Lecture 14, March 5, 2007

Flip-flops

In the last lecture, we talked about latches. Latches are a special kind of flip-flop. The difference between a latch and a flip-flop is that the output of a latch depends on the input level while the output of a flip-flop changes as a result of a change in the input level. We say that a flip-flop is edge-triggered while a latch is level triggered.

In sequential circuits the output of a flip-flop (or latch) is fed back to the combinational circuit that is connected to the input of that flip-flop and other flip-flops.

Assume that we have a D-latch in a sequential circuit. Assume that the control input is at level 1. Now, any change in the D input level changes the output of the latch. But this change,

after some delay, appears at the input of the latch. If at this time, the control input (the clock) is still active, the output will change again. This results in an unpredictable situation.

So, we need flip-flops (and not latches) for the implementation of the sequential circuits.

You may ask: "Then why did we waste our time learning about the latches?"

The answer is: we need the latches to build flip-flops.

There are two ways in which a flip-flop may respond to a change in its control input level. Some flip-flops respond to rising edge:

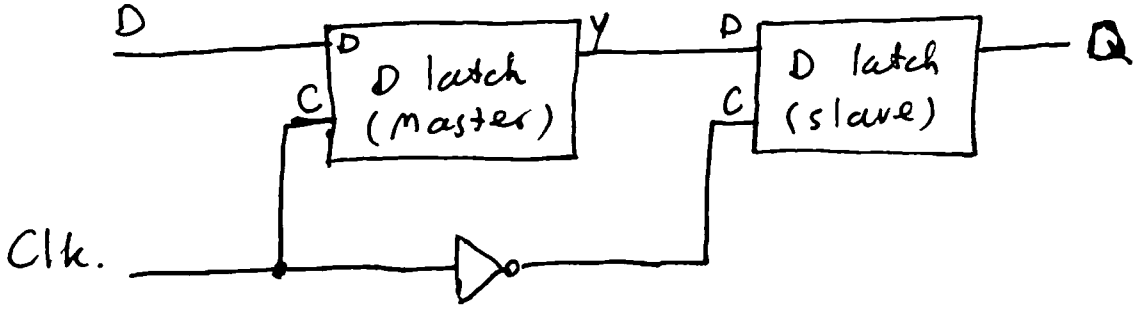
while others respond to the falling edge:

There are two ways to implement edge-triggered flip-flops:

- Isolate the input from the output and allow the output change only after control (clock) signal has been removed.
- Make a flip-flop that only changes when level of its clock goes from 0 to 1 or from 1 to 0 and remains unchanged rest of the time.

Implementing edge-triggered flip-flop

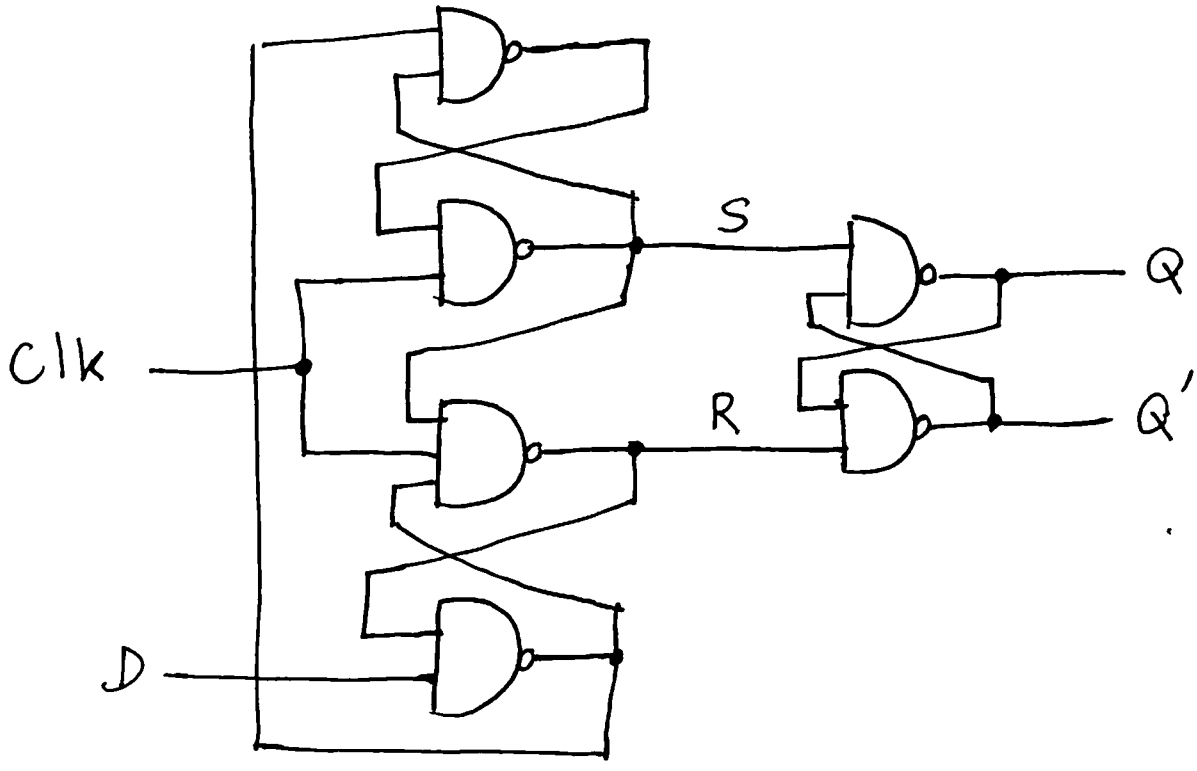
We can implement an edge triggered D flip-flop using two D-latches.



