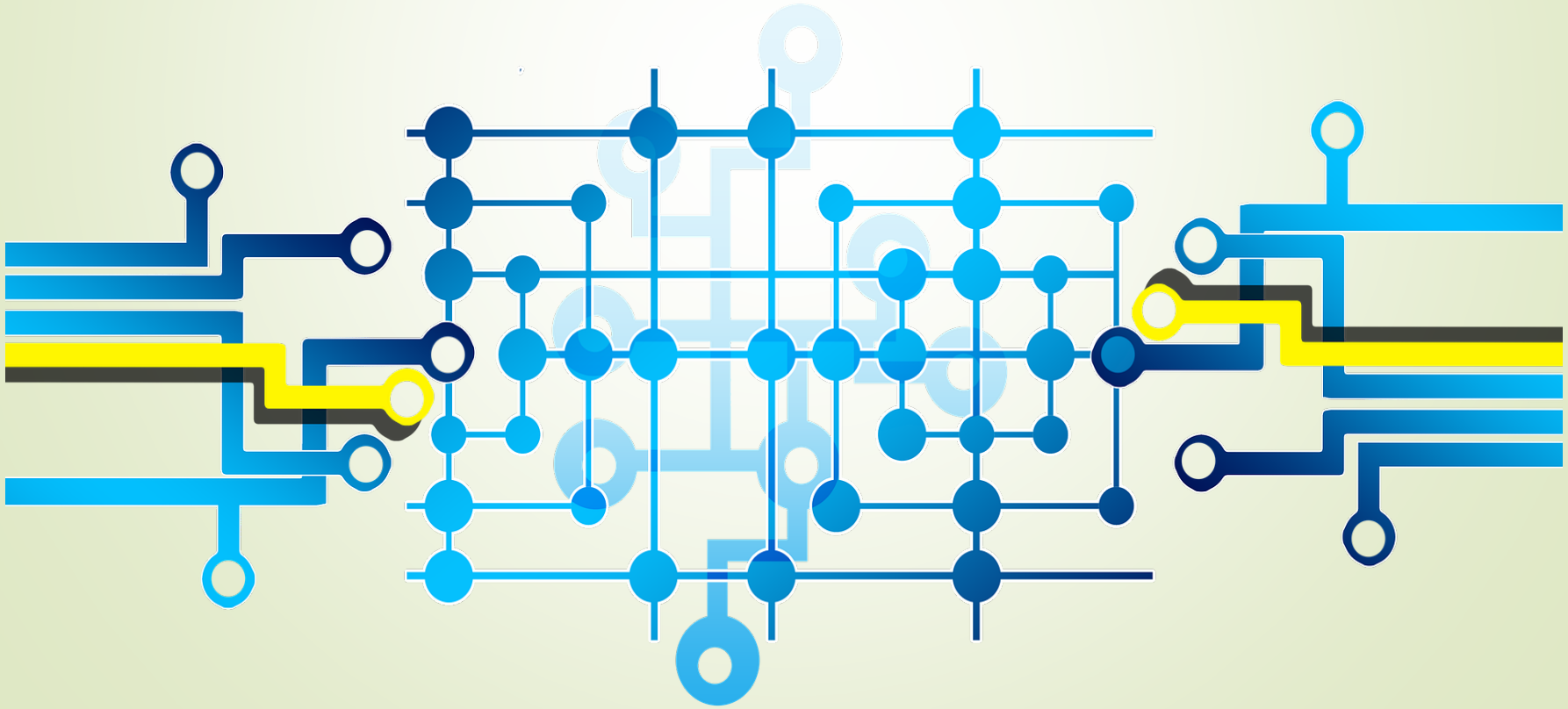
Digital Electronics (203105201)

Saurabh Srivastava, Assistant Professor
Mechatronics Engineering



CHAPTER-4

Sequential Circuits

A 1-bit memory, the circuit properties of Bi-stable latch, the clocked SR flip flop, J- K- T and D types flip flops, applications of flip flops, shift registers, applications of shift registers, ring counter, sequence generator, ripple (Asynchronous) counters, synchronous counters, special counter IC’s, asynchronous sequential counters, applications of counters.

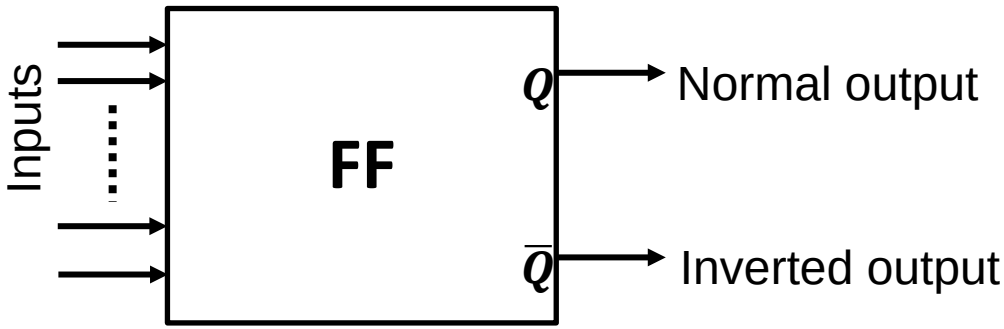
Sequential Circuits

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.**

1. Sequential circuits are **commonly used in digital systems** to implement **state machines, timers, counters, and memory elements.**
2. The memory elements in sequential circuits can be implemented using **flip-flops**, which are circuits that store binary values and maintain their state even when the inputs change.

Sequential Circuits

Flip-flop: Another name for bistable multivibrator. Used for storing a single bit (memory element)

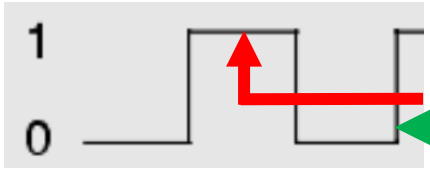


State of FF: State of Q

State of FF:
High/1/SET , for Q=1
Low/0/RESET, for Q=0

- The inputs cause change in output state (Flip-Flop)
- FF input is applied temporarily to change its output state.
- Input applied in form of pulses.
- The output now remains in new state, even when input is removed (**Memory**)
- Used to store 1-bit of data.
- Basic component of shift-registers and counters.

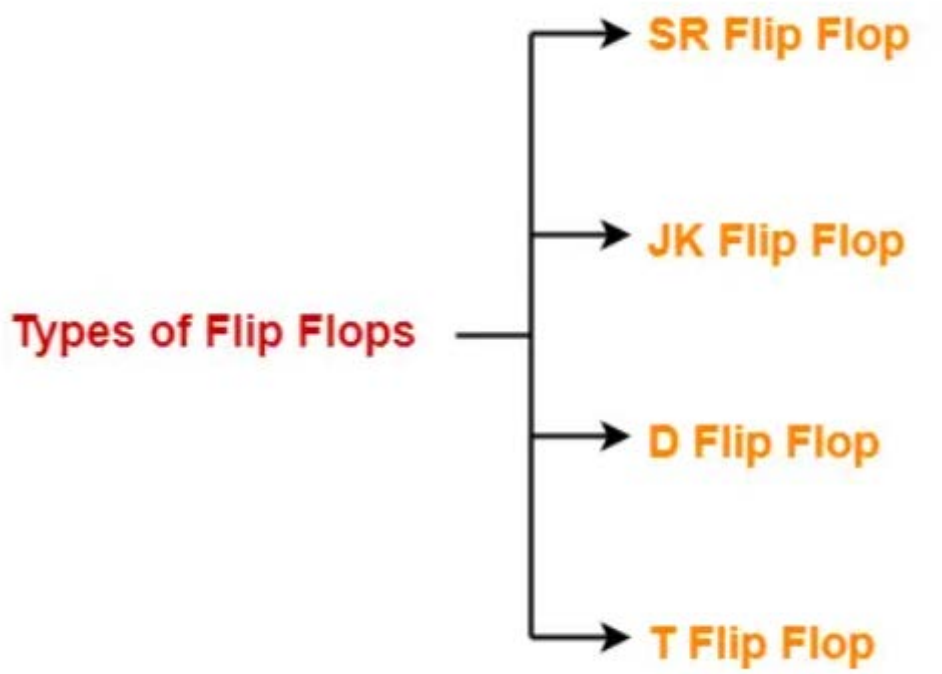
***Non-clocked FF is called Latch**



Latch: Level-triggered
FF: Edge-triggered

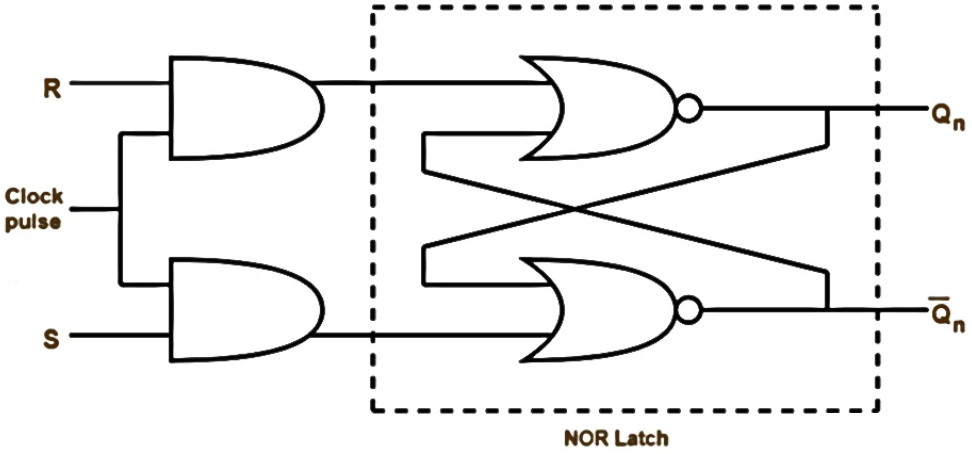
Sequential Circuits

- A Flip Flop is a memory element that is capable of storing one bit of information.
- It is also called as a **Bistable Multivibrator** since it has two stable states either 0 or 1.

