UTM
UNIVERSITI TEKNOLOGI MALAYSIA

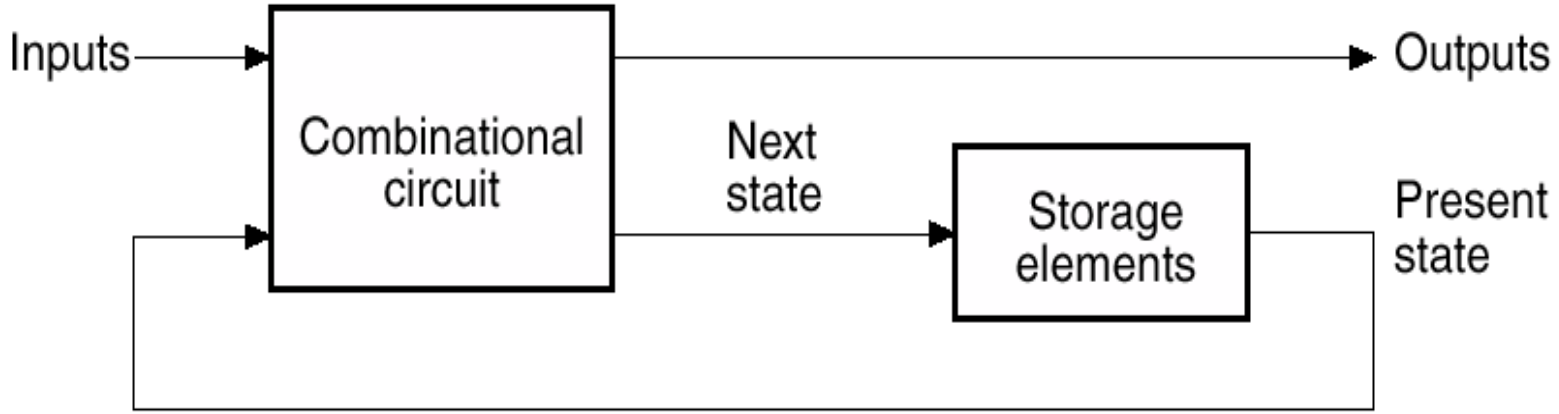
SKEE1223: Digital Electronics

11 – Latches and Flip-Flops

Dr Michael Tan Loong Peng
PhD (Cambridge)
Senior Lecturer
Faculty of Electrical Engineering
Universiti Teknologi Malaysia

Introduction

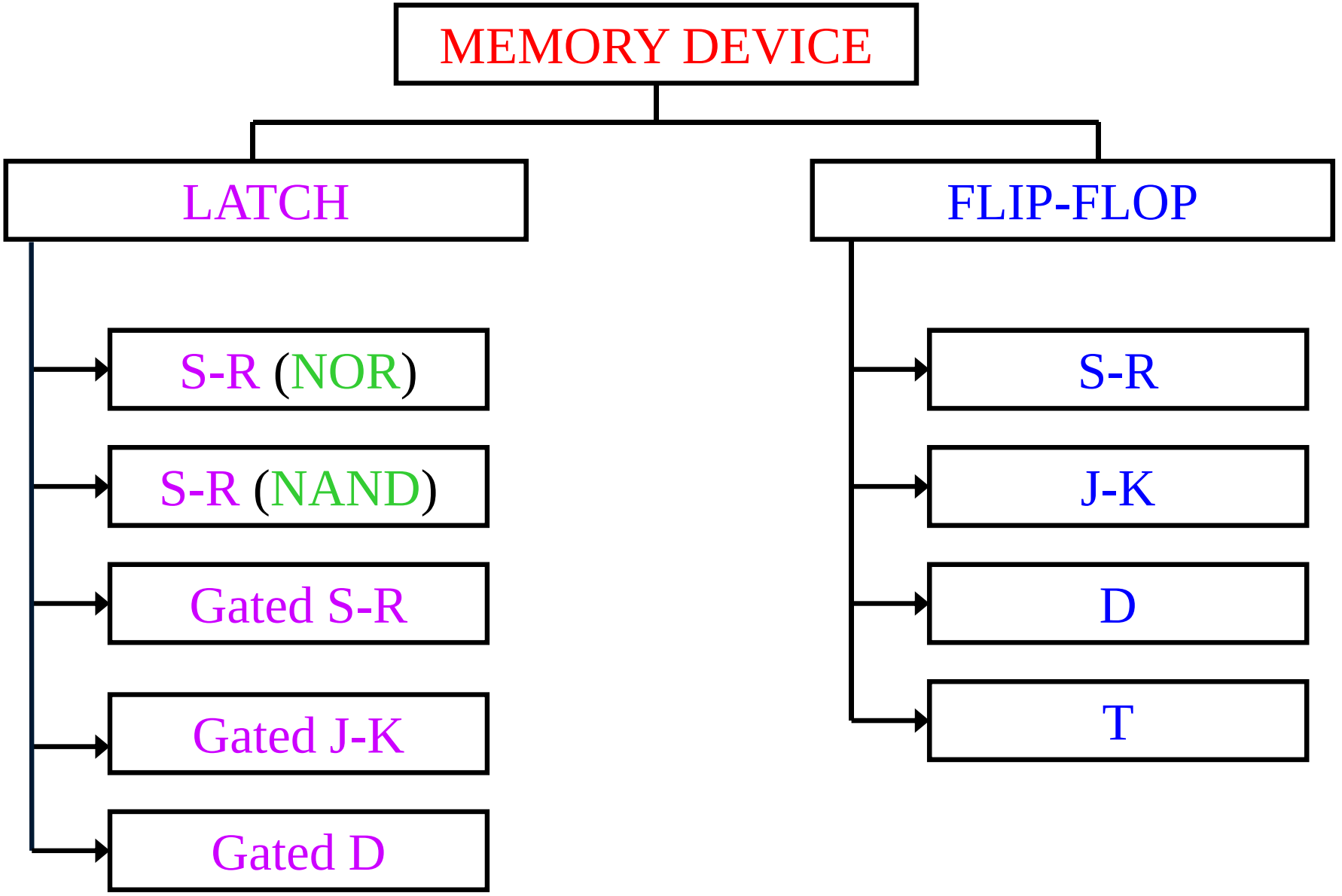
- **A combinational circuit is a circuit whose outputs depends only upon its current inputs.**
  - Do not have memory
  - Cannot store state
- **A sequential circuit is a circuit whose outputs depends on the current inputs and the system's current state** (past behavior).
  - Require use of storage elements.
  - Contents of storage elements is called state.
  - Circuit goes through sequence of states as a result of changes in inputs.



Storage or Memory Element

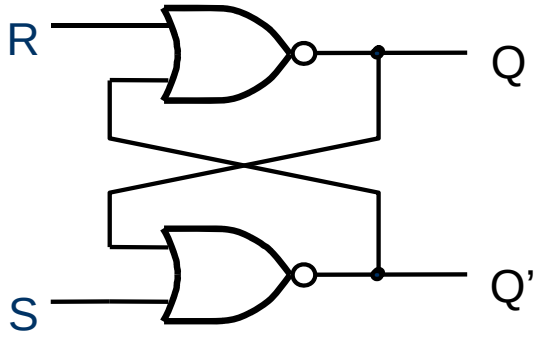
- Two types of memory element :-
- **LATCH**: temporary storage device that has two stable states (bi-stable). Output is executed upon changes in the input.
- **FLIP-FLOP**: synchronous bi-stable device. Synchronous means that the output changes state only at a specified point on the triggering input called the clock (CLK). Output changes are synchronize with the changes of the clock state.

Memory Element

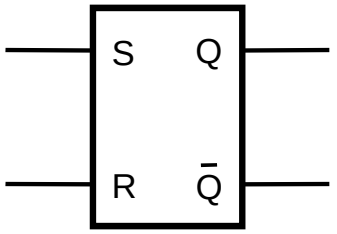


SR Latch (NOR version)

- Set-Reset latch
- Qn is initial state, Qn+1 is next state.



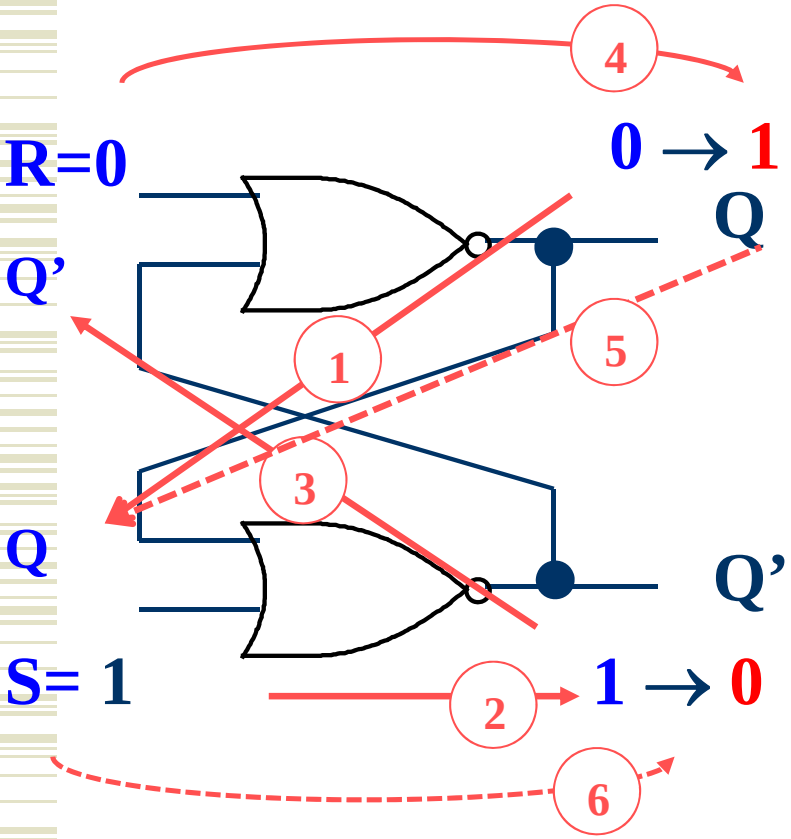
Function Table			
S	**R**	**Qn+1**	**Function**
0	0	Qn	Hold
0	1	0	Reset
1	0	1	Set
1	1	?	Not allowed



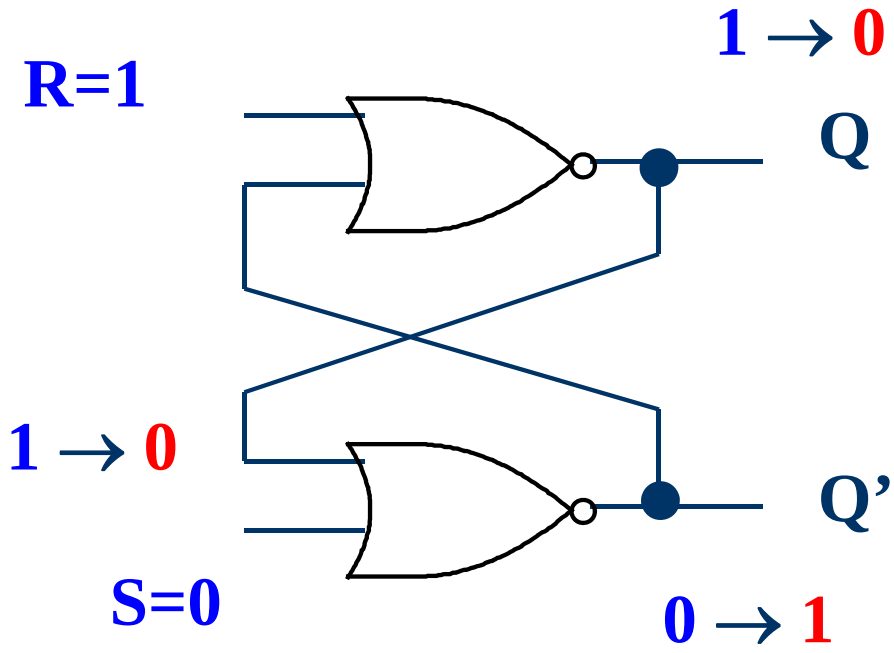
Graphical symbol

- If S & R both 1 at same time, illegal inputs because it will cause Q = Q' = 0

SR Latch Operation



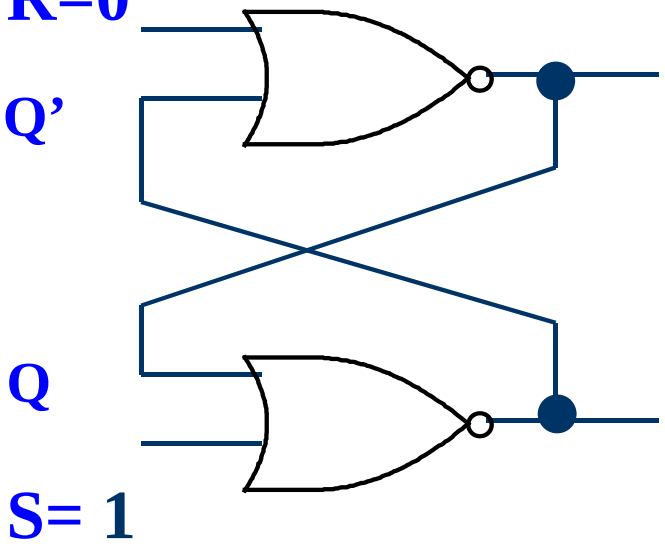
SET operation



RESET operation

SR Latch Operation

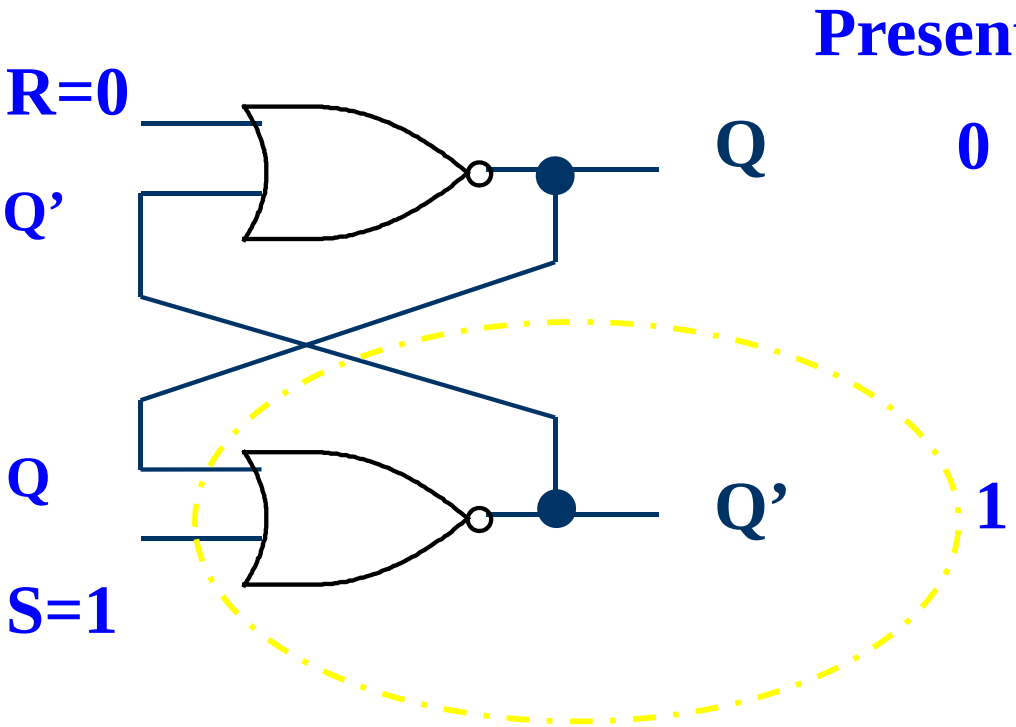
SET operation



	Present State		Next State
Q	0	→	1
Q'	1	→	0

SR Latch Operation

SET operation



Locate NOR with a high "1" input then find the corresponding Q

NOR Gate

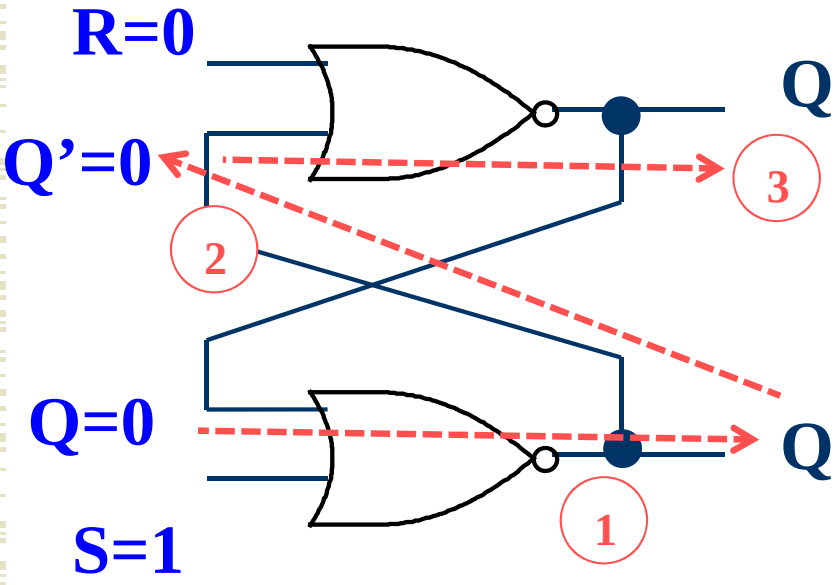
Boolean Expression

Y = (A + B)'

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

SR Latch Operation

SET operation



	Present State		Next State
Q	0	→	1
Q'	1	→	0

Locate NOR with a high "1" input then find the corresponding Q

NOR Gate

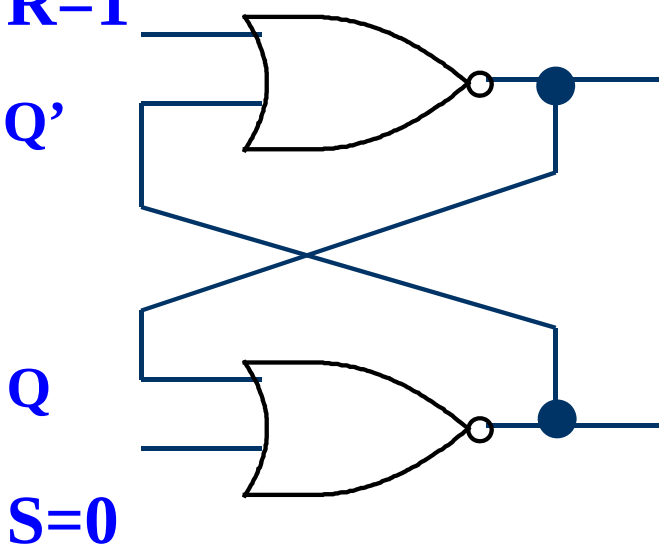
Boolean Expression

Y = (A + B)'

