IS1102 Computer Systems

Sequential Logic Circuits

Digital Logic Circuits

There are two types of digital logic circuits

1. Combinational Logic circuits
2. Sequential Logic circuits

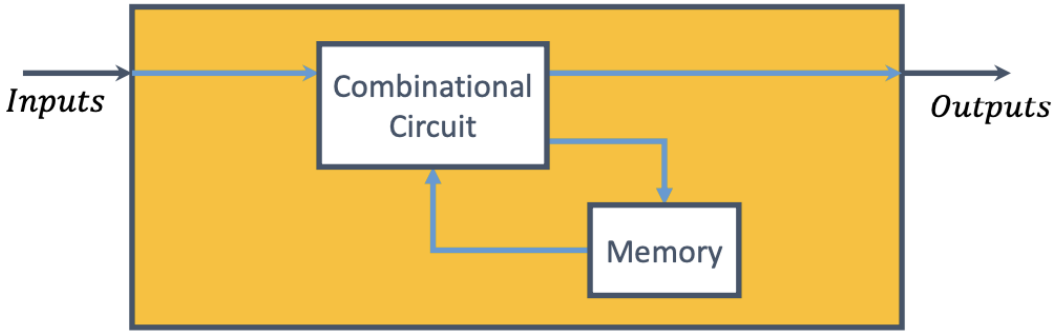
Combinational Circuit - Recap

- 2^n possible input patterns
- Circuits can be represented using
  - Truth table
  - Boolean function
  - Logic gates

- Examples:
  - half adders, full adders, multiplexers, demultiplexers, encoders and decoder

Sequential Logic circuit

- A combination of combinational circuit and memory
- Present output depends on the present input and past output
- Examples : Flip flops , Counters, Registers



Combinational vs Sequential

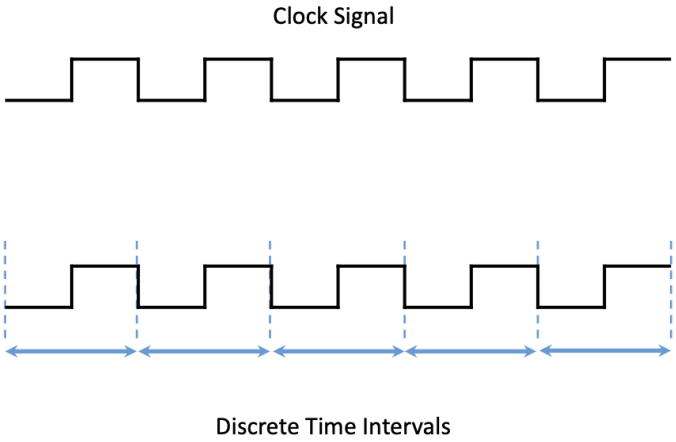
Combinational Circuits	Sequential Circuits
The output of a combinational circuit depends only on its current inputs.	Consider not only its present inputs, but also the previous outputs that are stored in storage elements.
Purely built upon logic gates	Stores previous outputs.
No feedback paths or memory elements	

Classification of Sequential Logic circuits

Synchronous Sequential Circuits (Clock Driven)	Asynchronous Sequential Circuits (Event Driven)
State changes only within discrete time intervals.	State changes immediately after inputs are changed.
Easy to design	Difficult to design
Use clock signals	Do not use clock signals
Slower	Faster as clock is absent

What is a clock signal?

- A signal which oscillates between logic level 0 and logic level 1, repeatedly.
- It controls the outputs of the sequential circuit
- It determines when and how the memory elements change their outputs



