SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE
www.sathyabama.ac.in

SCHOOL OF ELECTRICAL AND ELECTRONICS

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

UNIT- 3 INTERFACING

SECA1404- MICROPROCESSOR ANDMICROCONTROLLER BASED SYSTEMS

III. INTERFACING

I. BASIC INTERFACE CONCEPTS

1. INTERFACING MEMORY AND I/O DEVICES WITH 8085

The programs and data that are executed by the microprocessor have to be stored in ROM/EPROM and RAM, which are basically semiconductor memory chips. The programs and data that are stored in ROM/EPROM are not erased even when power supply to the chip is removed. Hence, they are called non-volatile memory. They can be used to store permanent programs. In a RAM, stored programs and data are erased when the power supply to the chip is removed. Hence, RAM is called volatile memory. RAM can be used to store programs and data that include, programs written during software development for a microprocessor based system, program written when one is learning assembly language programming and data enter while testing these programs. In the memory-mapped I/O scheme, each I/O device is assumed to be a memory location. Input and output devices, which are interfaced with 8085, are essential in any microprocessor based system. They can be interfaced using two schemes: I/O mapped I/O and memory-mapped I/O. In the I/O mapped I/O scheme, the I/O devices are treated differently from memory.

2. INTERFACING MEMORY CHIPS WITH 8085

8085 has 16 address lines (A0 - A15), hence a maximum of 64 KB (= 216 bytes) of memory locations can be interfaced with it. The memory address space of the 8085 takes values from 0000H to FFFFH.

The 8085 initiates set of signals such as IO/$\overline{M}$, $\overline{RD}$ and $\overline{WR}$ when it wants to read from and write into memory. Similarly, each memory chip has signals such as CE or CS (chip enable
or chip select), $\overline{OE}$ or $\overline{RD}$ (output enable or read) and $\overline{WE}$ or $\overline{WR}$ (write enable or write) associated with it.

Generation of Control Signals for Memory:

When the 8085 wants to read from and write into memory, it activates IO/$\overline{M}$, $\overline{RD}$ and $\overline{WR}$ signals as shown in Table 1.

Table 1 Status of IO/$\overline{M}$, $\overline{RD}$ and $\overline{WR}$ signals during memory read and write operations

IO/$\overline{M}$	$\overline{RD}$	$\overline{WR}$	Operation
0	0	1	8085 reads data from memory
0	1	0	8085 writes data into memory

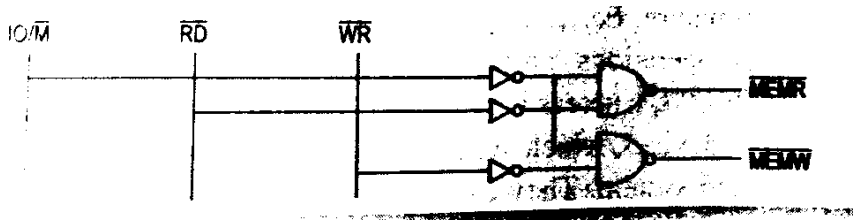


Fig. 3.1 Circuit used to generate $\overline{MEMR}$ and $\overline{MEMW}$ signals

Using IO/$\overline{M}$, $\overline{RD}$ and $\overline{WR}$ signals, two control signals MEMR (memory read) and MEMW (memory write) are generated. Fig. 3.1 shows the circuit used to generate these signals.

When is IO/$\overline{M}$ high, both memory control signals are deactivated irrespective of the status

of $\overline{RD}$ and $\overline{WR}$ signals.

Ex: Interface an IC 2764 with 8085 using NAND gate address decoder such that the address range allocated to the chip is 0000H – 1FFFH.

Specification of IC 2764:

- 8 KB (8 x 2^{10} byte) EPROM chip
- 13 address lines (2^{13} bytes = 8 KB) Interfacing:

13 address lines of IC are connected to the corresponding address lines of

8085.

- Remaining address lines of 8085 are connected to address decoder formed using logic gates, the output of which is connected to the CE pin of IC.
- Address range allocated to the chip is shown in Table 2.
- Chip is enabled whenever the 8085 places an address allocated to EPROM chip in the address bus. This is shown in Fig. 3.2

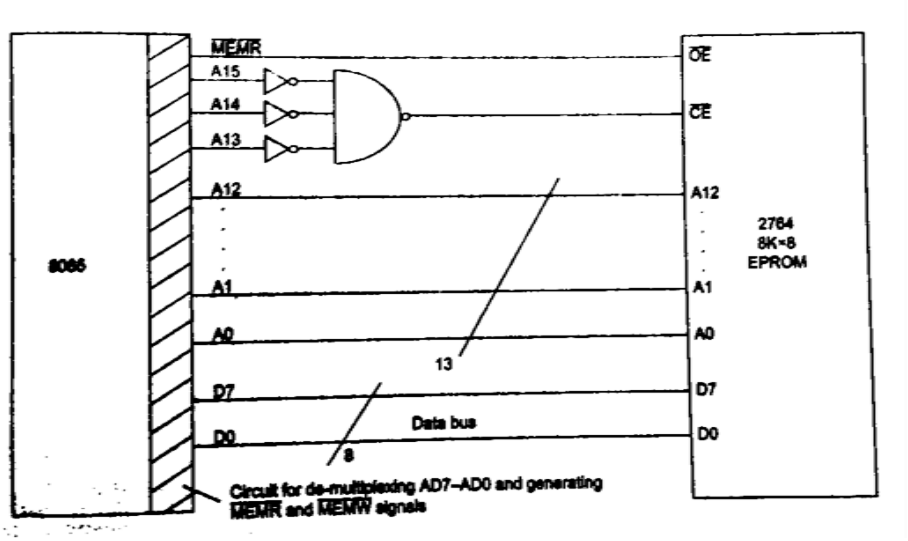


Fig. 3.2 Interfacing IC 2764 with the 8085

Table 2 Address allocated to IC 2764

A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	Address
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000H
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0001H
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	1FFEH
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1FFFH

Ex: Interface a 6264 IC (8K x 8 RAM) with the 8085 using NAND gate decoder such that the starting address assigned to the chip is 4000H.

Specification of IC 6264:

- 8K x 8 RAM
- 8 KB = 2^{13} bytes
- 13 address lines

The ending address of the chip is 5FFFH (since 4000H + 1FFFH = 5FFFH). When the address 4000H to 5FFFH are written in binary form, the values in the lines A15, A14, A13 are 0, 1 and 0 respectively. The NAND gate is designed such that when the lines A15 and A13 carry 0 and A14 carries 1, the output of the NAND gate is 0. The NAND gate output is in turn connected to the CE1 pin of the RAM chip. A NAND output of 0 selects the RAM chip for read or write operation, since CE2 is already 1 because of its connection to +5V. Fig. 18 shows the interfacing of IC 6264 with the 8085.

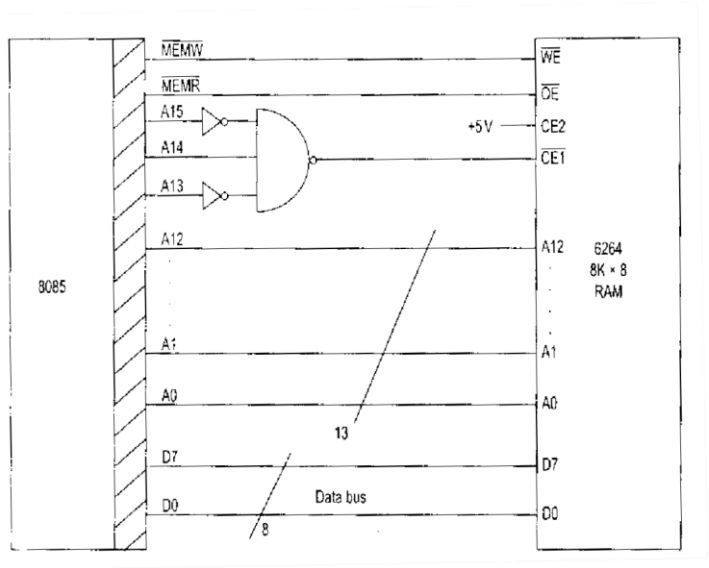


Fig. 3.3 Interfacing 6264 IC with the 8085

Ex: Interface two 6116 ICs with the 8085 using 74LS138 decoder such that the starting addresses assigned to them are 8000H and 9000H, respectively. Specification of IC 6116:

- 2 K x 8 RAM
- 2 KB = 2^{11} bytes
- 11 address lines

6116 has 11 address lines and since 2 KB, therefore ending addresses of 6116 chip 1 is and chip 2 are 87FFH and 97FFH, respectively. Table 3 shows the address range of the two chips.

Table 3 Address range for IC 6116

A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	Address
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8000H
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
1	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	87FFH (RAM chip 1)
1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	9000H
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
1	0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	97FFH (RAM chip 2)

Interfacing:

- Fig. 3.3 shows the interfacing.
- A0 – A10 lines of 8085 are connected to 11 address lines of the RAM chips.
- Three address lines of 8085 having specific value for a particular RAM are connected to the three select inputs (C, B and A) of 74LS138 decoder.
- Table 3 shows that A13=A12=A11=0 for the address assigned to RAM 1 and A13=0, A12=1 and A11=0 for the address assigned to RAM 2.
- Remaining lines of 8085 which are constant for the address range assigned to the two RAM are connected to the enable inputs of decoder.
- When 8085 places any address between 8000H and 87FFH in the address bus, the select inputs C, B and A of the decoder are all 0. The Y0 output of the decoder is also 0, selecting RAM 1.
- When 8085 places any address between 9000H and 97FFH in the address bus, the select inputs C, B and A of the decoder are 0, 1 and 0. The Y2 output of the decoder is also 0, selecting RAM 2.

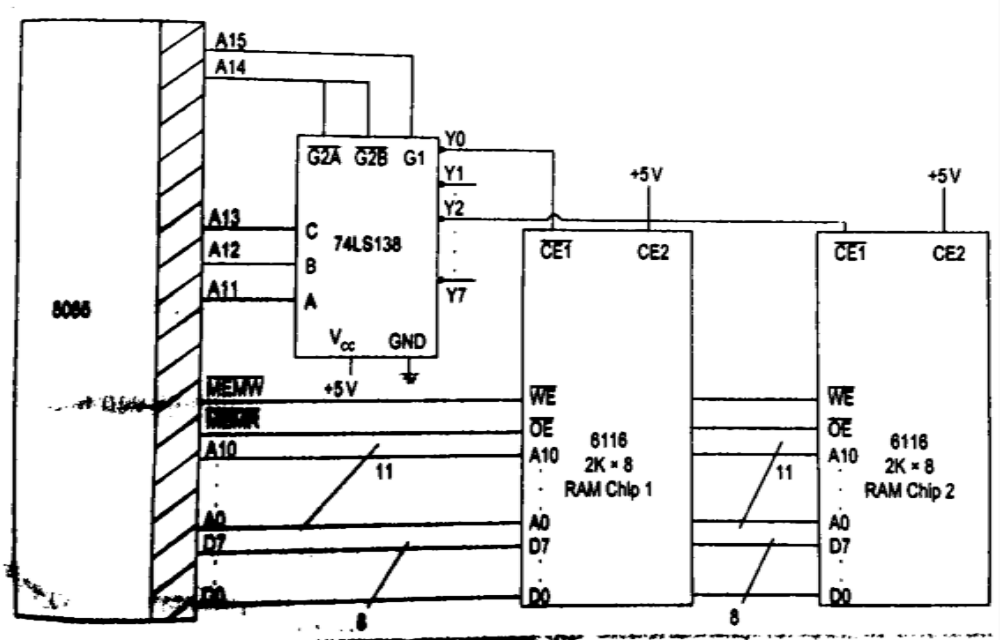


Fig. 3.4 Interfacing two 6116 RAM chips using 74LS138 decoder

2. I/O MAPPED I/O INTERFACING

In this method, the I/O devices are treated differently from memory chips.

The control signals I/O read ($\overline{IOR}$) and I/O write ($\overline{IOW}$), which are derived from the IO/M , RD and WR signals of the 8085, are used to activate input and output devices, respectively. Generation of these control signals is shown in Fig. 3.5. Table 4 shows the status of $\overline{IO/M}$, RD and WR signals during I/O read and I/O write operation.

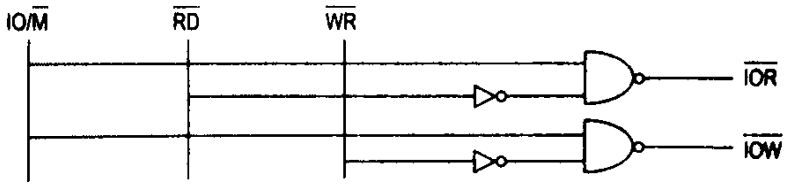


Fig. 3.5 Generation of $\overline{IOR}$ and $\overline{IOW}$ signals

IN instruction is used to access input device and OUT instruction is used to access output device. Each I/O device is identified by a unique 8-bit address assigned to it.

Since the control signals used to access input and output devices are different, and all I/O device use 8-bit address, a maximum of 256 (2^8) input devices and 256 output devices can be interfaced with 8085.

Ex: Interface an 8-bit DIP switch with the 8085 such that the address assigned to the DIP switch if F0H.

