

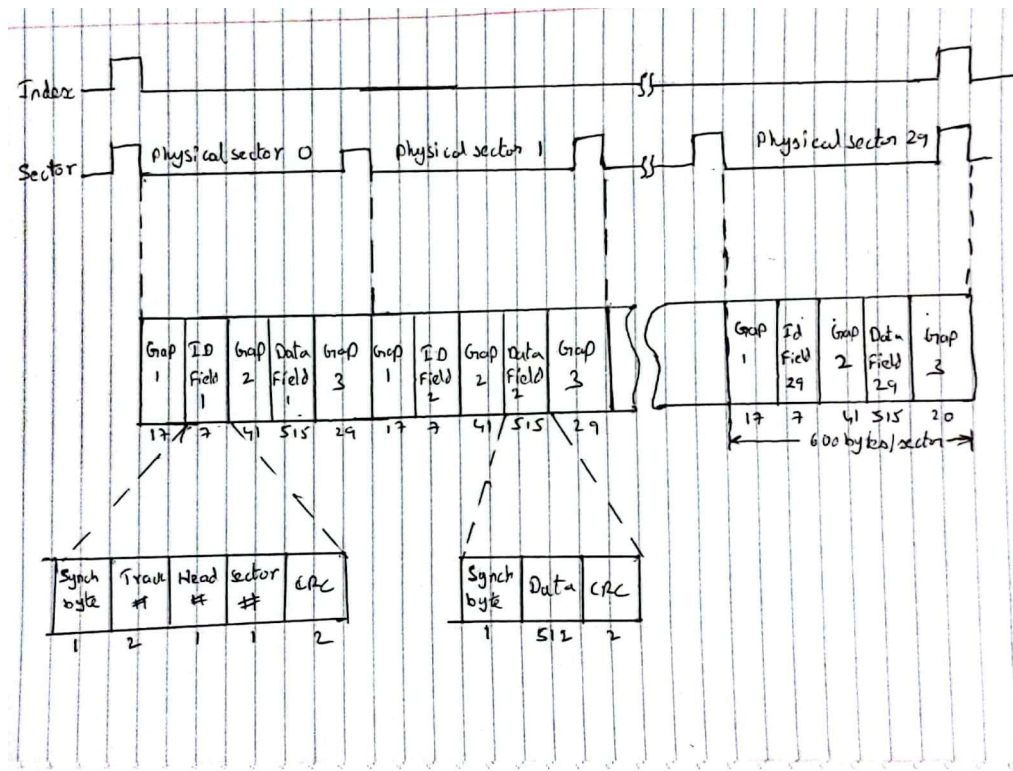
1.

A **track** is a circular path on the surface of a disk on which information is magnetically recorded. There are thousands of tracks per surface. Adjacent tracks are separated by intertrack gaps.

A **Sector** is subdivision of a track on a magnetic disk. Each sector stores a particular amount of user accessible data. Data are transferred to and from the disk in sectors. There are typically hundreds of sectors per track, and these may be of either fixed or variable length.

Some disk drives accommodate multiple platters stacked vertically a fraction of an inch apart. Multiple-platter disks employ a movable head, all of the heads are mechanically fixed so that all are at the same distance from the center of the disk and move together. Thus, at any time, all of the heads are positioned over tracks that are of equal distance from the center of the disk. The set of all the tracks in the same relative position on the platter is referred to as a **cylinder**.

Winchester Disk Format:



2.

To read or write, the head must be positioned at the desired track and at the beginning of the desired sector on that track, once the track is selected, the disk controller waits until the appropriate sector rotates to line up with the head. The time it takes for the beginning of the sector to reach the head is known as **rotational delay** or *rotational latency*. the time it takes to position the head at the track is known as seek time. The sum of the seek time and the rotational delay equals the **access time**. Access time is the time it takes to get into position to read or write. After the data has been accessed and read by the head, the time it takes to transfer the portion of the data from the disk is called **transfer time**.

The CDROM runs on CLV. With the simplest CAV system, the number of bits per track is constant. MZR is used to achieve a high density. In MZR the surface is divided into a number of zones and zones farther from the center contains more bits than zones closer to the center. Although this technique increases capacity, it is still not optimal. To achieve greater capacity, CDs and CD-ROMs do not organize information on concentric tracks. Instead, the disk contains a single spiral track, beginning near the center and spiraling out to the outer edge of the disk. Sectors are the same length outside and inside and information is packed evenly across the disk in segments of the same size and these are scanned at the same rate by rotating the disk at a variable speed. The pits are then read by the laser at a constant linear velocity (CLV). The disk rotates more slowly for accesses near the outer edge than for those near the center. The capacity of a track and the rotational delay both increase for positions nearer the outer edge of the disk.

