CSE-Digital Electronics ES-207A

Rajeev Kumar

AP ECE Department (PIET)

Research Domain: Soft Computing, Li-Fi

AI, ML & DL

IT-Digital Electronics and Logic Design
ES-217S

Rajeev Kumar

AP ECE Department (PIET)

Research Domain: Soft Computing, Li-Fi
AI, ML & DL

Course Outcome (CSE)

CO1	Explain the concept of Boolean algebra and logic gates.
CO2	Illustrate the realisation of Boolean expressions using logic gates.
CO3	Apply the design procedure to construct basic combinational circuits.
CO4	Analysis of synchronous and asynchronous sequential circuits using flip flops.
CO5	Interpretation of various types of memories with their operations.
CO6	Discuss the various programmable logic devices.

Syllabus (CSE)

UNIT 1 MINIMIZATION TECHNIQUES AND LOGIC GATES

- Binary Digits, Logic Levels, and Digital Waveforms, Logic Systems-Positive and negative, Logic Operations, Logical Operators, Logic Gates-AND, OR, NOT, NAND, NOR, Exclusive-OR and Exclusive-NOR, Active high and Active low concepts, Universal Gates and realization of other gates using universal gates, Gate Performance Characteristics and Parameters. Boolean Algebra: Rules and laws of Boolean algebra, Demorgan's Theorems, Boolean Expressions and Truth Tables, Standard SOP and POS forms; Minterm and Maxterms, Canaonical representation of Boolean expressions, Duality Theorem, Simplification of Boolean Expressions, Minimization Techniques for Boolean Expressions using Karnaugh Map and Quine McCluskey Tabular method. Introduction of TTL and CMOS Logic and their characteristics, Tristate gates.

UNIT 2 COMBIMATIONAL CIRCUITS

- Introduction to combinational Circuits, Adders-Half-Adder and Full-Adder, Subtractors- Half and Full Subtractor; Parallel adder and Subtractor; Look-Ahead Carry Adders. BCD adder, BCD subtractor, Parity Checker/Generator, Multiplexer, Demultiplexer, Encoder, Priority Encoder; Decoder ,BCD to Seven segment Display Decoder/Driver, LCD Display, and Comparators.
-

Cont…

UNIT 3 SEQUENTIAL CIRCUITS

- **Introduction to Sequential Circuits**, Flip-Flops: Types of Flip Flops -RS, T, D, JK; Edge triggering, Level Triggering; Flip Flop conversions; Master-Salve JK.
- Introduction to shift registers, Basic Shift Register Operations, types of shift registers, Bidirectional Shift Registers, Shift Register Counters. Introduction to counters, Types of Counters-Asynchronous and synchronous counters, Up/Down Synchronous Counters, Modulo-n Counter, State table, excitation table concepts, Design of asynchronous and synchronous counters, Ring Counter, Applications of counters.

UNIT 4 CONVERTER and MEMORY DEVICES

- **Digital to Analog Converter**, Weighed Register: R-2R Ladder Network: **Analog to Digital Conversion**, Successive Approximation Type, Dual Slope Type.
- **Classification of memories** - ROM: ROM organization, PROM, EPROM, EEPROM, EAPROM, RAM: - RAM organization - Write operation, Read operation, Memory cycle, Timing wave forms, memory expansion, Static RAM Cell, MOSFET RAM cell structure, Dynamic RAM cell structure, Programmable Logic Devices - Programmable Logic Array (PLA), Programmable Array Logic (PAL), Implementation of PLA, PAL using ROM.

Books

Suggested Books

- R. P. Jain , “Modem Digital Electronics (Edition III)”; TMH
- Anand Kumar , “Fundamentals of digital circuits” ; PHI
- Malvino & Leach, “Digital Principles and Applications”, McGraw Hill.
- Thomas L. Floyd, “Digital Fundamentals”, Pearson Education Inc,
-
- **Note: The Examiner will be given the question paper template and will have to set the question paper according to the template provided along with the syllabus.**

DE Lab (CSE)

ES- 209AL

LIST OF EXPERIMENTS

- Familiarization with Digital Trainer Kit and associated equipment.
- Study of TTL gates AND, OR, NOT, NAND, NOR, EX-OR, EX-NOR.
- Design and realize a given function using K-Maps and verify its performance.
- To verify the operation of Multiplexer and De-multiplexer.
- Universal Gate
- HA and FA
- To verify the operation of Comparator.
- To verify the truth table of S-R, J-K, T, D Flip-flops.
- To verify the operation of Bi-directional shift register.
- To design and verify the operation of 3-bit asynchronous counter.
- To design and verify the operation of asynchronous Up/down counter using J-K FFs.
- To design and verify the operation of asynchronous Decade counter.
- Study of TTL logic family characteristics.
- Study of Encoder and Decoder.
- Study of BCD to 7 segment Decoder.

Course Outcome (IT)

ES-217A	Digital Electronics and Logic Design					
Lecture	Tutorial	Practical	Major Test	Minor Test	Total	Time
3	0	0	75	25	100	3 Hrs.
Course Outcomes (CO)						
ES-217A.1	Apply Boolean algebra to understand binary logic and logic circuits.					
ES-217A.2	Apply the design procedure to construct basic combinational circuits and verify their functionalities.					
ES-217A.3	Design and analysis of synchronous and asynchronous sequential circuits using flip flops.					
ES-217A.4	Classify various types of converters with their operations.					
ES-217A.5	Implement combinational logic circuits using various programmable logic devices.					

Syllabus (IT)

UNIT 1 MINIMIZATION TECHNIQUES AND LOGIC GATES

- **Fundamentals of Digital Techniques:** Review of logic gates and number system; 1's and 2's compliment Arithmetic; Introduction to Boolean algebra using basic postulates and theorems; **Binary codes: BCD, Excess-3, Gray codes**; Standard representation of logic functions: SOP and POS forms; Simplification of switching functions using K map and Quine-McCluskey methods.

UNIT 2 COMBIMATIONAL CIRCUITS

- **Design of Combinational circuits:** Half and Full Adders; Half and Full Subtractors; Multiplexers and Demultiplexers / Decoders; Implementation of SOP logic functions using multiplexers and Demultiplexers / Decoders; Encoders. Decoders / Drivers for display devices, code converters.

Cont…(IT)

UNIT 3 SEQUENTIAL CIRCUITS

- **Sequential circuits:** Latches, Flip Flops: S-R- J-K. T, D, master-slave, edge triggered flip flop; Race around condition; Excitation table; Interconversion among flip flop, Design of Synchronous and Asynchronous counters; Modulo N counter design; Shift registers.

UNIT 4 CONVERTER and MEMORY DEVICES

- **A/D and D/A converters:** Sample and hold circuit, Quantization , weighted resistor and R -2 R ladder Digital to Analog Converters, Specifications for D/A converters., Flash type Analog to digital Converter; Successive approximation type Analog to digital Converter, specifications of ADCs.
- **Programmable Logic Devices:** Introduction to PLA and PAL, Implementation of simple functions using PLA and PAL.

DE Lab-ES- 213LA (IT)

LIST OF EXPERIMENTS

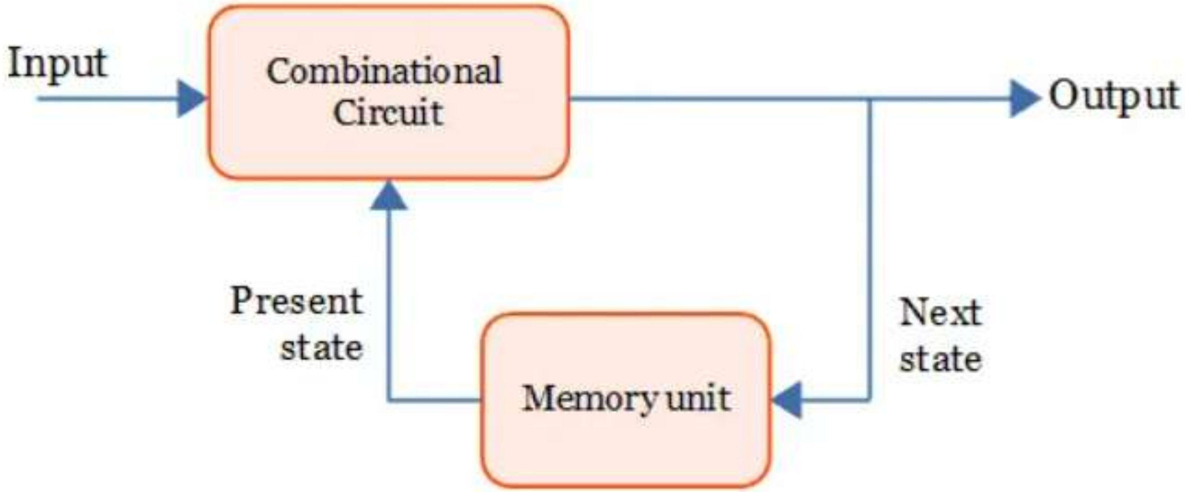
- Study of TTL gates – AND, OR, NOT, NAND, NOR, EX-OR, EX-NOR.
- Design & realize a given function using K-maps and verify its performance.
- To verify the operation of multiplexer & Demultiplexer.
- To verify the operation of comparator.
- To verify the truth tables of S-R, J-K, T & D type flip flops.
- To verify the operation of bi-directional shift register.
- To design & verify the operation of 3-bit synchronous counter.
- To design and verify the operation of synchronous UP/DOWN decade counter using J K flipflops & drive a seven-segment display using the same.
- To design and verify the operation of asynchronous UP/DOWN decade counter using J
- K flipflops & drive a seven-segment display using the same.
- To design & realize a sequence generator for a given sequence using J-K flip-flops.
- Study of CMOS NAND & NOR gates and interfacing between TTL and CMOS gates.
- Design a 4-bit shift-register and verify its operation.

Note: A student has to perform at least ten experiments. Seven experiments should be performed from the above list. Remaining three experiments may either be performed from the above list or designed & set by the concerned institution as per the scope of the syllabus

Introduction to Sequential Circuits

- In a sequential circuit, the output produced depends not only on the applied input but also on the past history of the outputs.

- The sequential circuit consists of a combinational logic, which gets the external input and the previous state output through the inbuilt memory unit as feedback. The memory unit may be a latch or a flip flop.



- The present state and the external inputs determine the outputs and the next state of the sequential circuit. So, a sequential circuit works on a time sequence of external inputs, internal states and outputs.

Digital Circuits

Combinational Circuits

1. In this output depends only upon present input.
2. Speed is fast.
3. It is designed easy.
4. There is no feedback between input and output.
5. This is time independent.
6. Elementary building blocks: Logic gates
7. Used for arithmetic as well as Boolean operations.
8. Combinational circuits don’t have capability to store any state.
9. As combinational circuits don’t have clock, they don’t require triggering.
10. These circuits do not have any memory element.
11. It is easy to use and handle.
12. **Example –** Encoder, Decoder, Multiplexer, De-multiplexer

Sequential Circuits

1. In this output depends upon present as well as past input.
2. Speed is slow.
3. It is designed tough as compared to combinational circuits.
4. There exists a feedback path between input and output.
5. This is time dependent.
6. Elementary building blocks: Flip-flops
7. Mainly used for storing data.
8. Sequential circuits have capability to store any state or to retain earlier state.
9. As sequential circuits are clock dependent they need triggering.
10. These circuits have memory element.
11. It is not easy to use and handle.
12. **Examples –** Flip-flops, Counters

Types of Sequential Circuits

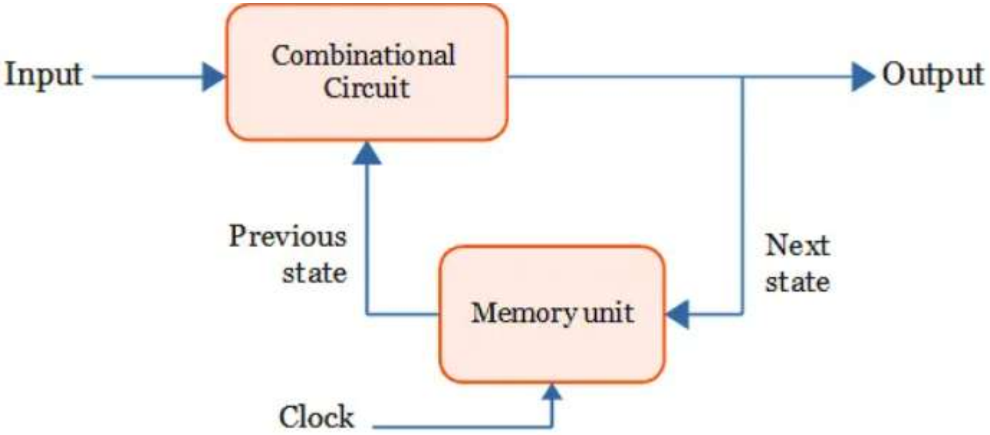
- The sequential circuits are classified based on the clock pulses given to the memory units. There are two types as follows

  Synchronous sequential circuit

  Asynchronous sequential circuit

Synchronous sequential circuit

- In these circuits, change in input can affect the memory elements only upon the activation of the clock pulse. The memory units are clocked flip-flops.
- The circuit will change its state for every clock pulse. Hence it is also called clocked sequential circuits.

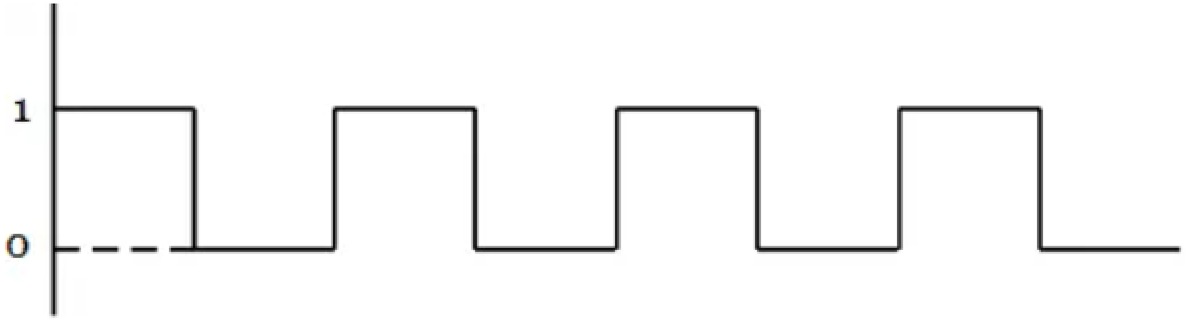


Asynchronous sequential circuit

- The change in input signals can affect the memory element at any instant of time. The memory units do not have clock pulses. Instead, they have time delay elements.

Clock Pulse

- A clock pulse is a continuously changing signal that oscillates between a high state and a low state. The common type of clock signal is a square type, which has 50% duty cycle with a fixed and constant frequency. A clock pulse is shown below.



- While applying the clock pulse to the flip flop, it gets triggered by two ways, Level triggering and edge triggering.

Level triggering

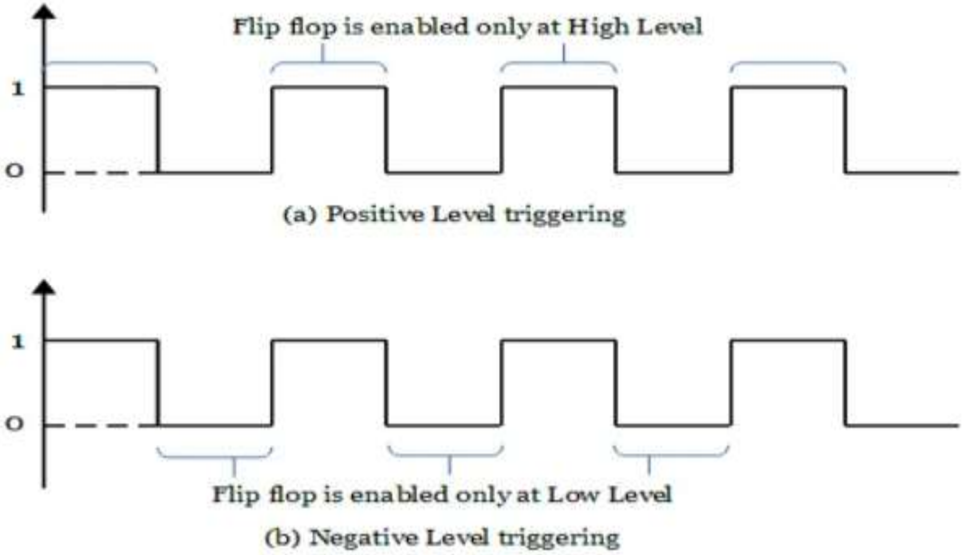
Edge Triggering

Level triggering

Level triggering (LATCH)

- In this, the flip flop is triggered only during the high-level or the low level of the clock pulse. In other words, the output changes its state, when active low or high level is maintained at the clock signal. Based on the level of triggering, it is of two types

- **Positive level triggering** – If the flip flop is triggered at the positive level of the clock pulse, then it is said to be a positive level triggering.
- **Negative level triggering** – If the flip flop is triggered at the negative level of the clock pulse, then it is said to be negative level triggering.


Flip flop is enabled only at High Level
1
0
1
0
Flip flop is enabled only at Low Level

(a) Positive Level triggering

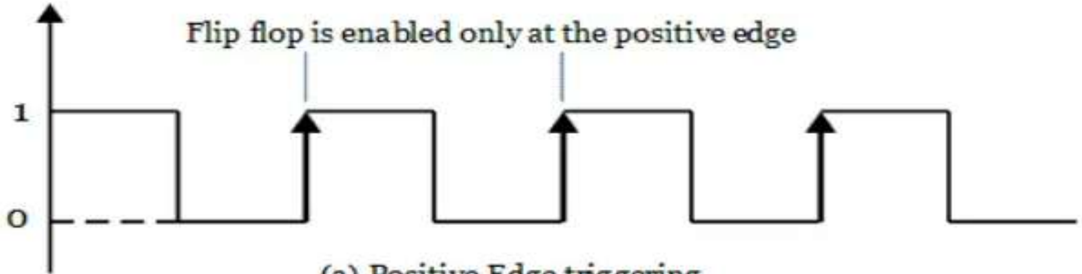
(b) Negative Level triggering

Edge Triggering

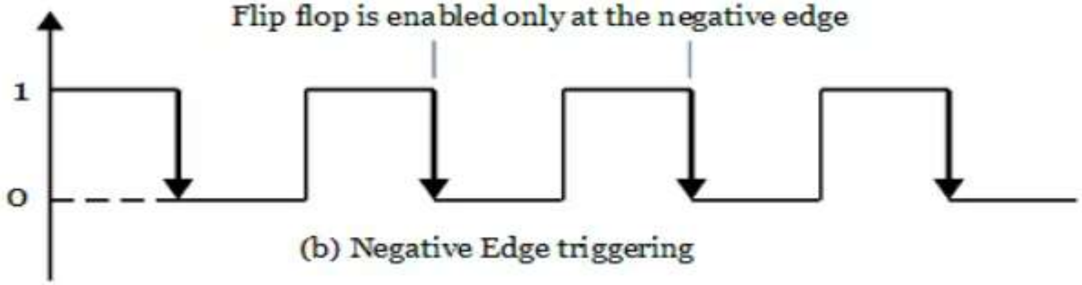
Edge triggering (FLIP FLPO)

- In edge triggering, the flip flop changes its state during the positive edge or negative edge of the clock pulse. There are two types of edge triggering.

- **Positive edge triggering** – When the output responds to the change in the input only at the positive edge of the clock pulse, then the clock pulse is said to be a positive edge triggered.
- **Negative edge triggering** – When the output responds to the change in the input only at the negative edge of the clock pulse, then the clock pulse is said to be a negative edge triggered.



(a) Positive Edge triggering



(b) Negative Edge triggering

Important Point

NOTE:

Level Triggering (LATCH)

- In **Level Triggering circuit i**f input is constant, clock pulse applied then output is constant
- If input change multiple time in single clock then output also changes accordingly multiple time in single clock

Latch vs Flip Flop

FLIP FLOP

1. Flip-flop is a bistable device i.e., it has two stable states that are represented as 0 and 1.
2. It checks the inputs but changes the output only at times defined by the clock signal or any other control signal.
3. It is a edge triggered device.
4. Gates like NOR, NOT, AND, NAND are building blocks of flip flops.
5. They are classified into asynchronous or synchronous flipflops
6. It forms the building blocks of many sequential circuits like counters.
7. A Flip-flop always have a clock signal
8. Flip-flop can be build from Latches
9. **Example:** D Flip-flop, JK Flip-flop

LATCH

1. Latch is also a bistable device whose states are also represented as 0 and 1.
2. It checks the inputs continuously and responds to the changes in inputs immediately.
3. It is a level triggered device.
4. These are also made up of gates.
5. There is no such classification in latches.
6. These can be used for the designing of sequential circuits but are not generally preferred.
7. Latches doesn’t have a clock signal
8. Latches can be build from gates
9. **Example:** SR Latch, D Latch

Important Point

Note:

1. In Flip Flop first apply data then clock
2. **Set up Time:** is minimum to keep input data ready before applying clock
3. **Hold Time:** is minimum to keep same data after applying clock to store data properly

$$\mathbf{t_{set\ up} > t_{hold}}$$

Flip- Flop & Types

Flip Flop:

- Basic memory element
- Used to store 1 bit
- FF have two output which are complement to each other
- FF have two stable states hence it is known as Bistable Multivibrator
- Also used in frequency divider circuit

Flip Flop Types

- **SR FF**
- **JK FF**
- **D FF**
- **T FF**
- **Master Slave FF**

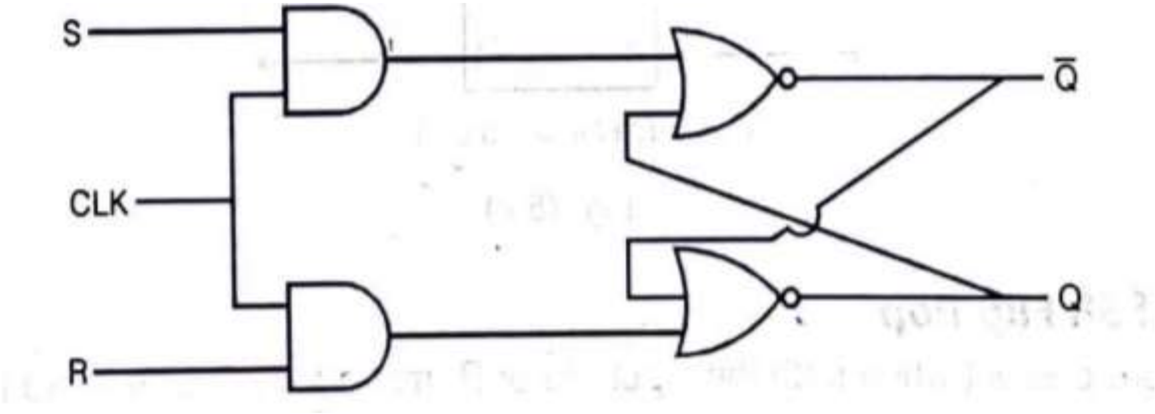
Flip- Flop

Flip Flop:

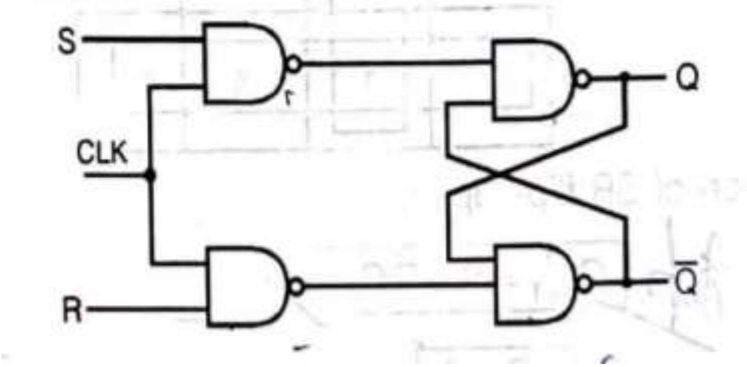
- Logic Circuit
- Truth Table
- Characteristics Table
- Characteristics Equation
- Excitation Table
- State Diagram

SR Flip-Flop

Clocked SR FF using NOR Latch

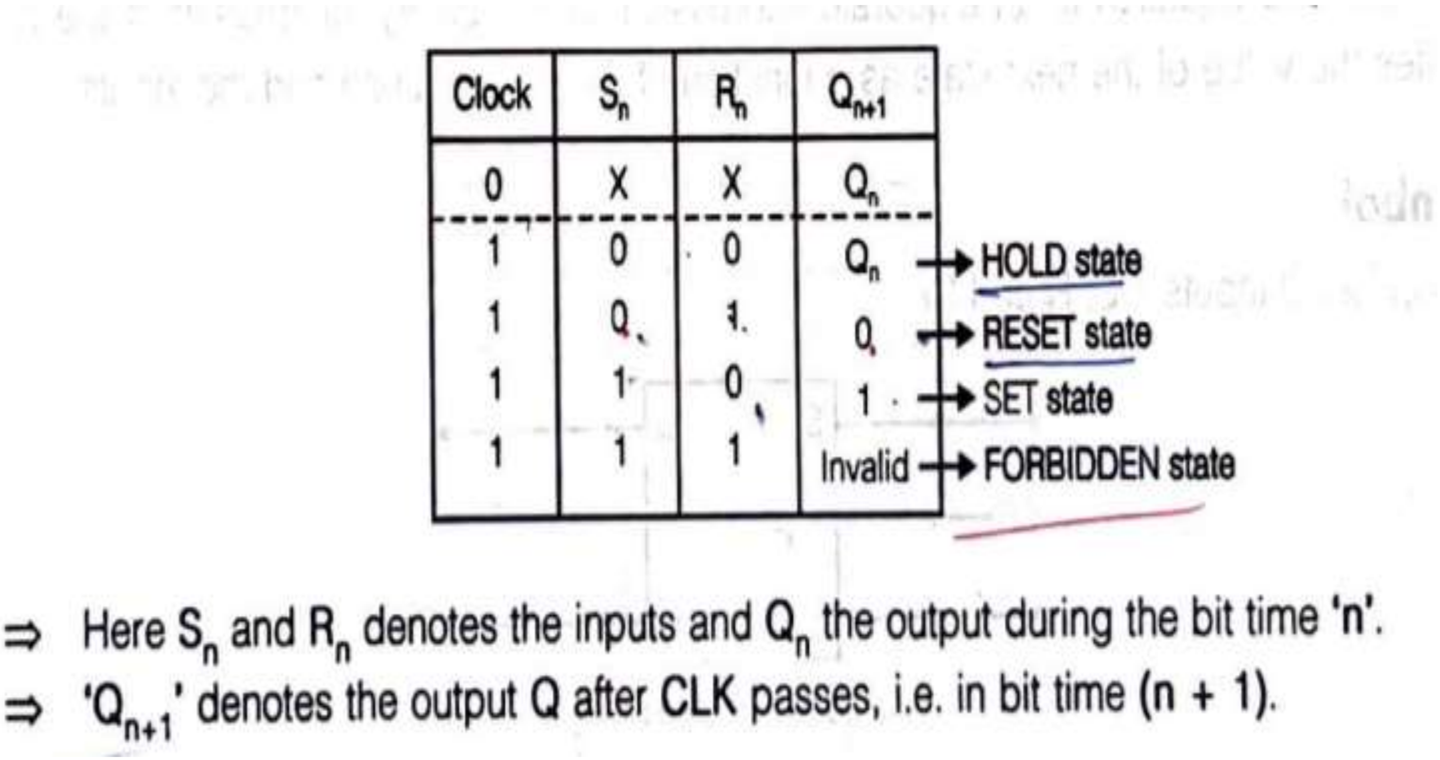


Clocked SR FF using NAND Latch



SR Flip-Flop

SR FF Truth Table



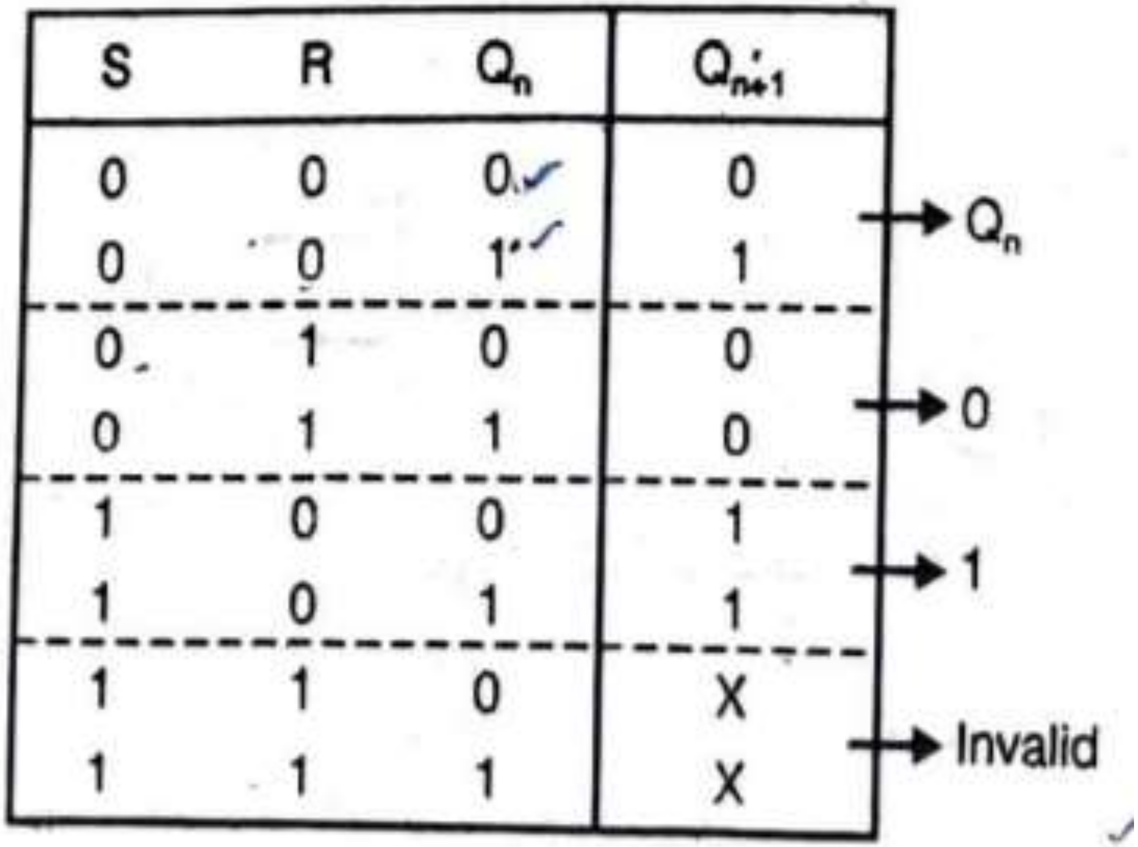
Clock	S_n	R_n	Q_{n+1}	
0	X	X	Q_n	
1	0	0	Q_n	→ HOLD state
1	0	1	0	→ RESET state
1	1	0	1	→ SET state
1	1	1	Invalid	→ FORBIDDEN state

⇒ Here S_n and R_n denotes the inputs and Q_n the output during the bit time **'n'**.

