COMPUTER ORGANIZATION AND ARCHITECTURE
BEC306C

MODULE-4

Memory System: Basic Concepts, Semiconductor RAM Memories-Internal organization of memory chips, Static memories, Asynchronous DRAMS, Read Only Memories, Cash Memories, Virtual Memories, Secondary Storage, Magnetic Hard Disks. **(5.1, 5.2, 5.2.1, 5.2.2, 5.2.3, 5.3, 5.5(except 5.5.1 to 5.5.4), 5.7 (except5.7.1), 5.9, 5.9.1 of Chap 5 of Text).**

BASIC CONCEPTS

The maximum size of the memory that can be used in any computer is determined by the addressing scheme. For example, a computer that generates 16-bit addresses is capable of addressing up to 2^{16} = 64K memory locations. Similarly, machines with 32-bit addresses can utilize a memory up to 2^{32} = 4G locations. The number of locations represents the size of the address space of the computer.

Most modern computers are byte-addressable. The memory is usually designed to store and retrieve data in word-length quantities. Data transfer between the memory and the processor takes place through the use of two processor registers, usually called MAR (memory address register) and MDR (memory data register). If MAR is k bits long and MDR is n bits long, then the memory unit may contain up to 2^k addressable locations. During a memory cycle, n bits of data are transferred between the memory and the processor. This transfer takes place over the processor bus, which has k address lines and n data lines. The bus also includes the control lines Read/Write' (R/W') and Memory Function Completed (MFC) for coordinating data transfers. The connection between the processor and the memory is shown in Figure 5.1.

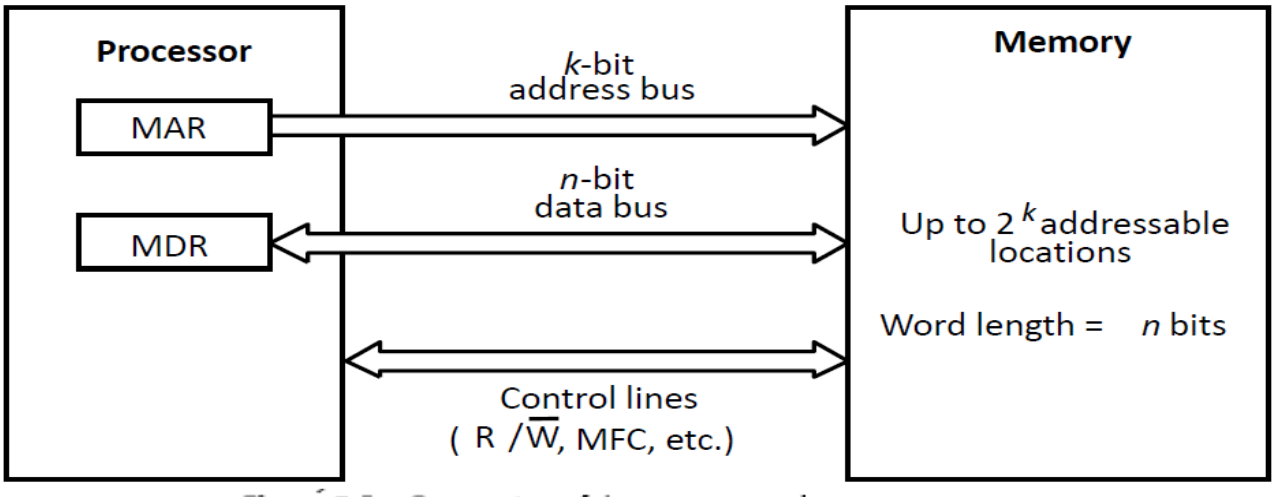


Figure 5.1 Connection of the memory to the processor.

The processor reads data from the memory by loading the address of the required memory location into the MAR register. The R/W' line is set to 1. The memory responds by placing the data from the addressed location onto the data lines, and confirms this action by asserting the MFC signal. Upon receipt of the MFC signal, the processor loads the data on the data lines into the MDR register.

The processor writes data into a memory location by loading the address of this location into MAR and loading the data into MDR. The R/W' line is set to 0. If read or write operations involve consecutive address locations in the main memory, then a "block transfer" operation can be performed.

Memory accesses may be synchronized using a clock, or they may be controlled using special signals that control transfers on the bus.

Memory access time: Time that elapses between the initiation of an operation and the completion of that operation.

Memory cycle time: Minimum time delay required between the initiations of two successive memory operations.

Random Access Memory (RAM): A memory unit in which any location can be accessed for a Read or Write operation in some fixed amount of time that is independent of the location's address.

Cache Memory: Is a small, fast memory that is inserted between the larger, slower main memory and the processor. It holds the currently active segments of a program and their data. It reduces memory access time.

Virtual memory: Data is stored in physical memory locations that have addresses different from those specified by the program. An address generated by the processor is referred to as a *virtual or logical address*. The virtual address space is mapped onto the physical memory where data are actually stored. The mapping function is implemented by a special memory control circuit, often called the *memory management unit*. Virtual memory is used to increase the apparent size of the physical memory.

SEMI CONDUCTOR RAM MEMORIES

Internal Organization of Memory Chips: Each memory cell can hold one bit of information. Memory cells are organized in the form of an array. One row is one memory word. All cells of a row are connected to a common line, known as the *word line*. Word line is connected to the address decoder. The cells in each column are

connected to a Sense/Write circuit by two *bit lines*. The Sense/Write circuits are connected to the data input/output lines of the chip.

