

# Flip-Flops and Related Devices

**Presented By**

Firoz Mahmud

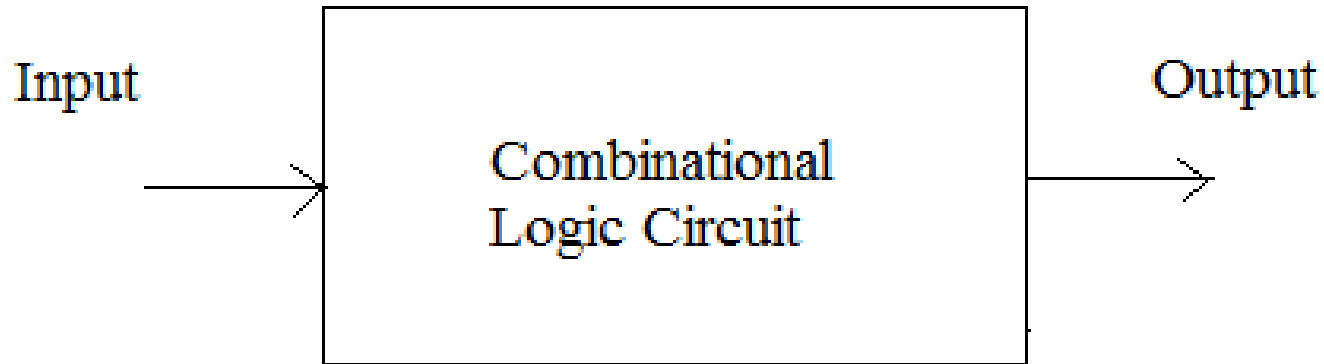
Assistant Professor

Dept. of Computer Science & Engineering

Rajshahi University of Engineering & Technology

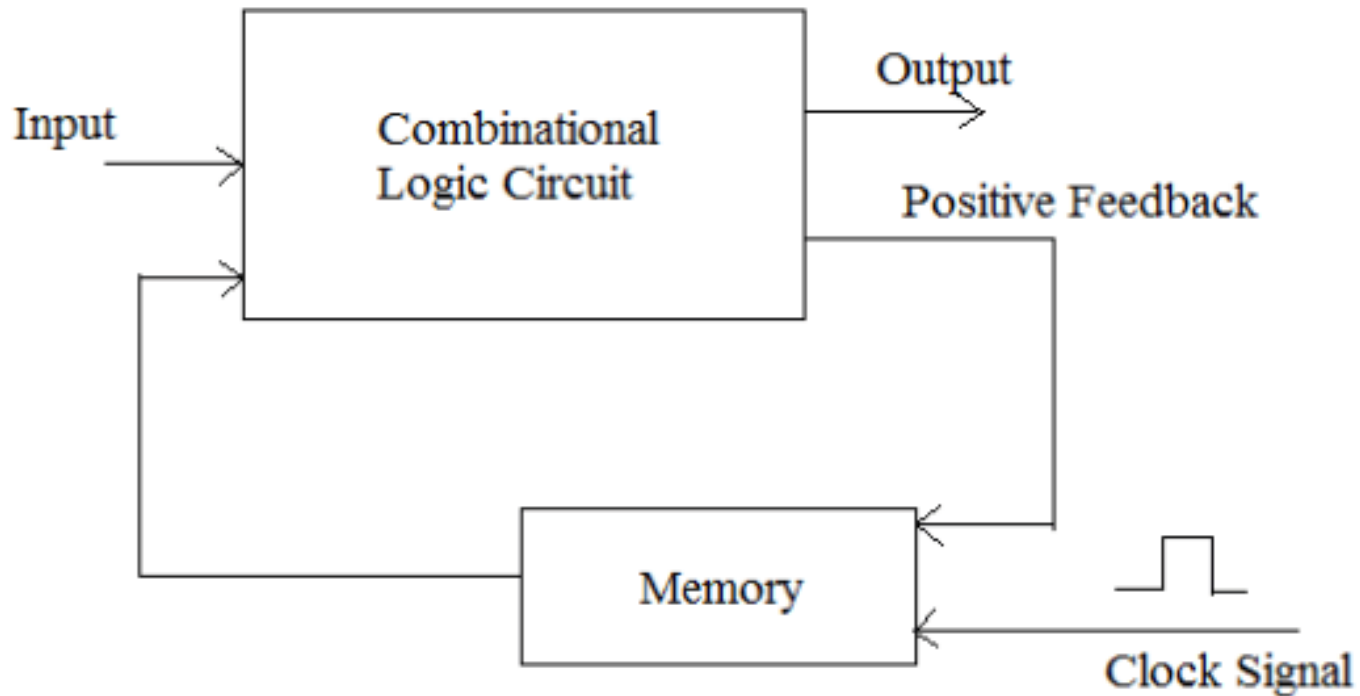
## Chapter 5 Combinational Circuit Vs. Sequential Logic

- Combinational logic (sometimes also referred to as **time-independent** logic) is a type of digital logic which is implemented by Boolean circuits, where the **output** is a pure function of the **present** input **only**.



## Chapter 5 Combinational Circuit Vs. Sequential Logic

- Sequential logic is a type of logic circuit whose **output** depends **not** only on the **present** value of its input signals but on the **sequence** of past inputs.

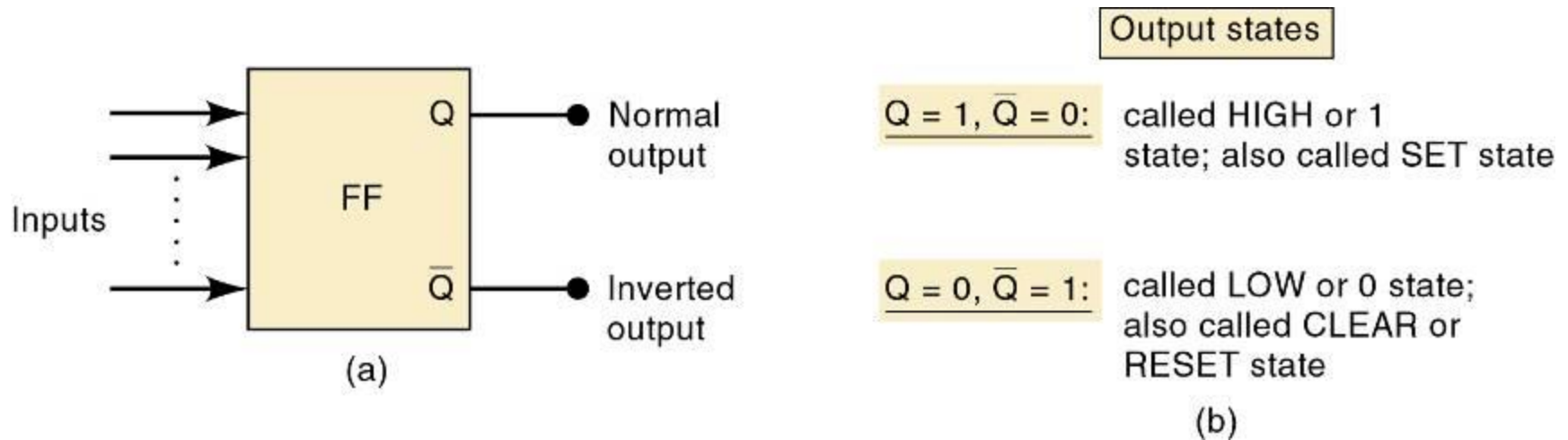


## Chapter 5 Combinational Circuit Vs. Sequential Logic

| Combinational Circuit  | Sequential Logic   |
|--|--|
| Output is a function of the <b>present</b> inputs.               | Output is a function of <b>clock</b> , <b>present</b> inputs and the <b>previous</b> states of the system.                                 |
| Do not have the ability to <b>store</b> data.                    | Have <b>memory</b> to <b>store</b> the <b>present</b> states that is sent as <b>control</b> input (enable) for the next <b>operation</b> . |
| It does not require any <b>feedback</b> .                        | It involves <b>feedback</b> from output to input.  |
| Used mainly for <b>Arithmetic</b> and <b>Boolean</b> operations. | Used for <b>storing</b> data (and hence used in <b>RAM</b> ).  |
| <b>Logic</b> gates are the elementary building blocks.           | <b>Flip flops</b> (binary storage device) are the elementary building unit.  |

## Chapter 5 Introduction

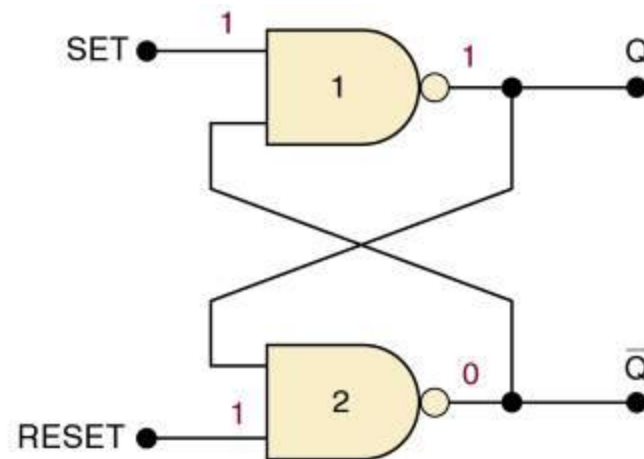
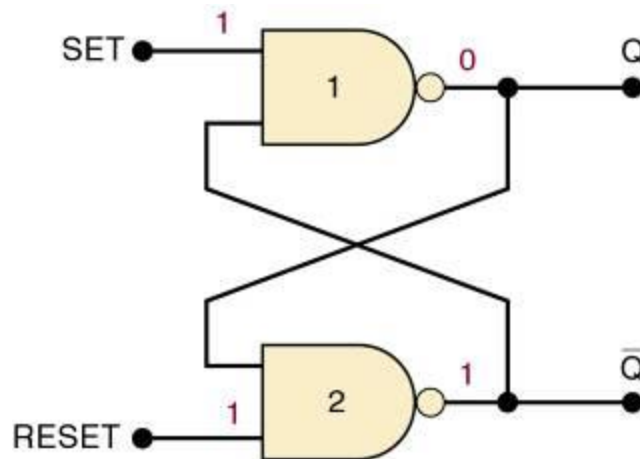
- The most important memory element is the **flip-flop** (FF)—made up of an assembly of logic gates.



The flip-flop is known by other names, including *latch* and *bistable multivibrator*.

