ICS 122 Computer Organization

Jobish John

Indian Institute of Information Technology Kottayam

January 6, 2020

Some interesting questions?

- How does a computer work?
- What is present inside a computer?
- How various components interact to accomplish tasks?

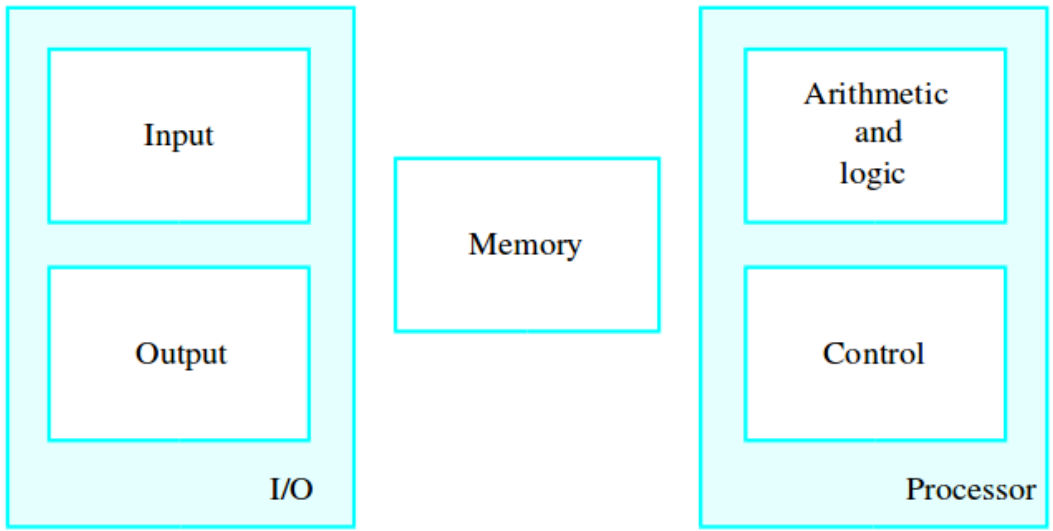
Some interesting questions?

- How does a computer work?
- What is present inside a computer?
- How various components interact to accomplish tasks?

https://www.anandtech.com

Some interesting questions?

- How does a computer work?
- What is present inside a computer?
- How various components interact to accomplish tasks?



Basic functional units of a computer

https://www.anandtech.com
Hamacher, C., Vranesic, Z., & Zaky, S. (2002). Computer organization. McGraw-Hill.

Expected outcome of the course

- Measure the performance of a computer system
- Arithmetic in computers
- Understand how the functional units of a computer works
- Understand how the hardware interacts and co-ordinate to execute a program
- Organization of memory, CPU, and I/O
- Pipelining

