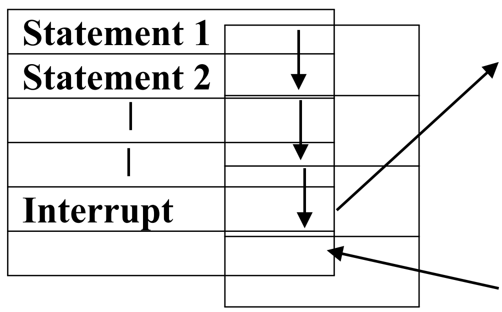
UNIT-4

What is Interrupt? Explain in detail?

- **An Interrupt is as subroutine call initiated by external devices through hardware (hardware interrupt) or microprocessor itself (software interrupt).**
- **An interrupt can also be viewed as signal which suspends the normal sequence of microprocessor and then microprocessor gives service to that device which has given the signal. After completing the service microprocessor again return to main program.**
- **Microprocessor is connected to different peripheral devices. To communicate with this devices microprocessor 8085 uses interrupt method.**
- **An interrupt is an input signal which transfer control to specific routine known as Interrupt Services Routine (ISR) after execution ISR, control is again transfer to main program.**
- **There are two types of interrupt :-**
  1) Hardware interrupt
  2) Software interrupt
  Software interrupt has more priority than any Hardware interrupt.



Main program **ISR**

Fig :-Interrupt

- **Software interrupt are not requested by external peripheral devices. All software interrupt are non-maskable. Some hardware interrupt are maskable.**

THERE ARE 5 HARDWARE INTERRUPT:-

1) TRAP: High priority (non maskable)

2)RST 7.5

3)RST 6.5

4) RST 5.5

5) INTR : Low priority

1) These interrupt are vectored interrupt. It means that when the interrupt are given, it is directed (or vectored) to transfer the control to specific memory location given by :

TRAP = 4.5*8=0024H

RST 7.5 = 7.5* 8 =003CH

RST 6.5 = 6.5*8=0034H

RST 5.5 = 5.5*8=002CH

2) Among these Trap is non-maskable interrupt which cannot be disabled. But the other four interrupt are maskable interrupt, which can be disabled.

3) Trap has highest priority and INTR has lowest priority among the hardware interrupt. The hardware interrupt in descending order of priority are listed below:

A) TRAP: Highest priority

B) RST 7.5

C) RST 6.5

D) RST 5.5

E) INTR: Lowest priority

Non-maskable interrupt:-

1) The interrupt cannot be masked or cannot be kept pending.

2) Non -maskable interrupt disables all maskable interrupts.

3) It has highest priority than maskable interrupts.

4) It is always vectored interrupt.

5) Response time for non-maskable interrupt is slow.

Maskable interrupt :-

1) The interrupt can be masked or kept pending.

2) A maskable interrupt cannot disable any non-maskable interrupt.

3) It has lowest priority than any non-maskable interrupt.

4) It may be vectored or non-vectored interrupt.

5) Response time for non- maskable interrupt is high.

Hardware interrupt:-

1) To handle asynchronous events.

2) These interrupt are requested by external devices.

3) After execution of these interrupt, program counter is not incremented.

4) The microprocessor executes either interrupt acknowledge cycle to acknowledge interrupt.

5) These interrupt may be non-maskable or maskable.

6) They have lower priority than any other software interrupt.

7) These interrupt affects an interrupt control logic.

8) It improves throughput of the system.

9) There are 5 hardware interrupts:-

TRAP, RST 5.5, RST 6.5, RST 7.5, INTR.

Software interrupt:-

1) Software interrupt are used to handle asynchronous events.

2) These interrupt are not requested by external devices but by microprocessor.

3) After execution of these interrupt program counter is incremented.

4) The microprocessor does not execute any interrupt acknowledge cycle. It executes normal instruction cycle.

5) They cannot be masked or ignored.

6) Software interrupt have more priority than hardware interrupt.

7) These interrupt does not effect on interrupt control logic.

