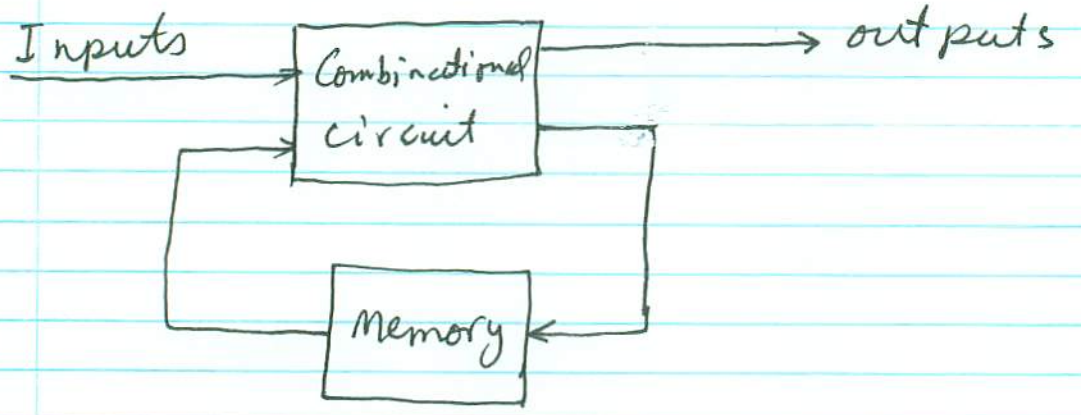
Lecture 13, Feb. 28, 2007

Sequential Circuits

The circuits we have seen so far, in this course, all have been Combinational circuits. A Combinational circuit has no memory. That is, its output only depends on its input.

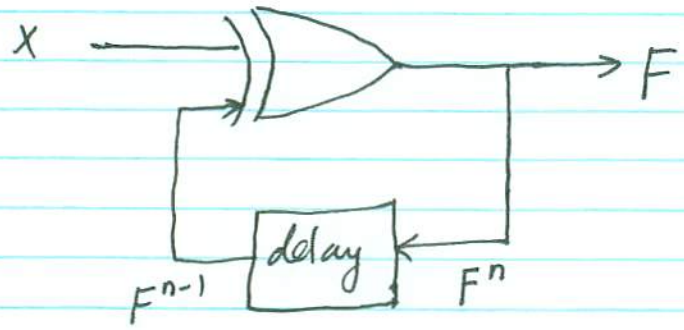
A sequential circuit, on the other hand has some storage (memory) element. Depending on the content of its memory, a sequential circuit can be in a different state.

A block Diagram of a sequential circuit is:



Let's take a simple example. Assume that we have an exclusive-OR with two inputs x and y and an output F.

If we take the output, delay it and connect it to the input y, we have the following sequential system:



The output of this system not only depends on x, but also on the previous value of the output. Since, this circuit has one memory element (one delayed bit), it has two states. Say, it is in state 0 if the value of F in the previous stage (call it F^{n-1}) was zero and is in state 1 otherwise. The <u>state table</u> for this circuit is:

Present state	Next state		output (F^n)	
F^{n-1}	x=0	x=1	x=0	x=1
0	0	1	0	1
1	1	0	1	0

There are two types of sequential circuits:

- The Synchronous circuits
- The Asynchronous circuits.

In the synchronous circuits, state and also outputs change at discrete times dictated by a clock.

In an asynchronous sequential circuit, the output of each gate is defined based on its input and gates delay. In other words, the state and outputs can change at any time. The problem with asynchronous circuits is the possibility of encountering instability due to feedback. The delay (or memory element) of asynchronous circuits is the delay of the gates itself.

In this course, we only study synchronous circuits.

The delay in the Synchronous Sequential circuits is implemented using Latches and Flip-Flops.

A Latch or a Flip-flop is a device that store one bit. Depending on its content a latch (or a flip-flop) can be in state 0 or 1.

The difference between a latch and a flip-flop is that a latch is a flip-flop that operates with the signal level. That is the output of a latch depends on the level of its input.

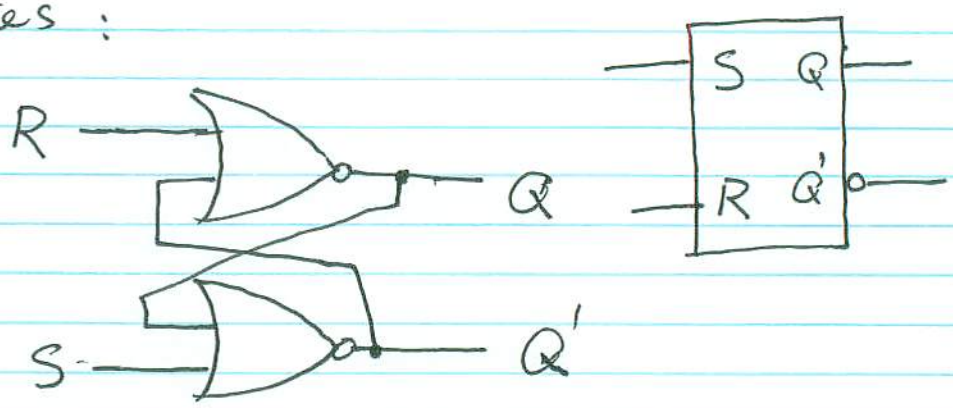
A flip-flop, in general, however, operates with the signal edge (transition). Latches are used to implement flip-flops.

Here are different types of Latches and Flip-flops:

- SR Latch

This is a circuit with two inputs Set (S) and Reset (R). It has two inputs: Q and its inverted version Q'.

A latch can be constructed with two cross-coupled NOR or two cross-coupled NAND gates:



When $Q=1$ and $Q'=0$ we say that the latch is in the set state.

