DIGITAL ELECTRONICS (BCA 103)

DEPARTMENT OF COMPUTER SCIENCE

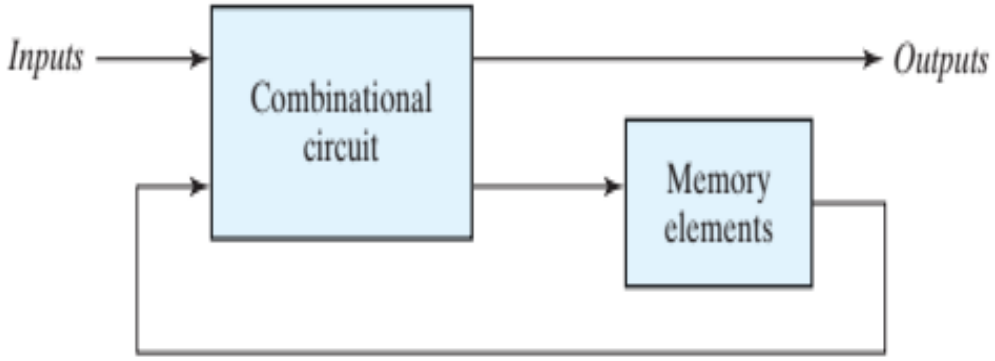
PROGRAMME: BCA

CENTRAL UNIVERSITY OF ODISHA

KORAPUT

SEQUENTIAL CIRCUIT

- Combinational Circuit—their output depends only and immediately on their inputs—they have no memory, i.e., dependence on past values of their inputs.
- Sequential circuits, however, act as storage elements and have memory.
- They can store, retain, and then retrieve information when needed at a later time.
-



- 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.
- The sequential circuit receives binary information from external inputs that, together with the present state of the storage elements, determine the binary value of the outputs.
- 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.

TYPES OF SEQUENTIAL CIRCUIT

- Based on **function of the timing signals**, sequential circuit are two types:
- i) Synchronous sequential circuit
- ii) Asynchronous sequential circuit

Synchronous

- The feedback to the input for next output generation is governed by clock signals.
- The memory unit which is being get used for governance is clocked flip flop.
- The states of Synchronous sequential circuits are always predictable and thus reliable.

Asynchronous

- Not governed by clock signals.
- Unclocked flip flop or time delay is used as memory element.
- There are chances for the Asynchronous circuits to enter into a wrong state because of the time difference between the arrivals of inputs. This is called as race condition.

Synchronous

- Output behaviour depends on the input at discrete time.
- Inputs and outputs are considered at discrete time instants.

Asynchronous

- Output behaviour depends on the sequence in which the input changes.
- Inputs and outputs signals are defined at every value of time.

Synchronous

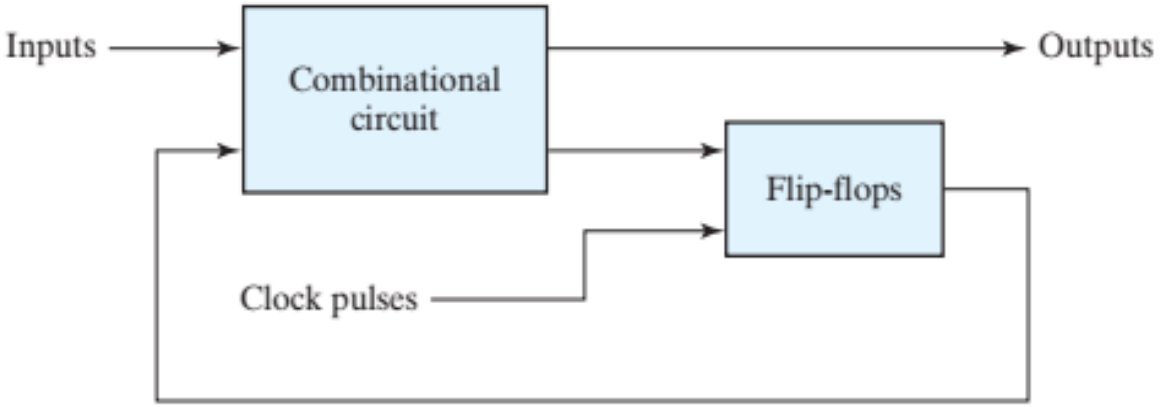
- Easy to describe, analyze and design.
- Due to the propagation delay of clock signal in reaching all elements of the circuit the Synchronous sequential circuits are slower in its operation speed.
- Ex: counters, shift registers, memory units.

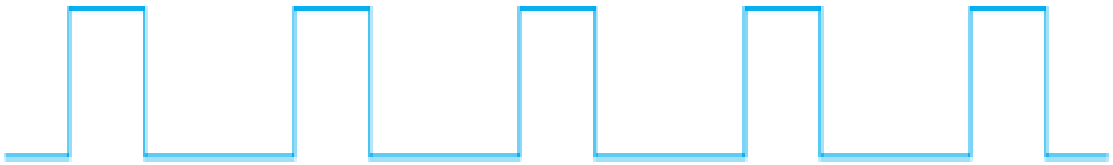
Asynchronous

- Difficult to describe, analyze and design.
- Since there is no clock signal delay, these are fast compared to the Synchronous Sequential Circuits.
- Ex: simple microprocessors, digital signal processing units and in communication systems for email applications, internet access and networking.

Synchronous Clocked Sequential Circuit

•



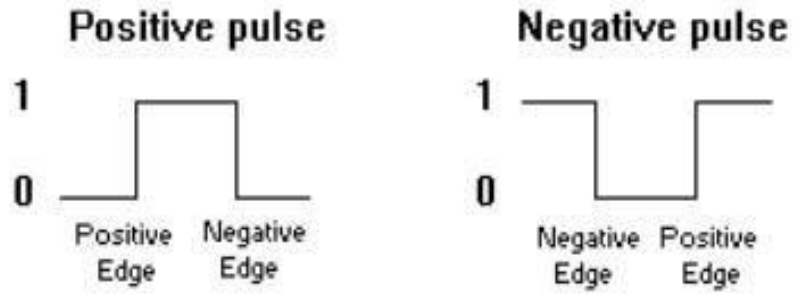


Timing diagram of clock pulses

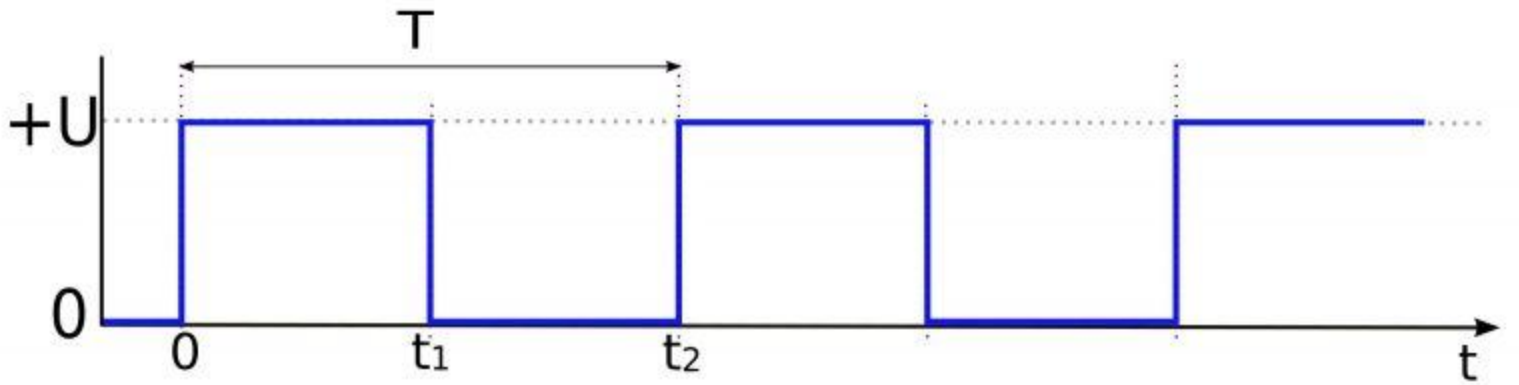
- The storage elements (memory) used in clocked sequential circuits are called *flipflops.*
- A flip-flop is a binary storage device capable of storing one bit of information.
- In a stable state, the output of a flip-flop is either 0 or 1.
- A sequential circuit may use many flip-flops to store as many bits as necessary.
- The *outputs* are formed by a combinational logic function of the inputs to the circuit or the values stored in the flip-flops (or both).
- The value that is stored in a flip-flop when the clock pulse occurs is also determined by the inputs to the circuit or the values presently stored in the flip-flop (or both).
- The new value is stored (i.e., the flip-flop is updated) when a pulse of the clock signal occurs.

