CSE 4205
Digital Logic Design

Sequential Logic

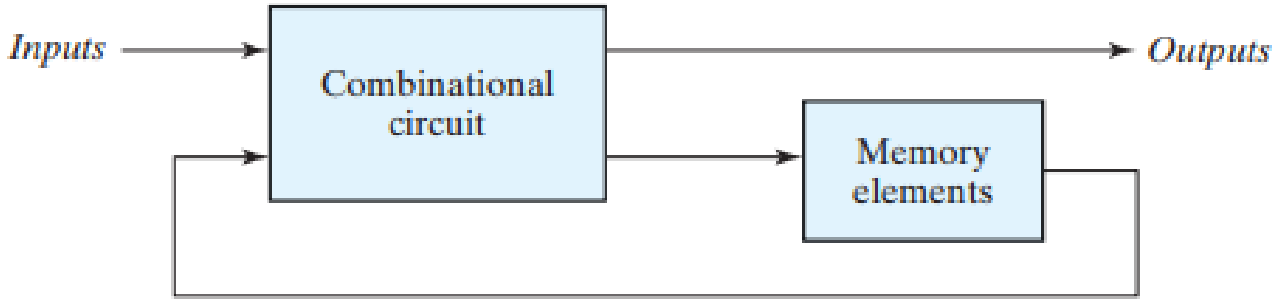
Course Teacher: Md. Hamjajul Ashmafee

Lecturer, CSE, IUT

Email: ashmafee@iut-dhaka.edu

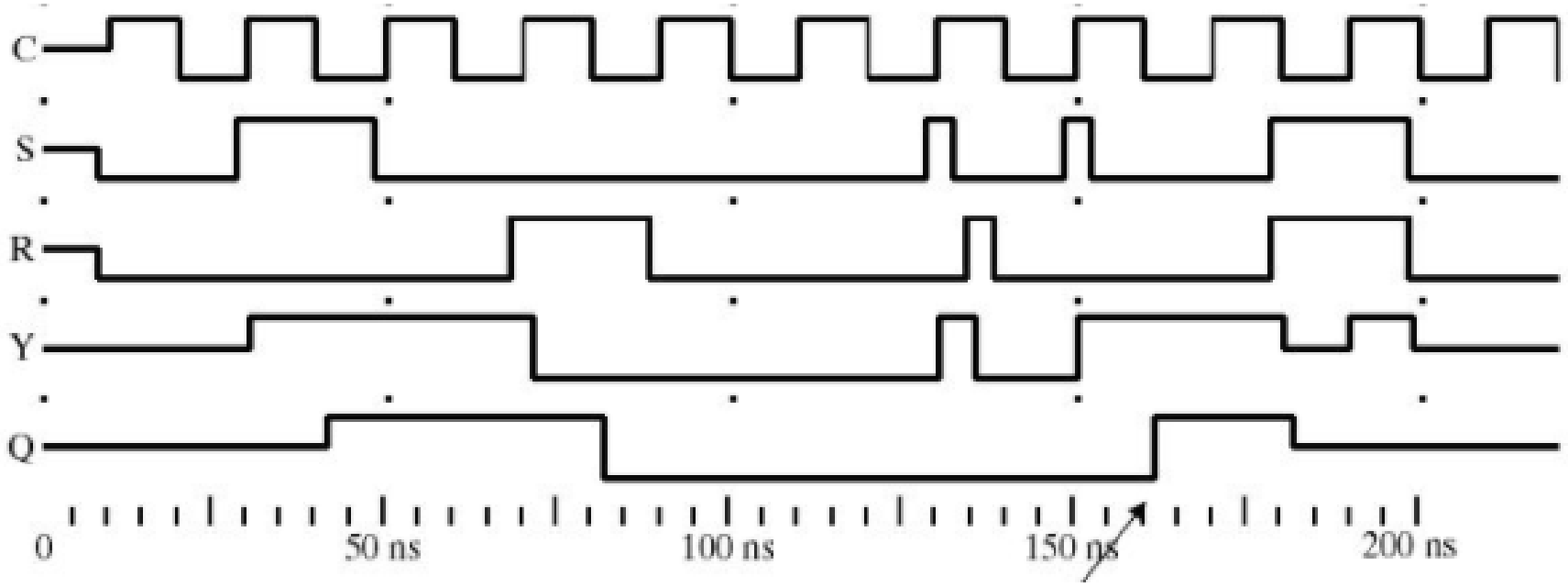
Sequential Circuit

- $\boldsymbol{Combinational\ circuit\ +\ storage\ elements\ =\ Sequential\ circuit}$
- **Storage elements** stores binary information (1 or 0) which are called **states**
  - **This states are time variant**
- ***Two possible outcomes of a sequential circuit:***
  - ***Present output of the sequential circuit*** = F_1 *(present input, present state)*
  - ***Next state of memory element*** = F_2 *(present input, present state)*
- **Storage element feeds back its state as input** to the combinational circuit through a **feedback path**
  - **This feedback path is called a loop**
