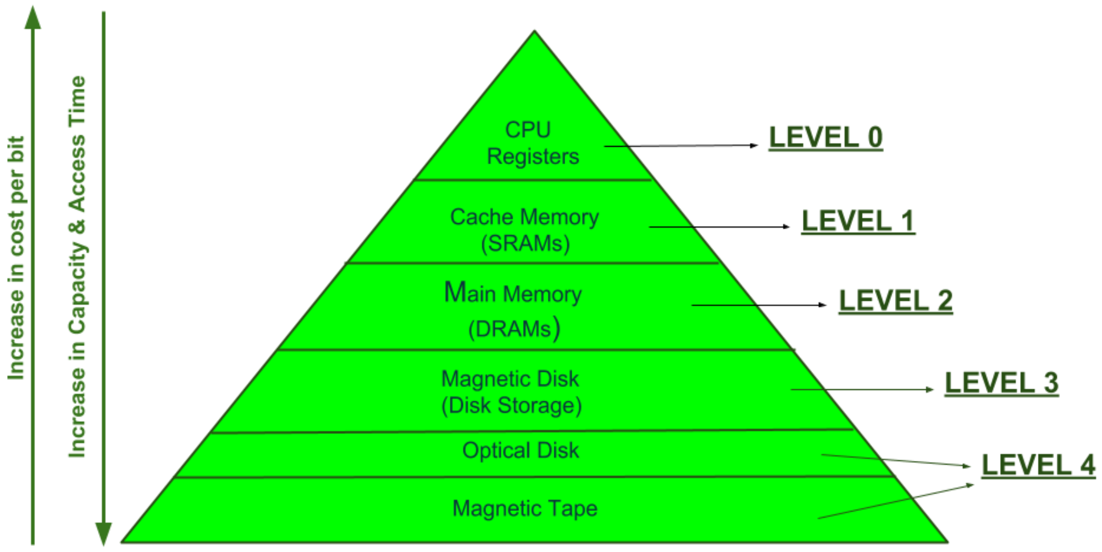
Memory Hierarchy in Computer Architecture

In the Computer System Design, Memory Hierarchy is an enhancement to organize the memory such that it can minimize the access time. The Memory Hierarchy was developed based on a program behavior known as locality of references. The figure below clearly demonstrates the different levels of memory hierarchy :



MEMORY HIERARCHY DESIGN

This Memory Hierarchy Design is divided into 2 main types:

1. External Memory or Secondary Memory –
   The secondary memory is also known as external memory, and this is accessible by the processor through an input/output module. This memory includes an optical disk, magnetic disk, and magnetic tape.
2. Internal Memory or Primary Memory –
   The primary memory is also known as internal memory, and this is accessible by the processor straightly. This memory includes main, cache, as well as CPU registers.

We can infer the following characteristics of Memory Hierarchy Design from above figure:

1. **Capacity:**
   It is the global volume of information the memory can store. As we move from top to bottom in the Hierarchy, the capacity increases.

2. **Access Time:**
   It is the time interval between the read/write request and the availability of the data. As we move from top to bottom in the Hierarchy, the access time increases.
3. **Performance:**
   Earlier when the computer system was designed without Memory Hierarchy design, the speed gap increases between the CPU registers and Main Memory due to large difference in access time. This results in lower performance of the system and thus, enhancement was required. This enhancement was made in the form of Memory Hierarchy Design because of which the performance of the system increases. One of the most significant ways to increase system performance is minimizing how far down the memory hierarchy one has to go to manipulate data.
4. **Cost per bit:**
   As we move from bottom to top in the Hierarchy, the cost per bit increases i.e. Internal Memory is costlier than External Memory.

SRAM

SRAM is a type of semiconductor memory that uses Bistable latching circuitry to store each bit. In this type of RAM, data is stored using the six transistor memory cell. Static RAM is mostly used as a cache memory for the processor (CPU). SRAM is relatively faster than other RAM types, such as DRAM. It also consumes less power. The full form of SRAM is Static Random Access Memory.

DRAM

It is a type of RAM which allows you to stores each bit of data in a separate capacitor within a particular integrated circuit.It is a standard computer memory of any modern desktop computer. The full form of DRAM is Dynamic Random Access Memory.

DRAM is constructed using capacitors and a few transistors. In this type of RAM, the capacitor is used for storing the data where bit value, which signifies that the capacitor is charged and a bit value 0, which means that the capacitor is discharged.

