

DIRECT MEMORY ACCESS AND THE 8237A DMA CONTROLLER

DIRECT MEMORY ACCESS

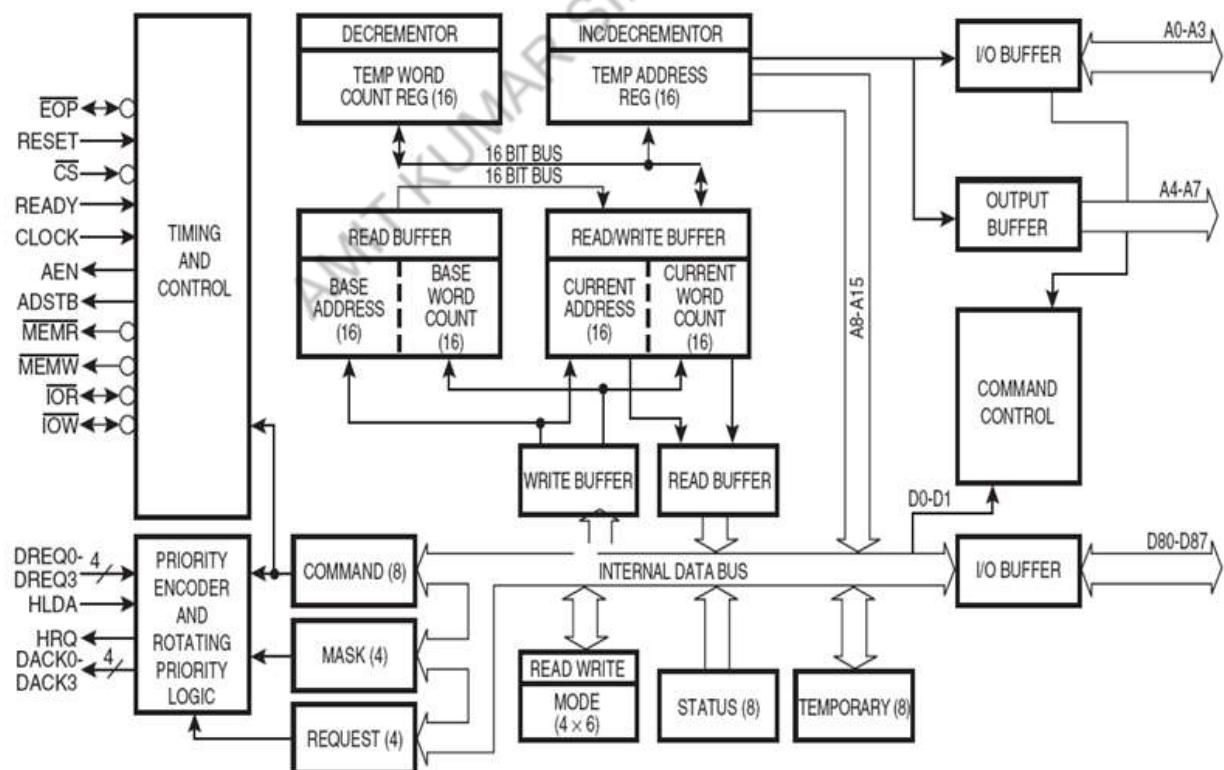
Direct Memory Access is an I/O technique commonly used for high-speed data transfer. In DMA, the MPU releases the control of the buses to a device called DMA controller. The controller manages data transfer between memory and a peripheral under its control bypassing the MPU. DMA uses two signals (HOLD, HLDA) of 8085.

A DMA uses these signals as if it were a peripheral requesting the MPU for the control of the buses. The MPU communicates with the controller by using the Chip Select line, buses and control signals. Though, once the controller has gained control it plays the role of a processor for data transfer. To perform his function the DMA controller should have data bus, address bus, Read/ Write control signals and control signal to disable its role as a peripheral and to enable its role as a processor.

THE 8237A DMA CONTROLLER

- The 8237A is actually a special-purpose microprocessor whose job is high-speed data transfer between memory and the I/O.
- The 8237A programmable Direct Memory Access controller is a 40 pin IC package.
- It has four channels with each channel capable of transferring 64 K bytes. The 8237A is capable of DMA transfers at rates of up to 1.6M bytes per second.
- It must interface with two types of devices: the MPU and peripherals.
- It plays two roles in a given system: it is an I/O to microprocessor (slave mode) and it is a data transfer processor to peripherals (master mode).

Block Diagram of 8237A DMA Controller



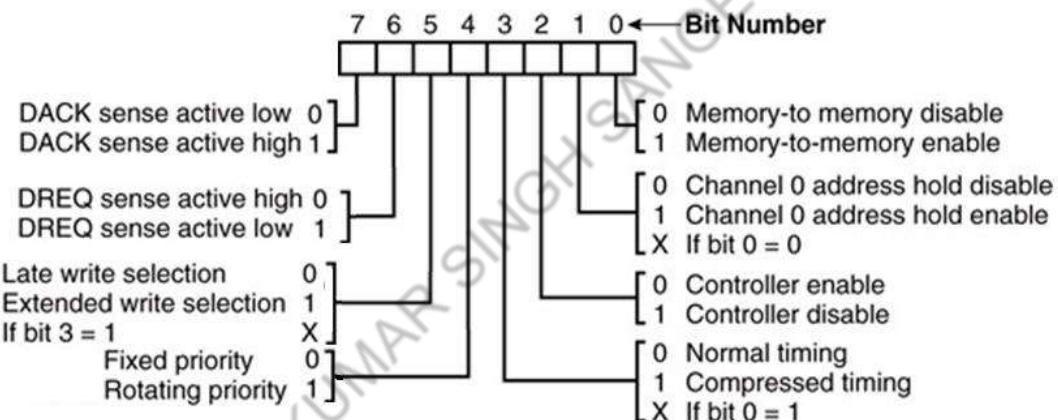
The 8237A DMA Controller Block Diagram: Internal Registers and Pin Outs

DMA Signals

CLK	The clock input is connected to the system clock signal as long as that signal is 5 MHz or less.
\overline{CS}	Chip select enables the 8237A for programming. The pin is normally connected to the output of a decoder.
RESET	The reset pin clears the command, status, request, and temporary registers. It also clears the first/last flip-flop and sets the mask register. This input primes the 8237A so it is disabled until programmed otherwise.
READY	A logic 0 on the ready input causes the 8237A to enter wait states for slower memory components.
HLDA	A hold acknowledges signals the 8237A that the microprocessor has relinquished control of the address, data, and control buses.
DREQ₀–DREQ₃	The DMA request inputs are used to request a DMA transfer for each of the four DMA channels. Because the polarity of these inputs is programmable, they are either active-high or active-low inputs.
DB0–DB7	The data bus pins are connected to the microprocessor data bus connections and are used during the programming of the DMA controller.
\overline{IOR}	I/O read is a bidirectional pin used during programming and during a DMA write cycle.
\overline{IOW}	I/O write is a bidirectional pin used during programming and during a DMA read cycle.
EOP	End-of-process is a bidirectional signal that is used as an input to terminate a DMA process or as an output to signal the end of the DMA transfer. This input is often used to interrupt a DMA transfer at the end of a DMA cycle.
A₀–A₃	These address pins select an internal register during programming and also provide part of the DMA transfer address during a DMA action. The address pins are outputs that provide part of the DMA transfer address during a DMA action.
HRQ	Hold request is an output that connects to the HOLD input of the microprocessor in order to request a DMA transfer.
DACK₀ – DACK₃	DMA channel acknowledge outputs acknowledge a channel DMA request. These outputs are programmable as either active-high or active low signals. The DACK outputs are often used to select the DMA controlled I/O device during the DMA transfer.
AEN	The address enable signal enables the DMA address latch connected to the DB7–DB0 pins on the 8237A. It is also used to disable any buffers in the system connected to the microprocessor.
ADSTB	Address strobe functions as ALE, except that it is used by the DMA controller to latch address bits A15–A8 during the DMA transfer.
\overline{MEMR}	Memory read is an output that causes memory to read data during a DMA read cycle.
\overline{MEMW}	Memory write is an output that causes memory to write data during a DMA write cycle.

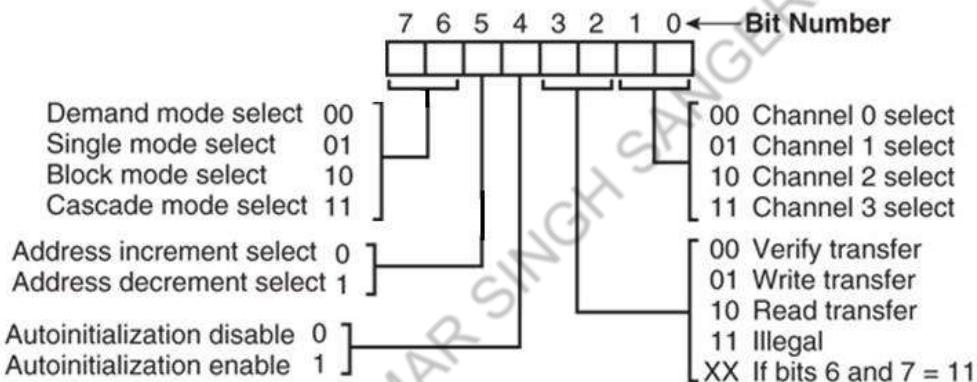
Internal Registers

- **Current Address Register:** The current address register is used to hold the 16-bit memory address used for the DMA transfer. Each channel has its own current address register for this purpose. When a byte of data is transferred during a DMA operation, the CAR is either incremented or decremented, depending on how it is programmed.
- **Current Word Count Register:** The current word count register programs a channel for the number of bytes (up to 64K) transferred during a DMA action. The number loaded into this register is one less than the number of bytes transferred. For example, if a 10 is loaded into the CWCR, then 11 bytes are transferred during the DMA action.
- **Base Address and Base Word Count Registers:** The base address (BA) and base word count (BWC) registers are used when auto-initialization is selected for a channel. In the auto-initialization mode, these registers are used to reload both the CAR and CWCR after the DMA action is completed. This allows the same count and address to be used to transfer data from the same memory area.
- **Command Register:**
 - The command register programs the operation of the 8237A DMA controller.



- Function of the command register are:
 - *To select the memory-to-memory DMA transfer mode.*
Memory-to-memory DMA transfers use DMA channel 0 to hold the source address and DMA channel 1 to hold the destination address. A byte is read from the address accessed by channel 0 and saved within the 8237A in a temporary holding register. Next, the 8237A initiates a memory write cycle in which the contents of the temporary holding register are written into the address selected by DMA channel 1. The number of bytes transferred is determined by the channel 1 count register.
The channel 0 address hold enable bit programs channel 0 for memory-to-memory transfers.
 - *To enable/disable the controller.* The controller enable/disable bit turns the entire controller on and off.
 - *To Select Normal/Compressed Timing:* The normal and compressed determines whether a DMA cycle contains two (compressed) or four (normal) clocking periods.

- *To Select Write pulse:* Bit position 5 is used in normal timing to extend the write pulse so it appears one clock earlier in the timing for I/O devices that require a wider write pulse.
- *To select the Channel priority:* Priority for the four DMA channel DREQ inputs is selected with the help of command register. In the fixed priority scheme, channel 0 has the highest priority and channel 3 has the lowest. In the rotating priority scheme, the most recently serviced channel assumes the lowest priority.
- *To program the polarities of DREQ and DACK:* Whether the polarities of the DREQ inputs and the DACK outputs are active high or active low.
- **Mode Register:**
 - The mode register programs the mode of operation for a channel. Note that each channel has its own mode register, as selected by bit positions 1 and 0. The remaining bits of the mode register select the operation, auto-initialization, increment/decrement, and mode for the channel. Verification operations generate the DMA addresses without generating the DMA memory and I/O control signals.



- The modes of operation include
 - Demand mode,
 - Single mode,
 - Block mode, and
 - Cascade mode.

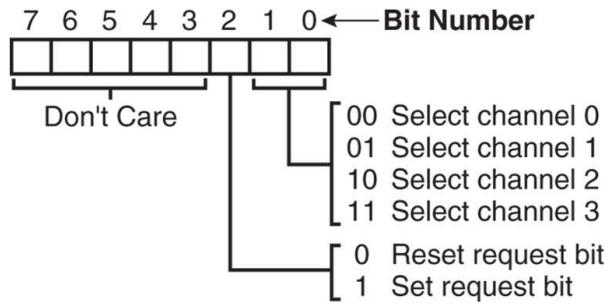
Demand mode transfers data until an external EOP is input or until the DREQ input becomes inactive.

Single mode releases the HOLD after each byte of data is transferred. If the DREQ pin is held active, the 8237A again requests a DMA transfer through the DRQ line to the microprocessor's HOLD input.

Block mode automatically transfers the number of bytes indicated by the count register for the channel. DREQ need not be held active through the block mode transfer.

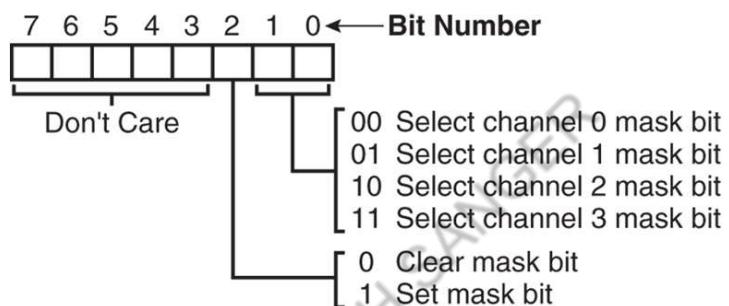
Cascade mode is used when more than one 8237A is present in a system.

- **Bus Request Register:**
 - The bus request register is used to request a DMA transfer via software. This is very useful in memory-to-memory transfers, where an external signal is not available to begin the DMA transfer.

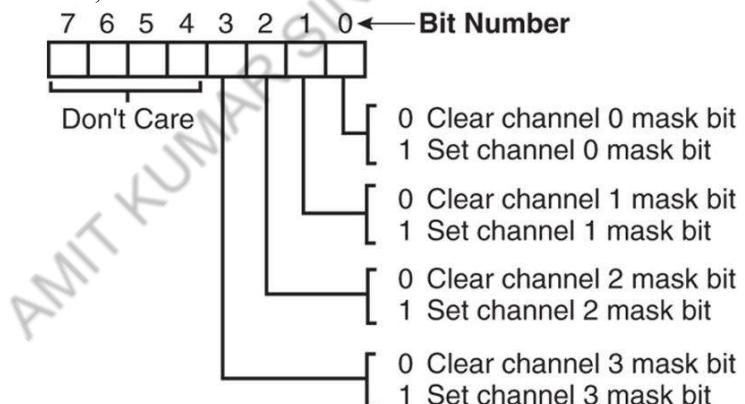


- **Mask Register Set/Reset:**

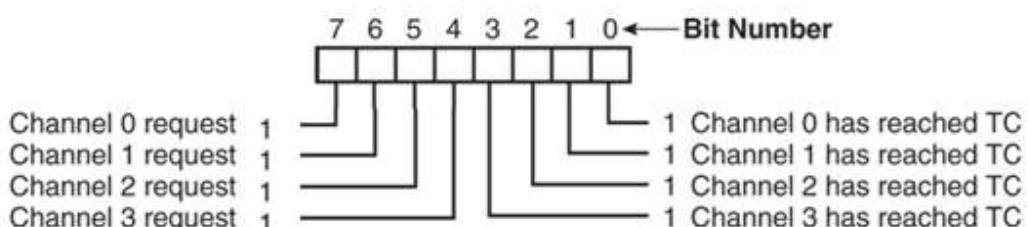
- The mask register set/reset sets or clears the channel mask. If the mask is set, the channel is disabled. Recall that the RESET signal sets all channel masks to disable them.



- The mask register clears or sets all of the masks with one command instead of individual channels, as with the MCSR.



- **Status Register:** The status register shows the status of each DMA channel. The TC bits indicate whether the channel has reached its terminal count (transferred all its bytes). Whenever the terminal count is reached, the DMA transfer is terminated for most modes of operation. The request bits indicate whether the DREQ input for a given channel is active.



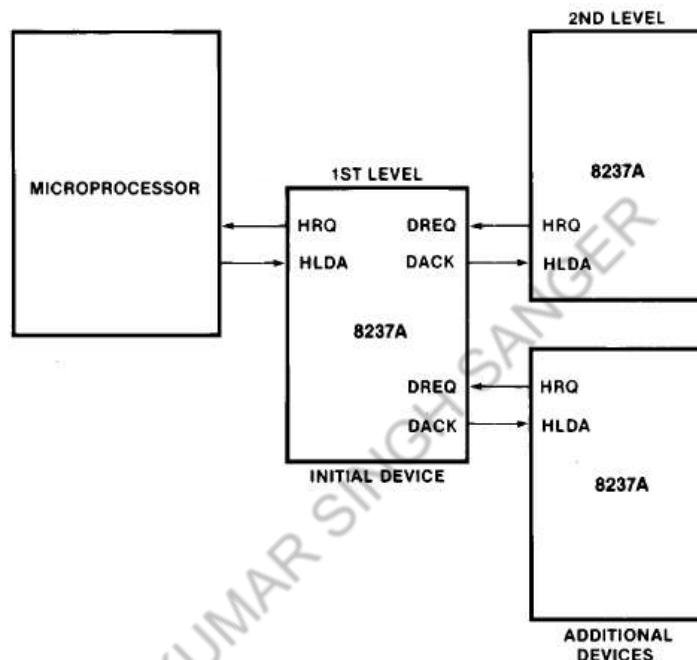
DMA Operation

- The 8237A is designed to operate in two major cycles. These are called Idle and Active cycles. Each device cycle is made up of a number of states. The 8237A can assume seven separate states, each composed of one full clock period.
 - State I (SI) is the inactive state. It is entered when the 8237A has no valid DMA requests pending. While in SI, the DMA controller is inactive
 - State S0 (S0) is the first state of a DMA service. The 8237A has requested a hold but the processor has not yet returned an acknowledge. The 8237A may still be programmed until it receives HLDA from the CPU. An acknowledge from the CPU will signal that DMA transfers may begin. S1, S2, S3 and S4 are the working states of the DMA service.
 - If more time is needed to complete a transfer than is available with normal timing, wait states (SW) can be inserted between S2 or S3 and S4 by the use of the Ready line on the 8237A.
- Memory-to-memory transfers require a read-from and a write-to-memory to complete each transfer. The states, which resemble the normal working states, use two digit numbers for identification. Eight states are required for a single transfer. The first four states (S11, S12, S13, S14) are used for the read from-memory half and the last four states (S21, S22, S23, S24) for the write-to-memory half of the transfer.

