Chapter 11
Latches and Flip-Flops
SKEE1223 Digital Electronics

Mun'im/Arif/Izam

FKE, Universiti Teknologi Malaysia

December 8, 2015

Types of Logic Circuits

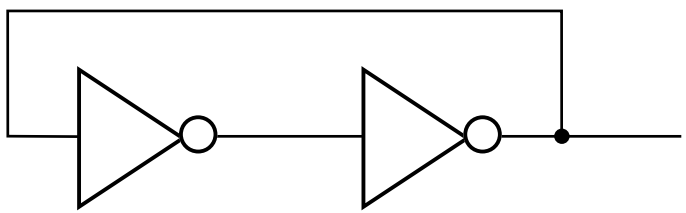
- **Combinational logic**:
  - Output depends solely on the present input.
  - Has no memory.
- **Sequential logic**:
  - Output depends not only on the present input and also on past history of inputs.
  - Has memory.
  - Synchronous sequential logic
    - Use a ‘clock’ signal to regulate operations.
    - Simpler to design
  - Asynchroous sequential logic
    - Does not use a clock.

Types of Memory Elements

- Ungated Latches
- Gated Latches
- Flip-flops

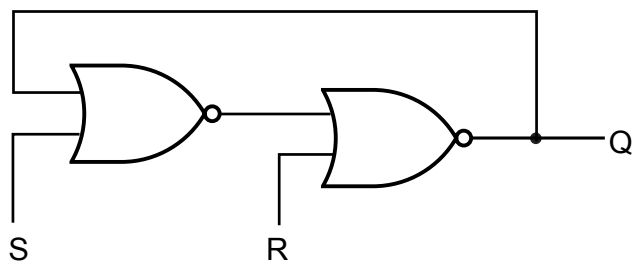
Bistable Circuits

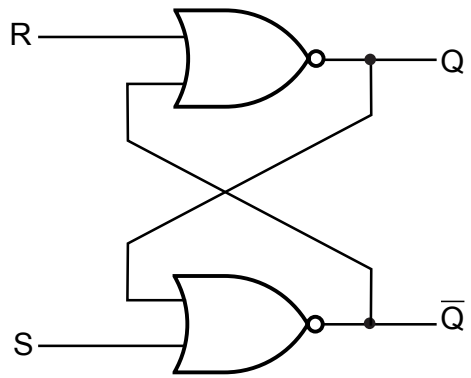
- Bistable circuit
  - Any circuit stable in 0 or 1
  - Has memory
  - Value does not change by itself
  - Simplest bistable circuit : cascaded inverters



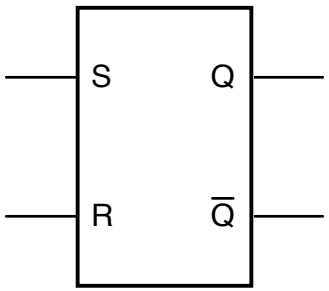
SR Latch

- SR = set/reset
- Replace inverters in prev page with NOR gates
- When S = R = 0, operates exactly as cascaded inverters





SR Latch

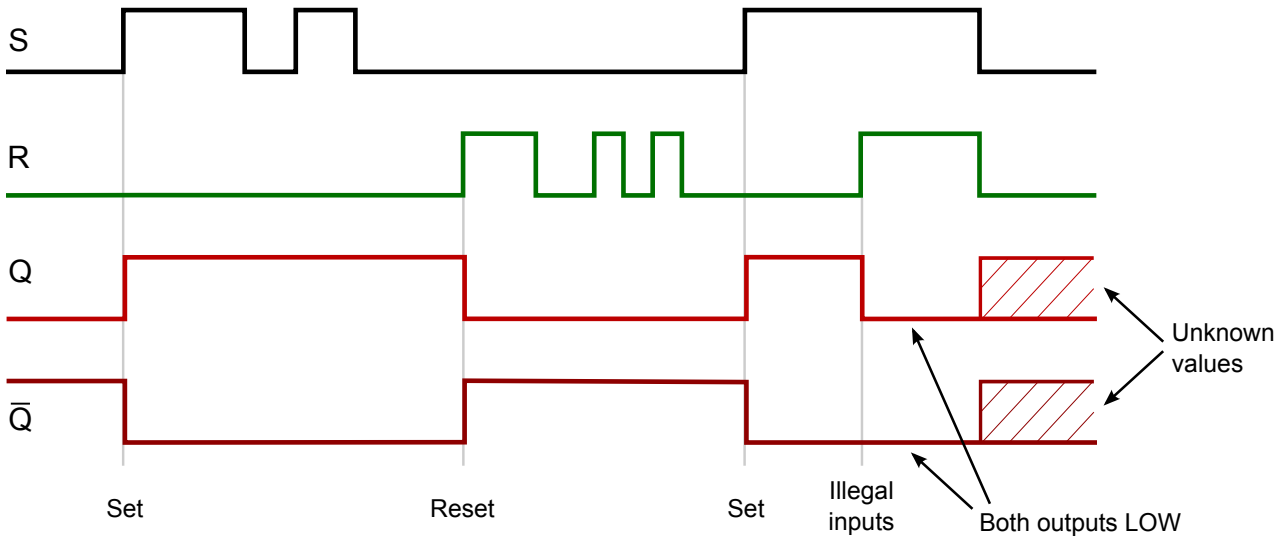


Logic symbol.

