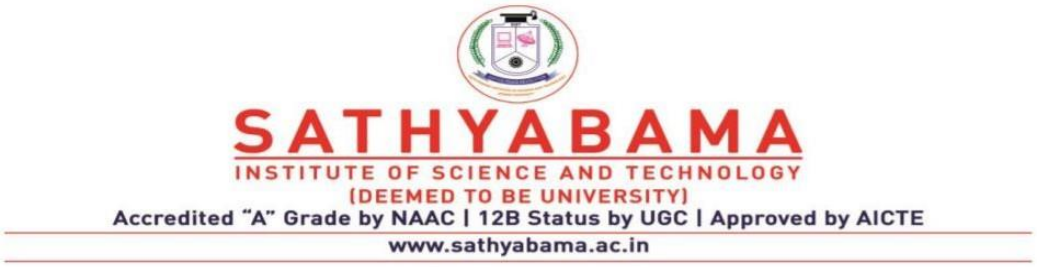


SCHOOL OF COMPUTING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

UNIT – III – SBS1203 – COMPUTER ARCHITECTURE

3.1.MEMORY

A memory is just like a human brain. It is used to store data and instructions. Computer memory is the storage space in the computer, where data is to be processed and instructions required for processing are stored. Memory have divided into small space called cells. Each cell have its unique (own) address.

Address will various from 0 to memory size -1.

Example, If your computer have 32words 32* 1024 = 32,768 words. Memory size will be 0 to 32,767.

Memory organization Hierarchy

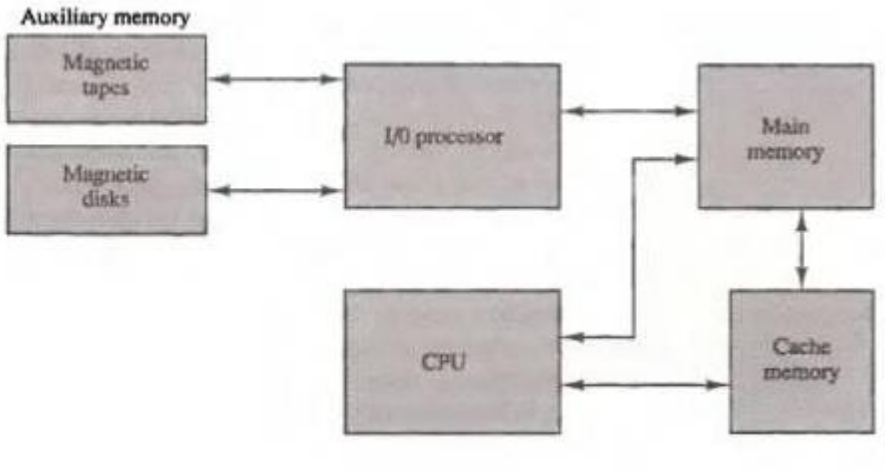


Fig.3.1. Memory Hierarchy in a Computer System

MEMORY TYPES

- Cache Memory
- Virtual Memory
- Auxiliary Memory

- Associative Memory

3.1.1.CACHE MEMORY

Cache memory is a very high speed semiconductor memory which can speed up the CPU. It acts as a buffer between the CPU and the main memory.

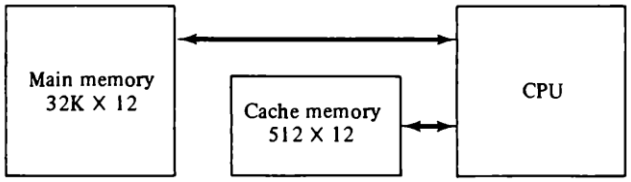


Figure 12-10 Example of cache memory.

Fig.3.2. Example of Cache Memory

Operation of Cache Memory:

When the CPU needs to access memory, the cache is examined. If the word is found in the cache, it is read from the fast memory. If the word addressed by the CPU is not found in the cache, the main memory is accessed to read the word.

A block of words containing the one just accessed is then transferred from main memory to cache memory. The block size may vary from one word (the one just accessed) to about 16 words adjacent to the one just accessed. In this manner, some data are transferred to cache so that future references to memory find the required words in the fast cache memory. The performance of cache memory is frequently measured in terms of a quantity called hit ratio.

The transformation of data from main memory to cache memory is referred to as a mapping process.

