Logic Circuits Course

Ch. 7-2

Latches, Flip-Flops and Timers

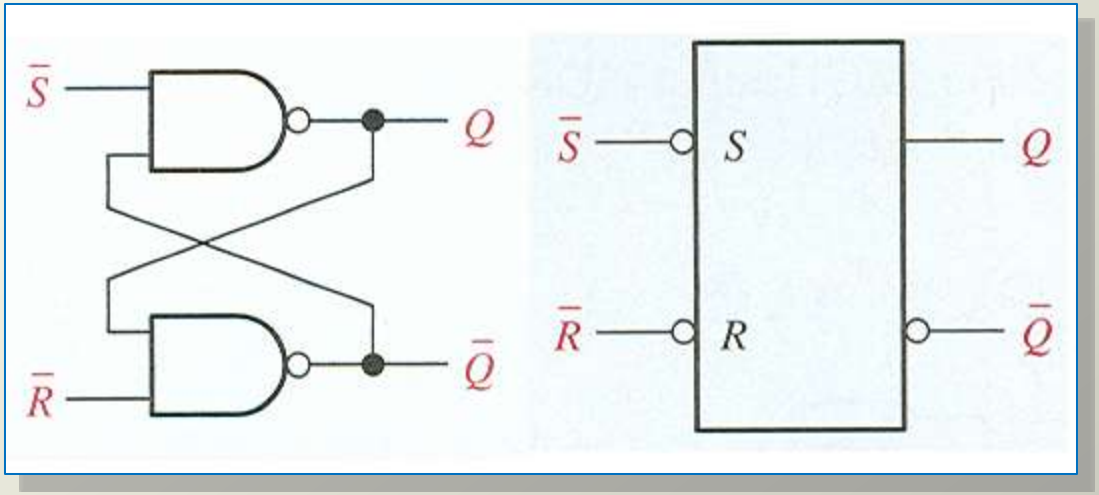
Dr. Abul Kareem Alaloosy

Latches, Flip-Flops, and Timers

1. Latches
2. Edge-Triggered Flip-Flops
3. Flip-Flop Operating Characteristics
4. Flip-Flop Applications
5. One-Shots
6. The Astable Multivibrator

Latches

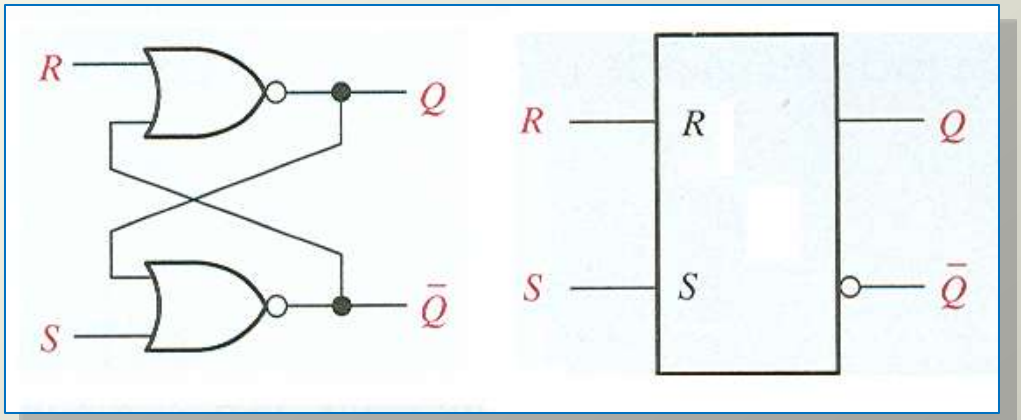
- $\overline{S}$-$\overline{R}$ latch



INPUTS		OUTPUTS		COMMENTS
$\overline{S}$	$\overline{R}$	Q	$\overline{Q}$	
1	1	NC	NC	No change. Latch remains in present state.
0	1	1	0	Latch SET.
1	0	0	1	Latch RESET.
0	0	1	1	Invalid condition

Latches

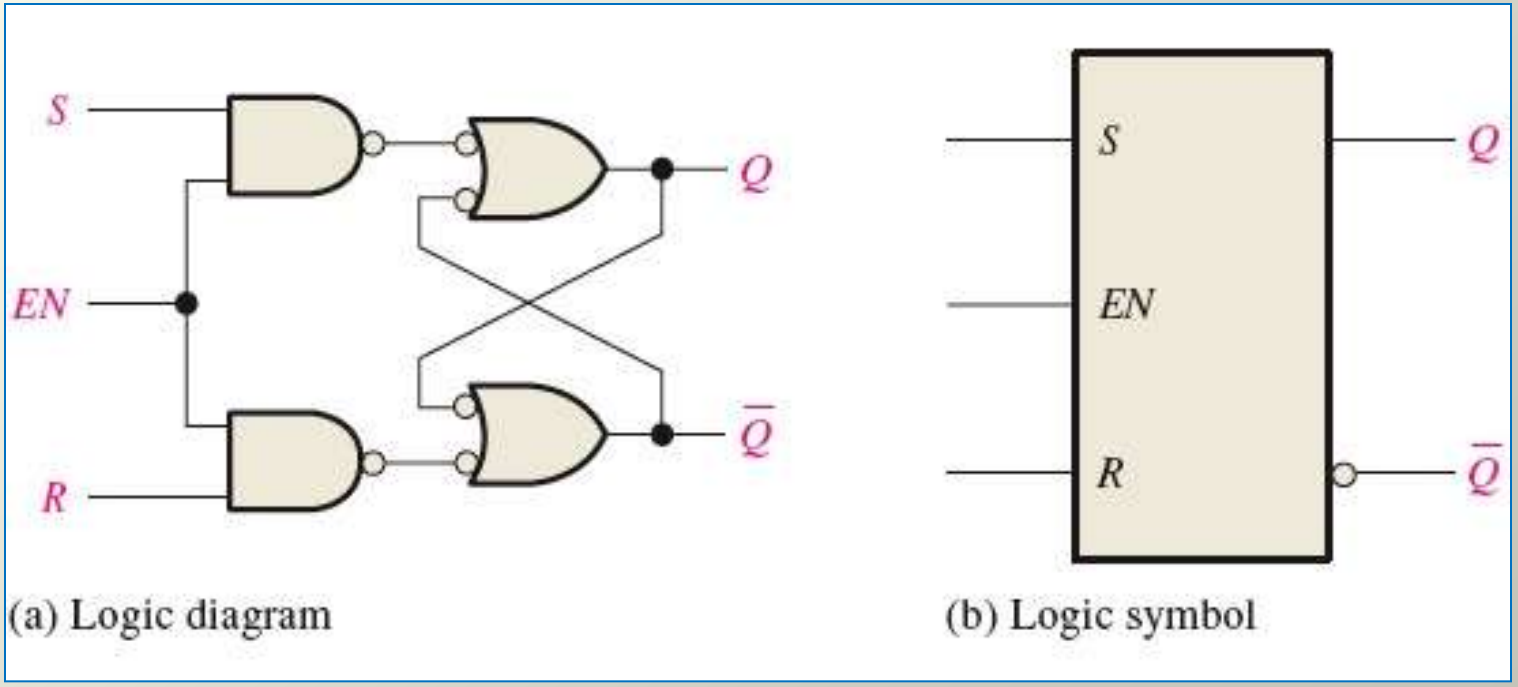
- S-R latch



INPUTS S	INPUTS R	OUTPUTS Q	OUTPUTS $\bar{Q}$	COMMENTS
0	0	NC	NC	No change. Latch remains in present state.
0	1	0	1	Latch RESET.
1	0	1	0	Latch SET.
1	1	0	0	Invalid condition