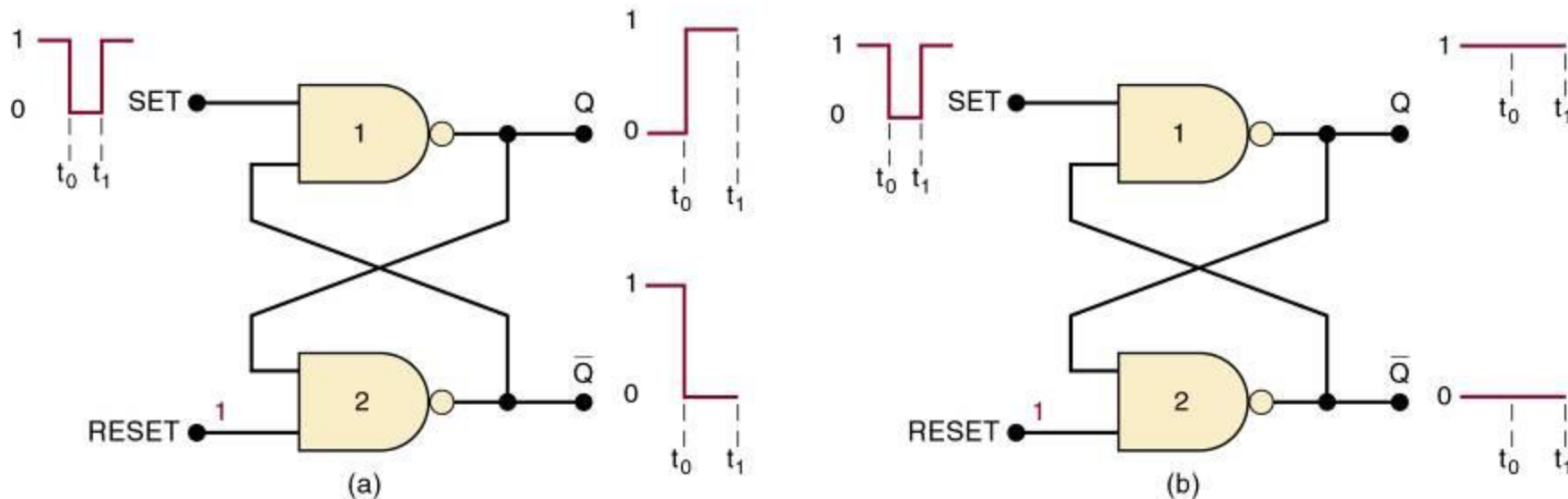
## 5-1 NAND Gate Latch

- The **NAND** gate latch or simply latch is a basic FF.
  - Inputs are **SET** and **CLEAR** (*RESET*).
- Inputs are **active-LOW**—output will change when the input is pulsed **LOW**.
  - When the latch is set:  $Q = 1$  and  $\bar{Q} = 0$
  - When the latch is clear or reset:  $Q = 0$  and  $\bar{Q} = 1$



## 5-1 NAND Gate Latch – Setting the Latch (FF)

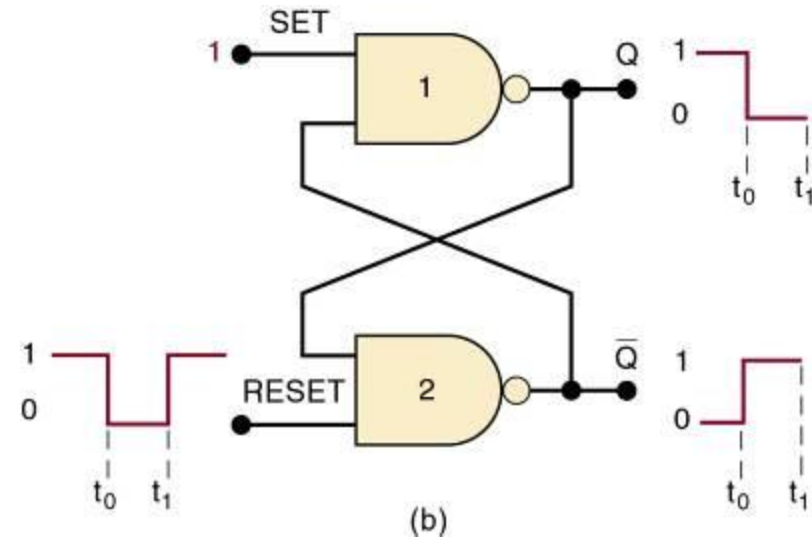
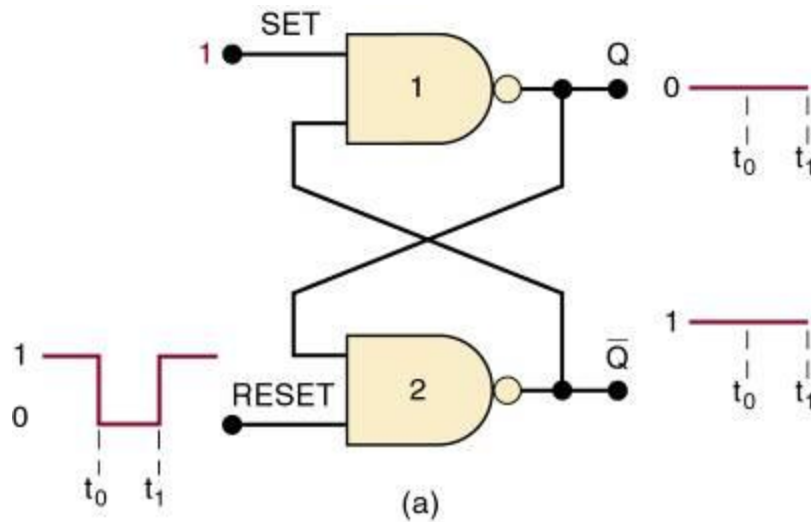
- Pulsing the SET input to the 0 state...
  - (a)  $Q = 0$  prior to SET pulse.
  - (b)  $Q = 1$  prior to SET pulse.



**In both cases, Q ends up HIGH.**

## 5-1 NAND Gate Latch – Resetting the Latch (FF)

- Pulsing RESET LOW when...
  - (a)  $Q = 0$  prior to the RESET pulse.
  - (b)  $Q = 1$  prior to the RESET pulse.

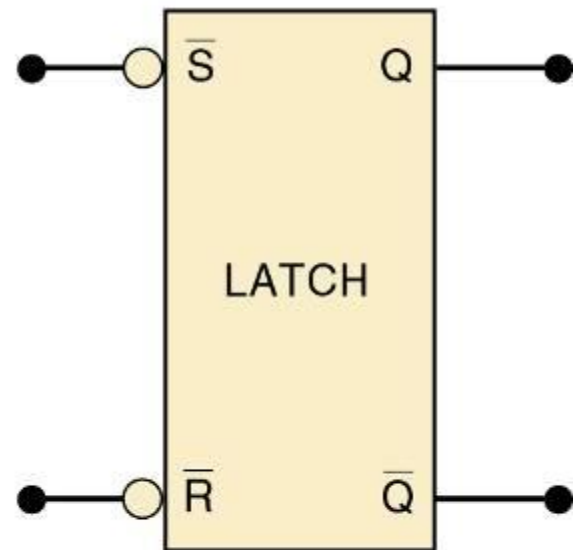
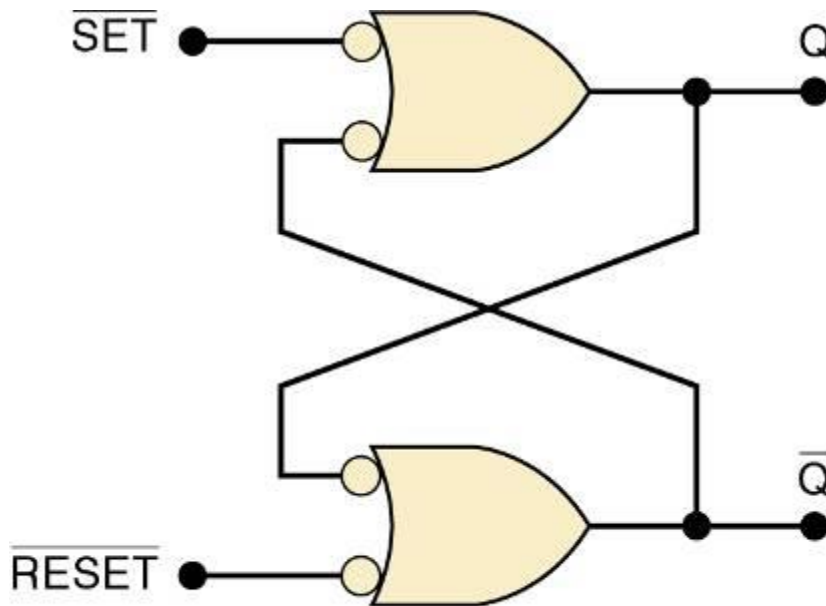


**In each case, Q ends up LOW.**



## 5-1 NAND Gate Latch – Alternate Representations

**NAND** latch equivalent representations  
and simplified block diagram.

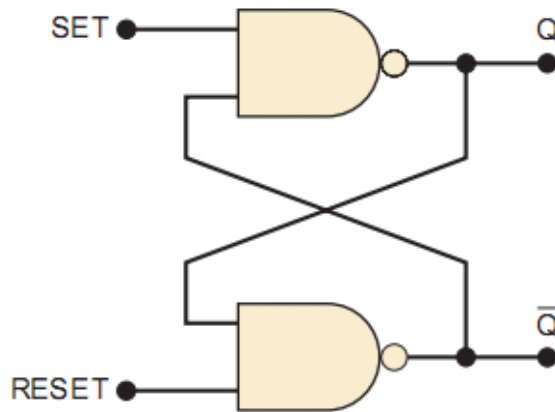


## 5-1 NAND Gate Latch - Summary

- Summary of the **NAND** latch:
  - **SET = 1, RESET = 1**—Normal resting state, outputs *remain* in state they *were* in prior to input.
  - **SET = 0, RESET = 1**—Output will go to  **$Q = 1$**  and remains there, even after SET returns HIGH.
    - Called *setting* the latch.
  - **SET = 1, RESET = 0**—Will produce  **$Q = 0$**  LOW and remains there, even after RESET returns HIGH.
    - Called *clearing or resetting* the latch.
  - **SET = 0, RESET = 0**—Tries to set and clear the latch at the same time, and produces  $Q = \bar{Q} = 1$ .
    - Output is *unpredictable*, and this input condition should not be used.

