# Digital Circuits

## ECS 371

**Dr. Prapun Suksompong**

prapun@siit.tu.ac.th

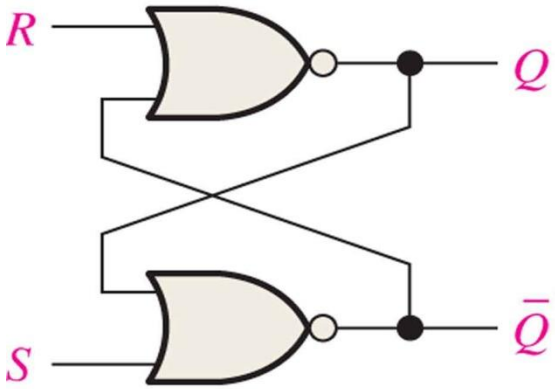
## Lecture 16

**Office Hours:**
**BKD 3601-7**
**Monday 9:00-10:30, 1:30-3:30**
**Tuesday 10:30-11:30**

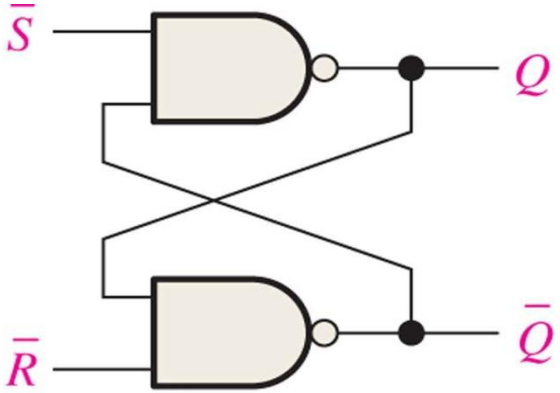
ECS371.PRAPUN.COM

# S-R Latch

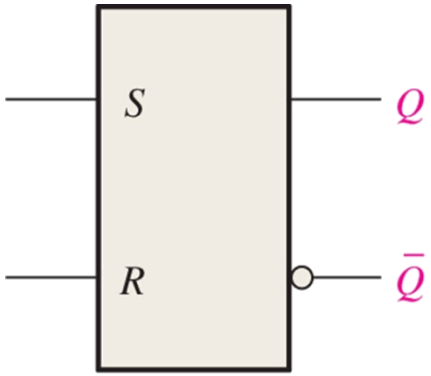
- There are two versions of SET-RESET (S-R) latches.

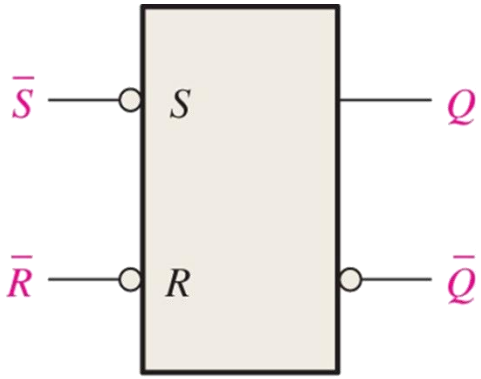



(a) Active-HIGH input S-R latch



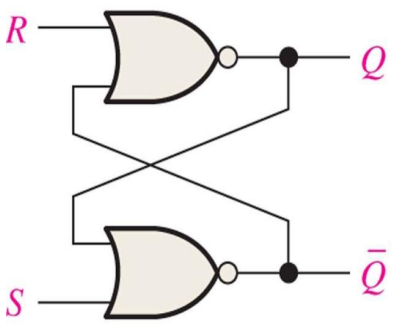

(b) Active-LOW input $\bar{S}$-$\bar{R}$ latch



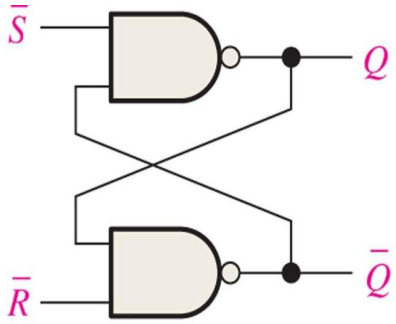



# S-R Latch (Remember This!)

- Two inputs
  - **S** for **set**
  - **R** for **reset**
- Two useful states (for normal operation)
  - When output Q = 1 and $\overline{Q}$ = 0, the latch is said to be in the **set state**.
  - When output Q = 0 and $\overline{Q}$ = 1, the latch is said to be in the **reset state**.

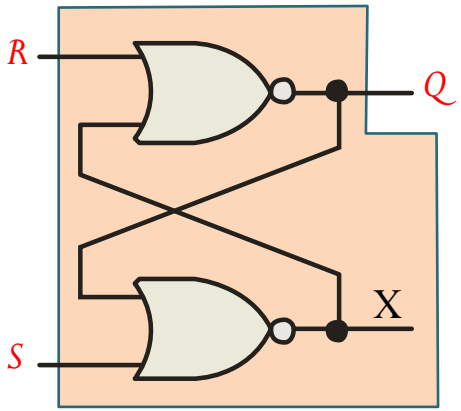



(a) Active-HIGH input S-R latch




(b) Active-LOW input $\overline{S}$-$\overline{R}$ latch

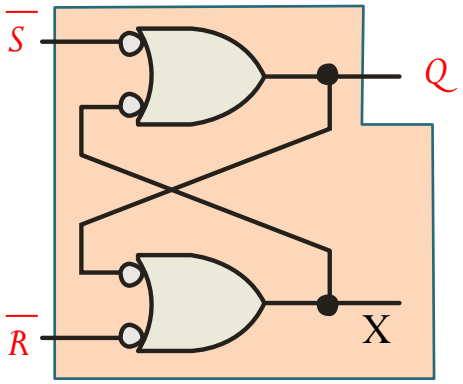
# The “Old Q”-“New Q” Analysis




$$Q_{new} = \overline{R + X}$$
$$= \overline{R + \overline{\overline{Q_{old} + S}}}$$
$$= \overline{R} \cdot \left( Q_{old} + S \right)$$

| Input | | Output |
|---|---|---|
| S | R | $Q_{new}$ |
| 0 | 0 | $Q_{old}$ |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

# The “Old Q”-“New Q” Analysis (2)



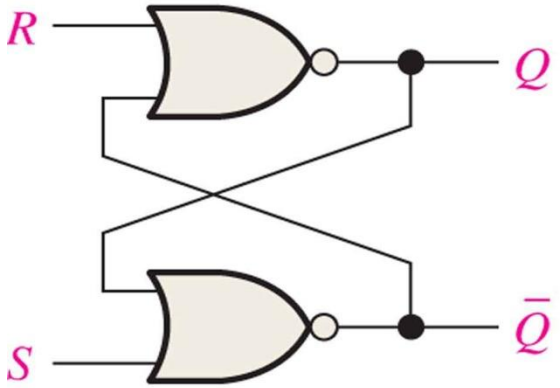

$$Q_{new} = \overline{\overline{S}} + \overline{X}$$

$$= \overline{\overline{S}} + \overline{\left(\overline{Q_{old}} + \overline{\overline{R}}\right)}$$