Three types of mapping procedures are of practical interest when considering the organization of cache memory:

1. Associative mapping
2. Direct mapping
3. Set-associative mapping

Associative Mapping

The fastest and most flexible cache organization uses an associative memory.

The associative memory stores both the address and content (data) of the memory word.

The address value of 15 bits is shown as a five-digit octal number and its corresponding 12 -bit word is shown as a four-digit octal number. A CPU address of 15 bits is placed in the argument register and the associative memory is searched for a matching address.

if the address is found, the corresponding 12-bit data is read and sent to the CPU.

If no match occurs, the main memory is accessed for the word.

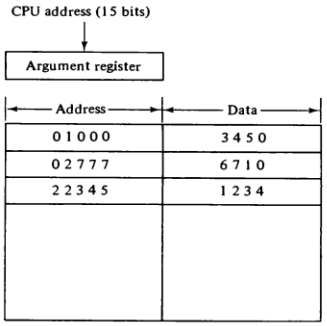


Fig.3.3. Associative Mapping Cache

Direct Mapping

Associative memories are expensive compared to random-access memories because of the added logic associated with each cell.

The CPU address of 15 bits is divided into two fields. The nine least significant bits constitute the index field and the remaining six bits form the tag field.

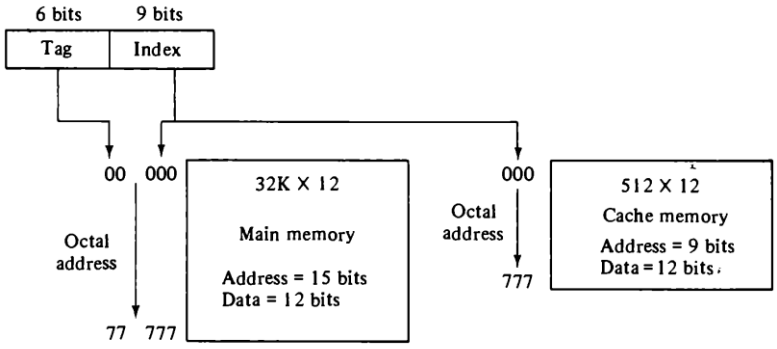


Fig.3.4. Addressing relationship between Main and Cache Memory

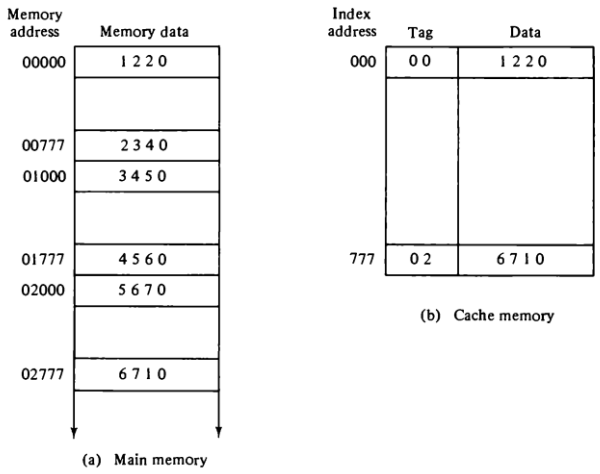


Fig.3.5. Direct Mapping cache organization

The main memory needs an address that includes both the tag and the index bits.

The number of bits in the index field is equal to the number of address bits required to access the cache memory. There are 2' words in cache memory and 2" words in main memory.

The n-bit memory address is divided into two fields: k bits for the index field and n - k bits for the tag field. The direct mapping cache organization uses the n-bit address to access the main memory and the k-bit index to access the cache.

Set-Associative Mapping

A third type of cache organization, called set-associative mapping, 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.

The size of cache memory is 512 x 36. It can accommodate 1024 words of main memory since each word of cache contains two data words. In general, a set-associative cache of set size k will accommodate k words of main memory in each word of cache.

When the CPU generates a memory request, the index value of the address is used to access the cache. The tag field of the CPU address is then compared with both tags in the cache to determine if a match occurs.

The comparison logic is done by an associative search of the tags in the set similar to an associative memory search: thus the name "set-associative." The hit ratio will improve as the set size increases because more words with the same index but different tags can reside in cache.

ADVANTAGES :

Cache memory is faster than main memory.

It consumes less access time as compared to main memory.

It stores the program that can be executed within a short period of time.

It stores data for temporary use.

