \\ \bar{A} \end{matrix} + \begin{matrix} 1 \\ AB \end{matrix} = \sum(1, 2, 3) \times$$

$$= B\bar{A} + \bar{B}A + BA = \sum(1, 2, 3) \checkmark$$

→ note the order, function should be written in same order.

Q8. Minimize the expression : $f(A, B, C) = \bar{A}\bar{B} + \bar{A}C + \bar{B}C$

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

Q9. Minimize the expression : $f(A, B, C) = (A+B)(A+C)(\bar{B}+C)$

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

Q10. Write the function for truth table and minimize it.

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

SOP form:

$$Y = \bar{A}\bar{B} + \bar{A}B0 + A\bar{B}.1 + AB.1$$

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

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

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

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

POS:

$$Y = (A+B+1).(\bar{A}+B+0).(\bar{A}+\bar{B}+1).$$

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

$$= 1(A+\bar{B}).1.1$$

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

Q11. Write the function for truth table and minimize it.

Ans.

A	B	y
0	0	C
0	1	\bar{C}
1	0	1
1	1	1

Sop:

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

Q12 Two way switch is an example of which logic?

Ans.

X-OR

Q13 If we have 4 variable, then total different expressions will be?

Ans.

$$2^{2^n} = 2^{2^4} = 2^{16} = 2^6 \times 2^{10} = \underline{\underline{64K}}$$

Q14 $A + BC + \bar{A}C$ is equal to :

$$\begin{aligned} & A + \overbrace{BC + \bar{A}C} \\ &= (A + \bar{A})(A + C) + BC \\ &= A + C + BC \\ &= A + C [1 + BC] \\ &= A + C \quad (\text{not in options!}) \end{aligned}$$

$$\text{eg: } f = AB + \bar{A}\bar{B} + ABC + \bar{A}\bar{B}\bar{C}$$

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

$$= AB + \bar{A}\bar{B} \quad (\text{Semi minimised}) \rightarrow AB + \bar{A}\bar{B} + \bar{A}BC$$

Modified Venned Diagram.

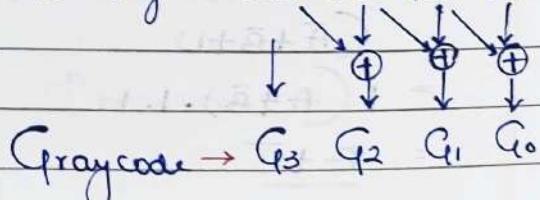
It is also known as K-map.

Minimization by K-map.

* Based on gray code

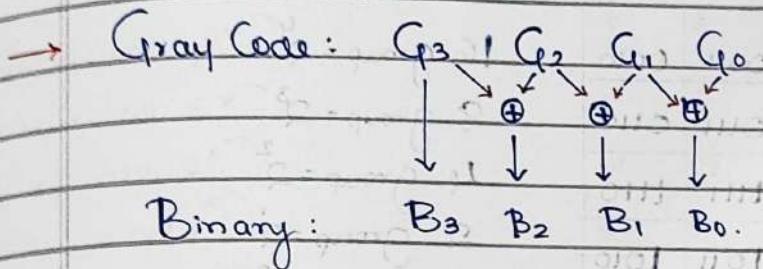
* Gray code:

→ Binary → $B_3 B_2 B_1 B_0$



eg: Binary: 0110101
Gray code: 0101111

eg: Binary: 111001011
Gray code: 100101110



eg: Gray code: 1101011
Binary: 1001101

Gray Code: Gray code is the code in which successive numbers differ by 1 bit.

Decimal	Binary	Gray Code
0	00	00
1	01	01
2	10	11
3	11	10

-
1. Unity Hamming distance code
 2. Cyclic code
 3. Reflected code.

Decimal	Binary	Gray Code
0	000	000
1	001	001
2	010	011
3	011	010
4	100	110
5	101	111
6	110	101
7	111	100

* eg.: $f(A, B)$

MSB (Most Significant bit)
LSB (Least Significant bit)

\bar{A}	\bar{B}	$\bar{A}\bar{B}$	$\bar{A}B$
0	0	00	01
1	0	01	11
0	1	10	10
1	1	11	01

→ $f(A, B, C)$

\bar{C}	$\bar{B}C$	$\bar{B}C$	BC	BC
X	00	01	11	10

\bar{A}	\bar{B}	\bar{C}	000	001	011	010
A	0	0	0	1	3	2
A	1	0	4	101	111	110

$$\rightarrow f(A, B, C, D)$$

AB	C	D	C	D
00	00	01	11	10
00	0000	0001	0011	0010
01	0100	0101	0111	0110
11	1100	1101	1111	1110
10	1000	1001	1011	1010

* Group can be formed in Order of 2^n .

16 group = $2^4 \rightarrow$ 4 variables reduce

8 group = $2^3 \rightarrow$ 3 variables reduce

4 group = $2^2 \rightarrow$ 2 variables reduce

2 group = $2^1 \rightarrow$ 1 variable reduce

1 group = $2^0 \rightarrow$ 0 variables reduce

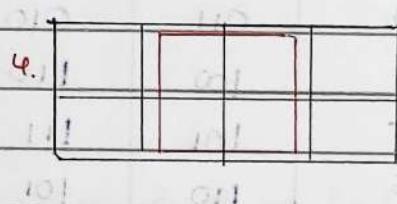
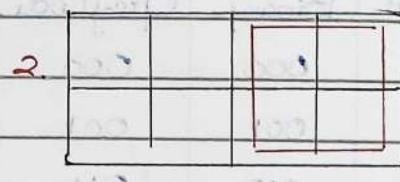
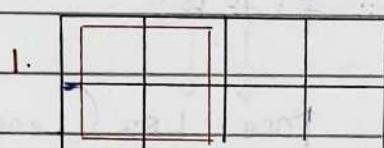
Rule of Minimization:

Term cuiu minimize

CJ Baba Rule: Kam se kam group banana hai and bade se bade group banana hai.

Variable cuiu minimize.

Quad (making group)



And many
Similar combi-
nations possibl

etc.

Q1. $f(A, B) = \bar{A}\bar{B} + \bar{A}B + A\bar{B} = \sum m(0, 1, 3)$

A	B	C	I
0	1	1	\bar{A}
1	0	1	B

$$f(A, B) = \bar{A}\bar{B} \cdot I + \bar{A}B \cdot I + A\bar{B} \cdot 0 + AB \cdot 1$$

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

Q2 $f(A, B) = \bar{A}\bar{B} + \bar{A}B + A\bar{B} + AB = \sum_m(0, 1, 2, 3)$

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

→ 1.

Q3. $f(A, B, C) = \sum_m(0, 1, 3, 6, 7)$

\bar{A}	\bar{B}	\bar{C}	$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$
0	0	0	00	01	11	10
1	1	1	4	5	6	7

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

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

$00 \quad 01 \quad 11 \quad 10$

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

$00 \quad 01 \quad 11 \quad 10$

\bar{A}	\bar{B}	\bar{C}	$\bar{B}\bar{C}$	$\bar{B}C$	BC
0	0	0	00	01	11
1	1	1	4	5	6

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

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

$00 \quad 01 \quad 11 \quad 10$

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

$00 \quad 01 \quad 11 \quad 10$

Semi minimized expression.

Question Practice Session

Q1. If $x \odot y = \bar{x} + y$ and $z = x \odot y$. Then $z \odot y$ will be:

$$\begin{aligned} z \odot y &= \bar{z} + y = \bar{x} \odot y + y \\ &= \bar{x} + y + y = \bar{x} + y \\ &= x + y \\ &= (x + y)(\bar{y} + y) \\ &= \underline{\underline{x + y}}. \end{aligned}$$

Q2. If $A^*B = AB + \bar{A}\bar{B}$ and $C = A^*B$. Then which one is correct?

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

$$C = A \odot B$$

$$(A \odot B) \odot (A \odot B)$$

$$B \odot C = A$$

$$x \odot x = 1$$

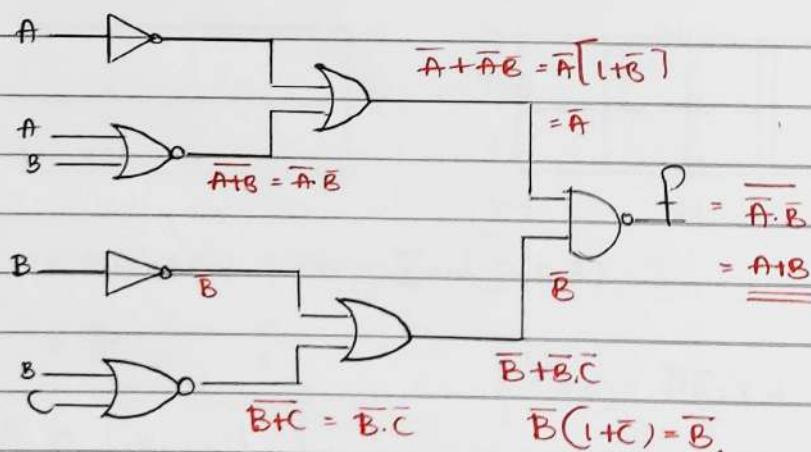
$$(i) A = B^*C - \checkmark$$

$$(ii) B = A^*C - \checkmark$$

$$(iii) A \odot B \odot C = 1 - \times$$

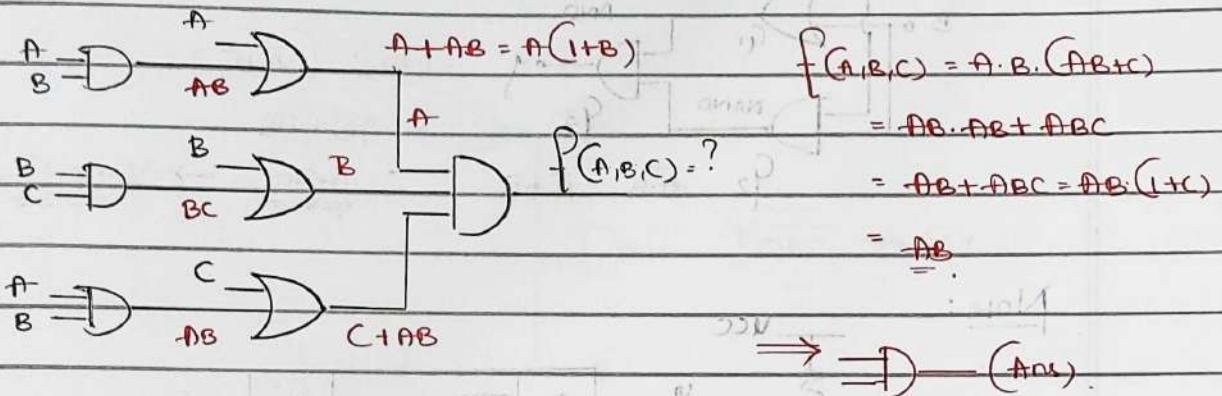
$$(iv) A = B - \times$$

Q3. The output for the given logic circuit will be:

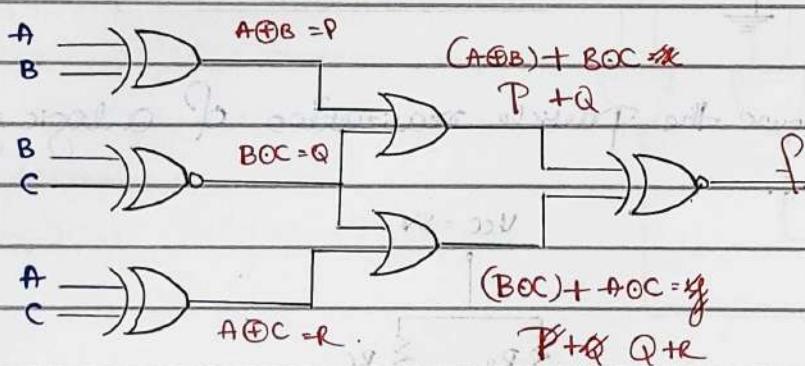


$$\rightarrow F = \underline{\underline{A + B}}$$

Q4. Consider the given logic circuit with the inputs A, B and C then $f(A, B, C)$ will be.



Q5. The output for the given logic circuit can be:



$$f = \bar{P}QR + PQ + PR + QR + QP$$

$$f = \bar{P}\bar{Q}\bar{R} + Q[P + 1 + R] + PR$$

$$f = \bar{P}\bar{Q}\bar{R} + Q + PR$$

$$= \bar{P}\bar{Q}\bar{R} + Q + PR$$

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

$$f = \underline{\bar{P}R + PR + Q} \Rightarrow (\bar{P}OR) + Q$$

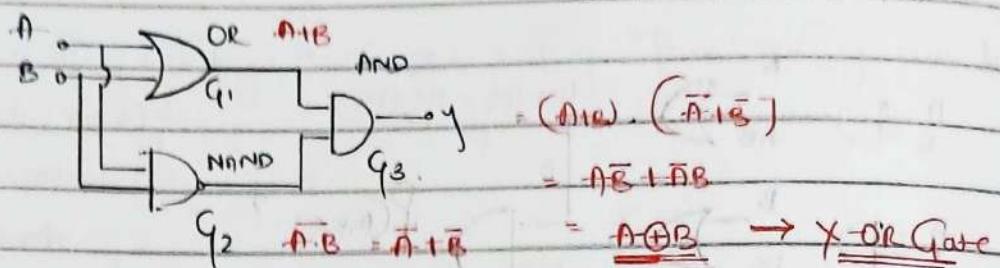
A	B	C	00	01	11	10
0	1		1			
1	1			1		

ABC	P $A \oplus B$	Q $B \oplus C$	R $A \oplus C$	x	y	z or y
000	0	1	0	1	1	1
001	0	0	1	0	1	0
010	1	0	0	1	0	0
011	1	1	1	1	1	1
100	1	1	1	1	1	1
101	1	0	0	1	0	0
110	0	0	1	0	1	0
111	0	1	0	1	1	1

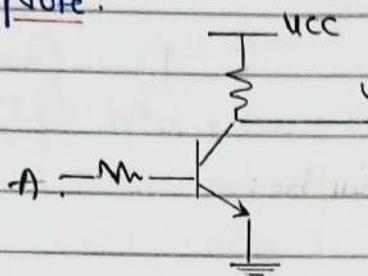
$$f = \bar{P}C + PC$$

$$= \underline{PC}$$

Q6. The following logic gate circuit equivalent to



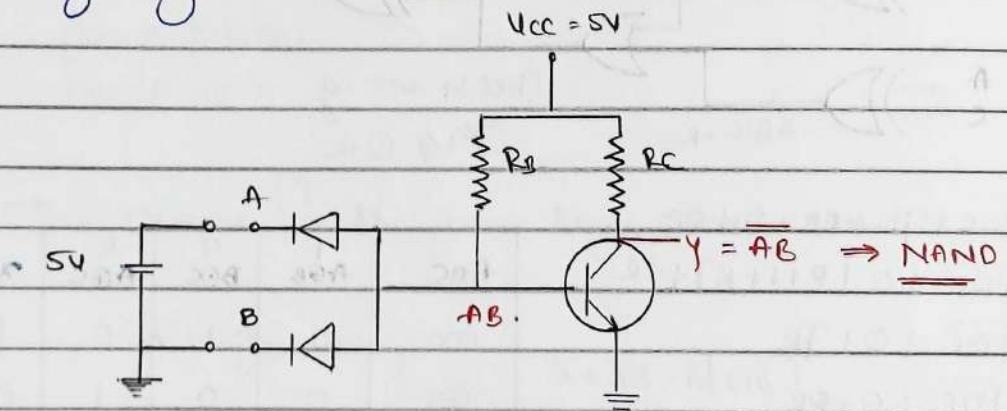
Note:



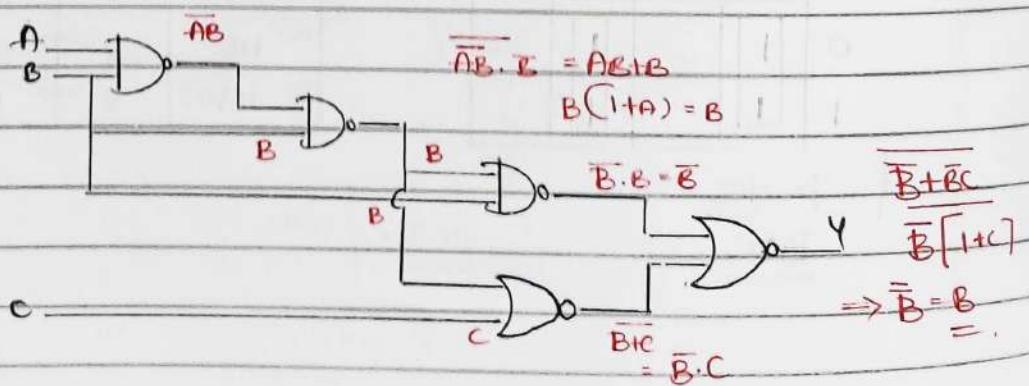
A	Transistor	Y
0	cutoff	1
1	Saturation	0

} NOT Gate!

Q7. Figure Shows the particle realizations of a logic gate. Identify the logic gate.

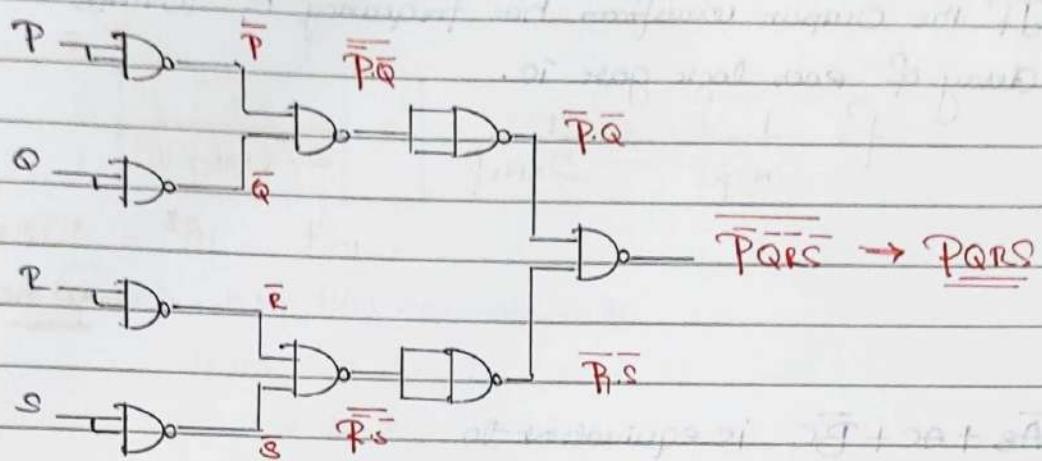


Q8. For the logic circuit shown, the Simplified boolean expression for the output y is.



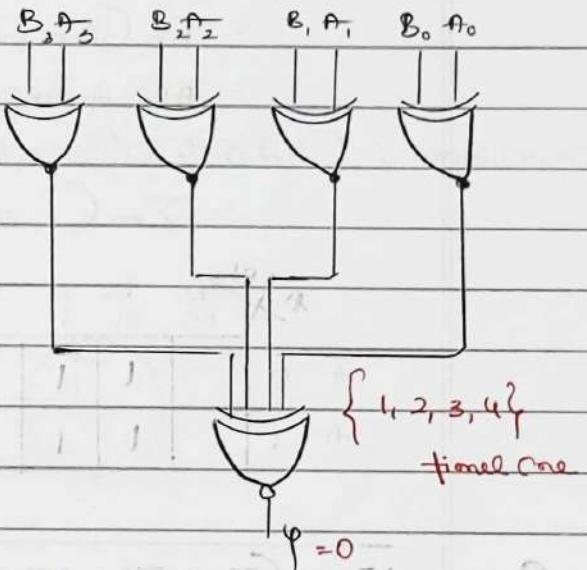
Q13

For the circuit shown in Fig. the boolean expression for the output y in terms of inputs P, Q, R and S is.

Q14. $A_3, A_2, A_1, A_0, B_3, B_2, B_1, B_0$ is shown

in figure. To get output $y=0$, choose one pair of correct input numbers.

$$\Rightarrow 0010, 1011$$

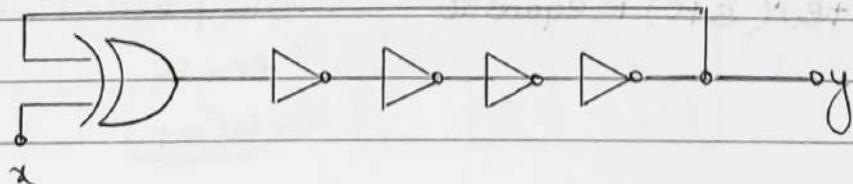


Q15. The minimum number of two input NAND gate required to implement $y = abcd$ is.

$$y = abcd$$

$$\text{NAND} = (2n-2) + c = (2+4-2) + 0 = 6.$$

Q16. All the logic gate in the circuit shown below, have equal finite propagation delay.



Q17. The Circuit can be used as a Clock generator, if $x=1$

Q18. If the Output waveform has frequency of 10 MHz the propagation delay of each logic gate is.

$$f = \frac{1}{2N\tau_{pd}} = \frac{1}{2 \times N \times \tau} = \frac{1}{2 \times 5 \times 10 \times 10^6}$$

$$= \frac{10^{-7}}{10} = 10^{-8} = 10 \times 10^{-9}$$

$$= \underline{\underline{10 \text{ ns}}}$$

Q19. $\bar{A}\bar{B} + A\bar{C} + \bar{B}\bar{C}$ is equivalent to

$$\begin{aligned} f &= \bar{A}\bar{B} + A\bar{C} + \bar{B}\bar{C} \\ &= \bar{A}\bar{B}(C+\bar{C}) + A(\bar{B}+B)C + (\bar{A}+\bar{A})BC \\ &= \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + A\bar{B}\bar{C} + A\bar{B}C + \bar{A}\bar{B}C + A\bar{B}C \\ &= \sum m(2, 3, 5, 7, 1) \\ &= \sum m(1, 2, 3, 5, 7). \end{aligned}$$

		BC	BC	BC	BC	
		00	01	11	10	
		A	0	1	1	1
\bar{A}	0		1	1	1	
A	1		1	1	1	

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

Q20. $\bar{A}\bar{B} + (\bar{A}\bar{B} + \bar{B}\bar{C} + A\bar{B}D + A\bar{B}\bar{D})$ is equal to

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

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

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

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

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

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

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

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

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

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

Q21. $(\bar{A} + \bar{B})(\bar{B} + C)$ is equal to :

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

$$\begin{aligned} &AB + BC \\ &\Rightarrow B(\underline{\underline{A + C}}) \end{aligned}$$

Q22. $\bar{A}\bar{B} + AC + \bar{B}C$ is equivalent to

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

$$\sum_m(0, 1, 5, 7)$$

\bar{A}	\bar{B}	C	00	01	11	10
0	1	1				
1		1	1			

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

Q23. $(A+B)(A+C)(A+\bar{C})$ is equivalent to

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

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

$$\Rightarrow \underline{\underline{A}}$$

HW.

Q24. A logical function is given as

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

Q25. $f(A, B, C) = \sum_m(0, 1, 3, 5, 6, 7)$

\bar{A}	\bar{B}	\bar{C}	\bar{D}	A	B	C	D	$\bar{A}\bar{B}$	\bar{C}	$\bar{A}\bar{B} + AB + C$
0	1	1	1	0	1	1	1	1	1	1
1	0	1	1	1	0	1	1	0	0	0

Q26. $f(A, B, C) = \sum_m(0, 2, 4, 6)$

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

Q27. $f(A, B, C) = \sum_m(0, 3, 5, 6)$

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

Q28. $f(A, B, C, D) = \sum m(0, 2, 4, 6, 10, 11, 13, 15)$

AB	CD	00	01	11	10	
$\bar{A}\bar{B}$	00	1	0	1	1	
$\bar{A}\bar{B}$	01	1	1	0	1	$\bar{A}D$
$A\bar{B}$	11	1	1	1	1	
$A\bar{B}$	10	1	1	1	1	
		1	1	1	1	
		1	1	1	1	

$\overbrace{\quad\quad\quad}^{ABD.} \quad \overbrace{\quad\quad\quad}^{ABC}$

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

Q29. $f(A, B, C, D) = \sum m(0, 1, 2, 4, 6, 9, 10, 11, 12, 13, 15)$

AB	CD	00	01	11	10	
$\bar{A}\bar{B}$	00	1	1		1	$\bar{A}\bar{D}$
$\bar{A}\bar{B}$	01	1			1	
$A\bar{B}$	11	1	1	1		
$A\bar{B}$	10	1	1	1	1	$\bar{A}C$
		1	1	1	1	
		1	1	1	1	

$\overbrace{\quad\quad\quad}^{ABC} \quad \overbrace{\quad\quad\quad}^{AB} \quad \overbrace{\quad\quad\quad}^{ABC}$

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

Q30. $f(A, B, C, D) = \sum m(1, 5, 6, 7, 11, 12, 13, 15)$

AB	CD	00	01	11	10	
$\bar{A}\bar{B}$	00	1				
$\bar{A}\bar{B}$	01	1	1	1		$\Rightarrow \bar{A}\bar{C} + \bar{A}\bar{C}D + A\bar{B} + A\bar{C}$
$A\bar{B}$	11	1	1	1		
$A\bar{B}$	10		1			

Q31. $f(A, B, C, D) = \sum m(1, 2, 5, 6, 8, 10, 12, 13, 14, 15)$

AB	CD	00	01	11	10	
$\bar{A}\bar{B}$	00	1		1		
$\bar{A}\bar{B}$	01	1		1		$\Rightarrow AB + \bar{C}\bar{D} + A\bar{D} + \bar{A}\bar{C}D$
$A\bar{B}$	11	1	1	1	1	
$A\bar{B}$	10	1			1	

$\overbrace{\quad\quad\quad}^{Quad} \quad \overbrace{\quad\quad\quad}^{AD}$

Q32. $f(A, B, C, D) = \sum m(1, 5, 6, 7, 11, 12, 13, 15)$

AB	00	01	11	10
00	1	1	1	1
01	1	1	1	1
10	1	1	1	1
11				
10	1		1	1

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

eg::

AB	00	01	11	10
00	1	1		1
01		1	1	
11	1	1		1
10	1	10		1

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

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

Don't Care Condition:

Combinations of inputs on which the output may or may not depends are called don't care conditions.

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

Q1. $f(A, B, C) = \sum m(0, 1, 6, 7) + \sum d(3)$

ABC	00	01	11	10
0	1	1	X	1
1			1	1

don't care.

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

$$f(A, B, C) = \sum m(0, 1, 2) + \sum d(3, 6)$$

ABC	00	01	11	10
0	1	1	X	1
1	.	.		X

eg::

ABC	00	01	11	10
0	1	X	X	X
1			X	X

$$\bar{A}$$

ABC	00	01	11	10
0	X	1	1	
1		X	1	X

$$\hookrightarrow C$$

Q2. $f(A, B, C, D) = \sum_m(0, 2, 4, 6, 7, 8, 10, 11, 12, 14, 15) + \sum_d(1, 3)$.

	AB	CD	00	01	11	10	C	D	00	01	11	10	00	01	11	10
	AB	CD	00	01	11	10			00	01	11	10	00	01	11	10
	X	X	1	X	X	1			1	1	1	1	1	1	1	1
	X	X	1	1	1	1			1	1	1	1	1	1	1	1
	X	X	1	1	1	1			1	1	1	1	1	1	1	1
	X	X	1	1	1	1			1	1	1	1	1	1	1	1

$$\Rightarrow C + \bar{D}$$

eg.:

	AB	CD	00	01	11	10
$\bar{A}\bar{B}$	00	X	1	1	X	1
$\bar{A}\bar{B}$	01			1	X	1
$A\bar{B}$	11		X	1		
$A\bar{B}$	10	X	X		1	

$$\Rightarrow \bar{A}\bar{B} + \bar{B}\bar{D} + BCD$$

	AB	CD	00	01	11	10
$A\bar{B}$	00	X	1	1		
$\bar{A}\bar{B}$	01		1	X		
$A\bar{B}$	11			X	X	1
$A\bar{B}$	10				1	

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

Q3. $F(A, B, C) = \prod_m(0, 1, 3, 6, 7) = \sum_m(2, 4, 5)$

	A	B	C	00	01	11	10
$A\bar{B}\bar{C}$	0	X	X	0	0	0	1
$A\bar{B}C$	1	X	X	1	1	0	0

	A	B	C	00	01	11	10
$\bar{A}\bar{B}\bar{C}$	0	X	X	0	0	0	1
$\bar{A}\bar{B}C$	1	X	X	1	1	0	0

M2 (Method 2)

P.S. $\left\{ \begin{array}{l} F = (A+B)(\bar{A}+\bar{B})(A+\bar{C}) \\ F = (A+B)(\bar{A}+\bar{B})(\bar{B}+\bar{C}) \end{array} \right.$

SOP $\left\{ \begin{array}{l} f = A\bar{B} + \bar{A}BC \\ \checkmark \end{array} \right.$

$$\begin{matrix} B+C \\ \bar{B}\bar{C} \\ 00 \end{matrix}$$

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

Q4. $f(A, B, C, D) = \sum_m(0, 1, 3, 6, 7, 10, 11)$ minimize it in Pos form
 $= \pi_m(2, 4, 5, 8, 9, 12, 13)$

AB	C+D	C+D	C+D	C+D
00	01	11	10	
A+B 00	1	1	1	0
A+B 01	0	0	1	1
A+B 11	0	0	0	0
A+B 10	0	0	1	1

AB	C+D	C+D	C+D	C+D
00	01	11	10	
A+B 00	0	0		
A+B 01	0	1	0	x
A+B 11	x	x		
A+B 10	0	0		x

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

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

Variable inside the K-map.

eg:

AB	B	B
0	0	1
A 0	C	C
A 1		

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

eg:

AB	B	B
0	0	1
A 0	1	C
A 1		

$$f(A, B, C) = \bar{A}\bar{B}.1 + \bar{A}BC \\ = \bar{A}[\bar{B}+BC] \\ = \bar{A}[(\bar{B}+B) \cdot (\bar{B}+C)] \\ = \bar{A}(\bar{B}+C)$$

Method 4

Case 1: $C=0$

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

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

Case 2: $C=1$

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

$$\rightarrow \bar{A}C$$

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

Method 2:

$$\begin{aligned}
 f(A, B, C) &= \bar{A}\bar{B}(\bar{C}+C) + \bar{A}BC \\
 &= \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + \bar{A}BC \\
 &= \sum_m (0, 1, 3) = \prod_m (2, 4, 5, 6, 7)
 \end{aligned}$$

		$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$
		00	01	11	10
\bar{A}	0	1	1	1	0
A	1	0	0	0	0

		$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$
		00	01	11	10
\bar{A}	0	1	1	1	0
A	1	0	0	0	0

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

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

Method 3:

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

Q1.

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

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

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

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

$\bar{ABC} + \bar{AC} + BC$

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

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

Q2.

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

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

$\Rightarrow \underline{AC + \bar{AB} + BC}$

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

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

$\bar{ABC} + AC + BC$

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

Implicants and Prime Implicants.

1. **Implicants:** The total number of minterms in the boolean expression are called implicants. Or in kmap the total no. of 1's is called implicants.
2. **Prime Implicants:** Total number of possibilities of formation of group are called prime implicants.
3. **Essential Prime Implicants / Subtractive PI:** If there are two answers for a boolean expression then the terms common in both the answers are called essential prime implicants.
4. **Reduced Prime Implicants:** The terms that are not essential prime implicants are called reduced prime implicants.

Q1. $f(A, B, C) = \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} + ABC$
 $= \sum_m(0, 1, 3, 6, 7)$

		$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$
		00	01	11	10
\bar{A}	0	1	1	1	
	1			1	1

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

Implicants = 5

Prime Implicants = 4

Essential prime implicants =

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

SPI = 1

RPI = 1

		$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$
		00	01	11	10
\bar{A}	0	1	1	1	
	1			1	1

EPI.

$P_I \{ \bar{A}\bar{B}, AB, \bar{A}C, BC \}$

Q2. $f(A, B, C) = \sum_m(2, 3, 4, 5, 7)$

$P_I \{ \bar{A}\bar{B}, \bar{A}B, BC, AC \}$

Q5. $f(A, B, C, D) = \sum_m(1, 5, 6, 7, 11, 12, 13, 15)$

$$P_I \{ \bar{A}CD, ABC, ACD, \bar{ABC}, BD \}$$

$\bar{AB} \times \bar{C}D$	00	01	11	10								
00		1										
01		1	1	1								
11	1	1	1									
10				1								

Implicants = 8

 $P_I = 5$ $EPI = 4$ $SpI = 0$ $RPI = 1$

$$EPI \{ \bar{A}CD + \bar{ABC} + ACD + ABC \}$$

$$+ ABC \}$$

Q6.

$\bar{A} \times BC$	00	01	11	10							
0	1	1	1								
1		1	1	1							

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

Implicants = 6

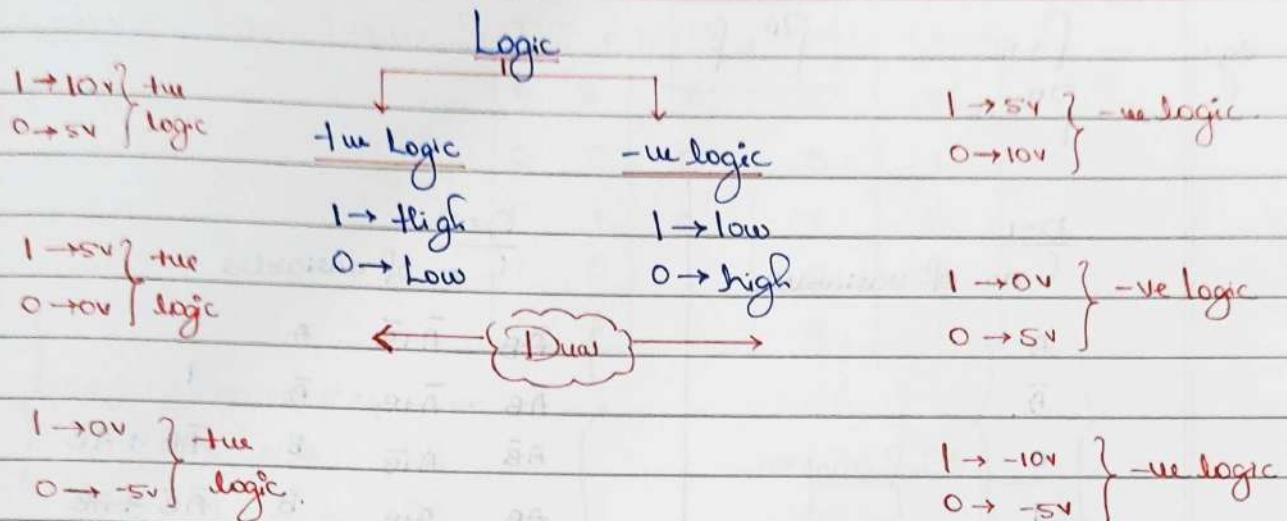
 $P_I = 3$ $EPI = 3$ $SpI = 0 \quad RPI = 0$ HW

Q7.

$\bar{AB} \times \bar{C}D \quad \bar{CD} \quad \bar{CD} \quad CD \quad CD$	00	01	11	10							
\bar{AB}	00	1	1								
\bar{AB}	01		1	1							
\bar{AB}	11		1	1	1						
\bar{AB}	10	1	1		1						

Q8.

$\bar{AB} \times \bar{C}D \quad \bar{CD} \quad \bar{CD} \quad CD \quad CD$	00	01	11	10							
\bar{AB}	00	1	1								
\bar{AB}	01	1		1							
\bar{AB}	11		1	1	1						
\bar{AB}	10		1	1	1						

And Gate (true)

A	B	Y		A	B	Y
0	0	0	Logic	1	1	1
0	1	0	Complement	1	0	1
1	0	0	Dual	0	1	1
1	1	1		0	0	0

1 → -10V } -ve logic
0 → -5V }

Dual → true logic ← → -ve logic

1. $I \longleftrightarrow 0$
2. AND \longleftrightarrow OR
3. $\cdot \longleftrightarrow +$
4. NAND \longleftrightarrow NOR
5. X-OR \longleftrightarrow X-NOR
6. Buffer \longleftrightarrow Buffer
7. Inverter \longleftrightarrow Prowter.

$$f = A \cdot 1$$

$$f^o = A + 0 = A$$

eg: $f = AB + CD + \bar{E}F$
 $f^o = (A+B) \cdot (C+D) \cdot (\bar{E}+F)$

$$f = \overline{ABC + DEF}$$

$$f^o = \overline{(A+B+C) \cdot (D+E+F)}$$

eg: $f = AB + CD$
 $f^o = (AB + B) \cdot (C + D)$
 $f^o = AB + CD$

$$f^o = f$$

Q9.

	$\bar{A}B$	$C\bar{D}$	$\bar{C}D$	CD	$\bar{C}\bar{D}$
$\bar{A}B$	00	1	1	1	1
$\bar{A}B$	01		1	1	
$\bar{A}B$	11	1		1	1
$\bar{A}B$	10	1		1	

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

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

Implicants = 10

 $P_I = 4$ $EPI = 2$ $SPI = 1$ $R_{PD} = 1$

AD-

Ans 1

AD-

Q10. Let a function F which has 3 input variables (x, y, z). The function F will be high only when atleast two of the input variables are set to high. Draw the K-map for the given functions. Let the number of PR in K-map = 'a' and the no. of EPP in K-map = 'b'. Find the quadratic mean of 'a' and 'b'.

Ans:

	x	y	z	f
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

$$f(x, y, z) = \sum m(3, 5, 6, 7)$$

\bar{x}	\bar{y}	\bar{z}	00	01	11	10
\bar{x}	\bar{y}	\bar{z}	1			
\bar{x}	y	\bar{z}		1		
x	\bar{y}	\bar{z}			1	
x	y	\bar{z}				1
x	\bar{y}	z				1

 $a = P_I = 3$ $b = EPI = 3$

$$f = xy + yz + zx \quad \text{Quadratic mean:}$$

$$= \sqrt{\frac{a^2 + b^2}{2}} = \sqrt{\frac{3^2 + 3^2}{2}} = \sqrt{\frac{18}{2}} = \sqrt{9} = 3$$

Q11. Find the number of prime implicants & essential prime implicants in the given K.

\bar{x}	00	01	11	10
00	1			
01		1		1
11	1			
10		1		1

 $P_I = 6$ $EPI = 6$

Q3.

$A \times BC$

	00	01	11	10
0	1	1	1	
1	1		1	1

$$f(A, B, C) = \sum m(0, 1, 3, 4, 6, 7)$$

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

Pimplicants = 6

PI = 6

EPI = 0 (Term common to all)

SPE = 3

RPI = 3.

$A \times BC$

	00	01	11	10
0	1	1	1	
1	1		1	1

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

$$PI \{ \bar{A}\bar{B} + BC + AC + \bar{B}\bar{C} + \bar{A}\bar{C} + AB \}$$

Q2.

(continued)

$A \times BC$

	00	01	11	10
0			1	1
1	1	1	1	

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

Pimplicants = 5

PI = 4

EPI = 2

SPE = 1

RPI = 1.

