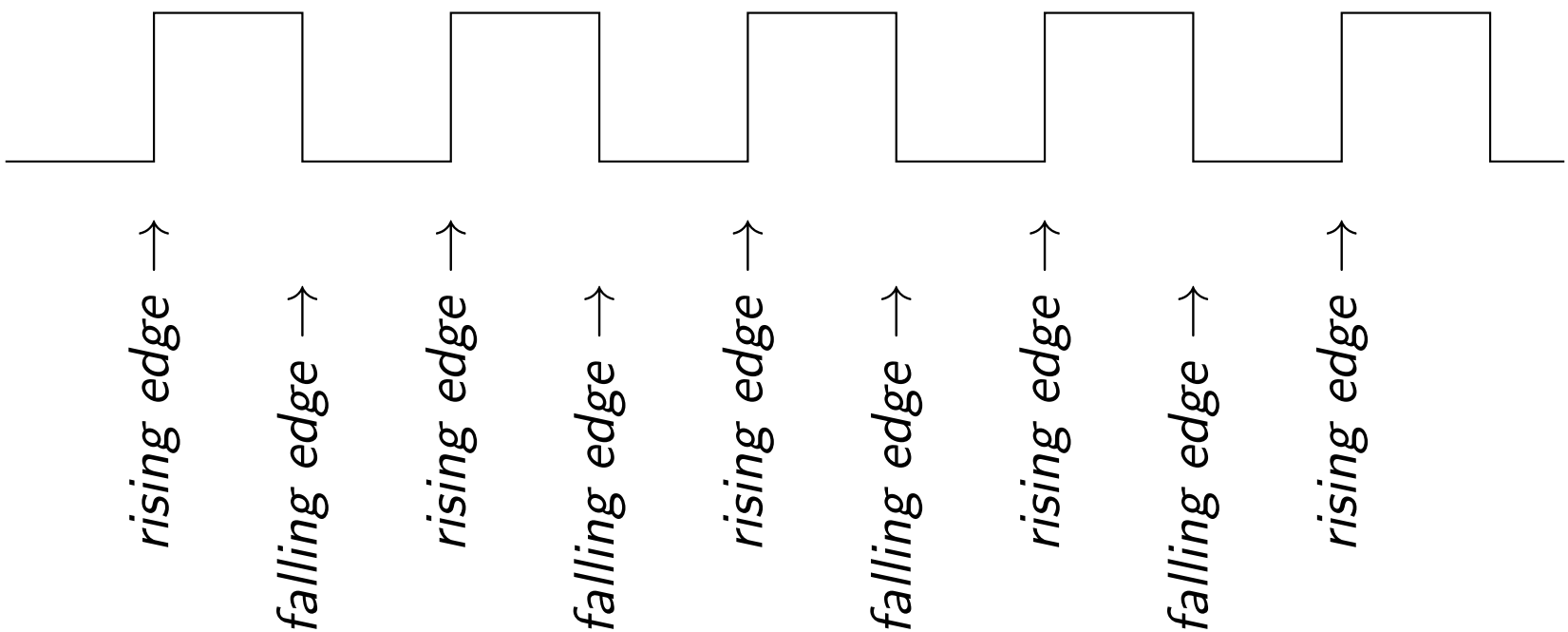
Flip flops

- One type of *storage element* — exhibits the concept of memory.
- Flip flops differ from latches in that flip flops are *edge triggered* devices.
- The output of a flip flop *changes* depending on the values of the flip flop inputs *and* when a *clock* input changes from $0 \rightarrow 1$ (or $1 \rightarrow 0$).
- If the output changes when the clock transitions from $0 \rightarrow 1$, the flip flop is *positive edge triggered*. If the output changes when the clock transitions from $1 \rightarrow 0$, then the flip flop is *negative edge triggered*.

