CHAPTER 2: CENTRAL PROCESSING UNIT (CPU)

2.1 Introduction to CPU

The **Central Processing Unit (CPU)** is the most important component of a computer system. It is responsible for:

- Executing instructions
- Processing data
- Controlling all operations of the computer

Standard Definition

The CPU is a hardware device that processes data and converts it into meaningful information.

2.2 Why CPU is Called the Brain of Computer

The CPU is called the **brain of the computer** because:

1. It controls all operations
2. It performs calculations
3. It makes decisions based on instructions
4. It coordinates between all components

Conceptual Comparison

Human Brain	CPU
Takes decisions	Executes instructions
Controls body	Controls computer
Processes thoughts	Processes data

2.3 Processor and Microprocessor

Processor

A processor is a device that:

- Takes input
- Processes it
- Produces output

Microprocessor

A **microprocessor** is:

- A single chip containing the entire CPU
- Capable of performing arithmetic and logical operations

Key Difference

Processor	Microprocessor
General term	Specific implementation
Can be multi-chip	Single chip
Larger system	Compact integrated circuit

2.4 Examples of Processors

Modern processors include:

- Intel Core i3, i5, i7, i9
- Intel 386, 486
- Celeron
- DEC Alpha
- IBM RS6000

Mobile processors:

- Snapdragon
- Exynos
- Helio
- Tegra

2.5 Functions of CPU The CPU performs several important functions:

1. Data Collection (Input Handling)

- Receives data from input devices
- Example: Keyboard input

2. Instruction Handling

- Receives instructions from user or program
- Decides what operation to perform

3. Data Processing

- Performs calculations
- Arranges data in proper sequence

4. Conversion of Data into Information

- Converts raw data into meaningful results

2.6 Processing – Core Concept

Processing means:
Manipulating data to produce meaningful information.

Types of Processing Operations

Operation	Example
Arithmetic	Addition, subtraction
Logical	Comparison (>, <, =)
Data movement	Transfer data

2.7 Speed of CPU

CPU speed determines how fast a computer works.

Measurement Units

Unit	Meaning
Hertz (Hz)	Cycles per second
Megahertz (MHz)	Million cycles/sec
Gigahertz (GHz)	Billion cycles/sec
FLOPS	Floating point operations/sec

Concept Insight

Higher speed means:

- Faster processing
- Better performance

2.8 Word Length (Very Important Concept)

Word length is the number of bits that a processor can process at one time.

Examples

Word Size	Meaning
8-bit	Processes 8 bits at a time
16-bit	Processes 16 bits
32-bit	Faster processing
64-bit	High-performance computing

Concept Insight

- Larger word size → Faster processing
- More data handled at once

2.9 CPU and Memory Relationship

The CPU continuously interacts with memory:

- Fetches instructions from memory
- Stores results back into memory

Important Point

Primary memory stores:

- Currently executing programs
- Active data

2.10 Basic Working of CPU

The CPU works in a continuous cycle:

1. Takes input
2. Processes data
3. Produces output

This is aligned with:
Input → Process → Output cycle

2.11 Internal Components of CPU (Overview)

The CPU consists of several internal components:

- Arithmetic Logic Unit (ALU)
- Control Unit (CU)
- Registers
- Cache Memory
- Buses
- Clock

These components work together to execute instructions efficiently.

2.12 Conceptual Flow of CPU Operation

The working of CPU can be summarized as:

Input → CPU Processing → Output

Internally:

Instruction → Decode → Execute → Result

2.13 Important Exam-Oriented Points

1. CPU is the main processing unit of computer
2. CPU is also called:
    - Brain of computer
    - Microchip
3. CPU performs:

- Arithmetic operations
- Logical operations

4. CPU speed is measured in:
    - MHz and GHz

2.14 Memory-Based Conceptual Understanding

To deeply understand CPU:

- CPU does not work alone
- It always works with memory
- Memory provides:
    - Data
    - Instructions

2.15 Introduction to Internal Structure of CPU

The CPU is not a single simple unit. Internally, it is composed of multiple specialized components that work together to execute instructions.