S	R	Q(next)	$\overline{Q(next)}$	Action
0	0	Q	$\overline{Q}$	No change
0	1	0	1	Reset
1	0	1	0	Set
1	1	0	0	Forbidden

Characteristic table.

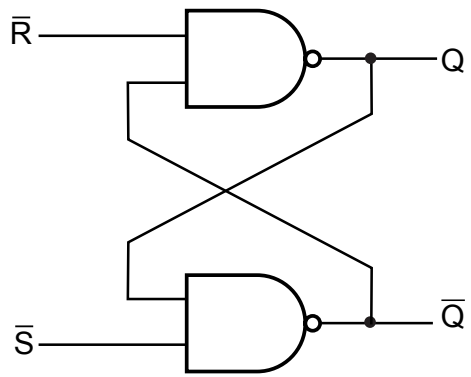
SR Latch



- Q and $\bar{Q}$ are always opposite
- S = R = 0 holds previous value
- S = 1, R = 0 sets Q to 1
- S = 0, R = 1 resets Q to 0
- S = R = 1, both Q and $\bar{Q}$ goes to 0, $\rightarrow$ forbidden state

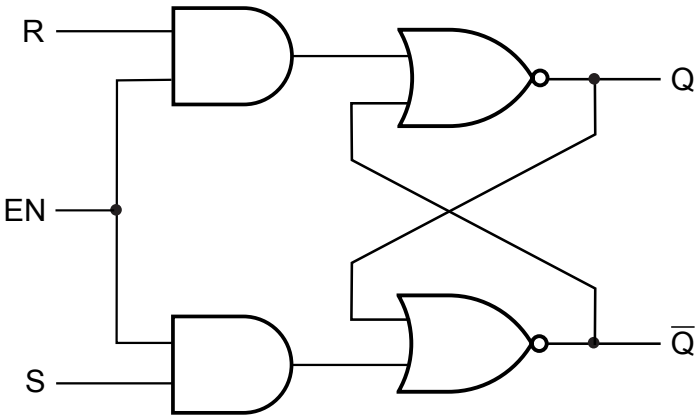
$\bar{S}\bar{R}$ Latch

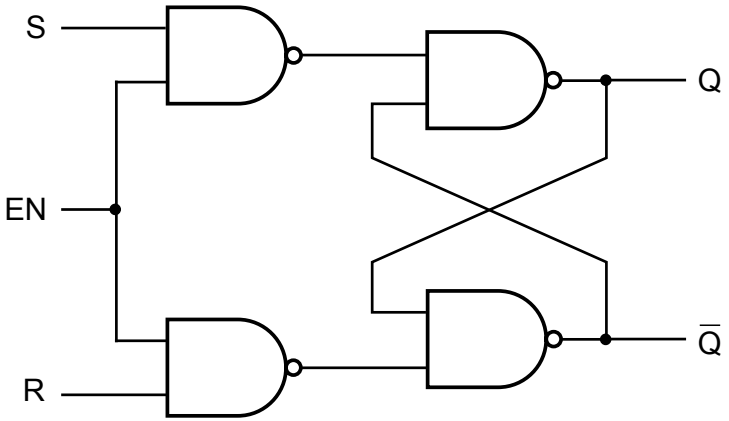
- Same idea as SR latch
- S and R are active low



$\bar{S}$	$\bar{R}$	Q(next)	$\overline{Q(next)}$	Action
0	0	1	1	Forbidden
0	1	0	1	Set
1	0	1	0	Reset
1	1	Q	$\overline{Q}$	No change

