- **Derive output equations and simplify it if possible**
From the truth table, derive the boolean expressions for each output. Simplify the boolean expression using boolean algebra rules, K-map method etc. The goal is used to reduce the number of logic gates required, minimizing the circuit's complexity.

- **Draw logic circuit diagram**
select appropriate logic gates based on the simplified boolean expressions. Create a diagram using choosen logic gates to represent the simplified boolean expressions.

March - April 2023

Group - A

Brief Answer Questions (10 X 1 = 10)

1) What is digital system?

⇒ Digital system is an electronic system that works with discrete values represented by 0 (low or off) and 1 (high or on) and store data in the form of binary bit.

2) How many different combination can be represented by 5-bit binary data?

⇒ $2^5 = 32$.

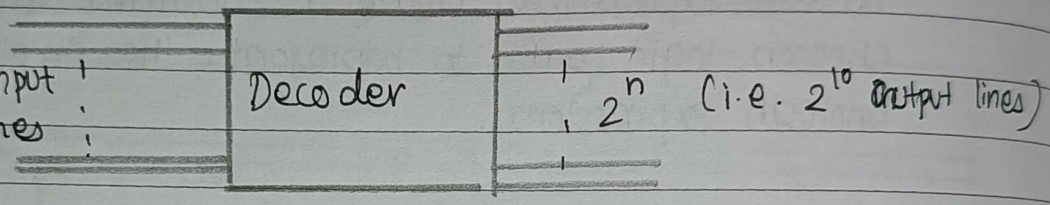
In 32 different combination, 5-bit binary data can be represented.

3) What is the use of Karnaugh Map?

⇒ The use of Karnaugh Map is to simplify the boolean expression and reading logic circuits. This leads in reducing the size and complexity of digital circuit.

4) How many input lines are required to generate 995 decoded lines?

⇒ $2^{10} \geq 995$. Therefore, 10 input lines are required to generate 995 decoded lines.



5) Differentiate between combinational and sequential circuit.

⇒

Combinational circuits	Sequential circuit
• They are time independent circuits which do not depends upon previous inputs to generate any output.	• They are dependent to clock cycles and depends on present as well as past inputs to generate any output.
• It is easy to use and handle.	• It is not easy to use and handle.

6) What is the problem of "T" flip-flop.

⇒ The T-flip-flop can only store a single bit of information and cannot perform more complex operations. It produce noise on the input signal.

7) What is the number of clock pulses required to input and output "5" bit data in parallel in serial out shift register?

⇒ The number of clock pulse required to input and output "5" bit data in parallel in serial out shift register are 6. one is to load and five to shift out.

8) Define don't care condition.

⇒ In K-Map, each minterm is represented by 1 and max-term with 0. A situation where we cannot differentiate 1 and 0, such situation is called don't care condition.

9) How many don't care conditions are possible in 8x3 encoder?

⇒ $2^8 - 8 = 248$ don't care conditions are possible in 8x3 encoder.

10) Why flip flop is said to be 1 bit memory?

⇒ Flip-flop is said to be 1 bit memory because it can store a single binary digit (0 or 1) and retains that data until it is changed by a control signal.

Group-B

Short answer questions (Any Five) (5x3 = 15)

11) Convert $(2A3B.2A)_{16}$ into binary, octal and decimal.

⇒

· Converting to Binary

$(2A3B.2A)_{16} \rightarrow (?)_8$

⇒ ~~(001010~~

⇒ $(0010\,1010\,0011\,1011.0010\,1010)_2$

· Converting to octal

$(0010\,1010\,0011\,1011.0010\,1010)_2$

⇒ $(025073.124)_8$

· Converting to Decimal

$(2A3B.2A)_{16} \rightarrow (?)_{10}$

$= 2\times16^3 + A\times16^2 + 3\times16^1 + B\times16^0 + 2\times16^{-1} + A\times16^{-2}$

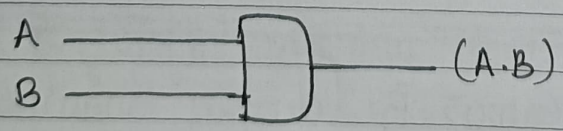
$= (10811.16486)_{10}$

---

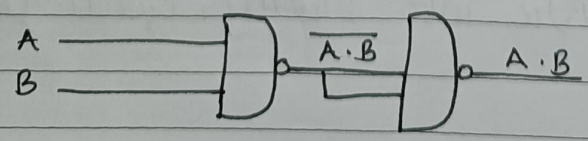
12) Realize the property of AND, OR and NOT gate using NAND gate.

