Overview of the working of 8259

The interrupt requests are accepted by 8259 from many interrupting devices IR0 to IR7 pins. After that, it identifies the highest priority interrupt request from those inputs that are already active. To configure the 8259 for fixed priority mode of operation, among them IR0 has the highest and IR7 has the lowest priority. If the inputs IR2, IR4, and IR6 are active, thenIR2 has the highest priority interrupt request among the active requests than the other. The details of the interrupt requests those are active are stored in the Interrupt Request Register (IRR).

By loading the Interrupt Mask Register (IMR), it is possible to mask the interrupt request. If the interrupt requests IR2 and IR3 are masked, then IR4 will get the highest priority interrupt request among the active requests that are not masked. It is possible that the processor is already servicing IR5 interrupt request. All the information about the interrupt requests that are presently services is kept in In-Service Register in short abbreviation (ISR). Apriority resolver unit exists in 8259 which receives inputs requests from IRR, ISR, and IMR and identifies the highest priority interrupt request. Since the priority of IR4 is much greater than IR5 which is currently being serviced, the INT (Interrupt Request) of the output is activated. At the same time, bit no 4 of ISR is set to 1 by the8259. The output of INT of 8259 is connected to INTR input of 8085 as shown in the figure below. The INT output of 8259 should not get connected to any other interrupt pin of 8085.

Hence the priority resolver decides to activate the INT output only when the following conditions are satisfied otherwise not.

- •Activation to IR input is done
- •When the masking of the IR input is not done
- •When the processor is currently not servicing an IR request with a higher priority.

Fig − 8259 interfaced along with 8085 processor

8085 executes the instruction during which the INTR input was activated. Then the 8085sends out INTA* output thrice in succession assuming that the 8085 interrupt system is enabled, and higher priority interrupts of 8085 are not active. In response to the activation of INTA*, the 8259 sends to the 8085 using the D7-0 pins a 3-byte CALL instruction. The first time the INTA* is activated, the 8259 sends the code for CALL (5CDH) to the 8085 on D7-0 pins. It is received in the IR register of 8085. The second time the INTA*is activated the 8259 sends LS byte of interrupt vector (IV) address to the 8085 on D7-0 pins. It is received in the Z register of 8085. The third time the INTA* is activated the 8259 sends MS byte of IV address to the 8085 on D7-0 pins. It is received in the W register of 8085. The IV address supplied by the 8259 to the 8085 depends on the IR input of 8259 that is being serviced. This results in a branch to the appropriate ISS. After finishing the ISS the control returns to the main program.

The important thing to note is that the processor is not required to identify the source of the interrupt on the INTR pin. The 8259 has the mechanism to identify the source of an interrupt from among IR0 to IR7. It sends to the 8085the CALL instruction with appropriate ISS address accordingly. Thus the problem of polling is eliminated and so the interrupt response time is reduced. Second, the 8259 could be configured to operate in“rotating priority mode”. Then the disadvantage of fixed priority is also taken care of.

By using a single 8259 in the system, eight interrupting devices can interrupt on the input of INTR of 8085. When a number of devices need toper form the interrupt-driven data transfer scheme, multiple 8259s are used.

