1. **Define computer peripheral interfacing with example.**

Ans: Computer Peripheral Interfacing is the interactions between computer processor and computer peripherals. Any input or output process that occurs through a peripheral device is known as computer peripheral interfacing. Peripheral devices are those devices that are linked either internally or externally to a computer. These devices are commonly used to transfer data. The most common processes that are carried out in a computer are entering data and displaying processed data. Several devices can be used to receive data and display processed data. The devices used to perform these functions are called peripherals or I/O devices. Peripherals read information from or write in the memory unit on receiving a command from the CPU. Example: Keyboard, Printer, Magnetic Tape, Magnetic Disk.

2. **Compare port-addressed I/O and memory-mapped I/o on basis of address length, control signal, and instruction.**

Ans:

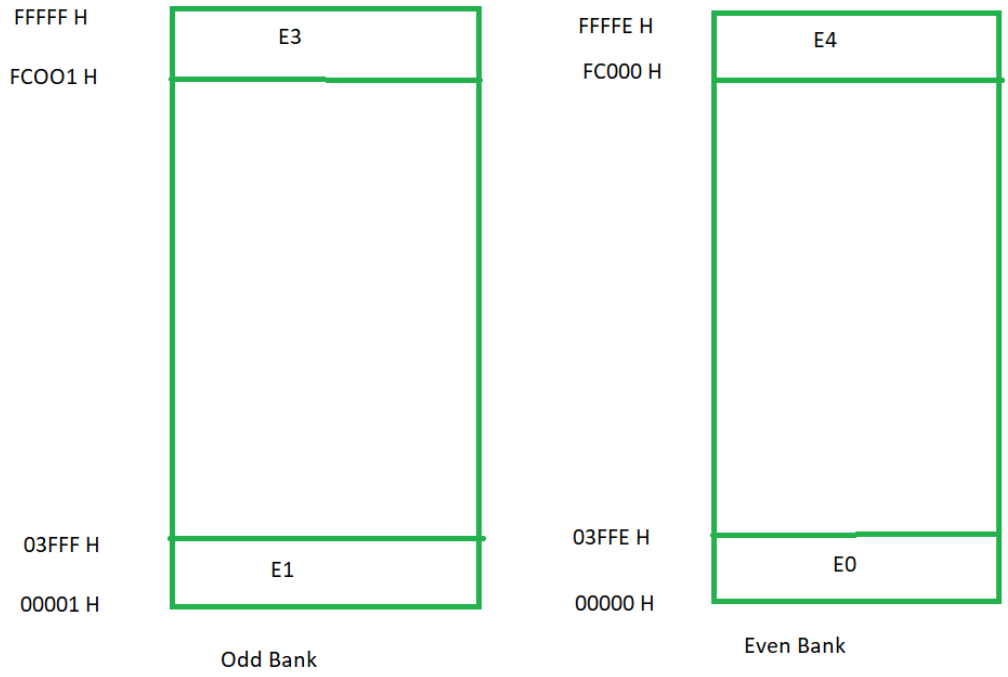
FEATURES	MEMORY MAPPED IO	IO MAPPED IO
ADDRESSING	IO devices are accessed like any other memory location.	They cannot be accessed like any other memory location.
ADDRESS SIZE	They are assigned with 16-bit address values.	They are assigned with 8-bit address values.
INSTRUCTIONS USED	The instruction used are LDA and STA, etc.	The instruction used are IN and OUT.
CYCLES	Cycles involved during operation are Memory Read, Memory Write.	Cycles involved during operation are IO read and IO writes in the case of IO Mapped IO.

REGISTERS COMMUNICATING	Any register can communicate with the IO device in case of Memory Mapped IO.	Only Accumulator can communicate with IO devices in case of IO Mapped IO.
SPACE INVOLVED	2^{16} IO ports are possible to be used for interfacing in case of Memory Mapped IO.	Only 256 I/O ports are available for interfacing in case of IO Mapped IO.
IO/M` SIGNAL	During writing or read cycles (IO/M` = 0) in case of Memory Mapped IO.	During writing or read cycles (IO/M` = 1) in case of IO Mapped IO.
CONTROL SIGNAL	No separate control signal required since we have unified memory space in the case of Memory Mapped IO.	Special control signals are used in the case of IO Mapped IO.
ARITHMETIC AND LOGICAL OPERATIONS	Arithmetic and logical operations are performed directly on the data in the case of Memory Mapped IO.	Arithmetic and logical operations cannot be performed directly on the data in the case of IO Mapped IO.

3. **What do you mean by odd address bank and even address bank of 8086 processor?**

Ans: The memory chip is equally divided into two parts(banks). One of the banks contains even addresses called **Even bank** and the other contains odd addresses called **Odd bank**. Even bank always gives lower byte So Even bank is also called **Lower bank** (LB) and Odd bank is also called **Higher bank** (HB). This banking scheme allows to access two aligned memory locations from both banks

simultaneously and process 16-bit data transfer. Memory banking doesn't make it compulsory to transfer 16 bits, it facilitates the 16-bit data transfer.



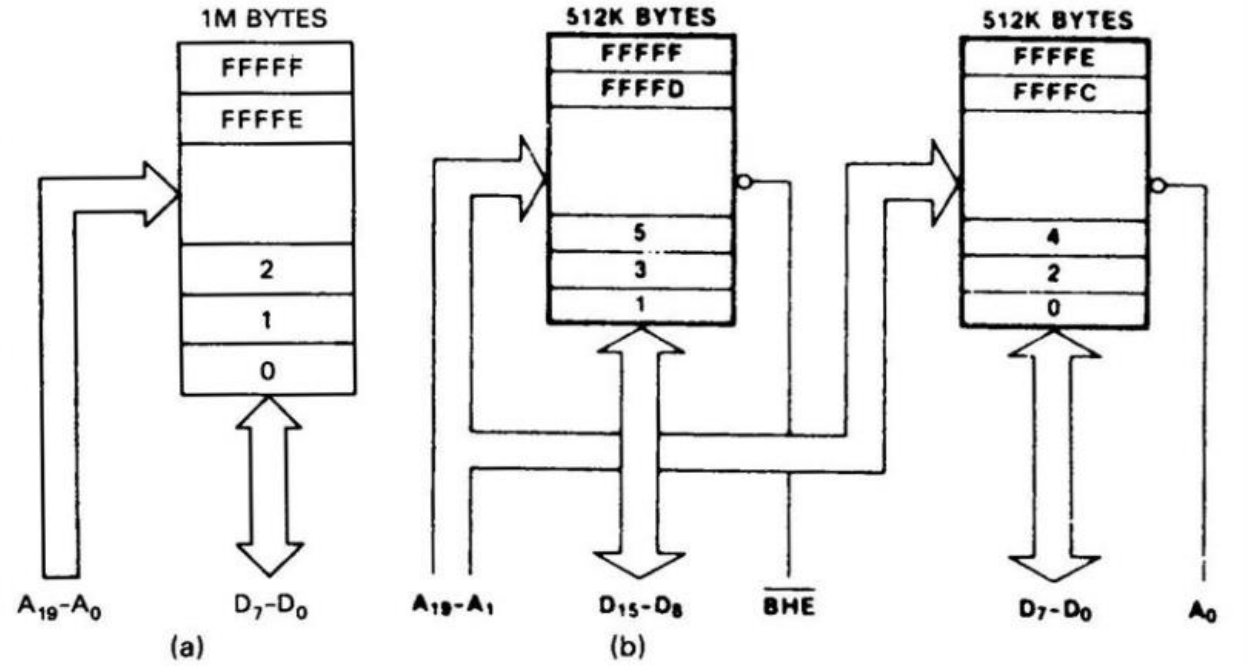
4. Distinguish between absolute and partial address decoding.

Ans:

Absolute decoding	Partial decoding
Used all the address lines [A0-A7] for decoding of IO devices	Few address lines are not used for decoding IO devices
High cost	Less Cost
Happen in case of memory interface	Happen in case of memory interface
More Hardware is required to design decoding logic	Less hardware is required and sometimes it can be eliminate
Used large systems	Used in small syatems

5. Discuss how a byte and a word data is read from the odd boundary address and even boundary address of 8086 processor.

Ans:



- In Figure a memory addresses are not divided into odd and even this figure is given only for comparison.
- Figure b and c show when memory is divided into equal odd and even Bank.
- Since each even and odd bank constitutes 2^19 address, only 19 bits can be used for addressing, a1 to 19 are used for addressing, and a0 is used for enabling / disabling the even bank, another signal, BHE is used for enabling / disabling odd bank. When signal is low a bank is enabled and high bank is disabled.
- A 20-bit address is placed on internal address bus of CPU but a1 to a19 are connected to memory bank for selection of one particular Byte.
- We can conclude from above when even memory address is placed on internal address bus, for example 4 ,

addressed memory and next memory is addressed (4 and 5) match and when odd memory address is placed on internal address bus , say 5 , addressed memory and the previous memory is addressed (5 and 4) are matched.

