27|Dec|2024

Unit-IV

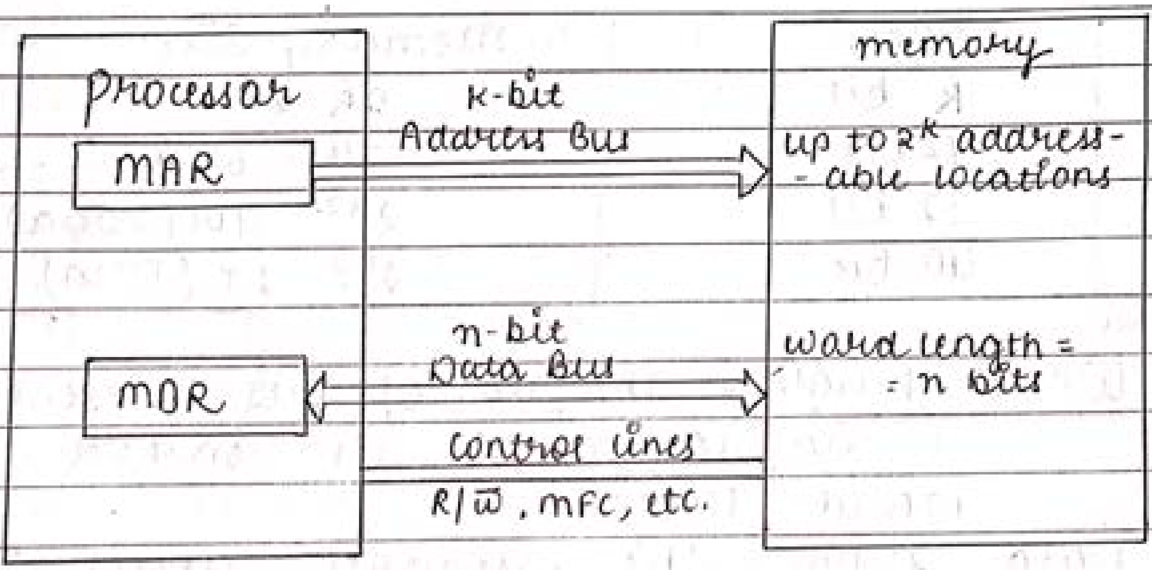
Inst - opcode
data - operand

Memory

Basic concepts -

One of the major advantage of computer is its storage capacity, where huge amount of info. can be stored.

- memory is the storage space in the computer where the data and insts. are stored.
- Each location of the memory has a unique address, which varies from 0 to (memory size -1).
-



If the computer has - 64K words, then this memory unit has

$= 64\text{K}$

$= 64 \times 1024$

$= 2^6 \times 2^{10}$

$= 2^{16} = 65536$ memory locations.

Then, to address above size of memory locations, we need 16 bit of address bus. To deal with the memory organization, some basic concepts about memory systems are given as—

i) **Maximum Size** – maximum size of the memory that can be used is dependent on the no. of address lines, that is present in the system. Therefore, K bit address lines are required to address M-size memory, where $\boxed{M = 2^K}$

K → no. of address lines.

Address	Memory Size.
K bit	2^K
16 bit	$2^{16} = 64K$
32 bit	$2^{32} = 4G$ (Giga)
40 bit	$2^{40} = 1T$ (Tera)

ii) **Word Length** - The no. of bits present in a word is called word length.

- Data from the memory system is accessed, based on the word length.

For example,
a system with word length of 32-bit can access 32 bit of data on a single access.

$\overset{\ulcorner \text{ address}}{2^K} \times n$ no of bit

- In the case of 64K X 32 memory size has a word length of 32-bit and ~~a~~ $64 = 2^{16}$, then we need a 16-bit address bus to access the each location.

iii) <u>Data Transfer</u> – Data ~~pro~~ Transfer between the memory and the processor takes place through two processors register (MAR and MDR)

→ K-bit MAR and n-bit MDR indicates that the memory unit may contain upto 2^K addressable locations and n-bit of data can be transfer b/w the memory and the processor.

→ Therefore, the processor has K-bit address bus and n-bit data bus.

→ The control bus include the control line for read/write or $(R/\overline{W})$ and memory function completed (MFC) for ~~que~~ co-ordinating data transfer.

iv) <u>Reading a Data</u> – The processor read data from the memory by loading address of the required memory location into the MAR regi. and set $R/\overline{W} = 1$.

- The memory responses by placing the data from the address location and the data line and then place the MFC signal.
- On receiving the MFC signal, the processor loads the data from the data line into the MDR register.

5) Writing the data – The processor write data into a memory location by loading its address into MAR and loading the data from MDR into the memory. The write operation is indicated by $R/\overline{W} = 0$.

Ques- Consider a CPU which has 13 bit address bus and 16 bit data bus. Determine maximum size of memory which can be supported by CPU.

Soln- address bus = 13

$k = 13$

Data Bus = 16

$n = 16$

memory size $= 2^k \times n = 2^{13} \times 2^1 \times 8$ bit

$= 2^{13} \times 2 \times 8$ bit

$= 2^{13} \times 2B$

$= 2^{10} \times 2^3 \times 2^1 \times B$

$= 16KB$

Ques- A 32 wide memory has 24 bit addresses to be accessed. Determine maximum memory capacity in MB.

Soln- $n = 32$

Address line = 24

$k = 24$

memory size $= 2^k \times n$

$= 2^{24} \times n$

$= 2^{24} \times 32$

$= 2^{24} \times 4 \times 8$ bit

$= 2^{24} \times 4B$

$= 2^{20} \times 2^{4} \times 4B$

$= 2^{4} \times 2^{20} \times 2^{2} \times B$

$= 64MB$

Memory System in a Computer:-

To have fast and uninterrupted access to external memory where the program and data are stored. The processor can operate its maximum speed but the memory are not responding the processor at the same speed.

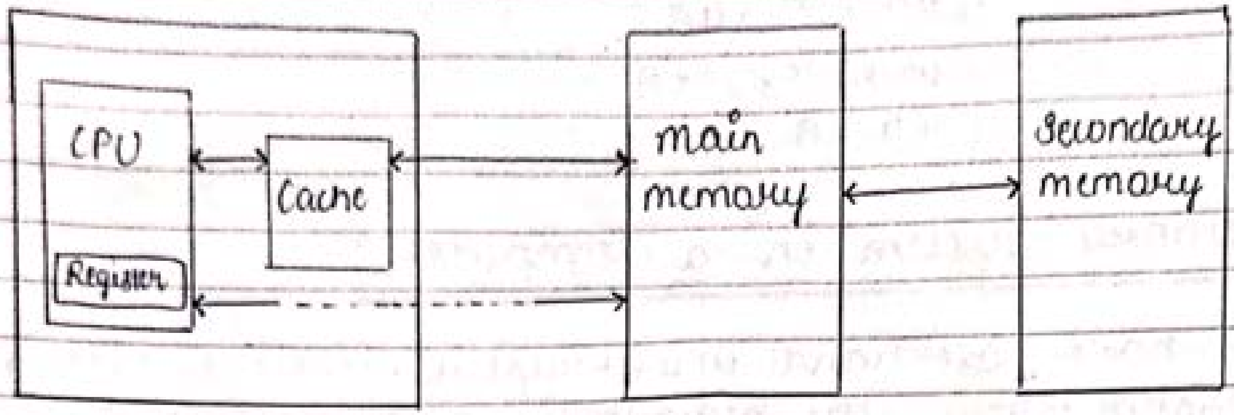
- To overcome this speed gap, the info. is distributed over memory unit and having different performance and cost.
- The components used for information change storage can be classified into four main groups-

1) Processor Register - temporary storage for inst. and data within the processor.

2) Main memory - store program and data that are in active used.

3) Cache memory - stores the data that are frequently used by processor currently.

4) Secondary memory - store system program and large data file.

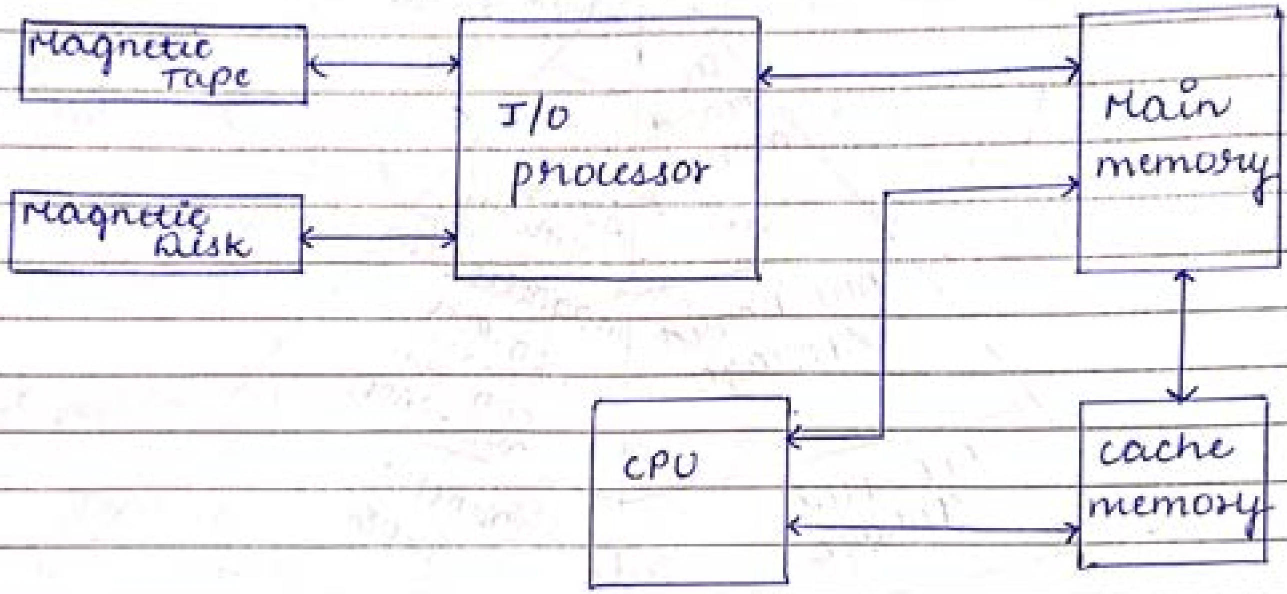


Memory Heirarchy :- The memory heirarchy is a structured arrangement of different types of memory in a computer system. organised based on speed, cost and size.

→ As the storage capacity of memory increases, the cost per bit for storing binary info. decreases and the access time of memory becomes longer. Basically, it is necessary to balance the trade off b/w performance (speed or access), cost and storage capacity.

- Without a memory heirarchy, computer system would ~~have~~ struggle to achieve high performance and efficient resource utilization.
- The memory heirarchy system consist of all storage devices employed in a computer system from the slow but high capacity auxiliary memory to a relatively faster main memory to an even smaller and faster cache memory, accessible to the high speed processing logic.

