Module IV

Interfacing Memory, I/O, 8255 - Detailed study - Architecture, Control word format and modes of operation, Architecture and modes of operation of 8279 and 8257 (Just mention the control word, no need to memorize the control word format)

Interfacing I/O Ports

I/O ports or input/output ports are the devices through which the microprocessor communicates with other devices or external devices.

- Input activity, is the activity that enables the microprocessor to read data from external devices, for example keyboard, joysticks, mouser etc. The devices are known as input devices as they feed data into a microprocessor system.
- Output activity transfers data from the microprocessor to the external devices, for example CRT display, 7 segment displays, printer, etc, the devices that accept the data from a microprocessor system are called output devices.

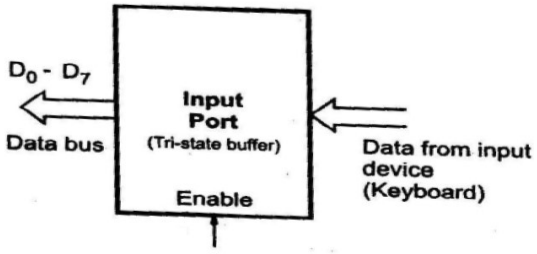
Steps in Interfacing an I/O Device

The following steps are performed to interface a general I/O device with a CPU:

1. Connect the data bus of the microprocessor system with the data bus of the I/O port.
2. Derive a device address pulse and use it as the chip select of the device.
3. Use a suitable control signal, i.e. IORD and /or IOWR to carry out device operations, i.e. connect IORD to RD input of the device if it is an input devise, otherwise connect IOWR to WR input of the device.

Input Port

The input device is connected to the microprocessor through buffer. The simplest form of a input port is a buffer as shown in the figure.

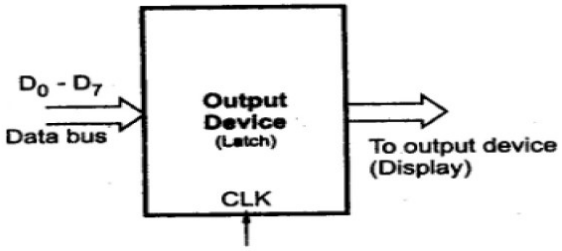


When microprocessor wants to read data from the input device (keyboard), the control signals from the microprocessor activates the buffer by Enable input of the buffer. Once the buffer is enabled, data from the device

is available on the data bus. Microprocessor reads this data by initiating read command.

Output Port

It is used to send the data to the output device such as display from the microprocessor. The simplest form of the output port is a latch.



The output device is connected to the microprocessor through latch as shown in the figure. When microprocessor wants to send data to the output device it puts the data on the data bus and activates the clock signal of the latch.

I/O Interfacing Techniques

Input/output devices can be interfaced with microprocessor systems in two ways :

1. I/O mapped I/O

2. Memory mapped I/O

Difference between Memory Mapped I/O & I/O mapped I/O

Memory Mapped I/O	I/O Mapped I/O
Memory & I/O share the entire address range of processor	Processor provides separate address range for memory & I/O
Processor provides more address lines for accessing memory	Less address lines for accessing I/O
More Decoding is required	Less decoding is required
Memory control signals used to control Read & Write I/O operations	I/O control signals are used to control Read & Write I/O operations

IO mapped IO V/s Memory Mapped IO

Memory Mapped IO	IO Mapped IO
• Memory Instructions are used.	• Special Instructions are used like IN, OUT.
• Memory control signals are used.	• Special control signals are used.
• Arithmetic and logic operations can be performed on data.	• Arithmetic and logic operations can not be performed on data.
• Data transfer b/w register and IO.	• Data transfer b/w accumulator and IO.

34

Problem:

Interface an input port 74LS245 to read the status of the switches SW1 to SW8. the switches when shorted, input a '1' else input a '0' to the microprocessor system. Store the status in register BL. The address of the port is **0740H**

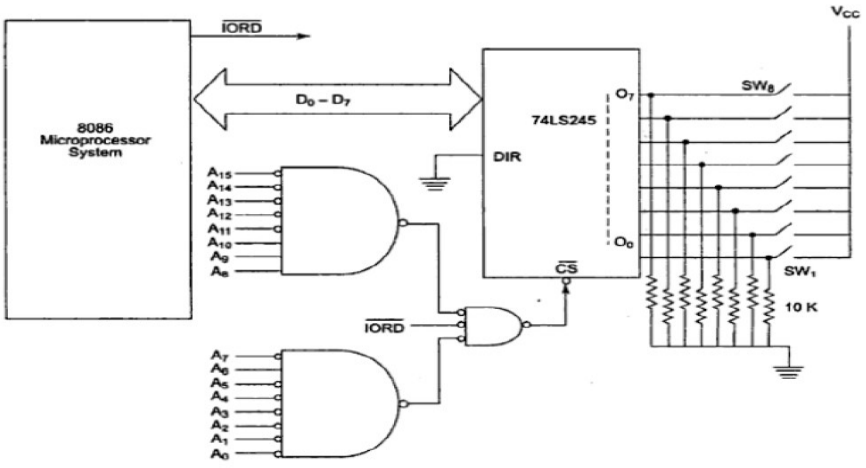


Fig. *Interfacing Input Port 74LS245*

Solution:

The hardware interface circuit is shown in figure. The address, control and data lines are assumed to be readily available at the microprocessor system The ALP is given as follows:

```
MOV BL, 00H        ; clear BL for status
MOV DX, 0740H      ; 16-bit Port address in DX
IN AL,DX           ; Read Port 0740H for switch positions.
```

```
MOV BL, AL          ; Store status of switches from AL into BL
HLT                 ; Stop
```

Here LSB bit of BL corresponds to the status of SW1 and likewise the MSB of BL corresponds to the status of SW8.

Problem:

Design an interface of input port 74LS245 to read the status of switches SW1 to SW8 and output port 74LS373 with 8086. Display the number of key that is pressed with the help of output port on 7 segment display. Write an ALP for this task. The input port address is 08H and output port address is 0AH.

Solution: Status of the switches is first read into the AL. Displaying the shorted switch number in the 7 segment display. Instead of using 16 address lines, one may use only A3– A0.