## 5-1 NAND Gate Latch - Summary

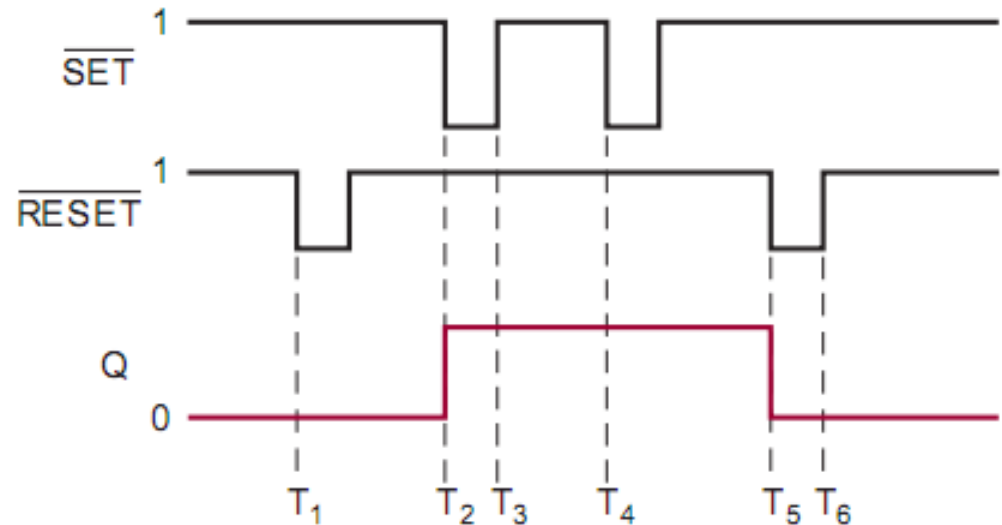
- Summary of the **NAND** latch:



| Set | Reset | Output    |
|-----|-------|-----------|
| 1   | 1     | No change |
| 0   | 1     | $Q = 1$   |
| 1   | 0     | $Q = 0$   |
| 0   | 0     | Invalid*  |

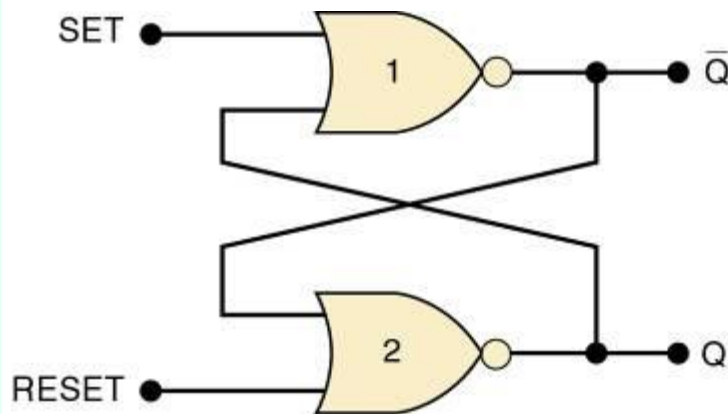
\*Produces  $Q = \bar{Q} = 1$ .

Assume that initially  $Q=0$



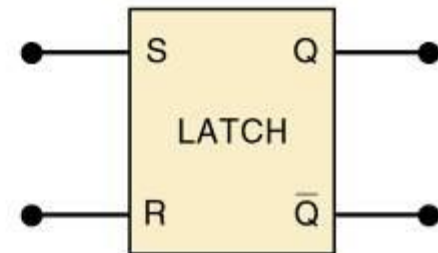
## 5-2 NOR Gate Latch

- Two cross-coupled **NOR** gates can be used as a **NOR** gate latch—similar to the **NAND** latch.
  - The **Q** and  **$\bar{Q}$**  outputs are reversed.



| Set | Reset | Output    |
|-----|-------|-----------|
| 0   | 0     | No change |
| 1   | 0     | Q = 1     |
| 0   | 1     | Q = 0     |
| 1   | 1     | Invalid*  |

\*Produces  $Q = \bar{Q} = 0$ .



The SET and RESET inputs are **active-HIGH**.  
Output will **change** when the input is pulsed **HIGH**.

## 5-1 NOR Gate Latch - Summary

- Summary of the **NOR** latch:
  - **SET = 0, RESET = 0**—Normal resting state, **No effect** on output state.
  - **SET = 1, RESET = 0**—will always set  $Q = 1$ , where it remains even after SET returns to 0.
  - **SET = 0, RESET = 1**—will always clear  $Q = 0$ , where it remains even after RESET returns to 0.
  - **SET = 1, RESET = 1**—Tries to set and reset the latch at the same time, and produces  $Q = \bar{Q} = 0$ .
    - Output is **unpredictable**, and this input condition should not be used.

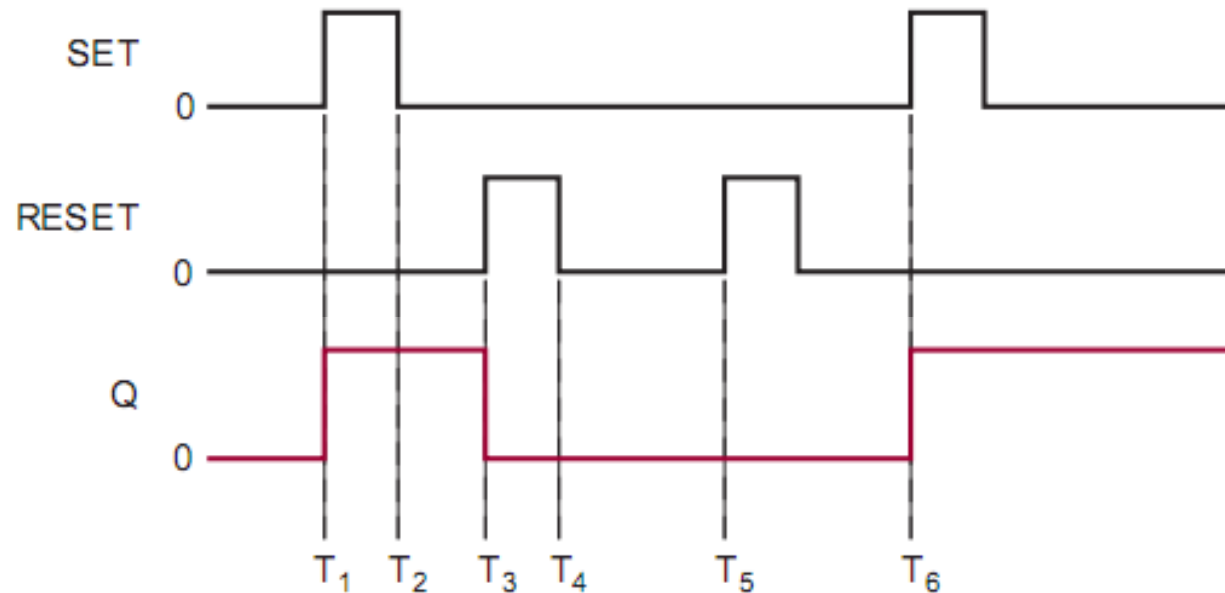
## 5-1 NAND Gate Latch - Summary

- Summary of the **NOR** latch:

Assume that initially  $Q=0$

| Set | Reset | Output    |
|-----|-------|-----------|
| 0   | 0     | No change |
| 1   | 0     | $Q = 1$   |
| 0   | 1     | $Q = 0$   |
| 1   | 1     | Invalid*  |

\*Produces  $Q = \bar{Q} = 0$ .



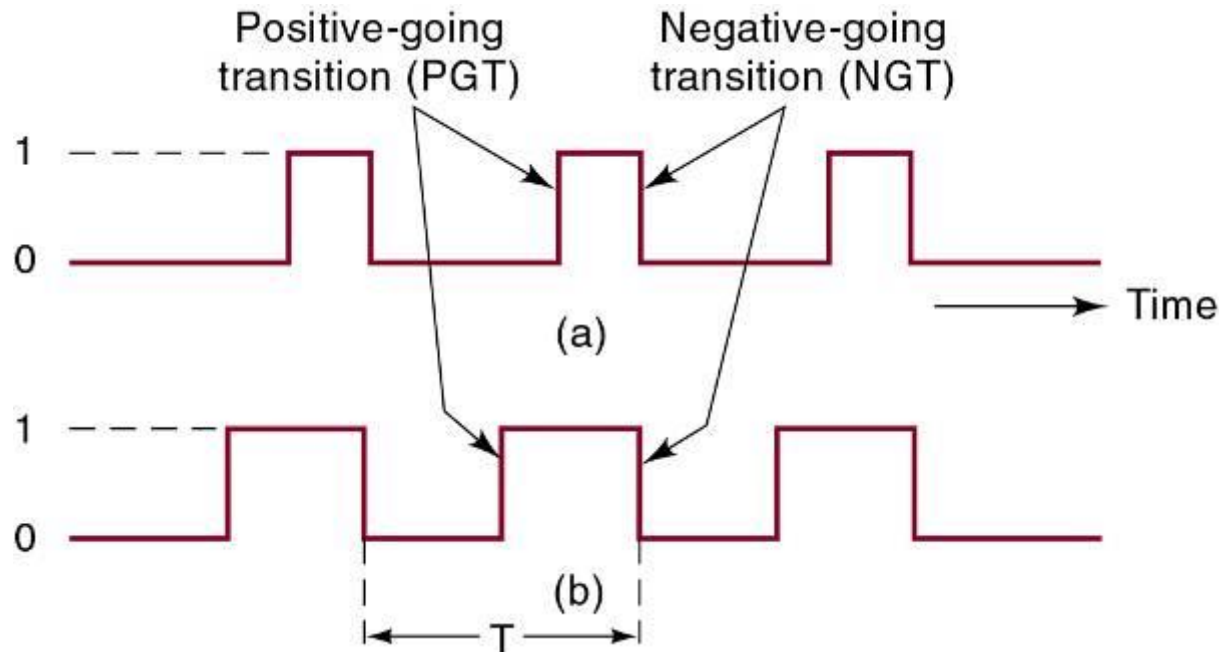
## 5-5 Clock Signals and Clocked Flip-Flops

- Digital systems can operate either *synchronously* or *asynchronously*.
  - **Asynchronous system**—outputs can change state at **any time** the input(s) change.
  - **Synchronous system**—output can change state only at a **specific time** in the clock cycle.

## 5-5 Clock Signals and Clocked Flip-Flops

- The clock signal is a rectangular pulse train or square wave.
  - Positive going transition (**PGT**)—clock pulse goes from 0 to 1.
  - Negative going transition (**NGT**)—clock pulse goes from 1 to 0.

Transitions are also called **edges**.