$A \times BC$

	00	01	11	10
0			1	1
1	1	1	1	

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

$$PI \{ A\bar{B} + \bar{A}B + AC + BC \}$$

Q4.

$$f(A, B, C) = \sum m(0, 2, 3, 4, 5, 7)$$

$A \times BC$

	00	01	10	11
0	1		1	1
1	1	1	1	

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

Pimplicants = 6

PI = 6

EPI = 0

SPE = 3

RPI = 3

$A \times BC$

	00	01	11	10
0	1		1	1
1	1	1	1	

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

$$PI \{ \bar{B}\bar{C} + AC + \bar{A}B + \bar{A}\bar{C} + BC + A\bar{B} \}$$

eg. $f = A$ $f^D = f$ \rightarrow Self Dual.
 $f^D = A$

$n=1$
(no. of variables.)

A	
\bar{A}	
0	
1	Self dual = 2

$n=2$
(no. of variables)

$\bar{A}\bar{B}$	$\bar{A}+\bar{B}$	A	0	Self dual = 4
$\bar{A}B$	$\bar{A}+B$	\bar{A}	1	
$A\bar{B}$	$A+\bar{B}$	B	$\bar{A}B + A\bar{B}$	
AB	$A+B$	\bar{B}	$\bar{A}\bar{B} + A\bar{B}$	

* n variables $\longrightarrow 2^{2^n}$ $\longrightarrow 2^{2^{n-1}}$ Self dual expression.
Expressions.

Combinational Circuit.

- * A circuit without feedback or memory are called Combinational circuit.
- * Static circuit.

Designing of Combinational Circuit:

Step 1: Find the number of inputs and outputs.

Step 2: Write the truth table

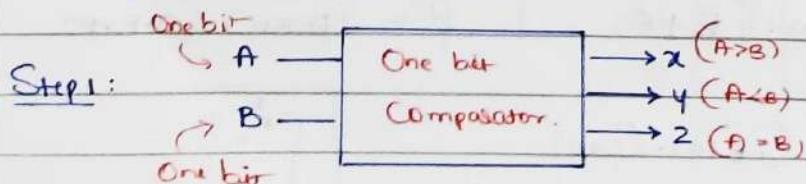
Step 3: Write the logical expression

Step 4: Minimize the logical expression.

Step 5: Hardware implementation.

Comparator

1. Design a one bit Comparator



Step 2: Truth Table

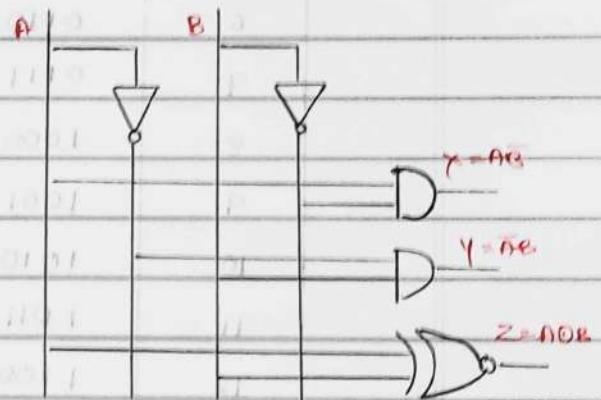
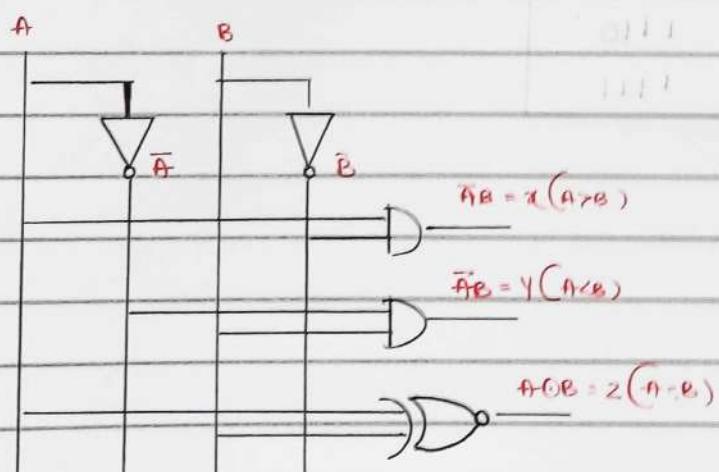
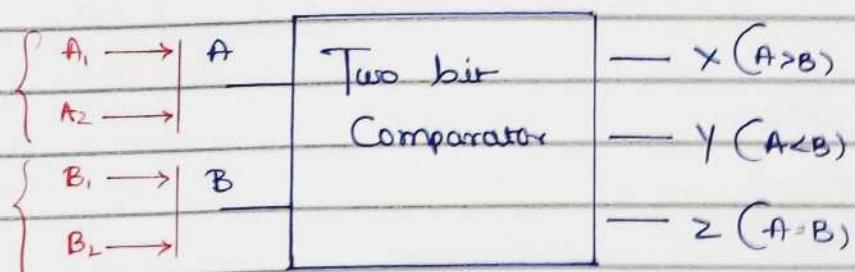
		$(A > B)$	$(A < B)$	$(A = B)$	
A	B	x	y	z	
0	0	0	0	1	
0	1	0	1	0	
1	0	1	0	0	
1	1	0	0	1	

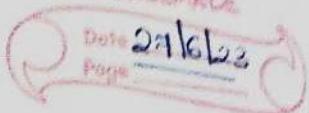
Step 3: Logical expression:

$$x(A > B) = \bar{A}B$$

$$y(A < B) = \bar{A}B$$

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

Step 4: Minimization (Already minimized)Step 5: Hardware implementation.Two Bit Comparator:



Decimals	Binary	<u>Step 2.</u>	A ₁	A ₀	B ₁	B ₀	X	Y	Z
0	0000	- Then	0	00	0	0	0	0	1
1	0001	- Table	0	00	1	0	1	0	0
2	0010		0	01	0	0	1	0	0
3	0011		0	01	1	0	1	0	0
4	0100		0	10	0	1	0	0	0
5	0101		0	10	1	0	0	1	0
6	0110		0	11	0	0	1	0	0
7	0111		0	11	1	0	1	0	0
8	1000		1	00	0	1	0	0	0
9	1001		1	00	1	1	0	0	0
10	1010		1	01	0	0	0	1	0
11	1011		1	01	1	0	1	0	0
12	1100		1	10	0	1	0	0	0
13	1101		1	10	1	1	0	0	0
14	1110		1	11	0	1	0	0	0
15	1111		1	11	1	0	0	1	0

Step 3 Logical expression:

$$x(A \geq B) = \sum m(4, 8, 9, 12, 13, 14)$$

$$y(A < B) = \sum m(1, 2, 3, 6, 7, 11)$$

$$z(A = B) = \sum m(0, 5, 10, 15)$$

Step 4. Minimization

f_{for x}

		$\bar{B}B_0$	$\bar{B}B_1$	B_0B_1	$B_0\bar{B}_1$
		00	01	11	10
$\bar{A}A_0$	00				
$\bar{A}A_0$	01	1			
$A\bar{A}_0$	11	1	1		1
$A\bar{A}_0$	10	1	1		

		$\bar{B}B_0$	$\bar{B}B_1$	B_0B_1	$B_0\bar{B}_1$
		00	01	11	10
$\bar{A}A_0$	00	1		1	1
$\bar{A}A_0$	01				
$A\bar{A}_0$	11				
$A\bar{A}_0$	10				1

f_{for y}

Semi minimised

$$\begin{aligned}
 X(A \geq B) &= A_1 \bar{B}_1 + \bar{A}_1 A_0 \bar{B}_1 \bar{B}_0 + A_1 A_0 B_1 \bar{B}_0 \\
 &= A_1 \bar{B}_1 + (\bar{A}_1 \bar{B}_1 + A_1 B_1) A_0 \bar{B}_0 \\
 &= A_1 \bar{B}_1 + (A_1 \odot B_1) A_0 \bar{B}_0.
 \end{aligned}$$

for '2'

~~A₁ ⊕ B₁~~

	00	01	11	10
00	1			
01		1		
11			1	
10				1

Minimised

$$\begin{aligned}
 X(A \geq B) &= A_1 \bar{B}_1 + A_0 \bar{B}_1 \bar{B}_0 + A_1 A_0 \bar{B}_0 \\
 &= A_1 \bar{B}_1 + (A_1 + B_1) A_0 \bar{B}_0.
 \end{aligned}$$

$$Z(A=B) = (A_1 \odot B_1) \cdot (A_0 \odot B_0)$$

Semi minimised

$$\begin{aligned}
 Y(A \leq B) &= \bar{A}_1 B_1 + \bar{A}_1 \bar{A}_0 \bar{B}_1 \bar{B}_0 + \bar{A}_1 \bar{A}_0 B_1 B_0 \\
 &= \bar{A}_1 B_1 + (\bar{A}_1 \bar{B}_1 + A_1 B_1) \bar{A}_0 \bar{B}_0 \\
 &= \bar{A}_1 B_1 + (A_1 \odot B_1) \bar{A}_0 \bar{B}_0.
 \end{aligned}$$

Minimised

$$Y(A \leq B) = \bar{A}_1 B_1 + (\bar{A}_1 + B_1) \bar{A}_0 \bar{B}_0$$

H.W.Step 5

Hardware Implementations

Comparators

One bit Comparator

Total Conditions = 4

Equal Condition = 2

(Unequal Condition = 2)

Greater = Less Cond = 1

Two bit Comparators

Total conditions = 16

Equal Condition = 4

(Unequal " = 12)

Greater = Less " = 6.

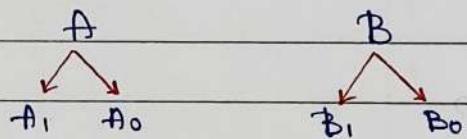
Three bit comparators

Total combinations = 64

Equal Condition = 8

(Unequal " = 56)

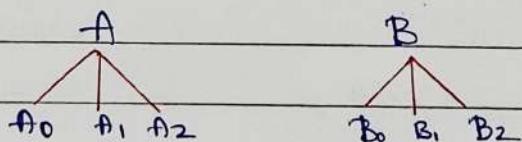
Greater = Less " = 28.

'N' bit ComparatorsTotal condition = 2^{2n} Equal Condition = 2^n Unequal Condition = $2^{2n} - 2^n$.Greater = Less = $2^{2n} - 2^n / 2$.

$$X(A \geq B) = A_1 \bar{B}_1 + (A_1 \oplus B_1) A_0 \bar{B}_0$$

$$Y(A < B) = \bar{A}_1 B_1 + (A_1 \oplus B_1) \bar{A}_0 B_0$$

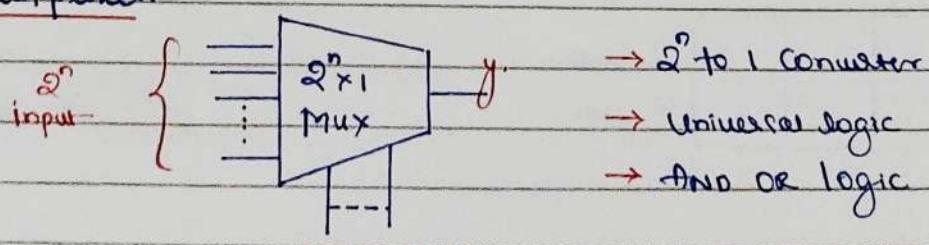
$$Z(A=B) = (A_1 \oplus B_1) \cdot (A_0 \oplus B_0)$$



$$X(A \geq B) = A_2 \bar{B}_2 + (A_2 \oplus B_2) \bar{A}_1 \bar{B}_1 + (A_2 \oplus B_2) (A_1 \oplus B_1) A_0 \bar{B}_0$$

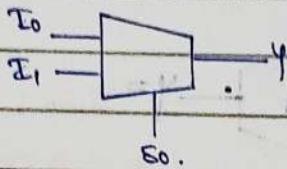
$$Y(A < B) = \bar{A}_2 B_2 + (A_2 \oplus B_2) \bar{A}_1 B_1 + (A_2 \oplus B_2) (A_1 \oplus B_1) \bar{A}_0 B_0$$

$$Z(A=B) = (A_2 \oplus B_2) (A_1 \oplus B_1) (A_0 \oplus B_0)$$

Multiplexer

Q1. Design a 2×1 Mux.

Step 1:



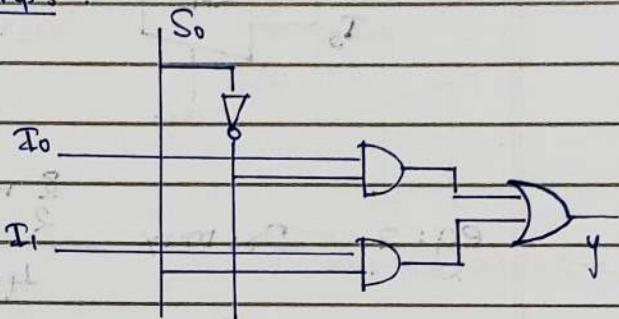
$$\text{Step 3: } Y = \bar{S}_0 I_0 + S_1 I_1$$

Step 4: Already Minimized

Step 2:

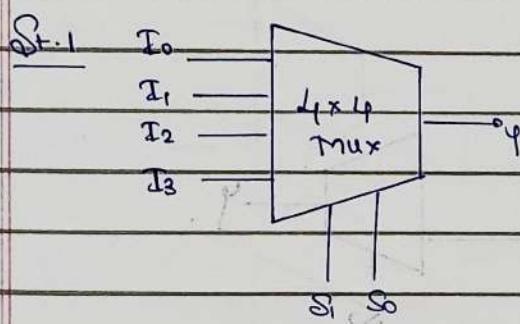
S_0	Y
0	I_0
1	I_1

Step 5:



Q2. Design a 4×1 mux.

Step 1:



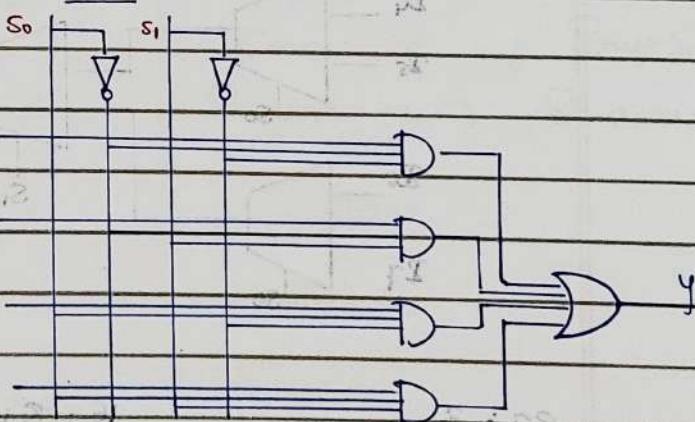
$$\text{Step 3: } Y = \bar{S}_1 \bar{S}_0 I_0 + \bar{S}_1 S_0 I_1 + S_1 \bar{S}_0 I_2 + S_1 S_0 I_3$$

Step 4: No minimization

Step 2:

S_1	S_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

Step 5:



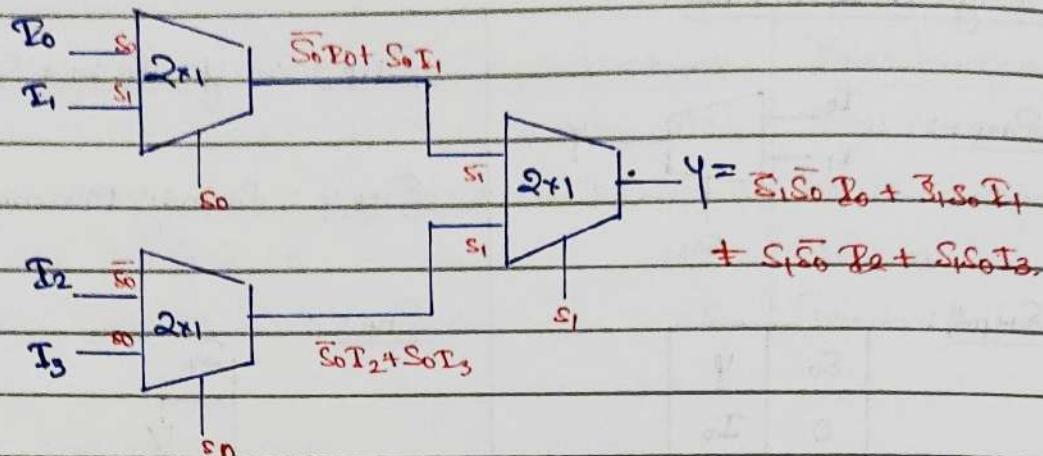
Type: 1 Designing of higher order mux by using lower order mux.

eg. 2×1 mux

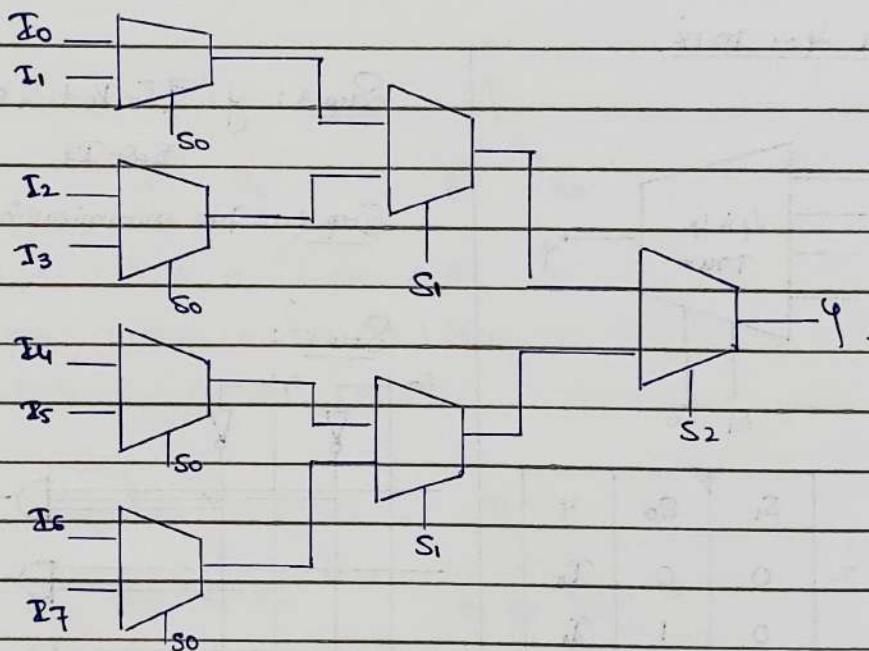
$$\frac{1}{2}^2 + \frac{1}{2}$$

$$2 + 1 = 3$$

4×1 mux



eg: 2 2x1 mux $\frac{8}{2} + \frac{4}{2} + \frac{2}{2}$ 8x1 mux
 $4+2+1 = 7.$

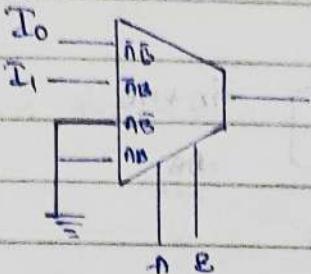


eg: 3 2x1 mux $\frac{16}{2} + \frac{8}{2} + \frac{4}{2} + \frac{2}{2}$ 16x1 mux
 $8+4+2+1 = \underline{\underline{15}}$

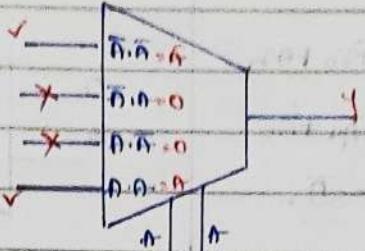
2x1 mux 63 \rightarrow 64x1 mux

2x1 mux 2⁷-1 \rightarrow 2⁷x1 mux

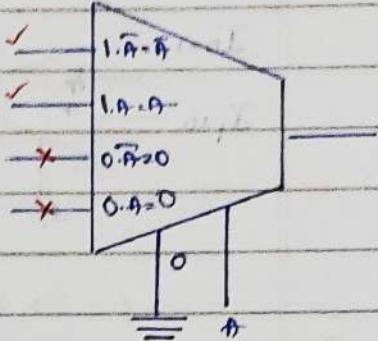
Q1. 4×1 mux $\frac{S}{4} + \frac{R}{4}$ $\rightarrow 8 \times 1$ mux
 $\underbrace{\text{2 select line}}_{\text{2 select line}}$ $S+1 = 3$ $\underbrace{\text{3 select line}}$

Method 1

Select line = 2

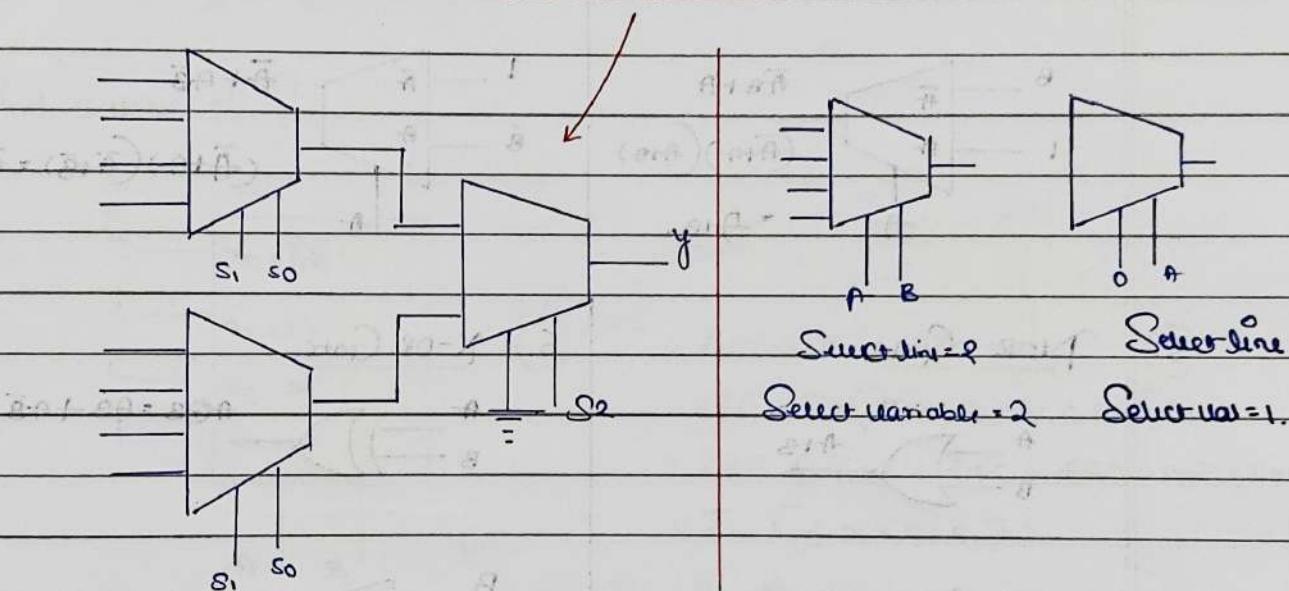
Method 2

Select line = 1

Method 3

Select variable = One

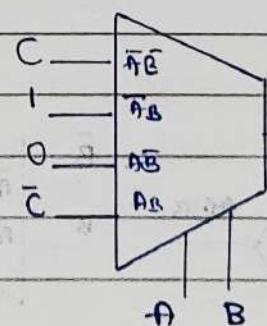
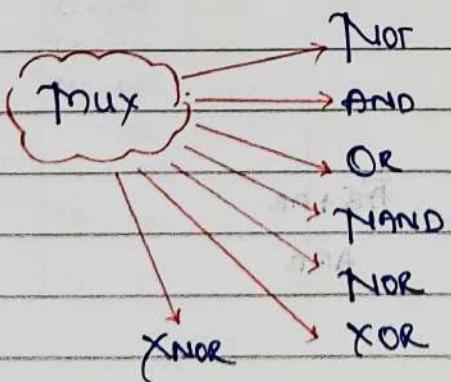
4×1 mux $\rightarrow 8 \times 1$ mux
 $\underbrace{\text{3 Select variables}}$



Select variable = 2

Select variable = 2

Select variable = 1

Type 2: Mux as a universal logic.

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

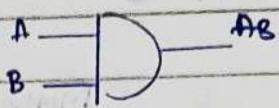
1.

NOR Gate

$I_0 = 1$

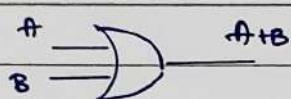
$$\begin{aligned} I_1 &= 0 \\ \text{F-bar} &= \overline{A \oplus B} = \overline{A} \cdot \overline{B} = \overline{A} + \overline{B} \end{aligned}$$

2.

ANo Gate

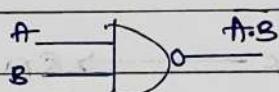
$$\begin{aligned} O &= \overline{A} \cdot \overline{B} \\ &= \overline{A} + \overline{B} \end{aligned}$$

3.

OR Gate

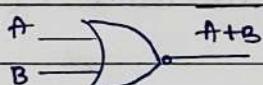
$$\begin{aligned} B &= \overline{A} \cdot \overline{B} = \overline{A} + \overline{B} \\ I &= \overline{A} \cdot \overline{B} = (\overline{A} + \overline{B})(\overline{A} + \overline{B}) = \overline{A} + \overline{B} \end{aligned}$$

4.

NAND Gate

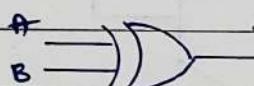
$$\begin{aligned} I &= \overline{A} \cdot \overline{B} = \overline{A} + \overline{B} \\ \bar{B} &= \overline{A} \cdot \overline{B} = (\overline{A} + \overline{B})(\overline{A} + \overline{B}) = \overline{A} + \overline{B} = \overline{A} \cdot \overline{B} \end{aligned}$$

5.

NOR Gate

$$\begin{aligned} \bar{B} &= \overline{A} \cdot \overline{B} = \overline{A} + \overline{B} \\ O &= \overline{A} \cdot \overline{B} = \overline{A} + \overline{B} \end{aligned}$$

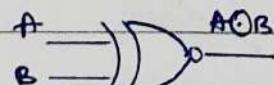
6.

X-OR Gate

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

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

7.

X-NOR Gate

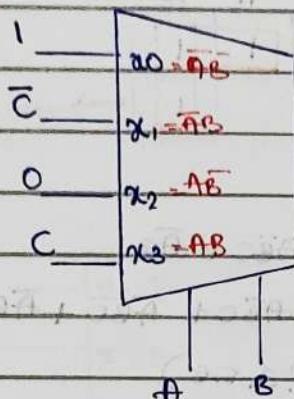
$$\begin{aligned} B &= \overline{A} \cdot \overline{B} = \overline{A} + \overline{B} \\ O &= \overline{A} \cdot \overline{B} = \overline{A} + \overline{B} \end{aligned}$$

Note: To design NOT, AND, OR \rightarrow One 2×1 mux required

To design NAND, NOR, XOR-XNOR, \rightarrow Two 2×1 mux required.

Type 3 Minimization.

Q. Find the output f :

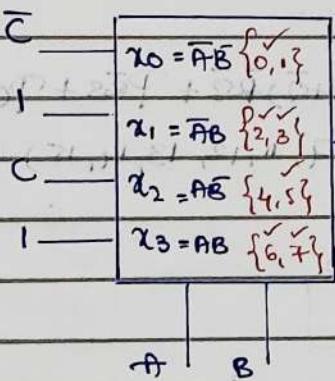


X	00	01	11	10
0	1	1		1
1			1	

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

$$\begin{aligned}
 f(A, B, C) &= \bar{A}\bar{B}, 1 + \bar{A}B\bar{C} + A\bar{B}, 0 + A\bar{B}C \\
 &= \bar{A}\bar{B} + \bar{A}B\bar{C} + A\bar{B}C \\
 &= \bar{A}\bar{B}(\bar{C} + C) + \bar{A}B\bar{C} + A\bar{B}C \\
 &= \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + A\bar{B}\bar{C} + A\bar{B}C \\
 &= \sum_m(0, 1, 2, 7)
 \end{aligned}$$

Q2. Find output of "f".

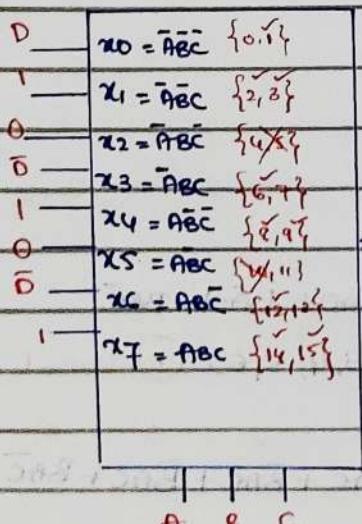


$$\begin{aligned}
 f(A, B, C) &= \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + A\bar{B}\bar{C} + AB \\
 &= \bar{A}\bar{B}\bar{C} + \bar{A}B(\bar{C} + C) + A\bar{B}\bar{C} + AB(C + \bar{C}) \\
 &= \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + A\bar{B}\bar{C} + A\bar{B}C + ABC \\
 &= \sum_m(0, 2, 3, 5, 6, 7)
 \end{aligned}$$

X	00	01	11	10
0	1	1		1
1			1	1

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

Q3.

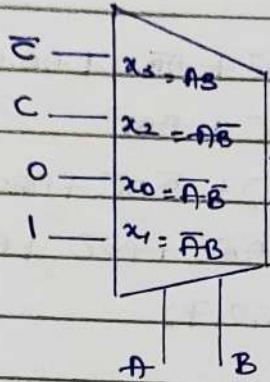


$$\begin{aligned}
 f(A, B, C, D) &= \bar{A}\bar{B}\bar{C}D + \bar{A}\bar{B}C\bar{D} + \bar{A}B\bar{C}\bar{D} + A\bar{B}\bar{C}D + A\bar{B}C\bar{D} \\
 &\quad + ABC\bar{D} + A\bar{B}CD + ABCD \\
 &= \sum_m(1, 2, 3, 6, 8, 9, 12, 14, 15)
 \end{aligned}$$

~~AB~~

00	00	01	11	10
00		1	1	1
01				1
11	1		1	1
10	1	1		

(Spain)

Q.4 Find $F = ?$ 

~~ABC~~

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

$$\overline{AB} + \overline{BC} + \overline{AC}$$

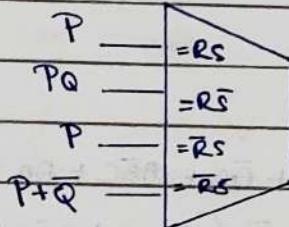
$$\begin{aligned}
 f(A, B, C) &= ABC + A\bar{B}C + \bar{A}BC \\
 &= AB\bar{C} + A\bar{B}C + \bar{A}B\bar{C} + \bar{A}B\bar{C} \\
 &= \sum_m(2, 3, 5, 6)
 \end{aligned}$$

$$\rightarrow P(\bar{Q}+Q)\bar{P}\bar{E}$$

$$P\bar{Q}\bar{R}\bar{C} = 1000$$

$$PQ\bar{R}\bar{C} = 1100$$

Q.5



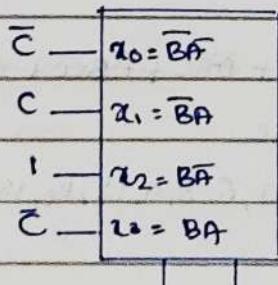
$$\begin{aligned}
 (P, Q, R, S) &= (P+\bar{Q})\bar{R}\bar{S} + \bar{P}R\bar{S} + P\bar{R}\bar{C} + PS \\
 &= \sum_m(0, 8, 9, 11, 12, 13, 14, 15)
 \end{aligned}$$

~~RS~~

00	1			
01				
11	1	1	1	1
10	1	1	1	

$$\Rightarrow PQ + P\bar{S} + \bar{Q}\bar{R}\bar{S}$$

Q.6.



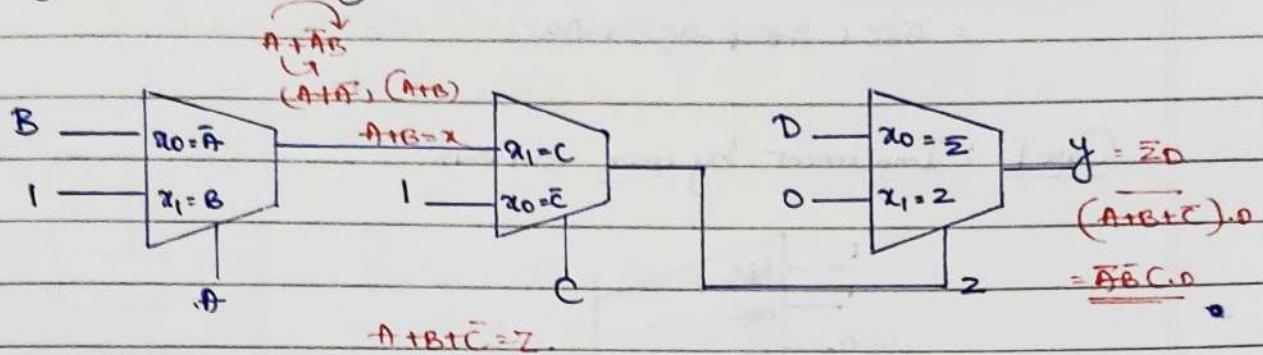
$$\begin{aligned}
 f(A, B, C) &= \bar{B}\bar{A}\bar{C} + \bar{B}\bar{A}C + \bar{B}\bar{A}\bar{C} + \bar{B}\bar{A}C \\
 &= \sum_m(0, 3, 4, 5, 6)
 \end{aligned}$$

① mated

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

② mated

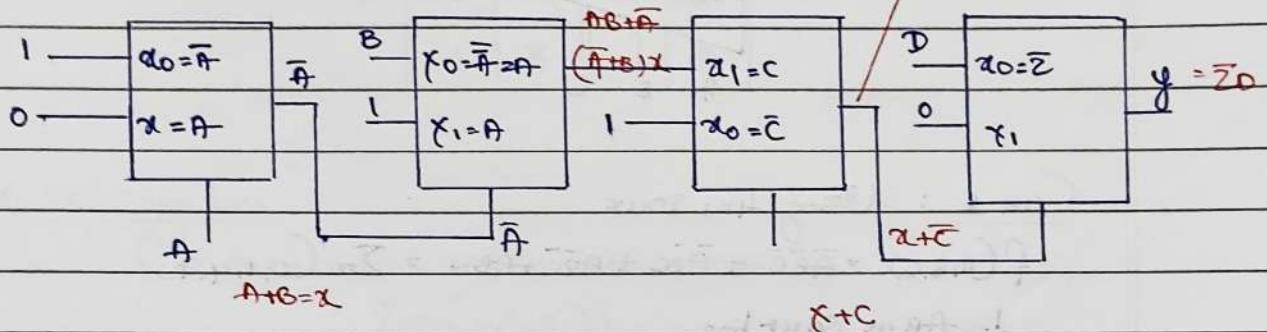
Type 4: Cascading of Mux:



Q1. Find the output of y .

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

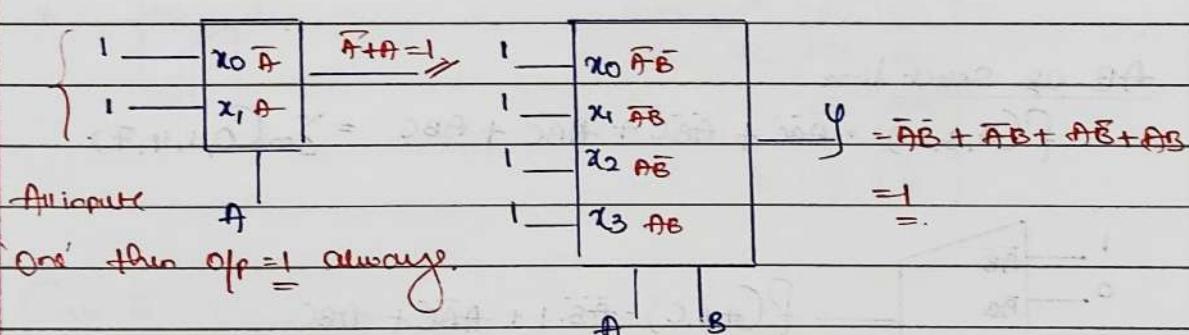
$$xC + \bar{C} = (x + \bar{C})(C + \bar{C}) \\ = x + \bar{C}$$



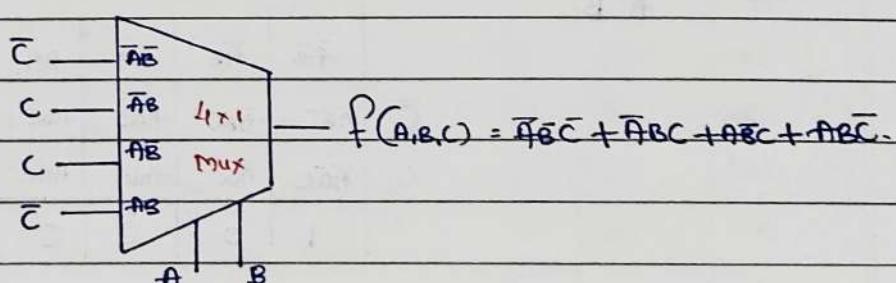
$$y = \bar{Z}D = (\bar{A} + \bar{B} + \bar{C})D = \bar{A}\bar{B}CD$$

Note :

$$NAND = \bar{F}$$

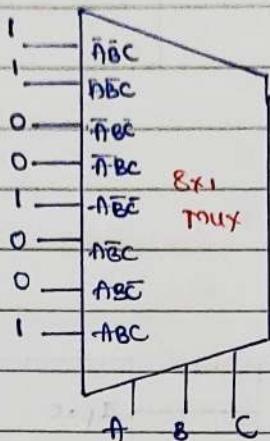


Type 5: Implementation of Function:



Ex. $f(A, B, C) = \sum m(0, 1, 4, 7)$
 $= \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} + ABC.$

Case I : Implement by using 8×1 mux

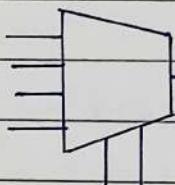


$$f(A, B, C) = \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} + ABC \\ = \sum m(0, 1, 4, 7)$$

Case II : Using 4×1 mux

$$f(A, B, C) = \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + A\bar{B}\bar{C} + ABC = \sum m(0, 1, 4, 7)$$

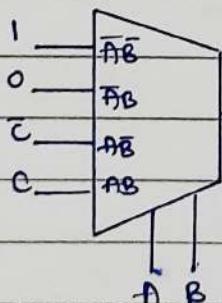
1. AB as select line
2. BC as select line
3. AC as select line



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

1. AB as select line

$$f(A, B, C) = \bar{A}\bar{B} + \bar{A}\bar{B}C + A\bar{B}\bar{C} + ABC = \sum m(0, 1, 4, 7)$$



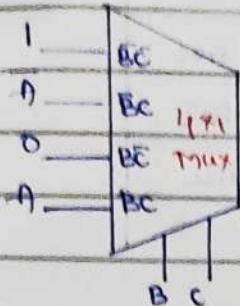
$$f(A, B, C) = \bar{A}\bar{B} \cdot 1 + \bar{A}\bar{B}C + ABC \\ = \bar{A}\bar{B}(\bar{C} + C) + \bar{A}\bar{B}C + ABC \\ = \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + A\bar{B}\bar{C} + ABC$$

	$\bar{A}\bar{B}$	$\bar{A}B$	$A\bar{B}$	AB
\bar{C}	$\bar{A}\bar{B}\bar{C}$ ①	$\bar{A}\bar{B}C$ ②	$A\bar{B}\bar{C}$ ③	$A\bar{B}C$ ④
C	$\bar{A}\bar{B}C$ ⑤	$\bar{A}B C$ ⑥	$A\bar{B}C$ ⑦	ABC ⑧
	1	0	\bar{C}	C

2. BC as Select Line

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

$$= \sum m(0, 1, 4, 7)$$



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

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

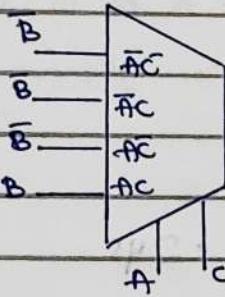
$$= \bar{A}\bar{B}C + A\bar{B}C + \bar{A}\bar{B}C + A\bar{B}C$$

	BC	$\bar{B}C$	$B\bar{C}$	BC
\bar{A}	$\bar{B}C$	0 (red)	$\bar{B}C$	1 (red)
-A	$A\bar{B}C$	4 (red)	$A\bar{B}C$	5 (red)
	1	\bar{A}	0	A

3. AC is select line

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

$$= \sum m(0, 1, 4, 7)$$



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

	$\bar{A}C$	$\bar{A}C$	$A\bar{C}$	AC
\bar{B}	$\bar{A}\bar{B}C$	0 (red)	$\bar{A}\bar{B}C$	1 (red)
B	$\bar{A}\bar{B}C$	2	$\bar{A}\bar{B}C$	3

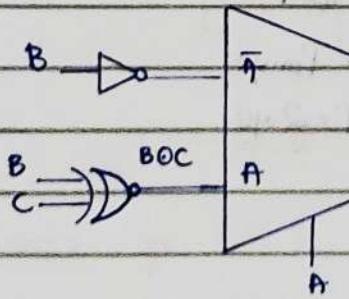
	\bar{B}	B	\bar{B}	B
	B	B	\bar{B}	B

Case III: Using 2x1 Mux

(i) Ā as Select line

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

$$= \sum m(0, 1, 4, 7)$$



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

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

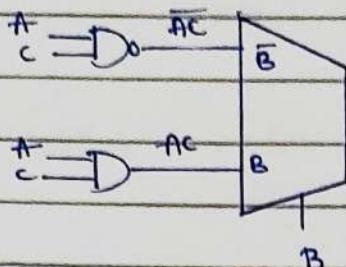
$$+ ABC$$

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

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

	\bar{A}	A		
$\bar{B}C$	$\bar{A}\bar{B}C$	0 (red)	$A\bar{B}C$	4 (red)
$\bar{B}\bar{C}$	$\bar{A}\bar{B}C$	1	$A\bar{B}C$	5
$B\bar{C}$	$A\bar{B}C$	2	$A\bar{B}C$	6
$B\bar{C}$	$A\bar{B}C$	3	$A\bar{B}C$	7

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

(ii) B as select line

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

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

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

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

$$= \underline{\bar{A}C}$$

~~H/W~~(iii) C as select line $f(A, B, C) = \sum m(0, 1, 4, 7)$

Q1. $f(A, B, C, D) = \sum m(0, 1, 3, 5, 7, 9, 12, 15)$

1. ABD is select line $\rightarrow 16 \times 1 \text{ mux}, 8 \times 1 \text{ mux}, 4 \times 1 \text{ mux}$ 2. ACD is select line $2 \times 1 \text{ mux}$

(AB, AC, AD or select line)

A as select line.

