VLSI Design

EEC 116

Lecture 1

Bevan M. Baas
Thursday, September 27, 2018
Today

- Administrative items
  - Syllabus and course overview
  - Course objective and strategies
- My background
- Chapter 1
- Read Chapter 1
- Homework 1 posted on web page; due next Thursday
Course Communication

• Email list
  – Urgent announcements

• Web page
  – Primary source of course information

• Office hours
  – Tentatively:
    • Tue 4:30 after lecture
    • Mon 2-3pm
    • Th 4:30 after lecture

• Please see me (or TA) in person with questions rather than email
Teaching Assistants

- Tim Andreas
- TBD
Course Workload

• 4 unit course
• Upper division
• This course requires significant effort and time
  – Circuits
  – Layout
  – Tools
    • Magic
    • Irsim
  – Major project with report
Lectures

• Ask questions at any time
  – Please raise your hand
• Be respectful of others
  – Hold conversations outside of class
  – Silence phones
  – Sit in the back if you come in late or need to leave early
My Teaching Philosophy

• Primary goal (mine and yours):
  
  \textit{Learn VLSI design well}

• Achieve this through:
  – Reading textbook
  – Lectures
  – Solving problems on paper
  – Solving problems in lab
  – Discussions with other students, TA, myself
    • Come to office hours
My Grading Philosophy

- Grading serves two main purposes:
  - 1. Motivate you to do the work required to learn
  - 2. Give others an indication of how well you know the material
    - Requires honest work and fair grading
I assign a letter grade only for the final course grade

You can see score statistics for each graded item on Canvas

I look at the final exams and course record of the class and assign two key dividing points: the A/A+ and D+/C- boundaries, and assign course grades from there using equally-sized intervals

- No required numbers of any particular letter grades
- Absolute scores are not important; the boundaries shift according to the difficulty of the exams in any quarter
- Ignore any letter grades you might see on canvas

(not actual grade data)
Working With Others

• Collaboration
  – Asking questions and explaining principles produces better work and dramatically increases learning
  – Working with others
    • Do homework and prelabs with classmates nearby
    • Ask each other questions, help each other—regarding principles, and general approaches to solving only

• Final Project
  – Groups of 2

• Dishonesty
  – Copying produces similar work, stunts learning, is not fair to honest students, and is not allowed in this course
    • Students engaged in dishonest work will be referred to Student Judicial Affairs
    • I will try to keep in-class exams honest
    • Steps will be taken to keep out of class work honest
Penalties for Violating the *Policy on Student Conduct and Discipline*

- **Penalties**
  - Minimum penalty: meetings with SJA officer, zero grade on work, record with SJA
  - One to three quarter suspension from the university
  - Permanent dismissal from all ten campuses of the University of California. Permanent notation on your transcript.

- The purpose of the penalties and me mentioning them is so that no one will get a penalty!!! Don’t do anything that violates the Policy on Student Conduct!
Penalties for Violating the Policy on Student Conduct and Discipline

• Typical scenario:
  – Someone shares code/design with another
  – They get caught
  – The “Copier” feels terrible guilt for causing a friend to get a zero
  – The “Sharer” deeply regrets sharing resulting in a zero when he/she should have had a full score
Exam and Quiz Regrades

• Some number of exams and quizzes will be scanned before being returned

• Key take-away messages:
  – Do not change anything on your work if you request a regrade
  – One student did last quarter and got in big BIG trouble!!!
Cheating Websites chegg, coursehero, etc.

- Key take-away messages:
  - Do not post assignments
  - Of course do not use any unpermitted outside material in work you submit
  - Of course do not post solutions
  - Two students did last quarter and got caught!!!
Course Announcements

- Most announcements via email by myself or TAs
- Announcements normally not posted but they may if email volume becomes too large
Course Web Page

Colored pencils

- Buy colored pencils or pens whose colors match *magic layout tool layer colors*
  - green
  - brown (orange next closest?)
  - red
  - blue
  - purple
- Used for “stick diagrams”
- Slightly transparent pencils or pens work best
Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten
New plot and data collected for 2010-2015 by K. Rupp
New data added by B. Baas
Number of Processors on a Single Die vs. Year

Note: Each processor capable of independent program execution
Processor Eras

- **Transistor Era:** the Intel 4004 was the first commercial single-chip microprocessor and it contained 2300 hand-drawn transistors.

