

# TDTS08: Advanced Computer Architecture

Lesson

2011



**Linköping University**  
expanding reality

# Outline

- Lab organization and goals
- SimpleScalar architecture and tools
- Assignment on multiprocessor systems
- Exercises

# Organization

- Assistants
  - Group A1 & A2: Bogdan Tanasa
  - Group B1 & B2: Ke Jiang
- Web page
  - <http://www.ida.liu.se/~TDTS08>
  - Check the lab pages!

# Organization

- Sign up in Webreg
- Deadline for the assignments:

|                 |                     |
|-----------------|---------------------|
| Lab 1 and Lab 2 | November 24th, 2011 |
| Lab 3 and Lab 4 | December 20th, 2011 |
| Lab 5           | January 9th, 2012   |

- **Rules:** Read them! (linked from the lab pages)

# Examination

- Written report for each lab
  - Hand in the report enclosed in a lab cover; it must be signed by **both** group members
    - Hand in at a lab session
    - Put in the box outside your assistant's office
  - Returned in the box outside your assistant's office

# Labs

- 5 labs
  - 1. Cache memories
  - 2. Pipelining
  - 3. Superscalar architectures
  - 4. VLIW processors
  - 5. Multiprocessor systems
- Labs homepage
  - <http://www.ida.liu.se/~TDTS08/labs>

# Goals

- Obtain knowledge about computer organization and architectures
- Insights in various trade-offs involved in the design of a processor
- Become familiar with a set of tools necessary for evaluation of computer architectures
  - Simulation tools!

# Environment

- Unix
  - Simulations are started from the command line
    - ! If you are not familiar with the Unix environment:
      - Search the Internet
      - Tutorials
      - List of basic commands
  - Make sure that you learn the basic commands in order to be able to work in a command-line environment