Type-6: Decay in Mux

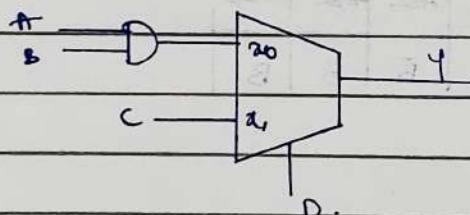
$$T_{AND} = 14\mu s, T_{mux} = 24\mu s$$

Case 1: D=0

$$T = T_{AND} + T_{mux}$$

$$= 14\mu s + 24\mu s$$

$$T = \underline{38\mu s}$$

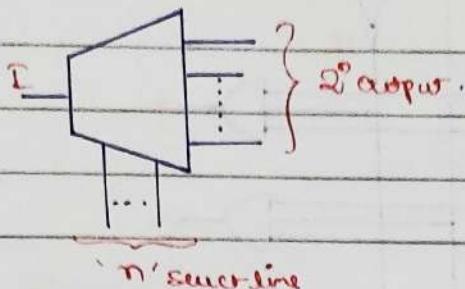


Case 2: D=1

$$T = T_{mux}$$

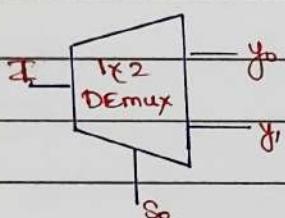
$$T = \underline{24\mu s}$$

DE-mux: DE-mux is called AND Logic



Q1. Design 1×2 DE-mux

Step 1:



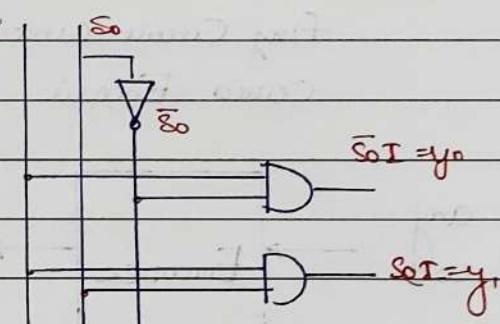
Step 2:

S0	y0	y1
0	I	0
1	0	I

$$\text{Step 3: } y_0 = \bar{S}_0 I$$

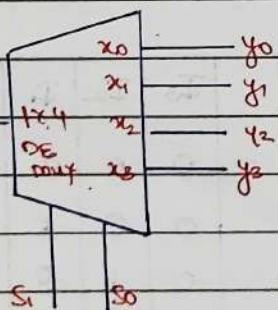
$$y_1 = S_0 I$$

Step 5:



Q2. Design a 1×4 DEMUX?

Step 1:



Step 2:

S1	S0	y0	y1	y2	y3
0	0	I	0	0	0
0	1	0	I	0	0
1	0	0	0	I	0
1	1	0	0	0	I

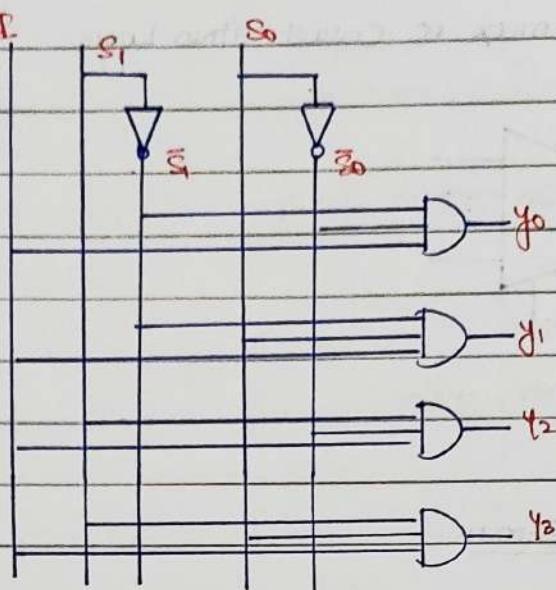
$$\text{Step 3: } y_0 = \bar{S}_1 \bar{S}_0 I$$

$$y_1 = \bar{S}_1 S_0 I$$

$$y_2 = S_1 \bar{S}_0 I$$

$$y_3 = S_1 S_0 I$$

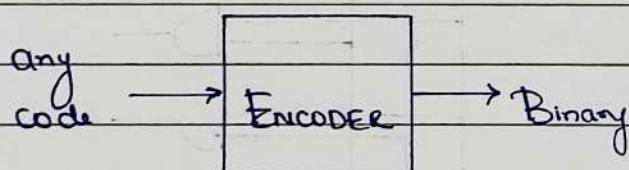
Step 4:

Step 5:Q.W.

- Q. Design a 1×8 DEMUX
 Q. Design a 1×16 DEMUX.

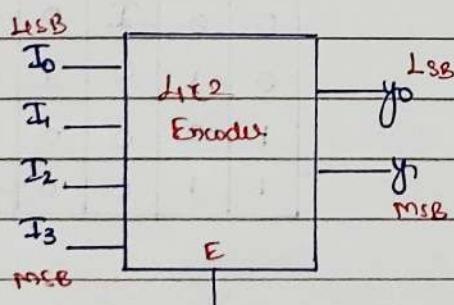
Encoder:

A logic circuit which is used to convert any code into Binary is called Encoder.



1) 4×2 Encoder (Quarto Binary)
 2) 8×3 Encoder (Octo to Binary)
 3) 16×4 Encoder (Hexa to Binary).

- Q. Design a 4×2 Encoder.

Step 1:Step 2:

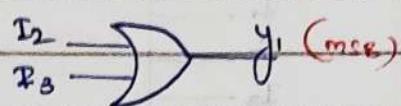
I ₃	I ₂	I ₁	I ₀	y ₁	y ₀
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1

Step 3: $y_0 = I_1 + I_3$

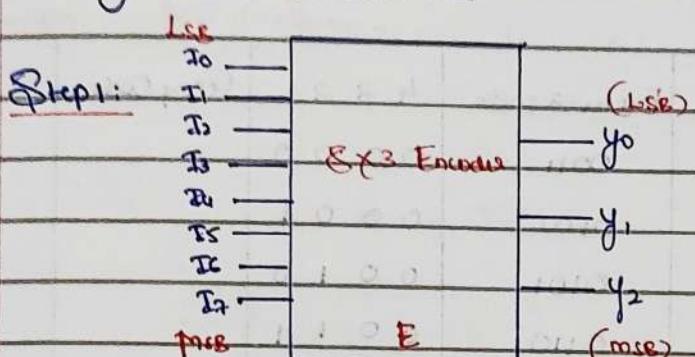
$$y_1 = I_2 + I_3$$

Step 4:

Step 5:



Q. Design a 8×3 Encoder.



HW
Q. Design a 16×4 Encoder

Step 2:

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

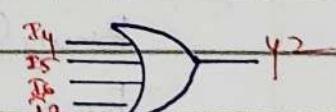
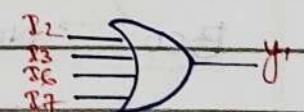
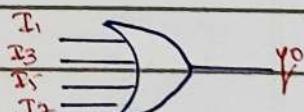
Step 3: $y_0 = I_1 + I_3 + I_5 + I_7$

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

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

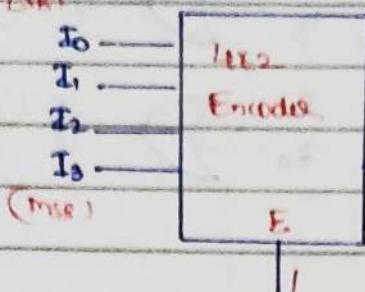
Step 4:

Step 5:



Note: Priority Encoder.

(L&R)



L&R Priority Encoder.

(MSB)	I ₃	I ₂	I ₁	I ₀ (L&R)	y ₁	y ₀
	x	x	x	1	0	0
	x	x	1	0	0	1
	x	1	0	0	1	0
	1	0	0	0	1	1

→ weighted, Self-complemented

Decimal	BCD	Exten-3 code	4 2 3 1	Gray Code
0	0000	0011	00 00	
1	0001	0100	00 01	
2	0010	0101	00 10	
3	0011	0110	00 11	
4	0100	0111	01 10	
5	0101	1000	10 01	
6	0110	1001	11 00	
7	0111	1010	11 01	
8	1000	1011	11 10	
9	1001	1100	11 11	

Decimal → Weighted code
→ Self Complemented

$\tau \rightarrow$ Base (Radix)

Complement

($n-1$)'s

→ Complement

$\tau=2$

1's

2's

$\tau=8$

7's

8's

$\tau=10$

9's

10's

BCD → Weighted Code
→ Not a Self Complement Code.

→ Binary Coded Decimal

Each decimal number is represented by 4 bits.

eg. { Decimal - 0
BCD - 0000

{ Decimal - 9
BCD - 1001

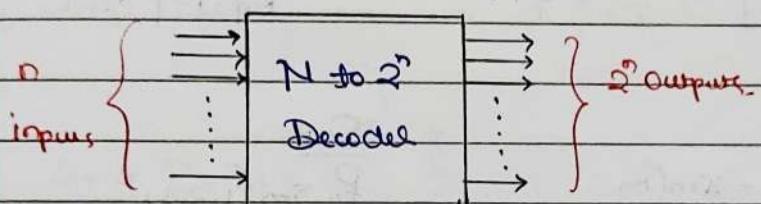
{ Decimal - 29
BCD - 0010 0001

Excess - 3 code:

- It is not a weighted code
- Sub Complemented code

Decoder: Circuit which is used to convert Binary into any other code.

1. 2×4 Decoder
2. 3×8 Decoder
3. 4×16 Decoder.



2×4 Decoder.

E-1

Step 1:

(L _B) I ₀	2×4 Decoder	y ₀ (L _B)
		y ₁
(M _B) I ₁		y ₂ (M _B)

Step 2:

I ₁	I ₀	y ₀	y ₁	y ₂	y ₃
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0

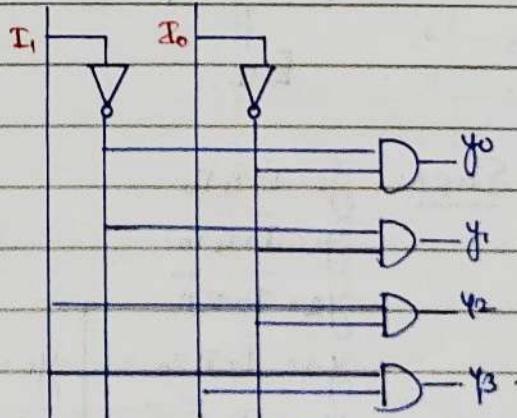
Step 3: $y_0 = \bar{I}_1 I_0$

$$y_1 = \bar{I}_1 \bar{I}_0$$

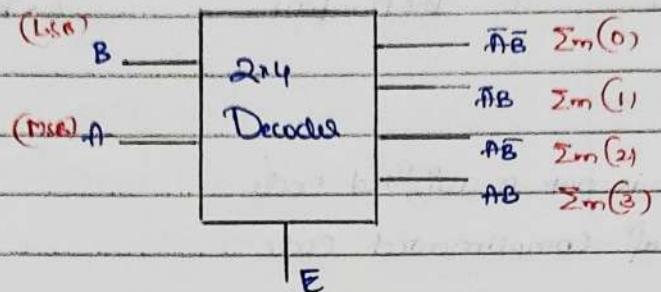
$$y_2 = I_1 \bar{I}_0$$

$$y_3 = I_1 I_0$$

Step 4:



Step 4: minimization

Example:

eg:

		A	B	P
(LSB)	$\bar{A}B$	0	0	0
B.	$\bar{A}B$ 0	0	1	1
(MSB)	$\bar{A}B$ 1	1	0	1
	$\bar{A}B$ 2	1	1	1
	$\bar{A}B$ 3			

Pos =

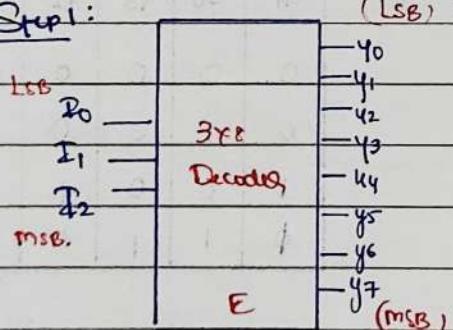
$$f = (\bar{A}B) = \Sigma m(0)$$

SOP

$$\overline{f} = \Sigma m(1, 2, 3)$$

3x8 Decoder

Step 1:



Step 2:

I ₂	I ₁	I ₀	y ₂	y ₆	y ₅	y ₄	y ₃	y ₂	y ₁	y ₀
0	0	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0	1	0	0
0	1	1	0	0	0	0	0	1	0	0
1	0	0	0	0	0	1	0	0	0	0
1	0	1	0	0	1	0	0	0	0	0
1	1	0	1	0	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0

$$y_0 = \bar{I}_2 \bar{I}_1 \bar{I}_0$$

$$y_1 = \bar{I}_2 \bar{I}_1 I_0$$

$$y_2 = \bar{I}_2 I_1 \bar{I}_0$$

$$y_3 = \bar{I}_2 I_1 I_0$$

$$y_4 = I_2 \bar{I}_1 \bar{I}_0$$

$$y_5 = I_2 \bar{I}_1 I_0$$

$$y_6 = I_2 I_1 \bar{I}_0$$

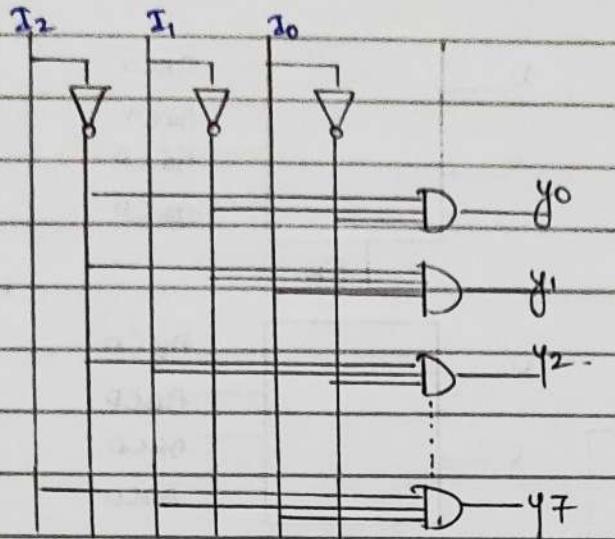
$$y_7 = I_2 I_1 I_0$$

Step 4: Minimization.

Steps : Hardware Implementation

三

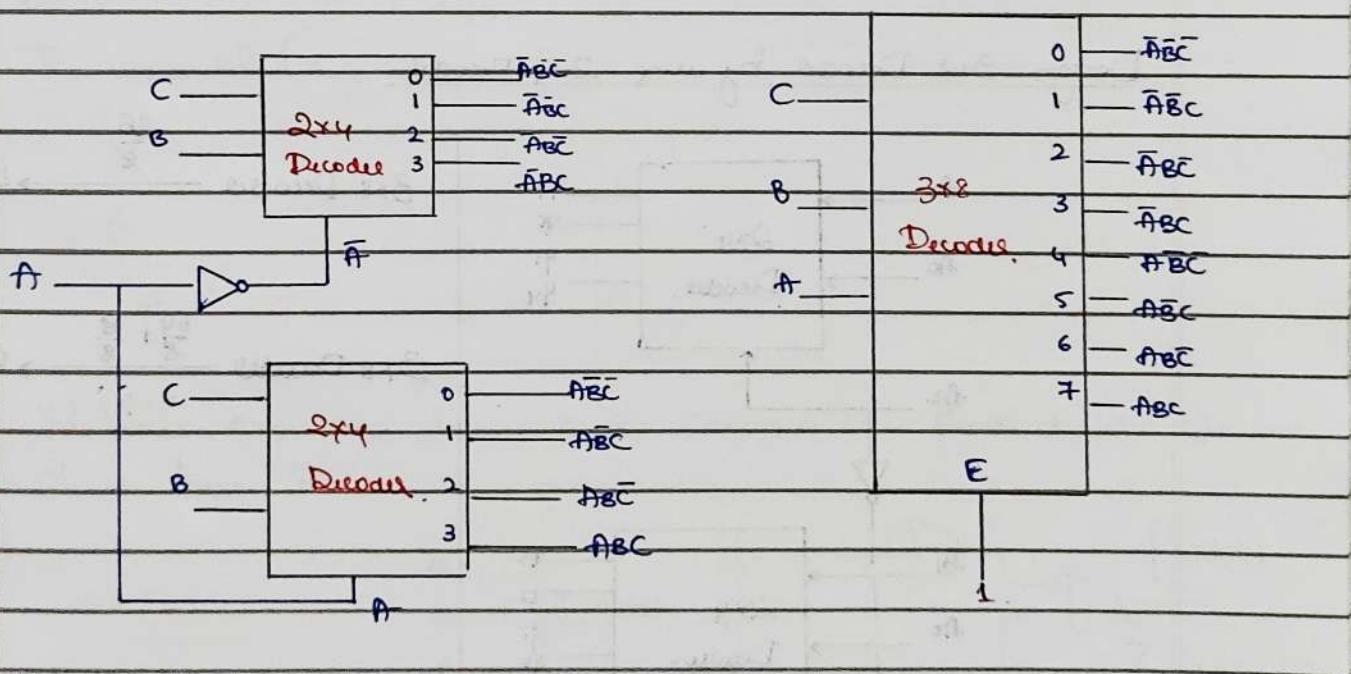
Design 4x16 Decoder



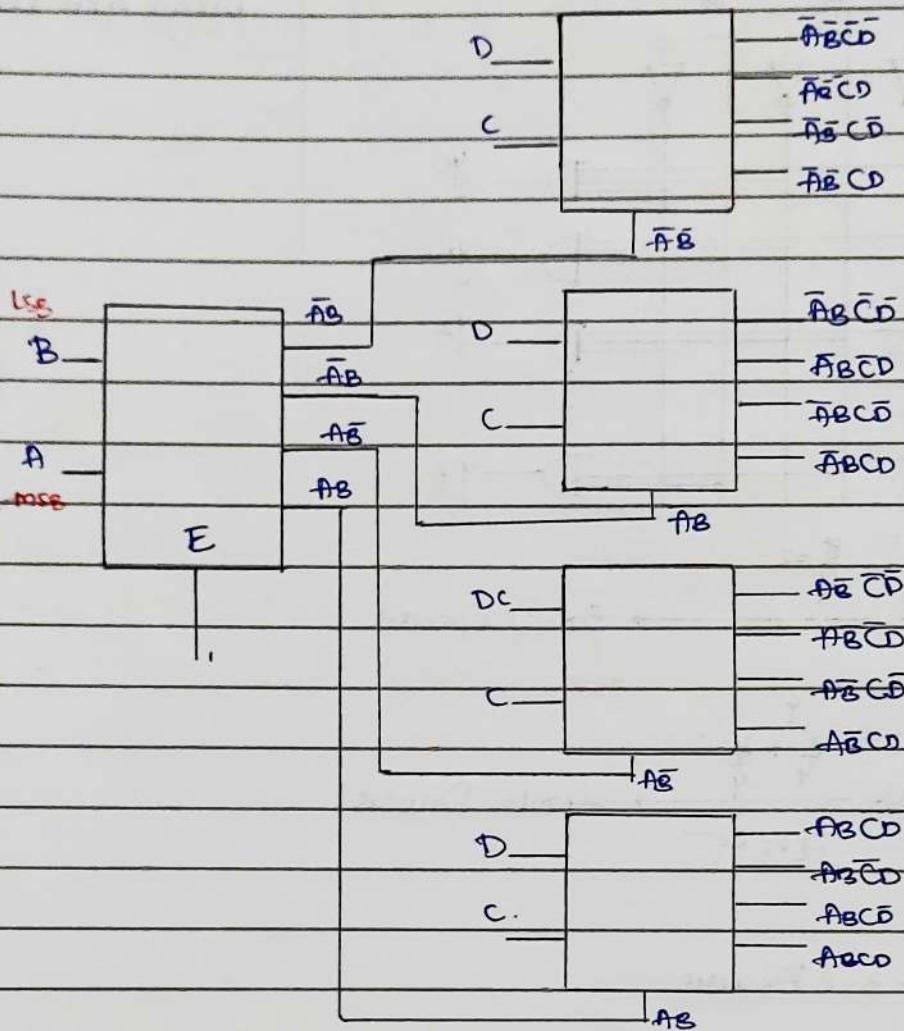
Q.1 2×4 Decoder $\xrightarrow{\frac{S}{4} = 2 \checkmark \frac{m}{n}}$ 3×8 Decoder
 Input \uparrow Output

$$Q.2 \quad \text{2x4 Decoder} \xrightarrow{\text{Input}} \text{1x16 Decoder}$$

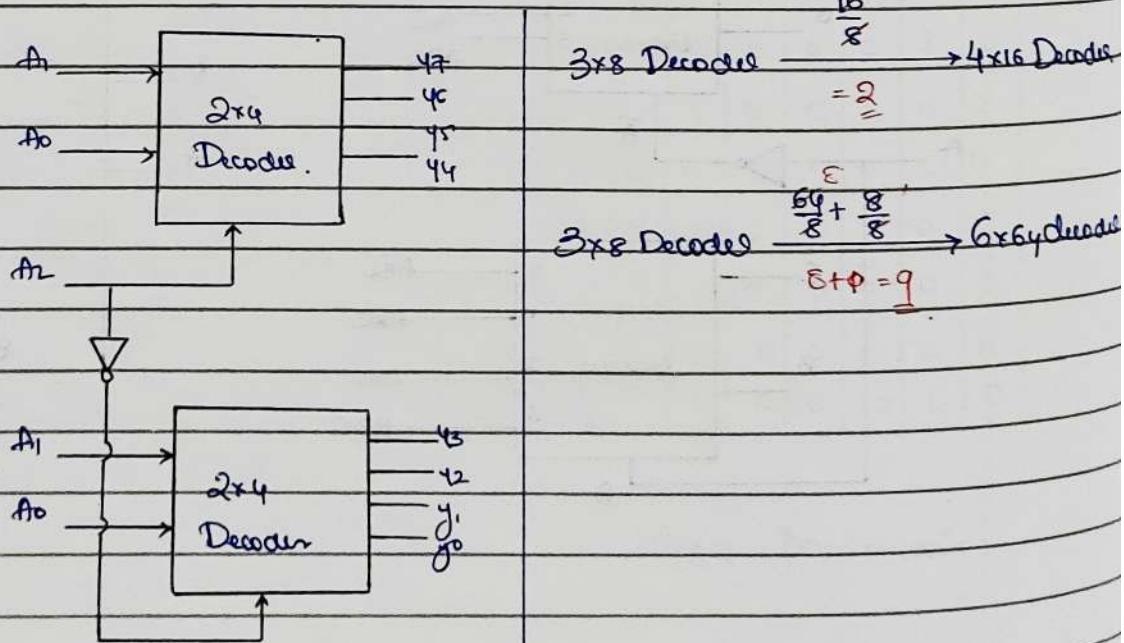
$2 \times 4 \rightarrow 3 \times 8$ Decoder



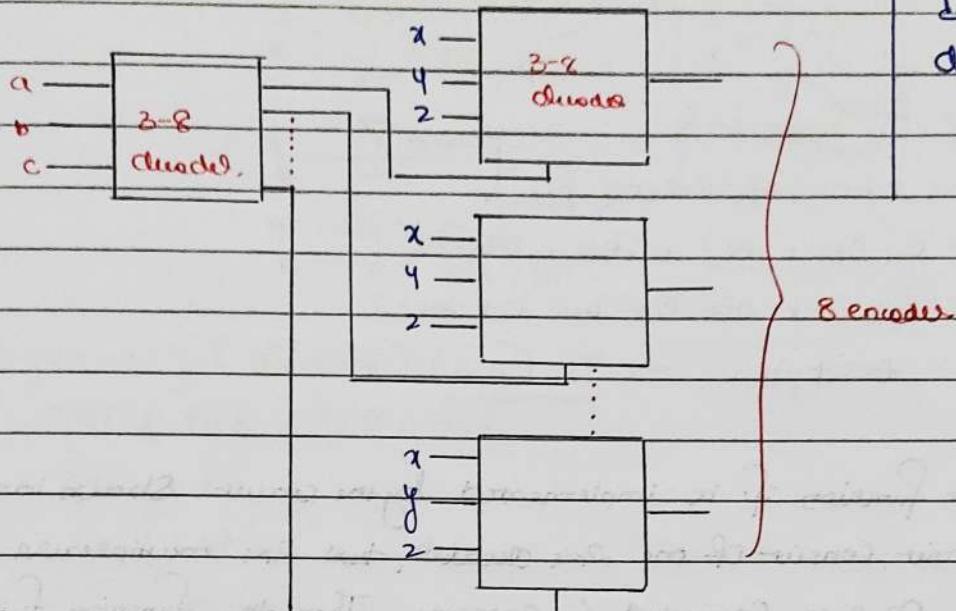
2x4 Decoder \rightarrow 4x16 Decoder



Design 3x8 Decoder by using 2x4 Decoder.



3x8 Decoder → 6x64 Decoder



Design Binary to decimal converter.

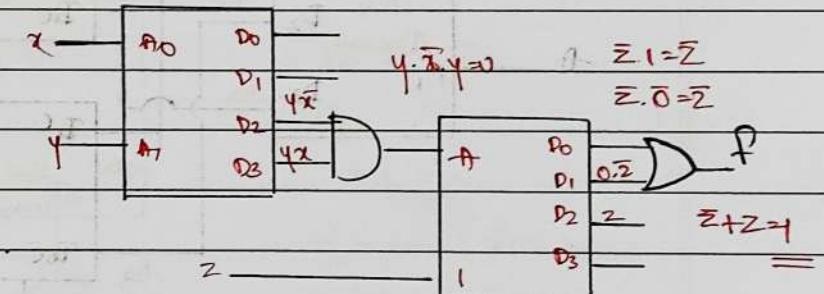
Q.1 A logic circuit consists of 2x4 diodes as shown in the figure. The output are as follows:

$$D_0 = 1 \text{ when } A_0 = 0, A_1 \neq 0$$

$$D_1 = 1 \text{ when } D_0 = 1, A_1 = 0$$

$$D_2 = 1 \quad " \quad A_0 = 0, A_1 = 1$$

$$D_3 = 1 \quad " \quad A_0 = 1, A_{1,1}$$



The value of (x, y, z) is.

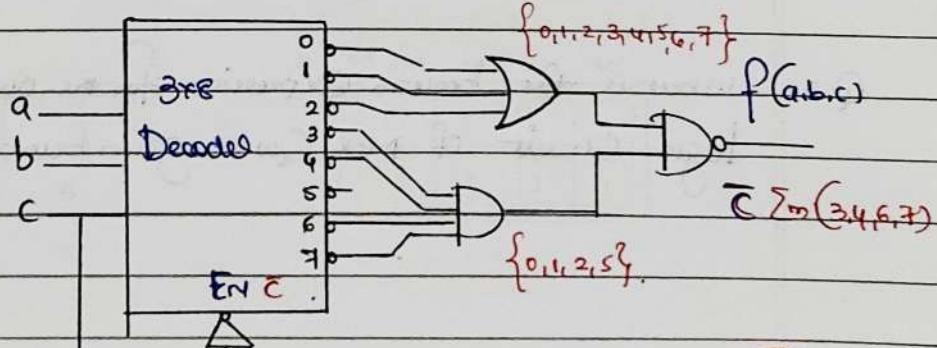
Ans

f_1	f_0	y
0	0	$D_0 = 1$
0	1	D_1
1	0	D_2
1	1	$D_3 = -$

Q2 The boolean expression $f(a,b,c)$ in its Canonical Form for the decode circuit shown below is

~~Ans.~~

$$\Rightarrow \sum_m (4, 6)$$

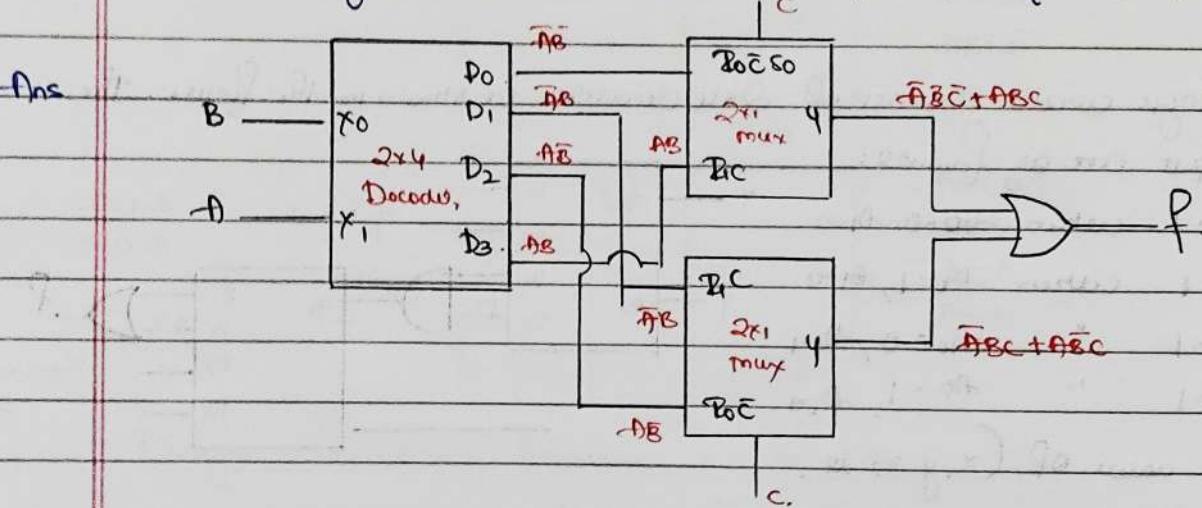


$$\begin{array}{c} \{0,1,2,3,4,5,6,7\} \\ \quad \quad \quad D \\ \{0,1,2,3\} \end{array} \quad \begin{array}{c} \{0,1,2,3,4,5,6,7\} \\ \quad \quad \quad D \\ \{0,1,2,3\} \end{array} \quad \begin{array}{c} \{3,4,5,6,7\} \\ \quad \quad \quad D \\ \{0,1,2,3\} \end{array}$$

$$P = C \cdot \Sigma m(3, 4, 6, 7)$$

- = $C [ABC + A\bar{B}C + \bar{A}BC + \bar{A}\bar{B}C]$
- = $\bar{C} [\bar{A}BC + A\bar{B}C + \bar{A}BC + \bar{A}\bar{B}C]$
- = $\bar{A}BC\bar{C} + A\bar{B}C\bar{C} + \bar{A}BC\bar{C} + \bar{A}\bar{B}C\bar{C}$
- = $A\bar{B}C + \bar{A}BC \rightarrow \Sigma m(4, 6)$

Q3. A logic function f_1 is implemented by the circuit shown in the figure below. The circuit consists of one 2×4 decoder, two 2×1 multiplexers and a two input OR gate connected in cascade. Then the function f_1 is equal to



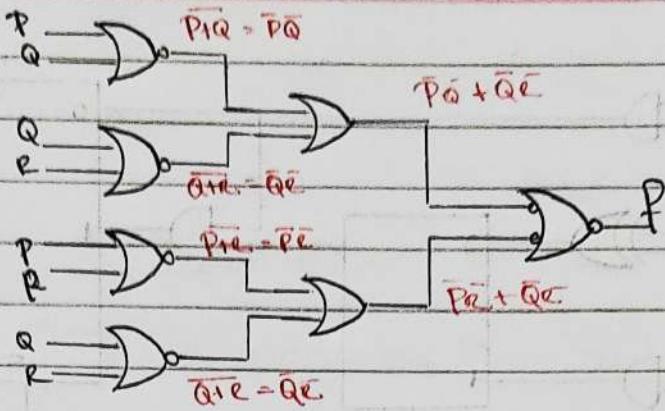
$$f_1 = \bar{A}\bar{B}C + ABC - \bar{A}BC - A\bar{B}C$$

$$= \Sigma m(0, 3, 4, 7)$$

~~$$X \quad \begin{matrix} 00 \\ 01 \\ 11 \\ 10 \end{matrix} \quad \Rightarrow \bar{B}C + BC = BOC$$~~

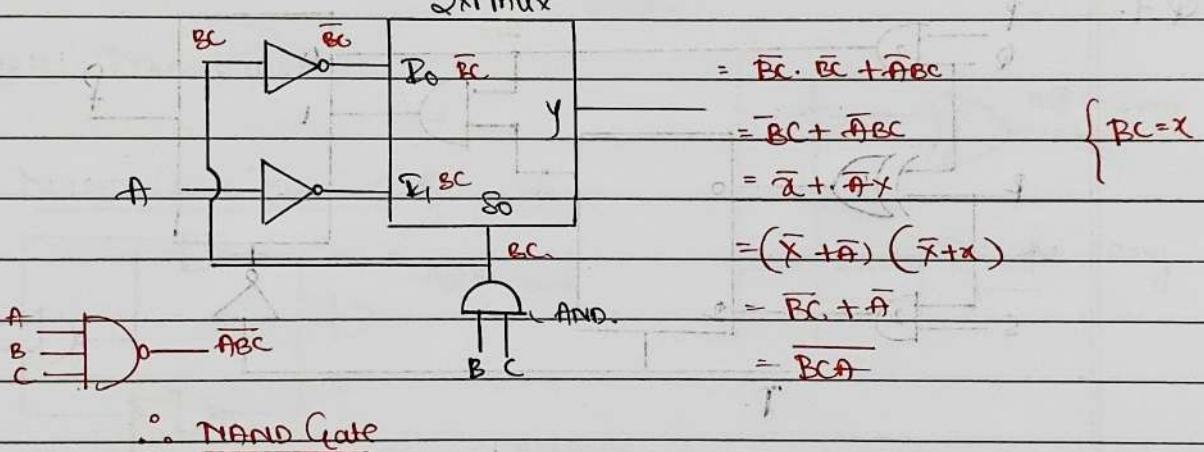
0	1		1
1	1		1

Q2. What is the Boolean expression for the output f of the combinational logic circuit of NOR Gate given below.



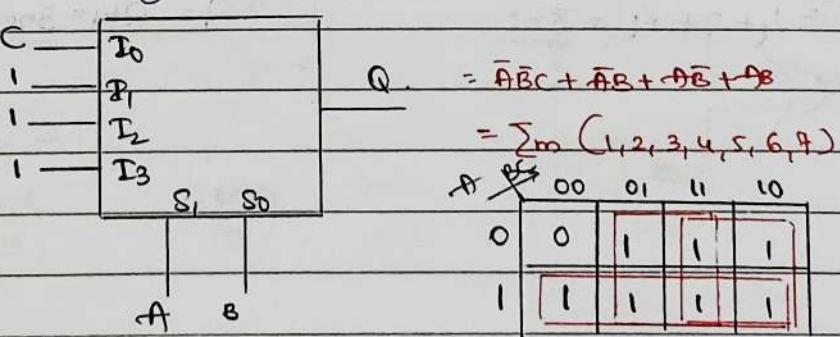
$$\begin{aligned}
 f &= [\bar{P}\bar{Q} + \bar{Q}\bar{R}] \cdot [\bar{P}\bar{R} + \bar{Q}\bar{R}] \\
 &= \bar{P}\bar{Q}\bar{R} + \bar{P}\bar{Q}\bar{R} + \bar{P}\bar{Q}\bar{R} + \bar{Q}\bar{R} \\
 &= \bar{P}\bar{Q}\bar{R} + \bar{Q}\bar{R} \\
 &= \bar{Q}\bar{R}[1 + \bar{P}] \\
 &= \bar{Q}\bar{R} \quad \Rightarrow \underline{\underline{Q+R}}
 \end{aligned}$$

Q4. The Combinational Circuit given below implements which of the following.



