

CPSC 614 – COMPUTER ARCHITECTURE

Summer 2004

Project Description

Released: 06/25/2004

Due: 07/02/2004 (2359hrs)
Submit via email to the TA.

Objective: To study the effects of adding a second-level TLB.

You are required to do the following:

1. Implement a 2-level TLB architecture in SimpleScalar. You will work with `sim-cache.c`.
2. Configure `sim-cache` to accept command line arguments (and via config files) for setting up the 2-level TLB.
3. Obtain the following results:
 - a. AMAT (Avg Memory Access Time),
 - b. miss-rate,
 - c. hit-ratew/ and w/o 2nd level TLBs; for the given set of benchmarks, and varying sizes of TLBs (128 bytes -> 512 bytes).
4. Prepare a report containing your results and their analysis. You are also required to document the code modifications that you make. (*Only modified code!*)

You are provided with the following:

1. *Benchmarks and their corresponding inputs for Alpha architecture:*
`galgel, mcf, swim, vpr, lucas, twolf`.
These can only be executed on little endian systems – for eg. linux.cs.tamu.edu.
2. Directions for specifying command line inputs for each of the benchmarks (`benchmark.txt`).

Hints:

- See how other command line arguments are handled.
- You will need to introduce a TLB into the architecture. How is the first level TLB introduced?
- You will need to introduce a miss handler to handle misses in this new architecture. How do the other components (caches/TLBs) handle these misses?
- How are statistics tracked?
- Don't go running to your TA for everything! Try to address the problem by yourself.

Note: Keep checking back to the class website for updates to the project handout!