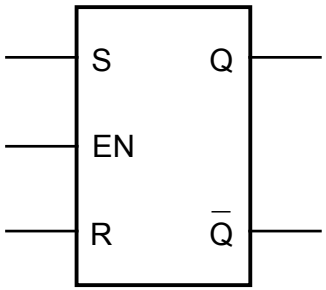
Gated SR Latch



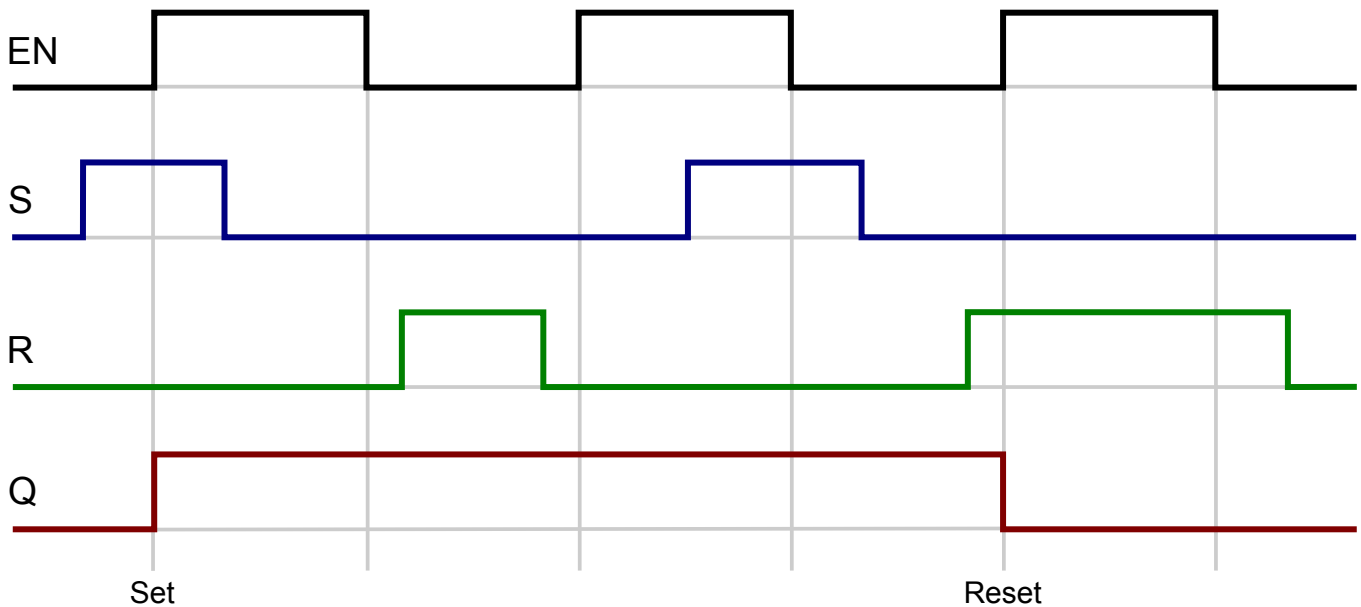


- EN = 1
  - enables latch
  - opens **gate** for SR inputs to cross-coupled gates
  - works like ungated SR latch
- EN = 0
  - disables latch
  - holds prev state
  - ignores SR inputs

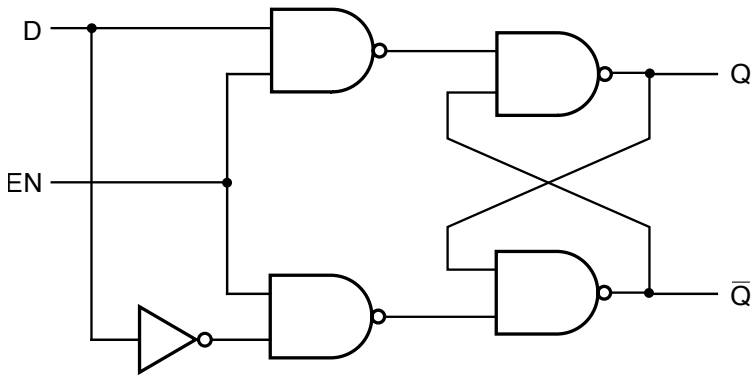
Gated SR Latch

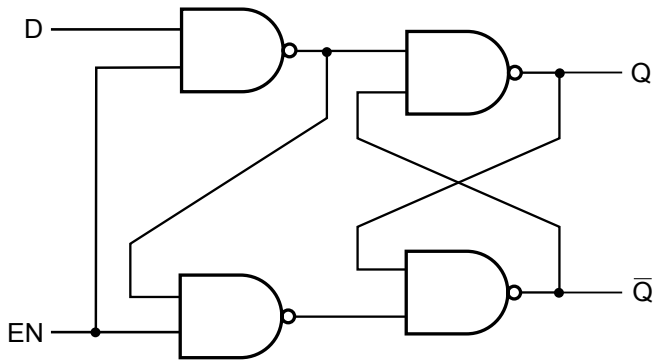


EN	S	R	Q(next)	Action
0	X	X	Q	No change
1	0	0	Q	No change
1	0	1	0	Reset
1	1	0	1	Set
1	1	1	X	Forbidden



