

CS 31007

Autumn 2019

COMPUTER ORGANIZATION AND ARCHITECTURE

Instructors

Section I - Rajat Subhra Chakraborty
Class Room NR 423

Section II - Bhargab B. Bhattacharya
Class Room NR 124

Indian Institute of Technology Kharagpur
Computer Science and Engineering

Today's agenda

- ❖ Overview of the course
- ❖ Evolution and history of computer design
- ❖ Moore's law
- ❖ Basic components of a computer
- ❖ Instruction Set Architecture (ISA)
- ❖ Computer organization and computer architecture: Bottom-up and Top-down view

Acknowledgement: Patterson and Hennessy; V. D. Agrawal; S. R. Sarangi

CS 31007

Autumn 2019

COMPUTER ORGANIZATION AND ARCHITECTURE

Class Schedule

12:00-12:55 (Mon), 10:00-11:55 (Tue), and 8:00-8:55 (Thurs)

Tutorial class will be held on **every Thursday**

Lab Course (39001): Computer Organization Laboratory

Prerequisites

Basic logic design, combinational and sequential circuits,
knowledge of high-level programming language

CS 31007

Autumn 2019

COMPUTER ORGANIZATION AND ARCHITECTURE

Prerequisites

Basic logic design, combinational and sequential circuits,
knowledge of high-level programming language

Announcement

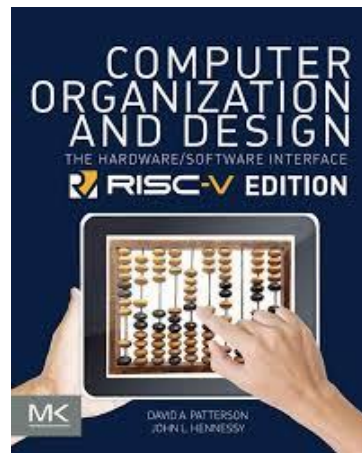
Pre-Requisite In-Class Quiz: **Monday, 22 July 2019**

12:00-12:55 (Open-Book, Open-Notes)

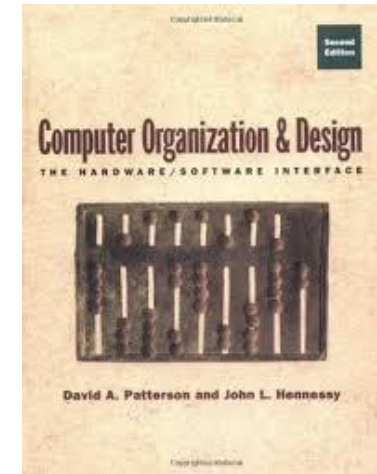
Textbook



D. A. Patterson and J. L. Hennessy
Computer Organization and Design
- The Hardware Software Interface
5th Edition, Morgan Kaufmann, 2014



RISC-V Edition 2018



Second Edition 1998



2017 ACM Turing
Award

Further Reading

1. Smruti R. Sarangi, *Computer Organisation and Architecture*, McGraw Hill India, 2014
2. William Stallings, *Computer Organization and Architecture: Designing for Performance*, Eight Edition, Prentice Hall, 2010.
3. John P. Hayes, *Computer Architecture and Organization*, 3rd Edition, Tata McGraw Hill, 2012.

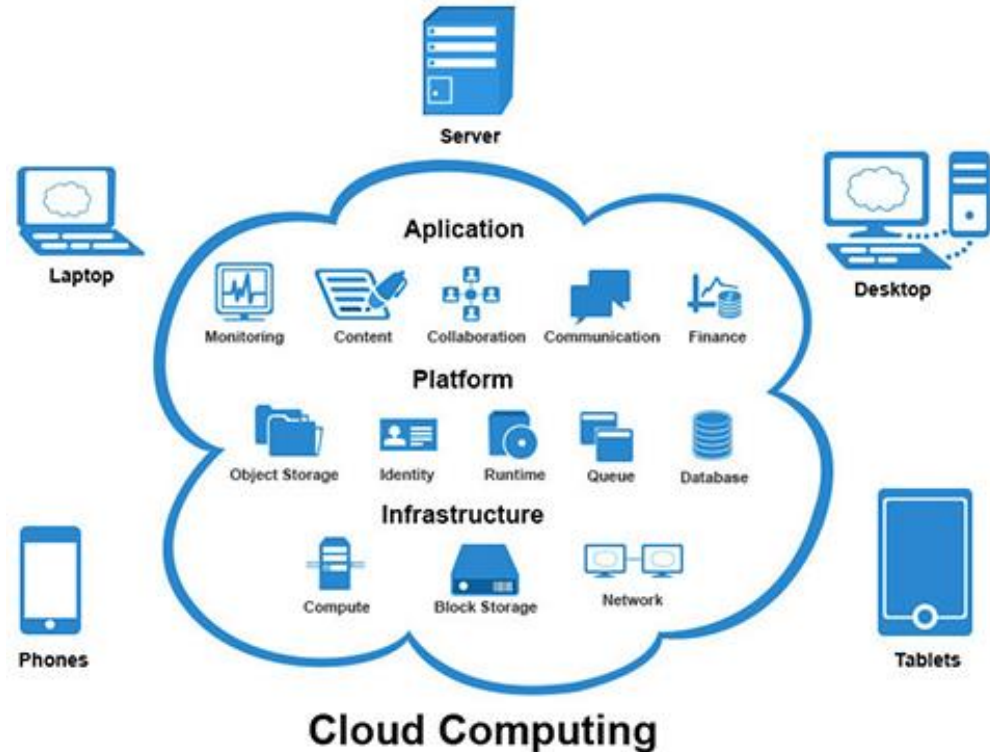
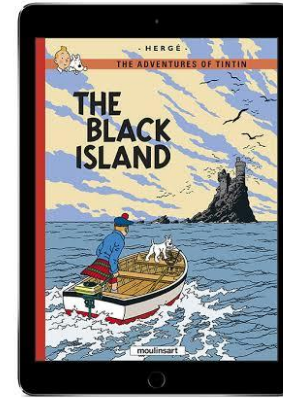
Grading Policy

- Homework and Quiz for practicing
- In-Class Quiz on Tutorial Days: 10% ($=5 \times 2$)
- Class-Test 1 and Class-Test 2: 10% ($=2 \times 5$)
- Mid-Sem Exam (2 hour): 30%; 16-24 Sept. 2019
- End-Sem Exam (3 hour): 50%; 18-27 November 2019

Goal is to understand

- What are the principles of computer design?
- What is the hardware-software interface?
- What is computer organization and computer architecture?
- How to design the instruction set of a machine?
- How instructions are executed in a machine?
- What are the techniques for performing fast computation?
- How a complete processor is designed?

What is a Computer ?



Car or Computers?

BMW
345i



- 2,000,000 lines of code
- Fiftythree 8-bit microprocessors
- Eleven 32-bit microprocessors
- Seven 16-bit microprocessors

The Computer Revolution

- Makes novel applications feasible
 - Computers in automobiles
 - Cell phones
 - Human genome project
 - World Wide Web
 - Search Engines
- Computers are pervasive

Classes of Computers

- Personal computers
 - 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

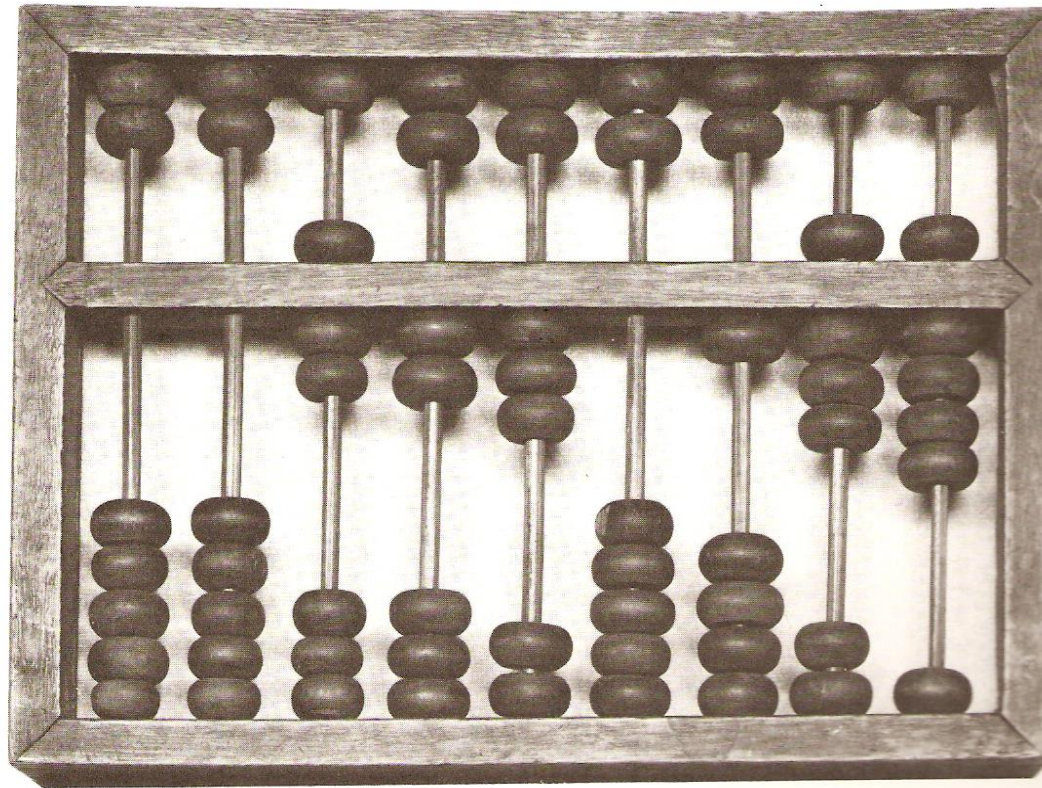
Classes of Computers

- Supercomputers
 - 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

A computer is a general purpose machine which can process Information and yield results

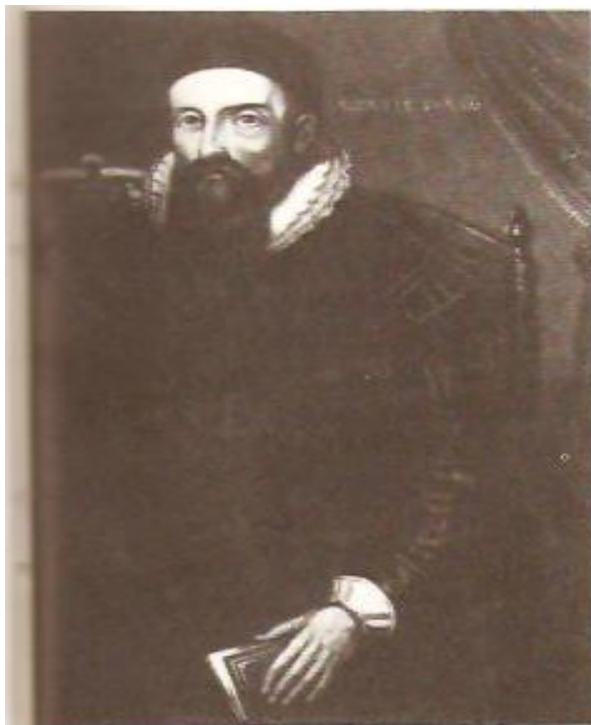
However, centuries ago, by a computer was meant ...

