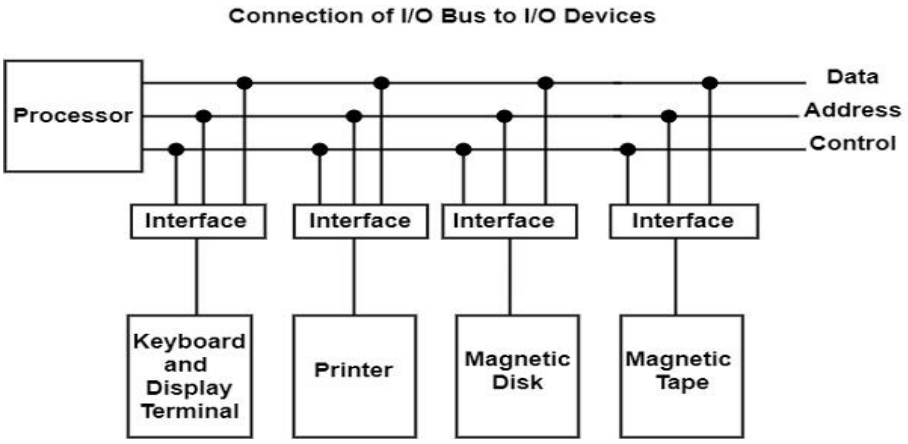
[Unit-5] Input Output Organization

❖ Memory Bus

I/O Bus and Interface Modules

The I/O bus is the route used for peripheral devices to interact with the computer processor. A typical connection of the I/O bus to I/O devices is shown in the figure.



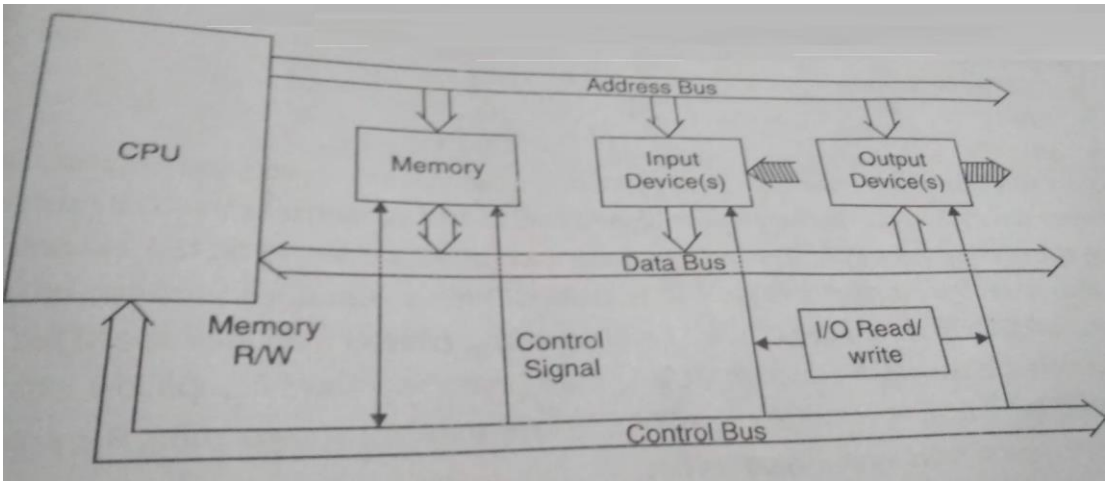
[Fig. Block Diagram of Memory Buses]

- The I/O bus includes data lines, address lines, and control lines.
- In any general-purpose computer, the magnetic disk, printer, and keyboard, and display terminal are commonly employed.
- Each peripheral unit has an interface unit associated with it.
- Each interface decodes the control and address received from the I/O bus.
- It can describe the address and control received from the peripheral and supports signals for the peripheral controller.
- It also conducts the transfer of information between peripheral and processor and also integrates the data flow.
- The I/O bus is linked to all peripheral interfaces from the processor.
- The processor locates a device address on the address line to interact with a specific device. Each interface contains an address decoder attached to the I/O bus that monitors the address lines.
- When the address is recognized by the interface, it activates the direction between the bus lines and the device that it controls.
- The interface disables the peripherals whose address does not equivalent to the address in the bus.

