Digital Circuits

ECS 371

Dr. Prapun Suksompong

prapun@siit.tu.ac.th

Lecture 15

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

ECS371.PRAPUN.COM

Digital Logic Circuit Types

- Combinational Circuit
  - Output depends only on current inputs
  - No feedback loops
  - "memoryless"
- Sequential Circuit
  - Output depends on past history plus current inputs
  - Contains feedback loops
  - Has memory

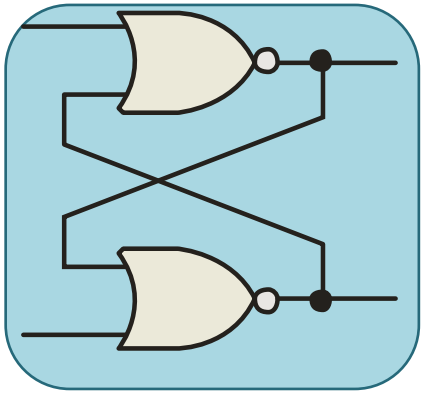
- Up to this point, we have focused on "**combinatorial logic circuits**" (i.e. the output of the circuit is dependent on the current input ONLY).
- Now we will shift our focus to "**sequential logic circuits**" (i.e. the output depends not only on the present input but also on the **history** of the input.
- The basic building blocks for sequential logic circuits are "**latches**" and "**flip-flops**"

Sequential Logic Circuits

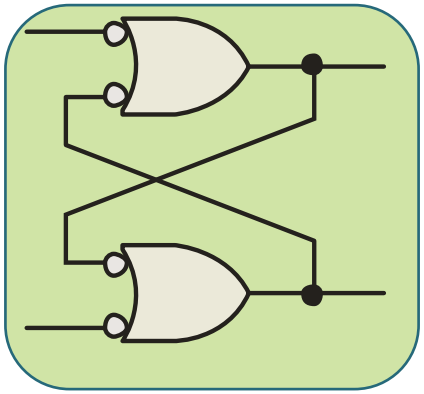
- Memory is represented in the form of states.
- "State" embodies all the information about the past needed to predict current output based on current input.
- State variables are one or more bits of information representing logic signals in a circuit
  - Tell you "where the circuit is"
  - Used in conjunction with inputs to derive current outputs of a sequential circuit
- In combinational circuits, only need to look at the current inputs to get the current output.

S-R Latch

- A **latch** is a temporary storage device that has two stable states (bistable). It is a basic form of memory.
- The **S-R (Set-Reset) latch** is the most basic type.
  - It can be constructed from NOR gates or NAND gates.
  - With two cross-coupled **NOR gates**, the latch responds to **active-HIGH inputs**.

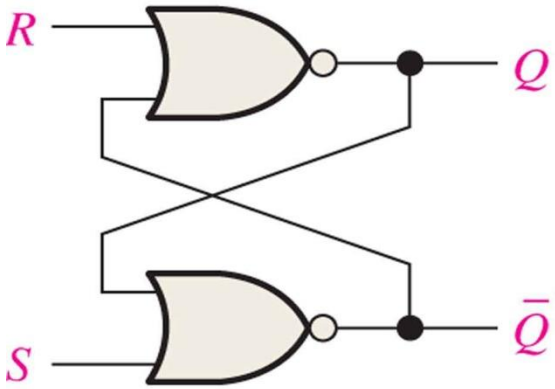


  - With two cross-coupled **NAND gates**, the latch responds to **active-LOW inputs**.

