SACRAMENTO
STATE

MARIO PALACIOS

LAB COURSE (CpE 64) SECTION #20

WEDNESDAY

INSTRUCTOR: EMMANUEL DUPART

PART 1. NAND GATE VERSION OF RS LATCH

Description:

Write a data flow description in Verilog for a NAND gate version of the RS latch. Also draw a diagram with the signal names. Then explain the purpose of the schematic found in figure four.

Problem Description:

This part of the lab allows you to explore what a latch is and its functions. The NAND version latch is an active low. Often the inputs of the switches can be true when they are closed or shorted to the ground. The diagram of the NAND RS latch can be found in figure one, along with its Verilog code on figure two, and finally the waveform displaying the outputs on figure three.

Engineering Data:

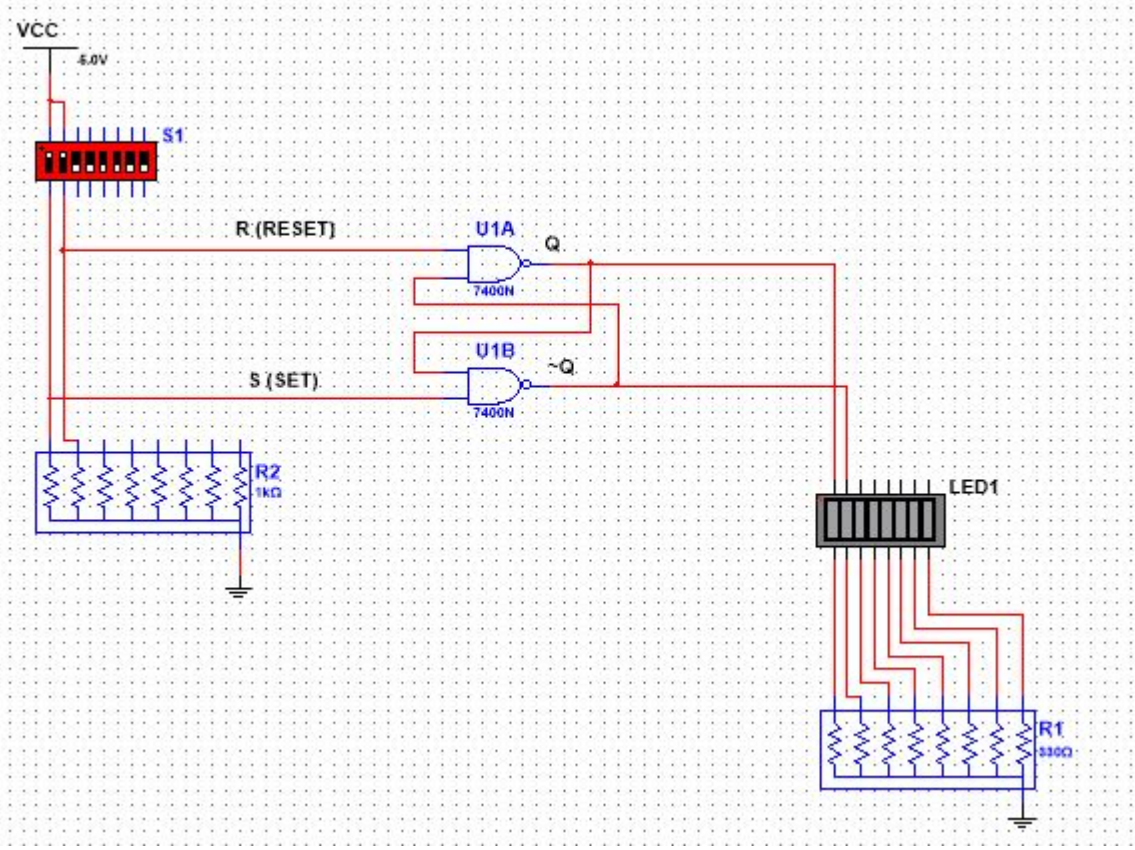

Figure 1. Multisim Design for NAND RS Latch

```
1  //NAND RS Latch
2  module rs_latch(R,S,Q,NQ);
3      input R,S;
4      output Q,NQ;
5
6      assign Q = ~(S&NQ);          //S(SET) is active low
7      assign NQ = ~(R&Q);          //R(RESET) is active low
8  endmodule
```