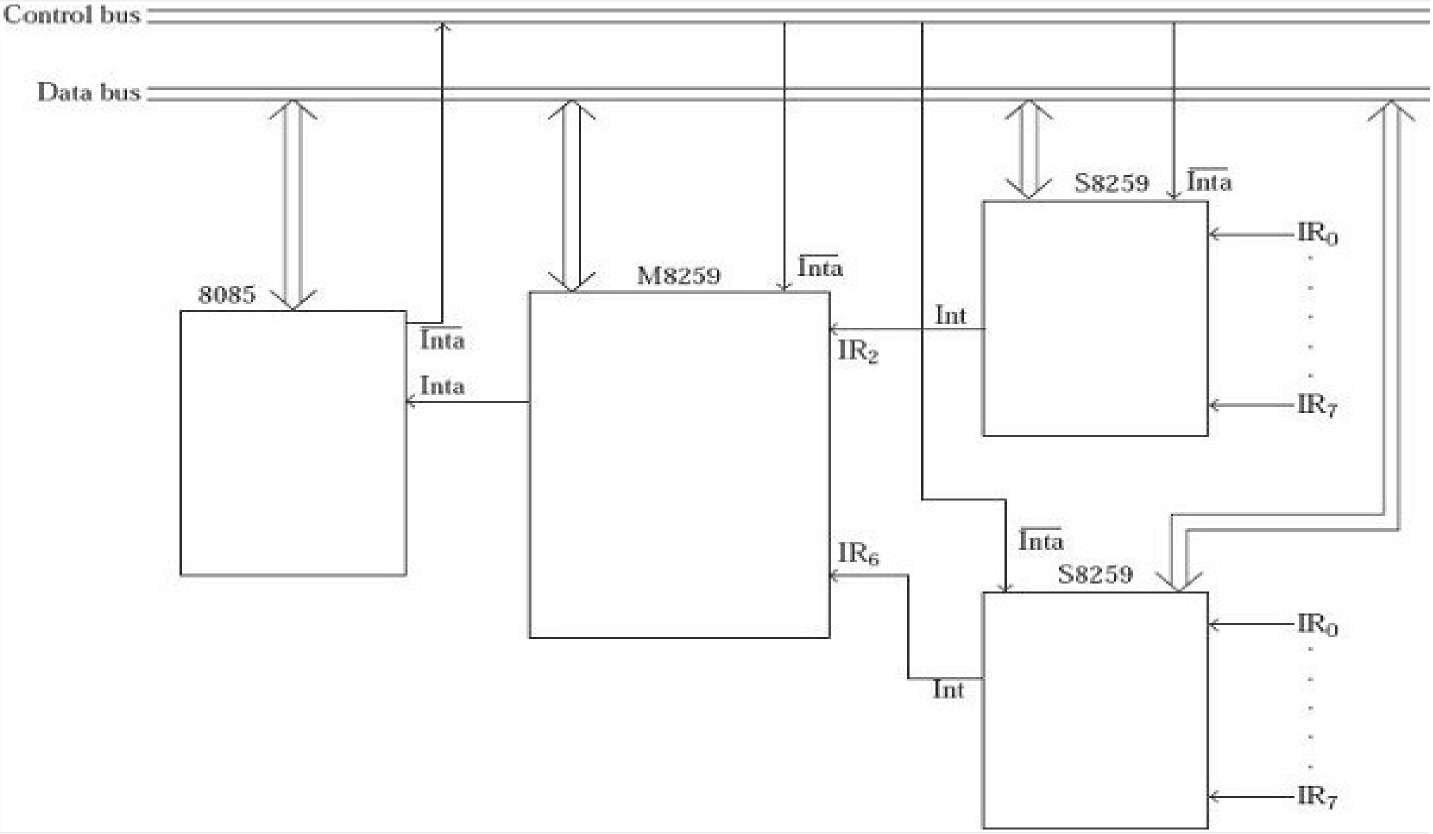
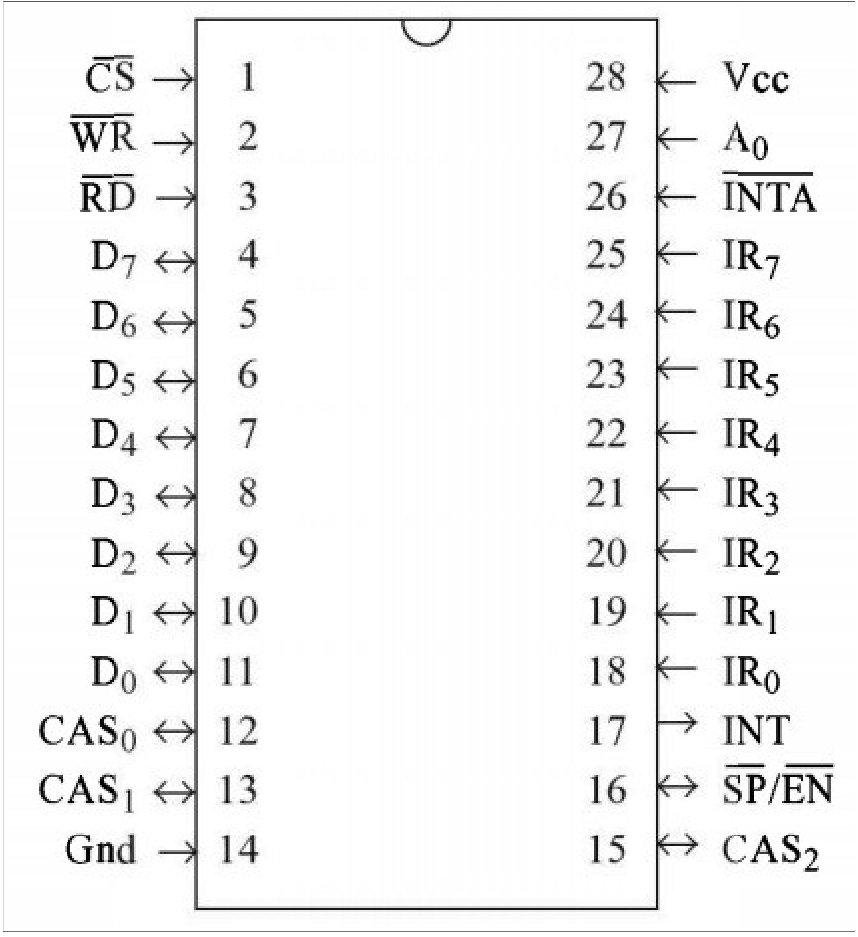


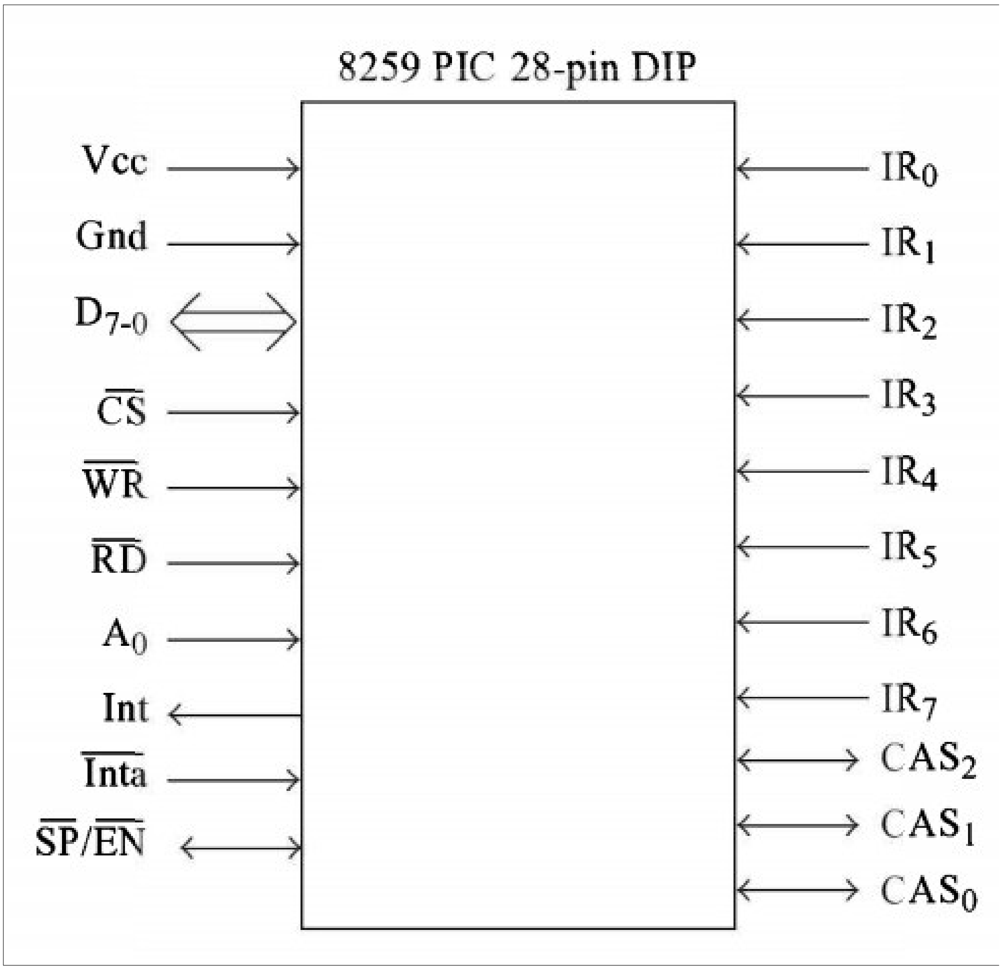
Fig – Multiple 8259s are used.

