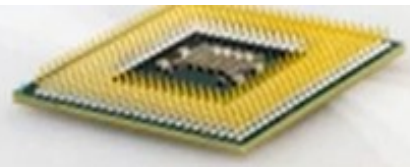
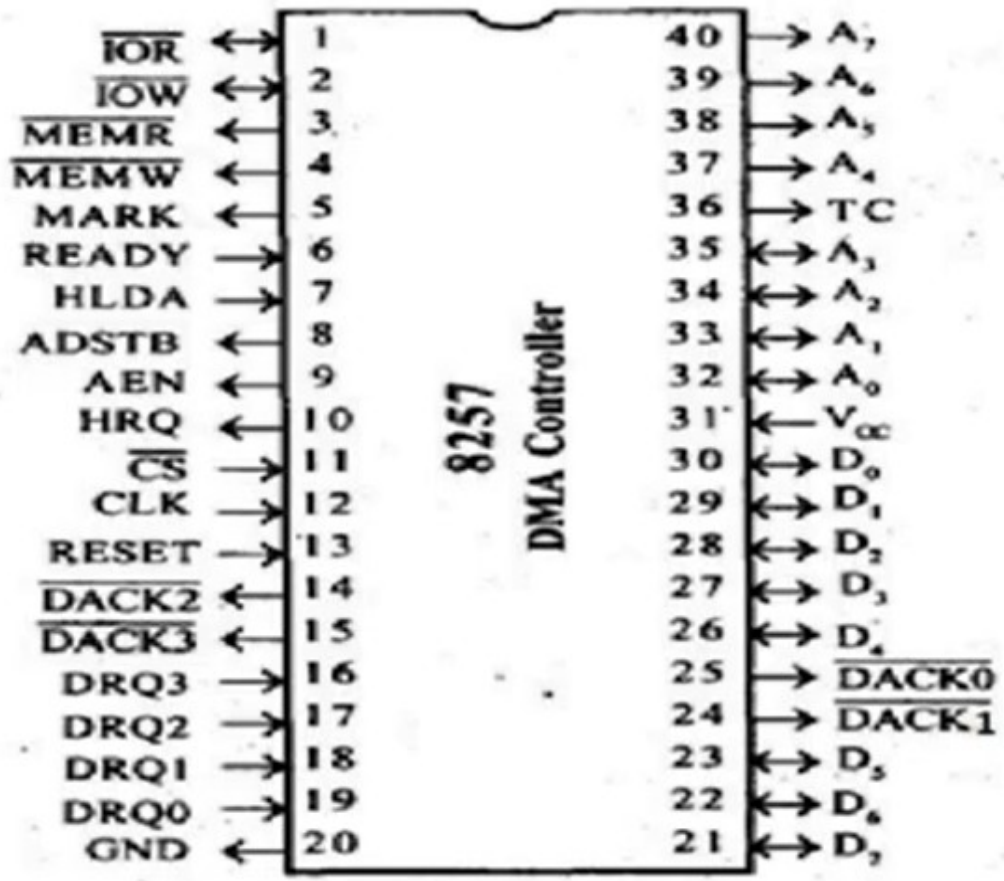
8257 DMA Controller

