UNIT V: Memory Organization

FACULTY

Dr. K. ARTHI MCA, M.Phil., Ph.D.,
Assistant Professor,
Postgraduate Department of Computer Applications,
Government Arts College (Autonomous),
Coimbatore-641018.

INTRODUTION TO MEMORY:

Memory unit is an essential component in digital computers since it is needed for storing programs and data. Two or three levels of memory such as

- Main memory
- Secondary memory and
- Cache memory

are provided in a digital computer. The main memory is a fast memory.

MEMORY TYPES :

A memory unit is the collection of storage units or devices together. The memory unit stores the binary information in the form of bits. Generally, memory/storage is classified into 2 categories.

Volatile Memory :

This loses its data, when power is switched off.

Non-Volatile Memory :

This is a permanent storage and does not lose any data when power is switched off.

INTRODUTION FOR MEMORY HIERARCHY:

The **memory hierarchy** system consists of all storage devices contained in a computer system like slow Auxiliary **Memory,** fast Main **Memory** and smaller Cache **memory**. Auxillary **memory** access time is generally 1000 times that of the main **memory**, hence it is at the bottom of the **hierarchy**.

KEY CHARACTERISTICS OF MEMORY SYSTEM

Location
- Internal (e.g. processor registers, main memory, cache)
- External (e.g. optical disks, magnetic disks, tapes)

Capacity
- Number of words
- Number of bytes

Unit of Transfer
- Word
- Block

Access Method
- Sequential
- Direct
- Random
- Associative

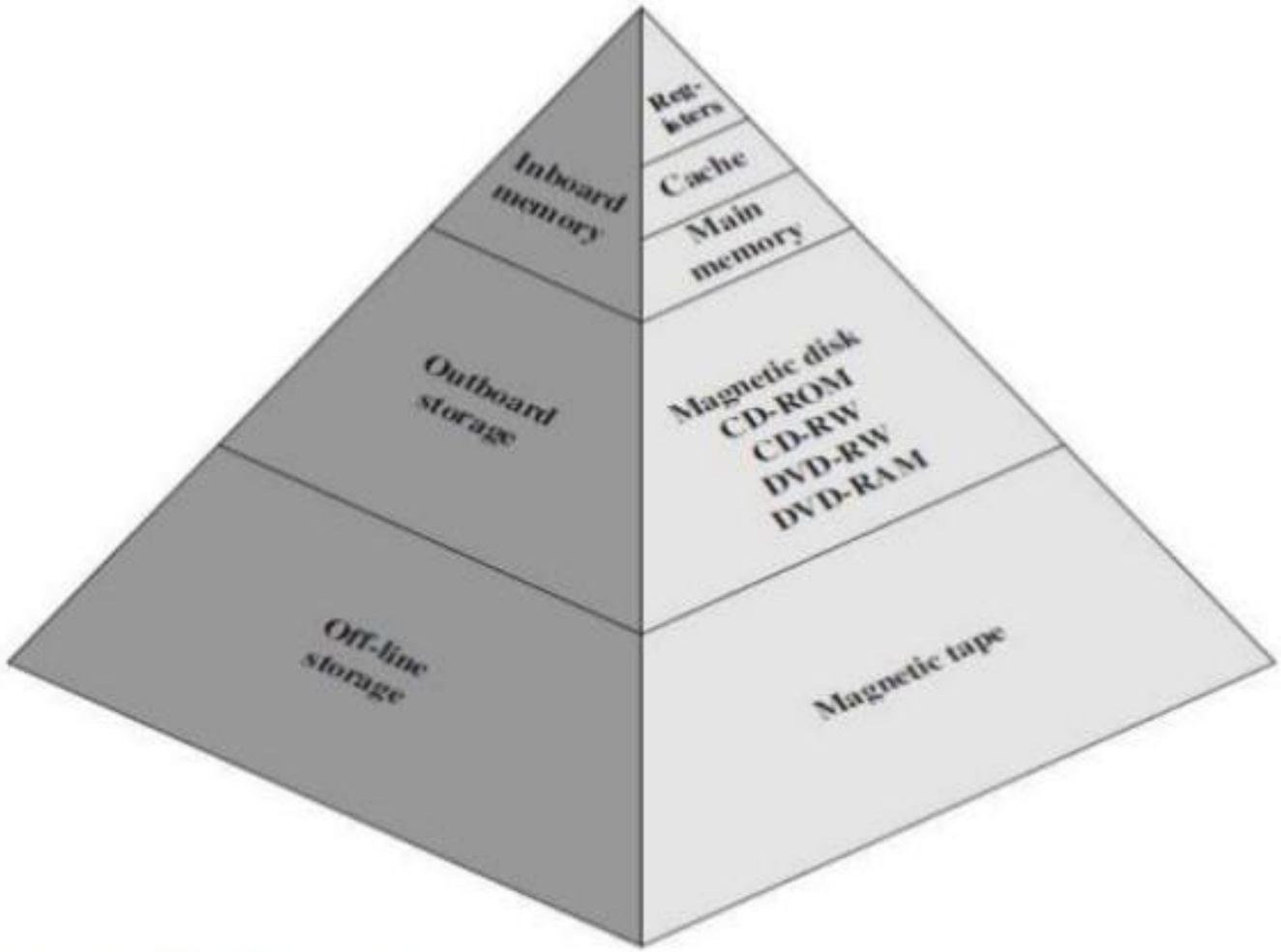
Performance
- Access time
- Cycle time
- Transfer rate

