Ethan Liu, Work Log

M1

December 29, 2023 - January 3, 2024 [20 Hours]

Draft and test instruction set
Draft hardware resources available to the hardware
Draft calling convention
Draft psuedo-instructions
Optimize procedure calling
Create proof-of-concept circuit for program testing
Create assembler
Write single input implementation of relprime

January 8, 2024
[1 hour]
Remodel instructions to have fixed length opcode and different func codes. Create instruction types.

January 9, 2024
[2 hours]
Create and test some example operations
[15 minutes]
Revise processor to have input and output lines 16 bits wide. Add instructions for read and write
[15 minutes]
Plan and create memory map

M2

January 11, 2024
[30 minutes] Update instruction format table for greater visibility
[1 hour] Update overall document structure for pulling information easier

January 12, 2024
[2 hours] Cleaned up assembled instructions for better visibility, fixed some examples with errors.
[1 hour] Review and update verilog code to a more usable format

January 13, 2024
[3 hours] Update Veilog code into format usable for our design, create the logic component as proof-of-concept for writing verilog.

January 15, 2024
[2 hours] Clean up and fix RTL, plan & create components for the processor

M3

January 17, 2024
[30 minutes] Create a table of write operations by instruction category, used for setting up control later in the design phase.
[2 hours] Write a sample implementations of control module and partially tested.
[1 hour] Write components decoder, 3 bit mux and register.

January 20, 2024
[30 minutes] Compact the general purpose registers into a register file, update the design to use the register file instead of instantiating each general purpose register.
[2 hours] Create tempelates for implementing different functions within the ALU. In this design, the overall unit, somewhat inappropriately named "ALU Complex", will host the control, inputs and outputs to all smaller ALUs (which there are 17 by current count) with shared IO for maintainability and clarity at the datapath diagram level.
[1 hour] Test implement the memory. A notable issue is found where the data read/write address isn't immediately available at the rising clock edge, and either the memory needs to support dual-port, dual-clock operation, or these instructions will require two cycles for the two reads to occur.

January 21, 2024
[1 hour] Finish wiring processor at the top level
[30 minutes] Verify datapath correctness, and generate datapath diagram from the RTL viewer.
[1 hour] Fix single cycle RTL, notably removing reading from special registers into variable. Use of oldPC is retained for clarity purposes.

January 22, 2024
[1 hour] Create all wiring template for ALU parts, to be distributed to teammates
[1 hour] Hold team meeting to discuss work for M4

M4

[4 hours] Refactor datapath, lay foundation for reusability of ALU and similarity to datapath discussed in class. Though, the design choice of "putting the cart before the horse" has led to a rather strange configuration - one where the ALU is used sometimes before memory, and sometimes after, leading to a kind of double loop. Old design of ALU and pads are now deprecated.
[2 hours] Created a table showing what values should be routed for each instruction. This has not shown up in the design document due to a layout issue (table is wide), but we may decide to rotate the whole document horizontally to accomodate it.
[2 hours] Created base testbenches
[1 hour] Devised unit and integration testing plan. Minding the high complexity of the instruction and control flags - the tests can require more than 100 cases each with more than 10 parameters, I am opting to cut down as much partial integration as possible, and instead rely on unit testing and full processor testing to find errors, as assembly programs are far easier to construct.
[1 hour] Make modifications to the assembler for easy conversion from assembly to bytecode
[30 minutes] Updated RTL categories to differentiate the use of different special registers.
[2 hours] Team meetings

M5

[10 hours] Work on testbench of processor, fix wiring issues, performed modifications such as the routing of PC to work with reset, optimize clocking logic
[2 hours] Update assembler to print out memory content file for direct use

M6

[4 hours] Collect performance metric of processor, investigate max delay path
[4 hours] "Try to" implement processor on FPGA