DISADVANTAGES

Cache memory has limited capacity.

It is very expensive.

3.3. ASSOCIATIVE MEMORY

The memory unit accessed by content is called an associative or Content Addressable memory (CAM). This type of memory accessed simultaneously and parallel on the basis of data content rather than the specific address or location.

When a word is written in an associative memory. No address is given. The memory is capable of finding an empty unused location to store a word.

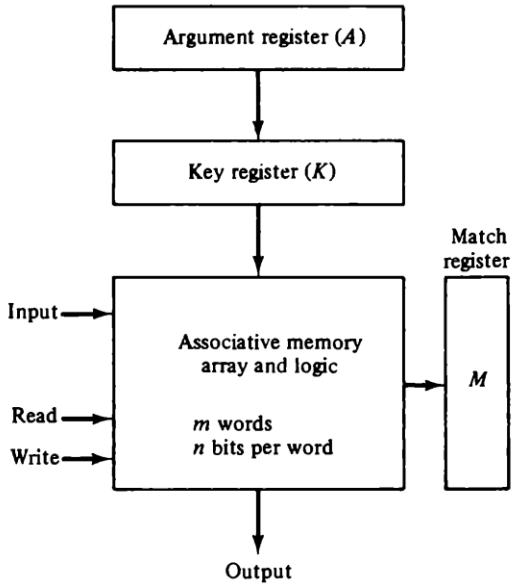


Fig.3.6. Block Diagram of Associative Memory

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

Fig.3.7. Word Comparison

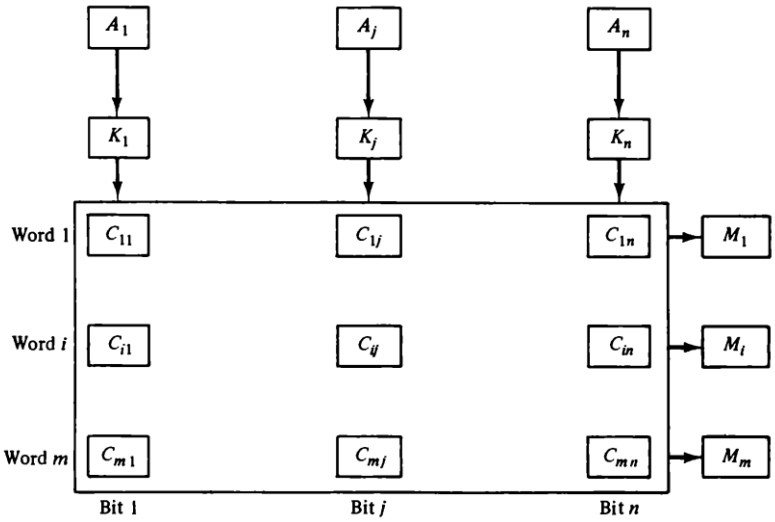


Fig.3.8. Associative Memory of m word, n cells per word

3.4.VIRTUAL MEMORY

Virtual memory is the separation of logical memory from physical memory. This separation provides large virtual memory for programmers when only small physical memory is available.

Virtual memory is used to give programmers the illusion that they have a very large memory even though the computer has a small main memory. It makes the task of programming easier because the programmer no longer needs to worry about the amount of physical memory available.

An address used by a programmer will be called a virtual address, and the set of such addresses the address space.

An address in main memory is called a location or physical address.

