Quinn Roemer

Engineering – 303

Lab 12

4/24/2017

# Introduction/Description

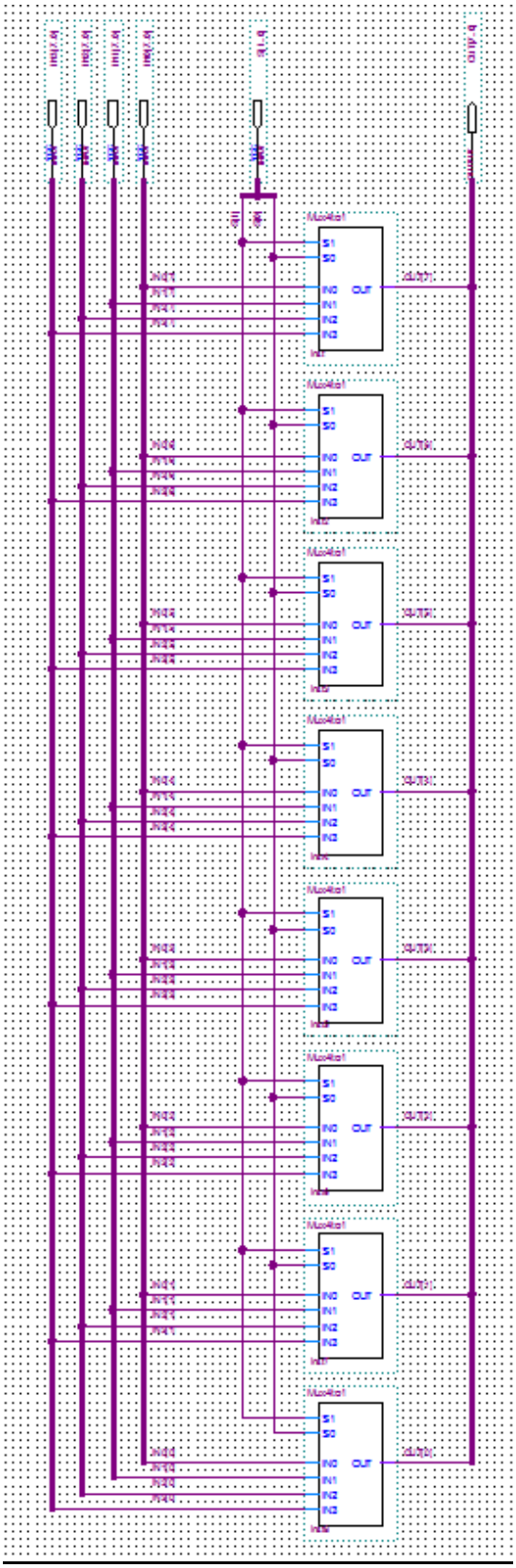
The goal of this lab was to implement several circuit designs that would be used to create the Register File of our Central Processing Unit. This Register File is capable of storing within its memory a grand total of four 8 bit numbers. Since it contains only four 8 bit registers each made from eight 1 bit registers. This Register was put into a circuit that was capable of reading a couple inputs and determining what register to send the data to and what register to output. In this lab, I was instructed to create a total of 4 circuits.

# Design

## Part 0 – 8 bit 4 to 1 Mux.

The first circuit I had to implement was a design for a circuit called an 8 bit 4 to 1 Mux. This circuit essentially looks at four 8 bit binary numbers and then chooses what number to output based on the current value of the 2-bit selection number. Please note, for this circuit and all of the other circuits contained in this lab no Truth Tables or Kmaps were used or created.