❖ An interface receives any of the following four commands / Function−

- **Control** − A command control is given to activate the peripheral and to inform its next task. This control command depends on the peripheral, and each peripheral receives its sequence of control commands, depending on its mode of operation.
- **Status** − A status command can test multiple test conditions in the interface and the peripheral.
- **Data Output** − A data output command creates the interface counter to the command by sending data from the bus to one of its registers.
- **Data Input** − the data input command is opposite to the data output command. In data input, the interface gets an element of data from the peripheral and places it in its buffer register.

❖ Address bus, Data bus ,Control bus/lines, I/O buses



What is the Address Bus?

→The address bus is used by the CPU to send the address of the memory location or the input/output port that is to be accessed at the instant. It is a unidirectional bus i.e. the address can be transferred in one direction only and that is form CPU to the required port or location.

Whether it is a read operation or writes operation the CPU calculates the address of the required data and sends it on the data bus for the execution of the required operation. The maximum number of memory locations that can be accessed in a system is determined by the number of lines of an address bus. An address bus of n lines can be addressed at the most 2n locations directly. Thus a 16-bit address bus can allow access 2 16 bits or 64 K Byte of memory.

What is Data Bus?

→A data bus is used to carry the data and instructions from the CPU to memory and peripheral devices and vice versa. Thus it is a bidirectional bus. It is one of most important parts of the connections to the CPU because every program instruction and every byte of data must travel across the bus at some point.

The size of the data bus is measured in bits. The data bus size has much influence on the computer architecture because the important parameters of it like word size, the quantum of data etc. are determined and manipulated by the size of the data bus.

Generally, a microprocessor is called n-bit processors. Thus as the CPU became more advanced, the data bus grew in size.

A 64-bit data bus can transfer 8 bytes in every bus cycle and thus its speed is much faster as compared to the 8-bit processor that can transfer one byte in every bus cycle.

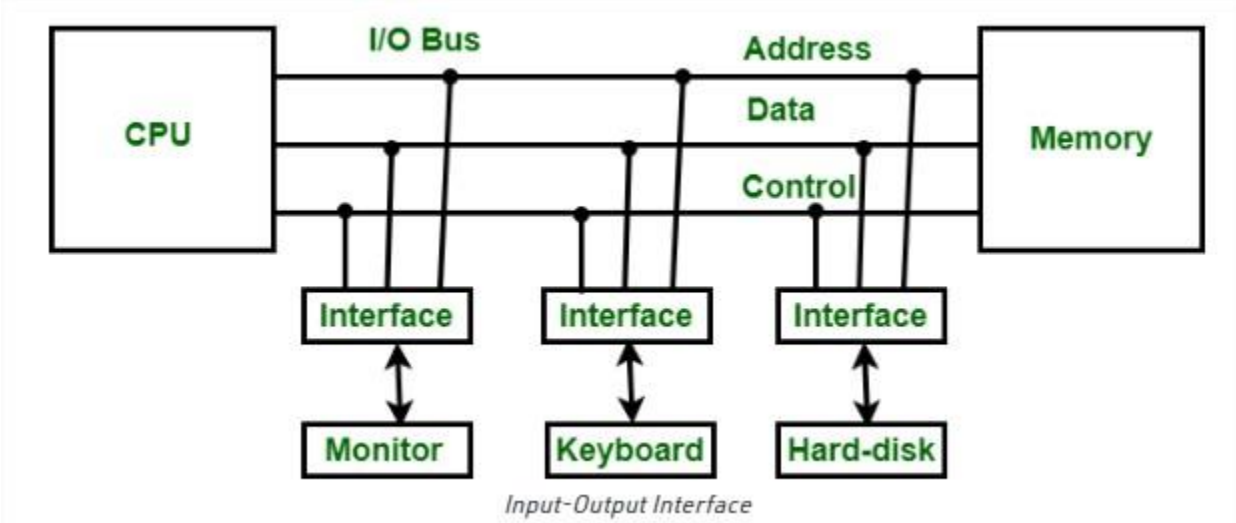
What is a Control Bus?

→A control bus contains various individual lines carrying synchronizing signals that are used to control. Various peripheral devices connected to the CPU. The common signals that are transferred to the control bus from CPU to devices and vice versa are memory read, memory writes, I/O read, I/O write etc.

