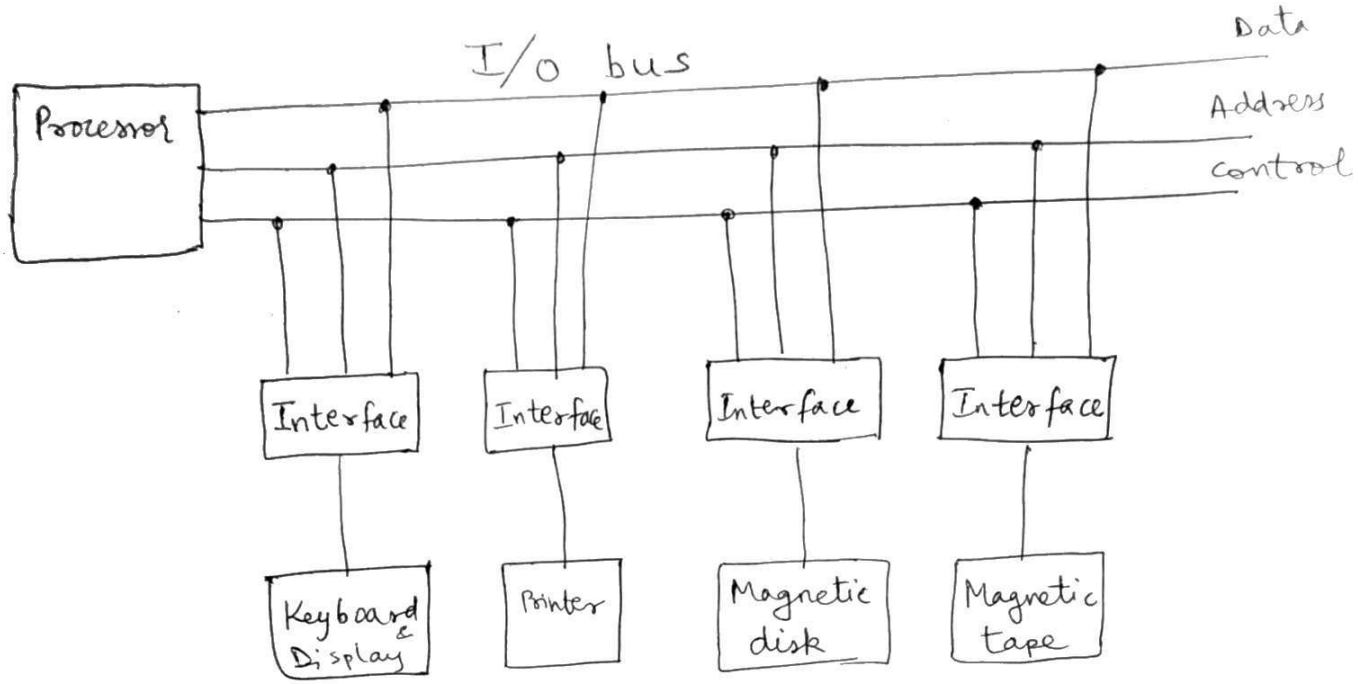
Chapter -11

Input - Output Organization :→

Connection of I/O bus to input - output devices is shown in the fig 1 →



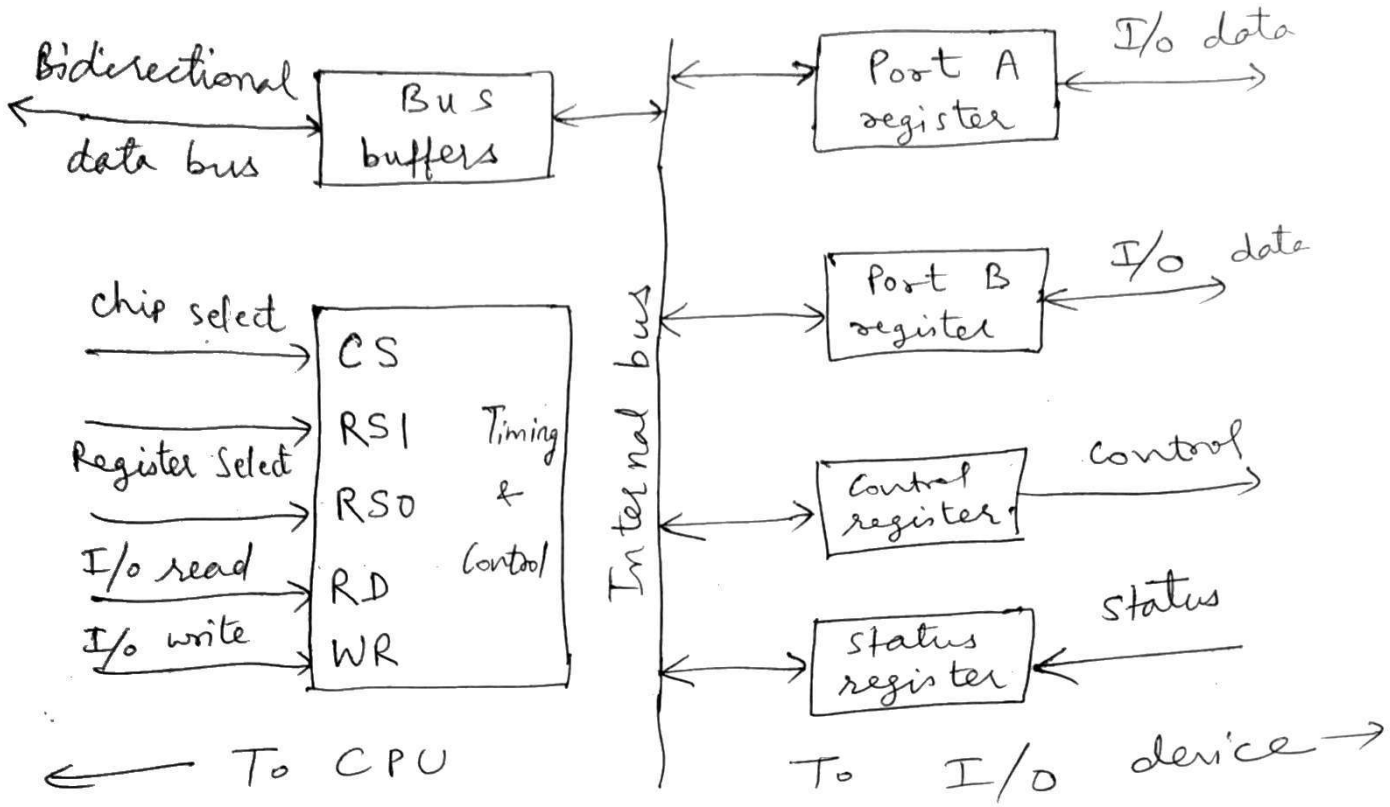
The main function of input - output interface circuit are data conversion, synchronization and device selection. I/O bus consists of data lines, address lines and control lines.

When the address is made available in the address lines, the processor provides a function code ie I/O command in the control lines.

A control command is issued to activate the peripheral and to inform it what to do.

A status command is used to test various status conditions in the interface and the peripheral.

Example of an I/O interface unit is shown



CS	RS1	RS0	Register selected
0	X	X	None: data bus in high-impedance
1	0	0	Port A register
1	0	1	Port B register
1	1	0	Control register
1	1	1	Status register

Two data register are available called as ports, i) Control register, a status register, bus buffers and timing & control circuits

Data bus is used to communicate with CPU & interface

Chip select & register select inputs determine the address assigned to the interface.

Transfer of data, control and status information is always via the common data bus.
Data is transfered to & from ports A & B registers
External ckt is provided to enable the chip select (CS) i/p when interface is selected by the address bus.
Two registers select i/ps RS1 & RS0 are usually connected to the two least significant lines of the address bus.

Asynchronous Data Transfer

Asynchronous data transfer between two independent units requires that control signals be transmitted b/w the communicating units to indicate the time at which data is being transmitted. This is done by →

Strobe pulse :→ is supplied by one of the units It indicates to the other unit when transfer has to occur.

Another method commonly used is to accompany each data item being transfered with a control signal that indicates the presence of data in the bus. The unit receiving the data item responds with another control signal to acknowledge receipt of the data and it is called

Handshaking.