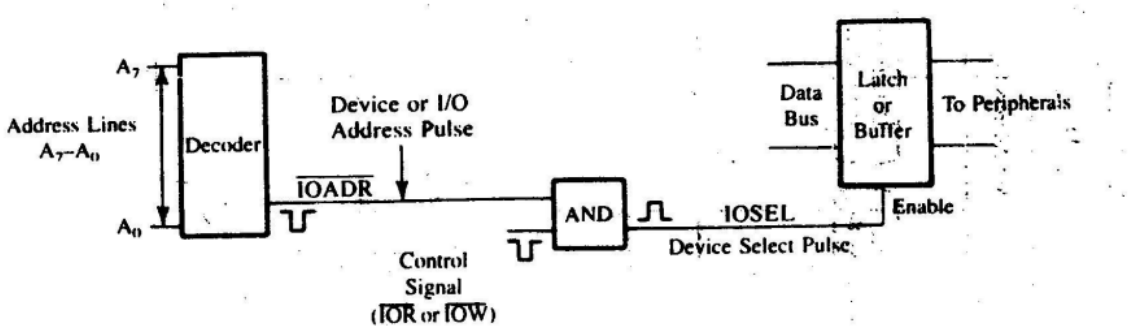
6. **Discuss how address buses and data buses of 8086 microprocessor family are multiplexed and how they are de-multiplexed?**

   Intel 8086 is a 16-bit HMOS microprocessor. It is available in 40 pin DIP chip. It uses a 5V DC supply for its operation. The 8086 uses 20-line address bus. It has a 16-line data bus. The 20 lines of the address bus operate in multiplexed mode. The 16-low order address bus lines have been multiplexed with data and 4 high-order address bus lines have been multiplexed with status signals.

7. **Discuss how device select pulse is generated prior to transfer data in between processor and I/O device.**

Ans:

a) Decode the address bus to generate a unique pulse called the device address pulse or I/O address pulse.
b) Combine (AND) the device address pulse with the control signal to generate a device select pulse (I/O select pulse).
c) Use the I/O select pulse to activate the interfacing device (I/O port).



8. Draw the circuitry to interface eight DIP switches using a 74LS138 decoder and a 74LS244 buffer.

Ans: The circuit used for interfacing eight DIP switches, as shown in Figure 4.11. The circuit shows the 74LS138 3-to-8 decoder to decode the low-order bus and tri-state octal buffer (74LS244) to interfaces the switches to the data bus. The port can be accessed with the address 84H. The 74LS244 tri-state octal buffer used as an interfacing device. When OE signal goes low, the input data show up on the

output lines (connected to the data bus).

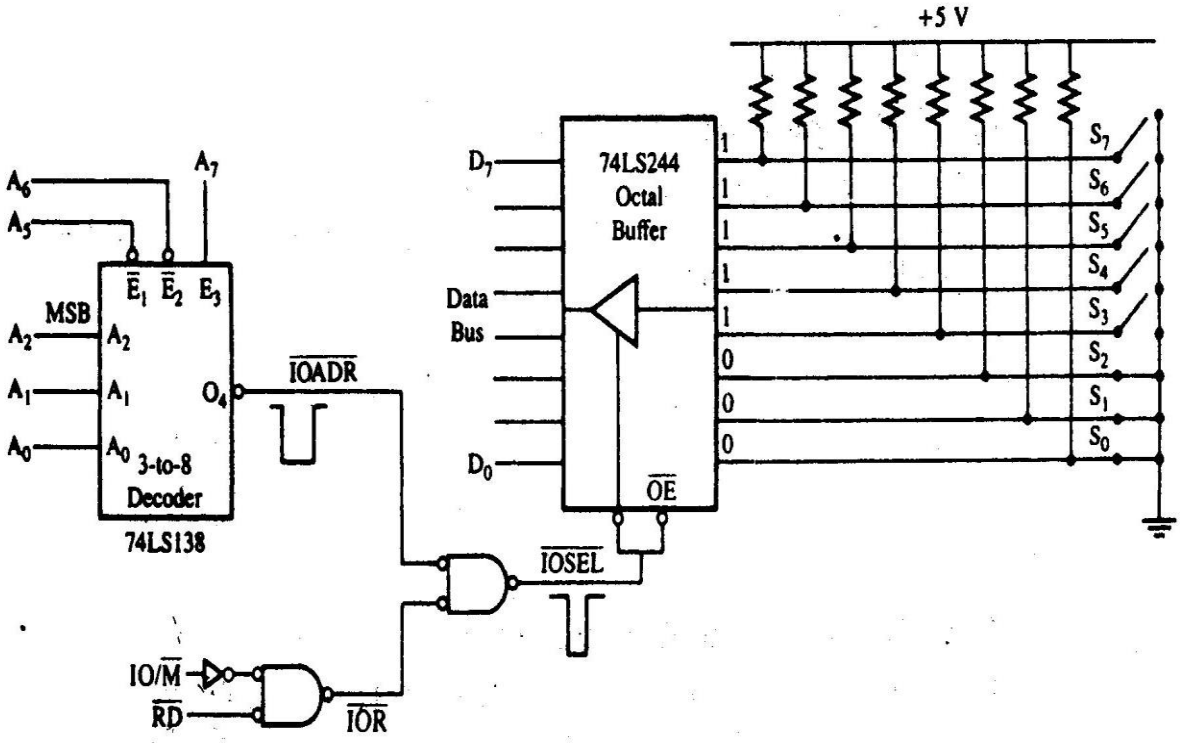
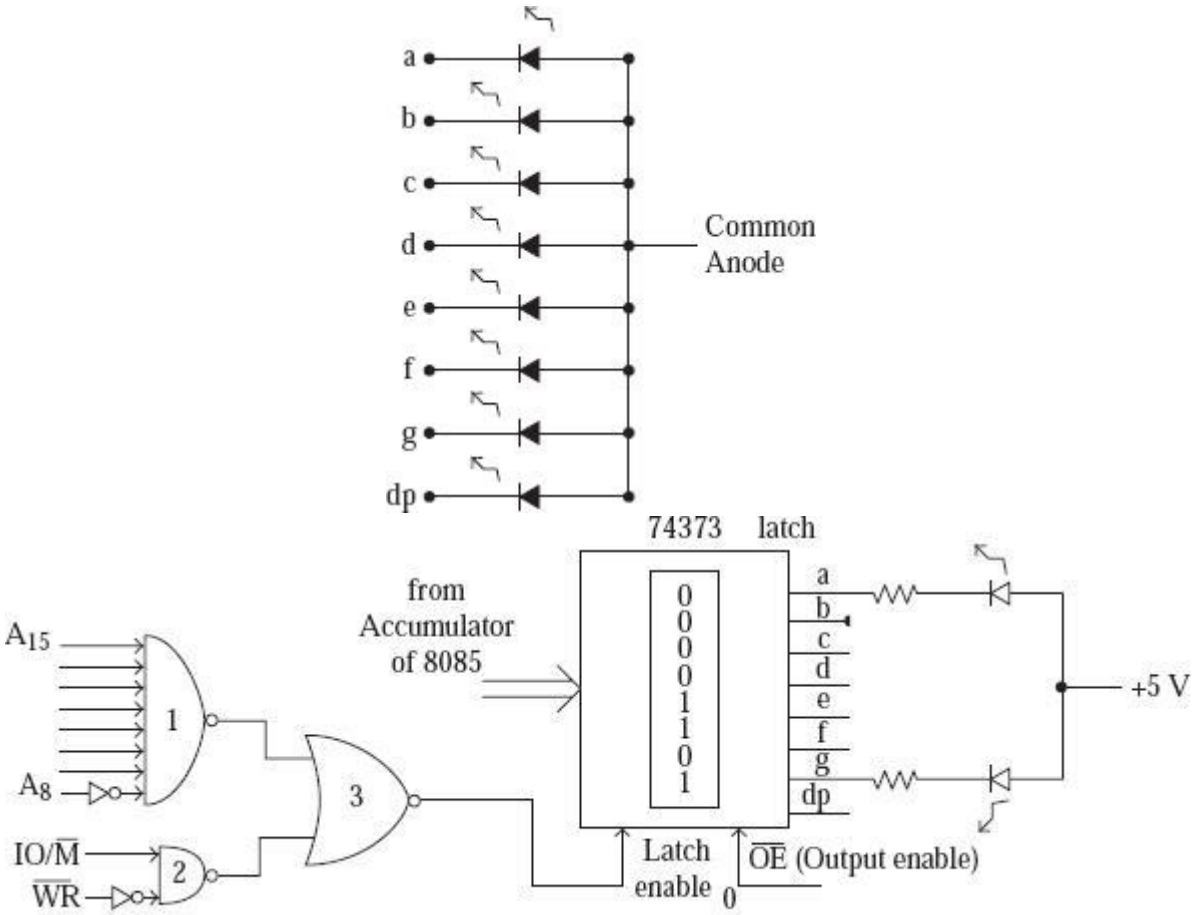


FIGURE 4.11
Interfacing DIP Switches

9. Discuss how to interface common anode seven-segment LED with 8085 with proper circuit diagram. Also write the program to display sequentially digit 0-9 in the seven-segment LED.

Ans: An output device which is very common is, especially in the kit of 8085 microprocessor and it is the Light Emitting Diode consisting of seven segments. Moreover, we have eight segments in a LED display consisting of 7 segments which includes '.', consisting of character 8 and having a decimal point just next to it. We denote the segments as 'a, b, c, d, e, f, g, and dp' where dp signifies '.' which is the decimal point.



Note: To keep the figure simple, only two of the eight segment connections are shown. The other six segment connections are similar.

here are two types of 7-segment LED: They are the common anode type and the common cathode type. We have discussed the common anode-type which is 7 segmented Light Emitting Diode. In the LED which is common anode and is 7-segmented, here we connect all the eight LED anodes together and the eight external pin is brought to display. And this pin gets connected to a DC supply of +5 Volt. The cathode ends of the eight segments are brought out on the pins of the display.

10. What happens when a microprocessor is interrupted?

Ans: An interrupt is an event or signal that requests the CPU's attention. This halt allows peripheral devices to access the microprocessor. Whenever an interrupt occurs, the processor completes the current instruction and starts the

implementation of an Interrupt Service Routine (ISR) or Interrupt Handler.

In the 8086 microprocessor following tasks are performed when the microprocessor encounters an interrupt:

a) The value of the flag register is pushed into the stack. It means that first, the value of SP (Stack Pointer) is decremented by two then the value of the flag register is pushed to the memory address of the stack segment.
b) The value of starting memory address of CS (Code Segment) is pushed into the stack.
c) The value of IP (Instruction Pointer) is pushed into the stack.
d) IP is loaded from word location (Interrupt type) * 04.
e) CS is loaded from the following word location.
f) Interrupt, and Trap flags are reset to 0.

