

Unit - 5

RAM



SRAM

DRAM

Usage

Cache memory

main memory

Speed

very fast

fast

Cost

Costly

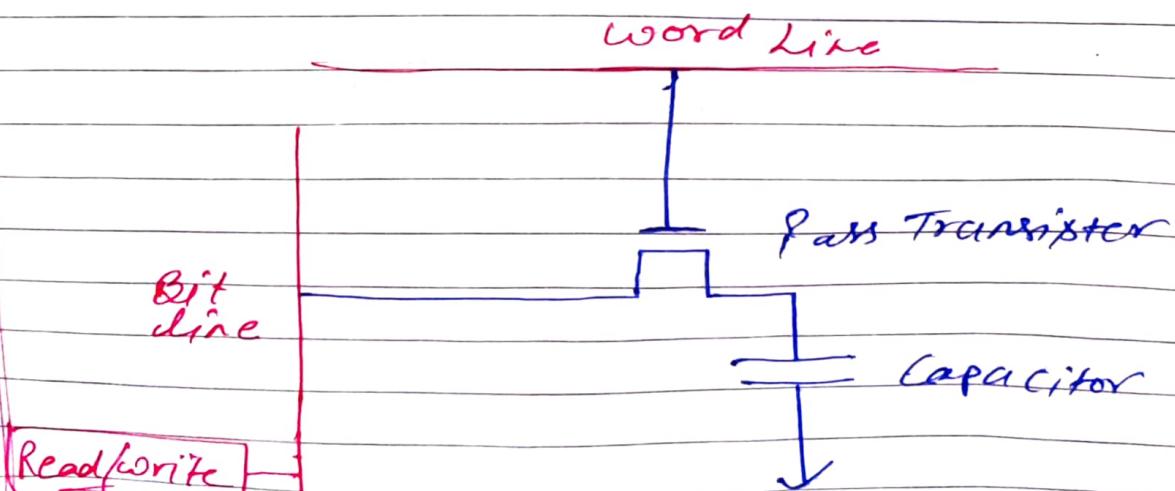
cheaper than SRAM

Density

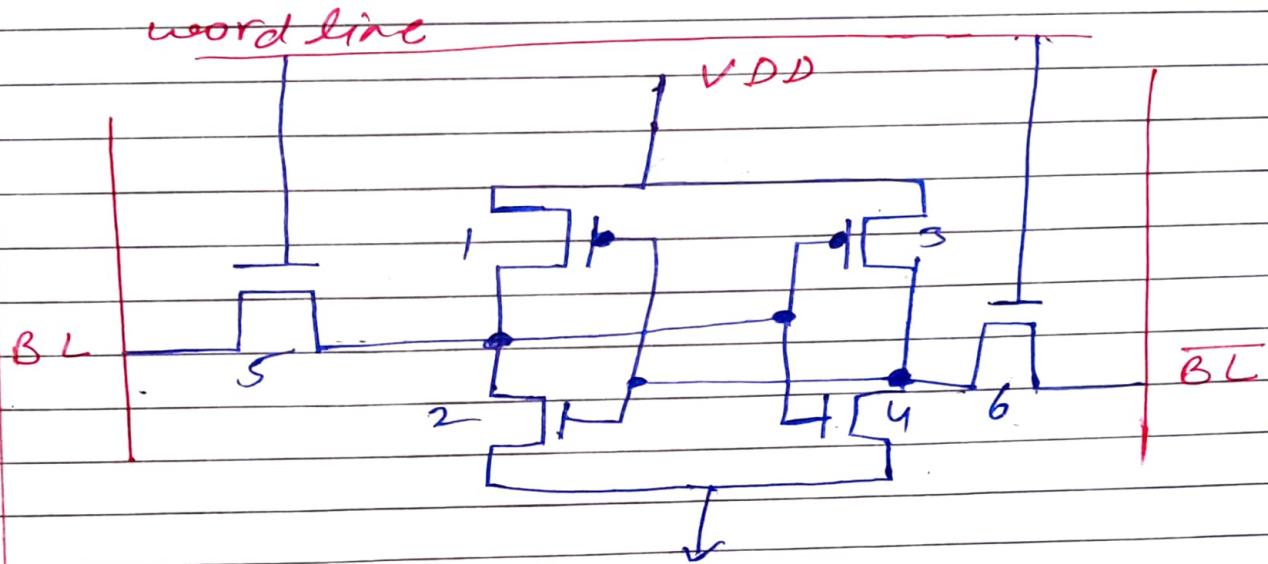
Low

High

DRAM cell:-



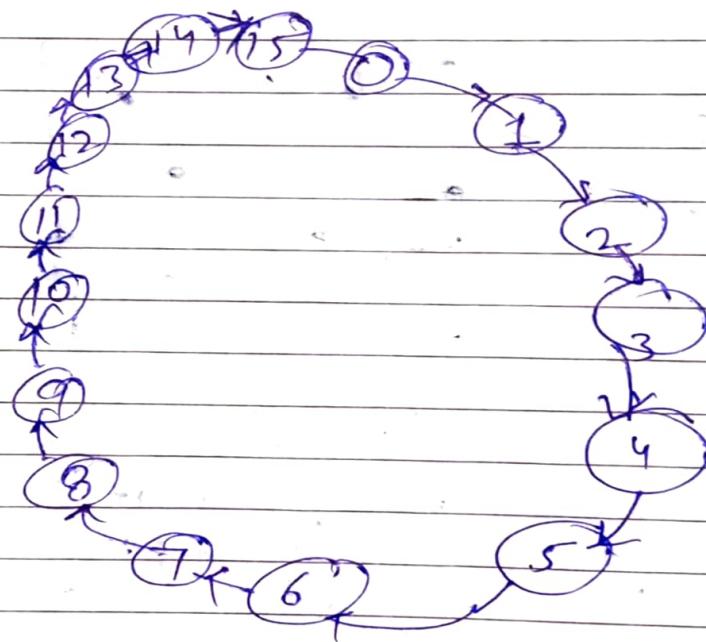
SRAM Cell :-



- (*) Constructed using flip flops.
- (*) As long as power is applied it is able to retain its data.
- (*) This reason helps avoiding refreshing.
- (*) It has :- (1) 6 transistors
- (*) Cache m/r orgⁿ (2) Control lines
- (*) Virtual m/r orgⁿ (3) 1 DC voltage
- (*) Associative m/r. (4) Bit lines (B, B̄)
- (5) Address line.

(*) 4 bit binary synchronous counter with D FF.

Sol:- State diagram:-



Excitation table of D flip flop:-

A	A_{n+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

Excitation Table of 4 bit Counter using D flip-flop

Present State				Next State				Flip-flop Input			
Q_3	Q_2	Q_1	Q_0	$Q_3(n+1)$	$Q_2(n+1)$	$Q_1(n+1)$	$Q_0(n+1)$	D_3	D_2	D_1	D_0
0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	1	0	0	1	0	0	0	1	0
0	0	1	0	0	0	1	1	0	0	1	1
0	0	1	1	0	1	0	0	0	1	0	0
0	1	0	0	0	1	0	1	0	1	0	1
0	1	0	1	0	1	1	0	0	1	1	0
0	1	1	0	0	1	1	1	0	1	1	1
0	1	1	1	1	0	0	0	1	0	0	0
1	0	0	0	1	0	0	1	1	0	0	1
1	0	0	1	1	0	1	0	1	0	1	0
1	0	1	0	1	0	1	1	1	0	1	1
1	0	1	1	1	1	0	0	1	1	0	1
1	1	0	0	1	1	0	1	1	1	0	0
1	1	0	1	1	1	1	0	1	1	1	1
1	1	1	0	1	1	1	1	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0	0