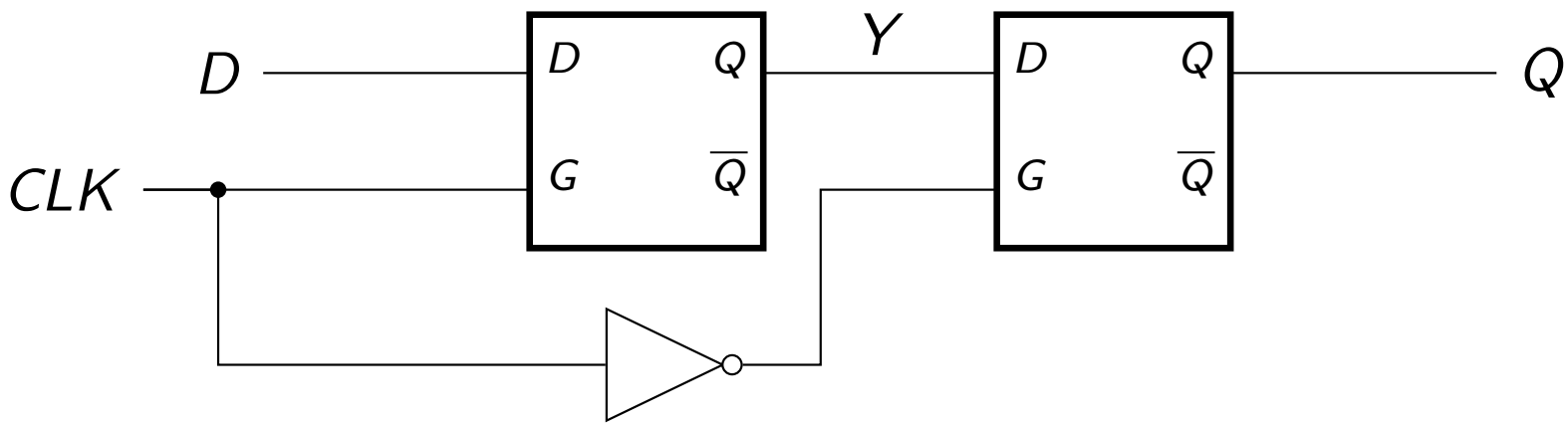
Flip flops

- What's a clock? A *clock* is a periodic signal. By using flip flops, we can restrict when flip flop outputs change to discrete instances in time — either on the *rising edge* ($0 \rightarrow 1$) of the clock, the *falling edge* ($1 \rightarrow 0$) of the clock, or both.
- A clock signal...



Master-slave flip flop

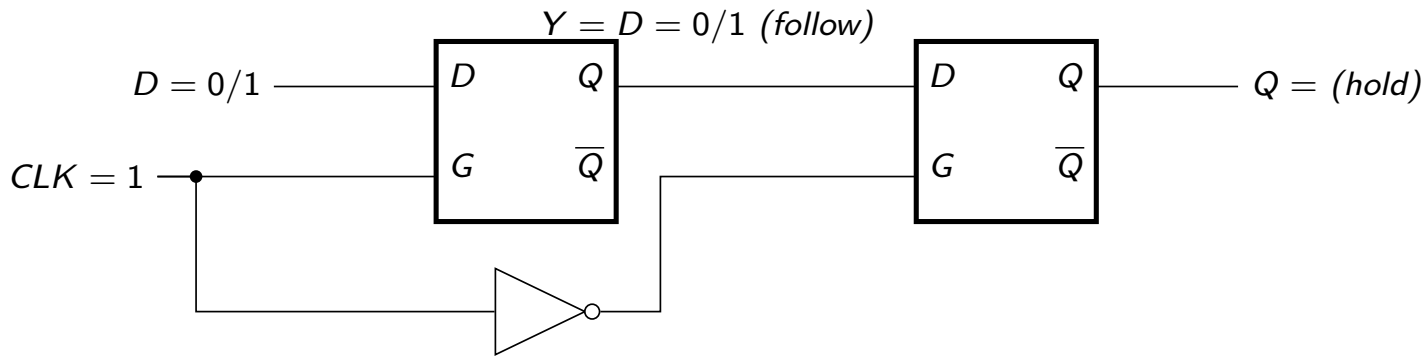
- Can build a flip flop from two latches to get an idea of how a flip flop works using a previously understood circuit (the latch).
- The master-slave flip flop:



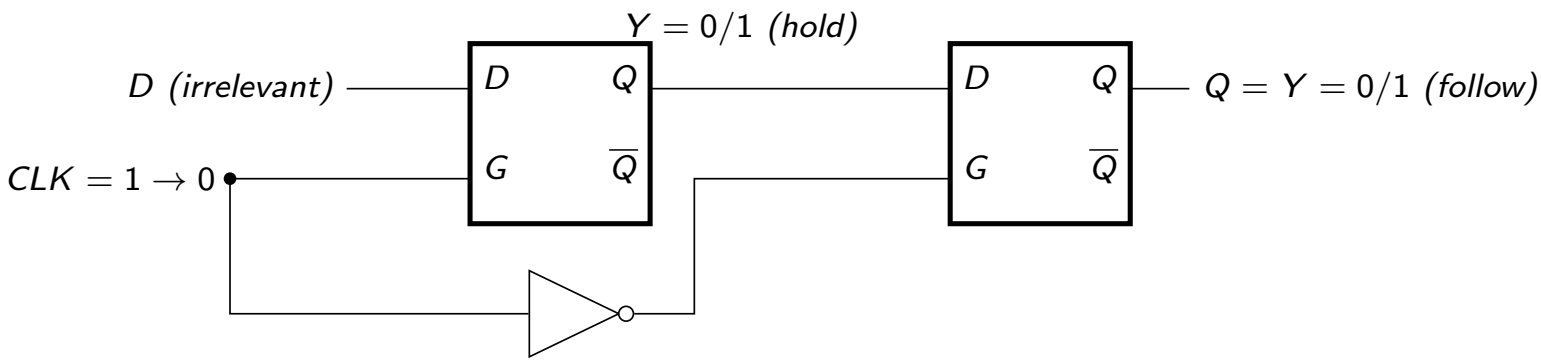
- First latch is called the *master* and the second latch is called the *slave*.

Master-slave flip flop

- Behaviour: Assume $CLK = 1$. The output of the master latch will *follow* the input D so $Y = D$. The slave latch, however, is in *hold* so the output Q will not change and it will be at its previous value.



- Behaviour: Assume CLK changes from $1 \rightarrow 0$. The output of the master latch will *hold* — it will be held at *the value of D just prior to the clock changing*. The slave latch, however, will have its output Q equal to its input $Q = Y$.

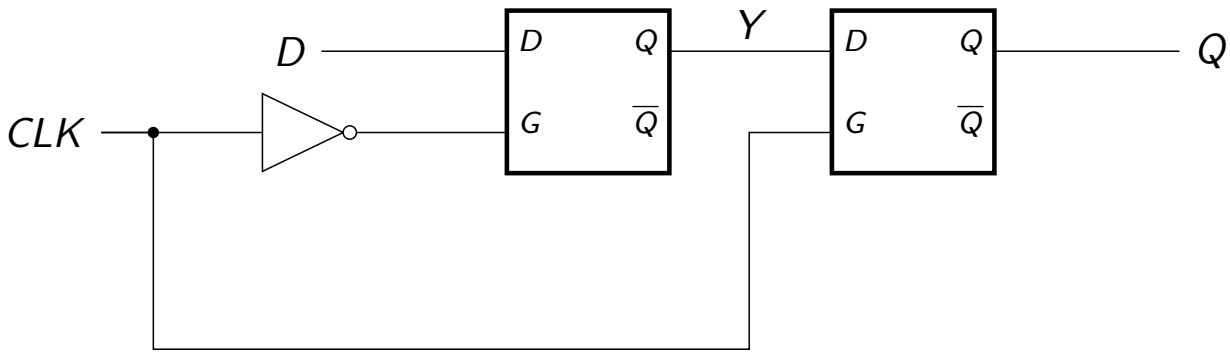


Master-slave flip flop

- The net result — it appears that the value of the D input just prior to the CLK changing from $1 \rightarrow 0$ gets transferred to the output Q.
- This exhibits the behaviour of the definition of a flip flop — the output changes according to the inputs based on the transition of a clock signal.
- This device is an example of a negative edge triggered flip flop.