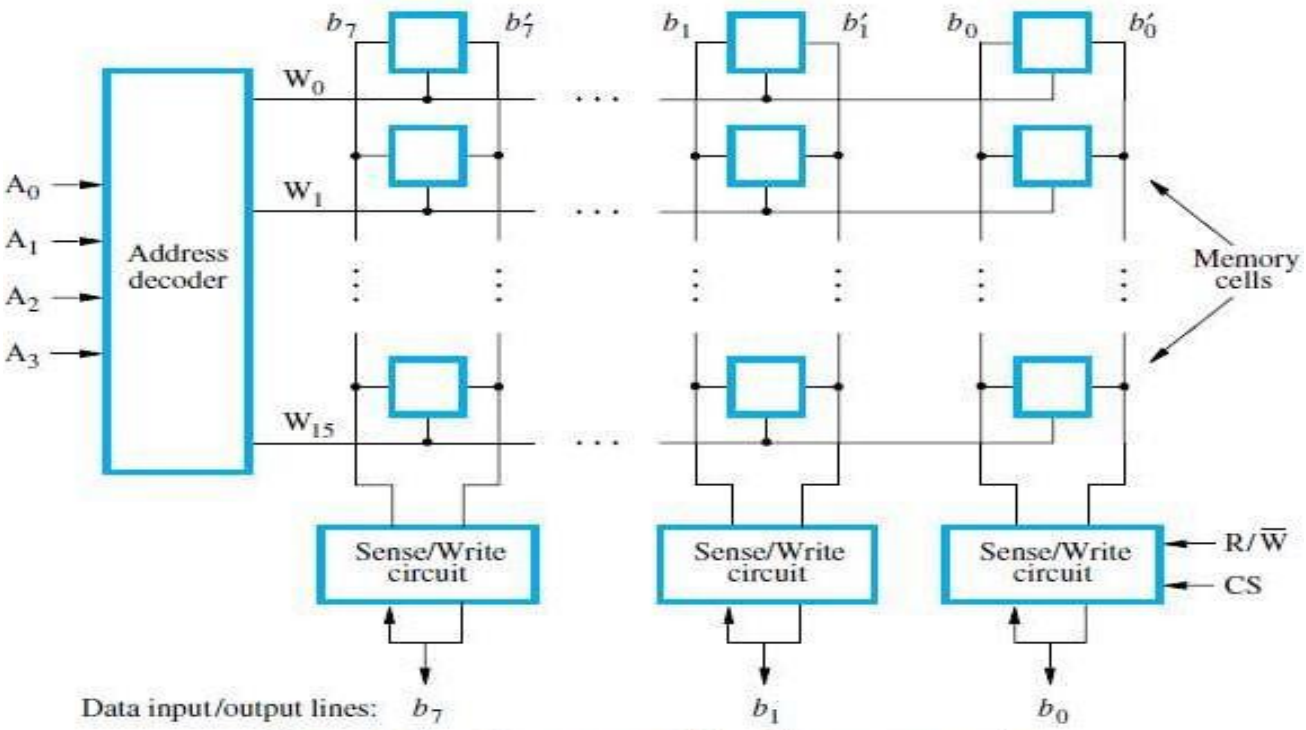


Figure 8.2 Organization of bit cells in a memory chip.

Figure 8.2 is an example of a very small memory circuit consisting of 16 words of 8 bits each. This is referred to as a 16 × 8 organization. The data input and the data output of each Sense/Write circuit are connected to a single bidirectional data line that can be connected to the data lines of a computer. Two control lines, R/W' and CS, are provided. The R/W' (Read/Write') input specifies the required operation, and the CS (Chip Select) input selects a given chip in a multichip memory system.

The memory circuit in Figure 8.2 stores 128 bits and requires 14 external connections for address, data, and control lines. It also needs two lines for power supply and ground connections. If the circuit has 1K (1024) memory cells, this circuit can be organized as a 128 × 8 memory, requiring a total of 19 external connections. Alternatively, the same number of cells can be organized into a 1K × 1 format. In this case, a 10 bit address is needed, but there is only one data line, resulting in 15 external connections.

Figure 5.3 shows such an organization. The required 10-bit address is divided into two groups of 5 bits each to form the row & column addresses for the cell array. A row address selects a row of 32 cells, all of which are accessed in parallel. But, only one of these cells is connected to external data line, based on column address.

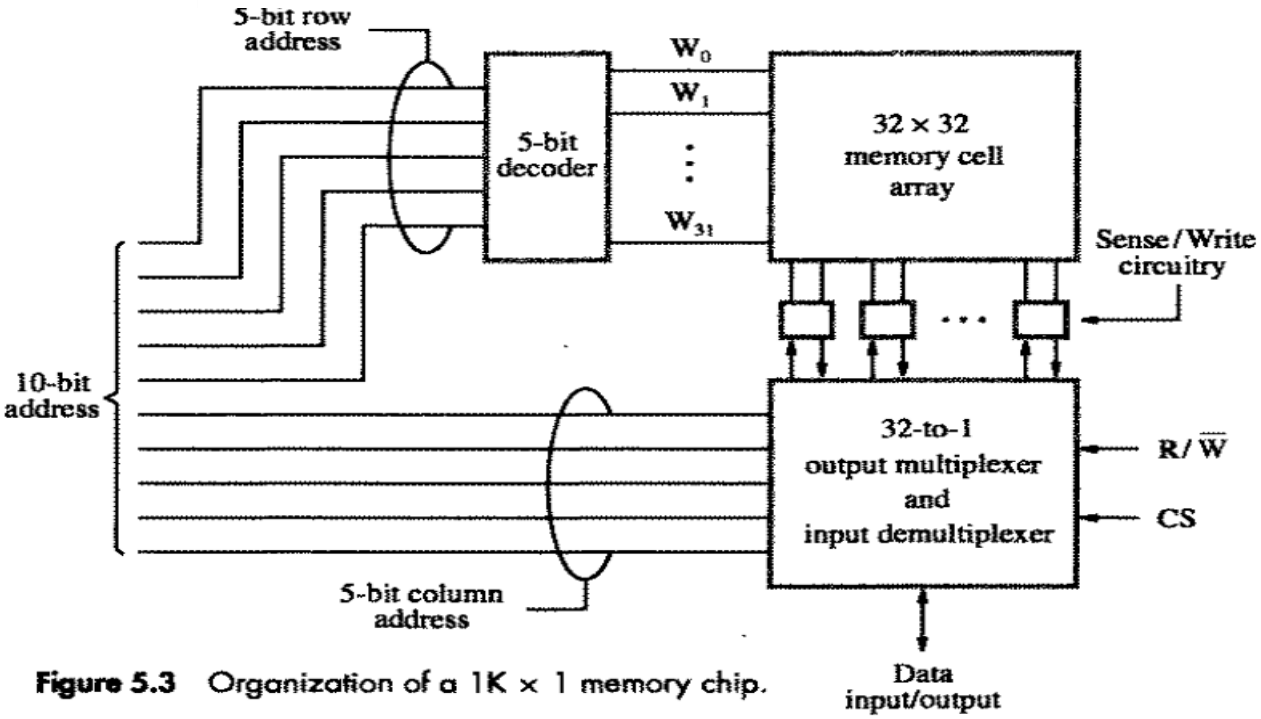


Figure 5.3 Organization of a 1K × 1 memory chip.

STATIC MEMORIES

Memories that consist of circuits capable of retaining their state as long as power is applied are known as *static memories*. Figure 8.2 illustrates how a *static RAM* (SRAM) cell may be implemented. Two inverters are cross-connected to form a latch. The latch is connected to two bit lines by transistors T_1 and T_2. These transistors act as switches that can be opened or closed under control of the word line. When the word line is at ground level, the transistors are turned off and the latch retains its state. For example, if the logic value at point *X* is 1 and at point *Y* is 0, this state is maintained as long as the signal on the word line is at ground level.