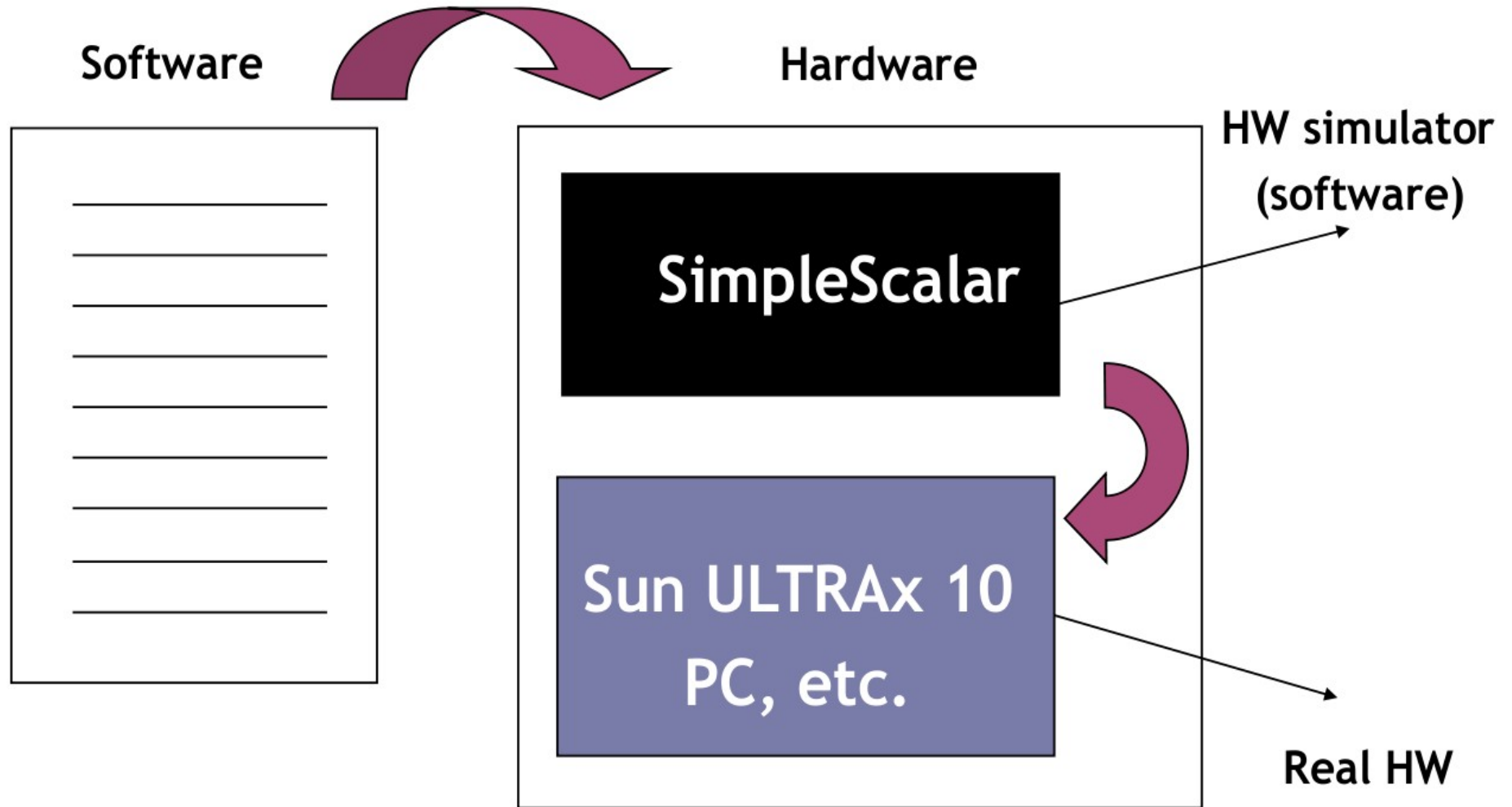
# Outline

- Lab organization and goals
- **SimpleScalar architecture and tools**
- Assignment on multiprocessor systems
- Exercises

# Introduction

- Issues covered in our lab:
  - Cache memories
  - Instruction pipelining
  - Superscalarity
  - VLIW (Very Long Instruction Word) processors
  - Multiprocessor and Multi-computer systems

# Architecture Simulation



# SimpleScalar: Literature

- “The SimpleScalar Tool Set, Version 2.0” by Doug Burger and Todd M. Austin
  - Very important preparation for the labs
  - Used all the time during all labs!
- User’s and Hacker’s guide (slides by Austin)
  - Linked from the lab pages

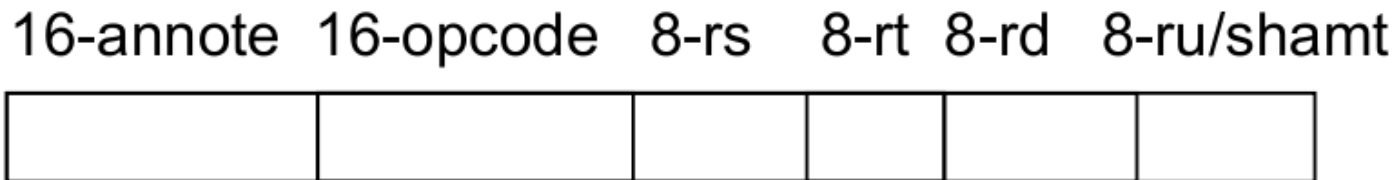
# SimpleScalar Architecture

- Virtual architecture derived from MIPS-IV
  - SimpleScalar ISA semantics are a superset of MIPS
    - Control (j, jr,..., beq, bne,...)
    - Load/Store (lb, lbu, ...)
    - Integer Arithmetic (add, addu, ...)
    - Floating Point Arithmetic (add.s, add.d, ...)
    - Miscellaneous (nop, syscall, break)

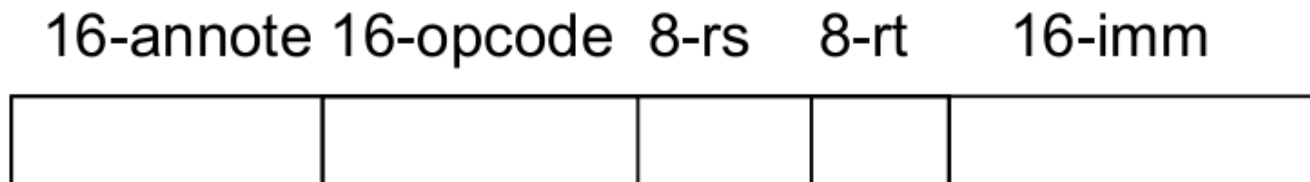
# SimpleScalar Architecture (cont'd)

