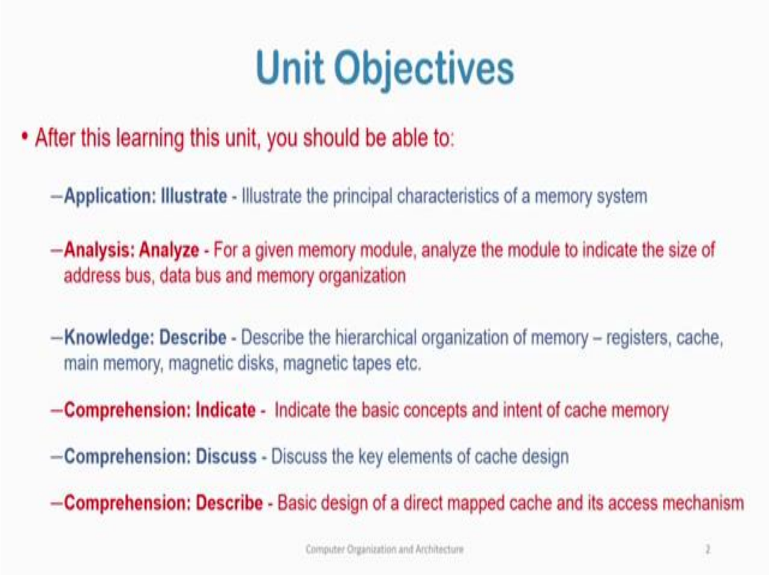
# Computer Organization and Architecture: A Pedagogical Aspect

Prof. Jatindra Kr. Deka
Dr. Santosh Biswas
Dr. Arnab Sarkar
Department of Computer Science & Engineering
Indian Institute of Technology, Guwahati

## Memory System
## Lecture – 24
## Basics of Memory and Cache

In this lecture we begin our discussion with the module Memory System. The first unit of this module deals with the Basics of Memory and Cache.

(Refer Slide Time: 00:38)



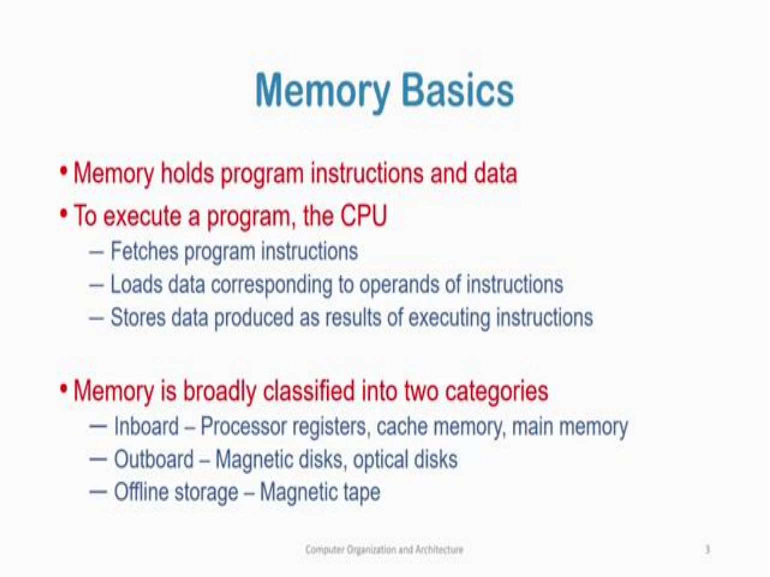
### Unit Objectives

- After this learning this unit, you should be able to:
  - –**Application: Illustrate** - Illustrate the principal characteristics of a memory system
  - –**Analysis: Analyze** - For a given memory module, analyze the module to indicate the size of address bus, data bus and memory organization
  - –**Knowledge: Describe** - Describe the hierarchical organization of memory – registers, cache, main memory, magnetic disks, magnetic tapes etc.
  - –**Comprehension: Indicate** - Indicate the basic concepts and intent of cache memory
  - –**Comprehension: Discuss** - Discuss the key elements of cache design
  - –**Comprehension: Describe** - Basic design of a direct mapped cache and its access mechanism

Computer Organization and Architecture 2

After learning this unit you should be able to illustrate the principle characteristics of a memory system. You should be able to analyze the module a memory module to indicate the size of the address bus, the data bus and also the memory organization basically. You should be able to describe the hierarchical organization of memory composed of registers, cache, main memory, magnetic disks, magnetic tapes etcetera. You should be able to indicate the basic concepts and the intent of behind the usage of cache memory. You should be able to discuss the key elements of any cache design. And you should be able to describe the basic design of a direct mapped cache and its basic access mechanism.

(Refer Slide Time: 01:32)



## Memory Basics

- Memory holds program instructions and data
- To execute a program, the CPU
  - Fetches program instructions
  - Loads data corresponding to operands of instructions
  - Stores data produced as results of executing instructions
- Memory is broadly classified into two categories
  - Inboard – Processor registers, cache memory, main memory
  - Outboard – Magnetic disks, optical disks
  - Offline storage – Magnetic tape

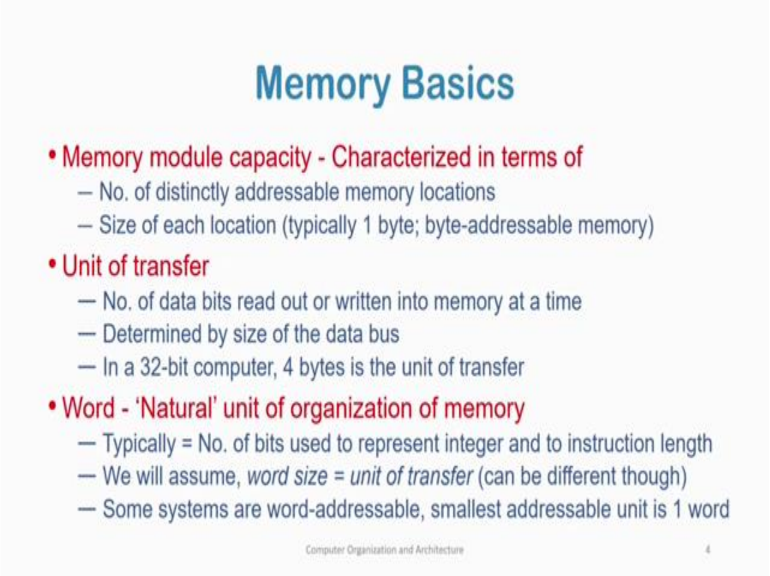
Computer Organization and Architecture 3

Memory basics: So, what is memory? Memory is that unit in the computer which holds program instructions and data. To execute a program the CPU fetches the program instructions from memory, it also loads the data corresponding to the operands of these instructions from the memory. After the execution of the instructions it stores the data produced after executing that instruction also into the memory.

Memory is broadly classified into two categories: Inboard memory and Outboard memory. Also, we also have offline storage which are basically bulk storage devices. So, what is inboard memory? Inboard memory are those memory units which are directly plugged into the motherboard of the computer. So, we have processor registers, cache memory, main memory; those are within on the motherboard of the computer itself, either on the processor or on the motherboard. Output memory on the other hand like magnetic disks or hard disks, optical disks etcetera are outboard memory, are not on the motherboard, are not plugged on the motherboard on the of the computer.

Offline storage as I said are bulk storage devices like magnetic tapes.

(Refer Slide Time: 03:00)



## Memory Basics

- Memory module capacity - Characterized in terms of
  - No. of distinctly addressable memory locations
  - Size of each location (typically 1 byte; byte-addressable memory)
- Unit of transfer
  - No. of data bits read out or written into memory at a time
  - Determined by size of the data bus
  - In a 32-bit computer, 4 bytes is the unit of transfer
- Word - 'Natural' unit of organization of memory
  - Typically = No. of bits used to represent integer and to instruction length
  - We will assume, *word size = unit of transfer* (can be different though)
  - Some systems are word-addressable, smallest addressable unit is 1 word

