2/16/23

- Finished execute and memory cycle
- Wrote entire load-store processor
- Revised design doc

2/15/23

- Finished fetch/decode cycle
    - Wires out of each cycle to intermediate registers

2/14/23

- Finished memory implementation for our processor
    - Modified memory lab implementation

2/11/23

- Bugfixing, fixing compile problems on tests
- Completed immgen tests

2/10/23

- Working on compile issues, multiple problems with compiler not working with modelsim

2/8/23

- Started alu tests
    - Tested +, -, for outputs of arithmetic results, zero?, and positive?

2/6/23

- Started memory lab
    - Issues getting test bench to run

2/2/23

- Started imm gen tests
- Started fixing reg file
    - Issues with arrays of buses

1/31/23

- Went over how to write tests and components with team
- Helped compile project
    - Fixed quartus errors
- Wrote control
    - ALU op defaults to “-” since unused

        - Others default to 0

1/30/23

- Created test bench folder
    - Separate test benches from components
- Started writing tests for immgen
    - Created skeleton for reg file

1/28/23

- Finished assembler
    - Needed to clarify some formatting issues
    - Added rules for formatting and syntax

1/25/23

- Added clocks to components
    - Need to have clocked registers for pipelines
- Listed controls for rtl
- Implemented muxes and regs
    - This process let me know that they have to be clocked

1/22/23

- Met with team to do rtl
    - Changed specifications of components for pipelining (esp. branch)
    - Completed stages 1-3 of datapath
    - Datapath design is still open for hazard controls
- TODO: implementations of a few components

1/18/23

- First revision for translating full files
    - Split into two entry points, one for single instructions and once for files in order to make debugging easier and abstract out common logic
    - Loop through file to get label addresses, loop again to replace labels with addresses
    - Added parameters for flexibility in case of multi-file instructions
- Met with team
    - Generated naming conventions - based off of common sense, but made it consistent
    - Generated component specifications

1/15/23

- Worked on creating RTL for m-type instructions
    - Two different rtl paths chosen, due to the number/similarity of instructions to each other. This allows us to reduce the size of our datapath, increasing the speed of our hardware
- Initial planning for labels on assembler
    - Used so we can put entire assembly programs and have it translated to machine code
- Fixed machine translation for relprime and gcd
    - Original label byte offsets were off

1/11/23

- Adjusted assembler based on changes to instruction set
    - New instruction types, modified codes for each one
- Adjusted memory map
    - Works with 16 bits instead of a lot more
- Translated euclid's algo and gcd into machine code

1/9/23

- Built assembler

1/8/23

- Met with team members to start milestone 1
- Chose load-store
    - Simple to implement, familiar instruction set
- Worked on building instruction set, instruction types, and registers