Figure 2. Verilog Code in "Data Flow" Modeling for a NAND RS Latch

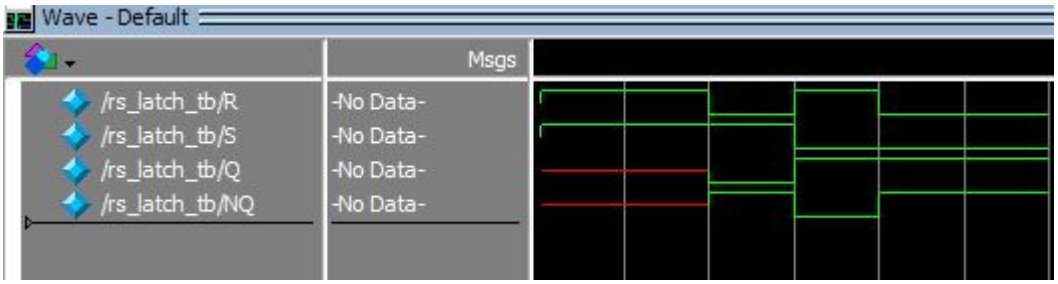

Figure 3. Waveform of the NAND RS Latch Displaying the Inactive State

Conclusion:

The NAND RS latch is an active low, latch meaning that it will activate (latch) when there is a low output and will reset when there is a high output. However when it has two low inputs the output will be undefined or "not valid." This part of the lab took me about 15 minutes to complete.

Question #1:

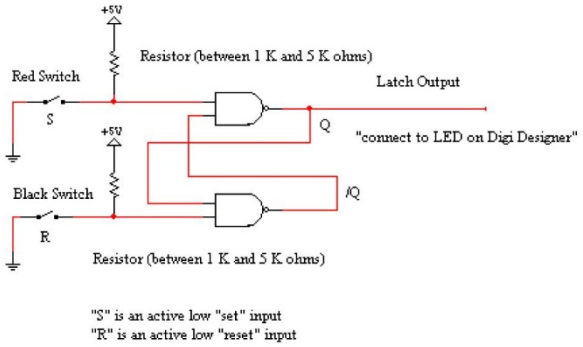

Figure 4. NAND RS Latch

The purpose of the two resistors for the R and S is to set the two switches to active low. An active low will be turned on when the input is at 0V and turned off when the input is +5V. The resistors will make sure when the switches are turned on (closed), the current from the +5V will flow to ground, creating the active low state.

PART 2. NOR GATE RS LATCH

Description:

Write a data flow description in Verilog for a NOR gate version of the RS latch. Then draw a diagram with the signal names.

Problem Description:

This part of the lab allow you to continue exploring what a latch is and its functions. The NOR version latch is an active high. The diagram of the NOR RS latch can be found in figure five, along with its Verilog code on figure six, and finally the waveform displaying the outputs on figure seven.

Engineering Data:

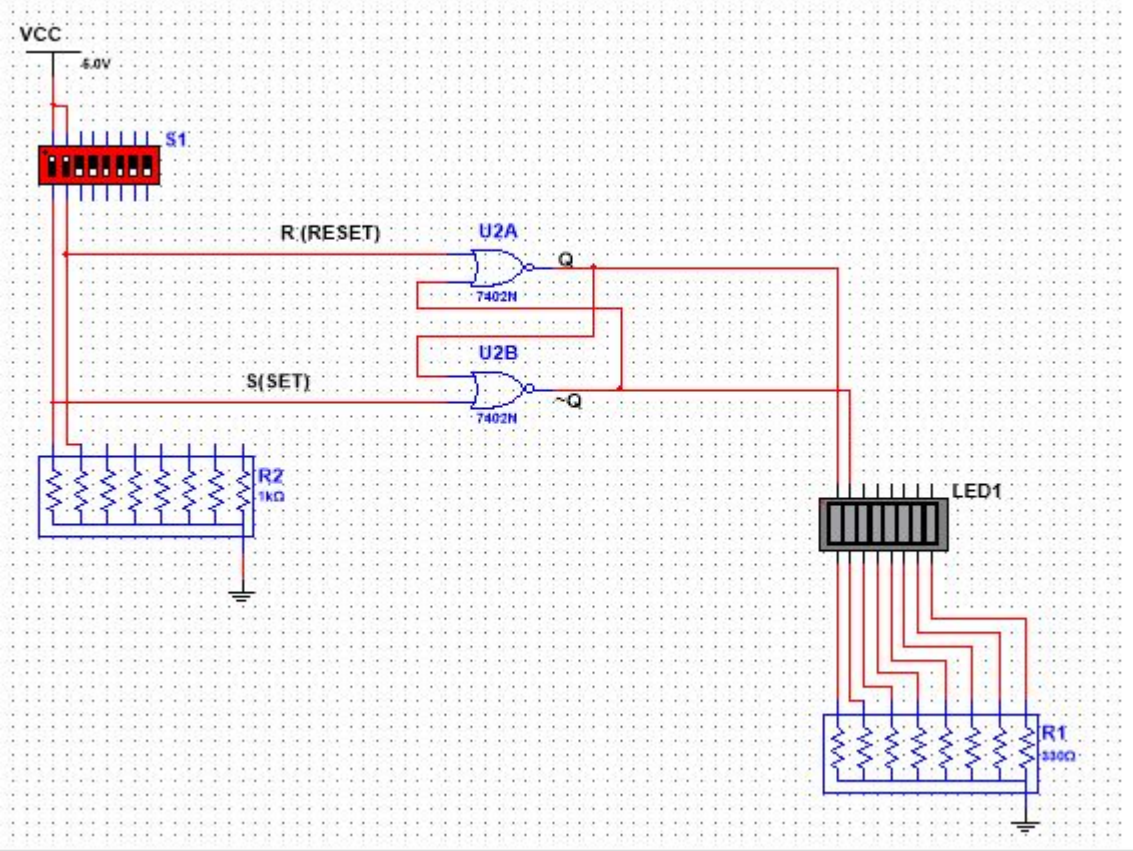

Figure 5. Multisim Design for NOR RS Latch

```
1  //NOR gate RS Latch
2  `timescale 1ns / 1ps
3  module rs_latch(R,S,Q,nQ);
4      input R,S;
5      output Q,nQ;
6
7      assign Q = ~(R|nQ);
8      assign nQ = ~(S|Q);
9  endmodule
```

Figure 6. Verilog Code in "Data Flow" Modeling for a NOR RS Latch

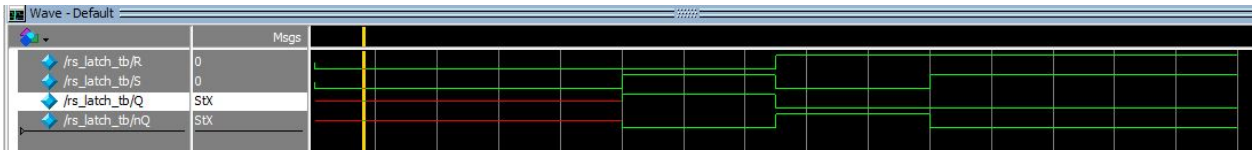

Figure 7. Waveform of the NOR RS Latch Displaying the Inactive State

Conclusion:

The NOR RS latch works as an "active high" latch, meaning it latches when the output is high and will reset when it has a low output. However it does have an undefined state where the inputs are both high, as seen by the red lines in figure seven. When connected to an LED bar the output lights will continuously turn off and on, one after another. This part of the lab took about 15 minutes to complete.

Question #2

The difference when using the S and R inputs with the NOR gate version is that in order for it to latch (set) the inputs must be an active high. As for the NAND gate version the inputs are set to active low. This set means for a NAND the S and R are negated, so when ~S = 0 and ~R = 1, the state is set. For a NOR gate to be in a Set State the inputs must be S = 1 and R = 0. Both of them having undefined states, for the NOR version both inputs must be a one and for the NAND version both inputs must be zero.

PART 3. NOR GATE VERSION OF THE D FLIP-FLOP

Description:

Write a Data Flow model description for a NOR gate version of the D Flip-Flop. Draw a diagram with signal names. We will explore the meaning of "edge-triggered.

Problem Description:

The D Flip-Flop works on an edge-trigger case where it can set and reset on either positive or negative edge of a clock signal. The diagram for the NOR D Flip-Flop can be found on figure 8. There were a total of four equations that were used to generate the code. The equations found were used to develop Verilog code and it's waveform that can be seen on figure nine and ten.