Characteristics of SRAM

Here, are important characteristics of SRAM

- SRAM is faster than DRAM
- Several times more expensive than DRAMs
- Takes up much more space than DRAMs
- Consume less power than DRAMs
- Usage: level 1 or level 2 cache
- Cycle time is much shorter compared to DRAM because it does not require to pause between accesses.
- It is often used only as a memory cache

Characteristics of DRAM

Here, are important characteristics of DRAM

- Cost-effective
- It has a short data lifetime
- Requires to refresh
- Slower compared to SRAM
- More power consumption

Associative Memory

Associative memory searches stored data only by the data value itself rather by an address. This type of search helps in reducing the search time by a large extent.

What is associative memory?
When data is accessed by data content rather than data address, then the memory is referred to as associative memory or content addressable memory.
How associative memory works?

Given below is a series of steps that depicts working of associative memory in computer architecture:

- Data is stored at the very first empty location found in memory.
- In associative memory when data is stored at a particular location then no address is stored along with it.

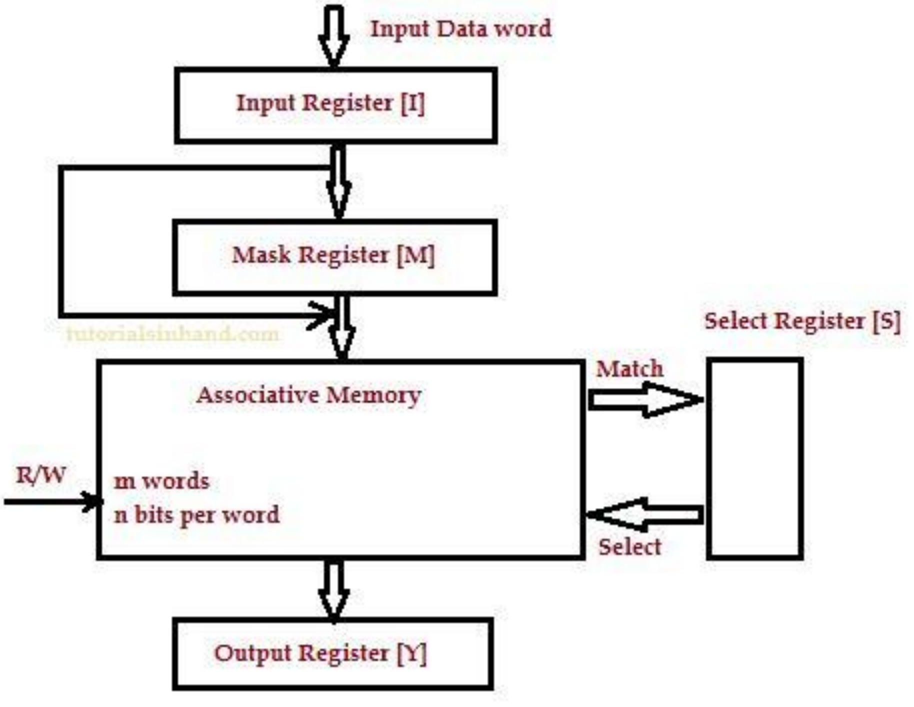
- When the stored data need to be searched then only the key (i.e. data or part of data) is provided.
- A sequential search is performed in the memory using the specified key to find out the matching key from the memory.
- If the data content is found then it is set for the next reading by the memory.

Associative memory organization

The associative memory hardware structure consists of:

- memory array,
- logic for m words with n bits per word, and
- several registers like input register, mask register, select register and output register.

The block diagram showing organization of associative memory is shown below:



Block Diagram of Associative Memory

<u>Functions of the registers</u> used in associative memory is given below:

- Input Register (I) hold the data that is to be written into the associative memory. It is also used to hold the data that is to be searched for. At a particular time it can hold a data containing one word of length (say n).
- Mask Register (M) is used to provide a mask for choosing a key or particular field in the input register's word. Since input register can hold a data of one word of length n so the maximum length of mask register can be n.

- Select Register (S) contains m bits, one for each memory words. When input data in I register is compared to key in m register and match is found then that particular bit is set in select register.
- Output Register (Y) contains the matched data word that is retrieved from associative memory.

Advantages of associative memory

- Associative memory searching process is fast.
- Associative memory is suitable for parallel searches.

Disadvantages of associative memory

- Associative memory is expensive than RAM

Associative Memory

It is also known as content addressable memory (CAM). It is a memory chip in which each bit position can be compared. In this the content is compared in each bit cell which allows very fast table lookup. Since the entire chip can be compared, contents are randomly stored without considering addressing scheme. These chips have less storage capacity than regular memory chips.

