Quinn Roemer

Engineering – 303

Lab 6

3/8/2017

Introduction/Description

The purpose of this lab was to design several circuits that would be capable of remembering their input. In other words, these circuits should be capable of retaining within themselves whatever was inputted by the user of the circuit. In this lab, we learned about two types of circuits that would enable us to perform this task. The first one was called a Latch gate and the second called a Flip-Flop gate. A latch gate works by storing data asynchronously and a Flip-Flop gate works by storing data synchronously based on a clock. Please note, for the duration of the lab, in the truth tables that are presented an X represents an indeterminate value and the words "Last Value" mean that the last value held by the circuit still holds true.

Design

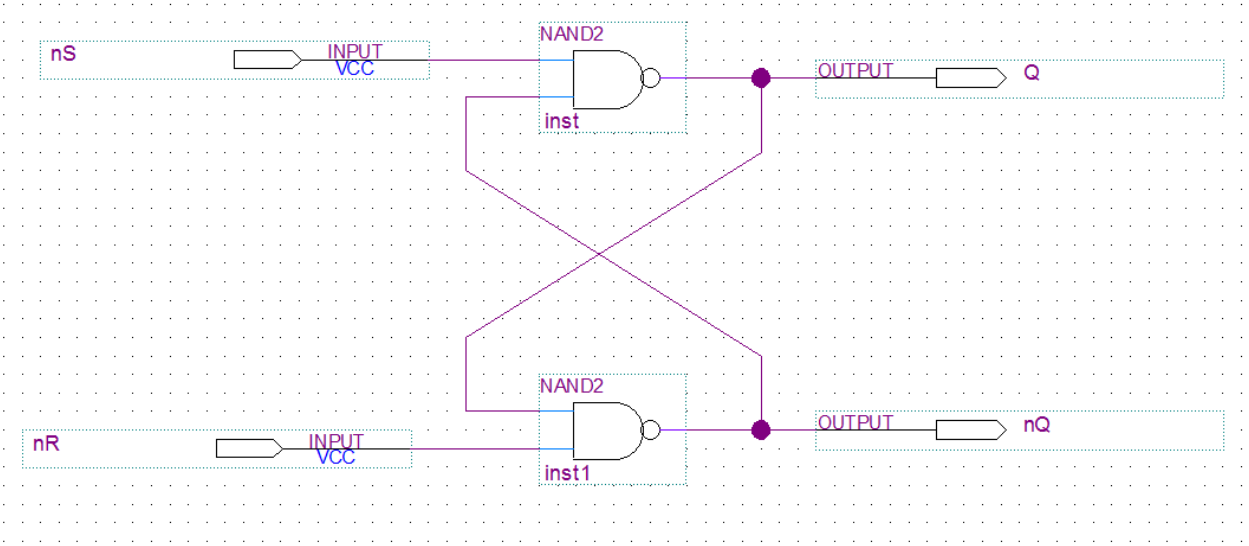
Part 0 – S'R' Latch

In the first part of the lab, I was instructed to build a memory circuit using the Latch design. This circuits two inputs would be normally high. Or in other words a binary 1. When the input named nS becomes binary 0 the output named Q will hold a binary 1. Likewise, when the input name nR becomes a binary 0 the output Q resets to binary 0. The output nQ is just the opposite of what Q is currently outputting.

Here is the truth table for the circuit.

Inputs		Outputs	
nS	nR	Q	nQ
0	0	x	x
0	1	0	1
1	0	1	0
1	1	Last Value	Last Value

Here is the Block-Diagram design for the S'R' Latch circuit.



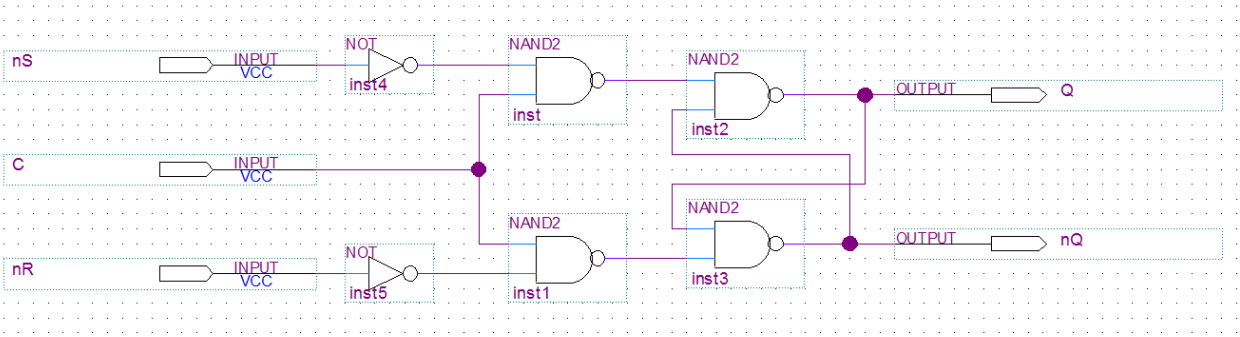
Part 1 – S'R' Latch with Control

This circuit performs the same task as the previous circuit and is very similar in design. The only addition to this circuit is an additional input called C. If the input C is relaying a binary value of 1 to the circuit it performs just as the previous circuit did. However, if the input C is relaying a binary value of 0 then the last value that Q held will remain true and the other inputs named nS and nR will be ignored.

