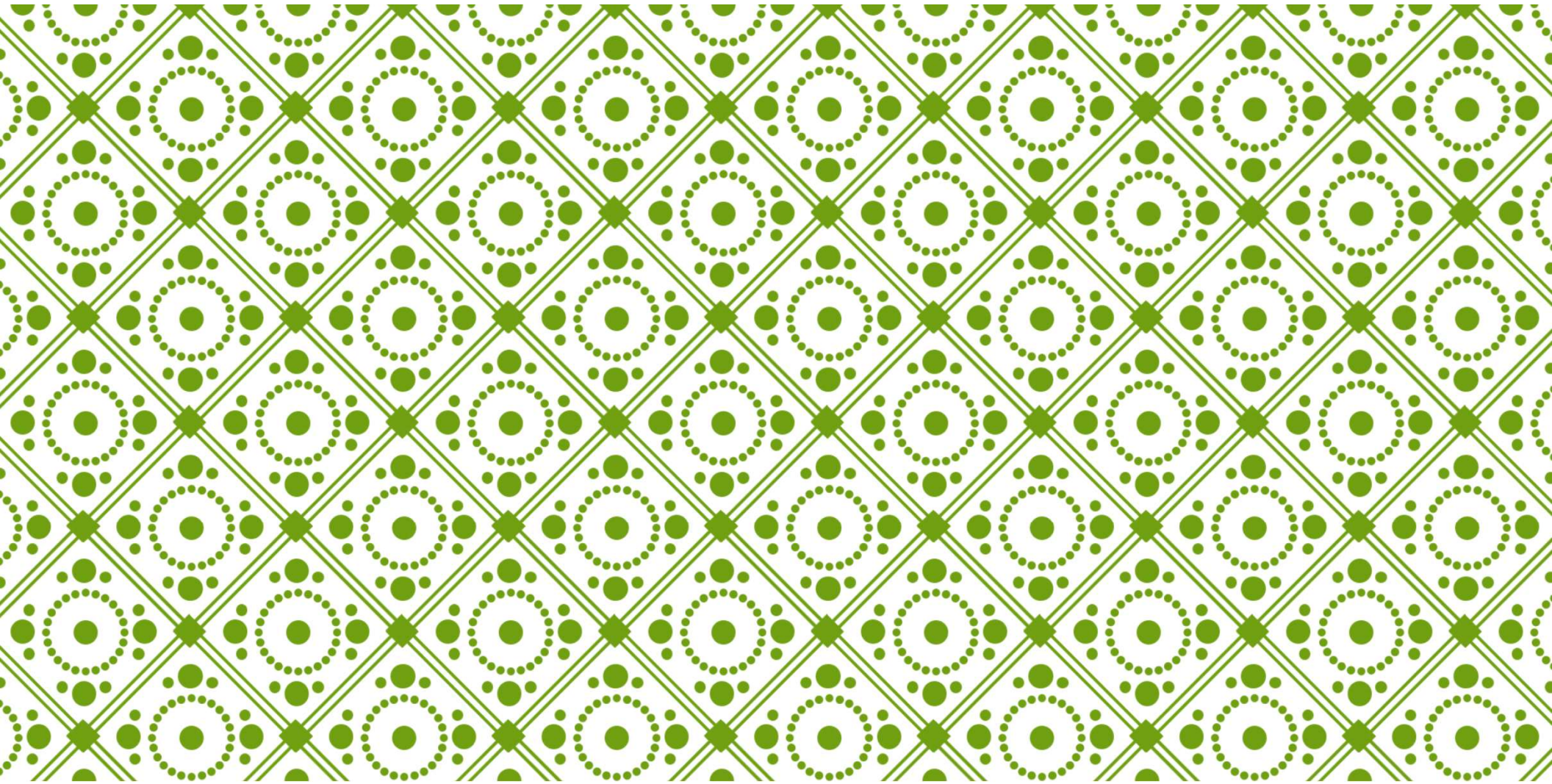


COMPUTER SYSTEM ARCHITECTURE
DSE 2157

COURSE OBJECTIVE

- Explain the different process in Instruction Set Architecture
- Infer the hardware implementation of addition, subtraction, multiplication and division and perform arithmetic operations
- Explain the concept of memory system and I/O handling techniques

COURSE TOPICS

- Basic operation of Computer System
- Instruction Set Architecture
- ALU Unit
- Basic Input/Output
- Memory System
- Introduction to Parallel Architecture

TEXTBOOKS

1. Carl Hamacher, ZvonkoVranesic and SafwatZaky, “Computer Organization and Embedded Systems”, (6e), McGraw Hill Publication, 2012

2. William Stallings, “Computer Organization and Architecture – Designing for Performance”, (9e), PHI, 2015

3. Mohammed Rafiquzzaman and Rajan Chandra, “ Modern Computer Architecture”, Galgotia Publications Pvt. Ltd., 2010

4. D.A. Patterson and J.L.Hennessy, “Computer Organization and Design-The Hardware/Software Interface”, (5e), Morgan Kaufmann, 2014

5. J.P.Hayes, “Computer Architecture and Organization”, McGraw Hill Publication, 1998

TOPICS OUTLINE

- Introduction
- Distinguish between Computer Architecture and Organization
- Block Diagram of a Computer
- Basic Operation of Computer

INTRODUCTION

- Computers have become part and parcel of our daily lives
- There is a requirement to understand how the computer works
- It is important to distinguish between two terms: ***Computer architecture and organization***

