COMPUTER ORGANIZATION AND ARCHITECTURE

Memory Organization

Dr. Bore Gowda S B
Additional Professor
Dept. of ECE
MIT, Manipal

Topics to be covered

- Memory Hierarchy
- Main Memory
- Auxiliary Memory
- Associative Memory
- Cache Memory,
- Virtual Memory
- Memory Management

- **Reference book:** M Morris Mano, “Computer System Architecture”, 3rd Edition

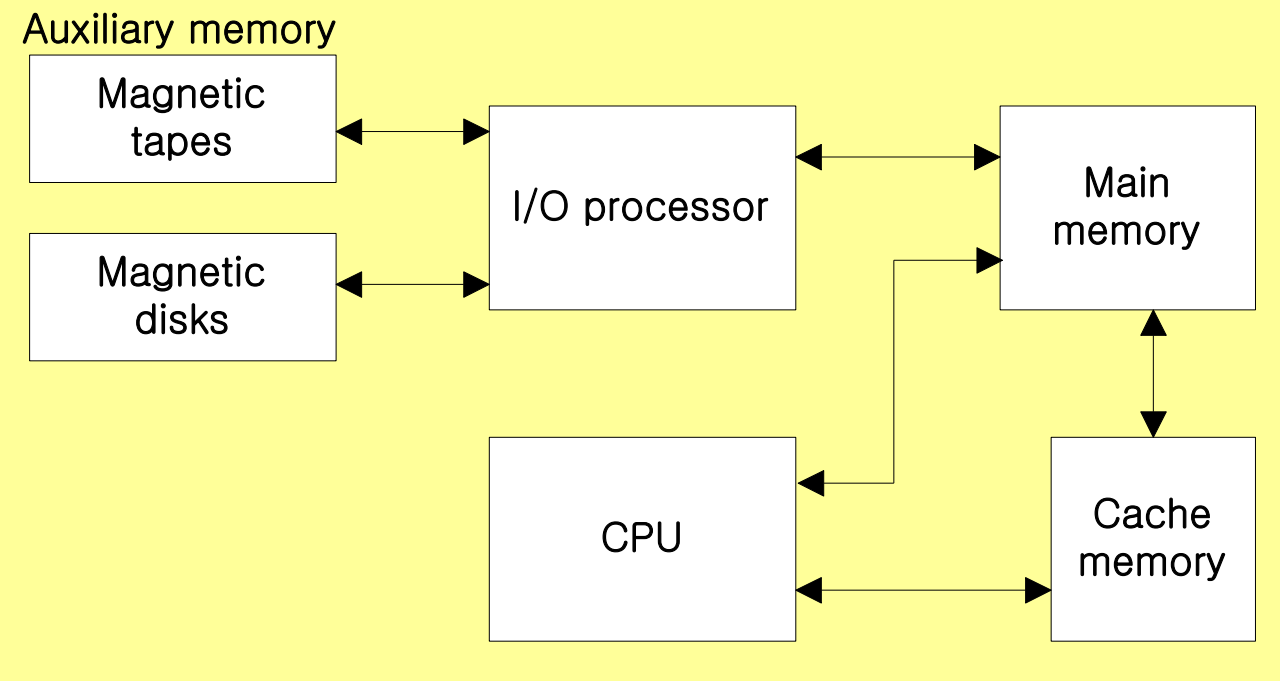
Chapter 12 - Memory Organization

Memory Hierarchy

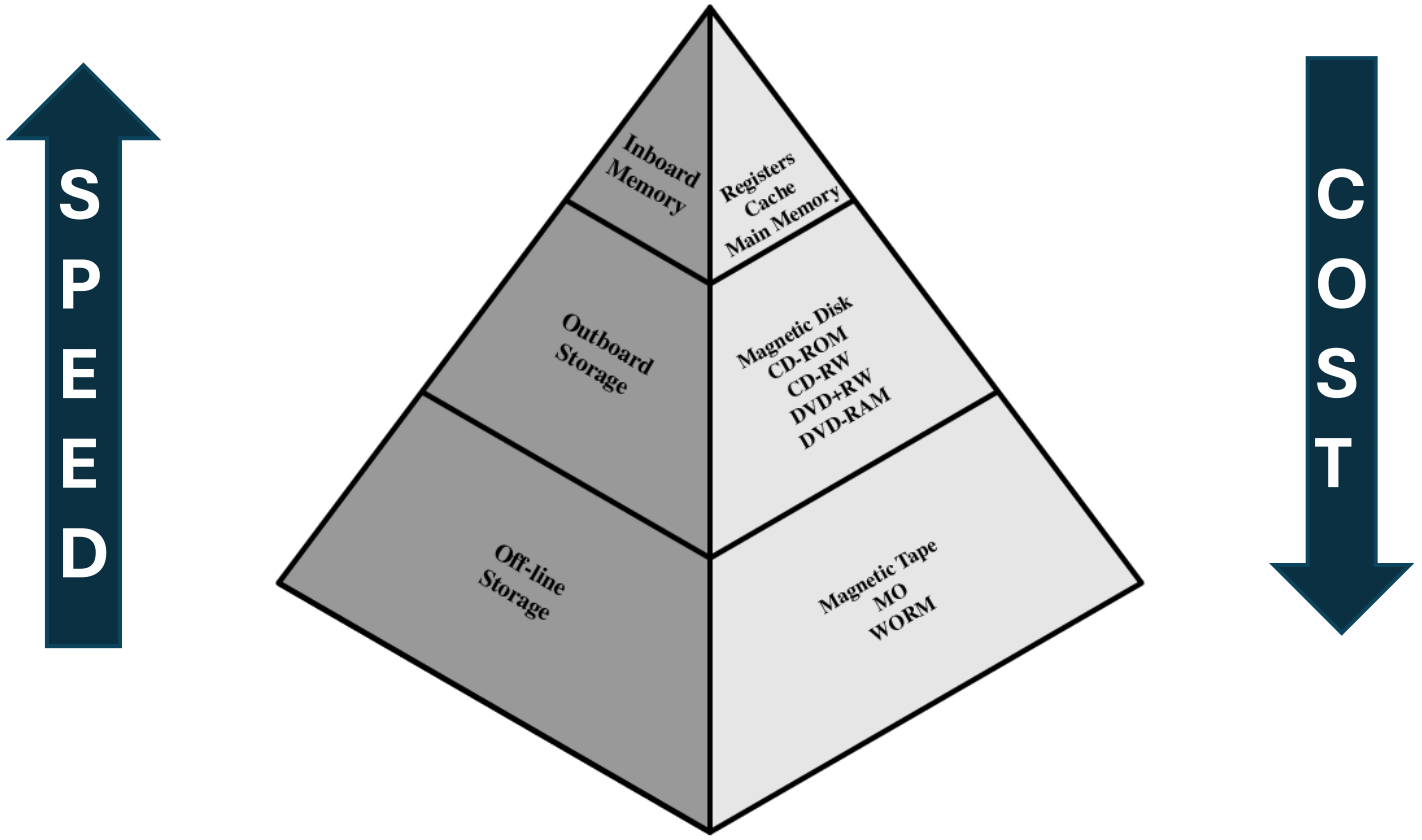
- Memory hierarchy in a computer system
- **Main memory:** The memory unit that communicates directly with the CPU
- **Auxiliary memory:** Devices that provide backup storage
  - The most common auxiliary memory devices used in computer systems are magnetic disks and tapes.
- **Cache:** It is high speed memory used to increase the speed of processing by making current programs and data available to CPU at a rapid rate.
- The total memory capacity of a computer can be visualized as being a hierarchy of components.
- The memory hierarchy system consists 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 Cache memory.