The First Mechanical Computing Device



The nineteenth century Chinese abacus, numbers are entered by sliding the beads towards the crossbar.

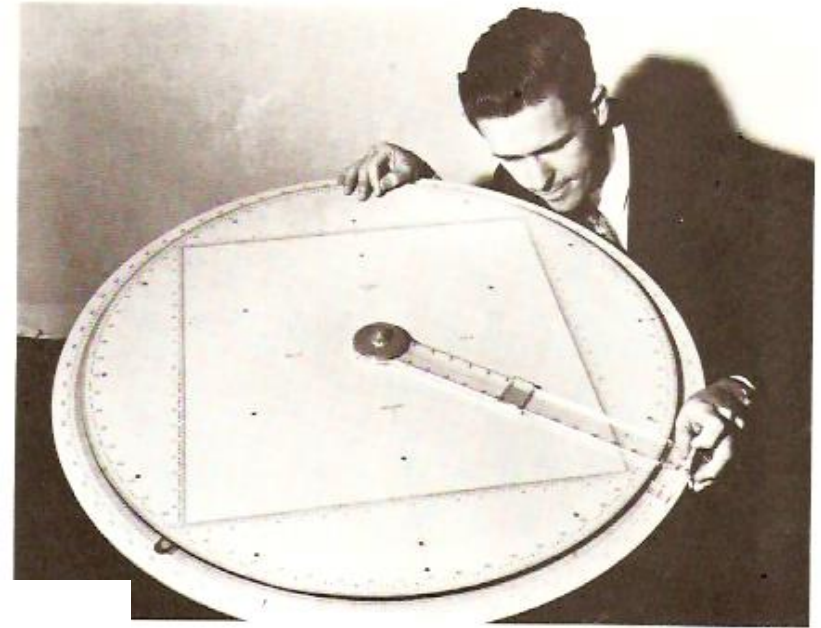
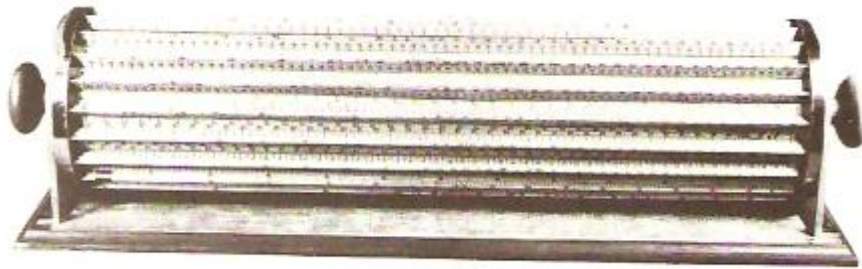
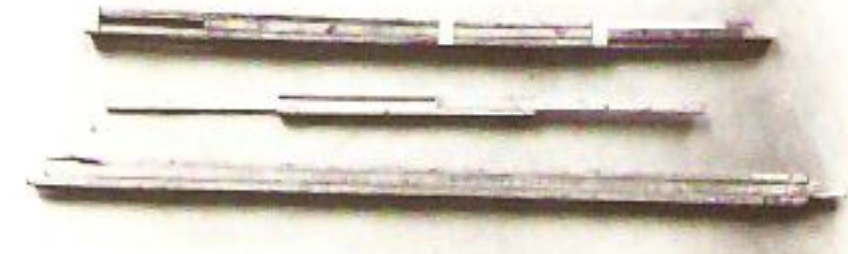
First Steps in Computing



John Napier (1550-1617)

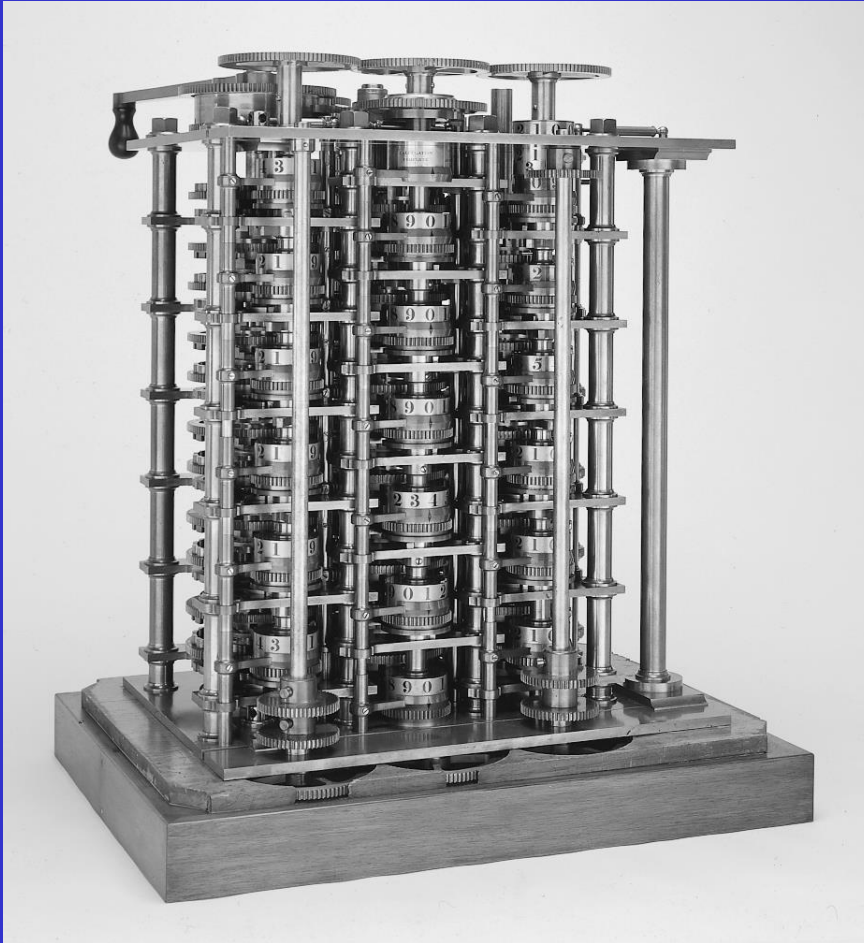
The front page of Napier's famous paper (1614) that introduced logarithm and contained ninety pages of tables.

Slide Rule



The first slide rule (top left) was made in 1654; the cylindrical slide rule (below left) and circular slide rules are useful modifications that were used for various computations; Slide rule (left-bottom) used by early generation of engineers.

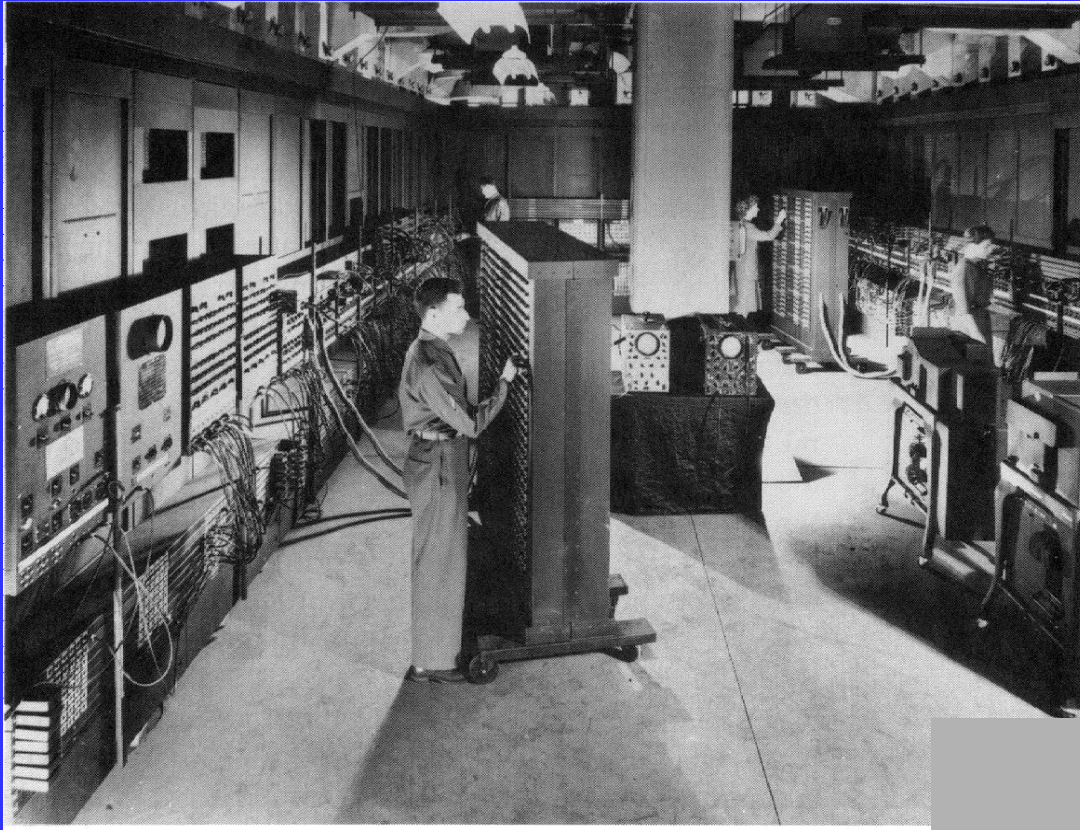
The First Computer (1832)



Babbage Difference
Engine

25000 parts

ENIAC - The First Electronic Computer (1943-1946)



Length = 80 ft

Height = 8.5 ft

Floor area = 1500 sq ft

Weight = 30 tons

18000 vacuum tubes

70,000 resistors

140 kw of power

John Mauchly (professor) and J. Presper Eckert (graduate student) built ENIAC at the University of Pennsylvania, Philadelphia

The Snapshots of ENIAC (1946)



HOW MUCH IS $\sqrt[3]{2589^{16}}$?

The Army's ENIAC can give you the answer in a fraction of a second!

Think that's a stumper? You should see *some* of the ENIAC's problems! Brain twisters that if put to paper would run off this page and feet beyond . . . addition, subtraction, multiplication, division—square root, cube root, any root. Solved by an incredibly complex system of circuits operating 18,000 electronic tubes and tipping the scales at 30 tons!