8) They does not improve throughput of the system.

9) Software interrupt are from RST 0 TO RST 7, Where RST stands for restart.

Application I/O interface:-

Devices may vary in many dimensions, as shown in the table

- ***Character-stream or block*:- A character stream device transfer bytes one by one, whereas a block devices transfer a block of bytes as a unit.**

- ***Sequential or random access* :- A sequential devices transfer data in fixed order that is determined by the devices, whereas the user of random access devices can instruct the devices to seek any of the available data storage locations.**

Aspect	Variation	Example
Data –transfer mode	Character Block	Terminal Disk
Access method	Sequential Random	Modem CD-ROM
Transfer schedule	Synchronous Asynchronous	Tape Keyboard
Sharing	Dedicated Sharable	Tape Keyboard
Device speed	Latency Seek time Transfer rate Delay between-operations	
I/O operation	Read only Write only Read-write	CD-ROM Graphics controller disk

Table:- Characteristics of I/O devices

Synchronous:-

A synchronous device is one that performs data transfer with predictable responses time.

Asynchronous:-

An Asynchronous device exhibits irregular or unpredictable responses time.

Sharable or Dedicated:-

Shareable devices can be used concurrently by several processes or threads; dedicated devices cannot.

Speed of operation:-

Devices speed range from fewer bytes per second to few gigabytes per second.

Read-Write,read only,or write only:-

Some devices perform both input and output, but others support only one direction.

***Block and character devices*:-**

Block devices interfaces capture all the aspect necessary for accessing disk drivers and other block-oriented devices. The expectations is that the devices understand command such as read and write, and if it is random access devices, it has a seek command to specify which block is to transfer next applications normally access such as a devices through a file system interfaces.

A keyboard is a device that is accessed through a character stream interface. The basic system calls in this interface enable an application to get or put one character.

KERNEL I/O SUBSYSTEM

Services provided by kernel's are as follows:

1. I/O Scheduling

- **I/O request scheduling can reduce the overall system performance, can share devices access fairly among processes, and can reduce the average waiting time for I/O to complete.**

Example:- let a disk arm is near the beginning of a disk, and that three applications issue blocking read calls to that disk. Application 1 request to block at the end of the disk, application 2 request one near the disk and application 3 request at the middle of the disk. The operating system can reduce the distance that the disk arm travel by serving the applications in order 2, 3, 1 rearranging the order of service in this way is the essence of I/O scheduling. Another way is by using storage space in main memory or on disk via techniques called buffering, caching and spooling.

2) *Buffering*

- **A Buffer is a main memory area that stores data while they are transformed between two devices or between a device and an application. Buffering is done for three reasons.**

a) To cope with a speed mismatch between the produce and consumer of the data stream b) Use of buffering is to adapt between devices that have different data transfer sizes. Such disparities are especially common in computer networking, where buffers are used widely for fragmentation and reassembly of messages. c) Use of buffering is to support copy of semantics for application I/O.

3) *Caching* :-It is region of fast memory that holds copies of data. Access to the cached copy is more efficient than access to the original. For instance, the instructions of the currently running process are stored on disk, Cached in physical memory, and copies again in the CPU's secondary and primary cache. The difference between a buffer and a cache is that a buffer may hold the only existing copies of a data item, whereas the cache just holds copy on faster storage of an item that resides elsewhere.

4)*Spooling*:-(SPOOL: Simultaneous Peripheral Operation on-line) It is buffer that holds output for a devices, such as a printer , that cannot accept interleaved data streams. Although a printer can serve only one job at a time, several applications may wish to print their output concurrently, without having their output mixed together. The operating system solves this problem by intercepting all output to the printer. Each application's output is spooled to a separate disk file. When an application finishes printing, the spooling system queues the corresponding spool file for output to the printer. Each application output is spooled to a separate disk file. When an application finishes printing, the spooling system queues the corresponding spool file for output to the printer.