Memory Hierarchy

- **Typical memory hierarchy**
- At the bottom of the hierarchy are the relatively slow magnetic tapes used to store removable files.
- Next are the magnetic disks used as backup storage.
- The main memory able to communicate directly with the CPU and with auxiliary memory devices through an IO processor.
- When programs not residing in main memory are needed by the CPU, they are brought in from auxiliary memory.
- Programs not currently needed in main memory are transferred into auxiliary memory to provide space for currently used programs and data.
- A special very-high speed memory called a Cache is sometimes used to increase the speed of processing by making current programs and data available to the CPU at a rapid rate.



Memory Hierarchy



Memory Hierarchy

- Registers – semiconductor memory on the CPU; fastest
- L1 Cache – Static RAM placed very close to CPU
- L2 Cache – static RAM placed close to CPU
- Main memory – dynamic RAM placed on board
- Disk – for bulk data
- Optical – for bulk data
- Tape – for bulk data

Main Memory

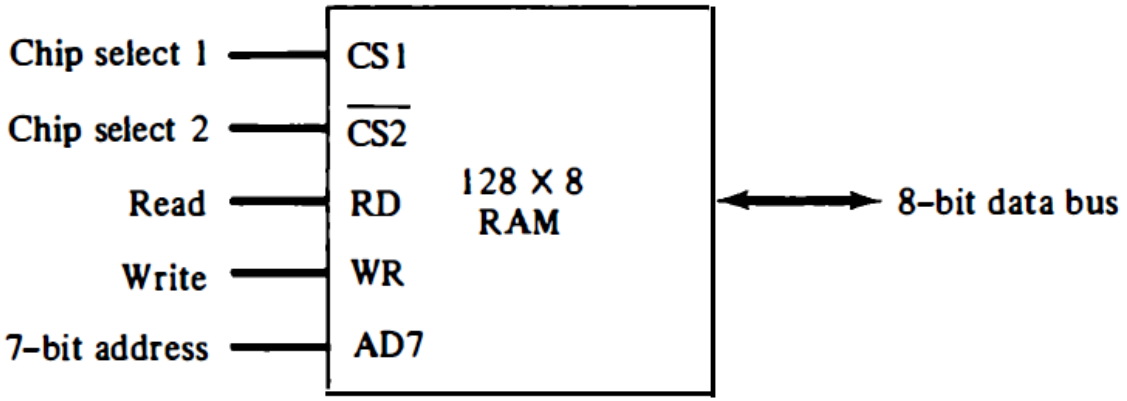
- It is the central storage unit in a computer system.
- Relatively large and fast memory used to store programs and data during the computer operation.
- Main memory is based on semiconductor integrated circuits.
- ICs RAM chips are available in two possible operating modes,
  - Static RAM and dynamic RAM.
- **Static RAM**
  - consists essentially of internal flip-flops
  - stored information remains valid as long as power is applied to the unit.
- **Dynamic RAM**
  - stores the binary information in the form of electric charges that are applied to capacitors.
  - The periodically capacitors are recharged by refreshing the dynamic memory.
  - Refreshing is done by cycling through the words every few milliseconds to restore the decaying charge.
- The dynamic RAM offers reduced power consumption and larger storage capacity in a single memory chip.
- The static RAM is easier to use and has shorter read and write cycles.

Main Memory

- A portion of the memory may be constructed with ROM chips.
- ROM is used for storing programs that are permanently resident in the computer
- ROM portion of main memory is needed for storing an initial program called a **bootstrap loader.**
- The bootstrap loader is a program whose function is to start the computer software operating when power is turned on.
- The bootstrap program loads a portion of the operating system from disk to main memory and control is then transferred to the operating system
- RAM is volatile, but ROM is non-volatile

Main Memory

Typical RAM Chips

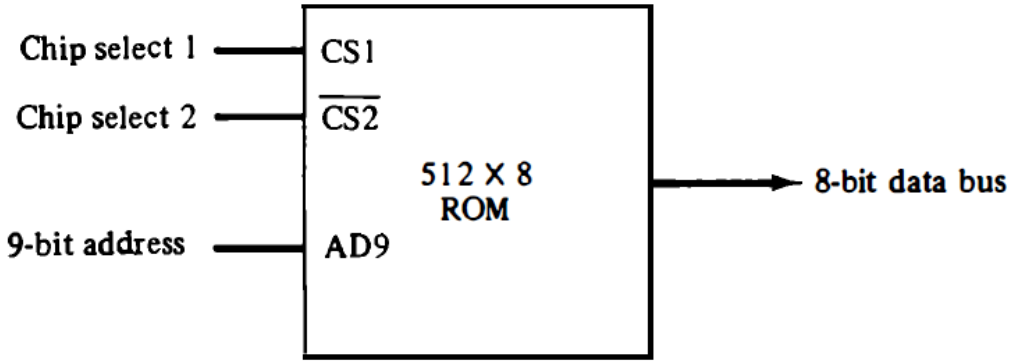


(a) Block diagram

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

Typical ROM Chips



Main Memory

- **Memory Address Map**
- The designer of a computer system must calculate the amount of memory required for the particular application and assign it to either RAM or ROM.
- The interconnection between memory and processor is then established from knowledge of the size of memory needed and the type of RAM and ROM chips available.
- The addressing of memory can be established by means of a table that specifies the memory address assigned to each chip.
- The table, called a memory address map, is a pictorial representation of assigned address space for each chip in the system.

Main Memory

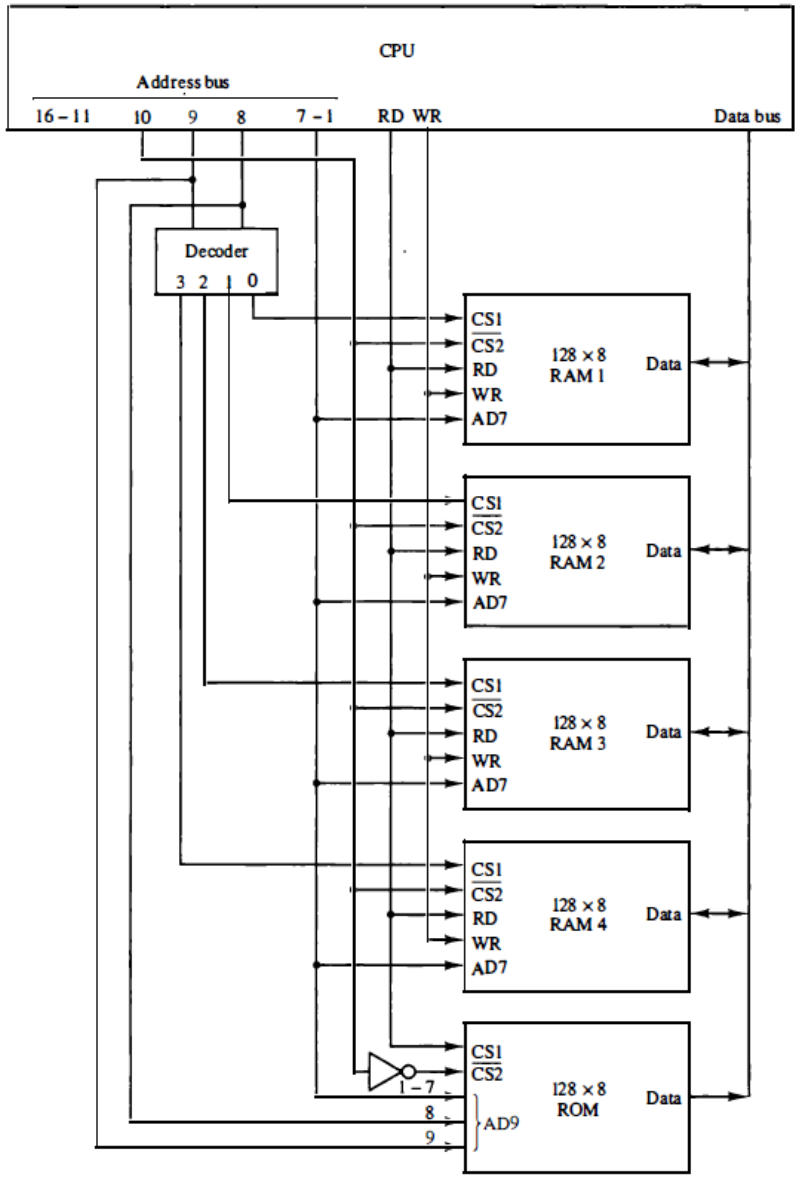
- **Memory Interfacing**
- *The microprocessor in the computer system has 16-bit address lines and 8-bit data lines. Design a computer system with 512 bytes of RAM and 512 bytes of ROM. The memory chips with following capacity are available: 128x8 RAM chip and 512x8 ROM chip*

