# UNIT-III

# SEQUENTIAL CIRCUITS-I

## Modules

**3.1** Classification of sequential circuits

- **3.1.1** Synchronous Sequential Circuits
- **3.1.2** Asynchronous Sequential Circuits

**3.2** Latches and Flip flops

- **3.2.1** Operation of NAND latch
- **3.2.2** Operation of NOR latch
- **3.2.3** RS flip-flop
- **3.2.4** JK flip-flop
- **3.2.5** T flip-flop
- **3.2.6** D flip-flop

**3.3** Conversion from one flip-flop to another flip- flop

- **3.3.1** SR to JK flip-flop
- **3.3.2** JK to T flip-flop
- **3.3.3** SR to D flip-flop

**3.4** Design of 4-bit Ripple or Asynchronous Counters

- **3.4.1** 4-bit up counter
- **3.4.2** 4-bit Down counter
- **3.4.3** Mod-10 Counter

**3.5** Design of Synchronous Counters

- **3.5.1** 4-bit Synchronous up counter
- **3.5.2** 4-bit Synchronous Down counter
- **3.5.3** Mod-10 Synchronous Counter or Decade Counter

**3.6** Johnson Counter

**3.7** Ring Counter

**3.8** Design of registers

- **3.8.1** Buffer register
- **3.8.2** Control Buffer Register
- **3.8.3** Shift Register

## I. Introduction:

- Sequential circuits are digital circuits that store and use the previous state information to determine their next state. Unlike combinational circuits, which only depend on the current input values to produce outputs, sequential circuits depend on both the current inputs and the previous state stored in memory elements. The differences between Combinational and Sequential Circuits are listed in Table 3.1.

**Table.3.1 Differences between Combinational and Sequential Circuits:**

| Combinational Circuits | Sequential Circuits |
|---|---|
| In Combinational Circuits, the output variables are dependent on only the combination of input variables. | In sequential circuits, the output variables are dependent not only on the present input variables but they also depend on the past history of the input variables |
| Memory unit is not required in combinational circuits | Memory unit is used to store the past history of the input variables |
| Combinational circuits are faster in speed because the delay between input and output is due to the propagation delay of gates | Sequential circuits are slower than combinational circuits |
| Combinational circuits are very easy to design | Sequential circuits are harder to design |
| Ex: Half adder, Full adder, parallel adder, etc | Ex: Serial adder |

- Examples of sequential circuits include latches and Flip-Flops.
- Latches are digital circuits that store a single bit of information and hold its value until it is updated by new input signals. They are used in digital systems as temporary storage elements to store binary information. Latches can be implemented using various digital logic gates, such as AND, OR, NOT, NAND, and NOR gates.
- The memory or storage elements (memory) used in clocked sequential circuits are called flipflops.
- A basic flip-flop can be constructed using four-NAND or four-NOR gates. Flip-flop is popularly known as the basic digital memory circuit. It has its two states as logic 1(High) and logic 0(low) states. A flip flop is a sequential circuit which consist of single binary state of information or data.

The differences between Latches and Flip flops are listed in Table 3.2.

**Table.3.2 Differences between Flip-flops and Latches:**

| Flip-Flop | Latch |
|---|---|
| Flip-flop is a bistable device i.e., it has two stable states that are represented as 0 and 1. | Latch is also a bistable device whose states are also represented as 0 and 1. |
| It checks the inputs but changes the output only at times defined by the clock signal or any other control signal. | It checks the inputs continuously and responds to the changes in inputs immediately. |
| It is a edge triggered device. | It is a level triggered device. |
| These are made up of logic gates | These are also made up of logic gates. |
| They are classified into asynchronous or synchronous flip-flops. | There is no such classification in latches |
| Flip-flop always have a clock signal | Latches doesn’t have a clock signal |
| ex: D Flip-flop, JK Flip-flop, SR flip-flop, T flip-flop | ex: SR NAND Latch, SR NOR latch |

**Counters:**

- A counter is a sequential logic circuit that can accumulate the number of input pulses. It can be used not only to count the number of clock pulses, but also to perform various tasks such as frequency division, timing, generation of precise time ticks and pulse trains, and even number crunching.

**Registers:**

- A register is a small and temporary storage unit inside a computer’s (CPU). It plays a vital role in holding the data required by the CPU for immediate processing and is made up of flip-flops. Registers act as intermediate storage for data during arithmetic logic and other processing operations.

## II. Applications:

**Applications of Sequential Logic Circuits:**

The extensive applications of sequential logic circuits are:

- Shift registers
- Flip flops
- Analog to digital and digital to analog converters
- Counters
- Clocks
- Used as registers inside microprocessors and controllers to store temporary information
- Applied in programmable devices such as CPLD, PLD, and FPGA

**Applications of Flip-flops:**

- **Counters:** The Flip Flop are used in the Counter Circuits for Counting pulse or events.
- **Frequency Dividers:** The Flip Flop are used in Frequency Dividers to divide the frequency of a input signal by a specific factor.
- **Shift Registers:** The Shift registers consist of interconnected flip-flops that shift data serially.
- **Storage Registers:** The Storage Resistor uses Flip Flop to store data in binary information.
- **Bounce elimination switch:** The Flip Flop are used in Bounce elimination switch to eliminate the contact bounce.
- **Data storage:** The Flip Flop are used in the Data Storage to store binary data temporarily or permanently.
- **Data transfer:** The Flip Flops are used for data transfer in different electronic parts.
- **Latch:** The Latches are the Sequential circuit which uses Flip Flop for temporary storage of data
- **Registers:** The Registers are mode from the array of flip flop which are used to store data temporarily**.**
- **Memory:** The Flip Flops are the main components in the memory unit for data storage.

**Applications of the counters:**

- **Frequency Measurement and Division:** The counter is used to measure the frequency of a signal, simply by counting the no of cycles in a particular given time period and the counter is also used to divide the input clock frequency by a fixed integer value.
- **Timing:** The counter is also used to generate timing signals like pulse-width modulated (PWM) signals.
- **Binary Arithmetic:** Binary arithmetic operations like addition, subtraction, multiplication, and division are used in digital systems by counters.
- **Data Storage:** The binary counter can be used to store a binary value that represents a state in a digital system.
- **Digital Signal Processing:** Counters are also used in digital signal processing applications like filtering and signal analysis.

**Applications of registers:**

- **Arithmetic and Logic Operations:** The Registers are extensively used during arithmetic and logic operations in the CPU.
- **Instruction Execution:** They hold instructions and data needed for immediate processing ensuring smooth program execution.
- **CPU Pipelining:** The pipelining is employed to increase performance by overlapping the execution of multiple instructions.
- **Context Switching:** Registers are used to store the CPU's state including the program counter stack pointer and general-purpose registers during context switching.
- **Memory Addressing:** The memory address register (MAR) and memory data register (MDR) are involved in memory operations.
- **Input/output Operations:** The Registers are used to buffer data during input/output operations.
- **Floating-Point Operations:** These registers can store and manipulate floating-point numbers with higher precision.
- **Control Unit Operations:** The registers hold control signals and flags that determine the control flow and sequencing of the instructions during program execution.

## 3.1 Classification of sequential circuits

- The logic circuits whose outputs at any instant of time depend on the present inputs as well as on the past outputs are called sequential circuits.

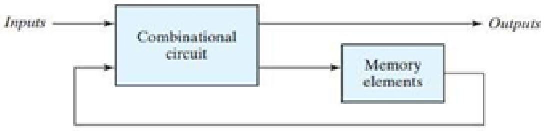


**Fig.3.1 Block diagram of a sequential circuit**

- In sequential circuits, the output signals are feedback to the input side.
- A block diagram of a sequential circuit is shown in Figure 3.1.
- It consists of a combinational circuit to which storage elements are connected to form a feedback path.
- The storage elements are devices capable of storing binary information.
- The binary information stored in these elements at any given time defines the ***state*** of the sequential circuit at that time.
- These external inputs also determine the condition for changing the state in the storage elements.
- The next state of the storage elements is also a function of external inputs and the present state. Thus, **a sequential circuit is specified by a time sequence of inputs, outputs, and internal states.**
- There are two types of sequential circuits. They are classified as follows:

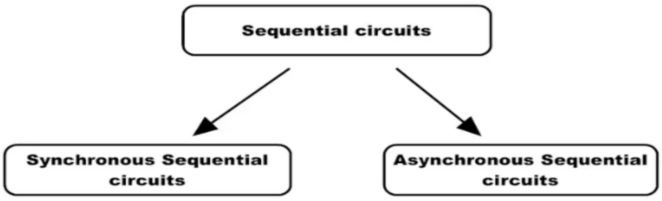

Sequential circuits
Synchronous Sequential circuits
Asynchronous Sequential circuits

**Fig.3.2 Classification of Sequential Circuits**

### 3.1.1 Synchronous Sequential Circuits

- A sequential circuit whose behavior can be defined from the knowledge of its signal at discrete instants of time is referred to as a ***synchronous sequential circuit***.

- In these systems, the memory elements are affected only at discrete instants of time.
- The synchronization is achieved by a timing device known as a system clock, which generates a periodic train of clock pulses.
- The outputs are affected only with the application of a clock pulse.

### 3.1.2 Asynchronous Sequential Circuits.

- A sequential circuit whose behavior depends upon the sequence in which the input signals change is referred to as an ***asynchronous sequential circuit***.
- The output will be affected whenever the input changes.
- The commonly used memory elements in these circuits are time-delay devices.
- Therefore, in general, asynchronous circuits are faster than synchronous sequential circuits.
- The differences between Synchronous Sequential Circuits and Asynchronous Sequential Circuits are listed in Table 3.3.

**Table.3.3 Comparison between Synchronous and Asynchronous Sequential Circuits**

| S. No. | Synchronous sequential circuit | Asynchronous sequential circuit |
|---|---|---|
| 1 | In synchronous sequential circuits, memory elements are clocked flip flops. | In Asynchronous sequential circuits, memory elements are either unclocked flip-flops or time delay elements. |
| 2 | In synchronous circuits, change in input signals can affect memory elements upon activation of the clock signal. | In asynchronous circuits, change in input signals can affect memory elements at any instant of time. |
| 3 | The operating speed of synchronous sequential circuits is low. | Asynchronous sequential circuits can operate faster than Synchronous sequential circuits |
| 4 | These are more expensive. | These are less expensive. |
| 5 | Easier to design | More difficult to design |
| 6 | Synchronous circuits are used in counters, shift registers, memory units. | Asynchronous circuits are used in simple microprocessors, digital signal processing units, and in communication systems. |

## 3.2 Latches and Flip-flops

- Storage elements that operate with signal levels (rather than signal transitions) are referred to as **latches**.

- Storage elements those controlled by a clock transition are **flip-flops**. The memory or storage elements (memory) used in clocked sequential circuits are called ***flipflops.***

- Latches are said to be level sensitive devices. Flip-flops are edge-sensitive devices.

### 3.2.1 NOR latch

- The logic symbol and logic diagram of NOR latch are shown in Figure 3.3 and 3.4 respectively. The truth table of NOR latch is tabulated in Table 3.4.

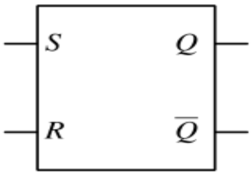


**Fig.3.3 Logic Symbol of SR NOR latch**

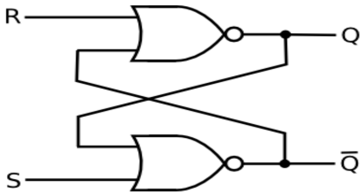


**Fig.3.4 Logic Diagram of SR NOR latch**

- When both ‘S’ and ‘R’ are 0, the latch maintains its state. That is, the latch will retain its previous state as the output. Hence this state is referred to as No Change State.

- When ‘S’ is 0 and ‘R’ is 1, the latch outputs Q=0 and $\overline{Q}$=1.

- When ‘S’ is 1 and ‘R’ is 0, the latch outputs Q=1 and $\overline{Q}$=0.

- When both ‘S’ and ‘R’ are 1, it’s an invalid condition resulting in both Q and $\overline{Q}$ being the same equal to 0. As Q and $\overline{Q}$ cannot be same, this state is referred to as Forbidden or Invalid or Indeterminate state.

**Table.3.4 Truth table of NOR latch**

| **Inputs** | | **Outputs** | | **State** |
|---|---|---|---|---|
| **S** | **R** | **Q** | $\overline{Q}$ | |
| 0 | 0 | **$Q_n$** | $\overline{Q}_n$ | No change |
| 0 | 1 | 0 | 1 | RESET |
| 1 | 0 | 1 | 0 | SET |
| 1 | 1 | **0** | **0** | FORBIDDEN |

### 3.2.2 NAND latch

- The logic symbol and logic diagram of NAND latch are shown in Figure 3.5 and 3.6 respectively. The truth table of NAND latch is tabulated in Table 3.5.

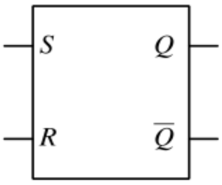


**Fig.3.5 Logic Symbol of SR NAND Latch**

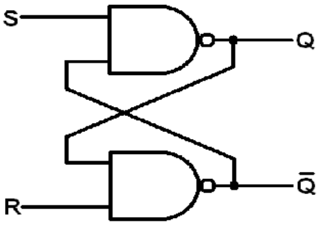


**Fig.3.6 Logic Diagram of SR NAND Latch**

- When both ‘S’ and ‘R’ are 1, the latch maintains its state. That is, the latch will retain its previous state as the output. Hence this state is referred to as No Change State.
- When ‘S’ is 0 and ‘R’ is 1, the latch outputs Q=1 and $\overline{Q}$=0.
- When ‘S’ is 1 and ‘R’ is 0, the latch outputs Q=0 and $\overline{Q}$=1.
- When both ‘S’ and ‘R’ are 0, it’s an invalid condition resulting in both Q and $\overline{Q}$ being the same equal to 1. As Q and $\overline{Q}$ cannot be same, this state is referred to as Forbidden or Invalid or Indeterminate state.

**Table.3.5 Truth table of NAND Latch**

| Inputs | | Outputs | | State |
|---|---|---|---|---|
| **S** | **R** | **Q** | $\overline{Q}$ | |
| 0 | 0 | 1 | 1 | FORBIDDEN |
| 0 | 1 | 1 | 0 | SET |
| 1 | 0 | 0 | 1 | RESET |
| 1 | 1 | $Q_n$ | $\overline{Q}_n$ | No change |

**Important terms:**

**Characteristic table:**

A characteristic table defines the next state of a flip-flop based on its current state and the input(s). It's a table that maps input conditions and present state to the resulting next state.

**Characteristic Equation:**

The "characteristic equation" describes the relationship between the current state, the inputs, and the next state of the flip-flop. It's a fundamental equation used to understand and predict the behavior of different types of flip-flops.

