Dalton Bumb, WORK LOG

MILESTONE 1 WORK:

Meeting 4/2/23 [1hr]
- Discussed architectures, narrowed it down to accumulator or stack and we chose stack
- Began design document, I started the table with the instructions, instruction types, and their descriptions
- Went through pseudocode examples of our stack architecture

Prior to Meeting 4/4/23 [3hr]
- Changed majority of instructions on green sheet.
- I misunderstood how our design worked. Before, push used a register and so did pop. Now I see that all the regs are on the stack and were only pushing memory addresses/immediates, and popping to memory addresses.
- Developed talking points for meeting

Meeting 4/4/23 [4hr]
- Confirmed two stack architecture
- Helped write code for Relprime
- I put the new procedure stack instructions on the green sheet and touched up the high-level design summary.

MILESTONE 2 WORK:

Meeting 4/18/23 [2.5hr]
- Discussed RTL, Group decided to move to three stack architecture. 1 for data, 1 for procedure arguments, and 1 for return addresses.
- Created a rough sketch of the datapath.

In class 4/19/23
- Worked on RTL

MILESTONE 3 WORK:

Meeting 4/23/23 [2hr]
- Meeting to design datapath
- The datapath with a stack architecture was interesting. Rather than having a single register file that goes to an alu, our datapath has 3 stacks instead of a register file. Two of these three stacks interact with each other

Meeting 4/26/23 [2hr]
- Started implementing the stack component in Verilog
- Ran into various syntax errors, fixed quartus errors by referring to error messages

MILESTONE 4 WORK:

Prior to Meeting 5/1/23 [4hr]

- Worked on stack component. Originally used stack pointer for internal stack logic, but decided stack pointer could run into unexpected edge case issues down the road. Settled for the slower, but more reliable method of just cascading all other values in the stack when the stack has to be shifted.
- Tested stack component, originally didn't have clock signal on stack and stack was working. Decided that although the stack was working in the test bench without a clock signal, the stack would need a clock signal when put in the datapath.

Meeting 5/1/23 [2hr]

- The group decided on adding two new logic for the input to the pc register. With conditional jumps, the alu must perform an operation before the pc can decide what value it gets. This means that the mux to the pc must have a control signal not coming straight from the control unit, the control signal comes from the alu operation.