LESSON

10

MEMORY HIERARCHY

CONTENTS

10.0 AIMS AND OBJECTIVES

After studying this lesson, you will be able to:

- Describe memory hierarchy
- Explain virtual memory

- Define cache memory
- Understand memory management hardware

10.1 INTRODUCTION

Memory system is at the heart of a computer system. It is the memory system that makes what a computer is. The input data, the instructions necessary to manipulate the input data as also the output data are all stored in the memory.

Memory unit is an essential part of any digital computer because computer processes data only if it is stored somewhere in its memory. For example, if computer has to compute f(x)=sinx for a given value of x, then first of all x is stored in memory somewhere, then a routine is called that contains program that calculates sine value of a given x. It is an indispensable component of a computer. Computers now a days are capable of storing several programs in memory simultaneously so that a user can switch from one application to another. The operating system has to allocate memory to each application as well as to itself. It is the duty of the OS to ensure that the different types of memory in the system must be used properly so that each process can run most effectively. It also keeps track of free memory. The memory manager performs all these tasks. In this lesson we will discuss various types of memory, their hierarch and memory management.

10.2 MEMORY HIERARCHY

Memories vary in their design as also in their capacity and speed of operation. A typical computer can have all types of memories. According to their nearness to the CPU, memories form a hierarchy structure as shown below.

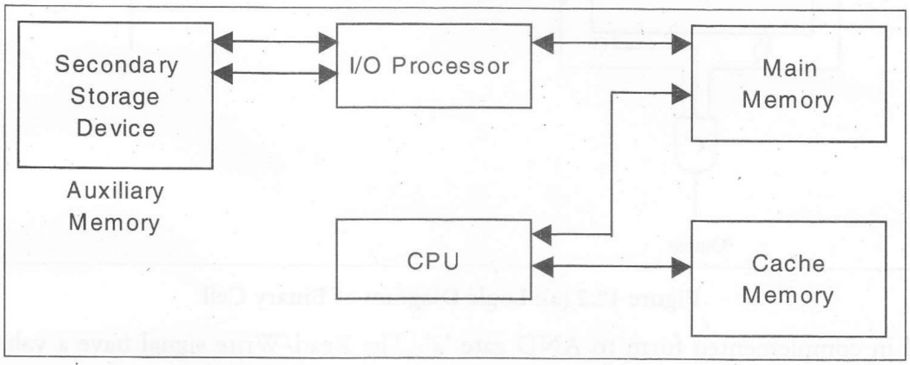


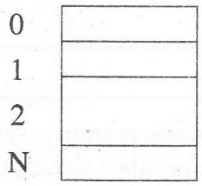
Figure 10.1: Memory Hierarchy in a Computer System

10.2.1 Main Memory or Primary Memory

Random Access Memory

Here we will confine out discussions in general to the Random Access Memory only, which also contains the discussion on Main Memory. Main memory is a random access memory. It is normally organized (locally) as words of fixed length. The length of a word is called word length. Each of these memory words have an independent address and each have same number of bits. Normally the total number of words in memory are some power of 2.

Address Word length



The access time and cycle time in RAMs are constant and independent of the location accessed. But let us first discuss how a bit can be stored using a sequential circuit. Figure 10.2(a) shows the logic diagram of a binary cell. A 4 × 4 RAM is shown in Figure 10.2(b).

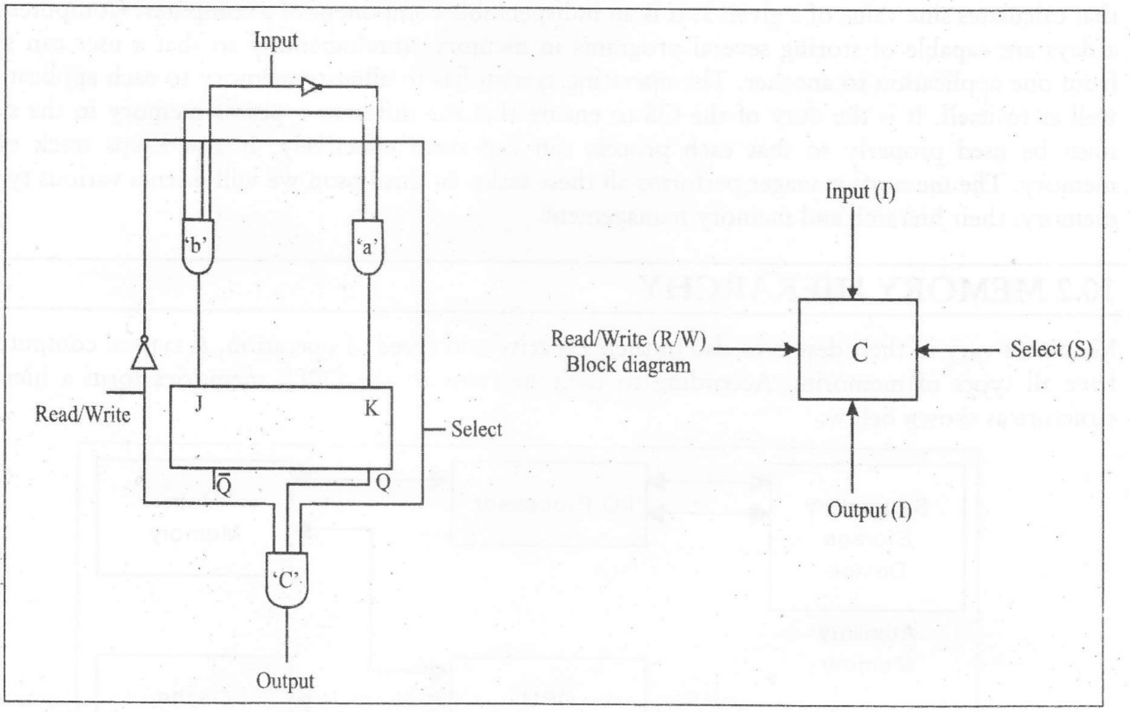


Figure 10.2 (a): Logic Diagram of Binary Cell

Input is fed in complemented form to AND gate 'a'. The Read/Write signal have a value 1 if it is a read operation. Therefore, during the read operation only the AND gate 'c' becomes active. Since AND gate 'a' & 'b' have 0 inputs, and if the select is 1, i.e. this cell is currently being selected, then the output will become equal to the state of flip-flop. In other words the data value stored in flip-flop has been read. In write operation only 'a' & 'b' gates become active and they set or clear the JK flip flop depending on the input value. Please note that in case input is 0, the flip flop will go to clear state and if input is 1, the flip flop will go to set state. In effect, the input data is reflected in the state of the flip flop. Thus, we say that the input data has been stored in flip flop or binary cell.

