UNIVERSITY of INFORMATION TECHNOLOGY

# Basic Concepts

Dr. Nwe Nwe Myint Thein
Associate Professor

# Outcomes

- To know the basic concept of computer system
- To understand the computer as a layered system
- To recognize the function of the computer components

# Language, Level and Virtual Machine

UNIVERSITY of INFORMATION TECHNOLOGY

- **Programming Language:**
  - Each computer has a native machine language (language L0) that runs directly on its hardware
  - A more human-friendly language is usually constructed above machine language, called Language L1
- **Programs written in L1 can run two different ways:**

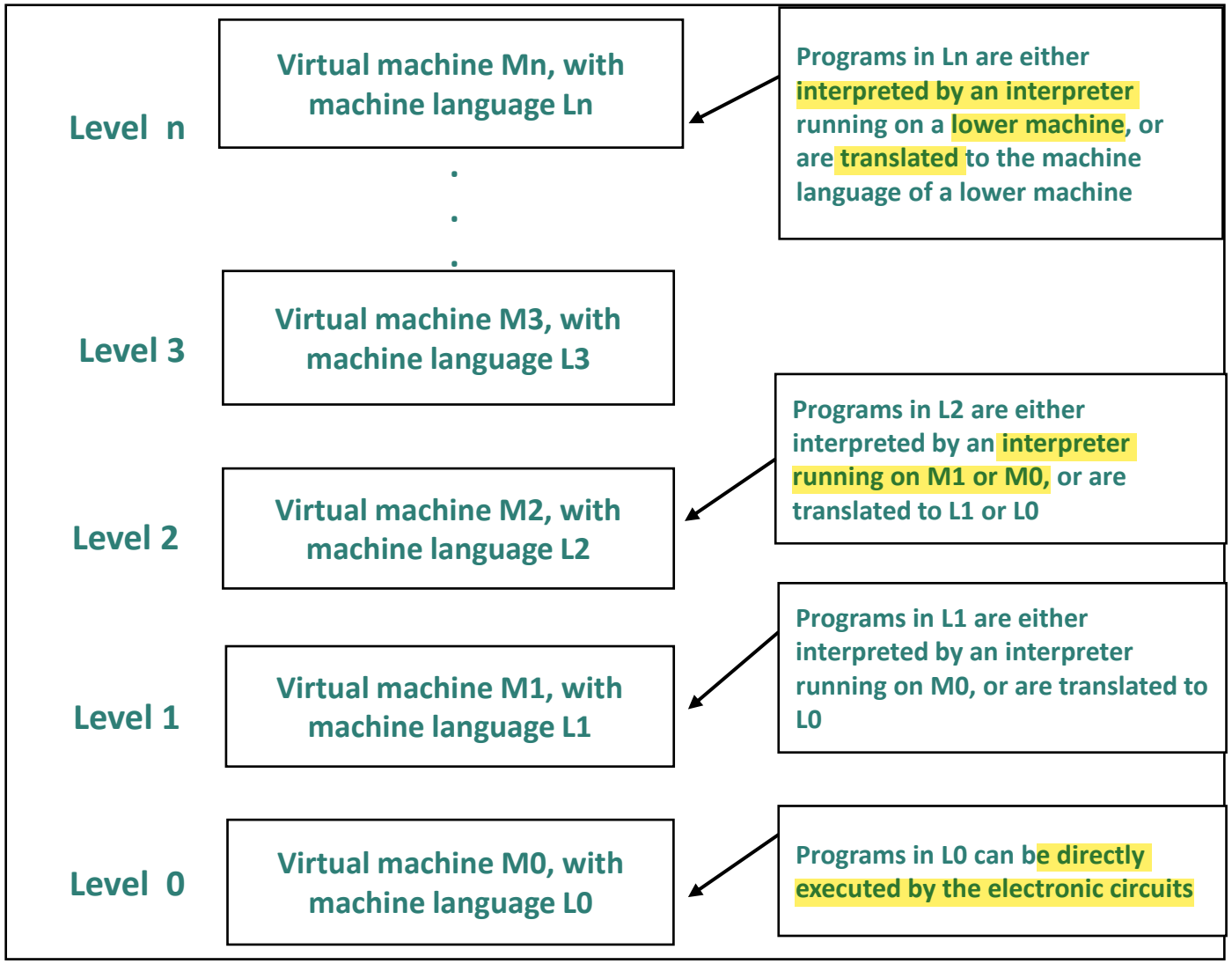
native machine language

  - Interpretation – L0 program interprets and executes L1 instructions one by one
  - Translation – L1 program is completely translated into an L0 program, which then runs on the computer hardware

human friendly language => native machine language

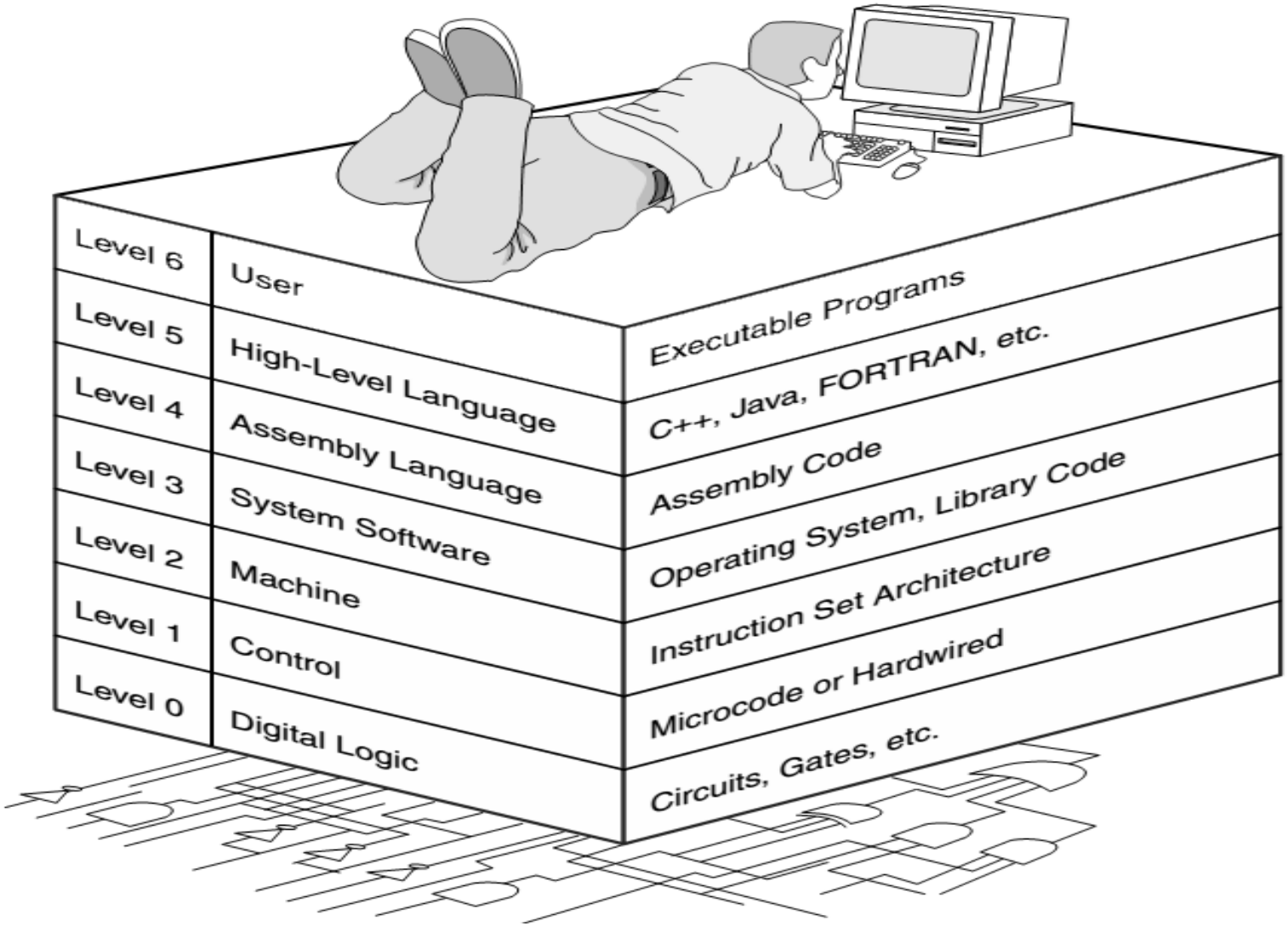
L1 = human friendly language
L0 = native machine language - run directly on hardware