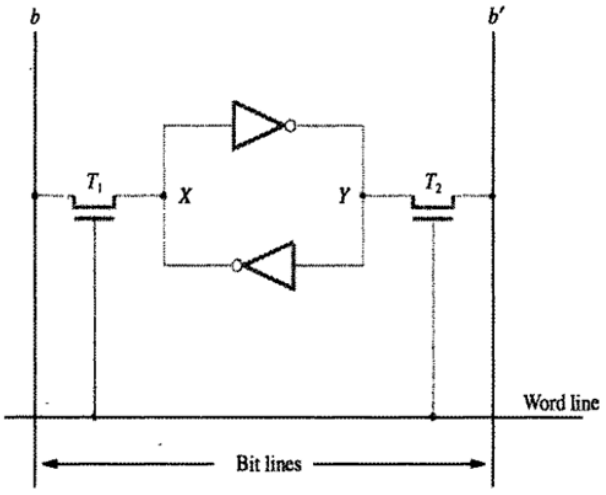


Figure 5.4 A static RAM cell.

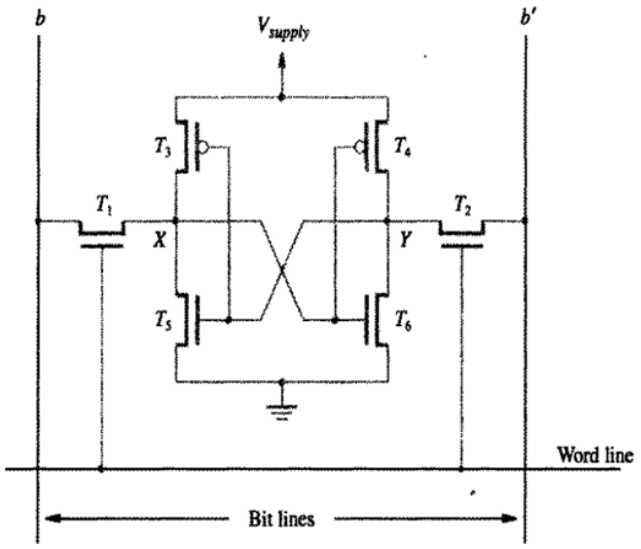


Figure 5.5 An example of a CMOS memory cell.

Read operation: The word line is activated to close switches T_1 and T_2. If the cell is in state 1, the signal on bit line b is high and the signal on bit line b' is low. The opposite is true if the cell is in state 0. Thus, b and b' are always complements of each other. The Sense/Write circuit at the end of the two bit lines monitors their state and sets the corresponding output accordingly.

Write operation: The Sense/Write circuit drives bit lines b and b', instead of sensing their state. It places the appropriate value on bit line $b$and its complement on $b'$and activates the word line. This forces the cell into the corresponding state, which the cell retains when the word line is deactivated.

CMOS Cell

A CMOS realization of the cell in Figure 5.4 is given in Figure 5.5. Transistor pairs (T_3 and T_5) and (T_4 and T_6) form the inverters in the latch. The state of the cell is read or written as just explained. For example, in state 1, the voltage at point X is maintained high by having transistors T_3 and T_6 on, while T_3 and T_5 are off. If T_1 and T_2 are turned on, bit lines b and b'will have high and low signals, respectively.

SRAMs are said to be *volatile* memories because their contents are lost when power is interrupted. Advantage of CMOS SRAMs is their very low power consumption, because current flows in the cell only when the cell is being accessed. Otherwise, T1, T2, and one transistor in each inverter are turned off, ensuring that there is no continuous electrical path between V_{supply} and ground. Static RAMs can be accessed very quickly. SRAMs are used in applications where speed is of critical concern.

ASYNCHRONOUS DRAMS

Information is stored in a dynamic memory cell in the form of a charge on a capacitor. This charge can be maintained for only tens of milliseconds. Since the cell is required to store information for a much longer time, its contents must be periodically refreshed by restoring the capacitor charge to its full value. This occurs when the contents of the cell are read or when new information is written into it. An example of a dynamic memory cell that consists of a capacitor, C, and a transistor, T, is shown in Figure 5.6.

To store information in this cell, transistor T is turned on and an appropriate voltage is applied to the bit line. This causes a known amount of charge to be stored in the capacitor.

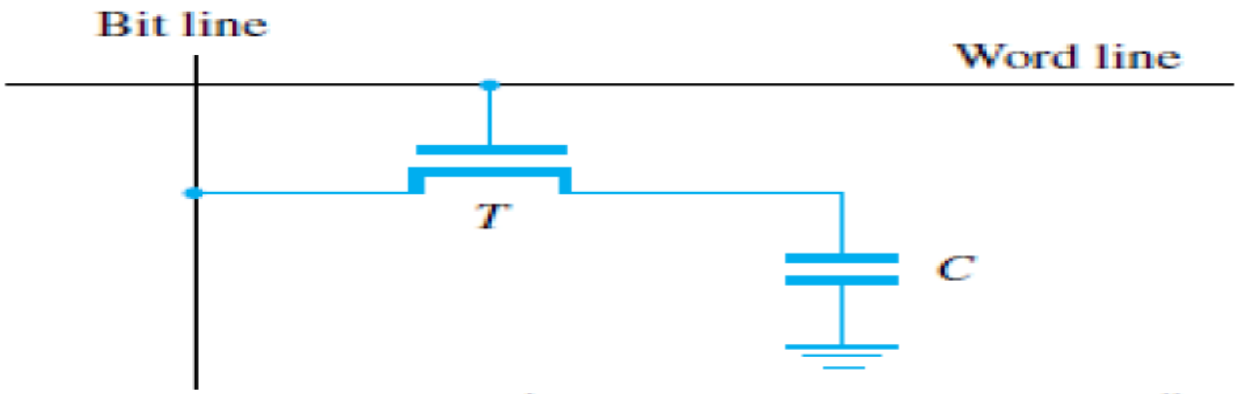


Figure 5.6 A single-transistor dynamic memory cell.

After the transistor is turned off, the capacitor begins to discharge. The information stored in the cell can be retrieved correctly only if it is read before the charge in the capacitor drops below some threshold value. During a Read operation, the transistor in a selected cell is turned on. A sense amplifier connected to the bit line detects whether the charge stored in the capacitor is above or below the threshold value. If the charge is above the threshold, the sense amplifier drives the bit line to the full voltage representing the logic value 1. As a result, the capacitor is recharged to the full charge corresponding to the logic value 1. If the sense amplifier detects that the charge in the capacitor is below the threshold value, it pulls the bit line to ground level to discharge the capacitor fully. Thus, reading the contents of a cell automatically refreshes its contents. Since the word line is common to all cells in a row, all cells in a selected row are read and refreshed at the same time.

