Introduction

CS 3410: Computer System Organization and Programming

[K. Bala, A. Bracy, E. Sirer, Z. Susag, and H. Weatherspoon]

Cornell Bowers CIS
Computer Science

WELCOME BACK!

How excited are you to take this class??

0

(A) I've been waiting my whole life to take 3410. I couldn't sleep last night I was so excited.

0%

(B) I'm excited.

0%

(C) I've heard good things, but my excitement is on hold.

0%

(D) Excited, not sure. Anxious? Yes.

0%

(E) Help! I'm a CS minor trapped in this class. Please rescue me. (Seriously.)

0%

What's your enrollment status?

0

I'm enrolled and in the lab of my choice.

0%

I'm enrolled but urgently need to change my lab.

0%

I'm enrolled but would prefer a different lab.

0%

I'm not yet scheduled in the class and could fit multiple labs in my schedule.

0%

I'm not yet enrolled in the class and can only fit a single lab in my schedule.

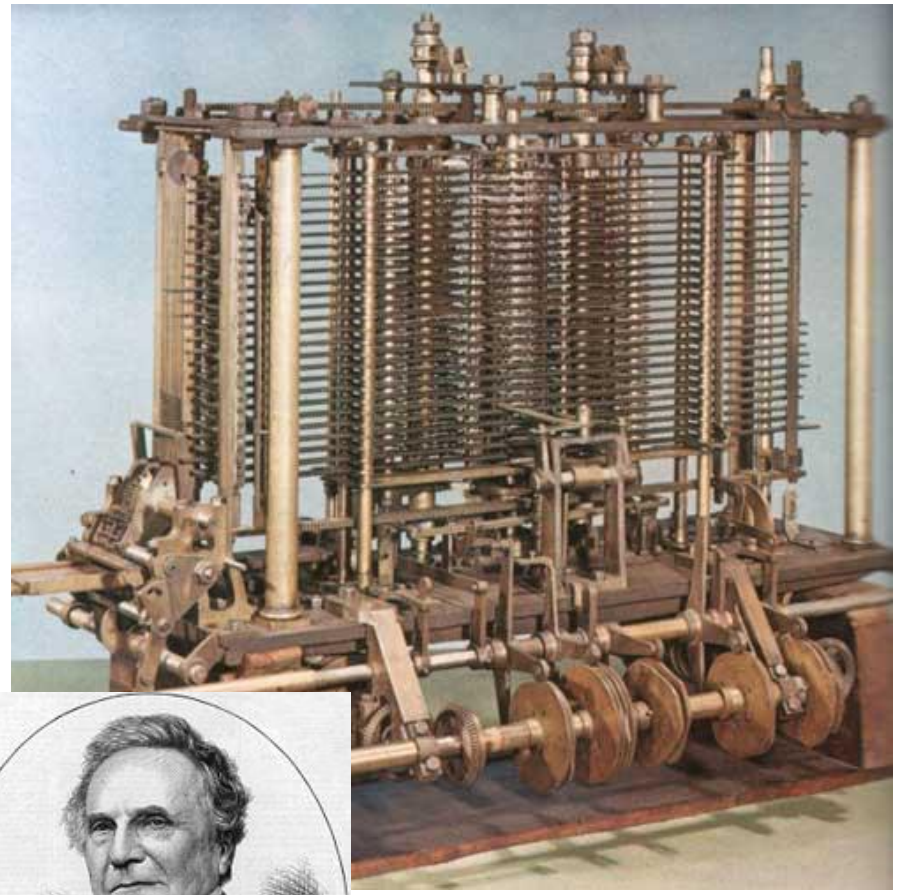
0%

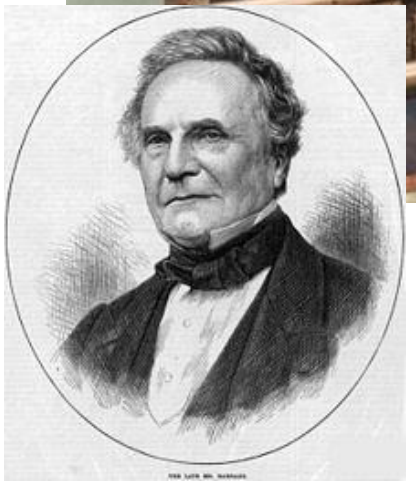
Lecture Basics

- Tu Th 10:10-11:25
- **Poll Everywhere:** Every Lecture
  - missing a few times is okay
- **No cell phones, except Poll Everywhere**
- **Laptops only in laptop zone**
- See Syllabus for policies and more information

The Analytical Engine

- Designed by Charles Babbage from 1834 – 1871
- Considered to be the first digital computer
- Built from mechanical gears, where each gear represented a discrete value (0-9)
- Babbage died before it was finished

