9. PERIPHERAL CHIPS

9a

8255: Programmable Peripheral Interface

1. Draw the pin diagram of PPI 8255.

Ans. The pin diagram of 8255 is shown in Fig. 9a.1

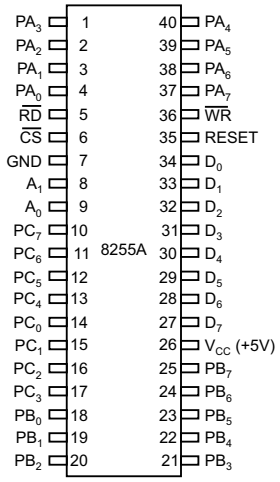


Fig. 9a.1: 8255 Pin diagram
(*Source:* Intel Corporation)

2. Draw the block diagram of 8255.

Ans. The block diagram is shown in Fig. 9a.2.

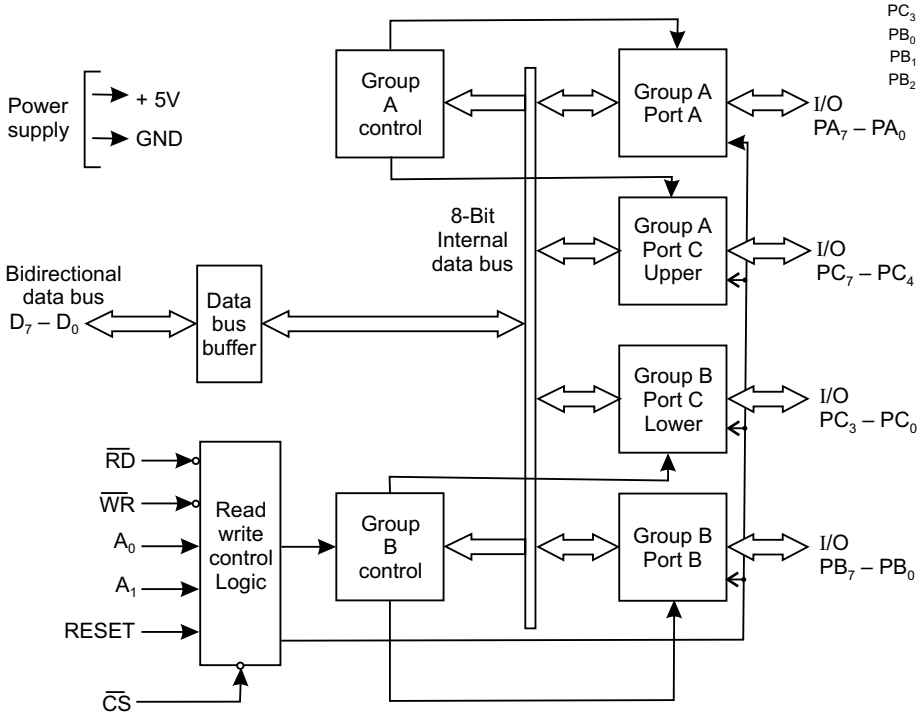


Fig. 9a.2: Block diagram of 8255 (*Source:* Intel Corporation)

3. How many ports are there in 8255 and what are they?

Ans. Basically there are three ports in 8255, viz., Port A, Port B and Port C, each having 8 pins. Again Port C can be divided into Ports C_{upper} and Port C_{lower}—each having four pins i.e., a nibble. Thus 8255 can be viewed to have four ports—Port A, Port B, Port C_{upper} and Port C_{lower}.

4. What pins are associated with Read/Write control logic block?

Ans. There are six pins associated with Read/Write control logic block. These are $\overline{CS}$, $\overline{WR}$, A_0, A_1, Reset and $\overline{CS}$ signals.

5. In how many modes can 8255 operate?

Ans. PPI 8255 can operate in three modes. (*a*) Mode 0 (*b*) Mode 1 and (*c*) Mode 2.

Apart from the above, there is another mode called BSR mode (Bit Set/Reset mode).

6. Distinguish between the three modes of 8255.

Ans. The three modes are Mode 0, Mode 1 and Mode 2. These are I/O operations and selected only if D_7 bit of the control word register is put as 1.

The three operating modes of 8255 are distinguished in the following manner:

Mode 0: This is a basic or simple input/output mode, whose features are:

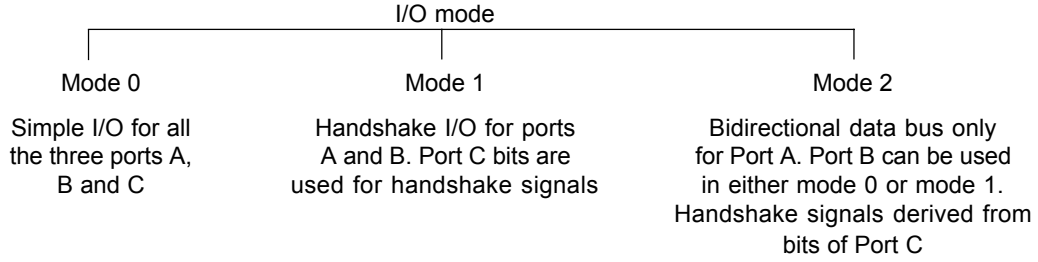
- Outputs are latched.
- Inputs are not latched.
- All ports (A, B, C_U, C_L) can be programmed in either input or output mode.
- Ports don't have handshake or interrupt capability.
- Sixteen possible input/output configurations are possible.

Mode 1: In this mode, input or outputting of data is carried out by taking the help of handshaking signals, also known as strobe signals. The basic features of this mode are:

- Ports A and B can function as 8-bit I/O ports, taking the help of pins of Port C.
- I/Ps and I/Ps are latched.
- Interrupt logic is supported.
- Handshake signals are exchanged between CPU and peripheral prior to data transfer.
- In this mode, Port C is called status port.
- There are two groups in this mode—group A and group B. They can be configured separately. Each group consists of an 8-bit port and a 4-bit port. This 4-bit port is used for handshaking in each group.

Mode 2: In this mode, Port A can be set up for bidirectional data transfer using handshake signals from Port C. Port B can be set up either in mode 0 or mode 1.

The basic operations of the three modes are shown below:



7. Which word determines the operating mode of 8255?

Ans. A single control word determines the operating mode of 8255.

8. For data transfer using 8255, when mode 0 should be selected?

Ans. When unconditional or non-handshaking I/O is required, mode 0 is chosen.

9. How many categories of handshake signals are there? Which is advantageous?

Ans. Handshake signals can be used with either (1) status check I/O or (2) Interrupt I/O.

In the status check I/O, the CPU gets tied up in a loop until the status of the I/O becomes ready while it is the I/O device which interrupts the CPU in interrupt I/O.

10. Explain how the different ports and control words are selected for 8255.

Ans. The two address lines, along with $\overline{CS}$ signal, determine the selection of a particular port or control register. This is explained below:

$\overline{CS}$	A_1	A_0	*Selection*
0	0	0	Port A
0	0	1	Port B
0	1	0	Port C
0	1	1	Control Register
1	X	X	8255 not selected (because $\overline{CS}$ = 1)

$\overline{CS}$ signal is made 0 by choosing $A_7 = 1$ and A_6 though $A_2 = 0$

Thus,

A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0			
1	0	0	0	0	0	0	0	=	80_H	Port A selected
1	0	0	0	0	0	0	1	=	81_H	Port B selected
1	0	0	0	0	0	1	0	=	82_H	Port C selected
1	0	0	0	0	0	1	1	=	83_H	Control Register selected

11. What is BSR mode and what are its characteristics?

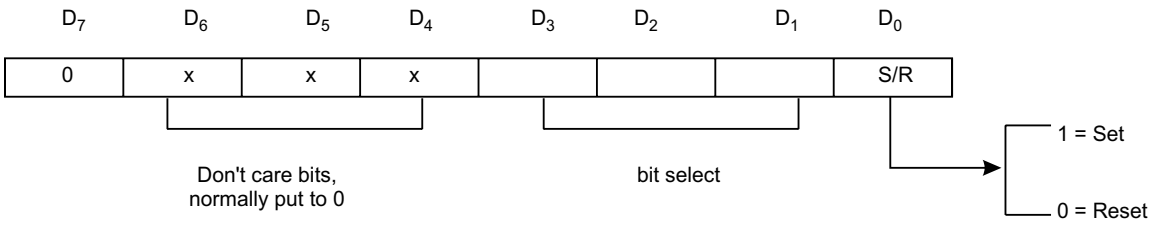
Ans. BSR mode stands for Bit Set Reset mode.

The characteristics of BSR mode are:

- BSR mode is selected only when $D_7 = 0$ of the Control Word Register (CWR).
- Concerned with bits of port C.
- Individual bits of Port C can either be Set or Reset.
- At a time, only a single bit of port C can be Set or Reset.
- Is used for control or on/off switch.
- BSR control word doesn't affect ports A and B functioning.

12. Discuss the control word format in the BSR mode.

Ans. The content of the control word register will be as follows, when used in the BSR mode and selects (either Sets or Resets) a particular bit of Port C at a time.



D_3	D_2	D_1	*Particular bit of Port C selected*
0	0	0	Bit 0
0	0	1	Bit 1
0	1	0	Bit 2
0	1	1	Bit 3
1	0	0	Bit 4
1	0	1	Bit 5
1	1	0	Bit 6
1	1	1	Bit 7

Fig. 9a.3: The CWR in the BSR mode

13. Write a BSR control word to set bits PC_7 and PC_0 and to reset them after 1 second delay.

Ans. To set or reset any particular bit of Port C in the BSR mode, the control word register is to be appropriately loaded. The above is done by loading the accumulator and sending the same to the control register (i.e., by sending the same to the address of the control register). The address of control word register (CWR) is 83_H.

Program:

MVIA, $0F_H$	(Accumulator loaded with $0F_H$ to set PC_7 bit of Port C)
OUT 83_H	(This sets PC_7 bit of Port C)
MVIA, 01H	(Accumulator loaded with 01_H to set PC_0 bit of Port C)
OUT 83_H	(This sets PC_0 bit of Port C)
CALL DELAY	(Assume the DELAY is for 1 second)
MVIA, 00H	(Accumulator loaded with 00_H to reset PC_0 bit of Port C)
OUT 83_H	(This resets PC_0 bit of Port C)
MVIA, $0E_H$	(Accumulator loaded with $0E_H$ to reset PC_7 bit of Port C)
OUT 83_H	(This resets PC_7 bit of Port C)

14. Show the control word format for I/O mode operation of PPI 8255.

Ans. The control word format, when 8255 is operated in I/O mode, is shown below:

For 8255 PPI to be operated in I/O mode, D_7 bit must be 1.

The three ports are clubbed into two groups—Groups A and B. Group A consists of Port A and C_U. Port A can be operated in any of the modes—0, 1 or 2. Group B consists of Port B and C_L. Here Port B can be operated in either mode 0 or 1.

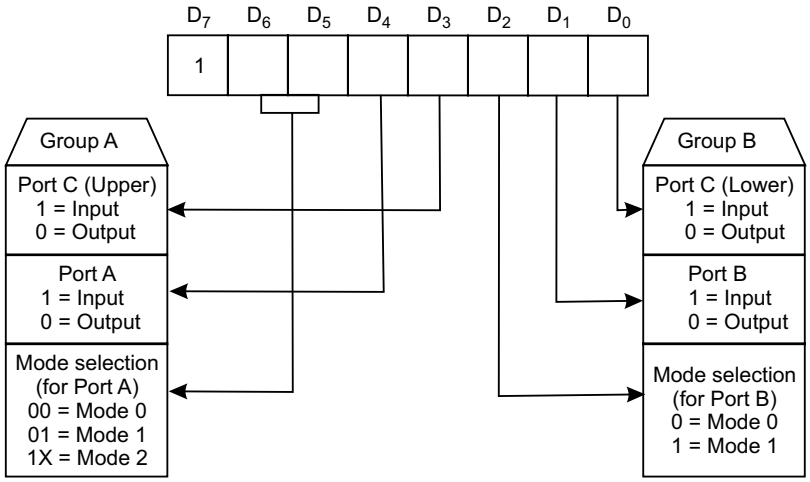


Fig. 9a.4: The CWR in the I/O mode

15. What happens when RESET pin of 8255 is made high?

Ans. When a 1 is applied on RESET pin of 8255, the three ports are put in the input mode. All flip-flops are cleared and interrupts are reset. This condition is not altered even when RESET goes low. 8255 can then be programmed in any mode by appropriately loading the control word register. The mode operation can be changed by altering the content of the control word register, whenever needed.

16. Write down the mode 0 control words for the following two cases:

(*a*) Port A = Input port, Port B = not used, Port C_U = Input port and Port C_L = Output port.

(*b*) Port A = Output port, Port B = Input port, Port C = Output port

Ans. The control words for the two cases will be as follows:

(*a*)

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0		
1	0	0	1	1	0	0	0	=	98_H
I/O mode	Mode 0 for Port A		Port A input	Port C_U input	Port B not used		Port C_L output		

Thus, the control word would be = 98_H

(*b*)

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0		
1	0	0	0	0	0	1	0	=	82_H
I/O mode	Mode 0 for Port A		Port A output	Port C_U output	Mode 0 for Port B	Port B input	Port C_L output		

Thus, the control word would be = 82_H.

17. In mode 1 what are the control signals when ports A and B act as input ports. Discuss the control signals. Draw the timing waveforms for such a strobed input.

Ans. The following are the control signals when ports A and B act as input ports (under mode 1) $\overline{STB_A}$, IBF_A, $INTE_A$ for Port A and $\overline{STB_B}$, IBF_B, $INTE_B$ for Port B, respectively. The details about the input control signals are discussed below:

- $\overline{STB}$ *(Strobe input):* This is an active low signal generated by a peripheral device. When a peripheral device has some valid data, it sends the same via Port A or B and sends a low $\overline{STB}$ signal. This data is accepted by 8255 and it generates a IBF and INTR (provided INTE is set previously).
- *IBF (Input buffer full):* On receipt of $\overline{STB}$ signal from peripheral device, data is stored in 8255 by its input latch. In its turn, 8255 generates a high IBF. IBF is reset when CPU reads the data.
- *INTR (Interrupt request):* This active high output signal is generated only if $\overline{STB}$, IBF and INTE are all set at the same time. This signal interrupts the CPU via its INTR (pin no. 10 of 8085).
- *INTE (Interrupt Enable):* This is an internal F/F which can be set/reset using the BSR mode. It must be set if INTR signal is to be effective.

The following figure shows Port A in Mode 1 (input), along with the timing diagrams.

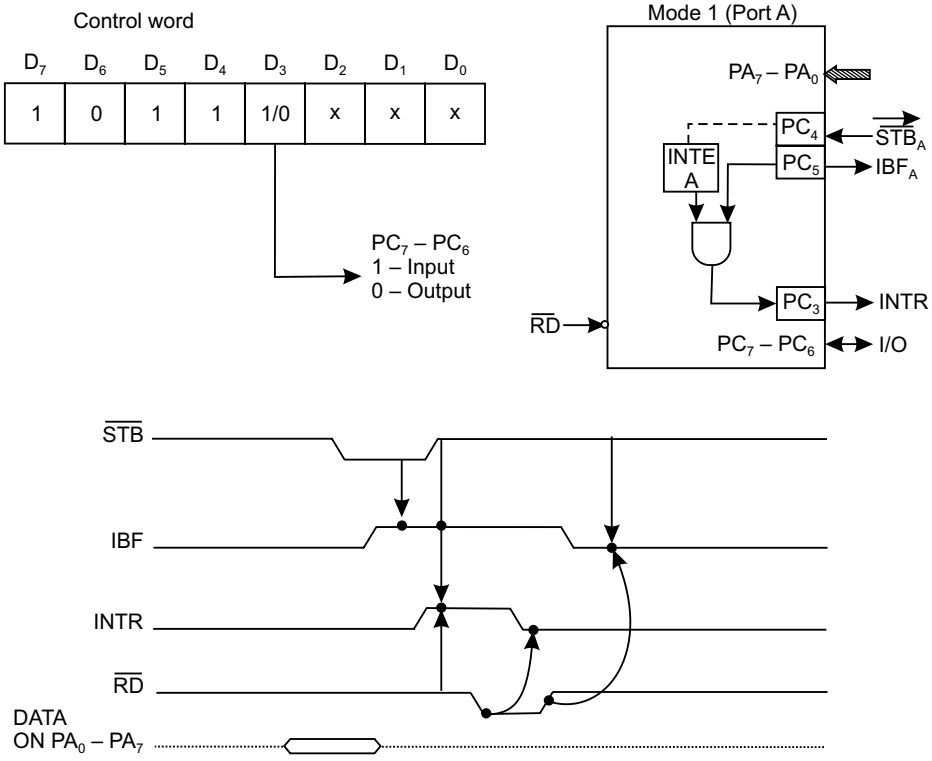


Fig. 9a.5: Port A in Mode 1 (Input) (*Source:* Intel Corporation)

18. In mode 1, what are the control signals when ports A and B act as output ports. Discuss the control signals. Draw the timing waveforms for such a strobed output.

Ans. The following are the control signals when ports A and B act as output ports (under mode 1) $\overline{OBF_A}$, $\overline{AC}_A$, $INTE_A$ for Port A and $\overline{OBF_B}$, $\overline{ACK_B}$, $INTE_B$ for Port B respectively.

The details about the output control signals are discussed below:

- $\overline{OBF}$ *(Output buffer full):* This is an active low output signal. This signal becomes low when the CPU writes data into the output latch of 8255. This output signal from 8255, which goes to a peripheral, indicates to the peripheral that the data on the output latch of 8255 is ready to be read.
- $\overline{ACK}$ *(Acknowledge):* When data reading by the peripheral from the output latch of 8255 is complete (i.e., the peripheral has accepted the data), it (the peripheral) outputs a low signal which is connected to the $\overline{ACK}$ (input signal) pin of 8255. On receipt of this low signal by 8255 (from peripheral), the $\overline{OBF}$ line of 8255 goes high.
- *INTR (Interrupt):* This signal is set only if $\overline{OBF}$, $\overline{ACK}$ and INTE (internal F/F) are all at high(1) state. This output signal from 8255 goes to INTR (pin 10 of 8085) to interrupt the CPU. The INTR signal is reset on the falling edge of $\overline{WR}$.
- *INTE (Interrupt Enable):* This is an internal F/F which can be set/reset in BSR mode. This must be set if INTR signal is to be effective.

The following figure shows Port A in mode 1 (output), along with the timing waveforms.

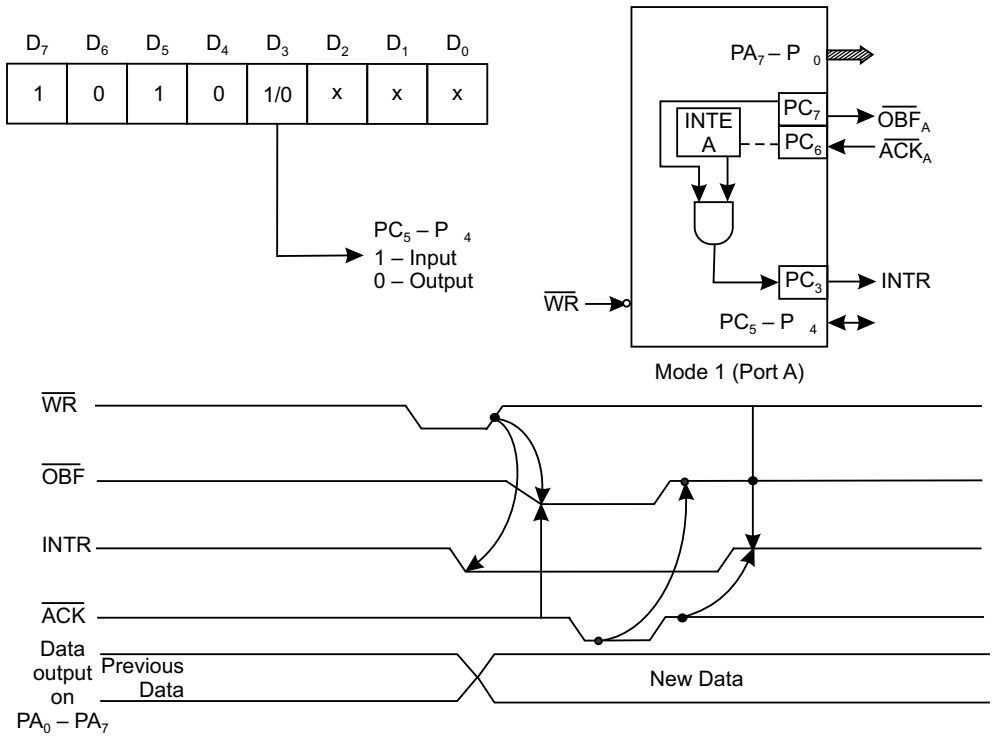


Fig. 9a.6: Port A in mode 1 (Output) (*Source:* Intel Corporation)

19. In mode 1, what are the methods available for data transfer? Which method is advantageous?

Ans. In mode 1, data transfer is possible involving 8255 when it is programmed to function either in (*a*) Status check I/O (also called Program Controlled I/O), (*b*) Interrupt I/O (also called Interrupt Controlled I/O).

The simplified flowcharts for the two schemes are shown below:

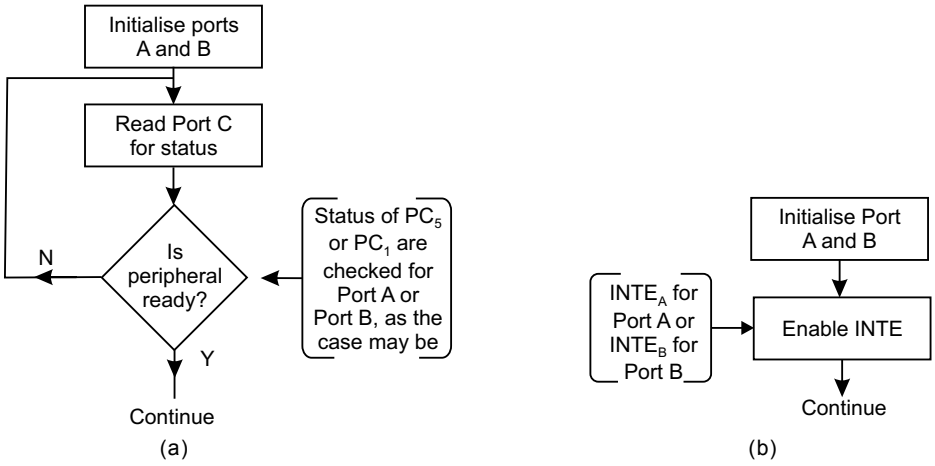


Fig. 9a.7: Flow charts for (a) Status check I/O, (b) Interrupt I/O

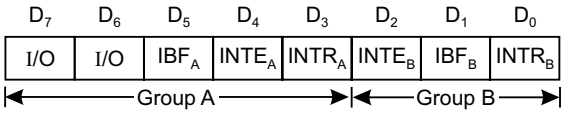
In status check I/O, CPU continues to check the status of IBF_A or IBF_B until they are high. This is done by reading the status word (port C) to check (PC_5 for IBF_A and PC_1 for IBF_B) for existence of IBF. It is known as 'polling' (reading) the status word. Here it is assumed that both Ports A and B act as input ports.

In interrupt I/O scheme, the status of either $INTR_A$ or $INTR_B$ (as the case may be) will have to be ascertained to know the port (A or B) which has requested (interrupted) for service. This is done by reading the status of $INTR_A$ (PC_3) or $INTR_B$ (PC_0) of the status word. Here again, it is assumed that both ports A and B act as input ports.

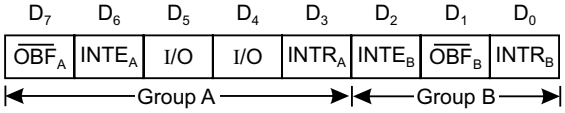
Status check I/O is disadvantageous because in this the CPU gets tied up in the loop until the IBF line (IBF_A or IBF_B, as the case may be) goes high.

20. Show the mode 1 status word format and discuss.

Ans. There are two mode 1 status word formats—one for input configuration and the other for output configuration. These are shown below:



(a) Status Word for Mode 1 Input Configuration



(b) Status Word for Mode 1 Output Configuration

Fig. 9a.8: Status word for mode 1 (a) Input (b) Output configuration

The mode 1 status format (either input or output configuration) can be read by an input read of Port C.

When 8255 is operated in mode 1, the processor has two choices — either polling or interrupt scheme—this is true irrespective of whether data is inputted to the CPU or otherwise.

When data is inputted (i.e., from peripheral to CPU), the CPU can poll (status check) the IBF line for presence of valid data. Else the processor can be interrupted via the INTR line. To know whether $INTR_A$ or $INTR_B$ has interrupted can be ascertained by reading the Port C for status of $INTR_A$ (bit D_3) or $INTR_B$ (bit D_0).

Again, when data is outputted (by CPU to 8255), again two choices are there. The CPU can poll the $\overline{OBF}$ or else be interrupted by INTR line.

If interrupt driven scheme is followed, then INTE (either $INTE_A$ or $INTE_B$) has to be previously set in BSR mode.

A confusion arises when interrupt driven I/O scheme is undertaken. For input configuration, it is seen that $\overline{STB}_A$ signal is connected to PC_4 and $INTE_A$ is controlled by PC_4. For Port B, the corresponding bit is PC_2—i.e., $\overline{STB}_B$ is connected to PC_2 and $INTE_B$ is also controlled by PC_2. But this poses no problem because INTE is set/reset in BSR mode and the BSR control word has no effect when ports A or B are set in mode 1.

21. Discuss the mode 2 of PPI 8255 in brief.

Ans. This mode is usually used for transferring data between two computers.

