LOG for Ethan Townsend
Log begin 10-01-2023:

The current date is October first twenty twenty three. The project group met for a total time of 3 hours to discuss the implementation of the processor. The meeting started with debate over the chosen architecture. The original architecture was a load-store type that would model other load-store architecture in its implementation. The process of the group went as follows:
the professor will grade us more strictly if we do load-store, we need to do another type of processor. The options are an accumulator or memory to memory type, the stack type was not considered for concerns for the sanity of the group. For a time the idea of a hybrid memory to memory accumulator was discussed. This idea was scrapped in favor of only using memory to memory because memory to memory was faster/more efficient than the accumulator under most metrics.
The final decision for processor architecture was a memory to memory based system.

I was responsible for a large portion of the debate over what architecture to design. I also created the current opcode schema and list of commands that are currently possible with the architecture.
Our group plans to meet on Wednesday to discuss assignment of work for the next milestone.

Of important note for our current design position is that there are really only three types of instructions, an ‘arithmetic’ type called a, a ‘immediate arithmetic’ type ai, and an argumentless stack operation. There is also a one argument jump, but that is not particularly relevant right now. The stack currently only stores the values of return addresses and the only operations that access it as of now are the ‘push’ and ‘pop’ commands that also move the PC to the value that was on the stack. The current plan is to have a specific place in memory that keeps track of the value of the stack pointer.

Log Begin 10-3-2023:

I have split the stack access into two types: access that knows the PC and access that does not. I chose to do it in this manner because I am unsure what address we would hand the process if we were wanting to push the program counter, because as far as I know, it does not exist in memory.

Log Begin 10-11-2023

We met between the last two log entries, I was there, I forgot to write a log for the meeting.
We discussed and fully fleshed out the RTL. The decision was made to flip the position of s1 and s2 in the machine code so that all instructions load their address first, rather than loading the variable argument first (the one that can be either a register or an address). It also allows the addresses to be loaded A then B, where A is (in most cases) a register and the nature of B depends on the instruction.

In this meeting (the one today), we verified the RTL as it exists. For the next milestone, I am in charge of making quartus files and making sure they exist in Github. We are all going to work on writing components and making tests for them (we're probably going to shuffle so people aren't making tests for their own stuff, but that is not set in stone).

Log begin 10-12-2023

The current day is Thursday, October 12th 2023. We met to discuss formatting on the google doc. It was a violent and brutal struggle. After much struggling, we made negative progress (this is a good thing). We plan to never meet again (work individually on the next milestone).

Log begin 10-18-2023

Today is Wednesday. As of present time, the tests for the memory module and alu are not yet written. I am making this entry because the tests will not be complete before the time things are pulled from the github. In other news, for the next milestone I will be working on the integration of the instruction memory and the registers, specifically targeting the op code and OP regesters to make sure they hold the right values after the clock cycles.

Log begin 10-23-2023

Happy mol day. We met today to work on the implementation steps of the project. The tests for this were interesting to write. I wasn't able to get modelsim to find the register8 file, even though it was located next to the rest of the files. Regardless, I have full confidence that the tests will succeed. The only place I foresee a potential failure is within the register8 not chopping the bits off the memory output correctly. In other news, one of our group members friends spent 5 hours in the NAB "studying 230" with a girl (this is not the first time this has happened, no progress on CS topics is made during this time). There was a discussion about whether he should purchase a halloween basket (or whatever they are) for her. The group was split on what he should do until it was pointed out that this was not our money we were voting to spend. I will record updates here as they become available.

Log begin 10-24-2023

The implementation stage 2 is complete, at least to the spec I believe we need. The important thing is that the modules all seem to work properly.

Log begin 11-1-2023

I have created an assembler, as of now, it is not functioning properly, but the ideas are there. I intended to work on it today, however, I was sidetracked by other projects. I will be implementing the input and output module(?). It will be part of memory, the design doc reflects the allocation of the addresses, however, I will need to make sure the diagrams and memory specifications are updated after the implementation is complete. Progress is being made on the secret special feature. There have been no significant developments in the groupmember's friend's relationship.

Log begin 11-7-2023

The processor works in Minecraft. The assembler takes another step towards being functional. The output address has been changed to f010. The in-module io has been wrapped into a wrapper module.

Log begin 11-12-2023

We have a presentation. We have a final document. We have no hope. I re-wrote a good portion of the report doc to have better language. We practiced the presentation a total of twice, which is more than I've done for any other presentation I've ever given.

An update on the saga: He does not know if she likes him back, so he isn't doing anything. I knew this would happen, but I'm still saddened to see it.