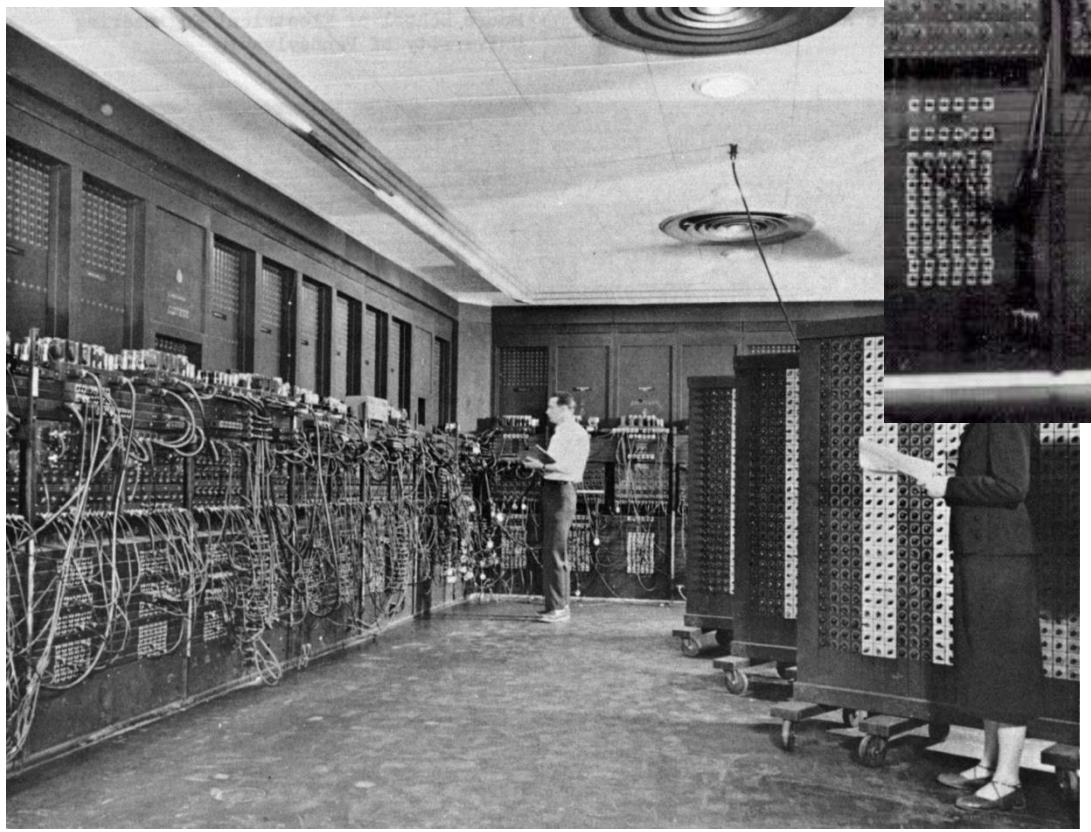


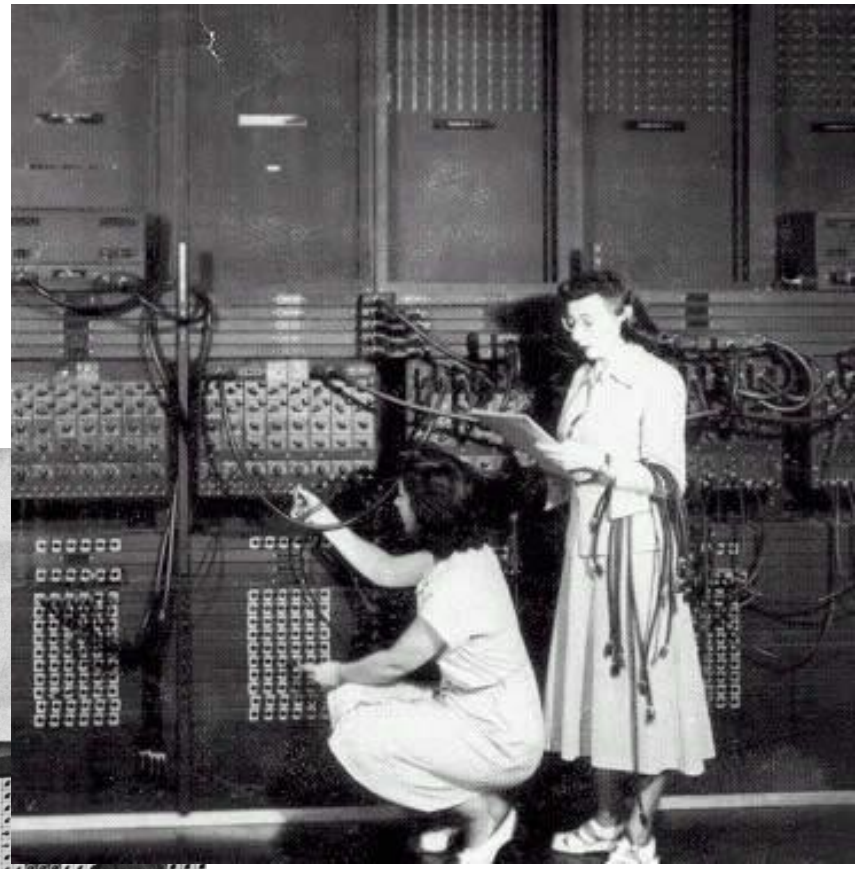


http://history-computer.com
http://wikimedia.com

ENIAC

Electronic Numerical Integrator And Computer





1946

John Mauchly

J. Presper Eckert

IBM 7090

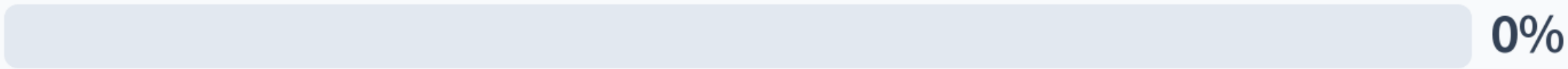
Human Computers programming the IBM 7090

Mary Jackson

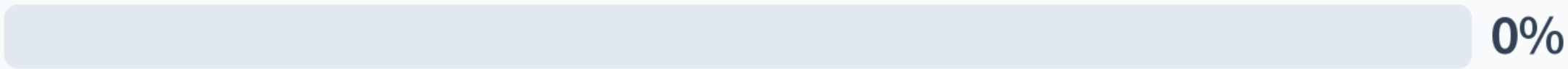
Hidden Figures

How many people do you know you are taking this class?
("Know" means if you saw their face you would know their name.)