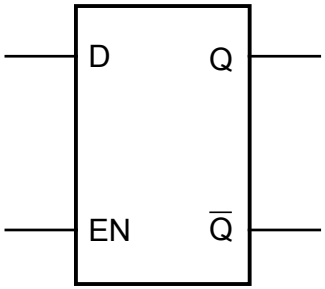
D Latch



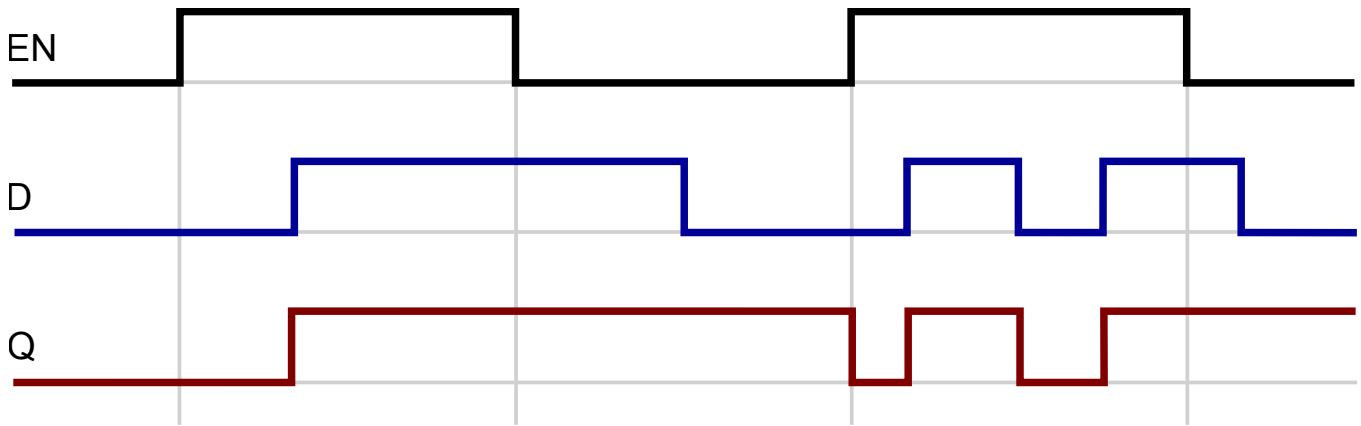


- Modify SR latch to avoid SR=11 condition
- EN = 1
  - latch become ‘transparent’
  - input D is passed to output Q after some delay
- EN = 0
  - disables latch
  - ignores D inputs
  - holds last D input when EN went 1 $\rightarrow$ 0

D Latch



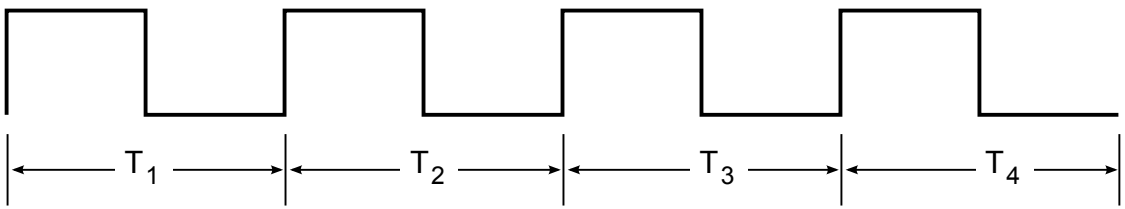
EN	D	Q(next)	Action
0	X	Q	Storage state
1	0	0	Transparent mode
1	1	1	Transparent mode



Clock Inputs

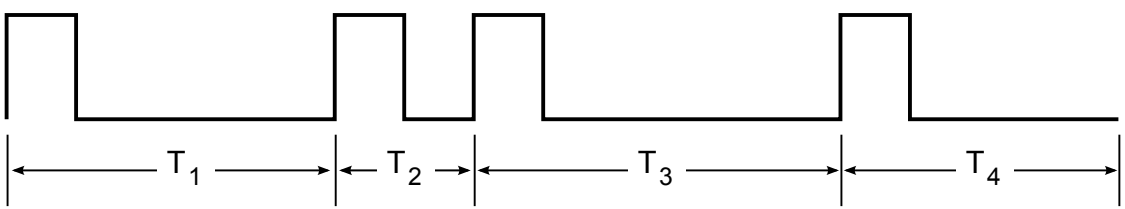
- Synchronous digital systems use a clock
- Clock signal is distributed to all system components
- All outputs change simultaneously when a clock pulse arrives

Clock.



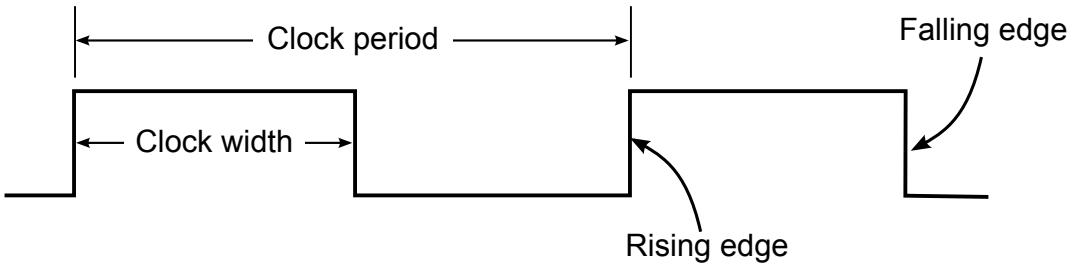
$T_1 = T_2 = T_3 = T_4$

Not a clock.