Figure 10.2(b) is the extension of this binary cell to a IC RAM circuit, where a 2 × 4 bit decoder is used. Please note that each decoder output is connected to a 4 bit word and the read/write signal is supplied to each binary cell. The output is derived using an OR gate, since all the non-selected cells will produce a zero output. The word which is selected will determine the overall output.

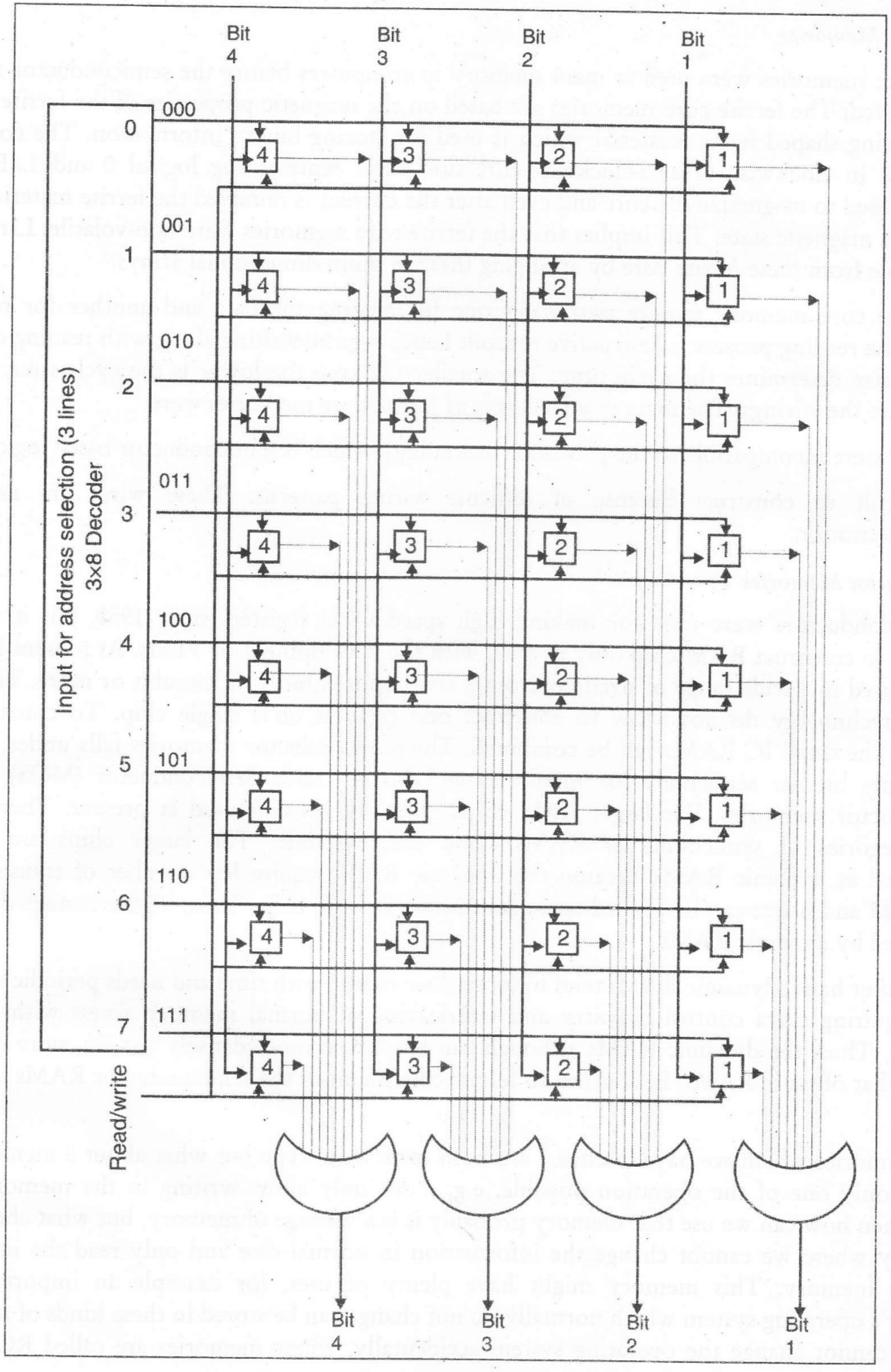


Figure 10.2(b): Logic Diagra.n of RAM

After discussing so much about the general configuration of RAMs. Let us discuss few technologies and techniques used in RAMs.

Ferrite-core Memories

Ferrite core memories were used as main memory in computers before the semiconductor memories were invented. The ferrite core memories are based on the magnetic properties of the ferrite material. Core is a ring shaped ferrite material which is used for storing binary information. The core can be magnetized in clockwise of anti-clockwise direction, thus representing logical 0 and 1. Electronic current is used to magnetize the core and even after the current is removed the ferrite material stays in the specific magnetic state. This implies that the ferrite core memories were non-volatile. Large RAMs can be made from these ferrite core by arranging these in multi-dimensional arrays.

The ferrite core memory require two wires one for writing the data and another for reading or sensing. The reading process is destructive readout hence require writing along with reading operation. The core size determines the cycle time. The smaller the core the lower is the cycle time but more complex are the wirings. The main disadvantages of ferrite-core memories were:

1. They were incompatible to the processor technology which is semiconductor based logic circuits.
2. Difficult to construct because of difficult wiring patterns. These wires are needed for magnetization.

Semiconductor Memories

The semiconductors were used for making high speed CPU registers since 1950, but it was made economic to construct RAM chip only in 1970 with the development of VLSIs. At present RAMs are manufactured in a wide range of sized i.e. from a few hundred bits to a megabit or more. The present limit on technology do not allow to construct one giga bit on a single chip. To construct large memories the small IC RAMs can be combined. The semiconductor memories falls under two main technologies bipolar semiconductor memories and Metal Oxide Semiconductor (MOS) transistor semiconductor memories. For larger RAM chips normally MOS is used at present. There are two main categories of semiconductor RAMs static and dynamic. The larger chips are normally constructed as dynamic RAMs because the dynamic RAM require less number of transistors than static RAM and hence can be packed more densely on a single chip. Thus, higher storage density can be achieved by dynamic RAMs.