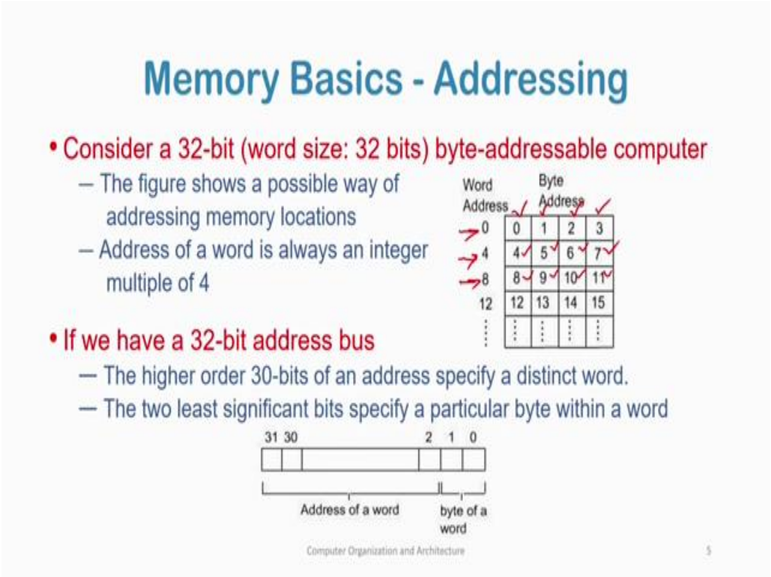
Computer Organization and Architecture 4

Now we will go on through a few basic definitions and terminologies which characterize memory. The first of them is capacity of a memory module. The capacity of a memory module is characterized in terms of the number of distinctly addressable memory locations and also the size of each of those locations. So, typically of the typically the size of a memory location is 1 byte for byte addressable memory. Although, there could be something called word addressable memory where I can address in higher more than a word, more than a byte; meaning that suppose if a word consists of 4 bytes, it and if it is word addressable I won't be able to address and find out each byte within a word. But I will be able to access the byte the words themselves.

Unit of transfer: The number of data bits read out or written into the memory at a given time is called the unit of transfer. Unit of transfer is basically determined by the size of the data bus in the computer. For a 32 bit computer, 4 bytes is the unit of transfer. A word is defined as the natural unit of organization of memory. So, this is the unit with which the processor basically executes its works. This is the unit with which the processor basically works. This is that this is the typical number of bits used to represent integers within the processor and also the average instruction length.

We will assume in this course that word size is equal to the unit of transfer. However, in general word size can be different from the unit of transfer, but as I said in this course for us word size equals to unit of transfer. Some computer systems are word addressable as I said at the beginning they are not byte addressable and therefore, the basic addressable unit in these computers is 1 word. So, if 1 word is 4 bytes the basic addressable units are in multiples of 4 bytes.

(Refer Slide Time: 05:26)



## Memory Basics - Addressing

- Consider a 32-bit (word size: 32 bits) byte-addressable computer
  - The figure shows a possible way of addressing memory locations
  - Address of a word is always an integer multiple of 4

| Word Address | Byte Address | | | |
|---|---|---|---|---|
| 0 | 0 | 1 | 2 | 3 |
| 4 | 4 | 5 | 6 | 7 |
| 8 | 8 | 9 | 10 | 11 |
| 12 | 12 | 13 | 14 | 15 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |

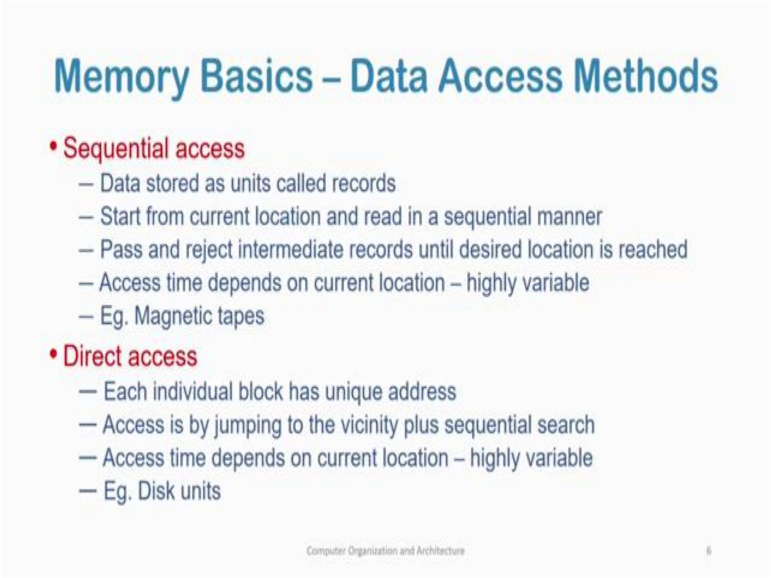
- If we have a 32-bit address bus
  - The higher order 30-bits of an address specify a distinct word.
  - The two least significant bits specify a particular byte within a word