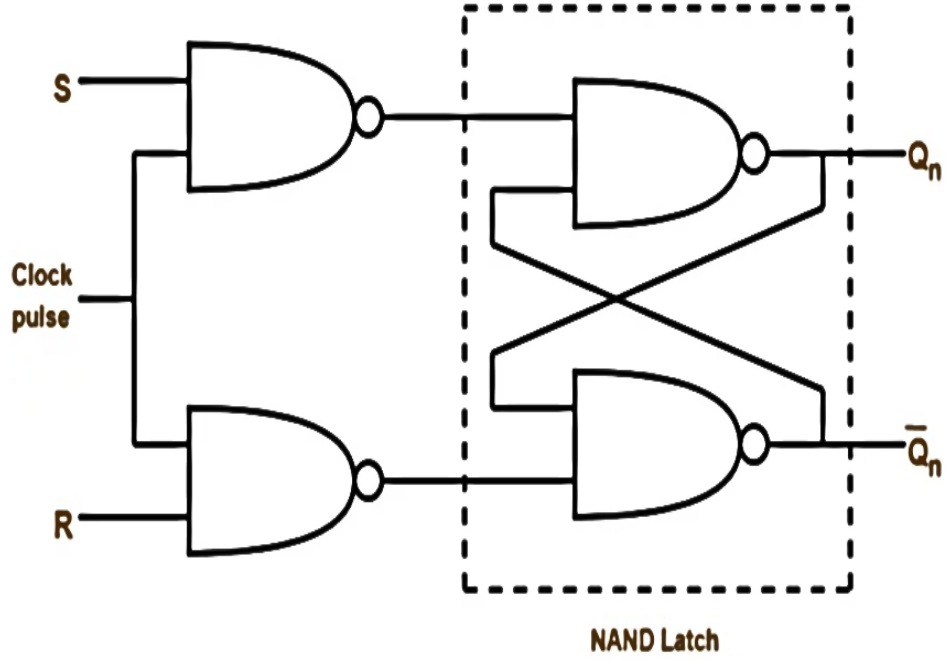


Sequential Circuits

Using NOR gates



Using NAND gates

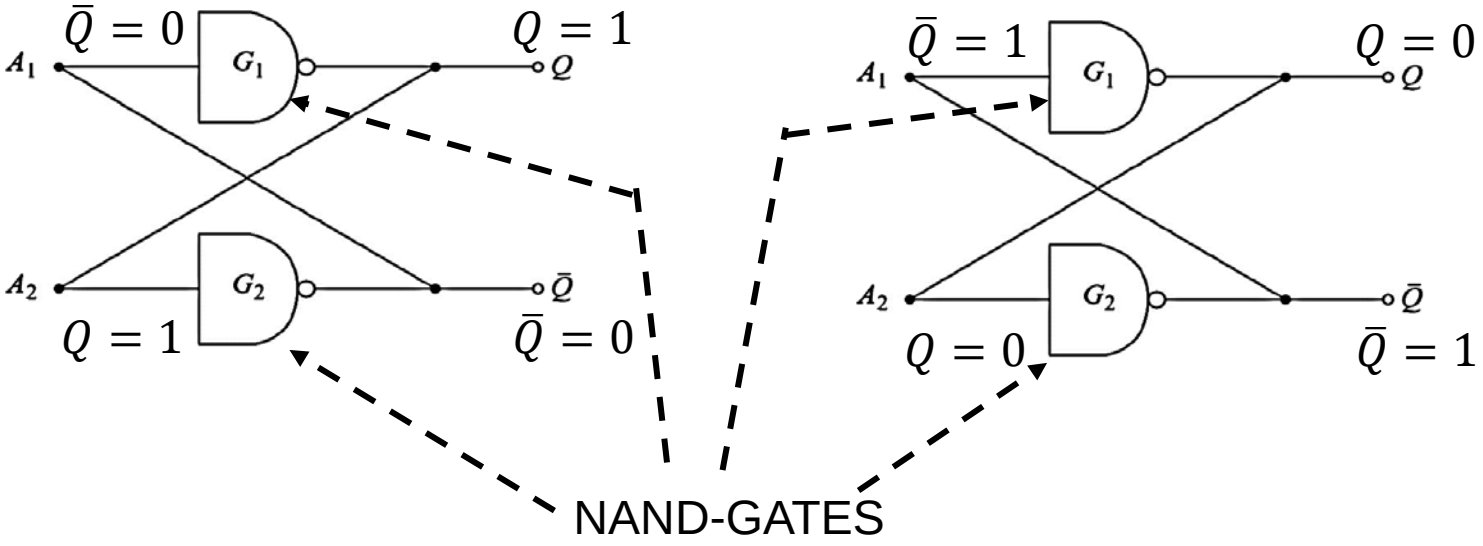


Sequential Circuits

Bistable Latch/RS Flip-flop: Circuit properties

Case I: $Q = 1, \bar{Q} = 0$

Case II: $Q = 0, \bar{Q} = 1$

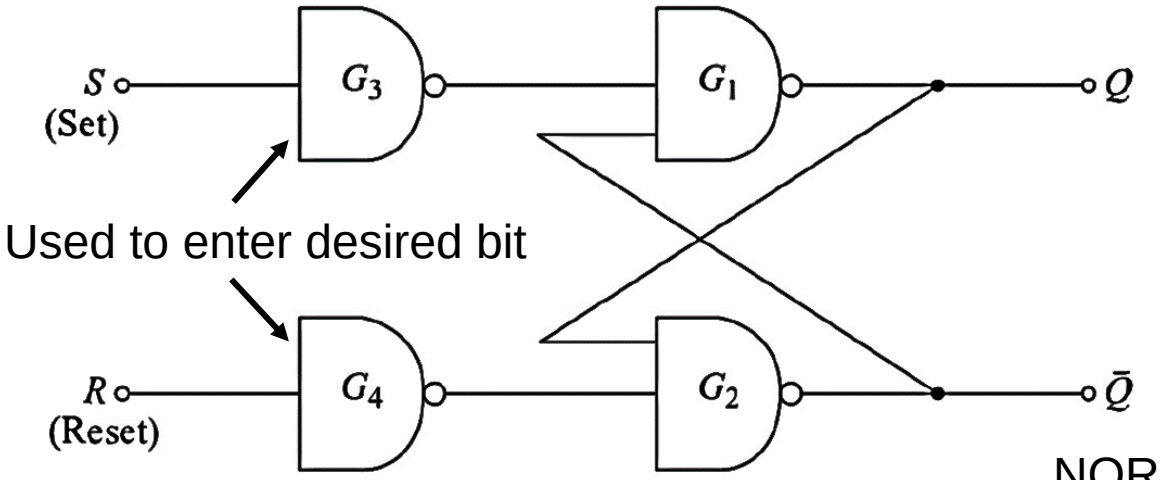


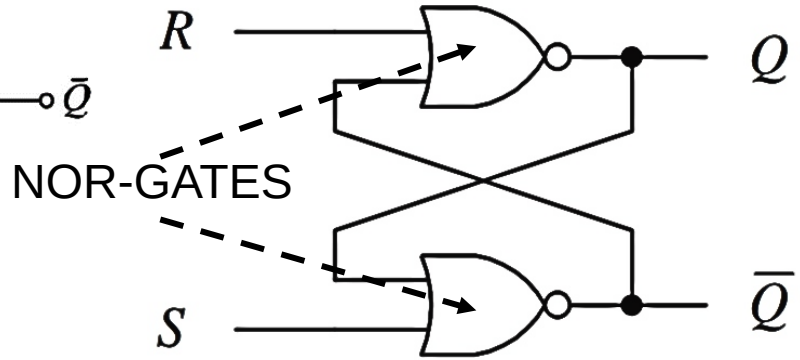
Sequential Circuits

Bistable Latch/RS Flip-flop: Circuit Properties

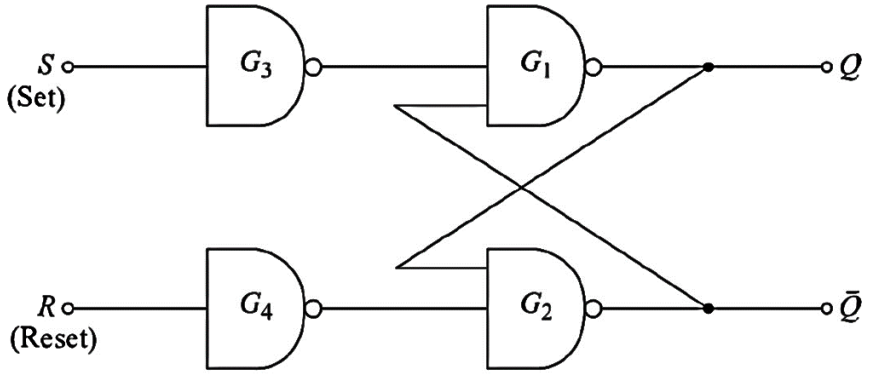
- The outputs Q and $\bar{Q}$ are always complementary.
- The circuits has two stable states. One state is $Q = 1$ (Set state), Another is $Q = 0$ (Reset state).
- The circuits continues to be in the same state, zero or one.

How to store the desired bit ?





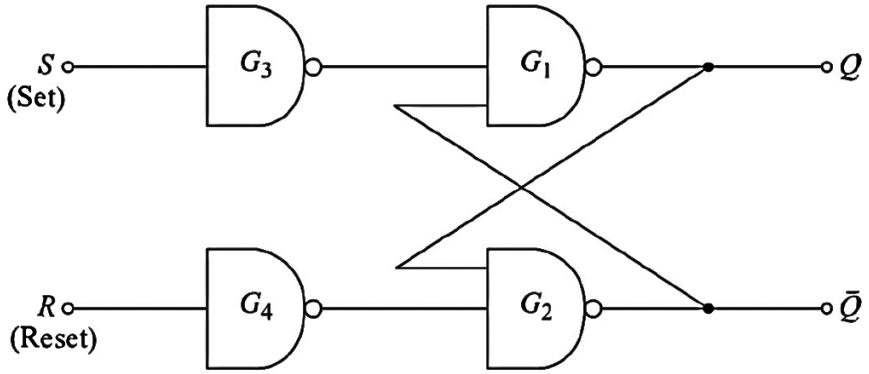
Sequential Circuits



A	B	$Y = \overline{A.B}$
0	0	1
0	1	1
1	0	1
1	1	0

Case I: $Q = 1$, $\bar{Q} = 0$	S	Input to G_1	Output of G_1 (Q)	R	Input to G_2	Output of G_2 $(\bar{Q})$
	0	1,0	1	0	1,1	0
Case II: $Q = 0$, $\bar{Q} = 1$	S	Input to G_1	Output of G_1 (Q)	R	Input to G_2	Output of G_2 $(\bar{Q})$
	0	1,1	0	0	1,0	1

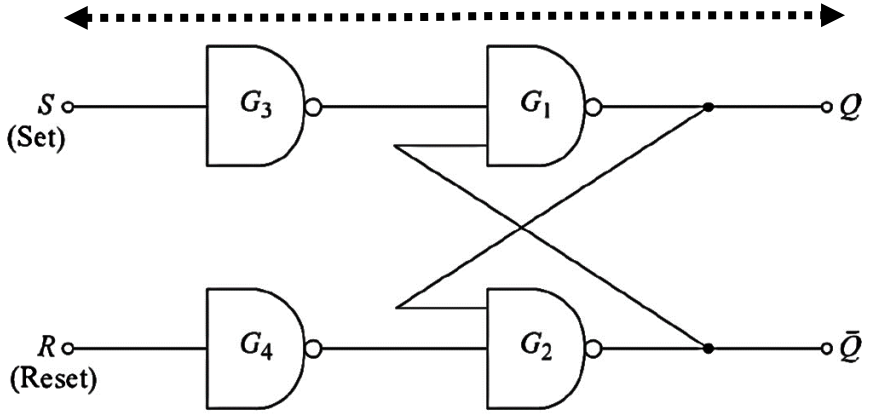
Sequential Circuits

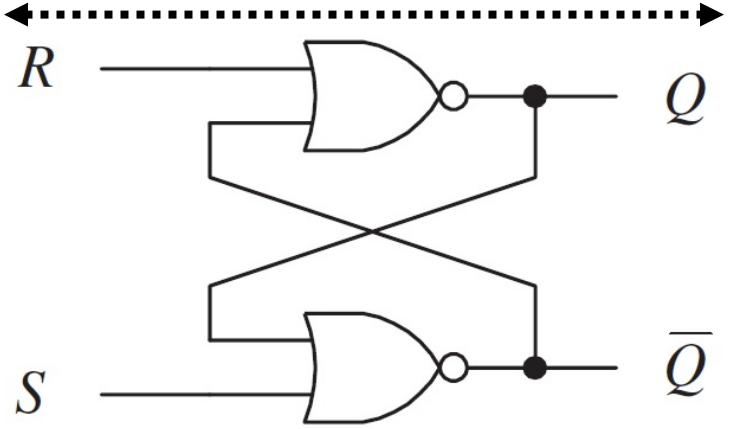


A	B	$Y = \overline{A.B}$
0	0	1
0	1	1
1	0	1
1	1	0

Case I: $Q = 1$, $\bar{Q} = 0$	S	Input to G_1	Output of G_1 (Q)	R	Input to G_2	Output of G_2 $(\bar{Q})$
	1	0,0	1	0	1,1	0
Case II: $Q = 0$, $\bar{Q} = 1$	S	Input to G_1	Output of G_1 (Q)	R	Input to G_2	Output of G_2 $(\bar{Q})$
	1	0,1	1	0	1,1	0

