Ian Barthel Work Log

Milestone 1 Work:

Sunday, April 11, 2021
Met with team [1 hour 15 minutes]

We started with determining how each instruction would run, along with what extra registers we may need to fully implement the basic processes, along with what would be needed to run Euclid's algorithm.

Monday, April 12, 2021
Met with team [1hr 15 minutes]

We finished writing the code fragments and Euclid's algorithm in our assembly code format and finished the rest of the design document. We also assigned opcodes to all our initial 16 instructions we have so far. We are waiting on feedback from our meeting Wednesday before deciding how to move on with the project.

Milestone 2 Work:

Monday, April 19, 2021

Met with Jordan [1 hr].

We started and finished the RTL table, while I also decided to throw all of the opcodes and syntax into one table so that it was easier to read.

Tuesday, April 20, 2021

Met with team [1 hr 30 min]

We officially decided to make the opcodes different lengths depending on the instruction. In order to differentiate, J commands start with 1 while any other instruction starts with 0. This was done because we decided it would be nice to use the extra bits in order to further extend the jump range of J type instructions. We are currently planning for some sort of mux or control to read only the first bit of instruction and know exactly which component to send the rest of the instruction to.

Wednesday, April 21, 2021

Met with team [2 hr]

During this class time, with the help of the instructor, we finished and made sure that the RTL table was correct. We also put in the machine code fragments for every fragment in the design document, along with many visual touchups, such as combining tables.

Milestone 3 Work:

Monday, April 26, 2021

Worked on Lab 7 alone [1 hr]

I started work on lab 7, so that Jordan and I could both be memory “experts” for the project, but I mistakenly shut down my laptop when I was going to another room, and I lost the lab after being about 75% done. I never went back and finished as Jordan seemed to fully understand everything anyway.

Tuesday, April 27, 2021

Met with team [1 hr 15 min]

We briefly met with the team to discuss everything we must do for the milestone and decided that Logan and I will make the register files (he made the cool register & I made the other register file). We also decided that we will get a head start on the code and make Verilog tests for both components we made.

Wednesday, April 28, 2021

Met in Classroom [2 hr] Worked Alone or with Logan [1 hr 30 min]

In class, I finished work on the register file. One big issue I had was with ISE Design crashing and corrupting all my files. I had the symbol still saved, but the component itself was gone. I tried moving files to new locations, renaming them, and much more but it was not until Prof Williamson told me I was overlooking the very simple solution of just readding the schematic as a source document. After class, I created tests for the register file. The tests I made tested things such as reading and writing data into a register, testing if the enabler actually functions, etc.

Milestone 4 Work:

Sunday, May 2, 2021

Worked alone [1 hr 30 min]

I made the complete components for the IR and fixed any known flaws on the register file.

Tuesday, May 4, 2021

I made the ALU using Verilog code, and turned it into a schematic. At this point, I went through and tried to find any bugs in the code, and also spent a decent amount of time (over an hour) trying to help Hunter find bugs within the CRFile, as his tests were not functioning properly. Once we find out how to make all the components function flawlessly, it should not be too difficult to put them together.

Milestone 5 Work:

Sunday, May 9, 2021

Worked Alone [2 hr]

When Hunter went to make the tests for the register file, we found out that we lost some components due to my earlier mistake a week or two ago. Since I could not find or recover the missing .sch files, I had to remake the entire register file from scratch. In doing so I realized I mistakenly had the register file take in two inputs, instead of just one. So, we would have had to redo the file eventually anyways. Our register file only needs one readRegister because we have an accumulator with a separate register file for all of our accumulators.

Monday, May 10, 2021

Worked alone [1 hr]

We decided to make a few changes to the IR, which include adding an opcode output instead of just splitting the instruction after it passes through the IR. Since we have opcodes of various lengths, we just output the opcode as 5 bits, regardless if it is a J-Type or R-Type, and the control will read the first bit to see if it should read the full 5, or just the 3 bit opcode.

Tuesday, May 11, 2021

Met with team [3 hr]

Worked Alone [1 hr]

We met as a team to discuss the milestone ahead and any plans we need to finish for the week. We also talked about how we should split up the datapath into different pieces. I then made the first subsystem and gave it to Hunter for testing

Wednesday, May 12, 2021

Worked in class [1 hr 30 min]

In class, I made the remaining subsystems and a lot of the datapath (I finished and connected everything except for the control, mem file, and PC). Hunter also helped me set up the datapath.

Milestone 6 Work:

Tuesday, May 18, 2021

Worked with Hunter [2 hr]

After fully completing the datapath (minus memory) last week, we decided to run every instruction we have, and they all passed. Now all that is left is testing and adding Jordan’s memory unit.

Wednesday, May 19, 2021

Met with team [1 hr 30 min]

In class, I worked with the team to further test and finish the datapath of the system, also tested PC jump instructions.

Thursday, May 20, 2021

Worked with Hunter [1 hr]

Worked alone [30 min]

I spent about half an hour finishing and adding the memory file to the datapath, and then made the tests alongside hunter in order to test our full datapath.