ECE 3056: Architecture, Concurrency and Energy in Computation

Sudhakar Yalamanchili
School of Electrical and Computer Engineering
Georgia Institute of Technology

The World Today

Warehouse Scale Computers

SUN MD S20: Water cooled containers 187.5KW

Google data center in Oregon
Power densities of 10-20 kW/m² → footprint → cost


Inside the Processor

• AMD Barcelona: 4 processor cores
Combination of compute cores, memory and I/O devices

Moore’s Law
Moore’s Law: Consequences

**Goal:** Sustain Performance Scaling

**Metric:** Ops/sec

---

**New Rules: The End of Dennard Scaling**

- Voltage is no longer scaling at the same rate
  - Feature size
- Slower scaling in power per transistor → increasing power densities

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

\[
\times 30 \quad \rightarrow \quad \times 1000
\]

The Power Wall
The Rise of Accelerators

CPUs are transitioning to System on Chip (SoC) Designs

Multiple Instruction Set Architectures (ISA)

Memory Wall

"Moore’s Law for Logic vs. Memory"

Processor-Memory Performance Gap: (grows 50% / year)

Metrics:
- Bandwidth: bytes/sec
- Latency: secs

100’s of ns
Course Objectives

• Core concepts of microprocessor architecture
  ❖ Instruction Set Architecture (built on ECE 2020 and 2035 concepts)
  ❖ The Core: ALU, datapath, & control implementation
  ❖ Multiple Cores: synchronization & communication
  ❖ Memory hierarchy: cache hierarchy, virtual memory
  ❖ Input/output systems

❖ Sources of Energy and Power dissipation
  o Microarchitecture level models of energy and power

❖ Basics of Operating Systems
  o Processes, scheduling & concurrency support
  o Supporting exceptions, VM, I/O

---

(11)

Course Objectives

• Understand how computing systems work
  ❖ Fundamental architectural concepts and challenges
  ❖ Understanding performance: time, space, and energy

• Acquire knowledge to optimize systems
  ❖ As a designer
  ❖ As a user
  ❖ As a researcher

• Assignments
  ❖ Extend simple processor models and apply several hardware optimizations
  ❖ Build and simulate memory models
  ❖ Basic operating system concepts and hardware support
  ❖ Evaluate energy and power consumption
What You Will Learn

• How programs are translated into the machine language
  ❖ And how the hardware executes them

• The hardware/software interface

• What determines program performance
  ❖ And how it can be improved

• How hardware designers improve performance

• What is parallel processing?

• Where does the energy go?

Course Information

• Web page: http://ece3056-sy.ece.gatech.edu/spring2019/

• Prerequisite: ECE2031, ECE 2035

• Textbooks
    - On-line text as an alternative to the hard copy: Zybooks (check announcement from Canvas)
  ❖ Online OS text: http://pages.cs.wisc.edu/~remzi/OSTEP/
  ❖ Online, supplemental content for the text http://booksite.elsevier.com/9780124077263/?ISBN=9780124077263
Grading Policy

• Homework Assignments: 20%
   Individual work, no collaboration other than specified!
   Late Assignment Policy: TBD
   You must make a passing grade (>50%) on the assignments to pass the course!

• Exams
   2 in-class exams: 50% (25% each, February 20th and April 3rd)
   Final: 30%; Friday, April 26th, 2:40 – 5:30 pm

• Teaching Assistants: check the class webpage

• Weekly Help Sessions: check the class webpage for location and timing

Grading Questions

• Provide the TA with your questions in writing, noting any specific disagreements and request for correction and/or regrading

• If you do not agree with the result, submit the request to me and see me.

• Please do not engage the TAs in discussions of grading disagreements. The course instructor is responsible for resolving any such issues.
Where Are We?

- **High-level language**
  - Level of abstraction closer to problem domain
  - Provides for productivity and portability

- **Assembly language**
  - Textual representation of instructions

- **Hardware representation**
  - Binary digits (bits)
  - Encoded instructions and data

Below Your Program

- **Application software**
  - Written in high-level language

- **System software**
  - **Compiler**: translates HLL code to machine code
  - **Operating System**: service code
    - Linker/loader
    - Handling input/output
    - Managing memory and storage
    - Scheduling tasks & sharing resources

- **Hardware**
  - Processor, memory, I/O controllers
Simulators

- Simulator for the MIPS assembly language: QtSpim
- Simulator for VHDL models: ModelSim
- Check the Class Resources link from the class webpage for links to candidate simulators (the ones I will use)

http://ece3056-sy.ece.gatech.edu/spring2019/

* You may use alternative simulators – just make sure they are compliant with the MIPS and VHDL standards.

Course Layout

- Instruction set architecture
- Basic Datapath
- Pipelined Datapath
  - Chapter 2, 3, 4, Appendix A, D
- Cache design
- Memory Hierarchy
- Virtual Memory System
- Chapter 5
- Parallelism & Concurrency
- Data, Thread, and Instruction Level Parallelism
  - Chapter 6, Appendix C
- Storage Technologies
- I/O Systems
  - Chapter 5.11, 6.9
- Operating System Basics
- Online text

(Intel Ivy Bridge)
Getting A Good Grade

• Attend class and ask questions in and out of class!

• Follow reading assignments

• Start early on homework assignments

• Listen in class – pay attention to suggestions

• Regular study habits: work the practice problems. Use the Glossary as a study guide

• Pay attention to the study guide

• Worry about why in addition to what

• Make use of the TA, Piazza, my office hours, and email

Getting a Poor Grade

• Miss class

• Go weeks without studying or following up on lecture material

• Cram for tests

• Late start on homework assignments

• Ignore reading assignments and practice problems

• Rely on others to tutor you before the exams

• Do not get software installed in time

• Ignore gaps in your understanding

• Pay poor attention to homework assignments
Let's Get Started.........