5)*Device Reservation*:

I/O Subsytem supervises:-

1) **Management of the name space for files and devices.**
2) **Access control to file and devices.**
3) **Operation control (for example, a modem cannot seek)**
4) **File system space allocation**
5) **Device allocation**
6) **Buffering, caching and spooling.**

7) I/O scheduling.

8) Devices status monitoring, error handling, and failure recovery.

9) Device driver configuration and initialization.

The upper levels of the I/O subsystem access devices via the uniform interface provided by the devices drivers.

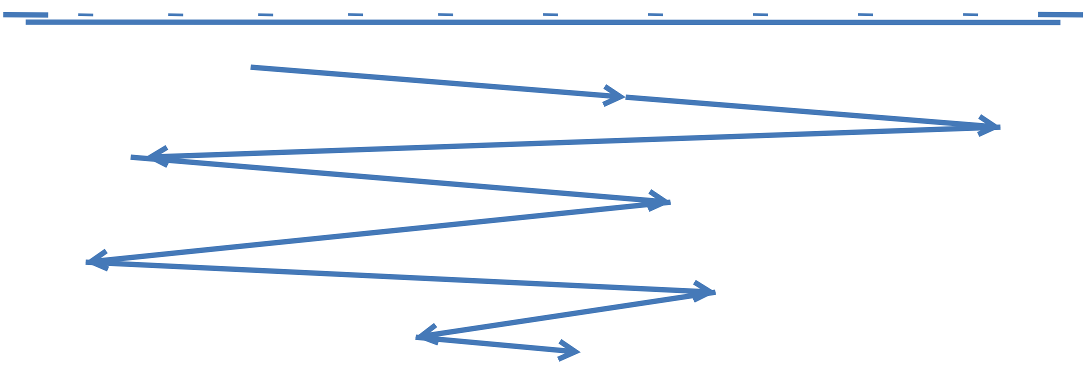
Disk scheduling:

Various disk scheduling algorithms to have fast access time and disk bandwidth. Bandwidth is the total number of bytes transferred, divided by the total time between first request for service and completion of the last transfer.

1. *FCFS scheduling*(first come first serve):

Request are served on first served basis. This algorithm does not provide fast access. Consider a disk queue with requests for I/O to blocks on cylinder 98,183,37,122,14,124,65,67.

Head starts at cylinder 53.



Total head movement of 640 cylinders.

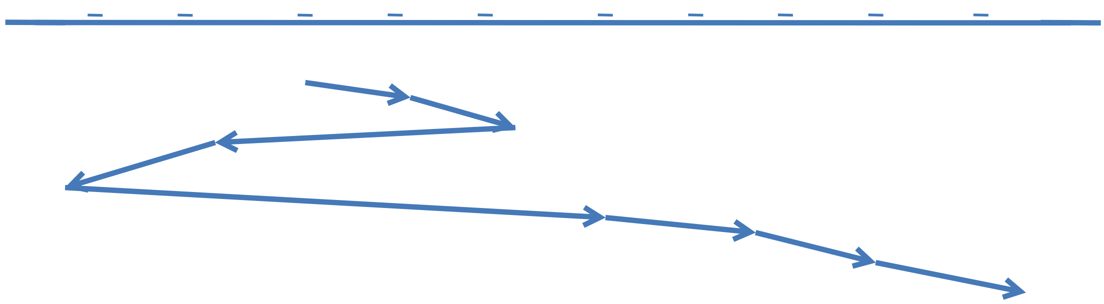
Drawback:

Head may swing wildly from track temporarily skipping some tracks that need to be used. Very inefficient.

2. ***SSTF Scheduling*(Shortest seek time first)**

This scheduling processes I /O in tracks closest to the current position of the head.

