Computer Organization and Architecture

INTRODUCTION

Instructor :
Dr. Anjali Gautam
Department of Information Technology
IIIT Allahabad

Recommended Books

1. John L. Hennessy and David A. Patterson, Computer Architecture -- A Quantitative Approach, 5th Edition, Morgan Kaufmann Publications, Elsevier, Inc., 2012.
2. Carl Hamachar, Zvonco Vranesic and Safwat Zaky, Computer Organization, McGraw Hill
3. William Stallings, Computer Organization and Architecture: Designing for Performance, Pearson Education
4. John P. Hayes , Computer Architecture and Organization, McGraw Hill
5. Linda Null and Julia Lobur, The Essentials of Computer Organization and Architecture.

What is the course all about ???

What are the components of a computer and how do they work?

- Introduction
- How to program a computer?
- How to store different kinds of data in a computer ?
- How can I run my programs faster ?
- Use of techniques such as caching and pipelining
- How to work with multiple processors?
- What are GPUs and how can they improve performance?
- Can I build my own computer?

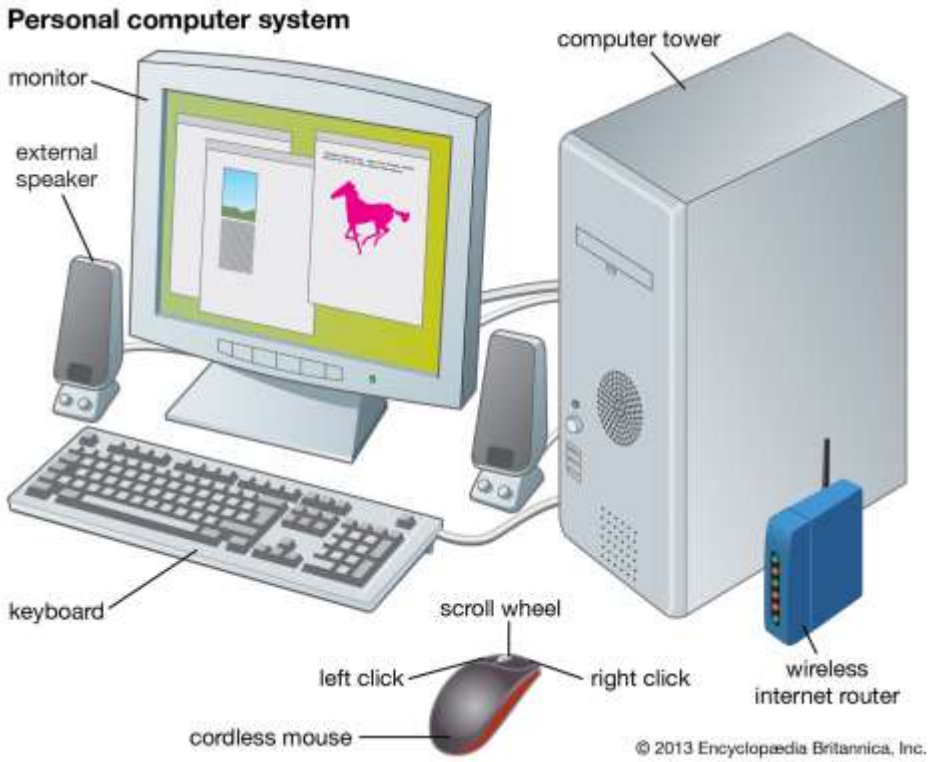
..... and more

Computer

The third revolution

Before that everything related to it “Computer Science Fiction”

- Computer in Automobiles
- Cell phones
- Human genome project
- www, etc.



Need for a Computer : Human Efficiency Vs. Machine Performance

Q: If I am devising the solution (algorithm) to a problem, why do I need a machine?

Ans: The machine performs the task in lesser time and more accurately

Ex: Compare the task of sorting 1000 numbers when performed by a human and using a computer.

A computer clearly outperforms the human.

Classes of Computing Applications

Personal computer

Server

Supercomputer

Embedded computer

High-performance computing (HPC) Vs Supercomputer

While supercomputing typically refers to the process of complex and large calculations used by supercomputers.

HPC is the use of multiple supercomputers to process complex and large calculations.

However, both terms are often used interchangeably.

Computer Architecture and Computer Organization

Computer Architecture:

- The view of a computer presented to S/W designers
- *Architecture describes what the computer does.*

Computer Organization:

- The actual implementation of a computer in H/W.
- *The organization describes how it does it.*

Computer architecture

Computer architecture refers to those attributes of a system visible to a programmer or, put another way, those attributes that have a direct impact on the logical execution of a program.

A term that is often used interchangeably with computer architecture is **instruction set architecture (ISA)**.

- It defines instruction formats, instruction opcodes, registers, instruction and data memory;
- The effect of executed instructions on the registers and memory; and an algorithm for controlling instruction execution.

Computer organization

Computer organization refers to the operational units and their interconnections that realize the architectural specifications.

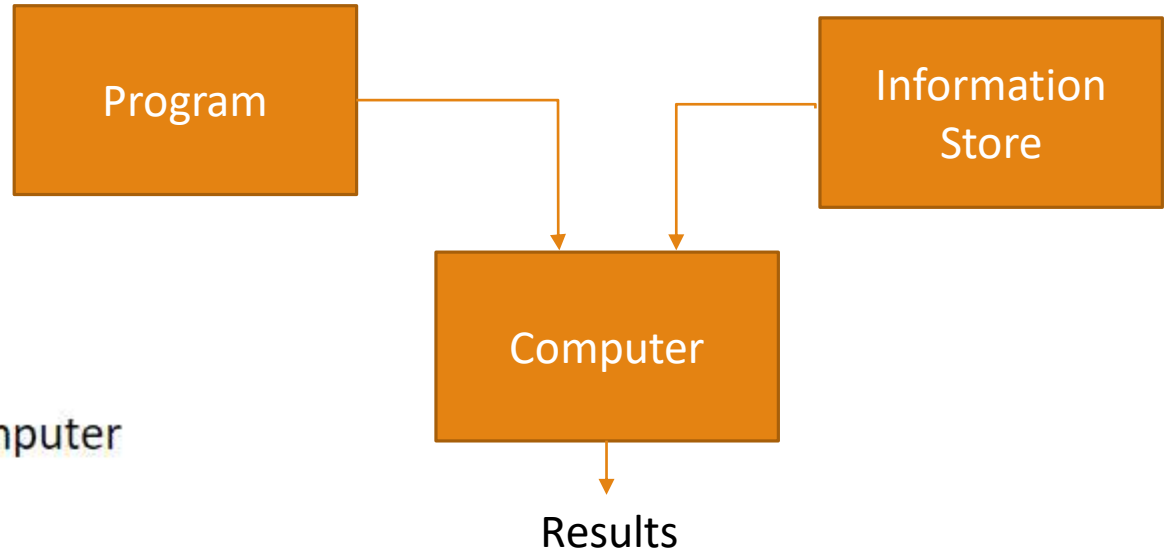
Example

- it include those hardware details transparent to the programmer, such as control signals; interfaces between the computer and peripherals; and the memory technology used.

Computer

It is a general purpose device that can be programmed to process information and yield meaningful results

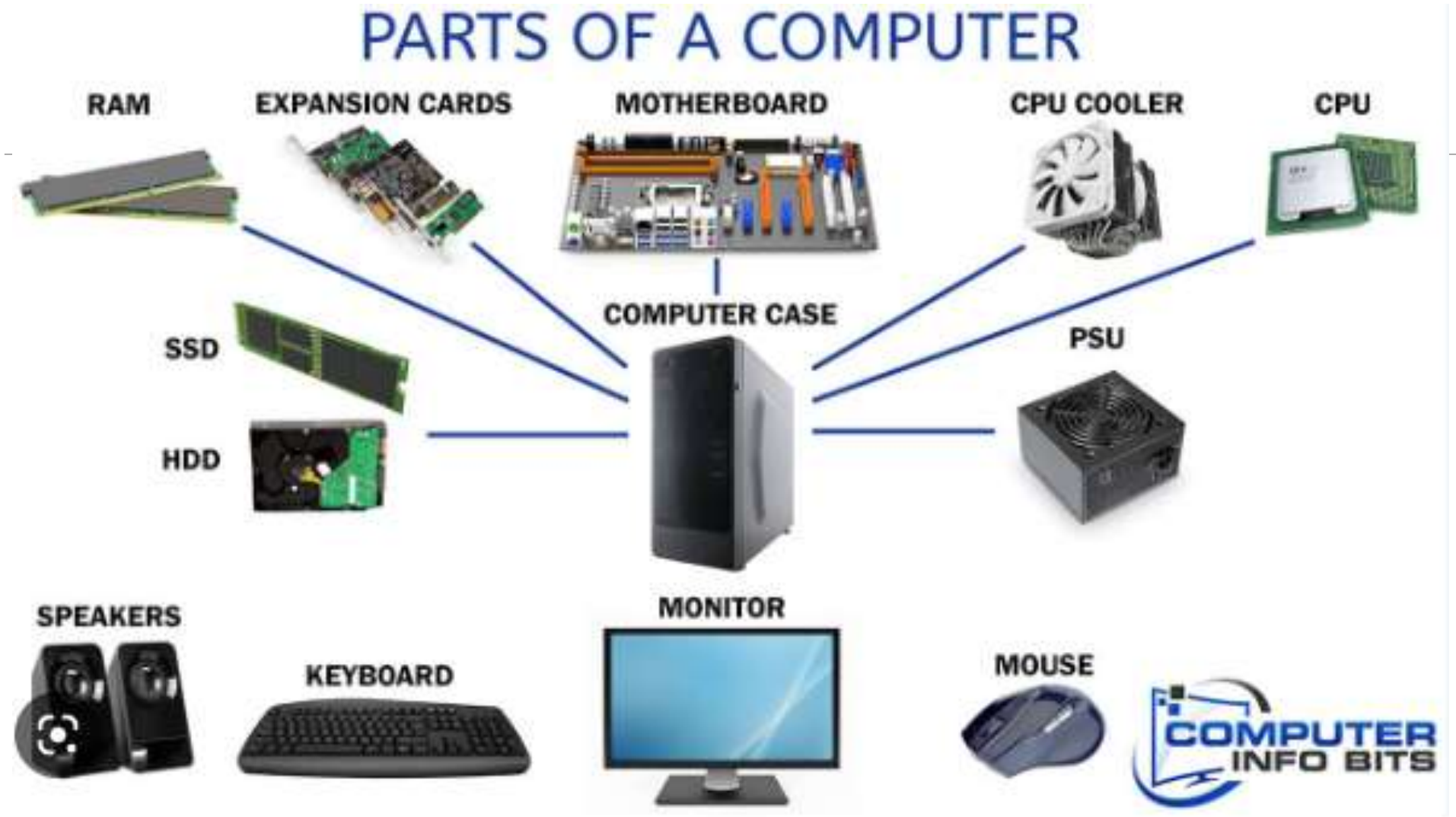
It is a machine that can be programmed to carry out sequences of arithmetic or logical operations (computation) automatically.



