# Flip-Flops

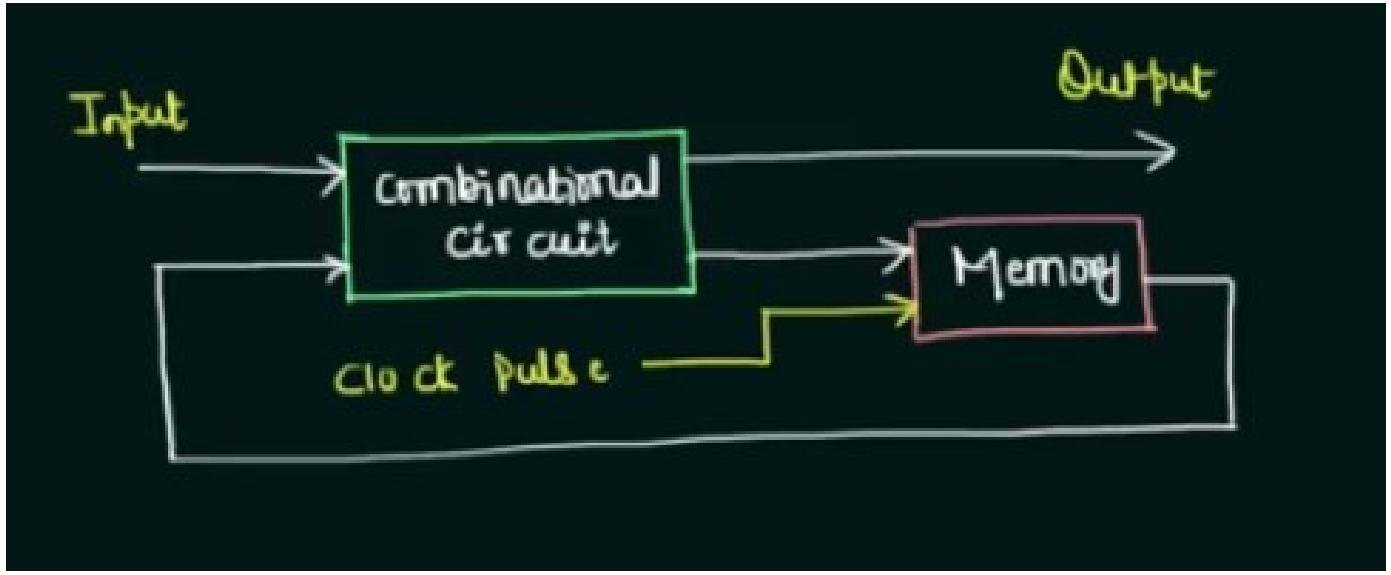
# Flip-Flop

- The state of a latch or flip-flop is switched by a change in the control input. This momentary change is called a *trigger*, and the transition it causes is said to trigger the flip-flop.
- Sequential  circuit has a feedback path from the outputs of the flip-flops to the input of the combinational circuit.
- The  inputs of the flip-flops are derived in part from the outputs of the same and other flip-flops.
- The state transitions of the latches start as soon as the clock pulse changes to the logic-1 level.
- The new state of a latch appears at the output while the pulse is still active.
- Flip-flop circuits are constructed in such a way as to make them operate properly when they are part of a sequential circuit that employs a common clock.
- The key to the proper operation of a flip-flop is to trigger it only during a signal *transition.*
- A clock pulse goes through two transitions: from 0 to 1 and the return from 1 to 0.

| LATCH | FLIP – FLOP |
|---|---|
| Latches do not require clock signal. | Flip – flops have clock signals |
| A latch is an asynchronous device. | A flip – flop is a synchronous device. |
| Latches are transparent devices i.e. when they are enabled, the output changes immediately if the input changes. | A transition from low to high or high to low of the clock signal will cause the flip – flop to either change its output or retain it depending on the input signal. |
| A latch is a Level Sensitive device (Level Triggering is involved). | A flip – flop is an edge sensitive device (Edge Triggering is involved). |
| Latches are simpler to design as there is no clock signal (no careful routing of clock signal is required). | When compare to latches, flip – flops are more complex to design as they have clock signal and it has to be carefully routed. This is because all the flip – flops in a design should have a clock signal and the delay in the clock reaching each flip – flop must be minimum or negligible. |
| The operation of a latch is faster as they do not have to wait for any clock signal. | Flip - flops are comparatively slower than latches due to clock signal. |
| The power requirement of a latch is less. | Power requirement of a flip – flop is more. |
| A latch works based on the enable signal. | A flip – flop works based on the clock signal. |

# Triggering Method

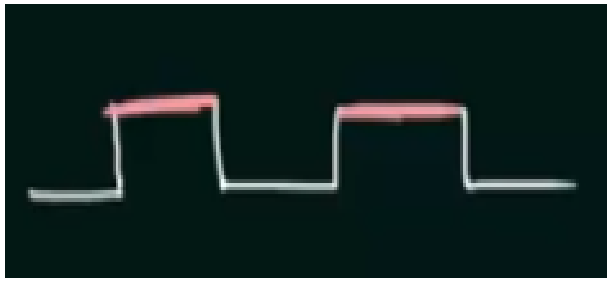
- **Memory**: flip-flop
- **Clock pulse**: control input
- **Trigger the memory element** →state stored in it is switched(change from one state to the other state depending upon the input and previous state in it(memory))
- **State of the circuit will be changed depending upon the clock pulse**



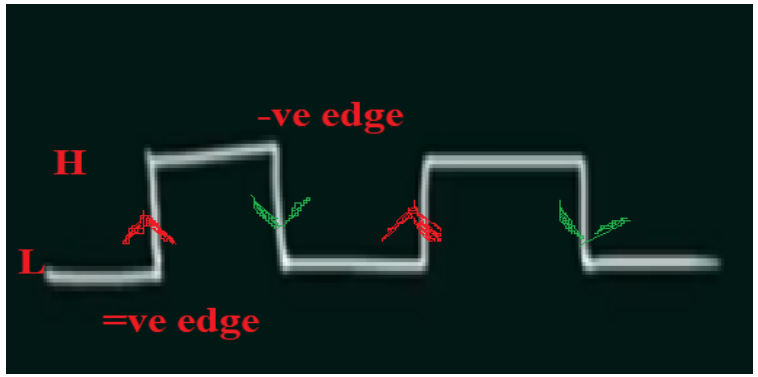
- Types:
  - Level triggering
  - Edge triggering
- **<u>Level Triggering:</u>**

