Memory Organisation

Memory organization is essential for efficient data processing and storage. The memory hierarchy ensures quick access to data by the CPU, while larger, slower storage devices hold data for the long term. Effective memory management ensures the system operates efficiently, providing programs with the memory they need and preventing unnecessary delays in processing.

Types of Memory in a Computer System

Auxiliary Memory (Non-Volatile)

Devices that provide secondary or backup storage are called auxiliary memory. For example, Magnetic disks and tapes are commonly used auxiliary devices. It is not directly accessible to the CPU, is accessed using the Input/Output channels.

- **Hard Disk Drive (HDD)**: A permanent storage device that holds large amounts of data even when the computer is turned off. It is slower than RAM but offers much more capacity.
- **Solid-State Drive (SSD)**: A faster alternative to HDDs with no moving parts. SSDs provide faster read/write speeds compared to HDDs.
- **Optical Discs and USB Flash Drives**: Optical discs and USB flash drives are other forms of secondary memory used for storage, though they are less common in modern high-speed systems.

Main Memory (Volatile)

The memory unit that communicates directly within the CPU, Cache memory is called main memory. It is fast memory used to store data during computer operations. Main memory is made up of **RAM** and **ROM**, majority part consists of RAM.

RAM Random Access Memory

- **DRAM**: Dynamic RAM, is made of capacitors and transistors. It is slower and cheaper than SRAM.
- **SRAM**: Static RAM, retains data, until powered off.

ROM Read Only Memory

Read Only Memory, is non-volatile and is more like a permanent storage for information. It also stores the **bootstrap loader** program, to load and start the operating system when

computer is turned on. **PROM** (Programmable ROM), **EPROM** (Erasable PROM) and **EEPROM** (Electrically Erasable PROM) are some commonly used ROMs.

Cache Memory

The **cache memory** is used to store program data that is currently being executed in the CPU. Whenever the CPU needs to access memory, it first checks the cache memory. If the data is not found in cache memory then the CPU moves onto the main memory.

Registers

These are small, ultra-fast memory locations within the CPU used to hold data that is being processed. Registers are crucial for executing instructions efficiently.

Tertiary and Offline Memory

- **Tertiary memory** refers to storage devices used for backups and archives, like magnetic tapes.
- **Offline memory** is storage that is not directly accessible by the computer (e.g., external hard drives, optical discs) but data can be retrieved when connected.

Other Types of Memory Based on Storage Time

Volatile Memory: This loses its data, when power is switched off.

Non-Volatile Memory: This is a permanent storage and does not lose any data when power is switched off.

Memory Organization

- **Program Load**: When a program is executed, it is loaded from secondary storage (HDD/SSD) into main memory (RAM). It may also be loaded partially into cache memory to speed up execution.
- **Accessing Data**: The CPU accesses data through registers and cache for quick computations. If the data is not in the cache, it will fetch it from RAM. If it's not in RAM either, it will fetch it from secondary storage.
- **Swapping and Virtual Memory**: If the system runs out of physical RAM, parts of the program (pages) may be swapped out to secondary storage. This process, known as paging, is managed by the operating system's memory manager.

Memory Hierarchy Design

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 (same data or nearby data is likely to be accessed again and again).

Memory Hierarchy helps in optimizing the memory available in the computer. There are multiple levels present in the memory, each one having a different size, different cost, etc. Some types of memory like cache, and main memory are faster as compared to other types of memory but they are having a little less size and are also costly whereas some memory has a little higher storage value, but they are a little slower. Accessing of data is not similar in all types of memory, some have faster access whereas some have slower access.

Types of Memory Hierarchy

This Memory Hierarchy Design is divided into 2 main types:

- **External Memory or Secondary Memory:** Comprising of Magnetic Disk, Optical Disk, and Magnetic Tape i.e. peripheral storage devices which are accessible by the processor via an I/O Module.
- **Internal Memory or Primary Memory:** Comprising of Main Memory, Cache Memory & CPU registers. This is directly accessible by the processor.

Memory Hierarchy Design

1. Registers

Registers are small, high-speed memory units located in the CPU. They are used to store the most frequently used data and instructions. Registers have the fastest access time and the smallest storage capacity, typically ranging from 16 to 64 bits.

2. Cache Memory

Cache memory is a small, fast memory unit located close to the CPU. It stores frequently used data and instructions that have been recently accessed from the main memory. Cache memory is designed to minimize the time it takes to access data by providing the CPU with quick access to frequently used data.

3. Main Memory

Main memory, also known as RAM (Random Access Memory), is the primary memory of a computer system. It has a larger storage capacity than cache memory, but it is slower. Main memory is used to store data and instructions that are currently in use by the CPU.

Types of Main Memory

- **Static RAM:** Static RAM stores the binary information in flip flops and information remains valid until power is supplied. Static RAM has a faster access time and is used in implementing cache memory.
- **Dynamic RAM:** It stores the binary information as a charge on the capacitor. It requires refreshing circuitry to maintain the charge on the capacitors after a few milliseconds. It contains more memory cells per unit area as compared to SRAM.

read more about - Different Types of RAM (Random Access Memory)

4. Secondary Storage

Secondary storage, such as hard disk drives (HDD) and solid-state drives (SSD) , is a non-volatile memory unit that has a larger storage capacity than main memory. It is used to store data and instructions that are not currently in use by the CPU. Secondary storage has the slowest access time and is typically the least expensive type of memory in the memory hierarchy.