**Excitation table:**

An excitation table helps determine the required inputs (e.g., J, K, S, R, D, T) for a flip-flop to change from its current state ($Q_n$) to a specific next state ($Q_{n+1}$).

**Preset and Clear Inputs:**

- In many applications, it is required to initially set or reset the flip-flop., i.e., the initial state of the flip-flop is to be assigned. This can be done by using the Preset (Pr) and Clear (Clr) inputs. These inputs may be applied at any time. Consider active low Preset and Clear inputs.
- If Pr = 1 and Clr =0, the output of NAND gate 4 is forced to be 1, i.e., Q' =1 and Q=0. The flip flop is reset.
- If Pr = 0 and Clr = 1, the output of NAND gate 3 is forced to be 1, i.e., Q = 1 and the flip-flop is set.
- The condition Pr=0 and Clr=0 must not be applied, since this leads to an uncertain state.
- If Pr=1 and Clr=1, the circuit operates according to the truth table of the corresponding flip flop.

### 3.2.3 RS flip-flop

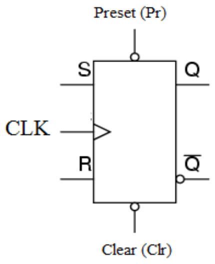


**Fig.3.7 Logic Symbol of SR flip-flop**

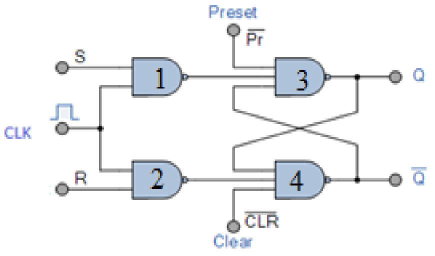


**Fig.3.8 Logic Diagram of SR flip-flop**

- The logic symbol and logic diagram and timing diagram of SR flip-flop are shown in Figure 3.7, 3.8 and 3.9 respectively.
- If Pr = 1 and Clr =0, flip-flop is reset.

- If Pr = 0 and Clr=1, the flip-flop is set.
- If Pr=1 and Clr=1, the circuit operates according to the truth table of the SR flip flop.
- **Case 1**: If S=R=0, and the clock pulse is not applied, the output of the flip-flop remains in the present state. Even if S=R=0, and the clock pulse is applied, the output at the end of the clock pulse is the same as the output before the clock pulse, i.e., $Q_{n+1}=Q_n$. This state is called No Change State.
- **Case 2:** For S=0and R= 1, if the clock pulse is applied (CLK= 1), Q=0. This state is called Reset state.
- That is the output of NAND gate 1 becomes 1; whereas the output of NAND gate 2 will be 0. Now a 0 at the input of NAND gate 4 forces the output to be 1i.e. Q' = 1. This 1 goes to the input of NAND gate 3 to make both the inputs of NAND gate 3 as 1, which forces the output of NAND gate3 to be 0,i.e.,Q=0.
- **Case 3:** For S= 1 and R= 0, if the clock pulse is applied (i.e., CLK = 1), Q=1. This state is called Set state.
- That is the output of NAND gate 2 becomes1; whereas the output of NAND gate 1 will be 0. Now a 0 at the input of NAND gate 3 forces the output to be 1, i.e., Q = 1. This 1 goes to the input of NAND gate 4 to make both the inputs of NAND gate 4 as 1, which forces the output of NAND gate4 to be 0,i.e.,Q'=0.
- **Case 4:** For S= 1 and R= 1, if the clock pulse is applied (i.e. CLK = 1), Q and Q’ are same, which is not possible. So it is called Invalid state.
- That is the outputs of both NAND gate 2 and NAND gate 1 becomes0. Now a 0at the input of both NAND gate 3and NAND gate 4forces the outputs of both the gates to be 1, i.e., Q =1 and Q' =1.When the CLK input goes back to 0 (while S and R remain at 1), it is not possible to determine the next state, as it depends on whether the output of gate 1 or gate 2 goes to 1 first.
- The truth table, characteristic table and excitation table of SR flip-flop is tabulated in Table 3.6, 3.7 and 3.8 respectively.

**Table.3.6 Truth Table of SR Flip-Flop**

| **Inputs** | | | | | **Output** | State |
|---|---|---|---|---|---|---|
| **Preset (Pr)** | **Clear (Clr)** | **Clk** | **S** | **R** | **$Q_{n+1}$** | |
| 1 | 0 | x | x | x | 0 | Clear |
| 0 | 1 | x | x | x | 1 | Preset |
| 1 | 1 | ↓ | x | x | $Q_n$ | No change |

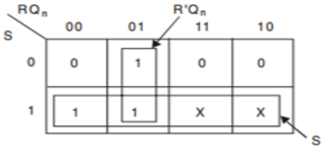
| | | | | | | |
|---|---|---|---|---|---|---|
| **1** | **1** | ↑ | **0** | **0** | **$Q_n$** | **No change** |
| **1** | **1** | ↑ | **0** | **1** | **0** | **Reset** |
| **1** | **1** | ↑ | **1** | **0** | **1** | **Set** |
| **1** | **1** | ↑ | **1** | **1** | **x** | **Invalid** |

- The next state of the flip-flop output ($Q_{n+1}$) depends on the present inputs as well as the present output ($Q_n$).

**Table.3.7 Characteristic Table of SR Flip-Flop**

| *Flip-flop inputs* | | *Present output* | *Next output* |
|---|---|---|---|
| *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 |

- Now determine the $Q_{n+1}$ in terms of $Q_n$ and S, R



- From the Karnaugh map above we find the expression for $Q_{n+1}$ as

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

**Timing diagram**

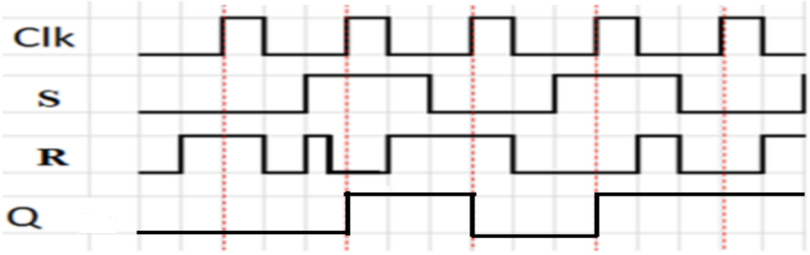


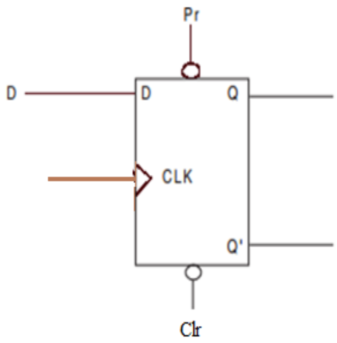
**Fig.3.9. Timing Diagram of SR flip flop**

**Table.3.8 Excitation table of SR flip-flop**

| $Q_n$ | $Q_{n+1}$ | S | R |
|---|---|---|---|
| 0 | 0 | 0 | X |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | X | 0 |

### 3.2.4 D flip-flop

- The logic symbol and logic diagram and timing diagram of D flip-flop are shown in Figure 3.10, 3.11 and 3.12 respectively.
- One way to eliminate the undesirable condition of the indeterminate state in the *SR* latch is to ensure that inputs *S* and *R* are never equal to 1 at the same time. This is done in the *D* flip flop.
- It can be easily constructed from an S-R flip-flop by simply incorporating an inverter between S and R.
- The D flip-flop has only one input referred to as the D (**data**) input & two outputs as Q and Q'.
- If the clock is high, the output Qn+1 is always equal to the input D.
- If D= 0, then $Q_{n+1}$ = 0.
- If D= 1, then $Q_{n+1}$ =1.
- A D flip-flop is like a buffer. It is most often used in the construction of sequential circuits like registers.
- The truth table, characteristic table and excitation table of D flip-flop is tabulated in Table 3.9, 3.10 and 3.11 respectively.



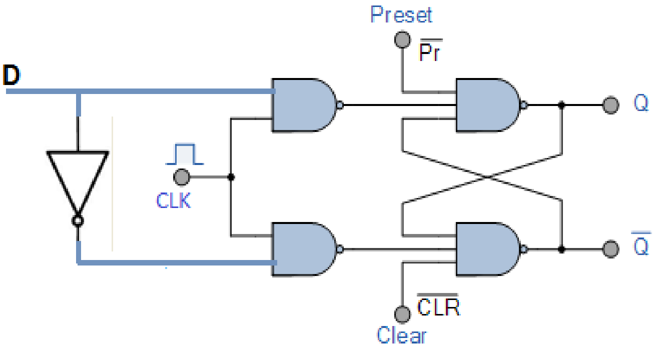
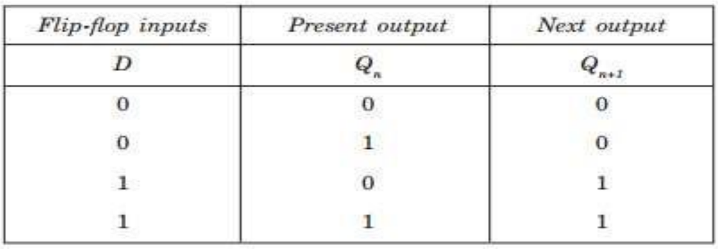


**Fig.3.10 Logic Symbol of D flip-flop** **Fig.3.11 Logic Diagram of D flip-flop**

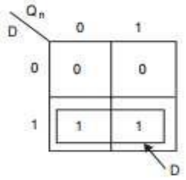
**Table.3.9 Truth Table of D Flip-Flop**

| Inputs | | | | Current output | State |
|---|---|---|---|---|---|
| **Preset (Pr)** | **Clear (Clr)** | **Clk** | **D** | **$Q_{n+1}$** | |
| 1 | 0 | x | x | 0 | Clear |
| 0 | 1 | x | x | 1 | Preset |
| **1** | **1** | ↑ | **0** | **0** | **Data Buffer** |
| **1** | **1** | ↑ | **1** | **1** | |

**Table.3.10 Characteristic Table of a D Flip-flop**



| *Flip-flop inputs* | *Present output* | *Next output* |
|---|---|---|
| *D* | $Q_n$ | $Q_{n+1}$ |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |



- ➢ Simplify the K-map and find out the characteristic equation from the characteristic table.
- ➢ Hence, the characteristic equation of a D flip-flop is

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

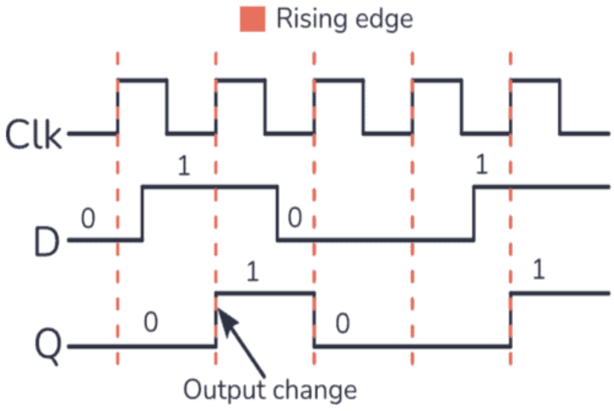
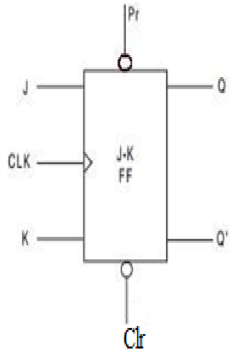


**Fig.3.12 Timing Diagram of D flip-flop**

**Table.3.11 Excitation table of D flip flop**

| $Q_n$ | $Q_{n+1}$ | D |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

### 3.2.5 JK flip-flop



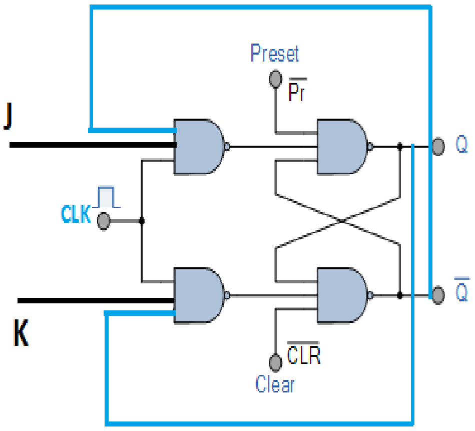


**Fig.3.13 Logic Symbol of JK flip-flop** **Fig.3.14 Logic Diagram of JK flip-flop**

- The logic symbol, logic diagram and timing diagram of JK flip-flop are shown in Figure 3.13, 3.14 and 3.15 respectively.

- Inputs J and K behave like inputs S and R to set and reset the flip-flop respectively.

- When J =K = 1, the flip-flop is said to be in a ***toggle state***, which means the output switches to its complementary state every time a clock passes.

- **Case 1.**When the clock is applied, J = 0, and K=0, the flip flop remains in same state (Qn+1=Qn). This state is called No change state.

- That is when J=0, whatever the value of $Q'_n$ (0 or 1), the output of NAND gate 1 is 1. Similarly, when K =0, whatever the value of $Q_n$ (0 or 1), the output of gate 2 is also 1. Therefore, when J=0 and K =0, the inputs to the basic flip-flop are S =1andR=1.This condition forces the flip-flop to remain in the same state.

- **Case 2.** When the clock is applied, J = 0, and K=1, the flip flop output is 0. This state is called Reset state.

- That is, when J = 0 and K = 1 & the previous state of the flip-flop is reset (*i.e.*, $Q_n$= 0and $Q'_n$= 1), then S = 1 and R = 1. Since S = 1 and R = 1, the basic flip-flop does not alter the state and remains in the reset state. But if the flip-flop is in set condition (*i.e.*, $Q_n$= 1 &$Q'_n$= 0), then S = 1 and R = 0. Since S = 1 and R = 0, the basic flip-flop changes its state and resets.

- **Case 3.** When the clock is applied, J=1, and K=0, the flip flop output is 1. This state is called Set state.

- That is when J = 1 and K = 0 and the previous state of the flip-flop is reset (*i.e.*, $Q_n$= 0 and $Q'_n$= 1), then S = 0 and R = 1. Since S = 0 and R = 1, the basic flip-flop changes its state and goes to the set state. But if the flip-flop is already in set condition (*i.e.*, $Q_n$= 1 and $Q'_n$= 0), then S = 1 and R = 1.Since S = 1and R = 1, the basic flip-flop does not alter its state and remains in the set state.

- **Case 4.** When the clock is applied, J=1, and K=1, the flip flop output toggles. This state is called Toggle state.

- That is when J = 1 and K = 1 and the previous state of the flip-flop is reset (*i.e.*, $Q_n$= 0 and $Q'_n$= 1), then S = 0 and R = 1. Since S = 0 and R = 1, the basic flip-flop changes its state and goes to the set state. But if the flip-flop is already in set condition (*i.e.*, $Q_n$= 1 and $Q'_n$= 0), then S = 1 and R = 0.Since S = 1and R = 0, the basic flip-flop changes its state and goes to the reset state. So we find that for J = 1 and K = 1, the flip-flop toggles its state from *set* to *reset* and vice versa. Toggle means to switch to the opposite state.