Clock remains high then there will be a

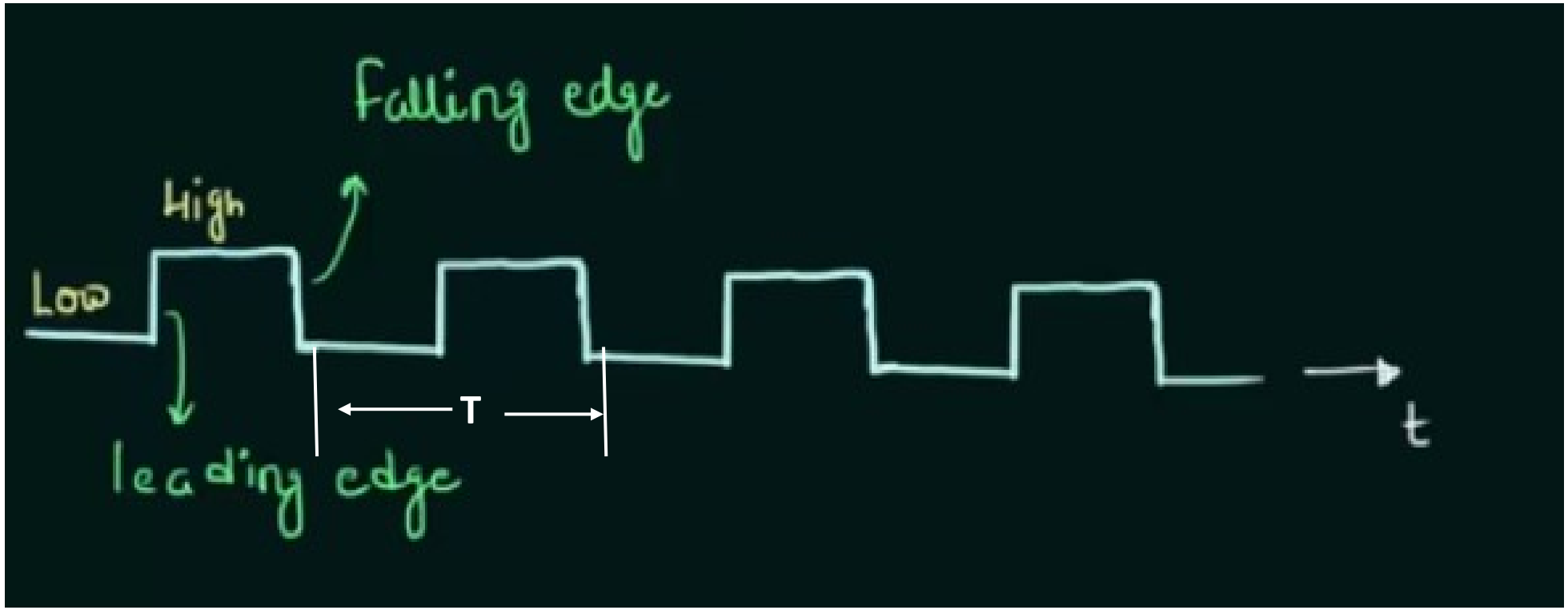
Transition in latch/flip-flop



- **<u>Edge Triggering:</u>**
  - +ve edge(Low to high)
  - -ve edge(high to low)



# What is Clock?




Edge Triggering

- If T is the time period
- Then f=1/time period

  f= 1/T

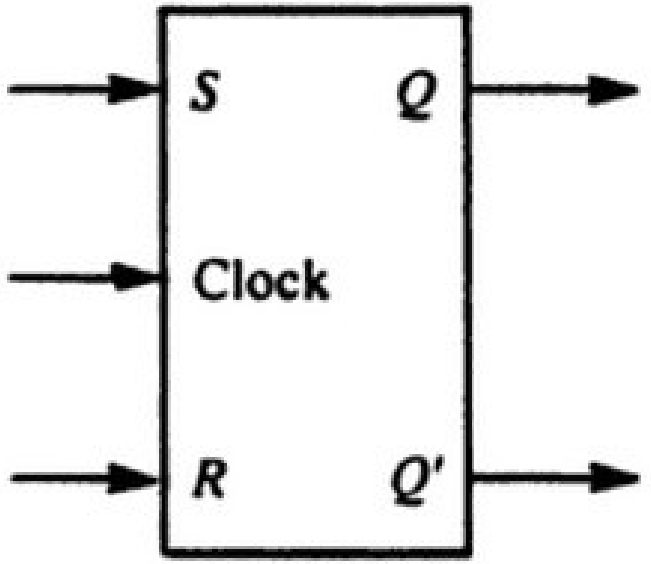
## Duty cycle:

Ratio of time for which the signal is high to the total time.