When operated in mode 2, only Port A can be used as a bidirectional 8-bit I/O bus, using $PC_3 – PC_7$ for handshaking. Port B can be programmed only in mode 0 ($PC_0 – PC_2$ as input or output) or in mode 1 ($PC_0 – PC_2$ used as handshaking signals).

22. How many possible combinations of 8255 would be there when it is operated in mode 2.

Ans. Only Port A can be operated in mode 2 and Port B in either mode 0 or mode 1.

Thus, there would be four possible combinations in mode 2. These are:

(*a*) Mode 2 and Mode 0 (input)
(*b*) Mode 2 and Mode 0 (output)
(*c*) Mode 2 and Mode 1 (input)
(*d*) Mode 2 and Mode 1 (output)

23. Discuss the Mode 2 control word.

Ans. The mode 2 control word is as shown below:

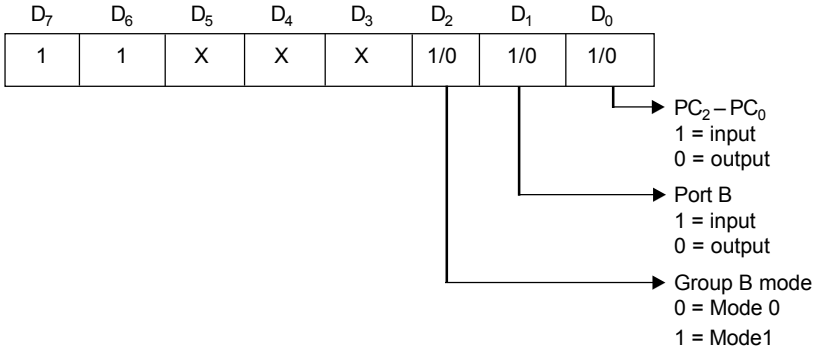


Fig. 9a.9: CWR in Mode 2

This control word is to be loaded into the control port to configure 8255 in mode 2. Bit D_0 of control port determines the I/O operations of $PC_2 - PC_0$. D_1 bit indicates the Port B input/output operation whereas bit D_2 determines Group B to be either in mode 0 or in mode 1 operation.

24. Draw Port A and the associated control signals when 8255 is operated in mode 2.

Ans. This is shown in the following figure:

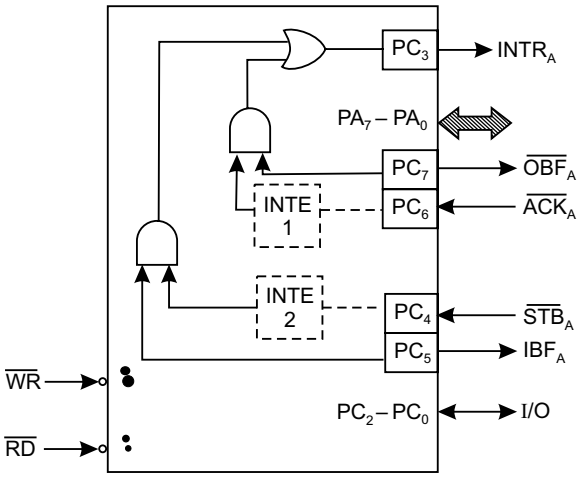


Fig. 9a.10: Mode 2 operation (*Source:* Intel Corporation)

25. What are the output control signals in mode 2 and discuss them?

Ans. The output control signals in mode 2 are $\overline{\text{OBF}}$, $\overline{\text{ACK}}$ and INTE 1. These control signals are discussed below: $\overline{\text{OBF}}$ stands for output buffer full, an active low signal. In its active condition (i.e., low), it indicates that the CPU has written data into Port A.

$\overline{\text{ACK}}$, an active low signal, stands for acknowledge. This acknowledgement signal is generated by a peripheral and it enables the tri-state output buffer or Port A and makes Port A data available to the peripheral.

INTE 1 is an internal F/F associated with output buffer full. INTE 1 can be used to enable or disable the interrupt (INTR) by setting or resetting PC_6 in BSR mode.

26. What are the input control signals in mode 2 and discuss them?

Ans. The input control signals in mode 2 are $\overline{\text{STB}}$, IBF and INTE 2. These control signals are discussed below:

$\overline{\text{STB}}$ (strobe input), an active low signal, enables Port A to latch the data available at its input. This happens if $\overline{\text{STB}}$ = 0.

IBF (Input buffer full), an active high signal, indicates the data has been loaded into the input latch of Port A. This happens if IBF = 1.

INTE 2 is an internal F/F associated with input buffer full. INTE 2 can be used to enable or disable the interrupt (INTR) by setting or resetting PC_4 in BSR mode.

27. Draw a table that summarises the pin functions of Ports A, B and C for various modes of operation.

Ans. The following table shows the pin summary of Ports A, B and C for various modes of operation.

Table 9a.1: Pin summary for all modes of operation (*Source*: Intel Corporation)

	Mode 0		Mode 1		Mode 2	
	In	Out	In	Out	Group A only	
PA_0	In	Out	In	Out	↔	
PA_1	In	Out	In	Out	↔	
PA_2	In	Out	In	Out	↔	
PA_3	In	Out	In	Out	↔	
PA_4	In	Out	In	Out	↔	
PA_5	In	Out	In	Out	↔	
PA_6	In	Out	In	Out	↔	
PA_7	In	Out	In	Out	↔	
PB_0	In	Out	In	Out	—	Mode 0 or Mode 1 only
PB_1	In	Out	In	Out	—	
PB_2	In	Out	In	Out	—	
PB_3	In	Out	In	Out	—	
PB_4	In	Out	In	Out	—	
PB_5	In	Out	In	Out	—	
PB_6	In	Out	In	Out	—	
PB_7	In	Out	In	Out	—	
PC_0	In	Out	$INTR_B$	$INTR_B$	I/O	
PC_1	In	Out	IBF_B	$\overline{OFB}_B$	I/O	
PC_2	In	Out	$\overline{STB}_B$	$\overline{ACK}_B$	I/O	
PC_3	In	Out	$INTR_A$	$INTR_A$	$INTR_A$	
PC_4	In	Out	$\overline{STB}_A$	I/O	$\overline{STB}_A$	
PC_5	In	Out	IBF_A	I/O	IBF_A	
PC_6	In	Out	I/O	$\overline{ACK}_A$	$\overline{ACK}_A$	
PC_7	In	Out	I/O	$\overline{OBF}_A$	$\overline{OBF}_A$	

28. What are the status of Port A outputs until they are enabled.

Ans. Port A outputs remain in the tri-state condition until they are enabled. This enabling is done by a low on the $\overline{ACK}$ signal generated by a peripheral.

29. On which line interrupts are generated in mode 2 for both input and output operations?

Ans. The same ($INTR_A$) is generated on PC_3 line, for both input and output operations.

30. Draw the status word in mode 2 and discuss.

Ans. The status word in mode 2 is as shown:

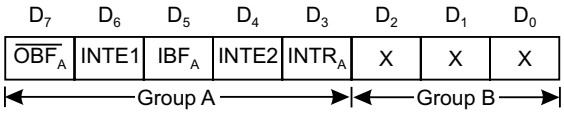


Fig. 9a.11: Status word in mode 2

This status word is accessed by reading Port C. $D_7 - D_3$ bits of the status word carry the status of $\overline{OBF_A}$, INTE 1, IBF_A, INTE 2, $INTR_A$. The status of the remaining three bits i.e., $D_2 - D_0$ depend on the mode setting of Group B. If Group B is programmed to be in mode 0, then $D_2 - D_0$ are simply $PC_2 - PC_0$ (simple I/O). But if Group B is in mode 1, then the three bits $D_2 - D_0$ carry the information about the control signals for Port B—and they depend on whether B is acting as an input port or output port.

31. What are the types of devices with which data transfer takes place via the 8255 PPI?

Ans. The input devices from which data are read and delivered to the microprocessor via the input ports of 8255 PPI are ADCs, keyboards, control signals from process devices, paper tape readers, etc.

The output devices to which data are delivered via the output ports of 8255 are DACs, printers, video display, plotters, etc.

32. What kind of functions do the input and output ports play?

Ans. A port, whether an input or an output port, is a set of D F/Fs consisting of several pins in parallel. When used as input pins, they act as buffers and when used as output pins, they act as latches.

9b

8155/8156: Programmable I/O Ports and Timer

1. In what way 8155 and 8156 differs?

Ans. The Chip Enable (CE) signal is active low for 8155, whereas it is active high for 8156.

2. What are the essential features of 8155.

Ans. The essential features of 8155 are

- 8-bit 256 word RAM memory
- Two programmable 8-bit I/O port
- One programmable 6-bit IO port
- One programmable 14-bit binary Timer/Counter
- An internal address latch
- A control/status (C/S) register
- An internal decoder.

3. Functionally, how many sections are there in 8155?

Ans. Functionally, it has two sections—(*a*) a R/W memory and (*b*) programmable I/O and timer section.

4. Is it necessary to demultiplex the lower order bus AD_7 –AD_0 externally for 8155 to be connected to 8055?

Ans. No, it is not. This is because ALE, IO/$\overline{M}$, $\overline{RD}$ and $\overline{WR}$ signals of 8085 can be connected directly with 8155.

5. Draw the pin diagram of 8155.

Ans. The pin diagram of 8155 is shown in Fig. 9b.1.

6. Draw the functional block diagram of 8155.

Ans. The functional block diagram of 8155 is shown in Fig. 9b.2 :

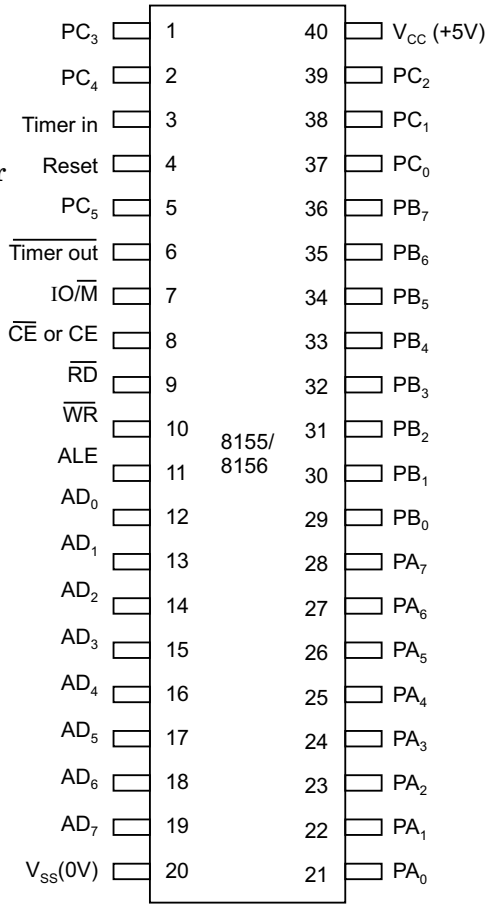


Fig. 9b.1: Pin diagram of 8155 (*Source:* Intel Corporation)

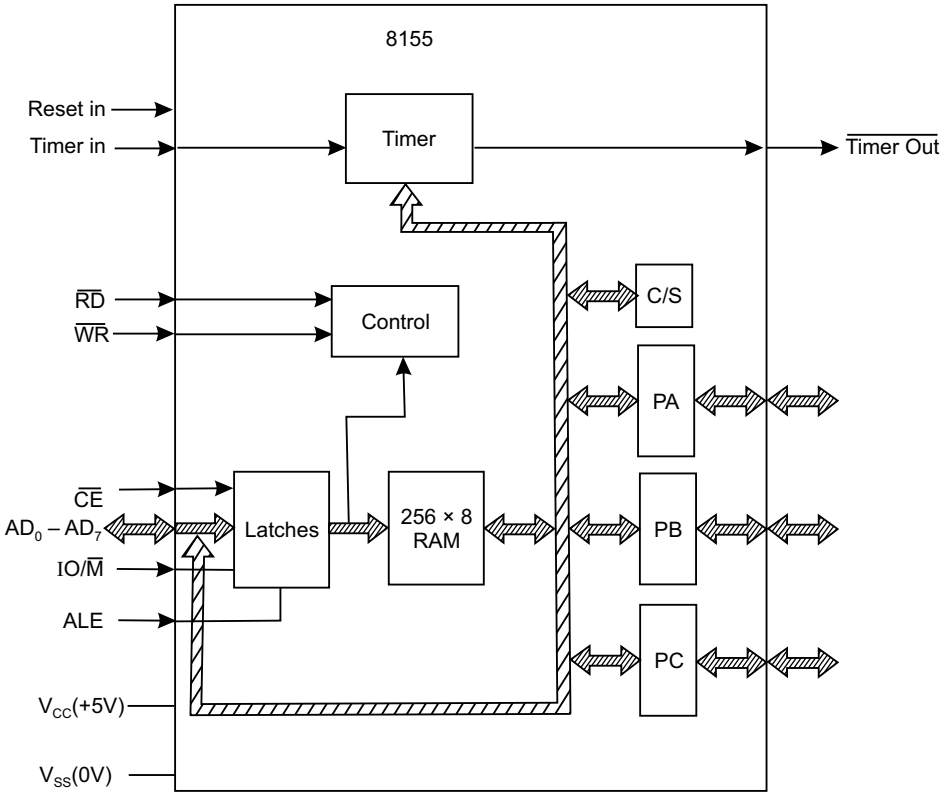


Fig. 9b.2: Functional diagram of 8155 (*Source:* Intel Corporation)

7. How the different ports, control/status register, timers are accessed? Write their addresses also.

Ans. The different combinations on the address lines A_2, A_1, A_0 select one of the above, as shown:

A_2	A_1	A_0		
0	0	0	⇒	Control/Status Register
0	0	1	⇒	Port A
0	1	0	⇒	Port B
0	1	1	⇒	Port C
1	0	0	⇒	LSB Timer
1	0	1	⇒	MSB Timer

The other five (viz., A_7 to A_3) on the address lines are as: 0 0 1 0 0. Thus

A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	Address		Register/Port/Timer
0	0	1	0	0	0	0	0	⇒ 20_H	—	Control/Status register
0	0	1	0	0	0	0	1	⇒ 21_H	—	Port A
0	0	1	0	0	0	1	0	⇒ 22_H	—	Port B
0	0	1	0	0	0	1	1	⇒ 23_H	—	Port C
0	0	1	0	0	1	0	0	⇒ 24_H	—	LSB timer
0	0	1	0	0	1	0	1	⇒ 25_H	—	MSB timer

It is to be noted that the control/status register is having the same address 20_H, but the control register is accessed with $\overline{WR} = 0$ and $\overline{RD} = 1$. For status register access, $\overline{WR} = 1$ and $\overline{RD} = 0$. The control register can never be read. For any future reference, the control register content is stored in some accessible memory location.

8. Draw the control word format and discuss the same in detail.

Ans. The control word loaded in the control register configures the different ports and the timer of 8155. The control word format is shown below:

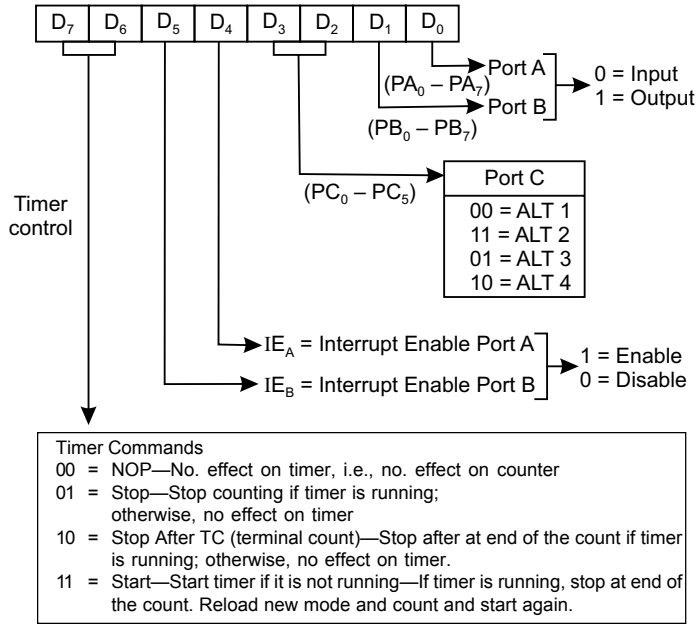


Fig. 9b.3: The control word format

The control register contains eight latches. The content of the lower 2 bits, viz., $D_1 - D_0$ configure ports A and B as input/output. Bits D_3 and D_2 configure bits $PC_0 - PC_5$ of port C (Port C is a 6-bit port while ports A and B both are of 8-bits) and can have four combinations—ALT1, ALT2, ALT3, ALT4 depending on the combinations of D_3 and D_2. Bits D_5 and D_4 are enable/disable pins for ports A and B respectively which enable/disable the internal flip-flop of 8155. Bits D_7 and D_6 contain the timer commands.

As already mentioned, combinations of $D_3 - D_2$ bits give rise to ALT1 to ALT4 modes, which assigns port C bits in different configurations and shown below:

Table 9b.1: Port C pin assignment (*Source:* Intel Corporation)

Pin	ALT1	ALT2	ALT3	ALT4
PC0	Input Port	Output Port	A INTR (Port A Interrupt)	A INTR (Port A Interrupt)
PC1	Input Port	Output Port	A BF (Port A Buffer Full)	A BF (Port A Buffer Full)
PC2	Input Port	Output Port	A $\overline{STB}$ (Port A Strobe)	A $\overline{STB}$ (Port A Strobe)
PC3	Input Port	Output Port	Output Port	B INTR (Port B Interrupt)
PC4	Input Port	Output Port	Output Port	B BF (Port B Buffer Full)
PC5	Input Port	Output Port	Output Port	B $\overline{STB}$ (Port B Strobe)

ALT1 and ALT2 correspond to simple input/output of Port C respectively. In ALT3 mode, $PC_0 - PC_2$ bits are used as control signals for port A, while pins $PC_3 - PC_5$ act as output pins. In ALT4 mode, $PC_0 - PC_2$ bits are used as control signals for port A, while $PC_3 - PC_5$ bits are used as control signals for port B.

9. Draw the status word format and discuss the same.

Ans. The status word format of 8155 is given below:

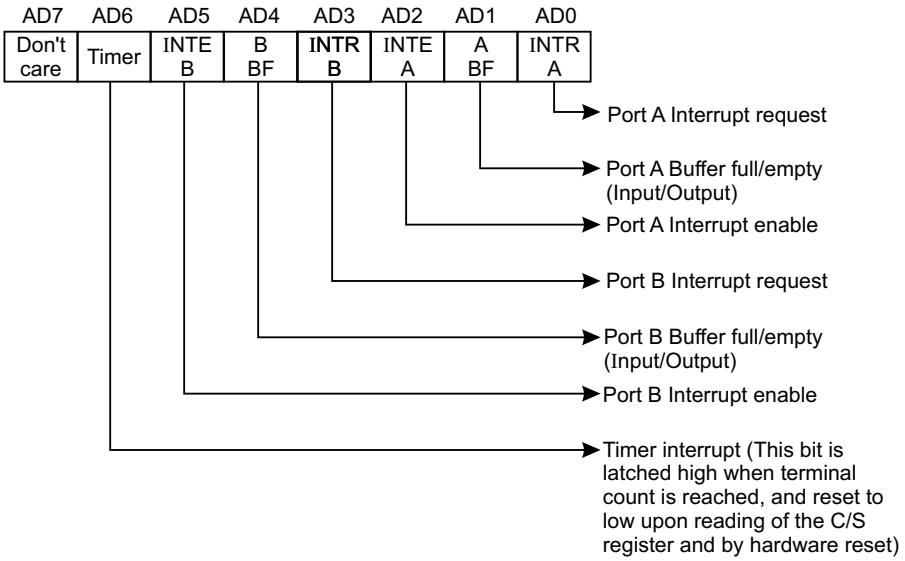


Fig. 9b.4: Status word format (*Source:* Intel Corporation)

It has seven latches. Bit D_7 is the 'don't care' bit. Bit D_6 contains the status of the timer. Bits $D_5 - D_3$ pertain to status of port B while bits $D_2 - D_0$ to that of Port A.

10. Discuss the timer section of 8155 and discuss its operating modes.

Ans. The timer section consists of two 8-bit registers. 14-bits of the two registers comprise to specify the count of the timer, which counts in a count-down manner. Contents of bits 6 and 7 of the most significant byte of the register decide the mode of operation of the counter. The following shows the timer register format. The timer section needs a 'TIMER IN' pulse, which is fed via pin 3 of 8155. A square wave or a pulse is obtained via pin 6 $(\overline{\text{TIMER OUT}})$ when the terminal count (TC) is reached. The maximum and minimum values of the count down timer are $3FF_H$ and 002_H respectively. A single square wave or a continuous square wave or a single pulse on TC or a pulse on each TC (i.e., continuous pulses) are obtained, depending on the mode setting bits M_2 and M_1.

The following figure shows the nature of the outputs for the different modes.

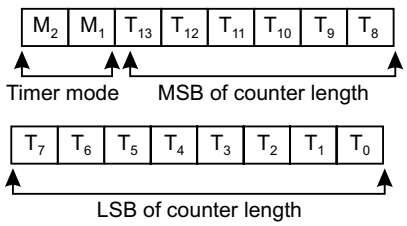


Fig. 9b.5: Timer registers format (*Source:* Intel Corporation)

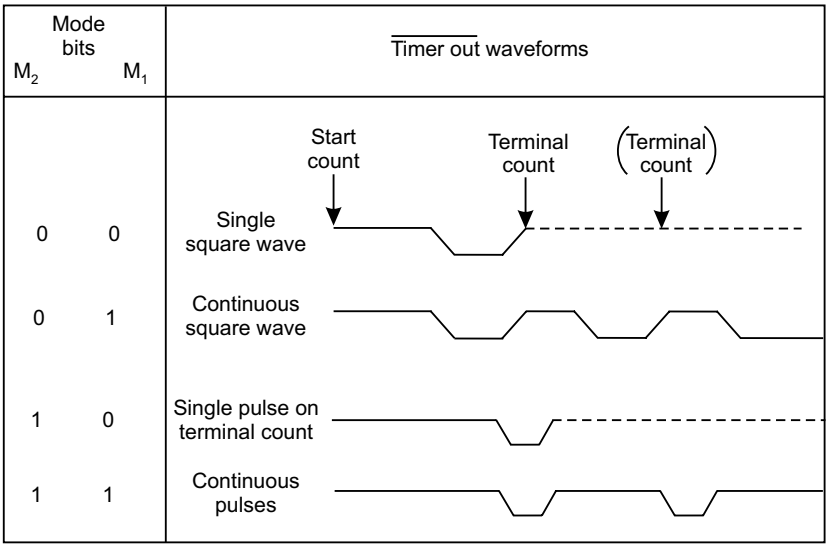


Fig. 9b.6: Timer modes and outputs (*Source:* Intel Corporation)

11. What happens when a high is applied on RESET ?

Ans. A high reset input resets the counter. To restart counting after resetting, a START command is required through the control register.

9c

8355/8755: Programmable I/O Ports with ROM/EPROM

1. Draw the pin connection diagram of 8355.

Ans. The pin connection diagram of 8355 is shown in Fig. 9c.1.

2. Draw the functional block diagram of 8355 and discuss.

Ans. The functional block diagram of 8355 is shown in Fig. 9c.2.

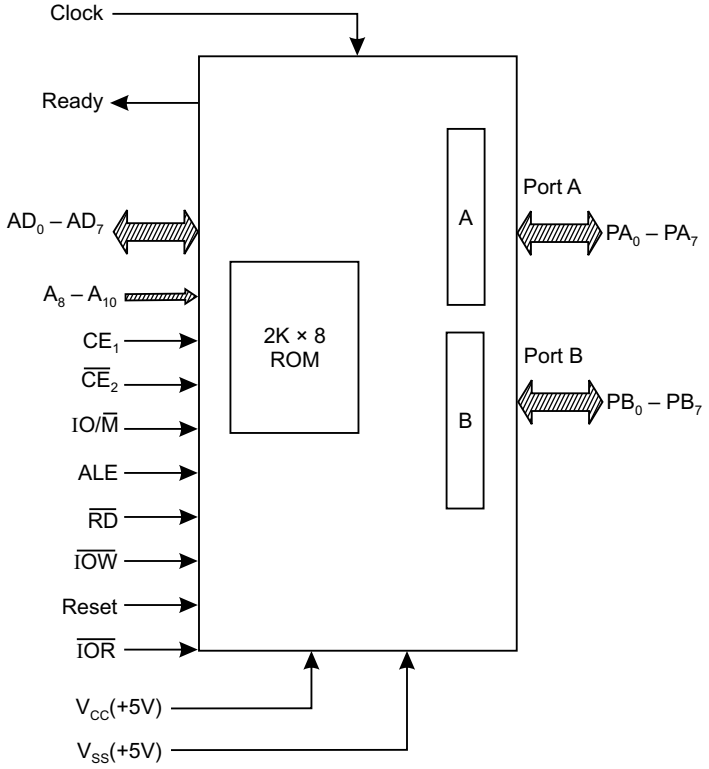


Fig. 9c.2: Functional diagram of 8355
(*Source:* Intel Corporation)

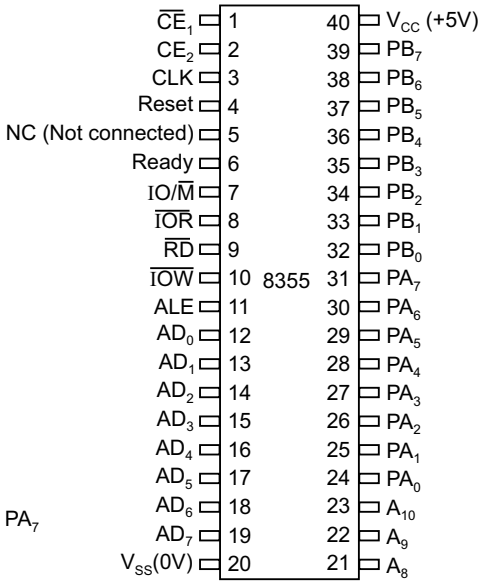


Fig. 9c.1: Pin diagram of 8355
(*Source:* Intel Corporation)

3. What is the difference between 8355 and 8755?

Ans. The 2 KB memory for 8355 is a ROM, while that for 8755, it is EPROM.

