

COMPUTER ORGANIZATION

(1)

ASSIGNMENT - 3

D. Neha

245319733015

CSE-A

PART-1

- 1) Distinguish between write back and write through methods of writing data to cache memory.

A).

Write Back

Write Through

→ Data is updated only in cache memory.	→ Data is updated both in cache and main memory.
→ Simple method to implement.	→ Complex method to implement.
→ When a cache block is not required then selected cache block is written back to main memory to UPDATE the main memory.	→ When cache block is not required, there is no need to write back to main memory.
→ Commonly used method.	→ unpopular method.
→ unreliable method	→ reliable method.
→ No data redundancy	→ Data redundancy.
→ No wastage of time as data update occurs only in cache memory	→ Wastage of time as each data update both in cache and main memory.

- 2) List the advantages of virtual memory.

A) Advantages of virtual memory:

- The degree of multiprogramming will be increased.
- User can run large application with less real RAM.
- There is no need to buy more memory RAMs.

- 3) Define hit ratio.

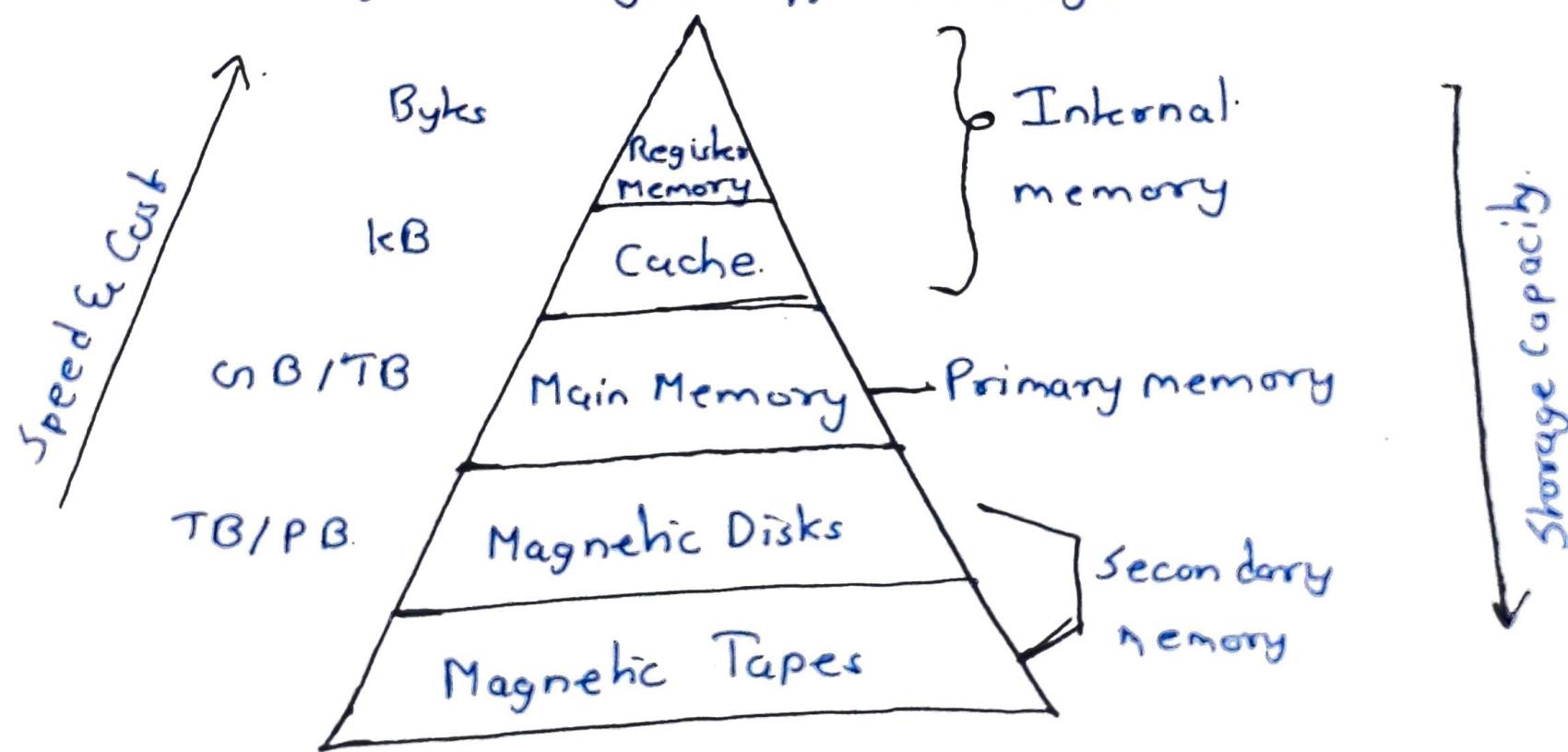
A) The hit ratio is the number of cache hits divided by the total number of memory requests over a given time interval. The value is expressed as a percentage.