Kmap for D_3

D_3	$Q_3 Q_2$	$Q_1 Q_0$	00	01	11	10
00	00	00				
01	01	11	11	11	11	11
10	11	11	11	11	11	11

$$D_3 = Q_3 Q_2' + Q_3 Q_1' + Q_3 Q_0' + Q_3' Q_2 Q_1 Q_0$$

$$= Q_3 (Q_2' + Q_1' + Q_0') + Q_3' (Q_2 Q_1 Q_0)$$

$$= Q_3 (Q_2 Q_1 Q_0)' + Q_3' (Q_2 Q_1 Q_0)$$

$$D_3 = Q_3 \oplus (Q_2 Q_1 Q_0)$$

Kmap for D_2

		$Q_2 Q_1$	$Q_2 Q_0$	$Q_1 Q_0$	D_2
		00	01	11	10
$Q_3 Q_2$	00	1	1	1	1
	01	1	1	1	1
11	1	1	1	1	1
10	1	1	1	1	1

$$D_2 = Q_2 Q_1' + Q_2 Q_0' + Q_2' Q_1 Q_0$$

$$= Q_2 (Q_1' + Q_0') + Q_2' (Q_1 Q_0)$$

$$= Q_2 (Q_1 Q_0)' + Q_2' (Q_1 Q_0)$$

$$D_2 = Q_2 \oplus (Q_1 Q_0)$$

Kmap for D_1 :-

		$Q_2 Q_1$	$Q_2 Q_0$	$Q_1 Q_0$	D_1
		00	01	11	10
$Q_3 Q_2$	00	1	1	1	1
	01	1	1	1	1
11	1	1	1	1	1
10	1	1	1	1	1

$$D_1 = Q_1' Q_0 + Q_1 Q_0'$$

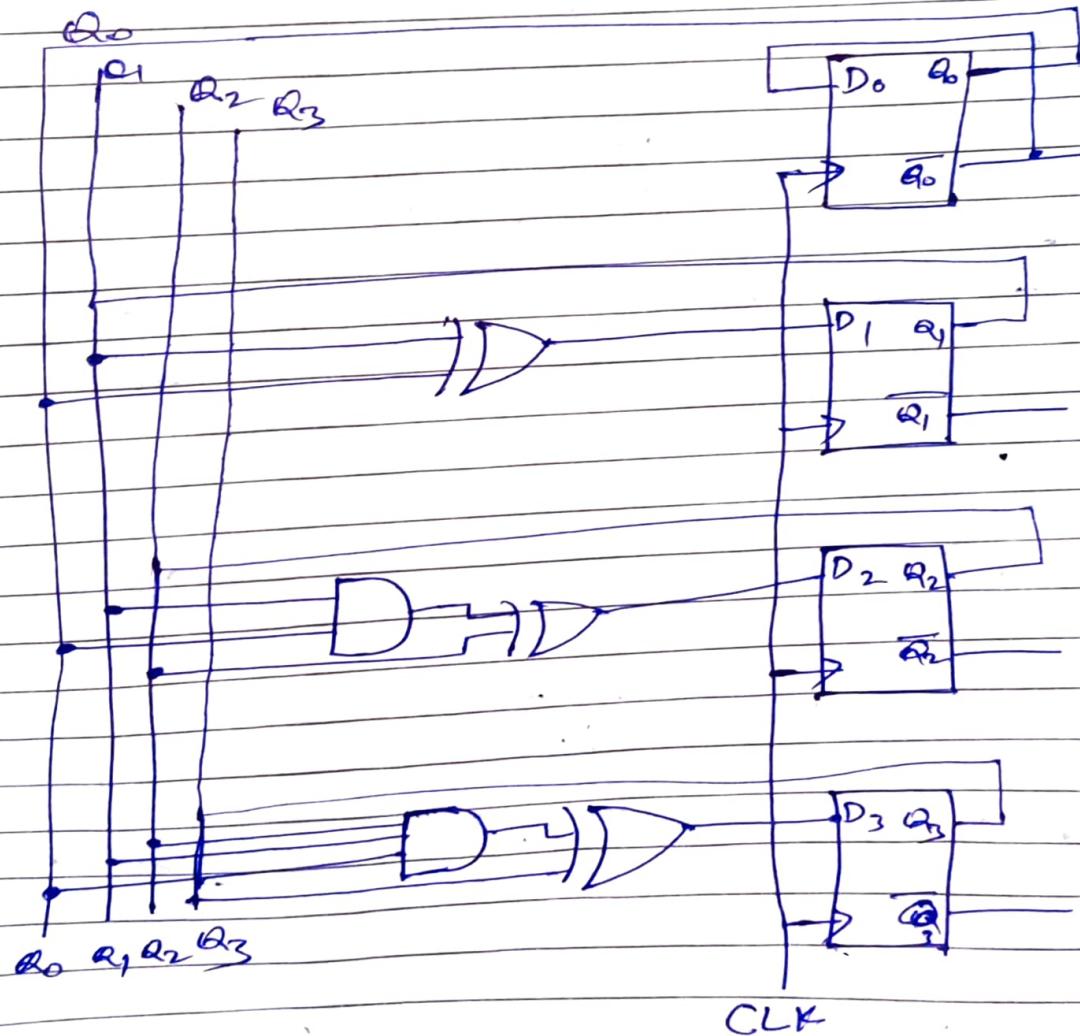
$$D_1 = Q_1 \oplus Q_0$$

K-map for D_0 :-

$Q_3 Q_2 \swarrow Q_1 Q_0$

		00	01	11	10	
		00	1	0	0	1