The Gated S-R Latch

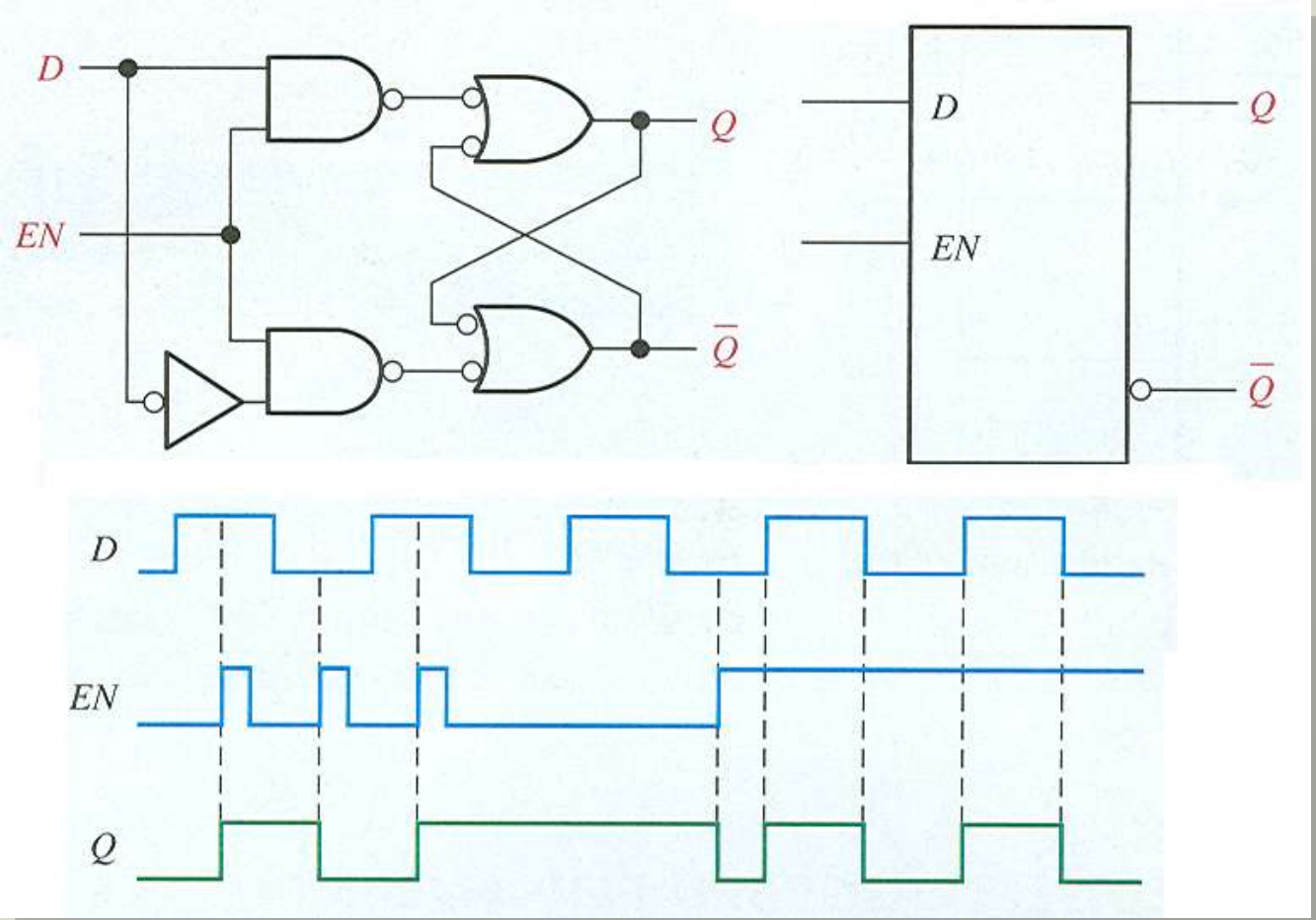
A gated latch requires an enable input, *EN.* The S and R inputs control the state to which the latch will go when a HIGH level is applied to the *EN* input. The latch will not change until *EN* is High, but as long as it remains HIGH, the output is controlled by the state of the S and R inputs. Invalid state occurs if both S and R is HIGH.



(a) Logic diagram (b) Logic symbol

The Gated D Latch

Another type of gated latch is Called the D latch. It differs from S-R latch because it has only one input in addition to EN. When the D input is HIGH and the EN input is HIGH, the latch will set. When the D input is LOW, and EN is HIGH the latch will reset.

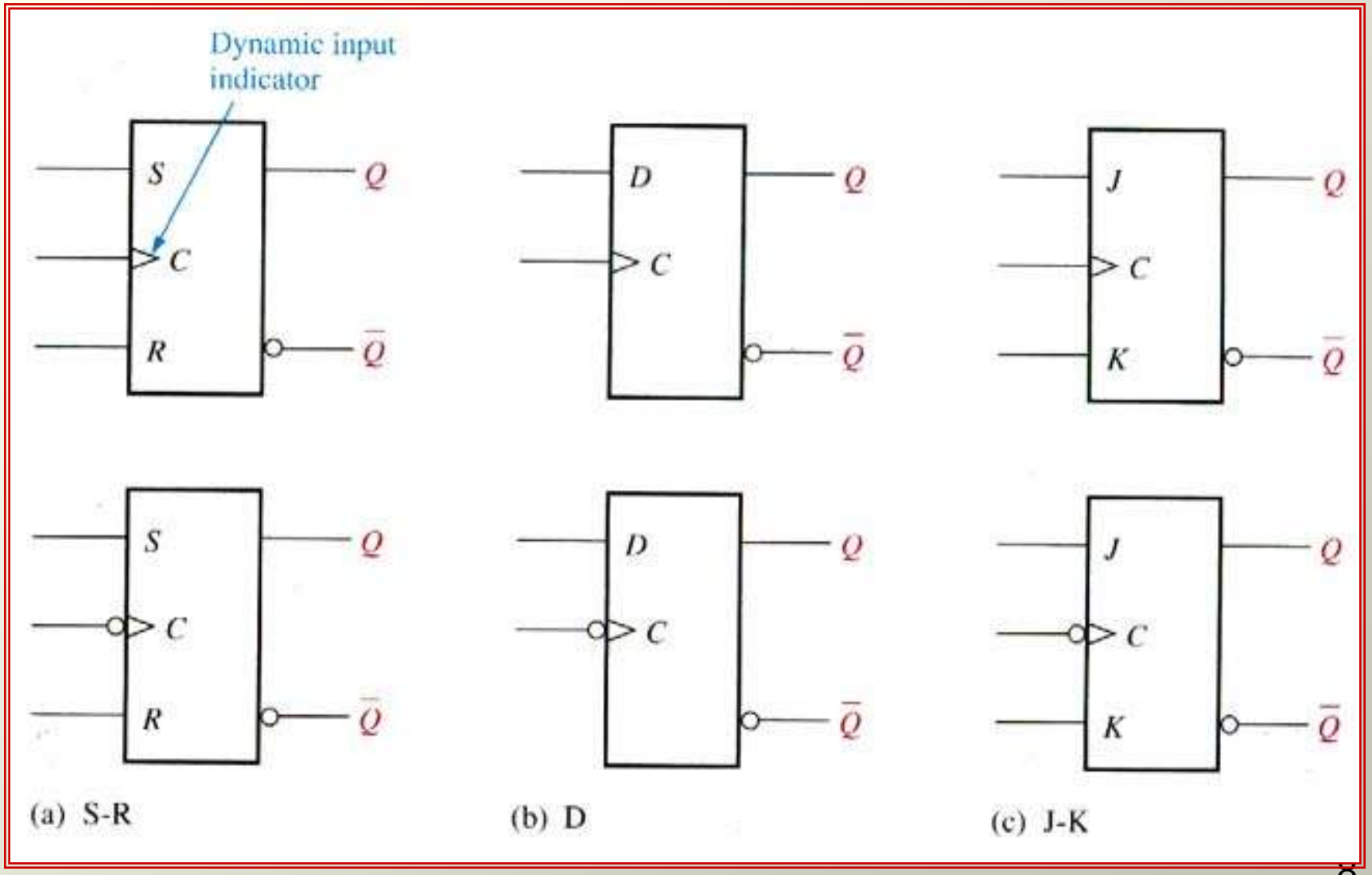


2. Edge-Triggered Flip-Flops

Flip-Flops are synchronous Bistable devices, also known as *Bistable multivibrator.* The term synchronous means that the output changes states only at a specified point on a triggering input called the *clock (CLK)* which is designed as a control input. An edge-triggered flip-flop changes state either at the positive edge (riding edge) or at the negative edge (falling edge) Of the clock pulse and is sensitive to its input only at this transition of the clock .

Three types of flip-flop are covered: S-R, D and J-K flip-flops. The key to identifying an edge-triggered flip-flop by its logic symbol is the small triangle inside the block at the clock (C) input. The triangle is called the *dynamic input indicator.*

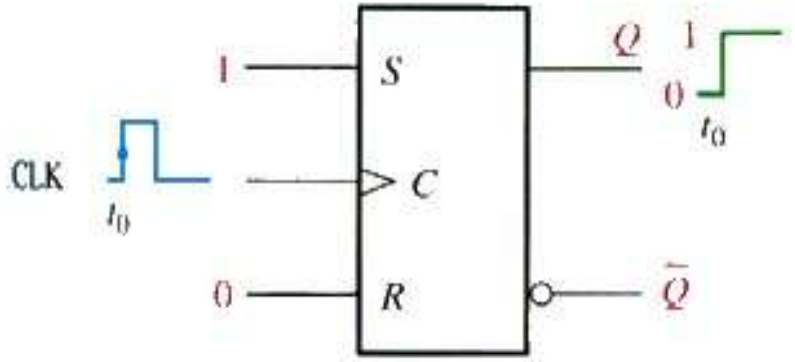
Figure 7.13 Edge-triggered flip-flop logic symbols (top: positive edge-triggered; bottom: negative edge-triggered).



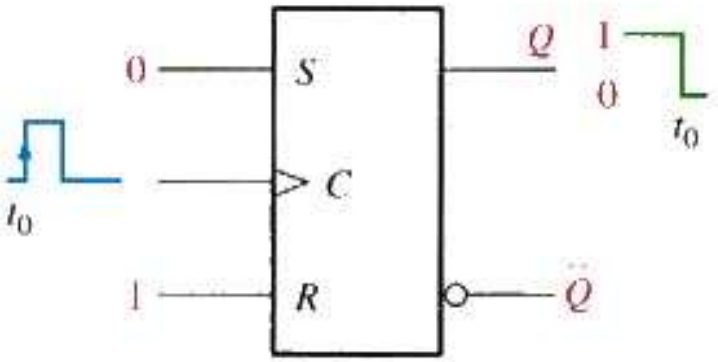
The Edge-Triggered S-R Flip-Flop

The S and R inputs of the S-R flip-Flops are called ***synchronous*** inputs because data on these inputs are transferred to the flip-flop's output only on the triggering edge of the clock pulse. When S is HIGH and R is LOW, the Q output goes HIGH on the triggering edge of the clock pulse, and the flip-flop is SET. When S is LOW and R is HIGH, the Q output goes LOW on the triggering edge, the output does not change from its prior state. An invalid condition exists when both S and R are HIGH.