The first latch, called the master is enabled when Clk. input is in state 1 while the

the slave is enabled when clock transitions to 0. So, any change in the input D does not affect the output Q as long as Clk is high. The output Q only changes when Clk goes low. In this situation the master is disabled and any change in D has no effect on Y and, consequently on Q.

Another implementation, using three SR latches is:

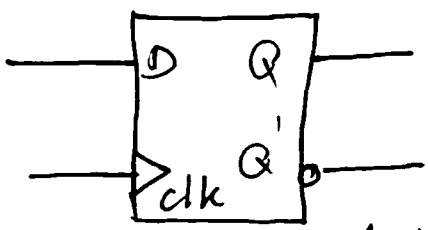


In this circuit as long as Clk = 0, we have S = 1 and R = 1 and, therefore, the output

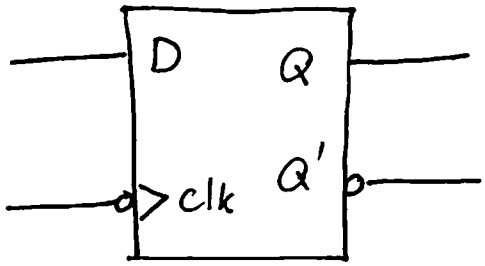
remains unchanged. When Clk goes to 0 if $D=0$ we will have $R=0$. This will make $Q'=1$ and $Q=0$, i.e., it resets the flip-flop. Now, if there is any change in the input D, since $R=0$, it will not change the output of the lower most NAND gate and, therefore, it will not affect the output. When Clk goes to 0, R becomes 1 and we will have $S=R=1$ and the output remains unchanged.

On the other hand, if $D=1$ when Clk goes from 0 to 1 then the output of the lower most NAND will be 0 and this makes $S=0$. This causes the circuit to go to the set state ($Q=1$, $Q'=0$). Any further change in D while $Clk=1$ will have no effect on the output.

A D-flip-flop is represented by the symbol



if it is positive-edge triggered, and by:



if it is negative-edge triggered.

J-K flip-flop

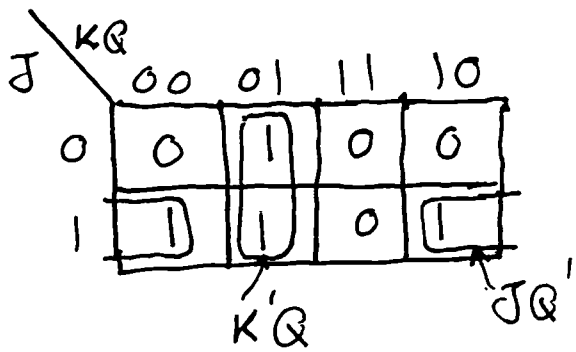
J-K flip-flop has two inputs J and K when J=1, K=0, the flip-flop is set, i.e., Q=1. When J=0, K=1, it resets. When J=K=0, the output does not change and when J=K=1 the output is inverted. Let's try to design a J-K flip-flop using a D-flip-flop. Note that the output of a D flop-flop will have the value of the D input. So, in order to build a JK flip-flop out of a D-flip-flop, we need to design a Combinational circuit that has as its input J, K, Q and Q' and its output is the value required for the above mentioned changes.

The output of the combinational circuit is connected to D.

The following table gives the value of D for different combinations of J, K, Q.