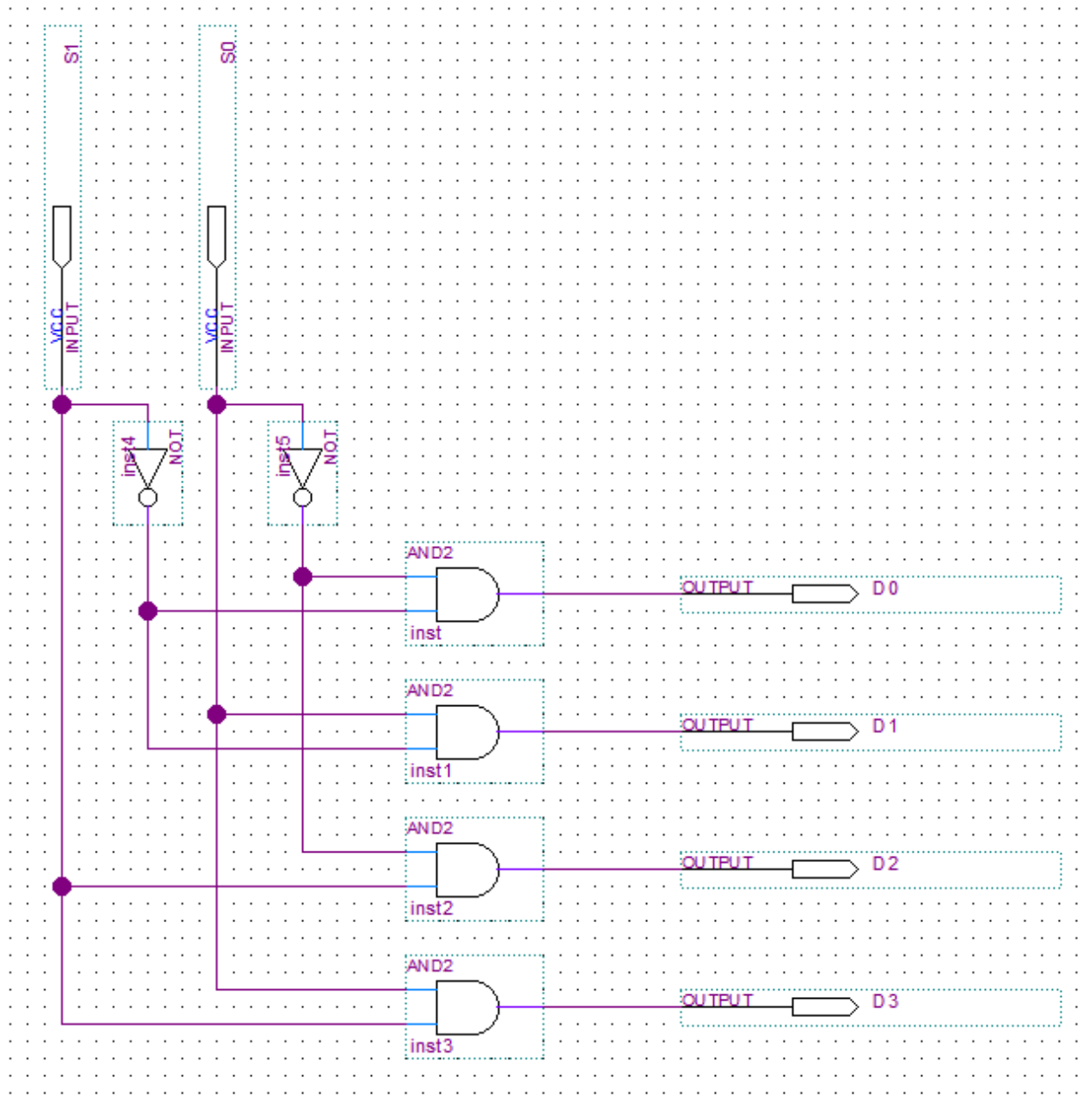
Here is the Block-Diagram design for the 8 bit 4 to 1 Mux.



## Part 1 – 2 to 4 Line Decoder.

In this part of the lab, I was instructed to implement a circuit called a 2 to 4 Line Decoder. This circuit accepts a 2-bit input and then outputs a binary 1 on one of its four outputs depending on what binary number is currently be inputted. For example, if its input is currently 00 it would output a 1 on output 0. If the input is 01 it would output a 1 on output 2, and so on.

