
RGPVNOTES.IN

Program : **B.Tech**

Subject Name: **Computer System Organisation**

Subject Code: **EC-504**

Semester: **5th**

# Unit IV

## Memory Organization

### 4.1 Memory Maps

A memory map is a massive table, in effect a database, that comprises complete information about how the memory is structured in a computer system. A memory map works something like a gigantic office organizer. In the map, each computer file has a unique memory address reserved especially for it, so that no other data can inadvertently overwrite or corrupt it.

In order for a computer to function properly, its OS (operating system) must always be able to access the right parts of its memory at the right times. When a computer first boots up (starts), the memory map tells the OS how much memory is available. As the computer runs, the memory map ensures that data is always written to, and read from, the proper places. The memory map also ensures that the computer's debuggers can resolve memory addresses to actual stored data.

If there were no memory map, or if an existing memory map got corrupted, the OS might (and probably would) write data to, and read data from, the wrong places. As a result, when data was read, it would not always pertain to the appropriate files or application programs. The problem would likely start out small and unnoticeable, worsen with time, and become apparent only after considerable damage had been done to stored data and programs. In the end, some or all of the applications would fail to run, and many critical data files would be ruined.

### 4.2 Memory Hierarchy

Memory hierarchy is a concept that is necessary for the CPU to be able to manipulate data.

Computer memory is classified in the below hierarchy.

1. Internal register is for holding the temporary results and variables. Accessing data from these registers is the fastest way of accessing memory.

2. Cache is used by the CPU for memory which is being accessed over and over again. Instead of pulling it every time from the main memory, it is put in cache for fast access. It is also a smaller memory, however, larger than internal register.

Cache is further classified to L1, L2 and L3:

a) L1 cache: It is accessed without any delay.

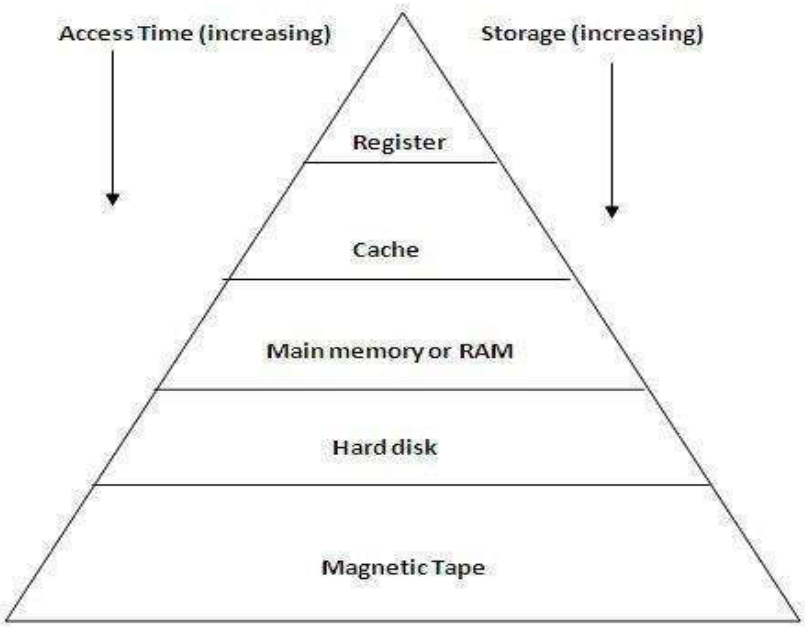
b) L2 cache: It takes more clock cycles to access than L1 cache.

c) L3 cache: It takes more clock cycles to access than L2 cache.

3) Main memory or RAM (Random Access Memory): It is a type of the computer memory and is a hardware component. It can be increased provided the operating system can handle it.

4) Hard disk: A hard disk is a hardware component in a computer. Data is kept permanently in this memory. Memory from hard disk is not directly accessed by the CPU, hence it is slower. As compared with RAM, hard disk is cheaper per bit.

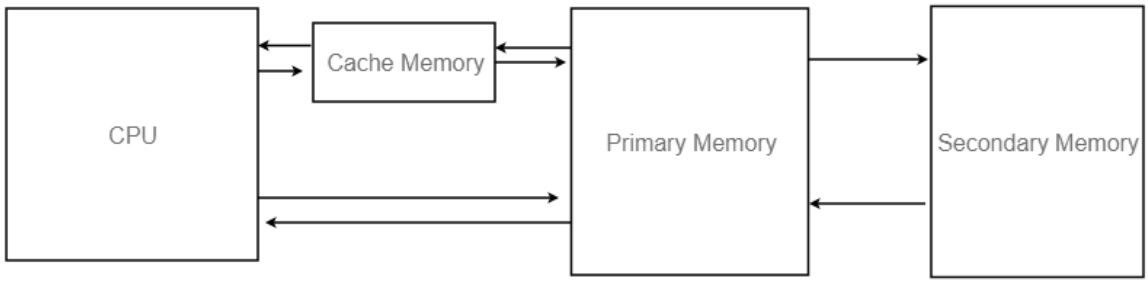
5) Magnetic tape: Magnetic tape memory is usually used for backing up large data. When the system needs to access a tape, it is first mounted to access the data. When the data is accessed, it is then unmounted. The memory access time is slower in magnetic tape and it usually takes few minutes to access a tape.