Signals are designed, keeping in mind, the design philosophy of the microprocessor and the requirement of the various devices connected to the CPU. So different types of the microprocessor have different control signals. See Below for better understanding.

❖ Concept of Input output Interface

- Input-Output Interface is used as an method which helps in transferring of information between the internal storage devices i.e. memory and the external peripheral device.
- A peripheral device is that which provide input and output for the computer, it is also called Input-Output devices.
- For Example: A keyboard and mouse provide Input to the computer are called input devices while a monitor and printer that provide output to the computer are called output devices.
- Just like the external hard-drives, there is also availability of some peripheral devices which are able to provide both input and output.



Input-Output Interface

Functions of Input-Output Interface:

- It is used to synchronize the operating speed of CPU with respect to input-output devices.
- It selects the input-output device which is appropriate for the interpretation of the input-output device.
- It is capable of providing signals like control and timing signals.
- In this data buffering can be possible through data bus.
- There are various error detectors.
- It converts serial data into parallel data and vice-versa.
- It also converts digital data into analog signal and vice-versa.

❖ Input Output Processor (IOP)

- Below is a block diagram of a computer along with various I/O Processors.
- The memory unit occupies the central position and can communicate with each processor.
- The CPU processes the data required for solving the computational tasks.
- The IOP provides a path for transfer of data between peripherals and memory.
- The CPU assigns the task of initiating the I/O program.
- The IOP operates independent from CPU and transfer data between peripherals and memory.

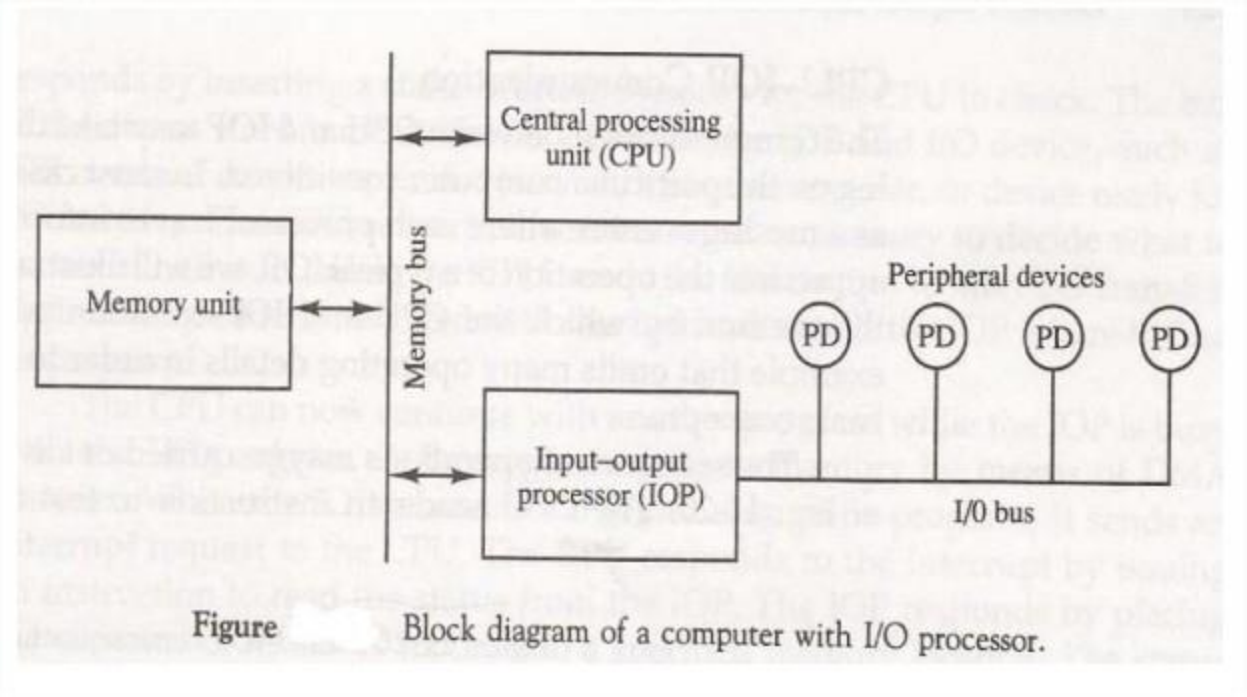


Figure Block diagram of a computer with I/O processor.

