Computer Organization & Architecture
CS 204

Anil Kumar Sao, EECS, IIT Bhilai
anil@iitbhilai.ac.in

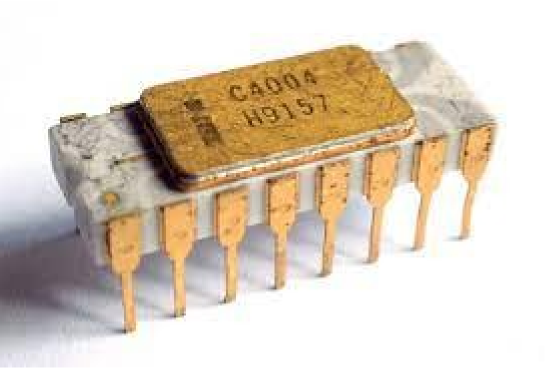
Computer Organization & Architecture
CS 204

Prerequisites: IC150 and IC10

Credits: 4

Class: Tue: 2.00 to 3.30
Thu: 3.30 to 5.00pm

Microprocessors



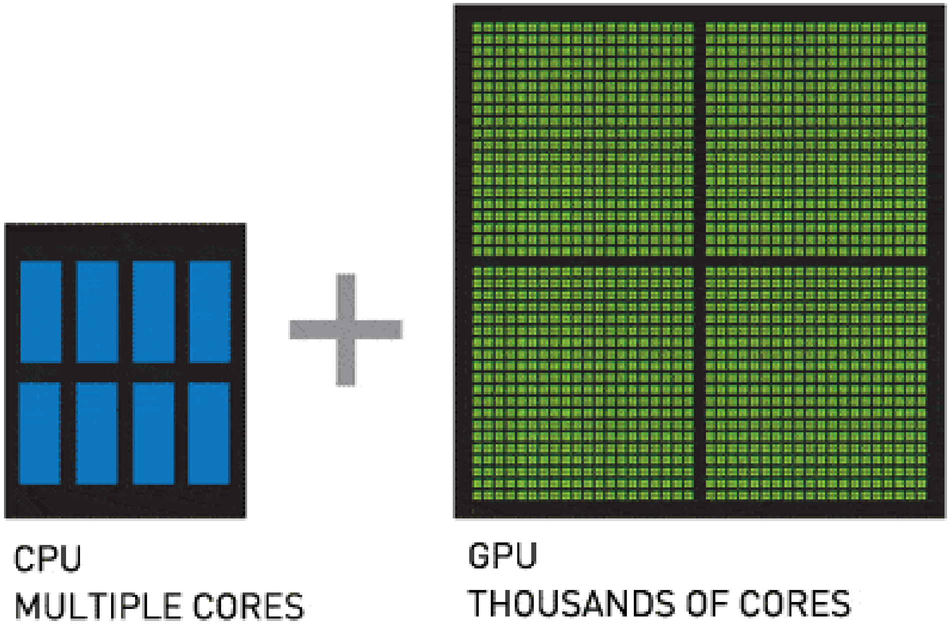
Intel C4004
Produced 1971-81
No cache
Data width: 4
Address width: 12(M)
Clock:730-749 KHz
10 µm technology

Produced 1982-1990
No cache
Data width: 16
Address width: 24
Clock: 6 MHz
1.5 µm technology

Launched 2010
4 M cache
Three levels of cache
Data width: 64
Address width: 64
(Max) Clock: 3.06 Ghz
32 nm technology
2 core

Microprocessor

Launched 2015
64 M cache
Three levels of cache
Data width: 64
Address width: 64
(Max) Clock: 4.13
Ghz
20 nm technology
8 core



Computer ?

