ECE 550
Fundamentals of Computer Systems and Engineering

Introduction

Admin
- Professor: Andrew ("Drew") Hilton
  - E-mail: adhilton@ee.duke.edu
  - Office: Hudson 211
  - Office Hours: TBA
    - Or by appointment (e-mail me, we'll setup a time)
- TAs:
  - John O'Hollaren (Recitation)
  - Ziqiang (Patrick) Huang
  - Pat Pensabene

A bit about me
- Teaching is my primary job
- Don't be afraid to come to my office hours!
- Don't be afraid to ask me to setup some other office hours time!
- Please, feel free to call me "Drew"
  - Actually, I strongly prefer that to "Professor Hilton"
  - Why?
  - Formalism creates distance
  - Familiarity creates comfort
  - Typically makes me more accessible
  - Students who call me "Drew" often more willing to approach me for help

A bit about you
- Before we get too much further, I'd like you all to introduce yourselves:
  - Will try to learn your names quickly

Overview
- For: MS/MEng students who want Comp Eng focus..
  - ...but don't have Comp Eng undergrad
- Background for
  - ECE 522: Advanced Computer Architecture
  - ECE 554: Fault-Tolerant and Testable Computer Systems
  - ECE 556: Wireless Networking and Mobile Computing
  - ECE 559: Advanced Digital System Design
  - CS 510: Operating Systems
  - CS 512: Computer Networks/Distributed Systems

What we will learn: 10K feet
- Transistors -> Processor
  - Logic gates, combinational logic, sequential logic, FSMs
  - Adders, multipliers, shifters
  - Latches, Flip-flops, SRAMs, DRAMs, CAMs
  - Single-cycle datapaths, pipelining
  - Caches, memory hierarchy, virtual memory
  - Interrupts, exceptions, IO
- Hardware/software interface (ISA)
  - MIPS assembly
- Operating System basics
  - System calls, protection, multi-tasking, ...
- Networking basics
  - 7 layer OSI model, TCP/IP, routing,...
How We Will Learn It

- Must "learn by doing":
  - 4/5 homeworks: Implement something VHDL
- Write VHDL, synthesize it
  - Load it on Altera DE2 board
  - Run it, demo it to TAs
- 1/5 homework: Write MIPS assembly
  - Run it in SPIM

A Word About Varying Backgrounds

- I expect wide variations in backgrounds for this class
  - E.g., some know VHDL, some have never seen it
- Even if you are familiar with a topic we are covering...
  - You may learn something new
  - You may refresh rusty memory
  - I may present it slightly differently than you are used to
  - You may be able to help other students learn it
- If you missing some background (feeling lost)
  - Please come talk to me or a TA sooner rather than later!

A Few Notes on Grades

- I would love for everyone to earn an A (or even A+)
  - You have to earn it though: I don't give it away
- Last time: 96% As and Bs (53% As / 43% Bs)
  - Tests are "imminently reasonable" (quote a former student)
  - Show me you know the stuff I taught you, you will do well
- If you earn an F, you will get an F
  - Has happened in the past
  - Dire consequences for graduate students!

Grading Scale

- Grading: 10 points per letter
  - Top 3 points: +
  - Middle 6 points: no modifier
  - Bottom point: -
- This scale: minimum guaranteed letter
  - Get an 87?: Guaranteed at least B+

- I may adjust the thresholds slightly
  - Completely at my discretion

- Two reasons I typically might do this
  - Clustering of grades shows different break points
    - 90.2, 90.15, 90.1, 90.06, 90.01, 89.9, 89
Grading Scale

- Grading: 10 points per letter
  - Top 3 points: +
  - Middle 6 points: no modifier
  - Bottom point: -
- This scale: minimum guaranteed letter
  - Get an 87? Guaranteed at least B+
- I may adjust the thresholds slightly
  - Completely at my discretion
- Two reasons I typically might do this
  - Clustering of grades shows different break
    - 90.2, 90.15, 90.1, 90.06, 90.01, 89.9, 88
  - More accurate reflection of effort/knowledge
    - Hard work, class attendance...