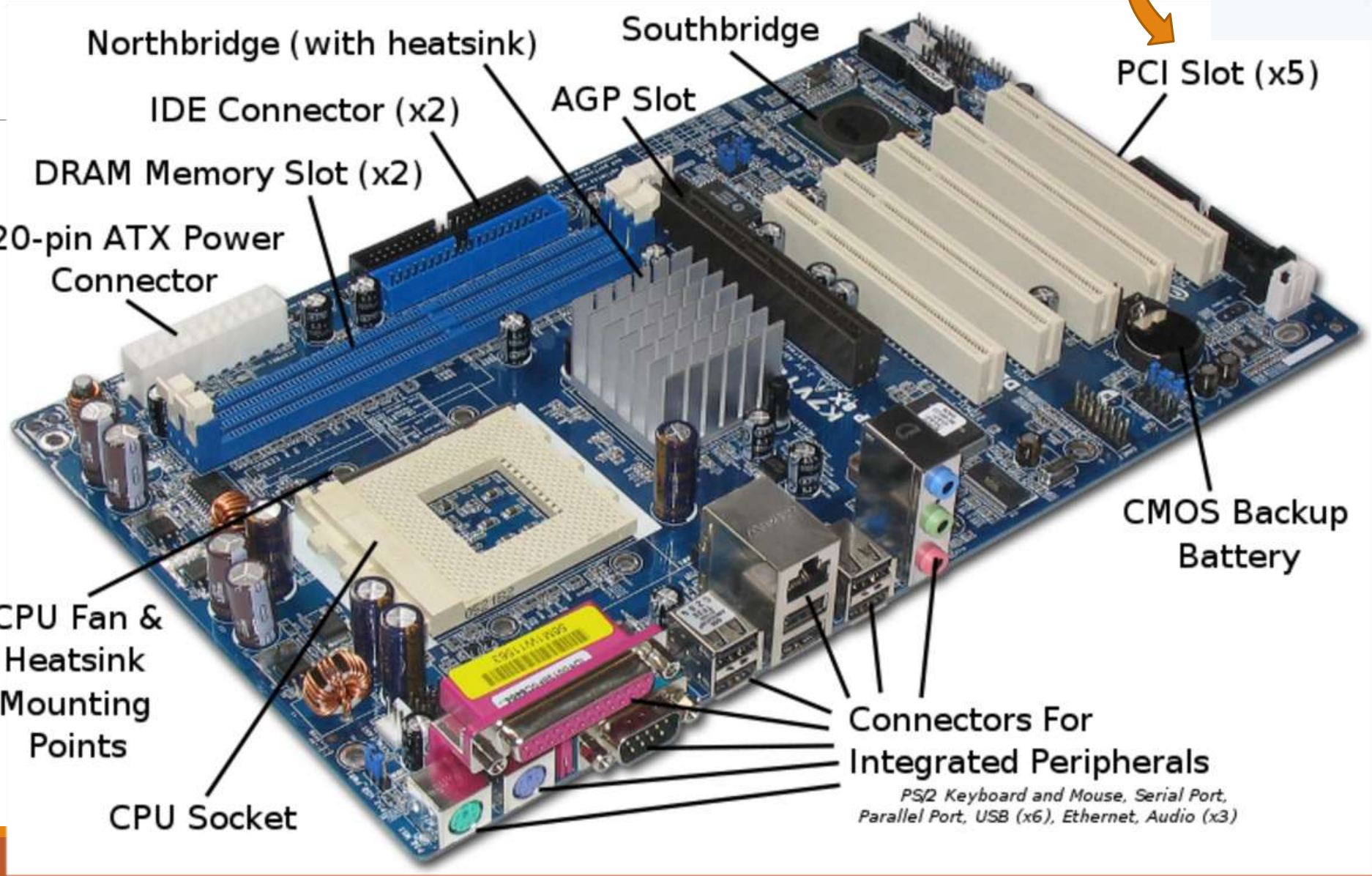
Program – List of instructions given to the computer

Information store – data, images, files, videos

Computer – Process the information store according to the instructions in the program


PARTS OF A COMPUTER
RAM
EXPANSION CARDS
MOTHERBOARD
CPU COOLER
CPU
SSD
COMPUTER CASE
PSU
HDD
SPEAKERS
MONITOR
KEYBOARD
MOUSE
COMPUTER
INFO BITS

Computer Motherboard



https://turbofuture.com/computers/the-motherboard-components

The von Neumann Architecture

Based on the concept of stored program machine architecture.

It refers to the storage of instructions in computer memory to enable it to perform a variety of tasks