IN instruction is used to get data from DIP switch and store it in accumulator. Steps involved in the execution of this instruction are:

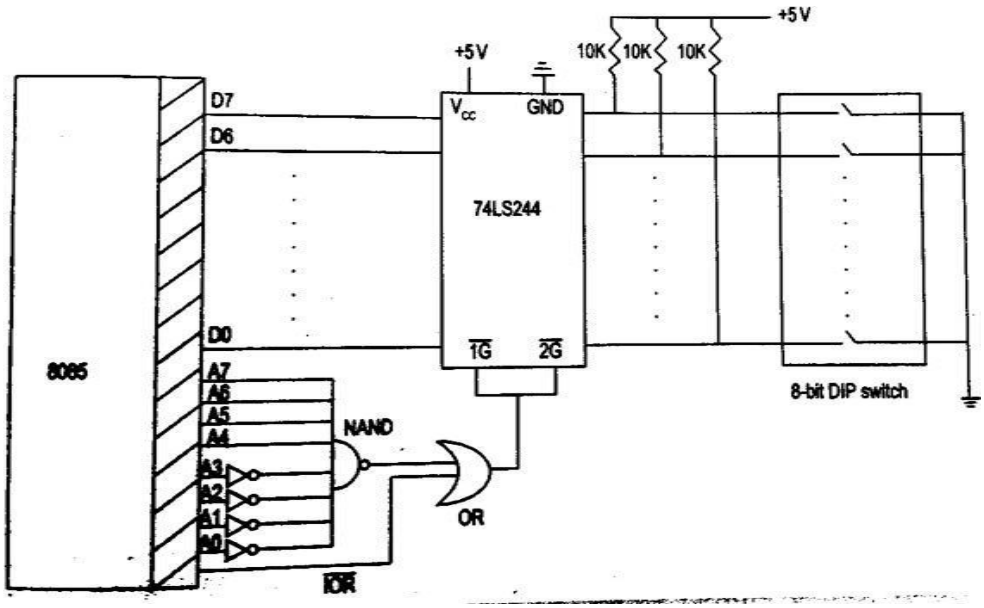
i. Address F0H is placed in the lines A0 – A7 and a copy of it in lines

A8 – A15.

ii. The IOR signal is activated (IOR = 0), which makes the selected input device to place its data in the data bus.

iii. The data in the data bus is read and store in the accumulator. Fig. 3.6 shows the interfacing of DIP switch.

A0 – A7 lines are connected to a NAND gate decoder such that the output of NAND gate is 0. The output of NAND gate is ORed with the IOR signal and the output of OR gate is connected to 1G and 2G of the 74LS244. When 74LS244 is enabled, data from the DIP switch is placed on the data bus of the 8085. The 8085 read data and store in the accumulator. Thus data from DIP switch is transferred to the



accumulator.

Fig.3.6 Interfacing of 8-bit DIP switch with 8085

4. MEMORY MAPPED I/O INTERFACING

In memory-mapped I/O, each input or output device is treated as if it is a memory location. The $\overline{MEMR}$ and $\overline{MEMW}$ control signals are used to activate the devices. Each input or output device is identified by unique 16-bit address, similar to 16-bit address assigned to memory location. All memory related instruction like LDA 2000H, LDAX B, MOV A, M can be used.

Since the I/O devices use some of the memory address space of 8085, the maximum memory capacity is lesser than 64 KB in this method.

Ex: Interface an 8-bit DIP switch with the 8085 using logic gates such that the address assigned to it is F0F0H.

Since a 16-bit address has to be assigned to a DIP switch, the memory-mapped I/O technique must be used. Using LDA F0F0H instruction, the data from the 8-bit DIP switch can be transferred to the accumulator. The steps involved are:

i. The address F0F0H is placed in the address bus A0 – A15.

ii. The $\overline{MEMR}$ signal is made low for some time.

iii. The data in the data bus is read and stored in the accumulator. Fig. 3.7 shows the interfacing diagram.

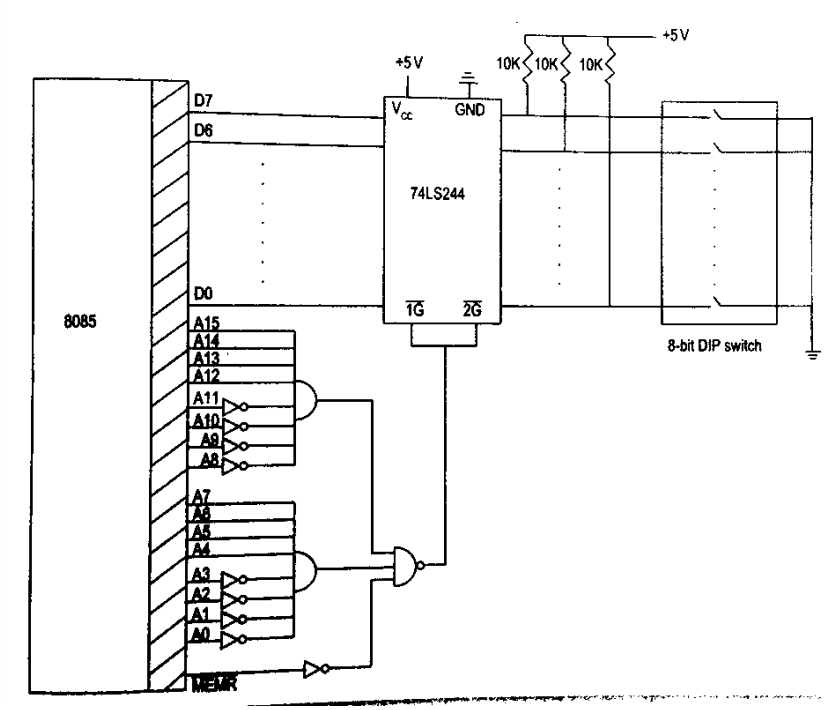


Fig. 3.7 Interfacing 8-bit DIP switch with 8085

When 8085 executes the instruction LDA F0F0H, it places the address F0F0H in the address lines A0 – A15 as:

The address lines are connected to AND gates. The output of these gates along with MEMR signal are connected to a NAND gate, so that when the address F0F0H is placed in the address bus and MEMR = 0 its output becomes 0, thereby enabling the buffer 74LS244. The data from the DIP switch is placed in the 8085 data bus. The 8085 reads the data from the data bus and stores it in the accumulator.

When 8085 executes the instruction LDA F0F0H, it places the address F0F0H in the address lines A0 – A15 as:

A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	
1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	=F0F0H

The address lines are connected to AND gates. The output of these gates along with MEMR

8255 - PROGRAMMABLE PERIPHERAL INTERFACE (PPI)

The **Intel 8255** (or **i8255**) Programmable Peripheral Interface (**PPI**) chip is a peripheral chip, is used to give the CPU access to programmable parallel I/O. It can be programmable 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.

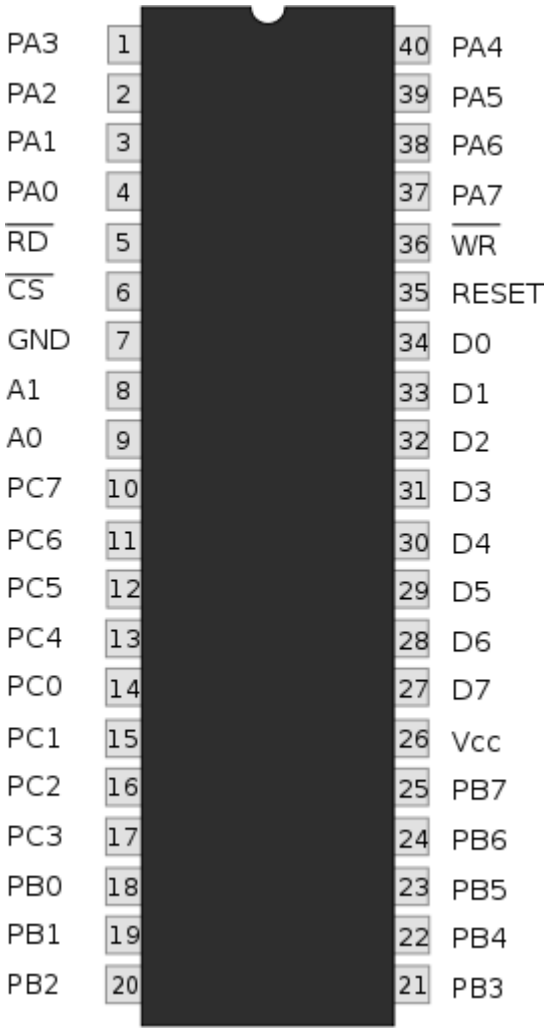


Fig 3.8: Pin diagram of 8255

FUNCTIONAL BLOCK OF 8255 – PROGRAMMABLE PERIPHERAL INTERFACE(PPI)

The 8255A has 24 I/O pins that can be grouped primarily in two 8-bit parallel ports: A and B with the remaining eight bits as port C as in Figure 3.8. The eight bits of port C can be used as individual bits or be grouped in to 4-bit ports: C_{Upper} (C_u) and C_{Lower} (C_L) as in Figure 3.9. The function of these ports is defined by writing a control word in the control register as shown in Figure 3.10.

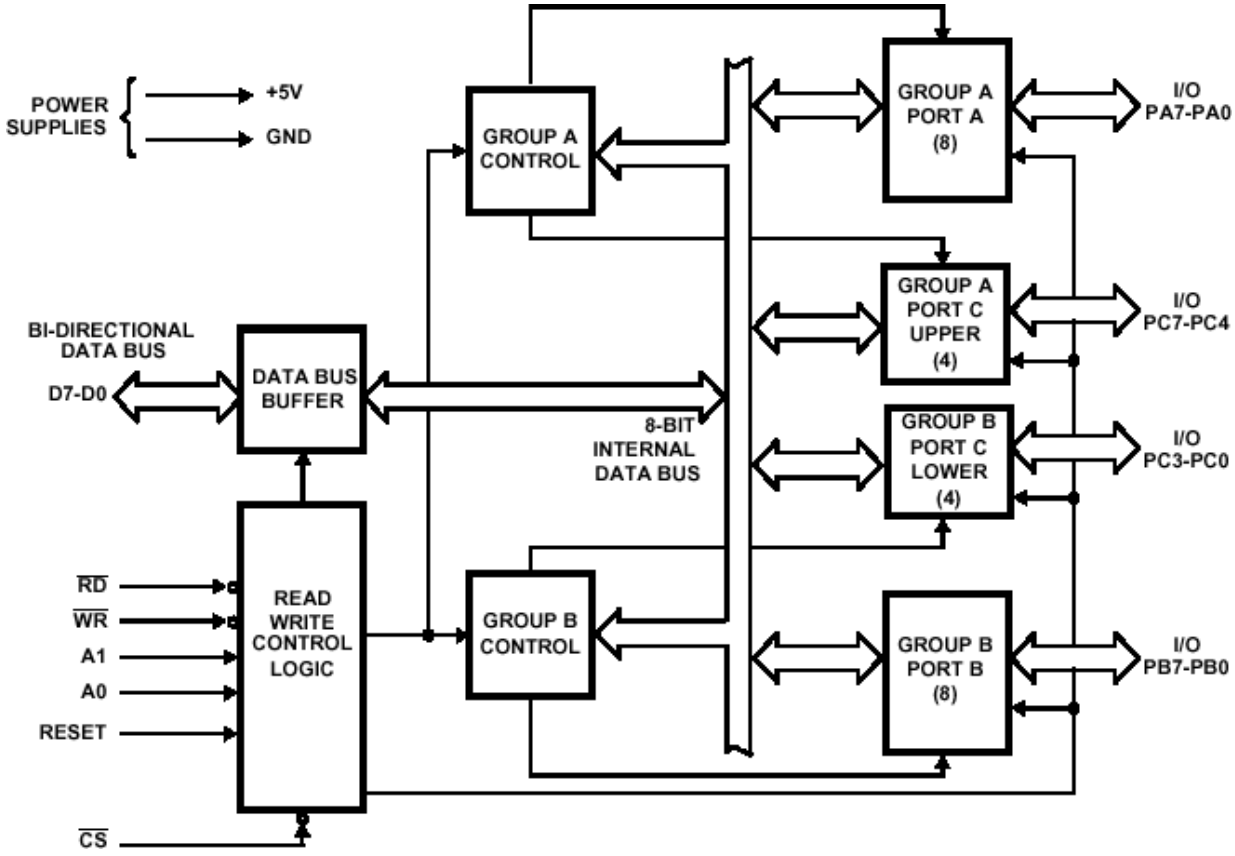


Fig 3.9. Block diagram of 8255

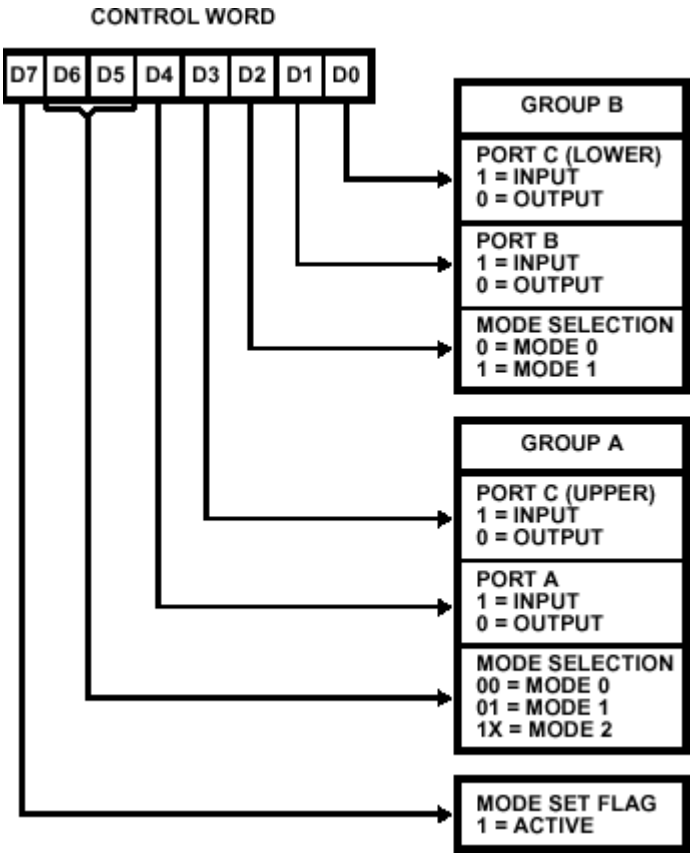


Fig 3.10 Control word Register format

DATA BUS BUFFER

This three-state bi-directional 8-bit buffer is used to interface the 8255 to the system data bus. Data is transmitted or received by the buffer upon execution of input or output instructions bythe CPU. Control words and status information are also transferred through the data bus buffer.