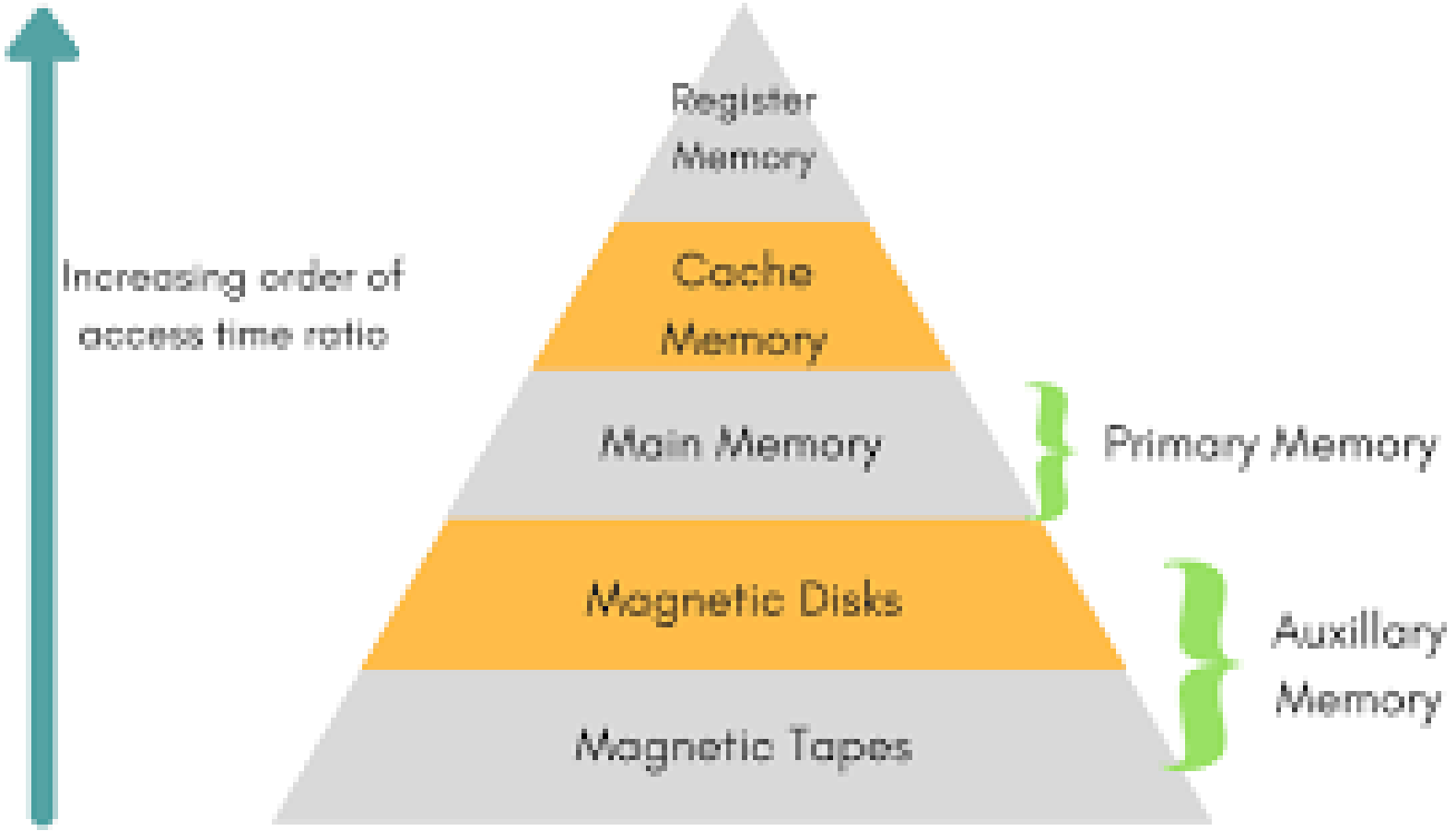
Physical Type
- Semiconductor
- Magnetic
- Optical
- Magneto-optical