Sequential Circuits





Q_n \ SR	00	01	11	10
0			X	1
1	1		X	1

Characteristic eq. of SR-FF:

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

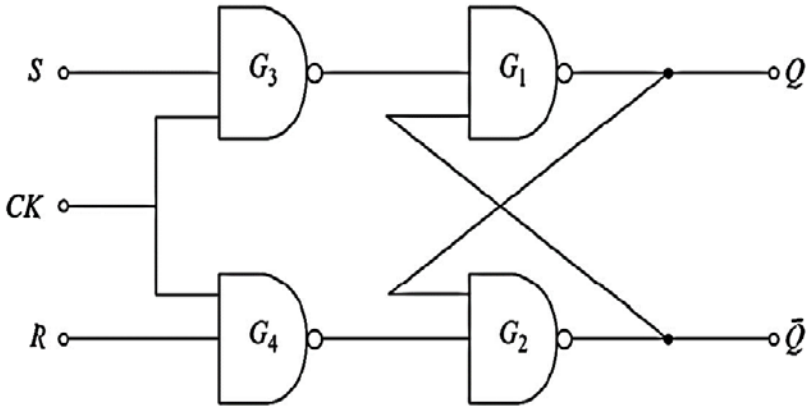
S	R	Q_{n+1}	$\overline{Q_{n+1}}$	Remarks
0	0	Q_n	$\overline{Q_n}$	No change
0	1	0	1	reset condition
1	0	1	0	set condition
1	1	Not valid	Not valid	**Invalid combination**

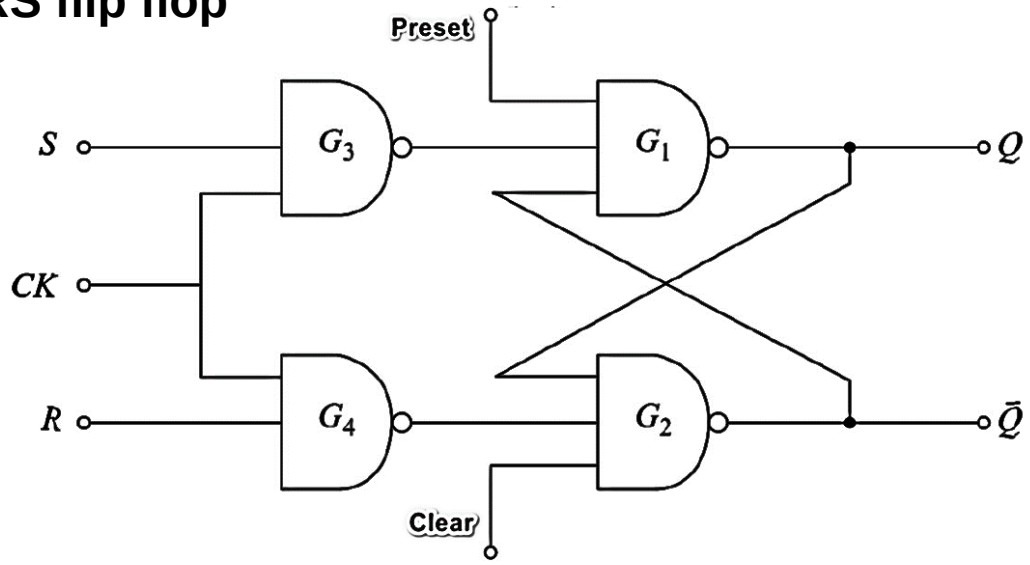
S — NAND — Q

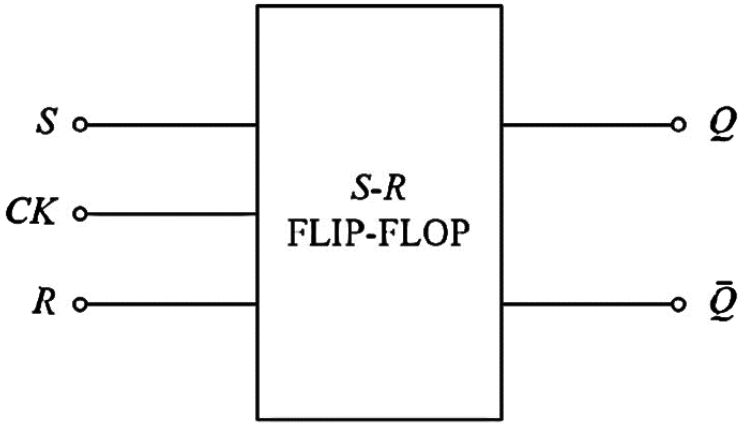
R — NOR — Q

Sequential Circuits

Clocked RS flip flop







Inputs			Output Q	Operation
Clock (CK)	Clear (Cr)	Preset (Pr)		
1	1	1	Q_{n+1}	Normal FF
0	0	1	0	Clear
0	1	0	1	Preset

Sequential Circuits

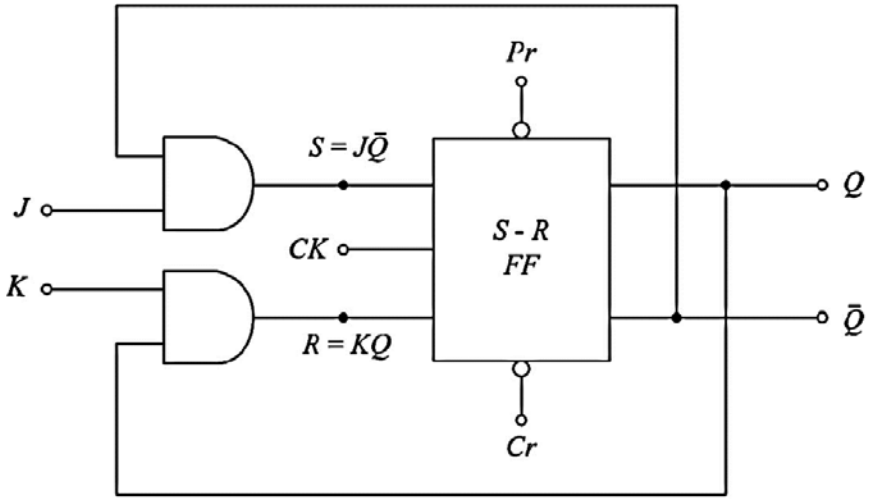
JK- flip flop

S	R	Q_{n+1}	$\overline{Q_{n+1}}$	Remarks
.....				
1	1	Not valid	Not valid	**Invalid combination**

$$S = J\bar{Q}$$
$$R = KQ$$

A	B	$Y = AB$
0	0	0
0	1	0
1	0	0
1	1	1

Data inputs		Outputs		Inputs to SR FF		Output Q_{n+1}
J_n	K_n	Q_n	$\overline{Q_n}$	S_n	R_n	
0	0	0	1	0	0	0 (=Q_n)
0	0	1	0	0	0	1 (=Q_n)
1	0	0	1	1	0	1
1	0	1	0	0	0	1
0	1	0	1	0	0	0
0	1	1	0	0	1	0
1	1	0	1	1	0	1 (=$\overline{Q_n}$)
1	1	1	0	0	1	0 (=$\overline{Q_n}$)



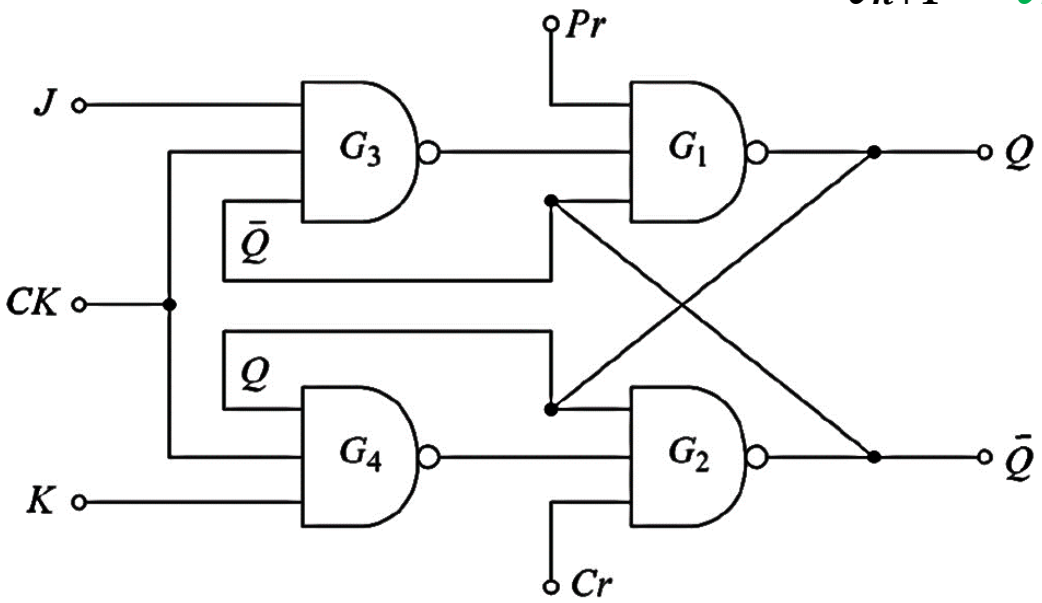
Sequential Circuits

Inputs		Output
J_n	K_n	Q_{n+1}
0	0	Q_n
0	1	0
1	0	1
1	1	$\overline{Q_n}$

JK- flip flop

Q_n \ JK	00	01	11	10
0	0	0	1	1
1	1	0	0	1

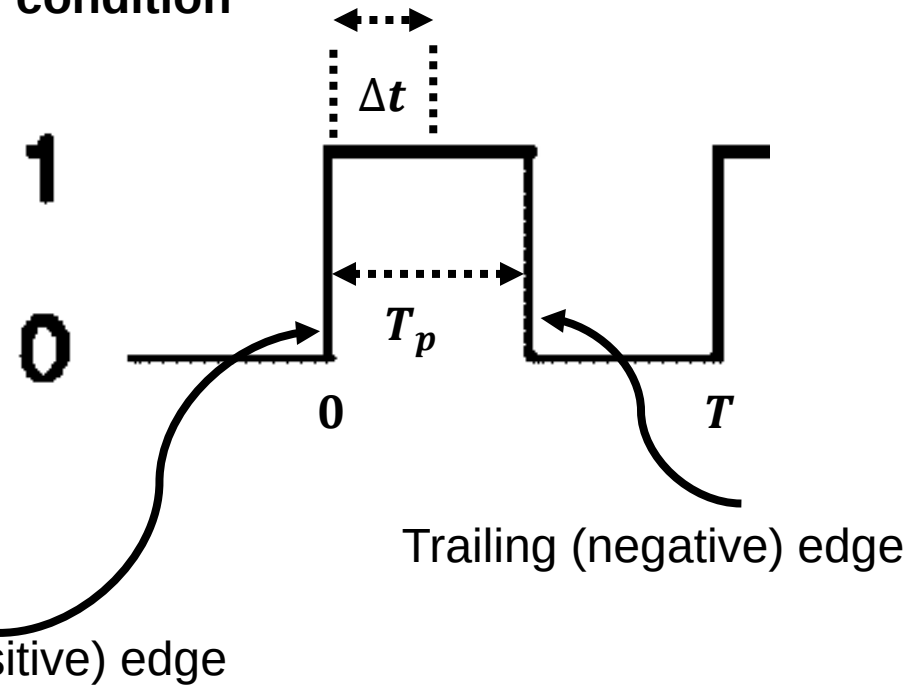
$$Q_{n+1} = \overline{Q_n}J + Q_n\overline{K}$$



Sequential Circuits

Inputs		Output
J_n	K_n	Q_{n+1}
0	0	Q_n
0	1	0
1	0	1
1	**1**	$\overline{Q_n}$

Race-around condition



Δt: The propagation delay through two NAND gates in series.