- **Example**: Counter, Register



Time Sequence

- Sequential circuit includes **time sequence** of inputs, outputs and internal memory states
  - This time-sequence determines the behavior of a sequential circuit.



Time sequence of a S-R flip-flop

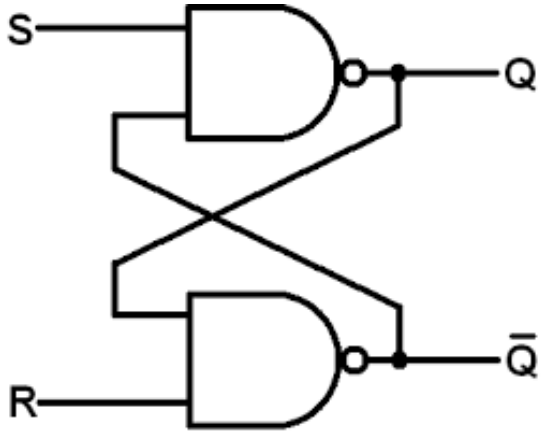
Classification of Sequential Circuit

***Two** types of sequential circuit* – based on **timing** of their input signal

1. Synchronous or clocked sequential circuit
2. Asynchronous or unclocked sequential circuit

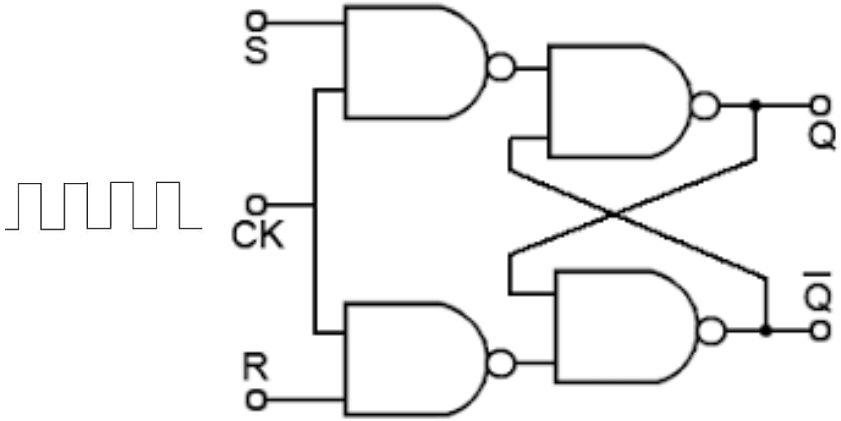
Asynchronous Sequential Circuit

- Behavior defined based on the **order/sequence** of input signals at **any instant** of time
  - Only affected whenever the inputs change
  - **No synchronization** with clock pulse
  - Sometimes, activated with **enable input**
- **Example**: Latch

