CSE 141 -- Introduction to Computer Architecture

Larry Carter
What is Computer Architecture?

- **Instruction Set Architecture (ISA):**
  - Anything a programmer needs to know to make an assembly-language program work correctly.
    - Instruction formats
    - What the instructions do
    - number and types of registers
    - addressing modes, exceptional conditions, ...

- **Architecture (broader definition):**
  - ISA + implementation
  - Structure, behavior and **performance** of computer systems (primarily hardware)

You should know meaning of all underlined words
140 vs 141

CSE 140
• Hardware Designer
  - thinks about circuits, components, timing, functionality, ease of debugging

“construction engineer”

CSE 141
• Computer Architect
  - thinks about high-level components, how they fit together, how they work together to deliver performance.

“building architect”
The Challenge of Computer Architecture

• The industry changes faster than any other.

• The ground rules change every year.
  - new problems
  - new opportunities
  - different tradeoffs

• It’s all about making programs run faster than the next guy’s machine.
What we’ll study

• Internal organization of computers
• How it affects performance
• Vocabulary
• Clever ideas
• Historical perspective
• Architectural trends and tradeoffs
• Value of abstraction and standards
Meta subjects

• Algebra (word problems, logarithms, ...)
• Numeracy
• Economics & business
• Common sense
• Bright ideas vs. good ideas vs. successful ideas
Administrivia

• Instructor -- Larry Carter
  - Office hours (4101 APM)
    • Monday & Friday 10 - 11
    • Wednesday 3-4
  - carter@cs.ucsd.edu

• TAs:
  - Vineet Kumar (vineet@cs) Mon 6:30-7:30, Thurs 5:30-6:30
  - Rakesh Kumar (rakumar@cs) Tues, 10-12
  - Greg Chun (gchun@cs), Mon 2:30-3:30 and Wed, 11-12

• Section meeting
  - 2:30 Fridays, CSB 001
  - We can set up a second section if desired
More administrivia

• Tests & homework:
  - 3 30-minute quizzes (4\textsuperscript{th}, 7\textsuperscript{th}, and 10\textsuperscript{th} Wednesday)
  - 4 graded homeworks (relatively short)
  - Ungraded homeworks
    • discussed in sections
    • good practice for quizzes

• Grading:
  - Best 2 quizzes - 20% each
    • worst quiz score ignored (I may use in borderline decisions)
    • NO MAKEUP QUIZZES!
  - Graded homeworks - 5% each
  - Final - 40%
More administrivia

• Integrity
  - Ungraded homework: you are encouraged to work with others.
  - Graded homework: to be done entirely by yourself.
    - If you need clarification or a hint, ask me or a TA.
    - I will persecute violators to my fullest ability!
  - Quizzes & final. Closed book, but you may bring in one HANDWRITTEN page of notes. No calculators.

• Relationship with 141L
  - Dean Tullsen & I will coordinate material.
  - TA’s are (to some extent) working on both courses.
  - 141L is a LOT of work. (But it’s a good course!)
Website

• www.cs.ucsd.edu/classes/wi02/cse141
• Assignments, late-breaking news, etc.
• *May* have these slides before class
  - but perhaps not until after class.
Textbook

• Patterson & Hennessy, second edition of "Computer Organization, the Hardware/Software Interface"
  - Exceptionally good book. We’ll read most of it.
    • Published 1997 (Pentium Pro was latest Intel processor)
  - Patterson is professor at Berkeley;
    • lead RISC project (foundation of SPARC processor)
    • lead RAID (redundant array of inexpensive disks) project
  - Hennessy is professor at Stanford
    • now Dean of Engineering
    • co-founded of MIPS Computer Systems
  - Note: same authors wrote the graduate textbook, “Computer Architecture, A Quantitative Approach".
Approximate Course Outline

Weeks 1-3: Chapters 1-4 + some math
  Performance and Performance Metrics
  Instruction Set Architectures

Weeks 4-6: Chapters 5-6:
  Three implementations of basic MIPS instructions
    Single-cycle, Multiple-cycle, Pipelined implementation

Weeks 7-10:
  The Memory/Cache Hierarchy
  Superscalars, parallel machines, ... (as time allows)
The five classic components of computers

- Processor
- Datapath
- Control
- Memory
- Input
- Output
Output devices

• **Display** (aka Monitor)
  - **CRT** (cathode ray tube)
  - **LCD** (liquid crystal diode)
    - lighter, thinner, uses less power
  - Typical size: "1 million *pixels* (picture element).
  - 8 to 24 bits per pixel

• **Printer**
  - **Ink jet**: cheaper
  - **Laser**: faster

"\*" means "about". Numbers like this are useful for "back of the envelope" estimates.
Disks & Tape

Also considered I/O devices

Fine print: portion of disk used as “virtual memory” could be called “memory”.

