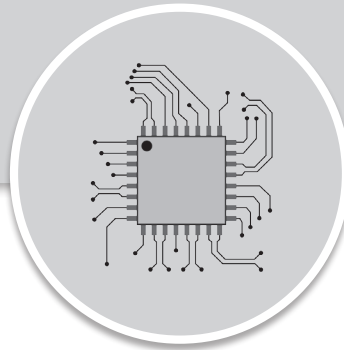


ELECTRONICS ENGINEERING

Computer Organization



Comprehensive Theory
with Solved Examples and Practice Questions





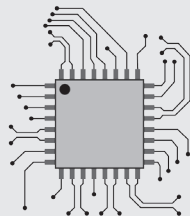
MADE EASY Publications Pvt. Ltd.

Corporate Office: 44-A/4, Kalu Sarai (Near Hauz Khas Metro Station), New Delhi-110016 | **Ph. :** 9021300500

Email : infomep@madeeasy.in | **Web :** www.madeeasypublications.org

Computer Organization

Copyright © by MADE EASY Publications Pvt. Ltd.
All rights are reserved. No part of this publication may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photo-copying, recording or otherwise), without the prior written permission of the above mentioned publisher of this book.



MADE EASY Publications Pvt. Ltd. has taken due care in collecting the data and providing the solutions, before publishing this book. In spite of this, if any inaccuracy or printing error occurs then **MADE EASY Publications Pvt. Ltd.** owes no responsibility. We will be grateful if you could point out any such error. Your suggestions will be appreciated.

EDITIONS

First Edition : 2020

Second Edition : 2021

Third Edition : 2022

Fourth Edition : 2023

Fifth Edition : 2024

Sixth Edition : 2025

CONTENTS

Computer Organization

CHAPTER 1

Basics of Computer Design3-47

- 1.1 Computer System 3
- 1.2 Data Storage in the Memory 11
- 1.3 Machine Instructions 12
- Student Assignments* 41

CHAPTER 2

CPU Design 48-78

- 2.1 Introduction 48
- 2.2 Datapath 50
- 2.3 Control Unit 54
- 2.4 Program Interrupt 71
- 2.5 Booth's Algorithm 72
- Student Assignments* 75

CHAPTER 3

Instruction Pipelining 79-112

- 3.1 Introduction 79
- 3.2 RISC Pipelining 80
- 3.3 Pipeline Hazards 86
- 3.4 Pipeline Performance Analysis 95
- 3.5 Speedup 97
- Student Assignments* 101

CHAPTER 4

Integer and Floating Point Representation 113-129

- 4.1 Fixed and Floating Point Format 113
- 4.2 IEEE Floating-Point Number Representation 122
- 4.3 Multiplying Floating-Point Numbers 124
- Student Assignments* 126



Computer Organization

Goal of the Subject

Basic understanding of computer organization includes:

- Understanding roles of processors.
- Understanding the concept of programs as sequences of machine instructions.
- Understanding the relationship between assembly language and machine language.
- Understanding the relationship between high-level compiled languages and assembly language.
- Understanding arithmetic and logical operations
- Understanding simple data path and control designs for processors.

Computer Organization

INTRODUCTION

In this book we tried to keep the syllabus of Computer Organization around the GATE syllabus. Each topic required for GATE is crisply covered with illustrative examples and each chapter is provided with Student Assignment at the end of each chapter so that the students get the thorough revision of the topics that he/she had studied. This subject is carefully divided into eight chapters as described below.

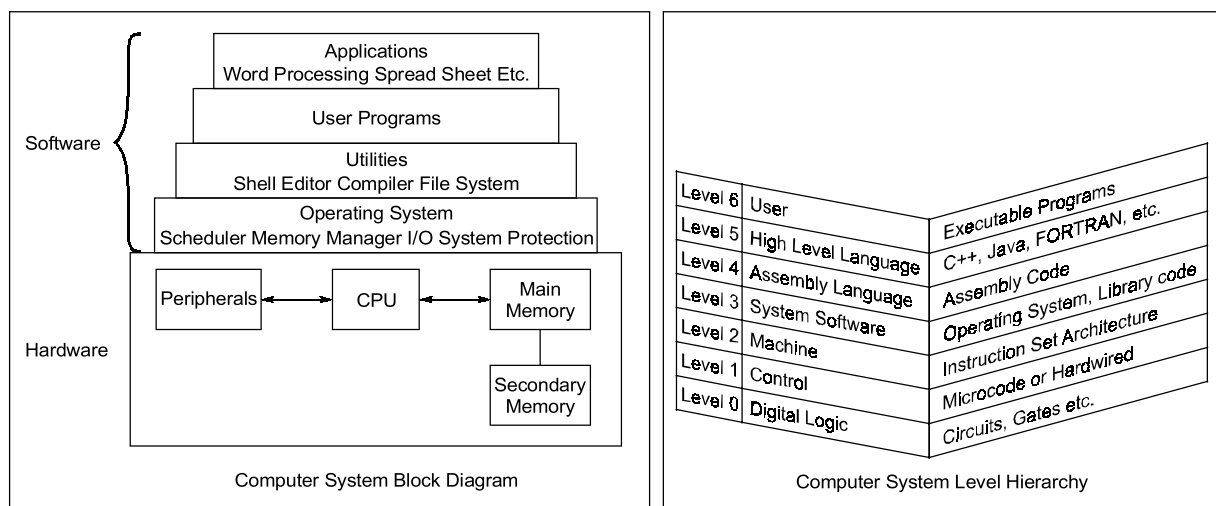
1. **Basics of Computer Design:** In this chapter we discuss the Computer System, Data storage in the memory and Machine instructions
2. **CPU Design:** In this chapter we discuss the Datapath and Control unit design first is hardwired control unit second is microprogrammed.
3. **Instruction Pipelining:** In this chapter we discuss Performance, Instruction processing, Pipeline design and issue, Pipeline hazards, Pipeline performance analysis and Speedup.
4. **Integer and Floating Point Representation:** In this chapter we discuss the Fixed and floating point format, IEEE floating -point number representation, Computer arithmetic, Adding 2's complement numbers and Multiplying floating-point numbers.



