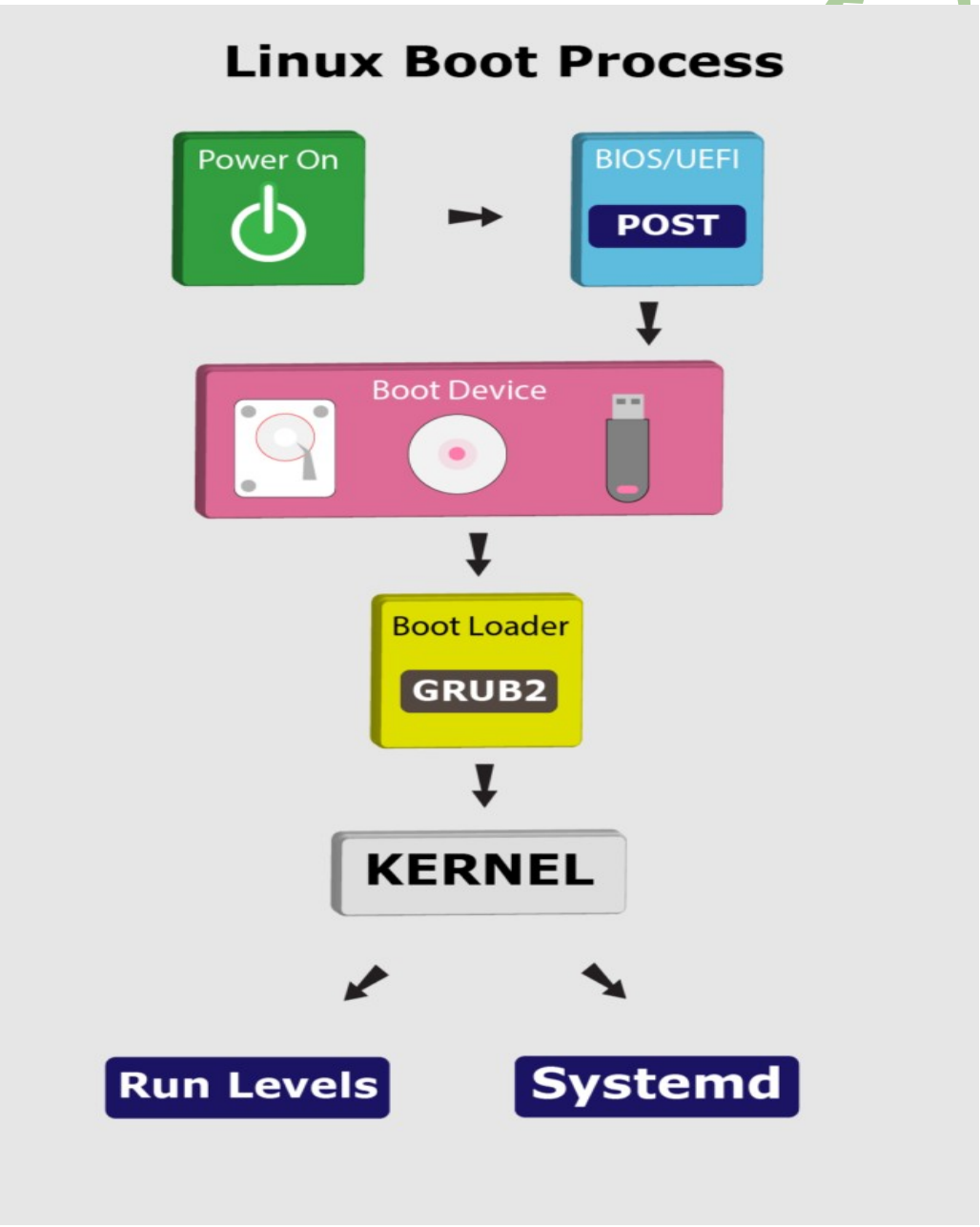
Booting Process

What is Boot process?

- The initlization of harware and software through the sequence process is known as Booting.

Basically two types of Booting are present 1.Primary Booting 2.Secondary Booting.