Qs. The Combinational logic circuit shown in the given figure has an output Q

Celthick is



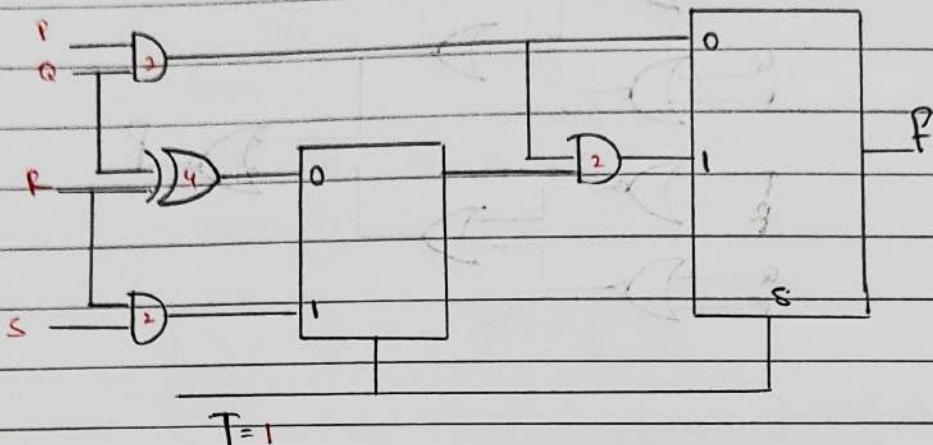
$$= \bar{A}\bar{B}C + \bar{A}B + A\bar{B} + AB$$

$$= \sum_{S \subseteq \{1, 2, 3, 4, 5, 6, 7\}} C_S$$

	00	01	11	10
0	0	1	1	1
1	1	1	1	1

$$\rightarrow \underline{A + B + C}$$

Q6.

Case 1 : $T=0$

$$T = T_{\text{prop}} + T_{\text{mux}}$$

$$T = 2+1 = \underline{\underline{3 \text{ ns}}}$$

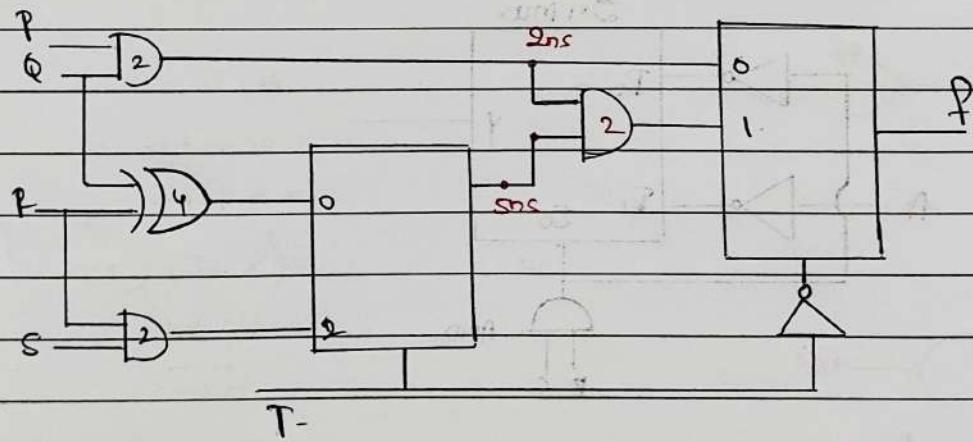
Case 2 : $T=1$

$$T = T_{\text{prop}} + T_{\text{mux}} + T_{\text{prop}} + T_{\text{mux}}$$

$$= 2+1+2+1 = \underline{\underline{6 \text{ ns}}}$$

$$\therefore T_{\text{max}} = \underline{\underline{6 \text{ ns}}}$$

Q7.

Case (1) : $T=0$

$$T = T_{\text{prop}} + T_{\text{mux}} + T_{\text{prop}} + T_{\text{mux}}$$

$$= 1+2+1+1 = \underline{\underline{8 \text{ ns}}}$$

Case (2) : $T=1$

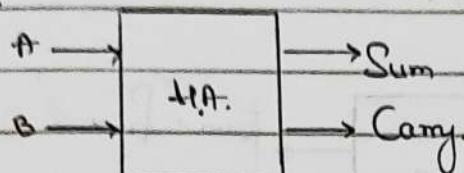
$$T = T_{\text{prop}} + T_{\text{mux}}$$

$$T = 2+1 = \underline{\underline{3 \text{ ns}}}$$

Half Adder

$$\begin{array}{r}
 & & & 0 \\
 & 0 & 0 & 1 & 1 \\
 + & 1 & 0 & 1 & 0 \\
 \hline
 0 & 0 & 1 & 0
 \end{array}
 \quad
 \begin{array}{r}
 0 \\
 -10 \\
 \hline
 00
 \end{array}
 \quad
 \begin{array}{l}
 \text{Sum} \\
 \text{Carry}
 \end{array}$$

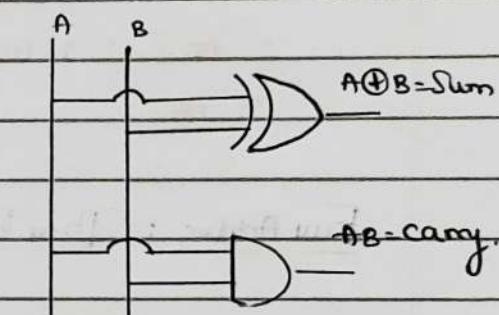
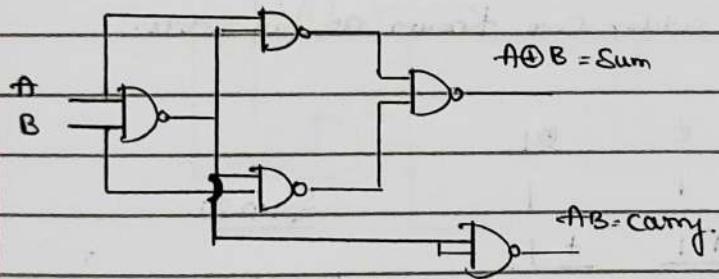
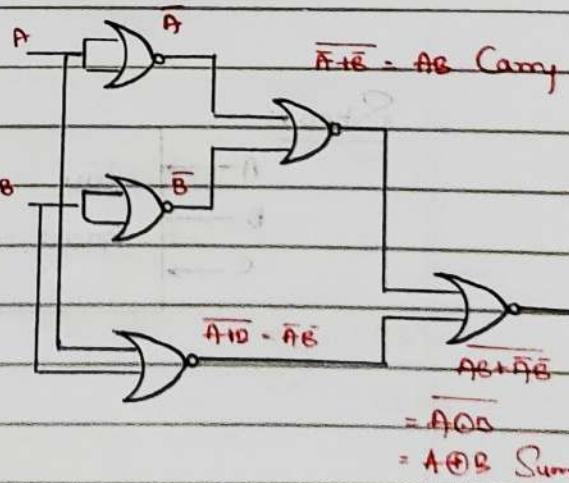
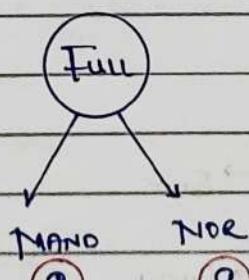
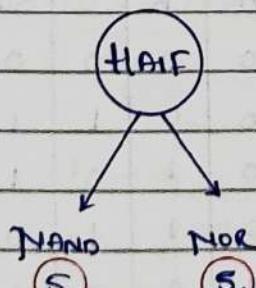
* Two bit adder are known as half adder.

Step 1:Step 2:

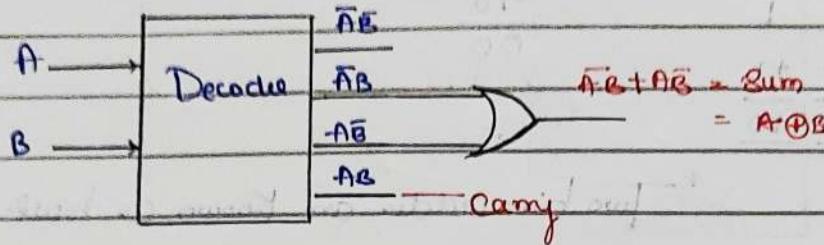
A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Step 3: $\text{Sum} = \overline{A}B + A\overline{B} = A \oplus B$

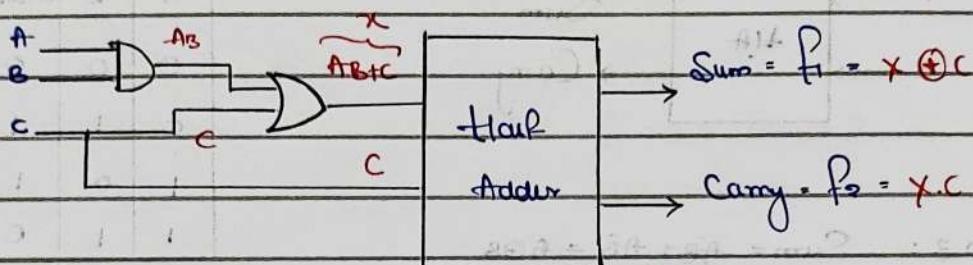
$\text{Carry} = AB$

Step 4:Minimization.NAND Gate Required = 5By NOR Gate

Half Adder By Using Decoder.



Q1. What is the value of f_1 and f_2 ?



$$f_1 = (AB + C) \oplus C$$

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

$$= \overline{AB} \cdot \bar{C} \cdot C + AB\bar{C} + C\bar{C}$$

$$= \underline{\underline{ABC}}$$

$$f_2 = (AB + C)C$$

$$= ABC + C \cdot C = ABC + C$$

$$= C[ABC]$$

$$= \underline{\underline{CAB}}$$

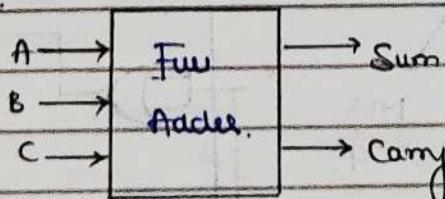
Full Adder : Three bit adder are known as full adder.

$$\begin{array}{r}
 & 0 & 0 & 0 & 01 \\
 & 0 & 0 & 1 & 1 \\
 + 0 & + 1 & + 1 & + 1 \\
 \hline
 00 & 01 & 10 & 11
 \end{array}$$

Step 2:

A	B	C	Sum	Carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

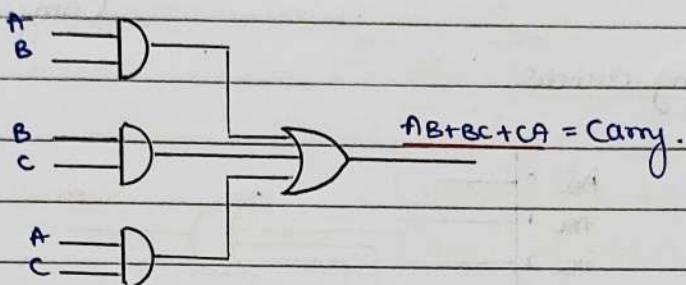
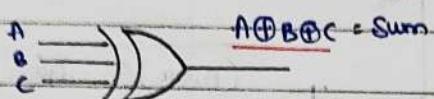
Step 1:



Step 3: $\bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}C + AB\bar{C}$

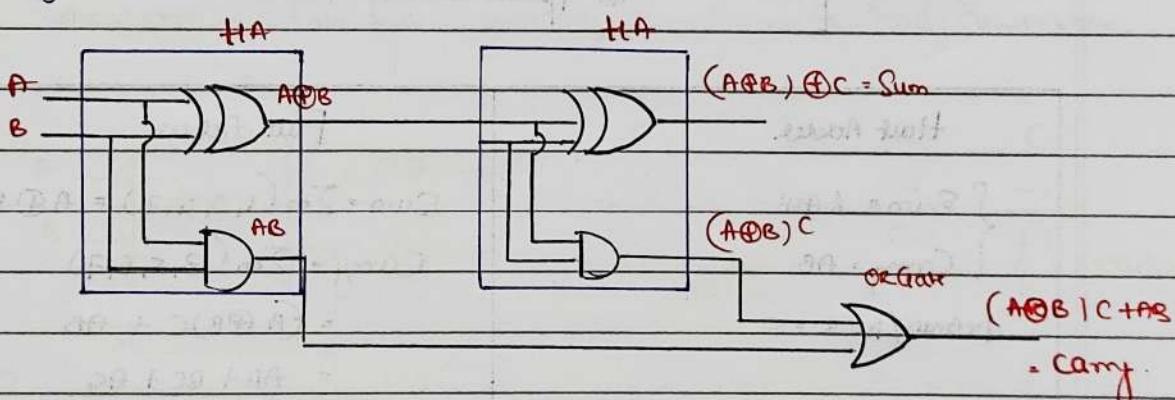
$$\text{(Sum)} = \sum m(1, 2, 4, 7)$$

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

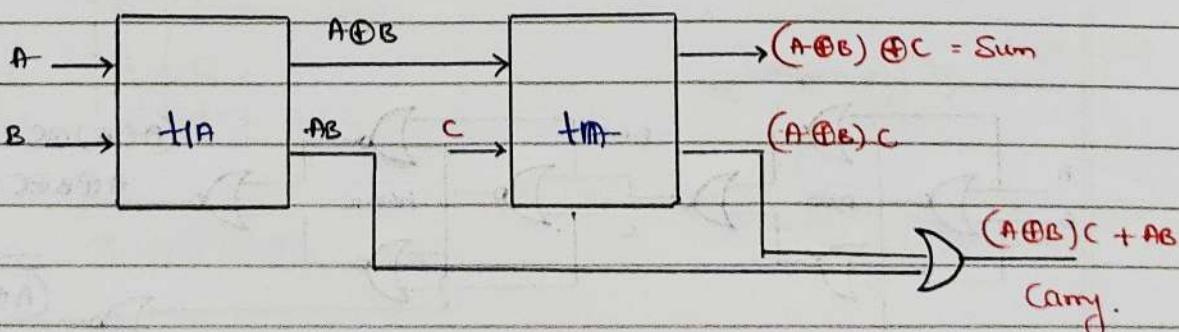
Step 5:

$$\text{Sum} = (A \oplus B) \oplus C$$

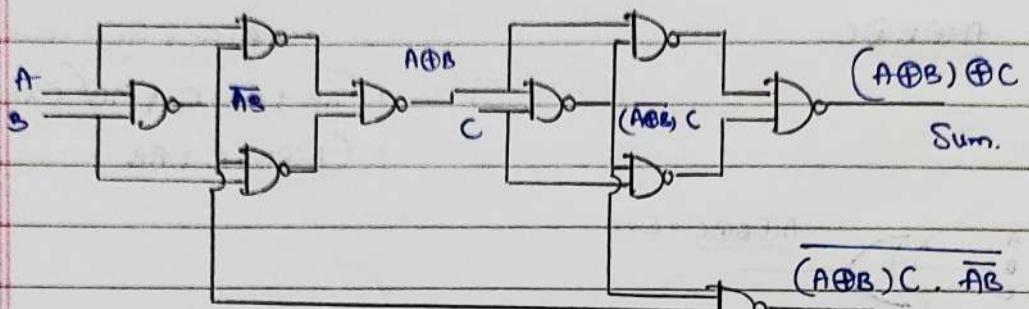
$$\text{Carry} = (A \oplus B)C + AB$$



1 Full Adder = 2 HA + 1. OR Gate

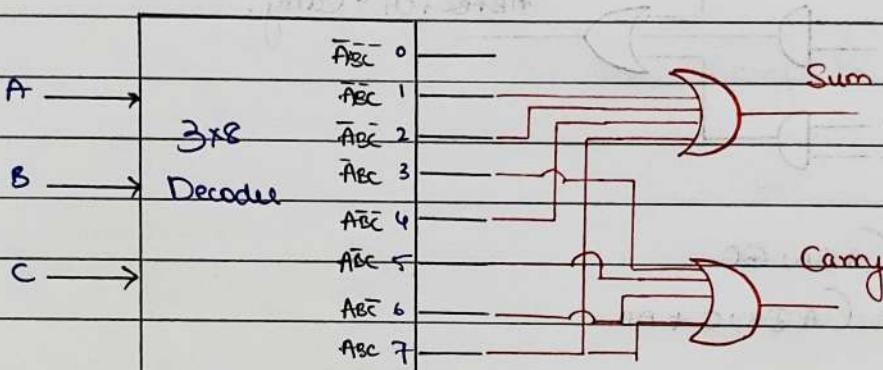


Full Adder by using NAND Gate.

HW

Full Adder by
using NOR
GATE.

Full Adder by using decoder.



Half Adder.

$$\begin{cases} \text{Sum} = A \oplus B \\ \text{Carry} = AB \end{cases}$$

NAND/NOR = 5.

Full Adder.

$$\text{Sum} = \sum m(1, 2, 4, 7) = A \oplus B \oplus C$$

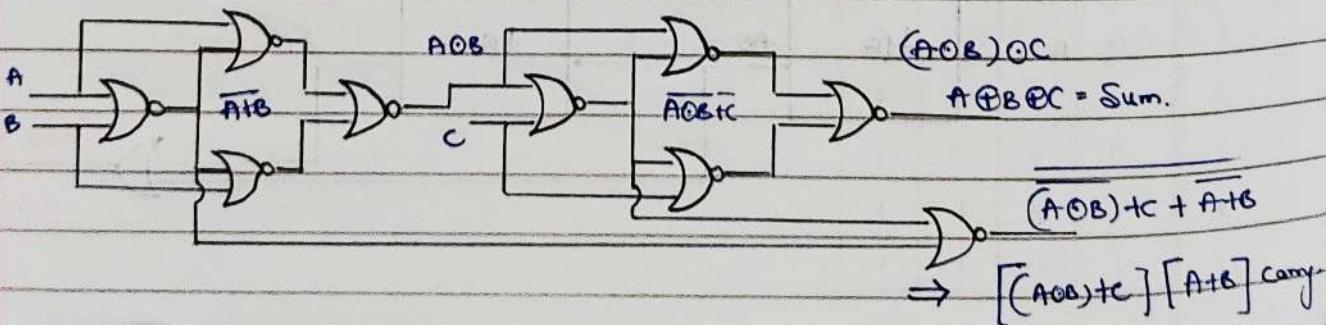
$$\text{Carry} = \sum m(3, 5, 6, 7)$$

$$= (A \oplus B)C + AB$$

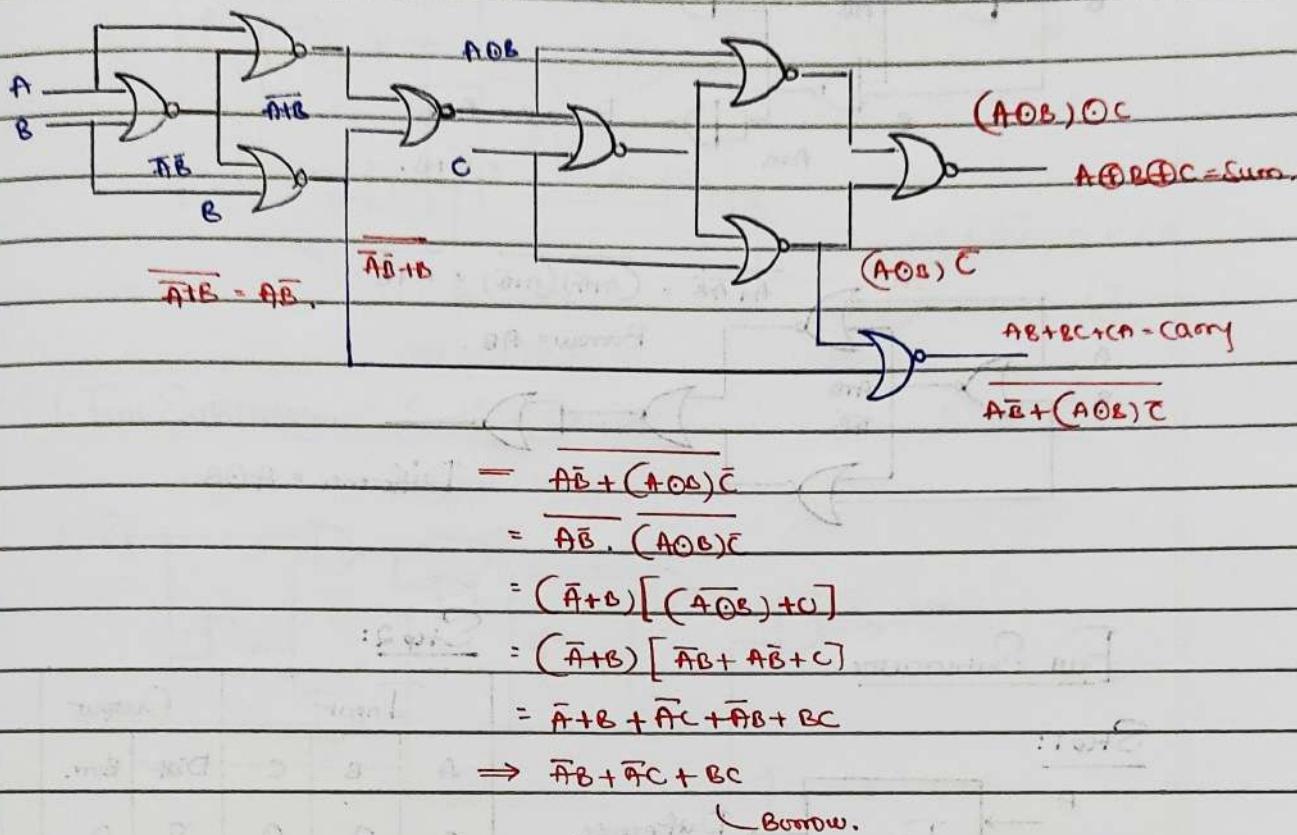
$$= AB + BC + AC$$

$$\text{NAND/NOR} = 9.$$

$$\text{Full} = 2 + 4 + 1 \text{ OR}$$

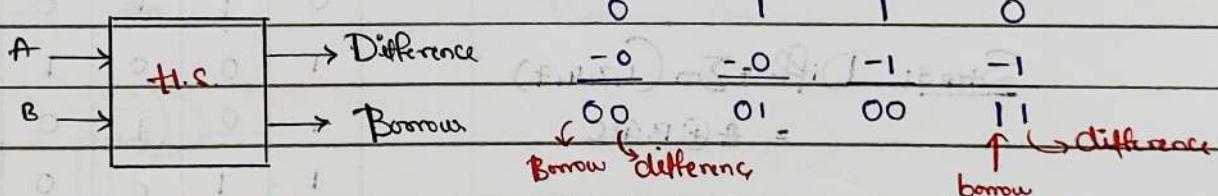


Full Subtractor



Half Subtractor : Two bit Subtractor are known as half subtractor.

Step 1:

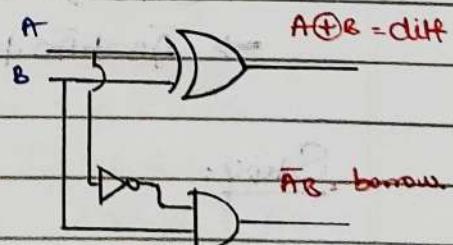


Step 2:

A	B	Dif.	Borrow
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

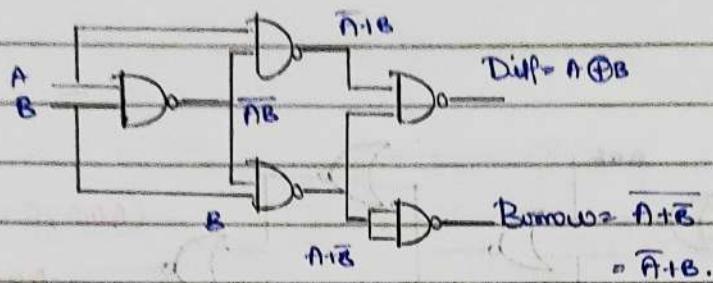
Step 4:

Step 5:

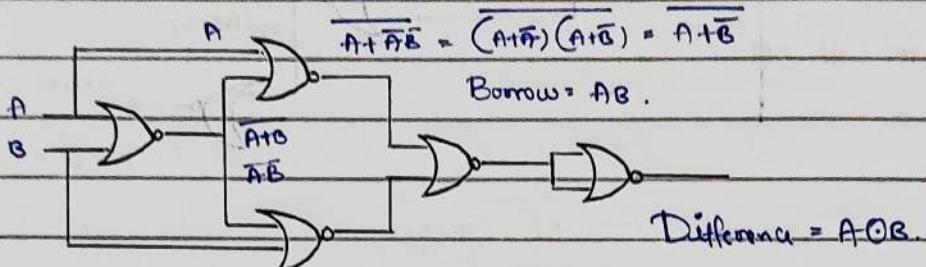


Step 3: $\text{Diff} = A \oplus B$

$\text{Borrow} = \overline{AB}$.

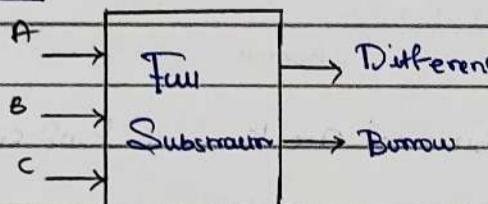


Half Subtractor using NAND Gates.



Full Subtractor.

Step 1:



Step 2:

Input			Output	
A	B	C	Diff	Borr.
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

$$\text{Step 3: } \text{Diff} = \sum m(1, 2, 4, 7)$$

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

$$\text{Borrow} = \sum m(1, 2, 3, 7)$$

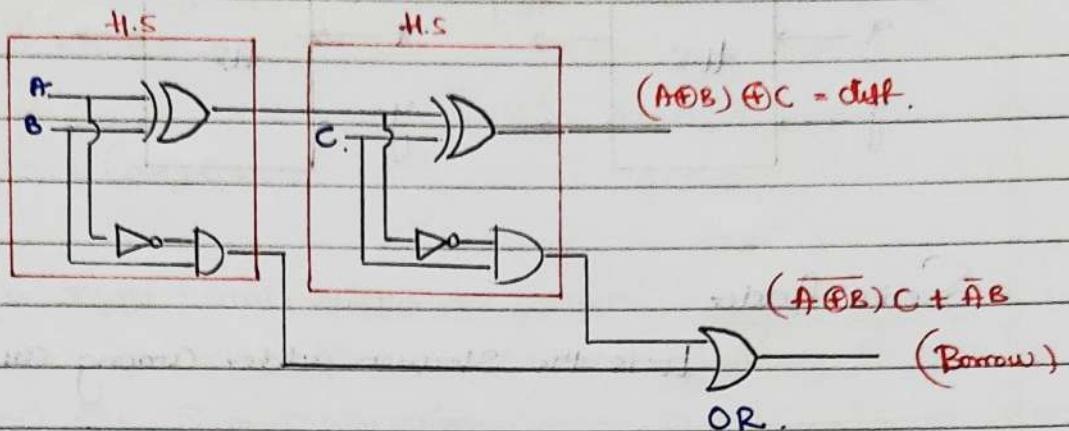
$$= \bar{A}\bar{B}C + \bar{A}B\bar{C} + \bar{A}BC + ABC$$

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

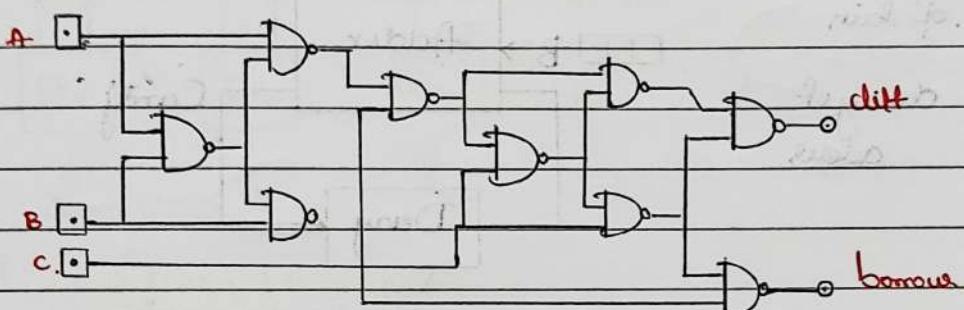
$$= (\bar{A} \oplus B)C + \bar{A}B \rightarrow \text{Semi minimized expression}$$

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

Step 4:

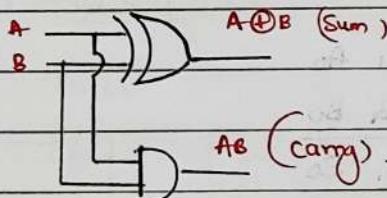
Steps:

$$\boxed{1 \text{ Full Subtractor} = 2 \cdot \text{H.S} + \text{Or Gate.}}$$

Half adder

$$\text{Sum} = A \oplus B$$

$$\text{Carry} = AB$$



NAND
NOR

Full Adder

$$\text{Sum} = A \oplus B \oplus C$$

$$\text{Carry} = \sum_m(3, 5, 6, 7)$$

$$= (A \oplus B)C + AB$$

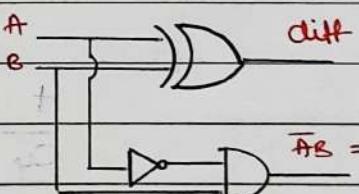
$$= AB + AC + BC$$

$$\text{NAND/NOR} = 9.$$

Half Subtractor

$$\text{Diff} = A \oplus B$$

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



NAND
NOR

Full Subtractor

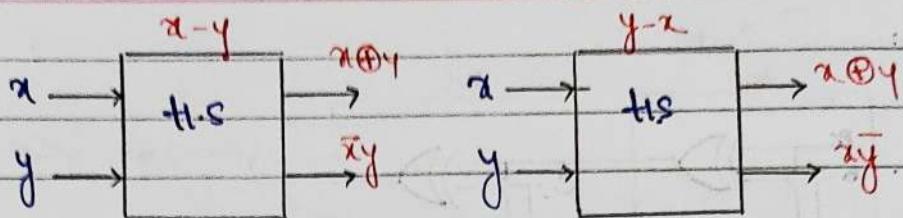
$$\text{Difference} = A \oplus B \oplus C$$

$$\text{Borrow} = \sum_m(1, 2, 3, 7)$$

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

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

$$\text{NAND/NOR} = 9.$$



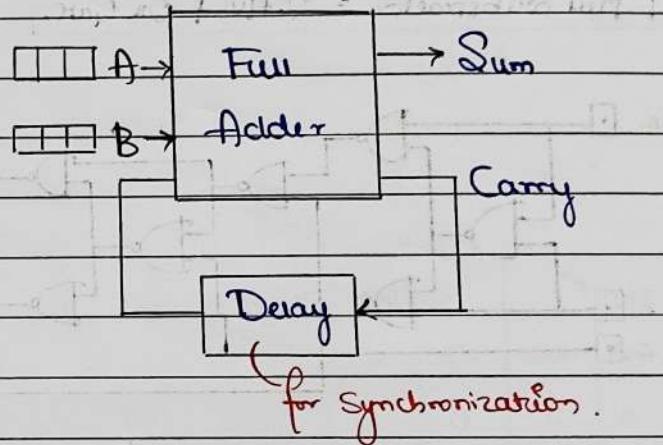
Serial Adder "One by one addition"

→ It is the slowest adder among all the adder.

$$T = n \cdot T_{delay}$$

$n \rightarrow$ no. of bits

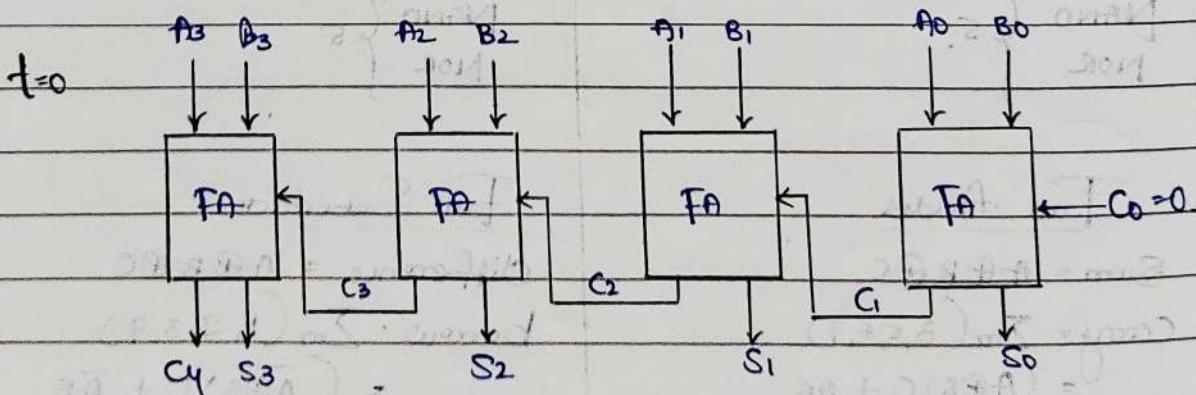
$T_{delay} \rightarrow$ delay of adder.



Parallel Adder (Ripple Carry Adder)

→ Inputs are Connected Simultaneously

$$\begin{array}{r}
 & C_3 & C_2 & C_1 & C_0 \\
 & A_3 & A_2 & A_1 & A_0 \\
 + & B_3 & B_2 & B_1 & B_0 \\
 \hline
 & C_4 & S_3 & S_2 & S_1 & S_0
 \end{array}$$



n bit Parallel adder

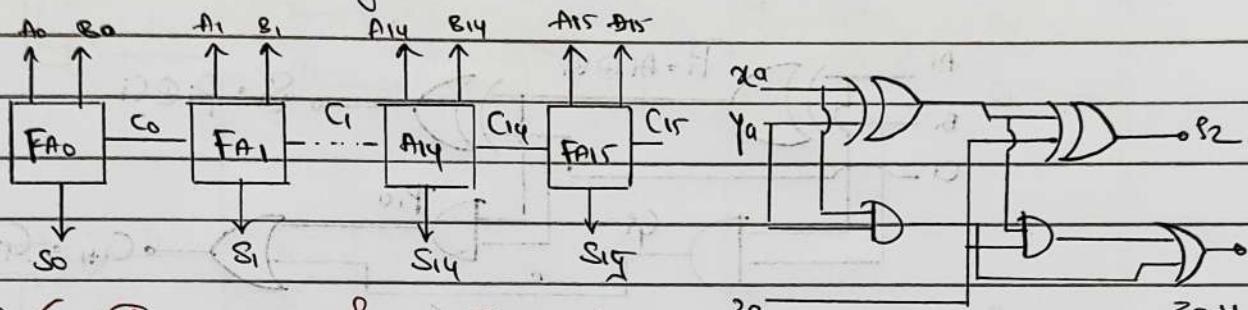
1. $(n-1)$ FA + 1 t_{ff}

2. n -FA

3. (2^{n-1}) t_{ff} + $(n-1)$ OR.

$$T = (n-1) T_{carry} + \max(T_{sum}, T_{carry})$$

Q1. A 16-bit ripple carry adder is realised using 16 identical full adders (FA) as shown. The carry-propagation delay of each FA is 12 ns and the sum-propagation delay of each FA is 15 ns. The worst case delay (in ns) of the 16-bit adder will be _____



$$T = (n-1) T_{carry} + \max\{T_{sum}, T_{carry}\}$$

$$= (16-1) \times 12 \text{ ns} + \max\{15 \text{ ns}, 12 \text{ ns}\}$$

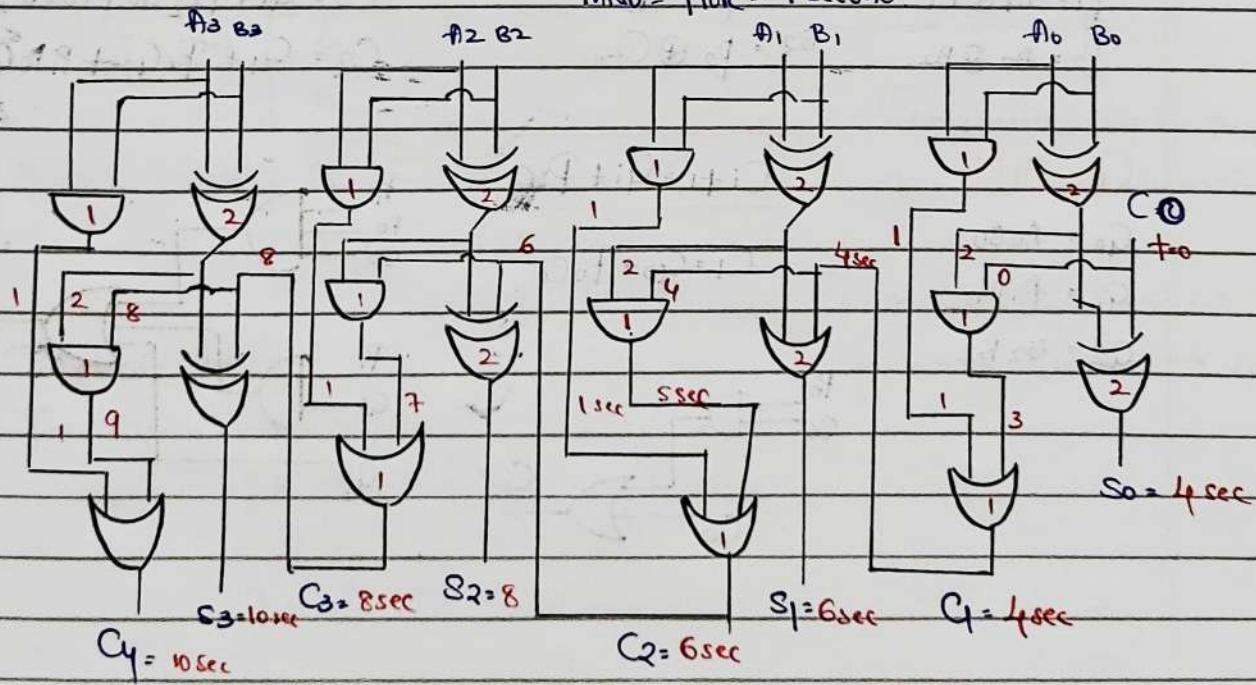
$$= 15 \times 16 + 15$$

$$= \underline{\underline{195 \text{ ns}}}$$

Parallel Adder [4 bit]

$$T_{FF} = 2 \text{ second}$$

$$T_{AND} = T_{IOR} = 1 \text{ second}$$



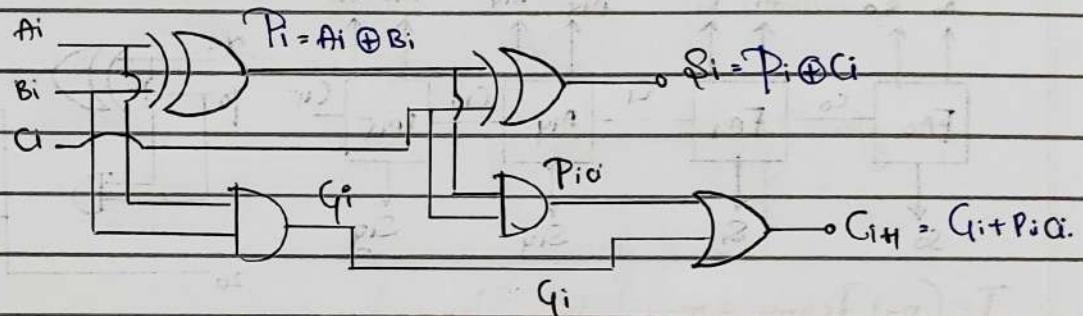
$$T = (n-1) \{ T_{\text{AND}} + T_{\text{OR}} \} + \max \{ T_{\text{sum}}, T_{\text{carry}} \}$$

$$\begin{aligned} T &= (4-1) \{ 1+1 \} + \max \{ 4, 4 \} \\ &= 3 \times 2 + 4 \\ &= \underline{10 \text{ ns}}. \end{aligned}$$