⇒ 'Q_{n+1}' denotes the output Q after CLK passes, i.e. in bit time (n + 1).

SR Flip- Flop

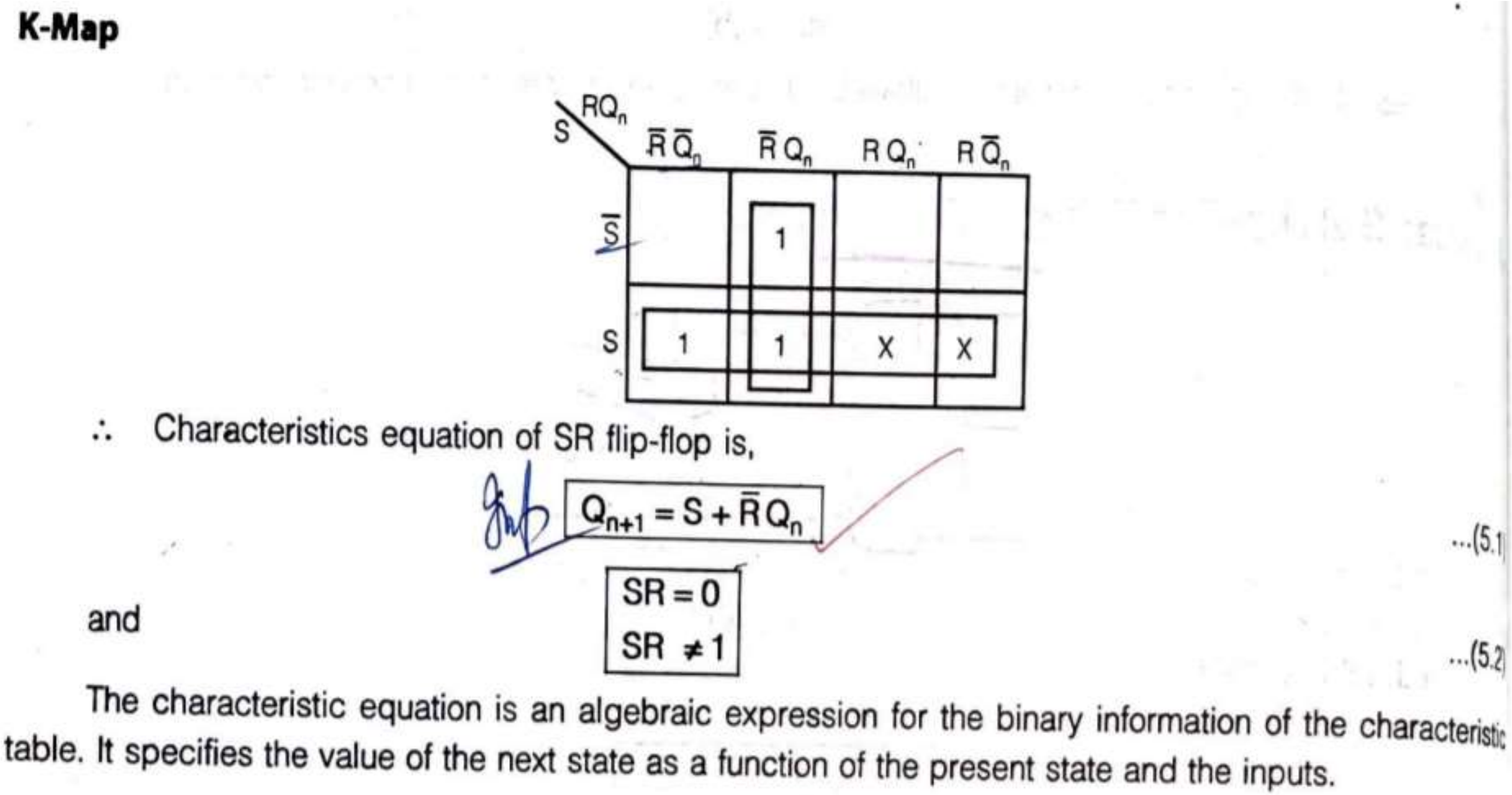
Characteristics Table of SR FF



S	R	Q_n	Q_{n+1}	
0	0	0 ✓	0	→ Q_n
0	0	1 ✓	1	
0	1	0	0	→ 0
0	1	1	0	
1	0	0	1	→ 1
1	0	1	1	
1	1	0	X	→ Invalid
1	1	1	X	

SR Flip-Flop

Characteristics Equation of SR FF



K-Map

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

∴ Characteristics equation of SR flip-flop is,

$$\boxed{Q_{n+1} = S + \bar{R}Q_n} \quad \text{...(5.1)}$$

and

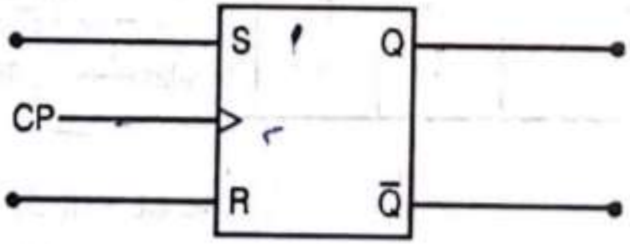
$$\boxed{\begin{matrix} SR = 0 \\ SR \neq 1 \end{matrix}} \quad \text{...(5.2)}$$

The characteristic equation is an algebraic expression for the binary information of the characteristic table. It specifies the value of the next state as a function of the present state and the inputs.

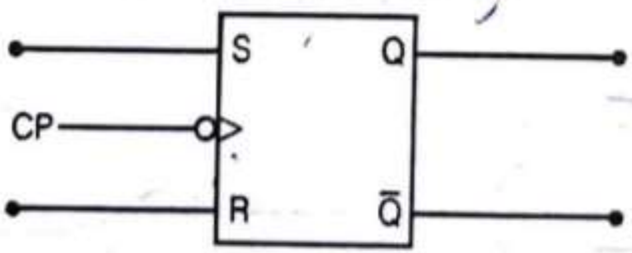
SR Flip-Flop

Graphical Symbol of SR FF

SR flip-flop has 3 inputs – S, R and CP



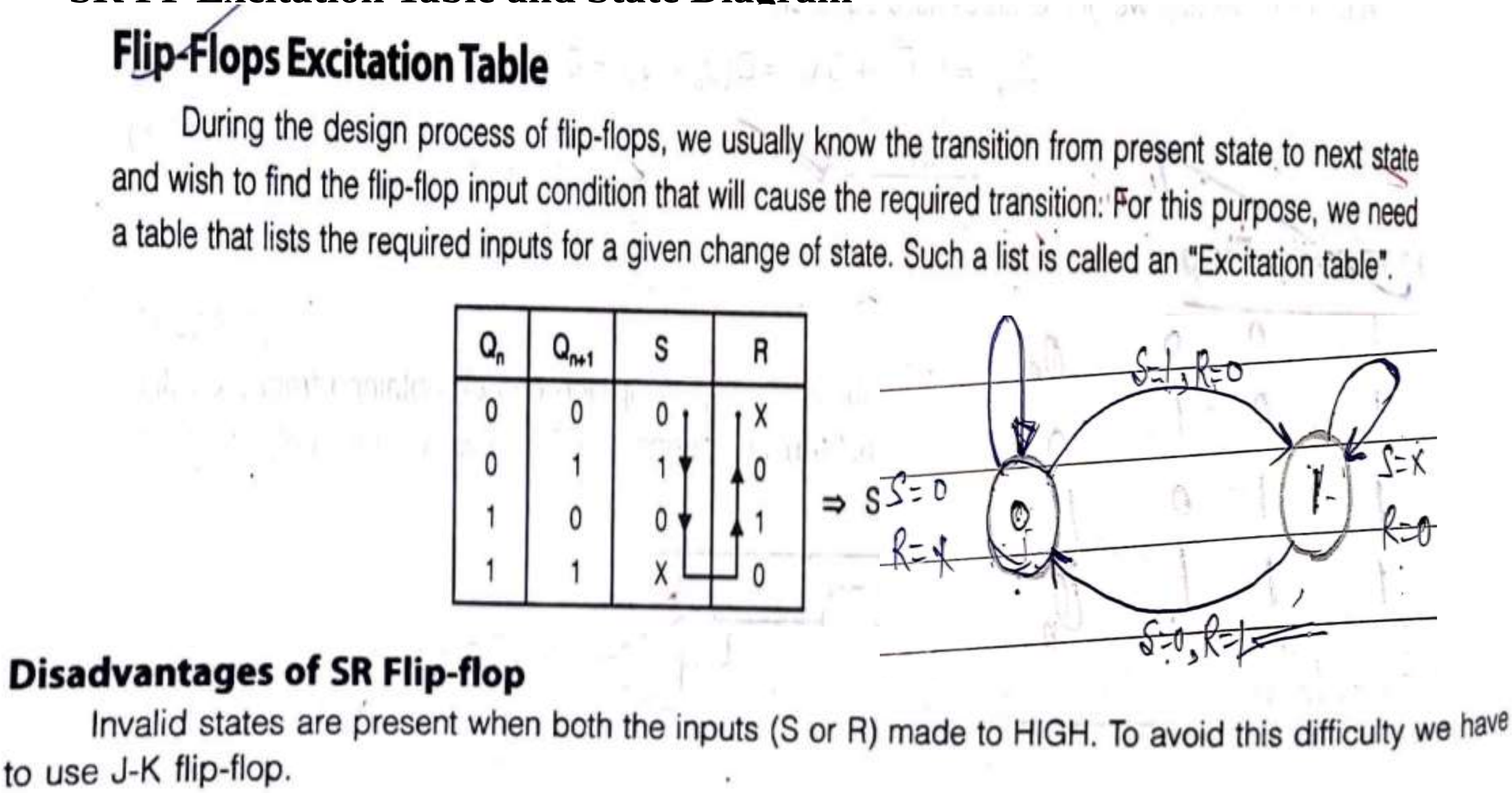
(+ve edge triggered SR-FF)



(–ve edge triggered SR-FF)

SR Flip-Flop

SR FF Excitation Table and State Diagram



Flip-Flops Excitation Table

During the design process of flip-flops, we usually know the transition from present state to next state and wish to find the flip-flop input condition that will cause the required transition. For this purpose, we need a table that lists the required inputs for a given change of state. Such a list is called an "Excitation table".

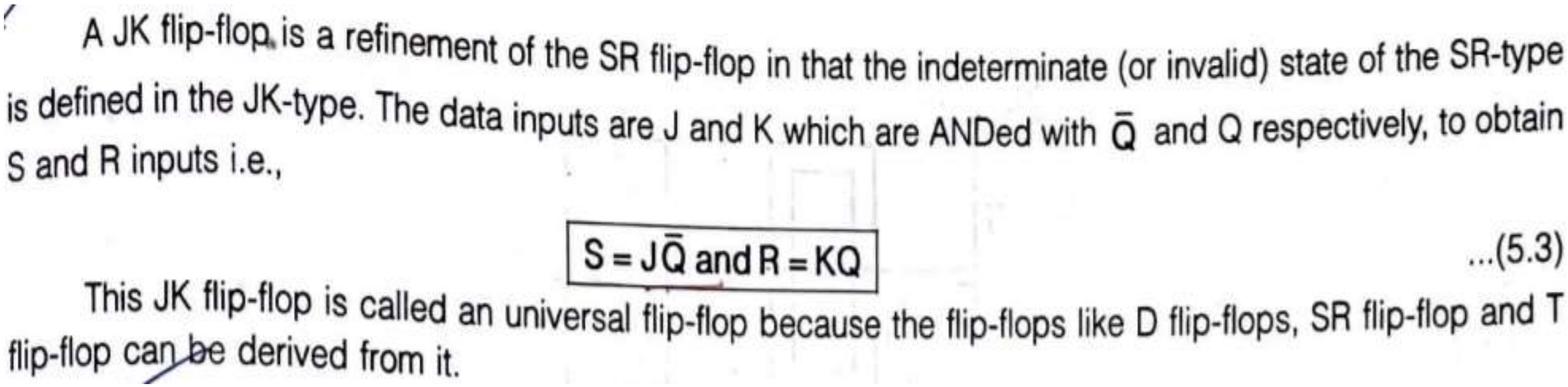
Q_n	Q_{n+1}	S	R
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0

Disadvantages of SR Flip-flop

Invalid states are present when both the inputs (S or R) made to HIGH. To avoid this difficulty we have to use J-K flip-flop.

JK Flip-Flop

JK FF

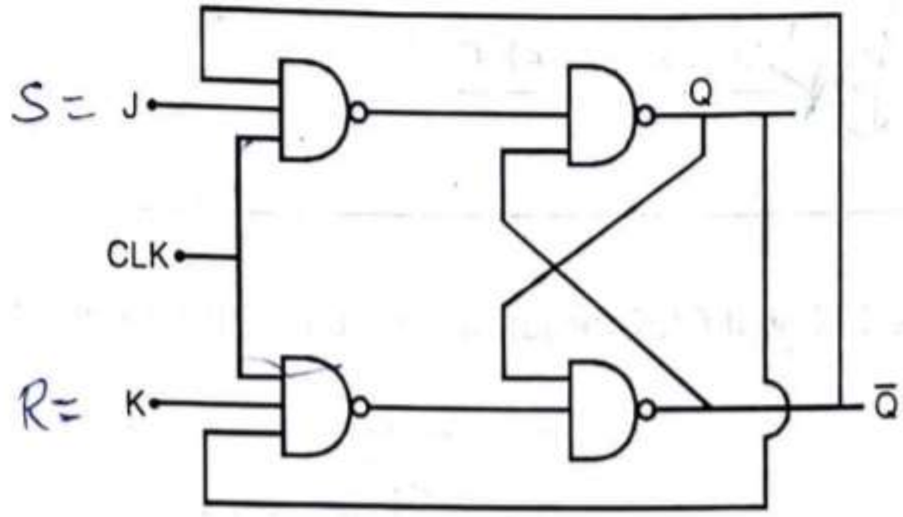


A JK flip-flop is a refinement of the SR flip-flop in that the indeterminate (or invalid) state of the SR-type is defined in the JK-type. The data inputs are J and K which are ANDed with $\bar{Q}$ and Q respectively, to obtain S and R inputs i.e.,

$$\boxed{S = J\bar{Q} \text{ and } R = KQ} \qquad ...(5.3)$$

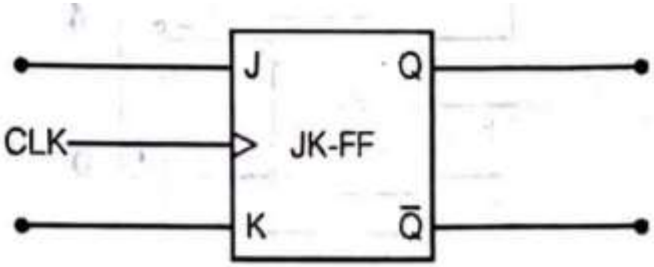
This JK flip-flop is called an universal flip-flop because the flip-flops like D flip-flops, SR flip-flop and T flip-flop can be derived from it.

Logical Diagram

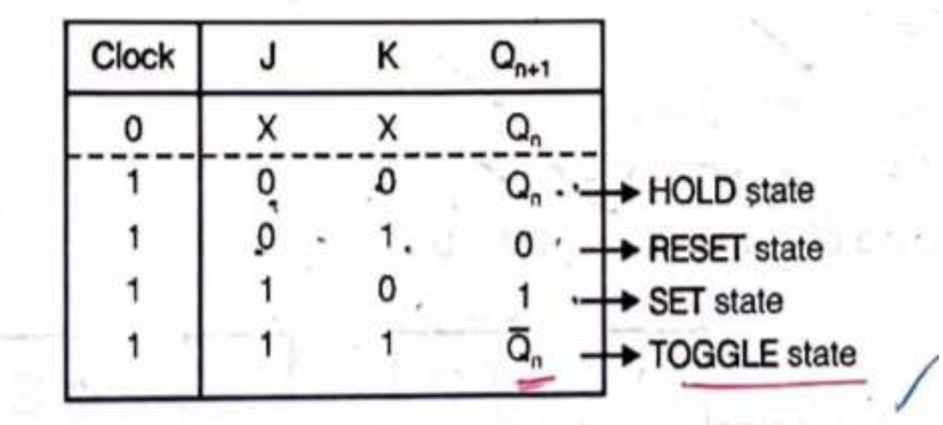


JK Flip-Flop

JK FF Graphical Diagram



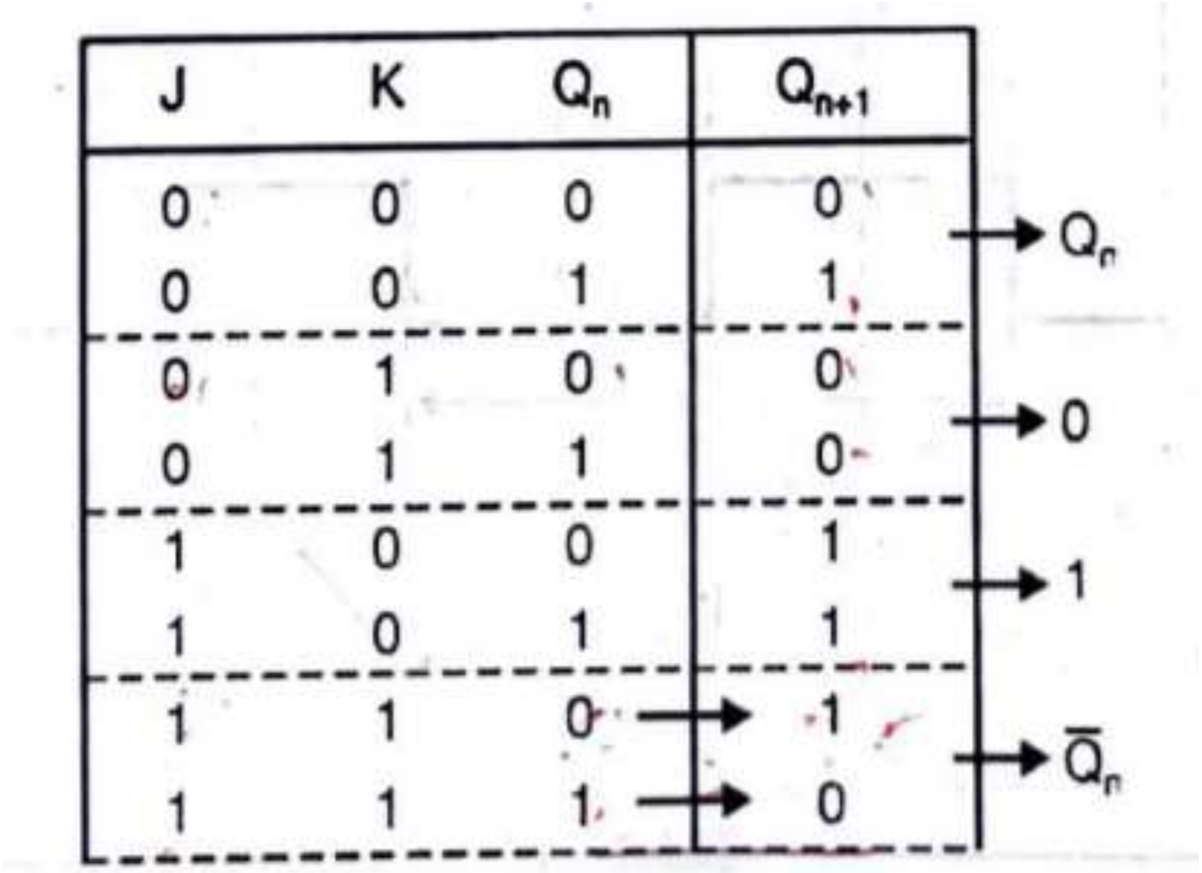
Truth Table of JK FF



Clock	J	K	Q_{n+1}	
0	X	X	Q_n	
1	0	0	Q_n	→ HOLD state
1	0	1	0	→ RESET state
1	1	0	1	→ SET state
1	1	1	$\bar{Q}_n$	→ TOGGLE state

JK Flip-Flop

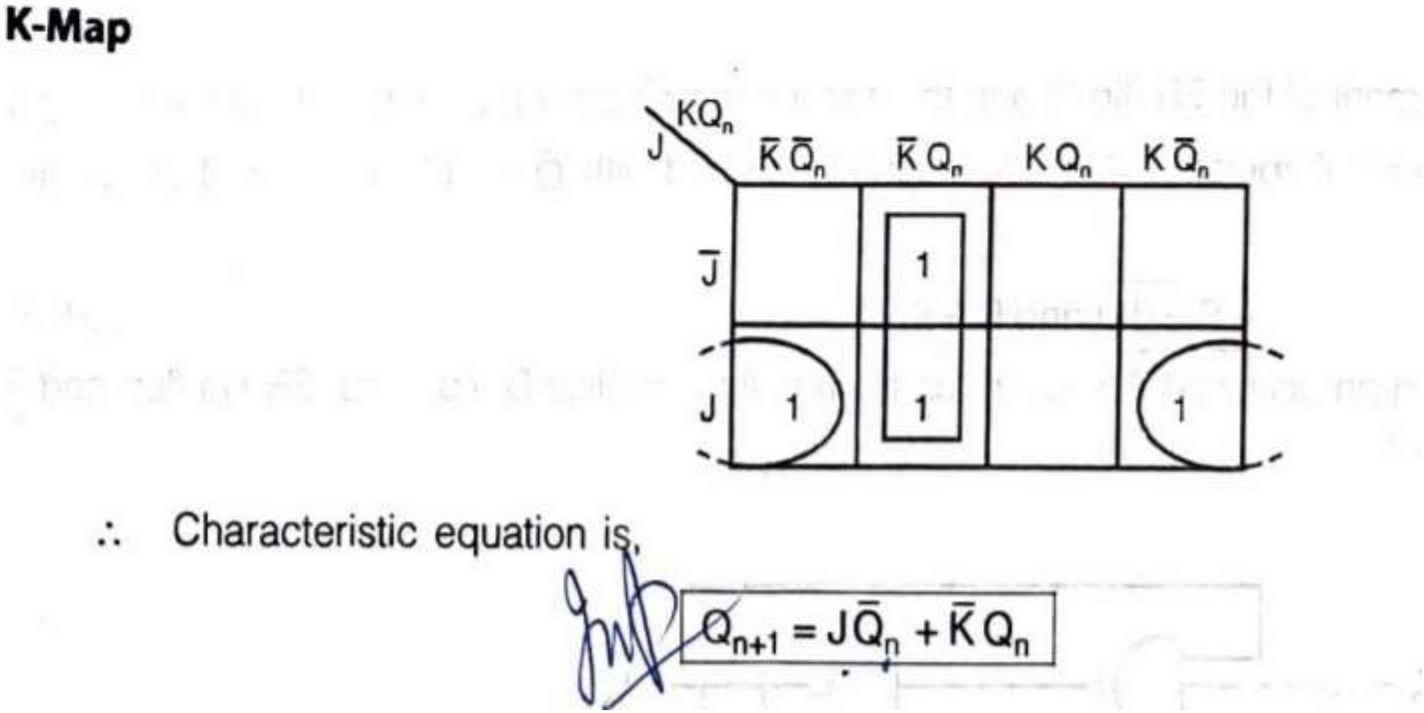
Characteristics Table of JK FF



J	K	Q_n	Q_{n+1}	
0	0	0	0	→ Q_n
0	0	1	1	
0	1	0	0	→ 0
0	1	1	0	
1	0	0	1	→ 1
1	0	1	1	
1	1	0 →	1	→ $\overline{Q}_n$
1	1	1 →	0	

JK Flip-Flop

Characteristics Equation of JK FF



K-Map

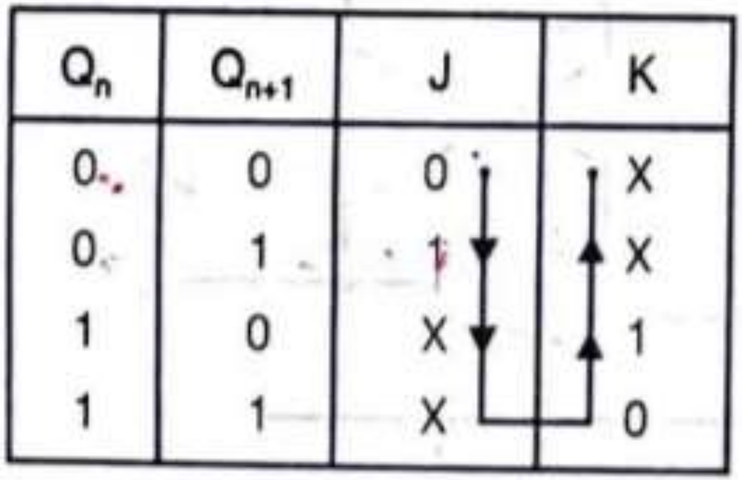
$J \backslash KQ_n$	$\bar{K}\bar{Q}_n$	$\bar{K}Q_n$	KQ_n	$K\bar{Q}_n$
$\bar{J}$		1		
J	1	1		1

∴ Characteristic equation is,

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

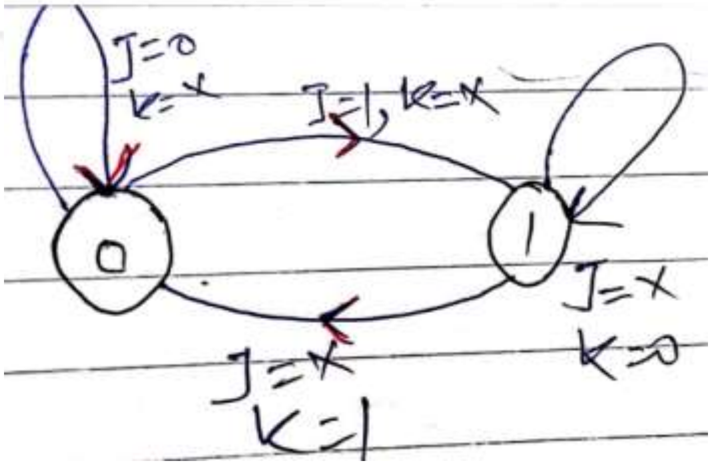
JK Flip-Flop

Excitation Table of JK FF and State Diagram



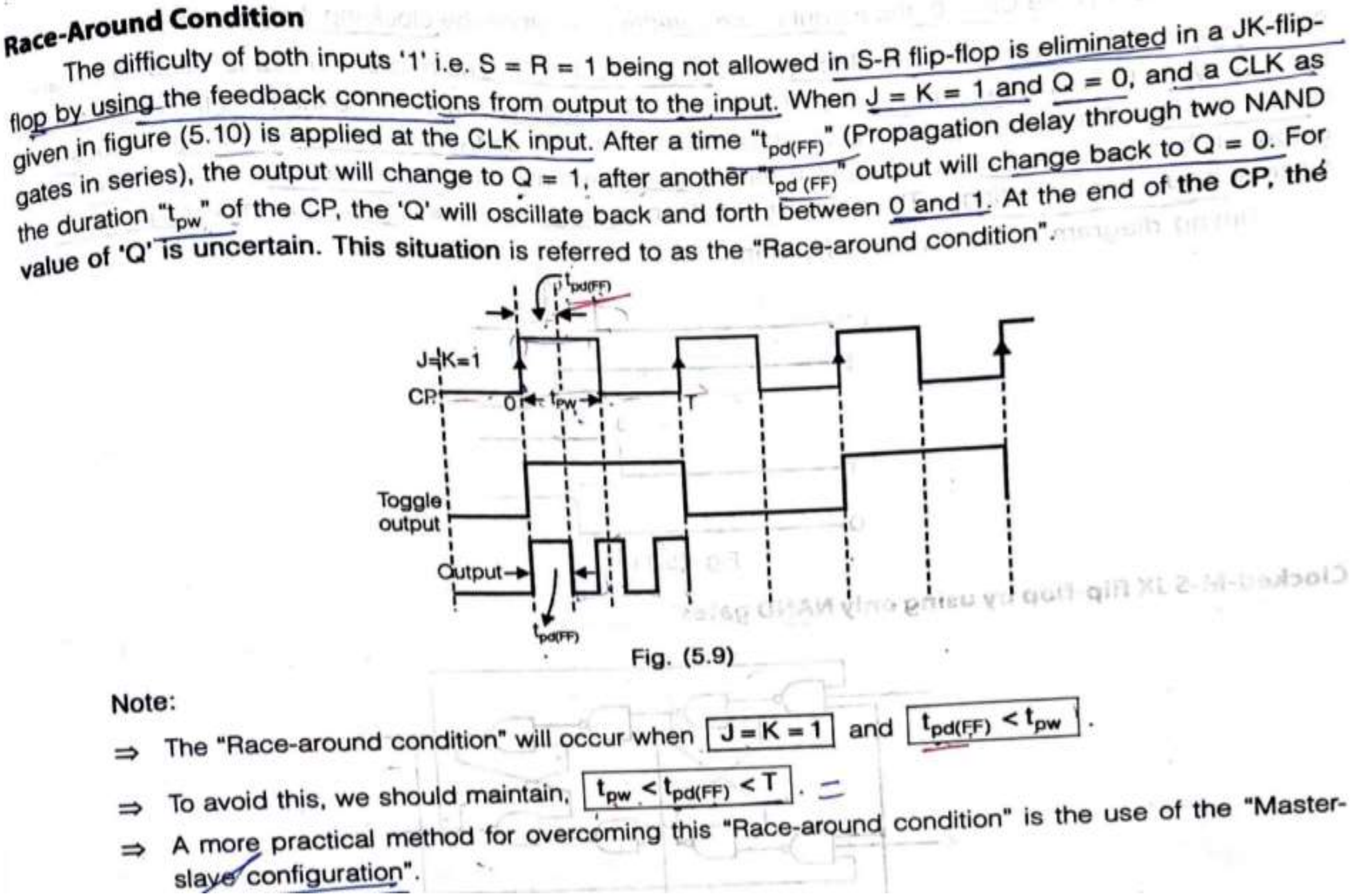
Q_n	Q_{n+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

⇒ JK-FF



JK Flip-Flop

Disadvantage of JK FF



Race-Around Condition

The difficulty of both inputs '1' i.e. S = R = 1 being not allowed in S-R flip-flop is eliminated in a JK-flip-flop by using the feedback connections from output to the input. When J = K = 1 and Q = 0, and a CLK as given in figure (5.10) is applied at the CLK input. After a time "$t_{pd(FF)}$" (Propagation delay through two NAND gates in series), the output will change to Q = 1, after another "$t_{pd\ (FF)}$" output will change back to Q = 0. For the duration "t_{pw}" of the CP, the 'Q' will oscillate back and forth between 0 and 1. At the end of the CP, the value of 'Q' is uncertain. This situation is referred to as the "Race-around condition".

