

# CSC 411

Computer Organization (Spring 2022)

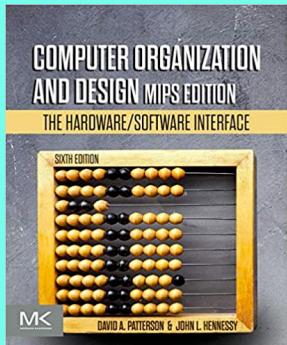
Lecture 3: Basics and Computer Abstractions

Prof. Marco Alvarez, University of Rhode Island

## Disclaimer

The following slides are adapted from:

Computer Organization and Design (Patterson and Hennessy)  
The Hardware/Software Interface



## Quick notes

### From previous lecture

- number systems (binary, decimal, hexadecimal, and others)
  - conversions from decimal to any
  - conversions from any to decimal
- bitwise operators (C/C++ programming)
  - &, |, ^, <<, >>

### Required reading

- chapter 1 (P&H)

## The computer revolution

### Progress in computer technology

- underpinned by domain-specific accelerators

### Novel applications:

- computers in automobiles
- cell phones
- human genome project
- World Wide Web
- search engines

### Computers are pervasive

## Classes of computers

### Personal computers (PC)

- general purpose, variety of software
- subject to cost/performance tradeoff

### Server computers

- network based
- high capacity, performance, reliability
- range from small servers to building sized

## Classes of computers

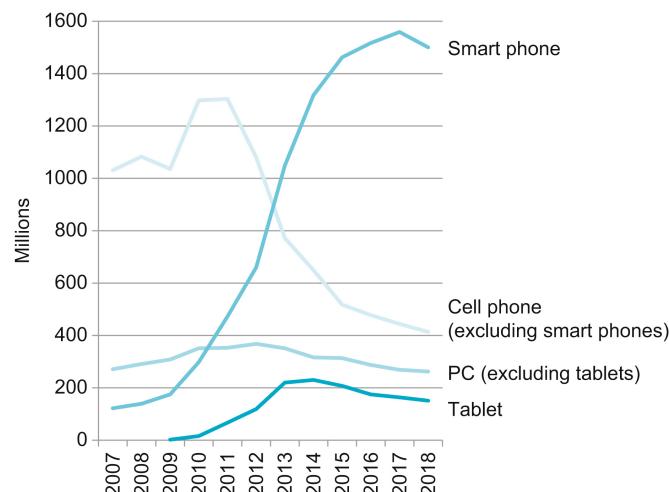
### Supercomputers

- type of server
- high-end scientific and engineering calculations
- highest capability but represent a small fraction of the overall computer market

### Embedded computers

- hidden as components of systems
- stringent power/performance/cost constraints

## The post PC era



## The post PC era

### Personal Mobile Device (PMD)

- battery operated
- connects to the Internet
- hundreds of dollars
- smart phones, tablets, electronic glasses

### Cloud computing

- Warehouse Scale Computers (WSC)
- Software as a Service (SaaS)
- portion of software run on a PMD and a portion run in the Cloud
- Amazon and Google

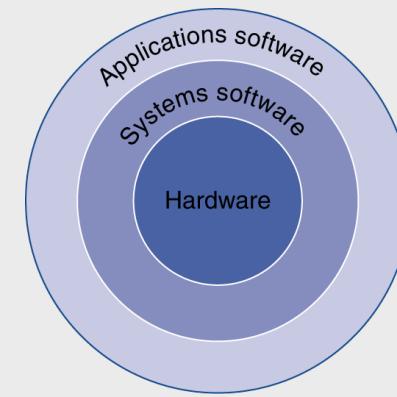
## Seven great ideas

- › Use **abstraction** to simplify design
- › Make the **common case fast**
- › Performance via **parallelism**
- › Performance via **pipelining**
- › Performance via **prediction**
- › **Hierarchy** of memories
- › **Dependability** via redundancy