0



1



2-5

SEE MORE

Who are you?

0

Freshman

0%

Sophomore

0%

Junior

0%

Senior

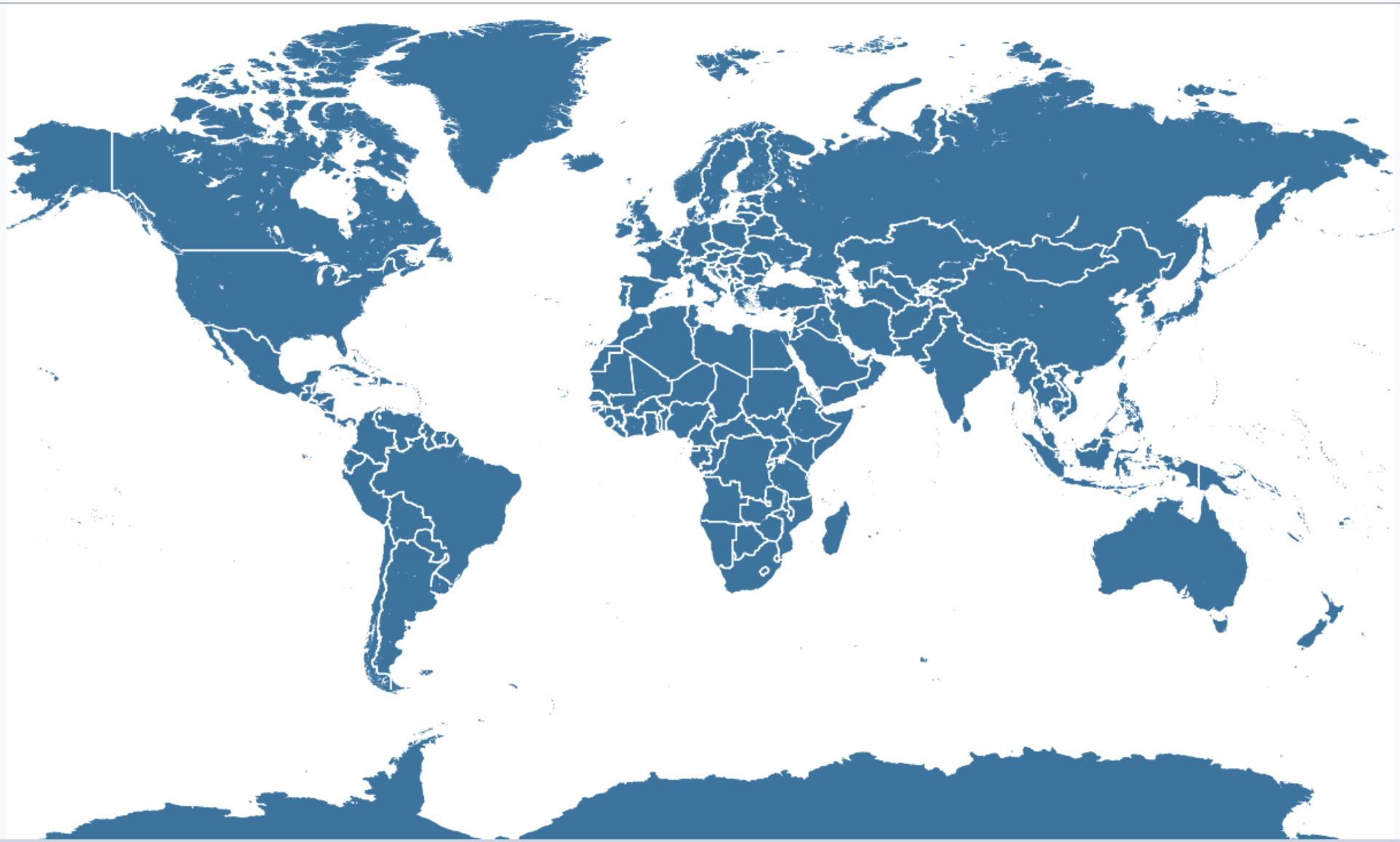
0%

Other

SEE MORE

Where are you from?

0



Who am I?

Hakim Weatherspoon

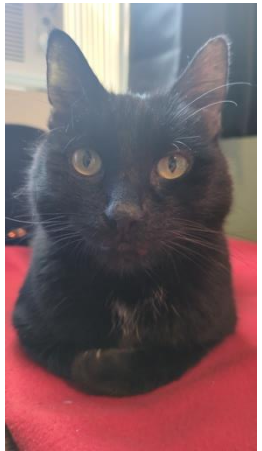
- Undergrad: Computer Engineering at University of Washington
- PhD: Computer Science, Distributed Systems at University of California, Berkeley
- Academia: Cornell
  - Taught 3410 and 4410 more than 15 times for nearly 20 years!
- CEO -> Chief Scientist of a cloud-based startup company,
- Husband, father of four

exostellar

Who am I?

Zach Susag

- **Undergrad**: Computer Science and Mathematics at Grinnell College
- **PhD *(ongoing)***
  - Started at University of Wisconsin – Madison
  - Programming Languages w/ Prof. Justin Hsu
- **Research**: Verification of probabilistic programs (esp. differential privacy mechanisms)
- Hope to be a future teaching professor!
- Cat dad, DIYer, cook



Who are you?

"Sometimes it is the people that no one imagines anything of who do the things that no one can imagine."