		01	1	0	0	1
		11	1	0	0	1
		10	1	0	0	1

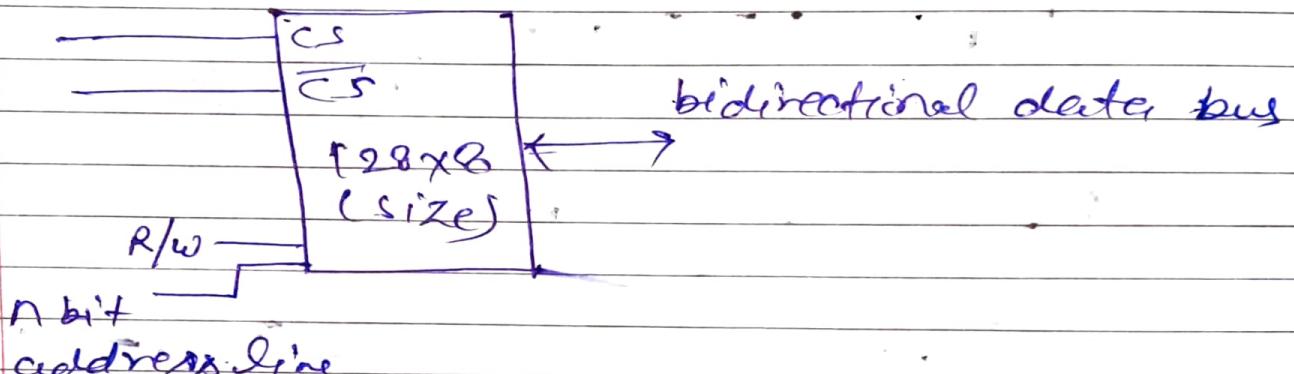
$$D_0 = \overline{Q_0}$$



Logic circuit - 4 bit binary synchronous counter using D-flip flop.

Building large memories using chips:-

↓
Primary memory (RAM)



$128 \rightarrow \text{words}$
Size of 1 word = 8 bit

Size of Ram chip = no. of words \times size of each word

128×8

↓
→ address line

Ram organization \rightarrow making large memory by using small chips.

- (1) make 256×8 Ram using 128×8 Ram
 - (2) 128×8 Ram using 128×1 Ram
 - (3) 256×8 using 128×1
- 3 types
7 Questions

- (i) No. of chips required?
- (ii) No. of Address bits required
- (iii) Decoder size
- (iv) Pictorial Representation.

d) Design a 512×8 Ram using 128×8 Ram Chips.

Sol:- (i) No. of chips Required

$$\frac{\text{Size of desired chip}}{\text{Size of basic Ram chip}} \Rightarrow \frac{512 \times 8}{128 \times 8} = ①$$