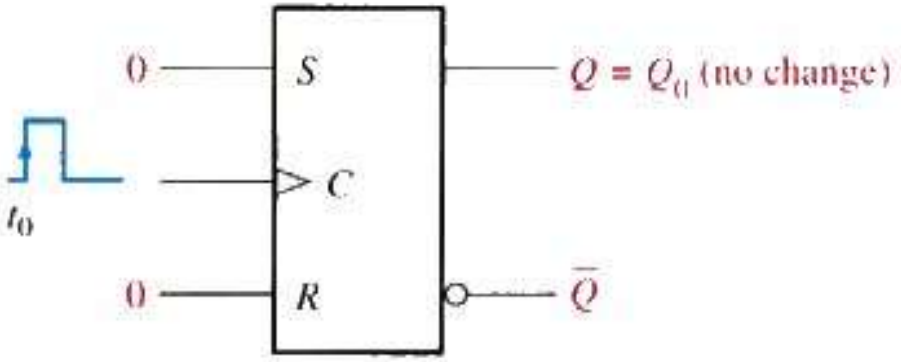
Figure 7.14 Operation of a positive edge-triggered S-R flip-flop.



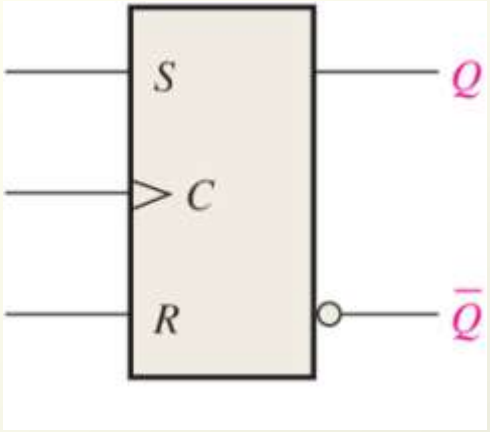
(a) $S = 1$, $R = 0$ flip-flop SETS on positive clock edge. (If already SET, it remains SET.)



(b) $S = 0$, $R = 1$ flip-flop RESETS on positive clock edge. (If already RESET, it remains RESET.)



(c) $S = 0$, $R = 0$ flip-flop does not change. (If SET, it remains SET; if RESET, it remains RESET.)



INPUTS			OUTPUTS		COMMENTS
S	R	CLK	Q	$\overline{Q}$	
0	0	X	Q_0	$\overline{Q}_0$	No change
0	1	↑	0	1	RESET
1	0	↑	1	0	SET
1	1	↑	?	?	Invalid

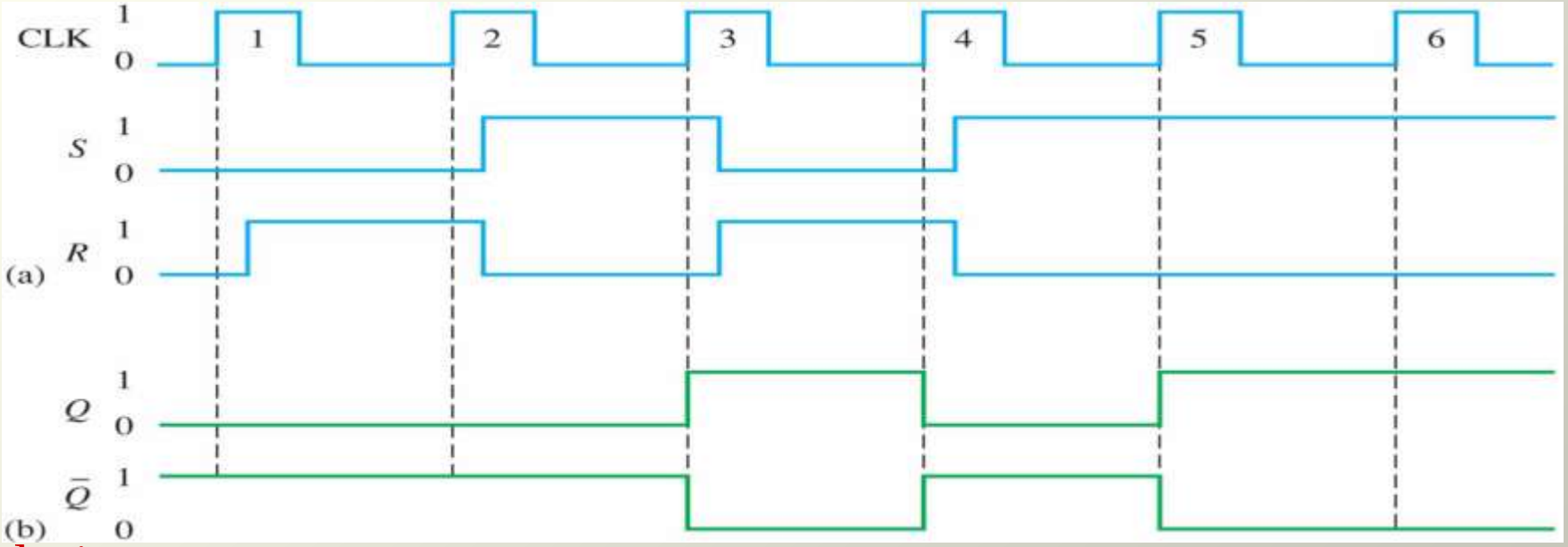
↑ = clock transition LOW to HIGH

X = irrelevant ("don't care")

Q_0 = output level prior to clock transition

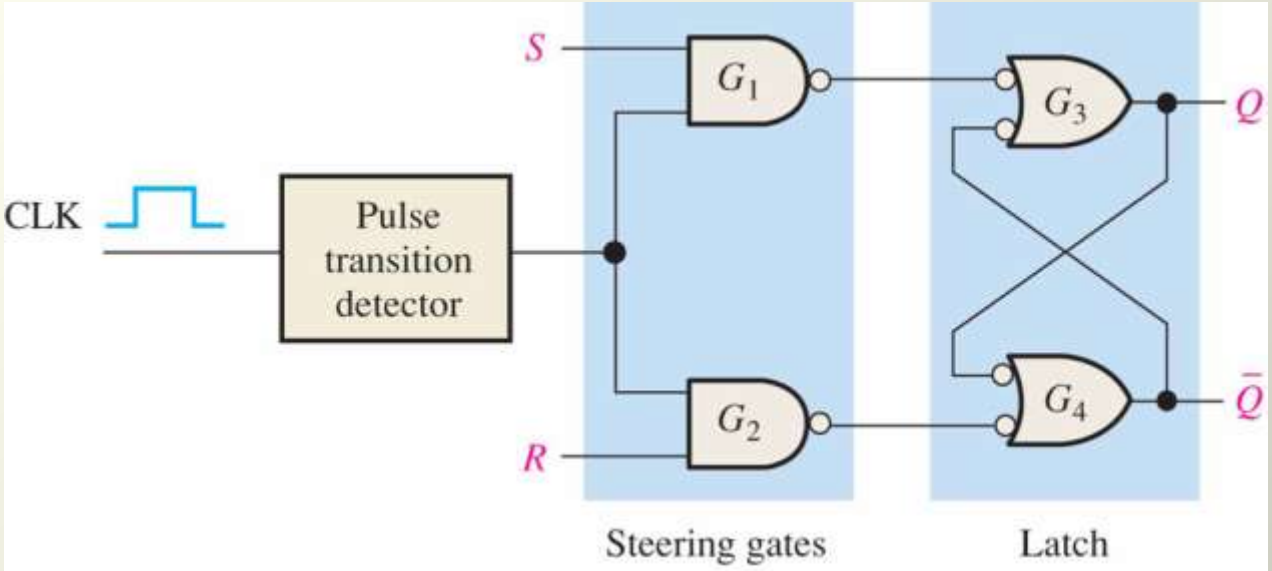
Example:- Determine the output waveforms of the flip-flop for the S,R and CLK inputs, assuming that the positive edge-triggered flip-flop is initially RESET

INPUTS			OUTPUTS		
S	R	CLK	Q	$\bar{Q}$	COMMENTS
0	0	X	Q_0	$\bar{Q}_0$	No change
0	1	↑	0	1	RESET
1	0	↑	1	0	SET
1	1	↑	?	?	Invalid

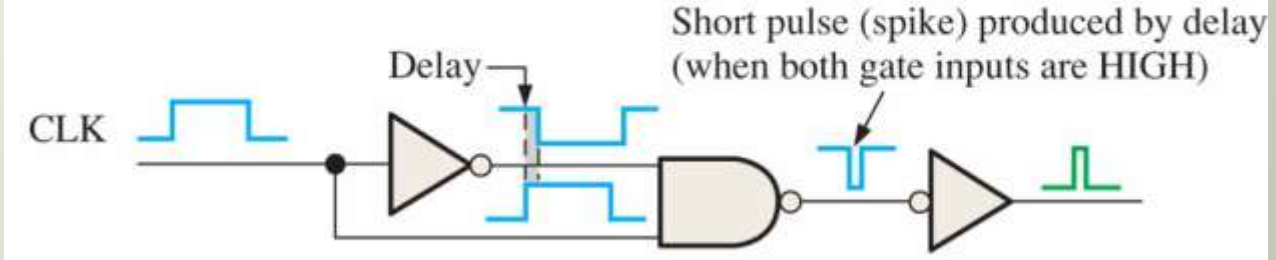


Solution:- 1-At clock pulse 1, S is LOW and R is LOW, so Q does not change
2-At clock pulse 2, S is LOW and R is HIGH, so Q remains LOW(RESET)
3-At clock pulse 3, S is HIGH and R is LOW, so Q goes HIGH (SET).
4-At clock pulse 4, S is LOW and R is HIGH, so Q goes LOW (RESET)
5-At clock pulse 5, S is HIGH and R is LOW, so Q goes HIGH (SET).
6-At clock pulse 6, S is HIGH and R is LOW, so Q stays HIGH.