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

SR Latch Operation

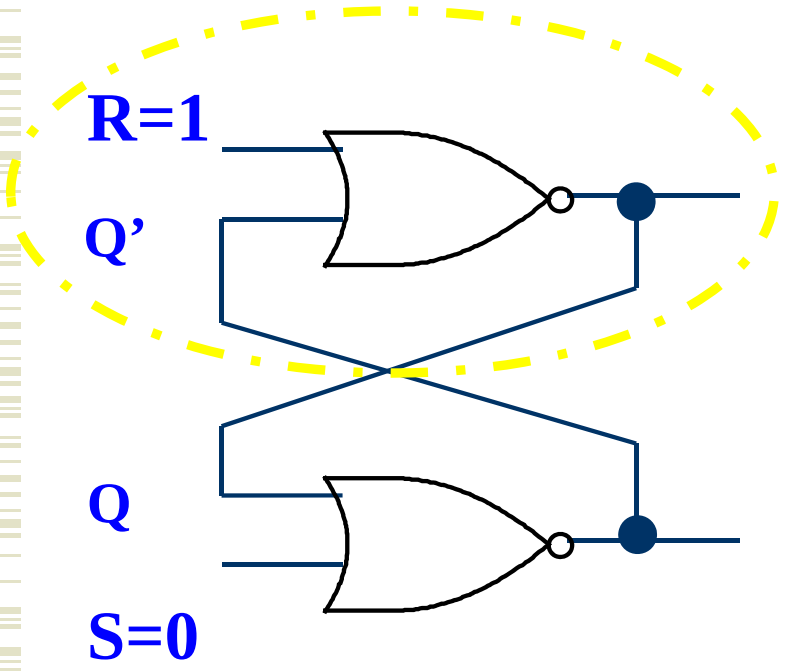
RESET operation



	Present State		Next State
Q	1	→	0
Q'	0	→	1

SR Latch Operation

RESET operation



Present State

1

0

Locate NOR with a high "1" input then find the corresponding Q

NOR Gate

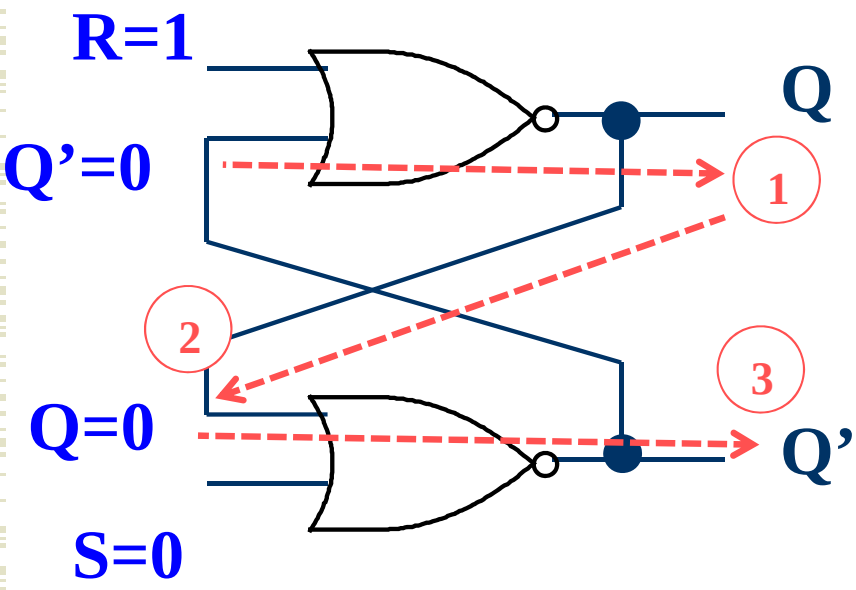
Boolean Expression

Y = (A + B)'

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

SR Latch Operation

RESET operation



Present State		Next State
1	→	0
0	→	1

Locate NOR with a high "1" input then find the corresponding Q

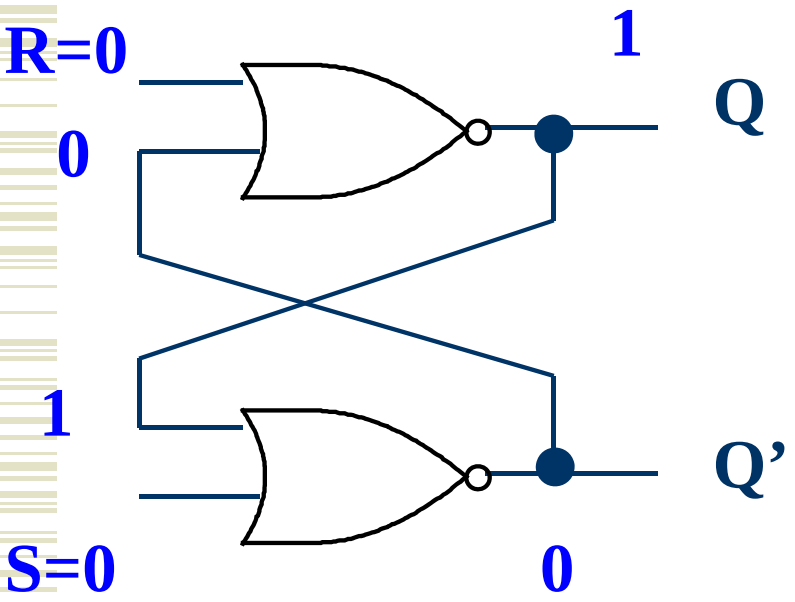
NOR Gate

Boolean Expression

Y = (A + B)'

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

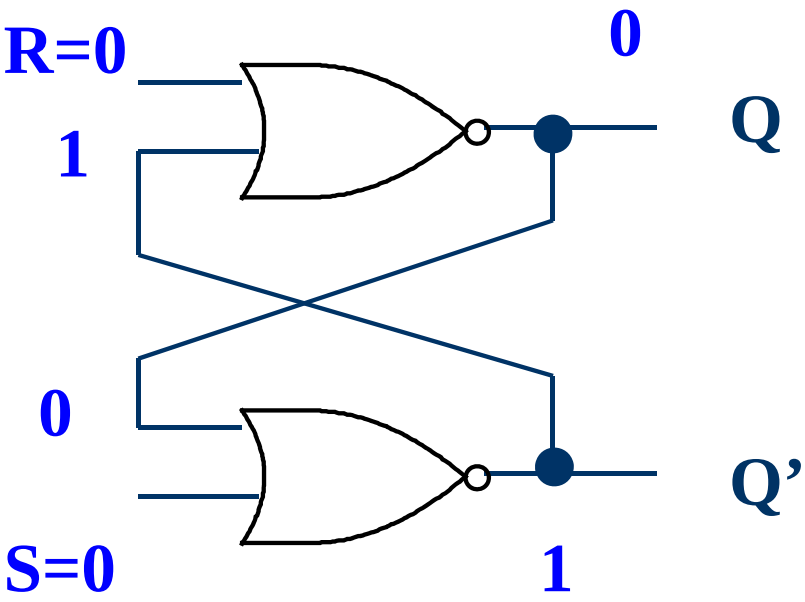
SR Latch Operation (cont)



Stable when

S = R = 0 ,

Q = 1 , Q'=0

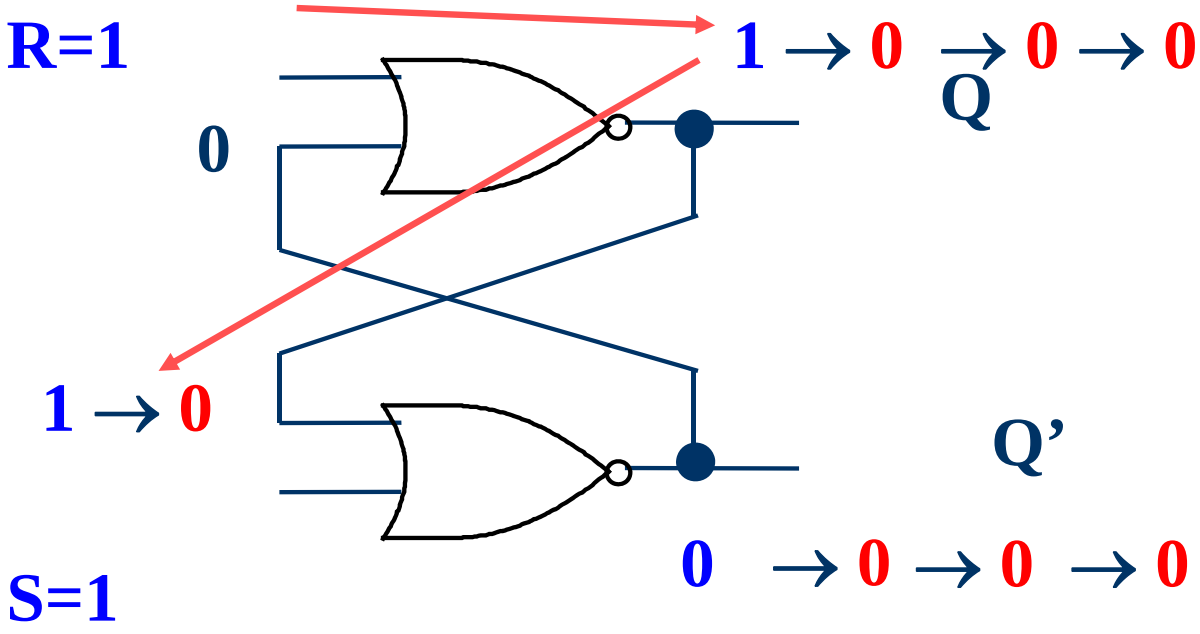


Stable when

S = R = 0 ,

Q =0 , Q' = 1

SR Latch Operation (cont)



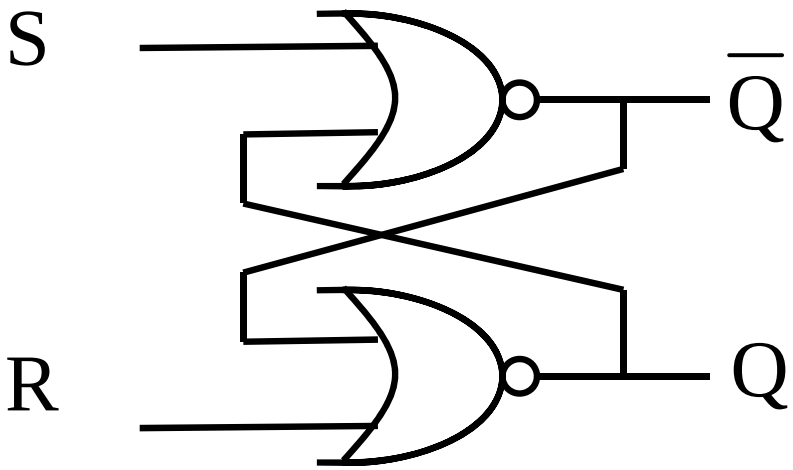
What about S=R=1? (Set,Reset both true?)

Assume that S, R both transition to 1 simultaneously. Q becomes '0', but Q' remains '0'! Outputs are no longer complements of each other.

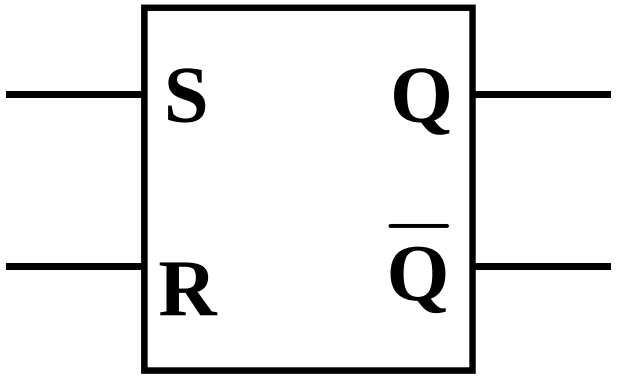
This is an INVALID STATE.

S-R LATCH

- S-R using **NOR** gates (ACTIVE-HIGH INPUT)



S-R Latch Logic Circuit



S-R Latch Logic Symbol

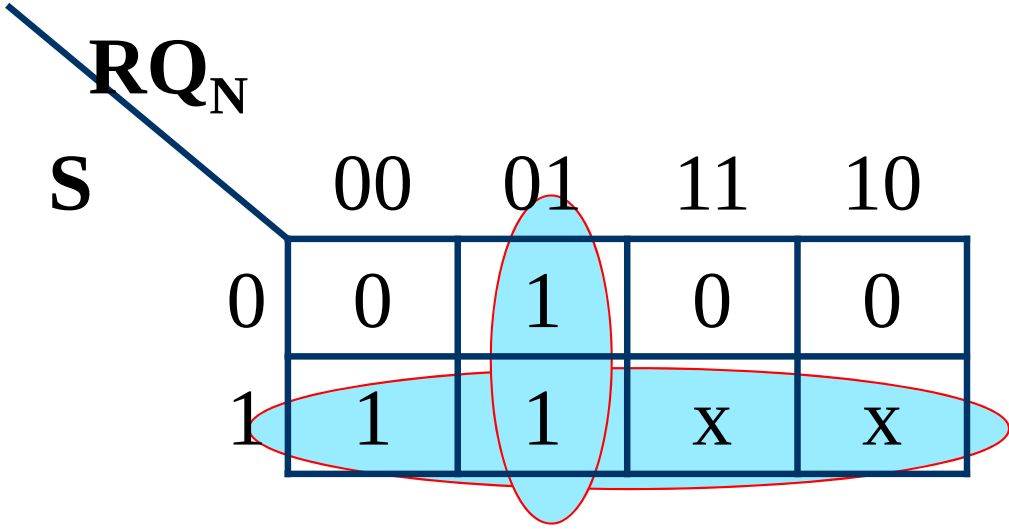
S-R Latch Truth Table

S (Set)	R (Reset)	Output
0	0	No Change
1	0	Q = 1
0	1	Q = 0
1	1	Invalid

S-R LATCH

- S-R using **NOR** gates (ACTIVE-HIGH INPUT)

S	R	Q_N	Q_{N+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	X
1	1	1	X

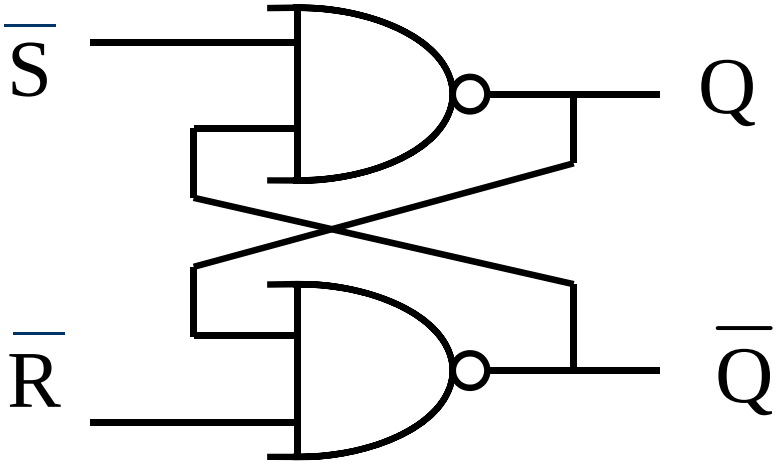


Characteristic equation

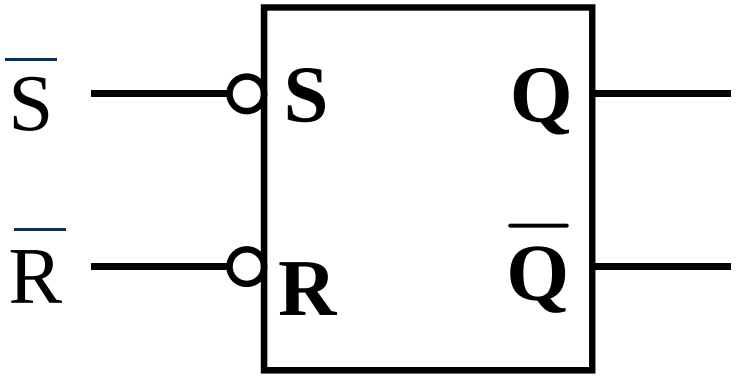
$$Q_{N+1} = S + \overline{R}Q_N$$

S-R Latch (NAND)

- $\overline{S}$-$\overline{R}$ Latch using **NAND** gates (ACTIVE-LOW INPUT)



$\overline{S}$-$\overline{R}$ Latch Logic Circuit



$\overline{S}$-$\overline{R}$ Latch Logic Symbol

$\overline{S}$-$\overline{R}$ Latch Truth Table

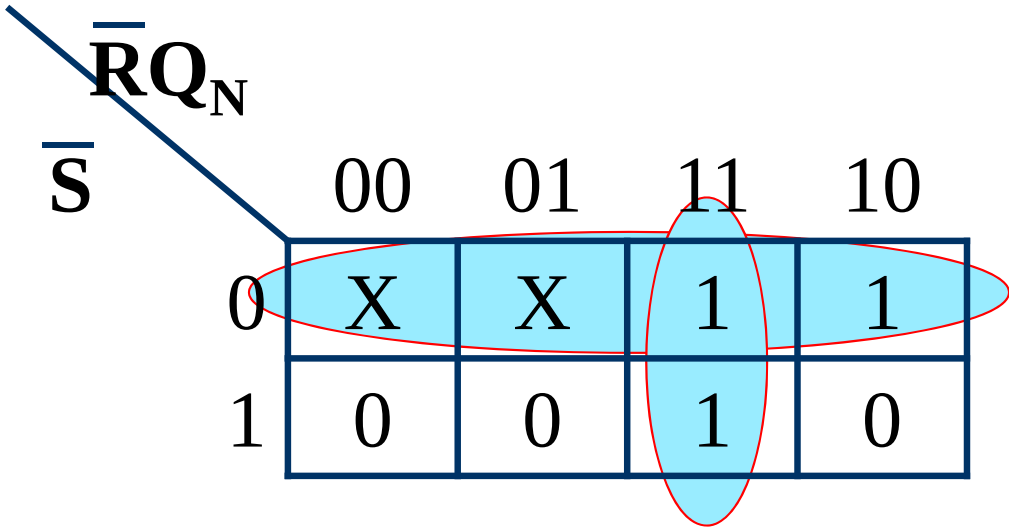
This is the S'-R' table. If S-R table, it will be the same as S-R NOR.