- **Hard disks** (magnetic surface on metal)
  - Very slow access time (~ 5 ms)
  - Getting inexpensive very fast

- **Floppy disks** (magnetic surface on mylar)
  - Cheap and convenient

- **CD’s** (compact disks) – optical
  - Even cheaper
  - Slow (or impossible) to write

- **Magnetic tape** – a dying technology (?)
Memory

- **SRAM** (static random access memory)
  - very fast: ~ 1 ns access time
- **DRAM** (dynamic random access memory)
  - very dense (1 transistor per bit)
  - low power
  - ~ 30 ns access time
    - slow compared to SRAM
    - but over 100,000 times faster than disk access
- **SDRAM** (synchronous DRAM)
  - DRAM, but improved speed for getting a block of data
- **VRAM** (Video RAM)
  - SDRAM with extra port for streaming data to display
Why care about power consumption?

• California’s energy crisis??
  - Not really

• Heat is hard to get rid of!
  - Workstation processor might use 70 Watts
  - Limits how densely components can be packaged

• Battery power is limited!
  - Embedded processors in portable devices
Typical home computer

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Price:</strong></td>
<td>$4000</td>
<td>$1700</td>
</tr>
<tr>
<td><strong>Speed:</strong></td>
<td>120 MHz</td>
<td>733 MHz</td>
</tr>
<tr>
<td><strong>Memory:</strong></td>
<td>16 MB DRAM</td>
<td>128 MB SDRAM</td>
</tr>
<tr>
<td><strong>Disc:</strong></td>
<td>1 GB</td>
<td>40 GB</td>
</tr>
<tr>
<td><strong>Video RAM:</strong></td>
<td>2 MB</td>
<td>32 MB</td>
</tr>
<tr>
<td><strong>Extras:</strong></td>
<td>10Mbit Ethernet, 4x CD ROM, ...</td>
<td>1 Gbit Ethernet, 32x CD R/W, ...</td>
</tr>
</tbody>
</table>

5 years → < 1/2 the price, 5-10 times better
A Quote from Robert Cringely

“If the automobile had followed the same development at the computer, a Rolls-Royce would today cost $100, get a million miles per gallon, and explode once a year killing everyone inside.”
Who needs all this computation?

Back Of The Envelope Estimate (BOTEE®): what’s needed for full-screen animation?

- Bandwidth (Bytes/sec) to monitor
- Bandwidth to disk (assuming MPEG compression)
- Processing power
- Disk capacity

(Worked out in class, if time permits)
Computer of the day

Computers:
4000 BC to 1940’s

• "Computer" was once a job title.
• Often assisted by mechanical devices (abacus, calculators ...)
• "Roman numeral" architecture (I, II, III, IV, V, ..., X, L, C, D, M)
  - Addition is pretty easy
• "Arabic numbers" architecture (..., -1, 0, 1, 2, ..., 10, 11, ...)
  - Easier multiplication
  - Much better for large numbers
Concluding Remarks

• Some things that I want you to get out of today’s (and future) class:
  - Vocabulary
  - Approximate characteristics of current computers and components
  - Ability to make estimates
  - Awe at how quickly computer industry evolves
Reading Assignment

• Read Chapter 1

• Skim Chapter 3
  - Particularly “Hardware/Software Interface”, and sections 3.12 – 3.16.
  - Brings together issues from languages, compilers, operating systems, and architecture.
  - You already know assembly-language programming (??).
  - We’ll go over details of relevant MIPS instructions later.

• Think about implications of, and reasons for, ISA choices.

• See website for homework problems (posted Weds).