A 16 Megabit DRAM chip, configured as 2M × 8, is shown in Figure 5.7. The cells are organized in the form of a 4K x 4K array. The 4096 cells in each row are divided into 512 groups of 8. A row can store 512 bytes of data. 12 address bits are needed to select a row. Another 9 bits are needed to specify a group of 8 bits in the selected row. Thus, a 21 bit address is needed to access a byte in this memory. The high order 12 bits constitute the row address and the low order 9 bits of the address constitute column address of a byte.

During a Read or a Write operation, the row address is applied first. It is loaded into the row address latch in response to a signal pulse on the Row Address Strobe (RAS) input of the chip. Then a Read operation is initiated, in which all cells on the selected row are read and refreshed. Shortly after the row address is loaded, the column address is applied to the address pins and loaded into the column address latch under control of the Column Address Strobe (CAS) signal. The

information in this latch is decoded and the appropriate group of 8 Sense/Write circuits are selected. If the R/W' control signal indicates a Read operation, the output values of the selected circuits are transferred to the data lines, D_{7-0}. For a Write operation, the information on the D_{7-0} lines is transferred to the selected circuits. This information is then used to overwrite the contents of the selected cells in the corresponding 8 columns.

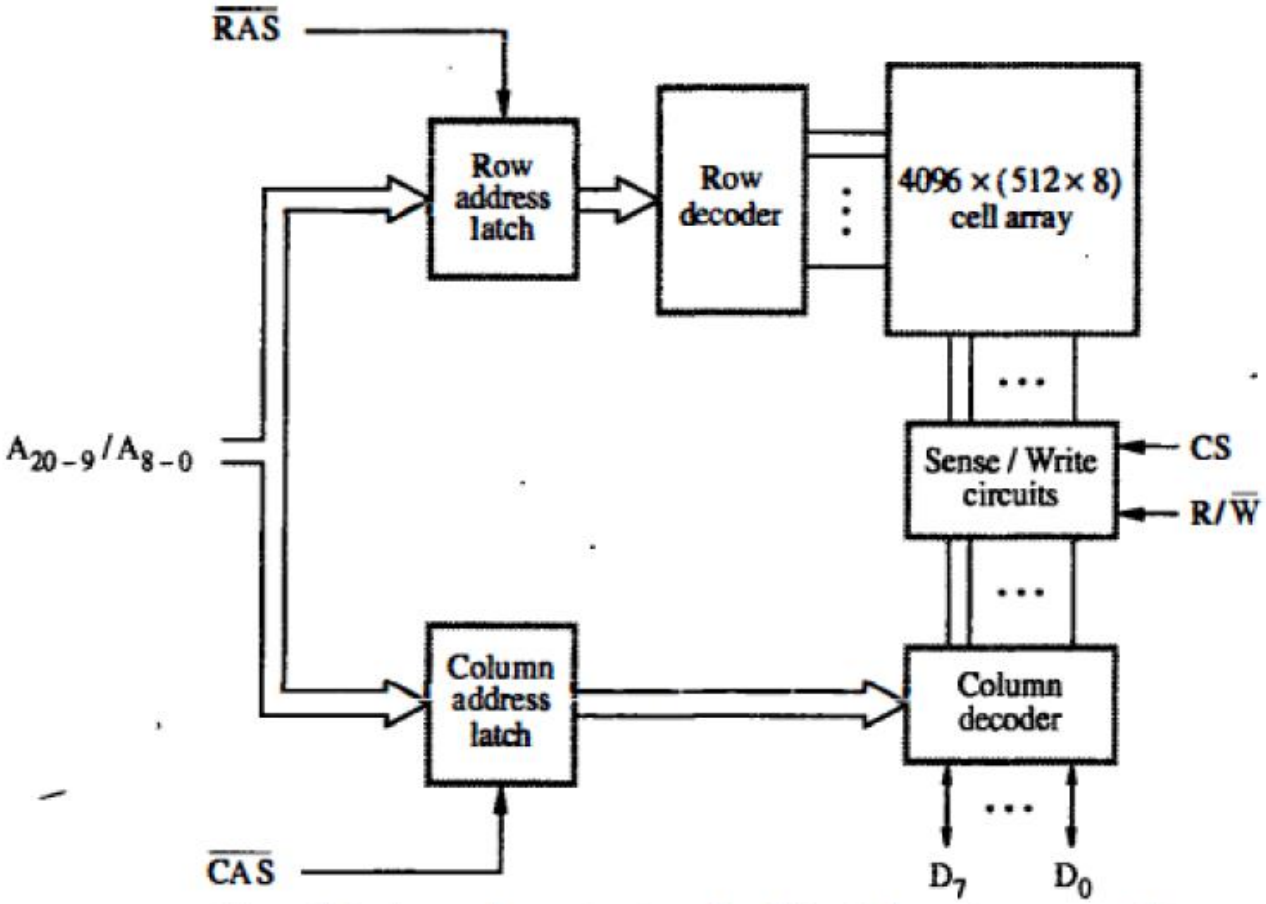


Figure 5.7 Internal organization of a 2M × 8 dynamic memory chip.

To ensure that the contents of a DRAM are maintained, each row of cells must be accessed periodically. A refresh circuit usually performs this function automatically.

Fast Page Mode: Suppose if we want to access the consecutive bytes in the selected row. This can be done without having to reselect the row. Add a latch at the output of the sense circuits in each row. All the latches are loaded when the row is selected. Consecutive sequence of column addresses can be applied under the control signal CAS, without reselecting the row. This allows a block of data to be transferred at a much faster rate than random accesses. A small groups of bytes is referred to as blocks and larger groups as pages. This transfer capability is referred to as the fast page mode feature.

READ ONLY MEMORIES (ROMs)

Introduction: Both SRAM and DRAM chips are volatile: i.e. they lose the contents when the power is turned off. Many applications need memory devices to retain the stored information if power is turned off. For example, when the computer is turned on, the operating system must be loaded from the disk into the memory. This requires non-volatile memory. Non-volatile memory is used extensively in embedded systems. Normal operation involves only reading of data. This type of memory is called Read only memory (ROM).

Read Only Memory (ROM): Figure 5.12 shows a possible configuration for a ROM cell. A logic value 0 is stored in the cell if the transistor is connected to ground at point P; otherwise, a 1 is stored. The bit line is connected through a resistor to the power supply.

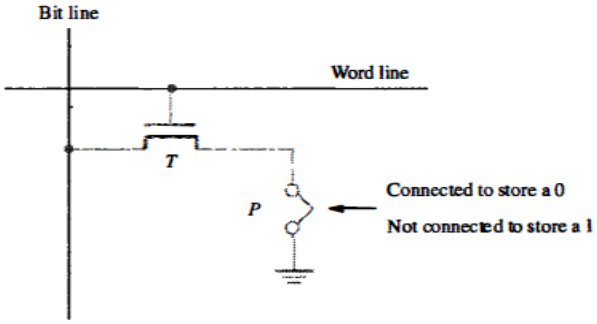


Figure 5.12 A ROM cell.