Auxiliary Memory

Devices that provide backup storage are called auxiliary memory. For example: Magnetic disks and tapes are commonly used auxiliary devices. Other devices used as auxiliary memory are magnetic drums, magnetic bubble memory and optical disks.

It is not directly accessible to the CPU, and is accessed using the Input/Output channels.

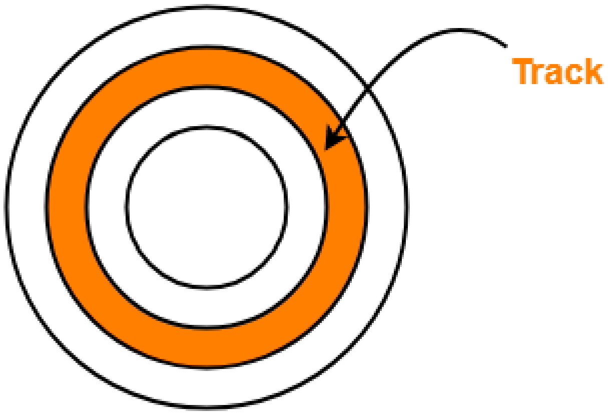
Magnetic Disk in Computer Architecture-

In computer architecture,

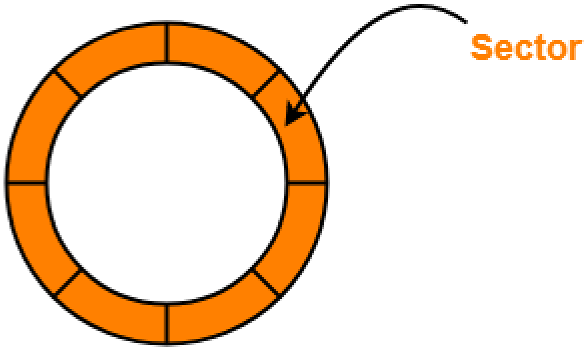
- Magnetic disk is a storage device that is used to write, rewrite and access data.
- It uses a magnetization process.

Architecture-

- The entire disk is divided into platters.
- Each platter consists of concentric circles called as tracks.
- These tracks are further divided into sectors which are the smallest divisions in the disk.