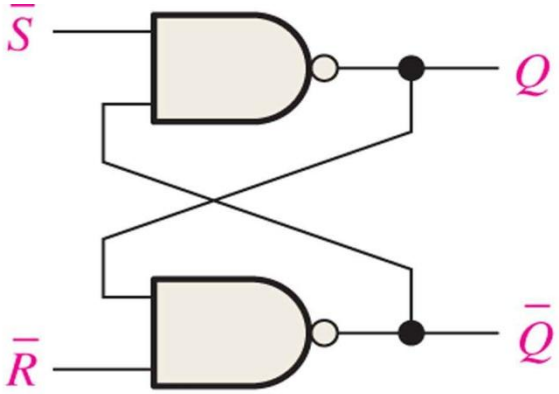
$$= \overline{\overline{S}} + Q_{old} \cdot \overline{R}$$

| Input | | Output |
|---|---|---|
| $\overline{S}$ | $\overline{R}$ | $Q_{new}$ |
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 1 | $Q_{old}$ |

# “Old Q”/“New Q” Analysis




(a) Active-HIGH input S-R latch

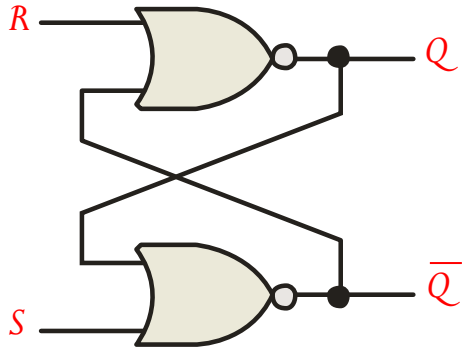



(b) Active-LOW input $\bar{S}$-$\bar{R}$ latch

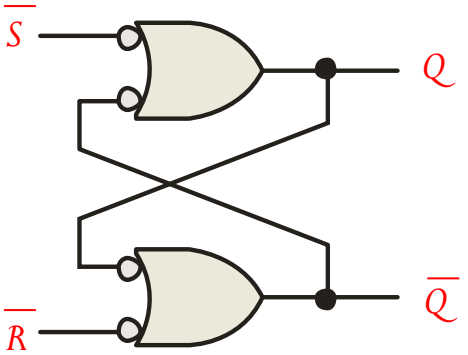
| Input | | Output |
|---|---|---|
| S | R | $Q_{new}$ |
| 0 | 0 | $Q_{old}$ |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

| Input | | Output |
|---|---|---|
| $\bar{S}$ | $\bar{R}$ | $Q_{new}$ |
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 1 | $Q_{old}$ |

# Expanded Version

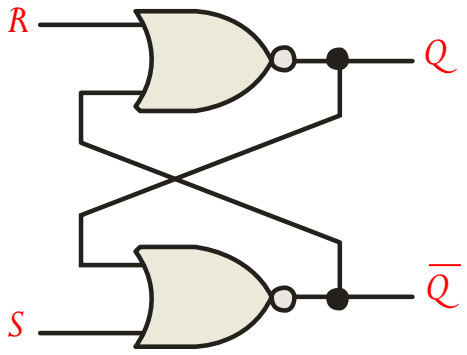



| Inputs | | Outputs | | Mode of Operation | Comment |
|---|---|---|---|---|---|
| $S$ | $R$ | $Q$ | $\overline{Q}$ | | |
| 0 | 0 | NC | NC | Hold | No change. |
| 0 | 1 | 0 | 1 | Reset | For RESETting Q to 0 |
| 1 | 0 | 1 | 0 | Set | For SETting Q to 1 |
| 1 | 1 | 0 | 0 | Prohibited | Invalid Condition |

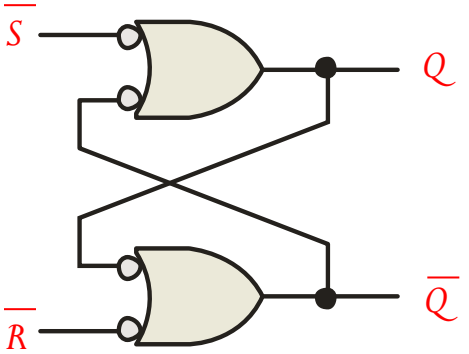



| Inputs | | Outputs | | Mode of Operation | Comment |
|---|---|---|---|---|---|
| $\overline{S}$ | $\overline{R}$ | $Q$ | $\overline{Q}$ | | |
| 0 | 0 | 1 | 1 | Prohibited | Invalid Condition |
| 0 | 1 | 1 | 0 | Set | For SETting Q to 1 |
| 1 | 0 | 0 | 0 | Reset | For RESETting Q to 0 |
| 1 | 1 | NC | NC | Hold | No change. |

# Short Version (Remember This!)




| Input | | Mode |
|---|---|---|
| S | R | |
| 0 | 0 | HOLD |
| 0 | 1 | RESET |
| 1 | 0 | SET |




| Inputs | | Mode |
|---|---|---|
| $\overline{S}$ | $\overline{R}$ | |
| 0 | 1 | SET |
| 1 | 0 | RESET |
| 1 | 1 | HOLD |

# Operating S-R latch

| Input | | Mode |
|---|---|---|
| S | R | |
| 0 | 0 | HOLD |
| 0 | 1 | RESET |
| 1 | 0 | SET |

- Under normal conditions, both inputs of the latch remain at 0 unless the state is to be change.
- The application of a 1 to the **S input** causes the latch to go to the **set state**.
  - The S input must go back to 0 before R is changed to 1 to avoid occurrence of the undefined state.
  - Applying a 0 to S with R = 0 leaves the circuit in the same state.
- The application of a 1 to the **R input** causes the latch to go to the **reset state**.
  - We can then remove the one from R, and the circuit remains in the reset state.

# (1,1) Problem for S-R Latch

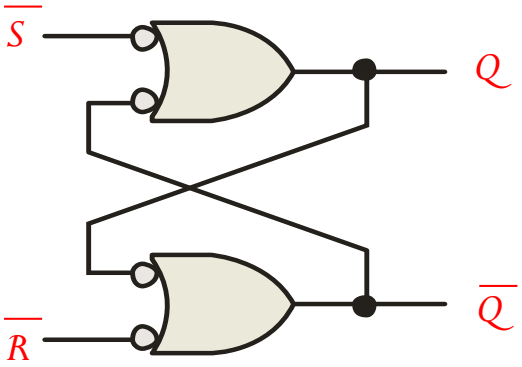
- If a 1 is applied to both the inputs of the latch, both outputs go to 0.
- This produces an undefined state.
- It results in an indeterminate or unpredictable next state when both inputs return to 0 simultaneously.
- In normal operation, these problems are avoided by making sure that 1's are not applied to both inputs simultaneously.