⇒

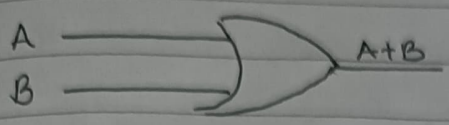
· AND Gate



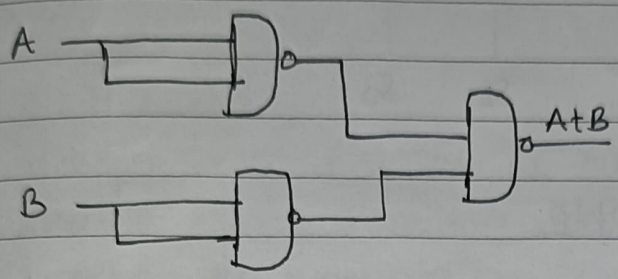
NAND Gate



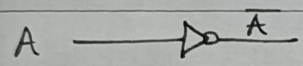
- **OR Gate**



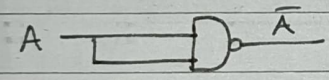
NAND Gate



- **NOT Gate**



NAND Gate

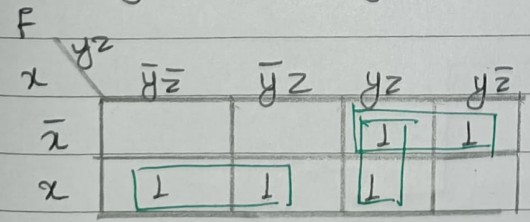


---

13) If a Boolean functions $F = \bar{x}y + x\bar{y} + xyz$, Solve the given functions using K-map.

⇒

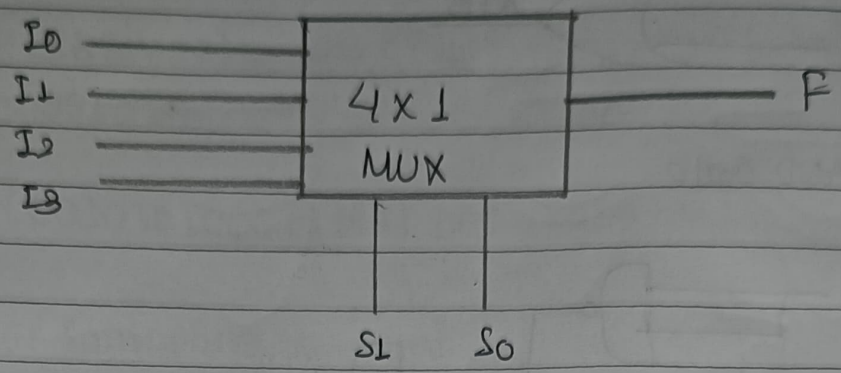
$F = \bar{x}y + x\bar{y} + xyz$



$\therefore F = \bar{x}y + x\bar{y} + yz$

14) Design 4x1 multiplexer.

⇒

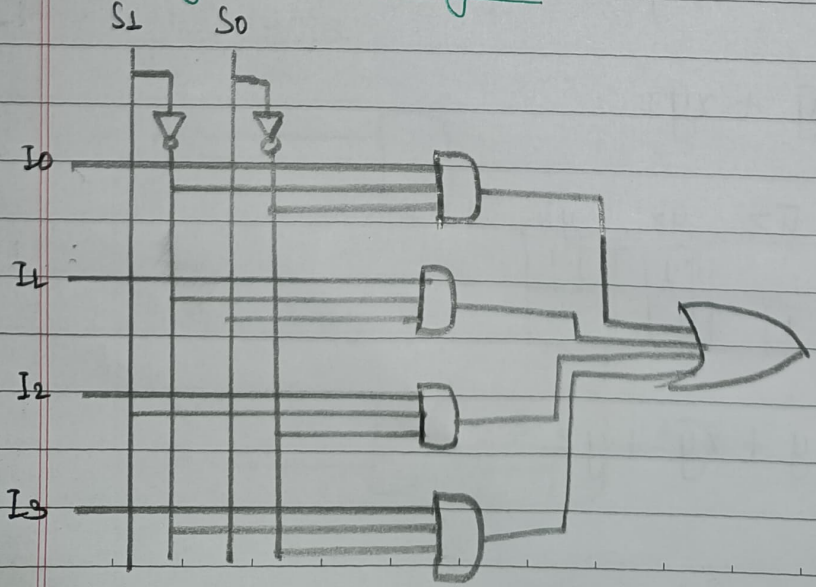


Operational Table

Selection		Output
S_1	S_0	F
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

$$F = I_0 \bar{S_1} \bar{S_0} + I_1 \bar{S_1} S_0 + I_2 S_1 \bar{S_0} + I_3 S_1 S_0$$

Logic Circuit Diagram



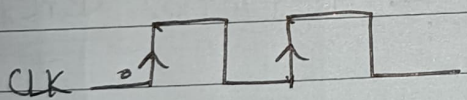
15) Explain triggering of a flip-flop with its types.