Transfer Modes of 8237A

1. **Single Transfer Mode--** In Single Transfer mode the device is programmed to make one transfer only. The word count will be decremented and the address decremented or incremented following each transfer. When the word count ``rolls over'' from zero to FFFFH, a Terminal Count (TC) will cause an Autoinitialize if the channel has been programmed to do so.
2. **Block Transfer Mode--** In Block Transfer mode the device is activated by DREQ to continue making transfers during the service until a TC, caused by word count going to FFFFH, or an external End of Process (EOP) is encountered. DREQ need only be held active until DACK becomes active. Again, an Autoinitialization will occur at the end of the service if the channel has been programmed for it.
3. **Demand Transfer Mode--** In Demand Transfer mode the device is programmed to continue making transfers until a TC or external EOP is encountered or until DREQ goes inactive. Thus transfers may continue until the I/O device has exhausted its data capacity. After the I/O device has had a chance to catch up, the DMA service is re-established by means of a DREQ. During the time between services when the microprocessor is allowed to operate the intermediate values of address and word count are stored in the 8237A Current Address and Current Word Count registers. Only an EOP can cause an Autoinitialize at the end of the service. EOP is generated either by TC or by an external signal. DREQ has to be low before S4 to prevent another Transfer.
4. **Cascade Mode--** This mode is used to cascade more than one 8237A together for simple system expansion. The HRQ and HLDA signals from the additional 8237A are connected to the DREQ and DACK signals of a channel of the initial 8237A. This allows the DMA requests of the additional device to propagate through the priority network circuitry of the preceding device. The priority chain is preserved and the new device must wait for its

turn to acknowledge requests. Since the cascade channel of the initial 8237A is used only for prioritizing the additional device, it does not output any address or control signals of its own. These could conflict with the outputs of the active channel in the added device. The 8237A will respond to DREQ and DACK but all other outputs except HRQ will be disabled. The ready input is ignored. Figure shows two additional devices cascaded into an initial device using two of the previous channels. This forms a two level DMA system. More 8237As could be added at the second level by using the remaining channels of the first level. Additional devices can also be added by cascading into the channels of the second level device, forming a third level.

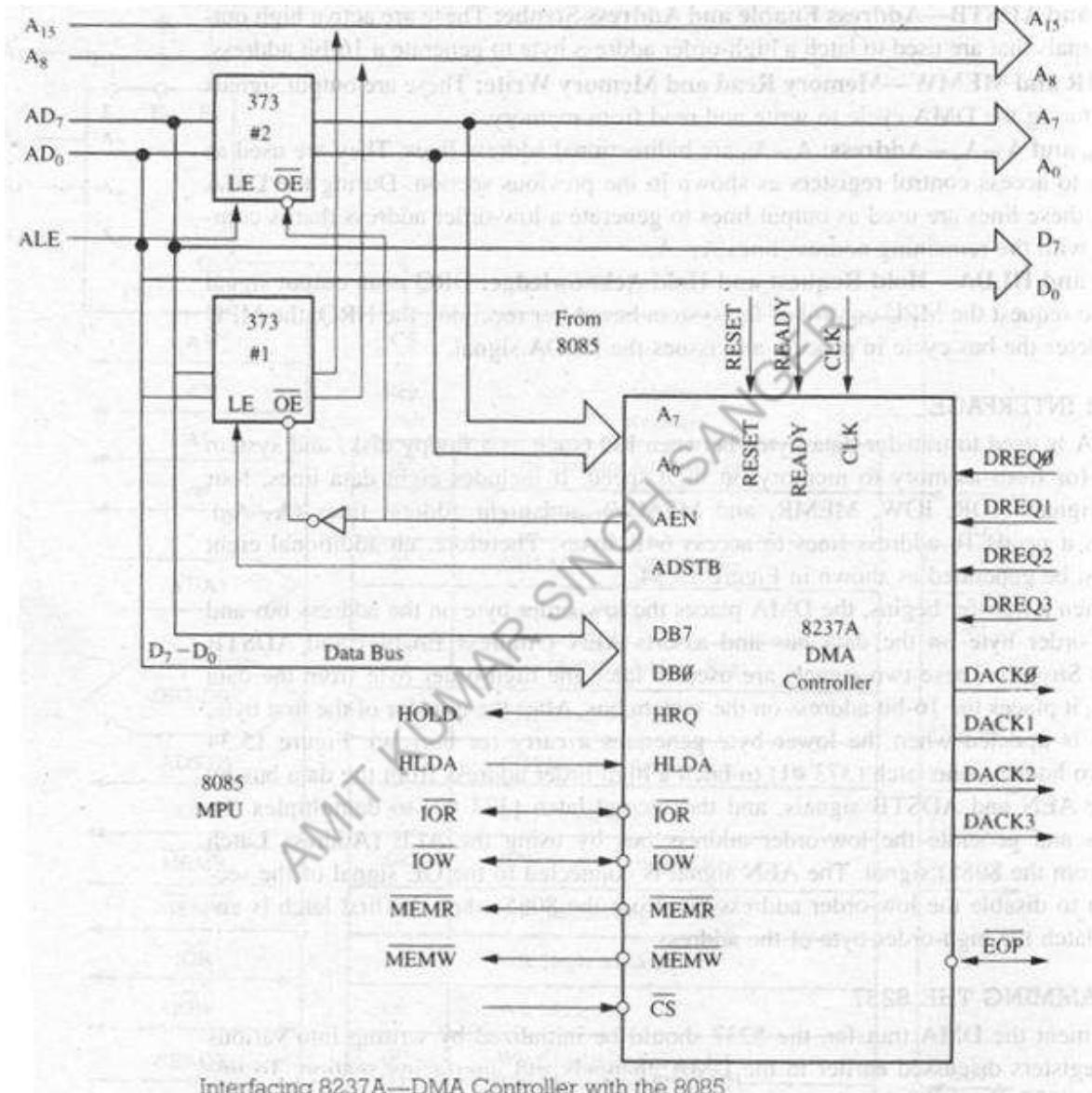


- 5. Memory-to-Memory--** To perform block moves of data from one memory address space to another with a minimum of program effort and time, the 8237A includes a memory-to-memory transfer feature.

- Programming a bit in the Command register selects channels 0 and 1 to operate as memory-to-memory transfer channels. The transfer is initiated by setting the software DREQ for channel 0. The 8237A requests a DMA service in the normal manner. After HLDA is true, the device, using four state transfers in Block Transfer mode, reads data from the memory.
- The channel 0 Current Address register is the source for the address used and is decremented or incremented in the normal manner.
- The data byte read from the memory is stored in the 8237A internal Temporary register. Channel 1 then performs a four-state transfer of the data from the Temporary register to memory using the address in its Current Address register and incrementing or decrementing it in the normal manner.
- The channel 1 current Word Count is decremented. When the word count of channel 1 goes to FFFFH, a TC is generated causing an EOP output terminating the service.
- Channel 0 may be programmed to retain the same address for all transfers. This allows a single word to be written to a block of memory.

- The 8237A will respond to external EOP signals during memory-to-memory transfers. Data comparators in block search schemes may use this input to terminate the service when a match is found. Memory-to-memory operations can be detected as an active AEN with no DACK outputs.

Interfacing 8237A- DMA controller with 8085



The DMA is used to transfer data bytes between I/O and system memory at high speed. It includes eight data lines, four control signals and eight address line (A₇-A₀). Although it needs 16 address lines to access 64 K bytes.

Address Generation

- When transfer begins, the DMA places the low order address byte on the address bus and the high order address byte on the data bus and asserts the AEN (Address Enable)

- and ADSTB (Address Strobe). These two singles are used to latch the high order byte from the data bus; therefore it places the 16-bit address on the system bus.
- During Block and Demand Transfer mode services, which include multiple transfers, the addresses generated will be sequential. For many transfers the data held in the external address latch will remain the same. This data need only change when a carry or borrow from A7 to A8 takes place in the normal sequence of addresses.

Software Commands or 8237A Commands and Programming

Three software commands are used to control the operation of the 8237A. These commands do not have a binary bit pattern, as do the various control registers within the 8237A. A simple output to the correct port number enables the software command.

The functions of the software commands are explained as follows:

1. Clear the first/last flip-flop

- Clears the first/last (F/L) flip-flop within the 8237A. The F/Lflip-flop selects which byte (low or high order) is read/written in the current address and currentcount registers.
- If F/L = 0, the low-order byte is selected; if F/L = 1, the high-order byteis selected. Any read or write to the address or count register automatically toggles the F/Lflip-flop.

2. Master clear

- Acts exactly the same as the RESET signal to the 8237A. As with the RESETsignal, this command disables all channels.
- Using this command, all the internal registers of8237A are cleared, while allthebits of the mask register are set. This means after executing this command, the DMA controller disables Allthe DMA channels and enters an idle cycle.

3. Clear mask register

- As a mask set register when set, may disable the DMA channels, so that the DMA requests are not entertained. A clear mask register command will clearthe bits of the mask register individually or collectively, so that the DMA channels are enabled for accepting DMA requests.

Along with these commands, a few others also used for manipulating the current address registers and the current word count registers. These commands may be executed while the 8237A is under the control of the CPU.

PROGRAMMABLE INTERVAL TIMER 8254

- The 8254 is used to generate the accurate time delays.
- Instead of setting up timing loops in system software, the programmer configures the 8254 to match his requirements, initializes one of the counters of the 8254 with the desired quantity, then upon command the 8254 will count out the delay and interrupt the CPU when it has completed its tasks.
- The 8253/54 includes three identical 16 bit counters that can operate independently.
- To operate a counter,
 - a 16-bit count is loaded in its register and,
 - on command, it begins to decrement the count until it reaches 0.
 - At the end of the count, it generates a pulse that can be used to interrupt the CPU.
- The counter can count either in binary or BCD.
- In addition, a count can be read by the CPU while the counter is decrementing.

Difference between 88053 and 8254

8253	8254
1. Operating frequency 0 - 2.6 MHz.	1. Operating frequency 0 - 10 MHz.
2. Uses N-MOS technology.	2. Uses H-MOS technology.
3. Read-Back command not available.	3. Read-Back command available
4. Reads and writes of the same counter can not be interleaved.	4. Reads and writes of the same counter can be interleaved.

Features of 8254

1. Three independent 16-bit down counters.
2. 8254 can handle inputs from DC to 10 MHz (5MHz 8254-5 8MHz 8254 10MHz 8254-2) whereas 8253 can operate up to 2.6 MHz.
3. Three counters are identical presetable, and can be programmed for either binary or BCD count.
4. Counter can be programmed in six different modes.
5. Compatible with all Intel and most other microprocessors.
6. 8254 has powerful command called READ BACK command which allows the user to check the count value, programmed mode and current mode and current status of the counter.

Block Diagram

- It includes:
 - three counters,
 - a data bus buffer,
 - Read/Write control logic, and
 - a control register.
- Each counter has two input signals CLOCK and GATE and one output signal OUT.

Data Bus Buffer : This tri-state, bi-directional, 8-bit buffer is used to interface the 8254 to the system data bus. The Data bus buffer has three basic functions.

1. Programming the modes of 8254.
2. Loading the count registers.
3. Reading the count values.

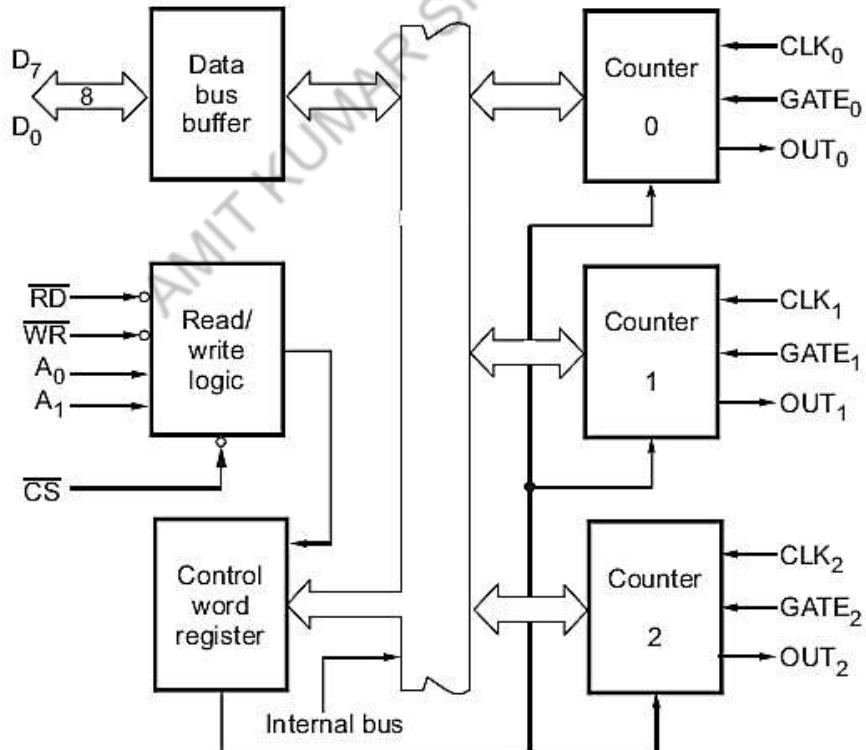
Read/Write Logic : The Read/Write logic has five signals : \overline{RD} , \overline{WR} , \overline{CS} and the address lines A0 and A1. In the peripheral I/O mode, the RD, and WR signals are connected to \overline{IOR} and \overline{IOW} , respectively. In memory-mapped I/O, these are connected to \overline{MEMR} and \overline{MEMW} .

Address lines A0 and A1 of the CPU are usually connected to lines A0 and A1 of the 8254, and \overline{CS} is tied to a decoded address. The control word register and counters are selected according to the signals on lines A0 and A1.

A0	A1	Selection
0	0	Counter 0
0	1	Counter 1
1	0	Counter 2
1	1	Control Word Register

Table shows the selected operations for various control inputs.

\overline{CS}	\overline{RD}	\overline{WR}	A1	A0	SELECTED OPERATION
0	1	0	0	0	Write Counter0
0	1	0	0	1	Write Counter1
0	1	0	1	0	Write Counter1
0	1	0	1	1	Write Control Word
0	0	1	0	0	Read Counter0
0	0	1	0	1	Read Counter1
0	0	1	1	0	Read Counter1
0	0	1	1	1	No operation (Tristated)
0	1	1	X	X	No operation (Tristated)
1	X	X	X	X	Disabled (Tristated)



Block Diagram of 8254

Control Word Register : This register is accessed when lines A0 and A1 are at logic 1. It is used to Read a command word which specifies the counter to be used (binary or BCD), its mode, and either a read or Read operation.

Counters : These three functional blocks are identical in operation. Each counter consists of a single, 16 bit, pre-settable, down counter. The counter can operate in either binary or BCD and its input, gate and output are configured by the selection of modes stored in the control word register. The counters are fully independent. The programmer can read the contents of any of the three counters without disturbing the actual count in process.

Operational Description

- **Programming the 8253/54 (Control Word of 8253/54)** : Each counter of the 8253/54 is individually programmed by writing a control word into the control word register (A0 - A1 = 11). The figure shows the control word format. Bits SC1 and SC0 select the counter, bits RW1 and RW0 select the read, Read or latch command, bits M2, M1 and M0 select the mode of operation and bit BCD decides whether it is a BCD counter or binary counter.

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
SC ₁	SC ₀	RW ₁	RW ₀	M ₂	M ₁	M ₀	BCD

SC - Select counter

SC₁ SC₀

0	0	Select counter 0
0	1	Select counter 1
1	0	Select counter 2
1	1	Illegal for 8253 Read -Back command for 8254 (See Read operations)

M - Mode

M₂ M₁ M₀

0	0	0	Mode 0
0	0	1	Mode 1
x	1	0	Mode 2
x	1	1	Mode 3
1	0	0	Mode 4
1	0	1	Mode 5

RW - Read /Write

RW₁ RW₀

0	0	Counter latch command (See Read operations)
0	1	Read / Write least significant byte only
1	0	Read / Write most significant byte only
1	1	Read / write least significant byte first, then most significant byte

BCD :

0	Binary counter 16 - bits
1	Binary coded decimal (BCD) Counter (4 Decades)

Note : Don't care bits (x) should be 0 to ensure compatibility with future Intel products

- **READ Operation :**

1. Read a control word into control register.
2. Load the low-order byte of a count in the counter register.
3. Load the high-order byte of count in the counter register.