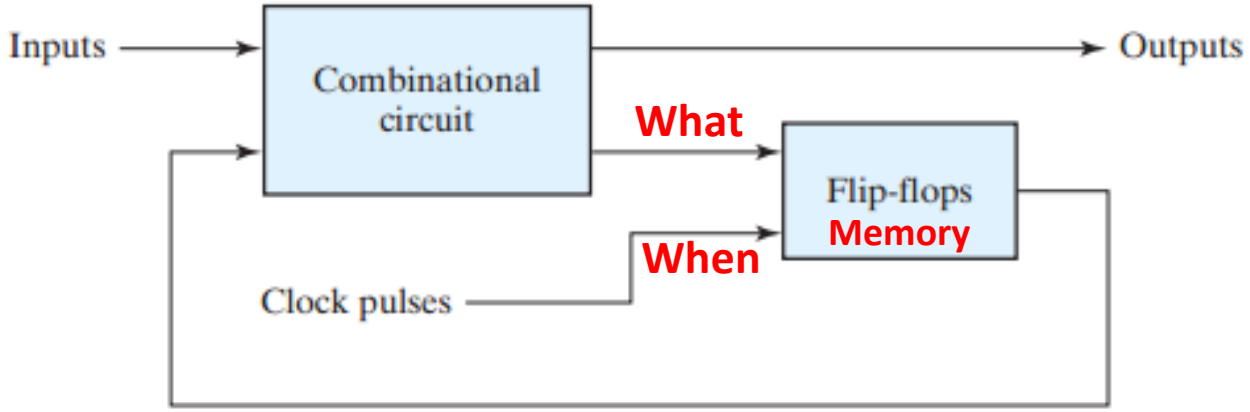


Synchronous Sequential Circuit

- Behavior defined based on the input signals at **discrete** instants of time
- **Example**: Flip-flop