11. Define instruction cycle, machine cycle and T-state.

Ans: Time required to execute and fetch an entire instruction is called *instruction cycle.*

The time required by the microprocessor to complete an operation of accessing memory or input/output devices is called *machine cycle.*

One time period of frequency of microprocessor is called *t-state*. A t-state is measured from the falling edge of one clock pulse to the falling edge of the next clock pulse.

12. Why serial data transfer is mostly preferred over parallel data transfer? Give reasons.

Ans: In Serial Transmission, data-bit flows from one computer to another computer in bi-direction. In this transmission, one-bit flows

at one clock pulse. In Serial Transmission, 8 bits are transferred at a time having a start and stop bit.

In Parallel Transmission, many bits are flow together simultaneously from one computer to another computer. Parallel Transmission is faster than serial transmission to transmit the bits. Parallel transmission is used for short distance.

Generally, Serial Transmission is used for long-distance. Serial Transmission is full duplex as sender can send as well as receive the data. Serial transmission is reliable and straightforward.

13. Mention the functions of READY and HOLD signal of 8086-processor.

Ans:

a) Ready: If the signal of 8085/8080A READY pin is low, the microprocessor enters into a Wait state. The pin is used to synchronize with slower peripherals with microprocessor.

b) Hold: When the HOLD is activated by an external signal, the microprocessor relinquishes control of system bus for external peripheral to use them. For example, HOLD signal is used in DMA data transfer.

14. What is ALE? Discuss how the low-order address buss of 8086 is demultiplexed using ALE signal.

Ans: ALE (Address Latch Enable) : This is a positive going pulse and generated every time 8085 begins an operation; it indicates that bits on AD_7 - AD_0 are address bits. This signal are used to latch the low-order address bus an generate a separate set of eight address lines; A_7 - A_0. The low-order address (05H) is lost after the first clock period. The address needs to be latched for identifying the memory

address. Otherwise, the address 2005H will change to 204FH after the first clock period.

Figure 3.4 shows that the uses of a latch (74LS373) and the ALE signal demultiplexes the low-order bus.

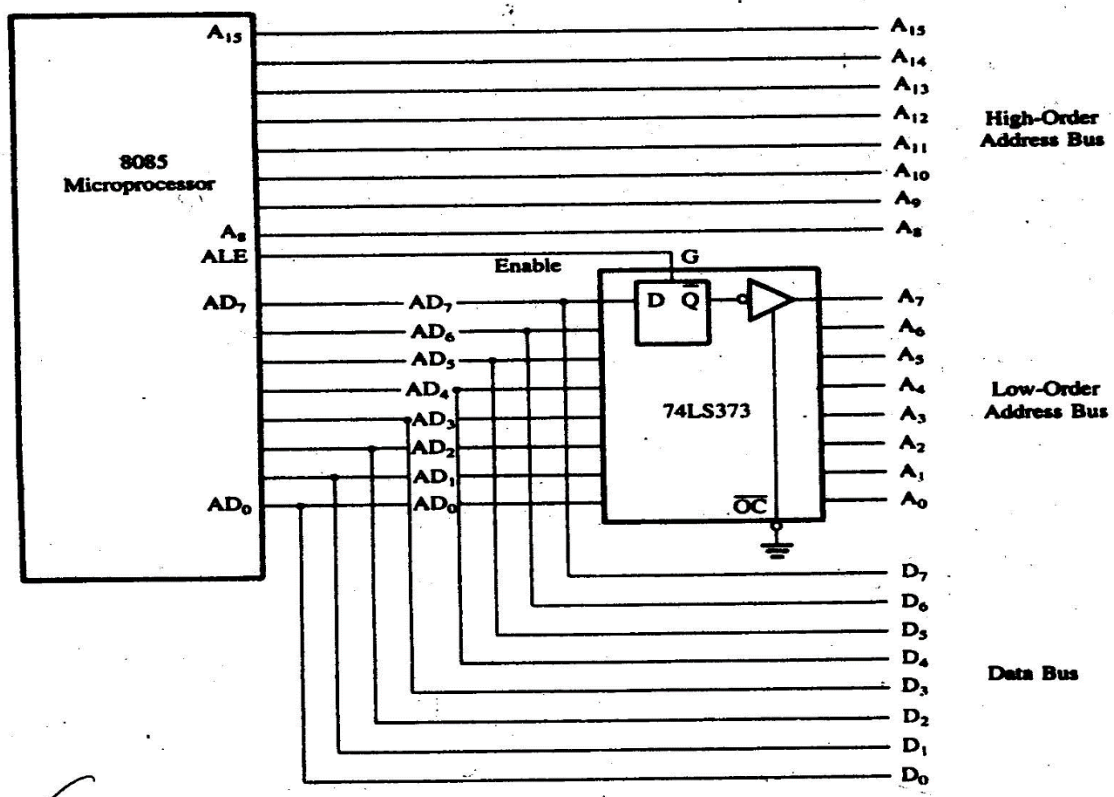


FIGURE 3.4
Schematic of Latching Low-Order Address Bus

15. Safety Control System Using Memory-Mapped I/O Technique

Ans: The various process control devices are connected to the data bus through the latch 74LS373 and solid state relays. When LE is high, the data enter the latch and when LE goes low, data are latched. The latched data are available on the output lines enabled

by OE. To assert the I/O select pulse, the output port address should be FFF8H, as shown below:

A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	
1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	= FFF8H

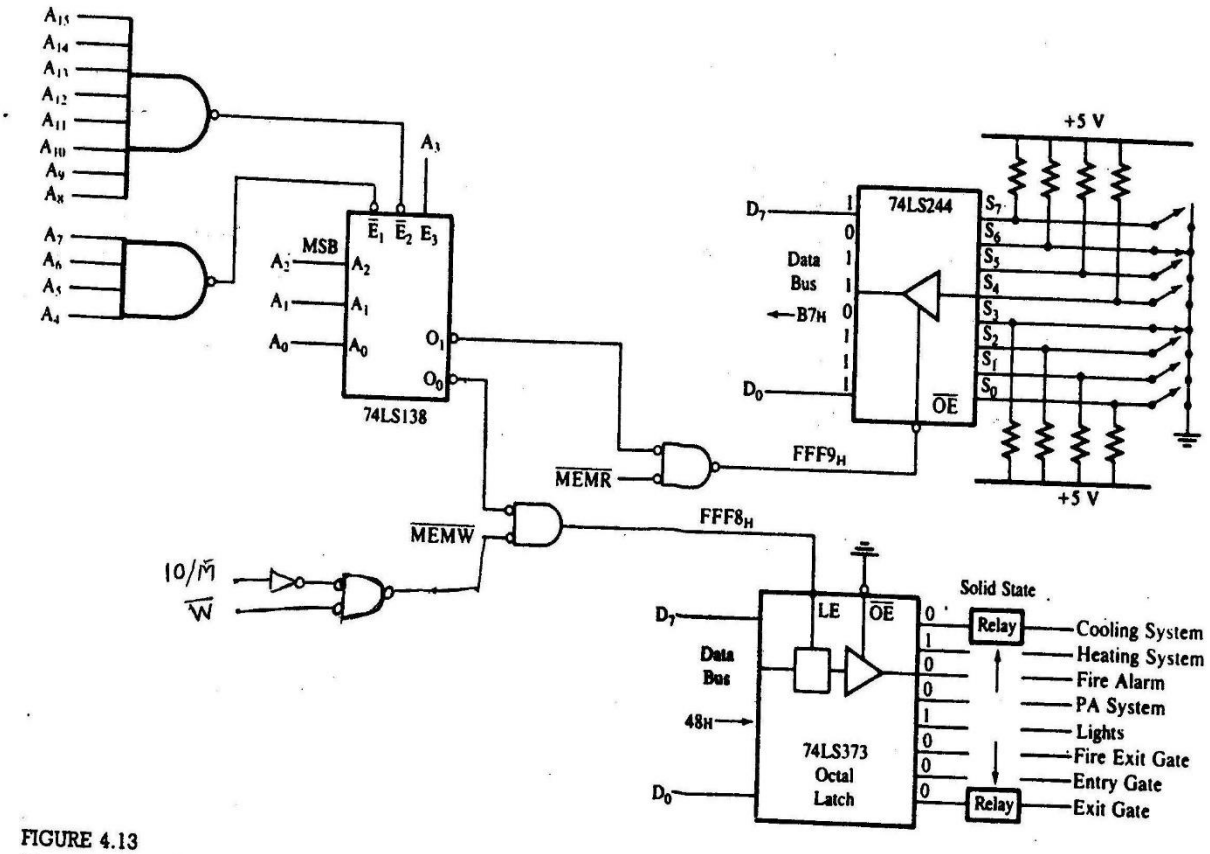


FIGURE 4.13
Memory-Mapped I/O Interfacing

16. Programmable Device with a Handshake Signals

Ans: The MPU and peripherals operate at different speeds; therefore, signals are exchanged prior to data transfer between fast-responding MPU and slowresponding peripherals such as printers and data converters. These signals are called handshake signals. the following similarities among the handshake signals:

1. Handshake signals ACK and STB are input signals to the device and perform similar functions, although they are called by different names.
2. Handshake signals OBF and IBF are output signals from the device and perform similar functions (Buffer Full).