J	K	Q	D
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
1	1	1	0

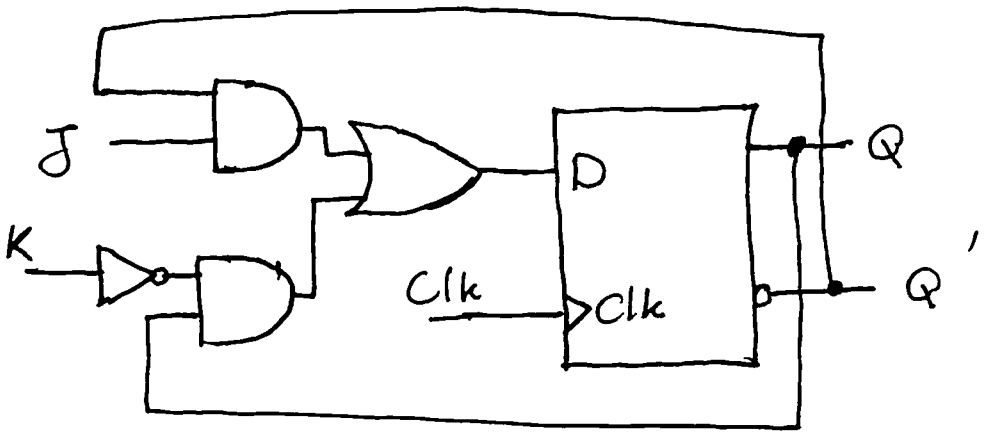
The K-map for D is



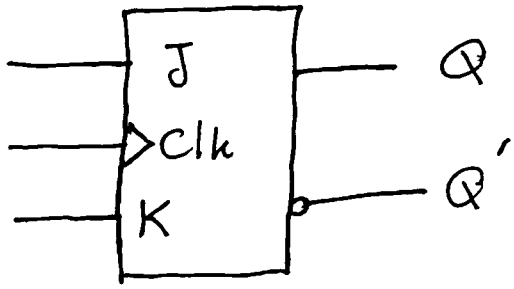
So,

$$D = JQ' + K'Q$$

The circuit diagram for the J-K flip-flop is:



The Symbol for J-K flip-flop is



T - flip-flop

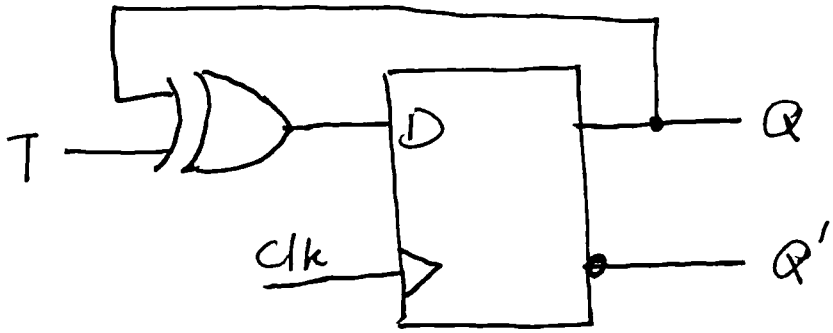
Another type of flip-flop is toggle (T) flip-flop.

T flip-flop has one input, T. When T=0 the state of the flip-flop remains unchanged. When T=1, the flip-flop toggles, i.e., if Q=1, after the application of the clock, Q becomes 0. While if Q=0 and T=1, then Q changes to 1.

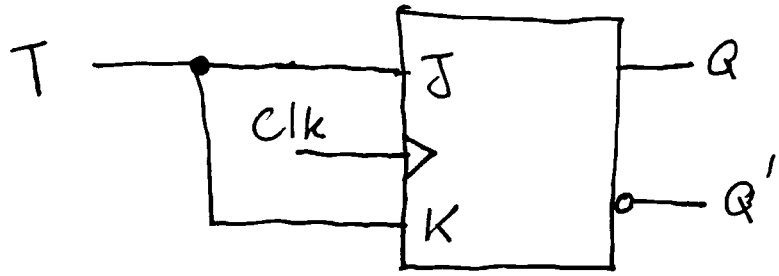
Again if we want to implement a T-flip-flop using a D-flip-flop, we relate the T and Q to the D input using a truth table:

T	Q	D	
0	0	0	no change
0	1	1	no change
1	0	1	toggle
1	1	0	toggle

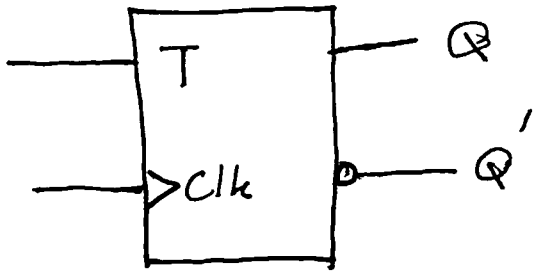
It is clear that $D = T \oplus Q$. So,



We can also implement a T-flip-flop using a JK flip-flop by connecting T to both J and K inputs:



The symbol for T flip-flop is:



Characteristic Tables and Characteristic Equations:

Characteristic Tables (or Equations) describe the operation of a sequential circuit, such as a flip-flop in tabular (or mathematical)

For a JK flip-flop the characteristic table is:

J	K	Q(t+1)	
0	0	Q(t)	no change
0	1	0	reset
1	0	1	set
1	1	Q'(t)	invert

and the characteristic Equation is:

$$Q(t+1) = JQ'(t) + K'Q(t)$$

where Q(t) and Q(t+1) are the state of the flip-flop, before and after the application

of the clock signal, respectively.