The 8259 is known as the Programmable Interrupt Controller (PIC) microprocessor. In 8085 and 8086 there are five hardware interrupts and two hardware interrupts respectively. Bu adding 8259, we can increase the interrupt handling capability. This chip combines the multi-interrupt input source to single interrupt output. This provides 8-interrupts from IR0 to IR7. Let us see some features of this microprocessor.

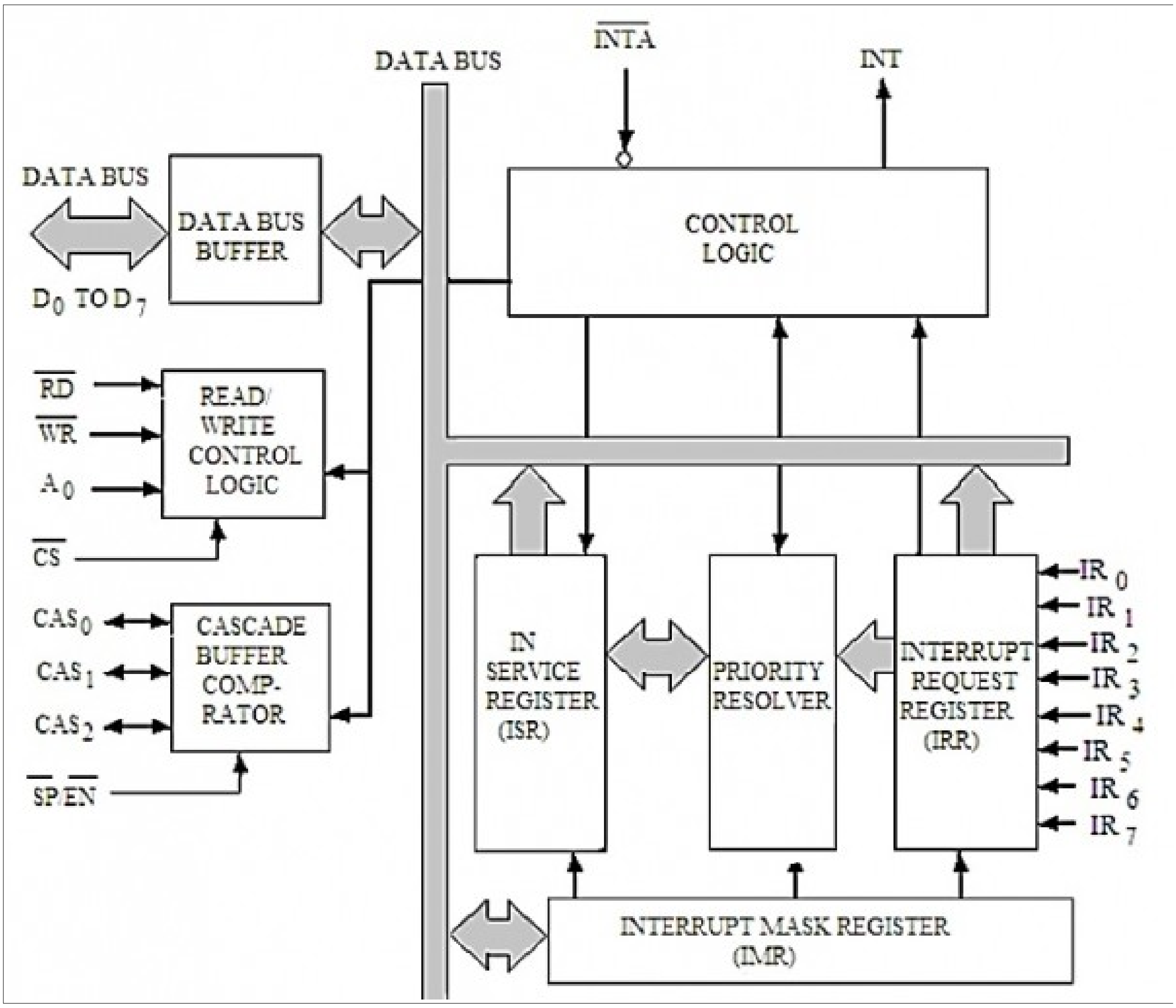
- This chip is designed for 8085 and 8086.
- It can be programmed either in edge triggered, or in level triggered mode
- We can mask individual bits of Interrupt Request Register.
- By cascading 8259 chips, we can increase interrupts up to 64 interrupt lines
- Clock cycle is not needed.