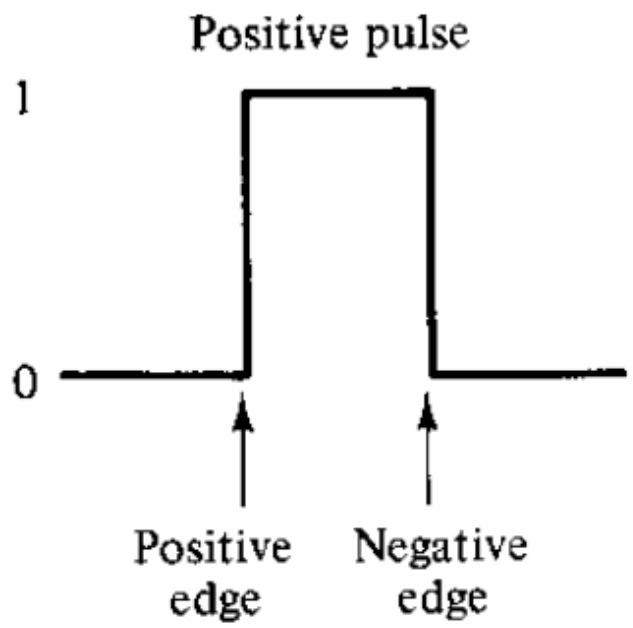
- Prior to the occurrence of the clock pulse, the combinational logic forming the next value of the flip-flop must have reached a stable value.
- If the clock (synchronizing) pulses arrive at a regular interval, the combinational logic must respond to a change in the state of the flip-flop in time to be updated before the next pulse arrives.
- Propagation delays play an important role in determining the minimum interval between clock pulses that will allow the circuit to operate correctly.
- A change in state of the flip-flops is initiated only by a clock pulse transition.

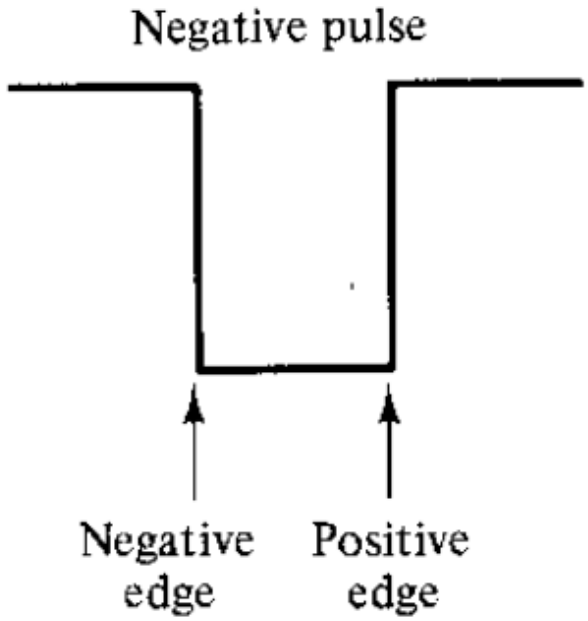
- A storage element in a digital circuit can maintain a binary state indefinitely (as long as power is delivered to the circuit), until directed by an input signal to switch states.
- The major differences among various types of storage elements are in the number of inputs they possess and in the manner in which the inputs affect the binary state.
- Storage elements that operate with signal levels (rather than signal transitions) are referred to as latches; those controlled by a clock transition are flip-flops.
- Latches are said to be level sensitive devices; flip-flops are edge-sensitive devices.