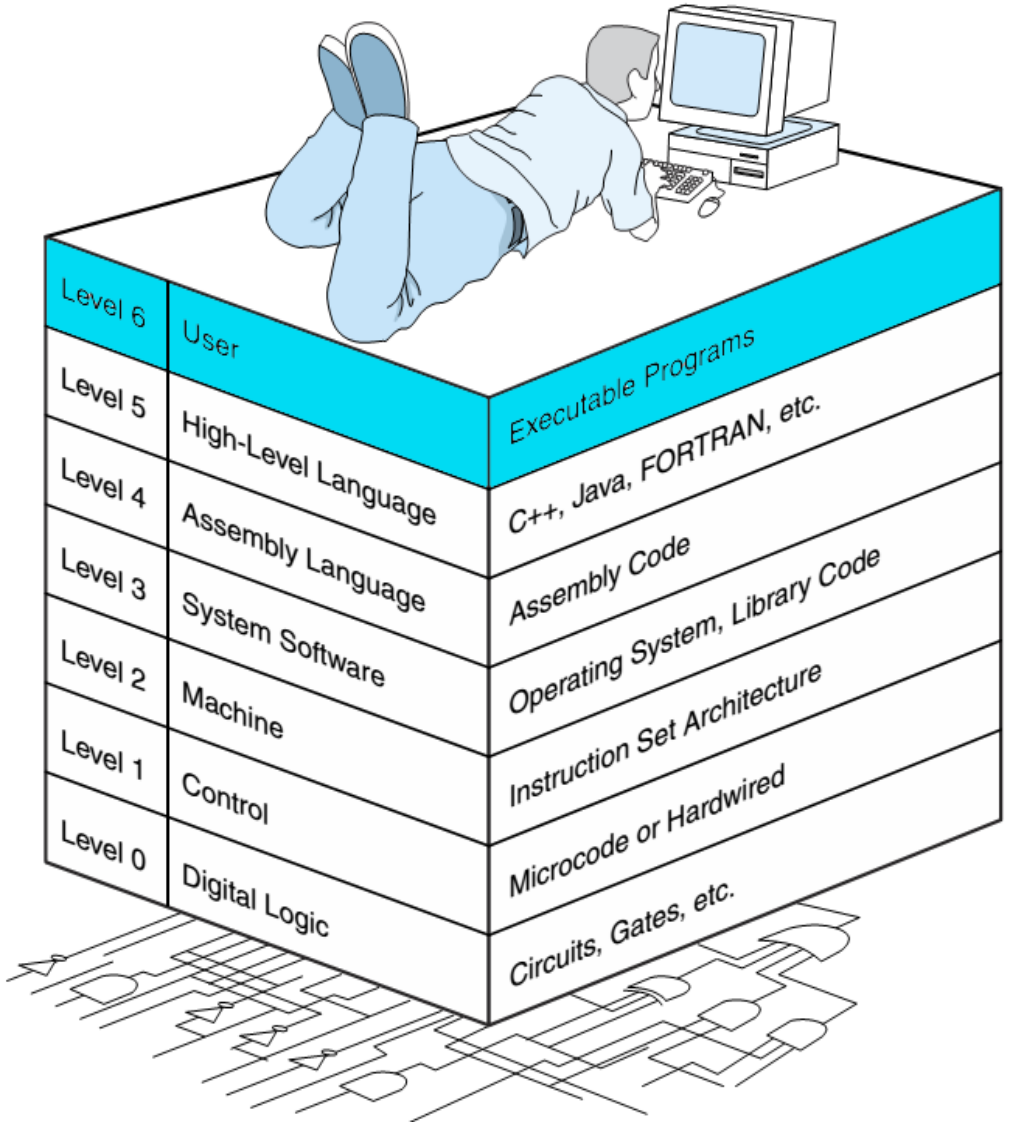
# Multilevel Machine




- **Virtual machine layer**
  - An abstraction of the level below it.
- Execute their own particular instructions
- Calling upon machines at lower levels to perform tasks as required
- Computer circuits ultimately carry out the work

