Project Status Report #5
David Gronlund
Group A3
3/23/19

This week I worked on finishing the CPU design, which was mostly focused on the decode module since that is where most of the complexity is. My logic really centered around the asynchronous updates from branches and writebacks that happened regardless of the normal flow control logic and were a lot of my debugging effort. To simulate the core I used a single dual-port memory that provided both the data and instruction memory, where I then loaded instructions into and let the simulation run in Vivado.

First I tried running assembly that I compiled with the RISC-V toolchain and it worked mostly on the processor but I kept running down bugs, especially with the instruction counter. After this I tried to run some C code I compiled for it, but I had more issues with trying to get the linker to target a single flat address space, but eventually got this working too. The C code would run and I was having some initial success with running a small printf, but I had some intermittent issues and realized I may have other CPU bugs.

To find these bugs I ran the RISC-V tests on the core, which was its own adventure in trying to get the tests, written in assembly, to build and then generate the proper success or failure flags from the CPU. I eventually got them running, and every single one failed (the 447 test cases are actually garbage, they never caught these bugs), which I then went and fixed.

With the RISC-V tests working I was back to running C code, which correctly printed to a file out of the simulation, moved the CPU into a halt/fault state with the EBREAK instruction, and with the Dhrystone benchmark running as well. This resulted in an IPC of 0.4, so the next goals for the CPU are to add forwarding and a very dumb branch predictor to improve this, as well as the infrastructure to run floating point instructions.

I also implemented the core on an FPGA and got a clock frequency of about 220 MHz, but with another pipeline stage out of the instruction and data memory a clock speed of 300 MHz is very achievable. I have yet to do this because I wanted to compare the IPC of a single versus double stage memory first before committing to that design for clock frequency reasons.