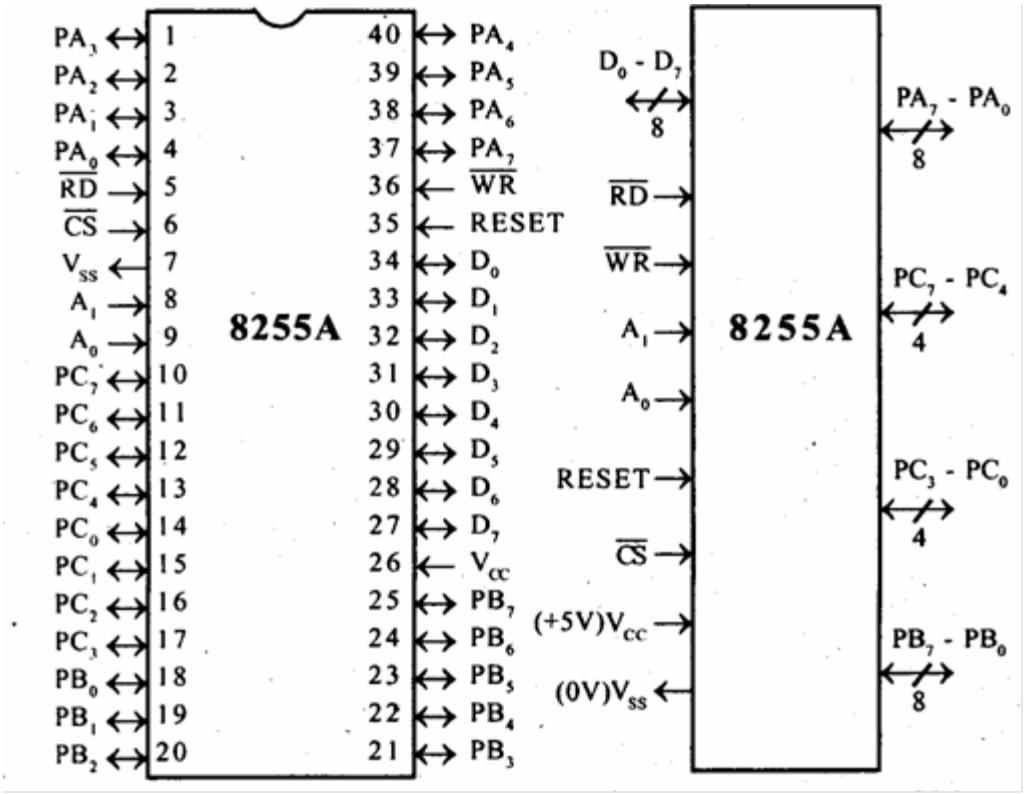
Unit –III Notes

1. **Explain in detail about Programmable Peripheral Interface - Intel 8255**

Pins, Signals and internal block diagram of 8255:

- It has 40 pins and requires a single +5V supply.



- The INTEL 8255 is a device used to parallel data transfer between processor and slow peripheral devices like ADC, DAC, keyboard, 7-segment display, LCD, etc.
- The 8255 has three ports: Port-A, Port-B and Port-C.
- Port-A can be programmed to work in any one of the three operating modes mode-0, mode-1 and mode-2 as input or output port.
- Port-B can be programmed to work either in mode-0 or mode-1 as input or output port.
- Port-C (8-pins) has different assignments depending on the mode of port-A and port-B.
- If port-A and B are programmed in mode-0, then the port-C can perform any one of the following functions.
- As 8-bit parallel port in mode-0 for input or output.
- As two numbers of 4-bit parallel ports in mode-0 for input or output.
- The individual pins of port-C can be set or reset for various control applications.
- If port-A is programmed in mode- 1/mode-2 and port-B is programmed in mode-1 then some of the pins of port-C are used for handshake signals and the remaining pins can be used as input/ output lines or individually set/reset for control applications.
- The read/write control logic requires six control signals. These signals are given below.

1. RD (low): This control signal enables the read operation. When this signal is low, the microprocessor reads data from a selected I/O port of the 8255A.

2. WR (low): This control signal enables the write operation. When this signal goes low, the

3. RESET: This is an active high signal. It clears the control register and set all ports in the input mode.

4. CS (low), A0 and A1: These are device select signals. They are,

Interfacing of 8255 with 8085 processor:

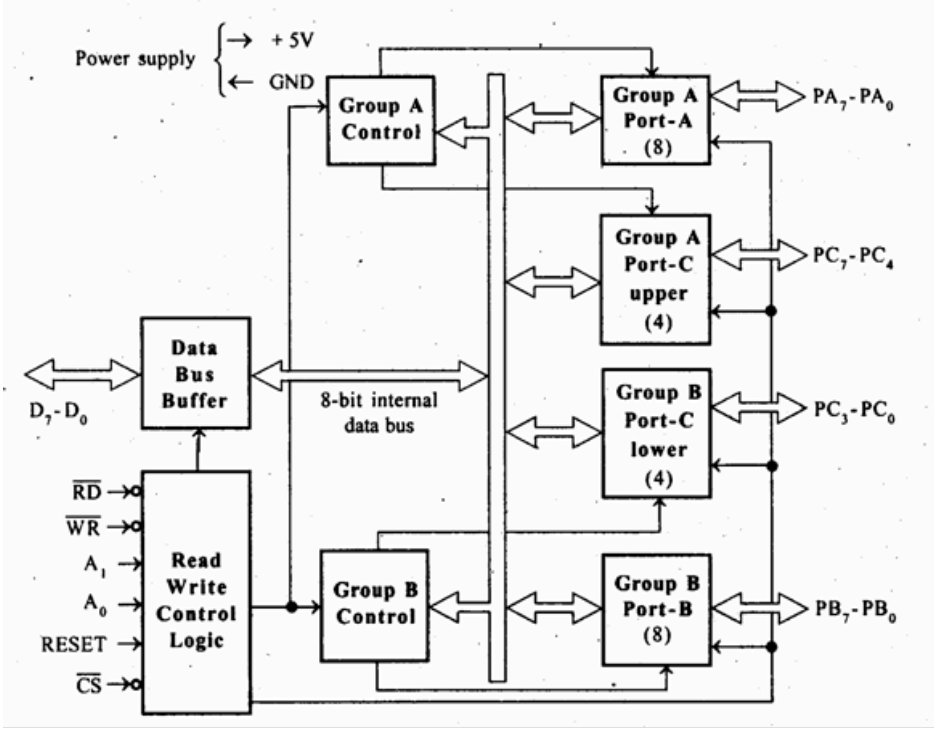
- A simple schematic for interfacing the 8255 with 8085 processor is shown in fig.

PIN DESCRIPTION - INTEL 8255

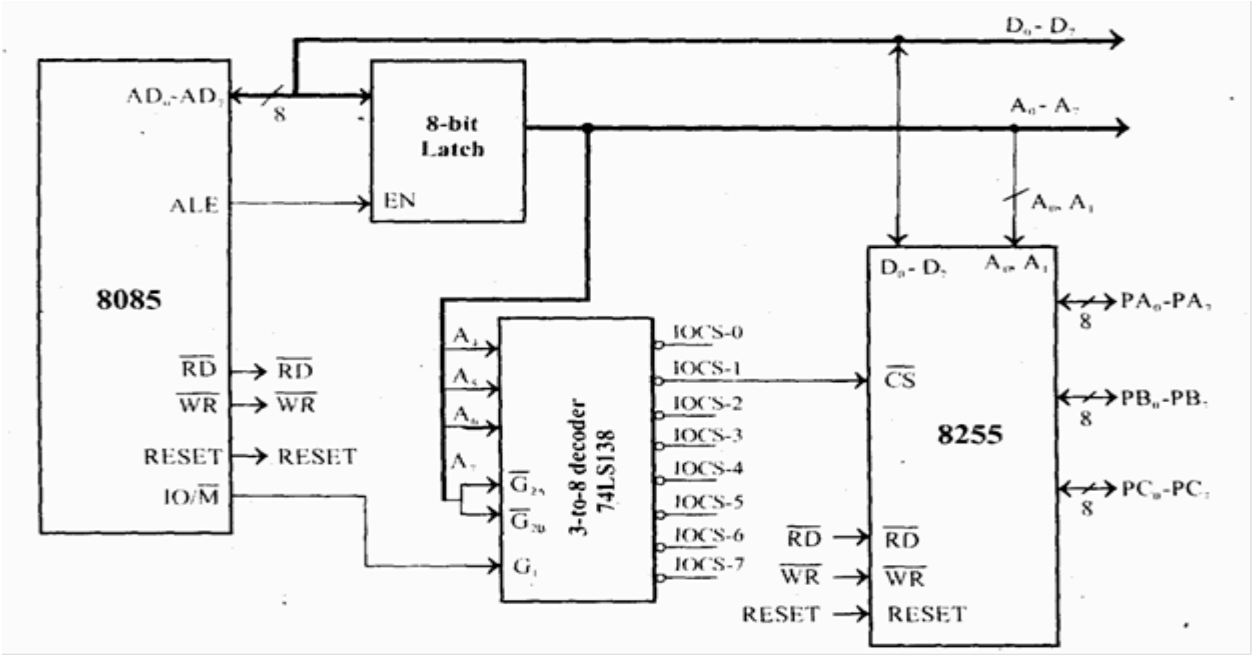
Pin	Description
D_0 - D_7	Data lines
RESET	Reset input
$\overline{CS}$	Chip select
$\overline{RD}$	Read control
$\overline{WR}$	Write control
A_0, A_1	Internal address
PA_7 - PA_0	Port-A pins
PB_7 - PB_0	Port-B pins
PC_7 - PC_0	Port-C pins
V_{CC}	+5V
V_{SS}	0V (GND)