Engineering Data:

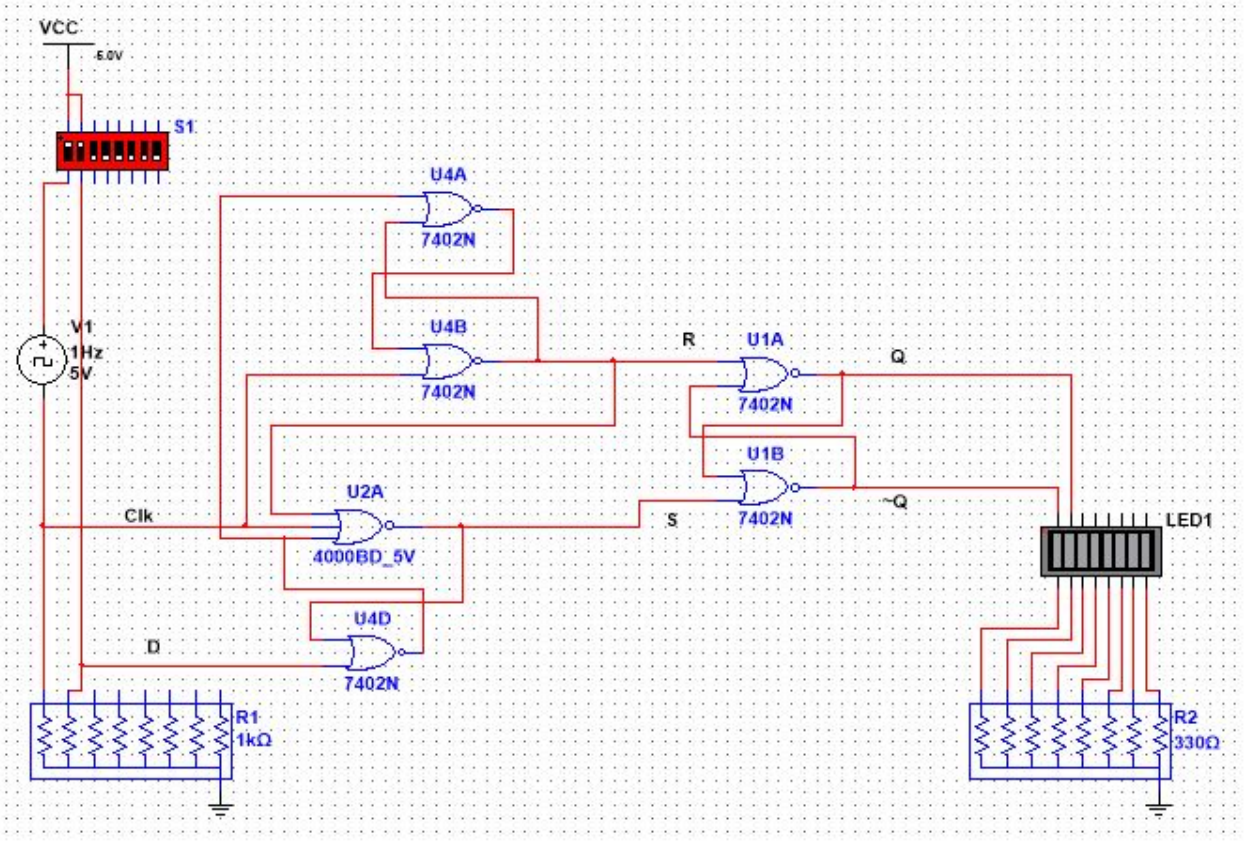

Figure 8. NOR D Flip-Flop with Signal Names and Clock Signal

```
1   //NOR Gate D Flip-Flop
2   module D_FF(D,clk,S,R,Q,nQ);
3       input D,clk;
4       output S,R,Q,nQ;
5
6       assign S = ~(R|clk|~(D|S));
7       assign R = ~(clk|~(R|~(D|S)));
8       assign Q = ~(R|nQ);
9       assign nQ = ~(S|Q);
10  endmodule
```

Figure 9. NOR D Flip-Flop in "Data Flow" Modeling