– Alan Turing

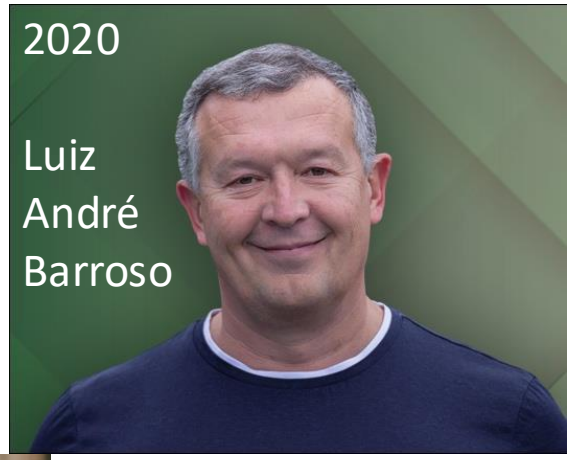
Turing Award Winners?

2017

“Sometimes it is the people that no one imagines anything of who do the things that no one can imagine.”

– Alan Turing

Eckert Mauchly Award Winners?



Course Objective

- Understand the HW / SW interface
  - How a processor works
  - How a computer is organized
- Establish a foundation for building applications
  - How to write a good program
    - Good = correct, fast, and secure
  - How to understand where the world is going
- Understand technology (past, present, future)

What is this?

```
#include <stdio.h>

int main() {
  printf("Hello world!\n");
  return 0;
}
```

How does it work?

I'm glad you asked…

15 weeks later and you'll know!

"I know Kung Fu."

C

```
int x = 10;
x = 2 * x + 15;
```

compiler

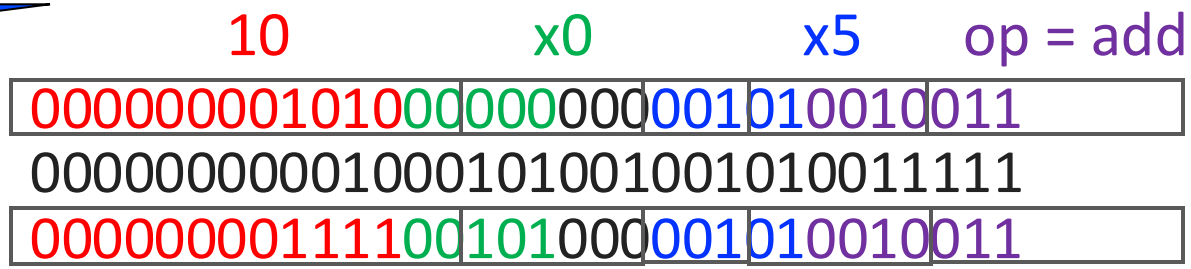
RISC-V assembly language

```
addi   x5, x0, 10      x5 = x0 + 10
muli   x5, x5, 2       x5 = x5 * 2
addi   x5, x5, 15      x5 = x5 + 15
```

x0 = 0

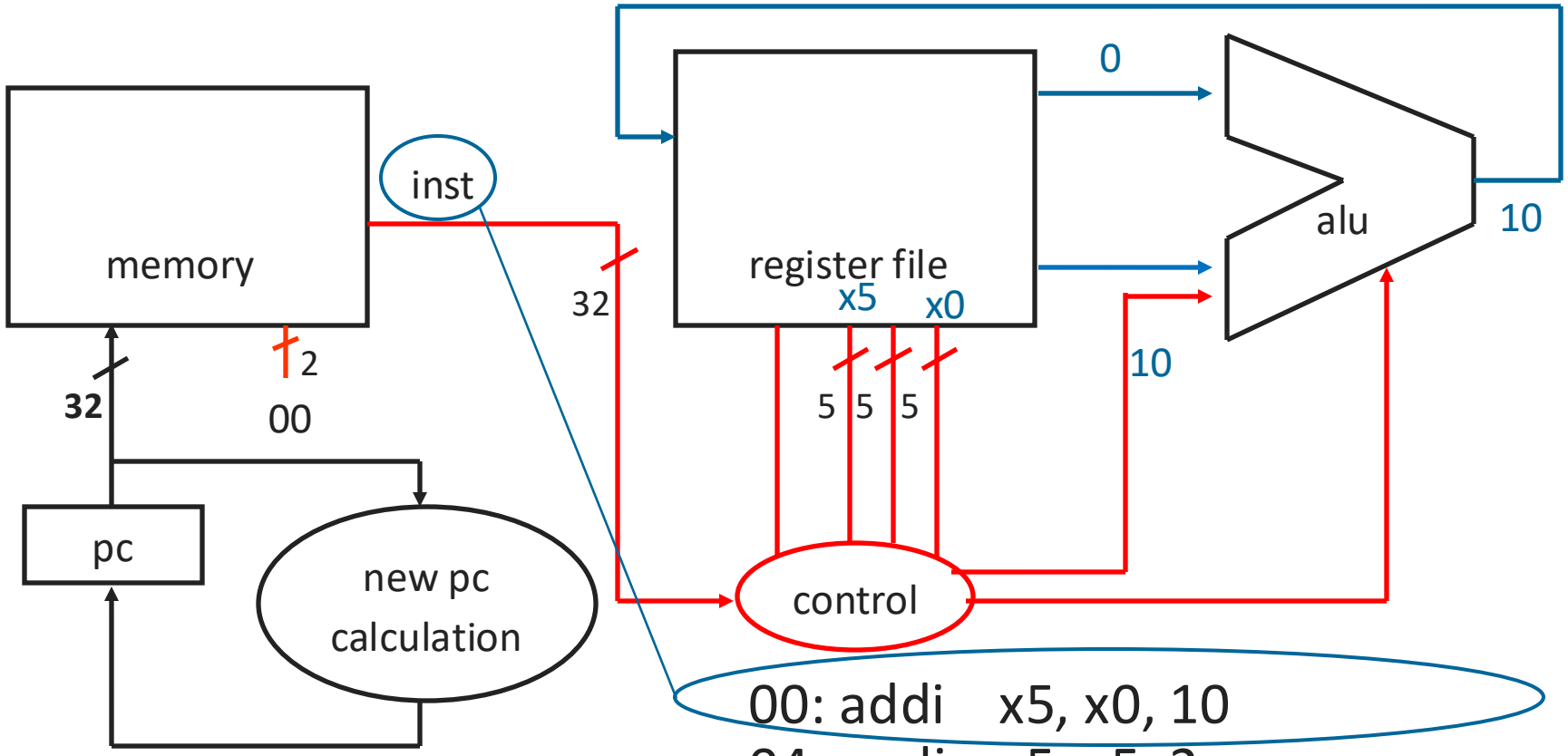
assembler

RISC-V machine language



EVERYTHING IS A NUMBER!