- **READ Operation :** In some applications, especially in event counters, it is necessary to read the value of the count in process. This can be done by three possible methods:

1. Simple Read:

- It involves reading a count after inhibiting the counter by controlling the gate input or the clock input of the selected counter, and two I/O read operations are performed by the CPU.
- The first I/O operation reads the low-order byte, and the second I/O operation reads the high order byte.

2. Counter Latch Command:

- In the second method, an appropriate control word is written into the control register to latch a count in the output latch, and two I/O read operations are performed by the CPU.
- The first I/O operation reads the low-order byte, and the second I/O operation reads the high order byte.

3. Read-Back Command (Available only for 8254) :

- The third method uses the Read-Back command.
- This command allows the user to check the count value, programmed Mode, and current status of the OUT pin and Null count flag of the selected counter(s).
- Figure shows the format of the control word register for Read-Back command.

$$A_0, A_1 = 11 \quad \overline{CS} = 0 \quad \overline{RD} = 1 \quad \overline{WR} = 0$$

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
1	1	COUNT	STATUS	CNT ₂	CNT ₁	CNT ₀	0

D₅ : 0 = Latch count of selected counter(s)

D₄ : 0 = Latch status of selected counter(s)

D₃ : 1 = Select counter 2

D₂ : 1 = Select counter 1

D₁ : 1 = Select counter 0

D₀ : Reserved for future expansion : must be 0.

Control word register for read-back command

- The Read-Back command may be used to latch multiple counter output latches by setting the COUNT bit D5 = 0 and selecting the desired counter (s).
- Each counter's latch count is held until it is read (or the counter is reprogrammed). That counter is automatically unlatched when read.
- The Read-Back command may be used to latch multiple counter output latches by setting the COUNT bit D5 = 0 and selecting the desired counter (s). Each counter's latch count is held until it is read. That counter is automatically unlatched when read.

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
OUTPUT	NUL COUNT	RW ₁	RW ₀	M ₂	M ₁	M ₀	BCD

D₇ 1 = OUT pin is 1.

0 = OUT pin is 0.

D₆ 1 = NULL count.

0 = Count available for reading.

D₅-D₀ = Counter programmed mode.

4. Interleaved Read and Read:

- Another feature of the 8254 is that reads and Reads of the same counter may be interleaved. For example, if the counter is programmed for the two byte counts, the following sequence is valid.
 - Another feature of the 8254 is that reads and Reads of the same counter may be interleaved. For example, if the counter is programmed for the two byte counts, the following sequence is valid.
- | | |
|---|---|
| 1. Read least significant byte.
3. Read most significant byte. | 2. Read new least significant byte.
4. Read new most significant byte. |
|---|---|

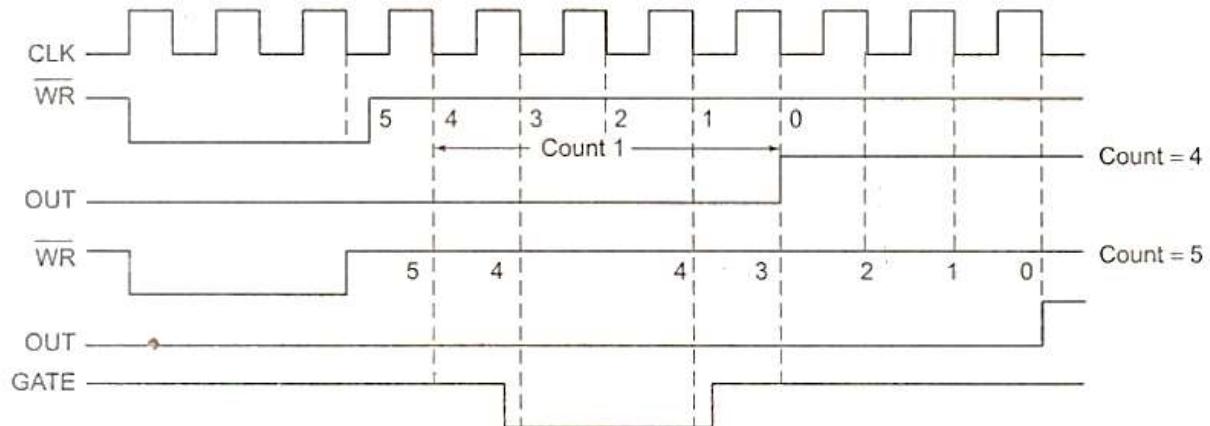
Operating Modes of 8254

Each of the three counters of 8254 can be operated in one of the following six modes of operation:

1. Mode 0 (Interrupt on terminal count)
2. Mode 1 (Programmable monoshot)
3. Mode 2 (Rate generator)
4. Mode 3 (Square wave generator)
5. Mode 4 (Software triggered strobe)
6. Mode 5 (Hardware triggered strobe)

1. MODE 0 (Interrupt on terminal count):

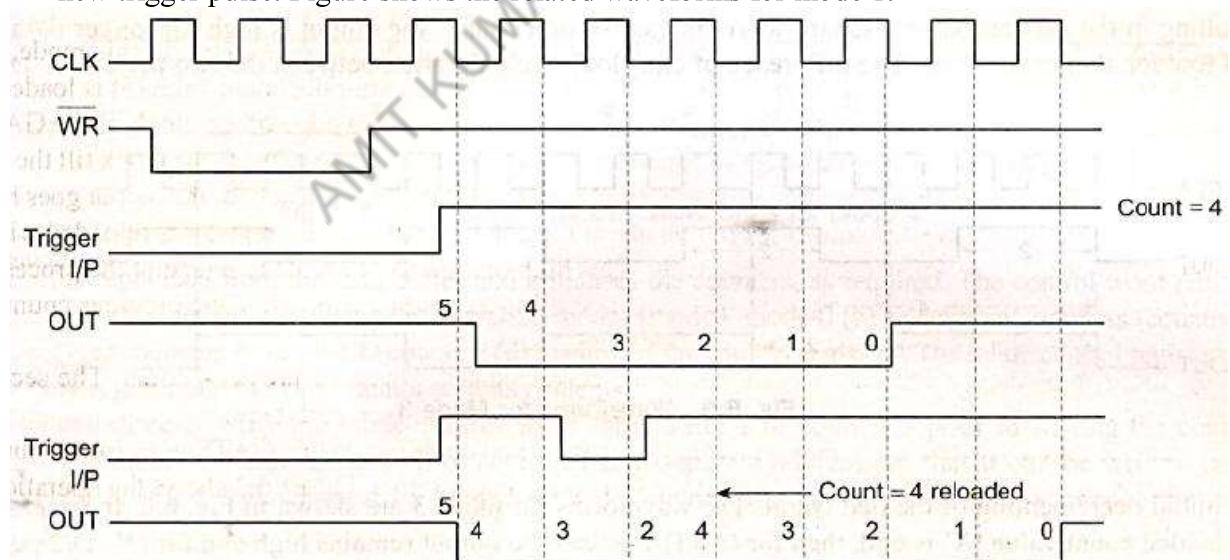
- This mode of operation is generally called as interrupt on terminal count.
- In this mode, the output is initially low after the mode is set. The output remains low even after the count value (5) is loaded in the counter.
- The counter starts decrementing the count value after the falling edge of the clock, if the GATE input is high.
- The process of decrementing the counter continues at each falling edge of the clock till the terminal count is reached, i.e. the count becomes zero.
- When the terminal count is reached, the output goes high and remains high till the selected control word register or the corresponding count register is reloaded with a new mode of operation or a new count, respectively. This high output may be used to interrupt the processor whenever required, by setting a suitable terminal count.
- Writing a count register while the previous counting is in process, generates the following sequence of response.
 - o If a new count is written to the counter, it will be loaded on the next CLK pulse and counting will continue from the new count.
 - o The first byte of the new count when loaded in the count register, stops the previous count. The second byte when written, starts the new count, terminating the previous count then and there.
- The GATE signal is active high and should be high for normal counting. When GATE goes low counting is terminated and the current count is latched till the GATE again goes high. Figure shows the operational waveforms in mode 0.



Waveforms of \overline{WR} , OUT and GATE in Mode 0

2. Mode 1 (Programmable monoshot):

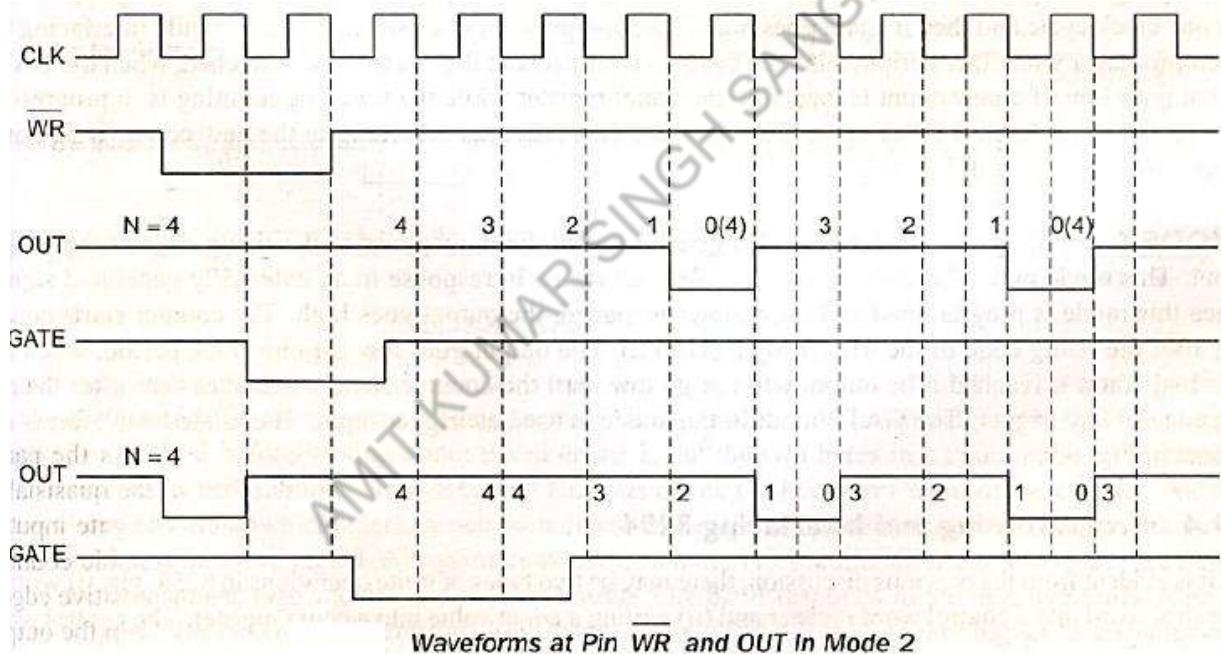
- This mode of operation of 8254 is called as programmable one-shot mode. As the name implies, in this mode, the 8254 can be used as a monostable multivibrator.
- The duration of the quasistable state of the monostable multivibrator is decided by the count loaded in the count register.
- The gate input is used as trigger input in this mode of operation.
- Normally the output remains high till the suitable count is loaded in the count register and a trigger is applied. After the application of a trigger (on the positive edge), the output goes low and remains low till the count becomes zero.
- If another count is loaded when the output is already low, it does not disturb the previous count till a new trigger pulse is applied at the GATE input. The new counting starts after the new trigger pulse. Figure shows the related waveforms for mode 1.



\overline{WR} , GATE and OUT Waveforms for Mode 1

3. Mode 2 (Rate generator):

- This mode is called either rate generator or divide by N counter.
- In this mode, if N is loaded as the count value, then, after $(N-1)$ cycles, the output becomes low only for one clock cycle.
- The count N is reloaded and again the output becomes high and remains so for $(N-1)$ clock pulses.
- The output is normally high after initialization or even a low signal on GATE input can force the output to go high.
- If GATE goes high, the counter starts counting down from the initial value.
- The counter generates an active low pulse at the output initially, after the count register is loaded with a count value. Then count down starts and whenever the count becomes zero another active low pulse is generated at the output. The duration of these active low pulses are equal to one clock cycle.
- The number of input clock pulses between the two low pulses at the output is equal to the count loaded.
- Figure shows the related wave- forms for mode 2. Interestingly, the counting is inhibited when GATE becomes low.



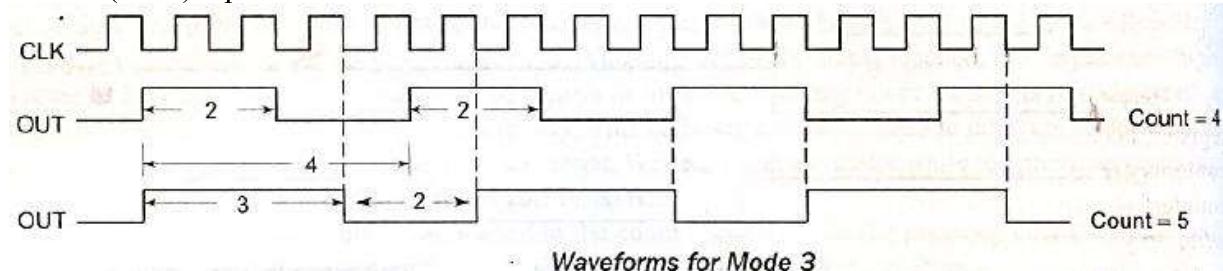
4. Mode 3 (Square wave generator):

- In this mode, the 8254 can be used as a square wave rate generator. In terms of operation this mode is somewhat similar to mode 2.
- When, the count N loaded is even, then for half of the count, the output remains high and for the remaining half it remains low.
- If the count loaded is odd, the first clock pulse decrements it by 1 resulting in an even count value (holding the output high). Then the output remains high for half of the new count and goes low for the remaining half.
- This procedure is repeated continuously resulting in the generation of a square wave. In case of odd count, the output is high for longer duration and low for shorter duration. The difference of

one clock cycle duration between the two periods is due to the initial decrementing of the odd count.

- The waveforms for mode 3 are shown in figure.

- In general, if the loaded count value N is odd, then for $(N + 1)/2$ pulses the output remains high and for $(N - 1)/2$ pulses it remains low.



Waveforms for Mode 3

5. Mode 4 (Software triggered strobe):

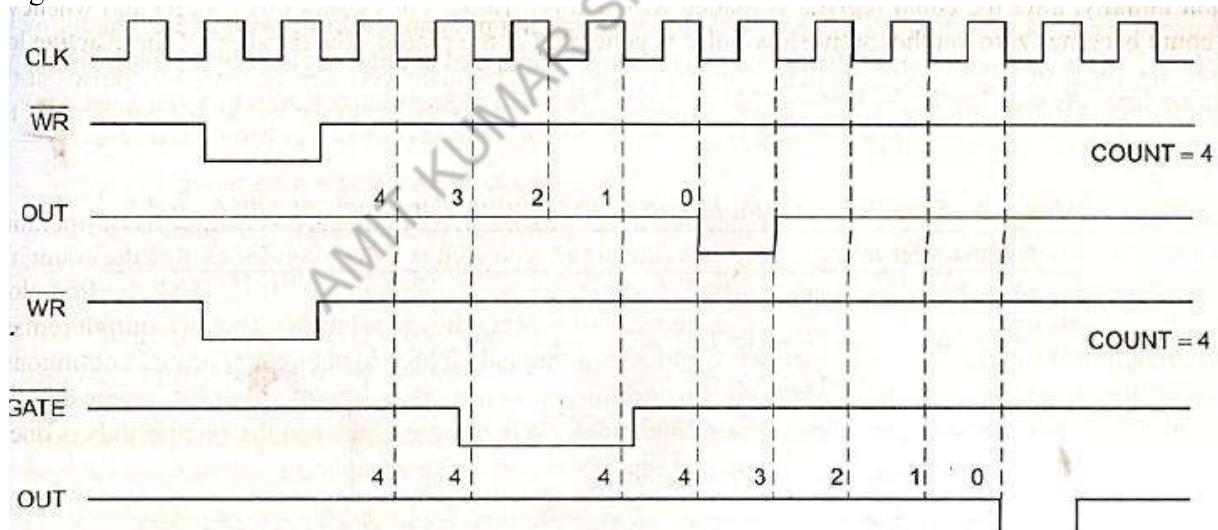
- This mode of operation of 8254 is named as software triggered strobe.

- After the mode is set, the output goes high.

- When a count is loaded, counting down starts. On terminal count the output goes low for one clock cycle, and then it again goes high. This low pulse can be used as a strobe, while interfacing the microprocessor with other peripherals.

- The count is inhibited and, the count value is latched, when the GATE signal goes low. If a new count is loaded in the count register while the previous counting is in progress, it is accepted from the next clock cycle.

- The counting then proceeds according to the new count the related waveforms are shown in figure.