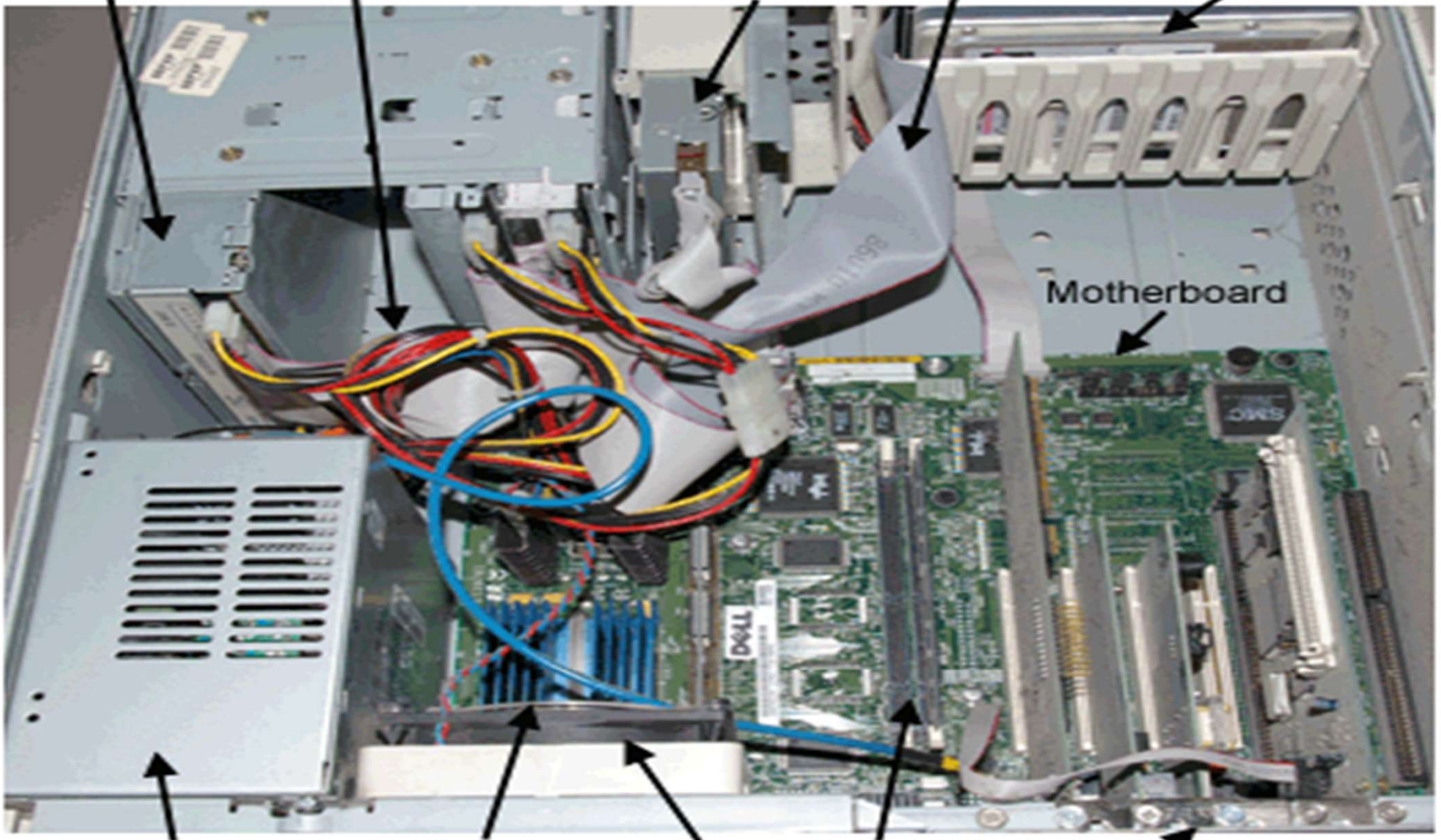
COMPUTER ARCHITECTURE

- Computer Architecture is concerned with the structure and behavior of the computer as seen by the user
- How to integrate the components to build a computer system to achieve a desired level of performance
- 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
- Architectural attributes include the instruction set, the number of bits used to represent various data types (e.g., numbers, characters), I/O mechanisms, and techniques for addressing memory

COMPUTER ORGANIZATION

- Computer Organization is concerned with the way the hardware components operate and the way they are connected together to form the computer system
- Design of the components and functional blocks using which computer systems are built
- It refers to the operational units and their interconnections that realize the architectural specifications
- Organizational attributes include those hardware details transparent to the programmer, such as control signals; interfaces between the computer and peripherals