Storage elements in sequential circuits

- Flip Flops
- Latches

Four main types of flip flops and latches

1. SR
2. D
3. JK
4. T

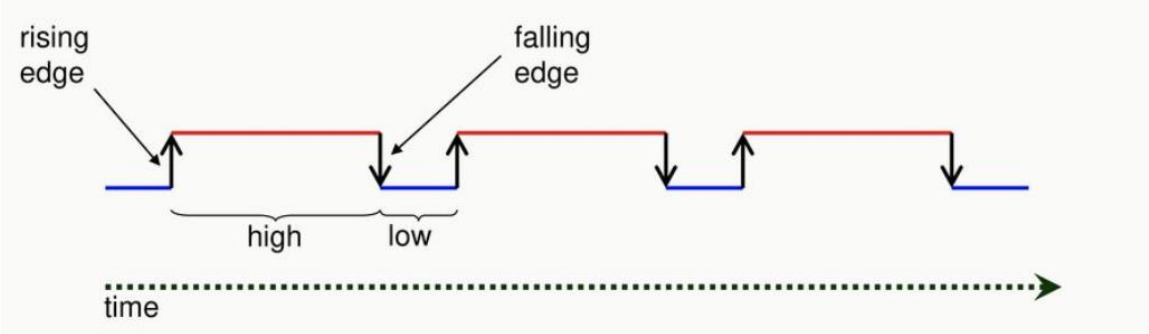
Flip flops vs. Latches

Flip Flops	Latches
Building block of sequential circuits, built using latches along with an additional clock signal.	Building block of sequential circuits, built using logic gates
slow compared to the latches	fast compared to the Flip-Flops
Has a clock signal	Does not has a clock signal
An edge sensitive device	A level sensitive device

Clock signal vs. Enable Signal

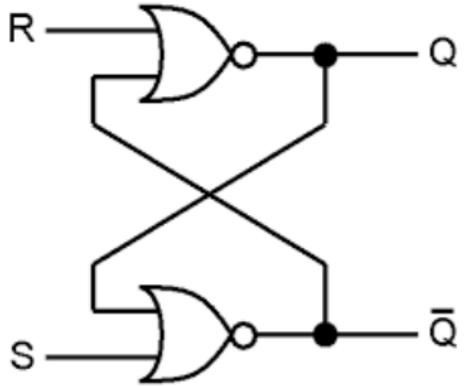
A clock (clk) is a signal which is used to make the flip flop work at its positive or negative edge (in exceptional case both edge) in the digital system.

An enable is a signal which makes the flip flop function as long as it is high (1). It can be made low (0) to make the flip flop stops its function.



S-R Latch

- **S** – Set, **R** – Reset
- Two cross coupled NAND gates can also be used.
- Two outputs; One is the complement of the other.
- When Q=1 -> Set State
- Otherwise Reset State

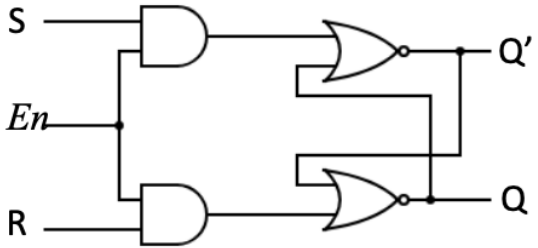


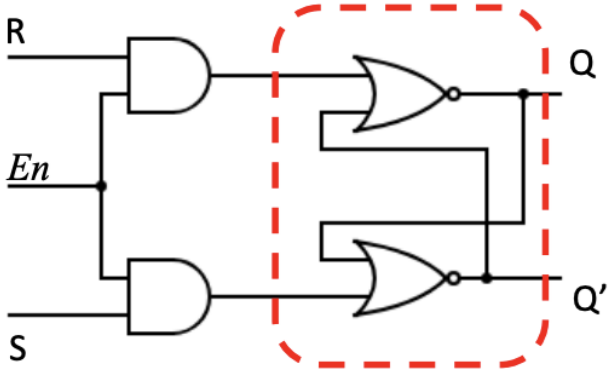
S	R	Q(t+1)
0	0	Q(t)
0	1	0
1	0	1
1	1	Forbidden

S-R Latch with Control Signal

- *En* – Enable
- When *En* = 0 circuit is a SR Latch with S=0 and R=0

- *En* should be triggered to make the S-R Latch operational.