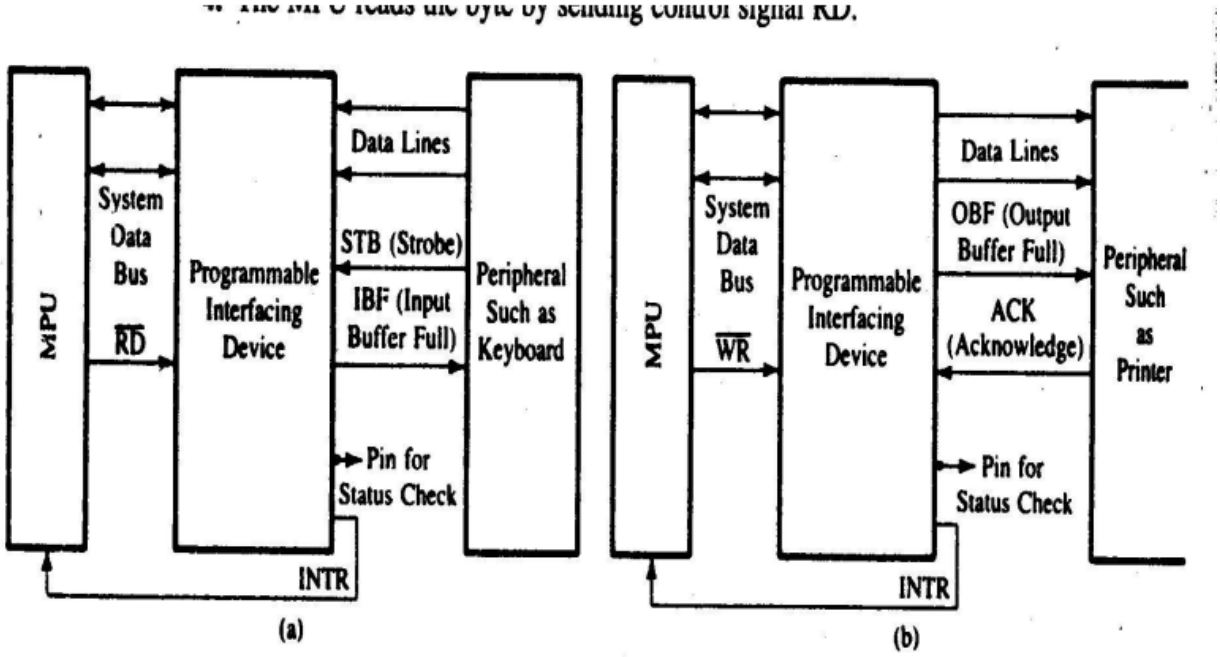


FIGURE 14.4
Interfacing Device with Handshake Signals for Data Input (a) and Data Output (b)

17. Discuss the I/O ports and Timer of 8155.

Ans: The 8155 is a device with two sections: the first is a 256 bytes R/W memory, and the second is a programmable I/O. Functionally, these two sections can be viewed as two independent chips.

The I/O section includes two 8-bit parallel I/O ports (A and B), one 6-bit port (C), and a timer (Figure 14.5). All the ports can be configured simply as input/output ports. Ports A and B also can be programmed in the handshake mode, each port using three signals from port C. The timer is a 14-bit down-counter and has four modes.

THE 8155 I/O PORTS:

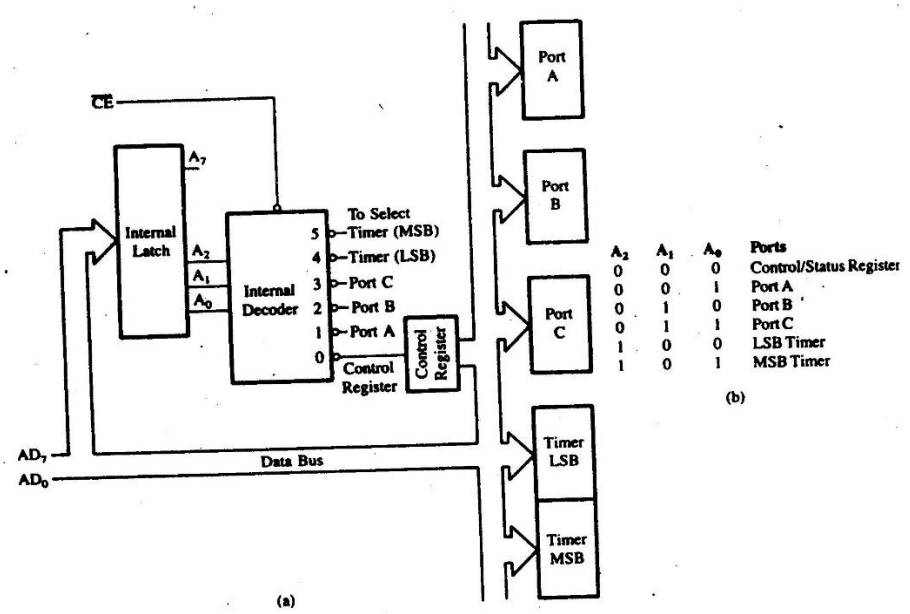


FIGURE 14.6
Expanded Block Diagram of the 8155 (a) and Its I/O Address: Selection (b)

The I/O section of the 8155 includes a control register, three I/O ports, and two registers for the timer (Figure 14.6).

To communicate with the peripherals through the 8155 the following steps are necessary:

1. Determine the address (port numbers of the registers and I/O's) based on

   the Chip Enable logic and address lines AD_0, AD_1, and AD_2.

2. Write control word in the control register to specify I/O functions of the ports

   and the timer characteristics.

3. Write I/O instructions to port addresses to communicate with peripherals.

4. Read the status register, if necessary, to verify the status of the I/O ports

and the timer. In simple applications, this step is not necessary.

18. Design a square-wave generator with a pulse width of 100µs by using the 8155 timer. Set up the timer in mode1 if the clock frequency is 3MHz.

Ans: <u>Timer Count</u>: The pulse width required is 100 µs; therefore, the count should be calculated for the period of 200 µs. The timer output stays high for only half the count.

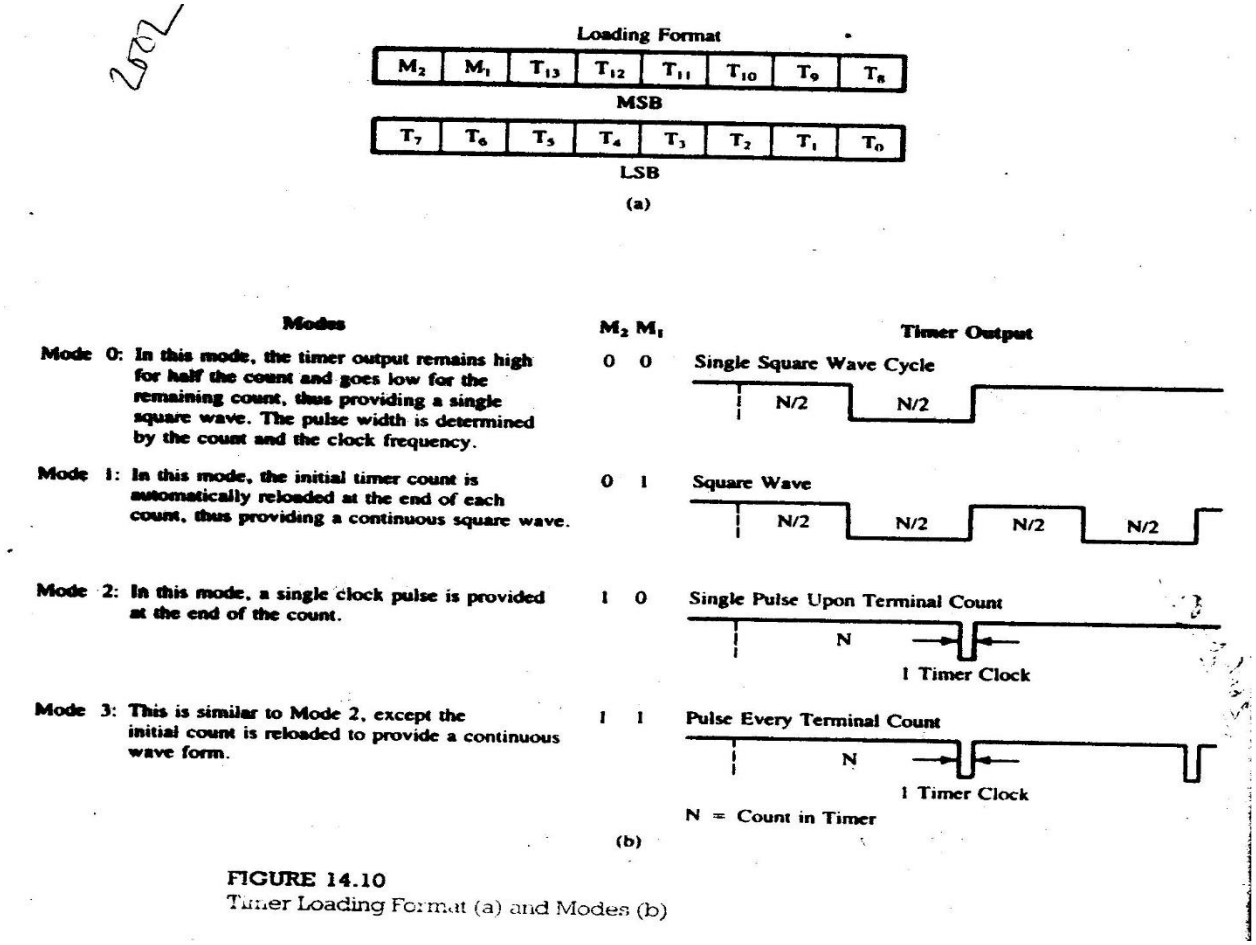


FIGURE 14.10
Timer Loading Format (a) and Modes (b)

Clock Period = 1/f = 1/3 x 10^6 = 330 ns

Timer Count = Pulse Period / Clock Period = 200 x 10^{-6} / 330 x 10^{-9} = 606

Count = 025EH

The port addresses for the timer registers are

Timer LSB = 24H

Timer MSB = 25H

The least significant byte, 5EH (of the count 025EH), should be loaded in the timer register with address 24H. The most significant byte is determined as follows:

M_2	M_1	T_{13}	T_{12}	T_{11}	T_{10}	T_9	T_8		
0	1	0	0	0	0	1	0	=	42H

Therefore, 42H should be loaded in the timer register with the address 25H.