⇒ The state of flip-flop is changed by a momentary change in the input signal. This change is called a trigger. Triggering is the way of activating a flip-flop so that it can respond to the inputs and can update its output state as per the input data. Its types are:

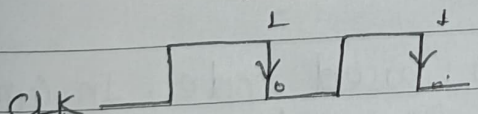
- **Edge-Triggered**

A flip flop that changes its output state only at a moment of a transition from one logic level to another. Its types are:

i) **Positive edge triggered:**
It changes state on the rising edge.
For example:



ii) **Negative edge triggered:**
It changes state on the falling edge.
For example:



16) If a Boolean function $F = AB + \bar{B}C$, find a equivalent POS.

⇒

$$F = AB + \bar{B}C$$
$$\bar{F} = \overline{AB + \bar{B}C}$$
$$\bar{F} = \overline{AB} \cdot B\bar{C} \quad \text{[using De-morgan's theorem]}$$
$$\bar{F} = \bar{A} + \bar{B} \cdot B + \bar{C}$$
$$\therefore \bar{F} = (\bar{A} + \bar{B}) \cdot (B + \bar{C})$$

Finding SOP from F'.

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

∴ The equivalent POS from boolean function ~~$F = Ab + b'C$ is~~ $F = AB + \bar{B}C$ is $(\bar{A} + \bar{B}) \cdot (B + \bar{C})$

Group-C

Long Answer Questions (Any three) (3x5 = 15)

17) What is gray code? Design 3-bit synchronous counter.

⇒ Gray code is a transition based code. In Gray code, all one bit is interchanged during the transition in a sequence from previous transition.

3-bit Synchronous counter

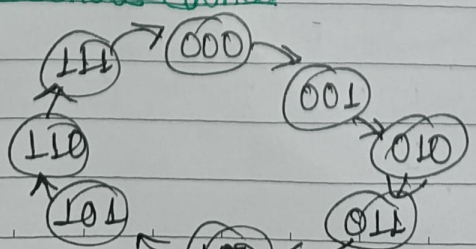
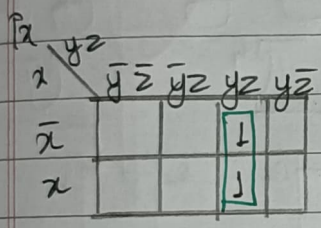


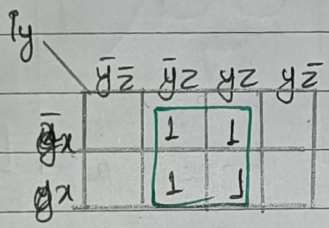
Fig: State Diagram

State Table

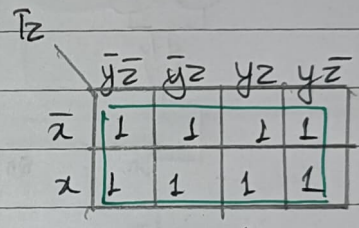
Present state			Next state			Flip-Flop input		
x	y	z	x	y	z	Tx	Ty	Tz
0	0	0	0	0	1	0	0	1
0	0	1	0	1	0	0	1	1
0	1	0	0	1	1	0	0	1
0	1	1	1	0	0	1	1	1
1	0	0	1	0	1	0	0	1
1	0	1	1	1	0	0	1	1
1	1	0	1	1	1	0	0	1
1	1	1	0	0	0	1	1	1