Parallel Adder, LACA

[Look-Ahead Carry Adder]

→ faster adder among all the adders



$P_i \rightarrow$ Carry propagating term

$G_i \rightarrow$ Carry generating term

$$P_i = A_i \oplus B_i$$

$$S_i = P_i \oplus C_i$$

$$P_0 = A_0 \oplus B_0$$

$$S_0 = P_0 \oplus C_0$$

$$C_2 = G_1 + P_1 C_1$$

$$P_1 = A_1 \oplus B_1$$

$$S_1 = P_1 \oplus C_1$$

$$C_0 = G_1 + P_1 [G_0 + P_0 C_0]$$

$$P_2 = A_2 \oplus B_2$$

$$S_2 = P_2 \oplus C_2$$

$$C_1 = G_1 + P_1 G_0 + P_1 P_0 C_0$$

$$G_i = A_i B_i$$

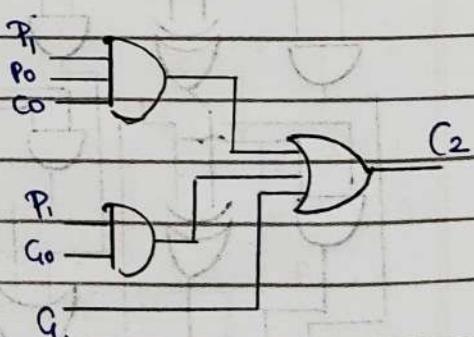
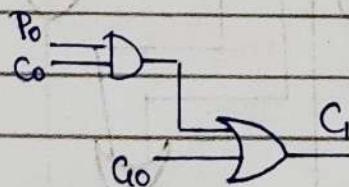
$$C_i + 1 = G_i + P_i C_i$$

$$G_0 = A_0 B_0$$

$$G_1 = A_1 B_1$$

$$G_2 = A_2 B_2$$

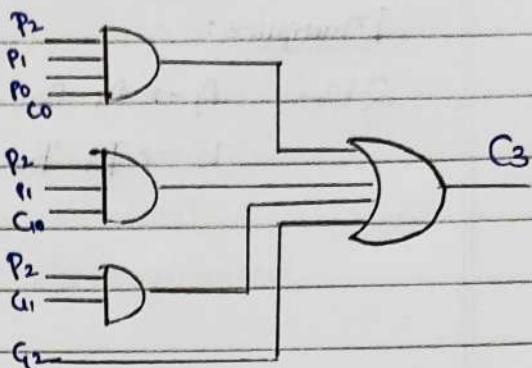
$$C_1 = G_0 + P_0 C_0$$



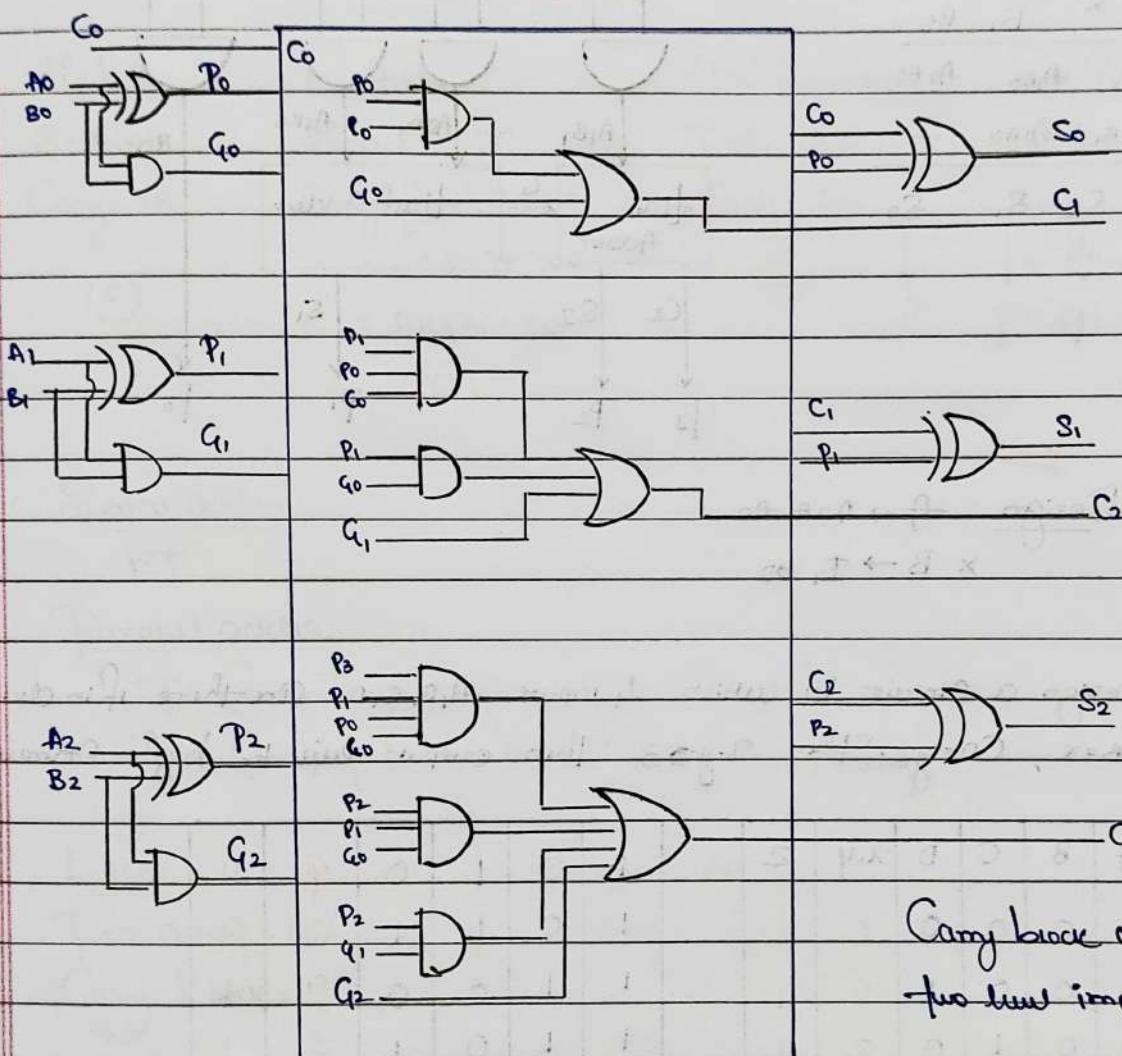
$$C_3 = G_2 + P_2 C_2$$

$$C_8 = G_2 + P_2 [G_1 + P_1 G_0 + P_1 P_0 G_0]$$

$$C_3 = G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 G_0.$$



CARRY BLOCK



Carry block can be always two level implementation.

* Delay for Carry Block $\Rightarrow 2T$

* For entire circuit $\Rightarrow 4T$

n bit LACA

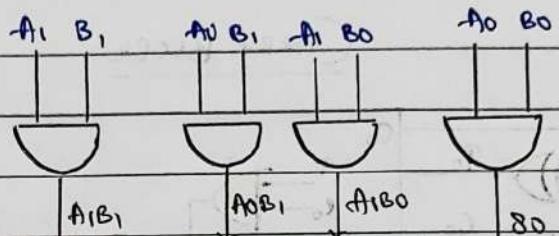
For Carry block:

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

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

Multiplication:2 bit. $A \rightarrow A_1, A_0$ $B \rightarrow B_1, B_0$ 52 52 $\underline{2704}$ P_3, P_2, P_1, P_0

$$\begin{array}{r} A_1 \ A_0 \\ \times \ B_1 \ B_0 \\ \hline C \ A_1 B_0 \ A_0 B_0 \end{array}$$

 $A_1 B_1$ 

$$\begin{matrix} C_2 & S_2 & S_1 & S_0 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ P_3 & P_2 & P_1 & P_0 \end{matrix}$$

~~HW~~

Half Adder

Half Adder

80

Design. $A \rightarrow A_2 A_1 A_0$ $B \rightarrow B_1 B_0$

Q1. Design a circuit in which 4 inputs A, B, C, D are there if in decimal $AB=2$, $CD=4$. Then $x.y \leq 3$. Thus output will be high otherwise low.

A	B	C	D	x.y	z	1	0	1	0	4	0
0	0	0	0	0	1	1	0	1	1	6	0
0	0	0	1	0	1	1	1	0	0	0	1
0	0	1	0	0	1	1	1	0	1	3	1
0	0	1	1	0	1	1	1	1	0	6	0
0	1	0	0	0	1	1	1	1	1	9	0
0	1	0	1	1	1	$AB \neq 2$					
0	1	1	0	2	1	00	01	11	10	1	1
0	1	1	1	3	1	00	01	11	10	1	1
1	0	0	0	0	1	11	11	11	11	1	1
1	0	0	1	2	1	10	11	11	11	1	1

$\begin{matrix} 00 & 01 & 11 & 10 \end{matrix}$

$\begin{matrix} 00 & 01 & 11 & 10 \end{matrix}$

$\begin{matrix} 01 & 11 & 11 & 11 \end{matrix}$

$\begin{matrix} 11 & 11 & 11 & 11 \end{matrix}$

$\begin{matrix} 10 & 11 & 11 & 11 \end{matrix}$

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

Combinational Circuit - { Static Circuits }

Quick Summary:

Comparator

$$(A > B) \Leftrightarrow A_1\bar{B}_1 + (A_1 + \bar{B}_1)A_0\bar{B}_0 \rightarrow \text{minimised}$$

$$A_1\bar{B}_1 + (A_1 \oplus B_1)A_0\bar{B}_0 \rightarrow \text{Semi-minimised}$$

MUX, DEMUX

Encoder, Decoder.

Half Adder

$$S = A \oplus B$$

$$\text{Carry} = AB$$

5.

Full Adder

$$\text{Sum} = A \oplus B \oplus C$$

$$\text{Carry} = \sum_m (3, 5, 6, 7)$$

$$= (A \oplus B)C + AB$$

$$= AB + AC + BC$$

9.

Half Subtractor

$$\text{Diff} = A \oplus B$$

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

5.

Full Subtractor

$$\text{Diff} = A \oplus B \oplus C$$

$$\text{Carry} = \sum_m (1, 2, 3, 7)$$

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

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

9.

Serial adder

D.T.

Parallel adder

$$T = (n-1) T_{\text{carry}} + m_{\text{or}} \{ T_{\text{sum}} + T_{\text{carry}} \}$$

$$T = (n-1) \{ T_{\text{and}} + T_{\text{or}} \} + m_{\text{or}} \{ T_{\text{sum}}, T_{\text{carry}} \}.$$

LACA

$$\text{Total delay} = 4T$$

$$\text{Carry block} = 2T$$

Carry Block

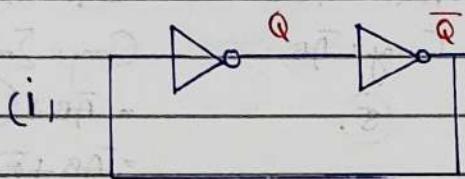
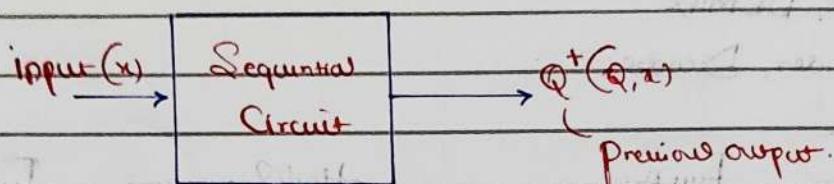
$$ANR = \frac{n(n+1)}{2}$$

$$OR = n$$

SEQUENTIAL CIRCUIT

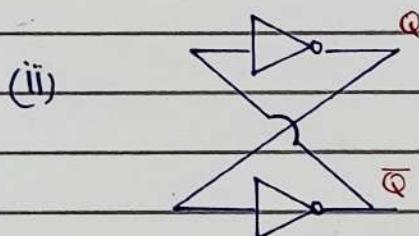
Sequential Circuit:

- * A circuit with feedback and memory are called Sequential Circuit.
- * Output of the Sequential Circuit depends on previous output as well as present state of input.



Basic Memory Element

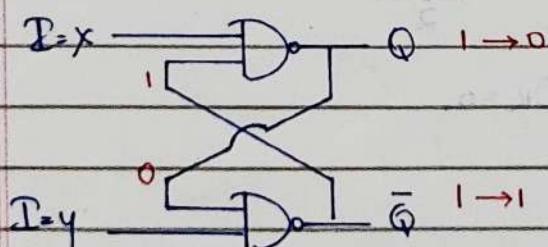
$$i = ii$$



Latchee : Basic memory element

* Latchee are level triggered

* Latchee have two output which is complement of each other



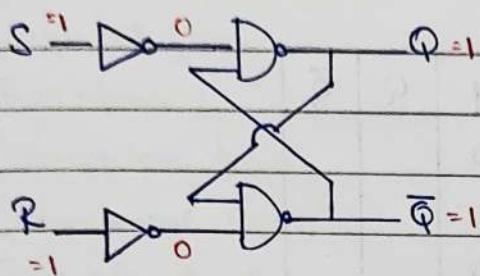
X	Y	Q	\bar{Q}	
0	0	1	1	→ Jurdid
0	1	1	0	
1	0	0	1	
1	1	Q	\bar{Q}	→ Hold

DANDO

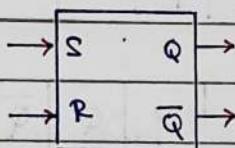
A	B	$y=A \cdot B$
0	0	1
0	1	1
1	0	1
1	1	0

Note: Whenever $x=y=1$ is applied and invalid condition occurs than a NANO having lower propagation delay first change its output and others remain on its previous state are called racing problem or racing problem.

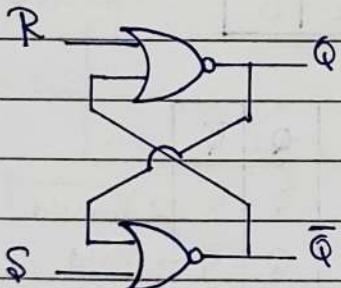
SR : Set-Reset Latch



S	R	Q	\bar{Q}	
0	0	0	1	Hold
0	1	0	1	Reset
1	0	1	0	Set
1	1	X	X	(Invalid)

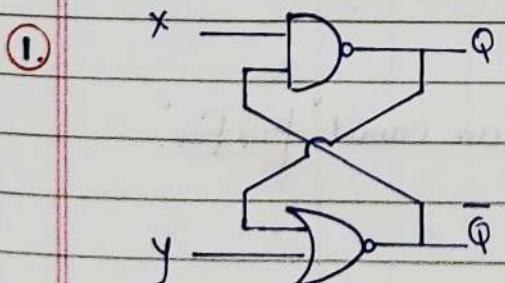


S-R Latches by using NOR Gates:



S	R	Q	\bar{Q}	NOR	A	B	$y = \overline{A+B}$
0	0	0	1	Hold	0	0	1
0	1	0	1	Reset	0	1	0
1	0	1	0	Set	1	0	0
1	1	X	X	Invalid.	1	1	0

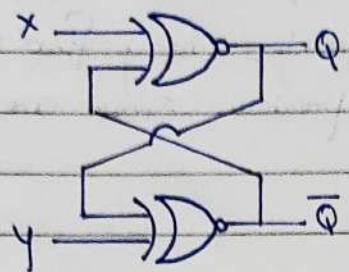
HW



X	Y	Q	\bar{Q}
0	0		
0	1		
1	0		
1	1		

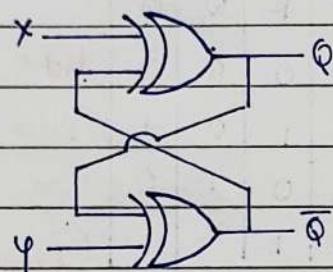
HW

②



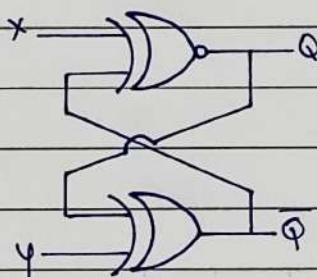
X	Y	Q	\bar{Q}
0	0		
0	1		
1	0		
1	1		

③.



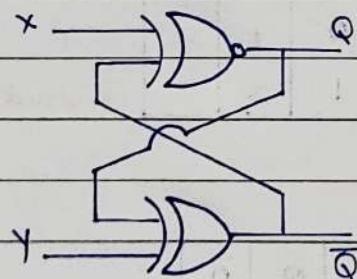
X	Y	Q	\bar{Q}
0	0		
0	1		
1	0		
1	1		

④.



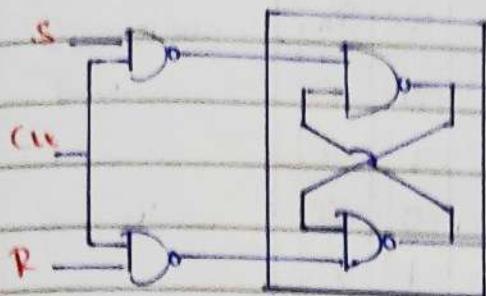
X	Y	Q	\bar{Q}
0	0	1	0
0	1	1	0
1	0	Q	\bar{Q}
1	1	1	0

⑤

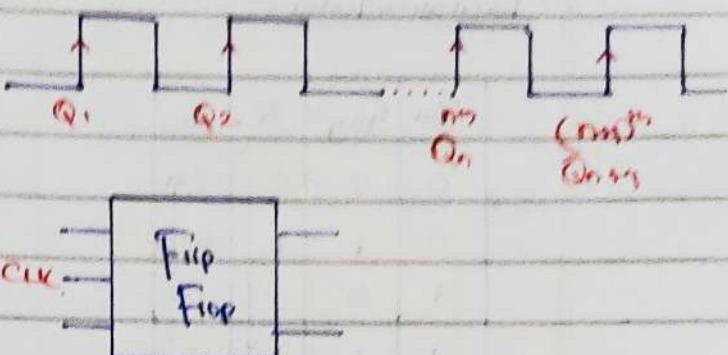


X	Y	Q	\bar{Q}
0	0	Q	\bar{Q} → Forward
0	1	Q	\bar{Q} → Hold.
1	0	\bar{Q}	Q → Toggle
1	1	\bar{Q}	\bar{Q} → Invert

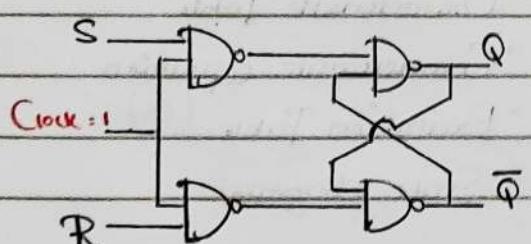
→ Latches with Control Phenomena are called Flip-flop.



Clock:

Flip Flop Setups

1. Circuit Diagram
2. Truth Table
3. Characteristic Table
4. Characteristic Equation
5. Excitations Table
6. State Diagram

SR Flip Flop (Ser-Reset Flip flop)Circuit Diagram:Truth Table:

S	R	Q_n	\bar{Q}_n	
0	0	Q_n	\bar{Q}_n	Hold/Pres.
0	1	0	1	→ Reset.
1	0	1	0	→ Set.
1	1	x	x	→ Transition

Characteristics

Table:

	S	R	Q_n	Q_{n+1}
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	0
4	1	0	0	1
5	1	0	1	1
6	1	1	0	x
7	1	1	1	x

$$Q_{n+1} = (S, R, Q_n) = \sum m(1, 4, 5) + \sum d(6, 7)$$

Characteristic Equations:

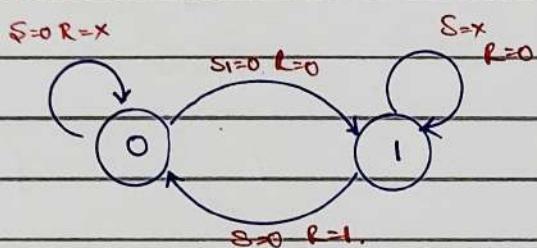
S	\bar{Q}_n	$\bar{R}Q_n$	$\bar{R}Q_n$	RQ_n	$R\bar{Q}_n$
00	1	1	0	0	0
01	1	0	1	0	0
11	0	1	0	1	0
10	0	0	0	0	1

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

Excitation Table:

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

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

State Diagram:

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

HW

Q Truth Table : C J Q_{n+1}

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

Write :

* Characteristic Table

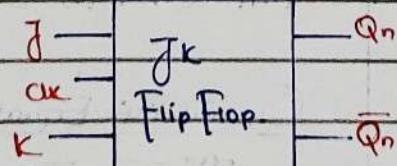
* Characteristic Equations

* Excitation Table

* State Diagram.

JK Flip Flop

Symbol:



Truth Table:

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

CharacteristicTable:

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

$$Q_{n+1}(J, K, Q_n) = \sum (1, 4, 5, 6)$$

Characteristic Equation:

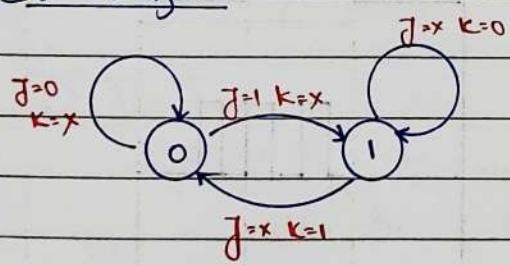
JQ_n	$\bar{K}Q_n$	$\bar{K}Q_n$	KQ_n	KQ_n
00	01	11	10	
$\bar{J}0$		1		
$\bar{J}1$	1	1		1

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

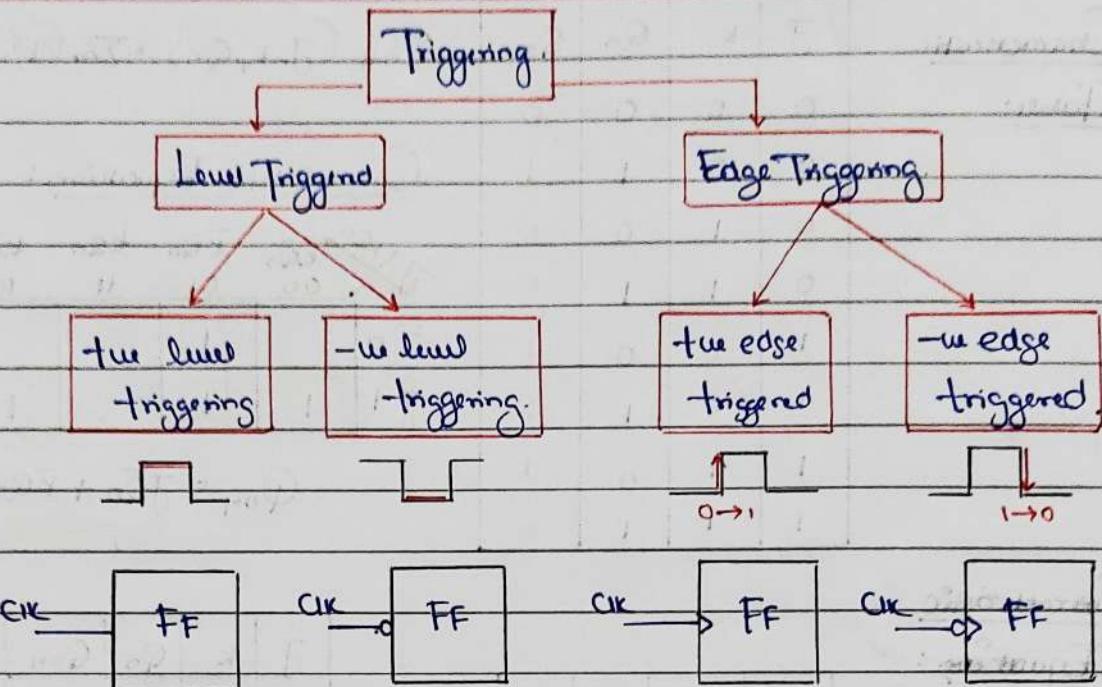
ExcitationTable:

Q_n	Q_{n+1}	J	K
0	0	0	$x\uparrow$
0	1	1	x
1	0	x	1
1	1	$x\downarrow$	0

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

State Diagram:

Q_n	Q_{n+1}	J	K
0	0	0	$x\uparrow$
0	1	1	x
1	0	x	1
1	1	x	0

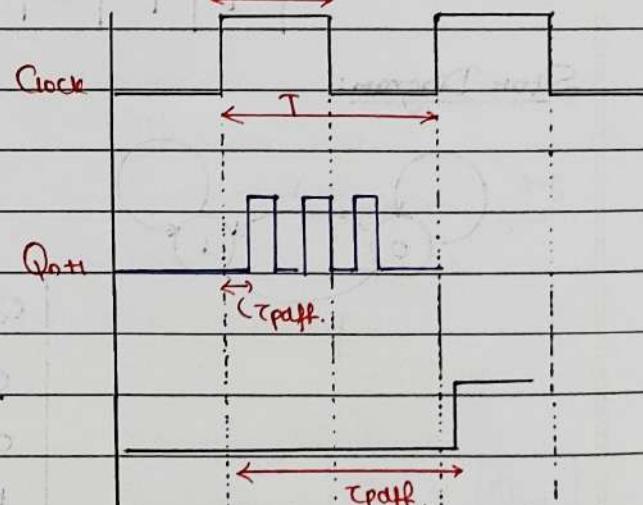
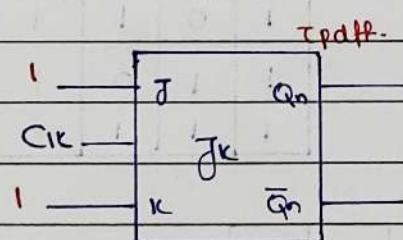


Note: Level triggered JK flip-flop suffers from the problem of "Race Around".

(T_{pw})

Pulse width

Race-around Problem:

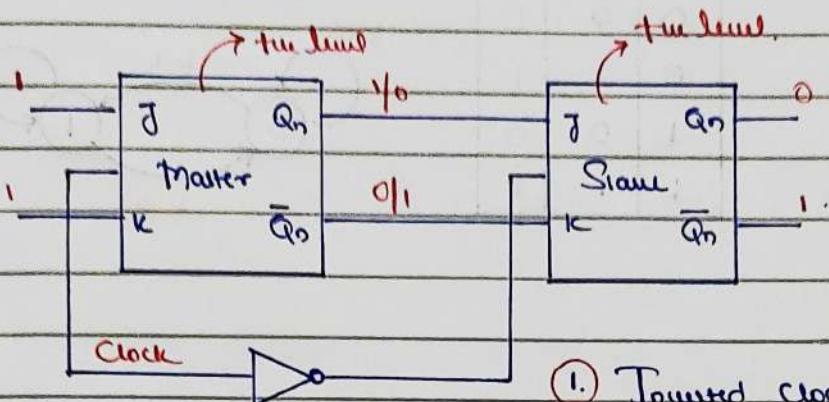


To avoid the Race Around Problem,

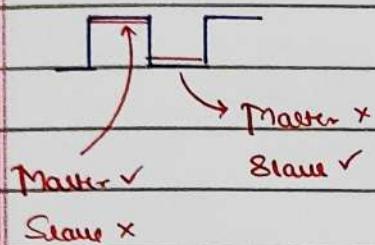
1. $T_{pw} < T_{pdff} < T_{clock}$
2. By Master Slave ff.

→ When $J=K=1$ is applied to the level sensitive of JK FF, then within the duration of pulse output of the FF toggles more than one times are called "Race Around Problem".

Master Slave FF.



① Tristate clock is applied to the slave as compared to master.



② Master-Slave flip flop is used to store single bit because output is taken only from slave.

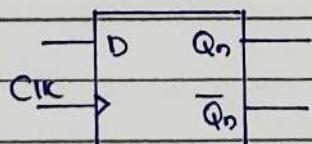
③ By the operation it seems that master is true triggered whereas slave is negative edge triggered.

D Flip Flop.

* It is known as Delay FF or Transparent FF

* In the D FF circuit the input is applied it will directly come to the output along with the clock.

Symbol:



Characteristic Table:

D	Qn	Qn+1
0	0	0
0	1	0
1	0	1
1	1	1

Truth

Table:

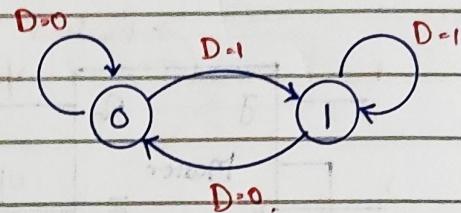
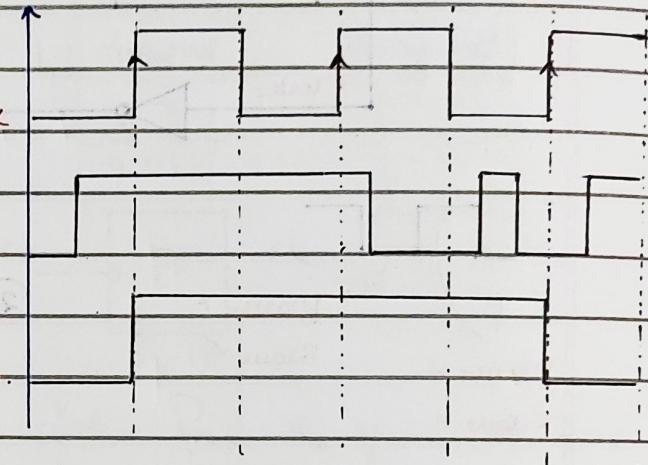
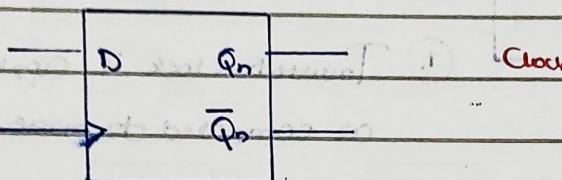
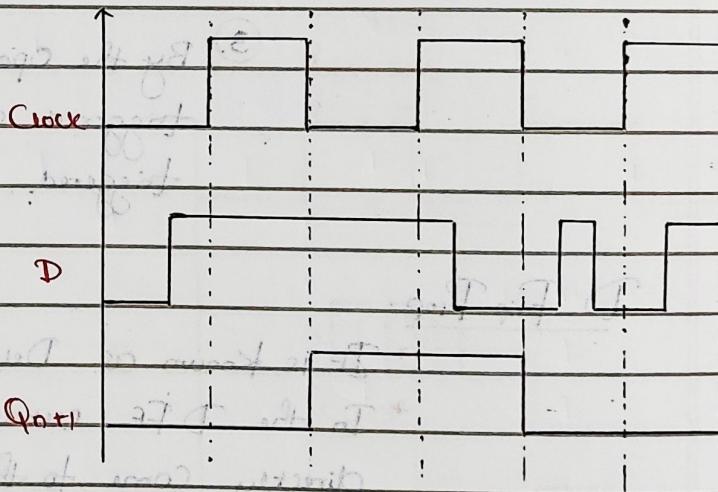
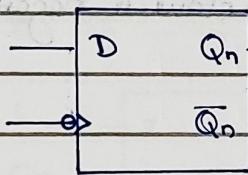
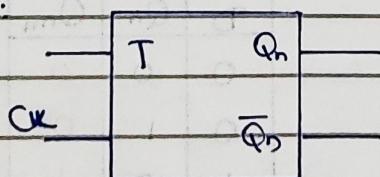
D	Qn+1
0	0
1	1

Characteristic Equation:

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

Excitation Table:

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

State Diagram: Q_1 : Q_2 :T Flip Flop (Toggle Flip Flop)Symbol:TruthTable:

Characteristic Table:

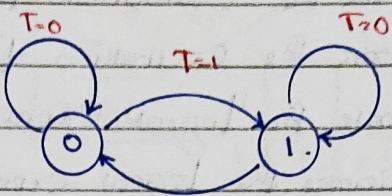
T	Q_n	Q_{n+1}
0	0	0
0	1	1
1	0	0
1	1	1

Characteristic Equations: $Q_{n+1} = \bar{T}Q_n + T\bar{Q}_n$

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

Excitation Table:

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

State Diagram:

Note: J-K FF is also known as universal flip flop.

	J	K	Q_{n+1}								
SE FF	0	0	Q_n								
	0	1	0								
	1	0	1								
	1	1	\bar{Q}_n								

SR FFJK FFD FFT FF

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

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

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

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

S	R	Q_{n+1}	J	K	Q_{n+1}	D	Q_{n+1}	T	Q_{n+1}
0	0	Q_n	0	0	Q_n	0	0	0	Q_n
0	1	0	0	1	0	1	1	1	\bar{Q}_n
1	0	1	1	0	1				
1	1	x	1	1	\bar{Q}_n				

Excitation Table

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

Designing of Flip Flop:

- Step 1 : Write the characteristic table of desired FF
 - Step 2 : Write the excitation table of available FF
 - Step 3 : Write the logical expression
 - Step 4 : Minimize the logical expression
 - Step 5 : Hardware implementation.

Q3. Design JK flip flop using SR flip flop.
'desired' 'available.'

<u>Step 1</u>	J	K	Q _n	Q _{n+1}	S	R
<u>Step 2</u>	0	0	0	0	0	x
	0	0	1	1	x	0
	0	1	0	0	0	x
	0	1	1	0	0	1
	1	0	0	1	1	0
	1	0	1	1	x	0
	1	1	0	1	1	0
	1	1	1	0	0	1

$$\underline{\text{Step 3:}} \quad S(J, k, Q_n) = \sum_m (J_{m,6}) + \sum_d (J_{m,5})$$

$$R(J, K, Q_0) = \sum_m (3, 7) + \sum_d (6, 2).$$

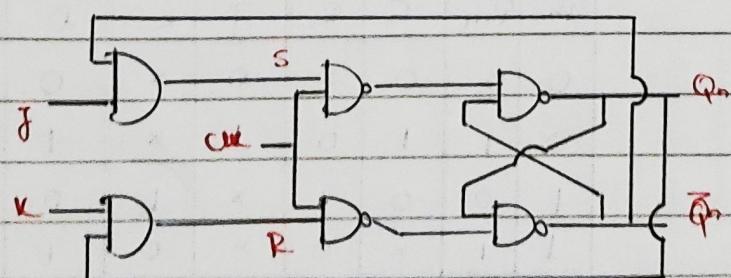
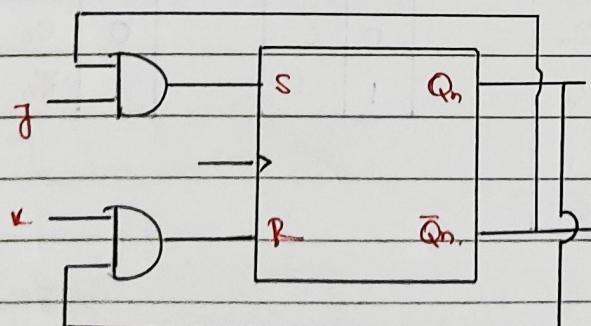
Step 4: ~~JKQC~~

$S \Rightarrow$	0	x		$S = \bar{J}Q_n$
	1	x	1	

~~J~~ ~~Q2~~

$$R \Rightarrow 0 \quad x \quad | \quad x \quad R = kq_0$$

Steps:



Q4. Design a D-FF by using SR FF.

Step 1:

Step 2:

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

Step 3:

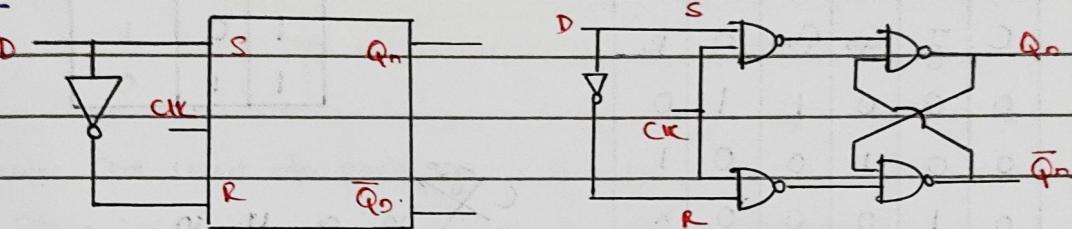
& Step 4:

D	Q _n	Q _{n+1}	D	Q _n	Q _{n+1}
0	0	1	0	0	1
1	1	x	0	x	1
1	1	x	1	1	0

S = D

R = \bar{D} .

Steps:



Q5. Design a T-FF using SR FF.

Step 1:

Step 2:

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

Step 3 &

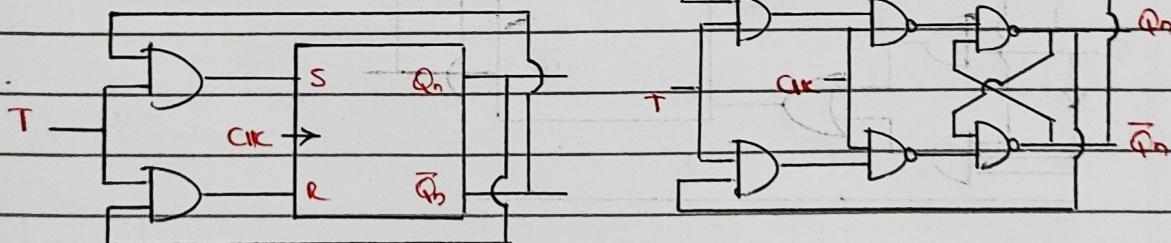
Step 4:

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

S = T \bar{Q}_n

R = T Q_n .

Steps:



Q6. Design T-FF using D-FF.

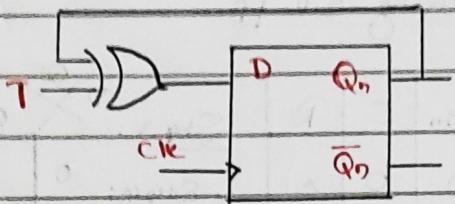
Step 1:

Step 2:

T	Q _n	Q _{n+1}	D
0	0	0	0
0	1	1	1
1	0	1	1
1	1	0	0

Step 3 & Step 4:

D = T $\oplus Q_n$.

Steps:

Q7. Design a Cg FF by using SR FF.