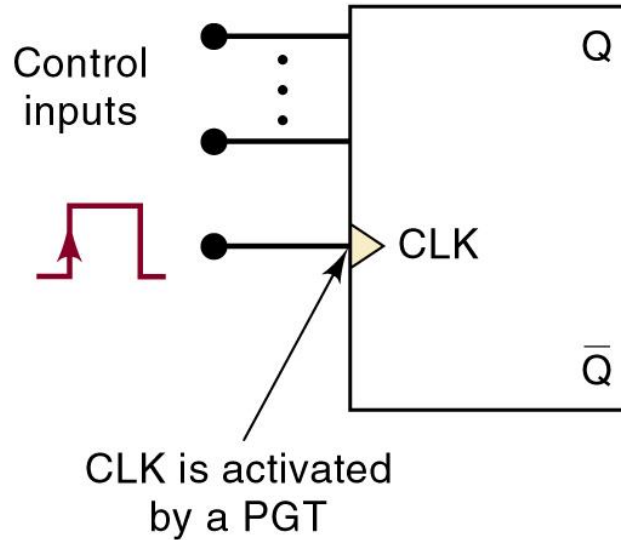




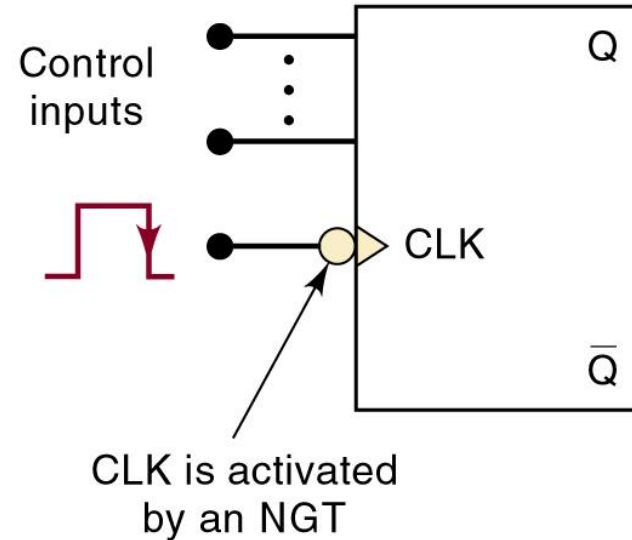
## 5-5 Clock Signals and Clocked Flip-Flops

- Clocked FFs change state on one or the other clock transitions.
  - Clock inputs are labeled **CLK, CK, or CP**.

A small triangle at the CLK input indicates that the input is activated with a PGT.

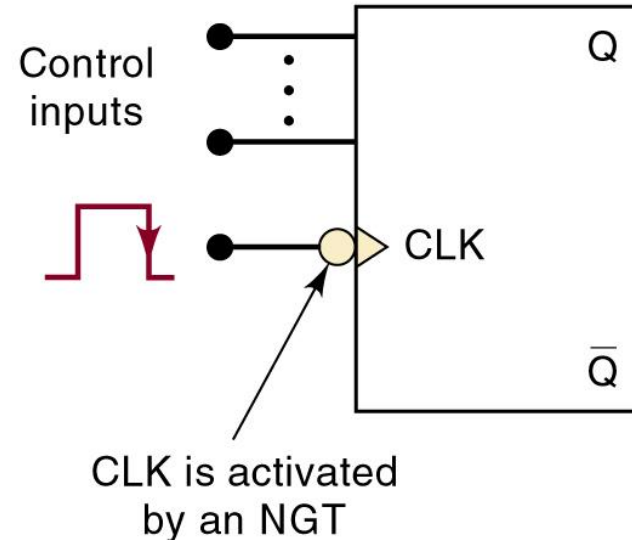
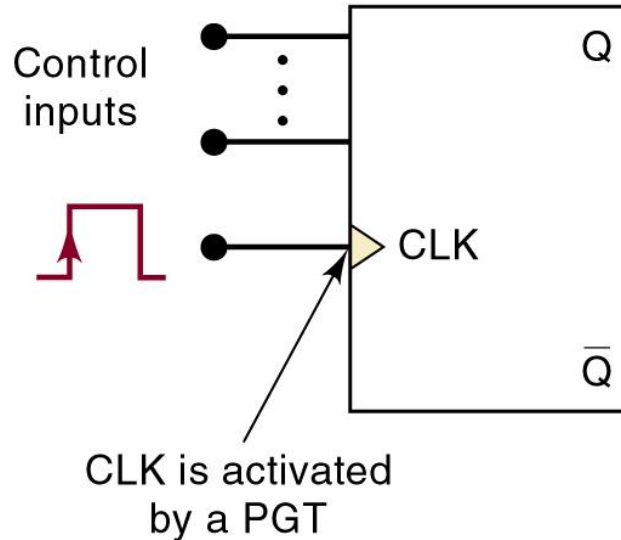


A bubble and a triangle indicates that the CLK input is activated with a NGT.



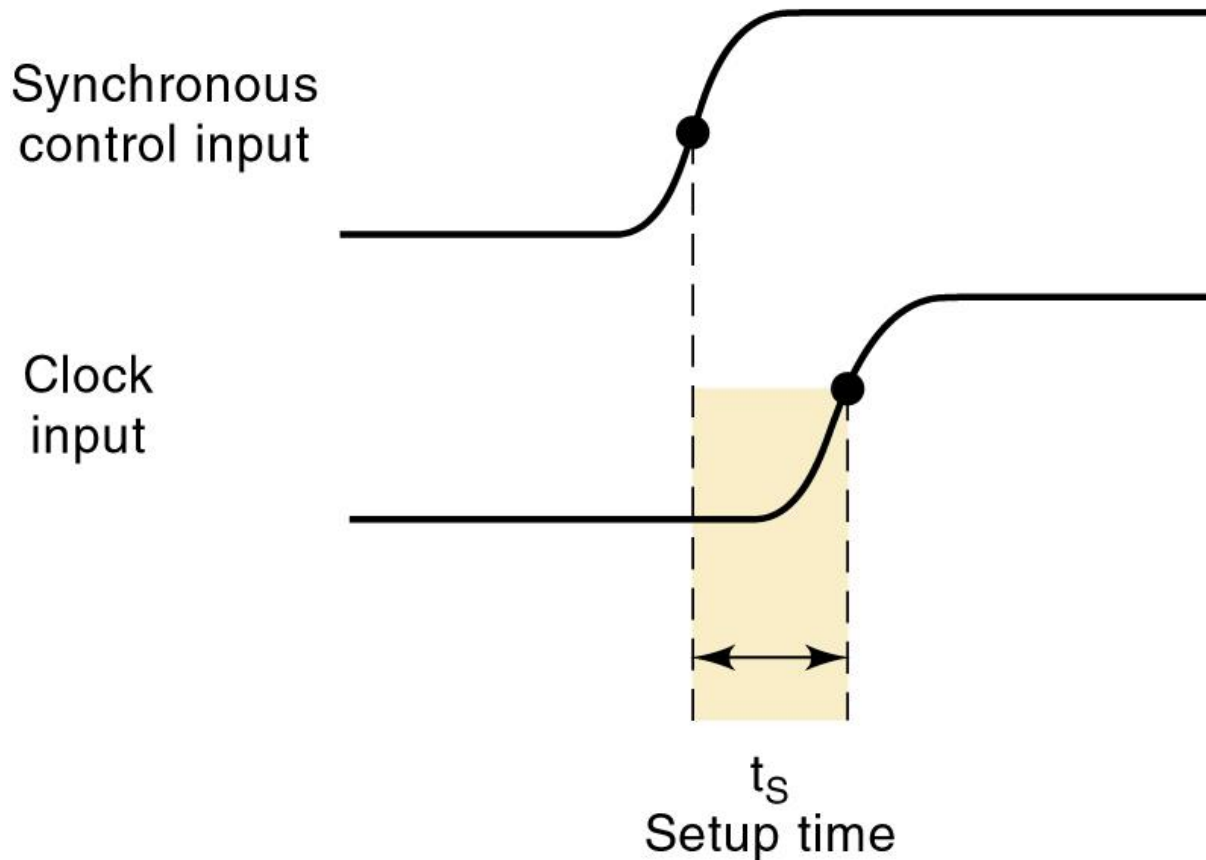
## 5-5 Clock Signals and Clocked Flip-Flops

- Control inputs have an **effect** on the output only at the **active** clock transition (**NGT** or **PGT**)—also called **synchronous** control inputs.
  - The control inputs get the outputs ready to change, but the change is not triggered until the CLK edge.



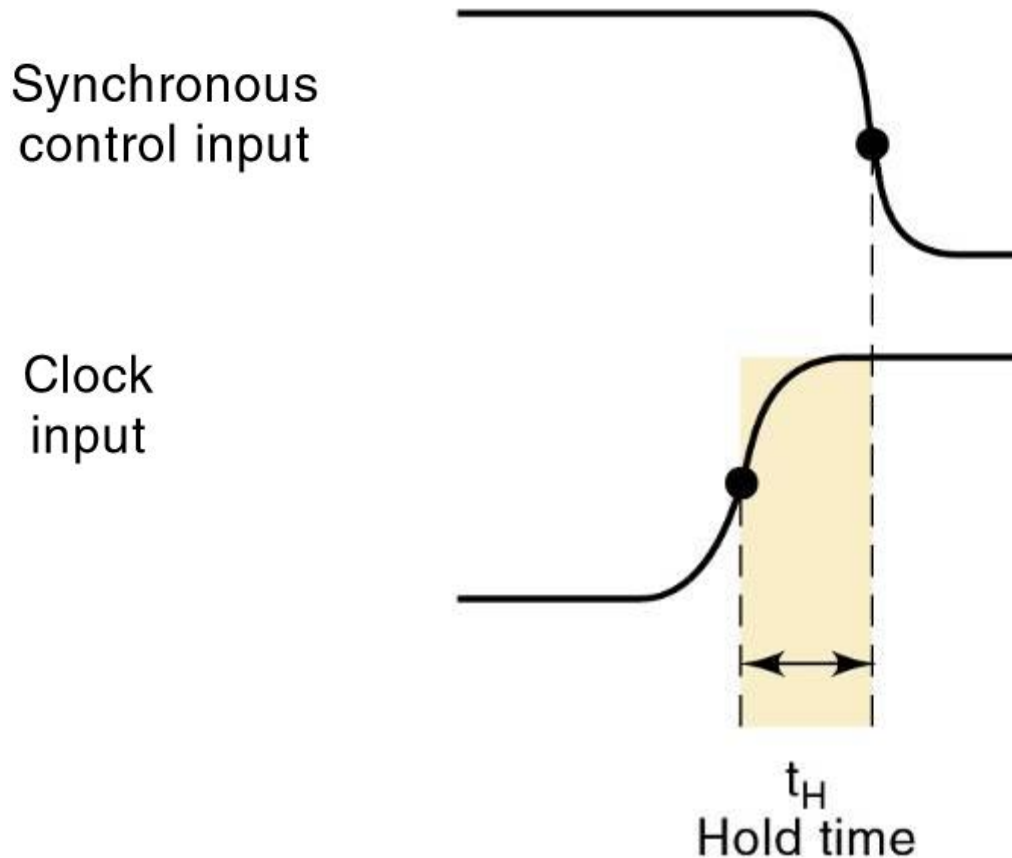
## 5-5 Clock Signals and Clocked Flip-Flops

- *Setup time* ( $t_s$ ) is the **minimum** time interval before the active CLK transition that the control input must be kept at the proper level.



