

# Digital Circuits

ECS 371

**Dr. Prapun Suksompong**

[prapun@siit.tu.ac.th](mailto: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**

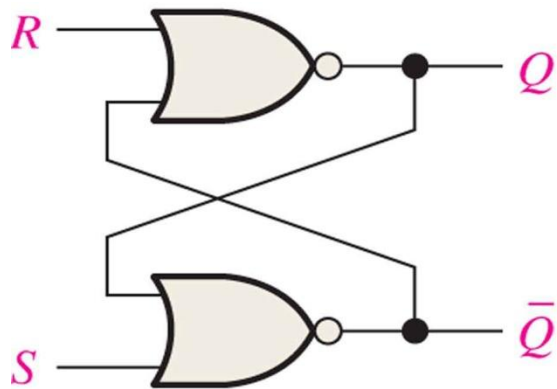


# Announcement

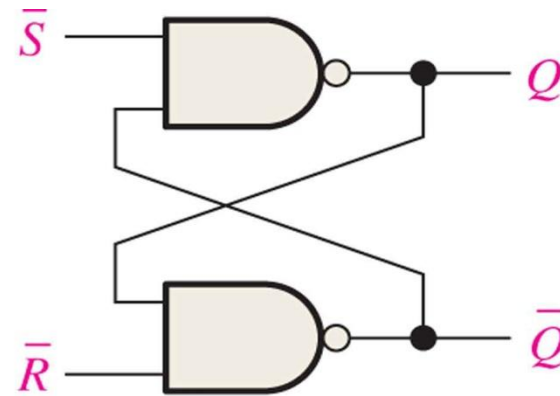
- Reading Assignment:
  - Chapter 7: 7-1, 7-2, 7-4
- When the grading is done (Monday?), your midterm scores will be posted on the course web site.

# S-R Latch

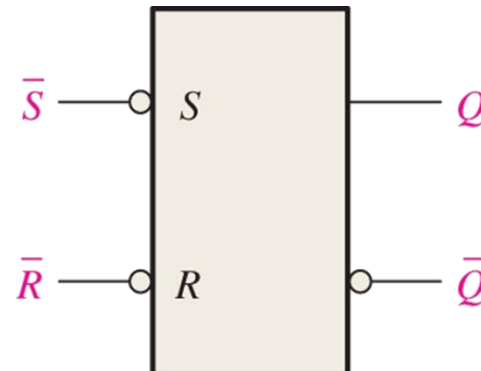
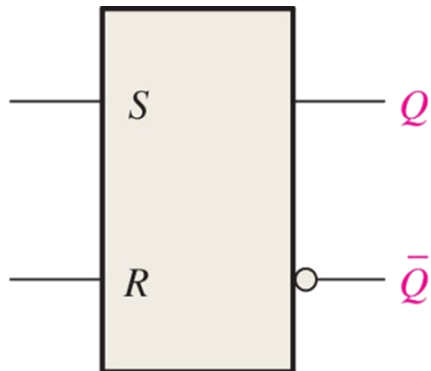
- There are two versions of SET-RESET (S-R) latches.



(a) Active-HIGH input S-R latch

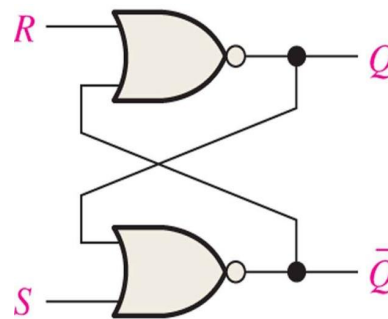


(b) Active-LOW input  $\bar{S}$ - $\bar{R}$  latch

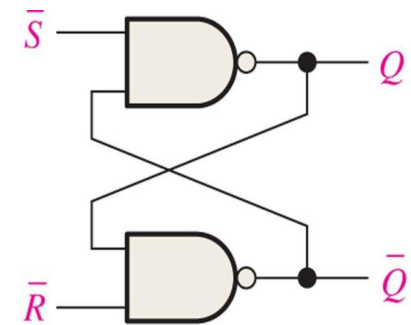


# S-R Latch (Remember This!)

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

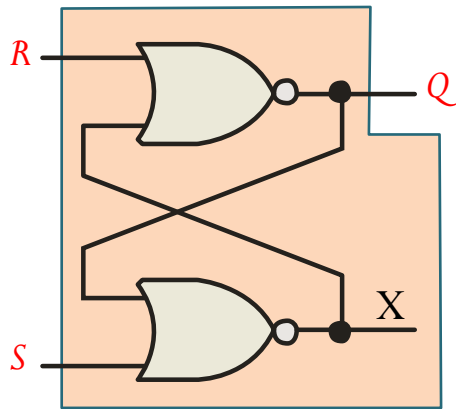


(a) Active-HIGH input S-R latch



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

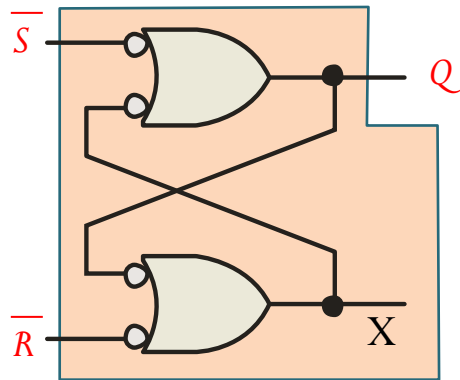
# The “Old Q”-“New Q” Analysis



$$\begin{aligned} Q_{new} &= \overline{R + X} \\ &= \overline{R + Q_{old} + S} \\ &= \overline{R} \cdot (Q_{old} + S) \end{aligned}$$

