Unit -3

Memory system

Basic concept :-

→ programs & the data that processor operate are held in the main memory of the computer during execution. And the data with high storage requirement is stored in the secondary memories such as floppy disk & hard disk etc.

→ The Maximum size of the Memory that can be used in any computer is determined by the addressing scheme.

→ If it is a 16-bit computer, that generates 16-bit addresses, is capable of addressing upto 2^{16} = 64K memory Locations.

→ If it is a 32-bit addresses, is capable of addressing up to 2^{32} = 4G (Giga) memory Locations.

→ If it is a 40-bit addresses, is capable of addressing upto 2^{40} = 1T (Tera) memory Locations

→ The Number of Locations Represents the size of the address space.

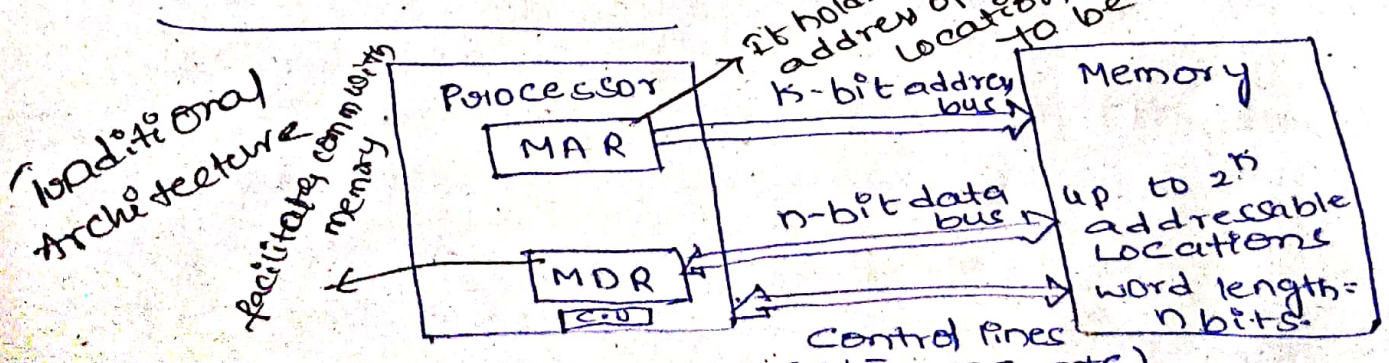


fig: Connection of the Memory to the Processor

→ Data transfer between the Memory to the Processor.

→ The processor takes place through the use of two processor registers, usually called MAR (Memory Address Register) & MDR (Memory Data Reg)

→ If MAR is 'k' bits long and MDR is 'n' bits long then the memory unit may contain upto 2^k addressable Locations.

→ During a memory cycle, n bits of data are transferred between the Memory & the processor.

→ This transfer takes place over the processor bus, which has k address lines & n data lines.

→ This bus also includes the control lines Read/Write (R/$\bar{W}$) & Memory function completed (MFC) for co-ordinating data transfers.

→ The processor reads from the memory by Loading the address of the required memory Location into the MAR register & Setting the R/$\bar{W}$ line to 1.

→ The Memory responds by placing the data from the addressed Location on to the data lines & confirms this action by asserting the MFC signals.

→ The processor writes data into a memory Location by Loading the address of this Location in to MAR & Loading the data in to MDR.

→ It indicates that a write operation is involved by

by setting the R|$\overline{W}$ line to 0.

Characteristics of Memory system :

⇒ **Locations :-**

The computer Memory is placed in three different "Locations".

1. CPU
2. Internal (Main Memory)
3. External (secondary ")

1. **CPU :-** It is in the form of CPU registers and its internal cache Memory (.64K byt

2. **Internal :-** It is the Main Memory of the syste which CPU can access directly.

3. **External :-** It is in the form of secondary (storage devices such as Magnetic disk, tapes The CPU accesses this memory with the help of I/O controllers.

⇒ **Capacity :-** It is expressed using two terms.

1. Word size
2. Number of words.

1. **Word size :-** It is expressed in bytes (8-bit) The common word sizes are 8, 16 & 32-b

2. **Number of words :-** It specifies the no. of word available in the particular Memory devic

Eg: If Memory capacity is 4K×8, then its word size is 8 & the No. of words are 4K = 4096

⇒ Unit of Transfer :- It is the Maximum of bits that can read (or) written in to the Memory at a time.

→ In case of main memory, unit of transfer is equal to word size in most of the times.

→ In case of External Memory, unit of transfer is not limited to word size, it is often larger than a word size and it is referred to as Blocks.

⇒ Access Methods :-

There are two different methods generally used for Memory access.

1. sequential access :- Memory is organized into units of data called "records". If current record is 1, then in order to read record N, it is necessary to read physical records 1 through N-1.

→ A Tape drive is an example, of sequential access memory.

2. Random Access :- In this, each addressable Location in memory has an unique address. It is possible to access any memory Location at random.

⇒ Performance :- The performance of the memory is determined using 3 parameters.

1. Access time
2. Memory cycle time
3. Transfer rate.

Access time :-

In random Access Memory, it is the time taken by Memory to complete read / write operations from the instant that an address is sent to the Memory.

Memory cycle time :-

This term is used only in concern with random Access Memory & it is defined as access time plus addition time required before a second access can commence.

Transfer rate :-

It is defined as the rate at which data can be transferred into (or) out of a memory unit.

Physical type :- Two most common physical types used today are semiconductor Memory & Magnetic surface Memory.

Physical characteristics :- RAM, ROM.

Volatile / Non-volatile :- If Memory can hold data even if power is turned off, it is called as non-volatile Memory; otherwise it is called volatile Memory.

Erasable / Non-erasable :- The memories in which data is once programmed can not be erased are called as Non-erasable Memories.

→ On the other hand, if data in the Memory is erasable, then memory is called as erasable Memory.

Topic 2 : Semiconductor RAM Memories :-

→ Semiconductor memories are available in a wide range of speeds. Their cycle times range from 100ns to less than 10ns.

Internal Organization of Memory chips :-

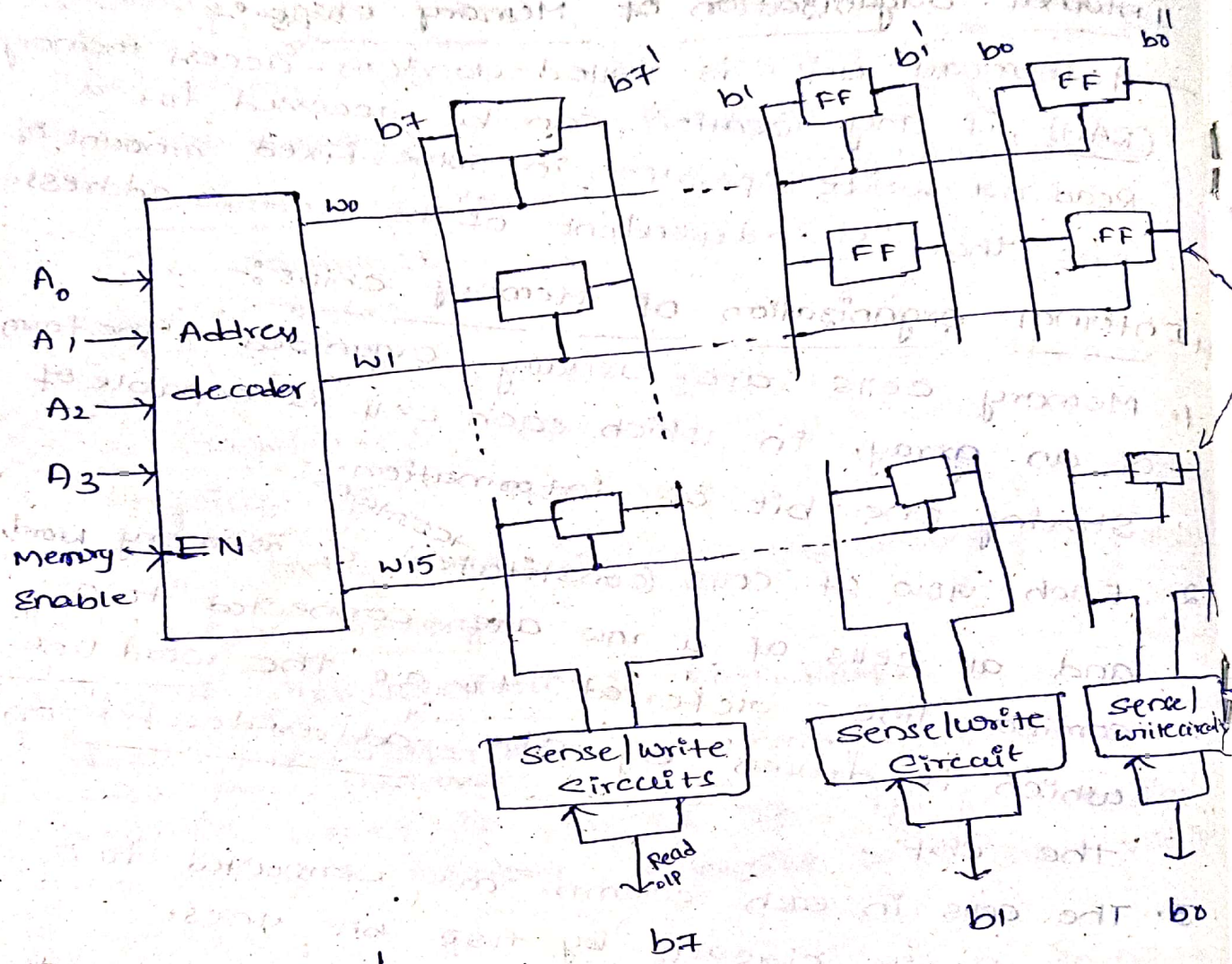
→ A memory unit is called random-access memory (RAM), if any location can be accessed for a Read or write operation in some fixed amount of time that is independent of the location's address.

2.1 Internal organization of Memory chips :-

1. Memory cells are usually organized in the form of an array, in which each cell is capable of storing one bit of information.
2. Each row of cells (constitutes) a memory word, and all cells of a row are connected to a common line, referred to as the "word line" which is driven by the "address decoder" on the chip.
3. The cells in each column are connected to a Sense/write circuit by two bit lines.
4. The Sense/write circuits are connected to the data input/output lines of the chip.
5. During a Read operation, these circuits sense or read the info stored in the cells selected by a word line and transmit this info to the output data lines.

6. During a write operation, the sense/write circuits receive i/p i/f and store it in the cells of the selected word.

Organization using single decoder :-



Data i/p input/output lines

Fig: Organization of bit cells in Memory chip

→ Referred to as a 16×8 organization. (16 word lines & 8-bit lines)

→ The data input and the data output of each sense/write circuit are connected to a single bidirectional data line that can be connected to the data bus of a computer.

→ Two control lines, R/$\bar{W}$ and CS, are provided in addition to the address and data lines.

→ The R/W input specifies the required operation, and the CS (chip select) input selects a given chip in a multichip memory system.

→ Now will see how 1K x 1 memory chip.

→ There are different ways to arrange, so one form is shown here.

→ This circuit can be organized as a 128 x 8 memory, requiring a total of 19 external connections. Alternatively the same no. of cells can be organized into a 1K x 1 format.

→ In this case, a 10-bit address is needed, but there is only one data line, resulting in 15 external connections

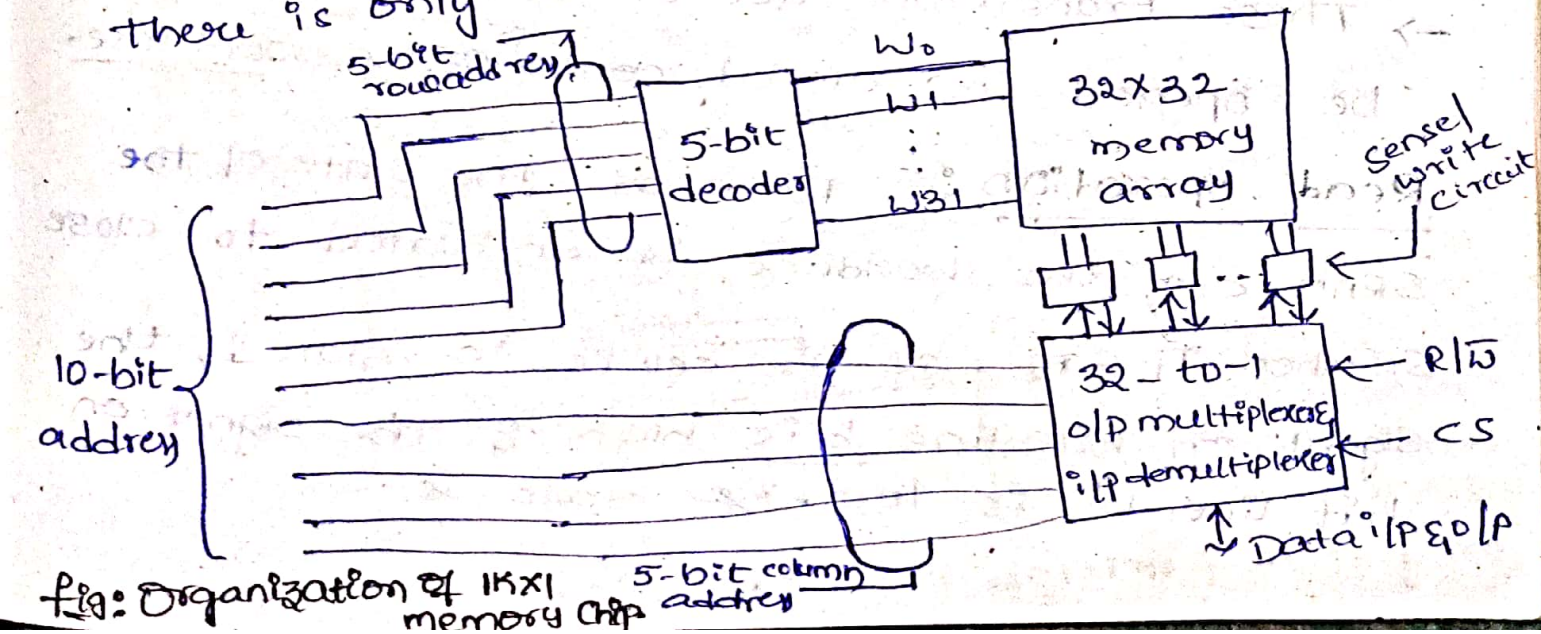


fig: Organization of 1K x 1 memory chip

2.2 Static Memories

Memories that consist of circuits capable of retaining their state as long as power is applied are known as "Static Memories".