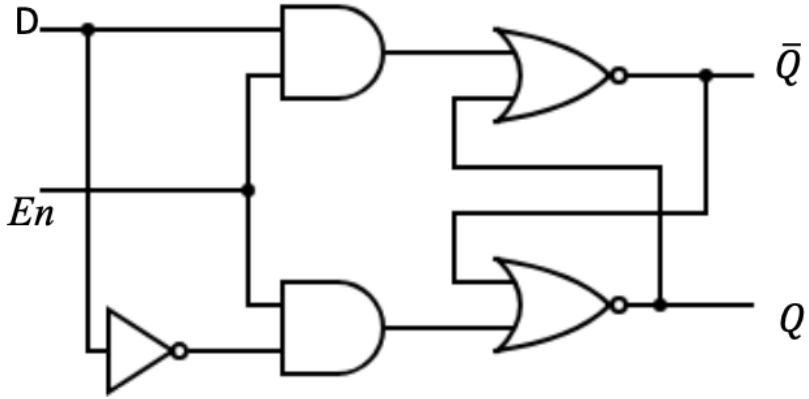




En	S	R	Q(t+1)
0	X	X	Q(t)
1	0	0	Q(t)
1	0	1	0
1	1	0	1
1	1	1	Forbidden

D Latch

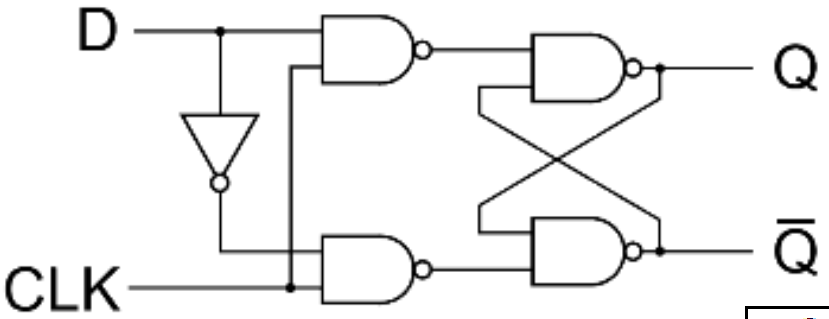
- Eliminate undesireable condition of S-R Latch

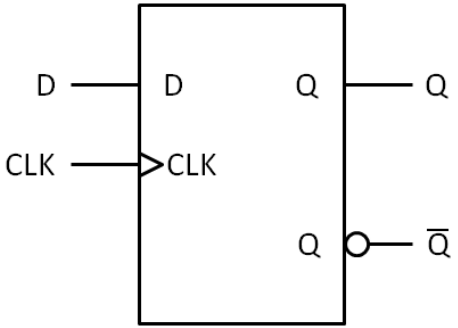


En	D	Q(t+1)
0	X	Q(t)
1	0	0
1	1	1

- Value in goes to out (Q) when the circuit is enabled.
- When disabled value Q remains unchanged.