Block diagram of 8255:

- The internal block diagram of 8255 is shown in fig:



Internal Devices	A_1	A_0
Port A	0	0
Port B	0	1
Port C	1	0
Control Register	1	1



- The 8255 can be either memory mapped or I/O mapped in the system. In the schematic shown 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.
- 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 connected to A1 of 8255 to provide the internal addresses.
- The data lines D0-D7 are connected to D0-D7 of the processor to achieve parallel data transfer.
- 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.*

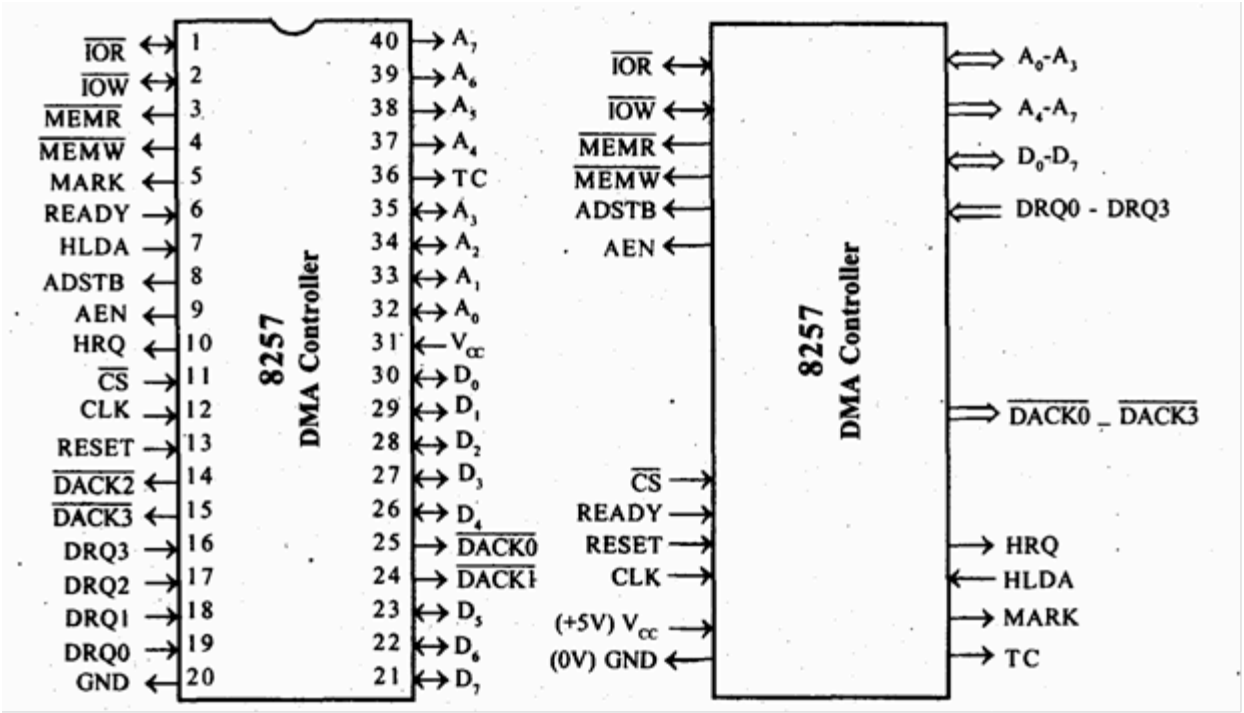
2.Explain in detail about Programmable DMA Controller - Intel 8257

- It is a device to transfer the data directly between IO device and memory without through the CPU. So it performs a high-speed data transfer between memory and I/O device.

The features of 8257 is,

- The 8257 has four channels and so it can be used to provide DMA to four I/O devices
- Each channel can be independently programmable to transfer up to 64kb of data by DMA.
- Each channel can be independently perform read transfer, write transfer and verify transfer.

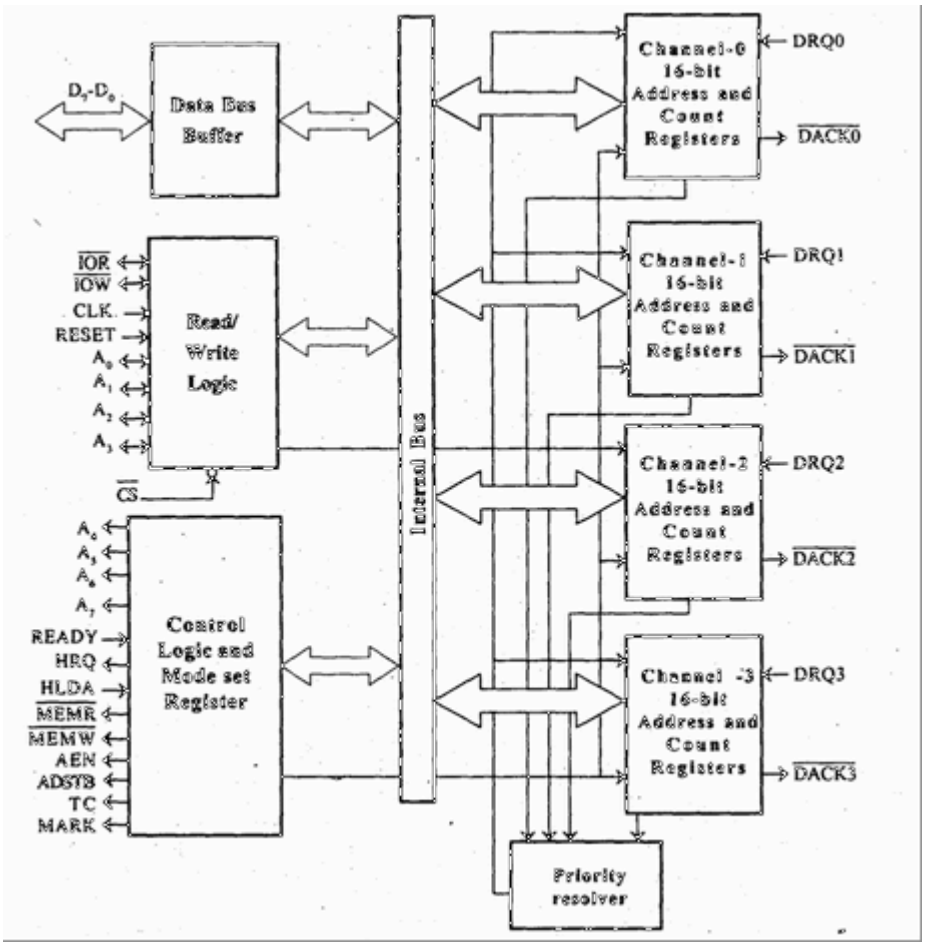
It is a 40 pin IC and the pin diagram is,



Functional Block Diagram of 8257:

- The functional blocks of 8257 are data bus buffer, read/write logic, control logic, priority resolver and four numbers of DMA channels.

  The functional block diagram of 8257 is shown in fig.



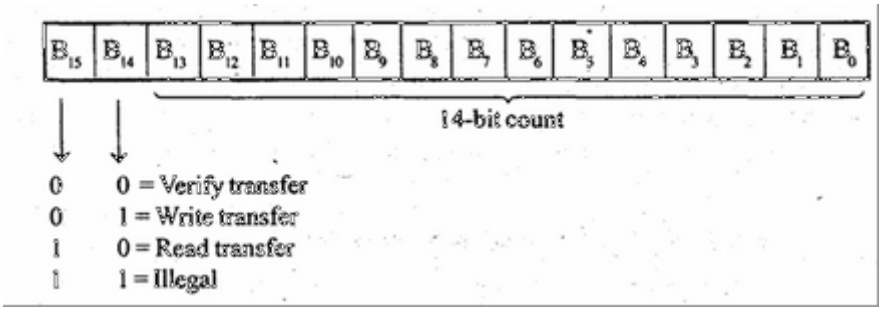
OPERATING MODES OF INTEL 8257

- Each channel of 8257 Block diagram has two programmable 16-bit registers named as address register and count register.

- Address register is used to store the starting address of memory location for DMA data transfer.

- The address in the address register is automatically incremented after every read/write/verify transfer.