Fig. (5.9)

Note:

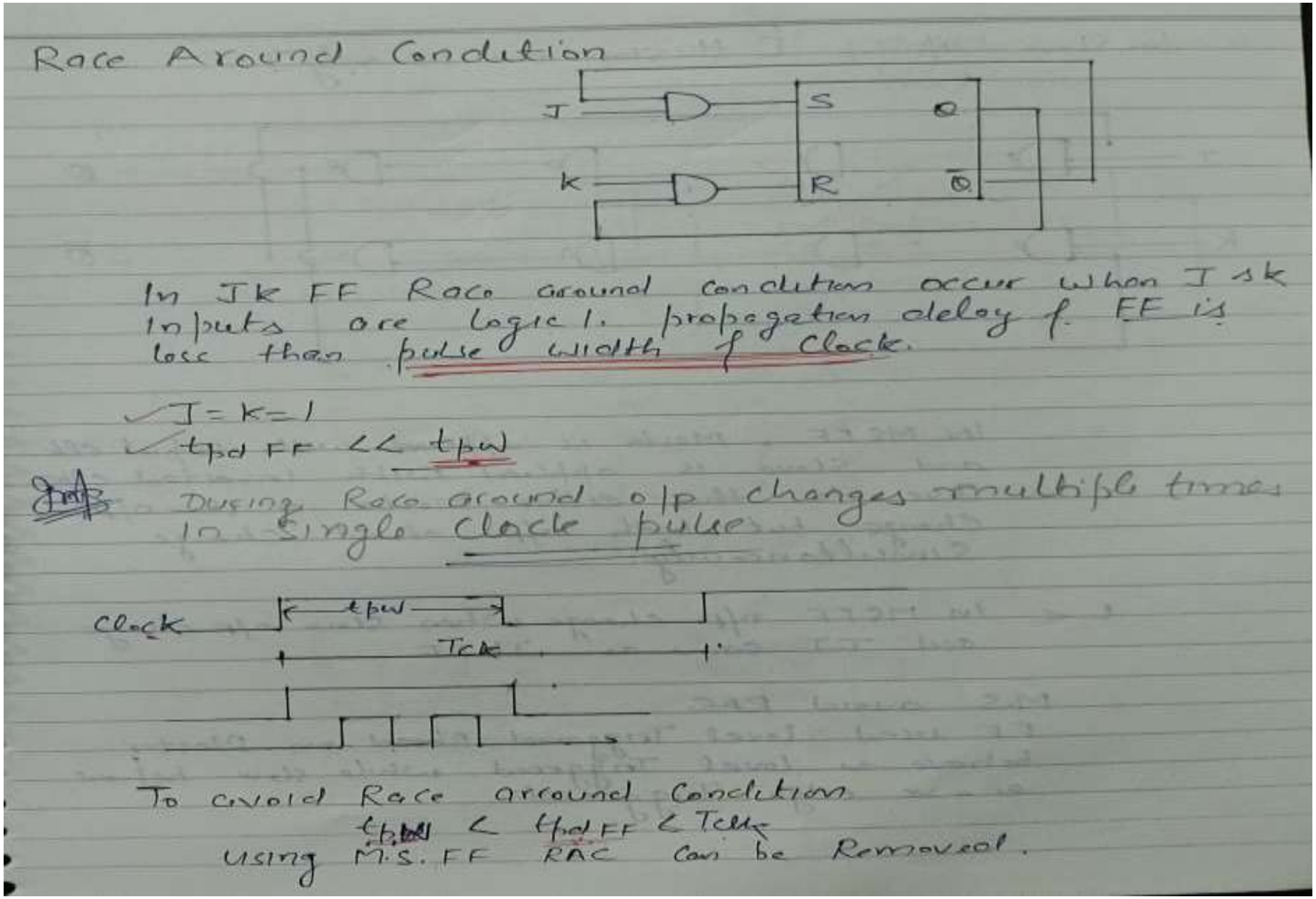
⇒ The "Race-around condition" will occur when $\boxed{J = K = 1}$ and $\boxed{t_{pd(FF)} < t_{pw}}$.

⇒ To avoid this, we should maintain, $\boxed{t_{pw} < t_{pd(FF)} < T}$.

⇒ A more practical method for overcoming this "Race-around condition" is the use of the "Master-slave configuration".

JK Flip-Flop

Disadvantage of JK FF



Race Around Condition

J
K
S
R
Q
$\bar{Q}$

In JK FF Race around condition occur when J & k inputs are logic 1. propagation delay of FF is less than pulse width of Clock.

✓ $J = k = 1$

✓ t_{pd} FF $<< t_{pw}$

Imp During Race around o/p changes multiple times in single clock pulse

Clock
t_{pw}
T_{clk}

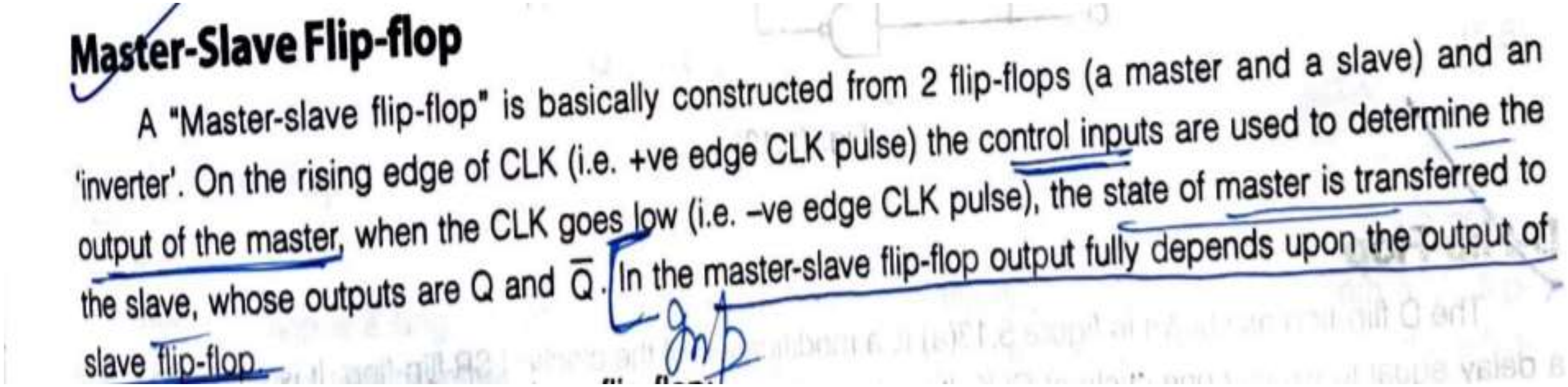
To avoid Race around Condition

$t_{pw} < t_{pd}$ FF $< T_{clk}$

using M.S. FF RAC can be Removed.

Master Slave Flip- Flop

MS FF

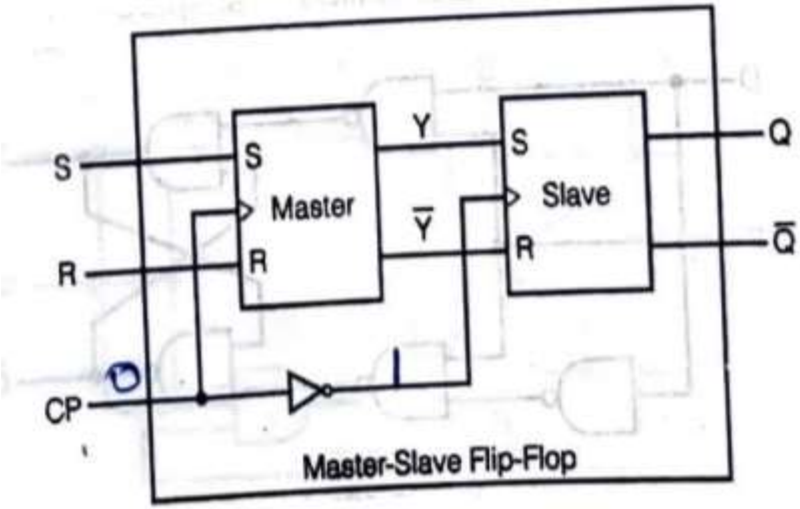


Master-Slave Flip-flop

A "Master-slave flip-flop" is basically constructed from 2 flip-flops (a master and a slave) and an 'inverter'. On the rising edge of CLK (i.e. +ve edge CLK pulse) the control inputs are used to determine the output of the master, when the CLK goes low (i.e. –ve edge CLK pulse), the state of master is transferred to the slave, whose outputs are Q and $\overline{Q}$. In the master-slave flip-flop output fully depends upon the output of slave flip-flop.

Imp

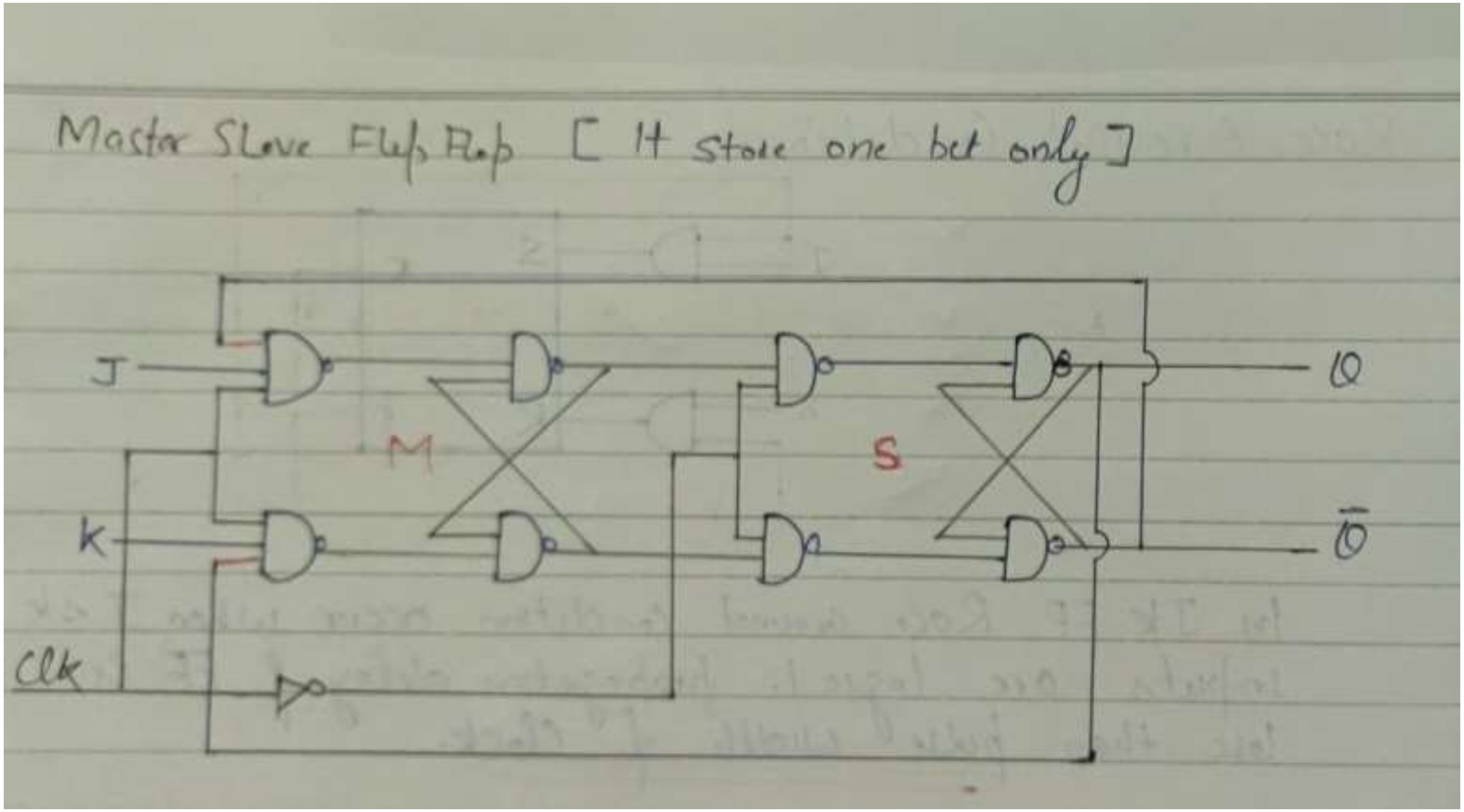
MS FF Logical Diagram



Master-Slave Flip-Flop

Flip- Flop

Clocked MS FF using NAND Gate



Master Slave Flip Flop [It store one bit only]

D Flip-Flop

D FF

The D FF as shown in figure 5.13(a) is a modification of clocked SR FF. It is a FF with a delay equal to exactly one cycle of CLK. It is also called **“Transparent Latch”**

Logical Diagram

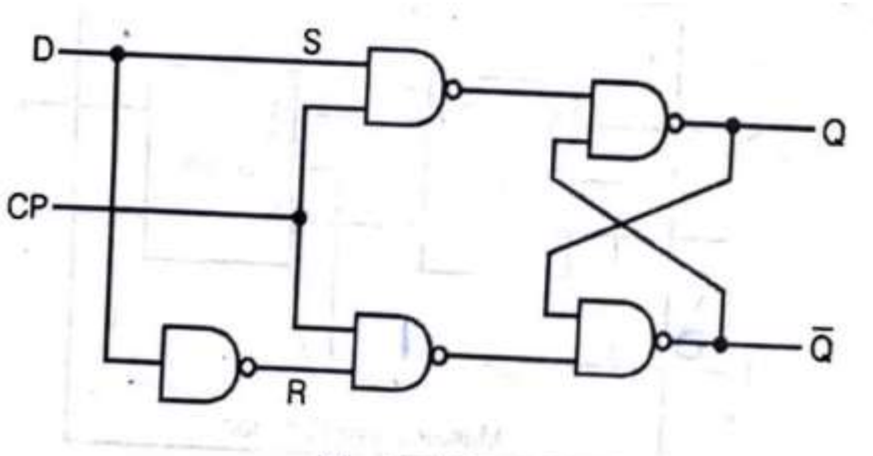
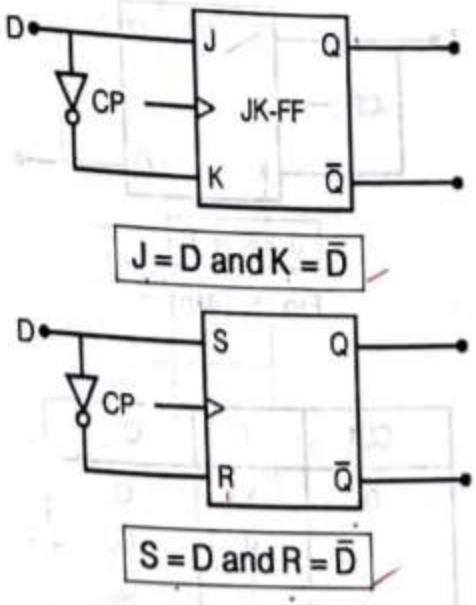


Fig. 5.13(a)

D Flip-Flop

Graphical Diagram



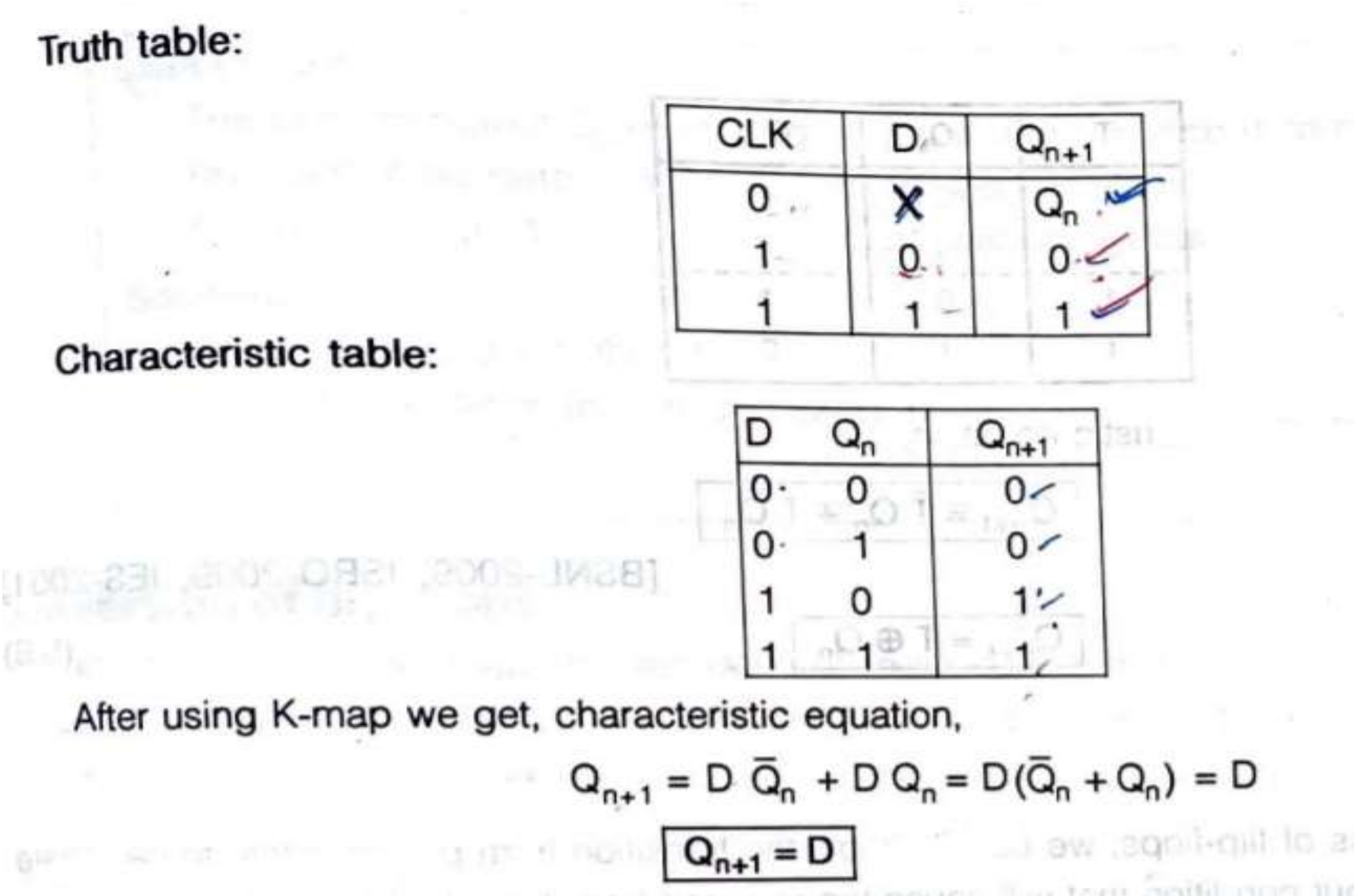
$J = D$ and $K = \bar{D}$

$S = D$ and $R = \bar{D}$

Fig. 5.13(b)

D Flip-Flop

Truth Table and Characteristics Table



Truth table:

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

Characteristic table:

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

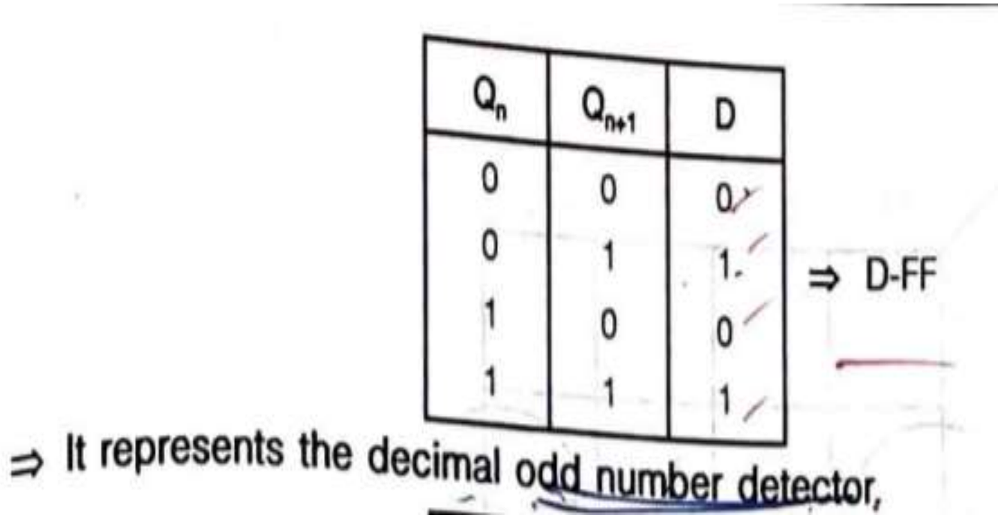
After using K-map we get, characteristic equation,

$$Q_{n+1} = D\,\bar{Q}_n + D\,Q_n = D(\bar{Q}_n + Q_n) = D$$

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

D Flip-Flop

Excitation Table

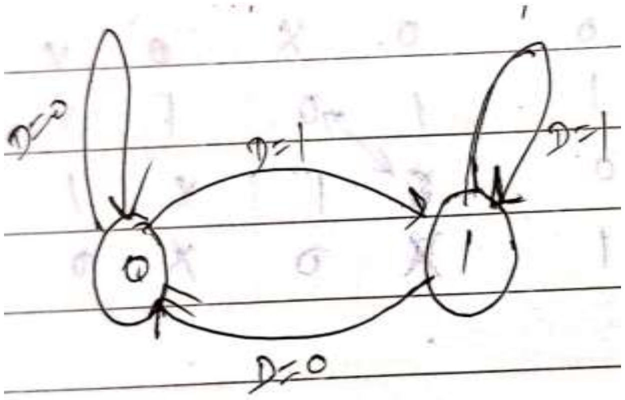


Q_n	Q_{n+1}	D
0	0	0
0	1	1.
1	0	0
1	1	1

⇒ D-FF

⇒ It represents the decimal odd number detector,

State Diagram



T Flip-Flop

T FF

The T FF is a single input inversion of the JK FF. T FF can be obtained from JK FF if J and K tied together as in figure 5.14(a). The designation “T” comes from the ability of the FF to **“TOGGLE”** or **“Change State”**

Logical Diagram

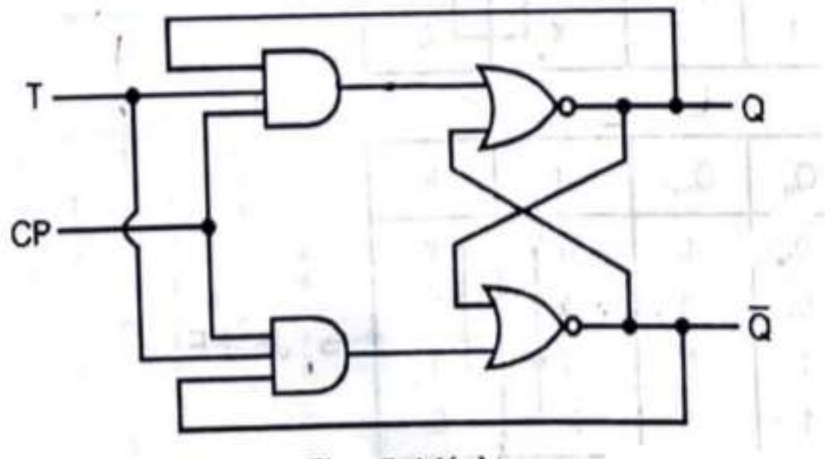


Fig. 5.14(a)

T Flip-Flop

Graphical Symbol

Graphical symbol of T flip-flop:

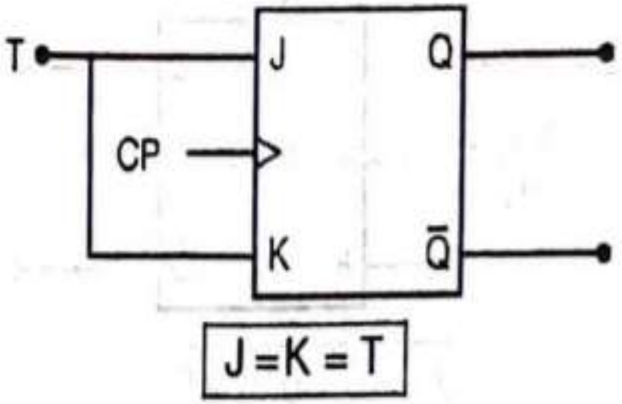
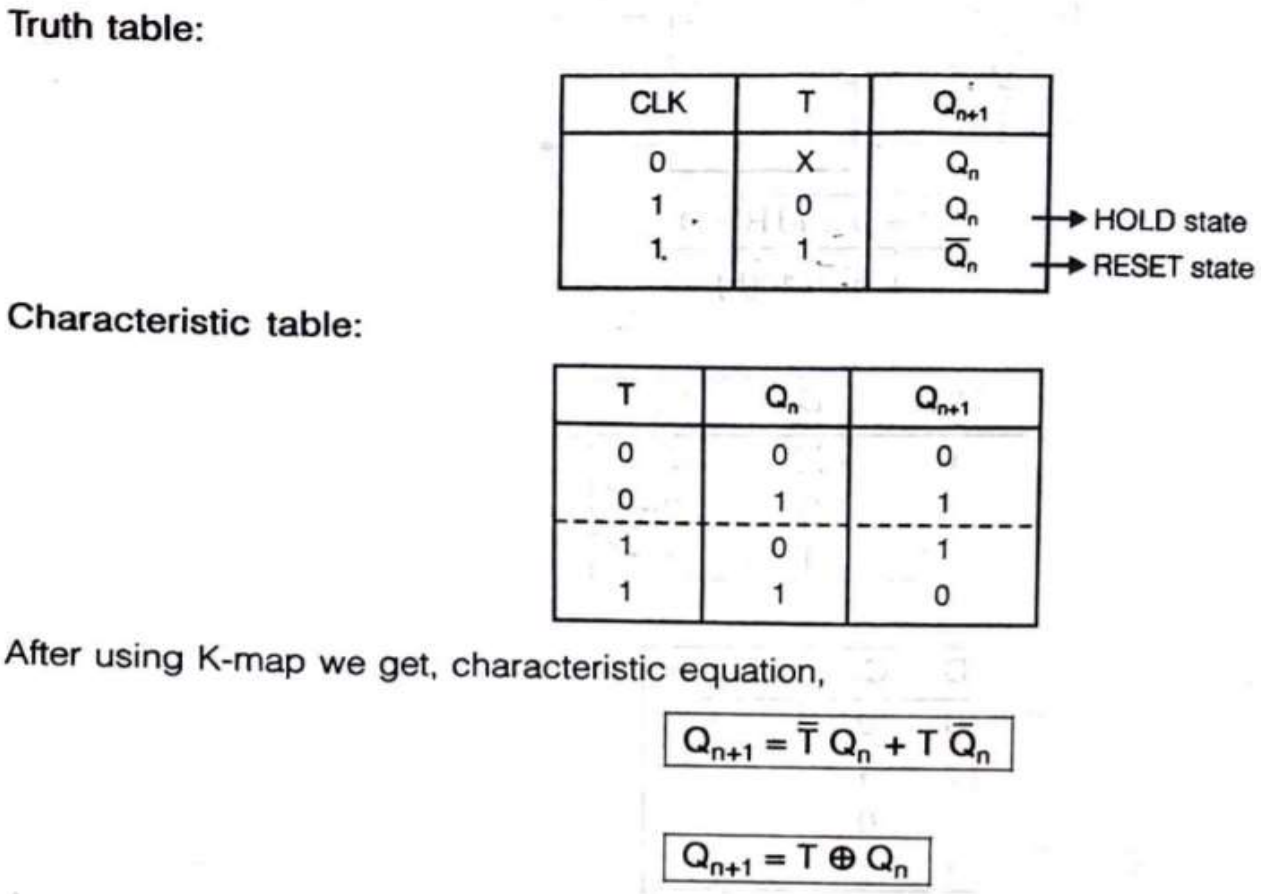


Fig. 5.14(b)

T Flip-Flop

Truth Table and Characteristics Table



Truth table:

CLK	T	Q_{n+1}	
0	X	Q_n	
1	0	Q_n	→ HOLD state
1	1	$\overline{Q}_n$	→ RESET state

Characteristic table:

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

After using K-map we get, characteristic equation,

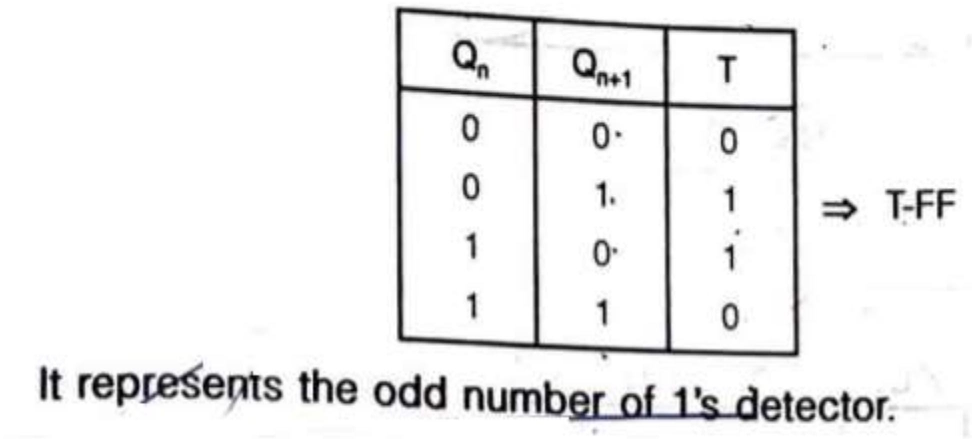
$$\boxed{Q_{n+1} = \overline{T}\, Q_n + T\, \overline{Q}_n}$$