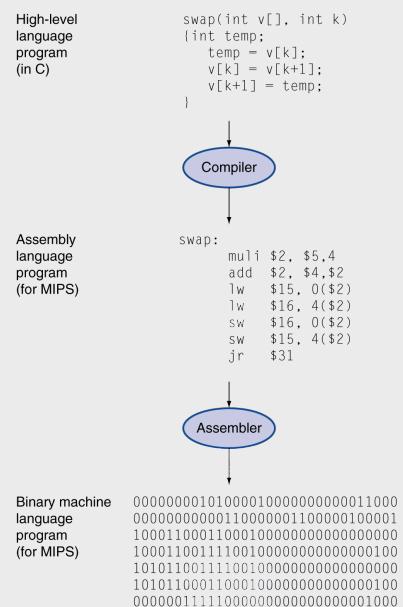
## Below your program

- › Application software
  - written in high-level language
- › System software
  - compiler: translates HLL code to machine code
  - operating system: service code
    - handling input/output
    - managing memory and storage
    - scheduling tasks & sharing resources
- › Hardware
  - processor, memory, I/O controllers



## Levels of program code

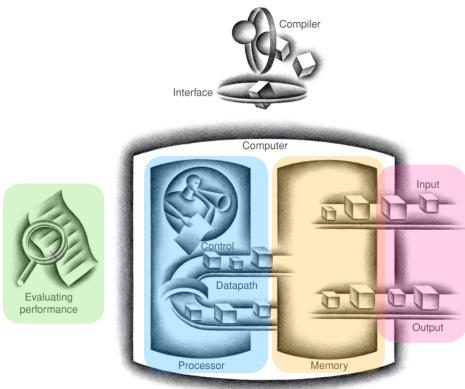
- › High-level language
  - level of abstraction closer to problem domain
  - provides for productivity and portability
- › Assembly language
  - textual representation of instructions
- › Hardware representation
  - binary digits (bits)
  - encoded instructions and data



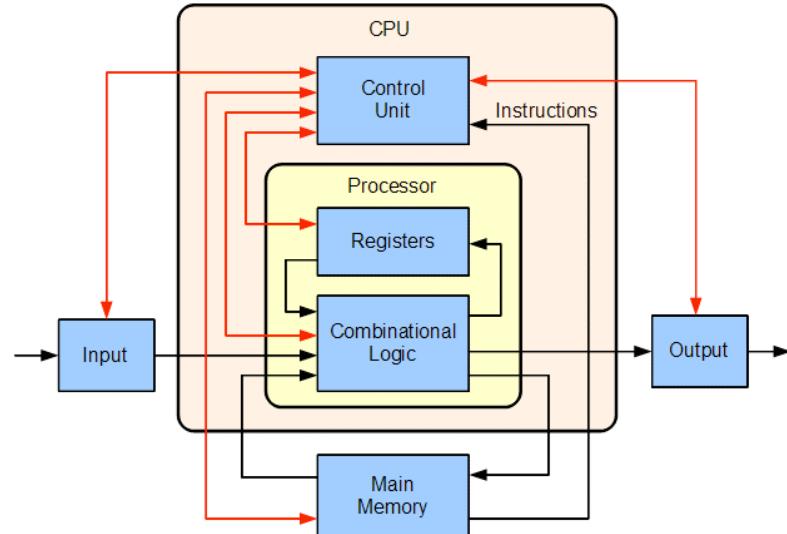
```
$ gcc hello.c -o prog
$ gcc -S hello.c -o prog
$ hexdump prog
$ xxd prog
```