Output oscillates after every Δt when the clock pulse (T_p) is high, when both JK inputs are high.

- At the end of clock pulse, the output is not certain.

Avoided, if T_p is made smaller than Δt. Practical method - use Master Slave flip flop

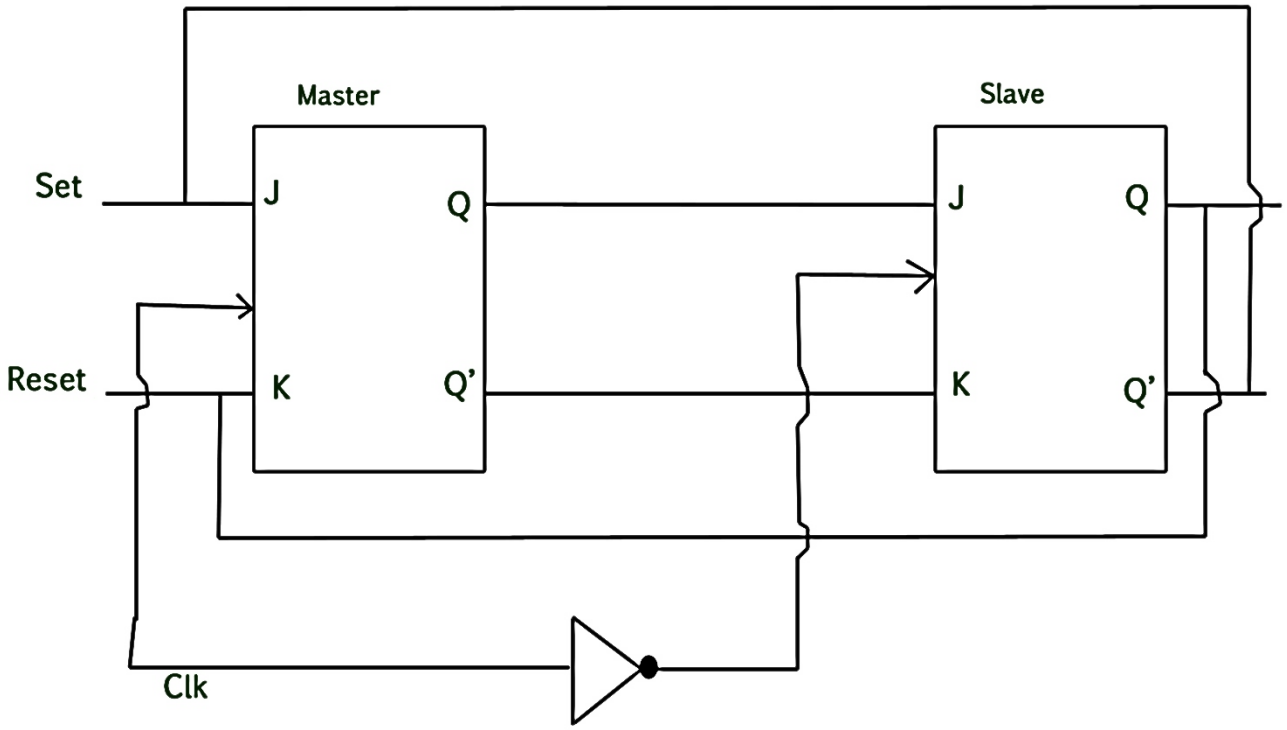
Sequential Circuits

Master-Slave Flip-flop (Pulse triggered FF)

- The Race-around Condition can be avoided by ensuring that the clock input is at logic “1” only for a very short time.
- The Master-Slave Flip-Flop is a **combination of two JK flip-flops connected in a series** configuration.
- One acts as the “master” and the other as a “slave”.
- The output from the master flip-flop is connected to the two inputs of the slave flip flop whose output is fed back to the inputs of the master flip-flop.
- In addition to these two flip-flops, the **circuit also includes an inverter.**
- The inverter is connected to the clock pulse in such a way that the inverted clock pulse is given to the slave flip-flop. In other words if CLK=0 for a master flip-flop, then CLK=1 for a slave flip-flop, and if CLK=1 for a master flip flop then it becomes 0 for a slave flip-flop.

Sequential Circuits

Master-Slave Flip-flop (Pulse triggered FF)



Master-slave JK Flip-Flop

Sequential Circuits

D (Delay) flip flop

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

Characteristic Table of D-FF

$$Characteristic\ Eq.: Q_{n+1} = D$$

Q_n \ D	0	1
0	0	1
1	0	1

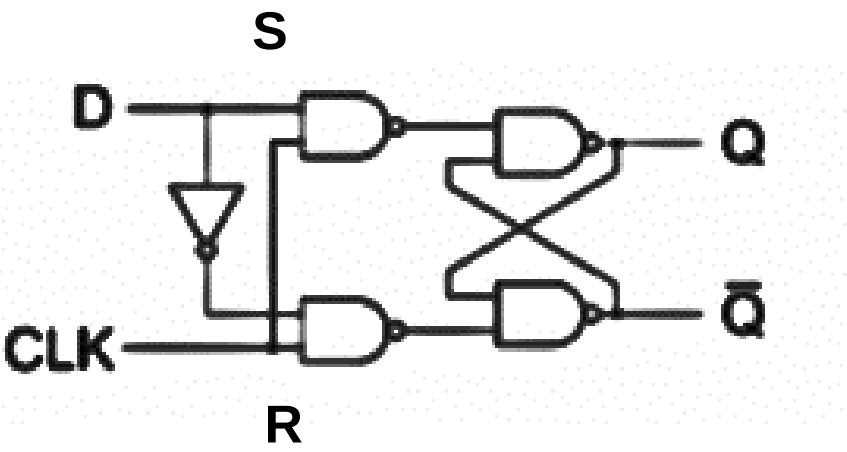
Sequential Circuits

D flip flop: From SR flip flop

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

Characteristic Table of D-FF

Excitation table of SR-FF



Q_n \ D	0	1
0	0	1
1	0	X

$S = D$

Q_n \ D	0	1
0	X	0
1	1	0

$R = \bar{D}$

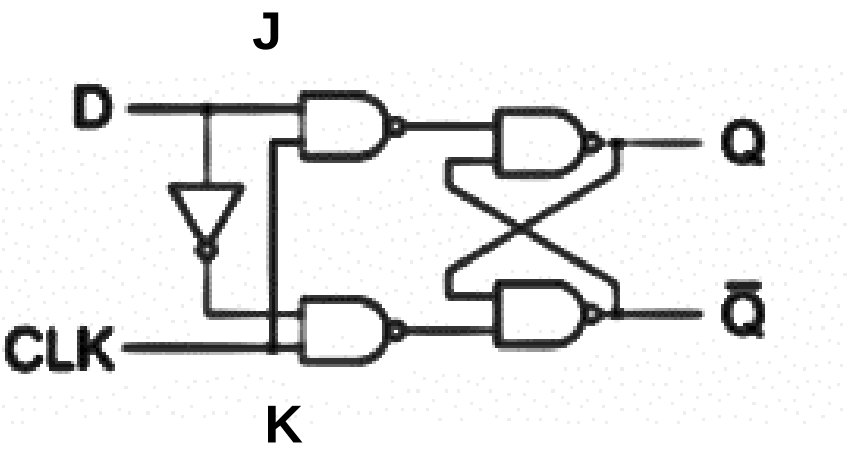
Sequential Circuits

D flip flop: From JK flip flop

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

Characteristic Table of D-FF

Excitation table of JK-FF



Q_n \ D	0	1
0	0	1
1	X	X

$J = D$

Q_n \ D	0	1
0	X	X
1	1	0

$K = \bar{D}$

Sequential Circuits

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

Characteristic Table of T-FF

T (Toggle) flip flop

$$Characteristic\ Eq.: Q_{n+1} = T\overline{Q_n} + \overline{T}Q_n = T \oplus Q_n$$

Q_n \ T	0	1
0	0	1
1	1	0

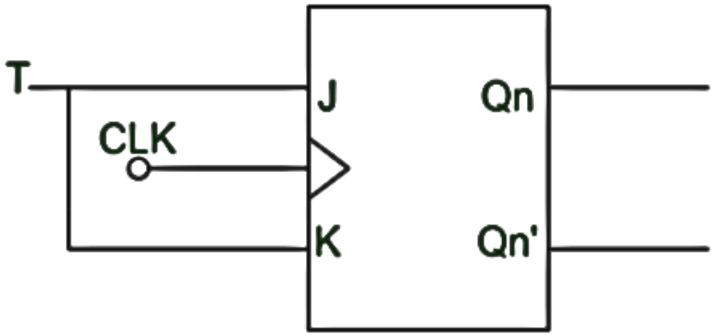
Sequential Circuits

T flip flop: From JK flip flop

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

Characteristic Table of T-FF

Excitation table of JK-FF



Q_n \ T	0	1
0	0	1
1	X	X

$J = T$

Q_n \ T	0	1
0	X	X
1	0	1

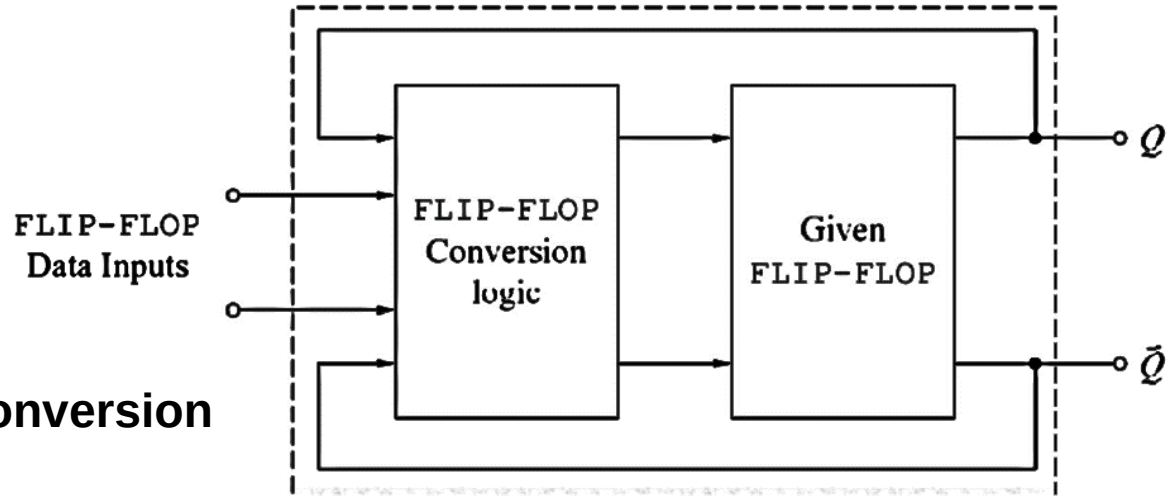
$K = T$

Sequential Circuits

Excitation Tables of FFs:

Present State	Next State	S–R FF S_n	S–R FF R_n	J–K FF J_n	J–K FF K_n	T-FF T_n	D-FF D_n
0	0	0	×	0	×	0	0
0	1	1	0	1	×	1	1
1	0	0	1	×	1	1	0
1	1	×	0	×	0	0	1

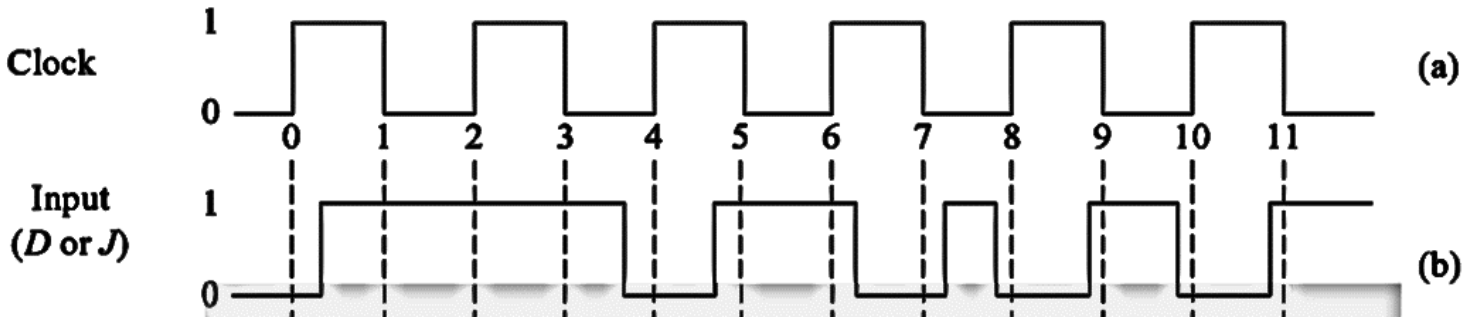
Flip-flop conversion



Sequential Circuits

Timing diagrams of FFs:

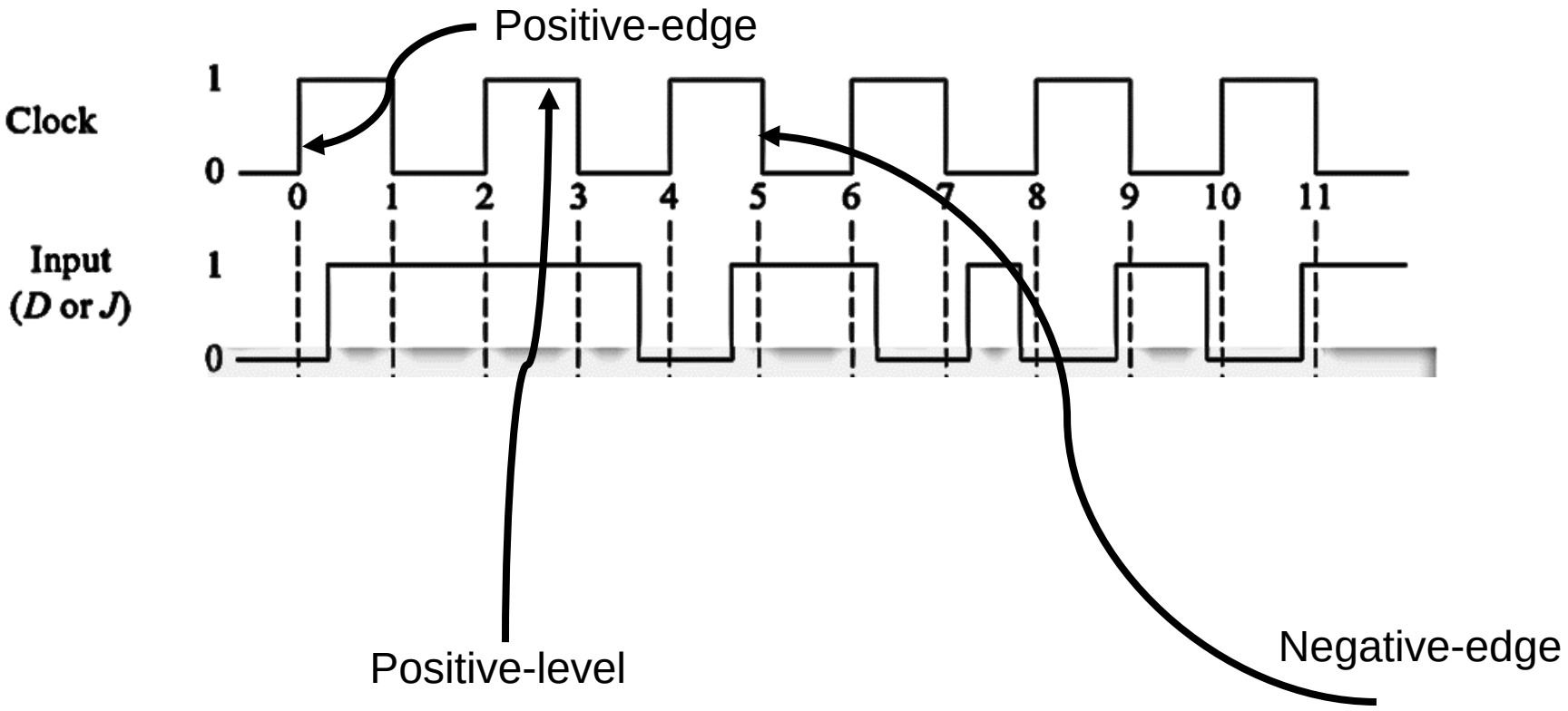
Q: Consider the following Clock and Input waveforms to the FF.



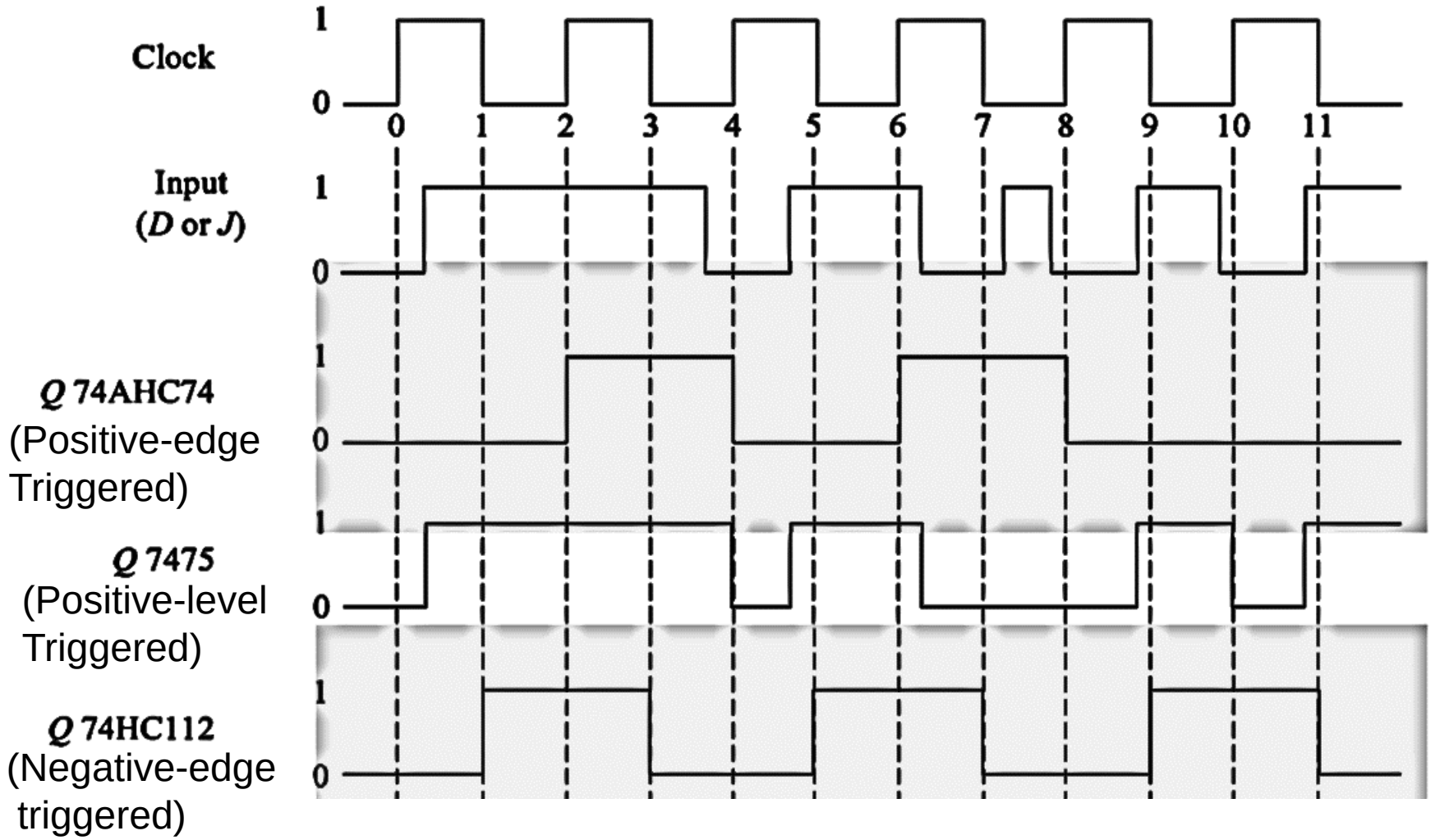
Sketch the output waveforms for the following FFs:

a) Positive-edge triggered D-type FF 74 AHC74.
b) Positive-level triggered D-type FF 7475.
c) Negative-edge triggered JK-Flip flop 74HC112.