Physical Characteristics
- Volatile/nonvolatile
- Erasable/nonerasable

Organization
- Memory modules

MEMORY HIERARCHY




Increasing order of access time ratio
Register Memory
Cache Memory
Main Memory
Magnetic Disks
Magnetic Tapes
Primary Memory
Auxillary Memory

ADVANTAGE OF MEMORY HIERARCHY:

- Allocating memory is easy and cheap
- Any free page is ok, OS can take first one out of list it keeps
- Eliminates external fragmentation
- Data (page frames) can be scattered all over PM
- Pages are mapped appropriately anyway
- Allows demand paging and prepaging
- More efficient swapping
- No need for considerations about fragmentation
- Just swap out page least likely to be used

DISADVANTAGE OF MEMORY HIERARCHY:

- Longer memory access times (page table lookup)
- Can be improved using TLB
- Guarded page tables
- Inverted page tables
- Memory requirements (one entry per VM page)
- Improve using Multilevel page tables and variable page sizes (super-pages)
- Guarded page tables
- Page Table Length Register (PTLR) to limit virtual memory size
- Internal fragmentation

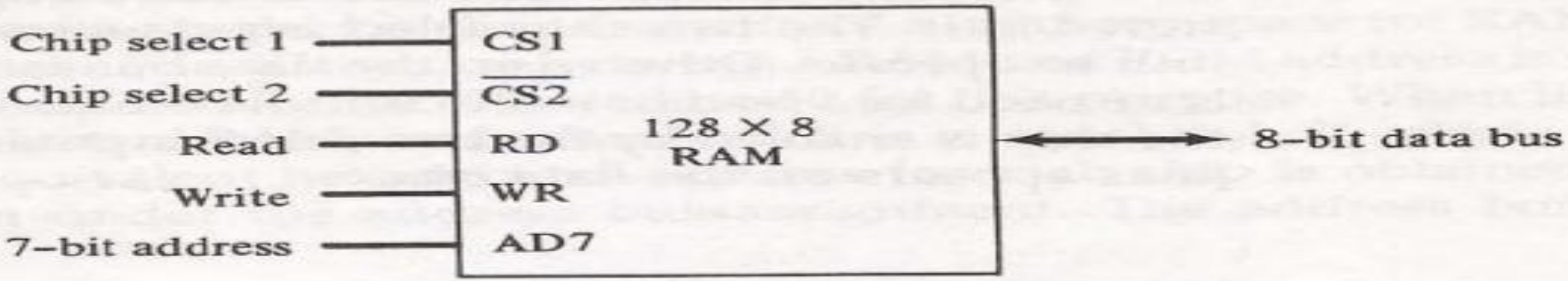
Main Memory

- It is the memory used to store programs and data during the computer operation.
- The principal technology is based on semiconductor integrated circuits.
- It consists of RAM and ROM chips.
- RAM chips are available in two form static and dynamic.

SRAM	DRAM
Uses capacitor for storing information	Uses Flip flop
More cells per unit area due to smaller cell size.	Needs more space for same capacity
Cheap and smaller in size	Expensive and bigger in size
Slower and analog device	Faster and digital device
Requires refresh circuit	No need
Used in main memory	Used in cache

- ROM is uses random access method.
- It is used for storing programs that are permanent and the tables of constants that do not change.
- ROM store program called bootstrap loader whose function is to start the computer software when the power is turned on.
- When the power is turned on, the hardware of the computer sets the program counter to the first address of the bootstrap loader.

Figure Typical RAM chip.



(a) Block diagram

CS1	$\overline{CS2}$	RD	WR	Memory function	State of data bus
0	0	×	×	Inhibit	High-impedance
0	1	×	×	Inhibit	High-impedance
1	0	0	0	Inhibit	High-impedance
1	0	0	1	Write	Input data to RAM
1	0	1	×	Read	Output data from RAM
1	1	×	×	Inhibit	High-impedance

(b) Function table

TABLE Memory Address Map for Microprocomputer

Component	Hexadecimal address	Address bus 10	9	8	7	6	5	4	3	2	1
RAM 1	0000–007F	0	0	0	x	x	x	x	x	x	x
RAM 2	0080–00FF	0	0	1	x	x	x	x	x	x	x
RAM 3	0100–017F	0	1	0	x	x	x	x	x	x	x
RAM 4	0180–01FF	0	1	1	x	x	x	x	x	x	x
ROM	0200–03FF	1	x	x	x	x	x	x	x	x	x

Figure 12-4 Memory connection to the CPU.