The pin level diagram and functional pin diagram is like below -





The block diagram is like below -


DATA BUS
INTA
INT
DATA BUS
DATA BUS BUFFER
D0 TO D7
CONTROL LOGIC
RD
WR
A0
CS
READ/ WRITE CONTROL LOGIC
CAS0
CAS1
CAS2
SP/EN
CASCADE BUFFER COMP-RATOR
IN SERVICE REGISTER (ISR)
PRIORITY RESOLVER
INTERRUPT REQUEST REGISTER (IRR)
IR 0
IR 1
IR 2
IR 3
IR 4
IR 5
IR 6
IR 7
INTERRUPT MASK REGISTER (IMR)

Block	Description
Data Bus Buffer	This block is used to communicate between 8259 and 8085/8086 by acting as buffer. It takes the control word from 8085/8086 and send it to the 8259. It transfers the opcode of the selected interrupts and address of ISR to the other connected microprocessor. It can send maximum 8-bit at a time.
R/W Control Logic	This block works when the value of pin CS is 0. This block is used to flow the data depending upon the inputs of RD and WR. These are active low pins for read and write.
Control Logic	It controls the functionality of each block. It has pin called INTR. This is connected to other microprocessors for taking the interrupt request. The INT pin is used to give the output. If 8259 is enabled, and also the interrupt flags of other microprocessors are high then this causes the value of the output INT pin high, and in this way this chip can responds requests made by other microprocessors.
Interrupt Request Register	It stores all interrupt level that are requesting for interrupt service.

Block	Description
Interrupt Service Register	It stores interrupt level that are currently being execute.
Interrupt Mask Register	It stores interrupt level that will be masked, by storing the masking bits of interrupt level.
Priority Resolver	It checks all three registers, and set the priority of the interrupts. Interrupt with the highest priority is set in the ISR register. It also reset the interrupt level which is already been serviced in the IRR.
Cascade Buffer	To increase number of interrupt pin, we can cascade more number of pins, by using cascade buffer. When we are going to increase the interrupt capability, CSA lines are used to control multiple interrupts.

Programming the 8259 with no slaves

Now in this topic we assume that 8085 is the processor which is used in this microcomputer system. In this slave, no 8259 slaves are used. We should examine properly before 8259 PIC is used in the microcomputer system for performing the interrupt control application. 8259 is configured in such a fantastic way that is found that a variety of information are provided like for IR0 request IV, interrupts like level or edge-triggered, whether 8259s are used single or many, if ICW4 is in need or not and whether for the interrupt requests masking should be done or not. This information is only provided to 8259 if and only if the processor issues some of the following commands LIKE Initial Command Words and Operation Command Words which are stated below:

- Operation command word1 (OCW1)
- Operation command word2 (OCW2)
- Operation command word3 (OCW3)
- Initial command word1 (ICW1)
- Initial command word2 (ICW2)
- Initial command word3 (ICW3) and so on.

The point to be noted that only two initialization command words (ICW1 and ICW2) should be used since they are compulsory for performing the programming task. When slave 8259 is in the system ICW3 or the third initial command is provided. When the processor used is 8086 or some special modes of8259 is used ICW4 or the forth initial command is needed. Here in comparison, we can say that operation command words are not compulsory. Hence we may term operation command words as optional command words. But there is a rule for writing the command words that is they should be written only to the low port or to the high port. On the low port of 8259, the command words ICW1, ICW2, and ICW3 are written. And to the high port of 8259, the command words like ICW3, ICW2, OCW1, and OCW1 commands are written. In the system, the port address depends only on the chip used. Let us assume that the low port address is 50H selected when A0=0, similarly the address of the high port is 51H selected when A0=1.

Architecture of 8259

8259 Microprocessor is architected in a unique style. It can program by means of some interrupts conditions by means of level or interrupt level often called edge-triggered interrupt level. Masking is done to individual interrupt bits. As the number of 8259 increases interrupt pins up to 64 can be obtained. There are 3 registers 8259 contains along with one priority resolver(PR). They are as follows −

- **Interrupt Request Register(IIR)** − It stores the bits who requests the interrupt.
- **Interrupt service register(ISR)** − It stores the currently interrupt levels.
- **Interrupt Mask Register(IMR)** − Stores the interrupt levels to be masked.
- **PriorityResolver(PR)** − Set the priority of interrupts by examining all the three registers and set the interrupt level inISR having the highest priority.
- **SP/EN (low active pin)** − When its value is 1 it works in master mode and when its value is 0 it works in slave mode.
- **Cascade Buffer** − Used for cascading more Programmable Interrupt Controller.

Fig −8259 interfaced along with 8085 processor

Registers used in 8259

The 8259 is a specialized I/O port chip. It is never used in the interfacing of I/O devices but is only used for controlling the interrupts in a microcomputer.8259 consists of A0 as the only address input pin. Hence for a microprocessor, only two addresses are possible for the 8259 ports. The two ports can be termed as low port and high port.

The processor selects the low port when A0 = 0

The processor selects the high port when A0 = 1

The processor issues some words termed as command words to these ports so as to configure the 8259 better. There are several command words often classified as initialization command words and operation command words. 8259 has four initialization command words namely ICW1, ICW2, ICW3, and ICW4 and three operation command words namely as OCW1, OCW2, and OCW3. The processor reads the status of 8259 by reading at the two ports termed as low port and the high port. Moreover, there are several status words to be read further.

8259 always makes use of a number of 8-bit registers shown as follows for its working procedure.

Interrupt request register, Interrupt mask register and Interrupt service register

- IRR stores all the levels of interrupt requesting for Interrupt services.
- ISR stores the currently executed levels of interrupt.
- IMR stores the masking bits of the interrupt levels.

The processor writes and reads the command and status words, or accesses registers using only the high port and the low port. Identification of a command or status word or a register is completely based on A0 value.

