

Computer Fundamentals and Applications

Unit 1: Introduction to Computer System

(16 Hours)

Course Code: CACS101 | Credit: 4 Cr. | Year/Semester: I / I

Class Load: 8 Hrs./Week (Theory: 4 Hrs., Practical: 4 Hrs.)

Contents

1. Introduction to Computer
2. Characteristics of Computer
3. Applications of Computer
4. Classification of Computer
5. Mobile Computing
6. Anatomy of a Digital Computer
7. Computer Architecture
8. Memory & Its Classification
9. Input Devices
10. Output Devices
11. Interfaces

Key Takeaways

A computer is an electronic device that accepts data as input, processes that data according to a set of stored instructions, and produces information as output, while also being able to store data and instructions for later use. The word itself originally described a person who performed calculations, but since the mid-twentieth century it has referred to the programmable electronic machines that now underpin nearly every aspect of modern life, from communication and commerce to scientific research and entertainment.

Every computer, regardless of size or purpose, operates on the same basic cycle: input, process, output, and storage. Data enters the system through an input device, is manipulated by the processor according to program instructions, is presented to the user through an output device, and can be retained in storage for future use. This four-stage cycle is often summarised as the IPOS cycle and forms the conceptual foundation on which every subsequent topic in this unit is built.

Modern computing traces its development through five broad generations, each defined by the switching technology used to build the processor: vacuum tubes, transistors, integrated circuits, microprocessors, and, in the current and emerging fifth generation, architectures built around artificial intelligence and parallel processing. Each generation delivered dramatic gains in speed, reliability, and affordability while shrinking the physical size of the machine.

Generations of Computers

Generation	Period	Key Technology
First	1940 - 1956	Vacuum tubes; machine language programming
Second	1956 - 1963	Transistors; assembly language, smaller and faster
Third	1964 - 1971	Integrated circuits; keyboards and monitors introduced

Generation	Period	Key Technology
Fourth	1971 - present	Microprocessors; personal computers, GUIs, internet
Fifth	Present - emerging	AI-driven and parallel-processing architectures

Table 1.1 — The five generations of computer development.

Example 1.1 — The IPOS Cycle in Everyday Use

A student typing an assignment provides input through a keyboard; the processor formats and checks the text according to the word processor's instructions; the formatted document is shown as output on the screen and later sent to a printer.

The finished file is saved to the hard disk for storage, illustrating all four stages of the input-process-output-storage cycle within a single, ordinary task.

Computers are valued not because they can think creatively, but because they perform certain classes of task far better than humans can. Their usefulness rests on a small set of defining characteristics: speed, accuracy, diligence, storage capacity, versatility, and automation. Together these traits explain why computers have become indispensable in fields ranging from scientific research to everyday office work.

Speed refers to the extraordinary rate at which a computer can execute instructions, commonly measured in millions or billions of operations per second. Accuracy means that, barring a fault in hardware or an error in the program itself, a computer produces the same correct result every time it repeats a calculation. Diligence describes the computer's immunity to fatigue or boredom: it can repeat the same operation millions of times without any drop in performance, unlike a human operator.

Versatility allows a single computer to perform an enormous variety of unrelated tasks, from payroll processing to video editing, simply by running different software. Automation allows a computer to carry out an entire sequence of operations without human intervention once a program has been started, and its large storage capacity allows it to retain vast quantities of data and instructions for immediate or future use.

Defining Characteristics of a Computer

Characteristic	Description
Speed	Executes millions to billions of instructions per second
Accuracy	Produces error-free results when hardware and software are correct
Diligence	Performs repetitive tasks indefinitely without fatigue or boredom
Storage	Retains vast amounts of data and instructions for later use
Versatility	Performs a wide range of tasks by running different programs

Characteristic	Description
Automation	Executes a full sequence of steps without ongoing human input

Table 2.1 — The six characteristics most often used to define a computer.

Example 2.1 — Diligence and Accuracy in Payroll Processing

A company with five thousand employees must calculate monthly salaries, tax deductions, and bonuses. Performed manually, this task would be slow and prone to arithmetic error as clerks tire toward the end of the run.

A computer running payroll software completes the same calculations for every employee in seconds, applying identical rules with no loss of accuracy regardless of how many records remain, demonstrating both speed and diligence in a single real-world task.