Strobe control :→ method of asynchronous data transfer employs a single control line to time each transfer. The strobe may be activated by either the source or the destination unit, shown→

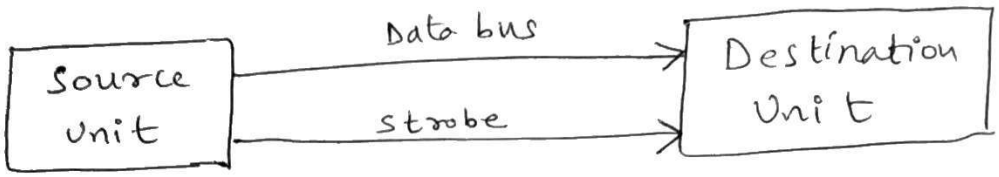


fig (a) block dig

This block dig shows a source-initiated transfer Data bus carries the binary information source unit to the destination unit. Bus has multiple lines to transfer an entire byte or word.
strobe is a single line that informs the destination unit when a valid data word is available in the bus.

Timing dig :→

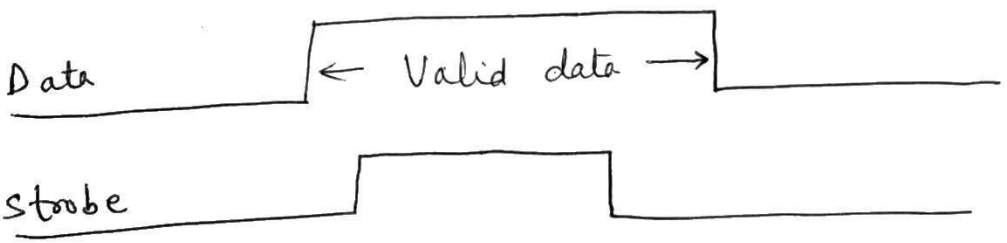
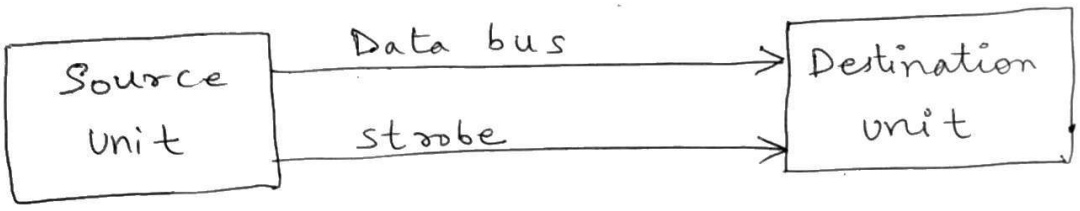


fig (b)

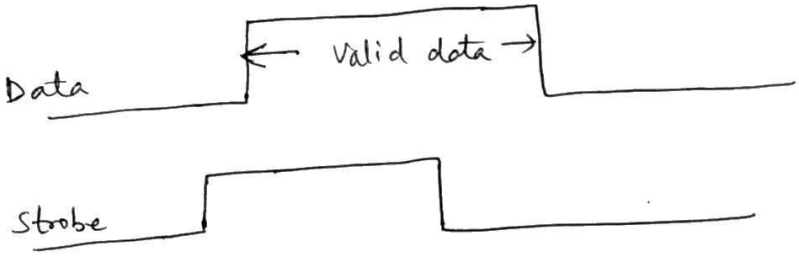
The source unit first places the data on the data bus. After a brief delay to ensure the data settle to a steady value, the source activates the strobe pulse.
The information on the data bus and strobe signal remain in the active state for a sufficient time period to allow the destination unit to receive the data.

Source removes the data from the bus after the strobe pulse is disabled from source.
New valid data will be available only after the strobe is enabled again.

(ii) Destination - initiated strobe for data transfers



(a) block dig



(b) Timing dig

Here destination unit activates the strobe pulse and informs the source to provide data. Source unit responds by placing the requested binary information on the data bus.
The destination unit disables the strobe, then source removes the data from the bus after a predetermined time interval.