- The count register is used to count the number of byte or word transferred by DMA. The format of count register is,



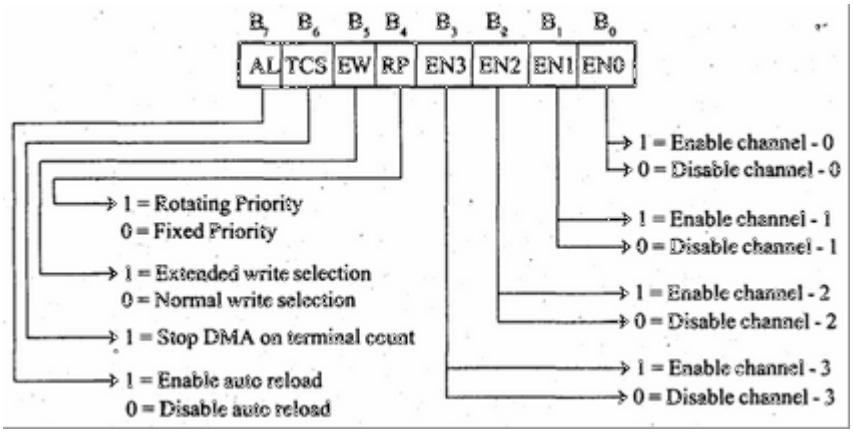
- 14-bits B0-B13 is used to count value and a 2-bits is used for indicate the type of DMA transfer (Read/Write/Veri1 transfer).

- In read transfer the data is transferred from memory to I/O device.

- In write transfer the data is transferred from I/O device to memory.

- Verification operations generate the DMA addresses without generating the DMA memory and I/O control signals.

- The 8257 has two eight bit registers called mode set register and status register. The format of mode set register is,



- The use of mode set register is,

    1. Enable/disable a channel.
    2. Fixed/rotating priority
    3. Stop DMA on terminal count.
    4.Extended/normal write time.
    5. Auto reloading of channel-2.

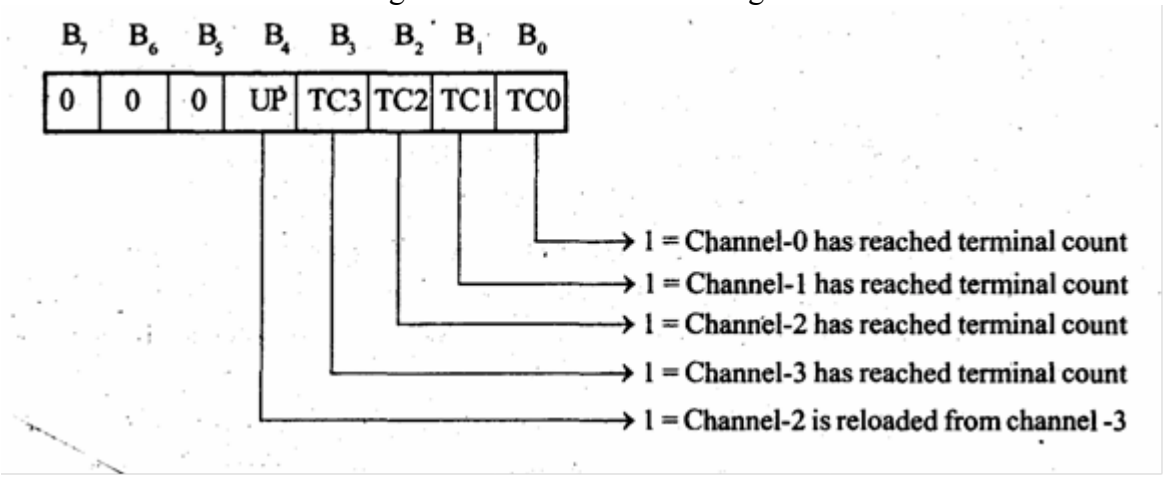
- The bits B0, B1, B2, and B3 of mode set register are used to enable/disable channel -0, 1, 2 and 3 respectively. A one in these bit position will enable a particular channel and a zero will disable it.

- If the bit B4 is set to one, then the channels will have rotating priority and if it zero then the channels wilt have fixed priority.

    1. In rotating priority after servicing a channel its priority is made as lowest.
    2. In fixed priority the channel-0 has highest priority and channel-2 has lowest priority.

- If the bit B5 is set to one, then the timing of low write signals (MEMW and IOW) will be extended

- If the bit B6 is set to one then the DMA operation is stopped at the terminal count.
- The bit B7 is used to select the auto load feature for DMA channel-2.
- When bit B7 is set to one, then the content of channel-3 count and address registers are loaded in channel-2 count and address registers respectively whenever the channel-2 reaches terminal count. When this mode is activated the number of channels available for DMA reduces from four to three.
- The format of status register of 8257 is shown in fig.

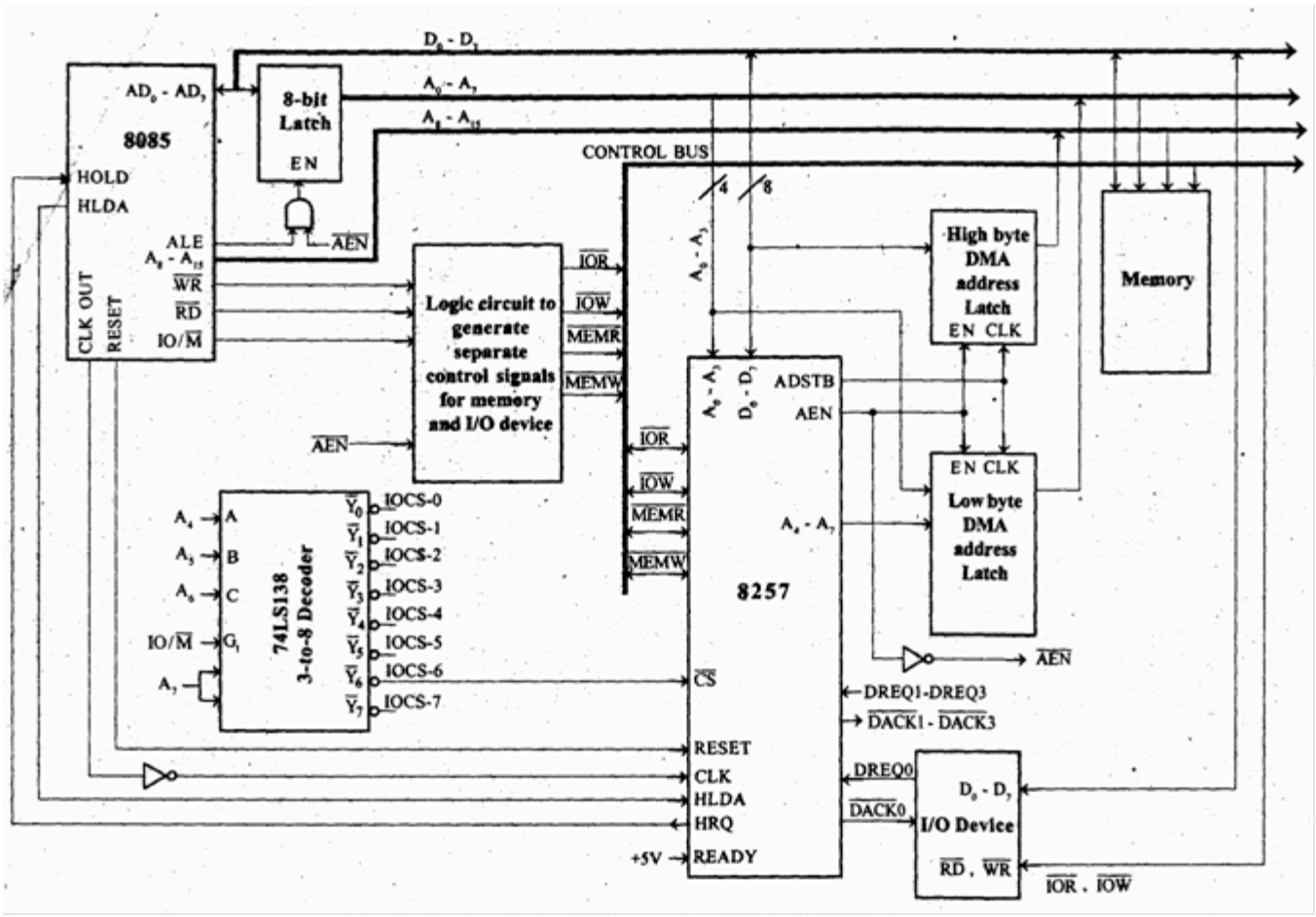


- The bit B0, B1, B2, and B3 of status register indicates the terminal count status of channel-0, 1,2 and 3 respectively. A one in these bit positions indicates that the particular channel has reached terminal count.
- These status bits are cleared after a read operation by microprocessor.
- The bit B4 of status register is called update flag and a one in this bit position indicates that the channel-2 register has been reloaded from channel-3 registers in the auto load mode of operation.
- The internal addresses of the registers of 8257 are listed in table.