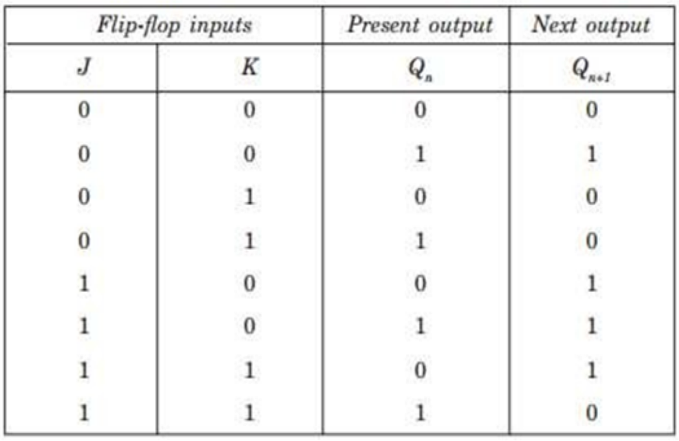
- The truth table, characteristic table and excitation table of JK flip-flop is tabulated in Table 3.12, 3.13 and 3.14 respectively.

**Table.3.12 Truth table of JK flip-flop**

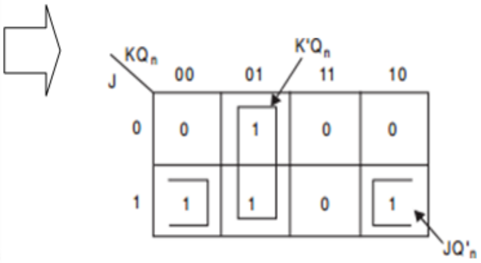
| **Inputs** | | | | | **Output** | State |
|---|---|---|---|---|---|---|
| **Preset (Pr)** | **Clear (Clr)** | **Clk** | **J** | **K** | **$Q_{n+1}$** | |
| 1 | 0 | x | x | x | 0 | Clear |
| 0 | 1 | x | x | x | 1 | Preset |
| 1 | 1 | ↓ | x | x | $Q_n$ | No change |
| **1** | **1** | ↑ | **0** | **0** | $\mathbf{Q_n}$ | **No change** |
| **1** | **1** | ↑ | **0** | **1** | **0** | **Reset** |
| **1** | **1** | ↑ | **1** | **0** | **1** | **Set** |
| **1** | **1** | ↑ | **1** | **1** | $\overline{Qn}$ | **Toggle** |

- From the characteristic table, we have to find out the characteristic equation of the J-K flip-flop.

**Table.3.13 Characteristic table of JK flip flop**



| *Flip-flop inputs* | | *Present output* | *Next output* |
|---|---|---|---|
| *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 |



From the Karnaugh map, we obtain the characteristic equation of the JK flip-flop

$$\boxed{Q_{n+1}=JQ'_n+K'Q_n}$$

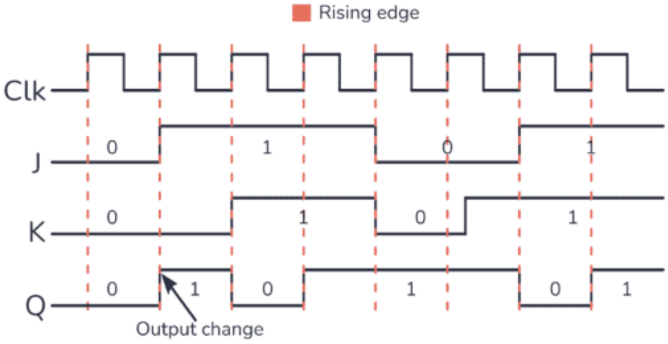


**Fig.3.15 Timing Diagram of JK flip-flop**

**Table.3.14 Excitation table of JK Flip-Flop**

| $Q_n$ | $Q_{n+1}$ | J | K |
|---|---|---|---|
| 0 | 0 | 0 | x |
| 0 | 1 | 1 | x |
| 1 | 0 | x | 1 |
| 1 | 1 | X | 0 |

### 3.2.6 T flip-flop

- The logic symbol, logic diagram and timing diagram of T flip-flop are shown in Figure 3.16, 3.17 and 3.18 respectively.

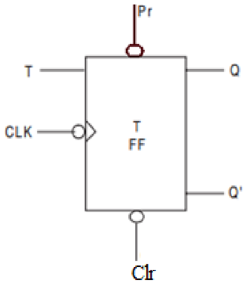


Fig.3.16 Logic Symbol of T flip-flop

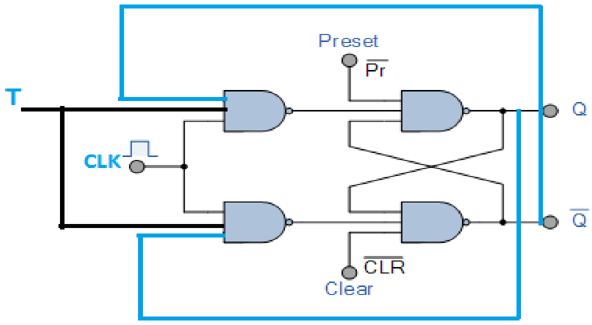


Fig.3.17 Logic Diagram of T flip-flop

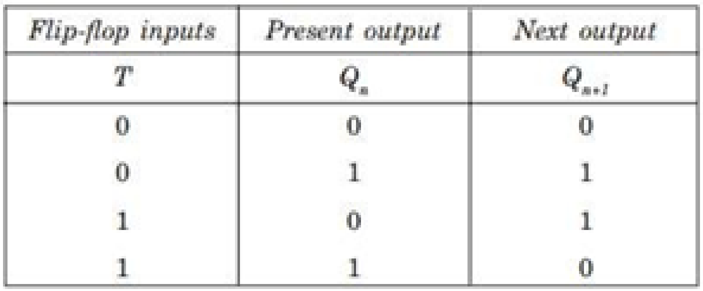
**Table.3.15 Truth table of T Flip-flop**

| Inputs | | | | Current output | State |
|---|---|---|---|---|---|
| **Preset (Pr)** | **Clear (Clr)** | **Clk** | **T** | **$Q_{n+1}$** | |
| 1 | 0 | x | x | 0 | Clear |
| 0 | 1 | x | x | 1 | Preset |
| **1** | **1** | **↑** | **0** | **Qn** | **No change** |
| **1** | **1** | **↑** | **1** | $\overline{Qn}$ | **Toggle** |

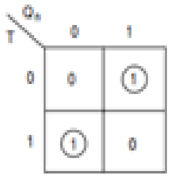
- If the T input is 0 state (*i.e.*, J=K =0) prior to a clock pulse, the Q output will not change with the clock pulse. On the other hand, if the T input is in 1 state (*i.e.*, J = K = 1) prior to a clock pulse, the Q output will change to Q' with the clock pulse.
- In other words, we may say that, if T = 1 and the device is clocked, then the output toggles its state.
- If T = 0, then $Q_{n+1} = Q_n$
- If T = 1, then $Q_{n+1} = Q'_n$

- The truth table, characteristic table and excitation table of T flip-flop is tabulated in Table 3.15, 3.16 and 3.17 respectively.

**Table.3.16 Characteristic Table of a T Flip-flop**



| Flip-flop inputs | Present output | Next output |
|---|---|---|
| T | $Q_n$ | $Q_{n+1}$ |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |



| T \ $Q_n$ | 0 | 1 |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 1 | 0 |

- Hence, the characteristic equation of a T flip-flop is

$$\boxed{Q_{n+1}=TQ'_n+T'Q_n}$$

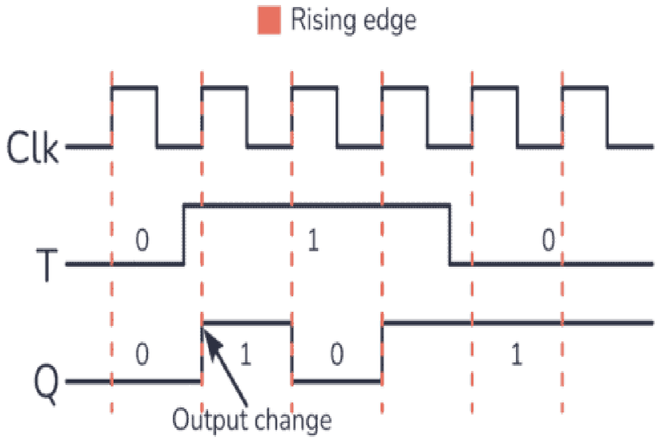


**Fig.3.18 Timing Diagram of T flip-flop**

**Table.3.17 Excitation table of T Flip-flop**

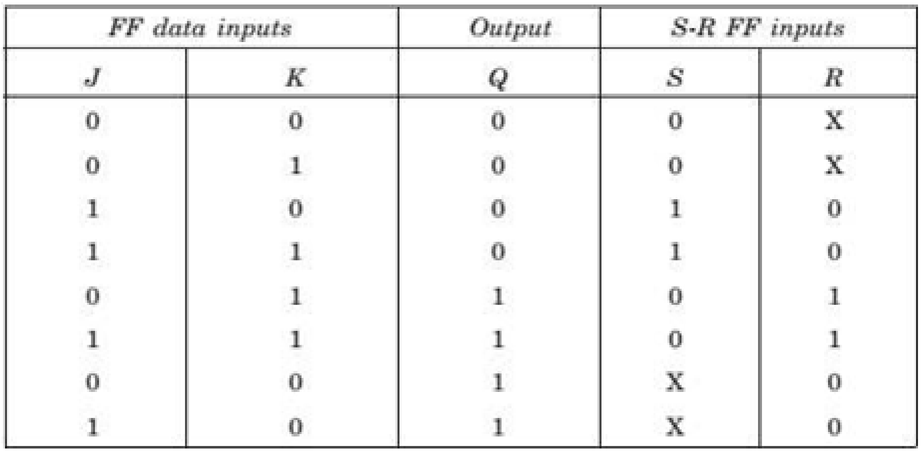
| Present state<br>$Q_n$ | Next state<br>$Q_{n+1}$ | T |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

## 3.3 Conversion from one flip-flop to another flip- flop

### 3.3.1 SR to JK flip-flop

**Step 1:** The excitation tables of S-R and J-K flip-flops are tabulated in Table 3.18.

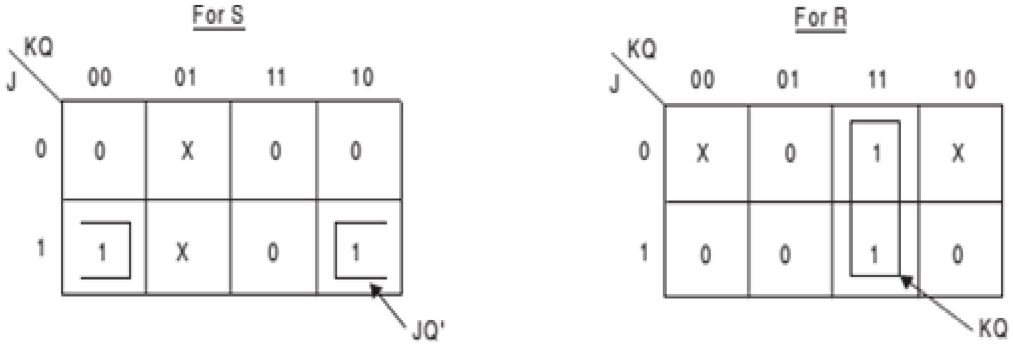
**Table.3.18 Excitation table of JK and SR Flip-Flop**



| *FF data inputs* | | *Output* | *S-R FF inputs* | |
|---|---|---|---|---|
| *J* | *K* | *Q* | *S* | *R* |
| 0 | 0 | 0 | 0 | X |
| 0 | 1 | 0 | 0 | X |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 | 1 |
| 0 | 0 | 1 | X | 0 |
| 1 | 0 | 1 | X | 0 |

**Step 2:** The excitation table is simplified by using K-Map technique to obtain the expression of inputs

as follows



Hence, the Boolean expression for S and R as

$$\boxed{\begin{array}{c} S=JQ' \\ R= KQ \end{array}}$$

**Step 3:** Hence the logic diagram is realized. The logic diagram is shown in Fig.3.19.

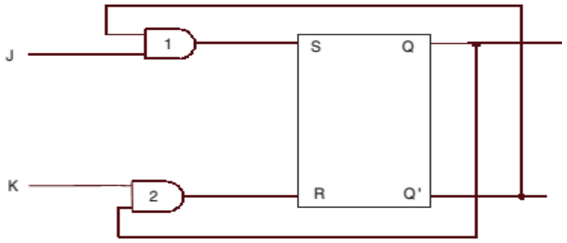


**Fig.3.19 Logic Diagram for Conversion of SR to JK flip-flop**

### 3.3.2 JK to T flip-flop

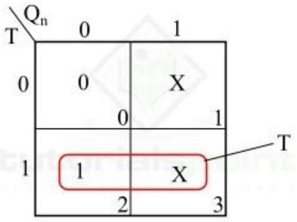
**Step 1**: The excitation table for the conversion of JK flip flop to T flip flop is tabulated in Table 3.19.

**Table.3.19 Excitation table of T and JK Flip-Flops**

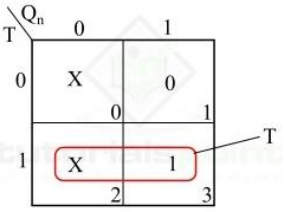
| T flip flop Input | Output | Jk flip flop inputs | |
|---|---|---|---|
| T | $Q_n$ | J | K |
| 0 | 0 | 0 | X |
| 0 | 1 | X | 0 |
| 1 | 0 | 1 | X |
| 1 | 1 | X | 1 |

**Step 2**: The excitation table is simplified by using K-Map technique to obtain the expression of inputs as follows:

The K-map simplification for the input J and K is as follows:



J = T



K = T

**Step 3**: The logic diagram of the T flip-flop using JK flip flop is shown in Figure 3.20.

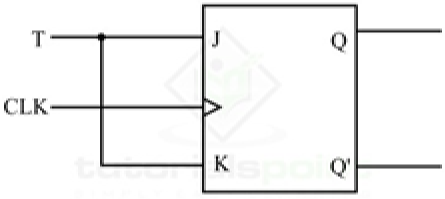
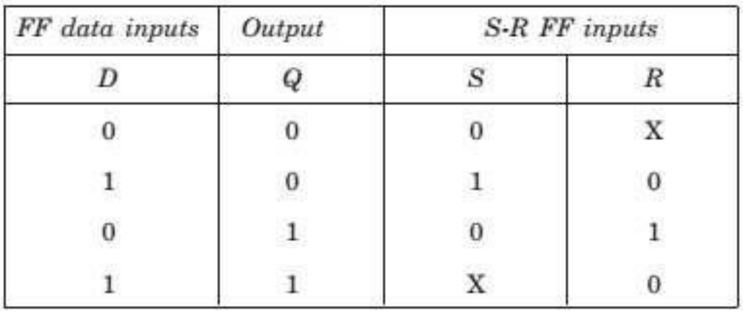


**Fig.3.20 Logic Diagram for Conversion of JK to T flip-flop**

### 3.3.3 SR to D flip-flop

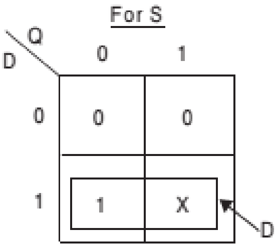
**Step 1:** The excitation tables of S-R and D flip-flops are tabulated in Table.3.20.