$$\text{hit ratio} = \left(\frac{\text{cache hit}}{\text{memory request}} \right) \times 100$$

Q) Illustrate the components in memory hierarchy of the computer system with neat figure.

A) There are 4 major storage levels:

- i) Internal - Processor registers and cache.
- ii) Main - The system RAM and controller cards.
- iii) On-Line Mass storage - secondary storage.
- iv) Off-Line bulk storage - Tertiary and off line storage.



5. Distinguish between static RAM and Dynamic RAM.

Ans)	Static RAM (SRAM)	Dynamic RAM (DRAM)
→ SRAM uses transistors to store a single bit of data.	→ DRAM uses a separate capacitor to store each bit of data.	
→ SRAM does not need periodic refreshment to maintain data.	→ DRAM needs periodic refreshment to maintain the charge in the capacitors for data.	
→ complex structure	→ simple structure.	
→ expensive than DRAM.	→ less expensive than SRAM.	
→ faster process	→ slower process	
→ used in cache memory	→ used in main memory.	

6. A certain memory system has 128 MB main memory and 2 MB of cache. Cache blocks are 32 bytes in size. Show the fields in a memory address if the cache is

- Direct
- 8-set associative mapped.

A) i) Direct:

Given: Size of MM = 128 MB = 2^{27} bytes

Size of Cache = 2 MB = 2^2 bytes

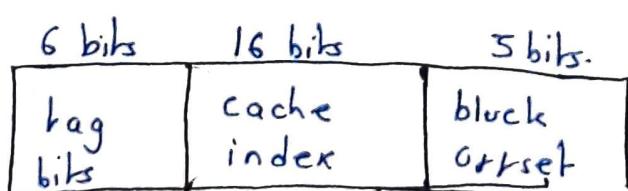
Size of cache block = 32 bytes = 2^5 bytes

$$\text{No. of blocks} = \frac{\text{MM}}{\text{BS}} = \frac{2^{27}}{2^5} = 2^{22} = 4,19,4304.$$

$$\text{No. of lines} = \frac{\text{CS}}{\text{BS}} = \frac{2^{21}}{2^5} = 2^{16} = 65,536$$

Cache index = 16 bits

Tag bits = 6 bits, block offset = 5 bits.



(ii) 8 set associate mapping :-

$$\text{No of sets} = \frac{\text{No of lines}}{K} = \frac{2^{16}}{2^3} = 2^{13} = 8192$$

Tag bits 9 bits.	Set index. 13 bits.	block offset 3 bits.
---------------------	------------------------	-------------------------

Q. A 128 MB main memory has a 64 kB direct mapped cache with 16 bytes per line. How many lines are there in code and show main memory bits are partitioned.

Given: Size of main memory = 128 MB = 2^{27} bytes

$$\text{Cache size} = 64 \text{ kB} = 2^{10} \times 2^6 = 2^{16} \text{ bytes}$$

$$\text{Block size} = 16 \text{ bytes} = 2^4 \text{ bytes}$$

$$\text{No of lines} = \frac{CS}{BS} = \frac{2^{16}}{2^4} = 2^{12}$$

$$\text{No of blocks in MM} = \frac{MS}{BS} = \frac{2^{27}}{2^4} = 2^{23}$$

11 bits	12 bits.	4 bits.
tag bits	Cache memory	block offset.

Part-2.