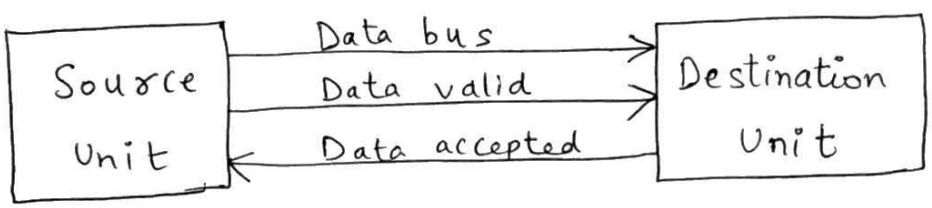
Handshaking :→

The disadvantage of the strobe method is that the source unit that initiates the transfer has no way of knowing whether the destination unit has actually received the data item that was placed in the bus, similarly for destination unit.

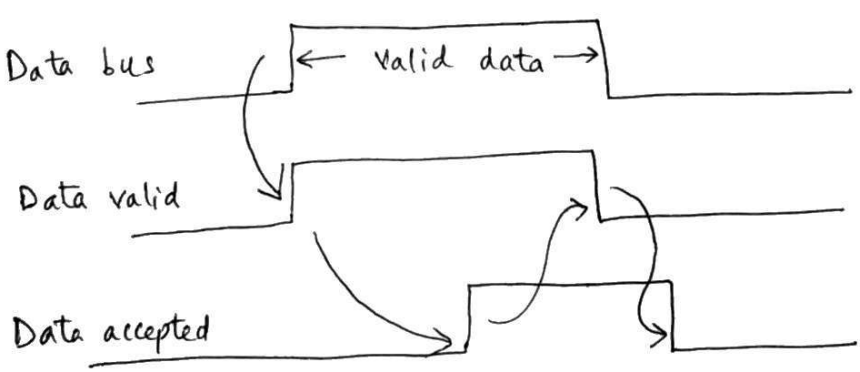
Hand shake method solves this problem by introducing a second control signal that provides a reply to the unit that initiates the transfer.

