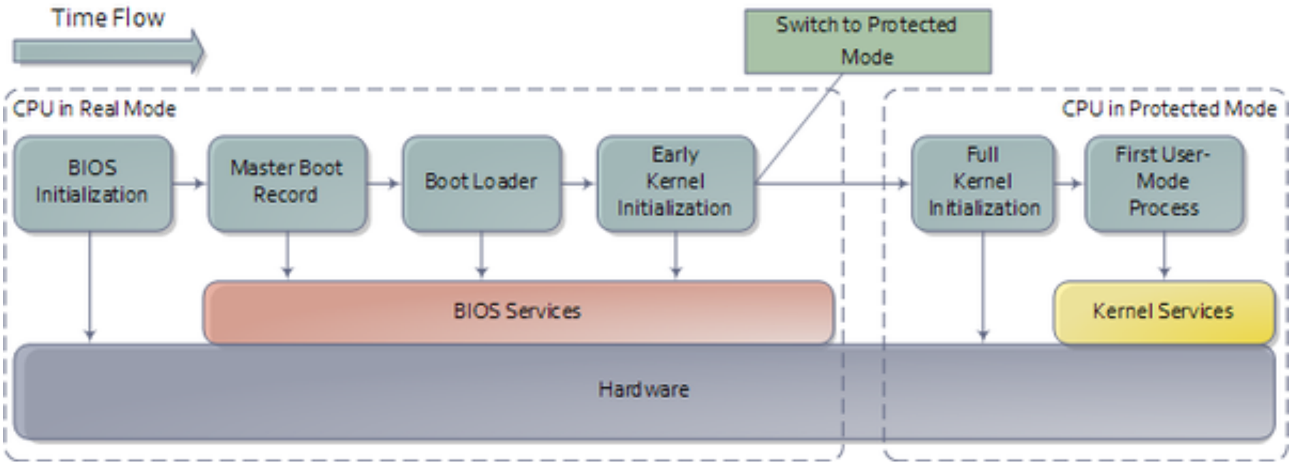
Regardless of the computer or operating system, standard ("IBM-compatible") desktop PCs and laptops all power on and start up using one of two ways: the traditional BIOS-MBR method and the newer UEFI-GPT method, used by the latest versions of Windows, Linux, and Mac OS X on newer PCs, laptops, and tablets.

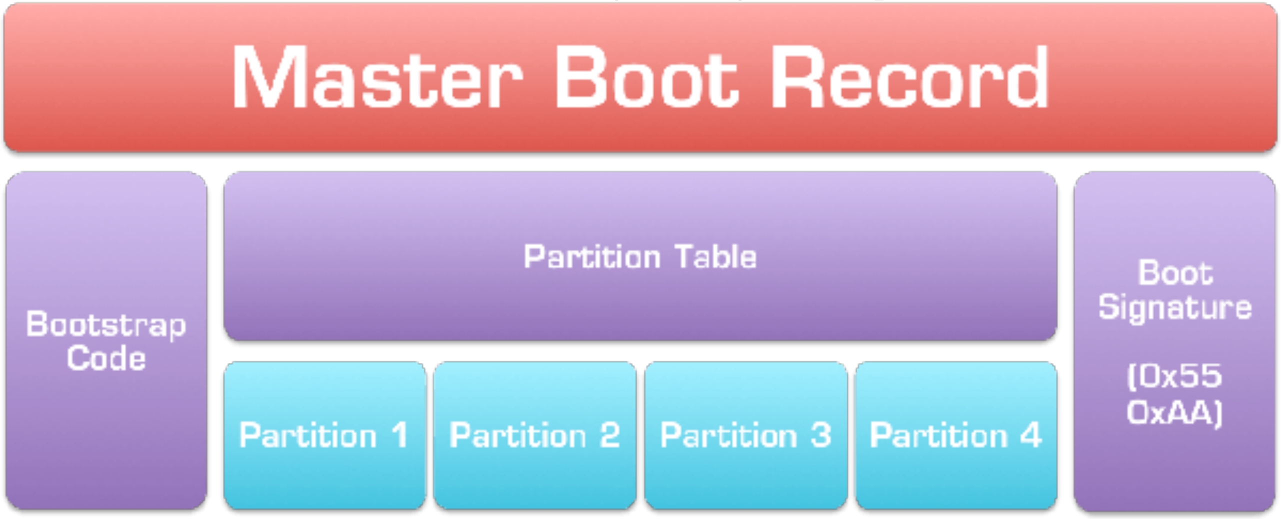
- **Step 1:Power Supply**
  - Normal not much important .
- **Step 2: Bootstrapping**
  - *Its like computer wakes up and does not what to do next. So, this is performed.*
  - Something has to be programmed by default, so that the CPU knows where to search for instructions.
  - This is an address location in the ROM. This address location is almost always constant in X86 based computers. The address location is FFFF:0000h.
  - This address location is the last region of the ROM. It only contains one instruction. The instruction is to jump to another memory address location. This JUMP command, will tell the location of the BIOS program in the ROM.
  - This is how the computer will come to know where the BIOS program is located.
- **Step 3: The Role of BIOS in booting process**
  - (Now from here something real starts)
  - The BIOS is the lowest level of software that interfaces with the hardware as a whole, and is the interface by means of which the bootloader and operating system kernel can communicate with and control the hardware.
- **BIOS stands for Basic Input Output System.**