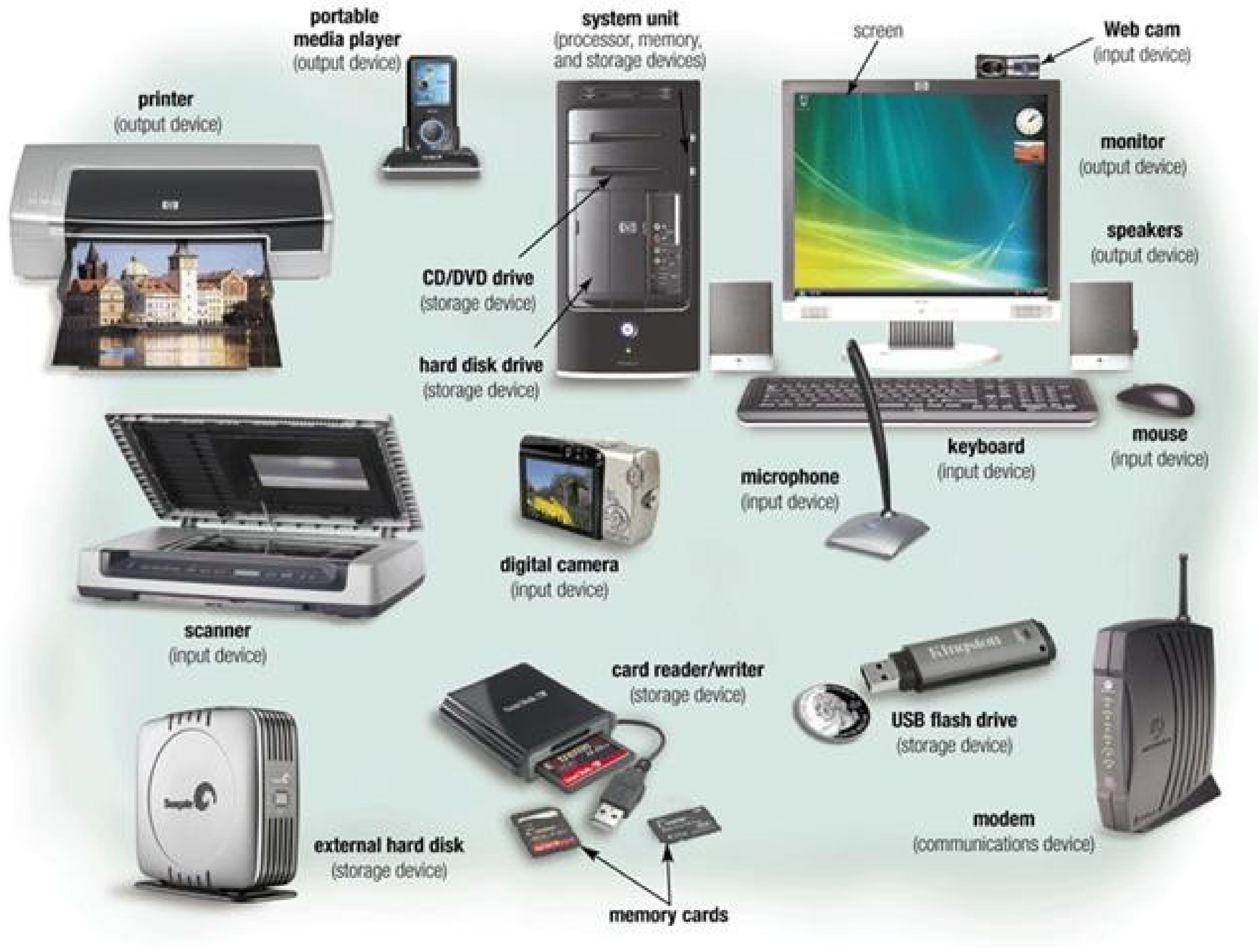
An electronic device which is capable of receiving information (data) in a particular form and of performing a sequence
of operations in accordance with a predetermined but variable set of procedural instructions (program) to produce a result in the form of information or signals.