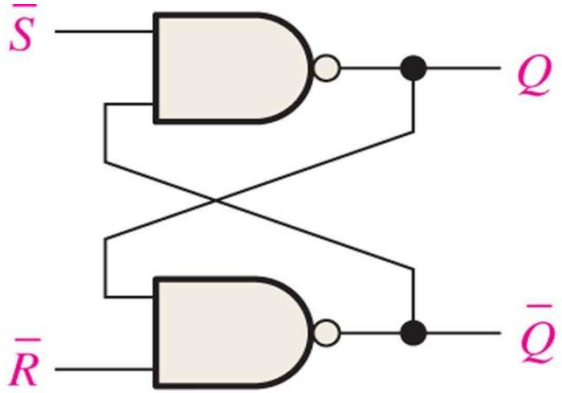


S-R Latch

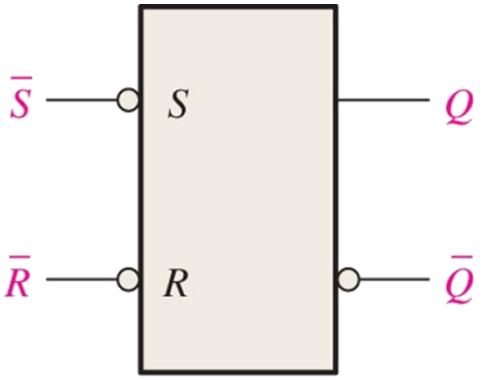
- There are two versions of SET-RESET (S-R) latches.

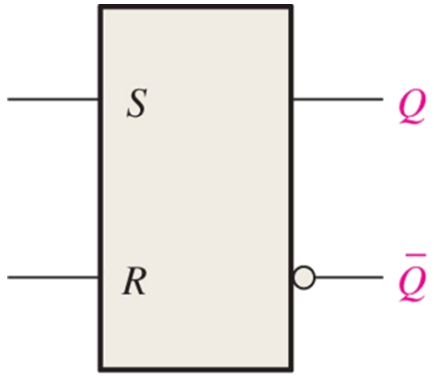


(a) Active-HIGH input S-R latch

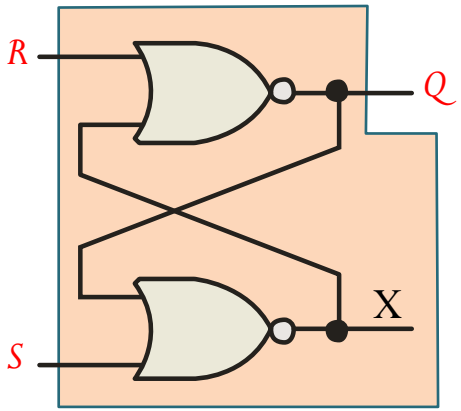


(b) Active-LOW input $\bar{S}$-$\bar{R}$ latch





The “Old Q”-“New Q” Analysis



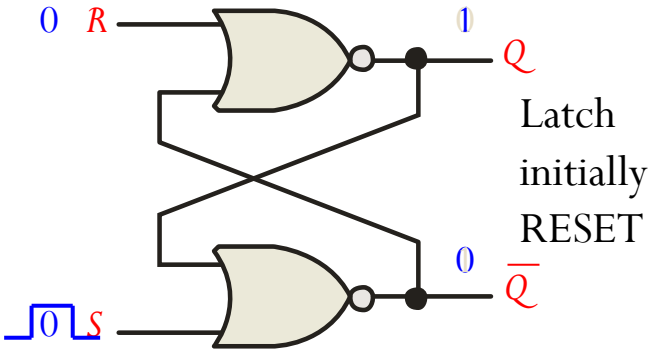
$$Q_{new} = \overline{R + X}$$
$$= \overline{R + \overline{\overline{Q_{old} + S}}}$$
$$= \overline{R} \cdot \left(Q_{old} + S \right)$$

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

Active-LOW S-R latch

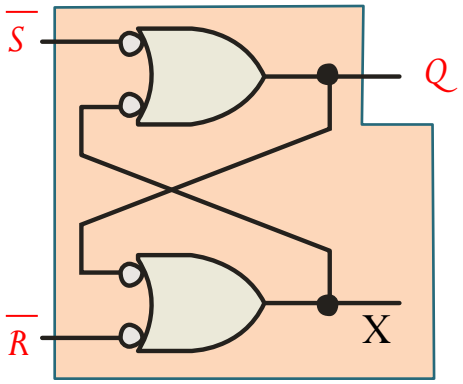
Input		Output
S	R	Q_{new}
0	0	Q_{old}
0	1	0
1	0	1
1	1	0

Assume the latch is initially RESET ($Q = 0$) and the inputs are at their inactive level (0). To SET the latch ($Q = 1$), a momentary HIGH signal is applied to the *S* input while the *R* remains LOW.