For D flip-flop, we have

D	Q(t+1)
0	0
1	1

So, $Q(t+1) = D$ is the characteristic equation.

For T flip-flop, the characteristic table is:

T	Q(t+1)	
0	Q(t)	no change
1	Q'(t)	toggle (invert)

The characteristic equation is:

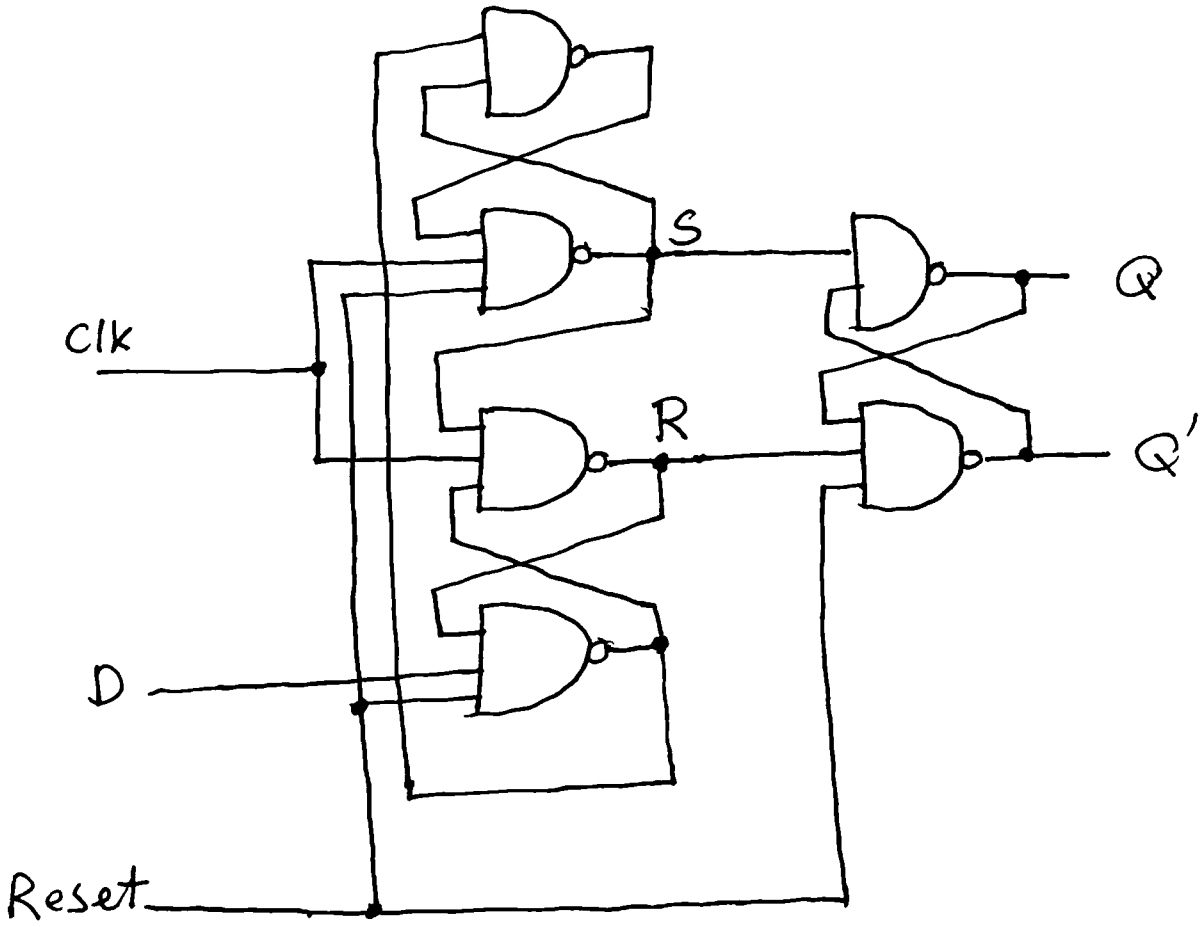
$$Q(t+1) = T \oplus Q(t) = T'Q(t) + TQ'(t)$$

Direct inputs

Sometimes, we need inputs that put a flip-flop in a given state no matter what the clock is. For example, we can have a reset input that resets a flip-flop. The reset inputs of

all flip-flops in a circuit may be connected and used to put all memory elements in a reset (zero) state.

Following is an example of a flip-flop with a reset state



The symbol for this flip-flop is