The ENIAC is symbolic of many amazing Army devices with a brilliant future for you! The new Regular Army needs men with aptitude for scientific work, and as one of the first trained in the post-war era, you stand to get in on the ground floor of important jobs

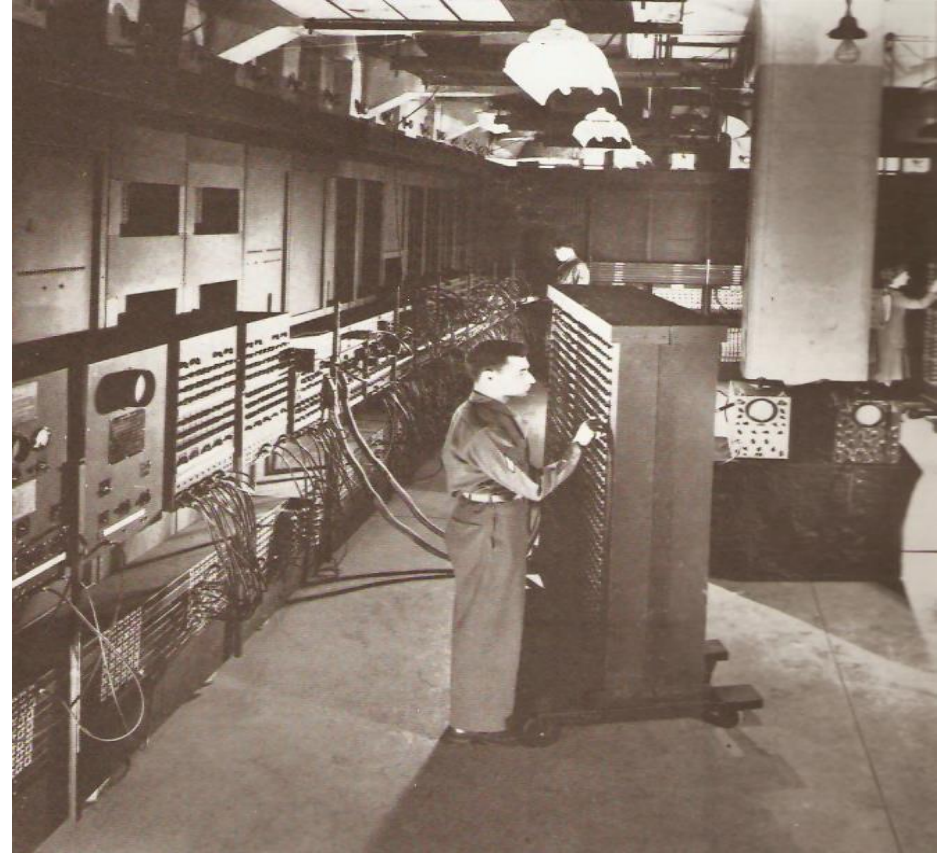
**A GOOD JOB FOR YOU
U. S. Army
CHOOSE THIS
FINE PROFESSION NOW!**

**YOUR REGULAR ARMY SERVES THE NATION
AND MANKIND IN WAR AND PEACE**

which have never before existed. You'll find that an Army career pays off.

The most attractive fields are filling quickly. Get into the swim while the getting's good! 1½, 2 and 3 year enlistments are open in the Regular Army to ambitious young men 18 to 34 (17 with parents' consent) who are otherwise qualified. If you enlist for 3 years, you may choose your own branch of the service, of those still open. Get full details at your nearest Army Recruiting Station.

This advertisement shows the amazing calculating powers of the ENIAC



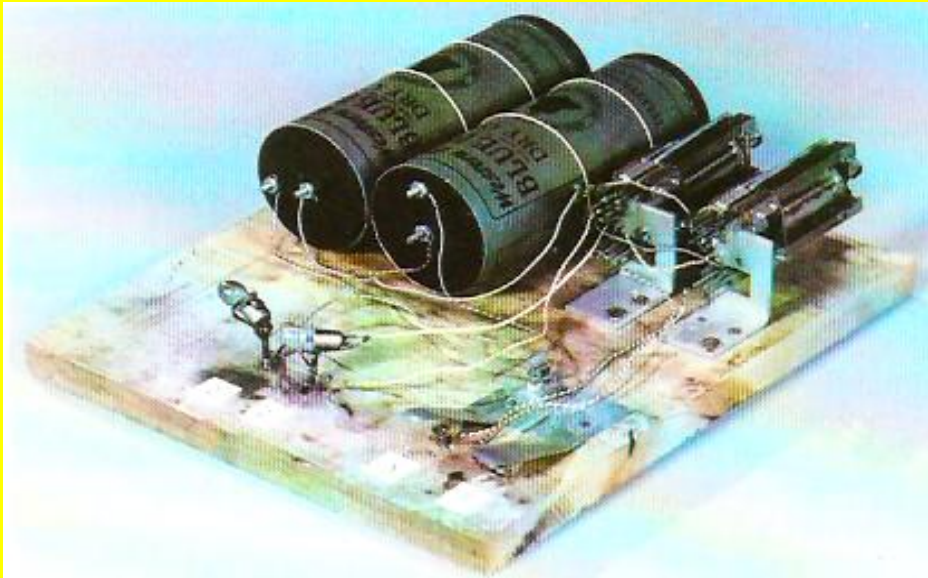
A technician tracks down a misplaced cable in the ENIAC.

Courtesy: S. Augarten, Bit by Bit

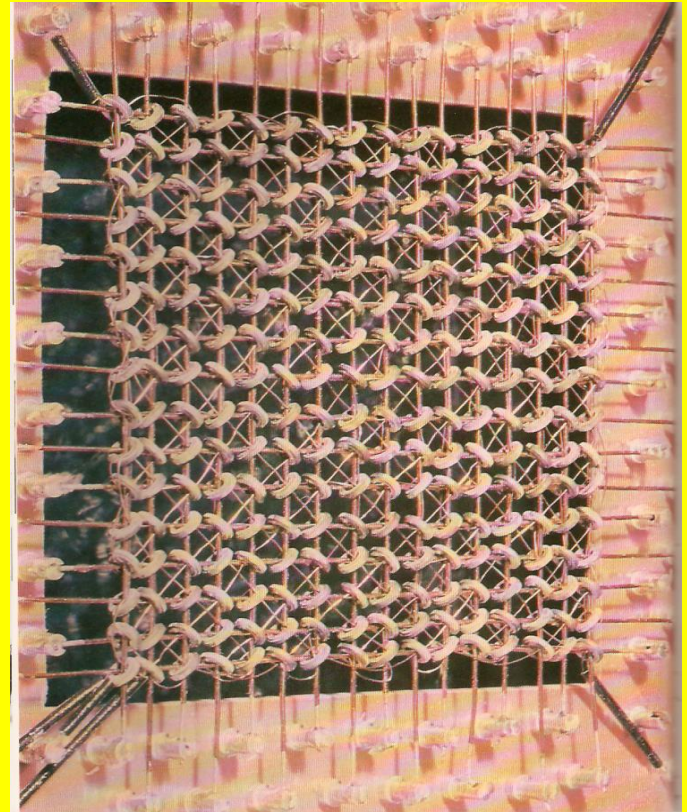
The Miracle of Electronic Evolution

Battery??

No, an early 1024-bit memory!

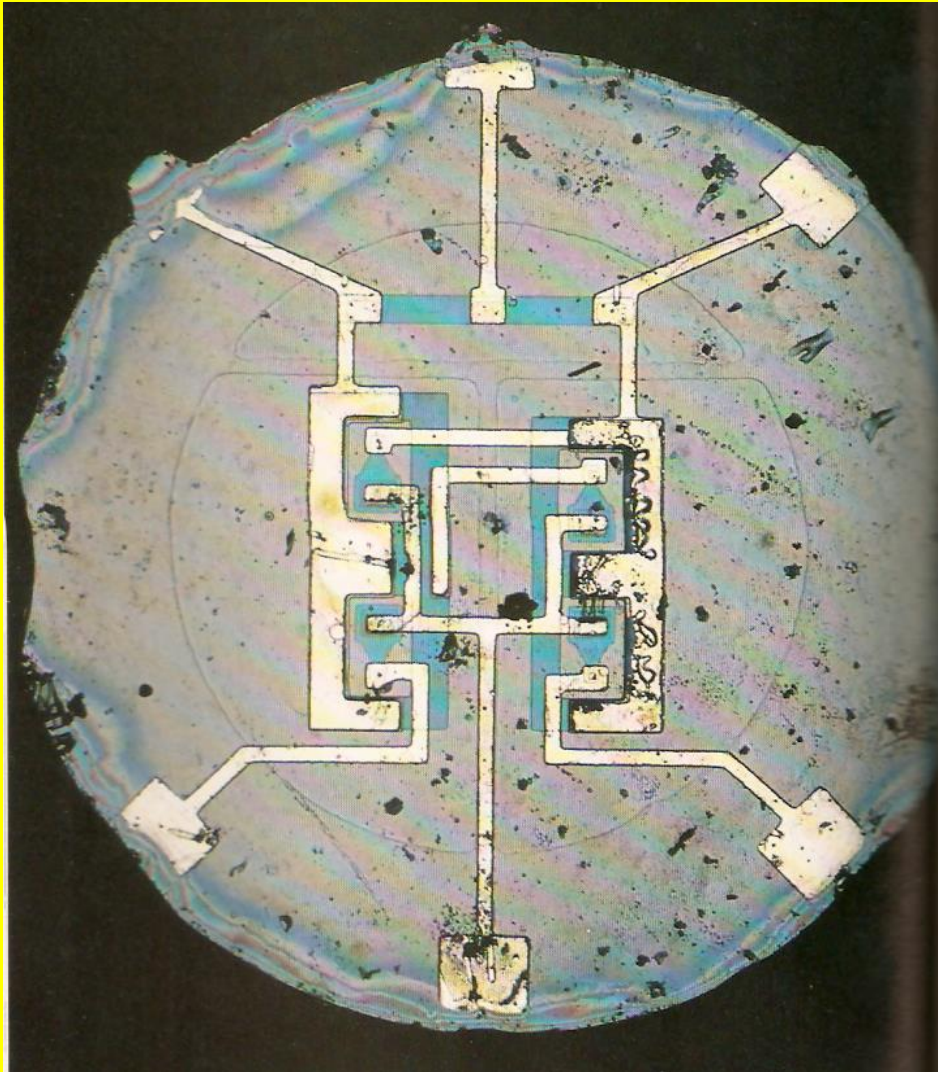


No, an early binary adder:
built on a kitchen table!



An electrical relay network??