- 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,
- For this reason the diagram specifies 512 byte ROM and 128 bytes RAM.

Memory address Map

- Designer must specify the size and the type(RAM or ROM) of memory to be used for particular application.
- The addressing of the memory is then established by means of table called memory address map that specifies the memory address assign to each chip.
- Let us consider an example in which computer needs 512 bytes of RAM and ROM as well and we have to use the chips of size 128 bytes for RAM and 512 bytes for ROM.

TABLE Memory Address Map for Microprocomputer

Component	Hexadecimal address	Address bus									
		10	9	8	7	6	5	4	3	2	1
RAM 1	0000–007F	0	0	0	x	x	x	x	x	x	x
RAM 2	0080–00FF	0	0	1	x	x	x	x	x	x	x
RAM 3	0100–017F	0	1	0	x	x	x	x	x	x	x
RAM 4	0180–01FF	0	1	1	x	x	x	x	x	x	x
ROM	0200–03FF	1	x	x	x	x	x	x	x	x	x

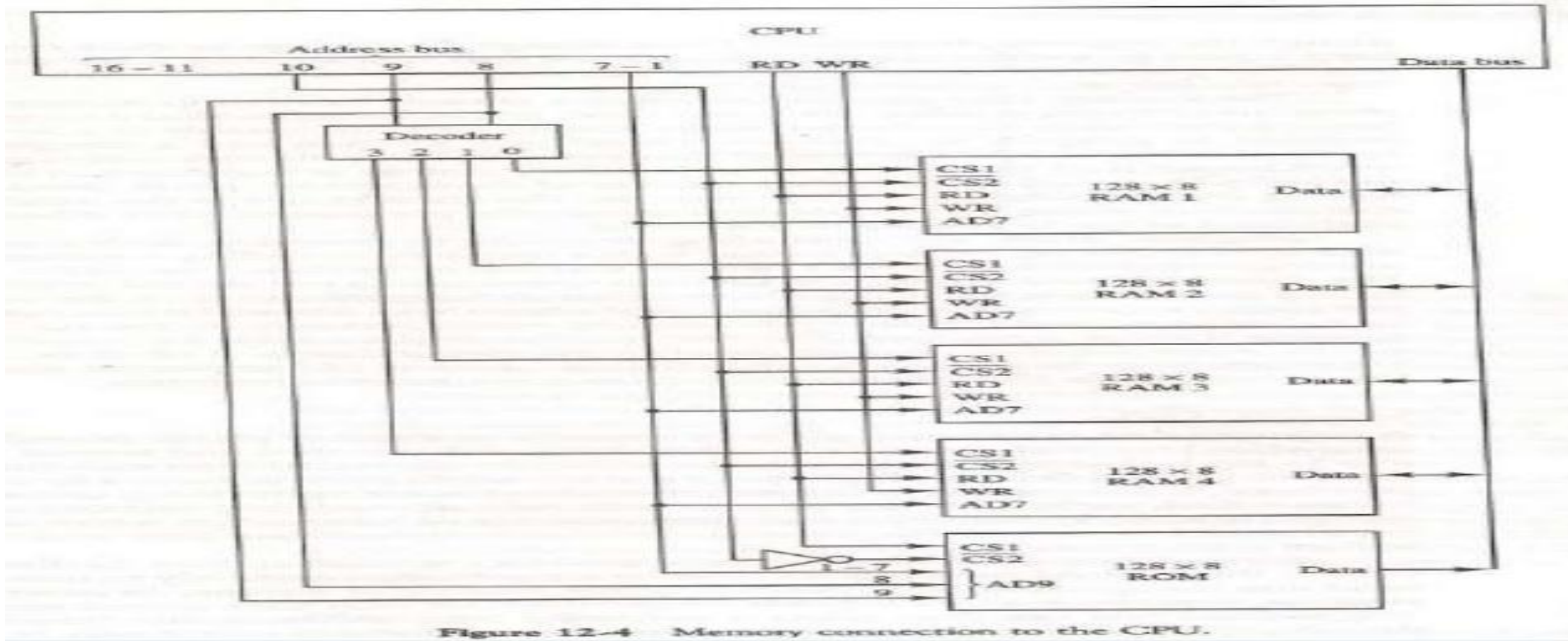


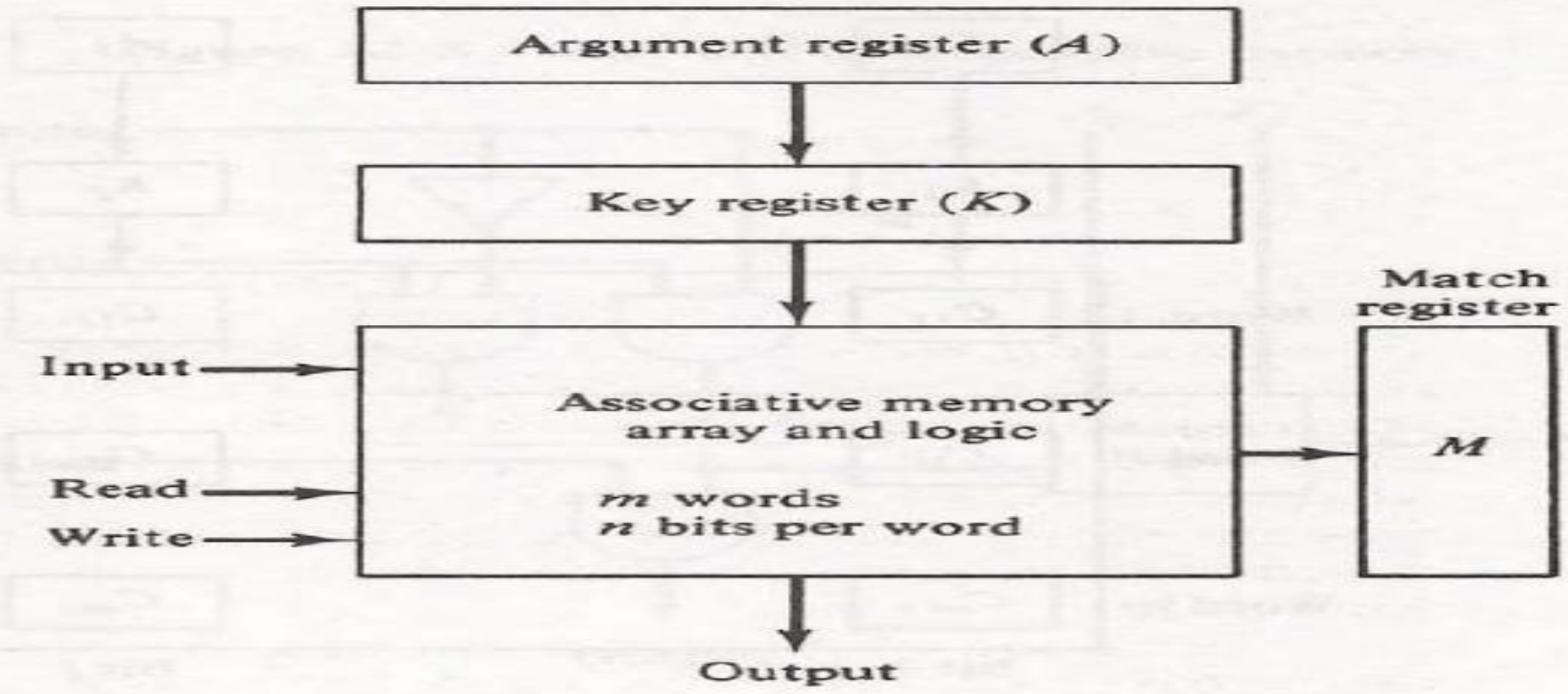
Figure 12-4 Memory connection to the CPU.

Associative Memory

- To search particular data in memory, data is read from certain address and compared if the match is not found content of the next address is accessed and compared.
- This goes on until required data is found. The number of access depend on the location of data and efficiency of searching algorithm.
- The searching time can be reduced if data is searched on the basis of content.

- A memory unit accessed by content is called associative memory or content addressable memory(CAM)
- This type of memory is accessed simultaneously and in parallel on the basis of data content.
- Memory is capable of finding empty unused location to store the word.
- These are used in the application where search time is very critical and must be very short.

Figure Block diagram of associative memory.



- It consists memory array of m words with n bits per words
- Argument register A and key register K have n bits one for each bit of word.
- Match register has m bits, one for each memory word.
- Each word in memory is compared in parallel with the content of the A register. For the word that match corresponding bit in the match register is set.

- Key register provide the mask for choosing the particular field in A register.
- The entire content of A register is compared if key register content all 1.
- Otherwise only bit that have 1 in key register are compared.
- If the compared data is matched corresponding bits in the match register are set.
- Reading is accomplished by sequential access in memory for those words whose bit are set.

A	101 111100	
K	111 000000	
Word 1	100 111100	no match
Word 2	101 000001	match

Figure Associative memory of m word, n cells per word.