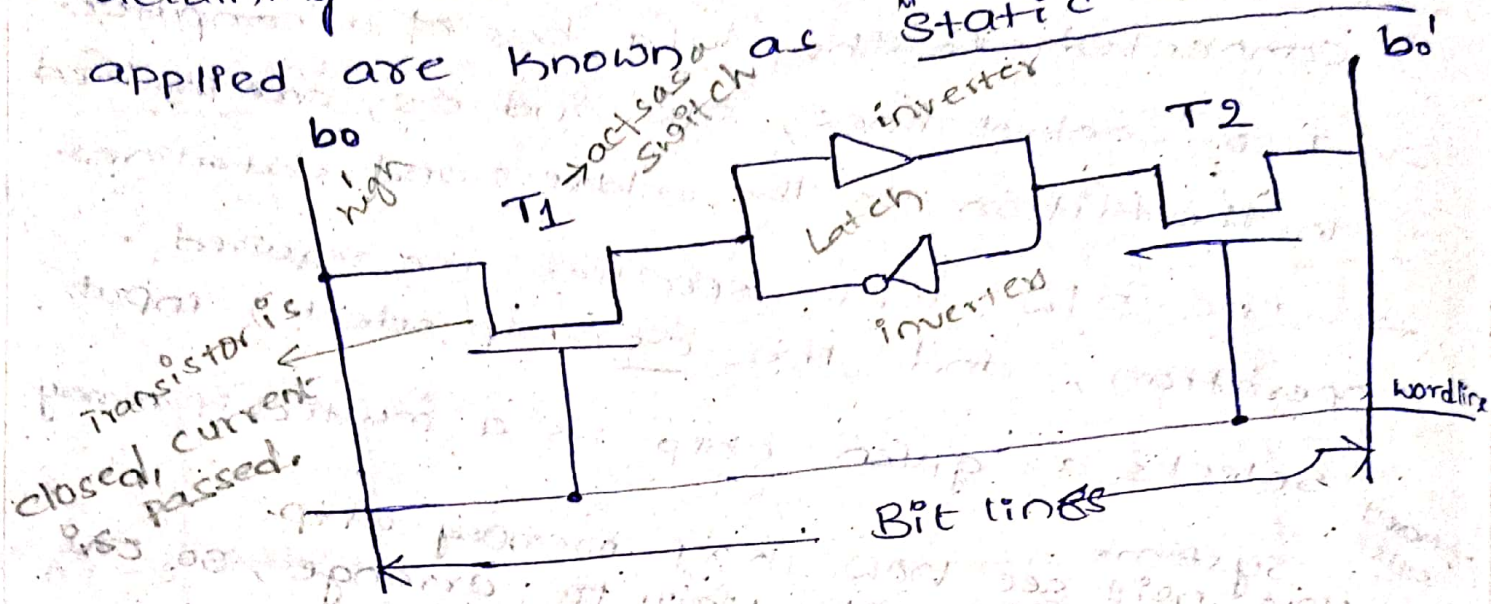


fig: Static RAM cell

→ The above figure is called as a static RAM (SRAM) cell.

→ Two inverters are cross connected to form a Latch. The Latch is connected to two bit lines by transistors T1 & T2.

→ These transistors act as switches that can be opened (or) closed control of the word line.

Read operation :- To read the state of the SRAM cell, the wordline is activated to close switches T1 & T2. If the cell is in state 1, the signal on bit line b is high & the signal on bit line b' is low. If state is.

→ If the cell is in state 0, the signal on bit Line b is Low & the signal on bit Line b' is high.

Write Operation :-

→ The state of the cell is set by placing the appropriate value on bit Line b & its complement on b' & then activating the word Line.

→ This forces the cell into the corresponding state.

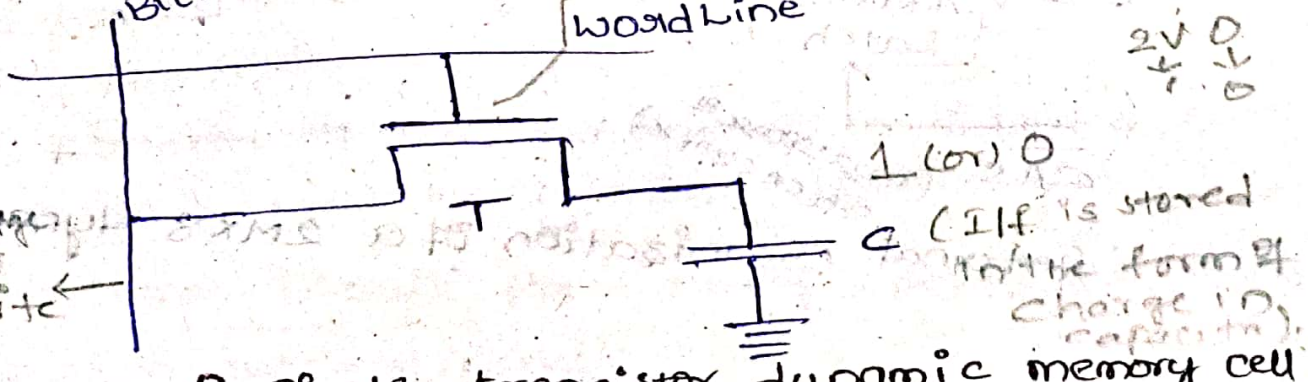
2.3 Asynchronous DRAMS :-

→ Static RAM's are fast, but they come at a high cost because their cells require several transistors.

→ Less Expensive RAM's can be implemented if simpler cells are used.

→ Information is stored in a dynamic Memory cell in the form of a charge on a capacitor.

→ The cell is required to store i/f for a Much longer time, it contents must be periodically refreshed by restoring the capacitor charge to its full value.



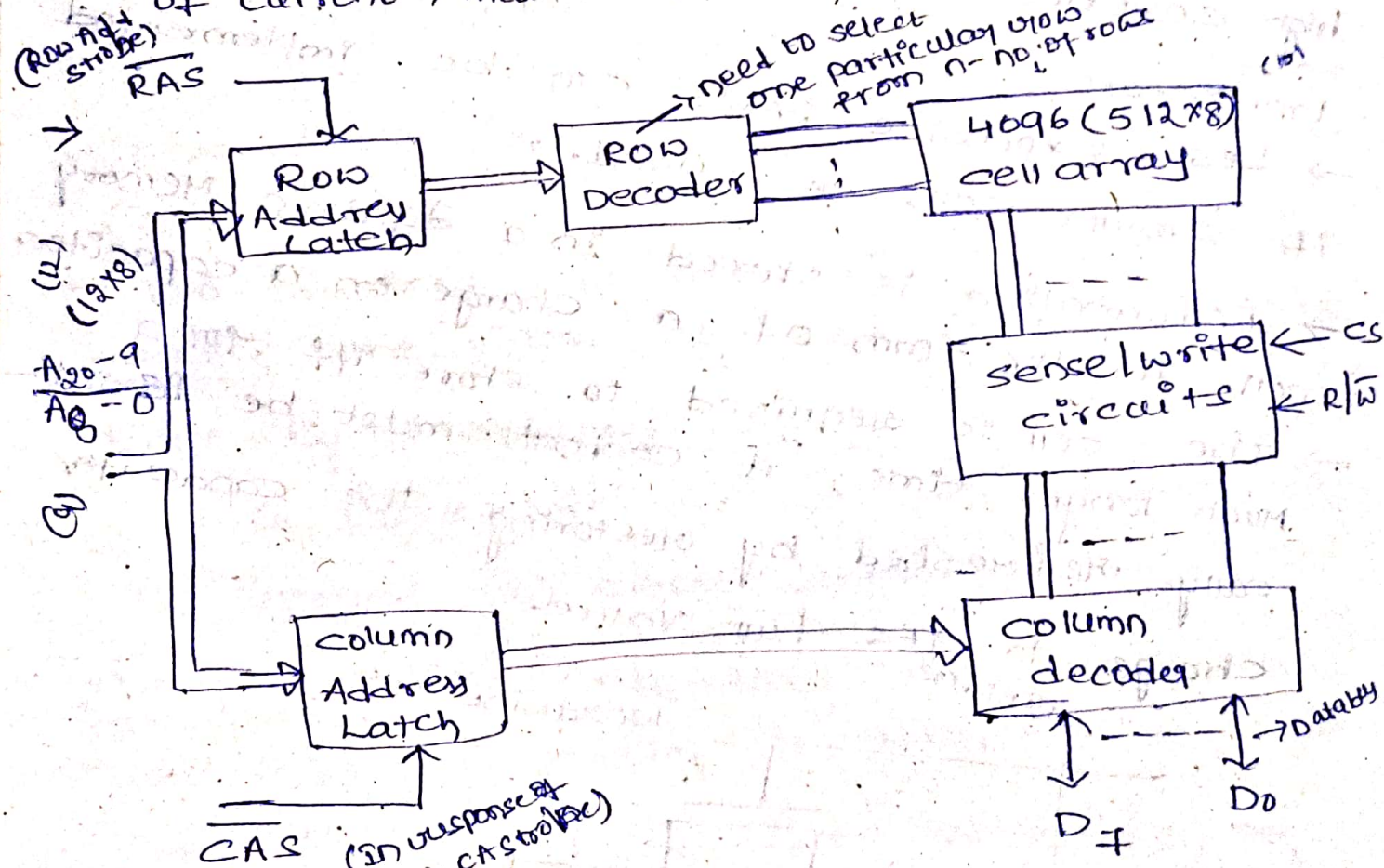
A Single-transistor dynamic memory cell

→ The dynamic Memory cell consists of a capacitor C & a transistor T.

→ Inorder to store 1/f in this cell, transistor 'T' is turned on and an appropriate voltage is applied to the bit line.

→ This causes a known amount of charge to be stored in the capacitor.

→ The transistor is turned off, the capacitor begins to discharge. This is caused by the capacitor's own leakage resistance & by the fact that the transistor continues to conduct a tiny amount of current, measured in picoampere, after it is turned off



Internal organization of a 2Mx8 dynamic Memory chip.

→ A 16-bit Megabit DRAM chip, configured as 2M×8
→ The 4096 cells in each row are divided in to 512 groups of 8, so that a row can store 512 bytes of data.
→ In this 21-bit address is needed to access a byte in this Memory.
→ The high-Order 12 bits and the Low-Order 9 bits of the address constitute the row & column address of a byte.
→ During a read (or) write operation, the row address is applied first. It is loaded into the row address Latch under the control RAS (Row Address strobe) signal. The Read operation is initiated all the cells on the selected row are read & refreshed.
→ Now the address is Loaded into the column address Latch under the control of CAS (column Address strobe)
→ If the R/W control signals indicate a Read operation, the output values of the selected circuits are transferred to the Data lines D_7-D_0.
→ If it is a write operation, the i/p on the D_7-D_0 line is transferred to the selected circuits.
→ A row address causes all cells on the corresponding row to be read and refreshed during both Read & write operation.
→ The contents of a DRAM are maintained, each row of cells must be accessed periodically.
→ A refresh circuit usually performs this function automatically.

→ Because of their high density & low cost DRAMs are widely used in the memory units of the computer.

→ Chips ranges in size from 1M to 256M bits & even larger.

→ To reduce the no. of memory chips needed in a given computer, a DRAM chip is organized to read (or) write a no. of bits in parallel.