Basics of Computer Design

1.1 COMPUTER SYSTEM

Computer system is divided into two functional entities: Hardware and Software.

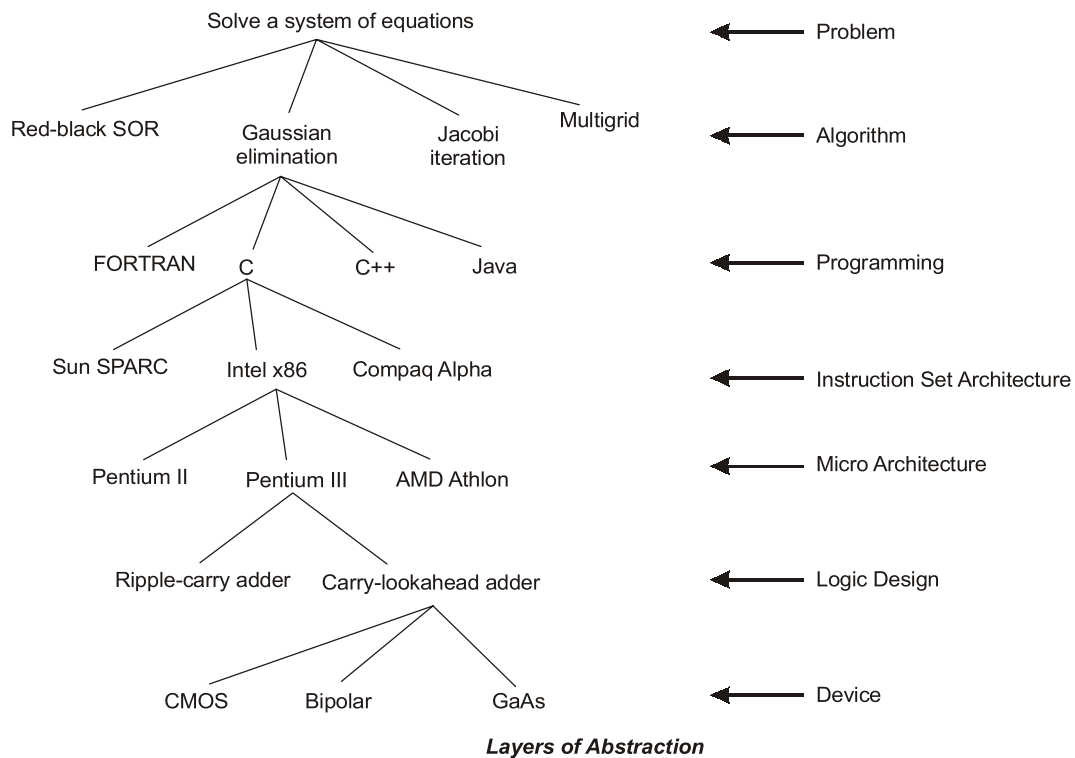


- **Hardware:** Lowest level in a computer are all the electronic circuits and physical devices from which it is built.
Hardware consisting of its physical devices (CPU, memory, bus, storage devices, ...)
- **Software:** Sequences of instructions and data that make computers do useful work.
Software, consisting of the programs it has (Operating system, applications, utilities, ...)
Program is a sequence of instructions for a particular task.
- Operating system is set of programs included in system software package and Link between hardware and user needs.

1.1.1 Layers of Abstraction

- **Problem Statement:** stated using “natural language”. It may be ambiguous or imprecise.
- **Algorithm:** step-by-step procedure, guaranteed to finish. It is definiteness, effective computability, and finiteness.

- **Program:** Express the algorithm using a computer language such as high-level language and low-level language.
- **Instruction Set Architecture (ISA):** It specifies the set of instructions the computer can perform using data types and addressing modes.
- **Micro-architecture:** It is detailed organization of a processor implementation.
- **Logic Circuits:** Combine basic operations to realize micro-architecture.
- **Devices:** Which is properties of materials and manufacturability.