19. Discuss the control signals in handshake mode for input and output of 8155 ports with timing waveform.

Ans: In the handshake mode, data transfer occurs between the MPU and peripherals using control signals called handshake signals. Two I/O ports of the 8155, A and B, can be configured in the handshake mode; each uses three signals from port C as control signalsWhen both ports A and B are configured in the handshake mode, port A uses the lower three signals of port C (PC0 , PC1 , and PC2), and port

B uses the upper three signals (PC3 , PC4 , and PC5), as shown in Figure 14.11. The function of these signals are:

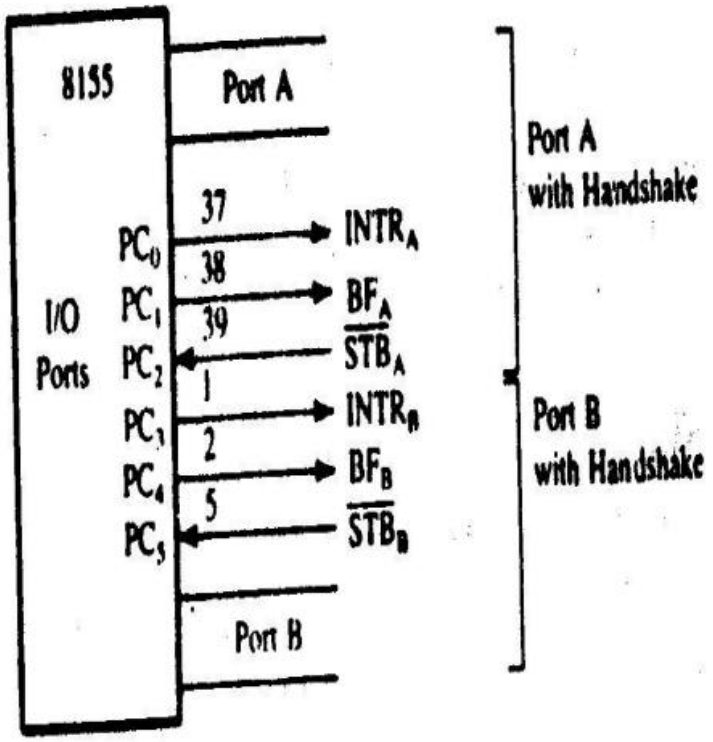
❑ STB (Strobe Input): The low signal on this pin informs the 8155 that data are strobed into the input port.

❑ BF (Buffer Full): The high signal on this pin indicates the presence of a data byte in the port

❑ INTR (Interrupt Request): This signal interrupts the processor to read an existing data from its (8155) input port or write a new data to its output port.

❑ INTE (Interrupt Enable) : This is an internal flip-flop used to enable or disable the interrupt capability of the 8155. The interrupts for port A and B are controlled by bits D4 and D5 respectively, in the control register.

FIGURE 14.11
8155 with Handshake Mode



20. Difference between Maskable and non-maskable interrupt

&

Vectored and non-vectored interrupt.

Ans:

SR.NO.	Maskable Interrupt	Non Maskable Interrupt
1	Maskable interrupt is a hardware Interrupt that can be disabled or ignored by the instructions of CPU.	A non-maskable interrupt is a hardware interrupt that cannot be disabled or ignored by the instructions of CPU.
2	When maskable interrupt occur, it can be handled after executing the current instruction.	When non-maskable interrupts occur, the current instructions and status are stored in stack for the CPU to handle the interrupt.
3	Maskable interrupts help to handle lower priority tasks.	Non-maskable interrupt help to handle higher priority tasks such as watchdog timer.
4	Maskable interrupts used to interface with peripheral device.	Non maskable interrupt used for emergency purpose e.g power failure, smoke detector etc .
5	In maskable interrupts, response time is high.	In non maskable interrupts, response time is low.
6	It may be vectored or non-vectored.	All are vectored interrupts.

SR.NO.	Maskable Interrupt	Non Maskable Interrupt
7	Operation can be masked or made pending.	Operation Cannot be masked or made pending.
8	RST6.5, RST7.5, and RST5.5 of 8085 are some common examples of maskable Interrupts.	Trap of 8085 microprocessor is an example for non-maskable interrupt.

Vectored Interrupt	Non-Vectored Interrupt
The source that interrupts the CPU provides the branch information. This information is called the interrupt vector.	In non-vectored interrupt, the branch address is assigned to the fixed address in the memory
It has a memory address.	It does not have a memory address.
It allows the CPU to be able to know what ISR(Interrupt Service Routine) to carry out in software.	When a non-vectored interrupt is received it jumps into the program counter to fixed address in hardware.

21. 8085 VECTORED INTERRUPTS

Ans :

12.2 8085 VECTORED INTERRUPTS

The 8085 has five interrupt inputs (Figure 12.5). One is called INTR, three are called RST 5.5, RST 6.5, and RST 7.5 respectively, and the fifth is called TRAP, a nonmaskable interrupt. These last four (RSTs and TRAP) are automatically vectored (transferred) to a specific locations on memory page 00H without any external hardware. The do not require the INTA signal or an input port; the

necessary hardware is already implemented inside the 8085. These interrupts and their call locations are as follows:

Interrupts	Call Locations
1. TRAP	0024H
2. RST 7.5	003CH
3. RST 6.5	0034H
4. RST 5.5	002CH

The TRAP has highest priority, followed by RST 7.5, 6.5, 5.5, and INTR, in that order; however, the TRAP has lower priority than the Hold signal used for DMA.

22. ***Can the microprocessor be interrupted again before the completion of the first interrupt service routine?***

Ans: The answer to this question is determined by the programmer. After the first interrupt, the interrupt process is automatically disabled. In the Illustrative program in section 12.12, the service routine enables the interrupt at the end of the service routine; in this case, the microprocessor cannot be interrupted before the completion of this routine. If the instruction EI is written at the beginning of the routine, the microprocessor can be interrupted again during the service routine.

23. **Time Delay Using a Loop within a Loop Technique**

Ans: Label	Opcode	Operand	T-states
	MVI	B,38H	7
LOOP2:	MVI	C,FFH	7
LOOP1:	DCR	C	4
	JNZ	LOOP1	10/7
	DCR	B	4

	JNZ	LOOP2	10/7

DELAY CALCULATIONS

The delay in LOOP1 is T_{L1} = 1783.5 µs. The counter value of LOOP2 is 56_{10} (38H). Delay for the LOOP2 is calculated as follows:

T_{L2} = 56 (T_{L1} + 21 T-states x 0.5 µs)

= 56 (1783.5 µs + 10.5 µs)

= 100.46 ms

24. Time Delay Using a Register Pair

Ans: Here the counter value can be a 16-bit number and maximum of FFFFH.

Label	Opcode	Operand	T-states
	LXI	B,2384H	10
LOOP:	DCX	B	6
	MOV	A,C	4
	ORA	B	4
	JNZ	LOOP	10/7

TIME DELAY

The loop includes 24 clock periods for execution. The decimal equivalent of counter value is

2384H = 9092_{10}

If the clock period of the system = 0.5 µs, the delay in the loop T_L is

T_L = (T x Loop T-states x N_{10})

= (0.5 x 24 x 9092_{10})

≈ 109 ms (without adjusting for the last cycle)

Total Delay T_D = 109 ms + T_O

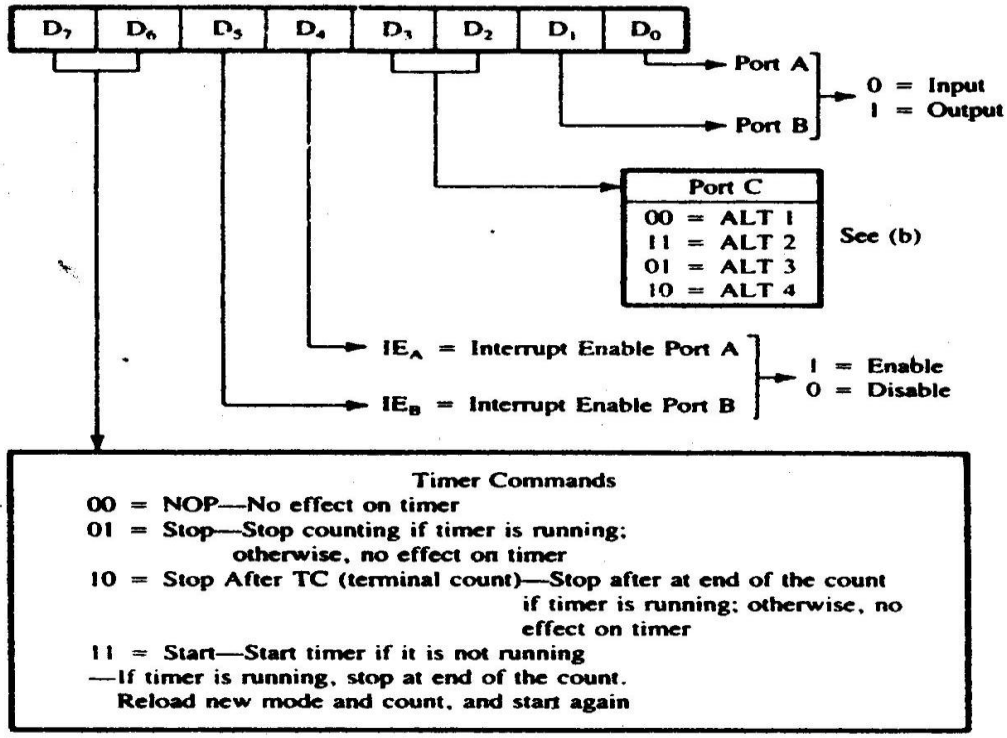
≈ 109 ms (the instruction LXI adds only 5 μs)

25. What is timer? Write the control word format of 8253/8155 programmable interval timer.

Ans: A **timer** is a specialized type of clock which is used to measure time intervals.