1. The BIOS does several things. This is its usual sequence:
2. Check the CMOS Setup for custom settings
3. Load the interrupt handlers and device drivers
4. Initialize registers and power management
5. Perform the power-on self-test (POST)
6. Display system settings
7. Determine which devices are bootable
8. Initiate the bootstrap sequence

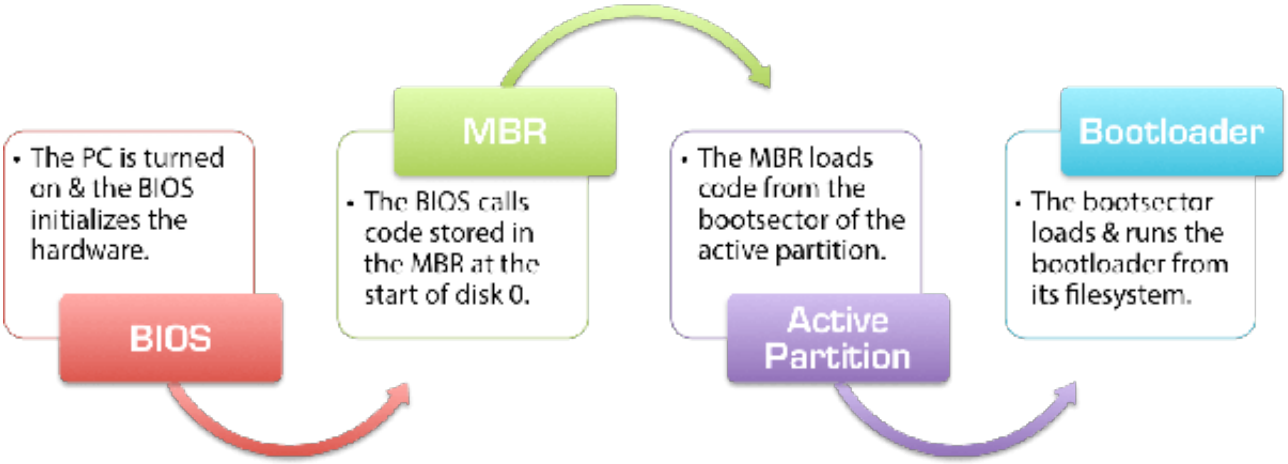
- CMOS is a small memory RAM chip that's present in the motherboard. This RAM is different from the computers main RAM chip (which are replaceable memory chips. Unlike the main RAM chip, CMOS RAM does not flush its memory when a computer is turned off. It remembers all the configuration with the help of a battery called CMOS battery.
- **Its normally a misconception that people have is that when they modify the boot order they change the NIOS setting but in reality they modify the CMOS settings is the place where you modify the boot order etc.**
- Once the POST check is completed successfully,.
- **BIOS will look CMOS settings to know what is the boot order. Boot order is nothing but a user defined order which tells where to look for the operating system.(In case we changed the boot order)**
- In other word if we have the order in which we BIOS have to look for the Bootable file. (Eg ;- USB stick, Disk, Harddrive, Network)