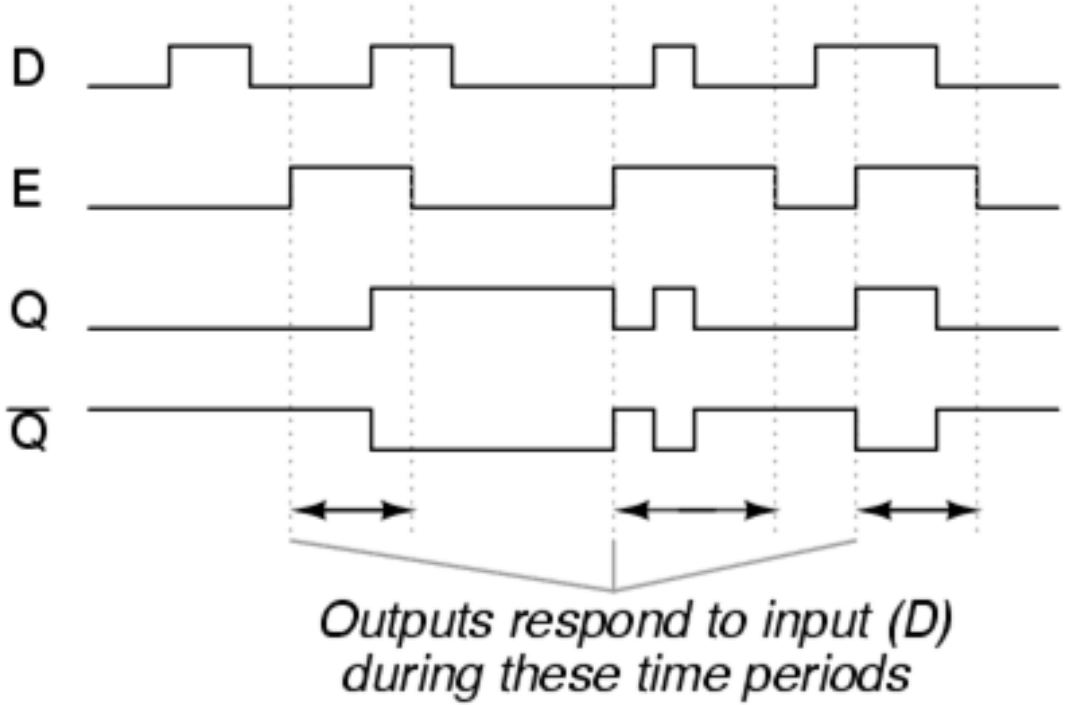
D flip flop



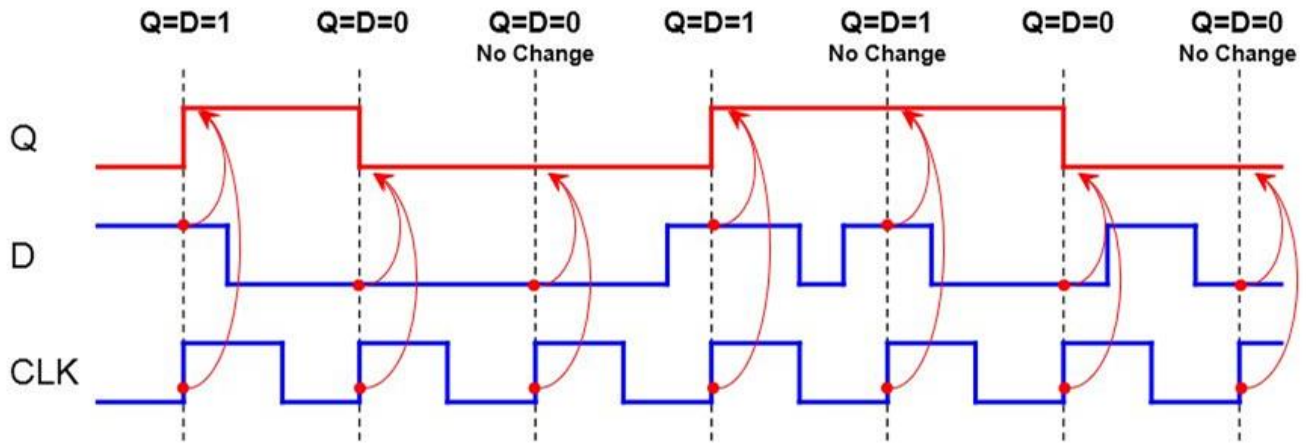


Clock	D	Q	Q′	Description
↓ » 0	X	Q	Q′	Memory no change
↑ » 1	0	0	1	Reset Q » 0
↑ » 1	1	1	0	Set Q » 1

D latch : timing (example)

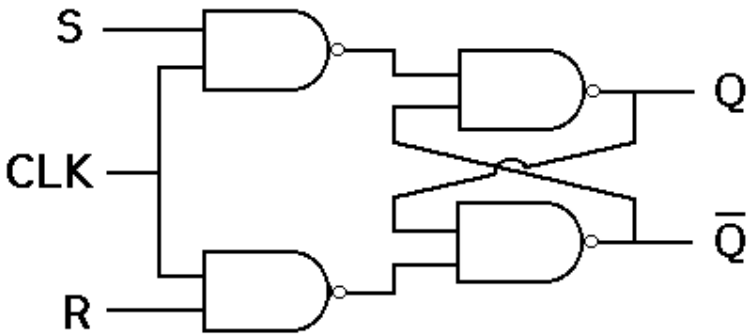


D flip flop : timing (example)