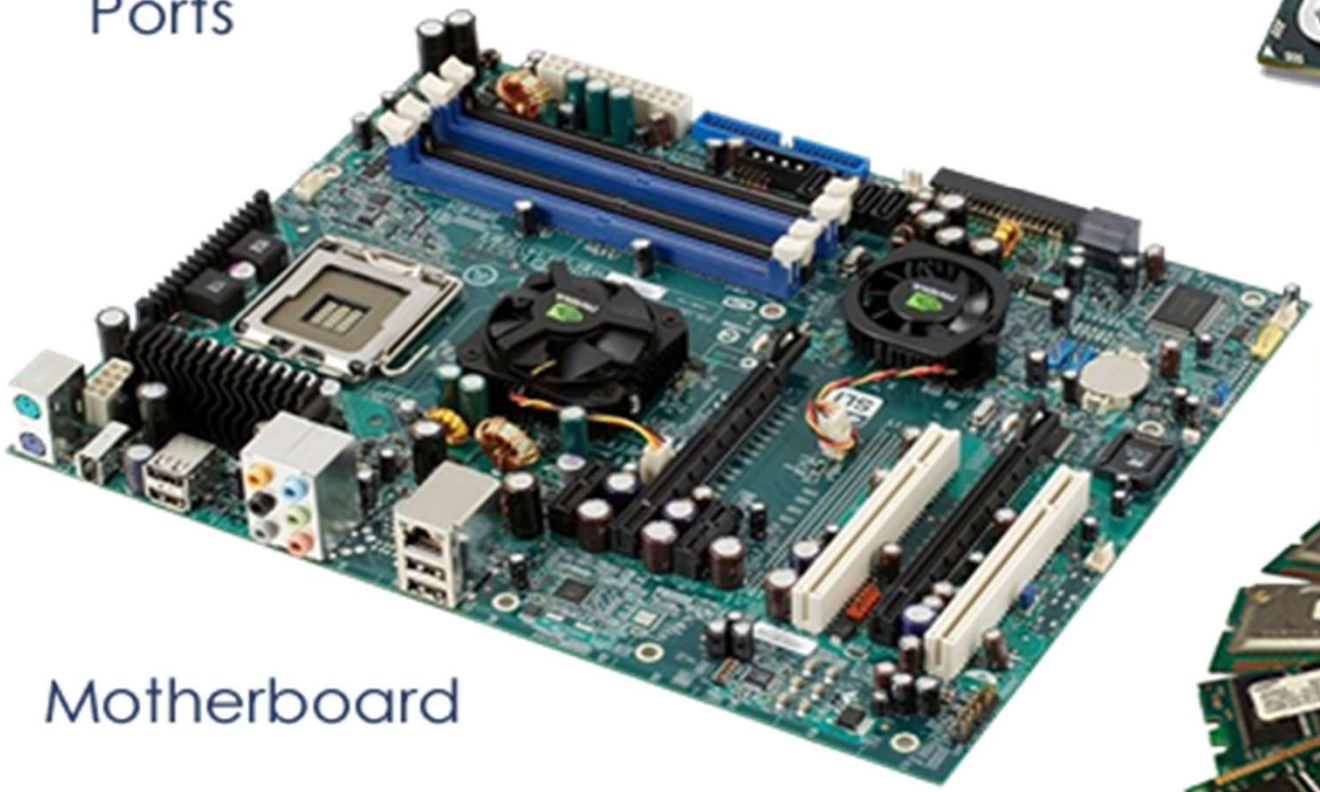
Inside of a computer case



Internal Computer Parts

Ports

Central Processing Unit



Motherboard

Hard Drive

RAM

BLOCK DIAGRAM OF A COMPUTER SYSTEM

- All instructions and data are stored in memory
- An instruction and the required data are brought into the processor for execution
- Input and Output devices interface with the outside world
- Referred to as von-Neumann architecture.

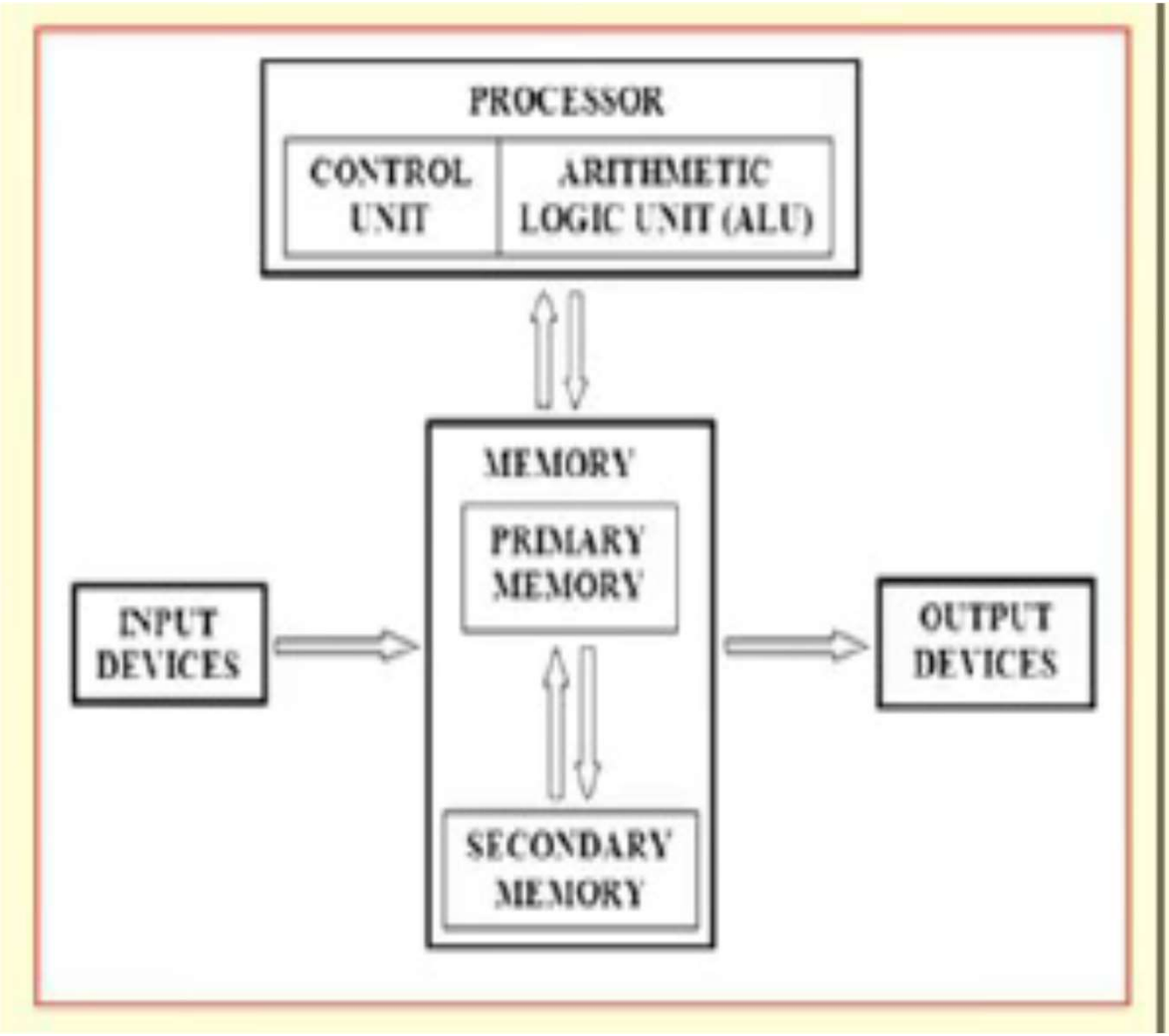


Figure 1 Block Diagram of a Computer System

THE PROCESSOR

- Also called Central Processing Unit (CPU)
- Consists of a Control unit and an Arithmetic Logic Unit (ALU)
  - All calculations happen inside the ALU
  - The Control Unit generates sequence of control signals to carryout all operations
- The processor fetches an instruction from memory for execution.
  - An instruction specifies the exact operation to be carried Out.
  - It also specifies the data that are to be operated on
  - A program refers to a set of instructions that are required to carry out some specific task (e.g. sorting a set of numbers).