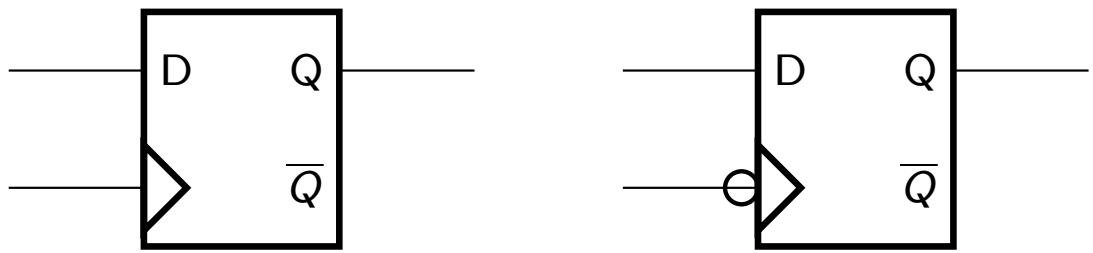
Master-slave flip flop

- We can make a positive edge triggered master-slave flop-flop by moving the position of the inverter:



DFF

- The master-slave flip flop is one way to make the *D* type flip flop, or *DFF*.
- Symbols for both a *positive edge triggered* DFF and a *negative edge triggered* DFF.



DFF

- Typical to describe a flip flop using a table — the table should show what the flip flop output will become given the current flip flop inputs after the clock makes its active edge transition. This is the *characteristic table*.

D	$Q(t)$	$Q(t+1)$
0	0	0
0	1	0
1	0	1
1	1	1

or

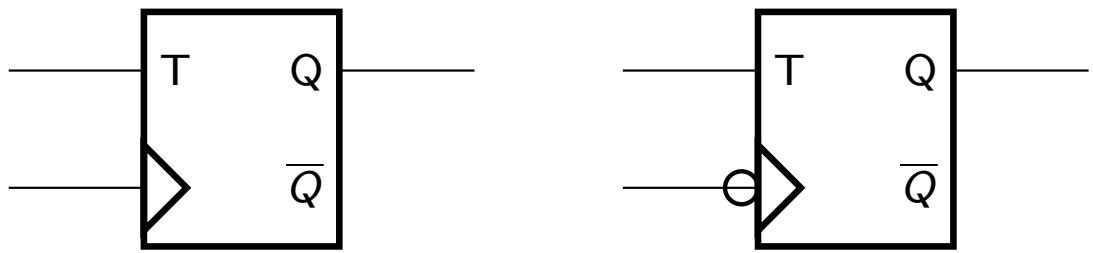
D	$Q(t+1)$
0	0
1	1

- Can also use *characterstic equation*:

$$Q(t+1) = D$$

TFF

- Another type of flip flop that behaves differently compared to the *DFF*; known as a *TFF* or *toggle flip flop*.
- Symbols for both a *positive edge triggered* TFF and a *negative edge triggered* TFF.



TFF

- The *characteristic table* for the *TFF*.