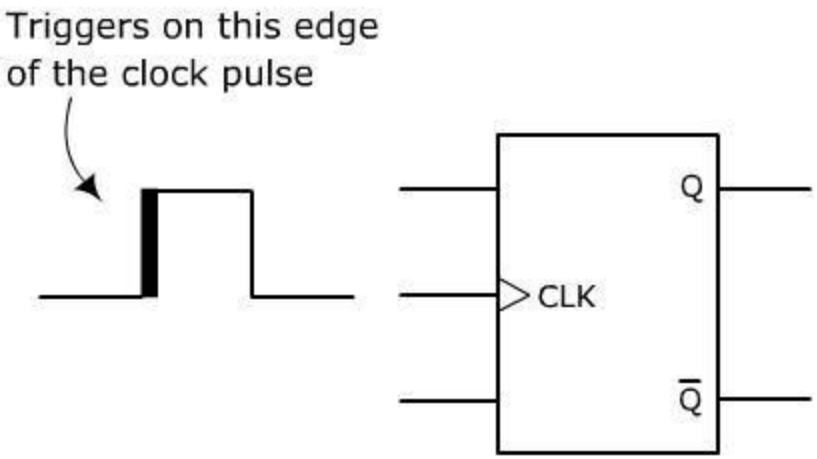


Definition of clock pulse transition

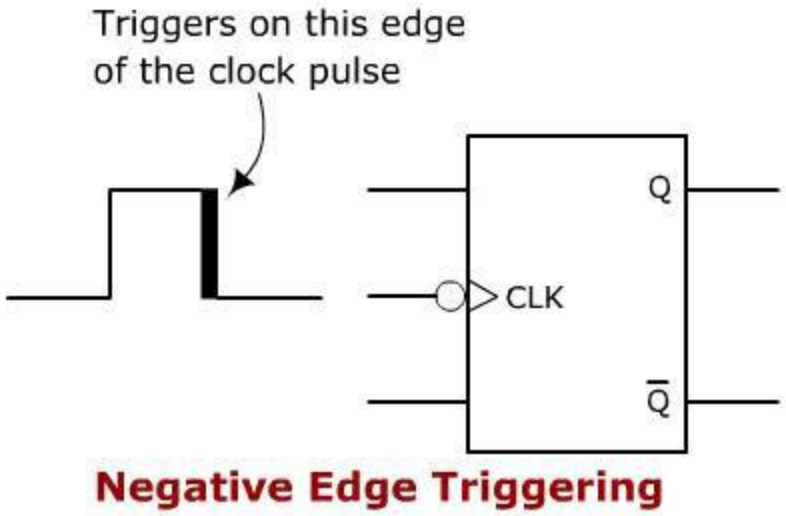



Positive pulse
1
0
Positive edge
Negative edge

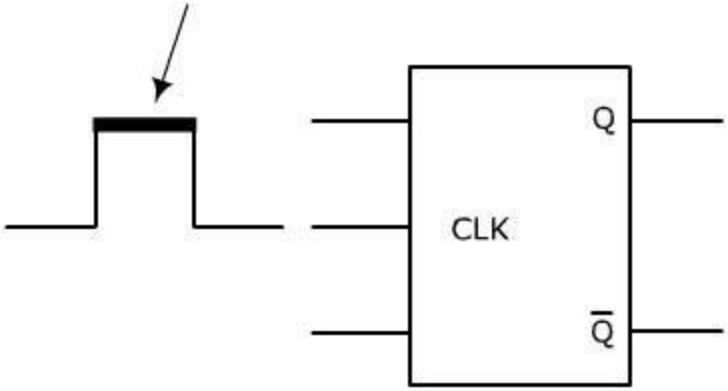

Negative pulse
Negative edge
Positive edge



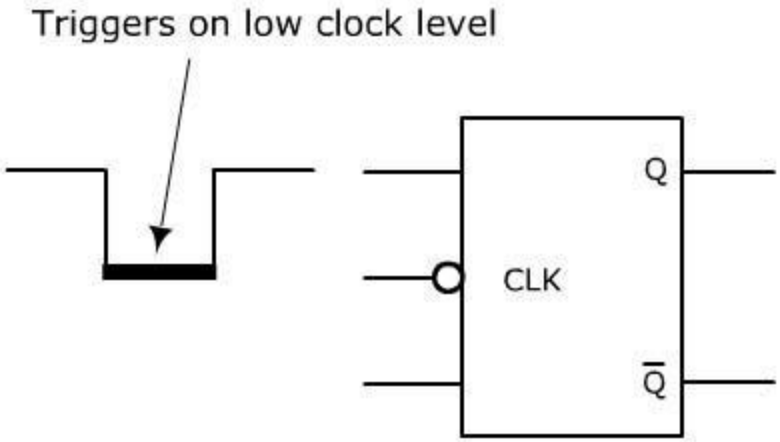
Positive Edge Triggering



Negative Edge Triggering



High Level Triggering



Low Level Triggering

Triggering

- **Triggering:** This means making a circuit active.
- Making a circuit active means allowing the circuit to take input and give output.
- There are basically two types of triggering.
- The triggering is given in form of a clock pulse or gating signal.
- Depending upon the type of triggering mechanism used, the circuit will become active at specific states of the clock pulse.
- 1. **Level Triggering:** In the sequential circuit, if the output changes during the high voltage period or low voltage period, it is called level triggering.
- In level triggering the circuit will become active when the gating or clock pulse is on a particular level.
- This level is decided by the designer. We can have a negative level triggering in which the circuit is active when the clock signal is low or a positive level triggering in which the circuit is active when the clock signal is high.

- 2. **Edge Triggering:** In a sequential circuit, if the output changes when the signal transits from a high level to a low level or from a low level to a high level, we call it edge triggering.
- In edge triggering the circuit becomes active at negative or positive edge of the clock signal.
- For example if the circuit is positive edge triggered, it will take input at exactly the time in which the clock signal goes from low to high.
- Similarly input is taken at exactly the time in which the clock signal goes from high to low in negative edge triggering.
- But keep in mind after the the input, it can be processed in all the time till the next input is taken.

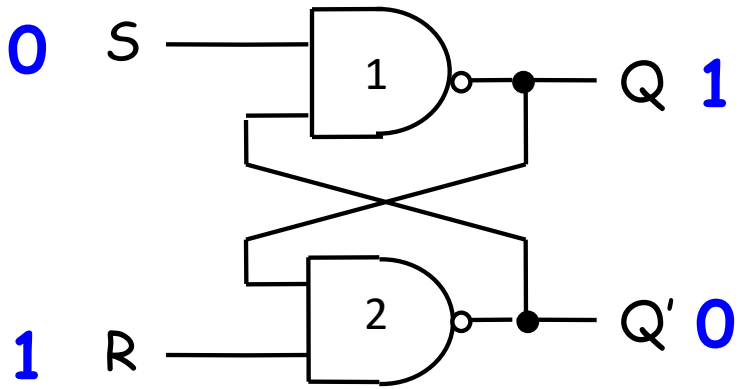
Latch

- Latches are the basic circuits from which all flip-flops are constructed.
- Although latches are useful for storing binary information and for the design of asynchronous sequential circuits, they are not practical for use as storage elements in synchronous sequential circuits.

Flip-Flop

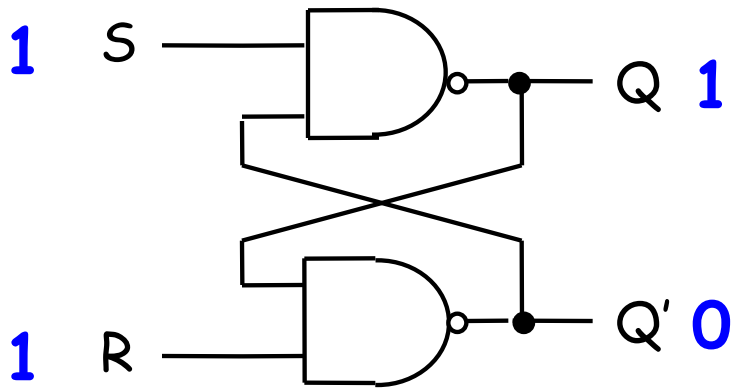
- A flip-flop circuit can maintain a binary state indefinitely(as long as power is deliver to the circuit) until directed by an input signal to switch states.
- The major differences among various types of flip-flops are in the number of inputs they possess and in the manner in which the inputs affect the binary state.
- A flip-flop can be constructed from two **NAND** gates or **NOR** gates.
- Each circuit forms a basic flip-flop upon which other more complicated types can be built.
- The cross-coupled connection from the output of one gate to the input of the other gate constitutes a feedback path.
- For this reason, the circuits are classified as asynchronous sequential circuits.
- Each flip-flop has **two outputs Q and Q',** and **two inputs, Set(S) and Reset(R).**
- This type of flip-flop is sometimes called **a direct-coupled RS flip-flop, or SR latch.**