$\overline{S}$ (Set)	$\overline{R}$ (Reset)	Q (Output)
0	0	Invalid
0	1	Q = 1
1	0	Q = 0
1	1	No change

S-R Latch (NAND)

- $\overline{S}$-$\overline{R}$ Latch using **NAND** gates (**ACTIVE-LOW** INPUT)

$\overline{S}$	$\overline{R}$	Q_N	Q_{N+1}
0	0	0	X
0	0	1	X
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

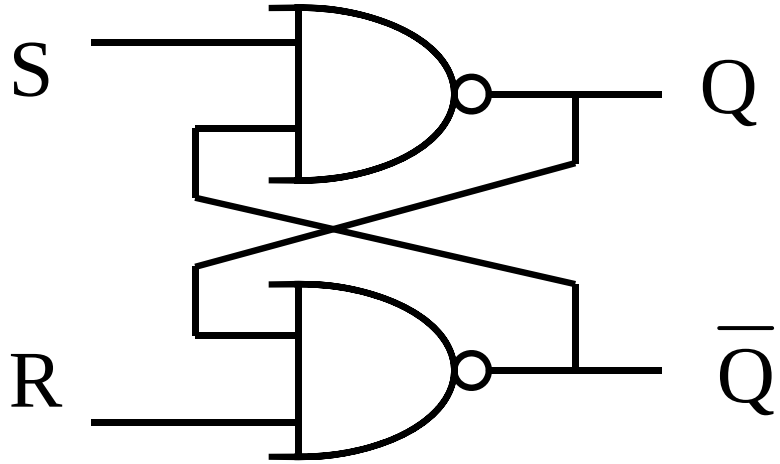


Characteristic equation

$$Q_{N+1} = S + \overline{R}Q_N$$

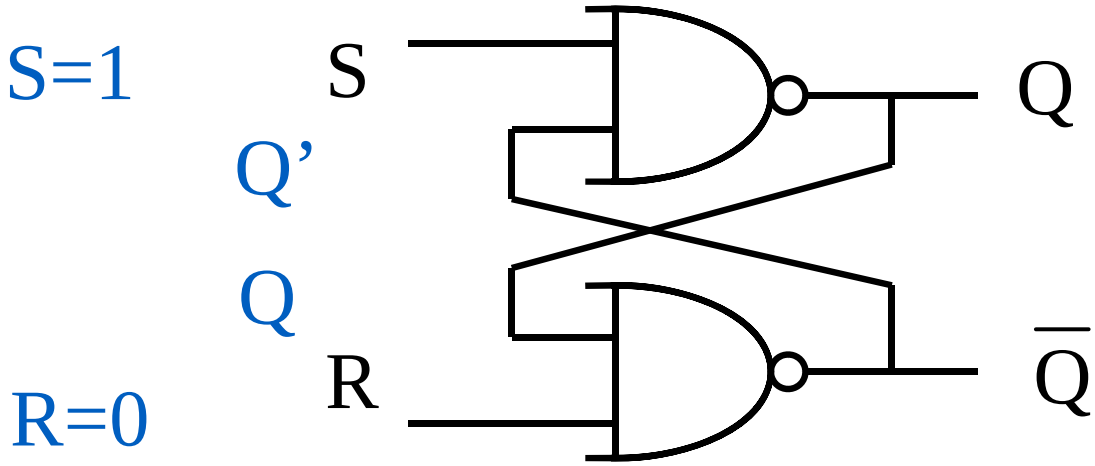
Exercise

- Complete the truth table for this latch



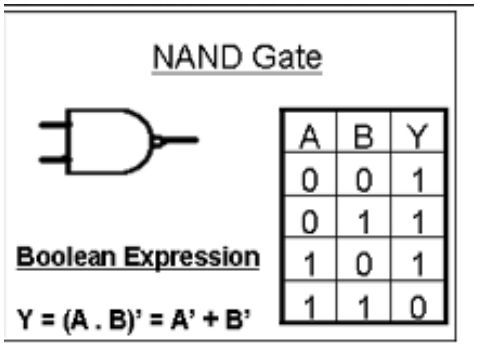
Exercise

- Complete the truth table for this latch



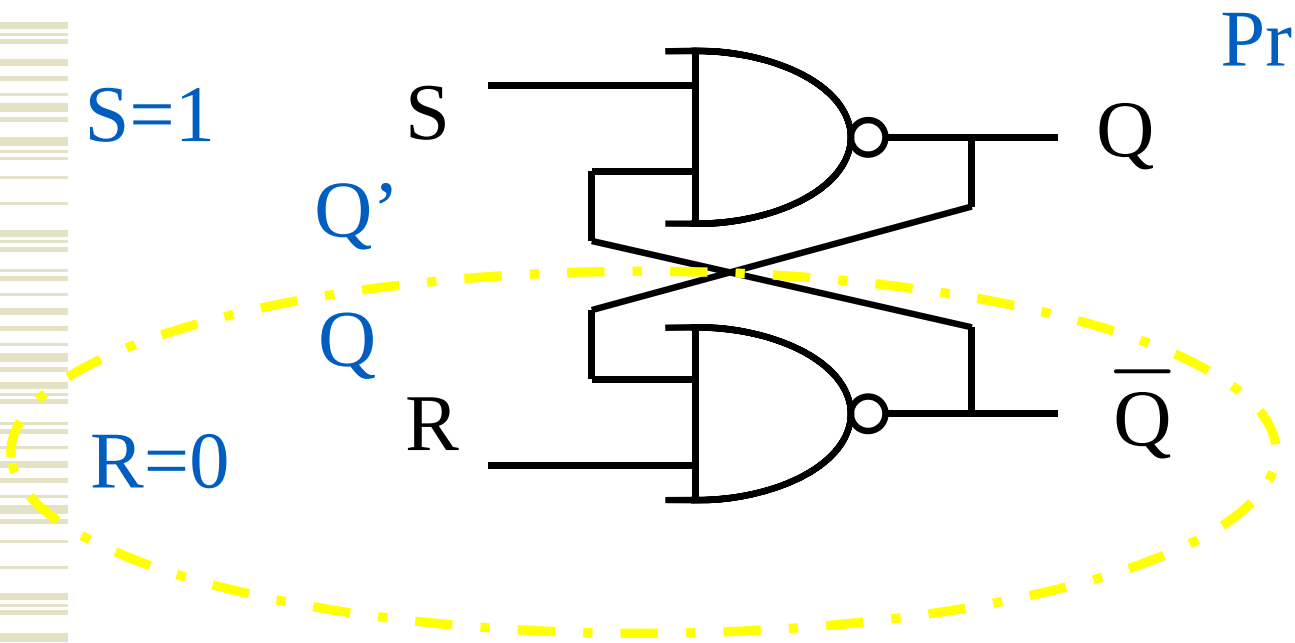
Present State	Next State
0	?
1	?

Condition when S=1 and R=0



Exercise

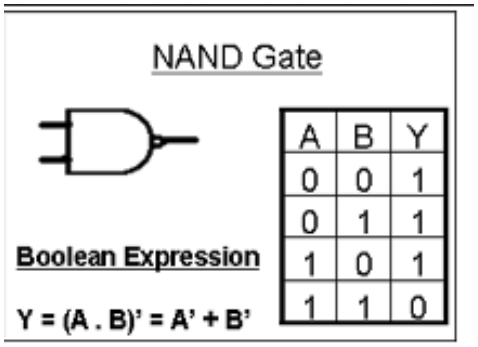
- Complete the truth table for this latch



Present State	Next State
1	?
0	?

Locate NAND with a low "0" input then find the corresponding Q

Condition when S=1 and R=0



Exercise

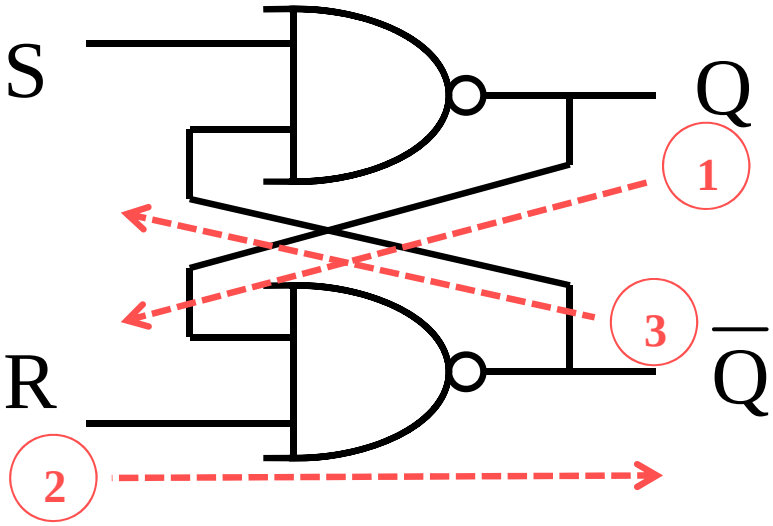
- Complete the truth table for this latch

S=1

Q'=1

Q=1

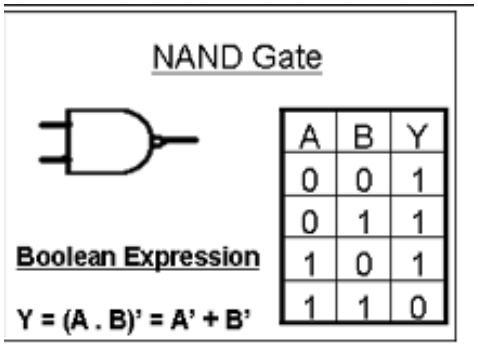
R=0



Present State	Next State
1	0
0	1

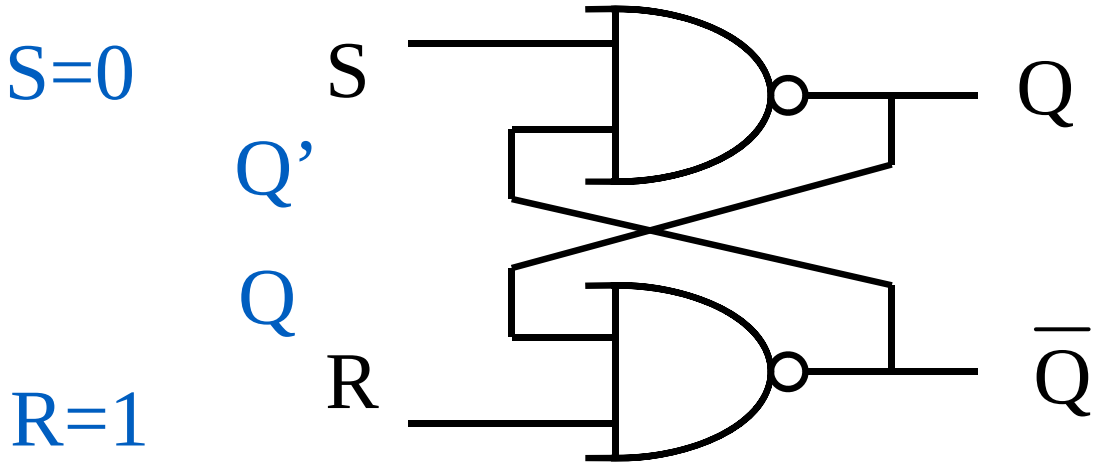
Locate NAND with a low "0" input then find the corresponding Q

Condition when S=1 and R=0



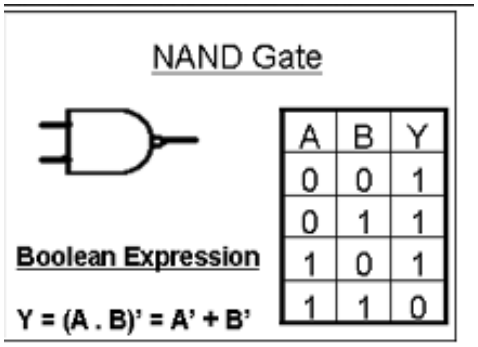
Exercise

- Complete the truth table for this latch



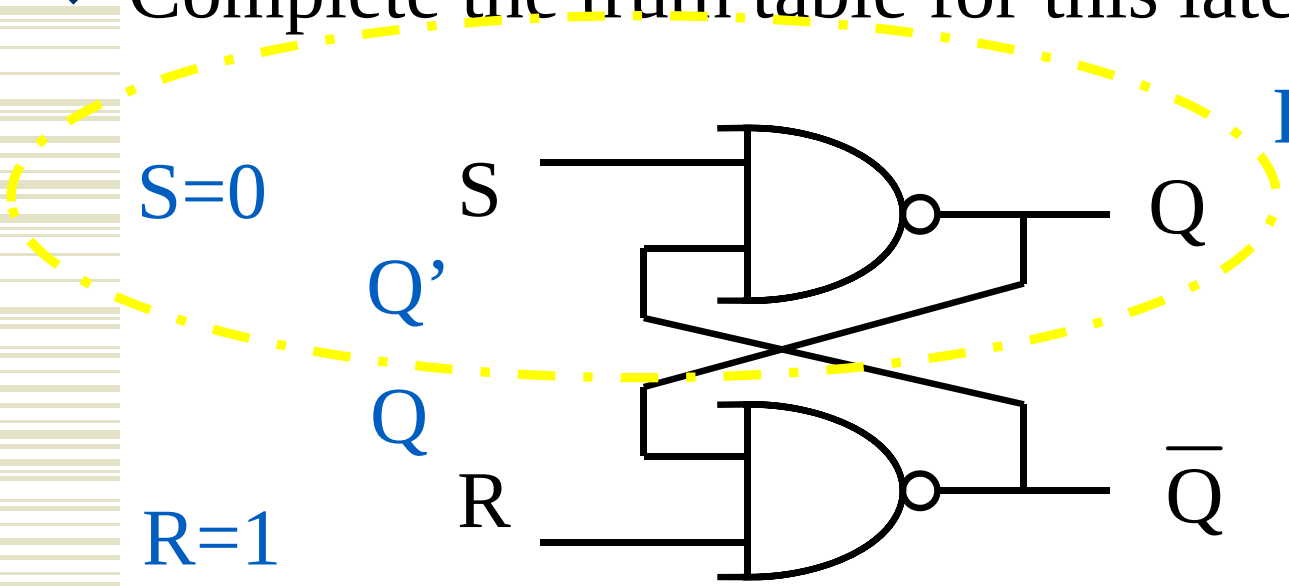
Present State	Next State
0	?
1	?

Condition when S=0 and R=1



Exercise

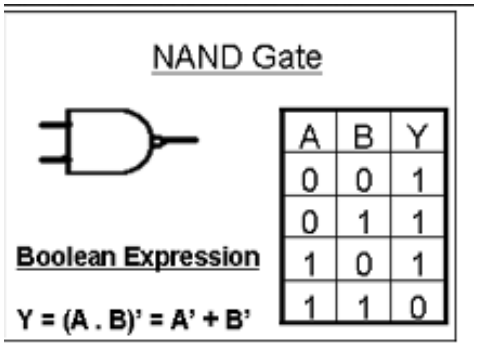
- Complete the truth table for this latch



Present State	Next State
0	?
1	?

Locate NAND with a low "0" input then find the corresponding Q

Condition when S=0 and R=1



Exercise

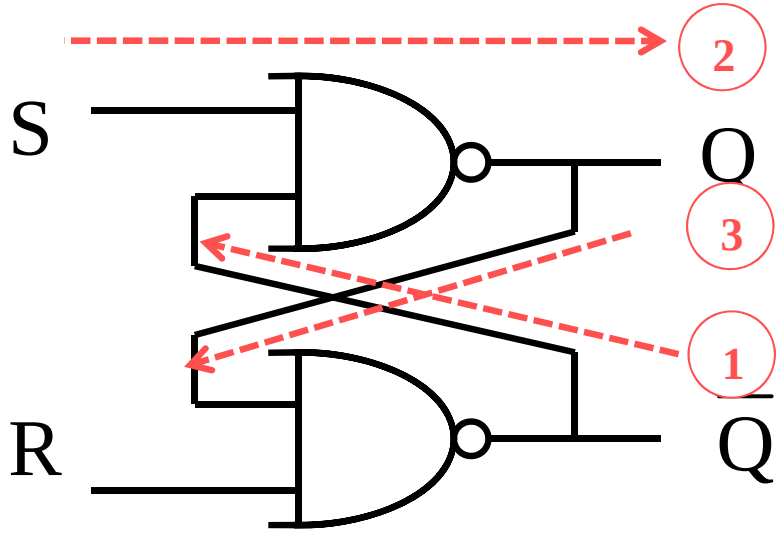
- Complete the truth table for this latch

S=0

Q'=1

Q=1

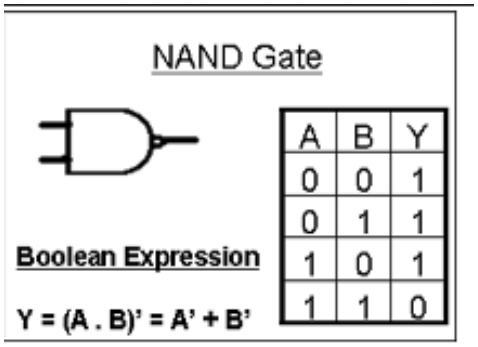
S=1



Present State	Next State
0	1
1	0

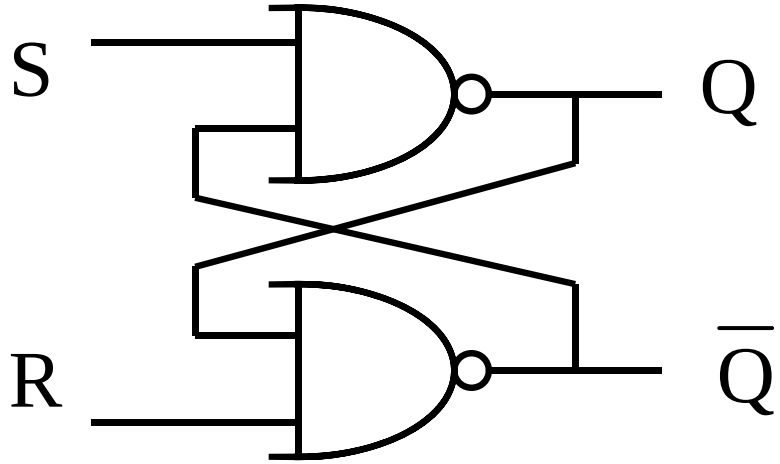
Locate NAND with a low "0" input then find the corresponding Q

Condition when S=0 and R=1



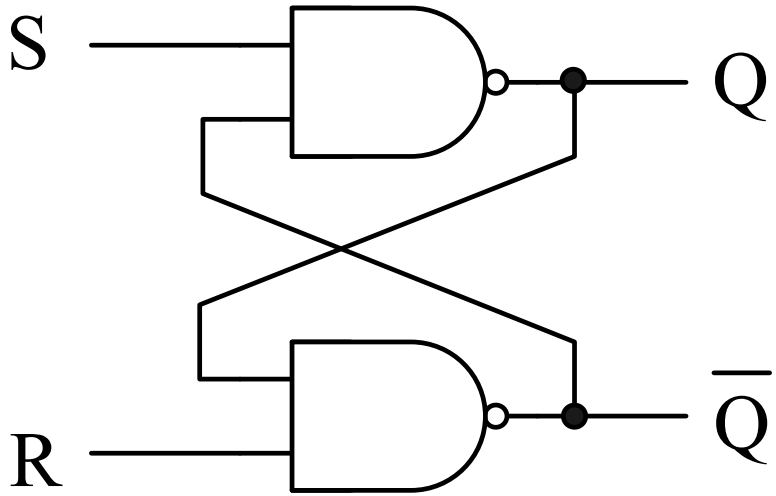
Exercise

- Complete the truth table for this latch



Solution Exercise

- S-R latch (**NAND**) with **active high** inputs.