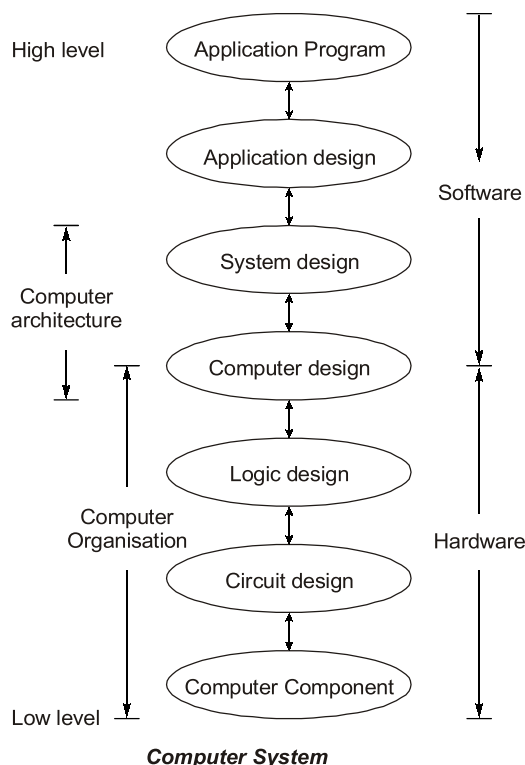


1.1.2 Computer Organization and Computer Architecture

Computer design: The determination of how to interconnect the components and which components to use based upon some specifications.

1.1.3 Computer Architecture (CA)

- Computer architecture is the conceptual design and fundamental operational structure of a computer system. It is a functional description of requirements and design implementations for the various parts of a computer.
- It is the science and art of selecting and interconnecting hardware components to create computers that meet functional, performance and cost goals.
- It deals with the architectural attributes like physical address memory, CPU and how they should be designed and made to coordinate with each other keeping the goals in mind.



1.1.4 Computer Organization (CO)

- Computer architecture comes before computer organization.
- Computer organization is how operational attributes are linked together and contribute to realise the architectural specifications.
- It encompasses all physical aspects of computer systems e.g. Circuit design, control signals, memory types.

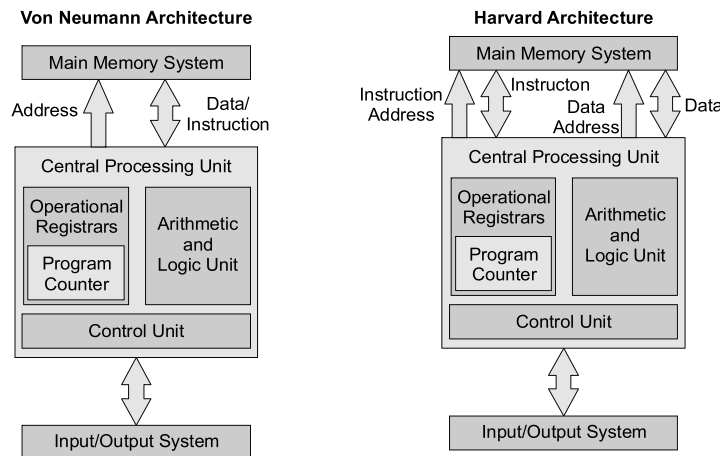
1.1.5 Computer Architecture Vs Computer Organization

Architecture and organization are independent, you can change the organization of a computer without changing its architecture.

1. The architecture indicates its hardware whereas the organization reveals its performance.
2. For designing a computer, its architecture is fixed first and then its organization is decided.

Computer Organization	Computer Architecture
<ul style="list-style-type: none"> • Computer organization deals with structural relationships that are not visible to the programmer (like clock frequency or the size of the physical memory). 	<ul style="list-style-type: none"> • Computer architecture deals with the functional behavior of a computer system as viewed by a programmer (like the size of a data type – 32 bits to an integer).
<ul style="list-style-type: none"> • A computer's organization expresses the realization of the architecture. 	<ul style="list-style-type: none"> • A computer's architecture is its abstract model and is the programmer's view in terms of instructions, addressing modes and registers.
<ul style="list-style-type: none"> • Organization describes how it does it. 	<ul style="list-style-type: none"> • Architecture describes what the computer does.