- Instruction encodings (64 bits)

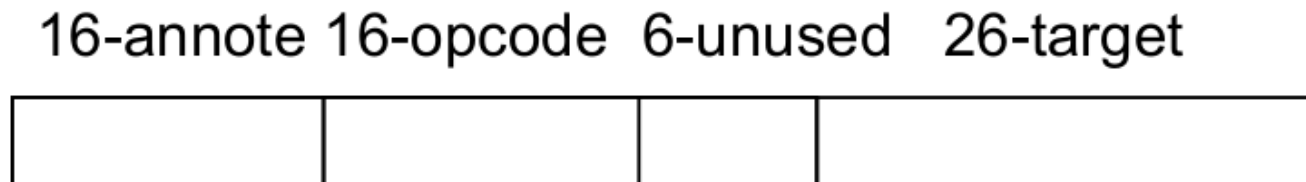
- Register format



- Immediate format



- Jump format



# SimpleScalar Architecture (cont'd)

- Registers
  - 32 integer registers + PC, HI, LO
  - 32 floating-point registers + FCC
- Virtual memory:
  - 0x00000000 - 0x003fffff unused
  - 0x00400000 - 0x0fffffff text (code)
  - 0x10000000 - ..... data
  - ..... - 0x7fffc000 stack
  - 0x7fffc000 – 0x7fffffff Args and Env

# SimpleScalar Architecture (cont'd)

- Several simulators
  - Sim-fast: Fast, only functional simulation (no timing)
  - Sim-safe: Sim-fast + memory checks
  - Sim-cache: Sim-safe + cache simulation and various timing properties (simulation time, measured time, ...)
  - Sim-cheetah: Simulation of multiple cache configurations
  - Sim-outorder: Superscalar simulator



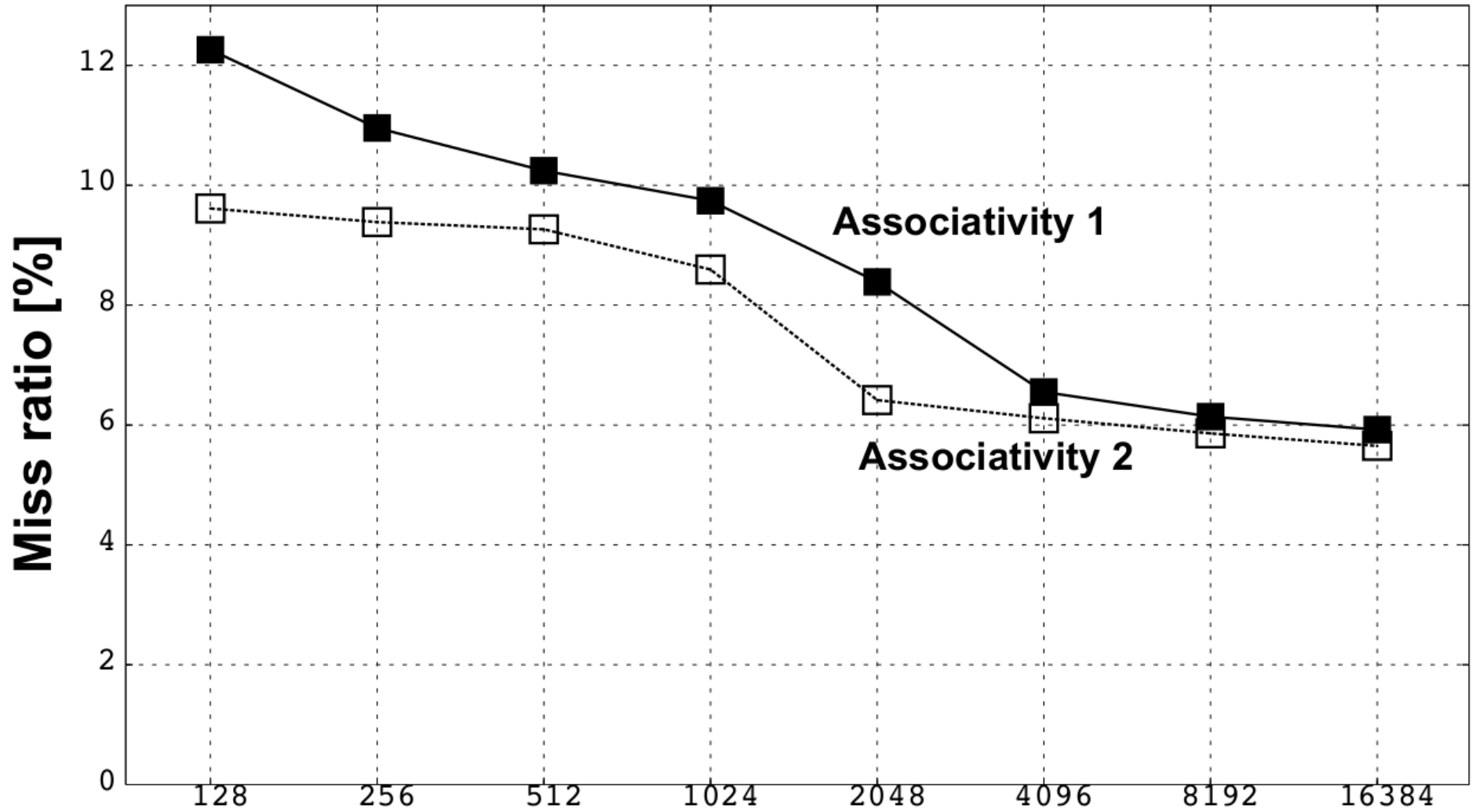
# SimpleScalar Architecture (cont'd)