The DVD's greater capacity is due to three differences from CD's

- Bits are packed more closely on a DVD. The DVD uses a laser with shorter wavelength and achieves a track spacing of 0.74 mm and a minimum distance between pits of 0.4 mm. The result of these two improvements is about a seven-fold increase in capacity, to about 4.7 GB.
- The DVD employs a second layer of pits and lands on top of the first layer. A dual-layer DVD has a semireflective layer on top of the reflective layer, and by adjusting focus, the lasers in DVD drives can read each layer separately. This technique almost doubles the capacity of the disk.
- The DVD-ROM can be two sided, whereas data are recorded on only one side of a CD. This brings total capacity up to 17 GB.

3.

RAID 0:

For RAID 0, the user and system data are distributed across all of the disks in the array. The data are *striped* across the available disks. The advantage of this layout is that if a single I/O request consists of multiple logically contiguous strips, then up multiple strips for that request can be handled in parallel. It doesn't consist of a mirrored disk and hence there is no concept of synchronization. To achieve a high data transfer rate with RAID-0, First, a high transfer capacity must exist along the entire path between host memory and the individual disk drives. Application must make I/O requests that drive the disk array efficiently. This requirement is met if the typical request is for large amounts of logically contiguous data, compared to the size of a strip. In this case, a single I/O request involves the parallel transfer of data from multiple disks, increasing the effective transfer rate compared to a single-disk transfer. If the strip size is relatively large, so that a single I/O request only involves a single disk access, then multiple waiting I/O requests can be handled in parallel, reducing the queuing time for each request. Applications of RAID 0 include supercomputers, gaming, graphic applications, Image editing and video production.

RAID 1:

In RAID 1, redundancy is achieved by the simple expedient of duplicating all the data. Data striping is used, as in RAID 0. Each logical strip is mapped to two separate physical disks so that every disk in the array has a mirror disk that contains the same data. A read request can be serviced by either of the two disks that contains the requested data. A write request requires that both corresponding strips be updated, but this can be done in parallel. Since we need to write the data to the mirrored disk as well both the disks should be synchronized. When a drive fails, the data may still be accessed from the second drive. RAID 1 is the costly since it requires two disks. In a transaction-oriented environment, RAID 1 can achieve high I/O request rates if the bulk of the requests are reads. RAID provide improved performance over RAID 0 for data transfer intensive applications with a high percentage of reads. Improvement occurs if the application can split each read request so that both disk members participate. Applications include drives that store system software and data and highly critical files, Accounting, Payroll and Financial.

#### RAID 2:

RAID levels 2 and 3 make use of a parallel access technique, where all member disks participate in the execution of every I/O request. Typically, the spindles of the individual drives are synchronized so that each disk head is in the same position on each disk at any given time. Data striping is used. In RAID 2 and 3, the strips are very small, often as small as a single byte or word. An error-correcting code is calculated across corresponding bits on each data disk, and the bits of the code are stored in the corresponding bit positions on multiple parity disks. On a single read, all disks are simultaneously accessed. The requested data and the associated error-correcting code are delivered to the array controller. On a single write, all data disks and parity disks must be accessed for the write operation. RAID 2 is costlier than RAID 1. Applications include environment in which many disk errors occur.

#### RAID 3:

RAID 3 is similar to RAID 2, the difference is that RAID 3 requires only a single redundant disk. RAID 3 employs parallel access, with data distributed in small strips. Instead of an error-correcting code, a simple parity bit is computed for the set of individual bits in the same position on all of the data disks. In the event of a disk failure, all of the data are still available in a reduced mode. For reads, the missing data are regenerated using the exclusive-OR calculation. When data are written to a reduced RAID 3 array, consistency of the parity must be maintained for later regeneration, this is synchronization. Because data are striped in very small strips, RAID 3 can achieve very high data transfer rates. Any I/O request will involve the parallel transfer of data from all of the data disks. Only one I/O request can be executed at a time. Applications include video editing and image editing.