# Components

- Same components for all kinds of computers
  - desktop, server, embedded

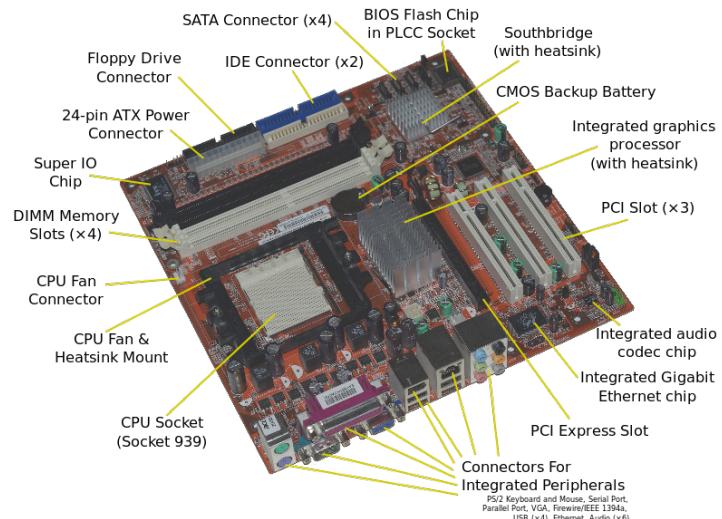


- Input/output includes
  - user-interface devices
    - display, keyboard, mouse
  - storage devices
    - hard disk, CD/DVD, flash
  - network adapters
    - for communicating with other computers



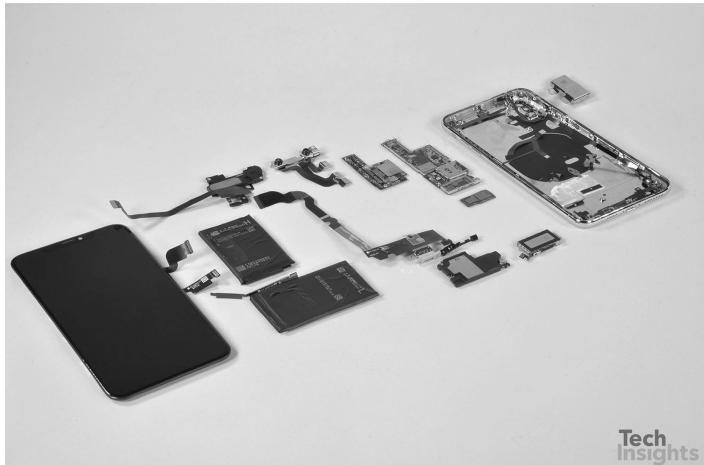
[https://en.wikipedia.org/wiki/Computer\\_architecture](https://en.wikipedia.org/wiki/Computer_architecture)

## Opening the box



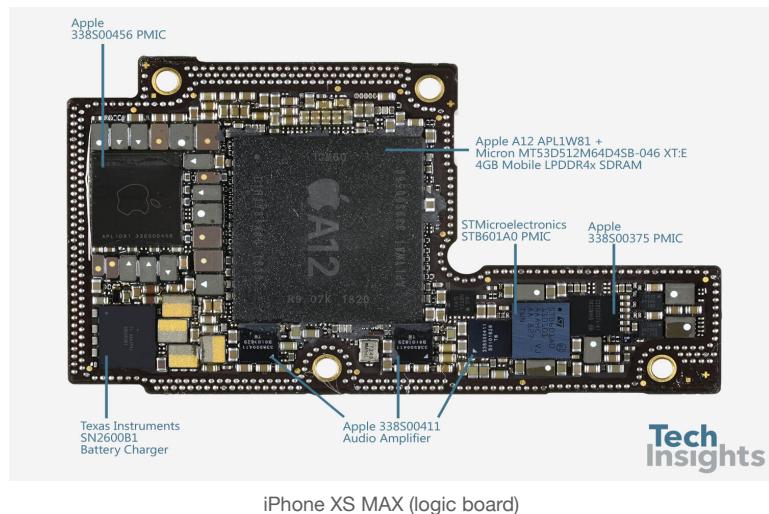
<https://electricalacademia.com/computer/motherboard-components/>

## Opening the box



iPhone XS MAX (components)

## Opening the box



## Inside the processor (CPU)

### ▪ Datapath

- performs operations on data

### ▪ Control

- sequences datapath, memory, ...