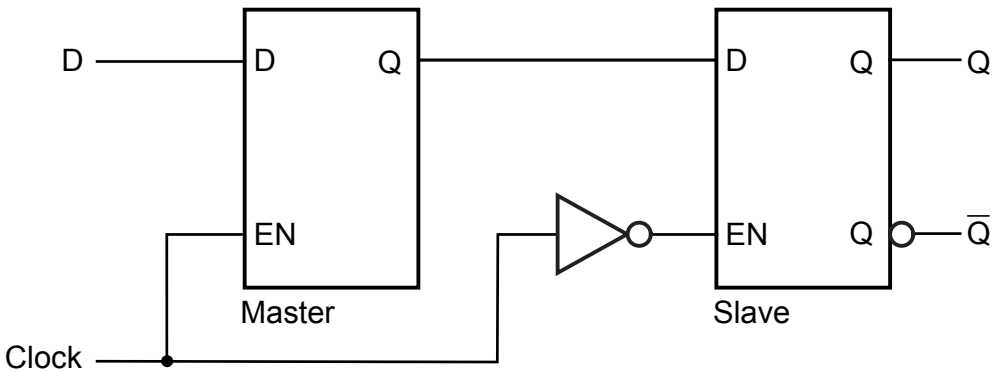
$T_1 \neq T_2 \neq T_3 \neq T_4$

Parts of a Clock Signal



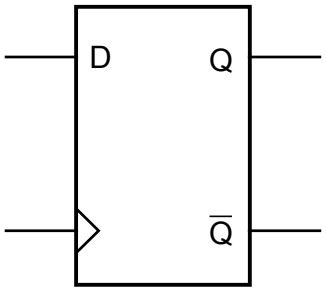
Device	How it uses the clock
Latch	Does not use the clock
Gated latch	Enabled when clock is high
Flip-flop	Triggered by rising or falling clock edge

Master Slave Flip-Flop

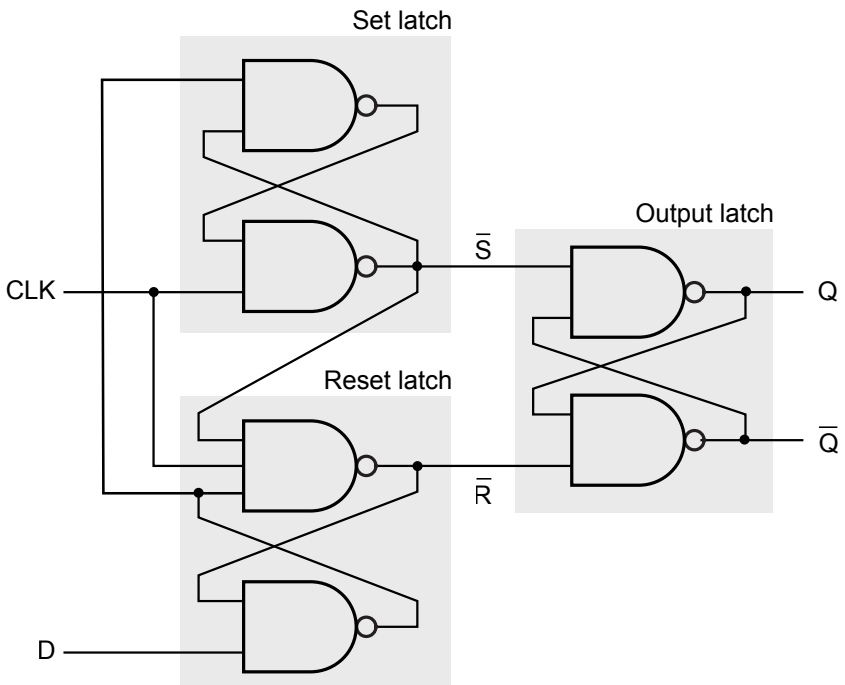


- Clock high
  - Master enabled, slave disabled
  - Input D is transparently passed on to Q_{master}
- Clock low
  - Master disabled, slave enabled
  - Q_{master} is locked, and read by slave
  - Q_{slave} is updated