E.g., Calculator, Mobile, Abacus

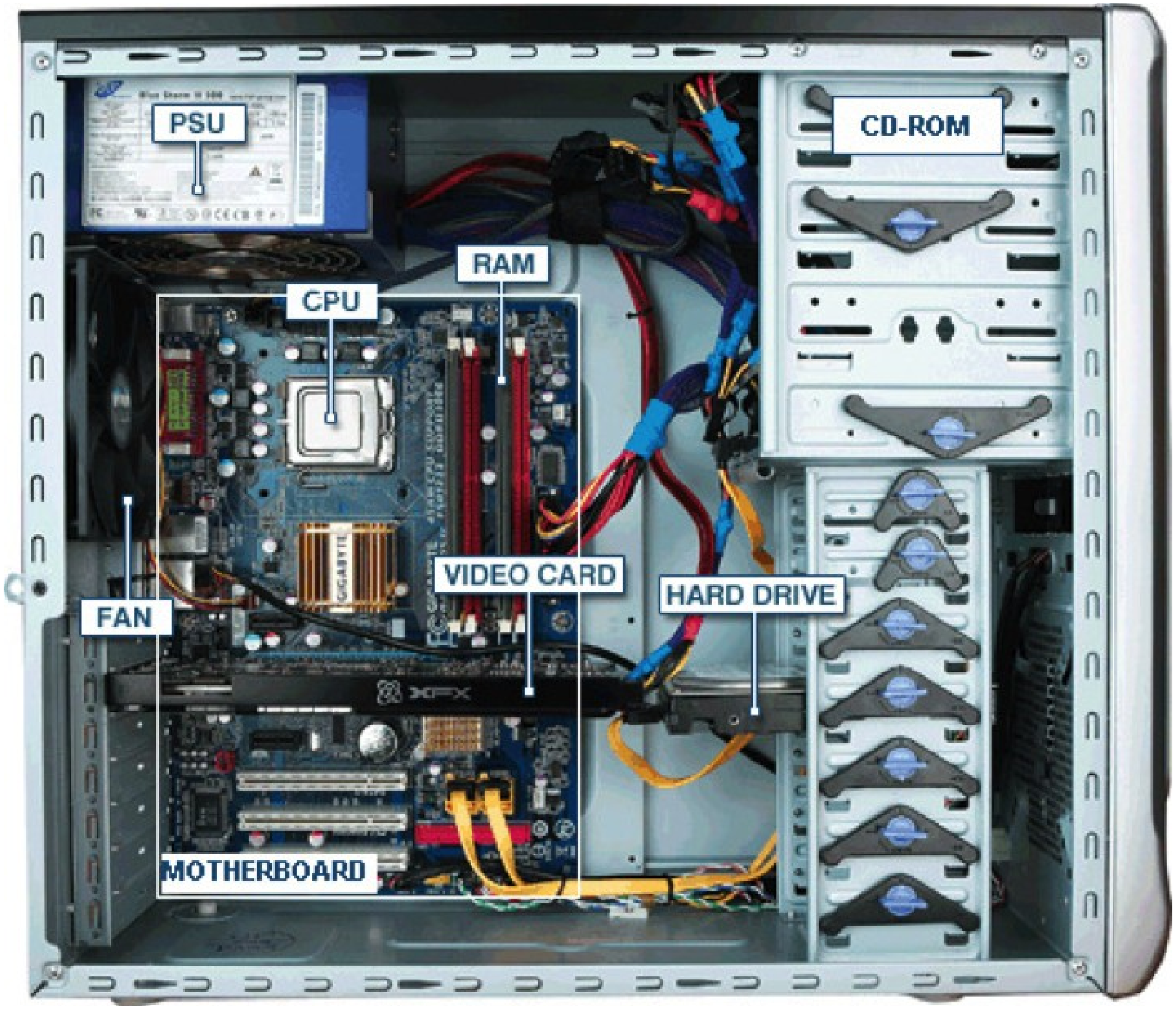
Digital Computer

- Computer is a fast electronic computing/calculating machine that
  - Accepts digitized input information
  - Processes it according to a list of internally stored instructions
  - Produces the resulting output information
- Internal storage is called computer memory
- List of instructions is called a computer program
- Many types of computers exists that differ widely in size, cost, computational power and purpose of use

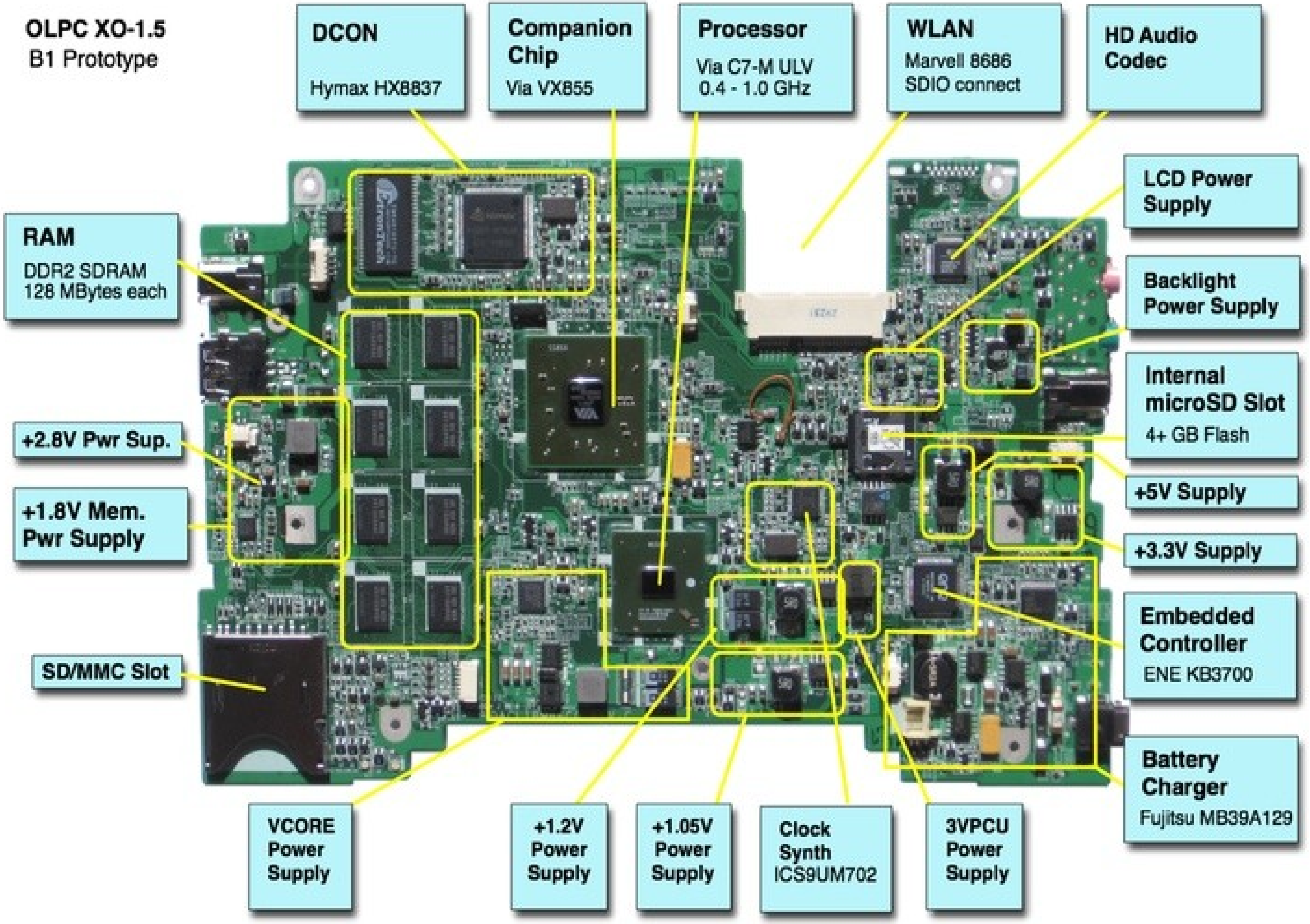
Digital Computer



Digital Computer



Inside view of Laptop



https://i.ytimg.com/vi/KVHu51Fe1e4/maxresdefault.jpg

Basic Operational Concepts

- High-level program segment

```
scanf(“%d, %d”, a,b);
c=a+b;
printf(“&d”, c);
```

Assembly level program segment

IN PORTA, LOCA - Read operand from input port and store into a memory location, LOCA