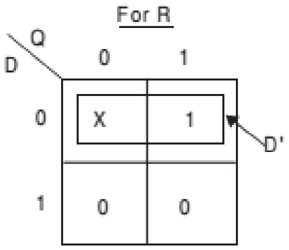
**Table.3.20 Excitation table of D and SR Flip-Flops**



| *FF data inputs* | *Output* | *S-R FF inputs* | |
|---|---|---|---|
| *D* | *Q* | *S* | *R* |
| 0 | 0 | 0 | X |
| 1 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 1 | 1 | X | 0 |

**Step 2:** From the above table, solve the Karnaugh maps for inputs S and R





Simplifying with the help of the Karnaugh maps, we obtain **S=D**

**R=D'**

**Step 3:** The logic diagram of D flip flop using SR flip flop is shown in Figure 3.21.

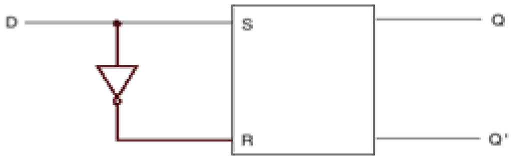


**Fig.3.21 Logic Diagram for Conversion of SR to D flip-flop**

## 3.4 Design of 4-bit Ripple Counter

### 3.4.1 4-bit up counter

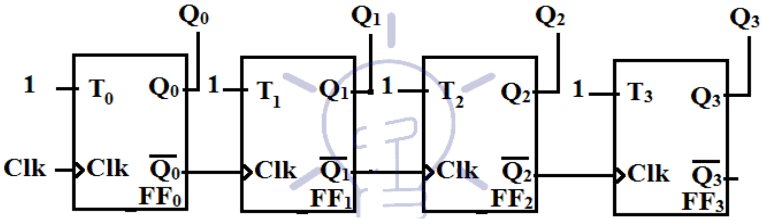


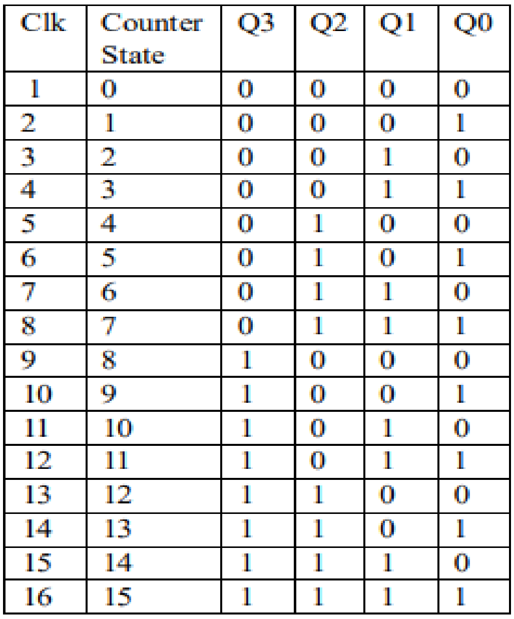
**Fig.3.22. Logic Diagram of 4-bit Ripple UP Counter**

- A 4 bit asynchronous UP counter with T flip flop is shown in Figure 3.22.
- An n-bit counter has n flip-flops and 2^n output states.
- Since it is a 4-bit up counter, it has 4 flip flops.
- The output states of the counter are $2^4$ (16 states). That is, the states are from 0000 to 1111.
- The output of the flip flop is Q3Q2Q1Q0. Q3 is the MSB and Q0 is the LSB.
- That means the flip flops will toggle at each active edge or positive edge of the clock signal. The clock input is connected to first flip flop
- For 4-bit ripple up counter, the other flip flops in counter receive the clock signal input from Q' output of previous flip flop.
- Let us assume that the outputs of the four flip flops are initially 0000. When the rising edge of the clock pulse is applied to the FF0, then the output Q0 will change to logic 1 and the next clock pulse will change the Q0 output to logic 0.
- As the $Q_0$' of FF0 is connected to the clock input of FF1, the Q1 toggles for every positive edge of $Q_0$'.
- As the $Q_1$' of FF1 is connected to the clock input of FF1, the Q2 toggles for every positive edge of $Q_1$'.
- As the $Q_2$' of FF1 is connected to the clock input of FF1, the Q3 toggles for every positive

edge of $Q_2$'.

- Hence the up counter counts from 0000 to 1111.

**Table.3.21 Truth table of 4-bit UP Counter (Ripple)**



| Clk | Counter State | Q3 | Q2 | Q1 | Q0 |
|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 0 |
| 2 | 1 | 0 | 0 | 0 | 1 |
| 3 | 2 | 0 | 0 | 1 | 0 |
| 4 | 3 | 0 | 0 | 1 | 1 |
| 5 | 4 | 0 | 1 | 0 | 0 |
| 6 | 5 | 0 | 1 | 0 | 1 |
| 7 | 6 | 0 | 1 | 1 | 0 |
| 8 | 7 | 0 | 1 | 1 | 1 |
| 9 | 8 | 1 | 0 | 0 | 0 |
| 10 | 9 | 1 | 0 | 0 | 1 |
| 11 | 10 | 1 | 0 | 1 | 0 |
| 12 | 11 | 1 | 0 | 1 | 1 |
| 13 | 12 | 1 | 1 | 0 | 0 |
| 14 | 13 | 1 | 1 | 0 | 1 |
| 15 | 14 | 1 | 1 | 1 | 0 |
| 16 | 15 | 1 | 1 | 1 | 1 |

- The timing diagram of the 4-bit ripple UP counter is shown in the figure 3.23.
- The truth table of the 4-bit ripple UP counter is tabulated in Table 3.21.

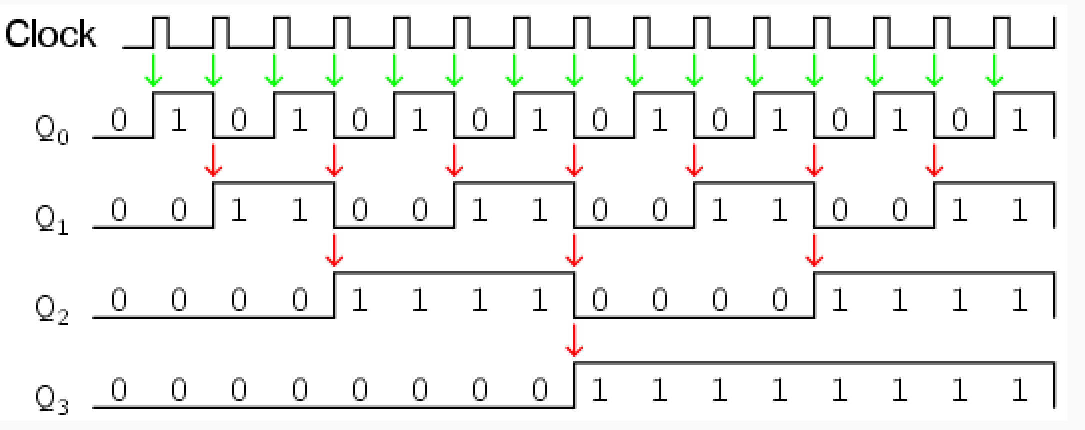

Clock
$Q_0$ 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1
$Q_1$ 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1
$Q_2$ 0 0 0 0 1 1 1 1 0 0 0 0 1 1 1 1
$Q_3$ 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1

**Fig.3.23 Timing Diagram of 4-bit Ripple UP Counter**

### 3.4.2 4-bit Down counter

- A 4 bit asynchronous or Ripple DOWN counter with T flip flop is shown in Figure 3.24.
- Since it is a 4-bit down counter, it has 4 flip flops and the output states of the counter are 24 (16 states). That is, the states are from 15 (1111) to 0 (0000).

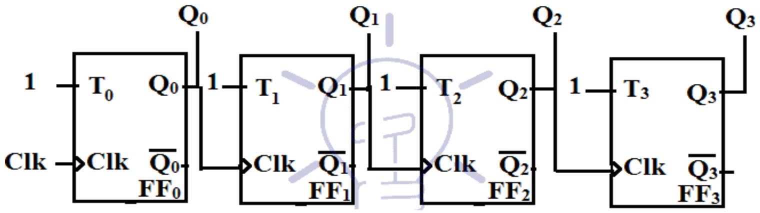


**Fig.3.24 Logic Diagram of 4-bit Ripple DOWN Counter**

- The output of the flip flop is Q3Q2Q1Q0. Q3 is the MSB and Q0 is the LSB.

- That means the flip flops will toggle at each active edge or positive edge of the clock signal. The clock input is connected to first flip flop.

- For 4-bit ripple down counter, the other flip flops in counter receive the clock signal input from Q output of previous flip flop.

- The output of the first flip flop will change, when the positive edge of clock signal occurs. For example, if the present count = 3, then the up counter will calculate the next count as 2.

- As the Q0 of FF0 is connected to the clock input of FF1, the Q1 toggles for every positive edge of Q0.

- As the Q1 of FF1 is connected to the clock input of FF1, the Q2 toggles for every positive edge of Q1.

- As the Q2 of FF1 is connected to the clock input of FF1, the Q3 toggles for every positive edge of Q2.

- So the down counter counts from 15, 14, 13…0 i.e. (15 to 0) or 1111 to 0000.

- The timing diagram of the 4-bit ripple DOWN counter is shown in the figure 3.25.

  - The truth table of the 4-bit ripple DOWN counter is tabulated in Table 3.22.

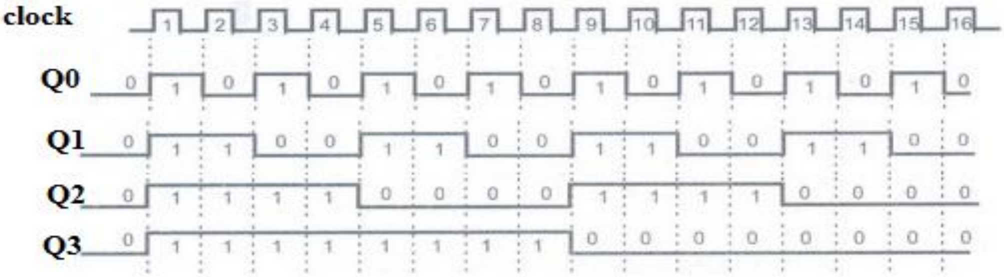
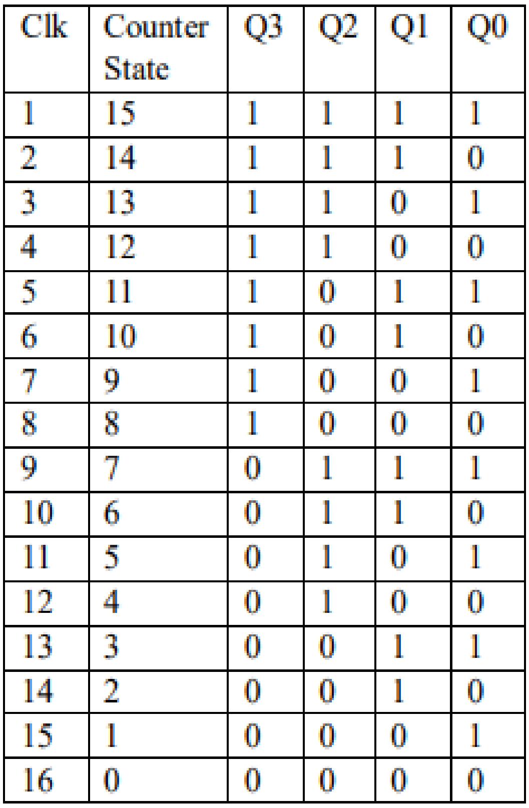


**Fig.3.25 Timing Diagram of 4-bit Ripple DOWN Counter**

**Table.3.22 Truth table of 4-bit DOWN Counter (Ripple)**



| Clk | Counter State | Q3 | Q2 | Q1 | Q0 |
|---|---|---|---|---|---|
| 1 | 15 | 1 | 1 | 1 | 1 |
| 2 | 14 | 1 | 1 | 1 | 0 |
| 3 | 13 | 1 | 1 | 0 | 1 |
| 4 | 12 | 1 | 1 | 0 | 0 |
| 5 | 11 | 1 | 0 | 1 | 1 |
| 6 | 10 | 1 | 0 | 1 | 0 |
| 7 | 9 | 1 | 0 | 0 | 1 |
| 8 | 8 | 1 | 0 | 0 | 0 |
| 9 | 7 | 0 | 1 | 1 | 1 |
| 10 | 6 | 0 | 1 | 1 | 0 |
| 11 | 5 | 0 | 1 | 0 | 1 |
| 12 | 4 | 0 | 1 | 0 | 0 |
| 13 | 3 | 0 | 0 | 1 | 1 |
| 14 | 2 | 0 | 0 | 1 | 0 |
| 15 | 1 | 0 | 0 | 0 | 1 |
| 16 | 0 | 0 | 0 | 0 | 0 |

### 3.4.3 Mod-10 Counter or Decade Counter or BCD counter

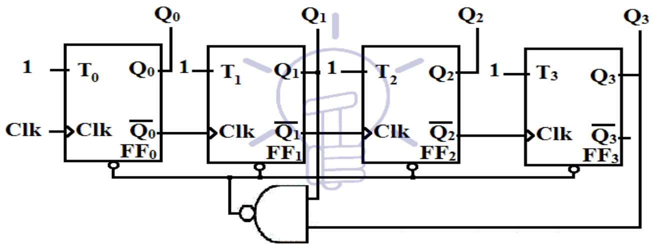


**Fig.3.26 Logic Diagram of Mod-10 or Decade Counter (Ripple Counter)**

- The logic diagram of MOD-10 counter or Decade counter (Ripple) is shown in Figure 3.26.
- This counter has 4 flip flops and counts from 0 to 9. This counter has 10 output states and the output states of the counter are from 0000 to 1001.
- Since it has 10 output states, it is called MOD-10 counter or Decade counter.
- The binary value of 10 is $Q_3Q_2Q_1Q_0$=1010. Both outputs Q1and Q3 are equal to logic "l" and the output from the NAND gate is logic "0" and this output is connected to the CLEAR

(CLR) inputs of all the T Flip-flops and clears all the flip flops.