Consider the example of FCFS



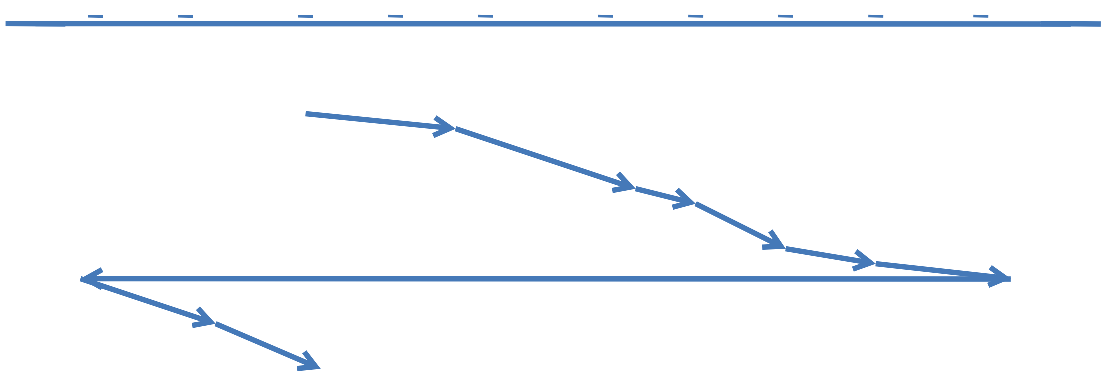
Total head movements of 236 cylinders.

Drawbacks:

It may result in starvation of some requests, if most request are clustered together within a few tracks, but others are far away.

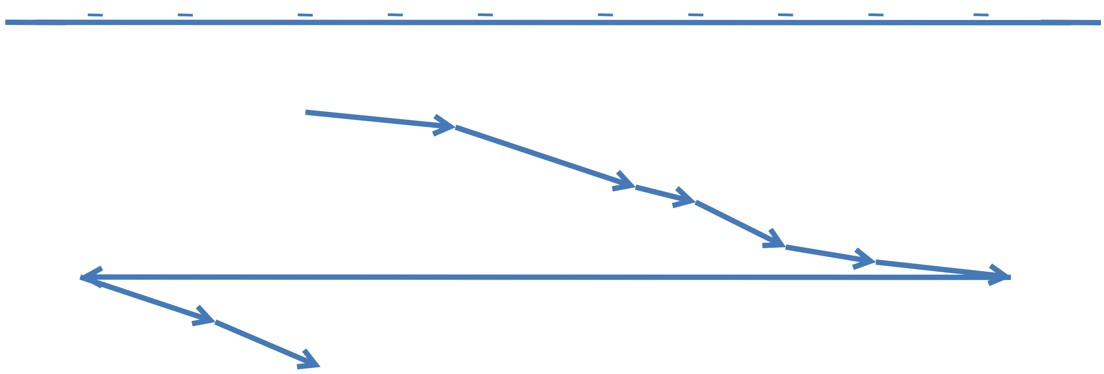
3. ***SCAN Scheduling:***

Start with lowest track number requests, and process requests in track order, until highest is found and then proceed in reverse order. This algorithm is also called elevator algorithm.

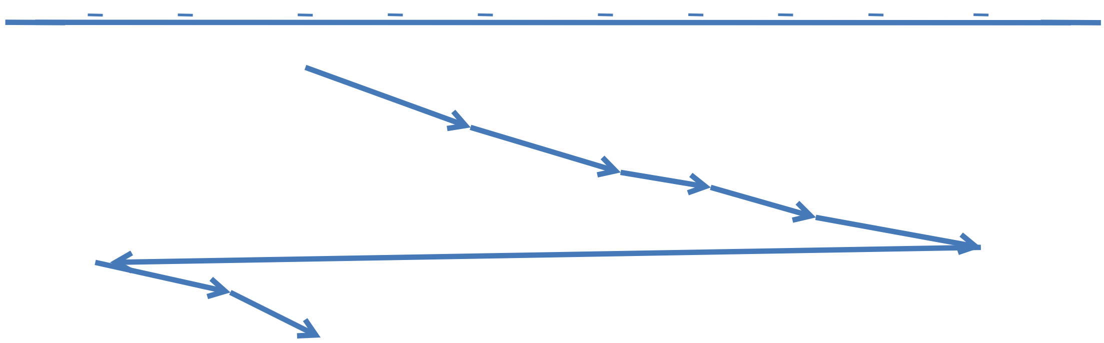


4. ***C-SCAN Scheduling:* (circular)**

It is designed to provide a more uniform wait time. C-SCAN moves the head from one end of the disk to the other, services request along the way. When the head reaches the other end, it immediately returns to the beginning of the disk.