2.4. Synchronous DRAMs

→ The operation is directly synchronized with a clock signal. Such memories are known as synchronous DRAMs (SDRAMs)

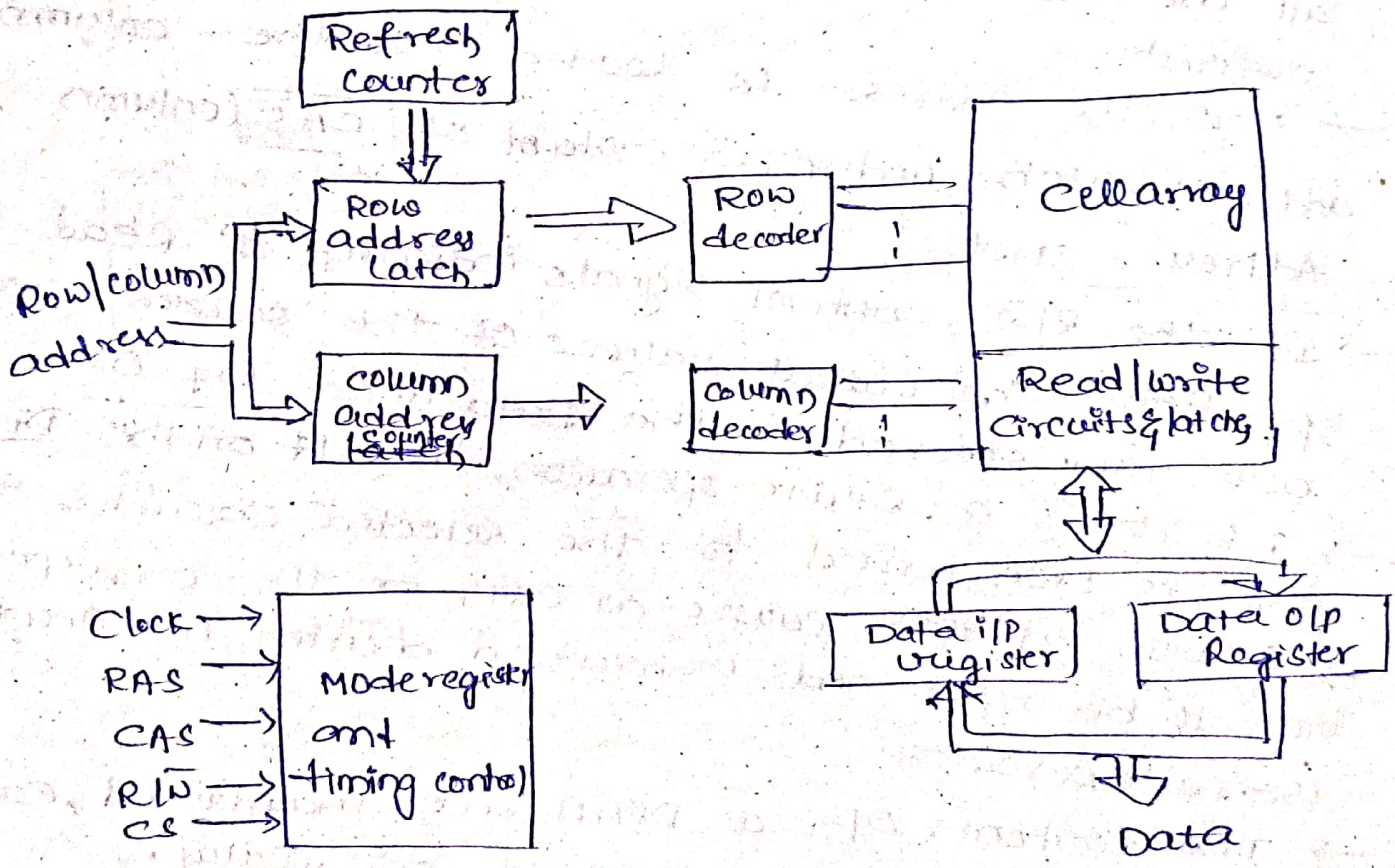


fig: Synchronous DRAM

→ In SDRAM address and data connections are buffered by registers.
→ The O/P of individual sense amplifier is connected to a latch.
→ Mode register is present which can be set to operate the memory chip in different modes.
→ To select successive columns it is not required to provide externally generated pulse on CAS line.
→ A column counter is used internally to generate the required signals.

→ FOR READ operation, the row address is applied first, and in response to the column address, the data present in the latches for the selected columns are transferred to the data output register.
→ Then the data is available on the Databus.

→ FOR WRITE Operation, the row address is applied first, and in response to the column address, the data present in the data bus is made available to the latches through data input register.
→ The data is then written to the particular cell.

Latency and Bandwidth :- A good indication of performance is given by 2 parameters.
1. Latency :- Refers to the amount of time it takes to transfer a word of data to or from the Memory.
2. Bandwidth :- No. of bits or bytes that can be transferred in one second.

2.5. Structures of Larger Memories :-

Static Memory System :-

1. Consider a memory consisting of 2M words of 32 bits each.

2. This we can implement this memory using 512K x 8 static memory chips.

→ Each column consists of four chips which implement one byte position.

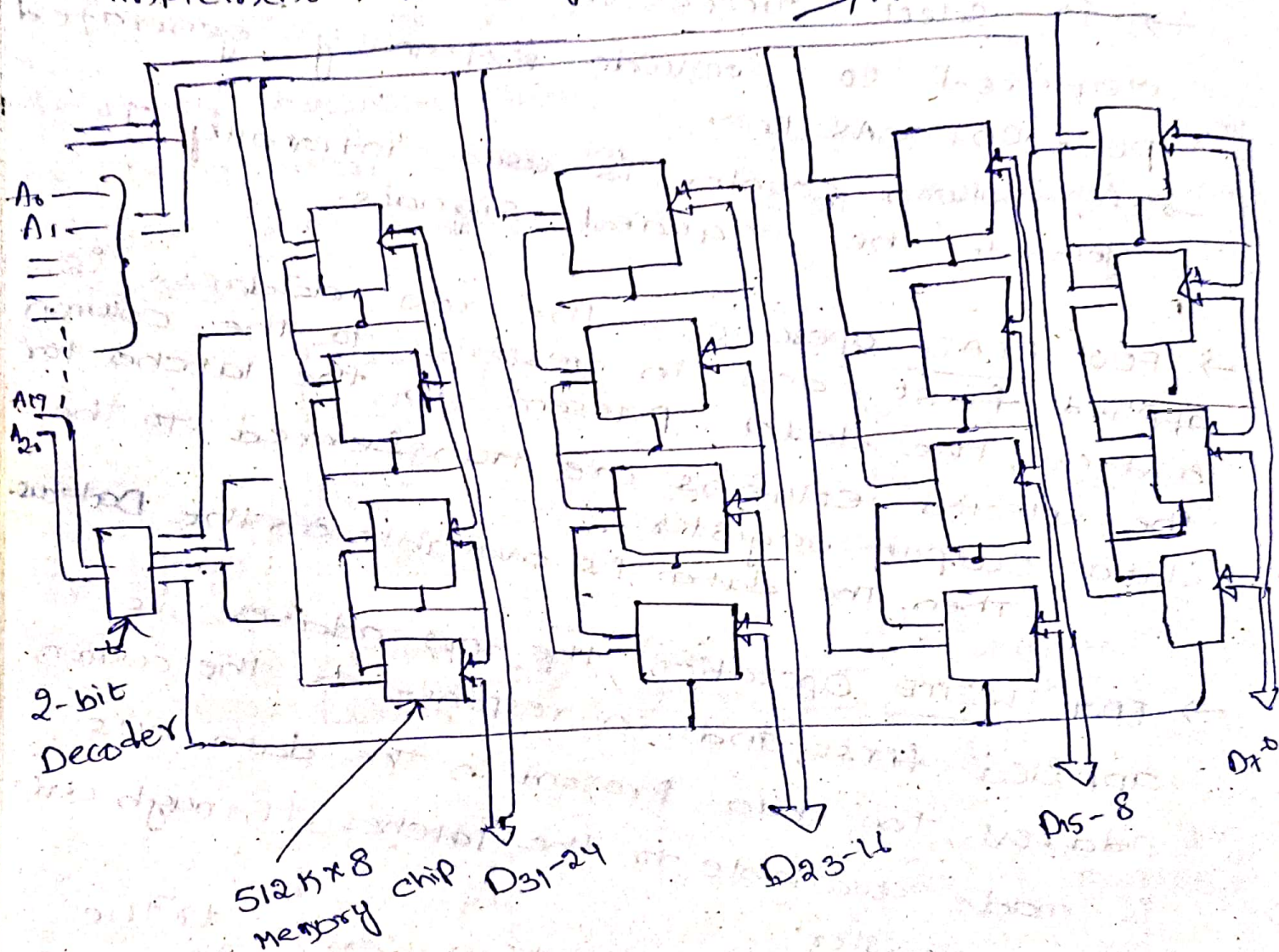
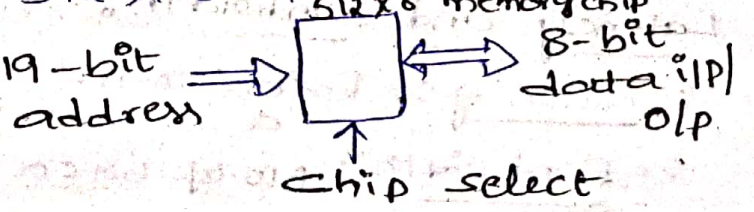


fig: Organization of a 2M x 32 memory module using 512K x 8 static memory chip.



→ when the input is set to 1, it enables the chip to accept data from CPU to Place data on the data lines.
→ Only the selected chip places data on the data output line, while all the other outputs are in high impedance state.

Dynamic Memory system:-

→ The organization of Large dynamic memory system is same as static memory system. However physical implementation is done in the form of memory module.
→ Modern computers use very large memories; even a small PC is likely to have atleast 32M bytes of Memory.
→ A large memory leads to better performance because more of the programs and data used in processing can be held in the memory.
→ A large memory is built by placing DRAM chips directly on the Motherboard.
→ So that it works faster.
→ SIMMs and DIMMs of different sizes are designed to use the same size socket.
↳ Single In-line Memory module ↳ Dual Inline Memory module
Ex: 4Mx32, 16Mx32 and 32Mx32 bit DIMMs all use the same 100 pin socket.
→ Such modules occupy a smaller amount of space on a motherboard, & they allow Easy Expansion by replacement of a larger module by the same socket.

Topic 3: Read Only Memories

→ The SRAM and DRAM chips are volatile.
→ It means that they Lose the stored information, if power is turned off.
→ Many Applications that need memory devices which retain the stored information if power is turned off.
→ Once written can not be changed - It is a permanent memory

1. ROM → Non-Volatile

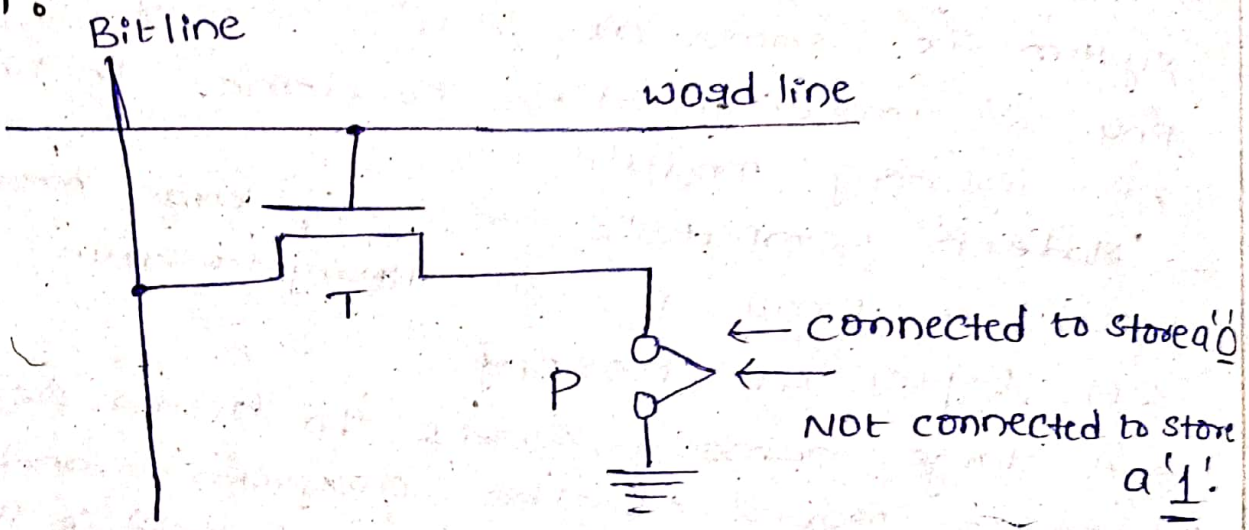


fig: A ROM Cell

→ A Logic value 0 is stored in the cell, if the transistor is connected to ground at point 'P'.
→ Otherwise '1' is stored. The bit line is connected through a resister to the Power supply.
→ To read the state of the cell, the word line is activated. The transistor switch is closed & the voltage on the bit line drops to near zero if there is a connection between the transistor & ground.

2. PROM :- (Programmable ROM)

→ Programmable read only Memory (PROM) is ROM (Read Only Memory) that can be modified once by a user.

→ some ROM designs allow the data to be loaded by the user, thus providing a programmable ROM.

→ storing information specific to a user in a ROM is expensive.

→ providing programming capability to a user may be better.