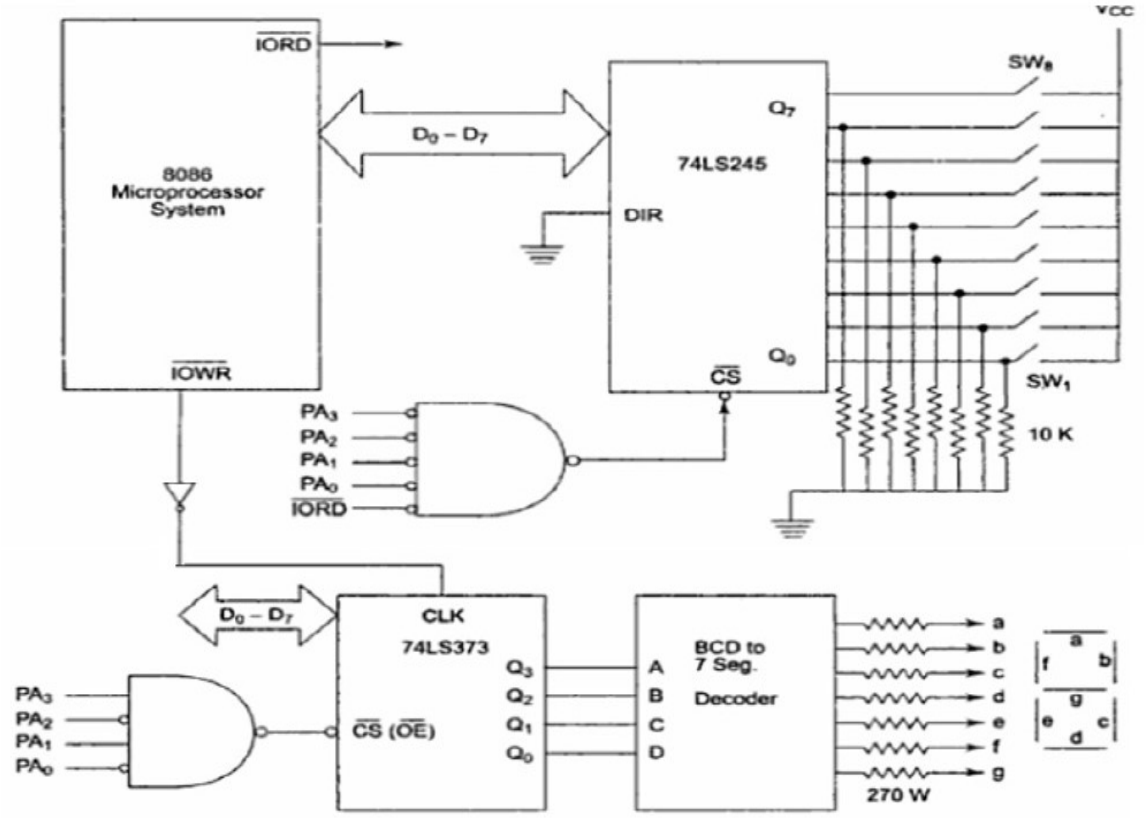


Fig. *Interfacing Switches and Displays for Problem*

I/O Interfacing of 8086 Using 8255A

8255A Programmable Peripheral Interface

8255 is a widely used, programmable parallel I/O device.

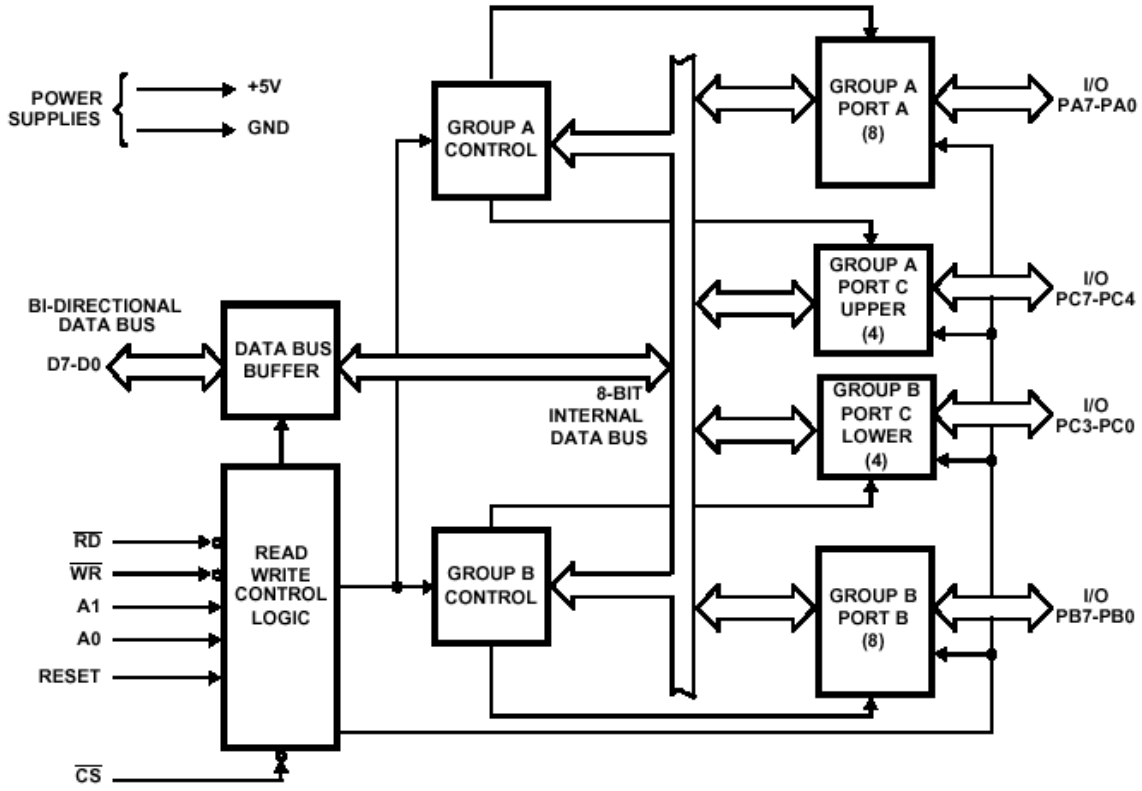
The 8255 has 24 I/O pins that can be grouped into two 8 bit parallel ports: A and B, with the remaining 8 bits as Port C. The 8 bits of port C can be used as individual bits or be grouped into two 4 bit ports : CUpper and CLower .

The functions of these ports are defined by writing a control word in the control register.

The port A lines are identified by symbols PA0-PA7 while the port C lines are identified as PC4-PC7 similarly.

Group A contains an 8-bit port A, containing lines PA0- PA7 and a 4-bit port C with upper bits PC4-PC7. **Group B contains** an 8-bit port B, containing lines PB0- PB7 and a 4-bit port C with lower bits PC0-PC3.

The port C upper and port C lower can be used in combination as an 8-bit port C. All of these ports can function independently either as input or as output ports. This can be achieved by programming the bits of an internal register of 8255 called as control word register (CWR).



Data Bus Buffer

This bi-directional 8-bit buffer is used to interface the 8255A to the system data bus. Data is transmitted or received by the buffer upon execution of input or output instructions by the CPU.

Control words and status information are also transferred through the data bus buffer.

Read/Write and Control Logic- The function of this block is to manage all of the internal and external transfers of both Data and Control or Status words.

(A0 and A1) Port Select 0 and Port Select 1 : These input signals, in conjunction with the RD and WR inputs, control the selection of one of the three ports or the control word register.

A1	A0	SELECTION
0	0	PORT A
0	1	PORT B
1	0	PORT C
1	1	CONTROL

(CS) Chip Select : A "low" on this input pin enables the communication between the 82C55 and the CPU.

