DMA Controller (8237)

In MP based systems, data transfer can be controlled by either software or hardware.

To transfer data MP has to do following tasks :-

1.) To fetch the instruction
2.) To decode the instruction
3.) To execute the instruction

To carry out these tasks MP requires considerable time. So this method of data transfer is not suitable for large data transfers such as data transfer from magnetic disk to

memory. In such situations, hardware controlled data transfer technique is used.

This hardware is known as direct memory access controller or DMA Controller.

In this technique, external device is used to control data transfer. External device generates address and control signals required to control data transfer and allows peripheral device to directly access the memory.

Hence this technique is refered to as DMA and external device which controls the data transfer is refered to as DMA Controller.

DMA Controller operates in 2 cycles:-

1.) DMA Idle Cycle

2.) DMA Active Cycle

1.) DMA Idle Cycle :-

When the system is turned on, the switches are in the 'A' position, so the buses are connected from the µP to the system memory and peripherals. µP then executes the program until it needs to read a block of data from the disk.

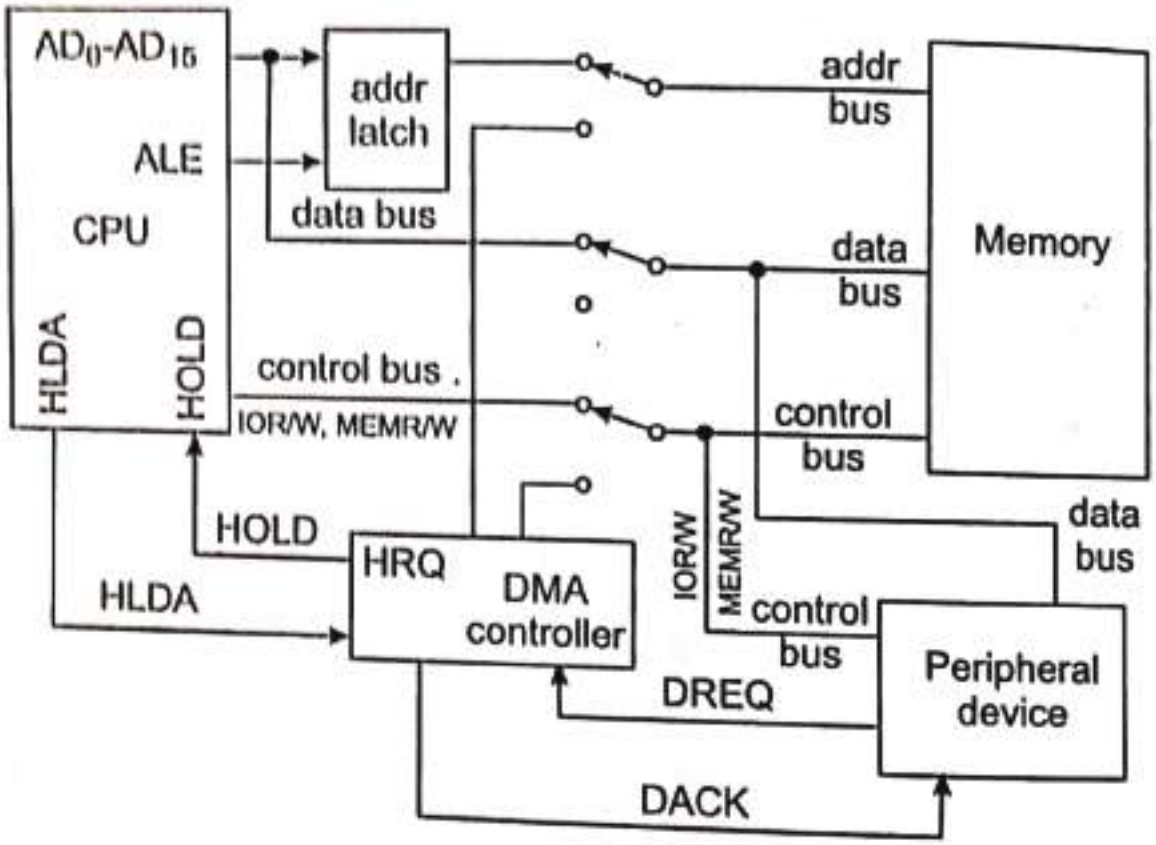
To read a block of data from the disk, µP sends a series of commands to the disk controller device telling it to search and read the desired block of data from the disk.

