

ECE 563: COMPUTER ARCHITECTURE (Fall 2018)

INSTRUCTOR: Prof. Sheng Wei

TA: Chen Wang

## SIMULATION PROJECT

ISSUE DATE: 10/5/2018

DUE DATE: Friday 11/16/2018, 23:00pm

POINTS: 15%

SUBMISSION: one report per project team

### INTRODUCTION

The main goal of this project is to run a set of simulations using the SimpleScalar simulator, which helps you understand the basic concepts introduced in the lectures, such as instruction set architecture, quantitative performance analysis, memory hierarchy, and instruction-level parallelism.

### PROJECT PREPARATION

#### Step 1: SimpleScalar setup

SimpleScalar is a simulator used for performance analysis and system verification at the microarchitectural level. It supports various ISAs, including Alpha, PISA, ARM, and x86. Since we focus on caches on this project, we will use sim-cache. More information can be found on [www.simplescalar.com](http://www.simplescalar.com)

1. Download SimpleScalar 3.0 ([www.simplescalar.com](http://www.simplescalar.com))
2. Extract it: `tar xvfz simplesim-3v0e.tgz`
3. Install SimpleScalar:

```
$ cd simplesim-3.0
```

```
$ make
```

4. Run a sample benchmark and verify it works

```
$ ./sim-cache tests-alpha/bin/test-math
```

Your output should include the following line, verifying that everything works:

```
“-1e-17 == -1e-17 Worked!”
```

#### Step 2: Benchmarks setup

Download benchmarks (distributed by Prof. Todd Austin at University of Michigan): Towards this end, I am attaching separately a number of complementary documents under folder Resources/Simulation Project/ in Sakai.

## PROJECT REQUIREMENTS

The project consists of the following 3 simulations, for which the detailed instructions and requirements are available online (courtesy of Dr Israel Koren from UMass):

<http://www.ecs.umass.edu/ece/koren/architecture/SimpleScalar/>

1. Lab Experiment 1 (General Instructions) (5%)

<http://www.ecs.umass.edu/ece/koren/architecture/SimpleScalar/lab1.htm>

2. Lab Experiment 6 (Caches I) (5%)

<http://www.ecs.umass.edu/ece/koren/architecture/SimpleScalar/lab1caches.htm>

3. Lab Experiment 5 (Branch Prediction) (5%)

[http://www.ecs.umass.edu/ece/koren/architecture/SimpleScalar/branch\\_prediction.htm](http://www.ecs.umass.edu/ece/koren/architecture/SimpleScalar/branch_prediction.htm)

Please follow the detailed instructions from the link above and complete the 3 simulations. Then, each team is expected to submit a simulation report that contains the following components:

- Names of all the team members.
- Descriptions about the simulation setup and configurations (e.g., tools/benchmarks used and the simulation parameters), as well as any problems your team encountered and later addressed. The descriptions should be clear and compressive enough for the readers to follow and reproduce your simulation results shown in the report.
- Answers (with justifications) to each question listed in the simulation pages. Please fill the tables and plot the figures as needed.
- Raw data files obtained from the simulations.

The simulation report should be written using either IEEE or ACM templates for conferences proceedings:

<https://www.ieee.org/conferences/publishing/templates.html>

<https://www.acm.org/publications/proceedings-template>

There is no page limit on the simulation report. Once completed, please submit the completed simulation report in a single PDF file to **Assignments in Sakai** no later than the stated deadline.

## NOTE

Combined effort will be required to install and run SimpleScalar. Each team needs to manage among all team members to first install and run Superscalar on one machine successfully and then provide the rest of the team more updated guidelines and tips for doing so. In the simulation report, you are expected to describe the problems that came up during the simulations and how you overcame them. **ALL TEAM MEMBERS** are expected to work equally on this project and be able to answer the questions related to the simulations.