Duty cycle= signal is high/ total time

$$= \frac{\frac{t}{2}}{t}$$

= ½(duty cycle for clock is 50%)

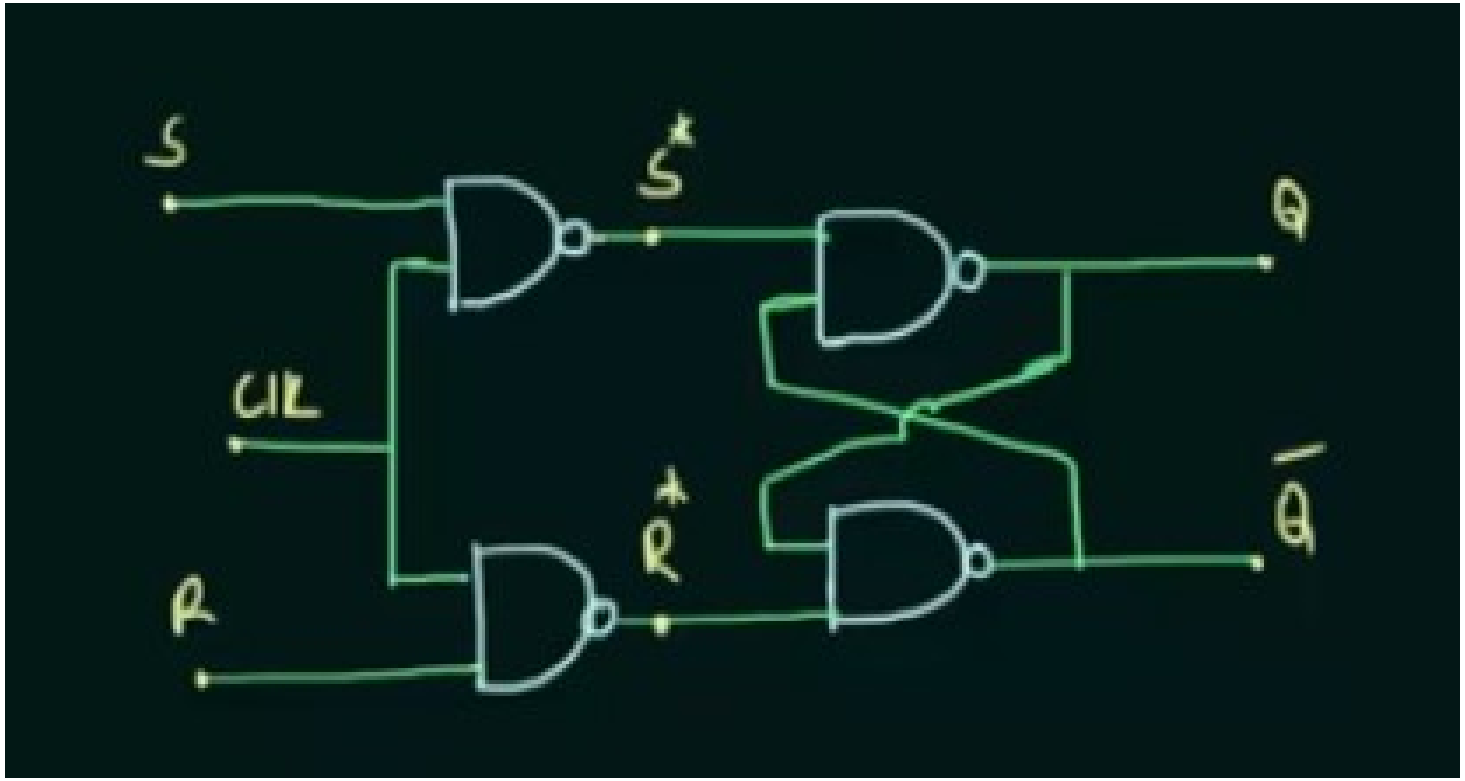



**SR Flip-Flop Graphical Symbol**

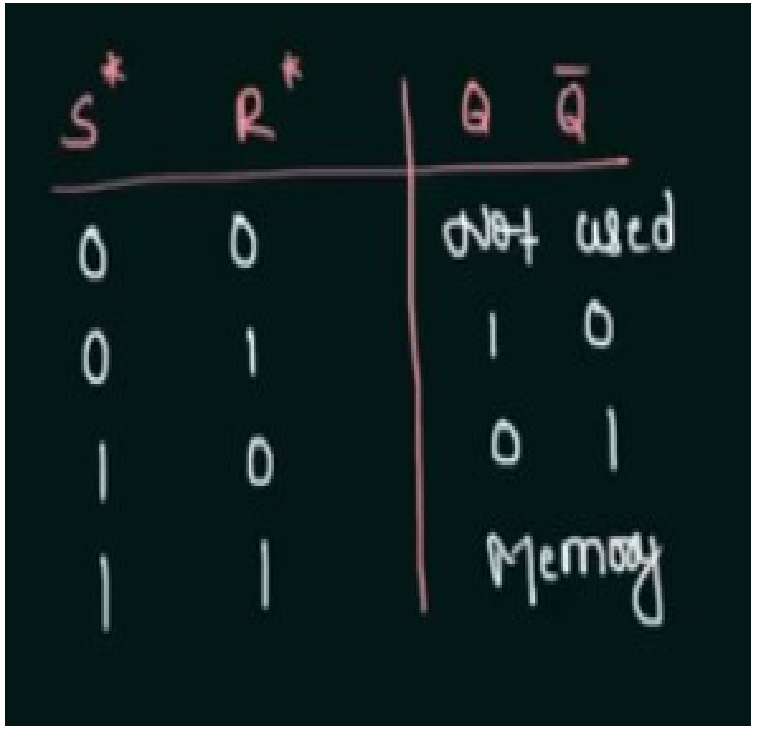
# Types of Flip-Flop

- Set-Reset (SR) flip-flop or Latch.
- JK flip-flop.
- D (Data or Delay) flip-flop.
- T (Toggle) flip-flop.