The major internal components of CPU are:

- Arithmetic Logic Unit (ALU)
- Control Unit (CU)
- Registers
- Cache Memory
- Buses
- Clock

Each component has a specific role, and the coordination among them ensures efficient processing.

2.16 Arithmetic Logic Unit (ALU)

Definition

The **Arithmetic Logic Unit (ALU)** is the part of the CPU responsible for performing all mathematical and logical operations.

Functions of ALU

The ALU performs two types of operations:

1. Arithmetic Operations

These include:

- Addition
- Subtraction
- Multiplication
- Division

2. Logical Operations

These include:

- Comparison (>, <, =)
- Decision making
- Boolean operations (AND, OR, NOT)

Example

If a program asks:
"Is 50 greater than 30?"

- ALU performs comparison
- Result = TRUE

Alternate Name

ALU is also known as:

- Integer Unit (IU)

Deep Concept Insight

The ALU does not store data permanently. It:

- Takes data from registers
- Performs operations
- Sends results back to registers

Key Points for Exams

- ALU performs all calculations
- ALU handles both arithmetic and logic
- ALU works closely with registers

2.17 Control Unit (CU)

Definition

The **Control Unit** is responsible for directing and coordinating all operations of the computer system.

Core Function

The Control Unit acts like a **manager**:

- It controls execution of instructions
- It tells other components what to do
- It ensures proper sequence of operations

Functions of Control Unit

Function	Explanation
Instruction Control	Decides which instruction to execute
Coordination	Coordinates ALU, memory, and I/O
Signal Generation	Sends control signals
Data Movement	Transfers data between units

Working Mechanism (Step-by-Step)

1. Receives instruction from memory
2. Decodes the instruction
3. Sends signals to ALU and other components
4. Controls execution

Important Characteristics

- Does not perform calculations
- Controls the entire system
- Known as **Nerve Center of Computer**

Deep Concept Insight

The Control Unit ensures:

- Correct timing
- Proper sequence
- Synchronization of operations

Key Points

- CU controls operations
- CU directs data flow
- CU does not process data

2.18 Registers

Registers are small, high-speed storage locations inside the CPU used to store data temporarily.

Characteristics of Registers

- Located inside CPU
- Very fast access
- Store data in bits
- Temporary storage

Why Registers are Important

Registers:

- Reduce processing time
- Provide immediate data access
- Improve CPU performance

Types of Registers

1. Memory Address Register (MAR)

- Stores address of memory location
- Used to access data from memory

2. Instruction Register (IR)

- Holds current instruction being executed

3. Program Counter (PC)

- Stores address of next instruction

4. Accumulator

- Stores intermediate results of calculations

Example

Program Counter → Gives next instruction address

1. Instruction Register → Holds instruction
2. ALU processes data
3. Accumulator stores result

Insight

Registers are like:

- “Working tables” of CPU
- Where immediate operations happen

Important Properties

- Writing new data replaces old data
- Used in micro-operations
- Integral part of CPU

2.19 Cache Memory (CPU Internal Memory)

Definition

Cache memory is a small, high-speed memory located inside or near the CPU used to store frequently accessed data.

Purpose of Cache Memory

- Reduces access time
- Improves performance
- Stores frequently used instructions

Working Concept

Instead of accessing slow main memory:

- CPU checks cache first
- If data is found → Fast access
- If not → Fetch from RAM

Conceptual Example

Opening same application repeatedly:

- First time → Slow
- Next time → Faster (due to cache)

Key Points

- Cache is faster than RAM
- Located inside CPU
- Temporary storage

2.20 Interconnection Between Components

All components of CPU work together:

Component	Role
ALU	Performs operations
CU	Controls operations
Registers	Store data temporarily
Cache	Provides fast access

2.21 Internal Data Flow in CPU

The data flow inside CPU is:

Memory → Registers → ALU → Registers → Memory

This continuous movement ensures efficient processing.

2.22 Introduction to CPU Communication System

Inside a computer, different components such as CPU, memory, and input/output devices must communicate with each other.

This communication is made possible through **Buses**.

Without buses, no data transfer or instruction execution is possible.

2.23 What is a Bus?

Definition

A **bus** is a communication pathway or channel that transfers data, instructions, and control signals between different components of a computer system.