SR Latch with NAND Gate



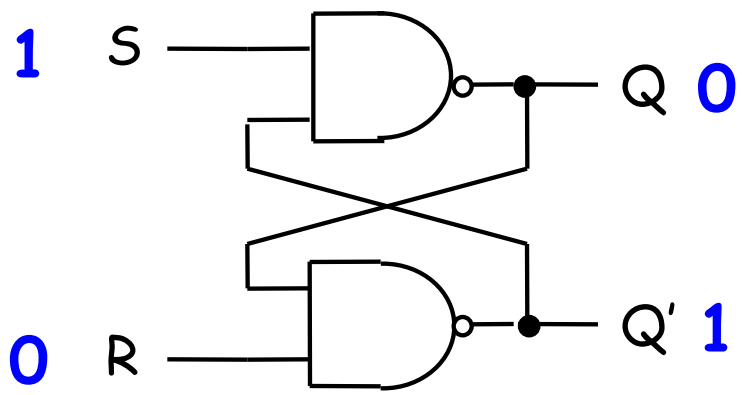
X	Y	NAND
0	0	1
0	1	1
1	0	1
1	1	0

S	R	Q	Q'	
0	0			
0	1	1	0	SET
1	0			
1	1			



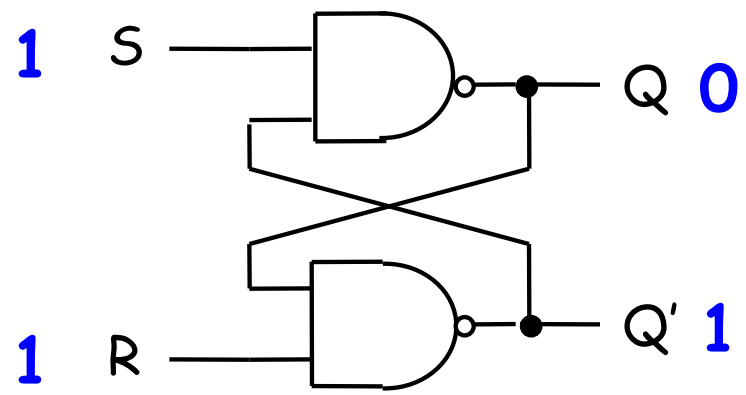
X	Y	NAND
0	0	1
0	1	1
1	0	1
1	1	0

S	R	Q	Q'	
0	0			
0	1	1	0	SET
1	0			
1	1	1	0	HOLD



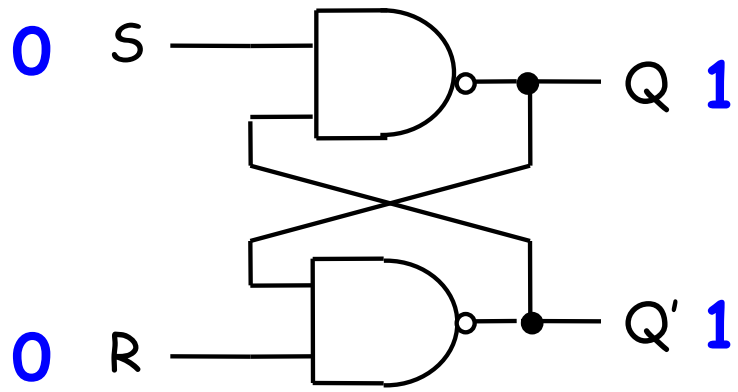
X	Y	NAND
0	0	1
0	1	1
1	0	1
1	1	0

S	R	Q	Q’	
0	0			
0	1	1	0	SET
1	0	0	1	RESET
1	1	1	0	HOLD After S=0, R=1



X	Y	NAND
0	0	1
0	1	1
1	0	1
1	1	0

S	R	Q	Q'	
0	0			
0	1	1	0	SET
1	0	0	1	RESET
1	1	1	0	HOLD After S=0, R=1
		0	1	HOLD After S=1, R=0



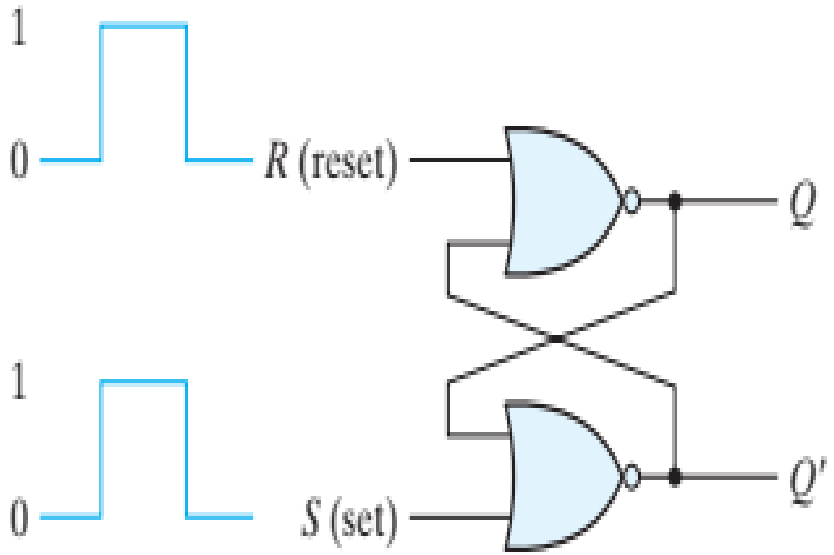
X	Y	NAND
0	0	1
0	1	1
1	0	1
1	1	0

S	R	Q	Q’	
0	0	1	1	Forbidden
0	1	1	0	SET
1	0	0	1	RESET
1	1	1	0	HOLD After S=0, R=1
		0	1	HOLD After S=1, R=0

- It operates with both inputs normally at 1, unless the state of the latch has to be changed.
- The application of 0 to the *S* input causes output *Q* to go to 1, putting the latch in the set state.
- When the *S* input goes back to 1, the circuit remains in the set state.
- After both inputs go back to 1, we are allowed to change the state of the latch by placing a 0 in the *R* input.
- This action causes the circuit to go to the reset state and stay there even after both inputs return to 1.
- The condition that is forbidden for the NAND latch is both inputs being equal to 0 at the same time, an input combination that should be avoided.

SR Latch with NOR Gate

X	Y	NOR
0	0	1
0	1	0
1	0	0
1	1	0



(a) Logic diagram

S	R	Q	Q'	
1	0	1	0	
0	0	1	0	(after $S = 1, R = 0$)
0	1	0	1	
0	0	0	1	(after $S = 0, R = 1$)
1	1	0	0	(forbidden)

(b) Function table

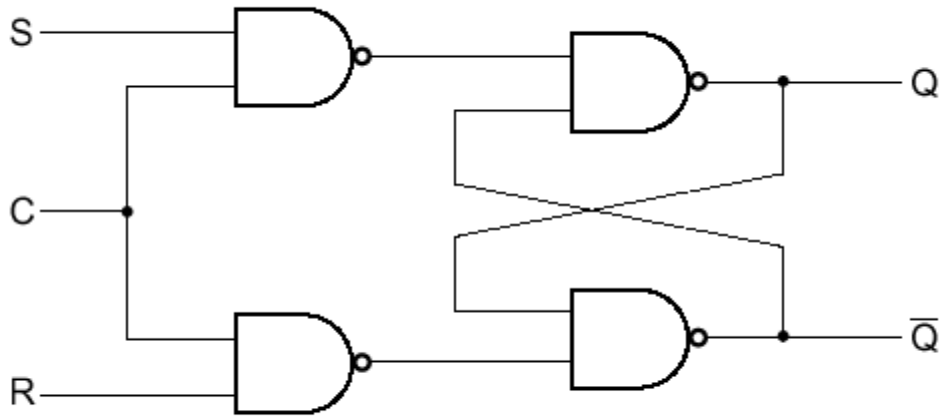
SR Latch with NOR Gate

- The output of a NOR gate is 0 if any input is 1, and that the output is 1 only when all inputs are 0.
- As a starting point, assume that the set input is 1 and the reset input is 0.
- Since gate 2 has an input of 1, its output Q' must be 0, which puts both inputs of gate 1 at 0, so that output Q is 1.
- When the set input is returned to 0, the outputs remain the same, because output Q remains a 1, leaving one input of gate 2 at 1.
- That causes output Q' to stay at 0, which leaves both inputs of gate number 1 at 0, so that output Q is a 1.
- In the same manner, it is possible to show that a 1 in the reset input changes output Q to 0 and Q' to 1.
- When the reset input returns to 0, the output do not change.

