Project 1 Report

Tristan Barber

Tridyo43@knights.ucf.edu

EEL 4768: Computer Architecture

Due Date: March 26th, 2023

Part A – Analyzing miss ratio with a variable cache size.

For this portion, associativity is set at 4, write-back is used, and the replacement policy is LRU. Cache size varies, and the results between MiniFE and XSBench are compared.

Cache Size	MiniFE	XSBench
8KB	Miss Ratio: 0.080739	Miss Ratio: 0.136261
16KB	Miss Ratio: 0.074205	Miss Ratio: 0.119363
32KB	Miss Ratio: 0.065607	Miss Ratio: 0.112539
64KB	Miss Ratio: 0.059663	Miss Ratio: 0.108198
128KB	Miss Ratio: 0.055835	Miss Ratio: 0.104894

As shown above, increasing cache size will directly decrease the overall miss ratio. This is intuitive, but also follows the law of diminishing returns. With every double in cache size, the miss ratio decreased by a factor less than its previous decrease. It should also be noted that the overall miss ratio of the XSBench trace is much higher, because that trace is much larger in size.

Part B – Analyzing how Write Policy affects total reads and writes with variable cache size.

For this portion, the same configuration from above is used, but this time write policy is changed between write-back and write-through.

Cache Size	MiniFE (wb)	MiniFE (wt)	XSBench (wb)	XSBench (wt)
8KB	Reads: 393526 Writes: 84702	Reads: 393526 Writes: 636483	Reads: 2885331 Writes: 55683	Reads: 2885331 Writes: 5013495
16KB	Reads: 361679 Writes: 77541	Reads: 361679 Writes: 636483	Reads: 2527505 Writes: 7318	Reads: 2527505 Writes: 5013495
32KB	Reads: 319773 Writes: 72064	Reads: 319773 Writes: 636483	Reads: 2383001 Writes: 600	Reads: 2383001 Writes: 5013495
64KB	Reads: 290799 Writes: 67124	Reads: 290799 Writes: 636483	Reads: 2291098 Writes: 69	Reads: 2291098 Writes: 5013495
128KB	Reads: 272145 Writes: 61239	Reads: 272145 Writes: 636483	Reads: 2221125 Writes: 36	Reads: 2221125 Writes: 5013495

The first trend shown above is that, when using write-through, the number of writes to memory is not affected by the size of the cache. This is extremely inefficient. On the other hand, when using write-back, increasing cache size decreases the number of writes to memory. This is also true for reads to memory, for wb or wt. The XSBench trace also shows that larger memory traces will benefit more from write-back, compared to smaller memory traces like MiniFE. Another reason for the stark difference between XSBench and MiniFE is that XSBench shares a significantly higher percentage of memory traces, compared to MiniFE, which has more unique entries.

Part C – Analyzing miss ratio with variable associativity.

For this portion, cache size is set at 32K, write-back is used, and the replacement policy is LRU. Associativity varies, and the results between MiniFE and XSBench are compared.

Associativity	MiniFE	XSBench
2	Miss Ratio: 0.066107	Miss Ratio: 0.114457
4	Miss Ratio: 0.065607	Miss Ratio: 0.112539
8	Miss Ratio: 0.065305	Miss Ratio: 0.112008
16	Miss Ratio: 0.065201	Miss Ratio: 0.111906
32	Miss Ratio: 0.065118	Miss Ratio: 0.111784
64	Miss Ratio: 0.065075	Miss Ratio: 0.111753

The results shown above are explained in a very similar way to part A. Increasing associativity in this situation decreases the miss ratio, with diminishing returns. The overall miss ratio of the XSBench trace is higher because that trace is larger. It should be noted that having an LRU replacement policy is important with an associativity increase. Larger associativity means more traversal time at each cache block, and it is important to minimize that traversal time by using LRU.

Part D – Analyzing how replacement policy affects miss ratio, with variable associativity.

For this portion, associativity is set at 4, and the replacement policy is FIFO. Cache size is set at 32K, and associativity is modulated, with write-back and FIFO being used. The results between MiniFE and XSBench are compared.

Associativity	MiniFE	XSBench
2	Miss Ratio: 0.068036	Miss Ratio: 0.120610
4	Miss Ratio: 0.067756	Miss Ratio: 0.120672
8	Miss Ratio: 0.067623	Miss Ratio: 0.120891
16	Miss Ratio: 0.068072	Miss Ratio: 0.121221
32	Miss Ratio: 0.068254	Miss Ratio: 0.121125
64	Miss Ratio: 0.068553	Miss Ratio: 0.121212

The results shown above are not intuitive. Increasing associativity beyond a certain point can actually become detrimental, when using FIFO. This is more apparent with larger memory traces, like XSBench. MiniFE starts off with a decreasing miss ratio, that eventually begins to trend upward. XSBench, however, immediately begins to suffer when associativity is increased. This is not to say that associativity increase is bad, it proves the point that an LRU style replacement policy is hugely important. This can be even further proven when comparing these results to the results found in part C.