To RESET the latch ($Q = 0$), a momentary HIGH signal is applied to the *R* input while the *S* remains LOW.

The “Old Q”-“New Q” Analysis (2)



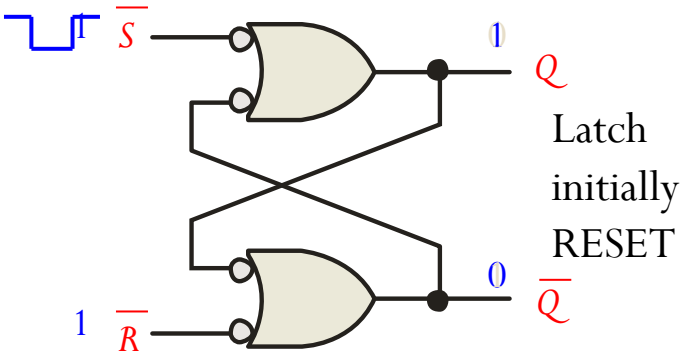
$$Q_{new} = ?$$

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

Active-Low S-R latch

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

Assume the latch is initially RESET ($Q = 0$) and the inputs are at their inactive level (1). To SET the latch ($Q = 1$), a momentary LOW signal is applied to the S input while the R remains HIGH.

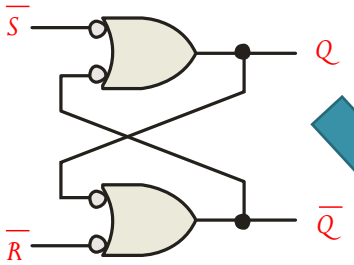


To RESET the latch a momentary LOW is applied to the R input while S is HIGH.

Never apply an active set and reset at the same time (invalid).