INFORMATION IN A COMPUTER -- *INSTRUCTIONS*

Instructions specify commands to:

- Transfer information within a computer (e.g., from memory to ALU)
- Transfer of information between the computer and I/O devices (e.g., from keyboard to computer, or computer to printer)
- Perform arithmetic and logic operations (e.g., Add two numbers, Perform a logical AND).

A sequence of instructions to perform a task is called a program, which is stored in the memory.

Processor fetches instructions that make up a program from the memory and performs the operations stated in those instructions.

What do the instructions operate upon?

INFORMATION IN A COMPUTER -- **DATA**

- Data are the "operands" upon which instructions operate.

Data could be:

- Numbers,
- Encoded characters.

- Data, in a broad sense means any digital information.
- Computers use data that is encoded as a string of binary digits called bits

ARITHMETIC AND LOGIC UNIT (ALU)

- ALU contains several registers, some general-purpose and some special- purpose, for temporary storage Of data
- It contains circuitry to Carry out logic operations, like AND, OR, NOT, shift, compare, etc
- It contains circuitry to carry out arithmetic operations like addition, subtraction, multiplication, division, etc
- During instruction execution, the data (operands) are brought in and stored in some registers, the desired operation carried out, and the result stored back in some register or memory

CONTROL UNIT

- Acts as the nerve center that senses the states of various functional unit and sends control signals to control their states
- To carry out a specific operation (say, R1←R2 + R3), the control unit must generate control signals in a specific sequence
  - Enable the outputs of registers R2 and R3
  - Select the addition operation
  - Store the output of the adder circuit into register R1
- When an instruction is fetched from memory, the operation (called opcode) is decoded by the control unit, and the control signals issued

MEMORY UNIT

- Memory unit stores instructions and data.
  - Recall, data is represented as a series of bits.
  - To store data, memory unit thus stores bits.
- Two main types of memory subsystems
  - Primary or Main memory, which stores the active instructions and data for the program being executed on the processor
  - Secondary memory, which is used as a backup and stores all active and inactive programs and data, typically as files
- The processor only has direct access to the primary memory
- Main memory is classified again as ROM and RAM
- ROM holds system programs and firmware routines such as BIOS, POST, I/O Drivers that are essential to manage the hardware of a computer
- RAM is termed as Read/Write memory or user memory that holds run time program instruction and data

MEMORY UNIT (CONTD…)

- Processor reads instructions and reads/writes data from/to the memory during the execution of a program.
  - In theory, instructions and data could be fetched one bit at a time.
  - In practice, a group of bits is fetched at a time.
  - Group of bits stored or retrieved at a time is termed as "word"
  - Number of bits in a word is termed as the "word length" of a computer.
- In order to read/write to and from memory, a processor should know where to look:
  - "Address" is associated with each word location.

MEMORY UNIT (CONTD..)

- Processor reads/writes to/from memory based on the memory address:
  - Access any word location in a short and fixed amount of time based on the address.
  - Random Access Memory (RAM) provides fixed access time independent of the location of the word.
- Access time is known as "Memory Access Time".
- Memory and processor have to "communicate" with each other in order to read/write information.
  - In order to reduce "communication time", a small amount of RAM (known as Cache) is tightly coupled with the processor.
- Modern computers have three to four levels of RAM units with different speeds and sizes:
  - LI cache, L2 cache, L3 cache, primary memory, secondary memory
  - Objective is to provide faster memory access at affordable cost.

MEMORY UNIT (CONTD..)

- Primary storage of the computer consists of RAM units.
  - Fastest, smallest unit is Cache.
  - Slowest, largest unit is Main Memory.
- Primary storage is insufficient to store large amounts of data and programs.
  - Primary storage can be added, but it is expensive.
- Store large amounts of data on secondary storage devices:
  - Magnetic disks and tapes,
  - Optical disks (CD-ROMS).
  - Access to the data stored in secondary storage in slower, but take advantage of the fact that some information may be accessed infrequently.
- Cost of a memory unit depends on its access time, lesser access time implies higher cost.

MEMORY UNIT (CONTD..)

Fig 2: Various Memory Units

INPUT UNIT

- Binary information must be presented to a computer in a specific format.
- Input unit:
  - Interfaces with input devices.
  - Accepts binary information from the input devices.
  - Presents this binary information in a format expected by the computer.
  - Transfers this information to the memory or processor.

INPUT UNIT(CONTD…)

Fig 3:Input Devices

OUTPUT UNIT

- Computers represent information in a specific binary form
- Output units:
  - Interface with output devices.
  - Accept processed results provided by the computer in specific binary form.
  - Convert the information in binary form to a form understood by an output device.

OUTPUT UNIT(CONT'D)

Fig 4:Output Devices

SUMMARIZATION

- The computer accepts information in the form of programs and data through an input unit and stores it in the memory
- Information stored in the memory is fetched under program control into an arithmetic and logic unit, where it is processed
- Processed information leaves the computer through an output unit
- All activities in the computer are directed by the control unit

BASIC OPERATION OF COMPUTER

- The basic mechanism through which an instruction gets executed will be illustrated
- Two special-purpose registers are used:
  - Memory Address Register (MAR): Holds the address of the memory location to be accessed
  - Memory Data Register (MDR): Holds the data that is being written into memory, or will receive the data being read out from memory
- Memory considered as a linear array of storage locations (bytes or words) each with unique address

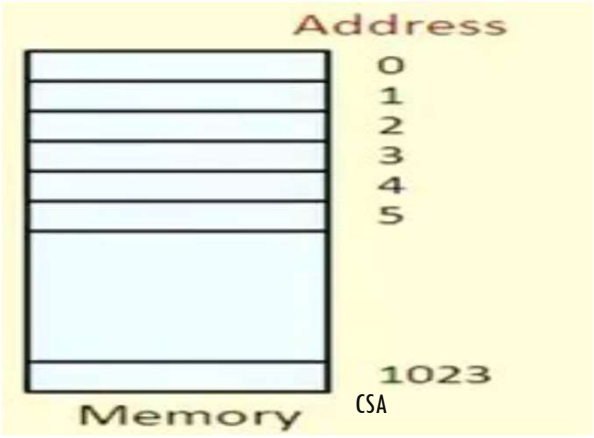


Fig 5:Primay Memory Location

CONNECTION BETWEEN THE PROCESSOR AND THE MAIN MEMORY

- To read data from memory
  a) Load the memory address into MAR
  b) Issue the control signal READ
  c) The data read from the memory is stored into MDR
- To write data into memory
  a) Load the memory address into MAR
  b) Load the data to be written into MDR
  c) Issue the control signal WRITE