On the other hand, dynamic RAMs tend to loose their charge with time and needs periodic refreshing. Thus, requiring extra control circuitry and interleaving of normal memory access with refreshing operation. Thus, the dynamic RAMs although can be packed more densely yet are more difficult to use than that of static RAMs. In contrast to ferrite core memory the semiconductor RAMs are volatile in nature.

These memories which we have discussed are both read/write type but what about a memory where we have only one of the operation possible, e.g. if we only allow writing in the memory and no reading then how can we use that memory probably it is a wastage of memory, but what about having a memory where we cannot change the information in normal case and only read the information from the memory. This memory might have plenty of uses, for example an important bit of computer's operating system which normally do not change can be stored in these kinds of memory so that one cannot change the operating system accidentally. These memories are called ROMs (Read Only Memories)

Read Only Memories

A ROM is basically a combinational circuit and can be constructed as:

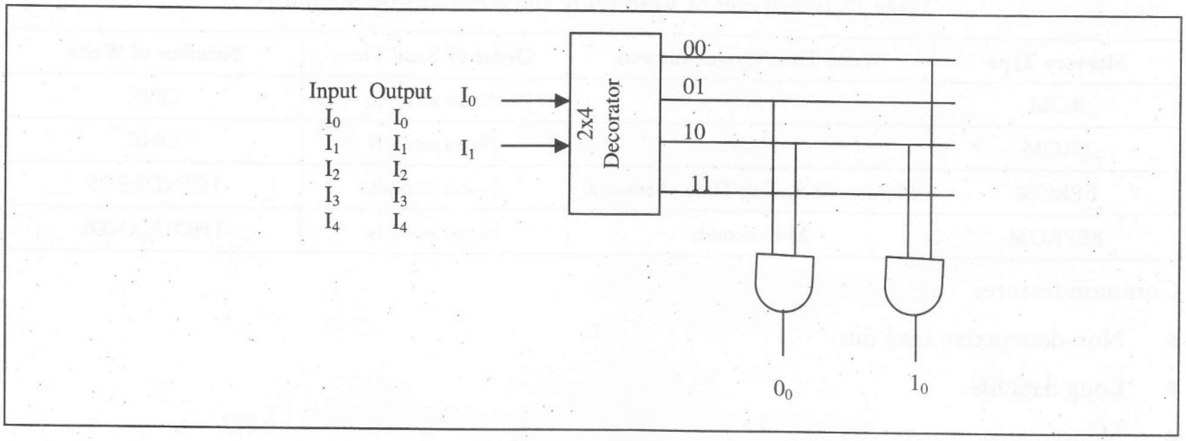


Figure 10.3: A Sample ROM

Thus, using this hardwired combinational circuit we can create ROM. Please note that on applying an Input $I_0 = 0$, $I_1 = 0$ we will get $O_0 = 0$ and $O_1 = 1$; on applying $I_0 = 0$ and $I_1 = 1$ we will get $O_0 = 1$ and $O_1 = 0$ as 01 line of decoder will be selected. This same logic can be used for constructing larger ROMs.

ROMs (Read Only Memories) are the memories on which it is not possible to write the data when they are on line to the computer. They can only be read. The ROMs can be used in storing Microprograms, Systems programs, subroutines: all these terms will be dealt with in greater details in course 2. ROMs are non-volatile in nature and need not be loaded in a secondary storage device. ROMs are fabricated in large number in a way where there is no room for even a single error.

But this is an inflexible process and requires mass production, therefore, a new kind a ROM called PROM was designed which is also non-volatile and can be written only once and hence the name Programmable ROM (PROM). The writing process in PROM can be performed electrically by the supplier or the customer. A special equipment is needed to perform this writing operation, Therefore, PROMs are more flexible and convenient that ROMs.

The ROMs/PROMs can be written just once (in ROMs at the time of manufacture and PROMs at any time later also), but in both the cases once whatever is written on, cannot be changed. But what about a case where you read mostly but write only very few times. This lead to the concept of Read mostly memories and the best examples of these are EPROMs (Erasable PROMs) and EEPROMs (Electrically Erasable ROMs). The EPROM can be read and written electrically. But, the write operation is not simple. It requires erasure of whole storage cells by exposing the chip to ultra violet light, thus bring them to same initial state. This erasure is a time consuming process. Once all the cells have been brought to same initial state, then the EPROM can be written electrically. EEPROMs are becoming increasingly popular as they do not require prior erasure of previous contents. However, in EEPROMS the writing time is considerably higher than reading time. The biggest advantage of EEPROM is that it is non-volatile memory and can be updated easily, while the disadvantages are the high cost and at present they are not completely non-volatile and the write operation takes considerable time. But all these advantages are disappearing with the growth in technology. In general, ROMs are considered slower than RAMs. Table 10.1 summarizes the features of these read only and read mostly memories.

Table 10.1: Features of Read Only and Read Mostly Memories

Memory Type	Write Time Cycles allowed	Order of Read Time	Number of Write
ROM	Once	Nano seconds	ONE
PROM	Hours	Nano seconds	ONE
EPROM	Minutes (including Time of erasure)	Nano seconds	HUNDREDS
EEPROM	Milliseconds	Nano seconds	THOUSANDS

Common features

- Non-destructive read out
- Long data life
- Non-volatile

One of the new memory technology is flash memory. These memories can be reprogrammed at high speed and hence the name flash. The flash memory characteristics such as cost and write time, etc. fall in between of EPROM and EEPROM. In flash memories the entire memory can be erased in few seconds (compare it to EPROM) by using electrical erasing technology. There is another possibility in flash memory which is erasure of a block is possible in it.

After discussing so much about semiconductor memories, let us discuss about the chip organization of these memories.

Design of Main Memory

Most of the semiconductor memories are packaged in chips. As discussed earlier these memory chips may store information ranging from 64K bits to 1M bits. There is several memory organization techniques used for a chip and the most common of these are 2D and 21/2D organization.

2D Memory Organization: In this organization the memory on a chip is considered to be a list of words in which any word can be accessed randomly e.g. the memory of PC-286 have 16 bit words and normally in a chip it have 64KB memory = 32K words. Figure 9.4(a) shows a typical 2D organization.