The First Integrated Circuit



An early integrated circuit made by Fairchild Corporation in 1961. This is a logic IC with two flip flops, the 4 blue structures in the center are transistors, and the white lines are aluminum connectors.

Hardware Design Creating History



Andy Grove, Robert Noyce, and Gordon Moore (1978)
Intel Founders

Entering the World of Microprocessors



**Announcing
a new era
of integrated
electronics**

**A micro-
programmable
computer
on a chip!**

Intel introduces an integrated CPU complete with a 4-bit parallel adder, sixteen 8-bit registers, an accumulator and a push-down stack of one chip. It's one of a family of four new ICs which comprise the MCS-4 microcomputer system. The final system is being put to the point and test by a dedicated general-purpose computer at Intel or as few as two dual in-line packages.

MCS-4 systems provide complete computing and control functions for test systems, data terminals, billing machines, measuring systems, numeric control systems and process control systems.

The heart of any MCS-4 system is a Type 4004 CPU, which includes a powerful set of 60 instructions. Adding one or more Type 4007 ROMs for program storage and data RAMs gives you a fully functioning micro-programmed computer. To this you may add Type 4002 RAMs for random access and Type 4003 registers to expand the output ports.

Using no circuits other than ICs from this family of four, you can create a system with 4096 bytes of ROM storage and 5120 bits of RAM storage. When you require rapid turn-around or need only a few systems, Intel's erasable and re-programmable ROM, Type 1701, may be substituted for the Type 4007 mask-programmed ROM.

MCS-4 systems interface easily with switches, keyboards, displays, teleprinters, printers, readers, A/D converters and other popular peripherals.

The MCS-4 family is new in which at Intel's Santa Clara headquarters and at our marketing headquarters in Europe and Japan. In the U.S., contact your local Intel representative for technical information and literature. In Europe, contact Intel at Avenue Louise 218, B-1050 Brussels, Belgium. Phone 482003. In Japan, contact Intel Japan, Inc., Parkside Plaza Bldg. 6th, 6-2-2 Sendagaya, Shibuya-Ku, Tokyo 151. Phone 03-403-4147.

Intel Corporation now produces micro computers, memory devices and memory systems at 3065 Bowers Avenue, Santa Clara, Calif. 95051. Phone (408) 246-7901.

**intel
delivers.**

Intel's first advertisement for the 4004 microprocessor that appeared in November 1971.

Evolution of Microprocessors

1970: 4004 μ P



#T = 2300
Auditorium

1982: 286 μ P



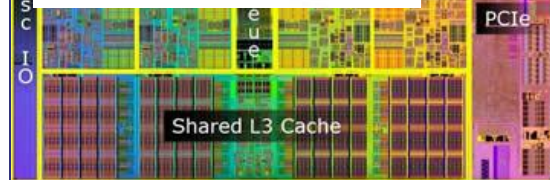
#T = 1,34,000
Stadium

1999: Pent III



#T = 32 million
Population of
Tokyo

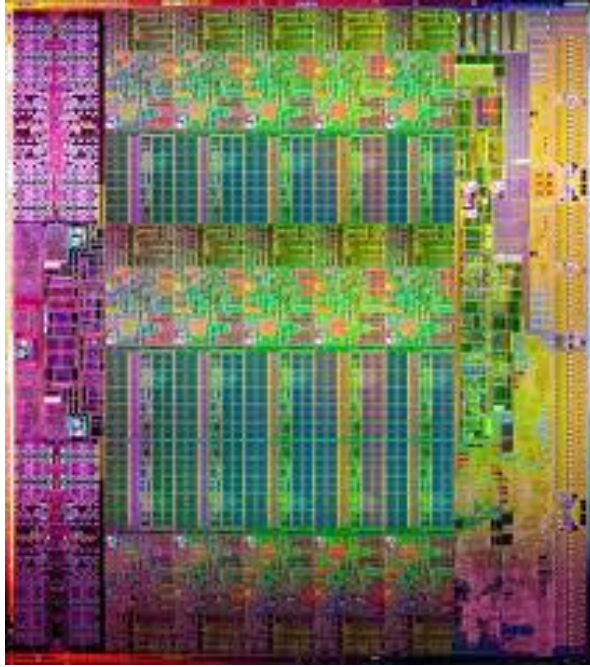
2008: Core i7



#T = 1.3 billion
Population of China

Number of transistors (#T)

Modern Processors



Intel 15-Core Xeon Ivy
Bridge-Ex (2014)

22 *nm* technology

4.3 billion transistors

Die size $\sim 600 \text{ mm}^2$

Intel Broadwell EP Xeon SoC
(2016) – 7.2 billion T; 14 *nm*

IBM Deep Blue defeated Grand
Master Garry Kasparov in May
1997



Gordon Moore: 1965

Progress in computer technology underpinned by Moore's Law

- Predicted that the number of transistors integrated on a die would grow exponentially (doubling every 12 to 18 months)
- Million transistors/chip barrier crossed in the 1980s
- Today:
Around several billion transistors per chip

Transistor Count in a Chip

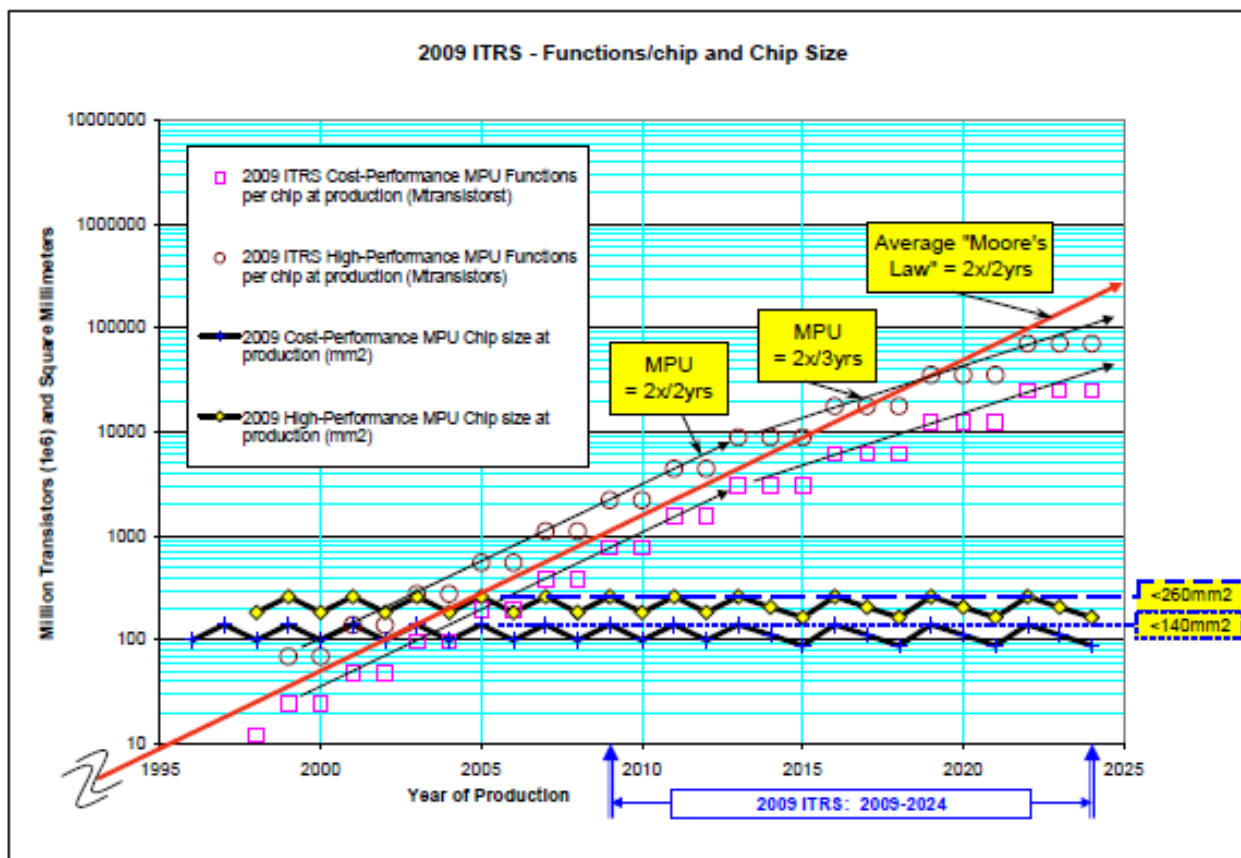
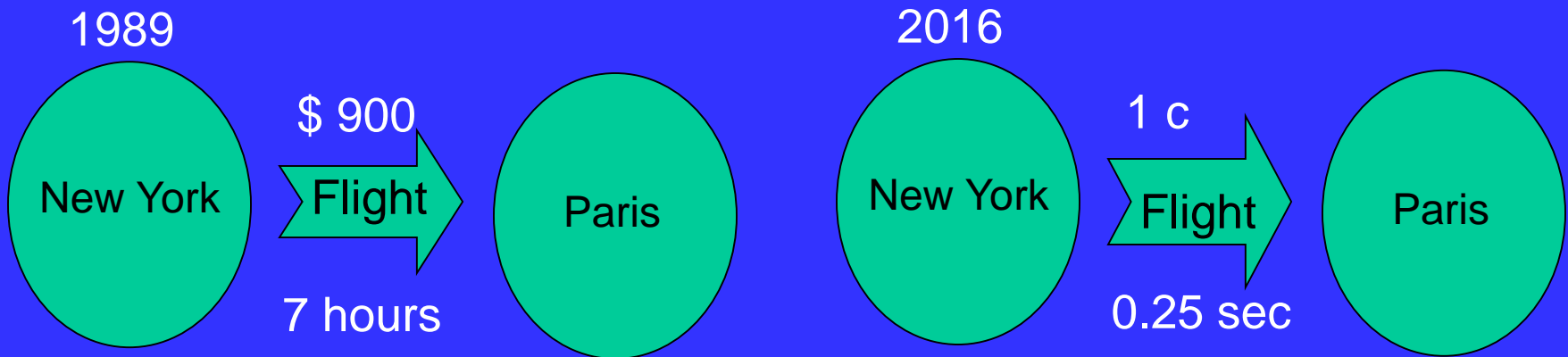
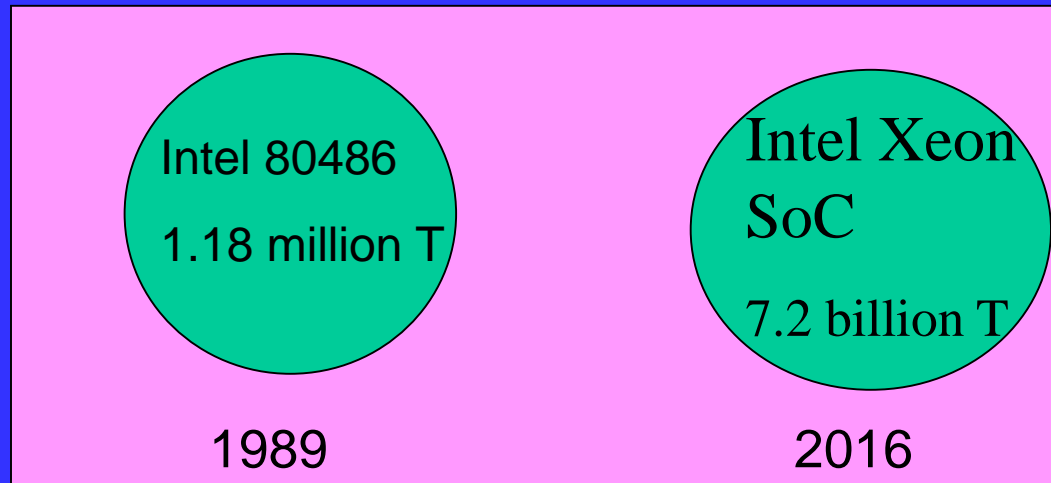