TABLE 12-1 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

Main Memory

Memory Interfacing

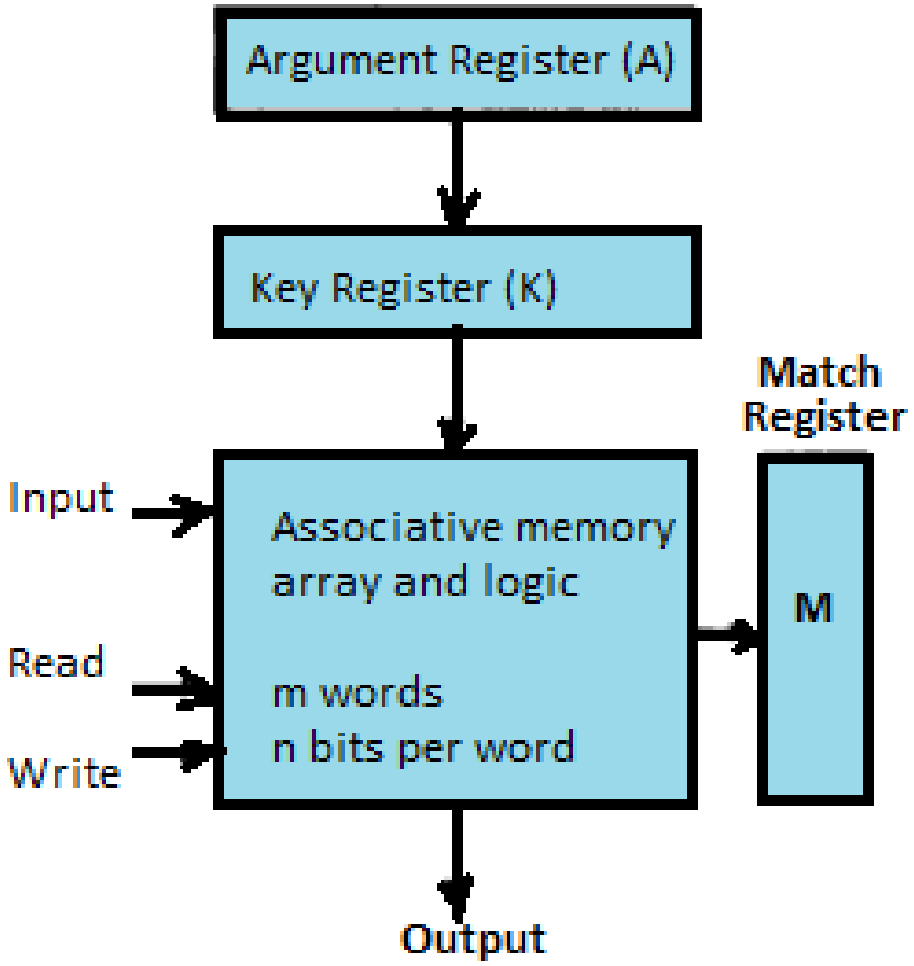


Associative Memory

- Many data-processing applications require the search of items in a table stored in memory.
- The time required to find an item stored in memory can be reduced considerably if *stored data can be identified for access by the content of the data itself rather than by an address*.
- *A memory unit accessed by content is called an associative memory or content addressable memory (CAM).*
- 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 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 marks them for reading.
- Searches can be done on an entire word or on a specific field within a word.
- An associative memory is more expensive than a random access memory because each cell must have storage capability as well as logic circuits for matching its content with an external argument.
- For this reason, associative memories are used in applications where the search time is very critical and must be very short.

Associative Memory

- **Hardware Organization**
- It consists of a memory array and logic for ***m words with n bits per word***.
- A - argument register
- K - key register: key register provides a mask for choosing a particular field or key in the argument word.
- A and K each have n bits, one for each bit of a word.
- The match register M has m bits, one for each memory word.
- **Procedure for searching**
- Each word in memory is compared in parallel with the content of the argument register.
- The words that match the bits of the argument register set a corresponding bit in the match register.
- After the matching process, those bits in the match register that have been set indicate the fact that their corresponding words have been matched.
- Reading is accomplished by a sequential access to memory for those words whose corresponding bits in the match register have been set.



Associative Memory

- **Numerical example.**
- Let A 101 1 1 1 1 0 0 and K 111 0 0 0 0 0 0

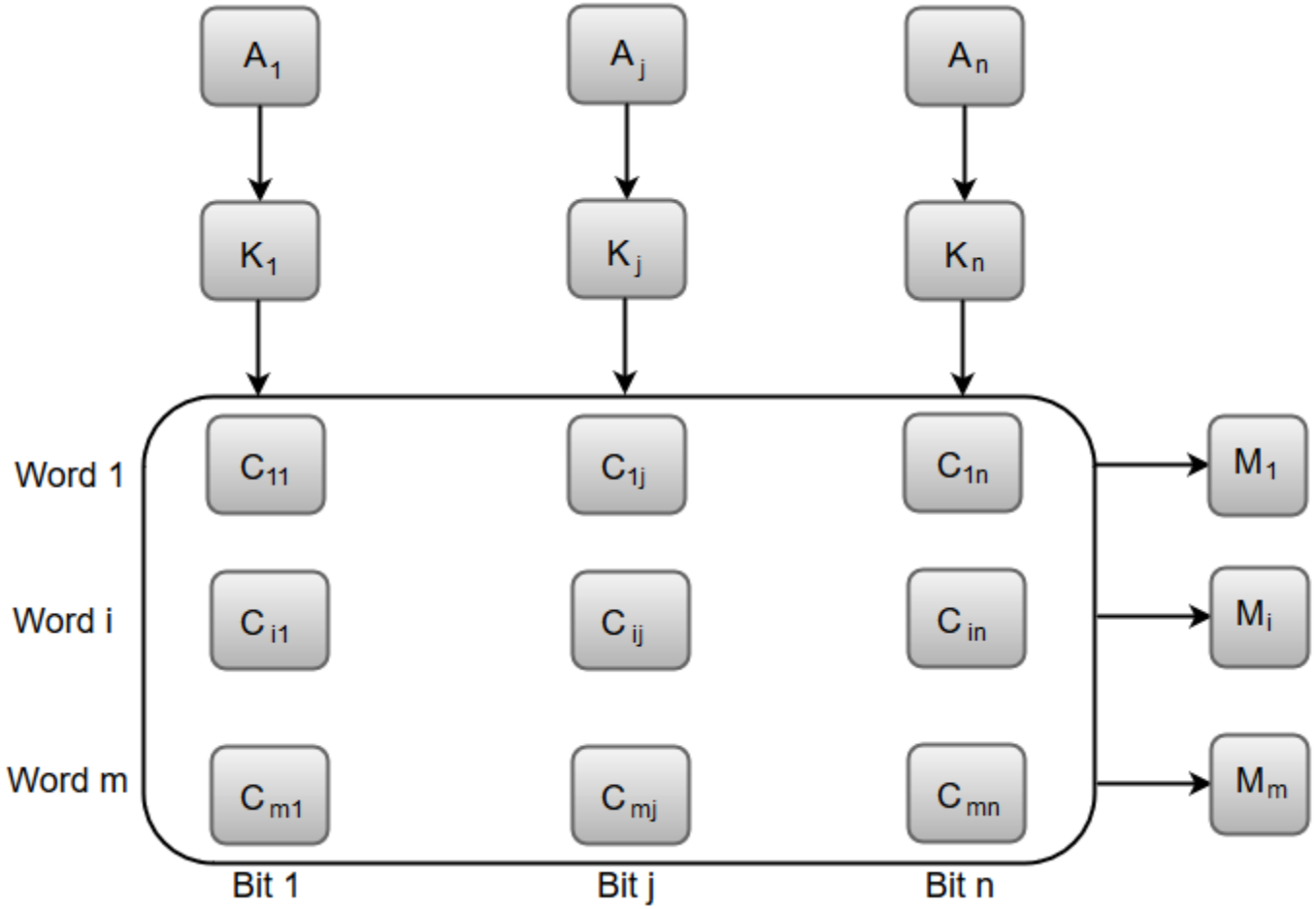
A	101	111100	
K	111	000000	
Word 1	100	111100	no match
Word 2	101	000001	match