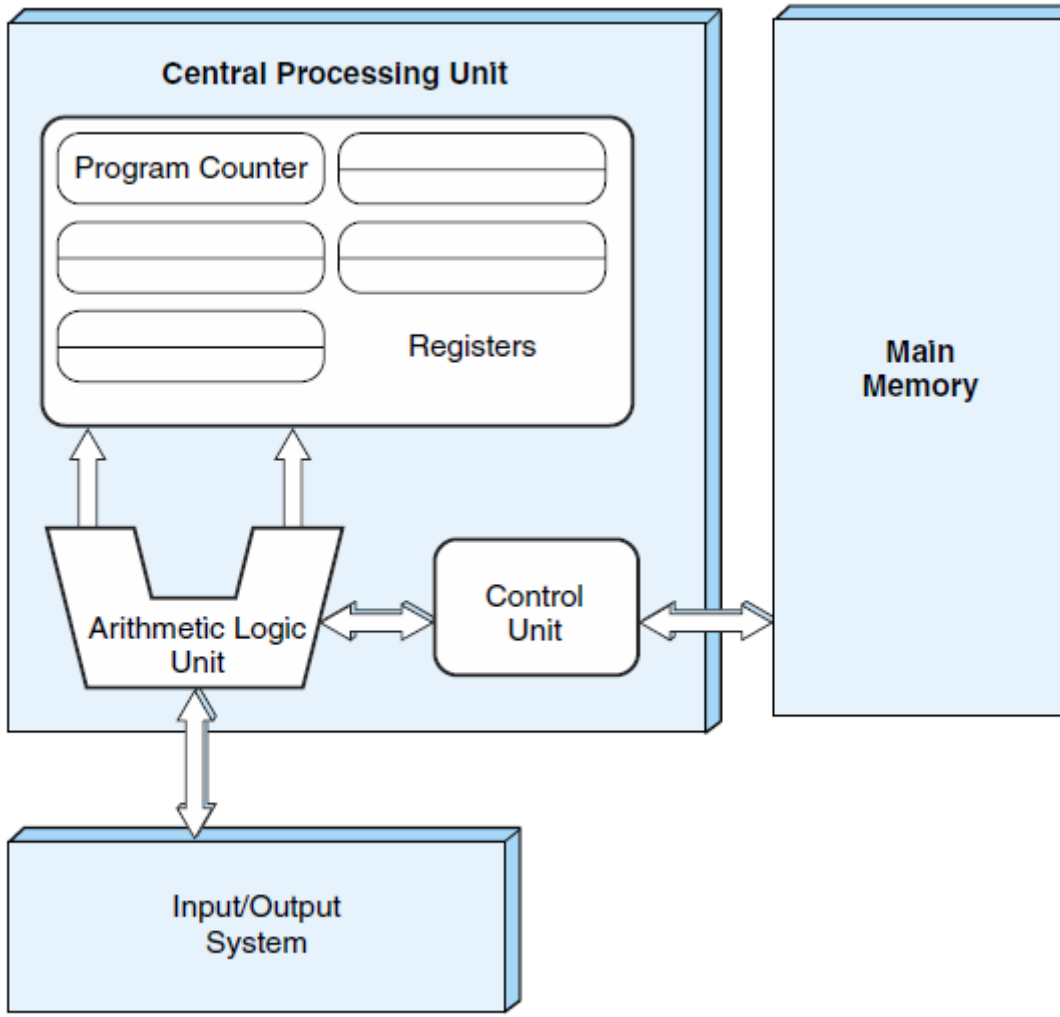


FIGURE 1.4 The von Neumann Architecture

Modified

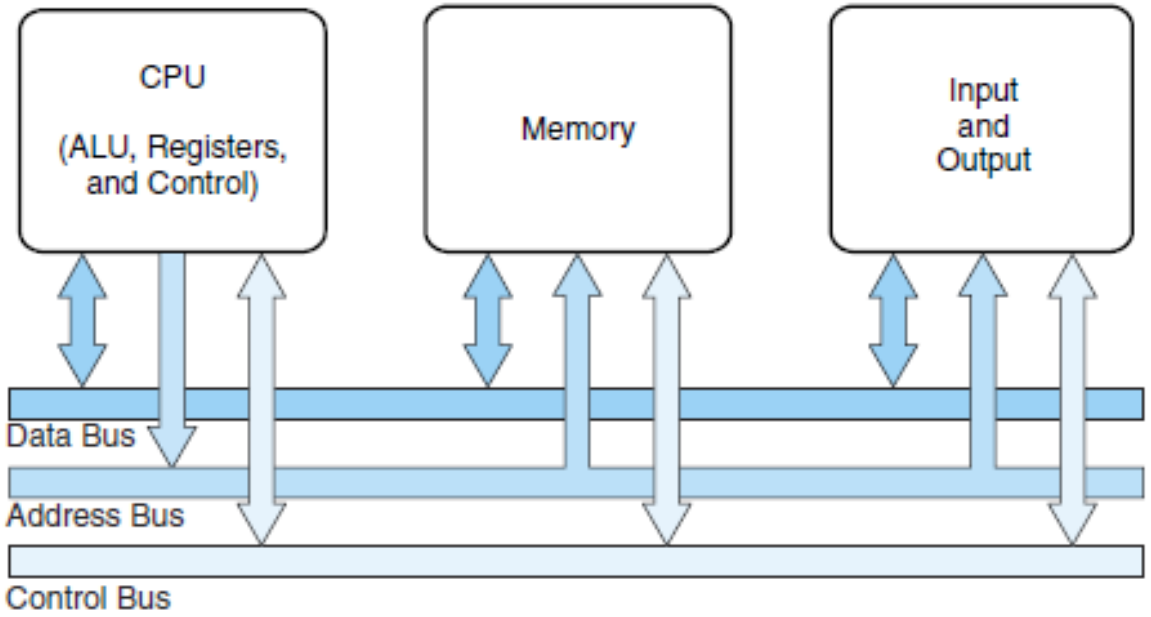


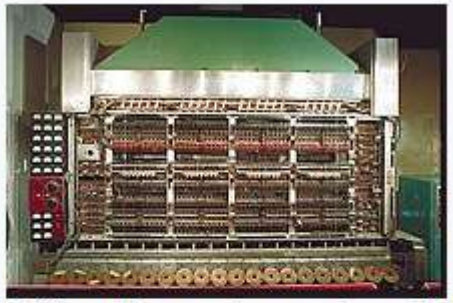
FIGURE 1.5 **The Modified von Neumann Architecture, Adding a System Bus**

IAS Computer

The most famous first-generation computer, known as the IAS computer (the Institute for Advanced Study, Princeton)

It is known as the *stored-program concept*.

IAS machine



The IAS machine an early computer on display at the Smithsonian Institution

Developer	John von Neumann
Manufacturer	Institute for Advanced Study (IAS)
Release date	June 10, 1952; 70 years ago
Lifespan	1952–1958
CPU	1,700 vacuum tubes
Memory	1,024 words (5.1 kilobytes) (Williams tubes)
Mass	1,000 pounds (450 kg)

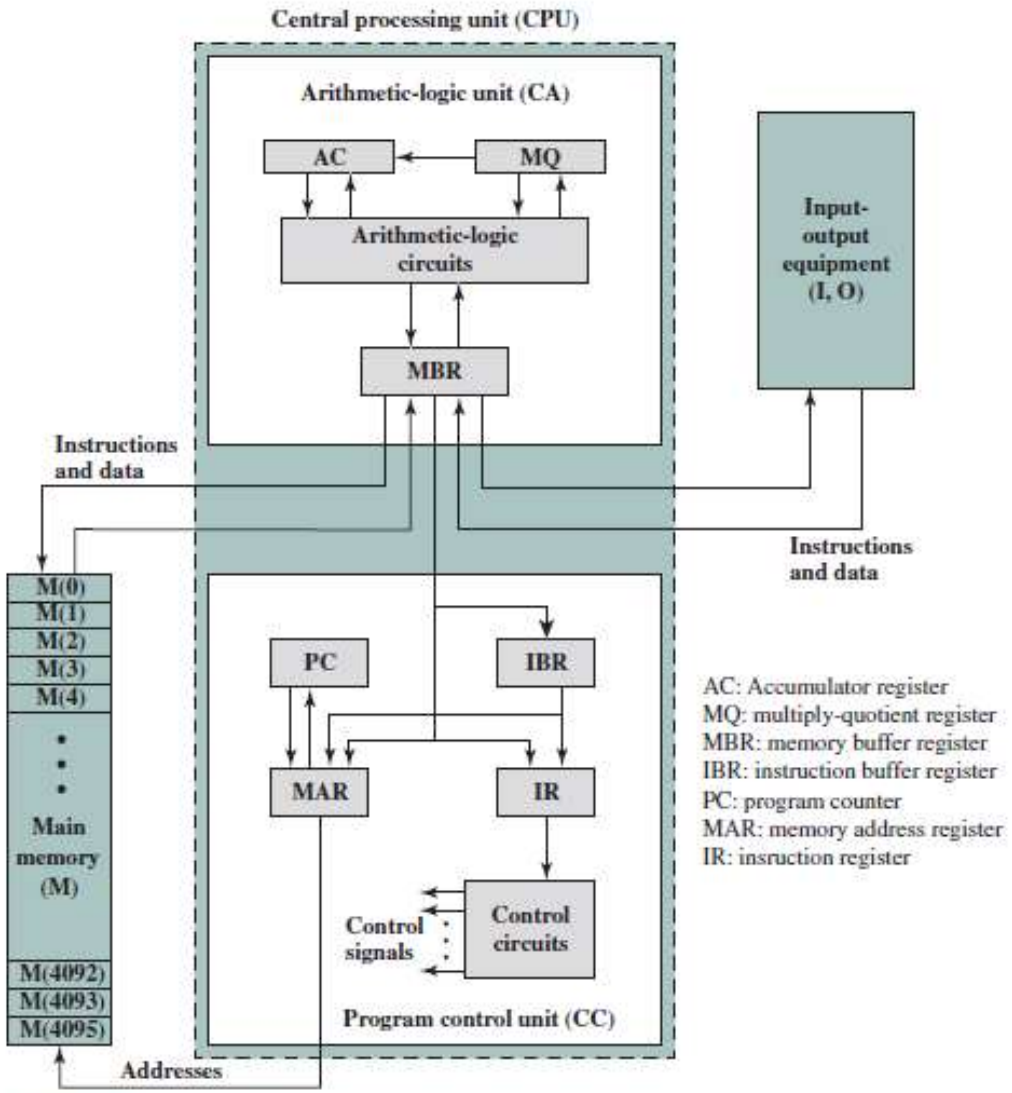


Figure 1.6 IAS Structure

Structure and Functions of Computer

Structure: The way in which the components are interrelated.

Function: The operation of each individual component as part of the structure.

Function

Data processing

Data storage

- the computer must temporarily store at least those pieces of data that are being worked on at any given moment. Thus, there is at least a short-term data storage function.

Data movement

- There are devices that serve as either sources or destinations of data.
  - When data are received from or delivered to a device that is directly connected to the computer, the process is known as *input–output* (*I/O*), and the device is referred to as a *peripheral*.
  - When data are moved over longer distances, to or from a remote device, the process is known as *data communications*.

Control

- Within the computer, a control unit manages the computer's resources and orchestrates the performance of its functional parts in response to instructions.

Structure

Simple single-processor computer provides a hierarchical view of the internal structure.