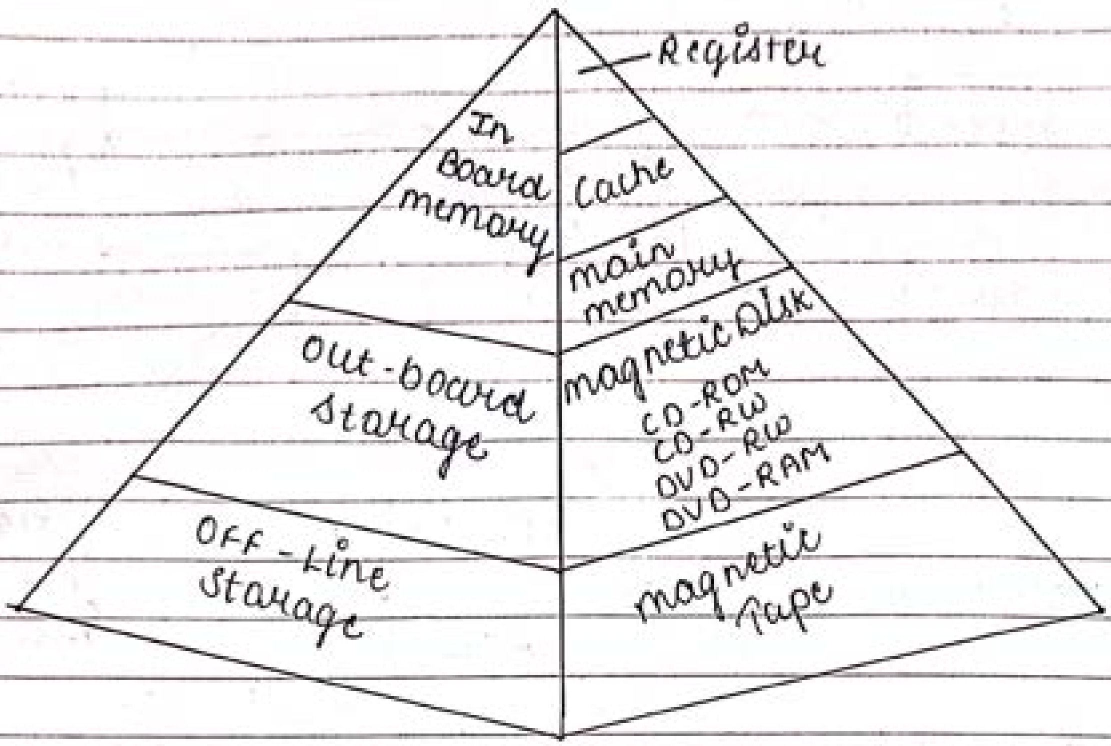
Auxiliary Memory

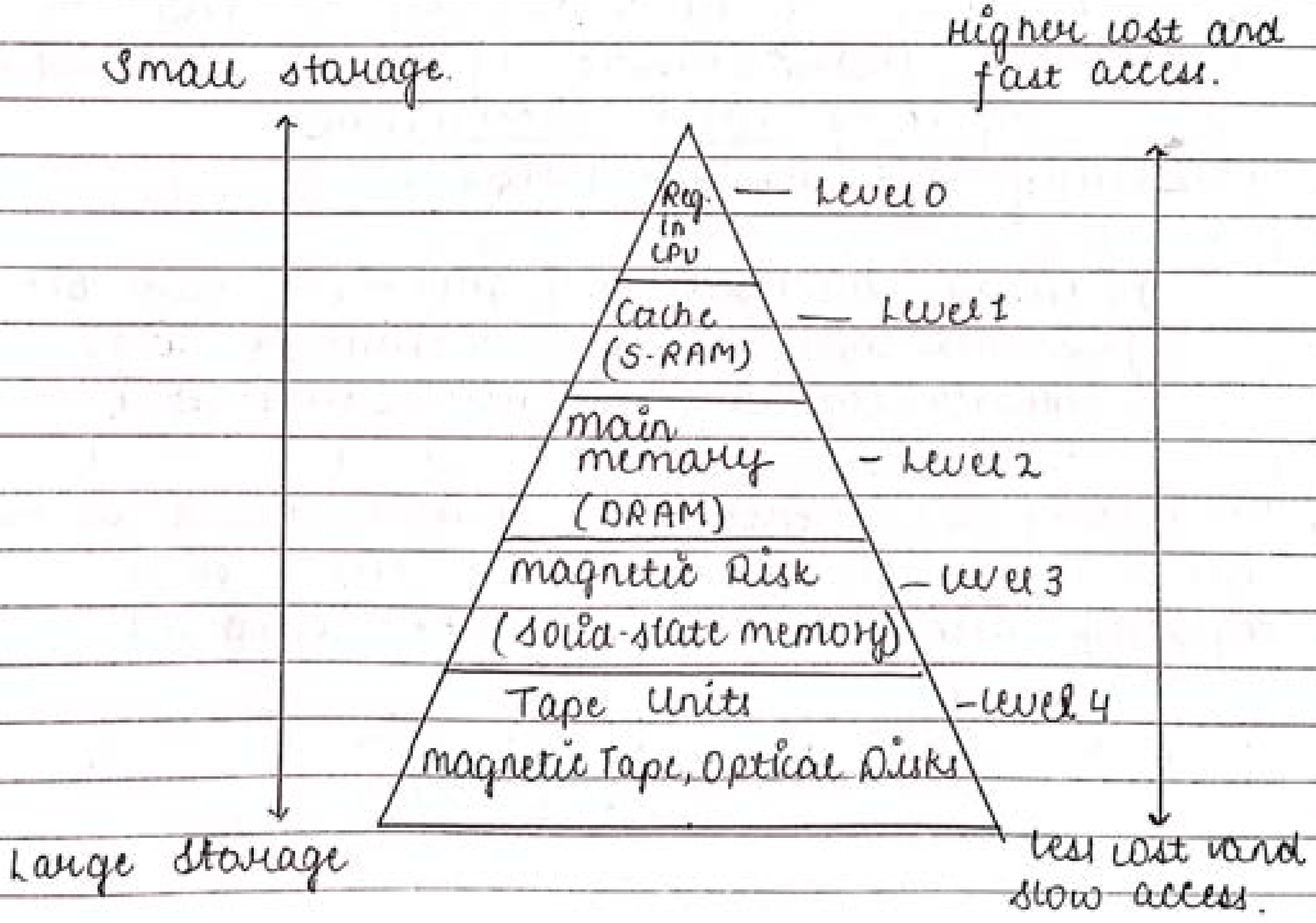


- At any given time, a variety of technologies are used to implement memory systems.
- Generally, there is a trade off among the three key characteristics of memory, namely, cost, capacity and access time.
- Following relationship holds –

1) Faster access time, greater cost per bit.
2) Greater capacity, smaller cost per bit.
3) Greater capacity, slower access time.

- To meet performance requirement, the designer needs to use expensive, relatively lower capacity memories with short access time.


Register
In Board memory
Cache
main memory
Out-board Storage
magnetic Disk
CD-ROM
CD-RW
DVD-RW
DVD-RAM
Off-Line Storage
magnetic Tape


Small storage.
Higher cost and fast access.
Reg. in CPU
— Level 0
Cache (S-RAM)
— Level 1
main memory (DRAM)
- Level 2
magnetic Disk (Solid-state memory)
— Level 3
Tape Units
magnetic Tape, Optical Disks
-Level 4
Large Storage
less cost and slow access.

- The memory in a computer can be divided. into five heirarchy based on the speed as well as use.
- The processor can move from one level to another based on its requirement.
- The memory are registers. cache, main memory magnetic disk and magnetic tape.

Register (Level-0):- The registers are present inside the CPU. since, they are inside CPU, they have least access time.

- Reg's are most expensive and smallest in size, generally in kilobytes (KB).
- They are implemented by using flip-flop.

Cache (Level-1):- Cache memory is used to store the segments of a program that are frequently accessed by the processor.

- It is expensive and smaller in size, generally in megabytes (MB) and is implemented by using Static-RAM.
- Basically. it behaves as a buffer b/w the CPU and main memory

Main memory (Level-2):- It is also known as primary memory.

- It directly communicates with the CPU and with auxiliary memory devices. through an I/O processor.

- It is the main storage unit of the computer.
- main memory is less expensive than cache memory and larger in size, generally in gigabytes (GB)
- It is implemented by using D-RAM.

Secondary Storage (level-3):-

Secondary storage devices (magnetic Disk) are used as a backup storage.

- They are less costly than main memory and larger in size (fewTB)
- It is slow as compared to all the other memory types.

Tape Units (Level-4):-

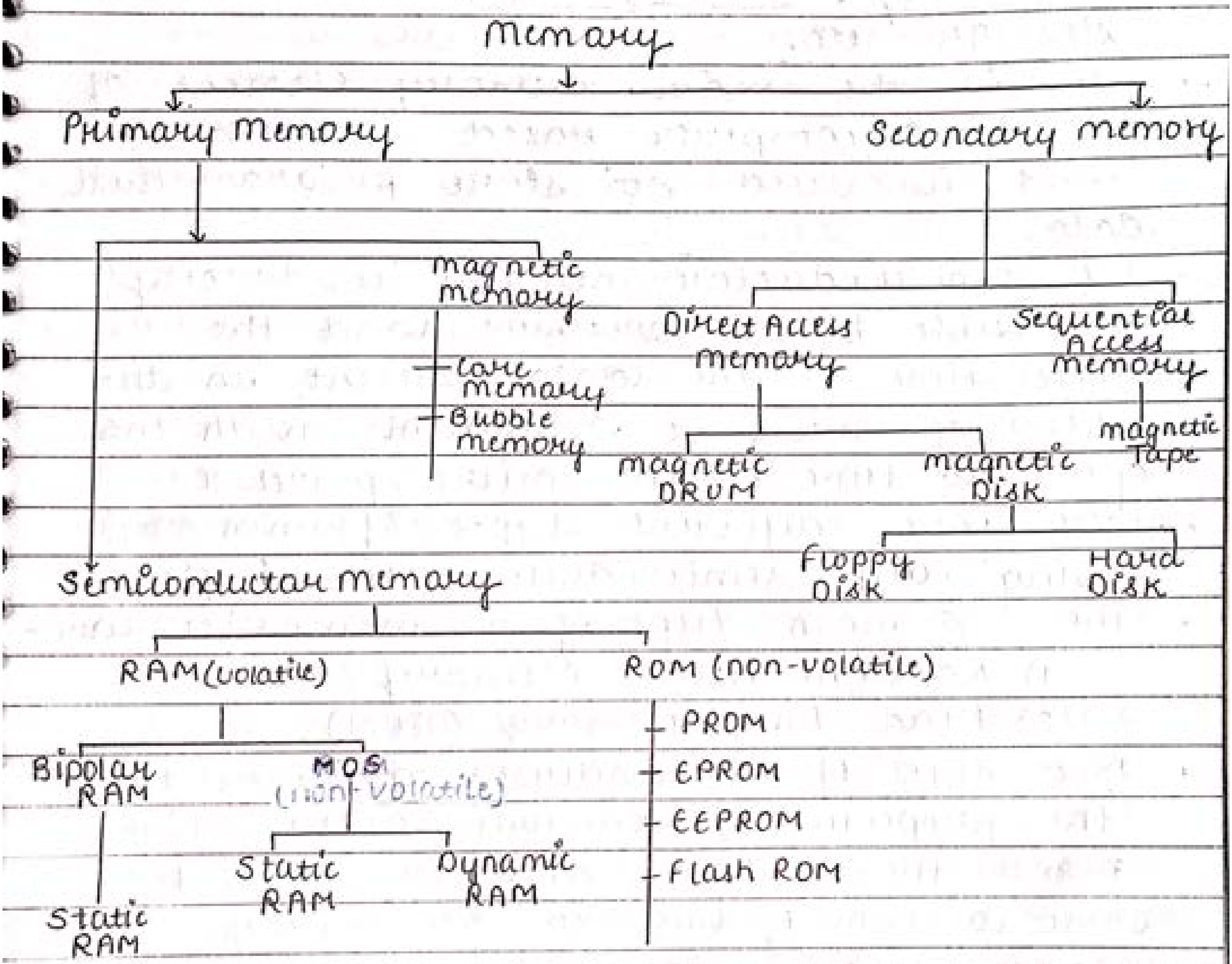
It is also known as tertiary storage.

- Tertiary storage devices like magnetic tape are present at level-4, mainly used to back-up huge data.
- They are used to store removable files and are the cheapest and largest in size. (1-20TB).

Advantages of memory Hierarchy:-

- memory Distribution is simple and economical.

- Reduce average cost per bit of entire memory system of computer.
- Improve the performance.
- It maintains avg data transfer rate of entire memory system.
- Energy efficient.



Semiconductor Memory -

It is a semi-conductor device used for digital data storage in the computer.

- It is also known as integrated circuit - memory, memory chip, semiconductor storage, etc.
- It is the main memory element of a micro-computer based system and is used to store program and data.
- The semi-conductor memory is directly accessible by the processor and the access time of the data present in the memory must be compatible with the operating time of the micro-processor.
- There are different types of memory using diff. semiconductor technologies.
- The 2 main types of " memories are-
  1) Random Access Memory (RAM)
  2) Read Only Memory (ROM)
- Most types of semiconductor memory have the property of Random access which ~~makes them~~ means that it takes the same amount of time to access any memory location so, data can be efficiently accessed in any random order.

- It also has much faster access time than other type of data storage.
- It is used for main memory of the computer to hold data.

Random Access Memory (RAM) :-

RAM is present on the motherboard and the computer's data is temporarily stored in the RAM.

- RAM is a form of semiconductor memory technology that is used for reading and writing data in any order.
- RAM is a volatile memory, which means it is present as long as the computer is in ON state, as soon as the comp. turns OFF, and the memory is erased.
- **Features of RAM :-**

RAM is known as the primary memory of the computer.

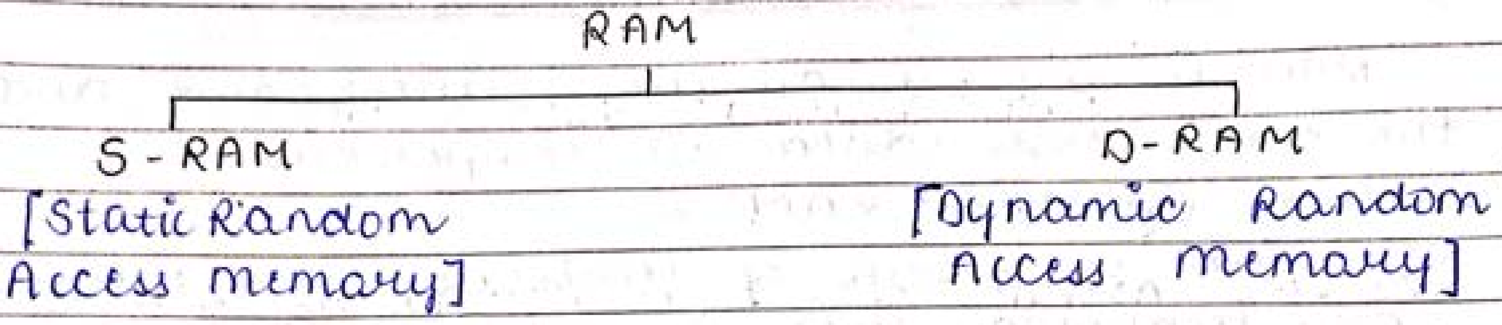
RAM is known to be expensive, since the memory can be accessed directly.

RAM is the fastest memory. Therefore, it is an internal memory for the computer.

The speed of the computer depends on RAM, if the comp. has less RAM, it will take more time to load the data and process will slow down.

Types of RAM :-

The RAM family includes two impartant memary devices. The primary difference between them is the life-time of data they store



1) D-RAM:-

D-RAM is the farm of semiconductar memary that is used in equipment including PCs, where it farms the main RAM of computer.

- D-RAM uses a capacitar to stare each bit of data and the level of charge on each capacitar determines whether that bit is logical 1 or 0.
- However, these capacitors do not hold their charge indefinitely and therefore, the data needs to be refreshed periodically. Therefore, this dynamic refreshing of the capacitar gives the name, known as Dynamic-RAM.

Disadvantages of D-RAM –

- Complex manufacturing process.
- Data reqquires refreshing.
- More complex external circuitry requirced.
- (read and refresh periodically)
- valatile memary
- relatively slow operational speed.

2) S-RAM – SRAM stands for Static RAM.

This form of semiconductar memory, unlike DRAM the data does not need to be refresh dynamically. These semiconductar devices are able to support faster read and write times than DRAM. However they consume mare power they are less dense and mare expensive than DRAM.
As a result of thisSRAM, is narmally used far cache memary while DRAM is used as the main semiconductar memary.