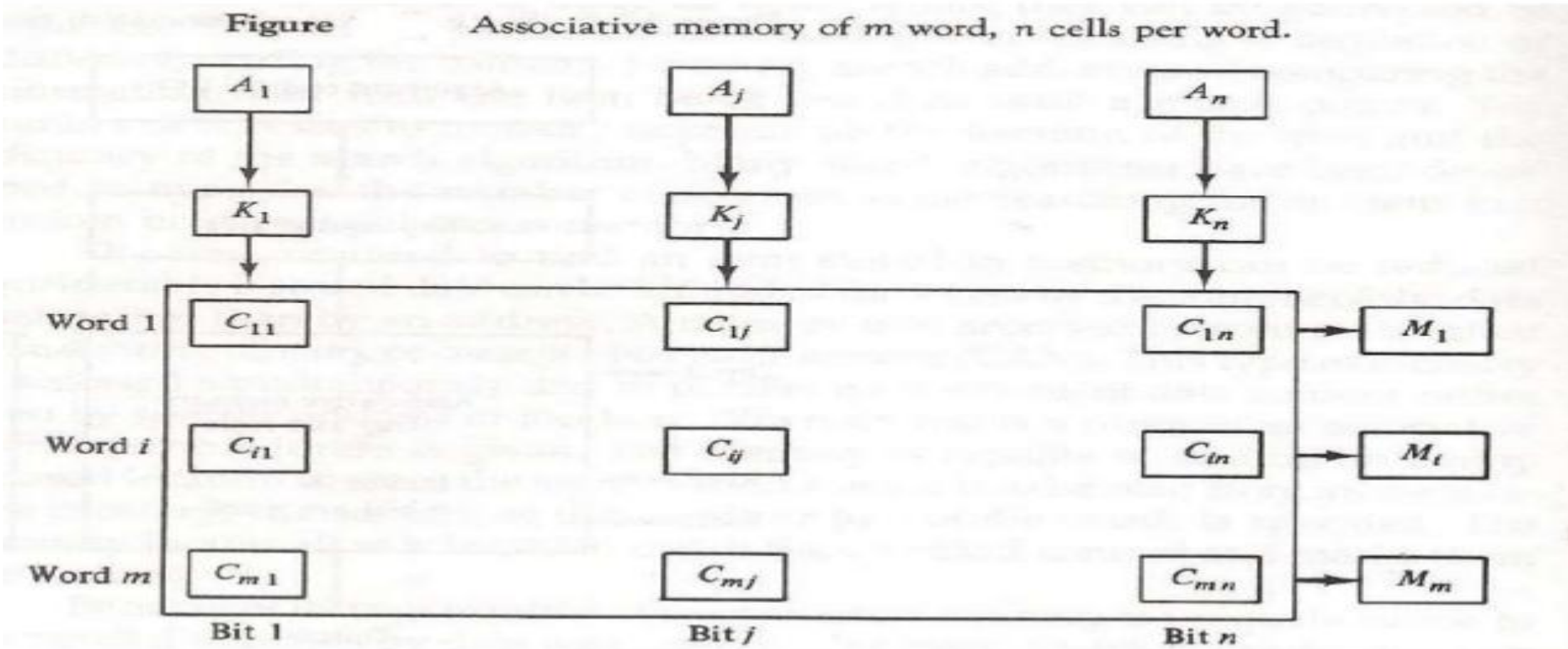