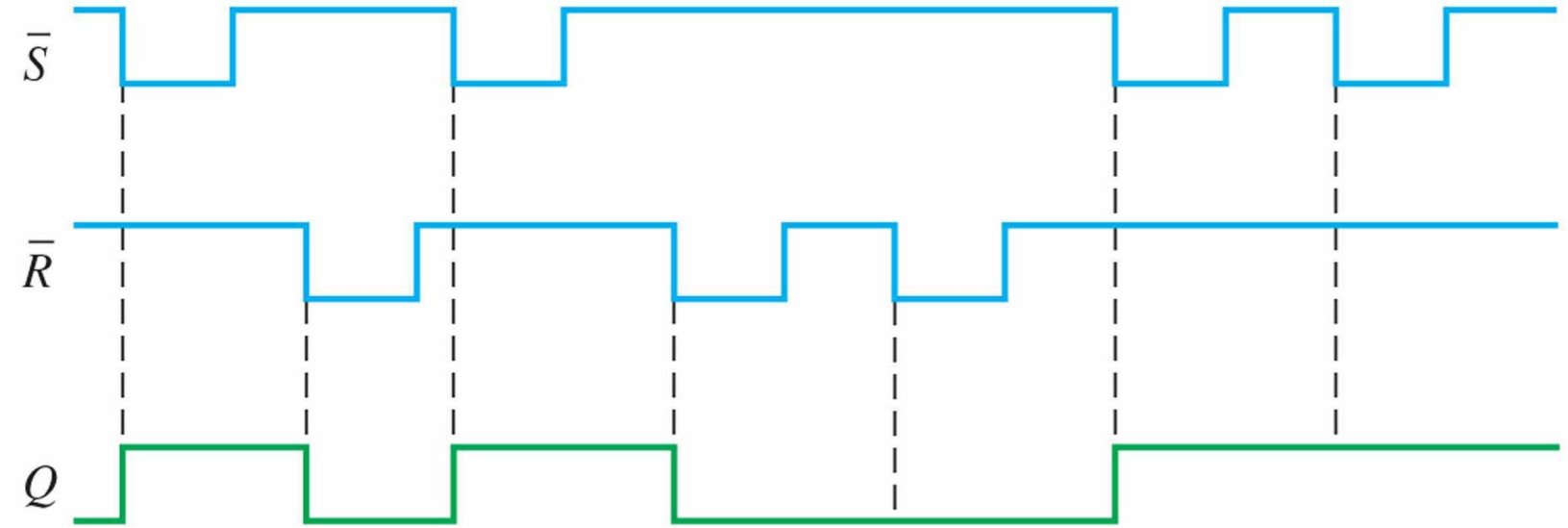
# Operating $\overline{S}$-$\overline{R}$ latch

| Inputs | | Mode |
|---|---|---|
| $\overline{S}$ | $\overline{R}$ | |
| 0 | 1 | SET |
| 1 | 0 | RESET |
| 1 | 1 | HOLD |

- Under normal conditions, both inputs of the latch remain at 1 unless the state is to be change.
- The application of a 0 to the **$\overline{S}$ input** causes the latch to go to the **set state**.
  - The S input must go back to 1 before R is changed to 1 to avoid occurrence of the undefined state.
  - Applying a 1 to S with R = 1 leaves the circuit in the same state.
- The application of a 0 to the **$\overline{R}$ input** causes the latch to go to the **reset state**.
  - We can then remove the 0 from R, and the circuit remains in the reset state.

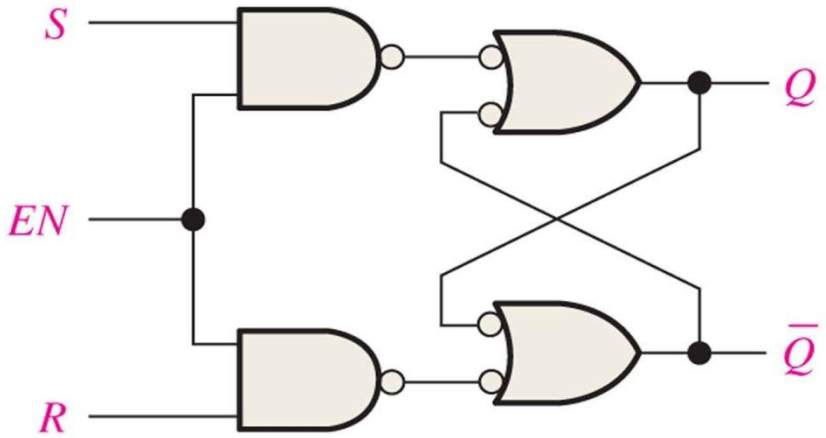
# Example



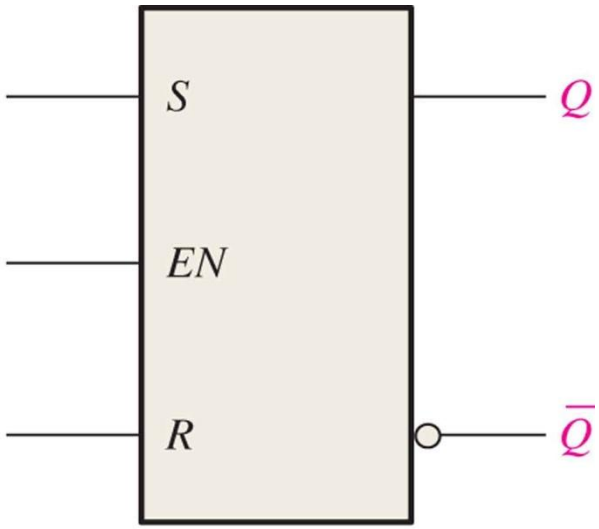



# Gated Latch

- A **gated latch** is a variation on the basic latch.
- The gated latch has an additional input, called enable (*EN*) that must be HIGH in order for the latch to respond to the *S* and *R* inputs.




(a) Logic diagram




(b) Logic symbol

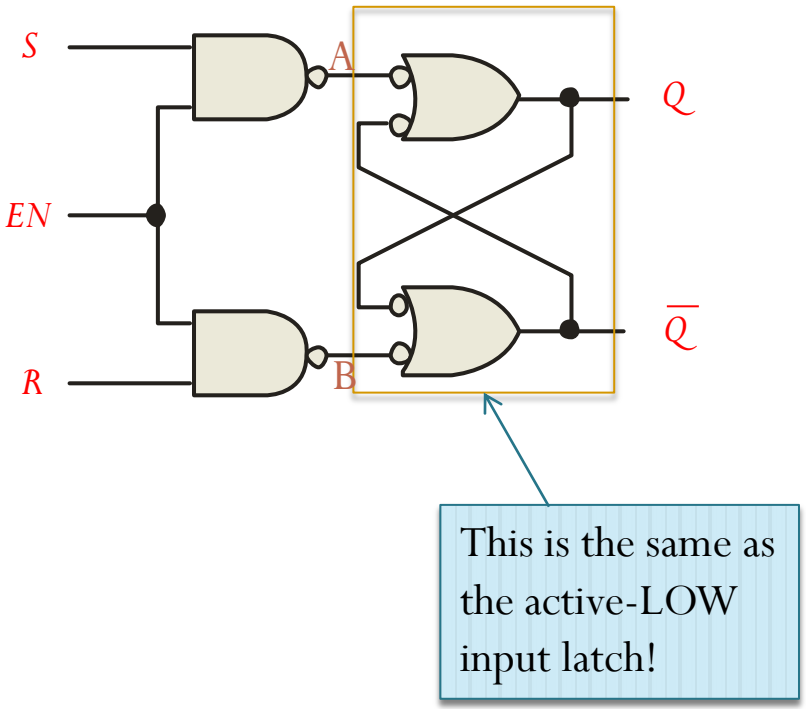
# Gated Latch

Observe that:

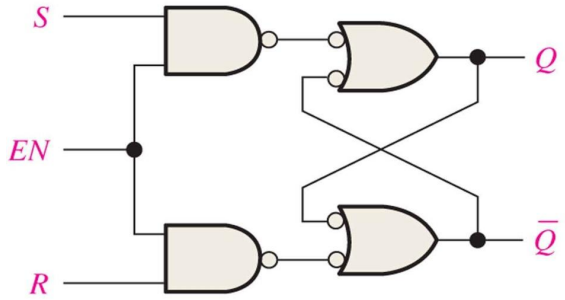
$$A = \overline{S \cdot EN} = \overline{S} + \overline{EN}$$

$$B = \overline{R \cdot EN} = \overline{R} + \overline{EN}$$

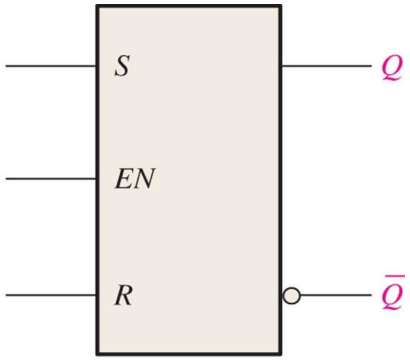
| EN | A | B |
|---|---|---|
| 0 ⇨ | 1 | 1 |
| 1 ⇨ | $\overline{S}$ | $\overline{R}$ |



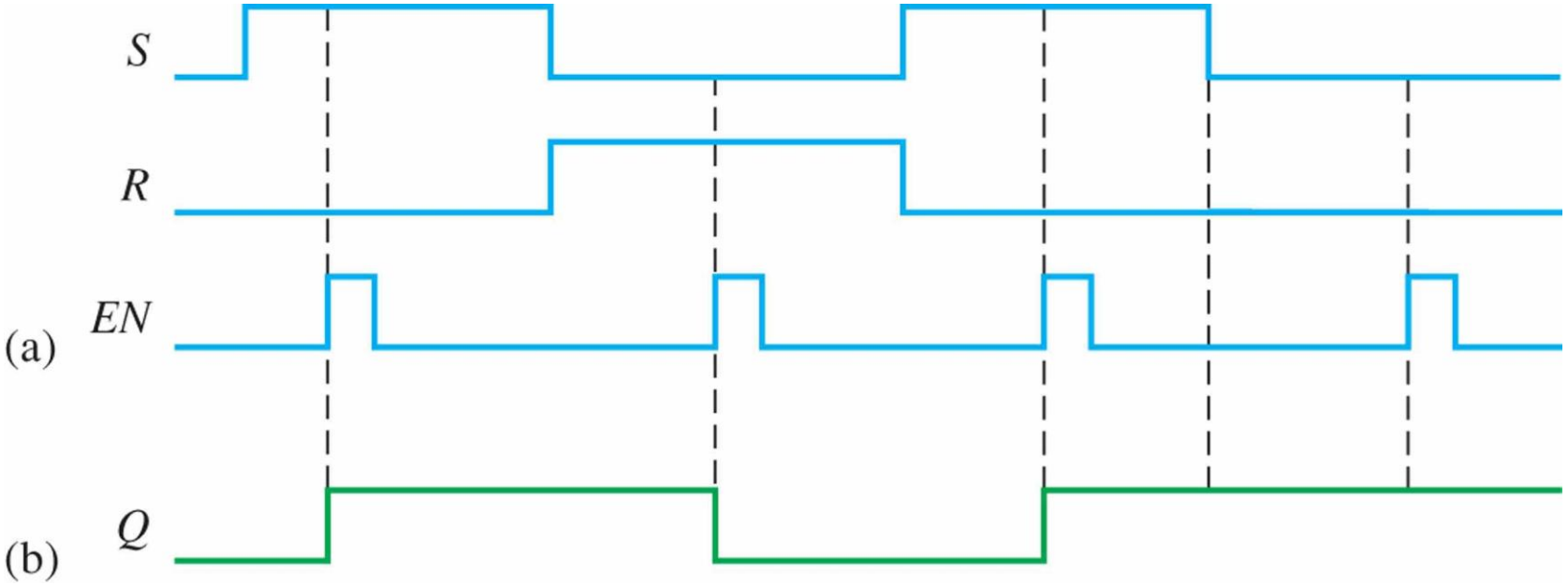
# Example: Gated S-R Latch




(a) Logic diagram

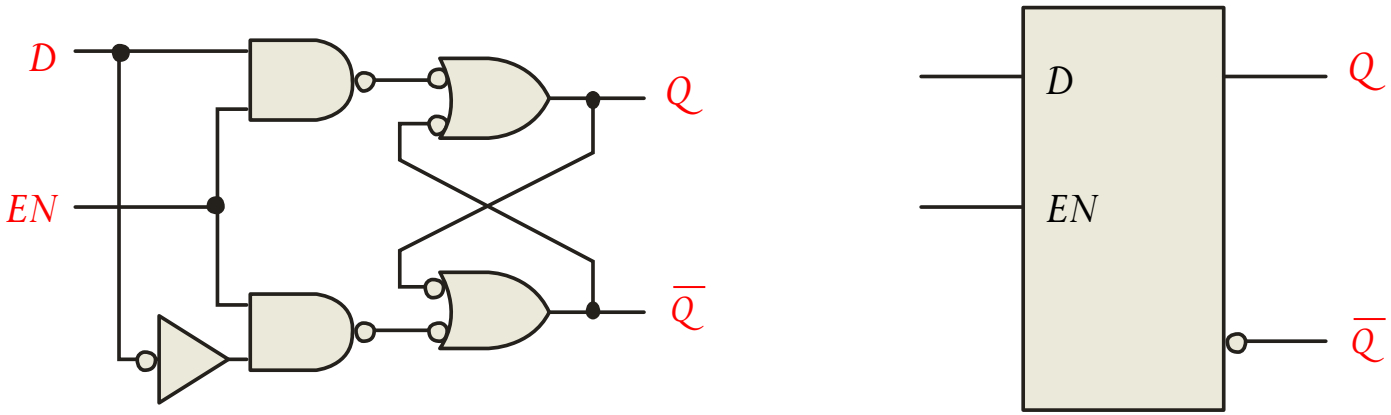



(b) Logic symbol

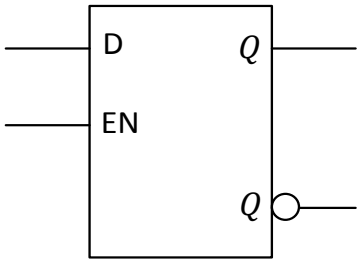


# Gated D latch

- The D latch is a variation of the S-R latch.
- Has only one input in addition to EN.
  - This input is called the D (data) input.
- Combine the S and R inputs into a single D input.