READ/WRITE AND CONTROL LOGIC

The function of this block is to manage all of the internal and external transfers of both Data and Control or Status words. It accepts inputs from the CPU Address and Control busses and in turn, issues commands to both of the Control Groups.

(CS) Chip Select. A "low" on this input pin enables the communication between the 8255 andthe CPU.

(RD) Read. A "low" on this input pin enables 8255 to send the data or status information to the

CPU on the data bus. In essence, it allows the CPU to "read from" the 8255.

(WR) Write. A "low" on this input pin enables the CPU to write data or control words into the8255.

(A0 and A1) Port Select 0 and Port Select 1. These input signals, in conjunction with the RD and WR inputs, control the selection of one of the three ports or the control word register. Theyare normally connected to the least significant bits of the address bus (A0 and A1).

(RESET) Reset. A "high" on this input initializes the control register to 9Bh and all ports (A,B, C) are set to the input mode.

A1	A0	SELECTION
0	0	PORT A
0	1	PORT B
1	0	PORT C
1	1	CONTROL

GROUP A AND GROUP B CONTROLS

The functional configuration of each port is programmed by the systems software. In essence, the CPU "outputs" a control word to the 8255. The control word contains information such as "mode", "bit set", "bit reset", etc., that initializes the functional configuration of the 8255. Each of the Control blocks (Group A and Group B) accepts "commands" from the Read/Write Control logic, receives "control words" from the internal data bus and issues the proper commands to its associated ports.

PORTS A, B, AND C

The 8255 contains three 8-bit ports (A, B, and C). All can be configured to a wide variety of functional characteristics by the system software but each has its own special features or "personality" to further enhance the power and flexibility of the 8255.

Port A One 8-bit data output latch/buffer and one 8-bit data input latch. Both "pull-up" and "pull-down" bus-hold devices are present on Port A.

Port B One 8-bit data input/output latch/buffer and one 8-bit data input buffer.
Port C One 8-bit data output latch/buffer and one 8-bit data input buffer (no latch for input). This port can be divided into two 4-bit ports under the mode control. Each 4-bit port contains a 4-bit latch and it can be used for the control signal output and status signal inputs in conjunction with ports A and B.

OPERATIONAL MODES OF 8255

There are two basic operational modes of 8255:

- Bit set/reset Mode (BSR Mode).
- Input/Output Mode (I/O Mode).

The two modes are selected on the basis of the value present at the D_7 bit of the Control WordRegister. When $D_7 = 1$, 8255 operates in I/O mode and when $D_7 = 0$, it operates in the BSR mode.

1. BIT SET/RESET (BSR) MODE

The Bit Set/Reset (BSR) mode is applicable to port C only. Each line of port C (PC_0 - PC_7) canbe set/reset by suitably loading the control word register as shown in Figure 3.11. BSR mode and I/O mode are independent and selection of BSR mode does not affect the operation of other ports in I/O mode.

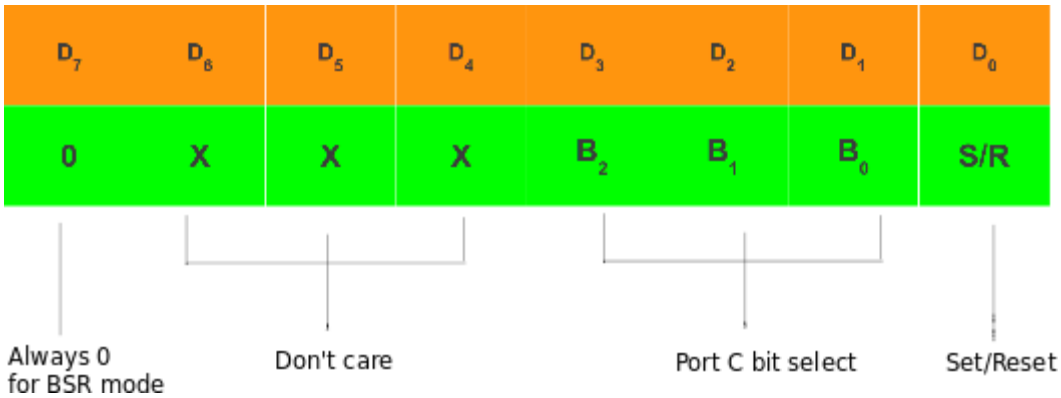


Fig 3.11: 8255 Control register format for BSR mode

- D_7 bit is always 0 for BSR mode.
- Bits D_6, D_5 and D_4 are don't care bits.
- Bits D_3, D_2 and D_1 are used to select the pin of Port C.
- Bit D_0 is used to set/reset the selected pin of Port C.
- Selection of port C pin is determined as follows:

B3	B2	B1	Bit/pin of port C selected
0	0	0	PC_0
0	0	1	PC_1
0	1	0	PC_2
0	1	1	PC_3
1	0	0	PC_4
1	0	1	PC_5
1	1	0	PC_6
1	1	1	PC_7

As an example, if it is needed that PC_5 be set, then in the control word,

1. Since it is BSR mode, **D_7 = '0'**.
2. Since D_4, D_5, D_6 are not used, assume them to be '**0**'.
3. PC_5 has to be selected, hence, **D_3 = '1', D_2 = '0', D_1 = '1'**.
4. PC_5 has to be set, hence, **D0 = '1'**.

Thus, as per the above values, 0B (Hex) will be loaded into the Control Word Register (CWR).

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	1	0	1	1

2. INPUT/OUTPUT MODE

This mode is selected when D_7 bit of the Control Word Register is 1. There are three I/O modes:

1. Mode 0 - Simple I/O
2. Mode 1 - Strobed I/O
3. Mode 2 - Strobed Bi-directional I/O

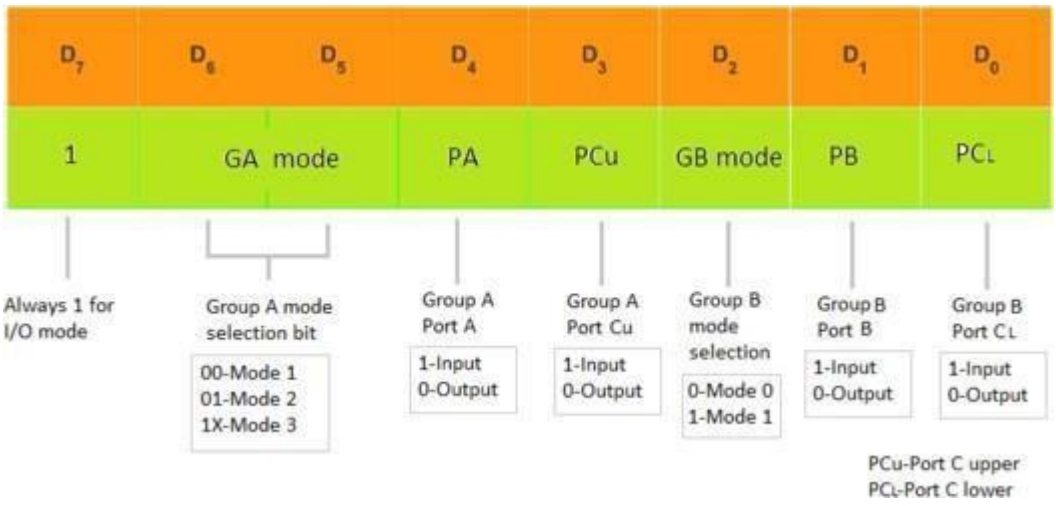


Figure 3.12: 8255 Control word for I/O mode

- **D_0, D_1, D_3, D_4** are assigned for lower port C, port B, upper port C and port A respectively. When these bits are **1**, the corresponding port acts as an input port. For e.g., if $D_0 = D_4 = 1$, then lower port C and port A act as input ports. If these bits are **0**, then the corresponding port acts as an output port. For e.g., if $D_1 = D_3 = 0$, then port B and upper port C act as output ports as shown in Figure 3.12
- **D_2** is used for mode selection of Group B (port B and lower port C). When $D_2 = 0$, mode 0 is selected and when $D_2 = 1$, mode 1 is selected.
- **D_5& D_6** are used for mode selection of Group A (port A and upper port C). The selection is done as follows:
- As it is I/O mode, **D_7** = 1.

For example, if port B and upper port C have to be initialized as input ports and lower port C and port A as output ports (all in mode 0):

1. Since it is an I/O mode, $D_7 = 1$.
2. Mode selection bits, D2, D5, D6 are all 0 for mode 0 operation.
3. Port B and upper port C should operate as Input ports, hence, $D_1 = D_3 = 1$.
4. Port A and lower port C should operate as Output ports, hence, $D_4 = D_0 = 0$.

Hence, for the desired operation, the control word register will have to be loaded with **"10001010" = 8A (hex)**.

- **Mode 0 - simple I/O**

In this mode, the ports can be used for simple I/O operations without handshaking signals. PortA, port B provide simple I/O operation. The two halves of port C can be either used together as an additional 8-bit port, or they can be used as individual 4-bit ports. Since the two halves of port C are independent, they may be used such that one- half is initialized as an input port while the other half is initialized as an output port.

The input/output features in mode 0 are as follows:

1. Output ports are latched.
2. Input ports are buffered, not latched.
3. Ports do not have handshake or interrupt capability.
4. With 4 ports, 16 different combinations of I/O are possible.

- **Mode 0 – input mode**

- In the input mode, the 8255 gets data from the external peripheral ports and the CPUreads the received data via its data bus.
- ¬ The CPU first selects the 8255 chip by making CS low. Then it selects the desired port using A_0 and A_1 lines.
- ¬ The CPU then issues an RD signal to read the data from the externalperipheral device via the system data bus.

- Mode 0 - output mode

  In the output mode, the CPU sends data to 8255 via system data bus and then the external peripheral ports receive this data via 8255 port.

  - CPU first selects the 8255 chip by making $\overline{CS}$ low. It then selects the desired port using A_0 and A_1 lines.
    CPU then issues a WR signal to write data to the selected port via the system data bus. Thisdata is then received by the external peripheral device connected to the selected port.

- Mode 1

  When we wish to use port A or port B for handshake (strobed) input or output operation, we initialize that port in mode 1 (port A and port B can be initialized to operate in different modes,i.e., for e.g., port A can operate in mode 0 and port B in mode 1). Some of the pins of port C function as handshake lines.

  For port B in this mode (irrespective of whether is acting as an input port or output port), PC0,PC1 and PC2 pins function as handshake lines.

  If port A is initialized as mode 1 input port, then, PC3, PC4 and PC5 function as handshake signals. Pins PC6 and PC7 are available for use as input/output lines.

The mode 1 which supports handshaking has following

features:1.
Two ports i.e. port A and B can be used as 8-bit i/o ports.
2.
Each port uses three lines of port c as handshake signal and remaining two signals canbe used as i/o ports.
3.
Interrupt logic is supported.
4.
Input and Output data are latched.

INPUT HANDSHAKING SIGNALS

1. IBF (Input Buffer Full) - It is an output indicating that the input latch contains information.

2. STB (Strobed Input) - The strobe input loads data into the port latch, which holds the information until it is input to the microprocessor via the IN instruction.
3. INTR (Interrupt request) - It is an output that requests an interrupt. The INTR pin becomes a logic 1 when the STB input returns to a logic 1, and is cleared when the data are input from the port by the microprocessor.
4. INTE (Interrupt enable) - It is neither an input nor an output; it is an internal bit programmed via the port PC4 (port A) or PC2(port B) bit position.

OUTPUT HANDSHAKING SIGNALS

- OBF (Output Buffer Full) - It is an output that goes low whenever data are output(OUT) to the port A or port B latch. This signal is set to a logic 1 whenever the ACK pulse returns from the external device.
- ACK (Acknowledge)-It causes the OBF pin to return to a logic 1 level. The ACK signal is a response from an external device, indicating that it has received the data from the 82C55 port.
- INTR (Interrupt request) - It is a signal that often interrupts the microprocessor when the external device receives the data via the signal. this pin is qualified by the internal INTE(interrupt enable) bit.
- INTE (Interrupt enable) - It is neither an input nor an output; it is an internal bit programmed to enable or disable the INTR pin. The INTE A bit is programmed using the PC6 bit and INTE B is programmed using the PC2 bit.

➢ **Mode 2**

Only group A can be initialized in this mode. Port A can be used for bidirectional handshake data transfer. This means that data can be input or output on the same eight lines (PA0 - PA7). Pins PC3 - PC7 are used as handshake lines for port A. The remaining pins of port C (PC0 - PC2) can be used as input/output lines if group B is initialized in mode 0 or as handshaking for port B if group B is initialized in mode 1. In this mode, the 8255 may be used to extend the system bus to a slave microprocessor or to transfer data bytes to and from a floppy disk controller. Acknowledgement and handshaking signals are provided to maintain proper data flow and synchronization between the data transmitter and receiver.