SR Latch with NOR Gate

- When a 1 is applied to both the set and the reset input, both Q and Q’ output go to 0.
- This condition violates the fact that output Q and Q’ are complement of each other.
- A flip-flop has two useful states.
- When Q=1 and Q’=0, it is in the set state(or 1-state).
- When Q=0 and Q’=1, it is in the clear state(or 0-state).
- The outputs Q and Q’ are complement of each other and are referred to as the normal and complement outputs, respectively.
- The binary state of the flip-flop is taken to be the value of the normal output.

RS Flip-Flop / SR Latch With Control Unit



(a) Logic diagram

C	S	R	Next state of Q
0	X	X	No change
1	0	0	No change
1	0	1	Q = 0; Reset state
1	1	0	Q = 1; Set state
1	1	1	Undefined

(b) Function table

RS Flip-Flop / SR Latch With Control Unit

- It consists of a basic flip-flop circuit and two additional NAND gates.
- The pulse input acts as an enable signal for the other two inputs.
- The outputs of NAND gates 3 and 4 stay at the logic 1 level as long as the CP input remains at 0.
- This is the quiescent condition for the basic flip-flop.
- When the pulse input goes to 1, information from the S or R input is allowed to reach the output.
- The set state is reached with S=1, R=0, and CP=1.
- This causes the output of gate 3 to go to 0, the output of gate 4 to remains at 1, and the output of the flip-flop at Q to go to 1.
- To change to the reset state, the inputs must be S=0, R=1, and CP=1.

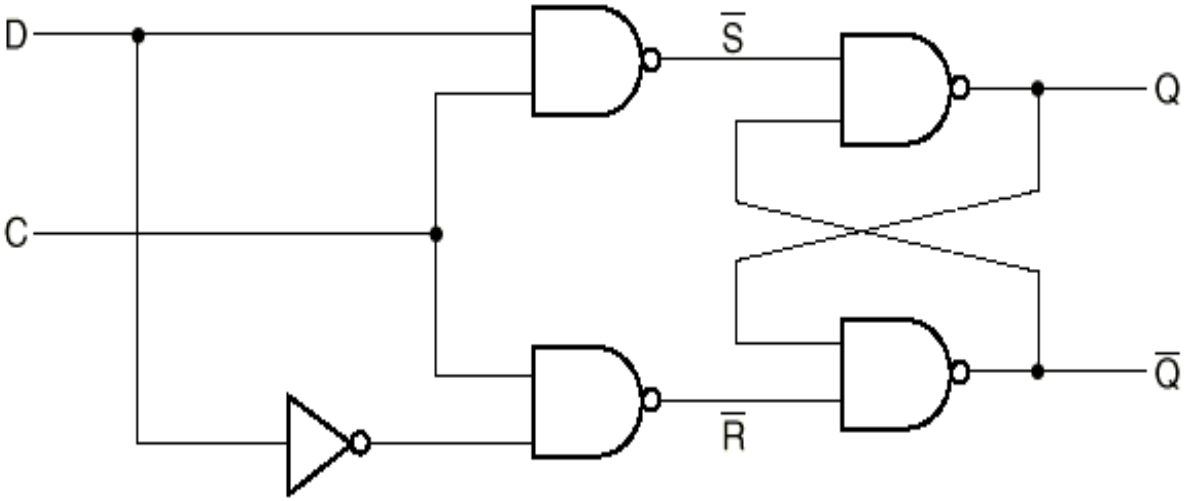
RS Flip-Flop / SR Latch With Control Unit

- In either case, when CP returns to 0, the circuit remains in its previous state.
- When CP=1 and both the S and R inputs are equal to 0, the state of the circuit does not change.
- An indeterminate condition occurs when CP=1 and both S and R are equal to 1.
- This condition places 0's in the outputs of gate 3 and 4 and 1's in both outputs Q and Q'.
- When the CP input goes back to 0(when S and R are maintained at 1), it is not possible to determine the next state, as it depends on whether the output of gate 3 or gate 4 goes to 1 first.

RS Flip-Flop / SR Latch With Control Unit

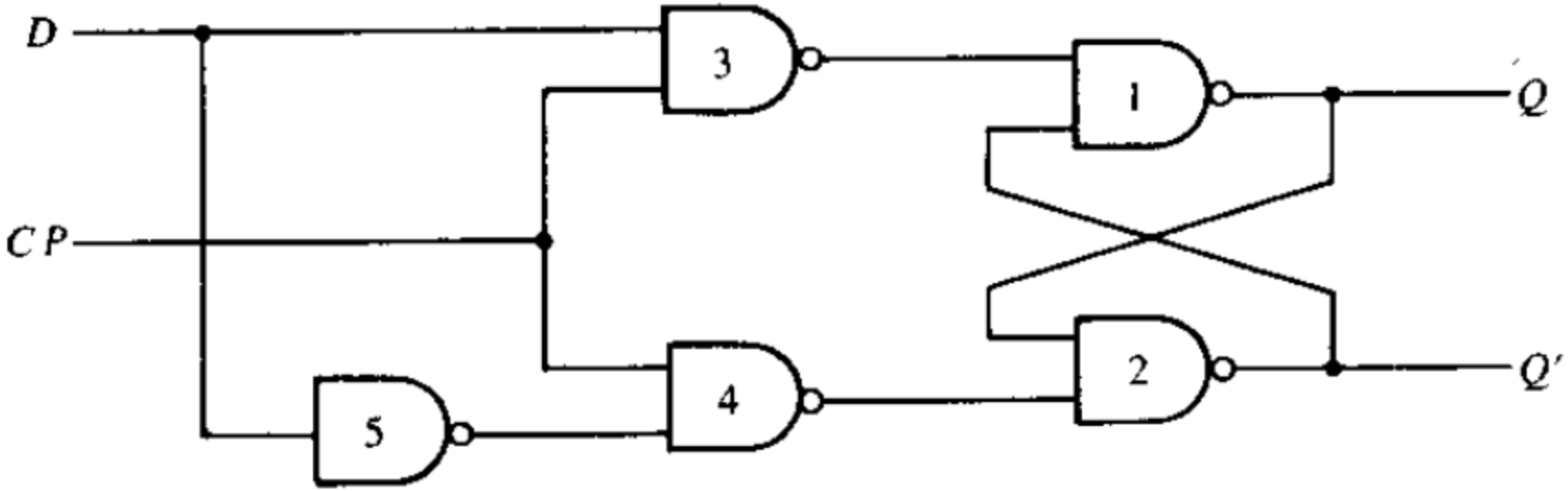
- The characteristic table shows the operation of the flip-flop in tabular form.
- Q is an abbreviation of Q(t) and stands for the binary state of the flip-flop before the application of a clock pulse, referred to as the present state.
- The S and R columns give the possible values of the inputs, and Q(t+1) is the state of the flip-flop after the application of a single pulse, referred to as the next state.
- The CP input is not included in the characteristic table.
- The two indeterminate states are marked with don't care X's in the map, since they may result in either 1 or 0.
- However, the relation SR=0 must be included as part of the characteristics equation to specify that both S and R can not equal to 1 simultaneously.