- Word 2 matches the unmasked argument field because the three leftmost bits of the argument and the word are equal.

Associative Memory

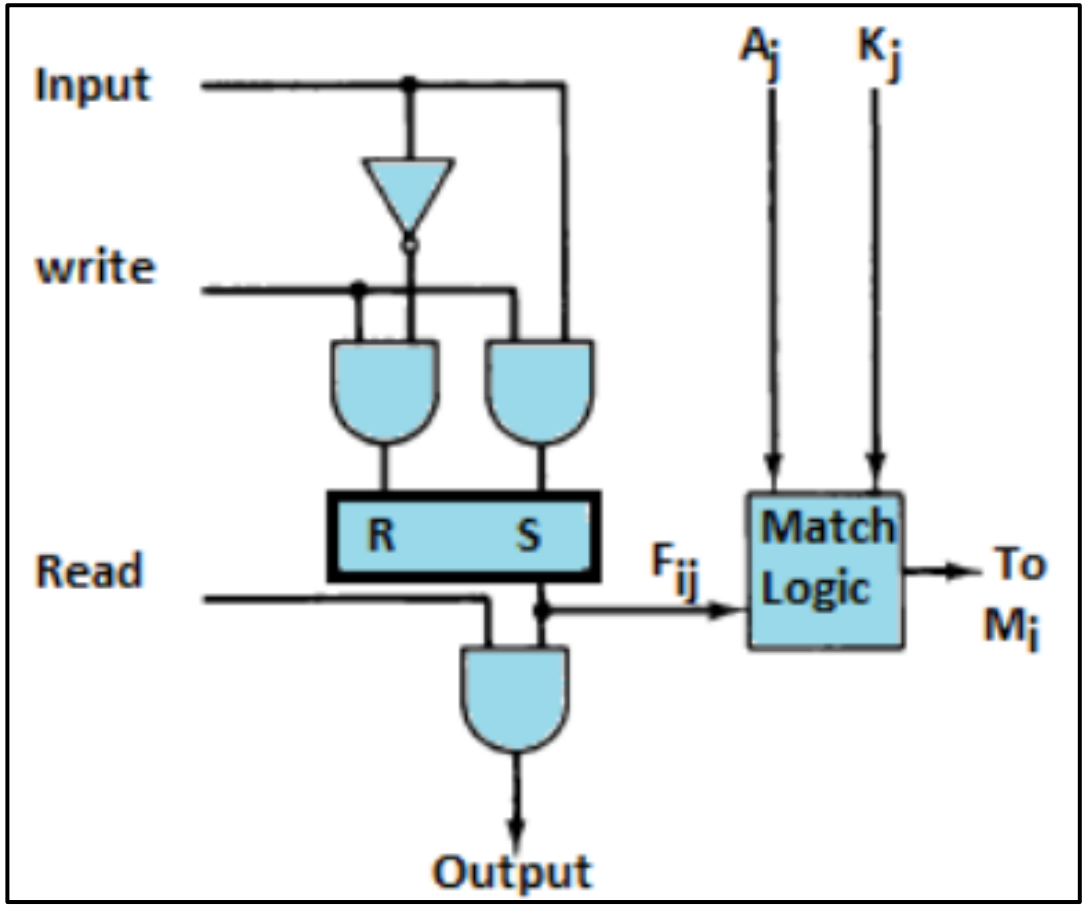
Associative memory of m word, n cells per word

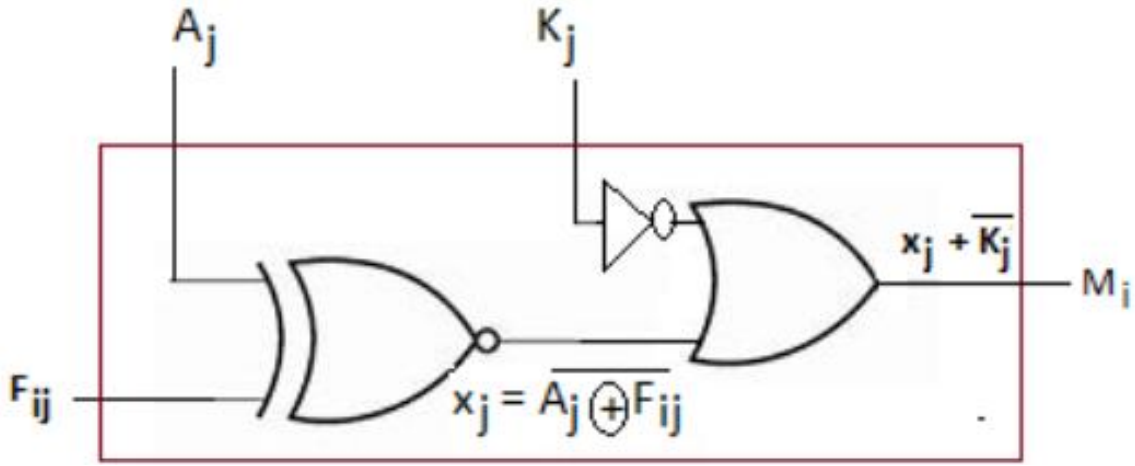
- The cells in the array are marked by the letter C with two subcripts. The first subscript gives the word number and the second specifies the bit position in the word.



Associative Memory

- One cell of associative memory





Associative Memory

- **Match Logic**
- First, we neglect the key bits and compare the argument in A with the bits stored in the cells of the words.
- Two bits are equal if they are both 1 or both 0. The equality of two bits can be expressed logically by the Boolean function:

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

where $x_j = 1$ if the pair of bits in position j are equal; otherwise, $x_j = 0$.

- For a word i to be equal to the argument in A we must have all x_j variables equal to 1. This is the condition for setting the corresponding match bit M, to 1. The Boolean function for this condition is

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

constitutes the AND operation of all pairs of matched bits in a word.