Because computers are versatile general-purpose machines, they have been adopted across virtually every sector of activity. In education, computers support digital learning platforms, research, and administration; in business, they drive accounting, inventory management, and communication; and in science, they enable simulation and analysis of problems too large or too complex for manual calculation.

Beyond these traditional domains, computers have become central to entertainment, through streaming, gaming, and digital media production; to healthcare, through diagnostic imaging, patient record systems, and robotic surgery; and to government, through e-governance platforms, national databases, and public service delivery. The common thread across every domain is the same IPOS cycle described earlier, applied to domain-specific data and problems.

Representative Applications by Sector

Sector	Typical Application
Education	Digital learning platforms, research computation, student records
Business & Commerce	Accounting, inventory management, e-commerce transactions
Healthcare	Diagnostic imaging, electronic health records, robotic surgery
Science & Engineering	Simulation, modelling, and data analysis
Entertainment	Streaming media, video games, digital content production
Government	E-governance portals, national databases, public records
Banking & Finance	Online banking, ATM networks, fraud detection
Transportation	Traffic control systems, GPS navigation, airline reservations

Table 3.1 — Representative applications of computers across major sectors.

Example 3.1 — Computers in Healthcare

A hospital uses a computerized system to store every patient's medical history, prescribed medication, and laboratory results in a single electronic health record.

When the patient is later admitted to the emergency department, doctors instantly retrieve this record instead of waiting for paper files, allowing faster and safer treatment decisions.

bcalink.com

Computers can be classified along several independent dimensions. By size and processing power, they range from supercomputers, the fastest and most powerful machines used for tasks such as weather modelling and nuclear simulation, down through mainframes, used by large organizations for bulk data processing, to minicomputers and the microcomputers, or personal computers, familiar from everyday use.

By purpose, computers are classified as general-purpose, capable of running many different kinds of software, or special-purpose, designed and built to perform one specific task, such as the embedded controller inside a washing machine or a traffic signal. By the type of data they process, computers are classified as digital, which operate on discrete binary values; analog, which operate on continuous physical quantities such as voltage or temperature; and hybrid, which combine both approaches, commonly found in specialized scientific and medical instruments.

Classification by Size and Processing Power

Type	Description	Typical Example
Supercomputer	Extremely fast, massively parallel systems for intensive computation	Weather forecasting, nuclear research
Mainframe	Large, reliable systems for high-volume centralized processing	Banking transaction processing
Minicomputer	Mid-sized systems bridging mainframes and microcomputers	Departmental servers (largely historical)
Microcomputer	Compact, affordable systems built around a single microprocessor	Desktop PCs, laptops, tablets

Type	Description	Typical Example
Embedded Computer	Special-purpose processors built into another device	Washing machines, traffic signals

Table 4.1 — Classification of computers by size and processing capability.

Example 4.1 — Choosing the Right Class of Computer

A national meteorological agency needs to model global weather patterns involving billions of calculations per second; this workload calls for a supercomputer.

A household appliance manufacturer, by contrast, needs only a tiny, inexpensive embedded computer to control a washing machine's wash and spin cycles, illustrating how the appropriate classification depends entirely on the task at hand.

Mobile computing refers to the use of portable computing devices, such as smartphones, tablets, laptops, and wearables, that allow users to access information and applications while moving from place to place. It combines three elements working together: mobile hardware that is small and battery-powered, mobile software adapted to smaller screens and touch input, and wireless communication that keeps the device connected without a physical cable.

The defining characteristics of mobile computing are portability, which allows the device to be carried and used anywhere; connectivity, which keeps the device linked to networks and services through technologies such as Wi-Fi, cellular data, and Bluetooth; and social interactivity, which enables real-time communication and data sharing between users. Location awareness, typically provided by GPS, has become an increasingly important fourth characteristic, enabling navigation and location-based services.

Technologies Enabling Mobile Computing

Technology	Role in Mobile Computing
Cellular Networks (4G/5G)	Provide wide-area wireless data connectivity while moving
Wi-Fi	Provides high-speed local wireless connectivity indoors or in hotspots
Bluetooth	Connects nearby peripherals such as headsets and wearables
GPS	Determines device location for navigation and location-based services
Cloud Storage & Sync	Keeps data available and consistent across multiple devices

Table 5.1 — Key technologies that make mobile computing possible.