Interrupt request register in 8259

An 8-bit register in which the tracks of active interrupt requests are kept. Whenever activation of an interrupt request input is done the bit corresponding in IRR register is set to 1. For example, if we activate the IR4 and IR6 inputs bits no 4 and 6 of IRR are set to 1 by making the contents of IRR as 01010000. But the processor is designed only to read the contents of this register but cannot write it to IRR. To read the IRR contents, the processor only has an issue the OCW3 command to the 8259 along with the LS 3 bits of the command. This results in 8259 by storing the IRR status in the low port of 8259. Sothe processor has to read the low port of 8259.

In-service register in 8259

Also, an 8-bit register which keeps track records of the interrupt requests that are currently being executed. If the request IR6 is currently being served, whose contents of ISR will be 01000000. If by any means the request to IR3 becomes active during the service process of IR6, 8259 sets bit 3 of ISR to 1 and activates the output INT. But bit 6 of ISR always remains set at 1 asIR6 request which is not fully serviced. Hence the contents ofISR become 01001000. The following assumptions stated below helps this to happen.

Use of 8259 in an 8086-based system

The interrupt requests are accepted by 8259 from eight interrupting devices on the pin ranging fromIR0 toIR7. After that, it identifies the priority interrupt having the highest request from the inputs which are active. It is possible for us to configure the 8259 for the mode of operation of "fixed priority" mode.Here, among the priorities, IR0 has the highest and IR7 has the lowest. If the three inputs IR2,IR4,and IR6 are in active state, then IR2 will have is the highest priority interrupt request than the other active requests.

We can mask the requests of the interrupts by installing the interrupt mask register. If the two requests of the interrupts IR2 andIR3 are masked, then the highest priority interrupt request among the others is IR4among the others which are not masked. Now the processor can check or service the interrupt request IR5. Now the Information about the requests of the interrupts which are presently getting serviced will be kept in (ISR).

We have a resolver unit called the priority resolver unit in 8259. The inputs are received from IRR, IMR, and ISR and identifies the priority request which has the highest priority. We know the priority of IR4is much greater comparable to IR5which is currently getting serviced.

Hence the priority resolver activates the output INT only when the given conditions are satisfied −

- •The input IR must be activated
- •The input IR should not be masked
- •The processor which is presently not doing the servicing an IR request should have the highest priority.