IN PORTA, LOCB

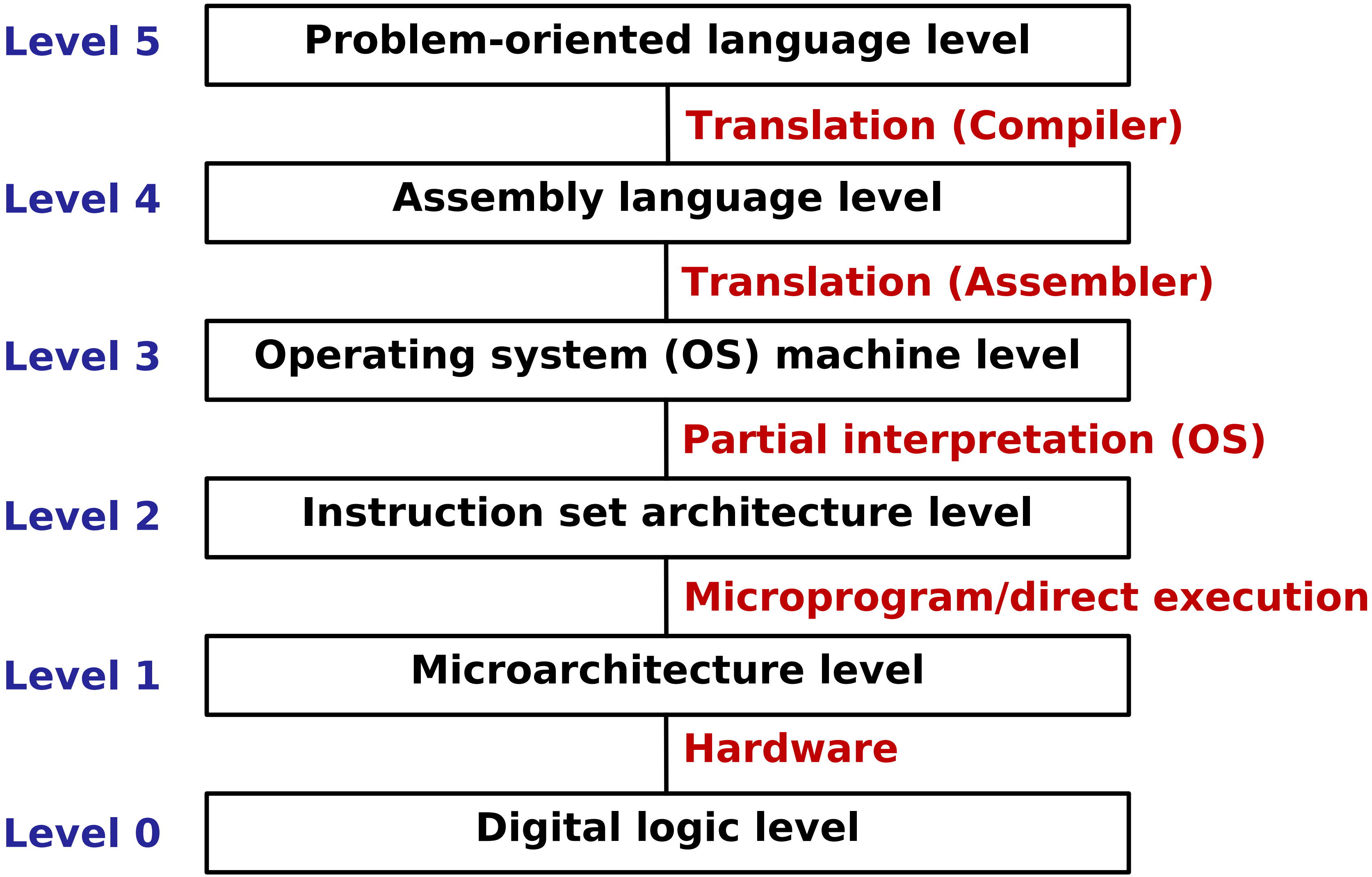
LOAD LOCA, R0 - Load the content from LOCA to processor register R0

ADDM LOCB, R0 - Add the operand at memory location LOCB to the operand in processor register R0 and places the result in R0

STORE R0, LOCC -Store the result in R0 to memory location LOCC

OUT PORTB

Computer as Multilevel Machines



Level 5 Problem-oriented language level

Translation (Compiler)

Level 4 Assembly language level

Translation (Assembler)

Level 3 Operating system (OS) machine level

Partial interpretation (OS)