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

T Flip-Flop

Excitation Table

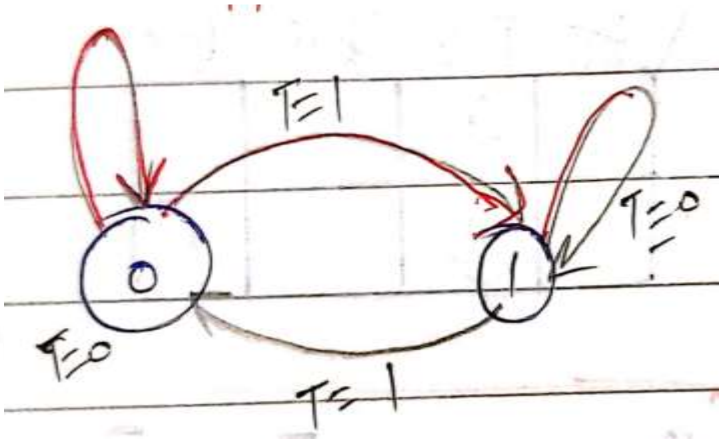
State Diagram



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

⇒ T-FF

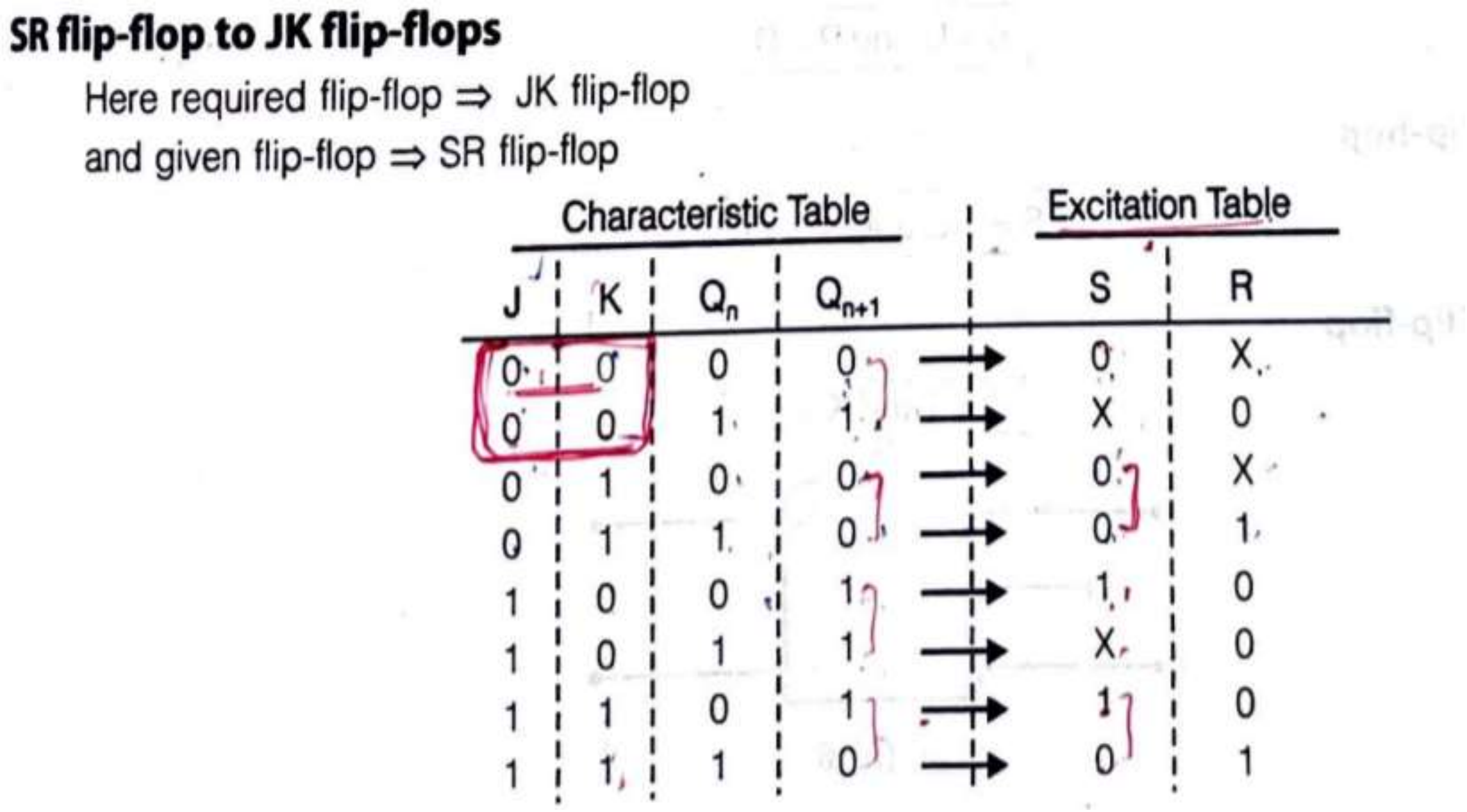
It represents the odd number of 1's detector.



Conversion of Flip-Flop

- **Firstly write the characteristics table of required FF**
- **Now write the excitation table of available or given FF**
- **Write excitation equation**
- **Minimize excitation equation**
- **Implement logic circuit**

SR FF to JK FF



SR flip-flop to JK flip-flops

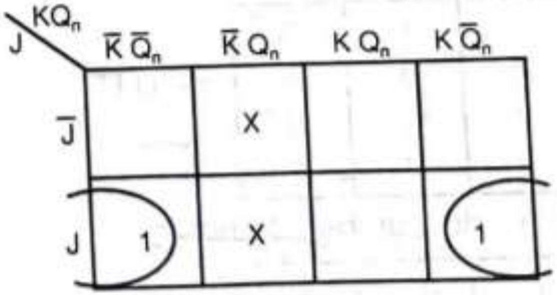
Here required flip-flop ⇒ JK flip-flop

and given flip-flop ⇒ SR flip-flop

Characteristic Table					Excitation Table	
J	K	Q_n	Q_{n+1}		S	R
0	0	0	0	→	0	X
0	0	1	1	→	X	0
0	1	0	0	→	0	X
0	1	1	0	→	0	1
1	0	0	1	→	1	0
1	0	1	1	→	X	0
1	1	0	1	→	1	0
1	1	1	0	→	0	1

SR FF to JK FF

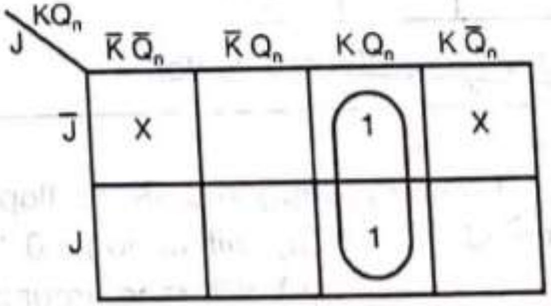
Now K-Map for S:



$J \backslash KQ_n$	$\bar{K}\bar{Q}_n$	$\bar{K}Q_n$	KQ_n	$K\bar{Q}_n$
$\bar{J}$		X		
J	1	X		1

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

K-Map for R:



$J \backslash KQ_n$	$\bar{K}\bar{Q}_n$	$\bar{K}Q_n$	KQ_n	$K\bar{Q}_n$
$\bar{J}$	X		1	X
J			1	

$$\boxed{R = KQ_n}$$

SR FF to JK FF

Now final circuit diagram is,

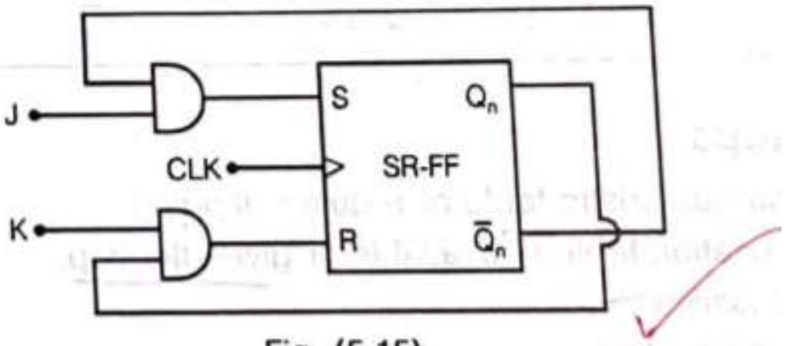


Fig. (5.15)

Applications of flip-flops

- Bounce elimination switch or chatterless switch
- Latch
- Registers
- Counters
- Memory etc.

FF Conversion

SR flip-flop to D flip-flop

$$\boxed{S = D \text{ and } R = \bar{D}}$$

SR flip-flop to T flip-flop

$$\boxed{S = T\bar{Q} \text{ and } R = TQ}$$

JK flip-flop to SR flip-flop

$$\boxed{J = S \text{ and } K = R}$$

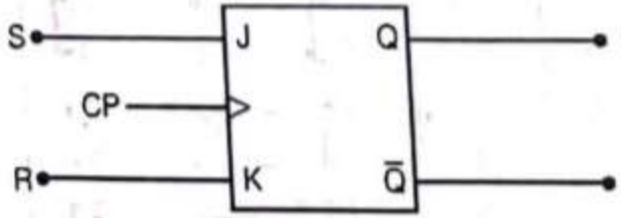
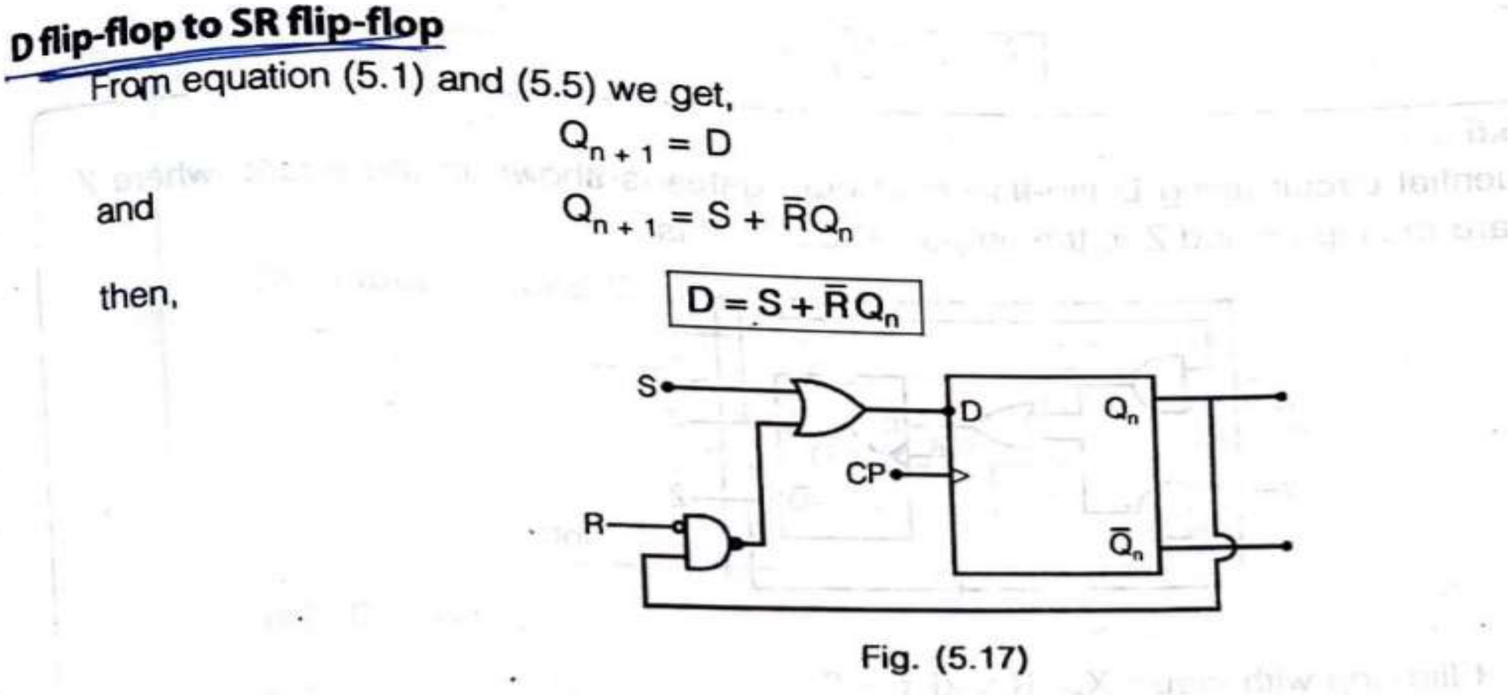


Fig. (5.16)

FF Conversion

- JK FF to D FF $\boxed{J = D \text{ and } K = \bar{D}}$
- JK FF to T FF $\boxed{J = K = T}$



D flip-flop to SR flip-flop

From equation (5.1) and (5.5) we get,

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

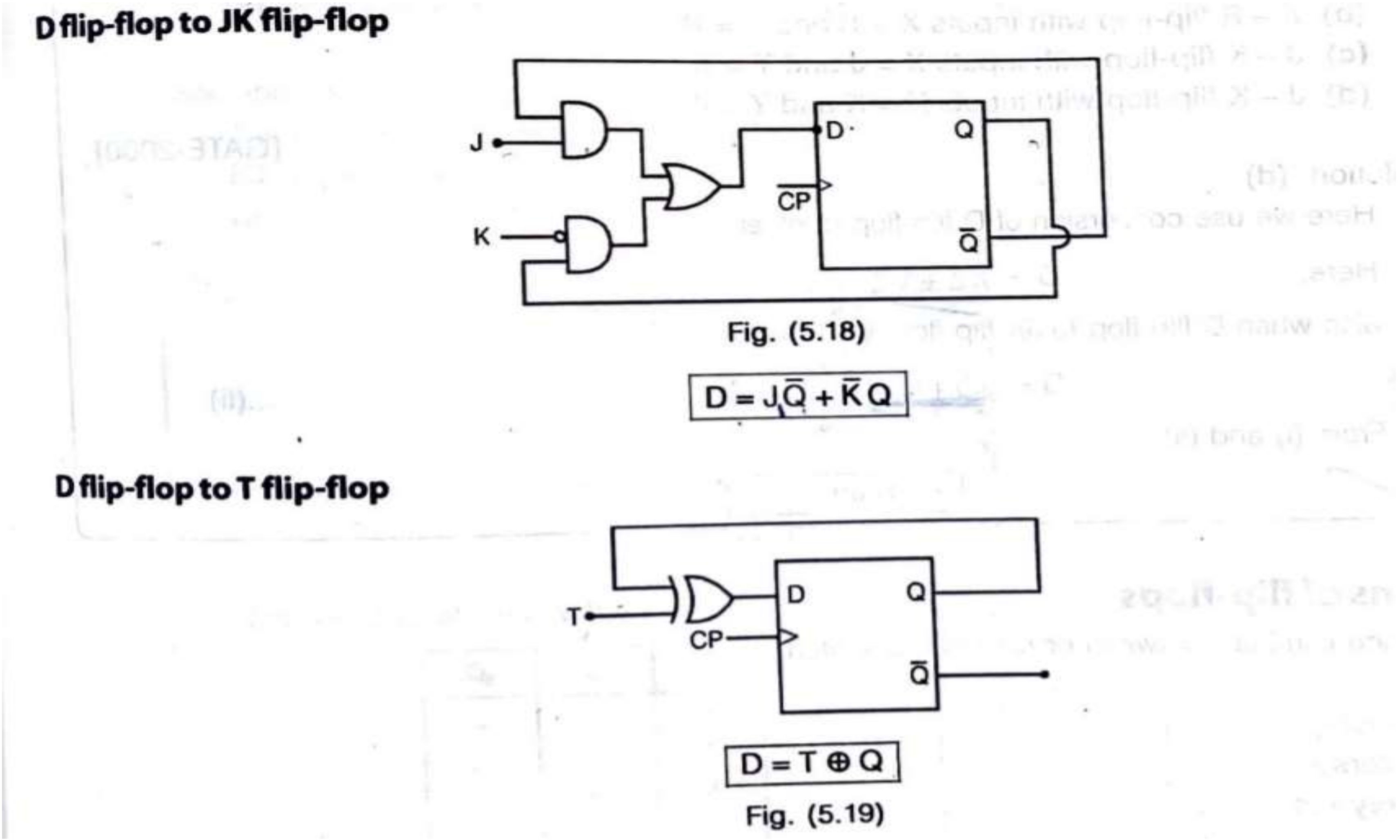
and $$Q_{n+1} = S + \bar{R}Q_n$$

then, $$\boxed{D = S + \bar{R}Q_n}$$

S
R
CP
D
Q_n
$\bar{Q}_n$

Fig. (5.17)

FF Conversion



D flip-flop to JK flip-flop

Fig. (5.18)

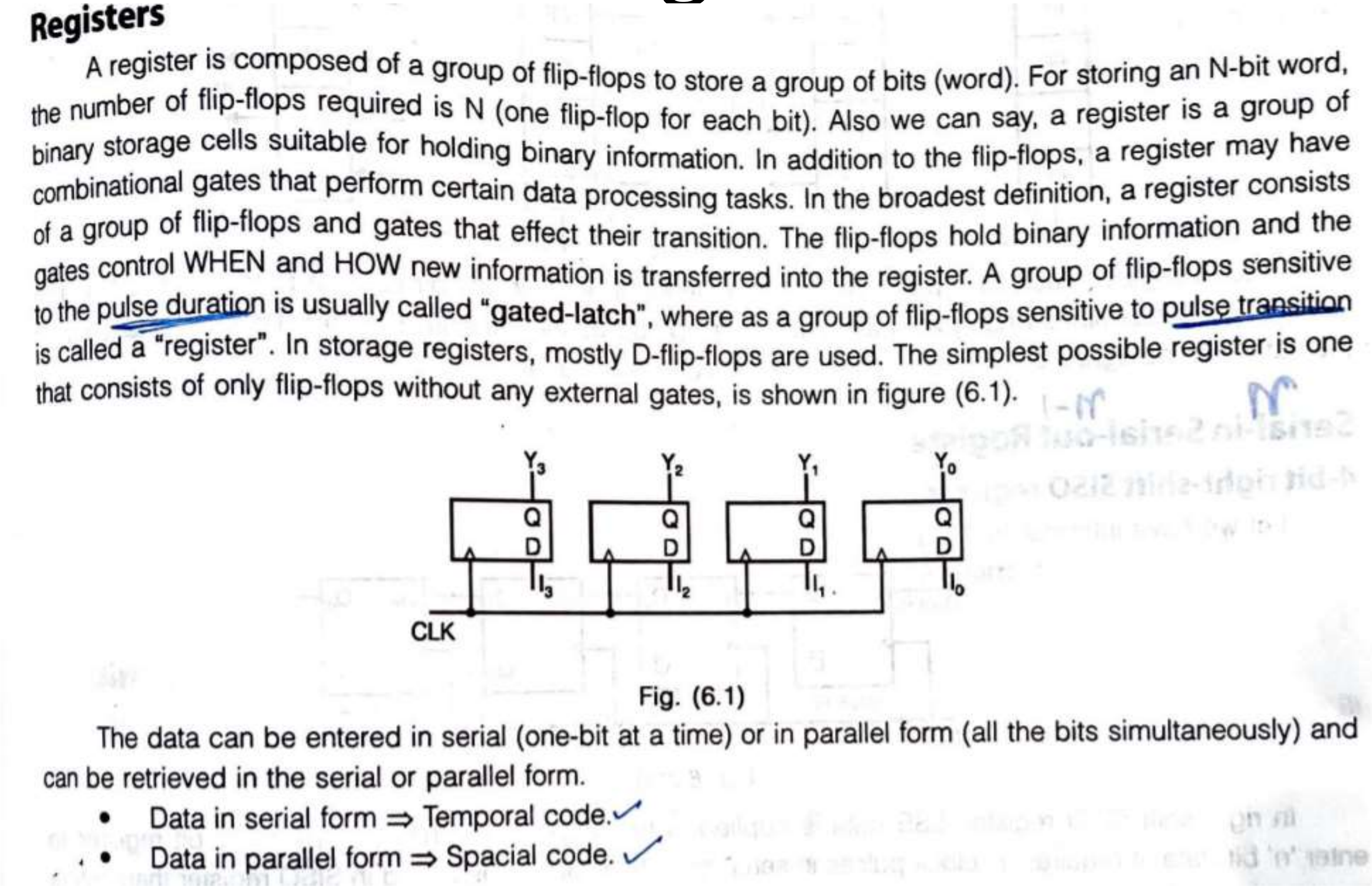
$$\boxed{D = J\overline{Q} + \overline{K}Q}$$

D flip-flop to T flip-flop

$$\boxed{D = T \oplus Q}$$

Fig. (5.19)

Register



Registers

A register is composed of a group of flip-flops to store a group of bits (word). For storing an N-bit word, the number of flip-flops required is N (one flip-flop for each bit). Also we can say, a register is a group of binary storage cells suitable for holding binary information. In addition to the flip-flops, a register may have combinational gates that perform certain data processing tasks. In the broadest definition, a register consists of a group of flip-flops and gates that effect their transition. The flip-flops hold binary information and the gates control WHEN and HOW new information is transferred into the register. A group of flip-flops sensitive to the pulse duration is usually called **"gated-latch"**, where as a group of flip-flops sensitive to pulse transition is called a "register". In storage registers, mostly D-flip-flops are used. The simplest possible register is one that consists of only flip-flops without any external gates, is shown in figure (6.1).

Fig. (6.1)

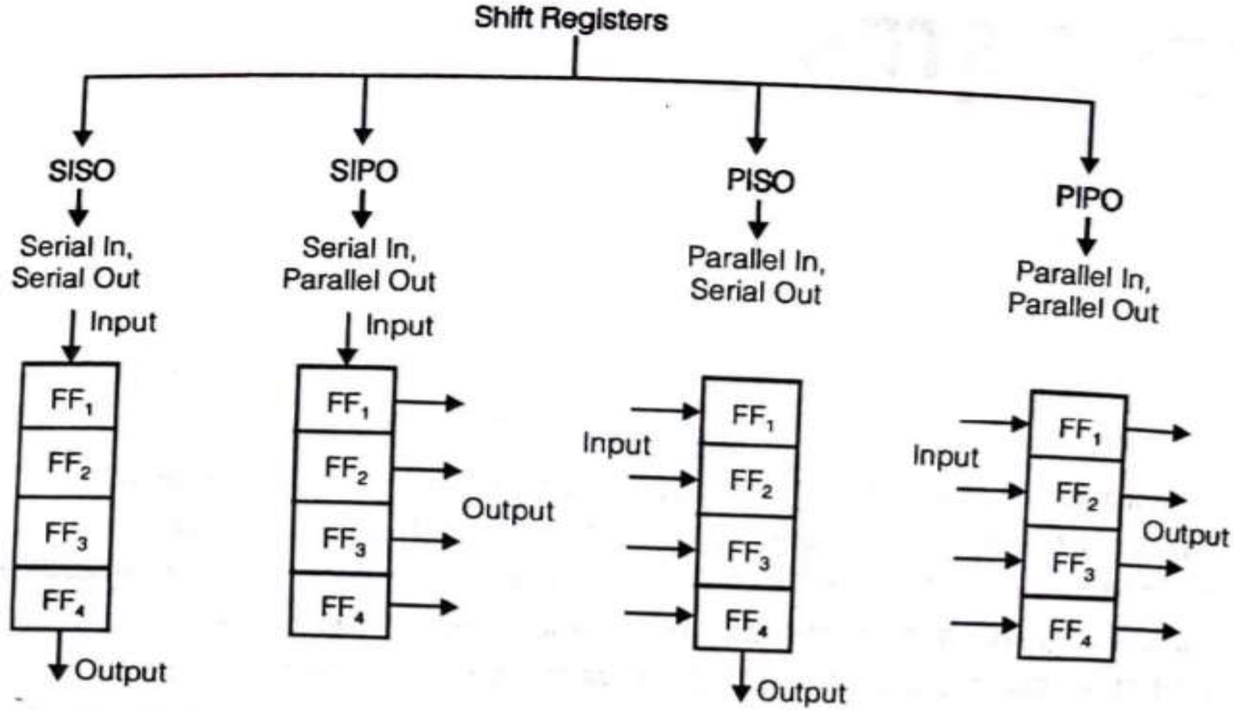
The data can be entered in serial (one-bit at a time) or in parallel form (all the bits simultaneously) and can be retrieved in the serial or parallel form.

- Data in serial form ⇒ Temporal code. ✓
- Data in parallel form ⇒ Spacial code. ✓

Shift Register

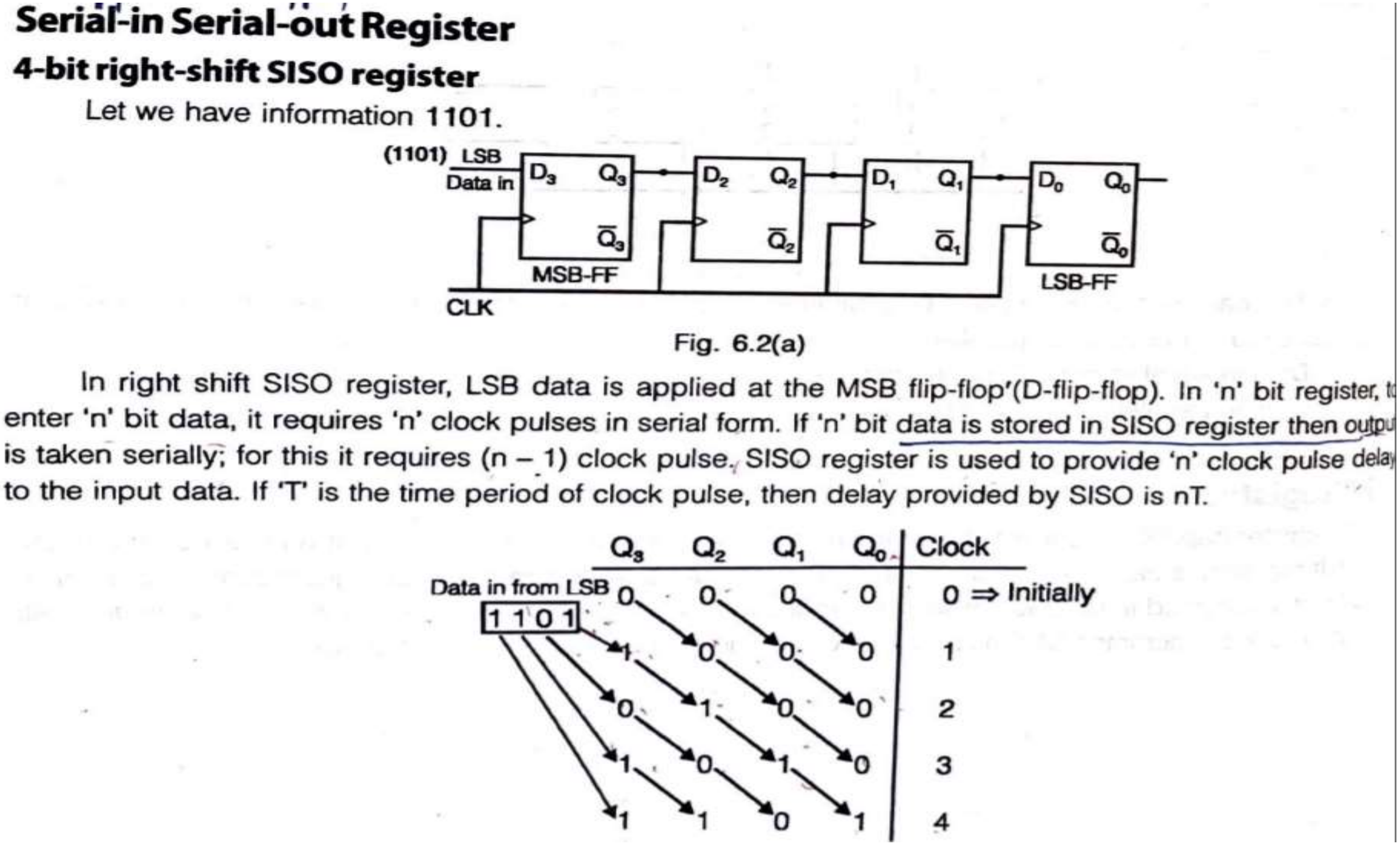
A register capable of shifting its binary information either to the right (right shift register) or to the Left (Left-shift register) is called the "Shift register". The logical configuration of a shift register consists of a chain of flip-flops connected in cascade, with the output of one flip-flop connected to the input of next flip-flop. All flip-flops receive a common CLK pulse which cause the shift from one state to the next.

Classification of shift register depending upon the way in which data are entered and received



SISO Right Shift Register

- In shift register each CLK pulse shifts the contents of register 1 bit position to the right or left. The "serial input" determines what goes into the leftmost FF during the shift. The "serial output" is taken from the output of the right most FF prior to the application of a pulse



(1101) LSB
Data in
D3
Q3
Q̄3
MSB-FF
D2
Q2
Q̄2
D1
Q1
Q̄1
D0
Q0
Q̄0
LSB-FF
CLK

Serial-in Serial-out Register

4-bit right-shift SISO register

Let we have information 1101.

Fig. 6.2(a)

In right shift SISO register, LSB data is applied at the MSB flip-flop'(D-flip-flop). In 'n' bit register, to enter 'n' bit data, it requires 'n' clock pulses in serial form. If 'n' bit data is stored in SISO register then output is taken serially; for this it requires (n – 1) clock pulse. SISO register is used to provide 'n' clock pulse delay to the input data. If 'T' is the time period of clock pulse, then delay provided by SISO is nT.

Data in from LSB 1101

Q_3	Q_2	Q_1	Q_0	Clock
0	0	0	0	0 ⇒ Initially
1	0	0	0	1
0	1	0	0	2
1	0	1	0	3
1	1	0	1	4

SISO Left Shift Register

4-bit left-shift SISO register

Let we have information 1101.

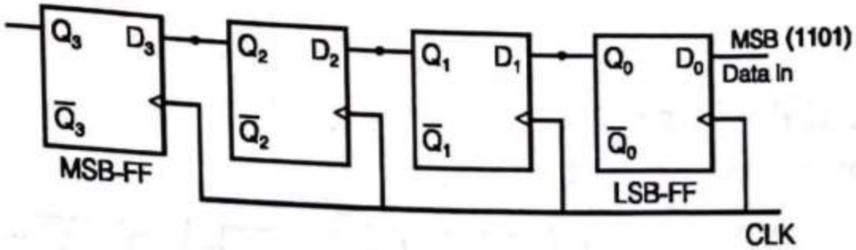


Fig. 6.2(b)

In this above SISO register MSB data is applied to the LSB flip-flop(D-flip-flop). To enter the 'n' bit data in serial form we require 'n' clock pulse. To exit or getting output of 'n' bit data as serially we require (n – 1) clock pulse.