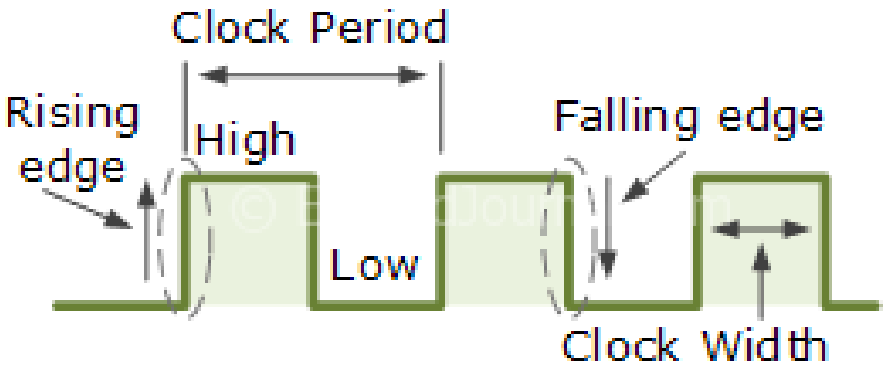
Clocked Sequential Circuit



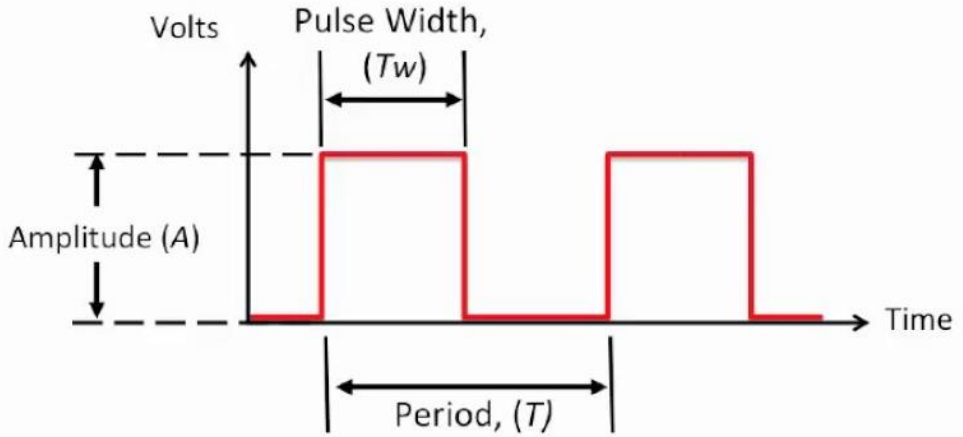
(a) Block diagram

(b) Timing diagram of clock pulses

Clock



Duty Cycle = ratio of the Pulse Width to the Period



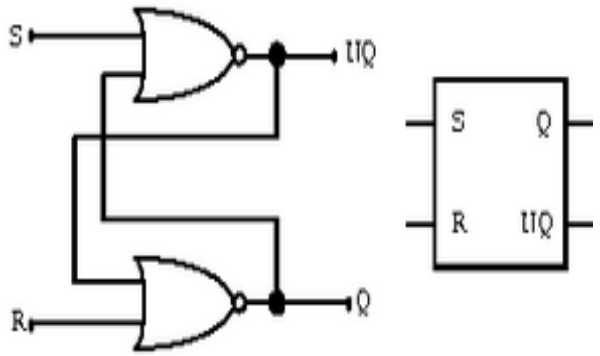
T_w = pulse duration

Flip-flops

- **Memory elements** in clocked sequential circuit
- A binary/digital memory cell that stores one bit of information
- **Number of inputs – One or more**
- **Two outputs** – **normal (Q)** and **complement (Q')** value of the bit stored
- Also known as a **bistable multivibrator**

Latch

- Another variant of **memory element**
  - **Propagation delay** is used in this regard
- It has two stable states
- Used to store binary information
  - Binary memory element

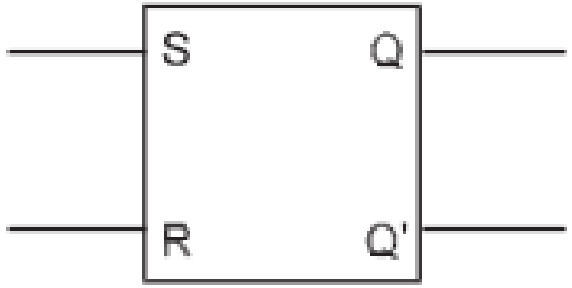


Types of Flip-flops

- Based on how the inputs and clock pulses cause transition between two states:
  1. **S-R** (Set-Reset) flip-flop
  2. **D** (Data) flip-flop
  3. **J-K** (J: set, K: reset) flip-flop
  4. **T** (Toggle) flip-flop

S-R (Set-Reset) flip-flop

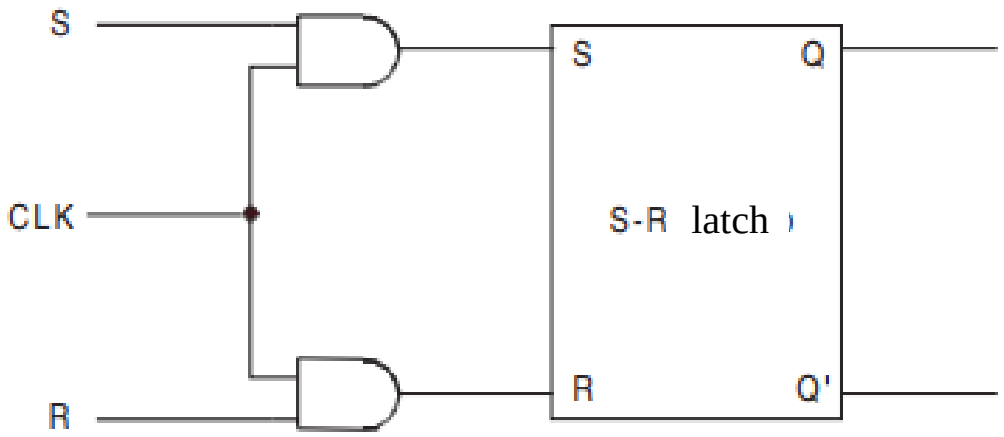
- Two inputs – Set (S) and Reset (R)
- Two outputs – Q and Q’ (complement to each other)
- Made of **cross-coupled connection** from output to input constituting a **feedback path**