Register	Binary Address								Hexa Address
	Decoder input and enable				Input to address pins of 8257				
	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	
Channel-0 DMA address register	0	1	1	0	0	0	0	0	60
Channel-0 count register	0	1	1	0	0	0	0	1	61
Channel-1 DMA address register	0	1	1	0	0	0	1	0	62
Channel-1 count register	0	1	1	0	0	0	1	1	63
Channel-2 DMA address register	0	1	1	0	0	1	0	0	64
Channel-2 count register	0	1	1	0	0	1	0	1	65
Channel-3 DMA address register	0	1	1	0	0	1	1	0	66
Channel-3 count register	0	1	1	0	0	1	1	1	67
Mode set register (Write only)	0	1	1	0	1	0	0	0	68
Status register (Read only)	0	1	1	0	1	0	0	0	68

INTERFACING OF DMA 8257 WITH 8085

- A simple schematic for interfacing the 8257 with 8085 processor is shown.
- The 8257 can be either memory mapped or I/O mapped in the system.
- In the schematic shown in figure 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 signal IOCS-6 is used to select 8257.
- The address line A7 and the control signal IO/M (low) are used as enable for decoder.
- The D0-D7 lines of 8257 are connected to data bus lines D0-D7 for data transfer with processor during programming mode.
- These lines (D0-D7) are also used by 8257 to supply the memory address A8-A15 during the DMA mode.
- The 8257 also supply two control signals ADSTB and AEN to latch the address supplied by it during DMA mode on external latches.
- Two 8-bit latches are provided to hold the 16-bit memory address during DMA mode. During DMA mode, the AEN signal is also used to disable the buffers and latches used for address, data and control signals of the processor.
- The 8257 provide separate read and write control signals for memory and I/O devices during DMA.
- Therefore the RD (low), WR (low) and IO/M (low) of the 8085 processor are decoded by a suitable logic circuit to generate separate read and write control signals f memory and I/O devices.
- The output clock of 8085 processor should be inverted and supplied to 8257 clock input for proper operation.
- The HRQ output of 8257 is connected to HOLD input of 8085 in order to make a HOLD request to the processor.
- The HLDA output of 8085 is connected to HLDA input of 8257, in order to receive the acknowledge signal from the processor once the HOLD request is accepted.
- The RESET OUT of 8085 processor is connected to RESET of 8257.
- The I/O addresses of the internal registers of 8257 are listed in table.



Register	Binary Address								Hexa Address
	Decoder input and enable				Input to address pins of 8257				
	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	
Channel-0 DMA address register	0	1	1	0	0	0	0	0	60
Channel-0 count register	0	1	1	0	0	0	0	1	61
Channel-1 DMA address register	0	1	1	0	0	0	1	0	62
Channel-1 count register	0	1	1	0	0	0	1	1	63
Channel-2 DMA address register	0	1	1	0	0	1	0	0	64
Channel-2 count register	0	1	1	0	0	1	0	1	65
Channel-3 DMA address register	0	1	1	0	0	1	1	0	66
Channel-3 count register	0	1	1	0	0	1	1	1	67
Mode set register (Write only)	0	1	1	0	1	0	0	0	68
Status register (Read only)	0	1	1	0	1	0	0	0	68

3.Explain programmable interrupt controller 8259 with neat block diagram

FEATURES OF 8259:

1. It is programmed to work with either 8085 or 8086 processor.

2. It manage 8-interrupts according to the instructions written into its control registers.

3. In 8086 processor, it supplies the type number of the interrupt and the type number is programmable. In 8085 processor, the interrupt vector address is programmable. The priorities of the interrupts are programmable.

4. The interrupts can be masked or unmasked individually.

5. The 8259s can be cascaded to accept a maximum of 64 interrupts.

FUNCTIONAL BLOCK DIAGRAM OF 8259:

It has eight functional blocks. They are,

1. Control logic
2. Read Write logic
3. Data bus buffer
4. Interrupt Request Register (IRR)
5. In-Service Register (ISR)
6. Interrupt Mask Register (IMR)
7. Priority Resolver (PR)
8. Cascade buffer.

The data bus and its buffer are used for the following activities.

1. The processor sends control word to data bus buffer through D0-D7.
2. The processor read status word from data bus buffer through D0-D7
3. From the data bus buffer the 8259 send type number (in case of 8086) or the call opcode and address (in case of 8085) through D0-D7 to the processor.

- The processor uses the RD (low), WR (low) and A0 to read or write 8259.
- The 8259 is selected by CS (low).

- The IRR has eight input lines (IR0-IR7) for interrupts. When these lines go high, the request is stored in IRR. It registers a request only if the interrupt is unmasked.

- Normally IR0 has highest priority and IR7 has the lowest priority. The priorities of the interrupt request input are also programmable.

- First the 8259 should be programmed by sending Initialization Command Word (ICW) and Operational Command Word (OCW). These command words will inform 8259 about the following,

1. Type of interrupt signal (Level triggered / Edge triggered).
2. Type of processor (8085/8086).
3. Call address and its interval (4 or 8)
4. Masking of interrupts.
5. Priority of interrupts.
6. Type of end of interrupts.

- The interrupt mask register (IMR) stores the masking bits of the interrupt lines to be masked. The relevant information is send by the processor through OCW.

- The in-service register keeps track of which interrupt is currently being serviced.

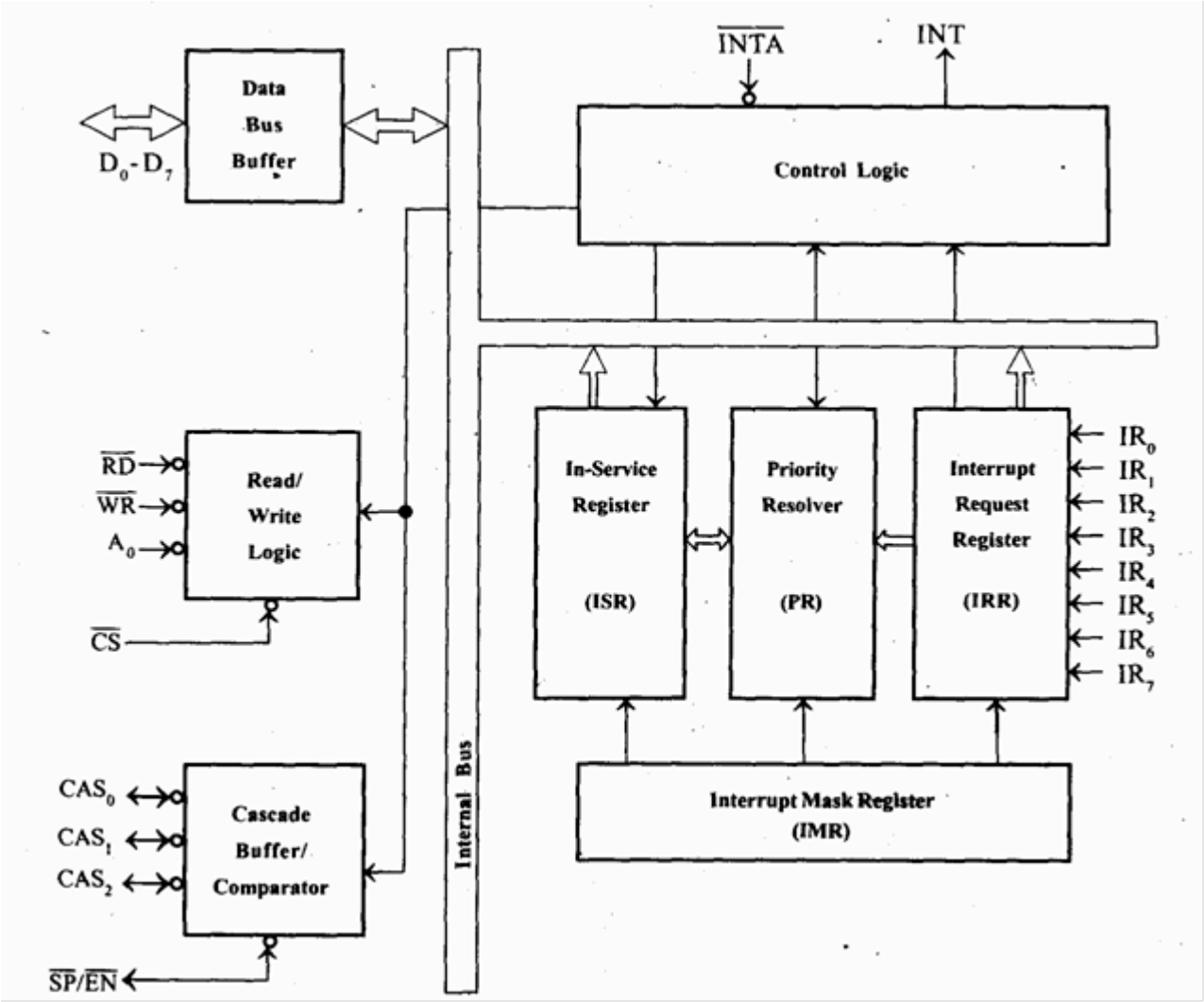
- The priority resolver examines the interrupt request, mask and in-service registers and determines whether INT signal should be sent to the processor or not.