There are four main structural components:

- Central processing unit (CPU),
- Main memory,
- I/O,
- System interconnection

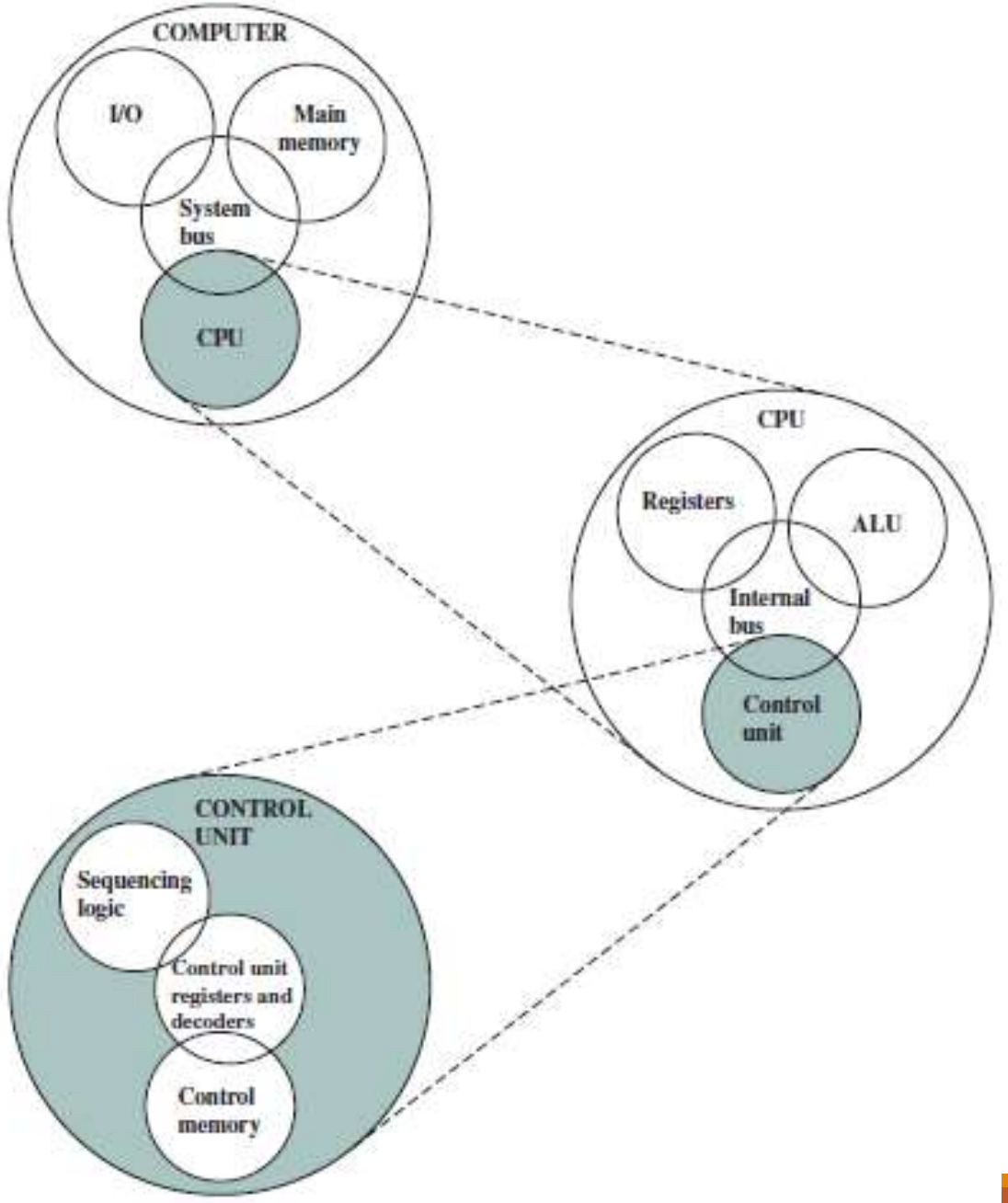


Figure 1.1 The Computer: Top-Level Structure

Inside CPU

Control unit: Controls the operation of the CPU and hence the computer.

Arithmetic and logic unit (ALU): Performs the computer's data processing functions.

Registers: Provides storage internal to the CPU.

CPU interconnection: Some mechanism that provides for communication among the control unit, ALU, and registers.

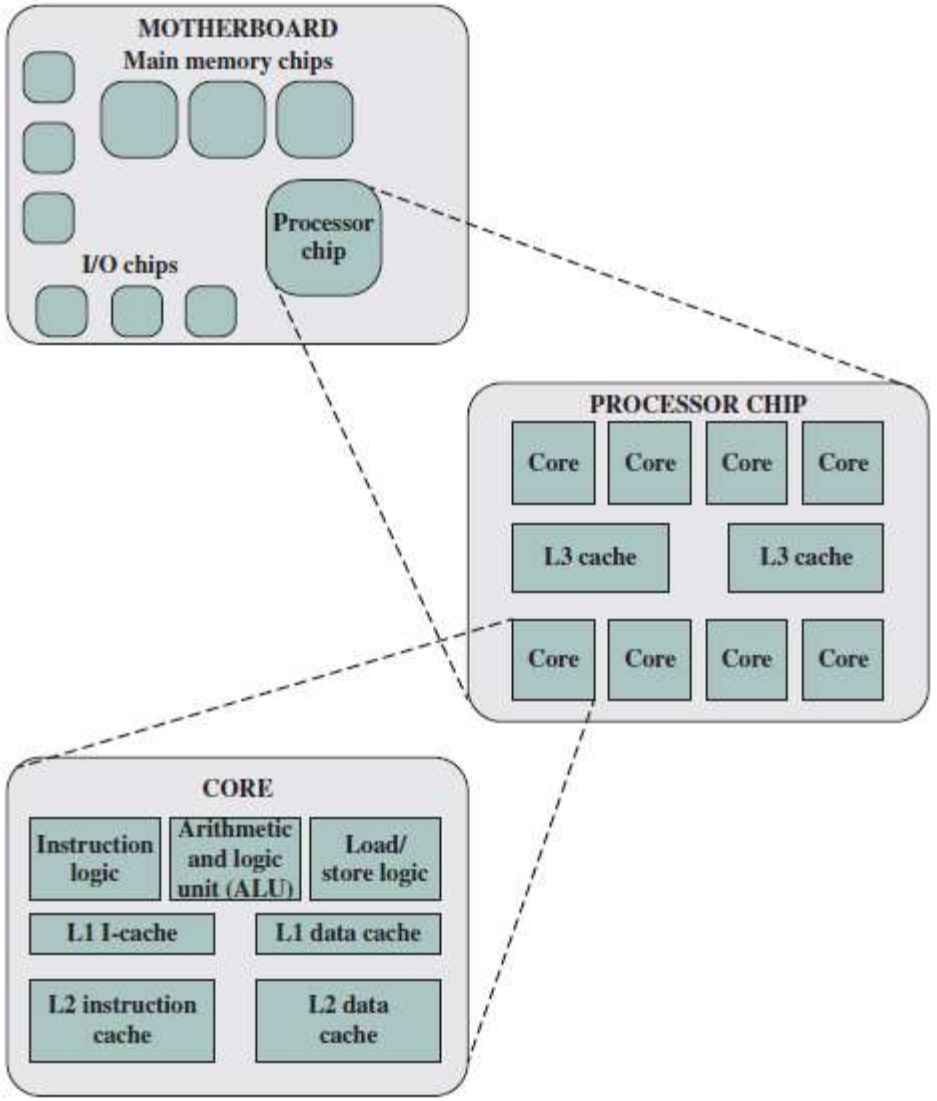
Multicore Computer Structure

Computers generally have multiple processors.

When these processors all reside on a single chip ➔ *multicore computer*, and

each processing unit (consisting of a control unit, ALU, registers, and perhaps cache) is called a *core*.

Major Elements of a Multicore Computer



Storage elements – Primary Memory

Memory unit is used to store the data and program.

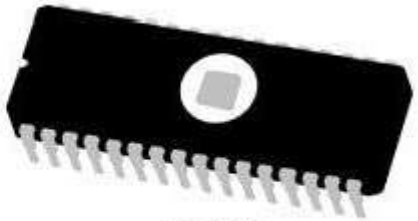
CPU can work with the information stored in memory unit. This memory unit is termed as primary memory or main memory module. These are basically semi conductor memories

Volatile Memory : RAM (Random Access Memory).

Non-Volatile Memory : ROM (Read only Memory), PROM (Programmable ROM) EPROM (Erasable PROM), EEPROM (Electrically Erasable PROM).

RAM
Random Access Memory

Vs



ROM
Read only Memory

Memory Organization

Memory consists of millions of storage cells each storing a bit (0/1)

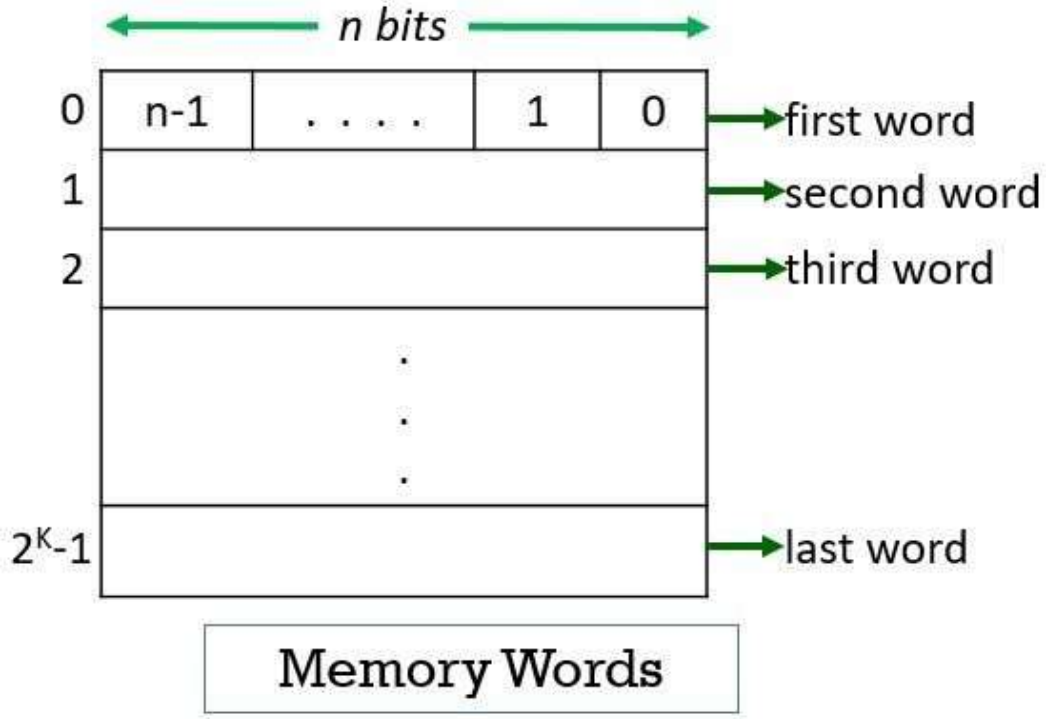
of information

Bits are held in groups of fixed size - requires basic operation

Each *n-bit* group is called a *Word (n* describes the word length *)*

Memory is a collection of words

Each location of memory has an unique address



Storage Elements – Secondary Memory

Secondary memories are non volatile memory and it is used for permanent storage of data and program. Example of secondary memories: Hard Disk, USB Drives, CD-ROM

Input Unit

Program or data is read into main storage from input device or secondary storage under the control of CPU input instruction

Output Unit

Used to Provide results to the user.