Clocked S-R Flip Flop

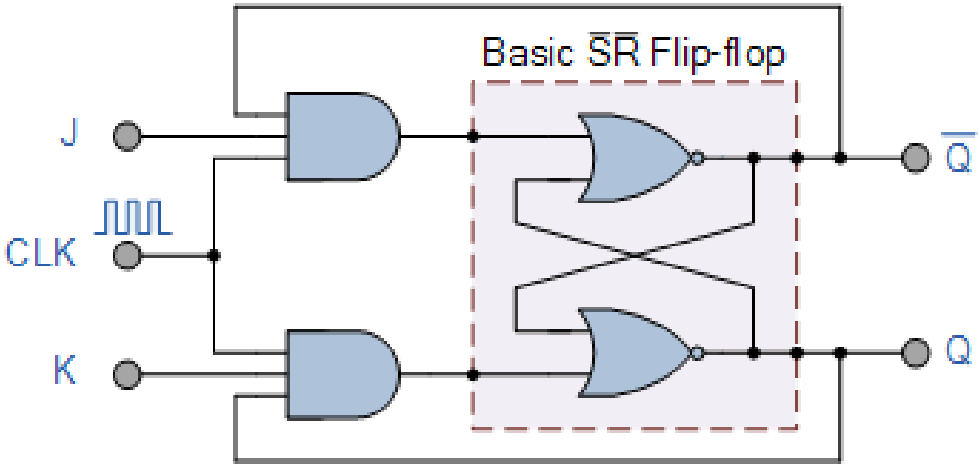
- Stands for SET-RESET flip-flops
- Basic flip-flop among all the flip-flops.
- All the other flip flops are developed after SR-flip-flop.
- When both S and R are 1, the SR flip-flop is unstable

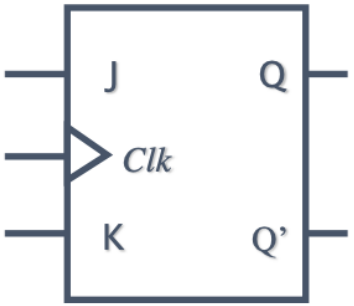


CLK	S	R	Q(t+1)
0	x	x	Q(t)
1	0	0	Q(t)
1	0	1	0
1	1	0	1
1	1	1	Forbidden

J-K Flip flop

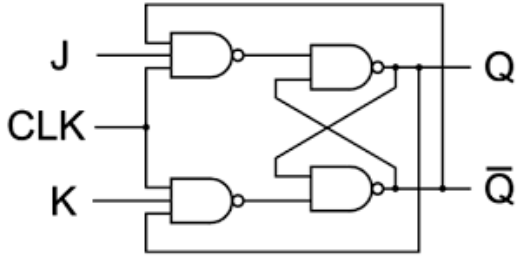
A refinement of SR flip-flop

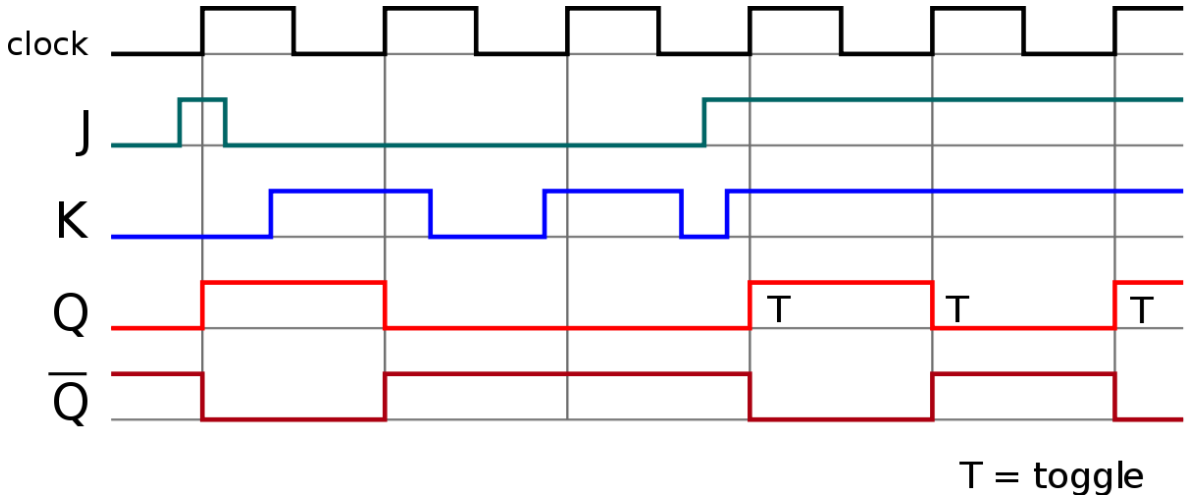




J	K	Q(t+1)
0	0	Q(t)
0	1	0
1	0	1
1	1	Q'(t)