- The communication between the IOP and the devices is similar to the program control method of transfer. And the communication with the memory is similar to the direct memory access method.
- In large scale computers, each processor is independent of other processors and any processor can initiate the operation.
- The CPU can act as master and the IOP act as slave processor.
- The CPU assigns the task of initiating operations but it is the IOP, who executes the instructions, and not the CPU.
- CPU instructions provide operations to start an I/O transfer.
- The IOP asks for CPU through interrupt.
- Instructions that are read from memory by an IOP are also called commands to distinguish them from instructions that are read by CPU.
- Commands are prepared by programmers and are stored in memory.
- Command words make the program for IOP.
- CPU informs the IOP where to find the commands in memory

❖Direct Memory Access (DMA)

[a] DIRECT MEMORY ACCESS (DMA)

Direct memory access is a sophisticated input-output technique in which a DMA controller replaces the CPU and takes care of the access of input-output device and the memory, for fast data transfers.

Direct memory access (DMA) is a feature of modern computers that allows certain hardware subsystems within the computer to access system memory for reading and/or writing independently of the central processing unit. Many hardware systems use DMA including disk drive controllers, graphics cards, network cards, and sound cards.

Central processing unit often limited data transfer speed between secondary memory and main memory. This problem can be resolved by removing CPU from the path. After removing CPU, different peripheral manage the memory buses directly, for the improvement of data transfer rate. This special kind of method is known as *Direct Memory Access* (DMA). It is important to note that during the DMA transfer CPU is idle and has no control of the memory bus.

➢ Advantages of DMA

Without DMA, using programmed input/output (PIO) mode, the CPU is typically fully occupied for the entire duration of the read or write operation and is thus unavailable to perform other work. With DMA, the CPU would initiate the transfer, do other operations while the transfer is in progress, and receive an interrupt from the DMA controller once the operation has been done. This is especially useful in real-time computing applications

➢ Working of DMA

DMA controller control the buses and the data transfer directly take place between the I/O device and memory.

CPU bus signal for DMA transfer is shown in following fig.6.2.

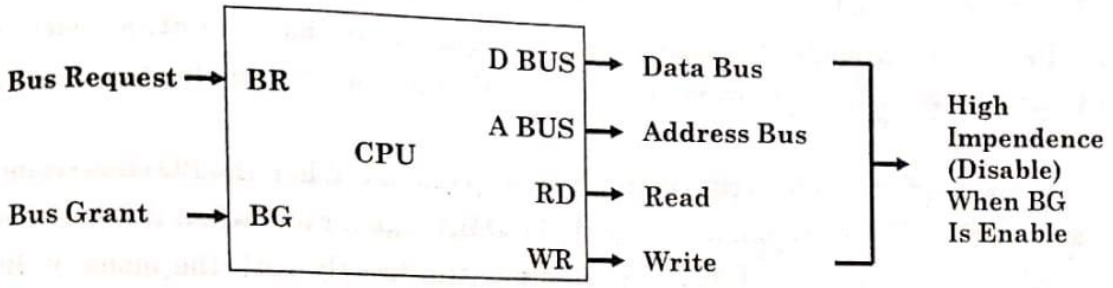


Fig. 6.2

The fig. 6.2 shows the two control signal that provides the DMA transfer. The bus request (BR) input is used by the DMA to request the CPU to release the control of buses. When this input is active the CPU terminates the execution of current instruction and places the address bus, data bus and read, write lines into a high impedance state.

Now CPU activates the bus grant (BG) output to inform the external DMA that the buses are in the high impedance state. Then DMA take control of all the bus and memory transfer without processor. When DMA terminates the transfer it disables the bus request line. Then CPU again takes control of the bus and returns its normal operation.

- There are two methods by which transfer take place in DMA
  - [a] Burst Transfer
  - [b] Cycle Stealing

[a] Burst Transfer

In this transfer a block consisting of number of memory word is transfer in a continuous burst. This mode of transfer is needed for fast devices search as magnetic disk, where data transmission can not be stop till entire block is transfer.