Computer Organization and Architecture 5

Memory addressing: Consider a 32 bit memory. So, word size is 32 bits in my computer. Unit of transfer equals to word size equals to 32 bits and this computer is byte addressable. So, I can uniquely identify each byte in the memory. The figure shows a possible way of addressing memory locations. Address of a word is always an integer multiple of 4. So, therefore, in this the byte addresses for word 0, the byte addresses for word 0 are 0, 1, 2, 3 ok. For the for word number 1, the byte addresses are 4, 5, 6, 7. For word number 2, the byte addresses are 8, 9, 10, 11 and so on ok.

So, if we have a 32 bit address bus, if we have a 32 bit address bus, the high the higher order 30 bits of an address will specify a distinct word. Why? Because we have 4 bytes per word so 2 bits are necessary to identify a byte within a word and the higher order 30 bits will identify a given word within the memory. The 2 least significant bits specify a particular byte within a word.

(Refer Slide Time: 07:00)



## Memory Basics – Data Access Methods

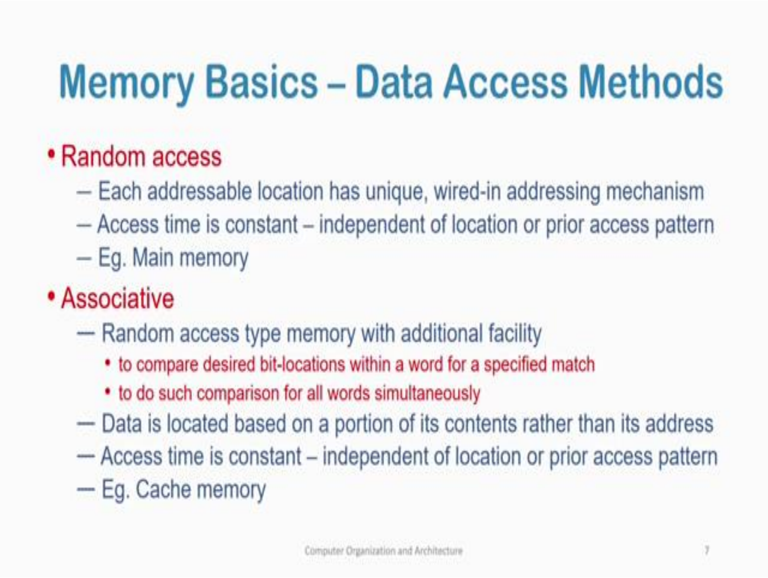
- Sequential access
  - Data stored as units called records
  - Start from current location and read in a sequential manner
  - Pass and reject intermediate records until desired location is reached
  - Access time depends on current location – highly variable
  - Eg. Magnetic tapes
- Direct access
  - Each individual block has unique address
  - Access is by jumping to the vicinity plus sequential search
  - Access time depends on current location – highly variable
  - Eg. Disk units

Computer Organization and Architecture 6

Memory data access methods. So, we have different types of memories and in these different types of memories there are various access methods. For example, we have sequential access memories in which data are stored as units called records and data are referenced in terms of its current location. So, there is a read-write head which is which is on which is typically on the last location from which data was read or written to. So, we start from this current location and read in a sequential manner. We pass over and reject intermediate records until we reach the desired location.

So, access time depends on the current location and is highly variable. An example of this type of memory is magnetic tapes. Second, we also have direct access type of memories, hard disks that is magnetic disks are direct access types of memories. Each individual block has an unique address, access is by jumping to the vicinity of the block and then by doing a sequential search. Access time again depends on the current location and is highly variable.