5. <u>Look scheduling:</u>
   Both SCAN and C-SCAN moves the disk across the full width of the disk. But neither algorithm is implemented this way. The arm goes only as far as the final request in each direction. Then it reverses direction immediately without first going all the way to the end of the disk. The version of scan and c scan is called as look and c-look.



DISK SCHEDULING EXAMPLE

The head of a moving head disk with 100 track numbered 0 to 99 is currently serving a request at track 55. If the queue of request kept in FIFO order is 10, 70,75,23,65 which of the disk scheduling algorithm FCFS and SSTF will require less head movement? find the total head movement for each algorithm.

<u>Solution:</u> in the queue 10, 70, 75, 23, 65, 10 must be served first and 65 to be served last.

<u>Case 1 FIFO:</u>The head moves from track 55 to 10, 10 to 70, 70 to 75, 75 to 23, and 23 to 65.

Hence the number of tracks the head moved in total

(55-10)+(70-10)+(75-70)+(75-23)+(65-23)

45 + 60 + 5 +52 +42 =204

<u>Case 2 :</u> In SSTF the head moves from 55 to 65, 65 to 70, 70 to 75 , 75 to 23 ,23 to 10

= (65-55)+(70-65)+(75-70)+(75-23)+(23-10)

10 + 5 +5 +52 +13 = 85

The above elements are chosen from following distance matrix chose minimum element in each row.

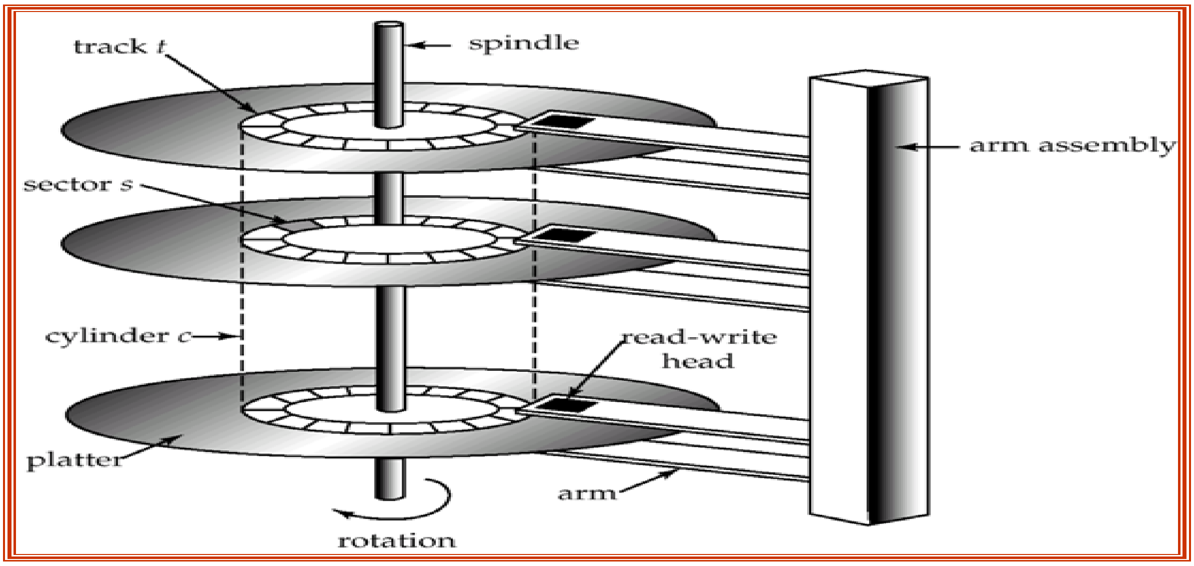
Present track	10	70	75	23	65	Next track
55	45	15	20	32	10	65
65	55	5	10	42	-	70
70	60	-	5	47	-	75
75	65	-	-	52	-	23
23	13	-	-	-	-	10

Hence the number of tracks the head moved in total=10+5 +52 +13=80

Therefore clearly SSTF is better than FIFO.