$\therefore Tx = yz$

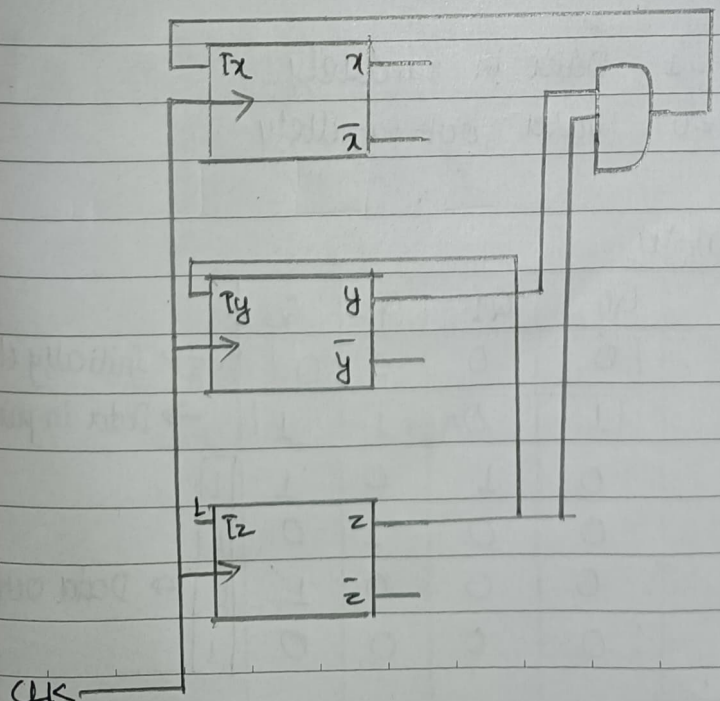


$\therefore Ty = z$



$\therefore Tz = 1$

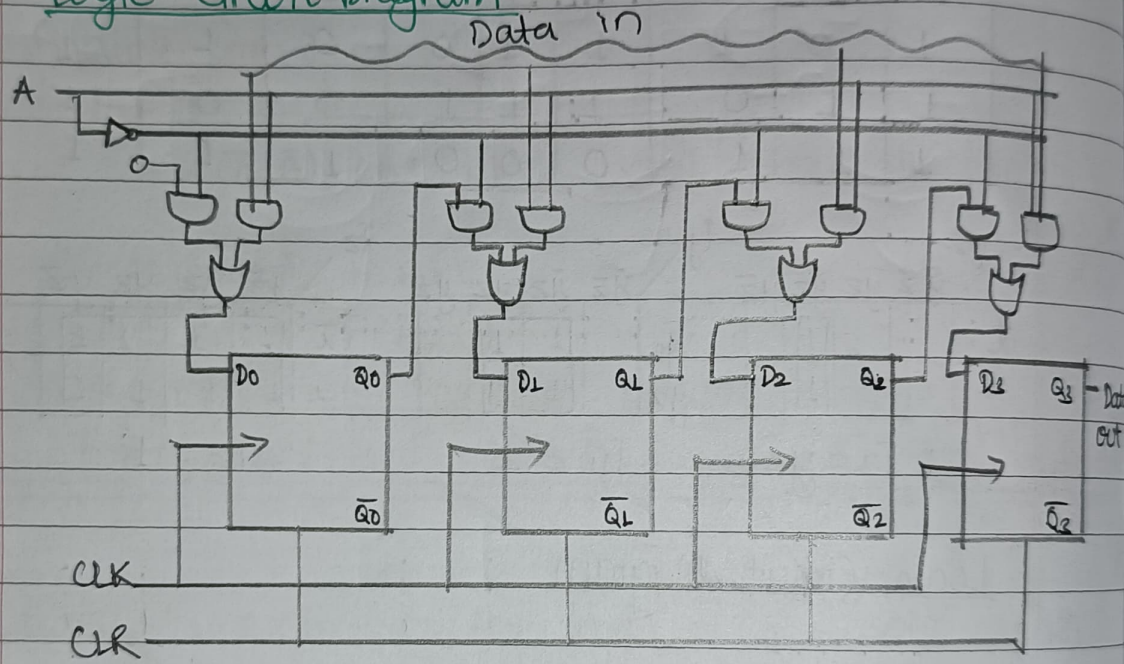
Logic Circuit Diagram



18) You are provided with data bits 1011. The register provided has the capability to store all bits in one clock pulse and can be retrieved those stored bits only one at a time from right most flip flop. Your work is to draw the circuit and show all necessary steps to store and retriere the provided data in provided register.