Data from main storage in transferred to the output units under control of CPU output instructions

History of Computer – Past to Present

Mathematical tables were created by people called ’computers’.

They were large charts showing the results of calculations, e.g. multiplication, division, and trigonometry

Charles Babbage noticed errors in mathematical tables (1820)

Created the difference engine to compute this math more accurately i.e. add, subtract, polynomial functions

1st Generation Computers

1951 – 1959

Based on Vacuum tubes

Vacuum tubes: Control electric current using the vacuum, and

Can be used to start/stop, or change the flow based on the current

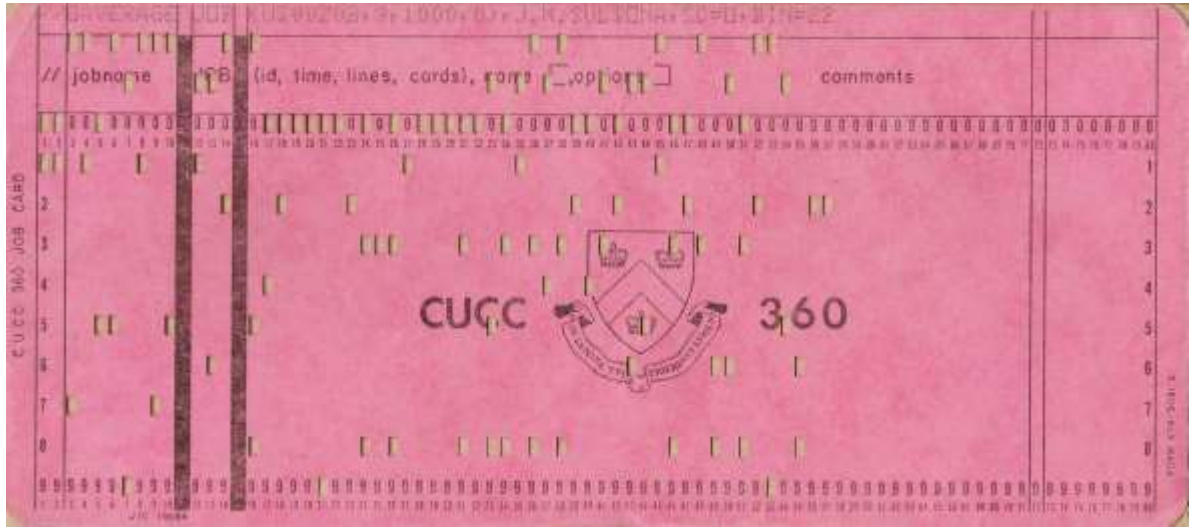
Second Generation Computers

Based on Transistors

1959-1965

Stored instructions in memory

Relied on punch cards for input and printers for output



IBM
026
Card
Punch

IBM
2501
Card
Reader

collection of Joe Sulsona.

The cards were fed into a card reader connected to a computer, which converted the sequence of holes to digital information.

Transistors

Replaced vacuum tubes

Invented at Bell laboratories

Enabled computers to be smaller, cheaper, more reliable, and efficient

Transistors work as switches on current, turning it on or off (like binary 0 or 1).

Still generate a lot of heat, but less than vacuum tubes

Third Generation Computers

Modern computers

1965-1971

Used Integrated Circuits

Keyboards instead of punch cards

Monitors for display

Different applications used through operating system

ICs are small chips containing thousands of transistors

Invented by Jack Kilby, Nobel Laureate of Physics

Fourth Generation Computers

Microprocessor

Development of the personal computer

1971 – 1980

Addition of GUI's, the mouse, and handheld devices

Microprocessor

Thousands of Integrated Circuits were built on a silicon chip.

Created by Intel corp.

Becomes the Central Processing Unit (CPU)

Allow computers to be smaller, more powerful, faster, and cheaper

Fifth Generation Computers

1980 - Present

Enhancement of Artificial Intelligence

Nanotechnology

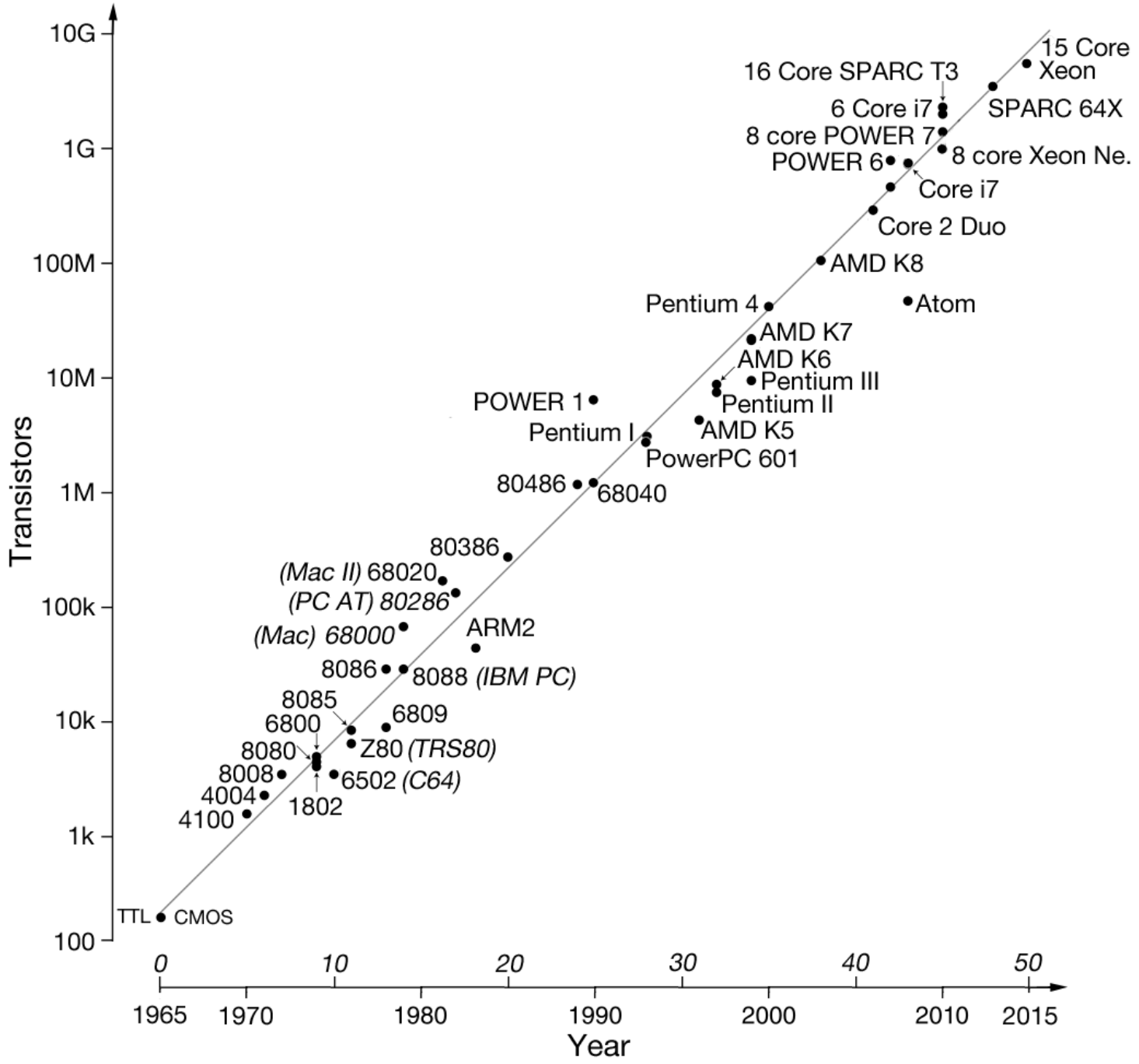
Natural Language Processing

…

Moore's law

- **Increased density of components on chip**
- **Gordon Moore – co-founder of Intel**
- **Number of transistors on a chip will double every year**
- **Since 1970's development has slowed a little**
  - Number of transistors doubles every 18 months
- **Cost of a chip has remained almost unchanged**
- **Higher packing density means shorter electrical paths, giving higher performance**
- **Smaller size gives increased flexibility**
- **Reduced power and cooling requirements**
- **Fewer interconnections increases reliability**

Contd.