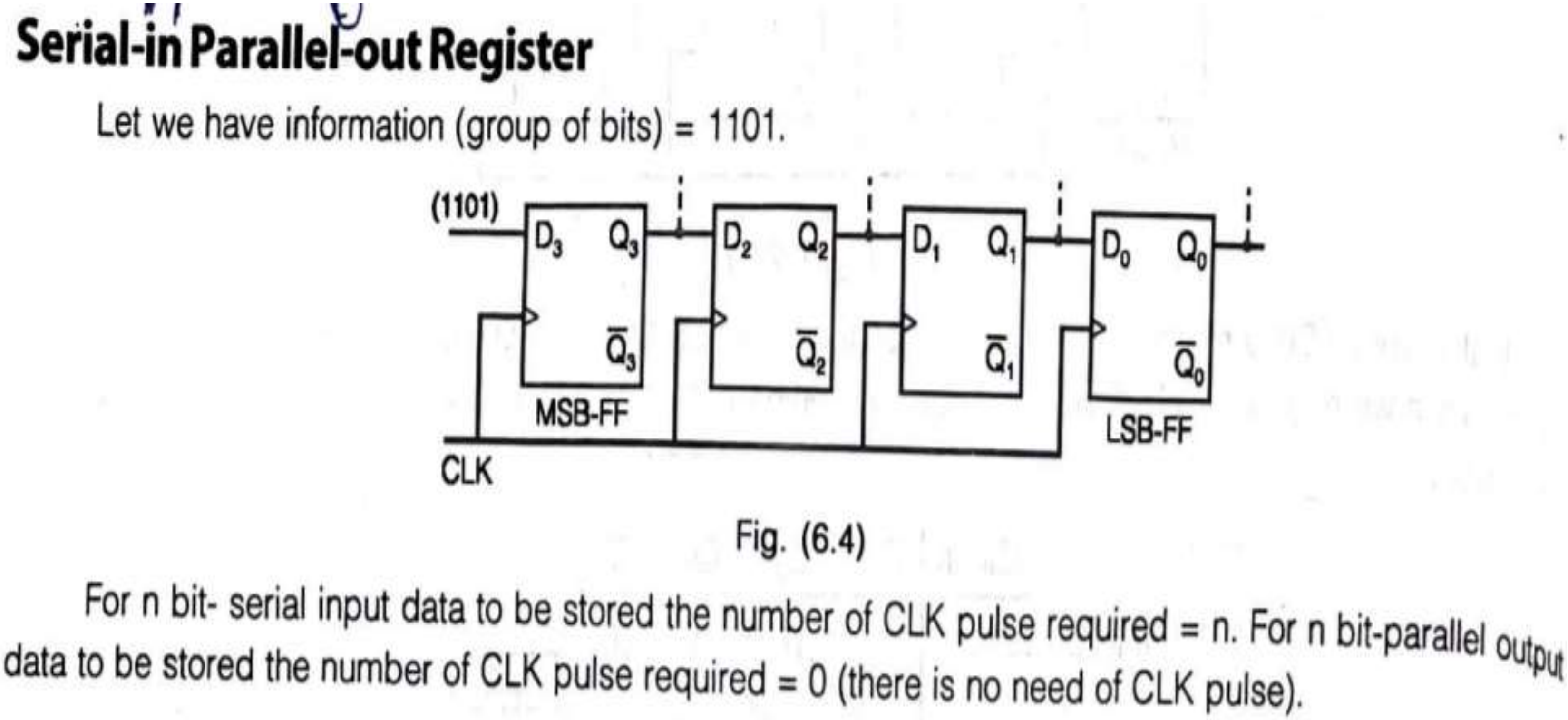
	Clock	Q_3	Q_2	Q_1	Q_0
Initially ⇐	0	0	0	0	0
	1	0	0	0	1
	2	0	0	1	1
	3	0	1	1	0
	4	1	1	0	1

Data in from MSB

1 1 0 1

⇒ From truth table it is clear that the data 1101 stores from MSB in the left shift way after 4 clock pulses.

SIPO Register



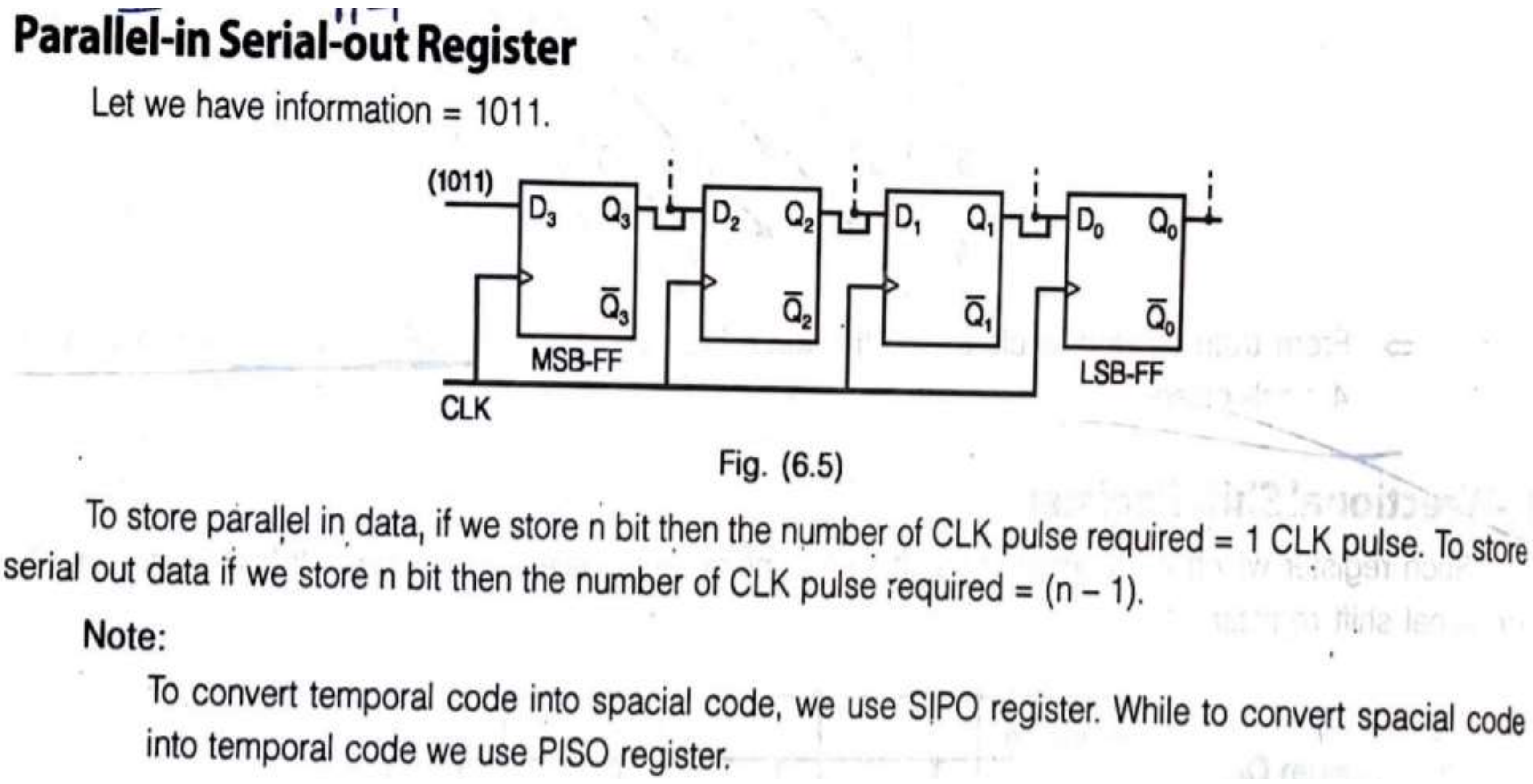
Serial-in Parallel-out Register

Let we have information (group of bits) = 1101.

Fig. (6.4)

For n bit- serial input data to be stored the number of CLK pulse required = n. For n bit-parallel output data to be stored the number of CLK pulse required = 0 (there is no need of CLK pulse).

PISO-Register



Parallel-in Serial-out Register

Let we have information = 1011.

(1011)
D3
Q3
Q̄3
MSB-FF
D2
Q2
Q̄2
D1
Q1
Q̄1
D0
Q0
Q̄0
LSB-FF
CLK

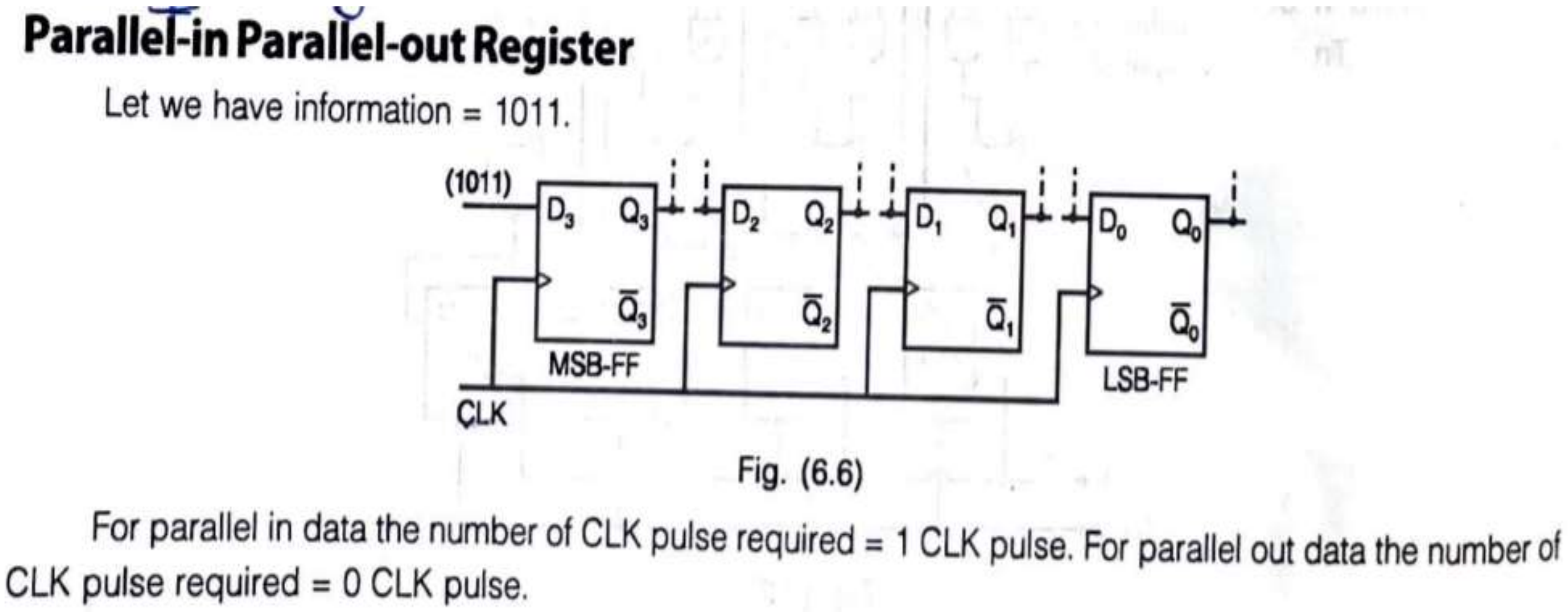
Fig. (6.5)

To store parallel in data, if we store n bit then the number of CLK pulse required = 1 CLK pulse. To store serial out data if we store n bit then the number of CLK pulse required = (n – 1).

Note:

To convert temporal code into spacial code, we use SIPO register. While to convert spacial code into temporal code we use PISO register.

PIPO-Register



Parallel-in Parallel-out Register

Let we have information = 1011.

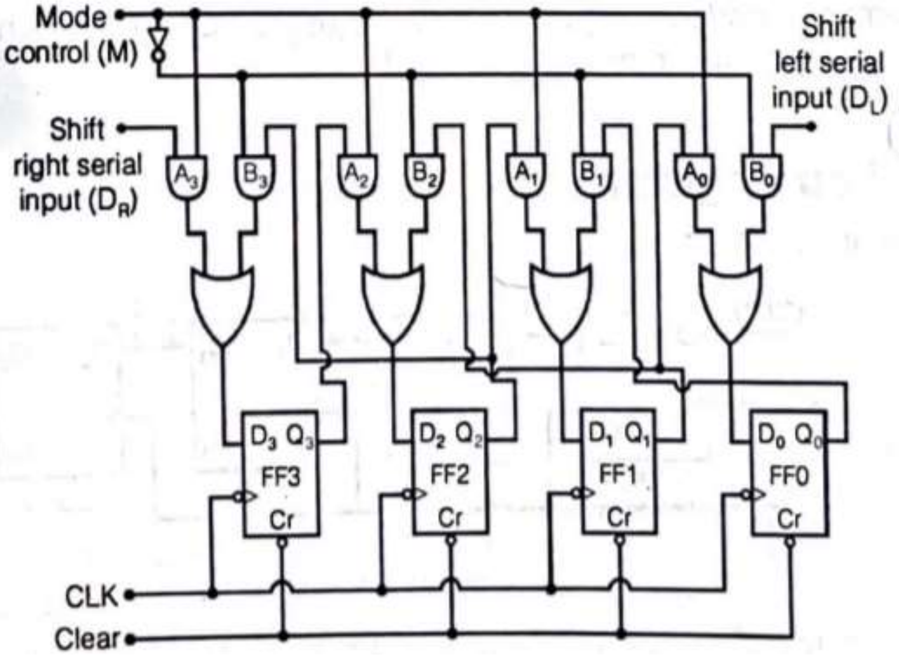
(1011)
D_3
Q_3
$\overline{Q}_3$
MSB-FF
D_2
Q_2
$\overline{Q}_2$
D_1
Q_1
$\overline{Q}_1$
D_0
Q_0
$\overline{Q}_0$
LSB-FF
CLK

Fig. (6.6)

For parallel in data the number of CLK pulse required = 1 CLK pulse. For parallel out data the number of CLK pulse required = 0 CLK pulse.

Bi Directional Shift Register

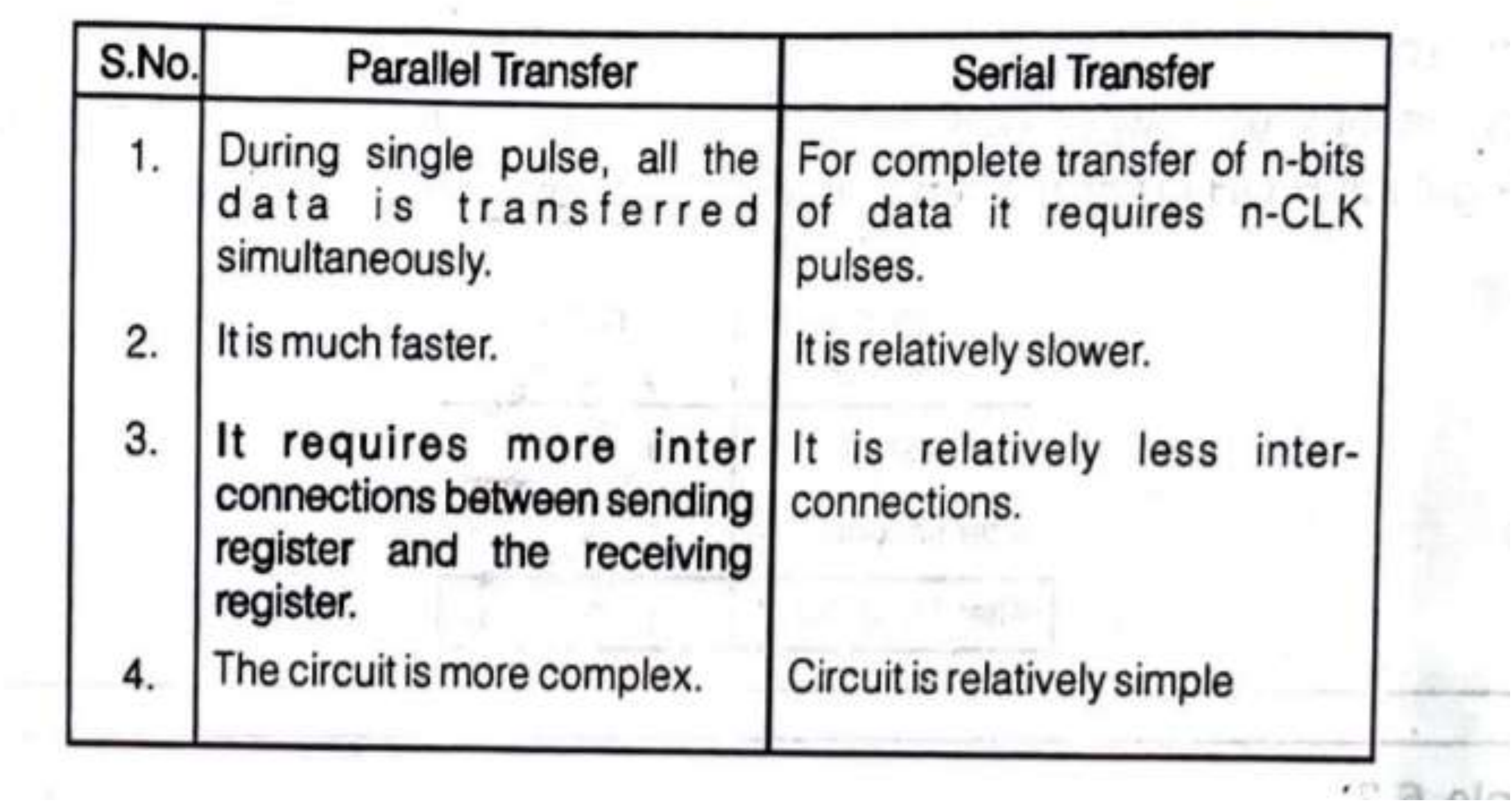
Such register which are capable of shifting the information (data) to right and left both is called "Bi-directional shift register".



Operation

When the Mode control M = 1, all the 'A' AND gates are enabled and the data at D_R is shifted to the righ when clock pulses are applied. When M = 0, the A gates are inhibited and B gates are enabled allowing th data at D_L to be shifted to the left. M should be changed only when CLK = 0, otherwise the data stored in th register may be altered.

Difference between Serial & Parallel Transfer



S.No.	Parallel Transfer	Serial Transfer
1.	During single pulse, all the data is transferred simultaneously.	For complete transfer of n-bits of data it requires n-CLK pulses.
2.	It is much faster.	It is relatively slower.
3.	It requires more inter connections between sending register and the receiving register.	It is relatively less inter-connections.
4.	The circuit is more complex.	Circuit is relatively simple

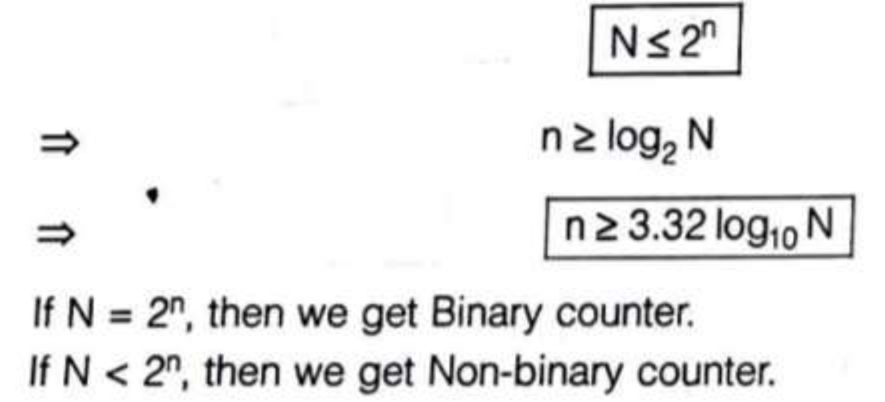
Application of Shift Register

- The primary use of shift registers are temporary data storage and bit manipulations. Despite of this other common applications are given below:

- Time Delay
- Data Conversion
- Ring Counter
- Sequence Generator
- Arithmetic Operation

COUNTERS

- The counters are composed with FF and combinational elements. It is a sequential circuit forming by the cascading of FF. A counter with n FF has maximum 2n possible states. Also, if N= total number of states and n = number of FF then,



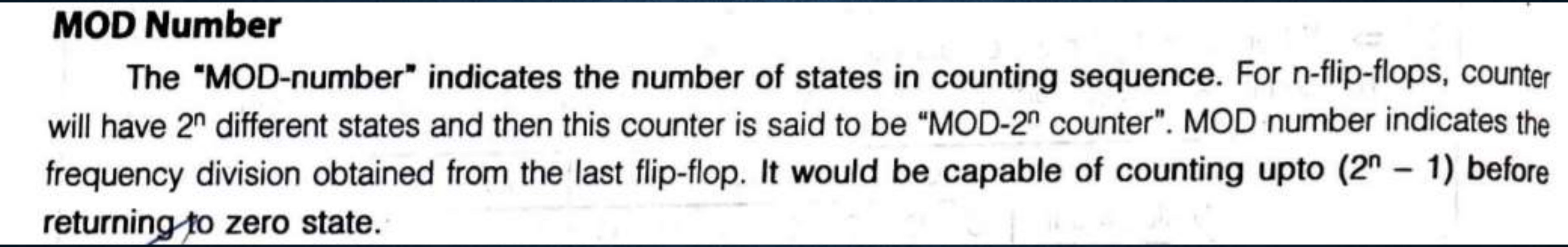
$$\boxed{N \leq 2^n}$$

$$\Rightarrow \quad n \geq \log_2 N$$

$$\Rightarrow \quad \boxed{n \geq 3.32 \log_{10} N}$$

If $N = 2^n$, then we get Binary counter.

If $N < 2^n$, then we get Non-binary counter.



MOD Number

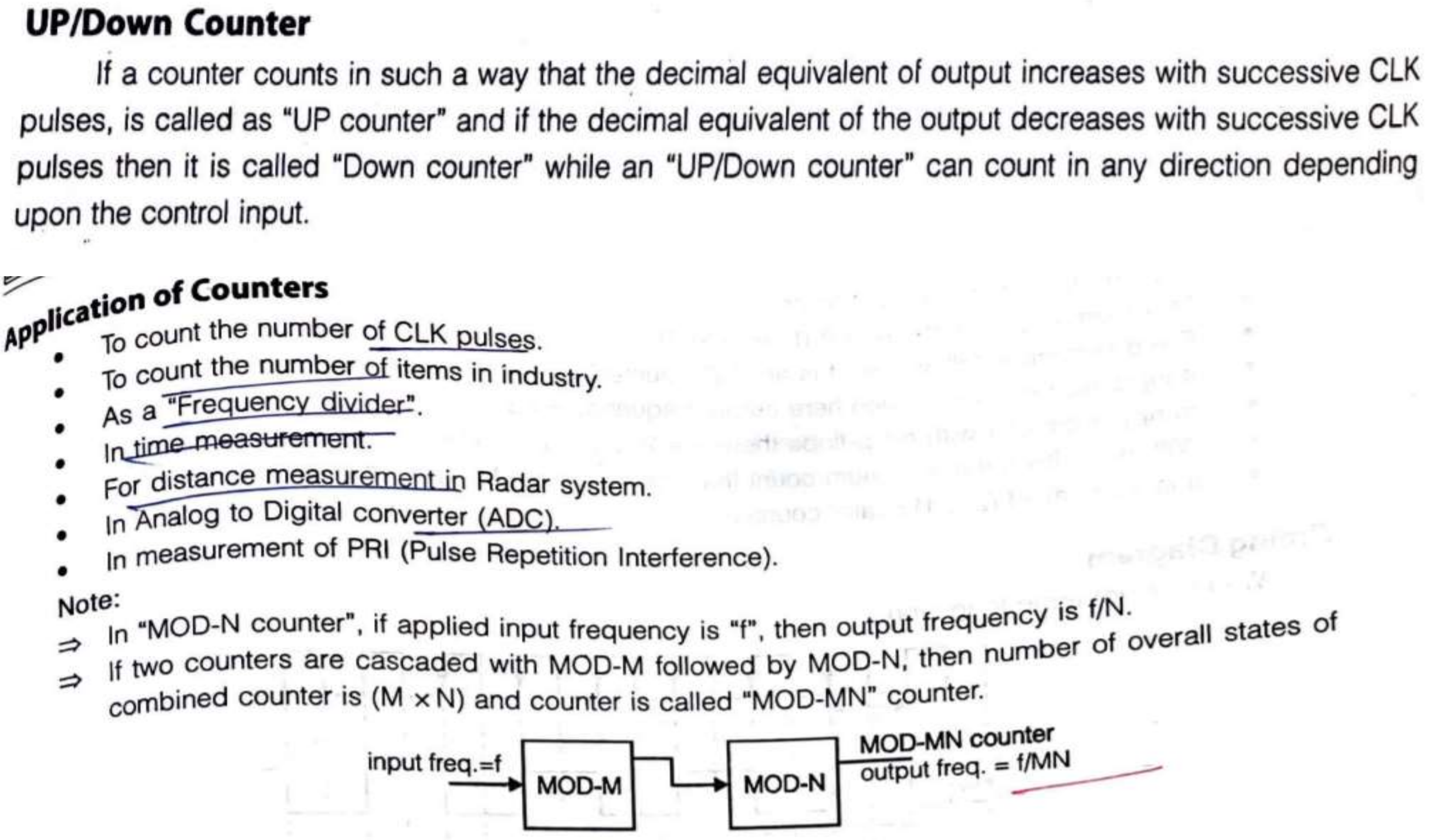
The "MOD-number" indicates the number of states in counting sequence. For n-flip-flops, counter will have 2^n different states and then this counter is said to be "MOD-2^n counter". MOD number indicates the frequency division obtained from the last flip-flop. It would be capable of counting upto $(2^n - 1)$ before returning to zero state.

Synchronous Counter vs Asynchronous Counter

Based upon the clock pulse, counters are of two type

Synchronous Counter	Asynchronous Counter
• All FF are triggered with same clock	• All FF are triggered with different clock
• Operation is faster	• Operation is slower
• Any required sequence can be designed	• Only fixed sequence can be designed
• No decoding error occurs	• Decoding error due to occurs
• Its design is complex	• Its design is relatively easy

Up/Down Counter & Application



UP/Down Counter

If a counter counts in such a way that the decimal equivalent of output increases with successive CLK pulses, is called as "UP counter" and if the decimal equivalent of the output decreases with successive CLK pulses then it is called "Down counter" while an "UP/Down counter" can count in any direction depending upon the control input.

Application of Counters

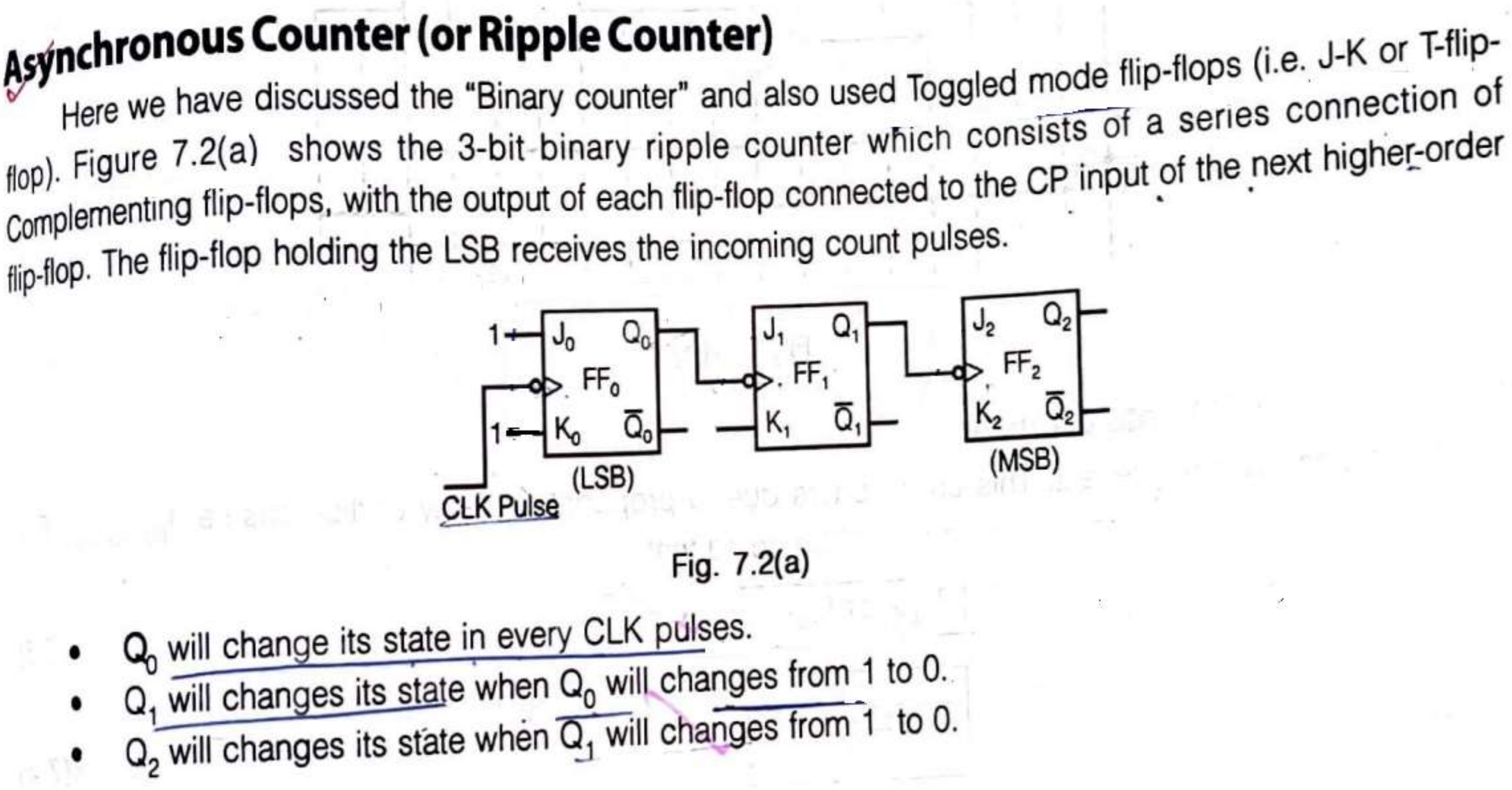
- To count the number of CLK pulses.
- To count the number of items in industry.
- As a "Frequency divider".
- In time measurement.
- For distance measurement in Radar system.
- In Analog to Digital converter (ADC).
- In measurement of PRI (Pulse Repetition Interference).

Note:

⇒ In "MOD-N counter", if applied input frequency is "f", then output frequency is f/N.