Again for 8355, there are two chip enable signals—CE_1 and $\overline{CE}_2$, while for 8755 this signal is designated as CE_2.

Both have two I/O Ports—each I/O line of either port can be programmed either as input or output.

4. What the DDR's do?

Ans. There are two internal control registers, called Data Direction Registers (DDRs)—both the registers are 1-byte in length and designated as DDR_A an DDR_B. Each bit in the two DDR registers control the corresponding bit in the I/O ports. For example bit D_0 of DDR_A controls D_0 bit of Port A and bit D_5 of DDR_B controls D_5 bit of Port B.

5. How the two ports and the two DDRs are selected?

Ans. The bits AD_1 and AD_0 controls/selects one out of the four of the above. This is like this:

AD_1	AD_0		Selected Port or DDR
0	0	⇒	Port A
0	1	⇒	Port B
1	0	⇒	DDR_A
1	1	⇒	DDR_B

The IO/$\overline{M}$ signal is to remain high during the above.

6. How the 2 KB ROM of 8355 is accessed?

Ans. The 2 KB ROM of 8355 is accessed by the $A_{10} – A_0$ latched address in conjunction with a low on IO/$\overline{M}$ signal.

9d

8279: Programmable Keyboard/Display Interface

1. Draw the pin diagram of 8279.

Ans. The pin diagram of 8279 is shown below:

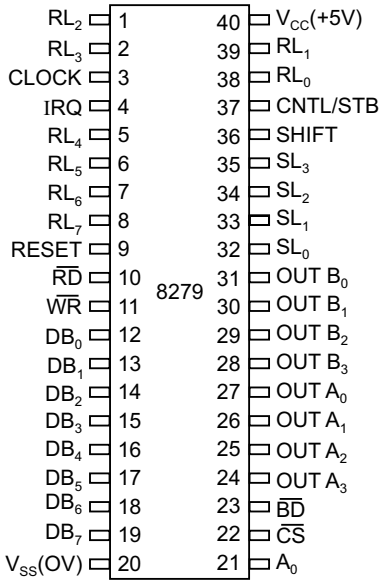


Fig. 9d.1: 8279 pin diagram
(*Source:* Intel Corporation)

2. Draw the functional block diagram of 8279 and elaborate on the different blocks.

Ans. The functional block diagram of 8279 is shown below:

The different functional blocks of 8279 are (*a*) a CPU interface, (*b*) a set of scan lines, (*c*) input lines for key data and (*d*) output lines for display data.

The CPU interface consists of 8-bit data bus along with $\overline{\text{CS}}$, $\overline{\text{RD}}$, $\overline{\text{WR}}$, CLK, RESET and IRQ lines. IRQ is an output line which becomes 1 (active) when key data exists in an internal RAM of 8279. This line is normally connected to one of the hardware interrupt

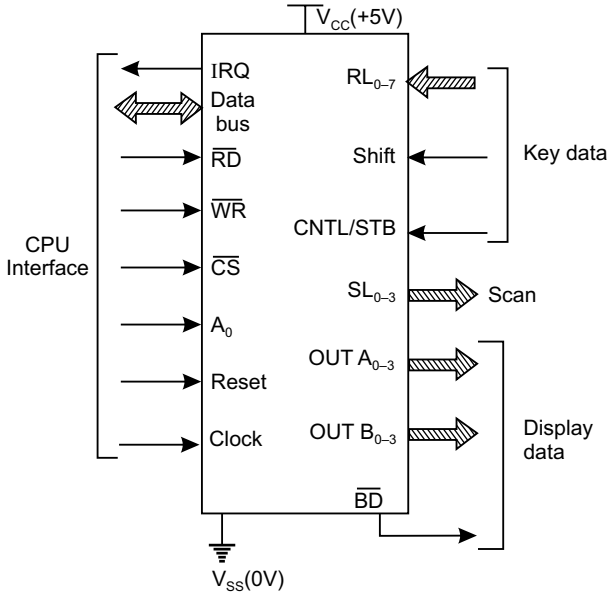


Fig. 9d.2: Functional block diagram of 8279
(*Source:* Intel Corporation)

lines of the CPU. A high on A_0 indicates that the signals in/out pertain to command/status while a low on A_0 indicate that they are data.

The scan lines (SL_{0-3}) along with the eight return lines (RL_{0-7}) can be used for construction of a keyboard matrix. SHIFT and CNTL/STB signals (both inputs) contribute the characteristics for individual keys.

The display output is available through A_{0-3} and B_{0-3} which can be used together as an 8-bit port. $\overline{BD}$ (output signal) is used for display blanking purposes.

3. What are the two most important functions performed by 8279?

Ans. The two most important functions performed by 8279 are as follows:

(*a*) It scans the keyboard, then detects the key press and transmits to the CPU information which corresponds to the particular key pressed.

(*b*) It puts out data received from the CPU, for use by the display devices.

4. What are the various input modes in which 8279 operate?

Ans. There are three input modes in which 8279 operates:

- Scanned Keyboard Mode
- Scanned Sensor Matrix Mode
- Strobed Input Mode.

5. How many character definitions are possible using 8279?

Ans. A maximum of 256 character definitions are possible using 8279.

6. When the CPU is actually involved for the scan and display functions to be realised?

Ans. For the above two functions to be realised, CPU involvement is required only when data is actually transmitted to or received from the CPU.

7. What are the modes in which the four scan lines can operate?

Ans. The four scan lines ($SL_0 - SL_3$) can be operated in two modes—encoded and decoded mode.

8. Discuss the encoded and decoded mode.

Ans. *Encoded mode:* Here 16 lines are generated using the 4 scan lines and a 4 to 16 *external* decoder, although the manufacturers recommend not to use the SL_3 line. Thus eight decoded scan lines are possible with $SL_0 - SL_2$ lines and a 3 to 8 decoder. These 8 lines, along with eight return lines ($RL_0 - RL_7$) can form a 8 × 8 keyboard matrix. Thus it leads to 64 different character definitions. With SHIFT and CONTROL input lines taken as two additional input lines, total character definitions possible = $64 \times 2^2 = 256$.

Decoded mode: Using the internal decoder present in 8279, SL_0–SL_3 lines are decoded. With SHIFT and CONTROL lines along with RL0 – RL7 lines, total character definition possible here is = 4 × 8 × 4 = 128.

9. Describe the Scanned Keyboard Mode.

Ans. Both encoded and decoded scan versions are applicable in this case. This mode can be divided into two ways.

- 2 key lockout
- N-key rollover

In this mode, the pressing of a key generates a unique 6-bit data (called 'position data') which is characteristic of the position of the key pressed. These 6-bits, along with CNTL and SHIFT form a 8-bit word, shown below. Of the position data $D_5 - D_0$, Scan bits correspond to $D_5 - D_3$ and Return bits correspond to $D_2 - D_0$. $D_5 - D_3$ bits correspond to the position of the row on which the key is pressed while $D_2 - D_0$ correspond to the position of the column on which the key is pressed. This 8-bit word gets stored in the RAM of 8279 (in FIFO order) and consequently the IRQ (interrupt request, an output line) line goes high. This IRQ line is connected to one of the hardware interrupt pins of the CPU. On recognition of the interrupt input by the CPU, the RAM in 8279 is read in FIFO form. Once this reading by CPU is over IRQ line of 8279 goes low but will become high if the RAM contains another data.

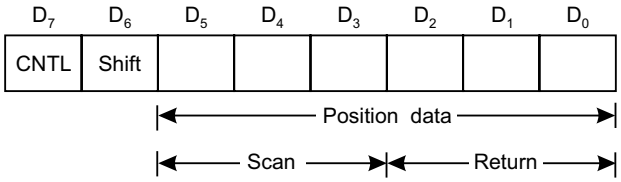


Fig. 9d.3: The scanned keyboard mode format

2 Key Lockout: In this 2 key lock out version of Scanned Keyboard Mode, when any key is pressed, it waits for next two scans to check whether any other key is pressed or not. Several possibilities do arise which need to be addressed separately.

(*a*) No other key press is detected. Then data corresponding to key press is taken to RAM in 8279 and IRQ output line goes into high state.
In case this internal RAM (of 8279) is already full, the keyed data is ignored and the error flag is set (= 1).

(*b*) If one or more additional key pressing occurs, no data entry into RAM is allowed. In this case two possibilities occur:

(*i*) If the first key (i.e., the key which was pressed first) is released ahead of others, then the key press is ignored.

(*ii*) If all the keys are released before the key first pressed, then data corresponding to first key pressed, is entered into RAM of 8279.

Another possibility is pressing of two keys within one debounce cycle (the time required for eliminating contact bounce effect is known as contact debounce time). In this case, no key is recognised. When one key is released, the other key that remains pressed is recognised as a single valid key depression.

N-Key Rollover: In this case, the debounce circuit waits for two scans after the first key press. It then checks whether key is still in the pressed condition or not. If the answer is yes, then the data corresponding to the key press is taken into RAM of 8279. No limit is there to the number of key presses. For simultaneous key presses, data are entered according to the order of key press.

If within a single debounce cycle, two keys are found pressed, the error flag is set and data entry into the RAM is prohibited. The error flag can be read from the FIFO STATUS word and can be cleared by a CLEAR command ($C_F = 1$).

10. Describe the Scanned Sensor Matrix Mode.

Ans. In the Scanned Sensor Matrix Mode of operation, the keys are arranged in the form of a matrix, with the scan lines ($SL_0 - SL_2$) forming the columns and return lines ($RL_0 - RL_7$) forming the rows. The open/closed condition of the key is stored in a RAM location. The size of the matrix be 8 × 8 or 4 × 8 for encoded and decoded scan lines respectively.

The data entering via the RL lines are admitted into eight columns of the sensor RAM—thus each RAM position corresponds to a specific switch position. Apart from switches, other logic circuit output lines can be connected to the RL lines.

11. Describe the Strobed Input Mode.

Ans. In this mode, data are placed on the return lines (RLs). The source of data may be an encoded keyboard or a switch matrix. The data so entering go to FIFO RAM and are accepted on the rising edge of a CNTL/STB pulse.

12. State the options available in the display mode.

Ans. The available options are:

- Display format—either left entry (also known as typewriter mode), or right entry (also known as calculator mode).
- Number of display characters: eight or sixteen.
- Organisation of characters—Single 8-bit or dual 4-bit type.

13. Discuss the Left Entry (Typewriter) Mode of Display format.

Ans. In the left entry (or typewriter) mode, the first entry goes to address 0, the second entry to address 1 and so on. The first entry goes to the left most display position. The second entry to the just right of the earlier one. Thus the 16th entry goes to 15th address position. It is to be remembered that the 17th entry goes to the RAM address 0 again, 18th entry goes to RAM address 1 etc, and is shown in Fig. 9d.4.

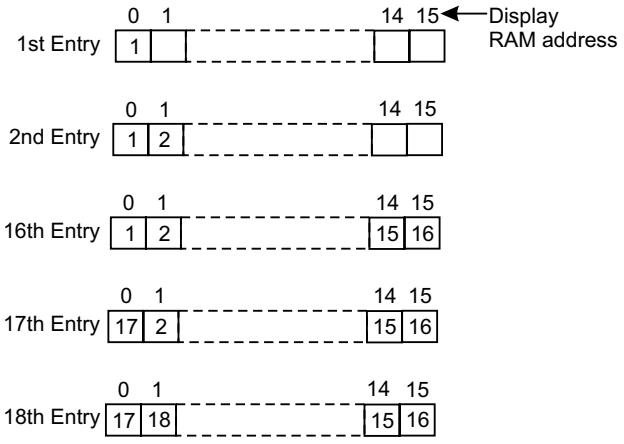


Fig. 9d.4: Left entry mode (Auto-increment)
(*Source:* Intel Corporation)

In this mode, data can be entered at any arbitrary RAM address position. Assuming a 8-position display, if a command 10010111 is inserted after the 2nd entry, then the next data will be displayed at 7th position. The explanation is like this: The most significant three bits 100 represent the code for WRITE display, the next bit, i.e., 1 is for auto-increment and the right most four bits i.e., 0111 (= 7) represent the position at which the next data will be filled in. This is shown in Fig. 9d.5.

14. Discuss the Right Entry (Calculator) Mode of Display format.

Ans. In the right entry (calculator) mode, the characters are entered from the right most position. As characters are entered one after another, the present data occupies the right most position, just the earlier one occupies the left of the right most position etc. This is explained in Fig. 9d.6.

In this mode, no correspondence exists between RAM address and the display position.

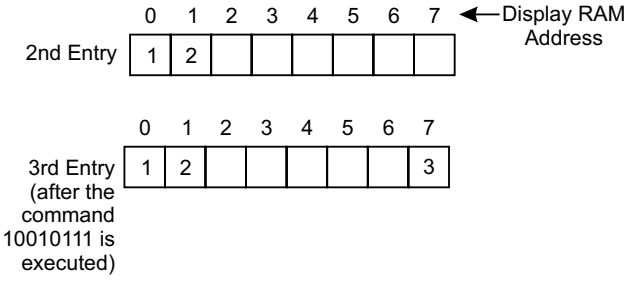


Fig. 9d.5

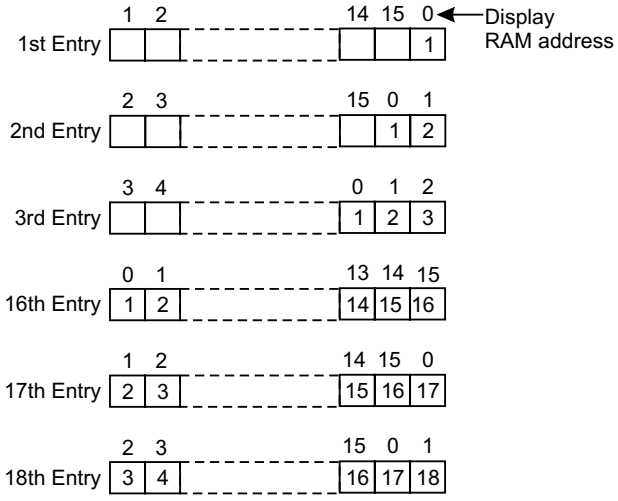


Fig. 9d.6: Right entry mode (Auto-Increment)
(*Source:* Intel Corporation)

15. What are the different types of software operations possible with 8279.

Ans. The following software operations are possible with 8279:

- Keyboard/display mode set
- Program clock
- Read FIFO/Sensor RAM
- Read Display RAM
- Write Display RAM
- Display Write Inhibit/Blanking
- Clear
- End Interrupt/Error Mode Set
- Status Word.

16. In how many ways data can be entered into a microprocessor?

Ans. There are three different ways of entering data into microprocessor—these are

- reading data from a DIP (on/off) switch.
- reading data from push-button keys.
- keys arranged in matrix form and read by software technique.

17. What is meant by contact bounce? How it is eliminated?

Ans. When an electromechanical switch is switched over from an off to on condition, the contact does not become firm on the first count. It loses contact and then makes it—this process repeats itself for a number of times before the contact is firmly placed. This occurs for a very small duration of time.

This thus leads to erroneous operation in digital circuits. This problem can be eliminated by a hardware circuit—called 'contact debouncers' or by software technique (by a delayed reading so that the transient period is over) in microprocessor based systems.

9e

Priority Interrupt Controller 8259

1. Draw the pin diagram of PIC 8259.

Ans. The following shows the pin details of PIC 8259.

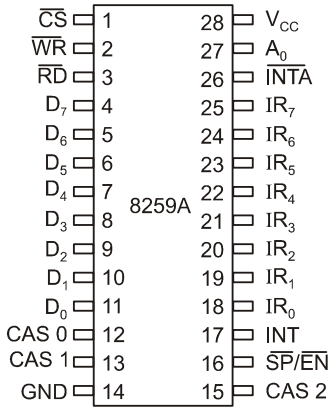


Fig. 9e.1: 8259 pin diagram (*Source:* Intel Corporation)

2. Draw the functional block diagram of PIC 8259.

Ans. The following shows the functional block diagram of 8259.

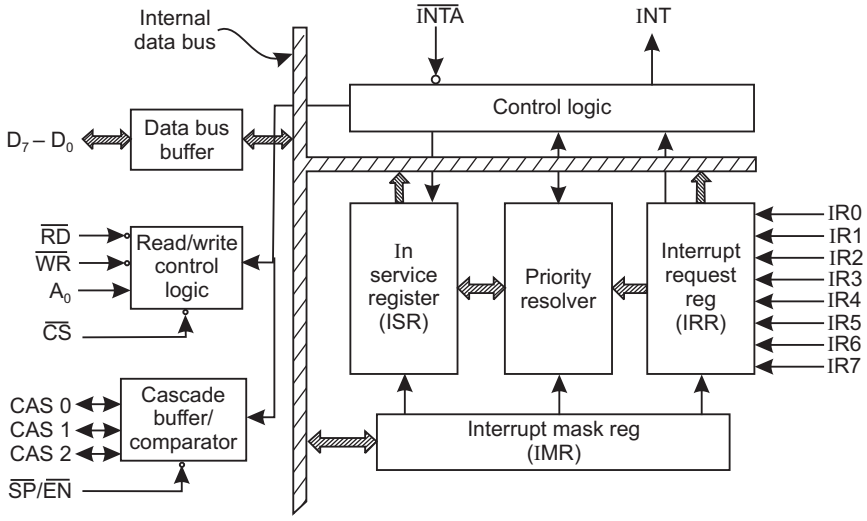


Fig. 9e.2: 8259 Functional block diagram (*Source:* Intel Corporation)

3. Draw the block schematic showing the interconnections between several I/O devices with PIC 8259, μP, RAM, ROM, etc.

Ans. The block schematic of the interconnections is shown below:

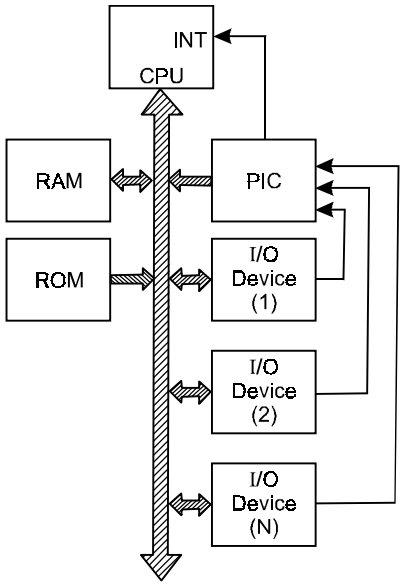


Fig. 9e.3: PIC in an interrupt-driven environment
(*Source:* Intel Corporation)

4. How many interrupt levels can be handled by 8259?

Ans. A single 8259 can handle up to 8 levels of interrupts.
Several 8259's can be cascaded to handle up to 64 levels of interrupts.

5. In how many interrupt modes can 8259 operate and which command word is utilised for this?

Ans. 8259 can be operated in the following categories of interrupt modes:

(*a*) fully nested mode
(*b*) rotating priority mode
(*c*) special mask mode
(*d*) polled mode

Operation Command Word (OCW) is used for 8259 to be operated in the above mentioned modes.

6. How the 8259 is programmed?

Ans. 8259 is programmed by a set of Initialisation Command Words (ICWs). Each 8259 attached to the system must be initialised through ICWs. In all there are four ICWs (ICW1 to ICW4).

7. What are the jobs performed by ICWs?

Ans. ICWs perform the following jobs:

- specifying the vectoring addresses for the individual interrupts.

- specifying single or cascaded mode of operation.
- level or edge triggering mode of operation.

8. What are the jobs performed by OCWs?

Ans. The operation command words (OCWs) are used to operate the PIC 8259 in various interrupt modes like fully nested mode, rotating priority mode, special mask mode and polled mode. The OCWs are also used for masking specific interrupts, status read operations, etc.

9. Describe how the PIC 8259 responds to interrupts?

Ans. PIC 8259 can accept a maximum of 8 interrupts from 8 different I/O devices, resolves the priority with regard to servicing the interrupts and issues an INT output signal, which is connected to INTR input pin (pin 10 of 8085). 8085, in its turn, issues an $\overline{\text{INTA}}$ signal via its pin 11, which is connected to $\overline{\text{INTA}}$ signal of 8259 (pin 26, an input pin for 8259). In response, 8259 puts out a CALL instruction code on the data bus. This is read and decoded by 8085, which then puts out two more $\overline{\text{INTA}}$s. This is done by 8085 to read the address where the Interrupt Service Subroutine (ISS) is written. Now the PIC 8259 puts out the address of this ISS on the data bus which is eventually read by 8085 and the program jumps to the ISS address as was previously programmed by 8259.

10. What are the different functional blocks in 8259?

Ans. PIC 8259 has four different functional blocks viz.,

(*i*) Interrupt and Control logic block
(*ii*) Data bus buffer
(*iii*) Read/Write control logic block and
(*iv*) Cascade buffer/comparator section.

11. What the Interrupt and Control Logic Section consist of?

Ans. This section consists of (*a*) Interrupt Request Register (IRR), (*b*) In Service Register (ISR), (*c*) Priority Resolver, (*d*) Interrupt Mask Register (IMR), (*e*) Control Logic Block.

12. Write down the sequence of operations for programming 8259.

Ans. 8259 is programmed by issuing initialisation command words and operation command words. Initialisation command words are issued in a sequence. The following is the algorithm for initialising 8259.

1. Write ICW1.
2. Write ICW2.
3. If not in the cascade mode of operation, Go To Step 5.
4. Write ICW3.
5. IF ICW4 is not needed, then Go To Step 7.
6. Write ICW4.
7. Ready to accept interrupt sequence.

The flowchart for the above is shown in Fig. 9e.4.

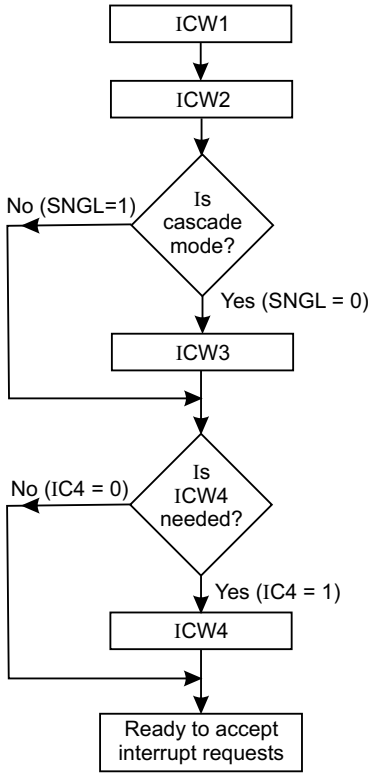


Fig. 9e.4: 8259 initialisation flow chart
(*Source:* Intel Corporation)

13. Write down the main features of 8259.

Ans. The main features of 8259 are as follows:

1. A single 8259 can handle 8 vectored priority interrupts.
2. 9 numbers of 8259 can be cascaded to have 64 levels of vectored priority interrupts in a μC system.
3. The priority modes can be changed or reconfigured dynamically at any time during the main program.
4. It can be operated in various interrupt modes—fully nested, rotating priority, special mask and polled.
5. 8259 can be used with either 8080/8085 or 8086/8088 microprocessor.
6. 8259 supports both edge and level triggered mode of interrupts.
7. The CALL address can be programmed to have a spacing of either 4 or 8 memory locations.
8. The data bus is buffered.
9. The AEOI (Automatic End Of Interrupt) can be programmed.

14. Write down the functions of IR0–IR7 pins.

Ans. There are 8 interrupt input lines from external devices with IR0 having the highest and IR7 the lowest priority. These interrupt requests are acknowledged by 8259 by (*a*) raising the corresponding IR input (i.e., any one of IR0 to IR7) from L to H and holding it high until acknowledged or (*b*) just by a high level.

15. What is the function of $\overline{SP}/\overline{EN}$ pin?

Ans. This is slave program/enable buffer pin of 8259. It has dual functions:

(*a*) In the buffered mode it is used as an output to control the buffer transreceivers (EN). It acts as an output pin to enable the data bus buffer of the system.

(*b*) In the non-buffered mode, it is used as an input pin to designate the 8259 to operate as a master (SP = 1) or slave (SP = 0).

The buffered or non-buffered mode of operation is determined at the time of initialisation of 8259 via ICW4.

16. Describe the functioning of the pins CAS0 – CAS2.

Ans. In a multiple 8259 structure, these three CAS lines form a private 8259 bus.

For a master 8259, these three pins act as output pins, but act as input pins for a slave 8259.

In a multiple 8259 structure, the master 8259 accepts interrupt requests from slave 8259. The master 8259 then generates a CALL opcode in response to the first $\overline{INTA}$. The slave 8259 provides the vectoring address. The master, via its three output pins (CAS0 – CAS2), gives out a code to identify the slave 8259. The slave 8259's, connected to the system, each accepts this code from master 8259 and compare it with the code assigned to it during initialisation. The slave, so identified, then puts out the address of the interrupt service subroutine on to DATA BUS during the second and third $\overline{INTA}$ pulses from the CPU.

The identification of this master and slave 8259's is done by the cascade buffer/ comparator block.

17. Discuss the vector data formats when 8085 is interrupted via INTR.

Ans. There are eight interrupt levels (IR0 – IR7) that generate CALL to eight equally spaced locations in the memory. These eight locations can be programmed to be spaced at an interval of either 4 or 8 memory locations.