2.24 Types of Buses (Very Important)

There are three main types of buses:

1. Data Bus
2. Address Bus
3. Control Bus

1. Data Bus

Function

- Transfers actual data between CPU, memory, and devices

Characteristics

- Bidirectional (data can move both ways)
- Carries instructions and results

Example

When CPU reads data from memory:

- Data travels through Data Bus

2. Address Bus

Function

- Carries address of memory location

Characteristics

- Unidirectional (CPU → Memory)
- Specifies where data should go or come from

Example

CPU sends address:
“Fetch data from location 2000”

3. Control Bus

Function

- Carries control signals

Examples of Signals

- Read signal
- Write signal
- Interrupt signal

Role

Coordinates activities of all components

2.25 Comparison of Buses

Bus Type	Function	Direction
Data Bus	Transfers data	Bidirectional
Address Bus	Carries address	Unidirectional
Control Bus	Sends signals	Both directions

2.26 Functions of Buses (Deep Understanding)

Buses perform several essential functions:

1. Addressing

- Identifies memory location

2. Data Transfer

- Transfers data between components

3. Control and Coordination

- Sends control signals

4. Timing and Synchronization

- Maintains proper timing between devices

5. Power Supply (in some cases)

- Supplies power to connected components

2.27 Types of Advanced Buses

1. System Bus

- Connects CPU, memory, and I/O devices

2. Processor Bus (Front Side Bus)

- Connects CPU to main memory

3. Peripheral Bus

- Connects external devices like printer, hard disk

4. External Bus

- Used for expansion devices

2.28 CPU Clock (Timing Mechanism)

Definition

The CPU clock is an electronic circuit that generates pulses to synchronize all operations of the CPU.

2.29 Clock Speed

Definition

Clock speed is the number of cycles executed by CPU per second.

Units of Measurement

Unit	Meaning
Hz	Cycles per second
MHz	Million cycles/sec
GHz	Billion cycles/sec

Conceptual Understanding

Higher clock speed means:

- Faster instruction execution
- Better system performance

Example

Clock Speed	Performance
1 GHz	Slower
3 GHz	Faster

2.30 Instruction Cycle (Core of CPU Working)

The CPU performs all tasks using a cycle called the **Instruction Cycle**.

Definition

Instruction cycle is the process by which CPU fetches, decodes, and executes instructions.

2.31 Stages of Instruction Cycle

There are four main stages:

1. Fetch
2. Decode
3. Execute
4. Write Back

1. Fetch Stage

- CPU retrieves instruction from memory
- Instruction is stored in Instruction Register

2. Decode Stage

- Control Unit interprets instruction
- Determines required operation

3. Execute Stage

- ALU performs the operation
- Processing is completed

4. Write Back Stage

- Result is stored in memory or register

2.32 Instruction Cycle Flow

Instruction Cycle follows this sequence:

Fetch → Decode → Execute → Store

This cycle repeats continuously.

2.33 Real-Life Example of Instruction Cycle

Task: Add two numbers

1. Fetch → Get instruction "ADD A + B"
2. Decode → Understand addition operation
3. Execute → ALU performs addition
4. Write Back → Store result

2.34 Relationship Between CPU Components in Instruction Cycle

Stage	Component Used
Fetch	Memory + Registers
Decode	Control Unit
Execute	ALU
Store	Registers/Memory

2.35 Micro-Operations (Advanced Concept)

Micro-operations are small internal operations performed inside CPU during instruction execution.

Examples:

- Data transfer
- Arithmetic operations
- Logical operations

2.36 Instruction Set (Important Concept)

Definition

Instruction set is the collection of all instructions that a CPU can execute.

Concept Insight

Each CPU has its own instruction set, which defines:

- What operations it can perform
- How it processes data

2.37 Overall Working of CPU (Integrated View)

The complete working of CPU involves:

1. Instruction is fetched from memory
2. Control Unit decodes it
3. ALU executes it
4. Result is stored
5. Buses transfer data
6. Clock synchronizes operations

2.38 Introduction to Registers (Deep Internal View)

Registers are one of the most critical components of the CPU. While memory stores large amounts of data, registers provide **ultra-fast access to small pieces of data** required immediately during processing.