INTERFACING 8255 WITH 8085 PROCESSOR

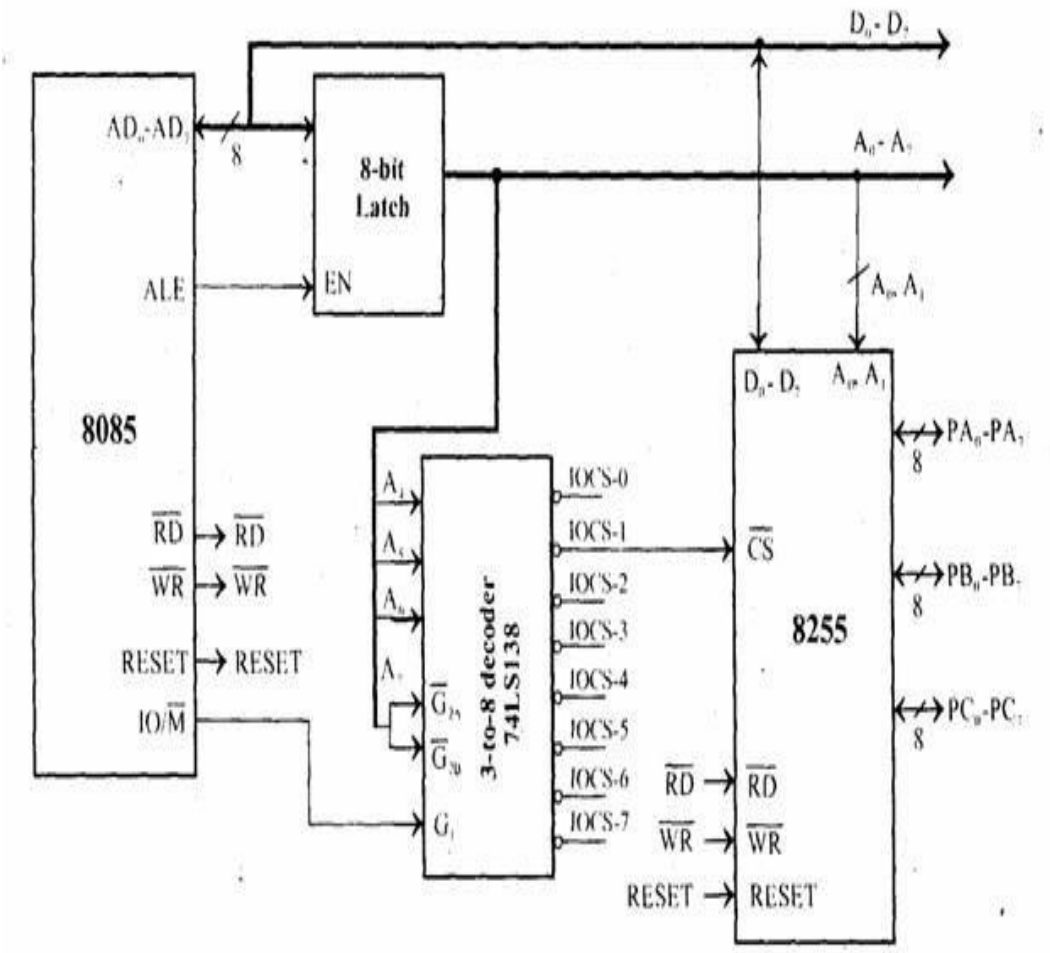


Fig 3.13. Interfacing 8255 with 8085 processor

- The 8255 can be either memory mapped or I/O mapped in the system. In the schematicshown in above is I/O mapped in the system.

- Using a 3-to-8 decoder generates the chip select signals for I/O mapped devices.

- The address lines A4, A5 and A6 are decoded to generate eight chip select signals (IOCS-0 to IOCS-7) and in this, the chip select IOCS- 1 is used to select 8255 as shown in Figure 3.13.

- The address line A7 and the control signal IO/M (low) are used as enable for the decoder.

- The address line A0 of 8085 is connected to A0 of 8255 and A1 of 8085 is connectedto A1 of 8255 to provide the internal addresses.

- The data lines D0-D7 are connected to D0-D7 of the processor to achieve parallel datatransfer.

- The I/O addresses allotted to the internal devices of 8255 are listed in table.

Internal Device	Binary Address								Hexa Address
	Decoder input and enable				Input to address pins of 8255				
	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	
Port-A	0	0	0	1	x	x	0	0	10
Port-B	0	0	0	1	x	x	0	1	11
Port-C	0	0	0	1	x	x	1	0	12
Control Register	0	0	0	1	x	x	1	1	13

Note : *Don't care "x" is considered as zero.*

8253(8254) PROGRAMMABLE INTERVAL TIMER:

The 8254 programmable Interval timer consists of three independent 16-bit programmable counters (timers). Each counter is capable of counting in binary or binary coded decimal. The maximum allowable frequency to any counter is 10MHz. This device is useful whenever the microprocessor must control real-time events. The timer in a personal computer is an 8253. To operate a counter a 16-bit count is loaded in its register and on command, it begins to decrementthe count until it reaches 0. At the end of the count it generates a pulse, which interrupts the processor. The count can count either in binary or BCD Each counter in the block diagram has3 logical lines connected to it. Two of these lines, clock and gate, are inputs. The third, labeledOUT is an output.

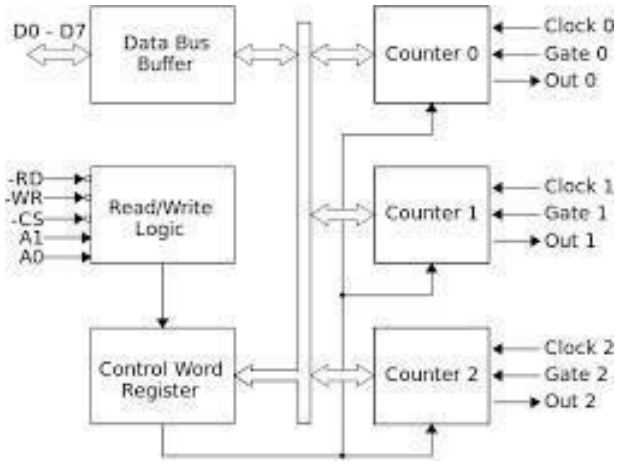


Fig 3.14 Block Diagram of 8253 programmable interval timer

Data bus buffer- It is a communication path between the timer and the microprocessor. The buffer is 8-bit and bidirectional. It is connected to the data bus of the microprocessor. Read
/write logic controls the reading and the writing of the counter registers. Control word register, specifies the counter to be used and either a Read or a write operation. Data is transmitted or received by the buffer upon execution of INPUT instruction from CPU as shown in figure 3.14.The data bus buffer has three basic functions,

- Programming the modes of 8253.
- Loading the count value in times

READING THE COUNT VALUE FROM TIMERS.

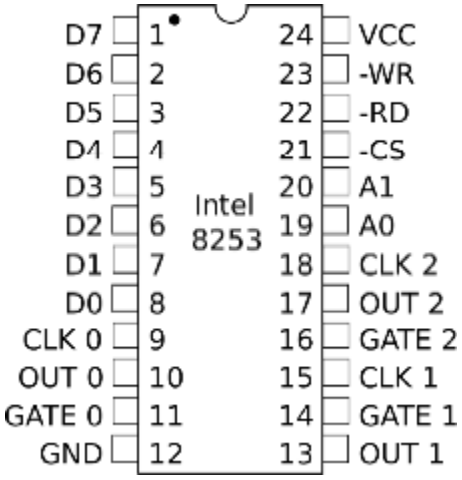


Fig 3.15 Pin Diagram of 8253

The data bus buffer is connected to microprocessor using D7 – D0 pins which are also bidirectional. The data transfer is through these pins. These pins will be in

high-impedance (orthis state) condition until the 8253 is selected by a LOW or *CS*

and either the read operation requested by a LOW $\overline{RD}$ on the input or a write operation *WR* requested by the input going LOW.

READ/ WRITE LOGIC:

It accepts inputs for the system control bus and in turn generation the control signals for overall device operation. It is enabled or disabled by $\overline{CS}$ so that no operation can occur to change the function unless the device has been selected as the system logic.

CS :

The chip select input is used to enable the communicate between 8253 and the microprocessorby means of data bus. A low an CS *enables* the data bus buffers, while

a high disables the buffer. The *CS* input does not have any effect on the operation of three times once they have been initialized. The normal configuration of a system employs an decodelogic which actives *CS* line, whenever a specific set of addresses that correspond to 8253 appearon the address bus.

RD & WR :

The read ($\overline{RD}$) and write $\overline{WR}$ pins central the direction of data transfer on the 8-bit

bus. Whenthe input *RD* pin is low. Then CPU is inputting data from 8253 in the form of counter value.

When *WR* pins is low, then CPU is sending data to 8253 in the form of mode information or loading counters. The *RD* & *WR* should not both be low simultaneously. When *RD* & *WR* pinsare HIGH, the data bus buffer is disabled.

A0 & A1:

These two input lines allow the microprocessor to specify which one of the internal register in the 8253 is going to be used for the data transfer. **Fig** shows how these two lines are used to select either the control word register or one of the 16-bit counters.

CONTROL WORD REGISTER:

$\overline{CS}$	$\overline{RD}$	$\overline{WR}$	A_1	A_0	operation
0	1	0	0	0	Load counter '0'
0	1	0	0	1	Load counter '1'
0	1	0	1	0	Load counter '2'
0	1	0	1	1	Write mode word
0	0	1	0	0	Read TM_0
0	0	1	0	1	Read TM_1
0	0	1	1	0	Read TM_2
0	0	1	1	1	No- operation 3- state
1	X	X	X	X	Disable -- state
0	1	1	X	X	No- operation 3- state

It is selected when A0 and A1 . It the accepts information from the data bus buffer and stores it in a register. The information stored in then register controls the operation mode of each counter, selection of binary or BCD counting and the loading of each counting and the loadingof each count register. This register can be written into, no read operation of this content is available.

COUNTERS:

Each of the times has three pins associated with it. These are CLK (CLK) the gate (GATE) andthe output (OUT).

CLK:

This clock input pin provides 16-bit times with the signal to causes the times to decrement max^{m} clock input is 2.6MHz. Note that the counters operate at the negative edge (H1 to L0) ofthis clock input. If the signal on this pin is generated by a fixed oscillator then the user has implemented a standard timer. If the input signal is a string of randomly occurring pulses, thenit is called implementation of a counter.

GATE:

The gate input pin is used to initiate or enable counting. The exact effect of the gate signal depends on which of the six modes of operation is chosen.

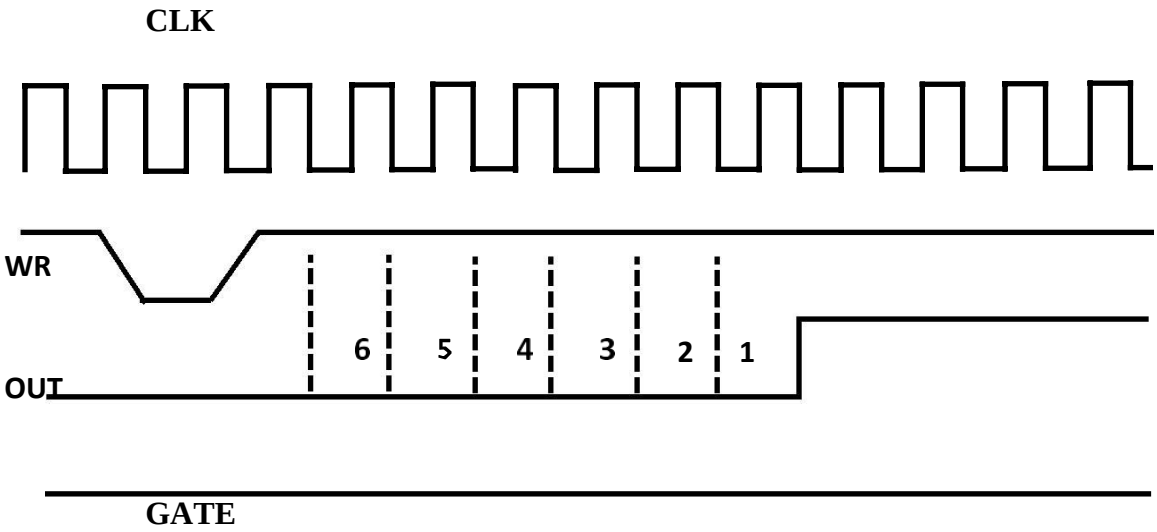
OUTPUT:

The output pin provides an output from the timer. It actual use depends on the mode of operation of the timer. The counter can be read ―in the fly‖ without inhibiting gate pulse or clock input.

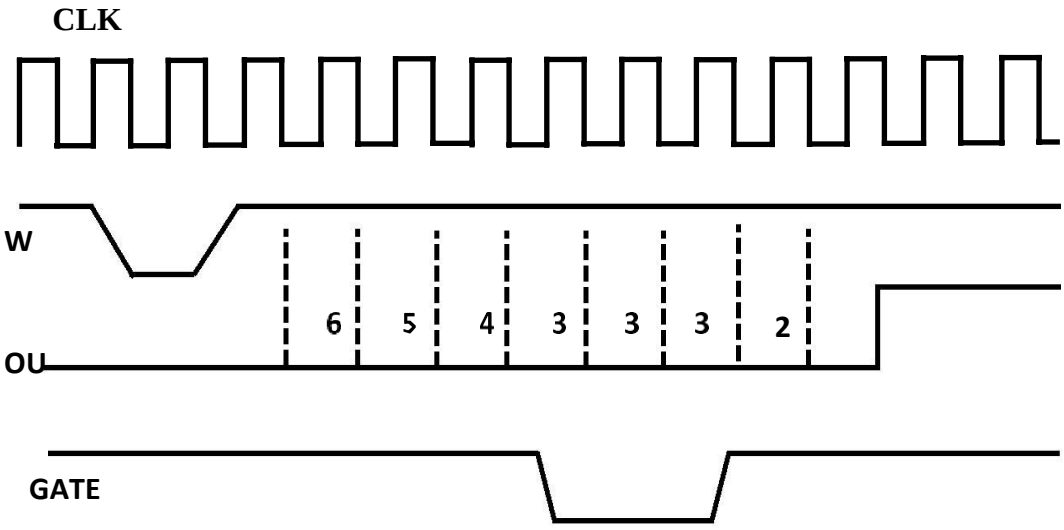
CONTROL REGISTER MODES OF OPERATION

Mode 0 Interrupt on terminal countMode 1 Programmable one shot Mode 2 Rate Generator
Mode 3 Square wave rate GeneratorMode 4 Software triggered strobe Mode 5 Hardware triggered strobe

Mode 0: The output goes high after the terminal count is reached. The counter stops if the Gateis low.. The timer count register is loaded with a count (say 6) when the WR line is made lowby the processor. The counter unit starts counting down with each clock pulse. The output goeshigh when the register value reaches zero. In the mean time if the GATE is made low the countis suspended at the value(3) till the GATE is enabled again .