⇒

Logic Circuit Diagram



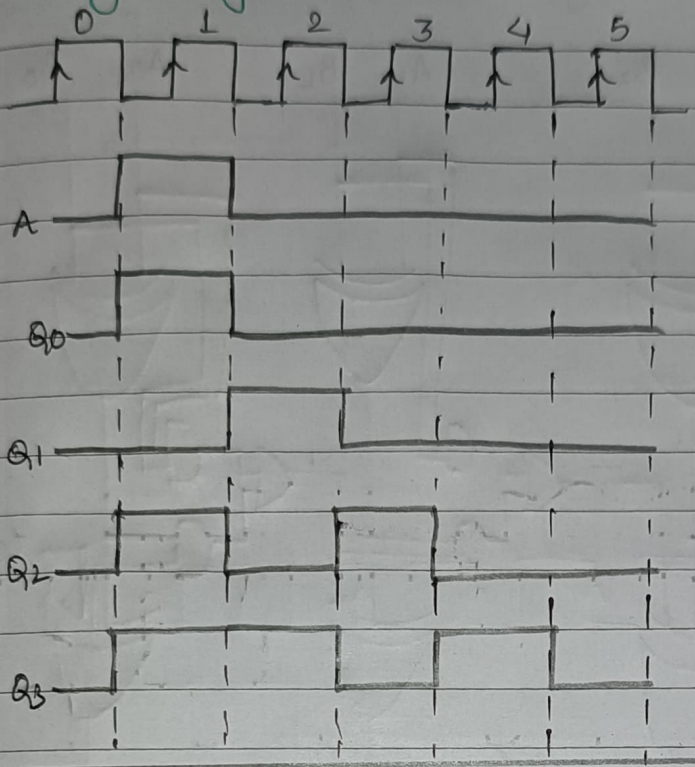
A = 1 Data in parallely

A = 0 Data out parallely

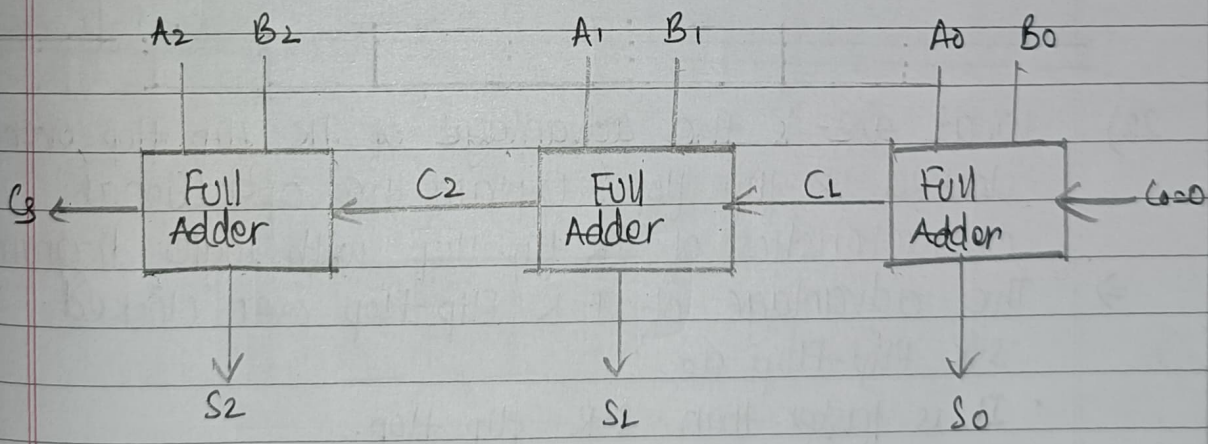
Operational Table

Clock	A	Q0	Q1	Q2	Q3	
0	X	0	0	0	0	→ Initially clear state
1	1	1	0	1	1	→ Data in parallely
1	0	0	1	0	1	1
1	0	0	0	1	0	1
1	0	0	0	0	1	0 → Data out serially
1	0	0	0	0	0	1
0	0					