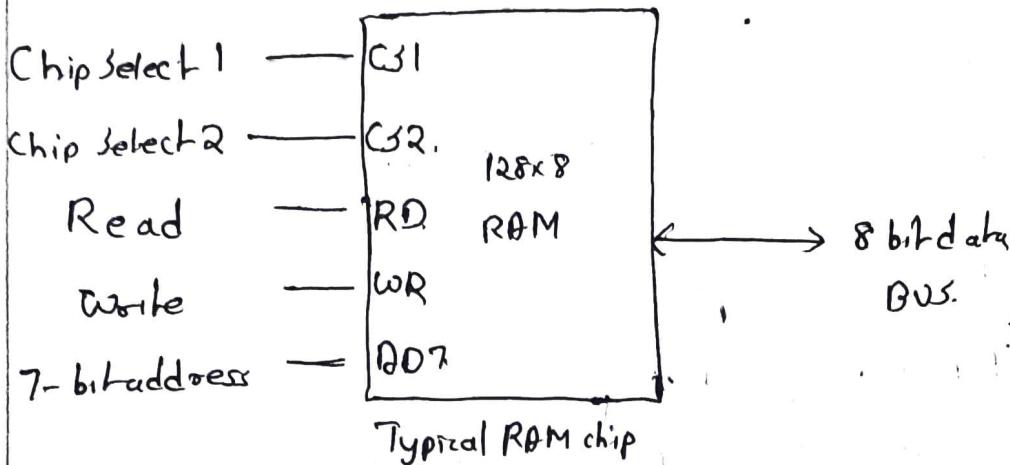
8) Explain about RAM and ROM chips.

A)

→ A RAM chip is better suited for communication with the CPU if it has one or more control units that select the chip only when needed. Another common feature is a bidirectional databus that allows the transfer of data either from memory to CPU during a write operation.

→ A bidirectional bus can be constructed with three state buffers.

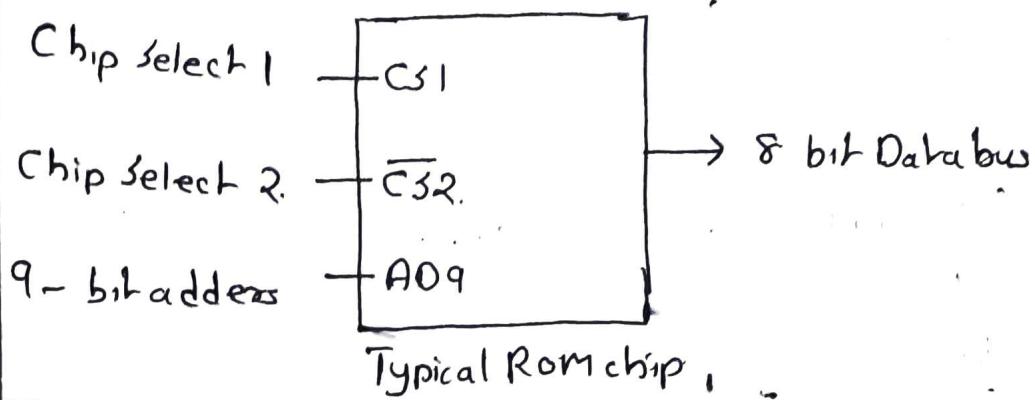
→ A three state buffer output can be placed in one of three possible states, a single equivalent to logic 1, a signal equivalent to logic 0, or a high impedance state. The logic 1 and 0 are normal digital signals. The high impedance state behaves like an open circuit, which means the output does not carry a signal and has no



- The function table listed specifies the operation of the RAM chip. The unit is in operation only when $CS_1 = 1$ and $\overline{CS}_2 = 0$
- The bar on top of the second select variable indicates that this input is enabled when it is equal to 0.
- If the chip select inputs are not enabled, or if they are enabled but the read or write inputs are not enabled, the memory is inhibited and its databus is a high impedance state.

CS_1	CS_2	RD	WR	Memory function	Status of databus
0	0	X	X	Inhibit	High Impedance
0	1	X	X	Inhibit	High Impedance
1	0	⊗	⊗	Inhibit	High Impedance
1	0	0	1	Write	Input data to RAM
1	0	1	X	Read	Output data from RAM
1	1	X	X	Inhibit	High Impedance