Associative Memory

- **Match Logic**
- We now include the key bit K_j in the comparison logic.
- The requirement is that if:

  $K_j = 0$, the corresponding bits of A_j and F_{ij} need no comparison

  $K_j = 1$, the corresponding bits of A_j and F_{ij} need comparison

- This is achieved by ORing each term:

$$x_j + K_j' = \begin{cases} x_j & \text{if } K_j = 1 \\ 1 & \text{if } K_j = 0 \end{cases}$$

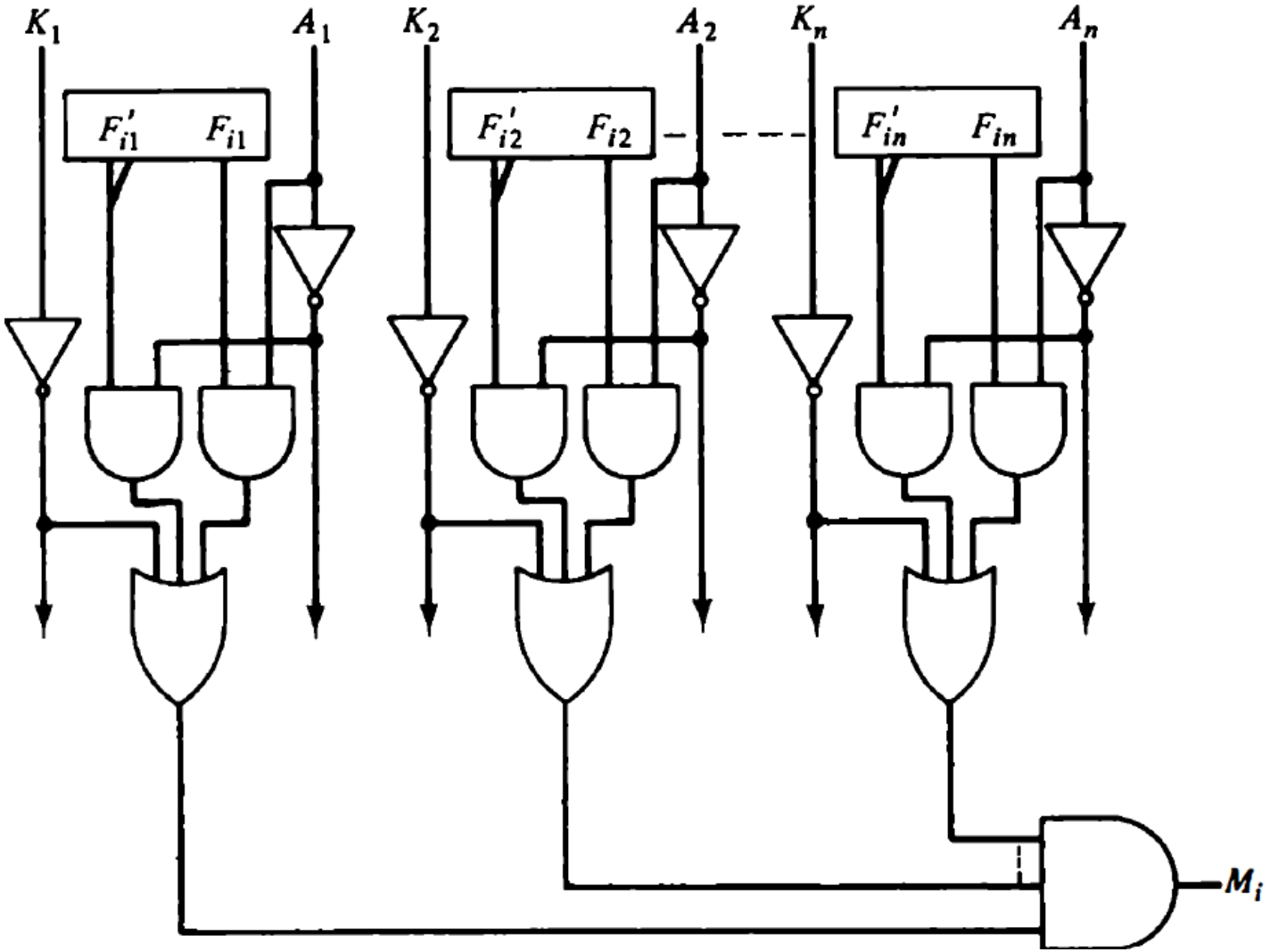
- The match logic for word ***i*** in an associative memory can now be expressed by the following Boolean function:

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

- I f we substitute the original definition o f x_j, the Boolean function above can be expressed as follows:

$$M_i \models \prod_{j=1}^{n} (A_j F_{ij} + A_j' F_{ij}' + K_j')$$

- Match logic for one word of associative memory



Auxiliary Memory

- The most common auxiliary memory devices used: ***magnetic disks and tapes***.
- Other components used, but not as frequently: ***magnetic drums, magnetic bubble memory, and optical disks***
- The important characteristics of any device are its ***access mode***, ***access time***, ***transfer rate***, ***capacity***, and ***cost***.
- **Access time:** The average time required to reach a storage location in memory and obtain its contents
- **Transfer rate:** the number of characters or words that the device can transfer per second
- In electromechanical devices with moving parts such as disks and tapes:

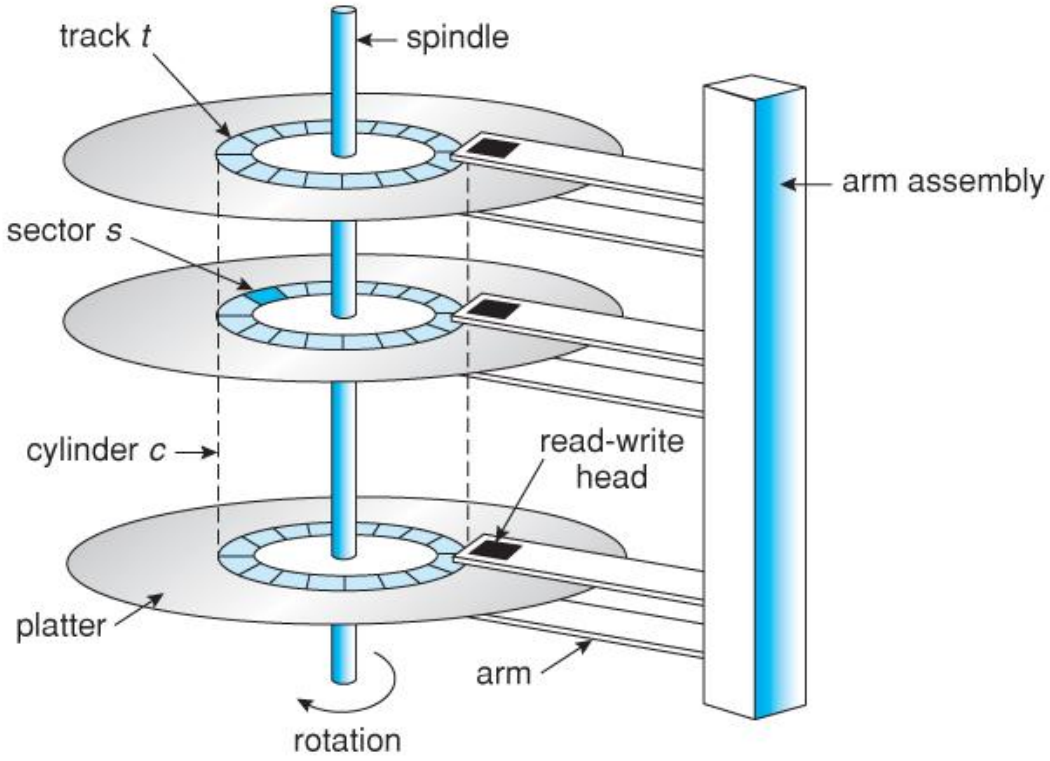
Access time = seek time + transfer time

Auxiliary Memory

- Magnetic drums and disks are quite similar in operation.
- Both consist of high-speed rotating surfaces coated with a magnetic recording medium.
- The rotating surface of the drum is a cylinder and that of the disk, a round flat plate.
- The recording surface rotates at uniform speed and is not started or stopped during access operations.
- Bits are recorded as magnetic spots on the surface as it passes a stationary mechanism called a **write head**.
- Stored bits are detected by a change in magnetic field produced by a recorded spot on the surface as it passes through a **read head**.
- The amount of surface available for recording in a disk is greater than in a drum of equal physical size.
- Disks have replaced drums in more recent computers