Von-Neumann Architecture Vs Harvard Architecture



1.1.6 Evolution of Digital Computers

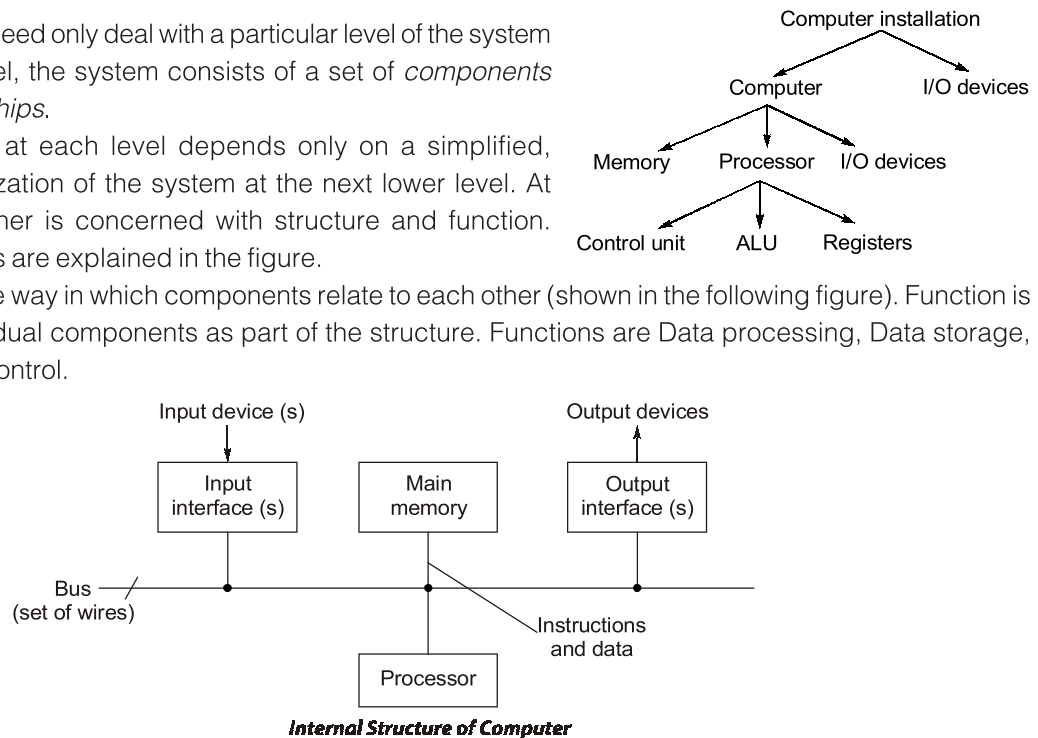
First generation: Vacuum tube computers (1945~1953)	Second generation: Transistorized computers (1954~1965)
<ul style="list-style-type: none"> Program and data reside in the same memory (stored program concepts: John von Neumann). Vacuum tubes were used to implement the functions (ALU & CU design). Magnetic core and magnetic tape storage devices are used. Using electronic vacuum tubes, as the switching components. Assembly level language is used. 	<ul style="list-style-type: none"> Transistor were used to design ALU & CU. High Level Language is used (FORTRAN). To convert HLL to MLL compiler were used. Separate I/O processor were developed to operate in parallel with CPU, thus improving the performance. Invention of the transistor which was faster, smaller and required considerably less power to operate.
Third generation: Integrated circuit computers (1965~1980)	Fourth generation: Very large scale integrated (VLSI) computers (1980~2000)
<ul style="list-style-type: none"> IC technology improved. Improved IC technology helped in designing low cost, high speed processor and memory modules. Multiprogramming, pipelining concepts were incorporated. DOS allowed efficient and coordinate operation of computer system with multiple users. Cache and virtual memory concepts were developed. More than one circuit on a single silicon chip became available. 	<ul style="list-style-type: none"> CPU termed as microprocessor INTEL, MOTOROLA, TEXAS, NATIONAL semiconductors started developing micro-processor. Workstations, microprocessor (PC) and Notebook computers were developed. Interconnection of different computer for better communication LAN, MAN and WAN. Computational speed increased by 1000 times. Specialized processors like Digital Signal Processor were also developed.
Fifth generation: System-on-chip (SOC) computers (2000~)	
<ul style="list-style-type: none"> E-Commerce, E-banking, home office. ARM, AMD, INTEL, MOTOROLA. High speed processor - GHz speed. Because of submicron IC technology lot of added features in small size. 	

1.1.7 Structure and Function of a Computer System

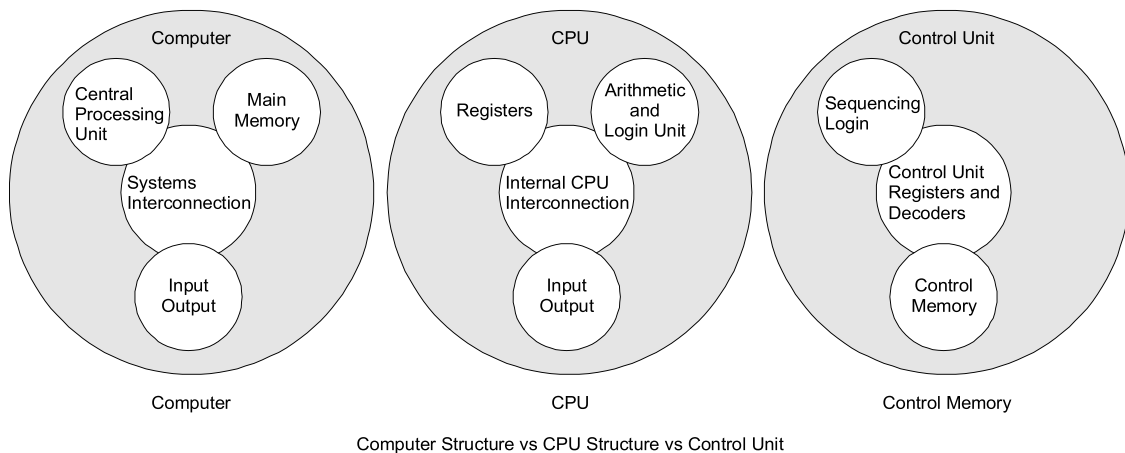
The designer need only deal with a particular level of the system at a time. At each level, the system consists of a set of *components and their interrelationships*.

The behavior at each level depends only on a simplified, abstracted characterization of the system at the next lower level. At each level, the designer is concerned with structure and function. Important relationships are explained in the figure.

Structure is the way in which components relate to each other (shown in the following figure). Function is the operation of individual components as part of the structure. Functions are Data processing, Data storage, Data movement and Control.



1.1.8 Components of Computer Structure



- 1. Input Unit:** Computers can understand only machine language. Therefore for converting data from human language to machine language we use some special peripheral devices which are called input device.
Examples: Keyboard, Mouse, Joystick, etc.
- 2. Output Unit:** After passing instructions for solving particular problem, the results came out from computer comes in machine language and this is very difficult to convert that results into human language. There are several such peripheral devices which help us to convert the machine language data into human acceptable data. These devices are called output devices.
Examples: Monitor, Printer, LCD, LED etc.