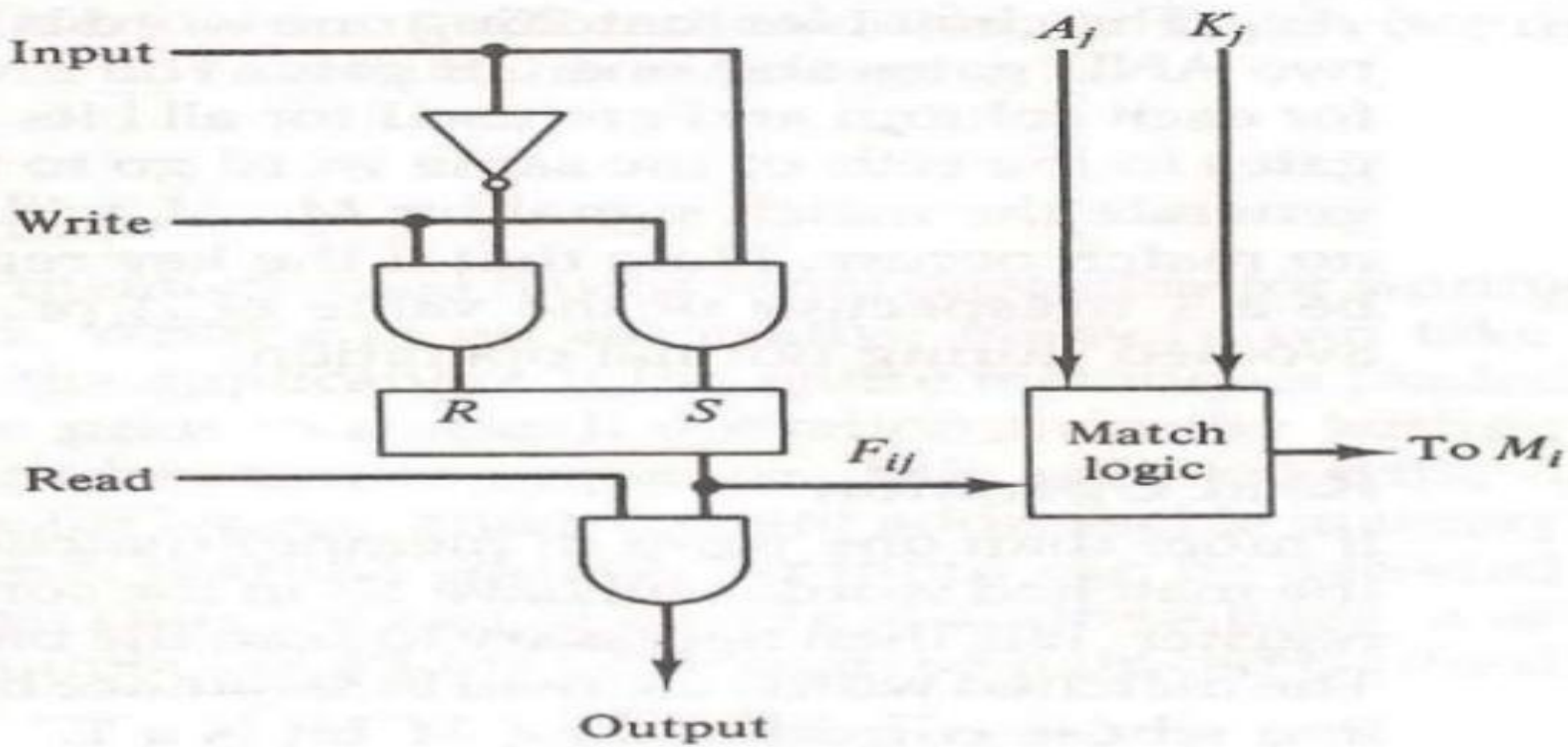