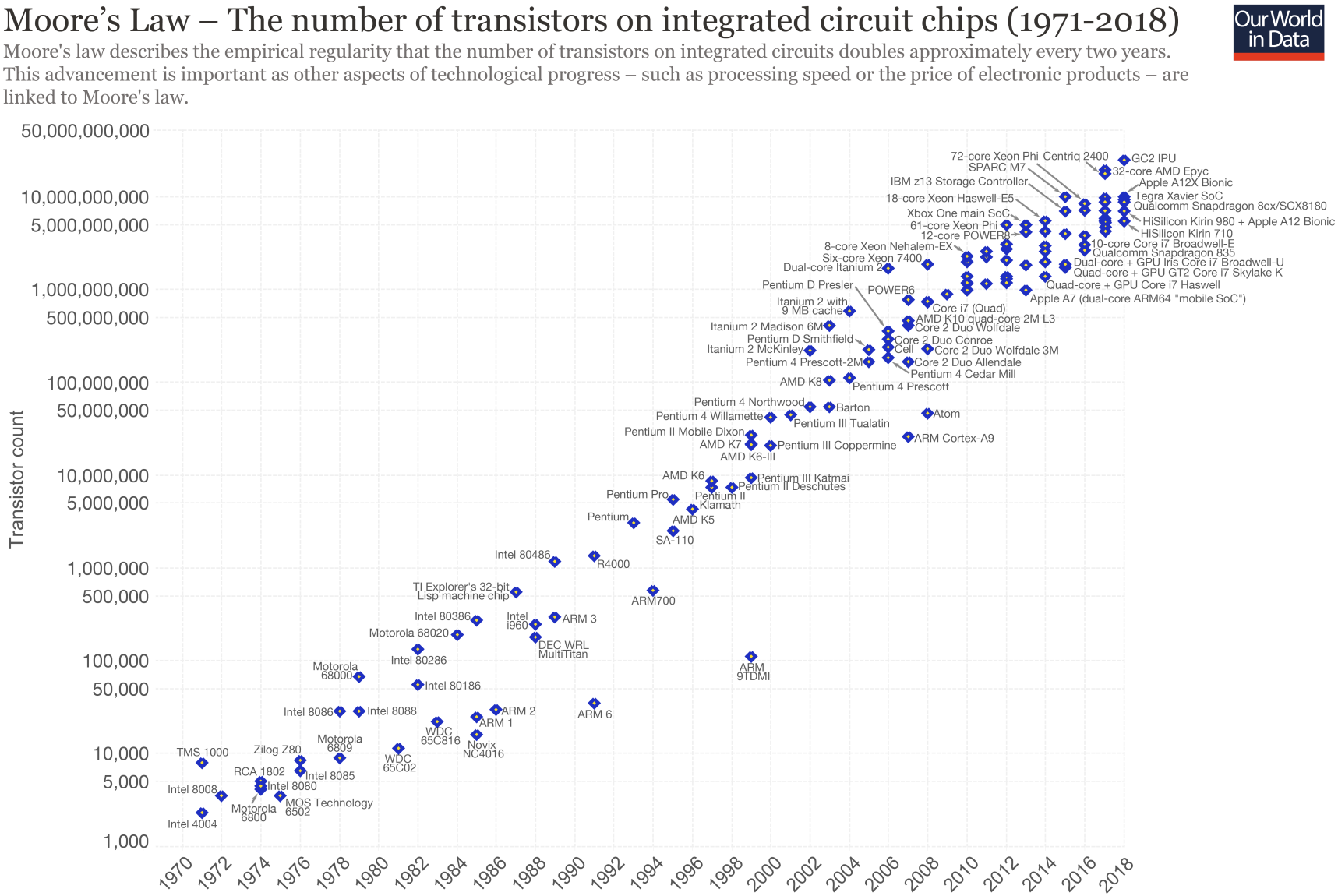
Moore's Law

Integrated circuit resources double every 18-24 months

- Gordon Moore - one of the founders of Intel
- Empirical relationship - Supported by experiments and observations
- Semiconductor device fabrication technology
- 180 nm (1999), 45 nm (2008), 22 nm (2012), 14 nm (2014)

https://www.sciencehistory.org

Moore's Law



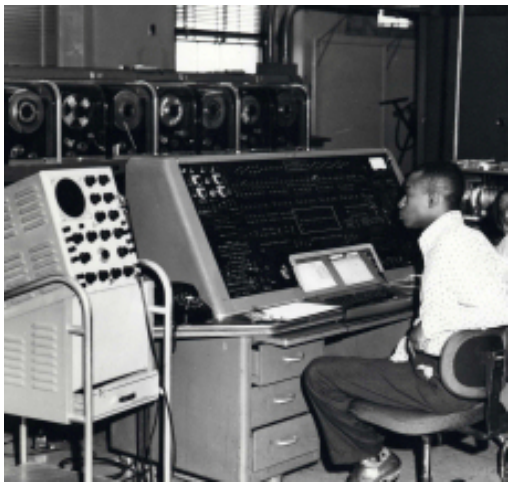
Data source: Wikipedia (https://en.wikipedia.org/wiki/Transistor_count)
The data visualization is available at OurWorldinData.org. There you find more visualizations and research on this topic.

Some historic highlights

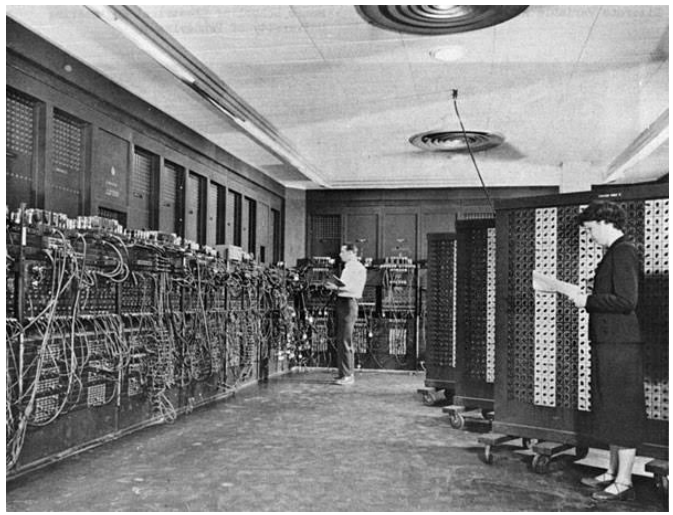
The principle of the modern computer was proposed by Alan Turing (1936)

First Generation: Vacuum Tubes (1940-1956)