Advantages of using SRAM –

- Speed :- RAM is faster than ather types of starage like ROM.
- Multitasking - More RAM allows a computer to handle multiple application simultaneously without slow down.
- Flexibility - RAM can be easily upgraded to enhance computer performance.

- Volatile storage- RAM automatically clears its data when the computer is turned OFF reducing the risk of unwanted data accumulation.

Disadvantages of RAM —

- Cost — RAM is more expensive as compared to other storage.

- Limited storage- RAM has a limited capacity so that it can't store large amount of data permanently.

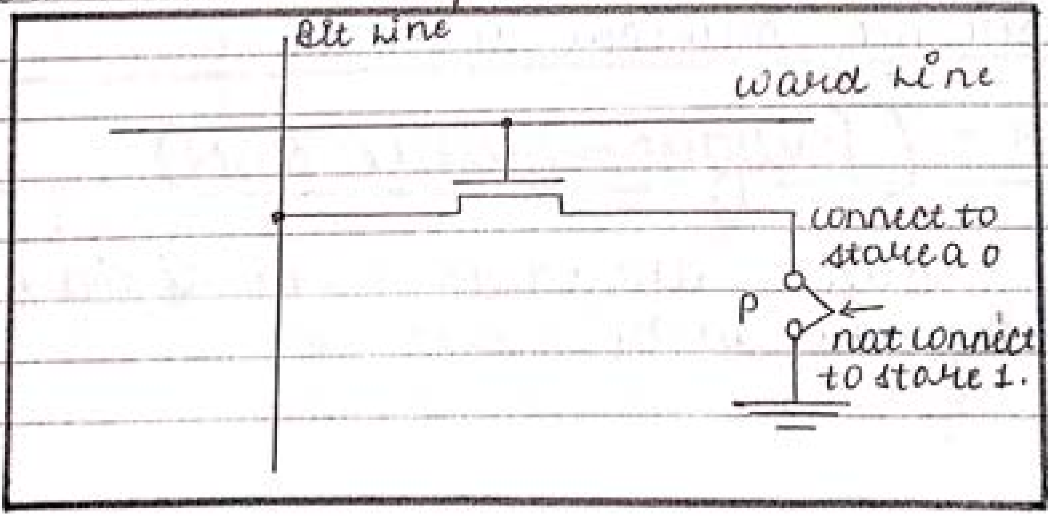
- Volatile — Data storage is also lost when the computer is turned OFF which means important data must be stored to permanent storage.

- Power consumption- RAM requires continuous power to retain data contributing in overall power consumption of the device.

- Physical space- Increasing RAM requires physical space in the computer which might be limited in smaller devices like laptop and tablet.

ROM [Read only Memory]:-

- ROM stands for Read Only Memory.
- The memory from which we can read but cannot write on it.
- This type of memory is non-valatile. The info. is stored permanently in such memories during manufacture.
- Many application requires non-valatile memory (which retain the stored information if power is turned off).
- Example - OS software has to be loaded from disk to memory which requires prog. that boots the OS. It requires non-valatile memory.
- Since, the normal operation involves only reading of stored data, a memory of this type is called ROM.
- So in general ROM is a form of semiconductor memory technology used where the data is written once and then not changed so it is used where data needs to be stored permanently, even when the power is removed.



Types of ROM –

Different types of non-valatile memary are —

1) MASK ROM
2) PROM
3) EPROM
4) EEPROM
5) Flash Memary

1) Mask Rom :-

- In this type of ROM, the specification of ROM is taken by the manufacturer from the customer in tabular farm in a specified farmat and then makes corresponding masks far the paths to produce the desired output.
- This is costly, (only if large quantity of the same ROM is required.

Uses :- They are used in networking OSs, server OSs, staring of fonts far laser printers, sound data in electronic musical instruments.

2) PROM - (Programmable ROM)

- PROM allows the data to be loaded by the user.

- It is first chip prepared as a blank memory, and then it is programmed to store the info.
- The difference b/w PROM and mask ROM is that PROM is manufactured as memory and programmed after manufacturing, where as a mask ROM is programmed during the manufacturing process.

- To program the PROM, a PROM programmer or PROM burner is used. The process of programming a PROM is called as <u>burning the PROM.</u>

  Also, the data stored in can't be modified. So, it is called as <u>one-time programmable device.</u>
- Programmibility is acheived by inserting a "fuse" at point P in a ROM cell.
- Before, it is programmed, the memory contains all 0's.
- The user can insert 1's at the rec location by burning out the fuse at these locations using high-current pulse. This process is "irreversible".

Merit:-

- It provides flexibility.
- It is faster.
- It is less expensive because they can be programmed directly by the user.

Uses:-

They have several different applications, including cell phones. video game consoles, medical devices. etc.

3) EPROM [Erasable Programmable ROM]:-

- It stands for Erasable programmable ROM.
- It overcomes the disadvantages of PROM that once programmed, the fixed pattern is permanent and can't be altered.
- If a pattern bit has been established, the PROM becomes unusable, If the bit pattern has to be changed. The problem has been overcome by EPROM, as when EPROM is placed under a special UV light for a length of time, the shortwave radiation makes the EPROM return to its initial state, which then can be programmed accordingly.
- Again for & erasing the content, PROM programmer or PROM burner is used.

Uses:-

Before advent of EEPROMs, some micro-controllers, like some versions of Intel 8048, the freescale 68HC11 used EPROM to store their program.

Merits -

- It provides flexibility during development phase of a digital system
- It is capable of retaining the stored information for a long time.

Demerits -

- The chip must be physically removed from the circuit for reprogramming and its entire contents are erased by UV light.

4) EEPROM -

- This is an Electrically Erasable programmable ROM.
- It is similar to EPROM, except in that case, the EEPROM is returned to its initial state by application of an electrical signal, in place of UV light.
- Thus, it provides the ease of erasing, as this can be done, even if the memory is positioned in computer. It erases or writes one byte of data at a time.
- Data can be written to it and it can be erased using an electric voltage. This is typically applied to an erase pin on the chip.
- Like other types of ROM, EEPROM retains the contents of memory even when the power is turned off. Also like other types of

EEPROM is not as fast as RAM.

- EEPROM memory cells are made from floating-gate MOSFETS (FGMOS).

- **Uses**- It is used for storing computer system BIOS.

Merits-

- It can be both programmed and erased electrically.
- It allows the erasing all cell contents selectively.

Demerits - It requires diff voltage for erasing, writing and reading the stored data.

5) Flash Memory:-

- Flash memory may be considered as development of EEPROM technology.
- The diff b/w EEPROM and Flash memory, is that in EEPROM, only 1byte of data can be deleted or written at particular time, whereas, in flash memory, blocks of data (512bytes) can be deleted or written at a particular time. So, Flash ROM is much faster than EEPROM.
- Data can be written to it and it can be erased, although, only in blocks, but data can be read on an individual cell-basis.

- To erase and re-program areas of chip, programming valtages, at levels there are available in electronic equipments are used. It is also non-valatile, and this makes it partiaularly useful.
- As a result, flash memory is widely used in many applications, using memory cards for digital cameras, mobile phones computer memory sticks and other applications.
- Flash memory stores data in an array of memory cells. The memory cells are made from floating-gate MOSFETS (FGMOS) These FGMOS have the ability to store an electrical change for extended periods of time (2-10 yrs) even without connecting to a power supply.

Disadvantages of Flash Memory :-

- Higher cost per bit than hard drives.
- Slower than other forms of memory.
- Limited number of write/erase cycles.
- Data must be erased before new data can be written.
- Data typically erased and written in blocks.

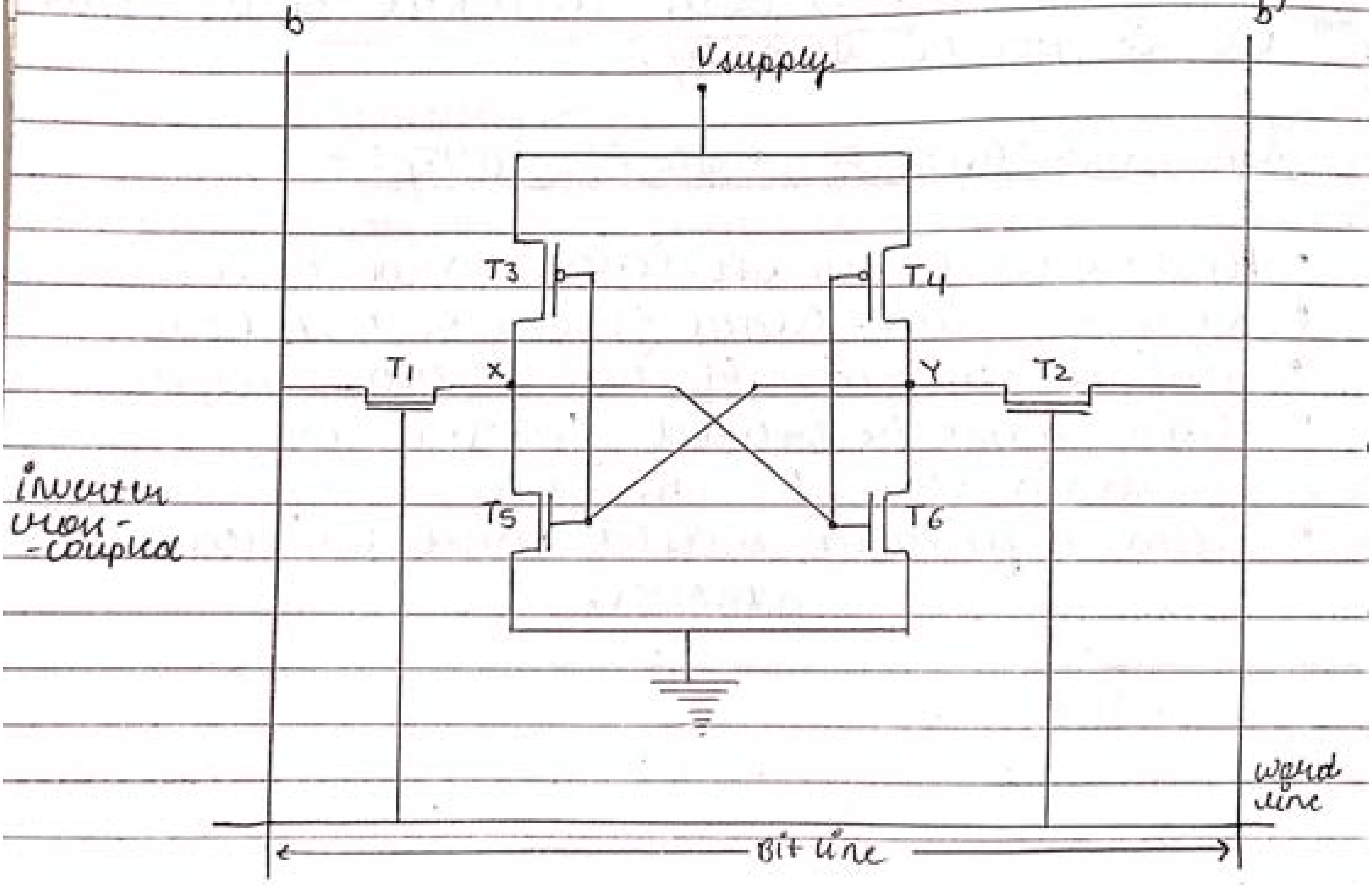
{ C → complementary }

Semiconductor RAM Memory:-

Semiconductar memories are available in a wide range of speed. Their avg cycle time range from 100ns to less than 10ns.

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

Following figure shows the S-RAM cell implemented with the help of CMOS.



latch → flip flop

- Two inverters are cross-connected to form a latch.
- The latch is connected to two ~~2~~-bit lines by transistors T1 and T2.
- These transistors behave as a switches that can be open or closed under the control of the word line.
- When the word line is at ground level(0V) the transistors are OFF and the latch retains its state.

Read Operations:-

In order to read the state of the SRAM cell. The word line is activated to close switches T1 and T2. ~~If it~~
If the cell is in the state '1', the signal on the bit line b is high and signal on bit line b' is low. The opposite is True, If the cell is state 0'. Therefore, b and b' are always complement to each other.

- The sense/ write circuit at the end of the 2- bit lines monitor their state and set the corresponding output accordingly.

Write Operations:-

During the write operation, the sense/ write circuit drives bit lines b and b'

- During the write op

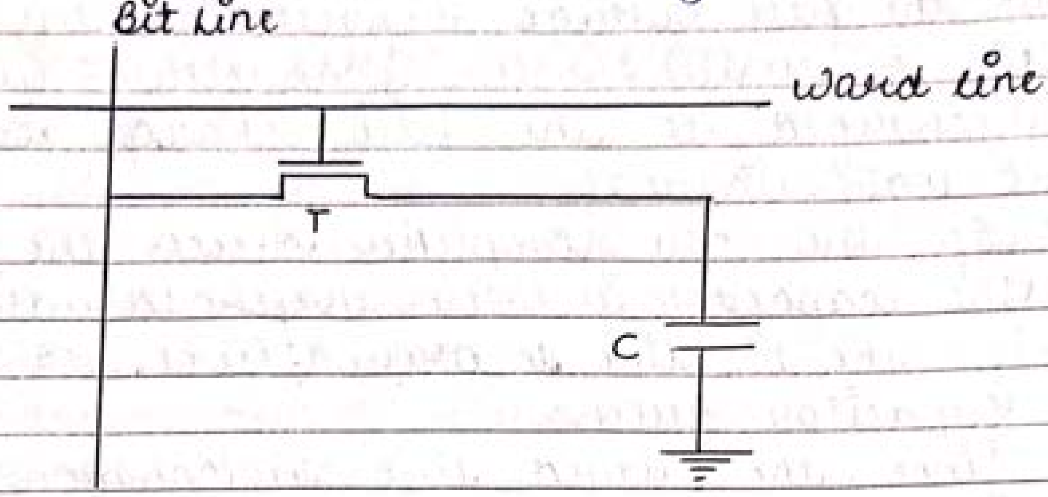
instead of sensing their state.
- It places the appropriate value on bit line b and its complement on b'. and activate the word line.
- This forces the cell into the corresponding state which the cell retains when word line is deactivated.