Definition

Registers are high-speed storage locations inside the CPU used to temporarily hold data, instructions, and addresses during execution.

2.39 Why Registers are Required

During execution, CPU needs data instantly. Accessing main memory repeatedly is slow. Registers solve this problem by:

- Providing fastest data access
- Reducing execution time
- Supporting real-time processing

2.40 Characteristics of Registers

Property	Description
Location	Inside CPU
Speed	Extremely fast
Size	Very small
Storage Type	Temporary
Data Format	Binary (bits)

2.41 Working Role of Registers in CPU

Registers act as:

- Temporary data holders
- Instruction holders
- Address trackers

Conceptual Flow

Memory → Register → ALU → Register → Memory

2.42 Important Types of Registers (Detailed Explanation)

1. Memory Address Register (MAR)

Function

- Stores the address of memory location to be accessed

Role

- Helps CPU locate data in memory

Example

If CPU needs data from address 500:

- MAR stores "500"

2. Instruction Register (IR)

Function

- Stores the current instruction being executed

Role

- Provides instruction to Control Unit

3. Program Counter (PC)

Function

- Stores address of next instruction

Role

- Maintains sequence of execution

Deep Concept

Program Counter ensures:

- Program runs in correct order
- No instruction is skipped

4. Accumulator Register

Function

- Stores intermediate results of ALU operations

Example

During addition:

- Result is stored in accumulator

5. Memory Buffer Register (MBR) / Data Register

Function

- Stores data transferred between memory and CPU

6. Index Register

Function

- Used for address calculation
- Helps in array processing

7. Status Register / Flag Register

Function

- Stores status of operations

Example Flags

- Zero flag
- Carry flag
- Sign flag

8. Program Status Register (PSR)

Function

- Holds information about current execution state

9. Floating Point Register

Function

- Stores decimal numbers
- Used in scientific calculations

10. Vector Register

Function

- Used in vector processing
- Handles multiple data at once

2.43 Summary Table of Registers

Register	Function
MAR	Stores memory address
IR	Holds instruction
PC	Stores next instruction address

Accumulator	Stores result
MBR	Transfers data
Index Register	Address calculation
Flag Register	Stores status
PSR	Execution state

2.44 Register Operations (Micro-Operations)

Micro-operations are small operations performed on data stored in registers.

Types of Micro-Operations

Type	Description
Data Transfer	Moving data between registers
Arithmetic	Addition, subtraction
Logical	AND, OR, NOT
Shift	Left or right shift

Example

If two registers contain values:

- R1 = 5
- R2 = 3

Operation:
R1 = R1 + R2

This is a micro-operation.

2.45 Data Handling Inside CPU

Data flows inside CPU in a structured way:

1. Data is fetched from memory
2. Stored in registers
3. Sent to ALU for processing
4. Result returned to registers
5. Stored back in memory

2.46 Role of Registers in Instruction Cycle

Registers play a key role in each stage:

Stage	Register Used
Fetch	MAR, PC
Decode	IR
Execute	Accumulator
Store	MBR

2.47 Word Size and Registers Relationship

Word size is directly related to register size.

Concept

- Register size = Word size
- Larger register → More data processed

Example

Word Size	Capability
8-bit	Basic operations
32-bit	Moderate processing
64-bit	High performance

2.48 Important Properties of Registers

1. Registers are fastest memory units
2. They store data temporarily
3. They are part of CPU
4. Writing new data replaces old data
5. Used in instruction execution

2.49 Conceptual Visualization of Register Usage

Instruction Execution:

- PC → Next instruction
- MAR → Address location
- IR → Instruction
- ALU → Process
- Accumulator → Result

2.51 Introduction to CPU Memory System

The CPU does not work alone. It continuously interacts with different types of memory. However, not all memory is equally fast.

To solve this problem, computers use a **memory hierarchy system**, where:

- Faster memory is smaller and closer to CPU
- Slower memory is larger and farther from CPU

2.52 Need for Cache Memory

Problem

- CPU is very fast
- Main memory (RAM) is comparatively slower

This creates a **speed gap**.

Solution

Cache memory is introduced to:

- Reduce this speed gap
- Improve performance

2.53 What is Cache Memory?

Cache memory is a small, high-speed memory located inside or very close to the CPU that stores frequently used data and instructions.