1. Primary : The booting which is used for the hardware initlization.
   - BIOS.
   - MBR.
2. Secondary : The Booting which comes from the OS initilization.
   - GRUB or LILO.
   - Kernel.
   - Init.
   - Run-levels.



Booting Process

BIOS:

BIOS is a program, stands for basic input/output systemwhich is stored in nonvolatile memory like ROM (Read Only Memory) or flash memory that allows you to set up and access your computer system at the greatest basic level. Although there is no need for most people to mess with the BIOS on a computer, it can be better to know about BIOS. It is found on motherboards that are a pre-installed program on Windows-based computers that executes when a computer is powered up. Before an OS is loaded, the CPU accesses the basic input/output system (BIOS). Then, the next function of BIOS is to examine all the hardware connections and detects all your devices.
The main function of BIOS is to set up hardware and start an OS, and it contains generic code that is needed to control display screens, the keyboard, and other functions. The BIOS is built-in software that manages the hard drives and cannot live on one. It cannot reside in the RAM as it is accessible before the computer system boots up. Actually, it lives on the ROM of the computer system, and mainly it is located on EPROM (erasable programmable read-only memory) chip. Therefore, the CPU accesses the EPROM when you turn on the computer and provides control to the BIOS.

Why We use EPROM other than ROM?

- Here Eprom is used store the basic usage data that is mostly important According to booting the code generating for hardware present in the EPROM it is activated when the power button is turned on.
- When the power button is turned on the transistor present near it tkes it as logical to activate the Eprom for the initilization of hardware code.

The primary intention of designing the BIOS system is to function with numerous devices that led to making a complimentary system chipset. There are some functions contained by the BIOS library that operates and controls system peripherals, and they can be initiated through external software.

Users can perform different functions by using the BIOS user interface, which is discussed Users can perform hardware configuration

- They can select boot drives
- They can set the system clock
- The BIOS allows users to enable and disable certain system components
- To BIOS user interface function, it provides set password prompts for secured access

BIOS Availability

The BIOS software is available in all modern computer motherboards. As the BIOS is a part of the motherboard; therefore, the BIOS's access and configuration on PCs are independent of any type of operating system. The BIOS is not dependent on anyone that means it does not matter which types of an operating system is running on the computer like Windows 7, Windows 8, Windows 10, Windows XP, Linux, Windows Vista, Unix, or no operating system at all, BIOS functions outside of the operating system environment.

Booting Process

What is included in a BIOS?

The directions to load basic computer hardware are included in BIOS software. A test is also included within it that is referred to as a POST (Power-On Self-Test). The POST helps the computer to boot up properly with verifying the computer meets requirements. Your computer produces various forms of beeps if the POST test fails at the time of test.

Functions of BIOS

The BIOS has different instructions that are required to load the hardware, and it is responsible for loading the operating system. The major functions of a Basic input/output system (BIOS) are discussed below:

- **BIOS Power on Self-Test (POST):** It ensures the proper functioning of the computer hardware as it is a built-in diagnostic program that. In the system, it verifies the computer meets the necessary parts and functionality. POST does this function efficiently. It ensures that the computer is loading tasks successfully, such as the use of memory, a keyboard, and other parts when it starts up. If the POST test fails at the time of test, the computer provides a combination of beeps to display the error type, and the system continues to boot when the POST test is passed completly.Once the self-test has been passed, and the basic instructions have been loaded, the computer starts to load the OS from one of the connected drives to the system. The BIOS settings can also be changed by the computer users with the help of a configuration screen on the computer. The BIOS information can also be stored on the flash memory, which can be updated by computer users after releasing an update by vendors. BIOS actually can be located in between the external devices and the computer because its name describes that it is used for reading and writing to and hard disk and floppy disc, displaying values on the screen, reading the keystroke, etc.
- **Bootstrap Loader:** The BIOS recognizes and locates the operating system when the POST running successfully. The program bootstrap loader is contained by BIOS, which searches and starts the OS boot program. When BIOS detects one, it transfers access to Operating System that is known as booting.
- **BIOS drivers:** BIOS drivers are stored in the non-volatile memory, whose primary function is to supply basic computer hardware information.
- **BIOS Setup Utility Program:** It is a configuration software, also known as CMOS setup, that allows users to configure hardware settings as well as device settings, time and date, computer password. The NVRAM, non-volatile memory, is used to store settings of memory, disk types, and information about the computer system; this information is not stored in the BIOS chip