Dynamic Ram -

- Static RAM are fast but their cells requires several transistors.
- Less expensive and higher density RAMs can be implemented by simpler cells but these cells do not retain their state(0,1) for a long period unless they are accessed frequently for read or write operation.
- Memories that used such cells are called DRAM.
- Information is stored in dynamic memory cell in the form of charge on a capacitor but this charge can be maintained only for tens of millisecond.
- Since, the cell is required to store info for a much longer time. So, its content must be periodically refreshed by restoring the capacitor charge to its full value. This occurs when the contents of the cell are

read or written into it.

- A DRAM cell that consist of a capacitor and a transistor is shown in the fig below:—



To store the information in this cell, transistor 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.

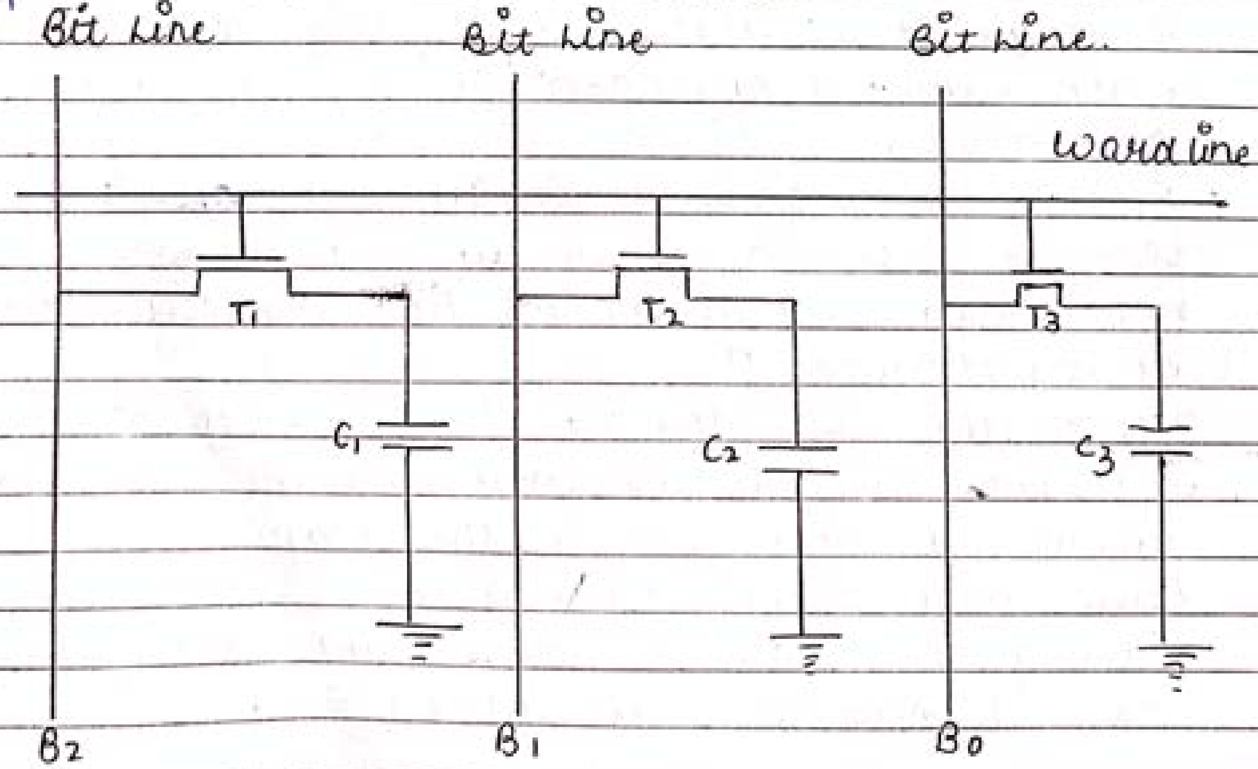
After the transistor is turned OFF, the charge remains store in the capacitor but not for a long time. So, the capacitor begins to discharge.

Hence, the info stored in the cell can be retrieved correctly only if it is read before the charge in the capacitor drops below some threshold value.

During a read operation, the transistor in a selected cell is turned ON,

A sense amplifier connected to the bit line detect whether the charge store in the is above or below threshold value.

- If the charge is above threshold value, the sense amplifier drives the bit-line to the full voltage representing the logic 1. As a result, the capacitor is recharged to the full charge corresponding to logic value 1.
- If the sense amplifier detects the charge in the capacitor is below threshold value, it pulls the bit-line to ground level, to discharge capacitor fully.
- Since, the ward line is common to all the cells in a row, therefore, all cells in a selected row are read and refreshed at the same time.



Internal organization of Memory chips:-

OR

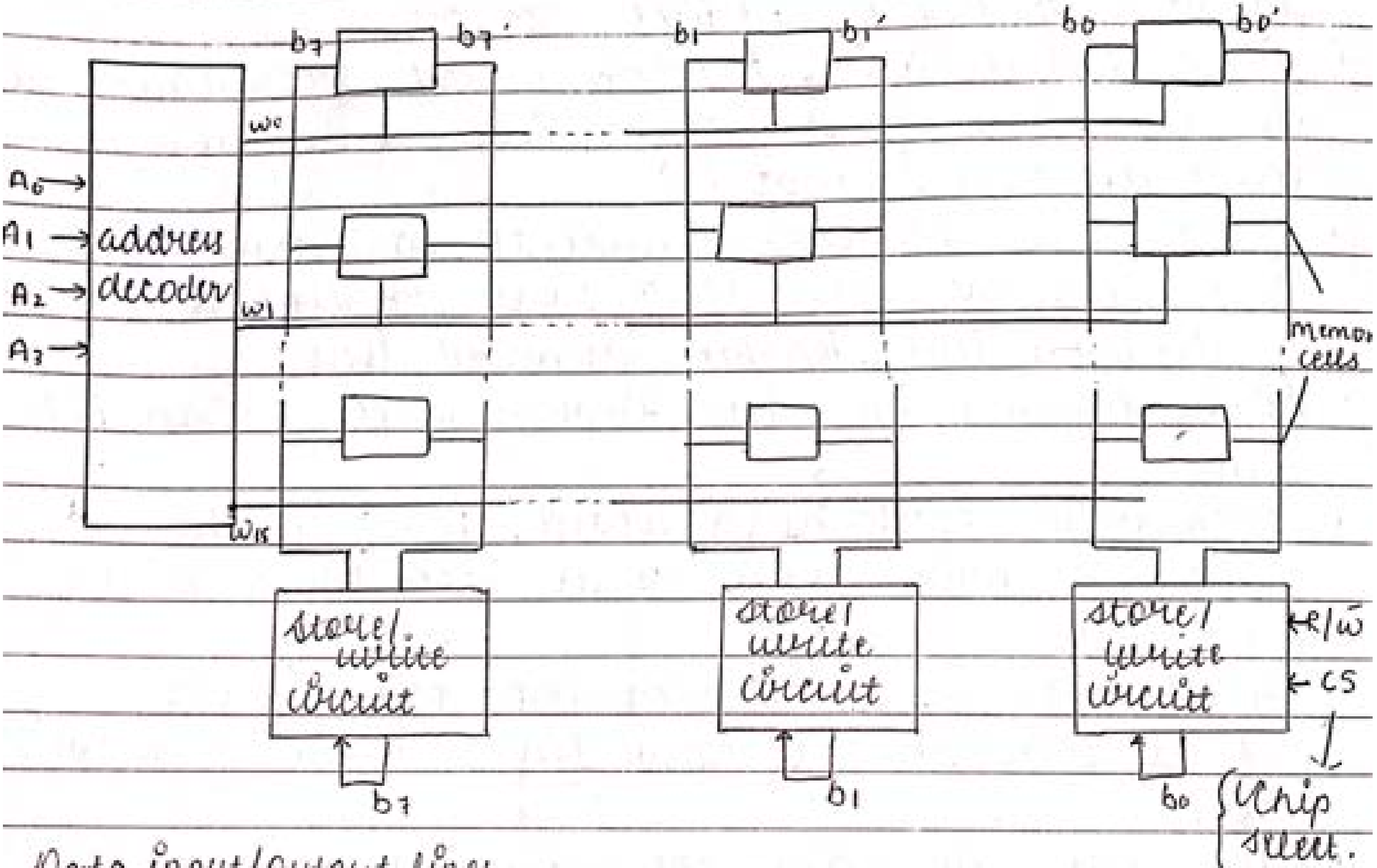
2D Memory organization

Memory cells are usually organized in the form of array, in which each cell is capable of storing one-bit of information.

- In 2D organization, memory cells are organized in the form of a 2D array with rows and columns. (matrix)
- Each row of cells constitutes a memory word and all the cells of a row are connected to a common line, known as word line.
- Each column in the array refers to a bit line.
- Each row contains a word, now, In this memory org., there is a decoder circuit, we use.
- A decoder is a combination logic circuit that contains n input lines and 2^n output lines.
- The cells in each column are connected to sense/write circuit by two bit lines.
- The sense/write circuits are connected to the data input or output lines of the IC chip.
- During a write operation, the sense/write circuit receive input info and store it in

in the cells of the selected word.

- The data input and the data output of each sense/write circuit are connected to a single bi-directional data lines that can be connected to a data bus of the computer.



Data input/output lines

Organization of Memory cells

$W \left< \begin{array}{l} \phi = \text{read}(1) \\ \phi = \text{write}(0) \end{array} \right.$

Size of Ram = 16 × 8 = 128 bit
(supply+ground)

where,

R/W = specifies the required operations

CS ↓ [chip select] - chip select input, select a given chip in the multichip memory system

The memory circuit in the above fig. stores 128 bits, and requires 14 external connections for address, data and control lines (4 for address lines, 8 for data, 1 for R/W and 1 for CS).

- It also needs two lines for power supply and ground connection.

Memory cells of the semiconductor RAM are organized in the form of a 2D array where each cell has the capacity of storing 1 bit of data information.

Above figure is the organization of 16 × 8 memory cells, where there are 16 rows and 8 columns with each row having 8 memory cells arranged in a column structure.

The organization can store 16 words, with each word has a word length of 8 bits.

All the cells arranged approx. the row are connected to one common line, called as word line.

$\left\{ \begin{array}{l} R/\overline{W} = 1 \text{ – for Read} \\ R/\overline{W} = 0 \text{ – for write} \end{array} \right\}$

which activates all the bits of the row when a particular row is accessed at a given time.

For circuit, shown in the given figure, there are 16 ward lines. W

ward lines are contralled by address decoder, which select the recquired ward from the address range between 0 to 15.

To select the recquired ward, there are four bit address input lines which ranges from 0000 to 1111.

A/c to the input value, one output out of 15 lines is activated at a given time.

Read Operation :-

To perform the read operation, a particular ward line is activated by the four bit address and the $R/\overline{W} = 1$.

All the cells in a specified row is activated and data from the row is read from each of the cell.

Write Operation :-

To perform a write operation, a particular ward line is activated by the four bit address line and set $R/\overline{W} = 0$, the

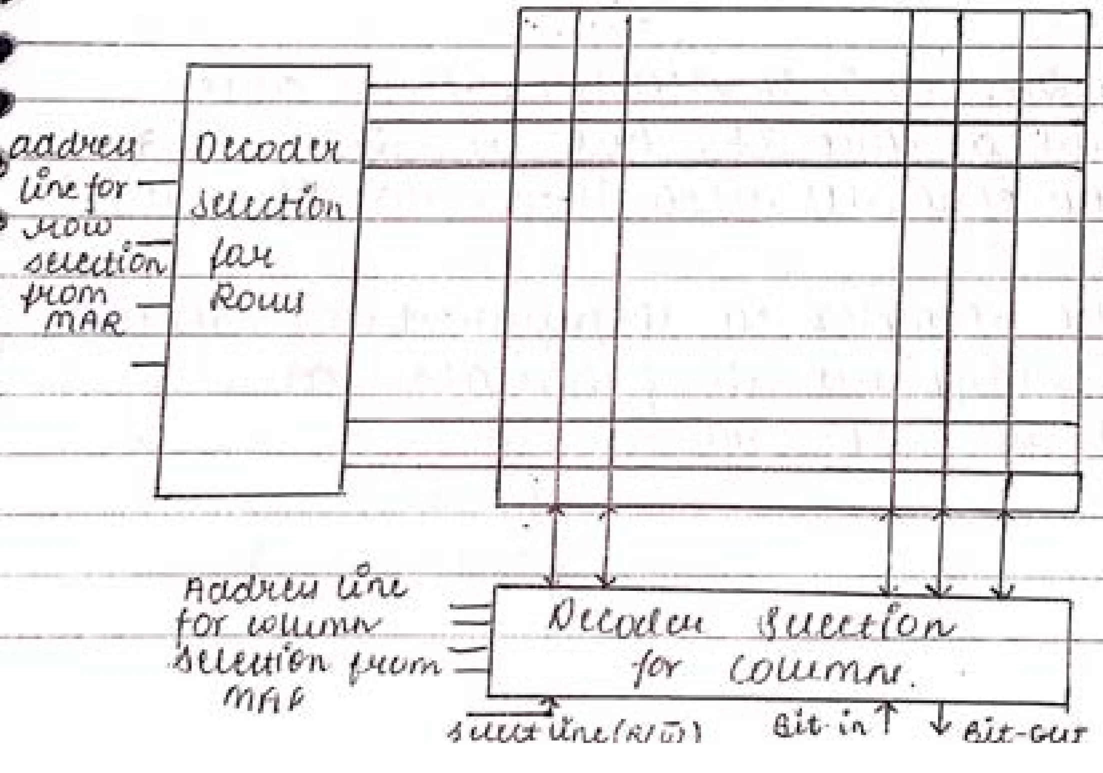
bit in → write
bit out → read

required data value is loaded on the data lines.