The first interrupt vector byte released in response to the first $\overline{INTA}$ is shown in Fig. 9e.5. It is equivalent to the opcode for CALL instruction.

	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
CALL CODE	1	1	0	0	1	1	0	1

Fig. 9e.5: First interrupt vector byte
(*Source:* Intel Corporation)

In response to the second $\overline{INTA}$, the second interrupt vector byte released is shown in figure 9e.6. which shows interval spacings of either 4 or 8.

When the interval spacing is 4, 8259 inserts $D_0 – D_4$ bits automatically as per the levels of interrupts (i.e., IR0 – IR7), while $D_5 – D_7$ bits are specified during programming of 8259 through ICW1. Again for an interval of 8 between consecutive memory locations, $D_0 – D_5$ bits are inserted by 8259 automatically, while $D_6 – D_7$ bits are specified during programming of 8259 through ICW1.

IR	Interval = 4							
	D7	D6	D5	D4	D3	D2	D1	D0
7	A7	A6	A5	1	1	1	0	0
6	A7	A6	A5	1	1	0	0	0
4	A7	A6	A5	1	0	1	0	0
5	A7	A6	A5	1	0	0	0	0
3	A7	A6	A5	0	1	1	0	0
2	A7	A6	A5	0	1	0	0	0
1	A7	A6	A5	0	0	1	0	0
0	A7	A6	A5	0	0	0	0	0

IR	Interval = 8							
	D7	D6	D5	D4	D3	D2	D1	D0
7	A7	A6	1	1	1	0	0	0
6	A7	A6	1	1	0	0	0	0
5	A7	A6	1	0	1	0	0	0
4	A7	A6	1	0	0	0	0	0
3	A7	A6	0	1	1	0	0	0
2	A7	A6	0	1	0	0	0	0
1	A7	A6	0	0	1	0	0	0
0	A7	A6	0	0	0	0	0	0

Fig. 9e.6: Second interrupt vector byte
(*Source:* Intel Corporation)

The following figure shows the format of the byte released by 8259 in response to the third $\overline{\text{INTA}}$.

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8

Fig. 9e.7: Third interrupt vector byte
(*Source:* Intel Corporation)

This byte is completely programmable and is specified by ICW2.

18. Discuss the Initialisation Command Word 1 (ICW1).

Ans. Whenever a Write Command is received with $A_0 = 0$ and $D_4 = 1$, it is interpreted by 8259 to be an initialisation command word ICW1. During ICW1, the following occur:

(*i*) The edge sense circuit is reset, which means that following initialisation, an interrupt request (IR) input must make a L to H transition to generate an interrupt.
(*ii*) The interrupt mask register (IMR) is cleared, i.e., all interrupts are now disabled.
(*iii*) IR7 input is assigned priority 7 (lowest).
(*iv*) The slave mode address is set to 7.
(*v*) The special mask mode is cleared and Status Read is set to IRR.
(*vi*) If D_0 bit in ICW1 (IC4) is set to 0, then all functions selected in ICW4 are set to zero.

The format of the byte to be followed for ICW1 is shown below:

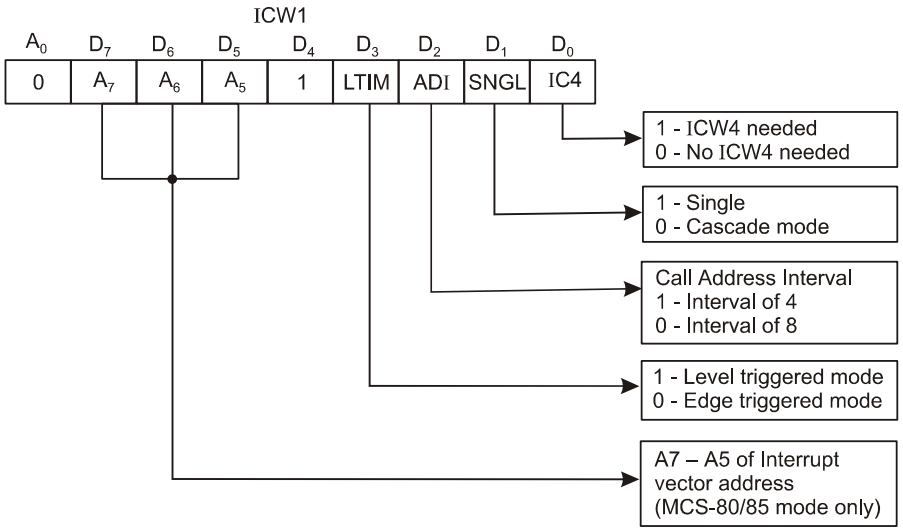


Fig. 9e.8: Initialisation command word 1 (*Source:* Intel Corporation)

- **Bit D_0 (IC4):** It indicates whether ICW4 is needed or not. If it is '1', then ICW4 is needed and if '0', then ICW4 is not needed.
- **Bit D_1 (SNGL):** If this bit is '0', then only one 8259 is in the system and if it is '1' then additional 8259's are there in the system.
- **Bit D_2 (ADI):** ADI stands for 'address interval'. If this bit is '0' then call address interval is 8 and if '1' then call address interval becomes 4.
- **Bit D_3 (LTIM):** This bit determines recognition of the interrupts either in level triggered or edge triggered mode. If this bit = '0', then it is edge triggered mode and if this bit is = '1' then the input interrupts will be recognised if they are in the level triggered mode.
- **$D_5 – D_7$:** These are $A_5 – A_7$ bits as shown under ICW1. For an interval spacing of 4, $A_0 – A_4$ bits are automatically inserted by 8259 while $A_0 – A_5$ are inserted automatically for an interval of 8. $A_5 – A_7$ bits are programmable as set by the bits $D_5 – D_7$ of ICW1.

19. Discuss the Initialisation Command Word 2 (ICW2).

Ans. The initialisation command word 2 i.e., ICW2 is shown below:

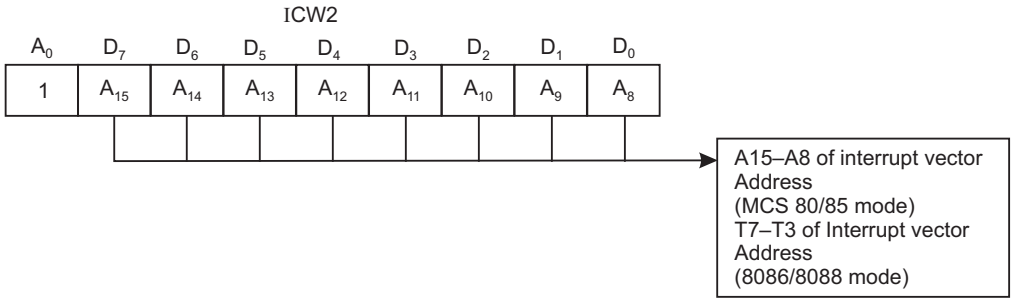


Fig. 9e.9: Initialisation command word 2 (*Source:* Intel Corporation)

If A_0 = 1 of the write command word is issued to 8259, following ICW1, it is interpreted as ICW2. ICW2 is used to load the high order byte of the interrupt vector address of all the interrupts. This byte is common for all the interrupts.

20. Discuss the Initialisation Command Word 3 (ICW3).

Ans. ICW3 can have two modes of operations: Master Mode ICW3 and Slave Mode ICW3. ICW3 is required only if several 8259's are used in the μC system in a cascaded form.

The format of the byte (both for Master Mode ICW3 and Slave Mode ICW3) are shown below:

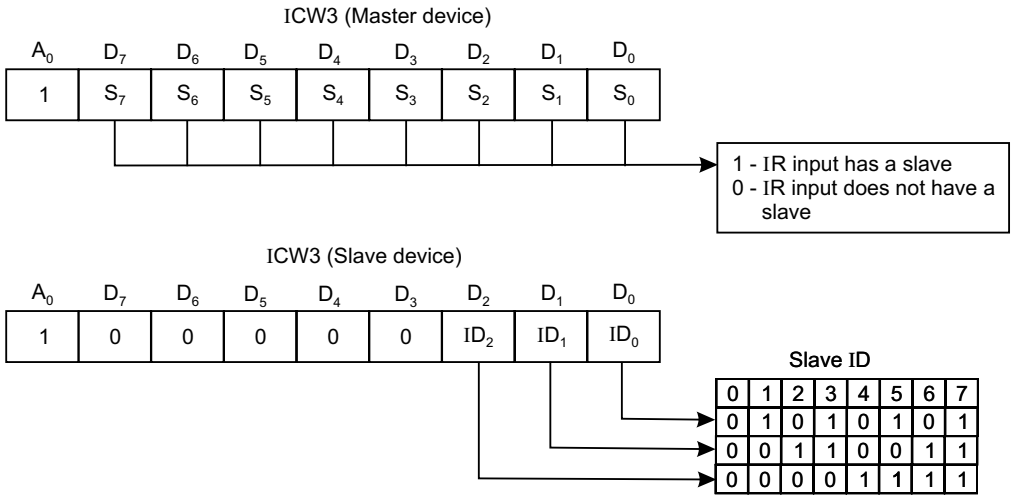


Fig. 9e.10: Initialisation command word 3
(*Source:* Intel Corporation)

Master Mode ICW3: For a 8259 to be treated as a master, we must have $\overline{SP}/\overline{EN}$ pin = 1 in a non-buffered environment and M/S = 1 in ICW4 in a buffered environment.

Then each bit in ICW3 is used to indicate to the master whether it has a slave 8259 attached to it on its corresponding interrupt request (IR) input pin. A '1' indicates the presence of a slave 8259 corresponding to that input and a '0' indicates the absence of a slave.

If now a particular slave 8259 raises its INTR output, then the master generates a CALL instruction opcode and puts out the slave identification number on its output CAS0 – CAS2 lines. This number goes to the slave 8259's via their CAS0 – CAS2 pins (these pins act as input pins for slave 8259's). Thus the number is compared with the individual slave identification number loaded during initialisation. The slave which initially placed the INTR output is thus identified and hence it releases the vector address during the second and third $\overline{INTA}$ cycles.

Slave Mode ICW3: For a 8259 to be treated as a slave, we must have $\overline{SP}/\overline{EN}$ pin = 0 in a non-buffered environment and M/S = 0 in a ICW4 in a buffered environment.

Bits $D_0 – D_2$ of ICW3 (in Slave Mode) assign the slave identification code (Slave ID). The slave ID is equivalent to the master IR input to which the INTR output of the slave is connected. The slave ID compares this number with its own CAS0 – CAS2 inputs so as to release the address vector.

21. Describe the Initialisation Command Word 4 (ICW4).

Ans. The format of ICW4 is shown below. ICW4 is loaded only if D_0 bit of ICW1 (IC4) is set.

As shown in Fig. 9e.11, ICW4 is loaded only if D_0 bit of ICW1 (IC4) is set. The format of ICW4 is shown below.

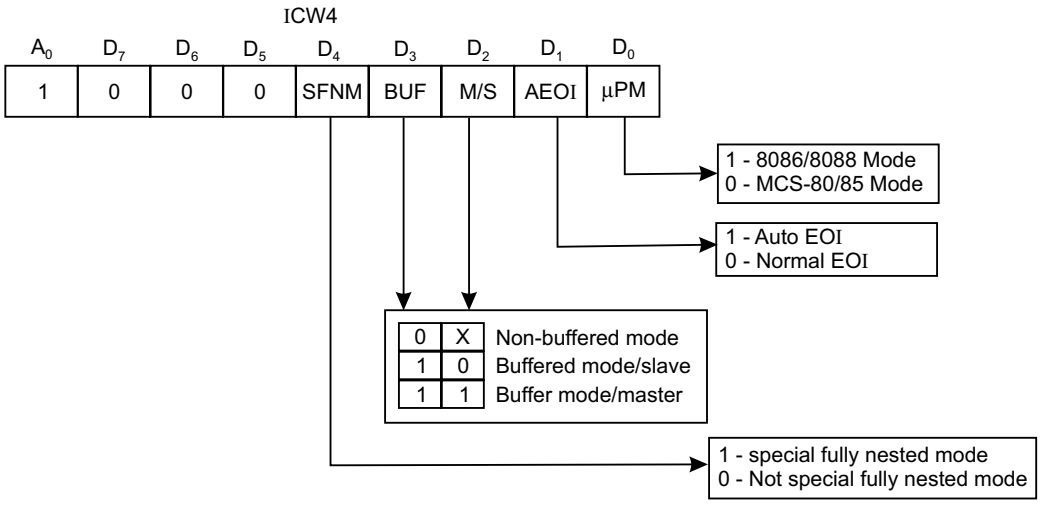


Fig. 9e.11: Initialisation command word 4
(*Source:* Intel Corporation)

The bit positions D_0 to D_4 are now explained:

µPM: This corresponds to D_0 bit position and differentiates between 8086/8088 mode and MCS-80/85 mode.

AEOI: It stands for 'automatic end of interrupt'. If AEOI bit (bit D_1) = 1, then it is in auto EOI mode and if it is = 0, it is in normal EOI mode.

M/S: In the buffered mode, if M/S = 1, then 8259 is initialised as a master and if M/S = 0, then 8259 acts as a slave.

In the non-buffered mode, M/S pin has no significance. In this case, the characteristics of 8259 (i.e., whether 8259 is a master or a slave) is determined by $\overline{SP}/\overline{EN}$ pin.

BUF: This bit position (D_3 bit) determines buffered/non-buffered mode of operation. If BUF = 1, then it is buffered mode of operation and the $\overline{SP}/\overline{EN}$ pin is used as an output to enable the data bus buffer of the system.

SFNM: This stands for Special Fully Nested Mode (bit D_4). If SFNM = 1, then this mode is programmed.

22. Describe the Operation Command Words (OCWs).

Ans. There are three OCWs. These OCWs may be required to change the manner in which the interrupts are to be processed. For this to be achieved, the OCWs may be loaded any time after initialisation is over.

The format for the OCW1 is shown in Fig. 9e.12.

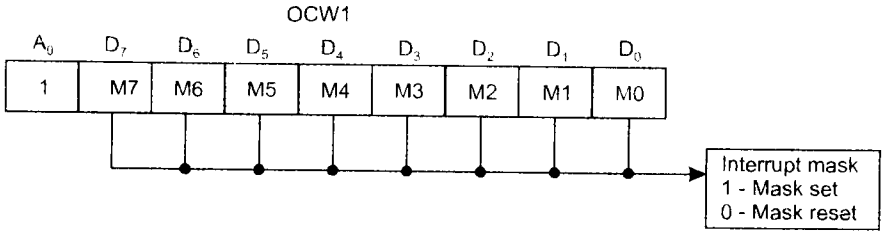


Fig. 9e.12: Format of operation command word 1
(*Source:* Intel Corporation)

OCW1: It is used to enable/disable a particular interrupt request by programming the Interrupt Mask Register (IMR). If M = 1, then the corresponding interrupt is masked and M = 0 indicates its unmasked condition.

A write command with $A_0 = 1$ is interpreted as OCW1, and written after ICW2.

OCW2: The format for OCW2 is shown below:

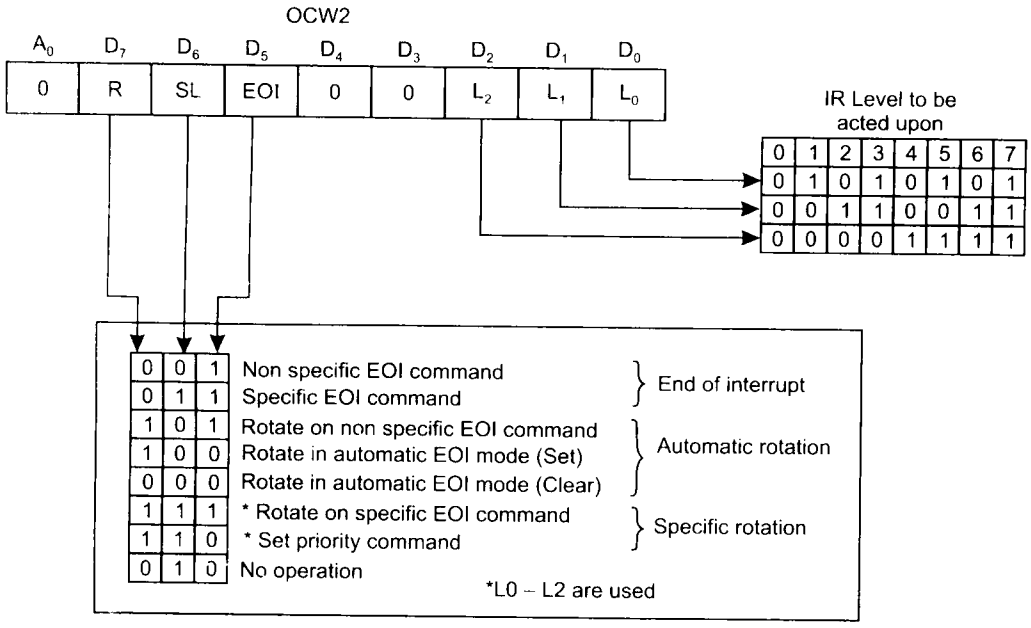


Fig. 9e.13: Format of operation command word 2
(*Source:* Intel Corporation)

A write command with $A_0 = 0$ and $D_4 D_3 = 00$ is interpreted as OCW2. 'R' and 'SL' stand for Rotate and Select Level respectively. The bits corresponding to R, SL and EOI control the Rotate and End of Interrupt modes. Bits $L_2 - L_0$ specify the interrupt level which is to be acted upon when SL is in active condition.

OCW3: The jobs performed by OCW3 are as follows:

(*a*) to read the status of registers.

(*b*) set/reset the Special Mask and Polled modes.

The format of OCW3 is shown Fig. 9e.14.

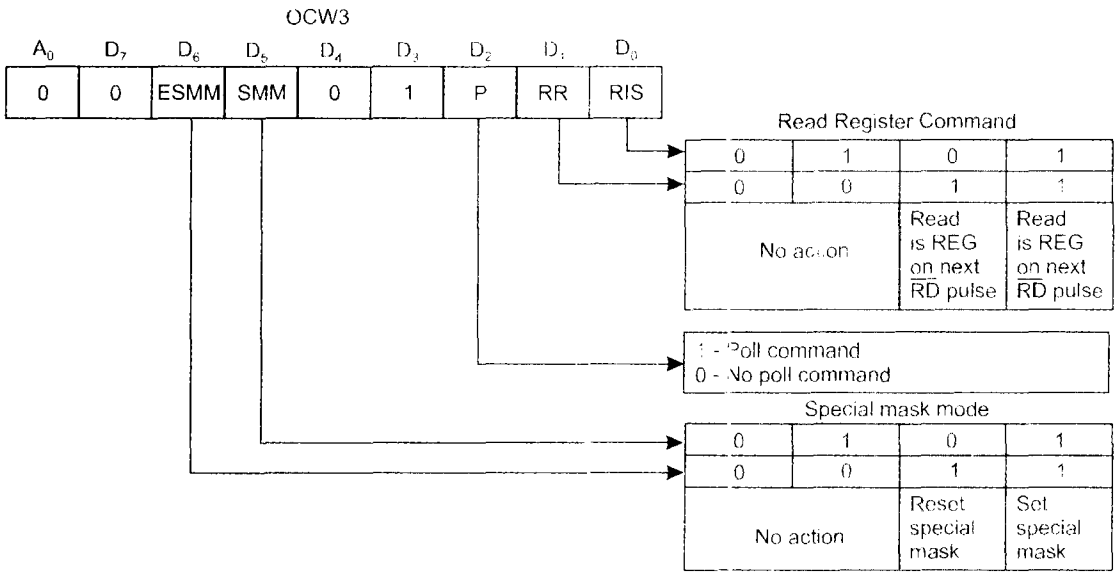


Fig. 9e.14: Format of operation command word 3
(*Source:* Intel Corporation)

23. Describe the FNM (Fully Nested Mode).

Ans. This mode is auto-set after initialisation is over—i.e., it is a default mode. This mode can only be changed through Operation Command Words (OCWs).

IR0 is assigned the highest priority (priority 0) and IR7 the lowest priority (priority 7). When 8259 acknowledges an interrupt request via its INTR pin, it finds out the highest priority and the corresponding bit in the In Service Register (ISR) is set.

The resetting of this bit in ISR can occur in one of the following ways:

(*a*) When the CPU issues an EOI (End of Interrupt) through OCW before coming out of ISR.

(*b*) If AEOI (Automatic End of Request) mode is set in ICW4 during initialisation, the corresponding ISR bit is automatically reset—it occurs on the trailing edge of the last (third) $\overline{\text{INTA}}$ pulse.

When a particular ISR bit is set, (*a*) all lower level interrupts and (*b*) same level interrupts remain in the inhibited condition.

But a higher level interrupt will force the 8259 to generate an INTR, but the same will be acknowledged if the Interrupt Enable F/F has already been enabled via software while the interrupt service subroutine is in progress.

24. Describe the EOI command.

Ans. This command stands for End of Interrupt (EOI).

The interrupt service bit (that is being currently serviced) can be reset by an End of Interrupt Command. This is issued by the CPU, usually just before coming out of the interrupt service routine.

There are two ways in which the EOI can be exerted—in the Fully Nested Mode (FNM) and non FNM.

In the FNM, a *non-specific* EOI command is issued by the CPU. This is an OUT instruction by the CPU to 8259. This is derived by $A_0 = 0$, $D_7\ D_6\ D_5\ D_4\ D_3 = 00100$ with $D_2\ D_1\ D_0$ can have any value. This is apparent from the format of the operation command word 2 i.e., OCW2. This OUT instruction is issued by CPU before exiting from the interrupt service subroutine. On receiving this instruction, 8259 resets the highest level of interrupt (i.e., the current one that which is being serviced).

In non FNM, 8259 cannot determine the last interrupt acknowledged. Thus in this case, the CPU will have to issue a *specific* EOI command signalling out the specific interrupt service bit that is to be resetted. This is done under OCW2 with $A_0 = 0$, $D_7\ D_6\ D_5\ D_4\ D_3 = 01100$ and $D_2 - D_0$ specifies the level on which the EOI command is to act.

25. In the cascade mode, how many EOI commands are to be issued?

Ans. In such a case, two EOI commands must be issued—one for the master and one for the slave.

26. Describe the AEOI command.

Ans. This mode can only be used for a master 8259 and is set by ICW4. 8259 performs a non-specific EOI on the trailing edge of the third (i.e., last) $\overline{\text{INTA}}$ pulse.

27. Explain Special Fully Nested Mode (SFNM).

Ans. In the cascaded mode of operation, if a slave receives a higher priority interrupt request than one which is in service (through the same slave), it would not be recognised by the master. This is because the master ISR bit is already in the set condition, thereby it ignores all requests of equal or lower priority. The higher priority interrupt won't be serviced until after the master ISR bit is reset by an EOI command. This is most likely to happen after the completion of the lower priority routine.

This is where the SFNM comes into. It is meant only for the master and done during master initialisation (through ICW4). In this mode, the master will ignore interrupt requests of lower priority, but responds to requests of equal or higher priority.

The following are the differences between FNM and SFNM:

(*a*) In SFNM, the slave is allowed to place an interrupt request (of higher priority than the one currently being serviced). The master recognises this higher level interrupt, which in its turn places this interrupt request to the CPU.

(*b*) In SFNM, the software must determine if any other slave interrupts are pending before issuing an EOI command to the slave and then reading its ISR (In Service Register). If the ISR contains all zeroes, then no interrupt from the slave is in service and an EOI command can be sent to the master. If the ISR is not all zeros, an EOI command should not be sent to the master. Clearing the master ISR bit with an EOI command while there are still slave interrupts in service would allow lower priority interrupt to be recognised by the master.

28. Mention the types of Rotating Priority Mode of interrupt.

Ans. The Rotating Priority Mode can be set in

(*a*) Automatic Rotation

(*b*) Specific Rotation.

29. Discuss the Automatic Rotation.

Ans. In situations where several communicating channels are connected to μC system, all the channels should be accorded equal priority in sharing information with the μC.

Thus when a peripheral is serviced, all other equal priority peripherals should be given a chance to be serviced before the original peripheral is serviced a second time around. This is accomplished by automatically assigning a peripheral the lowest priority after being serviced. Thus a device, presently being serviced, would have to wait until all other devices are serviced.

Automatic rotation is of two types:

(*a*) Rotate on non-specific EOI Command.

(*b*) Rotate on automatic EOI Mode.

Rotate on non-specific EOI Command: When the rotate on NSEOI command is issued, the highest ISR bit is reset and the corresponding IR level is assigned the lowest priority.

Let IR0 has the highest and IR7 the lowest priority. Let also that IR6 and IR4 are in service with IR4 accorded the highest priority. Bit 4 in the ISR is reset when a NSEOI command is executed. After this, IR4 becomes the lowest priority and IR5 becomes the highest priority. The situations are explained in the following figure with the left side indicating the situation before the command is executed and the right side after the command execution.

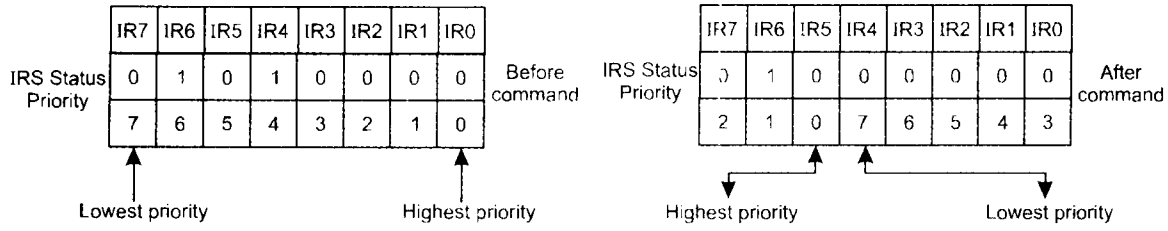


Fig. 9e.15: Rotation on non-specific EOI command

Rotate in Automatic EOI Mode: This mode works much like the rotate on NSEOI command. The main difference between the two lies in the priority routine done automatically after the last $\overline{\text{INTA}}$ pulse of an interrupt request. To enter or exit from this mode, a rotate-in-automatic EOI Set Command and rotate-in-automatic EOI Clear Command is provided.