The users run the BIOS setup program during the installation of a system and input the correct parameters. The CMOS (Complementary metal-oxide-semiconductor) is a required material to construct NVRAM. These CMOS chips store and maintain data on very low values of current. Therefore, the system's configurations are also called CMOS settings. With the help of using a capacitor, maintaining the battery backup, or by a battery built into the NVRAM chip, CMOS settings can be maintained. Additionally, a system clock is also included in this chip. The setting

remains for a short period of time if there is no battery. And there is a need to reset the system. Through its BIOS set up program is protected as there is loss of BIOS password.

With computers, BIOS come in-built as firmware on a chip on the motherboard. In contrast, an operating system such as iOS or Windows can be installed by the user or pre-installed by the vendor or manufacturer.

When BIOS boots up a computer, it verifies the computer meets the necessary attachments to boot up. The files are contained by any piece of hardware that needs for a computer to start up, this piece of hardware is known as a boot device.

Different types of BIOS

1. **UEFI:** The UEFI can accommodate 2.2 TB or larger drives, which stands for Unified Extensible Firmware Interface. It handles drives with the help of using the Master Boot Record instead of GPT technology, the more modern GUID Partition Table. Furthermore, Apple's Mac PCs have never used the BIOS.
2. **Legacy BIOS:** The Legacy BIOS was used in older motherboards to turn the PC on. Legacy BIOSes have limitations as they have no ability to handle or recognize drives bigger than 2.1 TB. However, it controls how the CPU and the components communicate with each other.

Upgrading a BIOS

Often, it is discovered that when the computer needs to be upgraded with latest hardware, it does not support all the features of the latest hardware. It needs to be upgraded hardware like more memory, a larger hard drive, or a new video card. Upgrading the BIOS chip is an easier solution to this problem. Generally, to upgrade BIOS on the system, the files and information are available on the computer's or motherboard maker's Web site.

Unluckily, it is most important to know that upgrading the BIOS can be a drastic step. If you are going to upgrade the BIOS, you will be better to back up all of your data from the hard drive. Also, make sure there is a recovery jumper that makes you capable of recovering the original BIOS. Although upgrading the BIOS is trouble-free, it is possible for the by upgrading BIOS the system can be computer unusable, damage or destroy the BIOS chip.

BIOS manufacturers

- Foxconn
- AMI
- Hewlett Packard (HP)
- Ricoh
- Asus

It may be very important to know about the motherboard manufacturer as users sometimes need to update their BIOS and cars. For example, a video card to the most recent versions. When you update drivers, the patch can cause recent BIOS-level security vulnerabilities, or computer performance can be improved.

Booting Process

To Know about the type of BIOS using

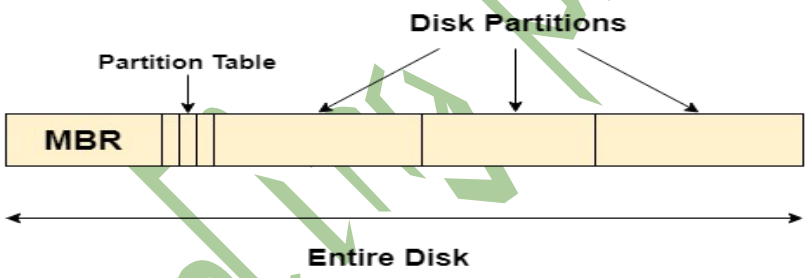
- In Linux : sudo dmidecode | less.
- In Windows : Windows + r then type (msinfo32)

BIOS ERRORS

- Failed to overclock Error.
- Failed Device Error.
- Fan Error.
- Bootdevice Error.

Master Boot Record (MBR)

- Master boot record is the information present in the first sector of any hard disk. It contains the information regarding how and where the Operating system is located in the hard disk so that it can be booted in the RAM.
- MBR is sometimes called master partition table because it includes a partition table which locates every partition in the hard disk.
- Master boot record (MBR) also includes a program which reads the boot sector record of the partition that contains operating system.



What happens when you turn on your computer?