(RD) Read : A "low" on this input pin enables 82C55A to send the data or status information to the CPU on the data bus. In essence, it allows the CPU to "read from" the 82C55.

(WR) Write : A "low" on this input pin enables¬ the CPU to write data or control words into the 82C55.

RESET: Logic high on this line clears the control word register of 8255. All ports are set as input ports by default after reset.

Group A and Group B controls:

- Group A and B get the Control Signal from CPU and send the command to the individual control blocks.
- Group A send the control signal to port A and Port C (Upper) PC7-PC4.
- Group B send the control signal to port B and Port C (Lower) PC3-PC0.

PORT A:

- This is a 8-bit buffered I/O latch.
- It can be programmed by mode 0 , mode 1, mode 2 .

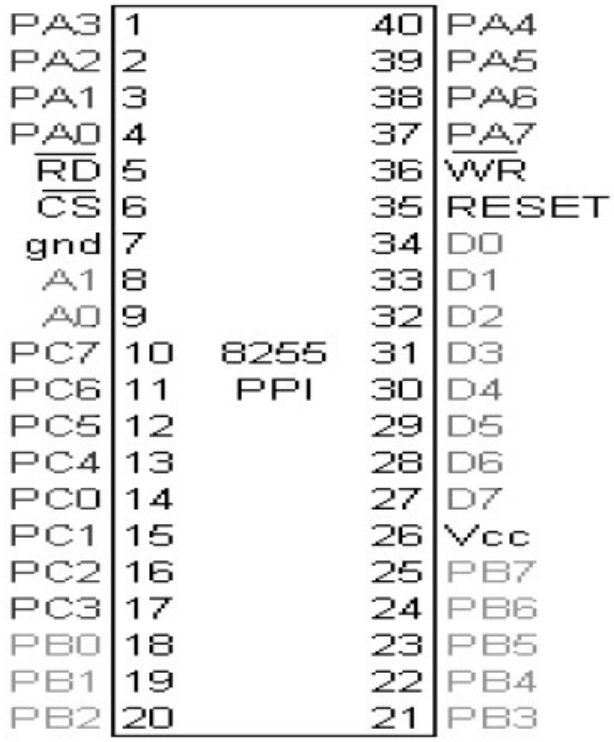
PORT B:

- This is a 8-bit buffer I/O latch.
- It can be programmed by mode 0 and mode 1.

PORT C:

- This is a 8-bit Unlatched buffer Input and an Output latch.
- It is splitted into two parts.
- It can be programmed by bit set/reset operation.

Pin Description of 8255



Working Modes of 8255 Mode Selection

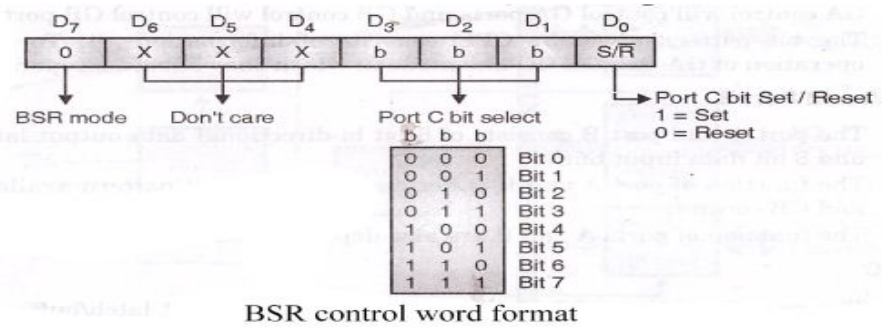
There are three basic modes of operation than can be selected by the system software:

- Mode 0 - Basic Input/Output
- Mode 1 - Strobed Input/Output
- Mode 2 - Bi-directional Bus

BSR mode-(Bit Set/ Reset)

1. The BSR mode is a port C bit set/reset mode.
2. The individual bit of port C can be set or reset by writing control word in the control register.

The control word format of BSR mode is as shown in the figure below



BSR control word format

I/O Modes:

a) Mode 0 (Basic I/O mode): This mode is also called as basic input/output Mode. This mode provides simple input and output capabilities using each of the three ports

The **salient features** of this mode are as listed below:

1. Two 8-bit ports (port A and port B) and two 4-bit ports (port C upper and lower) are available. The two 4-bit ports can be combined used as a third 8-bit port.
2. Any port can be used as an input or output port.
3. Output ports are latched. Input ports are not latched.
4. A maximum of four ports are available

b)Mode1(StrobedInput/Output):

features

(i) Two Groups (Group A and Group B)

(ii) Each group contains one 8-bit port and one 4-bit control/data port

(iii) The 8-bit data port can be either input or output. Both inputs and outputs are latched.

(iv) The 4-bit port is used for control and status of the 8-bit port.

Input control signal definitions (mode 1):

• **STB** (Strobe input) – If this lines falls to low, the data available at 8-bit input port is loaded into input latches.

• **IBF** (Input buffer full) – If this signal rises to logic 1, it indicates that data has been loaded into latches, i.e. it works as an acknowledgement.

• **INTR** (Interrupt request) – This active high output signal can be used to interrupt the CPU whenever an input device requests the service.

Output control signal definitions (mode 1):

• **OBF** (Output buffer full) – This status signal, whenever falls to low, indicates that CPU has written data to the specified output port.

• **ACK** (Acknowledge input) – ACK signal acts as an acknowledgement to be given by an output device. ACK signal, whenever low, informs the CPU that the data transferred by the CPU to the output device through the port is received by the output device.

• **INTR** (Interrupt request) – Thus an output signal that can be used to interrupt the CPU when an output device acknowledges the data received from the CPU.

Mode 2 Basic Functional Definitions:

(i) Used in Group A only

(ii) One 8-bit, bi-directional bus Port (Port A) and a 5-bit control Port (Port C)

(iii) Both inputs and outputs are latched

(iv)The 5-bit control port (Port C) is used for control and status for the 8-bit, bi-directional bus port (Port A)

Control signal definitions in mode 2:

INTR – (Interrupt request) As in mode 1, this control signal is active high and is used to interrupt the microprocessor to ask for transfer of the next data byte to/from it

Control Signals for Output operations:

OBF (Output buffer full) – This signal, when falls to low level, indicates that the CPU has written data to port A.

ACK (Acknowledge) This control input, when falls to logic low level, Acknowledges that the previous data byte is received by the destination and next byte may be sent by the processor.

INTE1 (A flag associated with OBF) This can be controlled by bit set/reset mode

Control signals for input operations:

STB (Strobe input)a low on this line is used to strobe in the data into the input Latches of 8255.