$2\frac{1}{2}$D Memory Organization :-
or 2·5D

In 2·5D organization, we have two different decoder, one is column decoder and another is a row decoder.

- column decoder is used to select the column and row decoder is used to select the row.
- The address from the MAR goes as the decoder input.
- Decoder will select the respective cell through the bit outline. Then, the data from that location will be read or write at that memory location


address line for row selection from MAR
Decoder selection for Rows
Address line for column selection from MAR
Decoder selection for column.
select line (R/W̄)
Bit-in ↑
↓ Bit-out

- The content of MAR is divided into two parts:- column address and row address.
- If the select line is in the read mode, then the word which is represented by the MAR that will be transferred to the data lines and get read.
- If the select line is in write mode, then the data line from memory DR (MDR) will go to the respective cell, which is addressed by the MAR.
- With the help of the select line, the data will get selected where, the read and write operation will takes place.

Comparison b/w 2D and 2.5D organization:-

- In 2D organization, hardware is fixed but In 2.5D, hardware changes.
- 2D org. requires more no of gates, while 2.5D requires less no. of gates.

For eg-

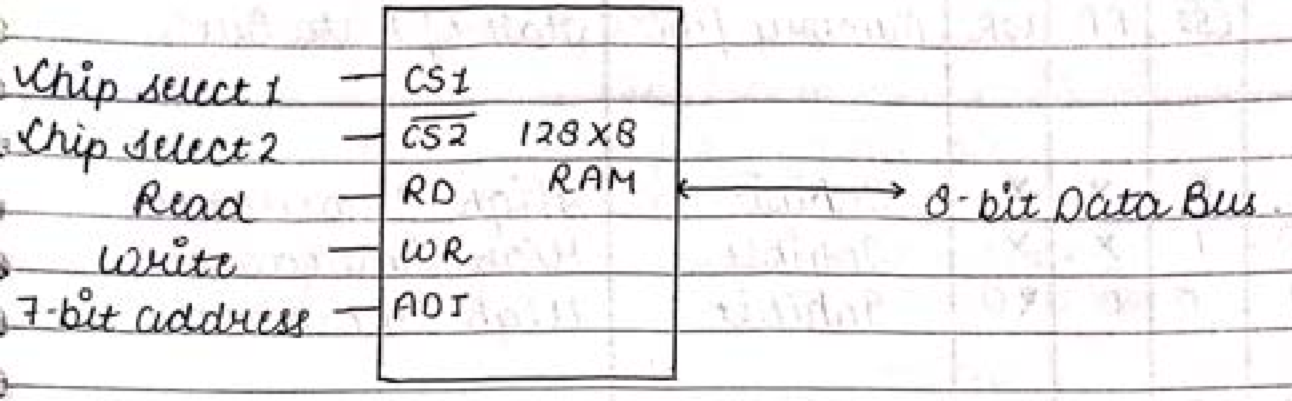
In 2D, for 5×32 decoder, total gates required are 37, but in 2.5D, total no. of gates required are 17. 3×8? {11+6}

- 2D is more complex as compared to 2.5D org.

2D is more difficult to fabricate as compared to 2.5D org.

RAM and ROM Chips:-

The block diagram of RAM chip is as shown in the figure.



A RAM chip is used for communication with the CPU if it has one or more control inputs that select the chip only when needed.

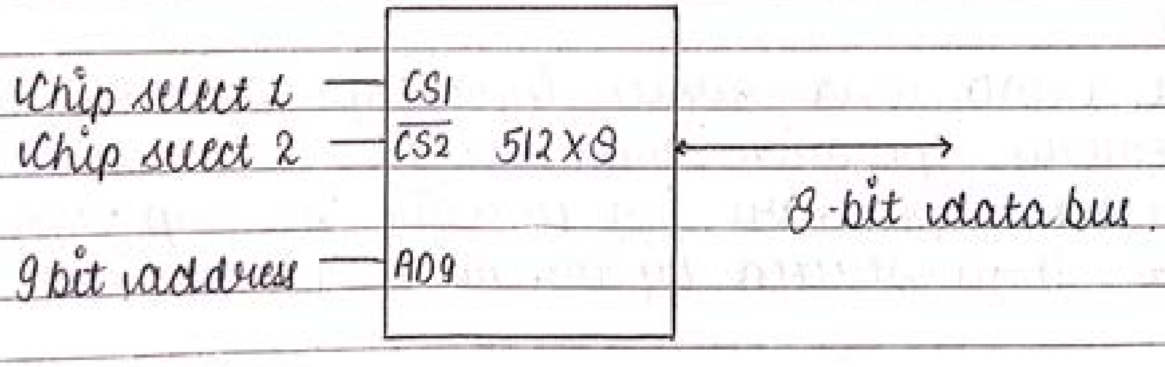
- It has a bidirectional data bus that allows the transfer of data either from the memory to CPU during read operation or from CPU to memory during write op^n.
- If the memory is 128 words of 8 bit (1byte) per word. Therefore, it requires 7-bit address bus and 8-bit bidirectional data bus.
- The read and write input specify the memory operation and the two chip select control inputs are for enabling the chip only when it is selected by the micro-processor.

- The availability of more than one control input to select the chip facilitates the decoding of the address line when multiple chips are used in the micro-computer.
- The function table is specifies the operation of the RAM chip is as shown below:-

CS1	$\overline{CS2}$	RD	WR	Memory fun^n	State of Data Bus
0	0	X	X	Inhibit	High Impedance.
0	1	X	X	Inhibit	High Impedance
1	0	0	0	Inhibit	High Impedance
1	0	0	1	WRITE	Input data to RAM
1	0	1	X	READ	Output data to RAM.
1	1	X	X	Inhibit	High impedance.

ROM

A ROM chip is organised in the similar manner as shown:-



Memory Address Mapping of RAM & ROM :-

Memory address map is the pictorial represent-ation of assigned address space for each chip in the system.

For e.g.

Computer system needs 512 bytes of RAM and 512 bytes of ROM

available size = 128 × 8

RAM = 512 Bytes (req. size)

= 512 × 8

no. of chips required = $\frac{\cancel{512}^{4} \times 8}{\cancel{128} \times 8} = 4$

= 4 chips

ROM = 512 Bytes

= 512 × 8

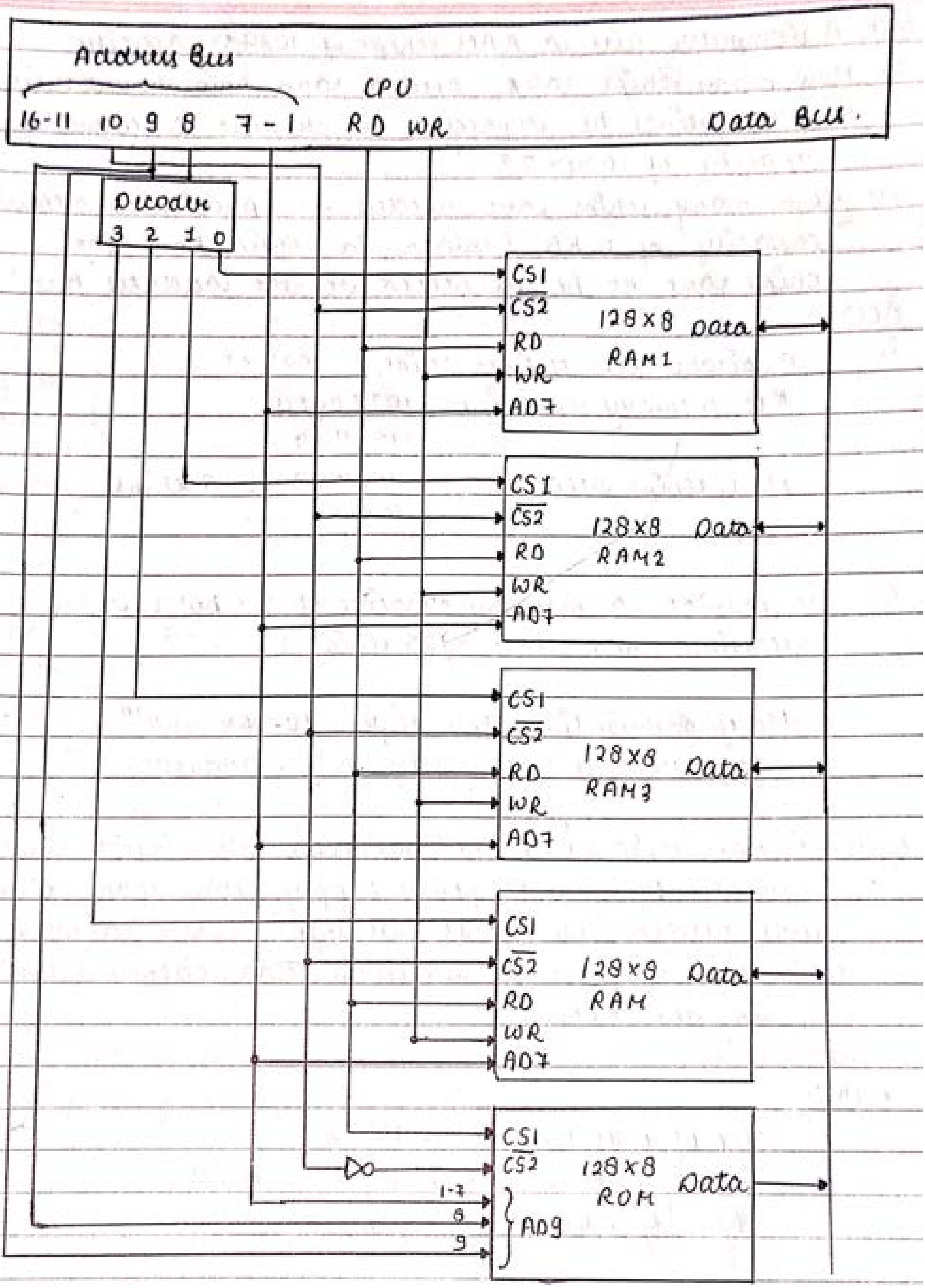
no. of chips required = $\frac{\cancel{512 \times 8}^{1}}{\cancel{512 \times 8}} = 1$

= 1 chips

- The RAM and ROM chips to be used are specified in the figure. The memory address map for this configuration is shown in Table.
- The component column specifies whether a RAM or a ROM chip is used.
- The hexadecimal address column assigns a range of hexadec. equivalent addresses for each chip.
- The address bus line are listed in third column. Although there are 16 lines in address bus.

- The RAM chips have 128 bytes and need seven address lines. The ROM chip has 512 bytes and needs 9 address lines.
- The selection b/w RAM and ROM is acheived through bus line 10. The RAMs are selected when the bit is in this line 0, and the ROM when the bit is 1.

Component	Dec Add.	Hex Add.	Address Bus															
			16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
	000	0000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RAM 1	127	007F	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1
	128	0080	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
RAM 2	255	00FF	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
	256	0100	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
RAM 3	383	017F	0	0	0	0	0	0	0	1	0	1	1	1	1	1	1	1
	384	0180	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0
RAM4	511	01FF	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
	512	0200	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
ROM	1023	03FF	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1


Address Bus
CPU
16-11
10
9
8
7 - 1
RD
WR
Data Bus.
Decoder
3
2
1
0
CS1
$\overline{CS2}$
RD
WR
AD7
128x8
RAM1
Data
CS1
$\overline{CS2}$
RD
WR
AD7
128x8
RAM2
Data
CS1
$\overline{CS2}$
RD
WR
AD7
128x8
RAM3
Data
CS1
$\overline{CS2}$
RD
WR
AD7
128x8
RAM
Data
CS1
$\overline{CS2}$
1-7
8
9
AD9
128x8
ROM
Data

Que. A computer uses a RAM chips of 1024 * 1 capacity.

i) How many chips are needed and how should their address lines be connected to provide a memory capacity of 1024 * 8?

ii) How many chips are needed to provide a memory capacity of 16 KB? Explain in words how the chips are to be connected to the address bus?

Solⁿ:-

i) Available size of RAM chips = 1024 × 1

Req. memory capacity = 1024 bytes.

= 1024 × 8

No. of chips required = $\frac{1024 \times 8}{1024 \times 1}$ = 8 chips

16KB

= 16 × K × B

= $\frac{16 \times 1024 \times 8}{~~1024~~}$

= 16 × 8

= 128 bytes

ii) To provide a memory capacity of 16K bytes, chips required are 16×8 = 128 chips.

16 KB = 16 × 2^{10} × B

= $2^{\boxed{14}}$ × B

no. of address lines

No. of address lines for 16 K = 14 · 16K = 2^{14}

So, 14 lines to specify chip address.

Que. A ROM chip ~~has~~ of 1024 × 8 has four select inputs and operates from a 5V power supply. How many pins are needed for the IC package? Draw a block diagram and label all input and output terminals in the ROM.

Solⁿ:-

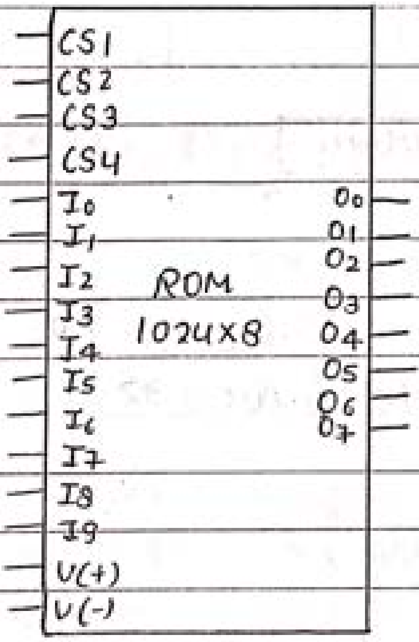
Size of ROM chip = 1024 × 8

No. of inputs = 10 pin [2^{10} = 1024]

No. of output = 8 pin

No. of chip select = 4 pin
Power = 2 pin

Total, 24 pins are required.