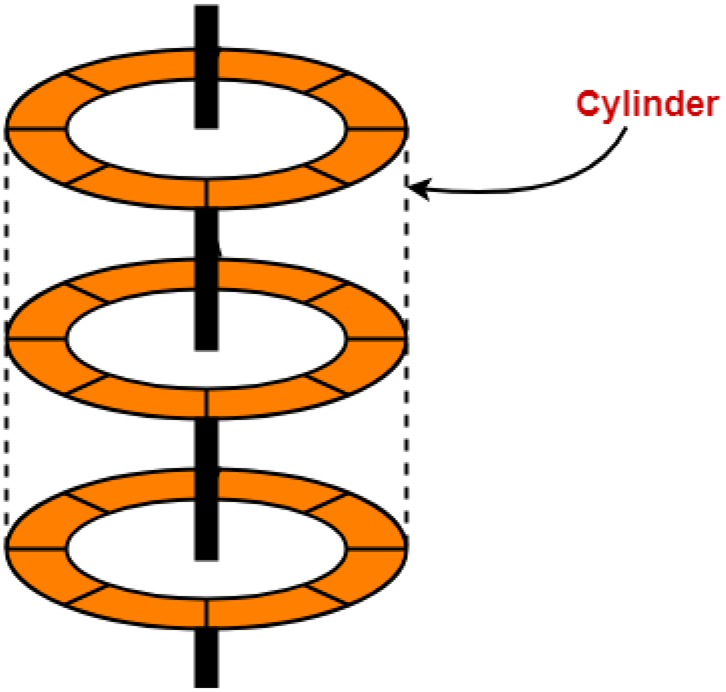


Disk divided into tracks

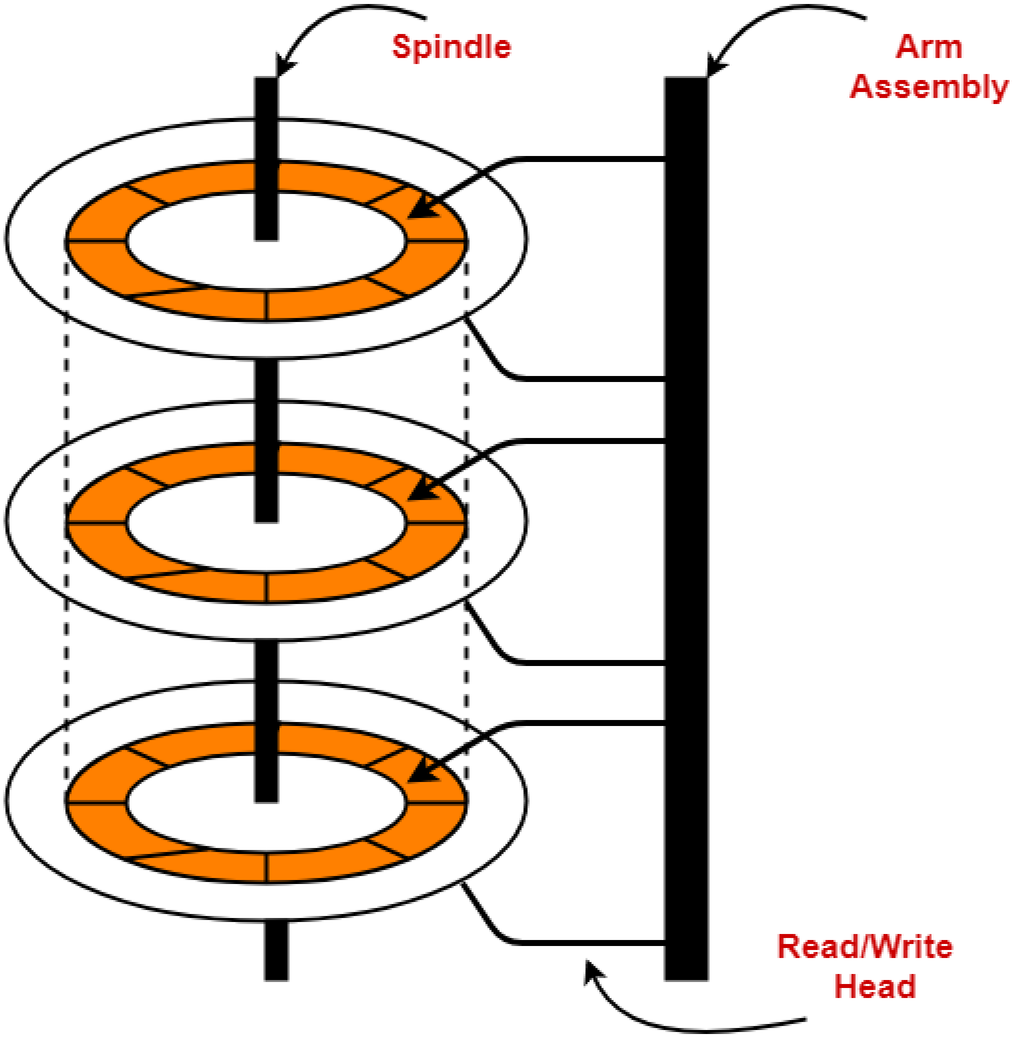


Track divided into sectors

- A cylinder is formed by combining the tracks at a given radius of a disk pack.