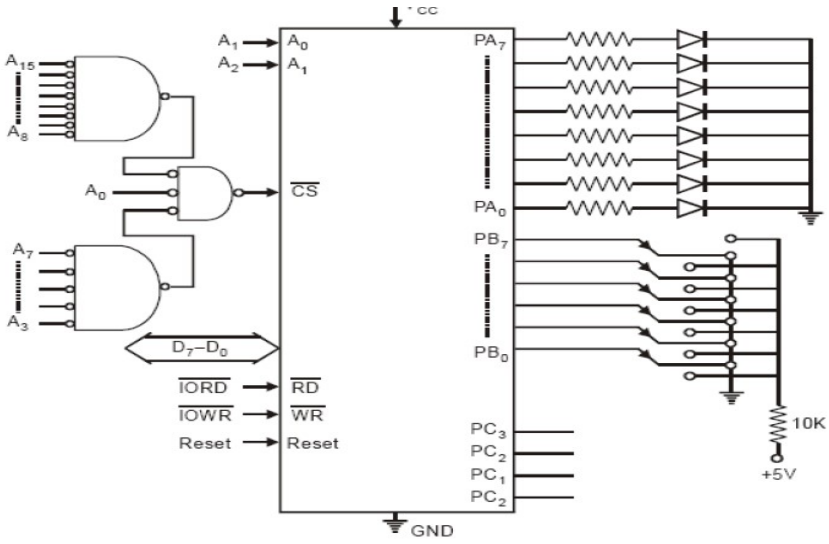
IBF (Input buffer full) when the data is loaded into input buffer, this signal rises to logic 1".

Problems

Example 1

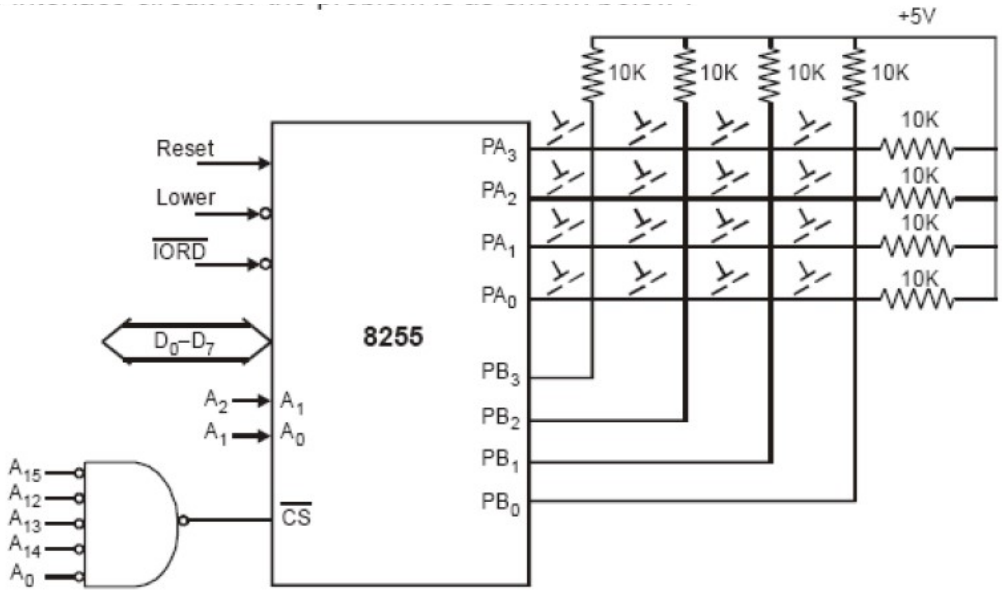
I/O Interfacing (LED's Interfaced with 8086) Example 1:- Interface an 8255 chip with 8086 to work as an I/O port. Initialize port A as output port, Port B as I/P port and Port C as O/P port. Port A address should be 0740H. Write an ALP to sense switch positions SW0–SW7 connected at port B. The sensed pattern is to be displayed on

port A, to which 8 LED's are connected, while port C lower displays number of on switches out of the total eight switches ?



Example 2

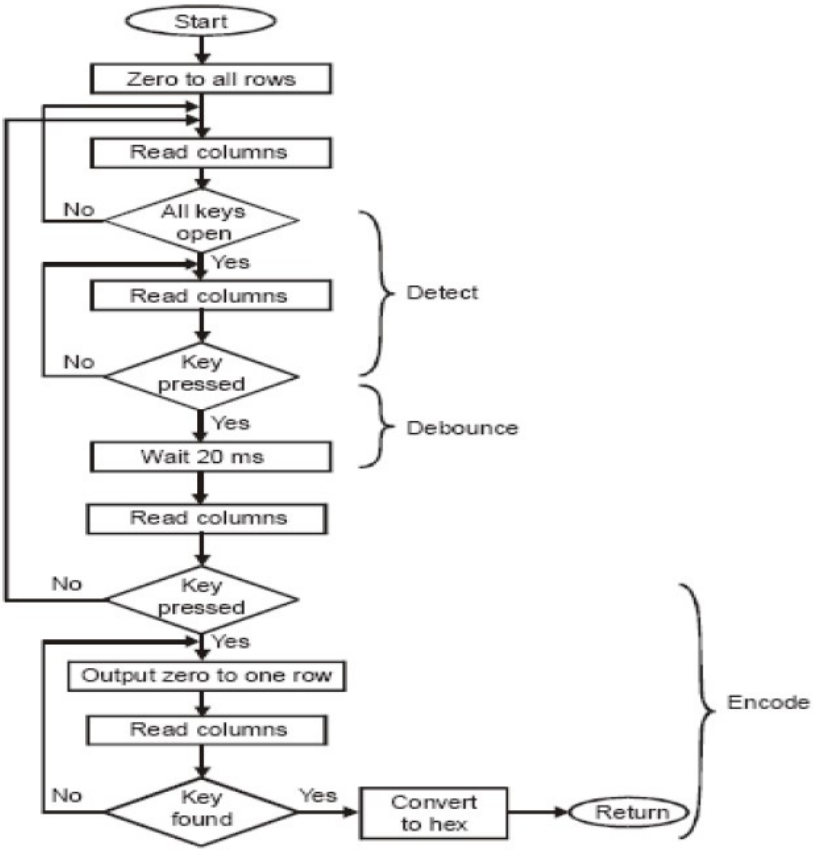
Interface a 4* 4 keyboard with 8086 using 8255, and write an ALP for detecting key closure and return the key code in AL. The debouncing period for a key in 20 ms?



Here we use port A as output port for selecting a row of keys while port B is used as an input port for sensing a closed key.

•Hence the keyboard lines are selected one by one through Port A and the Port B lines are polled continuously till a key closure is sensed.