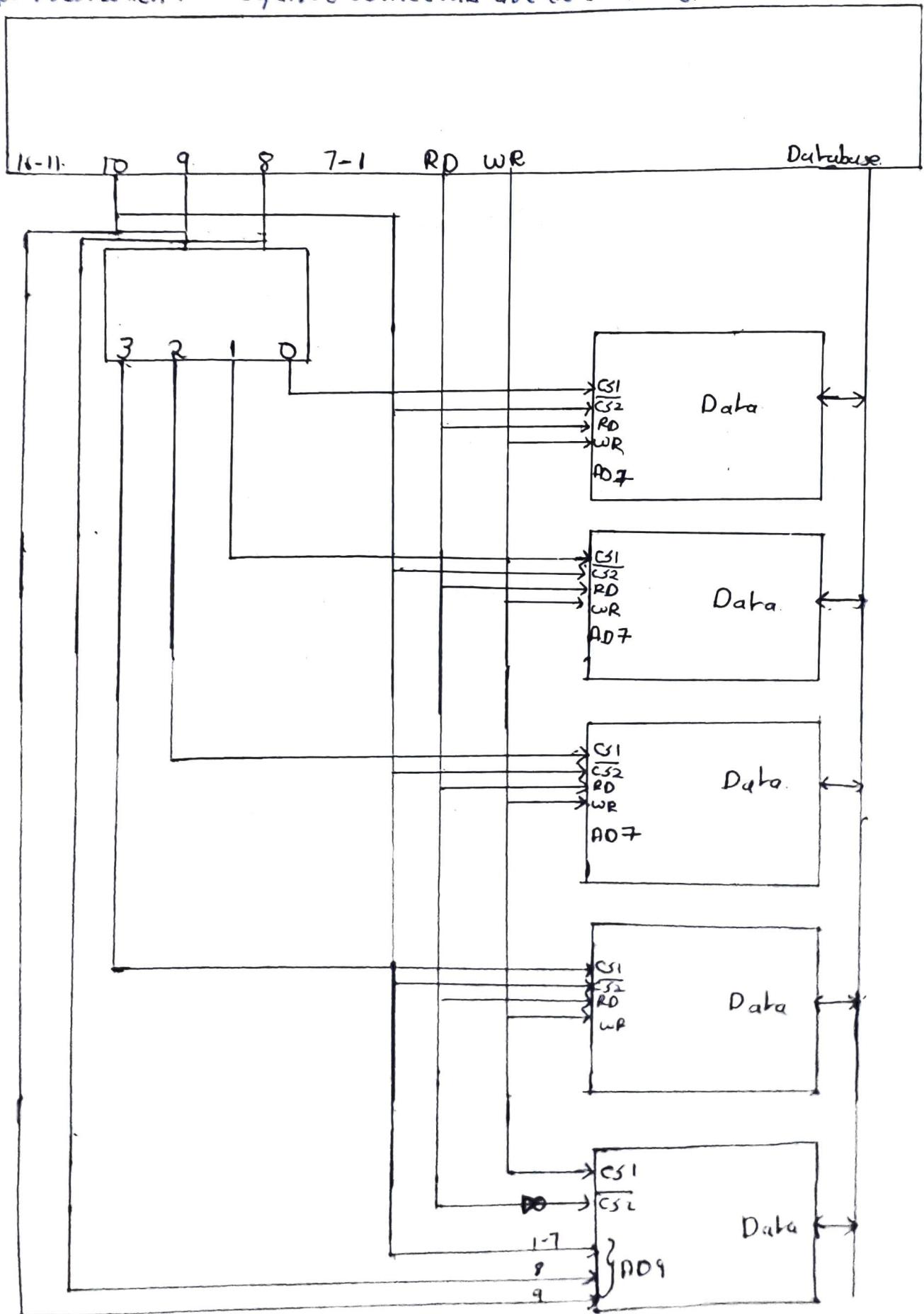
- * A ROM chip is organized externally in a similar manner. However, since a ROM can only read, the databus can only be in an output mode. The block diagrams shown below for the same-size chip, it is possible to have more bits of ROM than of RAM, because the internal binary cells in ROM occupy less space than in RAM.



The nine address lines in the ROM chip specify any one of the 512 bytes stored in it. The two chip select input must be $CS1 = 1$ and $\overline{CS2} = 0$ for the unit to operate.

- Q) With neat figure, explain memory connection to CPU
- A) RAM and ROM chips are connected to a CPU through data & address buses
- The lines 1-7 in the address bus select the byte within the RAM chip and the lines 1-9 select the byte within the ROM chip without going through decoder.
 - The RAM select is determined from lines 8 and 9. This is done through 2x4 decoder whose outputs go to the CS1 input of each RAM.
 - The selection between RAM and ROM is done through bus line 10 - 0-RAM, 1-ROM.
 - The databus of the RAM is bidirectional whereas the data of the ROM has only one direction.
 - When address lines 8 and 9 are equal to 00, the first RAM chip is selected when 01, second RAM is selected and so on.
 - The RD and WR outputs from the microprocessor are applied to the inputs of each RAM chip.
 - The other chip select (CB1) of ROM is selected to RD to enable only Read operation.
 - The Databus of the ROM has only an output capability whereas the Databus connected to the RAMs can transfer information in both directions.
 - The example shown gives us indication of the interconnection complexity that can exist between memory chips and the CPU.