S	R	Q_{n+1}
0	0	Illegal inputs
0	1	1
1	0	0
1	1	Q_n

Please correct this table in your printed slide

S-R Latch Summary

- S-R Latch in any configuration will
  - SET when S = 1 and R = 0 (or when S’ = 0 and R’ = 1)
  - SET means output Qn+1 is 1.
  - RESET when R = 1 , S = 0 (or when R’ = 0 and S’ = 1)
  - RESET means output Qn+1 is 0.

Gated S-R Latch

- S-R latch with trigger input, E
- High – level triggering (active high)
  - E=1: latch is enabled and output(future state) depends on present state and the inputs
  - E=0 : latch is disabled and no change to outputs
- Low-level triggering (active low)
  - E=1: latch is disabled and no change to outputs
  - E=0 : latch is enabled and output(future state) depends on present state and the inputs

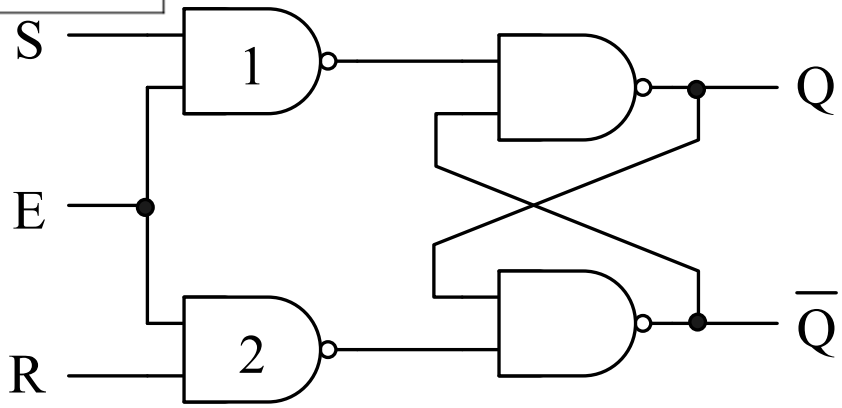
Gated S-R Latch

NAND Gate

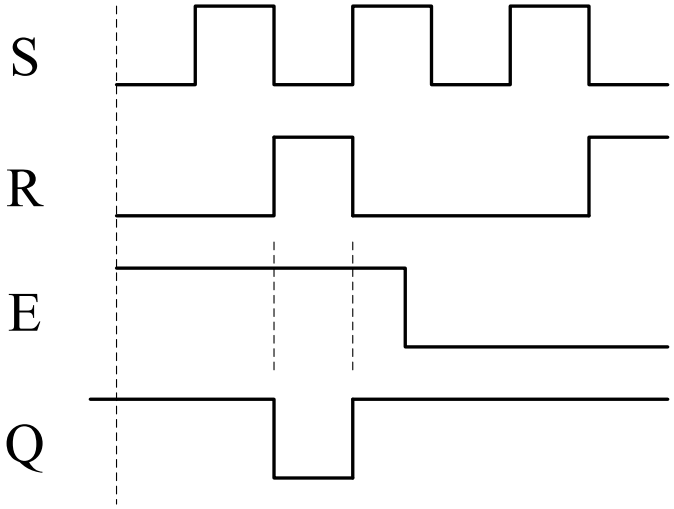
A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

Boolean Expression

$Y = (A \cdot B)' = A' + B'$



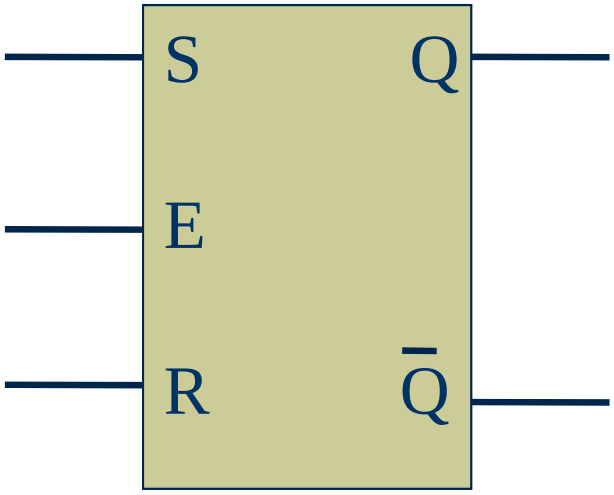
(a) HIGH level triggering input



(b) Example of timing diagram

E	R	S	Q_{N+1}
0	X	X	Q_N
1	0	0	Q_N
1	0	1	1
1	1	0	0
1	1	1	Illegal inputs

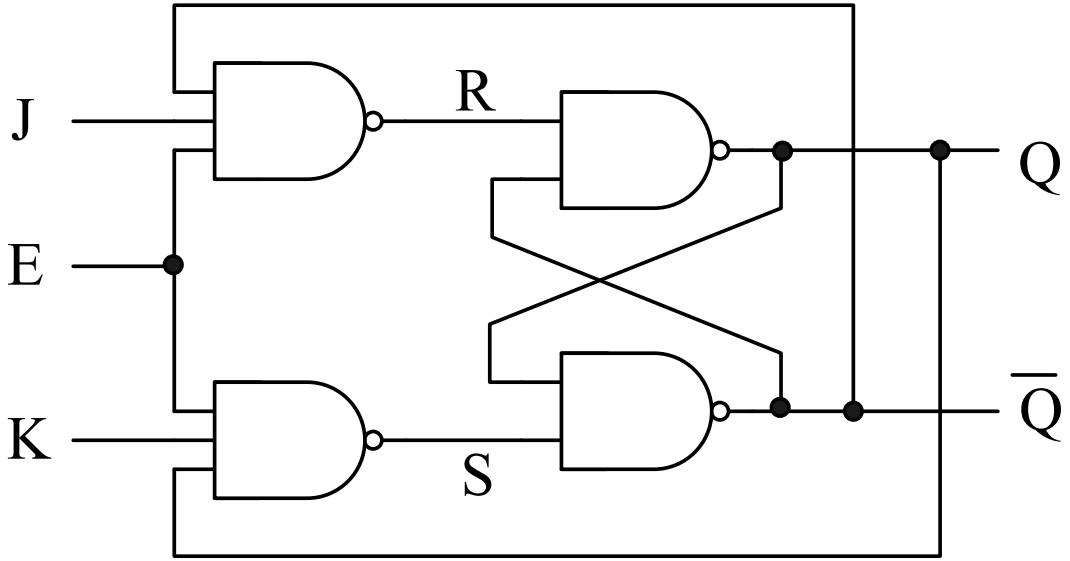
(c) Truth-table



(d) Logic symbol

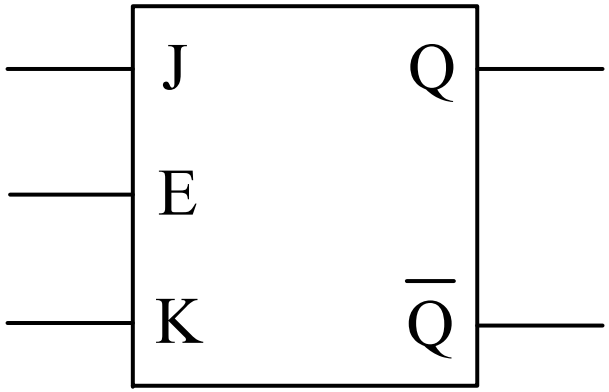
Gated J-K LATCH

- Solve the problem of illegal inputs in SR latch.



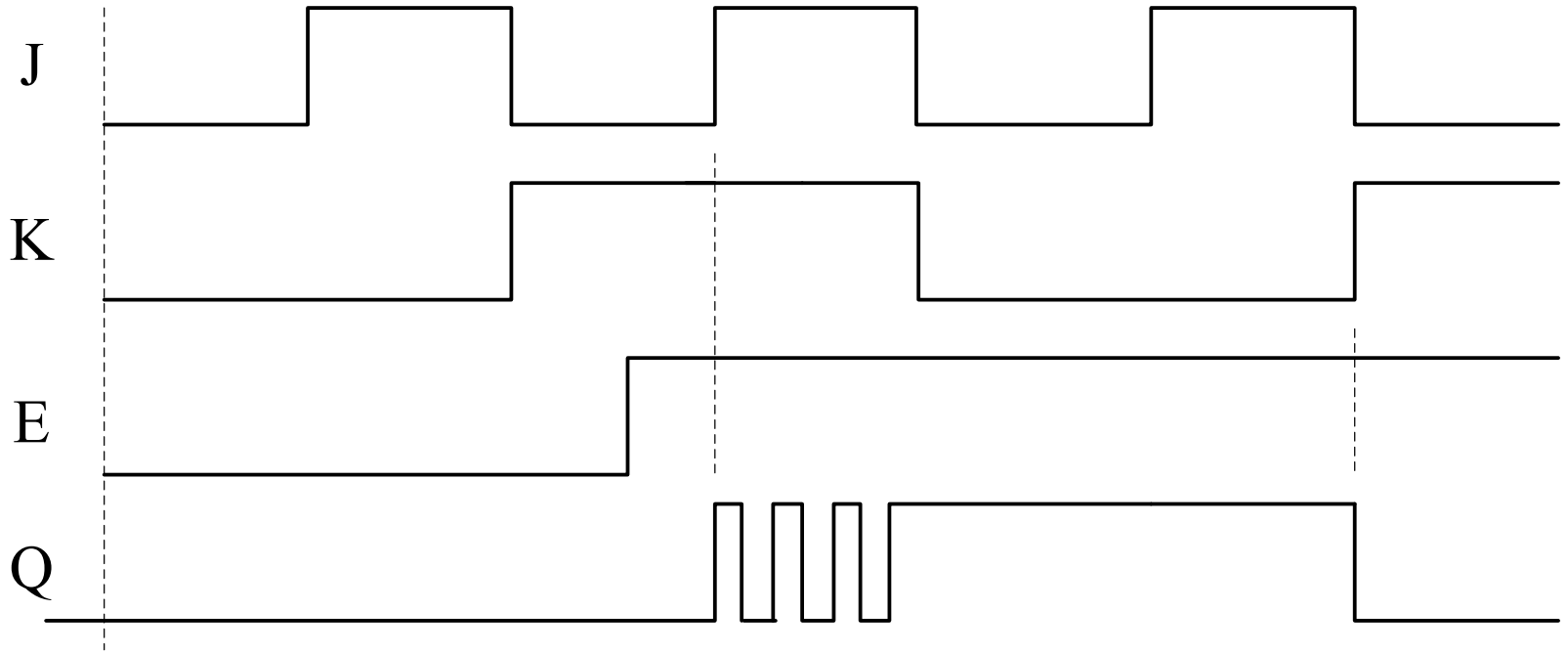
J-K Latch Logic Circuit

Gated J-K LATCH



E	J	K	Q_{n+1}
0	X	X	Q_n (no change)
1	0	0	Q_n (no change)
1	0	1	0 (reset)
1	1	0	1 (set)
1	1	1	$\overline{Q}_n$ (toggle)

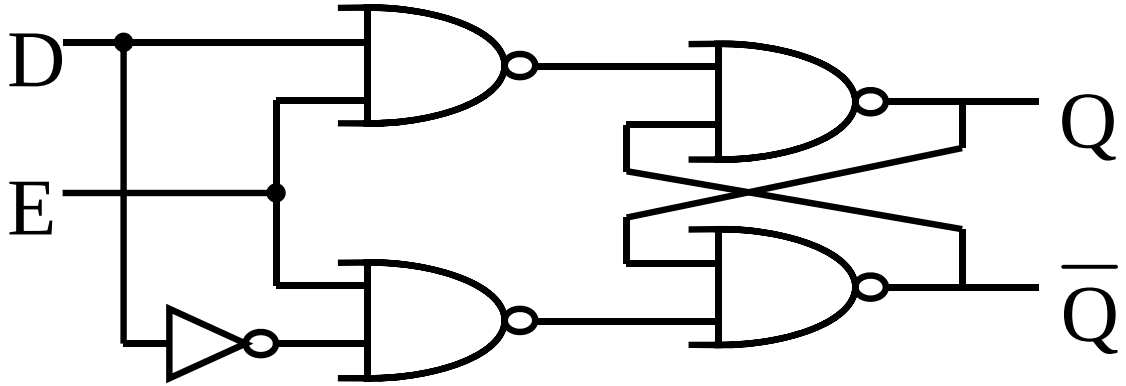
Gated J-K LATCH



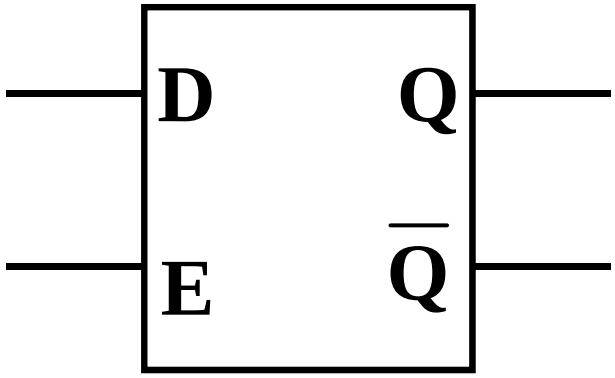
Weakness of level triggering : If the duration of triggering is long, the output may toggle a few times when J = K = 1.

D LATCH

- D Latch using S-R NAND implementation!



D Latch Logic Circuit



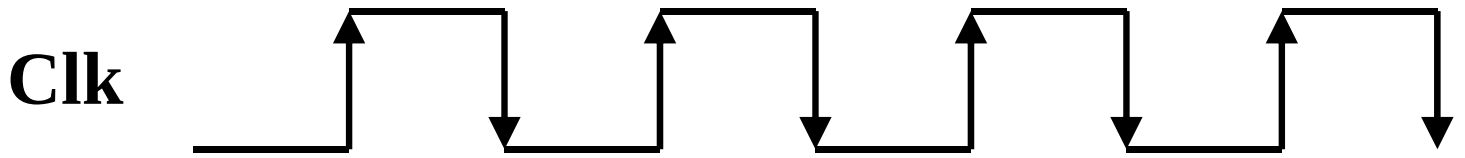
D Latch Logic Symbol

D Latch Truth Table

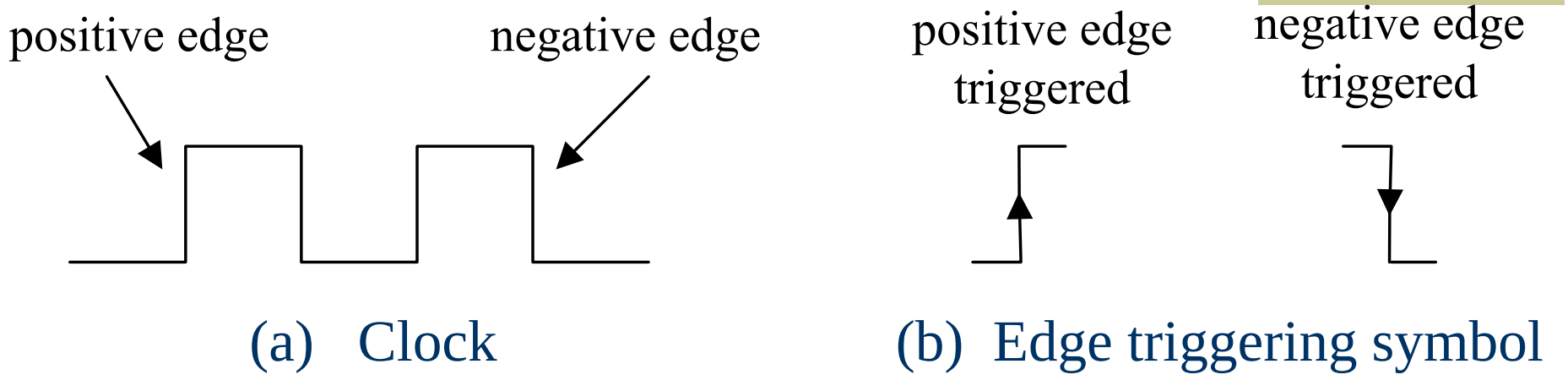
E	D	Q	Q_{n+1}
0	x	0	0
0	x	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Flip-Flop

- To synchronize a circuit, LATCH cannot be used. Thus, an EDGE-TRIGGERED FLIP-FLOP is used to synchronize the changes at the output state either at the positive edge or negative edge of the CLOCK (CLK).
- As mentioned before, CLK is a control input signal used such that the output is executed at a specified time/point.
- Figure below shows the positive (rising edge) and negative edge (falling edge) of the CLK signal.