To read the state of the cell, the word line is activated. Thus, the transistor switch is closed and the voltage on the bit line drops to near zero if there is a connection between the transistor and ground. If there is no connection to ground, the bit line remains at the high voltage, indicating a 1. A sense circuit at the end of the bit line generates the proper output value. Data are written into a ROM when it is manufactured.

Programmable Read Only Memory (PROM): Allows the data to be loaded by a user. Programmability is achieved by inserting a fuse at point P in Figure 5.12. Before it is programmed, the memory contains all 0s. The user can insert 1's at the required locations by burning out the fuses at these locations using high current pulses. This process is irreversible. PROMs provide flexibility & convenience. PROMs provide a faster & less expensive approach because they can be programmed directly by user.

Erasable Programmable Read Only Memory (EPROM): Allows the stored data to be erased and new data to be loaded. It provides considerable flexibility during the development phase of digital systems. They can be used in place of ROMs while software is being developed. Memory changes and updates can be easily made. An EPROM cell has a structure similar to the ROM cell in Figure 5.12. The connection to ground is always made at point P and a special transistor is used, which has the

ability to function either as a normal transistor or as a disabled transistor that is always turned off.

Advantage: Their contents can be erased and reprogrammed. Erasure is done by exposing the chip to ultraviolet (UV) light. EPROM chips are mounted in packages that have transparent windows.

Disadvantage: Chip must be physically removed from the circuit for reprogramming and that its entire contents are erased by the ultraviolet light.

Electrically Erasable Programmable Read Only Memory (EEPROM): They can be both programmed and erased electrically. They do not have to be removed for erasure. It is possible to erase the cell contents selectively. The only disadvantage of EEPROMs is that different voltages are needed for erasing, writing, and reading the stored data.

Flash Memory: A flash cell is based on a single transistor controlled by trapped charge, just like an EEPROM cell. In a flash device, it is possible to read the contents of a single cell, but it is only possible to write an entire block of cells. Flash devices have greater density, which leads to higher capacity and a lower cost per bit. They require a single power supply voltage, and consume less power in their operation.

Applications: Used in portable equipment that is battery driven. Hand held computers, cell phones, digital cameras, MP3 music players, etc.

Single flash chips do not provide sufficient storage capacity for the applications mentioned above. Larger memory modules consisting of a number of chips are needed. There are two popular choices for the implementation of larger memory modules: Flash cards & Flash drives.

Flash Cards: Flash chips are mounted on a small card. Such flash cards have a standard interface that makes them usable in a variety of products. A card is simply plugged into a conveniently accessible slot. Flash cards come in a variety of memory sizes. Typical sizes are 8, 32, and 64 Mbytes. A minute of music can be stored in about 1M byte of memory, using the MP3 encoding format. Hence, a 64 MB flash card can store an hour of music.

Flash Drives: Flash drives are designed to fully emulate the hard disks. The storage capacity of flash drives is significantly lower.

Advantages: i) Flash drives are solid state electronic devices.

ii) They have shorter seek and access times, which results in faster response.

iii) They have lower power consumption, which makes them attractive for battery driven applications, and they are also insensitive to vibration.

Disadvantages: Smaller capacity & higher cost per bit, compared to hard disk drives. Flash memory will deteriorate after it has been written a number of times (Typically one million times).

CACHE MEMORIES

Processor is much faster than the main memory. As a result, the processor has to spend much of its time waiting while instructions and data are being fetched from the main memory. This is the major obstacle towards achieving good performance. The speed of the main memory cannot be increased beyond a certain point. Cache memory is an architectural arrangement which makes the main memory appear faster to the processor than it really is. Cache memory is based on the property of computer programs known as "locality of reference".

Many instructions in localized areas of the program are executed repeatedly during some time period, and the remainder of the program is accessed relatively infrequently. This is called "locality of reference". There are two types:

- Temporal locality of reference: Recently executed instruction is likely to be executed again very soon.
- Spatial locality of reference: Instructions with addresses close to a recently executed instruction are likely to be executed soon.

If the active segments of a program can be placed in a fast cache memory, then the total execution time can be reduced significantly. The memory control circuitry is designed to take advantage of the property of locality of reference. The temporal aspect of the locality of reference suggests that whenever an information is first needed, this item should be brought into the cache. The spatial aspect suggests that instead of fetching just one item from the main memory to the cache, it is useful to fetch several items that reside at adjacent addresses as well. *Block* refers to a set of contiguous address locations of some size. *Cache line* refers to a cache block.

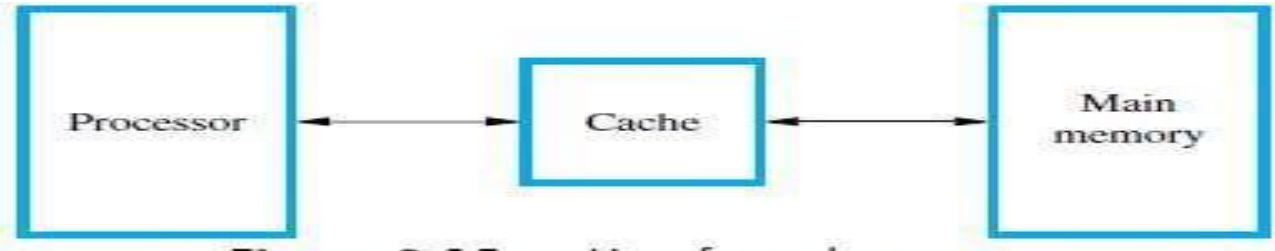


Figure 8.15 Use of a cache memory.

Operation of Cache Memories: Processor issues a Read request, a block of words is transferred from the main memory to the cache, one word at a time. At any given time, only some blocks in the main memory are held in the cache which is determined by a "mapping function". When the cache is full, and a block of words needs to be transferred from the main memory, some block of words in the cache must be replaced, which is determined by a "replacement algorithm".

Cache Hit: The processor does not need to know explicitly about the existence of the cache. It simply issues Read and Write requests using addresses that refer to locations in the memory. The cache control circuitry determines whether the requested word currently exists in the cache. If it does, the Read or Write operation is performed on the appropriate cache location. In this case, a *read hit* or *write hit* is said to have occurred.

In a Read operation, the main memory is not involved. Data is obtained from the cache. For a Write operation, the system can proceed in two ways. In first scheme, *write through protocol*, the cache location and the main memory location are updated simultaneously. In second scheme, *write back or copy back protocol*, only the cache location is updated and it is marked as updated with an associated flag bit, called the *dirty or modified bit*. The main memory location of the word is updated later.

