Progress Report 2

Qingyi Wang (qwang83)

Mingyuan Han (mh24)

Shitao Liu (sl53)

We generally split the task for Checkpoint 2 into three parts: cache and arbiter (memory interaction part), branch predictor & stalling & bubble insertion, and forwarding logic part. Specifically, everyone is responsible of implementing his/her design as described in the last proposed roadmap: Shitao designed, implemented, and tested the robustness of the memory interaction part, including a D-Cache and a I-Cache, which are both L1 caches, and the arbiter. Qingyi designed, implemented, and tested the branch prediction, flushing logic, stalling logic and bubble insertion logic. She is also responsible of the testing of the whole design after completion. And, finally, Mingyuan is responsible for design, implement and testing of the forwarding logic part.

The delivery of our CP2 is a pipelined CPU with L1 caches, an arbiter, a hazard detection and forwarding logic, and a static branch predictor. The arbiter was implemented as a finite state machine with seven states, which can be further simplified into three states. The forwarding logic is implemented by two 4 to 1 mux and some logic to determine whether we should carry out a forwarding. The branch predictor is a static-not-taken branch with the misprediction flushing logic. The stalling logic is to deal with the case when we need to wait for the memory resp signal. The bubble insertion is asserted when we have a lw + arith operation with data hazard.

We relied on both the provided test cases and the autograder to examine our design. To test the L1 caches and the arbiter, Shitao replaced the magic memory for CP1 with the ordinary param memory, connected the L1 caches and the arbiter to the CPU, and reran the CP1 test to check if the CPU works properly. Mingyuan manually creates codes with data hazard to test if the forwarding par is working as designed. Qingyi finishes the monitor files in hvl folder, integrates all parts together to debug and checks the waveform to ensure the correctness of the design.