CONTROL WORD 8155

The I/O ports and the timer can be configured by writing a control word in the control register. Bit D_2 and D_3 determine the functions of port C; their combination specifies one of the four alternatives, from simple I/O to interrupt I/O, as shown in Figure 14.8(b). Bits D_4 and D_5 are used only in the interrupt mode to enable or disable internal flip-flops of the 8155. These bits do not have any effect on the Internal Enable (INTE) flip-flop of the MPU.



(a)

Table: ALT 1–ALT 4: Port C Bit Assignments, Defined by Bits D_3 and D_2 in the Control Register

ALT	D_3	D_2	PC_5	PC_4	PC_3	PC_2	PC_1	PC_0
ALT 1	0	0	I	I	I	I	I	I
ALT 2	1	1	O	O	O	O	O	O
ALT 3	0	1	O	O	O	$\overline{STB}_A$	BF_A	$INTR_A$
ALT 4	1	0	$\overline{STB}_B$	BF_B	$INTR_B$	$\overline{STB}_A$	BF_A	$INTR_A$

I = Input, STB = Strobe, INTR = Interrupt Request
O = Output, BF = Buffer Full, Subscript A = Port A
B = Port B

(b)

FIGURE 14.8
Control Word Definition in the 8155 (a) and Table of Port C Bit Assignments (b)

8253/8254

8254 Control Word Format

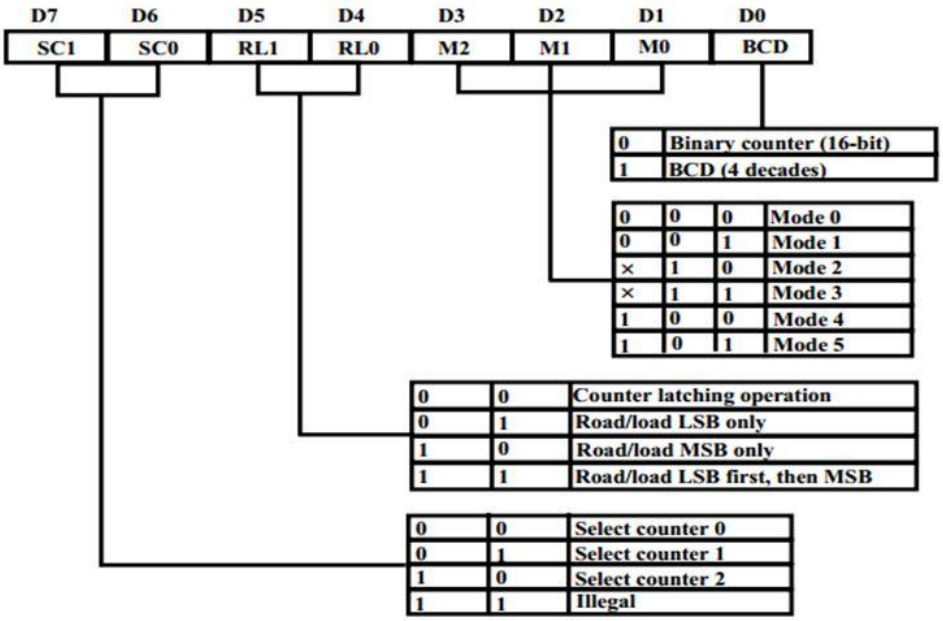


Fig. 14.9 Control Register

26. Discuss about successive-approximation A/D converter.

Ans: Figure 13.9(a) shows the block diagram of a successive approximation A/D converter includes three major elements: the D/A converter, the successive approximation register (SAR), and the comparator. The conversion technique involves the output of the D/A converter V_o with the analog input signal V_{in}. The digital input to the DAC is generated using the successive-approximation method (explain below). When the DAC output matches the analog signal, the input to the DAC is equivalent digital signal.

In the case of a 4-bit A/D converter, bit D_3 is turned on first and the output of the DAC is compared with an analog signal. If the comparator changes the state, indicating that the output generated

by D_3 is larger than the analog signal, bit D_3 is turned off in the SAR and bit D_2 is turned on. The process continues until the input reaches bit D_0.

Figure 13.9(b) illustrates a 4-bit conversion process. When bit D_3 is turned on, the output exceeds the analog signal and therefore, D_3 is turned off. When the next three successive bits are turned on, the output becomes approximately equal to the analog signal.

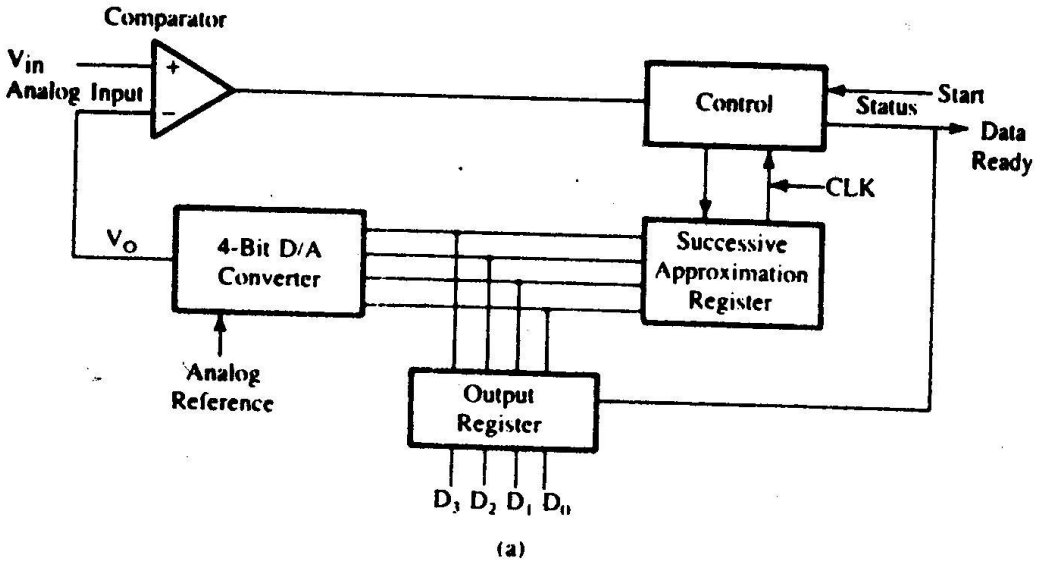
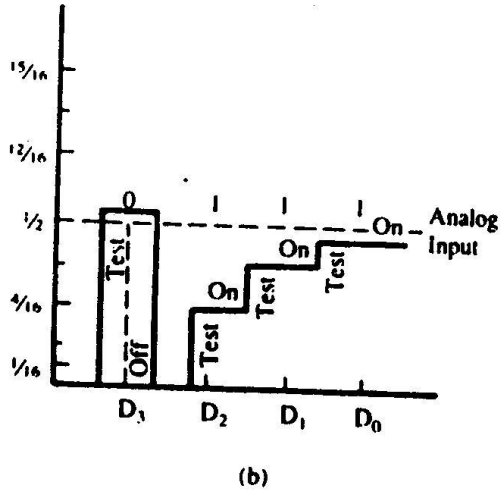


FIGURE 13.9
Successive-Approximation A/D Converter: Block Diagram (a) and Conversion Process for a 4-Bit Converter (b)



27. Draw and describe the functional diagram of 8257 DMA controller.

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

Following is the sequence of operations performed by a DMA −

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

28. What is DMA? What are uses of DMA?

Ans: DMA Controller is a hardware device that allows I/O devices to directly access memory with less participation of the processor. DMA controller needs the same old circuits of an interface to communicate with the CPU and Input/output devices.

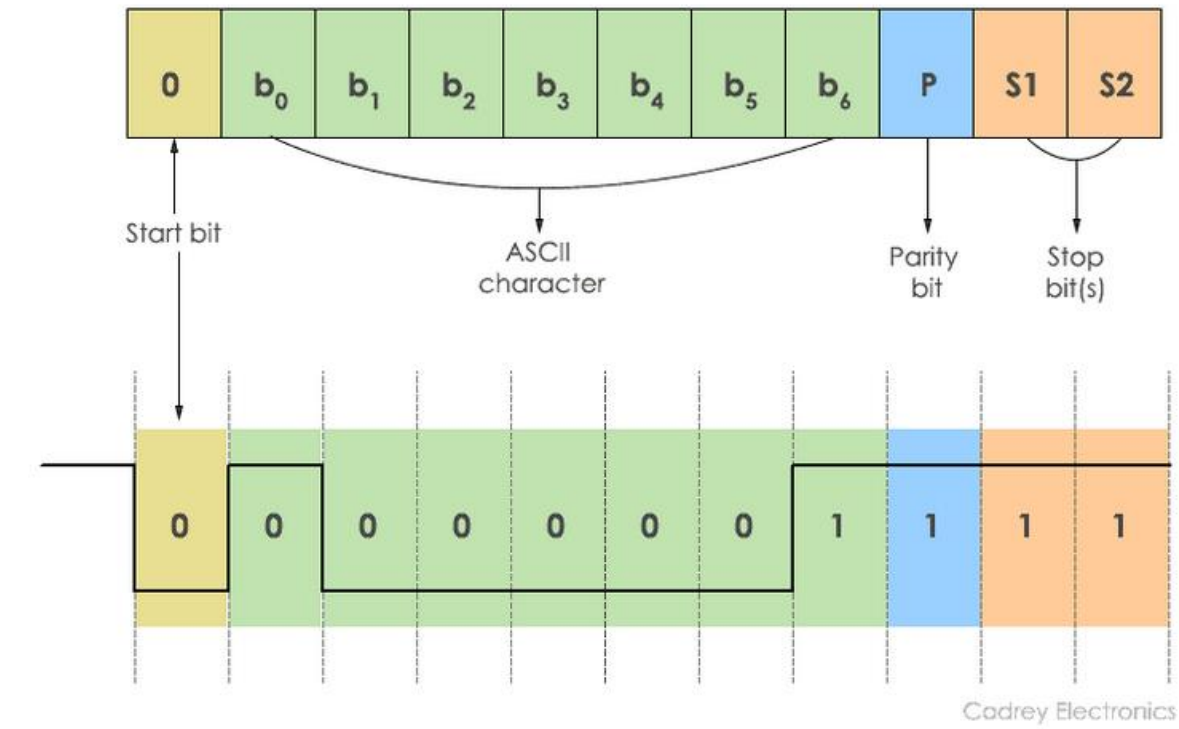
The Uses of DMA are:

- read or write command, sending through read and write control lines.
- number of words to be read or written, communicated on the data lines and stored in the data count register.
- starting location in memory to read from or write to, communicated on data lines and stored in the address register.
- address of the I/O device involved, communicated on the data lines.