4 Ram chips will be required.

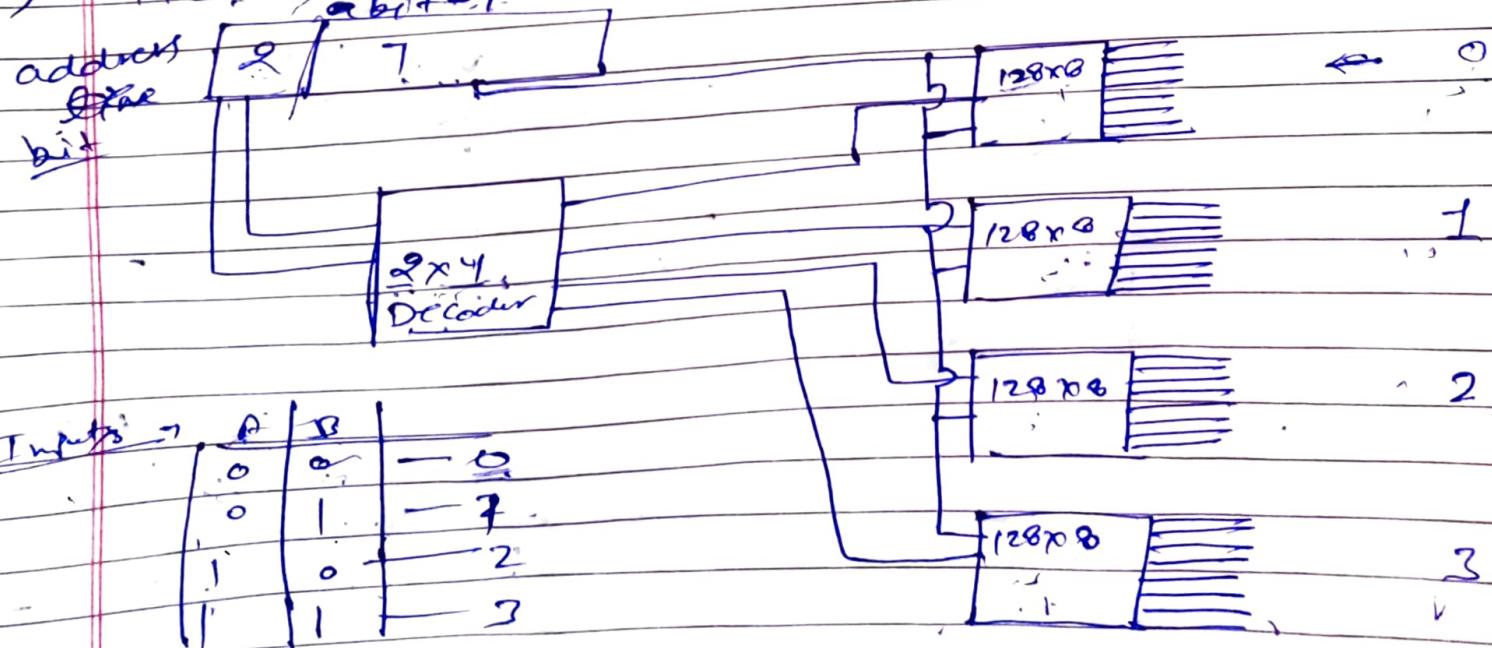
(ii) Address bits required:-

$$512 \times 8 \Rightarrow 2^9 \rightarrow 9 \text{ address bits required.}$$

(iii) Decoder size - No. of words increasing. = ④

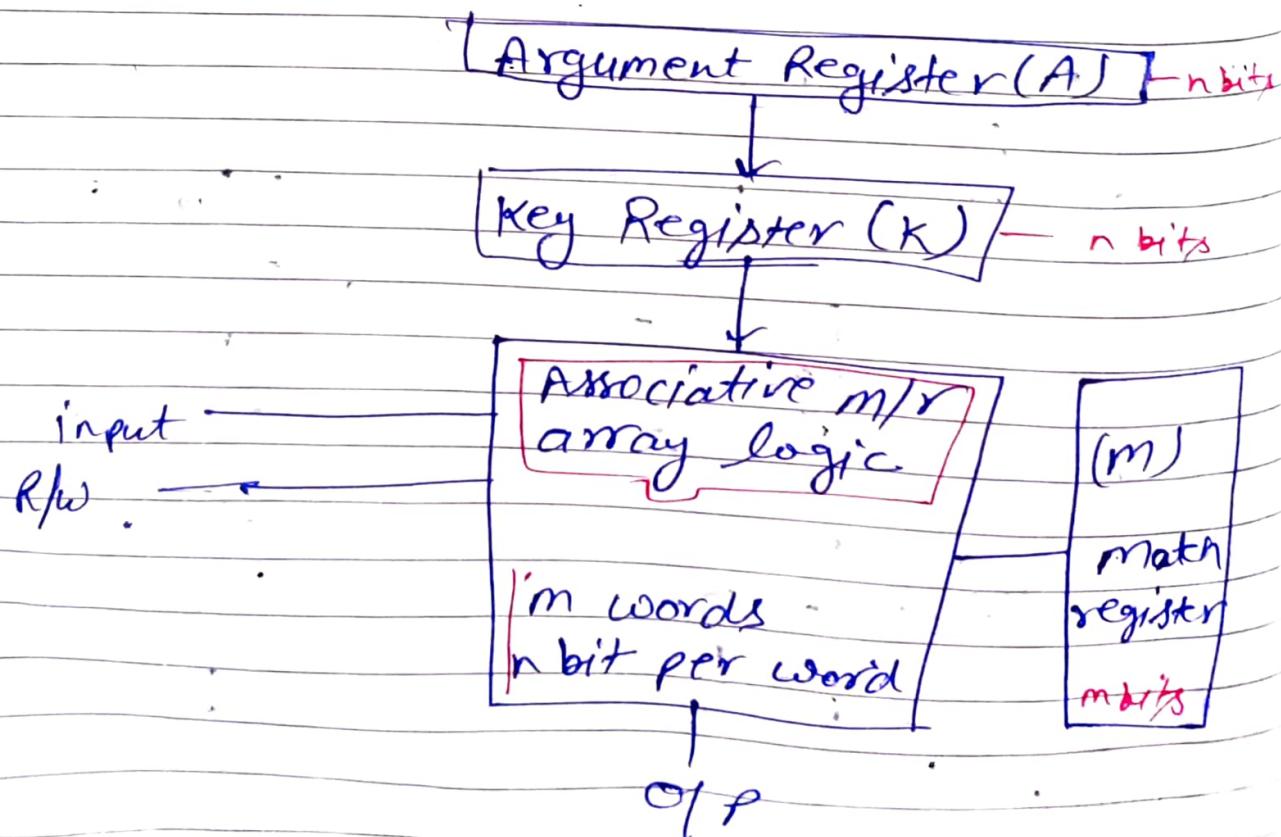
$$2^9 \rightarrow 4 \Rightarrow 2 \times 4 \text{ decoder.}$$

(iv) Pictorial Representation:-



Associative memory: - when data is accessed by data Content rather than data address, then the memory is called associative memory or content addressable memory (CAM).

- (*) Data is stored at the very first empty location found in memory.
- (*) In associative memory when data is stored at a particular location but no address is stored along with it.
- (*) When the stored data need to searched then only the key (i.e. data or part of data) is provided.



Applications of Associative memory:-

- (i) It can only used in mtr allocation format.
- (ii) It is widely used in the database management etc

Advantages:- It is used where search time needs to be less or short.

- (*) It is suitable for parallel searches.
- (*) It is often used to speedup database.
- (*) It is used in page tables used by the virtual memory.

Disadvantages:-

- (*) More expensive than Ram.
- (*) Each cell must have storage capability & logical circuits for matching its content with external argument.

Associative memory :-

- (1) A memory unit accessed by content is called an associative memory or Content Addressable memory (CAM).
- (2) Associative memory is accessed simultaneously & in parallel on the basis of data content rather by specific address or location.
- (3) When a word is written in an associative memory, no address is given.
- (4) This memory is capable of finding an empty unused location to store word.
- (5) When a word is to be "read" from an associative memory, the content of word is specified or part of word is specified.
- (6) The memory locates all words which match the specified content & marks them for reading.
- (7) Because of its organization, the associative memory is uniquely suited to do parallel searches by data association.
- (8) An associative is more expensive than RAM because each cell must have storage capability as well as logic circuits for matching its content with an external argument.

(9) That's why, Associative memory is useful in application where search time is very critical & must be very short.

Argument :- It contain words to be searched.

Register :- It has n bits (one for each bit of word)

Key Register ^(R) :- It has n bits (one for each bit of word). It provides a mask for choosing a particular field/key in arguments. or it specifies which part of the argument word needs to be compared with words in memory.