[b] Cycle Stealing

This technique allows the DMA controller to transfer one data word at a time, after which it must return the control of the buses to the CPU

❖DMA Controller

➢ **DMA Controller**

Fig. 6.3 shows the block diagram of DMA controller. The unit communicates with the CPU via the data bus and control lines. The register in the DMA are selected by CPU through the address bus by unable the DS (DMA Select) and RS (Register Select).

The RD (Read) and WR (Write) inputs are bidirectional. When the BG (Bus Grant) input is '0', the CPU can communicate with the DMA register and when BG is '1' then CPU release the buses and DMA can communicate directly with the memory. By activation the RD or WR control.

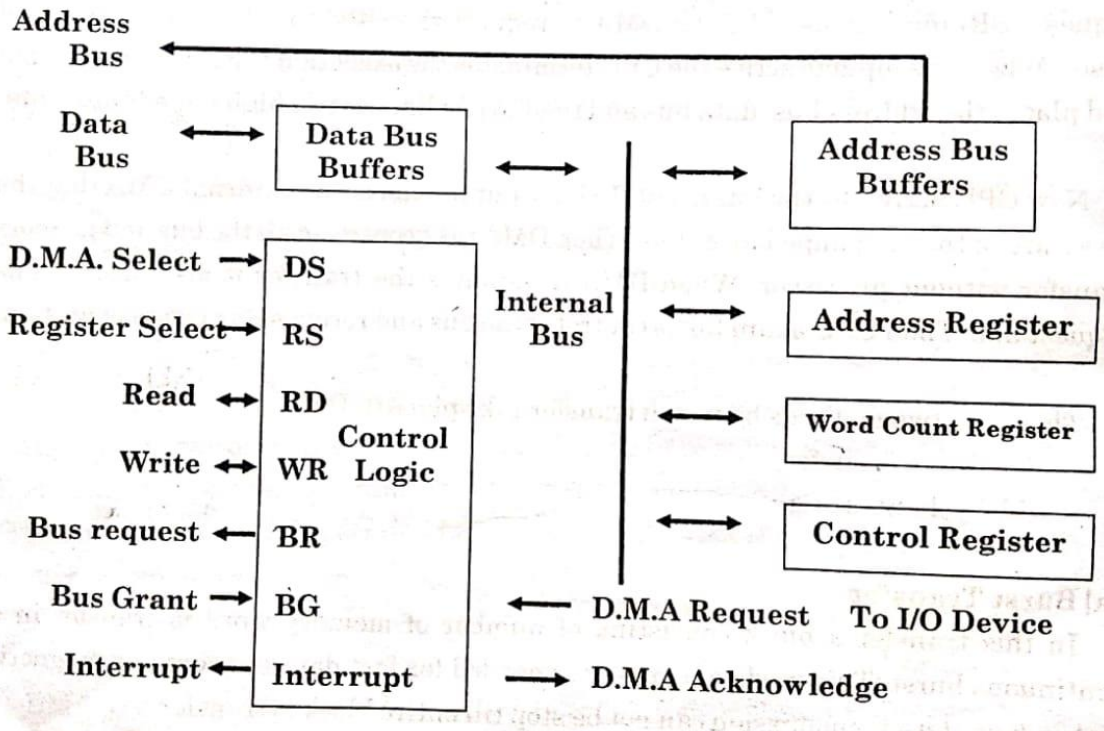


Fig. 6.3 : Block Diagram of DMA Controller

- DMA controller has three register as follows:

[a] Address Register

The address register contain the address to specify the desire location in memory this register is incremented after each word is transfer to memory.

[b] Word Count Register

This register holds the number of words to be transfer this register is decremented by one after each word transfer.

[c] Control Register

This register specifies the mode of transfer.

All the register in the DMA appear to the CPU as I/O interface register. Thus the CPU can read or write into the DMA registers under program control via the data bus.

DMA is first initialized by the CPU After that CPU send the following information via data bus.

- Starting address of the memory block where data are available (for read) or where data are to be stored (for write).
- The word count, which is the number of words in the memory block.
- Control to specify the mode of transfer like read or write.
- A control to start DMA transfer.

========== ++++ ==========