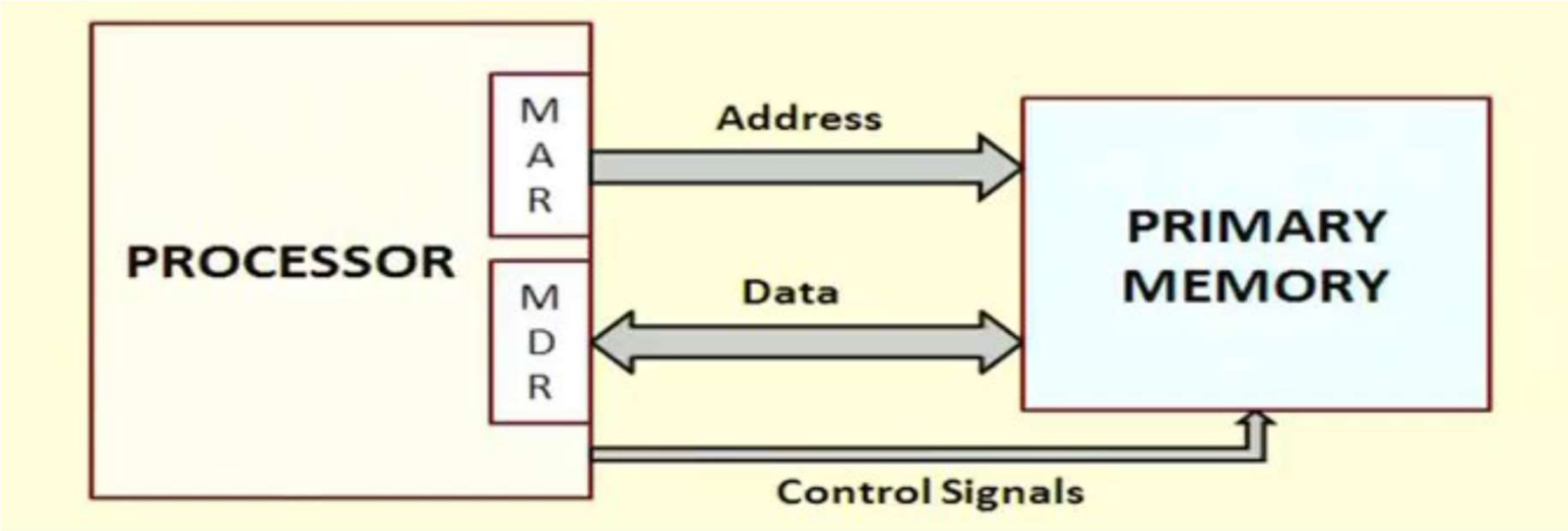
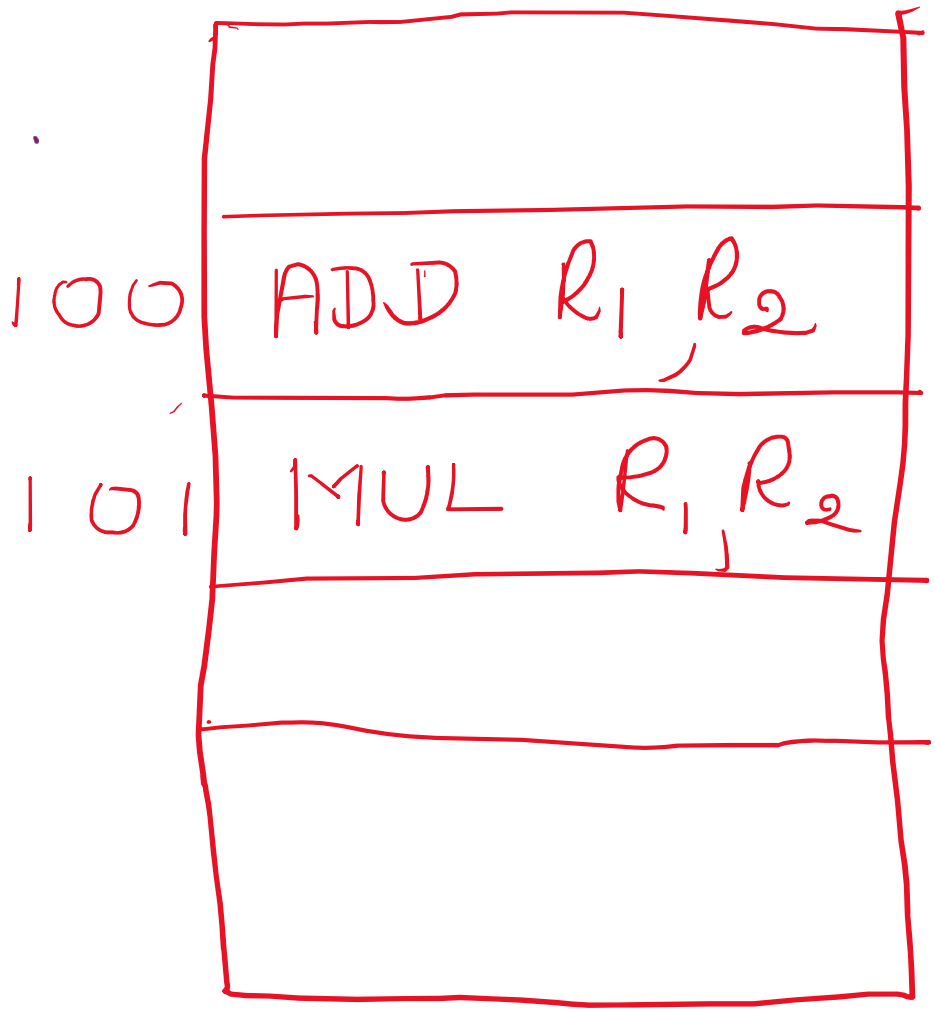


Fig 6:Connection between the processor and the main memory



PCounter = 100

101

IR = ADD R1, R2

01010 1001 1010

KEEPING TRACK OF PROGRAM / INSTRUCTIONS

Two special-purpose registers are used:

- Program Counter (PC): Holds the memory address of the next instruction to be executed
  - Automatically incremented to point to the next instruction when an instruction is being executed
- Instruction Register (IR): Temporarily holds an instruction that has been fetched from memory
  - Need to be decoded to find out the instruction type
  - Also contains information about the location of the data.