- Due to the fact that the main memory is volatile, when we turn on our computer, CPU cannot access the main memory directly. However, there is a special program called as BIOS stored in ROM is accessed for the first time by the CPU.
- BIOS contains the code, by executing which, the CPU access the very first partition of hard disk that is MBR. It contains a partition table for all the partitions of the hard disk.
- Since, MBR contains the information about where the operating system is being stored and it also contains a program which can read the boot sector record of the partition, hence the CPU fetches all this information and load the operating system into the main memory.

Erros in MBR : When you start your computer, it stops booting. It displays "Error loading operating system", "Missing operating system", "Invalid partition table", "NTLDR is missing" or "Master boot record corrupted".

Booting Process

GRUB:

GNU GRand Unified Bootloader (GRUB) is a boot loader created by the *GNU project*. GRUB allows the user to choose from a list of operating systems to load, allowing numerous operating systems to run on the same machine. GRUB is the default boot loader in most modern Linux distributions. GRUB may be customized dynamically since it permits changes to the configuration during boot. Users are given a simple command line interface through which they can dynamically input new boot configurations. GRUB provides several user-friendly characteristics, such as high portability, independence from geometry translation, support for many executable formats, and support for many file systems, including most UNIX systems, *NTFS*, *VFAT*, and *LBA (Logical Block Address)* mode. Most Linux distributions that use GRUB give a customized boot menu by leveraging its support for numerous GUIs (Graphical User Interfaces). GRUB2 is currently replacing GRUB, and GRUB has been renamed GRUB Legacy.

- GRUB is a boot loader that may be used with Linux, DOS, vSTA, and other OS. On the other hand,GRUB supports Windows, Linux, UNIX, macOS, BSD, and Solaris.

LILO:

LILO is a Linux boot loader found in Linux-based devices that have been one of the most widely used and oldest boot loaders. It only supports a single OS, which is Linux OS. It has been the default boot loader of Linux OS based devices for several years after gaining popularity from loading. Compared to GRUB, it's an outdated boot loader and lacks a graphical user interface menu option.Although GRUB is now present in most OS systems, LILO and ELILO are still incredibly popular in the modern days. LILO software has been good and dependable, keeping the propriety and operating system effectively.

Kernel

Kernel is a computer program that is a core or heart of an operating system. Before discussing kernel in detail, let's first understand its basic, i.e., Operating system in a computer.

Operating System

An operating system or OS is system software that works as an interface between hardware components and end-user. It enables other programs to run. Each computer system, whether it is desktop, laptop, tablet, or smartphone, all must have an OS to provide basic functionalities for the device. Some widely used operating systems are **Windows, Linux, MacOS, Android, iOS, etc.**

Role of Kernel in OS :