How to Design a Simple Processor

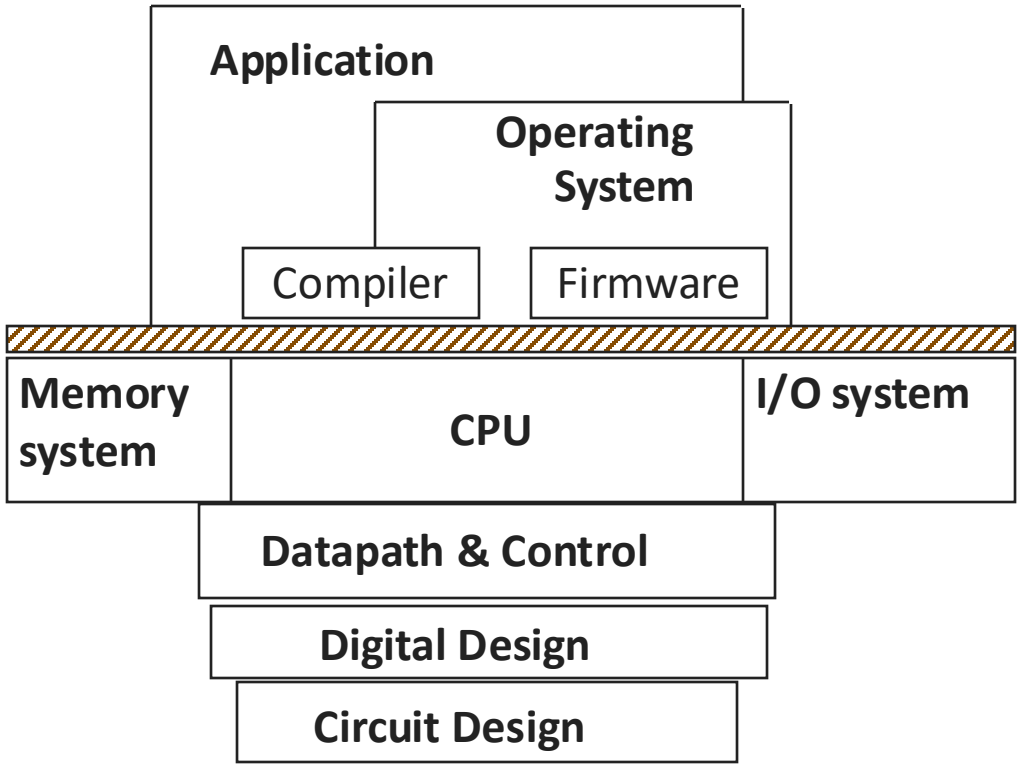


Instruction Set Architecture (ISA)

- abstract interface between hardware and the lowest level software

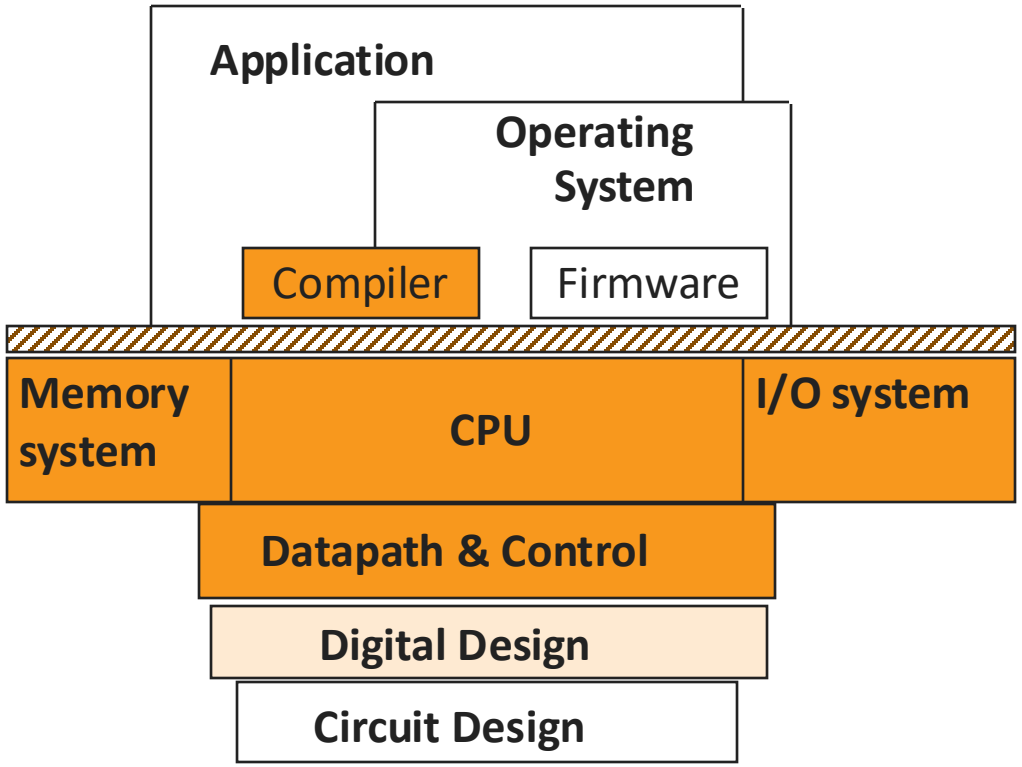
- user portion of the instruction set plus the operating system interfaces used by application programmers