Figure 7b 2009 ITRS Product Technology Trends:
MPU Product Functions/Chip and Industry Average "Moore's Law" and Chip Size Trends

Courtesy: ITRS 2010

Example: Moore's Law



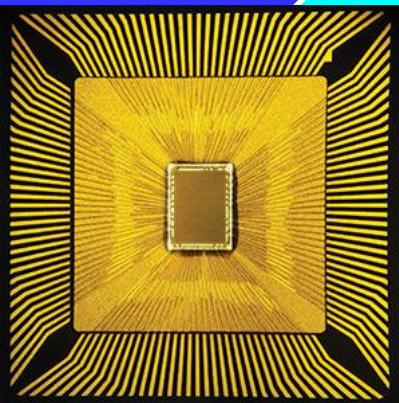
Computers of the Future ...

Neuromorphic Chips

bridging the gap between artificial and natural computation

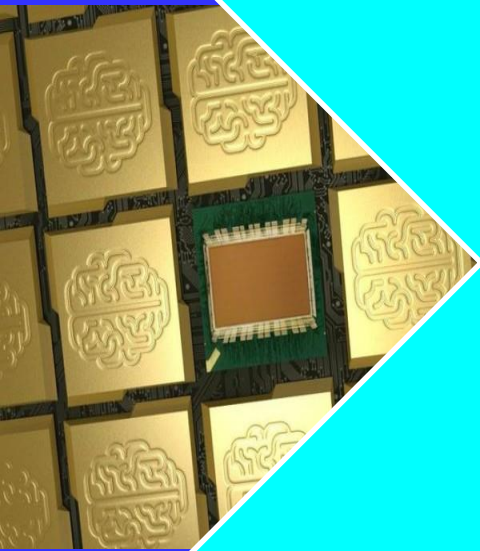


Advances in neuroscience and chip technology now enable to process data in a similar way like brain. These “neuromorphic” chips will play chess, drive cars reliably in all conditions, and empower smartphones to act as personal assistants.



Neuromorphic Chips

bridging the gap between artificial and natural computation



IBM TrueNorth
cores, which
human neuro
synapses, two
biological build
up the h

This IBM chip
patterns of puls
to one of t
neuroscientis
stores i

SyNAPSE (Systems of Neur
Scalable Electronics) project

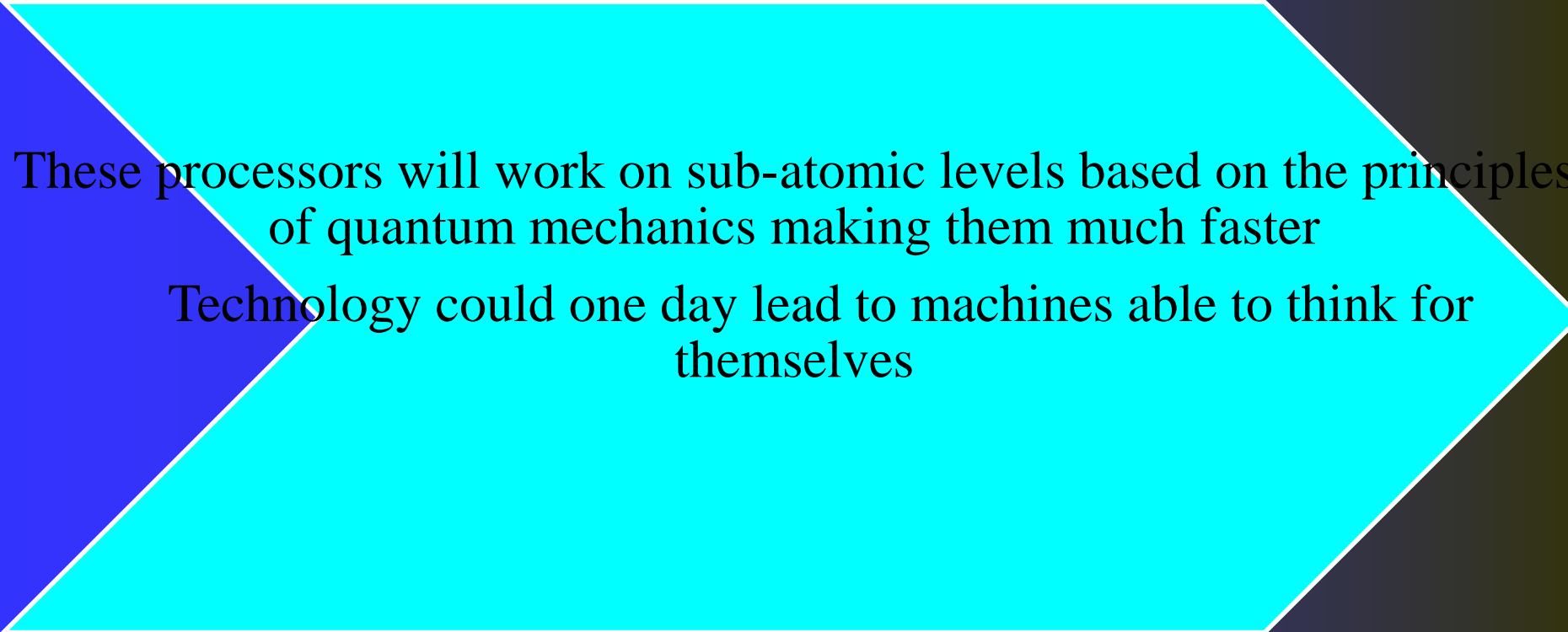
Recognizes people, cyclists, cars, buses, and trucks with about 80 percent accuracy.



Do not separate processor & memory

Burns only 63 mW of power to process streaming video with 30 frames per sec.

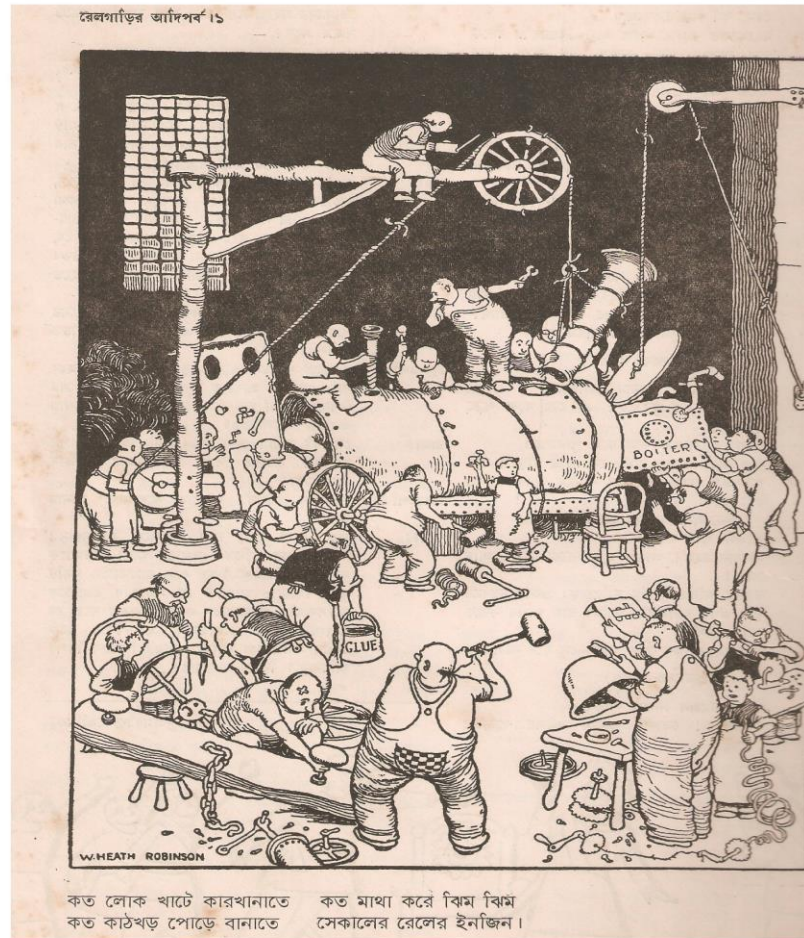
Google and IBM Announce Quantum Computers (2015 - 2019)



These processors will work on sub-atomic levels based on the principles of quantum mechanics making them much faster