10G
1G
100M
10M
1M
100k
10k
1k
100
Transistors
15 Core Xeon
16 Core SPARC T3
SPARC 64X
6 Core i7
8 core POWER 7
8 core Xeon Ne.
POWER 6
Core i7
Core 2 Duo
AMD K8
Atom
Pentium 4
AMD K7
AMD K6
Pentium III
Pentium II
POWER 1
Pentium I
AMD K5
PowerPC 601
80486
68040
80386
(Mac II) 68020
(PC AT) 80286
(Mac) 68000
ARM2
8086
8088 (IBM PC)
8085
6809
6800
Z80 (TRS80)
8080
8008
6502 (C64)
4004
1802
4100
TTL
CMOS
0
10
20
30
40
50
1965
1970
1980
1990
2000
2010
2015
Year

Generation Of Computers 1st To 5th

First Generation 1946-1959

Second Generation
1959-1965

Third Generation
1965-1971

Fourth Generation
1971-1980

Fifth Generation 1980- Present

https://studymuch.in/generation-of-computer-1st-to-5th-with-details/

How does the computer work ?

Computer needs to be programmed to do user defined tasks

Programming is the process of writing instructions in a language that can be understood by the computer so that a desired task can be performed by it

Program: sequence of instructions to do a task, computer processes the instructions sequentially one after the other

Software: programs for doing tasks on computers

CPU understands machine language

- Different strings of 0's and 1's (Hard to remember)
- Mnemonic names of the strings - Instruction set

Alternate way to instruct CPU – use high level language

Machine Instructions

Machine Instructions are commands or programs written in machine code of a machine (computer) that it can recognize and execute.

It consists of several bytes in memory that tells the processor to perform one machine operation.

The processor looks at machine instructions in main memory one after another, and performs one machine operation for each machine instruction.

The collection of machine instructions in main memory is called a **machine language program** which are executed directly by a CPU.

Each instruction performs a very specific task, such as a load, a jump, or an ALU operation on a unit of data in a CPU register or memory.

```
MUL R5, R0, R1
DIV R6, R2, R3
```

Add 1,3
0b1000000100010011

family	operation	input1	input2
10	000001	0001	0011

Programming a Computer using High Level Language

- **Instruction set of different CPUs are different**
  - Need to write different programs for computers with different types of CPUs even to do the same thing
- **Solution – High Level Language (C, C++, Java, ...)**
  - CPU neutral – One program for many
  - Compiler – to convert from high level program to low level program that the computer understands

High Level Program

```
Variables x, y;
Begin
Read (x);
Read (y);
If (x >y) then Write (x)
    else Write (y); End.
```

Low Level Program

```
0: Start
1: Read 20
2: Read 21
3: Compare 20, 21, 22
4: J_Zero 22, 7
5: Write 20
6: Jump 8
7: Write 21
8: Halt
```

High-level Language

```
temp   = v[k];
v[k]   = v[k+1];
v[k+1] = temp;
```

```
TEMP   = V(K)
V(K)   = V(K+1)
V(K+1) = TEMP
```

C/Java Compiler

Fortran Compiler

Assembly Language

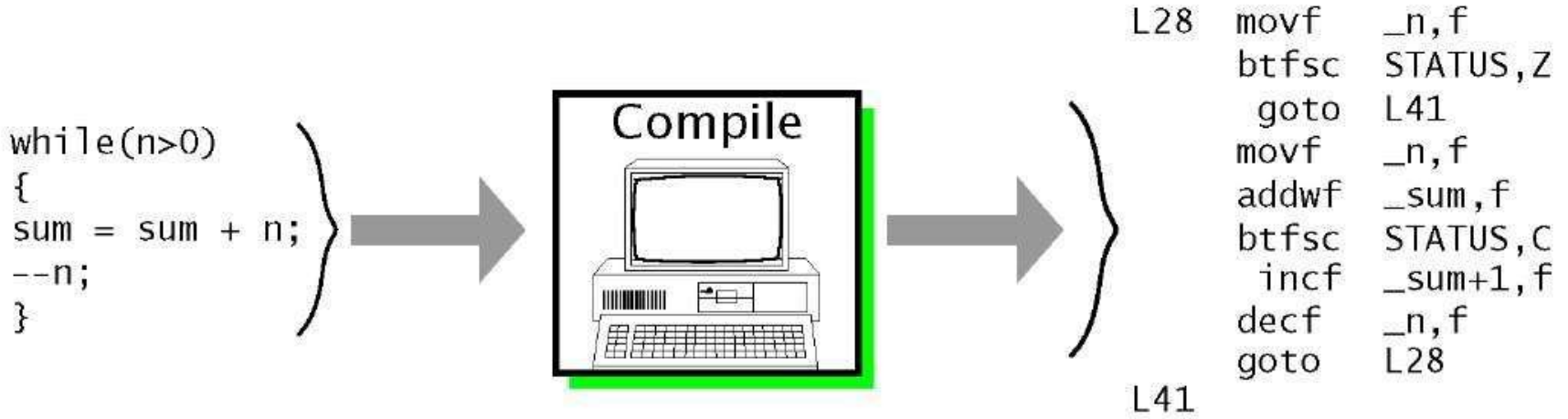
```
lw  $to,  0($2)
lw  $t1,  4($2)
sw  $t1,  0($2)
sw  $t0,  4($2)
```

MIPS Assembler

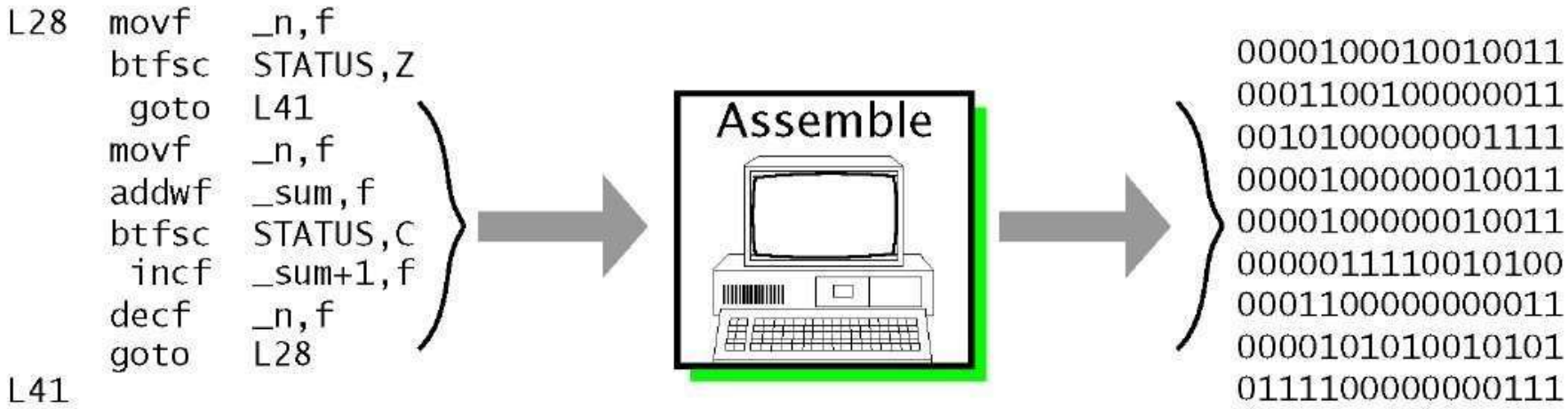
Machine Language

```
0000 1001 1100 0110 1010 1111 0101 1000
1010 1111 0101 1000 0000 1001 1100 0110
1100 0110 1010 1111 0101 1000 0000 1001
0101 1000 0000 1001 1100 0110 1010 1111
```