3. EPROM :- (Erasable Programmable ROM)

→ Another type of ROM chip allows the stored data to be erased and new data to be loaded.

→ Erasable and reprogrammable ROM is usually called an EPROM.

→ It is flexible, during the development phase of digital sys.

→ The important advantage of EPROM chips is that their contents can be erased and reprogramed.

→ Erasure requires Exposing the ROM to the UV Light

4 EEPROM :- (Electrically erasable programmable ROM)

→ The disadvantage of EPROM is that a chip must be physically removed from the circuit for re-programming and that its entire contents are erased by the "ultraviolent Light".

→ An another version of erasable PROMs that can be both Programmed & Erased Electrically.

Differences between PROM, EPROM, EEPROM:-

PROM	EPROM	EEPROM
→ Programmable ROM	→ Erasable PROM	→ Electrically EPROM
→ User can store Programe only once	→ It can be removed by ultra violet rays	→ It can be removed by electric signals.
→ User can write mica Program can be made that are needed mostly	→ It can be re-write after removing previous information	→ It is the simplest way to store information.
→ The process of making program in PROM is called "Burning"	→ It is cheaper than PROM because it is re-usable.	→ It is used to store BIOS in Memory.
Eg: CD-R	Eg: CD(RW)	Eg: Pendrive.

Differences between SRAM and DRAM

SRAM	DRAM
1. Stores data till the power is supplied	1. Stores data only for few milli seconds Even when power is supplied.
2. Uses an array of 6 transistors for Each Memory cell.	2. Uses a single transistor & capacitor for each Memory cell.
3. Does not refresh the Memory cell.	3. Needs to refresh the Memory cell.
4. Data access is faster	4 Data access is slower
5. Consumes more power	5. consumes less power
6. Cost per bit is high	6. Cost per bit is low.

5. Flash Memory :-

→ Special type of EEPROM.

→ Flash memories are read/write memories. In flash memories, it is possible to read the contents of a single cell, but it is only possible to write an entire block of cells.

→ A flash cell is based on a single transistor.

→ Flash devices have greater density than EEPROM.

→ Flash devices have higher capacity & a lower cost per bit, so they require a single power supply and consumes less power in their operation.

2→ In digital cameras, flash memory is used to store picture image data.

3→ In MP3 players, flash memory is used to store the sound data.

4→ The flash memories are available in modules. These modules are implemented in 2 types
1. flash cards 2. flash drives

1. **Flash cards:** In this flash chips are mounted (placed) on a small card. (Flash cards have a standard interface that makes them useable in a variety of products.)

2. **Flash drives:**

5→ Larger flash memory modules have been developed to replace the hard disk drives.

2→ it can be erased with the help of electricity in block.

3 → it can be serially erased & modified.

4. Speed, SIZE And Cost

Magnetic Disk :-

A magnetic disk is a storage device, that can be used to store huge amount of data.

Eg: Hard disks, zipdisks & Floppy disks.

Characteristics	SRAM	DRAM	MagneticDisk
Speed	very fast	slower	Much slower than DRAM.
Size	Large	Small	small
cost	Expensive	Less Expensive	Low price

Memory Hierarchy :-

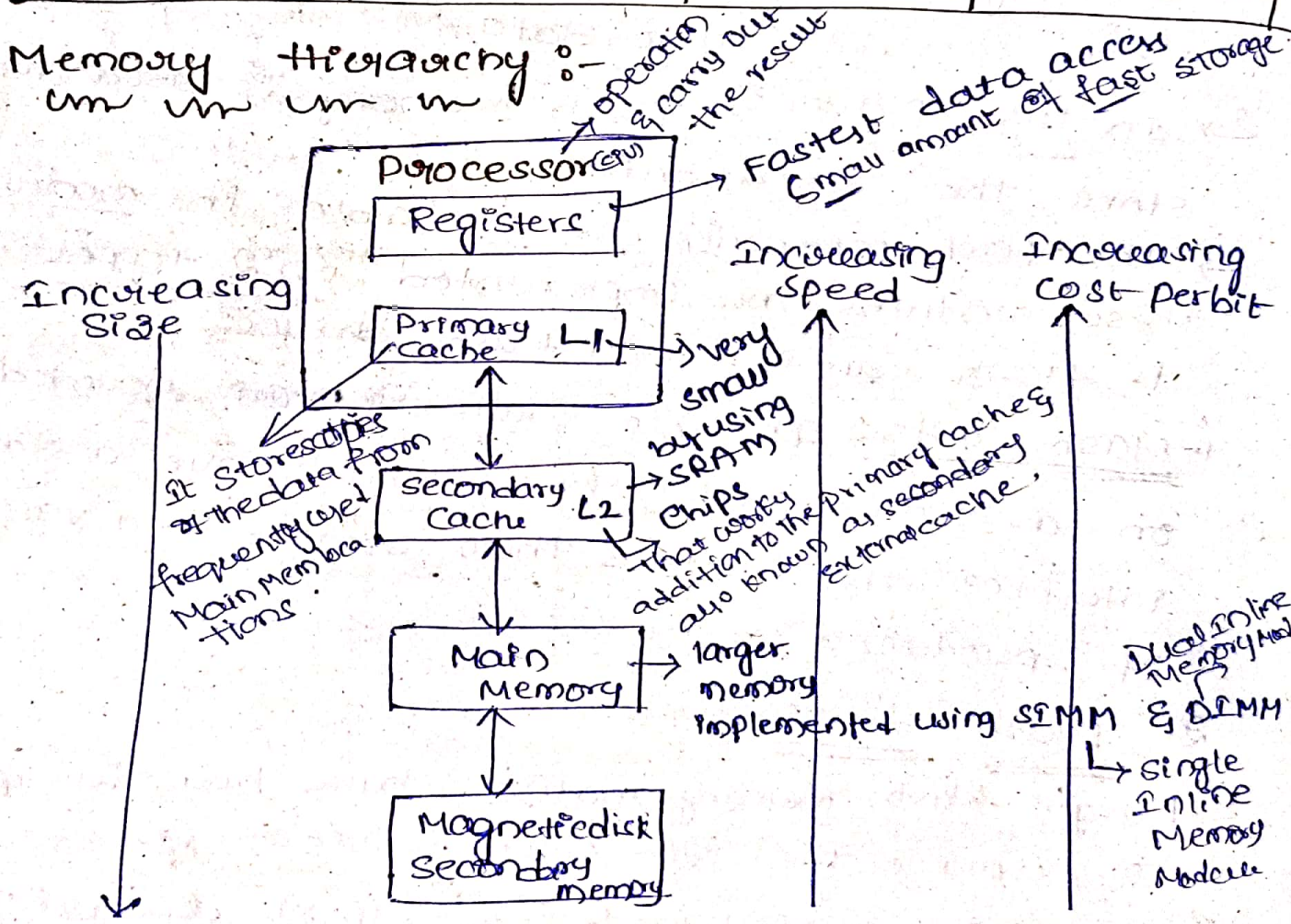


fig: Memory hierarchy.

→ The fastest access is to data held in processor registers. Processor registers are at the top in terms of speed access.

→ At the next level of the hierarchy is relatively small amount of memory that can be implemented directly on the processor chip. This memory, called a "processor cache."

→ There are often two levels of caches.

1. A primary cache is always located on the processor chip, it is very small and also referred to as level 1 (L1) cache.

2. A larger, secondary cache is placed between the primary cache and the main memory. It is referred to as level 2 (L2) cache. It is usually implemented using SRAM chips.

→ The next level in the hierarchy is called the "Main memory". This is larger memory, implemented using dynamic memory components, typically in the form of SIMMs, DIMMs, or RIMMs.

→ It is larger memory but slower than the "cache memory".

→ Disk devices provide a huge amount of inexpensive storage. They are very slow compared to semiconductor devices used to implement the main memory.

→ During program execution, the speed of memory access is utmost importance. This hierarchical memory system used to bring the instructions and data very fast.

5. Cache Memories

→ In a computer system the program which is to be executed is loaded in the main memory (DRAM)

→ Processor then fetches the code and data from the main memory to execute the program.

→ The DRAMs are slower devices. This reduces the speed of execution. To speed up the process, high speed memories such as SRAMs must be used.

→ In the memory system small system section of SRAM is added along with the main memory, referred to as "cache memory".

Hit Rate:

→ The percentage of accesses where the processor finds the code (or) data word it needs in the cache Memory is called the "Hit Rate".

→ The hit rate is normally greater than 90%.

$$\text{Hit rate} = \frac{\text{Number of hits}}{\text{Number of read/write bus cycles}} \times 100$$

Cache Memory System:-

→ A cache memory system includes a small amount of fast Memory (SRAM) and a Large amount of slow Memory (DRAM).

→ The cache memory system is configured to simulate a Large amount of fast memory.

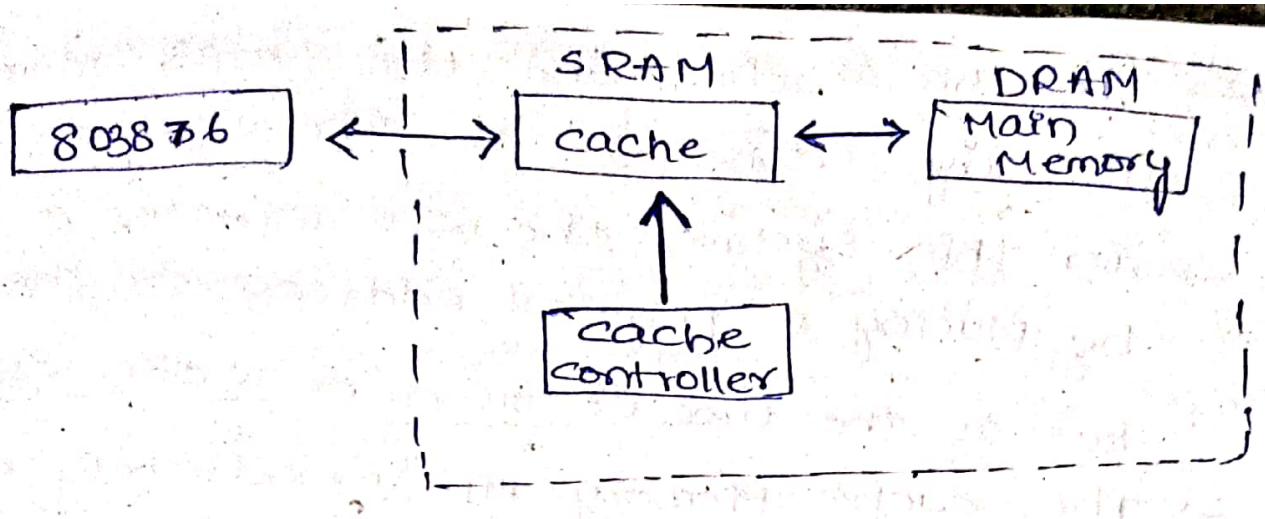


fig: Cache Memory system.

It consists of :-

1. cache :- This blocks consists of static RAM (SRAM).
2. Main Memory :- This blocks consists of Dynamic RAM (DRAM).
3. cache Controller :- This block implements the cache logic.
   ↳ It decides which block of memory should be moved in (or) out of the cache block and into (or) out of main memory, based on the requirements.

→ Most commonly used system organization for cache Memory are :-

1. Look - aside &
2. Look - through

Look - aside system organization :-

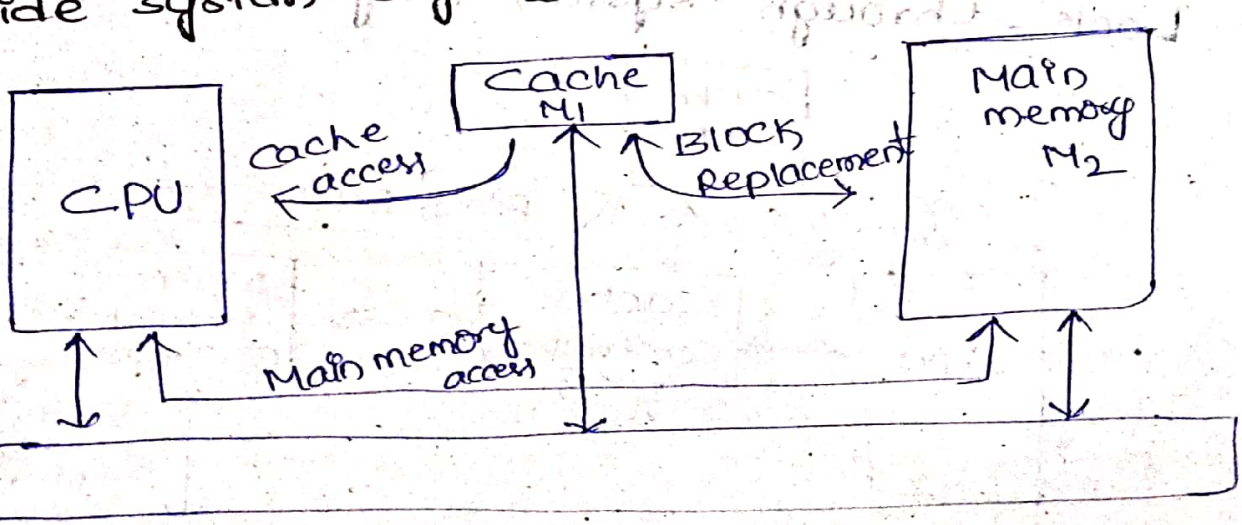


fig: Look aside cache system organization

→ In this cache and the main memory are directly connected to the system bus.

