**Progress Report**

The main point of checkpoint 2 is competing basic RV32I pipelined datapath along with implementing L1 caches specifically for separate purposes, one for data and one for instruction using an arbiter and a connection to shadow memory. L1 cache system includes two different cache types which may sound not too bad, but we had to consider that I cache doesn’t have to include ‘write’ functionality since we only read instructions inside, unlike data cache should deal with both read and write functionality. Arbiter works for choosing either of them including edge cases like when we need to access for both I and D caches and choosing which one we should access first for such process. Basic RV32I pipelined datapath was not something we felt too difficult since we already have planned how it should be done even before the first checkpoint due date. Splitting L1 caches after some discussion time we had wasn’t a big issue. After having a solid concepts behind these features, except the debugging part everything went pretty smoothly as expected.

**Roadmap (Modified from the last progress report)**

\* Marked with blue color: Modification made from the roadmap of last progress report

03/31: Discussion on L1 cache splitting with arbiter implementation

04/01: Discussion continued / TA meeting

04/02: TA meeting

04/03: L1 cache implementation / small debugging issue / basic RV32I pipelined datapath is completed

04/04: Arbiter codes roughly drafted / L1 cache almost done

04/05: Debugging with arbiter / L1 cache done

04/06: Last minute debugging

04/07: Studying for midterm II

04/08: Studying for midterm II