3. **Memory Unit:** Which is used to store data in computer.

Memory unit performs the following actions

- (a) Stores data and instructions required for processing.
- (b) Stores the intermediate results obtain during processing.
- (c) Stores final results before sending it to output unit.

Two class of storage units: (i) Primary Memory (ii) Secondary Memory

Two types of primary memory are RAM (Random Access Memory) and ROM (Read Only Memory). RAM is used to store data temporarily during the program execution. ROM is used to store data and program which is not going to change.

Secondary Memory is used for bulk storage or mass storage to store data permanently.

4. **CPU:** It is main unit of the computer system. It is responsible for carrying out computational task.

The major structural components of a CPU are:

- (a) **Control Unit (CU):** Controls the operation of the CPU and hence the computer.
- (b) **Arithmetic and Logic Unit (ALU):** Performs computer's data processing functions.
- (c) **Register:** Provides storage internal to the CPU.
- (d) **CPU Interconnection:** communication among the control unit, ALU, and register.

1.1.9 Bus Structure

Bus: It is a group of wires (lines or signals) which carries information from CPU to peripherals or peripherals to CPU. The CPU and Memory are connected by Data Bus, Address Bus and Control Bus.

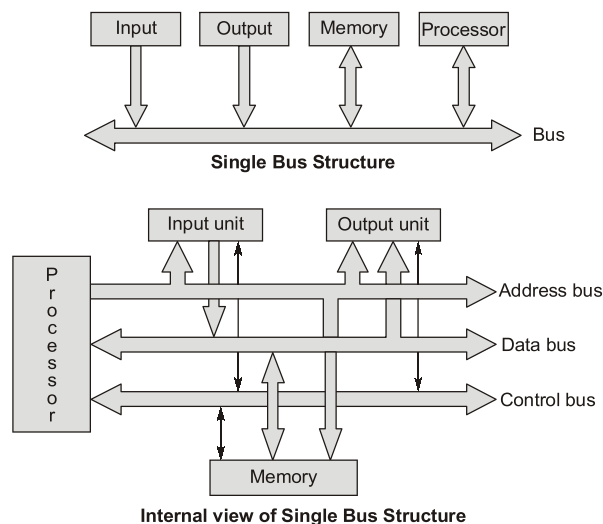
Address Bus: It is unidirectional bus which carries address information bits form processor to peripherals.

Data Bus: It is bidirectional bus which carries data information bit form processor to peripherals and vice-versa.

Control Bus: It is bidirectional bus which carries control signals form processor to peripherals and vice-versa.

1.1.10 Types of Bus Structure

- **Single bus structure:** Common bus used (shown in the following figure) to communicate between peripherals and microprocessor.

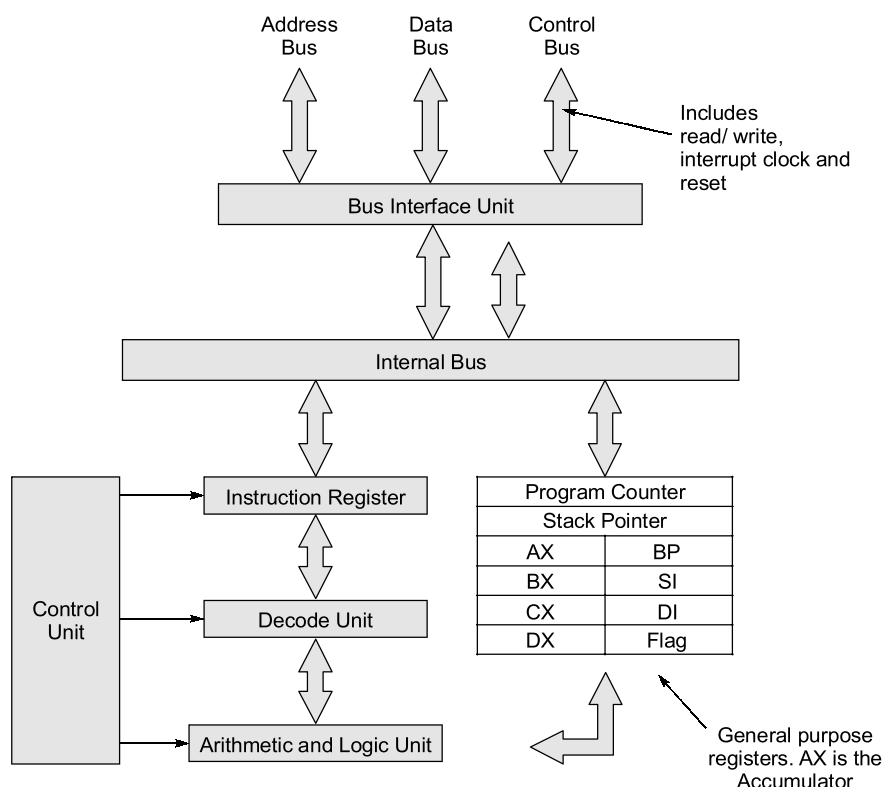


- **Two Bus Structure:** One bus can be used to fetch instruction other can be used to fetch data, required for execution. It improves the performance, but cost increases.