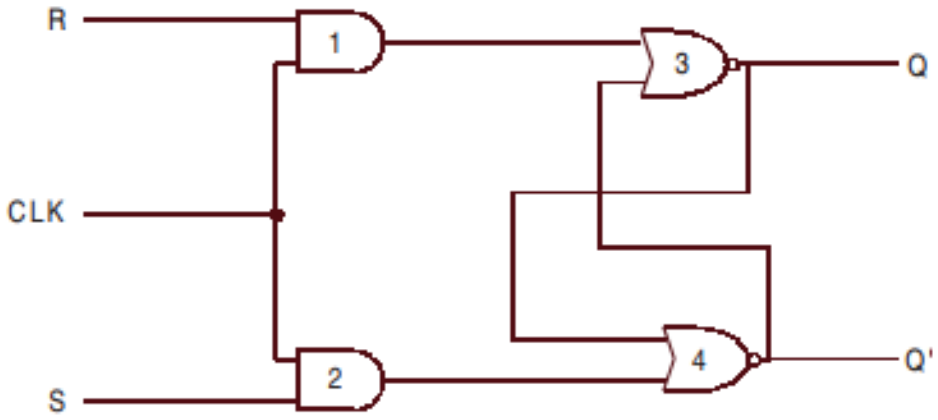


Inputs		*Output*
S_n	R_n	Q_{n+1}
0	0	Q_n
0	1	0
1	0	1
1	1	–

Clocked S-R Flip-Flop

- **Synchronous circuit**
  - Activated during a clock pulse is present
- From a latch, a **flip-flop** can be produced including **an additional input** to control the behavior of the circuit





Characteristic Table and Truth Table

- **Characteristic table** and **truth table**:
  - **Truth table** with **present input(s)** and **present state(s)** as input column and **next state(s)** as output column refers to a **characteristic table**
  - It refers to the **operational characteristics** of the flip-flop

S-R Flip-Flop - Characteristic table

- **Characteristic table** and **Characteristic equation** of S-R 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