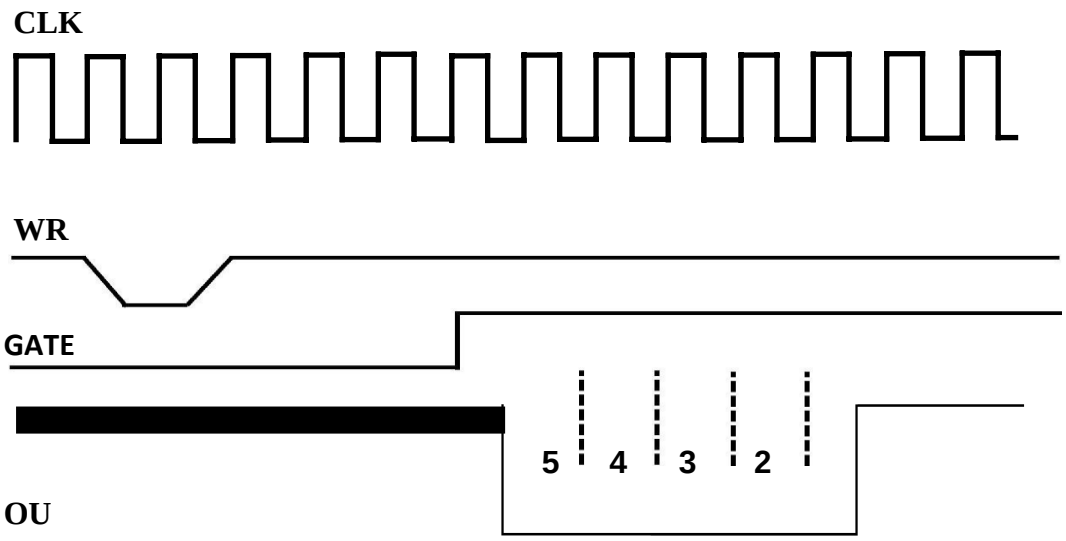
Mode 0 count when Gate is high (enabled)



Mode 0 count when Gate is low temporarily (disabled) Mode 1 Programmable mono-shot

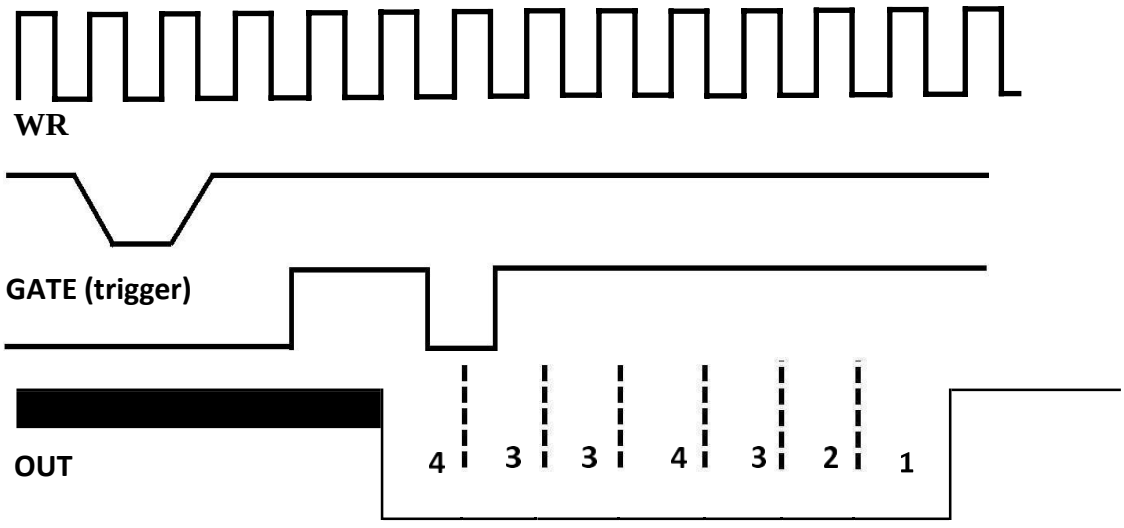
The output goes low with the Gate pulse for a predetermined period depending on the counter. The counter is disabled if the GATE pulse goes momentarily low.The counter register is loaded with a count value as in the previous case (say 5). The output responds to the GATE input andgoes low for period that equals the count down period of the register (5 clock pulses in this period). By changing the value of this count the duration of the output pulse can be changed. If the GATE becomes low before the count down is completed then the counter will be

suspended at that state as long as GATE is low. Thus it works as a mono-shot.



Mode 1 The Gate goes high. The output goes low for the period depending

on the countCLK

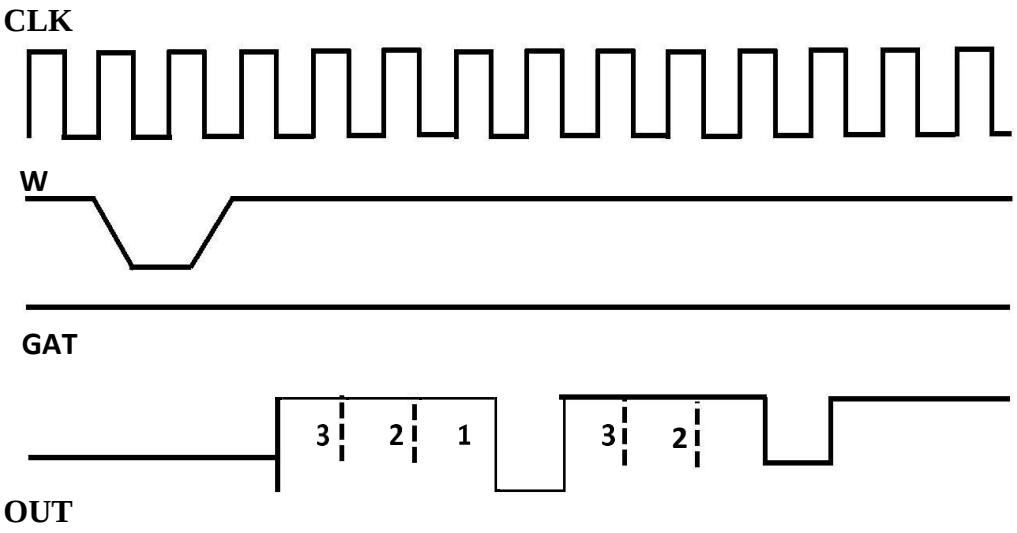


Mode 1 The Gate pulse is disabled momentarily causing the

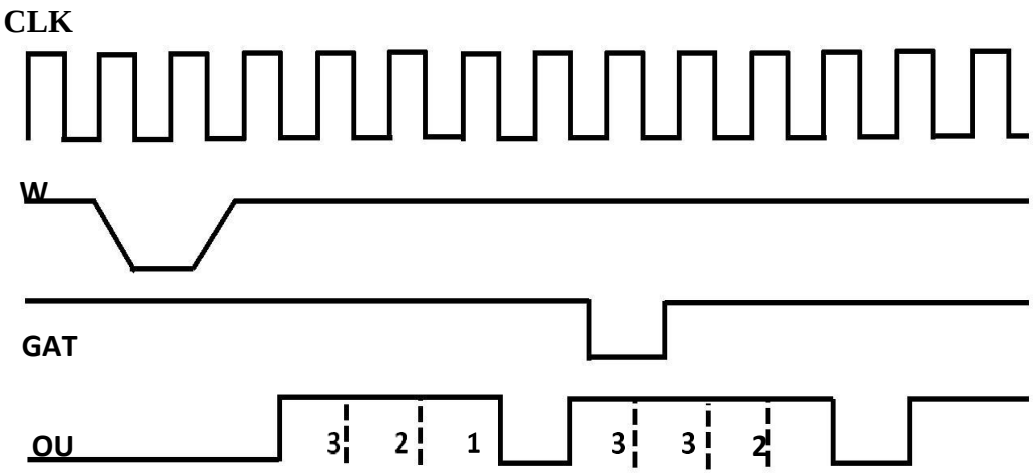
counter to stop.Mode 2 Programmable Rate Generator

In this mode it operates as a rate generator. The output goes high for a period that equals the

time of count down of the count register (3 in this case). The output goes low exactly for one clock period before it becomes high again. This is a periodic operation.

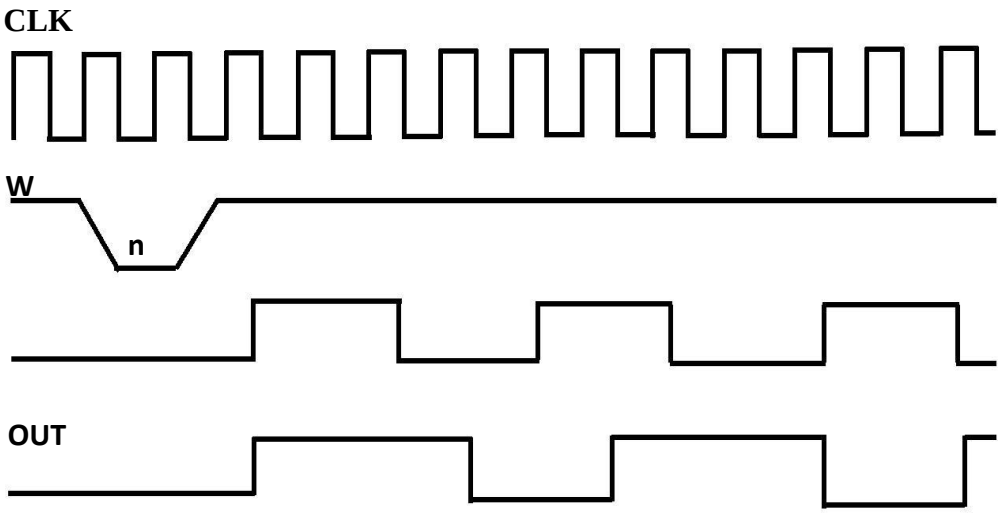


Mode 2 Operation when the GATE is kept high



Mode 2 operation when the GATE is disabled momentarily. Mode 3 Programmable Square Wave Rate Generator

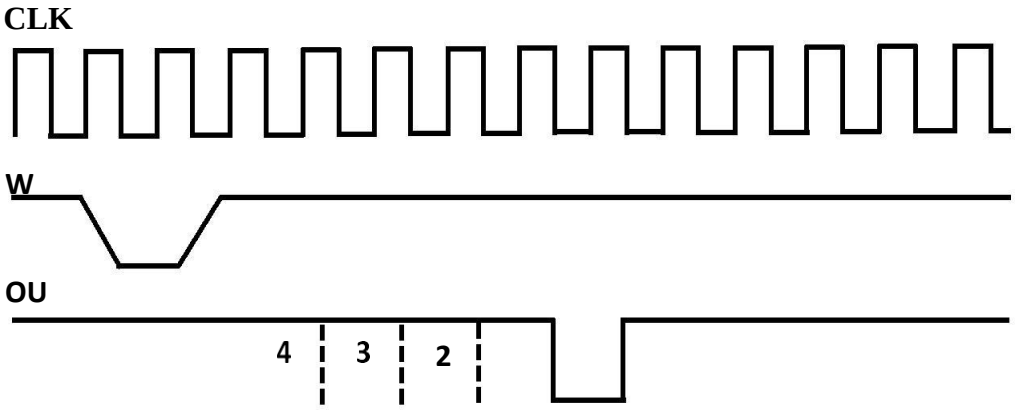
It is similar to Mode 2 but the output high and low period is symmetrical. The outputgoes high after the count is loaded and it remains high for period which equals the count down period ofthe counter register. The output subsequently goes low for an equal period and hence generates a symmetrical square wave unlike Mode 2. The GATE has no role here.



Mode3 Operation: Square Wave generatorMode 4 Software

Triggered Strobe

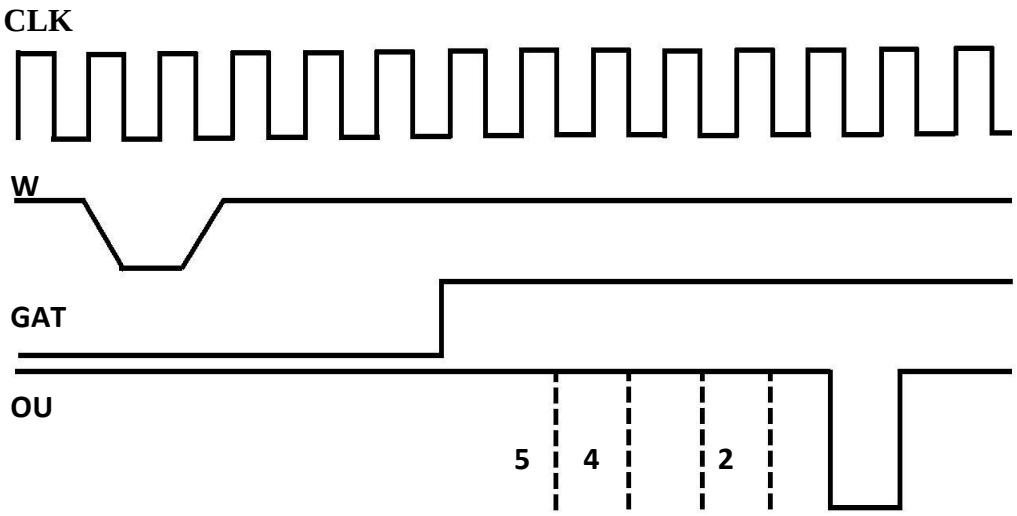
In this mode after the count is loaded by the processor the count down starts. The output goeslow for one clock period after the count down is complete. The count down can be suspendedby making the GATE low . This is also called a software triggered strobe as the count down isinitiated by a program.