2.54 Key Characteristics of Cache Memory

Property	Description
Speed	Very high
Size	Small
Location	Inside or near CPU
Cost	Expensive
Purpose	Faster access

2.55 Working Principle of Cache Memory

Cache memory works on the principle of:

Locality of Reference

Types of Locality

1. Temporal Locality
   - Recently used data is likely to be used again
2. Spatial Locality
   - Nearby data is likely to be used

Step-by-Step Working

1. CPU requests data
2. Cache is checked first
3. If data is found → Cache Hit
4. If not found → Cache Miss → Data fetched from RAM

2.56 Cache Hit and Cache Miss

Term	Meaning
Cache Hit	Data found in cache
Cache Miss	Data not found in cache

Concept Insight

Higher cache hit rate means:

- Better performance
- Faster execution

2.57 Levels of Cache Memory

Cache memory is divided into levels:

Level	Location	Speed	Size
L1 Cache	Inside CPU core	Fastest	Smallest
L2 Cache	Inside CPU	Fast	Larger
L3 Cache	Shared among cores	Moderate	Largest

Concept Understanding

- L1 is closest to CPU
- L3 is shared among processors

2.58 Cache vs RAM vs Secondary Memory

Feature	Cache	RAM	Secondary Storage
Speed	Very Fast	Fast	Slow
Size	Small	Medium	Large
Cost	High	Moderate	Low
Location	CPU	Motherboard	External/Internal

2.59 Memory Hierarchy (Very Important Concept)

Memory hierarchy organizes memory in levels based on speed and cost.

Hierarchy Structure

Registers → Cache → RAM → Secondary Storage

Detailed Explanation

Level	Memory Type	Speed
1	Registers	Fastest
2	Cache	Very Fast
3	RAM	Fast
4	Hard Disk	Slow

Concept Insight

- Closer to CPU → Faster
- Farther from CPU → Slower

2.60 Performance Optimization Using Cache

Cache improves performance by:

1. Reducing memory access time
2. Minimizing CPU waiting time
3. Increasing execution speed

Example

Opening a program:

- First time → Slow (RAM access)
- Next time → Fast (Cache access)

2.61 Cache Mapping Techniques (Advanced Concept)

Cache stores data using different techniques:

1. Direct Mapping

- Each memory block maps to one cache location

2. Associative Mapping

- Data can be placed anywhere in cache

3. Set-Associative Mapping

- Combination of both methods

Concept Insight

Technique	Speed	Complexity
Direct	Fast	Simple
Associative	Flexible	Complex
Set-Associative	Balanced	Moderate

2.62 Write Policies in Cache

1. Write Through

- Data written to cache and memory simultaneously

2. Write Back

- Data written to cache first, then memory later

Concept Insight

- Write Through → Safe but slower
- Write Back → Faster but complex

2.63 Role of Cache in Multicore Systems

In multicore processors:

- Each core may have its own cache
- Some cache levels are shared

Importance

- Reduces data duplication
- Improves parallel processing

2.64 Relationship Between CPU, Cache and RAM

The interaction is:

CPU → Cache → RAM → Storage

Data Flow

1. CPU checks cache
2. If not found → checks RAM
3. If still not found → checks storage

2.65 Important Exam-Oriented Points

1. Cache is fastest after registers
2. Cache is located inside CPU
3. Cache reduces access time
4. Cache works on locality principle
5. L1 cache is fastest

2.66 Conceptual Summary of Memory System

- Registers store immediate data
- Cache stores frequently used data
- RAM stores active programsUnderstood

2.67 Introduction to Instruction Set

Every CPU operates based on a predefined set of instructions.

An **Instruction Set** is the complete collection of commands that a CPU can understand and execute.

2.68 Instruction Set Architecture (ISA)

Definition

Instruction Set Architecture (ISA) is the design and structure of a CPU that defines:

- Types of instructions
- Data formats
- Addressing modes
- Execution methods

Concept Insight

ISA acts as a bridge between:

- Hardware (CPU)
- Software (Programs)

2.69 Types of Instruction Set Architectures

There are mainly three types:

1. CISC
2. RISC
3. EPIC

2.70 CISC (Complex Instruction Set Computer)

CISC is a type of processor design that uses a large number of complex instructions.