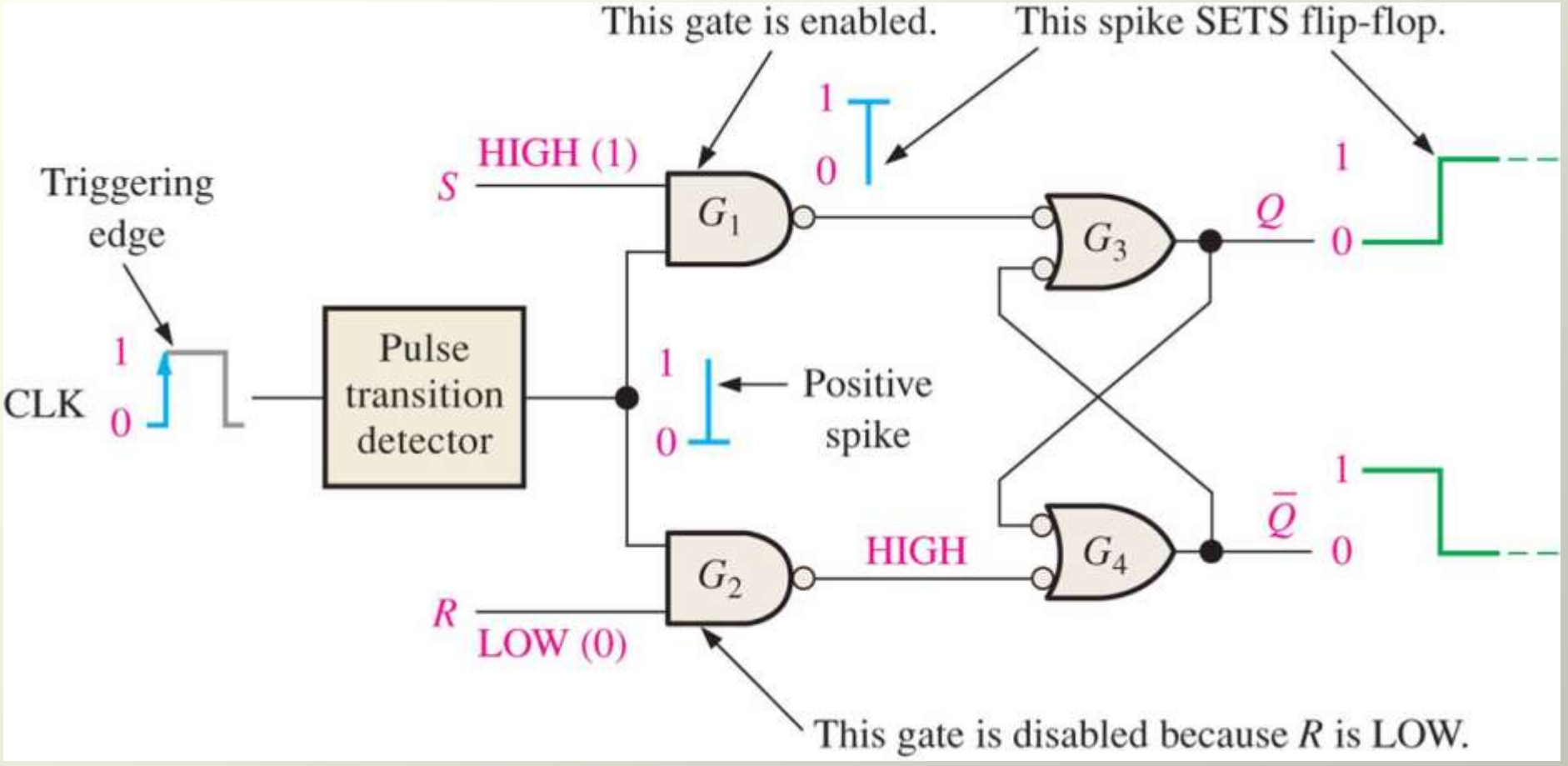
A Method of Edge-Triggering: concept

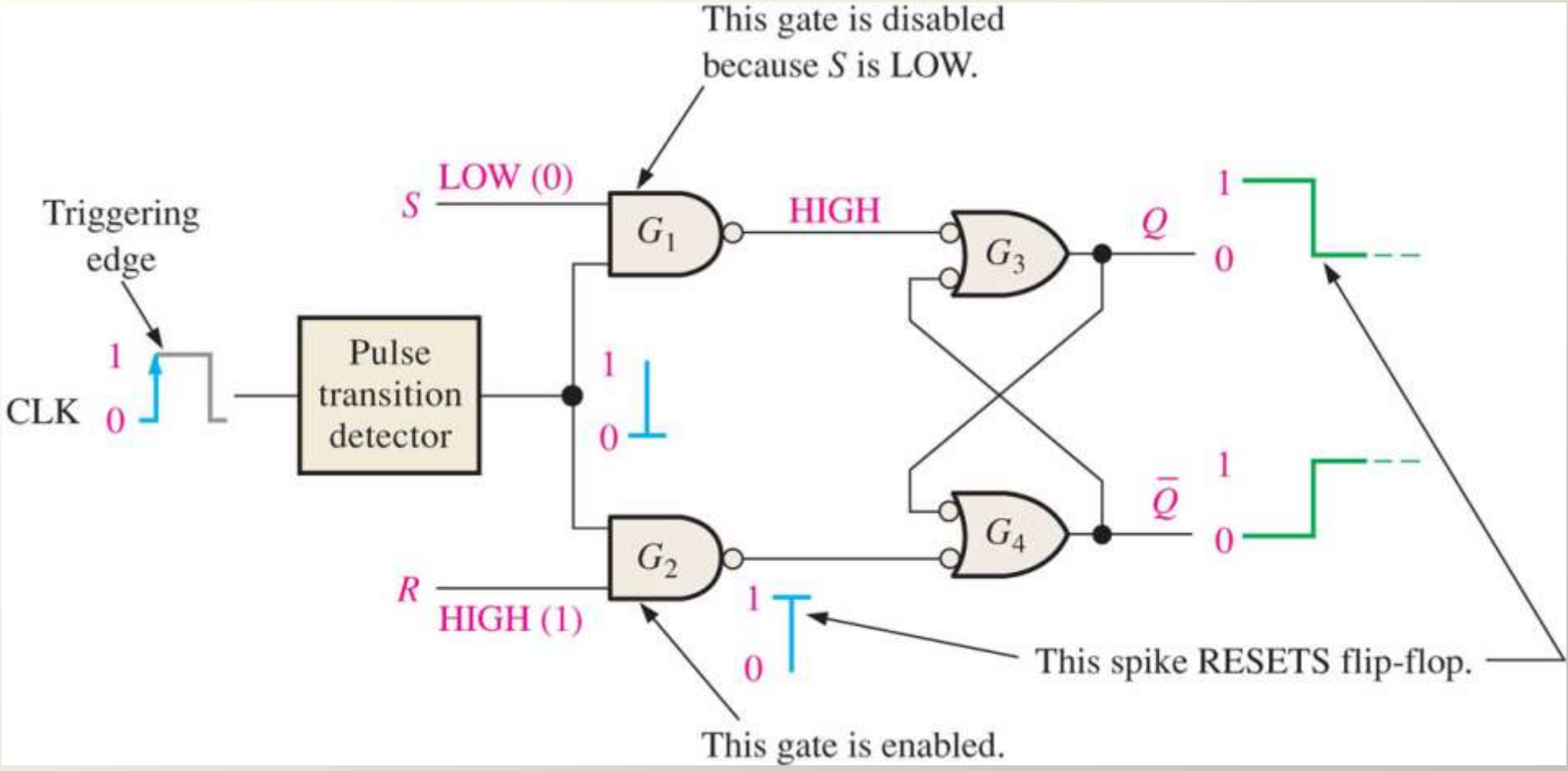


(a) A simplified logic diagram for a positive edge-triggered S-R flip-flop



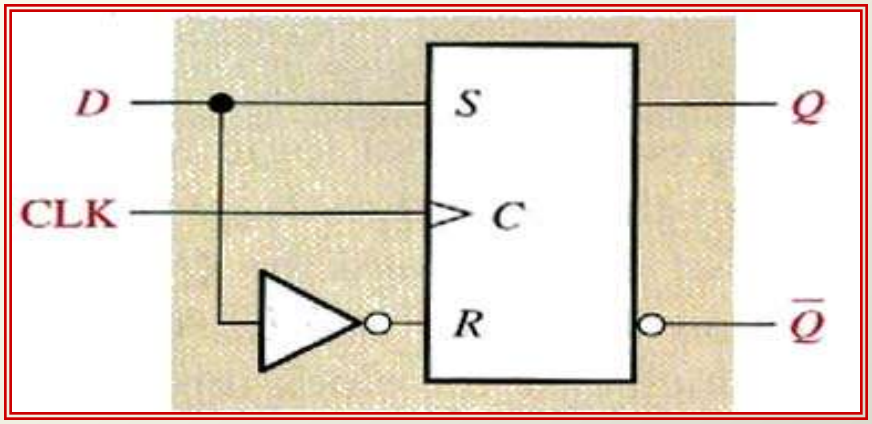
(b) A type of pulse transition detector


This gate is enabled.
This spike SETS flip-flop.
Triggering edge
CLK
1
0
Pulse transition detector
S
HIGH (1)
R
LOW (0)
G_1
G_2
G_3
G_4
1
0
Positive spike
HIGH
Q
$\bar{Q}$
This gate is disabled because R is LOW.