(Refer Slide Time: 08:20)



## Memory Basics – Data Access Methods

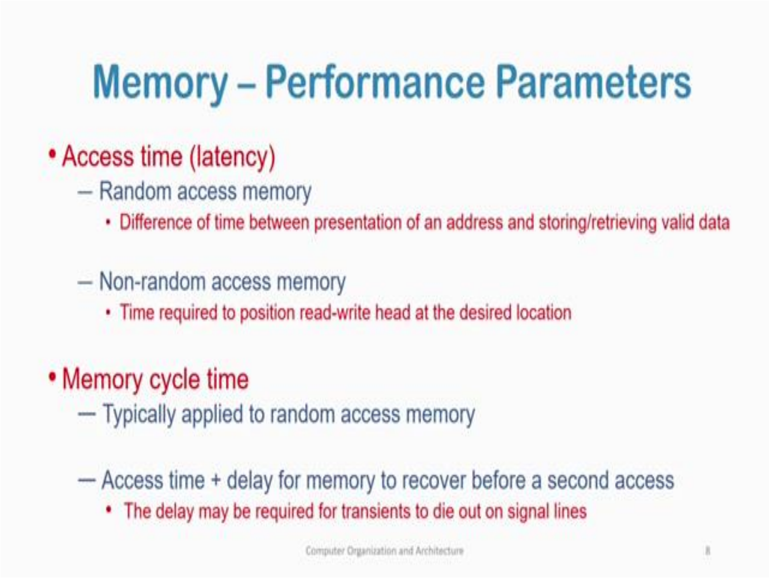
- Random access
  - – Each addressable location has unique, wired-in addressing mechanism
  - – Access time is constant – independent of location or prior access pattern
  - – Eg. Main memory
- Associative
  - — Random access type memory with additional facility
    - to compare desired bit-locations within a word for a specified match
    - to do such comparison for all words simultaneously
  - — Data is located based on a portion of its contents rather than its address
  - — Access time is constant – independent of location or prior access pattern
  - — Eg. Cache memory

Computer Organization and Architecture 7

The third access type is random access type of memory. In random access memories each addressable location has a unique wired-in addressing mechanism. And the access time is constant: Independent of the location or prior access pattern. Therefore, it is different from sequential access and direct access type of memories, in which the access time was dependent on the current position of the read-write head. So, here the time to access a given byte is independent of which byte I am addressing and which location I have currently accessed. The fourth type of memory is associative memories. Associative memories are basically random access type of memories in which there is an additional facility.

We can compare for a specific match of desired bit locations within the word and the memory allows to do this match of desired bit locations within the within a word for all words in this memory. Data is located; data is located or identified based on a portion of the contents rather than the address. That is along with the address of the memory location. I also use a part of the contents of a word to find out whether that the desired word is present in the memory. As because this is random access, access time is constant and is independent of the location or prior access pattern. An example of this type of memory is cache memory.

(Refer Slide Time: 10:08)



## Memory – Performance Parameters

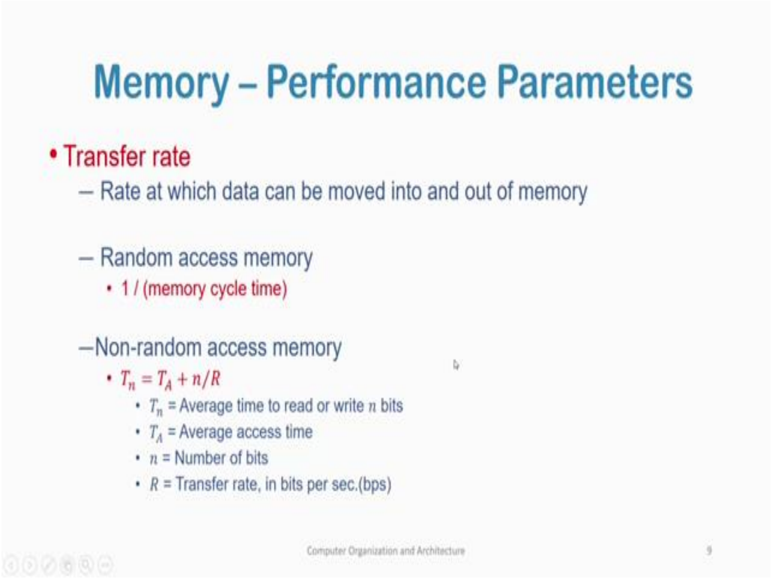
- Access time (latency)
  - Random access memory
    - Difference of time between presentation of an address and storing/retrieving valid data
  - Non-random access memory
    - Time required to position read-write head at the desired location
- Memory cycle time
  - Typically applied to random access memory
  - Access time + delay for memory to recover before a second access
    - The delay may be required for transients to die out on signal lines