HLL – Assembly – Machine Code

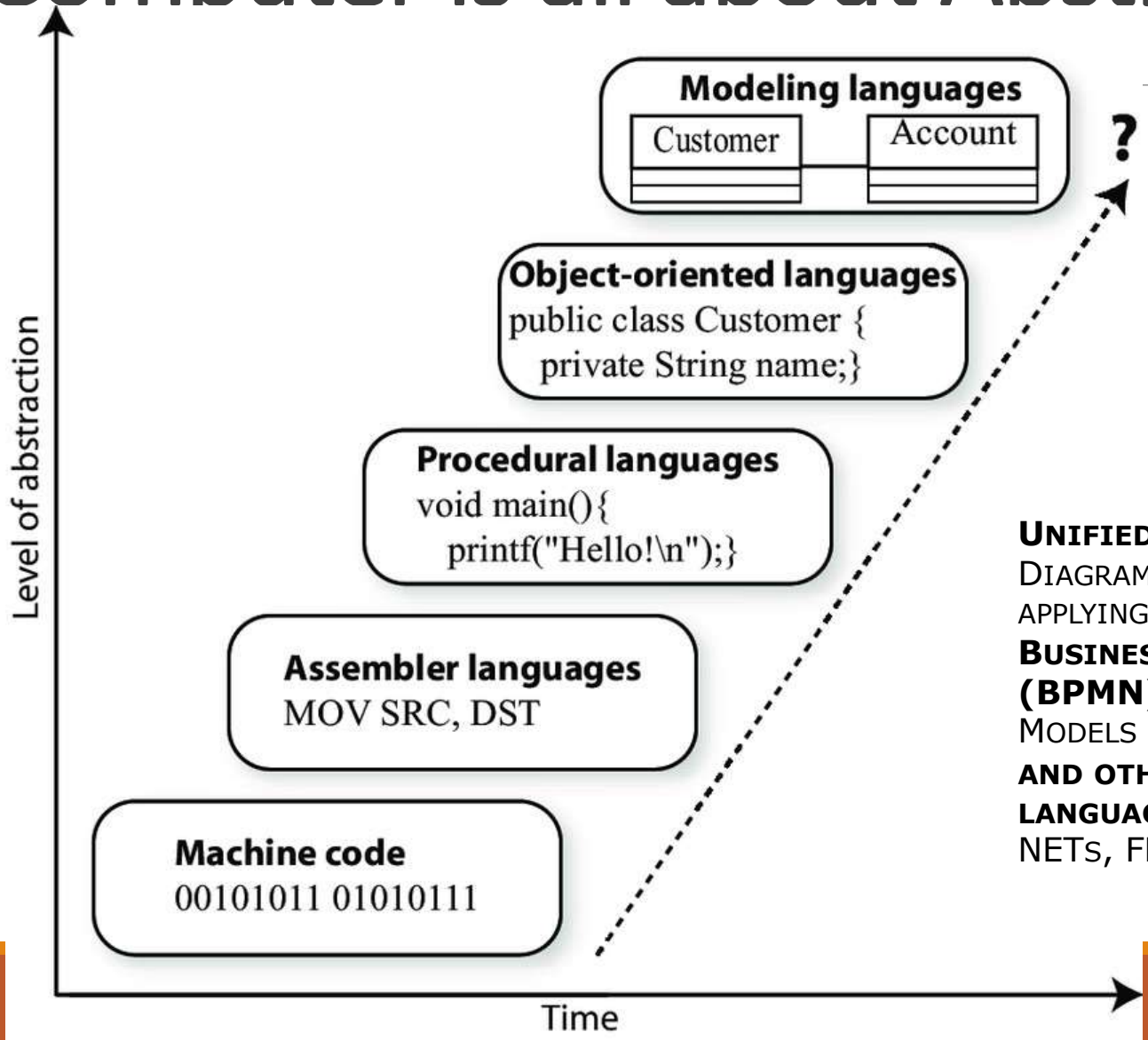


(a) First, compile to assembly-level code.



(b) Second, assemble-link to machine code.

Computer is all about Abstraction



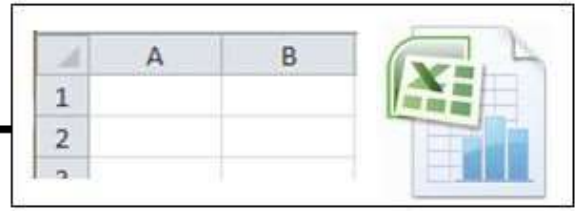
Unified Modeling Language (UML): UML, About, 2, UML Diagrams, 0, 1, 2, 3, Enterprise Architect UML Tutorial, (• applying UML)

Business Process Model and Notation (BPMN): BPMN, About, PDF, Altova's Enterprise Architect BPMN Models (• applying BPMN)

and other graphical modeling languages: EXPRESS, EEML, ORM, Petri NETs, FMC, LePUS3, Spec. and Desc. Language, ESL-Energy