Cache Miss: When the addressed word in a Read operation is not in the cache, a read miss occurs. The block of words that contains the requested word is copied from the main memory into the cache. After the entire block is loaded into the cache, the particular word requested is forwarded to the processor. Alternatively, this word may be sent to the processor as soon as it is read from the main memory. This approach is called *load through, or early restart*. This reduces the processor's waiting period, but at the expense of more complex circuitry.

During a Write operation, if the addressed word is not in the cache, a *write miss* occurs. Then, if the write-through protocol is used, the information is written directly into the main memory. In the case of the write-back protocol, the block containing the addressed word is first brought into the cache, and then the desired word in the cache is overwritten with the new information.

VIRTUAL MEMORIES

Virtual memory is an architectural solution to increase the effective size of the memory system. In most modern computer systems, the physical main memory is not as large as the address space spanned by an address issued by the processor. When a program does not completely fit into the main memory, the parts of it not currently being executed are stored on secondary storage devices, such as magnetic disks. All parts of a program that are eventually executed are first brought into the main memory. When a new segment of a program is to be moved into a full memory, it must replace another segment already in the memory. In modern computers, the operating system moves programs and data automatically between the main memory and secondary storage. Thus, the programmer does not need to be aware of limitations imposed by the available main memory.

Techniques that automatically move program and data blocks into the physical main memory when they are required for execution are called virtual memory techniques. If a virtual address refers to a part of the program or data space that is currently in the physical memory, then the contents of the appropriate location in the main memory are accessed immediately. If the referenced address is not in the main memory, its contents must be brought into a suitable location in the memory before they can be used.

Figure 8.24 shows a typical implementation of virtual memory. A special hardware unit, called the Memory Management Unit (MMU), translates virtual addresses into physical addresses. When the desired data (or instructions) are in the main memory, these data are fetched. If the data are not in the main memory, the MMU causes the operating system to bring the data into the memory from the disk. Transfer of data between the disk and the main memory is performed using the DMA scheme.

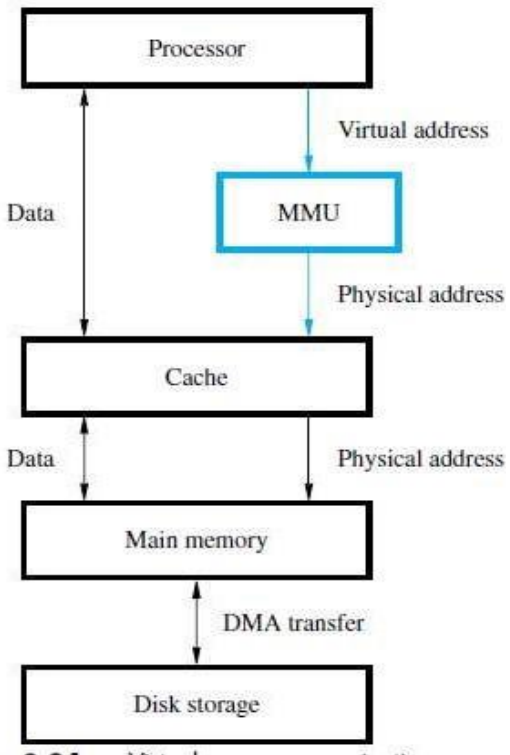
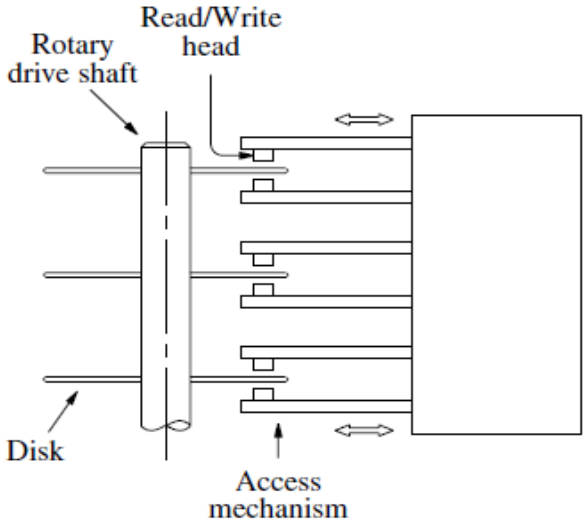


Figure 8.24 Virtual memory organization.

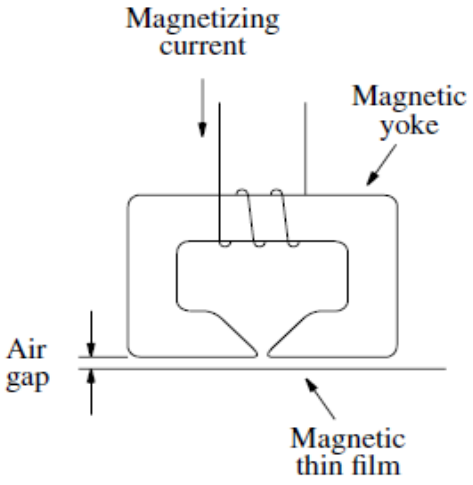
SECONDARY STORAGE

Semiconductor memories cannot be used to provide all of the storage capability needed in computers. Their main limitation is the cost per bit of stored information. Large storage requirements of most computer systems are economically realized in the form of magnetic disks, optical disks, and magnetic tapes, which are usually referred to as secondary storage devices.

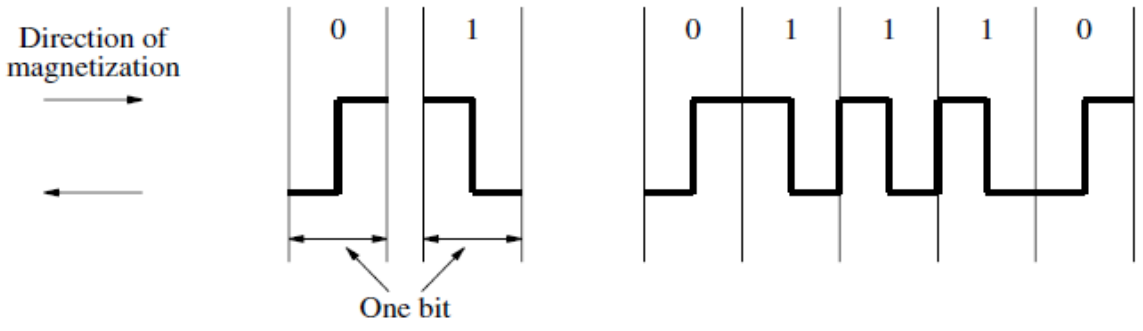
Magnetic Hard Disks: The storage medium in a magnetic disk system consists of one or more disks mounted on a common spindle. A thin magnetic film is deposited on each disk, usually on both sides. The disks are placed in a rotary drive so that the magnetized surfaces move in close proximity to read/write heads, as shown in Figure 5.29a. The disks rotate at a uniform speed. Each head consists of a magnetic yoke and a magnetizing coil, as indicated in Figure 5.29b.