T	$Q(t)$	$Q(t+1)$
0	0	0
0	1	1
1	0	1
1	1	0

or

T	$Q(t+1)$
0	$Q(t)$
1	$\overline{Q(t)}$

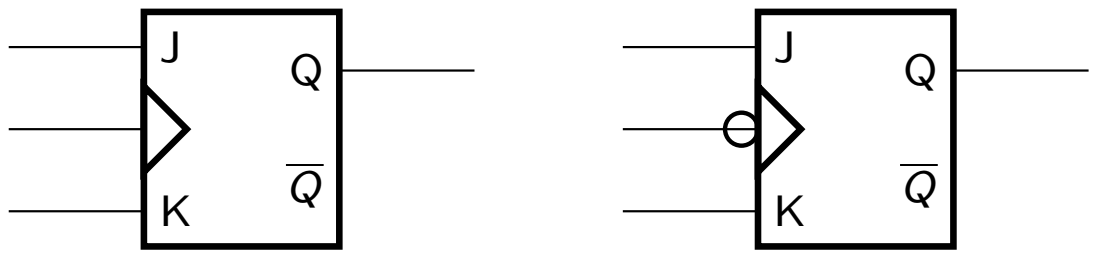
- Can also use *characterstic equation*:

$$Q(t+1) = Q(t)\overline{T} + \overline{Q(t)}T = Q(t) \oplus T$$

- The *TFF* flips or *toggles* the output if input $T = 1$ when the active clock edge arrives, otherwise the output does not change.

JKFF

- Another type of flip flop that behaves differently compared to the *DFF* and the *TFF* known as a *JKFF*.
- Symbols for both a *positive edge triggered JKFF* and a *negative edge triggered JKFF*.



JKFF

- The *characteristic table* for the *JKFF*.

J	K	$Q(t)$	$Q(t+1)$	
0	0	0	0	(hold)
0	1	0	0	(reset)
1	0	0	1	(set)
1	1	0	1	(toggle)
0	0	1	1	(hold)
0	1	1	0	(reset)
1	0	1	1	(set)
1	1	1	0	(toggle)

or

J	K	$Q(t+1)$	
0	0	$Q(t)$	(hold)
0	1	0	(reset)
1	0	1	(set)
1	1	$\overline{Q(t)}$	(toggle)

- Can also use *characterstic equation*:

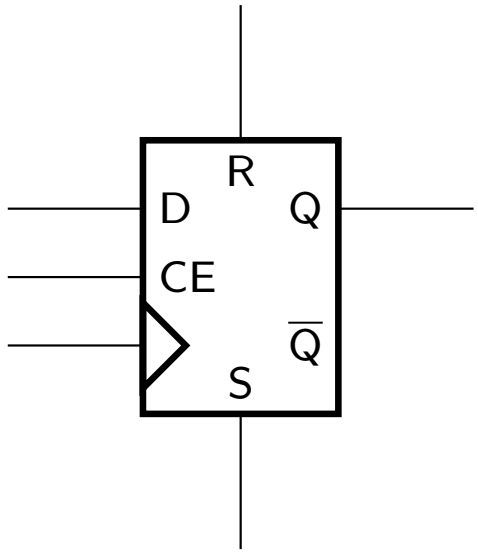
$$Q(t+1) = J\,\overline{Q(t)} + \overline{K}\,Q(t)$$

Extra control signals — sets, resets and enables

- We might see additional input pins on flip flops. Such additional signals might include *set*, *reset* and *enable* inputs.
  - A *set* signal is a signal which forces $Q = 1$ regardless of the flip flop input values and the behaviour of the flip flop;
  - A *reset* signal is a signal which forces $Q = 0$ regardless of the flip flop input values and the behaviour of the flip flop;
- Signals like sets and resets can either be *synchronous* with the clock (change happens at the active clock edge) or *asynchronous* with the clock (change happens immediately).
- An *enable* is a signal that prevents the clock signal from causing a change in Q (it effectively "blocks" the clock from the flip flop).

