Contents

References

[1] The Sniper Multi-Core Simulator

[2] O. Tange (2011): GNU Parallel - The Command-Line Power Tool

[3] S. C. Woo, M. Ohara, E. Torrie, J. P. Singh and A. Gupta, The SPLASH-2 Programs: Characterizaion and Methodological Considerations, Proceedings 22nd Annual International Symposium on Computer Architecture, Santa Margherita Ligure, Italy, 1995, pp. 24-36

[4] Bailey DH, Barszcz E, Barton JT, et al. The Nas Parallel Benchmarks. The International Journal of Supercomputing Applications. 1991;5(3):63-73. doi:10.1177/109434209100500306

[5] John L. Hennessy and David A. Patterson. 2017. Computer Architecture, Sixth Edition: A Quantitative Approach (6th. ed.). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

[6] S. M. Londono and J. P. de Gyvez, "Extending Amdahl's law for energy-efficiency," 2010 International Conference on Energy Aware Computing, Cairo, Egypt, 2010, pp. 1-4, doi: 10.1109/ICEAC.2010.5702300.

[7] Bui, Amy. EE156 Lab0 Report. 2023.

Benchmark	L3 Associativity	L3 Eviction Policy
splash2-ocean.cont	8	LRU
splash2-radix	16	SRRIP
npb-is	32	Round Robin

Table 1: Configuration parameters and values swept in the experiment.

1 Intro

2 Experimental Setup

Simulations ran for an x86 architecture simulator, Sniper 7.3 [1]. Each were configured with 4 cores and the same L1, L2, and L3 cache sizes (64 KB, 128 KB, and 512 KB, respectively, each using 64 byte blocks); remaining relevant configuration values were set in the `gainestown.cfg`. Input size used for all tests was preset `large`. Figure **??** visualizes the topologies for all simulations since cache sizes remained constant.

Three different L3 cache associativities, three different L3 replacement policies, and three benchmarks were swept in this experiment, for a total of 27 simulations (see Table 1). L3 is the largest and slowest cached memory unit compared to L1 and L2; it is also shared across the four cores, while each core has their own L1 and L2 caches. Therefore, we picked the lowest performing cache level used by the processing unit to sweep because it is likely to have common impact on performance and energy that is observable on all four cores. The different configurations were simulated with two `splash2` benchmarks (`ocean.cont` and `radix` [3]) and one NAS parallel benchmark (`npb`) (`is` [4]). The workloads are briefly described as follow:

splash2-ocean.cont : The `ocean` suite of test studies large-scale ocean movements based on currents, and uses 4D array grids and a red-black Gauss-Seidel multigrid equation solver.

splash2-radix : The `radix` suite uses an iterative radix sort algorithm that generates histograms and has each processor permute array index keys, a process that depends on processors communicating in order to determine keys thorough writes.

npb-is : The NASA Advanced Supercomputing (NAS) Parallel Benchmarks (NPB) are a set of benchmarks tuned for highly parallel workloads. The `is` kernal performs a sorting operation that is important as "particle method" code (ex. simulations of mechanics (solid, fluid, etc.) as discrete "particles"), testing both integer computation speed and communication performance. This benchmark excludes floating point arithmetic.

Three varied replacement policies were chosen in order to observe the effects of different replacement models on power and performance. They are as follow:

Least Recently Used (LRU) : LRU is a recency-based policy that replaces the least recently used block, which involved tracking when blocks are accessed/re-referenced.

Static Re-Reference Interval Prediction (SRRIP) : SRRIP is a policy that uses a re-

reference prediction value (RRPV) to "predict" the how likely a block will be referenced again; this policy uses 2-bit RRPV and is likely to evict recently inserted cache blocks.

Round Robin : Round robin is a queue-based policy that replaces the cache blocks in sequential order, evicting the oldest block in a set in a first-in-first-out (FIFO) manner.

Simulations had either 8-, 16-, or 32-way L3 set associativity in order to observe how power and performance changed with the number of ways. L1 and L2 caches remained 4- and 8-way set associative, respectively, and both used LRU by default. SniperSim McPAT could not output data for a 64-way set associativity for the given L3 cache size (512 KB) and block size (64 bytes) and so was excluded from the experiment.

All the simulations ran concurrently using bash script(s) and GNU `parallel` shell tool [2], and post processing of the data were handled with python (v2.7) and bash scripts (included separately). Simulations ran on a python virtual environment and in a detached `tmux` session, due to long duration of the experiments. Sniper provided data processing tools used were: `gen_topology.py`, `cpi-stack.py`, and `mcpat.py`.

3 Results & Analysis

4 Conclusion

5 Appendix: Raw Post Processed Data

5.1 benchmark

5.1.1 Power Results

5.1.2 CPI Stacks