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

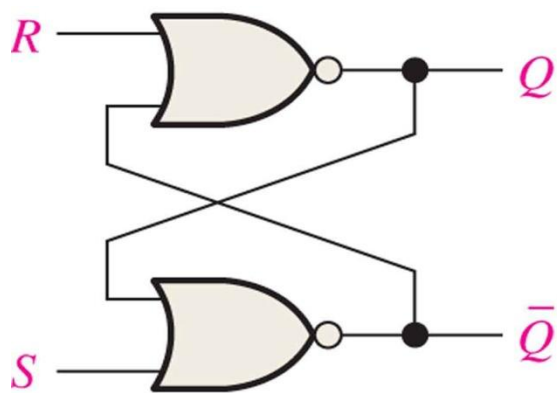
# The “Old Q”-“New Q” Analysis (2)



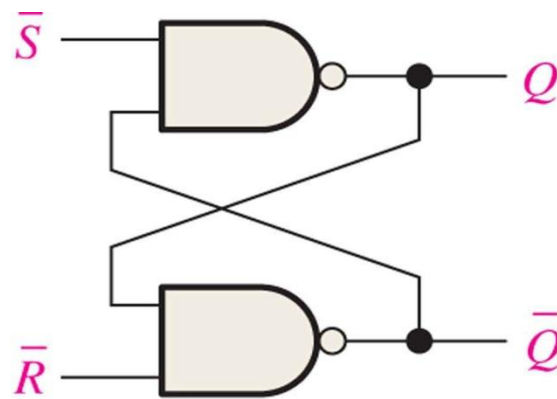
$$\begin{aligned}
 Q_{new} &= \overline{\overline{S}} + \overline{X} \\
 &= \overline{\overline{S}} + \overline{\overline{Q_{old} + R}} \\
 &= \overline{\overline{S}} + Q_{old} \cdot \overline{R}
 \end{aligned}$$

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

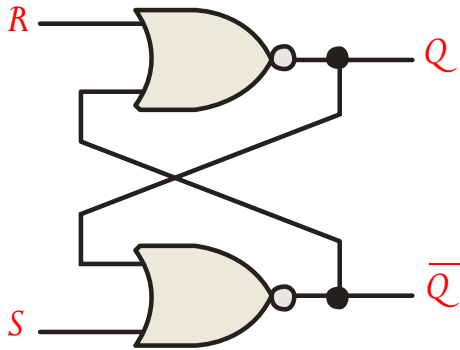


(b) Active-LOW input  $\bar{S}$ - $\bar{R}$  latch

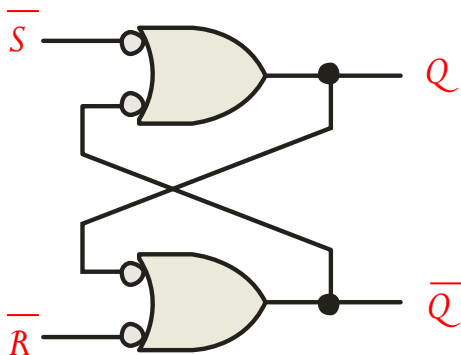
Input		Output
S	R	$Q_{\text{new}}$
0	0	$Q_{\text{old}}$
0	1	0
1	0	1
1	1	0

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

# Expanded Version



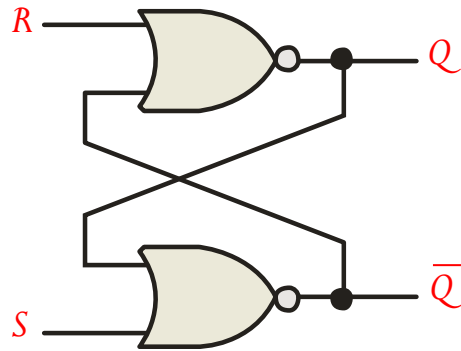
Inputs		Outputs		Mode of Operation	Comment
$S$	$R$	$Q$	$\bar{Q}$		
0	0	NC	NC	Hold	No change.
0	1	0	1	Reset	For RESETting Q to 0
1	0	1	0	Set	For SETting Q to 1
1	1	0	0	Prohibited	Invalid Condition



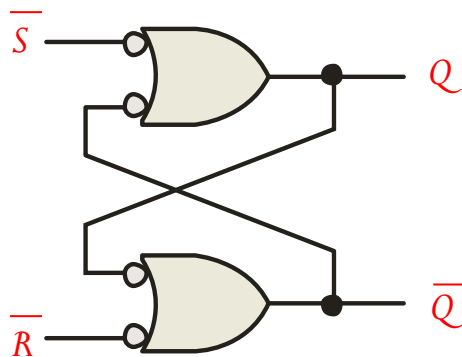
Inputs		Outputs		Mode of Operation	Comment
$\bar{S}$	$\bar{R}$	$Q$	$\bar{Q}$		
0	0	1	1	Prohibited	Invalid Condition
0	1	1	0	Set	For SETting Q to 1
1	0	0	1	Reset	For RESETting Q to 0
1	1	NC	NC	Hold	No change.



# Short Version (Remember This!)



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



Inputs		Mode
$\bar{S}$	$\bar{R}$	
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