Technology could one day lead to machines able to think for themselves

Building a Computer System



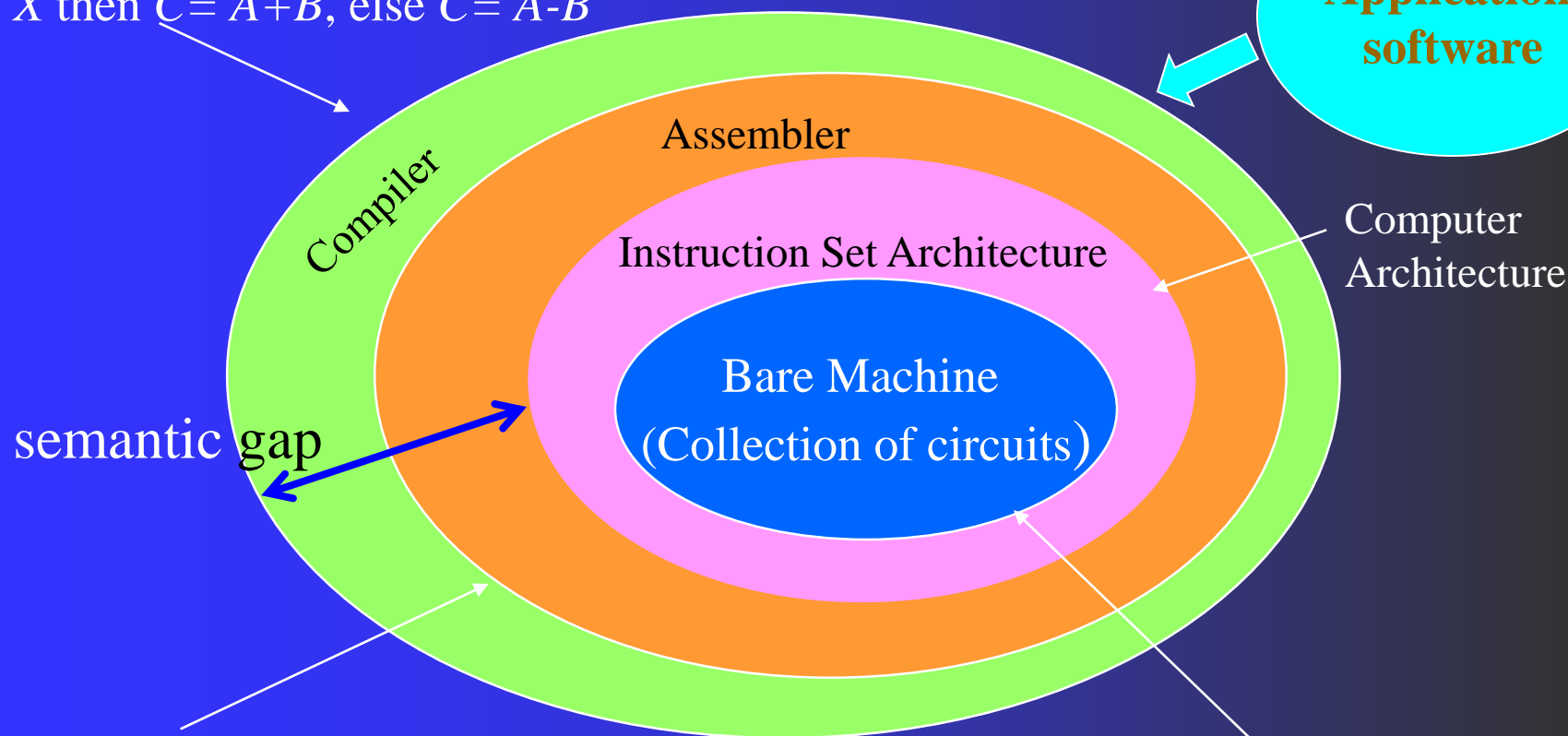
Courtesy: Satyajit Ray

Computing System Hierarchy

Can understand high-level language, e.g.,

If X then $C = A + B$, else $C = A - B$

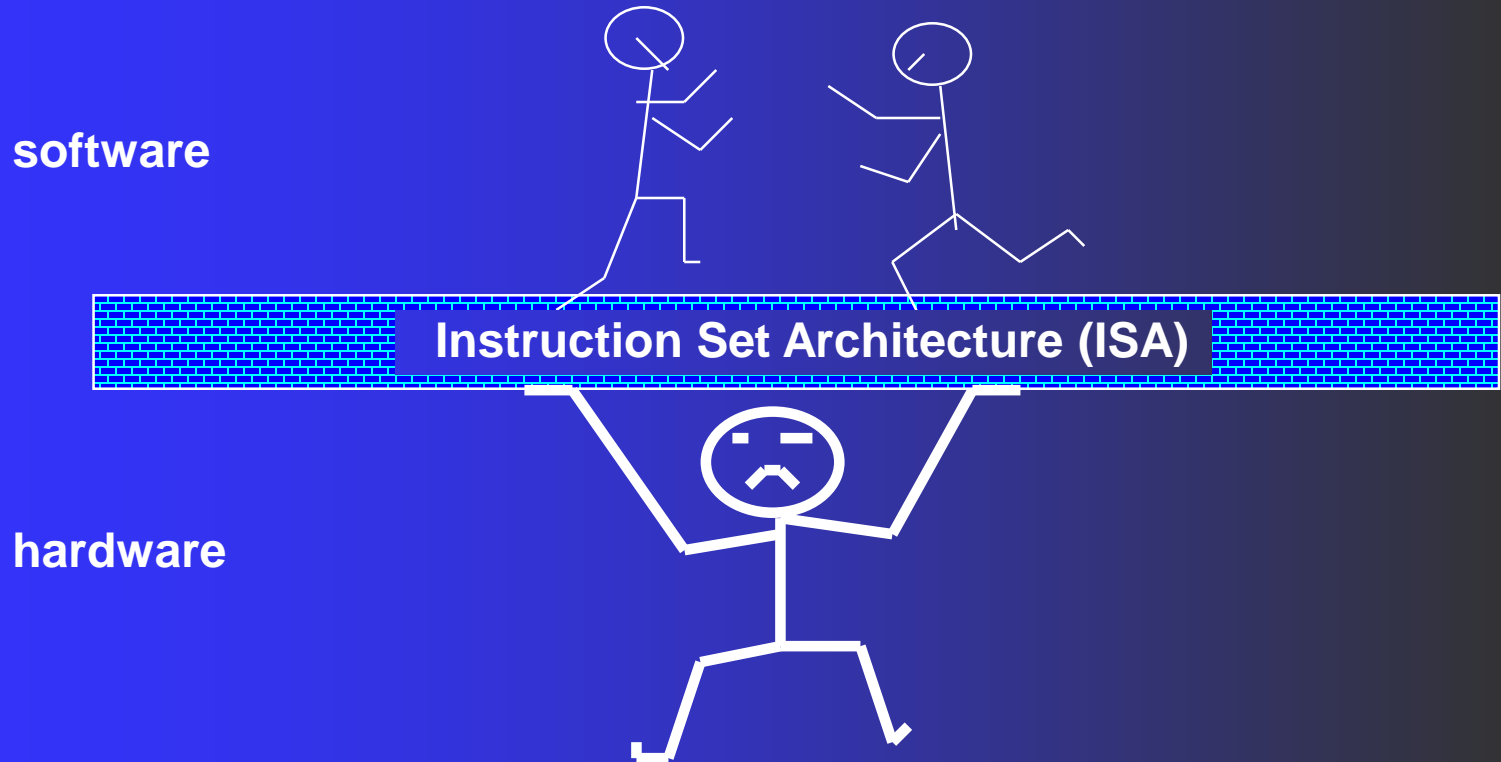
Application software



Can understand symbolic assembly language, e.g., add C, A, B

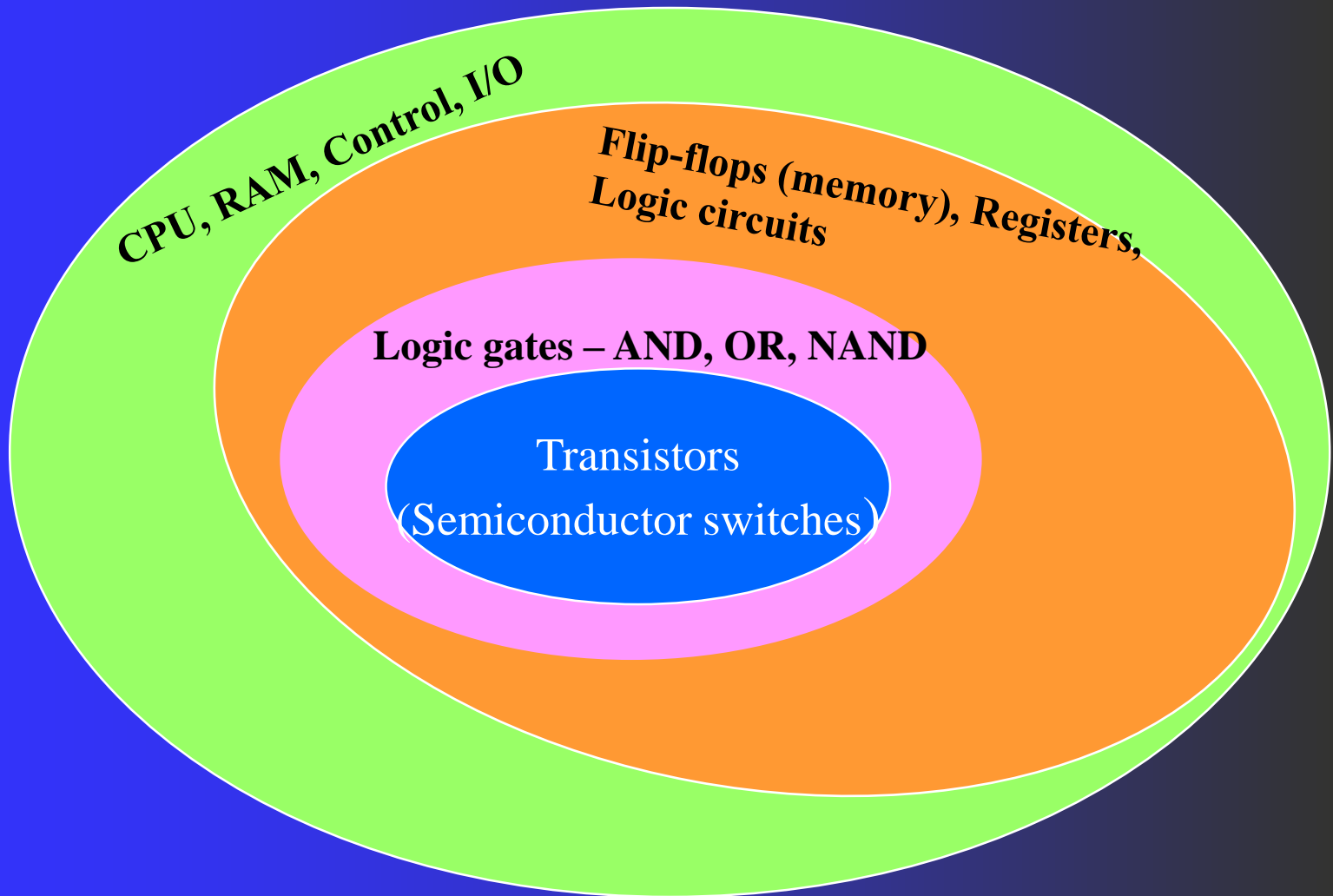
can understand only machine language – binary strings of 0's and 1's, (0101001100..) i.e., electrical *off* and *on* signals (Computer Organization)

