

Name :Vansh Kolte

Batch: C4

Roll_no:62

Sub:MD

Practical_6

Aim: LCD interfacing

Code :

```
ORG 0000H      ; Set the origin of the program
MOV A,#38H     ; Load the command to initialize LCD
ACALL COMMAND  ; Call the command subroutine
MOV A,#0EH     ; Load the command for display ON, cursor ON
ACALL COMMAND  ; Call the command subroutine
MOV A,#01H     ; Load the command to clear display
ACALL COMMAND  ; Call the command subroutine
MOV A,#06H     ; Load the command to increment cursor
ACALL COMMAND  ; Call the command subroutine
MOV A,#86H     ; Set the cursor to a specific position
ACALL COMMAND  ; Call the command subroutine
```

; Display "VANSH" on the LCD

```
MOV A,#'V'     ; Load 'V' into the accumulator
ACALL DATA_DISPLAY ; Display 'V'
MOV A,#'A'     ; Load 'A' into the accumulator
ACALL DATA_DISPLAY ; Display 'A'
MOV A,#'N'     ; Load 'N' into the accumulator
ACALL DATA_DISPLAY ; Display 'N'
MOV A,#'S'     ; Load 'S' into the accumulator
ACALL DATA_DISPLAY ; Display 'S'
MOV A,#'H'     ; Load 'H' into the accumulator
ACALL DATA_DISPLAY ; Display 'H'
```

HERE: SJMP HERE ; Infinite loop to stop the program from exiting

; Subroutine for sending commands to the LCD

COMMAND:

```
ACALL READY    ; Call the ready subroutine to ensure LCD is ready
MOV P1,A       ; Move the contents of the accumulator to Port 1 (data)
CLR P2.0       ; RS = 0, for command
CLR P2.1       ; R/W = 0, for write
SETB P2.2      ; Enable high
```

```
CLR P2.2    ; Enable low, to send command
RET         ; Return from subroutine
```

; Subroutine for displaying data on the LCD

DATA_DISPLAY:

```
ACALL READY ; Call the ready subroutine to ensure LCD is ready
MOV P1,A    ; Move the contents of the accumulator to Port 1 (data)
SETB P2.0   ; RS = 1, for data
CLR P2.1    ; R/W = 0, for write
SETB P2.2   ; Enable high
CLR P2.2    ; Enable low, to send data
RET         ; Return from subroutine
```

; Subroutine to ensure the LCD is ready before sending the next command/data

READY:

```
SETB P1.7   ; Set the busy flag (assumes P1.7 is used for busy flag)
CLR P2.0    ; RS = 0, for command
SETB P2.1   ; R/W = 1, for read
```

BACK:

```
SETB P2.2   ; Enable high
CLR P2.2    ; Enable low, to latch the data
JB P1.7,BACK ; If P1.7 (busy flag) is set, loop back and wait
RET         ; Return when LCD is ready
```

END ; End of **program**

OUTPUT :