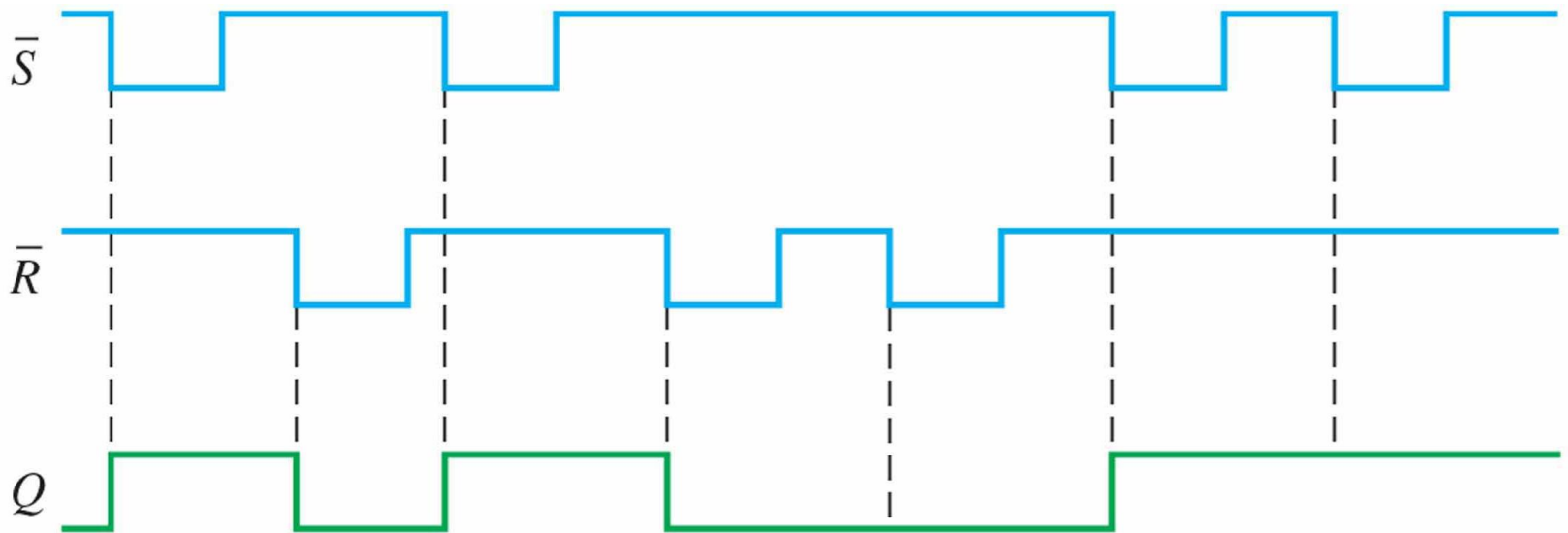
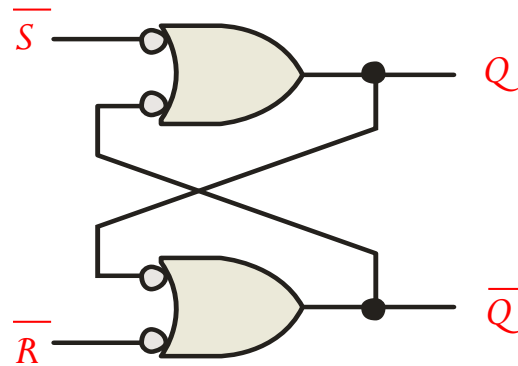
- If a 1 is applied to both the inputs of the latch, both outputs go to 0.
- This produces an undefined state.
- It results in an indeterminate or unpredictable next state when both inputs return to 0 simultaneously.
- In normal operation, these problems are avoided by making sure that 1's are not applied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

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

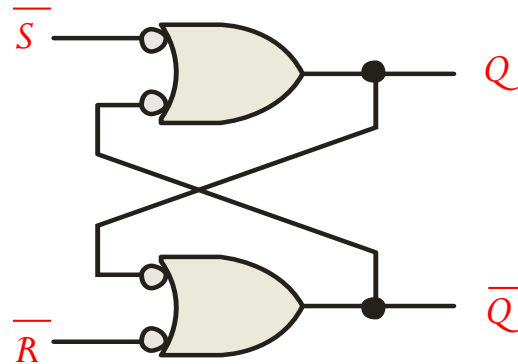
Inputs		Mode
$\overline{S}$	$\overline{R}$	
0	1	SET
1	0	RESET
1	1	HOLD

- 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  $\overline{S}$  input must go back to **1** before  $\overline{R}$  is changed to **0** to avoid occurrence of the undefined state.
  - Applying a **1** to  $\overline{S}$  with  $\overline{R} = 1$  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  $\overline{R}$ , and the circuit remains in the reset state.