Characteristics

- Large instruction set
- Instructions can perform multiple operations
- Complex hardware design
- Fewer lines of code required

Example of Operation

Single instruction:
ADD A, B, C

This performs multiple steps internally.

Advantages

- Reduced number of instructions
- Efficient use of memory

Disadvantages

- Slower execution
- Complex design

2.71 RISC (Reduced Instruction Set Computer)

Definition

RISC is a processor design that uses a small set of simple instructions.

Developed by IBM researcher **John Cocke**.

Characteristics

- Small instruction set
- Each instruction performs one task
- Faster execution
- Simple hardware design

Example

Instead of one complex instruction:

- Load
- Add
- Store

Advantages

- High speed
- Efficient processing
- Better pipeline performance

Disadvantages

- Requires more instructions
- Larger program size

2.72 EPIC (Explicitly Parallel Instruction Computing)

Definition

EPIC is a processor design that allows multiple instructions to be executed simultaneously.

Key Concept

- Compiler identifies parallel instructions
- CPU executes them together

Importance

- Improves performance
- Enables parallel processing

2.73 Comparison of CISC, RISC and EPIC

Feature	CISC	RISC	EPIC
Instruction Set	Large	Small	Moderate
Execution	Complex	Simple	Parallel
Speed	Moderate	Fast	Very Fast
Hardware	Complex	Simple	Advanced

2.74 Multicore Processors (Modern CPU Design)

Definition

A multicore processor contains multiple processing units (cores) within a single chip.

2.75 Types of Multicore Processors

Type	Number of Cores
Single Core	1
Dual Core	2
Quad Core	4
Hexa Core	6
Octa Core	8
Deca Core	10

2.76 Working of Multicore Processors

Each core can:

- Execute instructions independently
- Perform tasks simultaneously

Example

Opening multiple applications:

- Each core handles a separate task

2.77 Advantages of Multicore Processors

1. Parallel processing
2. Better multitasking
3. Reduced power consumption
4. Less heat generation

2.78 Concept of Parallel Processing

Parallel processing means:

- Multiple tasks executed at the same time

Example

Video editing:

- One core handles video
- Another handles audio

2.79 Power Efficient Processors

Modern processors are designed to:

- Consume less power
- Generate less heat
- Improve battery life

Concept Insight

Efficiency is achieved through:

- Multicore design
- Advanced architecture

2.80 Graphics Processing Unit (GPU)

GPU is a specialized electronic circuit designed to perform complex graphical and mathematical operations rapidly.

2.81 Functions of GPU

- Image rendering
- Video processing
- Gaming graphics
- Parallel computations

2.82 CPU vs GPU

Feature	CPU	GPU
Purpose	General processing	Graphics processing
Cores	Few	Many
Speed	Sequential tasks	Parallel tasks
Usage	All applications	Graphics-intensive tasks

2.83 Integration of CPU and GPU

Modern systems use:

- CPU for general tasks
- GPU for heavy computations

2.84 Modern Processor Trends

Modern CPUs focus on:

1. Multicore architecture
2. Parallel processing
3. Energy efficiency
4. High-speed execution

2.86 Introduction to Advanced Communication in Computer System

In modern computer systems, communication is not limited to CPU and memory only. A large number of external devices and network components are also connected.

To manage this complex communication, advanced bus systems and interface devices are used.

2.87 System Bus (Integrated Communication System)

Definition

A **system bus** is a single communication pathway that connects the major components of a computer system such as CPU, memory, and input/output devices.

Components of System Bus

System bus consists of:

- Data Bus
- Address Bus
- Control Bus

Conceptual Understanding

System Bus = Combination of all buses

2.88 Processor Bus (Front Side Bus)

Definition

The processor bus, also known as the **Front Side Bus (FSB)**, connects the CPU with the main memory.

Functions

- Transfers data between CPU and RAM
- Supports high-speed communication

Concept Insight

Higher FSB speed results in:

- Faster data transfer
- Improved performance

2.89 Peripheral Bus

Definition

A **peripheral bus** is used to connect external devices such as:

- Hard disk
- Printer
- Scanner

Importance

- Enables communication with external hardware
- Expands system capability

2.90 External Bus (Expansion Bus)

Definition

External bus is used to connect additional hardware components to the computer system.