→ In this system, the CPU initiates a memory access by placing a physical address on the memory address bus at the time of read or write cycle.

→ The cache Memory M1 immediately compares physical address to the tag address currently residing in its tag Memory.

→ If a match is found i.e in case of cache hit, the access is completed by a read (or) write executed in the cache.

→ In this case, the main memory is not involved in the process of read or write.

→ If match is not found i.e case of cache miss, the desired access is completed by a read (or) write operation directed to M_2.

→ The system bus is used for this transfer & hence it is unavailable for other uses like I/O operations.

Look-through system Organization :-

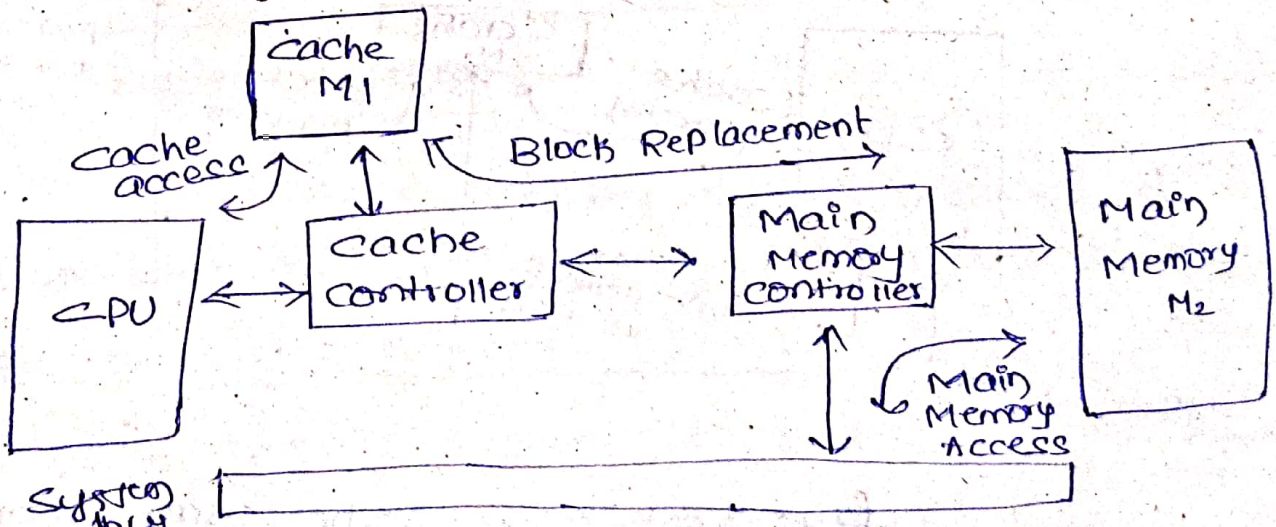


fig: Look-through cache system organization

→ CPU communicates with cache via a separate (Local) bus which is isolated from the main system bus.
→ During cache accesses, the system bus is available for use by other units, such as I/O controllers, to communicate with main Memory.
→ A Look-through cache system use wider Local bus to link M1 & M2.

Advantages:
→ It is faster.

Disadvantages:
→ It is complex.
→ It is costly.
→ It takes longer time for M2 to respond to the CPU when a cache miss occur.

Cache Read Operation:-

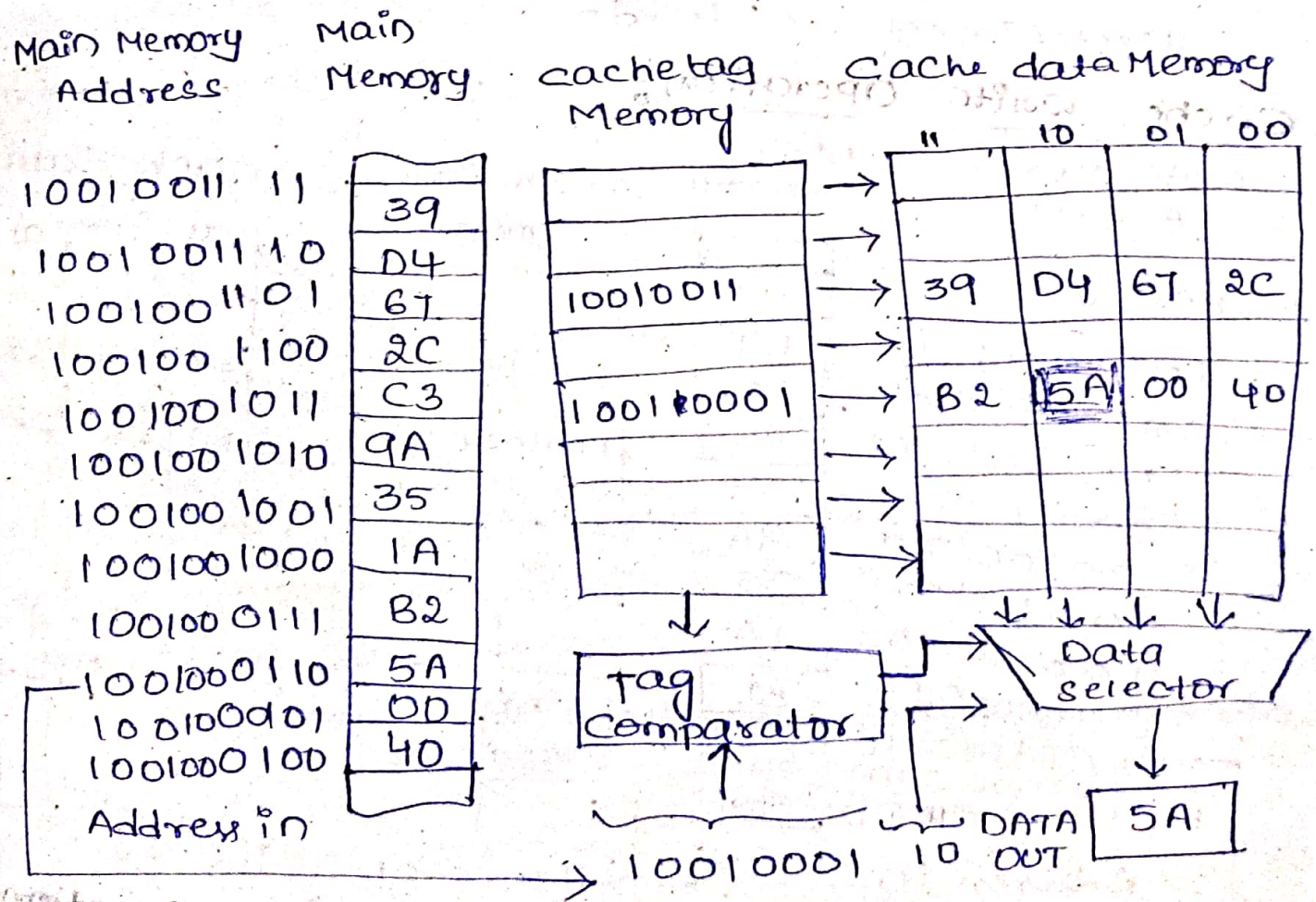


fig: Execution of Cache Read Operation

Cache Read Operation :-

→ Here each cache block is 4 bytes, each byte is of 8 bits.
→ And the each memory address is 10-bit long.
→ These 8-high-order bits form the tag (or) block address and the 2 low-order bits define a displacement address with in the block.
→ when a block is assigned to the cache data Memory its tag is also placed in the cache tag Memory
→ During read operation, the 8 high-order bits of an address are compared with stored tags in the cache tag Memory.
→ The stored tag pin points the corresponding block in cache data Memory & the 2-bit displacement is used.

Cache Write Operation :-

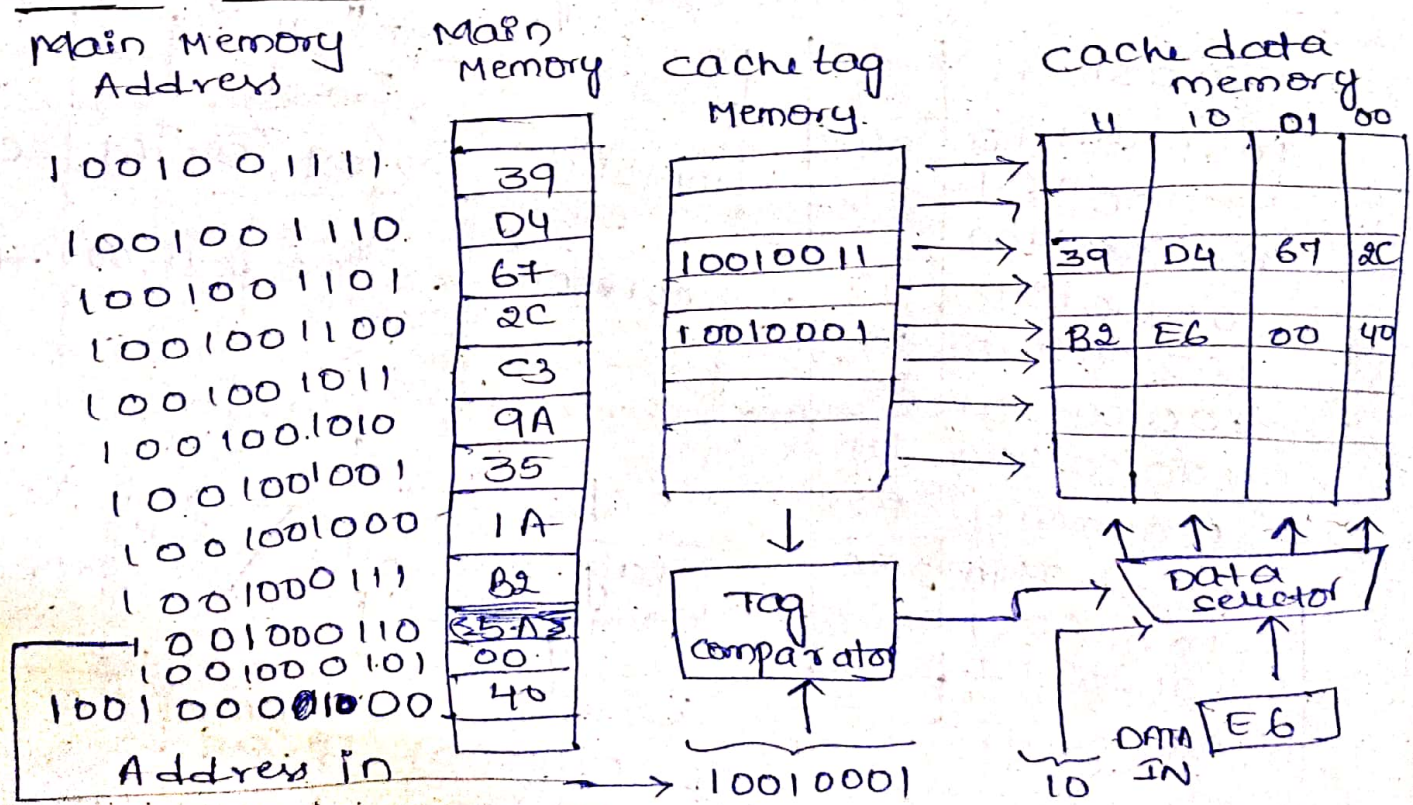


Fig: Execution of cache write operation

cache write operation :-

→ The above figure shows the execution of cache write operation. It uses same addressing technique as in case of read operation.

→ When cache hit occurs, the New data, in this E6, is stored at the Location pointed by address in the cachedata Memory, therefore overwriting the old data 5A.

→ Now, the data in the cache data Memory & Data in the Main Memory for given address is different this causes consistency problem.

Program Locality :- (It is a sub topic under cache memory system)

→ The prediction of Next memory address from the current Memory Address is known as program Locality.

→ It enables cache controller to get a block of memory instead of getting just a single address.

Locality Of Reference :-

→ In some programs, it may contain a simple loop, nested loops (or) a few procedures that repeatedly call each other.

→ Locality of Reference is also called the principle of Locality, is the term applied to situations where the same value (or) Related storage Locations are frequently accessed.

→ In two ways, the Locality of reference executes,

1. Temporal &
2. Spatial

→ In temporal means, that a recently executed instruction is likely to be executed again.

→ The spatial means, instructions stored near by to the recently executed instruction are likely to be executed.

→ The temporal aspect of the Locality of reference suggests that when ever an instruction (or) data is needed, it should be brought into the cache & should remain there until it is needed again.

→ The spatial aspect suggests that instead of bringing just one instruction (or) data from the main memory to the cache, it is wise to bring several instructions and data.

Block Fetch:- It is used to increase the hit rate of cache.

→ A block fetch can retrieve the data located before the requested byte (or) data located after the requested byte.

Elements of cache Design :-

The main elements of cache includes,

1. cache size
2. Mapping functions
3. Replacement Algorithm
4. Write policy
5. Block size
6. Number of caches.

