Checkpoint 1

Checkpoint 1 Progress Report:

Status:

We believe we have completed all of checkpoint 1 functionality, implementing a basic pipelined processor, supporting the instructions ADD, AND, NOT, LDR, STR, and BR. We do not handle any control hazards or data hazards in this checkpoint, as specified in the checkpoint documentation.

We implemented this pipeline hierarchy by writing separate modules for each of the pipeline stages, and also writing a single “buffer” module to use between each of the stages. This buffer has a unified input/output scheme to make wiring up easier for us. Input/outputs are simply filled into the next buffer as they are generated.

Our write-back stage is really just a few wires going back to the ID stage.

Testing:

We wrote test programs with several NOPs in between to individually test the instructions implemented this checkpoint. We kept building on this one program by adding new instructions as we tested them, so that we could not only verify existing functionality at every stage, but also check functionality of new instructions as we were adding/checking support of them.

Work Split:

Rohan worked on the IF and MEM stages of the pipeline
Tadas worked on the EX stage of the pipeline
Tanishq worked on the ID (control ROM generation logic) and WB stages of the pipeline

Checkpoint 2 Roadmap

Over break:
Tanishq and Tadas are going to implement the rest of the LC3b instructions
Rohan is going to implement basic Icache and Dcache, along with a basic arbiter

In the week back to school, together we will implement a basic multi-cycle L2 cache.