1.1.11 CISC and RISC Architectures

CISC (Complex Instruction Set Computers)	RISC (Reduced Instruction Set Computers)
• Large instruction set	• Compact instruction set
• Instruction formats are of different lengths	• Instruction formats are all of the same length
• Instructions perform both elementary and complex operations	• Instructions perform elementary operations
• Control unit is micro-programmed	• Control unit is simple and hardwired
• Not pipelined or less pipelined	• Pipelined
• Single register set	• Multiple register set
• Numerous memory addressing options for operands	• Compiler and IC developed simultaneously
• Emphasis on hardware	• Emphasis on software
• Includes multi-clock complex instructions	• Single-clock, reduced instruction only
• Memory-to-memory: "LOAD" and "STORE" incorporated in instructions	• Register to register: "LOAD" and "STORE" are independent instructions
• Small code sizes, high cycles per second	• Low cycles per second, large code sizes
• Transistors used for storing complex instructions	• Spends more transistors on memory registers
Examples of CISC processors: <ul style="list-style-type: none"> • VAX • PDP-11 • Motorola 68000 family • Intel x86 architecture based processors. 	Examples of RISC processors <ul style="list-style-type: none"> • Apple iPods (custom ARM7TDMI SoC) • Apple iPhone (Samsung ARM1176JZF) • Nintendo Game Boy Advance (ARM7) • Sony Network Walkman (Sony in-house ARM based chip)

1.1.12 General CPU Architecture (8086 Microprocessor)



Summary

- **Computer:** A device that accepts input, processes data, stores data, and produces output, all according to a series of stored instructions.
- **Hardware:** Includes the electronic and mechanical devices that process the data; refers to the computer as well as peripheral devices.
- **Software:** A computer program that tells the computer how to perform particular tasks.
- **Computer organization:** Interconnection of hardware to form the computer system
- **Computer architecture:** the structure and behaviour of the computer perceived by the user.
- **Input:** Whatever is put into a computer system.
- **Data:** Refers to the symbols that represent facts, objects, or ideas.
- **Information:** The results of the computer storing data as bits and bytes; the words, numbers, sounds, and graphics.
- **Output:** Consists of the processing results produced by a computer.
- **Main Memory:** Area of the computer that temporarily holds data waiting to be processed, stored, or output. *Example:* Cache and Main memory
- **Secondary Storage:** Area of the computer that holds data on a permanent basis when it is not immediately needed for processing. *Example:* Disk, Floppy, etc.
- **Instruction set:** It is a list of all the instructions, that a processor can execute.
- **Addressing Modes:** Specifies a rule for interpreting or modifying the address field of the instruction (before the operand is actually referenced)
- **Implied Mode:** Address of the operands are specified implicitly in the definition of the instruction. No need to specify address in the instruction.
- **Immediate Mode:** Instead of specifying the address of the operand, operand itself is specified.
- **Register Mode:** Address specified in the instruction is the register address. Designated operand need to be in a register.
- **Register Indirect Mode:** Instruction specifies a register which contains the memory address of the operand.
- **Direct Address Mode:** Instruction specifies the memory address which can be used directly to access the memory.
- **Indirect Addressing Mode:** The address field of an instruction specifies the address of a memory location that contains the address of the operand.

**OBJECTIVE
BRAIN TEASERS**

Q.1 What does CISC and RISC means?

- (a) common instruction set controller and rare instruction set controller
- (b) complex instruction set controller and reduced instruction set controller

- (c) compiled instruction set source code and recompiled instruction source code
- (d) none of the above

Q.2 A 32-bit address bus allows access to a memory of capacity

- (a) 64 MB
- (b) 16 MB
- (c) 1 GB
- (d) 4 GB

- Q.3** The system bus is made up of
 (a) data bus
 (b) data bus and address bus
 (c) data bus and control bus
 (d) data bus, control bus and address bus
- Q.4** Which of the following is not involved in a memory write operation?
 (a) MAR (b) PC
 (c) MDR (d) data bus
- Q.5** The read/write line
 (a) belongs to the data bus
 (b) belongs to the control bus
 (c) belongs to the address bus
 (d) CPU bus
- Q.6** _____ is a piece of hardware that executes a set of machine-language instructions.
 (a) controller (b) bus
 (c) processor (d) motherboard
- Q.7** Given below are some statements associated with the registers of a CPU. Identify the false statement.
 (a) The program counter holds the memory address of the instruction in execution.
 (b) Only opcode is transferred to the control unit.
 (c) An instruction in the instruction register consists of the opcode and the operand.
 (d) The value of the program counter is incremented by 1 once its value has been read to the memory address register.
- Q.8** The following are four statements about Reduced Instruction Set Computer (RISC) architectures.
 1. The typical RISC machine instruction set is small, and is usually a subset of a CISC instruction set.
 2. No arithmetic or logical instruction can refer to the memory directly.
 3. A comparatively large number of user registers are available.
 4. Instructions can be easily decoded through hard-wired control units.
 Which of the above statements is true?
 (a) 1 and 3 only (b) 1, 3 and 4 only
 (c) 1, 2 and 3 only (d) All of these
- Q.9** The word length of a CPU is defined as
 (a) the maximum addressable memory size
 (b) the width of a CPU register (integer or float point)
 (c) the width of the address bus
 (d) the number of general purpose CPU registers
- Q.10** Which of the following statements is false about CISC architectures?
 (a) CISC machine instructions may include complex addressing modes, which require many clock cycles to carry out.
 (b) CISC control units are typically micro-programmed, allowing the instruction set to be more flexible.
 (c) In the CISC instruction set, all arithmetic/logic instructions must be register based.
 (d) CISC architectures may perform better in network centric applications than RISC.
- Q.11** Consider a high-level language statement i – then which addressing mode is suitable for it?
 (a) autoincrement (b) indexed
 (c) displacement (d) autodecrement
- Q.12** In a 16-bit instruction code format 3 bit operation code, 12 bit address and 1 bit is assigned for address mode designation. How much data memory space is available
 (a) 4 MB (b) 8 KB
 (c) 2 KB (d) 2 GB
- Q.13** Match **List-I** with **List-II** and select the correct answer using the codes given below the lists:
- List-I**
 A. $\text{Regs}[R_4] \leftarrow \text{Regs}[R_4] + \text{Regs}[R_3]$
 B. $\text{Regs}[R_4] \leftarrow \text{Regs}[R_4] + 3$
 C. $\text{Regs}[R_4] \leftarrow \text{Regs}[R_4] + \text{Mem}[\text{Regs}[R_1]]$
- List-II**
 1. Immediate
 2. Register
 3. Register indirect
- Codes:**
- | | A | B | C |
|-----|---------------|---|---|
| (a) | 3 | 2 | 1 |
| (b) | 2 | 1 | 3 |
| (c) | 1 | 2 | 3 |
| (d) | None of these | | |