Assignments

- Homework [30%]
- Class Attendance/Quizzes [10%]
  - 50% being present in class
  - 50% checkup questions
- Midterm Exam [25%]
  - Nov 1, in recitation time
- Final Exam [35%]
  - According to registrar’s final exam schedule

Reading

- Text:
  - Computer Organization & Design (Patterson & Hennessy)
  - You are expected to complete the assigned readings
  - Some material on the CD (e.g., Appendix)
  - Note: get revised 4th edition, but not any of the unusual variants (e.g., ARM version)
- We are going to skip around a bit relative to the book
  - Digital logic earlier/more focus on it
  - Pipelining later, less focus
- Read
  - Start reading Chapter 1 and Appendix C now

Late Policy

- Late Policy
  - 5 days per group total for the semester
    - Does not change demo deadline, only submission deadline
  - Days, not classes
  - Used in entire days: 10 min late = on next day
  - After used up: must turn in on time
    - No credit for late work after this
  - Extenuating circumstances: talk to me
    - E.g., serious injury/illness, family emergency...
  - This course takes time: start early!
    - Debugging VHDL can take a while

Lecture

- You all will get more out of this if you participate than if I just talk for 75 minutes
  - Please ask questions, discuss things you are unclear on, etc..
  - I will ask you all to answer questions
    - Don’t be afraid of this, I’ll ask everyone
    - If you are wrong, its not the end of the world, we’ll stop and make sure you get it
  - Typically, I’ll work my way around the room, so nobody will be surprised that they are next to be called on

Class Attendance/Quizzes

- I expect you to attend class...
  - But understand that sometimes things come up and you need to miss
- We will have 8 “quizzes” throughout the semester in lecture
  - You get 50% for being there, 50% for correct answers
  - Drop 2 lowest quiz grades (count 6 out of 8)
  - Questions mostly check if paying attention/understanding discussion
    - Recommend asking questions if you are unclear on things
    - Reading in advance of lecture
    - May try out a few different quiz formats
  - Drop quizzes: account for needing to miss lecture
  - Long term circumstances, please talk to me
Homeworks

- Homeworks: 5 of them in the semester
  - Work in groups of 2 or 3
  - Fixed for semester once formed
  - Exceptional circumstances/dysfunctional group: talk to me
  - Do not work alone:
    - First assignment may be easy... but will want a group later
  - End of recitation this week: a few minutes to form groups
- "Demo" portion of homework
  - "Question and Answer" may be better term
  - TAs will ask each person questions about project: accountability
  - ALL group members MUST know how it all works
    - TAs will ask a particular group member a question
    - "I don't know, John did that part" will lose points

Academic Integrity

- Academic Integrity Expectations
  - I take academic integrity VERY seriously, and you should too
  - Basic principles for Duke in general:
    - I will not lie, cheat, or steal in my academic endeavors, nor will I accept the actions of those who do.
    - I will conduct myself responsibly and honorably in all my activities as a Duke student.
  - If I suspect academic misconduct in my class...
    - Reported to the appropriate Associate Dean
    - Due process to determine if you did commit academic misconduct
    - If found responsible,
      - I will give you a 0 on the assignment
      - Appropriate Associate Dean may apply additional penalties

Academic Integrity: Class work

- Classwork exercises: practice and learn
  - Do your work... but feel free to get help
  - I expect you to be the one typing your answer
    - Not copied/downloaded
  - I expect you to understand your submission
    - Should be able to explain how it works
    - Be able to do it, or similar problems

Academic Integrity: Homework

- Project: You + your group
  - Should not be getting help from other groups, students
  - Can ask TAs + me for help
  - Not many external resources you should be using

Academic Integrity: Mini-Quizzes

- Mini-Quizzes: Individual Effort
  - Open Notes
    - Can use paper resources
    - Will allow electronic reading of course lecture slides
      - But may not use laptop/tablet for anything else
      - If you use an electronic device during the quiz, I'd better see the course lecture slides on it
    - May not discuss with classmates
  - Closed Book

Academic Integrity: Exams

- Exams in this class are individual effort
  - Allowed 1 page of notes
  - Closed book
  - No electronic/interactive/human resources
  - Before I return your exams, I will photocopy and keep a random subset
  - If you request a re-grade, I will compare your solutions to my photocopy. If they have been changed, I will report the incident directly.

- Related exam policies:
  - Questions? Raise hand, TA or I will come to you (don't get up)
  - Need restroom? Raise hand, we will let you go one at a time
  - No calculators/smart phones: too easy to use to chat
**Academic Integrity: General**

- Some general guidelines
  - If you don’t know if something is OK, please ask me.
  - If you think “I don’t want to ask, you will probably say no” that is a good sign its NOT acceptable.
  - If you do something wrong, and regret it, please come forward—I recognize the value and learning benefit of admitting your mistakes. (Note: this does NOT mean there will be no consequences if you come forward).
  - If you are aware of someone else’s misconduct, you should report it to me or another appropriate authority.
- Within your homework group, this becomes even stronger: if you are aware that one of your group members has committed misconduct on a homework submission for your group, you are complicit in it if you do not report it.