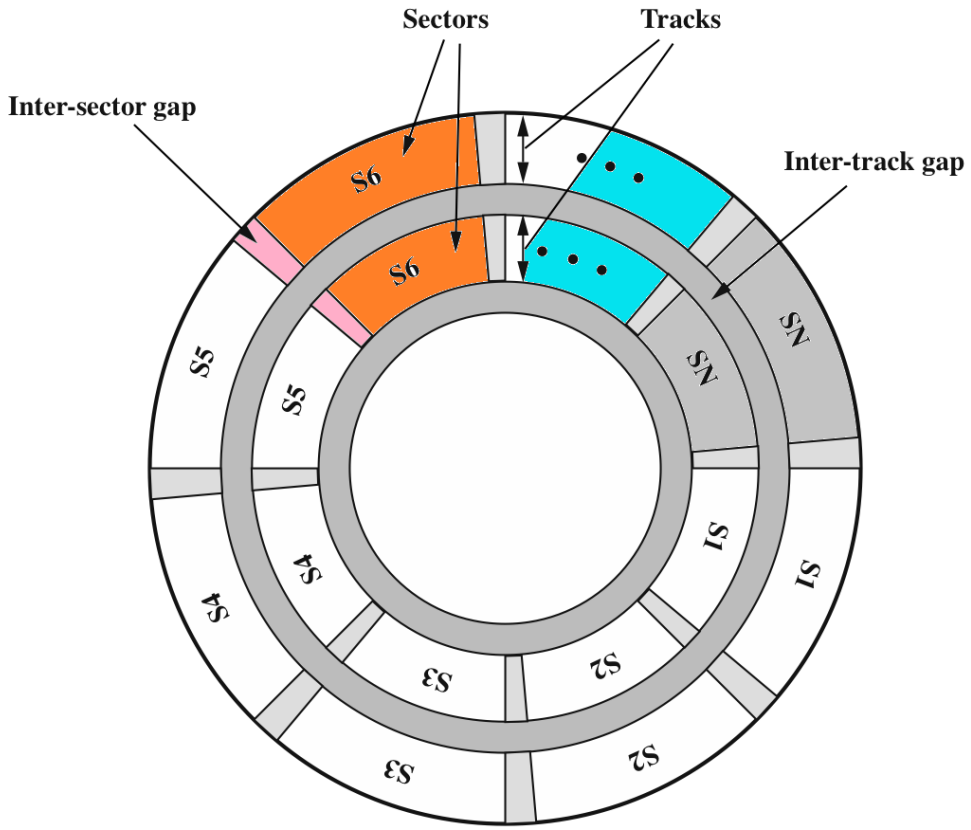
Auxiliary Memory - Magnetic Disks

- It is a circular plate constructed of metal or plastic coated with magnetized material.
- Often both sides of the disk are used
- Several disks may be stacked on one spindle with read/write heads available on each surface.
- All disks rotate together at high speed and are not stopped or started for access purposes.



Auxiliary Memory - Magnetic Disks

- Bits are stored in the magnetized surface in spots along concentric circles called **tracks**.
- The tracks are commonly divided into sections called **sectors**.
- In most systems, the minimum quantity of information which can be transferred is a sector.
- A disk system is addressed by address bits that specify the disk number, the disk surface, the sector number and the track within the sector.
- the track address bits are used by a mechanical assembly to move the head into the specified track position before reading or writing
- The address bits can then select a particular track electronically through a decoder circuit.



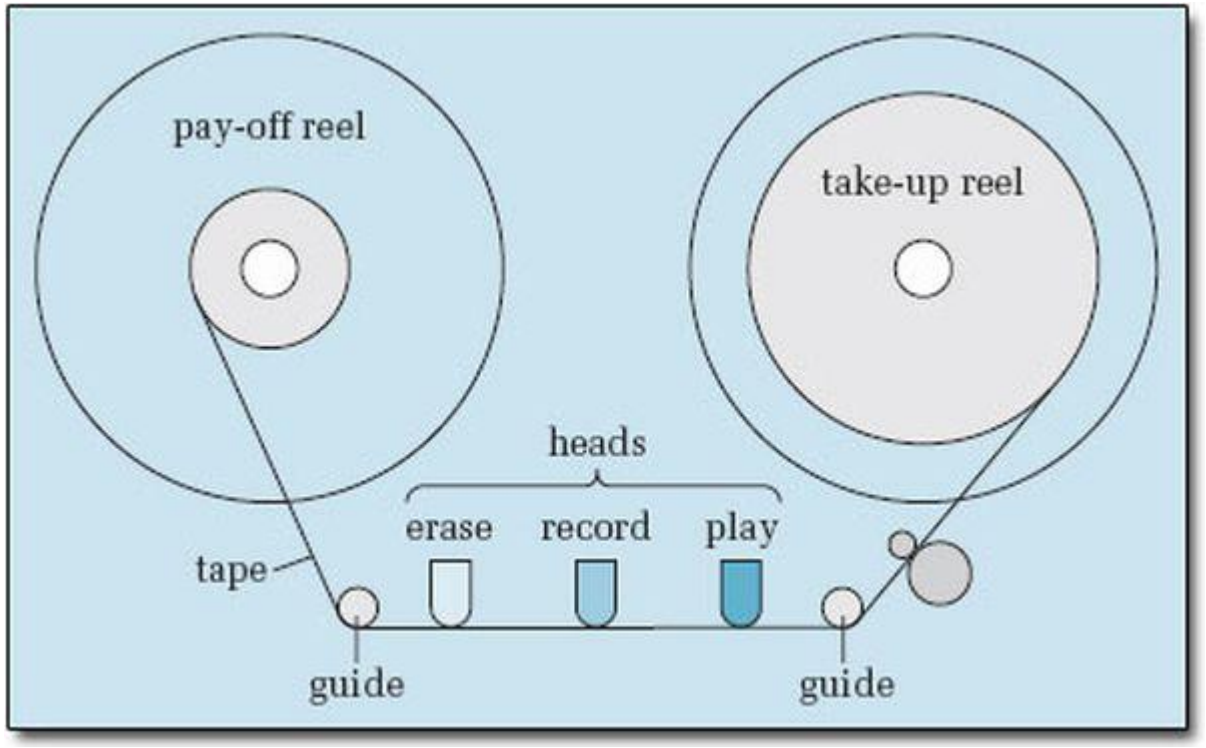
Auxiliary Memory - Magnetic Disks

- A track in a given sector near the circumference is longer than a track near the center of the disk
- If bits are recorded with equal density, some tracks will contain more recorded bits than others.
- To make all the records in a sector of equal length, some disks use a variable recording density with higher density on tracks near the center than on tracks near the circumference.
- ***Hard disks:*** Disks that are permanently attached to the unit assembly and cannot be removed
- ***Floppy disk:*** A disk drive with removable disks
  - Common sizes with diameters of 5.25 and 3. 5 inches.
  - Floppy disks are extensively used in personal computers as a medium for distributing software to computer users.