⇒ If two counters are cascaded with MOD-M followed by MOD-N, then number of overall states of combined counter is (M × N) and counter is called "MOD-MN" counter.

Asynchronous Counter (Ripple Counter)



Asynchronous Counter (or Ripple Counter)

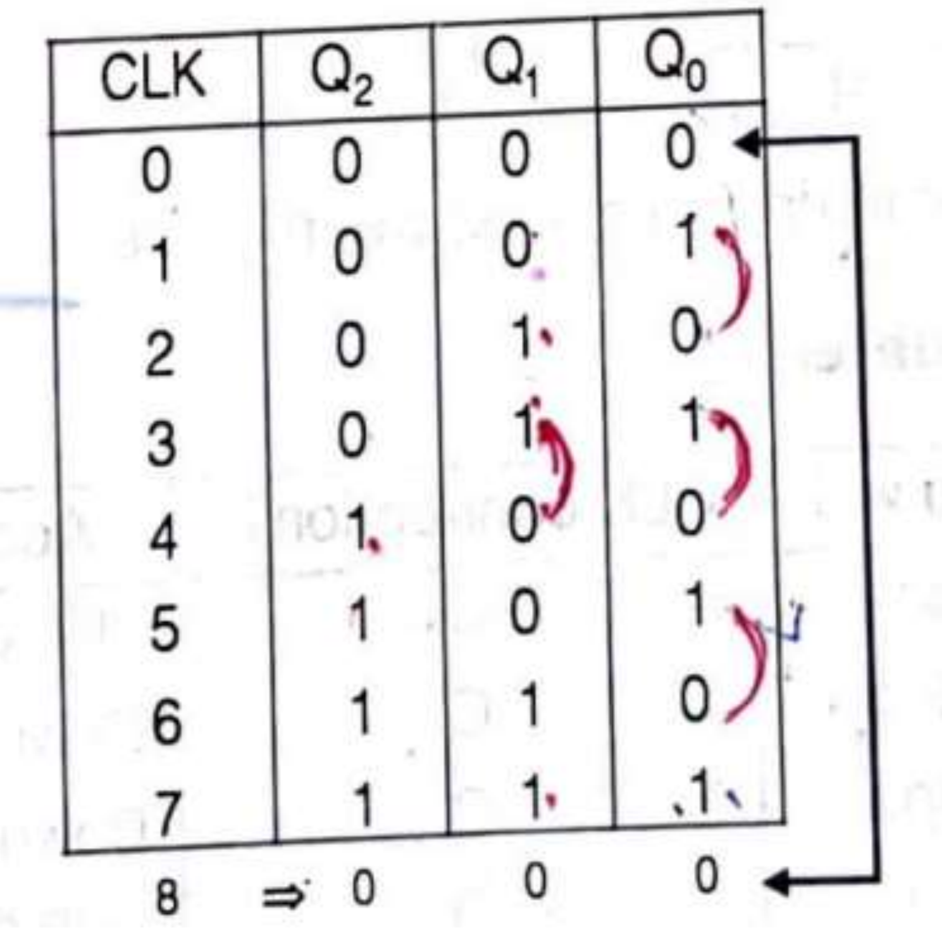
Here we have discussed the "Binary counter" and also used Toggled mode flip-flops (i.e. J-K or T-flip-flop). Figure 7.2(a) shows the 3-bit binary ripple counter which consists of a series connection of Complementing flip-flops, with the output of each flip-flop connected to the CP input of the next higher-order flip-flop. The flip-flop holding the LSB receives the incoming count pulses.

Fig. 7.2(a)

- Q_0 will change its state in every CLK pulses.
- Q_1 will changes its state when Q_0 will changes from 1 to 0.
- Q_2 will changes its state when $\overline{Q_1}$ will changes from 1 to 0.

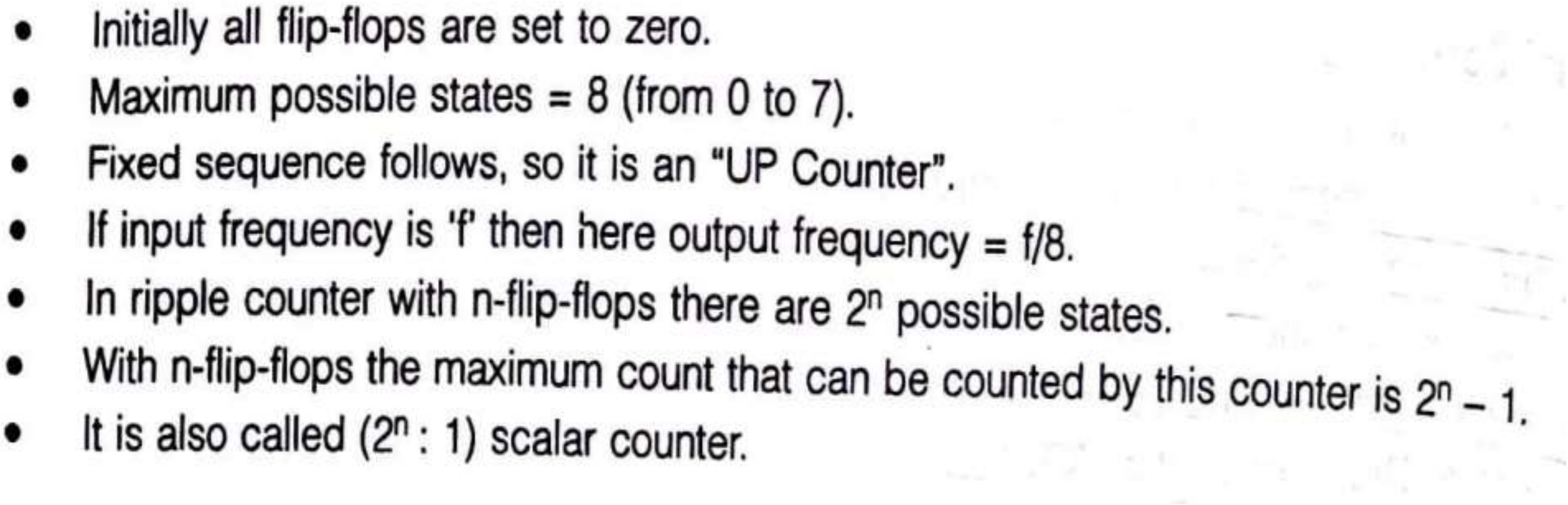
Asynchronous Counter (Ripple Counter)

Truth table:



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

Asynchronous Counter (Ripple Counter)



- Initially all flip-flops are set to zero.
- Maximum possible states = 8 (from 0 to 7).
- Fixed sequence follows, so it is an "UP Counter".
- If input frequency is 'f' then here output frequency = f/8.
- In ripple counter with n-flip-flops there are 2^n possible states.
- With n-flip-flops the maximum count that can be counted by this counter is $2^n - 1$.
- It is also called (2^n : 1) scalar counter.

Asynchronous Counter (Ripple Counter)

Timing Diagram

We use (–ve) edge triggering.

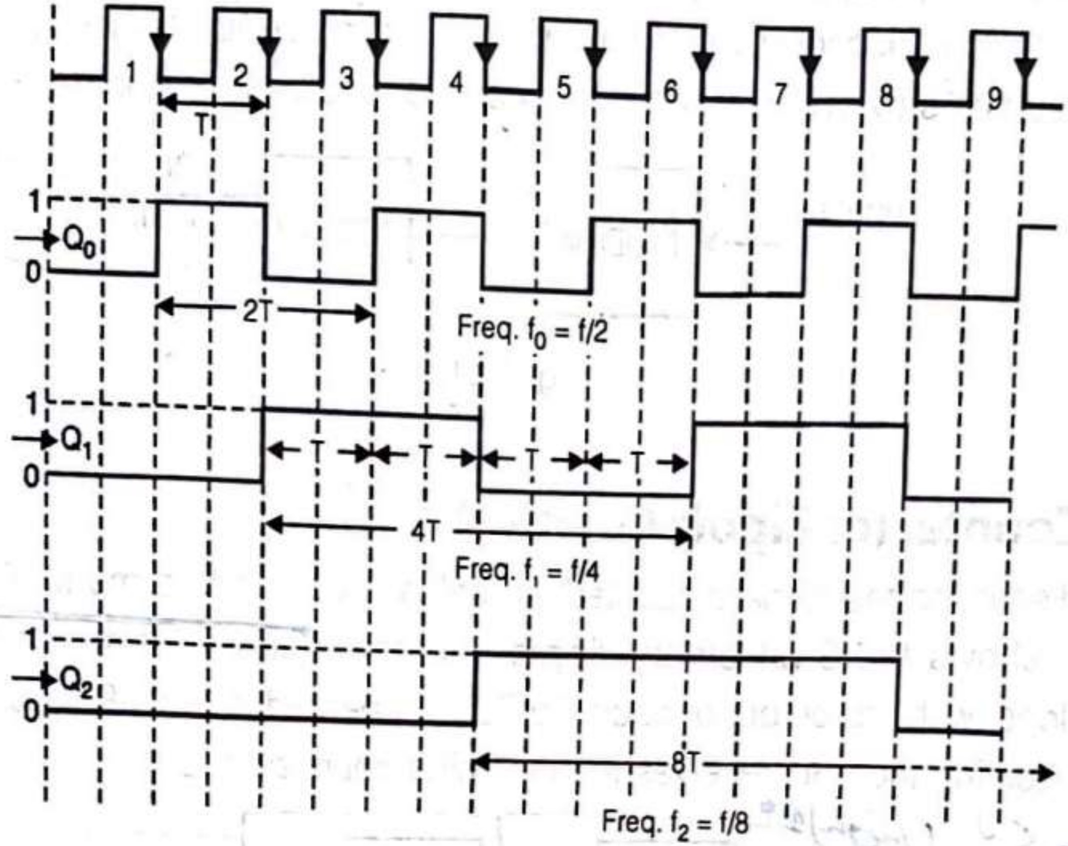
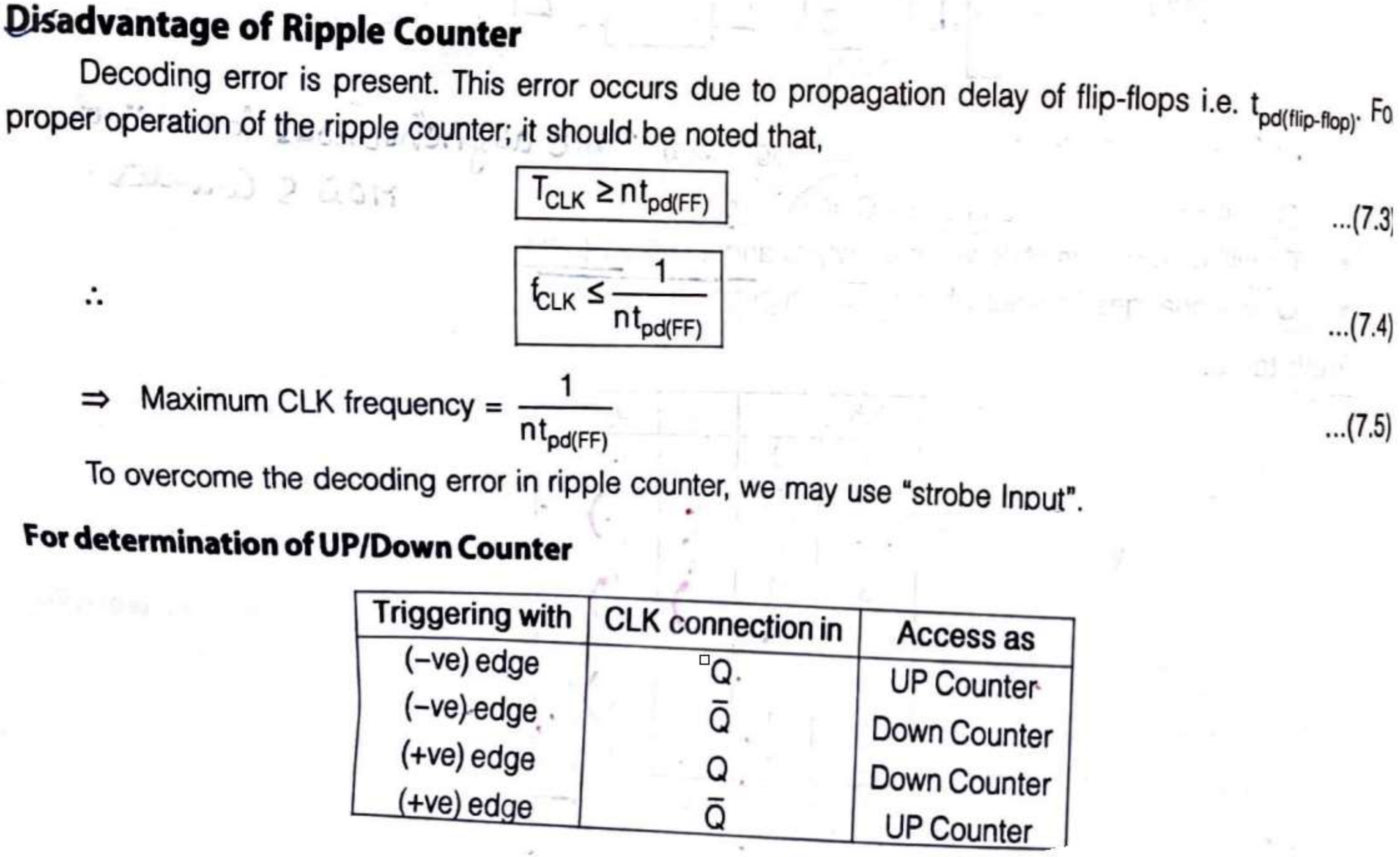


Fig. 7.2(b)

Asynchronous Counter (Ripple Counter)



Disadvantage of Ripple Counter

Decoding error is present. This error occurs due to propagation delay of flip-flops i.e. $t_{pd(flip\text{-}flop)}$. Fo proper operation of the ripple counter; it should be noted that,

$$\boxed{T_{CLK} \geq nt_{pd(FF)}} \quad ...(7.3)$$

$$\therefore \quad \boxed{f_{CLK} \leq \frac{1}{nt_{pd(FF)}}} \quad ...(7.4)$$

$$\Rightarrow \quad \text{Maximum CLK frequency} = \frac{1}{nt_{pd(FF)}} \quad ...(7.5)$$

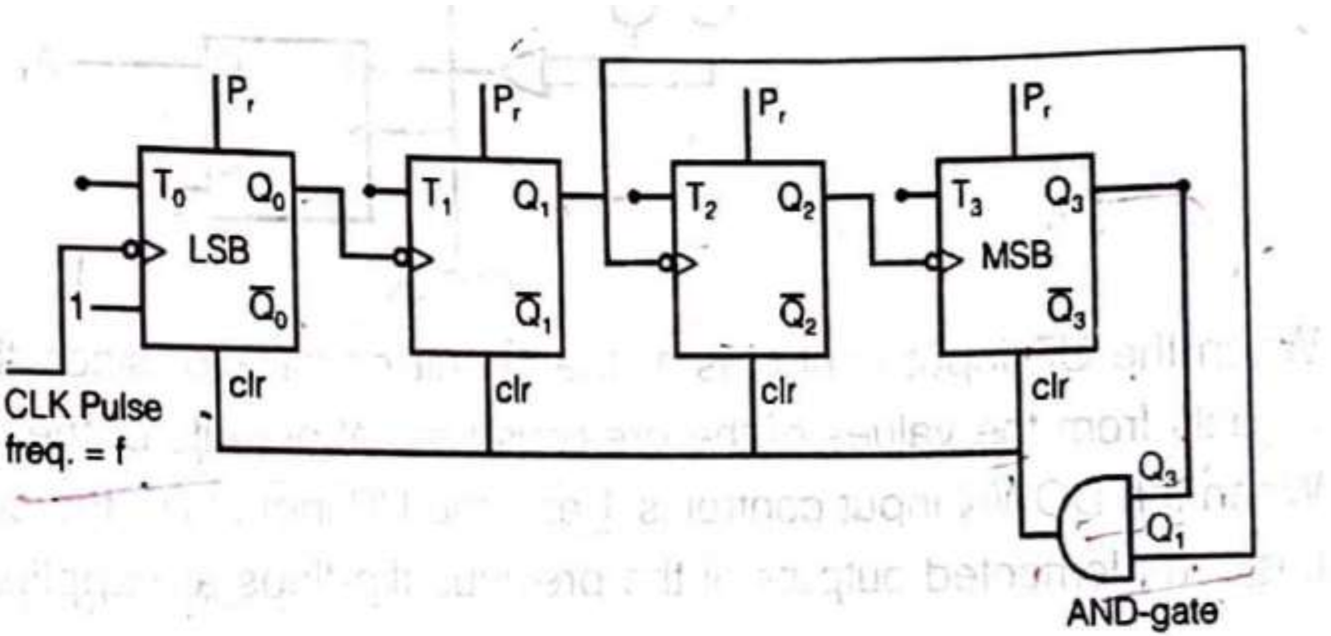
To overcome the decoding error in ripple counter, we may use "strobe Input".

For determination of UP/Down Counter

Triggering with	CLK connection in	Access as
(−ve) edge	Q	UP Counter
(−ve) edge	$\bar{Q}$	Down Counter
(+ve) edge	Q	Down Counter
(+ve) edge	$\bar{Q}$	UP Counter

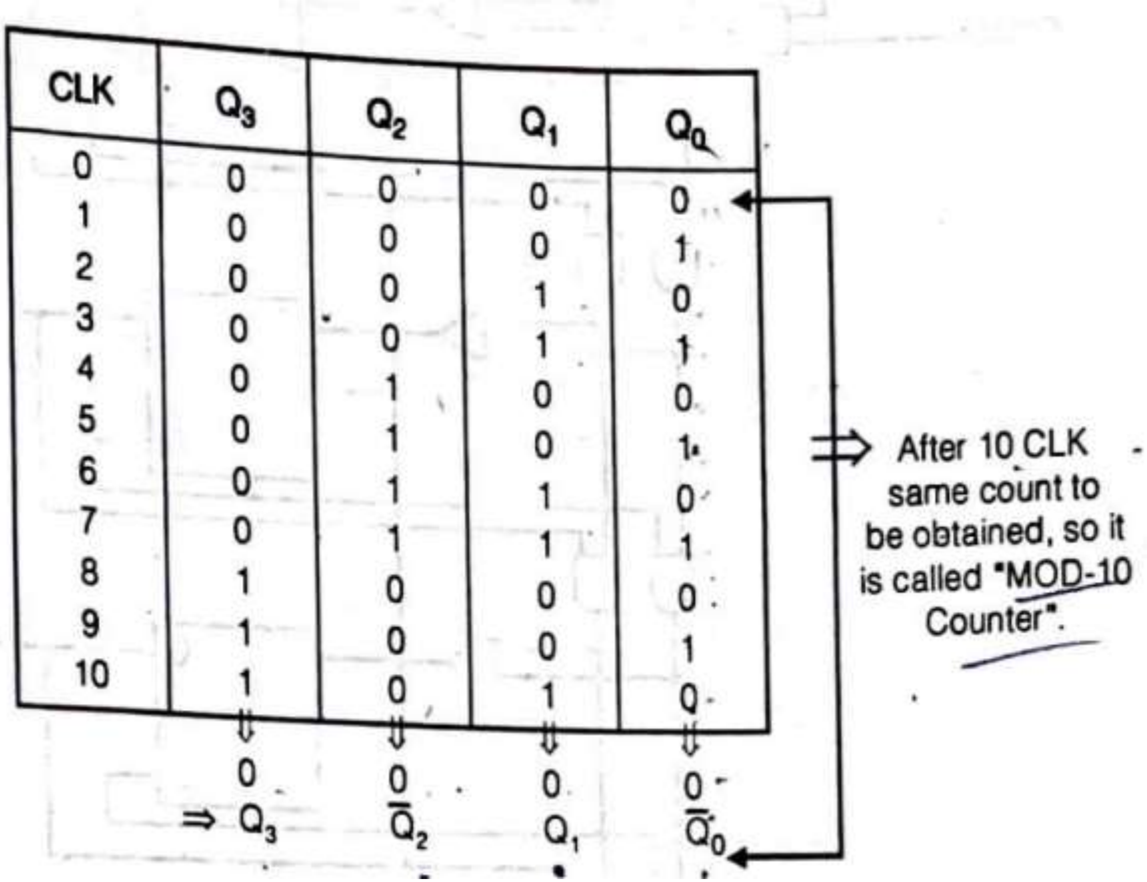
Non Binary Ripple Counter
Decade Counter
Mod-10 Counter

- Decade counter require four T FF. As we know that with 4 FF, there will be 16-states in total; so here used states are equal to 10 while remaining 6 states are unused.



Cont…

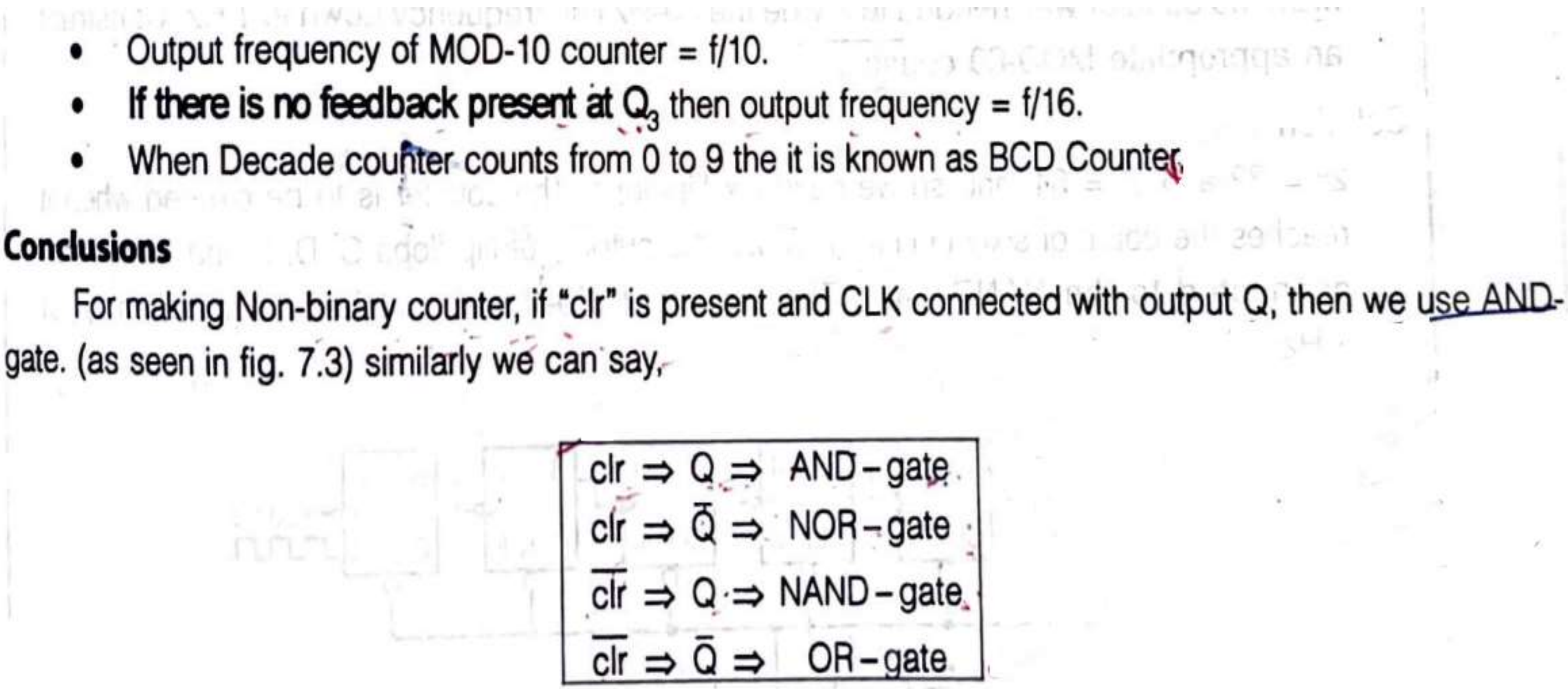
- **Truth Table**



CLK	Q_3	Q_2	Q_1	Q_0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
	⇓	⇓	⇓	⇓
	0	0	0	0
⇒	Q_3	$\overline{Q}_2$	Q_1	$\overline{Q}_0$

⇒ After 10 CLK same count to be obtained, so it is called "MOD-10 Counter".

Cont…



- Output frequency of MOD-10 counter = f/10.
- **If there is no feedback present at Q_3** then output frequency = f/16.
- When Decade counter counts from 0 to 9 the it is known as BCD Counter.

Conclusions

For making Non-binary counter, if "clr" is present and CLK connected with output Q, then we use AND-gate. (as seen in fig. 7.3) similarly we can say,

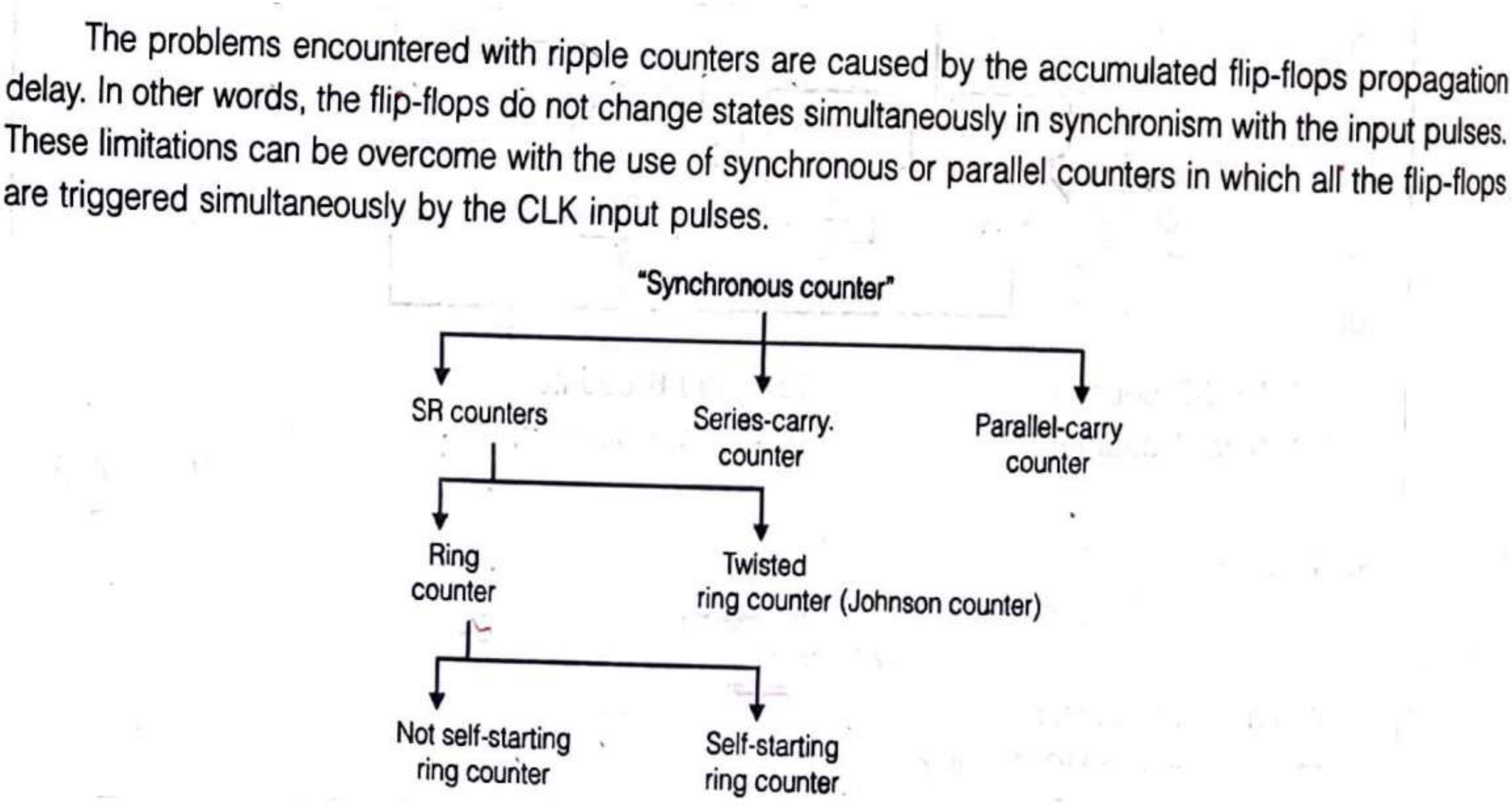
$\text{clr} \Rightarrow Q \Rightarrow$ AND – gate

$\text{clr} \Rightarrow \bar{Q} \Rightarrow$ NOR – gate

$\overline{\text{clr}} \Rightarrow Q \Rightarrow$ NAND – gate

$\overline{\text{clr}} \Rightarrow \bar{Q} \Rightarrow$ OR – gate

Synchronous (Parallel) Counter



The problems encountered with ripple counters are caused by the accumulated flip-flops propagation delay. In other words, the flip-flops do not change states simultaneously in synchronism with the input pulses. These limitations can be overcome with the use of synchronous or parallel counters in which all the flip-flops are triggered simultaneously by the CLK input pulses.