Timing Diagram



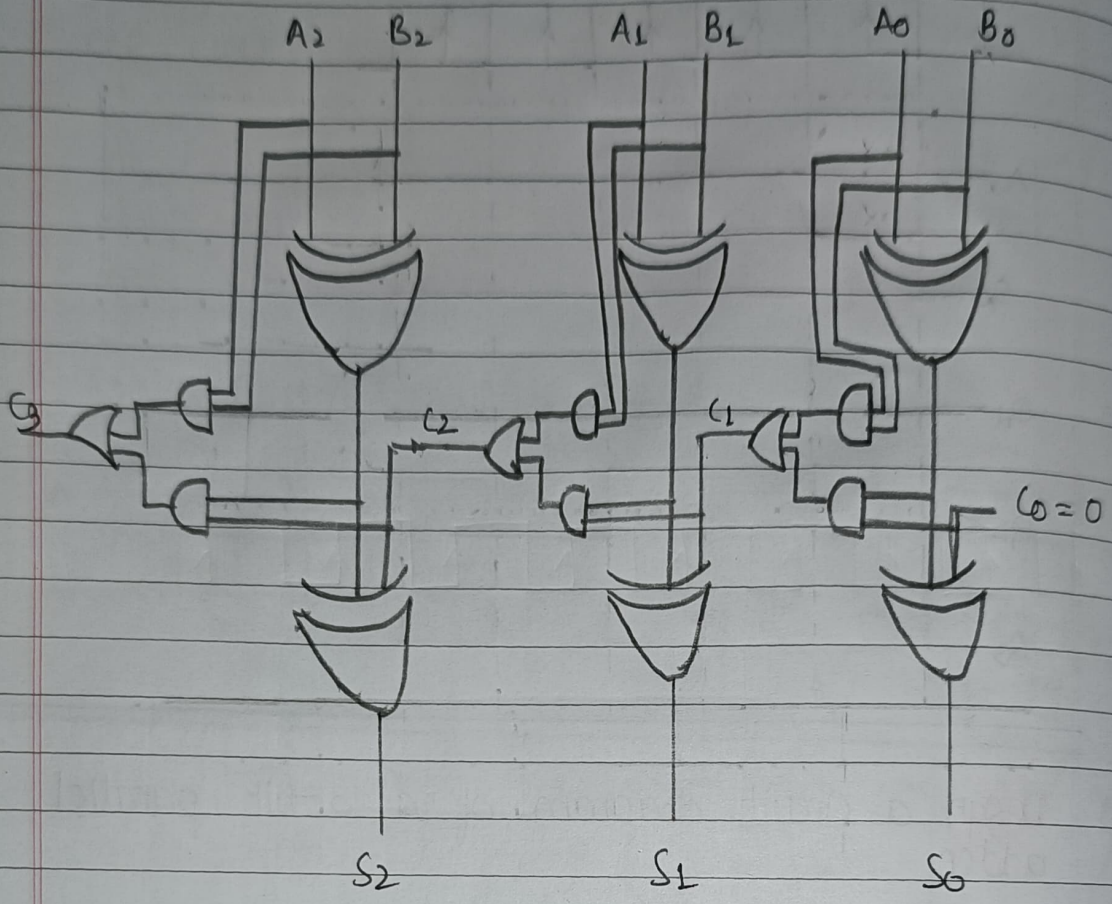
19) Design a circuit diagram of a 3-bit parallel adder.



$S = A \oplus B \oplus C$

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

Logic Circuit Diagram

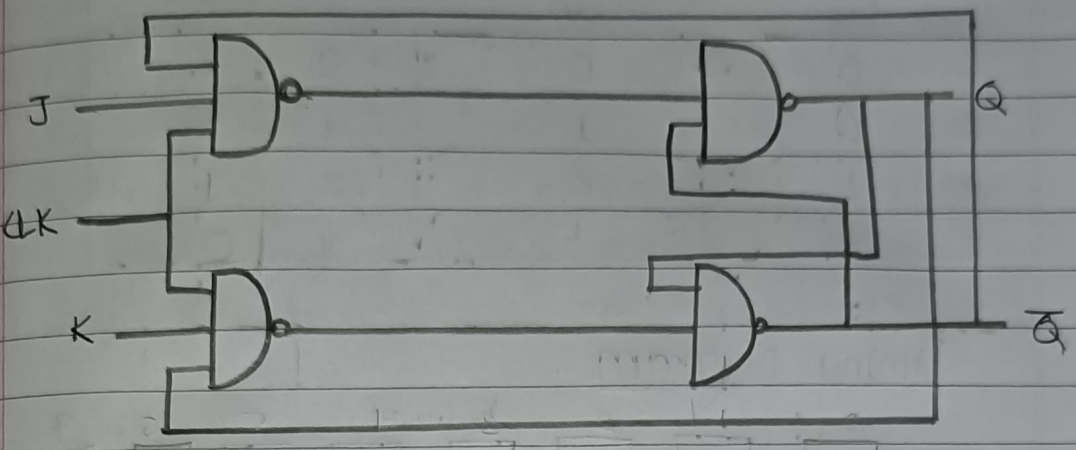


20) What ~~are~~ is the advantage of JK flip flop over clocked SR flip flop? Explain the operational characteristics of JK flip flop with logic diagram.

⇒ The advantage of J-K flip-flop over clocked S-R flip-flop are:

- It is faster than S-R flip flop.
- JK flip-flop has a feedback path.
- JK flip-flop doesn't have indeterminate state. (i.e. it works when both input are one)

Logic Circuit Diagram



Characteristics Table

Clock	J	K	Q_p	Q_n
0	X	X	X	No change
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