This gate is disabled
because S is LOW.
S
LOW (0)
Triggering
edge
1
CLK
0
Pulse
transition
detector
G1
HIGH
G3
Q
1
0
1
0
1
0
Q̄
G4
G2
R
HIGH (1)
1
0
This spike RESETS flip-flop.
This gate is enabled.

The Edge-Triggered D Flip-Flop

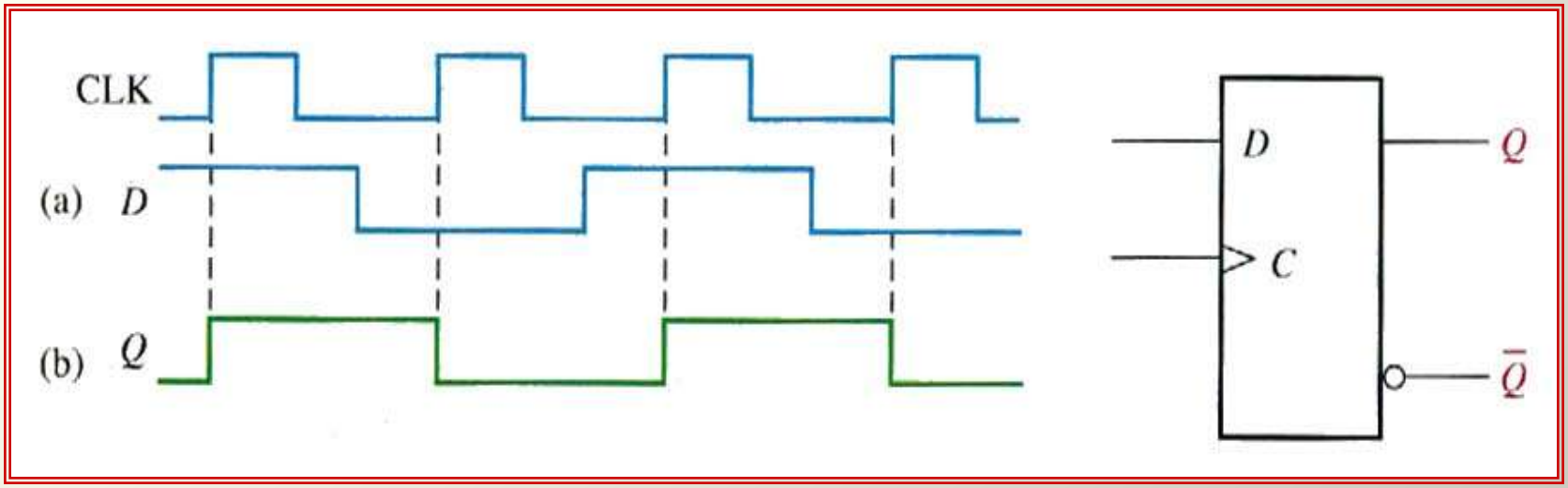
The D flip-flop is useful when a single data bit (1 or 0) is to be stored. The addition of an inverter to an S-R flip flop creates a basic D flip-flop.



INPUTS		OUTPUTS		COMMENTS
D	CLK	Q	$\overline{Q}$	
1	↑	1	0	SET (stores a 1)
0	↑	0	1	RESET (stores a 0)

↑ = clock transition LOW to HIGH

The Edge-Triggered D Flip-Flop

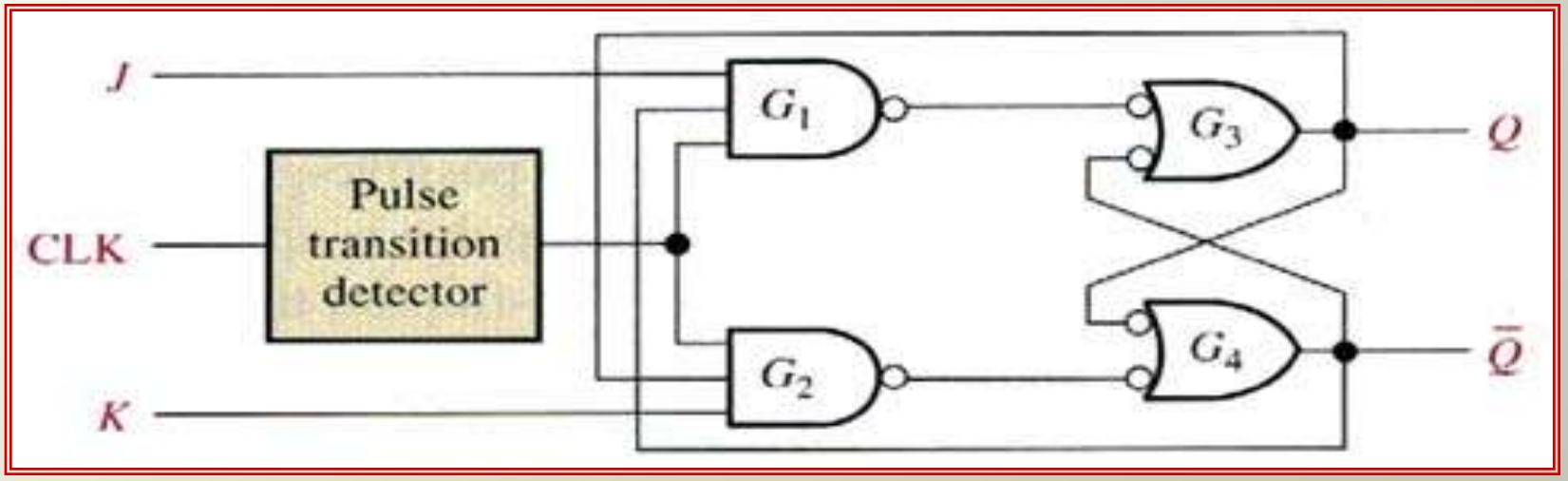


If there is a HIGH on the D input when a clock pulse is applied, the flip-flop will set, and the HIGH on the D input is stored by the flip-flop on the positive-going edge. If there is a low on the D input when the clock pulse is applied, the flip-flop will reset, and the LOW on the D input is stored by the flip-flop of the leading edge of the clock pulse. In the *SET* state, the flip-flop is storing 1 and in the *RESET* state it is storing a 0

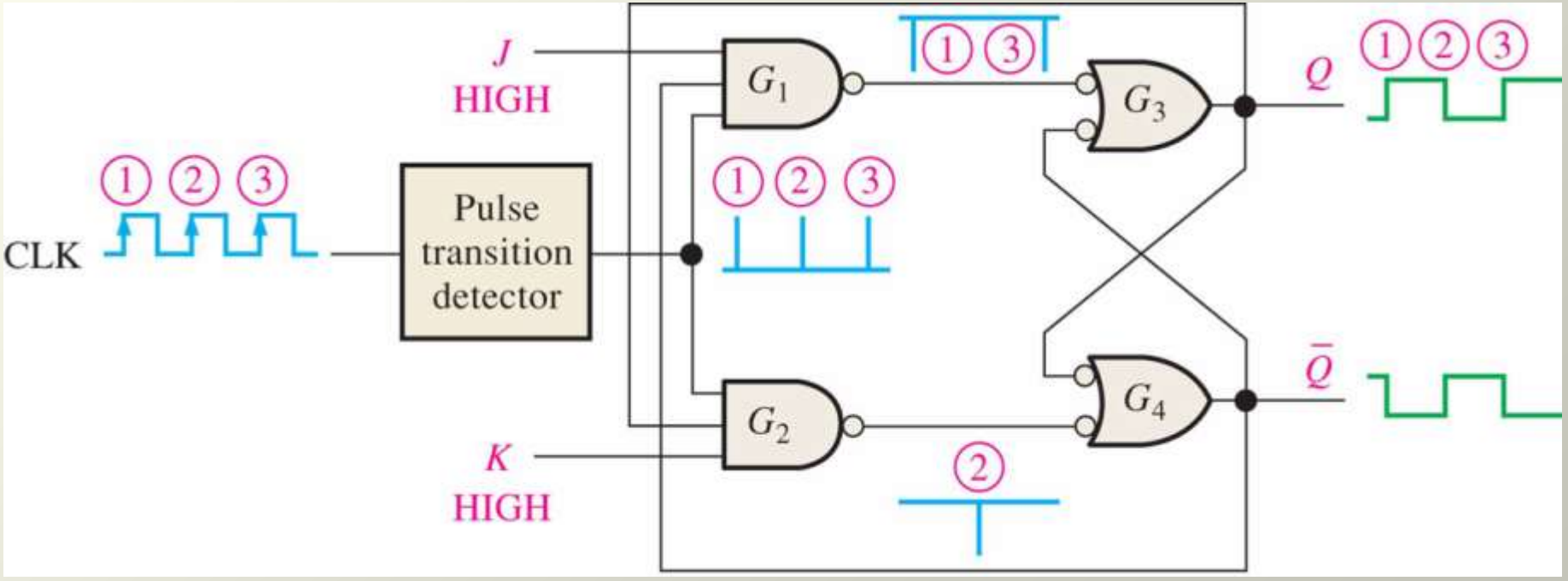
The Edge-Triggered J-K Flip-Flop

The functioning of the J-K flip-flop is identical to that of the S-R flip-flop in the SET, RESET, and no-change conditions of operation.