- The cascade buffer/comparator is used to expand the interrupts of 8259.

- In cascade connection one 8259 will be directly interrupting 8086 and it is called master 8259.

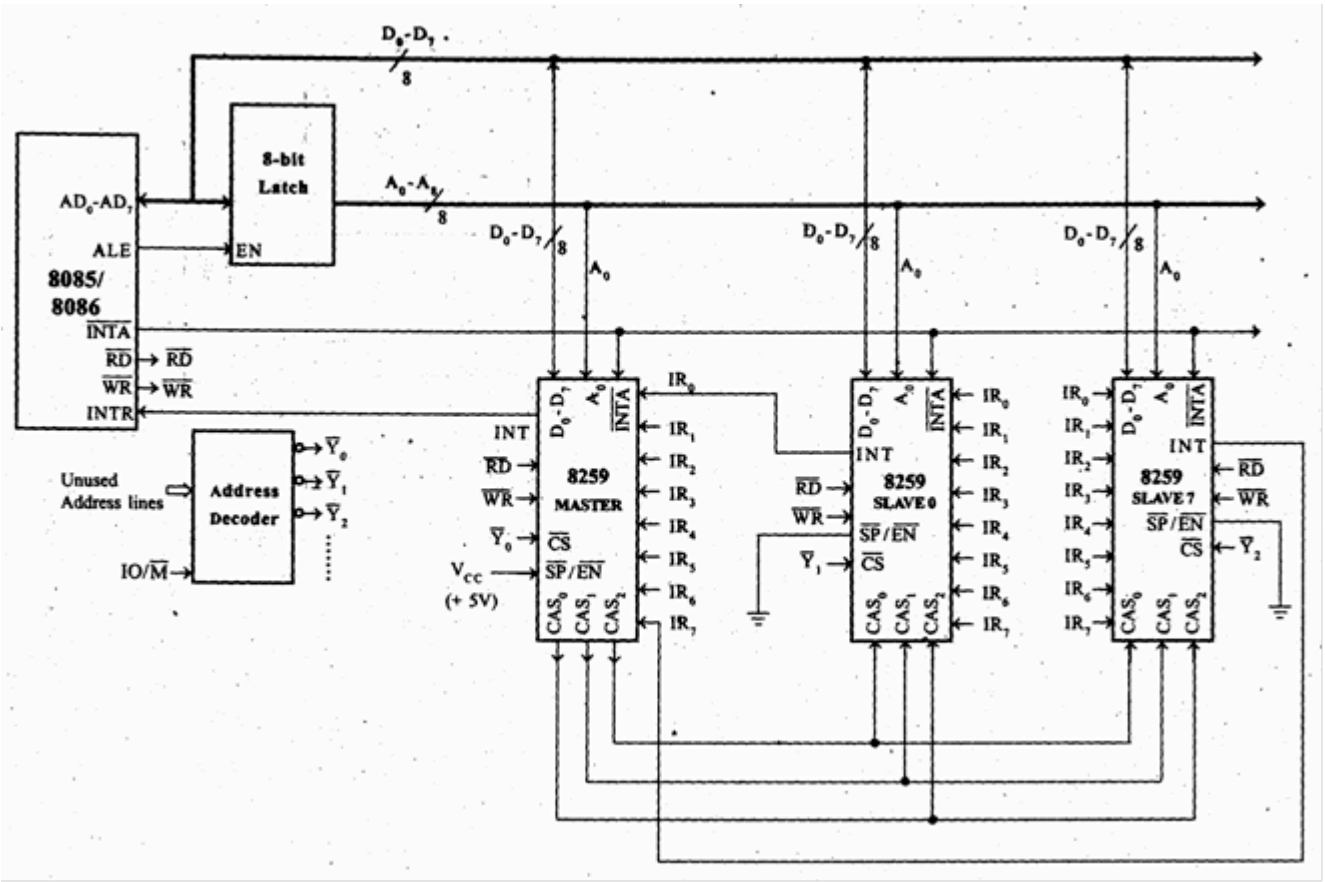
- To each interrupt request input of master 8259 (IR0-IR7), one slave 8259 can be connected. The 8259s interrupting the master 8259 are called slave 8259s.

- Each 8259 has its own addresses so that each 8259 can be programmed independently by sending command words and independently the status bytes can be read from it.



CASCADING OF 8259 & INTERFACING OF 8259 WITH 8085

- The cascade pins (CAS0, CAS1 and CAS2) from the master are connected to the corresponding pins of the slave.
- For the slave 8259, the SP (low) / EN (low) pin is tied low to let the device know that it is a slave.
- The SP (low) / EN (low) pin can be used as input or output signal.
- In non-buffered mode it is used as input signal and tied to logic-I in master 8259 and logic-0 in slave 8259.
- In buffered mode it is used as output signal to disable the data buffers while data is transferred from 8259A to the CPU.



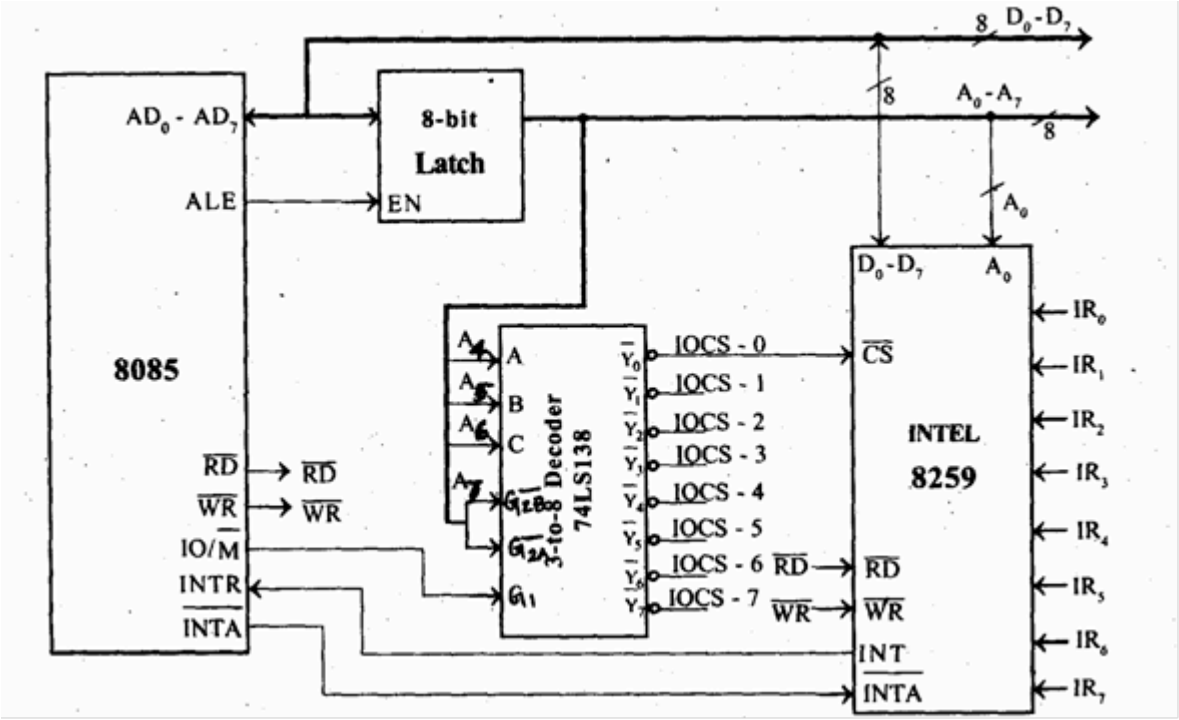
- It requires two internal address and they are A =0 or A = 1.
- It can be either memory mapped or I/O mapped in the system. The interfacing of 8259 to 8085 is shown in figure is I/O mapped in the system.
- The low order data bus lines D0-D7 are connected to D0-D7 of 8259.
- The address line A0 of the 8085 processor is connected to A0 of 8259 to provide the internal address.
- The 8259 require one chip select signal. Using 3-to-8 decoder generates the chip select signal for 8259.
- The address lines A4, A5 and A6 are used as input to decoder.
- The control signal IO/M (low) is used as logic high enables for decoder and the address line A7 is used as logic low enable for decoder.
- The I/O addresses of 8259 are shown in table.

