**Project Plan for Team “Better Than Your CPU”**

Project 2

CSCI 320

Jiayu Huang, Yuxuan Huang, Kenny Rader

**Contents:**

* Organization
* Team Roles
* Schedule
* Testing Plan
* Team Contract

**Organization:**

Our work for this project will be organized through Git version control. The repository for our project is located at <https://gitlab.bucknell.edu/klr020/CSCI_320_Project_2.git>. Each of the team members will develop functional and/or test modules on separate branches and will push these branches together when their respective feature works. Before beginning any work on a given day, all team members should pull the most recent iteration of the project to their local master branch, and then branch off into their feature branch to work on their module. This will allow us to work efficiently on separate modules within the project individually, as well as to share those modules with the rest of the team. In the event of a merge conflict, an emergency team meeting (whether in person or digitally through the team’s GroupMe) will need to be held to resolve the conflict before anything else can be pushed to the repository.

The project files for our system will be located in a folder called **modules** and any and all documentation related to our project will be kept in a folder called **docs**. There will also be a file in the root of the repository called **readme.md** that will reiterate much of what is written here, with a bit more detail on how to compile/run the project.

**Team Roles:**

All members of the team will work together as developers. Responsibilities of this title include, but are not limited to, writing new modules, modifying existing modules, writing test modules for the system, and actually testing the system by running those modules. Since this is a small team, these responsibilities will be shared equally among all team members.

All team members will also contribute to writing documentation required for the project. Any external documents that are needed to clearly explain the different aspects of the system will be co-written equally by all team members.

These core roles are the foundation of our team, but other, specialized roles may arise throughout the course of the project. For example, if one team member stands out as a particularly good tester, it might make sense for them to take on more responsibilities in that realm. Likewise, if one team member excels at writing documentation, than a larger portion of the whitepaper writing might be given to them, while the module and test coding is redistributed among the remaining members.

**Schedule:**

The following is a tentative schedule of development targets for our project:

|  |  |
| --- | --- |
| Date | Goal |
| Thurs. 9/29/16 | Finish Team Contract, Work Plan, and Activity 6 |
| Thurs. 10/6/16 | Implement Hazard Unit, Preliminary Testing of Hazard Unit |
| Wed. 10/12/16 | Implement Syscall Module, Pipeline registers |
| Thurs. 10/13/16 | Implement Required Programs, Debugging |
| Sun. 10/16/16 | Finalizing & Testing, |
| Wed. 10/19/16 | Prepare the Presentation |
| Thurs. 10/20/16 | Presentation for our final design |

**Testing Plan:**

We plan to test every module in our system and ensure that it works to specification before proceeding onto the next item on the schedule. This will allow us to build upon the foundation of an earlier form of our system without worrying about potentially catastrophic bugs that may pop up at the very end.

In testing our modules, we will write a test bench for each module that will test each of its individual outputs and verify that their outputs are correct. These testbenches will exist as modules within the files of the module that they test, for the sake of clarity. They can all be individually run or run as a whole once the system is complete.

**Team Contract**

Team Name: Better than Your CPU

Date: 9-22-16

|  |
| --- |
| GOALS: What are our team goals for this project?  What do we want to accomplish? What skills do we want to develop or refine? |
| * To complete the projects in full, with all functionality working correctly * To better our Verilog skills and our understanding of the MIPS CPU architecture * To improve our team skills |
| EXPECTATIONS: What do we expect of one another in regard to attendance at meetings, participation, frequency of communication, the quality of work, etc.? |
| * Every team member is expected to attend all team meetings unless there is a special condition * Every team member should participate and contribute equally * Communicate as frequent as necessary * Code comments is required for each module. |
| POLICIES & PROCEDURES: What rules can we agree on to help us meet our goals and expectations? |
| * Attend every team meeting and on time * Frequent Code Reviews will help us find out any bugs before they get too problematic * Test your own modules to make sure they work properly before integrating them into the final system |
| CONSEQUENCES: How will we address non-performance in regard to these goals, expectations, policies and procedures? |
| * Talk to underperforming team members first * Try to resolve conflicts within the group. You will have up to three strikes, then we will have to involve the professor. * In the event of a communication failure, the following seem like appropriate measures:   + Including but not limited to:     - Doing/Redoing the work     - ⅓ Grade loss in this project |

We share these goals and expectations, and agree to these policies, procedures, and consequences.

Team member name:Yuxuan Huang

Team member name: Jiayu Huang

Team member name: Kenny Rader