- Tool set installed in `~TDTTS08/bin`
- Configurable through command-line arguments or files (recommended):
  - `-dumpconfig <filename>`
  - `-config <file-name>`
- gcc cross-compiler available for generating binaries to be executed on SimpleScalar
  - Binaries have been generated and are available in the course directory

# Demonstration

- Set the environment
  - `setenv PATH "$PATH":/home/TDTS08/bin`
- Lab1, assignment 3
  - Dump the default configuration of *sim-cheetah*
  - Modify the configuration and simulate
  - Plot the results (e.g. OpenOffice, Gnuplot, Matlab, Excel)

# Demonstration



# Outline

- Lab organization and goals
- SimpleScalar architecture and tools
- **Assignment on multiprocessor systems**
- Exercises

# Lab 5: Multiprocessor Systems

- Assignment:
  - Select an article on a multi-core, multiprocessor, multi-computer system, or a graphics processor
    - List of papers is available on the course page
    - You may select other articles if your lab assistant agrees
  - Review the selected article
  - Write a review report on the article
  - Self-learning based, no lab session allocated

# Multiprocessor Systems (cont'd)

- Read and understand the paper
  - If the course literature does not help you, investigate the referenced papers.
  - Searching the Internet can help you find explanations of abbreviations and terms

# Multiprocessor Systems (cont'd)

- Analyze the paper
  - Classify the architecture (MIMD, SIMD, NUMA)
  - Possible questions to ask:
    - Why has the actual method/approach been selected?
    - What are the advantages and disadvantages?
    - What is the application area?
    - What has been demonstrated?
    - ...

# Multiprocessor Systems (cont'd)

- Write a report
  - ~1000 words
  - Submit, in PDF format, to your lab assistant
    - bogta62.liu@analys.urkund.se (Bogdan Tanasa)
    - kejiang.liu@analys.urkund.se (Ke Jiang)



# Outline

- Lab organization and goals
- SimpleScalar architecture and tools
- Assignment on multiprocessor systems
- Exercises

# Exercises

- Review questions (page 146)
  - 4.4 and 4.8
- Problems (pages 147–149)
  - 4.8 (mandatory, Lab 1.1)
    - Include your solution in the report for Lab 1
  - Additional exercises in this order:
    - 4.15 (locality, preparation for Lab 1.2)
    - 4.22 (average memory-access time)
    - 4.17 (performance enhancement using cache)