Mode 4 Software Triggered Strobe when GATE is high Mode 5

Hardware Triggered Strobe

The count is loaded by the processor but the count down is initiated by the GATE pulse. Thetransition from low to high of the GATE pulse enables count down. The output goes low for one clock period after the count down is complete.



Mode 5 Hardware Triggered Strobe

PROGRAMMABLE INTERRUPT

CONTROLLER-8259FEATURES OF 8259

1. 8086, 8088 Compatible
2. MCS-80, MCS-85 Compatible
3. Eight-Level Priority Controller
4. Expandable to 64 Levels
5. Programmable Interrupt Modes
6. Individual Request Mask Capability
7. Single +5V Supply (No Clocks)
8. Available in 28-Pin DIP and 28-Lead PLCC Package
9. Available in EXPRESS
10. Standard Temperature Range
11. Extended Temperature Range

The Intel 8259A Programmable Interrupt Controller handles up to eight vectored priority interrupts for the CPU. It is cascadable for up to 64 vectored priority interrupts without additional circuitry. It is packaged in a 28-pin DIP, uses NMOS technology and requires a

single a5V supply. Circuitry is static, requiring no clock input. The 8259A is designed to minimize the software and real time overhead in handling multi-level priority interrupts. It has several modes, permitting optimization for a variety of system requirements. The 8259A is fully upward compatible with the Intel 8259. Software originally written for the 8259 will operate the 8259A in all 8259 equivalent modes (MCS-80/85, Non-Buffered, Edge Triggered). Pin Diagram of 8259 is shown in figure 17.

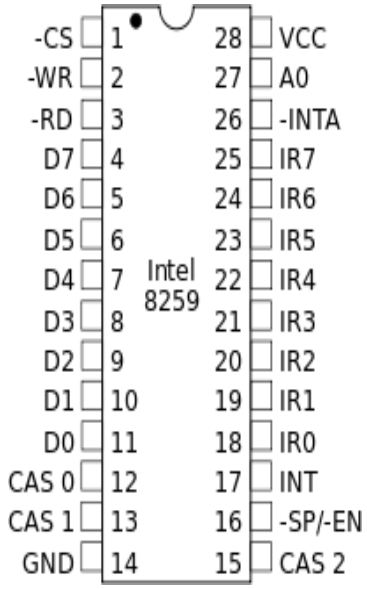


Fig.3.17 Pin Diagram of 8259

PIN DESCRIPTION OF 8259

Symbol	Pin No.	Type	Name and Function
V_{CC}	28	I	**SUPPLY:** +5V Supply.
GND	14	I	**GROUND**
$\overline{CS}$	1	I	**CHIP SELECT:** A low on this pin enables $\overline{RD}$ and $\overline{WR}$ communication between the CPU and the 8259A. INTA functions are independent of CS.
$\overline{WR}$	2	I	**WRITE:** A low on this pin when CS is low enables the 8259A to accept command words from the CPU.
$\overline{RD}$	3	I	**READ:** A low on this pin when CS is low enables the 8259A to release status onto the data bus for the CPU.
D_7-D_0	4–11	I/O	**BIDIRECTIONAL DATA BUS:** Control, status and interrupt-vector information is transferred via this bus.
CAS_0-CAS_2	12, 13, 15	I/O	**CASCADE LINES:** The CAS lines form a private 8259A bus to control a multiple 8259A structure. These pins are outputs for a master 8259A and inputs for a slave 8259A.
$\overline{SP}/\overline{EN}$	16	I/O	**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 (SP = 1) or slave (SP = 0).
INT	17	O	**INTERRUPT:** This pin goes high whenever a valid interrupt request is asserted. It is used to interrupt the CPU, thus it is connected to the CPU's interrupt pin.
IR_0-IR_7	18–25	I	**INTERRUPT REQUESTS:** Asynchronous inputs. An interrupt request is executed by raising an IR input (low to high), and holding it high until it is acknowledged (Edge Triggered Mode), or just by a high level on an IR input (Level Triggered Mode).
$\overline{INTA}$	26	I	**INTERRUPT ACKNOWLEDGE:** This pin is used to enable 8259A interrupt-vector data onto the data bus by a sequence of interrupt acknowledge pulses issued by the CPU.
A_0	27	I	**A0 ADDRESS LINE:** This pin acts in conjunction with the $\overline{CS}$, $\overline{WR}$, and $\overline{RD}$ pins. It is used by the 8259A to decipher various Command Words the CPU writes and status the CPU wishes to read. It is typically connected to the CPU A0 address line (A1 for 8086, 8088).

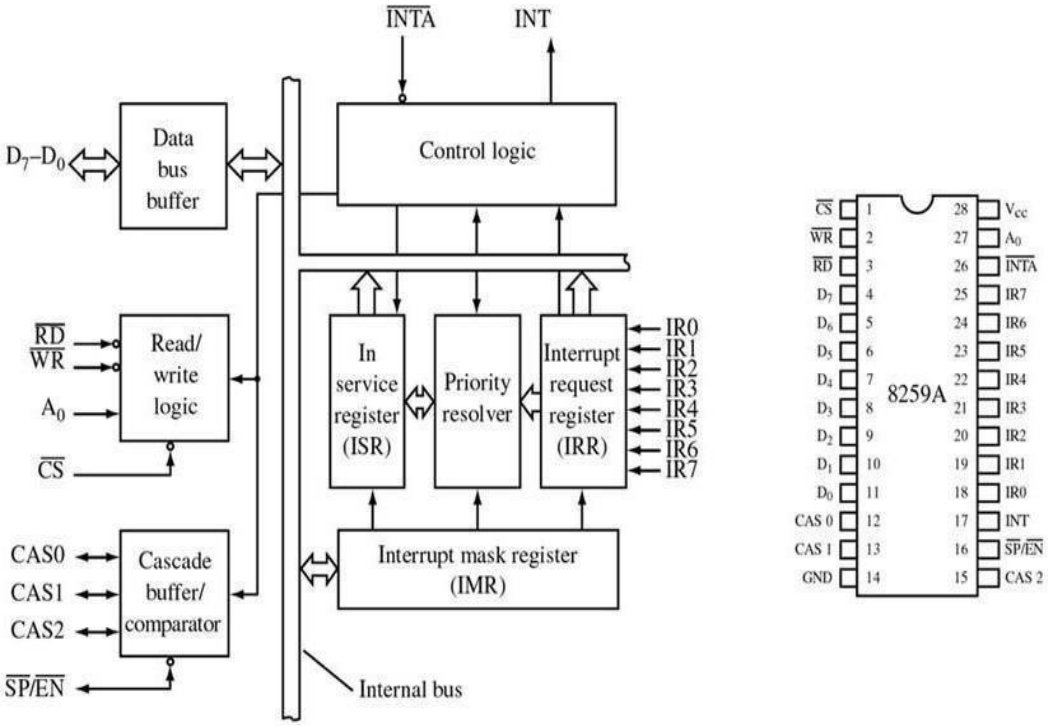


Fig.3. 18 Block Diagram of 8259

A more desirable method would be one that would allow the microprocessor to be executing its main program and only stop to service peripheral devices when it is told to do so by the device itself. In effect, the method would provide an external asynchronous input that would inform the processor that it should complete whatever instruction that is currently being executed and fetch a new routine that will service the requesting device. Once this servicing is complete, however, the processor would resume exactly where it left off. This method is calledInterrupt. It is easy to see that system throughput would drastically increase, and thus more tasks could be assumed by the micro-computer to further enhance its cost effectiveness. BlockDiagram of 8259 is shown in figure 3.18.

The Programmable Interrupt Controller (PIC) functions as an overall manager in an Interrupt- Driven system environment. It accepts requests from the peripheral equipment, determines which of the in-coming requests is of the highest importance (priori-ty), ascertains whether the incoming request has a higher priority value than the

level currently being serviced, and issues an interrupt to the CPU based on this determination.

The 8259A is a device specifically designed for use in real time, interrupt driven microcomputer systems. It manages eight levels or requests and has built-in features for expandability to other 8259A's (up to 64 levels). It is programmed by the system's software as an I/O peripheral. A selection of priority modes is available to the programmer so that the manner in which the requests are processed by the 8259A can be configured to match his system requirements. The priority modes can be changed or reconfigured dynamically at any time during the main program. This means that the complete interrupt structure can be defined as required, based on the total system environment.

INTERRUPT REQUEST REGISTER (IRR) AND IN-SERVICE REGISTER (ISR)

The interrupts at the IR input lines are handled by two registers in cascade, the Interrupt Request Register (IRR) and the In-Service (ISR). The IRR is used to store all the interrupt levels which are requesting service; and the ISR is used to store all the interrupt levels which are being serviced.

PRIORITY RESOLVER

This logic block determines the priorites of the bits set in the IRR. The highest priority is selected and strobed into the corresponding bit of the ISR during INTA pulse.

INTERRUPT MASK REGISTER (IMR)

The IMR stores the bits which mask the interrupt lines to be masked. The IMR operates on theIRR. Masking of a higher priority input will not affect the interrupt request lines of lower quality.

INT (INTERRUPT)

This output goes directly to the CPU interrupt input. The VOH level on this line is designed tobe fully compatible with the 8080A, 8085A and 8086 input levels.

INTA (INTERRUPT ACKNOWLEDGE)

INTA pulses will cause the 8259A to release vectoring information onto the data bus. The format of this data depends on the system mode (mPM) of the 8259A.

DATA BUS BUFFER

This 3-state, bidirectional 8-bit buffer is used to inter-face the 8259A to the system Data Bus. Control words and status information are transferred through the Data Bus Buffer.

READ/WRITE CONTROL LOGIC

The function of this block is to accept Output commands from the CPU. It contains the Initialization Command Word (ICW) registers and Operation Command Word (OCW) registers which store the various control formats for device operation. This function block also allows the status of the 8259A to be transferred onto the Data Bus.

CS (CHIP SELECT)

A LOW on this input enables the 8259A. No reading or writing of the chip will occur unless the device is selected.

WR (WRITE)

A LOW on this input enables the CPU to write con-trol words (ICWs and OCWs) to the 8259A.

RD (READ)

A LOW on this input enables the 8259A to send the status of the Interrupt Request Register (IRR), In Service Register (ISR), the Interrupt Mask Register (IMR), or the Interrupt level ontothe Data Bus.

A0

This input signal is used in conjunction with WR and RD signals to write commands into the various command registers, as well as reading the various status registers of the chip. This linecan be tied directly to one of the address lines.

INTERRUPT SEQUENCE

The powerful features of the 8259A in a microcomputer system are its programmability and the interrupt routine addressing capability. The latter allows direct or indirect jumping to the specific interrupt routine requested without any polling of the interrupting devices. The normal sequence of events during an interrupt depends on the type of CPU being used.

The events occur as follows in an MCS-80/85 sys-tem:

- One or more of the INTERRUPT REQUEST lines (IR7±0) are raised high, setting the correspond-ing IRR bit(s).
- The 8259A evaluates these requests, and sends an INT to the CPU, if appropriate.
- The CPU acknowledges the INT and responds with an INTA pulse.
- Upon receiving an INTA from the CPU group, the highest priority ISR bit is set, and the correspond-ing IRR bit is reset. The 8259A will also release a CALL instruction code (11001101) onto the 8-bit Data Bus through its D7±0 pins.
- This CALL instruction will initiate two more INTA pulses to be sent to the 8259A from the CPU group.
- These two INTA pulses allow the 8259A to re-lease its preprogrammed subroutine address onto the Data Bus. The lower 8-bit address is released at the first INTA pulse and the higher 8-bit address is released at the second INTA pulse.
- This completes the 3-byte CALL instruction re-leased by the 8259A. In the AEOI mode the ISR bit is reset at the end of the third INTA pulse. Otherwise, the ISR bit remains set until an appropriate EOI command is issued at the end of the interrupt sequence.
- The events occurring in an 8086 system are the same until step 4.
- Upon receiving an INTA from the CPU group, the highest priority ISR bit is set and the corresponding IRR bit is reset. The 8259A does not drive the Data Bus during this cycle.
- The 8086 will initiate a second INTA pulse. During this pulse, the 8259A releases an 8-bit pointer onto the Data Bus where it is read by the CPU.
- This completes the interrupt cycle. In the AEOI mode the ISR bit is reset at the end of the second INTA pulse. Otherwise, the ISR bit remains set until an appropriate EOI command is issued at the end of the interrupt subroutine.

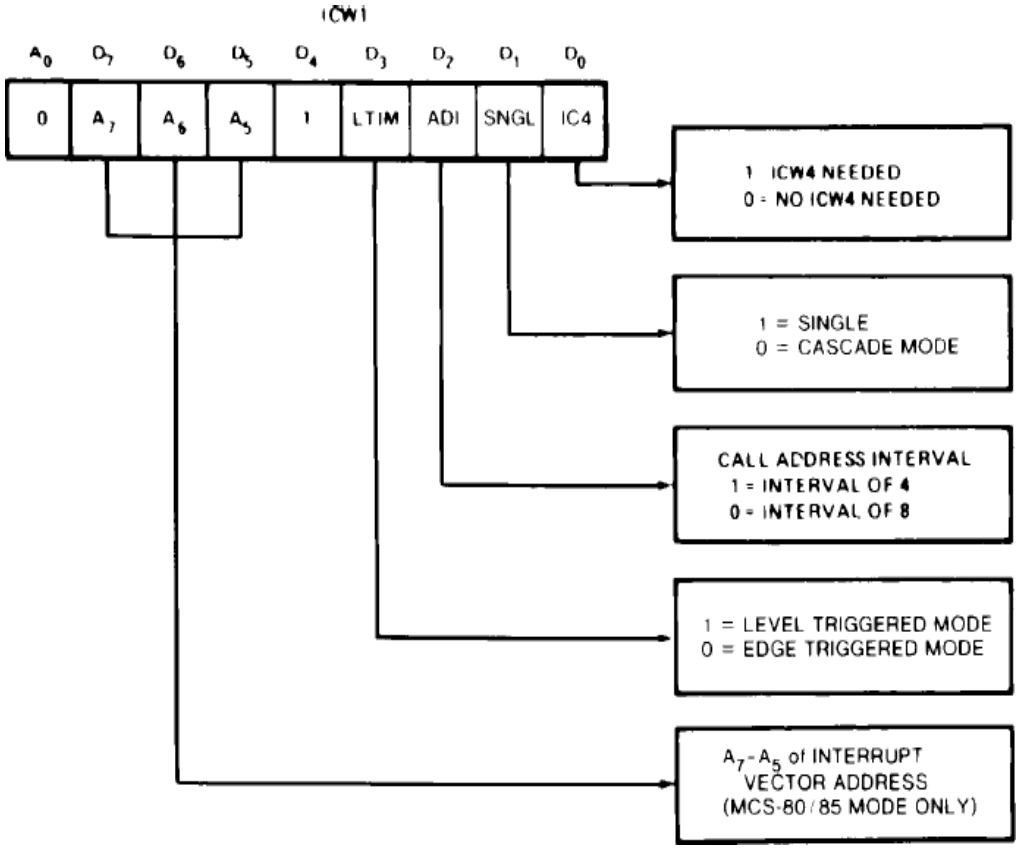
If no interrupt request is present at step 4 of either sequence (i.e., the request was too short in duration) the 8259A will issue an interrupt level 7. Both the vectoring bytes and the CAS lines will look like an interrupt level 7 was requested.