Summary: Active-LOW Input Latch

- Truth table for an active-LOW Input latch

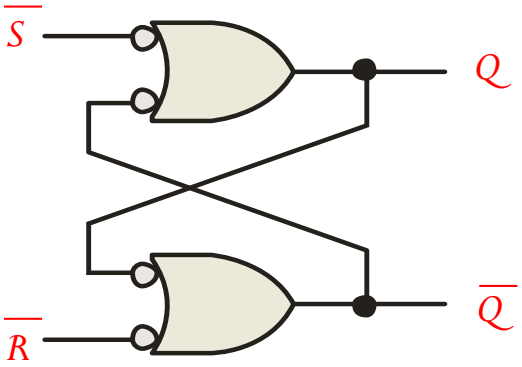


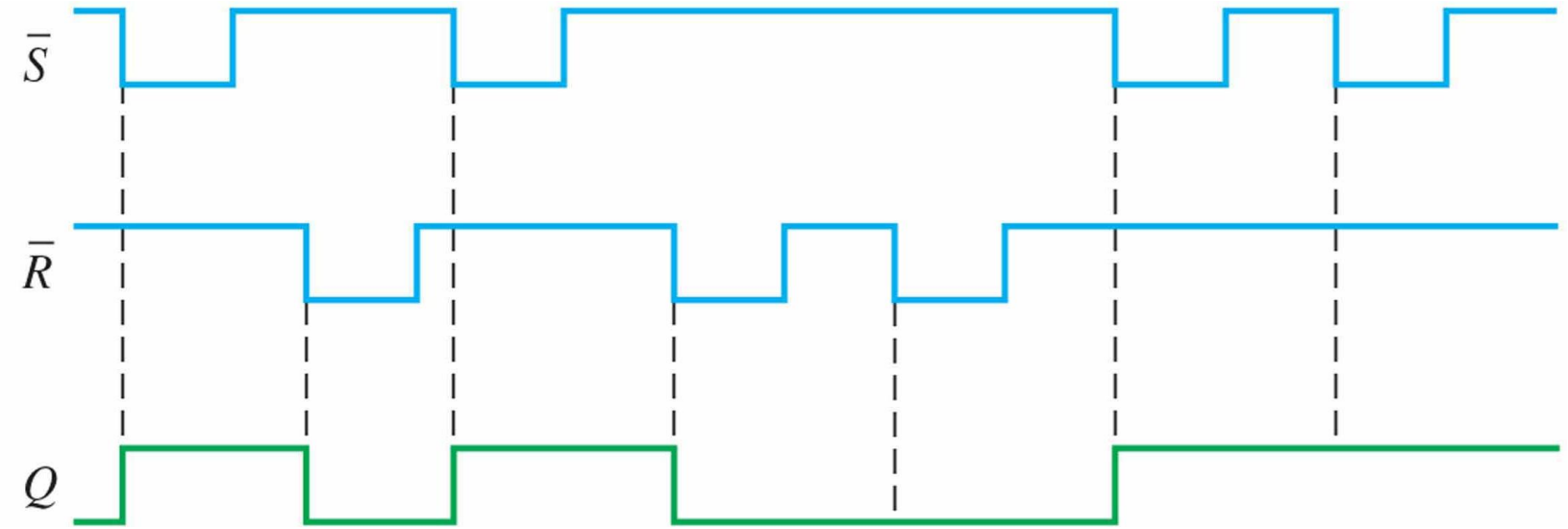
Input		Output
$\overline{S}$	$\overline{R}$	Q_{new}
0	0	1
0	1	1
1	0	0
1	1	Q_{old}

Input		Output		Comment
$\overline{S}$	$\overline{R}$	Q	$\overline{Q}$	
0	0	1		Invalid Condition
0	1	1	0	Latch SET
1	0	0	0	Latch RESET
1	1	NC	NC	No change.

- Condition $\overline{S} = \overline{R} = 0$ is avoided because it results in an invalid mode of operation and is a major drawback of any SET-RESET type of latch
- Operation of the active-HIGH input Latch is similar but requires the use of opposite logic levels.