Sequential Circuits



Sequential Circuits



Sequential Circuits

Applications of flip flop:

- Frequency dividers
- Counters
- Storage registers
- Shift registers
- Data storage
- Bounce elimination switch
- Latch
- Data transfer
- Memory
- Registers

Sequential Circuits

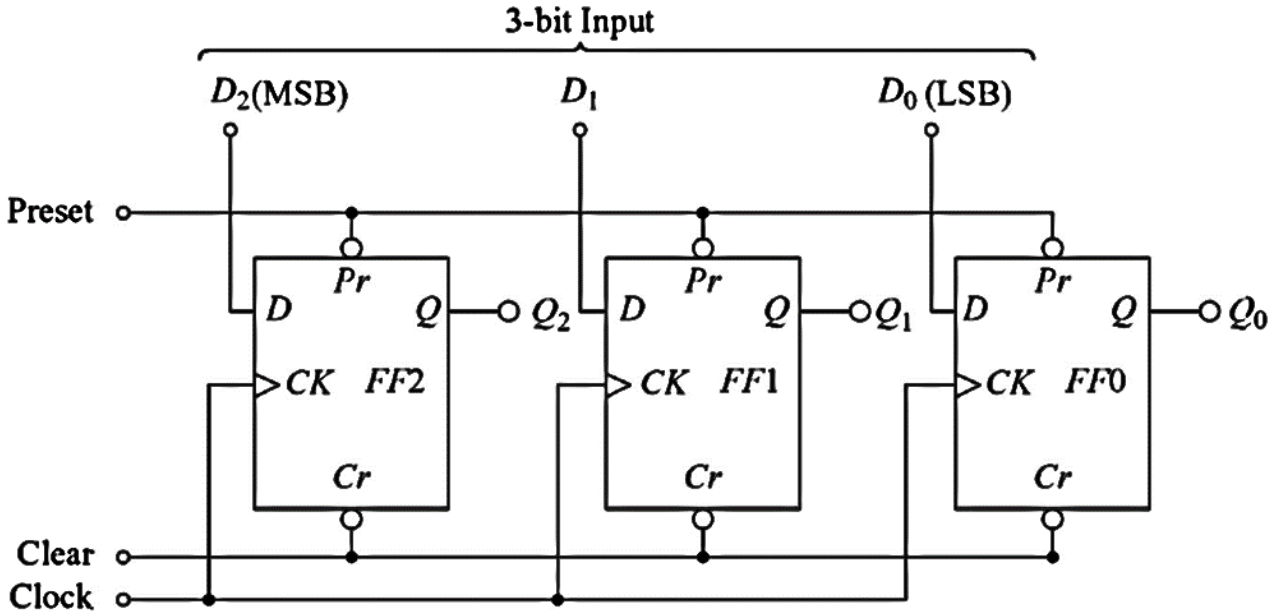
Shift Registers:

A **Register** is a device that is used to store such information.
Intel's 8085 processor contains many 8-bit registers for storage and result purposes,
It is a group of flip-flops connected in series used to store multiple bits of data. The information stored within these registers can be transferred with the help of **shift registers**.
Shift Register is a group of flip flops used to store multiple bits of data. The bits stored in such registers can be made to move within the registers and in/out of the registers by applying clock pulses.
An n-bit shift register can be formed by connecting n flip-flops, where each flip-flop stores a single bit of data.
The registers which will shift the bits to the left are called **Shift-left registers**.
The registers which will shift the bits to the right are called **Shift-right registers**

Sequential Circuits

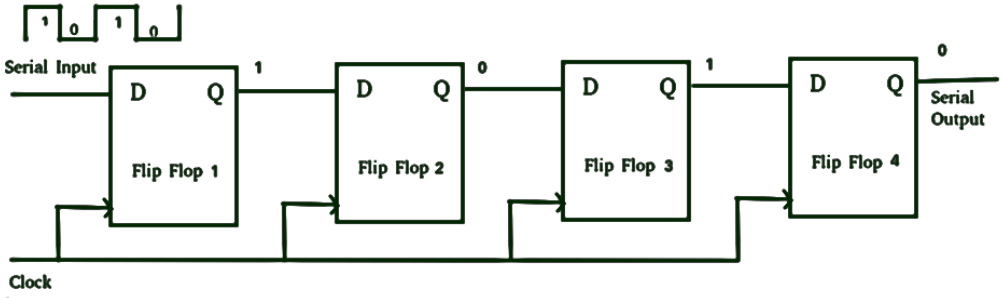
Data can be entered and retrieved in following forms:

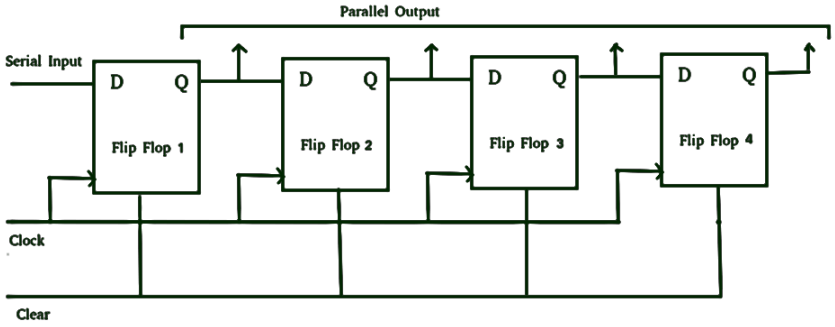
a) Serial-In Serial-out
b) Serial-in Parallel-out
c) Parallel-in serial-out
d) Parallel-in parallel-out

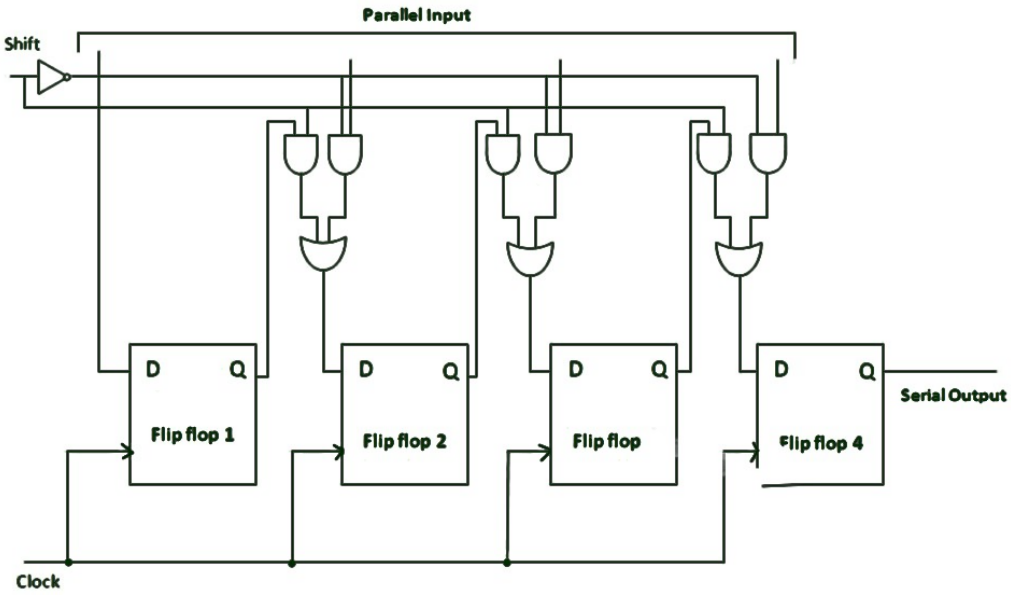


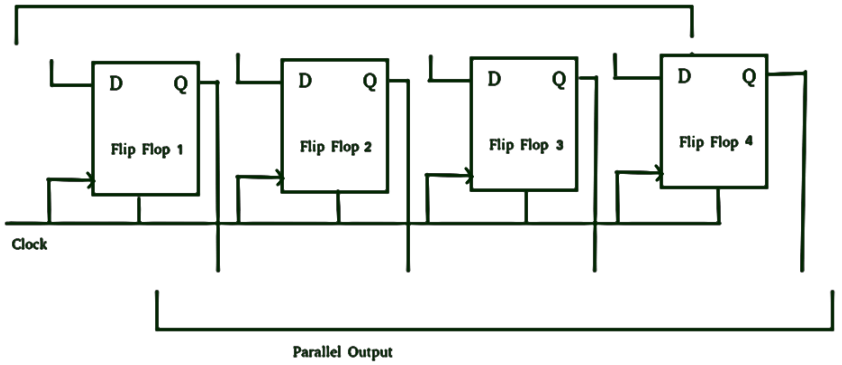
A 3-bit shift-register using 7474 positive-edge triggered FF

Sequential Circuits





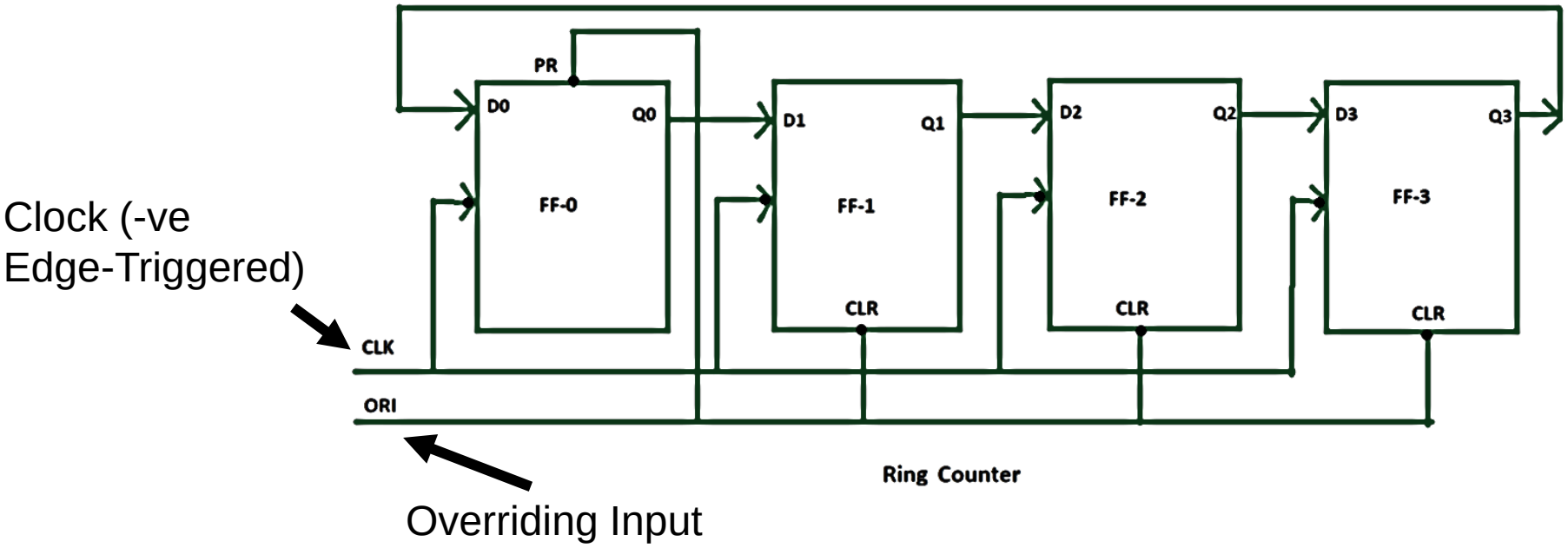




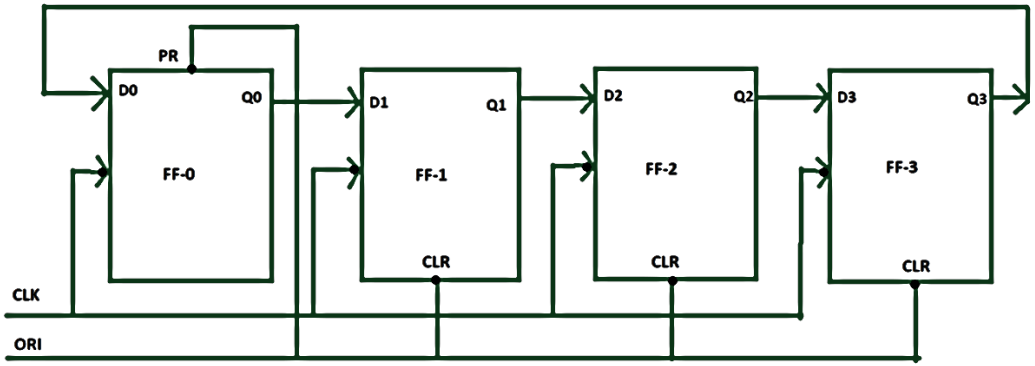
Sequential Circuits

Shift Register Application - Ring Counter:

- Typical **application of the Shift register**.
- The **output of the last flip-flop is connected to the input of the first flip-flop** in the case of the **ring counter** but in the case of the **shift register it is taken as output**.
- Except for this, **all the other things are the same.**



Sequential Circuits



Ring Counter

The clock pulse (**CLK**) is **simultaneously** applied to all the flip-flops. (**Synchronous Counter)**

Also, here we use Overriding input (**ORI**) for each flip-flop - **Preset (PR) and Clear (CLR) are used as ORI.**

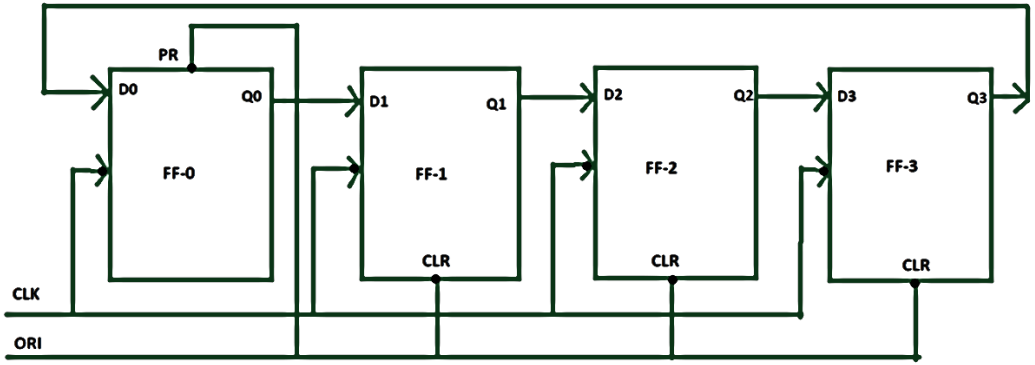
When **PR is 0, then the output is 1**, and when **CLR is 0, then the output is 0**.