1. **cache size:-** The size of the cache should be small enough so that the overall average cost per bit is close to that of main memory &

→ it should be Large enough, so that the overall average access time is close to that of cache.

2. **Mapping Function :-**

→ The Cache Memory can store a Reasonable Number of blocks at any given time, but this Number is small compared to the total no of blocks in the Main Memory.

→ Mapping is the technique to relate the main Memory blocks & Cache blocks.

Two types of Mapping functions commonly used:

→ Direct Mapping technique

→ Associative Mapping technique
  1. set Associative technique
  2. Fully associative technique.

Direct - Mapping :-

→ It is the simplest mapping technique.

→ In this technique, each block from the Main Memory has only one possible Location in the cache organization.

→ Address is divided into three fields;

→ The Lower order 4 bits select one of the 16 words in a block. This field is known as word field.

→ The second field is known as block field is used to distinguish a block from other blocks.

→ The length is 7 bits since $2^7 = 128$.

→ The third field is a tag field - It is used to store the higher-order 5 bits of memory address of the block.

→ When Memory is accessed, the 7-bit cache block field of each address generated by CPU points a particular block location in the cache.

→ The high-order 5 bits of the address are compared with the tag bits associated with that cache location.

→ If they match, then the desired word is in that block of the cache.

→ If there is no match, then the block containing the required word must be read from the main memory & loaded in to the cache.

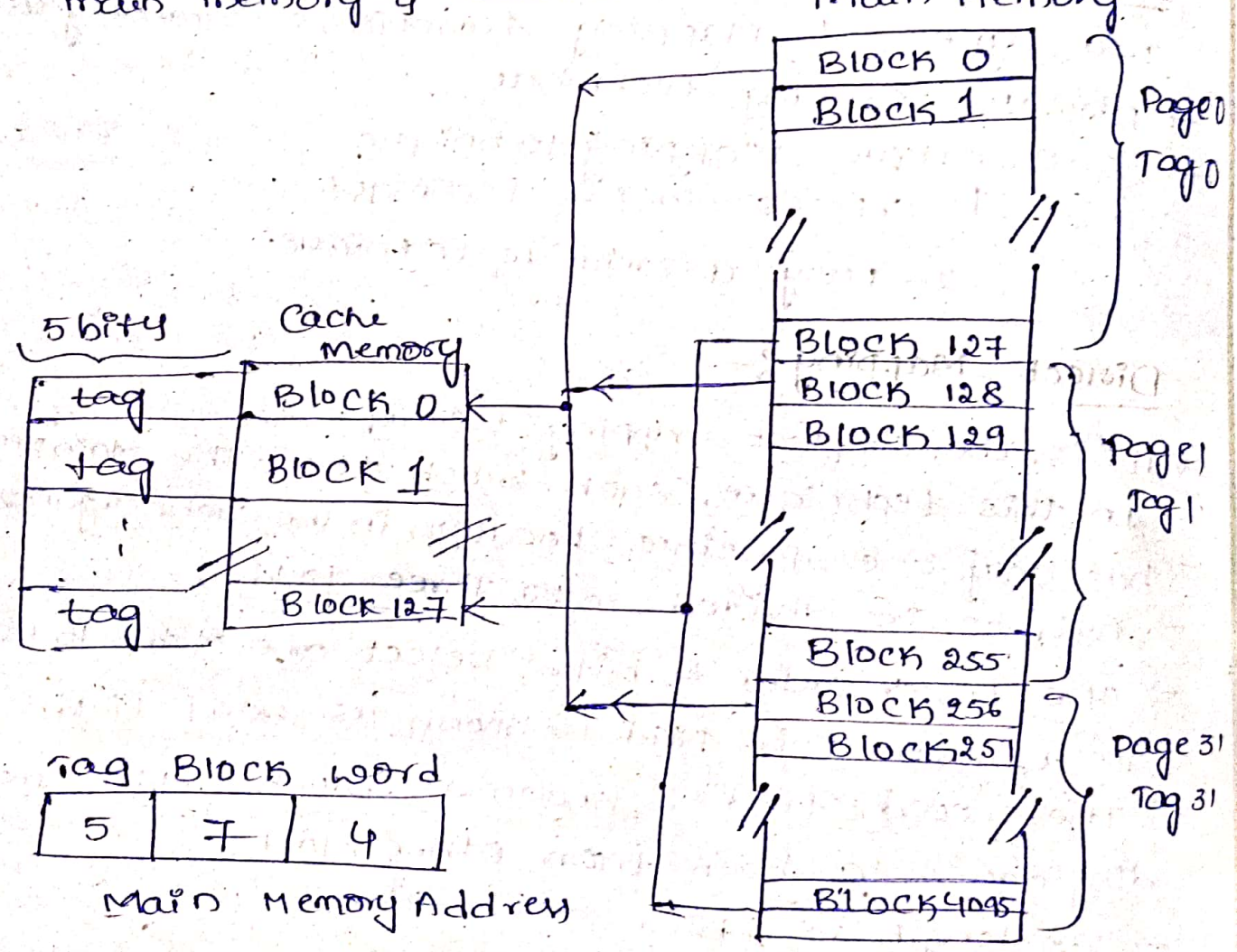


Fig: Direct - Mapped Cache

Associative Mapping :- (Fully Associative Mapping)

→ In this technique, a main memory block can be placed in to any cache block position.

→ Because, there is no fix block, the memory address has only two fields

(1.) word & (2.) Tag

→ The high-order 12 bits of an address received from the CPU are compared with the tag bits of each block of the cache to see the desired block is present.

→ In associative-mapped cache, it is necessary to compare the higher-order bits of address of the main memory with all 128 tag corresponding to each block to determine whether a given block is in the cache. So this is the main disadvantage of associative mapped cache.

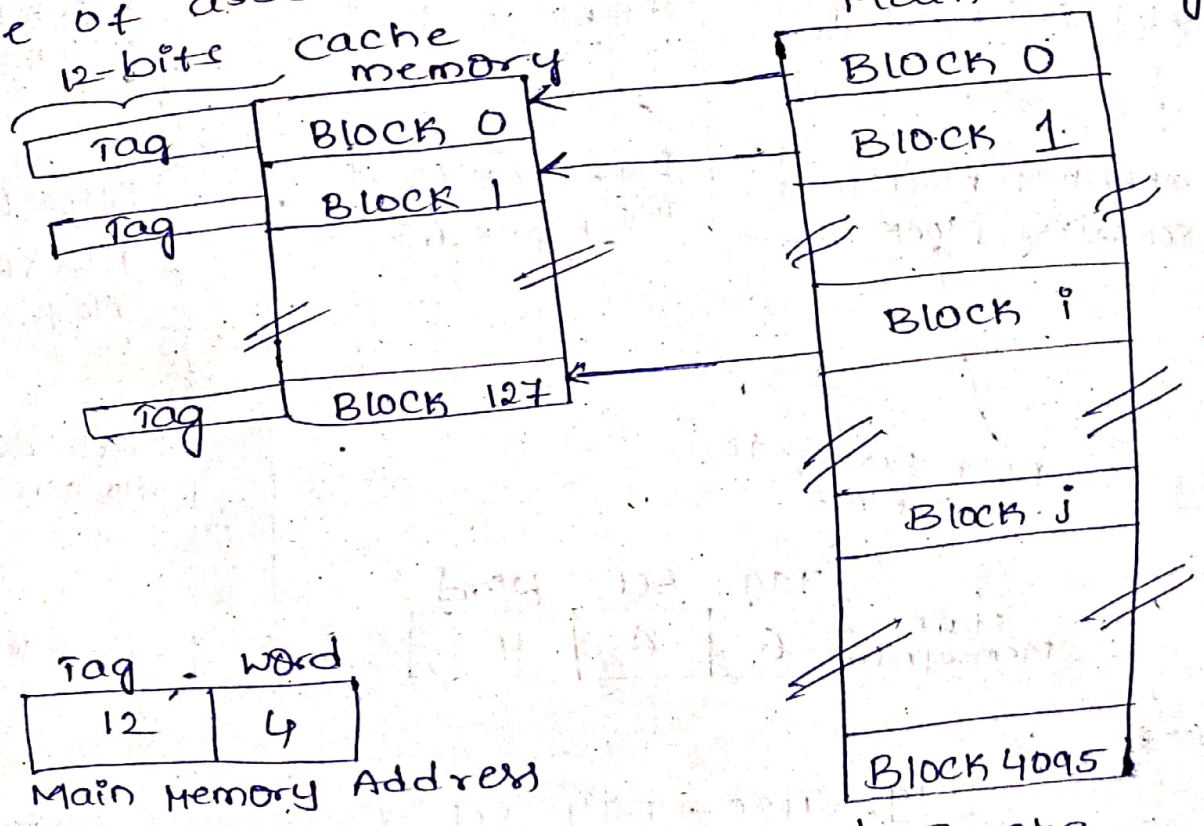


fig: Associative-Mapped Cache

Set - Associative Mapping :-

→ It is a combination of both direct & Associative Mapping

→ A block of data from any page in the main Memory goes in to a particular block location of any Direct Mapped cache.

→ Each page in the main memory is organized in such a way that the size of each page is same as the size of one directly mapped cache. It is known as the two way set-associative cache because each block from main memory has two choices for block placement.

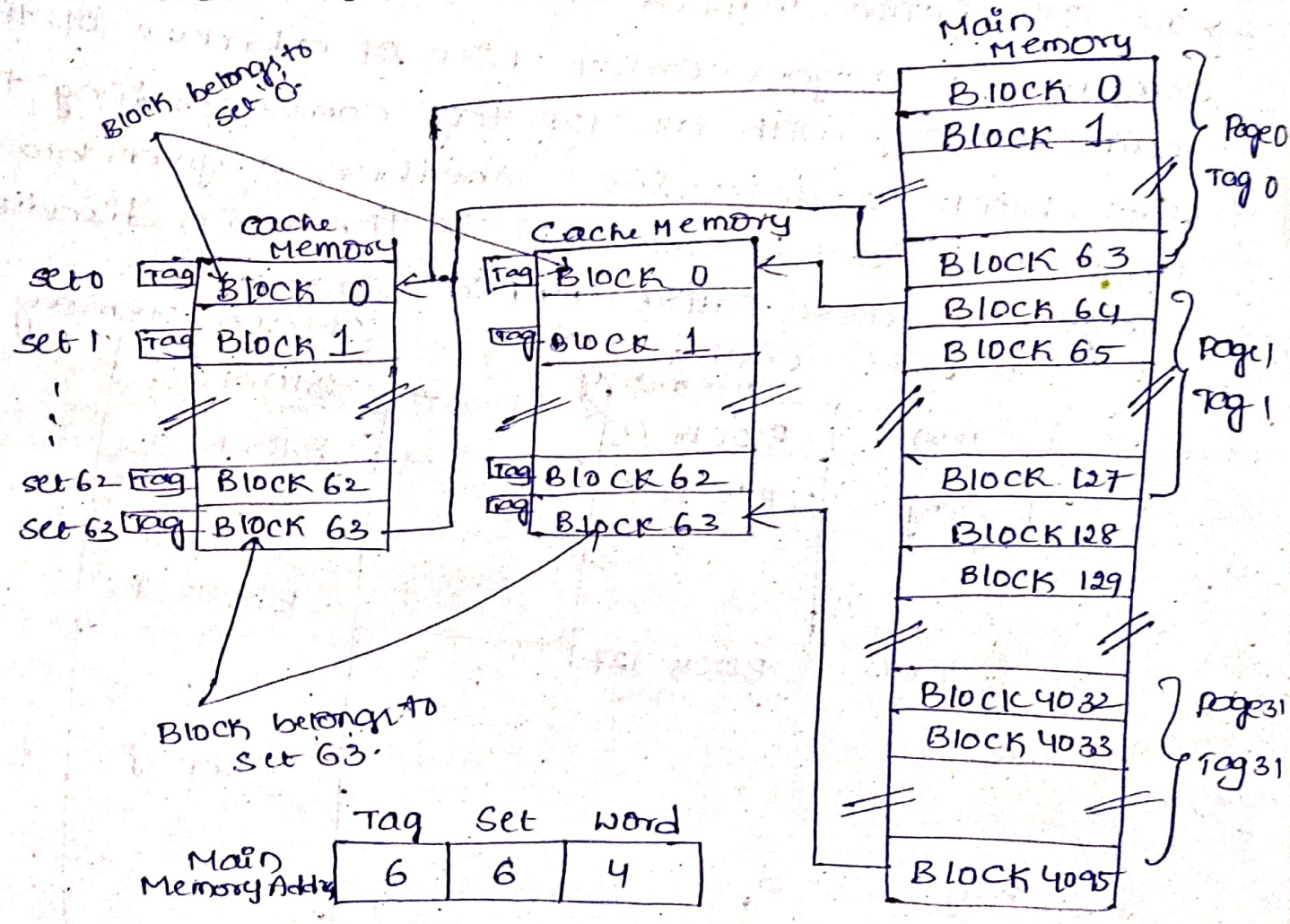


fig: Two-way set Associative Cache

→ In this technique block 0, 64, 128 --- 4032 of Main Memory can map into any of the two (blocks) blocks of set 0.