•The higher order lines of Port A and Port B are left unused. The flowchart of the ALP is as shown below:



We suppose that we use simple mechanical switches. For keyboard, then to get the meaningful data from a keyboard requires three steps:

(1) Detect a key press

(2) Debounce the key press

(3) Encode the key press

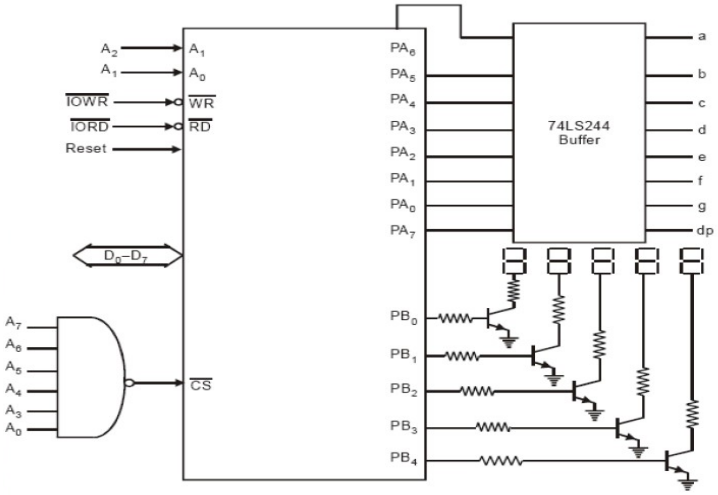
The three tasks can be done with hardware, software or a combination of the two.

The rows of the matrix are connected to four output port lines. The column line of the matrix are connected to four input port lines

Example 3

Interfacing 7-Seg Display with 8086

Interface an 8255 with 8086 at 80H as an I/O address of Port A.Interface five7segment displays with the 8255.Write an ALP to display1,2,3,4 and5over the 5 displays continuously as per their positions starting with1at the least significant position?



8279 – Programmable Keyboard/Display Interface

8255 can be used in interfacing keyboard and displays. The disadvantage of this method is that the processor has to refresh the display and check the status of the keyboard periodically.

Intel's 8279 is a general purpose Keyboard Display controller that simultaneously drives the display of a system and interfaces a Keyboard with the CPU.

The Keyboard Display interface scans the Keyboard to identify if any key has been pressed and sends the code of the pressed key to the CPU. It also transmits the data received from the CPU, to the display device.

Both of these functions are performed by the controller in repetitive fashion without involving the CPU. The Keyboard is interfaced either in the **interrupt or the polled mode**.

In the interrupt mode, the processor is requested service only if any key is pressed, otherwise the CPU can proceed with its main task.

In the polled mode, the CPU periodically reads an internal flag of 8279 to check for a key pressure.

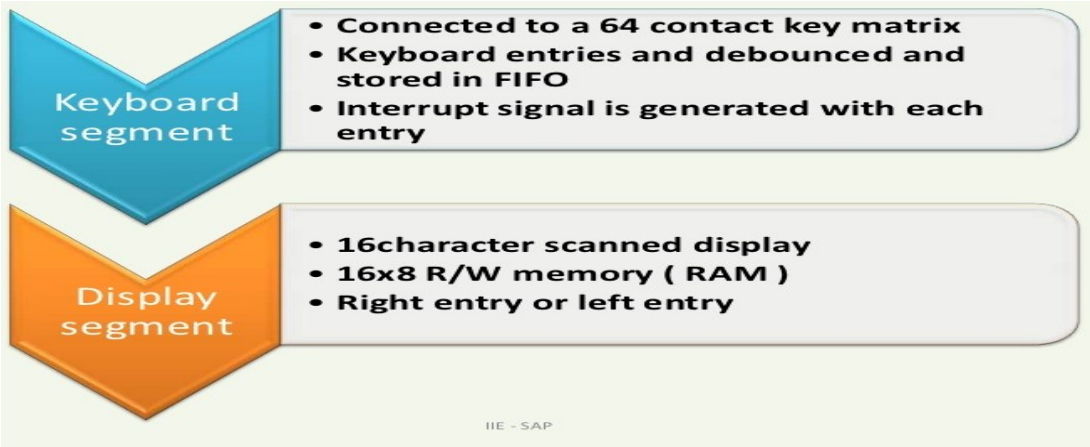
Two sections of 8279

The **Keyboard section** can interface an array of a maximum of 64 keys with the CPU. The Keyboard entries (key codes) are debounced and stored in an 8-byte FIFO RAM that is further accessed by the CPU to read the key codes. If more than eight characters are entered in the FIFO (i.e. more that eight keys are pressed), before

any FIFO read operation, the overrun status is set. If a FIFO contains a valid key entry, the CPU is interrupted (in interrupt mode) or the CPU checks the status (in polling) to read the entry.

Once the CPU reads a key entry, the FIFO is updated, i.e. the key entry is pushed out of the FIFO to generate space for new entries.

The 8279 normally provides a maximum of sixteen 7-seg **display interface** with CPU It contains a 16-byte display RAM that can be used either as an integrated block of 16x8-bits or two 16x4-bit block of RAM. The data entry to RAM block is controlled by CPU using the command words of the 8279.



Architecture and Signal Descriptions of 8279

I/O Control and Data Buffer

The I/O control section controls the flow of data to/from the 8279. The data buffer interface the external bus of the system with internal bus of 8279.

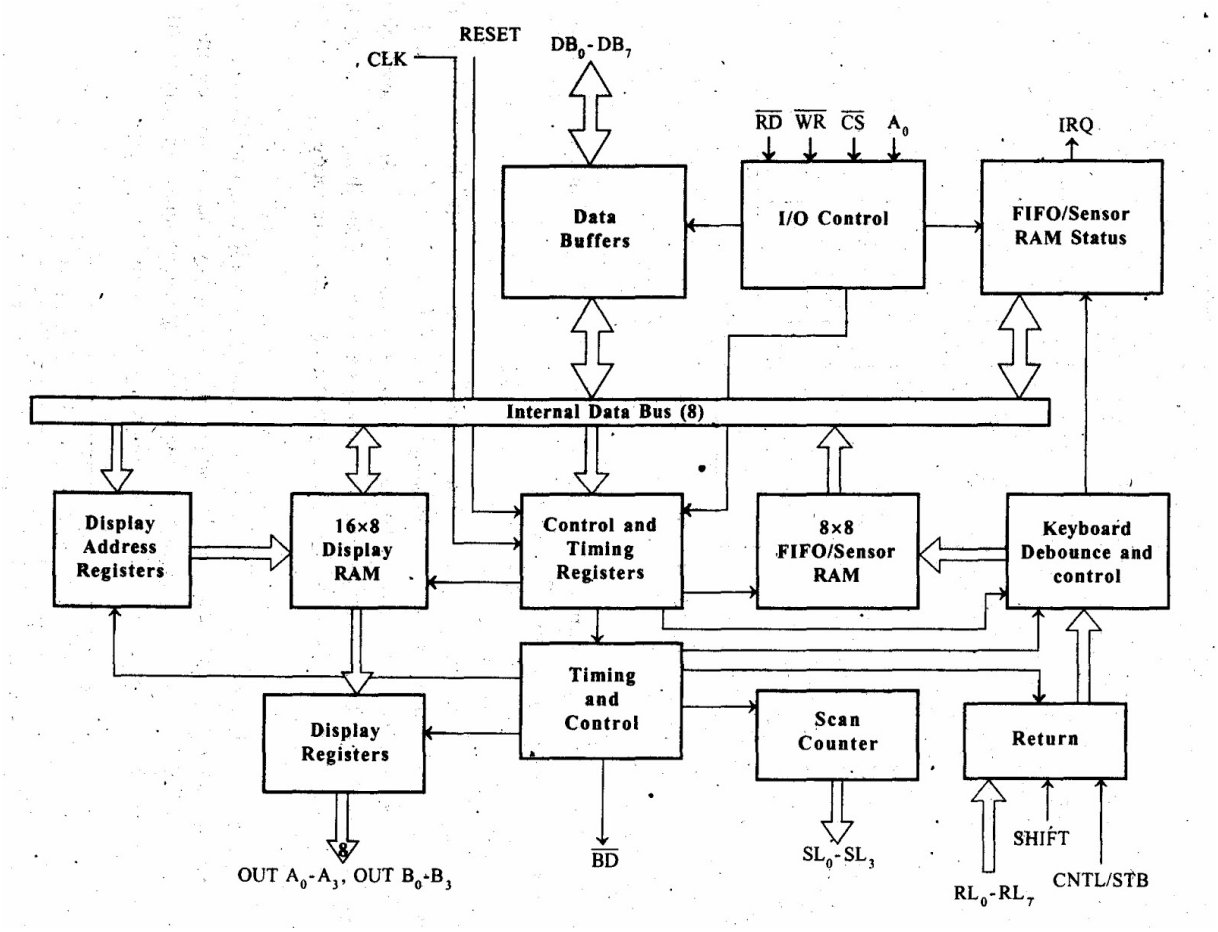
Control and Timing Register and Timing Control