If all bits in key register are 1's, the entire word should be compared, otherwise, only the bits having 1's in their corresponding position are compared.

Associative :- It contains the word that are to be compared with the ~~memory array~~ argument word in parallel.

Memory Array :- It consists of m words with n bits per word.

Match logic (M) :- It has m bits, one bit corresponding to each word in the memory array.

After the matching process, the bits corresponding to matching words in match register are set to 1.

Reading is accomplished by sequential access in memory for those words whose match bits are set (or 1).

eg if A 1 0 1 1 1 1 0 0

K 1 1 1 0 0 0 0 0

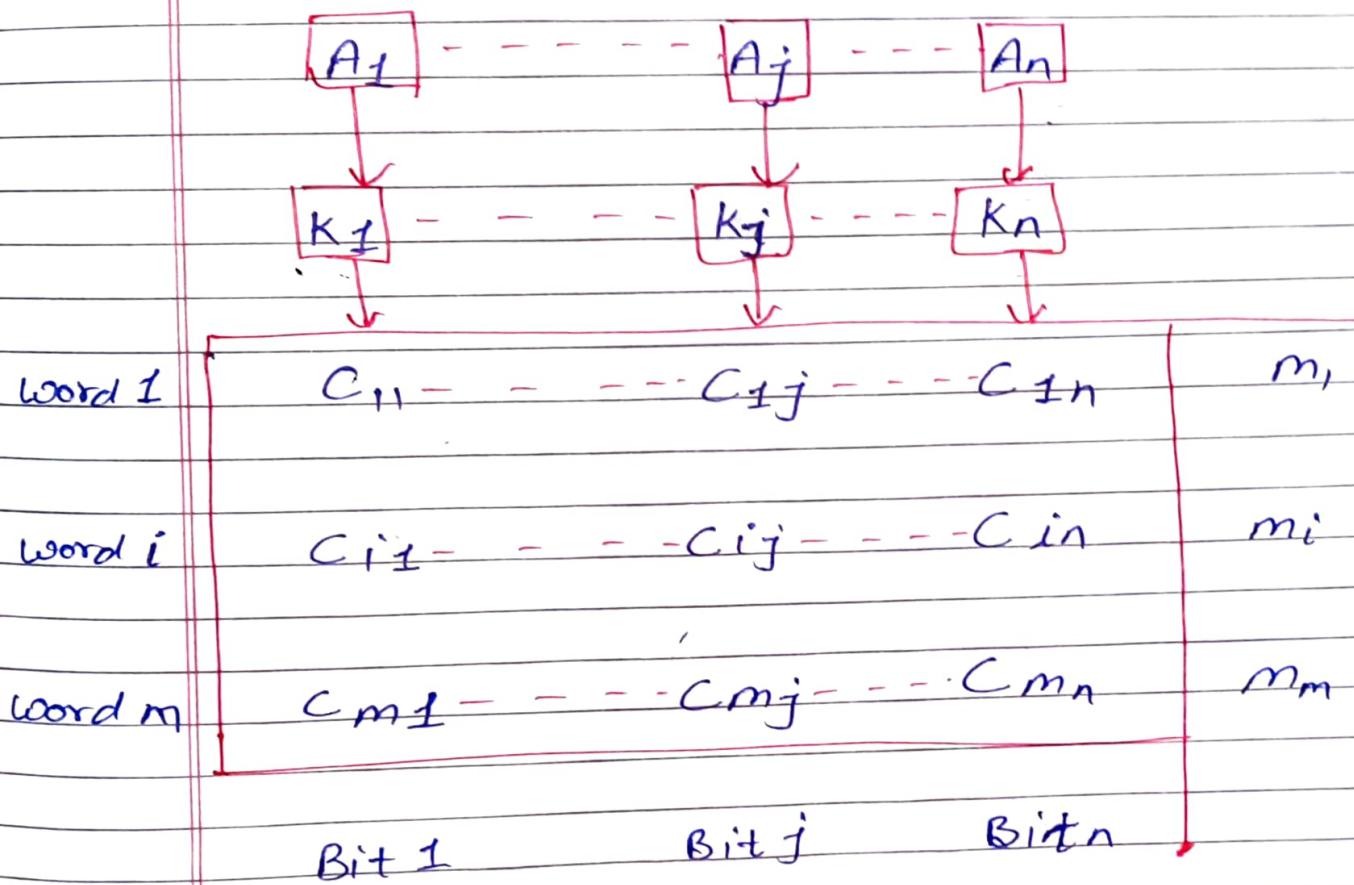
		M
Word 1	1 0 0 1 1 1 1 0 0	<input type="checkbox"/> 0
Word 2	1 0 1 0 0 0 0 0 1	<input type="checkbox"/> 1
Word 3	1 0 1 1 1 1 0 0	<input type="checkbox"/> 1
Word 4	1 1 0 0 0 1 0 1 0	<input type="checkbox"/> 0
Word 5	1 1 1 0 0 0 1 1 1	<input type="checkbox"/> 0