- **Step 4: MBR and GRUB**
    - **Pay attention here ,it is really imp.**
    - BIOS is programmed to look at a permanent location on the hard disk to complete its task. This location is called a **Boot sector.** This is nothing but the first sector of your hard disk. This area is sometimes called as **MBR (Master Boot Record)**. This is the location that contains the program that will help our computer to load the operating system. As soon as bios finds a valid MBR, it will load the entire content of MBR to RAM, and then further execution is done by the content of MBR.
    - ***Most important point to note is below:***
        - Regardless of whether the BIOS was configured to boot from a **local hard disk or from a removable USB stick**, the handoff sequence is the same. Once the BIOS POST and AddOn ROM procedures have completed, the BIOS loads the first 512 bytes from the hard drive of the selected boot device – these 512 bytes are what is commonly known as the MBR, or the Master Boot Record.
        - **The first sector on a hard disk is called the Master Boot Record (MBR).** This sector is only 512 bytes long and contains a small piece of code (446 bytes) called the primary boot loader and the partition table (64 bytes) describing the primary and extended partitions.
        - By default, MBR code looks for the partition marked as active and once such a partition is found, it loads its boot sector into memory and passes control to it.
        - **GRUB replaces the default MBR with its own code.**
    - **The MBR contains the first stage of the grub, and partition table information.**



- **GRUB is a BOOT LOADER.**
    - GRUB HAS 3 STAGES:-
        - **Grub stage 1 in the MBR will look for a partition with active flag set on it. Please remember the fact that there can only be one active partition from the 4.**
        - **MOST IMPORTANT PART STARTS:-**
        - The primary job of the **stage 1** bootloader is to load the second stage boot loader. The second stage boot loader is the **stage 2** grub, that actually does the job of loading the kernel and other initrd image files (we will come to that part in some time). **GRUB (Grand Unified Boot Loader) is the combined name given to different stages of grub.**
        - Now from where does this **stage 1.5** came. The hard disk sectors are counted form 0 to the last sector. As explained previously the first sector (sector 0) contains the GRUB stage 1. Normally partititons will not start before sector 63. So partitions will start form sector number 63. Hence we have sectors form 1-63 free. This space is used for storing GRUB stage 1.5. This free space between MBR and the beginning of the partitions is called as MBR GAP.
        - Now you might think what is the requirement of an additional stage in grub. If you are a linux guy, you might have already configured grub configuration file. If you remember the configuration, it contains the kernel file location and name, its partition. Now how will the grub access those kernel files without the file system drivers?
        - **Grub Stage 1.5 located in the MBR GAP (sector 1 to 63 before the beginning of the first partition) basically contains the drivers for reading file systems. So grub stage 1 will load grub stage 1.5 to the RAM, and will pass the control to it.**
        - Now grub stage 1.5 will load the file system drivers and once the file system drivers are loaded, it can now access /boot/grub/grub.conf file which contains other details about kernel path and initrd path etc.
        - **Now this is the point where you are presented with a beautiful TUI (Terminal user interface), where you can select your operating system kernal and press enter to boot it.**
        - **Thus after the above step we choose the OS that we want use of all the installed OS in our Machine.**


• The PC is turned on & the BIOS initializes the hardware.
BIOS
MBR
• The BIOS calls code stored in the MBR at the start of disk 0.
• The MBR loads code from the bootsector of the active partition.
Active Partition
Bootloader
• The bootsector loads & runs the bootloader from its filesystem.

- **Step 5 Loading The kernel Image**
    - Kernel is a compressed image file. The location of this compressed kernel image is specified in the grub 2 configuration file. Its basically an executable bzImage file.

- Normal OS loading will continue depending on OS chosen.