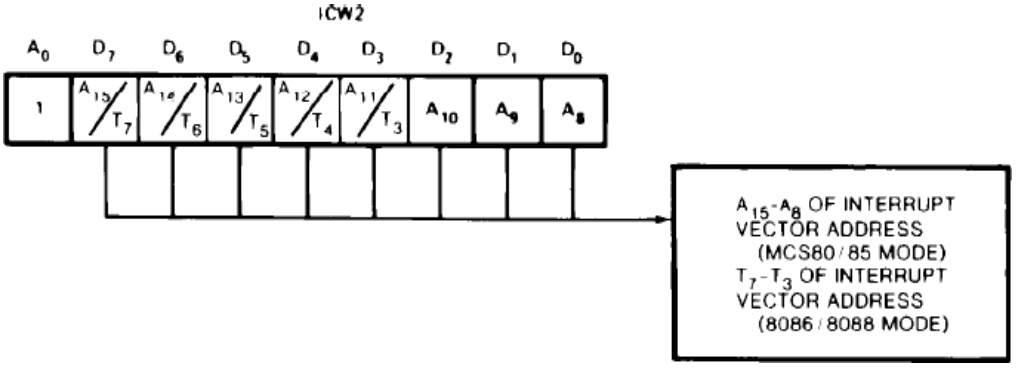
When the 8259A PIC receives an interrupt, INT be-comes active and an interrupt acknowledgecycle is started. If a higher priority interrupt occurs between the two INTA pulses, the INT linegoes inactive immediately after the second INTA pulse. After an un- specified amount of timethe INT line is activated again to signify the higher priority interrupt waiting for service. Thisinactive time is not specified and can vary between parts. The designer should be aware of this consideration when designing a sys-tem which uses the 8259A. It is recommended that proper asynchronous design techniques be followed.

INITIALIZATION COMMAND WORDS

Whenever a command is issued with A0 e 0 and D4 e 1, this is interpreted as Initialization Command Word 1 (ICW1). ICW1 starts the initialization sequence during which the followingautomatically occur.

- The edge sense circuit is reset, which means that following initialization, an interruptrequest (IR) input must make a low-to-high transition to generate an interrupt.
- The Interrupt Mask Register is cleared.
- IR7 input is assigned priority 7.
- The slave mode address is set to 7.
- Special Mask Mode is cleared and Status Read isset to IRR.
- If IC4 e 0, then all functions selected in ICW4are set to zero. (Non-Buffered mode(,noAuto-EOI, MCS-80, 85 system).
- Initialization Command Word Format is as shown in figure 3.19.


ICW1
A0 D7 D6 D5 D4 D3 D2 D1 D0
0 A7 A6 A5 1 LTIM ADI SNGL IC4
1 ICW4 NEEDED
0 = NO ICW4 NEEDED
1 = SINGLE
0 = CASCADE MODE
CALL ADDRESS INTERVAL
1 = INTERVAL OF 4
0 = INTERVAL OF 8
1 = LEVEL TRIGGERED MODE
0 = EDGE TRIGGERED MODE
A7-A5 of INTERRUPT
VECTOR ADDRESS
(MCS-80/85 MODE ONLY)


ICW2
A0 D7 D6 D5 D4 D3 D2 D1 D0
1 A15/T7 A14/T6 A13/T5 A12/T4 A11/T3 A10 A9 A8
A15-A8 OF INTERRUPT
VECTOR ADDRESS
(MCS80/85 MODE)
T7-T3 OF INTERRUPT
VECTOR ADDRESS
(8086/8088 MODE)

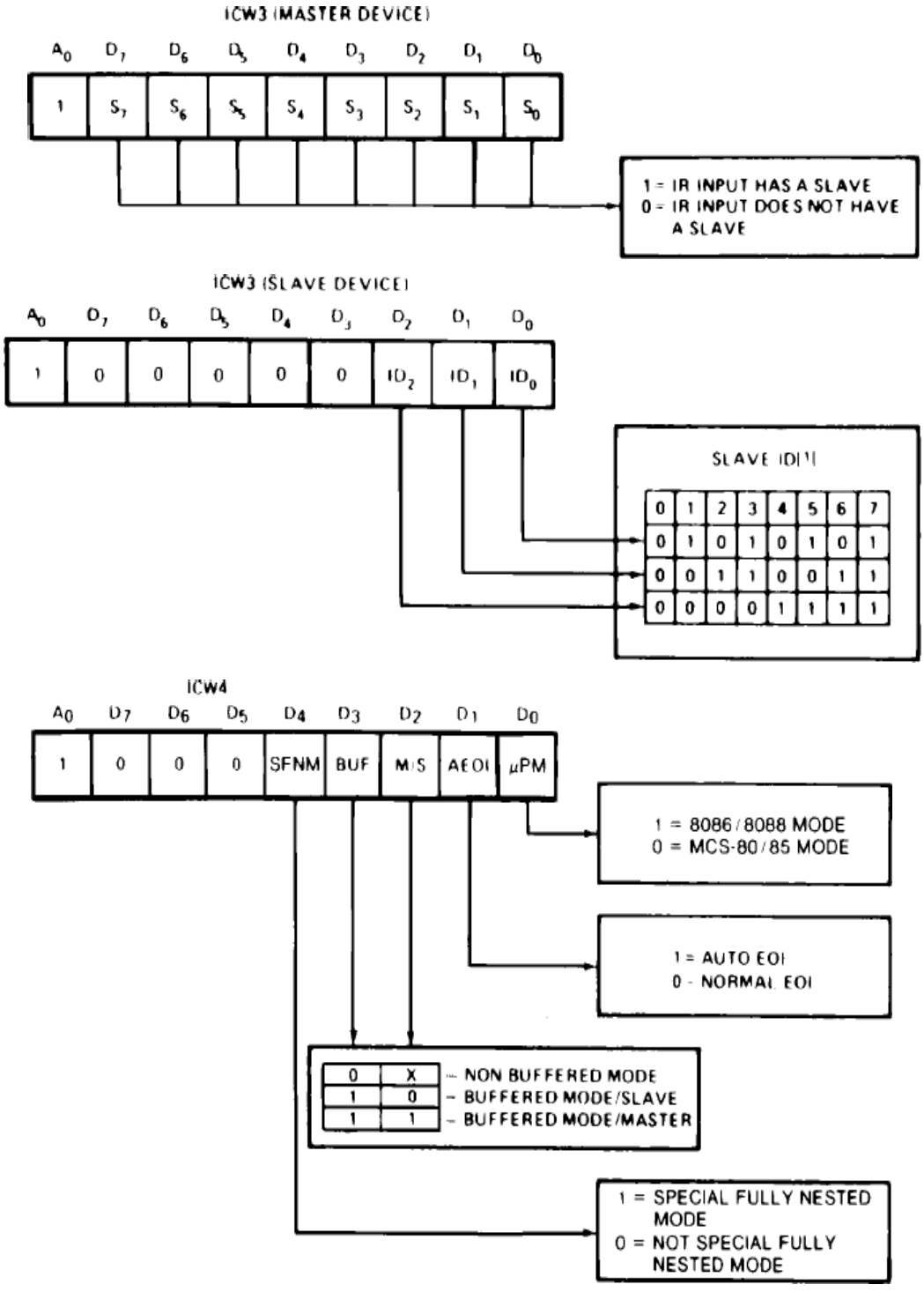


Fig 3.19 . Initialization Command Word Format

OPERATION COMMAND WORDS

After the Initialization Command Words (ICWs) are programmed into the 8259A, the chip is ready to accept interrupt requests at its input lines. However, during the 8259A operation, a selection of algorithms can command the 8259A to operate in various modes through the Operation Command Words (OCWs). Operation Command Word format is as shown in figure20

OCW1

A0	D7	D6	D5	D4	D3	D2	D1	D0
1	M7	M6	M5	M4	M3	M2	M1	M0

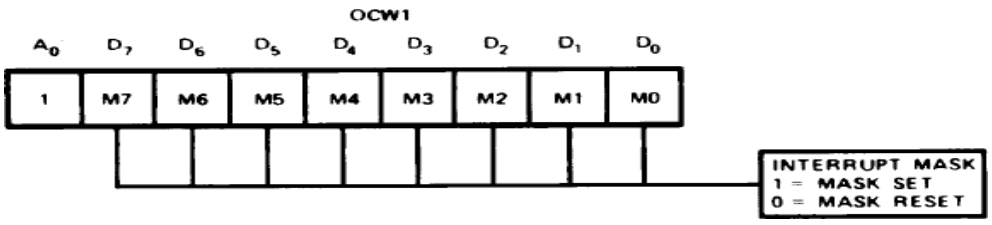
OCW2

0	R	SL	EOI	0	0	L2	L1	L0

OCW3

0	0	ESMM	SMM	0	1	P	RR	RIS

Fig . Operational Control Words



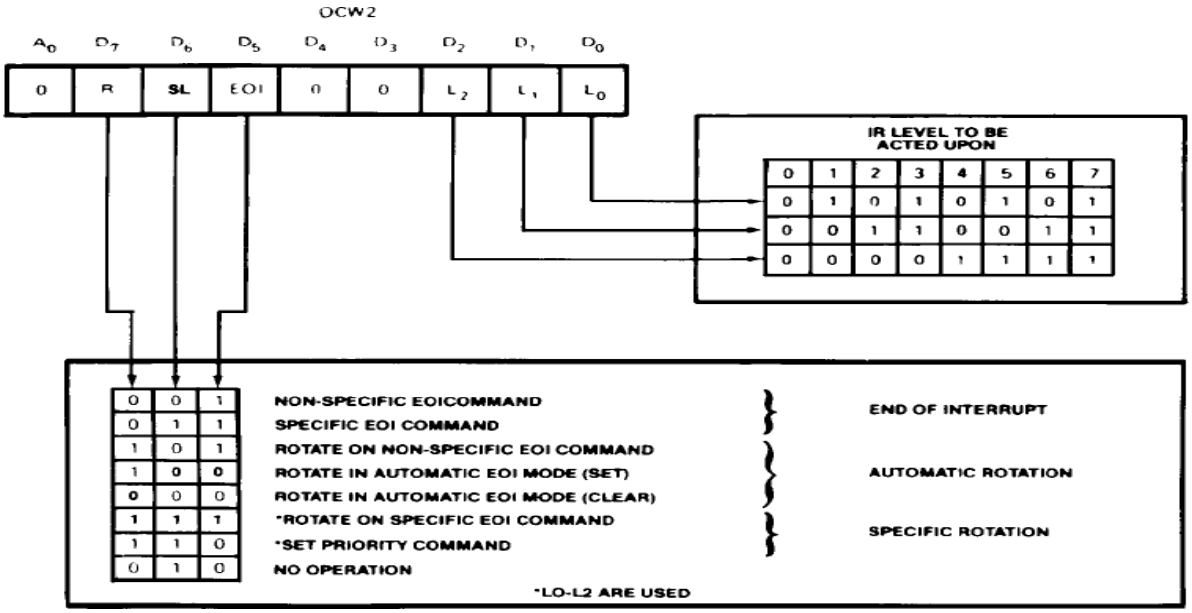


Fig 3.20 Operation Command Word Format

DMA CONTROLLER 8257

The Direct Memory Access or DMA mode of data transfer is the fastest amongstall the modes of data transfer. In this mode, the device may transfer data directly to/from memory without any interference from the CPU. The device requests the CPU (through aDMA controller) to hold its data, address and control bus, so that the device may transfer data directly to/from memory. The DMA data transfer is initiated only after receiving HLDA signal from the CPU. Intel's 8257 is a four channel DMA controller designed to be interfaced with their family of microprocessors. The 8257, on behalf of the devices, requests the CPU for bus access using local bus request input i.e. HOLD in minimum mode. In maximum mode of the microprocessor RQ/GT pin is used as bus request input. On receiving the HLDA signal (in minimum mode) orRQ/GT signal (in maximum mode) from the CPU, the requesting devices gets the access of the bus, and it completes the required number of DMA cycles for the data transfer and then hands over the control of the bus back to the CPU.

INTERNAL ARCHITECTURE OF 8257

The internal architecture of 8257 is shown in figure. The chip support four DMA channels, i.e. four peripheral devices can independently request for DMA data transfer

through these channels at a time. The DMA controller has 8-bit internal data buffer, a read/write unit, a control unit, a priority resolving unit along with a set of registers.

The 8257 performs the DMA operation over four independent DMA channels. Each of four channels of 8257 has a pair of two 16-bit registers, viz. DMA address register and terminal count register.