# Gated D Latch: Operation

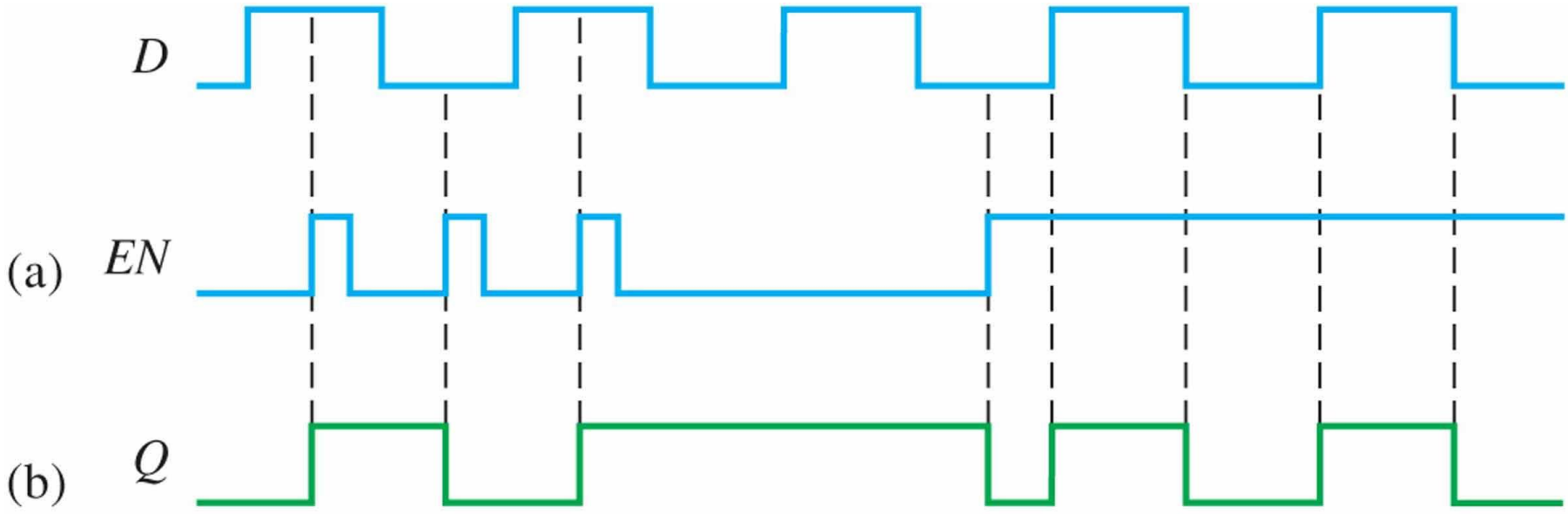



- A simple rule for the D latch is:
  - Q follows D when the Enable is active/asserted.
    - In this situation, the latch is said to be "open" and the path from D input to Q output is "transparent".
    - The circuit is often called a transparent latch for this reason.
- When EN is LOW, the state of the latch is not affected by the D input.
  - In this situation, the latch is said to be "close"
  - The Q output retains its last value and no longer changes in response to D, as long as EN remains negated.
- Output is "latched" at the last value when the enable signal becomes not asserted.
- Truth Table:

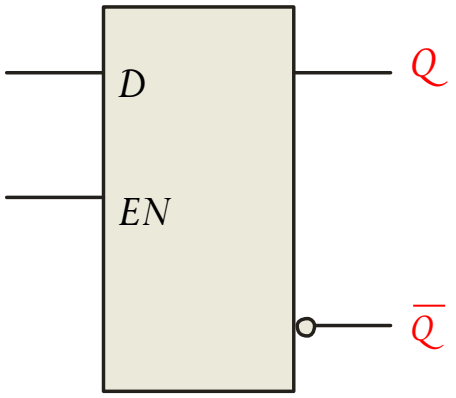
| Inputs | | Outputs | | |
|---|---|---|---|---|
| *D* | *EN* | *Q* | $\overline{Q}$ | Comments |
| 0 | 1 | 0 | 1 | RESET |
| 1 | 1 | 1 | 0 | SET |
| X | 0 | $Q_0$ | $\overline{Q}_0$ | No change |

$Q_0$ is the prior output level before the indicated input conditions were established.

# Example: Gated D Latch




Q follows D when the Enable is active.



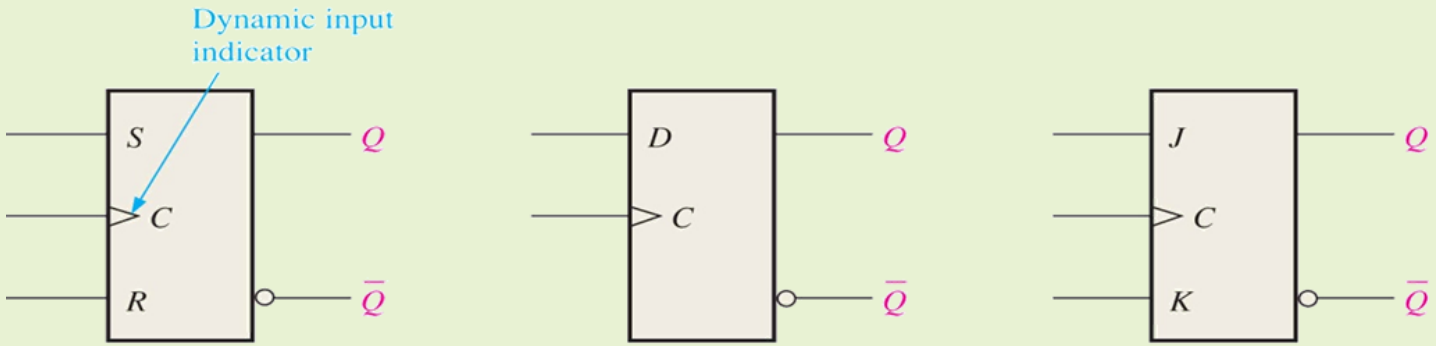
# Flip-Flop

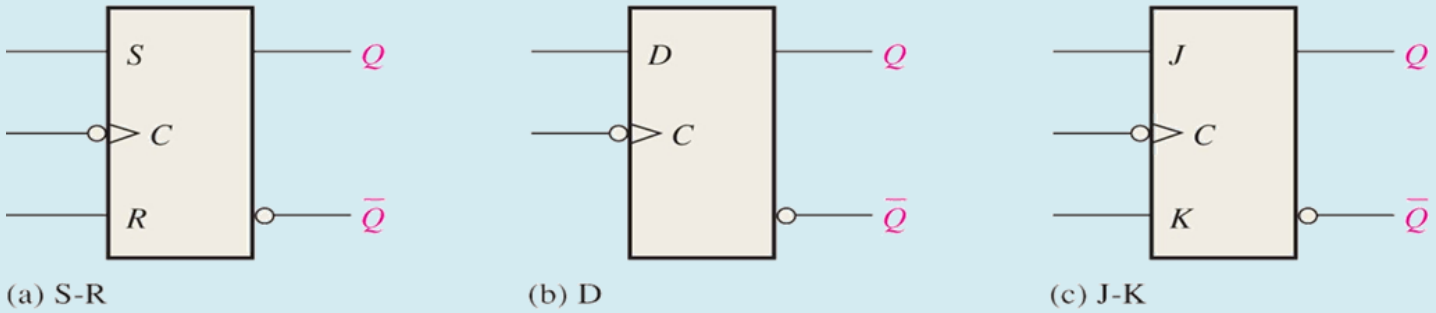
- Latches sample their inputs (and change states) any time the EN bit is asserted
- Many times we want more control over when to sample the input
- A **flip-flop** differs from a latch in the manner it changes states.
- A flip-flop is a *clocked* device.
- Flip-flops are **synchronous**: the output changes state only at a specified point on the triggering input called the **clock (CLK)**
  - In other words, changes in the output occur in synchronization with the clock.
- An **edge-triggered flip-flop** changes state either at the **positive edge (rising edge)** or at the **negative edge (falling edge)** of the clock pulse.