Level 2 Instruction set architecture level

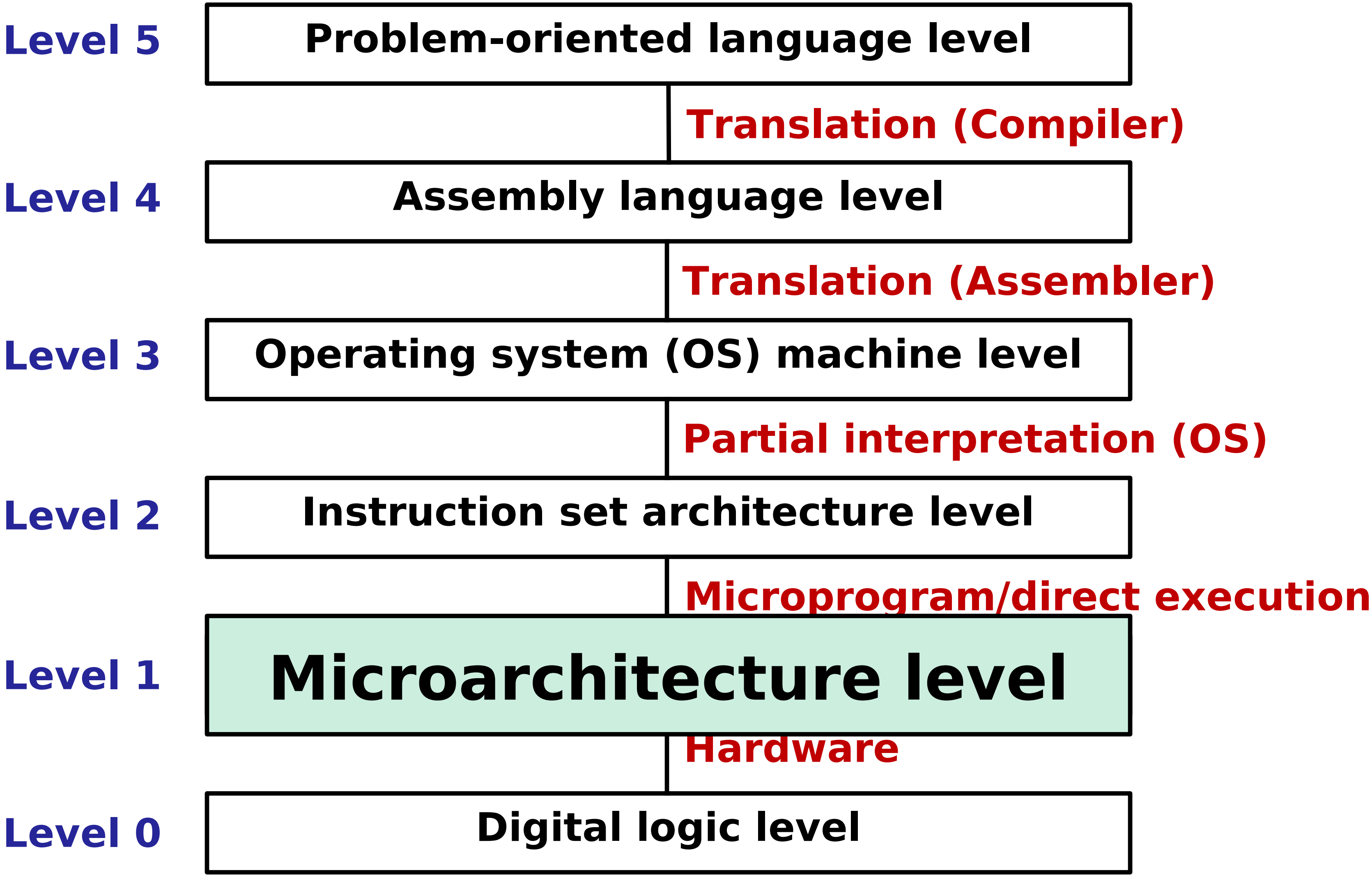
Microprogram/direct execution

Level 1 Microarchitecture level

Hardware

Level 0 Digital logic level

Computer as Multilevel Machines



Level 5 Problem-oriented language level

Translation (Compiler)

Level 4 Assembly language level

Translation (Assembler)

Level 3 Operating system (OS) machine level

Partial interpretation (OS)