Here is the truth table for the circuit.

Inputs			Outputs	
C	nS	nR	Q	nQ
1	0	0	X	X
1	0	1	0	1
1	1	0	1	0
1	1	1	1	0
0	0	0	Last Value	Last Value
0	0	1	Last Value	Last Value
0	1	0	Last Value	Last Value
0	1	1	Last Value	Last Value

Here is the Block-Diagram for the S'R' Latch with control.



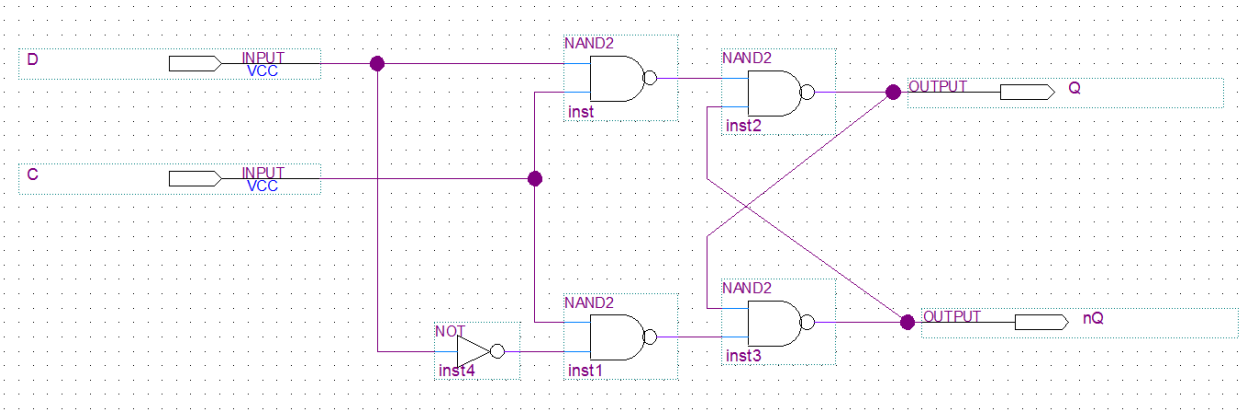
Part 2 - D Latch with Control

In this part of the lab, I was instructed to build another memory circuit that would use the Latch concept. The only difference with this circuit is that the inputs of nS and nR and now tied together with one input called D. This insures that they will always hold a value opposite of each other since one end of input D is connected to a NOT gate. This prevents the circuit from receiving a double negative input which would have previously caused the circuit to error.

Here is the truth table for this circuit design.

Inputs		Outputs	
C	D	Q	nQ
1	0	**0**	**1**
1	1	**1**	**0**
0	0	**Last Value**	**Last Value**
0	1	**Last Value**	**Last Value**

Here is the Block-Diagram for the D-Latch with Control circuit.



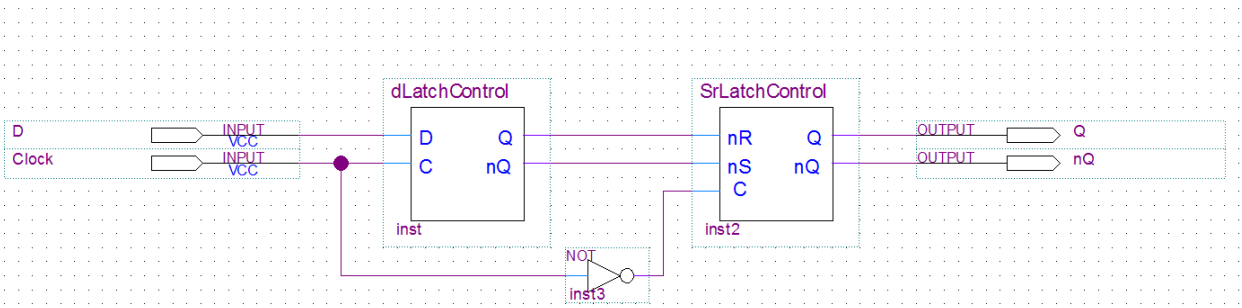
Part 3 - D Flip-Flop

In the last part of the lab, I was told to design a circuit that would use the Flip-Flop technique instead of the Latch method. To design this circuit I connected the two previously designed Latch circuits in a master-slave configuration. This circuit samples the input D on the input Clocks negative edge. In other words, whenever the input Clock dips down to a binary 0 the output Q will match the value of D, otherwise, this circuit will hold its previous value. As in the other circuits, the output nQ represents the opposite of the output Q.