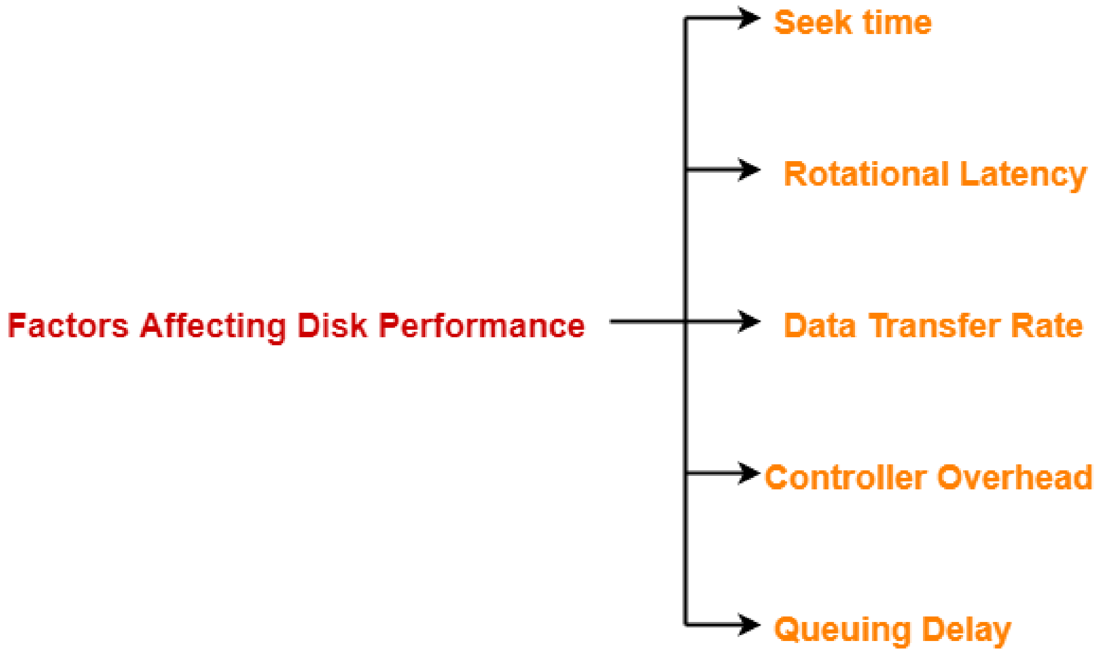
- There exists a mechanical arm called as Read / Write head.
- It is used to read from and write to the disk.
- Head has to reach at a particular track and then wait for the rotation of the platter.
- The rotation causes the required sector of the track to come under the head.
- Each platter has 2 surfaces- top and bottom and both the surfaces are used to store the data.
- Each surface has its own read / write head.



Disk Performance Parameters-

The time taken by the disk to complete an I/O request is called as disk service time or disk access time.

Components that contribute to the service time are-



1. Seek time
2. Rotational latency
3. Data transfer rate
4. Controller overhead
5. Queuing delay

1. Seek Time-

- The time taken by the read / write head to reach the desired track is called as seek time.
- It is the component which contributes the largest percentage of the disk service time.
- The lower the seek time, the faster the I/O operation.

Specifications

Seek time specifications include-

1. Full stroke
2. Average
3. Track to Track

1. Full Stroke-

- It is the time taken by the read / write head to move across the entire width of the disk from the innermost track to the outermost track

2. Average-

- It is the average time taken by the read / write head to move from one random track to another.

Average seek time = 1 / 3 x Full stroke

3. Track to Track-

- It is the time taken by the read-write head to move between the adjacent tracks.

2. Rotational Latency-

- The time taken by the desired sector to come under the read / write head is called as rotational latency.
- It depends on the rotation speed of the spindle.

Average rotational latency = 1 / 2 x Time taken for full rotation

3. **Data Transfer Rate-**

- The amount of data that passes under the read / write head in a given amount of time is called as data transfer rate.
- The time taken to transfer the data is called as transfer time.

It depends on the following factors-

1. Number of bytes to be transferred
2. Rotation speed of the disk
3. Density of the track
4. Speed of the electronics that connects the disk to the computer

4. **Controller Overhead-**

- The overhead imposed by the disk controller is called as controller overhead.
- Disk controller is a device that manages the disk.

5. **Queuing Delay-**

- The time spent waiting for the disk to become free is called as queuing delay.

NOTE-

All the tracks of a disk have the same storage capacity.

Storage Density-

- All the tracks of a disk have the same storage capacity.
- This is because each track has different storage density.
- Storage density decreases as we from one track to another track away from the center.

Thus,

- Innermost track has maximum storage density.
- Outermost track has minimum storage density.

Important Formulas-

1. **Disk Access Time-**

Disk access time is calculated as-

Disk access time

= Seek time + Rotational delay + Transfer time + Controller overhead + Queuing delay

2. **Average Disk Access Time-**

Average disk access time is calculated as-

Average disk access time

= Average seek time + Average rotational delay + Transfer time + Controller overhead + Queuing delay

3. **Average Seek Time-**

Average seek time is calculated as-

Average seek time

= 1 / 3 x Time taken for one full stroke

Alternatively,

If time taken by the head to move from one track to adjacent track = t units and there are total k tracks, then-

Average seek time

= { Time taken to move from track 1 to track 1 + Time taken to move from track 1 to last track } / 2

= { 0 + (k-1)t } / 2

= (k-1)t / 2

4. **Average Rotational Latency-**

Average rotational latency is calculated as-

Average rotational latency

= 1 / 2 x Time taken for one full rotation

Average rotational latency may also be referred as-

- Average rotational delay

- Average latency
- Average delay

5. Capacity Of Disk Pack-

Capacity of a disk pack is calculated as-

Capacity of a disk pack

= Total number of surfaces x Number of tracks per surface x Number of sectors per track x Storage capacity of one sector

6. Formatting Overhead-

Formatting overhead is calculated as-

Formatting overhead

= Number of sectors x Overhead per sector

7. Formatted Disk Space-

Formatted disk space also called as usable disk space is the disk space excluding formatting overhead.