30. Explain Specific Rotation.

Ans. In this mode, the lowest priority can be assigned to any of the IR levels (between 0 and 7) as specified by OCW2.

This mode is set by CPU by issuing an OUT instruction in the following manner:

$$A_0 = 0$$

$$D_7\ D_6\ D_5\ D_4\ D_3 = 1\ 1\ 0\ 0\ 0$$

$D_2\ D_1\ D_0$ bits specify which interrupt level is to be accorded lowest priority. This mode is independent of EOI command.

31. Describe Special Mask Mode.

Ans. The special mask mode enables interrupts from all levels except the level presently in service. This is done by masking the level that is in service and then issuing the special mask mode command. Once the special mask mode is set, it remains in effect until reset.

The Special Mask Mode can be set by making ESSM and SMM bits '1' in OCW3. When a mask bit is set in OCW1, all further interrupts at that level are inhibited, while interrupts on all other levels that have not been masked by OCW1 (both lower and higher) are enabled. Thus it is possible to selectively enable interrupts by programming the mask register.

The special mask mode is cleared by loading OCW3 as follows:

ESSM = 1
and SMM = 0

32. Describe the Polled Mode interrupt scheme.

Ans. In this mode, the interrupting devices seeking services from 8085 are polled one after another to detect which device has sought for interrupt request. The INT output of 8259 is either not connected to INTR input of 8085 or the interrupts are disabled by software means.

Bit D_2 (i.e., P) in OCW3 is set to '1' in the Polled mode. 8259 is then read by masking its $\overline{RD}$ and $\overline{CS}$ pins '0'. The ISR bit is set corresponding to the highest level interrupt in the IRR. A byte is put out on the data bus as shown below:

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	–	–	–	–	W_2	W_1	W_0

Fig. 9e.16: Polled mode output word
(*Source:* Intel Corporation)

If $D_7 = 1$, then it implies that an interrupt needs servicing, while $D_2 - D_0$ bits (i.e., $W_2 - W_0$) correspond to the highest priority interrupt level which is requesting service.

Since INTR line is not in use in the polled mode, hence more than one 8259 may be connected in the master mode. Hence it is possible to have more than sixty four levels of interrupts in this mode.

33. On which registers status read operations can be done?

Ans. Status read operations can be done on Interrupt Request Register (IRR), In-Service Register (ISR) and Interrupt Mask Register (IMR).

34. How IRR status read operation is done?

Ans. A particular 8259 can be set up for an Interrupt Request Register (IRR) read operation by inserting in OCW3 RR (read register) = 1 and RIS (Read ISR) = 0. Following this, $\overline{RD}$ = 0 and $\overline{CS}$ = 0 are made on the 8259, which thereby puts out the contents of IRR on the data bus.

In the non-polled mode of 8259, A_0 should be made '0' so as to put out the contents of IRR status word after the IRR has been set for status read operation.

35. Discuss the ISR status read operation.

Ans. A particular 8259 can be set up for an In-Service Register (ISR) read operation by inserting in OCW3 RR = 1 and RIS = 1. Following this $\overline{RD}$ = 0 and $\overline{CS}$ = 0 are made on the 8259,

which thereby puts out the contents of ISR on the data bus.

In the non-polled mode of 8259, A_0 should be made '0' so as to put out the contents of ISR status word after the ISR has been set for status read operation.

36. How IMR status read operation is done?

Ans. An Interrupt Mask Register (IMR) status read operation on a 8259 is done with $A_0 = 1$, $\overline{RD} = 0$, $\overline{CS} = 0$. This causes the contents of IMR to be put out on the data bus. For an IMR status read operation, OCW3 is not needed.

37. Which status read operation is performed by default after initialisation of 8259?

Ans. The default status read operation is the IRR status read, after the initialisation of 8259.

38. Mention when status read operation is not possible.

Ans. When OCW3 is set in the polled mode with P = 1 and RR = 1, status read operation is not possible.

39. Discuss the Default IR7 routine.

Ans. An interrupt via the INTR input will be treated as a valid interrupt if it remains high until after the falling edge of the first $\overline{INTA}$ pulse. Then a valid IR7 input occurs resulting in a *normal* CALL to the IR7 routine. If it is otherwise, a *default* CALL to the IR7 routine will be generated.

A normal IR7 operation sets the ISR bit while a default IR7 operation does not do so. For this IR7 is normally used for RET instructions. If IR7 is to be utilised for other purposes, then the default IR7 operation is first to be checked. This is done by a status read operation of ISR at the beginning of interrupt service routine for IR7. If after this read operation, IR7 input = 1, then it is a valid interrupt, otherwise not.

40. Distinguish between NSEOI, SEOI and AEOI.

Ans. These three stand for non-specific end of interrupt, specific end of interrupt and automatic end of interrupt.

An NSEOI command sent from the μP lets the 8259 know when a service routine has been completed, but without any specification of its *exact* interrupt level. The 8259 determines the interrupt level (the highest priority interrupt in service) and resets the correct bit in the ISR.

The NSEOI is best suited when servicing is always at the highest priority level. When 8259 receives a NSEOI command, it simply resets the highest priority ISR bit.

The main advantage of this mode is that it is not necessary to specify the IR level. NSEOI command is not suited for the following two cases:

(*a*) Using a set priority command within an interrupt service routine.
(*b*) Using a Special Mask Mode.

A SEOI command sent from μP to 8259 lets it know when a service routine of a *particular* interrupt level is completed. A SEOI command resets a specific (particular) ISR bit—thus any one of the eight IR levels can be specified.

A SEOI command is needed when 8259 is unable to determine the IR level.

A SEOI command is best suited for situations in which priorities of the interrupt levels are changed during an interrupt routine (Specific Rotation).

The AEOI mode scores over the EOI modes in that no command has to be issued in

AEOI mode. Thus AEOI mode greatly simplifies programming and lowers code requirements within interrupt routines.

AEOI mode should be used continuously because the ISR bit of a routine presently in service is reset right after its acknowledgement. It thus leaves behind nothing in the ISR about which particular bit is being serviced. If any interrupt request occurs during this time, it will be serviced (provided all interrupts are enabled) regardless of its priority—whether low or high.

Another peculiar problem—called 'over nesting' may happen in this case. It occurs when an IR input keeps as interrupting its own routine. It results in unnecessary stack pushes which could fill up the entire stack in a worst case situation.

41. Describe how several PICs are cascaded together.

Ans. Several PICs can be cascaded together—in all 9-with one PIC acting as the master and the rest eight as slaves and shown in Fig. 9e.17. $\overline{SP}$ pin of the master is connected to V_{cc}, whereas for slave PICs, their $\overline{SP}$ pins are connected to ground. The INT outputs of the slave PICs are connected to one of IR0-IR7 pins of the master. The registers within the PICs are allocated separate addresses by using separate $\overline{CS}$ signals. Initialisation of each of the PICs are done separately.

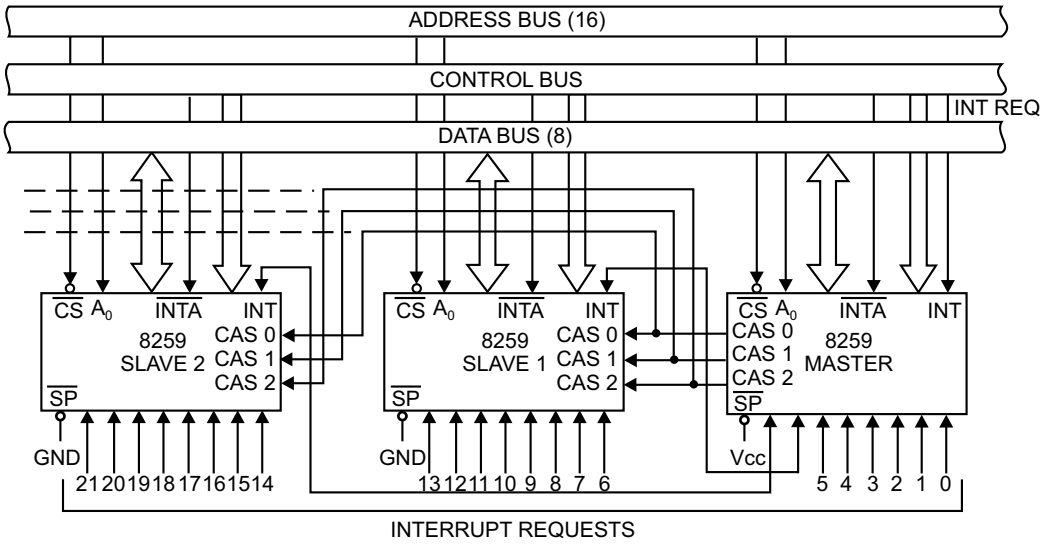


Fig. 9e. 17: Cascading of 8259 PICs

When an interrupt comes, it activates one of the IR input lines of a slave PIC and this in turn activates one of the IR lines of the master (via the INT output pin of the slave PIC which has been interrupted). This in turn interrupts the CPU. The INTA output from the master enables the corresponding CAS0-CAS2 lines of the slave PIC—this releases the vector address of the data bus in the second and third INTA cycles.

9f

Programmable DMA Controller (DMAC) 8257

1. Draw the pin diagram of 8257.

Ans. The following figure gives the pin connection diagram of 8257.

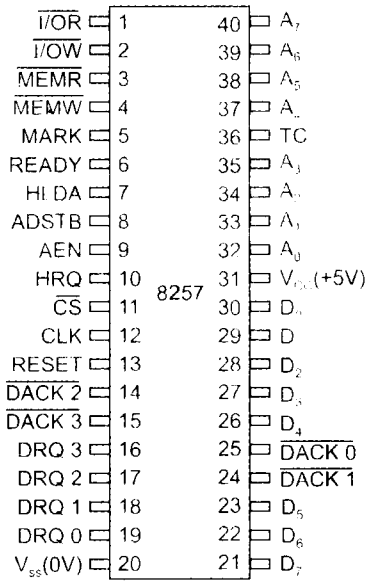


Fig. 9f.1: 8257 pin diagram (*Source:* Intel Corporation)

2. Draw the functional block diagram of 8257.

Ans. The following figure shows the functional block diagram of the DMAC 8257.

As is apparent from the figure, it consists of eight blocks: data bus buffer, read/write block, control logic and mode set register, priority resolver and four channel blocks.

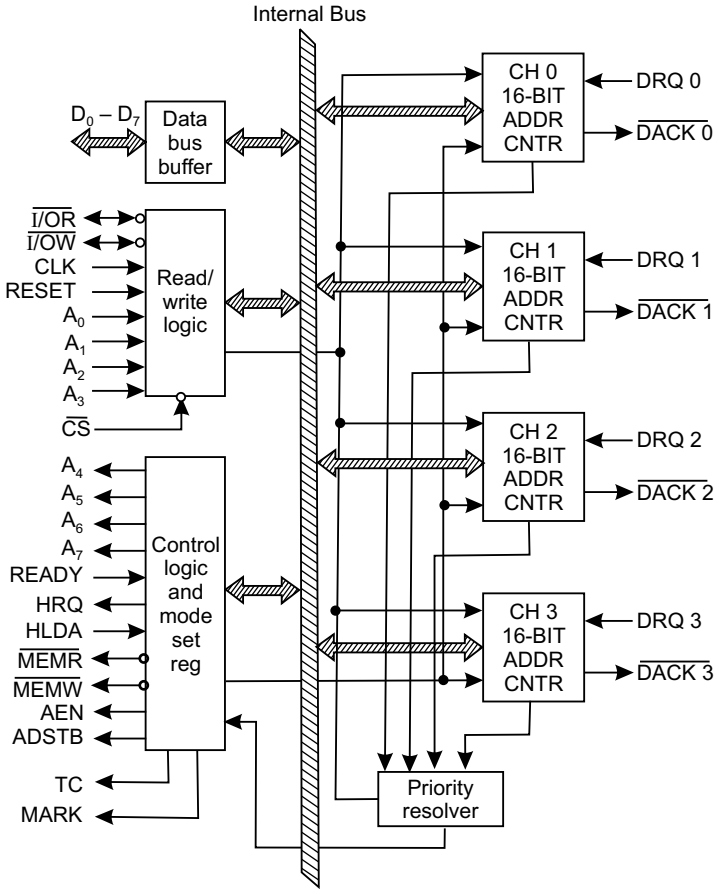


Fig. 9f.2: 8257 Functional block diagram (*Source:* Intel Corporation)

3. Describe the general features of 8257.

Ans. The general features of 8257 are as follows:

1. It is a 4-channel Direct Memory Access (DMA) interface IC which allows data transfer between memory and up to 4 I/O devices, bypassing CPU.
2. A maximum of 16 KB of data (= 2^{14}) can be transferred by this IC sequentially at a time. When a DMA request comes from a peripheral, the DMAC 8257, via its HRQ (Hold Request) pin (pin number 10, which is an output pin), requests the CPU on its HOLD pin (pin number 39 of CPU 8085). CPU then acknowledges this request via its HLDA (pin 38) pin which goes to HLDA pin (pin 7) of 8257. After this, DMAC generates the required $\overline{\text{MEMR}}$, $\overline{\text{MEMW}}$, $\overline{\text{I/OR}}$, $\overline{\text{I/OW}}$ signals through its 1-4 pins.
3. Initialisation of the DMAC is done under program control for each channel. The parameters which need to be initialised for each channel are starting address, number of bytes of data to be transferred, mode of operation, etc.
4. DMAC can be operated in three modes: (*a*) DMA Read (reading from memory, writing into peripheral), (*b*) DMA Write (writing into memory, reading from peripheral), (*c*) DMA verify.
5. Priority for each of the 4 channels can be set in (*a*) fixed priority, (*b*) rotating priority.

6. A Terminal Count Register exists for each of 4 channels. The number of bytes of data to be transferred is stored in the D_{13}–D_0 positions of the 16-bit Terminal Count Register. On completion of data transfer, the Terminal Count (TC) (pin 36, an output pin) goes high.

4. How many I/O devices can 8257 access?

Ans. Up to 4 I/O devices can be accessed by 8257.

5. What is the maximum value of KB of data that 8257 can transfer?

Ans. 8257 can transfer a maximum of 16 KB (16,384 = 2^{14}) of data.

6. What are the modes of operation of 8257?

Ans. 8257 can be operated in three modes. These are:
- DMA read
- DMA write and
- DMA verify

7. Comment on priority when 8257 services the external I/Os for data transfer.

Ans. Priorities, to service the external interrupts, can be
- fixed priority.
- rotating priority.

In the Mode Set Register (it is also called control register) of 8257, D_4 bit corresponds to 'enable Rotating Priority' bit.

If this bit is not set, then CH–0 is assigned highest priority and CH–3 lowest priority—this is by default. If D_4 is set to '1', the channel that has just been serviced is allocated the lowest priority. The DMA operation always assigns highest priority to CH–0.

If more than one channels are enabled, and they all place the request for DMA transfer, consecutive DMA cycles service different channels, beginning with CH–0.

8. What are the registers available with each channel of 8257?

Ans. The registers which are available with each channel of 8257 are:
- An Address Register (16-bit)
- A Terminal Count Register (TCR) (16-bit)

9. How the 8257 is initialised?

Ans. The 8257 is initialised by the CPU
(*a*) by loading the starting address of a DMA block for an I/O device in the 16-bit address register.
(*b*) by loading $D_{13} - D_0$ bits i.e., lower 14-bits of Terminal Count Register (TCR) with the number of bytes of data to be transferred.
(*c*) by loading D_{15} and D_{14} of TCR appropriately to set the mode of operation of 8257.
(*d*) by loading the Mode Set Register appropriately.

10. When does the status of pin 36 (TC = Terminal Count) of 8257 go high?

Ans. Pin 36 of 8257, which is the TC pin (an output pin, active high type) is raised high ('1' state) when the contents of TCR of the selected channel become zero.

11. What are meant by an enabled and a disabled peripheral?

Ans. The peripherals which are granted DMA transfer are called enabled peripherals and the

ones who are denied DMA transfer are called disabled peripherals. The above is done by Mode Set Register.

12. What are the jobs that are performed by 8257 sequentially, when it receives a request from an enabled peripheral?

Ans. 8257 does the following jobs sequentially, when it receives a request from an enabled peripheral:

(*i*) Gains control of the system buses, once the HOLD signal issued by 8257, is acknowledged by 8085.

(*ii*) 8257 sends an acknowledgement signal to the peripheral which is currently having the highest priority.

(*iii*) The lower 8-bits of the memory address are put out as $A_0 - A_7$ pins. These are connected to the $A_0 - A_7$ lines of the system bus. The most significant 8-bits of the memory address are put via the data bus lines $D_0 - D_7$. These are latched by 8212 latch which places them on the system address bus $A_8 - A_{15}$.

(*iv*) I/O read/write and memory read/write signals are generated.

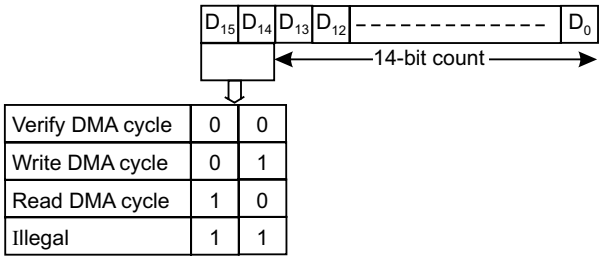
13. Which pin of 8257 acts in a similar fashion as the ALE (address latch enable) pin of 8085? Elaborate.

Ans. Pin 8 which is ADSTB (address strobe) is an active high output pin and functions in the same manner as the ALE pin of 8085.

At the beginning of each DMA cycle, 8257 puts the most significant byte of the DMA address register on its $D_0 - D_7$ pins. These are latched by 8212 latch using the ADSTB strobe pulse of 8257. Thus at the end of this cycle, the $D_0 - D_7$ pins of 8257 can be used for data transfer purpose.

14. Where the number of bytes of data, to be transferred by DMA mode, are stored?

Ans. The number of bytes of data to be transferred by DMA mode are stored in $D_0 - D_{13}$ bits of Terminal Count Register (TCR). They are loaded with a value which is the required number of DMA cycles minus one.



Verify DMA cycle	0	0
Write DMA cycle	0	1
Read DMA cycle	1	0
Illegal	1	1

Fig. 9f.3: The terminal count register (TCR)

As shown, the two bits D_{15} and D_{14} together are loaded to set the mode of operation for that channel—like DMA write or read cycle, etc.

15. With regard to data transfer, how many classes of DMA are possible?

Ans. With regard to data transfer under DMA control, two classes are possible:

(*i*) *Sequential DMA:* In this, the DMA controller reads a data byte from memory and then writes the same into I/O or vice-versa. For each of these read or write operations, 2 to 4 CLK cycles are required.

(*ii*) *Simultaneous DMA:* It is the fastest transfer process. Here Read and Write operations are performed at the same time. Thus both $\overline{\text{MEMR}}$ and $\overline{\text{IOW}}$ (or $\overline{\text{IOR}}$ and $\overline{\text{MEMW}}$) are active at the same time. Thus a speed improvement of twice the sequential DMA class is possible in this case.

When bulk data is to be transferred, (*i*) is used while (*ii*) is used for moderate data transfer.

16. Why DMA mode of data transfer scheme is the fastest?

Ans. In normal data transfer schemes, a data coming from an I/O is first taken to ACC of the CPU and then stored in memory.

But in DMA scheme, straightway data exchange takes place between memory and I/O, bypassing the ACC of CPU. Since ACC of CPU does not take part (it is absent) in DMA mode of data transfer scheme, thus this mode is fastest.

17. What are the basic building blocks of 8257?

Ans. The basic building blocks are as follows:

DMA channels, Data bus buffer, Read/Write block ($\overline{\text{I/OW}}$, $\overline{\text{I/OR}}$, $\overline{\text{CS}}$, Reset, CLK, $A_0 - A_3$), control logic block (HRQ, HLDA, $A_4 - A_7$, $\overline{\text{MEMW}}$, $\overline{\text{MEMR}}$, Ready, ADSTB, AEN, TC, MARK) and Mode Set Register.

18. Draw the basic flowchart of a DMA mode of data transfer scheme.

Ans. The flowchart will be, as shown below:

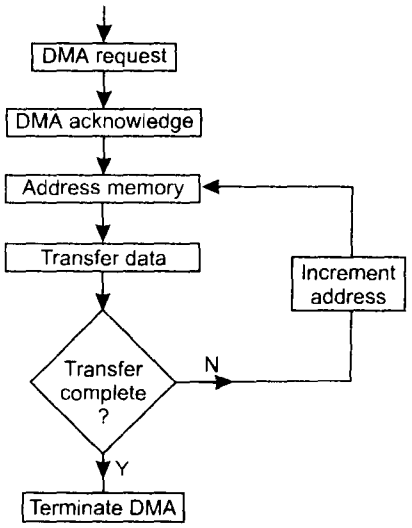


Fig. 9f.4: Flowchart of DMA

19. What is meant by a DMA cycle?

Ans. A DMA cycle indicates the transfer of a byte.

20. Through which pin a peripheral requests the 8257 for data transfer and through which pin the peripheral gets back the acknowledgement.

Ans. A peripheral requests for data transfer to 8257 via DRQ (DRQ 0 – DRQ 3) pin and it gets back the acknowledgement via $\overline{\text{DACK}}$ ($\overline{\text{DACK 0}}$ – $\overline{\text{DACK 3}}$) pin.

21. What determines the master or slave action of a DMA controller.

Ans. When the μP is in control of its buses (address bus, data bus and control bus), it acts as master and DMA controller acts as the slave. When DMA controller takes control of the buses, it becomes the master and μP becomes the slave.

22. What are the functions of Mode Set Register of 8257 DMA controller?

Ans. The functions of the Mode Set Register are as follows:

(*i*) To enable/disable a channel or channels.

(*ii*) To configure 8257 in the following four categories: Auto Load, TC Stop, Extended Write, Rotating Priority.

23. When DMA is undertaken, with whom the peripheral is synchronised?

Ans. When DMA is in progress, the peripheral is synchronised to the main memory, not the microprocessor.

24. When the DMA request line (i.e., HOLD pin of microprocessor) is sampled?

Ans. It is sampled at the end of each machine cycle and not instruction cycle.

Thus the response time for a DMA request is a maximum of one machine cycle plus one T state. For 8085 or Z–80 microprocessors the worst case scenario, i.e., maximum time is 7 (seven) T states.

25. When a non-maskable interrupt is not going to be recognised by a microcomputer system?

Ans. No interrupts—either maskable or non-maskable—will be recognised during a DMA request.

26. In how many modes DMA transfer is possible?

Ans. There are three modes. These are: (*a*) Byte or Single mode, (*b*) Burst or Demand mode, (*c*) Continuous or Block mode.

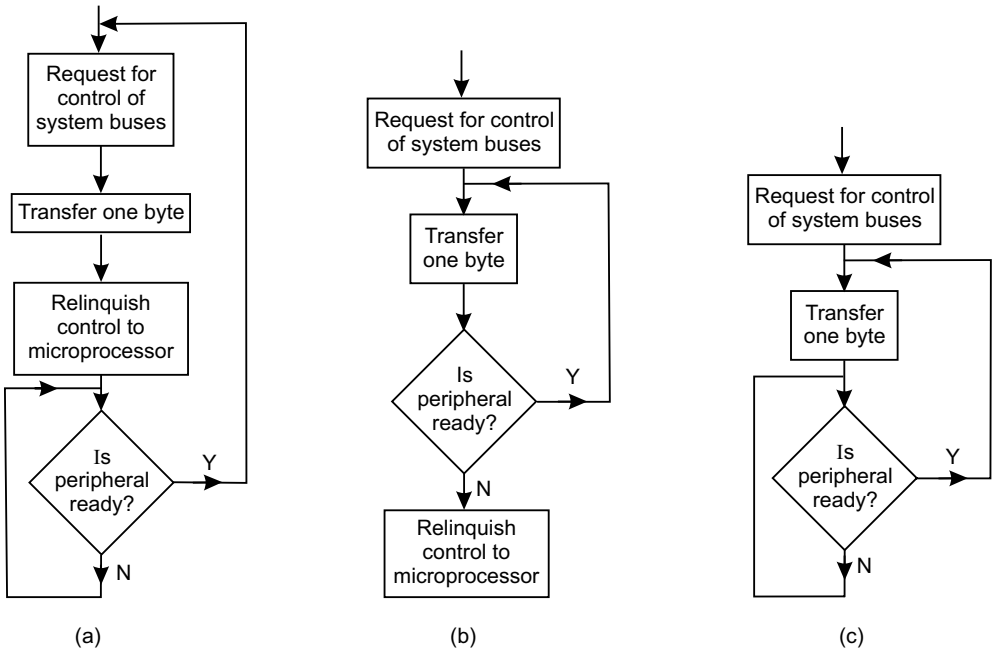


Fig. 9f.5: Flowcharts for (a) Byte or single mode (b) Burst or demand mode and (c) Continuous or block mode

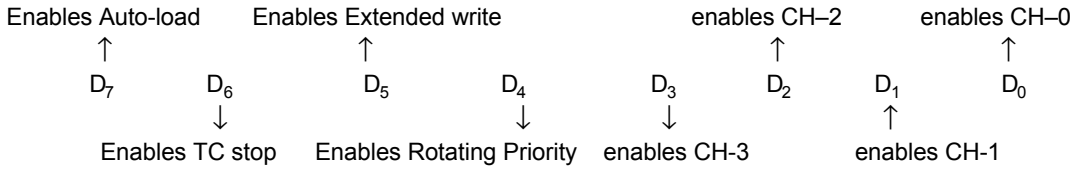
In byte or single mode, after transferring one byte of data, the bus control is relinquished and handed over to microprocessor.