D Latch (Transparent Latch) / D Flip-Flop



C	D	Next state of Q
0	X	No change
1	0	Q = 0; Reset state
1	1	Q = 1; Set state

D Latch (Transparent Latch) / D Flip-Flop



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

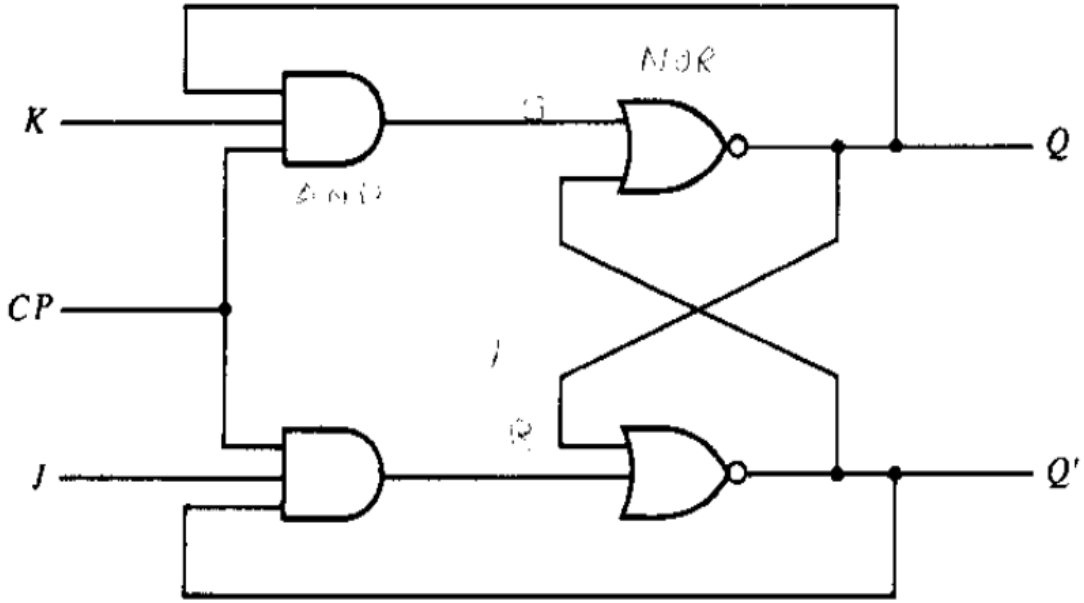
D Latch (Transparent Latch) / D Flip-Flop

- The undesirable condition of the indeterminate state in the RS flip-flop is that inputs S and R are equal to 1 at the same time.
- The D flip-flop has only two inputs: D and CP.
- The D input goes directly to S input and its complement is applied to the R input.
- As long as the pulse input is at 0, the outputs of gates 3 and 4 are at the 1 level and the circuit cannot change state regardless of the value of D.
- The D input is sampled when CP=1.
- If D is 1, the Q output goes to 1, placing the circuit in the set state.
- If D is 0, output Q goes to 0 and the circuit switches to the clear state.

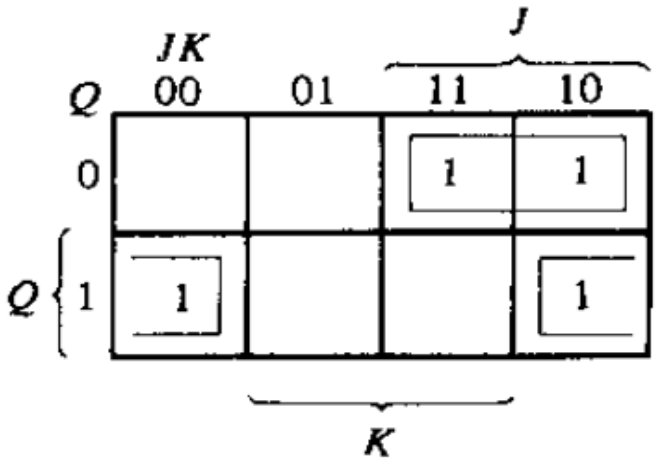
D Latch (Transparent Latch) / D Flip-Flop

- The D flip-flop receives the designation from its ability to hold data into its internal storage.
- This type of flip-flop is sometimes called a gated D-latch.
- The CP input is often give the designation G(for gate) to indicate that this input enables the gated latch to make possible data entry into the circuit.
- The binary information present at the data input of the D flip-flop is transferred to the Q output when the CP input is enabled.
- The output follows the data input as long as the pulse remains in its 1 state.
- When the pulse goes to 0, the binary information that was present at the data input at the time the pulse transition occurred is retained at the Q output until the pulse input is enabled again.

JK Flip-Flop



Q	J	K	Q (t + 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



$$Q(t+1) = JQ' + K'Q$$

JK Flip-Flop

- A JK flip-flop is a refinement of the RS flip-flop in that the indeterminate state of the RS type is defined in the JK type.
- Inputs J and K behave like inputs S and R to set and clear the flip-flop, respectively.
- The input marked J is for set and the input marked K is for reset.
- When both inputs J and K are equal to 1, the flip-flop switches to its complement state, that is, if Q=1, it switches to Q=0, and vice versa.

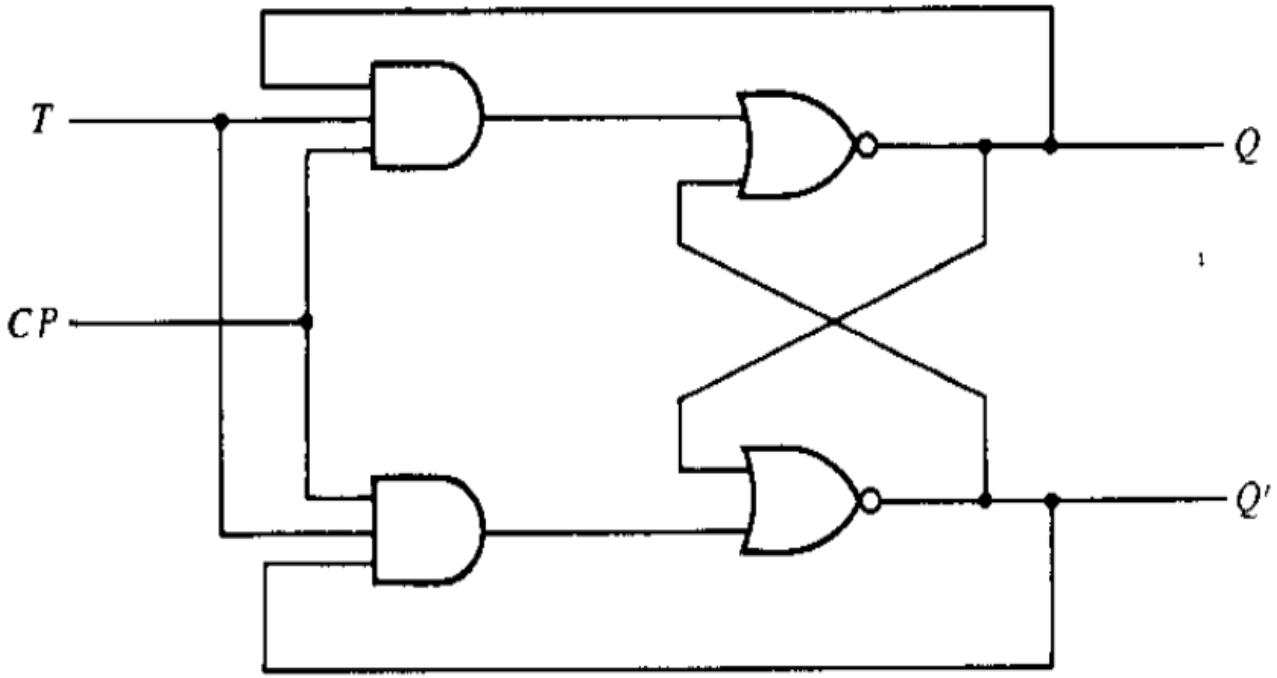
JK Flip-Flop

- Output Q is ANDed with K and CP inputs so that the flip-flop is cleared during a clock pulse only if Q was previously 1.
- Similarly, output Q' is ANDed with J and CP inputs so that the flip-flop is set with a clock pulse only when Q' was previously 1.
- When both J and K are 1, the input pulse is transmitted through one AND gate only; the one whose input is connected to the flip-flop output that is presently equal to 1.
- Thus, if Q=1, the output of the upper AND gate becomes 1 upon application of the clock pulse, and the flip-flop is cleared.
- If Q'=1, the output of the lower AND gate becomes 1 and the flip-flop is set.
- In either case, the output state of the flip-flop is complemented.