The memory in 2D chip is organized as an array of words. The horizontal lines are connected to the select input of the binary cells. Each vertical bit line is connected to the Data-in (or input) and sense (or output) terminal of each cell in its respective column. Each decoded line of decoder drives a word line. A complete word can be input or output from the memory simultaneously.

On write operation (input ot memory) the address decoder selects the required word and the bit lines are activated for a value 0 or 1 according to the data line values. Thus, enabling input of data to memory or in other words completes the write operation. On read (output from memory) the value of each bit line is passed through a sense amplifier and passed on to the data lines. Thus, enabling the read operation. The word line identifies the word which has been selected for reading or writing. Usually. ROMs and read mostly memories use 2D chip organization.

Another chip organization which is gaining popularity is 2½ D organization. In 2½ D organization, bits of a word are-spread over a number of chips. For example a 32 bit word can be stored on four chips containing 8 bits of the word. But the ideal organization in this will be to have 1 bit of a word on a single chip. A 2½ D circuit is given in Figure 9.4(b). The figure is a square array of cells. A row line and a column line are connected to each memory cell. The address supplied in this chip is divided into Row and Column address lines and are then used to input or output bit/bits from this memory chip. Rest of the bits of this word can be delivered y the other similar memory chips. This organization is increasingly finding its applications in RAM construction.

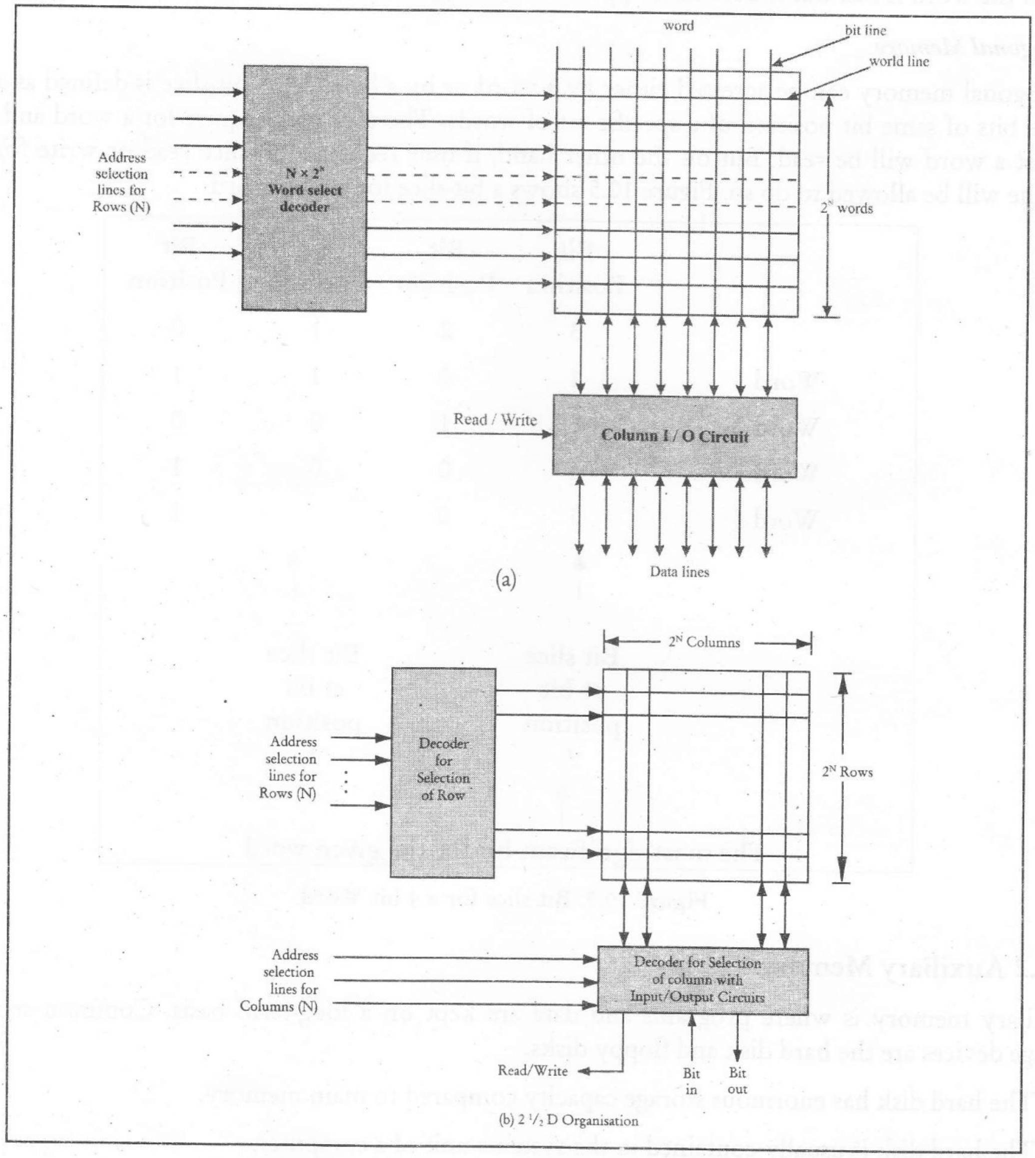

word
bit line
world line
Address selection lines for Rows (N)
N × 2^N Word select decoder
2^N words
Read / Write
Column I / O Circuit
Data lines
(a)
2^N Columns
Address selection lines for Rows (N)
Decoder for Selection of Row
2^N Rows
Address selection lines for Columns (N)
Decoder for Selection of column with Input/Output Circuits
Read/Write
Bit in
Bit out
(b) 2 1/2 D Organisation