→ It is necessary to compare address of Memory with the tag bits of corresponding two blocks Locations of particular set.

→ In this we require two comparisions to determine whether a given block is in the cache.

→ In two-ways-set associative cache, the address is divided in to three fields, the 4-bit word field selects one of the 16 words in a block.

→ The set field needs 6-bits to determine the desired block from 64 sets.

→ There are 64-pages, so six tag bits are required.

3. Replacement Algorithms :-

When a New block is brought in to the cache, one of the Existing blocks must be Replaced by a New block.

Four Most Common Replacement Algorithms :-

1. Least-Recently used (LRU)
2. First-in-First-out (FIFO)
3. Least-Frequently-used (LFU)
4. Random

1. **Least-Recently used :-** In this technique, the block in the set which has been in the cache longest with no reference to it, is selected for the Replacement.

→ suppose we assume that more recently used memory Locations are more likely to be referenced again

FIFO: This techniques uses same concept that block implementation uses in the micro processors. In this technique, the block which is first Loaded in the cache among the present block in the cache is selected for the Replacement.

Least-Frequently-Used:- In this technique, the block in the set which has the fewest reference is selected for the replacement.

Random:- Here, there is no specific criteria for replacement of any block. The Existing blocks are replaced randomly.

4. Write Policy:- Also known as cache updating policy.

In cache system, two copies of the same data can exist at a time, one in cache & another in Main Memory.

→ If one copy is altered and other is not, two two different sets of data become associated with the same address.

→ To Prevent this, the cache system has updating systems such as: 1. Write through system
2. Buffered write through system
3. Write-back system

5. Block size:- It should be optimum for cache Memory system

6. Number of Caches:- When one-chip cache is in-sufficient the secondary cache is used.

→ The Cache design changes as No. of caches used in the system changes.

6. Performance Considerations :-

→ Two key factors in the commercial success are the performance and cost.

→ A common measure of success is called the "price Performance Ratio".

→ Performance depends on how fast the machine instructions are brought to the processor and how fast they are executed.

→ To acheive parallelism interleaving is used.
↓
both are accessed in the same manner.

1. Interleaving

→ If the main memory is structured as a collection of physically separated modules, each with its own ABR (Address Buffer Register) and DBR (Data Buffer Register), memory access operations may proceed in more than one module at the same time. Thereby the aggregate rate of transmission of words to and from the main memory system can be increased.

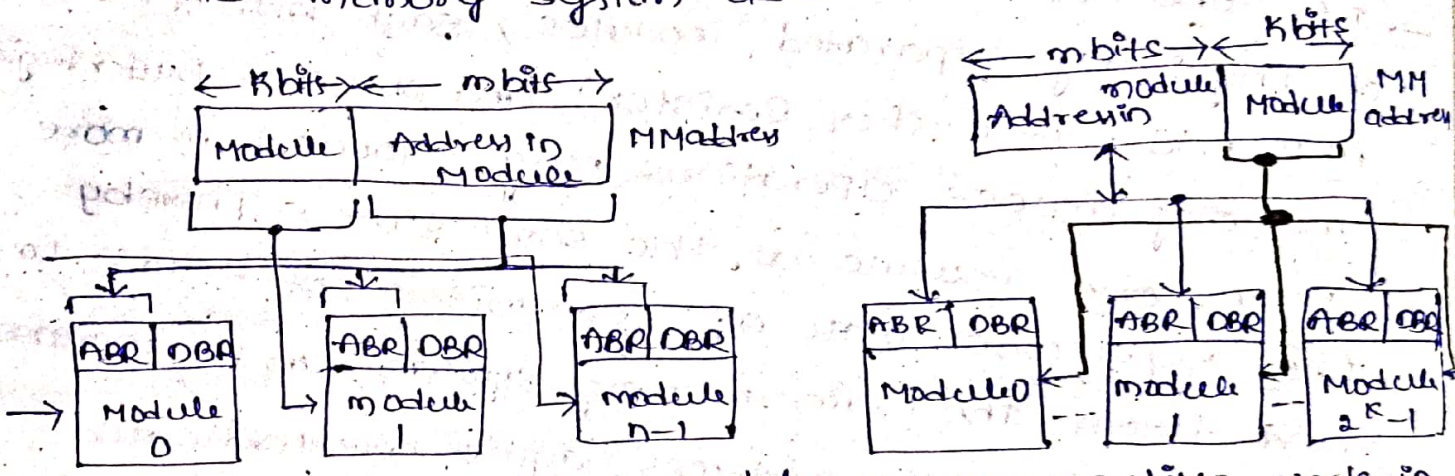
→ Two methods of address layout are indicated.

→ In the first case, memory address generated by the processor is decoded as shown in Part (a) of the figure. The high-order k-bits name one of 'n' modules and the low-order m-bits name a particular word in that module. When consecutive locations are accessed, only one module is involved. At the same time, devices with DMA ability may be accessing it in other modules.

→ In the second case, as shown in Part (b) of the figure, which is called "memory interleaving".

The low-order k bits of the memory address select a module, and the high-order m bits name a location with in the module.

Thus, any component of the system that generates requests for access to consecutive memory locations can keep several modules busy at any one time which results in both faster access to a blocks of data and higher average utilization of the memory system as a whole.



(a) consecutive words in a Module (b) consecutive words in consecutive modules

Fig: Addressing multiple-module memory systems

2. Hit Rate and Miss Penalty

An excellent indicator of the effectiveness of a particular implementation of the memory hierachy is the success rate in accessing ilf at various levels of the hierarchy. Recall that a successful access to data in a cache is called a hit. The no. of hits stated as a fraction of all attempted accesses is called

the hit rate, and the miss rate is the no. of misses stated as a fraction of attempted accesses.

High hit rates, over 0.9, are essential for high performance computers. The Extra time needed to bring the desired i/f into the cache is called the Miss penality.

Let h be the hit rate, M the miss penalty, that is, the time to access i/f in the main memory, and c the time to access i/f in the cache. The average access time experienced by the CPU is $hc+(1-h)M$.

3. caches on Proccessing chips

→ If separate caches are used, it is possible to access both caches at the same time, which leads to increased parallelism and, hence, better performance.

→ since, the size of a cache on the CPU chip is limited by space constraints, a good strategy for designing a high performance system is to use sucha chache as a primary chache.

4. Other Enhancements

→ For enhancing performance, there are several other possibilities. They are 1. Write Buffer
2. Pre-fetching & 3. Lookup-free.

1. Write Buffer:-

→ To improve performance, a write buffer can be included for temporary storage of write requests. The CPU places each write request into this buffer & Continues Execution of the next instruction.

2. Pre-fetching:- The action of pre-fetching stops other accesses to the cache until the pre-fetch is completed.

3. Lookup-free: A cache that can support for multiple outstanding misses is called lookup-free.

TOPIC 7: Virtual Memories

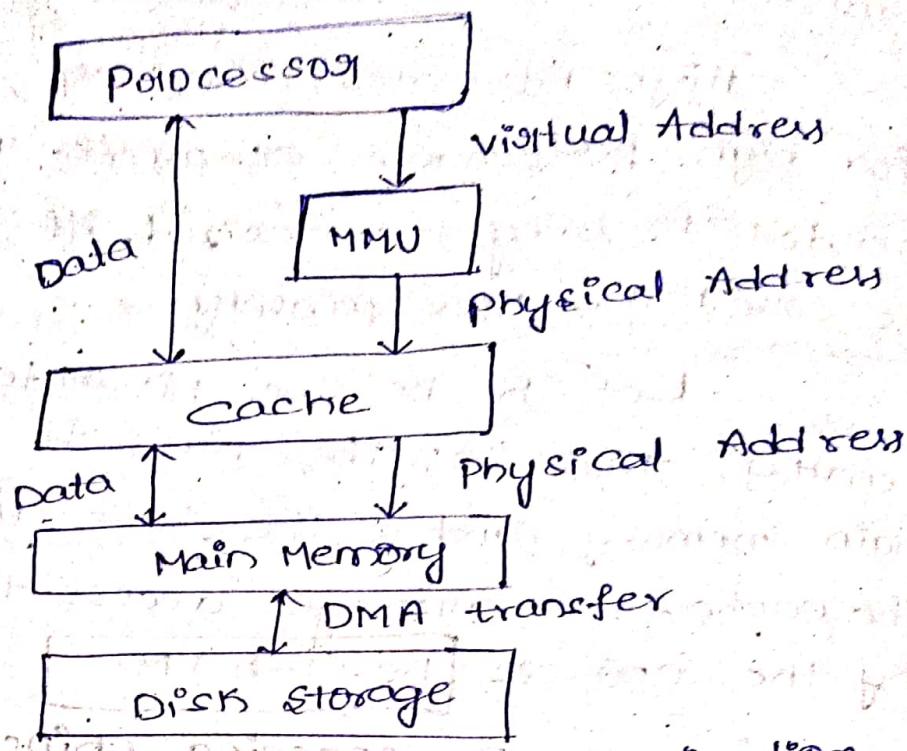


fig: Virtual Memory Organisation.

→ Virtual memory is an organization, which move data in & out of the physical memory. A processor that has the capability to provide 32-bit addresses will have about 4GB bytes of addressable space.

→ But, the physical memory is not that large enough & its size may be around few hundred MB to 1 GiB.

→ If a program is very large & can not be accommodated into the Main Memory, then those parts of the program that are not under Execution at a given time are placed in to the secondary Storage Memory called as Magnetic disks.

→ When ever those parts of the program need to be executed, they are Loaded in to the mainmemory by replacing the parts of the pgm already in the memory

→ The Operating system of modern computers will manage this movement of programs & data blocks from Main Memory to secondary memory & from secondary storage to main memory, without involving the Application Programmer.

→ The virtual memory techniques that move programs & data blocks from secondary storage to mainmemory when they need to be executed.

→ The processor provides the binary address for an instruction (or) data.

→ These binary addresses are called virtual (or) Logical addresses.

→ The hardware & software components together will translate these binary address into physical address.

→ If virtual address refers to a part of a program (or) a segment of data that is present in the main memory itself, then its contents are accessed easily.

→ But, when the virtual address, refers to a part of the program that is not present in the Main Memory then its contents are moved from secondary storage to Main storage, before they are accessed.

→ The MMU in the above figure, is a Memory Management unit which is Responsible for translating a given virtual address into its corresponding Physical address.

→ If the required data resides in the main memory, then it is read through cache & if the required data does not reside in the main memory, the MMU provokes the Operating system to get the data from the disk to the main memory.

→ The data is moved blw the main memory & the disk in the DMA scheme.

Address Translation:-

In virtual Memory, the address is broken in to virtual Page Number and a page offset.

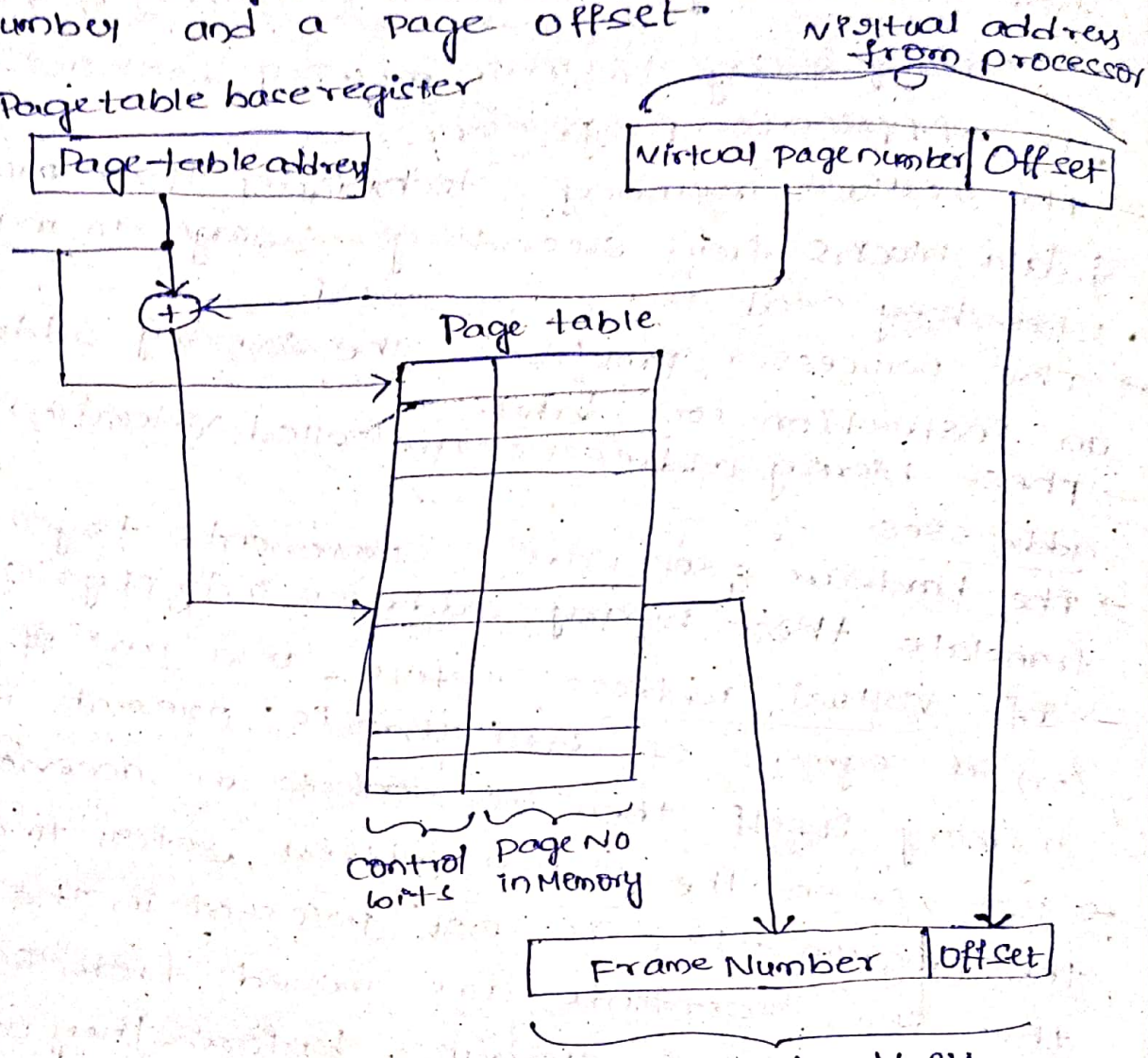


fig: Virtual to physical address translation.

