CS 365: Computer Systems Architecture

Spring 2006, Section 002

Prof. Yih Huang

Welcome to CS365

- Instructor: Yih (Ian) Huang
  - Office: ST2, 443
  - Office Hours: Wednesday 1 to 3pm
  - Email: huangyih@cs.gmu.edu

- All course materials will be available at the course home page
  
  cs.gmu.edu/~huangyih/365
Check the course page before class. You are responsible for
  – Print out slides before class
  – Complete reading assignments after class

We will also use emails for communications; you must have a GMU account and check the account for messages daily.

Teaching Assistant

TA: Abhilasha Chaudhary
Office: 365 S&T II
Office Hours: TBA
Email: achaudh9@gmu.edu
Textbook


- Slides, available on the course home page
  - Contributions by Dr. Setia, Dr. Chen, and Dr. Simon are gratefully acknowledged

Prerequisites

- CS 265 (Assembly Language Programming)
- ECE 301 (Digital Logic)
- Understanding of binary numbers, 2’s complement numbers are fundamental to the course.
Grading

- Midterm exam (25%)
- Final Exam (40%)
- Quizzes and Assignments (35%)
- Grading is proficiency-based (no curve). Cutoffs will be in the vicinity of, but not higher than:
  
  \[ A > 95\%, \quad A- > 90\%, \quad B+ > 85\%, \quad B > 80\%, \quad C > 70\%, \quad D > 60\% \]

Quizzes and Assignments

- There will be weekly/biweekly quizzes.
- Quizzes also serve as roll calls.
  - There will be no make-ups
  - If you cannot attend a class for good reasons, inform me in advance and the quiz is waived.

- Late assignment policy:
  - Unless under prearranged conditions, no late assignments will be accepted
Introduction

- What does your shining new PC really know?
  - The Start button?
  - Word processing?
  - Counter your moves in Doom?
- The processor knows very little.
- It merely follows instructions provided by humans.
  - All intelligence is ours

Programs

- Programs are instructions we give to computers to follow.

- High level language programs
  
  ```
  temp = v[k];
  v[k] = v[k+1];
  v[k+1] = temp;
  ```
Instruction Set

- A processor is hardwired to perform a set of very primitive actions, called machine instructions.
  - Integer arithmetic
  - Floating point arithmetic
  - Change of control flow based on (simple) conditions
- Machine instructions are expressed as binary numbers.
Languages

- Assembly languages allow us to give machine instructions in a relatively convenient way.
  - The machine does not understand assembly language
  - A translator, assembler, is needed
- High level language allow us to give instructions in human-friendly notations.
  - Very complicated translators, called compilers, needed to produce machine instructions.

Instruction Set Architecture

- User
- Application
- Operating System
- Compiler
- Firmware
- Instruction Set
- I/O system
- Datapath & Control
- Digital Design
- Circuit Design
- Layout
- Transistors / Semiconductor
- Electrons / Holes
Computer Architecture

- Computer Architecture =
  - Instruction Set Architecture +
    - Machine Organization

- Instruction Set Architecture is the
  *interface* between the hardware +
  - software
  - Focus of next few classes

Course Objectives

- By the end of the semester, you will understand
  - the interface between software and hardware
  - how the hardware executes the program
  - what determines computer performance and how
    to improve it

- It will help you
  - Appreciate how much existing software is doing
    for you
  - Procure a comprehensive understanding of this
    modern beast called computer
  - Be a better programmer
Topics

- Instruction set architecture
- MIPS instruction set
- Performance issues
- Arithmetic and how to build an ALU
- Constructing a processor
- Pipelining to improve performance
- Caches and virtual memory
- I/O

Why Do I Have to Take this Course?

- It’s required
- Don’t narrowly define you career
- Truly understand how computers work
- Communication with hardware guys
- Talk to nerds confidently
Isn’t this an EE Course?

- Computer performance is determined by the underlying VLSI technology (EE) and how all the hardware resources are organized and utilized (CS).
- Traditionally, the driving force of advances is clock rate --- An EE focus
- Recent trends, such as multi core processors, however put the CS in the driver’s seat.

Challenges

- Concepts are really not that hard (no difficult proves or algorithms)
- They are often buried in huge amount of details.
- In the exam, I provide cheat sheets to cover boring details.
- You learn how to use them in class, with the help of exercises, quizzes and homework.