CprE 381, Computer Organization and Assembly Level Programming

Team Contract – Project Part 1

Project Teams Group: ProjectGroup1-1

Team Members: Dylan Blattner

Caleb DeBoef

Course Goals: *List and acknowledge the goals of your individual team members.*

- Learn how a single-cycle processor works
- Be able to explain how a stored-program computer works from gates to Assembly to C
- Minimize points lost on the project
- Gain a better understand on the underlying workings of C code
- Stay ahead of content enough to not be required to rush through anything

Team Expectations:

- **Conduct:** *What are the expectations for personal conduct of group members?*

- Respect each others time
- Put in the required work

- **Communication:** *What is the best mode of communication for the group? How often should communication occur? How fast should a response be expected?*

- Overall communication through snap chat, discord, and texting
- Group members should check in at least 1 times outside of weekly lab
- Responses should be expected within 24 hours

- **Group conventions:** *Naming conventions? Compilation and simulation methodology? Testbench strategies? Do file usage? Version control strategies? Commenting standards?*

- camelCase for naming files
- Each test bench will start with tb_moduleUnderTest
- Creating of do files that will follow moduleUnderTest.do
- Git will handle version control
- Comments on at least each block of code explaining functionality, more as required

- **Meetings:**

- Additional meeting times:
    - T: 9am
    - M/W: Before lecture (8am)
- Meetings can be conducted through Discord/Microsoft teams, or in person as needed

- **Peer Evaluation Criteria:**

- Time spent working
    - Track using Git commit messages
- Personal deadline being met
- Quality of code and comments
- Communication and transparency

Role Responsibilities: *Complete the following planning table. Each lab part should be the responsibility of one team member. Also make sure that no one team member is the lead on both the design and test aspects of a single lab part. These guidelines aid in all students having a complete view of the lab. Note that the non-lead is encouraged to participate and support the lead wherever possible, increasing both the quality of the lab part and each team member's knowledge.*

Lab Part	Estimated Time	Design		Test	
		Lead	Timeline	Lead	Timeline
High-level design	1 hr	DB	10/13	CD	10/14
Test programs	4 hr	CD	10/17	DB	10/18
Control logic	2 hr	DB	10/20	CD	10/21
Fetch logic	3 hr	DB	10/20	CD	10/21
Barrel shifter	2 hr	CD	10/20	DB	10/21
ALU integration + Misc updates	2 hr	CD	10/20	DB	10/21
High-level integration	4 hr	CD	10/24	DB	10/25
Synthesis (human effort)	1.5 hr	DB	10/27	CD	10/27

*Estimated Time is given as a **very rough** guide for even distribution of tasks assuming you've already read through the lab document and have the prerequisite knowledge. Depending on your group's skill and prerequisite knowledge, some tasks may take disproportionately long or short. For your future planning, track this – for future prelabs you will be asked to note why past tasks took longer than expected and how you might avoid such issues in the future.*

Integrity of Work: *Do not delete the following.* We agree that the work we provide to other team members and ultimately submit for a grade is a direct result of our own work as described in the course syllabus. Specifically, we will generate all VHDL code ourselves and not copy VHDL code from online sources, other groups, book companion material, or past student projects to which anyone outside of my team has contributed.

Student Signature: Dylan Blattner **Date:** 10/6/22

Student Signature: Caleb DeBoef **Date:** 10/6/22