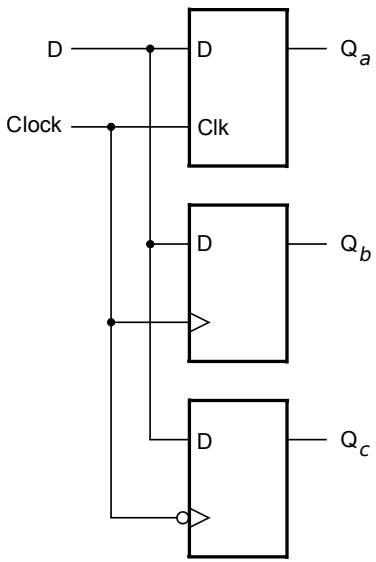
D Flip-Flop

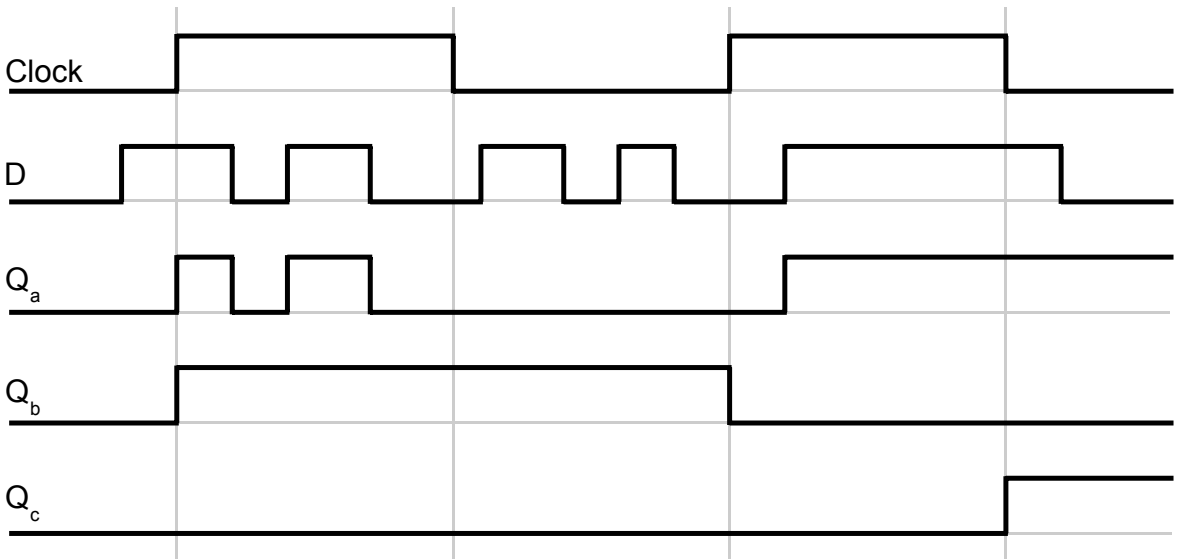


Clk	D	Q(next)	Action
0	X	Q	No change
1	X	Q	No change
↑	0	0	Reset
↑	1	1	Set

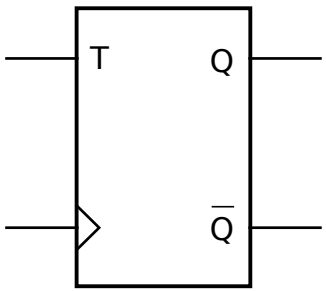


D Flip-Flop

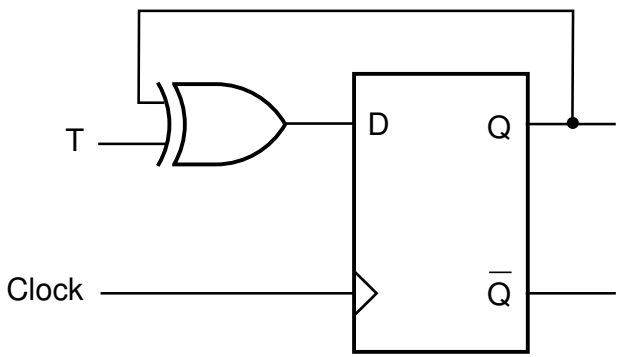




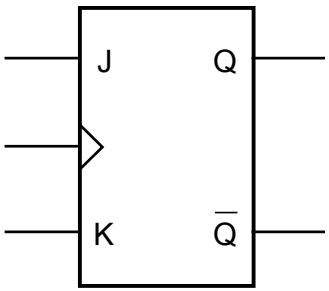
T Flip-Flop



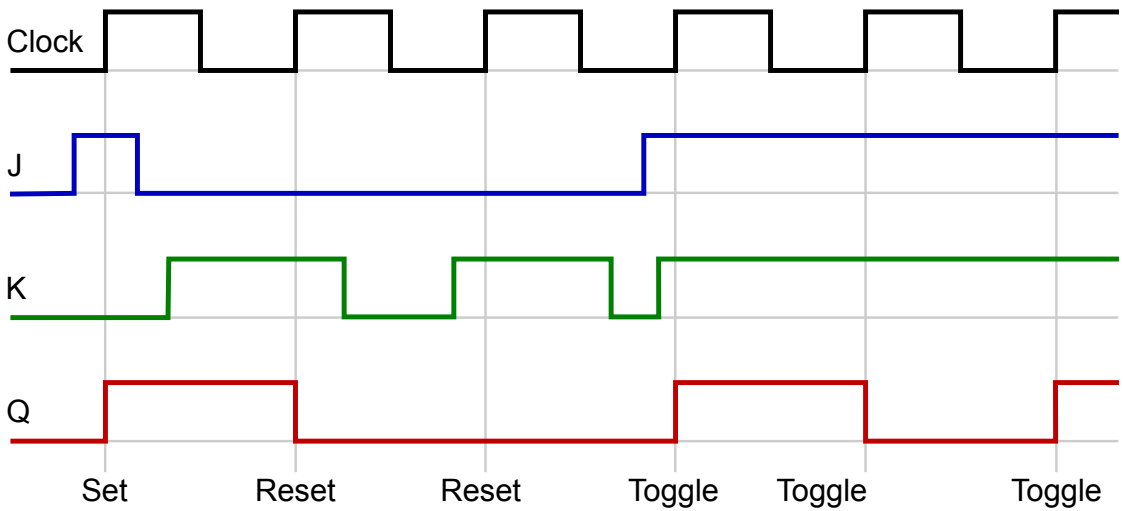
Clk	T	Q(next)	Action
0	X	Q	No change
1	X	Q	No change
↑	0	Q	Hold
↑	1	$\overline{Q}$	Toggle