## 5-5 Clock Signals and Clocked Flip-Flops

- *Hold time* ( $t_H$ ) is the time following the **active** transition of the CLK, during which the control input must be kept at the proper level.

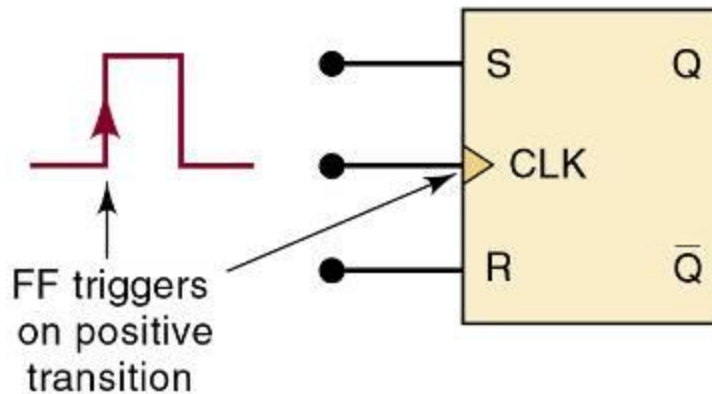


## 5-6 Clocked S-R Flip-Flop

- The  $S$  and  $R$  inputs are **synchronous control** inputs, which control the state the FF will go to when the **clock pulse** occurs.
  - The **CLK** input is the **trigger** input that causes the FF to change states according to the  $S$  and  $R$  inputs.
- SET-RESET (or SET-CLEAR) FF will change states at positive- or negative-going clock edges.

## 5-6 Clocked S-R Flip-Flop

**A clocked S-R flip-flop triggered by the positive-going edge of the clock signal.**



| Inputs |   |     | Output            |
|--------|---|-----|-------------------|
| S      | R | CLK | Q                 |
| 0      | 0 | ↑   | $Q_0$ (no change) |
| 1      | 0 | ↑   | 1                 |
| 0      | 1 | ↑   | 0                 |
| 1      | 1 | ↑   | Ambiguous         |

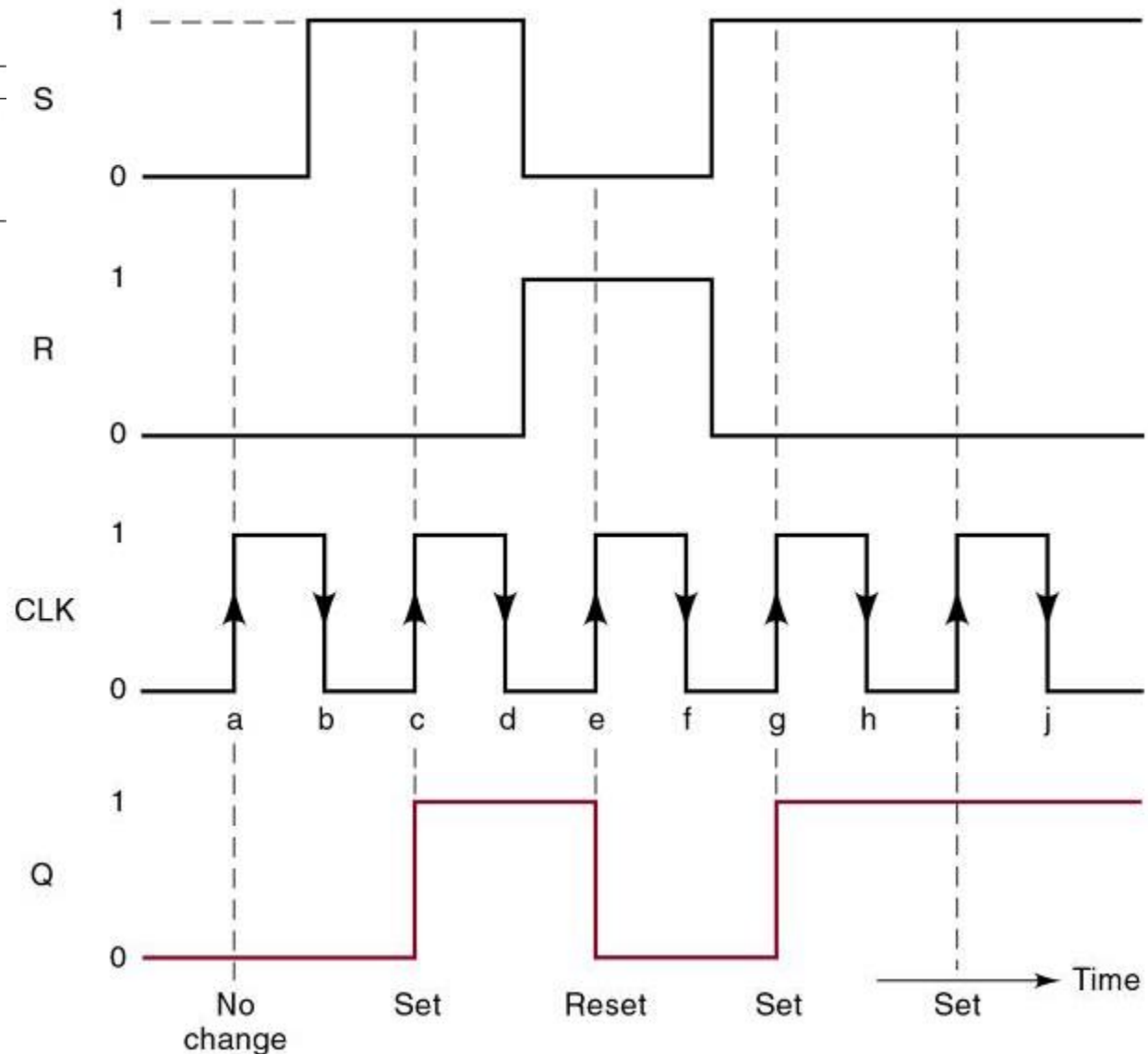
$Q_0$  is output level prior to ↑ of CLK.  
↓ of CLK produces no change in Q.

The  $S$  and  $R$  inputs control the state of the FF in the same manner as described earlier for the **NOR** gate latch, but the FF does *not* respond to these inputs *until* the occurrence of the PGT of the clock signal.

## 5-6 Clocked S-R Flip-Flop

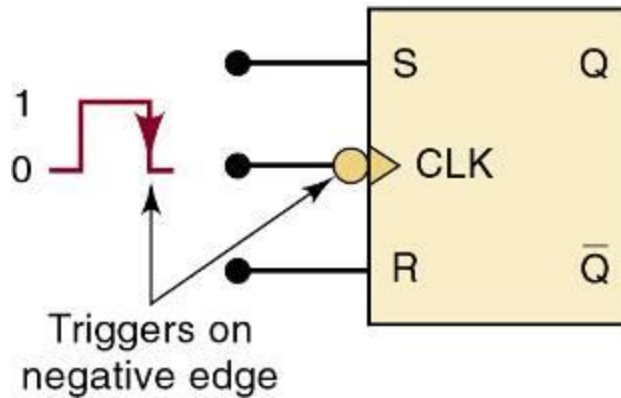
| Inputs |   |     | Output            |
|--------|---|-----|-------------------|
| S      | R | CLK | Q                 |
| 0      | 0 | ↑   | $Q_0$ (no change) |
| 1      | 0 | ↑   | 1                 |
| 0      | 1 | ↑   | 0                 |
| 1      | 1 | ↑   | Ambiguous         |

Waveforms of the operation of a clocked S-R flip-flop triggered by the positive-going edge of a clock pulse.



## 5-6 Clocked S-R Flip-Flop

**A clocked S-R flip-flop triggered by the negative-going edge of the clock signal.**

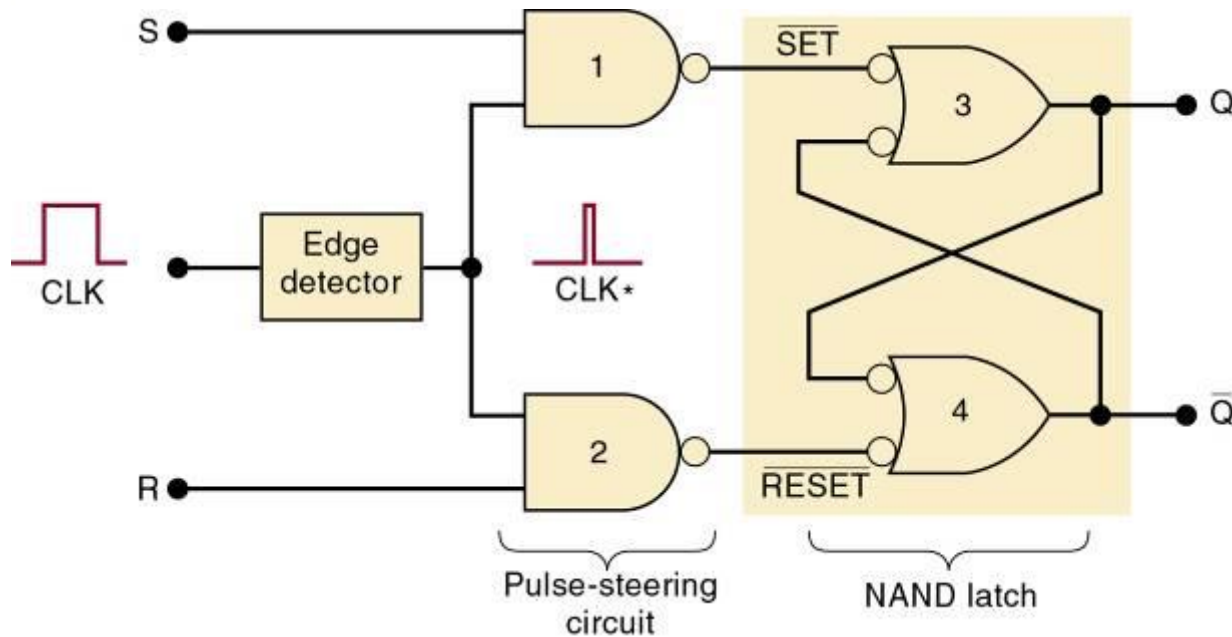


| Inputs |   |     | Output            |
|--------|---|-----|-------------------|
| S      | R | CLK | Q                 |
| 0      | 0 | ↓   | $Q_0$ (no change) |
| 1      | 0 | ↓   | 1                 |
| 0      | 1 | ↓   | 0                 |
| 1      | 1 | ↓   | Ambiguous         |

Both positive-edge and negative-edge triggering FFs are used in digital systems.