Overview



Instruction Set Architecture

Covered in this course



Instruction Set Architecture

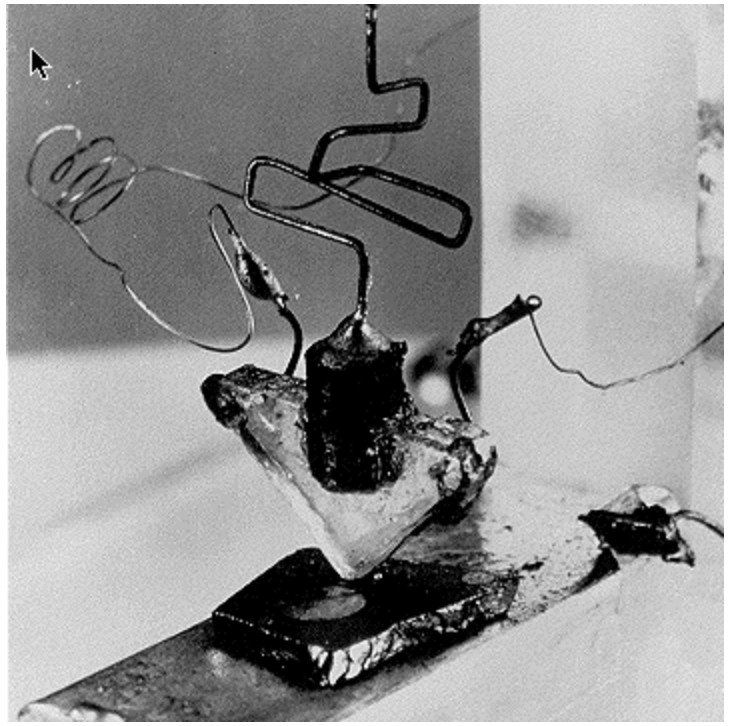
Where did it begin?

Electrical Switch

- On/Off
- Binary

Transistor

- Bardeen, Brattain, and Schokley



The first transistor on a workbench at AT&T Bell Labs in 1947

Moore's Law

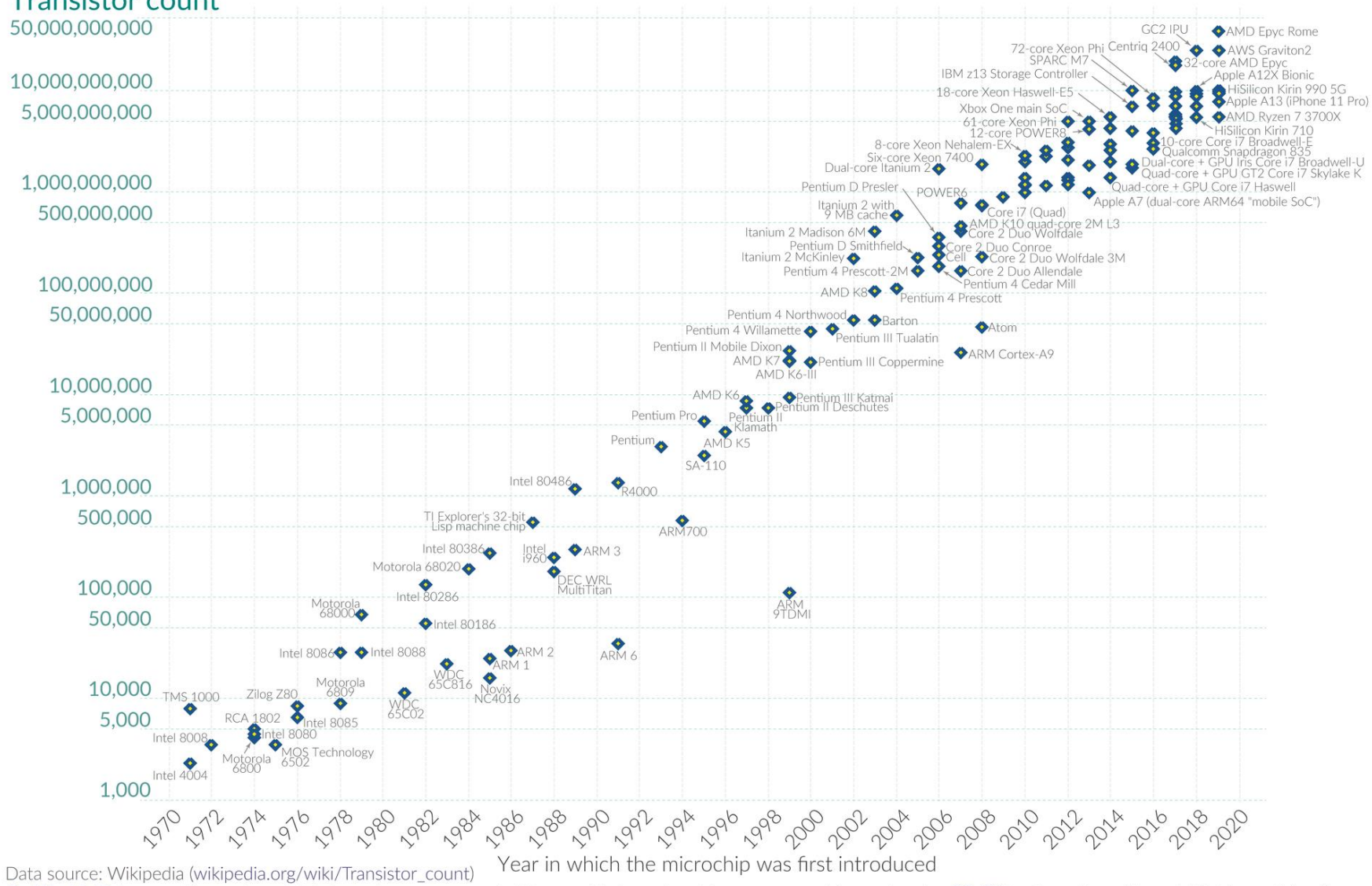
- 1965
  - # of transistors integrated on a die doubles every 18-24 months (*i.e.,* grows exponentially with time)