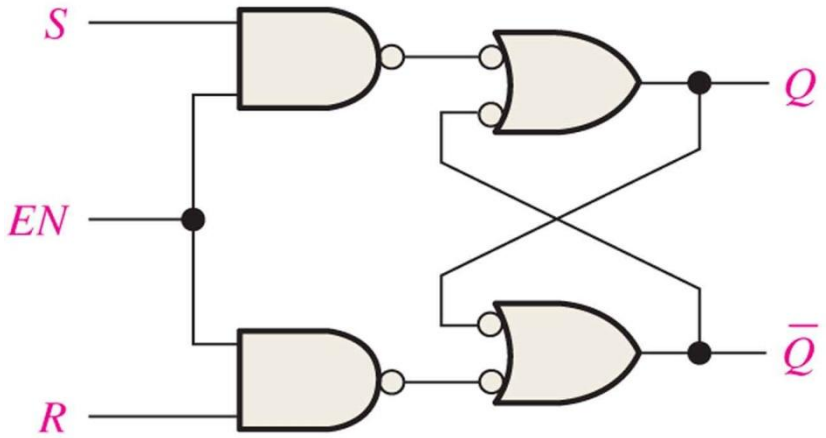
Example



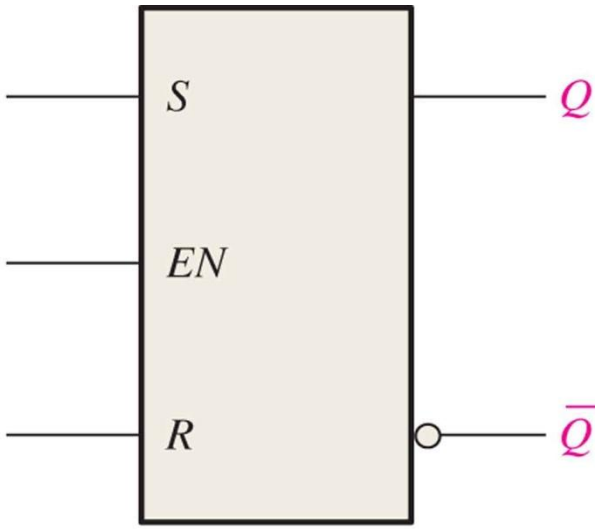


Gated Latch

- 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

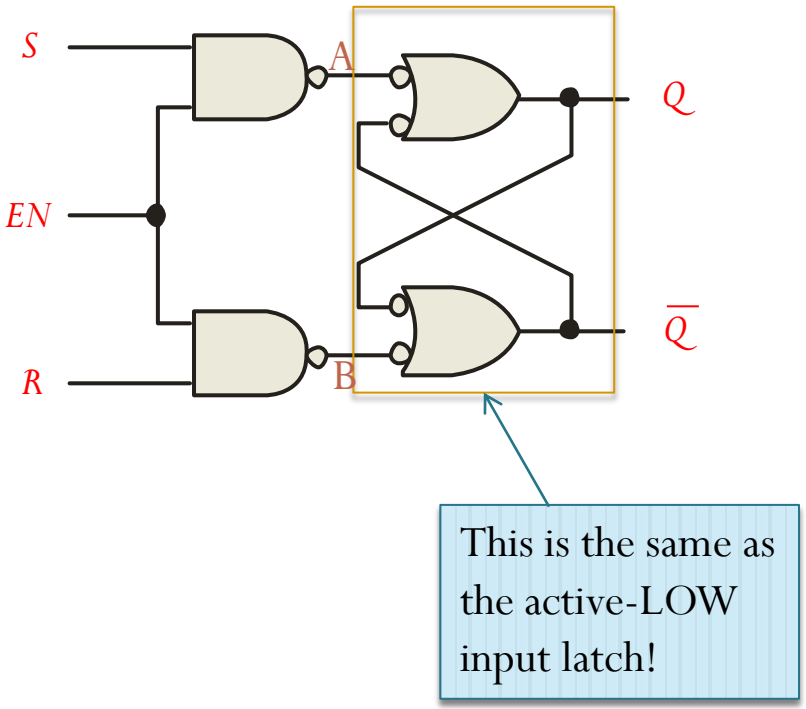
Gated Latch

Observe that:

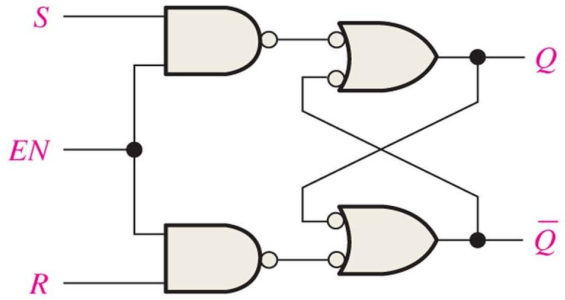
$$A = \overline{S \cdot EN} = \overline{S} + \overline{EN}$$

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

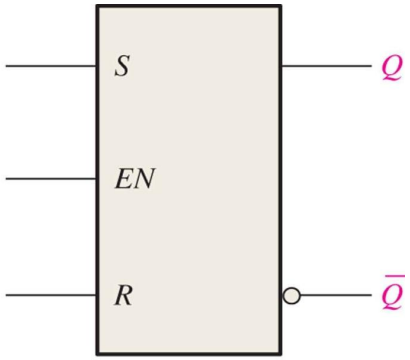
EN	A	B
0 ⇨	1	1
1 ⇨	$\overline{S}$	$\overline{R}$