Figure One cell of associative memory.



- Let us include key register. If K_j=0 then there is no need to compare A_j and F_{ij}.
- Only when K_j=1, comparison is needed.
- This achieved by ORing each term with K_j.

$$M_i = (x_1 + K_1')(x_2 + K_2')(x_3 + K_3') \cdots (x_n + K_n')$$

Match Logic

- Let us neglect the key register and compare the content of argument register with memory content.
- Word i is equal to argument in A if $A_j = F_{ij}$ for j=1,2,3,4.....n
- The equality of two bits is expressed as

$$x_j = A_j F_{ij} + A_j' F_{ij}'$$

- $x_j = 1$ if bits are equal and 0 otherwise.

$$M_i = x_1 x_2 x_3 \cdots x_n$$

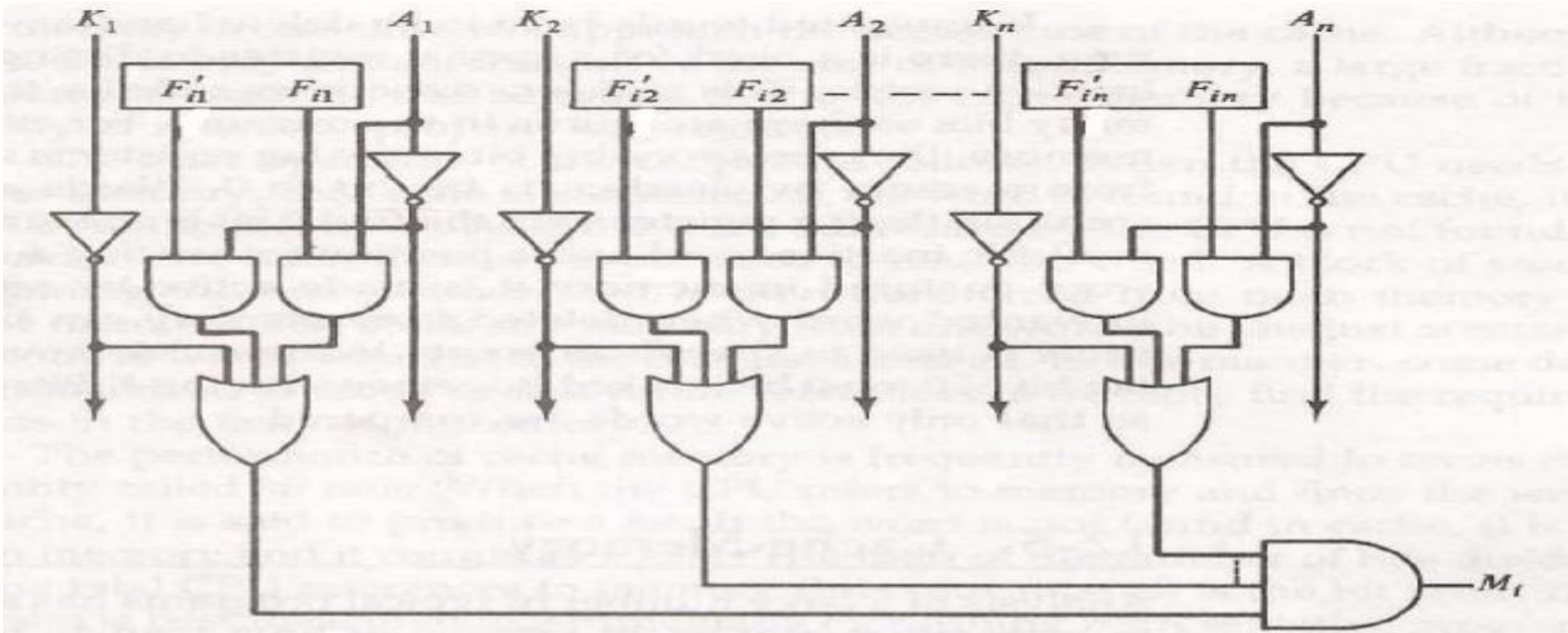


Figure Match logic for one word of associative memory.

Read Operation

- If more than one word match with the content, all the matched words will have 1 in the corresponding bit position in match register.
- Matched words are then read in sequence by applying a read signal to each word line.
- In most application, the associative memory stores a table with no two identical items under a given key.

Write Operation

- If the entire memory is loaded with new information at once prior to search operation then writing can be done by addressing each location in sequence.
- Tag register contain as many bits as there are words in memory.
- It contain 1 for active word and 0 for inactive word.
- If the word is to be inserted, tag register is scanned until 0 is found and word is written at that position and bit is change to 1.

Cache Memory

- Analysis of large number of program shows that reference to memory at any given interval of time tend to be confined to few localized area in memory. This is known as locality of reference.
- If the active portion of program and data are placed in fast memory, then average execution time of the program can be reduced. Such fast memory is called cache memory.
- It is placed in between the main memory and the CPU.

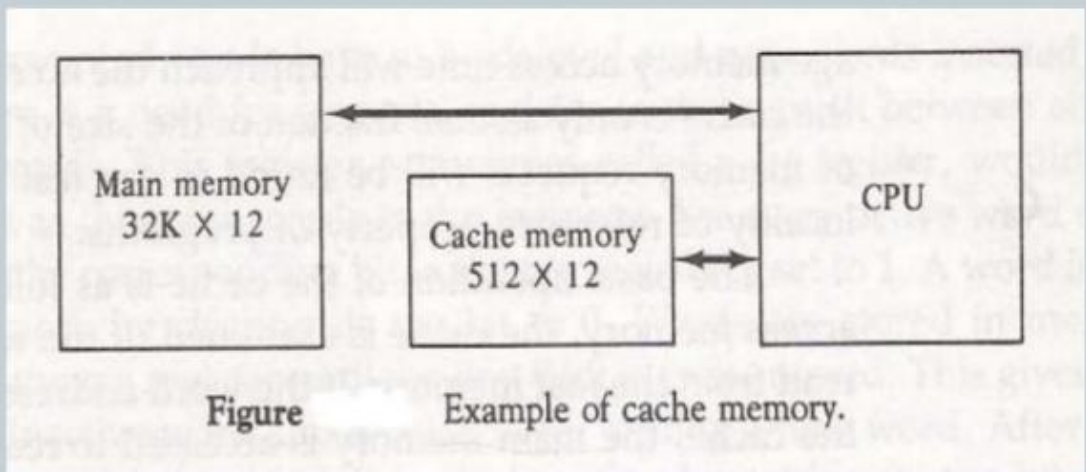


Figure Example of cache memory.

THANK YOU

This content is taken from the text books and reference books prescribed in the syllabus.