```
;===================================================================
; Main.asm file generated by New Project wizard
;
; Created:   [Current Date]
; Processor: AT89C52
; Compiler:  ASEM-51 (Proteus)
;===================================================================

$NOMOD51
$INCLUDE (80C52.MCU)

;===================================================================
; DEFINITIONS
;===================================================================

TOP equ P2.7      ; Input A
BOTTOM  equ P2.6      ; Input B
OUTPUT  equ P1.7      ; Output C

RS equ P1.0    ; Command/Data
RW equ P1.1    ; Read/Write
EN equ P1.2    ; Latch enable


;===================================================================
; RESET and INTERRUPT VECTORS
;===================================================================

      ; Reset Vector
      org   0000h
      jmp   Start

;===================================================================
; CODE SEGMENT
;===================================================================

Start:  CLR TOP         ; Initially turn off P2.7
    CLR BOTTOM      ; Initially turn off P2.6
    CLR OUTPUT      ; Initially turn off Motor
    mov A, #38h ;set up 2 line, 5x7 matrix display
    Acall cmd
    mov A, #0Fh ;Display ON, Cursor ON, Cursor blinking
    Acall cmd
    mov A, #01h ;Clear the display
    Acall cmd
    mov A, #06h ;Cursor increment mode(left to right)
    Acall cmd
    mov A, #80h ;Cursor Home(line1, position1)
    Acall cmd



MAIN_LOOP:
      JB TOP, STOP           ; If A (P1.0) is 1, jump to STOP
      JNB BOTTOM, RUN           ; If B (P1.1) is 0, jump to RUN

      SJMP MAIN_LOOP       ; Continue looping

```

```
STOP:
      CLR OUTPUT               ; Turn off Motor
      Acall Full
      SJMP MAIN_LOOP       ; Return to main loop

RUN:
      SETB OUTPUT        ; Turn on Motor
      Acall Loww
      SJMP MAIN_LOOP       ; Return to main loop

Loww:   mov A, #'T'
    Acall disp
    mov A, #'A'
    Acall disp
    mov A, #'N'
    Acall disp
    mov A, #'K'
    Acall disp
    mov A, #' '
    Acall disp
    mov A, #'L'
    Acall disp
    mov A, #'O'
    Acall disp
    mov A, #'W'
    Acall disp



On: mov A, #01h ;Clear the display
    Acall cmd
    mov A, #80h ;Cursor Home(line1, position1)
    Acall cmd
        mov A, #'M'
    Acall disp
    mov A, #'O'
    Acall disp
    mov A, #'T'
    Acall disp
    mov A, #'O'
    Acall disp
    mov A, #'R'
    Acall disp
    mov A, #' '
    Acall disp
    mov A, #'O'
    Acall disp
    mov A, #'N'
    Acall disp

loop:   jnb BOTTOM, loop
loop3:  jnb TOP, loop3
    ret

Full:   mov A, #01h ;Clear the display
    Acall cmd
    mov A, #80h ;Cursor Home(line1, position1)
    Acall cmd
```

```
    mov A, #'T'
    Acall disp
    mov A, #'A'
    Acall disp
    mov A, #'N'
    Acall disp
    mov A, #'K'
    Acall disp
    mov A, #' '
    Acall disp
    mov A, #'F'
    Acall disp
    mov A, #'U'
    Acall disp
    mov A, #'L'
    Acall disp
    mov A, #'L'
    Acall disp

loop1:  jb TOP, loop1

Off:    mov A, #01h ;Clear the display
    Acall cmd
    mov A, #80h ;Cursor Home(line1, position1)
    Acall cmd
        mov A, #'M'
    Acall disp
    mov A, #'O'
    Acall disp
    mov A, #'T'
    Acall disp
    mov A, #'O'
    Acall disp
    mov A, #'R'
    Acall disp
    mov A, #' '
    Acall disp
    mov A, #'O'
    Acall disp
    mov A, #'F'
    Acall disp
    mov A, #'F'
    Acall disp

loop2:  jb BOTTOM, loop2
    mov A, #01h ;Clear the display
    Acall cmd
    mov A, #80h ;Cursor Home(line1, position1)
    Acall cmd
    ret

cmd:    mov P3,A
    clr RS ; command
    clr RW ; write command
    setb EN ; set latch enable
    clr EN ; clear latch enable
    Acall delay
    ret
```

```
175.
176. disp:   mov P3,A
177.     setb RS ;data
178.     clr RW ;write command
179.     setb EN ;set latch enable
180.     clr EN ;clear latch enable
181.     Acall delay
182.     ret
183.
184. delay:
185.     mov R1, #0FFH
186. delay1:
187.     mov R2, #0FFh
188. delay2:
189.     djnz R2, delay2
190.     djnz R1, delay1
191.     ret
192.
193. ;=====================================================================
194.       END
195.
196.
197.
```