SECTION B

29. Discuss the role of RS232 interface in detail?

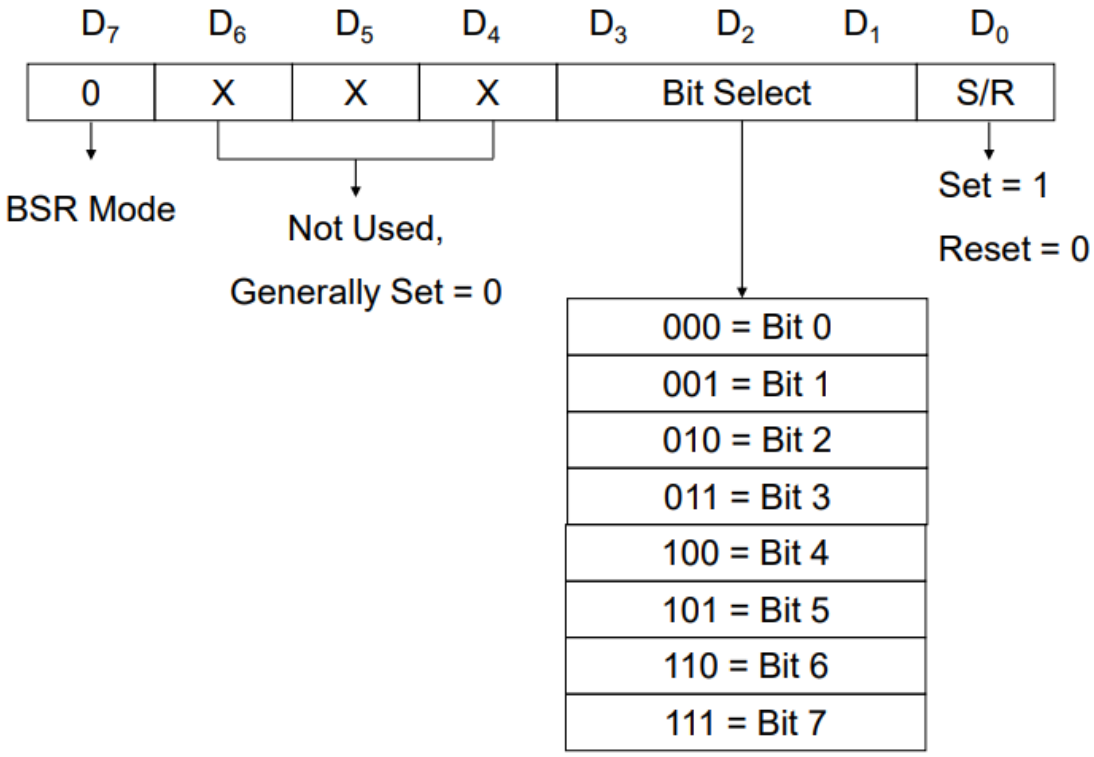
Ans: The working of RS-232 can be understood by the protocol format. As RS-232 is a point-to-point asynchronous communication protocol, it sends data in a single direction. Here, no clock is required for synchronizing the transmitter and receiver. The data format is initiated with a start bit followed by 7-bit binary data, parity bit and stop bit which are sent one after another.



RS232 Framing

30. 8255A control word for BSR mode.

Ans : The BSR mode is concerned only with the eight bits of port C, which can be set or reset by writing an appropriate control word in the control register. A control word with bit D7=0 is recognize as a BSR control word, and it does not alter any previously transmitted control word with bit D7=1; thus the I/O operations of ports A and B are not affected by a BSR control word. In the BSR mode, individual bits of port C can be used for applications such as an on/off switch.



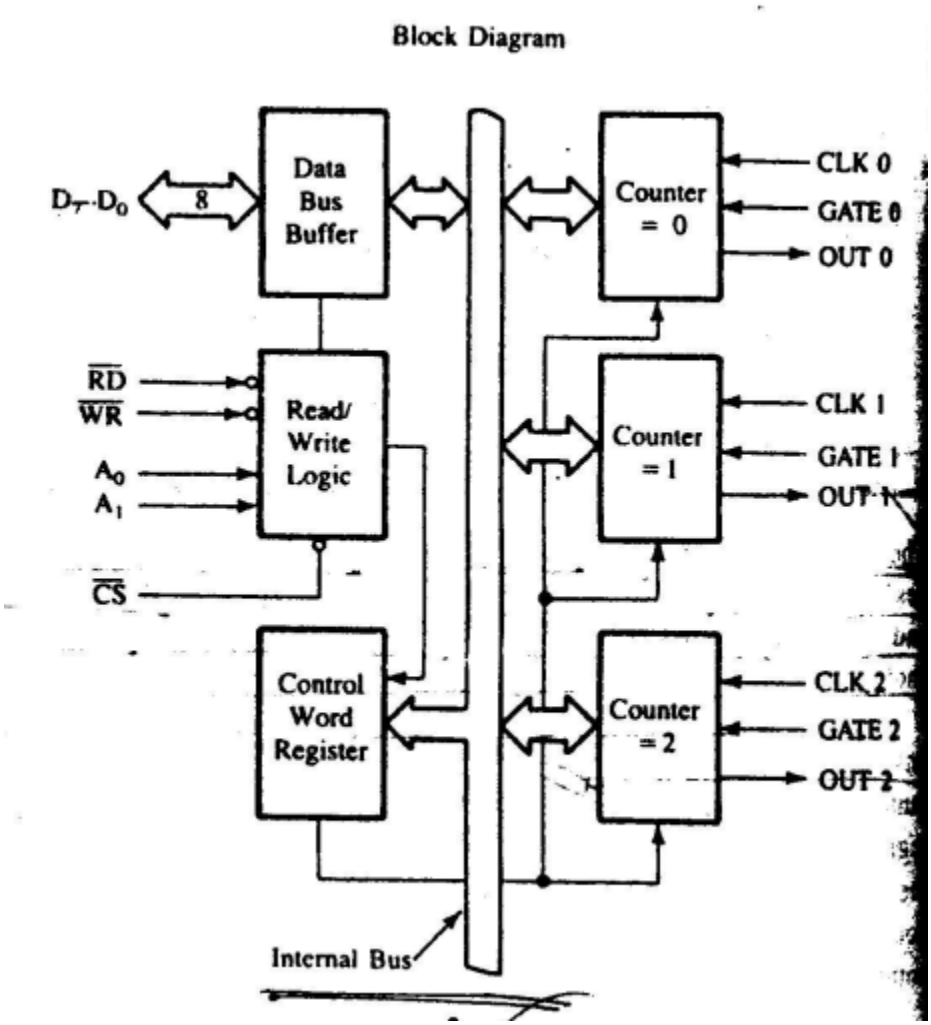
BSR Control Words

	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	
To set bit PC_7 =	0	0	0	0	1	1	1	1	= 0FH
To reset bit PC_7 =	0	0	0	0	1	1	1	0	= 0EH
To set bit PC_3 =	0	0	0	0	0	1	1	1	= 07H
To reset bit PC_3 =	0	0	0	0	0	1	1	0	= 06H

31. THE 8254 (8253) PROGRAMMABLE INTERVAL TIMER

Ans: The 8254 programmable interval timer/counter is functionally similar to the software designed counters and timers described in chapter 8. It generates accurate time delay and can be used for applications such as a real-time clock, an event counter, a square-wave generator, and a complex waveform generator. The 8254 includes three identical 16-bit counters that can operate independently in any one of the six modes. It is packaged in a 24-pin DIP and requires a single +5v power supply. To operate a counter, a

16-bit count is loaded in its register and, on command, begins to decrement the count until it reaches 0. At the end of the count, it generates a pulse that can be used to interrupt the MPU. The counter can count in binary or BCD. In addition, count can be read by the MPU while counter is decrementing.



32. Define Programmable Interface Devices. Discuss 8155 I/O

Ans: A programmable interface device is designed to perform various input/output functions. Such a device can be set up to perform specific functions by writing an instruction (or instructions) in its internal register, called control register.

THE 8155 I/O PORTS The I/O section of the 8155 includes a control register, three I/O ports, and two registers for the timer (Figure 14.6). To communicate with the peripherals through the 8155 the following steps are necessary:

1. Determine the address (port numbers of the registers and I/Os) based on the Chip Enable logic and address lines AD0 , AD1 , and AD2 .