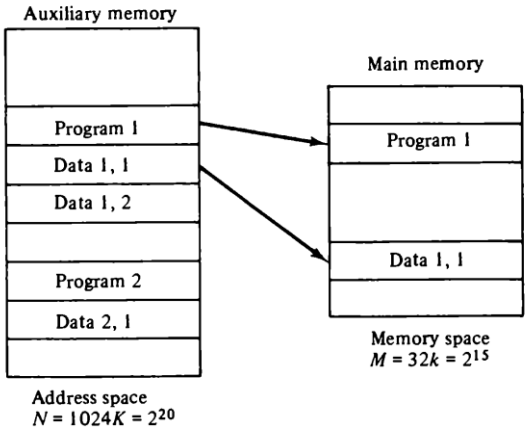


Fig.3.9.Relation Between Address and memory space in a virtual world

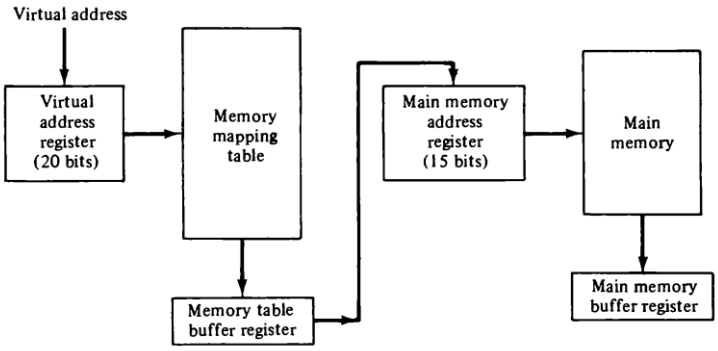


Fig.3.10. Memory table for mapping a virtual address

Consider a computer with a main-memory capacity of 32K words (K = 1024). Fifteen bits are needed to specify a physical address in · memory since 32K = 215• Suppose that the computer has available auxiliary memory for storing 220 = 1024K words.
Thus auxiliary memory has a capacity for storing information equivalent to the capacity of 32 main memories. Denoting the address space by N and the memory space by M, we then have for this example N = 1024K and M = 32K.

3.5. PERIPHERAL DEVICES

Input or output devices that are connected to computer are called peripheral devices. These devices are designed to read information into or out of the memory unit upon command from the CPU and are considered to be the part of computer system. These devices are also called peripherals.
Example: *Keyboards*, *display units* and *printers* are common peripheral devices.

Primary memory holds only those data and instructions on which the computer is currently working. It has a limited capacity and data is lost when power is switched off. It is generally made up of semiconductor device.
These memories are not as fast as registers. The data and instruction required to be processed resides in the main memory. It is divided into two subcategories RAM and ROM.
There are three types of peripherals:

Input peripherals : Allows user input, from the outside world to the computer.
Example: Keyboard, Mouse etc.

Output peripherals: Allows information output, from the computer to the outside world. Example: Printer, Monitor etc.

Input-Output peripherals: Allows both input(from outised world to computer) as well as, output(from computer to the outside world). E.g. Touch screen etc.

3.6.INTERFACES

Interface is a shared boundary between two separate components of the computer

system which can be used to attach two or more components to the system for communication purposes.

There are two types of interface:

- CPU Interface
- I/O Interface

3.7.INPUT-OUTPUT INTERFACE

Peripherals connected to a computer need special communication links for interfacing with CPU. In computer system, there are special hardware components between the CPU and peripherals to control or manage the input-output transfers. These components are called input-output interface units.

They provide communication links between processor bus and peripherals. They provide a method for transferring information between internal system and input-output devices.

3.7.1. MODES OF I/O DATA TRANSFER

Data transfer between the central unit and I/O devices can be handled in generally three types of modes which are given below:

- Programmed I/O
- Interrupt Initiated I/O
- Direct Memory Access

PROGRAMMED I/O

Programmed I/O instructions are the result of I/O instructions written in computer program. Each data item transfer is initiated by the instruction in the program.

Usually the program controls data transfer to and from CPU and peripheral. Transferring data under programmed I/O requires constant monitoring of the peripherals by the CPU.

In the programmed I/O method the CPU stays in the program loop until the I/O unit indicates that it is ready for data transfer. This is time consuming process because it keeps the processor busy needlessly.

This problem can be overcome by using **interrupt initiated I/O**. In this when the interface determines that the peripheral is ready for data transfer, it generates an interrupt. After receiving the interrupt signal, the CPU stops the task which it is processing and service the I/O transfer and then returns back to its previous processing task.

3.8.DIRECT MEMORY ACCESS

Removing the CPU from the path and letting the peripheral device manage the memory buses directly would improve the speed of transfer. This technique is known as DMA. In this, the interface transfer data to and from the memory through memory bus. A DMA controller manages to transfer data between peripherals and memory unit.

Many hardware systems use DMA such as disk drive controllers, graphic cards, network cards and sound cards etc. It is also used for intra chip data transfer in multicore processors.
In DMA, CPU would initiate the transfer, do other operations while the transfer is in progress and receive an interrupt from the DMA controller when the transfer has been completed.

