EE 4720—Computer Architecture

01-1

Call Number 1665 (Spring 2000)

URL: `http://www.ee.lsu.edu/ee4720`

Offered by:

David M. Koppelman
349 EE Building
388-5482, `koppel@ee.lsu.edu`, `http://www.ee.lsu.edu/koppel/koppel.html`
Tentative office hours: Monday 9:40–11:10, Thursday 14:00–16:30.

Should already know:

How to design a computer.

Will learn:

How to design a *good* computer.

01-2

Prerequisites By Course:

EE 3755, Computer Organization.

Prerequisites By Topic:

- Logic design.
- Computer organization.
- Assembly-language programming.

Text

"Computer architecture, a quantitative approach," John L. Hennessy & David A. Patterson, Second Edition.

01-3

Course Content

- Instruction set (ISA) design.
- Pipelined processor design.
- Multiple-issue processor design.
- Caches and memory.

Course content will closely follow text.

Lecture material not in book will be marked: (NIB).

Graded Material

01-4

Midterm Exam, 40%

Fifty minutes, closed book.

Final Exam, 40%

Two hours, closed book.

Yes, it's cumulative.

Homework, 20%

Written and computer assignments.

Lowest grade or unsubmitted assignment dropped.

Course Usefulness

01-5

Material in Course Needed For:

- General-purpose processor designers and testers.
- Special-purpose processor designers and testers.
- Compiler writers.
- Programmers of high-performance systems.
- Answering job interview questions.

Material Covered in This Set

01-6

Coverage: Sections 1.2 and 1.3.

Slides and other material via `http://www.ee.lsu.edu/ee4720`

Web site also has homework assignments, exams, grades, and other material.

ISA and Implementation Distinction

01-7

What is a computer?

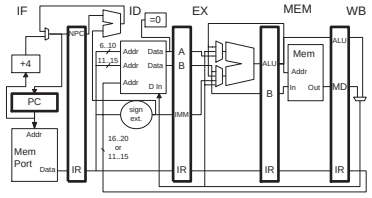
A machine that executes instructions which read and write memory.

What a computer engineer does:

- Develops an *instruction set architecture* (ISA):

```
ld  r1, 0(r2)  ! Load register r1 with contents of mem at r2.
add r3, r1, r4
sw  0(r2), r3
```

- Designs hardware to execute, *implement*, the instruction set:



01-8

Definitions

Instruction Set Architecture (ISA):
Precise definition of computer's instructions and their effects.

- It's all programmer needs to program machine.
- It's all hardware designer needs to design machine.

Implementation [of an ISA] (noun):
Hardware that executes instructions defined by ISA.

Instruction Set Architecture

01-9

ISA and Implementation Examples

ISA: SPARC V8. (Developed by Sun for its workstations.)

Impl: Cypress CY7C601 and Fujitsu MB86900/1A.

Who ISA Developed For

- Compiler writers.
- Compute-intensive library writers.
  E.g., graphics and scientific libraries.

Instruction set requirements don't change very much over time.

Scope of ISA Specification

Describes instruction codings, and what they should do.

Should specify action of all codings, used or not ...

EE 4720 Lecture Transparency. Formatted 13:07, 19 January 2000 from lsli01.

Implementation

01-10

Two aspects of implementation: *organization* and *hardware*.

Organization:
Details of functional units, data paths, control, etc.

Also called *microarchitecture*. (NIB[1]).

This includes memory system, bus, and CPU.

Hardware:
Logic design and packaging.

Course focus: ISA and organization, not hardware.

[1] Not in book.

EE 4720 Lecture Transparency. Formatted 13:07, 19 January 2000 from lsli01.

Technological Change

01-11

Technological Change and Computer Designer

Technology determines "raw materials" for designer.

ISA lifetime can be decades.

Raw materials greatly change over this time.

So, design ISA for now and future.

EE 4720 Lecture Transparency. Formatted 13:07, 19 January 2000 from lsli01.

01-12

How technological advancement affects processor.

Transistor Speed, Clock Rate
No changes to organization or ISA.

Number of Transistors Available
Changes to organization and possible changes to ISA.

Memory Size
Change ISA to use larger address space.
Can use ISA having larger instruction codings.

Memory Speed Compared to Processor Speed
Include more sophisticated caching in organization.

EE 4720 Lecture Transparency. Formatted 13:07, 19 January 2000 from lsli01.

01-13

Summary

What a computer engineer does:

- Develops an *instruction set* (ISA).
- Designs hardware to execute instruction set.

If instruction set *poorly* designed ...

... many instructions will not be used (wasting silicon) ...

... and instructions will execute slowly.

Why ISA design is surprisingly difficult:

- Hard to predict which instructions useful ...
  ... without writing and running software using instructions.
- Hard to predict which instructions fast ...
  ... in current *and future* technologies.

EE 4720 Lecture Transparency. Formatted 13:07, 19 January 2000 from lsli01.

01-13