Hardware-Software Interface



ISA: Collective attributes of the machine-language instruction set

Hardware Hierarchy



How a program is executed by a computer

Compiler

Assembler

*Application software,
a program in C:*

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

*Compiler translates it to
assembly language program
of the target machine*

swap;

mul	\$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	

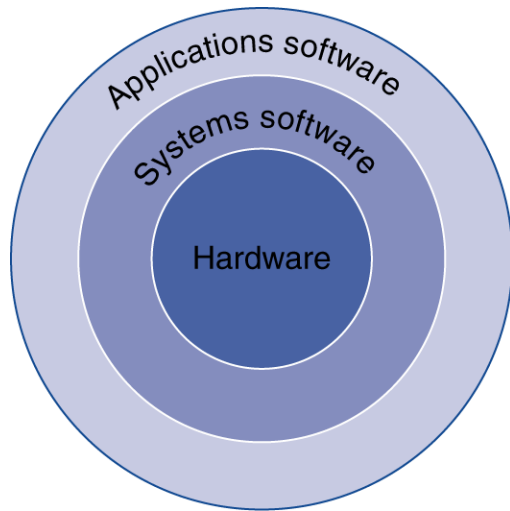
binary machine code

```
000000001010000100000000000011000
000000000000110000001100000100001
100011000110001000000000000000000
100011001111001000000000000000100
101011001111001000000000000000000
101011000110001000000000000000100
00000011111000000000000000001000
```

Hardware

output

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

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:
  muli $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)

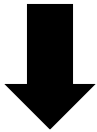
```
000000001010000100000000000011000
000000000000110000001100000100001
100011000110001000000000000000000
100011001111001000000000000000100
101011001111001000000000000000000
101011000110001000000000000000100
00000011111000000000000000001000
```


Program Execution

Program written in high-level language, e.g., *C*



Compiler



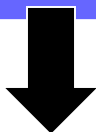
Assembly-language code

Assembler



Machine-language code (binary)

Hardware



Result

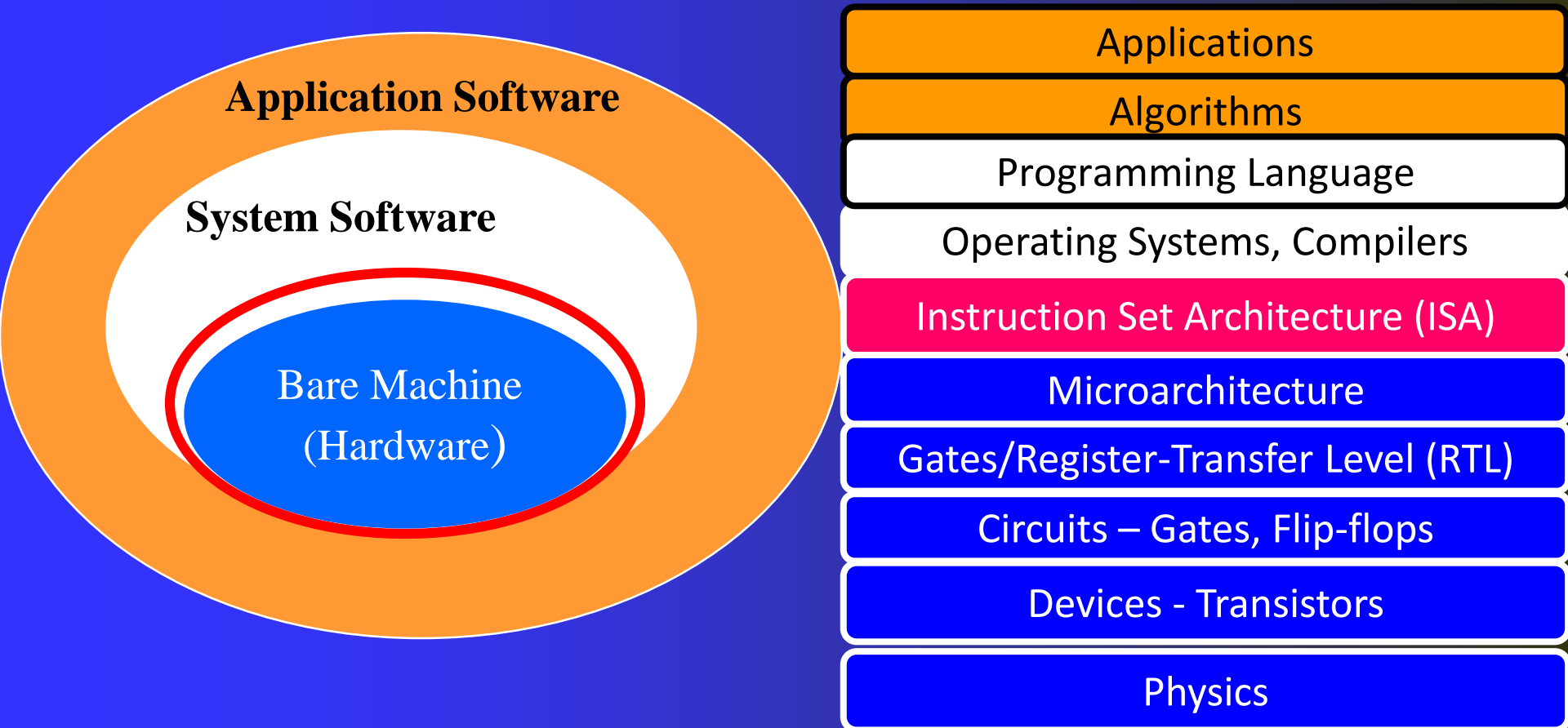
Question:

Where is the compiler running?

Who compiles the compiler?

How is the Assembler running?

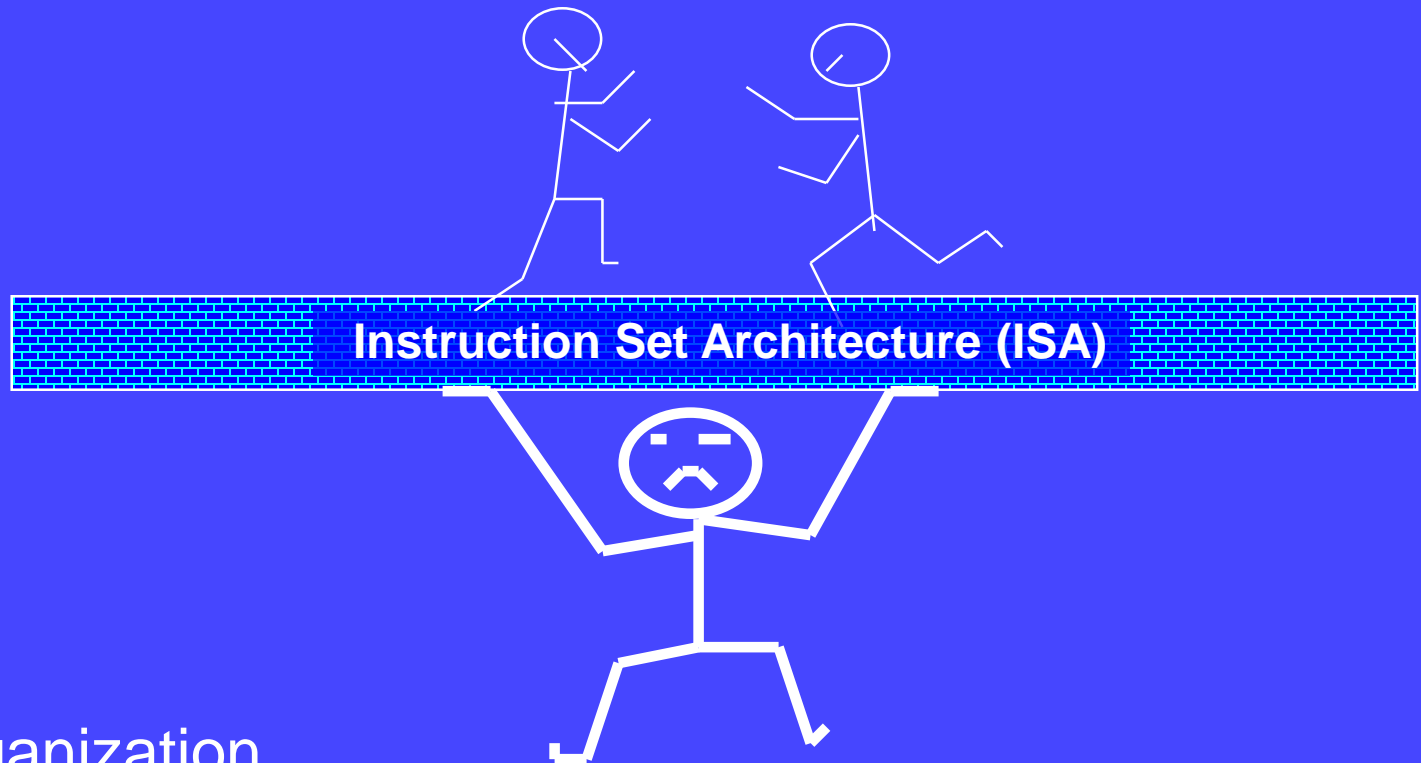
System Hierarchy



Computer Organization *versus* Computer Architecture

Computer Architecture

The view of a computer as perceived by software designers

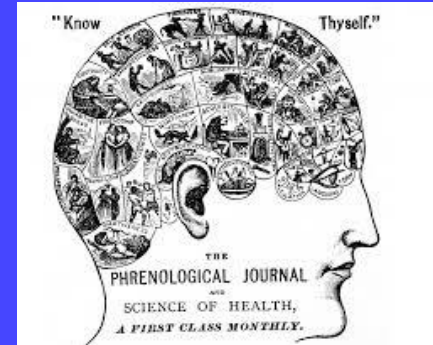
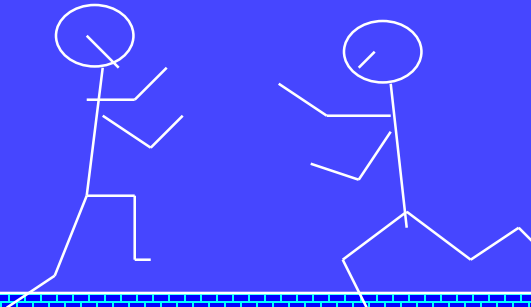


Computer Organization

The actual implementation of components in hardware

Computer Organization *versus* Computer Architecture

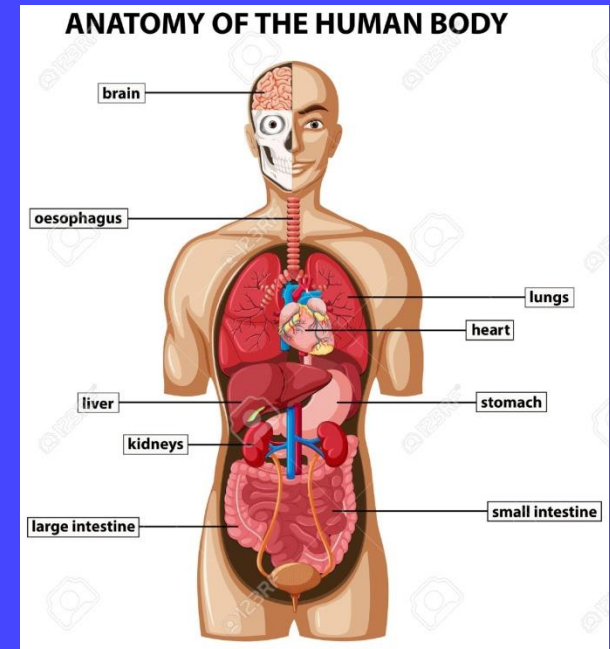
Computer Architecture: The personality of a machine
(behavioral view)