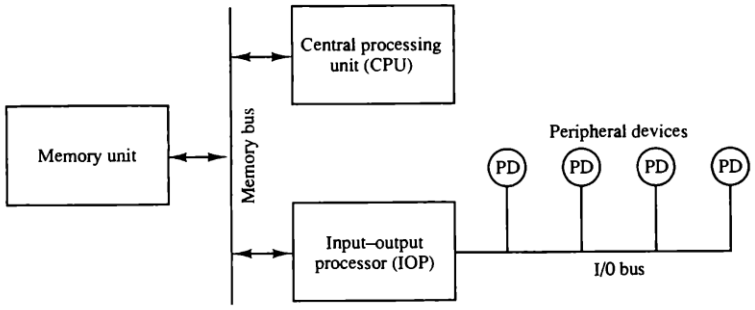


Fig. 3.11. Block Diagram of a Computer with I/O Processor

3.9.PRIORITY INTERRUPT

A priority interrupt is a system which decides the priority at which various devices, which generates the interrupt signal at the same time, will be serviced by the CPU.
The system has authority to decide which conditions are allowed to interrupt the CPU, while some other interrupt is being serviced.

Generally, devices with high speed transfer such as *magnetic disks* are given high priority and slow devices such as *keyboards* are given low priority. When two or more devices interrupt the computer simultaneously, the computer services the device with the higher priority first.

3.9.1.TYPES OF INTERRUPTS

Hardware Interrupts

When the signal for the processor is from an external device or hardware then this interrupts is known as hardware interrupt. Let us consider an example: when we press any key on our keyboard to do some action, then this pressing of the key will generate an interrupt signal for the processor to perform certain action.
Interrupt can be of two types:

Maskable Interrupt

The hardware interrupts which can be delayed when a much high priority interrupt has occurred at the same time.

Non Maskable Interrupt

The hardware interrupts which cannot be delayed and should be processed by the processor immediately.

3.10. SOFTWARE INTERRUPTS

The interrupt that is caused by any internal system of the computer system is known as a **software interrupt**. It can also be of two types:

Normal Interrupt

The interrupts that are caused by software instructions are called **normal software interrupts**.

Exception

Unplanned interrupts which are produced during the execution of some program are called **exceptions**, such as division by zero.

3.11. DAISY CHAINING PRIORITY

The interrupt priority consists of serial connection of all the devices which generates an interrupt signal. The device with the highest priority is placed at the first position followed by lower priority devices and the device which has lowest priority among all is placed at the last in the chain.

In daisy chaining system all the devices are connected in a serial form. The interrupt line request is common to all devices. If any device has interrupt signal in low level state then interrupt line goes to low level state and enables the interrupt input in the CPU.

When there is no interrupt the interrupt line stays in high level state. The CPU respond to the interrupt by enabling the interrupt acknowledge line. This signal is received by the device 1 at its PI input. The acknowledge signal passes to next device through PO output only if device 1 is not requesting an interrupt.

This problem is solved by following mechanism:

- Strobe
- Handshaking

Data is transferred from source to destination through data bus in between.

Strobe Mechanism:

Source initiated Strobe – When source initiates the process of data transfer. Strobe is just a signal.

(i) First, source puts data on the data bus and on the strobe signal.

(ii) Destination on seeing the ON signal of strobe, read data from the data bus.

(iii) After reading data from the data bus by destination, strobe gets OFF.

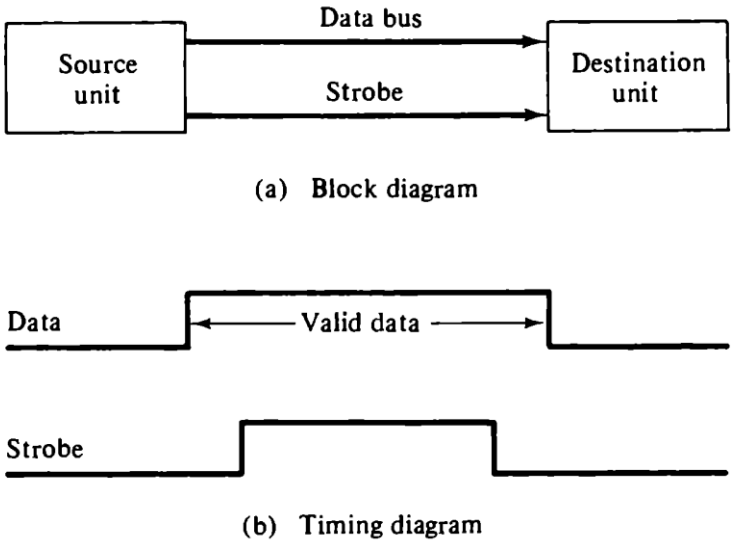


Fig.3.12. Source Initiated strobe

Destination initiated signal – When destination initiates the process of data transfer.