- The more chips that are connected, the more external decoders are required for selection among the chips.
- The designer must establish a memory map that assigns addresses to the various chips from which the required connections are determined.



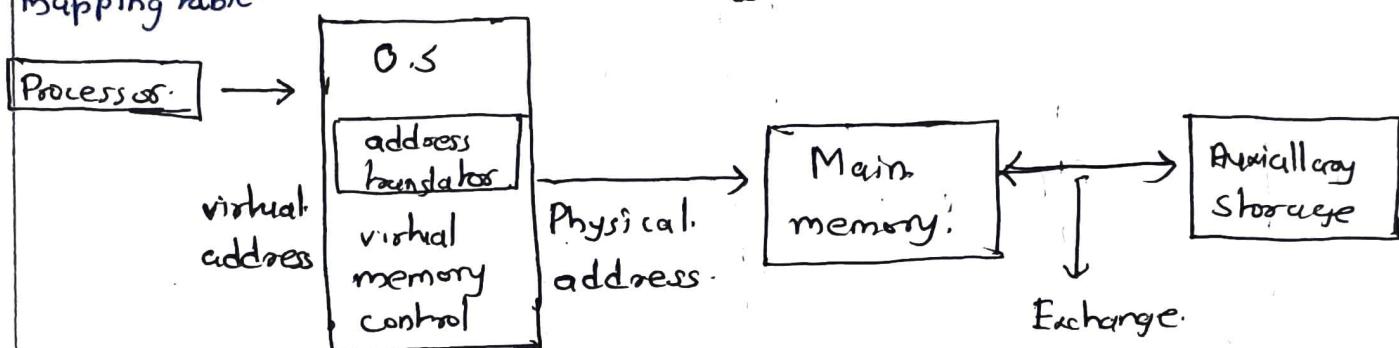
Q) Explain in detail about associative memory.

- A) → A memory unit accessed by content is called an associative memory or content addressable memory (CAM).
- Generally data is stored in tabular form in memory.
- So to get data from the table 2 ways can be used.
- 1) Choosing a sequence of address, reading the contents of the address, comparing to item with the contents until the match is found.
- 2) Search the item ~~with the contents~~ until the match is found using part of the data itself.
- The time required to find an item stored in memory can be identified for access by the content of the data itself rather than by an address.
- This type of memory is accessed simultaneously and in parallel on the basis of data content rather than by specific address or location.
- When a word is to be written in an associative memory, no address is given. The memory is capable of finding an empty unused location to store the word.
- When a word is to be read from an associative memory the content of the word or part of the word is specified. The memory locates all words which match the specified content and makes them for reading.
- An associative memory is more expensive than a random access memory (RAM) because each must have storage capacity as well as logic circuits for matching its contents.
- Thus associative memories are used in applications where the search time is very critical and must be very short.

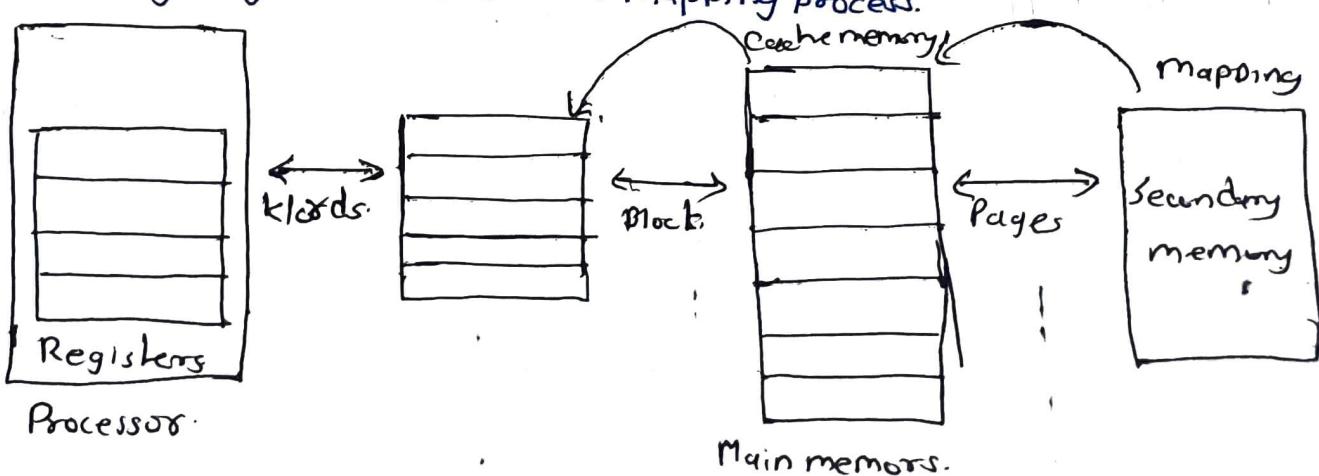
Q) Write short notes on virtual memory.