Characteristics Equation

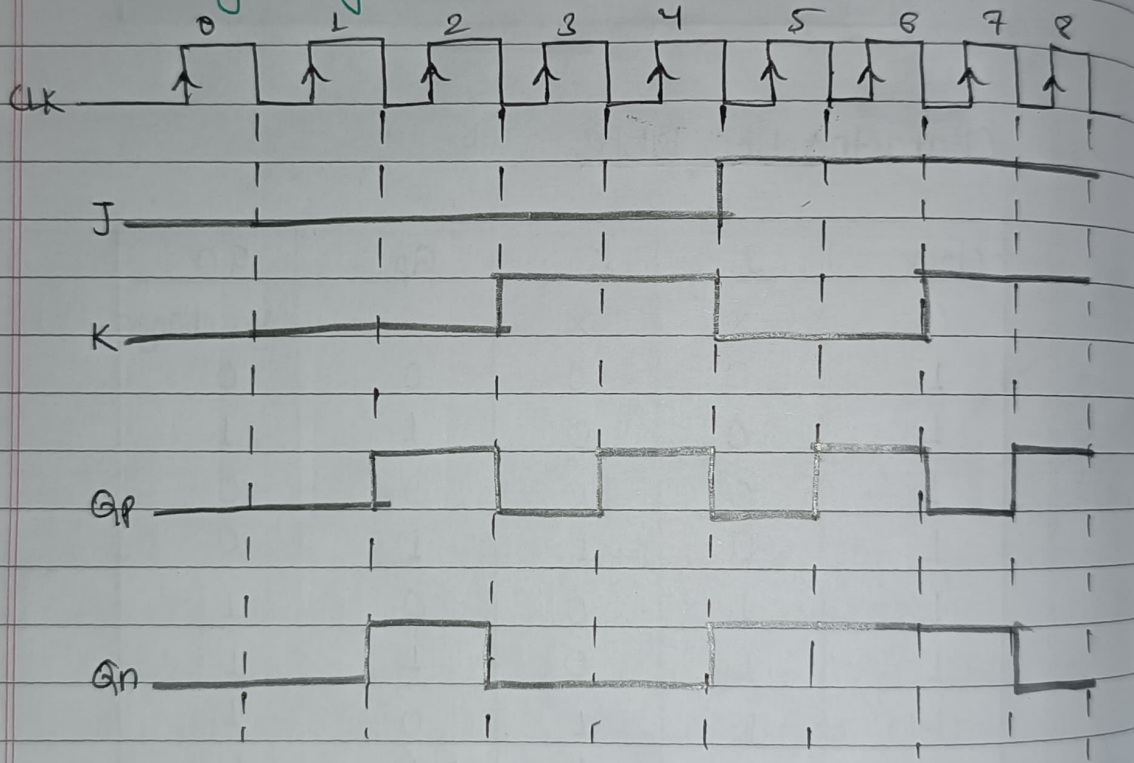
Q_n / JK / Q_p	$\bar{K}\bar{Q}_p$	$\bar{K}Q_p$	KQ_p	$\bar{K}Q_p$
J		1		
$\bar{J}$	1	1		1

$$\therefore Q_n = J\bar{Q}_p + \bar{K}Q_p$$

Excitation Table

Present state	Next state	FF input	
Q_P	Q_n	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

Timing Diagram



Group-D

Comprehensive Answer (2×10 = 20)

21) What is Boolean algebra? List any five Boolean rules and verify any two Boolean rules using truth table method.

⇒ Boolean algebra is a mathematical tool for designing and analyzing digital circuits. Some five Boolean rules are:

- **Commutative Law**

  $A+B = B+A$

  $A \cdot B = B \cdot A$

- **Associative Law**

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

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

- **Distributive Law**

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

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

- **De Morgan's Law**

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

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

- **Identity Law**

  $A \cdot 1 = A$

  $A+0 = A$

De-Morgan's Law

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

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

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

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

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

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

Distributive Law

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

A	B	C	B.C	A+B	A+C	A+(B.C)	(A+B).(A+C)
0	0	0	0	0	0	0	0
0	0	1	0	0	1	0	0
0	1	0	0	1	0	0	0
0	1	1	1	1	1	1	1
1	0	0	0	1	1	1	1
1	0	1	0	1	1	1	1
1	1	0	0	1	1	1	1
1	1	1	1	1	1	1	1

$\therefore A+(B.C) = (A+B).(A+C)$

② $A(B+C) = (A.B) + (A.C)$

A	B	C	B+C	A.B	A.C	A(B+C)	(A.B)+(A.C)
0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0
0	1	0	1	0	0	0	0
0	1	1	1	0	0	0	0
1	0	0	0	0	0	0	0
1	0	1	1	0	1	1	1
1	1	0	1	1	0	1	1
1	1	1	1	1	1	1	1

$\therefore A(B+C) = (A.B) + (A.C)$

22) Differentiate between PLA and ROM. Design MOD-150 asynchronous counter.