In burst or demand mode, data is transferred till the time the peripheral is 'ready'. After this the bus control is handed over to microprocessor.

The third method, i.e., continuous or block mode, is identical to the earlier one, but the bus control is not relinquished until the entire block of data has been transferred.

27. Which particular register is responsible for enabling (or disabling) a particular channel?

Ans. It is the shift mode set register which is responsible for enabling (or disabling) of a particular channel (CH0 to CH3). The mode set register is defined as:



For example, if channel 1 (CH–1) is to be selected (enabled), along with TC stop option facility, then the mode set register, as per above, should have the following content:

0 1 0 0 | 0 0 1 0 = 42 H

28. How the port addresses of the registers in 8257 are done?

Ans. The port addresses of each register in 8257 are determined by the four address lines A_3–A_0. The port assignments for mode register, CH–1 DMA register and CH–1 count register are given below:

		A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0		
Mode register	⇒	1	0	0	0	1	0	0	0	=	88_H
CH–1 DMA register	⇒	1	0	0	0	0	0	1	0	=	82_H
CH–1 count register	⇒	1	0	0	0	0	0	1	1	=	83_H

It is assumed that $\overline{CS}$ signal is connected to address line A_7 via an inverter and $A_6 - A_4$ are all at logic level 0.

29. Write a program to transfer AD_H bytes of data from a peripheral to memory, the memory address starting from 2459_H. Data to be inputed via CH–1.

Ans. The instructions can be sequentially written as follows:

MVIA, 42_H
OUT 88_H
} Enabling of CH–1 and sending the same to mode register.

MVIA, AD_H
OUT 83_H
} Send the low order byte count AD_H to be transferred to Terminal Count Register of CH–1

MVIA, 40_H
OUT 83_H
} Send high order byte count for 'Write DMA cycle' for which
$D_{15} = 0$, $D_{14} = 1$
and $D_{13} - D_8 = 0$

MVIA, 59_H
OUT 82_H
} Send the low-order byte (59_H) of the memory location 2459_H to the DMA register of CH–1

MVIA, 24_H
OUT 82_H
} Send the high-order byte (24_H) of the memory location 2459_H to the DMA register of CH–1

30. Which particular pin of 8257 is used to convert it into 'master' mode and MPU in 'slave' mode?

Ans. AEN (Address enable) pin of 8257 is utilised to convert it into 'master' mode and at the same time translate MPU into 'slave' mode.

31. Which particular pin of 8257 is used to interface it with a slow memory?

Ans. 'READY' pin of 8257 is used for the purpose. When the memory becomes ready, it sends a high signal which is connected to the 'READY' pin of 8257.

32. When 8257 is programmed to have fixed priority, which channel will have the lowest priority?

Ans. It is the CH–3 which will have lowest priority.

33. When data is transferred from memory to an I/O, which two of the four signals $\overline{\text{I/OR}}$, $\overline{\text{I/OW}}$, $\overline{\text{MEMR}}$, $\overline{\text{MEMW}}$ of 8257 become active?

Ans. The signals which become active are $\overline{\text{MEMR}}$ and $\overline{\text{I/OW}}$. These two signals become active low.

34. What is the function of 'MARK' output pin of 8257?

Ans. When this (MARK) output pin goes high, it informs the concerned I/O device that the current DMA cycle is the 128th cycle since the previous MARK output.

35. Explain the Auto Load Option of the Mode Set Register of 8257.

Ans. Bit D_7 is the 'Auto Load Option' bit of Mode Set Register 8257. It is enabled when D_7 is set.

This 'Auto Load Option' facility is used, i.e., D_7 bit of mode set register is set (= '1') when some DMA operation is repeatedly desired—like the sending of data to a CRT monitor. This is called repetitive or chained DMA operation and utilises CH–2 and CH–3.

36. What does 'TC stop' option do when it is set?

Ans. It is the D_6 pin of mode set register. When it is set, a channel is automatically disabled when the Terminal Count output goes high. If DMA operation is to be continued or else if another operation is to begin, the channel must be enabled by a fresh Mode Set Operation in the Mode Set Register.

37. Describe the status word register of 8257.

Ans.

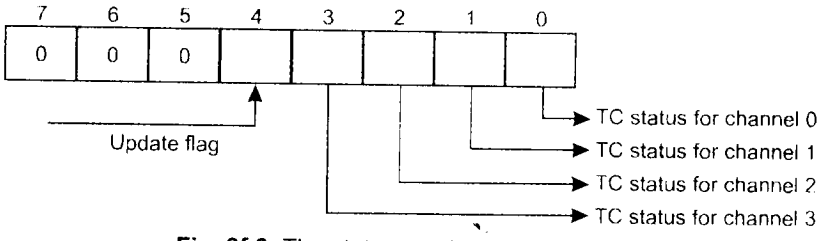


Fig. 9f.6: The status word register

The Status Word Register of 8257 is shown in Fig. 9f.6. It can be read to know the status of the terminal counts of the four channels CH0–CH3. Bit 4 corresponds to the update flag.

Any of the four bits—bit 0 to bit 3 of the status word register is set when the terminal count output corresponding to that channel goes high. When the status word is read, the TC status bits (bits 0 to 3) are cleared.

The update flag is cleared in the following cases:

- When 8257 is reset
- When 'Auto Load' option in the 'Mode Set Register' is reset.
- When it automatically goes low on the completion of update cycle.

The update flag is not affected when a status read operation is undertaken.

38. Indicate the lengths of the different registers within DMAC 8257.

Ans. 8257 has four channels. Each of these four channels has two 16-bit registers—Address Register and Terminal Count Register.

Again 8257 has two 8-bit registers—a Mode Set Register and a Status Register.

39. Describe the functions of the pins $D_0 - D_7$ (pins 21–23, 26–30).

Ans. The functions played by $D_0 - D_7$ bits are different for the following two cases: when 8257 is a slave, and when 8257 is a master.

In the 'slave' mode of 8257, $D_0 - D_7$ pins act as input pins. Eight bits of data at a time for the Address Register or Terminal Count Register (both 16-bits), (for a particular channel) or eight bits of data for the 8-bit Mode Set Register are received through these pins. The CPU can also read eight bits of data at a time from the Address or Terminal Count Registers or from the Status Register.

In the 'master' mode of 8257, the DMAC puts out the eight most significant bits of the DMA Address Register (for a particular channel), at the beginning of each DMA cycle, through $D_0 - D_7$. These are latched by 8212 latch and these latched values are put out on $A_8 - A_{15}$ of the system address bus. Once this operation is over, $D_0 - D_7$ pins are released so that through these pins memory data transfer can be executed for the remainder of the DMA cycle.

40. Discuss the functions of the pins $A_0 - A_3$.

Ans. Two different functions are played by these pins for the two cases when 8257 acts as the master or else as a slave.

In the master mode of 8257, these four pins $A_0 - A_3$ act as output pins. 8257 puts out the four least significant bits of the DMA Address Register on these four pins.

In the slave mode of 8257, while accessing the Mode Set Register or Status Word Register, the pins $A_2\ A_1\ A_0$ must all be '0's, while $A_3 = 1$. Again while mode set operation is done (This is a write only operation), the status of $\overline{\text{I/OW}}$ and $\overline{\text{I/OR}}$ pins would be 0, 1 while for status word (this is a read only operation), the status of the above two pins would be 1, 0 respectively.

41. Explain, in detail how the Address Registers and Terminal Count Registers for each of CH0–CH3 are selected as also the Mode Set Register and Status Word Register.

Ans. The four Address Registers and four Terminal Count Registers of Channels CH0–CH3 can be accessed only if $A_3 = 0$. For Mode Set Register and Word Register accessing, $A_3 = 1$.

For individual channel selection, bits $A_2\ A_1$ are used. With 00, 01, 10, 11 values for $A_2\ A_1$ select channels CH0, CH1, CH2 and CH3 respectively, while the status of bit A_0 distinguishes between channel and Terminal Count Register. If $A_0 = 0$, any of the channels would be selected and if $A_0 = 1$ then Terminal Count Register would be selected.

The channel registers (Address Register or Terminal Count Register) of each channel are 16-bits each. Thus two operations must be performed on a channel register—one for the lower byte and the other for the upper byte to access it fully—be it reading or writing operation. This distinction between the two halves of a channel are done by a special internal First/Last F/F (F/L F/F). This F/F toggles at the completion of each READ/WRITE operation. This F/L F/F assumes a '0' state for LSB and '1' for MSB accessing of a channel register.

A channel register, while being accessed, must be accessed fully—i.e., both LSB and MSB of the channel should be accessed. Therefore, before programming of a channel being initiated, the system interrupts must be disabled, otherwise an interrupt occurring after the first half of channel accessing will prevent the second half of the same channel from being accessed.

The F/L F/F is reset when 8257 gets a Reset input or whenever the Mode Set Register is programmed.

The following table shows in details how both the registers in each channel (CH0–CH3), as also the Mode Set Register and Status Word can be selected, so that they can be programmed accordingly.

Table 9f.1: 8257 Registers selection (*Source:* Intel Corporation)

Register	Byte	Address inputs				F/L	Bidirectional bus							
		A_3	A_2	A_1	A_0		D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
CH 0 DMA Address	LSB	0	0	0	0	0	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
	MSB	0	0	0	0	1	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8
CH 0 Terminal Count	LSB	0	0	0	1	0	C_7	C_6	C_5	C_4	C_3	C_2	C_1	C_0
	MSB	0	0	0	1	1	Rd	Wr	C_{13}	C_{12}	C_{11}	C_{10}	C_9	C_8
CH 1 DMA Address	LSB	0	0	1	0	0	Same as channel 0							
	MSB	0	0	1	0	1								
CH 1 Terminal count	LSB	0	0	1	1	0								
	MSB	0	0	1	1	1								
CH 2 DMA Address	LSB	0	1	0	0	0	Same as channel 0							
	MSB	0	1	0	0	1								
CH 2 Terminal count	LSB	0	1	0	1	0								
	MSB	0	1	0	1	1								
CH 3 DMA Address	LSB	0	1	1	0	0	Same as channel 0							
	MSB	0	1	1	0	1								
CH 3 Terminal count	LSB	0	1	1	1	0								
	MSB	0	1	1	1	1								
Mode set (Program only)	–	1	0	0	0	0	AL	TCS	EW	RP	EN3	EN2	EN1	EN0
Status (Read only)	–	1	0	0	0	0	0	0	0	UP	TC3	TC2	TC1	TC0

A0–A15: DMA starting address, C0–C13: Terminal Count Value (N–1), Rd and Wr: DMA verify (00), Write (01) or Read (10) Cycle selection, AL: AUTO Load, TCS: TC STOP, EW: Extended write, RP: Rotating priority, EN3–EN0: Channel enable mask, UP: Update Flag, TC3–TC0: Terminal count status bits.

42. What happens to Mode Set Register when a 'resetting' of the system is done?

Ans. When the system receives a 'reset', the mode set register is automatically cleared. It disables all the four DMA channels and inhibiting all options.

43. Compare data transfer rate of 8237 DMA and a 2 MHz 8080.

Ans. For the 8237 DMA, the data transfer rate between memory and I/O port is of the order of 1.6 MB/second while it is around 33,000 bytes/second using polling.

9g

Programmable Interval Timer 8253

1. Draw the pin diagram of 8253.

Ans. The pin diagram of 8253 is shown below:

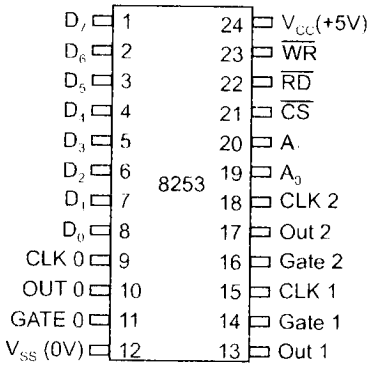


Fig. 9g.1: 8253 Pin diagram (*Source:* Intel Corporation)

2. Draw the functional block diagram of 8253.

Ans. The functional block diagram of 8253 is shown below:

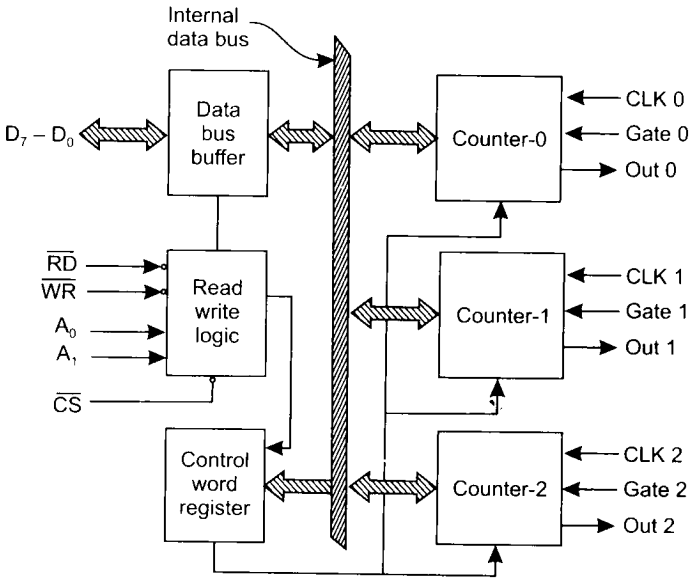


Fig. 9g.2: 8253 Functional block diagram (*Source:* Intel Corporation)

The figure shows that there are six blocks interconnected by an internal data bus. The six blocks include three counters (counter 0, 1 and 2), a data bus buffer, a Read/Write logic and a control word register.

3. How many counters are there in 8253 and how many Modes are there?

Ans. 8253 has three counters—Counter 0, Counter 1 and Counter 2. It operates in 6 different Modes—from Mode 0 to Mode 5.

Each of the three counters are 16-bit, presettable down counters. The counters can be operated in any of the six modes by proper programming.

4. Indicate the application areas of 8253.

Ans. 8253 can be used to generate accurate time delay, events counter, real-time clock, digital one-shot, a square wave generator or a complex wave generator, all under software control.

5. What is the maximum frequency of the waveform obtainable from 8253 Timer?

Ans. The maximum frequency of the waveforms obtainable from Timer 8253 is 2 MHz.

6. Indicate the types of outputs obtained under different modes.

Ans. The different types of outputs obtained from Mode 0 to Mode 5 are as under:

(*i*) An interrupt is obtained on the Terminal Count in Mode 0.
(*ii*) A negative pulse of controllable width is obtained in Mode 1.
(*iii*) A symmetric square wave of controllable frequency is obtained in Mode 2.
(*iv*) A symmetric square wave is obtained in Mode 3.
(*v*) A negative pulse of one clock period is obtained after a software controlled delay in Mode 4.
(*vi*) A delayed negative pulse of one clock period is obtained following a positive going trigger input in Mode 5.

7. What are the functions of the internal data bus buffer?

Ans. This internal data bus is interfaced with the system data bus. The functions of this bus are as follows:

- Loads the count registers.
- Programming the modes of 8253.
- Reads the count values.

8. What are the signals associated with Read/Write Logic block?

Ans. The five control signals associated with this block are: A_0, A_1, $\overline{RD}$, $\overline{WR}$ and $\overline{CS}$. 8253 can be enabled/disabled by $\overline{CS}$ signal.

9. What are the signals associated with each counter.

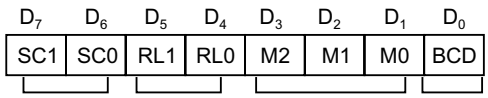
Ans. There are three counters in 8253. Each of these three counters has two input signals—Clock (CLK) and GATE and one output signal—OUT. The purpose of the GATE signal is to enable/disable a particular counter.

10. How the Control Word Register is selected?

Ans. The control word register is selected only if $A_1\ A_0 = 11$. Also the status of $\overline{CS}$, $\overline{RD}$, $\overline{WR}$ signals should be 0, 1 and 0 respectively.

11. Write down the Control Word format as also the mode definitions.

Ans. The details of the control word format, as also the mode definitions are detailed below:



Definition of Control:

SC-Select Counter:

SC1	SC0	
0	0	Select counter 0
0	1	Select counter 1
1	0	Select counter 2
1	1	Illegal

M-Mode

0	0	0	Mode 0
0	0	1	Mode 1
X	1	0	Mode 2
X	1	1	Mode 3
1	0	0	Mode 4
1	0	1	Mode 5

RL-Read/Load:

RL1	RL0	
0	0	Counter latching operation (see read/write procedure section)
1	0	Read/Load most significant byte only
0	1	Read/Load Least significant byte only
1	1	Read/Load Least significant byte first, then most significant byte

BCD:

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

Fig. 9g.3: Control word format and mode definitions
(*Source:* Intel Corporation)

The information stored in the control word gives the following details:

- Bits D_7 and D_6 (SC_1 and SC_0) select a counter.
- Bits D_5 and D_4 (RL1 and RL0) determine whether it is a Read or Load Count operation and also whether it is Least Significant Byte or Most Significant Byte of the count that is involved.
- Bits D_3, D_2, D_1 (M2, M1, M0) determine the operating mode (i.e., Mode 0 to Mode 5)
- Bit D_0 (BCD) determines the counting sequence in binary or BCD format.

12. How to ensure that a counter is loaded?

Ans. For the above to be ensured, it is necessary that:

- The count value be written (a single byte or double byte—which depends on the mode selection by RL1 and RL0 bits).
- The above is then to be followed by a rising and a falling edge of the clock. Data, read before the falling edge of the clock, is an invalid one.

13. How writing operation is done in a counter (i.e., counter 0 or 1 or 2)?

Ans. First the control word register is selected.
Secondly, the control word is to be appropriately chosen/written by

(*a*) selecting the counter in which writing is to be done ($D_7 - D_6$ bits).
(*b*) filling in $D_5 - D_4$ bits correctly (which takes care of 1 or 2 byte count).
(*c*) filling in $D_3 - D_1$ bits, which corresponds to Mode selection.
(*d*) filling in D_0 bit—its content reflects the count down to be done in binary/BCD.

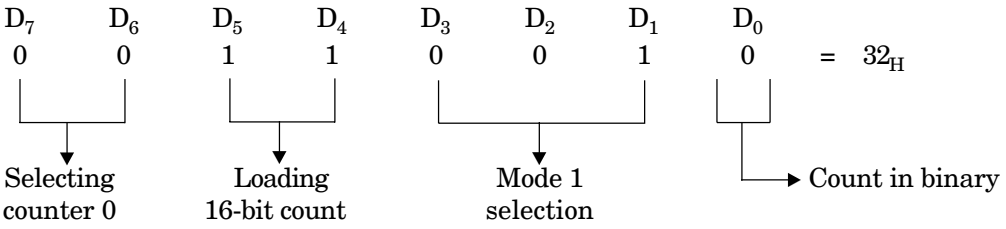
14. What will be the Control word register address if $\overline{CS}$ of 8253 is connected to A_7 via an inverter.

Ans. Since $\overline{CS}$ pin of 8253 is connected to A_7 via an inverter, then A_7 will have to be 1, i.e., $A_7 = 1$. Again A_1 and A_0 will both have to be 1 to ensure writing in the control register. Assuming A_6 to A_2 to be 0, then the port address of the control word register will be

A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0		
1	0	0	0	0	0	1	1	=	83H

15. Write the value of the control word if in a specific case counter 0 is to be selected in Mode 1. The counter should have a 16-bit count and that it should count down in binary. Lastly load the control word register with the control word value so formed. Assume address of control word register = 83_H.

Ans. Control word for load operation is as follows:



The program for loading the control word value 32_H in the control register address 83_H is as follows:

MVI A, 32_H = Control word is loaded in accumulator

OUT 83_H = The control word (= 32_H) is written into the control word register, having address 83_H.

16. Show in a tabular form the conditions of the different Modes corresponding to the different status of the Gate signal.

Ans. This is shown below in a tabular form:

Table 9g.1: Gate pin operations (*Source*: Intel Corporation)

Signal status / *Modes*	**Low or Going low**	**Rising**	**High**
0	Disables counting	——	Enables counting
1	——	1. Initiates counting 2. Resets output After next clock	——
2	1. Disables counting 2. Sets output Immediately high	1. Reloads counter 2. Initiates counting	Enables counting
3	1. Disables counting 2. Sets output Immediately high	Initiates counting	Enables counting
4	Disables counting	——	Enables counting
5	——	Initiates counting	——

17. Discuss the two methods of reading the value of the count in a counter while count is in progress.

Ans. There are two methods of doing the same

(a) Reading by halting/stopping the count

(*b*) Reading while counting is 'ON' (i.e., counting is in progress)

(*a*) Reading by halting: The procedure is as follows:

(*i*) The counter must be identified with appropriate A_1 A_0 status.

(*ii*) The counter is then halted by either disabling the Gate pin or inhibiting the CLK input.

(*iii*) Then I/O read operation is done—the first I/O Read gets the LSB and the second I/O Read gets the MSB.

(*b*) Reading while counting is in progress: For this to be effective, the mode register is loaded with a code that would load the present count (in a counter) to be latched on to a storage register.

The Mode Register Format for Latching Count is as follows:

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
SC1	SC0	0	0	X	X	X	X

$D_5 - D_4$ are loaded with 0 each for latching the present counter content.

Then the Read command is invoked to the selected counter which gives the content of the latched register—the counter must be programmed for two bytes and must be read prior to any Write Command to the concerned counter.

9h

8254: Programmable Interval Timer

1. Draw the pin diagram and functional block diagram of 8254.

Ans. The pin diagram and functional block diagram of 8254 are shown below:

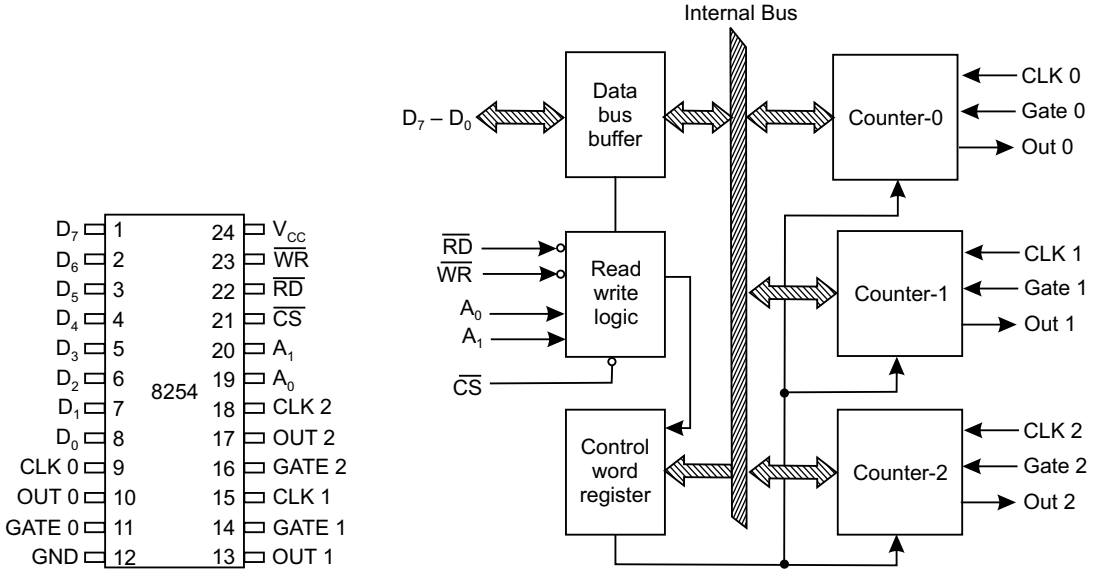


Fig. 9h.1: Block diagram and pin descriptions for the 8254 programmable interval timer. Three separate timer/counters are provided. (*Courtesy:* Intel Corporation)

2. How many counters are there in 8254?

Ans. There are three 16-bit counter registers, each of which can be programmed as a timer or an event counter. The counters are named as COUNTER 0, COUNTER 1 and COUNTER 2.

3. How programming is done in 8254?

Ans. The programming is done by writing a control word in the Control Word Register.

4. In how many modes can 8254 operate?

Ans. 8254 can operate in six possible operating Modes—Mode 0 to Mode 5.

5. What kind of outputs are available from the different operating modes.

Ans. From the six different operating modes, the outputs which are available are: event counter, one-shot, square-wave generator, divide-by-N counter, hardware triggered strobe and software triggered strobe.

6. What happens when the Terminal Count (TC) of a counter is reached?

Ans. The counters operate in count down mode. When a counter counts down to zero, it is called 'Terminal Count'. When TC for a particular counter occurs, the following may occur:

- an interrupt can be requested
- a one-shot pulse can be terminated
- a strobe pulse can be generated
- the logic level of a square wave can be toggled.

7. What are the differences between timers 8253 and 8254?

Ans. Timer 8254 is actually a "Superset" of timer 8253 and is pin compatible with the latter. There are two differences between the two, which are

- 8254 has read back mode facility—which means that the status of a particular mode can be read after programming. This facility is not available with 8253.
- The maximum clock frequency of 8254 is 10 MHz, whereas that of 8253 is 2 MHz.

8. In how many forms can the control word register be used?

Ans. The control word can be used in three formats—Standard mode, counter latch mode and Read back mode.

9. How one of the Six Modes (Mode 0 to Mode 5) is selected?

Ans. The particular mode is selected by the standard form of the control word in the control word register.

10. Draw the control word register format and discuss.

Ans. The control word of the control word register is shown Fig. 9h.2.

The control word byte is divided into four parts $D_7 - D_6$, $D_5 - D_4$, $D_3 - D_2 - D_1$ and D_0. The different combinations of these eight bits give rise to 'standard' mode, 'Counter latch' mode or 'Read back' mode and are self evident from the figure.

