James Tacey

12/3/2010

Computer Architecture

CPU Design Project Part 5

**What did you learn from this project?**

In this project I learned how to design, simulate, and verify and entire processor. Due to this project, I learned a wide variety of skills. One of the most important skills was the ability to work with VHDL, a hardware description language, in order to develop the various parts of the processor. I also learned how to interface with Quartus II, an important development environment that enables the user to program various FPGA boards, and ModelSim an associated simulation software. Using these tools I designed the cpu from the instruction set, all the way to component instantiation. I learned how to debug software problems and make corrections to my existing design in order to facilitate proper behavior. While my final design did not function properly when downloaded to the board, The simulation results showed that the software behaved as expected. This also taught me the import difference between simulation results, and a live working product. I think the most important thing that I learned was how to take theory and information learned in the class, and turn it into an actual project gave me important experience that I could use in later settings and represented the sum of my knowledge gained in this course.

**What would you do differently next time?**

The primary thing I would do next time is to download my design to the board at a much earlier time to ensure that it would run properly. I would also start work on portions of the project ahead of time as well. Often, the early stages of the project that were due did not represent the amount of work that would be required for the later parts. As I often had free time after completing the early parts, in the future I would work on getting a simple version of the cpu working properly on the FPGA. I would also ask the TA for help at a much earlier point. Near the end of the project, the TA was unavailable for days on end, leading to me being unable receive help from the TA until the day before the last portion of the project was due.

**What is your advice to someone who was going to work on a similar project?**

I would advise this individual to begin working on putting the final design together and testing the physical implementation at an early a stage as possible. Merely having simulation results and a project that is working properly in software does not necessarily mean that the physical implementation on the FPGA will be an easy and smooth process. Even on a prior stage such as combining components into the software cpu may be rife with problems as some components may be asynchronous as compared to clocked, and others may be signed or unsigned. Another important recommendation is to read the documentation and tutorials available on the website. While the specific version of software they use in their examples is older than the one available in the labs, the information contained is very useful and in some cases vital to the success of the project.

**What are the current problems and their possible solutions?**

At this time the cpu does not run on the board at all. I believe it is a race or latch condition associated with the timing of running the cpu at a relatively fast clock frequency of 27 MHz. One possible solution is to use a pushbutton as a clock in order to manually control the board, in addition to hard-coding values into the PC. I attempted this shortly after my presentation and it appeared to be giving some results. Due to time constraints I was not able to further explore this option. I believe the problem lies with either the clock timing or the PC register itself. The best solution would be to continue timing analysis and possible latching errors with the PC itself.