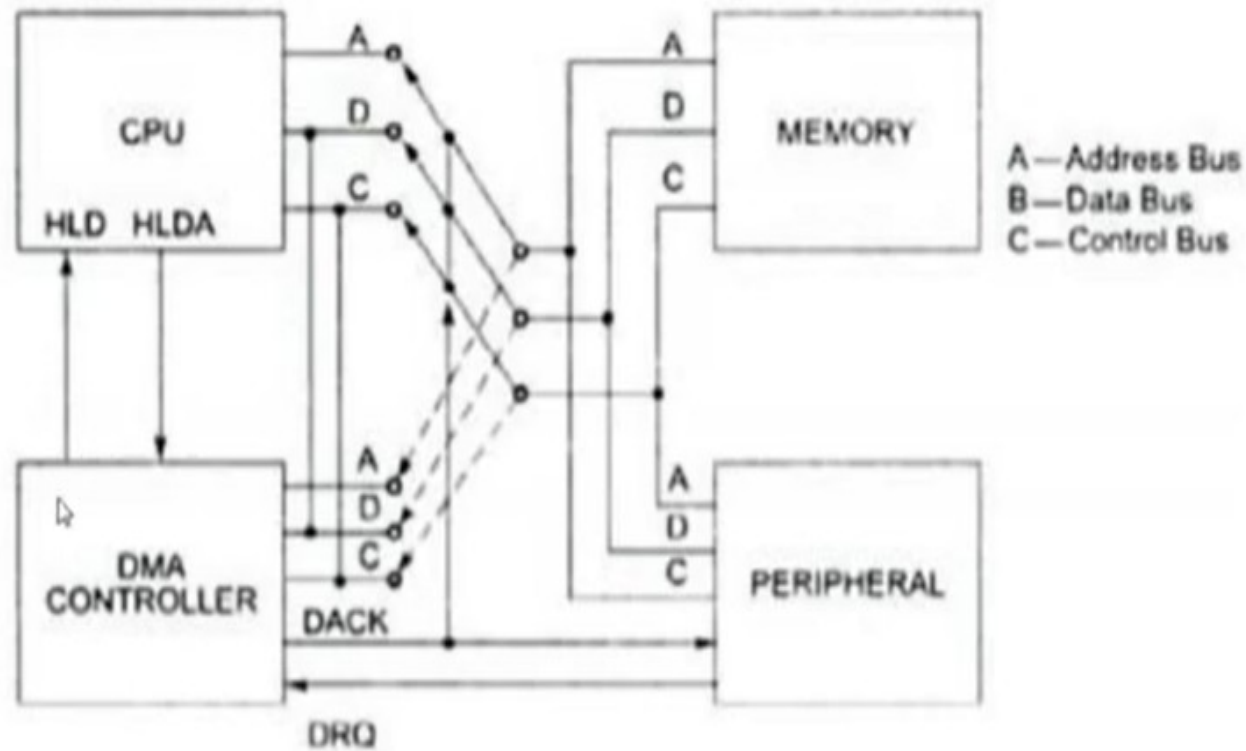


- 8257 DMA support 4-channel Direct Memory Access.
- It is specially designed by Intel for data transfer at the highest speed.
- Its initial function is to generate a peripheral request which allows the device to transfer the data directly to/from memory without any interference of the CPU.
- With the use of a DMA controller, the device sends requests to the CPU to hold its data ,memory and control bus which helps the device to transfer data directly to/fro m the memory.
- The DMA data transfer is initiated only after receiving HLDA signal from the CPU .

- Whenever a peripheral device want to start DMA mode of operation.it can generate a DMA access Request (DR)
- whenever such a request received in DMA controller it generates a request to CPU(HOLD).
- When the processor is ready to transfer data the control of internal bus structure it send acknowledgement back to DMA(HLDA).
- While receiving the DMA controller inform about this to peripheral by sending DACK(DMA access).
- After getting this acknowledgement peripheral can transfer data with the memory without any further intervention from CPU.

8257 PIN DESCRIPTION





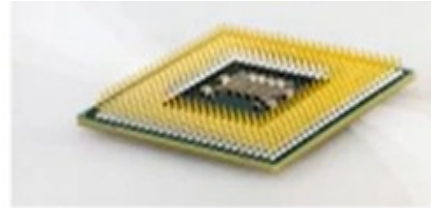


Fig 3.35 Interfacing 8257 with CPU

Features of 8257

- It has four channels which can be used over four I/O devices.
- Each channel has 16-bit DMA address and 16-bit terminal count register .
- Each channel can transfer data up to 64kb.
- Each channel can be programmed independently.
- Each channel can perform read transfer, write transfer and verify transfer operations.

It generates MARK signal to the peripheral device that 128 bytes have been transferred.

- It operates in 2 modes, i.e., **Master mode** and **Slave mode**.
- In **master mode,** the DMA controller is either serving a DMA slave's request for DMA cycles, generating refresh cycles, etc.
- In **slave mode**, DMA monitors the bus and decodes and responds to I/O read and write commands that addresses its registers.

