Christian Meinzen (meinzecp)
CM – 401

Milestone 1:
Friday 10/4/19

- Discussed for one hour at 4:00 – 5:00 PM
- Decided to make processor with Stack architecture
- Discussed possible calling conventions and addressing modes
- Created three ISA for pushing/popping, jumps/branches, and arithmetic/stack manipulation.

Sunday 10/6/19

- Met in O259 from 5:30 – 8:45 PM
- Most of the time was used to discuss the Stack architecture and make sure all team members are on the same page
- Will and I worked on framing some calling conventions and designed the assembly code for relPrime and GCD
- Used the idea that memory would store return addresses and argument values when making a procedure call
- Tori designed instruction set that we could use to get around our ISA and the 16-bit limitations.

Monday 10/7/19

- Met with Sid from 1:20 PM – 2:10 PM to discuss advice and clarification on addressing modes. Realize that using the stack for return addresses will be better than using memory.
- Met with group from 3:30 – 6:45 PM to redesign assembly code and instruction set.
- Removed ‘jal’ and added ‘dup’ and ‘flip’ instructions for better stack manipulation
- Rewrote code for relPrime and GCD in our new instruction set using polished calling conventions where the return address and arguments are placed within the stack, and the return value is placed in memory.
- Side notes:
    - It feels as though without pseudocode, our code will be complicated to efficiently program
    - While everyone has contributed in some way to the project, I feel as though I have created most of the assembly code. So, I am worried that my teammates might not understand how we are implementing our design through code.

Tuesday 10/8/19

- Met in F225 from 3:50 – (ins) to make final changes and commit Document for Milestone 1.
- Made documentation a bit neater by formatting and inserting a header with some footnotes
- Push personal journal and Document
- Discuss about how we are going to complete Milestone 2 so we can talk to Sid tomorrow and not sound like idiots
- I think that communication is key for this milestone because we are doing it during the break time.

Milestone 2:

Friday 10/11/19

- Self-conducted meeting at home.
- Polished up Milestone 1 using critiques that Sid gave at group meeting
- Worked on adding more common operations.
- Thought of a new method of procedure calling. Use a jump and link instruction that takes the PC and puts it on the stack.

Sunday 10/13/19

- Group met in F225 from 7 PM – 10:15 PM to fix remaining Milestone 1 issues and begin Milestone 2.
- Created RTL code for all instructions except duplicate and flip, which we are asking Sid about how to implement in hardware
- Make a better table of contents and organization of document
- Settle on pushM and pushR for more efficient coding. Also, memory allocation will be used by offsetting from the $sp register. Still need to have the $sp address on the execution stack
- This meeting went far better than the previous. It seems that everyone understood each change and new implementation.

Monday 10/14/19

- Group met in F225 from 4 PM- 6 PM to finish the RTL code and change some hardware components after talking with Sid.
- Created diagrams that show the inputs and outputs of each required piece of hardware
- Decided on what control signals might be needed for each hardware component
- Implemented a barrel shifter that will both shift right and shift left.
- Put all of the RTL code into a Table so that it is neater and, for the most part, split into a multicycle process. (Might have to change later when designing the datapath)
- Update the code for relPrime and GCD with our new instructions and calling procedures that changed throughout Milestone 2
- This meeting went well, but it seemed that there is still a bit of confusion within the group on duplicated and the execution stack hardware components. Resolved before submitting document.

Tuesday 10/15/19

- Tori and I met in F225 from 4-5 PM to polish document one more time
- Finished the RTL verification section by adding pictures of add command
- Change pushNum -> dupNum for better clarity
- Committed and pushed Milestone 2

Milestone 3:

Saturday 10/19/19

- Started to work on the components for milestone 3.
- Realized that I have no clue how to do Verilog, so I worked on tutorials instead
- This was solo time at Starbucks when I was at home

Monday 10/21/19

- Worked on my own time throughout the day.
- Completed components for Sign and Zero extenders as well as their tests

- Finished the register and register file, but their tests need some work
- I am a little worried that Tori and Will have not worked on M3 and focused on the labs instead.

Tuesday 10/22/19

- Met with Sid to discuss Lab 08 and some of the execution stack component
- Completed the tests for the register and register file. Tori said that she will work on the memory unit.
- I think I finished the component for the execution stack, but the unit test did not work as I wanted.

Wednesday 10/23/19

- Will said that he completed the datapath. Tori and I checked it and agreed that it looked good, but there might need to be a few changes depending on testing.
- Tori finished the Memory and started working on describing implementation
- I finished the execution stack and started figuring out the control unit and how everything fit together within the datapath.

Milestone 4:

Friday 10/25/19

- Rewriting some of the unit test explanations in the document for better understanding.
- Work by myself because it was hard to find a time today to get together with the group

Sunday 10/27/19

- Met with group in F225 for about two hours. Started to create the control unit with Will, while Tori was working on combining components together.
- Created a table to write down all of the control values for each instruction.
- Realized that we need to edit the RTL code a little bit to allow multiple execution stack operations

Monday 10/28/19

- In class, talked with Sid for better understanding how to tell the Execution Stack to do multiple operations in different cycles.
- Realize that we needed to update the ES component for an enable bit. Also got rid of the extra output registers because we never really used them.
- Found out that the group could not get together until Wednesday, in which I will not be on campus. Asked Sid for extension.