Associative memory of m word, n cells per word

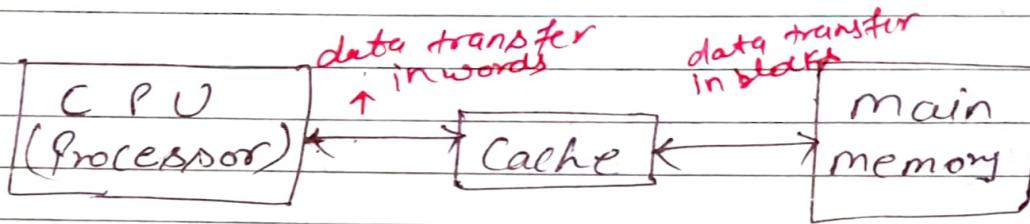
The cell in array is represented by c_{ij} : a cell for bit 'j' in word 'i'

i = word number

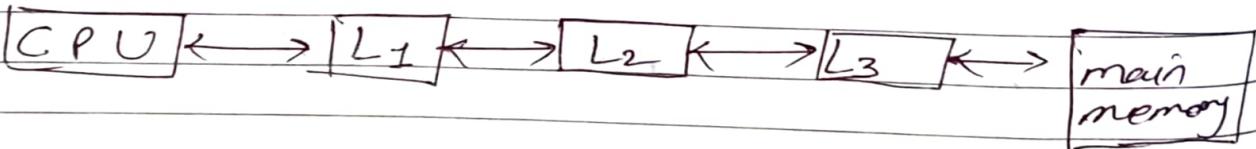
j = bit position in the word.



(*) Cache memory :-



- Small sized fast memory.
- Placed between main memory & CPU.
- High speed volatile memory.
- Contains most frequently accessed instruction & data.
- Located inside the CPU chip or motherboard.
 L₁, L₂ L₃



Working of Cache :-

- (*) The CPU initially looks in the Cache for the data it needs.
- (*) If the data is there, it will retrieve it & process it.
- (*) If the data is not there, then the CPU access the system main memory & then puts a copy of the new data in Cache before processing it.
- (*) Next time, if the CPU needs to access the same data again, it will just retrieve the data from the Cache instead of going through the whole loading process again.

Cache Performance:- It is measured in terms of Hit Ratio.

Cache hit :- If the required word is found in Cache is called hit.

$$\text{Hit Ratio} = \frac{\text{Hits}}{\text{Hits} + \text{miss}} = \frac{\text{no. of hits}}{\text{Total no. CPU references}}$$

Cache Miss:- If the required word is not found in Cache is called cache miss.

$$\text{Miss Ratio} = \frac{\text{miss}}{\text{Hits} + \text{miss}} = \frac{\text{no. of Miss}}{\text{Total no. of Reference}}$$

Cache Access Time:- Time required to access (Cache hit time) : word from the Cache.

Miss Penalty: (Cache miss Time Penalty)
The Time required to fetch the required block from main memory.

Virtual memory:- It appears to be present but actually it is not. It provides illusion of a large memory. virtual memory technique allows users to use more memory for a program than the real memory of a Computer.

Virtual memory is the concept that gives the illusion to the user that they will have main memory equal to the capacity of secondary storage media (or auxillary memory).

Need of virtual memory:-