Figure 10.4 (a) and (b): 2D and 2½D Chip Organization

Comparison of 2D and 2½ D Organization

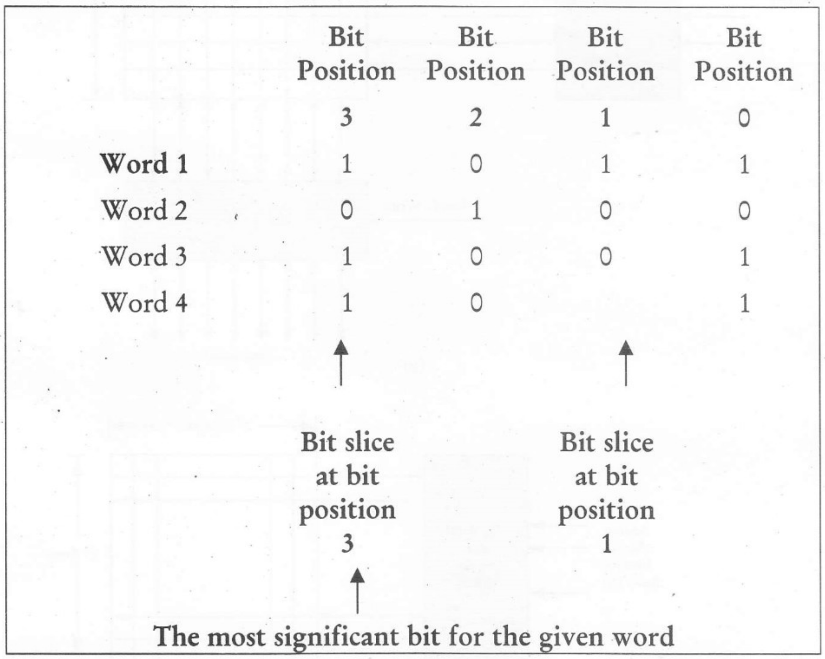
The 2½ D organization of chips is supposed to be more advantageous because:

1. It require less circuitry and Gates. (Why? Find out from Further readings)

2. The chip have only one input/output pin in 2½ D while in 2D it has to have 16 or 32 input/output pins, thus in the chip packages less number of pins are required for 2½ D organization which is a desirable feature.
3. In the 2D organization the error correction codes cannot be used effectively. For example, if an electromagnetic disturbance have effected a chip, in 2½ D we can rectify the error as only one bit of the word is lost but it does not happen in a 2D organization.

Orthogonal Memory

Orthogonal memory can be accessed either by a word or by a bit-slice. A bit-slice is defined as a set of all the bits of same bit position of a specific set of words. The user may request for a word and on his request a word will be read. But on the other hand, if user request a bit-slice read or write for a bit-slice, he will be allowed to do so. Figure 10.5 shows a bit-slice for a 4-bit word.



	Bit Position	Bit Position	Bit Position	Bit Position
	3	2	1	0
Word 1	1	0	1	1
Word 2	0	1	0	0
Word 3	1	0	0	1
Word 4	1	0		1

Bit slice at bit position 3
Bit slice at bit position 1
The most significant bit for the given word

Figure 10.5: Bit-slice for a 4 bit Word

10.2.2 Auxiliary Memory

Auxiliary memory is where programs and data are kept on a long-term basis. Common secondary storage devices are the hard disk and floppy disks.

- The hard disk has enormous storage capacity compared to main memory.
- The hard disk is usually contained in the systems unit of a computer.
- The hard disk is used for long-term storage of programs and data.
- Data and programs on the hard disk are organized into files--named sections of the disk.

A hard disk might have a storage capacity of 40 gigabytes. This is about 300 times the amount of storage in main memory (assuming 128 megabytes of main memory.) However, a hard disk is very slow compared to main memory. The reason for having two types of storage is this contrast:

Primary memory	Auxiliary memory
1. Fast	1. Slow
2. Expensive	2. Cheap
3. Low capacity	3. Large capacity
4. Connects directly to the processor	4. Not connected directly to the processor

Floppy disks are mostly used for transferring software between computer systems and for casual backup of software. They have low capacity, and are very, very slow compared to other storage devices.

Apart from main memory there is secondary memory too, which works slower than the main memory and is used to provide a backup. It is also called auxiliary memory. The main memory gathers the data required currently for processing and CPU uses this data.

10.2.3 Cache Memory

A cache memory is an intermediate memory between two memories having large difference between their speeds of operation. Cache memory may thus be between a hard disk and the RAM. It may also be inserted between CPU and RAM to hold the most frequently used data and instructions. Communicating with devices with a cache memory in between enhances the performance of a system significantly. We will discuss cache memory latter in this lesson in detail.

10.3 ASSOCIATIVE MEMORY

The access time to find an item can be reduced significantly 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.

An associative memory is more costly than a RAM because each cell must have storage capability as well as logic circuits for matching its contents with an external argument. Associative memories are used in applications where the search time is very critical and essentially short, because it performs parallel searches by data association; moreover searches can be done on an entire word or on a field within a word. This memory can also find the unused data location to store the word. When a word is to be read by such a memory, the content/part of the word is specified.

Whenever a word is written in this memory, no address is to be given.

10.3.1 Hardware Organization of Associative Memory

It consists of:

1. A memory array and logic for m words within bits per word.
2. There is an n-bit argument register one for each bit of word.
3. There is an n-bit key register one for each bit of word.
4. There is an m-bit match register one for each memory word.

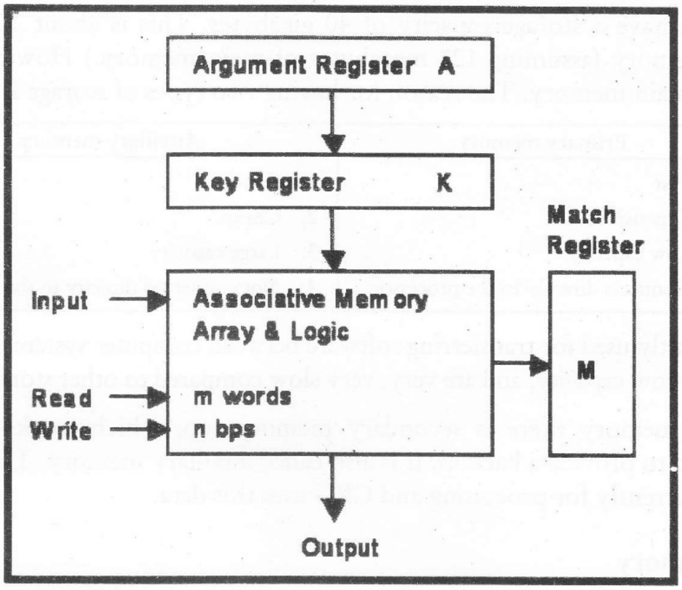


Figure 10.6: Block Diagram of Associative Memory

10.3.2 Working

Content of argument register and word are compared. The words that match the bits of the argument register set a corresponding bit in match register. After matching, the bits in match register that are set indicate the fact that their corresponding words have been matched. Reading is done by sequential access to memory for those words whose corresponding bits in the match register have been set.

