CS 320 Project 2: Pipelined MIPS CPU

Professor Joshua Stough

Team USB

Team Members:

Unrein Peter
Strizower Elias
Bhutwala Yash

Date of Submission:

September 29, 2016

Table of Contents

1. Project Executive Summary

The main goal of this project is to build a pipelined MIPS CPU that includes forwarding, and can handle hazards and exceptions. The aim of the team is to improve Verilog skills and gain a better understanding of the detailed workings of a pipelined CPU, while developing in a group environment.

The organization of the team is as follows: Peter is the Git/source code manager, Elias is the product owner, and Yash is the project manager. The details of each of the roles are specified further in the Project Roles section. The project structure is as follows: a directory for all the modules, a directory for the test files, a directory for the documents, as well as a directory for other files such as C or assembly files as needed. The overall timeline of the project is to finish activity 6, which is building a Single Cycle MIPS CPU, and then sequentially add functionalities of handling forwarding, hazards, and exceptions. Further details of the schedule can be found in the Project Schedule section. The team will communicate using slack and the appointments will be set using Google Calendar.

2. Project Roles

Role	**Assigned to**	**Responsibilities**
Project Manager	Yash	● Schedule meetings ● Keep everyone on task, maintain communication within team
Source code manager	Peter	● Maintain organization of code ● Resolve merge conflicts in Git
Product owner	Elias	● Address any problems/questions team has for the professor ● Maintain proper formatting of documents
Developer	Yash, Peter, Elias	● Develop, modify, and test modules ● Follow proper Verilog coding conventions

3. Project Schedule

The project was assigned Thursday, September 22nd and is due Thursday, October 20th. There are a total of four weeks allocated to finishing the project, and each week during lab we will report our current status. Below is a table showing the stages of our project and the dates by which they will be completed:

Milestones	Description	Milestone Criteria	Planned Date
M0	Activity 6: Single cycle CPU	Activity 6 Completed	09-22
	Complete a working version of activity 6	Individual modules perform desired tasks, are tested, well commented, and merged	09-29
M1	Add forwarding	Forwarding Integrated	09-29
	Complete a working version of the MIPS CPU with forwarding	Individual modules perform desired tasks, are tested, well commented, and merged	10-06
M2	Add Hazard Detection and Exceptions	Hazard Detection and Exceptions Integrated	10-06
	Successfully integrate hazard detection and exceptions into the previous week's version.	Individual modules perform desired tasks, are tested, well commented, and merged	10-13
M3	Write program 3 and perform testing	Program 3 Written and Tested	10-13
	Completely write the third program and test it against the simulator	Simulator run our program 3 and generates desired output, is tested, well commented, and merged	10-20

4. Project Testing Plan

The plan is to frequently test the main file as we add modules and functionality. We plan on developing and testing each module in isolation. For example, we would build a file for an ALU called alu.v and a test bench for just that module called alu_tb.v. The test benches will be kept in their own folder called “testing.”

Each file will be tested in isolation and then added to the main file where we will test the CPU as a whole to make sure that the new module doesn’t cause problems and provides all the capabilities it should. The largest module which will be created is the hazard unit. As other modules are developed, the hazard unit will grow. This makes the second round of testing, where the new module is added to the main file, all the more important. It is in the second test that any new functionality required from the hazard module will be tested.