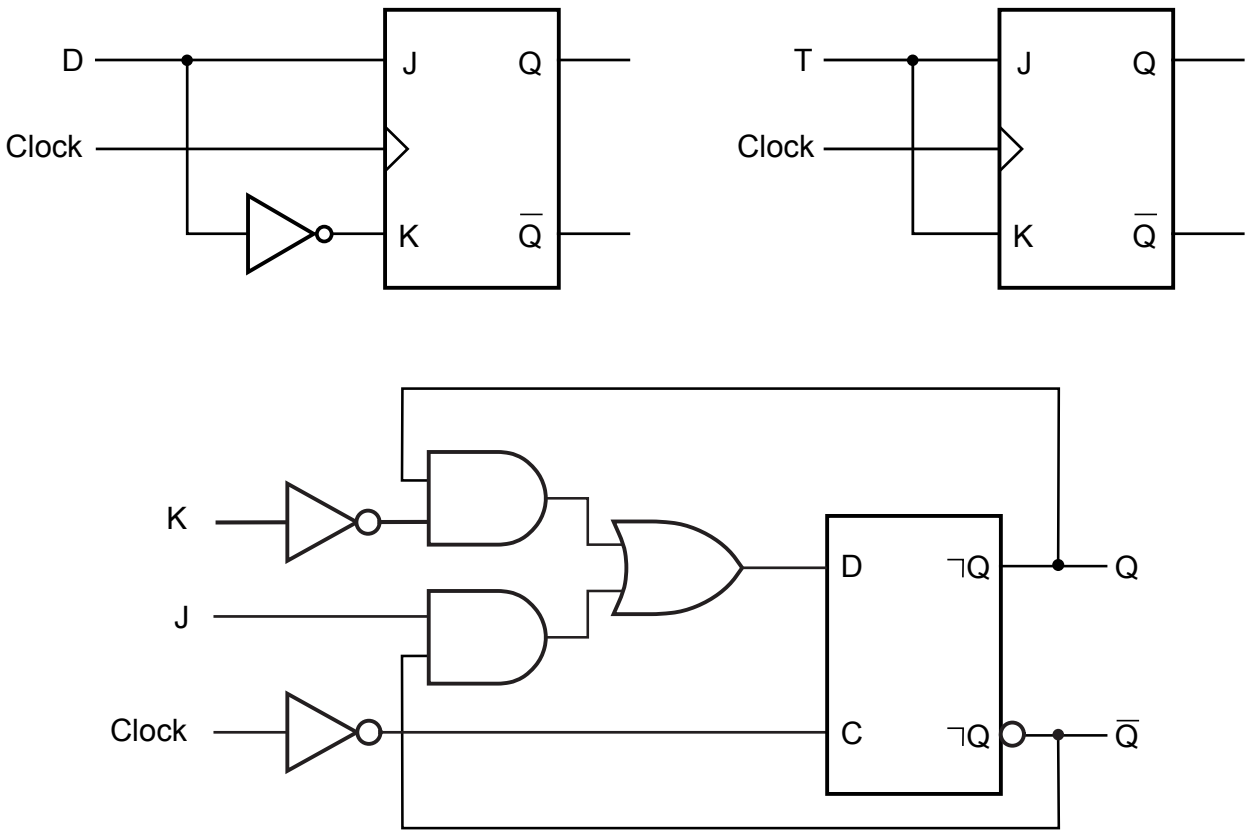
JK Flip-Flop



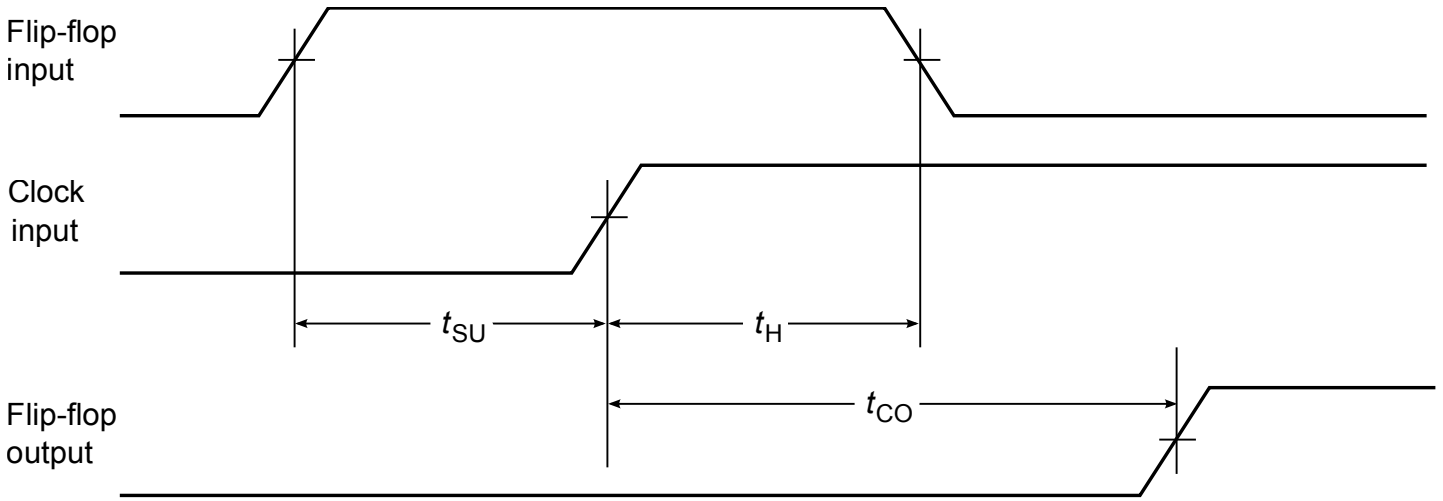
C	S	R	Q(next)	Action
0	X	X	Q	No change
1	X	X	Q	No change
↑	0	0	Q	No change
↑	0	1	0	Reset
↑	1	0	1	Set
↑	1	1	X	Toggle



JK Flip-Flop

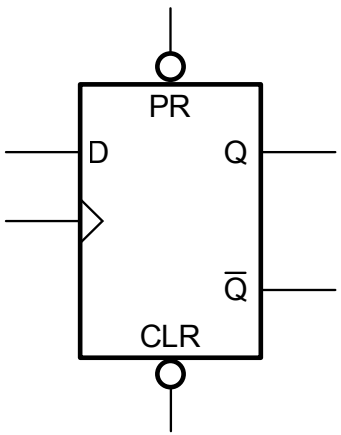

D
Clock
J
Q
K
$\overline{Q}$
T
Clock
J
Q
K
$\overline{Q}$
K
J
Clock
D
$\neg$Q
C
$\neg$Q
Q
$\overline{Q}$

Flip-Flop Timing



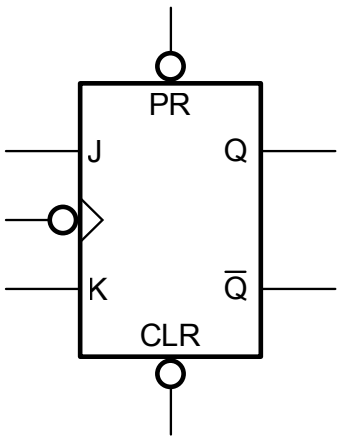
- t_s – Setup time
  - The time a control input must be maintained before the clock transition.
- t_h – Hold time
  - The time a control input must be maintained after the clock transition.
- t_{PCQ} – Propagation delay from clock to output change
  - The time a flip-flop changes after a clock edge is given

74x74 D PET FF



Preset	Clear	D	Clock	Q(next)
1	1	0	↑	0
1	1	1	↑	1
1	1	x	0	Q
1	1	x	1	Q
0	1	x	x	1
1	0	x	x	0
0	0	x	x	NA

74x76 Master Slave JK FF



Preset	Clear	J	K	Clock	Q(next)
1	1	0	0	↑	Q
1	1	0	1	↑	0
1	1	1	0	↑	1
1	1	1	1	↑	$\overline{Q}$
1	1	x	x	0,1	Q
0	1	x	x	x	1
1	0	x	x	x	0
0	0	x	x	x	NA

https://www.openlearning.com/courses/SKEE1223x