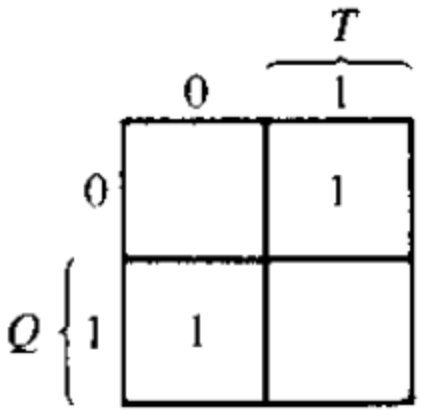
JK Flip-Flop

- It is very important to realize that because of the feedback connection in the JK flip-flop, a CP pulse that remains in the 1 state while both J and K are equal to 1 will cause the output to complement again and repeat complementing until the pulse goes back to 0.
- To avoid this undesirable operation, the clock pulse must have a time duration that is shorter than the propagation delay time of the flip-flop.

T Flip-Flop



Q	T	$Q(t+1)$
0	0	0
0	1	1
1	0	1
1	1	0

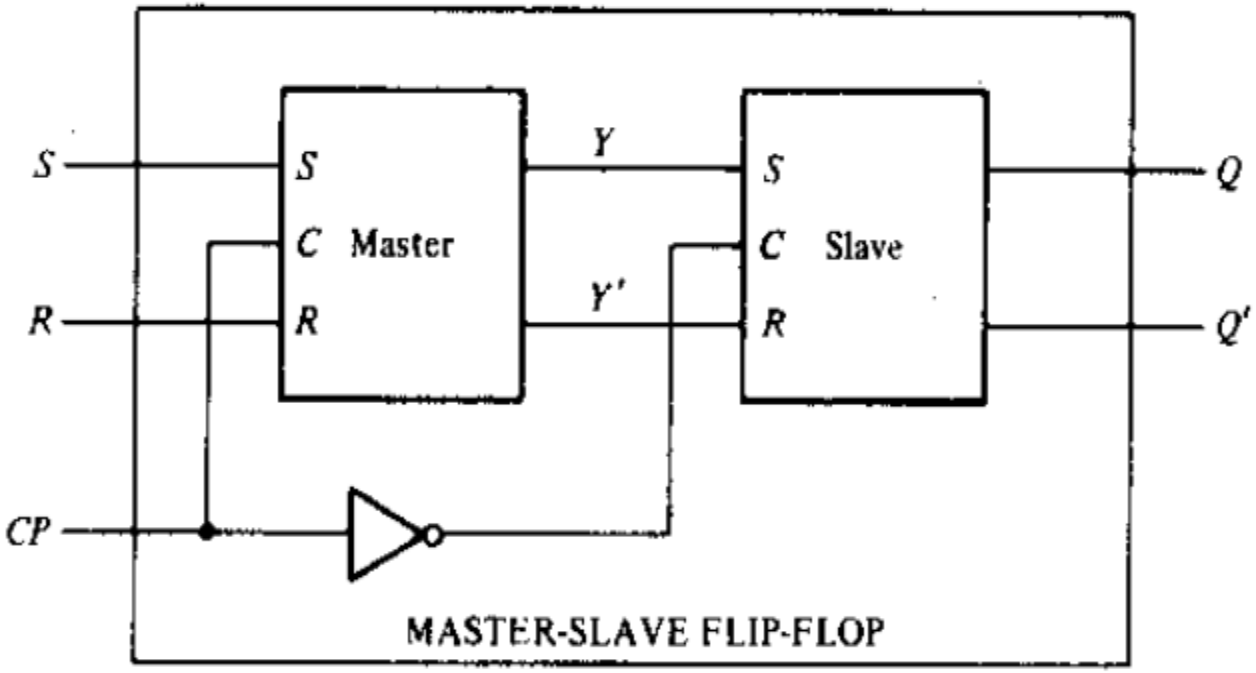


$$Q(t+1) = TQ' + T'Q$$

T Flip-Flop

- The T flip-flop is a single-input version of the JK flip-flop.
- The designation T comes from the ability of the flip-flop to “toggle”, or complement, its state.
- Regardless of the present state, the flip-flop complements its output when the clock pulse occurs while input T is 1.

Master-Slave Flip-Flop



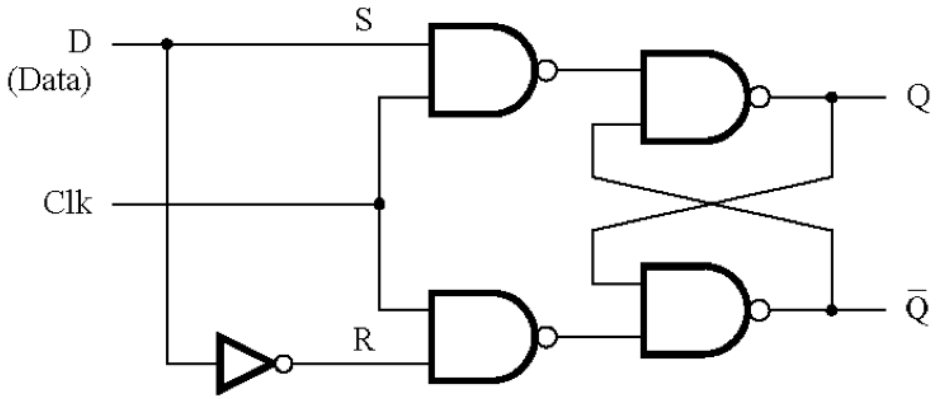
Master-Slave Flip-Flop

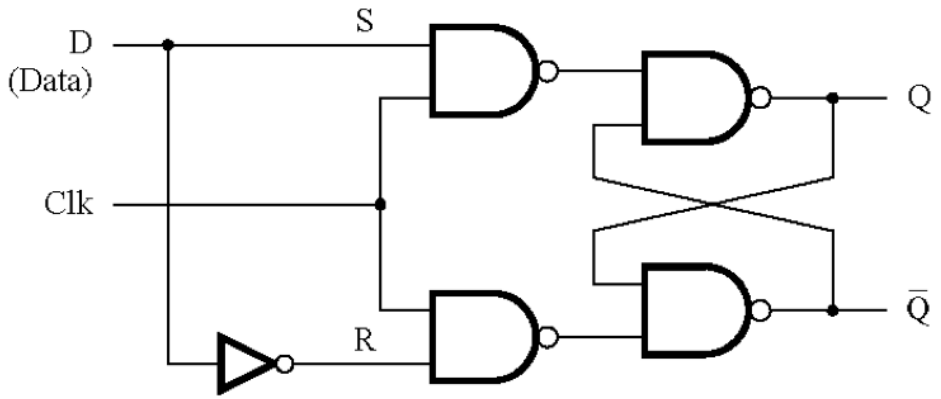
- A master-slave flip-flop is constructed from two separate flip-flops.
- One circuit serves as a master and the other as a slave, and the overall circuit is referred to as a master-slave flip-flop.
- It consists of a master flip-flop, a slave flip-flop, and an inverter.
- When clock pulse CP is 0, the output of the inverter is 1.
- Since the clock input of the slave is 1, the flip-flop is enabled and output Q is equal to Y, while Q' is equal to Y'.
- The master flip-flop is disabled because CP=0.
- When the pulse becomes 1, the information then at the external R and S inputs is transmitted to the master flip-flop.
- The slave flip-flop, however, is isolated as long as the pulse is at its 1 level, because the output of the inverter is 0.
- When the pulse returns to 0, the master flip-flop is isolated, which prevents the external inputs from affecting it.
- The slave flip-flop then goes to the same state as the master flip-flop.