## 4.3 Cache Memory- Organization and Mappings

As CPU has to fetch instruction from main memory speed of CPU depending on fetching speed from main memory. CPU contains register which has fastest access but they are limited in number as well as costly. Cache is cheaper so we can access cache. Cache memory is a very high speed memory that is placed between the CPU and main memory, to operate at the speed of the CPU.

It 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. Most CPUs have different independent caches, including instruction and data.




Types of Cache

- Primary Cache – A primary cache is always located on the processor chip. This cache is small and its access time is comparable to that of processor registers.
- Secondary Cache – secondary cache is placed between the primary cache and the rest of the memory. It is referred to as the level 2 (L2) cache. Often, the Level 2 cache is also housed on the processor chip.

Locality of reference

Since size of cache memory is less as compared to main memory. So to check which part of main memory should be given priority and loaded in cache is decided based on locality of reference.

Types of Locality of reference

1. Spatial Locality of reference – this says that there is chance that element will be present in the close proximity to the reference point and next time if again searched then more close proximity to the point of reference.
2. Temporal Locality of reference – In this Least recently used algorithm will be used. Whenever there is page fault occurs within word will not only load word in main memory but complete page fault will be loaded because spatial locality of reference rule says that if you are referring any word next word will be referred in its register that's why we load complete page table so complete block will be loaded.

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 chache
- 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

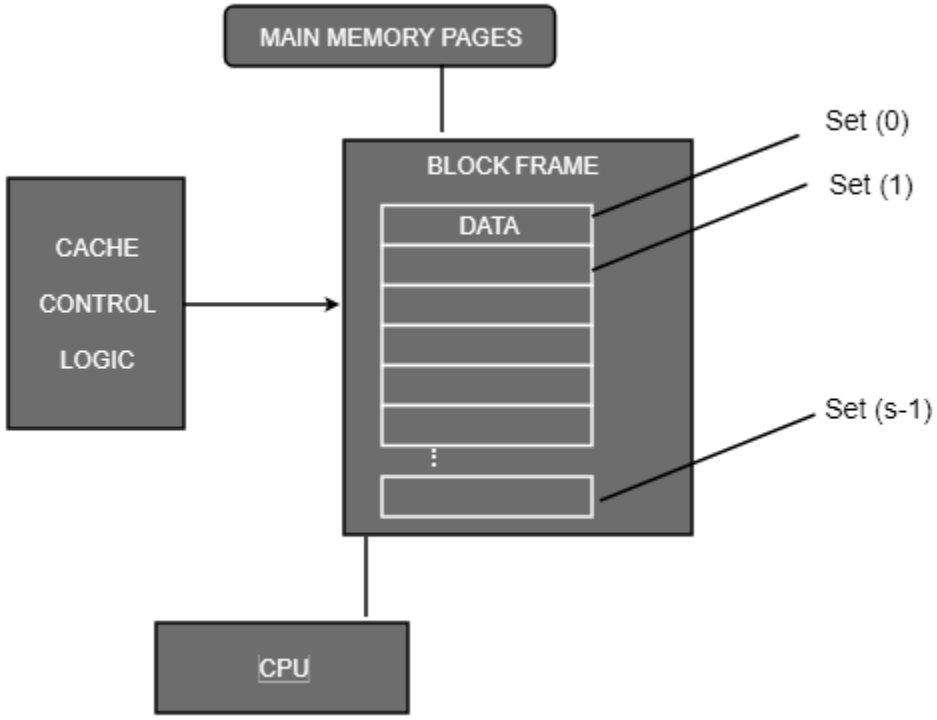
We can improve Cache performance using higher cache block size, higher associatively, reduce miss rate, reduce miss penalty, and reduce Reduce the time to hit in the cache.