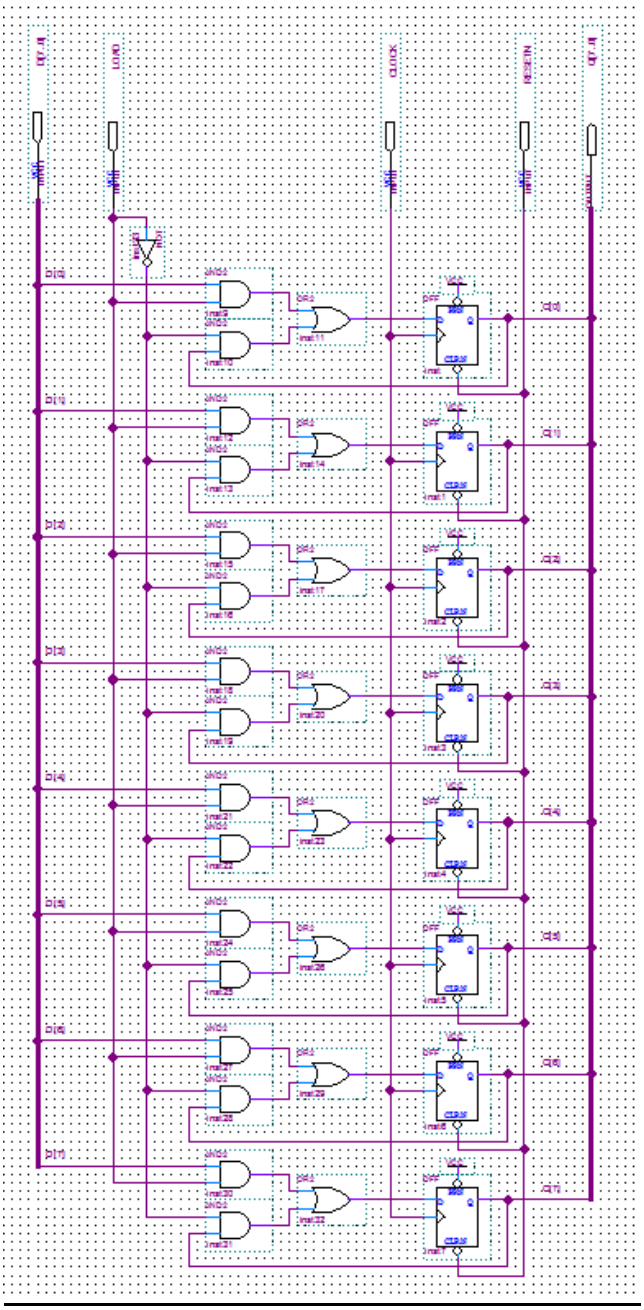
Here is the Block-Diagram for the 2 to 4 Line Decoder.



## Part 2 – General 8 bit Register.

This circuit in the lab is called an 8 bit Register. It is capable of storing one 8 bit number. This circuit accepts an input of an 8-bit number which is then stored in the registers of this design. Please note, this circuit does not store the number until the registers are triggered to accept the number on the positive edge of the clock input.

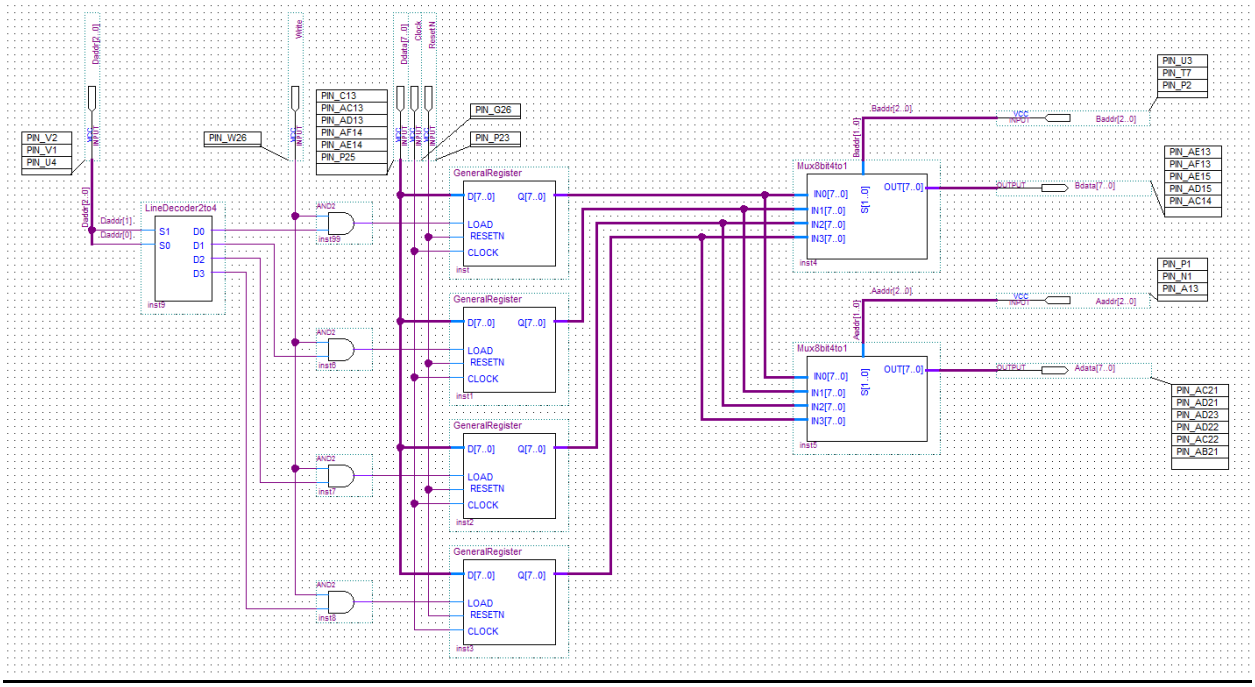
Here is the Block-Diagram design for General 8 bit Register.