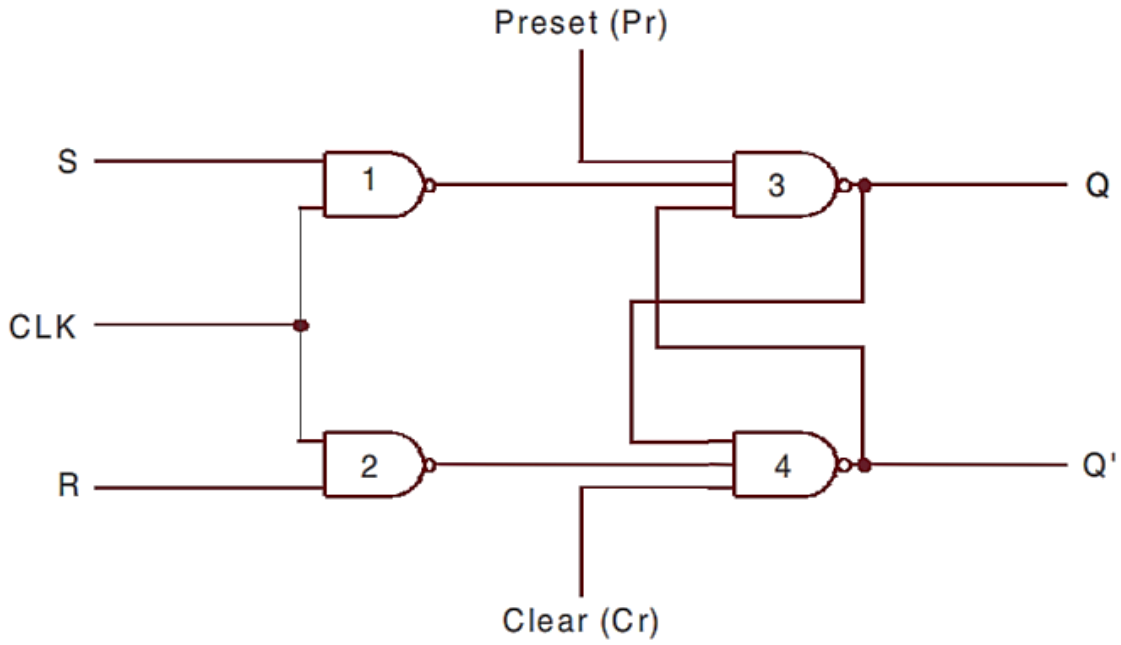
$$Q_{n+1} = S + R'Q_n$$

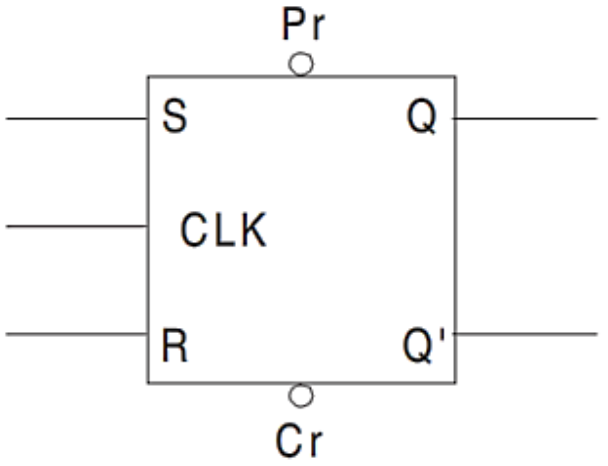
$$SR = 0.$$

Preset and Clear Inputs

- Initially (at t_0), a circuit is **uncertain** with Q and Q' states
- They should be assigned with either ***set*** or ***reset*** state with direct or asynchronous inputs
  - Also, called as **overriding** inputs
  - These inputs are individually named as ***preset (Pr)*** and ***clear (Cr)***
- These inputs are not synchronized with clock
  - They can be applied at any time
  - $CLK = 0 \; or \; 1$, doesn't care at all [but, preferred to be applied when clock is inactive]
- $\mathbf{Pr = 0}$ and $\boldsymbol{Cr = 0}$ must not be applied, as it leads to an uncertain state
- ***Pr*** and ***Cr*** inputs are used with bubbles considering them **active low**

Preset and Clear Inputs…





Preset and Clear Inputs…

Inputs		*Output*	*Operation*
Cr	*Pr*	*Q*	*performed*
1	1	**Based on inputs**	Normal flip-flop
1	0	1	Preset
0	1	0	Clear
0	0	–	Uncertain

D Flip-Flop

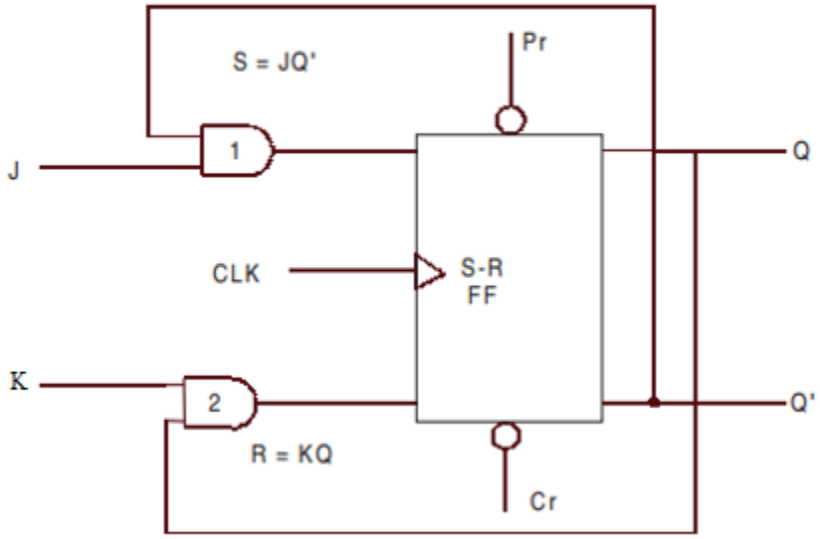
- Derivation from S-R flip-flop
- **Truth table** and **all cases** of D flip-flop
- **Block diagram** of D flip-flop
- D flip-flop with ***Pr*** and ***Cr*** inputs
- **Characteristic table** and **equation** of D flip-flop
- **Application – register**