11. Discuss the six different modes in which 8254 can operate.

Ans. The six different modes are Mode 0 to Mode 5. These are discussed below:

Mode 0: This is an 'event counter', when GATE = 1, the counter will start decrementing from its stored value on the falling edge of the second pulse of CLK input. OUT will go high when 'terminal count' is reached. This OUT signal can be utilised as an interrupt input to the microprocessor.

Mode 1: It is 'hardware triggered one shot', when the rising edge of the GATE pulse is received, OUT goes low and remains low till TC (terminal count) is reached. Then OUT goes high. This low one shot duration is equal to the stored value in the counter multiplied by the CLK period.

Mode 2: It is a 'Divide-by-N' counter, when GATE = 1, OUT goes low for one period of the CLK input after the stored count is decremented to 1. The initial (or stored) count will then automatically get reloaded and the cycle repeats.

Mode 3: It is a 'square wave generator'. It is identical to Mode 2, but the duty cycle (= on time/period) here is 50%. In case, the initial count is an odd number, OUT then will be high for one more clock cycle then it is low.

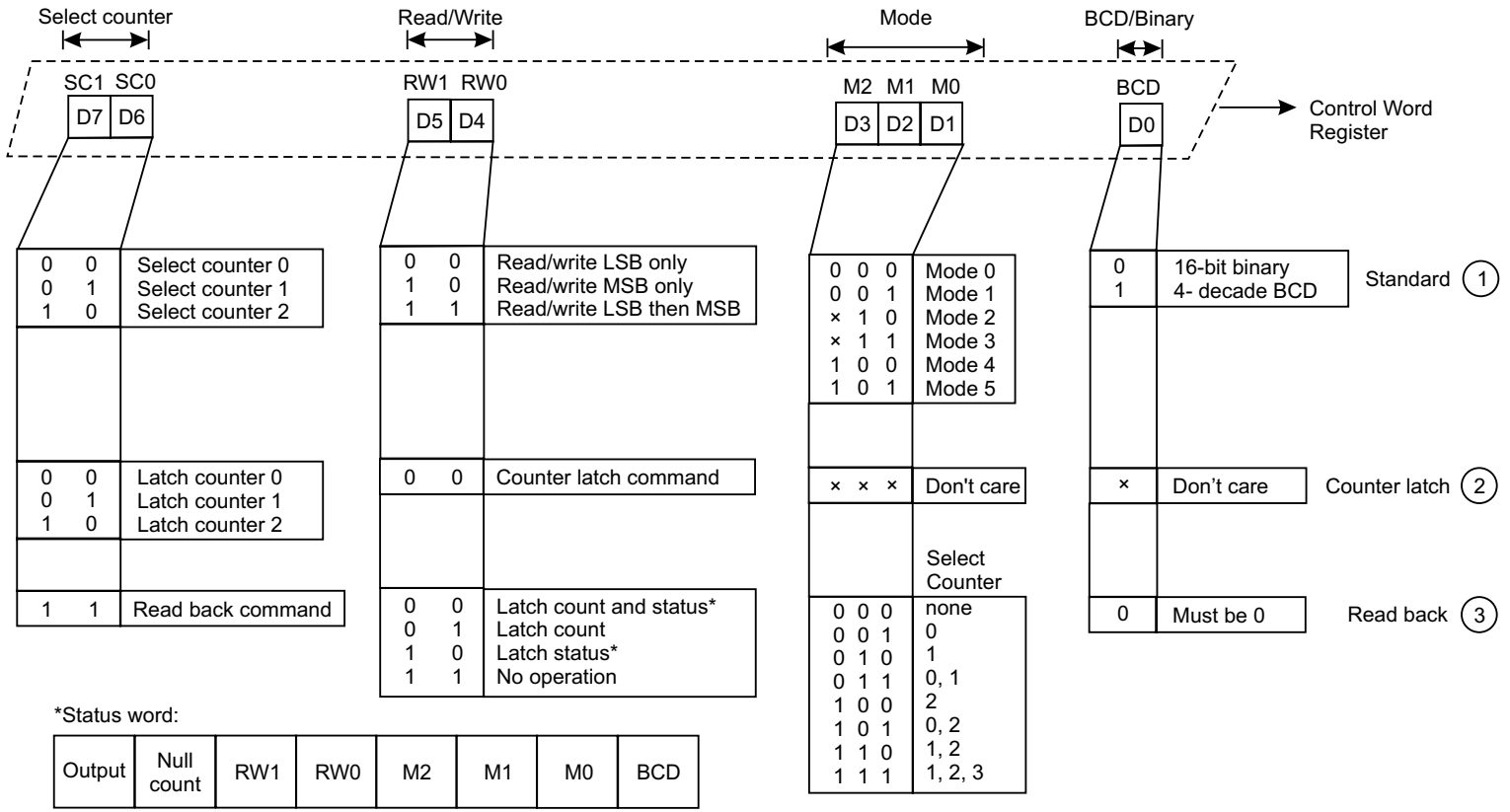


Fig. 9h.2: 8254 Control word. The standard form is used to specify the operating mode. The Counter Latch and Read Back Commands are used when the present count or status is to be read

N.B.: (1), (2) and (3) are the three types of commands possible by the Control Word Register.

Mode 4: It is a 'software triggered mode'. If GATE = 1, OUT goes low for one period of the CLK input N clock cycles after, where N is the initial (or stored) number in the counter. For second strobing to be done, N must be reloaded in the counter.

Mode 5: It is a 'Hardware triggered mode'. On the appearance of rising edge of the GATE input, the stored or initial count starts decrementing to 0. When TC occurs, OUT goes low for one period of the CLK input.

12. How the three counters and the control word are selected?

Ans. The combination of A_1 A_0 select the above and shown below:

A1	A0	
0	0	Write into Counter 0
0	1	Write into Counter 1
1	0	Write into Counter 2
1	1	Write Counter word

13. Write down the port addresses of the three counters and the control word register. The $\overline{CS}$ signal is derived on the basis of $A_7 - A_4 = 1111$ and $A_3 - A_2 = 00$.

Ans. The port addresses of the counters and the control word register are determined as follows:

A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0				Port address of
1	1	1	1	0	0	0	0	=	F0	⇒	Counter 0
1	1	1	1	0	0	0	1	=	F1	⇒	Counter 1
1	1	1	1	0	0	1	0	=	F2	⇒	Counter 2
1	1	1	1	0	0	1	1	=	F3	⇒	Control Word Register.

14. Write down the Control Word so that Counter 1 operates in Mode 0 in binary sequence.

Ans. By having a look at the control word register, the control word will be as follows:

0 1 1 1 0 0 0 0 = 0111 0000 = 70

While writing the above control word, it is assumed that Counter 1 is to be loaded with a 2-byte count.

9i

USART 8251 (Universal Synchronous/ Asynchronous Receiver Transmitter)

1. Draw the pin diagram of USART 8251.

Ans. The pin diagram of 8251 is as shown below:

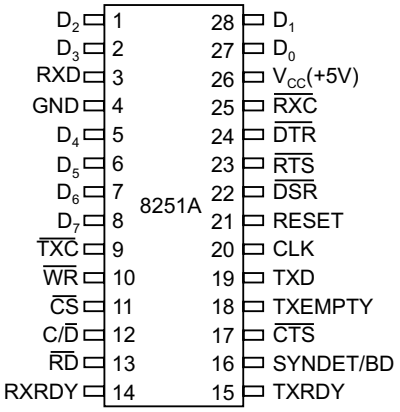


Fig. 9i.1: 8251 pin diagram (*Source:* Intel Corporation)

2. Draw the functional block diagram of 8251.

Ans. The functional block diagram of 8251 is shown below:

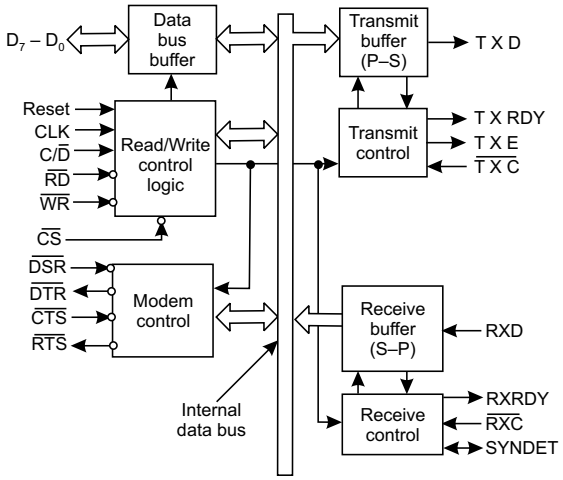


Fig. 9i.2: Functional block diagram of 8251

3. How many different sections does 8251 have?

Ans. 8251 has five sections: Read/Write control logic, data bus buffer, modem control, transmitter (including its control) and receiver (including its control).

4. How the 8251 is programmed?

Ans. 8251 is programmed by a 16-bit Control Word Register. This 16-bit register is divided into two bytes—the first byte corresponds to Mode Instruction Format while the second byte corresponds to Command Instruction Format. The content of control word register determines synchronous or asynchronous operation, Baud rate, number of bits per character, number of stop bits, nature of parity, etc.

5. What is the function of the Status Word Register of 8251?

Ans. The function of the status word register is to check or examine the preparedness of 8251 with regard to transmission or reception of data.

6. Describe the Read/Write Control logic and registers.

Ans. It contains three buffer registers:

- data buffer register
- control register
- status register.

The six input signals are: $\overline{CS}$, $C/\overline{D}$, $\overline{WR}$, $\overline{RD}$, RESET and CLK.

The particular 8251 is selected on $\overline{CS}$ signal going low. This pin is usually connected to a decoded address bus. $C/\overline{D}$ stands for control/data pin. When this pin is high, either the control register or status register is selected and when low, data bus buffer is selected. The control register and the status register are distinguished by $\overline{WR}$ and $\overline{RD}$ signals, respectively.

The figure below shows how the three registers: data buffer register, control register and status register are accessed by making respectively.

$C/\overline{D} = 0, \quad \overline{RD} = 0 \quad$ or else $\overline{WR} = 0$

$C/\overline{D} = 1, \quad \overline{WR} = 0$

$C/\overline{D} = 1, \quad \overline{RD} = 0$

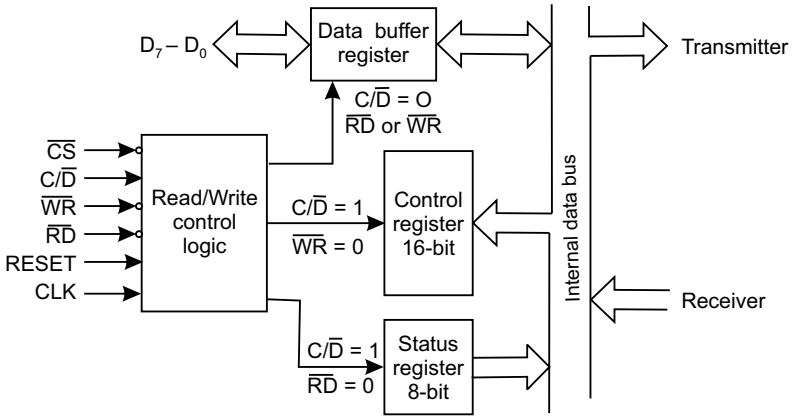


Fig. 9i.3: Accessing the data buffer register, control register and status register

The following table shows the status of the control signals $\overline{CS}$, C/$\overline{D}$, $\overline{RD}$, $\overline{WR}$ for accessing the different registers.

Table 9i.1: Summary of Control Signals for the 8251A

$\overline{CS}$	C/$\overline{D}$	$\overline{RD}$	$\overline{WR}$	*Function*
0	1	1	0	MPU writes instructions in the control register
0	1	0	1	MPU reads status from the status register
0	0	1	0	MPU outputs data to the Data Buffer
0	0	0	1	MPU accepts data from the Data Buffer
1	X	X	X	USART is not selected

7. Explain the operation of the transmitter section of 8251.

Ans.

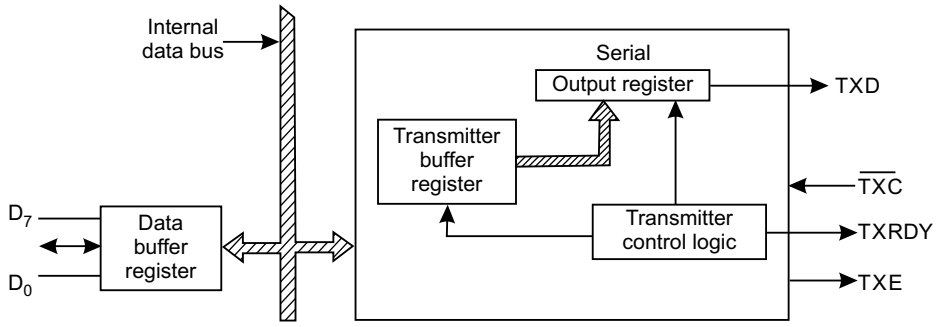


Fig. 9i.4: Transmitter section

The transmitter section consists of three blocks—transmitter buffer register, output register and the transmitter control logic block. The CPU deposits (when TXRDY = 1, meaning that the transmitter buffer register is empty) data into the transmitter buffer register, which is subsequently put into the output register (when TXE = 1, meaning that the output buffer is empty). In the output register, the eight bit data is converted into serial form and comes out via TXD pin. The serial data bits are preceded by START bit and succeeded by STOP bit, which are known as framing bits. But this happens only if transmitter is enabled and the $\overline{CTS}$ is low. $\overline{TXC}$ signal is the transmitter clock signal which controls the bit rate on the TXD line (output line). This clock frequency can be 1, 16 or 64 times the baud.

8. Explain the operation of the receiver section of 8251.

Ans.

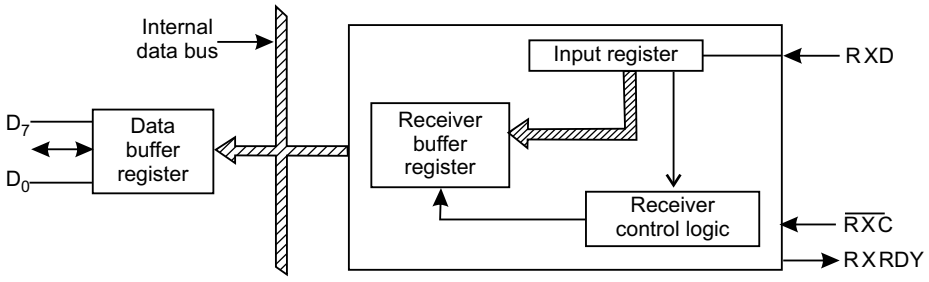


Fig. 9i.5: Receiver section

The receiver section consists of three blocks — receiver buffer register, input register and the receiver control logic block. Serial data from outside world is delivered to the input register via RXD line, which is subsequently put into parallel form and placed in the receiver buffer register. When this register is full, the RXRDY (receiver ready) line becomes high. This line is then used either to interrupt the MPU or to indicate its own status. MPU then accepts the data from the register.

$\overline{\text{RXC}}$ line stands for receiver clock. This clock signal controls the rate at which bits are received by the input register. The clock can be set to 1, 16 or 64 times the baud in the asynchronous mode.

9. How does the CPU know that the transmitter buffer is empty?

Ans. The CPU knows about the same by

- examining the TXRDY line of the transmitter.
- examining D_0 bit of the status word.

10. When the TxD line goes into the MARKING (HIGH) state?

Ans. It becomes high in the following cases:

- a RESET is received by 8251.
- transmitter is empty.
- transmitter is not enabled.
- $\overline{\text{CTS}}$ is off.

11. How the transmitter is enabled?

Ans. The transmitter is enabled by setting bit D_0 of the command instruction word.

12. What is the status of the start bit on the RXD line for 8251 (in asynchronous mode only) and how does it differentiate between a valid start pulse and transient pulse?

Ans. For any data to be received by the receiver, it first checks for a valid start bit which is zero. 8251 has an inbuilt false start bit detection circuit which can differentiate between an actual start bit pulse and a transient pulse.

13. What happens when a parity error or a framing error occurs in the received data bits (in asynchronous mode only)?

Ans. The Parity error and Framing error status bits in the status word are set if there is a parity error or if the stop bit is absent at the end of the received bits respectively.

14. When the RXRdy line goes high in asynchronous and synchronous mode of operation?

Ans. In the asynchronous mode, RXRdy line goes high

- if the receiver is in the enabled condition (this is made so by setting D_2 bit of the Command Instruction Word).
- and after the receiver has detected a valid start bit, assembled the character bits and transferred the character to the receiver buffer register.

Whereas in the synchronous mode RXRdy line goes high

- if the receiver is enabled.
- a character is assembled and transferred to the receiver buffer register.

15. What is overrun error?

Ans. D_4 bit of the status word stands for 'over run error'.

If the CPU cannot read the data from the receiver buffer register (this happens if the CPU fails to respond to RXRdy line), then on receipt of the next character, the previous data will be written over and the earlier character will be lost. When such is the case D_4 bit of the status word is set.

16. Discuss the SYNDET/BD pin.

Ans. This is pin 16 of 8251 and stands for sync. detect (SYNDET)/Break Detect (BRKDET).

This pin is used for detection of SYNC characters in synchronous mode and Break characters in asynchronous mode.

Synchronous mode: In the synchronous case this pin (SYNDET) can be used as either input or output pin with the help of control word. When the system is RESET, the status of this pin is low in the output mode. When 8251 is programmed to receive two synchronous characters, this output pin goes high at the mid point of the last bit of the second synchronous character. The status of this pin can also be known by reading the status word, but gets resetted on STATUS READ.

In the input mode—called the external synchronous detect mode—a rising edge on this pin causes 8251 to start collecting data characters on the rising edge of the next $\overline{\text{RXC}}$. This input signal can be removed once synchronisation is achieved. When external synchronisation is done, the internal synchronisation is disabled.

BRKDET: In this mode, this pin acts as an output pin to detect break characters. If RXD remains low for two consecutive stop bit sequences, this pin (BRKDET) goes high. Here also provision is there to read the status of this pin by STATUS READ operation.

This pin is reset

- on a Master Chip Reset
- when RXD becomes on 1.

17. What purpose does 8251 serve—DTE or DCE in a communication interface environment?

Ans. 8251 acts as a DTE (Data Terminal Equipment) in such a case.

18. Why modems (modulators-demodulators) are used in case of digital transmission of data?

Ans. The term 'modem' stands for modulator—demodulator. In a communication environment, two modems are used—one at the transmitting end side and one at the receiving end side. Modems are generally called DCE (Data Communication Equipment).

High frequency digital signals require a very wide transmission channel bandwidth which makes the system very costly. However, existing telephone line facilities (which carry analog signals in the range of 40 Hz to 4 KHz) can be used to transmit such high frequency digital signals. A modem converts a digital signal into audio tone frequencies (at the transmitting end side) and reconverts this audio frequencies into h.f. digital signals (at the receiving end side)—and it utilises Frequency Shift Keying (FSK) for this purpose. Thus a modem converts a logical '1' to 1200 Hz and '0' to 2200 Hz audio frequency. These signals are then transmitted over a telephone line over a 'carrier'. The inverse operation is done at the receiving end side.

19. Draw the block diagram of a DTE-DCE interface in a communication environment.

Ans. The block diagram is shown below:

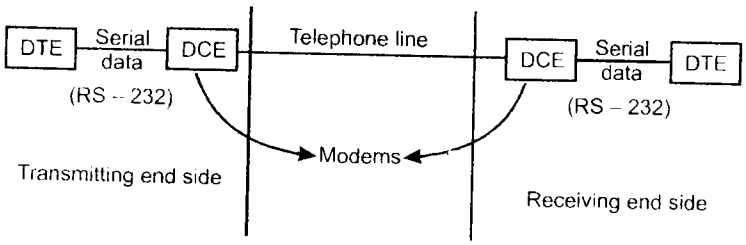


Fig. 9i.6: DTE-DCE interface

Digital data is delivered at the DTE (may be 8251) in parallel form, which is then converted into serial form and sent to DCE via RS-232 cable. The DCE (a modem) output is an audio signal carried through a telephone line.

At the receiving end side, the opposite process is carried out to retrieve the original data.

20. Draw the 8251 data loading sequence and explain the same.

Ans. The data loading sequence of 8251 is shown below:

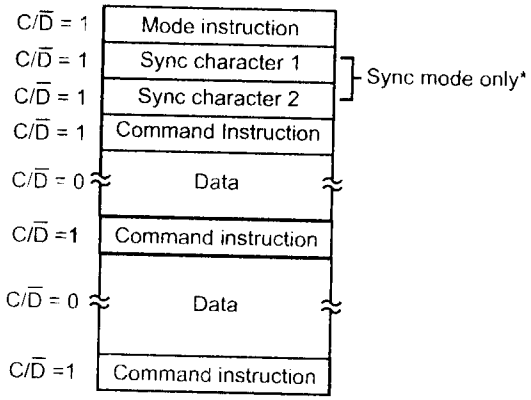


Fig. 9i.7: 8251 Data-loading sequence (*Source:* Intel Corporation)

The mode control is specified first, which indicates the general operating conditions. If the mode word indicates that it is a synchronous operation, then the synchronous character(s) is/are loaded. This is followed by loading the command instruction format. In all these, $C/\overline{D} = 1$. After this, $C/\overline{D}$ is made 0 when data is either transmitted/received. It is followed by command instruction and data in that order which is repeated all over again.

A command word with $D_6 = 1$ returns 8251 to mode instruction format.

21. Discuss the mode instruction format for asynchronous transmission/reception case.

Ans. The mode instruction format for asynchronous operation (transmission/reception) is shown below:

Bits D_0 D_1 cannot both be low for asynchronous communication. These two bits determine the baud rate factor. Bits D_2 D_3 determine the character length (which may be 5 to 8 bits in length)—depending on the content of these two bits. Bit D_4 stands for 'Parity Enable' (PEN) and is enabled if $D_4 = 1$ and otherwise if $D_4 = 0$. D_5 bit stands for 'Even Parity' (EP). Parity is even if $D_5 = 1$ and odd if $D_5 = 0$. Bits D_6 and D_7 determine the number of stop bits. There can be 1, 1½, 2 number of stop bits.

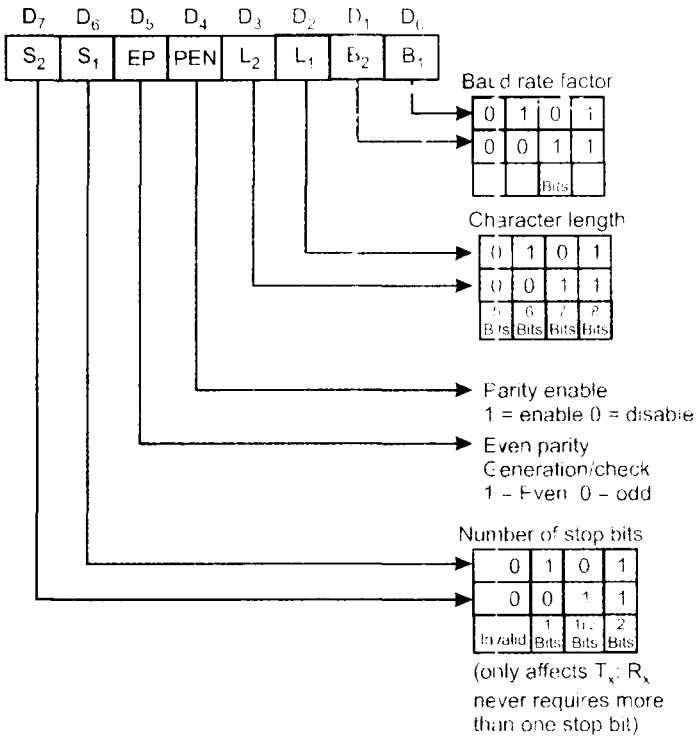


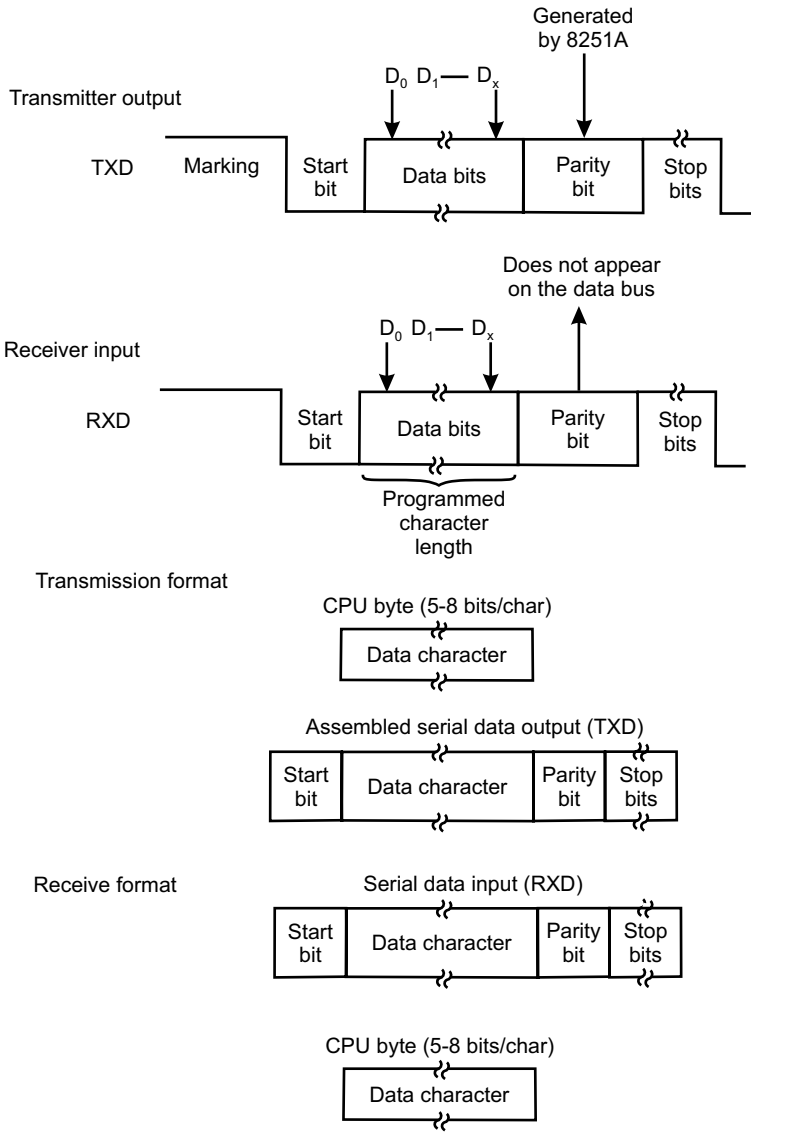
Fig. 9i.8: Mode instruction format: Asynchronous operation
(*Source:* Intel Corporation)

22. Draw the general transmission/receive format for asynchronous communication.

Ans. The general transmission/receive format for asynchronous communication is shown below:

The transmission format consists of start bit, data character, parity bit, stop bit(s)—in that order.

8251 starts sending data on the TXD (transmit data pin) pin with a start bit which is a 1 to 0 transmission. Then the data bits are transmitted, followed by stop bit(s). The data bits start with LSB of the serial output register. All these bits (start bit, data bits, stop bit(s) are shifted out on the falling edge of $\overline{TXC}$ (transmitter clock). In case when no data is transmitted, TXD output remains high. But if a 'break' is programmed, TXD line will go low.



*Note: If character length is defined as 5, 6, or 7 bits the unused bits are set to zero

Fig. 9i.9: Asynchronous transmission and reception (*Source:* Intel Corporation)

The receive format is identical to transmit format. Data reception starts with RXD (receive data pin) line going low—it indicates the arrival of start bit. This 1 to 0 transition on the RXD line triggers the 'False Start Bit Detection Circuit'. This circuit then samples the RXD line half-a-bit time later to ensure the presence of a genuine start bit. If this sampling results in a low on RXD line, it indicates a valid start bit. The bit counter is started on the second sampling—hence each subsequent data bit is

sampled at the middle of each bit period. This is called 'mid bit sampling'. The bit counter thus samples the data bits, parity bit and lastly the stop bit. The receiver needs only one stop bit—but the transmitter is affected by the number of stop bits. For any error during receiving of data with regard to Parity, Framing or Overrun—the corresponding flags in the status word are set.

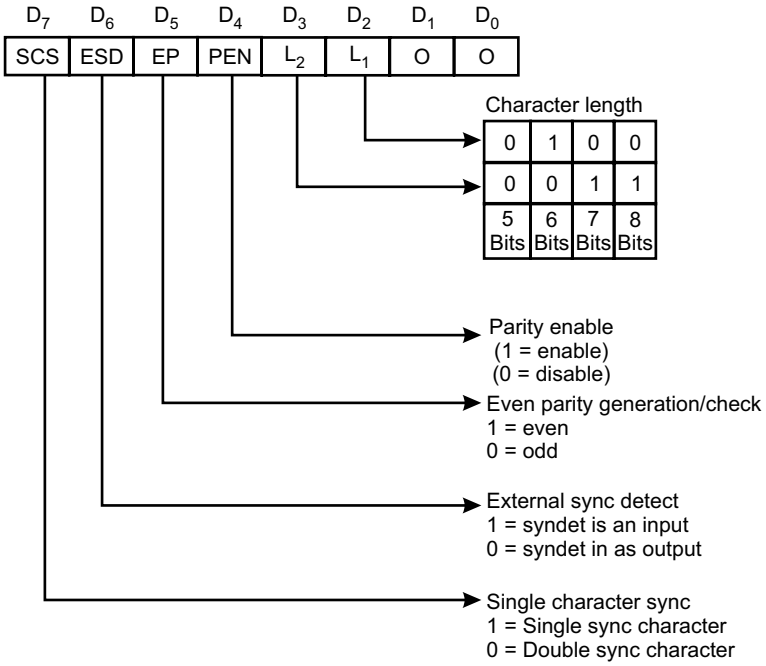
23. Why the 'false start bit detection circuit' is there in asynchronous reception case?

Ans. This is done to avoid any possibility of a false start bit detection due to a transient noise pulse.

24. Discuss the mode instruction format for synchronous transmission/reception case.

Ans. The mode instruction format for synchronous operation (transmission/reception) is shown below:

Bits D_0 D_1 both will have to be low for synchronous transmission/reception of data. Bits D_2 D_3 indicates the character length. Bits D_4 and D_5 stand for PEN and EP respectively—exactly same as in the case of asynchronous case. Bit D_6 stands for ESD (External Synchronous Detect). $D_6 = 1$ stands for input and $D_6 = 0$ stands for output. Bit D_7 stands for SCS (Single Character Sync.) with $D_7 = 1$ indicating a single synchronous character and $D_7 = 0$ indicating double synchronous characters.



Note: In External sync mode, programming double character sync will affect only the TX.

Fig. 9i.10: Mode instruction format: Synchronous operation (*Source:* Intel Corporation)

25. Draw the general transmission/receive format for synchronous communication.

Ans. The general transmission/receive format for synchronous communication is shown below.

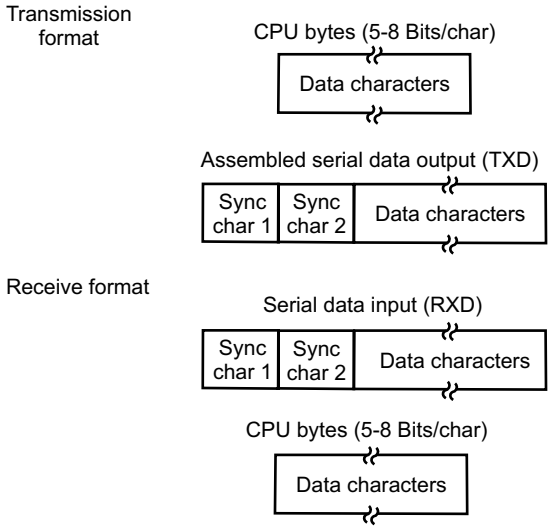


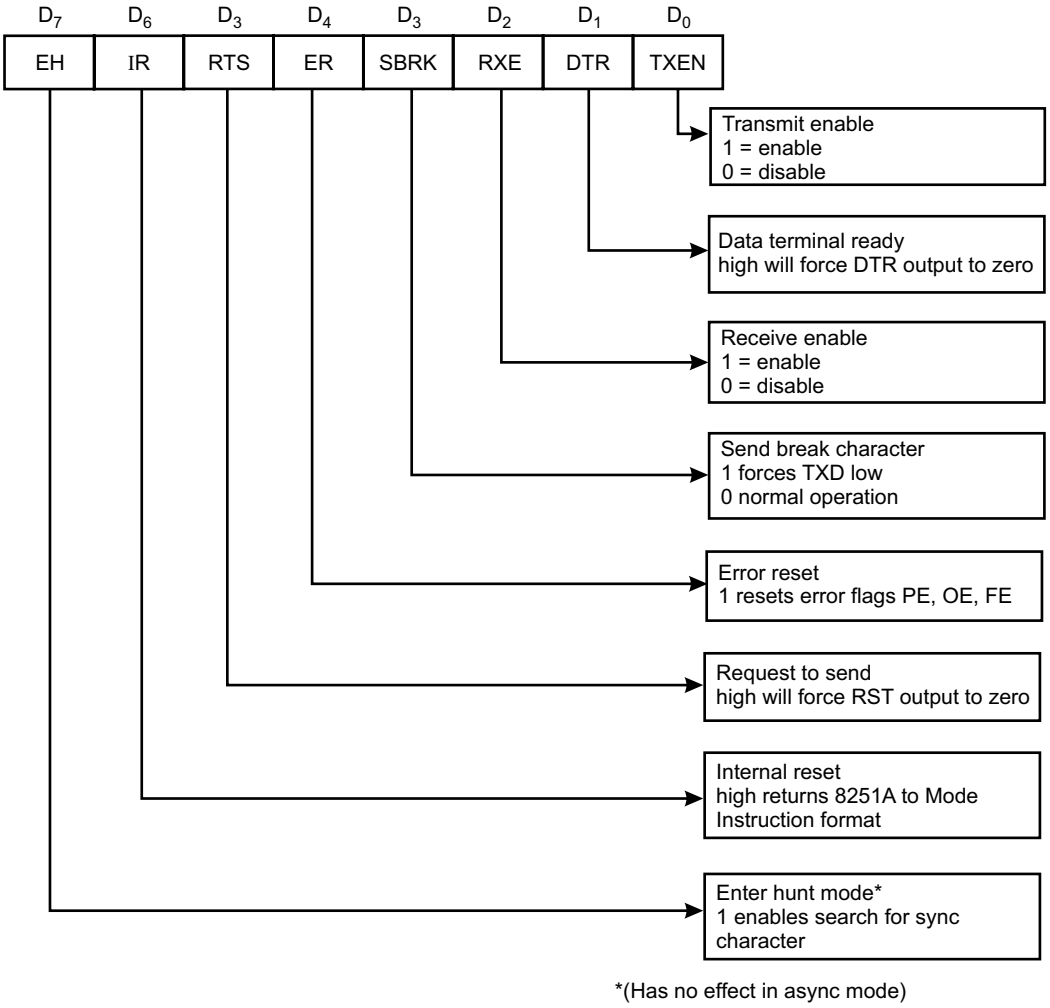
Fig. 9i.11: Synchronous mode transmission and reception
(*Source:* Intel Corporation)

In the transmission format, either one or two synchronous characters are sent, followed by data characters. The number of synchronous characters (i.e., 1 or 2) is previously decided by the bit D_6 in the mode instruction format for synchronous operation. For such communication to take place, $C/\overline{D}$ will have to be 1. The characters are shifted out of the serial output register on the falling edge of the $\overline{TXC}$ (transmitter clock), and at the same rate as the $\overline{TXC}$. Once transmission commences, it is the duty of the CPU to replenish the transmitter buffer register in response to TXRdy. If the CPU fails to provide a character before the transmitter buffer becomes empty, 8251 automatically sends SYNC character(s). In such a case, TXE (Transmitter Empty) pin becomes high to indicate that the transmitter buffer is empty.

In the receive format, $C/\overline{D}$ is maintained at high level. In the internal SYNC mode, the receiver samples the data available as the RXD pin on the rising edge of $\overline{RXC}$. The command word should be previously programmed with the 'ENTER HUNT' command (bit D_7 of the Command Instruction Format) in the Enabled Condition ($D_7 = 1$). The receiver buffer register content is compared at every bit boundary with the SYNC character (previously loaded) till a match occurs. The process is extended to two SYNC characters if the 8251 is initially programmed for two SYNC characters (bit D_7 of the Mode Instruction Format). After 'HUNTING' is over, the system goes for character boundary synchronisation so that it can assemble the serial data to be subsequently changed to parallel format. The SYNDET pin is set high, which can be ascertained with a status read. This is resetted once status read is over. The SYNDET pin gets set in the middle of the parity bit if the parity is enabled; otherwise in the middle of the last data bit. In the external SYNC mode, 8251 comes out of HUNT mode by a high level on the SYNDET pin, which acts as an input in such a case.

26. Show the Command Instruction Format and explain the same.

Ans. The Command Instruction Format is shown below:



Note: Error reset must be performed whenever RXEnable and enter hunt are programmed

Fig. 9i.12: Command instruction format (*Source:* Intel Corporation)

The command instruction format controls the functioning of 8251. A command word with $D_6 = 1$ returns 8251 in mode instruction format. If D_0 (TXEN) is mode high, data transmission is possible whereas making D_2 (RXE) high, enables the system for reception. If D_1 (DTR) is made high, the DTR output will be forced in the zero state. A high on D_4 (ER) forces resetting of error flags PE, OE and FE (Parity, overrun and Framing errors respectively) in the status word. A high on D_3 (SBRK) forces TXD low while a zero corresponds to normal operation.

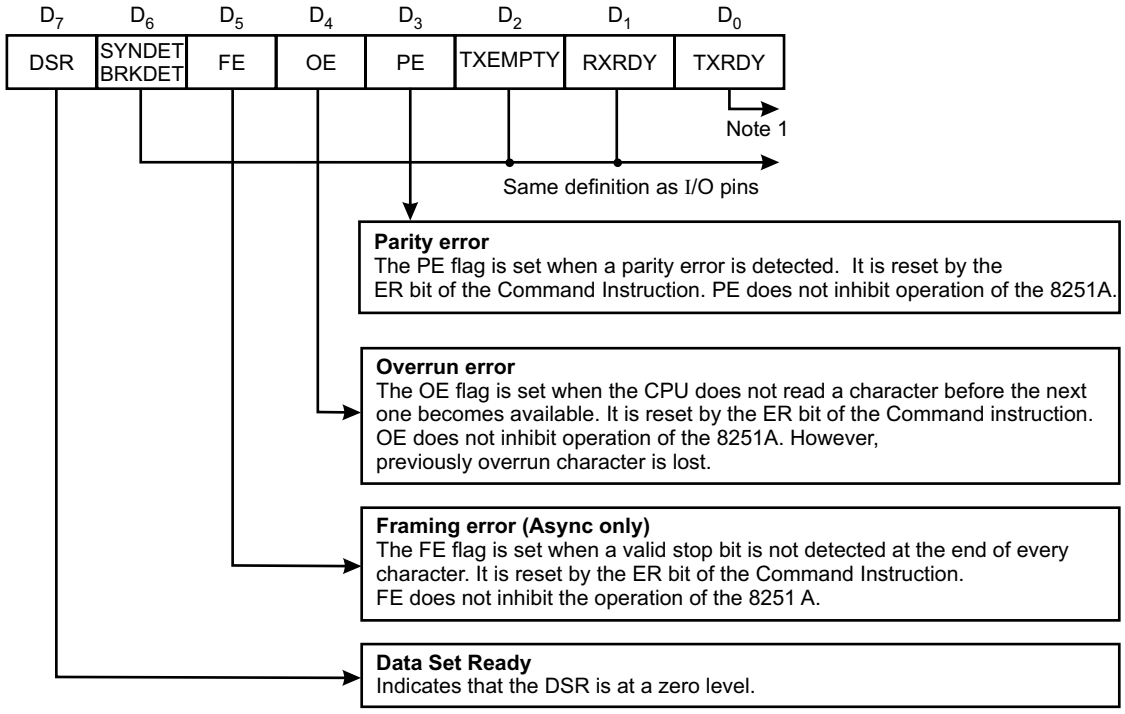
27. What happens when (*a*) power is switched on (*b*) the system is resetted?

Ans. On powering on the system, 8251 either enters into SYNC or command instruction format.

On resetting the system, 8251 returns to the mode instruction format from the command instruction format.

28. Draw the status word format and explain the same.

Ans. The status word format for 8251 is shown below:



Note 1: TXRDY status bit has different meaning from the TXRDY output pin. The former is not conditioned by $\overline{CTS}$ and TXEN; the latter is conditioned by both $\overline{CTS}$ and TXEN
i.e., TXRDY status bit = DB buffer empty
TXRDY pin out = DB buffer Empty. ($\overline{CTS}$ = 0). (TXEN = 1)

Fig. 9i.13: Status Word Format (*Source:* Intel Corporation)

The status word can be read with C/$\overline{D}$ = 1. The CPU, for its proper operation, needs various informations. These are provided by the status word. It should be borne in mind that the status word is continuously updated by 8251, but not while the CPU reads it.

29. What are the modem control pins associated with 8251? Describe the functioning of these pins.

Ans. The modem control section of 8251 are handled by these four pins: $\overline{DSR}$, $\overline{DTR}$, $\overline{CTS}$ and $\overline{RTS}$. Out of these, the first and third are input pins (input to 8251) and the rest two are output pins. All these pins are active low. The signals on these pins are also used for purposes other than modem control. The description of these pins are given below:

$\overline{DSR}$ (Data Set Ready): This is a 1-bit inverting input port. It is used by the modem to signal the 8251 (here DTE) that it (modem) is ready to accept data for transmission. The DSR bit is checked by reading (polling) the D_7 bit of the status word. If it is low, then the modem can send data to 8251.

$\overline{\text{DTR}}$ **(Data Terminal Ready):** This is a 1-bit inverting output port. It is used by 8251 to signal the modem about its readiness to accept/transmit data. D_1 bit of command instruction word can either be set/reset, with a high D_1 bit forcing DTR output to zero.

$\overline{\text{RTS}}$ **(Request to Send):** This is a 1-bit inverting output port. It is used by 8251 to signal the modem that it has data to send. Bit D_5 of the Command Instruction Format controls the status of this pin.

$\overline{\text{CTS}}$ **(Clear to Send):** This is a 1-bit inverting input port. It is used by modem to signal 8251 that it has the right of way over the communication channel and can send out serial data. Bit D_0 of the command instruction word should be enabled for the above to be realised.

If D_0 is made low in command instruction word while data transmission is taking place or if $\overline{\text{CTS}}$ is switched off, the transmitter will complete sending the data stored in its buffer prior to getting disabled.

30. What is the baud rate of 8251?

Ans. The asynchronous baud rate of 8251 is 9600, while for the improved version of 8251—i.e., 8251A, this is 19, 200.

31. Discuss how a noise pulse may be recognised as a valid start pulse. How this possibility is eliminated?

Ans. In the asynchronous case, a USART may be programmed for receive clock rates of 8,16,32,64 times the receive data rate (these correspond to 8X, 16X, 32X and 64X). Thus the receive clock rates may be 8X RCP, 16X RCP, 32X RCP and 64X RCP, apart from the normal 1X RCP. Actually, 1X RCP corresponds to the receive data rate.

Fig. 9i.14(a) shows the situations when the line is idle (i.e., in state '1') and is hit by a noise impulse. The receive clock pulse (RCP) is set at 1X RCP. The figure shows the uneventful situation of the clock pulse sampling the input line at the instant the noise is present (point AA') and the circuit detects a low. This gives rise to an invalid start bit and the subsequent clocks will interprete the high condition on the data line to be data bits—all at logic 1's. This gives rise to a serious error arising out of an accidental noise pulse.

Fig. 9i.14(b) shows the same situation with the exception that the receiver clock is now made sixteen times faster—i.e., 16X RCP. Once a low is detected, the receiver is made to wait for seven clock cycles before it resamples the input data (this corresponds to BB' in the fig.). Since in this case the receiver analyses the input line status to be '1', hence it concludes that the low input line status that it detected seven clock cycles earlier to be a noise pulse. Thus the possibility of the UART receiver accepting spurious noise pulse is eliminated. This can further be improved by increasing the clock rate to 32X, 64X, etc.

Fig. 9i.14(c) shows the input line scanned by the same 16X RCP. It shows a valid start bit followed by data bits. As in Fig. (b), here also the receiver waits for seven clock cycles after detecting a low. Here, the receiver detects a low for the second time and comes to the conclusion that a valid start bit has arrived. Thus a valid start bit is detected at CC'. Thereafter the input data is sampled once every 16 clock cycles—this makes the sample rate equal to the receive data rate. This way the stop bit is detected and immediately the receiver goes into start bit verification mode.

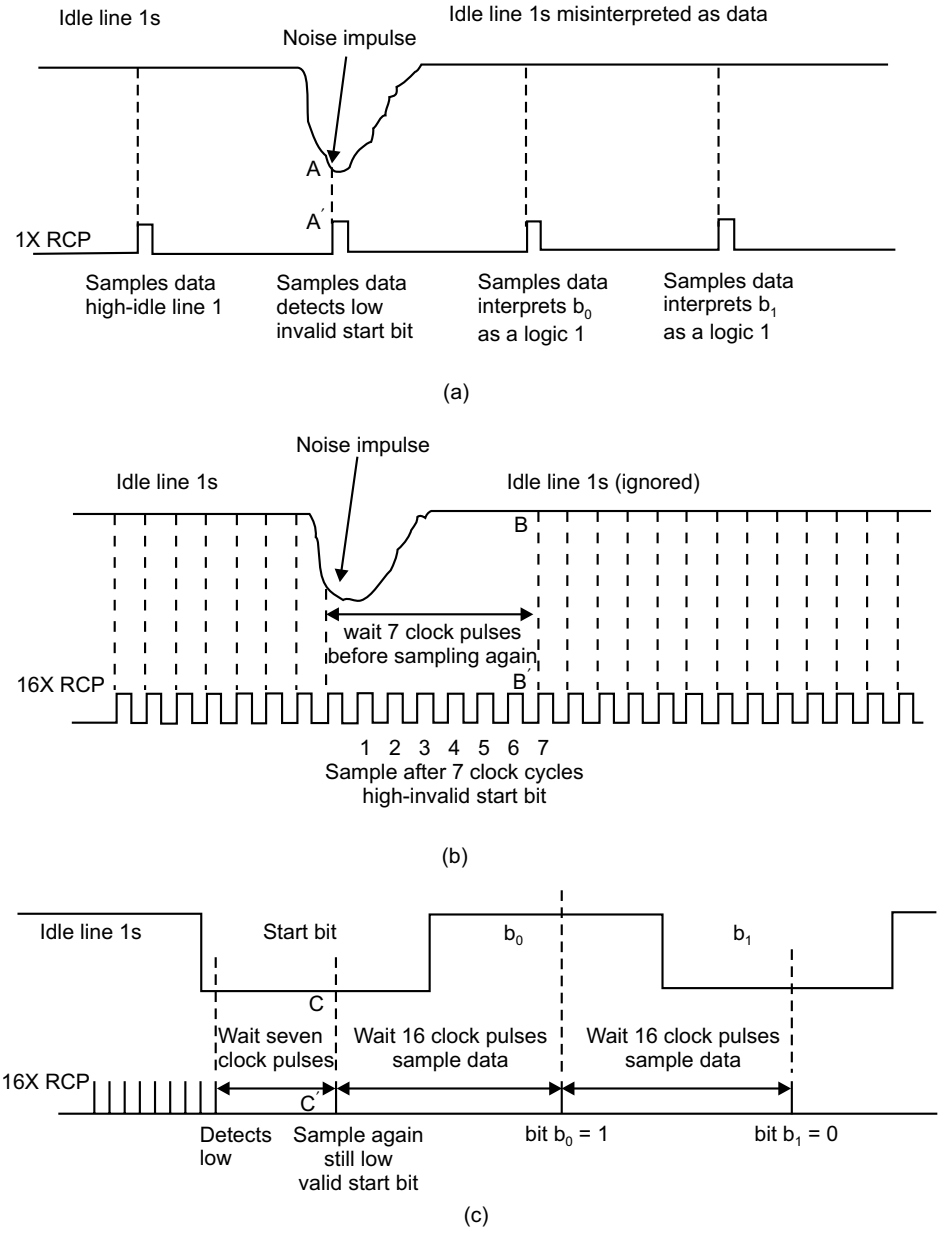


Fig. 9i.14: Start bit verification (a) 1X RCP, (b) 16 X RCP; (c) valid start bit.

32. Explain detection error and sampling error.

Ans. The situation is explained in Figure 9i.15 by clocking the UART receiver with 16X.

The difference in time between the beginning of a start bit and its detection is called detection error and is shown by t_d in the figure. The maximum time of detection is one RCP.

The difference in time between when a sample is taken (i.e., put into the receive shift register) and the actual centre of a data bit is called the sampling error and is shown by t_s in Fig. 9i.15.

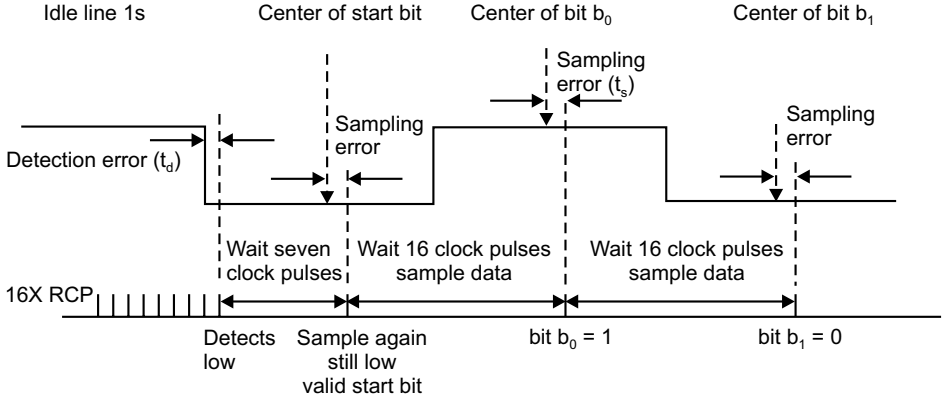


Fig. 9i.15: 16X receive clock rate.

33. What happens to the maximum detection error when receive clock rate equals the receive data rate?

Ans. The maximum detection error would approach one bit time. Thus a start bit would not be detected until the very end of the start bit.

34. What is meant by clock slippage?

Ans. Clock slippage, also known by the name of clock skew, is a problem faced in asynchronous communication system.

In this case, the magnitude of sampling error increases with each successive sample in the data bit pattern. Thus, the clock may slip over or slip under the data.

Sometimes it may so happen that a data bit (it would start occurring for latter data bits in the data stream) may be sampled twice or not sampled at all in the clock period—it depends on whether the receive clock is higher or lower the transmit clock.

35. How the sampling error is related to sampling rate?

Ans. As the sampling rate is continued to be increased, the sampling error goes on decreasing. As the sampling rate is increased, the sample time moves closer and closer to the centre of data bit, thereby decreasing the sampling error.