There are two common registers for all the channels, namely, mode set register and status register. Thus there are a total of ten registers. The CPU selects one of these ten registers using address lines Ao-A3. Table shows how the Ao-A3 bits may be used for selecting one of theseregisters.

DMA ADDRESS REGISTER

Each DMA channel has one DMA address register. The function of this register is to store theaddress of the starting memory location, which will be accessed by the DMA channel. Thus the starting address of the memory block which will be accessed by the device is first loaded in the DMA address register of the channel. The device that wants to transfer data over a DMAchannel, will access the block of the memory with the starting address stored in the DMA Address Register.

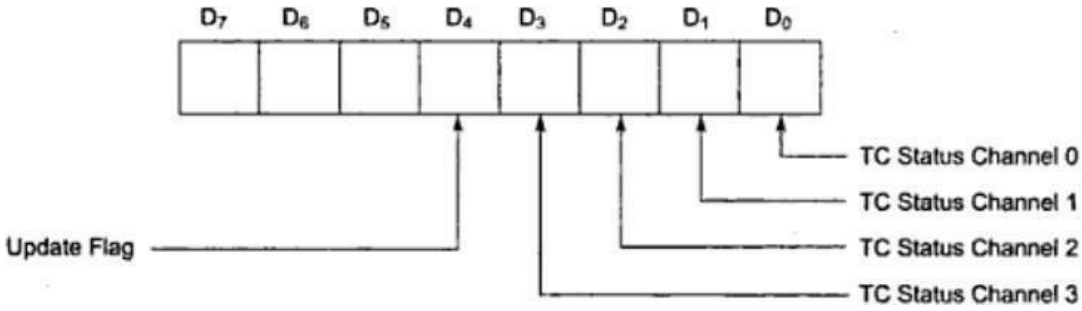
TERMINAL COUNT REGISTER

Each of the four DMA channels of 8257 has one terminal count register (TC). This 16-bit register is used for ascertaining that the data transfer through a DMA channel ceases or stops after the required number of DMA cycles. The low order 14-bits of the terminal count register are initialized with the binary equivalent of the number of required DMA cycles minus one. After each DMA cycle, the terminal count register content will be decremented by one and finally it becomes zero after the required number of DMA cycles are over. The bits14 and 15 of this register indicate the type of the DMA operation (transfer). If the device wants to write data into the memory, the DMA operation is called DMA write operation. Bit 14 of the registerin this case will be set to one and bit 15 will be set to zero.

STATUS REGISTER

The status register of 8257 is shown in figure. The lower order 4-bits of this register contain the terminal count status for the four individual channels. If any of these bits is set, it indicatesthat the specific channel has reached the terminal count condition.

These bits remain set till either the status is read by the CPU or the 8257 is reset. The update flag is not affected by the read operation. This flag can only be cleared by resetting 8257 or byresetting the auto load bit of the mode set register. If the update flag is set, the contents of the channel 3 registers are reloaded to the corresponding registers of channel 2 whenever the channel 2 reaches a terminal count condition, after transferring one block and the next block isto be transferred using the autoload feature of 8257.



The update flag is set every time, the channel 2 registers are loaded with contents of the channel3 registers. It is cleared by the completion of the first DMA cycle of the new block. This registercan only read.

DATA BUS BUFFER, READ/WRITE LOGIC, CONTROL UNIT AND PRIORITY RESOLVER

The 8-bit. Tristate, bidirectional buffer interfaces the internal bus of 8257 with the external system bus under the control of various control signals. In the slave mode, the read/write logicaccepts the I/O Read or I/O Write signals, decodes the Ao-A3 lines and either writes the contents of the data bus to the addressed internal register or reads the

contents of the selected register depending upon whether IOW or IOR signal is activated.

In master mode, the read/write logic generates the IOR and IOW signals to control the data flow to or from the selected peripheral. The control logic controls the sequences of operations and generates the required control signals like AEN, ADSTB, MEMR, MEMW, TC and MARK along with the address lines A4-A7, in master mode. The priority resolver resolves thepriority of the four DMA channels depending upon whether normal priority or rotating priorityis programmed.

SIGNAL

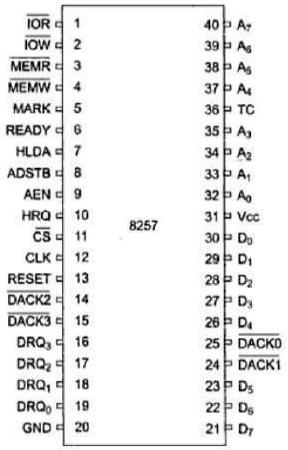
DESCRIPTION OF

8257DRQ0-DRQ3

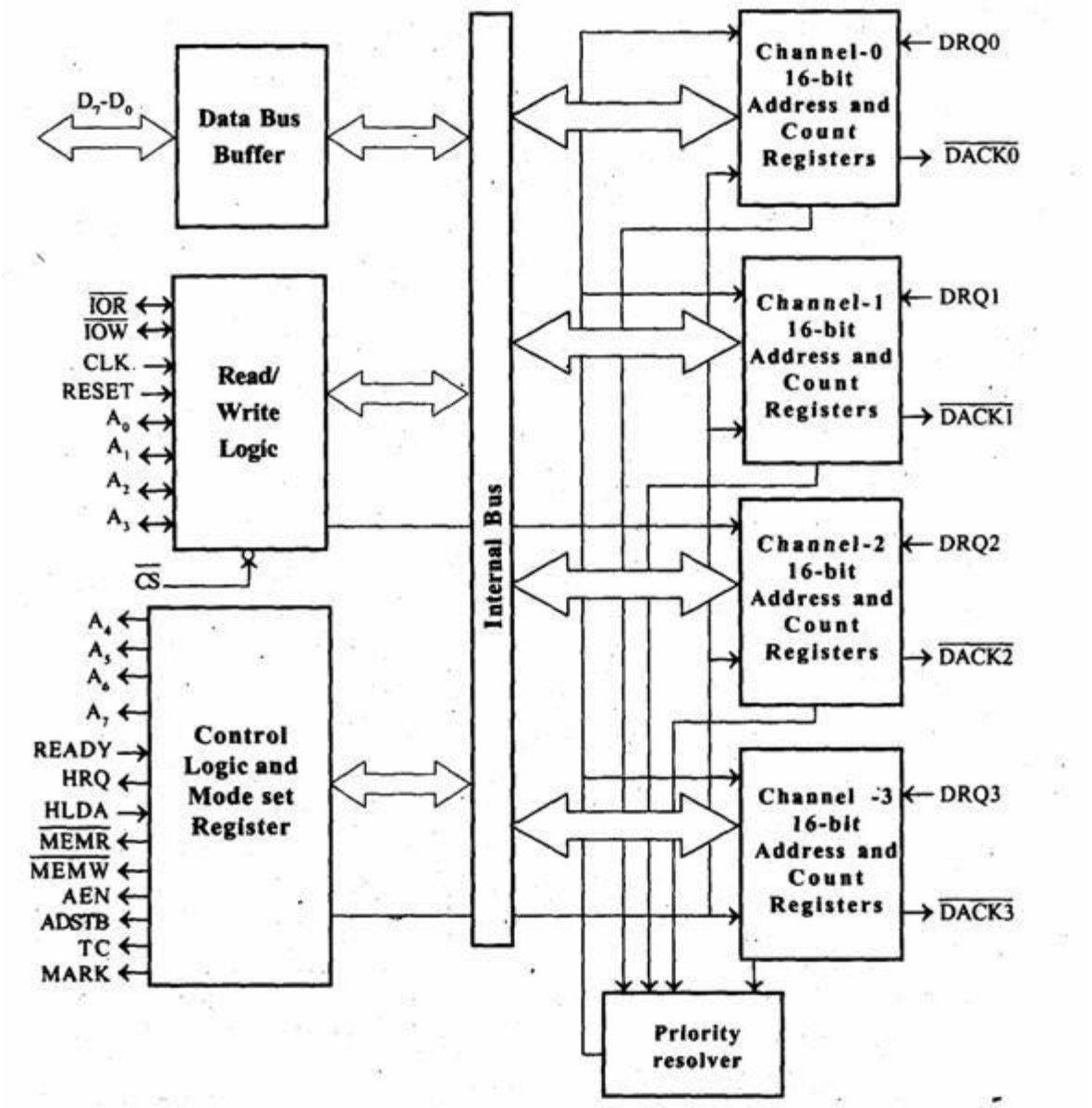
These are the four individual channel DMA request inputs, used by the peripheral devices forrequesting the DMA services. The DRQ0 has the highest priority while DRQ3 has the lowest one, if the fixed priority mode is selected.

DACK0-DACK3:

These are the active-low DMA acknowledge output lines which inform the requestingperipheral that the request has been honoured and the bus is relinquished by the CPU. These lines may act as strobe lines for the requesting devices



.Pin Description of 8257



Architecture of 8257

Do-D7:

These are bidirectional, data lines used to interface the system bus with theinternal data bus of 8257. These lines carry command words to 8257 and status wordfrom 8257, in slave mode, i.e.under the control of CPU.The data over these lines may be transferred in both the directions. When the 8257 is thebus master (master mode, i.e. not under

CPU control), it uses Do-D7 lines to send higherbyte of the generated address to the latch. Thisaddress is further latched using ADSTBsignal. the address is transferred over Do-D7 during the first clock cycle of the DMAcycle. During the rest of the period, data is available on the data bus.

IOR:

This is an active-low bidirectional tristate input line that acts as an input in theslave mode. In slave mode, this input signal is used by the CPU to read internal registersof 8257.this line actsoutput in master mode. In master mode, this signal is used to readdata from a peripheral duringa memory write cycle.

IOW:

This is an active low bidirection tristate line that acts as input in slave mode to load the contents of the data bus to the 8-bit mode register or upper/lower byte of a 16-bit DMA addressregister or terminal count register. In the master mode, it is a control output that loads the datato a peripheral during DMA memory read cycle (write to peripheral).

CLK:

This is a clock frequency input required to derive basic system timings for theinternal operationof 8257.

RESET :

This active-high asynchronous input disables all the DMA channels by clearing the mode register and tristates all the control lines.

Ao-A3:

These are the four least significant address lines. In slave mode, they act as input which selectone of the registers to be read or written. In the master mode, they are the four least significantmemory address output lines generated by 8257.

CS:

This is an active-low chip select line that enables the read/write operations from/to 8257, in slave mode. In the master mode, it is automatically disabled to prevent the chip from getting selected (by CPU) while performing the DMA operation.

A4-A7:

This is the higher nibble of the lower byte address generated by 8257 during the master modeof DMA operation.

READY:

This is an active-high asynchronous input used to stretch memory read and writecycles of 8257 by inserting wait states. This is used while interfacing slower peripherals.

HRQ:

The hold request output requests the access of the system bus. In the noncascaded8257 systems,this is connected with HOLD pin of CPU. In the cascade mode, this pin of a slave is connectedwith a DRQ input line of the master 8257, while that of the master is connected with HOLD input of the CPU.

HLDA :

The CPU drives this input to the DMA controller high, while granting the bus tothe device. This pin is connected to the HLDA output of the CPU. This input, if high, indicates to the DMA controller that the bus has been granted to the requesting peripheral by the CPU.

MEMR:

This active –low memory read output is used to read data from the addressed memory locationsduring DMA read cycles.

MEMW :

This active-low three state output is used to write data to the addressed memory location duringDMA write operation.

ADST :

This output from 8257 strobes the higher byte of the memory address generated by the DMA controller into the latches.

AEN:

This output is used to disable the system data bus and the control the bus driven by the CPU, this may be used to disable the system address and data bus by using the enable input of the bus drivers to inhibit the non-DMA devices from responding during DMA operations. If the 8257 is I/O mapped, this should be used to disable the other I/O devices, when the DMA controller addresses is on the address bus.

TC:

Terminal count output indicates to the currently selected peripherals that thepresent DMA cycleis the last for the previously programmed data block. If the TC STOP bit in the mode set registeris set, the selected channel will be disabled at the end of the DMA cycle. The TC pin is activatedwhen the 14-bit content of the terminal count register of the selected channel becomes equal tozero. The lower order 14 bits of the terminal count register are to be programmed with a 14-bitequivalent of (n-1), if n is the desired number of DMA cycles.

MARK:

The modulo 128 mark output indicates to the selected peripheral that the current DMA cycle is the 128th cycle since the previous MARK output. The mark will be activated after each 128cycles or integral multiples of it from the beginning if the data block (the first DMA cycle), ifthe total number of the required DMA cycles (n) is completely divisible by 128.

Vcc:

This is a +5v supply pin required for operation of the circuit. GND:This is a return line for the supply (ground pin of the IC).

REFERENCE BOOKS

1. Ramesh Gaonkar, "Microprocessor Architecture, Programming and applications with8085", 5th Edition, Penram International Publishing Pvt Ltd, 2010.
2. D. V. Hall, "Microprocessor Interfacing, Programming and Hardware", McGraw Hill,1993.
3. Nagoor Kani A, "Microprocessor (8085) and its Applications", 2nd Edition, RBApublications.
4. Mathur A.P, "Introduction to Microprocessor", Tata McGraw Hill, 1990.

PART – A QUESTIONS

S.NO	QUESTION
1	What is memory?
2	What is interfacing?
3	What Is the need of 8255?
4	What is the need of 8253?
5	What is the need of 8259?
6	What is the need of 8257?
7	What Is Chip Select? How It Is Generated?
8	What are the different types of semiconductor memory?
9	What is SRAM and DRAM
10	List the characteristics of SRAM and DRAM?

PART – B QUESTIONS

S.NO	QUESTION
1	Describe the operation and applications of 8257 with neat diagrams.
2	Explain the initialization sequence of 8259.
3	Write about the command words of 8259 and explain briefly about 8259.
4	Draw and explain the functional block diagram of programmable peripheral interface (8255). Write about the command words of 8255.
5	Draw and explain the functional block diagram of programmable counter/interval timer (8254). Write about the command words of 8253.