→ The above figure shows the translation of the virtual page Number to physical page Number.

→ The physical Page Number contains the upper portion of the physical address, while the page offset, which is not changed, constitutes the Lower portion. The no of bits in the page offset field decides the Pagesize.

→ The page table is used to keep the info about the main memory location of each page.

→ To obtain the address of the corresponding entry in the Page table the Virtual Page Number is added with the contents of page table base register, in which the starting address of the page table is stored.

→ If the page required by the processor is not in the main memory, the page fault occurs and the required page is loaded in to the main memory from the secondary storage memory by special routine called Page fault routine.

Difference between cache & virtual Memory:

Cache Memory	Virtual Memory
1. Cache Memory is physically present.	1. Virtual memory is not physically present.
2. Cache Memory has hierarchy in levels. For eg: Level 1, Level 2, cache.	2. There is no hierarchy present in Virtual Memory.
3. The unit of Memory in cache Memory is called cache block.	3. The unit of Memory in virtual memory is called either a page (or) segment.

Topic 8: Memory Management Requirements

Management routines are part of the O.S of the computer. It is convenient to assemble the operating system routines into a virtual address space, called the system space.

The MMU uses a page table base register to determine the address of the table use to be used in the translation process.

Def of Memory Management :-

→ Subdividing memory to accommodate multiple process

Memory Management Requirements :-

1. Relocation :-

→ Programmer does not know where the program will be placed in memory when it is executed.

→ While the program is executing, it may be swapped to disk and returned to main memory at a different location.

2. Protection :-

→ processes should not be able to reference memory locations in another process with out permission.

→ Memory protection requirement must be satisfied by the processor (H.W) rather than the Operating system (S.W)

→ O.S can not anticipate all of the memory references a program will make.

3. Sharing :-

→ Allow several processes to access the same portion of memory.

→ Better to allow each process access to the same copy of the program rather than have their own separate copy.

4. physical organization :-

→ programmer does not know how much space will be available.

5. Logical Organization :-

→ programs are written in modules.

→ Modules can be written and compiled independently.

→ Share modules among processes.

Topic 9: Secondary Storage

→ Large storage requirements of most computer systems are economically realized in the form of magnetic disks, optical disks, and magnetic tapes, which are usually referred to as "secondary storage devices"

→ secondary storage is for any amount of data, from a few megabytes to peta bytes.

→ These devices store almost all types of programs and applications.

1. Magnetic Hard Disk

→ A magnetic disk is a storage device that uses a magnetization process to read, write, rewrite and access data.

→ It is covered with a magnetic coating and stores data in the form of tracks, spots and sectors.

→ Hard disks, zip disks and floppy disks are common examples of magnetic disks.

Organization and Accessing of Data on a Disk:

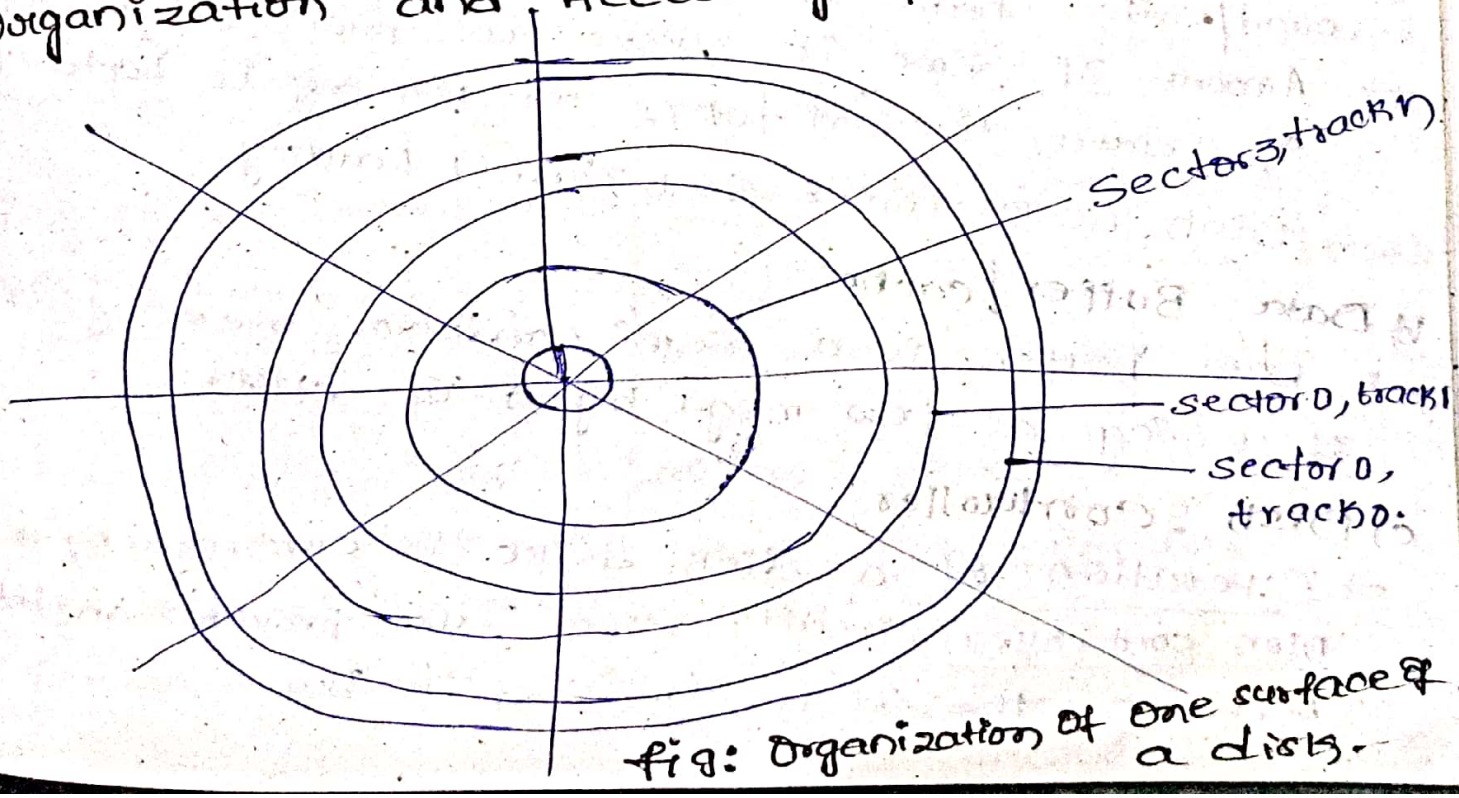


fig: Organization of one surface of a disk.

→ The organization of data on a disk is illustrated in above figure.
→ Each surface is divided into concentric tracks, and each track is divided into sectors.
→ The set of corresponding tracks on all surfaces of a stack of disks form a logical cylinder.
→ The data on all tracks of a cylinder can be accessed without moving the read/write heads.
→ The data are accessed by specifying the surface number, the track number, and the sector number.
→ The Read and write operations start at sector boundaries.
→ In the above fig: each track has the same no of sectors. So all tracks have the same storage capacity.

a). Access Time :-

Def: The sum of the two delays is called the Disk access time. First one is 1. Seek Time 2. Rotational Delay (or) Latency.

→ Seek Time means is the time required to move the read/write head to the proper track.
→ Amount of time it takes to transfer a word, of data to (or) from the Memory is Latency

Disk Access Time = Seek Time + Latency.

b) Data Buffer/cache

→ Data Buffer is a semi conductor memory, capable of storing a few mega bytes of data.

c) Disk Controller

→ Operation of a disk drive is controlled by a Disk controller circuit, which also provides an interface between the disk drive & the bus.

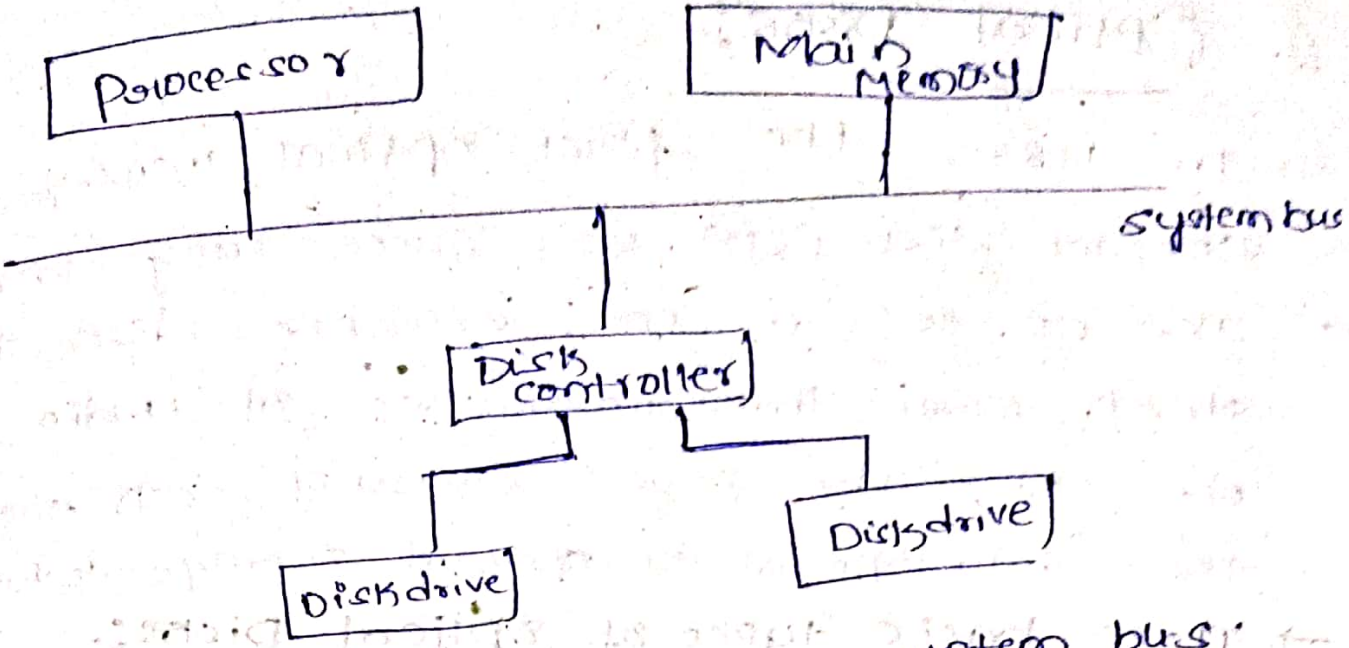


fig: Disk connected to the system bus.

→ The Disk controller may be used to control more than one drive. above fig shows a disk-controller which controls two disks drives.

7) SCSI Disk vs RAID DISK

SCSI	RAID
Small computer system Interface	
1. Disks having SCSI interface are caued SCSI disks.	1. RAID is the redundant array of inexpensive disks which operate independently & in parallel to improve the speed of data transfer.
2. They exhibit better performance.	2. The exhibit excellent performance & provide larger reliable storage.
3. These disks are used in appn's where there is a large no. of files requests for small files.	3. These disks are used in large-size computers.

Q. Optical Disks:-

→ In 1983, the first optical memory device, Compact disk (CD) was successfully launched.

→ The CD is a non-erasable disk that can store more than 60 minutes of audio ilf on one side. The huge success of CD encouraged the development of optical storage technolgy.

→ Three basic types of optical Disks:-

1. Compact Disk Read-only Memory (CD-ROM)
2. Write-once Read-many (WORM)
3. Erasable optical disks.

1. CD-ROM:-

→ A CD-ROM is a pre-pressed optical compact disk that contains data. computers can read but not write to or erase.

2. WROM: Write-once read many

→ The WORM storage provides one-time writing but unlimited reading of data.

→ Data can not be over written or erased but it can be updated.

3. Erasable optical Disk:

→ Erasable optical media are used in app^ns where stored data require frequent changes.