Makenzie Brian

December 4, 2015

ECE 272

Final Lab Write-up

Introduction:

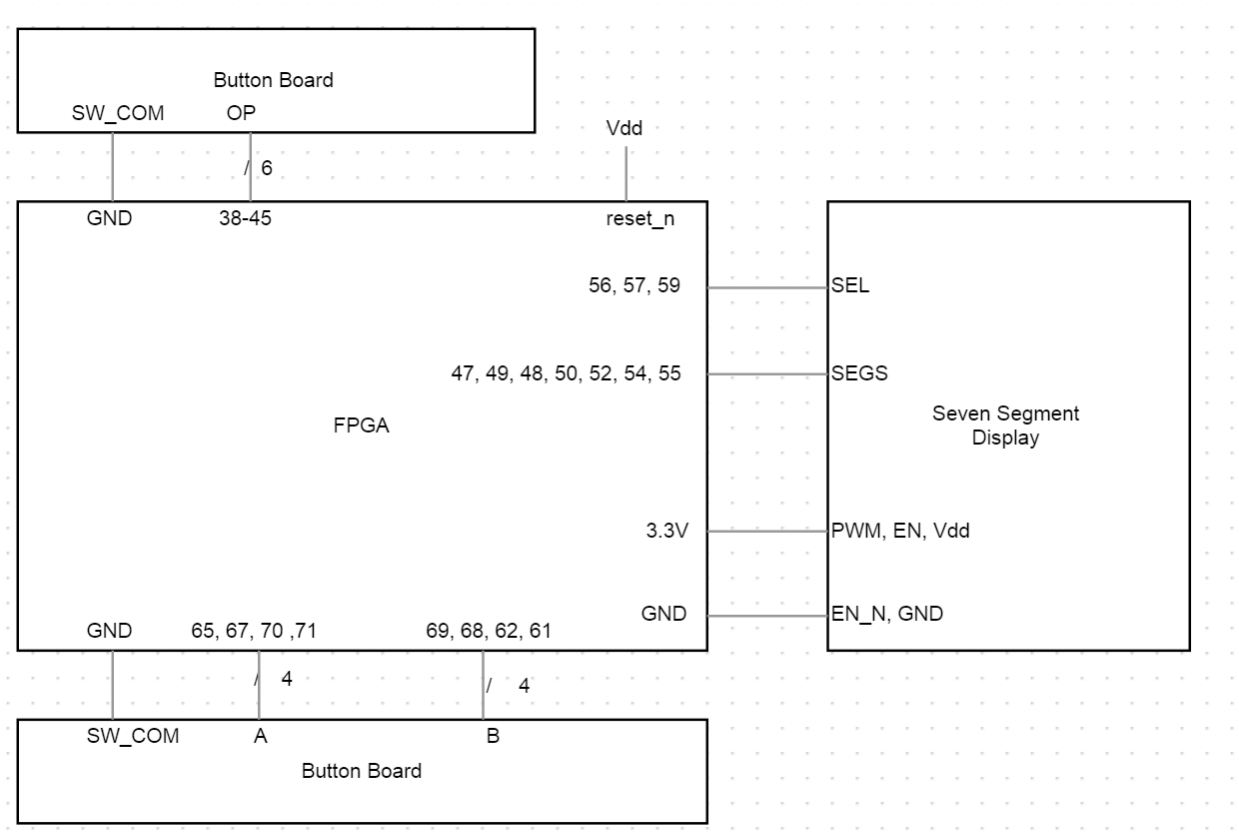
For this assignment, I made an ALU. The inputs are each 4 bits and there are 6 operations. I chose this project because it wasn't too complicated and fit within the time constraints I had, but also seemed interesting and worth spending time on.

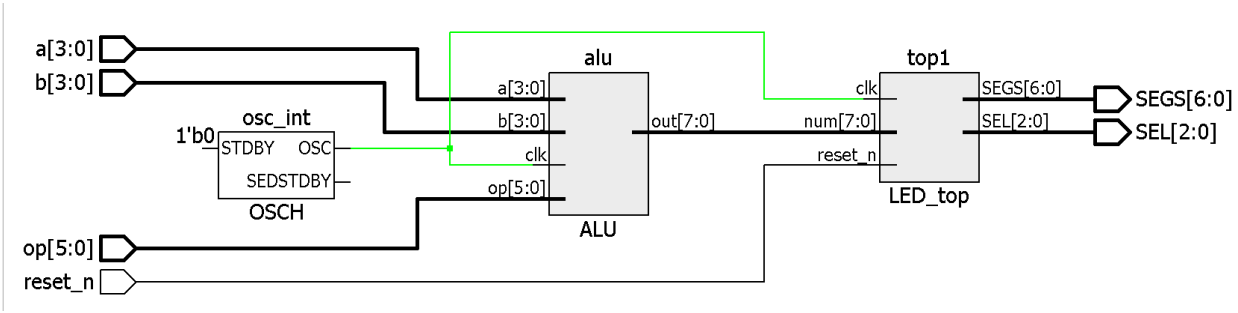
Background:

ALU means Arithmetic Logic Unit. An ALU is generally a component within a computer's central processing unit and performs bitwise mathematical operations on binary numbers and is the last component to perform calculations in the processor. An ALU selects from a preset variety of options based on the input to the system. This option is generally a function that will be performed on numbers that can be either preset or entered into the system (in the case of this project they are entered by the user). Common functions for an ALU include adding, adding with carry, subtracting, subtracting with borrow, two's complement, incrementing, and decrementing for arithmetic operations. Bitwise logical operations can include AND, OR, Exclusive OR, and One's Complement. Bit shifting operations include arithmetic shift, logical shift, rotate, and rotate through carry.

Design:

The inputs are each 4 bits and there are 6 operations. The operations are addition, subtraction, division, modulus, another modulus, and multiplication (there is also a default set to 255 for testing purposes). The button boards and seven segment display are connected to the FPGA as shown below via wires. In the Verilog, I used lab 4 to control the seven segment display and added the ALU and a top file to hold all the modules. My process was to write the Verilog and attach the previous files, which took about five minutes, then to fix any errors that were produced. Next, I connected the wires on my FPGA and set up my spreadsheet. Then I programed the FPGA and fix a few other issues before it worked.





Validation:

I tested the design by pressing the various buttons and observing that the output was correct based on the inputs to the system. For example, pressing the button to add and pressing the proper buttons for a=2 and b=2 should display 4 as a result and it did. I tested all the functions and various numbers through this method to test that the system worked properly. Another way to make sure it worked was to set the default to something besides 0 (which is why I set it to 255) so when I didn't press any buttons it displayed 255.

Conclusion:

I had some trouble with little things on this project (such as forgetting that I took the oscillator out of the file and forgot to put it back in). I also had a few wires connected incorrectly at various points, once to mu buttons for the operation selection and once to my segments on the display (which always happens for some reason). The Verilog was very easy to write because the seven segment display portion could be taken directly from lab 4. This made me wish I had chosen something more challenging for this project but I also had a time constraint.