- e.g. UNIVAC (Universal Automatic Computer) and ENIAC (Electronic Numerical Integrator And Computer)
- Solve only one problem at a time.
- Very expensive to operate
- Uses a great deal of electricity
- Generated a lot of heat



A UNIVAC computer



A ENIAC computer

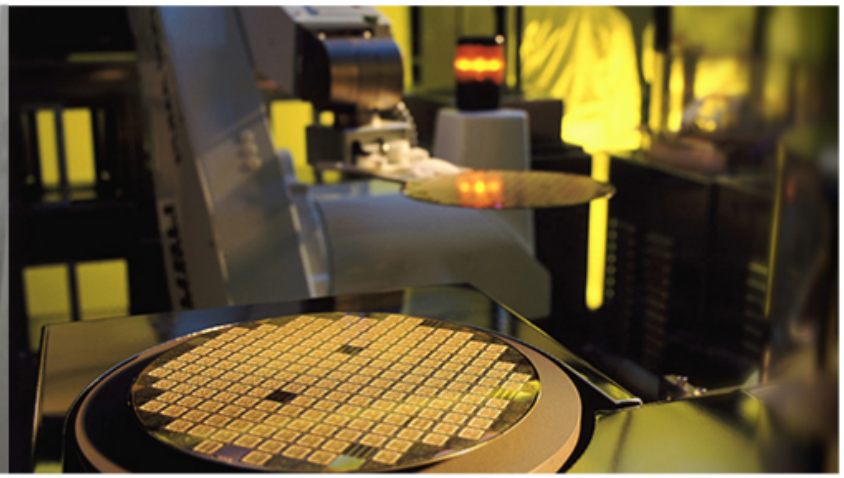
http://www.lgcnsblog.com, Census Bureau

Second Generation: Transistors (1956-1963)

- Invented at Bell Labs in 1947. Widely used in computers in the late 1950s
- Smaller, and require less power than vacuum tubes.

Third Generation: Integrated Circuits (1964-1971)

- Invented by **Jack Kilby** at Texas Instruments and **Robert Noyce** at Fairchild Semiconductor.
- Nobel prize in Physics - Jack Kilby (2000)
- Photolithography - IC production technique



https://www.computerhistory.org

Fourth Generation: Microprocessors (1971-Present)

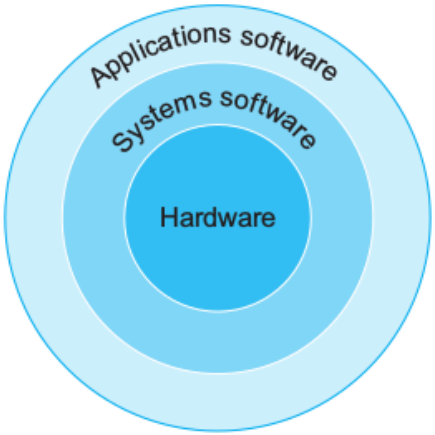
- 4004 developed by Italian physicist Federico Faggin in 1968
- Intel released the world's first commercial microprocessor - Intel 4004 (1971)
- Intel 8085 (1977), 8086 (1978), 8088 (1979), 80386 (1985), Intel Pentium (1993), Intel Pentium Pro (1995), Intel Pentium 3 (1999), Intel Pentium 4 (2000), Intel Core 2 duo, dual core (2006), Core i series (2008), Intel Atom, Xeon, Haswell
- In 1981 IBM introduced its first computer for the home user.
- GUI, mouse, other handheld devices and peripherals

Interacting with the computer

- High Level Language
- Assembly Language
- Machine language
  (binary digit/bit - 0/1)

System Software

- Compiler
- Assembler
- Operating System
- Linker
- Loader



High-level language program (in C)

```
swap(int v[], int k)
{int temp;
   temp = v[k];
   v[k] = v[k+1];
   v[k+1] = temp;
}
```

Compiler

Assembly language program (for MIPS)

```
swap:
      multi $2, $5,4
      add   $2, $4,$2
      lw    $15, 0($2)
      lw    $16, 4($2)
      sw    $16, 0($2)
      sw    $15, 4($2)
      jr    $31
```

Assembler

Binary machine language program (for MIPS)

```
00000000101000100000000100011000
00000000100000100001000000100001
10001101111000100000000000000000
10001110000100100000000000000100
10101110000100100000000000000000
10101101111000100000000000000100
```

D. A. Pattersen and J. L. Hennesy, Computer organization and design MIPS edition: the hardware/software interface.

Linker Vs Loader

Linker

- Generates the executable module of a source program
- Takes as input, the object code generated by an assembler.
- Function: Combines all the object modules of a source code to generate an executable module

Loader

- Loads the executable module to the main memory
- Input: Takes executable module generated by a linker
- Function: It allocates the addresses to an executable module in main memory for execution.