The difference is that the **J-K flip-flop has no invalid state** as does the S-R flip-flop. A J-K flip-flop can also be of the negative edge-triggered type, in which case the clock input is inverted.



If we assume that the flip-flop is RESET and that the J input is HIGH and the K input is LOW, when a clock pulse occurs, a leading-edge spike indicated by 1 is passed through gate G_1 because $\overline{Q}$ is HIGH and J is HIGH. This will cause the latch portion of the flip-flop to change to SET state.



The flip-flop is now set. If you now make J LOW and K HIGH, the next clock spike indicated by 2 will pass through gate G_2 because Q is HIGH and K is HIGH. This will cause the latch portion of the flip-flop to change to the RESET state.

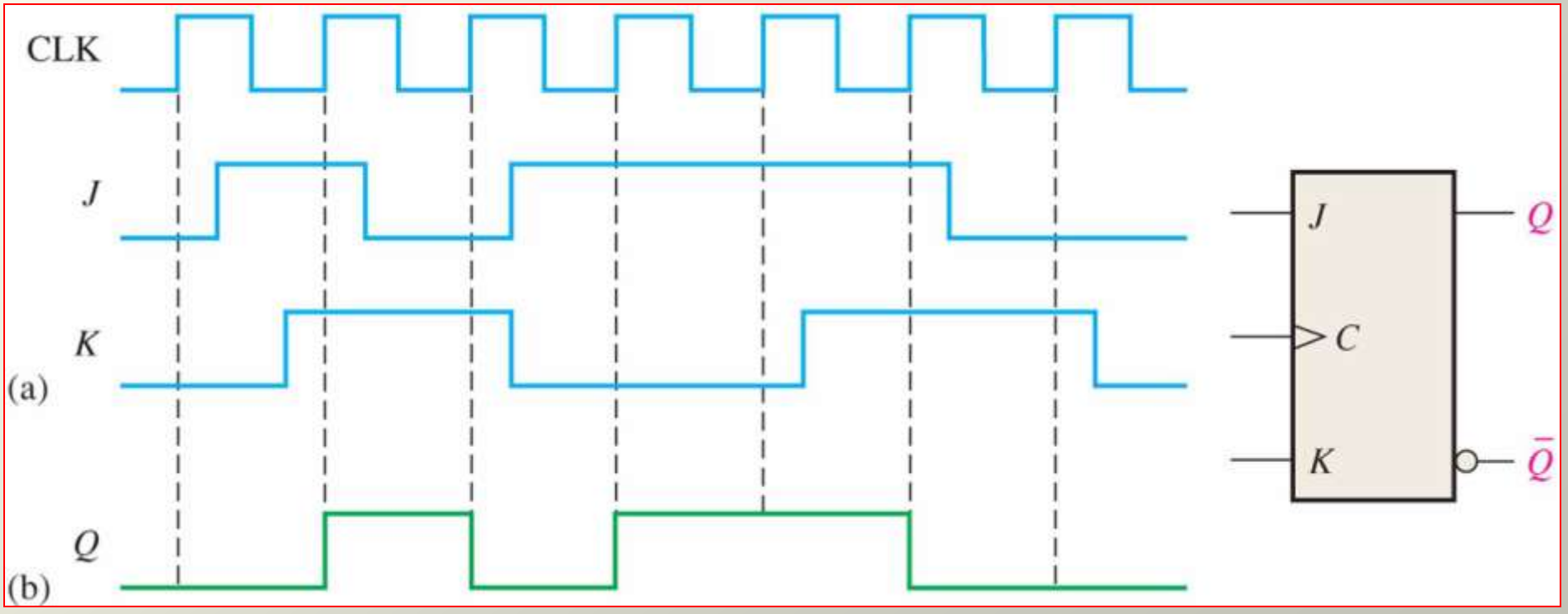
Now, if a LOW is applied to both the J and K inputs, the flip-flop will stay in its present state when a clock pulse occur.

When both the J and K inputs are HIGH, the HIGH on the $\overline{Q}$ enables gate G_1, so the clock spike indicated by 3 passes through to set the flip-flop, now there is a HIGH on Q, which allows the next clock spike to pass through gate G_2 and reset the flip-flop. In each successive clock spike, the flip-flop changes to opposite state. This mode is called *Toggle* operation.

INPUTS			OUTPUTS		COMMENTS
J	K	CLK	Q	$\overline{Q}$	
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

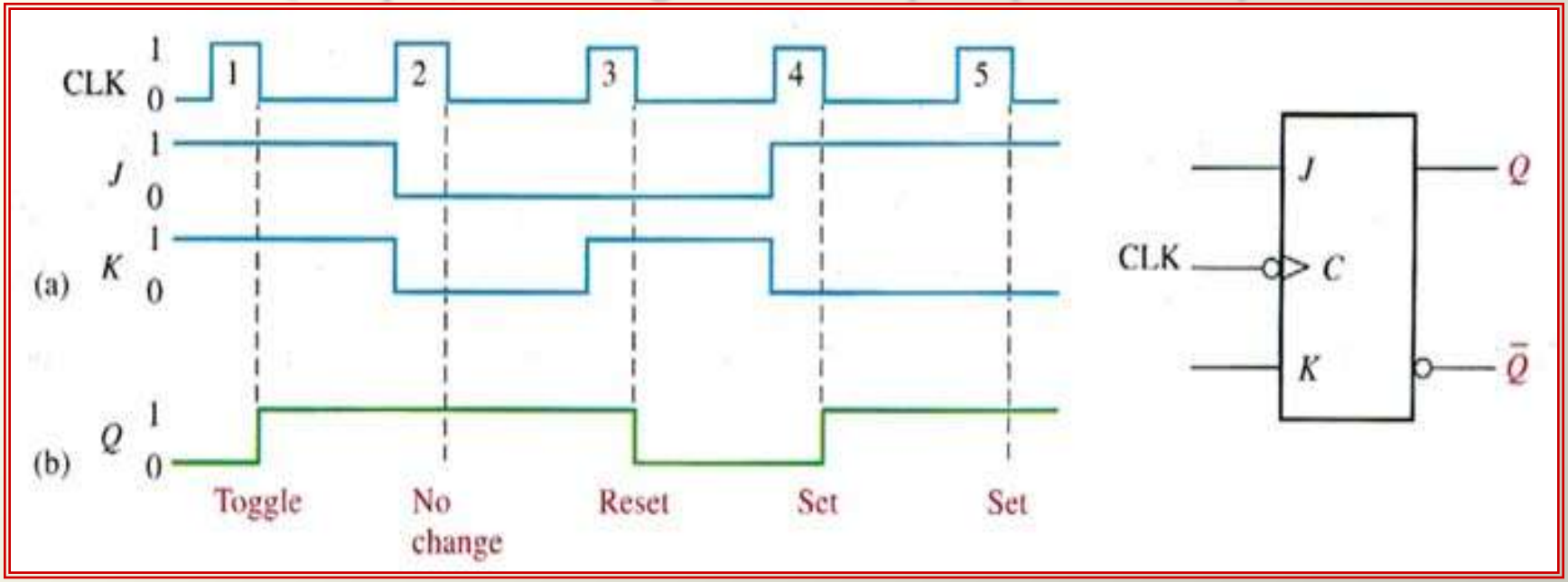
↑ = clock transition LOW to HIGH

Q_0 = output level prior to clock transition



A negative edge-triggered flip-flop as indicated by the “bubble” at the clock input. The Q output will change only on the negative-going edge edge of the clock pulse.

Example :- the waveform are applied to the J,K and Clock input, you determine the Q output, assuming that the flip-flop is initially RESET



1- At the first clock pulse, both J and K are HIGH, and because this is a toggle condition, Q goes HIGH

2- At clock pulse 2, a no-change condition exists, keeping Q at HIGH.