A source initiated transfer using handshaking is shown: →

block dig



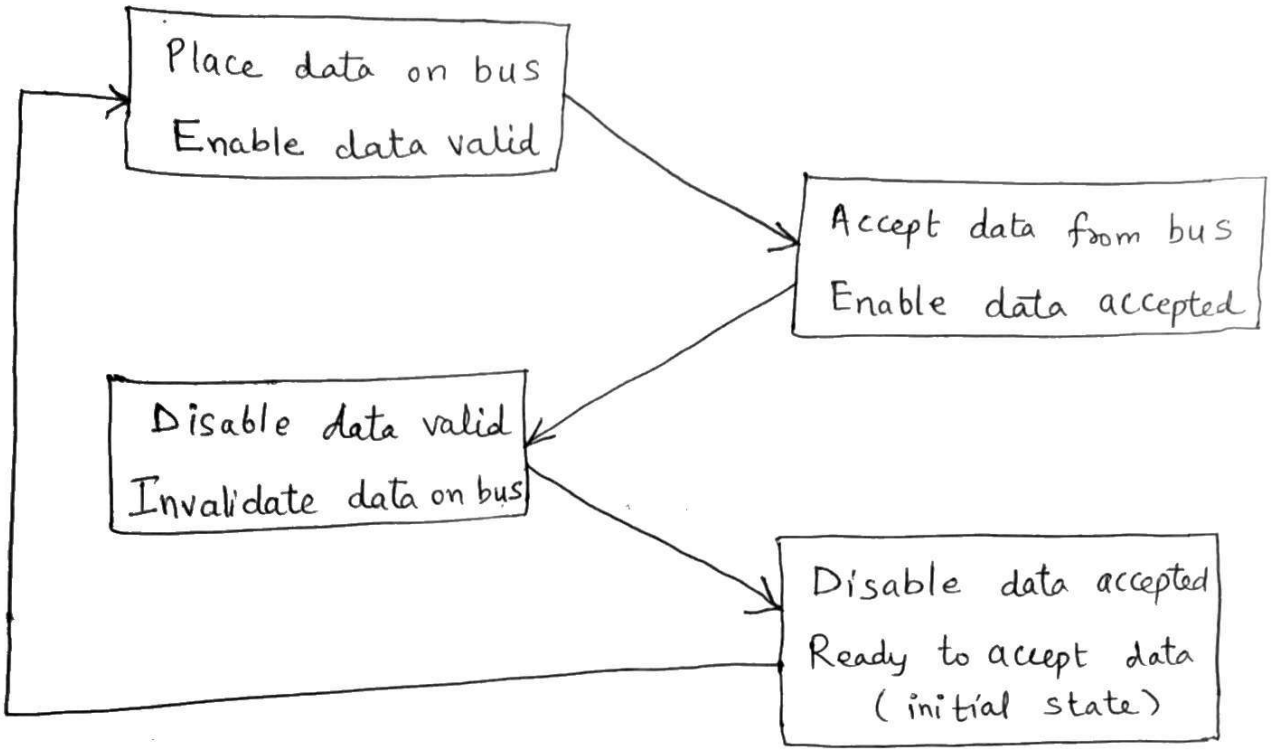
Timing dig



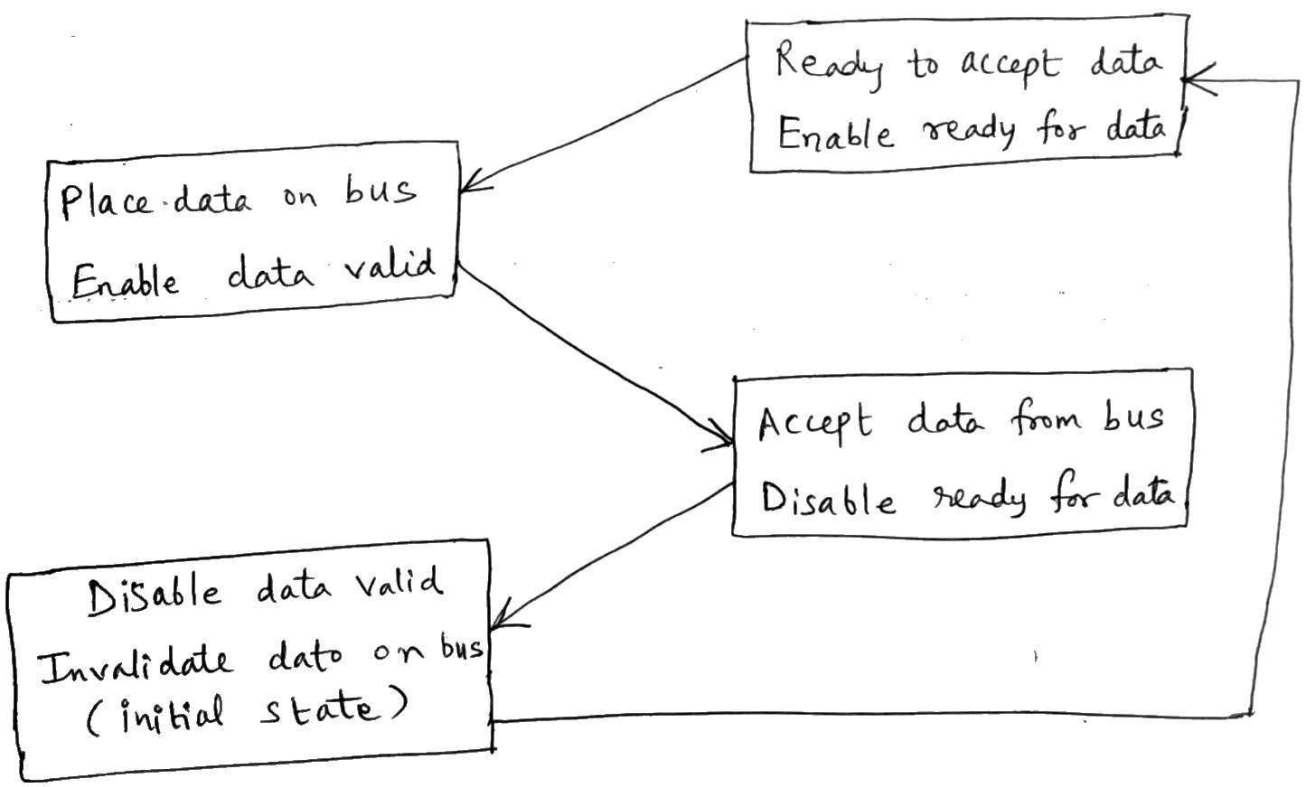
Two handshaking lines are (i) data valid, which is generated by the source unit,
(ii) data ~~valid~~ accepted, generated by the destination unit.

