IC220
Computer Architecture and Organization

Spring 2015
Assoc. Prof. Luke McDowell
LT Eric Watt

http://www.usna.edu/Users/cs/lmcdowel/courses/ic220/S15/

Outline

• Class Survey / Role Call
• What is:
  - a computer?
  - computer architecture?
  - this class?
• Course Admin
  – Policy Letter
  – Syllabus

Uniprocessor Performance

What We’ll Learn

• How do computers really work?
• How to analyze performance (and not to!)
• Issues affecting modern processors (caches, pipelines, wire delay, parallelism, power, mobile devices…)
• Constant tradeoffs:
  – Speed vs. Capacity vs. Cost
• Key concepts for using improving performance and using parallelism
• Comment from prior student on “how much learned”:
  “A great deal. One of those classes where you don’t realize how much you learned- you just come out understanding a lot of things that nobody else does.”
Why learn this stuff?

• You want to call yourself a “computer scientist” or “information technologist”
• You want to build software people use (need performance)
• You need to make a purchasing decision or offer “expert” advice

5 Classical Parts of a Computer

• 1.
• 2.
• 3.
• 4.
• 5.

What is a computer?

What is a computer, continued

• Our primary focus:
  – implemented using millions of transistors
  – Impossible to understand by looking at each transistor
  – We need...
Abstraction

- An abstraction helps us cope with complexity by:
  - Delving into the depths reveals more information

What is Computer Architecture?

Computer Architecture =

Instruction Set Architecture

- A very important abstraction
  - interface between hardware and low-level software
  - defines how a program interacts with the machine
  - standardizes instructions, machine language bit patterns, etc.
  - advantage:
    - disadvantage:

- Modern instruction set architectures:
  - 80x86, PowerPC, MIPS, SPARC, ARM

Example of Abstractions

1. Instruction set architecture (ISA)
   - The hardware/software interface
2. Application binary interface (ABI)
   - The ISA plus system software interface
3. “Virtual Memory”
   - Will see later
Multiprocessors

- Big new trend: Multicore microprocessors
  - More than one processor per chip
- Requires explicitly parallel programming
  - Compare with instruction level parallelism (ILP)
    - Hardware executes multiple instructions at once
    - Hidden from the programmer
  - Hard to do
    - Programming for performance
    - Load balancing
    - Optimizing communication and synchronization
- So why not just make faster single-core processors?

Where we are headed

- Computer Abstractions & Technology (Chapter 1)
- A specific instruction set architecture (Chapter 2)
- Logic Design (Appendix C)
- Arithmetic and how to build an ALU (Chapter 3)
- Performance issues (back to Chapter 1)
- Constructing a processor to execute our instructions (Chapter 4)
- Memory: caches and virtual memory (Chapter 5)
- I/O (various sections)
- Pipelining to improve performance (more Chapter 4)
- Multiprocessors and advanced topics (Chapter 6)

Power Trends

- In CMOS IC technology

\[
\text{Power} = \text{Capacitive load} \times \text{Voltage}^2 \times \text{Frequency}
\]

Eight Great Ideas

- Design for Moore’s Law
- Use abstraction to simplify design
- Make the common case fast
- Performance via parallelism
- Performance via pipelining
- Performance via prediction
- Hierarchy of memories
- Dependability via redundancy
Success in IC220

• In Class – Be Active
  – You **must** bring relevant slides/homework
  – Ask & answer questions
  – Be prepared to interact
  – Take notes – provided slides are not enough!
• On your own – Review
  – Review notes AND exercises (from HW) after class
  – Read the book – lecture won’t cover everything
  – See me for help and/or talk to friends

Assignments

• Get the textbook
  – Lots of chapter 1 & 2 reading – see calendar
• Get a 3-ring binder to keep track of notes
• Homework #1 soon (see calendar)
• Likely quiz at start of next class!

Admin

• Pet Peeves
• Policy
• Collaboration
• Syllabus
• Homeworks
  – *Some* exercises completed in class
  – *All* exercises must be completed & turned in
  – Not everything will be graded
  – Expect less points for exercises done in class, if graded
  – On due date: HW is complete when you enter the classroom
• Daily Quizzes
  – Direct from previous day’s notes OR (in-class) exercises!
  – Review notes+exercises → learning complete, ace quiz
  – Everyday: pick up blank paper as you come in
• All assignments must be turned in to possibly earn a passing grade