

## Progress Report:

For checkpoint 3, we implemented advanced features to improve the performance of our machine. We have implemented a local history branch predictor, BTB, hardware prefetching, a parameterized, 8-way L2 cache, and a 4-way fully associative eviction buffer.

### Below is the work distribution:

- Kelsey: Parameterized 8-way L2 cache, 4-way fully associative eviction buffer
- Murat: Hardware prefetching with stream buffer
- Xinpei: Local history branch predictor and BTB

### Testing Strategy:

Our strategy was to test individual components separately before combining and testing with the given MP4 code, using the given final register values for comparison, as seen below. For the caches, we used existing MP3 testing resources, for which we wrote tests to ensure correctness in both data and behavior, including eviction.

Register	Value
evb_tot[31:0]	4fc
evb_write[31:0]	16e -> 16f
l2_miss[31:0]	21f
l2_tot[31:0]	23f3

For the EVB, we counted the number of times it intercepted a writeback from the L2 and got 28.762% of all accesses.

For the L2 cache, our miss rate was 5.9%.

For the predictor, we implemented two counters that count the number of total branch instructions and the number of mispredictions by our predictor.

Unfortunately, we have not gotten the prefetching to work, but below are the statistics we've gathered so far:

Register	Value
l1_tot[31:0]	22c
l1_hit[31:0]	4d

In the screenshot above, l1\_tot stands for the number of times the L1 data cache would have missed without the prefetcher (556), and l1\_hit stands for the number of times these misses were reduced (77). Thus, the prefetcher reduced misses on the L1 data

cache by ~14% so far. The prefetcher was not merged with the rest of the code, and can be found in the “prefetching\_2” branch.

### **Roadmap:**

Checkpoint 4 is the competition checkpoint, where our scores will be based on the performance of our machines in terms of energy, delay, and fmax. We will first establish the baseline for our current design with the competition codes. Based on that information, we will likely keep our current advanced features but modify them as needed to improve performance over the baseline based on our performance counters and the synthesis data.

Since we are working on the same or more advanced versions of the features we already implemented for CP3, the work breakdown will be as follows:

- Kelsey: L2+ cache system, eviction buffer
- Murat: Hardware prefetching
- Xinpei: Tournament branch predictor