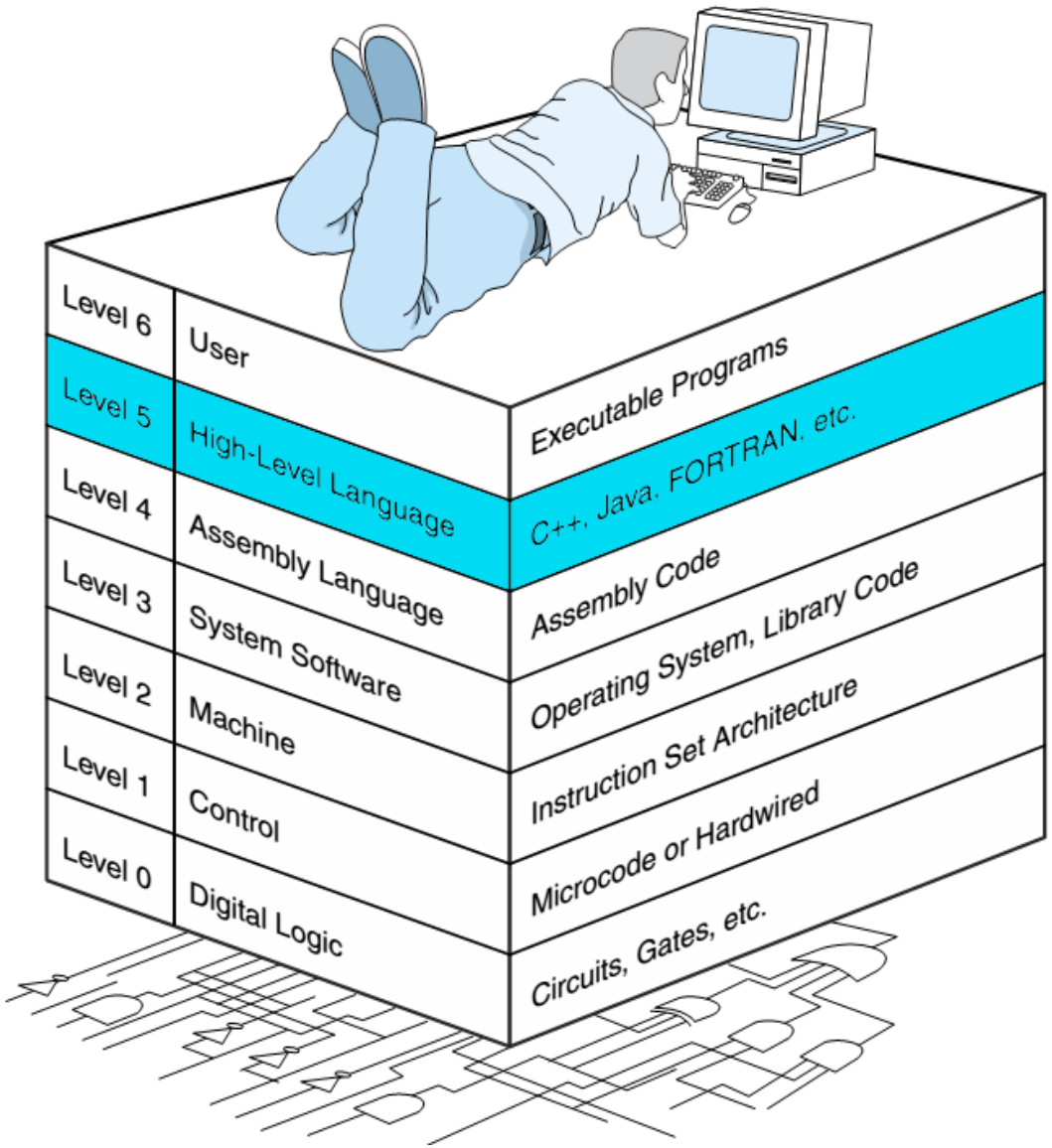
# The Computer Level Hierarchy





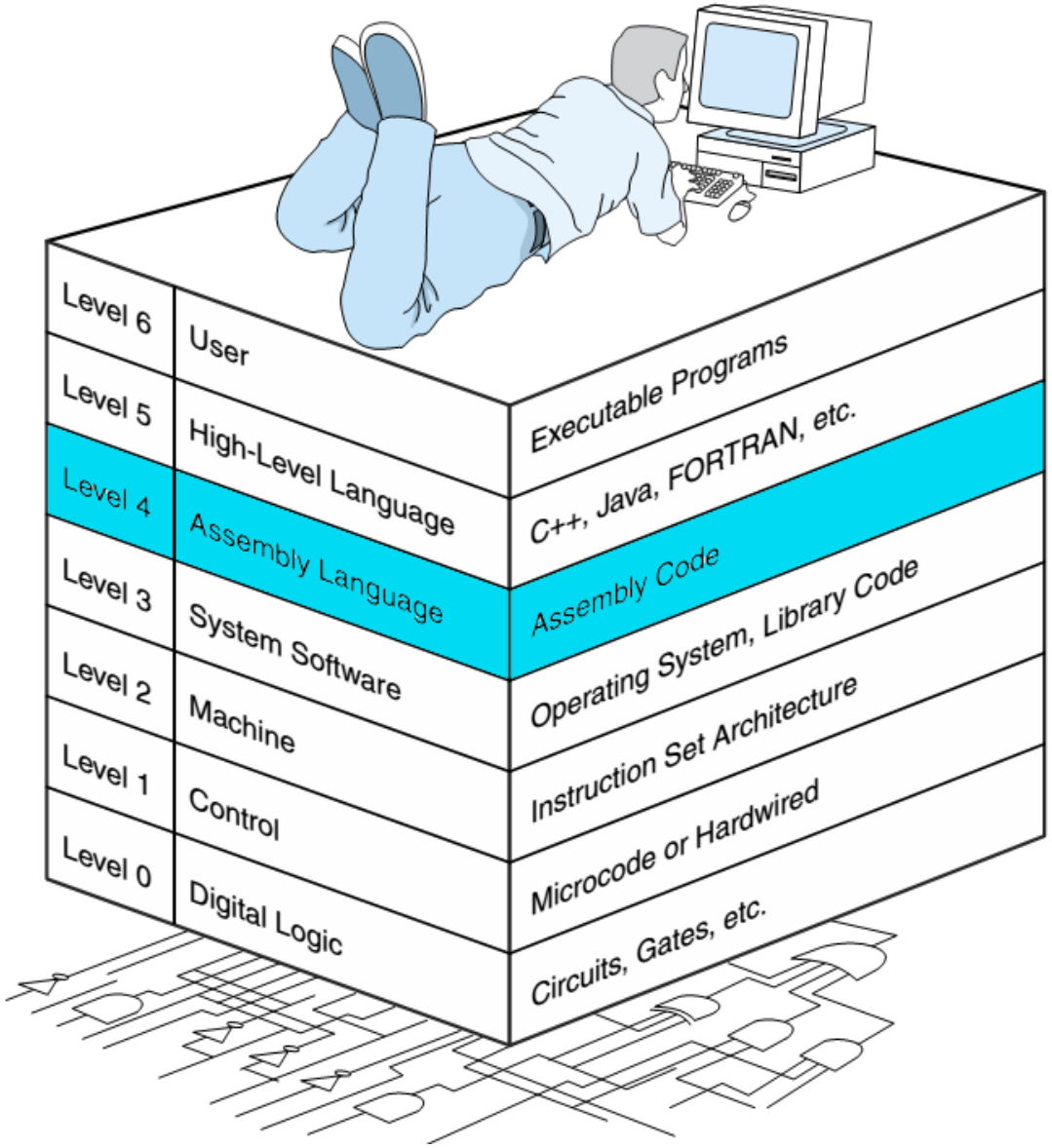

- Composed of applications (E.g. word processors, graphics packages, or games)
- Program execution and user interface level
- Most familiar level




- Consists of languages (e.g. C, C++, Fortran, Lisp, Pascal, and Prolog)
- Must be translated to a language the machine can understand.
- Use compiler or interpreter for translation process.



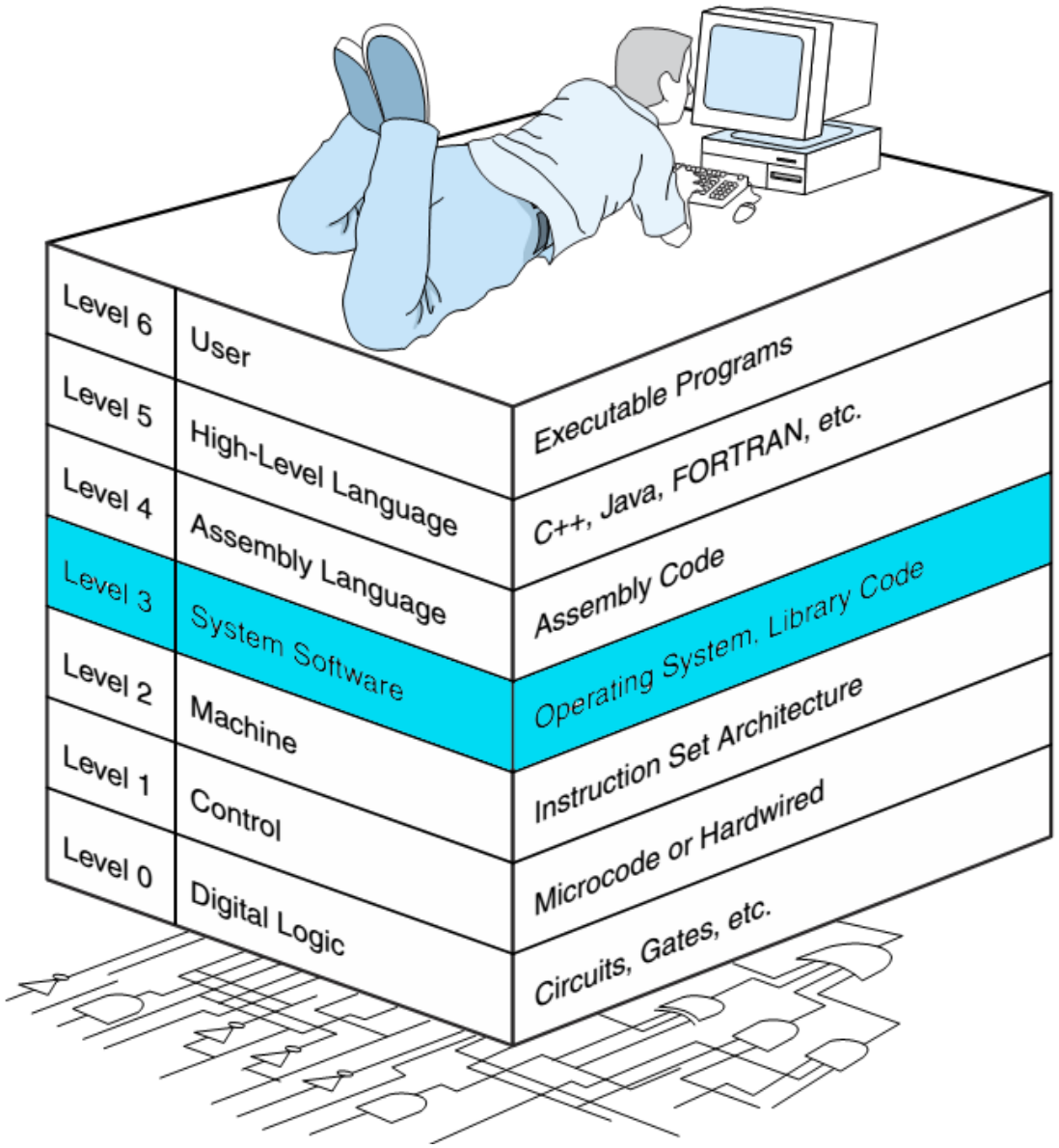

- A symbolic form for one of the underlying languages.
- Are first translated to level 1, 2, or 3 language.
- Require an assembler for the translation process.