5. Magnetic Disk

Magnetic Disks are simply circular plates that are fabricated with either a metal or a plastic or a magnetized material. The Magnetic disks work at a high speed inside the computer and these are frequently used.

6. Magnetic Tape

Magnetic Tape is simply a magnetic recording device that is covered with a plastic film. Magnetic Tape is generally used for the backup of data. In the case of a magnetic tape, the access time for a computer is a little slower and therefore, it requires some amount of time for accessing the strip.

Characteristics of Memory Hierarchy

- **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.
- **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.
- **Performance:** The Memory Hierarch design ensures that frequently accessed data is stored in faster memory to improve system performance.

- **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.

Advantages of Memory Hierarchy

- **Performance:** Frequently used data is stored in faster memory (like cache), reducing access time and improving overall system performance.
- **Cost Efficiency:** By combining small, fast memory (like registers and cache) with larger, slower memory (like RAM and HDD), the system achieves a balance between cost and performance. It saves the consumer's price and time.
- **Optimized Resource Utilization:** Combines the benefits of small, fast memory and large, cost-effective storage to maximize system performance.
- **Efficient Data Management:** Frequently accessed data is kept closer to the CPU, while less frequently used data is stored in larger, slower memory, ensuring efficient data handling.

Virtual Memory

Virtual memory is a memory management technique used by operating systems to give the appearance of a large, continuous block of memory to applications, even if the physical memory (RAM) is limited and not necessarily allocated in contiguous manner. The main idea is to divide the process in pages, use disk space to move out the pages if space in main memory is required and bring back the pages when needed.

Objectives of Virtual Memory

- A program doesn’t need to be fully loaded in memory to run. Only the needed parts are loaded.
- Programs can be bigger than the physical memory available in the system.
- Virtual memory creates the illusion of a large memory, even if the actual memory (RAM) is small.
- It uses both RAM and disk storage to manage memory, loading only parts of programs into RAM as needed.
- This allows the system to run more programs at once and manage memory more efficiently.

How Virtual Memory Works

- Virtual memory uses both hardware and software to manage memory.
- When a program runs, it uses virtual addresses (not real memory locations).

- The computer system converts these virtual addresses into physical addresses (actual locations in RAM) while the program runs.

Types of Virtual Memory

In a computer, virtual memory is managed by the Memory Management Unit (MMU), which is often built into the CPU. The CPU generates virtual addresses that the MMU translates into physical addresses. There are two main types of virtual memory:

1. Paging
2. Segmentation

1. Paging

Paging divides memory into small fixed-size blocks called pages. When the computer runs out of RAM, pages that aren't currently in use are moved to the hard drive, into an area called a swap file. Here,

- The swap file acts as an extension of RAM.
- When a page is needed again, it is swapped back into RAM, a process known as page swapping.
- This ensures that the operating system (OS) and applications have enough memory to run.

Page Fault Service Time: The time taken to service the page fault is called page fault service time. The page fault service time includes the time taken to perform all the above six steps.

- Let Main memory access time is: m*m*
- Page fault service time is: s*s*
- Page fault rate is : p*p*
- Then, Effective memory access time = (p∗s)+(1−p)∗m(*p*∗*s*)+(1−*p*)∗*m*

Page and Frame: Page is a fixed size block of data in virtual memory and a frame is a fixed size block of physical memory in RAM where these pages are loaded.

- Think of a page as a piece of a puzzle (virtual memory) While, a frame as the spot where it fits on the board (physical memory).
- When a program runs its pages are mapped to available frames so the program can run even if the program size is larger than physical memory.

2. Segmentation

Segmentation divides virtual memory into segments of different sizes. Segments that aren't currently needed can be moved to the hard drive. Here,

- The system uses a segment table to keep track of each segment's status, including whether it's in memory, if it's been modified and its physical address.
- Segments are mapped into a process's address space only when needed.

Memory Management Hardware

The history of memory management hardware dates back to the early days of computing, when physical memory was limited and manual management was required. Over the years, advancements in technology have led to the development of hardware-based memory management systems that automate the process, improving performance and reliability. Today, memory management hardware solutions are integrated into modern computer architectures, enabling seamless multitasking, efficient memory allocation, and effective utilization of available memory resources.

Memory management hardware in computer architecture plays a crucial role in optimizing system performance and resource allocation. It includes features such as memory caches, memory controllers, and memory management units (MMUs). These components work together to ensure efficient access to data, reduce latency, and enhance overall system responsiveness. Additionally, memory management hardware helps in handling memory allocation and deallocation, virtual memory management, and protection mechanisms. This advanced hardware technology is essential for modern computer systems to deliver optimal performance and meet the demands of complex applications and multitasking environments.