The key register provides a mask for choosing a particular field or key in the argument word. The entire argument is compared with each memory word if key register contains all 1's, else only those bits in argument that have 1's in their corresponding position of key are compared.

So key register provides information about how the reference to memory is made.

RAM is also called Read/Write memory.

Read Only Memory: It is the memory in which data is stored for permanent use. Data cannot be written in or deleted from this type of memory.

CDROM is a common example.

Building Large Memories using Chips

Given a particular RAM chip of specified capacity and some decoders, it is possible to extend memory by using similar types of chips with decoders.

An example:

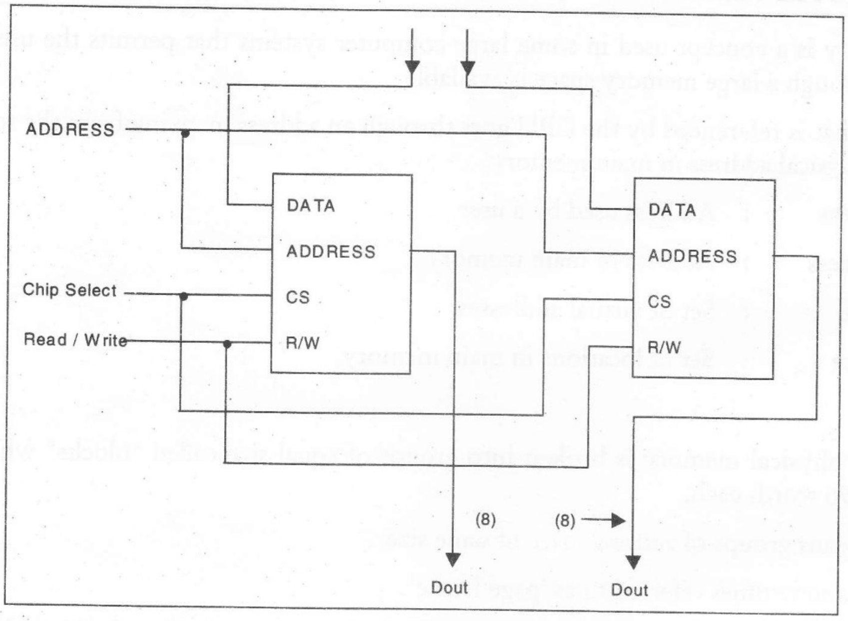


Figure 10.7: Block Diagram of 1K*16 RAM with 1K*8 RAMS (Parallel Connection)

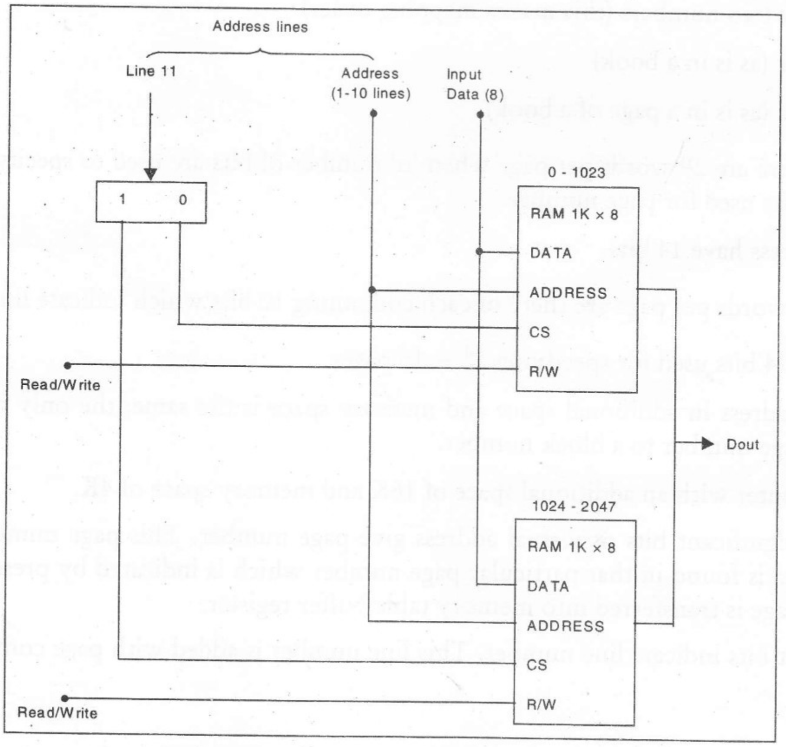


Figure 10.8: Block Diagram of a 2K*8 RAM (Series Connection)

10.4 VIRTUAL MEMORY

Virtual memory is a concept used in some large computer systems that permits the user to construct programs as though a large memory space is available.

Each address that is referenced by the CPU goes through an address mapping from the so-called virtual address to a physical address in main memory.

Virtual Address	:	Address used by a user.
Physical Address	:	Address in main memory.
Address Space	:	Set of virtual addresses.
Memory Space	:	Set of locations in main memory.

Paging

In paging the physical memory is broken into groups of equal size called "blocks" which may range from 64 to 4096 words each.

- "Page" means groups of address space of same size.
- "Block" is sometimes referred to as 'page frame'.

The mapping from address space to memory space is facilitated if each virtual address is considered to be represented by two numbers (this makes mapping easier)

1. Page number (as is in a book)
2. Line number (as is in a page of a book)

Now suppose there are 2^n words per page when 'n' number of bits are used to specify line address and rest high order bits used for page number.

Let a virtual address have 14 bits.

Since $2^{10} = 1$ K words per page are there of each containing 10 bits which indicate line address.

Rest of (14-10) = 4 bits used for specifying $2^4 = 16$ pages.

Note that line address in additional space and memory space is the same; the only mapping required for mapping a page number to a block number.

Consider a computer with an additional space of 16K and memory space of 4K.

The four most significant bits of virtual address give page number. This page number is searched in page table. If data is found in that particular page number which is indicated by presence bit, then the content of this page is transferred into memory table buffer register.

10 less significant bits indicate line number. This line number is added with page content to give block address.