# Difference between Compiler, Interpreter and Assembler

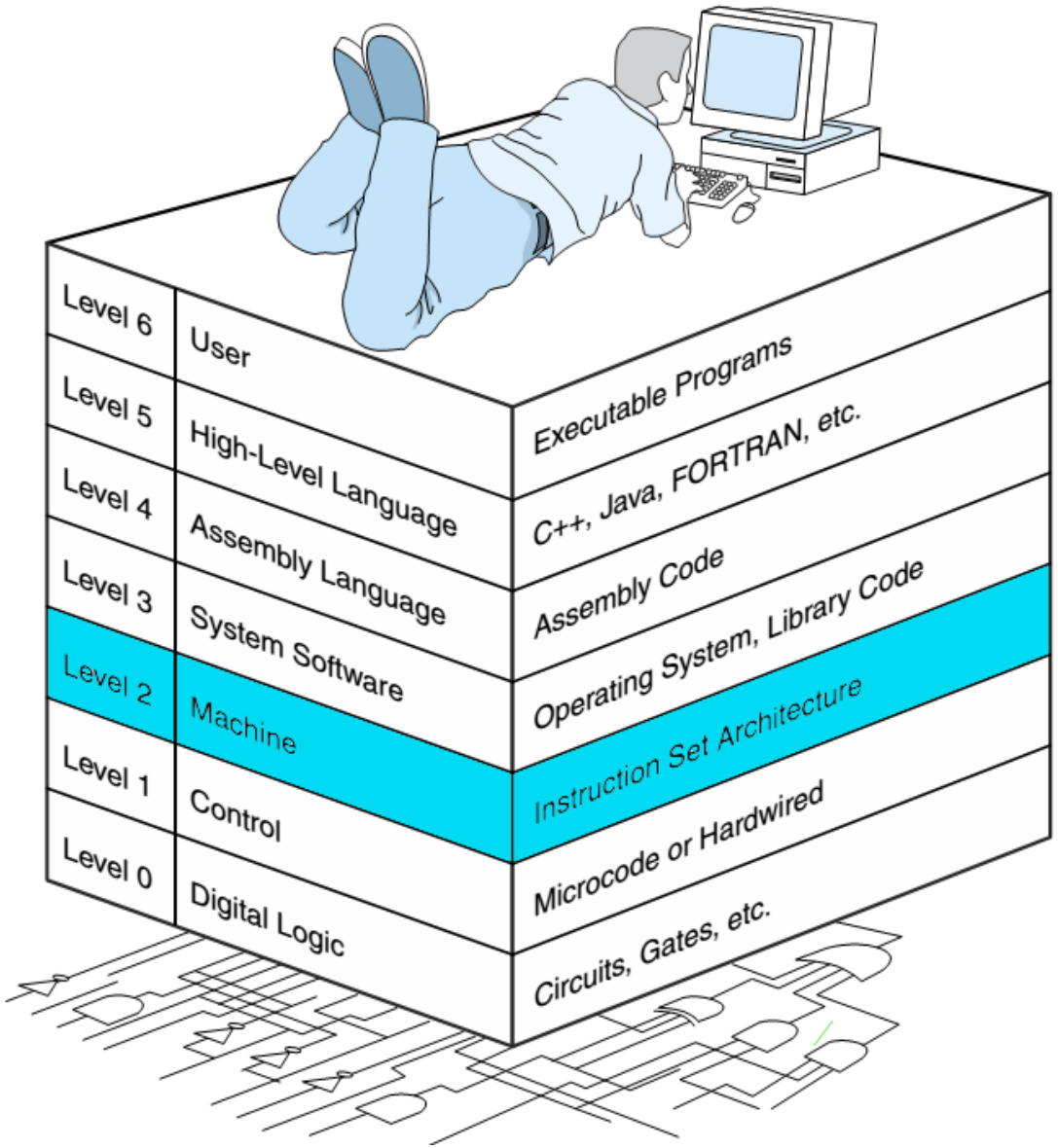
UNIVERSITY of INFORMATION TECHNOLOGY

| Compiler | Interpreter | Assembler |
|---|---|---|
| ➢ Software that converts programs written in a high-level language into machine language | ➢ Software that translates a high-level language program into machine language | ➢ Software that converts programs written in assembly language into machine language |
| ➢ Converts the whole high-level language program to machine language at a time | ➢ Converts the high-level language program to machine language line by line | ➢ Converts assembly language program to machine language |
| ➢ Used by C, C++ | ➢ Used by Ruby, Perl, Python, PHP | ➢ Used by assembly language |




- Controls executing processes on the system
- Protects system resources
- Often, Assembly language instructions often pass through Level 3 without modification.

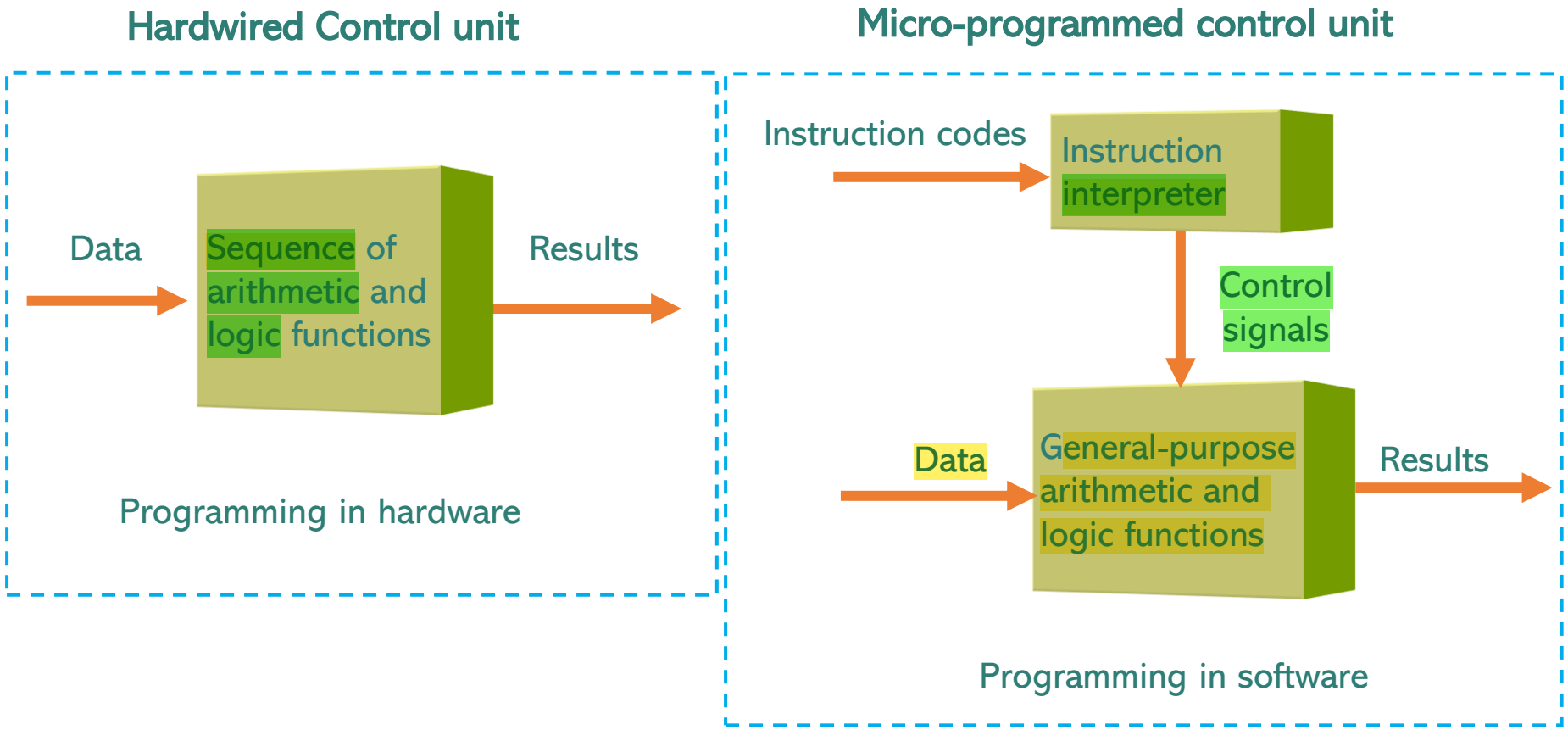



- Computer manufacturer publishes a manual for each of the computers it sells
  - "Machine Language Reference Manual"
- Describing the instructions carried out interpretively by the **microprogram** or **hardware execution circuits**
- Consists of instructions that are particular to the architecture of the machine.
- Can be executed directly by the electronic circuits.
- **No need** any interpreters, translators, or compilers.

