LED DISPLAY

The light emitting diodes (LEDs) are used to display limited data, results, events, or messages. The total number of characters that are displayed is fixed.

LEDs are commonly used on the front panel of instruments, digital clocks, etc. In simple terms, the light emitting diode is a diode which when conducts, emits light energy. The conduction starts when the anode is held at a higher voltage than the cathode.

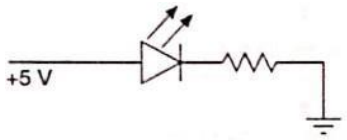

Fig : Light Emitting Diode

Microprocessor interface to LED

LEDs can be directly interfaced with the microprocessor through an output port.

- ✓ For common cathode interface whenever a bit is 1, the corresponding LED will glow.
- ✓ For common anode interface whenever a bit is 0, the corresponding LED will glow.

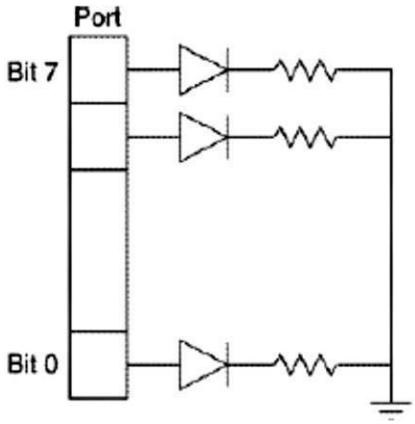

Fig : Microprocessor interface to LED (common cathode)

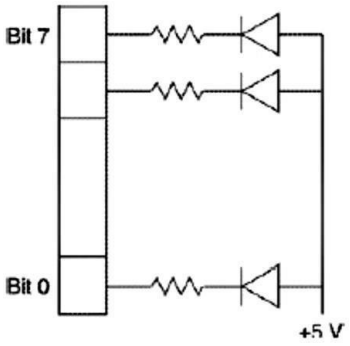

Fig : Microprocessor interface to LED (common anode)

Seven-Segment LED

The LEDs can be arranged in the following fashion

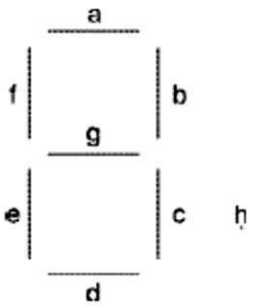

Fig : Seven-segment LED

This structure has eight segments marked a, b, c, d, e, f, g, and h. It is useful in displaying of the numeric and alphanumeric data. For example, to display character A, segments a, b, c, e, f and g should glow and to display character 6, segments a, c, d, e, f and g should glow.

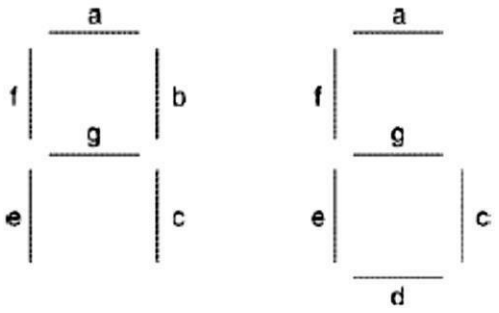

Fig : Character formation in seven-segment LEDs

Originally, the structure had only seven segments a to g, and the eighth segment h (to display the decimal point) was added subsequently. Therefore, even though it has eight segments, the name seven-segment LED has been retained.

The On/Off information for eight segments can be arranged in one byte. The bit information can be connected to respective segments in the manner shown in figure.

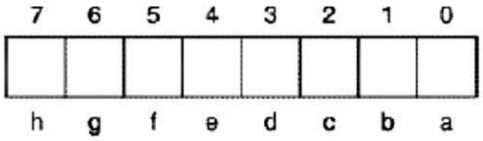

Fig : Control byte for seven-segment LEDs.

Anode 0 → glow the segment ; 1 → do not glow

Cathode 1 → glow the segment ; 0 → do not glow

Parallel interface

The parallel interface the anodes of all LEDs are held permanently at +5 V, whereas the cathodes are connected to the port bits. The microprocessor will load any bit pattern in the out port. Those segments stored in the bit pattern of the code, will be lighted.

Drawbacks:

- ✓ Used for a single character display.
- ✓ If the display continuously changes very fast, the eye will not be able to read anything.

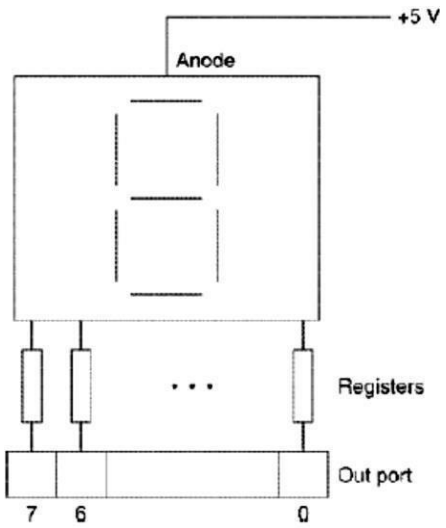

Fig : Microprocessor interface to seven-segment Led (parallel interface)

Serial interface

The serial interface overcomes the disadvantages of the complex hardware in case of a large number of seven-segment displays.