- This signal causes all of the Q outputs to be reset back to binary "0000". Then the counter again starts counting from 0000 till it reaches 1001.
- The truth table of MOD-10 counters or Decade counter (Ripple) is shown in Table 3.23.
- The timing diagram of MOD-10 counter or Decade counter (Ripple) is shown in Figure 3.27.

**Table.3.23 Truth table of Mod-10 or Decade Counter**

| Clock Pulse | Binary Output of the Mod-10 Counter | | | | Decimal Value |
|---|---|---|---|---|---|
| | $Q_D$ | $Q_C$ | $Q_B$ | $Q_A$ | |
| 1 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | 1 | 1 |
| 3 | 0 | 0 | 1 | 0 | 2 |
| 4 | 0 | 0 | 1 | 1 | 3 |
| 5 | 0 | 1 | 0 | 0 | 4 |
| 6 | 0 | 1 | 0 | 1 | 5 |
| 7 | 0 | 1 | 1 | 0 | 6 |
| 8 | 0 | 1 | 1 | **1** | 7 |
| 9 | 1 | 0 | 0 | 0 | 8 |
| 10 | 1 | 0 | 0 | 1 | 9 |
| 11 | Counter Resets its Outputs back to Zero | | | | |

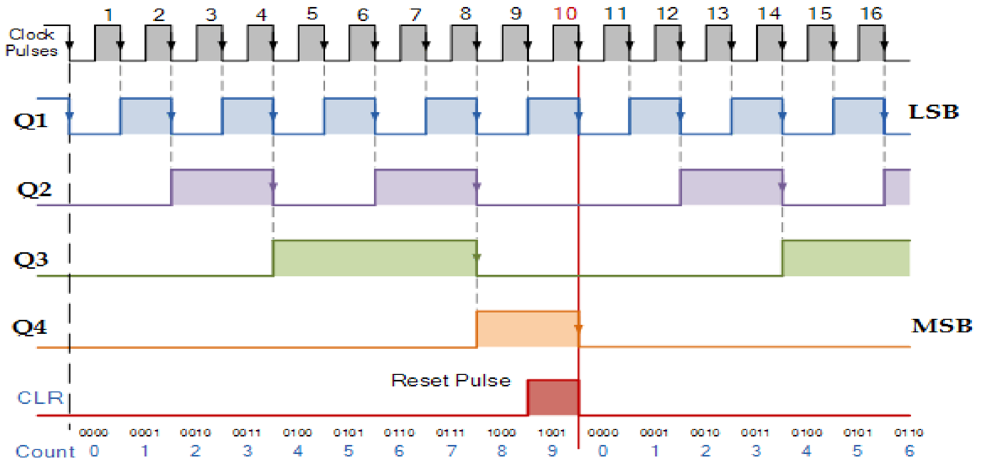


**Fig.3.27 Timing Diagram of Mod-10 or Decade Counter (Ripple Counter)**

## 3.5 Design of Synchronous Counter

### 3.5.1 4-bit Synchronous UP counter

- **Step 1:** To design a synchronous up counter, the number of flip flops required=4. The number of output states are 2^4=16 (i.e.; from 0 to 15).
- **Step 2:** Construct a state table or excitation table of the counter with excitation table of the T flip-flop. The excitation table for T flip flop is:

**T Flip Flop Excitation Table**

| Present state<br>$Q_n$ | Next state<br>$Q_{n+1}$ | T |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

- Let Q4Q3Q2Q1 represent the present state and Q4*Q3*Q2*Q1* represent the next state.
- When Q4 =0 which is present state and Q4*=0 which is next state then T4 become 0. Similarly, if Q4 is 0 and Q4* is 1 then T3 becomes 1. In similar way it goes on. The excitation table of 4-bit Synchronous UP counter is tabulated in Table 3.24.

**Table.3.24 Excitation Table of 4-bit Synchronous UP counter**

| Present State | | | | Next State | | | | Flip Flop | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $Q_4$ | $Q_3$ | $Q_2$ | $Q_1$ | Q4* | Q3* | Q2* | Q1* | $T_4$ | $T_3$ | $T_2$ | $T_1$ |
| 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 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |

- **Step 3:** After making the state table, derive the Boolean expressions for T4, T3, T2,T1 in terms of Q4,Q3, Q2,Q1 for the design of the counter.

**K-Map**

For $T_4$ Flip flop,

Q2Q1

Q4 Q3

| | | | |
|---|---|---|---|
| | | | |
| | | 1 | |
| | | 1 | |
| | | | |

$T_4 = Q_1.Q_2.Q_3$

For $T_3$ Flip flop,

Q2Q1

Q4 Q3

| | | | |
|---|---|---|---|
| | | 1 | |
| | | 1 | |
| | | 1 | |
| | | 1 | |

$T_3 = Q_1.Q_2$

For $T_2$ Flip flop,

Q2Q1

Q4 Q3

| | | | |
|---|---|---|---|
| | 1 | 1 | |
| | 1 | 1 | |
| | 1 | 1 | |
| | 1 | 1 | |

$T_2 = Q_1$

For $T_1$ Flip flop,

Q2Q1

Q4 Q3

| | | | |
|---|---|---|---|
| 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 |

$T_1=1$

- **Step 4:** According to the equations derived from K map, create the design for 4 bit synchronous up counter. The logic diagram and timing diagram of 4-bit Synchronous UP counter is shown in Figure 3.28 and 3.29 respectively.

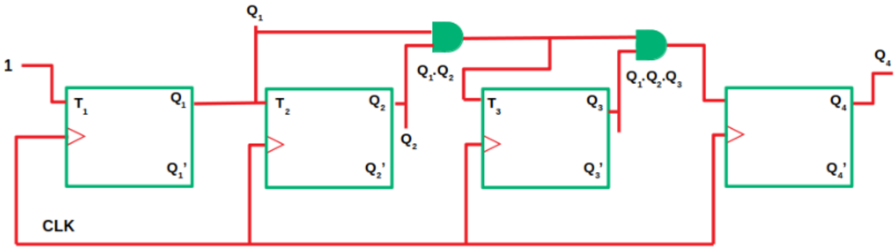


**Fig.3.28 Logic Diagram of 4-bit Synchronous UP Counter**

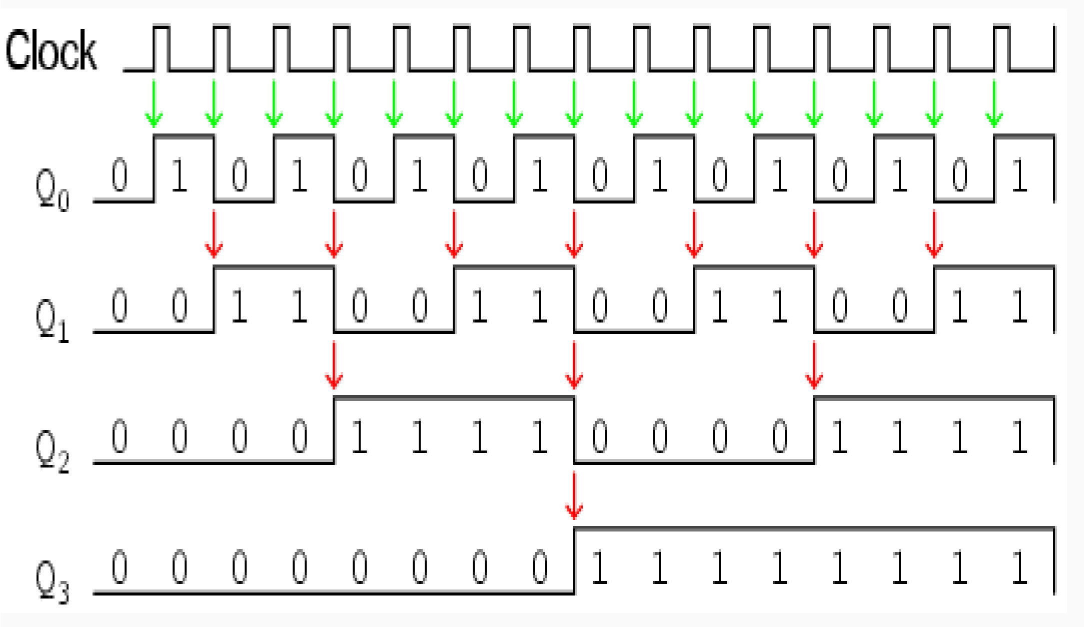


**Fig.3.29. Timing Diagram of 4-bit Synchronous UP Counter**

### 3.5.2 4-bit Down counter

- **Step 1:** To design a 4-bit synchronous DOWN counter, the no. of flip flops required =4 and the output states are $2^4$=16 (from 1111 to 0000).

- **Step 2:** After that, we need to construct a state table with excitation table. The excitation table of T flip flop is

**T Flip Flop Excitation Table**

| **Present state**<br>**$Q_n$** | **Next state**<br>**$Q_{n+1}$** | **T** |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

- Let Q4Q3Q2Q1 represent the present state and Q4*Q3*Q2*Q1* represent the next state.
- For instance, when Q4 =0 which is present state and Q4*=0 which is next state then T4 become 0. Similarly, if Q4 is 0 and Q4* is 1, then T3 becomes 1. The excitation table of 4-bit Synchronous DOWN counter is tabulated in Table 3.25.

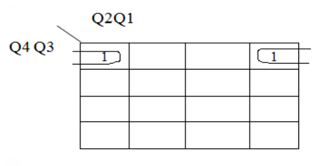
**Table.3.25 Excitation Table of 4-bit Synchronous DOWN counter**

| Present State | | | | Next State | | | | Flip Flop Inputs | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $Q_4$ | $Q_3$ | $Q_2$ | $Q_1$ | Q4* | Q3* | Q2* | Q1* | $T_4$ | $T_3$ | $T_2$ | $T_1$ |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

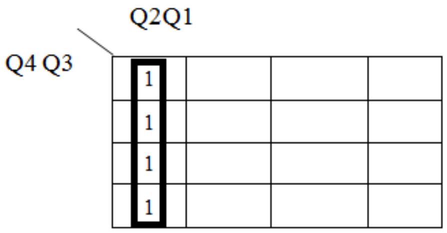
- **Step 3:** After making the state table, derive the Boolean expressions for T4, T3, T2,T1 in terms of Q4,Q3, Q2,Q1 for the design of the counter.

*For T4 Flip flop,*



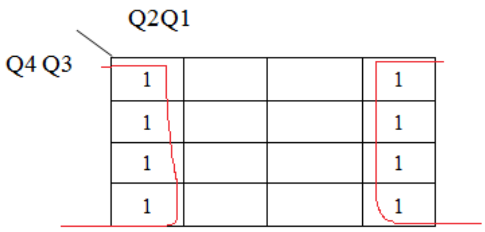
$T_4 = Q_1'.Q_2'.Q_3'$

For $T_3$ Flip flop,



$T_3 = Q_1.Q_2$

For T2 Flip flop,



$T_2 = Q_1$

For $T_1$ Flip flop,

| Q4 Q3 \ Q2Q1 | | | | |
|---|---|---|---|---|
| | 1 | 1 | 1 | 1 |
| | 1 | 1 | 1 | 1 |
| | 1 | 1 | 1 | 1 |
| | 1 | 1 | 1 | 1 |

$T_1$=1

**Step 4:** From the above equations, the 4-bit synchronous down counter is designed as in Figure. A common clock signal is fed to all the flip flops simultaneously. The logic diagram and timing diagram of 4-bit Synchronous DOWN counter is shown in Figure 3.30 and 3.31 respectively.

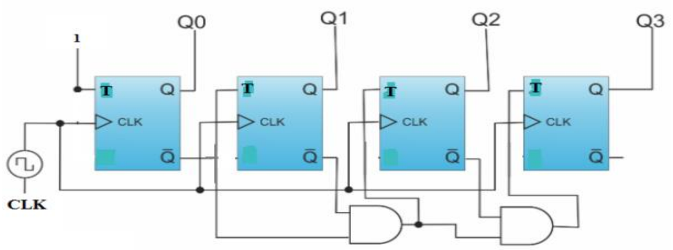


**Fig.3.30 Logic Diagram of 4-bit Synchronous DOWN Counter**

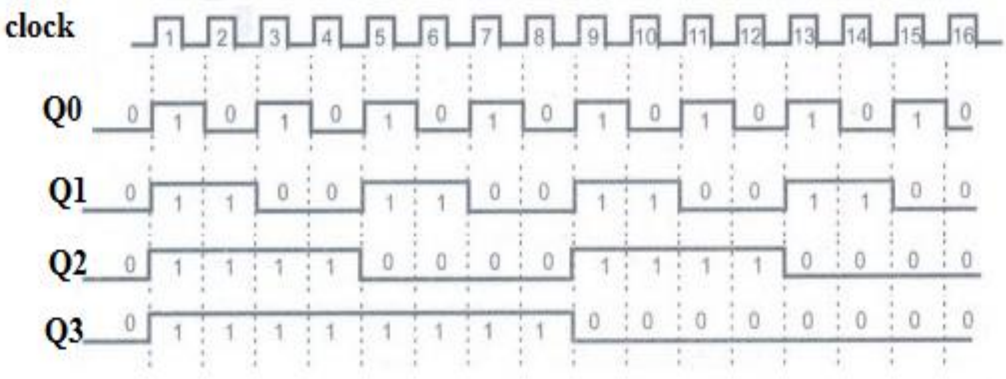


**Fig.3.31 Timing Diagram of 4-bit Synchronous DOWN Counter**

### 3.5.3 Mod-10 Synchronous Counter or Decade Counter or BCD counter

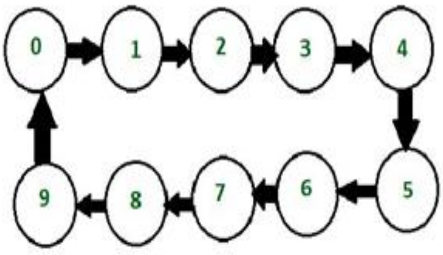
**Step 1:** Decision for number of flip-flops. For designing Mod-10 counter, N= 10 and number of FF required is 4 for Mod-10 counter.

**Step2:** Write excitation table of T Flip flop.

| Present state<br>$Q_n$ | Next state<br>$Q_{n+1}$ | T |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |

| | | |
|---|---|---|
| 1 | 0 | 1 |
| 1 | 1 | 0 |

**Step 3 :** Draw state diagram and circuit excitation table



**Circuit excitation table**

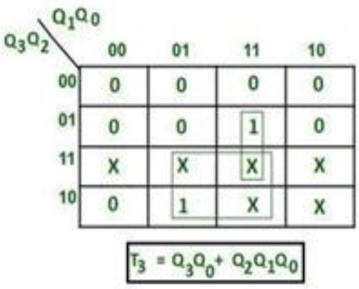
- Here Q4 Q3 Q2 Q1 are present states of four flip-flops and Q3* Q2* Q1* Q0* are next counting state of 4 Flip flops.
- If there is a transition in current state i.e; if Q3 value changes from 0 to 1 or1 to 0 then there's corresponding T(toggle) bit is written as 1 otherwise 0.
- The excitation table of Mod-10 Synchronous Counter or Decade Counter is tabulated in Table 3.26.

