Computer Sciences Department
University of Wisconsin-Madison
CS/ECE 552 – Introduction to Computer Architecture
Project Description

1. Summary

The CS/ECE 552 term project is the complete functional design of a microprocessor called the WISC-SP19. All components of your design will be written in Verilog. As with the course homeworks, the CS/ECE 552 Verilog restrictions apply, and all final code is expected to pass the Vcheck program.

The project will be completed in groups of three. Each group must contain at least one person who has not taken ECE 551 already. You can self-signup for your group on Canvas using the "Student Groups" tab under "People".

The specifics of the microarchitecture and WISC-SP19 architecture are found in separate documents, and will also be posted in Canvas under the "Project" section.

The project will progress in several distinct stages. Some of these stages are enforced through grading deadlines; others are not. The deadlines are (*all of these deadlines will also be added to the calendar on Canvas*):

Date	**Project Component**
2/13/19	Form project group (1% of project grade)
2/27/19	Design Review (4% of project grade)
3/13/19	Phase #1 (15% of project grade)
4/3/19	Phase #2 (30% of project grade)
N/A	Phase #2.1
N/A	Phase #2.2
4/17/19	Phase #2.3 Caching (10% of project grade)
5/1/19	Phase #3 (30% of project grade)
5/5/19	Final Project Report (10% of project grade)

Each stage of the design makes the processor progressively more complicated. For your own benefit, it is strongly recommended that you not proceed to a new stage before you are confident the current stage is working to specification. Debugging errors in a complex design is much harder. It is almost always better to test smaller, simpler components first.

Many of the Verilog problems in the homework assignments were designed to be compatible with the project. Please feel free to reuse these modules (of course, fixing any errors first!). Similarly, we have provided solutions for those homework assignments. You are welcome to use the provided files, as long as you use a comment to explain that you are using the provided file. In

general, you will learn more from fixing your own files and making them work though, so we encourage you to do that wherever possible.

In addition to the previous homework problems, you will be provided with several reusable modules that you can use in your design. Most of these are Verilog implementations of memory system components. Please note that these files do *not* follow the CS/ECE 552 Verilog restrictions, so don't include them when you run Vcheck. Download the project tar file or use https://classroom.github.com/a/d99g-WJx for a complete collection of the files you will need for all stages.

An assembler for the WISC-SP19 ISA is provided for your use. Sample test programs are also provided, although you are strongly encouraged to write custom tests to augment these. Be aware that these test programs were written for a slightly different ISA specification and therefore may not work as advertised. It will be your job as diligent designers to determine if unexpected behavior occurs due to a bug in your design or as the result of the change in ISA. See Canvas for a description of how to use the assembler, as well as

2. IMPORTANT NOTES

- Start early: This project is designed to take a considerable amount of time.
- Plan ahead: You may find that the instructor, TAs, and peer mentors will be very inaccessible the night before a deadline.
- Ask questions: If you are getting stuck on some problem ask for help. Ask me, the TAs, or your classmates.
- Functionality: Getting a working design is of paramount importance. Optimizations, clock-speek and bonus questions come 2nd. First make sure your design works!

If you finish really early, you will get the opportunity to earn extra credit by adding extra features to your processor, synthesizing your design, or possibly mapping your design (or a part of it) to a FPGA chip. Additional details will be posted about this after Phase 3 is released.

If you are able to complete this project without the unnecessary stress that procrastination imposes, it is our belief that you will find this to be a highly rewarding experience.

All of the files you will need are included in the project tar file on Canvas and on Github Classroom here: https://classroom.github.com/a/d99g-WJx (Note that all subsequent homework and project components will be pushed to this one repo, no need to clone each of them separately!).