- Q.14** Relative Addressing Mode is used to write Position-Independent code because
- The Code in this mode is easy to atomize
 - The Code in this mode is easy to make resident
 - The Code in this mode is easy to relocate in the memory
 - Code executes faster in this mode

- Q.15** Consider an $(n + k)$ bit instruction with a k -bit opcode and single n -bit address. Then this instruction allow _____ operations and _____ addressable memory cells.
- $2^k, 2^{n+k}$
 - $2^{n+k}, 2^{n+k}$
 - $2^k, 2^n$
 - $2^{n+k}, 2^{n+1}$

- Q.16** The register which holds the address of the location to or from which data are to be transferred is known as
- index register
 - instruction register
 - memory address register
 - memory data register

- Q.17** Halt operation comes under _____.
- data transfer
 - control transfer
 - conversion
 - I/O transfer

- Q.18** In four-address instruction format, the number of bytes required to encode an instruction is (assume each address requires 24 bits, and 1 byte is required for operation code)
- 9
 - 13
 - 14
 - 12

- Q.19** A CPU has an arithmetic unit that adds bytes and then sets its V, C and Z flag bits as follows. The V-bit is set if arithmetic overflow occurs (in 2's complement arithmetic). The C-bit is set if a carry-out is generated from the most significant bit during an operation. The Z-bit is set if the result is zero. What are the values of the V, C and Z flag bit after 8-bit byte 1100 1100 and 1000 1111 are added?

	V	C	Z
(a)	0	0	0
(b)	1	1	0
(c)	1	1	1
(d)	0	1	0

- Q.20** Match List-I with List-II and select the correct answer using the codes given below the lists:

List-I	List-II
A. MOV X, R ₁	1. Three-address instruction
B. STORE X	2. Zero-address instruction
C. POP X	3. One-address instruction
	4. Two-address instruction

Codes:

	A	B	C
(a)	4	3	2
(b)	3	2	1
(c)	2	3	4
(d)	4	1	1

- Q.21** The register which keeps track of the execution of a program and which contains the memory address of the instruction currently being executed is known as _____

- Index-Register
- Memory address register
- Program counter
- Instruction registers

- Q.22** A certain processor executes the following set of machine instructions sequentially.

```
MOV R0, # 0
MOV R1, 100 (R0)
ADD R1, 200 (R0)
MOV 100 (R0), R1
```

Assuming that memory location 100 contains the value 35 (Hex), and the memory location 200 contains the value A4 (Hex), what could be said about the final result?

- Memory location 100 contains value A4
- Memory location 100 contains value DA
- Memory location 100 contains value D9
- Memory location 200 contains value 35

- Q.23** A PC relative mode branch instruction is 5 B long. The address of the instruction in decimal is 238715. The branch target address if the signed displacement is -32 is _____.

Q.24 Consider the following program segment:

	Instruction	Meaning	Instruction size (in word)
I_1	Load r_0 , 300	$r_0 \leftarrow [300]$	2
I_2	MOV r_1 , 5000	$r_1 \leftarrow \text{Mem}[5000]$	2
I_3	MOV r_2 , (r_1)	$r_2 \leftarrow \text{Mem}[r_1]$	1
I_4	Add r_0 , r_2	$r_0 \leftarrow r_0 + r_2$	1
I_5	MOV, 6000, r_0	$\text{Mem}[6000] \leftarrow r_0$	2
I_6	HALT	Machine Halts	1

Consider that the memory is byte addressable with word size 16 bits and the program has been loaded starting from memory location $(2000)_{10}$. The return address will be saved in the stack, if an interrupt occurs while the CPU has been halted after executing the HALT instruction is ____.