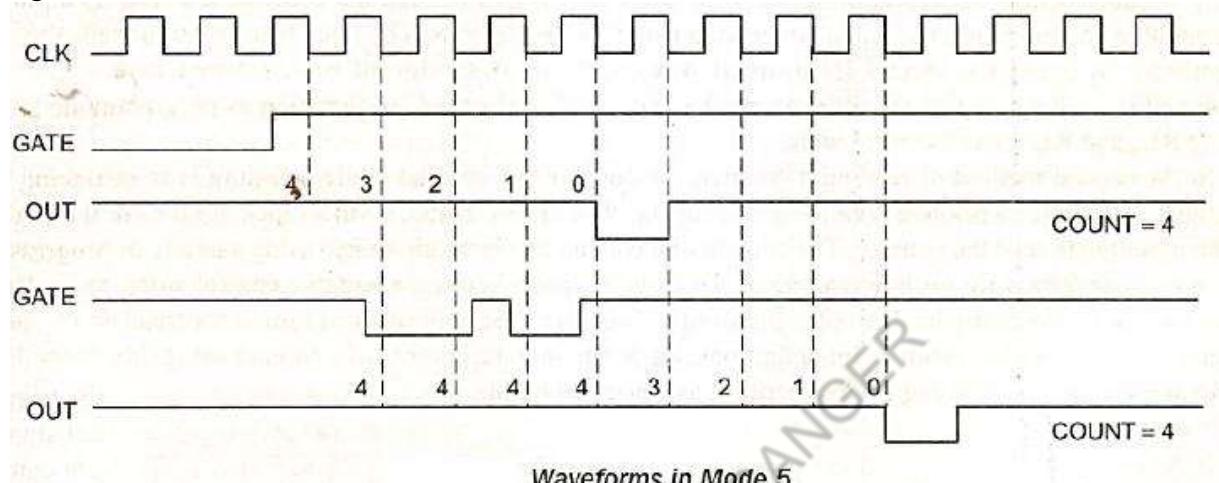
WR, GATE and OUT Waveforms for Mode 4

6. Mode 5 (Hardware triggered strobe):

- This mode of operation also generates a strobe in response to the rising edge at the trigger input.

- This mode may be used to generate a delayed strobe in response to an externally generated signal.

- Once this mode is programmed and the counter is loaded, the output goes high. The counter starts counting after the rising edge of the trigger input (GATE).
- The output goes low for one clock period, when the terminal count is reached. The output will not go low until the counter content becomes zero after the rising edge of any trigger.
- The GATE input in this mode is used as trigger input. The related waveforms are shown in figure.



Waveforms in Mode 5

THE 8259A PROGRAMMABLE INTERRUPT CONTROLLER

- The 8259A is a programmable interrupt controller designed to work with Intel microprocessors 8085, 8086, and 8088.
- The 8259A interrupt controller can
 1. manage eight interrupts according to the instructions written into its control registers.
 2. vector an interrupt request anywhere in the memory map. However, all eight interrupts are spaced at the interval of either four or eight locations. This eliminates the major drawback of the 8085 interrupts in which all interrupts are vectored to memory locations on page 00H.
 3. resolve eight levels of interrupt priorities in a variety of modes, such as fully nested mode, automatic rotation mode, and specific rotation mode (to be explained later).
 4. mask each interrupt request individually.
- S. read the status of pending interrupts, in-service interrupts, and masked interrupts.
- 6.. be set up to accept either the level-triggered or the edge-triggered interrupt request.
- 7. be expanded to 64 priority levels by cascading additional 8259As.
- 8. be set up to work with either the 8085 microprocessor mode or the 8086/8088 microprocessor mode.
- The 8259A is upward-compatible with its predecessor, the 8259. The main difference between the two is that the 8259A can be used with Intel's 8086/88 16-bit microprocessors. It also includes additional features such as the level-triggered mode, buffered mode and automatic-end-of-interrupt mode.

Block Diagram of the 8259A

It includes eight block control logic, Read/Write logic, data bus buffer, three registers (IRR, ISR, and IMR), priority resolver and cascade buffer. This diagram shows all the elements of a programmable device plus additional blocks.

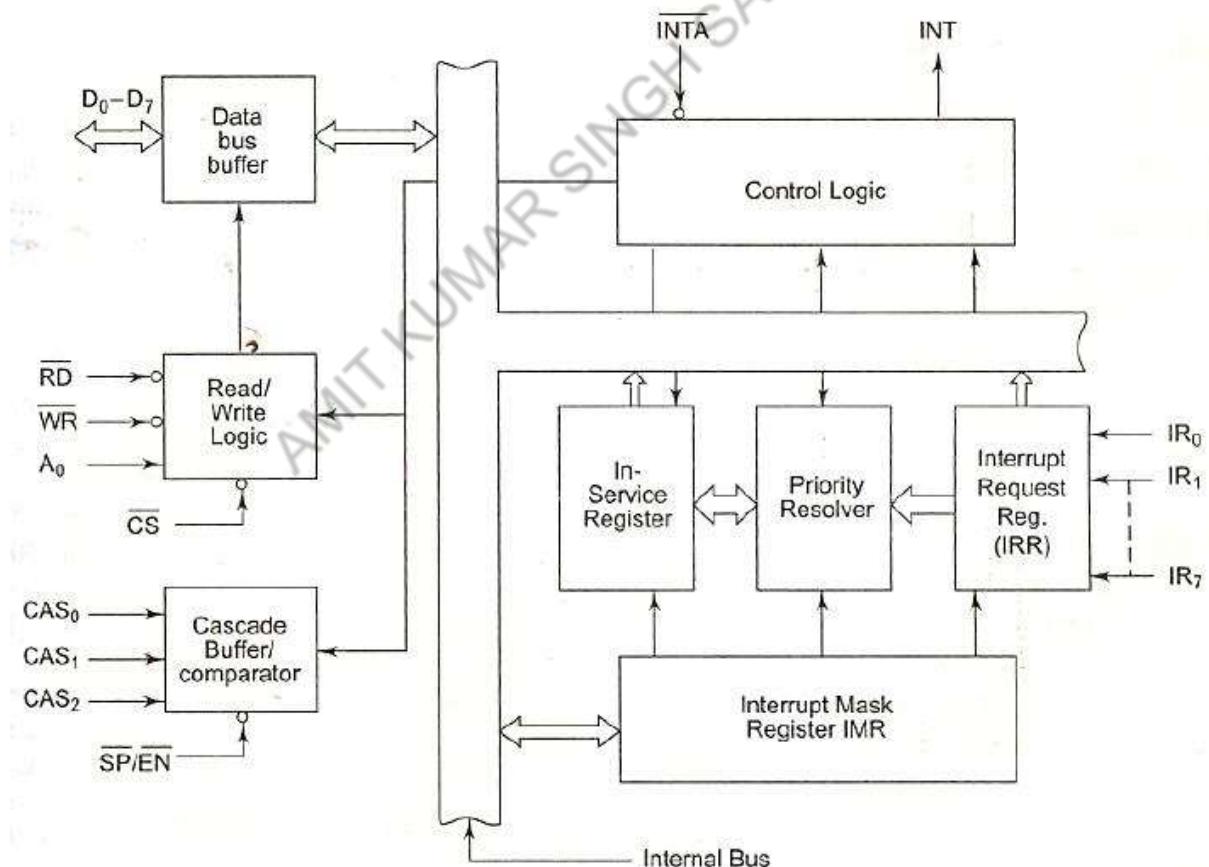
- **Data Bus Buffer:** This tristate bidirectional buffer interfaces internal 8259A bus to the microprocessor system data bus. Control words, status and vector information pass through data buffer during read or write operations.
This block has eight pins D7-D0.
- **Read /Write Logic:** This is a typical Read/Write control logic. When the address line A0 is at logic 0, the controller is selected to write a command or read a status. The Chip Select logic and A0 determine the port address of the controller.
This block has four pins:
 - RD: This is an active-low read enable input to 8259A
 - WR: This is an active-low write enable input to 8259A
 - A0: Address Pin, used to identify the various command words in conjunction with CS, RD and WR.
 - CS: This is an active-low chip select signal for enabling RD and WR operations of 8259A.
- **Control Logic:** This block has two pins:
 - INT (Interrupt) as an output, and
 - INTA (Interrupt Acknowledge) as an input.

The INT is connected to the interrupt pin of the MPU. Whenever a valid interrupt is asserted, this signal goes high. The INTA is the, Interrupt Acknowledgesignal.

- **Interrupt Registers and Priority Resolver:**
 - The **Interrupt Request Register (IRR)** has eight input lines (IR0-IR7) for interrupts. When these lines go high, the requests are stored in the register.
 - The **In-Service Register (ISR)** stores all the levels that are currently being serviced, and
 - The **Interrupt Mask Register (IMR)** stores the masking bits of the interrupt lines to be masked.
- **The Priority Resolver (PR)** examines these three registers and determines whether INT should be sent to the MPU.
- **Cascade Buffer/Comparator:** This block is used, to expand the number of interrupt levels by cascading two or more 8259As.

This block has four pins:

- CAS0-CAS2: These lines act as select lines for slaves 8259A in cascading mode.
- SP/ EN: SLAVE PROGRAM/ENABLE BUFFER: This is a dual function pin. When in the Buffered Mode it can be used as an output to control buffer transceivers (EN). When not in the buffered mode it is used as an input to designate a master ($\overline{SP} = 1$) or slave ($SP = 0$).



Interrupt Operation

- To implement interrupts, the Interrupt Enable flip-flop in the microprocessor should be enabled by writing the EI instruction, and the 8259A should be initialized by writing control words in the control register.
- The 8259A requires two types of control words:
 - Initialization Command Words (ICWs) and
 - Operational Command Words (OCWs).
- The ICWs are used to set up the proper conditions and specify RST vector addresses.
- The OCWs are used to perform functions such as masking interrupts, setting up status-read operations, etc. After the 8259A is initialized, the following sequence of events occurs when one or more interrupt request lines go high:
 1. The IRR stores the requests.
 2. The priority resolver checks three registers: the IRR for interrupt requests, the IMR for masking bits, and the ISR for the interrupt request being served. It resolves the priority and sets the INT high when appropriate.
 3. The MPU acknowledges the interrupt by sending INTA.
 4. After the INTA is received, the appropriate priority bit in the ISR is set to indicate which interrupt level is being served, and the corresponding bit in the IRR is reset to indicate that the request is accepted. Then, the opcode for the CALL instruction is placed on the data bus.
 5. When the MPU decodes the CALL instruction, it places two more INTA signals on the data bus.
 6. When the 8259A receives the second INTA, it places the low-order byte of the CALL address on the data bus. At the third INTA, it places the high-order byte on the data bus. The CALL address is the vector memory location for the interrupt; this address is placed in the control register during the initialization.
 7. During the third INTA pulse, the ISR bit is reset either automatically (Automatic-End-of-Interrupt-AEOI) or by a command word that must be issued at the end of the service routine (End-of-Interrupt-EOI). This option is determined by the initialization command word (ICW).
 8. The program sequence is transferred to the memory location specified by the CALL instruction.

- **Programming the 8259A**

The 8259A accepts two types of command words generated by the CPU:

1. Initialization Command Words (ICWs): Before it starts functioning, the 8259A must be initialized by writing two to four command words into the respective command word registers. These are called as Initialization Command Words (ICWs).

2. Operation Command Words (OCWs): These are the command words which command the 8259A to operate in various interrupt modes. These modes are:

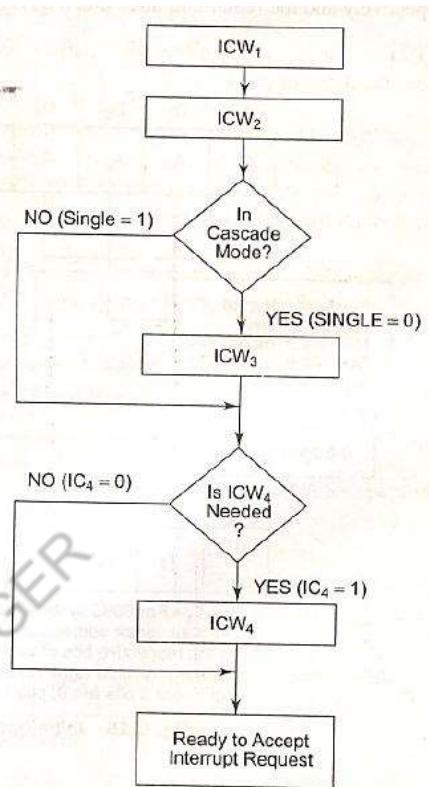
- a. Fully nested mode
- b. Rotating priority mode
- c. Special mask mode
- d. Polled mode

The OCWs can be written into the 8259A anytime after initialization.

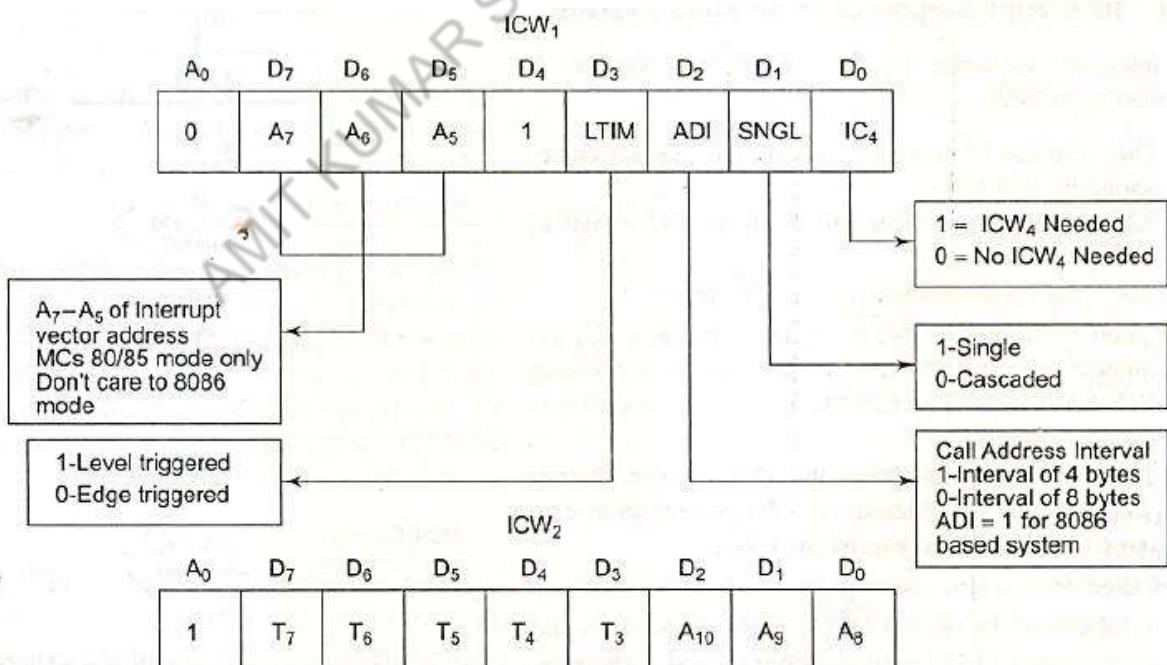
ICW1 and ICW2:

- If $A_0 = 0$ and $D_4 = 1$, the control word is recognized as ICW1, It contains the control bits for edge/level triggered mode, single/cascade mode, call address interval and whether ICW4 is required or not, etc.
- Once ICW₁ is loaded, the following initialization procedure is carried out internally.
 - The edge sense circuit is reset, i.e by default 8259A interrupts are edge sensitive
 - IMR is cleared
 - IR7 input is assigned the lowest priority
 - Slave mode address is set to 7
 - Special mask mode is cleared and the status read is set to IRR
 - If $IC_4 = 0$, all the functions of ICW4 are set to zero. Master/slave bit in ICW4 is used in the buffered mode only.
- If $A_0 = 1$, the control word is recognized as ICW2. The ICW2 stores details regarding interrupt vector addresses.

The initialization sequence of 8259A is described in form of a flow chart in figure.



Initialization Sequence of 8259A



$T_7 - T_3$ – For 8085 system they are filled by $A_{15} - A_{11}$ of the interrupt vector address and the least significant 3 bits are same as the respective bits of vector address. For 8086 system they are filled by most significant 5 bits of interrupt type and the least significant 3 bits are 0, pointing to Ir₀

ICW3: This word is read only when there is more than one 8259A in the system and cascading is used, in which case SNGL = 0. It will load the 8-bit slave register.

The functions of this register are:

a.) In the master mode (i.e. $\overline{SP} = 1$ or in buffer mode $M/S = 1$ in ICW4), the 8-bit slave register will be setbit-wise to '1' for each slave in the system, as shown in figure. The requesting slave will then release the second byte of a CALL sequence.

b.) In slave mode (i.e. $\overline{SP} = 0$ or if $BUF = 1$ and $M/S = 0$ in ICW4) bits D2 to D0 identify the slave, i.e. 000 to 111 for slave1 to slave8. The slave compares the cascade inputs with these bits and if they are equal, the second byte of the CALL sequence is released by it on the data bus.

ICW4: The use of this command word depends on the IC4 bit of ICW1. If IC4 = 1, ICW4 is used, otherwise it is neglected. The bit functions of ICW4 are described as follows:

- **SFNM** Special fully nested mode is selected, if SFNM = 1.
- **BUF** If $BUF = 1$, the buffered mode is selected. In the buffered mode, SP/EN acts as enable output and the master/slave is determined using the M/S bit of ICW4.
- **M/S** If $M/S = 1$, 8259A is a master. If $M/S = 0$, 8259A is a slave. If $BUF = 0$, M/S is to be neglected.
- **AEOI** If $AEOI = 1$, the automatic end of interrupt mode is selected.
- **mPM** If the mPM bit is 0, the MCS-85 system operation is selected and if mPM = 1, 8086/88 operation is selected.