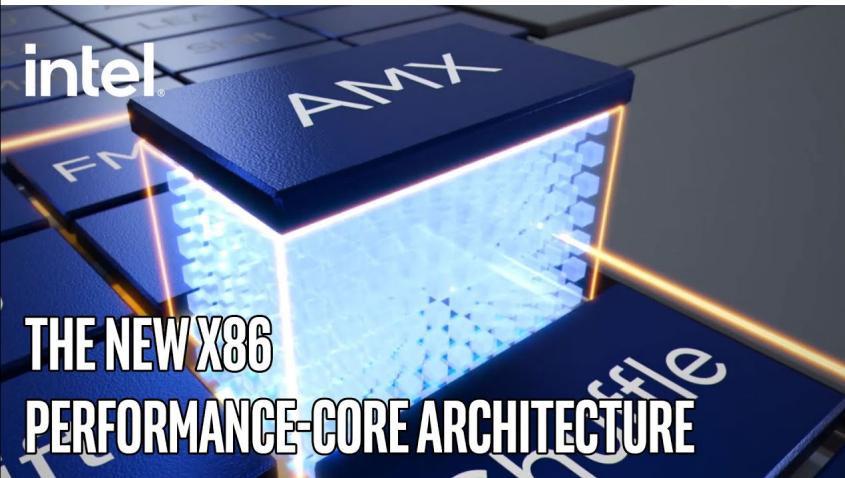
### ▪ Cache memory

- small fast SRAM memory for immediate access to data



A12 processor

## Intel's x86 core



<https://www.youtube.com/watch?v=ijTRvIQV7bE>

## Touchscreen

### ▪ PostPC device

### ▪ Supersedes keyboard and mouse

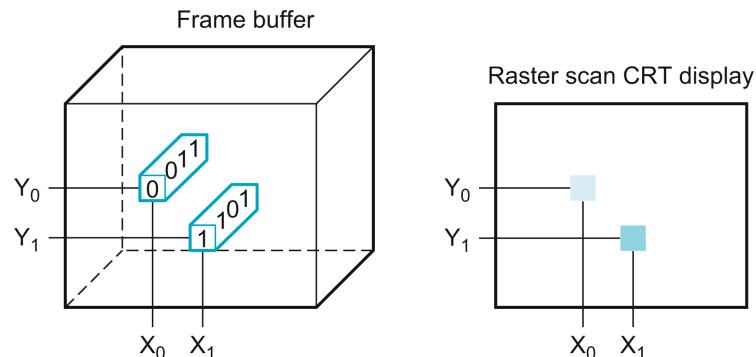
### ▪ Resistive and Capacitive types

- most tablets, smart phones use capacitive
- capacitive allows multiple touches simultaneously



## Through the looking glass

- LCD screen: picture elements (pixels)
  - mirrors content of frame buffer memory



## Safe place for data

- Volatile main memory
  - loses instructions and data when power off
- Non-volatile secondary memory
  - magnetic disk
  - flash memory
  - optical disk (CDROM, DVD)



## Networks

- Communication, resource sharing, nonlocal access
- Local area network (LAN): Ethernet
- Wide area network (WAN): the Internet
- Wireless network: WiFi, Bluetooth



## Technology trends

- Electronics technology continues to evolve
  - increased capacity and performance
  - reduced cost

Year	Technology used in computers	Relative performance/unit cost
1951	Vacuum tube	1
1965	Transistor	35
1975	Integrated circuit	900
1995	Very large-scale integrated circuit	2,400,000
2020	Ultra large-scale integrated circuit	500,000,000,000