# Types of Control Unit

- To execute an instruction, there are **two types of control units**



# Types of Control Unit (cont'd)

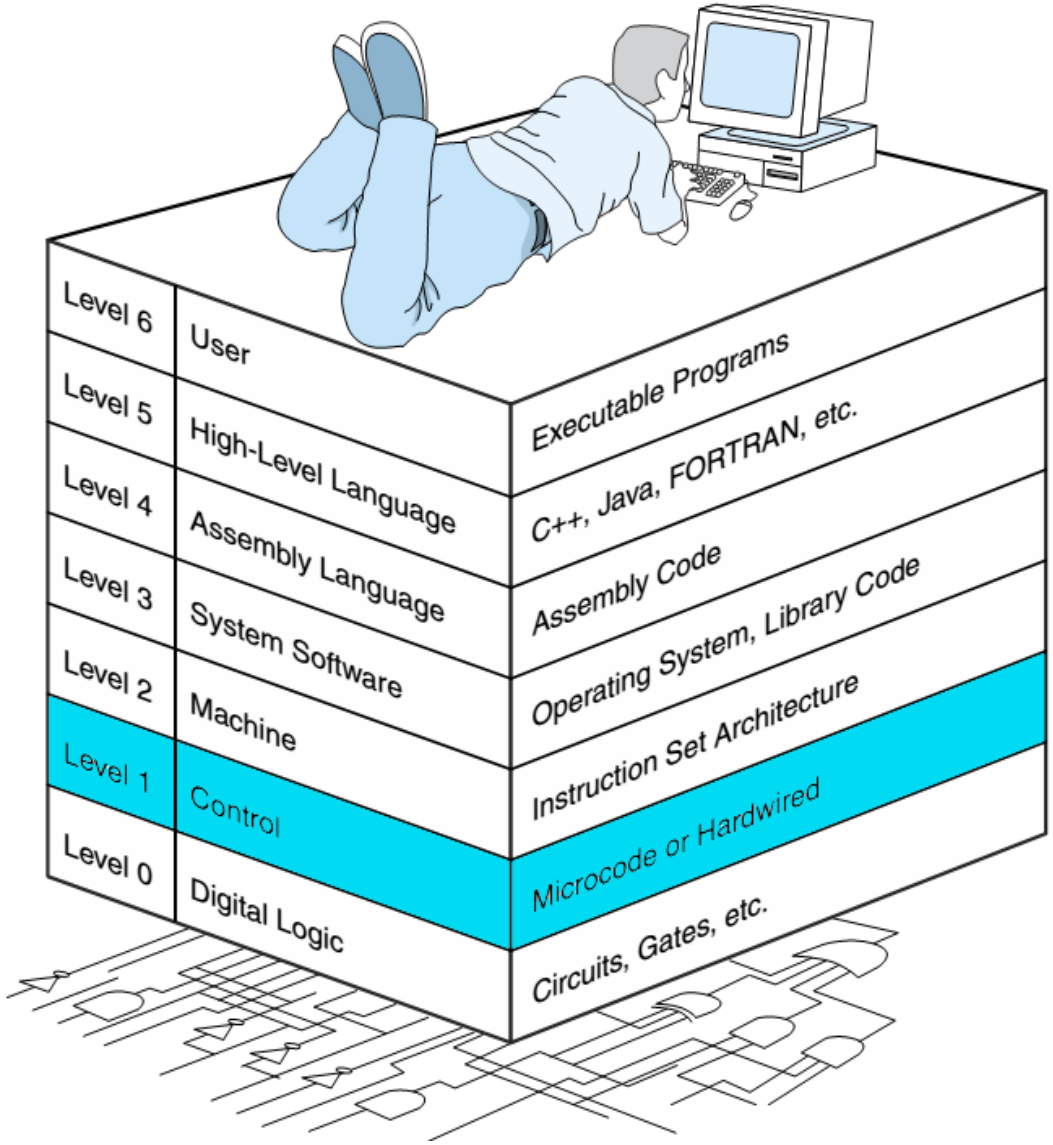
UNIVERSITY of INFORMATION TECHNOLOGY

- To execute an instruction…….

- A microprogram
  - A program written in a low-level language that is implemented by the hardware.
  - The slower instruction execution.
  - The additional layer of translation is required.

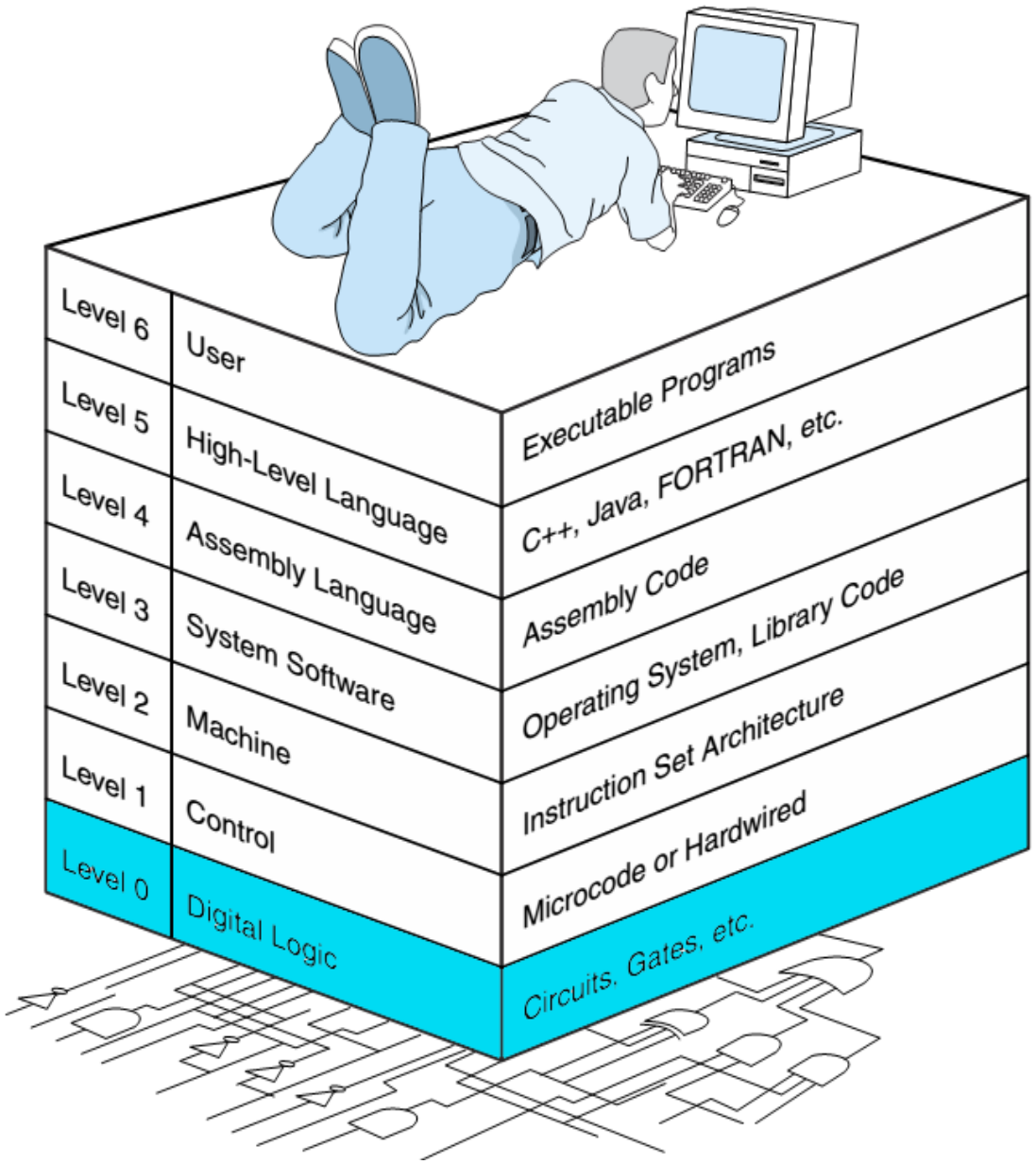
- Hardwired
  - The control units consist of hardware .
  - It directly executes machine instructions.
  - The faster instruction execution.

| Hardwired Control Unit | Microprogrammed Control Unit |
| --- | --- |
| Generates the control signals needed for the processor **using logic circuits.** | Generates the control signals with the help of **micro instructions stored in control memory.** |
| It is **faster.** | It is **slower.** |
| **Difficult to modify** as the control signals that need to be generated are hard wired. | **Easy to modify** as the modification need to be done only at the instruction level. |
| **Costlier** as everything must be realized in terms of logic gates. | **Less costly** as only micro instructions are used for generating control signals. |
| It **cannot handle complex instructions** as the circuit design for it becomes complex. | It can **handle complex instructions.** |
| **Only limited number of instructions** are used due to the hardware implementation. | Control signals for **many instructions** can be generated . |
| Used in computer that makes use of Reduced Instruction Set Computers (**RISC**). | Used in computer that makes use of Complex Instruction Set Computers (**CISC**). |




- Decodes and executes instructions and moves data through the system.
- A collection of (typically) 8 to 32 registers that form a local memory and a circuit (ALU), which is capable of performing simple arithmetic operations.
- The registers are connected to the ALU to form a data path, over which the data flow.
- The operation of the data path is controlled by a control unit.