Master mode ICW₃

A ₀	D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
1	S ₇	S ₆	S ₅	S ₄	S ₃	S ₂	S ₁	S ₀

S_n = 1 - IR_n Input has a slave

= 0 - IR_n Input does not have a slave

Slave mode ICW₃

A ₀	D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
1	0	0	0	0	0	ID ₂	ID ₁	ID ₀

→ D₂D₁D₀ - 000 to 111 for IR₀ to IR₇ or slave 1 to slave 8

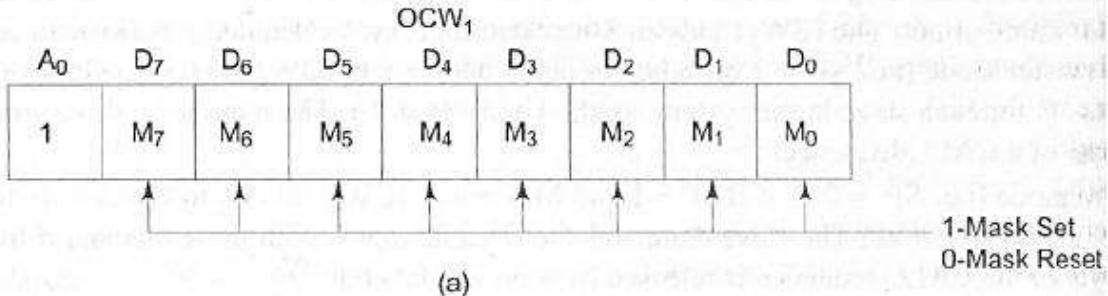
ICW₃ in Master and Slave Mode

ICW₄

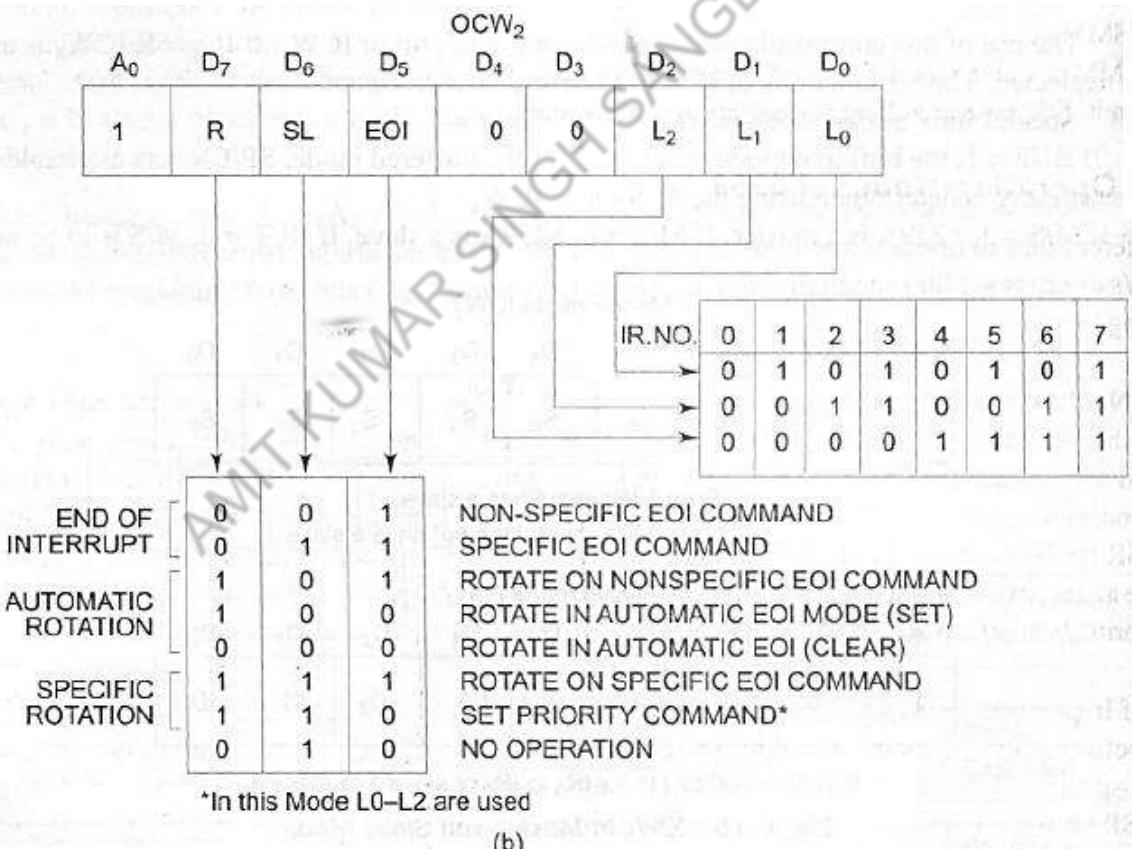
A ₀	D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
1	0	0	0	SFNM	BUF	M/S	AEOI	mPM

ICW₄ Bit Functions

Operation Control Word 1 (OCW1) OCW1 sets and clears the mask bits in the interruptMask Register (IMR). M7 to M0 represent the eight mask bits. M=1 indicates the channel is masked(inhibited), M=0 indicates the channel is enabled.



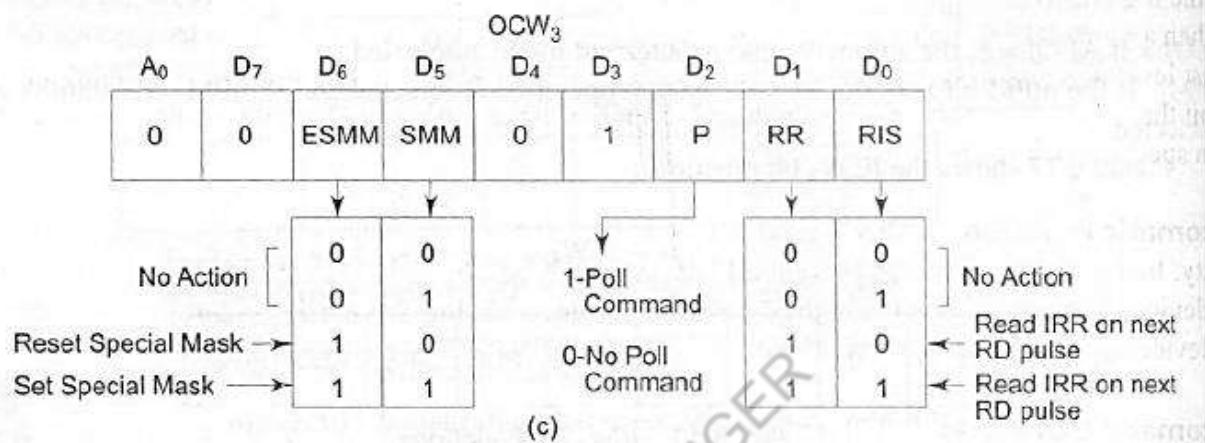
Operation Control Word 2 (OCW2) R, SL, EOI- These three bits control the Rotate and End of Interrupt modes and combinations of the two. A chart of these combinations can be found on the Operation Command Word Format. L2, L1, L0 -These bits determine the interrupt level acted upon when the SL bit is active.



Operation Control Word 3 (OCW3)

ESMM-Enable Special Mask Mode: When this bit is set to 1 it enables the SMM bit to set or reset the Special Mask Mode. When ESMM = 0 the SMM bit becomes a "don't care".

SMM-Special Mask Mode: If ESMM = 1 and SMM = 1 the 8259A will enter Special Mask Mode. If ESMM = 1 and SMM = 0 the 8259A will revert to normal mask mode. When ESMM = 0, SMM has no effect.

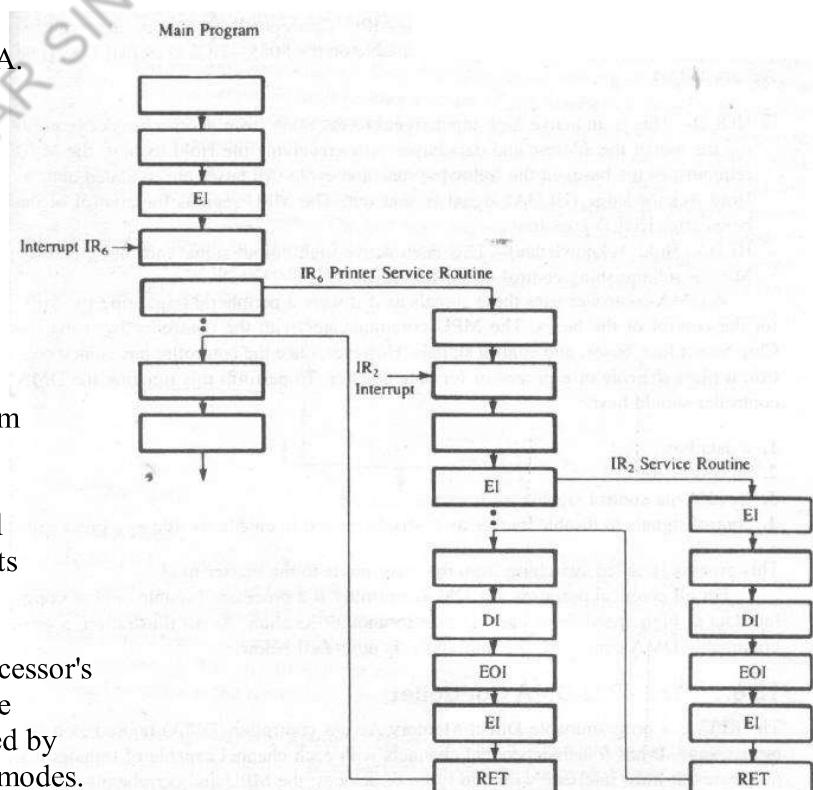


Operating Modes / Priority Modes

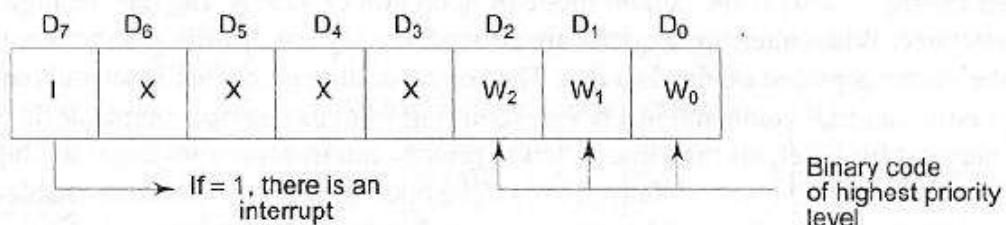
A number of operating/ Priority modes are available under software control in 8259A. They can be changed dynamically during the program by writing appropriate command words. Commonly used operating modes/ priority modes are as follows:

- **Fully Nested Mode:** This is the default mode of operation of 8259A. IR0 has the highest priority and IR7 has the lowest one. When interrupt requests are noticed, the highest priority request amongst them is determined and the vector is placed on the data bus. The corresponding bit of ISR is set and remains set till the microprocessor issues an EOI command just before returning from the service routine or the AEOI bit is set.

If the ISR (In Service) bit is set, all the same or lower priority interrupts are inhibited but higher levels will generate an interrupt, that will be acknowledged only if the microprocessor's Interrupt enable Flag (IF) is set. The priorities can afterwards be changed by programming the rotating priority modes.



- **Automatic Rotation (Equal Priority Devices):** This is used in the applications where all the interrupting devices are of equal priority. In this mode, an Interrupt Request (IR) level receives lowest priority after it is served while the next device to be served gets the highest priority in sequence. Once all the devices are served like this, the first device again receives highest priority.
- **Specific Rotation (Specific Priority):** This mode is similar to the automatic rotation mode, except that the user can select any IR for the lowest priority.
In this mode a bottom priority level can be selected. The selected bottom priority fixes other priorities. If IR5 is selected as a bottom priority, then IR5 will have least priority and IR4 will have a next higher priority. Thus IR6 will have the highest priority. These priorities can be changed during an EOI command.
- **End of Interrupt (EOI):** After the completion of an interrupt service, the corresponding ISR bit needs to be reset to update the information in the ISR. This is called the End-of-Interrupt (EOI) command. It can be issued in three formats:
 - **Non Specific EOI Command:** When non-specific EOI command is issued to 8259A it will automatically reset the highest ISR bit out of those already set. This command is used when 8259A is operated in fully nested structure is preserved.
 - **Specific EOI Command:** Specific EOI command is issued to reset a particular ISR bit. This command is used when 8259A is operated in a mode that may disturb the fully nested structure.
 - **Automatic EOI Mode:** In this mode, no command is necessary. During the third INTA the ISR bit is reset. This mode is used when a nested multilevel interrupt structure is not required with a single 8259A.
The major drawback with this mode is that the ISR does not have information on which IR is being serviced. Thus, any IR can interrupt the service routine, irrespective of its priority, if the Interrupt Enable flip-flop is set.
- **Reading 8259 Status:** The status of the internal registers of 8259A can be read using this mode. The OCW3 is used to read IRR and ISR while OCW1 is used to read IMR. Reading is possible only in no polled mode
- **Poll Command:** In the polled mode of operation, the INT output of 8259A is neglected. The poll mode is entered by setting P = 1 in OCW3.
The 8259A is polled by using software execution by microprocessor instead of the requests on INT input. The 8259A treats the next RD pulse to the 8259A as an interrupt acknowledge. An appropriate ISR bit is set, if there is a request. The priority level is read and a data word is placed on to data bus, after RD is activated. The data word is shown in figure. A poll command may give you more than 64 priority levels.



- **Special Fully Nested Mode:** This mode is used in more complicated systems, where cascading is used and is somewhat similar to the normal nested mode.

In this mode, when an interrupt request from a certain slave is in service, this slave can further send requests to the master, if the requesting device connected to the slave has higher priority than the one being currently served.

In this mode, the master interrupts the CPU only when the interrupting device has a higher or the same priority than the one currently being served. In normal mode, other requests than the one being served are masked out.

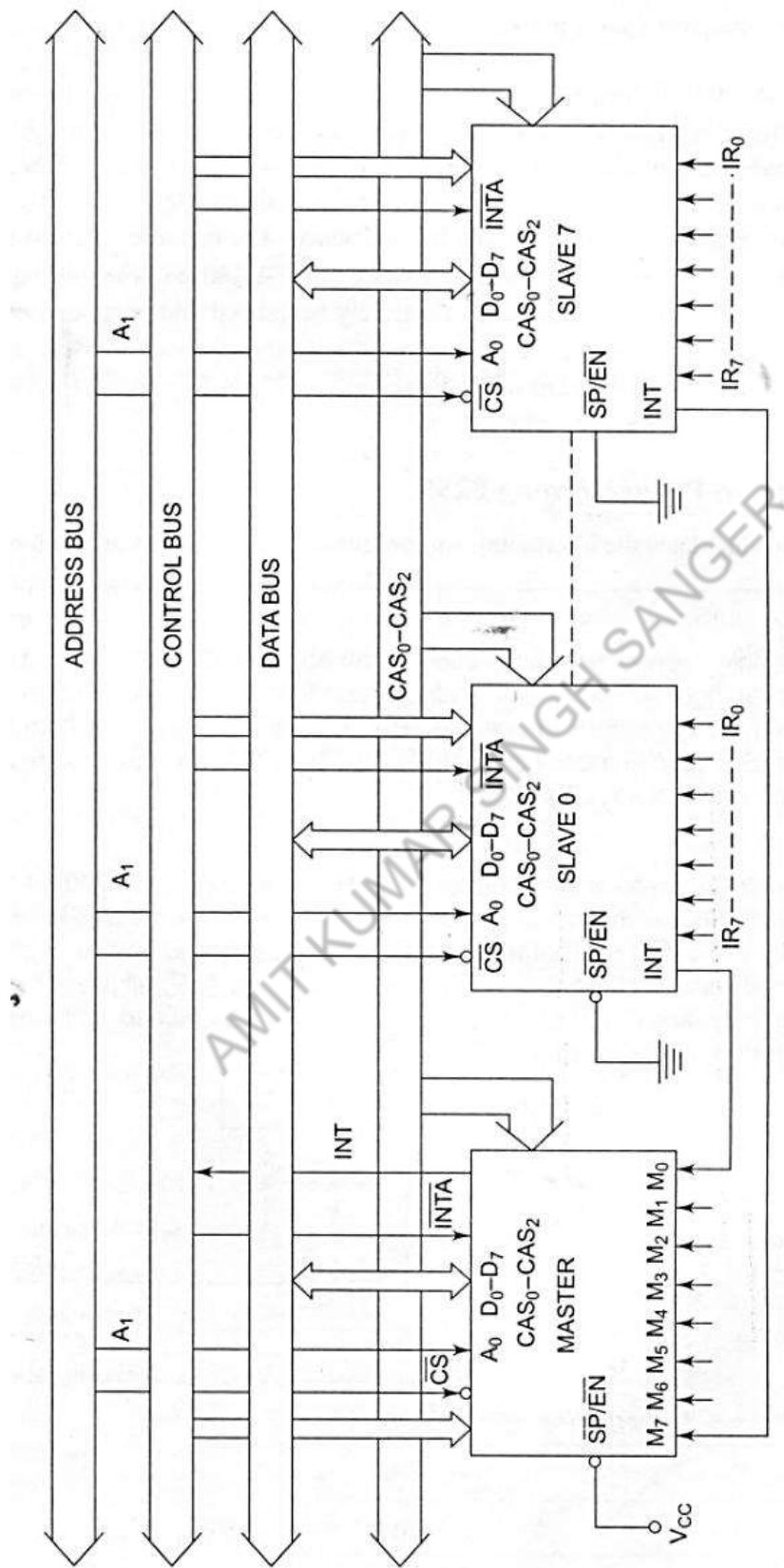
This mode is important, since in the absence of this mode, the slave would interrupt the master only once and hence the priorities of the slave inputs would have been disturbed.

- **Buffered Mode:** When the 8259A is used in the systems in which bus driving buffers are used on data buses (e.g. cascade system, the problem of enabling the buffers arises). The 8259A sends a buffer enable signal on $\overline{SP} / \overline{EN}$ pin, whenever data is placed on the bus.