Cache Mapping

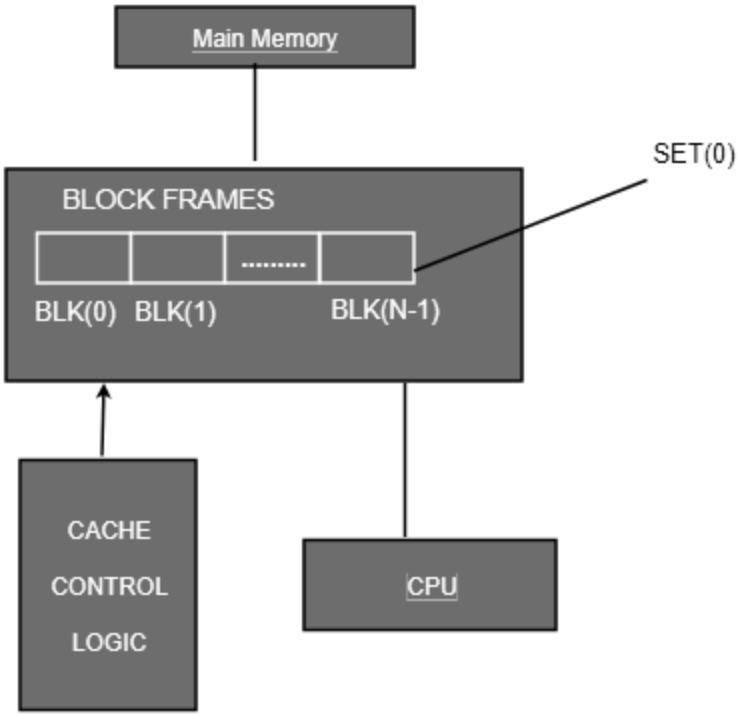
The two different types of mapping used for the purpose of cache memory are as follow,

- Direct mapping
- Associative mapping

Direct mapping: In direct mapping assigned each memory block to a specific line in the cache. If a line is previously taken up by a memory block when a new block needs to be loaded, the old block is trashed. An address space is split into two parts index field and tag field.




Associative mapping: In this type of mapping the associative memory is used to store content and addresses both of the memory word. Any block can go into any line of the cache. This means that the word id bits are used to identify which word in the block is needed, but the tag becomes all of the remaining bits. This enables the placement of the any word at any place in the cache memory. It is considered to be the fastest and the most flexible mapping form.




### 4.4 Virtual Memory

Virtual memory is a memory management capability of an OS that uses hardware and software to allow a computer to compensate for physical memory shortages by temporarily transferring data from random access memory (RAM) to disk storage. Virtual address space is increased using active memory in RAM and inactive memory in hard disk drives (HDDs) to form contiguous addresses that hold both the application and its data.

Computers have a finite amount of RAM so memory can run out, especially when multiple programs run at the same time. A system using virtual memory can load larger programs or multiple programs running at the same time, allowing each one to operate as if it has infinite memory and without having to purchase more RAM.

As part of the process of copying virtual memory into physical memory, the OS divides memory into page files or swap files that contain a fixed number of addresses. Each page is stored on a disk and when the page is needed, the OS copies it from the disk to main memory and translates the virtual addresses into real addresses.

Pros and cons of using virtual memory

Among the primary benefits of virtual memory is its ability to handle twice as many addresses as main memory. It uses software to consume more memory by using the hard disk as temporary storage while memory management units translate virtual memory addresses to physical addresses via the central processing unit. Programs use virtual addresses to store instructions and data; when a program is executed, the virtual addresses are converted into actual memory addresses. Other advantages of virtual memory are that it frees applications from managing shared memory and saves users from adding more memory modules when RAM space runs out.

However, the use of virtual memory has its tradeoffs, particularly with speed. It's generally better to have as much physical memory as possible so programs work directly from RAM or physical memory. The use of virtual memory slows a computer because data must be mapped between virtual and physical memory, which requires extra hardware support for address translations.

In a virtualized computing environment, administrators can use virtual memory management techniques to allocate additional memory to a virtual machine (VM) that has run out of resources. Such virtualization management tactics can improve VM performance and management flexibility.

### 4.5 Memory Management Hardware

A computer's memory management unit (MMU) is the physical hardware that handles its virtual memory and caching operations. The MMU is usually located within the computer's central processing unit (CPU), but sometimes operates in a separate integrated chip (IC). All data request inputs are sent to the MMU, which in turn determines whether the data needs to be retrieved from RAM or ROM storage. A memory management unit is also known as a paged memory management unit.


RGPVNOTES.IN

We hope you find these notes useful.

You can get previous year question papers at https://qp.rgpvnotes.in .

If you have any queries or you want to submit your study notes please write us at rgpvnotes.in@gmail.com

LIKE & FOLLOW US ON FACEBOOK
facebook.com/rgpvnotes.in