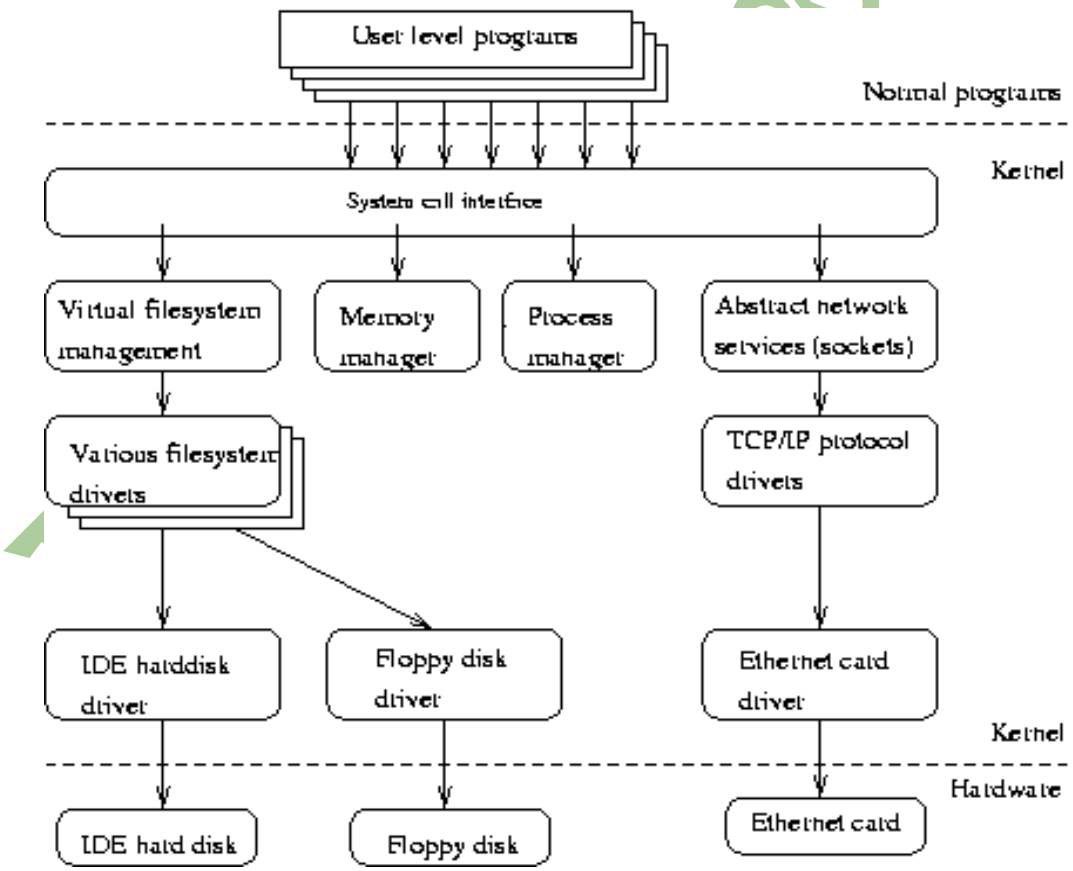
- Kernel is the core part of an OS(Operating system); hence it has full control over everything in the system. Each operation of hardware and software is managed and administrated by the kernel.
- It acts as a bridge between applications and data processing done at the hardware level. It is the central component of an OS.
- It is the part of the OS that always resides in computer memory and enables the communication between software and hardware components.

Booting Process

- It is the computer program that first loaded on start-up the system (After the bootloader). Once it is loaded, it manages the remaining start-ups. It also manages memory, peripheral, and I/O requests from software. Moreover, it translates all I/O requests into data processing instructions for the CPU. It manages other tasks also such as memory management, task management, and disk management.
- A kernel is kept and usually loaded into separate memory space, known as protected Kernel space. It is protected from being accessed by application programs or less important parts of OS. Other application programs such as browser, word processor, audio & video player use separate memory space known as user-space.
- Due to these two separate spaces, user data and kernel data don't interfere with each other and do not cause any instability and slowness.

Functions of a Kernel

A kernel of an OS is responsible for performing various functions and has control over the system. Some main responsibilities of Kernel are given below:



- **Device Management**
  To perform various actions, processes require access to peripheral devices such as a mouse, keyboard, etc., that are connected to the computer. A kernel is responsible for controlling these devices using device drivers. Here, a device driver is a computer program that helps or

enables the OS to communicate with any hardware device.A kernel maintains a list of all the available devices, and this list may be already known, configured by the user, or detected by OS at runtime.

- **Memory Management**
  The kernel has full control for accessing the computer's memory. Each process requires some memory to work, and the kernel enables the processes to safely access the memory. To allocate the memory, the first step is known as virtual addressing, which is done by paging or segmentation. Virtual addressing is a process of providing virtual address spaces to the processes. This prevents the application from crashing into each other.
- **Resource Management**
  One of the important functionalities of Kernel is to share the resources between various processes. It must share the resources in a way that each process uniformly accesses the resource.The kernel also provides a way for synchronization and inter-process communication (IPC). It is responsible for context switching between processes.
- **Accessing Computer Resources**
  A kernel is responsible for accessing computer resources such as RAM and I/O devices. RAM or Random-Access Memory is used to contain both data and instructions. Each program needs to access the memory to execute and mostly wants more memory than the available. For such a case, Kernel plays its role and decides which memory each process will use and what to do if the required memory is not available.The kernel also allocates the request from applications to use I/O devices such as keyboards, microphones, printers, etc.