## Part 3 – Register File.

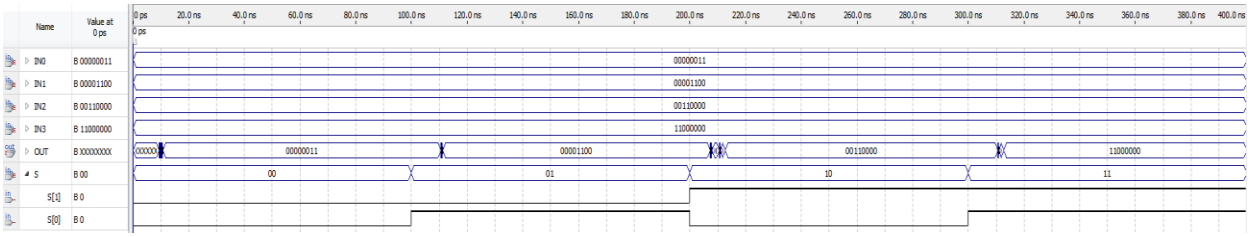
The last circuit that we created in the lab was called the Register File. This circuit involved all of the other circuits that we created in the lab. Essentially this circuit accepts an input number with a maximum length of 8 bits. It then stores that number into one of its registers depending on the current value of the selection input. Then, depending on the value of two separate selection inputs will output the number stored in one of the registers.

Here is the Block-Diagram for the Register File.

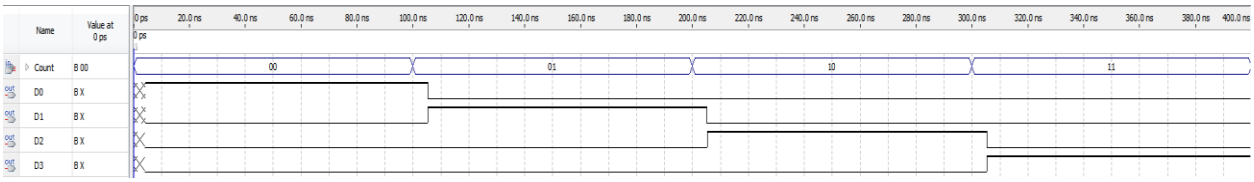


# Testing

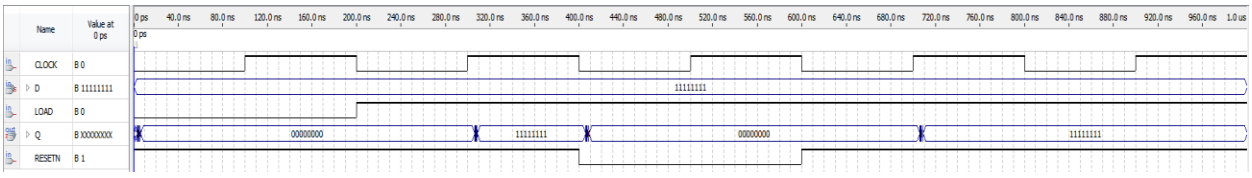
When testing the 8 bit 4 to 1 Mux I encountered no problems and it performed as expected for every single input combination.