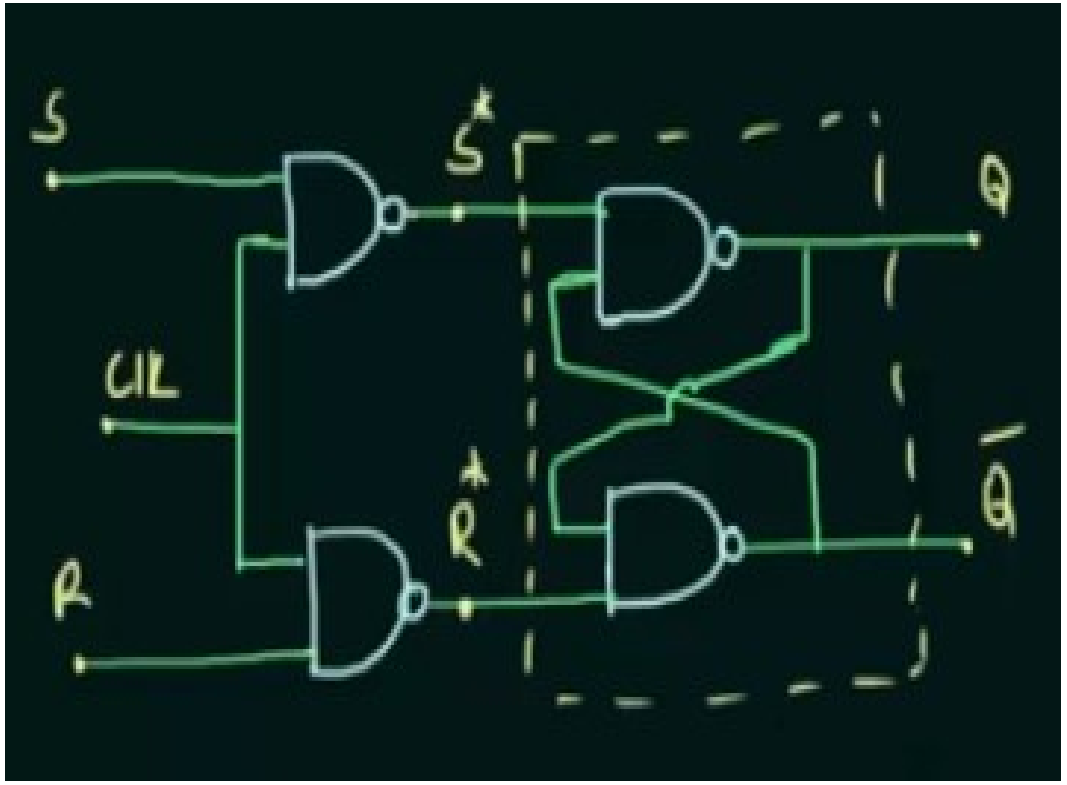
# SR Flip-Flop



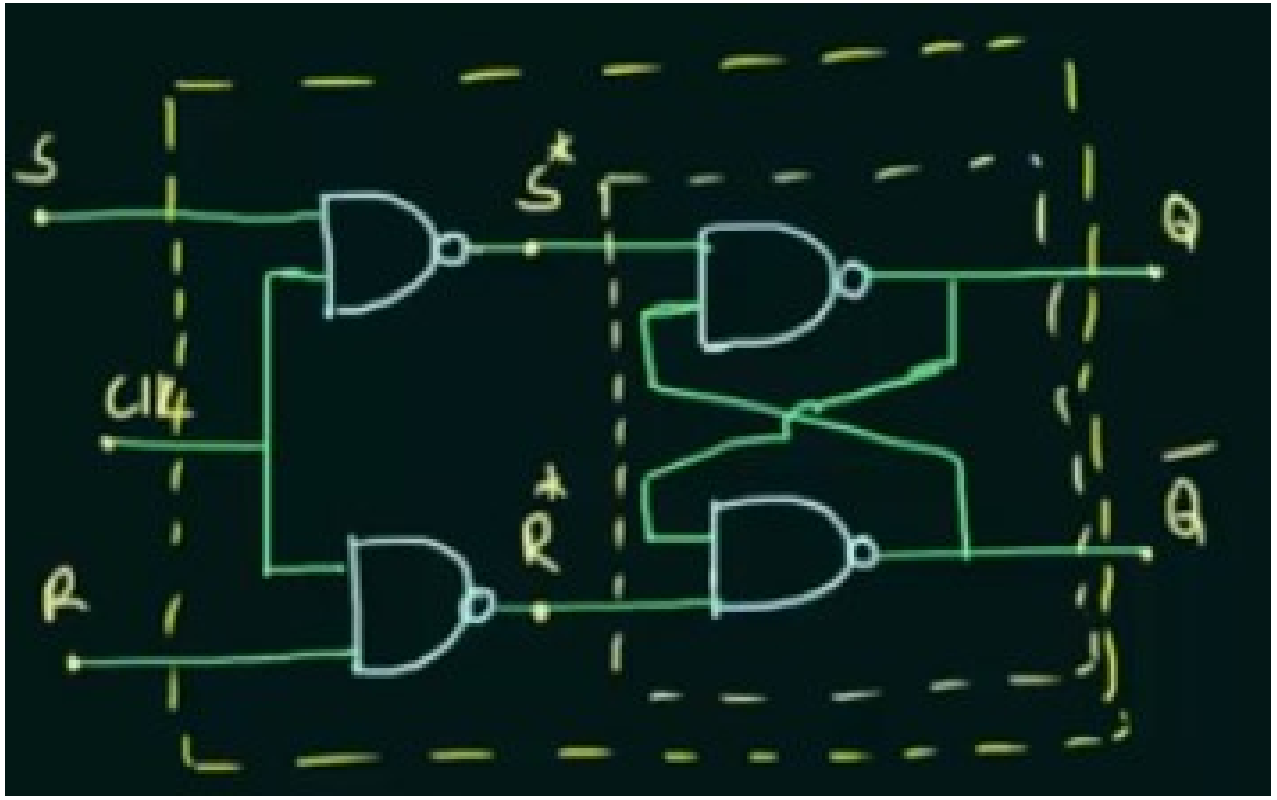

**SR Flip-Flop**




**SR Latch T.T with NAND gate**




Highlighted box is SR Latch with NAND gate




Outer Highlighted box is SR Flip-Flop

- **R*= (R.clk)'**
- **R*= R'+Clk'**

**S* = (S.Clk)'**
**S* =S'+ Clk'**

- **<u>When clk=0</u>**

S*=S'+clk'

S*= 1

R*=R'+clk'

R*=1

**<u>When clk=1</u> S*=S', R*=R'**

Case i)S=R=0:

S*=1, R*=1

Case ii) S=0, R=1

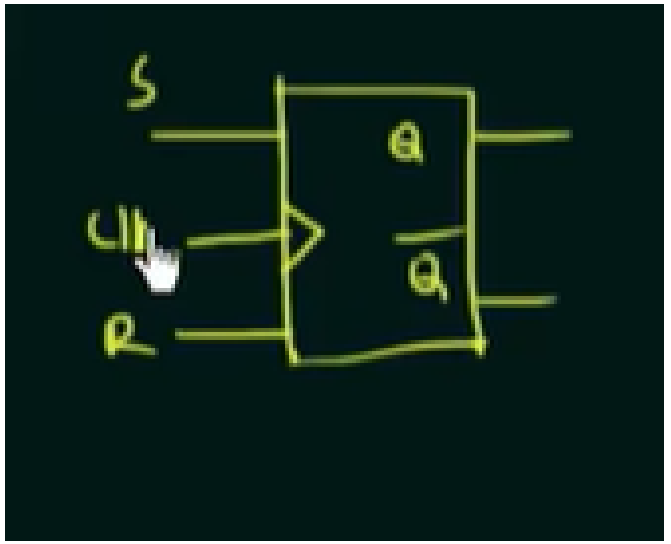
S*=1, R*=0

Case iii) S=1,R=0

S*=0, R=1

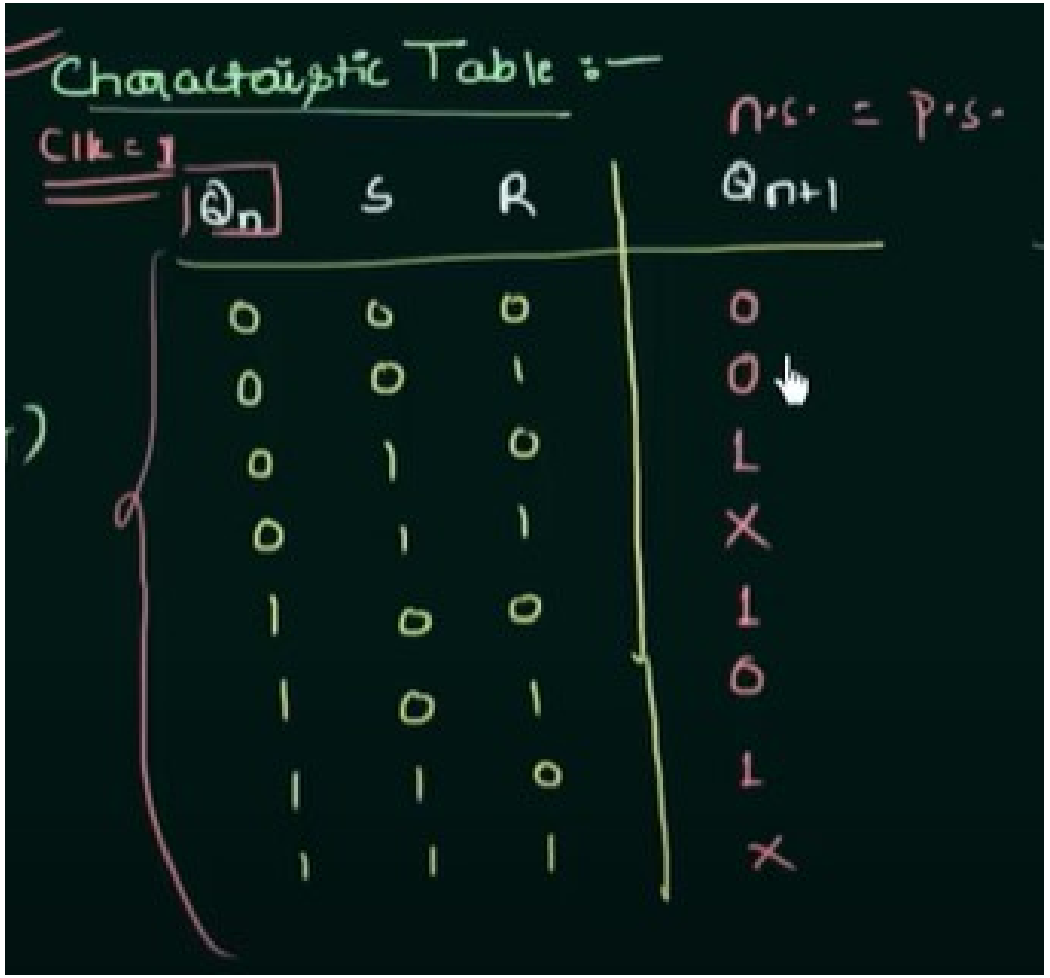
Case iv) S=R=1

S*=0, R*=0



| Clk | S | R | Q | Q’ |
|---|---|---|---|---|
| 0 | X | X | Memory | |
| 1 | 0 | 0 | Memory | |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | Not used | |

Truth Table for SR flip-flop



| Clk | S | R | Qn+1 |
|---|---|---|---|
| 0 | X | X | Qn |
| 1 | 0 | 0 | Qn |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | invlaid |

Truth Table

| Qn | S | R | Qn+1 |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | X/invalid |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | X/Invalid |