ARCHITECTURE OF PROCESSOR

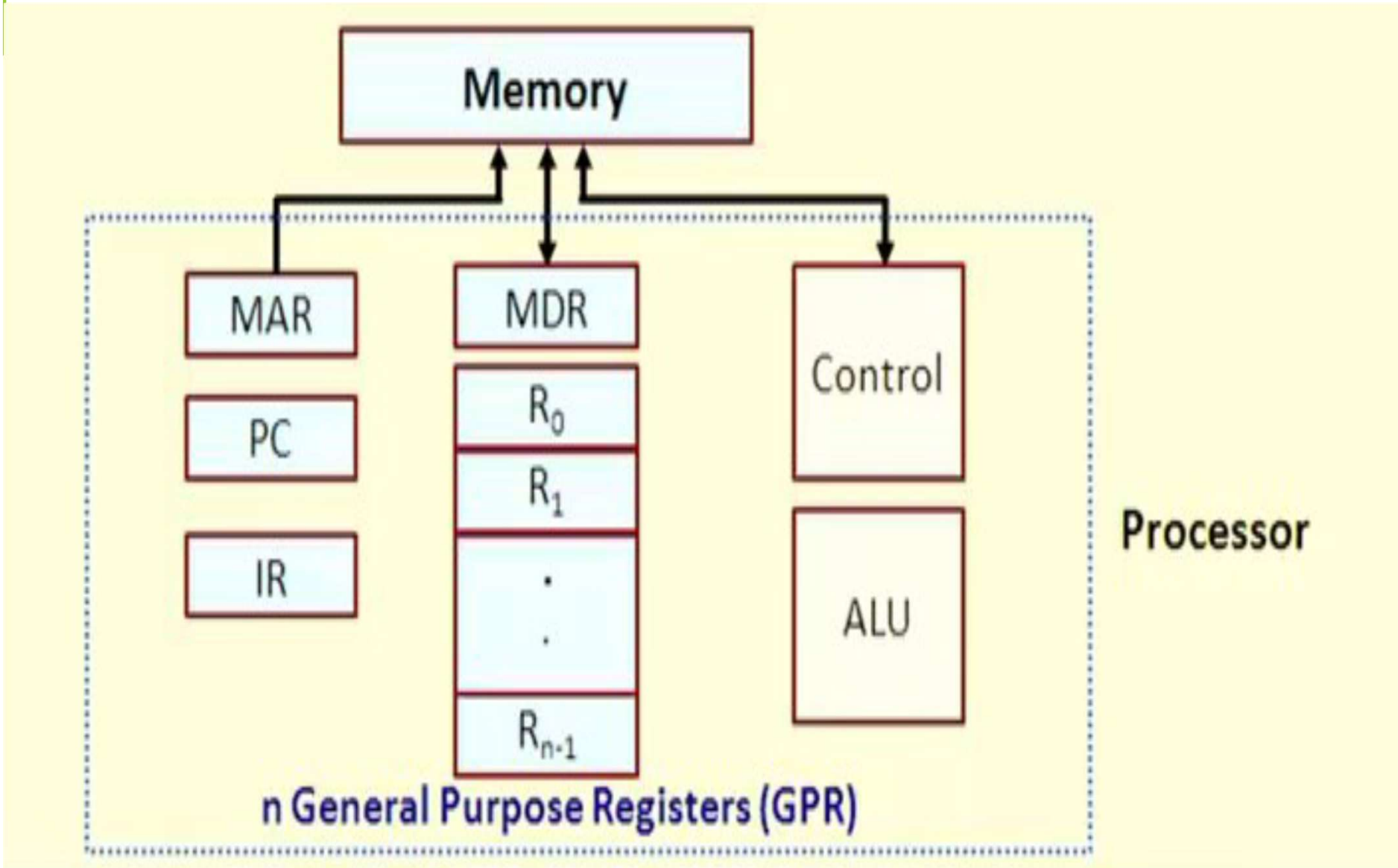


Fig 7:Architecture of Processor

EXAMPLE INSTRUCTIONS

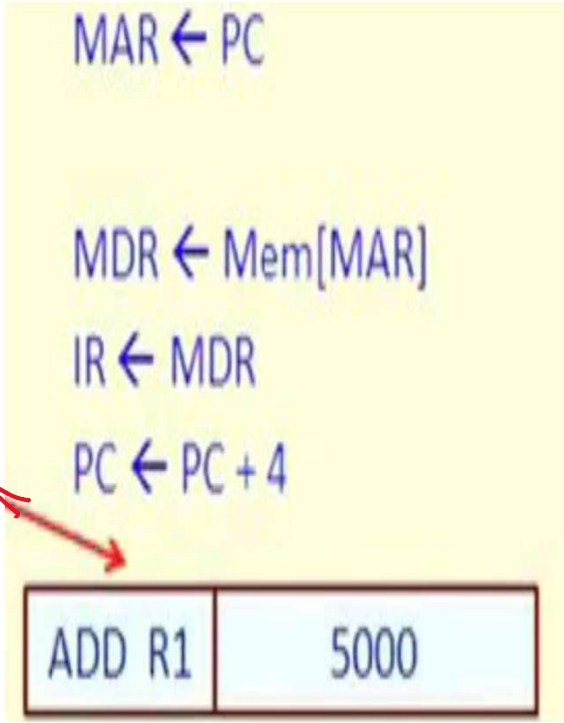
We shall illustrate the process of instruction execution with the help of the following two instructions:

a) ADD R1, LOCA - Add the contents of memory location LOCA (i.e. address Of the memory location is LOCA) to the contents of register R1. R1 + Mem[LOCA]

b) ADD R1, R2 - Add the contents of register R2 to the contents of register R1.