Level 2 Instruction set architecture level

Microprogram/direct execution

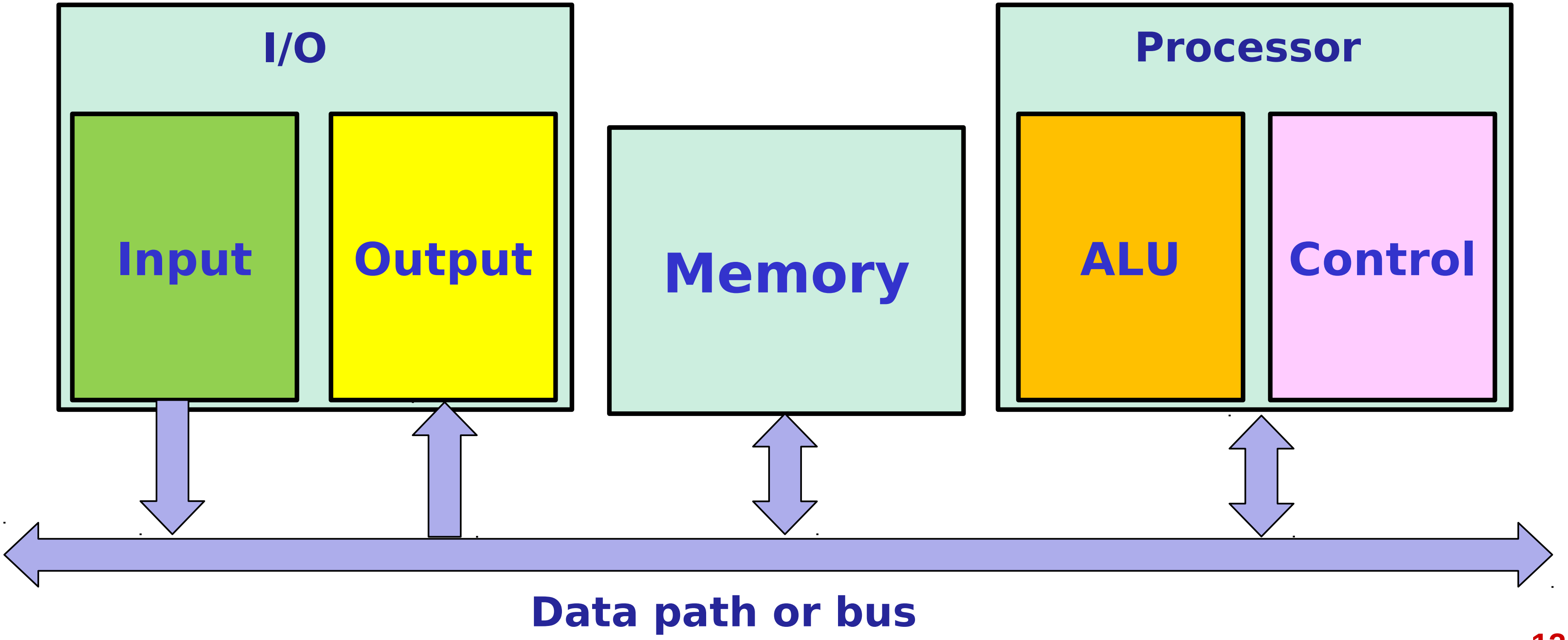
Level 1 **Microarchitecture level**

Hardware

Level 0 Digital logic level

Microarchitecture Level

- **Functional Units**:
  - Input Unit
  - Memory Unit
  - Arithmetic and Logic Unit (ALU)
  - Output Unit
  - Control Unit



Information Processed by Computer

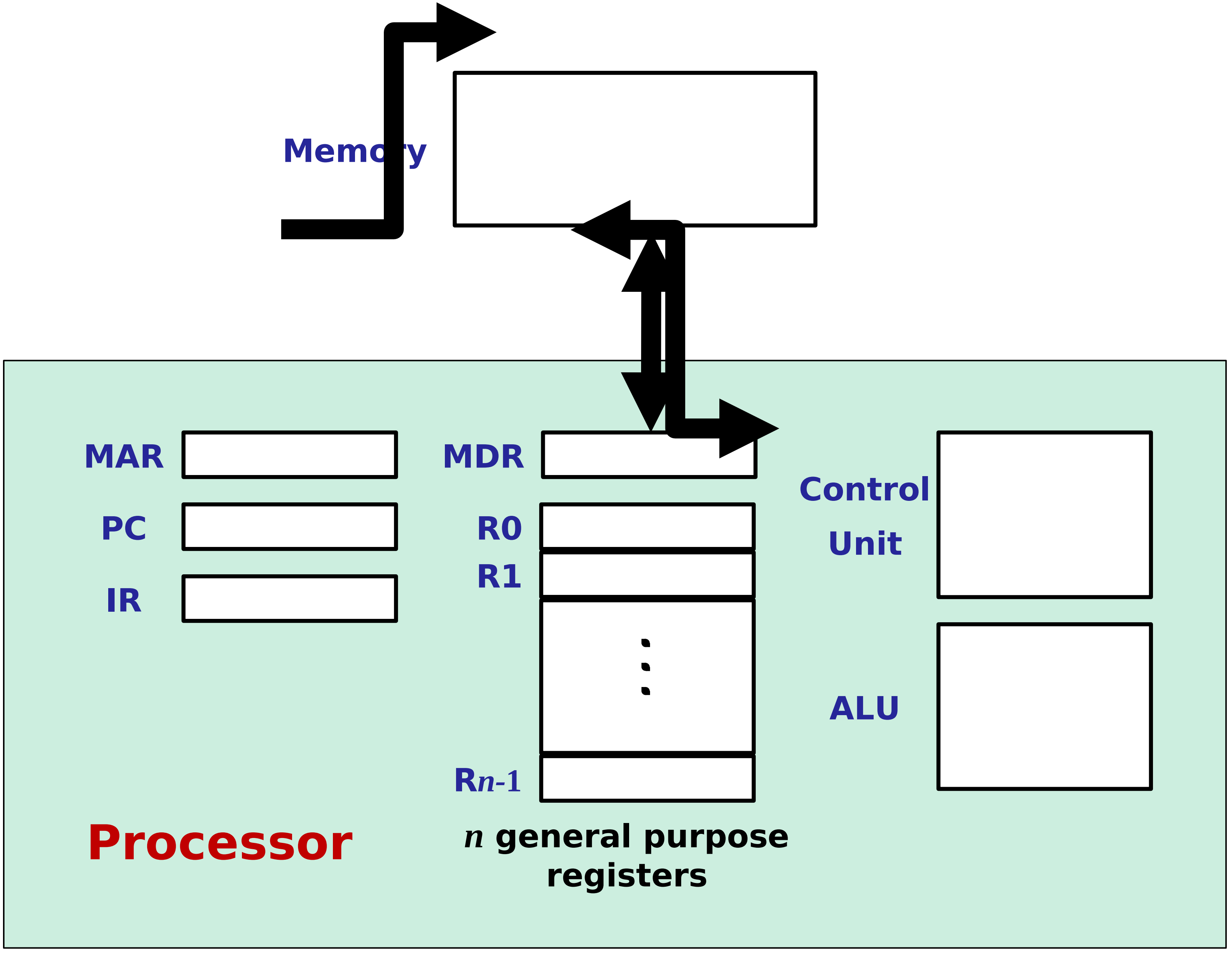
- **Instructions:**
  - Instructions are commands that
    - Govern the transfer of information with in computer as well as between the computer and its I/O devices
    - Specify the arithmetic and logic operations to be performed
  - A set of instructions that perform a task is called program
  - Usually a program is stored in memory
  - Processor fetches the instructions that make up the program from memory, one after another, perform desired operation
- **Data:**
  - They are numbers or encoded characters that are used as operands by the instructions
- Information handled by a computer is encoded in a suitable format (string of binary digits called bits – 0/1)