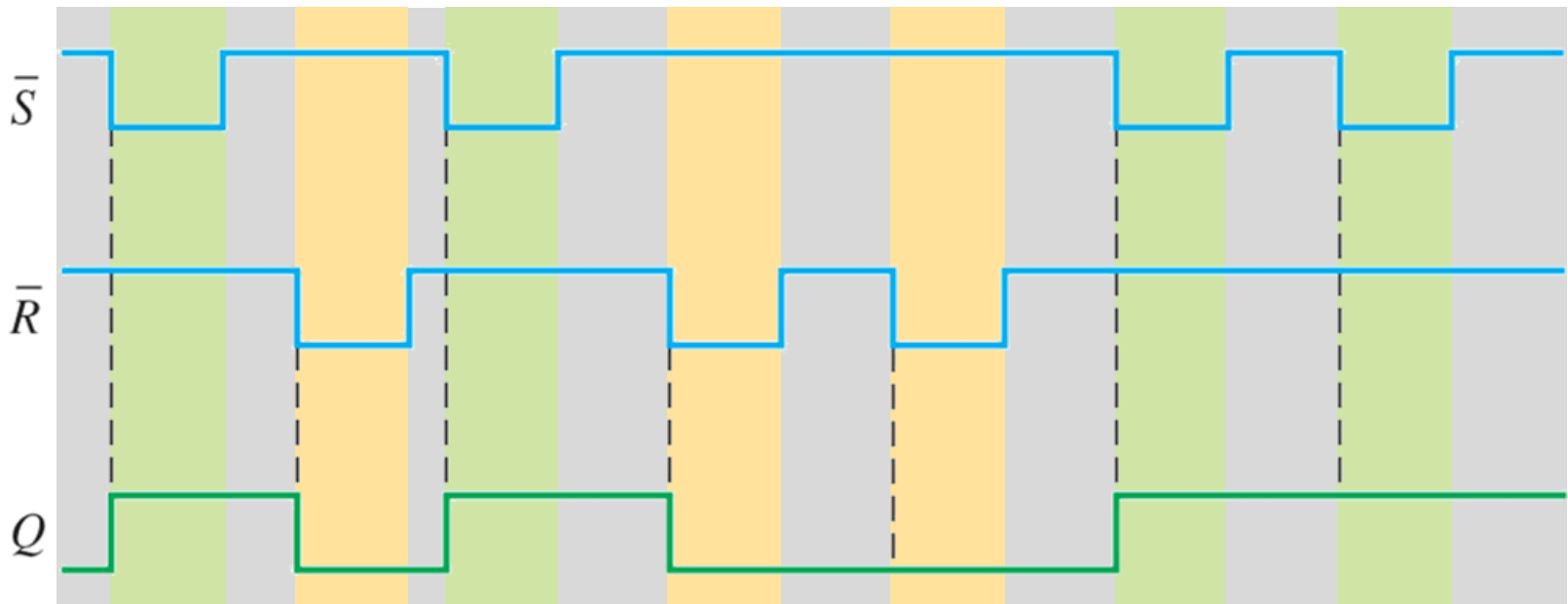
# Example



# Example

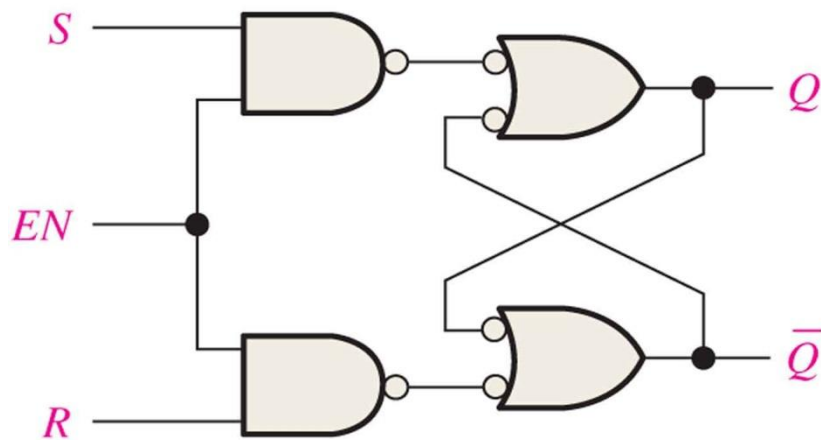


Input s		Mode
$\bar{S}$	$\bar{R}$	
0	1	SET
1	0	RESET
1	1	HOLD

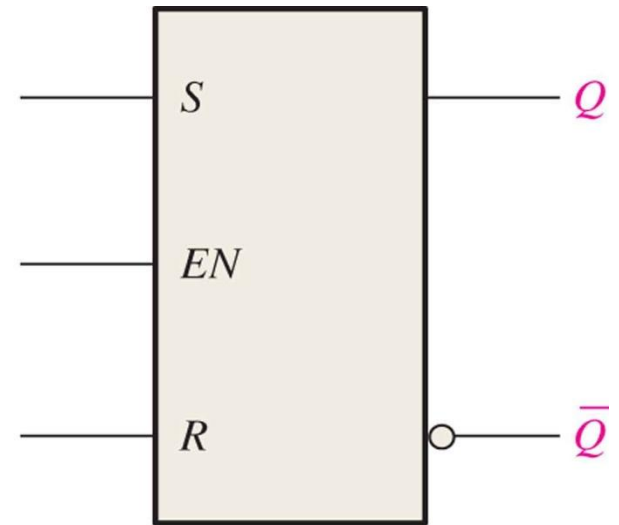


# Gated Latch

- 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

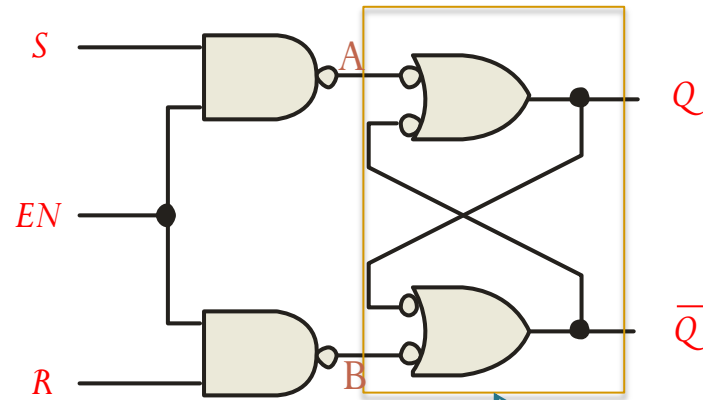
# Gated Latch

Observe that:

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

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

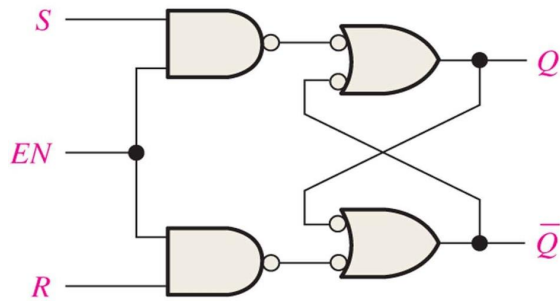
EN	A	B
0 $\Rightarrow$	1	1
1 $\Rightarrow$	$\overline{S}$	$\overline{R}$



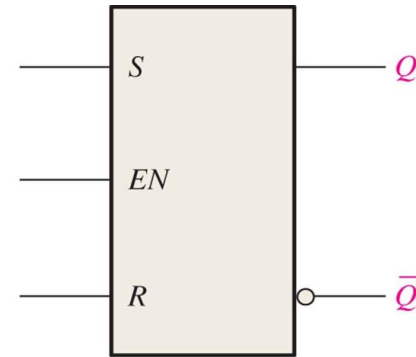
This is the same as the active-LOW input latch!