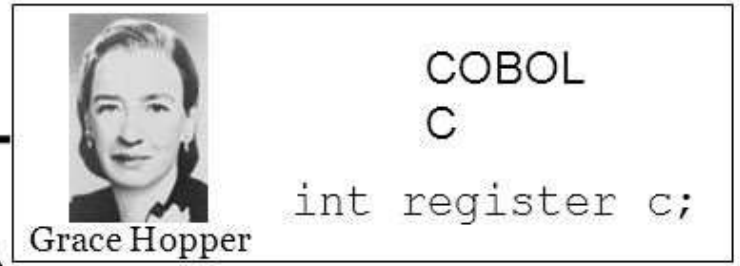
The Ladder of Abstraction

application software

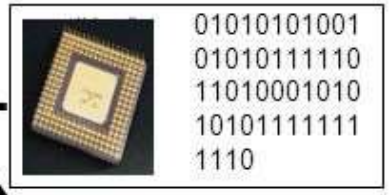


high-level language

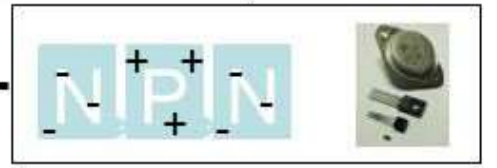
low-level language



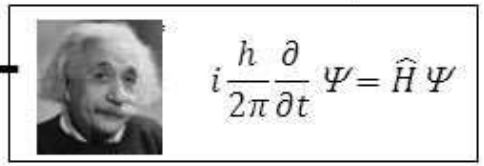
bytes and bits



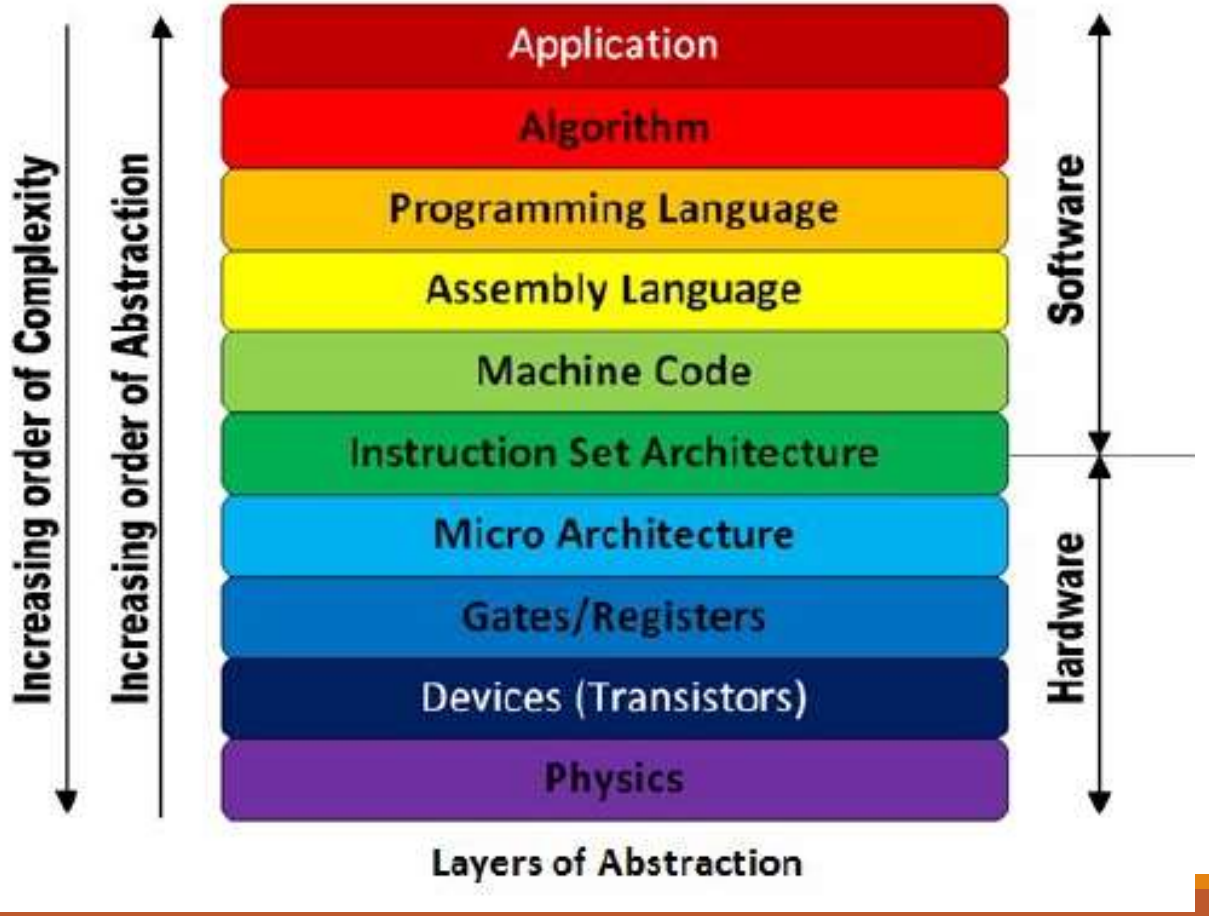
voltage and current



physics

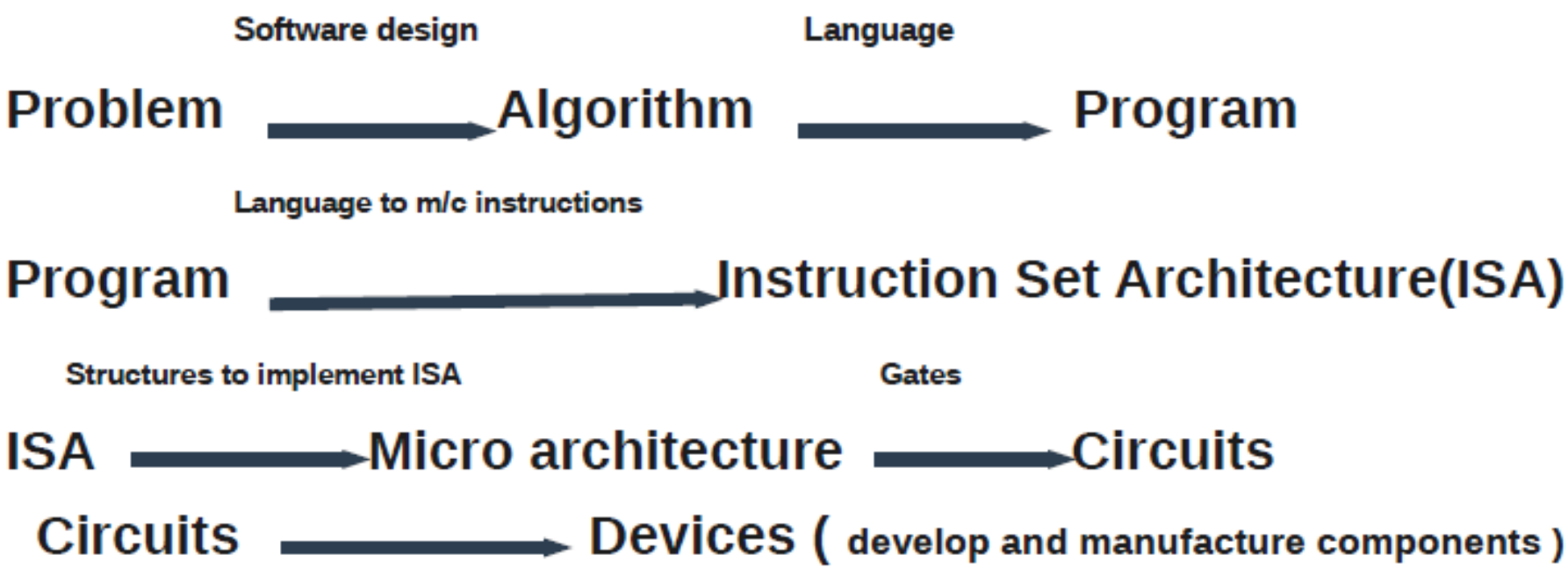


Layers of Abstraction

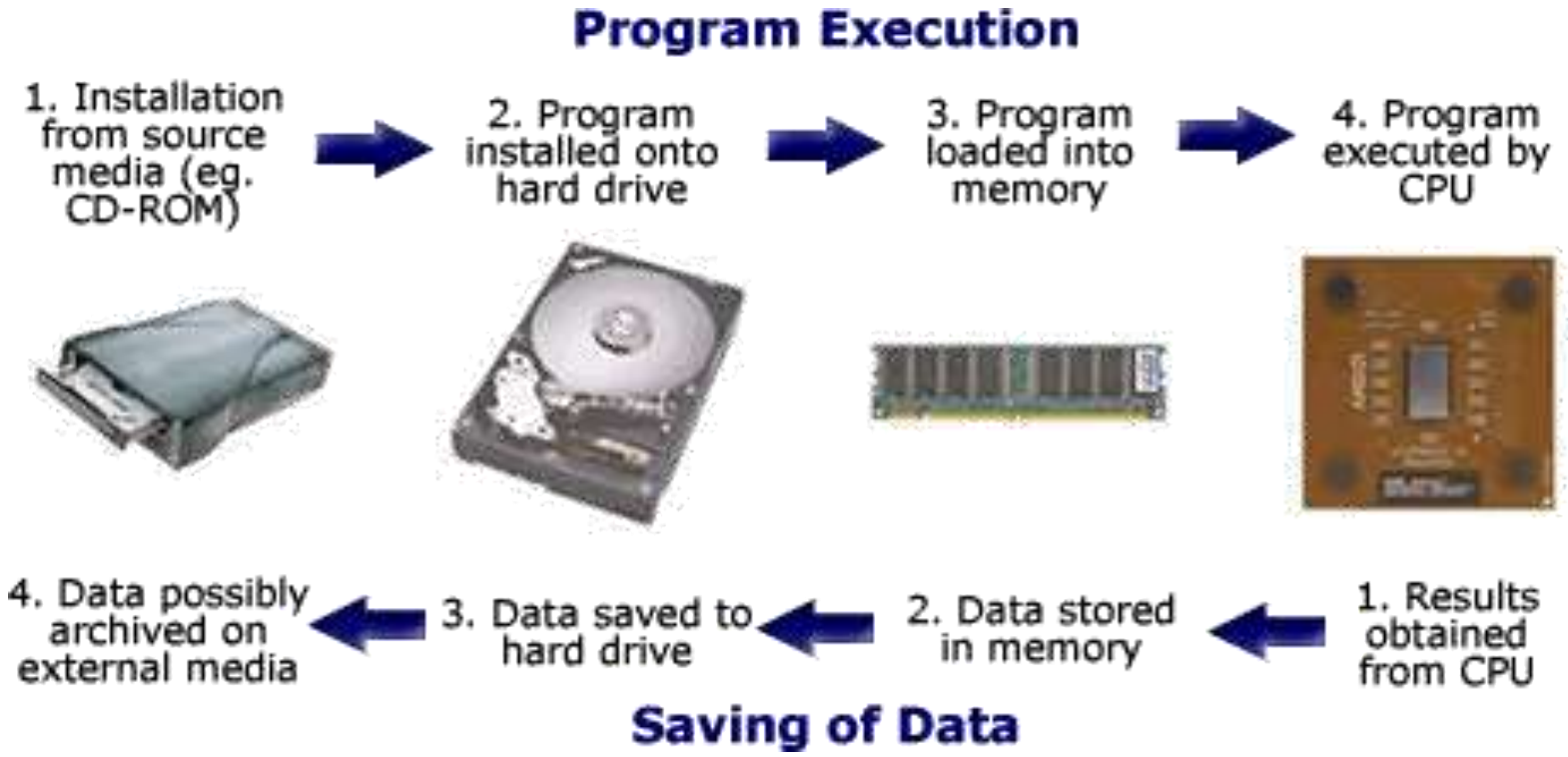


How Do We Solve a Problem by a Computer ?

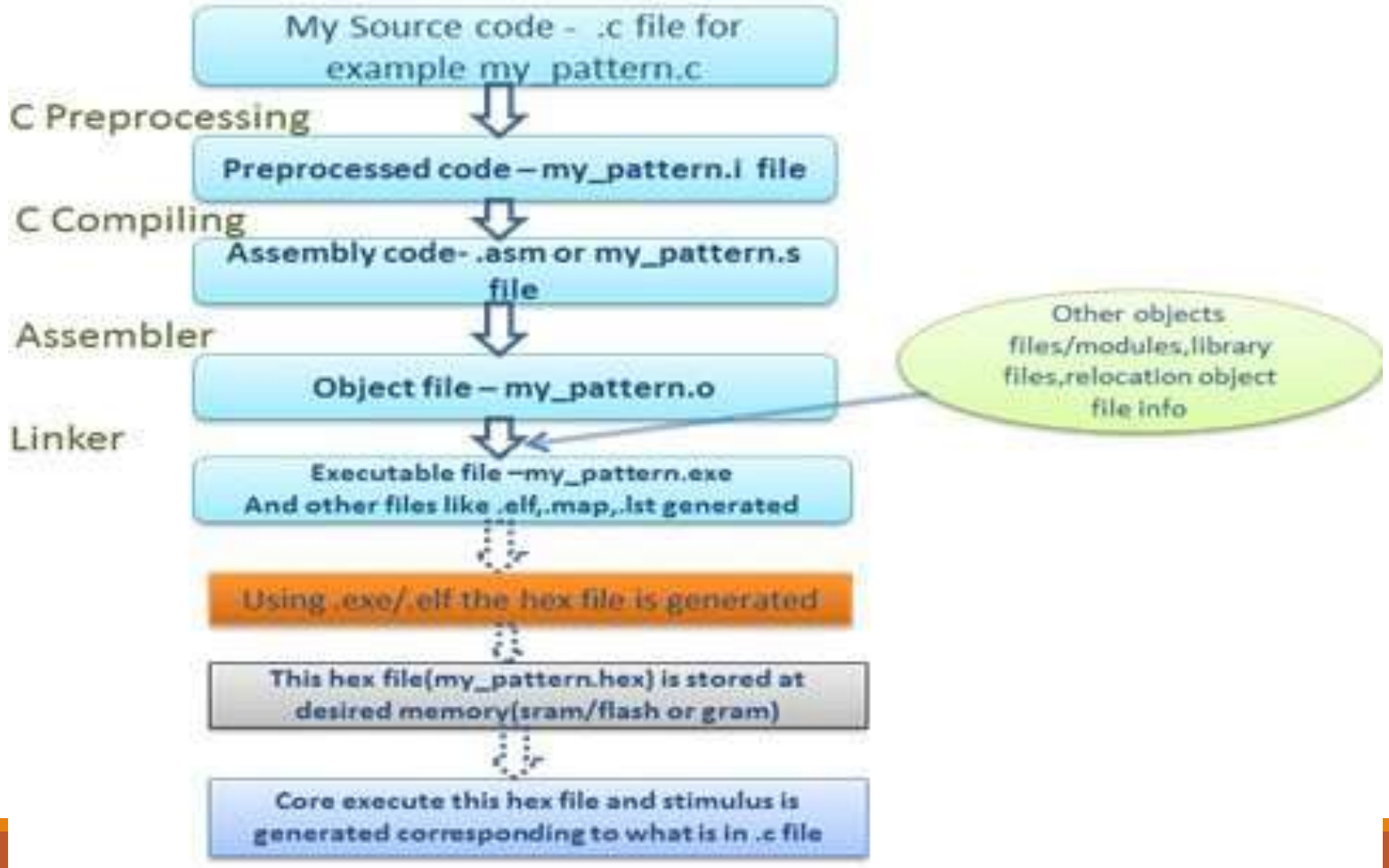
Ans : Through a systematic sequence of transformation between layers of abstraction



How are programs executed by the computer?



Compilation Steps of C Program



Thank You