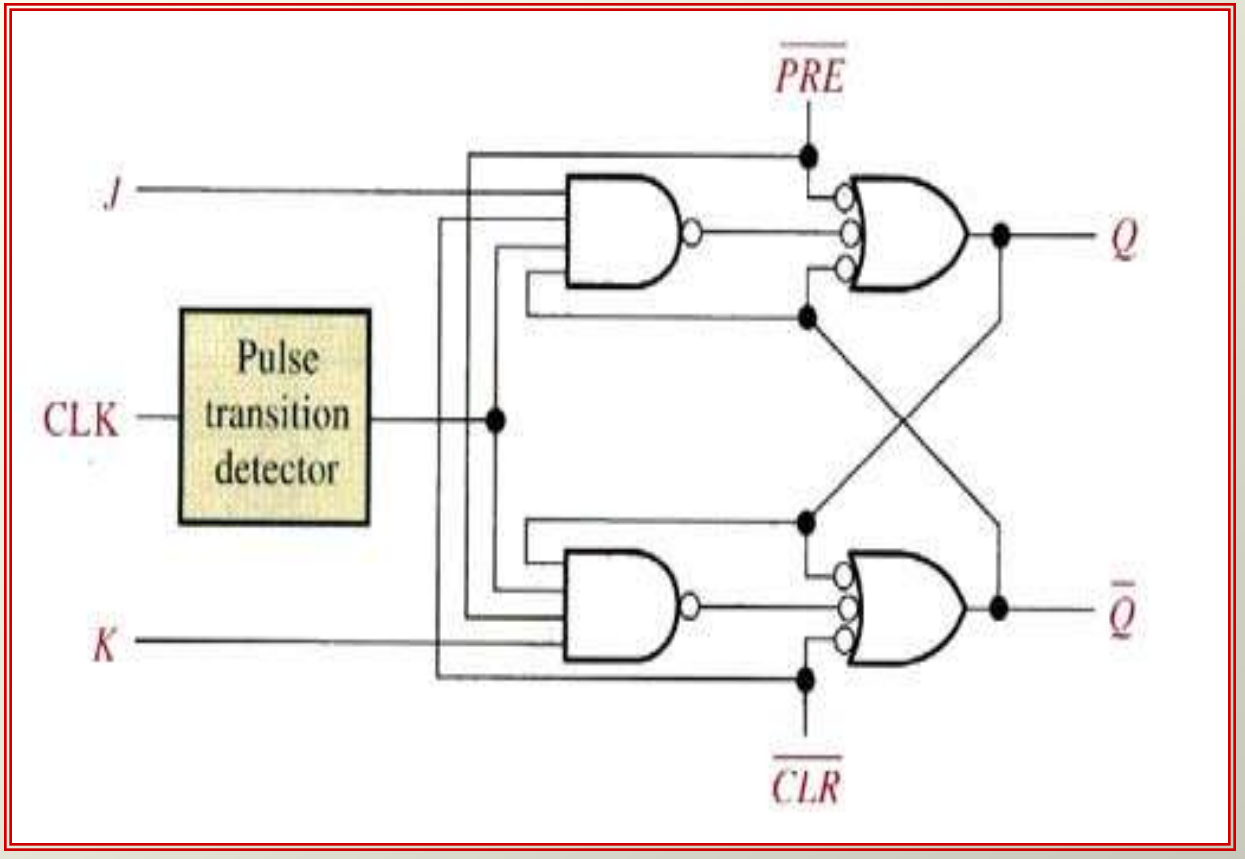
3- For clock pulse 3, J is LOW and K is HIGH, resulting RESET, Q goes LOW.

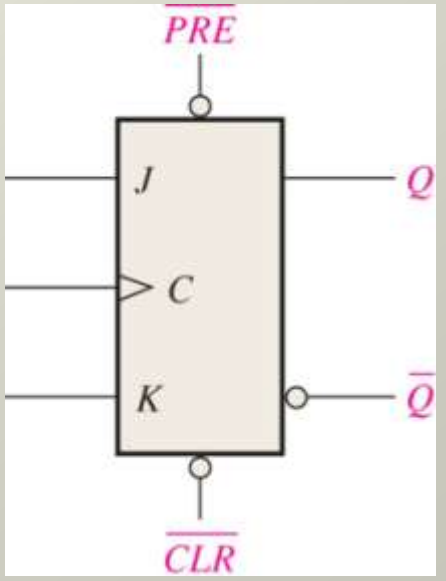
4- At clock pulse 4, J is HIGH and K is LOW resulting in a SET; Q goes HIGH.

5- A SET condition still exists on J and K with pulse 5 , so Q will remain HIGH

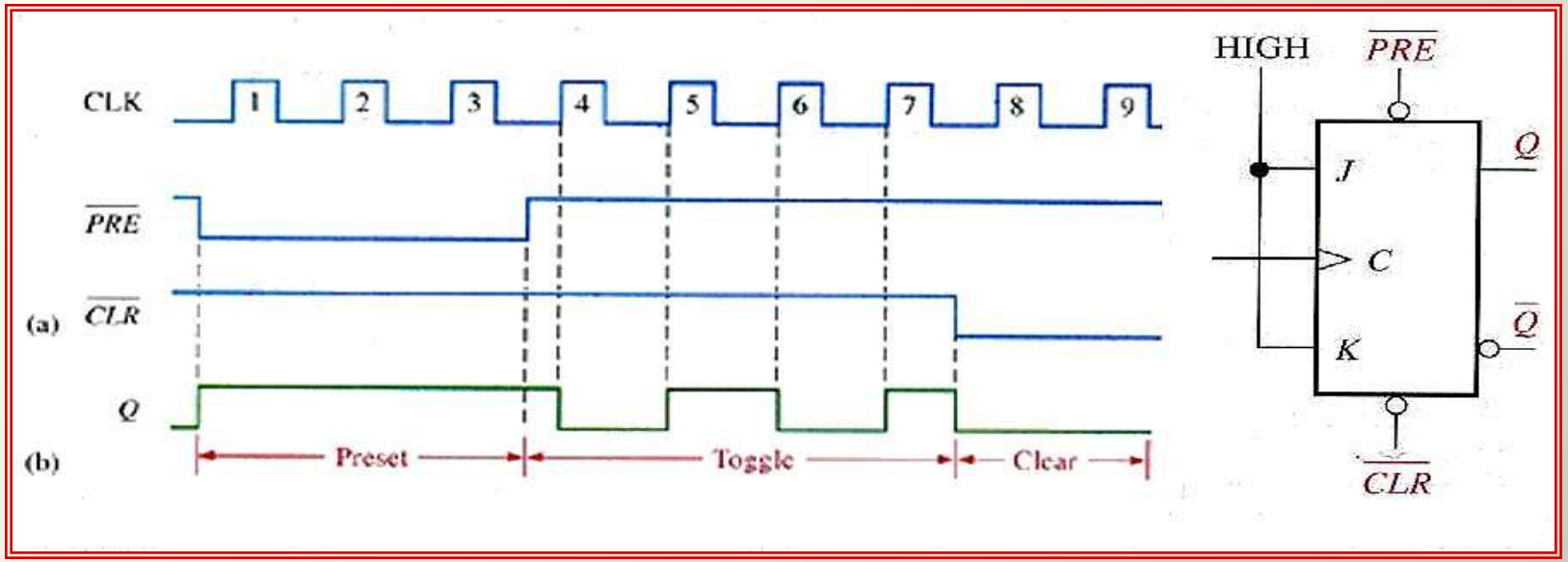
Asynchronous Preset and Clear input

➢The figure below shows a logic diagram for an edge-triggered J-K flip-flop with active LOW preset ($\overline{PRE}$) and clear ($\overline{CLR}$) inputs.





Example :- For the positive edge-triggering j-k flip-flop with preset and clear inputs, determine Q for the inputs if q is initially LOW.



Solution :-

1- During clock pulse 1,2 and 3, the preset ($\overline{PRE}$) is LOW, keep the flip-flop SET regardless of the synchronous J and K inputs.

2- For clock pulse 4,5,6, and 7 toggle operation occurs because J is High, K is High, and both $\overline{PRE}$ and $\overline{CLE}$ are High.

3-For clock pulses 8 and 9, the clear $\overline{CLE}$ input is LOW, keeping the flip-flop RESET regardless of the synchronous inputs.

Figure 7.29 Logic symbols for the 74AHC74 dual positive edge-triggered D flip-flop.