Ques:- A computer uses a memory unit with 256K words of 32 bits each. A binary Inst. code is stored in one word of memory. The inst. has four parts: an indirect bit, an operation code, a register code part to specify one of 64 register and an address part.

i) How many bits are there in operation code?

ii) Draw the Inst. word format and indicate the no of bits in each part.

iii) How many bits are there in the data and address inputs of the memory?

Sol^n:

a) Address: $2^8 \times 2^{10} = 2^{18} = 18$ bits
Register : 64 registers $= 2^6 = 6$ bits

OP code : (Total Bit - IndirectBit - Address Bit - Register Bit)
$= (32-1-18-6)$
$= 7$ bits

b)

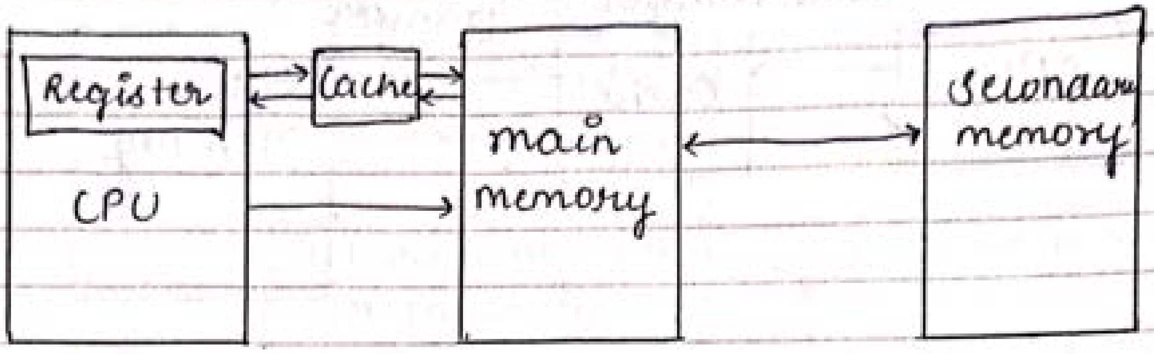
I	OP	Register	Address

31 30 23 17 0

c) No. of bits in address inputs = 18
No. of bits in data input = 32

Cache Memory:

cache memory is a small amount and very high speed memory.
It is used to speed up and synchronized with high speed CPU and it is used to reduce the access time of data from the main memory.
It is a volatile memory.
It behaves as a buffer b/w RAM & CPU.
It holds frequently request data so that they are immediately available to the CPU when needed.
cache memory is costly as compared to main memory but more economical than CPU register.



Operation of Cache Memory :-

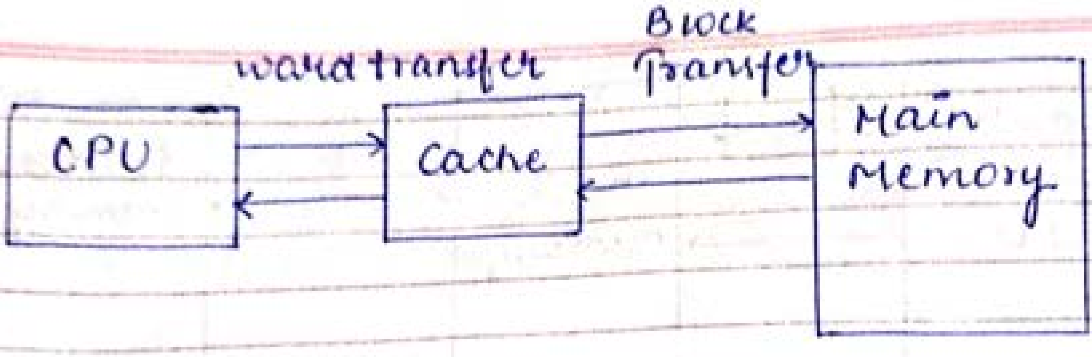
2M

Locality of Reference - for the analysis of large no. of program, a no. of inst's are executed repeatedly.

- This may be in the form of simple loops or few processors. It
- It is observed that many instructions in each of the few localised area of the prog. are repeatedly by the remain of the prog. less access relatively. This phenomenon is referred to as locality of reference.

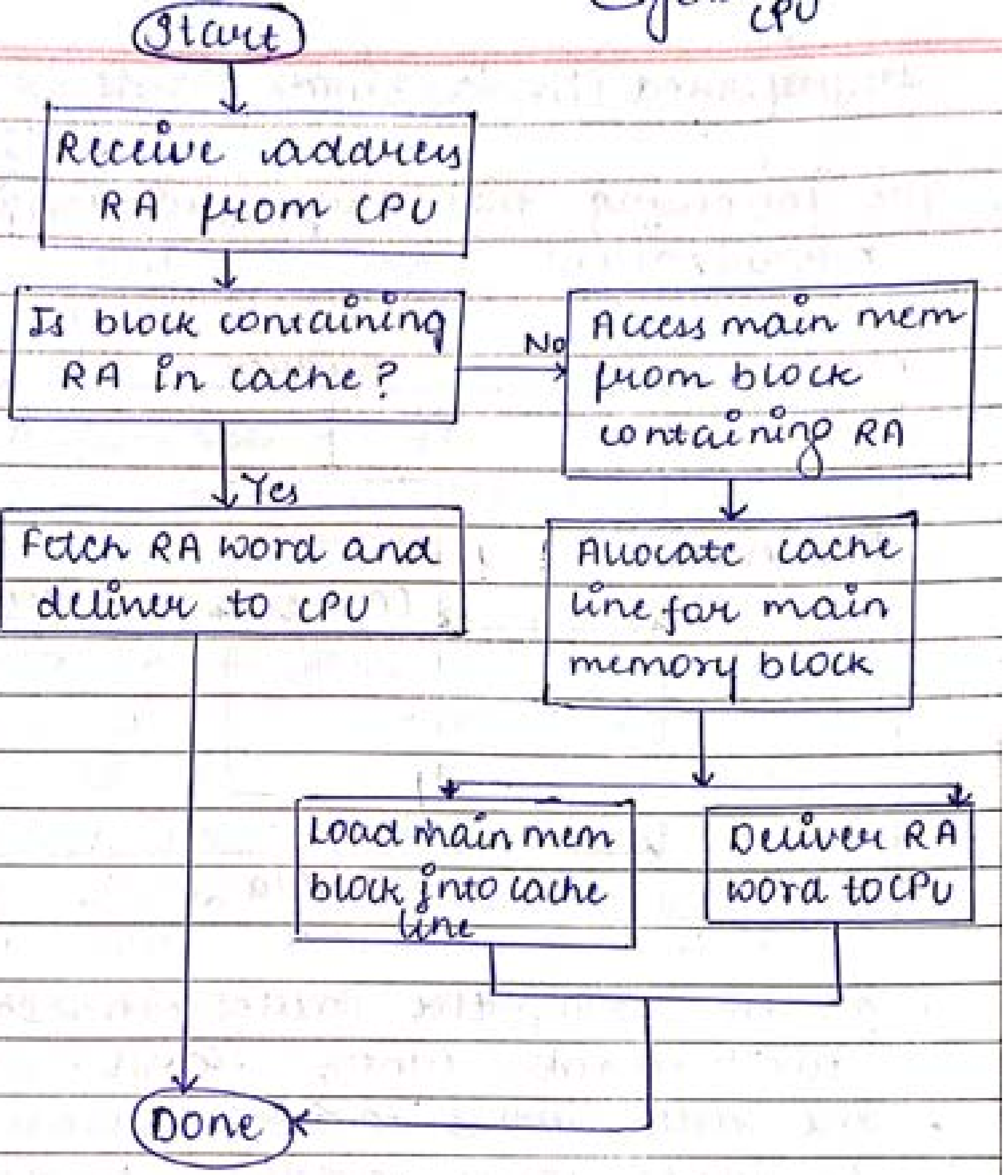
- Only the CPU can access the cache memory. This memory can be reserve part of the main memory / secondary device outside the CPU

- The cache holds data and program that the CPU used frequently. Thus, ensure that the info. is instantaneously aviable for the CPU and when the CPU needs this info.



- When the CPU needs to access memory, the cash is examined.
- If the ward is found, in the cache memory, it is read from the cache.
- If the ward addressed by the CPU is not found in the cache, the main memory is accessed to read a ward.
- A block of wards containing the one just accessed is then transferred from main memory to cache memory.
- The block size may vary from one ward to about 16 ward adjacent to the one just accessed.
- Therefore, some data are transferred to cache. So, that, future references to memory, find the required ward in the fast cache memory.

RA - Read Address
generated by CPU



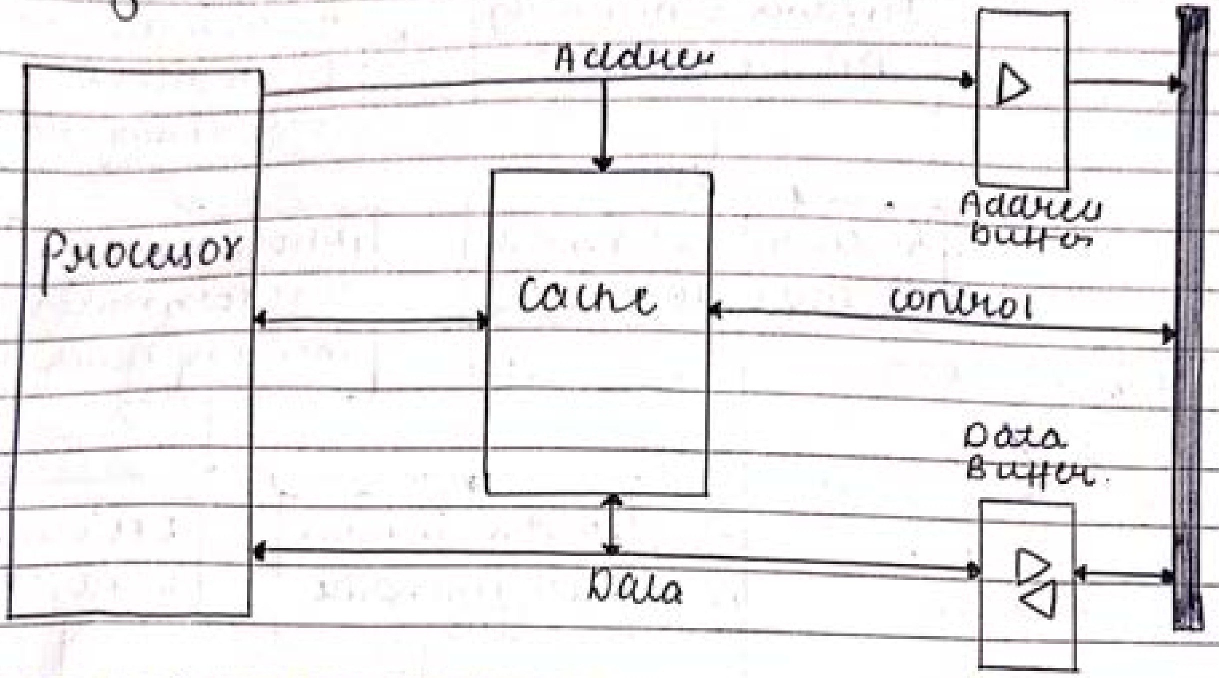
Start
Receive address RA from CPU
Is block containing RA in cache?
No
Access main mem from block containing RA
Yes
Fetch RA word and deliver to CPU
Allocate cache line for main memory block
Load main mem block into cache line
Deliver RA word to CPU
Done

The processar generates the read address (RA) of a word to be read. If the word is contained in the cache, it is delivered to the processar. If a cache miss occurs, two things must be accomplished:

- The block containing the word must be loaded in to the cache.
- The word must be delivered to the processar.

When a block is brought into a cache, in the event of a miss, the block is generally not transferred in a single event.

The following fig shows a typicaly cache organization:



- In this arg., the cache connects to the processor via data, control and address lines.
- The data and address lines also attach to data and address buffers, which attach to a system bus from which main memory is reached.
- When a cache hit occurs, the data and address buffers are disabled and communication is only b/w processor and cache, with no system bus traffic.
- When a cache miss occurs, the desired address is loaded onto system bus and the data are returned through data buffer to both the cache and the processor.

Working principle of cache Memory:-

The basic principle that cache Memory technology is based upon is known as locality of reference.

Locality of Reference:- It is a principle which states that many instrs. in the localized area of prog. are executed repeatedly while remaining are executed infrequently.

It is supported by two other aspects:-
1. Temporal locality of reference.
2. Spatial locality of reference.

1) Temporal locality of reference:-

- Temporal locality states that the same data objects are likely to be reused multiple times by the CPU during the execution of a program.
- Once a data object has been written into the cache on first miss, several subsequent hits on that object can be expected. In this, least recently used algom is used.
- Whenever there is a fault occurs within a word will not only load word in main memory but complete page fault will be [illegible] loaded because spatial

locality of reference rule says that If you are referring any word, next word will be referred in its registers that's why we load complete page table so the complete block will be loaded.

Spatial locality of reference:-

- It states that if a data object is referenced once, then there is high probability that its neighbour data objects will also be referred in near future.
- This says that there is a chance that element will be present in the proximity to the reference point. and next time if again searched then more close proximity to the point of reference. Implementing this type of transfer is called as block transfer. ....

Cache Performance:-

- The performance of cache is measured in terms of **Hit Ratio**.

- When CPU refers to memory and find the data or inst. within the cache memory, it is known as cache hits. The very first time, when CPU tries to find data in cache then there is sure miss. This miss is called.

Compulsary ~~for~~ Miss.