Sequence of Events are shown in fig →

Source unit initiates the transfer by placing the data on the bus and enabling its data valid signal. Destination unit activates the data accepted signal after it accepts the data from the bus. Source unit then disables its data valid signal



(2) Destination - initiated transfer handshaking :→



Direct Memory Access (DMA)

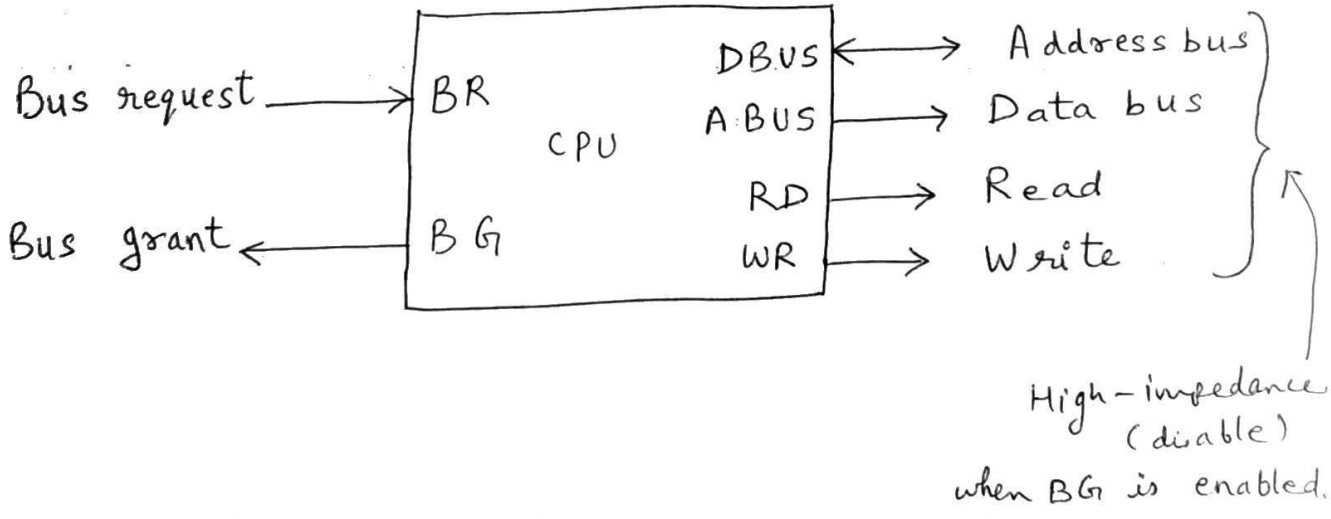
The transfer of data between a fast storage device such as magnetic disk and memory is often limited by the speed of the CPU.

The speed of transfer would improve if we remove the CPU from the path and letting the peripheral device manage the memory bus directly.

This transfer technique is called Direct Memory Access

→ In DMA transfer, the CPU is idle & has no control of the memory buses. DMA controller takes over the buses to manage the transfer directly b/w the I/O device & memory.

→ Two control signals in the CPU that facilitate the DMA transfer
(i) The bus request (ii) Bus grant is shown →