Execution of an Instruction

- Execution of an instruction requires to perform several steps
  - Instruction is fetched from memory into processor
  - If the instruction include operands, then the operands are fetched
  - If an instruction is for arithmetic operation, perform that operation on the fetched operands and store the results in destination location

- Transfers between memory and processor are started by sending the address of the memory location to be accessed to memory unit and issuing the appropriate control signals

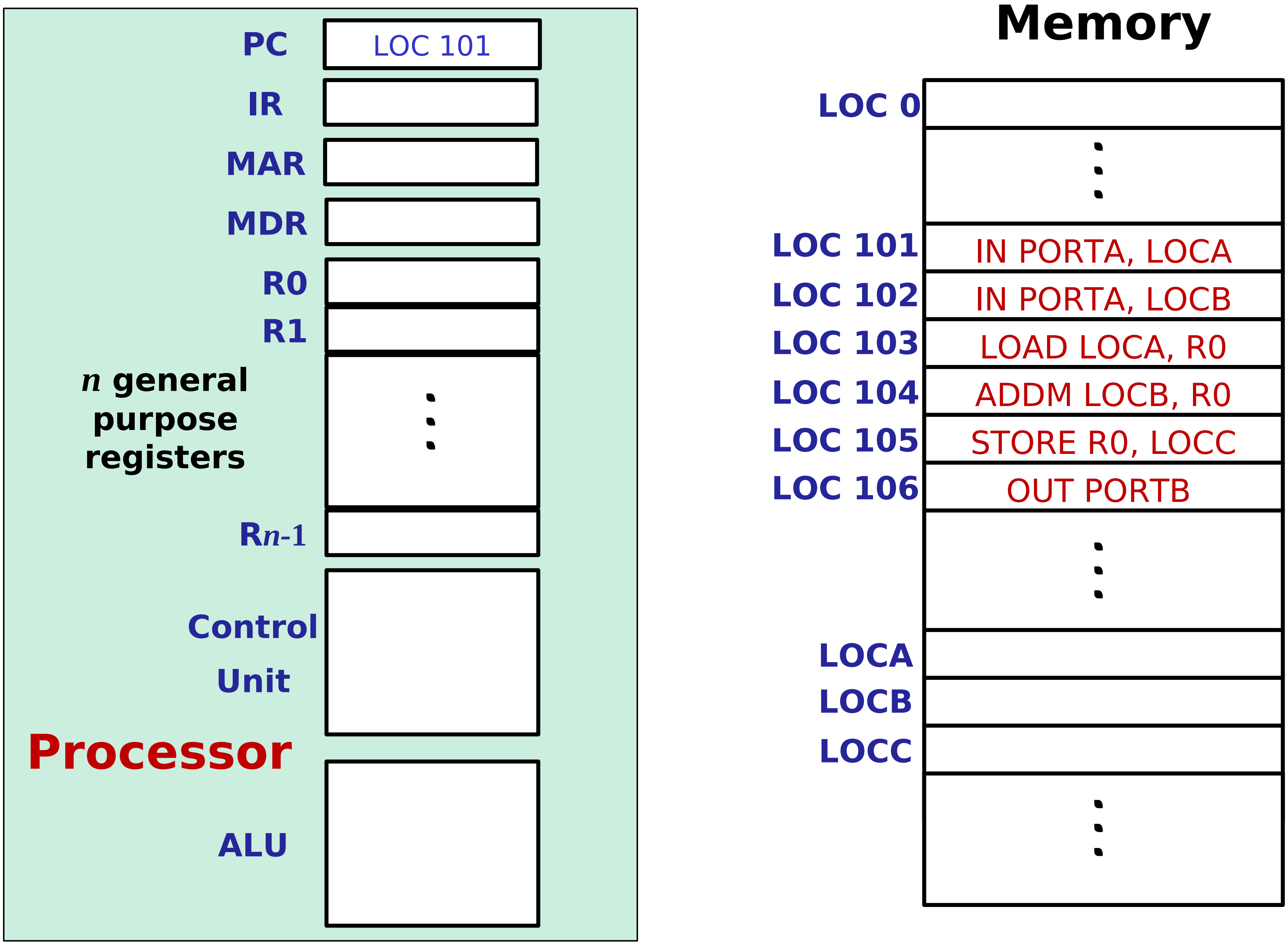
Operational Details



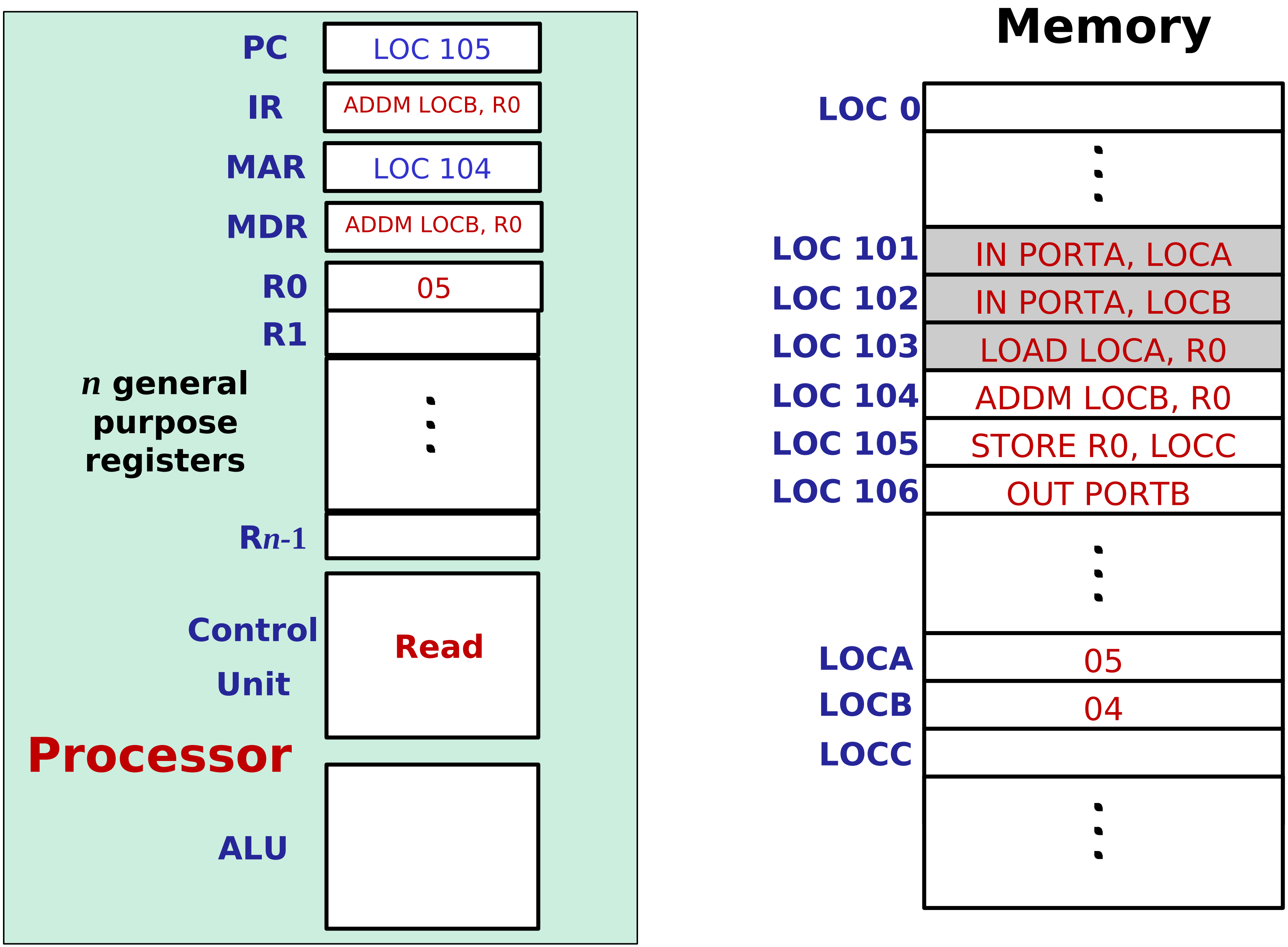
Registers in Processor

- General purpose registers
  - Hold the operands or address of the operand
  - Typically 16 to 32
- IR (Instruction Register)
  - Holds the instruction currently being executed
- PC (Program Counter)
  - Holds the memory address of the next instruction to be fetched and execured
- MAR (Memory Address Register)
  - Holds the address of the memory location to be accessed
- MDR (Memory Data Register)
  - Holds the data to be written into or read out of the addressed location

Execution of an Instruction



Fetching of an Instruction, ADDM



Processor

PC: LOC 105

IR: ADDM LOCB, R0

MAR: LOC 104