Here is the truth table for the circuit.

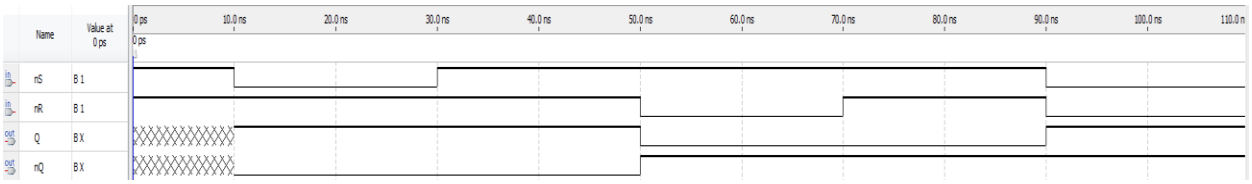
Inputs		Outputs	
D	Clock	Q	nQ
0	0	X	X
0	1	0	1
1	0	1	0
1	1	Last Value	Last Value

Here is the Block-Diagram for the D-Flip-Flop circuit.

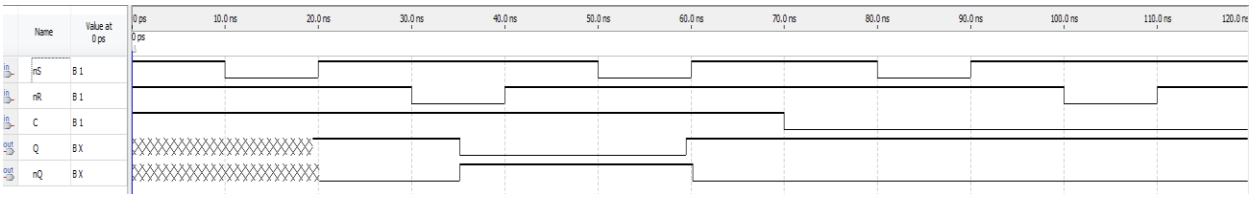


Testing

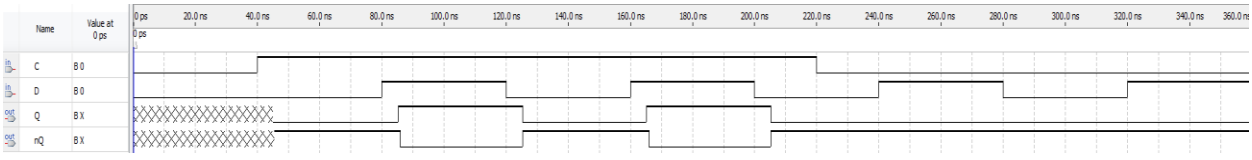
When testing the S'R' Latch circuit I encountered no problems and it performed as expected for every single input combination.



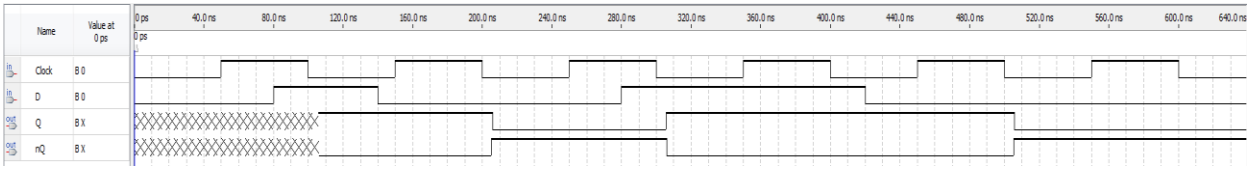
When testing the S'R' Latch with Control circuit I encountered no problems and it performed as expected for every single input combination.



When testing the D Latch with Control circuit I encountered no problems and it performed as expected for every single input combination.



When testing the D-Flip-Flop circuit I encountered no problems and it performed as expected for every single input combination.



Conclusion

In this lab, I learned how to create a few simple memory circuits that were capable of storing the value of previous inputs. I learned in this lab that there are two simple circuit designs that can be utilized to make these circuits. The first being the Latch circuit and the second the Flip-Flop method. If I was to perform this lab again I would delve deeper into the circuit's workings and concentrate on how the circuits perform their task. Overall this lab proved to be enjoyable and it was interesting to see how some basic memory circuits actually work.