PR and CLR are active low signals that always work in value 0.

PR = 0, Q = 1; CLR= 0, Q = 0

Sequential Circuits

- ORI is connected to Preset (PR) in FF-0 and it is connected to Clear (CLR) in FF-1, FF-2, and FF-3.
- Output Q = 1 is generated at FF-0, and the rest of the flip-flop generates output Q = 0.
- This output Q = 1 at FF-0 is known as Preset 1, which is used to form the ring in the Ring Counter.



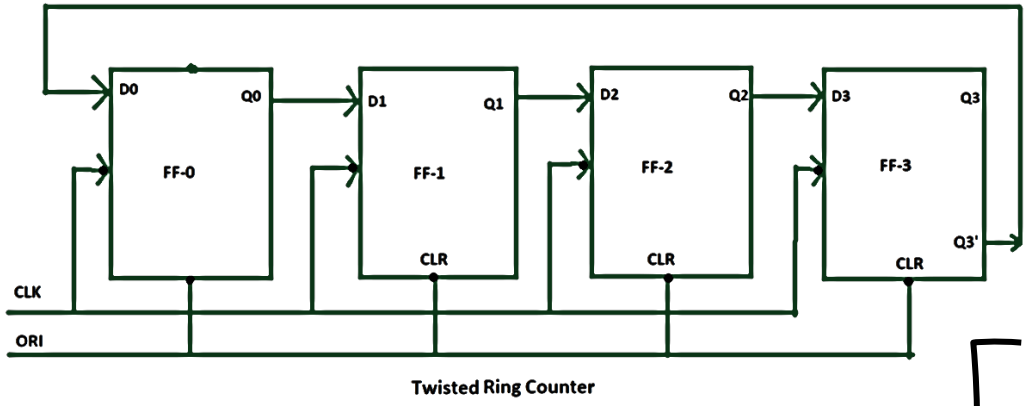
Ring Counter

Preset 1

ORI	CLK	Q0	Q1	Q2	Q3
0	X	1	0	0	0
1	0	0	1	0	0
1	0	0	0	1	0
1	0	0	0	0	1
1	0	1	0	0	0

Counter states (**modulo-n**)

Sequential Circuits



Twisted Ring Counter

Johnson's counter

- Counts 2*n states (**modulo-2n**)

Counter states

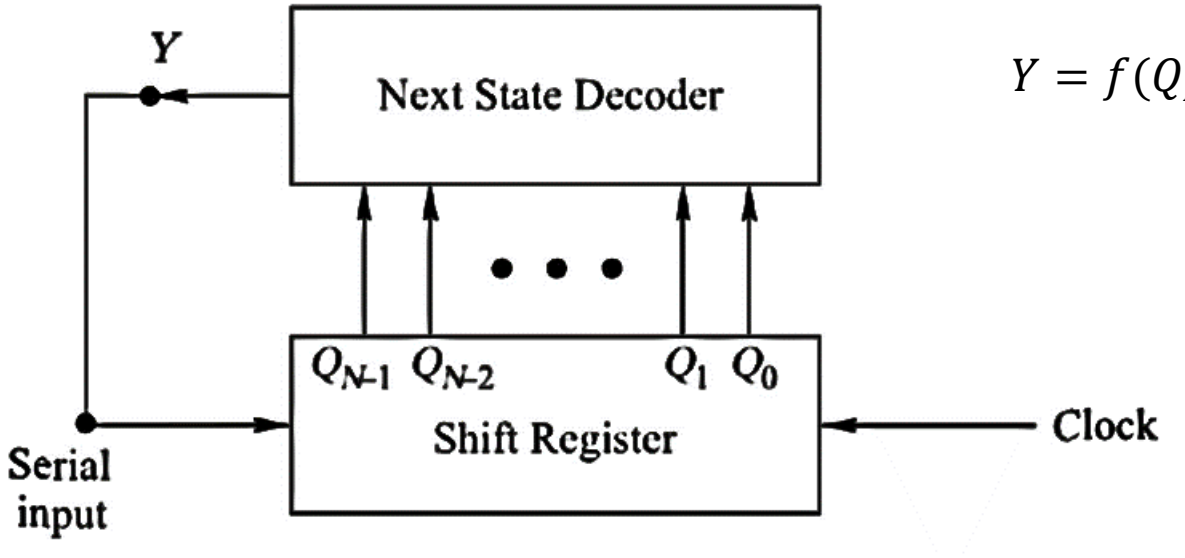
ORI	CLK	Q0	Q1	Q2	Q3
0	X	1	0	0	0
1	0	1	1	0	0
1	0	1	1	1	0
1	0	1	1	1	1
1	0	0	1	1	1
1	0	0	0	1	1
1	0	0	0	0	1
1	0	0	0	0	0
1	0	1	0	0	0

Sequential Circuits

STRAIGHT RING COUNTER	TWISTED RING COUNTER
It connects the output of the last shift register to the input of first shift register.	It connects the complement of output of the last shift register to the input of the first register.
It is known as One hot counter.	It is known as Walking ring counter or Johnson's counter.
Number of states = number of flip-flops	Number of states = 2 x number of flip-flops
It circulates a single bit (0 or 1) around the ring.	It circulates stream of 1 followed by stream of 0.
PRESET is used in first shift register.	PRESET is not used in twisted ring counter.
CLEAR is used for last (n-1) flip-flops.	CLEAR is used for all flip-flips in it.
It is used in successive approximation and stepper motor control.	It is used in phase shift or function generator.

Sequential Circuits

- Shift registers are also utilized in **Sequence generator** circuits
- **Sequence generator:** generates a given sequence of bits (in synchronism with a clock)



$$Y = f(Q_{N-1}, Q_{N-2}, \dots, Q_1, Q_0)$$

The minimum no. of FFs (N) required to generate a sequence of length S-

$$S \leq 2^N - 1$$

Sequential Circuits

Q: Generate a sequence …1101011…

The minimum no. of FFs (N) required to generate a sequence of length 7: $7 \leq 2^N - 1$. So, $N \geq 3$.

No. of clock pulses	Flip-flop outputs		
	Q_2	Q_1	Q_0
1	1	1	1
2	1	1	1
3	0	1	1
4	1	0	1
5	0	1	0
6	1	0	1
7	1	1	0
.	.	.	.
.	.	.	.

Same states $\Rightarrow N > 3$

$\Rightarrow N = 4$

Sequential Circuits

Q: Generate a sequence …1101011…

$N = 4$

No. of clock pulses	Flip-flop outputs				Serial input Y
	Q_3	Q_2	Q_1	Q_0	
1	1	1	1	0	1
2	1	1	1	1	0
3	0	1	1	1	1
4	1	0	1	1	0
5	0	1	0	1	1
6	1	0	1	0	1
7	1	1	0	1	1
1	1	1	1	0	1
2	1	1	1	1	0
3	0	1	1	1	1

Diff. states

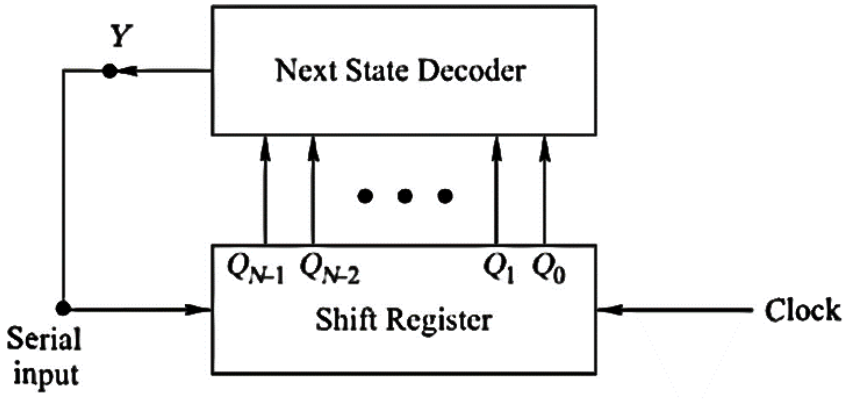
Sequential Circuits

Q: Generate a sequence …1101011…

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

$\overline{Q_3}$ $\overline{Q_1}$ $\overline{Q_0}$

$$Y = \overline{Q_3} + \overline{Q_1} + \overline{Q_0}$$



Sequential Circuits

Counters:

- Asynchronous (ripple counter): All FFs **not clocked simultaneously**
- Synchronous (Ring: **modulo-n,** Twisted ring: **modulo-2n**) counter: All FFs **clocked simultaneously**

However, total possible states with n-FFs**:** $\mathbf{2^n}$

$\Rightarrow$ **(**$\boldsymbol{modulo - 2^n}$**)** counter is possible

up-counter

Counter state	Count		
	Q_2	Q_1	Q_0
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

up-counter & down-counter

Q_0 changes from 0 to 1

Q_0 changes from 1 to 0, Q_1 changes from 0 to 1

Q_0 changes from 1 to 0, Q_1 changes from 1 to 0, Q_2 changes from 0 to 1

Q_0 changes with every clock pulse
Q_1 changes with every 2-clock pulses
Q_2 changes with every 4-clock pulses

Sequential Circuits

Design Procedure:

- Q_0 needs to change with every clock pulse.

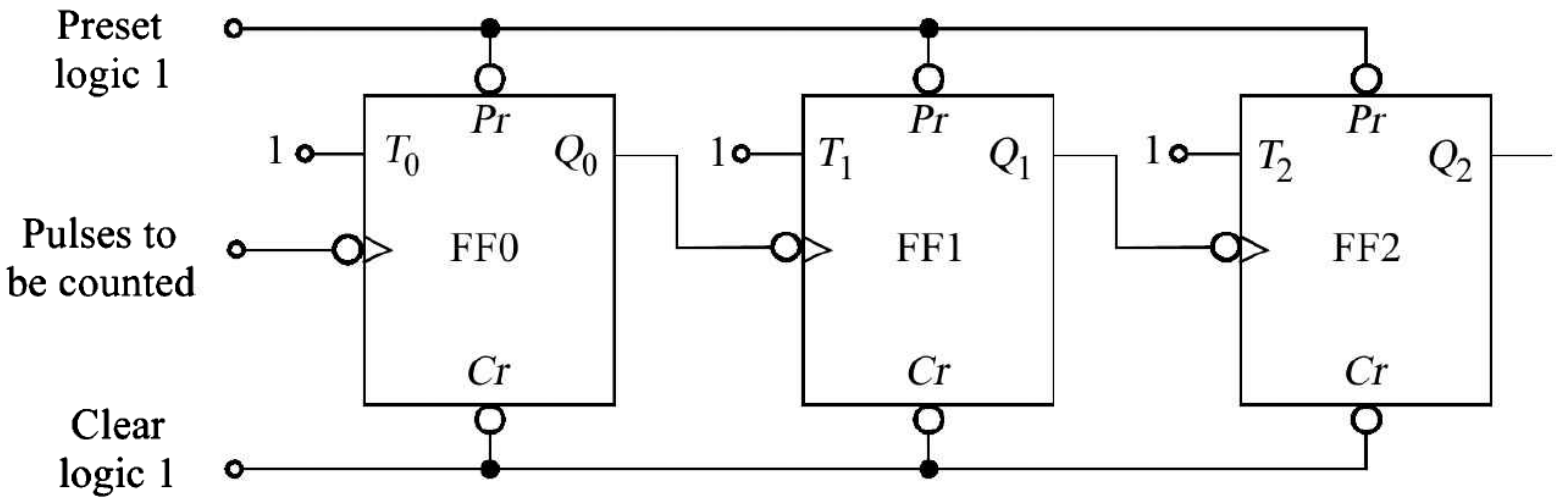
So, the input to FF_0 is T_0=1, and the external clock is provided to FF_0.

- Q_1 toggles when Q_0 goes from 1 to 0.

It means that the Negative edge of Q_0 toggles Q_1. So we can use Q_0 as the clock input for FF_1.

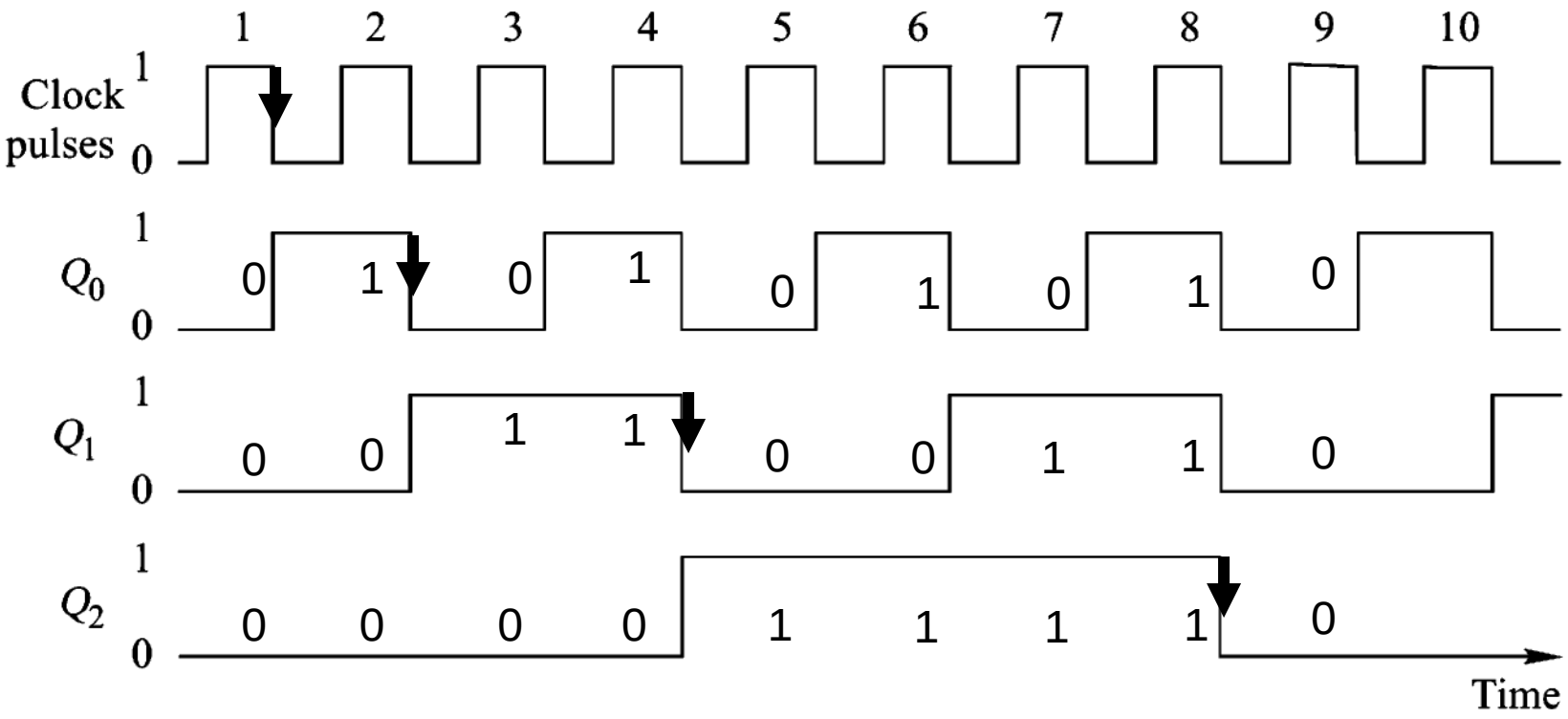
- Q_2 toggles when Q_1 goes from 1 to 0 (negative edge).

This is used as a clock signal for FF_2.



A 3-bit binary counter

Sequential Circuits



Sequential Circuits

Q. Design a mod-5 synchronous up counter using J-K flip flop

Q_c	Q_b	Q_a
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

Desired states

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

Excitation table of JK FF

Sequential Circuits

Q. Design a mod-5 synchronous up counter using J-K flip flop

Q_c	Q_b	Q_a
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

Desired states

Present state			Next state			Flip flop inputs					
Q_c	Q_b	Q_a	Q_{c+1}	Q_{b+1}	Q_{a+1}	J_c	K_c	J_b	K_b	J_a	K_a
0	0	0	0	0	1	0	X	0	X	1	X
0	0	1	0	1	0	0	X	1	X	X	1
0	1	0	0	1	1	0	X	X	0	1	X
0	1	1	1	0	0	1	X	X	1	X	1
1	0	0	0	0	0	X	1	0	X	0	X
1	0	1	X	X	X	X	X	X	X	X	X
1	1	0	X	X	X	X	X	X	X	X	X
1	1	1	X	X	X	X	X	X	X	X	X

Excitation table for mod-5 counter

Sequential Circuits

Q_a \ Q_cQ_b	00	01	11	10
0	0	0	x	x
1	0	1	x	x

$J_c = Q_aQ_b$

Q_a \ Q_cQ_b	00	01	11	10
0	x	x	x	1
1	x	x	x	x

$K_c = 1$

Q_a \ Q_cQ_b	00	01	11	10
0	0	x	x	0
1	1	x	x	x

$J_b = Q_a$

Q_a \ Q_cQ_b	00	01	11	10
0	x	0	1	x
1	x	x	x	x

$K_b = Q_c$

Q_a \ Q_cQ_b	00	01	11	10
0	1	1	x	0
1	x	x	x	x

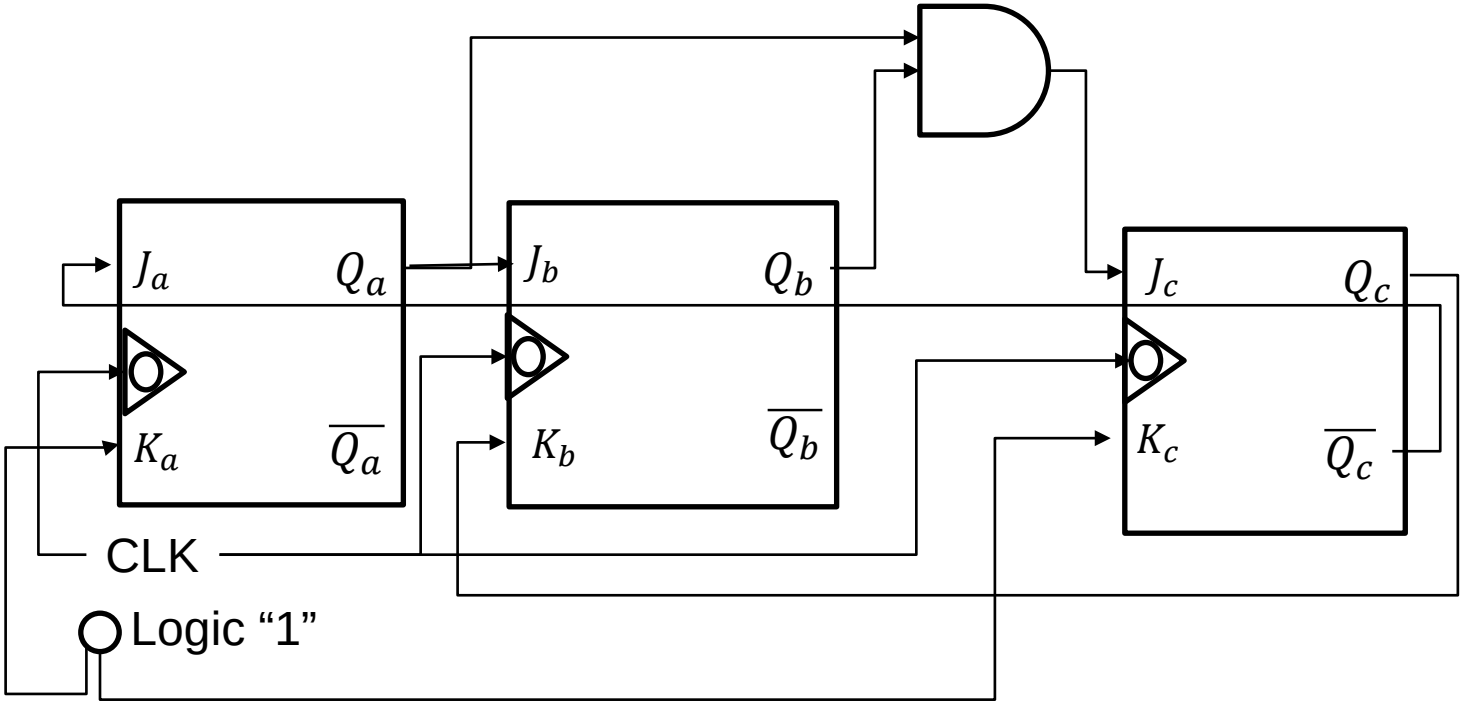
$J_a = \overline{Q_c}$

Q_a \ Q_cQ_b	00	01	11	10
0	x	x	x	x
1	1	1	x	x

$K_a = 1$

Sequential Circuits

$J_c = Q_a Q_b$ $K_c = 1$ $J_b = Q_a$ $K_b = Q_c$ $J_a = \overline{Q_c}$ $K_a = 1$



mod-5 synchronous up-counter using J-K flip flop

Sequential Circuits

Special counter ICs:

- 74HC161
- 74HC163
- 74HC191
- 74HC160
- CD4017B

Applications of counters:

- Pulse counting
- Frequency division
- Digital clocks
- Analog-to-digital converters