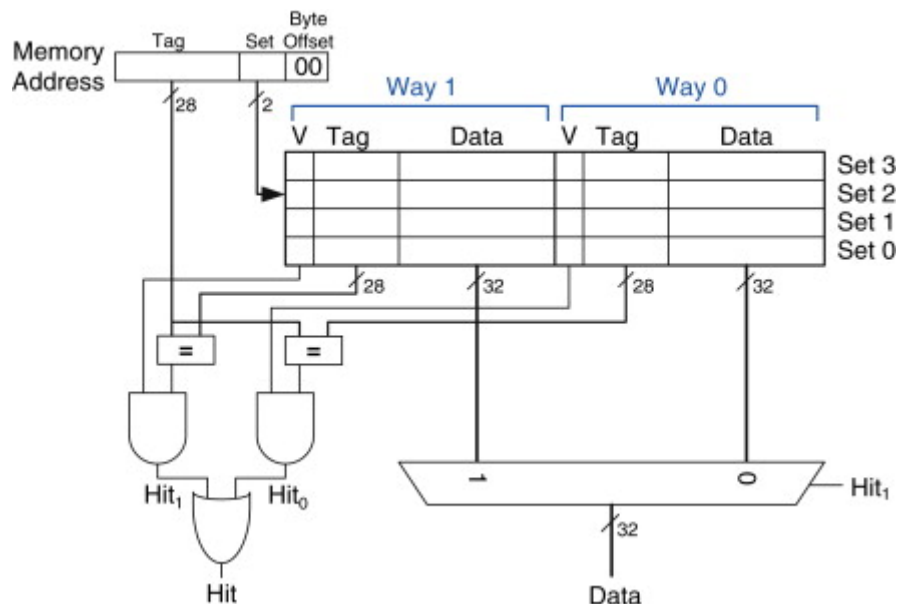
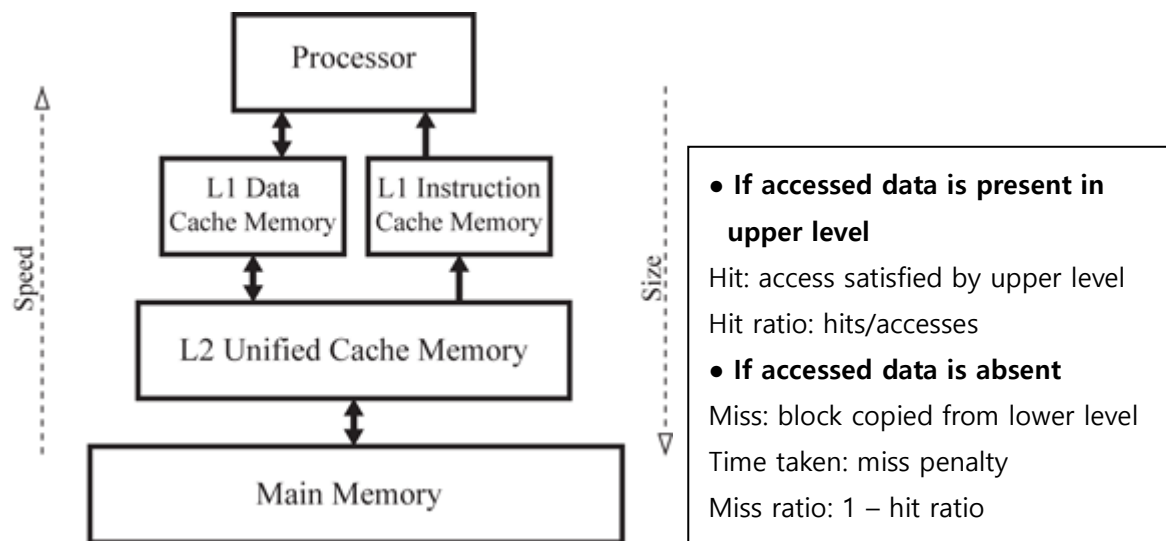


# Computer Architecture – Assignment #4

## Cache Design

### 1. Introduction

Cache is a smaller, faster memory which stores copies of the data from the most frequently used min memory locations



- Higher associativity increases the chance that the latest accessed block stays in the cache.
- Data accesses often show more irregular access patterns than instruction accesses do, which lead to more associativity.