DISK ORGANIZATION

Information on the disk is referenced by multiport address, which includes the drive number, surface and track. Collection of track on all surfaces that are at the same distance is called a cylinder.



In the above fig. 3 platter and 6 recording surfaces (2 on each platter).

Therefore, each of its cylinder consist of 6 tracks. (One track from each surface). The number of distinct track on single surface determines the total number of cylinder on given disk.

The disk surface is divided into tracks which are further divided into sectors. Information is organized in the form of blocks within a track whose size is equivalent to sector. Each sector can be separately read or written.

The second approach to organize information on track is in the form of variable length. A sector is the smallest unit of information that can be read from or written to the disk. Depending upon the disk drive, sector varies from 32 bytes to 4096 bytes. There are 4 to 32 sector per track and from 75 to 500 tracks per disk surface. To access a sector we must specify the surface, track and sector read /write heads.

The read/write heads are moved to the correct track called as seek time. Electronically switched to the correct surface and then wait (latency time) for the requester sector to rotate below head.

Disk management

- **The operating system may also be responsible for task such as disk formatting, booting from disk, and bad-block recovery.**
- **Low –level formatting: divides the disk into sectors, and is usually performed when the disk is manufactured.**
- **Logical formatting creates a file system on the disk, and is done by the OS.**
- **The OS maintains the boot blocks (or boot partition) that contains the bootstrap loader.**
- **Bad blocks: disk blocks may fail.**
  - **An error correcting code (ECC) stored with each block can detect possibly correct an error (if so, it is called as soft error).**
  - **Disk contain spare sectors which are substituted for bad sectors.**
  - **If the system cannot recover from the error ,it is called as hard error, and manual intervention may be required**

Disk formatting :

Disk formatting is the configuring process of a data storage media such as hard disk drive, floppy disk or flash drive for initial usage. Any existing file on the drive would be erased with disk formatting. A disk formatting is usually done before initial installation or before installation of a new operating system. Disk formatting is also done if there is a requirement for additional storage in the computer.

Disk formatting can be performed on both magnetic platter hard drives and solid state drive. The formatting comprises low level formatting, partition, high level formatting. Low level formatting aids in preparing the physical structure on the storage media. The partition process involves the division of the hard drive into logical volume of data storage. High level formatting helps in creating the file system format within the logical volume or within the disk partition.

Disk formatting is usually done with the help of disk formatting utility. While preparing the hard drive for initial use, disk formatting checks for the error in the drive. It can scan and repair bad sectors. Another benefit associated with disk formatting is its capability to erase bad applications and remove sophisticated viruses. Disk formatting is an action which must be done with caution. As it deletes data and remover program installed backup of the necessary data or application are required. It takes time frequent disk formatting can gradually decrease the life of a hard drive.

BOOT BLOCK:

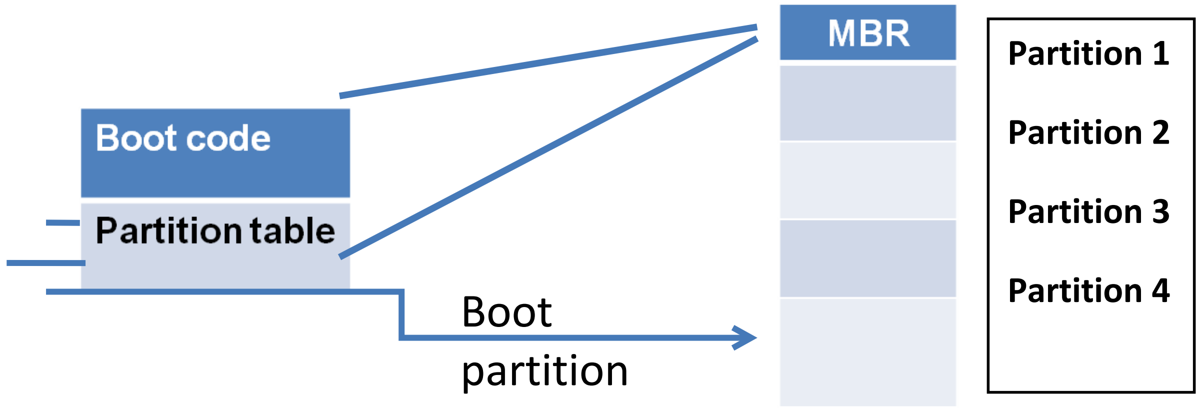
A program on a hard disk, floppy disk or other media which is loaded when the computer is turned on or rebooted and which controls the next phase of loading the actual operating system. The loading and execution of the boot block is usually controlled by firmware in ROM or PROM. It may be at some fixed location possibly or maybe pointed to by the master boot record. A boot sector or boot block is a region of hard disk, floppy disk, optical disc, or other data storage device that contains machine code to be loaded into random-access memory (RAM) by a computer systems built–in firmware. The purpose of a boot sector is to allow the boot process of a computer to load a program (usually but not necessarily, an operating system) stored on the same storage device. The location and size of the boot sector (perhaps corresponding to a logical disk sector) is specified by the design of the computing platform.

KINDS OF BOOT SECTORS :

Several major kinds of boot sectors could be encountered on the IBM PC compatible and similar storage device:

A master boot record (MBR) is the first sector of a data storage device that have been partitioned. The MBR sector may contain code to locate the active partition and invoke its volume boot record.

A volume boot record (VBR) is the first sector of a data storage device that has not been partition. It may contain code to load an operating system (or other standalone program) installed on that device or within the partition.



Bad block: a bad block is a sector on a computer's disk drive or flash memory that is either inaccessible or un-writeable due to permanent damage, such as physical damage to the disk surface or failed flashed memory transistors. By sector are usually detected by disk utility software such as CHKDSK or SCANDISK on Microsoft system or bad block on Unix – like systems. When found these programs may mark the sectors unusable and the operating system skips them in the future.

If any of the files uses a sector which is marked as 'bad' by a disk utility then the bad sector of the file is remapped to a free sector and unreadable data is lost. To avoid file corruption data recovery method should be performed 1st if bad sectors are found by OS as file system level. When a sector is found to be bad or un-stable by the firmware of a disk controller, the disk controller remaps the logical sector to be different physical sector. In the normal operation of a hard drive, the detection and remapping of bad sector should take place in a manner of transparent to the rest of the system in advance before data is lost. However, the damage to the physical body of the hard drive does not solely affect one area of the data stored.

SWAP –SPACE MANAGEMENT:

Swap space management is another low level task of the operating system. Virtual memory uses disk space as an extension of main memory. Since disk access is much slower than memory access, using swap space significantly decreases system performance. The main goal for the design and implementation of swap space is to provide the best throughput for the virtual memory system.

SWAP-SPACE USE:

Swap space is used in various ways by different OS's depending on the memory management algorithm in use. For instance, system that implements swapping may use swap space to hold an entire process image, including the code and data segments. Paging systems may simply store pages that have been pushed out of main memory. The amount of swap space needed on a system can therefore vary depending on the amount of physical memory, the amount of virtual memory it is backing and the way in which virtual memory is used. It can range from a few megabytes of a disk space to gigabytes.

SWAP-SPACE LOCATION:

A swap can reside in one of the two places:

It can be carved out of the normal file system, or it can be in separate disk partition. If the swap space is simply a large file within the file system, normal file system routines can be used to create it, name it, and allocate its space. Alternatively swap space can be created in a separate raw partition, as no file system or directory structure is placed in this space.