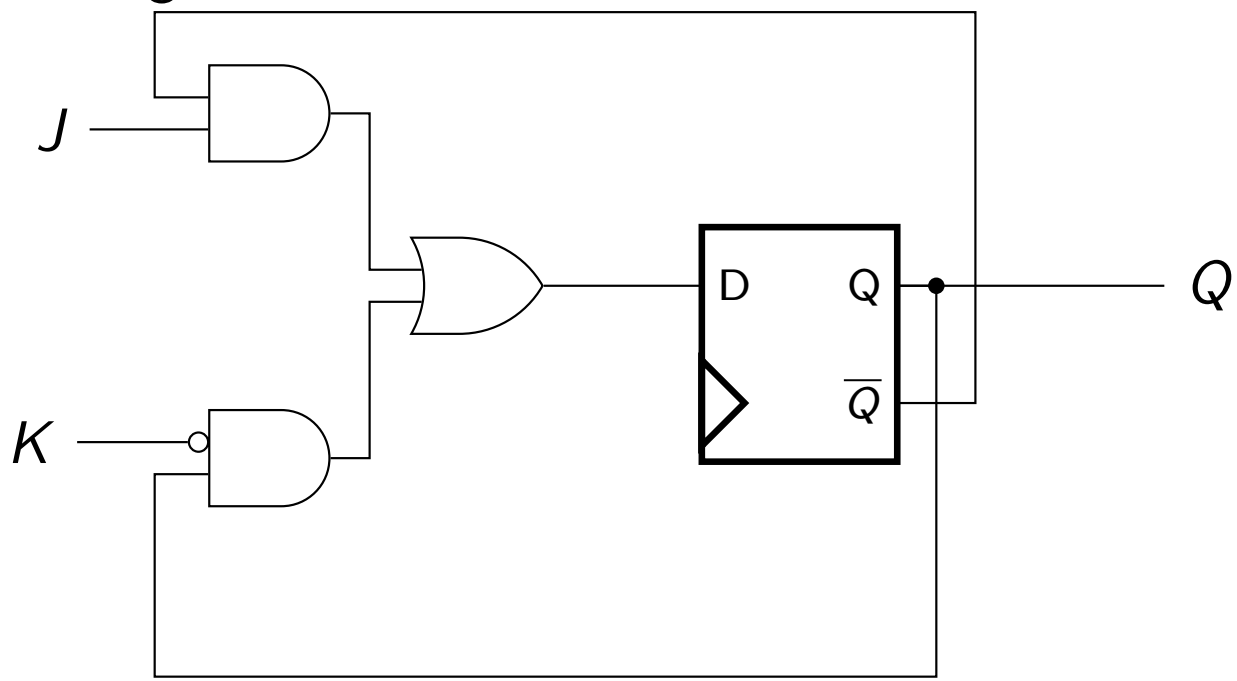
Extra control signals — sets, resets and enables

- Example of a *DFF* with additional input/control pins — set, reset and enable.



Constructing flip flops from other types of flip flops

- By considering the characteristic equations for each flip flop type, we can always build one type of flip flop from another type of flip flop.
- Example... Construct a *JKFF* using a *DFF* plus some additional logic...



- For the *DFF*, $Q(t+1) = D$ (characteristic equation), but that $D = J\,\overline{Q(t)} + \overline{K}\,Q(t)$. Therefore, the *DFF* plus logic acts exactly like a *JKFF*.

Timing parameters for flip flops

- There are some *restrictions* with respect to when signals can change in order to ensure that a flip flop behaves *correctly*. Observing these parameters is necessary to ensure that circuits involving flip flops work properly.
- Setup time (T_{SU}):
  - The *setup time* of a flip flop is defined as the amount of time that the data inputs need to be held stable (not changing) prior to the arrival of the active clock edge.
- Hold time (T_H):
  - The *hold time* of a flip flop is defined as the amount of time that the data inputs need to be held stable (not changing) after the arrival of the active clock edge.
- Both *setup* and *hold* times need to be obeyed in order to guarantee that the output of the flip flop changes to the correct value when the clock edge arrives based on the values of the data inputs.
- Clock to output time (T_{CO}):
  - The *clock to output time* of a flip flop is defined as the amount of time it takes for the output to change and become stable (reliable) after the arrival of the active clock edge.
- The output of a flip flop cannot be trusted until after an amount of time equal to the *clock to output* time.