These registers store the keyboard and display modes and other operating conditions programmed by CPU. The registers are written with Ao=1 and WR =0. The timing and control unit controls the basic timings for the operation of the circuit.

Scan Counter

The Scan Counter has two modes to scan the key matrix and refresh the display. In the Encoded mode, the counter provides a binary count that is to be externally decoded to provide the scan lines for keyboard and display

In the decoded scan mode, the counter internally decodes the least significant 2 bits and provides a decoded 1 out of 4 scan on SL0-SL3 (four internally decoded scan lines may drive up to 4 Displays). The Keyboard and Display both are in the same mode at a time.



Return Buffers and Keyboard Debounce and Control

This section scans for a Key closure row-wise. If it is detected, the Keyboard debounce unit debounces the key entry (i.e. wait for 10 ms). After the debounce period, if the key continues to be detected. The code of the Key is directly transferred to the sensor RAM along with SHIFT and CONTROL key status.

FIFO/Sensor RAM and Status Logic

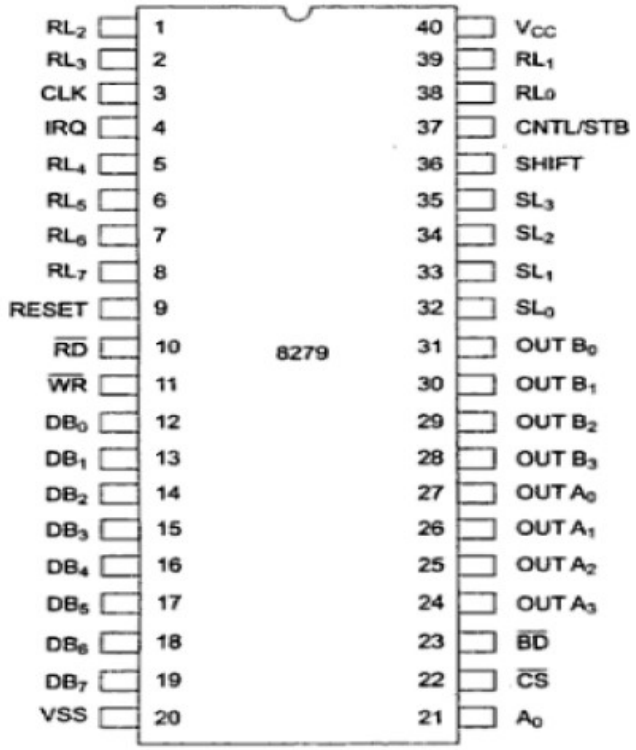
In Keyboard or strobed input mode, this block acts as 8-byte first-in-first-out (FIFO) RAM. Each key code of the pressed key is entered in the order of the entry, and in the meantime, read by the CPU, till the RAM becomes empty. The status logic generates an interrupt request after each FIFO read operation till the FIFO is empty.

In scanned sensor matrix mode, this unit acts as sensor RAM. Each row of the sensor RAM is loaded with the status of the corresponding row of sensors in the matrix. If a sensor changes its state, the IRQ line goes high to interrupt the CPU.

Display Address Registers and Display RAM.

The Display address registers hold the addresses of the word currently being written or read by the CPU to or from the display RAM. The contents of the registers are automatically updated by 8279 to accept the next data entry by CPU. The 16-byte display RAM contains the 16-byte of data to be displayed on the sixteen 7-seg displays in the encoded scan mode.

Pin diagram of 8279



Ao :

A high on the Ao line indicates the transfer of a command or status information. A low on this line indicates the transfer of data. This is used to select one of the internal registers of 8279.

IRQ:

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

SL0-SL3 – Scan Lines:

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

RL0-RL7 – Return Lines :

These are the input lines which are connected to one terminal of keys, while the other terminal of the keys are connected to the decoded scan lines. These are normally high,but pulled low when a key is pressed.

SHIFT :

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

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

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

BD – Blank Display :

This output pin is used to blank the display during digit switching or by a blanking command.

OUTA0 – OUTA3 and OUTB0 – OUTB3 :

These are the output ports for two 16x4 (or one 16 x 8) internal display refresh registers.The data from these lines is synchronized with the scan lines to scan the display and keyboard. The two 4-bit ports may also be used as one 8-bit port.

The Modes of operation of 8279

i. Input (Keyboard) modes

ii. Output (Display) modes

Keyboard Modes

1.Scanned Keyboard Mode with 2 Key Lockout

2 key lockout- if two keys are pressed with in a debounce cycle (simultaneously), no key is recognized till one of them remain closed and the other is released. The last key remains pressed is considered as single key press

In this mode of operation, when a key is pressed, a debounce logic comes into operation. The Key code of the identified key is entered into the FIFO with SHIFT and CNTL status, provided the FIFO is not full.

2. Scanned Keyboard with N-key Rollover

In this mode, each key depression is treated independently. When a key is pressed, the debounce circuit waits for 2 keyboard scans and then checks whether the key is still depressed. If it is still depressed, the code is entered in FIFO RAM. Any **number of keys can be pressed simultaneously and recognized in the**

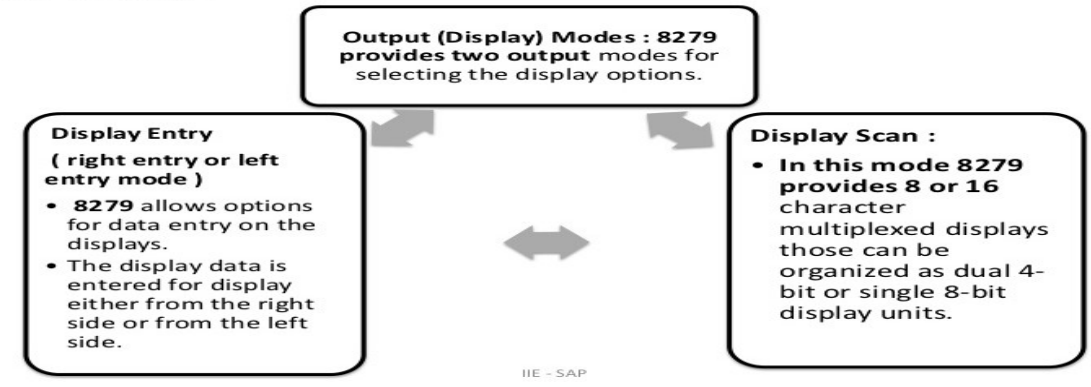
order, the Keyboard scan record them.

3. Scanned Keyboard Special Error Mode

This mode is valid only under the N-Key rollover mode. This mode is programmed using end interrupt/error mode set command. If during a single debounce period (two Keyboard scan) two keys are found pressed, this is considered a simultaneous depression and an error flag is set. This flag, if set, prevents further writing in FIFO but allows generation of further interrupts to the CPU for FIFO read.

4. Sensor Matrix Mode

In the Sensor Matrix mode, the debounce logic is inhibited the 8-byte memory matrix. The status of the sensor switch matrix is fed directly to sensor RAM matrix Thus the sensor RAM bits contains the row-wise and column-wise status of the sensors in the sensor matrix.



There are various options of data display The first one is known as left entry mode or type writer mode. Since in a type writer the first character typed appears at the left-most position, while the subsequent characters appears successively to the right of the first one.The other display format is known as right entry mode, or calculator mode, since the calculator the first character entered appears at the right-most position and this character is shifted one position left when the next character is entered.

1. Left Entry Mode

In the Left entry mode, the data is entered from the left side of the display unit. Address0 of the display RAM contains the leftmost display character and address 15 of the RAM contains the rightmost display character.

2. Right Entry Mode

In the right entry mode, the first entry to be displayed is entered on the rightmost display.The next entry is also placed in the right most display but after the previous display is shifted left by one display position.

Command Words of 8279

All the Command words or status words are written or read with Ao = 1 and CS = 0 to or from 8279.

a. Keyboard Display mode set

The format of the command word to select different modes of operation of 8279 is given below with its bit definitions.

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	A_0
0	0	0	D	D	K	K	K	1

D	*D*	*Display modes*
0	0	Eight 8-bit character Left entry
0	1	Sixteen 8-bit character Left entry (Default after reset)
1	0	Eight 8-bit character Right entry
1	1	Sixteen 8-bit character Right entry

K	*K*	*K*	*Keyboard modes*
0	0	0	Encoded scan, 2 key lockout (Default after reset)
0	0	1	Decoded scan, 2 key lockout
0	1	0	Encoded scan N-Key roll over
0	1	1	Decoded scan N-Key roll over
1	0	0	Encoded scan sensor matrix
1	0	1	Decoded scan sensor matrix
1	1	0	Strobed Input Encoded scan
1	1	1	Strobed Input Decoded scan

b. Programmable Clock

The clock for operation of 8279 is obtained by dividing the external clock input signal by a programmable constant called prescaler.

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	A_0
0	0	1	P	P	P	P	P	1

PPPPP is a 5-bit binary constant. The input frequency is divided by a decimal constant ranging from 2 to 31, decided by the bits of an internal prescalar, PPPPP.

c. Read FIFO/Sensor RAM

The format of this command is given as shown below

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	A_0
0	1	0	AI	X	A	A	A	1

X - don’t care

AI - Auto increment flag

AAA - Address pointer to 8 bit FIFO RAM

This word is written to set up 8279 for reading FIFO/Sensor RAM. In scanned keyboard mode, AI and AAA bits are of no use. The 8279 will automatically drive data bus for each subsequent read, in the same sequence, in which the data was entered.

d. Read Display RAM

This command enables a programmer to read the display RAM data

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	A_0
0	1	1	AI	A	A	A	A	1

The CPU writes this command word to 8279 to prepare it for display RAM read operation. AI is auto incremented flag and AAAA, the 4-bit address, points to the 16-byte display RAM that is to be read. If AI = 1, the address will be automatically, incremented after each read or write to the display RAM.

e. Write Display RAM

The format of this command is given as shown below

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	A_0
1	0	0	AI	A	A	A	A	1

AI - Auto increment flag

AAAA - 4-bit address for 16-bit display RAM to be written

Other details of this command are similar to the 'Read Display RAM Command.

f. Display Write Inhibit/Blanking

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	A_0
1	0	1	X	IW	IW	BL	BL	1
Output nibbles		→		A	B	A	B	

The IW (Inhibit write flag) bits are used to mask the individual nibble Here Do and D2 corresponds to OUTBo – OUTB3 while D1 and D3 corresponds to OUTAo-OUTA3 for blanking and masking respectively.

g. Clear Display RAM

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	A_0
1	1	0	CD_2	CD_1	CD_0	CF	CA	1

The CD2, CD1, CDo is a selectable blanking code to clear all the rows of the display RAM as given below. The characters A and B represents the output nibbles.

CD	CD1	CDo	
1	0	x	All Zeros (x don't care) AB = 00
1	1	0	A3-Ao = 2(0010) and B3-Bo = 00(0000)
1	1	1	All ones (AB = FF), i.e. clear RAM

Here, CA represents clear All and CF represents Clear FIFO RAM

End Interrupt/Error Mode Set

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	A_0
1	1	1	E	x	x	x	x	1

x—do not care

For the sensor matrix mode, this command lowers the IRQ line and enables further writing into the RAM. Otherwise, if a charge in sensor value is detected, IRQ goes high that inhibits writing in the sensor RAM.

Interfacing and Programming 8279

Problem :

Interface keyboard and display controller 8279 with 8086 at address 0080H. Write an ALP to set up 8279 in scanned keyboard mode with encoded scan, N-Key rollover mode.Use a 16 character display in right entry display format. Then clear the display RAM with zeros. Read the FIFO for key closure. If any key is closed, store it's code to register CL.Then write the byte 55 to all the displays, and return to DOS. The clock input to 8279 is 2MHz, operate it at 100KHz.