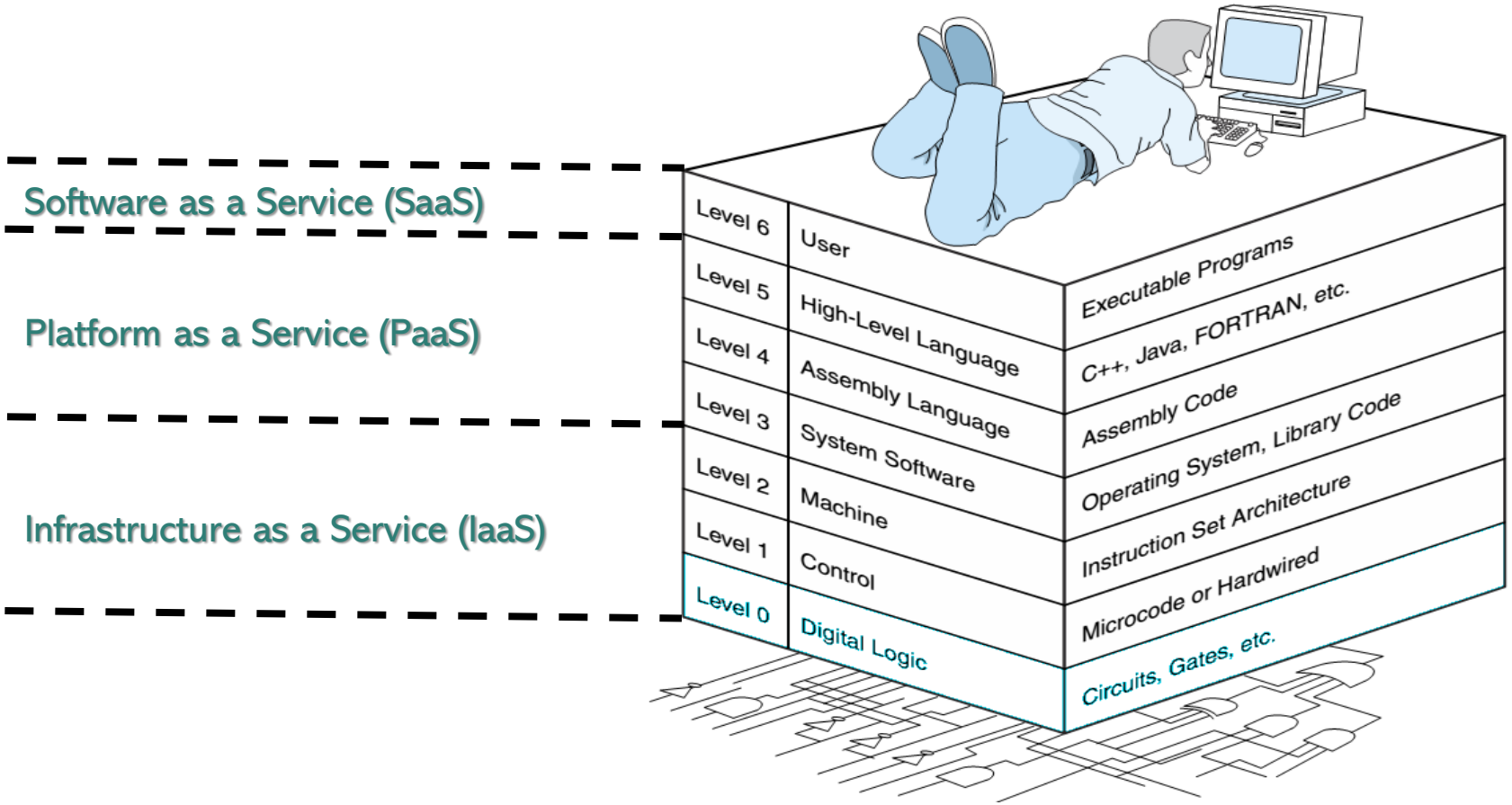



- Digital circuits consist of gates and wires.
- Implement the mathematical and control logic of all other levels.

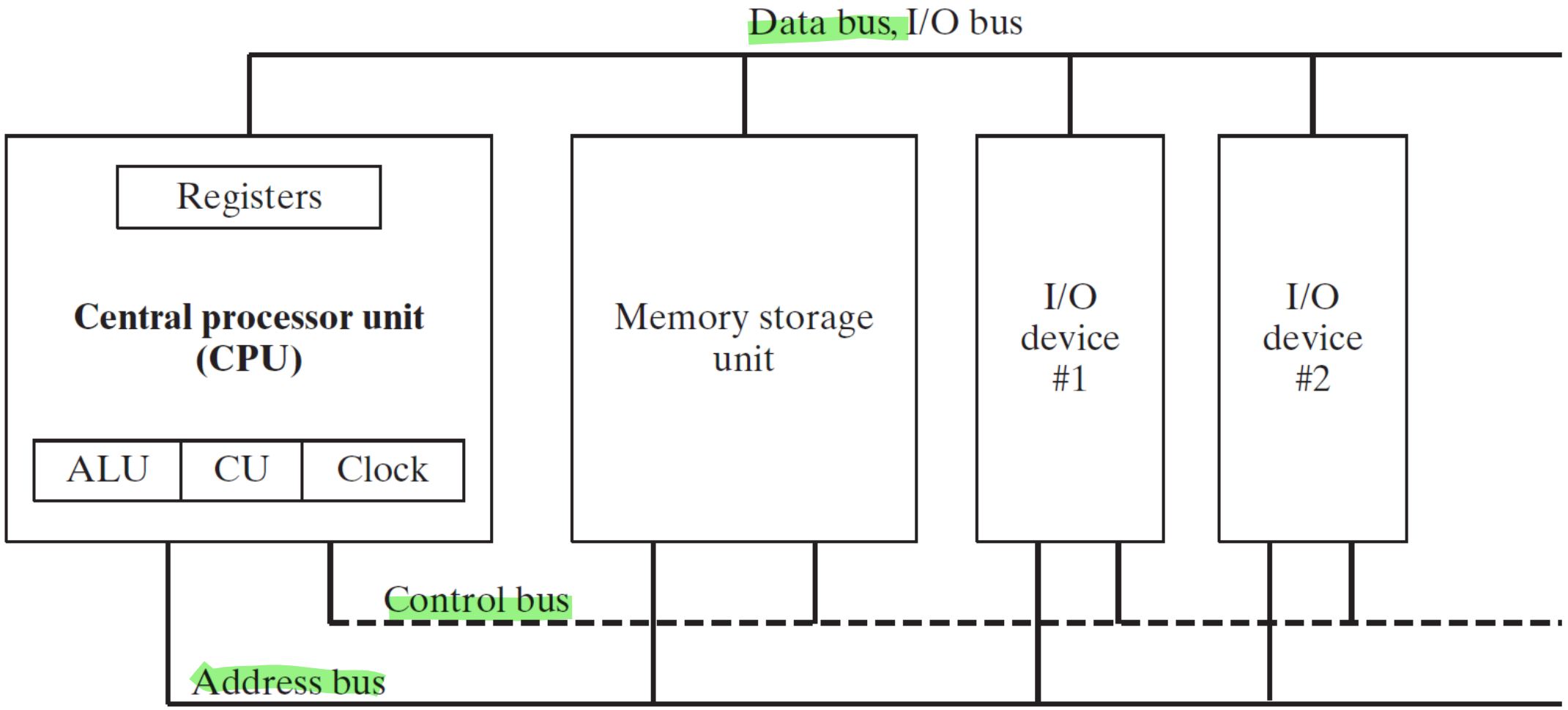
# Computing Services

- Cloud computing
  - The on-demand availability of computer system resources, especially data storage (cloud storage) and computing power, without direct active management by the user.



# Block Diagram of a Microcomputer



# Central Processing Unit

UNIVERSITY of INFORMATION TECHNOLOGY

## Control Unit

- Controls the operation of the computer/ CPU (sequence of execution steps)
- Performs its data processing functions
- It works through a cycle of fetch, decode, and execution.