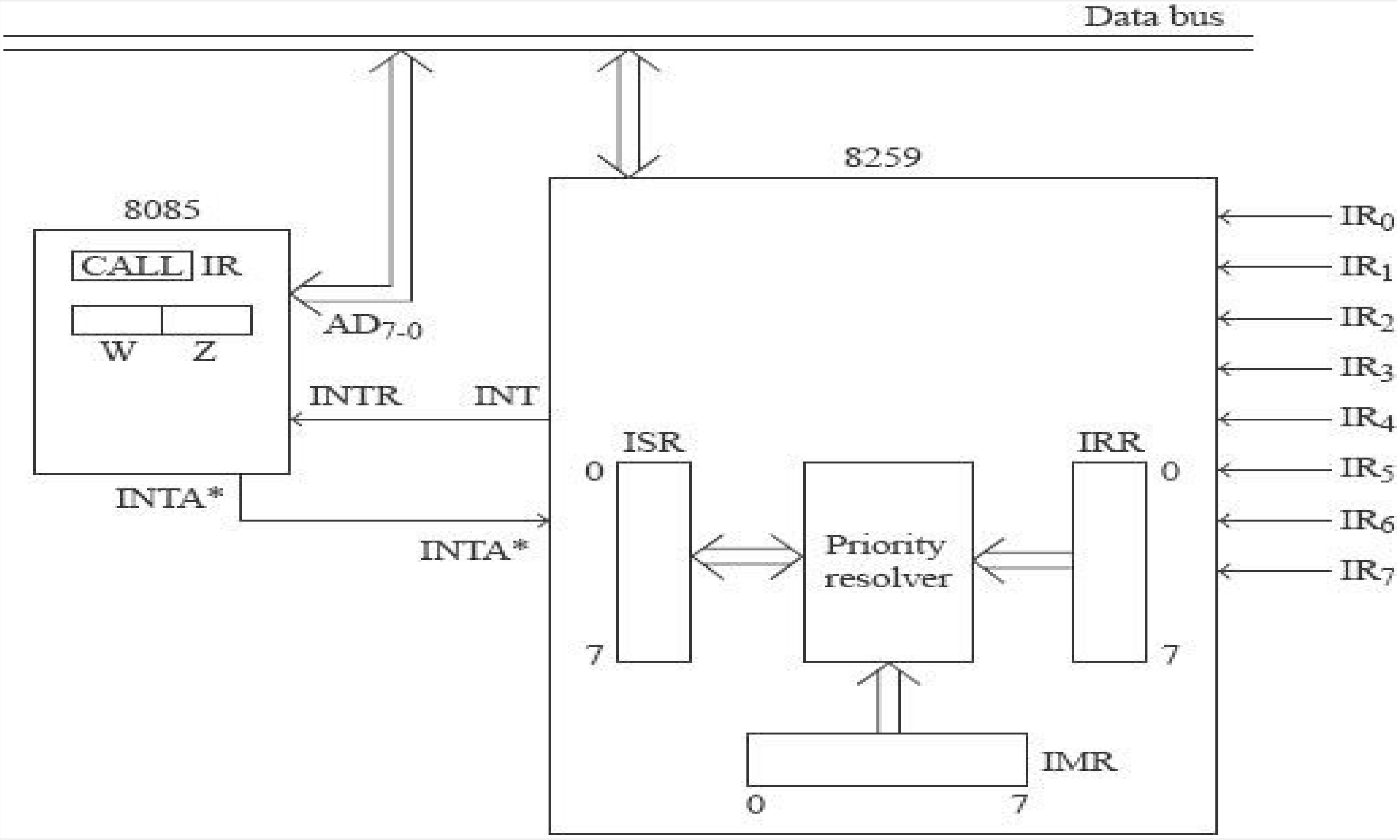


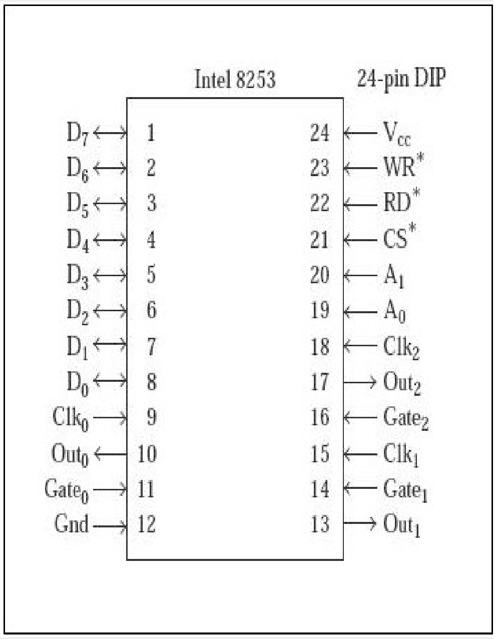
Fig:8259 interfaced along with 8085 processor

8253

Description of 8253 timer

As a DIP package Intel 8253 is a 24-pin programmable IC available. IChas three counters which work independently and whose width is of16-bits. In addition, we have a control port to decide what is the mode of working of the three counters. The physical and functional pin diagrams of them are indicated below.

Diagram of 8253 pin based



Pin diagram functional

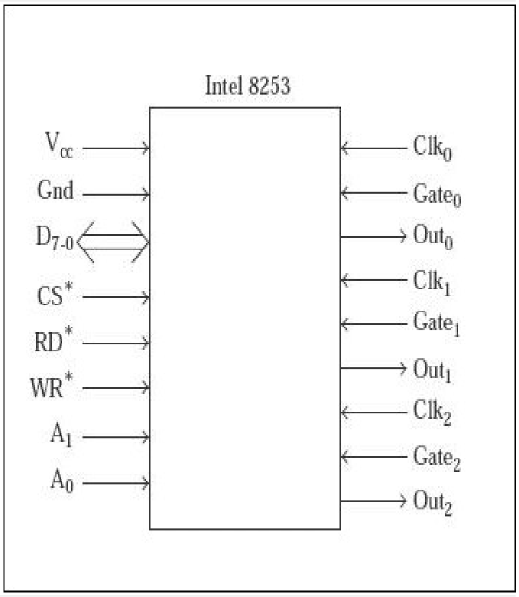


Fig.Pin diagram functional

Vcc and Gn	These are the Power supply and ground pins which 8253 uses +5V as power supply
D7-0	For the communication of the processor there are eight functional pins
RD*	This reads counter information it is active low pin
WR*	Writes control information
CS*	It selects the chip which is also active low pin
A1,A0	These are the address input pins.
CLK0	Clock input is provided for counter 0
CLK1	Clock input is provided for counter 1
CLK2	Clock input is provided for counter 2

Gate0	Controls counter function 0
Gate1	Controls function of counter 1
Gate2	Controls function of counter 2
Out0	Here counter 0 sends output
Out1	Here counter 1 sends output
Out2	Again counter 2 sends the output

Programming the 8253

According to the microprocessor point of view, the 8253 is designed and has some specialty port chip I/O. We don't use it for interfering the I/O devices. For performing the application of time it is used. 8253 has the addressed A1 and A0 input pins.

The counters have width of 16 bits. If they were 8-bits wide, the delay in time that would be generated is very small. The Least Significant Byte and the Most Significant Byte of a counter is selected by using the same address of the port.

The processor here writes to the control port to configure the working of the three timers. Actually, here the processor writes to the control port to configure the three counters working.

The control port contains conveys the following information to 8253.

Selects configuration counter.

Configures the selected counters to perform a particular operation.

To decide selected counter is in decimal or in hexadecimal.

The counters can be configured to work in any of the following six modes of operation.

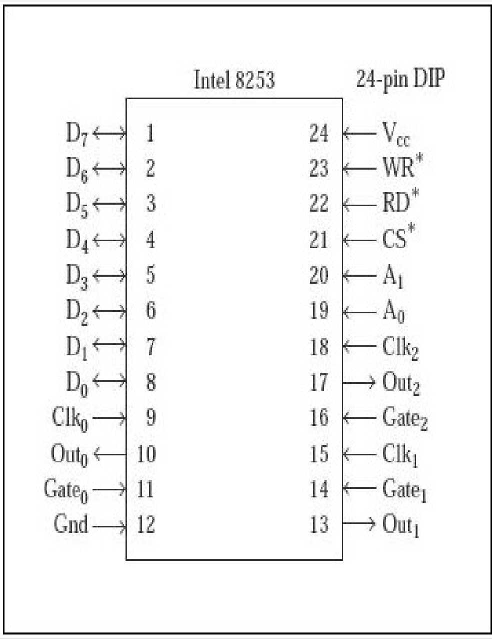
- Mode0 − It interrupts in the terminal count;
- Mode1 − re-triggerable input;
- Mode2 − It generates rate;
- Mode3 − It generates square wave;
- Mode4 − It generates software trigger;
- Mode5 − It generates hardware trigger.

For the discussion in this topic, the chip select circuit is assumed to be such that the port addresses are as follows.