J-K Flip-Flop

- Very similar to an S-R flip-flop
  - **J and K** behave like inputs **S and R** to set and reset the flip-flop
- **Advantage**: **Eliminates** the **undefined condition** of an S-R flip-flop
  - Only difference between S-R and J-K flip-flop
  - When J=K=1, the flip-flop is said to be in a ***toggle state***
  - ***Toggling*** means complementing the state in each clock pulse

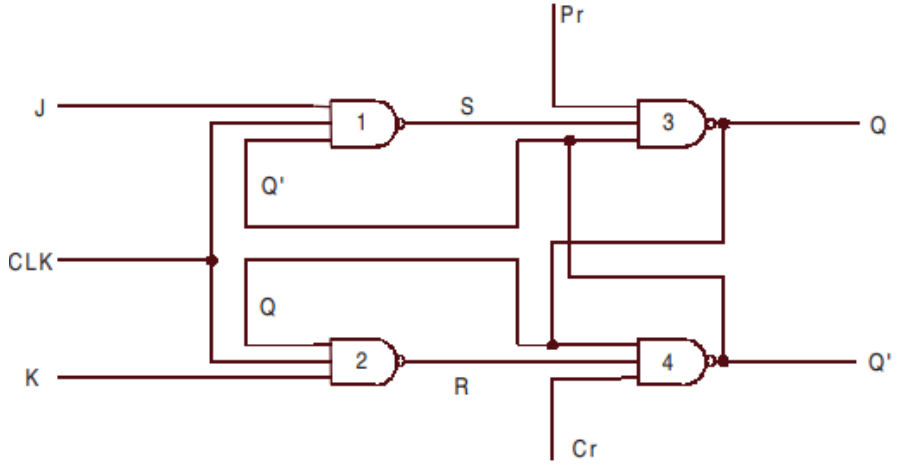
J-K Flip-Flop – Implementation

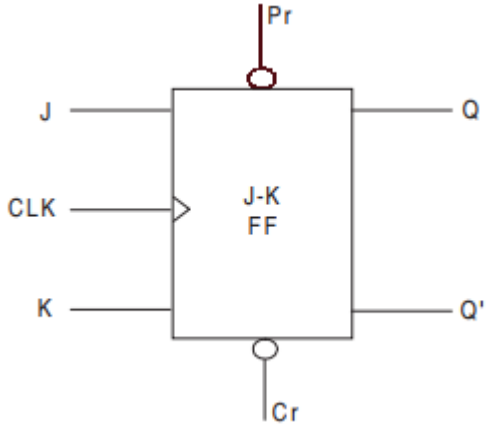
- Implementation **from S-R flip flop**
  - Another **level of feedback** path is included
  - Inputs **J and K** of J-K flip-flop are ANDed (**cross-coupled**) with **Q' and Q** respectively to obtain the inputs, S and R for S-R flip-flop



J-K Flip-Flop – Implementation…

- Circuit diagram and block diagram of a J-K flip flop with NAND gates





J-K Flip-Flop – Truth Table

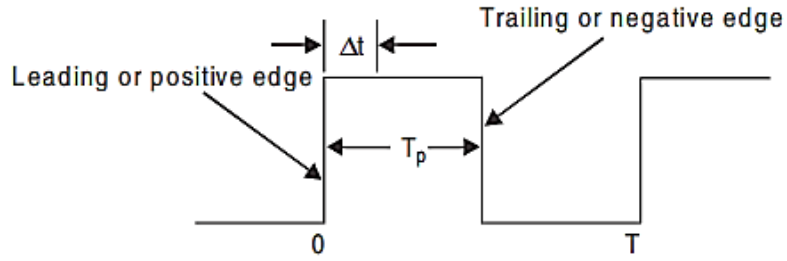
Inputs		*Output*
J_n	K_n	Q_{n+1}
0	0	Q_n
0	1	0
1	0	1
1	1	Q'_n