Master – Slave mode of DMA controller

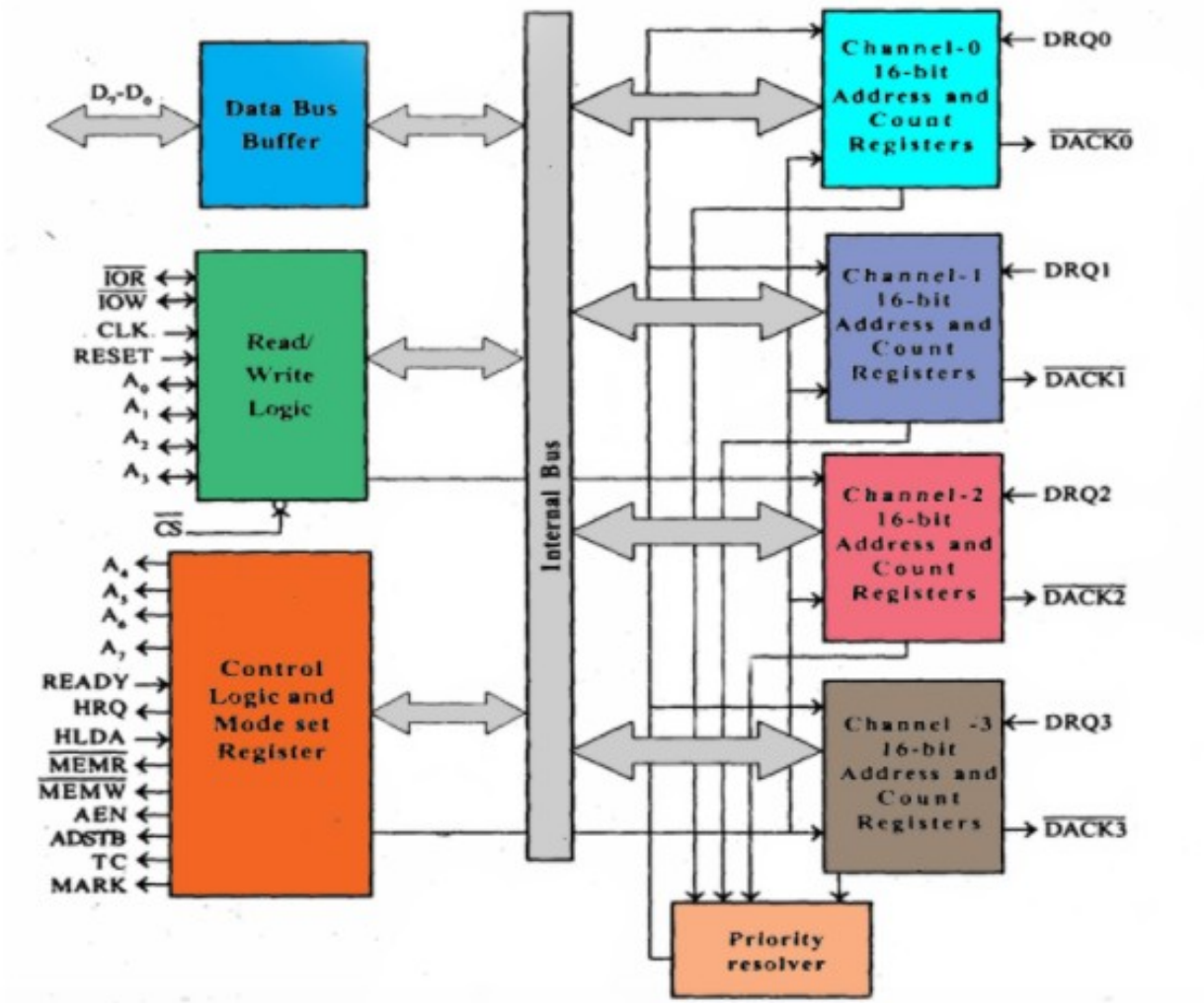
Bus master mode

- To perform data transfers, the DMA controller *acquires* the system bus from CPU.
- The CPU remains *idle* while DMA transfer takes place, the process is known as *Cycle Stealing*.

Bus slave mode

- The CPU accesses the DMA controller
- CPU programs the internal registers of DMA controller to conduct data transfer
- *DMA controller internal registers consist of*
  - source & destination address registers
  - transfer count registers for each DMA channel
  - control and status registers for initiating & monitoring the operation of the DMA controller

BLOCKDIAGRAM OF 8257



- 8257 having 4 channels each have 16 bit Address reg and Terminal Count regs.
- 4x16 bit DMA add reg
- 4x16 bit Terminal Count reg
- Priority resolver: extract the highest priority request coming through this different channels and is loaded into the control logic of 8257 controller..
- Read and Write logic support various input or output read/write operation.
- A0-A3 lines in slave mode hold the address of various registers of 8257 in order to perform various read/write operation over its internal register .In master mode A0-A3 together with the A4-A7 total 8 bit of address information act as output lines and generate address information or hold the add information generated by 8257 which corresponds to certain memory location.
- Ready: While dealing slower peripheral device. We want to extend DMA cycles while putting more no.of wait cycles. It require extra time for data transfer.
- HRQ,HLDA:generated by the 8257 to processor while getting this request send HLDA to DMA controller.