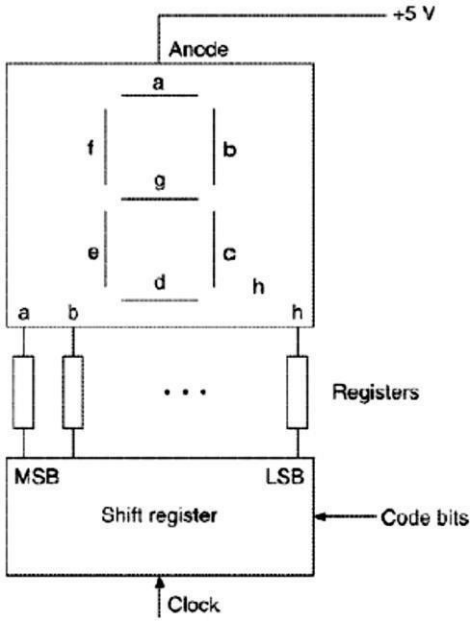

Fig : Microprocessor interface to seven segment LED (serial interface)

Cascading of seven-segment LEDs in a serial microprocessor interface

At the first step, the code of „a" will be loaded in the shift register. When the clock is loaded, the bit is passed to the LSB of the shift register.

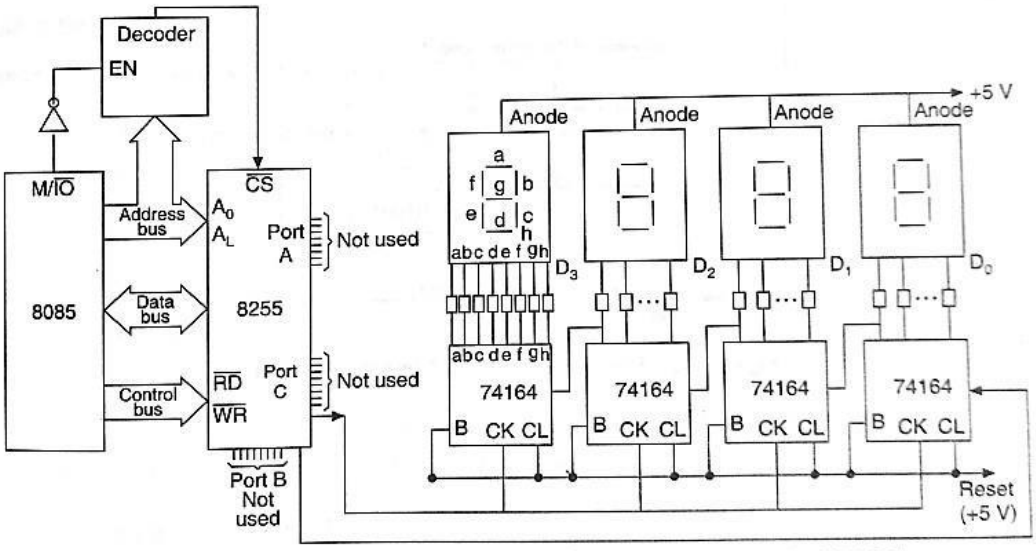

Fig : Seven-segment LED display interface to the 8086 using the 8255 PPI.

When the second bit of the code is loaded in the shift register with the clock the earlier bits shifts one step left and the later bit is included as the LSB of the shift register. The shifting occurs each time when a bit is loaded. When the seventh bit sis loaded the LSB of the display code reaches the MSB of the shift register and the character A appears.

Program:

Initialization: Port A – segment data inputs to display

Port B – Selecting a display position

```
MOV AL, 80
OUT CNTRLREG, AL        } Initialize 8255

        MOV BL, 08      ; load count
        MOV CL, 7F      ; load select pattern
        MOV SI, 1200    ; initialize memory pointer

LOOP:   MOV AL, CL      ; select digit
        OUT PB, AL
        MOV AL, SI      ; get the data

        OUT PA, AL

        CALL DELAY

        MOV AL, CL      ; Display the data

        ROR AL, 01      ; adjust select pattern
        INC SI          ; increment pointer
        DEC BL          ; decrement count
        JNZ LOOP        ; repeat 8 times

        HLT

DELAY : MOV DX, 0012
LOOP1 : DEC DX
        JNZ LOOP1
        RET
```

LCD DISPLAY

LCD is used in lot of application when compared to LEDs due to the following reasons:

1. The declining prices of LCDs.
2. The ability to display numbers, characters, and graphics.
3. Incorporation of a refreshing controller into the LCD, thereby relieving the CPU of the task of refreshing the LCD.
4. Ease of programming for characters and graphics

LCD pin descriptions (HD44780)

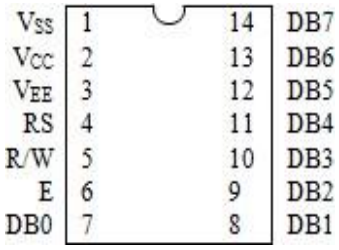

Fig: Pin Diagram

Vcc,Vss, and Vee

Vcc and Vss provide +5V and ground, respectively. VEE is used for controlling LCD contrast.