# Edge-Triggered Flip-Flops

"Edge-triggered flip-flop" is redundant (all flip-flops are edge-triggered

Positive edge-triggered (no bubble at C input)

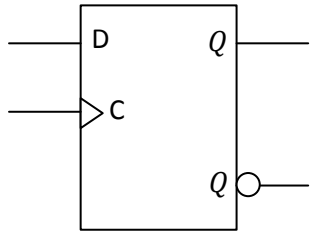



(a) S-R (b) D (c) J-K

Negative edge-triggered (bubble at C input)

# D Flip-Flop




- The truth table for a positive-edge triggered D flip-flop shows an up arrow to remind you that it is sensitive to its D input only on the **rising edge of the clock**.
- The truth table for a negative-edge triggered D flip-flop is identical except for the direction of the arrow.

| Inputs | | Outputs | | |
|---|---|---|---|---|
| *D* | CLK | *Q* | $\overline{Q}$ | Comments |
| 1 | ↑ | 1 | 0 | SET |
| 0 | ↑ | 0 | 1 | RESET |

(a) Positive-edge triggered

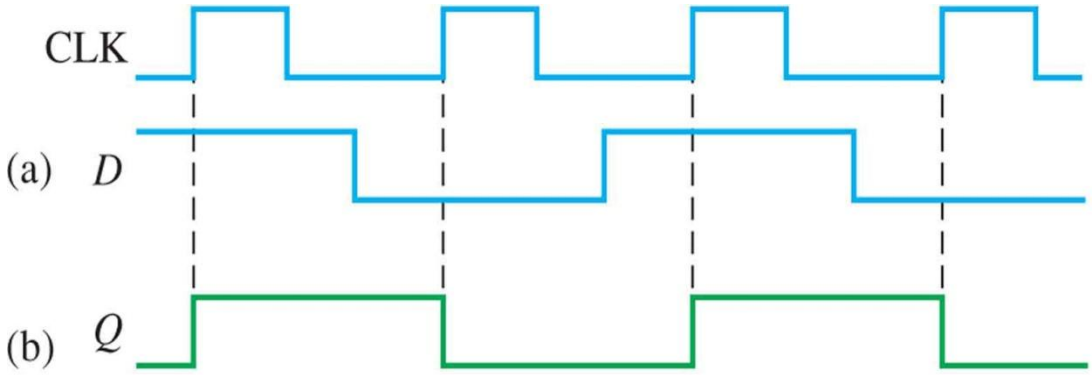
| Inputs | | Outputs | | |
|---|---|---|---|---|
| *D* | CLK | *Q* | $\overline{Q}$ | Comments |
| 1 | ↓ | 1 | 0 | SET |
| 0 | ↓ | 0 | 1 | RESET |

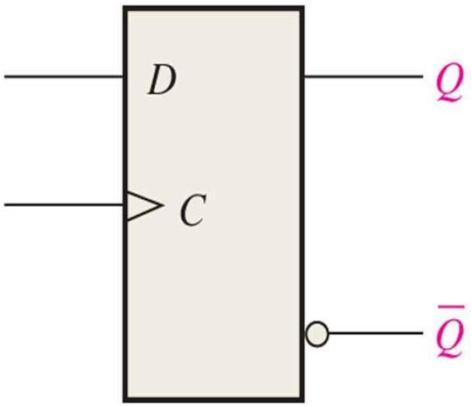
(b) Negative-edge triggered

↑= clock transition LOW to HIGH

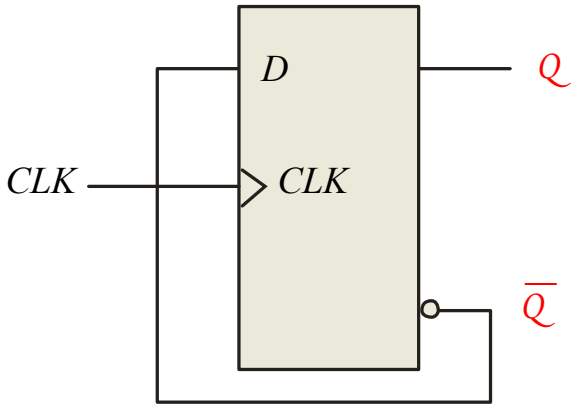
# Ex: Positive-edge triggered D Flip-Flop

- Determine the Q output waveform if the flip-flop starts out RESET



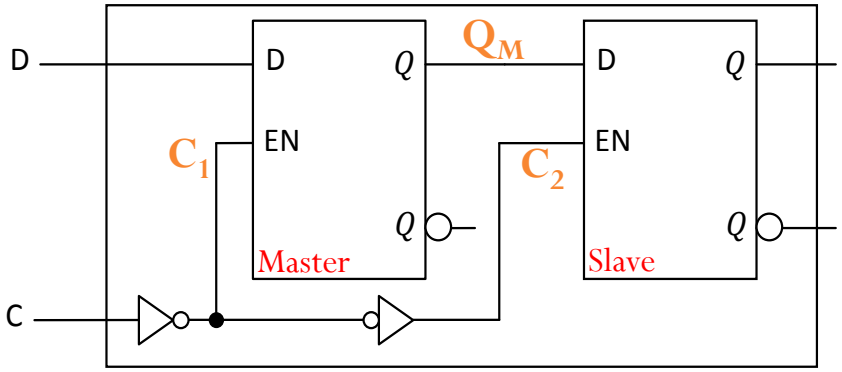



# Exercise: What is this?



# D Flip Flop: Implementation

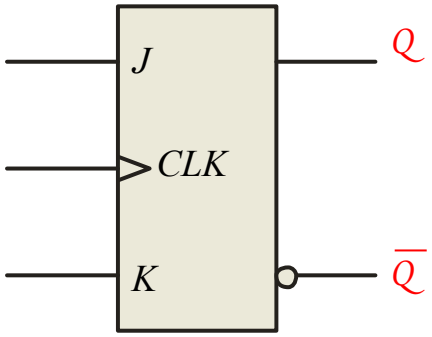
- Tie two D-latches together to make a D flip-flop




- When C is 0 ($C_1 = 1$), the master latch is open and follows the D input.
- When C is 1 ($C_1 = 0$, $C_2 = 1$), the master latch is closed and its output is transferred to the slave latch.
  - The slave latch is open all the while that C is 1, but changes only at the beginning of this interval, because the master is closed and unchanging during the rest of the interval.