- A) → Virtual memory provides a computer programmer with an addressing space many times larger than physically available addressing space of the main memory.
- Data and instructions are placed in this space with the use of virtual address.
- Each address that is referred by the CPU goes through an address mapping from the virtual address to a physical address in main memory.

- A virtual memory system provides a mechanism for translating programs generated address into correct main memory location. This can be done dynamically while programme being executed in the CPU.
- The translation or mapping is handled automatically by the hardware means of a mapping table.



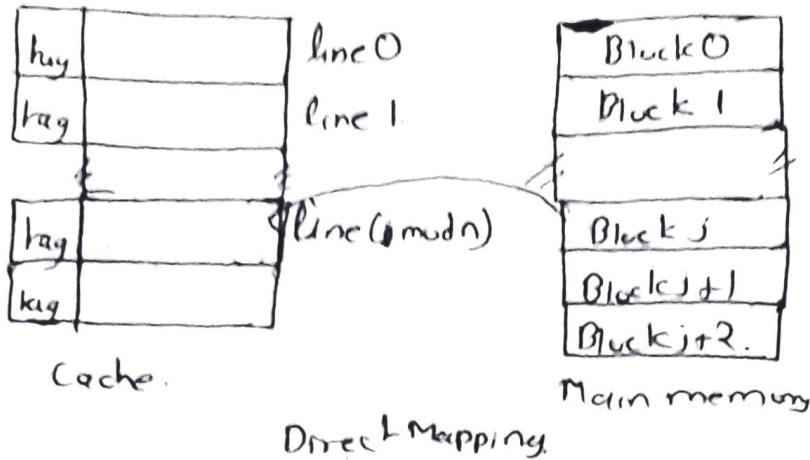
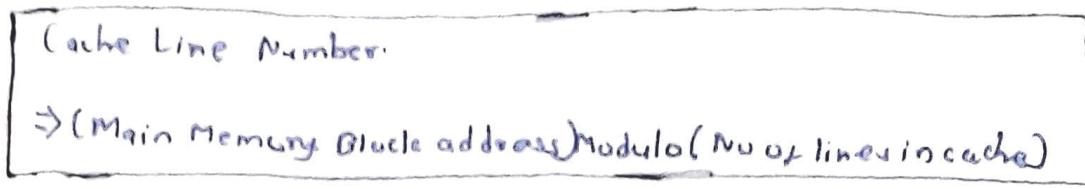
- 1) Discuss the mapping techniques of cache memory in detail.
- 2) Cache Mapping: defines how a block from the main memory is mapped to the cache memory in case of cache miss.
- It is a technique by which the contents of the main memory are brought into the cache memory.
- Following diagram illustrates the mapping process.



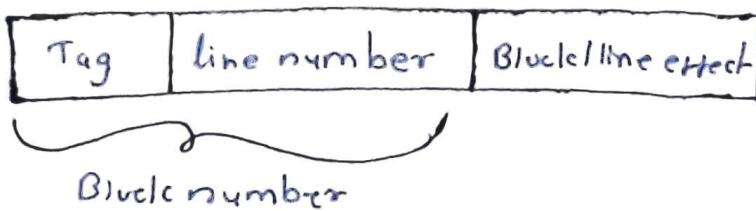
Cache Mapping Techniques:

- 1) Direct Mapping
 - 2) Fully Associative Mapping
 - 3) k-way set Associative mapping
- Direct Mapping: In direct mapping, a particular block of main memory can map only to a particular line of the cache.