(i) First, the destination ON the strobe signal to ensure the source to put the fresh data on the data bus.

(ii) Source on seeing the ON signal puts fresh data on the data bus.

(iii) Destination reads the data from the data bus and strobe gets OFF signal.

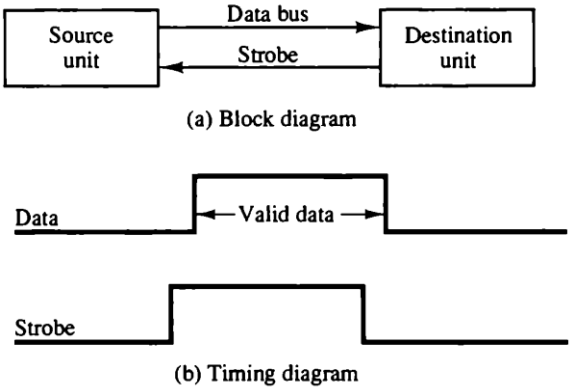


Fig.3.13. Destination Initiated Signal

Problems faced in Strobe based asynchronous input output

In Source initiated Strobe, it is assumed that destination has read the data from the data bus but their is no surety. In Destination initiated Strobe, it is assumed that source has put the data on the data bus but their is no surety. This problem is overcome by **Handshaking**.

Handshaking

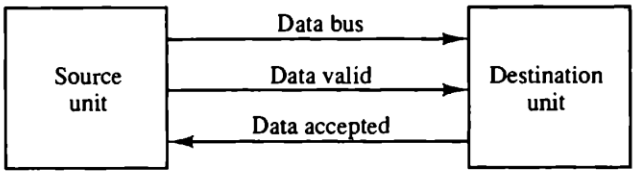
Handshaking Mechanism

When source initiates the data transfer process. It consists of signals:
DATA VALID: if ON tells data on the data bus is valid otherwise invalid.
DATA ACCEPTED: if ON tells data is accepted otherwise not accepted.

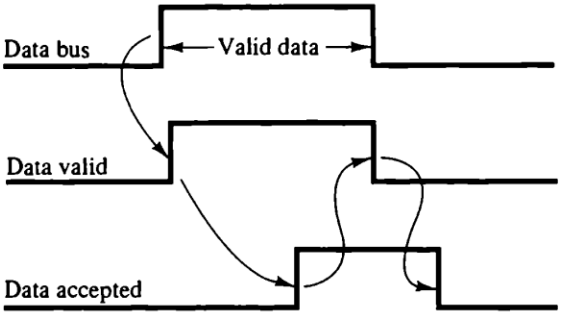
(i) Source places data on the data bus and enable Data valid signal.

(ii) Destination accepts data from the data bus and enable Data accepted signal.
(iii) After this, disable Data valid signal means data on data bus is invalid now.
(iv) Disable Data accepted signal and the process ends.

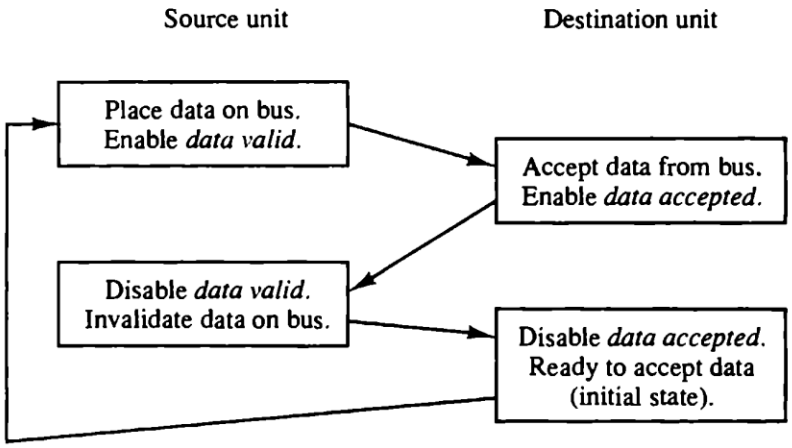
Now there is surety that destination has read the data from the data bus through data accepted signal.