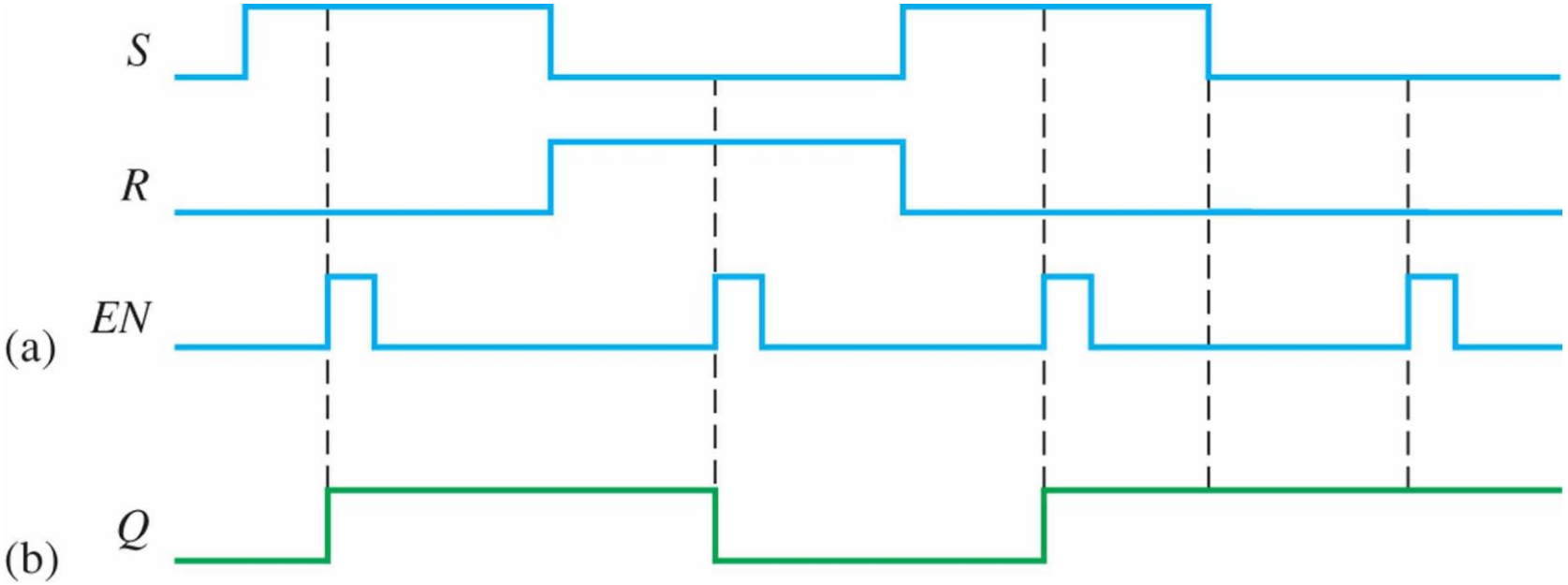
Example: Gated S-R Latch



(a) Logic diagram

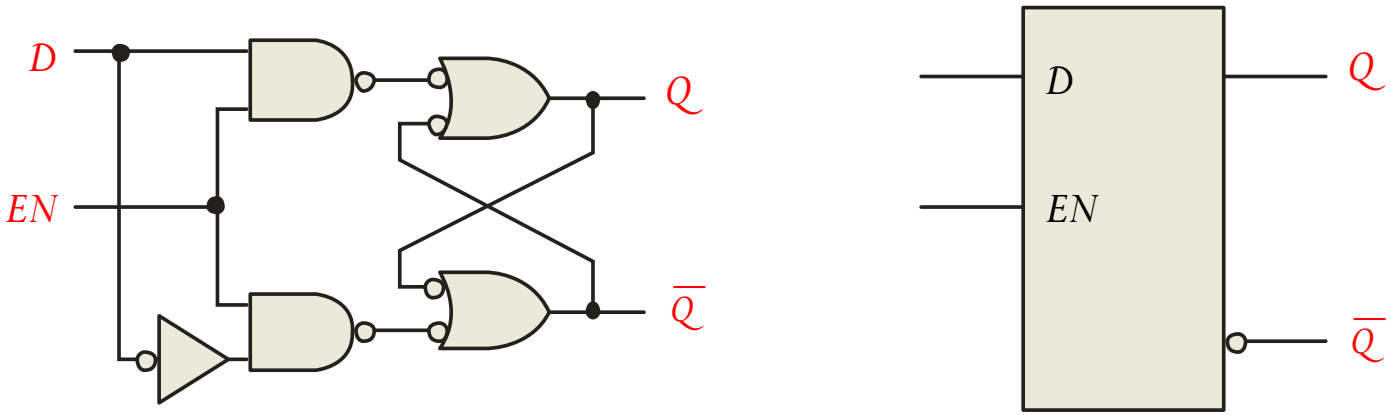


(b) Logic symbol



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.