J-K Flip flop (Timing example)



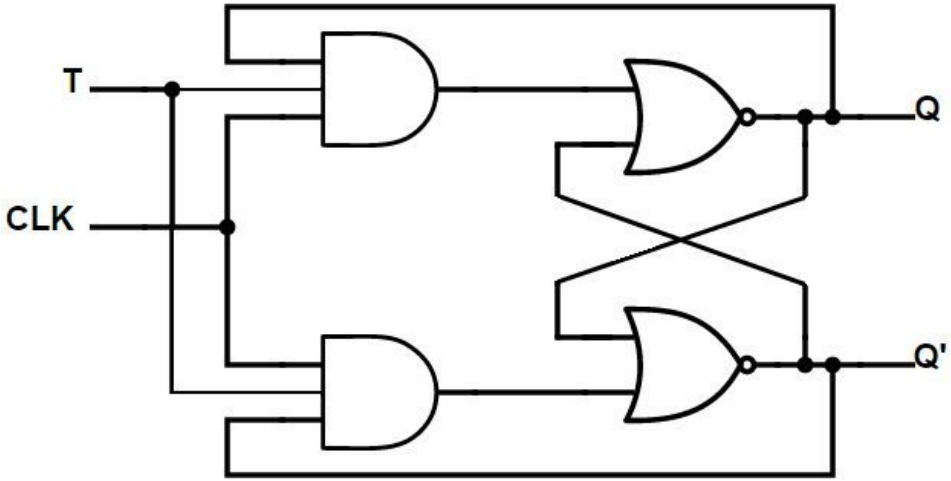


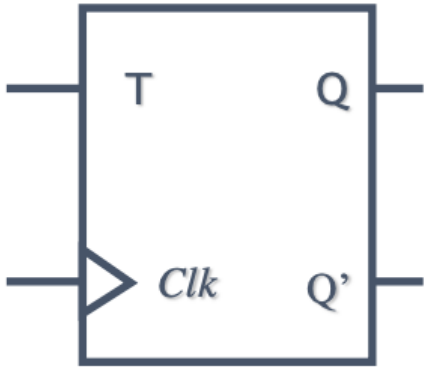
T = toggle

J	K	Q(t+1)
0	0	Q(t)
0	1	0
1	0	1
1	1	Q'(t)

T Flip flop

A single input version of JK flip-flop (both inputs connected together).



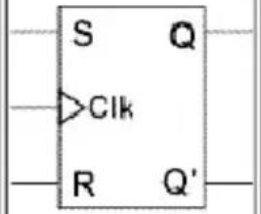


T Flip-Flop

T	Q(t+1)
0	Q(t)
1	Q'(t)

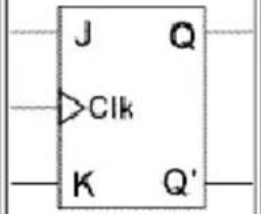
Flip flop - Summary

SR



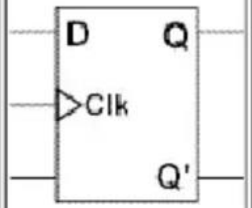
S	R	Q(next)
0	0	Q
0	1	0
1	0	1
1	1	?

JK



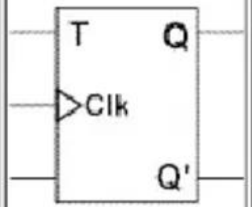
J	K	Q(next)
0	0	Q
0	1	0
1	0	1
1	1	Q'

D



D	Q(next)
0	0
1	1

T



T	Q(next)
0	Q
1	Q'

Thank You