# J-K Flip-Flop

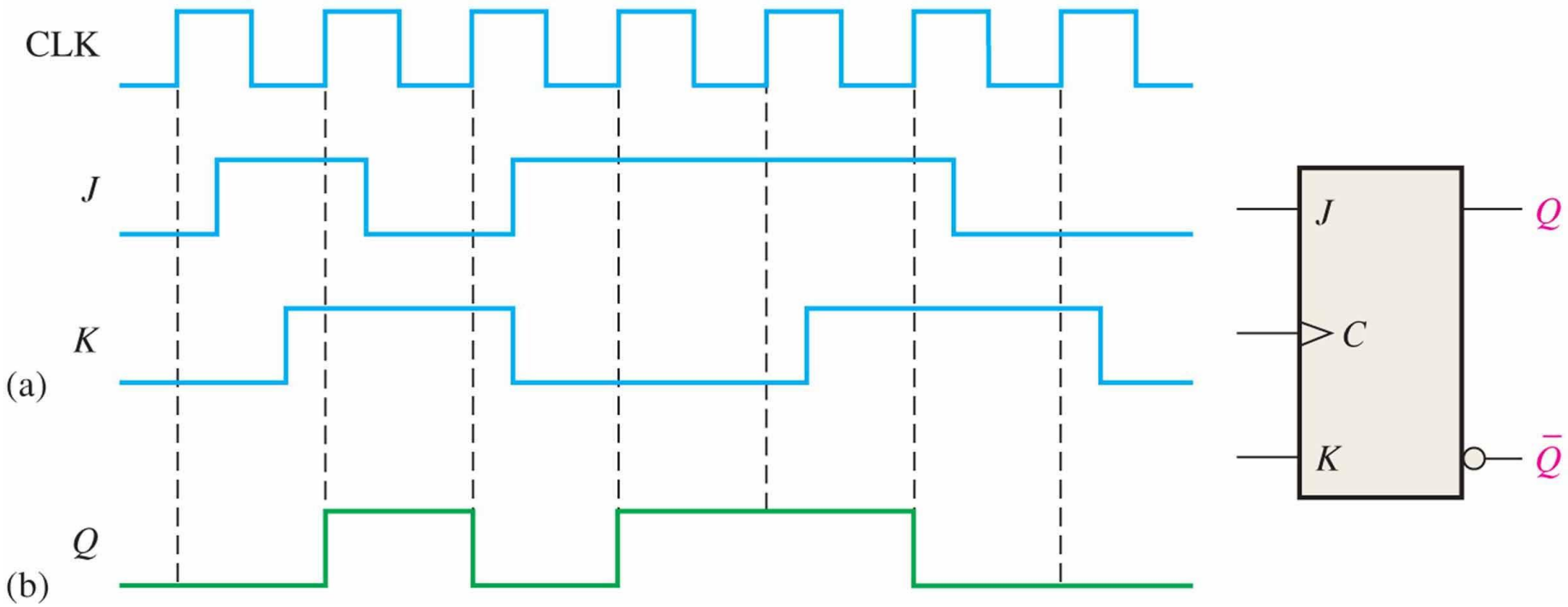
- Has two inputs, labeled J and K (along with the CLK).
- When both J and K = 1, the output changes states (toggles) on the rising clock edge.




| Inputs | | | Outputs | | |
|---|---|---|---|---|---|
| *J* | *K* | CLK | *Q* | $\overline{Q}$ | Comments |
| 0 | 0 | ↑ | $Q_0$ | $\overline{Q}_0$ | No change |
| 0 | 1 | ↑ | 0 | 1 | RESET |
| 1 | 0 | ↑ | 1 | 0 | SET |
| 1 | 1 | ↑ | $\overline{Q}_0$ | $Q_0$ | Toggle |

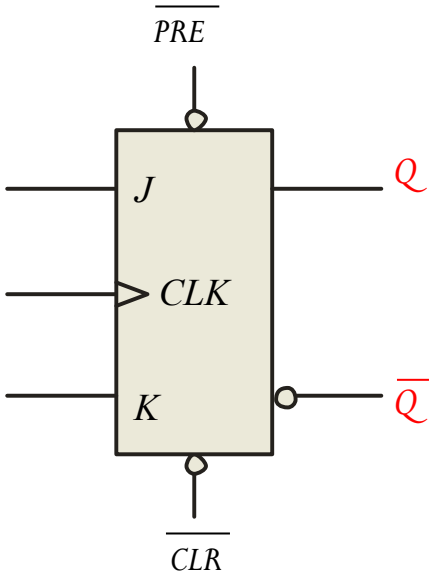
A J-K flip-flop connected for toggle operation is sometimes called a **T flip-flop**.