Instruction Set Architecture (ISA)



Computer Organization:
The anatomy of a machine
(structural view)



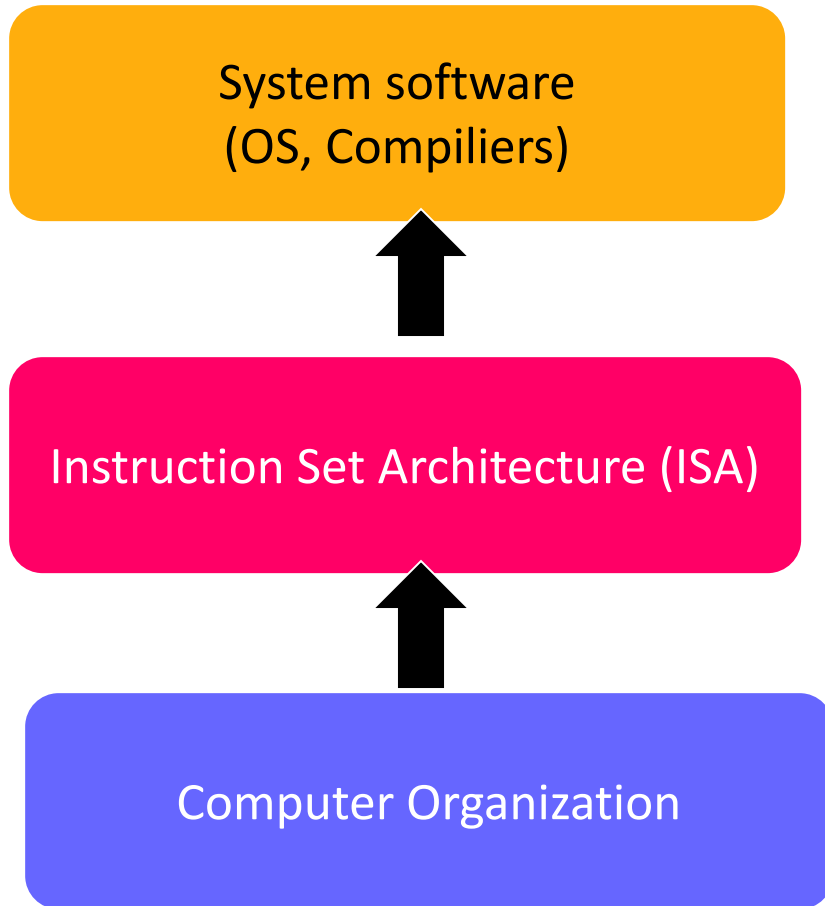
Computer Architecture

- Architecture is visible to a programmer
 - Instruction set
 - Registers
 - Data representation, addressing modes
 - I/O mechanisms
 - Virtual view of memory

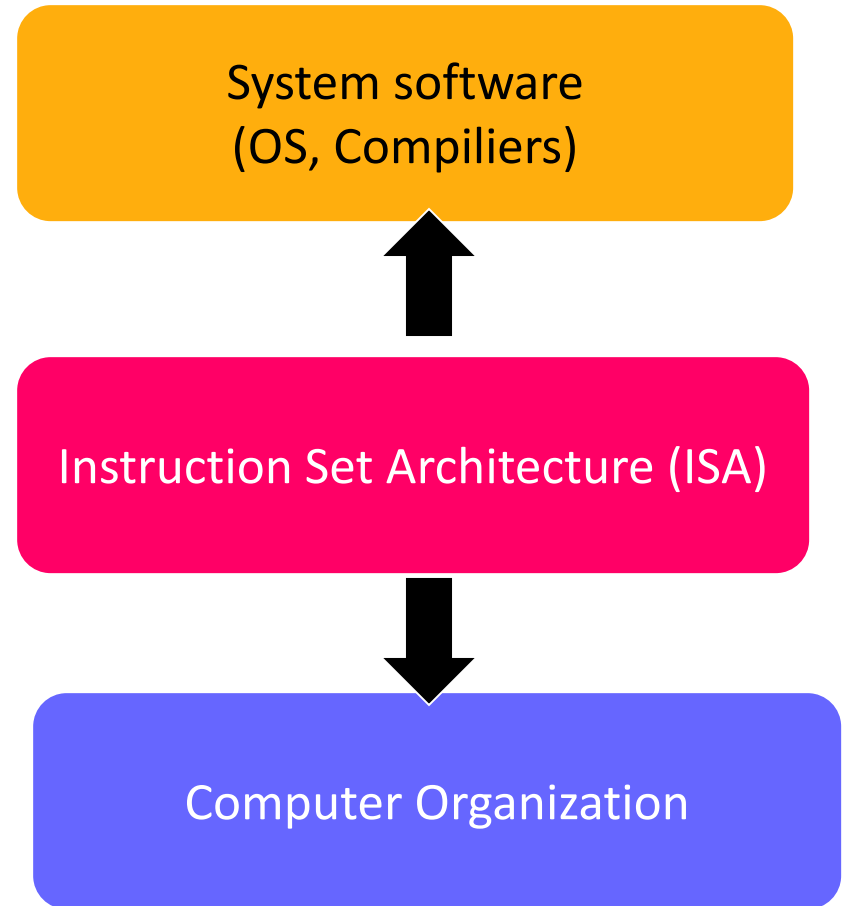
Computer Organization

- Organization is visible to hardware designer
 - Hardware implementation of an instruction
 - Registers, program counter
 - Arithmetic and logical units
 - Control logic, internal states
 - Pipelining hardware
 - Cache and main memory

Computer Design Approach



Bottom-up approach:
Hardware first, ISA later

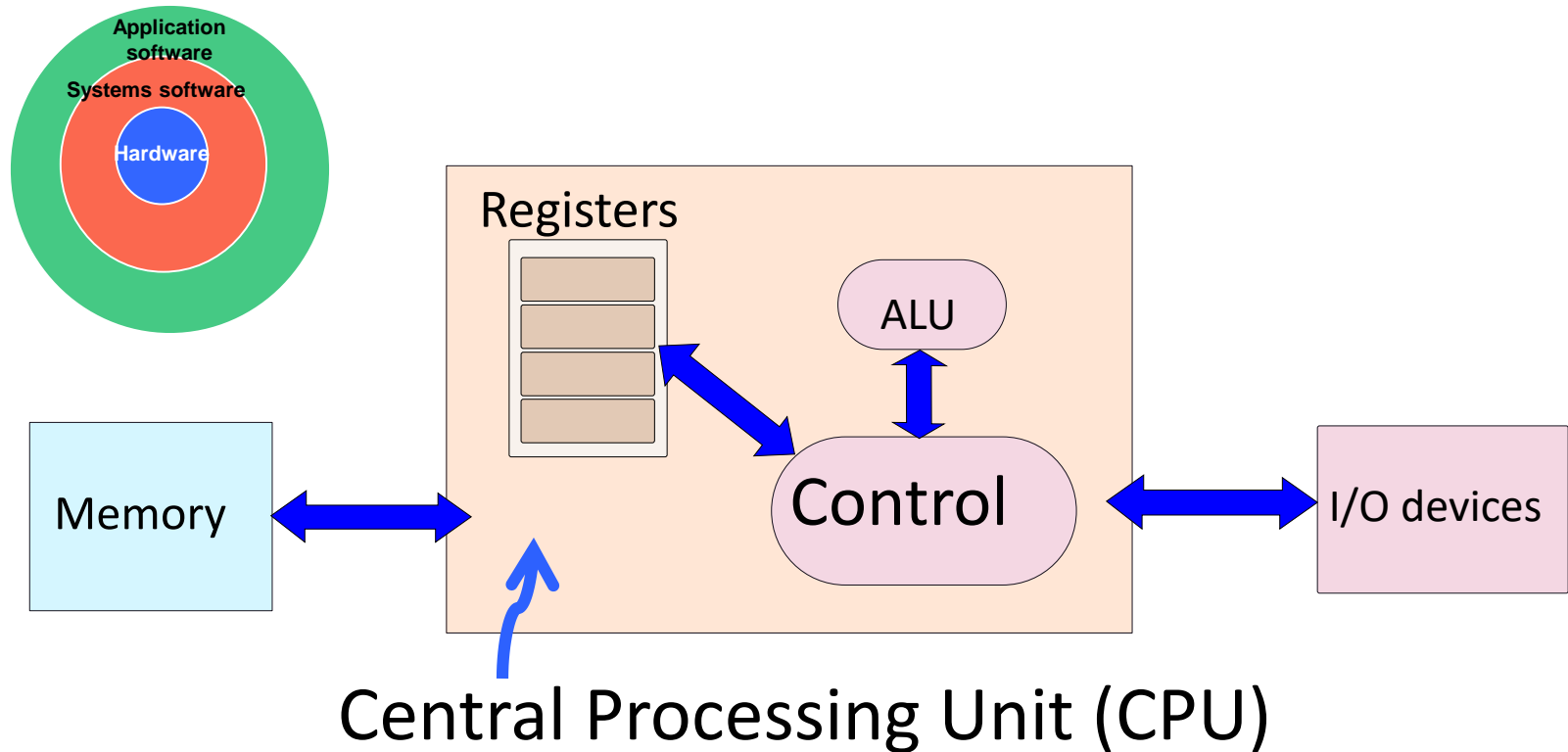


Top-down approach:
ISA first, hardware later

Instruction Set Architecture (ISA)

- A set of assembly language instructions that separates the interface between software and hardware
- In top-down design, given an ISA, an appropriate hardware platform is built to support it
- Based on ISA, OS and compilers are to be developed accordingly further going up
- Once ISA is fixed, software and hardware engineers can work independently
- ISA is designed to optimize the performance supported by the available hardware technology

The Hardware of a Computer



Five easy pieces:

Control, Arithmetic Logic Unit (ALU), Memory,
Input/Output, Datapath (Bus)

Next Class



- ❖ Model for computation and Turing Machine
- ❖ von Neumann Architecture
- ❖ Basic Features of Instruction Set Architecture (ISA)
- ❖ CPU Performance Equation
- ❖ Amdahl's Law
- ❖ RISC *versus* CISC