Course Outline

What we are going to discuss in this course

- Measuring Performance
- Advanced Risc Machine (ARM) architecture
- Arithmetic in computers(fast adders and multipliers)
- System Organization - complete execution of an instruction.
- Bus Organization and control signal generation
- Input / output organization and Interrupts
- Memory Organization - cache memory, virtual memory
- Pipelining - performance analysis and hazards

Evaluation

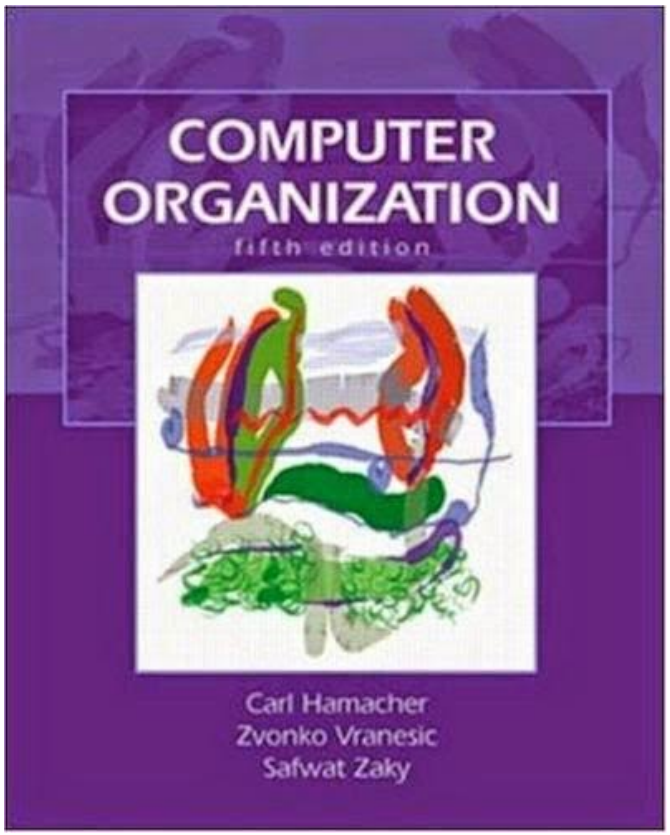
- Mid semester exams - 20 + 20
- End semester exam - 50
- Assignments/quizzes - 10

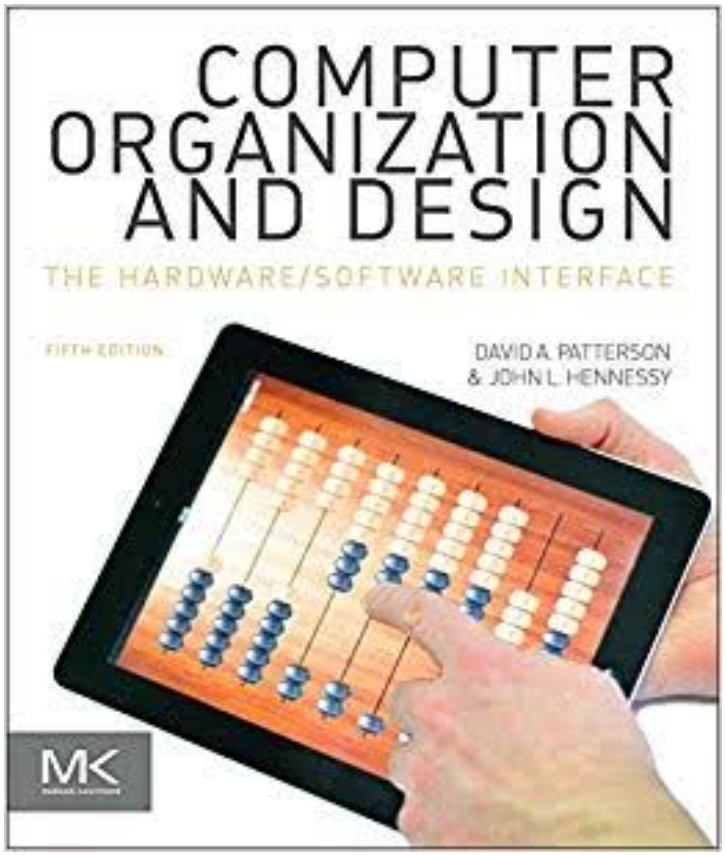
References/Books

1. Carl Hamacher, Zvonko Vranesic and Safwat Zaky, **Computer Organization**, Fifth Edition, McGraw Hill, 2002

2. D. A. Pattersen and J. L. Hennesy, **Computer organization and design MIPS edition: the hardware/software interface**. Fifth Edition (2013)

References/Books





Thank You