Constructing a Master-Slave D Flip-Flop From Two D Latches

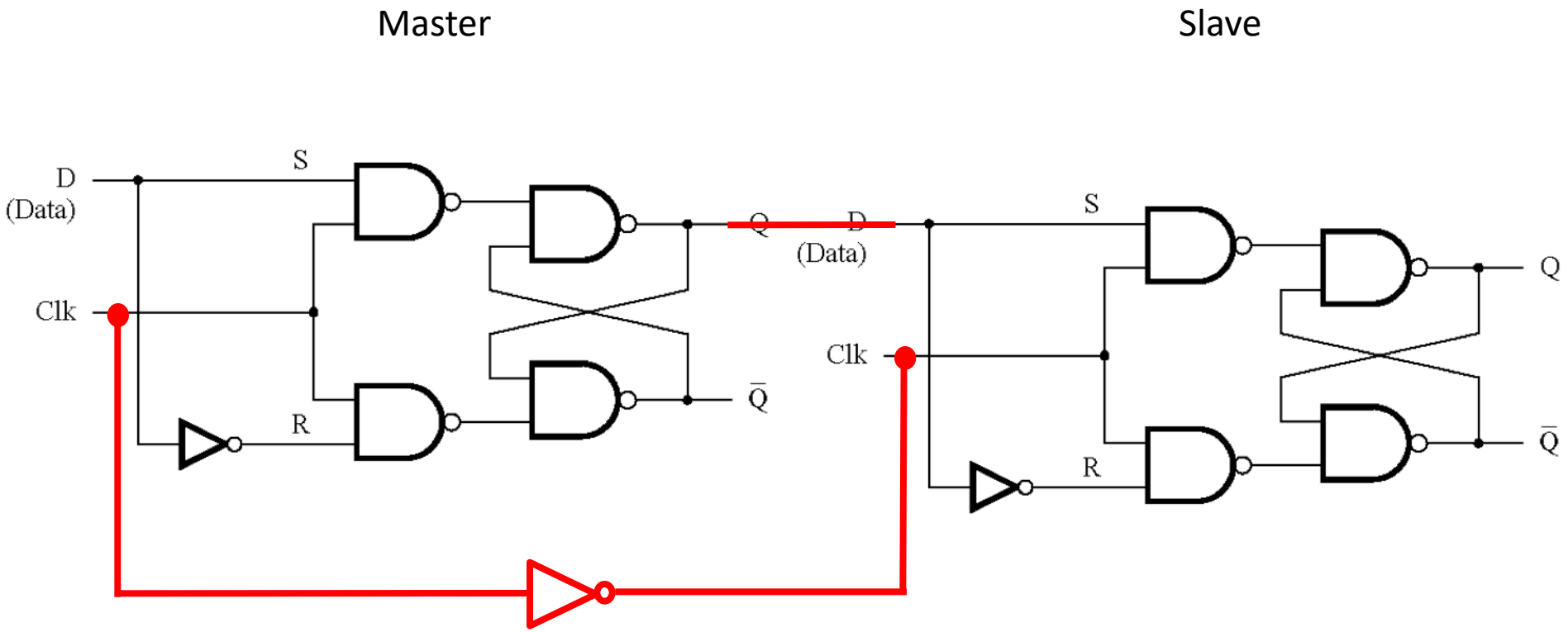
Master

Slave

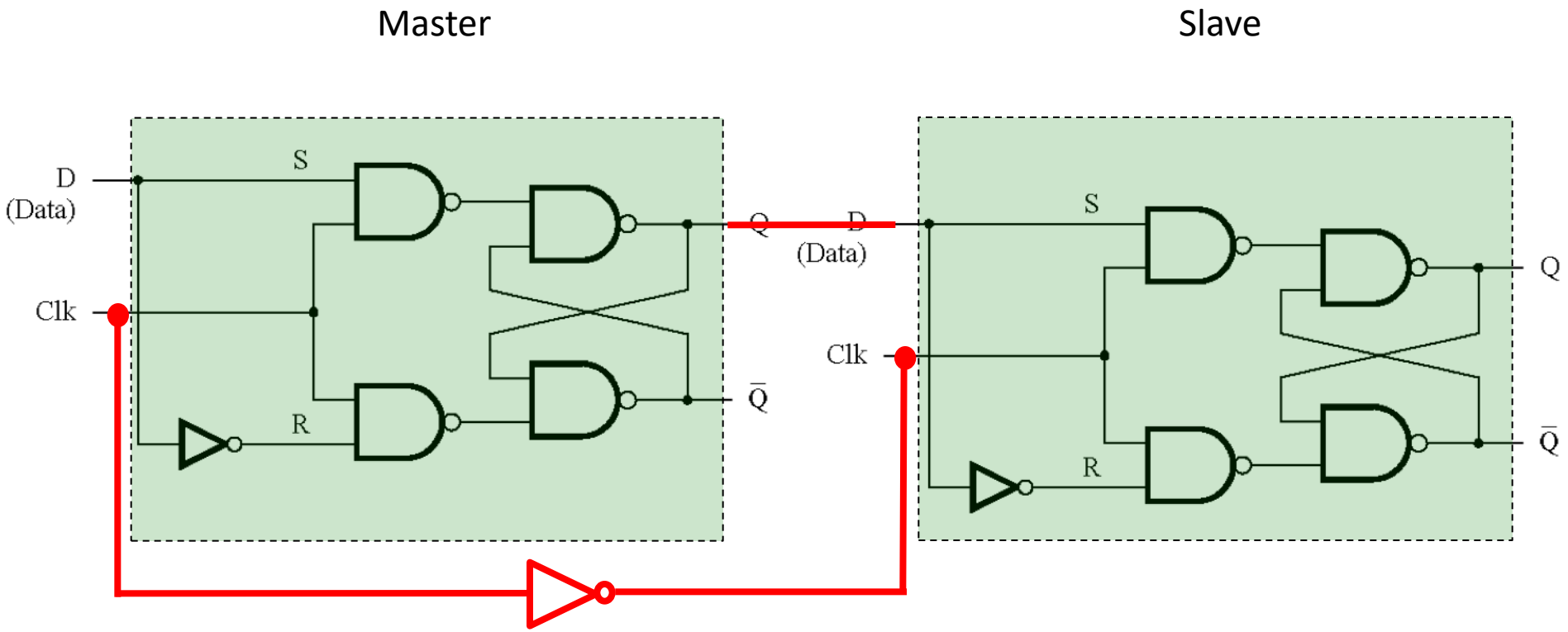




Constructing a Master-Slave D Flip-Flop From Two D Latches



Constructing a Master-Slave D Flip-Flop From Two D Latches



Constructing a Master-Slave D Flip-Flop From Two D Latches