Auxiliary Memory - Magnetic Tape

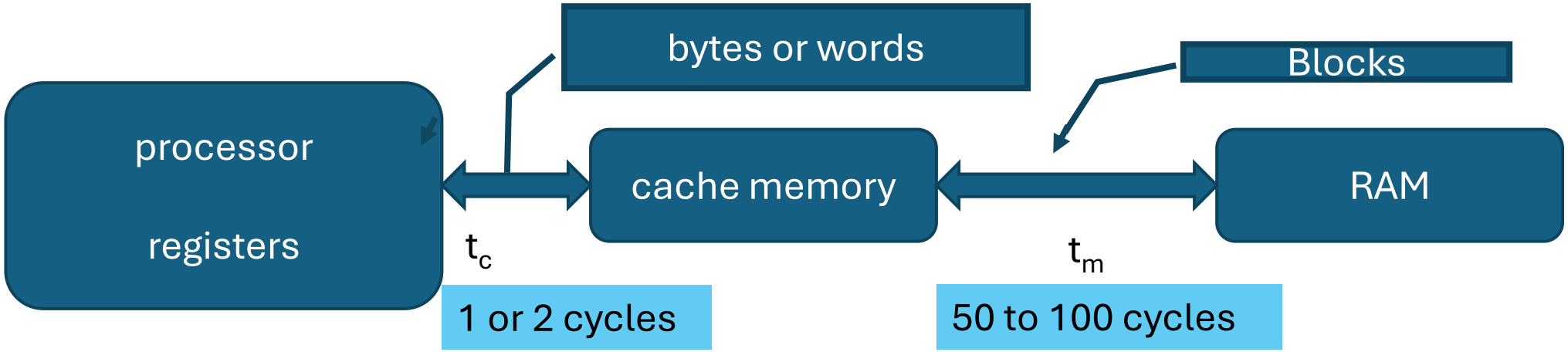
- The tape is a strip of plastic coated with a magnetic recording medium.
- Bits are recorded as magnetic spots on the tape along several tracks.
- Usually, seven or nine bits are recorded simultaneously to form a character together with a parity bit.
- Read/write heads are mounted one in each track so that data can be recorded and read as a sequence of characters
- Magnetic tape units can be stopped, started to move forward or in reverse, or can be rewound.
- They cannot be started or stopped fast enough between individual characters
- Information is recorded in blocks referred to as records.
- Gaps of unrecorded tape are inserted between records where the tape can be stopped.
- The tape starts moving while in a gap and attains its constant speed by the time it reaches the next record.
- Each record on tape has an identification bit pattern at the beginning and end.
- A tape unit is addressed by specifying the record number and the number of characters in the record.
- Records may be of fixed or variable length.

Auxiliary Memory - Magnetic Tape



Cache Memory

- Small amount of fast memory
- Sits between main memory (RAM) and CPU
- May be located on CPU chip or in system
- Objective is to make slower memory system look like fast memory.
- The data or contents that are used frequently by CPU are stored in the cache memory
- Processor can easily access that data in a shorter time.
- Whenever the CPU needs to access memory, it first checks the cache memory.
- If the data is not found in cache memory, then the CPU moves into the main memory.



Cache Memory

Cache operations

- When the CPU needs to access memory, the cache is examined. If the word is found in the cache, it is read from the fast memory.
- If the word addressed by the CPU is not found in the cache, the main memory is accessed to read the word.
- A block of words once just accessed is then transferred from main memory to cache memory. The block size may vary from one word (the one just accessed) to about 16 words adjacent to the one just accessed.
- Performance of the cache memory is measured in terms of a quantity called ***hit ratio***.
- When the CPU refers to memory and finds the word in cache, it is said to produce a ***hit***.
- If the word is not found in the cache, it is in main memory and it counts as a ***miss***.

The ratio of the number of hits divided by the total CPU references to memory (hits plus misses) is the hit ratio.

Cache Memory

Cache operations

Hit ratio (h) = Total no. of Hits / Total no. of attempts

Efficiency is maximum when h = 1

But h = 0.9 is common (next example)

Let, t_C → cache-access time

h → hit ratio

t_m → main memory access time

The avg. access time $(t_{avg}) = h\, t_C + (1 - h)\,(t_C + t_m)$

$$\text{Efficiency} = t_C / t_{avg}$$

$$\text{Efficiency} = 1 / [\,1 + \Upsilon\ (1 - h)\,] \quad \text{where } \Upsilon = t_m / t_C$$

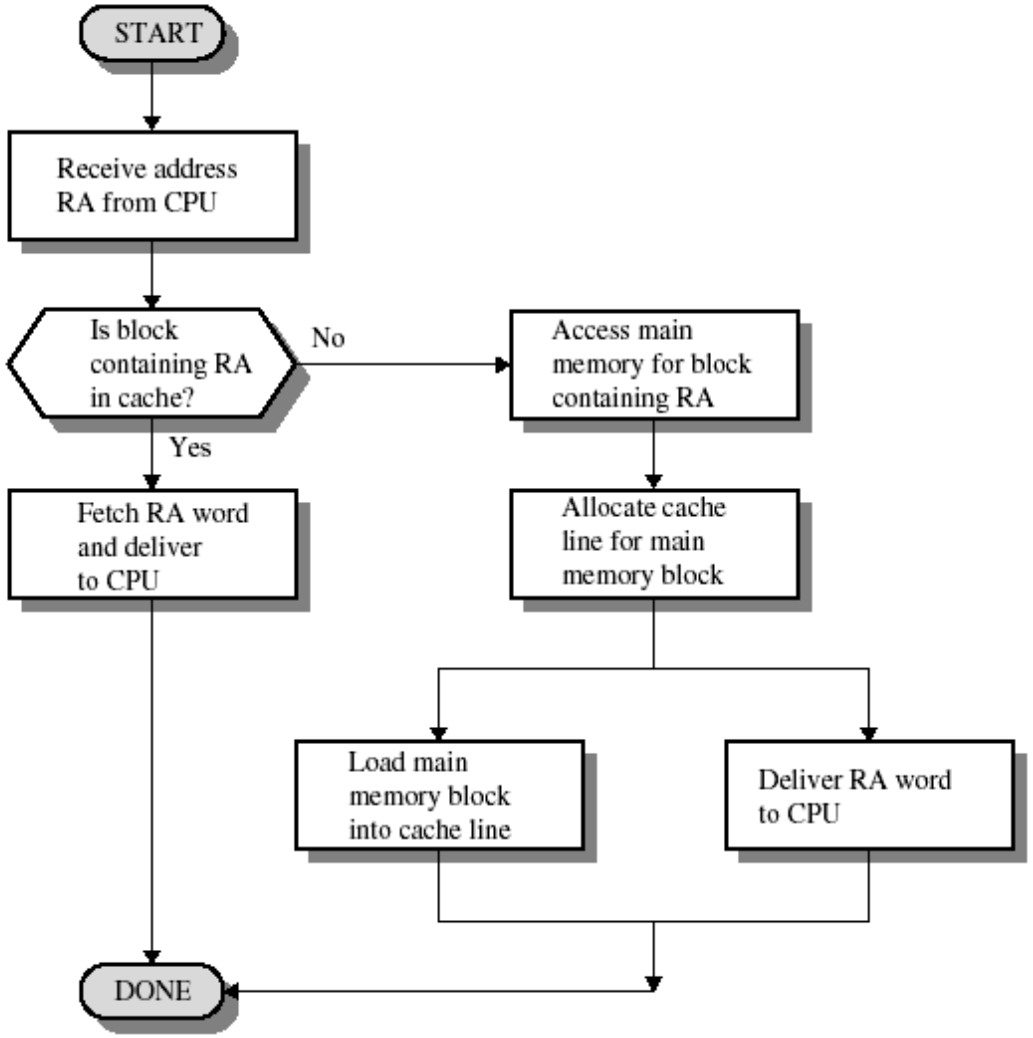
Access time (t_A): Avg. time taken to read a unit of information from the memory.