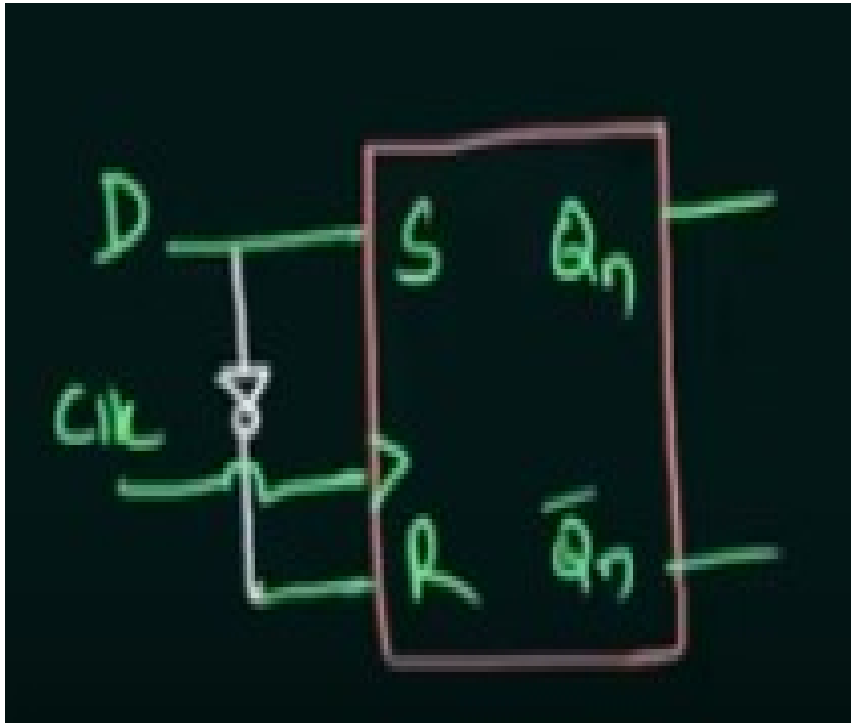
Characteristic Table

| Qn | Qn+1 | S | R |
|---|---|---|---|
| 0 | 0 | 0 | X |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | X | 0 |

Excitation Table

An **excitation table** shows the minimum inputs that are necessary to generate a particular next state (in other words, to **"excite"** it to the next state) when the current state is known. They are similar to truth tables and characteristic tables, but rearrange the data so that the current state and next state are next to each other on the left-hand side of the table, and the inputs needed to make that state change happen are shown on the right side of the table.

# D-(Data) Flip-Fop




| Clk | D | Qn+1 |
|---|---|---|
| 0 | X | Qn |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Truth Table

| Qn | D | Qn+1 |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Characteristic Table

| Qn | Qn+1 | D |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Excitation Table

# Disadvantages of SR and D

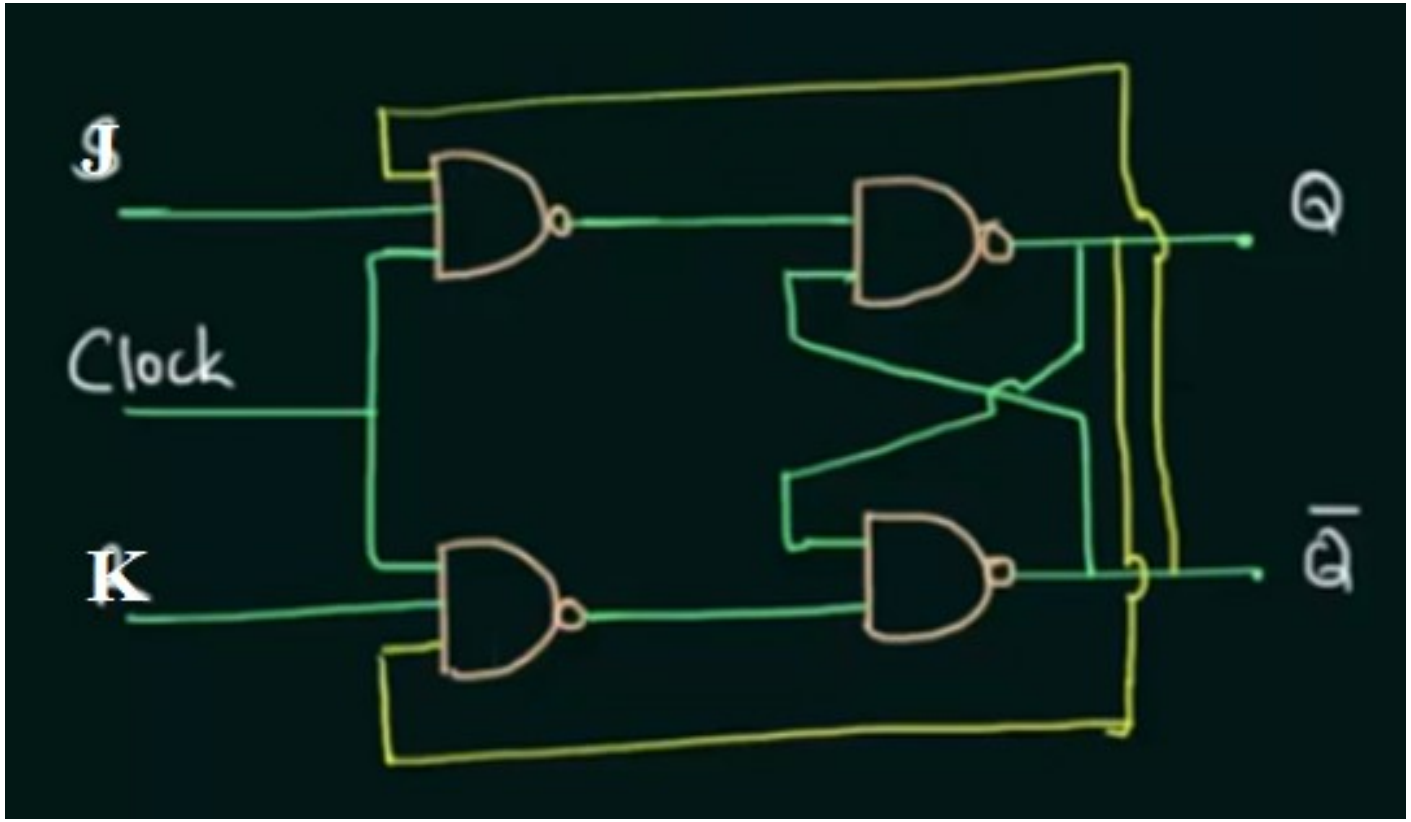
- When the S and R inputs of an SR flipflop are at logical 1, then the output becomes unstable and it is known as a race condition. So, the main disadvantage of the SR flip flop is **invalid output when both inputs are high**.