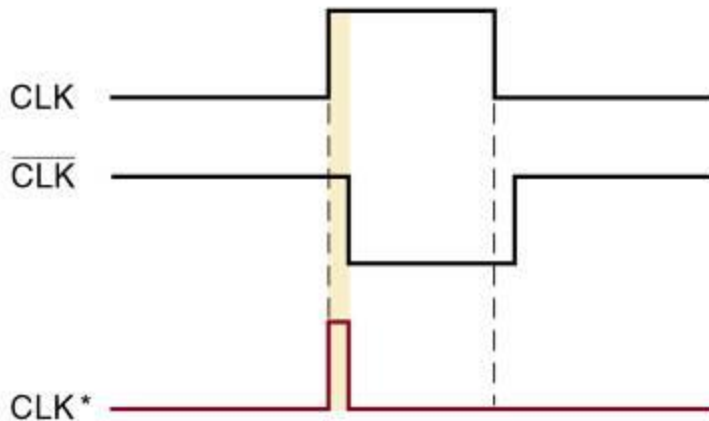
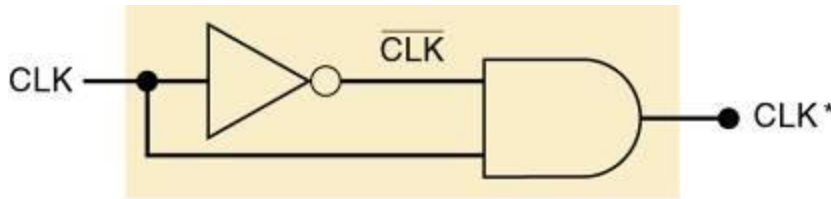
## 5-6 Clocked S-R Flip-Flop – Internal Circuitry



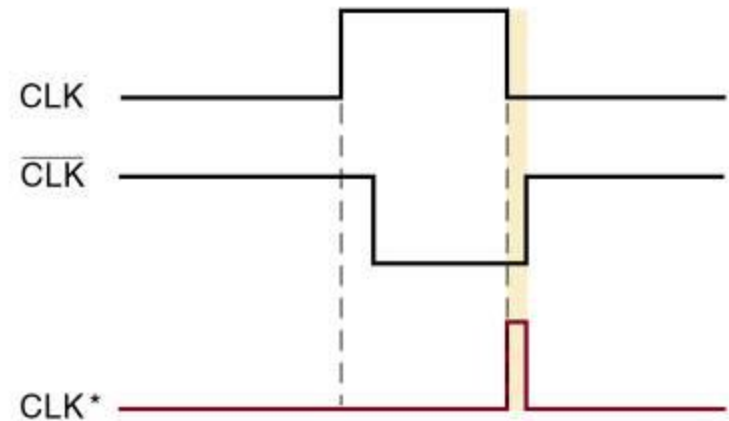
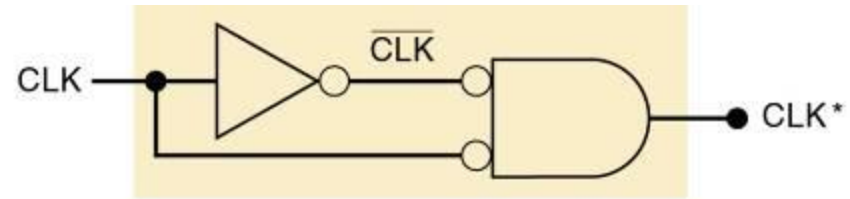
- An edge-triggered S-R flip-flop circuit features:
  - A basic **NAND** gate latch formed by **NAND-3** and **NAND-4**.
  - A **pulse-steering circuit** formed by **NAND-1** and **NAND-2**.
  - An **edge-detector circuit**.

## 5-6 Clocked S-R Flip-Flop – Internal Circuitry

- Implementation of edge-detector circuits used in edge-triggered flip-flops:
  - (a) PGT; (b) NGT.



(a)



(b)

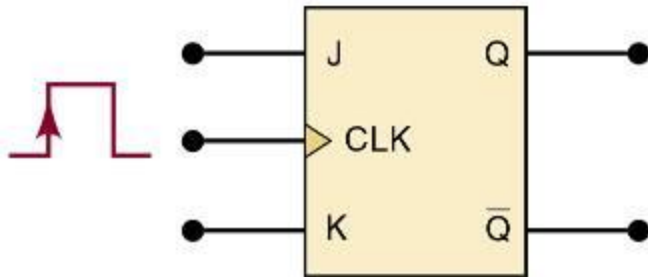
The duration of the  $CLK^*$  pulses is typically 2–5 ns.

## 5-7 Clocked J-K Flip-Flop

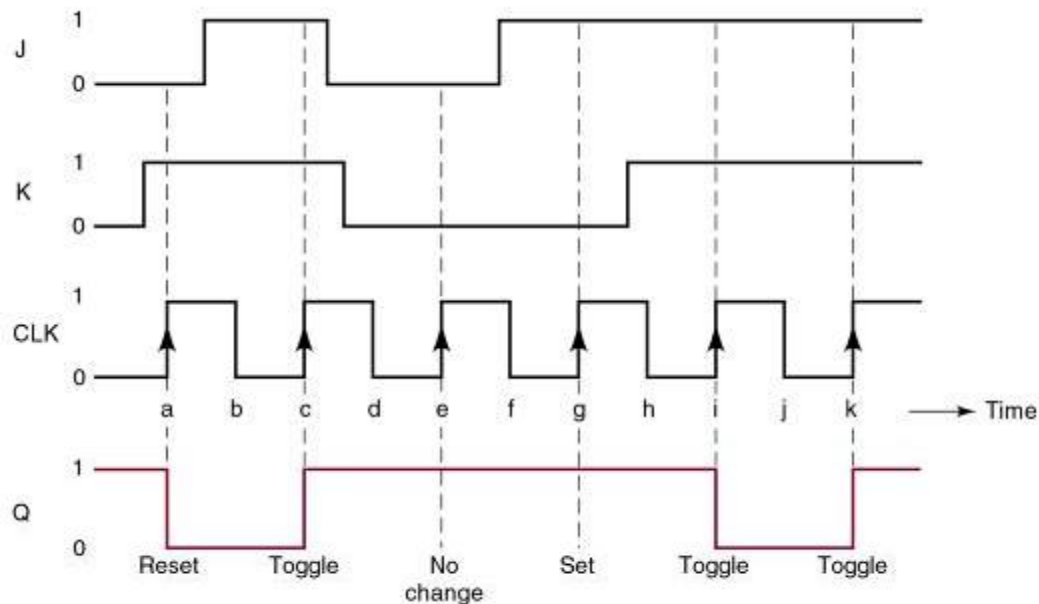
- Operates like the **S-R** FF.
  - J is SET, K is CLEAR.
- When **J** and **K** are both **HIGH**, output is **toggled** to the **opposite** state.
  - May be positive going or negative going clock trigger.
- Much more versatile than the S-R flip-flop, as it has no **ambiguous** states.
  - Has the ability to do everything the S-R FF does, plus operates in toggle mode.

## 5-7 Clocked J-K Flip-Flop

**Clocked J-K flip-flop that responds only to the positive edge of the clock.**

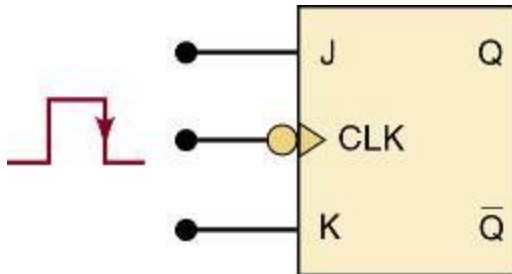


| J | K | CLK        | Q                     |
|---|---|------------|-----------------------|
| 0 | 0 | $\uparrow$ | $Q_0$ (no change)     |
| 1 | 0 | $\uparrow$ | 1                     |
| 0 | 1 | $\uparrow$ | 0                     |
| 1 | 1 | $\uparrow$ | $\bar{Q}_0$ (toggles) |



## 5-7 Clocked J-K Flip-Flop

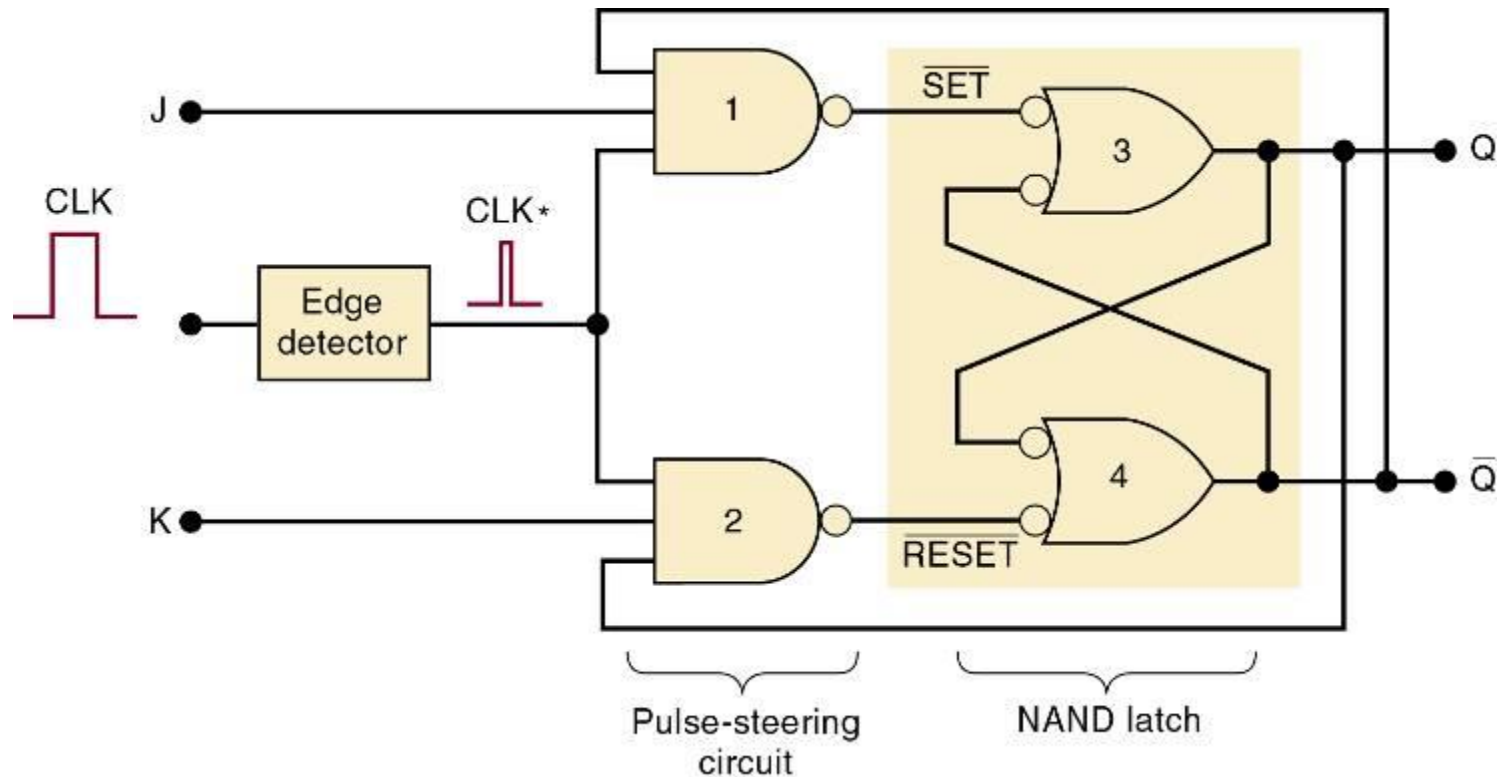
**Clocked J-K flip-flop that responds only to the negative edge of the clock.**



| J | K | CLK | Q                          |
|---|---|-----|----------------------------|
| 0 | 0 | ↓   | $Q_0$ (no change)          |
| 1 | 0 | ↓   | 1                          |
| 0 | 1 | ↓   | 0                          |
| 1 | 1 | ↓   | $\overline{Q_0}$ (toggles) |