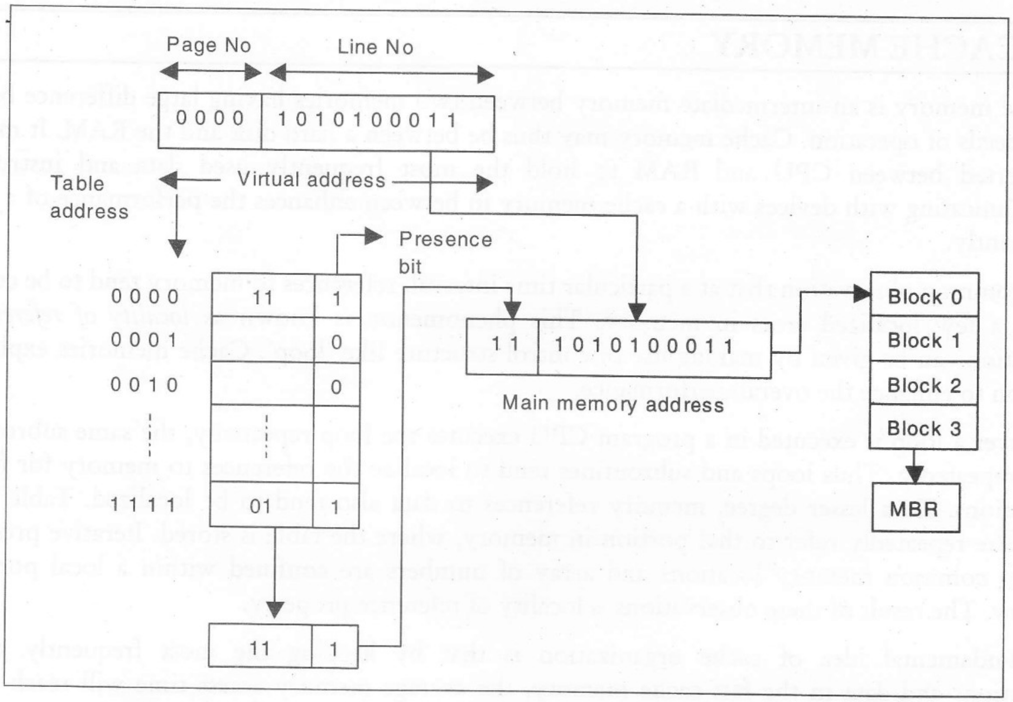


Figure 10.9 Memory Page Table

Page Replacement

For efficient utilization of memory space, memory management software system handles all the software operations. It must make decisions about:

1. When page is to be transferred from auxiliary memory to main memory.
2. Which page should be removed from main memory.
3. Where the new page is to be placed in main memory.

Page Fault

When a program starts, execution pages are to be transferred from auxiliary to main memory.

When the required page is not in main memory, page fault occurs. Until the required page is brought into memory, the process is suspended.

Page Replacement Algorithm

- FIFO (First In First Out)

  According to this policy that page is replaced with a new page (obviously when memory is full) which had entered the memory first.

- LRU (Least Recently Used)

  According to this policy the page that has taken a long time for not being used and lying in cache.

10.5 CACHE MEMORY

A cache memory is an intermediate memory between two memories having large difference between their speeds of operation. Cache memory may thus be between a hard disk and the RAM. It may also be inserted between CPU and RAM to hold the most frequently used data and instructions. Communicating with devices with a cache memory in between enhances the performance of a system significantly.

It is a common observation that at a particular time interval, references to memory tend to be confined within a few localized areas in memory. This phenomenon is known as *locality of reference.* Its illustration can be given by making use of control structure like 'loop'. Cache memories exploit this situation to enhance the overall performance.

Whenever a loop is executed in a program CPU executes the loop repeatedly, the same subroutine is called repeatedly. Thus loops and subroutines tend to localize the references to memory for fetching instructions. To a lesser degree, memory references to data also tend to be localized. Table lookup procedure repeatedly refer to that portion in memory, where the table is stored. Iterative procedures refer to common memory locations and array of numbers are confined within a local portion of memory. The result of these observations is locality of reference property.

The fundamental idea of cache organization is that by keeping the most frequently accessed instructions and data in the fast cache memory, the average memory access time will reach near to access time of cache.

Basic Operation of Cache

Whenever CPU needs to access the memory, cache is examined. If the word is found in the cache, it is read from the fast memory. If the word is not found in cache, main memory is accessed to read the word. A block of words just accessed by CPU is then transferred from main memory to cache memory.

Performance of Cache Memory

The performance of cache memory is frequently measurable in terms of a quality 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 cache it is called a 'miss'.

Hit ratio is a ratio of hits to misses. High hit ratio signifies validity of "locality of reference". If the hit ratio is high enough then most times the CPU accesses the cache instead of main memory, the average access time is closer to access time of fast cache memory.

The characteristic of cache memory is that it is very fast. So very little or no time is wasted when searching for words in the cache. The transformation of data from main memory to cache memory is referred to as a mapping process.

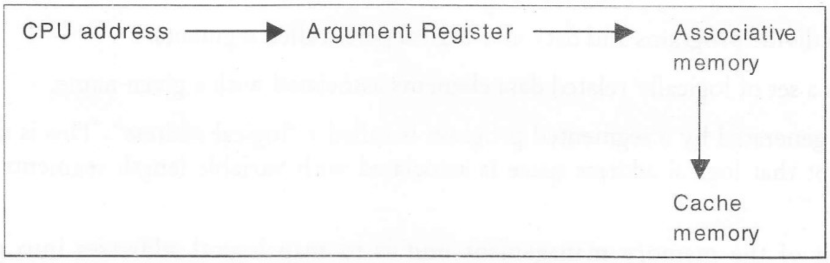
Three types of mapping procedures are there:

1. Associative Mapping
2. Direct Mapping
3. Set Associative Mapping

10.5.1 Associative Mapping

The fastest and most flexible cache organization uses associative mapping. The associative memory stores both the address and content of memory word. This permits any location in cache to store any word in main memory.

Example:



CPU address is first placed in argument register and then associative memory is searched for the match of the above address. If address is found in it somewhere, it has to be placed in cache memory immediately.

If cache memory has no vacant space for storage of new information, in such a case vacancy is created using page replacement policy.

10.5.2 Direct Mapping

Associative memories are expensive compared to RAMs because of added logic associated with each cell.

In general case, there are 2^K words in cache memory and 2^n words in main memory. The n-bit memory address is divided into two fields. K bits for index field and n-k bits for long field. The direct mapping cache organization uses n-bit address to access main memory and k-bit index to access the cache. Each word in cache consists of data word and its associated tag.

