ECSE 425 Projects Overview

Winter 2021

1 Deliverables

This semester, the final computer architecture project is to design, implement, test, and optimize a pipelined processor. The project will be completed in two phases: (a) an introductory phase, completed in pairs, and (b) a group phase, completed in groups of four. The purpose of the introductory phase is to reintroduce VHDL and build basic components using principles that will be expanded upon in the group phase.

The project is divided into a number of deliverables. The first two correspond to the work in pairs; the final project and bonus task will be completed in groups of four.

1. Finite-State-Machine **(Due February 1, 2021)**
2. Cache **(Due February 18, 2021)**
3. Final Project: Pipelined processor **(Due April 9, 2021)**
4. Bonus Task: Optimized processor **(Due April 9, 2021)**

All students in a given group will receive the same grade for written deliverable reports and final testing. However, an effort will be made by instructors to identify, and appropriately penalize, any students who fail to adequately participate in the project.

2 Project Overview

Phase 1

In the first phase, you will complete a series of exercises designed to get up to speed with VHDL. The first deliverable requires that you design a finite state machine for identifying comments in source code. In the second, you will design a write-back cache.

Phase 2

In the second phase, you will begin by implementing a pipelined processor (five stages) based on a subset of the MIPS ISA. In order to test your processor, you will also need to write

assembly for conversion into machine code (we will provide an assembler). Machine code in our case will be ASCII text used to populate a model of main memory in VHDL.

If you have time, you can also attempt the bonus task, processor optimization. Previously, caching and branch prediction have been possibilities. More details regarding the final project will be given in late-February.

3 Grading

The three projects account for 40% of your course grade, within this 40% course grade, the break-down of each project is shown below:

1. Finite-State-Machine (10%)
2. Cache (25%)
3. Final Project: Pipelined Processor (65%)

* These weights are subject to change.

Deliverables will be evaluated as is. To receive full credit, please provide detailed instructions and ensure that your VHDL is packaged to readily facilitate our evaluation of it using the given testbench infrastructure.

The grading of (1-2) above will be based on a combination of the correctness of implementation, the completeness of submitted testbench and the passing of tests based on the instructional staff's testbench.

The grading of (3) will be limited to functional testing using a combination of assembly language codes provided to and withheld from student groups. If all test codes execute properly, full marks will be given.

You are expected to write a short report (IEEE format) summarizing your implementation for all the projects.

4 Preparation and Resources

4.1 Forming groups

You can use the Discussions section on MyCourses to find partners and register your groups in the Groups section on MyCourses. Since ModelSim does not work on MacOS, we suggest that you make sure at least one person in the group can access the software on his/her personal device (just in case if remote access to Trottier lab computers does not work properly).

4.2 Remote access to Trottier lab computers

The software we will be using for the projects is called ModelSim and it cannot be run on MacOS. If you cannot access ModelSim on your personal device, you may consider remote accessing the Trottier lab computers, which have ModelSim installed and ready for use.

To learn how to remote access the lab computers, please refer to https://www.mcgill.ca/ece/department/it-and-technical-services/it-faqs under the topic **Remote Desktop to Trottier lab computers**.

McGill VPN Download:
https://mcgill.service-now.com/itportal?id=kb_article&sysparm_article=KB0010687

Microsoft Remote Desktop Software:
If you are using an operating system different from Windows (macOS, linux), you must first download a compatible RDP client for your operating system.
https://www.microsoft.com/en-ca/p/microsoft-remote-desktop/9wzdncrfj3ps?activetab=pivot:overviewtab

4.3 Download ModelSim

If you wish to work on your personal device, you can download ModelSim from the link:
https://fpgasoftware.intel.com/?edition=lite&platform=windows

Make sure you download the 'Lite' edition (this is the free edition). Once you click the download button, you will be asked to create an Intel account, create an account and follow the prompts to download the software.