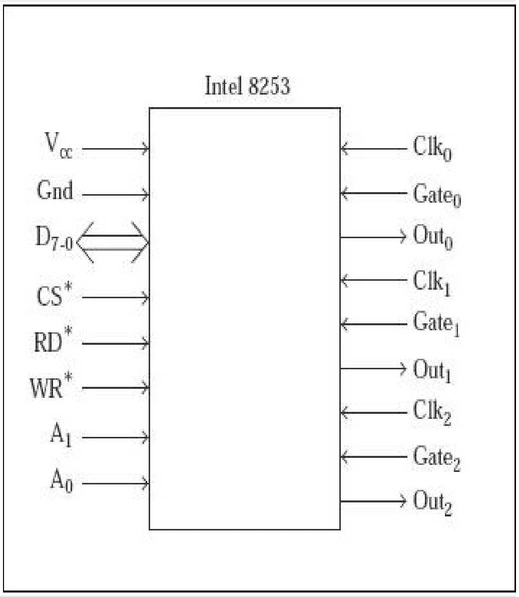
Address	Port
80H	It is the counter 0
81H	It is the counter 1
82H	It is the counter 2
83H	It is the Control port

As a DIP package Intel 8253 is a 24-pin programmable IC available. IC has three counters which work independently and whose width is of 16-bits. In addition, we have a control port to decide what is the mode of working of the three counters. The physical and functional pin diagrams of them are indicated below.

Diagram of 8253 pin based



Pin diagram functional



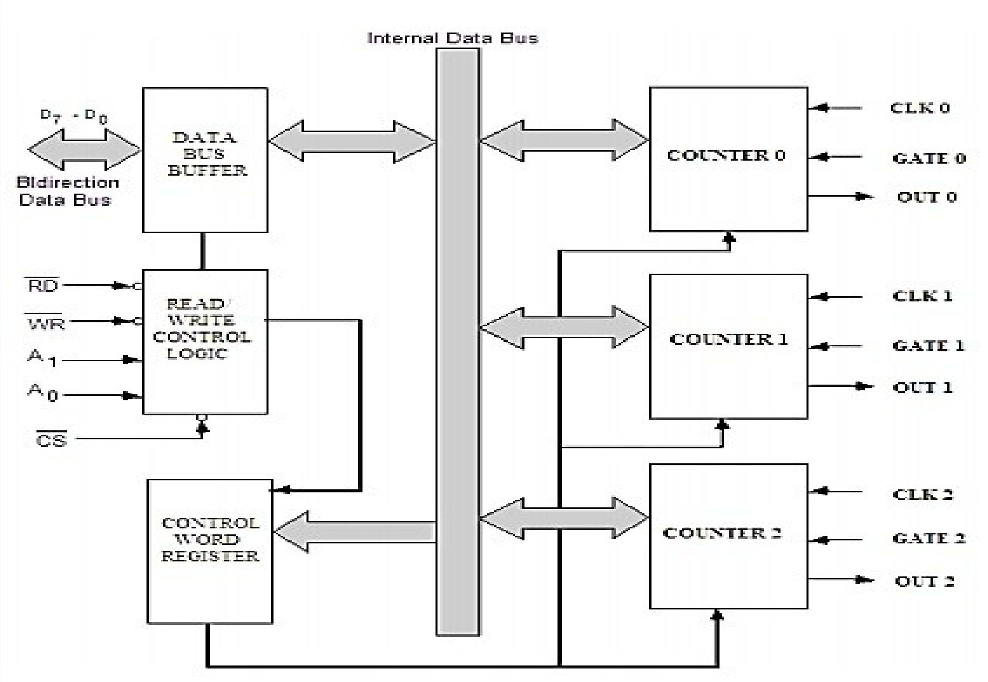
Interfacing 8253 (Timer IC) with 8085 Microprocessor

The Intel 8253 is programmable Interval Timers (PTIs) designed for microprocessors toper form timing and counting functions using three 16-bit registers. Each counter has 2 input pins, i.e. Clock &amp; Gate, and 1 pin for“OUT” output. To operate a counter, a 16-bit count is loaded in its register. On command, it begins to decrement the count until it reaches 0, then it generates a pulse that can be used to interrupt the CPU.

Features of 8253

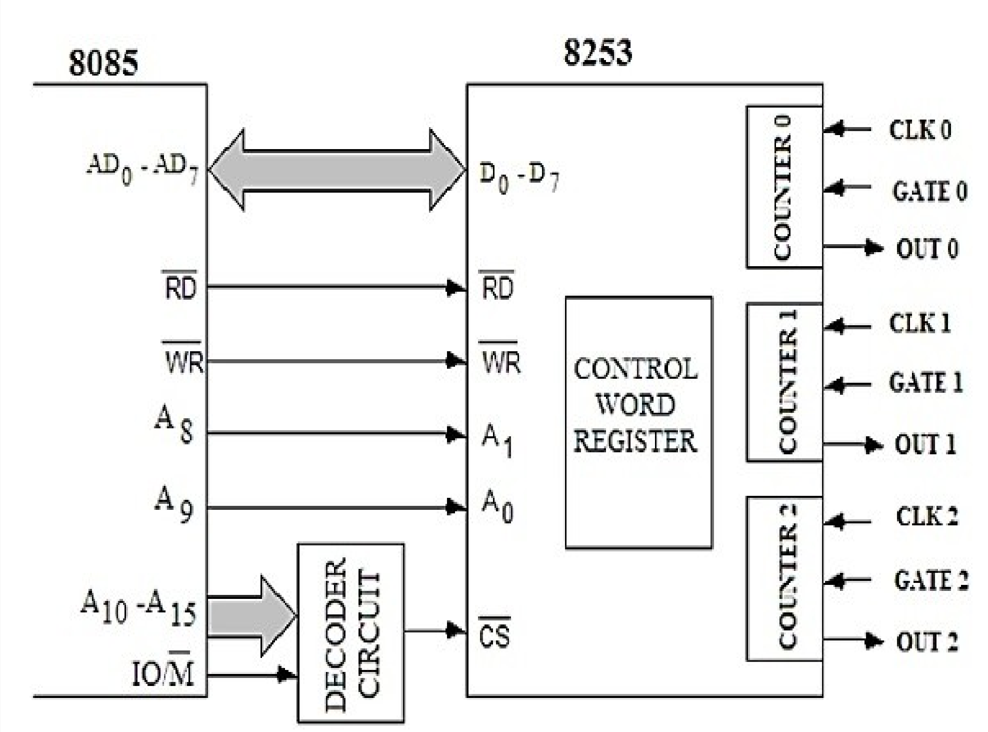
- •It has three independent 16-bit down counters.
- •It can handle inputs from DC to 10MHz.
- •These three counters can be programmed for either binary or BCD count.
- •It is compatible with almost all microprocessors.
- •8254has a powerful command called READ BACK command, which allows the user to check the count value, the programmed mode, the current mode, and the current status of the counter.