## 2. Assignment

In this project, the experiments have two phases.

### 2.1 Observation for Program Behaviors

First, discuss about the timing of how Instruction cache and Data cache operate with two programs: Bubble sort and Random access.

- The microprocessor is a MIPS-based 5-stage pipelined architecture machine and is implemented in Verilog.
- The Instruction cache is a direct-mapped cache with 32 blocks.
- The Data cache is a 2-ways set associative cache with 32 blocks (i.e, Each way has 16 blocks). The replacement policy is LRU.
- Each block has 4 32-bit words.

### 2.2 Cache Simulation

Second, you are going to configure two levels of caches for a microprocessor called SimpleScalar. To do so, you are going to measure the performance with three SPEC CINT95 programs.

**Table 1 – CINT95 Benchmarks**

Benchmark	Application Area	Specific Task
m88ksim	simulator	A chip simulator for the Motorola 88100 microprocessor.
compress95	utility	An in-memory version of the common UNIX utility.
vortex	database	An object oriented database.

- The SPEC benchmark suite is designed to measure the performance of a computer system, which means it requires a high performance simulator with many OS functionalities.
- SimpleScalar is a processor simulator tool that supports cache simulation, and it is sufficiently fast by implementing the simulator in C language.
- In this phase of experiments, discuss about the performance changes while altering several aspects of the cache configuration.
  - ◆ Unified cache / Separate cache
  - ◆ L1 cache size / L2 cache size
  - ◆ Large block size / Small block size
  - ◆ Direct-mapped / Set-Associateive
- The performance of the cache can be measured by AMAT. Use the following values.
  - ◆ L1 cache access time is 1 cycle, L2 cache access time is 20 cycles, and Main memory access time is 200 cycles.
  - ◆ By doubling the cache size the cycle time increase 4%, and by doubling the associativity the cycle time increase 2%.

### 3. Experiments Guide

For the first phase, you have to check the timing and discuss about:

- How I/D L1 hit operates, how I/D L1 miss operates
- When and how main memory (L2) operates
- The differences between Bubble sort and Random access in terms of cache behaviors
- What if data size increase?

For the second phase, finding the suitable cache for given benchmark programs:

- The original purpose of the benchmark programs given (not to measure the performance)
- Discuss about the memory access patterns of the benchmark programs in the aspect of algorithm (for example: there is a matrix multiplication, so the program may show 2d array access)
- Perform experiments with several cache configurations and discuss about the results
- Discuss about the best fit cache configuration for each benchmark program and compare and discuss about the differences among the configurations

\* There is no correct answer. Just analyze suitable cache and compare program

\* The larger the memory size, the higher the price, so Think of the appropriate price and set the cache

#### 4. 결과 Report(2장 이상으로 작성)

##### ■ 실험 내용

- 정렬 프로그램과 Benchmarks 프로그램에 각각 적합한 cache를 찾고, 각각의 프로그램의 적합한 cache가 왜 다른 건지 프로그램을 분석하여 비교

##### ■ 검증 전략, 분석 및 결과

- AMAT를 이용하여 적합한 Cache configuration 분석

##### ■ 문제점 및 고찰

- 프로젝트 내용 전체 정리 및 고찰
- Benchmarks에 최적화된 cache와 정렬 프로그램에 최적화된 cache 비교 및 분석

##### ■ Soft copy

- Due data: 2022/06/14(화요일) 23:59시까지. (딜레이 받지 않음.)
- 결과 Report를 작성한 파일을 압축하여 U-campus에 upload.
- 압축 파일명 양식: 학번\_이름\_Assignment#4.zip
- ex) 201X720064\_김땡땡\_Assignment#4.zip