	Binary Address								Hexa address
	Decoder input/ enable			Input to address pin of 8259					
	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	
For A_0 of 8259 to be zero	0	0	0	0	x	x	x	0	00
For A_0 of 8259 to be one	0	0	0	0	x	x	x	1	01

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

Fig - Cascade Connection of 8259

INTERFACING 8259 WITH 8085 MICROPROCESSOR
CASCADING 8259:



Working of 8259 with 8085 processor:

- First the 8259 should be programmed by sending Initialization Command Word (ICW) and Operational Command Word (OCW). These command words will inform 8259 about the following,

1. Type of interrupt signal (Level triggered / Edge triggered).
2. Type of processor (8085/8086).
3. Call address and its interval (4 or 8)
4. Masking of interrupts.
5. Priority of interrupts.
6. Type of end of interrupts.

- Once 8259 is programmed it is ready for accepting interrupt signal. When it receives an interrupt through any one of the interrupt lines IR0-IR7 it checks for its priority and also checks whether it is masked or not.

- If the previous interrupt is completed and if the current request has highest priority and unmasked, then it is serviced

- For servicing this interrupt the 8259 will send INT signal to INTR pin of 8085.
- In response it expects an acknowledge INTA (low) from the processor.
- When the processor accepts the interrupt, it sends three INTA (low) one by one.
- In response to first, second and third INTA (low) signals, the 8259 will supply CALL opcode, low byte of call address and high byte of call address respectively. Once the processor receives the call opcode and its address, it saves the content of program counter (PC) in stack and load the CALL address in PC and start executing the interrupt service routine stored in this call address.

4. Explain in detail aboutKEYBOARD/DISPLAY CONTROLLER - INTEL 8279

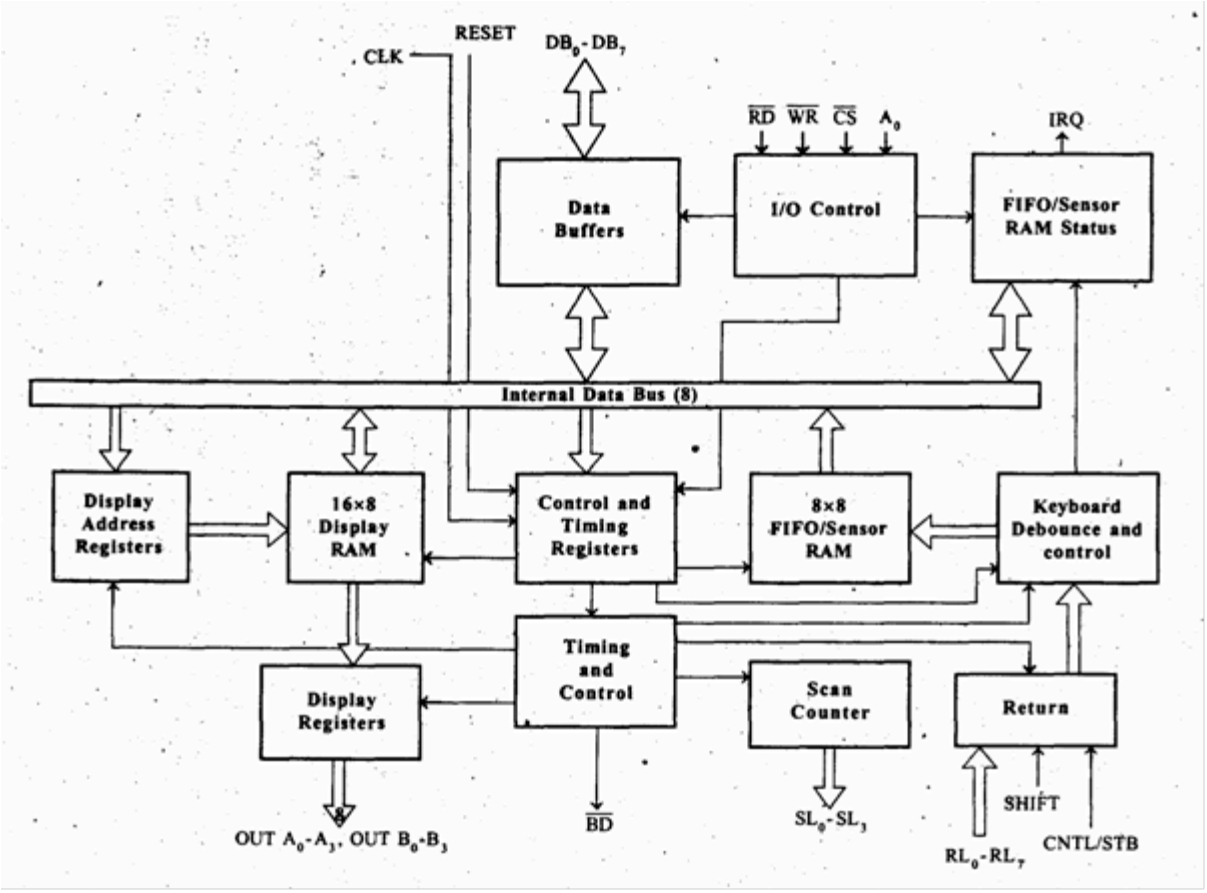
The INTEL 8279 is specially developed for interfacing keyboard and display devices to 8085/8086/8088 microprocessor based system. The important features of 8279 are,

- Simultaneous keyboard and display operations.
- Scanned keyboard mode.
- Scanned sensor mode.
- 8-character keyboard FIFO.
- 1 6-character display.
- Right or left entry 1 6-byte display RAM.
- Programmable scan timing.

- The four major sections of 8279 are keyboard, scan, display and CPU interface.

Keyboard section:

- The keyboard section consists of eight return lines RL0 - RL7 that can be used to form the columns of a keyboard matrix.
- It has two additional input : shift and control/strobe. The keys are automatically debounced.
- The two operating modes of keyboard section are 2-key lockout and N-key rollover.
- In the 2-key lockout mode, if two keys are pressed simultaneously, only the first key is recognized.
- In the N-key rollover mode simultaneous keys are recognized and their codes are stored in FIFO.
- The keyboard section also have an 8 x 8 FIFO (First In First Out) RAM.
- The FIFO can store eight key codes in the scan keyboard mode. The status of the shift key and control key are also stored along with key code. The 8279 generate an interrupt signal when there is an entry in FIFO. The format of key code entry in FIFO for scan keyboard mode is,

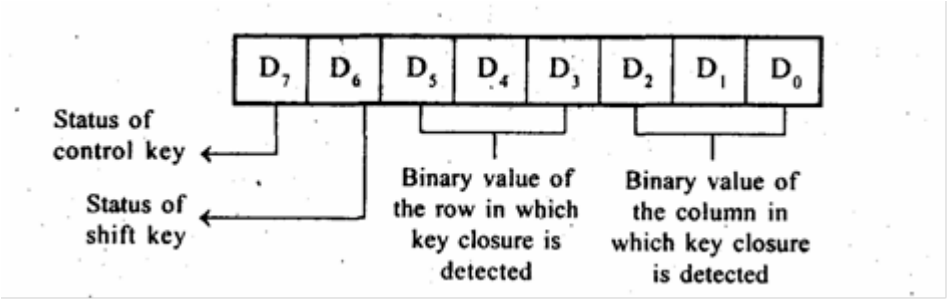


Functional block diagram

KEYBOARD AND DISPLAY INTERFACE USING 8279>>

Block diagram of 8279:

- The functional block diagram of 8279 is shown.



- In sensor matrix mode the condition (i.e., open/close status) of 64 switches is stored in FIFO RAM. If the condition of any of the switches changes then the 8279 asserts IRQ as high to interrupt the processor.

Display section:

- The display section has eight output lines divided into two groups A0-A3 and B0-B3.
- The output lines can be used either as a single group of eight lines or as two groups of four lines, in conjunction with the scan lines for a multiplexed display.

- The output lines are connected to the anodes through driver transistor in case of common cathode 7-segment LEDs.
- The cathodes are connected to scan lines through driver transistors.
- The display can be blanked by BD (low) line.
- The display section consists of 16 x 8 display RAM. The CPU can read from or write into any location of the display RAM.

Scan section:

- The scan section has a scan counter and four scan lines, SL0 to SL3.
- In decoded scan mode, the output of scan lines will be similar to a 2-to-4 decoder.
- In encoded scan mode, the output of scan lines will be binary count, and so an external decoder should be used to convert the binary count to decoded output.
- The scan lines are common for keyboard and display.
- The scan lines are used to form the rows of a matrix keyboard and also connected to digit drivers of a multiplexed display, to turn ON/OFF.

CPU interface section:

- The CPU interface section takes care of data transfer between 8279 and the processor.
- This section has eight bidirectional data lines DB0 to DB7 for data transfer between 8279 and CPU.
- It requires two internal address A =0 for selecting data buffer and A = 1 for selecting control register of8279.
- The control signals WR (low), RD (low), CS (low) and A0 are used for read/write to 8279.
- It has an interrupt request line IRQ, for interrupt driven data transfer with processor.
- The 8279 require an internal clock frequency of 100 kHz. This can be obtained by dividing the input clock by an internal prescaler.
- The RESET signal sets the 8279 in 16-character display with two -key lockout keyboard modes.