**Table.3.26 Excitation table of Mod-10 Synchronous Counter or Decade Counter**

| **Present State** | | | | **Next State** | | | | **Flip Flop Inputs** | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **$Q_4$** | **$Q_3$** | **$Q_2$** | **$Q_1$** | **Q4*** | **Q3*** | **Q2*** | **Q1*** | **$T_4$** | **$T_3$** | **$T_2$** | **$T_1$** |
| 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 |

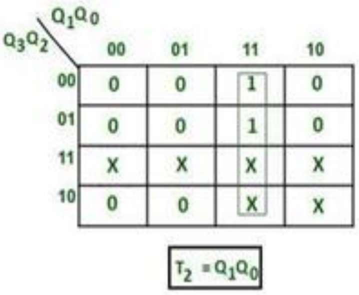
| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | 1 | 0 | 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 | 0 | 1 | 0 | 0 | 1 |

**Step 4:** Create Karnaugh map for each FF input in terms of flip-flop outputs. Simplify the K-map



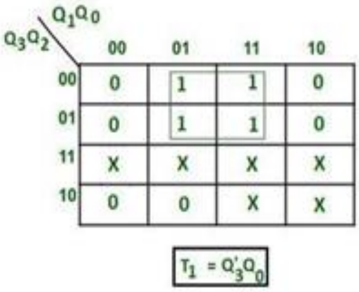
| $Q_3Q_2$ \ $Q_1Q_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 0 | 0 | 0 |
| 01 | 0 | 0 | 1 | 0 |
| 11 | X | X | X | X |
| 10 | 0 | 1 | X | X |

$T_3 = Q_3Q_0 + Q_2Q_1Q_0$



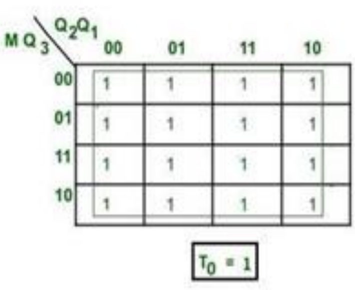
| $Q_3Q_2$ \ $Q_1Q_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 0 | 1 | 0 |
| 01 | 0 | 0 | 1 | 0 |
| 11 | X | X | X | X |
| 10 | 0 | 0 | X | X |

$T_2 = Q_1Q_0$



| $Q_3Q_2$ \ $Q_1Q_0$ | 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 |

$T_0 = 1$

**Step 5 :** Draw the logic diagram. The logic diagram and timing diagram of Decade Counter or MOD-10 Synchronous Counter is shown in Figure 3.32 and 3.33 respectively.

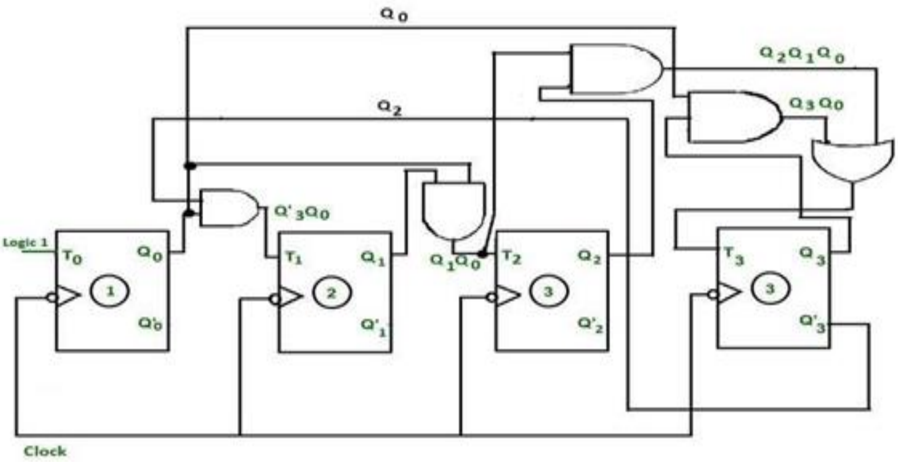


**Fig.3.32 Logic Diagram of Decade Counter or MOD-10 Synchronous Counter**

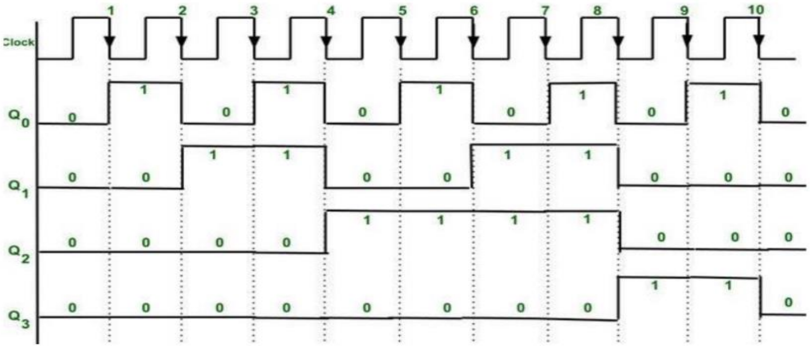


**Fig.3.33 Timing Diagram of Decade Counter or MOD-10 Synchronous Counter**

### 3.6 Johnson Counter or Twisted Ring Counter

- Johnson counter is also called Twisted Ring Counter.
- The logic diagram of Johnson Counter is shown in Figure 3.34.
- The Johnson Ring Counter or “Twisted Ring Counters”, is a counter with the inverted output Q of the last flip-flop fed back to the input D of the first flip-flop.
- ***The output states of the ring counter=2* No. of flip flops used.***
- So a “n-stage” Johnson counter will circulate a single data bit giving sequence of 2n different states and can therefore be considered as a “**mod-2n counter**”.
- Initially clear signal is applied to all the flip-flops to clear the output of all the flip flops. Hence the output is 0000.
- As the inverted output Q is connected to the input D, the next state is 1000. Hence the output states are “1000”, “1100”, “1110”, “1111”, “0111”, “0011”, “0001”, “0000” .

- ➢ The output states keep on repeating.
- ➢ The truth table of Johnson Counter is tabulated in Table 3.27.
- ➢ The timing diagram of Johnson Counter is shown in Figure 3.35.

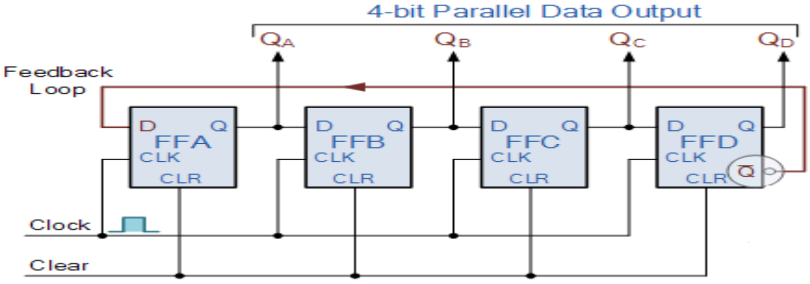


**Fig.3.34 Logic Diagram of Johnson Counter**

**Table.3.27 Truth Table of a 4-bit Johnson Ring Counter**

| Clock Pulse No | $Q_A$ | $Q_B$ | $Q_C$ | $Q_D$ |
|---|---|---|---|---|
| 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 |

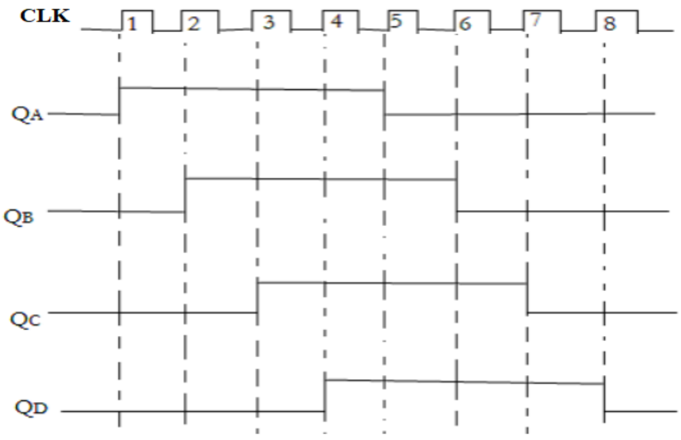


**Fig.3.35 Timing Diagram of Johnson Counter**

### 3.7 Ring Counter

- The synchronous Ring Counter must be preset so that exactly one data bit in the register is set to logic “1” with all the other bits reset to “0”.
- ***The output states of the ring counter= No. of flip flops used.***
- To achieve this, a “CLEAR” signal is firstly applied to all the flip-flops together in order to “RESET” their outputs to a logic “0” level and then a “PRESET” pulse is applied to the input of the first flip-flop (FFA ) before the clock pulses are applied. This then places a single logic “1” value into the circuit of the ring counter.
- So on each successive clock pulse, the counter circulates the same data bit between the four flip-flops over and over again around the “ring” every fourth clock cycle. This type of data movement is called “rotation”.
- The logic diagram and timing diagram of Ring Counter is shown in Figure 3.36 and 3.37 respectively.
- The truth table of Ring Counter is tabulated in Table 3.28.

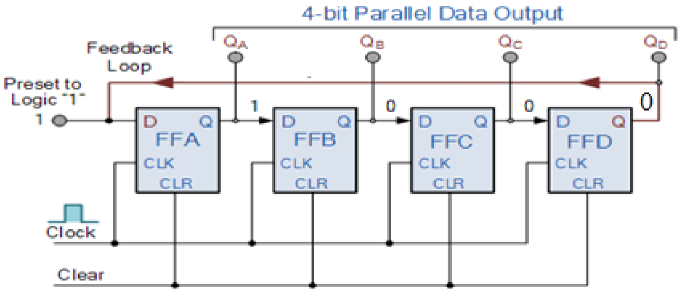


**Fig.3.36 Logic Diagram of Ring Counter**

**Table.3.28 Truth Table of a Ring Counter**

| Clock Pulse no. | QA | QB | QC | QD |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 2 | 0 | 0 | 1 | 0 |
| 3 | 0 | 0 | 0 | 1 |
| 4 | Again the same states will repeat | | | |

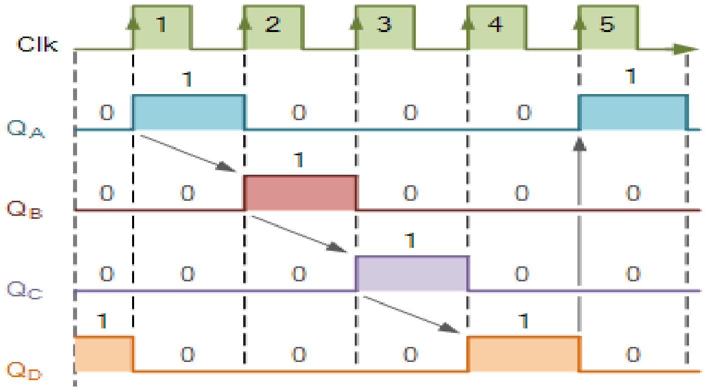


Fig.3.37 Timing Diagram of Ring Counter

## 3.8 Design of registers

### 3.8.1 Buffer register

- The buffer register is the simplest of all the registers. The logic diagram of the buffer register is shown in Figure 3.38.
- It simply stores the binary word.
- The buffer may be a controlled buffer.
- Most of the buffer registers use D flip-flops.
- The binary word to be stored is applied to the data terminals. On the application of clock pulse, the output words become the same as the word applied at the input terminals.
- That is, the input word is loaded into the register by the application of clock pulse.

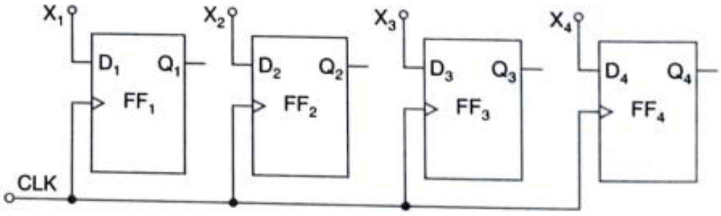


Fig.3.38 Logic Diagram of Buffer Register

- When the positive clock edge arrives, the stored word becomes: Figure 6.51 Logic diagram of a 4-bit buffer register.

$$Q_4Q_3Q_2Q_1 = X_4\,X_3\,X_2\,X_1$$

$$Q=X$$

### 3.8.2 Control Buffer Register

- The logic diagram of the Control buffer register is shown in Figure 3.39.

- If $\overline{CLR}$ goes LOW, all the FFs are RESET and output becomes, Q = 0000.
- When $\overline{CLR}$ is HIGH, the register is ready for action.
- LOAD is the control input.
- When LOAD is HIGH, the data bits X can reach the D inputs of FFs. At the positive-going edge of the next clock pulse. the register is loaded, i.e.

$$Q_4Q_3Q_2Q_1 = X_4\,X_3\,X_2\,X_1$$

$$Q=X$$

- When LOAD is LOW, output is feed back to its data input. Therefore, data is circulated or retained as each clock pulse arrives.

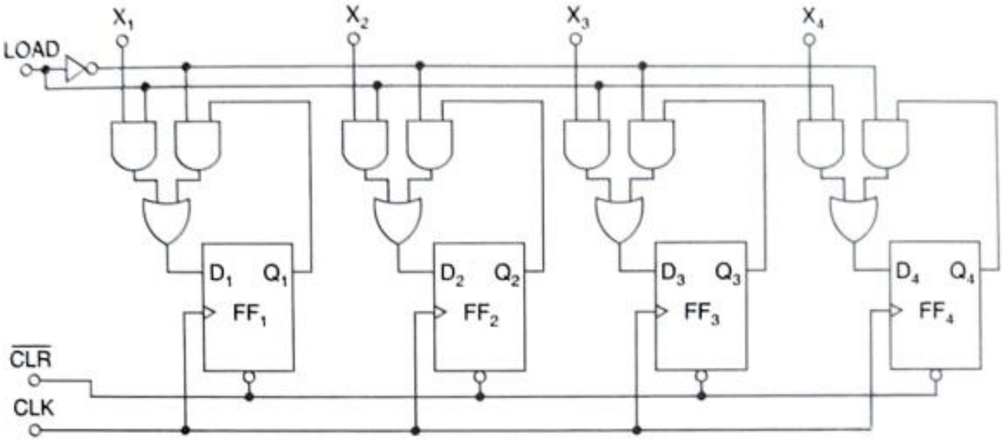


**Fig.3.39 Logic Diagram of Control Buffer Register**

### 3.8.3 Shift Registers

There are four types of Shift registers. They are:

- Serial In - Serial Out (SISO) Shift Register
- Serial In - Parallel Out (SIPO) Shift Register
- Parallel In - Serial Out (PISO) Shift Register
- Parallel In - Parallel Out (PIPO) Shift Register

**Serial In - Serial Out (SISO) Shift Register**

- The logic diagram of Serial-In Serial-Out (SISO) Shift Register is shown in Figure 3.40.
- Let all the flip-flop be initially in the reset condition i.e. Q3 = Q2 = Q1 = Q0 = 0. If an

entry of a four bit binary number 1 1 1 1 is made into the register, this number should be applied to Din bit with the LSB bit applied first. The D input of FF-3 i.e. D3 is connected to serial data input Din. Output of FF-3 i.e. Q3 is connected to the input of the next flip-flop i.e. D2 and so on.