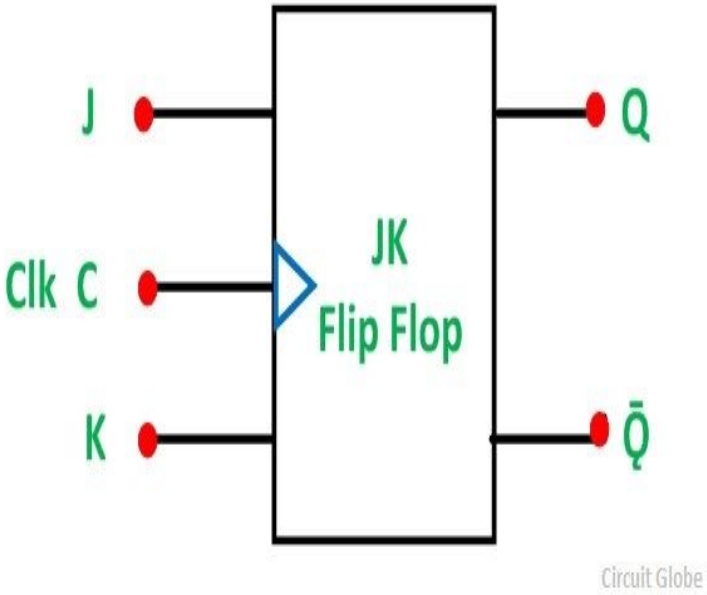
- Disadvantages of D flip flop :A delay flip flop in a circuit increases the circuit's size, often to about twice the normal. Additionally, they also make the circuits more complex.

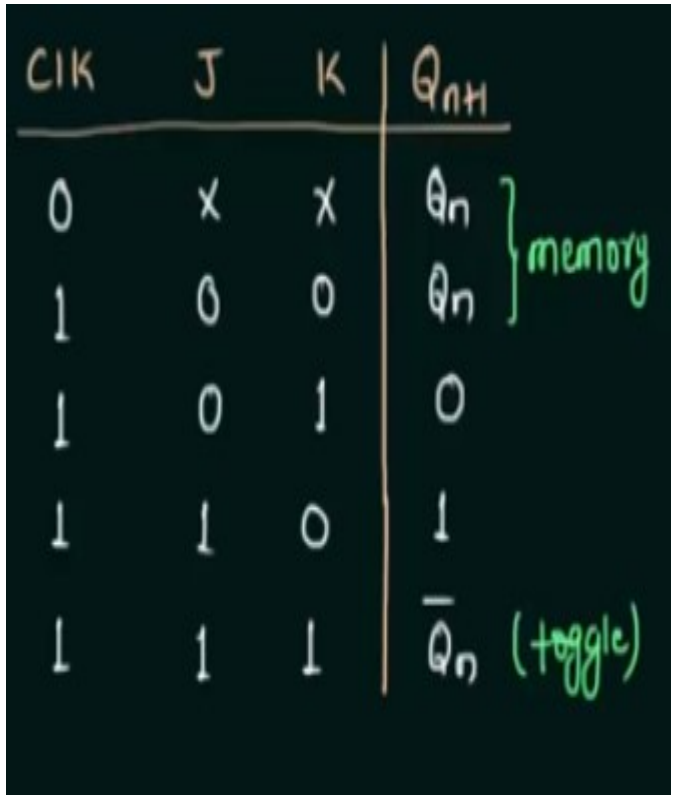
# JK Flip Flop

- JK Flip Flop is a universal flip-flop that makes the circuit toggle between two states and is widely used in shift registers, counters, PWM(pulse width modulation) and computer applications.
- It is a flip-flop, that can be either active-high or active-low based on the signal applied.
- It is an improved version of the SR Flip Flop and prevents the circuit from going in an *invalid* state.
- As the name suggests, it helps the circuit toggle between two states.
- The JK Flip-flop is also widely known as a  programmable flip-flop as it can disguise other flip-flops based on the inputs applied.

# JK Flip-Flop



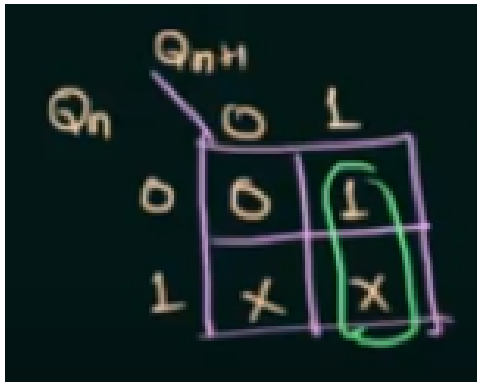





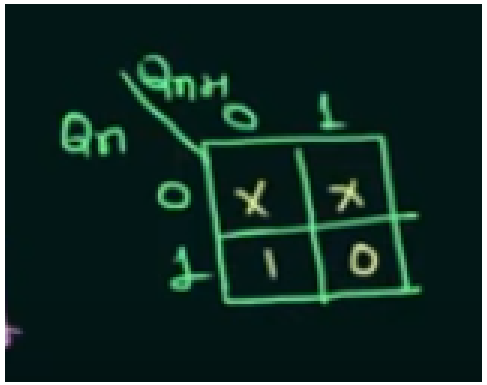

| Qn | J | K | Qn+1 |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 |

| Qn | Qn+1 | J | K |
|---|---|---|---|
| 0 | 0 | 0 | X |
| 0 | 1 | 1 | X |
| 1 | 0 | X | 1 |
| 1 | 1 | X | 0 |