Example 5.1 — Mobile Computing in Field Sales

A field sales representative visits clients throughout the day and uses a tablet with a mobile CRM application to check inventory, place orders, and process payments on the spot.

Cellular connectivity keeps the tablet synchronized with the company's central database in real time, so orders placed in the field are immediately visible to the warehouse without the representative ever returning to an office.

bcalink.com

Despite enormous differences in size and purpose, every digital computer is built from the same functional building blocks: an input unit, a central processing unit, a memory unit, and an output unit, all connected by pathways called buses along which data and instructions travel. The central processing unit, or CPU, is further divided into the control unit, which directs the operation of every other component, and the arithmetic logic unit, which performs calculations and logical comparisons.

Small, extremely fast storage locations called registers sit inside the CPU itself and hold the data and instructions currently being processed, since retrieving data from registers is far faster than retrieving it from main memory. The memory unit stores both the program instructions and the data they operate on, while the buses, comprising the data bus, address bus, and control bus, carry information between the CPU, memory, and input/output devices.

Core Functional Units of a Digital Computer

Unit	Function
Input Unit	Accepts data and instructions from the outside world
Control Unit (CU)	Directs and coordinates the activity of all other components
Arithmetic Logic Unit (ALU)	Performs arithmetic calculations and logical comparisons
Registers	Hold data and instructions currently in use by the CPU
Memory Unit	Stores programs and data for immediate or later use
Output Unit	Presents processed information to the user
Buses	Carry data, addresses, and control signals between components

Table 6.1 — The core functional units found in every digital computer.

Example 6.1 — Tracing a Simple Addition

When a spreadsheet adds two numbers, the control unit fetches the addition instruction from memory and loads the two operands into registers.

The arithmetic logic unit performs the addition, the control unit stores the result back in a register or in memory, and the output unit displays the sum on screen, illustrating how the functional units cooperate to complete even the simplest operation.

bcalink.com

Computer architecture describes how the functional units of a computer are organized and how they interact to execute programs. The most influential model is the Von Neumann architecture, in which program instructions and data share the same memory and the same bus, meaning the CPU cannot fetch an instruction and access data at exactly the same time. The Harvard architecture, by contrast, uses separate memories and buses for instructions and data, allowing both to be accessed simultaneously and improving performance in specialized systems such as digital signal processors.

Regardless of architecture, every instruction a computer executes passes through the same basic instruction cycle: fetch, in which the control unit retrieves the next instruction from memory; decode, in which the control unit interprets what the instruction requires; execute, in which the arithmetic logic unit or another component carries out the operation; and store, in which any result is written back to a register or memory. This fetch-decode-execute cycle repeats continuously, often billions of times per second, for as long as the computer runs.

Processor architectures are also classified by instruction set design. Complex Instruction Set Computing (CISC) processors support a large number of specialized instructions, reducing the number of instructions per program at the cost of more complex hardware, while Reduced Instruction Set Computing (RISC) processors use a small set of simple, fast instructions, trading a larger instruction count for simpler, more efficient hardware often favoured in mobile and embedded devices.

Von Neumann vs. Harvard Architecture

Feature	Von Neumann Architecture	Harvard Architecture
Memory	Single shared memory for instructions and data	Separate memories for instructions and data
Bus	Single bus shared by instructions and data	Separate buses allow simultaneous access

Feature	Von Neumann Architecture	Harvard Architecture
Speed	Limited by shared-bus bottleneck	Faster due to parallel instruction/data access
Typical Use	General-purpose PCs and servers	Digital signal processors, some embedded systems

Table 7.1 — Comparing the two dominant computer architecture models.

$$\text{Clock Cycle Time} = 1 / \text{Clock Frequency}$$

Example 7.1 — The Instruction Cycle in Action

Suppose a program instructs the computer to multiply two numbers stored in memory. The control unit fetches the multiply instruction, decodes it to determine which registers hold the operands, and signals the arithmetic logic unit to execute the multiplication.

The result is stored back in memory, completing one full fetch-decode-execute cycle before the control unit moves on to fetch the next instruction in the program.

