ECE411 Progress

PCB

Prateek Tenkale, Chris Shannon, Brendan Callas

Checkpoint 2

Who Worked on What

Chris: Worked on implementing the forwarding and hazard detection modules in the datapath, hooked up the MP3 cache to the main datapath, replacing magic mem

Brendan: Implemented the cache arbiter, aided in connecting the new cache/arbiter to our design from checkpoint 1 and modified caches from MP3

Prateek: Worked on testing new features and validating the design with additional test code, as well as branch prediction/squashing logic

Functionalities

For Checkpoint 2, we implemented the cache arbiter and forwarding/hazard designs that we finalized in the previous checkpoint, and hooked them up to our main datapath. We also switched out the magic memory provided for us in checkpoint 1 with our own cache from mp3, ensuring it would work in 1-cycle. We also fixed an edge case problem in our datapath for the sake of the critical path, and added a missing forwarding path to our design.

We also looked through the list of advanced features and, after discussing how we could implement them, decided on a few to add for the next checkpoint as well as the preliminary designs for those.

Testing Strategy

We used the provided mp4-cp2.s code to test the functionality of our design, using ModelSim to verify that the pipeline stages acted as expected, and provided the correct output. ModelSim was used for debugging in a similar way as well, by analyzing each stage to find errors. We also wrote our own code to test further instructions and instruction combinations, and to unit test features along the way as we implemented them. Finally, we plan to connect RVFI and Shadow Mem to further verify correctness.

Features for Next Checkpoint

For the next checkpoint, we plan to implement our advanced design features. For these design features, we plan to implement pipelined caches, L2 caches, and use BRAM for our caches. We also plan to implement the RISC-V M-extension, by adding a multiplier/divider to support those

instructions. Additionally, we plan to implement a basic hardware prefetcher and a branch predictor like a local branch history table.

Work Plan For Next Checkpoint

Chris: Pipelined L1 Caches and L2 Cache using BRAM

Brendan: RISC-V M-extension, prefetcher

Prateek: More advanced branch predictor, helping with other features if needed