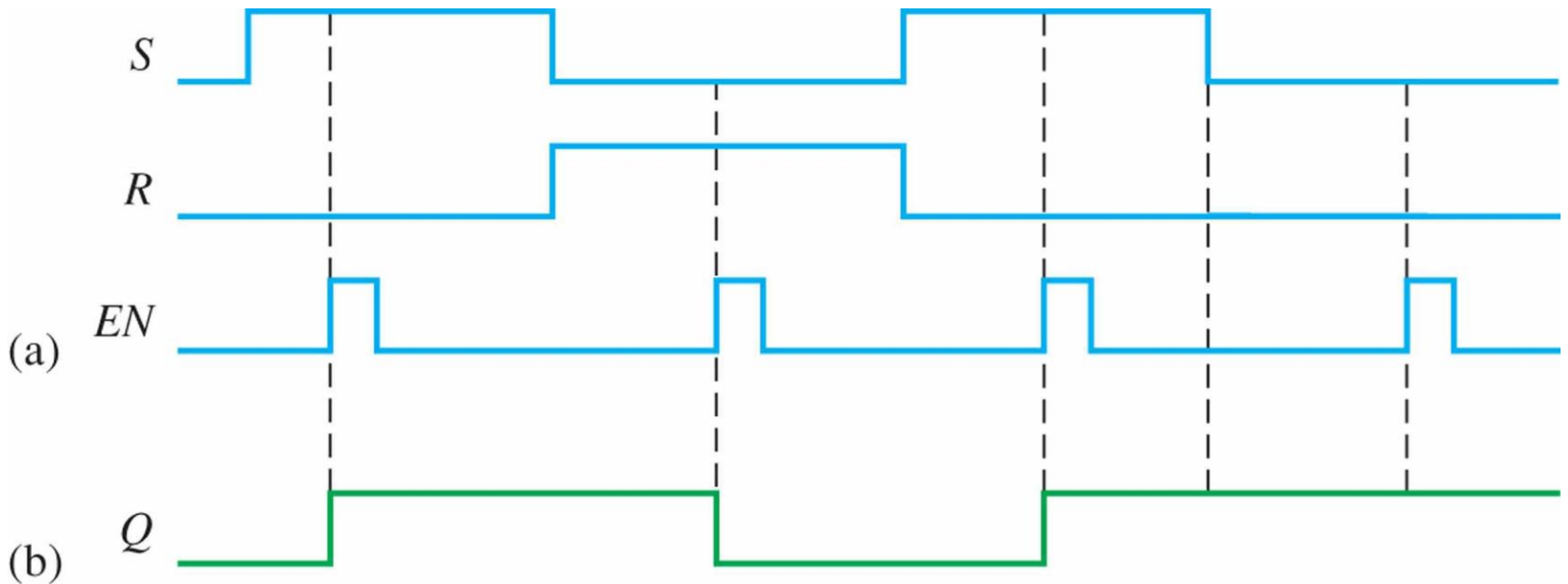
# Example: Gated S-R Latch



(a) Logic diagram

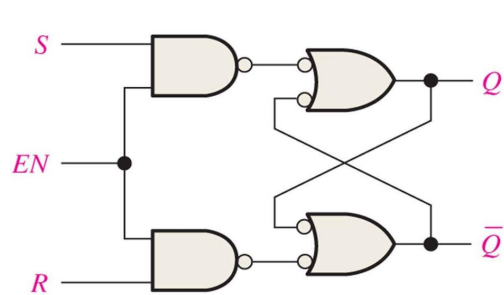
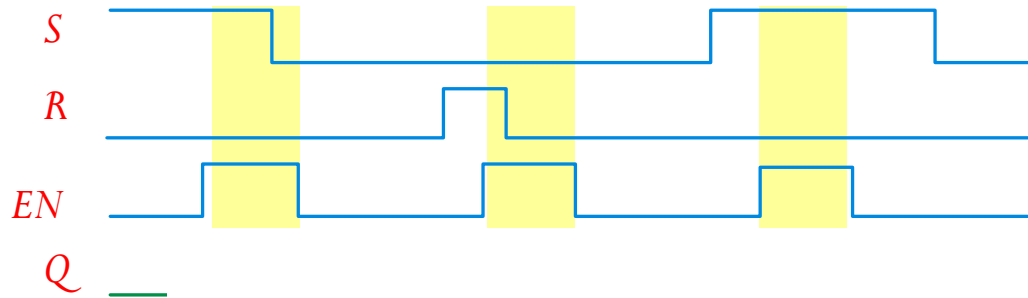


(b) Logic symbol

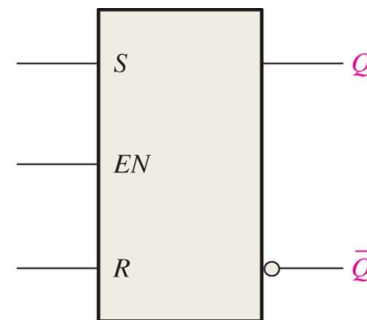


# Exercise: Gated S-R Latch

Show the Q output with relation to the input signals. Assume Q starts LOW.



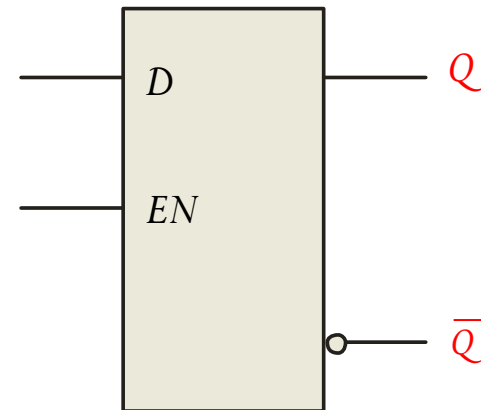
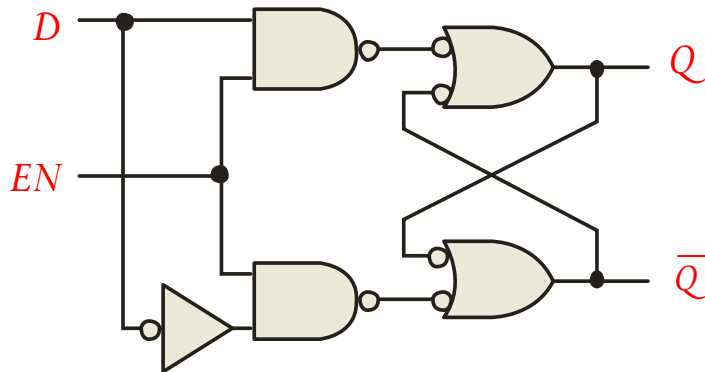
(a) Logic diagram