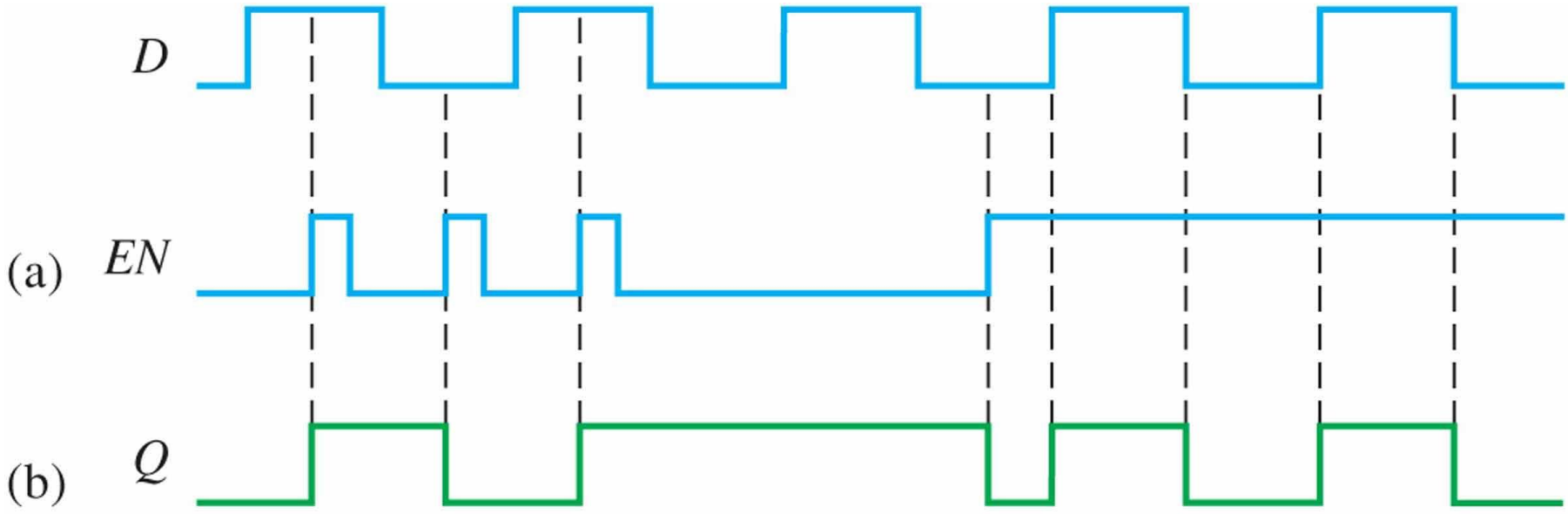
Gated D Latch: Operation

- A simple rule for the D latch is:
  - Q follows D when the Enable is active.
- When EN is LOW, the state of the latch is not affected by the D input.
- Output is "latched" at the last value when the enable signal becomes not asserted
- Truth Table:

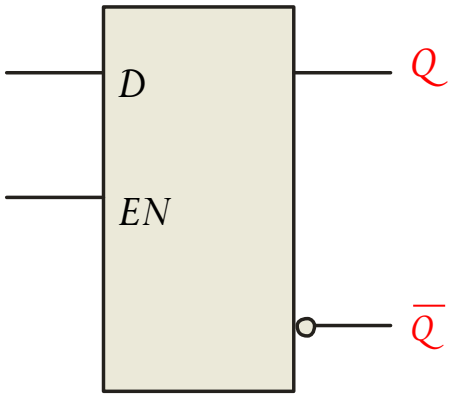
Inputs		Outputs		
D	*EN*	*Q*	$\bar{Q}$	Comments
0	1	0	1	RESET
1	1	1	0	SET
X	0	Q_0	$\bar{Q}_0$	No change

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.