**Course Problems**

- Can’t make midterms / final, other conflicts
  - Tell us early and we will schedule alternate time
- Struggling in course
  - Come see me: I’m here to help
- Other problems:
  - Feel free to talk to me, I’m generally understanding and will try to work with you
  - Some problems may extend well beyond my course
    - Academic Advisor
    - DGS Team

**Resources**

- Piazza
  - Discussions, questions, etc
  - Good place to discuss lecture videos, ask questions
  - Announcements I make: required reading
  - Other discussions: strongly recommended reading
- Assignment submission
  - TBD
- Course Web Page

**Recitation**

- Recitation: Fridays
  - TA: John O’Hollaren
- This week:
  - VHDL
  - Using Quartus
  - Simulation in ModelSim

**Wisdom From Prior Courses**

- Some things I have learned from prior students
  - Cultural barriers to asking for help:
    - No question is “too simple” to ask me
    - Office hours are not just for the advanced students to ask complicated questions
    - You will not offend me if you ask me questions/tell me you do not understand
    - You are not “bothering me” if you come to office hours
      - That is why I have them: to help you!
    - Applies very broadly: I’m here to teach you and help you!
  - Doing badly -> “study harder”
    - Not the answer!
    - Work smarter, not harder.
    - Studying may not help at all!
    - Struggling? Seek my help.

**Wisdom Continued**

- Practice Exams
  - I post a practice midterm and a practice final... With solutions!
  - Use them!... and use them well...
  - I’ve heard the following:
    - “I tried the practice exam, and didn’t do well... but just hoped the real exam would be easier”
    - “I skimmed the practice exam and figured I could do the questions if I tried”
    - “I started with the solutions, and they all made sense, so I figured I would do fine.”
    - “I didn’t have time to try the practice exam. I was too busy studying [for this class].”
  - Take the practice exam, like a real exam (time constraints too!)
  - Check your answers after you finish.
Processors are made of **transistors**
- Electrically controlled switches (more on this later)
- Anyone have any idea how many transistors are in a modern chip?

**2.6 BILLION**

How do you put together 2.6 Billion of anything
- ...and make sure the product works right?
- ...in every corner case
- ...and is really fast
- ...and do it within a reasonable budget/timeframe?

More fundamentally, how do you engineer any large system?

---

### Abstraction: The Key to Computer

- Abstraction: Divide interface from implementation
  - Interface: how it's used
  - Implementation: how it does it
- Build larger components from smaller ones
  - Larger ones use interface of smaller ones to do tasks
  - Don't care about implementation
- Tasks can be split between engineers:
  - You make a piece that does xyz, and I'll use it to do my job
- Components can be re-used
  - Also good: making them generic, so they can be re-used more

---

### Other key to engineering: tools

- Processors designed in Hardware Design Languages
  - Verilog
  - VHDL
  - Learn one: you can pick the other easily

- You don't layout ever transistor by hand...
  - Instead you write a description of the hardware in an HDL
  - ...a lot like a programming language...
  - ...then run it through synthesis tools
- We'll use VHDL and Quartus
  - With ModelSim to simulate
  - Friday's Recitation

---

### Levels of Abstraction

- Transistors: "electrical switch"
  - Can go lower (those with EE background have)
  - ...but no need for us
- Gates: a few transistors
  - Implement logical functions: And, Or, Nor, Xor
- Meaningful logic elements: a handful of gates
  - Combine into meaningful elements: muxes, 1-bit adders, flip-flops
  - May build larger items: N-bit adders from 1-bit adders
- Large elements (stages, units): combining logic elements
- Core
- Chip: now with multiple cores
A Different Kind of Abstraction

- Previous discussion: abstraction to build a processor
- Also: abstraction to use a processor
  - How/why?

A Different Kind of Abstraction

- Previous discussion: abstraction to build a processor
- Also: abstraction to use a processor
  - How/why?
  - Need software that can use the processor
  - Software should not rely on HOW processor is implemented
  - Abstraction between hardware implementation and interface
    - Interface = ISA = contract between hardware and software
    - Implementation: can vary from generation to generation
    - Consider x86
      - Can take a program written for an i386 (1985)
      - ...and run it on a modern core in 2012

Wrap up there for the day

- We’ll end there for the day
- Remaining time:
  - Any questions I need to take individually

- Next time:
  - Digital Logic: transistors to gates
  - Start reading Ch1 and Appendix C