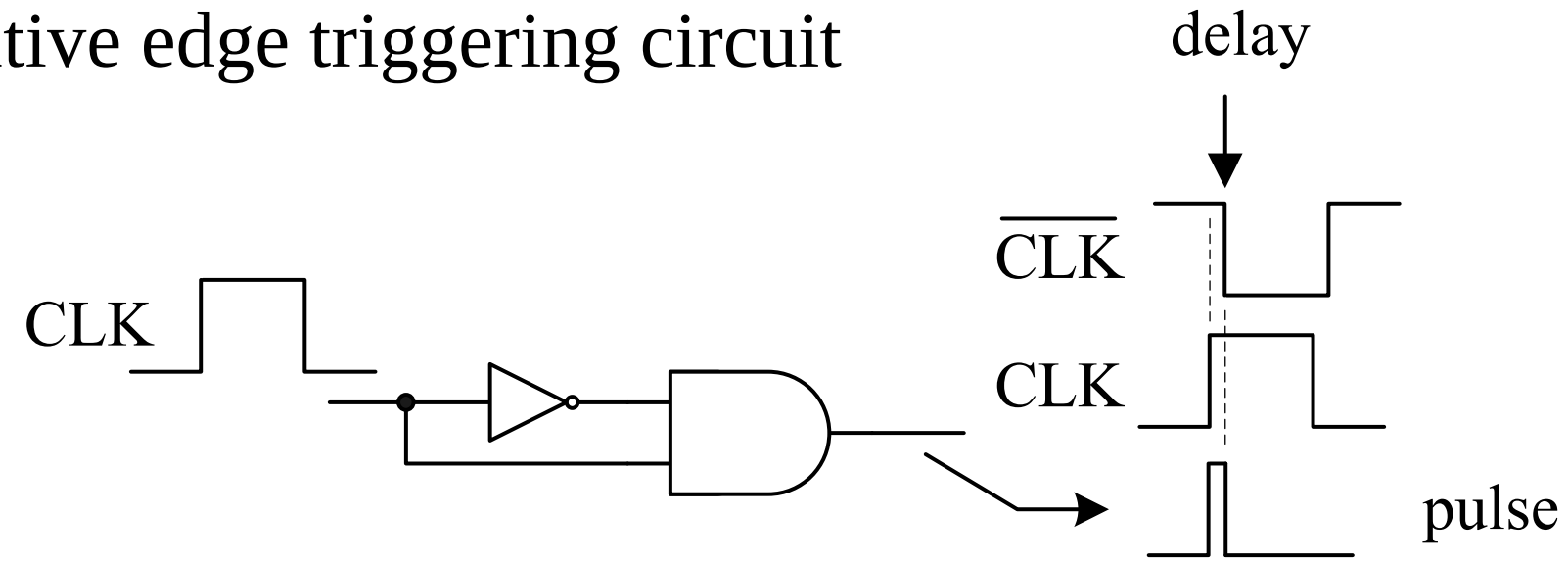


Flip-Flop



(a) Clock (b) Edge triggering symbol

- Positive edge triggering circuit



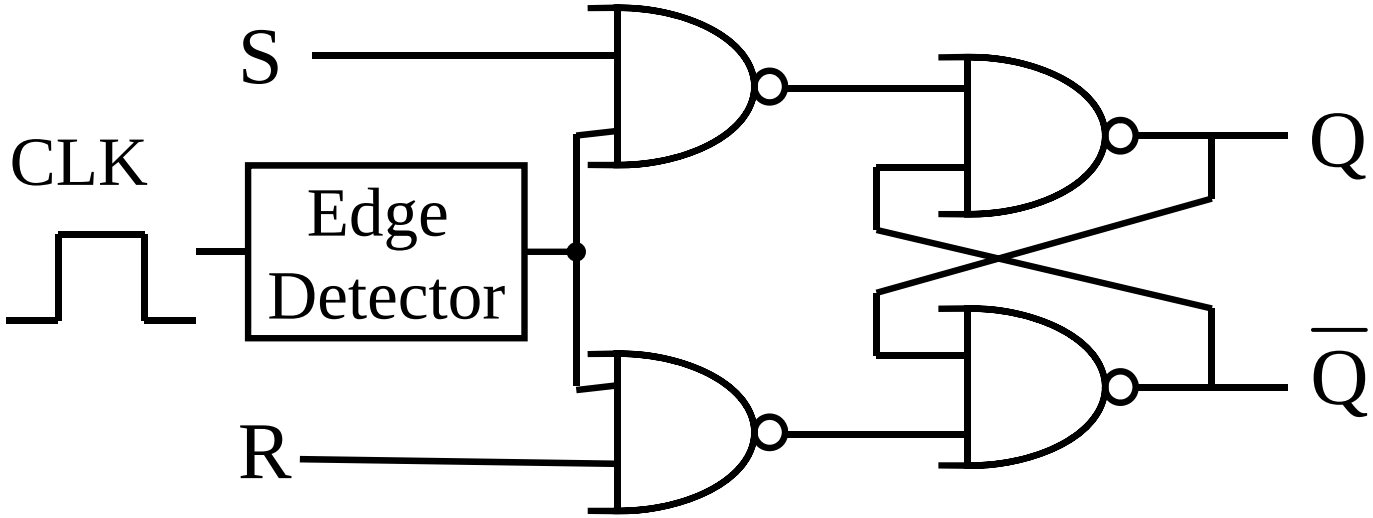
Exercise

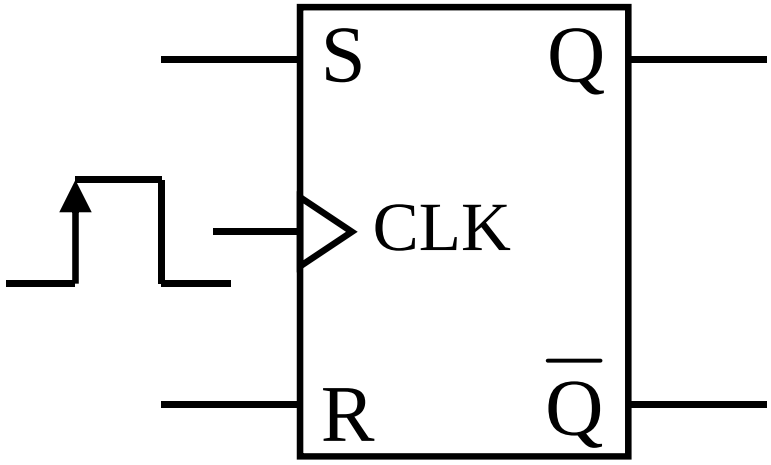
- Design a negative edge triggering circuit

- Ans : Change AND gate to NAND gate

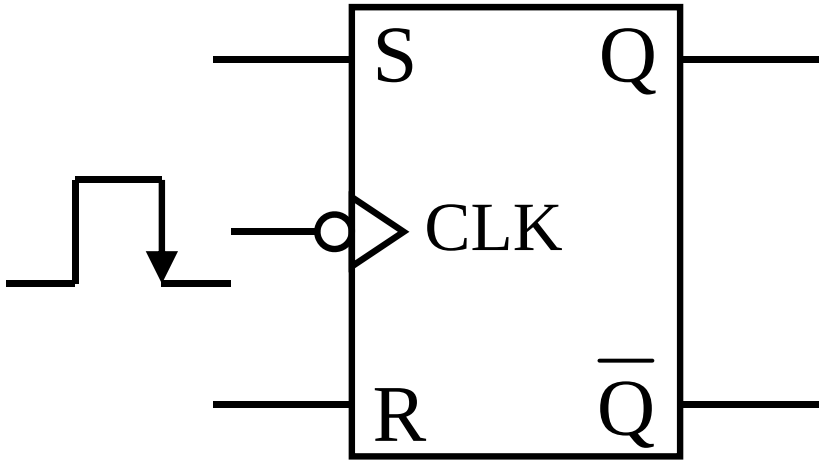
S-R FLIP-FLOP

- S-R Flip-Flop





S-R FF (+ve Edge-triggered)



S-R FF (-ve Edge-triggered)

S-R FLIP-FLOP

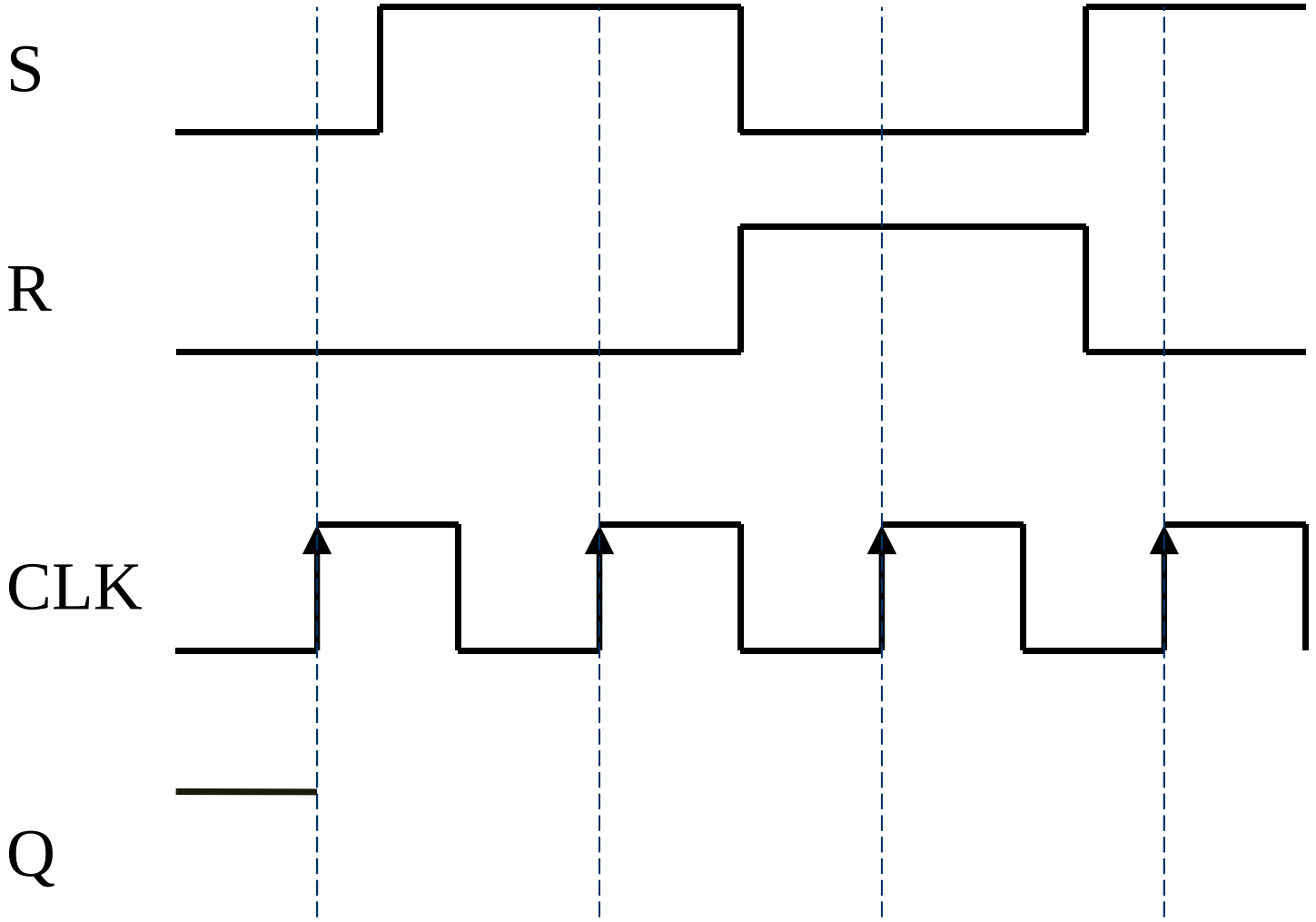
- Truth-table and Transition-table for S-R FF

S	R	Q_N	Q_{N+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	Invalid
1	1	1	Invalid

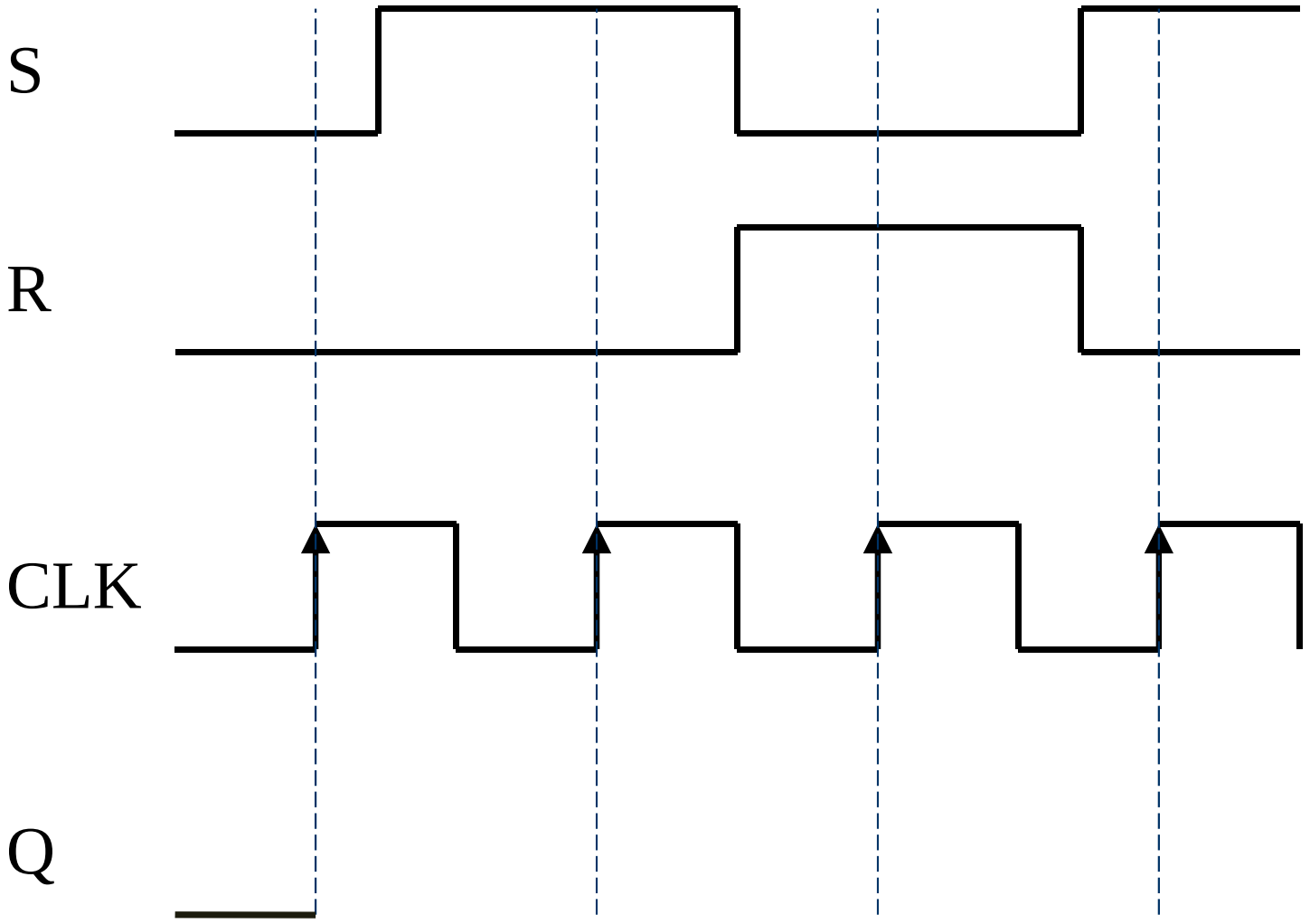
OUTPUT TRANSITION			FF INPUT	
Q_N		Q_{N+1}	S	R
0	→	0	0	X
0	→	1	1	0
1	→	0	0	1
1	→	1	X	0

$$Q_{N+1} = S + \overline{R}Q_N$$

Exercise S-R FF

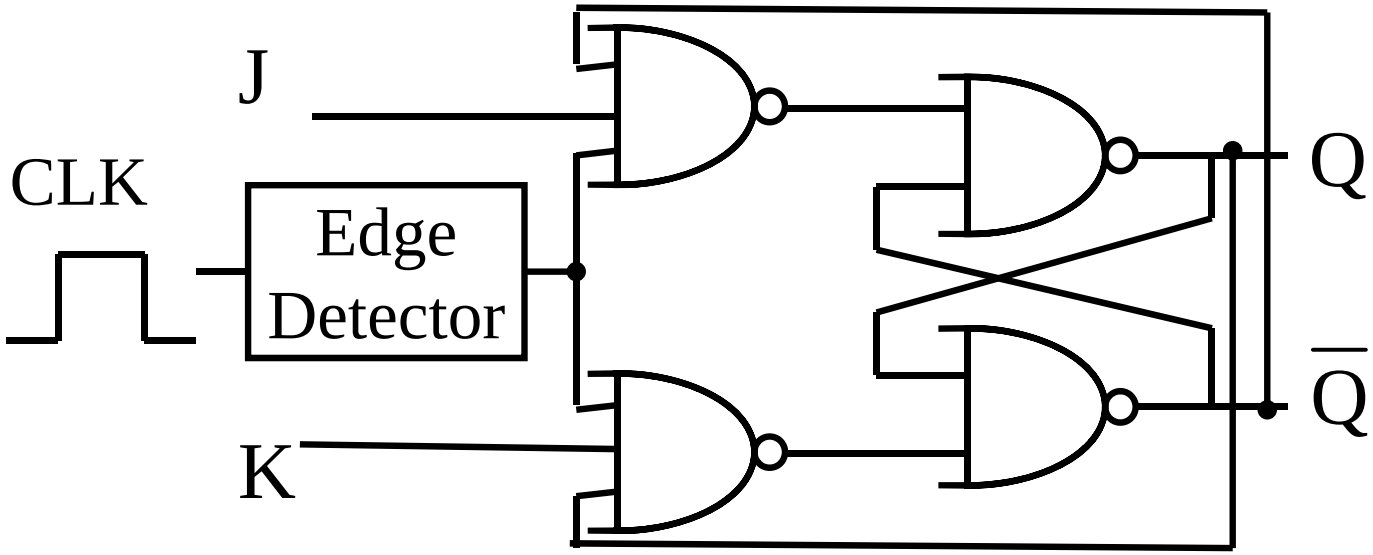


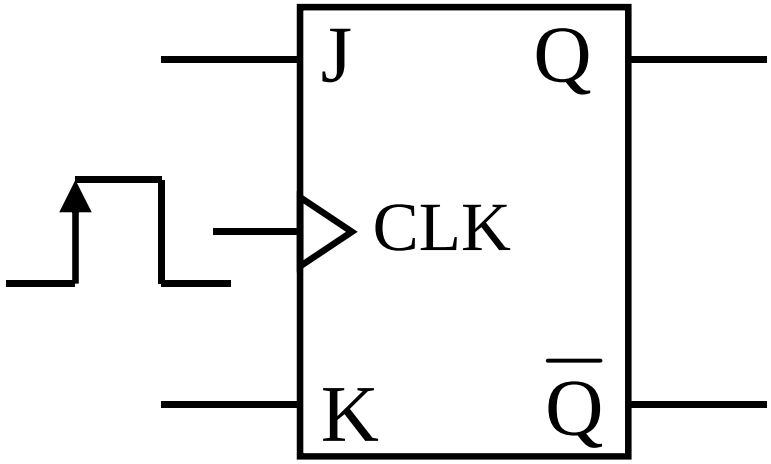
Exercise S-R FF



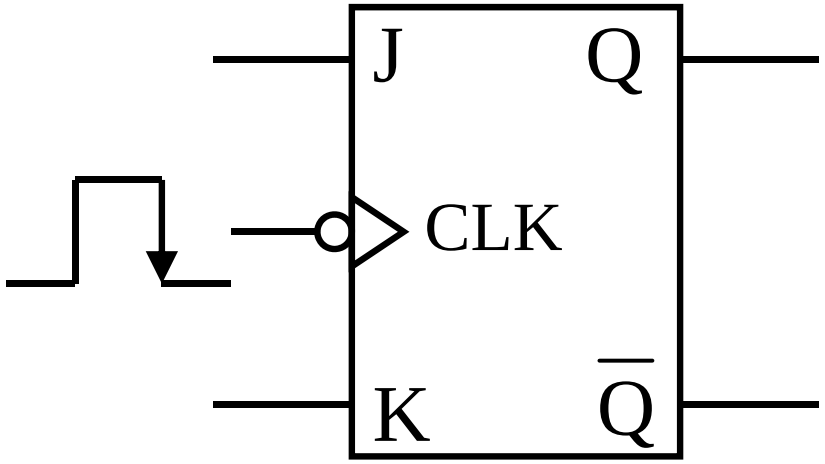
J-K FLIP-FLOP

- J-K Flip-flop





J-K FF (+ve Edge-triggered)