Examples

- USB devices
- External storage
- Expansion cards

Concept Insight

Expansion bus allows:

- System upgradation
- Addition of new features

2.91 Direct Memory Access (DMA) (Advanced Concept)

Definition

DMA allows devices to transfer data directly to memory without involving CPU.

Importance

- Reduces CPU workload
- Improves performance

Example

While copying files:

- Data transfer occurs without CPU interruption

2.92 Network Interface Card (NIC)

A **Network Interface Card (NIC)** is a hardware component that connects a computer to a network.

Features

- Provides network connectivity
- Has a unique address called MAC address
- Enables communication between devices

Key Concept

Each NIC has:

- Unique physical address
- Used for identification in network

Example

When connecting to internet:

- NIC acts as interface between computer and network

2.93 Wireless Access Point (WAP)

Definition

A **Wireless Access Point (WAP)** is a device that allows wireless devices to connect to a wired network.

Working Mechanism

- Acts as transmitter and receiver
- Connects Wi-Fi devices to LAN

Example

Wi-Fi router in home:

- Functions as WAP

2.94 Workstation (Specialized Computer System)

Definition

A workstation is a high-performance computer designed for technical and scientific applications.

Characteristics

- Designed for single user
- High processing power
- Used for specialized tasks

Applications

- Engineering design
- Scientific research
- Data analysis

2.95 Peripheral Devices and Their Role

Peripheral devices are external devices connected to computer.

Types of Peripheral Devices

Type	Examples
Input Devices	Keyboard, Mouse
Output Devices	Monitor, Printer
Storage Devices	Hard disk, USB

Role of Peripherals

- Input data
- Display output
- Store information

2.96 Communication Flow in Computer System

The complete communication system works as:

Input Device → Bus → CPU → Bus → Output Device

2.97 Synchronization Using Bus and Clock

All components must work in coordination.

This is ensured by:

- Bus (communication)
- Clock (timing)

2.98 Integration of CPU with External World

CPU communicates with:

- Memory
- Input/Output devices
- Network systems

Through:

- Buses
- Interface device

2.99 Practical Example of System Working

When a user prints a document:

1. CPU processes data
2. Sends data through bus
3. Printer receives data
4. Output is printed

2.100 Important Exam-Oriented Points

1. System bus connects all major components
2. NIC connects computer to network
3. WAP provides wireless connectivity
4. Peripheral bus connects external devices
5. DMA reduces CPU workload

2.101 Complete Concept of CPU (Integrated Understanding)

After studying all components, it is important to understand the CPU as a **complete system** rather than separate parts.

Integrated Definition

The CPU is an electronic unit that:

- Receives data and instructions
- Processes them using internal components
- Produces output
- Controls all operations of the computer

2.102 Complete Working of CPU (Step-by-Step)

The working of CPU can be understood in a complete cycle:

Step 1: Input Reception

- Data and instructions are given through input devices

Step 2: Instruction Fetching

- CPU fetches instruction from memory
- Stored in Instruction Register

Step 3: Instruction Decoding

- Control Unit interprets instruction

Step 4: Execution

- ALU performs required operation

Step 5: Result Storage

- Result stored in registers or memory

Step 6: Output Generation

- Result displayed through output devices

2.103 Internal Coordination of CPU Components

All components of CPU work together:

Component	Role
Control Unit	Directs operations
ALU	Performs calculations
Registers	Store temporary data
Cache	Provides fast access
Buses	Transfer data
Clock	Synchronizes operations

2.104 Flow of Data Inside CPU

The complete data flow can be represented as:

Memory → Registers → ALU → Registers → Memory → Output

This continuous flow ensures smooth processing.

2.105 CPU and Instruction Cycle (Final View)

CPU continuously follows the instruction cycle:

Fetch → Decode → Execute → Store

Concept Insight

This cycle runs millions of times per second.

2.106 Role of CPU in Overall Computer System

CPU acts as:

1. Controller → Controls all devices
2. Processor → Processes data
3. Coordinator → Connects all components