Tuesday 10/29/19

- Today, I wanted to update all of the RTL code so that it correlates to how we want the operations in our ES to execute and be called.
- After updating all of the RTL code, I wrote all of the Bubble-Control diagram for each instruction excluding branching and shifting (due to Will saying that we might be changing the slr and sll into one instruction).
- Updated the Verilog for the execution stack so that it will go along with our updated datapath/control system.
- Tori and Will said that they are able to finish M4 tomorrow.

Milestone 5:

Friday 11/1/2019

- Met alone and worked on completing the execution stack due to some bugs I found while thinking of component implementation.
- Wrote the test for execution stack so that it would incorporate my changes.

Monday 11/4/2019

- Worked in class with Tori and discussed what components to work on.
- Decided that she should work on more of the memory and I should focus on the execution stack.
- Will did not show up, but he said that he was working on a lot of the ALU stuff and shifters
- Realize that I am bad at creating tests – went to Sid for help during office hours
    - Gave me better insight on how to generate good tests
- Finished with the ES_and_ALU subsystem. Generated a test that looks at all of the different ALU functions. Could possibly be better to add more tests, but I feel it was sufficient to show correctness.

Tuesday 11/5/2019

- Started to work on the ES_and_pushVal subsystem. Realized that there was a problem with ex_stack thanks to Sid.
- Focused on fixing the ex_stack to adjust the register updating component.
    - I think there is still a problem…need to talk to Sid about the always conditionals for the stack.
- Finished the ES_and_pushVal subsystem later in Lab F217. This has not incorporated the problem mentioned above – need to check that it will not be affected later.
- Tori shown that she has worked on multiple parts of control and memory. Will says that he is working, but has not indicated via the design document
    - Need to talk to him about it tomorrow in class

Wednesday 11/6/2019

- In class:
    - Talked with Will, who said that he had a lot going on in other classes. He said that he was available today to go hard on the datapath.
    - Talked with Sid about the problem for conditionals. Inserted the clock as a conditional parameter, which seemed to do the trick.
        - This solved most of my problems with ex_stack and ES_and_pushVal
    - Started working on Mem_RegFile_ES subsystem. Talked with Tori about how the memory works with other components since she worked on memory the most.
- In F217:
    - Finished the Mem_RegFile_ES subsystem.
        - Understood how to generate own memory block using the code that Tori wrote.
        - I think that there might be an unnecessary delay when taking data from the execution stack to registers A and B.
    - Helped Will with the datapath implementation because he was getting connection errors, just as I once did with the execution stack subsystems
        - I do not think that we will have the datapath tested in time for the Milestone submission time.
        - Need to create a test for Friday so that we can talk to Sid about the full implementation.
    - Tori and I started to talk about I/O. Came up with questions to ask Sid:

- Make sure that I/O is in reserved memory. Make sure that relPrime code is also in memory
- Ask how I/O is to be interpreted and connected with hardware.
- Ask if it is alright to create a component that has logic mixed with the memory block (this might change the datapath a little bit)

Milestone 6:

Saturday 11/9/2019

- Tori and Will worked together on trying to get a series of instructions to work on the processor, I was unable to meet so I stayed in communication via Group Me and provided any debugging/tech support
- We realized that the clocks for registers and components need to be on different edges for faster processing
- After they pushed their changes to the repo, I looked at the new code. I realized that the ES needed a reset, certain multicycle signals needed to be changed in the diagram.
- Communicating in this way was very inefficient, but it was much better than postponing the work until the next day

Sunday 11/10/2019

- Met in F225 from 4:00 – 6:00 PM
- Since Will finished the assembler, he focused most of his time on updating the design document to see if there are any extra components that we don't need.
    - Found out that we could reduce the mux to ALUSrcB to one bit.
- Tori continued to work on writing code for the series of instructions. Once this is finished we can then implement our control.
- I spent this time helping Tori figure out some of the clocking issues and testing. Half way through, I switched over to rewriting code to align with the updates that Will found in our datapath.
- The testing and datapath seems to be complete. We are thinking that we can now implement control. After that, it is running our program followed by I/O.
    - I will be looking over our code for relPrime again tonight to make sure that it is efficient and updated to our datapath since it has been a while since it has been touched.

Monday and Tuesday – (I was dead in bed)

- Tori and Will said that they are putting together control and testing different programs using the assembler.
- I have rewritten some of the relPrime code to be faster/updated. I have also designed the I/O implementation via paper

Wednesday 11/13/2019

- Met in class and in F217 from 4:00 PM – 6:00PM
- Write the I/O implementation in Verilog. *Need to implement in datapath*
- Write recursive program to test stack pointer manipulation as well as procedure calling
- Update the memory layout to include I/O and get rid of the static data that is never used.