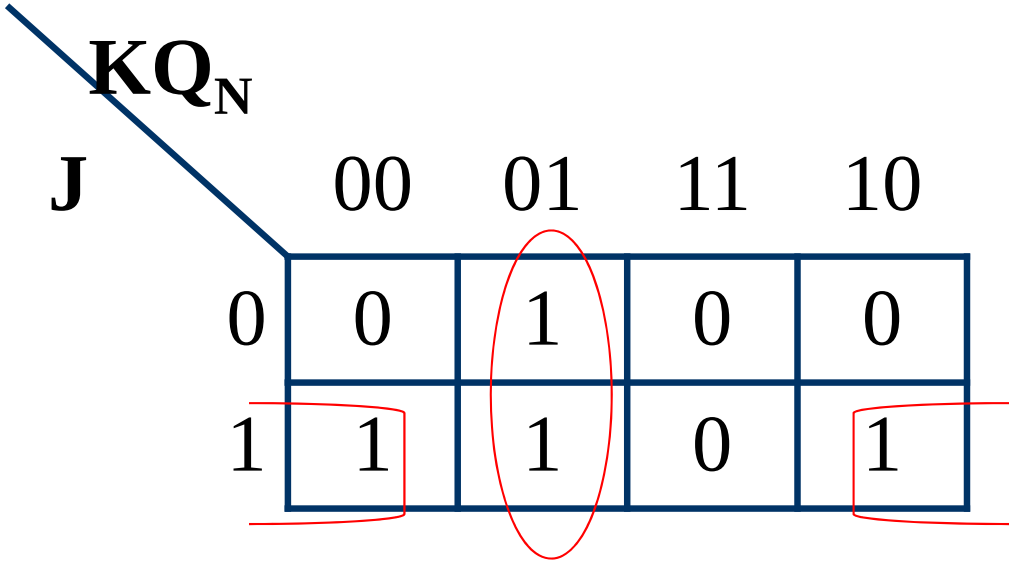


J-K FF (-ve Edge-triggered)

J-K FLIP-FLOP

- Truth table and Characteristic equation for J-K FF

J	K	Q_N	Q_{N+1}	
0	0	0	0	
0	0	1	1	
0	1	0	0	
0	1	1	0	
1	0	0	1	
1	0	1	1	
1	1	0	1	$\overline{Q}$
1	1	1	0	



Characteristic equation

$$Q_{N+1} = J\overline{Q_N} + \overline{K}Q_N$$

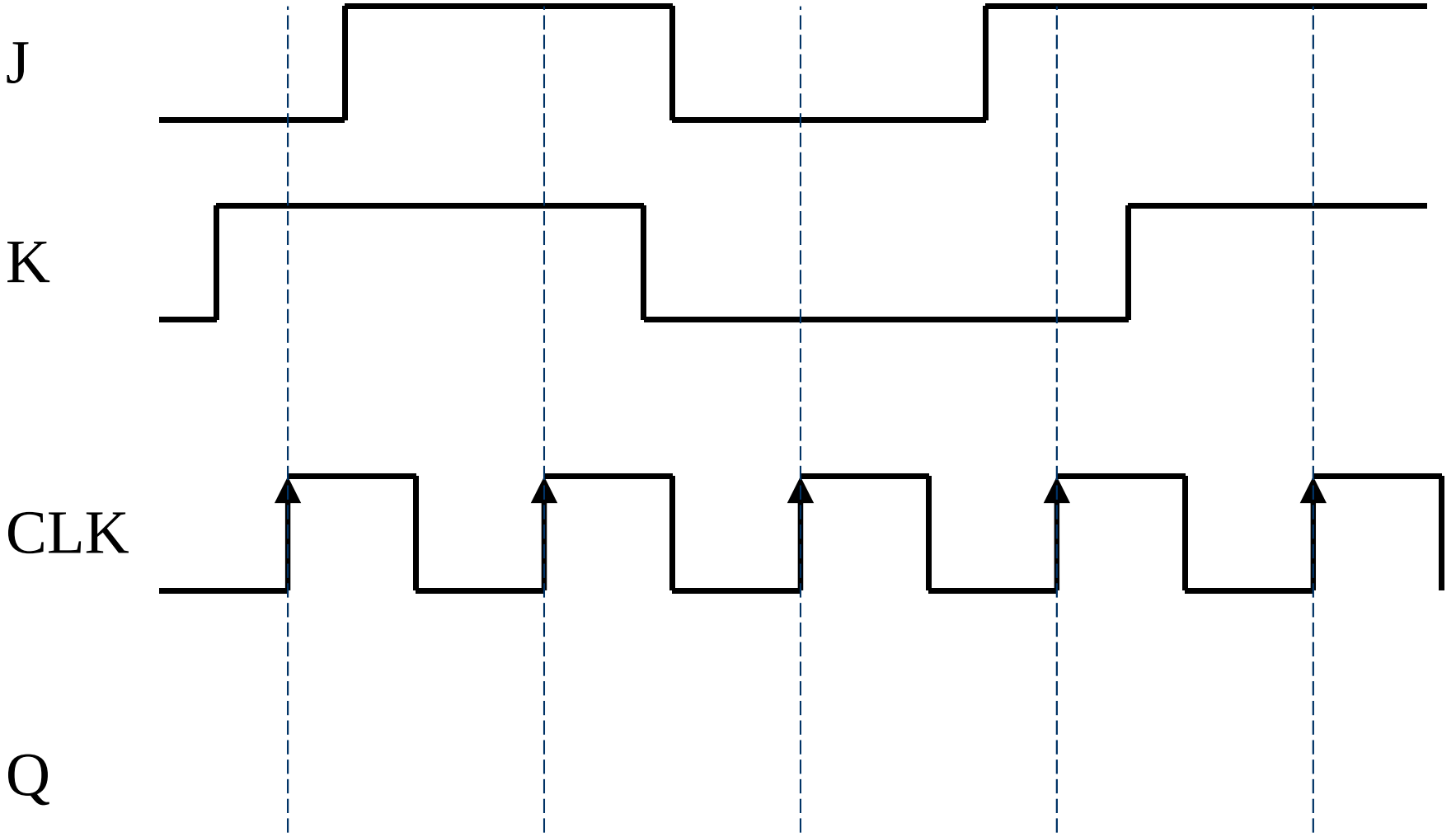
J-K FF improve the 'Invalid' condition in S-R FF

J-K FLIP-FLOP

- J-K FF Transition Table

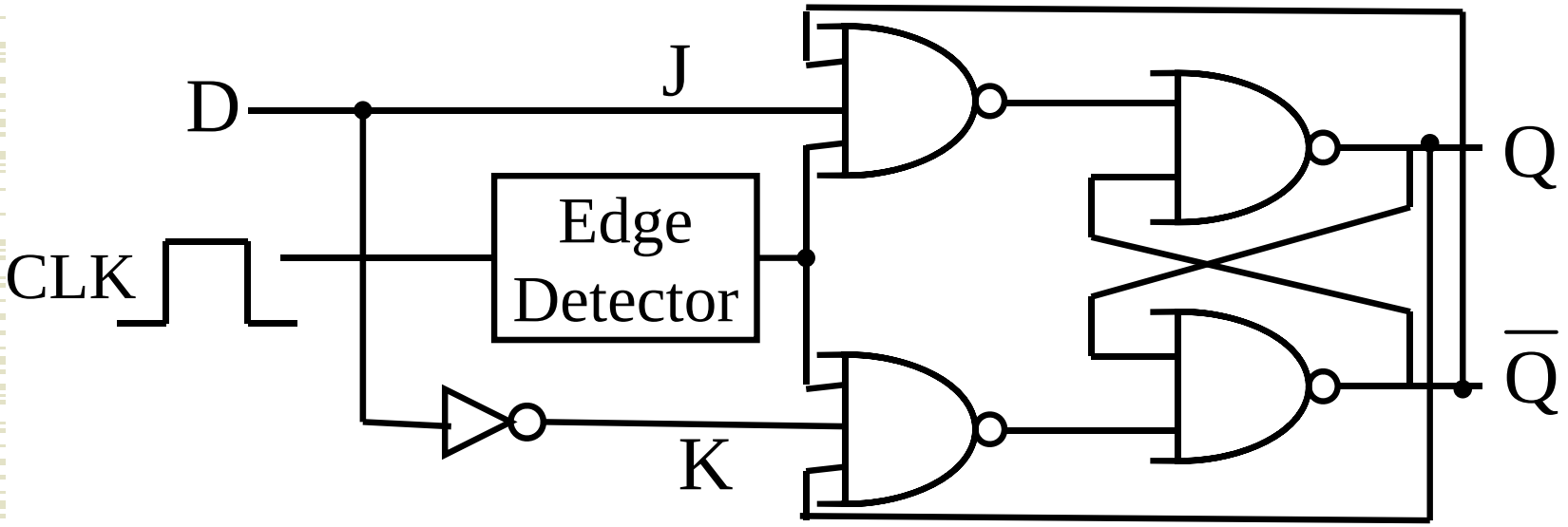
OUTPUT TRANSITION			FF INPUT	
Q_N		Q_{N+1}	J	K
0	→	0	0	X
0	→	1	1	X
1	→	0	X	1
1	→	1	X	0

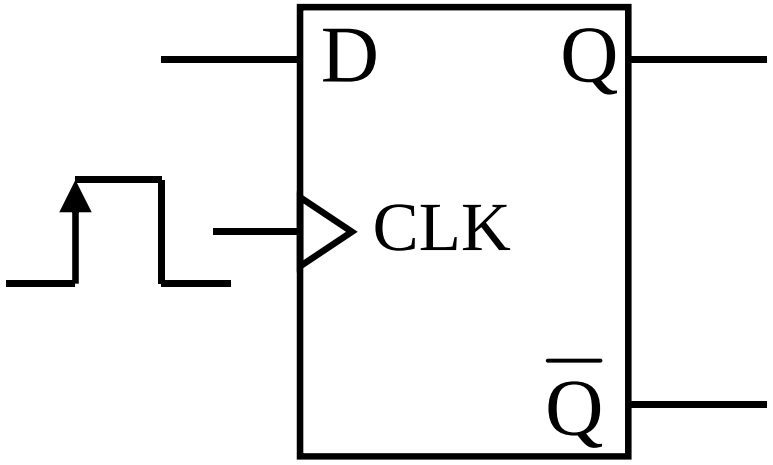
J-K FLIP-FLOP



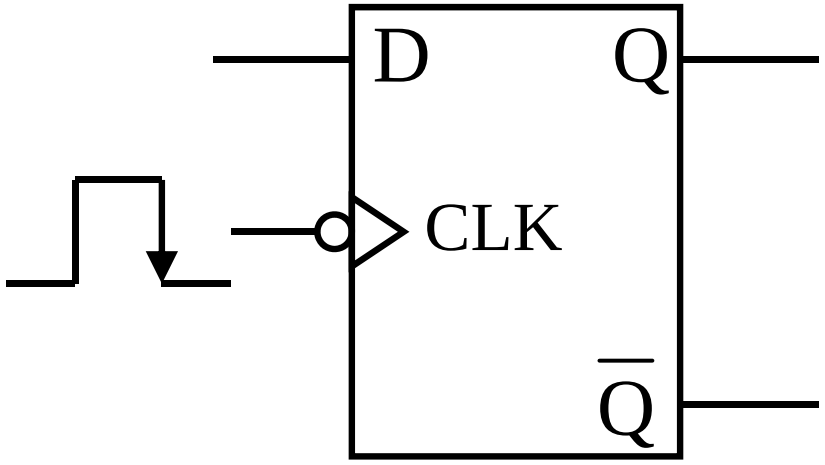
D FLIP-FLOP

- D Flip-flop





D FF (+ve Edge-triggered)

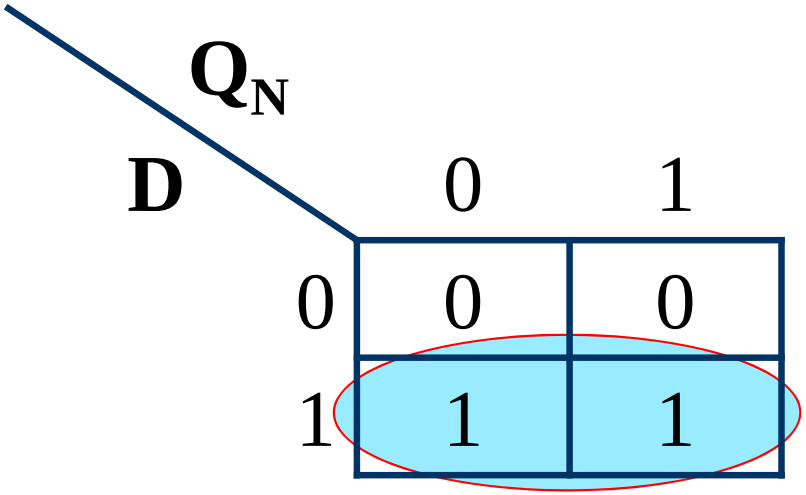


D FF (-ve Edge-triggered)

D FLIP-FLOP

- Truth table and Characteristic equation for D FF

D	Q_N	Q_{N+1}
0	0	0
0	1	0
1	0	1
1	1	1



Characteristic equation

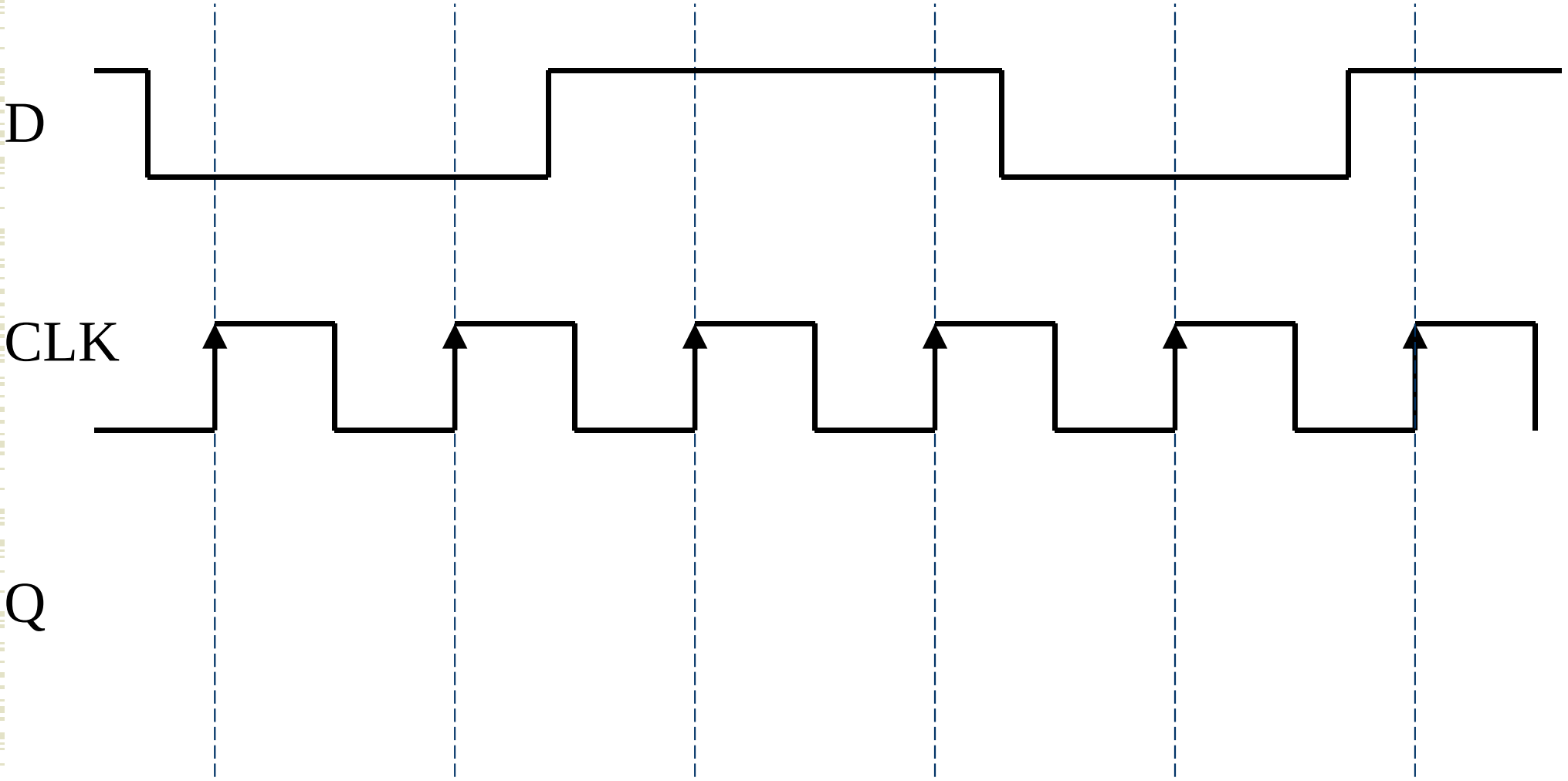
$$Q_{N+1} = D$$

D FLIP-FLOP

- D FF Transition Table!