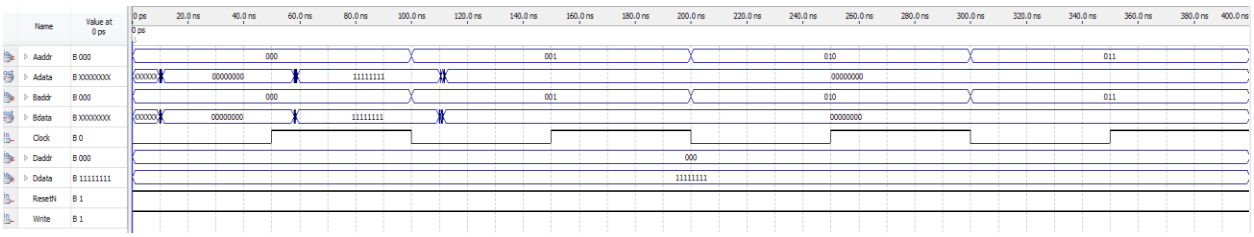

When testing the 2 to 4 Line Decoder I encountered no problems and it performed as expected for every single input combination.

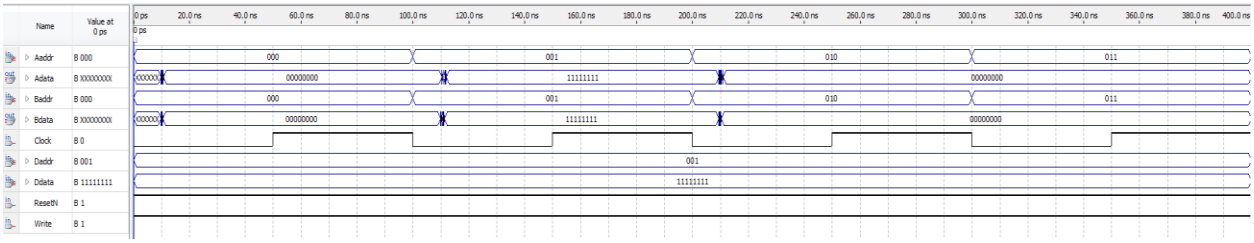


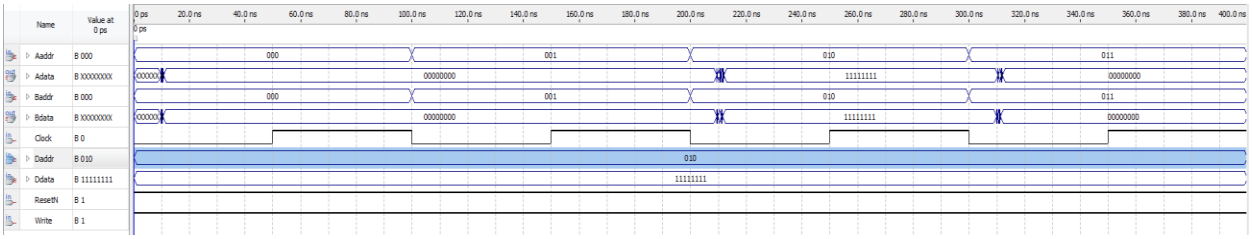

When testing the General Register I encountered no problems and it performed as expected for every single input combination.

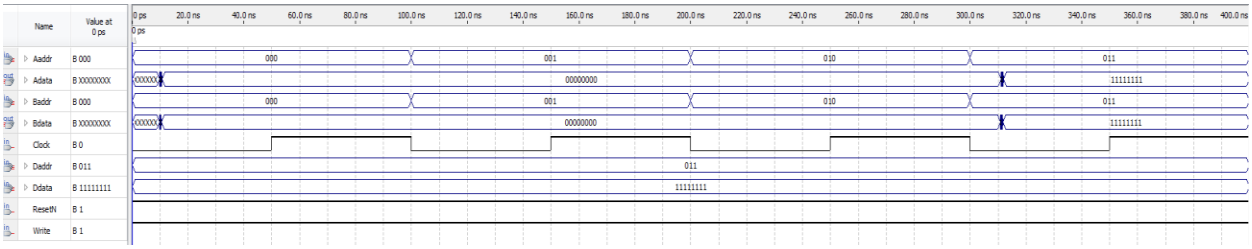



When testing the Register File I encountered no problems and it performed as expected for every single input combination. Please note, this circuit has 4 individual waveforms that each test one of the register in the design.






# Conclusion

In this lab, I learned how to create many of the individual parts contained in my Register File unit portion of the CPU. I learned that all of the separate circuits inside the Register File each serve a purpose. For example, the 2 to 4 Line Decoder serves the purpose of selecting which register to store the data inside. The 8 bit 4 to 1 Mux serves the purpose of determining which register to output from. And, the General Register serves the purpose of actually storing the byte. Overall this lab proved to be enjoyable and I am excited that I continue to understand the gist of everything inside the circuits that I have been creating.