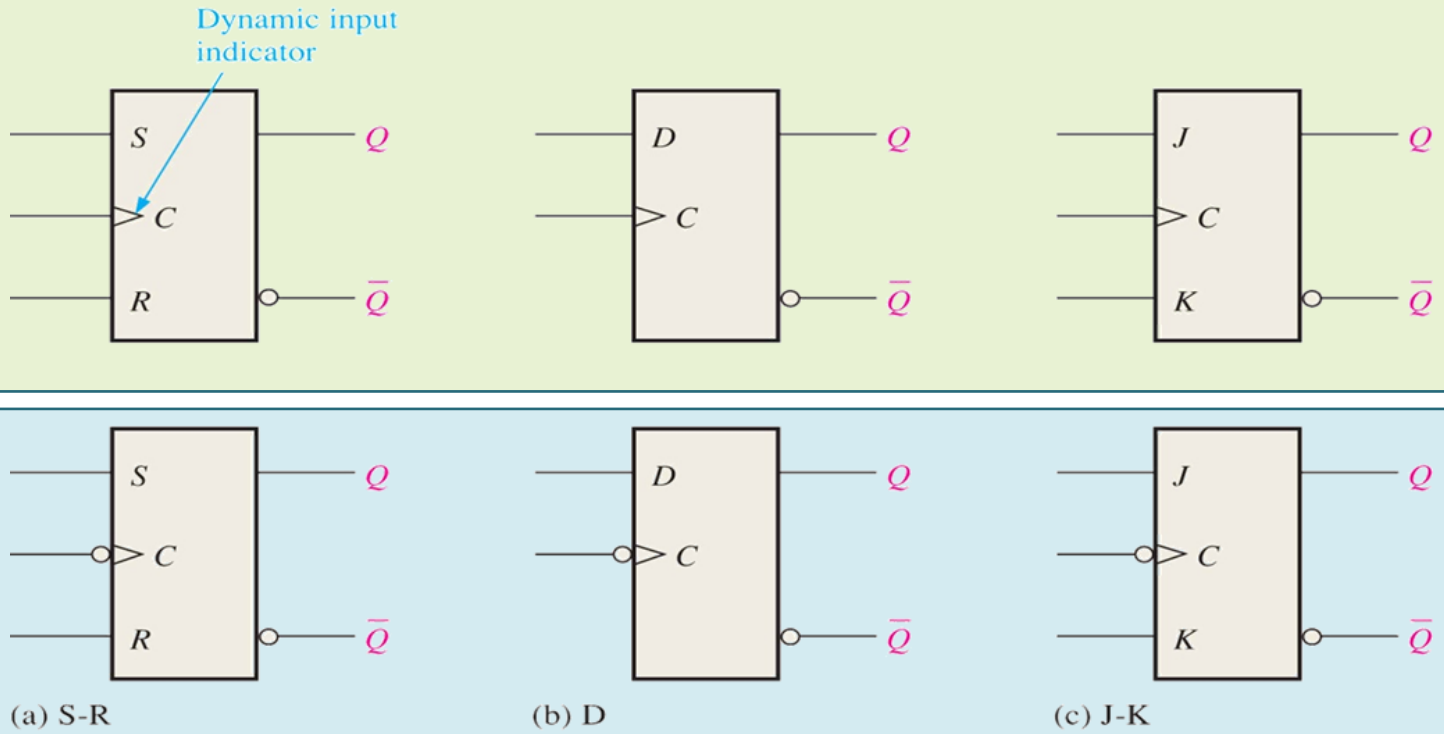
Flip-Flop

- 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)



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

Ex: Positive-edge triggered D Flip-Flop

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