#### RAID 4:

RAID levels 4 through 6 make use of an independent access technique. Each member disk operates independently, so that separate I/O requests can be satisfied in parallel. Because of this, independent access arrays are more suitable for applications that require high I/O request rates and are relatively less suited for applications that require high data transfer rates. Data striping is used, and in the case of RAID 4 through 6, the strips are relatively large. With RAID 4, a bit-by-bit parity strip is calculated across corresponding strips on each data disk, and the parity bits are stored in the corresponding strip on the parity disk. In the case of a larger size I/O write, the parity drive can be updated in parallel with the data drives and there are no extra reads or writes. Every write operation must involve the parity disk, which therefore can become a bottleneck. No real life applications for RAID 4.

#### RAID 5:

RAID 5 is organized in a similar fashion to RAID 4. The difference is that RAID 5 distributes the parity strips across all disks. A typical allocation is a round-robin scheme. For an  $n$ -disk array, the parity strip is on a different disk for the first  $n$  stripes, and the pattern then repeats. The distribution of parity strips across all drives avoids the potential I/O bottle-neck found in RAID 4. Make use of an independent access technique so, suitable for applications that require high I/O request rates and are relatively less suited for applications that require high data transfer rates. Data striping is used, the strips are relatively large. The Strip size are relatively large. Applications include Database servers, web and news servers.

#### RAID 6:

In RAID 6, two different parity calculations are carried out and stored in separate blocks on different disks. A RAID 6 array whose user data require  $N$  disks consists of  $N + 2$  disks, as a result, data can be recreated even if two drives containing user data is corrupted. Make use of an independent access technique so, suitable for applications that require high I/O request rates and are relatively less suited for applications that require high data transfer rates. Data striping is used, the strips are relatively large. The advantage of RAID 6 is that it provides extremely high data availability. Three disks would have to fail within the MTTR interval to cause data to be lost. RAID 6 incurs a substantial write penalty which is disadvantage. Applications include healthcare, banking and defence sectors.

4.

RAID 5 make use of an independent access technique so, suitable for applications that require high I/O request rates and are relatively less suited for applications that require high data transfer rates. Data striping is used, the strips are relatively large.

When a single process asks for a large data sets:

Generally the strips in RAID 5 are large but since we are accessing a large data set, it means the data set is larger than the strip size and the data set will be present in multiple disks. We can read the strips present on multiple disks in parallel, since RAID 5 uses independent access technique. Since we are reading the strips in parallel the effective transfer time would be the product of no of disks and the slowest effective access time among all the disks. In this case the data rate is high. If we have a case where the strip size is very large compared to the data set which is not mentioned in the question, but if we have a very large strip size the data set will fall only in one disk then the data rate will be same as single disk.

When Multiple Processes request several small sized data:

Since the strip size of RAID 5 is large, so the data requested may be located on the same disk. We need to make sure strip size is large enough such that the whole data is present in a single disk, or else the request will be transferred to multiple disks increasing the transfer time, in this case the data rate is high. If stripe is large enough and data is present only in one disk, multiple processes can access the data in parallel reducing overall transfer time, in this case data rate is low.

5.

Flash memory is a type of non volatile memory that erases data in units called blocks and rewrites data at the byte level. Flash memory is widely used for storage and data transfer in consumer devices, enterprise systems and industrial applications. Flash memory retains data for an extended period of time, regardless of whether a flash-equipped device is powered on or off. Flash memory is intermediate between EPROM and EEPROM in both cost and functionality. Like EEPROM, flash memory uses an electrical erasing technology. An entire flash memory can be erased in one or a few seconds. It is possible to erase just blocks of memory rather than an entire chip. Flash memory does not provide byte-level erasure. Flash memory uses only one transistor per bit, and so achieves the high density. An important characteristic of flash memory is that it is persistent memory, which means that it retains data when there is no power applied to the memory. Flash memory uses transistor logic and a second gate called floating gate is added to the transistor. Initially, the floating gate does not interfere with the operation of the transistor, this state represents a binary 1. Applying a large voltage across the oxide layer of floating gate causes electrons to tunnel through it and become trapped on the floating gate, where they remain even if the power is disconnected, and this represents binary 0.

Flash memory's lifespan is limited. Flash memory becomes unusable after a certain number of writes. Flash memory cells have a finite lifespan because they lose their capacity to store data after a specific number of write operations, as flash cells are stressed. A typical limit is 100,000 writes.

Techniques for prolonging the life of an SSD drive include front-ending the flash with a cache to delay and group write operations, using wear-leveling algorithms that evenly distribute writes across block of cells, and sophisticated bad block management techniques. Another approach is to perform write verification and remapping to spare sectors in case of write failure, a technique called bad block management (BBM). Most flash devices are also capable of estimating their own remaining lifetimes so systems can anticipate failure and take preemptive action.