## 5-7 Clocked J-K Flip-Flop – Internal Circuitry

- The internal circuitry of an edge-triggered J-K flip-flop contains the **same three** sections as the edge-triggered **S-R** flip-flop.

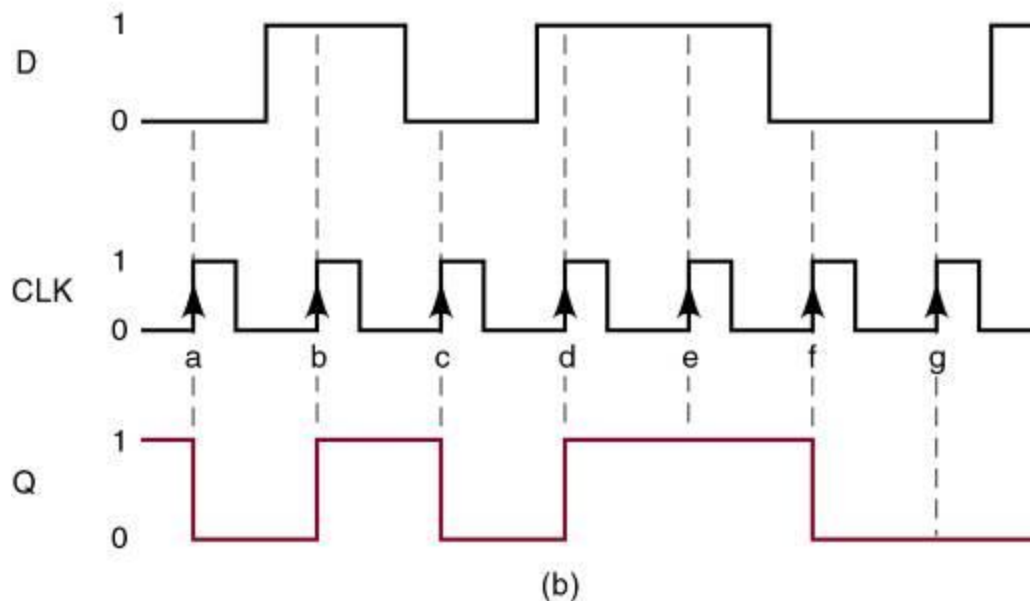
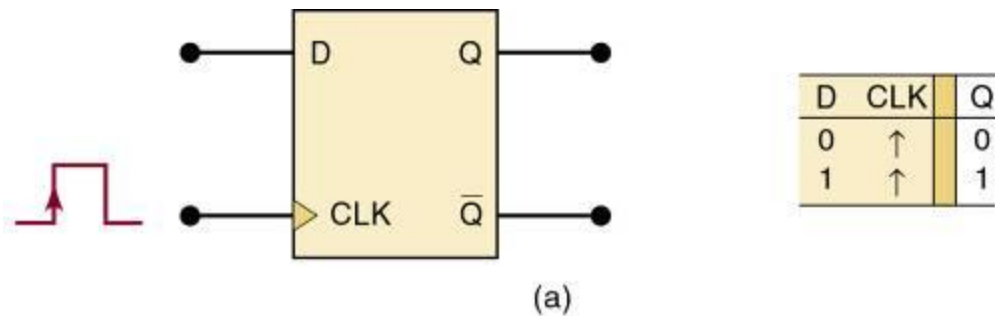


## 5-8 Clocked D Flip-Flop

- **One data input**—output **changes** to the value of the input at **either** the positive- or negative-going clock trigger.
- May be implemented with a J-K FF by tying the J input to the K input through an **inverter**.
- Useful for **parallel** data transfer.

## 5-8 Clocked D Flip-Flop

**D flip-flop that triggers only on positive-going transitions.**

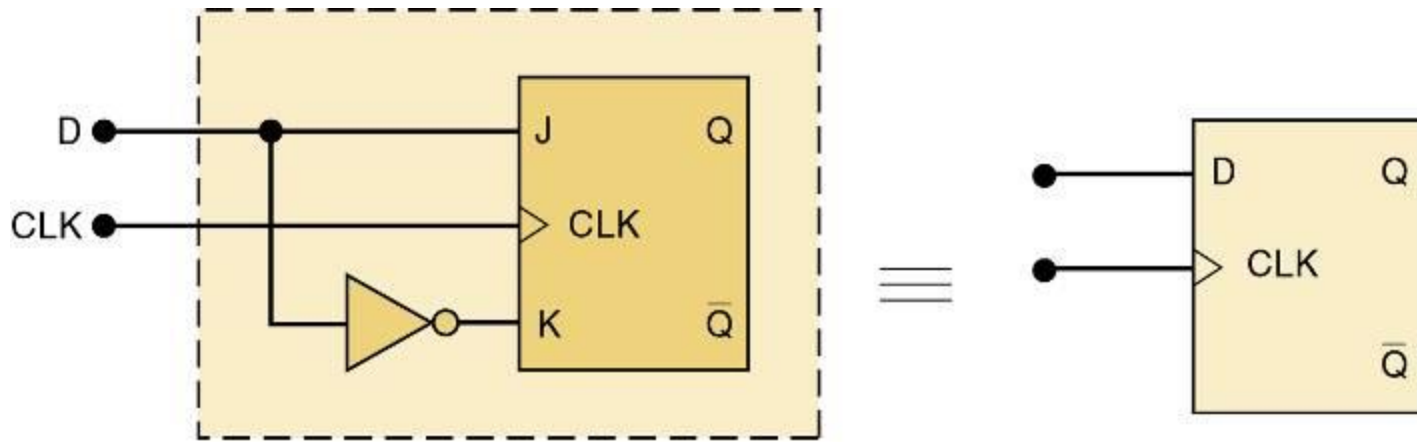




## 5-8 Clocked D Flip-Flop - Implementation

- An edge-triggered **D** flip-flop is implemented by adding a single **INVERTER** to the edge-triggered **J-K** flip-flop.
  - The same can be done to convert a S-R flip-flop to a D flip-flop.

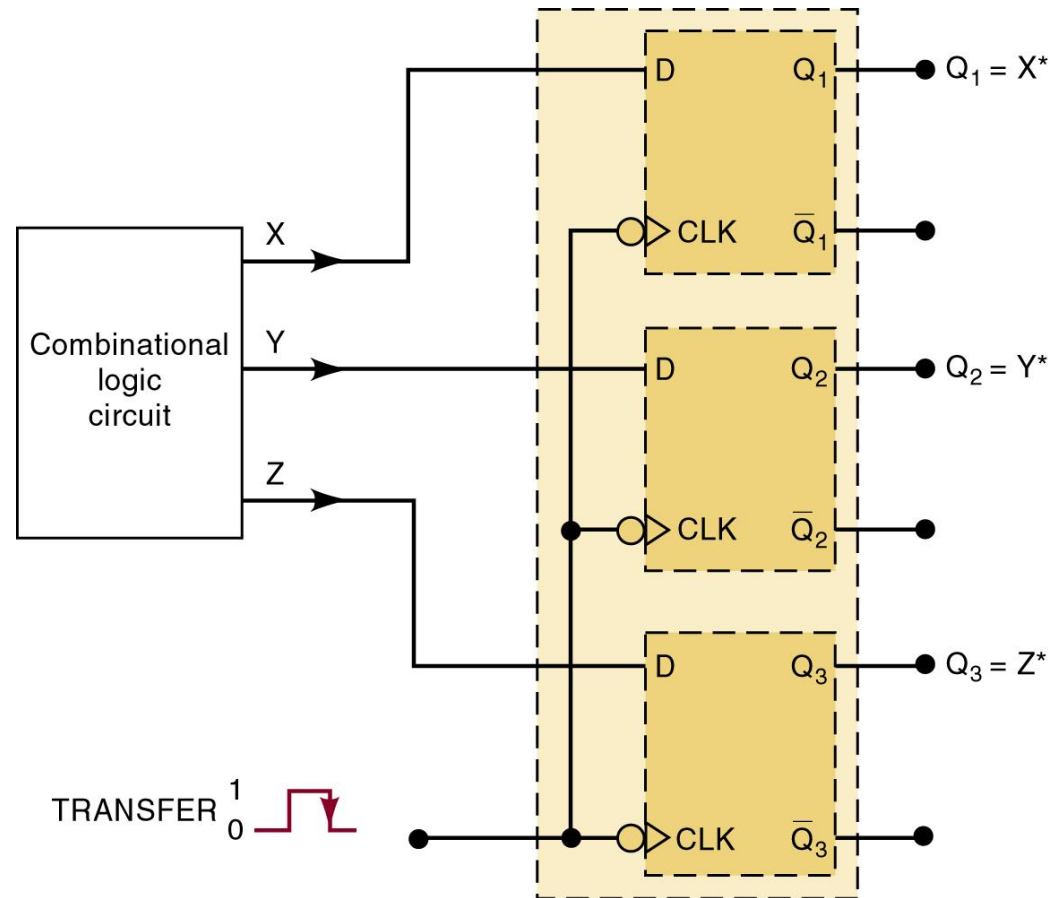
### Edge-triggered D flip-flop implementation from a J-K flip-flop.