Components of Memory Management Hardware

The memory management hardware consists of several key components that work together to ensure efficient memory usage and allocation. These components include:

- Memory Management Unit (MMU): The MMU is a critical component of memory management hardware. It translates virtual addresses to physical addresses, enabling the system to access the correct memory location.
- Translation Lookaside Buffer (TLB): The TLB is a cache that stores recently used virtual-to-physical address translations, speeding up the address translation process.
- Memory Segmentation Unit: This unit divides the memory into fixed-size segments to organize and manage memory resources efficiently.

- Memory Protection Unit (MPU): The MPU ensures the security and protection of memory by enforcing access permissions and preventing unauthorized access.

Memory management unit

MMU stands for Memory management unit also known as PMMU (paged memory management unit), Every computer system has a memory management unit , it is a hardware component whose main purpose is to convert virtual addresses created by the CPU into physical addresses in the computer's memory. In simple words, it is responsible for memory management In a device as it acts as a bridge between the CPU and the RAM, which ensures that programs can run smoothly and access the required data without clashes or unauthorized access.

Translation Lookaside Buffer (TLB)

The Translation Lookaside Buffer (TLB) is a cache in the memory management hardware that stores recently used virtual-to-physical address translations. It acts as a high-speed memory for address translation, improving the overall performance of the system.

When the CPU generates a virtual address, the TLB checks if the translation for that address is available in its cache. If the translation is found, the TLB provides the corresponding physical address, eliminating the need for a time-consuming lookup in the page tables or translation tables.

The TLB operates on the principle of locality, which states that recently accessed memory locations are likely to be accessed again in the near future. By storing frequently used translations, the TLB reduces the overhead of address translation, improving system performance.

Functions of Memory Management Hardware

The memory management hardware performs several critical functions to ensure efficient memory usage and allocation. These functions include:

- Address Translation: The memory management hardware translates virtual addresses into physical addresses, allowing the CPU to access the correct memory location.
- Memory Allocation: It allocates and deallocates memory resources to processes, ensuring that each process has sufficient memory to execute efficiently.
- Memory Protection: The memory management hardware enforces access permissions and prevents unauthorized access to memory areas.
- Virtual Memory Management: It manages the mapping of virtual addresses to physical addresses, enabling the efficient use of limited physical memory by utilizing disk-based virtual memory.

Address Translation

Address translation is one of the primary functions of memory management hardware. It involves converting virtual addresses generated by the CPU into physical addresses, allowing the system to access the correct memory location.

During address translation, the memory management hardware uses page tables or translation tables to map virtual addresses to physical addresses. This process ensures that each process has its isolated memory space, protecting it from interference by other processes.

Address translation is essential for enabling the efficient use of physical memory and facilitating the execution of multiple processes concurrently. By translating virtual addresses to physical addresses, the memory management hardware provides each process with a unique memory address space.

Memory Allocation

Memory allocation is another crucial function of memory management hardware. It involves assigning memory resources to processes and deallocating them when no longer needed.

The memory management hardware tracks the available memory blocks, allocating them to processes based on their memory requirements. It ensures that each process has a sufficient amount of memory to execute efficiently, preventing resource contention.

Efficient memory allocation allows for the simultaneous execution of multiple processes, maximizing system productivity. By managing memory resources effectively, the memory management hardware minimizes wastage and fragmentation, optimizing overall system performance.

Techniques Used in Memory Management Hardware

The memory management hardware employs various techniques to enhance memory utilization and performance. Some of the commonly used techniques include:

- Paging: Paging involves dividing memory into fixed-size blocks called pages and mapping them in a virtual address space. It allows for efficient memory allocation and utilization by allocating pages based on demand.
- Segmentation: Segmentation involves dividing memory into logical segments based on program structure and memory requirements. It provides a flexible memory allocation scheme but can lead to external fragmentation.
- Virtual Memory: Virtual memory is a technique that allows the execution of programs larger than the available physical memory. It uses disk-based storage to supplement the physical memory, transferring data between memory and disk as needed.
- Memory Protection: The memory management hardware implements various memory protection mechanisms, such as access permissions and privilege levels, to ensure the security and integrity of memory.

Paging

Paging is a commonly used memory management technique that involves dividing memory into fixed-size blocks called pages. These pages are then mapped into a virtual address space to facilitate efficient memory allocation and utilization.

Paging allows for the efficient allocation of memory resources by allocating pages based on demand. When a process requires additional memory, the memory management hardware allocates new pages to the process, ensuring that each process has sufficient memory to execute efficiently.

Paging also provides protection and isolation between processes by assigning each process its unique page tables. This prevents unauthorized access and interference between processes, enhancing system security and stability.

Virtual Memory

Virtual memory is a memory management technique that allows the execution of programs larger than the available physical memory. It provides an illusion of larger memory space by utilizing disk-based storage as an extension of the physical memory.

In virtual memory, the memory management hardware transparently transfers data between the physical memory and disk storage as needed. It uses techniques like demand paging or demand segmentation to efficiently manage memory resources.

Virtual memory significantly improves system performance by allowing the execution of larger programs without requiring an equivalent amount of physical memory. By utilizing disk storage as a supplement to physical memory, virtual memory enables the efficient execution of memory-intensive applications.