- Amazingly visionary
  - 2300 transistors, 1 MHz clock (Intel 4004) - 1971
  - 16 Million transistors (Ultra Sparc III)
  - 42 Million transistors, 2 GHz clock (Intel Xeon) – 2001
  - 55 Million transistors, 3 GHz, 130nm technology, 250mm^2 die (Intel Pentium 4) – 2004
  - 290+ Million transistors, 3 GHz (Intel Core 2 Duo) – 2007
  - 721 Million transistors, 2 GHz (Nehalem) - 2009
  - 1.4 Billion transistors, 3.4 GHz Intel Haswell (Quad core) – 2013
  - 20 Billion transistors, 3.49 GHz Apple M2 (8 core) — 2022
  - 28 Billion transistors, 4.4 GHz Apple M4 (16x core) — 2024

Moore's Law: The number of transistors on microchips doubles every two years

Our World in Data

Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years.
This advancement is important for other aspects of technological progress in computing – such as processing speed or the price of computers.



Data source: Wikipedia (wikipedia.org/wiki/Transistor_count)
OurWorldinData.org – Research and data to make progress against the world's largest problems.

What kind of machine is your laptop?

0

Pre-fab x86 (Dell, Lenovo, etc.)

0%

New(er) Mac Pro (M1, M2, M3, M4)

0%

Older Mac or PC

0%

Custom homemade...