(a) Mechanical structure



(b) Read/Write head detail



(c) Bit representation by phase encoding

Digital information can be stored on the magnetic film by applying current pulses of suitable polarity to the magnetizing coil. This causes the magnetization of the film in the area immediately underneath the head to switch to a direction parallel to the applied field. The same head can be used for reading the stored information. In this case, changes in the magnetic field in the vicinity of the head caused by the movement of the film relative to the yoke induce a voltage in the coil, which now serves as a sense coil. The polarity of this voltage is monitored by the control circuitry to determine the state of magnetization of the film. Only changes in the magnetic field under the head can be sensed during the Read operation. Therefore, if the binary states 0 and 1 are represented by two opposite states of magnetization, a voltage is induced in the head only at 0 to 1 and at 1 to 0 transitions in the bit stream. A long string of 0s or 1s causes an induced voltage only at the beginning and end of the string.

The modern approach is to combine the clocking information with the data (*self-clocking schemes*). One simple scheme, depicted in Figure 5.29c, is known as *phase encoding* or Manchester encoding. In this scheme, changes in magnetization occur for each data bit. A change in magnetization is guaranteed at the midpoint of each bit period, thus providing the clocking information. The drawback of Manchester encoding is its poor bit storage density. The space required to represent each bit must be large enough to accommodate two changes in magnetization.

Read/write heads must be maintained at a very small distance from the moving disk surfaces in order to achieve high bit densities and reliable read/write operations. The flexible spring connection between the head and its arm mounting permits the head to fly at the desired distance away from the surface.

In most modern disk units, the disks and the read/write heads are placed in a sealed, air filtered enclosure. This approach is known as *Winchester technology.* In such units, the read/write heads can operate closer to the magnetized track surfaces because dust particles are absent.

The disk system consists of three key parts. *Disk Platters* usually referred to as the disk. *Disk Drive* comprises the electromechanical mechanism that spins the disk and moves the read/write heads. *Disk Controller,* the electronic circuitry that controls the operation of the system. The disk controller may be implemented as a separate module.

Organization and Accessing of Data on a Disk: The organization of data on a disk is illustrated in Figure 8.28. Each surface is divided into concentric *tracks*, and each track is divided into *sectors*. The set of corresponding tracks on all surfaces of a stack of disks forms a logical *cylinder*. The data on all tracks of a cylinder can be accessed without moving the read/write heads. The data are accessed by specifying the surface number, the track number, and the sector number. The Read and Write operations start at sector boundaries.

Data bits are stored serially on each track. Each sector usually contains 512 bytes of data. The data are preceded by a sector header. Following the data, there are additional bits that constitute an *error correcting code* (ECC). The ECC bits are used to detect and correct errors that may have occurred in writing or reading of the 512 data bytes. To easily distinguish between two consecutive sectors, there is a small intersector gap.

An unformatted disk has no information on its tracks. The formatting process divides the disk physically into tracks and sectors. The formatting comprises the sector headers, the ECC bits, and intersector gaps. In a typical computer, the disk is subsequently divided into logical partitions. There must be at least one such partition, called the primary partition.

Figure 8.28 indicates that each track has the same number of sectors. So all tracks have the same storage capacity. Thus, the stored information is packed more densely on inner tracks than on outer tracks. This arrangement is used in many disks because it simplifies the electronic circuits needed to access the data.

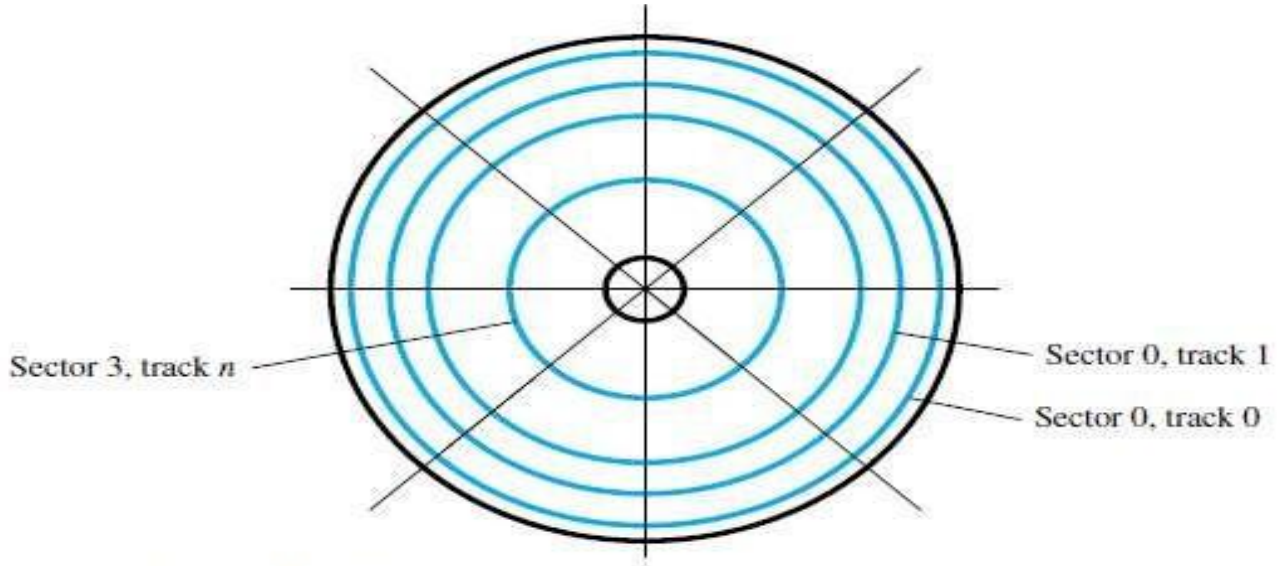


Figure 8.28 Organization of one surface of a disk.

Access Time: *Seek time* is the time required to move the read/write head to the proper track. It depends on the initial position of the head relative to the track specified in the address. The average values are in the 5 to 8 ms range. *Rotational delay (Latency time)* is the amount of time that elapses after the head is positioned over the correct track until the starting position of the addressed sector passes under the read/write head. On average, this is the time for half a rotation of the disk. The sum of these two delays is called the disk *access time*.