- **Cascade Mode:** The 8259A can be connected in a system containing one master and eight slaves (maximum) to handle upto 64 priority levels. The master controls the slaves using CAS0-CAS2 which act as chip select inputs (encoded) for slaves.

In this mode, the slave INT outputs are connected with master IR inputs. When a slave request line is activated and acknowledged, the master will enable the slave to release the vector address during the second pulse of \overline{INTA} sequence. The cascade lines are normally low and contain slave address codes from the trailing edge of the first \overline{INTA} pulse to the trailing edge of the second \overline{INTA} pulse.

Figure shows the details of the circuit connections of 8259As in cascade scheme.

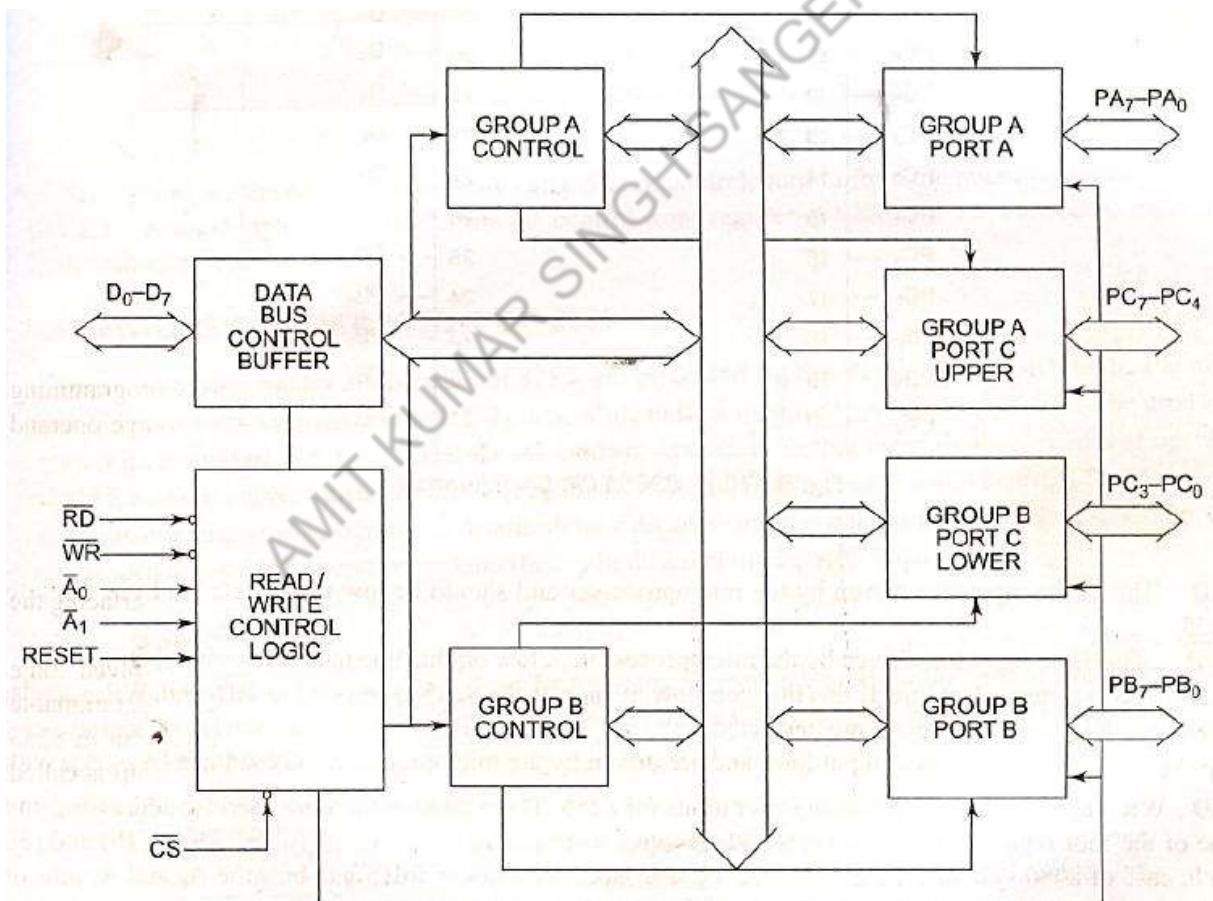


8259A in Cascaded Mode

8255 A PROGRAMMABLE PERIPHERAL INTERFACE

- The 8255A is a widely used, programmable, parallel I/O device. It can be programmed to transfer data under various conditions, from simple I/O to interrupt I/O.
- It is flexible, versatile, and economical (when multiple I/O ports are required), but somewhat complex.
- It is an important general-purpose I/O device that can be used with almost any microprocessor.
- The 8255A has 24 I/O pins that can be grouped primarily in two 8-bit parallel A and B, with the remaining eight bits as port C. The eight bits of port C can be used as individual bits or be grouped in two 4-bit ports: CUPPER (C_U) and CLOWER (C_L). The functions of these ports are defined by writing a control word in the control register.

Block Diagram of the 8255A



8255 Internal Architecture

The block diagram in Figure shows

- Two 8-bit ports (A and B),
- Two 4-bit ports (C_U and C_L),
- The data bus buffer, and
- Control logic.

Figure shows a simplified but expanded version of the internal structure, including a control register. This block diagram includes all the elements of a programmable device; port C performs functions similar that of the status register in addition to providing handshake signals.

Two 8-bit ports (A and B): These are eight Port A and Port B lines that act as either latched output or buffered input lines depending upon the control word loaded into the control word register.

Two 4-bit ports (C_U and C_L):

- $PC7-PC4$: Upper nibble of Port C lines (C_U). They may act as either output latches or input buffers lines. This port also can be used for generation of handshake lines in mode 1 or mode 2.
- $PC3-PC0$: These are the lower Port C lines (C_L). They may act as either output latches or input buffers lines. This port also can be used for generation of handshake lines in mode 1 or mode 2.

The data bus buffer: This 3-state bidirectional 8-bit buffer is used to interface the 82C55A to the system data bus. Data is transmitted or received by the buffer upon execution of input or output instructions by the CPU. Control words and status information are also transferred through the data bus buffer

Control logic: The control section has six lines. Their functions and connections are as follows:

- \overline{RD} (Read): This control signal enables the read operation. When the signal is low the MPU reads data from a selected I/O port of the 8255A.
- \overline{WR} (Write): This control signal enables the Write operation. When the signal is low, the MPU writes into a selected I/O port or the control register.
- RESET (Reset): This is an active high signal; it clears the control register and sets all ports in the input mode.
- \overline{CS} , A_0 and A_1 : These are device select signals. \overline{CS} is connected to a decoded address, and A_0 and A_1 are generally connected to MPU address lines A_0 and A_1 , respectively.

The \overline{CS} signal is the master Chip Select, and A_0 and A_1 specify one of the I/O ports or the control register as given below:

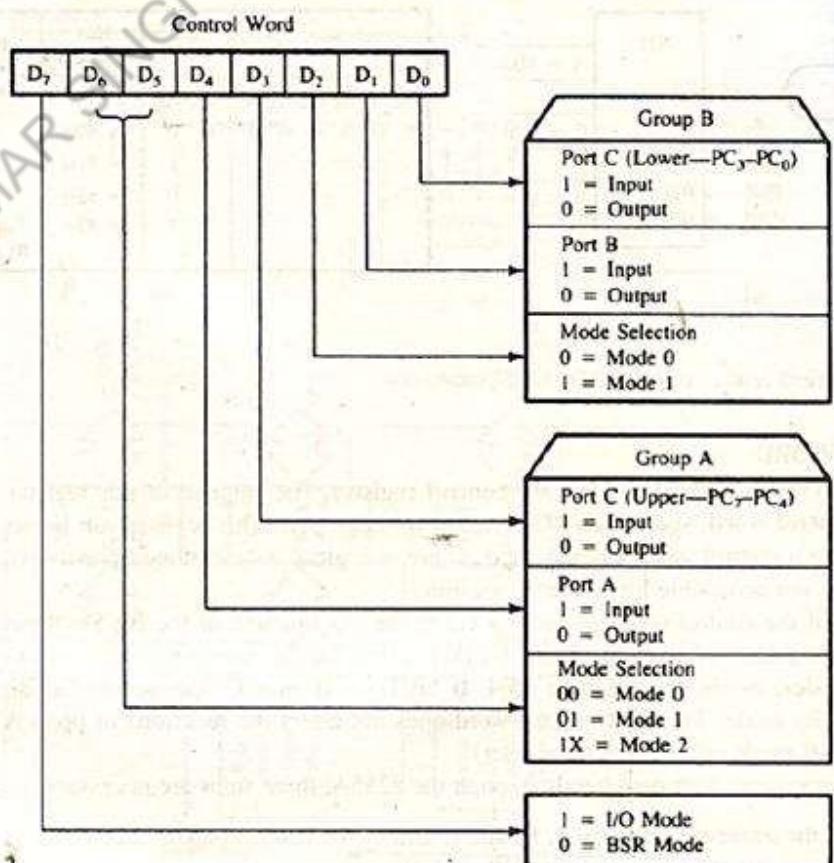
\overline{CS}	A_1	A_0	Selected
0	0	0	Port A
0	0	1	Port B
0	1	0	Port C
0	1	1	Control Register
1	X	X	8255A is not selected.

These lines ($A_1 - A_0$) with \overline{RD} , \overline{WR} and \overline{CS} form the following operations for 8255.

\overline{RD}	\overline{WR}	\overline{CS}	A_1	A_0	Function (Input / Output Cycle)
0	1	0	0	0	Port A to Data Bus
0	1	0	0	1	Port B to Data Bus
0	1	0	1	0	Port C to Data Bus
0	1	0	1	1	CWR to Data Bus
1	0	0	0	0	Data Bus to Port A
1	0	0	0	1	Data Bus to Port B
1	0	0	1	0	Data Bus to Port C
1	0	0	1	1	Data Bus to CWR
1	1	0	x	x	Data Bus Tristated
X	x	1	x	x	Data Bus Tristated

Control Word:

- Block diagram of 8255A shows a register called the control register. The contents of this register, called the control word, specify an I/O function for each port. This register can be accessed to write a control word when A_0 and A_1 , are at logic 1 as mentioned in table.
- The register is not accessible for a Read operation.
- Bit D7 of the control register specifies either the I/O function or the Bit Set/Reset function.
- If bit D7 = 1, bits D6-D0 determine I/O functions in various modes.



8255A Control Word Format for I/O Mode

Steps to communicate with peripherals

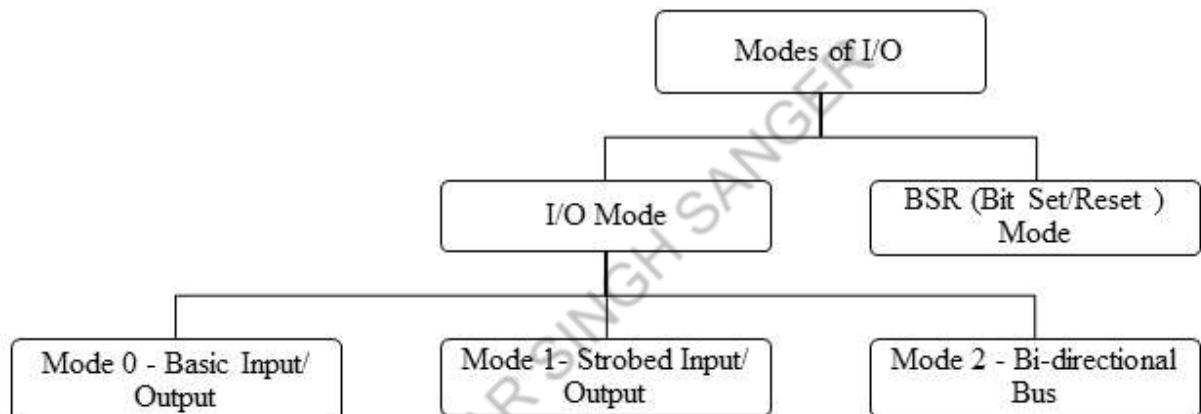
To communicate with peripherals through the 8255A, three steps are necessary:

1. Determine the addresses of ports A, B and C and of the control register according to the Chip Select logic and address lines A0 and A1.
2. Write a control word in the control register.
3. Write I/O instructions to communicate with peripherals through ports A, B and C.

Modes of I/O functions

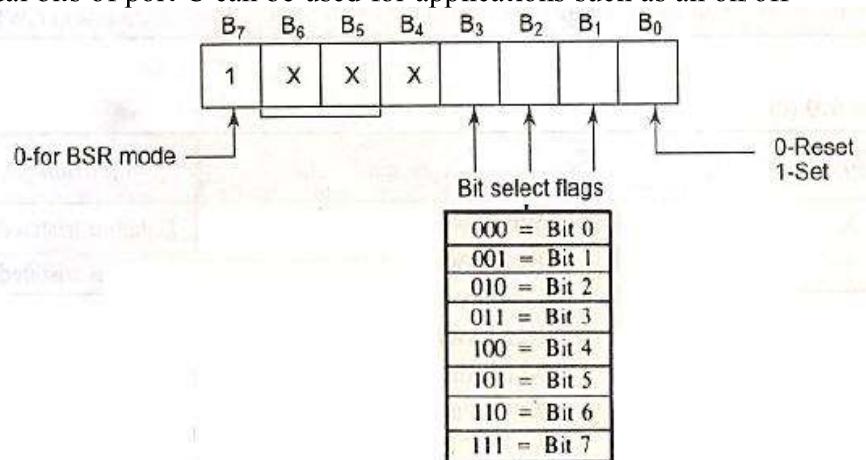
There are three basic modes of operation that can be selected by the system software:

1. BSR (Bit Set/Reset) Mode
2. Mode 0 - Basic input/output
3. Mode 1 - Strobed Input/output
4. Mode 2 - Bi-directional Bus



1. BSR (Bit Set/Reset) Mode:

- The BSR mode is concerned only with the eight bits of port C, which can be set or reset by writing an appropriate control word in the control register.
- A control word with bit D7= 0 is recognized as a BSR control word and it does not alter any previously transmitted control word with bit D7 = 1; thus the I/O operations of ports A and B are not affected by a BSR control word.
- In the BSR mode, individual bits of port C can be used for applications such as an on/off switch.
- **BSR CONTROL WORD**
This control word, when written in the control register sets or resets one bit at a time as specified in Figure



BSR Mode Control Word Register Format

2. Mode 0 - Basic input/output:

- This mode is also known as basic input/output mode.
- This mode provides simple input and output capability using each of the three ports.
- Data can be simply read from and written to the input and output ports respectively, after appropriate initialization.
- The salient features of this mode are:
 - o Two 8-bit ports (port A and port B) and two 4-bit ports (port C upper and lower) are available. The two 4-bit ports can be combinedly used as a third 8-bit port.
 - o Any port can be used as an input or output port.
 - o Output ports are latched. Input ports are not latched.
 - o A maximum of four ports are available so that overall 16 I/O configurations are possible.
 - o Ports do not have handshake or interrupt capability.

3. Mode 1 - Strobed Input/output:

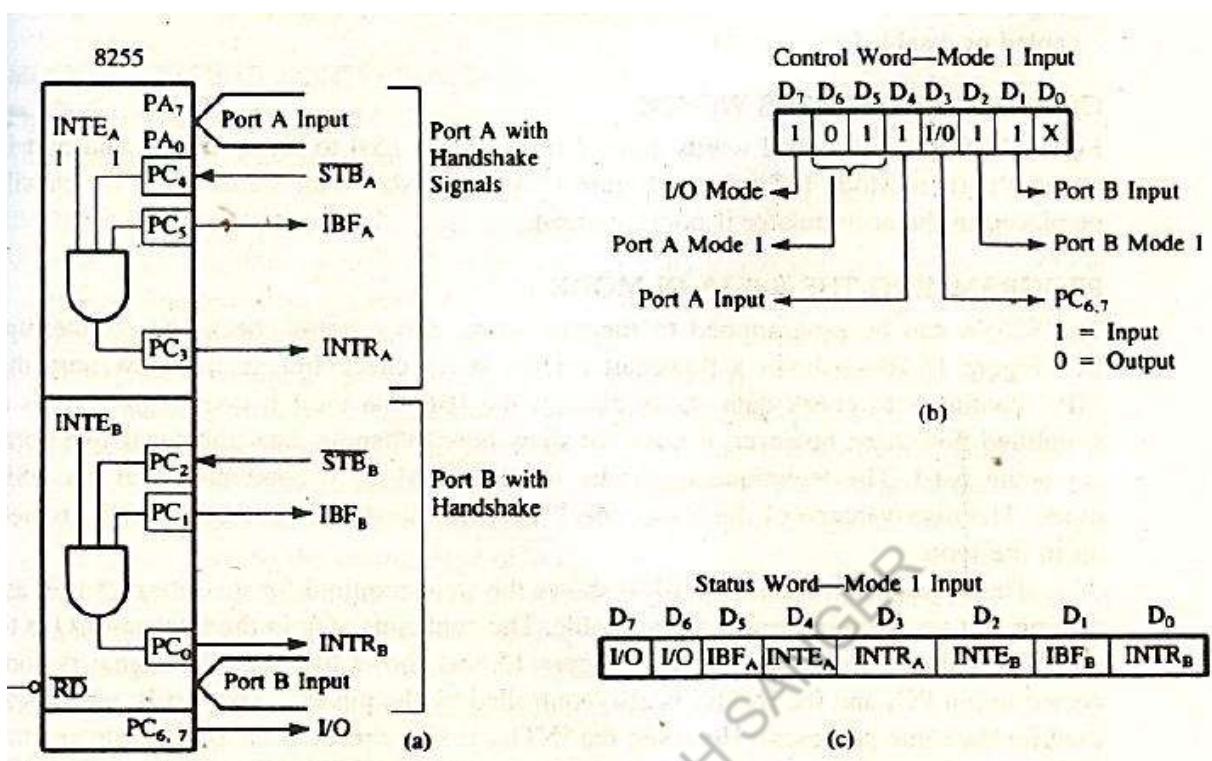
- This mode is also called as strobed input/output mode.
- In this mode the handshaking signals control the input or output action of the specified port.
 - o Port C lines PC0-PC2; provide strobe or handshake lines for port B. This group which includes port B and PC0-PC2 is called as group B for strobed data input/output.
 - o Port C lines PC3-PC5 provides strobe lines for port A. This group including port A and PC3-PC5 forms group A. Thus port C is utilized for generating handshake signals.
- The salient features of mode I are listed as follows:
 - (i) Two groups-group A and group B are available for strobed data transfer.
 - (ii) Each group contains one 8-bit data I/O port and one 4-bit control/data port.
 - (iii) The 8-bit data port can be either used as input or an output port. Both the inputs and outputs are latched.
 - (iv) Out of 8-bit port C, PC0-PC2 are used to generate control signals for port B and PC3-PC5 are used to generate control signals for port A. The lines PC6, PC7 may be used as independent data lines.