EXECUTION OF ADD R1,LOCA

- Assume that the instruction is stored in memory location 1000 the initial value of R1 is 50, and LOCA is 5000
- Before the instruction is executed, PC contains 1000
- Content of PC is transferred to MAR
- READ request is issued to memory unit
- The instruction is fetched to MDR
- Content of MDR is transferred to IR
- PC is incremented to point to the next instruction
- The instruction is decoded by the control unit



EXECUTION OF ADD R1,LOCA

- LOCA (i.e. 5000) is transferred (from IR) to MAR
- READ request is issued to memory unit
- The data is fetched to MDR
- The content of MDR is added to R1

MAR ← IR[Operand]

MDR ← Mem[MAR]

R1 ← R1 + MDR

EXECUTION OF ADD R1,LOCA

The steps being carried out are called micro-operations:

MAR ← PC
MDR ← Mem[MAR]
IR ← MDR
PC ← PC + 4
MAR ← IR[Operand]
MDR ← Mem[MAR]
R1 ← R1 + MDR

EXECUTION OF ADD R1,LOCA

R1	50

Address	Content
1000	ADD R1, LOCA
1004	...

5000	75

LOCA

1. PC = 1000
2. MAR = 1000
3. PC = PC + 4 = 1004
4. MDR = ADD R1, LOCA
5. IR = ADD R1, LOCA
6. MAR = LOCA = 5000
7. MDR = 75
8. R1 = R1 + MDR = 50 + 75 = 125

EXECUTION OF ADD RI,R2

- Assume that the instruction is stored in memory location 1500, the initial value of R1 is 50, and R2 is 200
- Before the instruction is executed, PC contains 1500
- Content of PC is transferred to MAR
- READ request is issued to memory unit
- The instruction is fetched to MDR
- Content of MDR is transferred to IR
- PC is incremented to point to the next instruction
- The instruction is decoded by the control unit
- R2 is added to R1

EXECUTION OF ADD RI,R2

R1	50
R2	200

Address	Instruction
1500	ADD R1, R2
1504	...

1. PC = 1500
2. MAR = 1500
3. PC = PC + 4 = 1504
4. MDR = ADD R1, R2
5. IR = ADD R1, R2
6. R1 = R1 + R2 = 250

BUS ARCHITECTURE

- The different functional modules must be connected in an organized manner to form an operational system
- Bus refers to a group of lines(wire) that serves as a connecting path for several devices
- The simplest way to connect the functional unit is to use the single bus architecture
  - Each wire in a bus can transfer one bit of information.
  - The number of parallel wires in a bus is equal to the word length of a computer
- Only one data transfer allowed in one clock cycle
- For multi-bus architecture, parallelism in data transfer is allowed