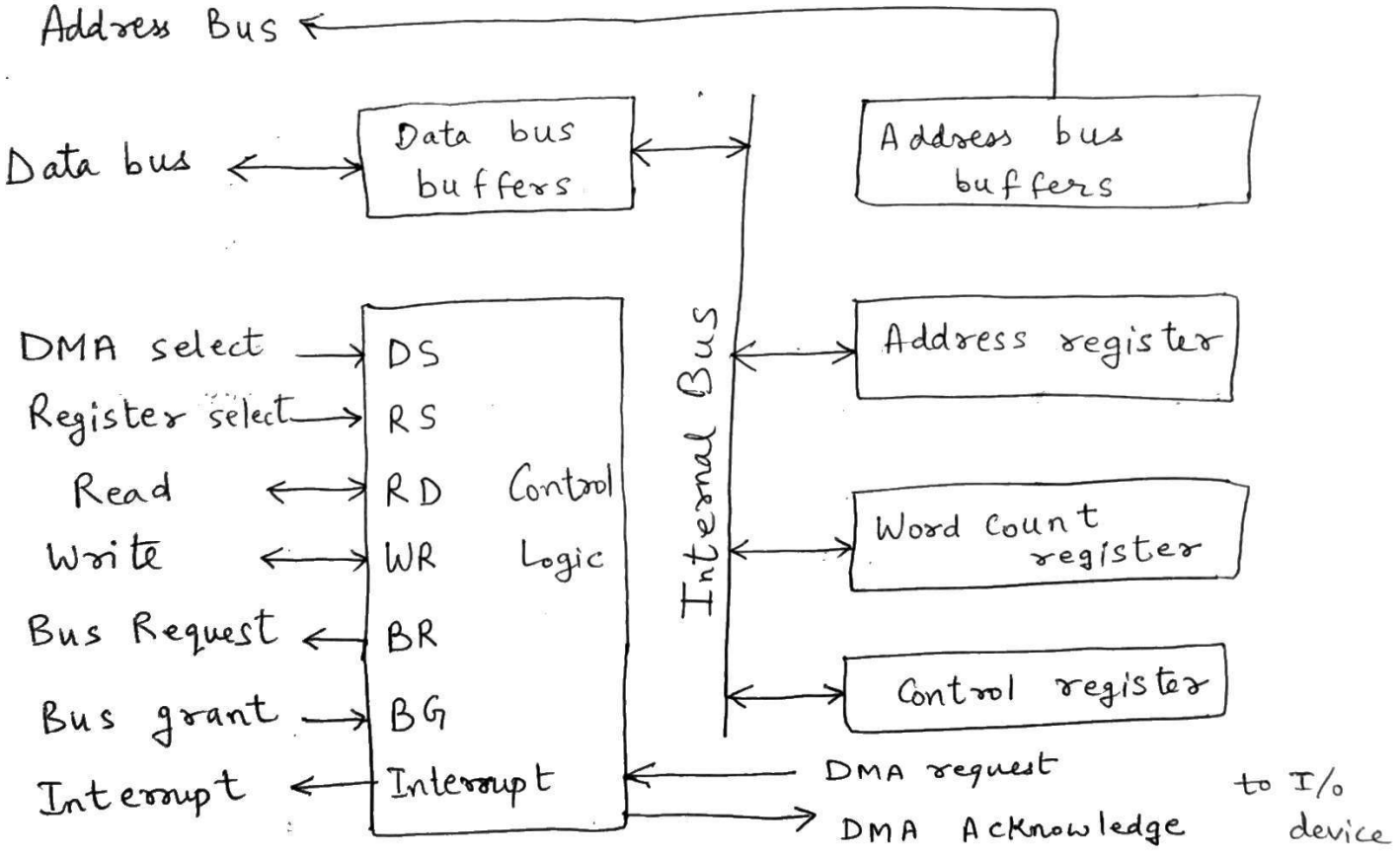


Bus request (BR) signal is used to request the CPU for control of the buses, BR is active, CPU terminates the execution of the current instruction and places the address bus, data bus, the read & write lines into a high-impedance state. (open ckt o/p)

Bus grant (BG) signal o/p is active to inform the external DMA that the buses are in high-impedance state & DMA can now take control of the buses to conduct memory transfers

When DMA takes control of the bus system, it communicates directly with the memory by <u>burst transfer</u>, . In DMA burst transfer, a block sequence consisting of a no. of memory words is transfered in a continuous burst.

<u>DMA Controller</u> :→ Block dig is shown below→



DMA controller communicates with CPU via the data bus and control lines.

DMA select (DS) and Register select (RS) i/ps enable the address bus for selecting registers in DMA

When BG i/p = 0, CPU can communicate with DMA registers through the data bus to read from or write to the DMA registers.

When BG = 1, the buses of the CPU are idle. DMA can communicate directly with the memory and activates RD or WR control.

DMA controller has 3 registers :→

1) Address register — contains an address to specify the desired location in memory

2) Word count register — holds the no. of words to be transferred.

3) Control register - specifies the mode of transfer