0%

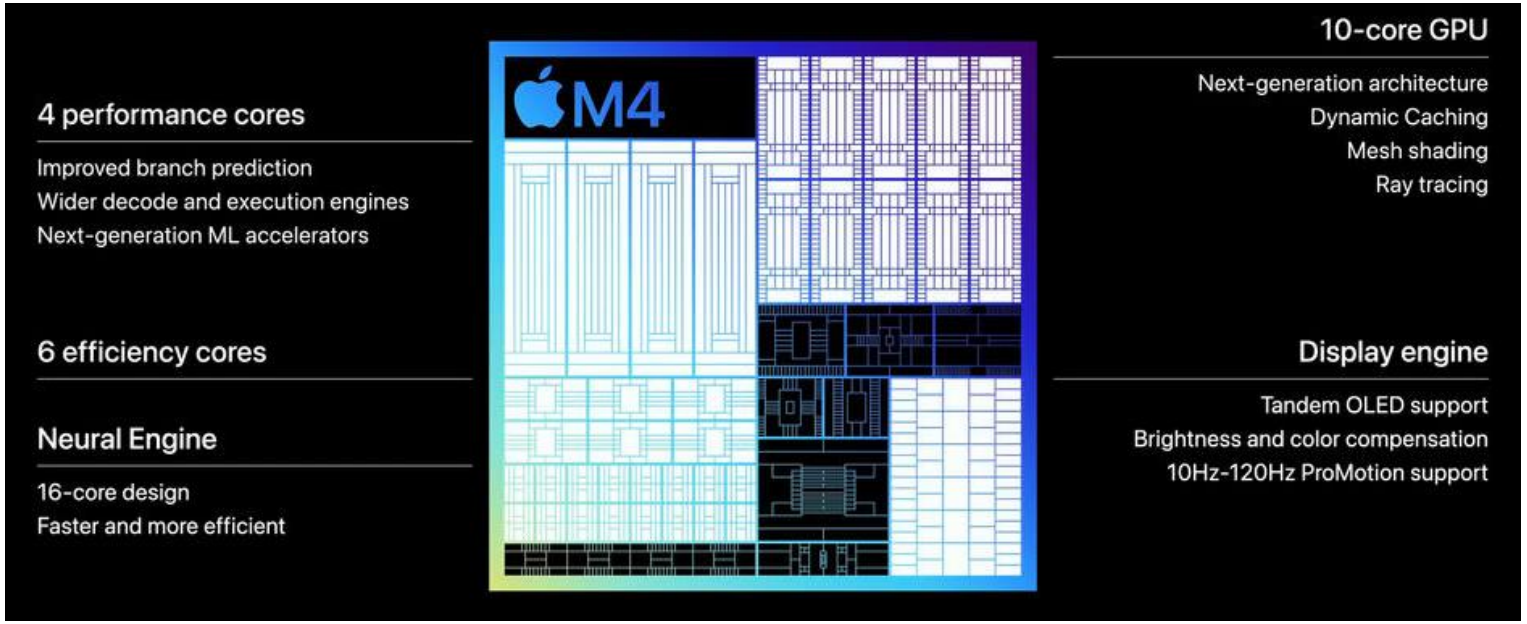
I have a device. This is all I know

SEE MORE

0%

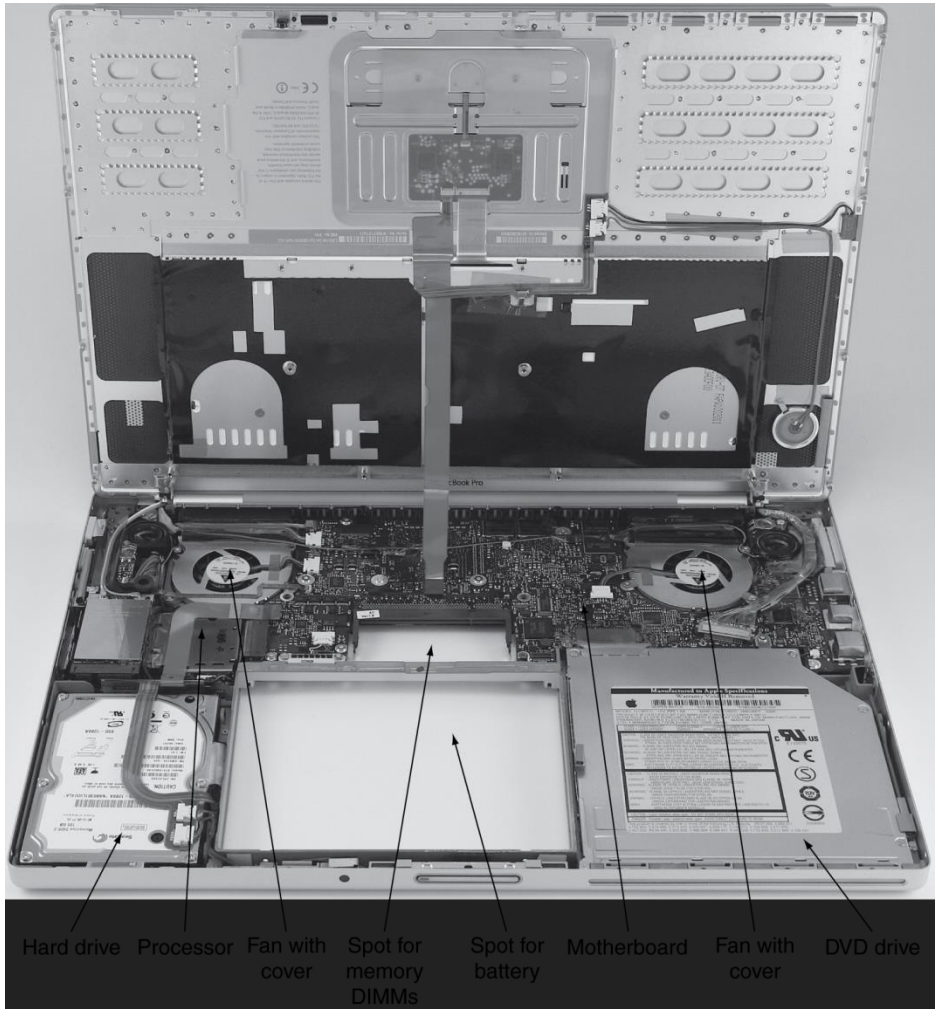
What to do with all these transistors?

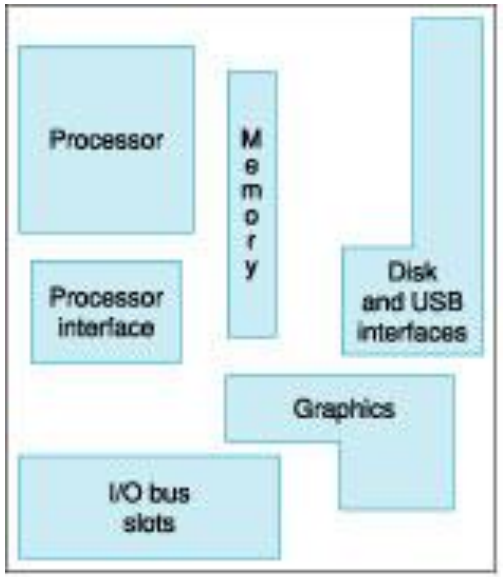
- Apple M4
  - 28 billion transistors, 3nm
  - 177 square millimeters
  - 4x-10x performance, 4x-6x efficiency, 8x-40x GPU, 16x Neural processing cores

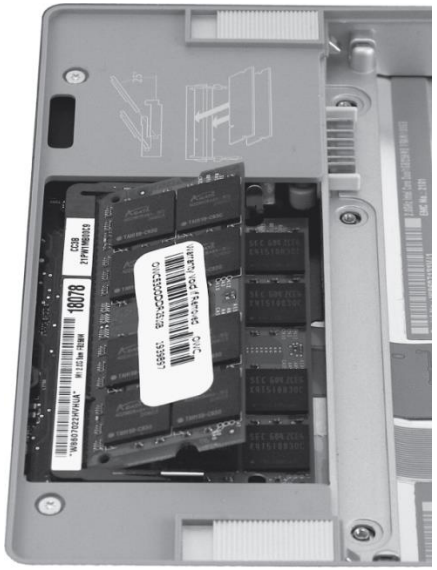


https://en.wikipedia.org/wiki/Transistor_count
https://en.wikipedia.org/wiki/Apple_M4

Computer System Organization







Reflect

Why take this course?

- Basic knowledge needed for *all* other areas of CS:
  operating systems, compilers, ...
- Levels are not independent
  hardware design ↔ software design ↔ performance
- Crossing boundaries is hard but important
  device drivers, system calls
- Good design techniques
  abstraction, layering, pipelining, parallel vs. serial, ...
- Understand where the world is going

The Mysteries of Computing will be revealed!

Get to know course website!

(see second announcement for best practices)

This class can be relentless.

Stay on top of it!