SYSTEM - LEVEL SINGLE BUS ARCHITECTURE

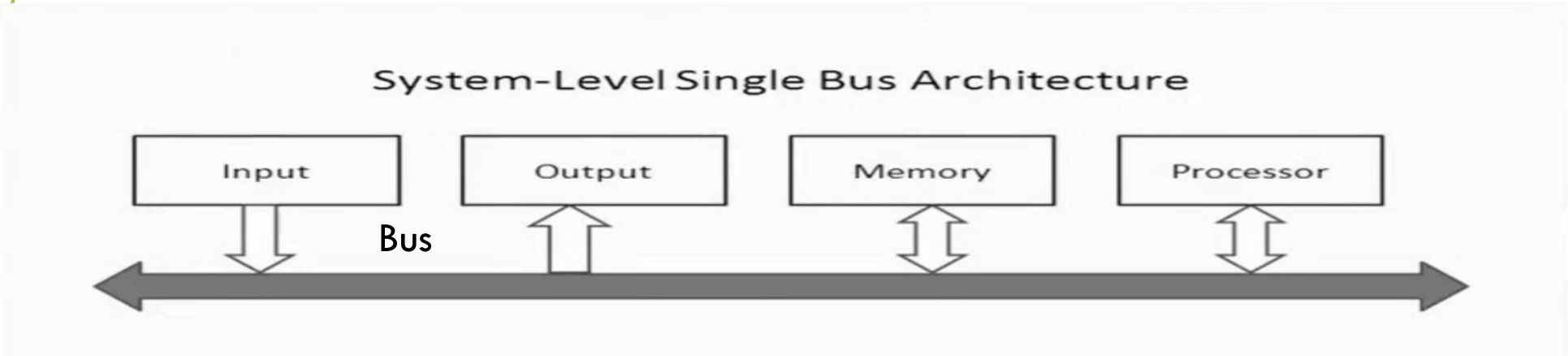


Fig 8:System-level Single Bus Architecture

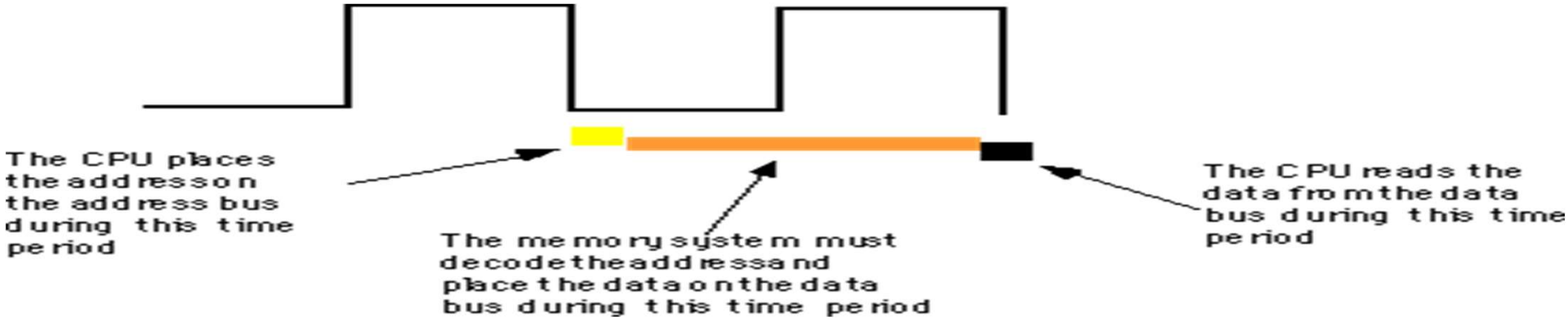
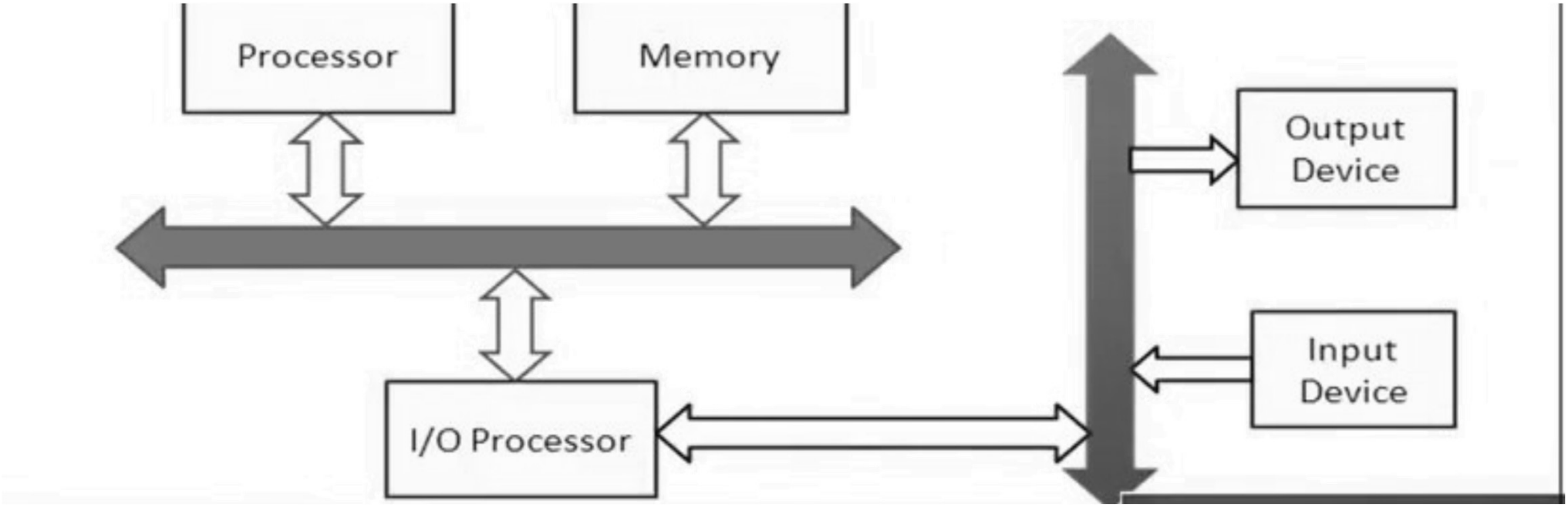


Fig 9 Processor clock cycle

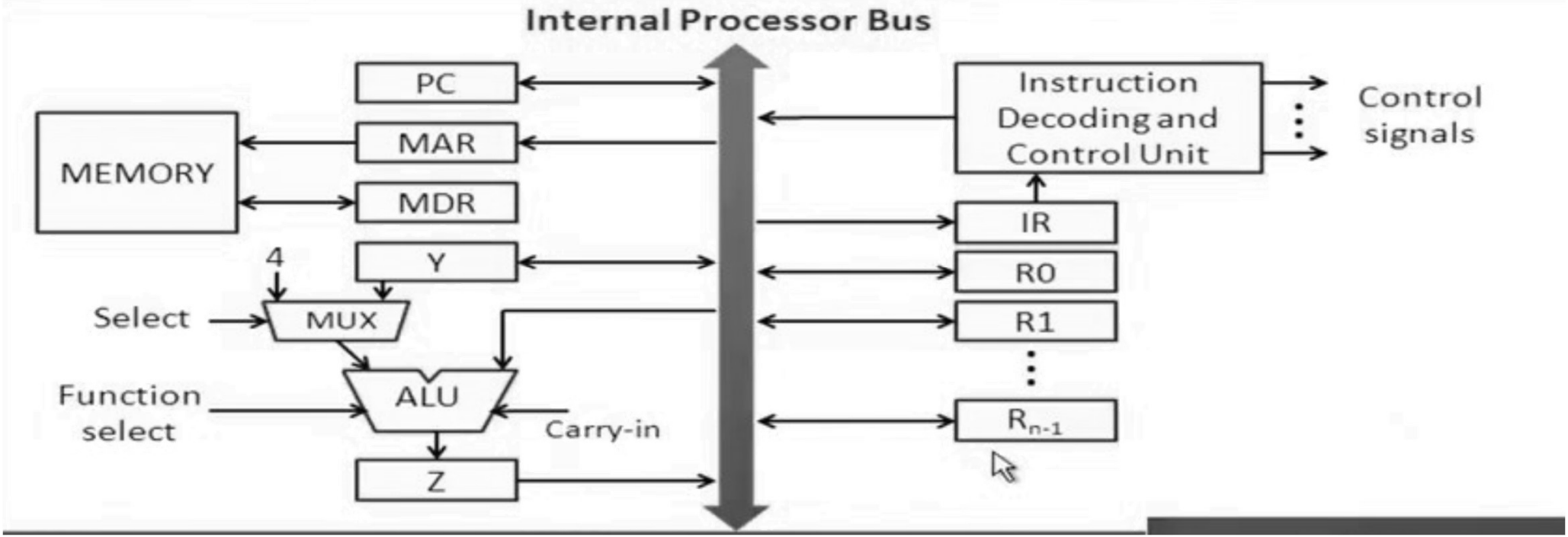
SYSTEM-LEVEL TWO-BUS ARCHITECTURE



- A bus dedicated to processor memory
- IO processor is also connected to it
- But for input and output there is a separate bus and this bus will be communicating with the IO processor in turn and the IO processor will then be communicating with the processor or the memory as and when it is required

INTERNAL PROCESSOR BUS

- There is a single bus inside the processor
  - ALU and the registers are all connected via the single bus
  - This bus is internal to the processor



END OF UNIT 1