- If desired data or inst. is not found in cache memory. and CPU refers to main memory to find the data or instruction. it is known as cache Miss.
- When the processor needs to read or write a location in main memory, it first checks for a corresponding entry in the cache, a cache hit has occured and data is read from the cache.
- If processor does not find memory location in the cache, a cache miss has occured. 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 quantity called **Hit Ratio.**

$$\text{Hit Ratio}(H) = \frac{\text{hit}}{(\text{hit} + \text{miss})}$$

$$\text{Hit Ratio}(H) = \frac{\text{no. of hits}}{\text{Total accesses}}$$

$$\text{miss Ratio} = \frac{\text{miss}}{(\text{hit}+\text{miss})} = \frac{\text{no. of miss}}{\text{Total accesses}}$$

$$\text{Miss Ratio} = 1 - \text{Hit Ratio}(H)$$

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

Important Definition :-

Cache Hit - When data is found in cache.

Hit Ratio - It is the fraction of accesses which are a hit.

Cache Miss :- When data is not found in cache.

Miss Ratio - It is fraction of accesses which are a Miss.

Hit Time - Time Taken to accesses the cache

Miss Penalty :- Time Taken to move the data from main memory to cache and then CPU, when data is not found in cache.

Block = $\frac{512}{4}$ ← Block size.

Mapping Function :-

The transformation of data from main memory to cache memory is referred to as a mapping process.

- In general, cache memory mapping mean how data is copied (mapped) from main mem. to cache memory.
- When considering the org. of cache memory, there are three types of mapping procedure —
  1) Direct Mapping
  2) Associative Mapping
  3) Set - Associative Mapping

Direct Mapping - In this mapping, main memory blocks are copied to a fixed block of cache (line number), but one at a time.

- consider

i) main memory size = 512 × 8

ii) Cache memory size = 64 × 8

iii) Block size = 4 words.

$$\text{No. of main memory Block} = \frac{\text{Total main memory word}}{\text{Block size}}$$

$$= \frac{512}{4} = 128 \text{ blocks}$$

Cache Memory Block or lines

No.of cache memory blocks = $\frac{\text{Total cache memory words}}{\text{Block size}}$

$= \frac{64}{4} =$ 16 Blocks or lines

- Cache mapping is expressed as —

Cache mem. block no. = [main mem. Block no.] modulo [No. of cache block]

OR

Cache line number = [main mem Block no.] modulo [no. of line in cache]

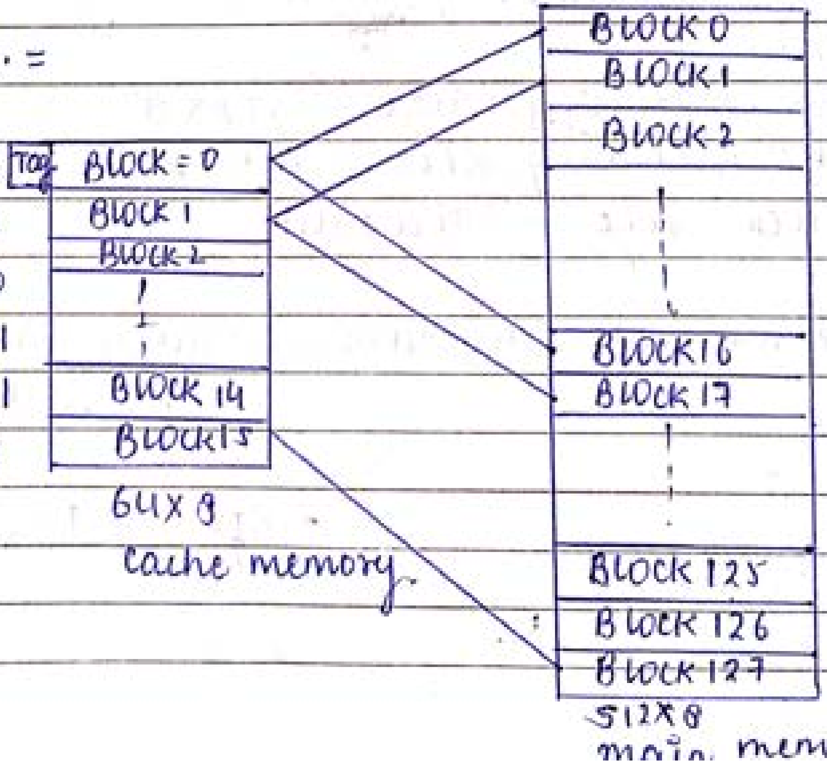
Mathematically -

$i = j$ modulo m

where, i = cache line no.
j = main mem. block no.
m = no. of line in cache.

for example -
cache line no. =

⇒ 0 modulo 16 = 0
⇒ 16 modulo 16 = 0
⇒ 32 modulo 16 = 0
⇒ 1 modulo 16 = 1
⇒ 17 modulo 16 = 1
⇒ 33 modulo 16 = 1

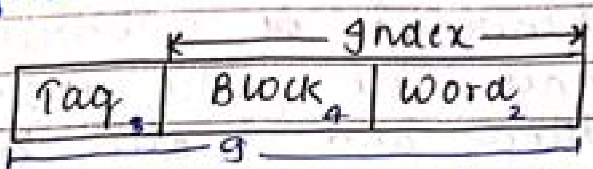


No. of main memory block in 1 block (line of cache) =

$$= \frac{\text{no of main memory block}}{\text{no. of cache memory block}}$$

$$= \frac{128}{16} = 8 \text{ Block}$$

Main Memory Address :-



- Word bits is to be determined by block size

Block size = 4 words

$= 2^2$ words $\rightarrow$ word bits

Word Bits = 2

- Block bits are determined by no. of block (line in cache)

No. of line OR Block in cache = 16 Blocks

$= 2^4$ blocks $\rightarrow$ Block bit

Block Bits = 4.

- Tag Bits are determined by no. of blocks of main memory can mapped into one block of cach So, no. of main memory blocks in one bloc (line) of cache) = 8

$= 2^3$ — Tag Bits

Size of cache memory word or cache memory word length = Tag Bits + Word size

$= 3 + 8$

$= 11$ bit

Ques. A digital computer has a memory unit of 64K × 16 and a cache memory of 1K words. The cache uses direct mapping with a blocksize of 4 words.

a) How many bits are there in the tag, index, block and word field of address format?

b) How many bits are there in each word of cache and how are they divided into fun's.? Include a valid bit.

c) How many blocks can the cache accommodate

Solution:-

Size of main memory = 64K × 16

OR
Cache = 1K
no. of words in cache = 1K

Blocksize = 4 words.

$$\text{No. of blocks in M/M} = \frac{\text{No. of words in M/M}}{\text{Blocksize}}$$

$$= \frac{64K}{4}$$

$$= 16K = 2^4 . 2^K = 2^4 . 2^{10} = 2^{14}$$

= 16,384 Blocks or Lines

No.of blocks in cache mem = $\frac{\text{No. of words in cache}}{\text{Blocksize}}$

$= \frac{1K}{4} = \cancel{210}\ \frac{2^{10}}{2^2} = 2^8$

$= 256$ Blocks or lines

No.of M/M block in 1 block blo of cache mem = $\frac{\text{No. of main mem. blocks}}{\text{No. of cache mem. blocks}}$

$= \cancel{10}\ \frac{16K}{256} = \frac{16\times 2^{10}}{2^8} = 2\,16\times 4$

$= 64$ Blocks. $= 2^6$

Tag Bits :- $\boxed{\text{Tag Bits} = 6}$

Address format -

index

6	8	2
Tag	Block	Word

16 bit

Block bits :- determined by blocks in cache mem.

$= 256$ Blocks

$= 2^8$

$\boxed{\text{Block bits} = 8}$

Word bits :- determined by block size —

$= 4$ words $= 2^2$

$\boxed{\text{Word bits} = 2}$

Index = Word Bit + Block Bit

= 8+2 = 10 Bit

Bits in each word of cache:

1	6	16
Valid Bit	Tag Bit	Word Bit

23 bit

Associatine Mapping :-

In associaine mapping, main memory blocks are copied into any block of cache memory.

- consider

Main memory size = 512 × 8

Cache memory size = 64 × 8

Blocksize = 4 words

Soln. No. of main memory block = $\frac{\text{Total M/M words}}{\text{Blocksize}}$

$$= \frac{512}{4} = 128$$

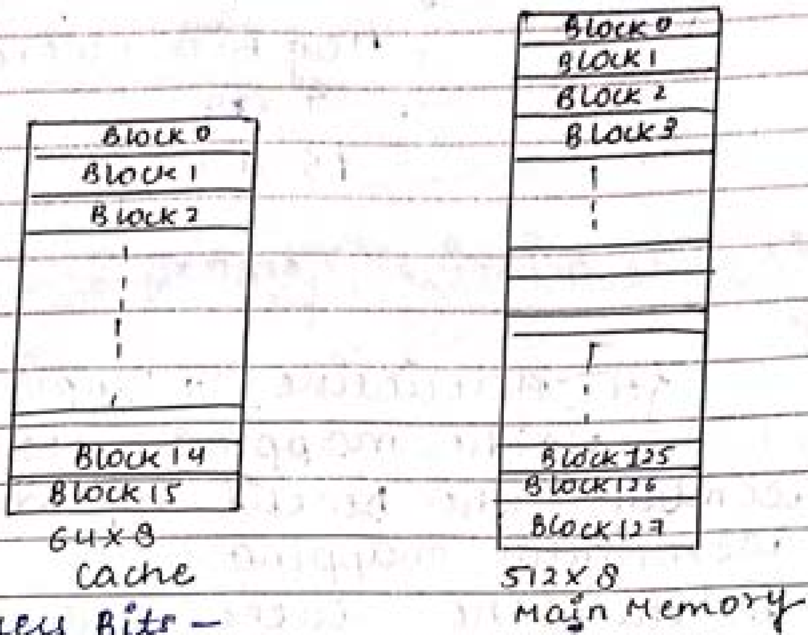
= 128 blocks

no. of cache mem. blocks = $\frac{\text{Total C/M words}}{\text{Block size}}$

$$= \frac{64}{4}\ 16$$

= 16 Blocks.

- no. of main mem. blocks in 1 block of cache :- 128 blocks.



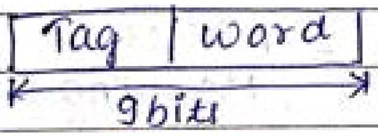
Main Memory Address Bits –

main memory size = 512 words = 2^9.

then,

address bits = 9 bits

Address Format –



- Word bits – determined by block side

= 4 words = 2^2 words

$\boxed{\text{Word Bits} = 2}$

- Tags bit – determined by no. of M/M blocks which can be mapped into cache mem. block.

= 128

= 2^7

$\boxed{\text{Tag Bits} = 7}$

- Cache Memory Word Length :-

  = Tag Bits + Word Bits ← (M/M)

  = 7 + 8

  = 15 Bits

Set-Associative Mapping :-

Set-Associative Mapping is a hybrid cache mapping technique that combines the benefit of direct mapping and associative mapping

- In this, the cache is divided into several sets and each memory block maps to exactly one set.
- Within a set, a memory block can be placed in any cache line. So,

$$\text{Set Associative Mapping} = \text{Direct Mapping} + \text{Associative Mapping}$$

- In this, cache memory is divided into set.

- Set = Group of blocks
- Block = Group of words

- In this, mapping is expressed as—

$$\text{Cache Memory Set No.} = \left[\text{Main memory Block no.}\right] \text{ modulo } \left[\text{no. of sets in cache memory}\right]$$

$$\text{No. of blocks in main memory} = \frac{\text{Total main mem words}}{\text{Block size}}$$

$$\text{No. of blocks in cache memory} = \frac{\text{Total cache mem word}}{\text{Block size}}$$

$$\text{no. of sets in cache memory} = \frac{\text{No. of cache mem. blocks}}{\text{Set size}}$$

consider, set size = 2 (Two-way set Ass. Mapping)

~~no. of blocks in m/m =~~

- Main memory size = 512 × 8
- Cache memory size = 64 × 8
- Blocksize = 4 words.

$$\text{no. of blocks in M/M} = \frac{512}{4} = 128 \text{ blocks}$$

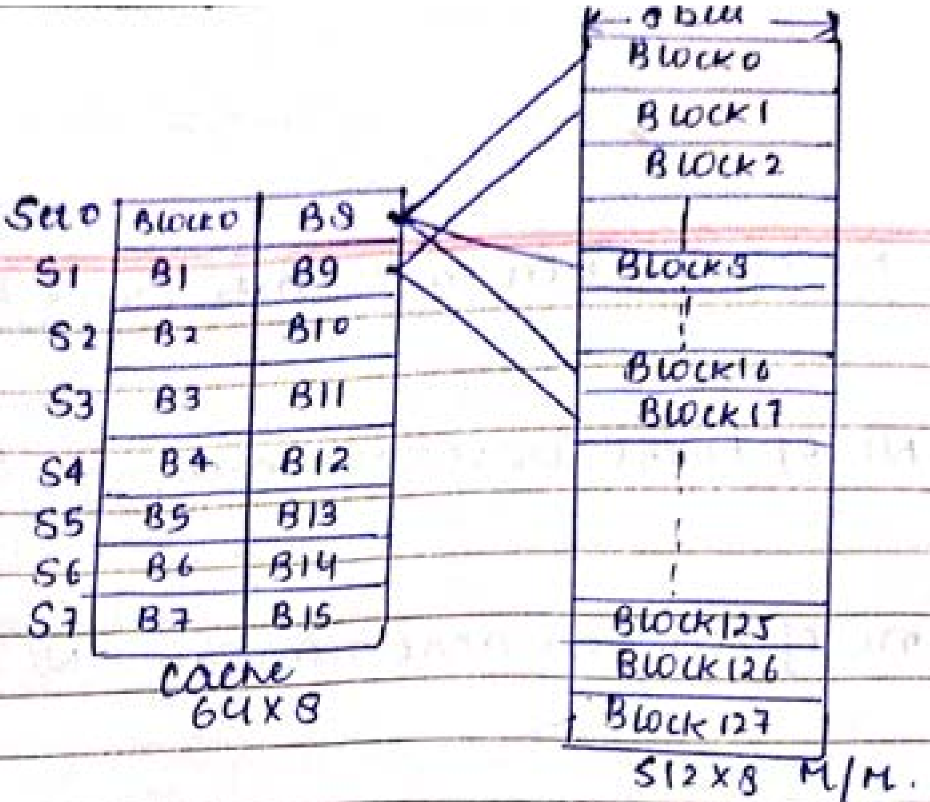
$$\text{no. of blocks in C/M} = \frac{64}{4} = 16 \text{ blocks.}$$

$$\text{no. of sets in cache mem.} = \frac{\text{No. of cache mem. blocks}}{\text{Set size}}$$

$$= \frac{16}{2} = 8 \text{ sets.}$$

mapping:-

0 modulo 8 = 0
8 modulo 8 = 0
16 modulo 8 = 0
1 modulo 8 = 1
9 modulo 8 = 1
17 modulo = 1



Address format:-

Tag 3	Set 4	word 2

9

~~Set~~ word Bits — (by blocksize) = 4 words
= 2^2 word
$\boxed{\text{word Bits} = 2}$

Tag Bits — 2 8 bits
= 2^3
$\boxed{\text{Tag} = 3 \text{ bits}}$

Set Bits = 9 + (3+2)
$\boxed{= 4 \text{ bits}}$

(M/M)
↑
Cache memory word length = 2(Tag + word Bit)
= 2 (4+8)
= 24 bits

Ques A 2-way set associative cache memory uses blocks of 4 words. The cache can accomadate a total of 2048 words from memory. The main memory size is 128 K x 32.

a) Formulate all nec. info. to construct the cache memory.

b) Determine the size of cache word and also size of cache memory.