## 5-8 Clocked D Flip-Flop – Parallel Data Transfer

**Outputs  $X$ ,  $Y$ ,  $Z$  are to be transferred to FFs  $Q_1$ ,  $Q_2$ , and  $Q_3$  for storage.**

Using D flip-flops, levels present at  $X$ ,  $Y$  &  $Z$  will be transferred to  $Q_1$ ,  $Q_2$  &  $Q_3$ , upon application of a TRANSFER pulse to the common  $CLK$  inputs.

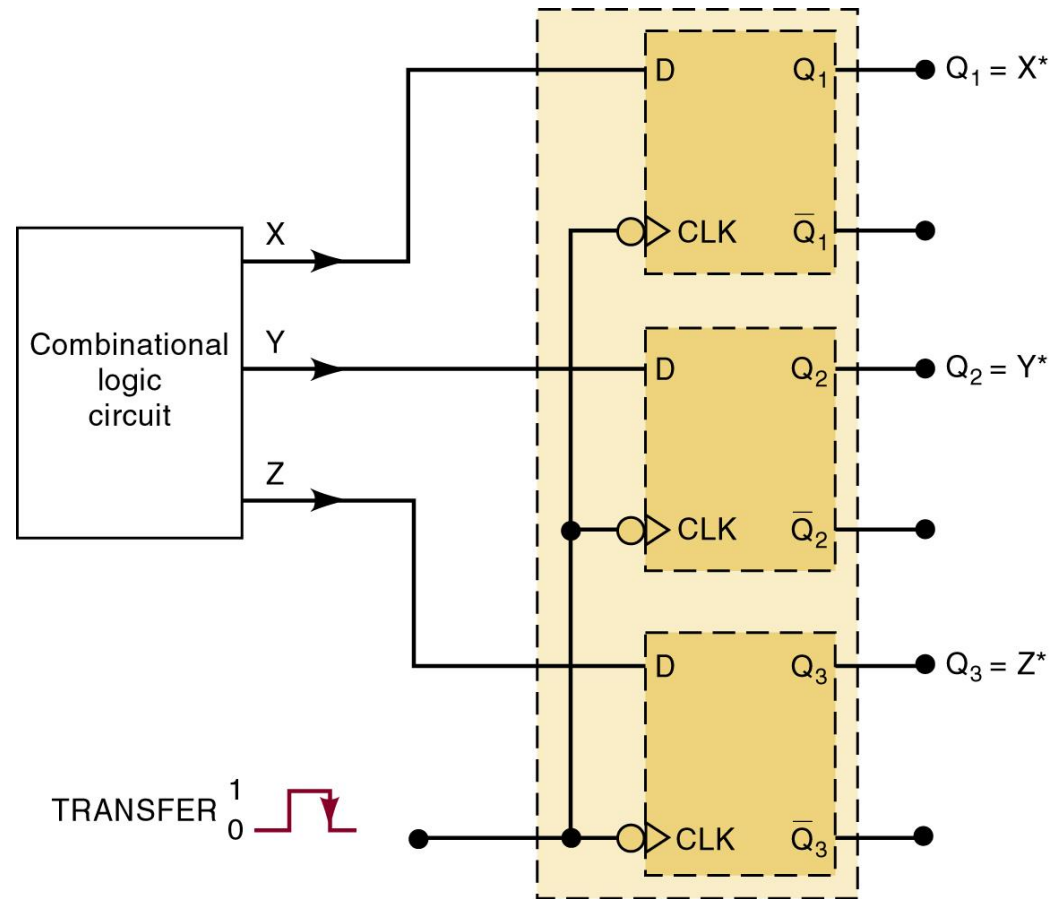


\*After occurrence of NGT

## 5-8 Clocked D Flip-Flop – Parallel Data Transfer

Outputs  $X$ ,  $Y$ ,  $Z$  are to be transferred to FFs  $Q_1$ ,  $Q_2$ , and  $Q_3$  for storage.

This is an example of **parallel data transfer** of binary data—the three bits  $X$ ,  $Y$  &  $Z$  are transferred *simultaneously*.



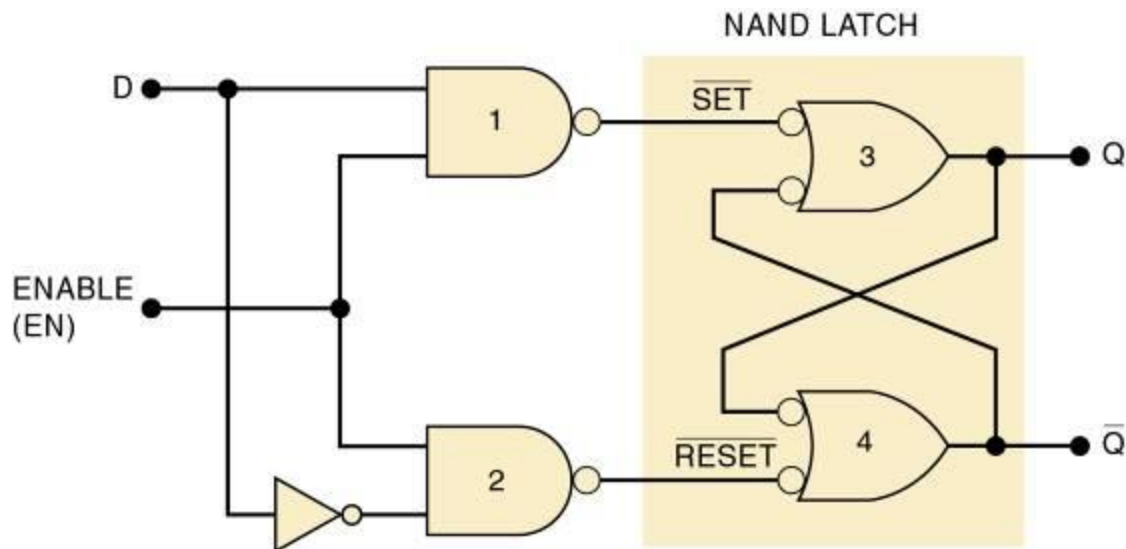
\*After occurrence of NGT

## 5-9 D Latch (Transparent Latch)

- The edge-triggered D flip-flop uses an edge-detector circuit to ensure the output **responds** to the *D* input *only* on active transition of the clock.
  - If this edge detector is not used, the resultant circuit operates as a **D latch**.

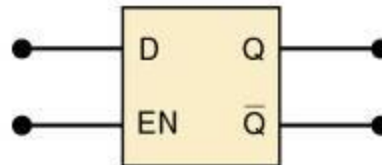
## 5-9 D Latch (Transparent Latch)

### D latch structure, function table, logic symbol.



| Inputs |   | Output            |
|--------|---|-------------------|
| EN     | D | Q                 |
| 0      | X | $Q_0$ (no change) |
| 1      | 0 | 0                 |
| 1      | 1 | 1                 |

"X" indicates "don't care."  
 $Q_0$  is state Q just prior to EN going LOW.



## 5-9 D Latch (Transparent Latch)

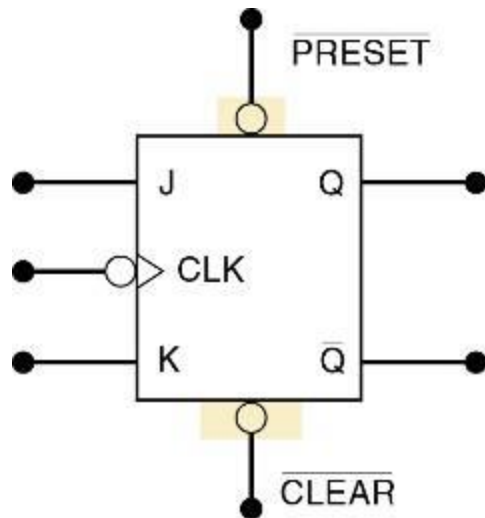
- The circuit contains the **NAND** latch and the steering **NAND** gates 1 and 2 *without* the edge-detector circuit.
- The common input to the steering gates is called an *enable* input (abbreviated *EN*)—rather than a clock input.
  - Its effect on the  $Q$  and  $\bar{Q}$  outputs is not restricted to occurring only on its transitions

## 5-10 Asynchronous Inputs

- Inputs that **depend** on the clock are **synchronous**.
- Most clocked FFs have **asynchronous** inputs that do **not** depend on the clock.
  - Labels **PRE** & **CLR** are used for asynchronous inputs.
- Active-LOW asynchronous inputs will have a **bar** over the labels and inversion bubbles.
- If the asynchronous inputs are not **used** they will be tied to their **inactive** state.

## 5-10 Asynchronous Inputs

### Clocked J-K flip-flop with asynchronous inputs.



| J | K | Clk | $\overline{\text{PRE}}$ | $\overline{\text{CLR}}$ | Q                        |
|---|---|-----|-------------------------|-------------------------|--------------------------|
| 0 | 0 | ↓   | 1                       | 1                       | Q (no change)            |
| 0 | 1 | ↓   | 1                       | 1                       | 0 (Synch reset)          |
| 1 | 0 | ↓   | 1                       | 1                       | 1 (Synch set)            |
| 1 | 1 | ↓   | 1                       | 1                       | $\bar{Q}$ (Synch toggle) |
| x | x | x   | 1                       | 1                       | Q (no change)            |
| x | x | x   | 1                       | 0                       | 0 (asynch clear)         |
| x | x | x   | 0                       | 1                       | 1 (asynch preset)        |
| x | x | x   | 0                       | 0                       | (Invalid)                |

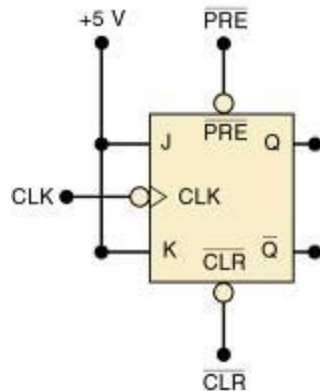


## 5-10 Asynchronous Inputs - Designations

- IC manufacturers do not agree on nomenclature for asynchronous inputs.
  - The most common designations are *PRE* (PRESET) and *CLR* (CLEAR).
    - Clearly distinguished from synchronous SET & RESET.
  - Labels such as *S-D* (direct SET) and *R-D* (direct RESET) are also used.

## 5-10 Asynchronous Inputs

A J-K FF that responds to a NGT on its clock input and has active-LOW asynchronous inputs.



| Point | Operation                                  |
|-------|--|
| a     | Synchronous toggle on NGT of CLK           |
| b     | Asynchronous set on $\overline{PRE} = 0$   |
| c     | Synchronous toggle                         |
| d     | Synchronous toggle                         |
| e     | Asynchronous clear on $\overline{CLR} = 0$ |
| f     | $\overline{CLR}$ overrides the NGT of CLK  |
| g     | Synchronous toggle                         |