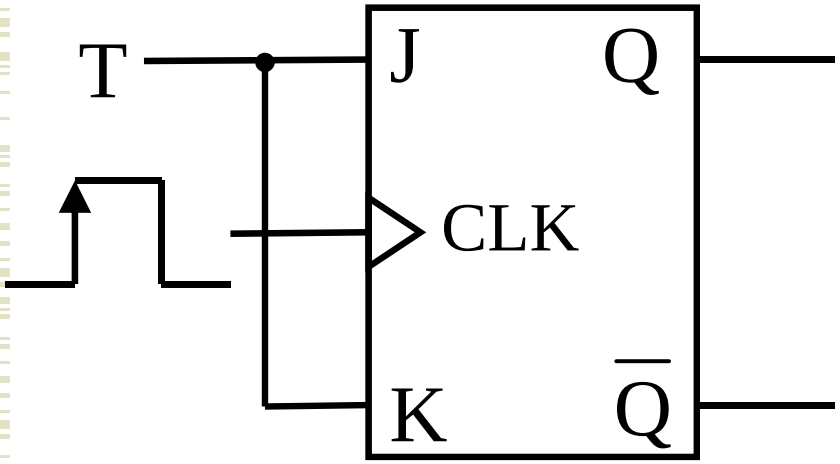
OUTPUT TRANSITION Q_N		Q_{N+1}	FF INPUT D
0	→	0	0
0	→	1	1
1	→	0	0
1	→	1	1

D FLIP-FLOP

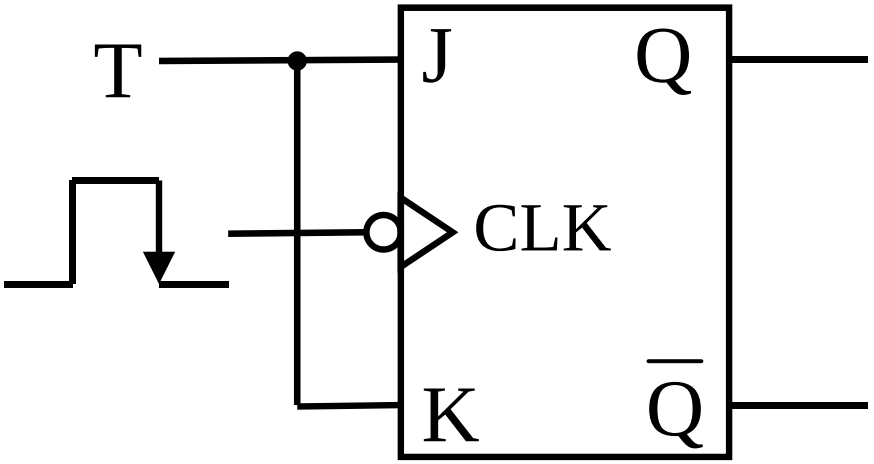


T FLIP-FLOP

- Flip-flop T (Toggle)!



T FF (+ve Edge-triggered)

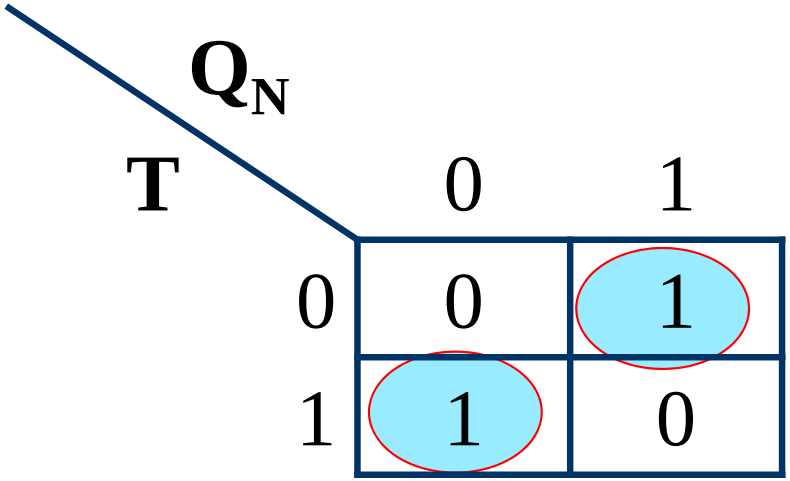


T FF (-ve Edge-triggered)

T FLIP-FLOP

- Truth table and Characteristic equation for T FF!

T	Q_N	Q_{N+1}	
0	0	0	
0	1	1	
1	0	1	$\overline{Q}$
1	1	0	



Characteristic equation

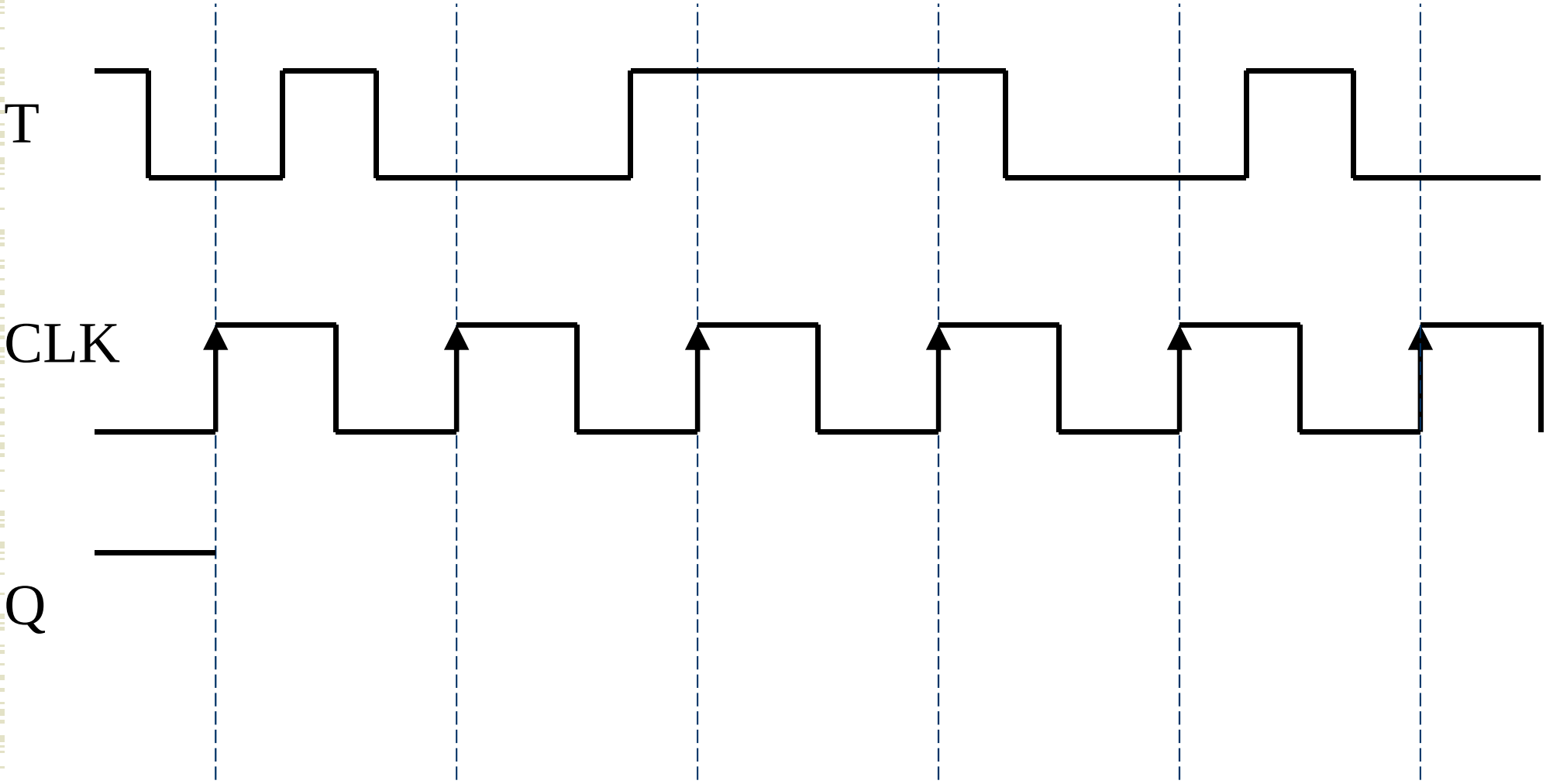
$$Q_{N+1} = T\overline{Q_N} + \overline{T}Q_N$$

T FLIP-FLOP

- T FF Transition Table!

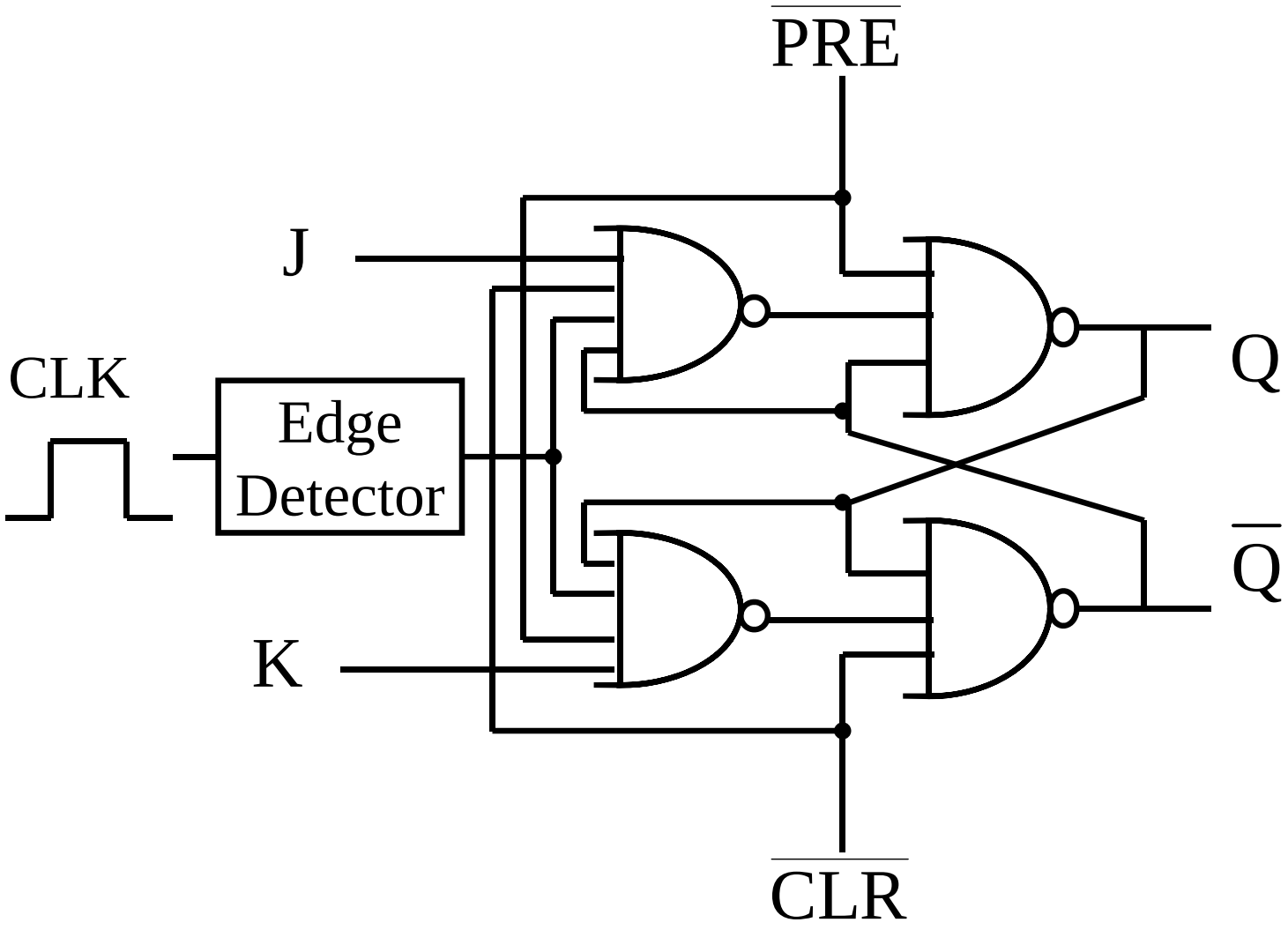
OUTPUT TRANSITION Q_N		Q_{N+1}	FF INPUT T
0	→	0	0
0	→	1	1
1	→	0	1
1	→	1	0

T FLIP-FLOP



J-K FLIP-FLOP with $\overline{PRE}$ and $\overline{CLR}$

- J-K Flip-flop with $\overline{PRE}$ and $\overline{CLR}$ inputs.



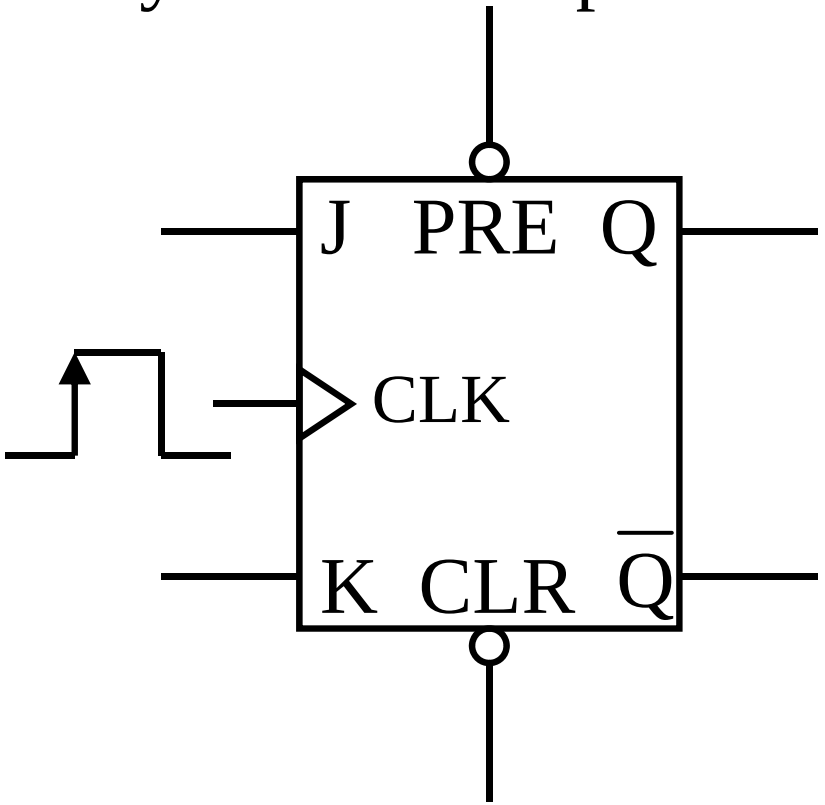
Asynchronous PRESET and CLEAR Inputs

- Inputs for SR, JK and D flip-flops are called *synchronous inputs* because this input will only be considered by the flip-flops during the triggering edge of the CLK.
- Most flip-flops also have ***asynchronous inputs.*** These inputs are used to change the output value **immediately** without considering the edge-triggered of the flip-flops.
- These inputs are labeled as PRESET (*PRE*) and CLEAR (*CLR*), or *direct set* (S_D) and *direct reset* (R_D).
- An active level of PRE will set the output (Q) of the flip-flop to logic ‘1’
- An active level of CLR will set the output (Q) of the flip-flop to logic ‘0’

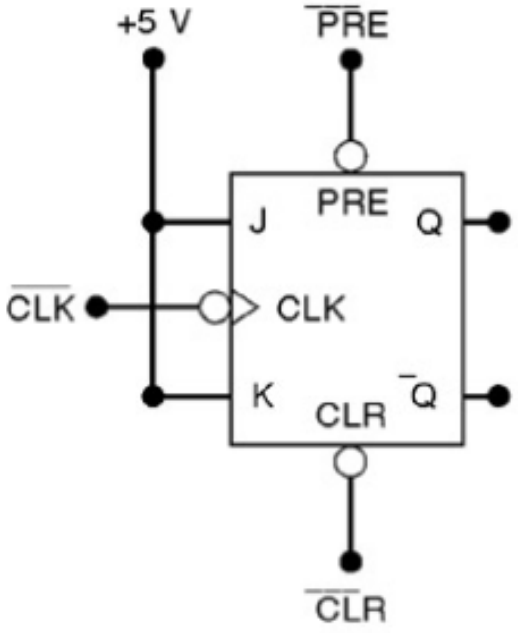
J-K FLIP-FLOP with $\overline{PRE}$ and $\overline{CLR}$