It is calculated as-

Formatted disk space

= Total disk space or capacity – Formatting overhead

8. **Recording Density Or Storage Density-**

Recording density or Storage density is calculated as-

Storage density of a track

= Capacity of the track / Circumference of the track

From here, we can infer-

Storage density of a track ∝ 1 / Circumference of the track

9. **Track Capacity-**

Capacity of a track is calculated as-

Capacity of a track

= Recording density of the track x Circumference of the track

10. Data Transfer Rate-

Data transfer rate is calculated as-

Data transfer rate

= Number of heads x Bytes that can be read in one full rotation x Number of rotations in one second

OR

Data transfer rate

= Number of heads x Capacity of one track x Number of rotations in one second

11. Tracks Per Surface-

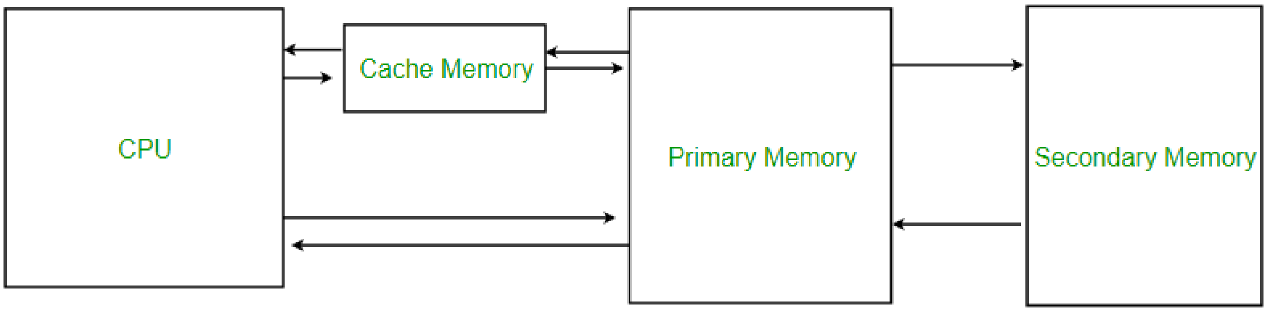
Total number of tracks per surface is calculated as-

Total number of tracks per surface

= (Outer radius – Inner radius) / Inter track gap

Cache Memory in Computer Organization

Cache Memory is a special very high-speed memory. It is used to speed up and synchronizing with high-speed CPU. Cache memory is costlier than main memory or disk memory but economical than CPU registers. Cache memory is an extremely fast memory type that acts as a buffer between RAM and the CPU. It holds frequently requested data and instructions so that they are immediately available to the CPU when needed.
Cache memory is used to reduce the average time to access data from the Main memory. The cache is a smaller and faster memory which stores copies of the data from frequently used main memory locations. There are various different independent caches in a CPU, which store instructions and data.



Levels of memory:

- Level 1 or Register –
  It is a type of memory in which data is stored and accepted that are immediately stored in CPU. Most commonly used register is accumulator, Program counter, address register etc.
- Level 2 or Cache memory –
  It is the fastest memory which has faster access time where data is temporarily stored for faster access.
- Level 3 or Main Memory –
  It is memory on which computer works currently. It is small in size and once power is off data no longer stays in this memory.
- Level 4 or Secondary Memory –
  It is external memory which is not as fast as main memory but data stays permanently in this memory.

Cache Performance:

When the processor needs to read or write a location in main memory, it first checks for a corresponding entry in the cache.

- If the processor finds that the memory location is in the cache, a cache hit has occurred and data is read from cache
- If the processor does not find the memory location in the cache, a cache miss has occurred. For a cache miss, the cache allocates a new entry and copies in data from main memory, then the request is fulfilled from the contents of the cache.

The performance of cache memory is frequently measured in terms of a quantity called Hit ratio.

Hit ratio = hit / (hit + miss) = no. of hits/total accesses

We can improve Cache performance using higher cache block size, higher associativity, reduce miss rate, reduce miss penalty, and reduce the time to hit in the cache.