RS- Register Select

If RS = 0, the instruction command code register is selected, allowing the user to send a command such as clear display, cursor at home, etc.
If RS = 1 the data register is selected, allowing the user to send data to be displayed on LCD.

R/W - Read/Write

R/W input allows the user to write information to the LCD or read information from it. R/W = 1 when reading; R/W = 0 when writing.

E - Enable

The enable pin is used by the LCD to latch information presented to its data pins.

DB0-DB7

The 8-bit data pins, DB0- DB7, are used to send information to the LCD or read the contents of the LCD"s internal registers.

Liquid crystals have the properties of liquids as well as solid crystals. Liquid crystal displays do not emit or generate light, but alter externally generated illumination. Their ability to modulate light when electrical signal is applied has made them very useful in flat panel display technology.

The main **advantage** of LCDs over seven segment displays is

1. Ability to display numbers, characters and graphics. But seven segment displays can display only numbers and a limited set of characters.
2. LCDs have an internal controller and a refreshing circuit. So, the CPU is relieved of the task of refreshing the display.
3. The programming of LCD device is easier.
4. The cost of LCD is moderate.

There are two types of liquid crystal displays(LCD) according to the method of operation: 1. Dynamic Scattering 2. Field Effect **Construction:**

It consists of two glass plates with a liquid crystal fluid in between. The back plate is coated with thin transparent layer of conductive material whereas the front plate is made of photo etched conductive coating with seven segment pattern.

In the absence of electrical signal, the orientation order is maintained in the crystal allowing light to transmit. This makes the LCD display clear. The current through the liquid crystal causes the orientation order to collapse. The random orientation results scattering of light with light display segment on a dark back ground.

Many LCD modules have built in drivers for LCD and interfacing circuitry to interface them to the microprocessor/microcontroller systems. These LCD modules allow display of characters and numbers.

Interfacing LCD module with 8255

1. The eight data lines are used to interface the LCD module with the 8086 microprocessor using 8255
2. The three control signals are RS, R/$\bar{W}$ and E
   - ✓ RS is used to select a command or data register.
   - ✓ R/$\bar{W}$ indicates the direction of data flow between the display and the microprocessor.
   - ✓ The E signal is used to enable the data transfer.

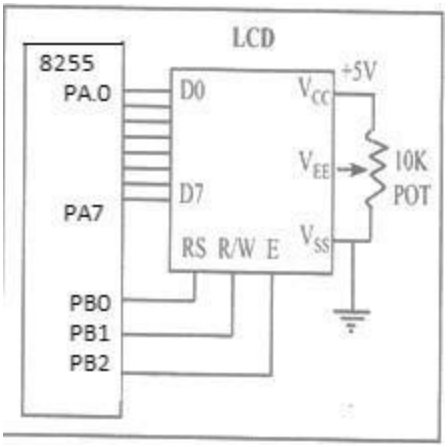

Fig: Interfacing LCD module with 8255

In the above figure the data lines are connected to the port A of 8255. The port C lines (PC0, PC1, PC2) are used to issue the control signals from the microprocessor to LCD.

- ✓ PC0 provide the enable (E) signal
- ✓ PC1 provides read/write (R/$\bar{W}$) control signal ✓ PC2 provides the register select signal.

The voltage at the VEE pin is adjusted by a potentiometer to adjust the contrast of LCD.

Program:

```
        MOV AL, 80
                                  } Initialize all ports as output port
        OUT CNTRLREG, AL          }

        MOV AL, 01
        CALL COMMAND
START:  MOV AL, 0E
        CALL COMMAND
        MOV SI, ADDR
        MOV CL, 0F
        MOV AL, SI
LOOP:   CALL DISP
        INC SI
        DEC CL
        JNZ LOOP

COMMAND SUBROUTINE

COMMAN  OUT PORT,AL

D:      MOV AL, 01
        OUT PORTC, AL

        MOV AL, 00
        OUT PORTC, AL
        CALL DELAY
        RET

DELAY SUBROUTINE
        MOV CL, COUNT1
; load control word to clear LCD display
; call the subroutine to issue command to LCD
; Initialize the cursor to home position
; switch on the display and the cursor
; initialize memory pointer
; initialize counter
; load the display data in accumulator
; call the subroutine to issue the data to LCD
```

; point to next data for display

; decrement counter

; if CL ≠ 0 loop again

; issue control word to data lines of LCD

RS = 0(command), R/$\bar{W}$ = 0 (write), E =1 (hig h pulse)

E =0 (low pulse) for a H to L transition

; wait for a predefined time delay

;return

; initialize counter1

```
LOOP2   MOV DL, COUNT2   ; initialize counter2
        DEC DL
LOOP1   JNZ LOOP1        } Decrement counter 2 till zero

        DEC CL
        JNZ LOOP2
        RET              ; return
```