MEMR,MEMW : Indicate whether to perform memory read or memory write operation.

Add enabler:AEN:This signal is used to disable the add/data bus

ADSTB: Add strobe : output line strobe the higher byte of memory at rest generated by the DMA controller into the latches.

- There are two common registers for all the channels **namely mode set register** and status regis ter
- **DMA address register** -store the starting address of the memory location which will be acces sed by the DMA channel .
- **Terminal count register** – used for ascertaining that the data transfer through a DMA channel stops after the required number of DMA cycles .
- Thus this register should be appropriately written before the actual DMA operation starts .
- Lower order 14 bits of the terminal count register are initialized with the binary equivalent of the number of required DMA cycles minus one .
- The terminal count register content will be decremented by one and finally it becomes zero after the required number of DMA cycles are over .

Bit 15	Bit 14	Type of operation
0	0	Verify DMA cycle
0	1	Write DMA cycle
1	0	Read DMA cycle
1	1	Illegal

Mode Set Register –used for programming the 8257 as per the requirement of the system .

The function of the mode set register is to enable the DMA channels individually and also to set the various modes of operation.

Mode set register should be programmed by the CPU for enabling the DMA channels only after initializing the DMA address register and terminal count register appropriately

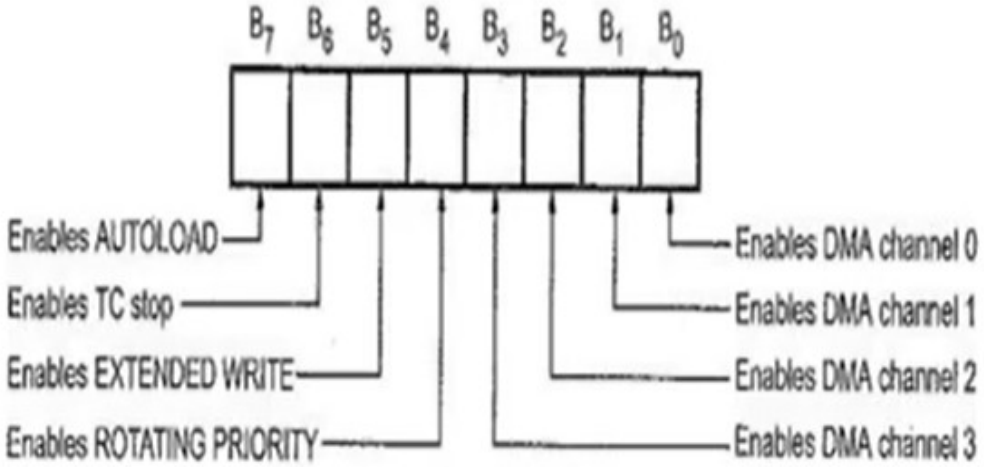


Fig. 14.65 Mode set register

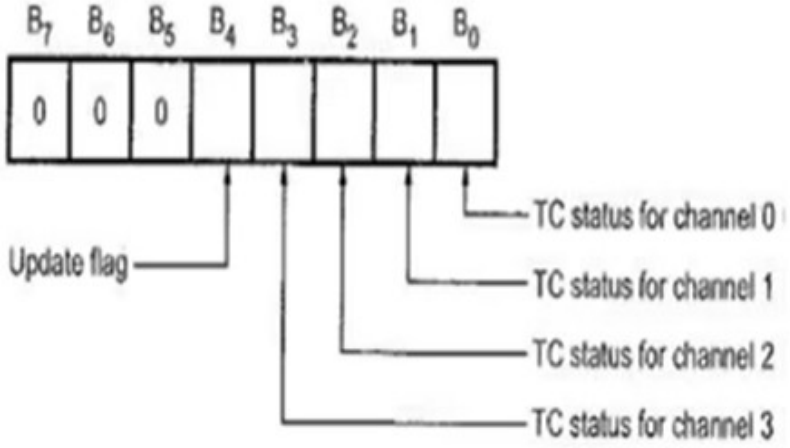


Fig. 14.66 Status register