⇒ The difference between PLA and ROM are as follows:

	Basis	PLA	ROM
1)	Full form	Programmable Logic Array.	Read only Memory
2)	Functionality	Implements logic functions using programmable AND and OR gates.	Stores data and doesn't perform logic function.
3)	Configurability	Both AND and OR arrays are configurable.	Only the OR gates array is configurable.
4)	Flexibility	Can accommodate future logic changes due to its reprogrammability.	Cannot be updated, modified ~~are~~ or replaced unless, specifically required to update data.
5)	Cost and manufacturing	Manufacturing costs could increase due to initial programming.	No initial programming required, potentially lower cost.
6)	Usability	It is more convenient to use.	It is less convenient as compared to PLA where don't care condition is excessive.

~~7) Block Diagram~~

MOD-150 asynchronous counter

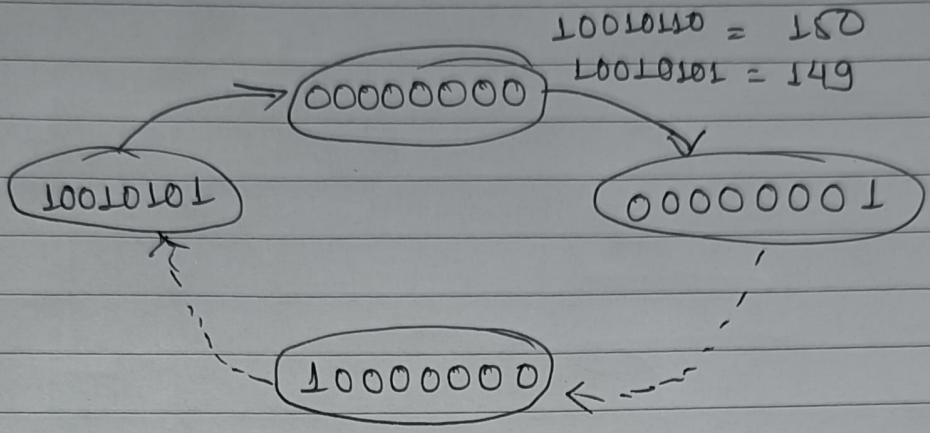
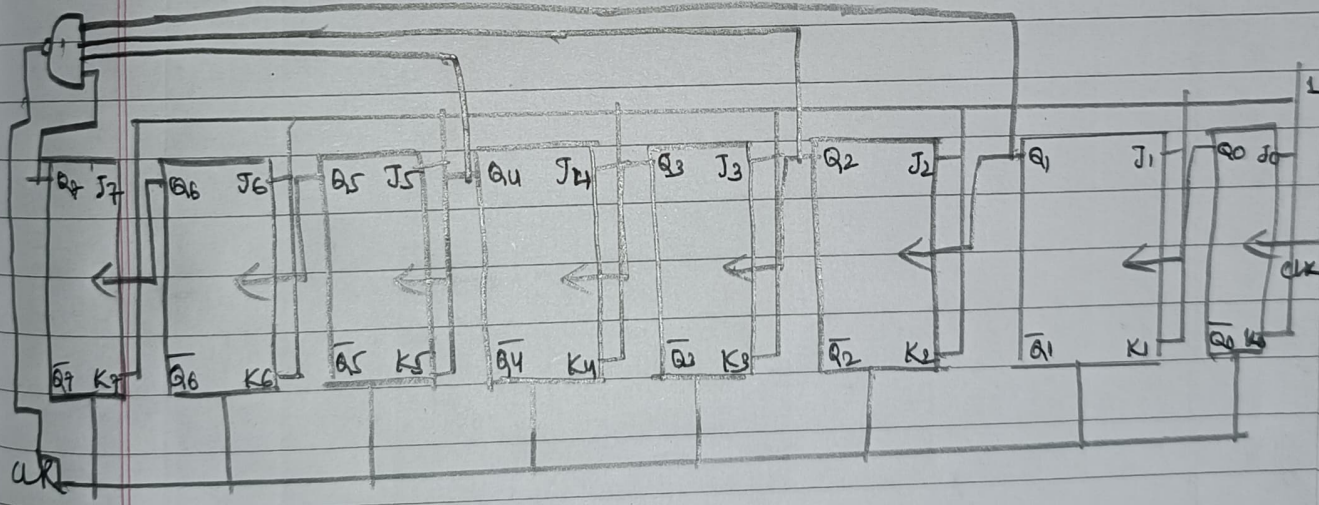


Fig: State Diagram

Logic Circuit Diagram



Operational Table

Clock	Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
149	1	0	0	1	0	1	0	1
150	0	0	0	0	0	0	0	0