Types of Kernels:

1. Microkernel : Linux,Unix.
2. Monolithic Kernel : AmigaOS,Minix.
3. Hybrid Kernel : combination of monolithic and micro kernels Ex.Windows,Netware,Beos.
4. Nanokernel : Eros.
5. Exokernel : Inprocess

Init

It is the first process executed by the kernel during the booting of a system. It is a daemon process which runs till the system is shutdown. That is why, it is the parent of all the processes. First of all, **init** reads the script stored in the file **/etc/inittab.** Command **init** reads the initial configuration script which basically take care of everything that a system do at the time of system initialization like setting the clock, initializing the serial port and so on.

By reading this file, **init** determines how the system should be set up in each runlevel and sets default run level.

After determining default runlevel for the system, **init** starts all background processes required to run the system. First it runs each of the kill script (their file name starts with a K) with a stop

parameter. Then it runs all start scripts (their file name starts with an S) to start all services and applications.

- **Getty :** Short for getty is “get tty” is takes care of the user login when the user id and password matches then it provides the user space for the user and it creates shell for the user. If the user id and password doesn’t matches then it wouldn’t creates shell. This runs as the system program not as the kernel.

Runlevels

A runlevel is a software configuration of Linux system which permits only a selected group of processes to exist. It defines what services are operating on the system.

Runlevels are identified by numbers. **init** can be in one of eight runlevels. It is changed by a privileged user run **telinit,** which sends appropriate signals to **init** to change runlevel.

Runlevel	Function
0	Halt the system
1	Single user mode
2	Multiuser mode without networking
3	Multiuser mode with networking
4	Not used
5	Multiuser with networking and X windows
6	Reboot the system
S/s	Not used directly

- 0,1 and 6 are reserved runlevels.
- Runlevel S or s are same.
- 7-9 are also valid runlevels, though they are not documented as traditional Unix variants, do'nt use them. But they are same as runlevels S or s. They are aliased.

Changing runlevels

There are many ways to change runlevels. To make the changes permanently, change the default level in the file **/etc/inittab.**

After all the processes, **init** waits for one of its descendent process to die, for a powerfail signal. Or wait until **telinit** signals to change the runlevel. When above conditions occur, it re-examines the **/etc/inittab** file. New entries can be added to this file any time.

When **init** is not in single user mode and receives a power fail signal **(SIGPWR),** it reads the file **/etc/powerstatus.** Based on the content of this file, **init** starts a command:

- **F** (FAIL) ? When power fails, UPS provides the power. Execute the powerfail and powerwait entries.
- **O** (OK) ? Power has been restored, execute powerokwait entries.
- **L** (LOW) ? Power is degrading and UPS has a low battery, executes powerfailnow entries.

Booting Process

If file **/etc/powerstatus** doesn't exist or contains anything other than **F, O, L,** then **init** behaves as if it has read the letter **F.**

To interact with **init, /dev/initctl** control channel should be used rather than **SIGPWR** and **/etc/powerstatus.**

When a request to change the runlevel is send to **init,** init sends warning signal **SIGTERM** to all the undefined processes in the new runlevel. It then waits for 5 seconds before terminating processes via **SIGKILL** forcibly.

All these processes should remain in the same process group which was created by the init, otherwise they will not be able to receive these signals and they need to be terminated separately.

Telinit

/sbin/telinit is linked up with **/sbin/init.** Telinit takes up an argument and signals **init** to perform the respective function.Arguments are as follows with are one character each:

Argument	Function
0,1,2,3,4,5,6	Switch to specified runlevel
a,b,c	Processes only file entries from **/etc/inittab** having runlevel a,b,c.
Q or q	Re-examine file **/etc/inittab**
S or s	Switch to single user mode
U or u	init re-execute itself. No re-examine occurs, runlevel should be from S,s,1,2,3,4,5 otherwise request would be ignored silently.

Default time to wait between sending the signals SIGTERM and SIGKILL is 5 seconds, but **telinit** can also ask **init** to change this time with the option **-t sec.**

Only users having appropriate privileges can execute **telinit.**

The **init** binary checks by the process ID whether it is **init** or **telinit, init's** process ID is always **1.** so one can also use **init** instead of **telinit** as shortcut.