When disk controller is ready to transfer first byte of data from disk, it sends

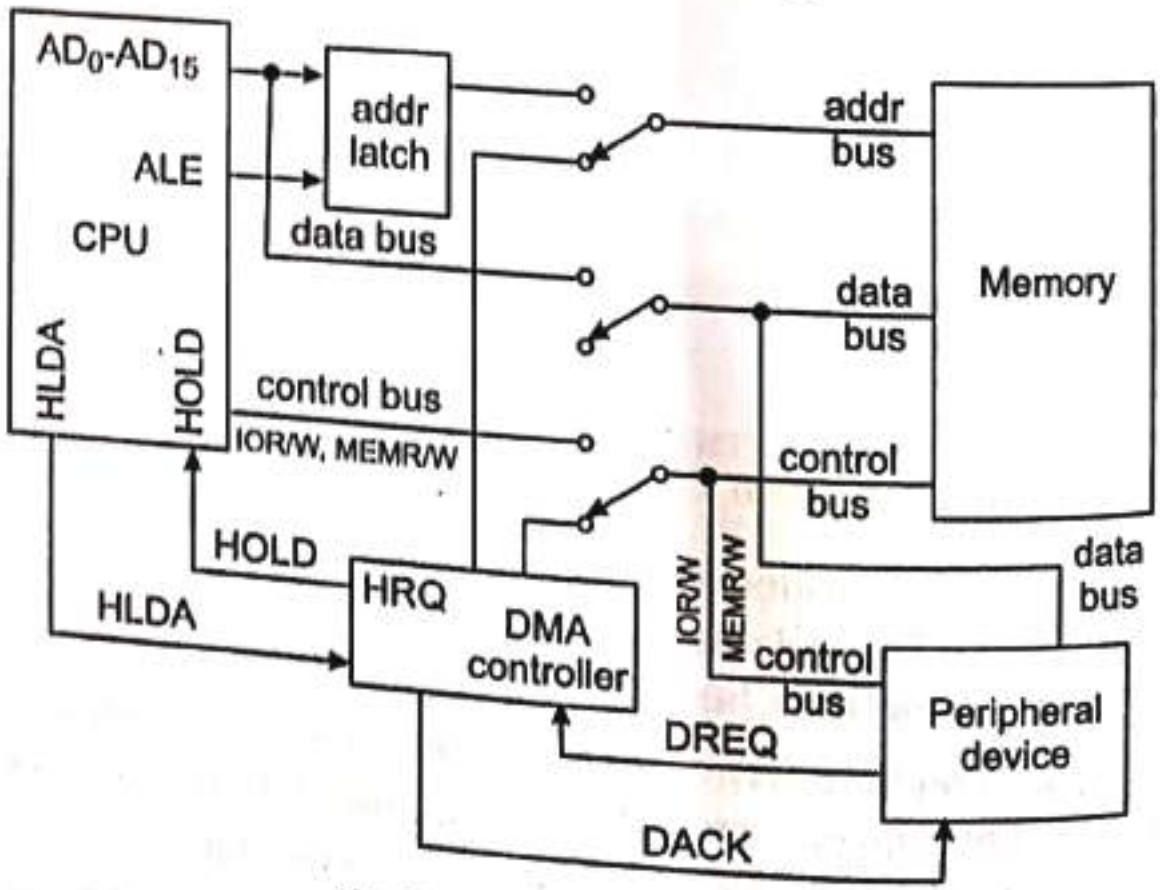
DMA request signal DRQ to the DMA controller. Then DMA controller sends a hold request HRQ signal to the MP HOLD input. The MP responds this HOLD S/g by floating its buses and sending out a hold acknowledge S/g HLDA, to the DMAC. When the DMAC (Direct memory access controller) receives the HLDA S/g, it sends a control S/g to change switch position from A to B. This disconnects the MP from the buses and connects DMAC controller to the buses.

② DMA Active Cycle :-

When DMAC gets control of the buses, it sends the memory address where the first byte of data from the disk is to be written. It also sends a DMA acknowledge IDACK S/g to the peripheral device, telling it to get ready for data transfer.



(*a*) When DMA does not operate



(*b*) When DMA operate

Fig. 5.44