When $Q=0$ and $Q'=1$, the latch is in the reset state.

When $S=1$ and $R=0$

we get $Q'=0$ and $Q=1$. That is the latch goes to the set state.

When $S=0$, and $R=1$

then $Q=0$ and $Q'=1$. That is the latch is reset.

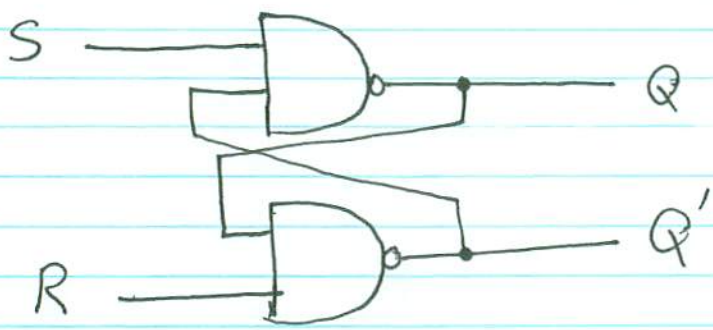
When $S=0$ and $R=0$ then the latch stays in the state it was, i.e., no change in state.

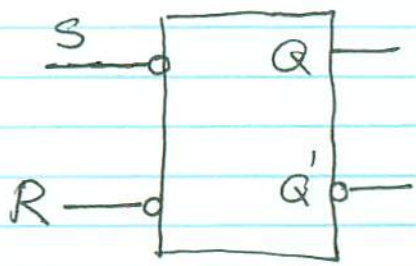
When $S=R=1$ then $Q=0$ and $Q'=0$.
This is an undefined state and makes the next state of the latch unpredictable when the inputs go back to zero.

The following table shows the relationship between S, R, Q, Q'.

S	R	Q	Q'	
1	0	1	0	
0	0	1	0	(after S=1, R=0)
0	1	0	1	
0	0	0	1	(after S=0, R=1)
1	1	0	0	← (forbidden state)

SR latch implementation using NAND gates

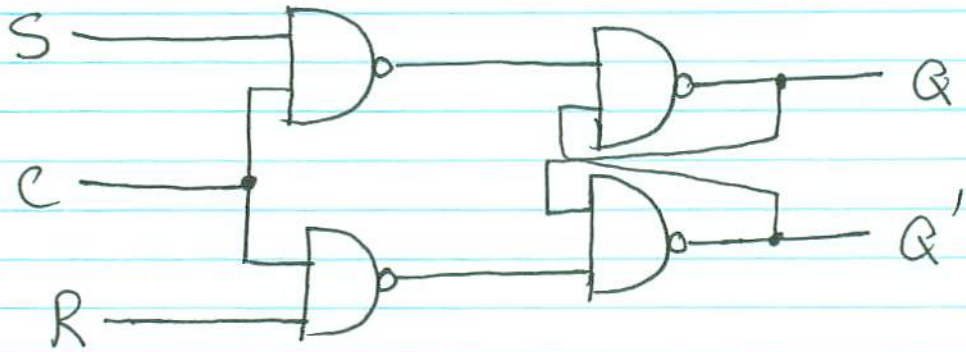




S	R	Q	Q'	
1	0	D	1	
1	1	1	0	(after S=1, R=0)
0	1	1	0	
1	1	1	0	(after S=0, R=1)
0	0	1	1	forbidden inputs (00)

Note that the inputs of the NAND latch are the inverted version of those for NOR implementation. That is why this is called S'-R' latch.

A control input can be added to a SR latch. As long as this control input, C, is at zero, the output remains unchanged. When, want the state of the latch to change, we set C=1 and apply the appropriate value to inputs S and R.

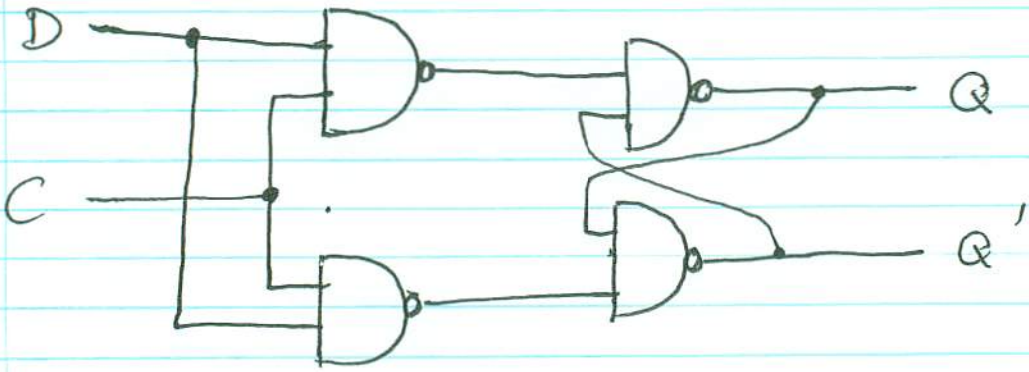


C	S	R	Next State
0	X	X	No change
1	0	0	No change
1	0	1	Q=0, Q'=1 (rest state)
1	1	0	Q=1, Q'=0 (Set state)
1	1	1	Indeterminate.

D latch

In order to avoid the problem of indeterminate state we make sure that S and R are never equal to 1 at the same time. We do this by always making $R = S'$. That is when $S=1$, $R=S'=0$ and when $S=0$, $R=S'=1$.

Note that when $R=S'$, we only need one input S. We call this input D for data. We give D to one input and D' to another.



C	D	Next State
0	X	No change
1	0	$Q=0$, $Q'=1$, reset
1	1	$Q=1$, $Q'=0$, set

The Symbol for D-latch is: