

# Intel 8253 - Programmable Interval Timer

---

The Intel 8253 and 8254 are Programmable Interval Timers (PTIs) designed for microprocessors to perform timing and counting functions using three 16-bit registers. Each counter has 2 input pins, i.e. Clock & 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.

## Difference between 8253 and 8254

The following table differentiates the features of 8253 and 8254 –

8253	8254
Its operating frequency is 0 - 2.6 MHz	Its operating frequency is 0 - 10 MHz
It uses N-MOS technology	It uses H-MOS technology
Read-Back command is not available	Read-Back command is available
Reads and writes of the same counter cannot be interleaved.	Reads and writes of the same counter can be interleaved.

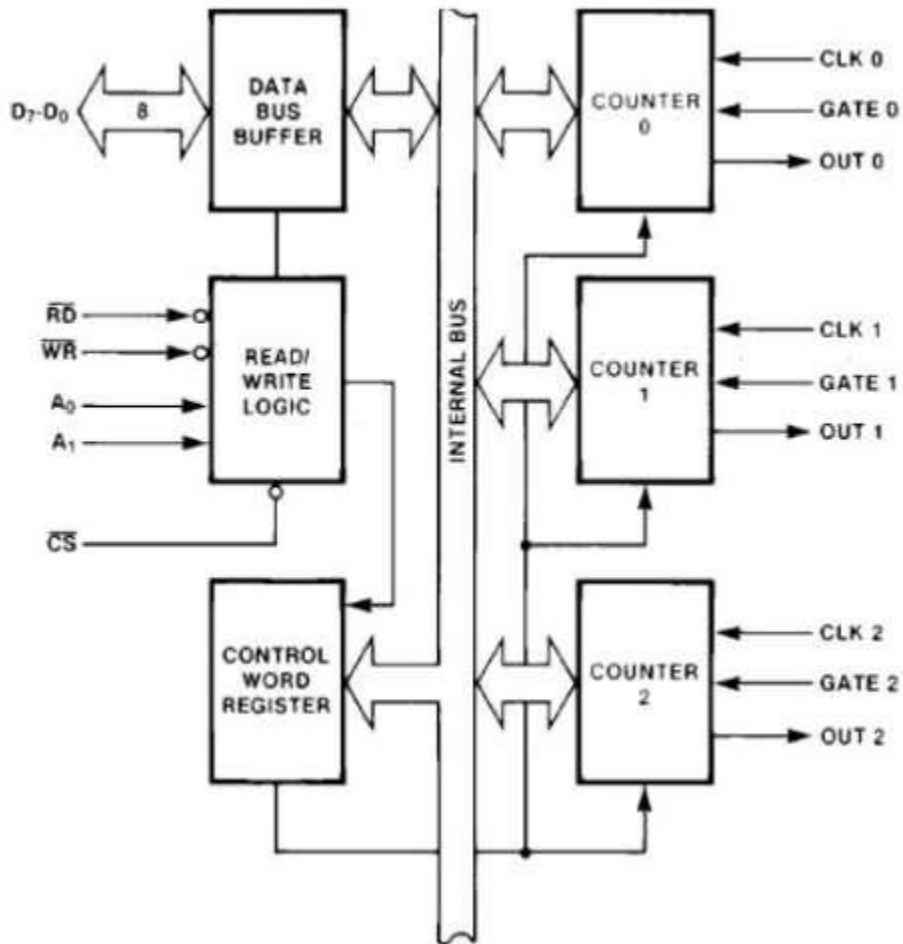
## Features of 8253 / 54

The most prominent features of 8253/54 are as follows –

- It has three independent 16-bit down counters.
- It can handle inputs from DC to 10 MHz.
- These three counters can be programmed for either binary or BCD count.
- It is compatible with almost all microprocessors.
- 8254 has 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.

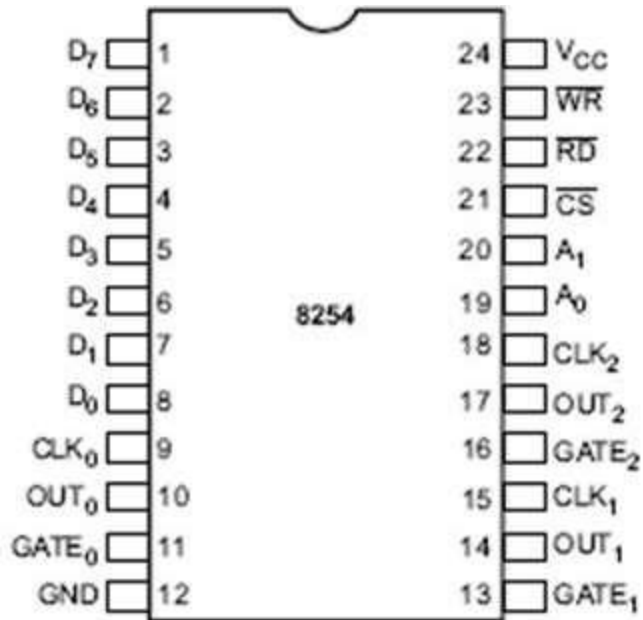
## 8254 Architecture

The architecture of 8254 looks as follows –



## 8254 Pin Description

Here is the pin diagram of 8254 –



In the above figure, there are three counters, a data bus buffer, Read/Write control logic, and a control register. Each counter has two input signals - CLOCK & GATE, and one output signal - OUT.

### Data Bus Buffer

It is a tri-state, bi-directional, 8-bit buffer, which is used to interface the 8253/54 to the system data bus. It has three basic functions –

- Programming the modes of 8253/54.
- Loading the count registers.
- Reading the count values.

### Read/Write Logic

It includes 5 signals, i.e. RD, WR, CS, and the address lines  $A_0$  &  $A_1$ . In the peripheral I/O mode, the RD and WR signals are connected to IOR and IOW, respectively. In the memory mapped I/O mode, these are connected to MEMR and MEMW.

Address lines  $A_0$  &  $A_1$  of the CPU are connected to lines  $A_0$  and  $A_1$  of the 8253/54, and CS is tied to a decoded address. The control word register and counters are selected according to the signals on lines  $A_0$  &  $A_1$ .

<b>A<sub>1</sub></b>	<b>A<sub>0</sub></b>	<b>Result</b>
0	0	Counter 0
0	1	Counter 1
1	0	Counter 2
1	1	Control Word Register
X	X	No Selection

### Control Word Register

This register is accessed when lines A<sub>0</sub> & A<sub>1</sub> are at logic 1. It is used to write a command word, which specifies the counter to be used, its mode, and either a read or write operation. Following table shows the result for various control inputs.

<b>A<sub>1</sub></b>	<b>A<sub>0</sub></b>	<b>RD</b>	<b>WR</b>	<b>CS</b>	<b>Result</b>
0	0	1	0	0	Write Counter 0
0	1	1	0	0	Write Counter 1
1	0	1	0	0	Write Counter 2
1	1	1	0	0	Write Control Word
0	0	0	1	0	Read Counter 0
0	1	0	1	0	Read Counter 1

1	0	0	1	0	Read Counter 2
1	1	0	1	0	No operation
X	X	1	1	0	No operation
X	X	X	X	1	No operation

## Counters

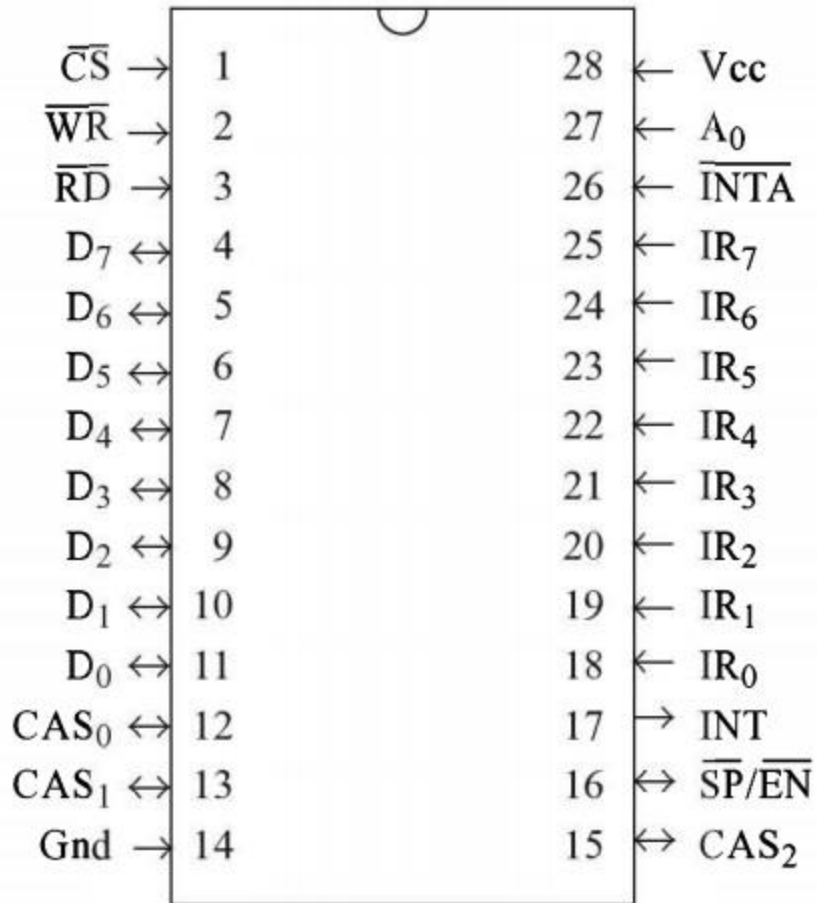
Each counter consists of a single, 16 bit-down counter, which can be operated in either binary or BCD. Its input and output is configured by the selection of modes stored in the control word register. The programmer can read the contents of any of the three counters without disturbing the actual count in process.

## 8259 –PRIORITY INTERRUPT CONTROLLER

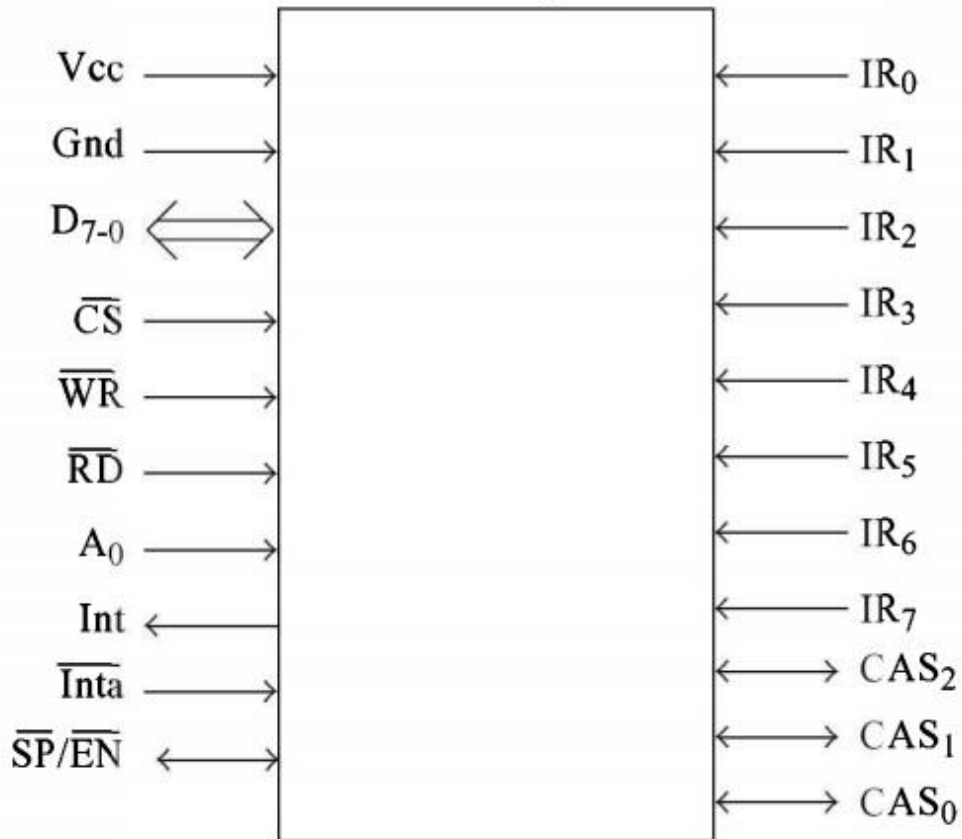
The 8259 is known as the Programmable Interrupt Controller (PIC) microprocessor. In 8085 and 8086 there are five hardware interrupts and two hardware interrupts respectively. Bu adding 8259, we can increase the interrupt handling capability. This chip combines the multi-interrupt input source to single 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.

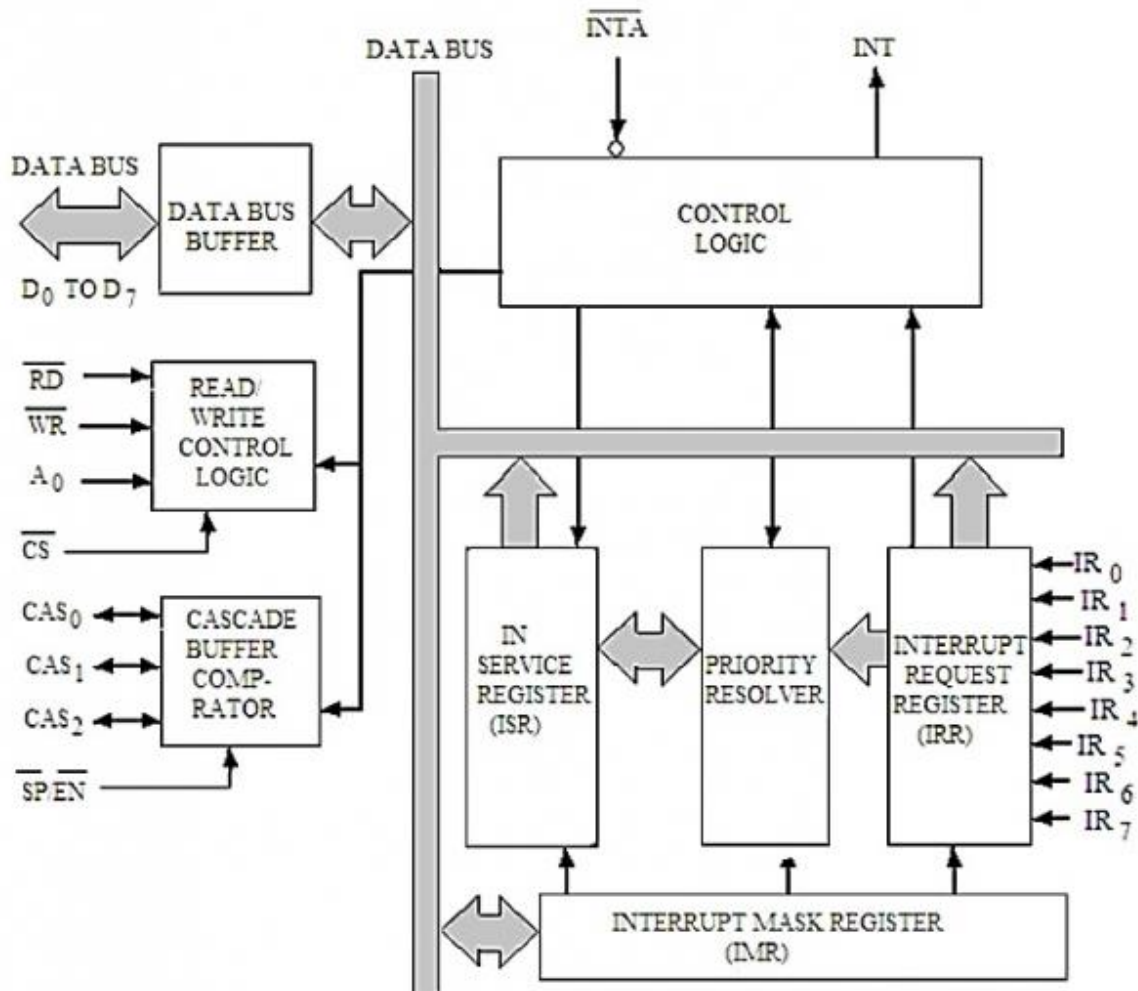
The pin level diagram and functional pin diagram is like below -



### 8259 PIC 28-pin DIP



The block diagram is like below -



Block	Description
<b>Data Bus Buffer</b>	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.
<b>R/W Control Logic</b>	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.

<b>Block</b>	<b>Description</b>
<b>Control Logic</b>	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.
<b>Interrupt Request Register</b>	It stores all interrupt level that are requesting for interrupt service.
<b>Interrupt Service Register</b>	It stores interrupt level that are currently being execute.
<b>Interrupt Mask Register</b>	It stores interrupt level that will be masked, by storing the masking bits of interrupt level.
<b>Priority Resolver</b>	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.
<b>Cascade Buffer</b>	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.

# 8279 - Programmable Keyboard

---

## Advertisements

[Previous Page](#)

[Next Page](#)

8279 programmable keyboard/display controller is designed by Intel that interfaces a keyboard with the CPU. The keyboard first scans the keyboard and identifies if any key has been pressed. It then sends their relative response of the pressed key to the CPU and vice-a-versa.

### How Many Ways the Keyboard is Interfaced with the CPU?

The Keyboard can be interfaced either in the interrupt or the polled mode. In the **Interrupt mode**, the processor is requested service only if any key is pressed, otherwise the CPU will continue with its main task.

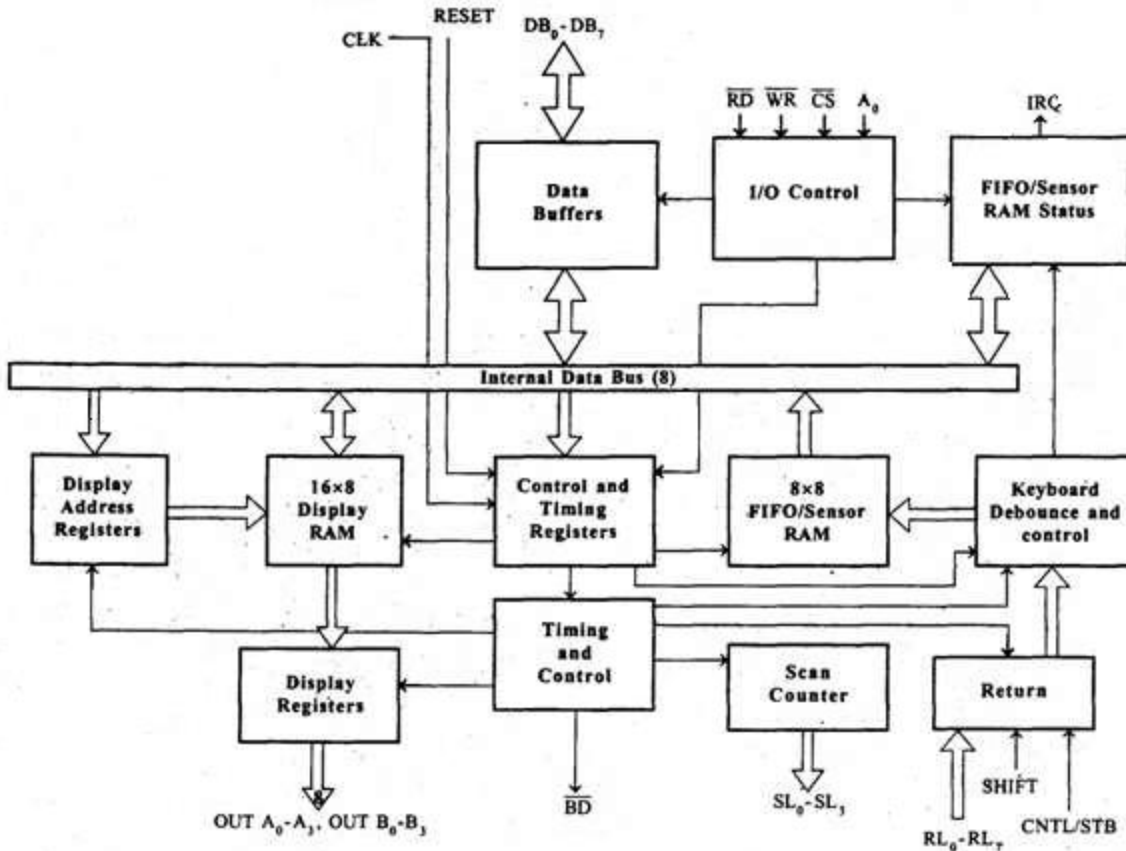
In the **Polled mode**, the CPU periodically reads an internal flag of 8279 to check whether any key is pressed or not with key pressure.

### How Does 8279 Keyboard Work?

The keyboard consists of maximum 64 keys, which are interfaced with the CPU by using the key-codes. These key-codes are de-bounced and stored in an 8-byte FIFORAM, which can be accessed by the CPU. If more than 8 characters are entered in the FIFO, then it means more than eight keys are pressed at a time. This is when the overrun status is set.

If a FIFO contains a valid key entry, then the CPU is interrupted in an interrupt mode else the CPU checks the status in polling to read the entry. Once the CPU reads a key entry, then FIFO is updated, and the key entry is pushed out of the FIFO to generate space for new entries.

### Architecture and Description



## I/O Control and Data Buffer

This unit controls the flow of data through the microprocessor. It is enabled only when D is low. Its data buffer interfaces the external bus of the system with the internal bus of the microprocessor. The pins A0, RD, and WR are used for command, status or data read/write operations.

## Control and Timing Register and Timing Control

This unit contains registers to store the keyboard, display modes, and other operations as programmed by the CPU. The timing and control unit handles the timings for the operation of the circuit.

## Scan Counter

It has two modes i.e. **Encoded mode** and Decoded mode. In the encoded mode, the counter provides the binary count that is to be externally decoded to provide the scan lines for the keyboard and display.

In the **decoded scan mode**, the counter internally decodes the least significant 2 bits and provides a decoded 1 out of 4 scan on SL<sub>0</sub>-SL<sub>3</sub>.

## Return Buffers, Keyboard Debounce, and Control

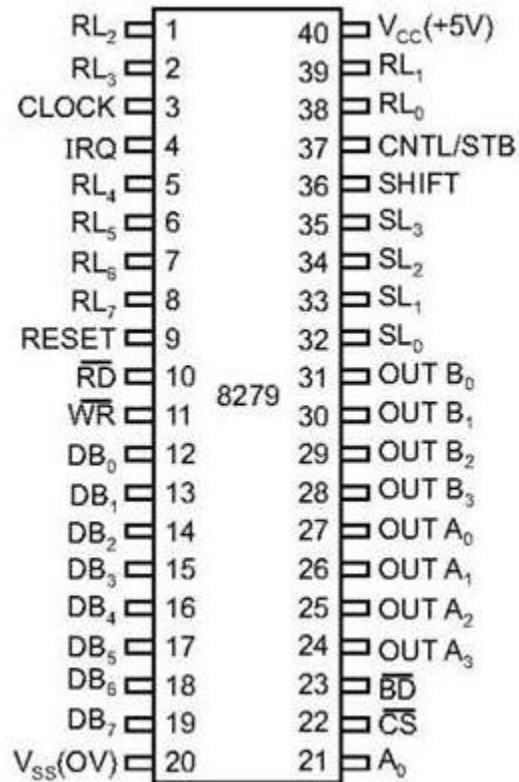
This unit first scans the key closure row-wise, if found then the keyboard debounce unit debounces the key entry. In case, the same key is detected, then the code of that key is directly transferred to the sensor RAM along with SHIFT & CONTROL key status.

## Display Address Registers and Display RAM

This unit consists of display address registers which holds the addresses of the word currently read/written by the CPU to/from the display RAM.

## 8279 – Pin Description

The following figure shows the pin diagram of 8279 –



## Data Bus Lines, DB<sub>0</sub> - DB<sub>7</sub>

These are 8 bidirectional data bus lines used to transfer the data to/from the CPU.

## CLK

The clock input is used to generate internal timings required by the microprocessor.

## RESET

As the name suggests this pin is used to reset the microprocessor.

## CS Chip Select

When this pin is set to low, it allows read/write operations, else this pin should be set to high.

## A<sub>0</sub>

This pin indicates the transfer of command/status information. When it is low, it indicates the transfer of data.

## RD, WR

This Read/Write pin enables the data buffer to send/receive data over the data bus.

## IRQ

This interrupt output line goes high when there is data in the FIFO sensor RAM. The interrupt line goes low with each FIFO RAM read operation. However, if the FIFO RAM further contains any key-code entry to be read by the CPU, this pin again goes high to generate an interrupt to the CPU.

## V<sub>ss</sub>, V<sub>cc</sub>

These are the ground and power supply lines of the microprocessor.

## SL<sub>0</sub> – SL<sub>3</sub>

These are the scan lines used to scan the keyboard matrix and display the digits. These lines can be programmed as encoded or decoded, using the mode control register.

## RL<sub>0</sub> – RL<sub>7</sub>

These are the Return Lines which are connected to one terminal of keys, while the other terminal of the keys is connected to the decoded scan lines. These lines are set to 0 when any key is pressed.

## SHIFT

The Shift input line status is stored along with every key code in FIFO in the scanned keyboard mode. Till it is pulled low with a key closure, it is pulled up internally to keep it high

### CNTL/STB - CONTROL/STROBED I/P Mode

In the keyboard mode, this line is used as a control input and stored in FIFO on a key closure. The line is a strobe line that enters the data into FIFO RAM, in the strobed input mode. It has an internal pull up. The line is pulled down with a key closure.

### BD

It stands for blank display. It is used to blank the display during digit switching.

### OUTA<sub>0</sub> – OUTA<sub>3</sub> and OUTB<sub>0</sub> – OUTB<sub>3</sub>

These are the output ports for two 16x4 or one 16x8 internal display refresh registers. The data from these lines is synchronized with the scan lines to scan the display and the keyboard

## Operational Modes of 8279

There are two modes of operation on 8279 – **Input Mode** and **Output Mode**.

### Input Mode

This mode deals with the input given by the keyboard and this mode is further classified into 3 modes.

- **Scanned Keyboard Mode** – In this mode, the key matrix can be interfaced using either encoded or decoded scans. In the encoded scan, an 8x8 keyboard or in the decoded scan, a 4x8 keyboard can be interfaced. The code of key pressed with SHIFT and CONTROL status is stored into the FIFO RAM.
- **Scanned Sensor Matrix** – In this mode, a sensor array can be interfaced with the processor using either encoder or decoder scans. In the encoder scan, 8x8 sensor matrix or with decoder scan 4x8 sensor matrix can be interfaced.

- **Strobed Input** – In this mode, when the control line is set to 0, the data on the return lines is stored in the FIFO byte by byte.

## Output Mode

This mode deals with display-related operations. This mode is further classified into two output modes.

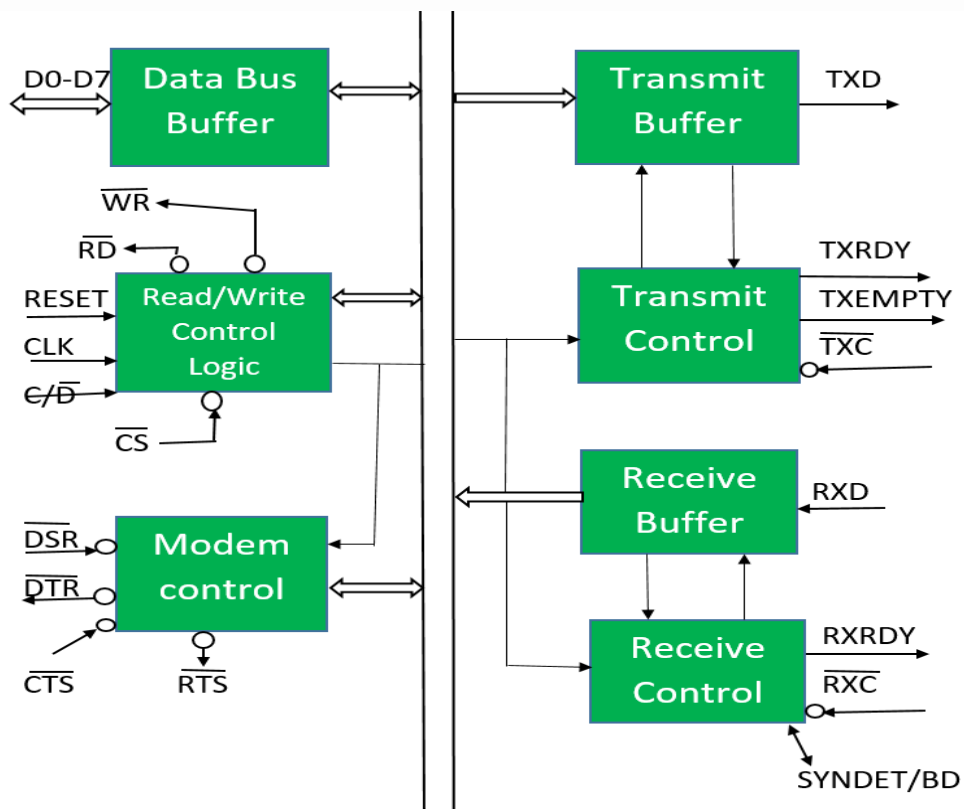
- **Display Scan** – This mode allows 8/16 character multiplexed displays to be organized as dual 4-bit/single 8-bit display units.
- **Display Entry** – This mode allows the data to be entered for display either from the right side/left side.

## Microprocessor | 8251 USART

8251 universal synchronous asynchronous receiver transmitter (USART) acts as a mediator between microprocessor and peripheral to transmit serial data into parallel form and vice versa.

1. It takes data serially from peripheral (outside devices) and converts into parallel data.
2. After converting the data into parallel form, it transmits it to the CPU.
3. Similarly, it receives parallel data from microprocessor and converts it into serial form.
4. After converting data into serial form, it transmits it to outside device (peripheral).

## Block Diagram of 8251 USART –



It contains the following blocks:

### 1. Data bus buffer –

This block helps in interfacing the internal data bus of 8251 to the system data bus. The data transmission is possible between 8251 and CPU by the data bus buffer block.

### 2. Read/Write control logic –

It is a control block for overall device. It controls the overall working by selecting the operation to be done. The operation selection depends upon input signals as:

$\overline{CS}$	$C/\overline{D}$	$\overline{RD}$	$\overline{WR}$	Operation
1	X	X	X	Invalid
0	0	0	1	data CPU < ----- 8251
0	0	1	0	data CPU ----- > 8251
0	1	0	1	Status word CPU < -----8251
0	1	1	0	Control word CPU----- > 8251

In this way, this unit selects one of the three registers- data buffer register, control register, status register.

### 3. Modem control (modulator/demodulator) –

A device converts analog signals to digital signals and vice-versa and helps the computers to communicate over telephone lines or cable wires. The following are active-low pins of Modem.

- **DSR:** Data Set Ready signal is an input signal.
- **DTR:** Data terminal Ready is an output signal.
- **CTS:** It is an input signal which controls the data transmit circuit.
- **RTS:** It is an output signal which is used to set the status RTS.

### 4. Transmit buffer –

This block is used for parallel to serial converter that receives a parallel byte for conversion into serial signal and further transmission onto the common channel.

- **TXD:** It is an output signal, if its value is one, means transmitter will transmit the data.

### 5. Transmit control –

This block is used to control the data transmission with the help of following pins:

- **TXRDY:** It means transmitter is ready to transmit data character.
  - **TXEMPTY:** An output signal which indicates that TXEMPTY pin has transmitted all the data characters and transmitter is empty now.
  - **TXC:** An active-low input pin which controls the data transmission rate of transmitted data.
6. **Receive buffer –**  
This block acts as a buffer for the received data.
- **RXD:** An input signal which receives the data.
7. **Receive control –**  
This block controls the receiving data.
- **RXRDY:** An input signal indicates that it is ready to receive the data.
  - **RXC:** An active-low output signal which controls the data transmission rate of received data.
  - **SYNDET/BD:** An input or output terminal. External synchronous mode-input terminal and asynchronous mode-output terminal.

## Microprocessor - 8237/8257 DMA Controller

---

DMA stands for Direct Memory Access. It is designed by Intel to transfer data at the fastest rate. It allows the device to transfer the data directly to/from memory without any interference of the CPU.

Using a DMA controller, the device requests the CPU to hold its data, address and control bus, so the device is free to transfer data directly to/from the memory. The DMA data transfer is initiated only after receiving HLDA signal from the CPU.

### DMA Operations

Following is the sequence of operations performed by a DMA –

- Initially, when any device has to send data between the device and the memory, the device has to send DMA request (DRQ) to DMA controller.
- The DMA controller sends Hold request (HRQ) to the CPU and waits for the CPU to assert the HLDA.
- Then the microprocessor tri-states all the data bus, address bus, and control bus. The CPU leaves the control over bus and acknowledges the HOLD request through HLDA signal.
- Now the CPU is in HOLD state and the DMA controller has to manage the operations over buses between the CPU, memory, and I/O devices.

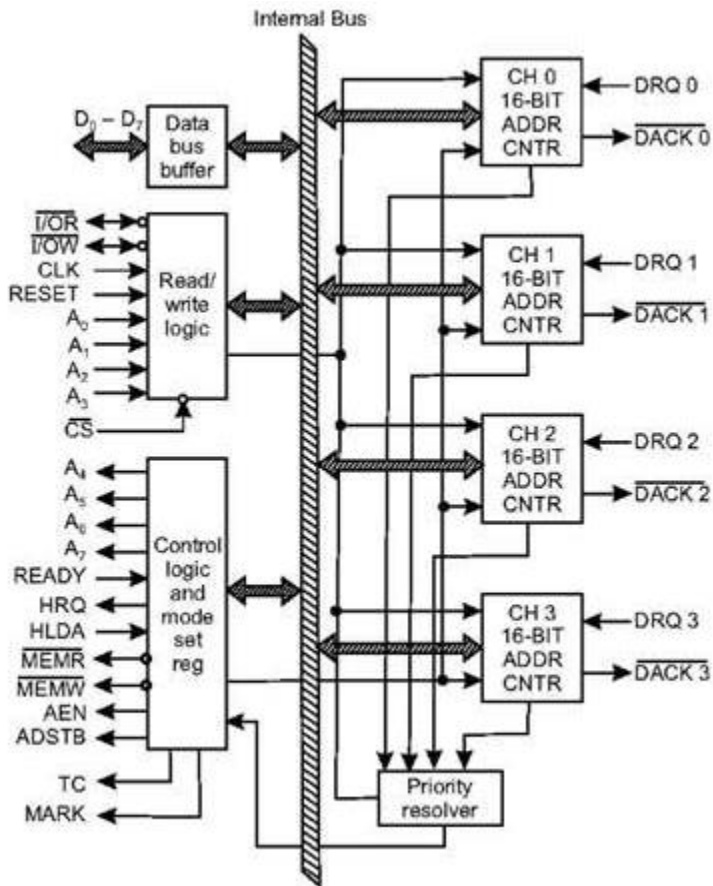
## Features of 8257

Here is a list of some of the prominent features of 8257 –

- It has four channels which can be used over four I/O devices.
- Each channel has 16-bit address and 14-bit counter.
- Each channel can transfer data up to 64kb.
- Each channel can be programmed independently.
- Each channel can perform read transfer, write transfer and verify transfer operations.
- It generates MARK signal to the peripheral device that 128 bytes have been transferred.
- It requires a single phase clock.
- Its frequency ranges from 250Hz to 3MHz.
- It operates in 2 modes, i.e., **Master mode** and **Slave mode**.

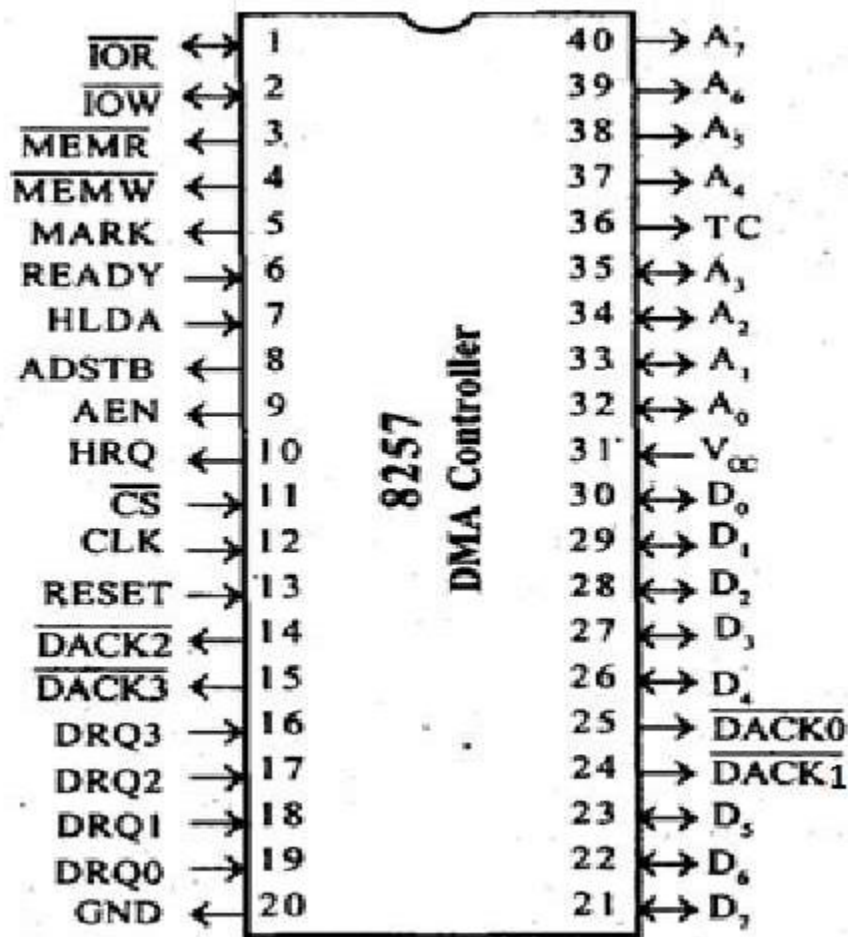
## 8257 Architecture

The following image shows the architecture of 8257 –



## 8257 Pin Description

The following image shows the pin diagram of a 8257 DMA controller –



### DRQ<sub>0</sub>–DRQ<sub>3</sub>

These are the four individual channel DMA request inputs, which are used by the peripheral devices for using DMA services. When the fixed priority mode is selected, then DRQ<sub>0</sub> has the highest priority and DRQ<sub>3</sub> has the lowest priority among them.

### DACK<sub>0</sub> – DACK<sub>3</sub>

These are the active-low DMA acknowledge lines, which updates the requesting peripheral about the status of their request by the CPU. These lines can also act as strobe lines for the requesting devices.

## $D_0 - D_7$

These are bidirectional, data lines which are used to interface the system bus with the internal data bus of DMA controller. In the Slave mode, it carries command words to 8257 and status word from 8257. In the master mode, these lines are used to send higher byte of the generated address to the latch. This address is further latched using ADSTB signal.

## IOR

It is an active-low bidirectional tri-state input line, which is used by the CPU to read internal registers of 8257 in the Slave mode. In the master mode, it is used to read data from the peripheral devices during a memory write cycle.

## IOW

It is an active low bi-direction tri-state line, which is used to load the contents of the data bus to the 8-bit mode register or upper/lower byte of a 16-bit DMA address register or terminal count register. In the master mode, it is used to load the data to the peripheral devices during DMA memory read cycle.

## CLK

It is a clock frequency signal which is required for the internal operation of 8257.

## RESET

This signal is used to RESET the DMA controller by disabling all the DMA channels.

## $A_0 - A_3$

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

## CS

It is an active-low chip select line. In the Slave mode, it enables the read/write operations to/from 8257. In the master mode, it disables the read/write operations to/from 8257.

## A<sub>4</sub> - A<sub>7</sub>

These are the higher nibble of the lower byte address generated by DMA in the master mode.

## READY

It is an active-high asynchronous input signal, which makes DMA ready by inserting wait states.

## HRQ

This signal is used to receive the hold request signal from the output device. In the slave mode, it is connected with a DRQ input line 8257. In Master mode, it is connected with HOLD input of the CPU.

## HLDA

It is the hold acknowledgement signal which indicates the DMA controller that the bus has been granted to the requesting peripheral by the CPU when it is set to 1.

## MEMR

It is the low memory read signal, which is used to read the data from the addressed memory locations during DMA read cycles.

## MEMW

It is the active-low three state signal which is used to write the data to the addressed memory location during DMA write operation.

## ADST

This signal is used to convert the higher byte of the memory address generated by the DMA controller into the latches.

## AEN

This signal is used to disable the address bus/data bus.

## TC

It stands for 'Terminal Count', which indicates the present DMA cycle to the present peripheral devices.

## MARK

The mark will be activated after each 128 cycles or integral multiples of it from the beginning. It indicates the current DMA cycle is the 128th cycle since the previous MARK output to the selected peripheral device.

## $V_{cc}$

It is the power signal which is required for the operation of the circuit.