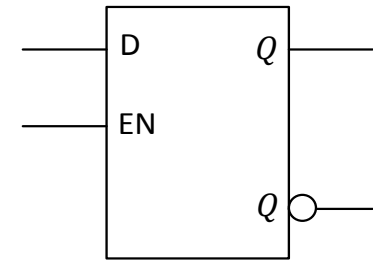
(b) Logic symbol

# 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



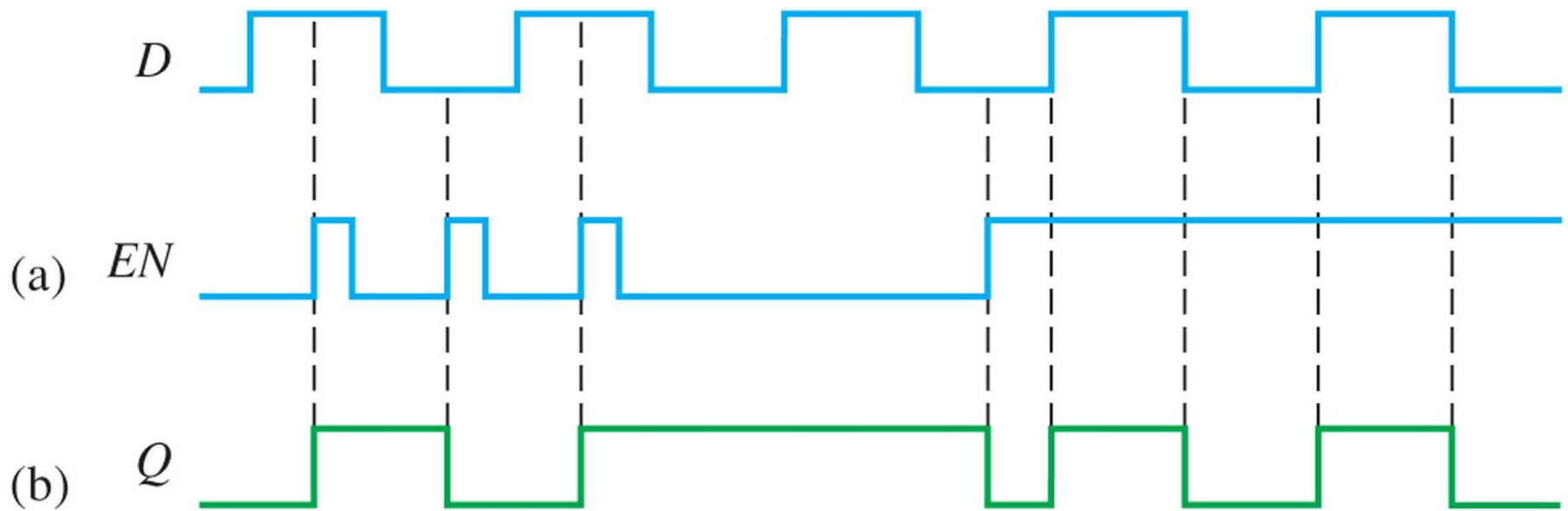
- A simple rule for the D latch is:
  - Q follows D when the Enable is active/asserted.
    - In this situation, the latch is said to be “open” and the path from D input to Q output is “transparent”.
    - The circuit is often called a transparent latch for this reason.
- When EN is LOW, the state of 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:  $\longrightarrow$

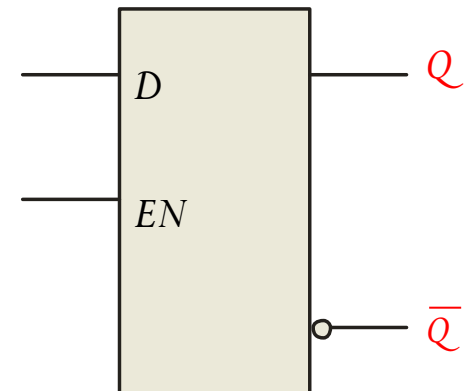
$Q_0$  is the prior output level before the indicated input conditions were established.

Inputs		Outputs		Comments
$D$	$EN$	$Q$	$\bar{Q}$	
0	1	0	1	RESET
1	1	1	0	SET
X	0	$Q_0$	$\bar{Q}_0$	No change

# Example: Gated D Latch



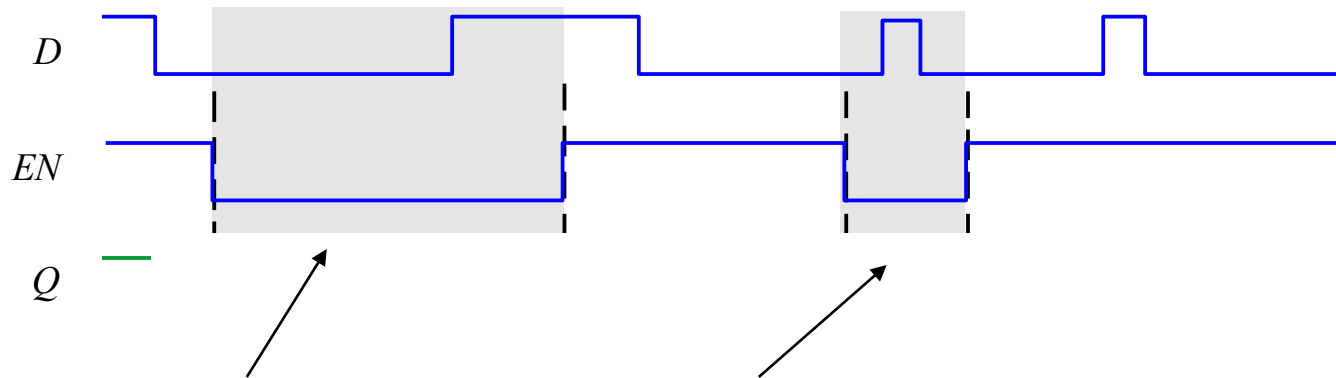
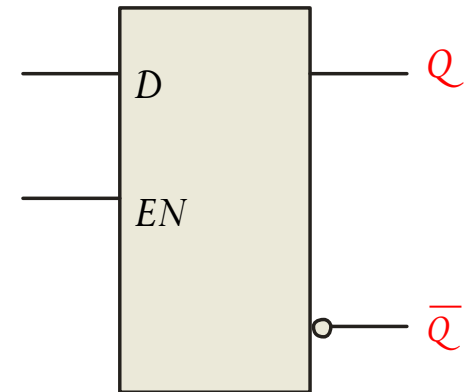
Q follows D when the Enable is active.





# Exercise: Gated D Latch

Determine the Q output for the D latch, given the inputs shown.



Notice that the Enable is not active during these times, so the output is latched.

# Latch: some final words....

- **Latch** = A *bistable* digital circuit used for storing a bit.
- **RESET**
  1. The state of a latch when the output (Q) is 0.
  2. The action of producing a RESET state.
- **SET**
  1. The state of a latch when the output (Q) is 1.
  2. The action of producing a SET state.

# Flip-Flop

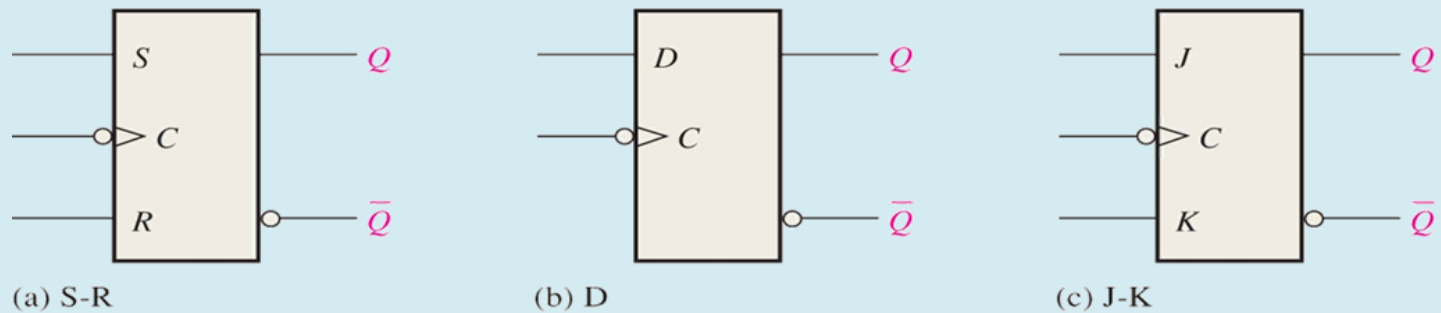
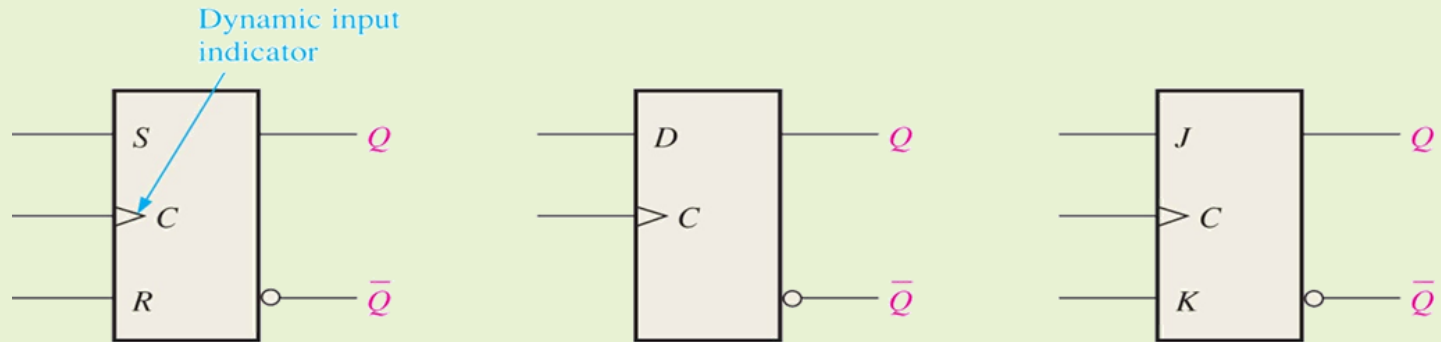
- Latches sample their inputs (and change states) any time the EN bit is asserted
- Many times we want more control over when to sample the input
- A **flip-flop** differs from a latch in the manner it changes states.
- A flip-flop is a *clocked* device.
- Flip-flops are **synchronous**: the output changes 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)

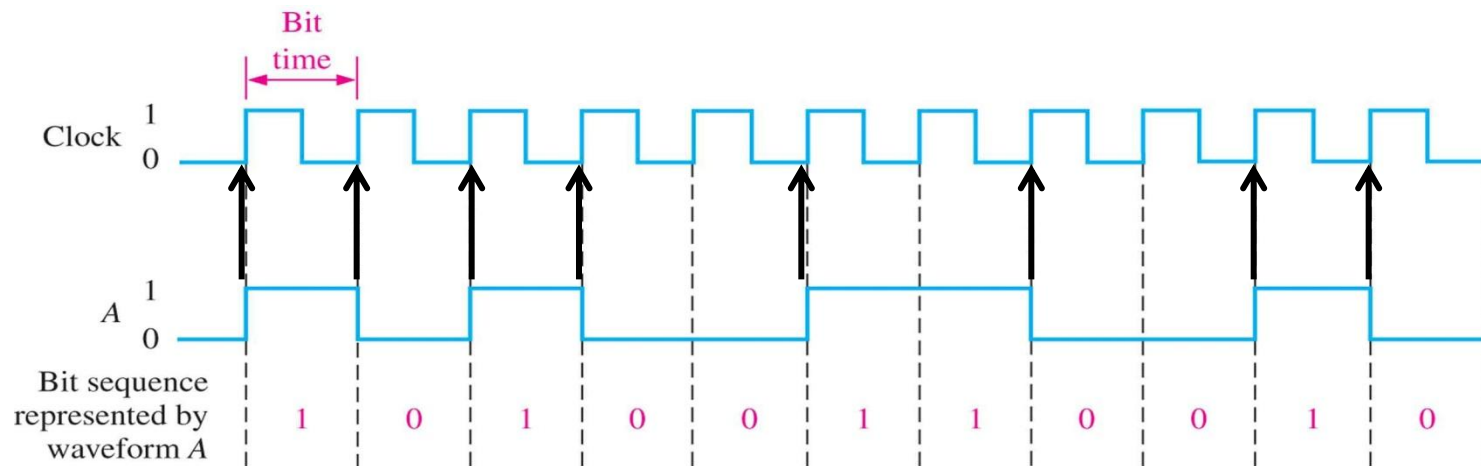
Positive edge-triggered (no bubble at C input)