Computer Organization and Architecture 8

The memory is characterized also based on different performance parameters. The first important performance parameter is access time or access latency. For random access memory access time is defined as the difference of time between the presentation of an address on the memory address register from which it goes to the address bus and the storing of data on the memory data register from the data bus. For non-random access memory this access time or access latency is the time required to position the read-write head, at the desired bit location and to read the first bit up to the time the first bit is read.

So, this is the access latency for non-random access memories. Memory cycle time is typically applied to random access type of memory and is defined as the access time plus delay for memory to recover before a second access is made. This delay may be required for transient signals on the buses to die down before the next access can be made.

(Refer Slide Time: 11:28)



## Memory – Performance Parameters

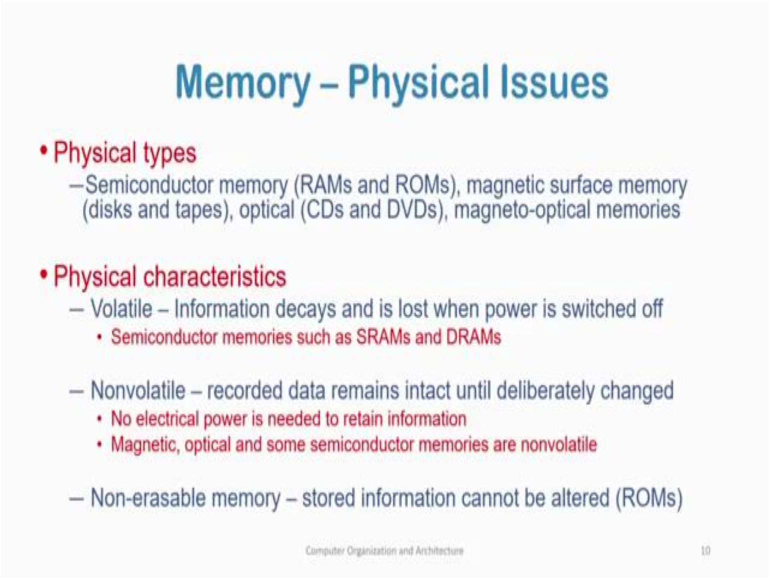
- Transfer rate
  - – Rate at which data can be moved into and out of memory
  - – Random access memory
    - 1 / (memory cycle time)
  - –Non-random access memory
    - $T_n = T_A + n/R$
      - $T_n$ = Average time to read or write $n$ bits
      - $T_A$ = Average access time
      - $n$ = Number of bits
      - $R$ = Transfer rate, in bits per sec.(bps)

Computer Organization and Architecture 9

The third performance parameter is transfer rate; the rate at which data can be moved into and out of memory. For random access memories this is defined as 1 by memory cycle time. For non-random access memories, let $T_n$ denote the average time to read or write $n$ bits. Then and $T_A$ is the average access time, $n$ is the number of bits and R is a transfer rate in bits per second. Then access time or sorry the transfer rate is defined as $T_n = T_A + n/ R$.

So, here for example, in sequential access memory as we know that the time to access a given memory word is dependent on the current position of the read-write head and is therefore, variable. So, over a given set of accesses we first find out what is the average time required to access any record between the memory; that is $T_A$ or the average access time. Now to transfer n bits of from the memory, we first need to access the first bit and then one by one transfer the next n bits. So, we have $T_A + n/ R$, where R is the transfer rate in bits per second.