Memory is the component of a computer that stores data and instructions, either temporarily while a program is running or permanently for later retrieval. Because no single memory technology can be simultaneously the fastest, largest, and cheapest, computers organize memory into a hierarchy: small amounts of extremely fast, expensive memory sit close to the CPU, while large amounts of slower, cheaper memory provide bulk storage further away.

Primary memory, comprising RAM and ROM, is directly accessible to the CPU. Random Access Memory (RAM) is volatile, meaning its contents are lost when power is removed, and is further divided into faster, more expensive Static RAM (SRAM), often used for cache, and slower, cheaper Dynamic RAM (DRAM), used for main memory. Read-Only Memory (ROM) is non-volatile and retains data such as firmware even without power, with variants including PROM, EPROM, and EEPROM that differ in whether and how they can be reprogrammed.

Secondary memory, such as hard disk drives, solid-state drives, and optical media, provides large-capacity, non-volatile storage for programs and files that are not currently in use, at the cost of being much slower to access than primary memory. Between the CPU and RAM sits cache memory, a very small, very fast layer that stores copies of frequently used data to reduce the average time the CPU spends waiting for memory access.

The Memory Hierarchy

Memory Type	Volatility	Typical Role
Registers	Volatile	Hold data currently being processed by the CPU
Cache (SRAM)	Volatile	Store frequently used data close to the CPU

Memory Type	Volatility	Typical Role
Main Memory (DRAM/RAM)	Volatile	Hold the active program and its working data
ROM / Firmware	Non-volatile	Store startup instructions such as the BIOS/UEFI
Secondary Storage (HDD/SSD)	Non-volatile	Store programs and files long-term

Table 8.1 — The memory hierarchy, from fastest and smallest to slowest and largest.

$$1 \text{ KB} = 2^{10} \text{ Bytes} \mid 1 \text{ MB} = 2^{20} \text{ Bytes} \mid 1 \text{ GB} = 2^{30} \text{ Bytes}$$

Example 8.1 — Cache in Everyday Use

A CPU repeatedly needs the same small block of data while running a loop in a program. The first time the data is requested, it is fetched from the slower main memory and a copy is placed in the CPU's cache.

On every subsequent pass through the loop, the CPU finds the data already sitting in cache and avoids the slower trip to main memory, noticeably speeding up the program's execution.

An input device is any piece of hardware used to enter data, commands, or signals into a computer for processing. Input devices are commonly grouped by the type of data they capture: text and command entry, pointing and gesture, image and video capture, and audio capture, though many modern devices, such as touchscreens, blur these categories by combining several input methods in one component.

The keyboard remains the primary device for entering text and issuing commands, while pointing devices such as the mouse, trackpad, and touchscreen allow users to select, drag, and manipulate objects on screen. Image and document input is handled by devices such as scanners, webcams, and barcode readers, while microphones capture audio input for voice commands, recording, and communication. Biometric input devices, including fingerprint and iris scanners, are increasingly used for secure authentication.

Common Input Devices by Category

Category	Example Devices	Typical Use
Text/Command Entry	Keyboard	Typing text and issuing commands
Pointing	Mouse, trackpad, touchscreen	Selecting, dragging, and navigating on screen
Image/Document Capture	Scanner, webcam, barcode reader	Digitizing documents, images, and codes
Audio Capture	Microphone	Voice commands, recording, communication
Biometric	Fingerprint scanner, iris scanner	Secure identity verification

Table 9.1 — Common input devices grouped by the type of data they capture.

Example 9.1 — Choosing an Input Device for Secure Banking

A bank wants customers at an ATM to confirm their identity more securely than a PIN alone allows. Instead of relying only on a keypad, the bank installs a fingerprint scanner as an additional biometric input device.

The customer's fingerprint is captured and compared against a stored template before the transaction proceeds, showing how the choice of input device is driven directly by the security requirements of the application.

bcalink.com

An output device presents the information a computer has processed back to the user, in either soft-copy form, which is temporary and viewed on a screen or heard through speakers, or hard-copy form, which is permanent and produced on a physical medium such as paper. The choice between these two forms depends on whether the information needs to be reviewed immediately, stored physically, or shared with someone else.