## Arithmetic Logic Unit (ALU)

- Performs the computer's data processing function → arithmetic and bitwise processing

## Registers

- Provide storage internal to the CPU
- Registers are small but very fast memory to hold data before and after being processed by the CPU.
- All ALU operations are accomplished within registers

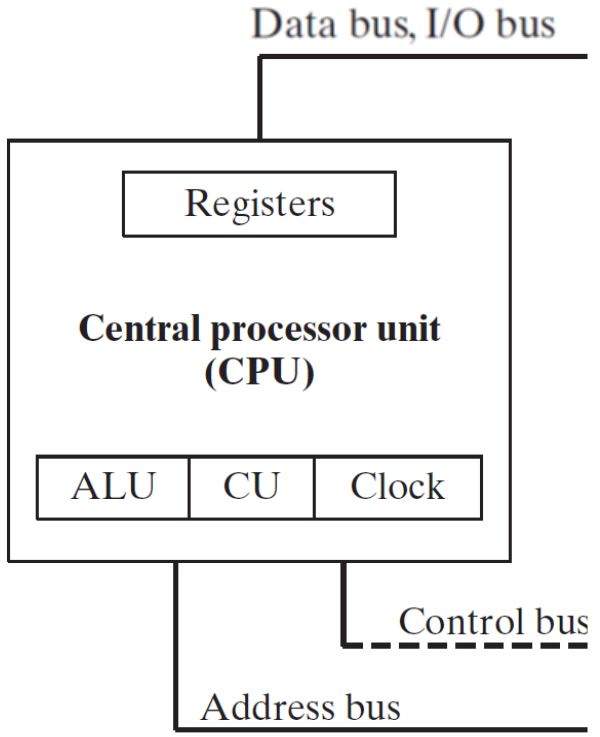
## CPU Interconnection (Internal Bus)

- Some mechanism that provides for communication among the control unit, ALU, and registers

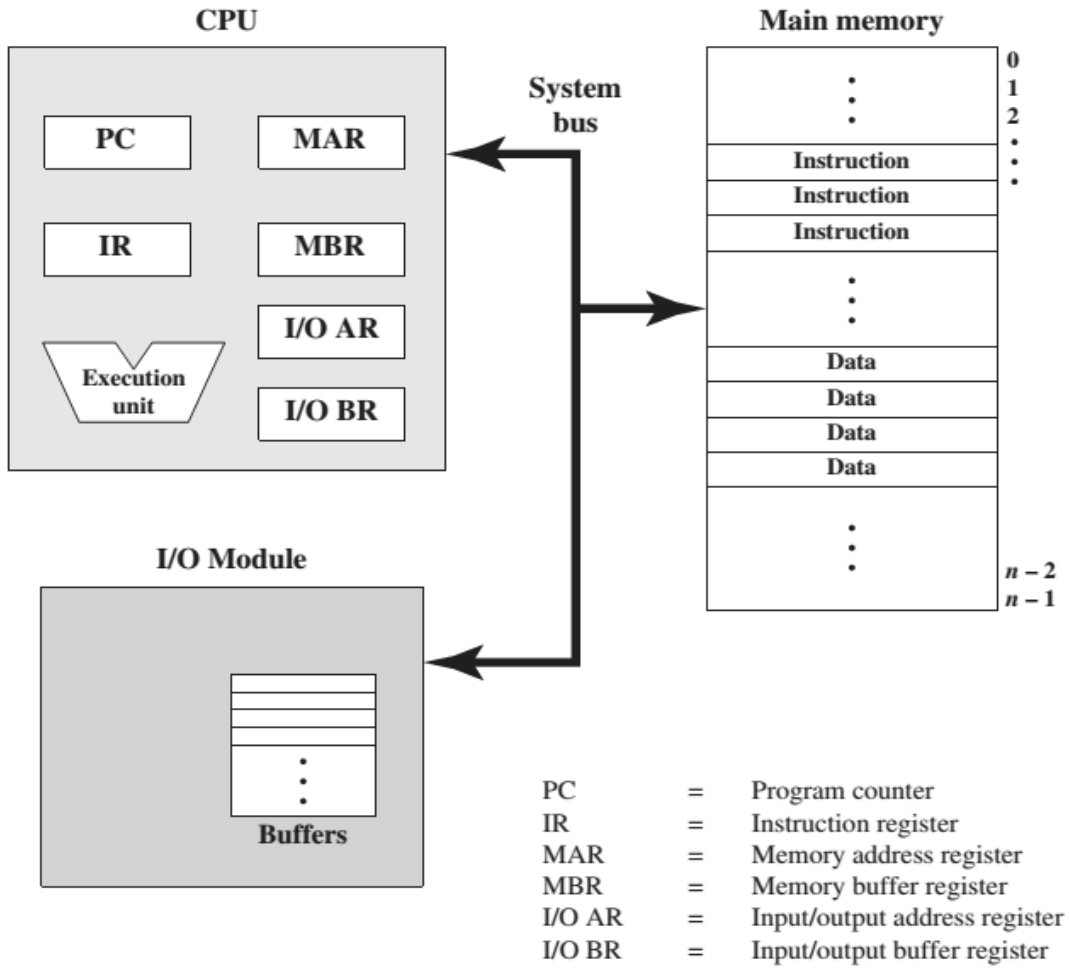
## Clock

same time

- synchronizes all CPU and BUS operations
- machine (clock) cycle measures time of a single operation
- clock is used to trigger event



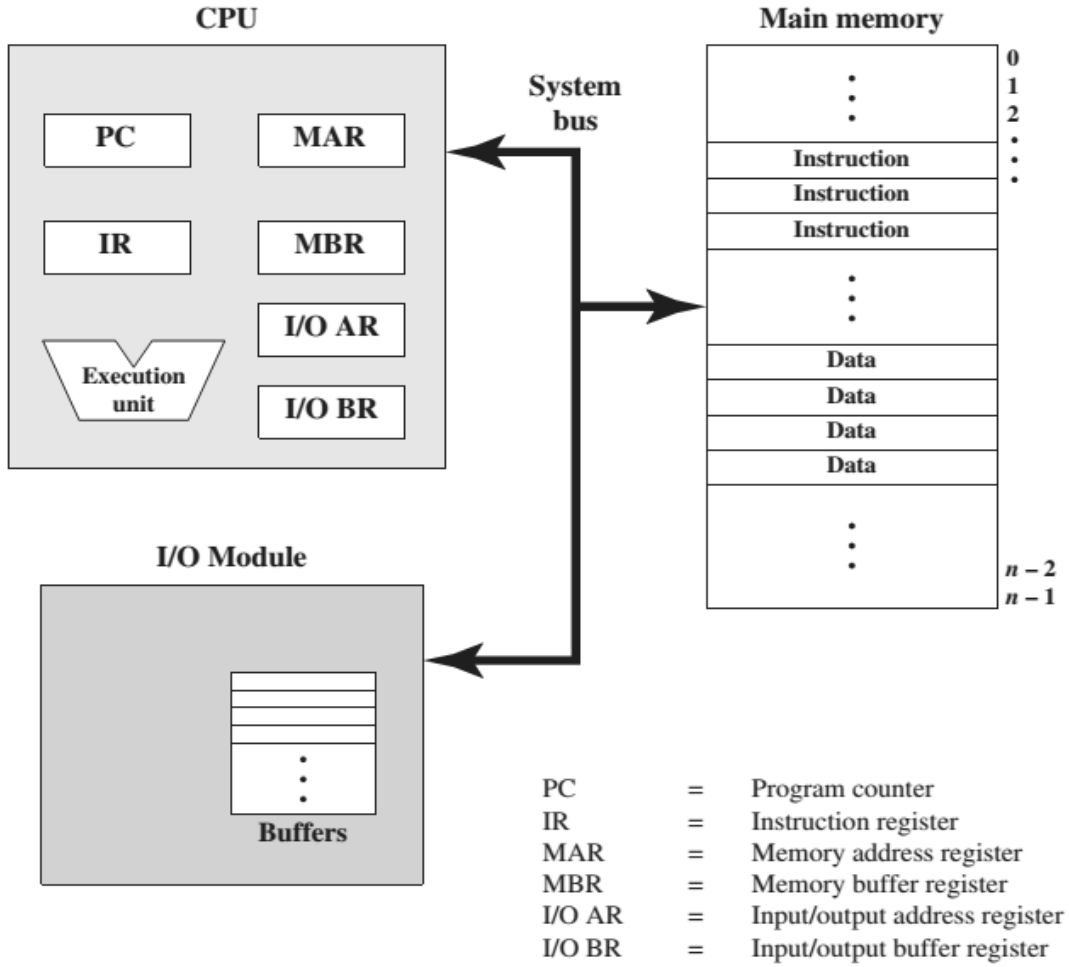
# Computer Components Top-Level View




- **Main Memory**
  - consists of a set of locations, defined by sequentially numbered addresses
  - each location contains a binary number that can be interpreted as either an instruction or data.

- **I/O Module**
  - moves data between the computer and its external environment
  - transfers data from external devices to CPU and memory, and vice versa
  - contains internal buffers for temporarily holding these data until they can be sent on.