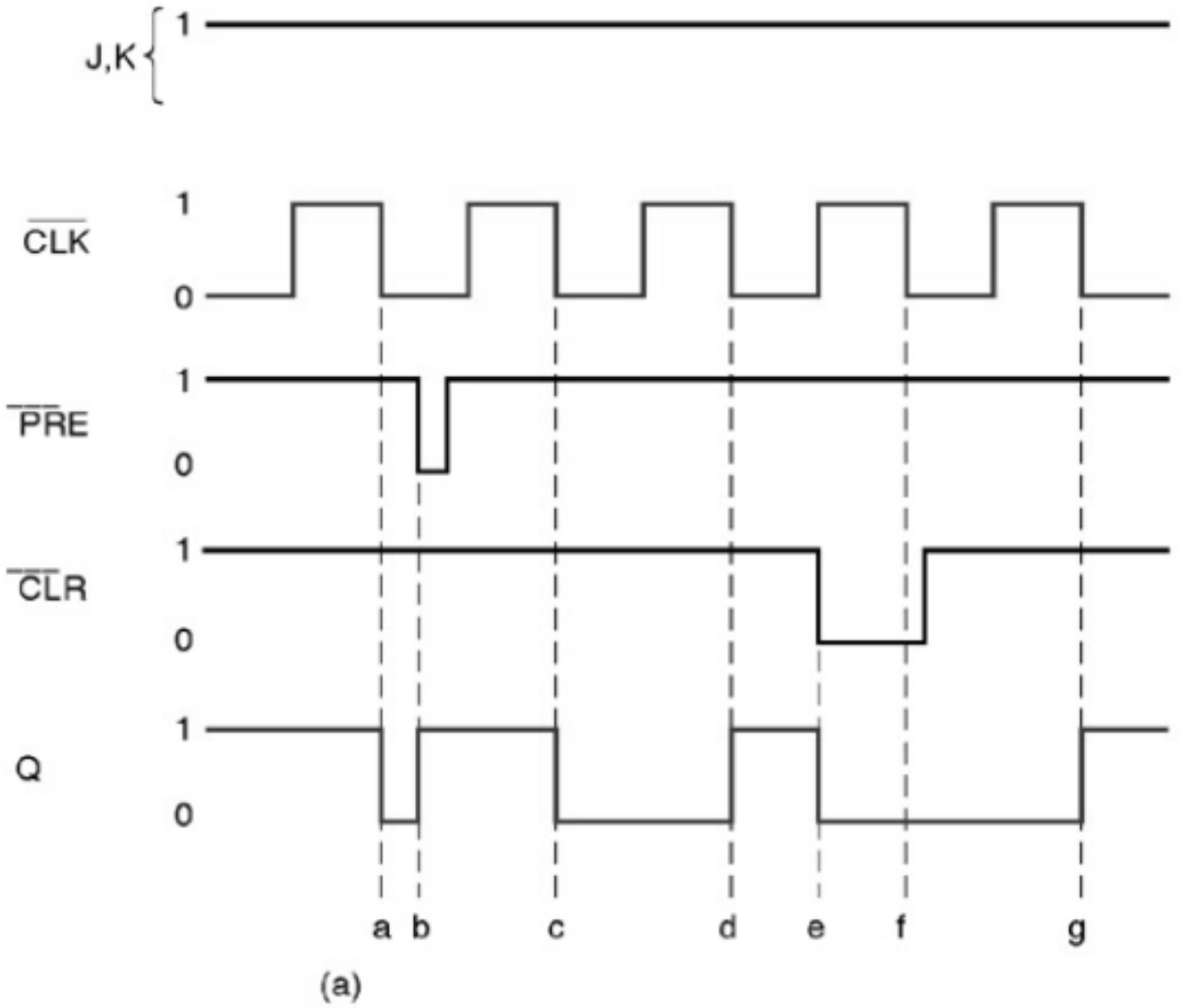
- Truth table for JK Flip-flop with PRESET (PRE) and CLEAR (CLR) inputs!
- Async input must be kept HIGH for synchronous operation.

$\overline{CLR}$	$\overline{PRE}$	FF Operation
0	0	Not allowed
0	1	Q = 0
1	0	Q = 1
1	1	Usual flip flop operation



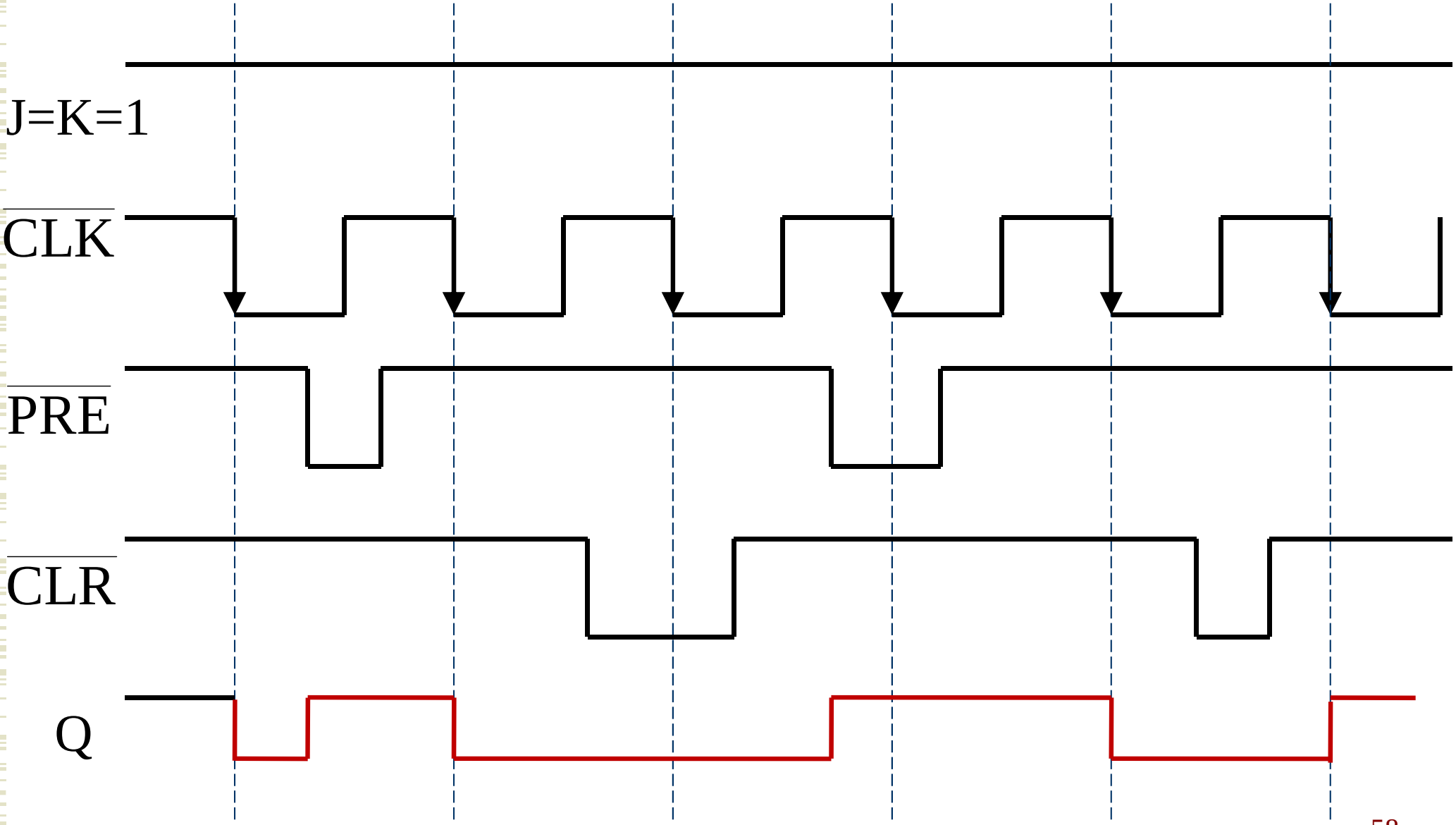
Exercise J-K FF with $\overline{PRE}$, $\overline{CLR}$, $\overline{CLK}$



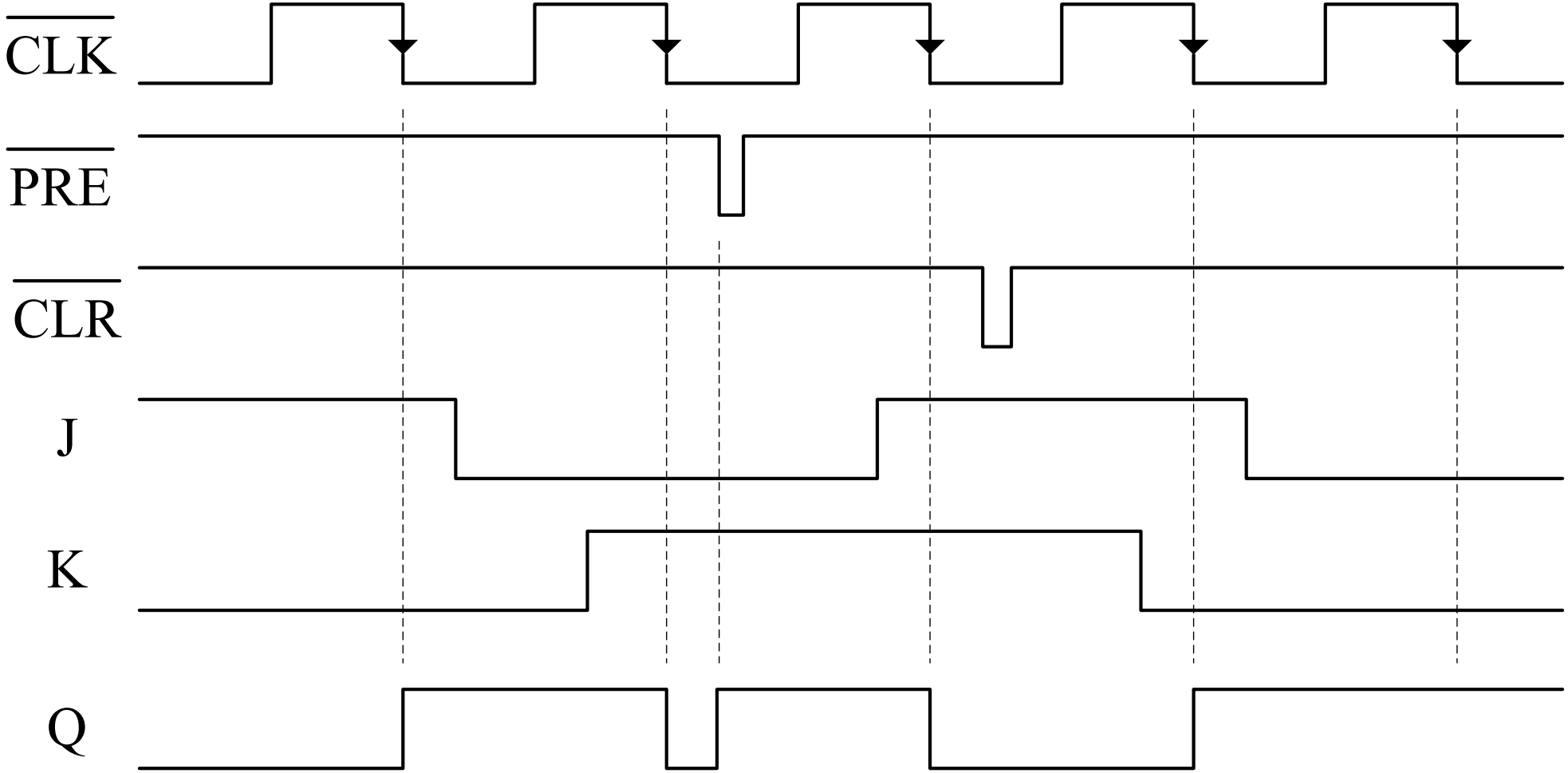


(a)

Exercise J-K FF with $\overline{PRE}$, $\overline{CLR}$, $\overline{CLK}$

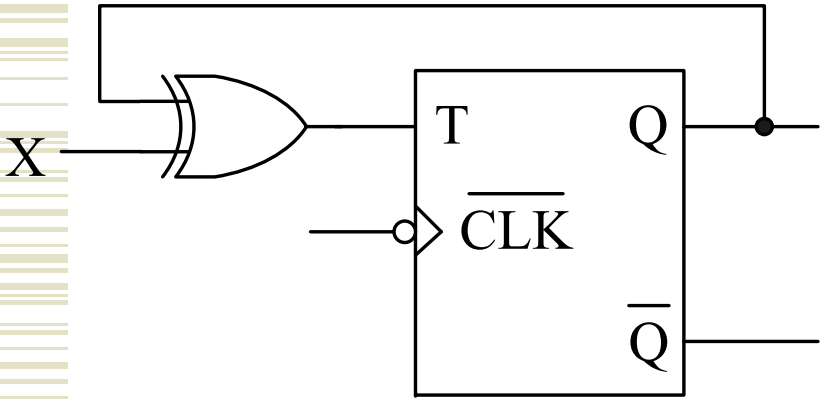


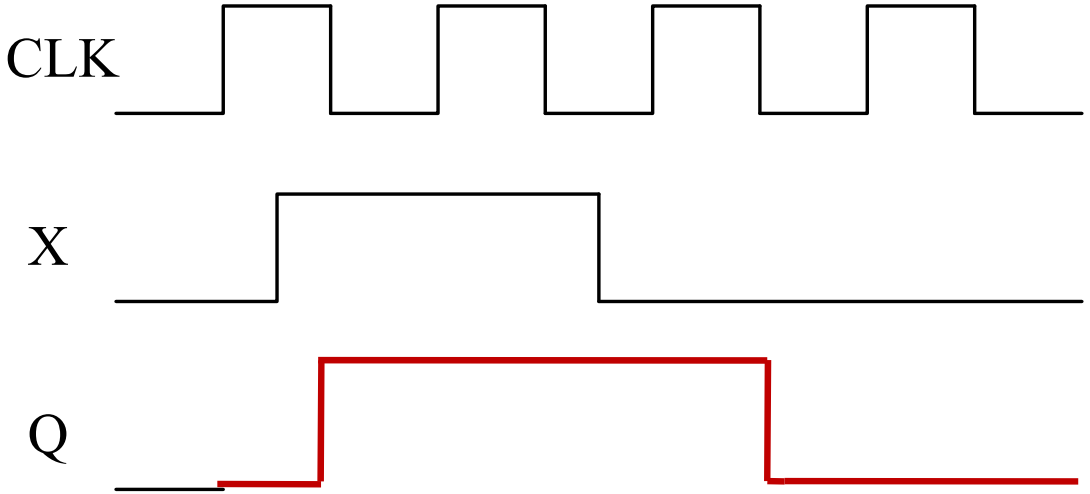
Exercise J-K FF with $\overline{PRE}$, $\overline{CLR}$, $\overline{CLK}$



Exercise

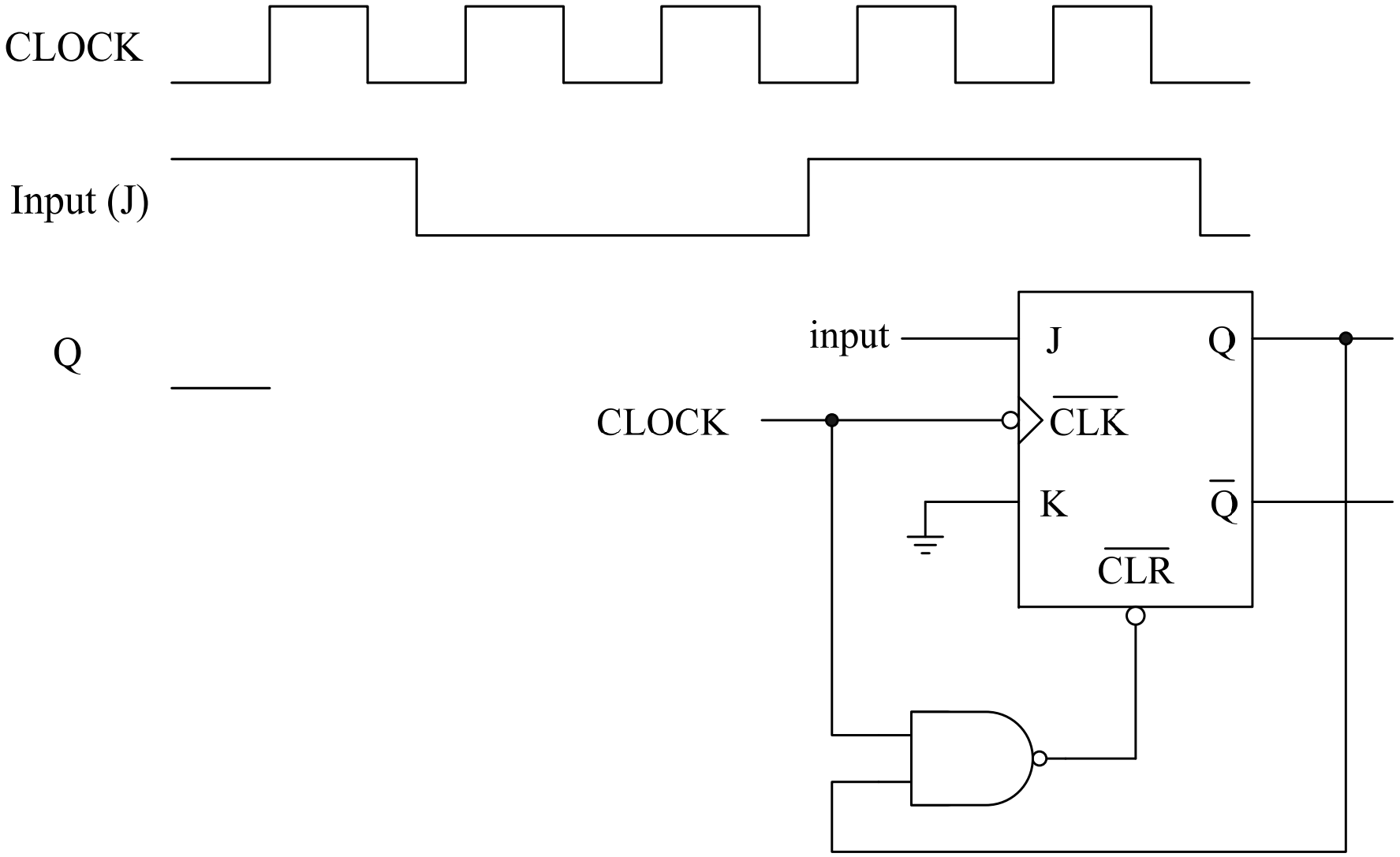
Sketch the output waveform for the circuit.





Exercise

Sketch the output waveform for the circuit. Q is initially 0.



Solution Exercise

Sketch the output waveform for the circuit. Q is initially 0.