(a) Block diagram



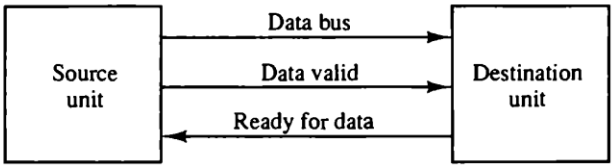
(b) Timing diagram



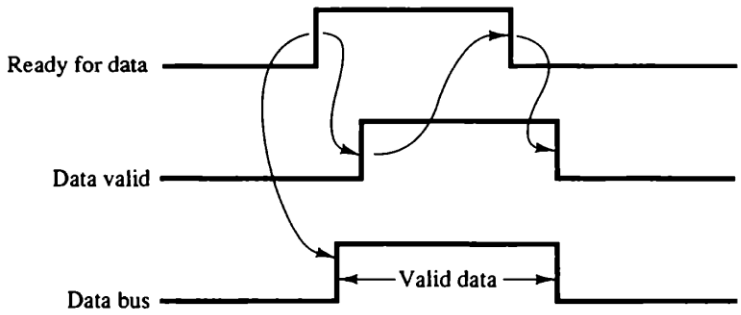
(c) Sequence of events

Fig.3.14. Source Initiated transfer using handshaking

(i) When destination is ready to receive data, Request for Data signal gets activated.
(ii) source in response puts data on the data bus and enabled Data valid signal.
(iii) Destination then accepts data from the data bus and after accepting data, disabled Request for Data signal.
(iv) At last, Data valid signal gets disabled means data on the data bus is no more valid data. Now there is surety that source has put the data on the data bus through data valid signal.



(a) Block diagram



(b) Timing diagram

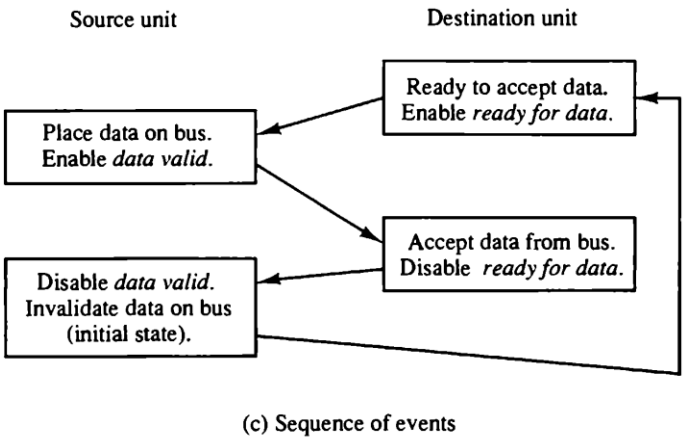


Fig.3.15. Destination Initiated transfer using Hanshaking

3.12.SERIAL COMMUNICATION

Serial Communication is a communication technique used in telecommunications wherein data transfer occurs by transmitting data one bit at a time in a sequential order over a computer bus or a communication channel. It is the simplest form of communication between a sender and a receiver.

A data communication processor is an I/O processor that distributes and collects data from numerous remote terminals connected through telephone and other communication lines to the computer. It is a specialized I/O processor designed to communicate with data communication networks.

Such a communication network consists of variety of devices such as printers, display devices, digital sensors etc. serving many users at once.

The data communication processor communicates with each terminal through a single pair of wire. It also communicates with CPU and memory in the same manner as any I/O processor does.

MODEM

In a Data Communication Network, the remote terminals are connected to the data communication processor through telephone lines or other wires.

Such telephone lines are specially designed for voice communication and computers use them to communicate in digital signals, therefore some conversion is required. These conversions are called modem (modulator-demodulator). A modem converts digital signal into audio tones to be transmitted over telephone lines and also converts audio tones into digital signal for machine use.

Modes Of Transmission

- Simplex
- Half Duplex
- Full Duplex

Types of Protocols

Character Oriented Protocol

It is based on the binary code of character set. The code is mostly used in ASCII. It includes upper case and lower case letters, numerals and variety of special symbols. The characters that control the transmission is called communication control characters.

Bit Oriented Protocol

It does not use characters in its control field and is independent of any code. It allows the transmission of serial bit stream of any length without the implication of character boundaries.