# Chip Manufacturing

## Semiconductor technology

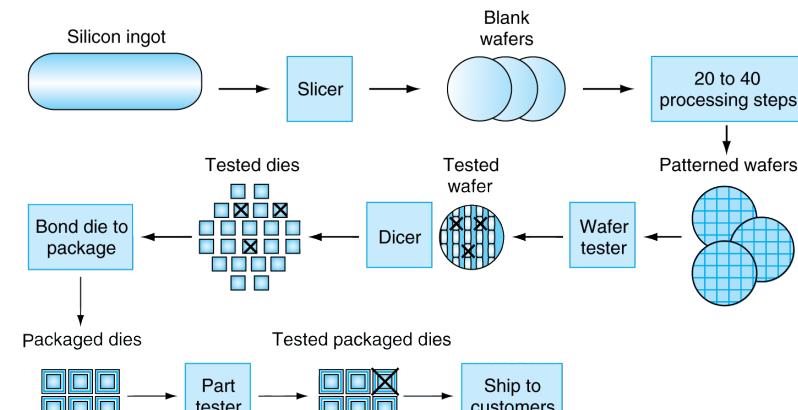
- Silicon



- Add materials to transform properties

- conductors of electricity
- insulators
- semiconductors (can conduct or insulate under specific conditions)

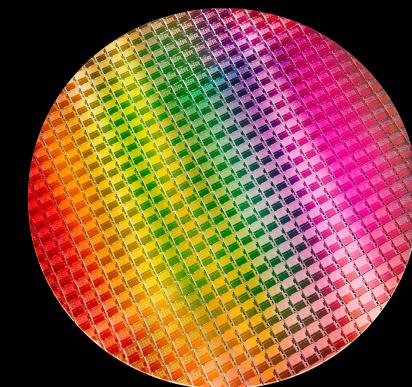
## Manufacturing ICs



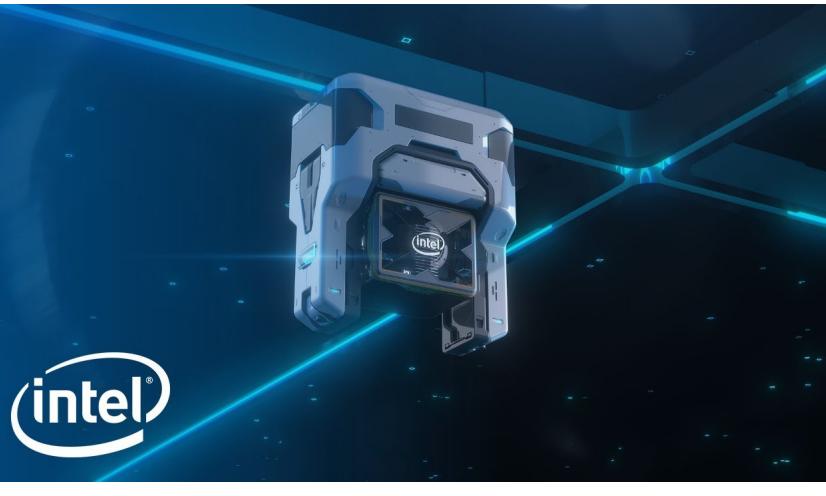
Yield: proportion of working dies per wafer

## Intel® Core 10th Gen

- 300mm wafer, 506 chips, 10nm technology
- Each chip is 11.4 x 10.7 mm



## From sand to silicon



[https://www.youtube.com/watch?v= VMYPLXnd7E](https://www.youtube.com/watch?v=VMYPLXnd7E)

## Integrated circuit cost

- Nonlinear relation to area and defect rate
  - wafer cost and area are fixed
  - defect rate determined by manufacturing process
  - die area determined by architecture and circuit design

$$\text{cost per die} = \frac{\text{cost per wafer}}{\text{dies per wafer} \times \text{yield}}$$

$$\text{dies per wafer} \approx \frac{\text{wafer area}}{\text{die area}}$$

$$\text{yield} = \frac{1}{(1 + (\text{defects per area} \times \text{die area}/2))^n}$$