- **Single/Multi-Processor Era:** focus on components of single processors and multi-processors, which generally scale well to only small numbers of processors.

- **1000-Processor Era:** focus on making systems scalable and working with processors as building blocks. The 32 nm 1000-processor KiloCore chip would contain approximately 2300-3700 processors if its area were the same as a 32 nm Intel Core i7 processor, or 11,000 processors if its area were the same as an Nvidia GP100.
Critical Challenges Facing Industry

- Energy Efficiency
- Performance
- Software development cost and time
- Hardware development cost and time
- Opportunity: Critical workloads sometimes/frequently have relatively simple tasks as critical kernels (e.g., machine learning, digital signal processing, multimedia, data record processing, pattern matching, etc.)
  - Embedded (e.g., IoT)
  - Mobile
  - Datacenter
Undergraduate Research

• Talk to me if you are interested!
• I will say that there is a very strong correlation with GPA and success in research
Future Applications

• Very limited power budgets
• Require significant digital signal processing
Some Points on Course Coverage

- Study CMOS
- Focus on digital circuits rather than analog circuits
- Emphasis on low-level design (full custom layout, circuits). 180B is higher-level
- Emphasis on VLSI-specific issues
- Limited coverage of digital circuits such as what is covered in 118
EEC 116 Outline

- Introduction
- Cost, yield
- CMOS fabrication
- CMOS VLSI layout
- Inverter characteristics
- MOS resistance, capacitance
- Sequential circuits
- Optimizing performance
- Complex combinational gates
- Logic circuit styles
- Wires
- Chip-level structures
- Chip-level issues
- Array memories
- Packaging
- Standard cell P&R overview
Chapter 1—Introduction

- History of computing
- What is inside a processor
- How they are designed
The First Computer

The Babbage Difference Engine (1832)

25,000 parts
cost: £17,470

Source: Digital Integrated Circuits, 2nd ©
ENIAC - The first electronic computer (1946)

Source: Digital Integrated Circuits, 2nd ©
What Is A *Processor*?

- Something that processes!

Example applications:
- Microwave oven controller
- CD or DVD player
- Hearing aid
- Personal computer
- Digital watch
- Cell phone
- Radar for an airplane
- Wireless network processor
What Is Inside One?

- Three main components
  - Memory
What Is Inside One?

• Three main components
  – Memory
  – Datapath
What Is Inside One?

- Three main components
  - Memory
  - Datapath
  - Control
Memory

• Stores information
• Examples
  – Temporary chip memory
    • “256 MB RAM”
  – Permanent chip memory
    • “Flash non-volatile memory”
  – Hard disk
    • “80 GB disk”
Datapath

- Processes information
- Example tasks
  - Add
  - Multiply
  - Move
  - Compare
Control

- Runs the show
- Examples
  - Software
    - Word, Excel
    - Large database
  - Hardware (best for simpler, fixed, high-speed tasks)
    - MP3 player
    - Signal processing
Simple Blocks Working Together
Example #1

- Auto cruise control
  - Watch car’s speed (*input*)
  - Compare with set speed (*datapath, compare*)
  - If adjustment is needed, add or subtract (*datapath*) from setting and send update to gas pedal (*output*)
Simple Blocks Working Together
Example #2

• Google search
  – Type in search words (*input*)
  – Look through large database of web pages (*memory*) until matches are found (*datapath, compare*)
  – When list is found, send it to user (*output*)
Simple Blocks Working Together

• Key point
  – Simple operations alone can not do much, but billions of simple operations per second can do a lot!