Ahmed Abdulkareem
ECE 351, HW4
05/30/2016

Introduction

This project was probably one of the most interesting things I did in this class. When I first took this class, I already had some knowledge in Verilog from other courses. For example, in ECE 585, my first homework involved a finite state machine in Verilog. However, I wasn't comfortable with it at all. I used blocking and nonblocking assignments and never really knew the difference between the two until I took this class. This project improved my coding skills in Verilog. What's even better is the fact that we ran the code on actual hardware, not just simulating it in a simulator. Running my code on the board was my first time ever doing that with Verilog. I've never used an FPGA board to run Verilog code on.

Experience

How I Succeeded

I succeeded in this project by doing the following things. I first started somewhat early so I know what to expect ahead of time. I then studied the file that did the pin mappings. I also read the DE1 documentation a little bit. This is to get myself an idea of what's really going on. I then looked at the code provided in the homework 4 release. The code was definitely scary at first, and too much to deal with. But after I had my 5 minutes of freaking out, I sat down carefully and studied the code provided. Then I studied the specs and what I'm supposed to develop. I made my datapath, and control logic modules, and then stimulated them both, but not thoroughly. What I did was I tested them with a simple test bench to make sure it operates just fine, because debugging hardware is not the most fun things you can do. After making sure both modules are functional, I went ahead and downloaded them on the DE1 board, and tested it thoroughly. This way, I didn't have to deal with a lot of debugging the code on the DE1.

Challenges

SInce I'm OK at coding in Verilog especially for the basic stuff we're doing in this course, I didn't have a lot of challenges. I knew how to code finite state machines, muxes, and basic things like that. I think the only challenge was understanding the code provided thoroughly. This wasn’t a requirement, but I hate working with something that I don’t know what it’s doing exactly.

Problems and Solutions

I only had one problem, and thanks to professor Roy for helping me on it. My problem was that I got an Err string displayed on the LCD when inputting the A and B operands. At first, someone might think the out_of_range flag would be the one causing the problem. My first attempt was to actually see the out_of_range flag and why it’s always high. I simulated it and looked at the out_of_range flag, and it wasn’t always high which led me to think that this isn’t the problem. It turned out the problem was that I was setting the mux to show an Err message and stay in that state until I got to inputting the carry in state. After talking to professor Roy, it turned out the problem was that I had a dangling else if statement which was causing a latch to be synthesized. Professor Roy helped me get rid of it by having an else statement to keep the output in the same state. This was the only problem I ran into.

Suggestions and Improvements

I think this was a great assignment. I learned a lot from it. I learned how to run code on the DE1 board which was really interesting. I don’t have any suggestions for this project. I think everything needed was provided.