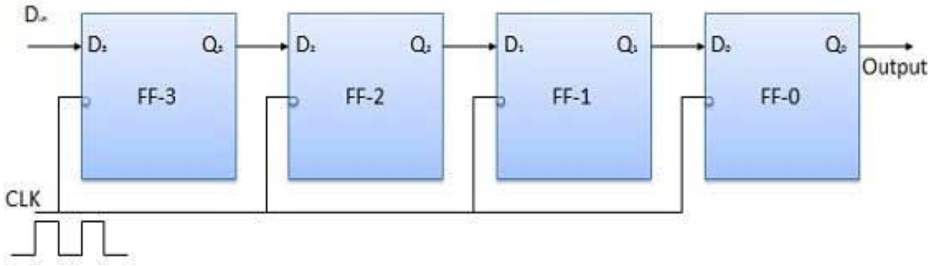


**Fig.3.40**. **Logic Diagram of Serial In Serial Out Shift Register**

**Serial In - Parallel Out (SIPO) Shift Register**

- ➢ The logic diagram of Serial-In Parallel-Out (SISO) Shift Register is shown in Figure 3.41.
- ➢ In such types of operations, the data is entered serially and taken out in parallel fashion.
- ➢ Data is loaded bit by bit. The outputs are disabled as long as the data is loading.
- ➢ As soon as the data loading gets completed, all the flip-flops contain their required data, the outputs are enabled so that all the loaded data is made available over all the output lines at the same time.
- ➢ 4 clock cycles are required to load a four bit word. Hence the speed of operation of SIPO mode is same as that of SISO mode.

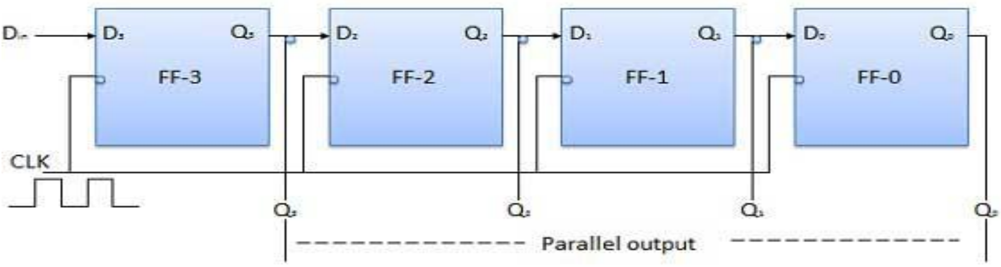


**Fig.3.41**. **Logic Diagram of Serial In Parallel Out Shift Register**

**Parallel In - Serial Out (PISO) Shift Register**

- ➢ The logic diagram of Parallel-In Serial-Out (SISO) Shift Register is shown in Figure 3.42.

- Data bits are entered in parallel fashion and the output is taken in serial fashion.
- There are two modes in which this circuit can work namely - shift mode or load mode.

**Load Mode**

- When the Shift/$\overline{Load}$ line is LOW (0), the AND gate 2, 4 and 6 become active. They will pass B1, B2, B3 bits to the corresponding flip-flops.
- On the negative edge of clock pulse, the binary input B0, B1, B2, B3 will get loaded into the corresponding flip-flops. Thus parallel loading takes place.

**Shift Mode**

- When the Shift/$\overline{Load}$ line is HIGH (1), the AND gates 1,3 and 5 become active. Therefore the shifting of data from left to right bit by bit takes place on application of the clock pulse. Thus the parallel in serial out operation takes place.

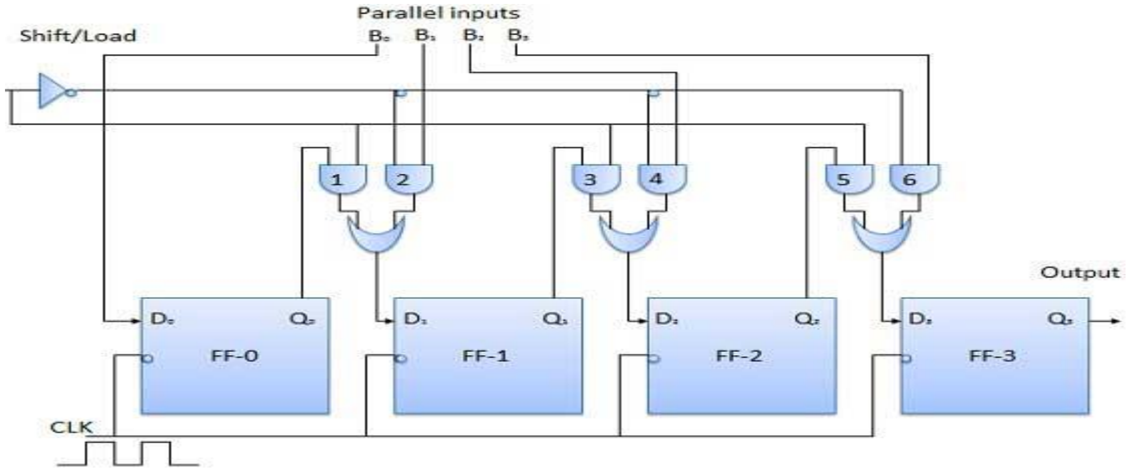


**Fig.3.42. Logic Diagram of Parallel In Serial Out Shift Register**

**Parallel In - Parallel Out (PIPO) Shift Register**

- The logic diagram of Parallel-In Parallel-Out (SISO) Shift Register is shown in Figure 3.43.
- In this mode, the 4 bit binary inputs B0, B1, B2, B3 is applied to the data inputs D0, D1, D2, D3 respectively of the four flip-flops.
- As soon as a negative clock edge is applied, the input binary bits will be loaded into the flip-flops simultaneously.
- The loaded bits will appear simultaneously as the output.

- Only one clock pulse is required to load all the bits.

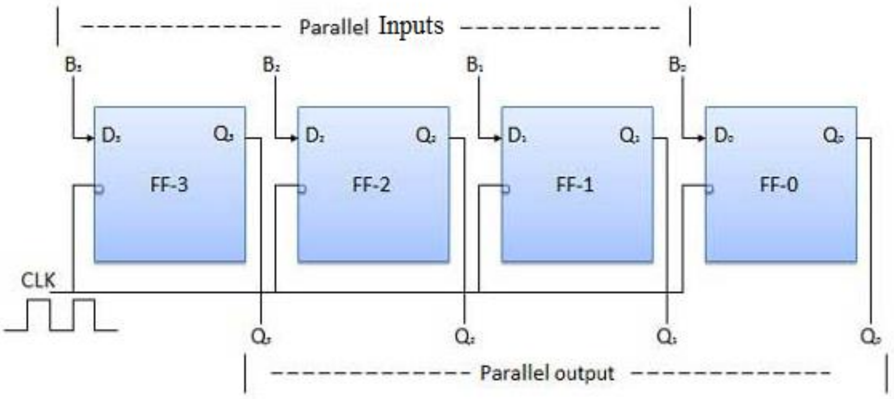


**Fig.3.43**. **Logic Diagram of Parallel In Parallel Out Shift Register**

### 3.8.4 Bi-directional shift Register

- A bi-directional, or reversible shift register is one in which the data can be shift either left or right. A four-bit bi-directional shift register using D-flip-flops is shown in Figure 3.44.
- Right/$\overline{Left}$ is the control signal.
- When Right/$\overline{Left}$ is 1, the circuit works as a shift right register.
- When Right/$\overline{Left}$ is 0, the circuit works as a shift left register.
- This bidirectional operation is achieved using the control signal, two AND gates and one OR gate.
- A HIGH on the Right/$\overline{Left}$ enables the AND gates G1,G2, G3,G4 and disables the G5,G6,G7,G8. When a clock pulse occurs, then the data bits are effectively shifted one position to the right.
- A LOW on the Right/$\overline{Left}$ enables the AND gates G5,G6,G7,G8 and disables the G1,G2,G3,G4. When a clock pulse occurs, then the data bits are effectively shifted one position to the left.
- Hence this circuit is called as bidirectional shift register.

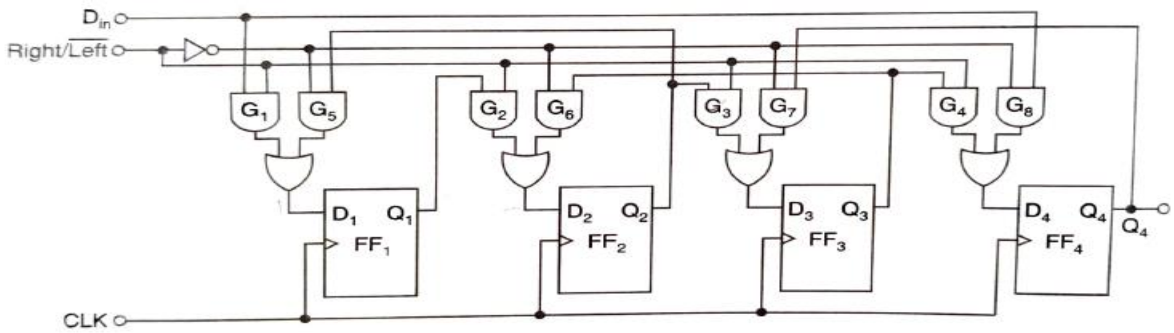


**Fig.3.44. Logic Diagram of a Bidirectional Shift Register**

### 3.8.5 Universal shift Register

- The logic diagram of Universal Shift register is shown in Figure 3.45.
- A Universal Shift register performs right shift, left shift and parallel load operation.
- A clear control is to clear the register to 0000.
- A CP input for clock pulse is to synchronize all operations
- A shift-right control to enable the shift-right operation and the serial input and output lines associated with the shift right.
- A shift-left control to enable the shift-left operation and the serial input and output lines associated with the shift left.
- A parallel-load control to enable a parallel transfer and n input lines associated with the parallel transfer.
- It consists of four D-flip-flops and four multiplexers which has two selection lines. The S1 and S0 selection lines control the mode of operation of the register which is specified in Table.3.29.

**Table.3.29 Truth Table of Universal Shift Register**

| **Mode Control** | | **Register Operation** |
|---|---|---|
| **S1** | **S0** | |
| 0 | 0 | Locked state<br>(No change) |
| 0 | 1 | Shift right |
| 1 | 0 | Shift left |
| 1 | 1 | Parallel Load |

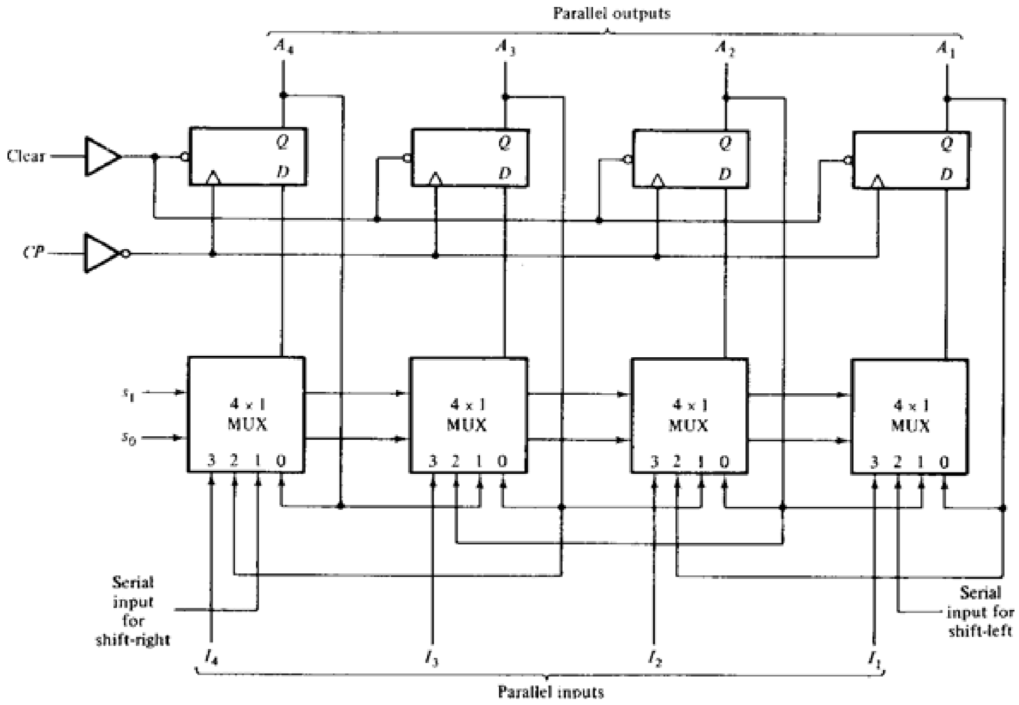


**Fig.3.45 Logic Diagram of 4-Bit Universal Shift Register**

- If the selected lines S1= 0 and S0 = 0, then this register doesn’t operate in any mode. That means it will be in a Locked state or no change state even though the clock pulses are applied.
- If the selected lines S1 = 0 and S0 = 1, then this register shifts the data to right and hence performs the shift-right operation.
- If the selected lines S1 = 1 and S0 = 0, then this register transfers or shifts the data to left and stores the data.
- If the selected lines S1 = 1 and S0 = 1, then this register loads the data in parallel. Hence it performs the parallel loading operation and stores the data.