Monitors remain the most common output device, displaying text, graphics, and video in soft-copy form, while speakers and headphones provide audio output. Printers, whether inkjet, laser, or 3D, produce hard-copy output, converting digital documents or models into physical objects. Projectors extend soft-copy output to a large shared surface, commonly used in classrooms and meeting rooms for presentations.

Common Output Devices

Device	Output Form	Typical Use
Monitor	Soft copy	Displaying text, graphics, and video
Speakers/Headphones	Soft copy	Producing audio output
Printer	Hard copy	Producing paper documents or 3D objects
Projector	Soft copy	Displaying output to a large shared audience
Plotter	Hard copy	Producing large-format technical drawings

Table 10.1 — Common output devices, grouped by whether they produce soft or hard copy.

Example 10.1 — Soft Copy and Hard Copy Together

An architect designs a building layout and first reviews it as soft copy on a monitor, adjusting dimensions and materials interactively on screen before the design is finalized.

Once approved, the final design is sent to a large-format plotter to produce a permanent hard-copy blueprint that can be taken to the construction site, illustrating how soft-copy and hard-copy output devices serve complementary roles in the same workflow.

bcalink.com

11

Interfaces

The connection points through which hardware and software communicate

An interface is the point of connection through which two systems, or a system and its user, communicate and exchange data. In computing, interfaces exist at both the hardware level, where physical ports and cables connect devices, and the software level, where programs and users interact through defined rules and visual controls.

Hardware interfaces such as USB, HDMI, and Ethernet define the physical connector, the electrical signalling, and the data-transfer protocol used to link peripherals such as printers, monitors, and storage devices to a computer. Wireless hardware interfaces, including Bluetooth and Wi-Fi, perform the same role without a physical cable, trading some speed and reliability for convenience and mobility.

Software interfaces govern how users and programs interact with a computer system. A Graphical User Interface (GUI) lets users interact through windows, icons, and menus, while a Command Line Interface (CLI) requires users to type text commands directly. An Application Programming Interface (API) allows two software programs to exchange data and instructions with each other, forming the backbone of most modern software integration.

Common Hardware and Software Interfaces

Interface	Type	Typical Use
USB	Hardware (wired)	Connecting keyboards, mice, storage, and other peripherals
HDMI	Hardware (wired)	Transmitting high-definition video and audio to a display
Bluetooth	Hardware (wireless)	Connecting nearby peripherals such as headsets and mice
Wi-Fi	Hardware (wireless)	Connecting a computer to a local or internet network

Interface	Type	Typical Use
GUI	Software	Allowing users to interact via windows, icons, and menus
CLI	Software	Allowing users to interact by typing text commands
API	Software	Allowing two programs to exchange data and instructions

Table 11.1 — Common hardware and software interfaces and their typical uses.

Example 11.1 — Hardware and Software Interfaces Working Together

A photographer connects a camera to a laptop using a USB cable, a hardware interface, to transfer photographs for editing.

The editing software then uses an Application Programming Interface (API) to send the finished images directly to a cloud storage service, showing how a single workflow can rely on both hardware and software interfaces in sequence.

Unit 1 — Key Takeaways

- A computer follows the input-process-output-storage (IPOS) cycle, and has evolved through five generations of switching technology.
- Speed, accuracy, diligence, storage, versatility, and automation are the defining characteristics that make computers useful.
- Computers are applied across education, business, healthcare, science, entertainment, and government, among many other sectors.
- Computers are classified by size and power (supercomputer to embedded), by purpose (general vs. special), and by data type (digital, analog, hybrid).
- Mobile computing combines portable hardware, adapted software, and wireless connectivity to let computing travel with the user.
- Every digital computer is built from the same functional units: input, control unit, ALU, registers, memory, output, and buses.
- Computer architecture, whether Von Neumann or Harvard, governs how the CPU repeatedly performs the fetch-decode-execute instruction cycle.
- Memory is organized into a hierarchy — registers, cache, main memory, and secondary storage — balancing speed, capacity, and cost.
- Input devices capture text, pointing, image, audio, and biometric data, while output devices present results as soft copy or hard copy.
- Interfaces, both hardware (USB, HDMI, Wi-Fi) and software (GUI, CLI, API), define how devices, programs, and users communicate.

— End of Unit 1: Introduction to Computer System —