The block diagram of 8253

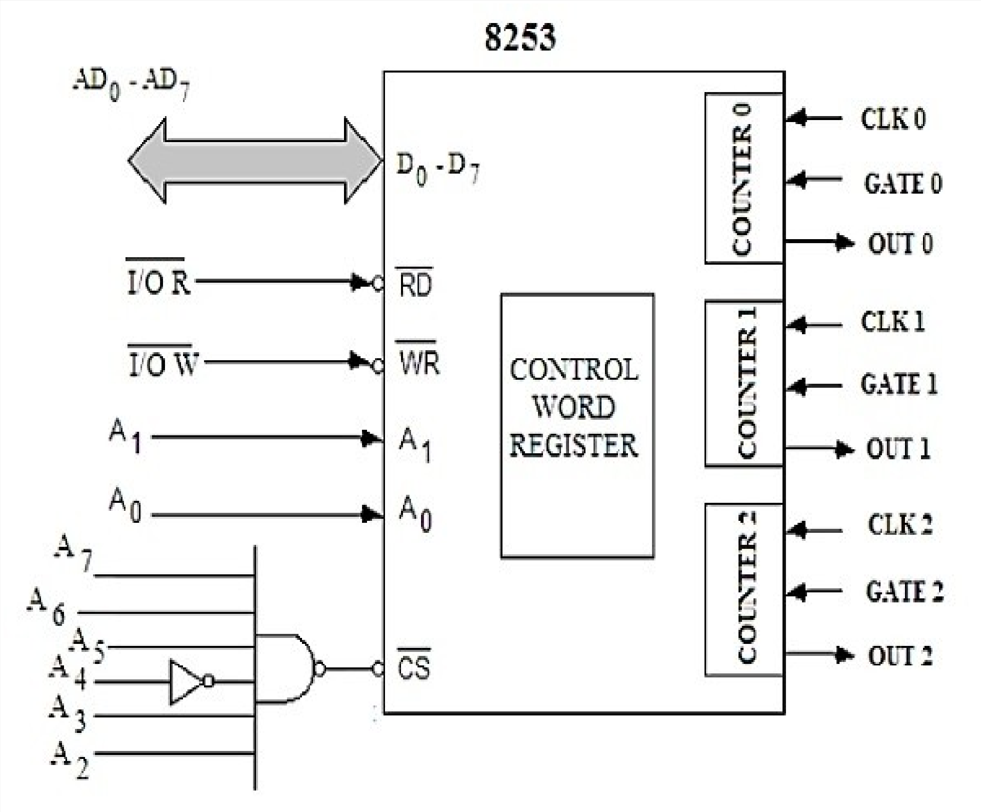


Interfacing 8253 with 8085

Now let us see how to interface this 8253 timer chip with the Intel 8085 microprocessor. From the following picture, we can see that the data bus D7-0 of 8085 is connected to the data pins D7 to D0 of 8253. So the higher order address bus is used as decoder input to select the chip and the A8 and A9 of 8085 are connected to the pin A1 and A0 respectively to select the counter.



In the next diagram, we can get the chip select logic of 8253. In that diagram, we can easily find that when A3-2 and A7-5 are at logic 0 and A4 at logic 1, then only the chip select CS pin of 8253 will be enabled.



This table is showing how the counter is being selected by using A1 and A0 pins of 8253.

CS								**HEX Address**	**Counter Selection**
A7	A6	A5	A4	A3	A2	A 1	A 0		
0	0	0	1	0	0	0	0	10H	Counter 0
0	0	0	1	0	0	0	1	11H	Counter 1
0	0	0	1	0	0	1	0	12H	Counter 2
0	0	0	1	0	0	1	1	13H	Control Word Register

By using the IN and OUT instruction the counter selection and Control Word Register (CWR) setup can be done. If the Accumulator is holding content to load CWR, then by using OUT 13H the CWR will be set. Similarly, by using IN instruction we can get the value of counter value, like IN 11H will get the value from counter 1 and so on.

So the following four steps are needed for counter operations:

- •Initialize 8253 chip
- •Load Control word register with Control Word value

- Load Lower Order count value
- Load Higher Order count value

Let us see a program to load counter 2 in mode 1 with a count value 500010 in mode 0. Also, read the count value on a fly.

At first, to initialize the8253, the Control word will be B2H

Counter 2		**Load LS and then MS**		**Mode 1 selection**			**0 for Binary**
1	0	1	1	0	0	1	0

Now the control word for latching operation for counter 2 is 80H.

Counter 2		**Latching Option**		**Don’t Care**			
1	0	0	0	0	0	0	0

We will load 500010 into the counter. The hexadecimal equivalent of 500010 is 1388H.

```
MVI A, B2H ;Load B2H as initialization byte for counter
OUT 13H ;Write Acc content CWR
MVI A, 88H ;Load LS byte of count value
OUT 12H ;Send to Counter 2
MVI A, 13H ;Load MS byte of count value
OUT 12H ;Send to Counter 2
MVI D, 00H ;clear the register D
L1: MVI A, 80H ;Set a with control word 80H of counter 2
OUT 13H ;Write Acc content CWR
IN 12H ;Read LS value of counter value
MOV B, A ;store LS value to B
IN 12H ;Read MS value of counter value
ORA B ;OR LS and MS to set Z flag
JNZ L1 ;If Z flag is not set, jump to Loop
HLT ;Halt the program
```