Lab 1 **Introduction Computer Architecture Experiments**

EE156-Comp140 Advanced Computer Architecture

Prof. Mark Hempstead Due Date: 2/17/23

TA: Parnian Mokri, Robert Costa and Bharat Kesari

1 Overview

This assignment intends to be an introduction to processor simulation done by computer architects to evaluate CPU designs. The assignment will show you how to do the following: Conduct an experiment which will explore the effect of last-level cache size and eviction policy on system performance and power.

2 Designing an Experiment to Study Cache Replacement

In this lab we want to design a larger experiment that can study the effect of cache associativity and replacement policy on performance and energy. You will use all of the skills you learned in lab0 to run sniper, kickoff parallel experiments with scripts, and process data.

You will design a larger experiment. Use a **last-level cache size of 512KB** and then you should sweep at least the following parameters:

Note: The total L2 size should always be smaller or equal to the size of your LLC since the Sniper cache architecture is inclusive. For this lab, you are allowed to shrink the L1 and L2 size to accommodate the 512KB LLC. Besides, in some sniper configuration, LLC is referred as L3.

- **Benchmark:** choose at least two benchmarks. Feel free to simulate more than two to find interesting comparisons. We recommend you use some of the same benchmarks you used in lab0 so you can reuse the data.
- **Eviction Policy:** Choose at least three eviction policies, examples include LRU and PLRU.

  Hint: The replacement policy implementations can be viewed in /path/to/sniper/common/core/memory_subsystem/cache.

  Note: PLRU in sniper is only implemented for 4-way and 8-way cache. SRRIP requires specifying the number of bits needed for holding the re-reference prediction value for each cache block. Checkout this Sniper Google Groups post for details.
- **Associativity:** Vary the associativity in the cache to test three different configurations. We suggest starting with an 8-way set-associative cache but not going below 4-way.

  Note: One way to confirm that a simulation did run with the indicated eviction policy and associativity will be to check the sim.out generated.

Notice how with this baseline configuration you are expected to run at least 18 experiments (54 if you sweep cache size). Scripting, using screen and parallel simulations will be very important here. The other parameters in your simulation setup should be held constant (e.g. number of threads, L1 cache size, CPU configuration). In your report you will want to clearly list these parameters.

2.1 Analyzing and Plotting Data

Good data analysis and presentation should be systematic and not a random plotting of all the data you have. When designing a figure you should first decide what main point you are trying to make with the plot (e.g. How cache size and replacement policy impact performance) which will inform your X-axis and the data series that you plot. Then design another plot that explores and evaluates another hypothesis. For example you might want another plot with different benchmarks and replacement policies but keeps associativity and cache size constant.

For example, Figures 1 and 2 from the textbook provide evidence for different relationships. Figure 1 sweeps cache size on the x-axis and has different series (different lines) for different cache associativity. In this figure the benchmark remained constant. In Figure 2 different benchmarks are swept but the rest of the configurations remained constant. Often writers will put a description of the figure and the experiment either in the caption or with the text that describes the figure.

Note: You might want to use metrics that combine both power and performance, e.g., energy-delay-product (EDP); if this is the case, you need to state why you did that and how it helps you to evaluate your topology.

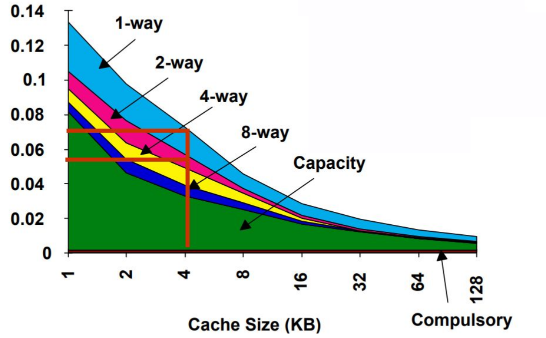

Figure 1: Associativity vs Cache Size

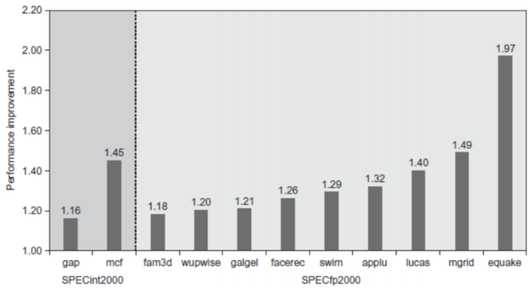

Figure 2: Effect of Prefetcher on Performance.

3 Submission

You should upload a report describing your experiment. **It is highly recommended that you use LaTeX for your report. There are many LaTex editors on our servers, online like Overleaf, and free editors like TeXstudio to download**. You should write your report from the perspective that you are a computer architect working in industry who is trying to inform his or her colleagues. Your report should include: introduction, experimental setup, results and analysis, and conclusion. Include topology, McPAT, and CPI diagrams. You should explain how varying the parameters affects your results (performance and power/energy). Then offer your analysis as to why you see the results you do. Do certain workloads yield better results (in terms of both performance and power/energy) than others for certain configurations? Why do you think that is?
Content of The Report

1. **Introduction:** Motivate the problem and provide an overview of what you are studying and why. For example: Describe what an eviction policy is and describe the eviction policies you are studying in this experiment. Describe why this is relevant to the design of microprocessors.

2. **Experimental Setup:** Describe the architecture you are modeling and include the topology diagram. Describe the experiment in words, what are the sweeps designed to show? For example what configurations did you sweep and why. Include a table or list that enumerates the configuration parameters and which parameters you swept. List and describe the Splash2 workloads you chose to run.

3. **Results and Analysis:** Plot and analyze your results. What performance metrics have you used to evaluate your architecture and why? What power/energy metrics have you used and why? You should study in detail **at least three different questions** each with a separate subsection. Each question will require at least one and perhaps multiple plots. For example, in a subsection called *Impact on Energy* you would explore how the choice of replacement policy impacts energy consumption.

4. **Conclusion:** Summarize the problem you solve, your results and discuss any future work. (Typically 1-2 paragraphs)

Files to Submit Using Provide

- ssh to homework or any other server : ssh homework
- zip all your files and provide it like this:

```
provide ee156 lab1  yourName-lab1.zip
```

1. Your lab report as a single PDF file lab1.pdf
2. README file describing the filename and content of the files you have submitted
3. Your submit script(s) used to run experiments, this can be either a shell script or python script
4. Your post-processing / data analysis scripts this could be either MATLAB, python or Excel.

4 Tips on Producing Relatively Interesting Results

1. Shrink L1 and L2 cache sizes, but no smaller than 32KB. A small L1 and L2 will move the memory activity to LLC. Therefore, power and performance results are more sensitive to changes of LLC configuration. McPAT do not have configurations for cache size smaller than 32KB, which will result in an inability to produce power.txt.
2. Simulate with large data set, using "-i large". A larger data set results in more instructions, thus better exercising the cache. It would however increase simulation time.
3. Use memory-intensive benchmarks, for example: npb-is. For more information on the NPB benchmarks, checkout YOUR_PATH_TO_SNIPER/benchmarks/npb
4. If necessary, you can shrink LLC size to 256KB, but it won't work for 32-way or higher associativities since McPAT doesn't have those configurations implemented.
5. Remember that you need to specify the output directory for each experiment. If you don't, it will dump every output file to a default location (Usually the directory that you ran it from). That means, it will overwrite new outputs over old outputs unless you make sure each output has a unique name before running it; you might need to write a script or edit the current scripts.