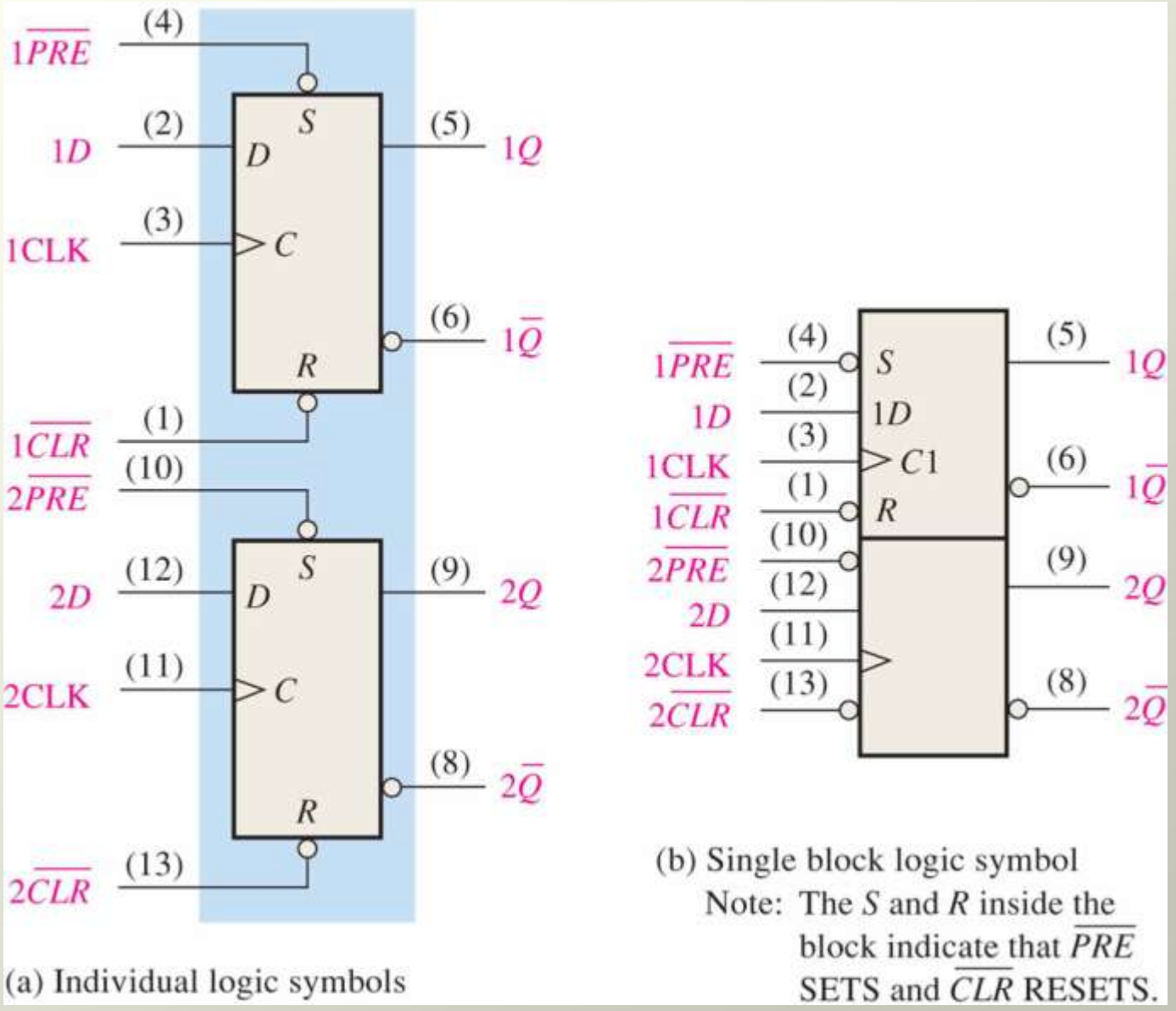
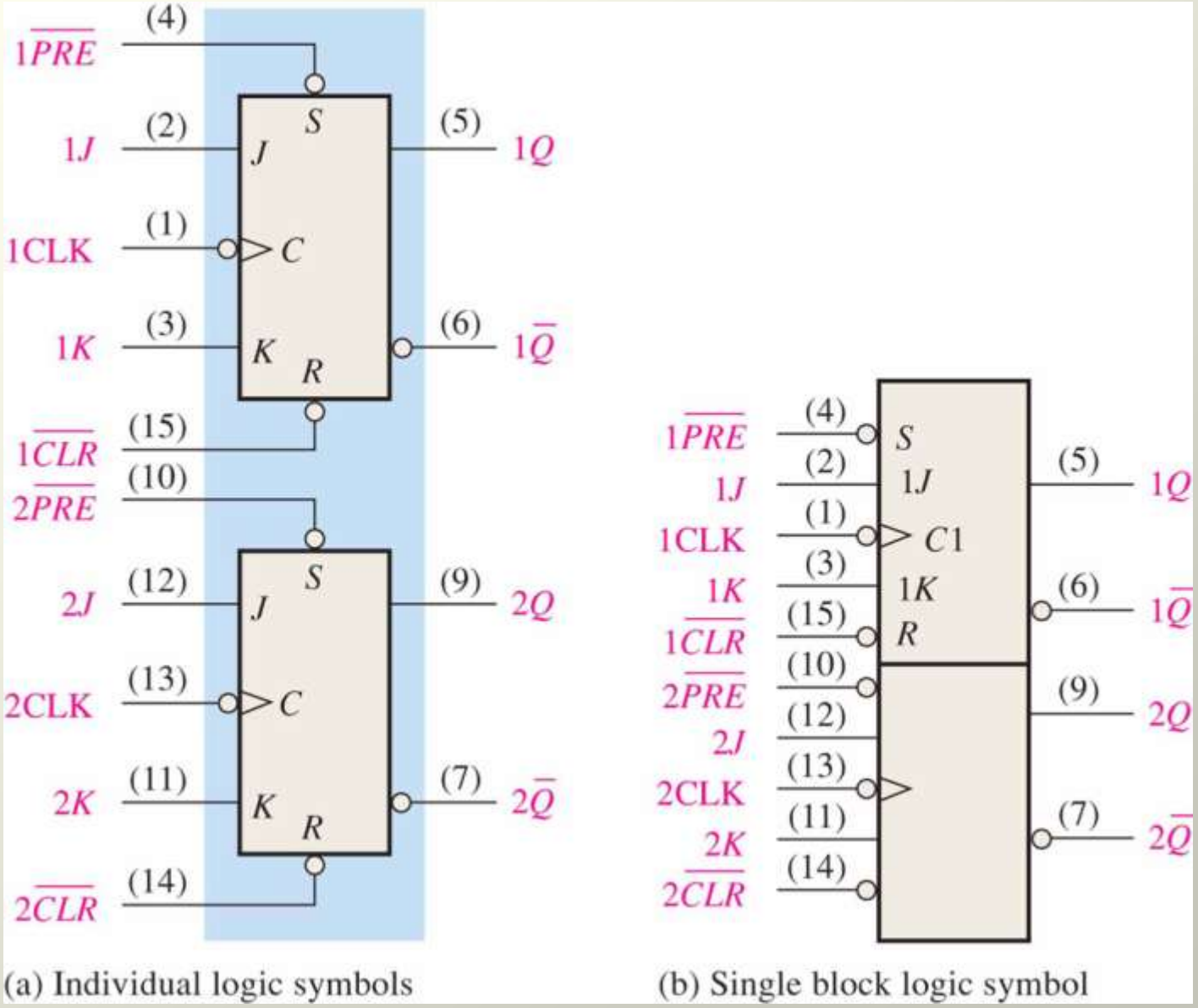
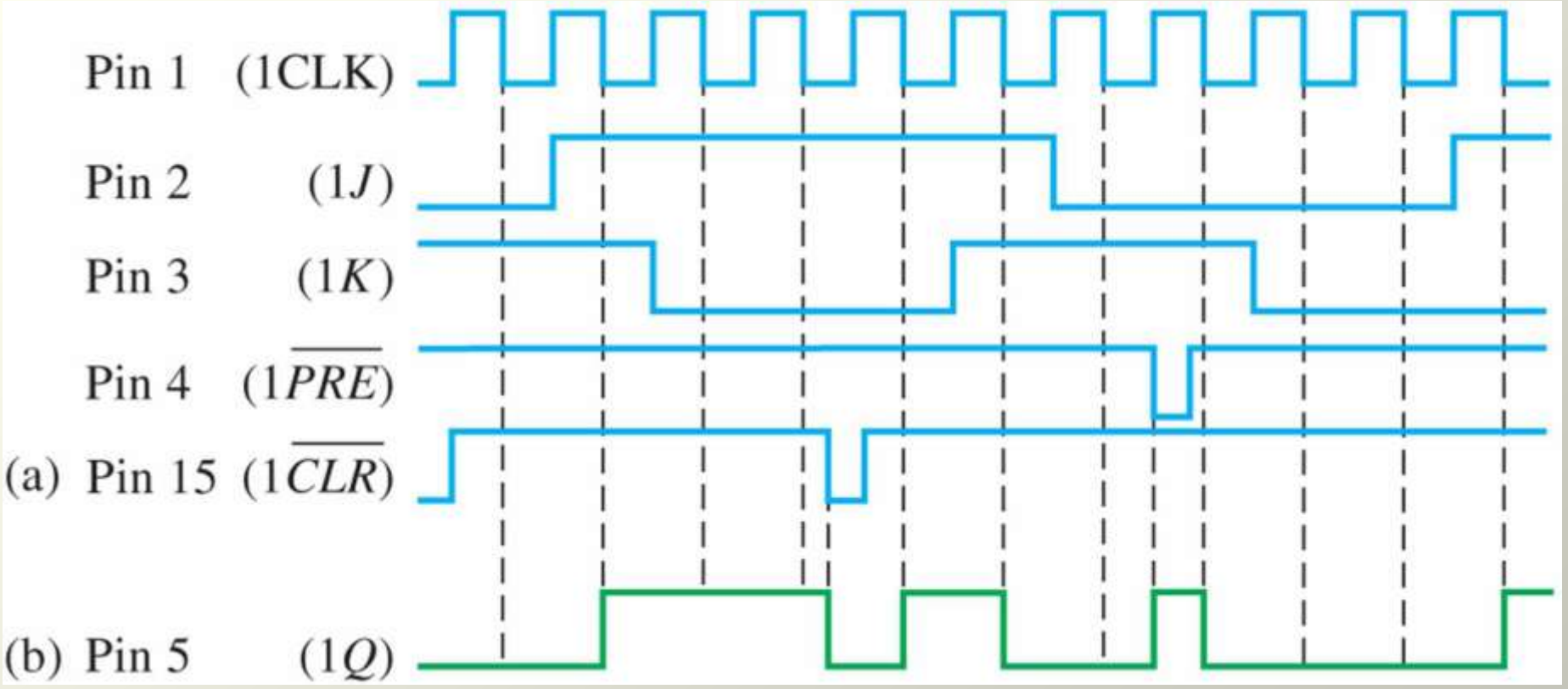


Figure 7.30 Logic symbols for the 74HC112 dual negative edge-triggered J-K flip-flop.



Example : 1J, 1K, 1CLK, $1\overline{PRE}$, $1\overline{CLR}$ waveforms are supplied to one of the negative edge-triggered flip-flop in a 74HC112 package determine the 1Q



Each time the LOW is applied to $1\overline{PRE}$ and $1\overline{CLR}$ inputs the flip-flop is set or reset regardless of the state of the inputs