Data Buffer/Cache: A disk drive is connected to the rest of a computer system using some standard interconnection scheme, such as SCSI (Small Computer System Interface) or SATA (Serial Advanced Technology Attachment). The interconnection hardware is usually capable of transferring data at much higher rates than the rate at which data can be read from disk tracks. An efficient way to deal with the possible differences in transfer rates is to include a *data buffer* in the disk unit. The buffer is a semiconductor memory, capable of storing a few megabytes of data. The requested data are transferred between the disk tracks and the buffer at a rate dependent on the rotational speed of the disk. Transfers between the data buffer & main memory can then take place at the maximum rate allowed by the bus.

The data buffer can also be used to provide a caching mechanism for the disk. When a read request arrives at the disk, the controller can first check to see if the desired data are already available in the cache (buffer). If so, the data are transferred to the memory in microseconds instead of milliseconds. Otherwise, the data are read from a disk track in the usual way, stored in the buffer, then transferred to memory.

Disk Controller: Operation of a disk drive is controlled by a disk controller circuit. It also provides an interface between the disk drive and the rest of the computer system. One disk controller may be used to control more than one drive. Figure 5.31 shows a disk controller which controls two disk drives.

A disk controller that communicates directly with the processor contains a number of registers that can be read and written by the operating system. Thus, communication between the OS and the disk controller is achieved in the same manner as with any I/ O interface. The disk controller uses the DMA scheme to transfer data between the disk and the main memory.

The OS initiates the transfers by issuing Read and Write requests. Controller's registers are loaded with necessary information like:

Main memory address: The address of the first main memory location of the block of words involved in the transfer.

Disk address: The location of the sector containing the beginning of the desired block of words.

Word count: The number of words in the block to be transferred.

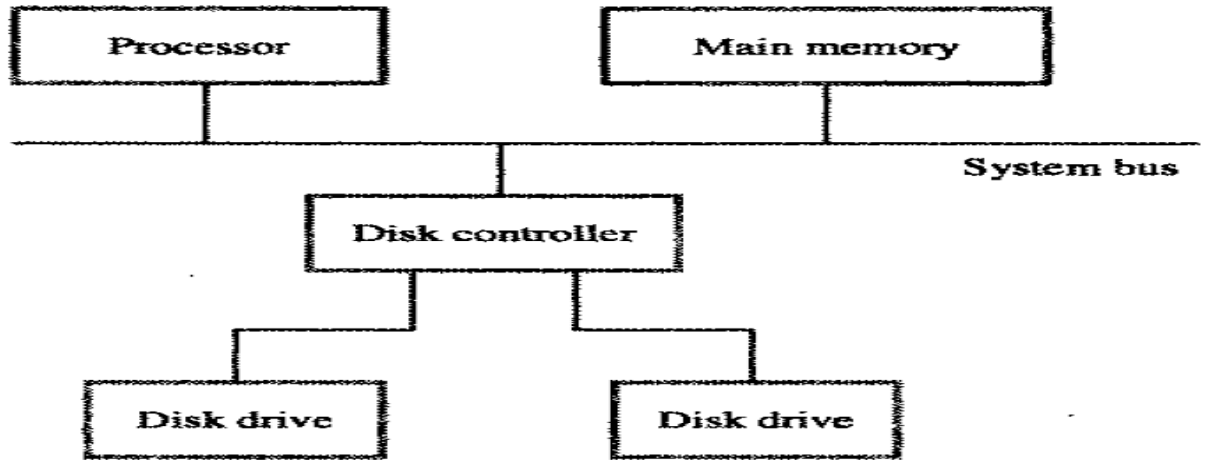


Figure 5.31 Disks connected to the system bus.

On the disk drive side, the controller's major functions are:

Seek: Causes the disk drive to move the read/write head from its current position to the desired track.

Read: Initiates a Read operation, starting at the address specified in the disk address register. Data read serially from the disk are assembled into words and placed into the data buffer for transfer to the main memory. The number of words is determined by the word count register.

Write: Transfers data to the disk, using a control method similar to Read operations.

Error checking: Computes the error correcting code (ECC) value for the data read from a given sector and compares it with the corresponding ECC value read from the disk. In the case of a mismatch, it corrects the error if possible; otherwise, it raises an interrupt to inform the OS that an error has occurred. During a Write operation, the controller computes the ECC value for the data to be written and stores this value on the disk.

Software and Operating System Implications: All data transfer activities involving disks are initiated by the operating system. The disk is a non-volatile storage medium, so the OS itself is stored on a disk. During normal operation of a computer, parts of the OS are loaded into the main memory and executed as needed. When

power is turned off, the contents of the main memory are lost. When the power is turned on again, the OS has to be loaded into the main memory, which takes place as part of a process known as *booting*.

To initiate booting, a tiny part of main memory is implemented as a non-volatile ROM. This ROM stores a small monitor program that can read and write main memory locations as well as read one block of data stored on the disk at address 0. This block, referred to as the *boot block*, contains a loader program. After the boot block is loaded into memory by the ROM monitor program, it loads the main parts of the OS into the main memory.

Floppy Disks: Floppy disks are smaller, simpler, and cheaper disk units that consist of a flexible, removable, plastic diskette coated with magnetic material. The diskette is enclosed in a plastic jacket, which has an opening where the read/write head can be positioned. A hole in the centre of the diskette allows a spindle mechanism in the disk drive to position and rotate the diskette.

The main feature of floppy disks is their low cost and shipping convenience. However, they have much smaller storage capacities, longer access times, and higher failure rates than hard disks. In recent years, they have largely been replaced by CDs, DVDs, and flash cards as portable storage media.

RAID Disk Arrays: Redundant Array of Independent Disks (RAID) is originally Redundant Array of Inexpensive Disks. Using multiple disks makes it cheaper for huge storage, and also possible to improve the reliability of the overall system. Different configurations were proposed, and many more have been developed since.

RAID0-data striping:

RAID1-identical copies of data on two disks

RAID2, 3, 4-increased reliability

RAID5-parity based error recovery

RAID10-combines the features of RAID 0 and RAID 1.

COMPUTER ORGANIZATION AND ARCHITECTURE
BEC306C

Important Questions:

1. Explain the connection of the main memory to the processor.
2. With a neat diagram, explain the internal organization of 16 x 8 memory chip.
3. Explain the internal organization of 1Mx1 dynamic memory chip with neat diagram.
4. Explain the working of 1-bit CMOS SRAM cell with a schematic. (Illustrate internal structure of static memory)
5. Discuss a single-transistor dynamic memory cell.
6. Explain the internal organization of 2M x 8 DRAM chip with neat diagram.
7. Discuss different types of ROM. (Discuss different types of non-volatile memory concepts)
8. Explain the operation of cache memories. (Discuss about the use of Cache memory in the processor system)
9. With a neat diagram, explain virtual memory organization.
10. With a neat diagram, explain the principal of working of magnetic disk.
11. Write a short note on magnetic hard disk.
12. What are the major functions of disk controller?