Solution :

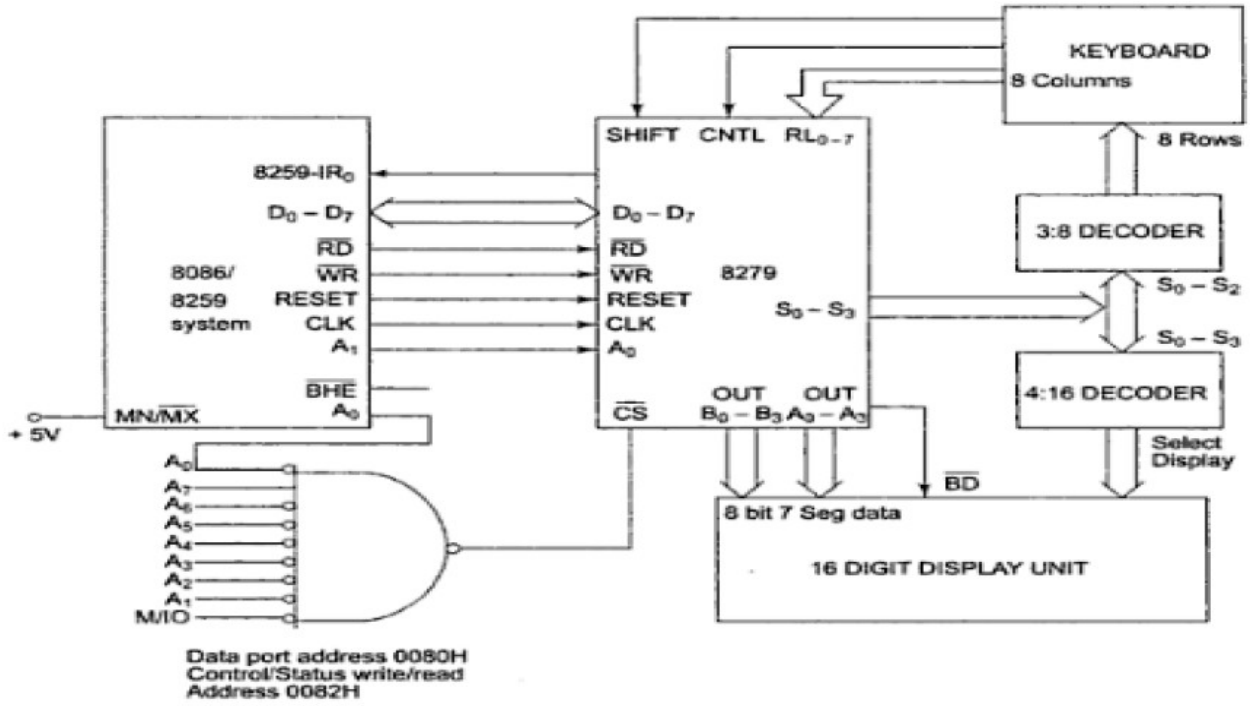
The 8279 is interfaced with lower byte of the data bus, i.e. Do-D7 . Hence the Ao input of 8279 is connected with address lineA1.

The data register of 8279 is to be addressed as 0080H, i.e.Ao=0.

For addressing the command or status word Ao input of 8279 should be 1.

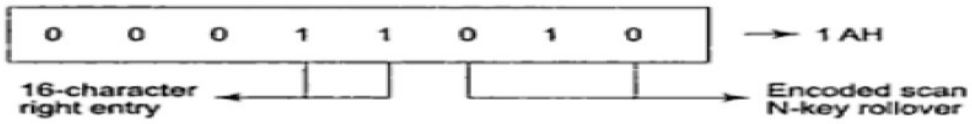
The next step is to write all the required command words for this problem.

Figure shows the interfacing schematic



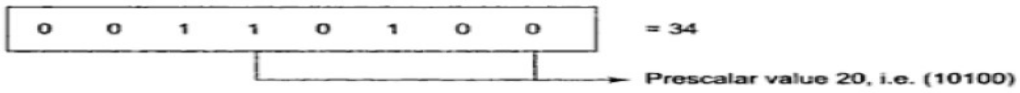
Keyboard/Display Mode Set CW :

This command byte sets the 8279 in 16-character right entry and encoded scan N-Key rollover mode.



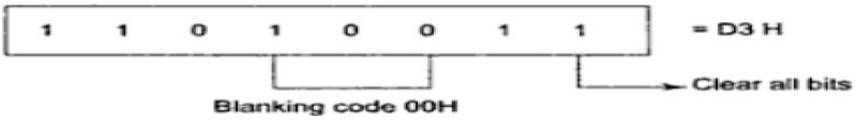
Program clock selection:

The clock input to 8279 is 2MHz, but the operating frequency is to be 100KHz, i.e. the clock input is to be divided by 20 (10100). Thus the prescalar value is 10100 and trhe command byte is set as given.



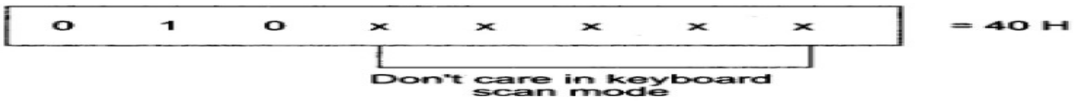
Clear Display RAM :

This command clears the display RAM with the programmable blanking code.



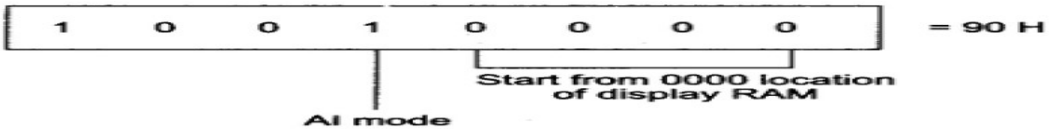
Read FIFO :

This command byte enables the programmer to read a key code from the FIFO RAM



Write Display RAM :

This command enables the programmer to write the addressed display locations of the RAM as presented below.



Program gives the ALP required to initialize the 8279 as required:

```
        Assume CS          : Code
        Code  Segment
Start : MOV AL, 1AH        ;     Set 8279 in Encoded scan,
        OUT 82H, AL        ;     N Key rollover, 16 display, Right entry mode.
        MOV AL, 34H        ;     Set clock prescalar to
        OUT 82H, AL        ;     100KHz
        MOV AL, 0D3H       ;     Clear display ram
        OUT 82H, AL        ;     command
        MOV AL, 40H        ;     Read FIFO command
        OUT 82H, AL        ;     for checking display RAM
Wait :  IN AL, 82H         ;     Wait for clearing of
        AND AL, 80H        ;     Display RAM by reading
        CMP AH, 80H        ;     FIFO Du bit of the status word i.e.
        JNZ Wait           ;     If Du bit is not set wait, else proceed.
        MOV AH, 40H        ;     Read FIFO command
        OUT 82H, AL        ;     for check key closure
        IN AL, 82H         ;     Read FIFO status
        AND AH, 07H        ;     Mask all bits except the
        CMP AH, 00         ;     number of characters bits
        JNZ Key code       ;     if any key is pressed, take
Warm: MOV AL, 90H          ;     required action, otherwise
        OUT 82H, AL        ;     Proceed to write display
        MOV AL, 55H        ;     RAM by using write display
        MOV CL, 10H        ;     command. Write the byte
Next:   OUT 80H, AL        ;     55H to all display RAM
        DEC CL             ;     Locations
        JNZ Next           ;
        JMP Stop           ;
Key code: Call Read code   ;     Call routine to read the key
        JMP Warm    ;    Code of the pressed key is
        assumed available
Stop    MOV AH, 4CH        ;     stop
        INT 21H
```