J-K Flip-Flop – Characteristic Table

- Characteristic table and characteristic equation for J-K flip-flop

J-K Flip-Flop – Race Around Condition

- Assumption for basic J-K flip-flop:
  - The inputs will not be changed any more during a clock pulse
  - But, it is not true because of the feedback connections
- Consider the scenario: **J=K=1**
    - If Q=1, after **Δt time**, it will be Q=0
    - Next, if Q=0, after another **Δt time**, it will be Q=1
  - So, basically, **during t_p time**, the output Q will oscillate between 0 and 1 after each Δt time
  - And, **after t_p time,** the output Q is **not certain**.
  - This situation is called "***Race-Around Condition***"



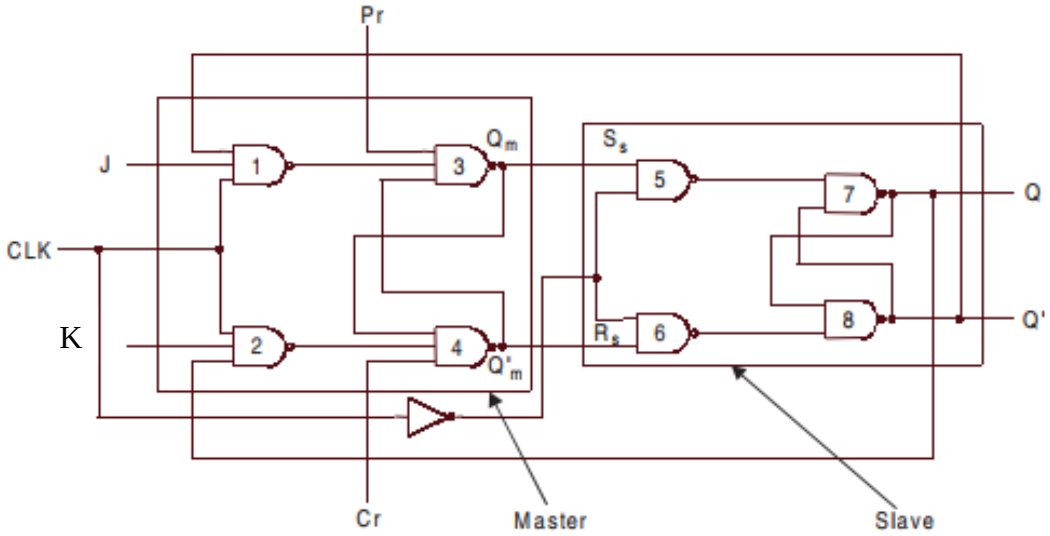
Δt = propagation delay
t_p = pulse duration
Level triggered clock pulse

Race Around Condition – Solutions

1. If $\triangle t$ is in nanoseconds and t_p in microseconds, output will be changed thousands of times within a clock pulse
2. Master-Slave (M-S) flip-flop
3. Edge triggered flip-flop

Race Around Condition – Master-Slave J-K Flip-Flop

- A system of two **cascaded** flip-flops – master and slave flip-flops
- A **clock pulse** is applied to **master** and the **inverted form of the same clock** is applied to the **slave**
- **How to solve RAC problem?**
  - When CLK=1, gates 5 and 6 will not respond. Output Q and Q' will only change when CLK=0.



T Flip-Flop

- A slight modification of the J-K flip-flop
  - J and K inputs are tied together – **have same input**
  - Referred as **T flip-flop**
    - **It has the ability to toggle**
- One input (**T**) and two outputs **Q** and **Q'**
- Two states – **toggle (T=1)** and **memory (T=0)** states
- **Usage:** Counter, switching circuits

Triggering of Flip-flops

- Level Triggering
- Edge Triggering

Excitation Table

- A **tabulation** to figure out **all possible input conditions** to obtain the **desired output condition**
  - It is required to **design** ***a sequential circuit*** when present state and next state of the flip-flop is specified

Excitation Table of all Flip-Flops

Present State (Q_n)	Next State (Q_{n+1})	S-R FF		D-FF	J-K FF		T-FF
		S_n	R_n	D_n	J_n	K_n	T_n
0	0	0	X	0	0	X	0
0	1	1	0	1	1	X	1
1	0	0	1	0	X	1	1
1	1	X	0	1	X	0	0

Interconversion of Flip-flops

- A **conversion logic** is required to be designed for such conversions of flip-flops