



(b) Desktop PCs use firmware during the startup process.

(i) Identify the item that **must** be present in the firmware for a PC to be able to start.

- ☐ **A** Video driver
- ☐ **B** Operating system
- ☐ **C** Bootloader
- ☐ **D** Processor

(ii) State the type of memory used to store firmware.

(1)

Firmware is software or a program written on to ROM (non- volatile memory)

Why firmware is written in low level languages?

Firmware is written in low-level languages like Assembly or C because they allow direct control of hardware, provide efficient performance, manage limited resources well. These languages are good for devices with small storage, need precise hardware interaction, and have historical compatibility.

Why firmware is not stored in RAM?

- Firmware needs to persistently exist even when the device is powered off. RAM is volatile memory, meaning it loses its data when the power is turned off. Firmware must be retained to initialize the hardware and manage system functions every time the device starts up.
- Firmware can be quite large, depending on the complexity of the device. RAM is typically smaller in capacity compared to non-volatile memory, so storing firmware in RAM might not be feasible due to size limitations.

Give 4 differences of software and firmware

Software	Firmware
A set of instructions that specify what computer does.	A set of software that controls hardware devices.
Application SW, System SW	BIOS, EFI
Large size, between KB and GB.	Small size, a few KB.
High-level and low-level programming languages.	Only low-level languages.
Easy to update and/or modify.	Hard or impossible to update and/or modify.
Visible and interactable for the user.	Hidden and locked from the user.
Runs on CPUs or GPUs.	Runs on purpose-built and/or tiny processors.
Is subject to constant change.	Changes extremely rarely or never.
Long development cycles and a lot of time required.	Short development cycles and little time required.

What is the purpose of BIOS in a computer

The BIOS (Basic Input/Output System) is a fundamental component of a computer's hardware and software architecture. Its primary purpose is to initialize and manage the essential hardware components of a computer system during the startup process and provide basic communication between the operating system and the hardware.

----- Extra Knowledge -----

Bios tasks

- **Hardware Initialization:** The BIOS is responsible for initializing critical hardware components such as the CPU (Central Processing Unit), memory modules, storage devices (hard drives, solid-state drives), and peripheral devices (keyboard, mouse, display) when the computer is powered on.
- **Power-On Self Test (POST):** The BIOS conducts a Power-On Self Test to check if all the hardware components are functioning properly. If any hardware issue is detected, the BIOS typically provides an error message or a series of beep codes to indicate the problem.
- **Bootstrap Loader:** After the hardware is initialized and tested, the BIOS loads the initial part of the operating system from the boot device (usually a hard

drive or solid-state drive) into the computer's memory. This process is known as bootstrapping.

- **BIOS Setup:** The BIOS provides a user-accessible interface called the BIOS setup utility. This utility allows users to configure various hardware settings, such as boot order, hardware clock, and peripheral configurations.
- **CMOS Setup:** The BIOS stores configuration settings in a small piece of non-volatile memory called CMOS (Complementary Metal-Oxide-Semiconductor). These settings include information about the hardware, system preferences, and system time and date. Users can access and modify these settings through the BIOS setup utility.
- **Driver Loading:** The BIOS provides basic drivers for essential hardware components. These drivers enable the operating system to communicate with the hardware during the early stages of startup before the operating system's own drivers are loaded.

- **Compatibility and Legacy Support:** Older BIOS systems were designed to provide compatibility with legacy hardware and software. Newer systems may use UEFI (Unified Extensible Firmware Interface) instead of traditional BIOS for improved compatibility, security, and faster boot times.
- **Security and Boot Integrity:** Modern BIOS and UEFI implementations include security features such as Secure Boot, which verifies the integrity of the bootloader and the operating system to prevent unauthorized or malicious code from running during startup.
- **In summary,** the BIOS is responsible for initializing, testing, and configuring the hardware components of a computer during startup. It also plays a crucial role in loading the initial part of the operating system and ensuring that the system boots up correctly.