Excitation table

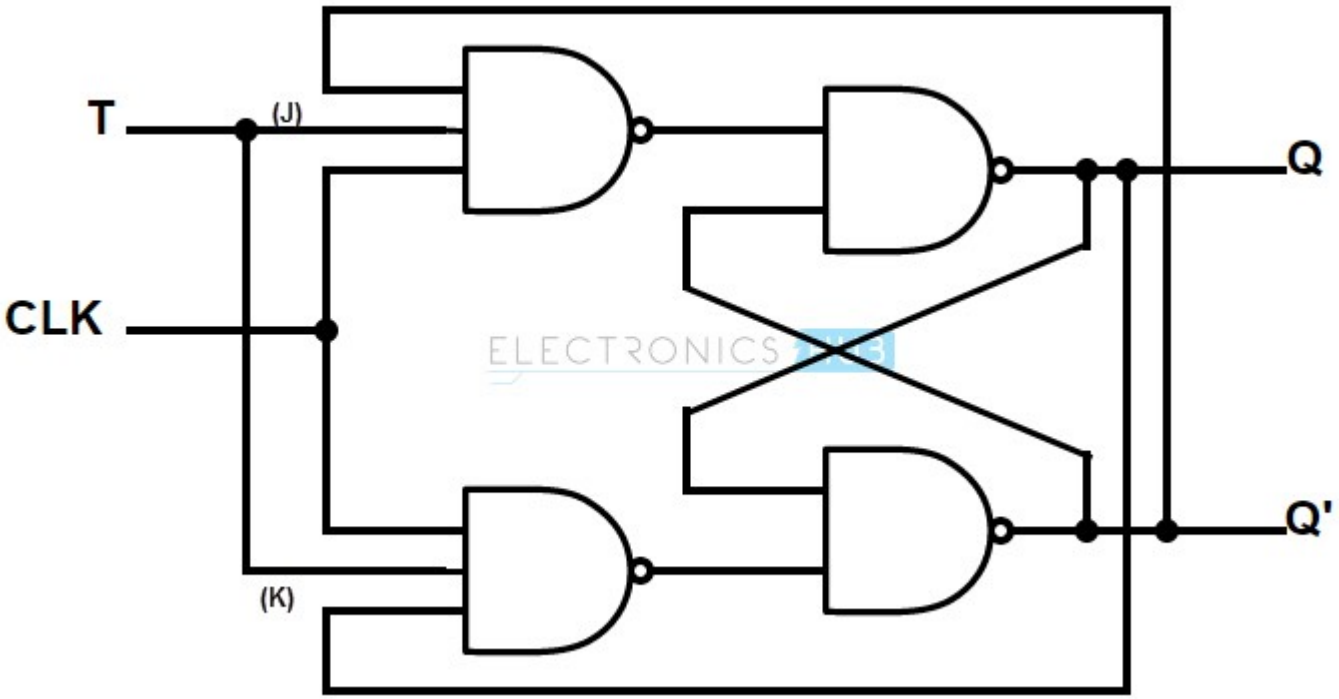


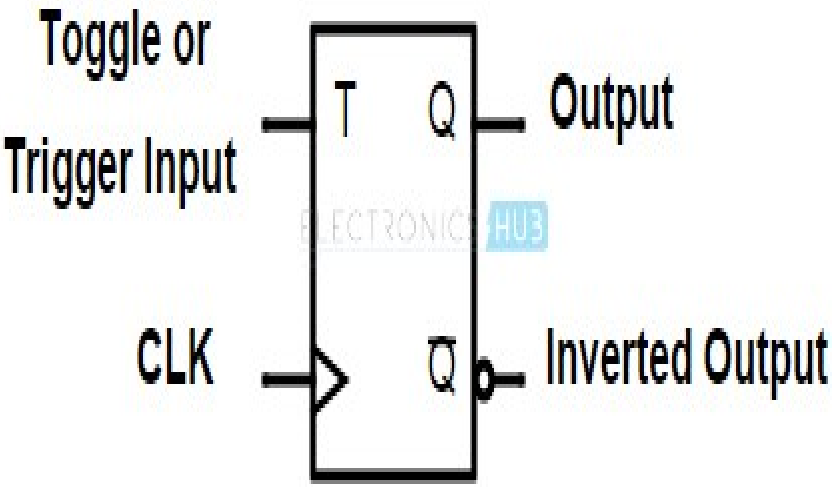

J= Qn+1

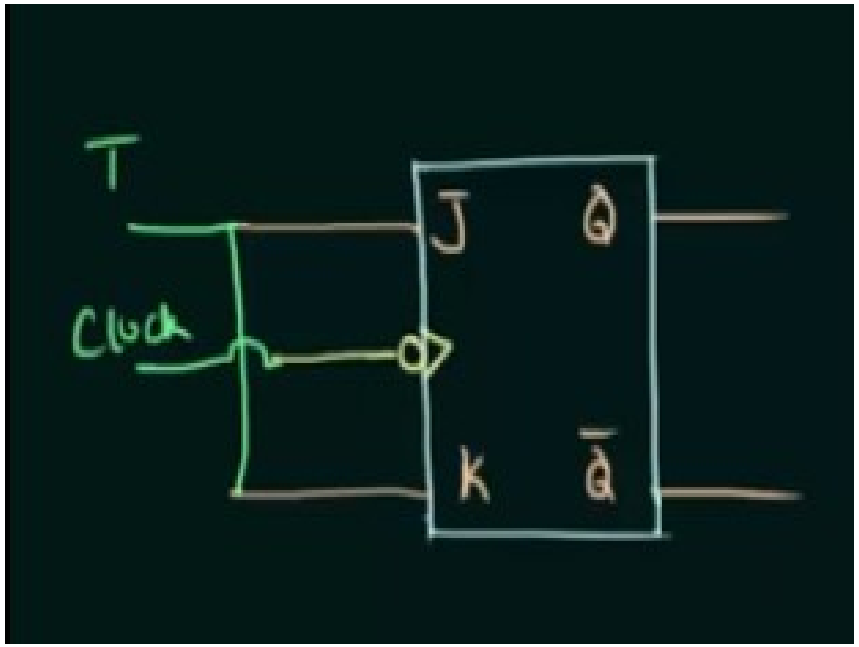



K= (Qn+1)’

# Toggle Flip-Flop






| Clk | T | Qn+1 |
|---|---|---|
| 0 | X | Qn/memory |
| 1 | 0 | Qn |
| 1 | 1 | Qn’ (toggle) |

| Qn | T | Qn+1 |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Characteristic Table

| Qn | Qn+1 | T |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Excitation table