Truth Table:

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

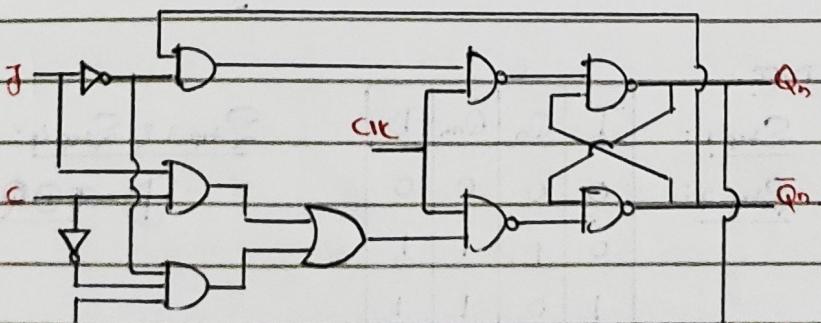
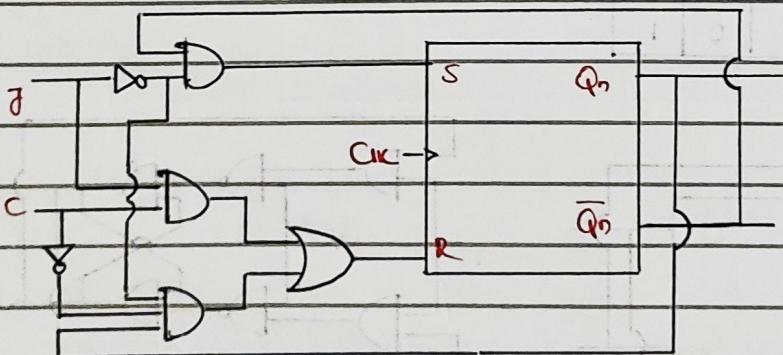
C	J	Q_n	Q_{n+1}	S	R
0	0	0	1	1	0
0	0	1	0	0	1
0	1	0	0	0	x
0	1	1	1	x	0
1	0	0	1	1	0
1	0	1	1	x	0
1	1	0	0	0	x
1	1	1	0	0	1

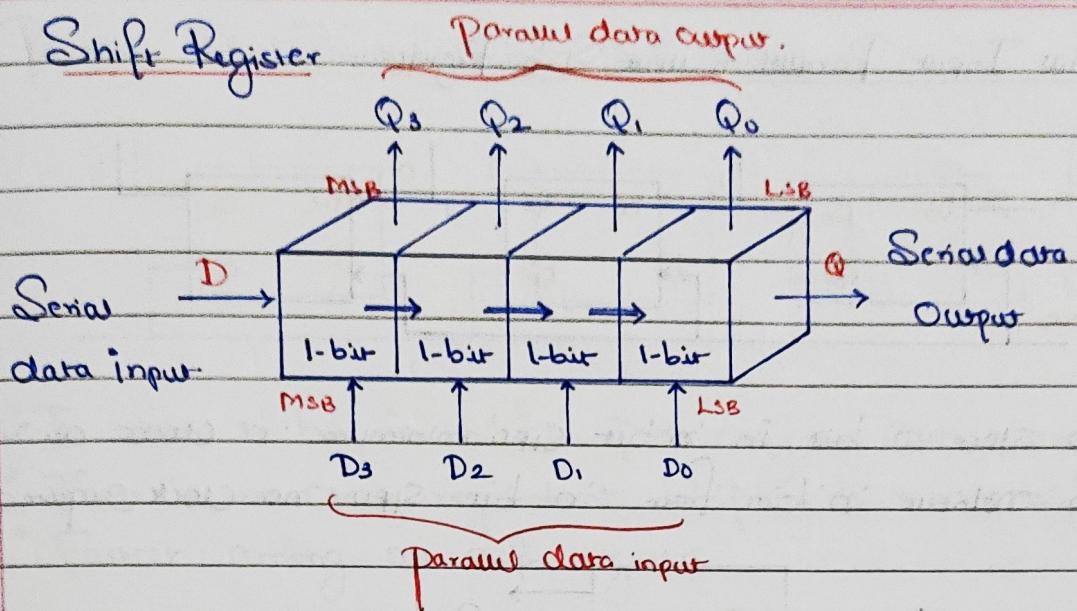
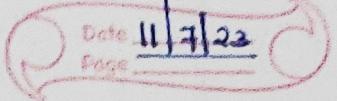
C	$\bar{J}Q_n$	00	01	11	10
0	1	1	x		

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

C	$\bar{J}Q_n$	00	01	11	10
1	0	1	x		

$$R = \bar{C} \bar{J} Q_n + C J$$

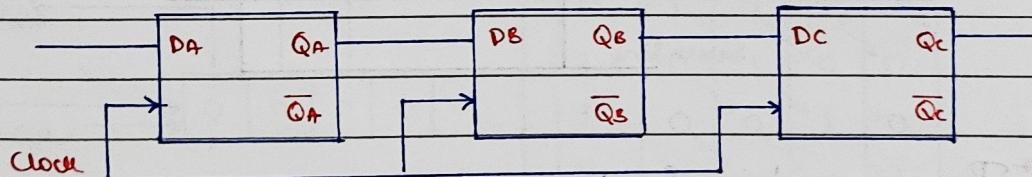




- * Registers are used to store group of bits
- * To store 'n' bits minimum 'n' flip flops are required.
- * Generally D flip flops are used to Design registers.

1. Serial input Serial output Shift register (SISO)
2. Serial input Parallel output Shift register (SIPo)
3. Parallel input Serial output Shift register (PISO)
4. Parallel input Parallel Output Shift register.. (PIPO)

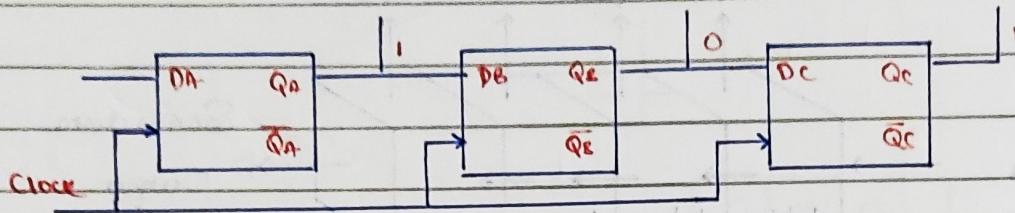
Serial Input Serial Output Shift Register



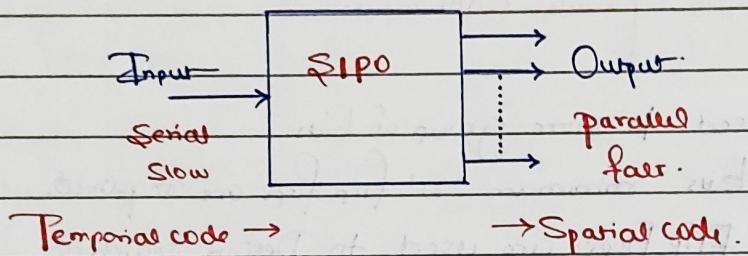
Clock	DA	QA	QB	QC
0	101	0	0	0
1		1	0	0
2		0	1	0
3		1	0	1

- * To store 'n' bit in SISO minimum 'n' clock are required
- * To retrieve 'n' bits from 'n' bit SISO minimum 'n' clock are required.
- * It is the slowest register among all the Shift register.

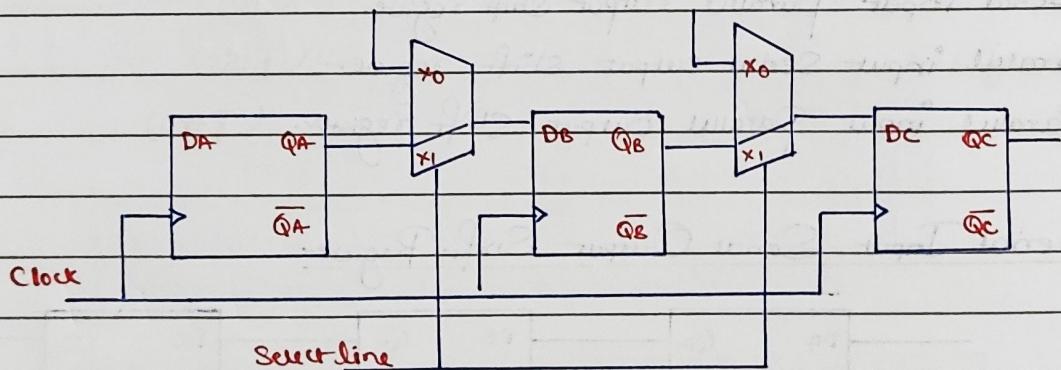
Serial Input Parallel Output Shift Register



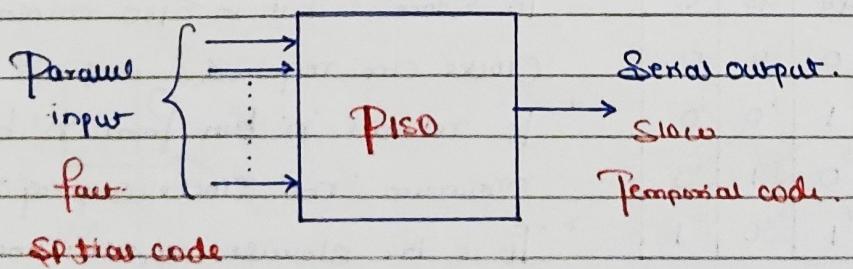
- * To store 'n' bit in 'n' bit SIPR minimum 'n' Clocks are required.
- * To retrieve 'n' bits from 'n' bit SIPR no clock required.



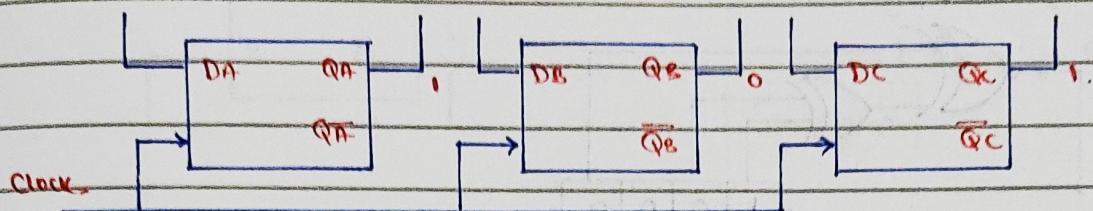
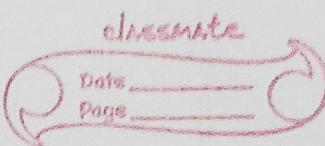
Parallel Input Serial Output Shift Register



- * To store 'n' bits in PISO only one clock is required.
- * To retrieve 'n' bits from 'n' bit PISO minimum (n-1) clock are required.



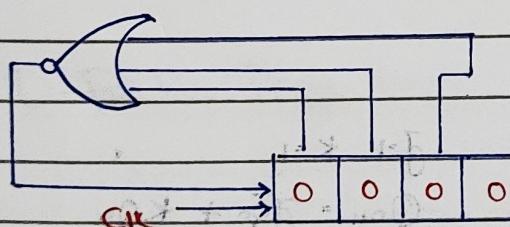
Parallel Input Parallel Output Shift Register.



- * To store 'n' bit in 'n' bit Pipo only one clock is required.
- * There is no clock requirement in Pipo to retrieve 'n' bits.
- * Fastest among all shift registers.

	Store	Retrieve	Total	
SISO	n	n-1	$2n-1$	→ Slower
SIPo	n	0	n	
PISO	1	n-1	n	
Pipo	1	0	1	→ Faster.

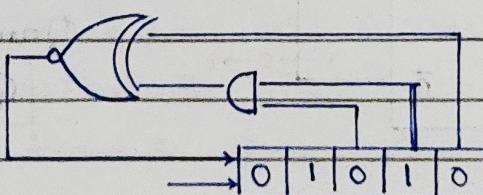
Q1. Write the state of the register given below? Assume all flip-flops are reset initially



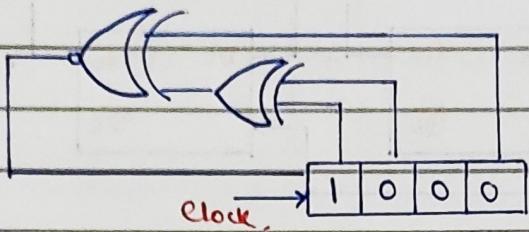
Clock	Q ₃	Q ₂	Q ₁	Q ₀
0	0	0	0	0
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1
5	1	0	0	0
6	0	1	0	0
7	0	0	1	0
8	0	0	0	1

Ans

Q2. Output after 137th clock will be?



Q3. After 9th clock the output will be?



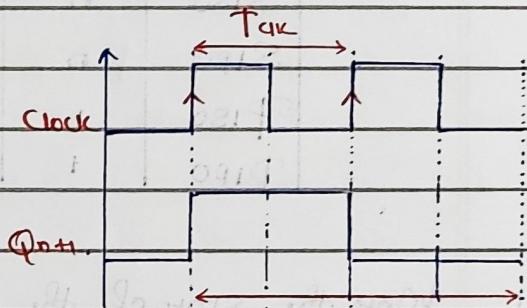
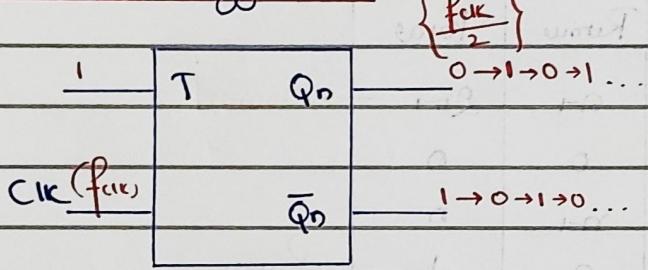
Toggle Mode of flip flop.

1. T Flip Flop toggle mode.

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

$$Q_{n+1} = 1 \oplus Q_n - \bar{Q}_n \Rightarrow Q_{n+1} = \bar{Q}_n$$

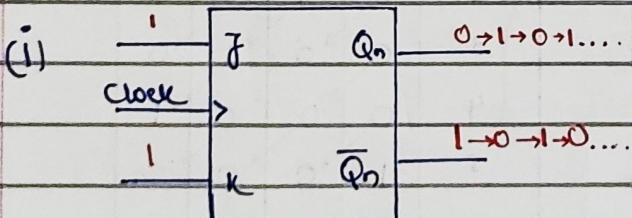
(Toggle Mode)



$$f_{clk} = \frac{1}{T_{clk}}$$

$$f_{Q_{n+1}} = \frac{1}{2T_{clk}} = \frac{f_{clk}}{2}$$

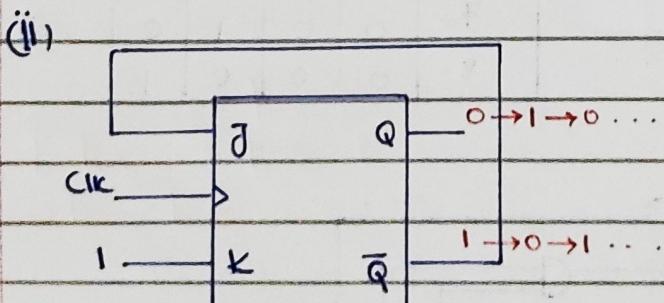
2. Jk Flip Flop toggle mode



$$J=1 \quad K=1$$

$$Q_{n+1} = J\bar{Q}_n + \bar{K}Q_n \\ = 1\bar{Q}_n + 0.Q_n$$

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



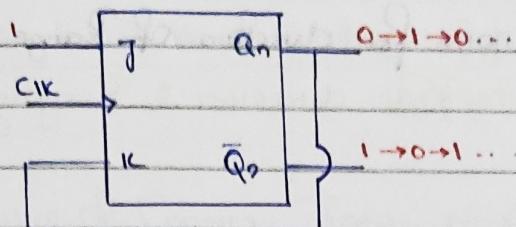
$$J = \bar{Q}_n \quad K = 1$$

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

$$Q_{n+1} = \bar{Q}_n \cdot \bar{Q}_n + 0.Q_n$$

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

(iii)



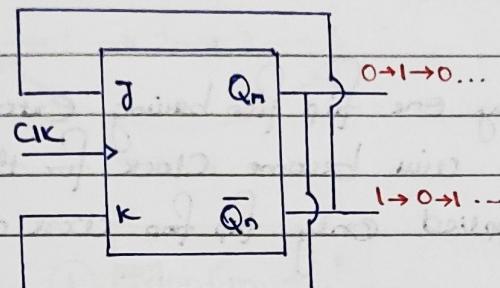
$$J=1 \quad K=Q_n$$

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

$$= 1 \cdot \bar{Q}_n + \bar{Q}_n \cdot Q_n$$

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

(iv)



$$J \cdot \bar{Q}_n \quad K=Q_n$$

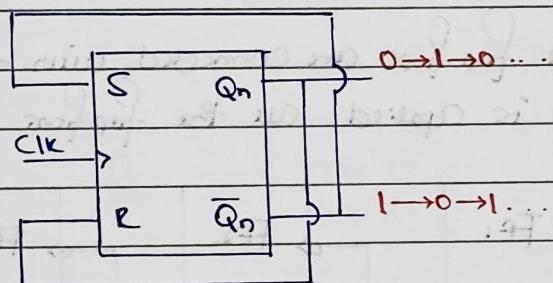
$$Q_{n+1} = J\bar{Q}_n + \bar{K}Q_n$$

$$= \bar{Q}_n \cdot \bar{Q}_n + \bar{Q}_n \cdot Q_n$$

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

3. Toggle Mode of SR Flip-Flop.

$$S=\bar{Q}_n \quad R=Q_n$$



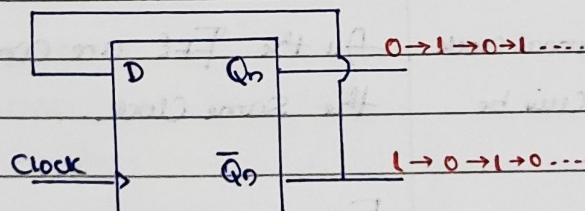
$$Q_{n+1} = S + \bar{R}Q_n$$

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

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

4. D-Flip flop Toggle Mode.

$$D=\bar{Q}_n$$



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

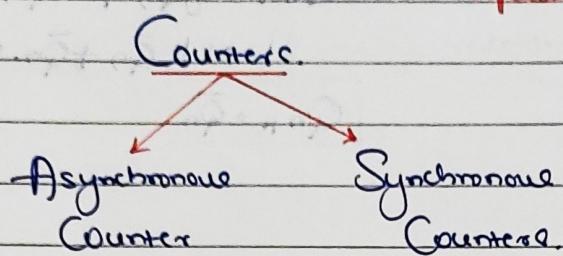
$$Q_{n+1} = \bar{Q}_n$$

Counters:

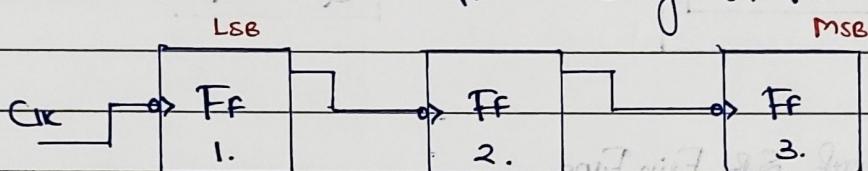
- * Counters are used to count number of clock
- * Counter are used in frequency divider circuit.
- * Counter are also used in ADC (Analog to Digital converter)
- * Counter are also known as Pulse stretcher circuit.

* Counters also used in RADAR for detection of Range

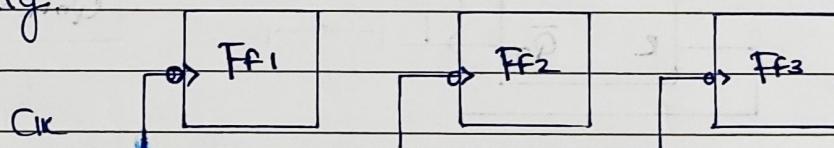
↳ Radio detection & Ranging



Asynchronous Counter: Only one flip flop having external clock and output of that flip flop will become clock for the next flip flop. So when clock applied only flip flop work at that time.



Synchronous Counter: All flip flops are connected with the same clock. Hence when the clock is applied all the flip flops work simultaneously.



Asynchronous Counter

1. Only one FF having external clock and output of that FF will be clock for next FF.
2. Slower.
3. Only increasing and decreasing Counting Possible.
4. There is transition error.
5. Eg: Ripple counter

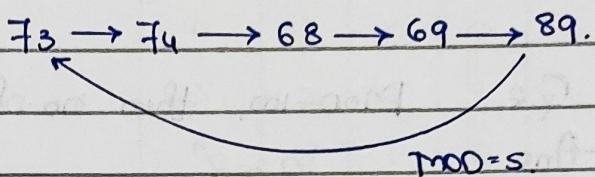
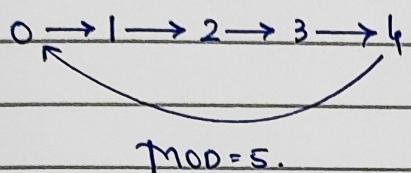
Synchronous Counter

1. All the FFs are connected with the same clock.
2. Faster
3. All type of countings are possible.
4. No transition error
5. Eg: Ring counter, Johnson Counter.

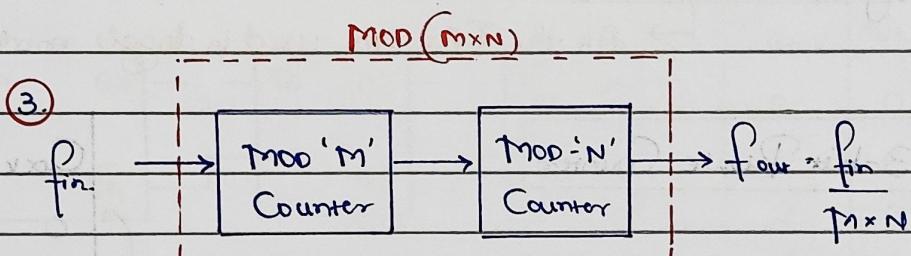
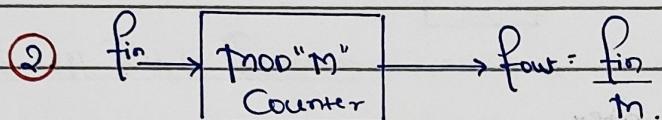
Maximum no. of States = 2^n .

$\hookrightarrow n$: no. of FFs.

Module of Counter: Total no. of State used by the Counter One
Can be mod of the counter.



Note: ① $Mod(m) \leq 2^n$.



BCD Counter:

0 → 0000

1 → 0001

2 → 0010

3 → 0011

4 → 0100

5 → 0101

6 → 0110

7 → 0111

8 → 1000

9 → 1001

$$m \leq 2^n$$

$$n \geq \log_2 m$$

$$n \geq \log_2 10$$

$$n \geq 3\dots$$

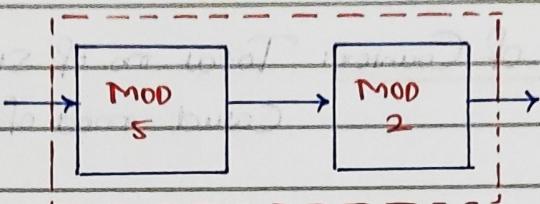
$$n=4$$

Q1. If mode-5 Counter is cascaded with mod-2 Counter, then it will become?

Ans.

$Mod-10$ counter.

$$Mod(5 \times 2) = Mod-10$$



Q2. Mod-100, then no. of FF?

Ans

$$M \leq 2^n$$

$$n \geq \log_2 M$$

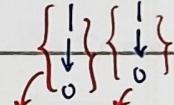
$$\geq \log_2 100$$

$$\geq 6 \dots$$

$$n \geq 7$$

Asynchronous Counter

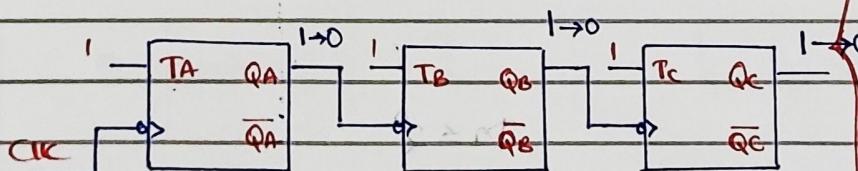
→ All the FFs are used in toggle mode.



3-bit Ripple Counter

Clock	Q _C	Q _B	Q _A
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1
8	0	0	0
9	0	0	1
10	0	1	0

Mod-8 Up Ripple Counter:

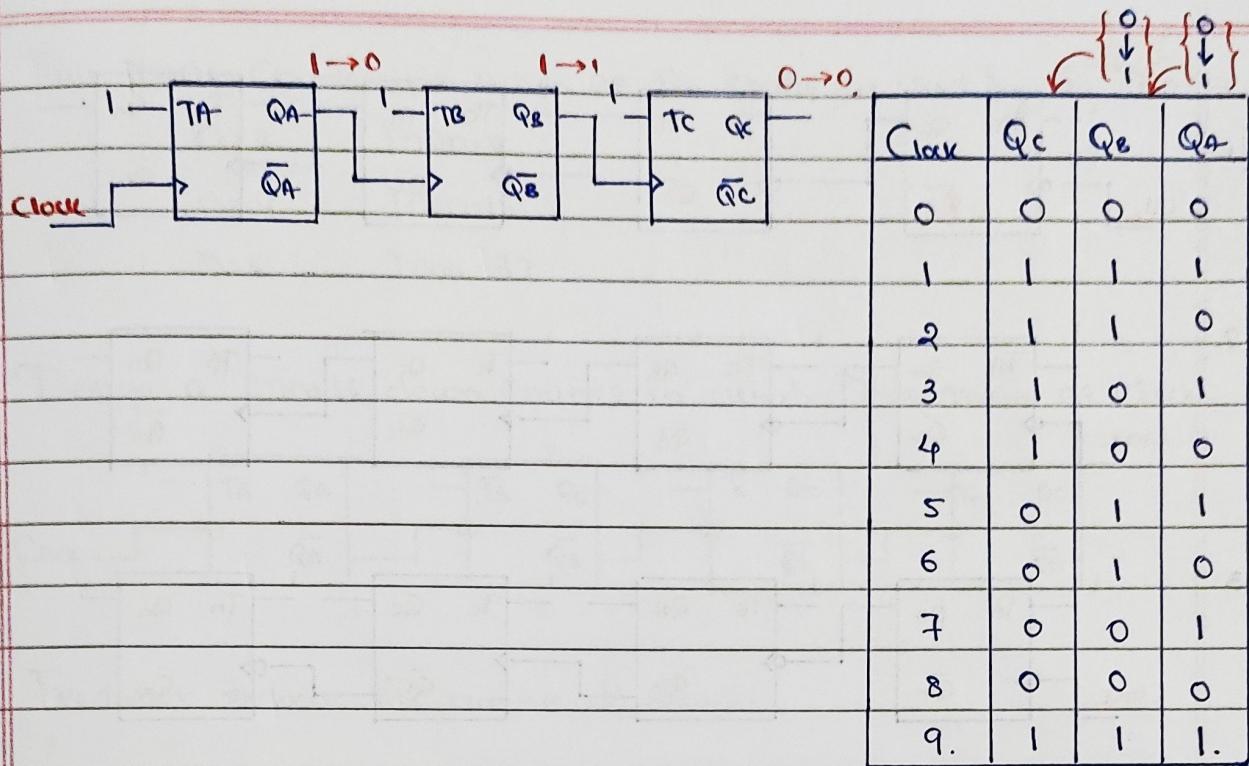


→ Q_A will toggle when one edge of the External Clock appears.

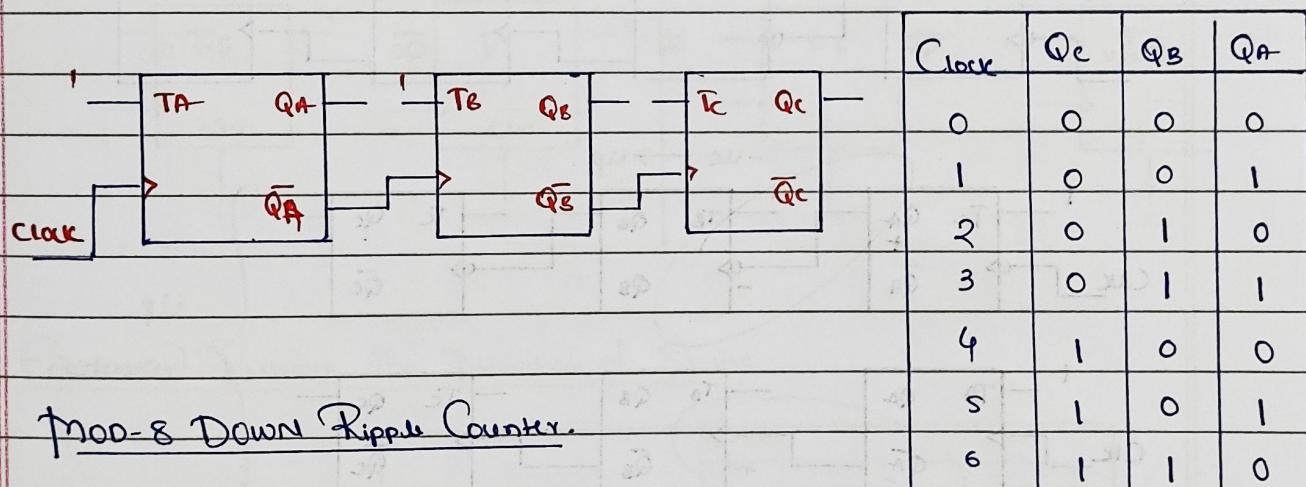
→ Q_B will toggle when Q_A goes from 1 to 0

Mod-8 Down Ripple Counter:



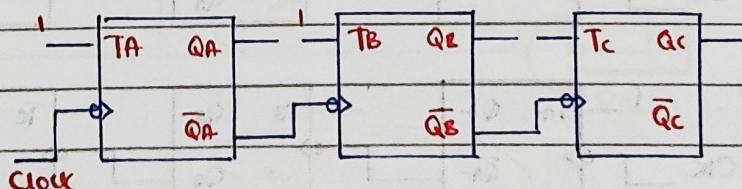


T_{mod-8} UP Ripple Counter.



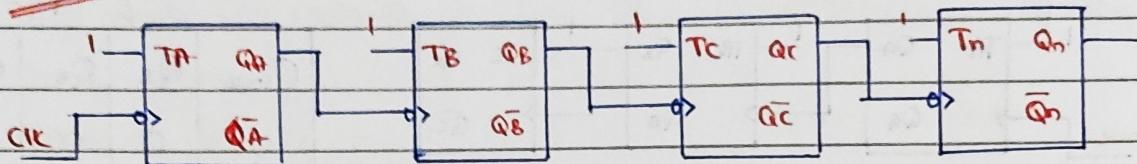
T_{mod-8} DOWN Ripple Counter.

Clock	Q_C	Q_B	Q_A
0	0	0	0
1	1	1	1
2	1	1	0
3	1	0	1
4	1	0	0
5	0	1	1
6	0	1	0
7	0	0	1
8	0	0	0

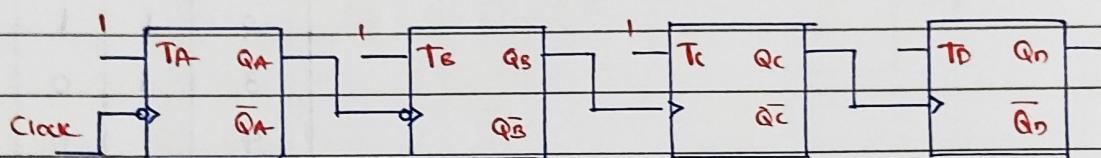


HW

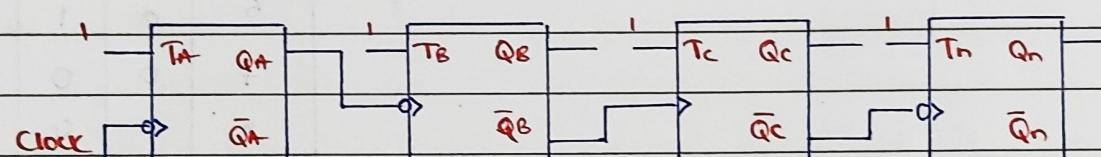
Q1.



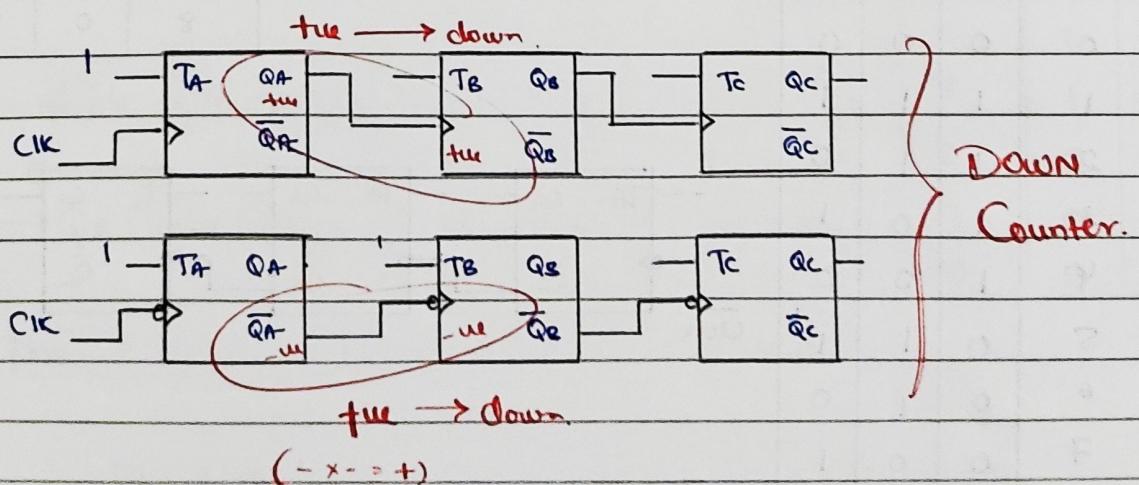
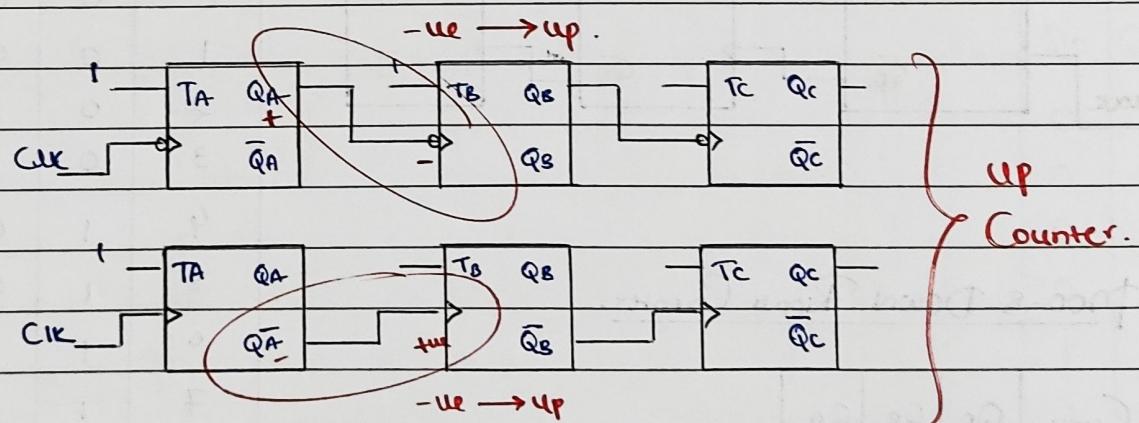
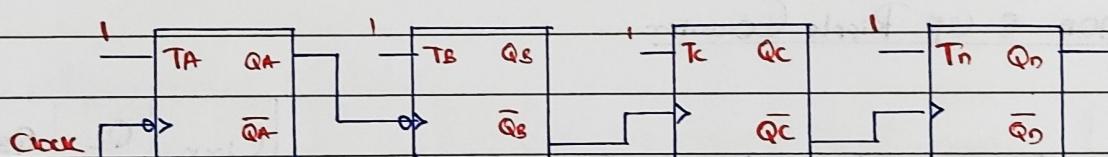
Q2.



Q3.



Q4.



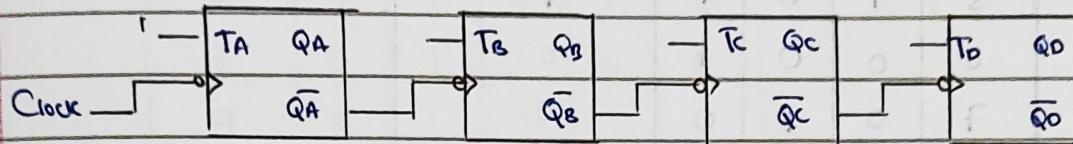
→ Full Mode Counter → when all the states are used by the counter.

$$n=3 \quad M_{100}=8$$

$$n=4 \quad M_{100}=16$$

$$n=5 \quad M_{100}=32$$

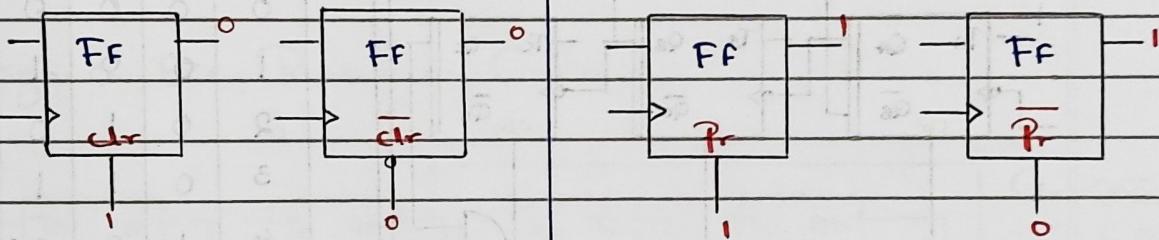
Q1. Design a M_{100} down Counter in which \bar{Q} is taken as Clock.



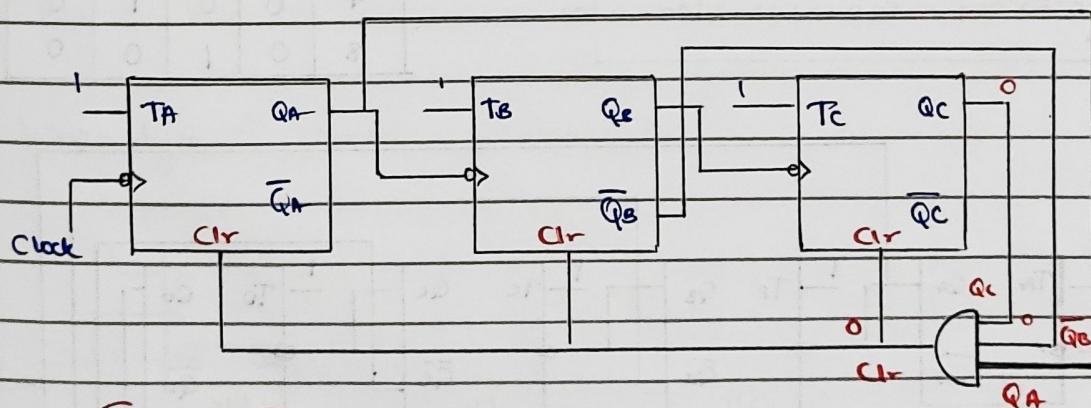
→ Feedback reduces the number of states:

RESET - Clear.

PRESET



Feedback Reduces the Number of States.