2. Write control word in the control register to specify I/O functions of the ports and the timer characteristics.

3. Write I/O instructions to port addresses to communicate with peripherals.

4. Read the status register, if necessary, to verify the status of the I/O ports and the timer. In simple applications, this step is not necessary.

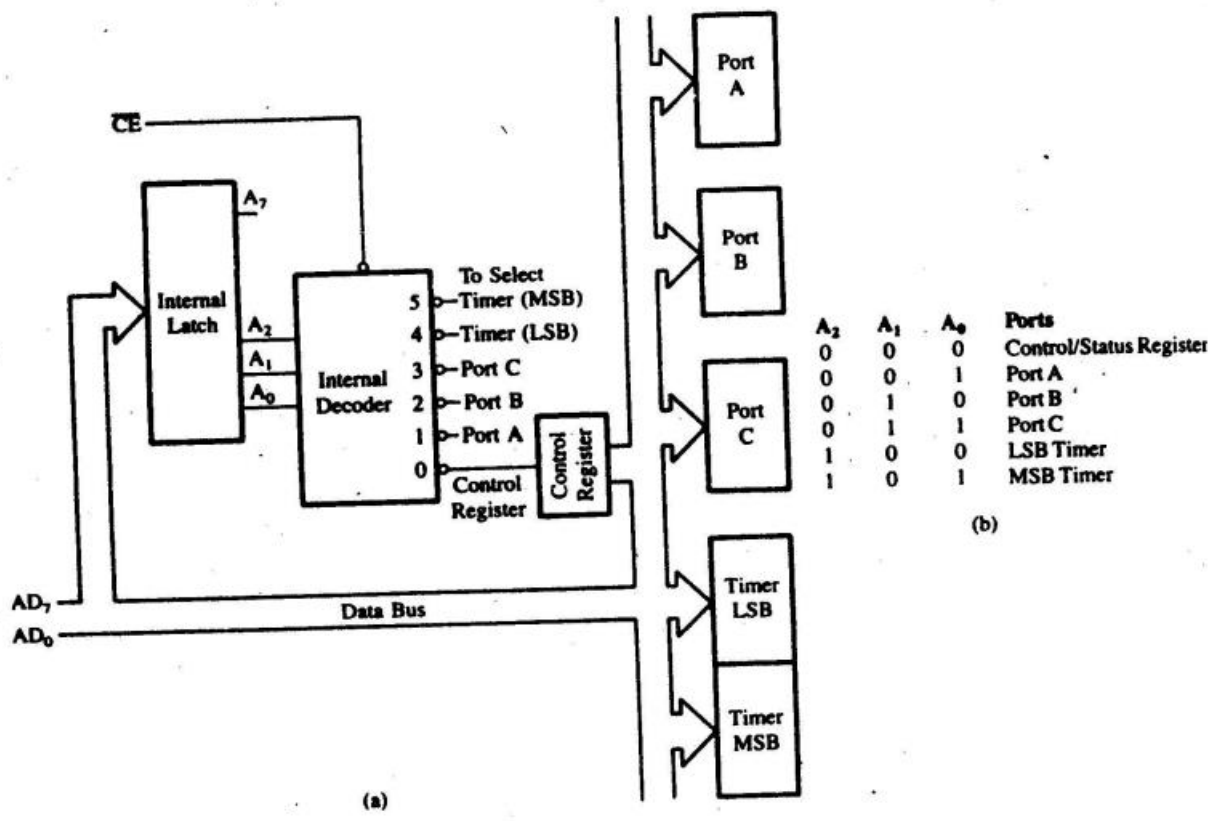


FIGURE 14.6
Expanded Block Diagram of the 8155 (a) and Its I/O Address: Selection (b)

33. Design an output port with the address FFH to interface the 1408 D/A converter that is calibrated for a 0 to 10 V range. Write a program to generate a continuous ramp waveform.

Ans: The total reference current source is determined by the resistor R14 and the voltage VRef . The resistor R15 is generally equal to R14 to match the input impedance of the reference source. The output IO is calculated as follows:

Io = VRef /R14 (A1 /2 + A2 /4 + A3 /8 + A4 /16 + A5 /32 + A6 /64 + A7 /128 + A8 /256)

Where inputs A1 through A8 = 0 or 1.

This formula is an application of the generalized formula for the current Io . For full scale input (D7 through Do = 1).

Io = 5 V /2.5 k (1/2 + 1/4 + 1/8 + 1/16 + 1/32 + 1/64 + 1/128 + 1/256)

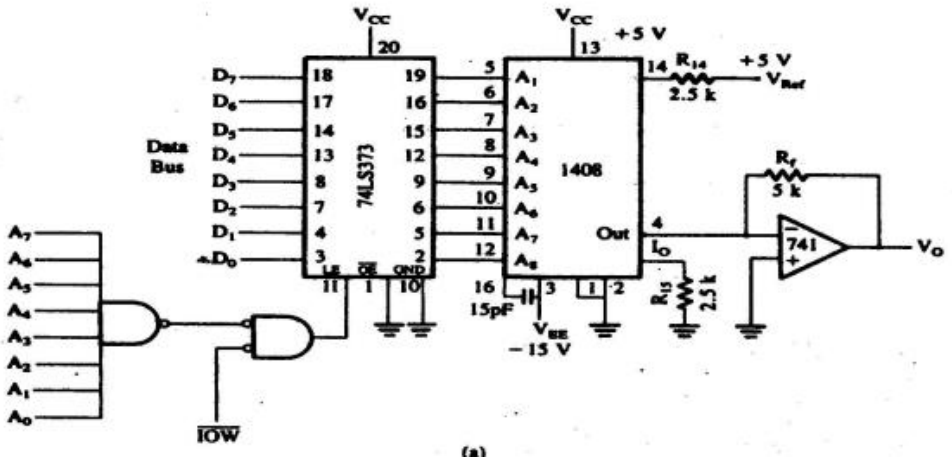
= 2 mA (255/256)

= 1.992 mA.

The output is 1 LSB less than full-scale reference source of 2 mA. The output voltage Vo for the full-scale input is

= 2 mA (255/256) x 5 k

= 9.961 V



PROGRAM

To generate a continuous waveform, the instructions are as follows:

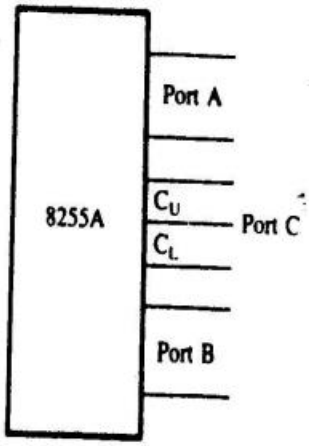
```
        MVI A, 00H      ; First input 00H for DAC
DTOA:   OUT FFH         ; Output to DAC
        MVI B, COUNT    ; COUNT for delay
DELAY:  DCR B
        JNZ DELAY
        INR A           ; Next input for DAC
        JMP DTOA
```

34. Port And Mode of 8255A

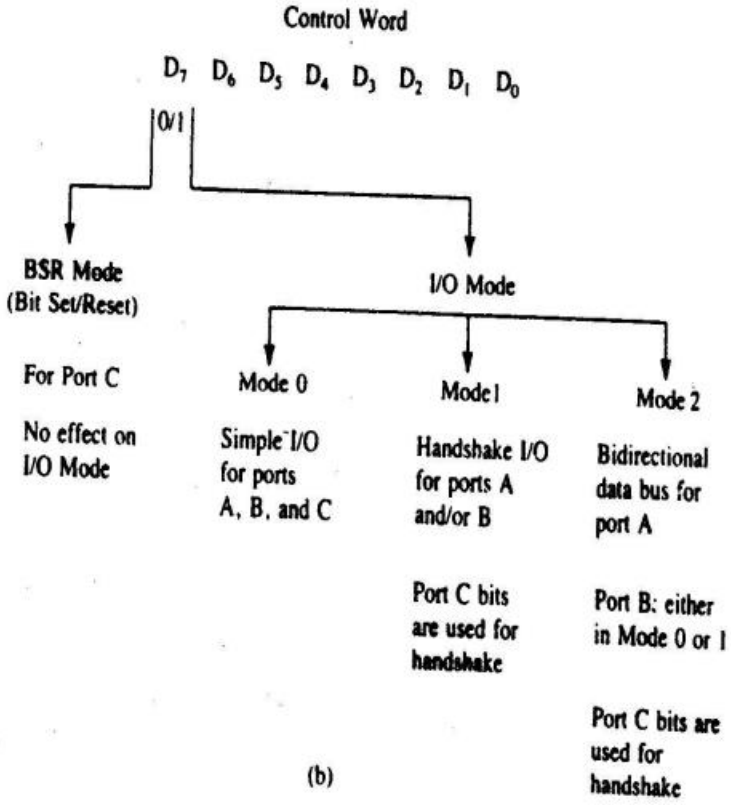
Ans: The 8255A has 24 I/O pins that can be grouped primarily in two 8-bit parallel ports: A and B, with the remaining 8-bits as port C. The eight bits of port C can be used as individual bits or grouped in two 4-bit ports: CUPPER (CU) and CLOWER (CL), as in Figure 15.1(a). The functions of these ports are defined by writing control word in the control register.

Figure 15.1(b) shows all the functions of the 8255A, classified according to two modes: the Bit Set/Reset (BSR) mode and the I/O mode. The BSR mode is used to set or reset the bits in port C. The I/O mode is further divided into three modes: Mode 0, Mode 1, and Mode 2.

- ❑ Mode 0: all ports function as simple I/O ports.
- ❑ Mode 1: handshake mode; ports A and/or B use bits from port C as handshake signals.
- ❑ Mode 2: port A for bidirectional data transfer using handshake signals from port C, and port B can be either Mode 0 or Mode 1.



(a)



(b)

FIGURE 15.1
8255A I/O Ports (a) and Their Modes (b)