"Synchronous counter"

SR counters

Series-carry. counter

Parallel-carry counter

Ring counter

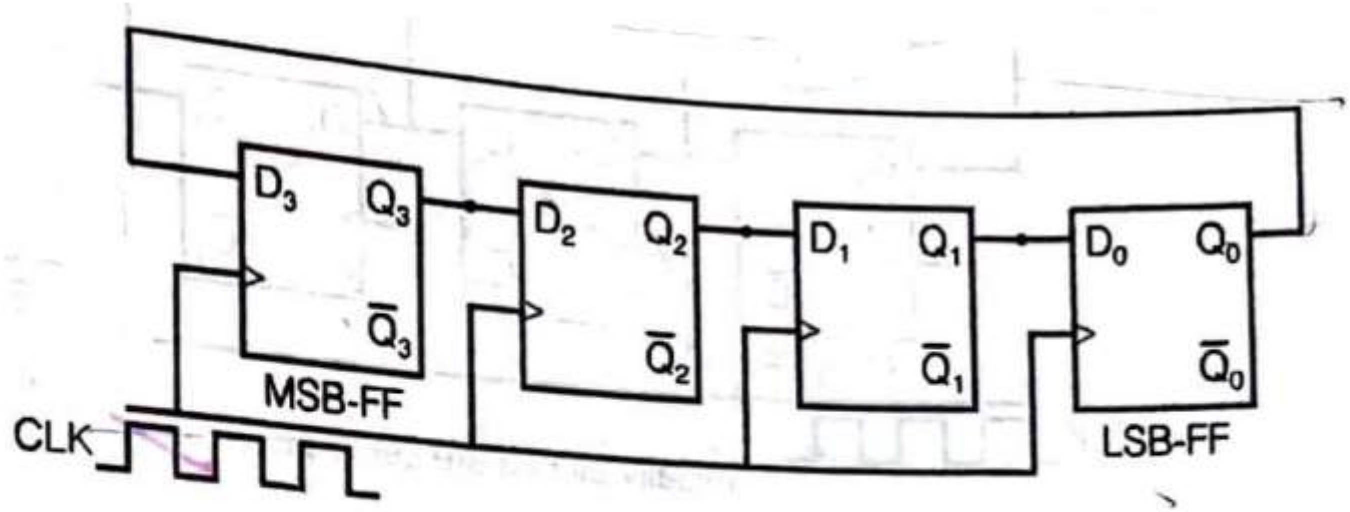
Twisted ring counter (Johnson counter)

Not self-starting ring counter

Self-starting ring counter

Not Self Starting Ring Counter

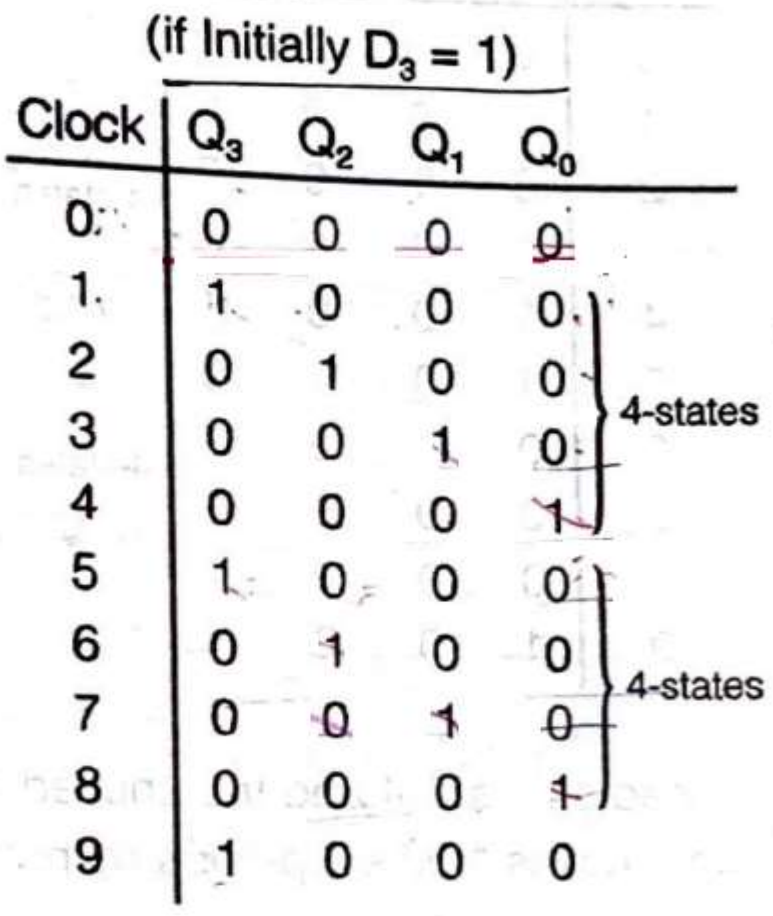
- It is nothing but SISO shift register. It is also known as “End-Carry Counter” and is a synchronous counter



- In this counter LSB FF output “Q0” is connected to the MSB FF input (D3). In this only one bit is high and circulates among all the FF

Not Self Starting Ring Counter

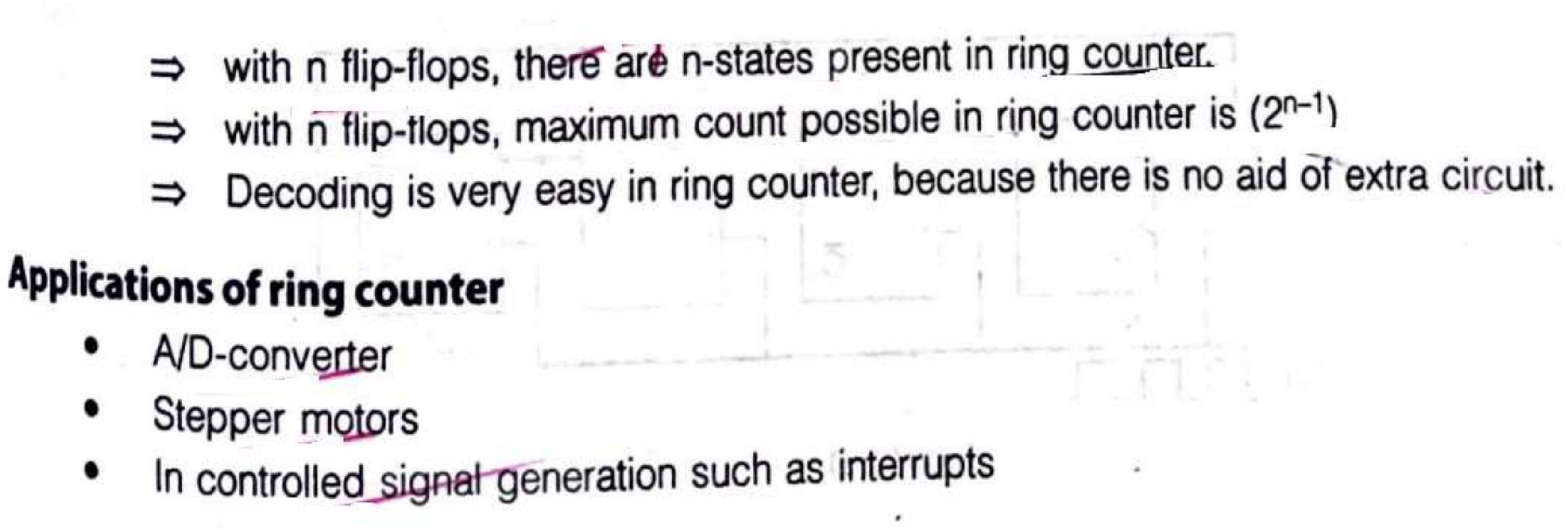
Truth Table



(if Initially $D_3 = 1$)

Clock	Q_3	Q_2	Q_1	Q_0	
0	0	0	0	0	
1.	1	0	0	0	4-states
2	0	1	0	0	
3	0	0	1	0	
4	0	0	0	1	
5	1	0	0	0	4-states
6	0	1	0	0	
7	0	0	1	0	
8	0	0	0	1	
9	1	0	0	0	

Not Self Starting Ring Counter



⇒ with n flip-flops, there are n-states present in ring counter.

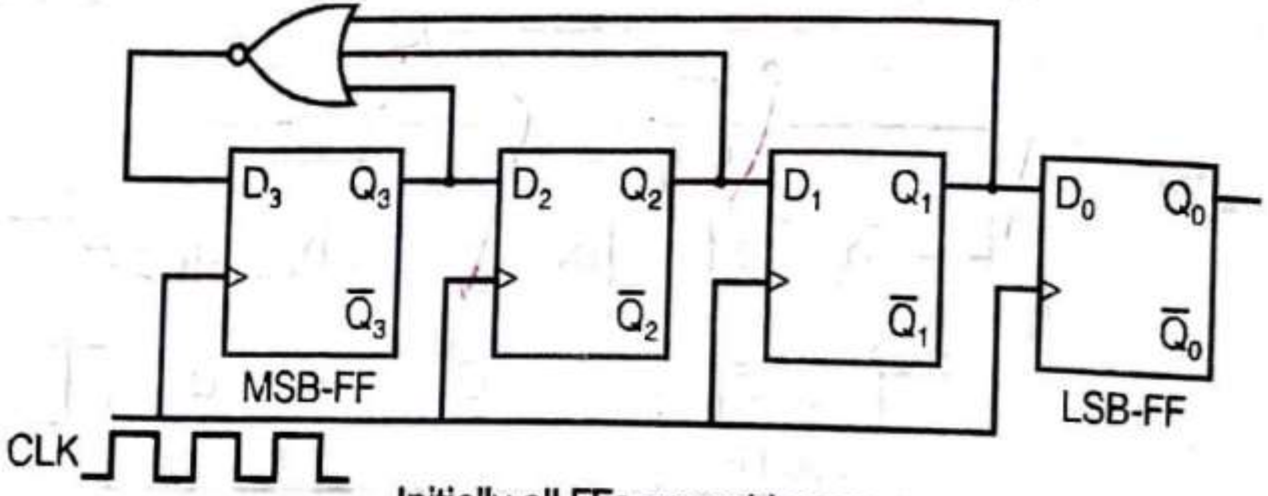
⇒ with n flip-flops, maximum count possible in ring counter is (2^{n-1})

⇒ Decoding is very easy in ring counter, because there is no aid of extra circuit.

Applications of ring counter

- A/D-converter
- Stepper motors
- In controlled signal generation such as interrupts

Self Starting Ring Counter



Initially all FFs are set to zero

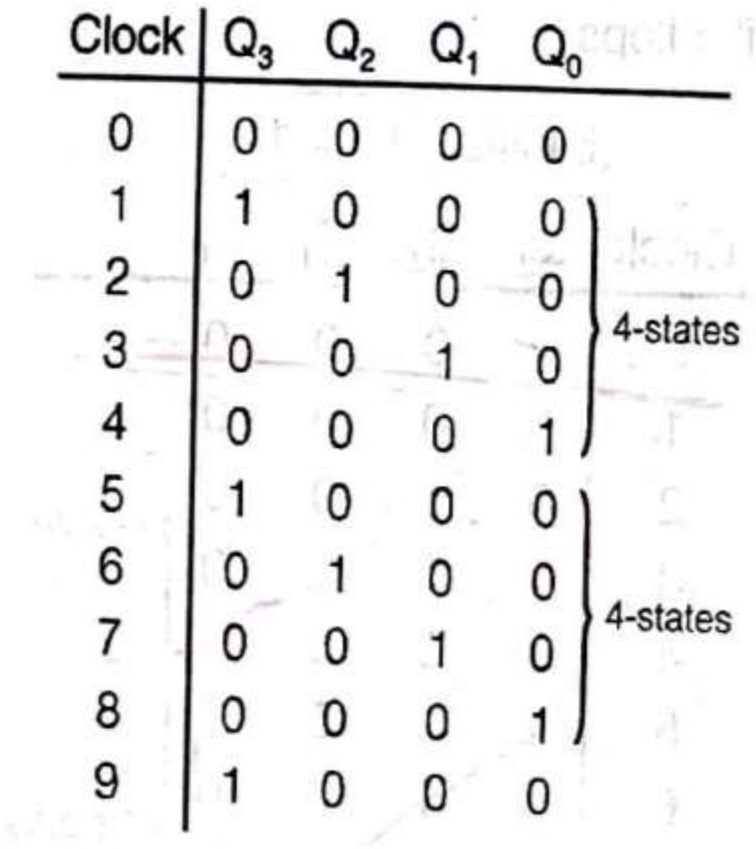
⇒ In 4-bit ring counter the used states = 4 and the unused input = $2^4 - 4 = 12$.

⇒ In any counter if CLK frequency is “f” the flip-flops output frequency is “f/N” (where N = No. of states).

⇒ This counter is also called “And carry counter”.

Self Starting Ring Counter

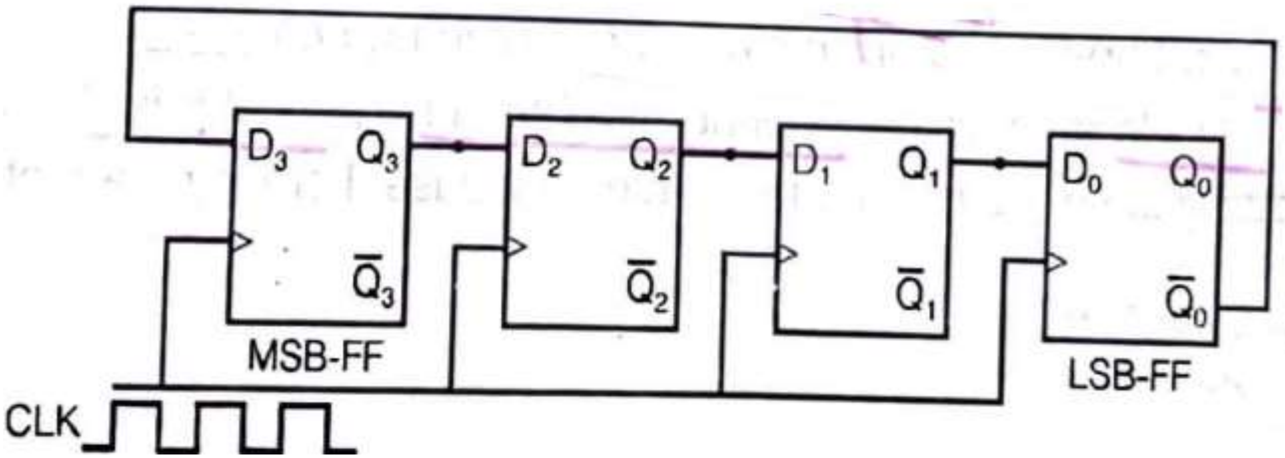
Truth Table



Clock	Q_3	Q_2	Q_1	Q_0	
0	0	0	0	0	
1	1	0	0	0	4-states
2	0	1	0	0	
3	0	0	1	0	
4	0	0	0	1	
5	1	0	0	0	4-states
6	0	1	0	0	
7	0	0	1	0	
8	0	0	0	1	
9	1	0	0	0	

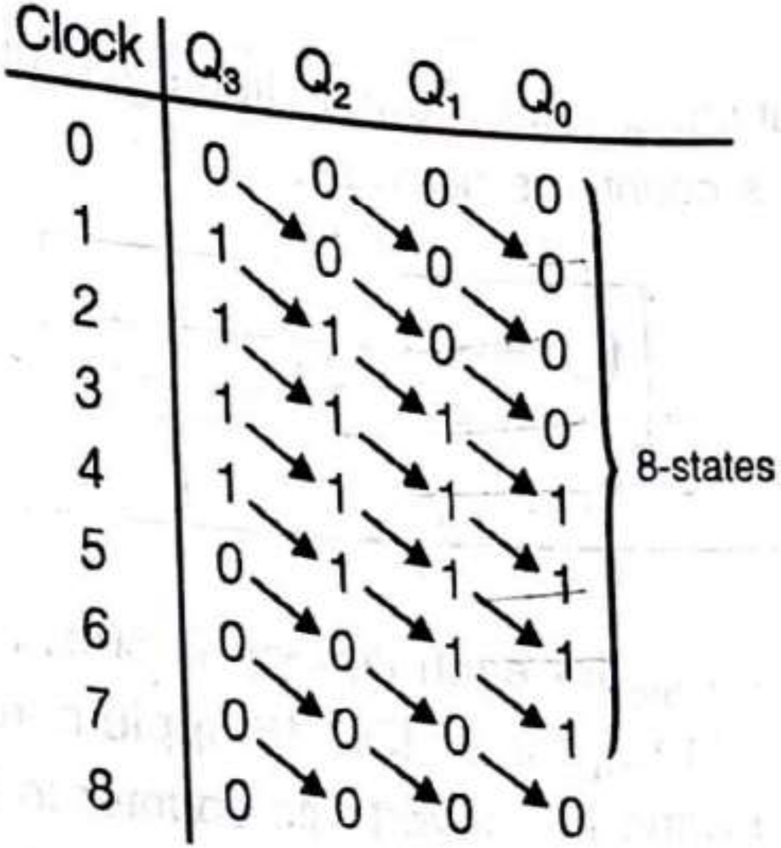
Twisted Ring Counter

- This counter is nothing but a shift register which is also known as:
- Johnson Counter
- Switch Tail Ring Counter
- Mobiles Counter
- Creeping Counter



Twisted Ring Counter

Truth Table



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

8-states

Twisted Ring Counter

- With n FF there are 2n states in this counter

- With n FF the maximum count by this counter is (2n-1)

- In normal **“Johnson Counter”** with n FF and the frequency is “f” then output frequency of FF is **“f/2n”.** (where N= No. of states = 2n).

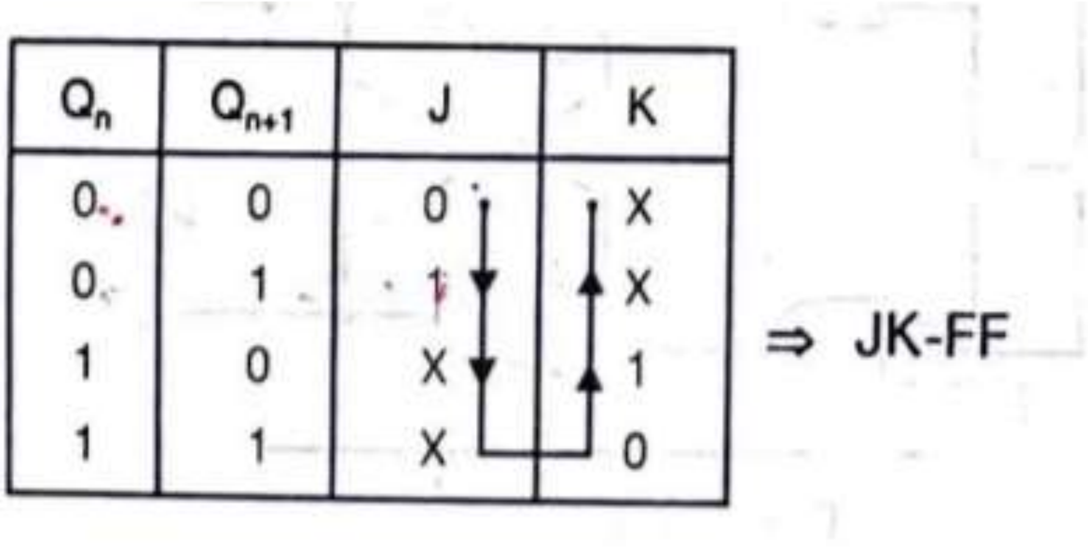
- When a counter is entered into the unused state and counter is function in only unused states then this counter is called **“Lock out Stage Counter”.** In a counter if a feedback is used the number of possible states will decreases.

Steps to Design Synchronous Counter

1. Identify number of bits and FF
2. Write Excitation Table of FF
3. Make State Diagram and State Table
4. Solve Boolean Expression
5. Make Circuit

2 Bits Synchronous UP/Down Counter using JK FF

- Step 1: n=2 bits, FF = JK FF
- Step 2: Excitation Table of JK FF

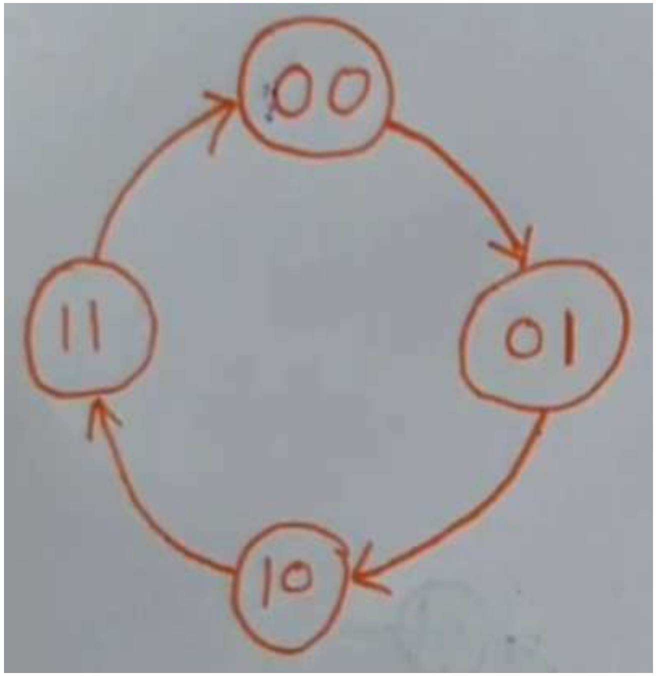


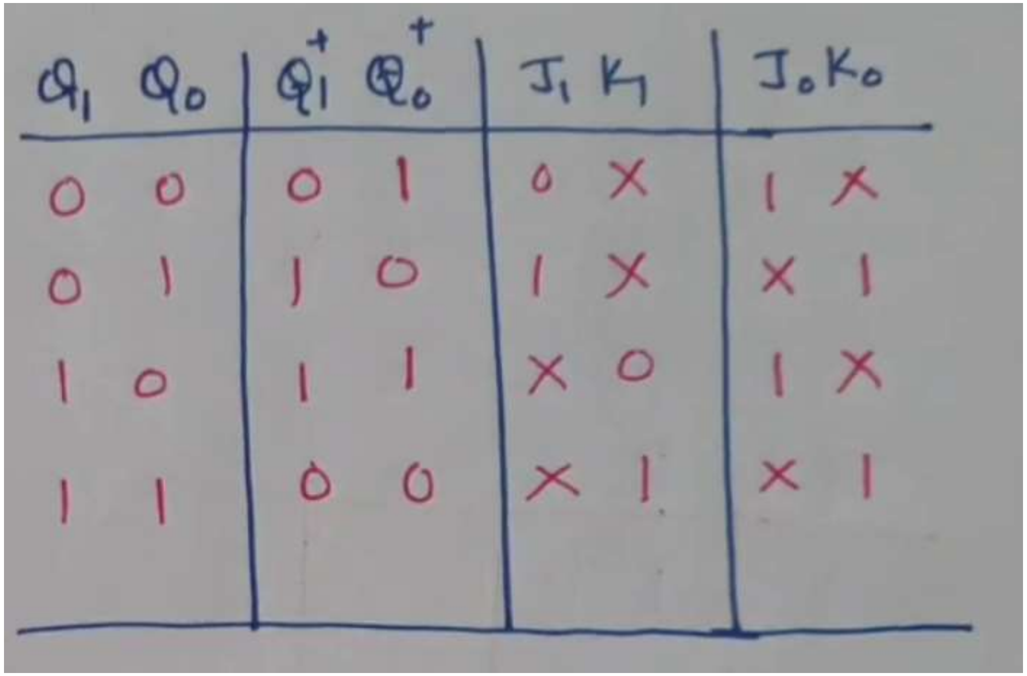
Q_n	Q_{n+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

⇒ JK-FF

2 Bits Synchronous UP/Down Counter using JK FF

- **Step 3:** Make State Diagram and State Table

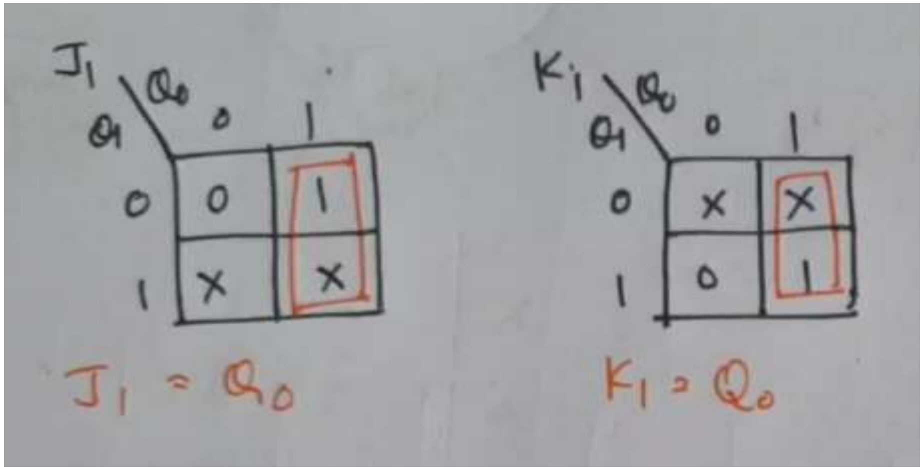




Q_1	Q_0	Q_1^+	Q_0^+	J_1	K_1	J_0	K_0
0	0	0	1	0	X	1	X
0	1	1	0	1	X	X	1
1	0	1	1	X	0	1	X
1	1	0	0	X	1	X	1

2 Bits Synchronous UP/Down Counter using JK FF

- **Step 4:** Find Boolean Expression



J_1

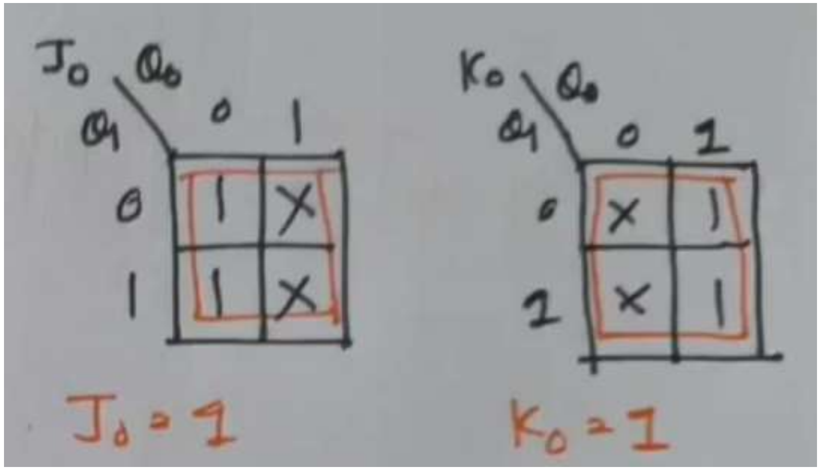
$Q_1 \backslash Q_0$	0	1
0	0	1
1	X	X

$J_1 = Q_0$

K_1

$Q_1 \backslash Q_0$	0	1
0	X	X
1	0	1

$K_1 = Q_0$



J_0

$Q_1 \backslash Q_0$	0	1
0	1	X
1	1	X

$J_0 = 1$

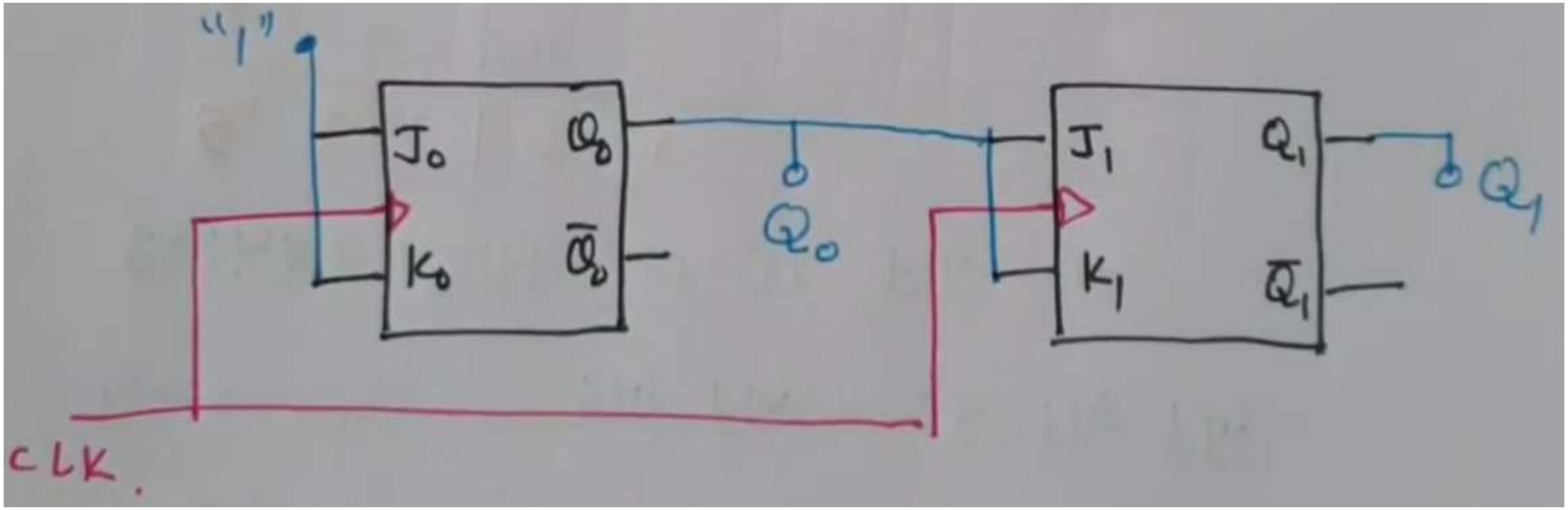
K_0

$Q_1 \backslash Q_0$	0	1
0	X	1
1	X	1

$K_0 = 1$

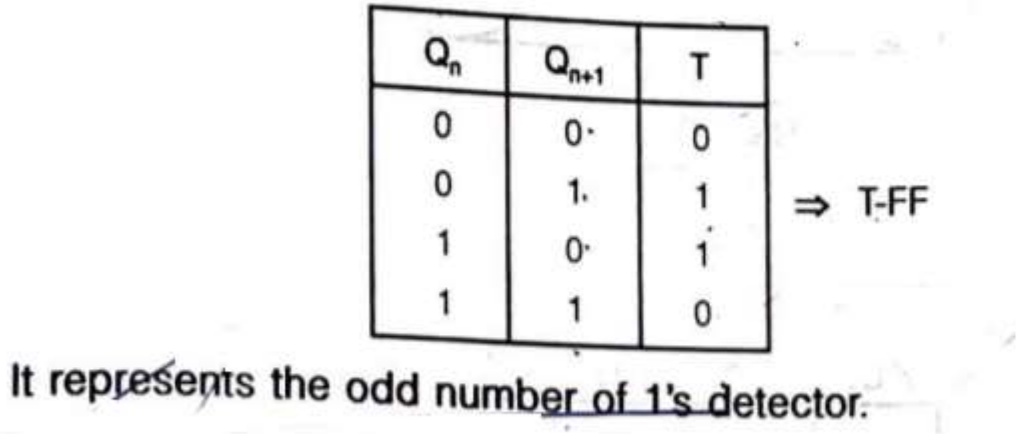
2 Bits Synchronous UP/Down Counter using JK FF

- **Step 5: Final Circuit Diagram**



3 Bits Synchronous UP/Down Counter using T FF

- Step 1: n= 3 bits, FF = T FF
- Step 2: Excitation Table of T FF



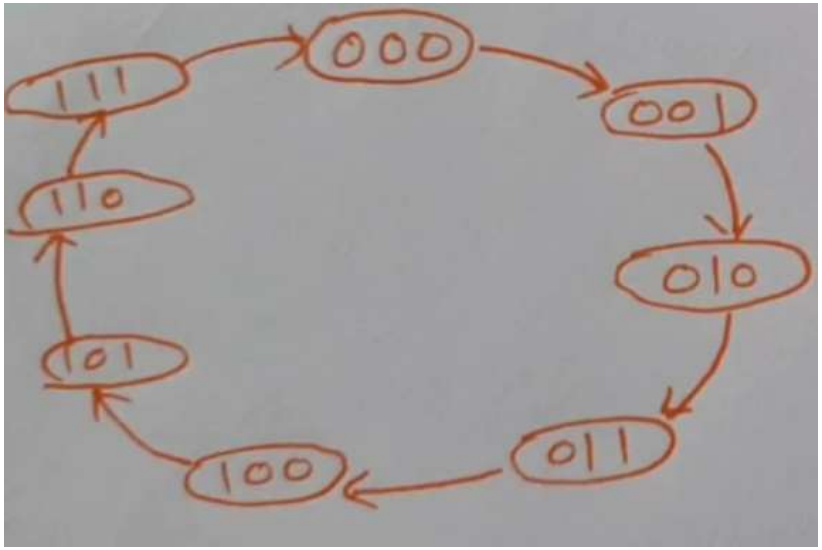
Q_n	Q_{n+1}	T
0	0	0
0	1	1
1	0	1
1	1	0

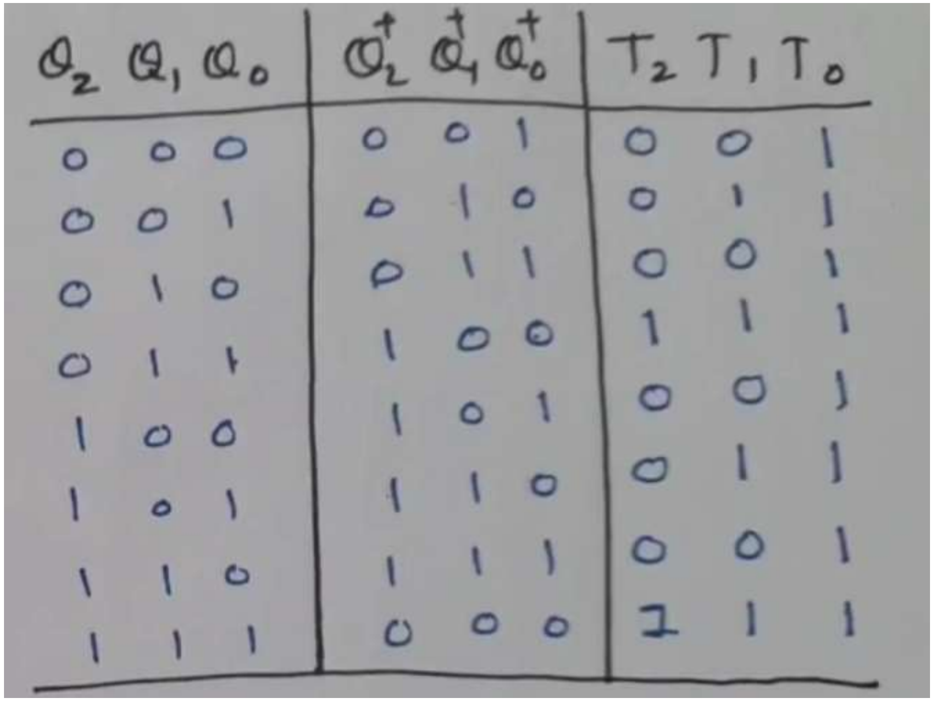
⇒ T-FF

It represents the odd number of 1's detector.