Q.25 Match **List-I** (Addressing Mode) with **List-II** (Location of operand) and select the correct answer:

List-I

- A. Implied
- B. Immediate
- C. Register
- D. Register Indirect

List-II

- 1. Registers which are in CPU
- 2. Register specifies the address of the operand.
- 3. Specified in the address field of an instruction
- 4. Specified implicitly in the definition of instruction

Codes:

- | | A | B | C | D |
|-----|---|---|---|---|
| (a) | 4 | 3 | 1 | 2 |
| (b) | 4 | 1 | 3 | 2 |
| (c) | 4 | 2 | 1 | 3 |
| (d) | 4 | 3 | 2 | 1 |

Q.26 Match **List-I** with **List-II** and select the correct answer using the codes given below the lists:

List-I

- A. To access constant value
- B. Global variable
- C. Pointer
- D. Array

List-II

- 1. Indirect addressing mode
- 2. Direct addressing mode
- 3. Based index addressing mode
- 4. Immediate addressing mode

Codes:

- | | A | B | C | D |
|-----|---|---|---|---|
| (a) | 1 | 2 | 3 | 4 |
| (b) | 2 | 3 | 4 | 1 |
| (c) | 4 | 2 | 1 | 3 |
| (d) | 2 | 3 | 1 | 4 |

Q.27 Consider we have the following values in the given memory locations:

Location	Value
1000	1300
1100	1200
1200	800
1300	1200

Consider that the index register R1 store 200 and is always implicitly used for the indexed addressing mode. What datum is loaded into the accumulator if the instruction is "LOAD 1000" for direct addressing mode, indirect addressing mode and base (indexed) addressing modes respectively.

- | | |
|---------------------|---------------------|
| (a) 1200, 800, 1100 | (b) 1200, 800, 1300 |
| (c) 1100, 1200, 800 | (d) 1300, 1200, 800 |

Q.28 A device with data transfer rate of 20 KBps is connected to a CPU byte wise. Assume byte transfer time between the device and CPU is negligible and interrupt overhead is $20 \mu\text{sec}$. The performance gain under interrupt mode over program controlled mode is _____.

Q.29 Assume that five instructions of the loop body starting memory locations are 1000, 1004, 1008, 1012 and 1016 respectively. Assume each instruction takes 4 bytes of memory. The offset needed to return to the loop are _____.

ANSWERS KEY

- | | | | | |
|------------|-----------|--------------|---------|---------|
| 1. (d) | 2. (d) | 3. (d) | 4. (b) | 5. (b) |
| 6. (c) | 7. (a) | 8. (d) | 9. (b) | 10. (c) |
| 11. (d) | 12. (b) | 13. (b) | 14. (b) | 15. (c) |
| 16. (c) | 17. (b) | 18. (b) | 19. (b) | 20. (d) |
| 21. (b) | 22. (c) | 23. (238688) | | |
| 24. (2016) | 25. (a) | 26. (c) | 27. (d) | |
| 28. (2.5) | 29. (-20) | | | |

HINTS & EXPLANATIONS

1. (d)
CISC: Complex instruction set controller
RISC: Reduced instruction set controller
2. (d)
Considering memory to be byte addressable thus, each cell is of 1B.
Number of cells = 2^{32} (because address bus contains 32 bits)
 $\Rightarrow 2^{32} \text{ B} = 4 \text{ GB}$
3. (d)
System bus is made up of data bus, address bus and control bus.
4. (b)
MAR is used to process address where you want to write.
MDR it will contains the data which you want to write.
Data bus it is the path through which data flows upto memory.
5. (b)
The read/write line belongs to the control bus.
6. (c)
Processor executes set of instructions.
7. (a)
(a) The program counter holds the memory address of the next instruction.
(b) Opcode determinate the control unit.
(c) IR register hold opcode.
(d) Same as (a).
8. (d)
 - RISC machine instruction set is small and of fixed length.
 - No arithmetic or logical instruction can refer to memory directly because it will slowdown its speed. Registers are used to perform arithmetic and logical functions.
 - Large number of user registers available because there is no hindrance of costs. Speed is the main motive.
 - Instructions are easily decoded since logic gates are used.
9. (b)
Word length of a CPU depends on number of data lines (or width of a CPU registers).
10. (c)
 - CISC (Complex Instruction Set computers) as the name says uses complex addressing modes to enhance the performance with limited resources.
 - CISC control units are typically micro-programmed. So that any changes made are easily acceptable.
 - In the CISC, arithmetic or logical instructions may be memory based.
11. (d)
 $i--$ is auto decrement instruction.
Example: store $i \rightarrow R_1$
 $(R_1) -$ will give $i--$
12. (b)

Opcode	Address	Mode
3 bit	12 bit	1 bit

Data memory space = $2^{12} \times 16 \text{ bit}$
 $= 2^{12} \times 2 \text{ B} = 8 \text{ KB}$
13. (b)

(a) $R_4 \leftarrow R_4 + R_3$
Register address in mode semi, operand is directly fetched from registers.

(b) $R_4 \leftarrow R_4 + \textcircled{3}$
 \Downarrow
 Immediate value
 Immediate addressing modes.

(c) $R_4 \leftarrow R_4 + \text{Mem}[\text{Regs}[R_1]]$ register indirect
14. (b)
The code in this mode is easy to relocate in the memory by using offset value.
15. (c)
 - k -bit opcode means 2^k operations.
 - n -bit address means 2^n addressable memory cells.
16. (c)
Memory address register is used to hold the address from which or to which data are to be transferred.