Input control signal definitions (mode 1)

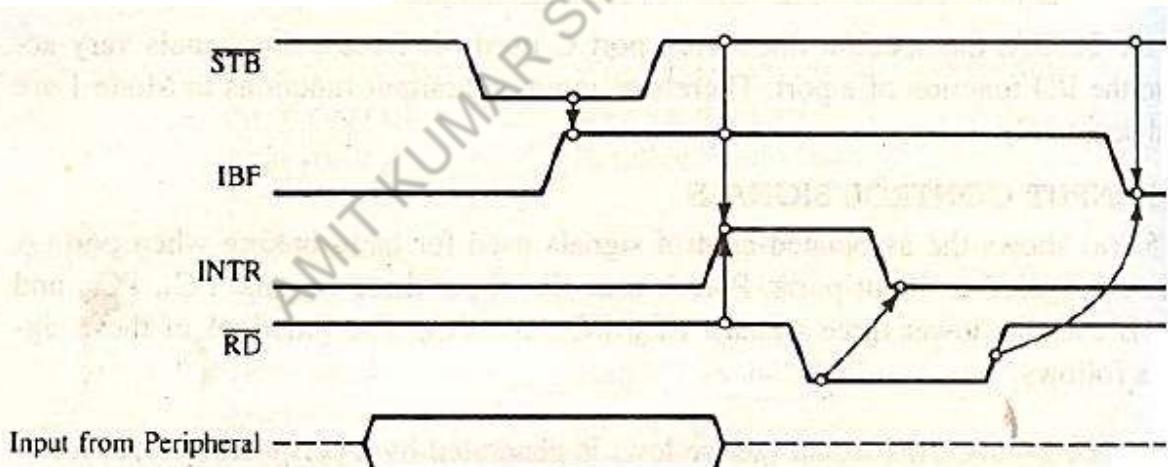
STB (Strobe input) - If this line falls to logic low level, the data available at 8-bit input port is loaded into input latches.

IBF (Input buffer full) - If this signal rises to logic 1, it indicates that data has been loaded into the latches, i.e. it works as an acknowledgement. IBF is set by a low on STB and is reset by the rising edge of RD input.

INTR (Interrupt request) - This active high output signal can be used to interrupt the CPU whenever an input device requests the service. INTR is set by a high at STB pin and a high at IBF pin. INTE is an internal flag that can be controlled by the bit set/reset mode of either PC4 (INTE_A) or PCl (INTE_B). INTR is reset by a falling edge on RD input. Thus an external input device can request the service of the processor by putting the data on the bus and sending the strobe signal.



8255A Mode 1: Input Configuration



8255A Mode 1: Timing Waveforms for Strobed Input (with Handshake)

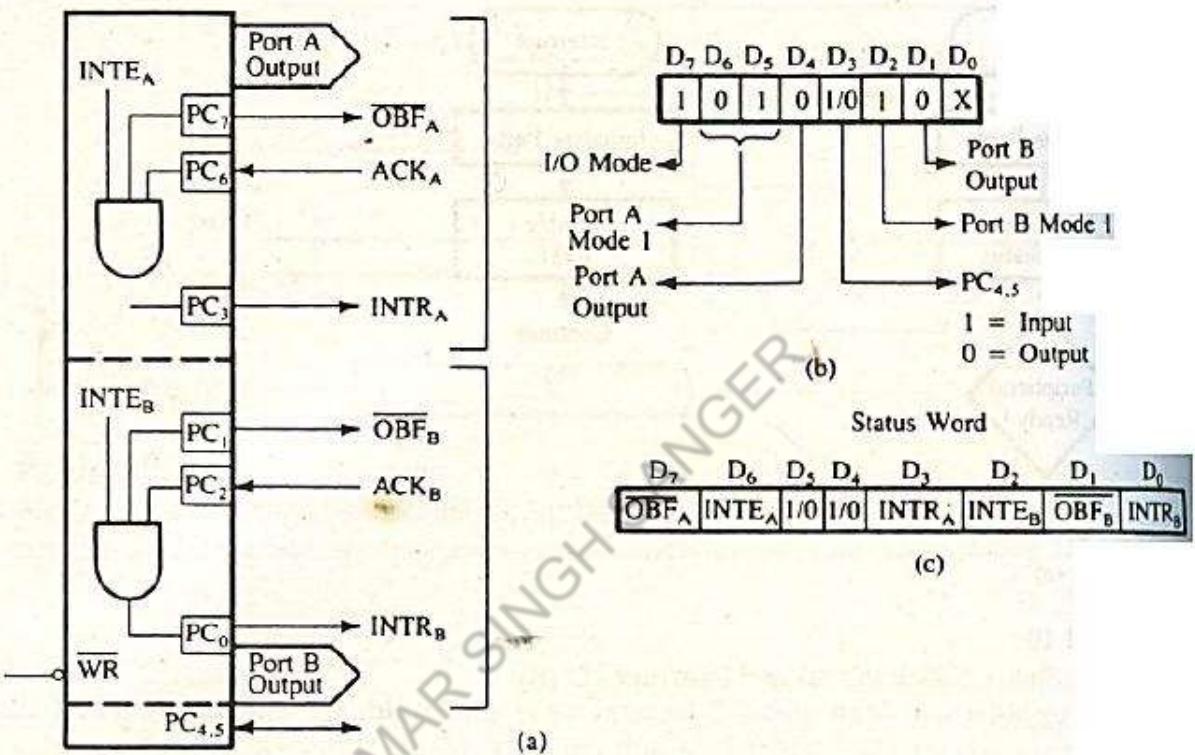
- **Output control signal definitions (mode 1)**

OBF (*Output buffer full*) - This status signal, whenever falls to logic low, indicates that the CPU has written data to the specified outport. The OBF flip-flop will be set by a rising edge of WR signal and reset by a low going edge at the ACK input.

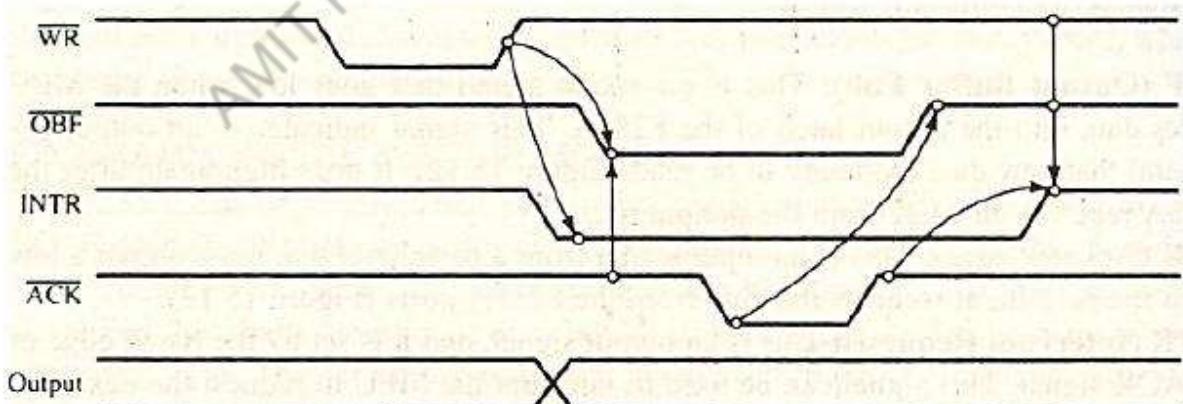
ACK (*Acknowledge input*) - ACK signal acts as an acknowledgement to be given by an output device. ACK signal, whenever low, informs the CPU that the data transferred by the CPU to the output device through the port is received by the output device.

INTR (Interrupt request) - Thus an output signal that can be used to interrupt the CPU when an output device acknowledges the data received from the CPU. INTR is set when ACK, OBF and INTE are '1'. It is reset by a falling edge on WR input. The INTE_A and INTE_B flags are controlled by the bit set-reset mode of PC6 and PC1 respectively.

PC_{4,5}: These two lines can be set up either as input or output.



8255A Mode 1: Output Configuration



8255 Mode 1: Timing Waveforms for Strobed (with Handshake) Output

4. Mode 2 - Bi-directional Bus:

- This mode of operation of 8255 is also known as strobed bidirectional I/O.
- This mode of operation provides 8255 with an additional feature for communicating with a peripheral device on an 8-bit data bus.
- Handshaking signals are provided to maintain proper data flow and synchronization between the data transmitter and receiver.
- The interrupt generation and other functions are similar to mode 1. Thus in this mode, 8255 is a bidirectional 8-bit port with handshake signals.
- The RD and WR signals decide whether the 8255 is going to operate as an input port or output port.
- The salient features of mode 2 of 8255 are listed as follows:
 1. The single 8-bit port in group A is available.
 2. The 8-bit port is bidirectional and additionally a 5-bit control port is available.
 3. Three I/O lines are available at port C, viz. PC2-PC0.
 4. Inputs and outputs are both latched.
 5. The 5-bit control port C (PC3-PC7) is used for generating/accepting handshake signals for the 8-bit data transfer on port A.

Control signal definitions in mode 2

INTR (Interrupt request) - As in mode 1, this control signal is active high and is used to interrupt the microprocessor to ask for transfer of the next data byte to/from it. This signal is used for input (read) as well as output (write) operations.

Control signals for output operations

OBF (Output buffer full) - This signal, when falls to logic low level, indicates that the CPU has written data to port A.

ACK (Acknowledge) - This control input, when falls to logic low level, acknowledges that the previous data byte is received by the destination and the next byte may be sent by the processor. This signal enables the internal tristate buffers to send out the next data byte on port A.

INTE1 (A flag associated with OBF) - This can be controlled by bit set/reset mode with PC6.

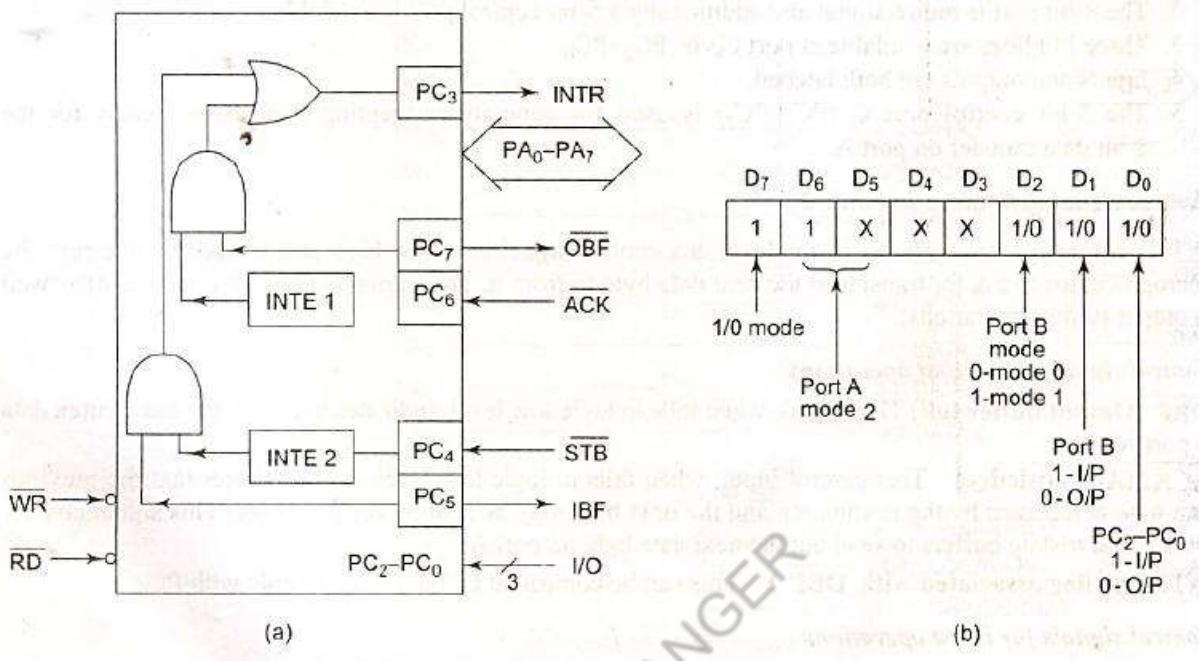
Control signals for input operations

STB (Strobe input) - A low on this line is used to strobe in the data into the input latches of 8255.

IBF (Input buffer full) - When the data is loaded into the input buffer, this signal rises to logic' 1 '. This can be used as an acknowledgement that the data has been received by the receiver.

WR must occur before ACK and STB must be activated before RD.

Figure (a) shows a schematic diagram containing an 8-bit bidirectional port, 5-bit control port and the relation of INTR with the control pins. Port B can either be set to mode 0 or mode 1 while port A (Group A) is in mode 2. Mode 2 is not available for port B. Figure (b) shows the necessary control word. The INTR goes high only if either IBF, INTE2, STB and RD go high or OBF, INTE1, ACK and WR go high. The port C can be read to know the status of the peripheral device, in terms of the control signals, using the normal I/O instructions.



Mode 2 pins (b) Mode 2 control word

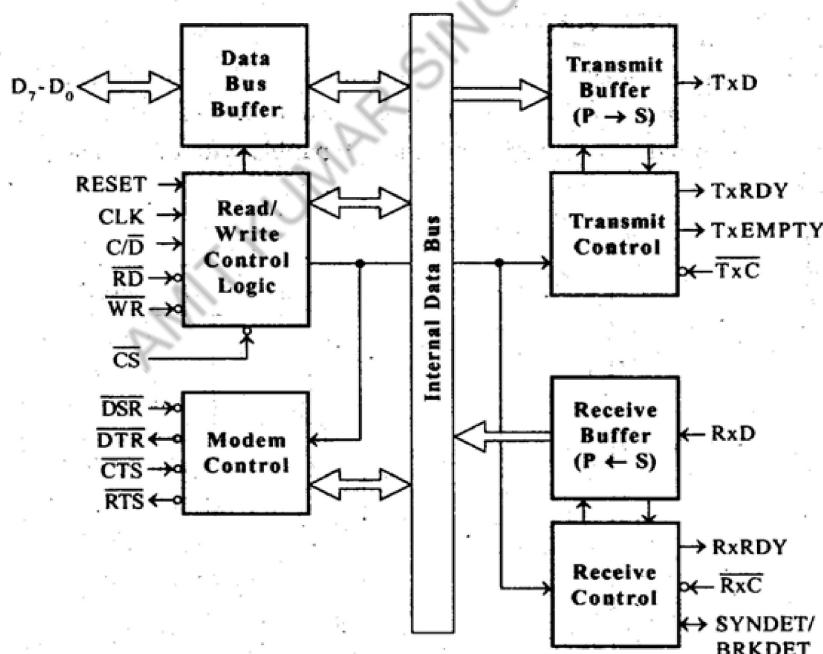
8251A (USART)

- The 8251 is a programmable chip designed for synchronous and asynchronous serial data communication
- The INTEL 8251 is the industry standard Universal Synchronous/Asynchronous Receiver/Transmitter (USART) designed for data communications
- The 8251 is used as a peripheral device and is programmed by the CPU to operate using virtually any serial data transmission technique
- The USART accepts data characters from the CPU in parallel format and then converts them into a continuous serial data stream for transmission
- Simultaneously, it can receive serial data streams and convert them into parallel data character for the CPU
- The USART will signal the CPU whenever it can accept a new character for transmission or whenever it has received a character for the CPU

Block Diagram :

The block diagram includes five section

- Read/write Control Logic
- Transmitter
- Receiver
- Data Bus Buffer and
- Modem Control

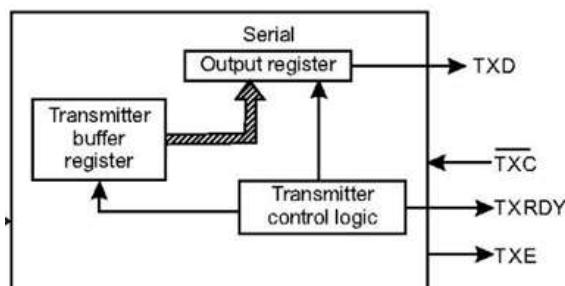


Read/Write control logic:

- The Read/Write Control logic interfaces the 8251A with CPU, determines the functions of the 8251A according to the control word written into its control register.
- It monitors the data flow.
- This section has three registers and they are control register, status register and data buffer.
- The active low signals RD, WR, CS and $C\bar{D}$ are used for read/write operations with these three registers.