No. of blocks in M/M = $\dfrac{\text{Total } \overset{M.}{\text{mem.}} \text{ words}}{\text{Blocksize}}$

$= \dfrac{\overset{32}{\cancel{128}} \text{ K}}{\cancel{4}} = 32 \text{ K}$

$= (2)^5 \times (2)^{10}$

$= (2)^{15}$

$= 32768$ blocks

No. of blocks in cache mem. = $\dfrac{2048}{4}$ = 512 blocks

No. of sets in cache mem. = $\dfrac{\text{Total mem. Blocks}}{\text{Setsize}} = \dfrac{512}{2}$

= 256 sets.

Address format —

Tag 8	Set 7	Word 2

17

word Bits = 4 words

= 2^2 words

$\boxed{\text{word Bits} = 2 \text{bit}}$

$\frac{32768}{256} = 128 = (2)^7$

Tag Bits = ~~512 b'256~~

= ~~(2)~~^{8}

$\boxed{\text{Tag Bits} = 7}$

{ Set Bits = 17 - (7 + 2)

= 5

$\boxed{\text{Set Bits} = 5\text{bit}}$ } ⓧ.

Cache Memory Word Length = 2(Tag + word)

= 2(7 + 32)

= 2(40)

= 80 bits

{Set Bits – determined by no. of sets in cache mem}

Set Bits = 256 = $(2)^8$

$\boxed{\text{Set Bits} = 8\text{ bits}}$

b)

Virtual Memory:-

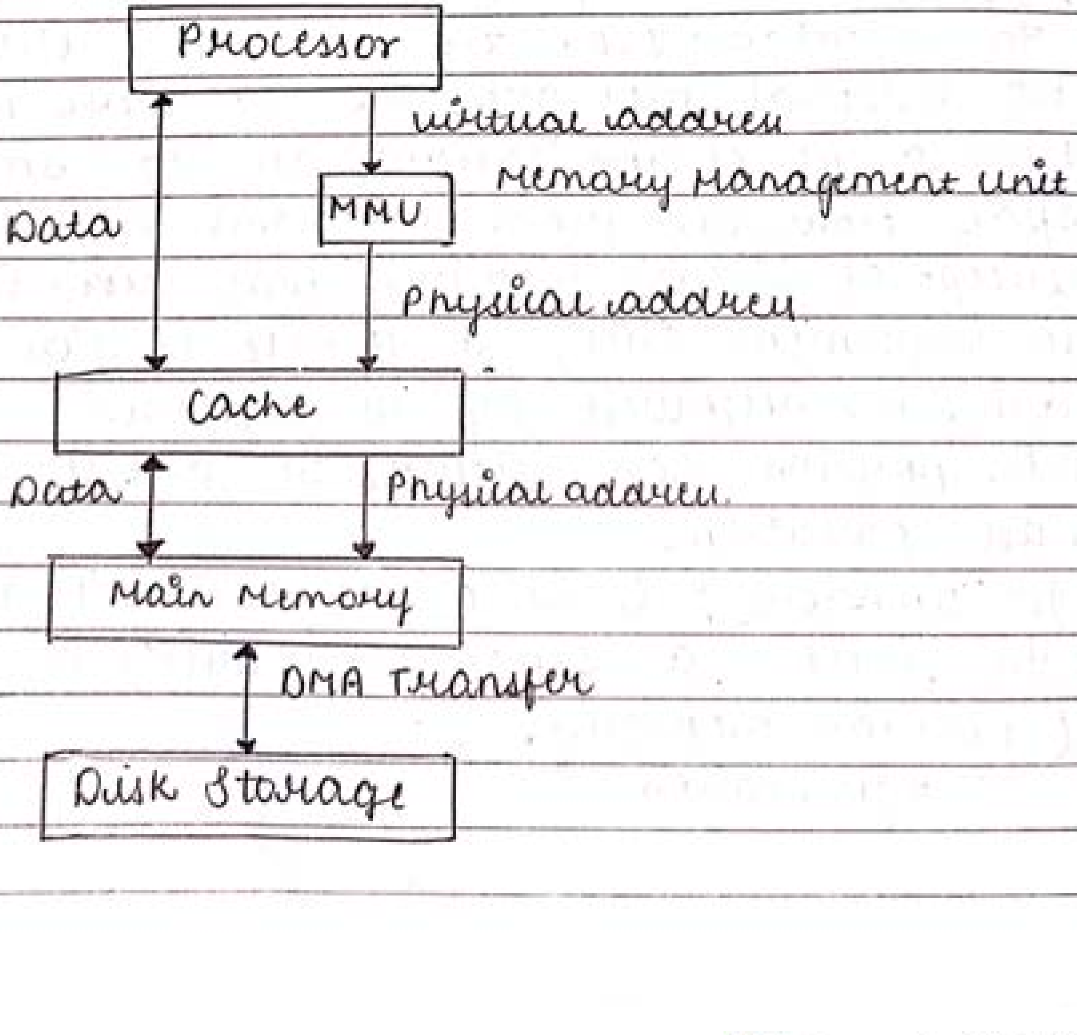
- Virtual memory is a concept that give an illusion to the user that he has sufficient mem. to execute any application /program of any size.
- Virtual memory allows a no of application having total size more than the main memory size to run at the same time.
- Virtual memory is a simulated memory that is written to a file on the hard drive. This file is called <u>Page File</u> or <u>Swap File</u>.
- It is used by operating system to simulate physical RAM by writing hard disk space.
- In windows 1.0, 2.0 version, there was no virtual memory. So, we were not able to run a no. of applications due to small RAM space. However, from windows 3.0 onwards, concept of virtual memory was ~~used~~ introduced.
- To implement this, a ~~inverse~~ portion of hard drive is reserved by the system.
- This portion can either be a file or a separate partition.
- In window, it is a file called Page File System.
- In Linux, a separate partition is used for virtual memory.

segmentation.

- When the system needs more memory, it maps some of its memory address to the hard disk drive. This extra mem. does not actually exist in RAM, it is the storage space on the disk drive.
- The more RAM, the computer has faster your program run.
- If lack of RAM, is slowing our computer, then we can increase ~~[illegible]~~ virtual memory to compensate.

However, adding more RAM gives high speed rather than virtual R memory because it takes more time to swap data from hard disk than the RAM.

- Following figure shows an organization that implements virtual memory -



- A special hardware unit, called the MMU, keeps the track of which part of the virtual address space are in the physical memory.
- When the desired data or inst. are in the main memory, the MMU translate the virtual address into the corresponding physical address, then the requested memory accessed proceeds in the usual manner.
- If the data are not in the main memory, the memory m.u. (MMU) causes the OS, to transfer the data from disk to the main memory. Such transfers are performed using DMA method.

For example,

In case of windows 10.0,

1) initial virtual memory = 1.5 X RAM Size
= 1.5 X 4GB
= 6GB

2) maximum virtual memory = 3 X Initial size
= 3 X 6GB
= 18GB.

Conversion from virtual (logical) address to physical address:

OR

Address Mapping :-

Address mapping is a technique which converts virtual (logical) add. to physical address.

Virtual Address :-

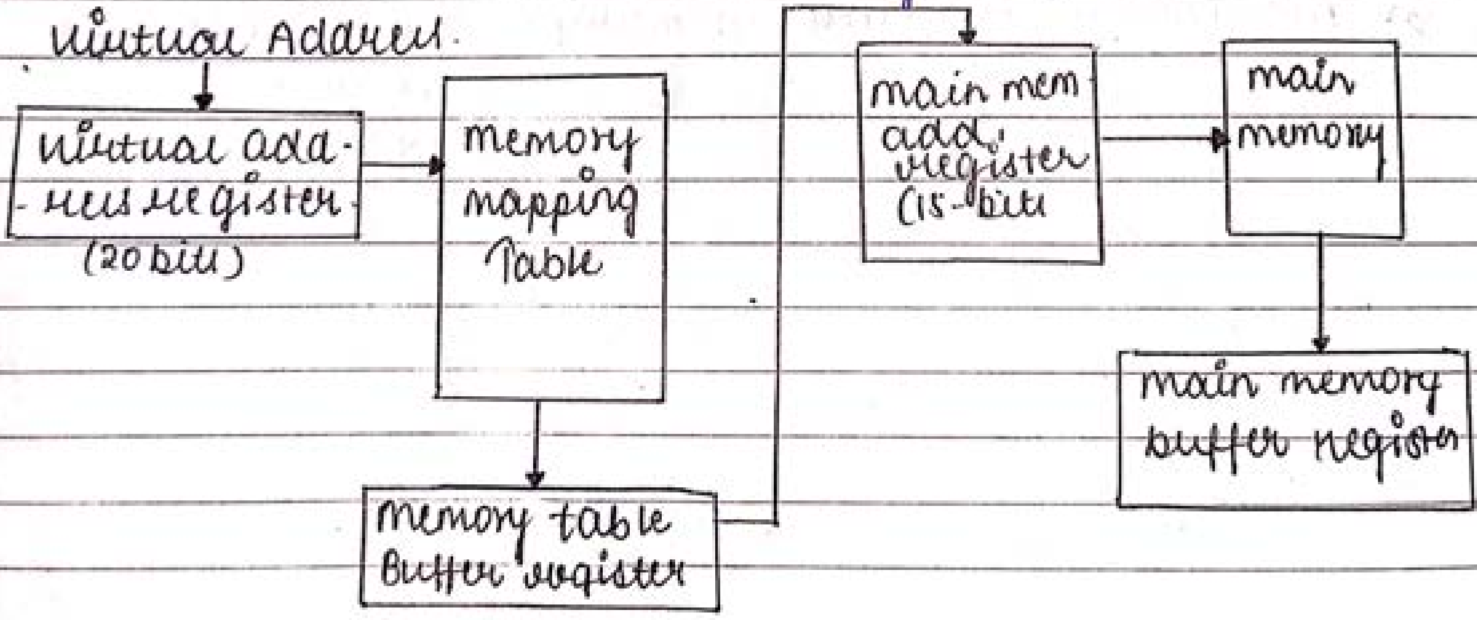
- Each address in virtual mem. is called virtual address.

Address space :- Set of all virtual addresses. is called address space.

Physical Address — [Memory Address]

Each address in main memory is called memory address.

Memory space :- The set of all memory addresses is called memory space.


Virtual Address.
Virtual address register
(20 bits)
memory mapping Table
memory table Buffer register
main mem add. register (15-bits
main memory
main memory buffer register

Swapping :- Swapping is a mechanism in which a process temparanily moved out from main memory to secondary memory. and an other process moved in from sec. storage to main mem.

After sometime, the first process again brought back to the main memory.

There are two methods for the virtual memory implementation :-

a) using Paging Method
b) using Segmentation Method.

Page Replacement Algorithm :-

(in memory organization)

i) FIFO (First In First Out)
ii) LRU (Least Recently used)

i) FIFO :- In the FIFO method, page come first in main memory will be moved out first.

Assume, address space = 8K words
↳ (virtual memory - sec. mem)

memory space = 4K words
↳ (Main Mem)

Page size = Block size = 1K words.

Page ↓ (sec.) ↓ (virtual)

Block ↓ (main) (cache → lines)

no. of pages = $\frac{\text{address space}}{\text{page size}}$

no. of pages = $\frac{8K}{1K}$

no. of blocks = $\frac{\text{memory space}}{\text{Block size}}$

no. of blocks = $\frac{4K}{1K}$

Example :-

4 2 0 1 2 6 1 4 0 1 0 2 3 5 7

4	4	4	4	6	6	6	6	5	5
	2	2	2	2	4	4	4	4	7
		0	0	0	0	2	2	2	2
			1	1	1	1	3	3	3

no. of Page faults = 10

ii) LRU Page Replacement Algorithm :-

In the LRU, the replace the pages that has been used least recently.

eg-

4 2 0 1 $\overset{\times}{2}$ 6 $\overset{\times}{1}$. 4 0 $\overset{\times}{1}$ $\overset{\times}{0}$ 2 3 5 7

4	4	4	4	6	6	6	2	2	2	2
	2	2	2	2	2	0	0	0	0	7
		0	0	0	4	4	4	3	3	3
		1	1	1	1	1	1	1	5	5