3 Bits Synchronous UP/Down Counter using T FF

- **Step 3:** Make State Diagram and State Table

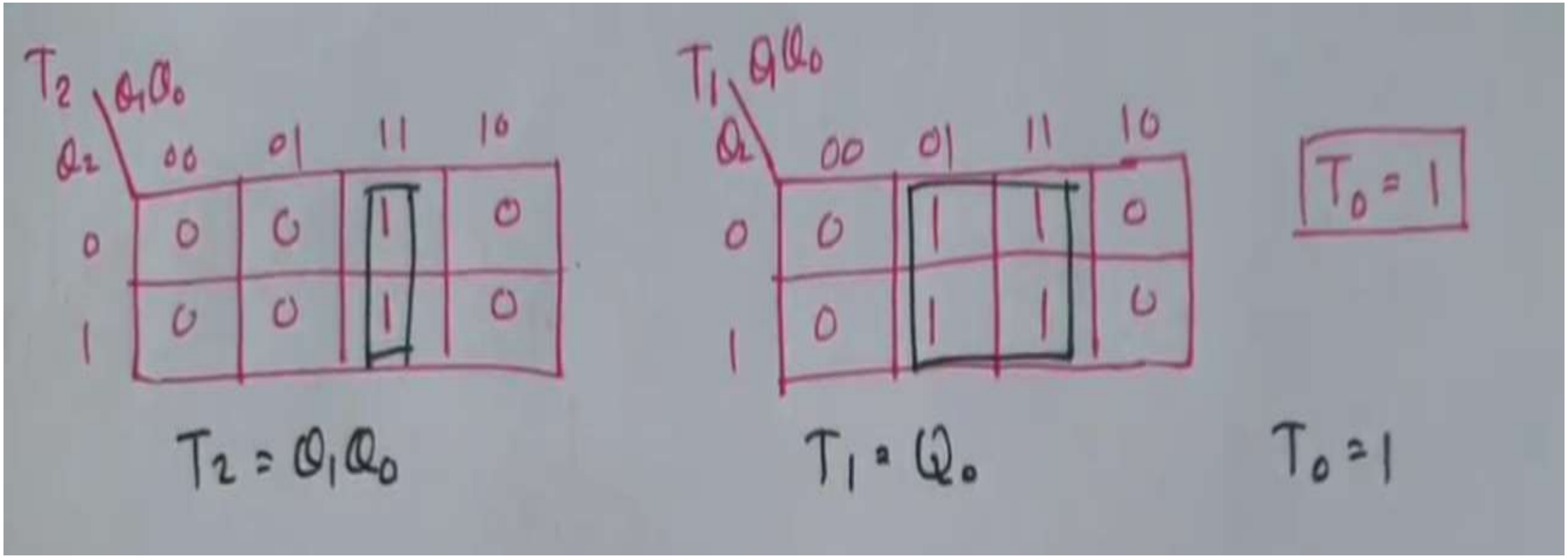




Q_2	Q_1	Q_0	Q_2^+	Q_1^+	Q_0^+	T_2	T_1	T_0
0	0	0	0	0	1	0	0	1
0	0	1	0	1	0	0	1	1
0	1	0	0	1	1	0	0	1
0	1	1	1	0	0	1	1	1
1	0	0	1	0	1	0	0	1
1	0	1	1	1	0	0	1	1
1	1	0	1	1	1	0	0	1
1	1	1	0	0	0	1	1	1

3 Bits Synchronous UP/Down Counter using T FF

- **Step 4:** Find Boolean Expression



T_2

$Q_2 \backslash Q_1Q_0$	00	01	11	10
0	0	0	1	0
1	0	0	1	0

T_1

$Q_2 \backslash Q_1Q_0$	00	01	11	10
0	0	1	1	0
1	0	1	1	0

$\boxed{T_0 = 1}$

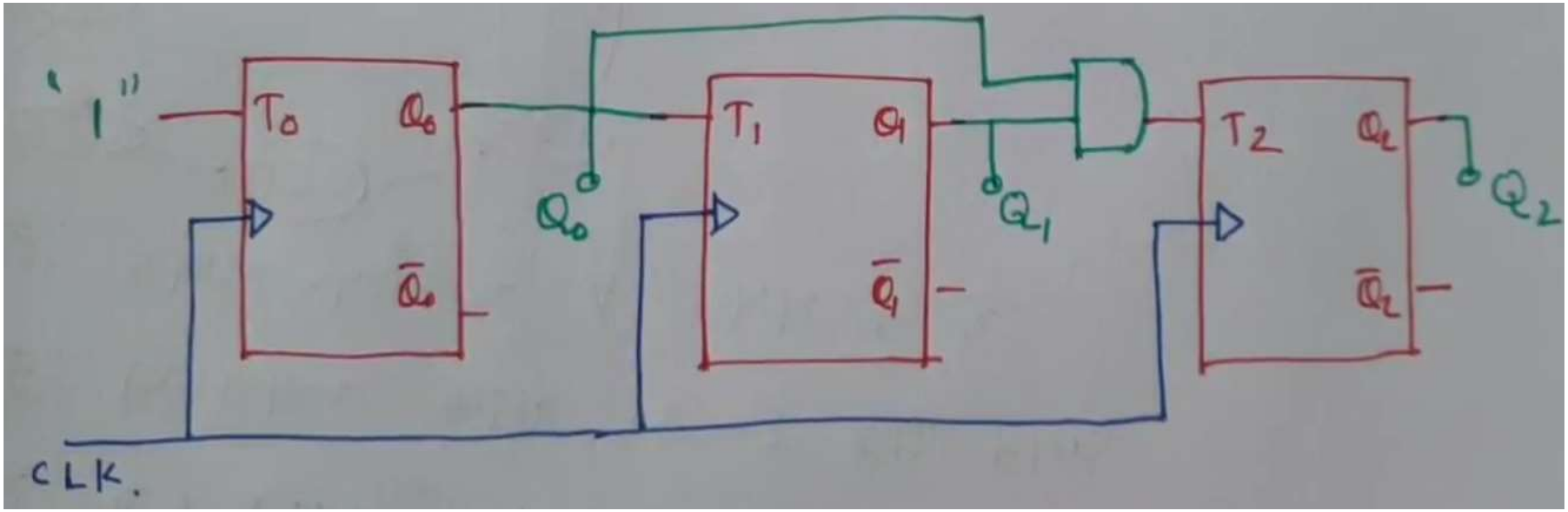
$T_2 = Q_1Q_0$

$T_1 = Q_0$

$T_0 = 1$

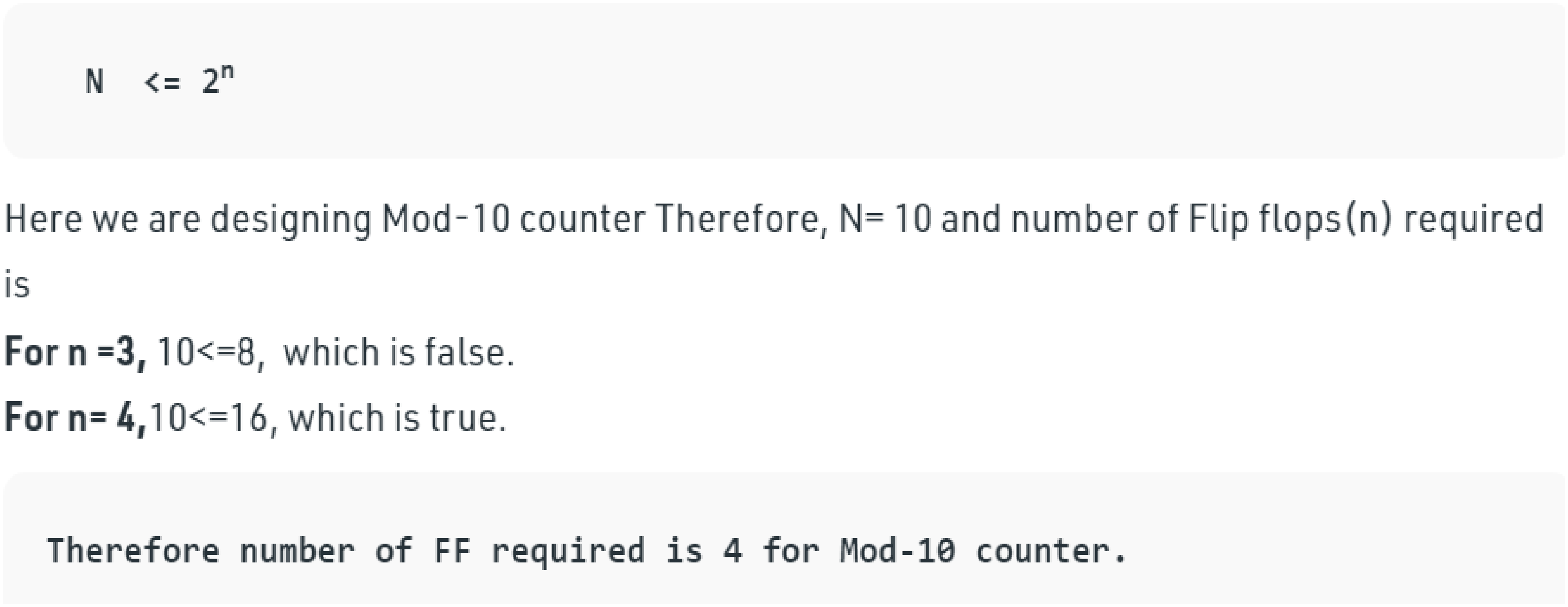
3 Bits Synchronous UP/Down Counter using T FF

- **Step 5: Final Circuit Diagram**



Design Mod – N synchronous Counter

- **Step 1 : Decision for number of flip-flops –**
  Example : If we are designing mod N counter and n number of flip-flops are required then n can be found out by this equation.



```
N  <= 2^n
```

Here we are designing Mod-10 counter Therefore, N= 10 and number of Flip flops(n) required is

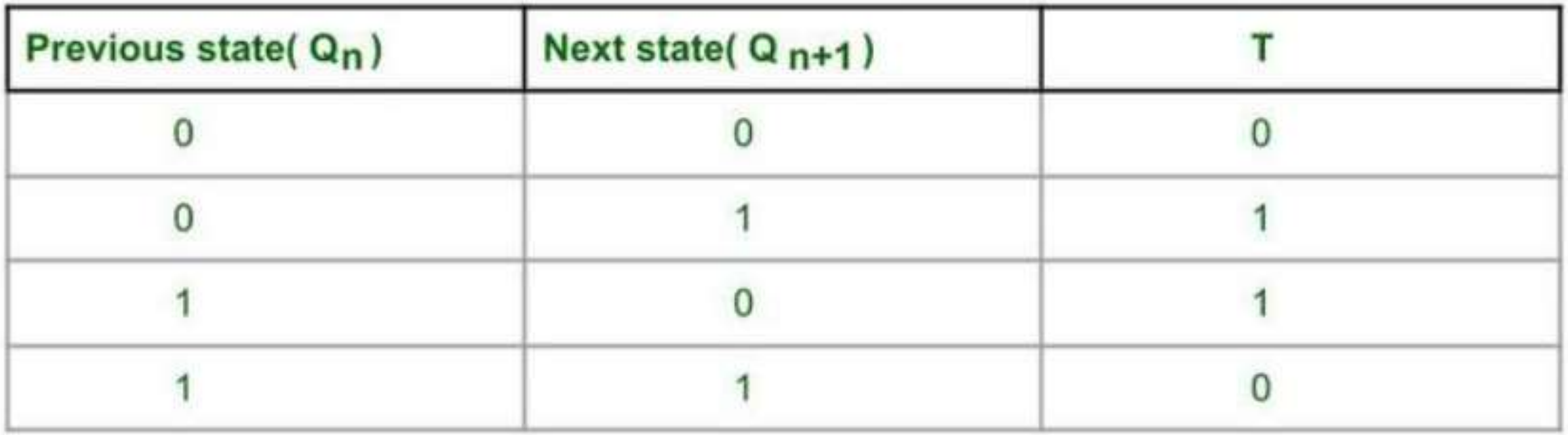
For n =3, 10<=8, which is false.

For n= 4,10<=16, which is true.

```
Therefore number of FF required is 4 for Mod-10 counter.
```

Design Mod – N synchronous Counter

- **Step 2 : Write excitation table of Flip flops –** Here T FF is used

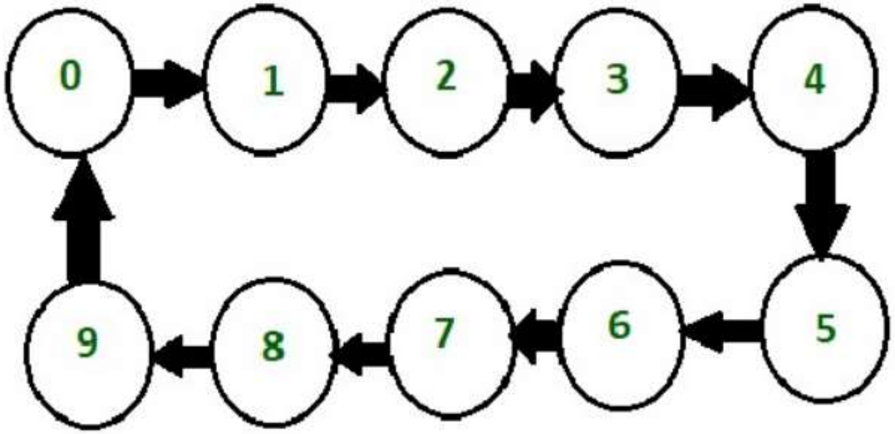


Previous state(Q_n)	Next state(Q_{n+1})	T
0	0	0
0	1	1
1	0	1
1	1	0

Excitation table of T FF.

Design Mod – N synchronous Counter

- **Step 3: Draw state diagram**

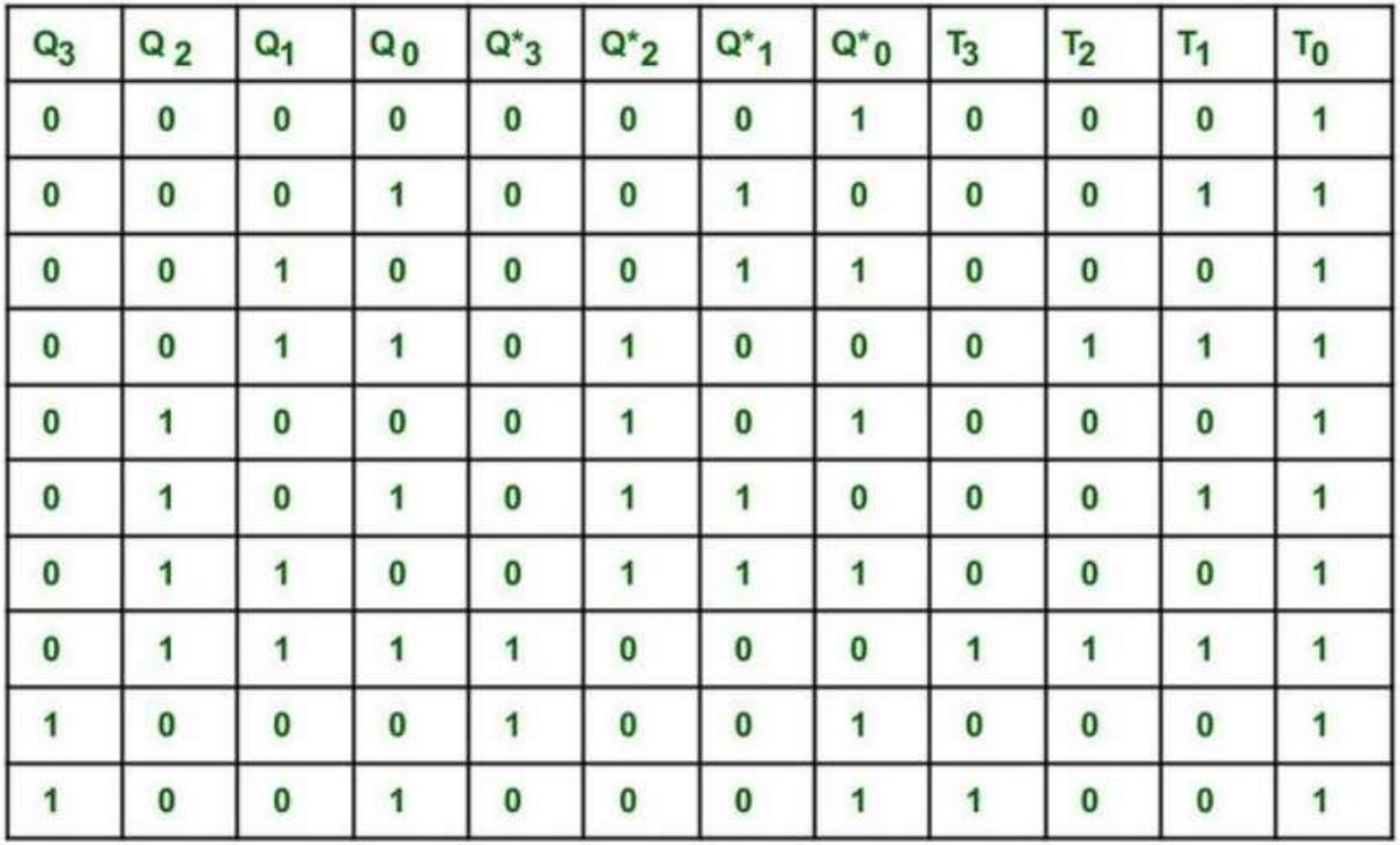


Counting Sequence of Decade counter

- A decade counter is called as mod -10 or divide by 10 counter. It counts from 0 to 9 and again reset to 0. It counts in natural binary sequence. Here 4 T Flip flops are used. It resets after Q_3 Q_2 Q_1 Q_0 = 1001.

Design Mod – N synchronous Counter

- **Step 3: Circuit excitation table:**

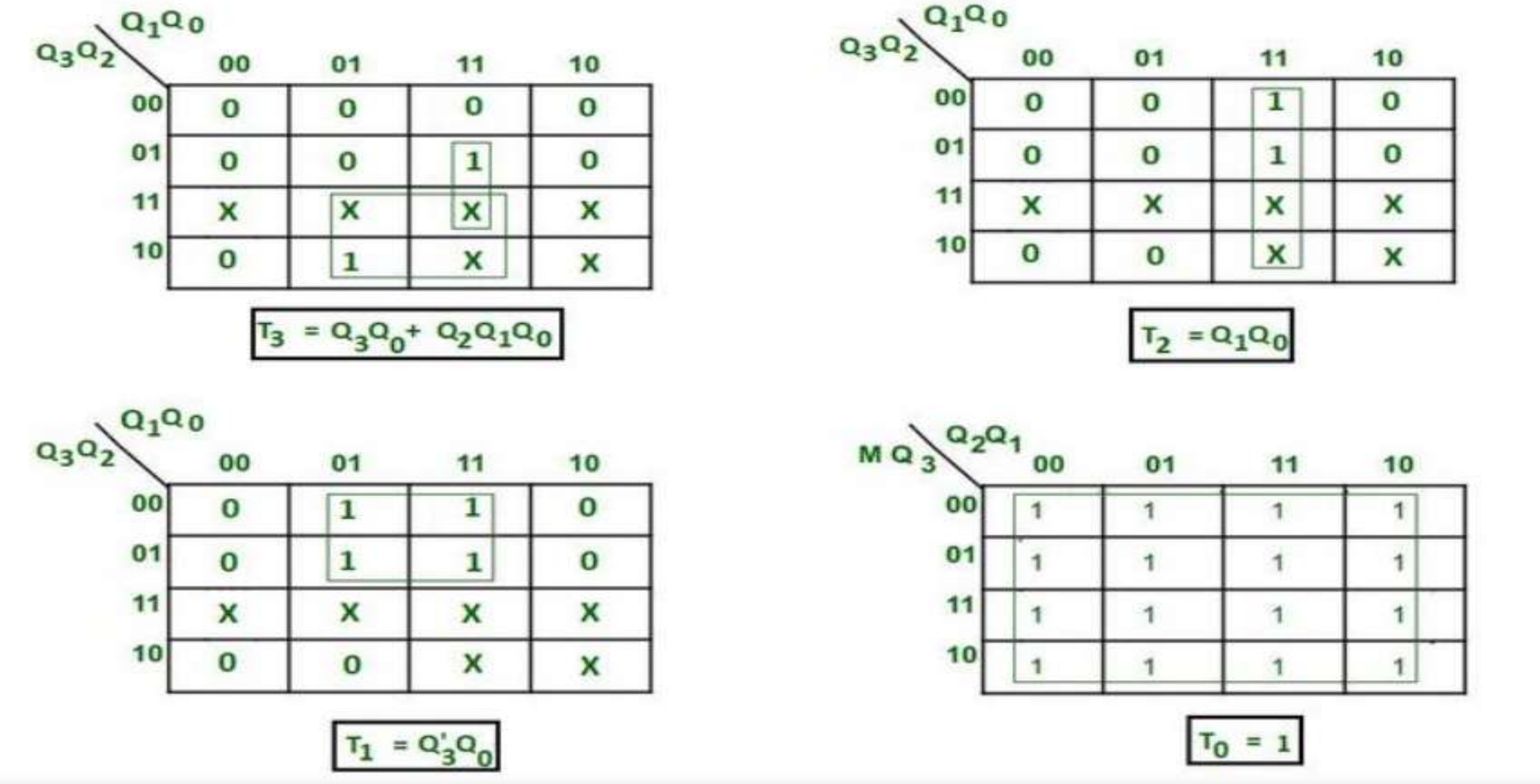


Q_3	Q_2	Q_1	Q_0	Q^*_3	Q^*_2	Q^*_1	Q^*_0	T_3	T_2	T_1	T_0
0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	1	0	0	1	0	0	0	1	1
0	0	1	0	0	0	1	1	0	0	0	1
0	0	1	1	0	1	0	0	0	1	1	1
0	1	0	0	0	1	0	1	0	0	0	1
0	1	0	1	0	1	1	0	0	0	1	1
0	1	1	0	0	1	1	1	0	0	0	1
0	1	1	1	1	0	0	0	1	1	1	1
1	0	0	0	1	0	0	1	0	0	0	1
1	0	0	1	0	0	0	1	1	0	0	1

Circuit excitation table.

Design Mod – N synchronous Counter

- **Step 4 : Create Karnaugh map for each FF input in terms of flip-flop outputs as the input variable –**



Q_3Q_2 Q_1Q_0	00	01	11	10
00	0	0	0	0
01	0	0	1	0
11	X	X	X	X
10	0	1	X	X

$T_3 = Q_3Q_0 + Q_2Q_1Q_0$

Q_3Q_2 Q_1Q_0	00	01	11	10
00	0	0	1	0
01	0	0	1	0
11	X	X	X	X
10	0	0	X	X

$T_2 = Q_1Q_0$

Q_3Q_2 Q_1Q_0	00	01	11	10
00	0	1	1	0
01	0	1	1	0
11	X	X	X	X
10	0	0	X	X

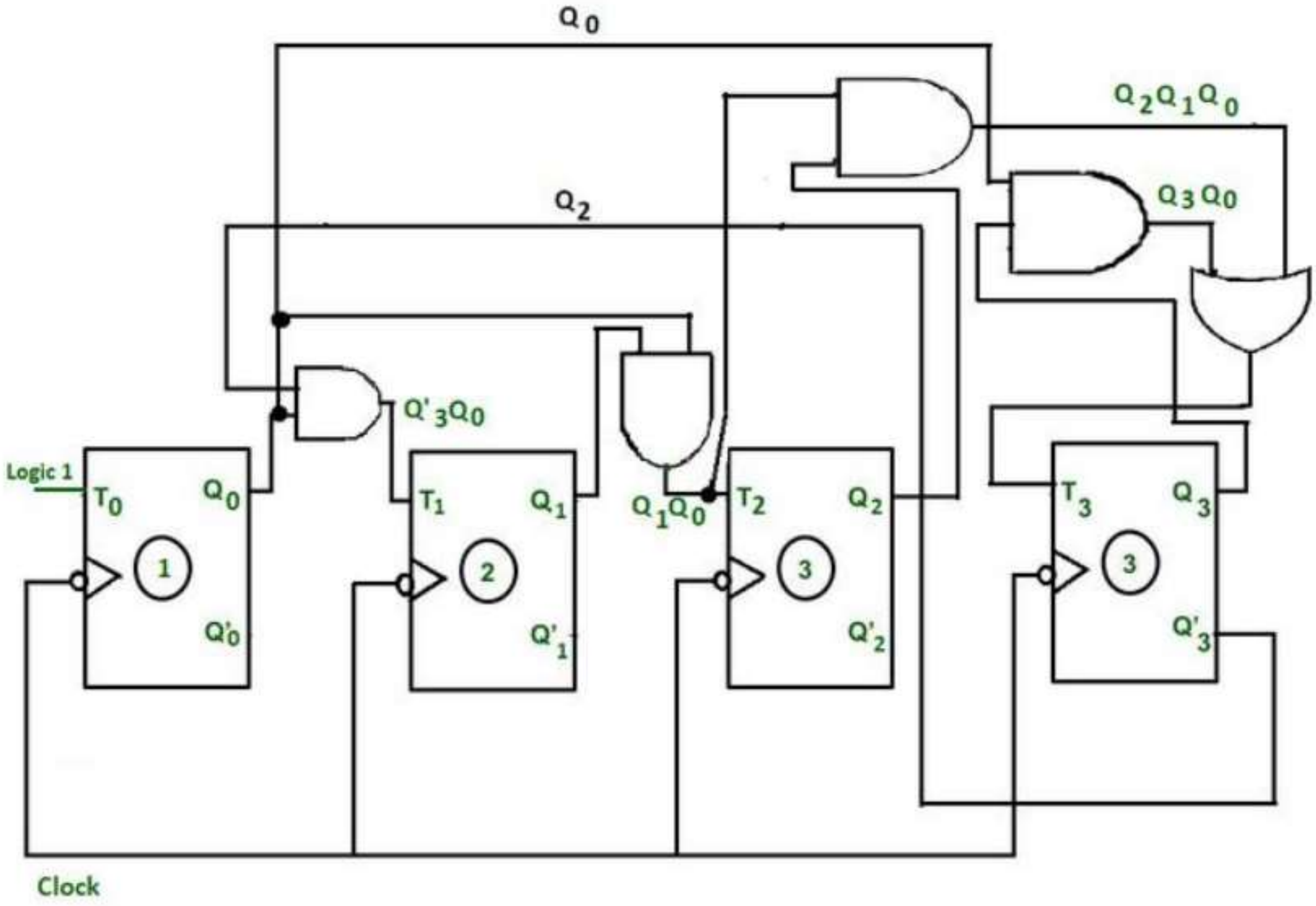
$T_1 = Q'_3Q_0$

MQ_3 Q_2Q_1	00	01	11	10
00	1	1	1	1
01	1	1	1	1
11	1	1	1	1
10	1	1	1	1

$T_0 = 1$

Design Mod – N synchronous Counter

- **Step 5 : Create circuit diagram:**


Q0
Q2Q1Q0
Q3Q0
Q2
Q'3Q0
Logic 1
T0
Q0
1
Q'0
T1
Q1
2
Q'1
Q1Q0
T2
Q2
3
Q'2
T3
Q3
3
Q'3
Clock