$$\text{Access rate } (r_A) = 1 / t_A$$

Cycle time (T_C): Avg. time lapse b/w two successive read operations.

$$\text{Data transfer rate (or BW), } r_C = 1/T_C$$

Cache operations



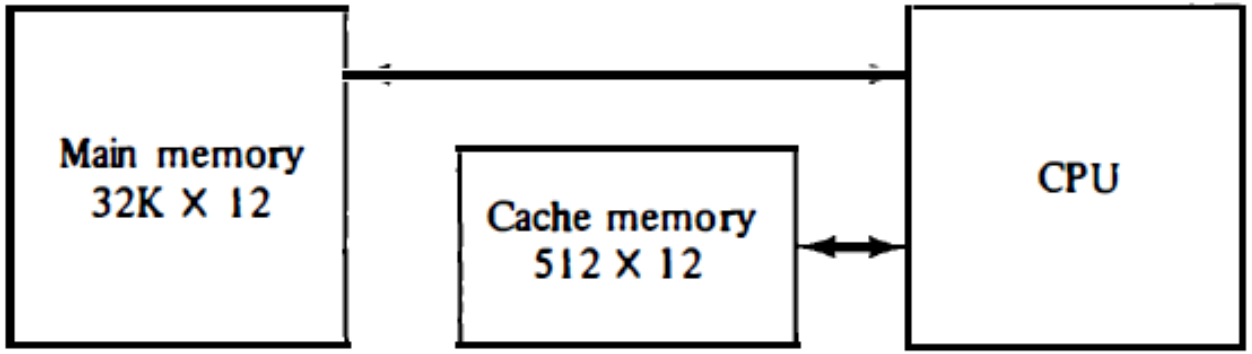
- The average memory access time of a computer system can be improved considerably by use of a cache.
- If the hit ratio is high enough so that most of the time the CPU accesses the cache instead of main memory,
- the average
- access time is closer to the access time of the fast cache memory.

Cache Memory

- The basic characteristic of cache memory is its fast access time.
- very little or no time must be wasted when searching for words in the cache.
- **Mapping process:** It is a process of transformation of data from main memory to cache memory
- Three types of mapping procedures
  1. Associative mapping
  2. Direct mapping
  3. Set-associative mapping

Cache Memory

- **Example of cache memory.**



For every word stored in cache, there is a duplicate copy in main memory.

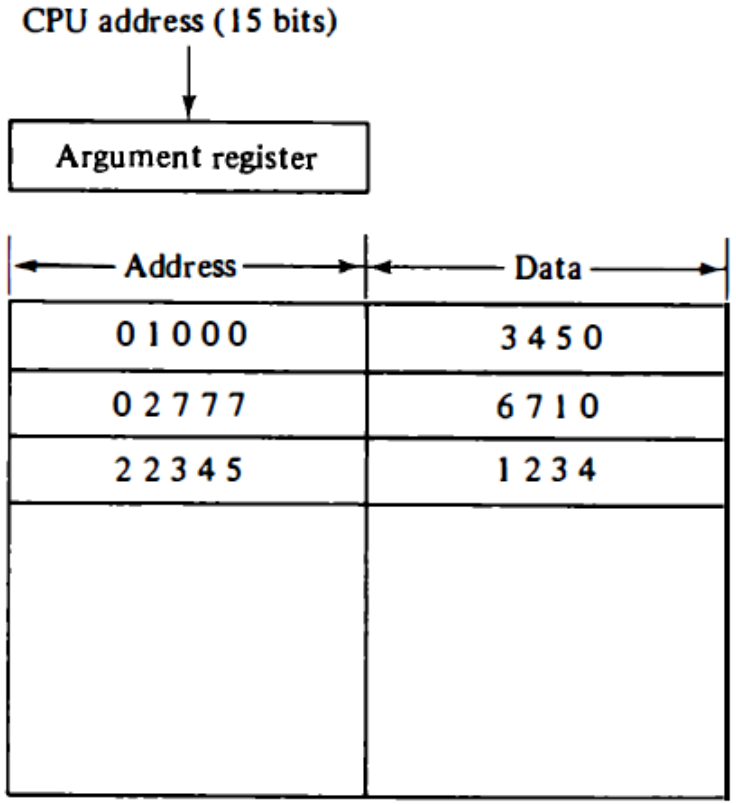
The CPU communicates with both memories.

1. It first sends a 15-bit address to cache.
2. If there is a hit, the CPU accepts the 12-bit data from cache.
3. If there is a miss, the CPU reads the word from main memory and the word is then transferred to cache.

Cache Memory

Associative Mapping

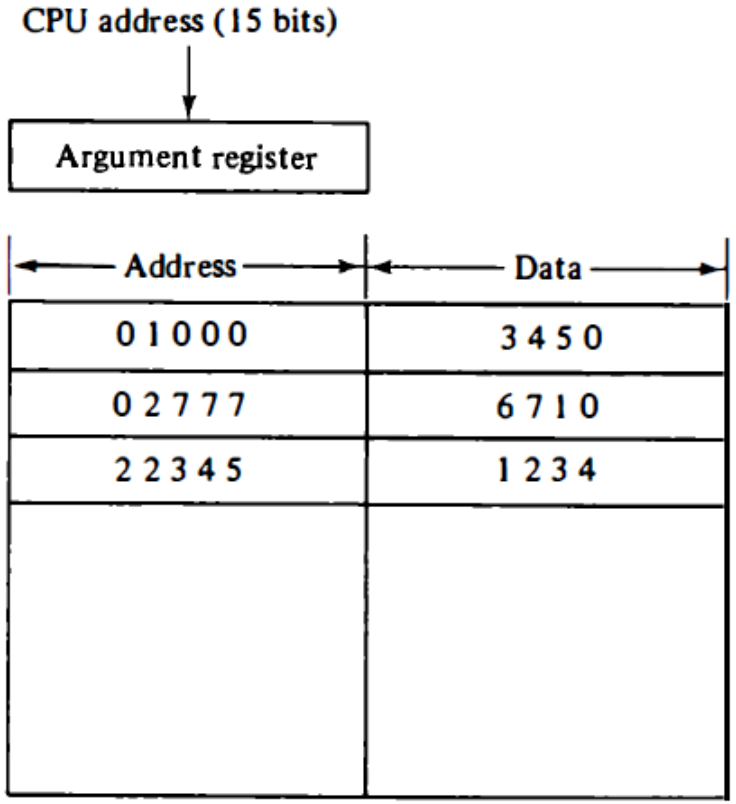
- The fastest and most flexible cache organization uses an associative memory.
- Associative memory stores both the **address and content (data) of the memory word**.
- **Example:**
- In the example Address and data word are represented in octal system
- address value of 15 bits is shown as a five-digit octal number and its corresponding 12 -bit word is shown as a four-digit octal number.
- A CPU address of 15 bits is placed in the argument register.
- *If the address is found, the corresponding 12-bit data is read and sent to the CPU.*
- *If no match occurs, the main memory is accessed for the word.*
- The address--data pair is then transferred to the associative cache memory.



Cache Memory

Associative Mapping

- The fastest and most flexible cache organization uses an associative memory.
- Associative memory stores both the **address and content (data) of the memory word**.
- **Example:**
- In the example Address and data word are represented in octal system
- address value of 15 bits is shown as a five-digit octal number and its corresponding 12 -bit word is shown as a four-digit octal number.
- A CPU address of 15 bits is placed in the argument register.
- *If the address is found, the corresponding 12-bit data is read and sent to the CPU.*
- *If no match occurs, the main memory is accessed for the word.*
- The address--data pair is then transferred to the associative cache memory.



Cache Memory