**Additional Resources:**

1. Sequential circuits, NAND Latch and NOR Latch: https://www.youtube.com/watch?v=ibQBb5yEDlQ (NPTEL-Lecture 16)
2. Flip flops: https://www.youtube.com/watch?v=xhH3Bk6BeKg
3. Conversion of flip flops: https://www.youtube.com/watch?v=NytpDrESCV8 https://www.youtube.com/watch?v=JEQVPKd3cUw https://www.youtube.com/watch?v=JNV7p4xZ9TY
4. Asynchronous counters: https://www.youtube.com/watch?v=eEeBh8jfDjg

https://www.youtube.com/watch?v=noUcCs2zNaI

https://www.youtube.com/watch?v=fKVZpupyP_o&t=7s

5. Design of Synchronous Counters: https://www.youtube.com/watch?v=5vkWccb7uO4

   https://www.youtube.com/watch?v=o9R1egALhNs

6. Registers: https://www.youtube.com/watch?v=zoEeQgQkPLA

   https://www.youtube.com/watch?v=AEGzpMlOsvc&t=6s

7. NPTEL Course on Digital Circuits by Prof. SantanuChattopadhyay , IIT Kharagpurhttps://onlinecourses.nptel.ac.in/noc21_ee75/preview

## Questions

### 3.1 Classification of sequential circuits

**Objective Questions:**

1. A sequential circuit is a circuit whose output depends on
   a) Current input
   b) Past input
   c) Circuit Internal State
   d) All of the above
2. The sequential circuits are
   a) Synchronous
   b) Asynchronous
   c) Both
   d) None
3. Which of the following is a key characteristic of a synchronous sequential circuit?
   (a) Output depends solely on the current input
   (b) All state transitions occur simultaneously
   (c) All state transitions are triggered by a common clock signal
   (d) Output depends on both current and past inputs
4. Which of the following is a key characteristic of an asynchronous sequential circuit?
   (a) Uses a global clock signal for synchronization.

   (b) Relies on a periodic clock signal for operation.

   (c) Does not use a global clock signal for synchronization.

   (d) Is always faster than synchronous circuits.

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Define a Sequential Circuit. Classify them. | L2 |
| 2 | Compare between Synchronous and Asynchronous Sequential Circuits. | L2 |

### 3.2 Latches and Flip flops

### 3.2.1 Operation of NAND latch

**Objective Questions:**

1. Latch is a device with ___________
   a) One stable state
   b) Two stable state
   c) Three stable state
   d) Infinite stable states
2. The NAND latch works when both inputs are ___________
   a) 1
   b) 0
   c) Inverted
   d) Don't cares

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Explain the operation of NAND Latch with logic diagram and truth table. | L2 |

### 3.2.2 Operation of NOR latch

**Objective Questions:**

1. The outputs of SR latch are ___________
   a) x and y
   b) a and b
   c) s and r
   d) q and q'
2. The NOR latch works when both inputs are ___________
   a) 1
   b) 0
   c) Inverted
   d) Don't cares

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Explain the operation of NOR Latch with logic diagram and truth table. | L2 |

### 3.2.3 RS flip-flop

**Objective Questions:**

1. **When both set and reset are disabled in S-R flip flop then the output will be __________**

a) Set

b) Reset

c) No change

d) Indeterminate

2. **The preset input is used to make output ______**

a) Q=1

b) Q=0

c) Invalid

d) No change

3. **The clear input is used to make output ______**

a) Q=1

b) Q=0

c) Invalid

d) No change

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Explain the operation of SR flip flop with logic diagram, truth table and timing diagram. | L2 |
| 2 | Illustrate the characteristic table and excitation table of SR flip flop | L2 |

### 3.2.4 D flip-flop

**Objective Questions:**

1. In which flip flop the present input will be the next output?

a) S-R

b) J-K

c) T

d) D

**2. When the clock input is low in a D flip flop then the input of the D flip flop is _______**

a) High

b) Low

c) No effect

d) None of the above

**3. When the clock input is high and D input is high then the output of a D flip flop will be _______**

a) High

b) Low

c) No effect

d) None of the above

**4. How many outputs does D-flip flop have?**

a) One

b) Two

c) Three

d) Four

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Explain the operation of D flip flop with truth table and timing diagram. | L2 |
| 2 | Illustrate the characteristic table and excitation table of D flip flop | L2 |

### 3.2.5 JK flip-flop

**Objective Questions:**

**1. When toggle condition occurs in JK flip flop?**

a. J=1, K=1

b. J=0, K=0

c. J=1, K=0

d. J=0, K=1

**2. The no-change conditions occur when ________ in JK flip flop**

a. J=1, K=1

b. J=0, K=0

c. J=1, K=0

d. J=0, K=1

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Explain the operation of JK flip flop with truth table and timing diagram. | L2 |
| 2 | Illustrate the characteristic table and excitation table of JKflip flop | L2 |

### 3.2.6 T flip-flop

**Objective Questions:**

1. When the clock input is high and T input is high then the output of a T flip flop will be ________
    a. Toggle
    b. Remains same as the previous output
    c. Remains high
    d. Remains low
2. What is the standard form of T flip flop?
    a. Trigger
    b. Toggle
    c. Trigger or Toggle
    d. None

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Explain the operation of T flip flop with truth table and timing diagram. | L2 |
| 2 | Illustrate the characteristic table and excitation table of T flip flop | L2 |

## 3.3 Conversion from one flip-flop to another flip- flop

### 3.3.1 SR to JK flip-flop

**Objective Questions:**

1. **What is the available flip flop in SR to JK flip flop conversion?**
    a) JK flip flop
    b) T Flip flop
    c) D flip flop

d) SR flip flop

2. **What is the required flip flop in SR to JK flip flop conversion?**

a) JK flip flop

b) T Flip flop

c) D flip flop

d) SR flip flop

3. **The equations for converting SR to JK flip flop are**

a) S=JQ and R= KQ

b) S=JQ' and R= KQ

c) S=Q' and R= KQ

d) S=JQ' and R= KQ’

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Convert SR flip flop to JK flip flop. | L3 |

### 3.3.2 JK to T flip-flop

**Objective Questions:**

1. **What is the available flip flop in JK to T flip flop conversion?**

a) JK flip flop

b) T Flip flop

c) D flip flop

d) SR flip flop

2. **What is the required flip flop in JK to T flip flop conversion?**

a) JK flip flop

b) T Flip flop

c) D flip flop

d) SR flip flop

3. **The equations for converting JK to T flip flop is**

a) J=T, K=T’

b) J=T’, K=T

c) J=T, K=T

d) J=T’, K=T’

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Convert JK flip flop to T flip flop. | L3 |

### 3.3.3 SR to D flip-flop

**Objective Questions:**

1. **What is the available flip flop in SR to D flip flop conversion?**
   a) JK flip flop
   b) T Flip flop
   c) D flip flop
   d) SR flip flop
2. **What is the required flip flop in SR to D flip flop conversion?**
   a) JK flip flop
   b) T Flip flop
   c) D flip flop
   d) SR flip flop
3. **The equations for converting SR to D flip flop is**
   a) S=D, R=D’
   b) S=D’, R=D
   c) S=D, R=D
   d) S=D’, R=D’

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Convert SR flip flop to D flip flop. | L3 |

## 3.4 Design of 4-bit Ripple or Asynchronous Counters

### 3.4.1 4-bit up counter

**Objective Questions:**

1. **In digital logic, a counter is a device which ____________**
   a) Counts the number of outputs
   b) Stores the number of times a particular event or process has occurred
   c) Stores the number of times a clock pulse rises and falls
   d) Counts the number of inputs

2. **Ripple counters are also called ____________**

   a) SSI counters

   b) Asynchronous counters

   c) Synchronous counters

   d) VLSI counters

3. **For designing an n-bit up counter we need _________ flip-flops.**

   a) n

   b) n-1

   c) n+1

   d) None

4. **An n-bit up counter has _________ output states.**

   a) 2^n

   b) 3^n

   c) 4^n

   d) 2^n-1

5. **A 4-bit up counter counts from**

   a) 0 to 15

   b) 0 to 16

   c) 0 to 10

   d) 15 to 0

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Explain the operation of 4-bit UP Ripple Counter with neat timing diagram | L2 |

### 3.4.2 4-bit Down counter

**Objective Questions:**

1. **What happens to the parallel output word in an asynchronous binary down counter whenever a clock pulse occurs?**

   a) The output increases by 1

   b) The output decreases by 1

   c) The output word increases by 2

   d) The output word decreases by 2

2. **A 4-bit Down counter counts from**

a) 0 to 15

b) 0 to 16

c) 0 to 10

d) 15 to 0

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Explain the operation of 4-bit DOWN Ripple Counter. | L2 |

### 3.4.3 Mod-10 Counter

**Objective Questions:**

1. **BCD counter is also known as ____________**

   a) Parallel counter

   b) Decade counter or MOD-10 Counter

   c) Synchronous counter

   d) VLSI counter

2. **How many flip-flops are required to construct a decade counter?**

   a) 4

   b) 8

   c) 5

   d) 10

3. **The 4-bit Synchronous UP counter can be made into MOD-10 counter by connecting Q4 and Q2 to NAND gate and giving it to**

   a) Clear signal of all the T flip flops

   b) Preset Signal of all the T flip flops

   c) Clock signal of all the T flip flops

   d) None

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Explain the operation of 4-bit MOD-10 Ripple Counter or Decade Counter. | L2 |

## 3.5 Design of Synchronous Counter

### 3.5.1 4-bit up counter

**Objective Questions:**

1. **The equations for inputs of T flip flop for a 4-bit Synchronous up counter is**
   a. $T_4= Q_1.Q_2.Q_3,\ T_3= Q_1.Q_2,\ T_2= Q_1, T_1=1$
   b. $T_4= Q_1.Q_2,\ T_3= Q_1.Q_2,\ T_2= Q_1, T_1=1$
   c. $T_4= Q_1.Q_2.Q_3,\ T_3= Q_1.Q_2,\ T_2= 1, T_1=1$
   d. $T_4= Q_1.Q_2.Q_3,\ T_3= Q_1,\ T_2= Q_1, T_1=1$

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Design a 4-bit Synchronous UP Counter. | L3 |

### 3.5.2 4-bit Down counter

**Objective Questions:**

1. **The equations for inputs of T flip flop for a 4-bit Synchronous up counter is**
   a. $T_4= Q_1'.Q_2'.Q_3',\ T_3= Q_1'.Q_2',\ T_2= Q_1', T_1=1$
   b. $T_4= Q_1.Q_2,\ T_3= Q_1.Q_2,\ T_2= Q_1, T_1=1$
   c. $T_4= Q_1.Q_2.Q_3,\ T_3= Q_1.Q_2,\ T_2= 1, T_1=1$
   d. $T_4= Q_1.Q_2.Q_3,\ T_3= Q_1,\ T_2= Q_1, T_1=1$

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Design a 4-bit Synchronous DOWN Counter. | L3 |

### 3.5.3 Mod-10 Counter

**Objective Questions:**

1. **The output states of MOD-10 counter are from**
   a) 0 to 9
   b) 0 to 10
   c) 0 to 15
   d) 15 to 0

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Design a Decade Synchronous Counter or MOD-10 Counter. | L3 |

### 3.6 Johnson Counter

**Objective Questions:**

1. **The 4-bit Johnson Counter has**
   a) 4 flip-flops and 4 output states.
   b) 4 flip-flops and 8 output states.
   c) 4 flip-flops and 16 output states.
   d) 4 flip-flops and 0 output states.
2. **In the 4-bit Johnson Counter, _______ is connected as feedback to the first flip-flop D input.**
   a. Q4’
   b. Q4
   c. Q1
   d. Q2
3. **If the Johnson Counter is initially at 0000 state, then after 6th clock pulse the output is**
   a. 0011
   b. 0010
   c. 0111
   d. 1000

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Explain the operation of 4-bit Johnson Counter with logic diagram and truth table. | L2 |

### 3.7 Ring Counter

**Objective Questions:**

1. **What is the preset condition for a ring shift counter?**
   a) All FFs set to 1
   b) All FFs cleared to 0
   c) A single 0, the rest 1
   d) A single 1, the rest 0
2. **The 4-bit Ring Counter has**
   a. 4 flip-flops and 4 output states.
   b. 4 flip-flops and 8 output states.
   c. 4 flip-flops and 16 output states.
   d. 4 flip-flops and 0 output states.

3. **To operate correctly, starting a ring shift counter requires __________**

   a) Clearing all the flip-flops

   b) Presetting one flip-flop and clearing all others

   c) Clearing one flip-flop and presetting all others

   d) Presetting all the flip-flops

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Explain the operation of 4-bit Ring Counter with logic diagram and truth table. | L2 |

### 3.8 Design of registers

### 3.8.1 Buffer register

**Objective Questions:**

1. **In Buffer Register, which flip flop is used.**
   a. D flip flop
   b. T flip flop
   c. JK flip flop
   d. None.
2. **In Buffer Register, the output $Q_1Q_2Q_3Q_4$=**
   a. $X_1X_2X_3X_4$
   b. X1’X2X3X4
   c. $X_1X_2X_3X_4$’
   d. None

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Explain the operation of Buffer register with logic diagram. | L2 |

### 3.8.2 Control Buffer Register

**Objective Questions:**

1. **In Controlled Buffer register, the output $Q_1Q_2Q_3Q_4$=$X_1X_2X_3X_4$ when**
   a. LOAD=1
   b. LOAD=0
   c. Does not depend on LOAD
   d. None

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Explain the operation of Control Buffer register with logic diagram. | L2 |

### 3.8.3 Shift Register

**Objective Questions:**

1. **How can parallel data be taken out of a shift register simultaneously?**

   a) Use the Q output of the first FF

   b) Use the Q output of the last FF

   c) Tie all of the Q outputs together

   d) Use the Q output of each FF

2. **In which register the data is fed in serially and taken out parallel**

   a. SISO

   b. SIPO

   c. PISO

   d. PIPO

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Explain the operation of Serial-In Serial-Out Shift register with logic diagram. | L2 |
| 2 | Explain the operation of Serial-In Parallel-Out Shift register with logic diagram. | L2 |
| 3 | Explain the operation of Parallel-In Serial-Out Shift register with logic diagram. | L2 |
| 4 | Explain the operation of Parallel-In Parallel-Out Shift register with logic diagram. | L2 |

### 3.8.4 Bi-directional shift Register

**Objective Questions:**

1. **When the control input Right/$\overline{Left}$ is HIGH (1), then the register performs ________ operation.**

   a. Shift Left operation

   b. Shift Right operation

   c. Parallel Load

   d. None of the above

2. **When the control input Right/$\overline{Left}$ is LOW (0), then the register performs ________ operation.**

   a. Shift Left operation

b. Shift Right operation

c. Parallel Load

d. None of the above

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Explain the operation of 4-bit Bi-directional Shift register with logic diagram. | L2 |

### 3.8.5 Universal shift Register

**Objective Questions:**

1. **To operate correctly, starting a ring shift counter requires __________**

   a) Clearing all the flip-flops

   b) Presetting one flip-flop and clearing all others

   c) Clearing one flip-flop and presetting all others

   d) Presetting all the flip-flops

2. **When S1S0=00, the Universal Shift Register operates in**

   a) No change

   b) Shift Right

   c) Shift Left

   d) Parallel load.

**Descriptive Questions:**

| S.no. | Question | BL |
|---|---|---|
| 1 | Explain the operation of 4-bit Universal Shift register with logic diagram. | L2 |