MDR: ADDM LOCB, R0

R0: 05

R1

n general purpose registers

R*n*-1

Control Unit: Read

ALU

Memory

LOC 0	
	⋮
LOC 101	IN PORTA, LOCA
LOC 102	IN PORTA, LOCB
LOC 103	LOAD LOCA, R0
LOC 104	ADDM LOCB, R0
LOC 105	STORE R0, LOCC
LOC 106	OUT PORTB
	⋮
LOCA	05
LOCB	04
LOCC	
	⋮

Syllabus

Introduction, Overview of basic digital building blocks; truth tables; basic structure of a digital computer,

Number representation, Assembly language programming for some processor,

Basic building blocks for the ALU, Adder,Subtractor, Shifter, Multiplication and division circuits, Control path microprogramming (only the idea), hardwired, logic;

External interface, Memory organization; Technology-ROM, RAM, EPROM, Flash etc.

Cache; Cache coherence protocol for uniprocessor (simple),

I/O Subblock, I/O techniques -interrupts, polling, DMA.

Evaluation

T1: 45 %

T2: 45 %

Assignment : 10%

Tue 2-3.30 pm, **Thu 3.30-5.00 pm**,

TA: Usha Kiran

Books

1. C.Hamacher, Z.Vranesic and S.Zaky, Computer Organization, 5th Ed., McGraw-Hill, 2002
2. J.P.Hayes, Computer Architecture and Organization, 3rd Ed
3. D.A.Patterson and J. L. Hennessy, Computer Organization and Design – The Hardware/Software Interface
4. William Stallings, “Computer Organization designing for performance”, 7 th Ed.