Negative edge-triggered (bubble at C input)

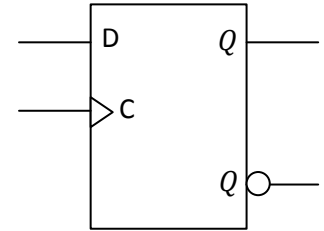
# Clock (CLK)

- In digital synchronous systems, all waveforms are synchronized with a clock.
  - The clock waveform itself does not carry information.
- The clock is a periodic waveform in which each interval between pulses (the period) equals the time for one bit.



- Notice that change in level of waveform *A* occurs at the **leading edge** of the clock waveform.

# 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		Comments
D	CLK	Q	$\bar{Q}$	
1	↑	1	0	SET
0	↑	0	1	RESET

(a) Positive-edge triggered

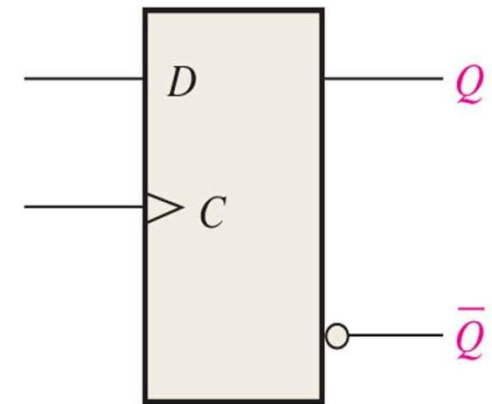
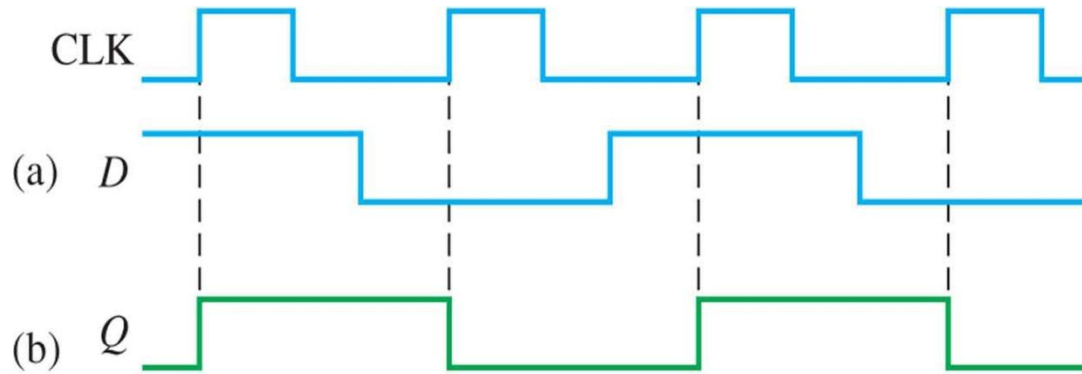
Inputs		Outputs		Comments
D	CLK	Q	$\bar{Q}$	
1	↓	1	0	SET
0	↓	0	1	RESET

(b) Negative-edge triggered

↑ = clock transition LOW to HIGH

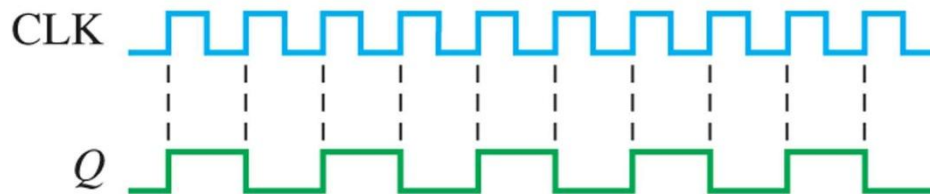
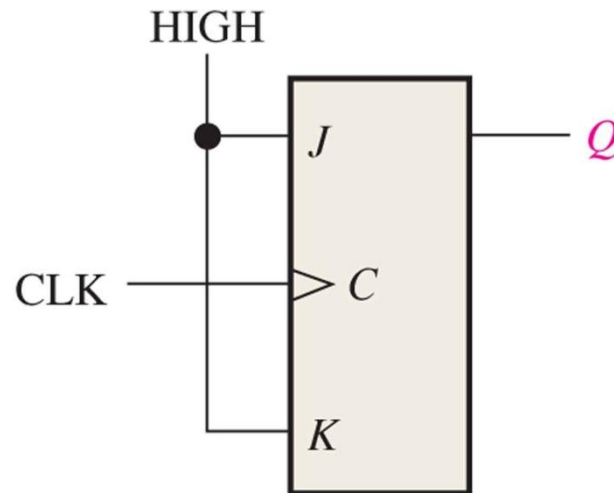
# Ex: Positive-edge triggered D Flip-Flop

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



# Some Applications

- Divide the clock frequency by 2



# Some Applications

- Divide the clock frequency by 4