# Example: J-K Flip-Flop

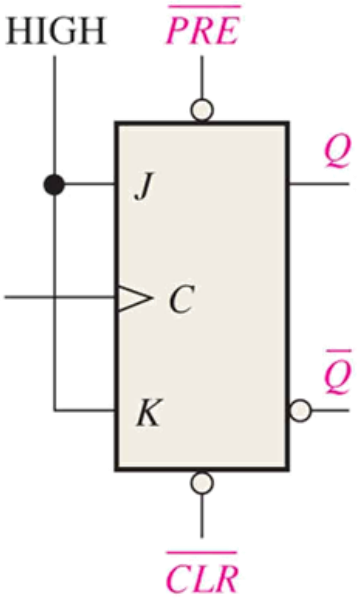


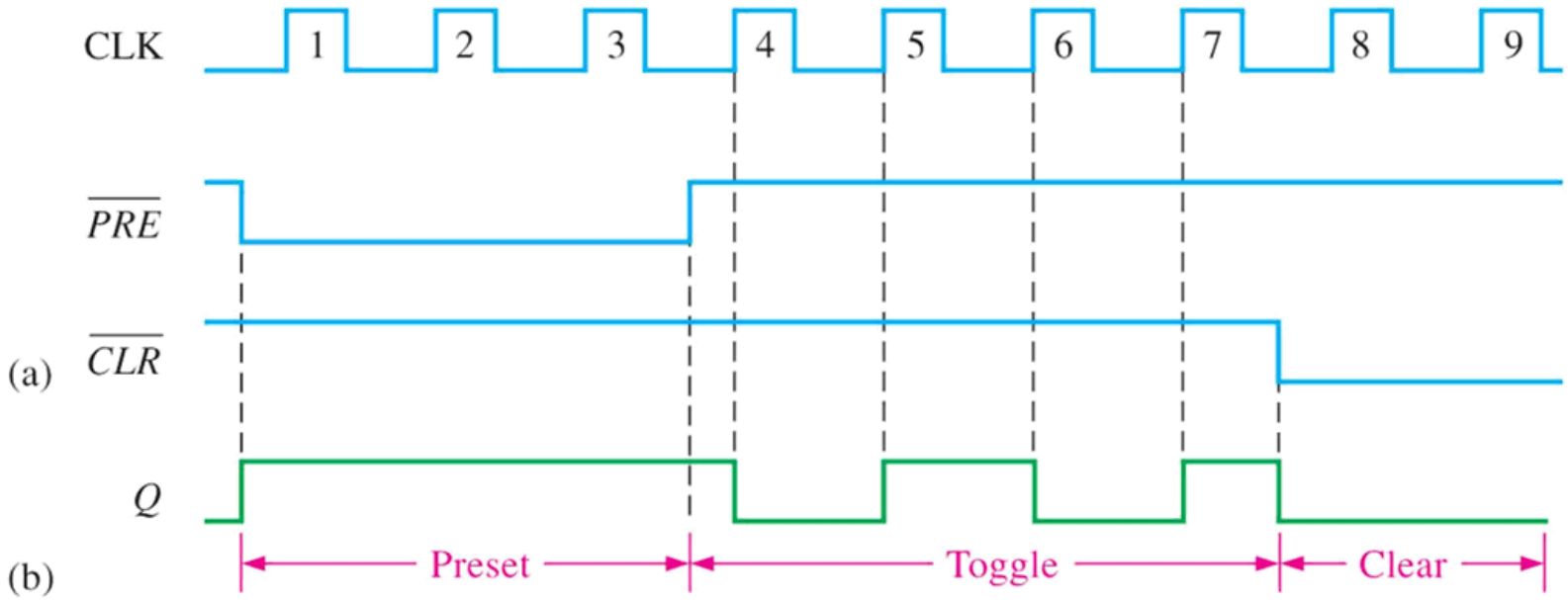
# Asynchronous Inputs

- Most flip-flops have other inputs that are *asynchronous*, meaning they affect the output independent of the clock.
- Two such inputs are normally labeled **preset (PRE)** and **clear (CLR)**.
- These inputs are usually active-LOW.
- A J-K flip flop with active-LOW preset and CLR is shown.

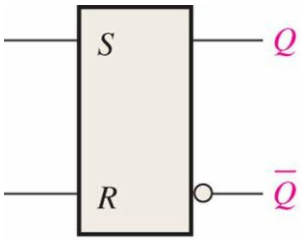


# Example

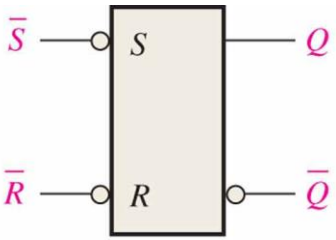

HIGH
$\overline{PRE}$
Q
J
C
K
$\overline{Q}$
$\overline{CLR}$


CLK
1
2
3
4
5
6
7
8
9
$\overline{PRE}$
$\overline{CLR}$
(a)
Q
(b)
Preset
Toggle
Clear

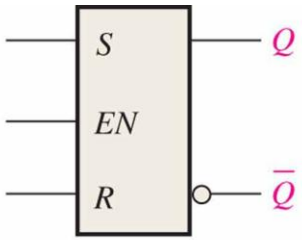
# Logic Symbols: Latches and Flip-Flops



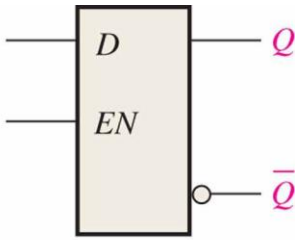

(a) Active-HIGH input S-R latch



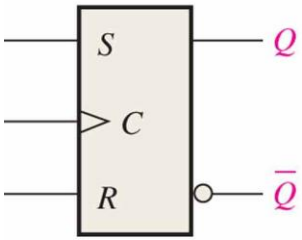

(b) Active-LOW input $\overline{S}$-$\overline{R}$ latch

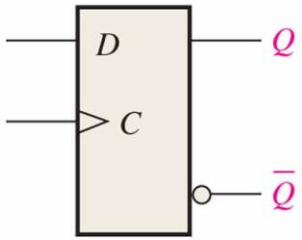


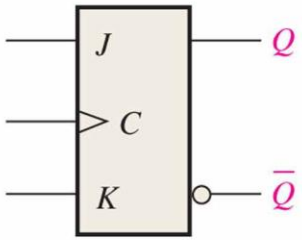

(c) Gated S-R latch

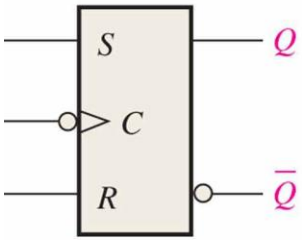



(d) Gated D latch

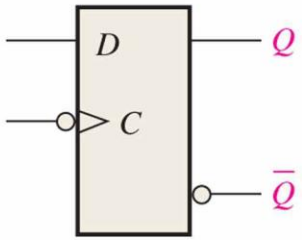


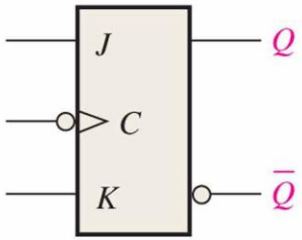




(e) S-R edge-triggered flip-flops




(f) D edge-triggered flip-flops



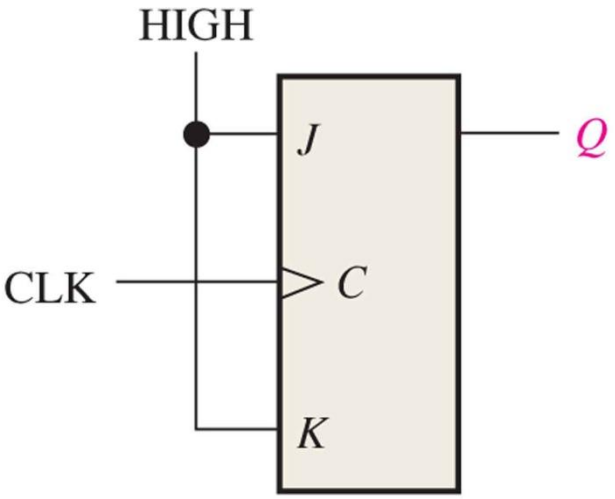

(g) J-K edge-triggered flip-flops

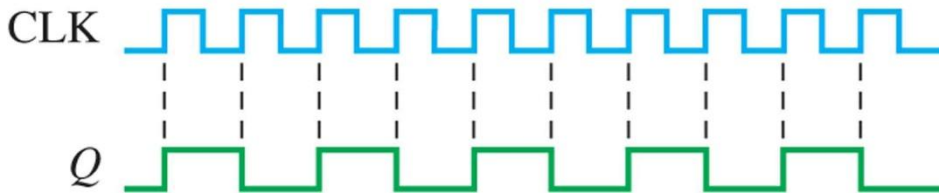
# Latches and Flip-Flops

- 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 the various types of latches and flip-flops are the number of inputs the process and the manner in which the inputs affect the binary state.
- The most basic storage elements are latches, from which flip-flops are usually constructed.
- Although latches are most often used within flip-flops, they can also be used with more complex clocking methods to implement sequential circuits directly.
  - The design of such circuits is, however, beyond the scope of this class.

# Some Applications

- Divide the clock frequency by 2



# Some Applications

- Divide the clock frequency by 4