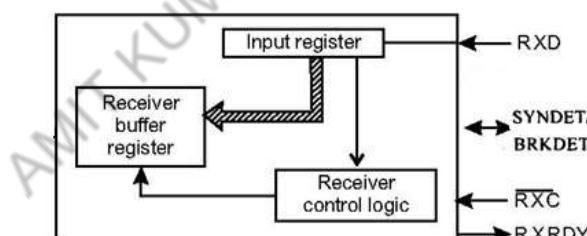
- When C/D is high, the control register is selected for writing control word or reading status word.
- When C/D is low, the data buffer is selected for read/write operation.
- When the reset is high, it forces 8251A into the idle mode.
- The clock input is necessary for 8251A for communication with CPU and this clock does not control either the serial transmission or the reception rate.

Transmitter section:



- The transmitter section accepts parallel data from CPU and converts them into serial data.
- The transmitter section is double buffered, i.e., it has a buffer register to hold an 8-bit parallel data and another register called output register to convert the parallel data into serial bits.
- When output register is empty, the data is transferred from buffer to output register. Now the processor can again load another data in buffer register.
- If buffer register is empty, then TxRDY is goes to high.
- If output register is empty then TxEMPTY goes to high.
- The clock signal, TxC (low) controls the rate at which the bits are transmitted by the USART.
- The clock frequency can be 1,16 or 64 times the baud rate.

Receiver Section:



- The receiver section accepts serial data and convert them into parallel data
- The receiver section is double buffered, i.e., it has an input register to receive serial data and convert to parallel, and a buffer register to hold the parallel data.
- When the RxD line goes low, the control logic assumes it as a START bit, waits for half a bit time and samples the line again.
- If the line is still low, then the input register accepts the following bits, forms a character and loads it into the buffer register.
- The CPU reads the parallel data from the buffer register.
- When the input register loads a parallel data to buffer register, the RxRDY line goes high.
- The clock signal RxC (low) controls the rate at which bits are received by the USART.
- During asynchronous mode, the signal SYNDET/BRKDET will indicate the break in the data transmission.
- During synchronous mode, the signal SYNDET/BRKDET will indicate the reception of synchronous character.

Data Bus Buffer

This is 8 bit bidirectional register can be addressed as an input and an output port when the **C/D** pin is low

Control/Data pin (**C/D**) : When this signal is high, the control register or the status register is addressed; when it is low, the data buffer is addressed. The control register and the status register are differentiated by and signals respectively.

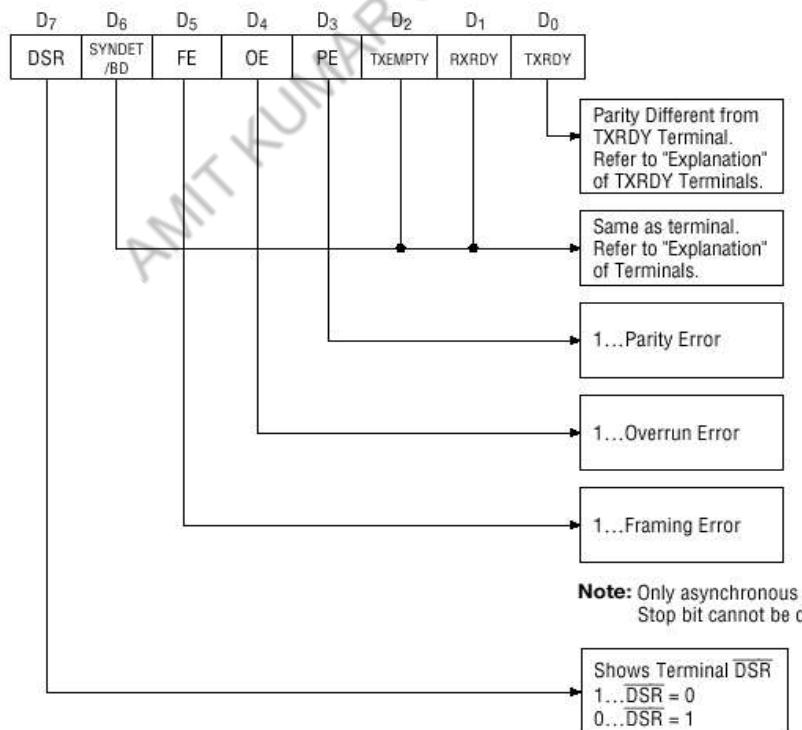
MODEM Control:

- DSR - Data Set Ready : Checks if the Data Set is ready when communicating with a modem.
- DTR - Data Terminal Ready : Indicates that the device is ready to accept data when the 8251 is communicating with a modem.
- CTS - Clear to Send : If its low, the 8251A is enabled to transmit the serial data provided the enable bit in the command byte is set to '1'.
- RTS - Request to Send Data : Low signal indicates the modem that the receiver is ready to receive a data byte from the modem.

Command register

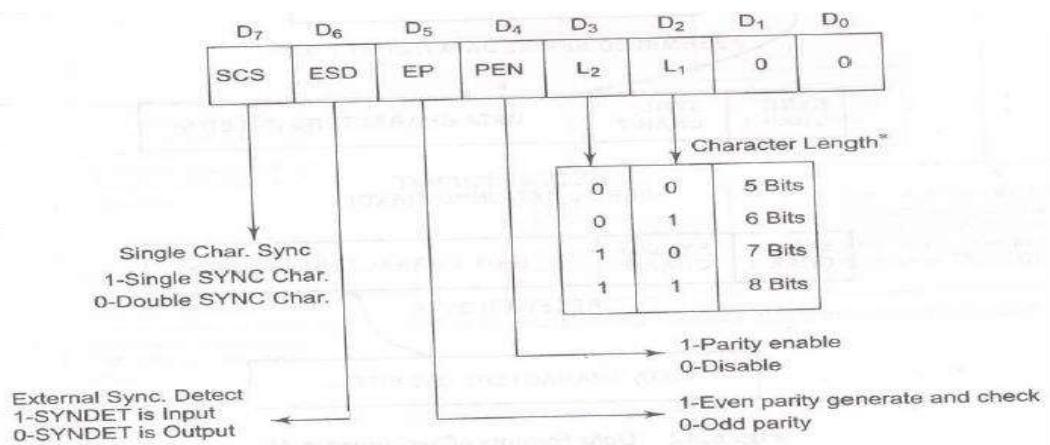
EH	IR	RTS	ER	SBRK	RxE	DTR	TxE
TxE:	transmit enable		ER:	error reset			
DTR:	data terminal ready		RTS:	request to send			
RxE:	receiver enable		IR:	internal reset			
➤ SBPRK:	send break character		EH:	enter hunt mode			

Status Register

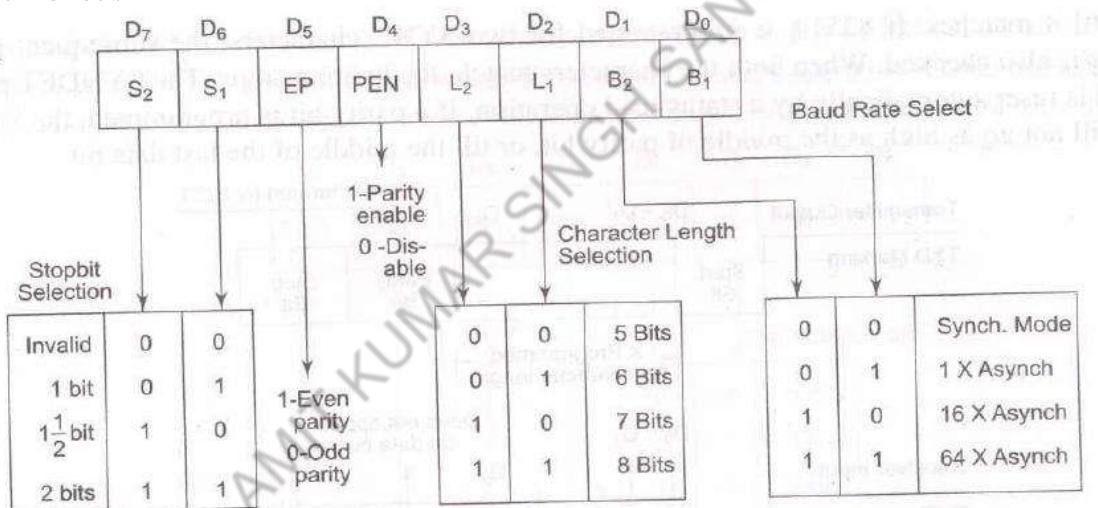


Mode Register :

Synchronous

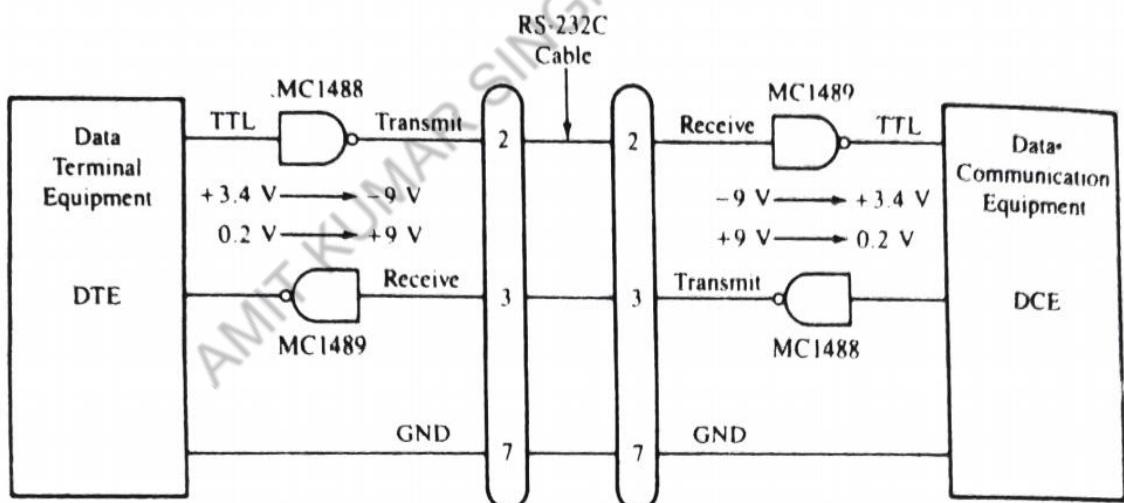


Asynchronous

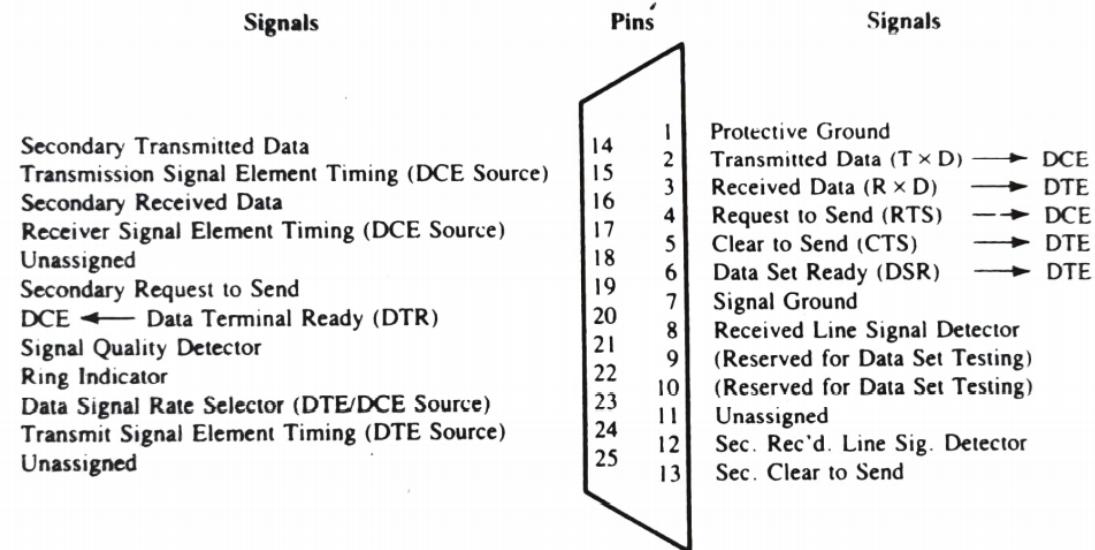


RS232C

- The serial I/O technique is commonly used to interface terminals, printers, and modems.
- These peripherals and computers are designed and manufactured by various manufacturers. Therefore, a common understanding must exist among various manufacturing and user groups, that can ensure compatibility among different equipment. When this understanding is defined and generally accepted in industry (and by users), it is known as a standard.
- A standard is normally defined by a professional organization (such as IEEE Institute of Electrical and Electronics Engineers).
- A standard may include such items as assignment of pin positions for signals, voltage levels, speed of data transfer, length of cables and mechanical specifications.
- In serial I/O, data can be transmitted as either current or voltage.
 - When current is used, typically 20 mA (or 60 mA) current loops are used in teletype equipment. When a teletype is at logic 1, current flows; when it is at logic 0, the current flow is interrupted.
The advantage of the current loop method is that signals are relatively noise-free and are suitable for transmission over a distance.
 - When data are transmitted as voltage, the commonly used standard is known as RS232C. It is defined in reference to Data Terminal Equipment (DTE) and Data Communication Equipment (DCE)-terminal and modem.



- However, its voltage levels are not compatible with TTL logic levels.
- The rate of data transmission in RS-232C is restricted to a maximum of 20 kbaud and a distance of 50 ft.
- The RS-232 standard was developed during the initial days of computer timesharing and its primary focus was to have compatibility between a terminal and a modem.
- However, the same standard is now being used for communications between computers and peripherals, and the roles of a data terminal and a modem have become ambiguous.
- Figure shows the RS-232C 25-pin connector and its signals.



- The signals are divided into four groups:
 - data signals,
 - control signals,
 - timing signals, and
 - grounds.

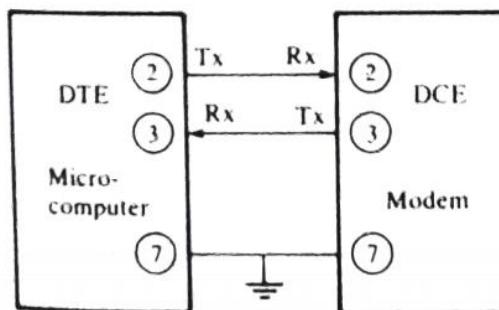
For data lines, the voltage level +3V to +15 V is defined as logic 0; from -3V to -15 V is defined as logic 1 (normally, voltage levels are ± 12 V). This is negative true logic. Other signals (control and timing) are compatible with the TTL level.

Due to incompatibility of the data lines with the TTL logic, voltage translators (line drivers and line receivers) are required to interface TTL logic with the RS-232 signals.

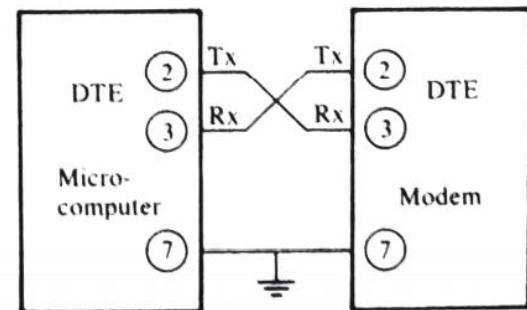
- The line driver, MC1488, converts logic 1 into approximately -9 V and logic 0 into +9 V. Before it is received by the DCE, it is again converted by the line receiver, MC1489, into TTL-compatible logic.

The minimum interface between a computer and a peripheral requires three lines: pins 2, 3, and 7.

- These lines are defined in relation to the DTE: the terminal transmits on pin 2 and receives on pin 3.
 - On the other hand, the DCE transmits on pin 3 and receives on pin 2.
 - For example,
 - The user may connect its microcomputer to a serial printer configured as a DTE. Therefore, to remain compatible with the defined signals of the RS-232C, the RS-232 cable must be reconfigured as shown in Figure (b).
 - In Figure given below the microcomputer is defined as a DTE, and it can be connected to the modem, defined as a DCE, without any modification in the RS-232 cable, as shown in Figure (a).
- However, when it is connected to the printer, the transmit and the receive lines must be crossed as shown in Figure (b); this is known as a null-modem connection.



(a)



(b)

RS-232C Connections (a) DTE to DCE and (b) DTE to DTE

- Typically, data transmission with a handshake requires eight lines listed in Table.

RS-232C Signals Used with Handshake Data Communication

Pin No.	Signals*		Functions
2	Transmitted Data	TxD	Output; transmits data from DTE to DCE
3	Received Data	RxD	Input; DTE receives data from DCE
4	Request to Send	RTS	General-purpose output from DTE
5	Clear to Send	CTS	General-purpose input to DTE; can be used as a handshake signal
6	Data Set Ready	DSR	General-purpose input to DTE; can be used to indicate that DCE is ready
7	Signal Ground	GND	Common reference between DTE and DCE
8	Data Carrier Detect	DCD	Generally used by DTE to disable data reception
20	Data Terminal Ready	DTR	Output; generally used to indicate that DTE is ready

^aSignals are referenced to DTE.

Specific functions of handshake lines differ in different peripherals and, therefore should be referred to in the manufacturer's manuals.