When a new word is first brought into cache, the tag bits are stored alongside the data bits. When CPU generates a memory request, the index field is used for the address to access cache. The tag field of CPU address is compared with the tag in word read from the cache. If the two tags match, there is a hit and the desired data word is in cache. If there is no match, there is a miss and the required word is read from main memory. It is then stored in cache together with the new tag, replacing the previous value.

Disadvantage of Direct Mapping

The disadvantage of direct mapping is that the hit ratio can drop considerably if two or more words whose addresses have the same index but different tags are accessed repeatedly.

10.5.3 Set Associative Mapping

It is a more general method that includes pure associative and direct mapping as special cases. It is an improvement over the direct mapping organization in that each word of cache can store two or more words of memory under the same index address.

Each data word is stored together with its tag and the number of tag data items in one word of cache is said to form a set.

10.6 MEMORY MANAGEMENT HARDWARE

10.6.1 Segmentation

Segmentation is a technique to handle problems with respect to program size and logical structure of programs.

It is easier to divide programs and data into logical parts called segments.

A segment is a set of logically related data elements associated with a given name.

The address generated by a segmented program is called a "logical address". This is similar to a virtual address except that logical address space is associated with variable length segments rather than fixed length pages.

The function of the memory management unit is to map logical addresses into physical addresses similar to the virtual memory mapping concept.

10.6.2 Segmented Page Mapping

Segmented Page Mapping is mapping of logical address to physical address.

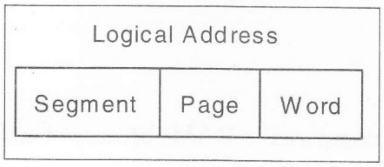


Figure 10.10: Logical Address

A logical address can be divided into three parts.

1. Segment
2. Page
3. Word

Segment field specifies a segment number.

- Page field specifies a page within a segment. A page field of 'P' bits can have maximum number of 2P pages.
- Word field specifies a specific word in the page.
- The length of segment is variable and varies according to number of pages assigned to it.
- The mapping of logical address to physical address requires two tables.

The segment address is like a pointer to page table base address. The page table base is added to the page number given in the logical address. The sum is pointer address to an entry in page table. The value found in page table provides the block number in physical memory. The concatenation provides the block number in physical memory. The concatenation of the block field with the word field produces final physical mapped address.

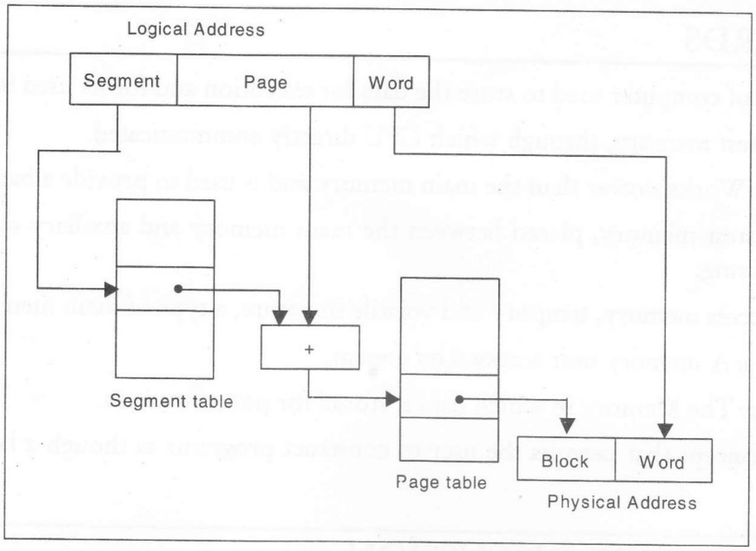


Figure 10.11: Mapping of Logical Address to Physical Address

10.6.3 TLB (Translation Lookaside Buffer)

The two separate mapping tables, if stored in different memory locations, then memory reference from CPU will require three accesses to memory i.e., for segment, page, word. To avoid this multi-access that causes delay, a fast associative memory is used to hold most recently referenced table entries. This type of memory is sometimes called Translation Lookaside Buffer.

Check Your Progress

Fill in the blanks:

1. Memories vary in their design as also in their capacity and .... of operation.
2. .... memories were used as main memory in computers before the semiconductor memories were invented.
3. Most of the .... memories are packaged in chips.
4. A cache memory is an .... memory between two memories having large difference between their speeds of operation.
5. Segmentation is a technique to handle problems with respect to program size and .... structure of programs.

10.7 LET US SUM UP

Memory system is at the heart of a computer system. It is an essential part of any digital computer. In this lesson we have discussed various storage technologies including electrical magnetic and optical store age. Memory management has been discussed. Various types of memories such as main memory, auxiliary memory, associative memory, Cache memory and virtual memory are also discussed. At the end, the concept of paging and demand paging were also discussed.

10.8 KEYWORDS

Memory Unit: Part of computer used to store the data for execution and for its used in future.

Main Memory: Fastest memory, through which CPU directly communicated.

Auxiliary Memory: Works slower than the main memory and is used to provide a backup

Cache Memory: Fastest memory, placed between the main memory and auxiliary memory to increase the speed of processing.

RAM: Random Access memory, tempory and volatile in nature, a type of main memory.

Associative Memory: A memory unit accessed by conent.

Read Only Memory: The Memory in which data is stored for permanent use.

Virtual Memory: concept that permits the user to construct programs as though a large memory space is available.

10.9 QUESTIONS FOR DISCUSSION

1. Explain how cache memory may be organized in a computer.
2. Write short notes on cache memory and virtual memory.
3. Write a short note on associative memory.
4. Explain what is associative memory. Explain how it is used in address mapping in cache memory system.
5. Explain the terms "segmentation" and "paging".
6. What is page fault?
7. What are page replacement policies? Explain any one of them.
8. How does replacement policy affect the performance of virtual memory?

Check Your Progress: Model Answers

1. Speed
2. Ferrite core
3. Semiconductor
4. Intermediate
5. Logical

10.10 SUGGESTED READINGS

Sajjan G. Shiva; *Computer Design and Architecture*; Marcel Dekker

Silvia Melitta Mueller, Wolfgang J. Paul; *Computer Architecture*; Springer

Joseph D. Dumas II; *Computer Architecture*; CRC Press

Nicholas P. Carter; *Schaum's Outline of Computer Architecture*; Mc. Graw-Hill Professional