Programming the 8279:

- The 8279 can be programmed to perform various functions through eight command words.

INTERFACING OF 8279 WITH 8085

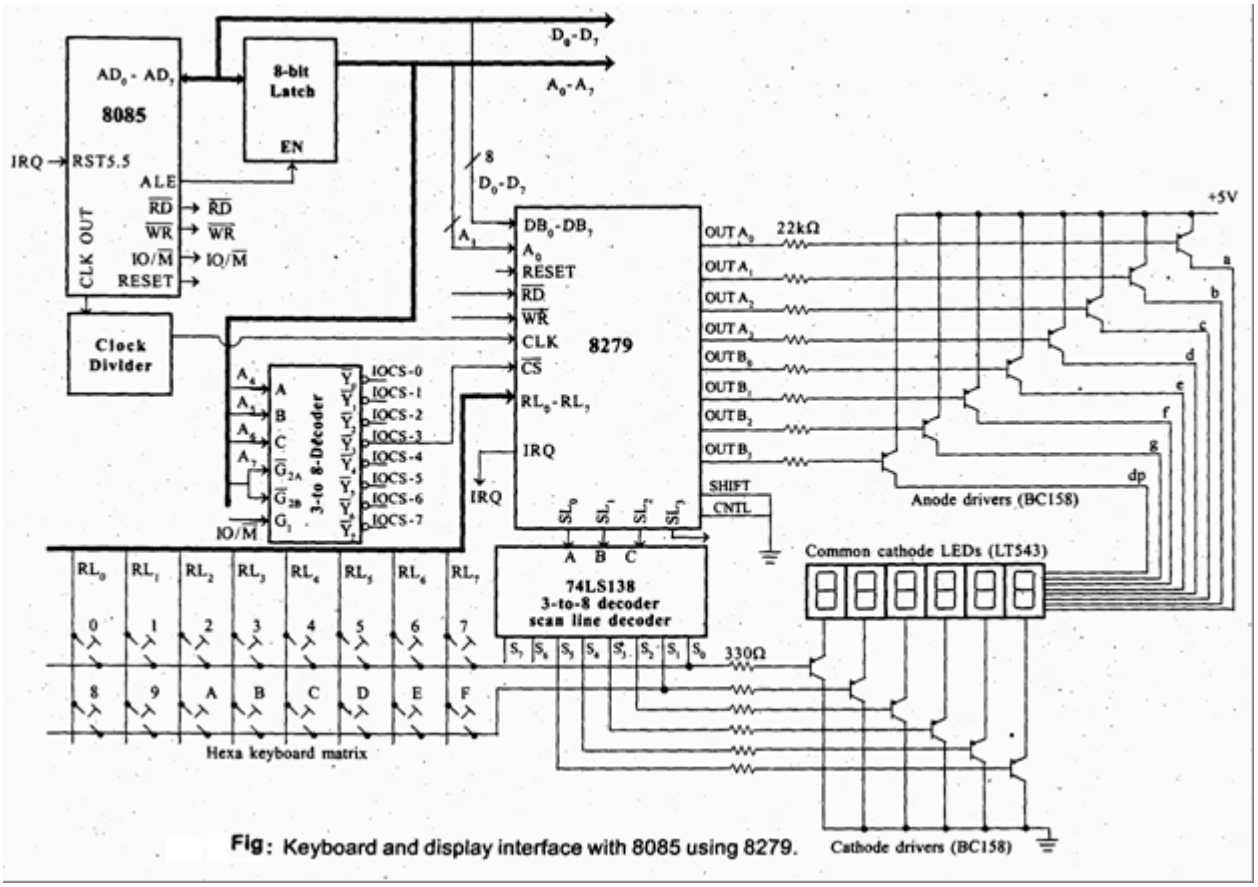
In a microprocessor b system, when keyboard and 7-segment LED display is interfaced using ports or latches then the processor has to carry the following task.

- Keyboard scanning
- Key debouncing
- Key code generation
- Sending display code to LED
- Display refreshing

Interfacing 8279 with 8085 processor:

- A typical Hexa keyboard and 7-segment LED display interfacing circuit using 8279 is shown.
- The circuit can be used in 8085 microprocessor system and consist of 16 numbers of hexa-keys and 6 numbers of 7-segment LEDs.
- The 7-segment LEDs can be used to display six digit alphanumeric character.
- The 8279 can be either memory mapped or I/O mapped in the system. In the circuit shown is the 8279 is I/O mapped.
- The address line A0 of the system is used as A0 of 8279.
- The clock signal for 8279 is obtained by dividing the output clock signal of 8085 by a clock divider circuit.
- The chip select signal is obtained from the I/O address decoder of the 8085 system. The chip select signals for I/O mapped devices are generated by using a 3-to-8 decoder.
- The address lines A4, A5 and A6 are used as input to decoder.
- The address line A7 and the control signal IO/M (low) are used as enable for decoder.
- The chip select signal IOCS-3 is used to select 8279.
- The I/O address of the internal devices of 8279 are shown in table.

Internal Device	Binary Address								Hexa Address
	Decoder input and enable				Input to address line of 8279				
	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	
Data register	0	0	1	1	x	x	x	0	30
Control register	0	0	1	1	x	x	x	1	31



- The circuit has 6 numbers of 7-segment LEDs and so the 8279 has to be programmed in encoded scan. (Because in decoded scan, only 4 numbers of 7-segment LEDs can be interfaced)

- In encoded scan the output of scan lines will be binary count. Therefore an external, 3-to-8 decoder is used to decode the scan lines SL0, SL1 and SL2 of 8279 to produce eight scan lines S0 to S7.

- The decoded scan lines S0 and S1 are common for keyboard and display.

- The decoded scan lines S2 to S5 are used only for display and the decoded scan lines S6 and S7 are not used in the system.

- Anode and Cathode drivers are provided to take care of the current requirement of LEDs.

- The pnp transistors, BC 158 are used as driver transistors.

- The anode drivers are called segment drivers and cathode drivers are called digit drivers.

- The 8279 output the display code for one digit through its output lines (OUT A0 to OUT A3 and OUT B0 to OUT B3) and send a scan code through, SL0- SL3.

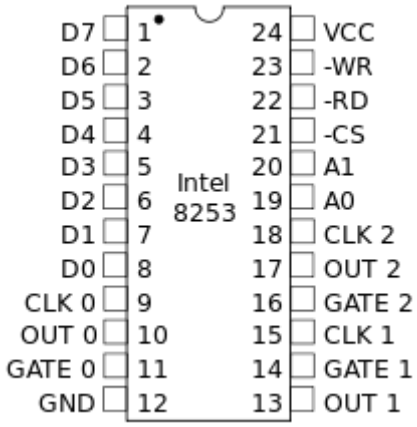
- The display code is inverted by segment drivers and sent to segment bus.

- The scan code is decoded by the decoder and turns ON the corresponding digit driver. Now one digit of the display character is displayed. After a small interval (10 milli-second, typical), the display is turned OFF (i.e., display is blanked) and the above process is repeated for next digit. Thus multiplexed display is performed by 8279.

- The keyboard matrix is- formed using the return lines, RL0 to RL3 of 8279 as columns and decoded scan lines S0 and S1 as rows.
- A hexa key is placed at the crossing point of each row and column. A key press will short the row and column. Normally the column and row line will be high.
- During scanning the 8279 will output binary count on SL0 to SL3, which is decoded by decoder to make a row as zero. When a row is zero the 8279 reads the columns. If there is a key press then the corresponding column will be zero.
- If 8279 detects a key press then it wait for debounce time and again read the columns to generate key code.
- In encoded scan keyboard mode, the 8279 stores an 8-bit code for each valid key press. The keycode consist of the binary value of the column and row in which the key is found and the status of shift and control key.
- After a scan time, the next row is made zero and the above process is repeated and so on. Thus 8279 continuously scan the keyboard.

5.Explain in detail about timer Intel 8253

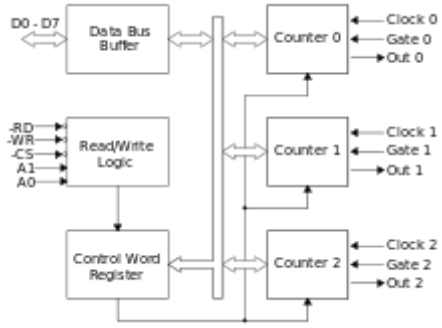
pin diagram



The Intel 8253 and 8254 are Programmable Interval Timers (PITs), which perform timing and counting functions. They were primarily designed for the Intel 8080/8085-processors, but later used in x86-systems.

The 8253 was used in IBM PC compatibles since their introduction in 1981.[1] In modern times, this PIT is not included as a separate chip in an x86 PC. Rather, its functionality is included as part of the motherboard's south bridge chipset. In some modern chipsets, this change may show up as measurable timing differences in accessing a PIT using the x86 I/O address space. Reads and writes to such a PIT's registers in the I/O address space may complete much faster.