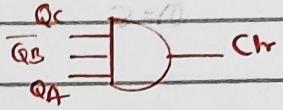


$$Clr = Q_C \bar{Q}_B \bar{Q}_A$$

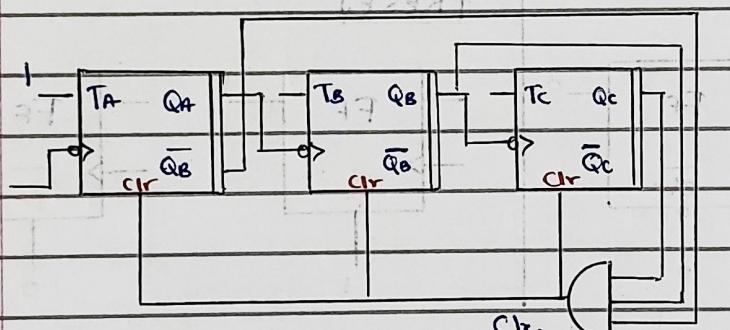
$$= 1 \ 0 \ 1$$

$$= (5)$$

Clock	Qc	QB	QA	$CLR = Qc \bar{Q}B \bar{Q}A$
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	0
4	1	0	0	0
5	X°	Ø°	X°	X°
6	0	0	1	0
7	0	1	0	0
8	0	1	1	0

Mod-6 Up Ripple Counter

$$Clr = Qc \bar{Q}B \bar{Q}A$$



$$Clr = Qc \bar{Q}B \bar{Q}A$$

$$Qc \bar{Q}B \bar{Q}A$$

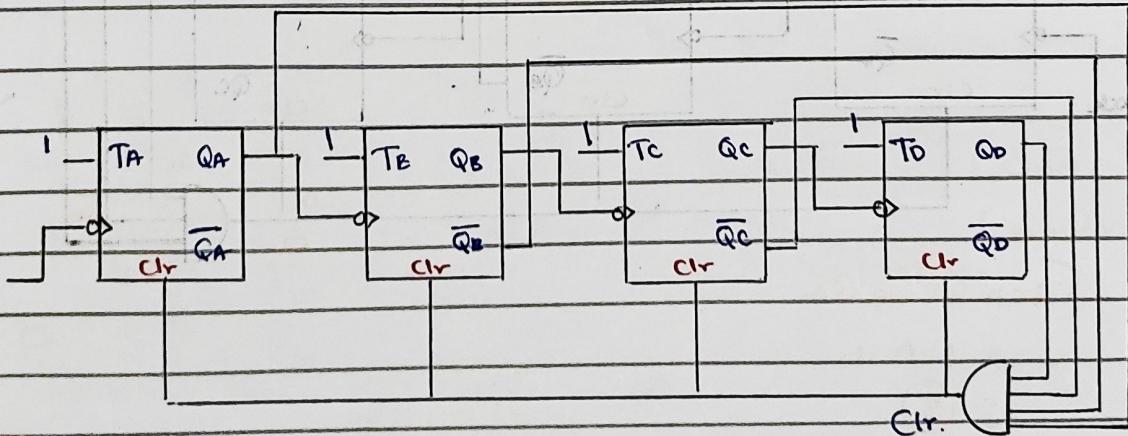
110 → 6

000 → 001 → 010 → 011 → 100 → 101



Clock	Qc	QB	QA	Clr
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	0
4	1	0	0	0
5	1	0	1	0
6	X°	Ø°	Ø°	X°
7	0	0	1	0
8	0	1	0	0

Q.2

Mod-9 Up Ripple Counter

$$Clr = Qc \bar{Q}B \bar{Q}A$$

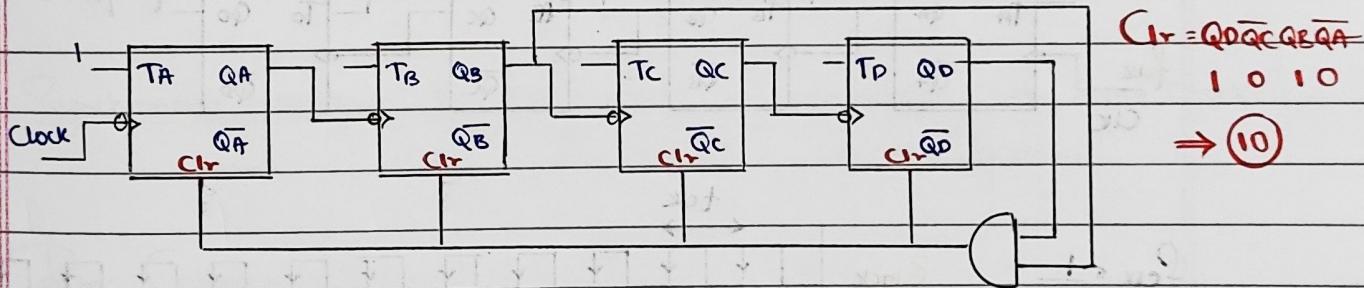
1001 → 9

Clock	Q _D	Q _C	Q _B	Q _A	Clr
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	0
4	0	1	0	0	0
5	0	1	0	1	0
6	0	1	1	0	0
7	0	1	1	1	0
8	1	0	0	0	0
9	X°	∅°	∅°	X°	X°
10	0	0	0	1	
11	0	0	1	0	

$$\text{Clr} = Q_D \bar{Q}_C Q_B \bar{Q}_A$$

1 0 0 1 $\Rightarrow 9$

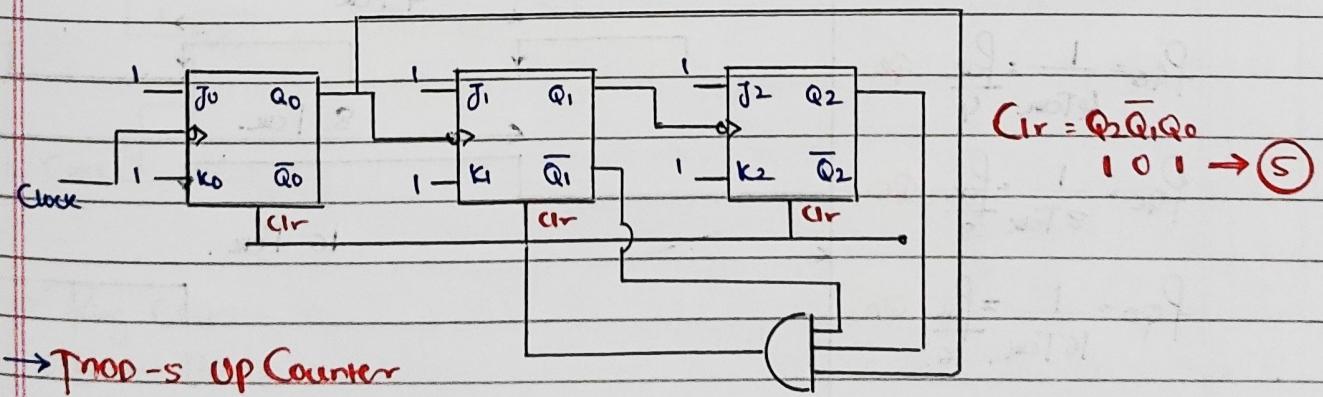
Q.3 Design a BCD Ripple Carry Counter.



$$\text{Clr} = Q_D \bar{Q}_C Q_B \bar{Q}_A$$

1 0 1 0 $\Rightarrow 10$

Q.4 Which type of counter is shown below?



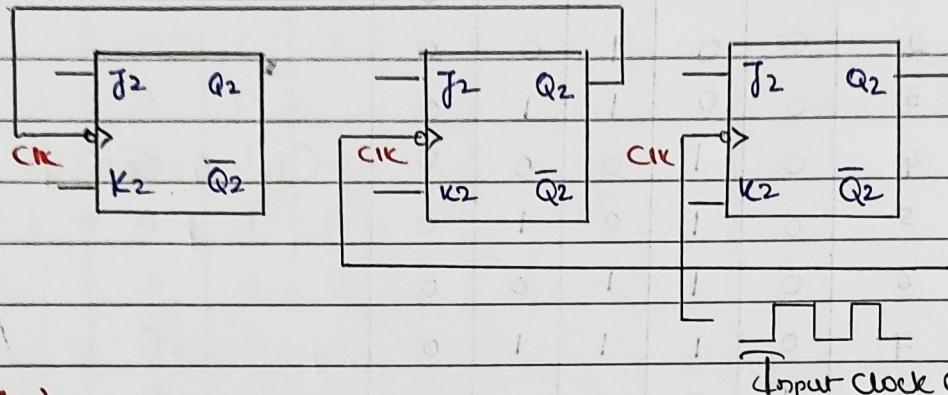
$$\text{Clr} = Q_2 \bar{Q}_1 Q_0$$

1 0 1 $\Rightarrow 5$

\rightarrow T₃₀₀-s Up Counter

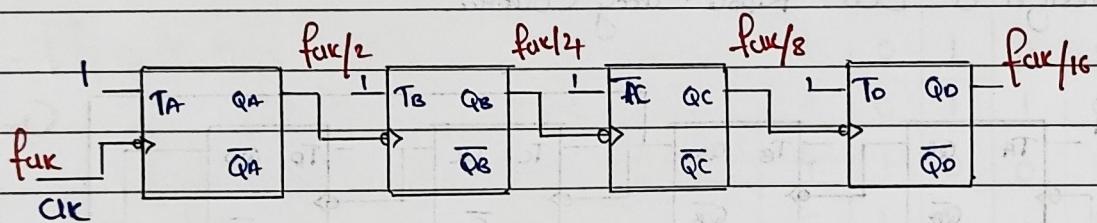
HW

Q5. Consider the following clock. If Counter starts at 000, what will be count after 13 clock pulses?

**HW**

Q6. Design MOD-13, MOD-21, MOD-27, MOD-14 (All up) Ripple Counter

Asynchronous [Ripple Counter].



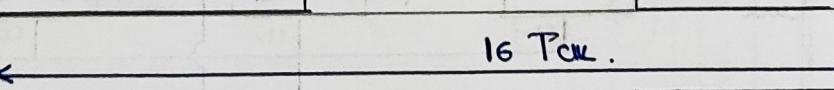
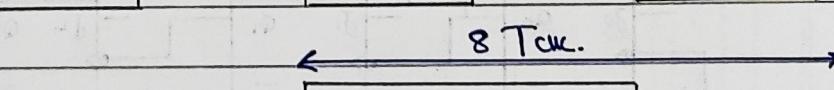
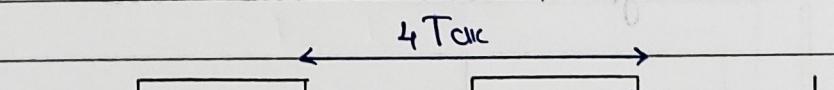
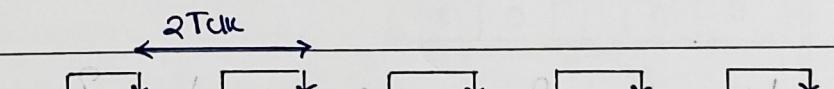
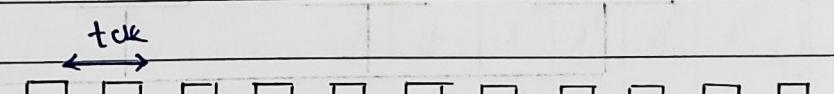
$$f_{C1K} = \frac{1}{2T_{clk}}$$

$$f_{Q_A} = \frac{1}{2T_{clk}} = \frac{f_{C1K}}{2} \quad Q_A$$

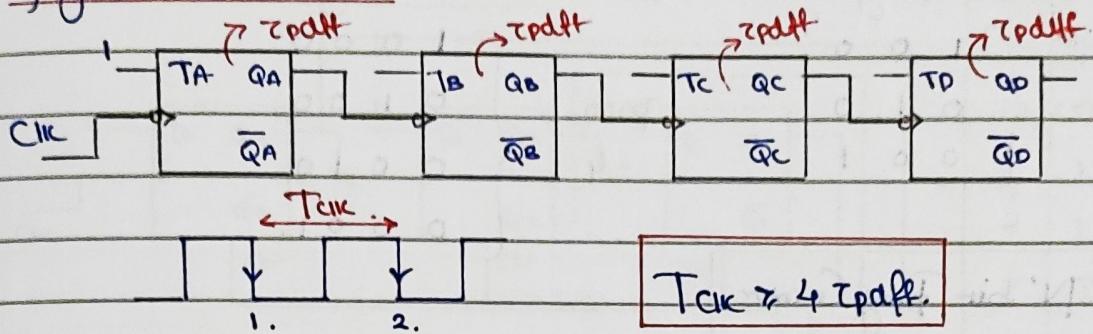
$$f_{Q_B} = \frac{1}{4T_{clk}} = \frac{f_{C1K}}{4} \quad Q_B$$

$$f_{Q_C} = \frac{1}{8T_{clk}} = \frac{f_{C1K}}{8} \quad Q_C$$

$$f_{Q_D} = \frac{1}{16T_{clk}} = \frac{f_{C1K}}{16} \quad Q_D$$



Aynchronous Counter.



n-bit Asynchronous counter

$$T_{clk} \geq n \cdot \tau_{pdff}$$

$$(f_{clk})_{\max} = \frac{1}{n \cdot \tau_{pdff}}$$

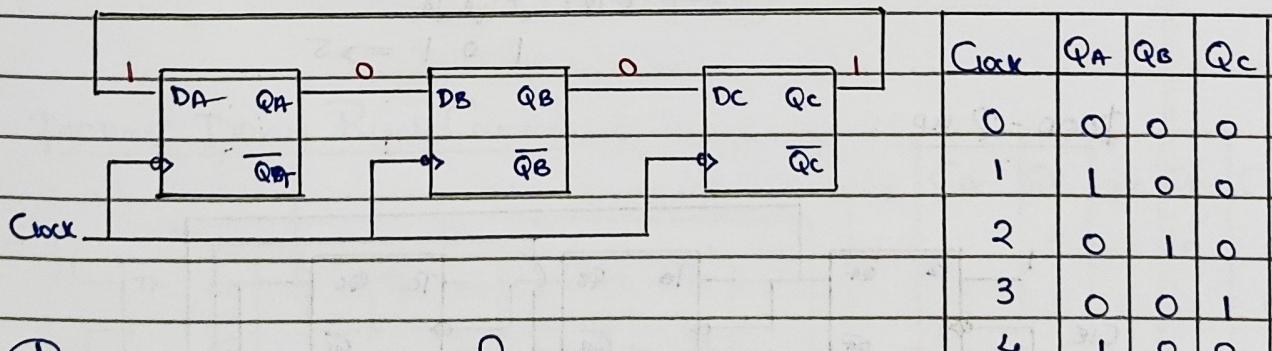
$$\frac{1}{T_{clk}} \leq \frac{1}{n \cdot \tau_{pdff}}$$

$$f_{clk} \leq \frac{1}{n \cdot \tau_{pdff}}$$

Synchronous Counter : All the Flip Flops are Synchronized using Same Clock
 eg: Ring Counter, Johnson Counter.

Ring Counter

→ It is also a SISO Shift register in the form of Ring.



→ Ring Counter is not a self starting counter, to.

Start the ring counter we have to place '1' at MSB then that '1' will rotate among all the flip flops.

3 bit Ring Counter

$$\begin{matrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{matrix}$$

$MOD = 3$

4 bit Ring Counter

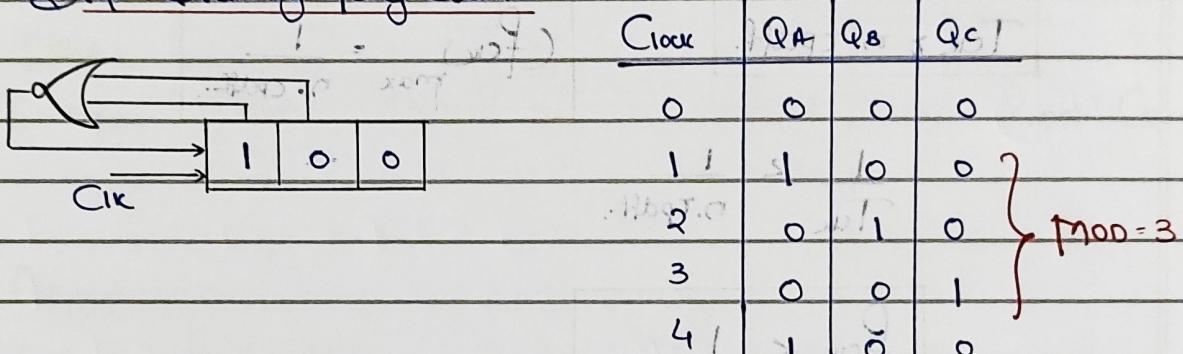
$$\begin{matrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{matrix}$$

$MOD = 4$

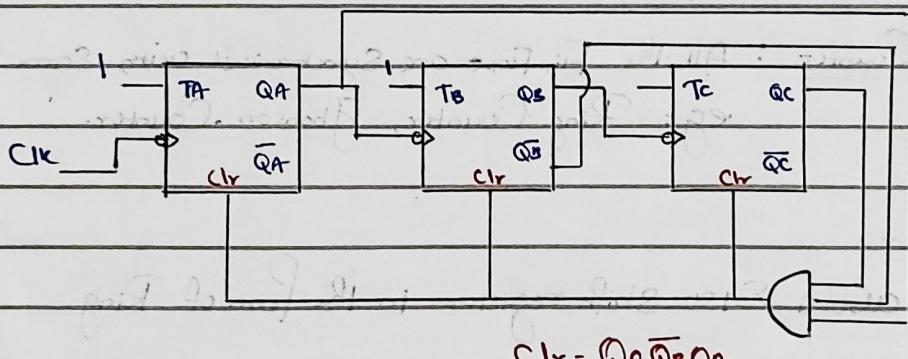
'N' bit Ring Counter

$$\Rightarrow MOD = N$$

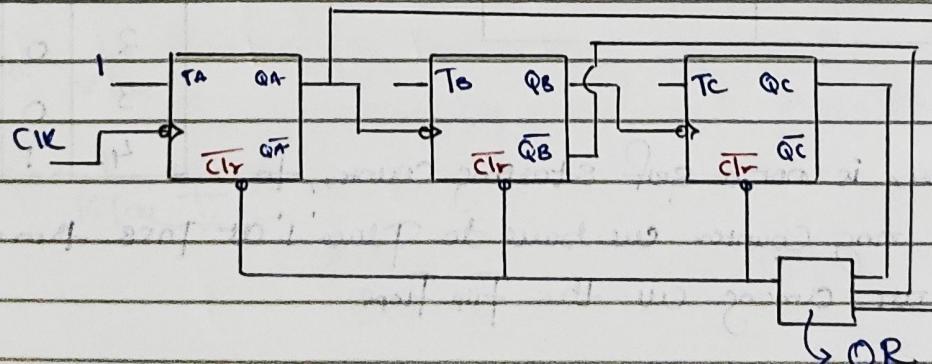
Self Starting Ring Counter

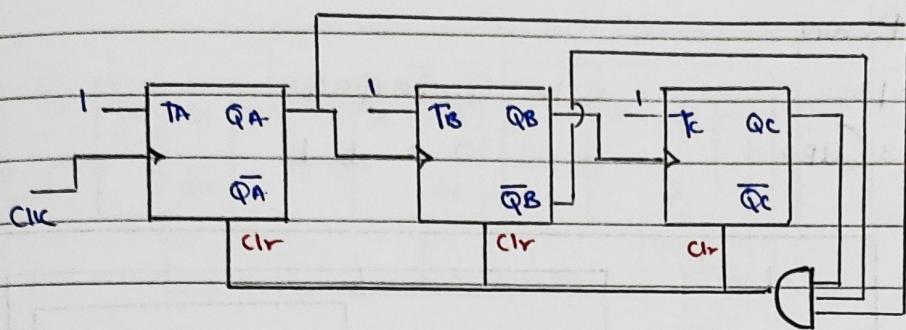


{ Hw }

Assume MOD=5 UPof previous
(class.)

$$101 \Rightarrow 5$$

MOD=2 UP

MOD-3 Down Ripple Counter

$111 \rightarrow 110 \rightarrow 000$

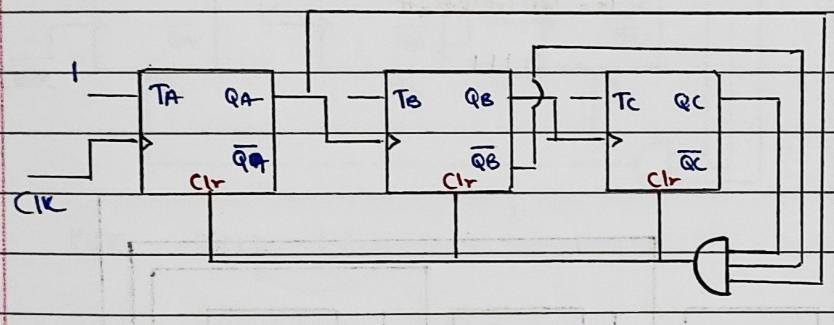
$$Clr = Q_C Q_B Q_A$$

Down = Full-up

$$= 8 - 5$$

$$= 3$$

Clk	Q _C	Q _B	Q _A	Clr
0	0	0	0	0
1	1	1	1	0
2	1	1	0	0
3	X°	∅°	X°	X°
4	1	1	1	0
5	1	1	0	0
c	X°	∅°	X°	X°

MOD-4 Down Ripple Counter

$111 \rightarrow 110 \rightarrow 101 \rightarrow 000$

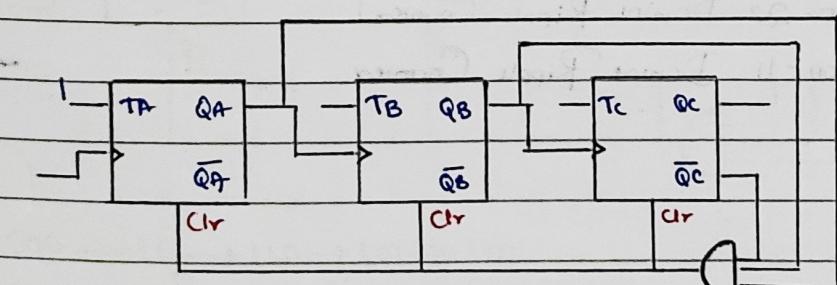
$$Clr = Q_C Q_B Q_A$$

100

Down = Full-up

$$= 8 - 4 \rightarrow 4$$

Clock	Q _C	Q _B	Q _A	Clr
0	0	0	0	0
1	1	1	1	0
2	1	1	0	0
3	1	0	1	0
4	X°	∅°	X°	X°

MOD-5 Down Ripple Counter

$111 \rightarrow 110 \rightarrow 101 \rightarrow 100 \rightarrow 000$

Down = Full-up

$$5 = 8 - 3$$

Clk	Q _C	Q _B	Q _A	Clr
0	0	0	0	0
1	1	1	1	0
2	1	1	0	0
3	1	0	1	0
4	1	0	0	0
5	∅°	X°	X°	1

$$011 \rightarrow 3.$$

Q1. Design a Mod-13 Down Ripple Counter.

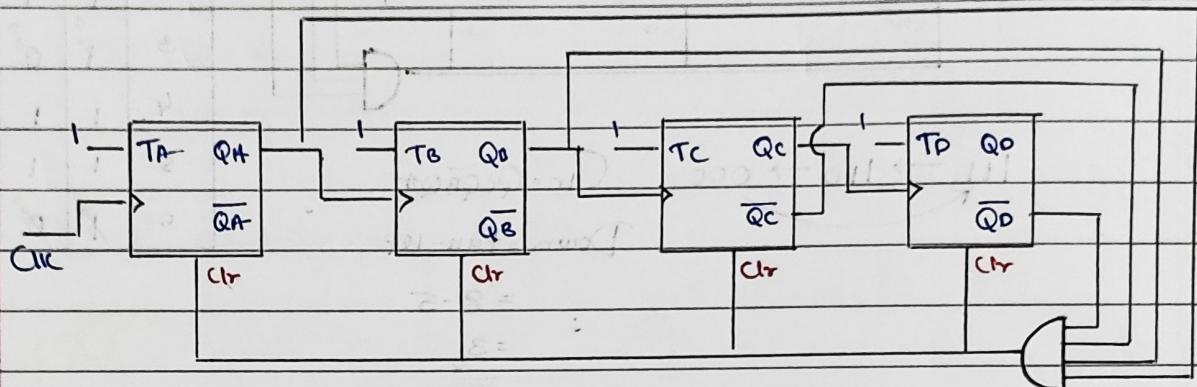
Full - UP = DOWN

$$16 - UP = 13$$

$$16 - 13 = 3 \text{ (UP)}$$

$$Clr = \overline{Q}_D \overline{Q}_C Q_B Q_A$$

0 0 1 1



Q2. Design a mod-9 Down Ripple Counter.

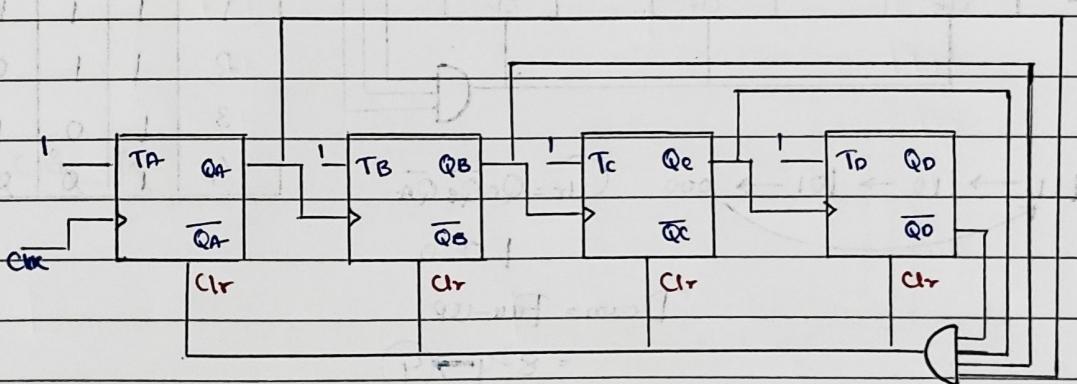
Full - UP = DOWN

$$Clr = \overline{Q}_D \overline{Q}_C Q_B Q_A$$

$$16 - UP = 9$$

$$UP = 16 - 9 = 7.$$

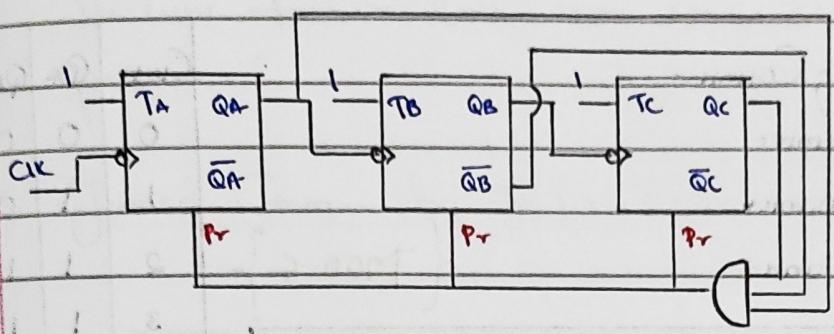
0 1 1 1



Q3. Design a mod-19 Down Ripple Counter.

Q4. Design a Mod-23 Down Ripple Counter.

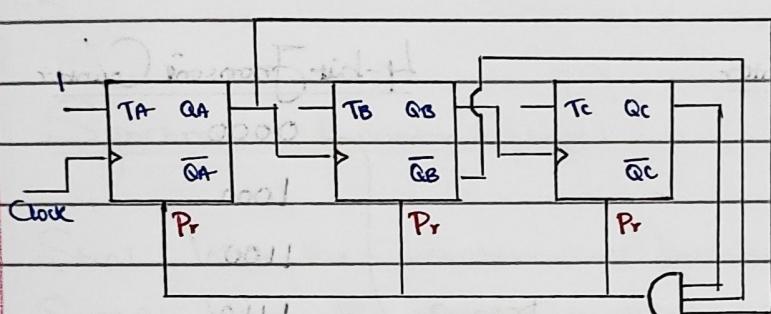
Q5. Design a mod-11 Down Ripple Counter.

T_{MOd-6}

000 → 001 → 010 → 011 → 100 → 111

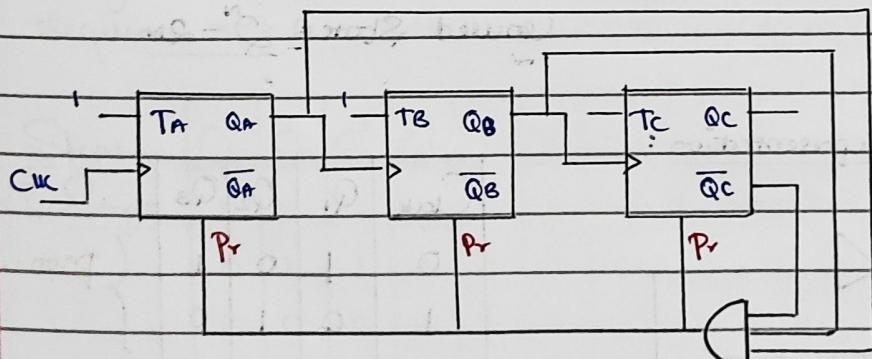
$$Pr = QC\bar{Q}B\bar{Q}A$$

Clock	QC	QB	QA	Pr
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	0
4	1	0	0	0
5	X'	X'	X'	X'
6	0	0	0	0
7	0	0	1	0

T_{MOd-2}

000 → 111 → 110

Clock	QC	QB	QA	Pr
0	0	0	0	0
1	1	1	1	0
2	1	1	0	0
3	X'	X'	X'	X'
4	1	1	0	0
5	X'	X'	X'	X'
6	0	1	0	0
7	X'	X'	X'	1

T_{MOd-4}

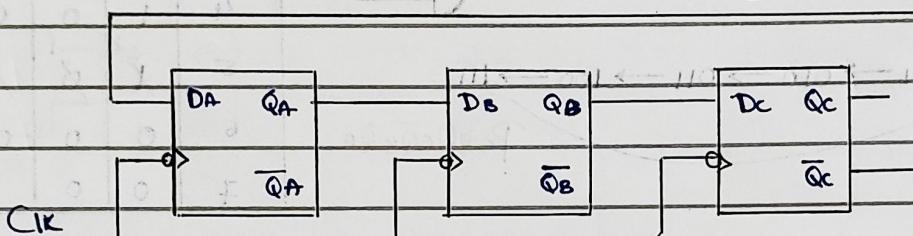
000 → 111 → 110 → 101 → 100

Clock	QC	QB	QA	Pr
0	0	0	0	0
1	1	1	1	0
2	1	1	0	0
3	1	0	1	0
4	1	0	0	0
5	X'	X'	X'	X'
6	1	1	0	0
7	1	0	1	0

Johnson Counter

- Twisted Ring Counter
- Creeping Counter
- Robbins Counter
- Walking Counter

{ MOD-6 }



Clock	Q _A	Q _B	Q _C
0	0	0	0
1	1	0	0
2	1	1	0
3	1	1	1
4	0	1	1
5	0	0	1
6	0	0	0
7	1	0	0

3-bit Johnson Counter

000
100
110
111
011
001

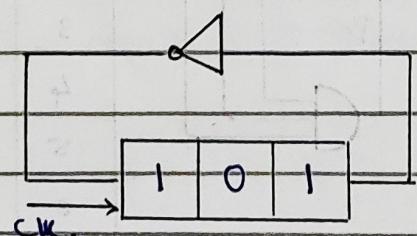
mod
64-bit Johnson Counter

0000
1000
1100
1110
1111
0111
0011
0001

mod-8

→ N-bit Johnson Counter \Rightarrow mod (used state) = 2^N

Unused State = $2^N - 2^N$

Symbolic Representation:

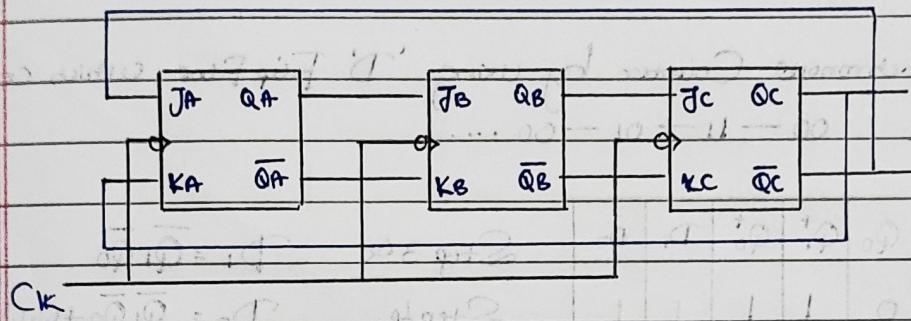
Lockout Problem!

Clock	Q ₁	Q ₂	Q ₃
0	1	0	1
1	0	1	0
2	1	0	1
3	0	1	0
4	1	0	1

{ mod-2 }

Lockout Problem: Whenever Johnson Counter enters its unused state, it will trap or lock into unused state, are called Lockout problem.

Johnson Counter by JK Flip Flop



Synchronous Counter Design:

Step 1: Write the Previous and Present State

Step 2: Write the excitation table of flip flop

Step 3: Write the logical expression

Step 4: Minimise the logical expression

Step 5: Hardware implementation

Q1. Design a Synchronous Counter using "T" Flip Flop which counts the sequence $\rightarrow 0 - 3 - 1 - 2 - 0 \dots \{ 00 - 11 - 01 - 10 \rightarrow 00 \dots \}$

↳ 2 FFs are required.

Step 1 & Step 2	Q ₁	Q ₀	Q ⁺	Q ₀ ⁺	T _i	T ₀
	0	0	1	1	1	1
	0	1	1	0	1	∅
	1	0	0	0	1	0
	1	1	0	1	1	0

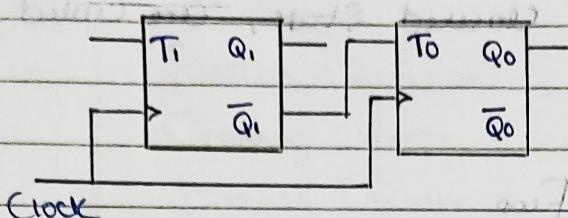
Step 3

& Step 4: T_i = 1

$$T_0 = \overline{Q}_1 \overline{Q}_0 + \overline{Q}_1 Q_0$$

$$T_0 = \overline{Q}_1 [\overline{Q}_0 + Q_0]$$

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

Step 5:

Justification:

Clock	Q_1	Q_0
0	0	0
1	1	1
2	0	1
3	1	0

Q2. Design a Synchronous Counter by using 'D' Flip Flops which count the Sequence 00 → 11 → 01 → 00 ...

Method 1Step 1 &

Q_1	Q_0	Q_1^+	Q_0^+	D_1	D_0
0	0	1	1	1	1
1	1	0	1	0	1
0	1	0	0	0	0

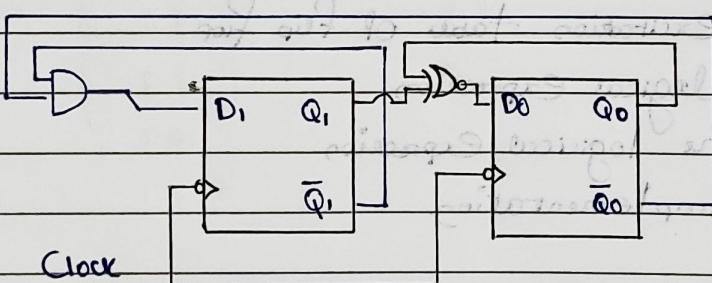
Step 3 &

$$D_1 = \bar{Q}_1 \bar{Q}_0$$

Step 2Step 4

$$D_0 = \bar{Q}_1 Q_0 + Q_1 Q_0$$

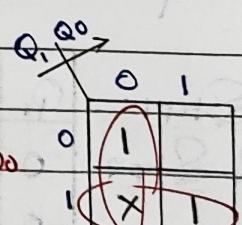
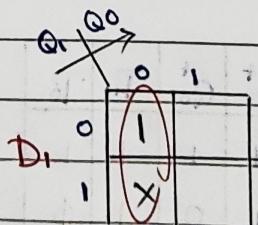
$$D_0 = Q_1 \oplus Q_0$$

Step 5:

Clock	Q_1	Q_0
0	0	0
1	1	1
2	0	1
3	0	0

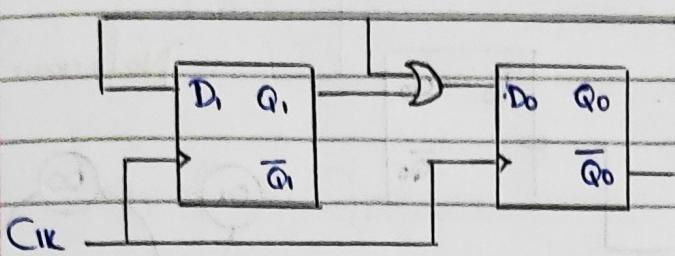
Method 2

Q_1	Q_0	Q_1^+	Q_0^+	D_1	D_0
0	0	1	1	1	1
0	1	0	0	0	0
1	0	x	x	x	x
1	1	0	1	0	1

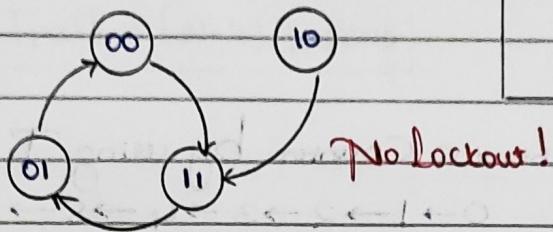


$$D_1 = \bar{Q}_0$$

$$D_0 = Q_1 + \bar{Q}_0$$



Clock	Q1	Q0
0	0	0
1	1	1
2	0	1
3	0	0
4	1	1
5	0	1

HW

Q3. Design a synchronous Counter by using T-FF which count:

000 → 001 → 010 → 011 → 100 → 101 → 110 → 111 → 000 ...

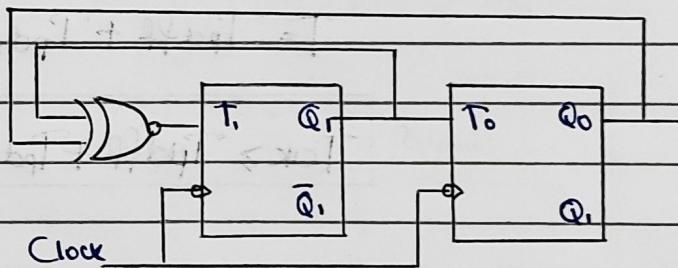
Q4. Design a synchronous Counter by using T-flip flop which count the Sequence 0 → 2 → 3 → 0 ...

00 → 10 → 11 → 00

2bin → 2 flip flops are required.

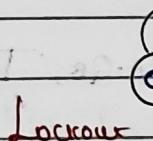
Method 1

Q1	Q0	Q1'	Q0'	T1	T0
0	0	1	0	1	0
1	0	1	1	0	1
1	1	0	0	1	1



$$T_1 = \bar{Q}_1 \oplus Q_0$$

$$T_0 = \underline{Q_1}$$

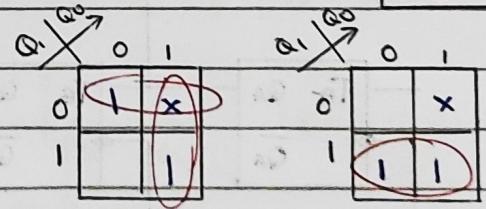


Problem!