# Computer Components Top-Level View

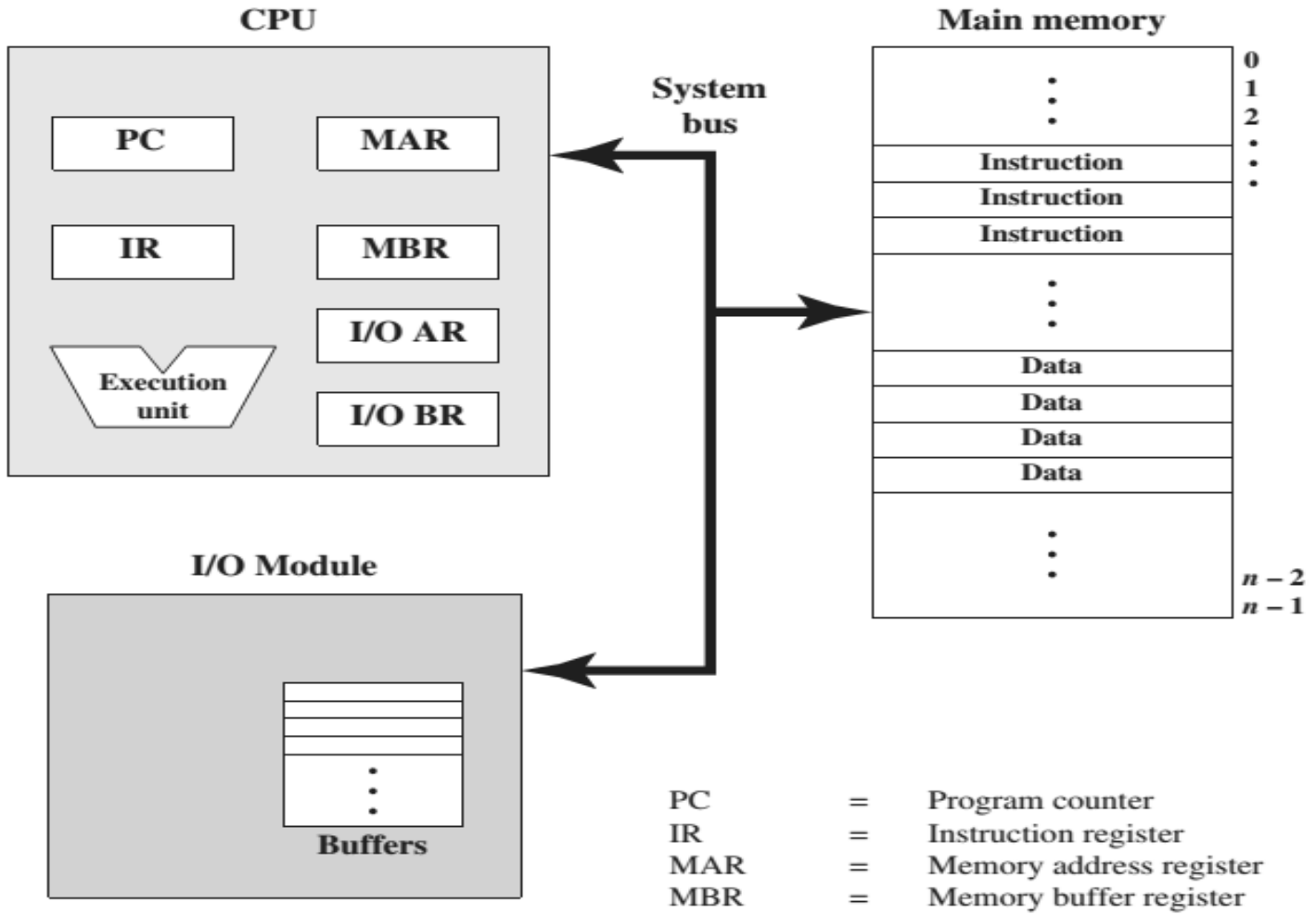



- **System Bus**
  - some mechanism that provides for communication among CPU, main memory, and I/O

- **Program Counter**
  - Contains the **address of the next instruction** pair to be fetched from memory.

- **Instruction register**
  - Once an **instruction** is fetched from main memory, it is **stored in the IR**.

# Registers




PC = Program counter
IR = Instruction register
MAR = Memory address register
MBR = Memory buffer register
I/O AR = Input/output address register
I/O BR = Input/output buffer register

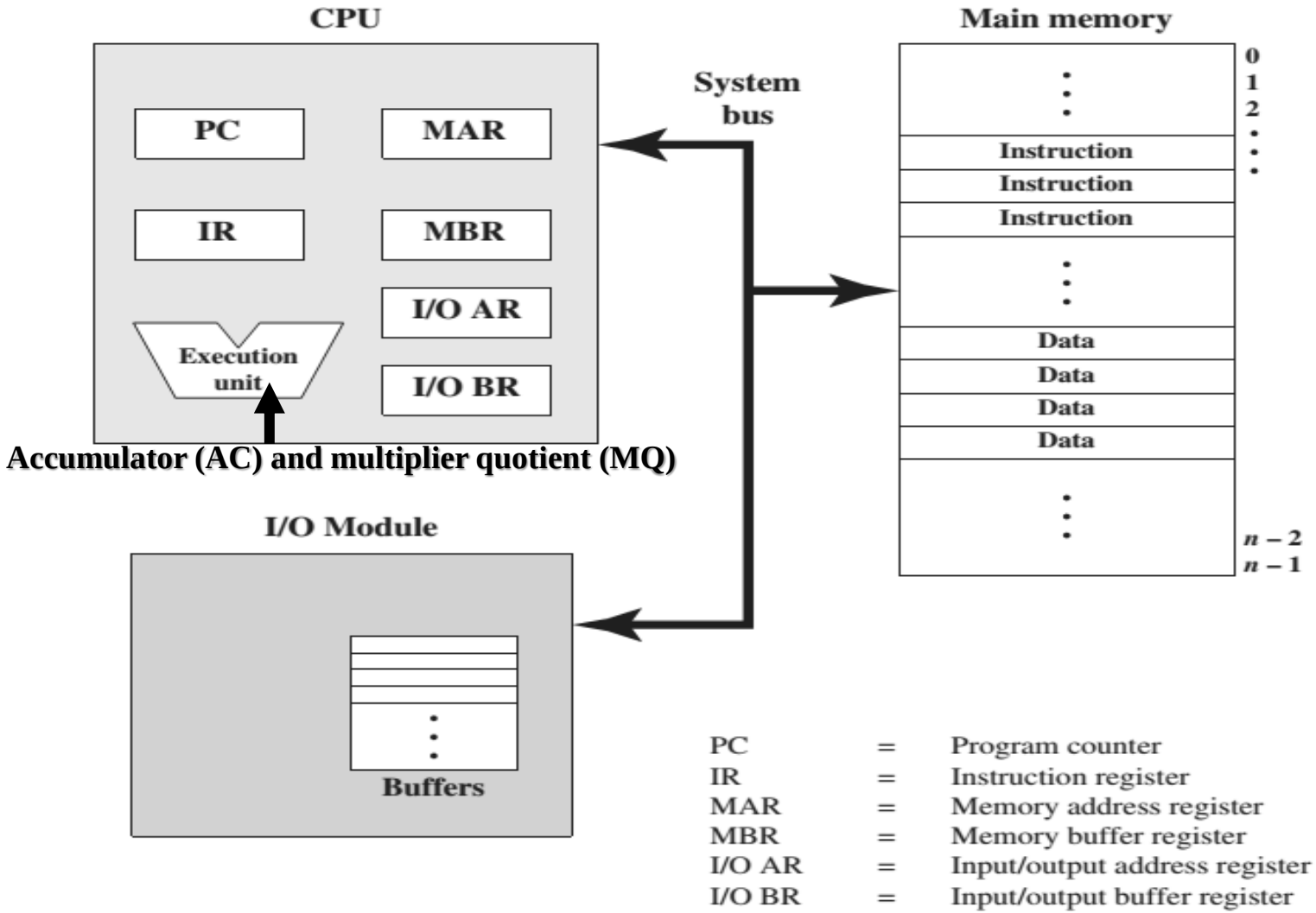
**Memory address register:** which specifies the **address in memory** for the next read or write.

**Memory buffer register:** which contains the data to be written into memory or which receives the data read from memory.

**Input/output address register:** which specifies a **particular I/O device**.

**Input/output buffer register:** It is used for the **exchange of data** between an I/O module and the CPU.

# Registers




**Execution unit:** also called a **functional unit**, is a part of the central processing unit (CPU) that **performs the operations and calculations** as instructed by the computer program.

**Accumulator (AC) and multiplier quotient (MQ):** Employed to temporarily hold operands and results of ALU operations