(Refer Slide Time: 12:57)



## Memory – Physical Issues

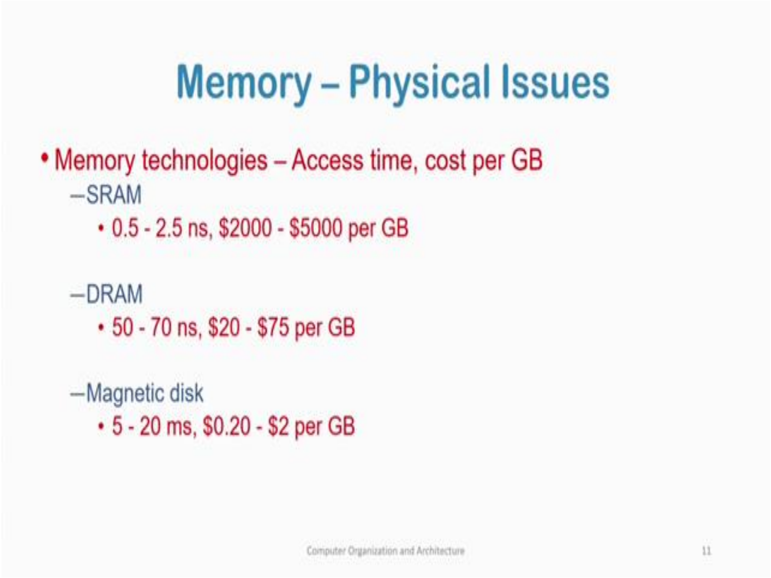
- Physical types
  - –Semiconductor memory (RAMs and ROMs), magnetic surface memory (disks and tapes), optical (CDs and DVDs), magneto-optical memories
- Physical characteristics
  - – Volatile – Information decays and is lost when power is switched off
    - Semiconductor memories such as SRAMs and DRAMs
  - – Nonvolatile – recorded data remains intact until deliberately changed
    - No electrical power is needed to retain information
    - Magnetic, optical and some semiconductor memories are nonvolatile
  - – Non-erasable memory – stored information cannot be altered (ROMs)

Computer Organization and Architecture 10

Physical issues of memory: So, what are the physical types of memories that are available? We have semiconductor memories that is RAMs and ROMs, we have magnetic surface memories, magnetic disks and magnetic tapes, we have optical memories CDs and DVDs, we also have magneto-optical type of disks. Physical characteristics: memory can be volatile, in which memory information decays and is lost when power is switched off. For example, in case of semiconductor memories such as SRAMs and in DRAMs they are volatile. The information is lost when power is switched off. We also have nonvolatile memories, in which recorded information remains intact until deliberately changed.

So, in these in these nonvolatile memories no electrical power is needed to retain information. Magnetic, optical and some semiconductor memories are non-volatile memories. Non erasable memory stored information cannot be altered for example, ROMs or read only memories. Obviously, non-erasable memories are also nonvolatile.

(Refer Slide Time: 14:10)



## Memory – Physical Issues

- Memory technologies – Access time, cost per GB
  - –SRAM
    - 0.5 - 2.5 ns, $2000 - $5000 per GB
  - –DRAM
    - 50 - 70 ns, $20 - $75 per GB
  - –Magnetic disk
    - 5 - 20 ms, $0.20 - $2 per GB

Computer Organization and Architecture 11

Now, we talk of different memory tech, when we talk of different memory technologies. Two important characteristics become very important. What is its access time and what is its cost per GB? For example, in a SRAM type SRAM type of memories, the access time is about 0.5 to 2.5 nanoseconds.

However, the cost per GB is about 2000 to 5000 dollars. So, we can understand that for SRAM, they are very fast. For typical processors today it will be about or at least one-tenth, it will run at least one-tenth the speed of the processor. However, the cost as we see is also very high. For DRAMs, the access time is of the order of 50 to 70 nanoseconds; that means, it is about 50 to 70 times. It is it is about 50 to 100 times slower than the SRAM memory units.

However, the cost per GB is also about 80 to 90 times lower. So, the cost per GB for DRAMs is about 20 dollars to 75 dollars per GB. In for magnetic disks or hard disks the access time is thousands of times slower than the processor. So, it is about 5 to 20 milliseconds; that means it is tens of thousands of times slower than the processor speed. DRAMs are hundreds of times slower than the processor speed. Magnetic discs are tens of thousands of times slower than the processor speed. However, the cost per GB is also very very low.

With this discussion we complete part 1 of unit 1.