Clock	Q1	Q0
0	0	0
1	1	0
2	1	1
3	0	0

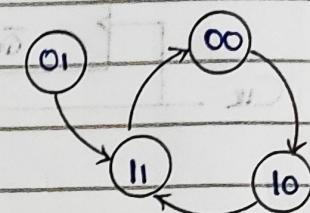
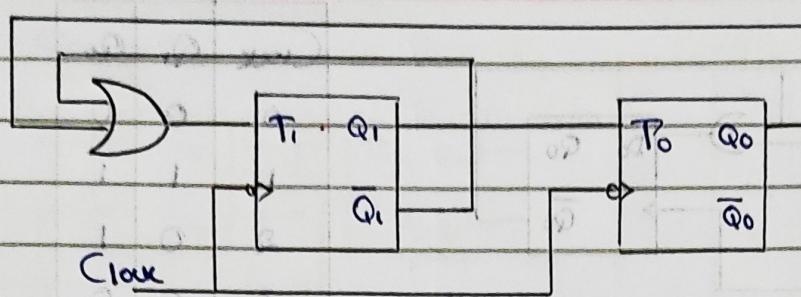
Method 2

Q1	Q0	Q1'	Q0'	T1	T0
0	0	1	0	1	0
0	1	x	x	x	x
1	0	1	1	0	1
1	1	0	0	1	1



$$T_1 = \bar{Q}_1 + Q_0$$

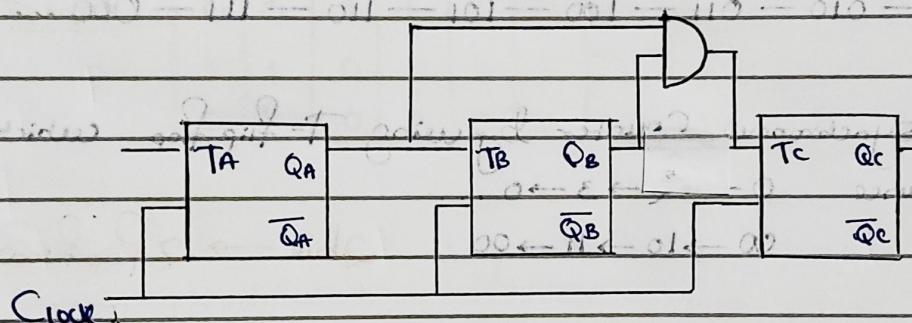
$$T_0 = \underline{Q_1}$$



Q1. Design a synchronous Counter by using T Flip-Flop which count the Sequence $0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 0$.

Three bit \rightarrow Three Flip-Flops Required

$$\rightarrow T_A = 1, T_B = Q_A, T_C = Q_A \cdot Q_B$$



$$T = T_{pdff} + T_{pdAND}$$

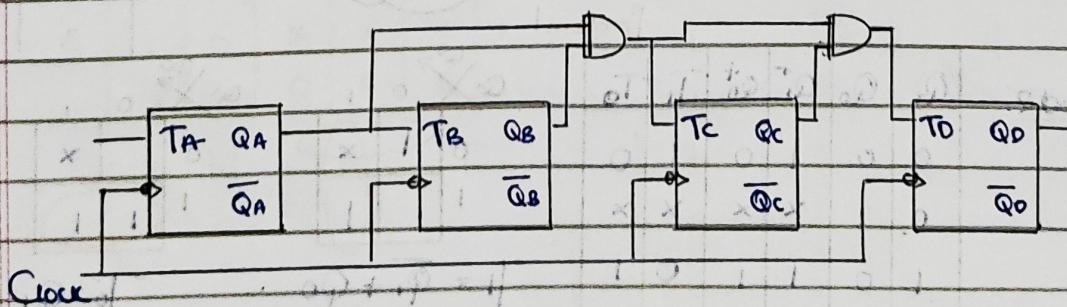
$$\frac{1}{f_{clk}} \leq \frac{1}{T_{pdff} + T_{pdAND}}$$

$$T_{clk} \geq T_{pdff} + T_{pdAND}$$

$$f_{clk} \leq \frac{1}{T_{pdff} + T_{pdAND}}$$

4 bit

$$T_A = 1, T_B = Q_A, T_C = Q_A \cdot Q_B, T_D = Q_A \cdot Q_B \cdot Q_C$$



$$T_{CK} \geq T_{pdff} + 2 \cdot T_{pdAND}$$

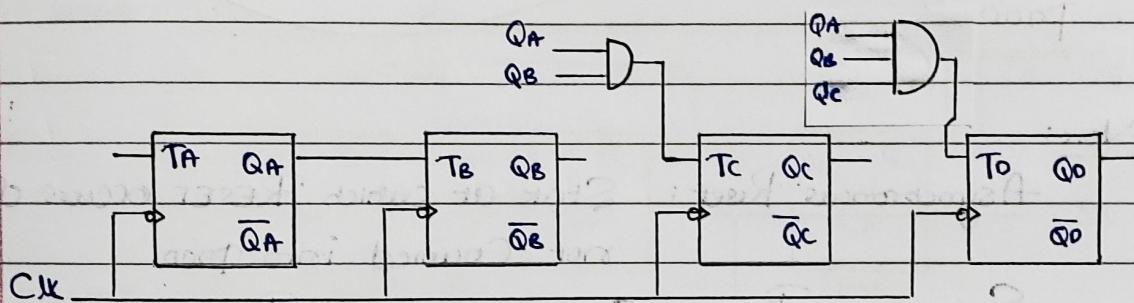
$$f_{CK} \leq \frac{1}{T_{pdff} + 2 T_{pdAND}}$$

Series Carry Synchronous Counter:

$$T_{CK} \geq T_{pdff} + (N-2) T_{pdAND}$$

$$f_{CK} \leq \frac{1}{T_{pdff} + (N-2) T_{pdAND}}$$

Parallel Carry Synchronous Counter:

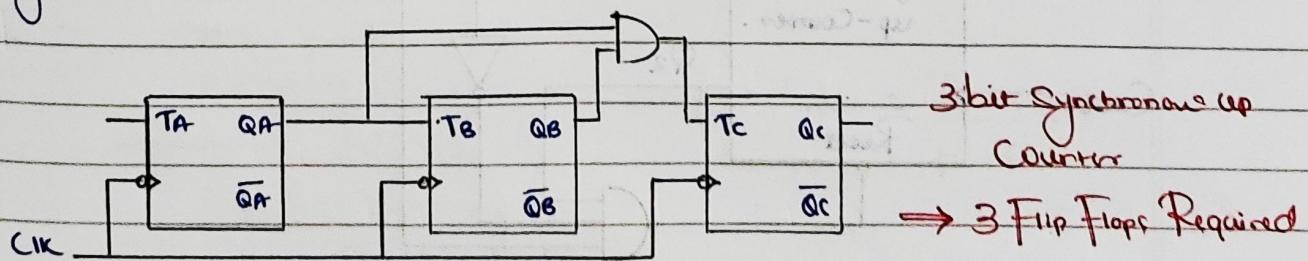


$$T_{CK} \geq T_{pdff} + T_{pdAND}$$

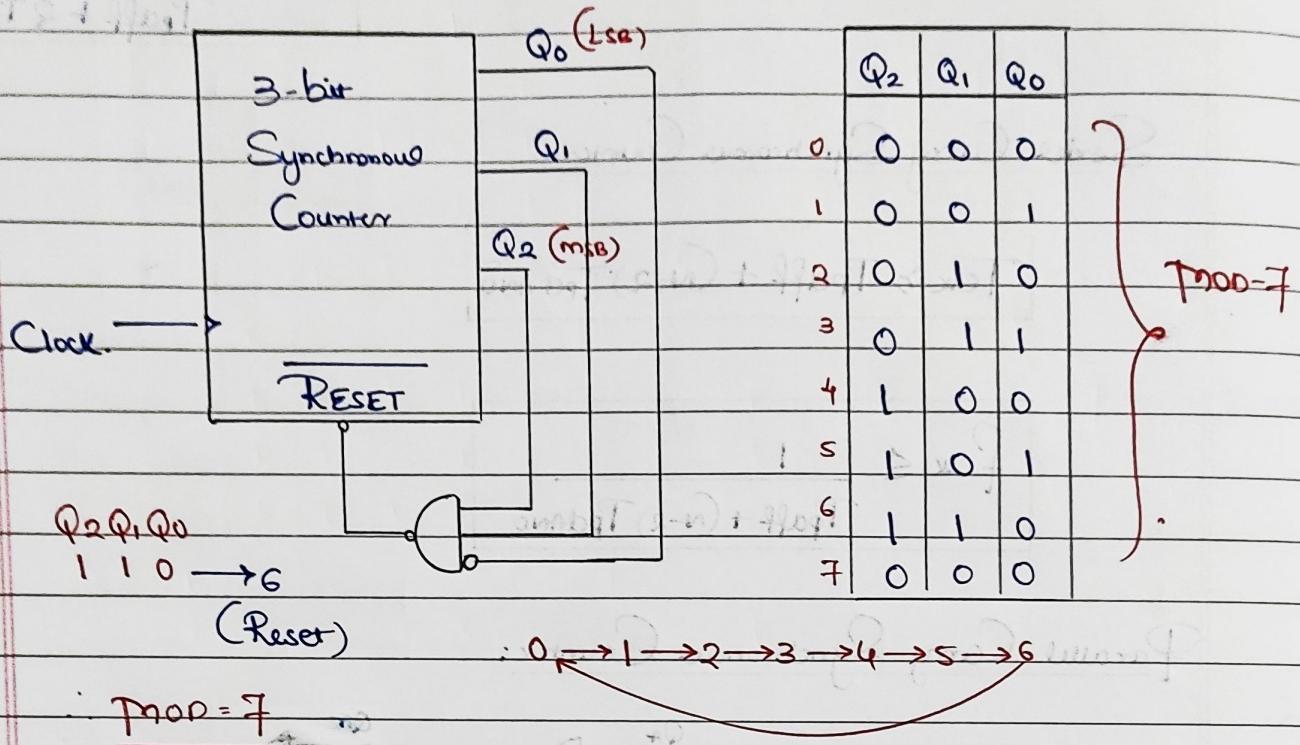
$$f_{CK} \leq \frac{1}{T_{pdff} + T_{pdAND}}$$

Q1. Minimum no. of flip flops required to construct BCD Counter is
Ans. 0000 } 4 bits → Four Flip Flops.
1001 }

Q2. Minimum no. of Flip Flops required to design Counter state given below $0 \rightarrow 0 \rightarrow 1 \rightarrow 1 \rightarrow 2 \rightarrow 2 \rightarrow 3 \rightarrow 3 \rightarrow 0 \rightarrow 0 \dots$



Q3. What is the mod of the counter?



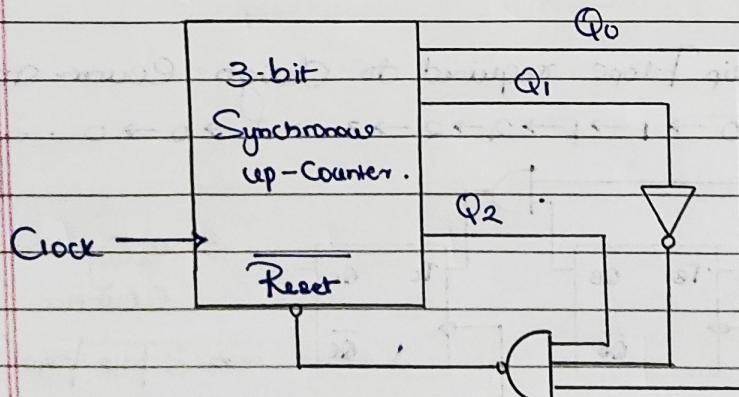
Note:

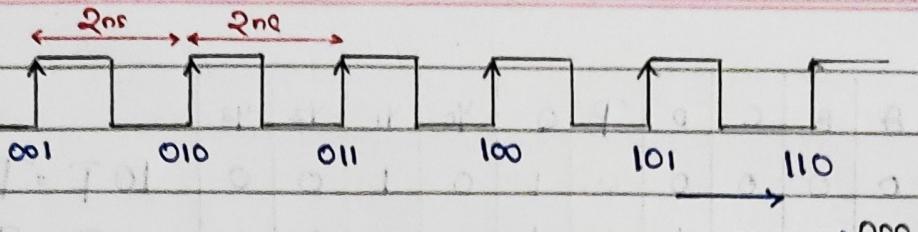
A Synchronous Reset: State at which RESET occurs does not Counted into T_{mod} .

A Synchronous Reset: State at which RESET occurs is counted into T_{mod} .

Q4. Consider the below circuit:

The delay of NAND, NOR, gate is 3ns, 1ns respectively and that of the Counter is assumed to be zero. If the clock frequency is $500 MHz$, then the Counter becomes as:



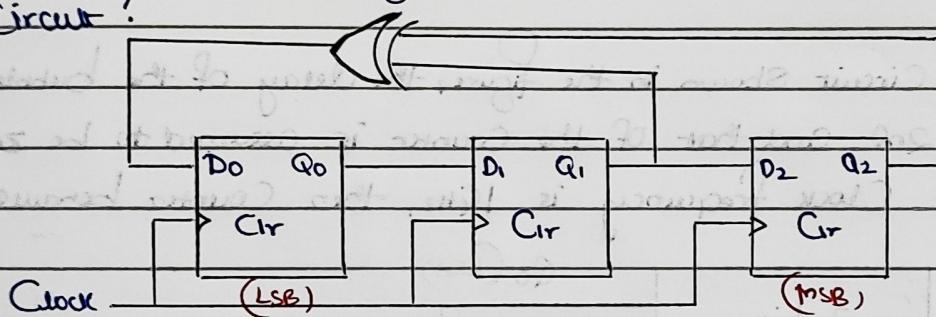


$$\text{Cir} = Q_2 \bar{Q}_1 Q_0$$

$$f_{CK} = 500 \times 10^6 \text{ Hz}$$

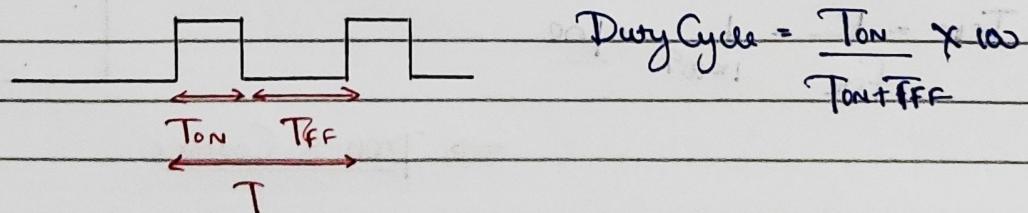
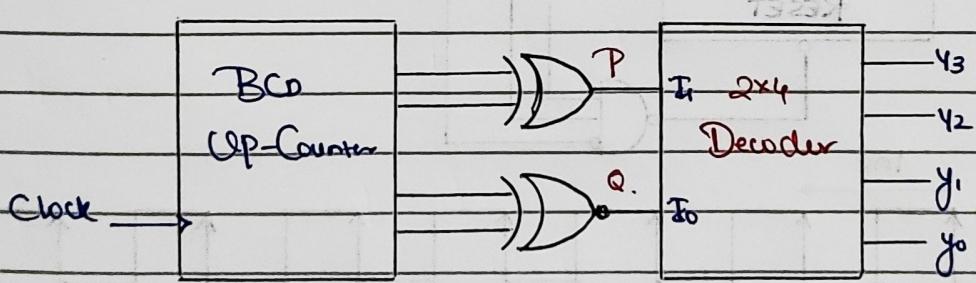
$$T_{CK} = \frac{1}{500 \times 10^6} \text{ Sec} = \frac{1000 \times 10^{-9}}{500} = 2 \text{ ns}$$

Qs. Consider the Circuit below with initial state $Q_0 = 1$, $Q_1 = Q_2 = 0$. The state of the circuit is given by the value of $4Q_2 + 2Q_1 + Q_0$. Which of the following is the correct state sequence of the circuit?



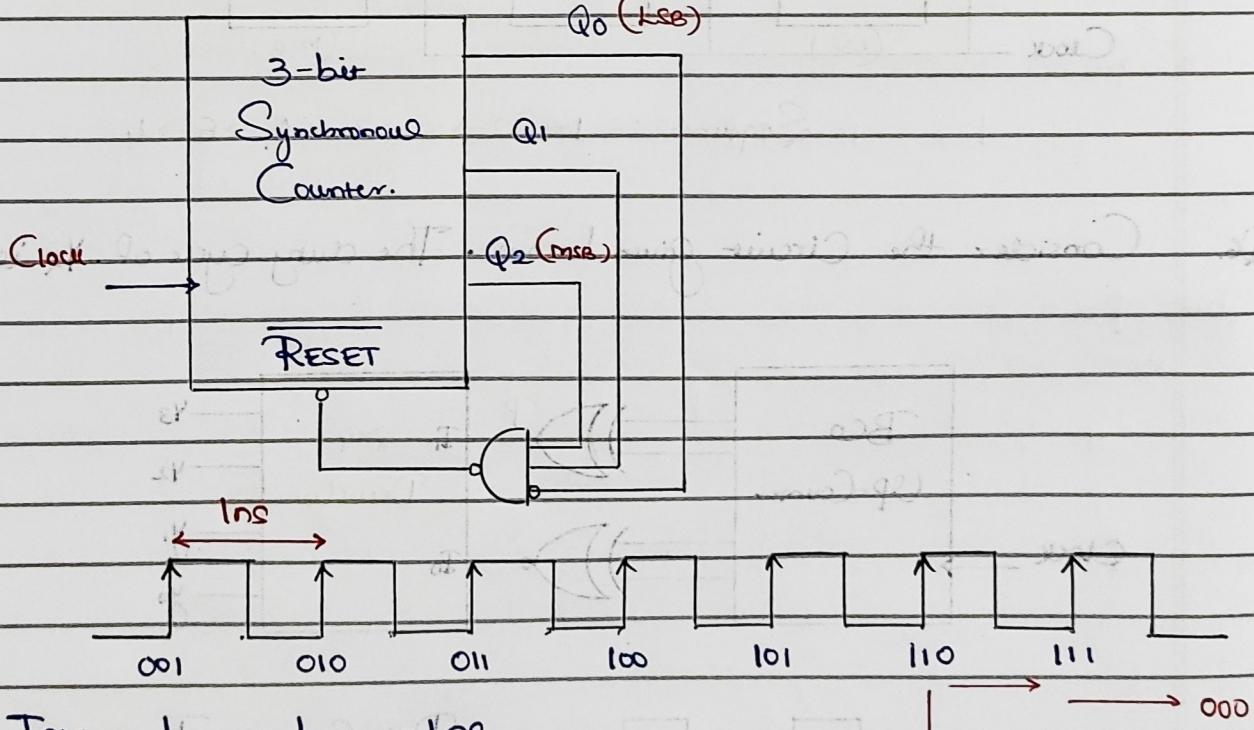
Sequence : $1 \rightarrow 2 \rightarrow 5 \rightarrow 3 \rightarrow 7 \rightarrow 6 \rightarrow 4$

Q6. Consider the circuit given below. The duty cycle of Y_2 is



A	B	C	D	P	Q	Y_0	Y_1	Y_2	Y_3	
0	0	0	0	0	1	0	1	0	0	$10T = \text{Total time}$
0	0	0	1	1	1	0	0	0	1	$3T = T_{on}$
0	0	1	0	1	1	0	0	0	1	
0	0	-1	1	0	1	0	1	0	0	$D = \frac{3T \times 10^3}{10T}$
0	1	0	0	0	0	1	0	0	0	
0	1	0	1	1	0	0	0	1	0	$= 30\%$
0	1	1	0	1	0	0	0	1	0	
0	1	1	1	0	0	1	0	0	0	
1	0	0	0	0	0	1	0	0	0	
1	0	0	1	1	0	0	0	1	0	

Q7. For the Circuit shown in the figure, the delay of the bubbled NAND gate is 2ns and that of the Counter is assumed to be zero. If the Clock frequency is $1f_{CK}$, then Counter behaves as a:



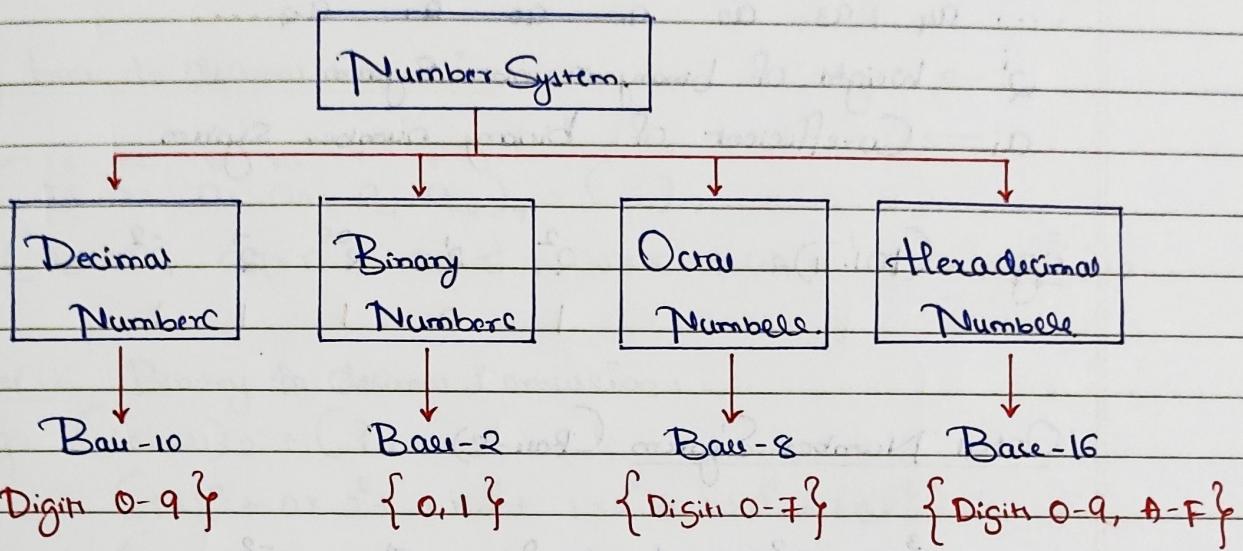
$$T_{CK} = \frac{1}{f_{CK}} = \frac{1}{1 \times 10^9} = 1 \text{ ns}$$

\Rightarrow T_{mod-8} Counter

NUMBER SYSTEM

Base (Radix)

Total number of digits used in the System.



Decimal Number System

$$\dots \quad 10^4 \quad 10^3 \quad 10^2 \quad 10^1 \quad 10^0 \quad 10^{-1} \quad 10^{-2} \quad 10^{-3} \dots$$

$$\dots \quad a_4 \quad a_3 \quad a_2 \quad a_1 \quad a_0 \quad a_{-1} \quad a_{-2} \quad a_{-3} \dots$$

a_i → Co-efficient of decimal number system.

10^i → Weight of decimal number system.

e.g. $(501.23)_{10}$

$$10^2 \quad 10^1 \quad 10^0 \quad 10^{-1} \quad 10^{-2}$$

5 0 1 2 3

Base	Digit
2	0 1
3	0 1 2
4	0 1 2 3
...	...
14	0 1 2 3 4 5 6 7 8 9 A B C D
15	0 1 2 3 4 5 6 7 8 9 A B C D E
16	0 1 2 3 4 5 6 7 8 9 A B C D E F

Binary Number System (Base=2)

$$\dots 2^4 \quad 2^3 \quad 2^2 \quad 2^1 \quad 2^0 \quad 2^{-1} \quad 2^{-2} \dots$$

$$\dots a_4 \quad a_3 \quad a_2 \quad a_1 \quad a_0 \quad a_{-1} \quad a_{-2} \dots$$

2^i → weight of binary number system.

a_i → Co-efficient of binary number system.

eg: $(101.11)_2$

2^2	2^1	2^0	2^{-1}	2^{-2}
1	0	1	1	1

Octal Number System (Base=8)

$$\dots 8^3 \quad 8^2 \quad 8^1 \quad 8^0 \quad 8^{-1} \quad 8^{-2} \dots$$

$$\dots a_3 \quad a_2 \quad a_1 \quad a_0 \quad a_{-1} \quad a_{-2} \dots$$

8^i → weight of Octal number system

a_i → Co-efficient of Octal number system {0-7}.

eg: $(720.64)_8$

8^2	8^1	8^0	8^{-1}	8^{-2}
7	2	0	6	4

Hexadecimal Number System (Base=16)

$$\dots 16^3 \quad 16^2 \quad 16^1 \quad 16^0 \quad 16^{-1} \quad 16^{-2} \dots$$

$$\dots a_3 \quad a_2 \quad a_1 \quad a_0 \quad a_{-1} \quad a_{-2} \dots$$

16^i → weight of Hexadecimal number system

a_i → Co-efficient of Hexadecimal number system {0-9, A-F}.

eg: $(A2C.F)_{16}$

16^2	16^1	16^0	16^{-1}
A	2	C	F

In base Conversion there are two key points:

- I. Any base to decimal conversion.
- II. Decimal to any other base conversion.

Any base to decimal Conversion:

$$(a_3 \underset{r^3}{a_2} \underset{r^2}{a_1} \underset{r^1}{a_0} \underset{r^{-1}}{a_1} \underset{r^{-2}}{a_2})_r = (?)_{10}$$

$$(a_3 \times r^3 + a_2 \times r^2 + a_1 \times r^1 + a_0 \times r^0 + a_1 \times r^{-1} + a_2 \times r^{-2})_{10}$$

Case(1) : Binary to decimal Conversion.

$$\text{eg: } (1011.11)_2 = (?)_{10}$$

$$= (1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2})$$

$$= (8 + 0 + 2 + 1 + 0.5 + 0.25)$$

$$= (11.75)_{10}$$

Case (2) : Octal to Decimal Conversion

$$\text{eg: } (721.4)_8 = (?)_{10}$$

$$= (7 \times 8^2 + 2 \times 8^1 + 1 \times 8^0 + 4 \times 8^{-1})$$

$$= 448 + 16 + 1 + 0.5$$

$$= (465.5)_{10}$$

Case (3) : Hexadecimal to Decimal Conversion

$$\text{eg: } (\text{AEB.C})_{16} = (?)_{10}$$

$$= (A \times 16^2 + E \times 16^1 + B \times 16^0 + C \times 16^{-1})$$

$$= (10 \times 256 + 14 \times 16 + 11 \times 1 + 12 \times 16^{-1})$$

$$= (2603.75)_{10}$$

Case (4) : Base 5 to Decimal Conversion

$$\text{eg: } (432.22)_5 = (?)_{10}$$

$$= (4 \times 5^2 + 3 \times 5^1 + 2 \times 5^0 + 2 \times 5^{-1} + 2 \times 5^{-2})$$

$$= (100 + 15 + 2 + 0.4 + 0.08)$$

$$= (117.48)_{10}$$

Q1.

$$(6_{12}c) = (?)_{10}$$

By minimum base Convert into decimal:

$$(6_{12}c)$$

$\hookrightarrow 13, 14, 15$, minimum = 13.

$$= (6 \times 13^2 + 2 \times 13^1 + c \times 13^0)$$

$$= (1052)_{10}$$

Decimal to any other base Conversion:

τ	$a_3 \ a_2 \ a_1 \ a_0$	Remainder	$a_3 \ a_2 \ a_1 \ a_0 \ . \ a_1 \ a_2$
		b_0	b_0
		b_1	b_1
		b_2	b_2
		b_3	b_3

Before After
decimal decimal

$$\begin{aligned}
 0 \cdot a_1 a_2 a_3 \times \tau &= x_0 \xrightarrow{x_0} x_0 \\
 0 \cdot x_1 x_2 \times \tau &= x_1 \xrightarrow{x_1} x_1 \\
 0 \cdot x_3 x_4 \times \tau &= x_2 \xrightarrow{x_2} x_2
 \end{aligned}$$

$$(a_3 a_2 a_1 a_0 \ . \ a_1 a_2 a_3)_{10}$$

$$= (b_3 b_2 b_1 b_0 \ . \ x_0 x_1 x_2)_\tau$$

Case (1): Decimal to Binary Base Conversion:

$$\text{eg: } (19.75)_{10} = (?)_2$$

2 19	1		
2 9	1	↑	0.75 \times 2 = 1.5
2 4	0		1
2 2	0		0.5 \times 2 = 1.0
1	1		1

After decimal

$$(19.75)_{10} = (10011.11)_2$$

Case (2): Decimal to Octal base Conversion:

$$\text{eg: } (210.23)_{10} = (?)_8$$

8 210	2	↑
8 26	2	
3	3	3

$$\begin{aligned} 0.23 \times 8 &= 1.84 \\ 0.84 \times 8 &= 6.72 \\ 0.72 \times 8 &= 5.76 \end{aligned}$$

$$(210.23)_{10} = (322.165)_8$$

Case (3) : Decimal to Hexadecimal base Conversion.

$$\text{eg: } (1228.55)_{10} = (?)_{16}$$

$$\begin{array}{r} 16 | 1228 & 12(C) \\ 16 | 76 & 12(C) \\ \downarrow & \downarrow \\ 4 & 4 \end{array} \quad \begin{array}{r} 0.55 \times 16 = 8.8 & 8 \\ 0.84 \times 16 = 12.8 & 12(C) \\ \downarrow & \downarrow \\ & 16 \end{array}$$

$$(1228.55)_{10} = (44C.8C)_{16}$$

Some Special Case

Case (1) : Binary to Octal base Conversion

$$\text{eg: } (10110111)_2 = (?)_8$$

Octal = base 8 $\Rightarrow 8 = 2^3$. (\because every 3 bin/digit of binary represent 1 oval).

010 110 111
2 6 7

$$\text{Hence: } (10110111)_2 = (267)_8$$

Case (2) : Binary to Hexadecimal base conversion.

$$\text{eg: } (10110111)_2 = (?)_{16}$$

Hexadecimal = base 16 $= 2^4$ (\because every 4 digits of binary \rightarrow 1 digit of hexadecimal)

0101 1011

5 11(B)

$$\text{Hence } (10110111)_2 = (5B)_{16}$$

$$\text{Q2. } (330123)_4 = (?)_8$$

$$\text{Method 1: } (330123)_4 = (?)_{10} = (?)_8$$

$$\text{Method 2: } (330123)_4 = (?)_2 = (?)_8$$

$$4 = 2^2$$

$$\begin{array}{r} (111\ 100\ 011\ 011) \\ (7433)_8 \end{array} \quad \begin{array}{l} \hookdownarrow 2^3 \\ \hookdownarrow 2^3 \end{array}$$

BCD (Binary Coded Decimal):

In this each digit is represented by its four-bit binary equivalent. It is also called as natural BCD or 8421-code and it is weighted code.

Excess-3 Code: This is a non weighted binary code used for decimal digits. Its code assignment is obtained from the corresponding value of BCD after the addition of 3.

Bco (Binary Coded Octal): In this each digit of the octal number is represented by its three-bit binary equivalent.

BCH (Binary Coded Hexadecimal): In this each digit of the hexa decimal number is represented by its four bit binary equivalent.

Decimal	BCD 8421	Excess-3	Octal	BCO	Hexadecimal	BCH
0	0000	0011	0	000	0	0000
1	0001	0100	1	001	1	0001
2	0010	0101	2	010	2	0010
3	0011	0110	3	011	3	0011
4	0100	0111	4	100	4	0100
5	0101	1000	5	101	5	0101
6	0110	1001	6	110	6	0110
7	0111	1010	7	111	7	0111
8	1000	1011			8	1000
9	1001	1100			9	1001
					A	1010
					B	1011
					C	1100
					D	1101
					E	1110
					F	1111

Q1. Find the base which satisfy the equation $\left[\left(\frac{39}{3} \right)_x = 13 \right]_x$

Ans $\frac{(39)}{(3)_x} = (13)_x$

$$\frac{3x+9}{3} = x+3$$

$3x+9 = 3x+9 \rightarrow$ This equation satisfies for all value of x
 \therefore Any base > 9 .

Q2. Find the base value which can satisfy the following two equations simultaneously.

(i) $2+3=5$

(i) $2_x + (3)_x = (5)_x$

(ii) $2 \times 4 = 10$

(ii) $2 \times 4 = 10$

$$2 \times x^0 + 3 \times x^0 = (5 \times x^0)_x$$

$$(2 \times x^0)_x \times (4 \times x^0) = (1 \times x^1 + 0 \times x^0)_x$$

Ans

$$2+3=5$$

$$x > 5$$

$$x > 6$$

$$2 \times 4 = 2+0$$

$$x = 8$$

$\therefore x = 8$ will satisfy both.

Q3. Find the number of solutions of x and y to satisfy $(43)_8 = (20)_y$

Ans

$$y > x \quad (43)_8 = (20)_y$$

$$(4 \times 8^1 + 3 \times 8^0)_{10} = (2 \cdot y + 0 \cdot x y)_y$$

$$35 = x \cdot y$$

\therefore No. of solutions = 2

x	y
35	1
1	35
5	7
7	5

Q4. Find the number of solutions of x & y to satisfy $(123)_5 = (28)_y$

Ans

$$y > x \quad (123)_5 = (28)_y$$

$$y > x \quad (1 \times 5^2 + 2 \times 5^1 + 3 \times 5^0) = x y + 8 x y^0$$

$$25 + 10 + 3 = x y + 8 x y^0$$

$$xy = 30$$

\therefore No. of solutions = 3

x	y
1	30
2	15
3	10
5	6

Q5. Result of addition of $34+43$ performed on minimum base is stored in an 8-bit register. The content will be

Ans $(34)_5 + (43)_5$

$$(3 \times 5^1 + 4 \times 5^0) + (4 \times 5^1 + 3 \times 5^0) \\ = (42)_{10} \rightarrow \underline{\underline{00101010}}$$

Q6 $7 \times 5^2 + 3 \times 250 + 6 \times 125 + 5 \times 25 + 3 \times 16 + 3$

If the number can be written in binary then number is present
can be?

Ans $2^9 - 1 = (4+2+1)2^9 + (2+1)2^8 + (4+2)2^7 + (1+1)2^6 + (3+1)2^5 + 2+1$
 $2^{10} - 1 = 2^9 + 2^8 + 2^7 + 2^6 + 2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0$
 $2^9 = 1000 - 1 = 2^8 + 2^7 + 2^6 + 2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0$
 $2^8 = 1000 - 1 = 2^7 + 2^6 + 2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0$
 $\Rightarrow \underline{\underline{1}}$

Magnitude Representation

	Unsigned	Signed.	Complement	
+5	101	0101	0101	2^5 Complement 0101
-5	x	1101	1010	1011

$\gamma=10$
 $\gamma-1$ γ (10's complement)
(9's complement)

eg. 732 find 9's complement

$$\begin{array}{r} 732 \\ - 267 \\ \hline \end{array}$$

Unsigned Representation

3 bit Number

Range : 0 to 7.

$0 \text{ to } 2^3 - 1$

($n=3$).

Binary	Decimal	Binary	Decimal
000	0	100	4
001	1	101	5
010	2	110	6
011	3	111	7

Signed Representation :

4 bit number : $\{2^{4-1} - 1\}$ to $\{2^4 - 1\}$.

Signed Number	Equivalent decimal	Signed Number	Equivalent decimal
0000	+0	1000	-0
0001	+1	1001	-1
0010	+2	1010	-2
0011	+3	1011	-3
0100	+4	1100	-4
0101	+5	1101	-5
0110	+6	1110	-6
0111	+7	1111	-7

1's Complement Representation :

4 bit number : -7 to +7.

'n' bit
 $-\{2^{n-1} - 1\}$ to $+\{2^{n-1} - 1\}$

1's Complement	Equivalent Decimal	1's Complement	Equivalent deci.
0000	+0	1000	-7
0001	+1	1001	-6
0010	+2	1010	-5
0011	+3	1011	-4
0100	+4	1100	-3
0101	+5	1101	-2
0110	+6	1110	-1
0111	+7	1111	-0

Q1. 001101 \rightarrow +13

Q2. 110010 \rightarrow -13

001101 \rightarrow +13

2's Compliment Representation.4 Bit Number: -8 to +7.

'n' bit

Range: $\{-2^{n-1}\} \text{ to } \{2^{n-1}-1\}$

<u>2's Compl.</u>	<u>Equivalence dec.</u>	<u>2's Compl.</u>	<u>Equivalence dec.</u>
0000	+0	1000	-8
0001	+1	1101	-7
0010	+2	1100	-6
0011	+3	1011	-5
0100	+4	11100	-4
0101	+5	1101	-3
0110	+6	1110	-2
0111	+7	1111	-1

Q1. Convert the number given in 1's Compliment into Equivalence decimal.

1. 01101 \rightarrow 13 5. 11110010 \rightarrow -13
 2. 001101 \rightarrow 13 6. 11101010 \rightarrow -21
 3. 00001101 \rightarrow 13 7. 0000 \rightarrow +0
 4. 110010 \rightarrow -13 8. 1111 \rightarrow -0

Q2. Convert the number given in 2's Compliment into Equivalence decimal?

1. 01101 \rightarrow +13 5. 11110010 \rightarrow -14
 2. 001101 \rightarrow +13 6. 11101010 \rightarrow -22
 3. 00001101 \rightarrow +13 7. 0000 \rightarrow 0
 4. 110010 \rightarrow -14 8. 1111 \rightarrow -1

Q3. F's Compliment of (2BFD)box is D402

FFFF

2BFD

D402

Q4. A number is expressed in binary two's Compliment as 10011. Its decimal equivalent value is -13

Q5. The greatest negative number, which can be stored in a Computer that has 8-bit word length and uses 2's Complement Arithmetic is.

$$\text{Ans} = \{2^7\} = -2^7 = \underline{-128}$$

Q6. What is the resultant of $(CH)_{16} - (FB)_{16}$?

Ans

Q7. The number of bytes required to represent the decimal number + 1856357 in Packed BCD (Binary Coded Decimal Form) is _____

Q8. 11001, 1001, and 111001 correspond to the 2's Complement representation of cubics of the following sets of numbers?

Ans -7, -7, and -7 respectively.

Q9. P, Q, and R are the decimal integers corresponding to the 4-bit binary number 1100 Considered in signed magnitude, 1's complement and 2's Complement representations respectively. The 6-bit 2's Complement representation of $(P+Q+R)$ is 110101

Ans. $P = -4$

$$Q = -3, R = -4 \quad P+Q+R = -11$$

$$\begin{array}{r} 001011 \\ \swarrow \quad \searrow \\ 110101 \end{array}$$

Q10. What is the possible base of the given number system. $\sqrt{4x+5} = 5$.

Ans $(\sqrt{4x+5})_x \times (5)_x \quad x > 5$

$$\sqrt{4x+5} = 5$$

$$4x+5 = 25$$

$$4x = 24 \quad \therefore x = \underline{\underline{6}}$$

Q11. -13 in 2's Complement can be -?

+13 - 01101

-13 - 10011.

To 2's Complement: 110011.

Q12. Let us consider the following equation in a 6-bit binary number system $X = A + BA$ is given. A is given as $(001010)_2$ in 1's Complement binary number system, B is given as $(111010)_2$ in signed number system. What would be X in 2's Complement number system?

Ans

$$A = 10 \quad X = A + BA$$

$$B = -26 \quad = 10 + (-26) \times 10$$

$$= -250$$

$$\begin{array}{r} 01111010 : 250 \\ 100000110 : -250 \\ \hline \end{array}$$

Q13. -24 in 2's Complement form is 11101000

$$\begin{array}{r} 00011000 \rightarrow +24 \\ 11101000 = -24 \end{array}$$