→ The line number of cache to which a particular block map is given by.



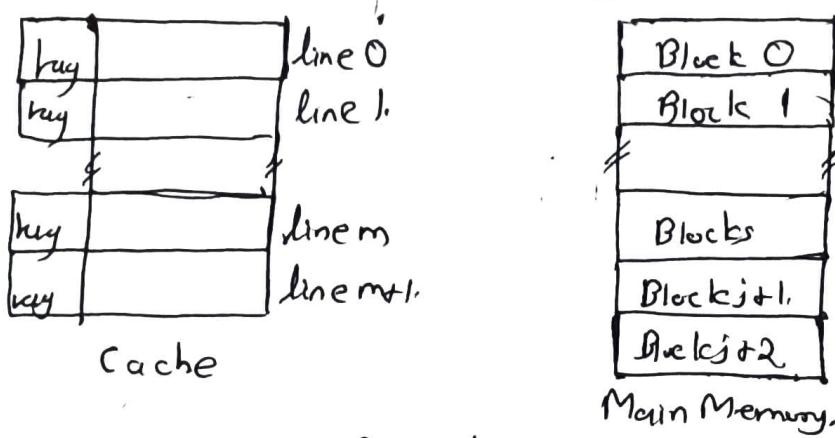
→ In direct mapping, the physical address is divided as.



2) Fully Associative Mapping: In fully associative mapping

→ A block of main memory can map to any lines of the cache that are freely available at that moment.

→ This makes fully associative mapping more flexible than direct mapping.



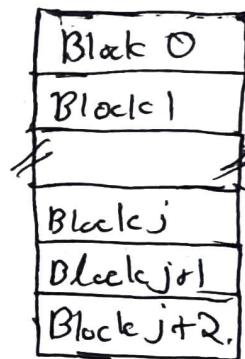
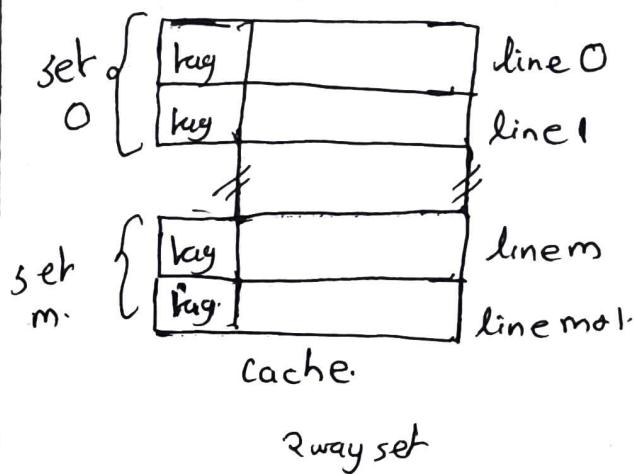
→ The physical address is divided as.



3) k -way set Associative Mapping: In this mapping

- Cache lines are grouped into sets where each set contains k -no. of lines
- A particular block of main memory can map to only one particular set of the cache.
- However, within that set, the memory block can map any cache line that is freely available.
- The set of the cache to which a particular block of the main memory can map is given by

$$\begin{array}{l} \text{Cache set number} \\ = (\text{Main memory block address}) \bmod (\text{No. of sets in cache}) \end{array}$$



Associative mapping

- Here $k=2$, suggest that each set contains two cache lines.
- Since cache contains 6 lines, so no. of sets in the cache = $6/2$ = 3 sets
- If all the cache lines are occupied, then one of the existing blocks will have to be replaced.

The physical address is divided as:

Tag	Set number	Block/line offset
-----	------------	-------------------

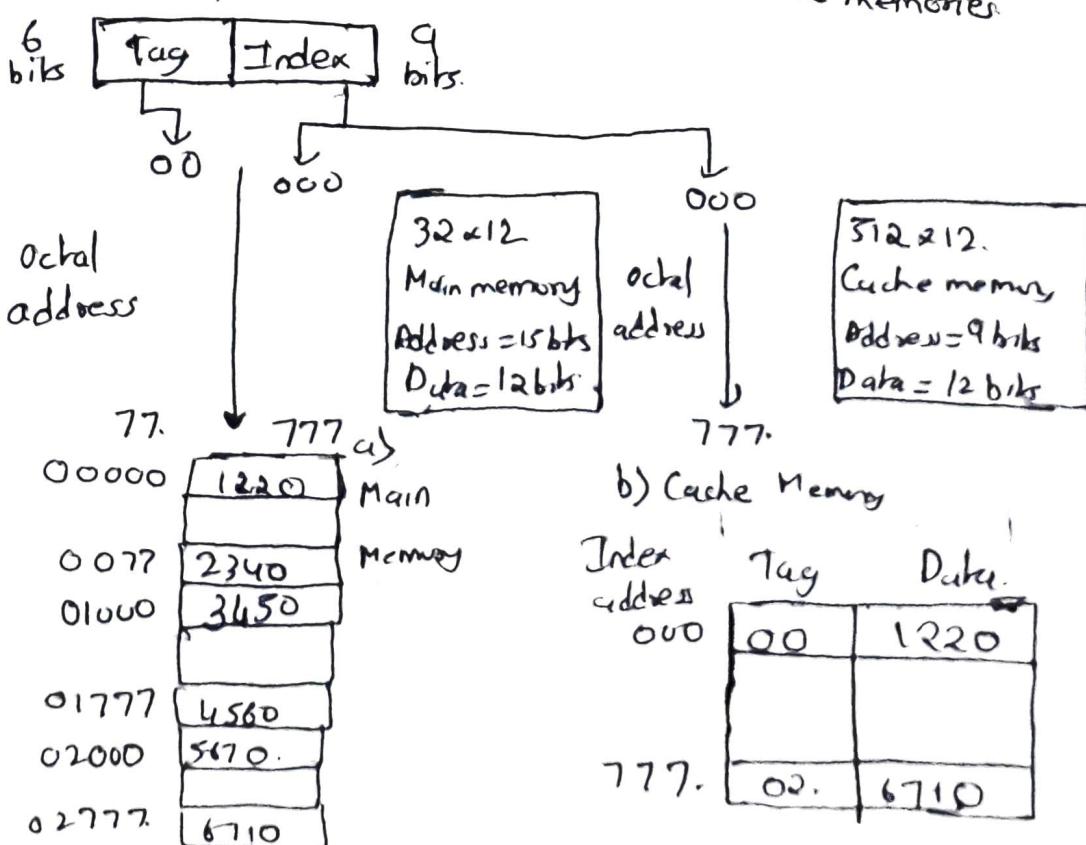
- If $k=1$, then k -way associative mapping becomes direct mapping i.e., 1-way set associative mapping = direct mapping.

b) Explain the direct mapping process of a cache memory of size 512×12 bits with memory of 32×12 bits. Give the relevant details.

- Direct mapping memories are added logic associated with each cell.
- The CPU address of 15 bits are divided into index field and the remaining 5 bits from the tag field. The no. of bits in the index field is equal to no of address bits required to access cache memory.
- In general, there are 2^k words in cache and 2^n words in main memory.
- When a new word is first brought into the cache, the tag bits are stored alongside the data bits.
- Suppose that the CPU now wants to access the word at address 02000. The index address is 000, so it is used to access the cache. The two tags are then compared. The cache tag is 00, but the address tag is 02, which does not provide a match. Therefore, the main memory is accessed and the data word 5670 is transferred to the CPU.

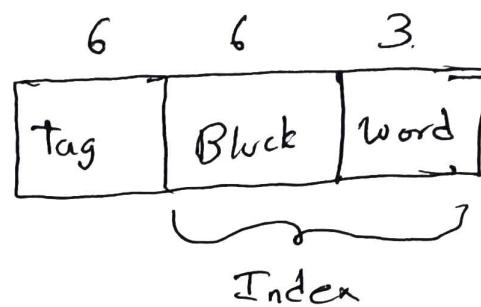
The cache word at index address 000 is then replaced with a tag of 02 and data of 5670.

Addressing relationships between main and cache memories.



→ In a 512-word cache. There are 64 blocks of 8 words each, since $64 \times 8 = 512$.
 The tag field is stored within the cache. is common to all eight words of the same block. Everytime a miss occurs an entire block of 8 words must be transferred from main memory to cache memory.

Index	Tag	Data
Block 0 000	01	3450
007	01	6578
Block 1 010		
017		
⋮	⋮	⋮
⋮	⋮	⋮
Block 2 100		
107		
⋮	⋮	⋮
Block 3 110		
117		
⋮	⋮	⋮
Block 63 111		
117	02	6710



Direct mapping cache with block size of 8 words

→ —