Newer motherboards also include a counter through the Advanced Configuration and Power Interface (ACPI), a counter on the Local Advanced Programmable Interrupt Controller (Local APIC), and a High Precision Event Timer. The CPU itself also provides the Time Stamp Counter (TSC) facility.



Block diagram of Intel 8253

The timer has three counters, called channels. Each channel can be programmed to operate in one of six modes. Once programmed, the channels can perform their tasks independently. The timer is usually assigned to IRQ-0 (highest priority hardware interrupt) because of the critical function it performs and because so many devices depend on it.

Counters

There are 3 counters (or timers), which are labeled as **Counter 0**, **Counter 1** and **Counter 2**. Each counter has 2 input pins – **CLK** (clock input) and **GATE** – and 1-pin, **OUT**, for data output. The 3 counters are 16-bit down counters independent of each other, and can be easily read by the CPU.

In the original IBM PCs, the first counter (selected by setting A1=A0=0, see **Control Word Register** below) is used to generate a timekeeping interrupt. The second counter (A1=0, A0=1) is used to trigger the refresh of DRAM memory. The last counter (A1=1, A0=0) is used to generate tones via the PC speaker.

Besides the counters, a typical Intel 8253 microchip also contains the following components:

Data/Bus Buffer

This block contains the logic to buffer the data bus to / from the microprocessor, and to the internal registers. It has 8 input pins, usually labelled as D7..D0, where D7 is the MSB.

Read/Write Logic

The **Read/Write Logic** block has 5 pins, which are listed below. Notice that **/X** denotes an active low signal.

- **/RD**: read signal
- **/WR**: write signal

- **/CS**: chip select signal
- **A0**, **A1**: address lines

Operation mode of the PIT is changed by setting the above hardware signals. For example, to write to the **Control Word Register**, one needs to set /CS=0, /RD=1, /WR=0, A1=A0=1.

Control Word Register

Port 43h R/W
Port 53h R/W – second chip ...

This register contains the programmed information which will be sent (by the microprocessor) to the device. It defines how the PIT logically works. Each access to these ports takes about 1 µs.

To initialize the counters, the microprocessor must write a control word (CW) in this register. This can be done by setting proper values for the pins of the **Read/Write Logic** block and then by sending the control word to the **Data/Bus Buffer** block.

The control word register contains 8 bits, labeled D7..D0 (D7 is the MSB).

Bit#	D7	D6	D5	D4	D3	D2	D1	D0	Short Description
Name	SC1	SC0	RW1	RW0	M2	M1	M0	BCD	
	0	0	x	x	x	x	x	x	Counter 0 at port 40h R/W
	0	1	x	x	x	x	x	x	Counter 1 at port 41h R/W
	1	0	x	x	x	x	x	x	Counter 2 at port 42h R/W
	x	x	0	0	x	x	x	x	Counter Latch, value can be read out in the way RW1, RW0 was set before. The value is held until it is read out or overwritten.
	x	x	0	1	x	x	x	x	Read/Write bits 0..7 of counter value
	x	x	1	0	x	x	x	x	Read/Write bits 8..15 of counter value
	x	x	1	1	x	x	x	x	2xRead/2xWrite bits 0..7 then 8..15 of counter value
	x	x	x	x	0	0	0	x	Mode 0: Interrupt on Terminal Count
	x	x	x	x	0	0	1	x	Mode 1: Hardware Retriggerable One-Shot
	x	x	x	x	0	1	0	x	Mode 2: Rate Generator
	x	x	x	x	0	1	1	x	Mode 3: Square Wave
	x	x	x	x	1	0	0	x	Mode 4: Software Triggered Strobe

	x	x	x	x	1	0	1	x	Mode 5: Hardware Triggered Strobe (Retriggerable)
	x	x	x	x	x	x	x	0	Counter is a 16 bit binary counter(0..65535,FFFFh)
	x	x	x	x	x	x	x	1	Counter is a 16 bit decimal counter 4 x 4bit decades(0..9999)
Name	1	1	‾‾‾‾	‾‾‾‾	C2	C1	C0	0	
			count	status					
	1	1	0	1	x	x	x	0	Counter(C0..C2) value(s) can be read out.
	1	1	1	0	x	x	x	0	Counter's(C0..C2) state(s) can be read out.
	1	1	0	0	x	x	x	0	see below Status Byte error !

When setting the PIT, the microprocessor first sends a control message, then a count message to the PIT. The counting process will start after the PIT has received these messages, and, in some cases, if it detects the rising edge from the **GATE** input signal.

On PCs the address for timer0 (chip) is at port 40h..43h like described and the second timer1 (chip) is at 50h..53h.

Status Byte

8 bit
The Status Byte is read like a 8 bit counter value (port 40h..42h R).

Bit#	D7	D6	D5	D4	D3	D2	D1	D0	
Name	output	null count	RW1	RW0	M2	M1	M0	BCD	
	0	x	x	x	x	x	x	x	Out Pin is 0
	1	x	x	x	x	x	x	x	Out Pin is 1
	x	0	x	x	x	x	x	x	The value of the latch is loaded into the counter.
	x	1	x	x	x	x	x	x	A new value can be written to the latch. Counter value is 0.
	x	x	=	=	=	=	=	=	like defined in the Control Word Register

Operation Modes

The D3, D2, and D1 bits of the **Control Word** set the operating mode of the timer. There are 6 modes in total; for modes 2 and 3, the D3 bit is ignored, so the missing modes 6 and 7 are aliases for modes 2 and 3. Notice that, for modes 0, 2, 3 and 4, **GATE** must be set to **HIGH** to enable

counting. For mode 5, the rising edge of **GATE** starts the count. For details on each mode, see the reference links.

Mode 0 (000): Interrupt on Terminal Count

Mode 0 is used for the generation of accurate time delay under software control. In this mode, the counter will start counting from the initial **COUNT** value loaded into it, down to 0. Counting rate is equal to the input clock frequency.

The **OUT** pin is set low after the **Control Word** is written, and counting starts one clock cycle after the **COUNT** programmed. **OUT** remains low until the counter reaches 0, at which point **OUT** will be set high until the counter is reloaded or the **Control Word** is written. The Gate signal should remain active high for normal counting. If Gate goes low counting get terminated and current count is latched till Gate pulse goes high again.

Mode 1 (001): Programmable One Shot

In this mode 8253 can be used as monostable multivibrator. **GATE** input is used as trigger input.

OUT will be initially high. **OUT** will go low on the **CLK** pulse following a trigger to begin the one-shot pulse, and will remain low until the Counter reaches zero. **OUT** will then go high and remain high until the **CLK** pulse after the next trigger.

After writing the Control Word and initial count, the Counter is armed. A trigger results in loading the Counter and setting **OUT** low on the next **CLK** pulse, thus starting the one-shot pulse. An initial count of **N** will result in a one-shot pulse **N CLK** cycles in duration.

The one-shot is retriggerable, hence **OUT** will remain low for **N CLK** pulses after any trigger. The one-shot pulse can be repeated without rewriting the same count into the counter. **GATE** has no effect on **OUT**. If a new count is written to the Counter during a oneshot pulse, the current one-shot is not affected unless the counter is retriggered. In that case, the Counter is loaded with the new count and the oneshot pulse continues until the new count expires.

Mode 2 (X10): Rate Generator

In this mode, the device acts as a divide-by-n counter, which is commonly used to generate a real-time clock interrupt.

Like other modes, counting process will start the next clock cycle after **COUNT** is sent. **OUT** will then remain high until the counter reaches 1, and will go low for one clock pulse. **OUT** will then go high again, and the whole process repeats itself.

The time between the high pulses depends on the preset count in the counter's register, and is calculated using the following formula:

$$\text{Value to be loaded into counter} = \frac{f_{input}}{f_{output}}$$

Note that the values in the **COUNT** register range from $n$to 1; the register never reaches zero.

Mode 3 (X11): Square Wave Generator

This mode is similar to mode 2. However, the duration of the high and low clock pulses of the output will be different from mode 2.

Suppose **n** is the number loaded into the counter (the **COUNT** message), the output will be

- high for $\frac{n}{2}$ counts, and low for $\frac{n}{2}$ counts, if **n** is even.
- high for $\frac{n+1}{2}$ counts, and low for $\frac{n-1}{2}$ counts, if **n** is odd.

Mode 4 (100): Software Triggered Strobe

After **Control Word** and **COUNT** is loaded, the output will remain high until the counter reaches zero. The counter will then generate a low pulse for 1 clock cycle (a strobe) – after that the output will become high again.

Mode 5 (101): Hardware Triggered Strobe

This mode is similar to mode 4. However, the counting process is triggered by the **GATE** input.

After receiving the **Control Word** and **COUNT**, the output will be set high. Once the device detects a rising edge on the **GATE** input, it will start counting. When the counter reaches 0, the output will go low for one clock cycle – after that it will become high again, to repeat the cycle on the next rising edge of **GATE**.