# Display Text on LCD 16*2

## Write Assembly Code for LCD 16x2 in 8bit Mode to display your name with call SRAM and hold your name on the screen 5sec approximately then clear the screen to display your department & Roll No. on 1st line, 2nd line respectively from SRAM. Hold these also 5sec approximately then repeat again your name for looping. The Circuit is same as above to simulate and make the hardware.

**Code for LCD 16x2 in 8bit Mode to display your name with call SRAM**

```
// LCD text display with SRAM calling
.INCLUDE "M32DEF.INC"
.ORG 0x0000
LDI R16,HIGH(RAMEND)
OUT SPH,R16
LDI R16,LOW(RAMEND)
OUT SPL,R16



  //Data Loading to SRAM
    LDI R26,0x80
                    LDI R27,0x00
                    LDI R16,'S'
                    ST X+,R16
                    LDI R16,'H'
                    ST X+,R16
                    LDI R16,'A'
                    ST X+,R16
                    LDI R16,'L'
                    ST X+,R16
                    LDI R16,'I'
                    ST X+,R16
                    LDI R16,'N'
                    ST X+,R16
                    LDI R16,'I'
                    ST X+,R16
                    LDI R16,' '
                    ST X+,R16
                    LDI R16,0
                    ST X+,R16
```

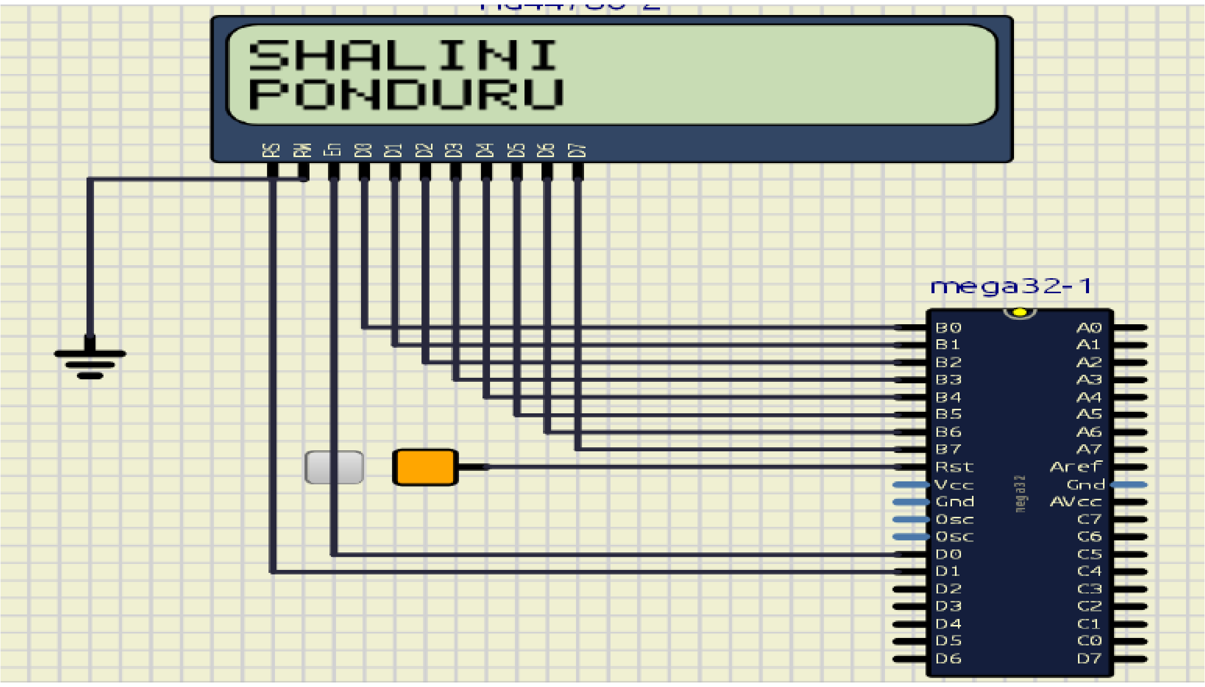
```
                        LDI R16,'P'
                        ST X+,R16
                        LDI R16,'O'
                        ST X+,R16
                        LDI R16,'N'
                        ST X+,R16
                        LDI R16,'D'
                        ST X+,R16
                        LDI R16,'U'
                        ST X+,R16
                        LDI R16,'R'
                        ST X+,R16
                        LDI R16,'U'
                        ST X+,R16
                        LDI R16,0
                        ST X,R16
                        // Data Direction Register Settings
LDI R16,0xFF
OUT DDRB,R16
SBI DDRD,PIND0 //Falling Edged Enable
SBI DDRD,PIND1 //Register Select

CALL SUB_R

  LDI R26,0x80
                        LDI R27,0x00
                        LDI R16,'E'
                        ST X+,R16
                        LDI R16,'L'
                        ST X+,R16
                        LDI R16,'E'
                        ST X+,R16
                        LDI R16,'C'
                        ST X+,R16
                        LDI R16,'T'
                        ST X+,R16
                        LDI R16,'R'
                        ST X+,R16
                        LDI R16,'I'
                        ST X+,R16
                        LDI R16,'C'
                        ST X+,R16
                        LDI R16,'A'
                        ST X+,R16
                        LDI R16,'L'
                        ST X+,R16
                        LDI R16,0
```

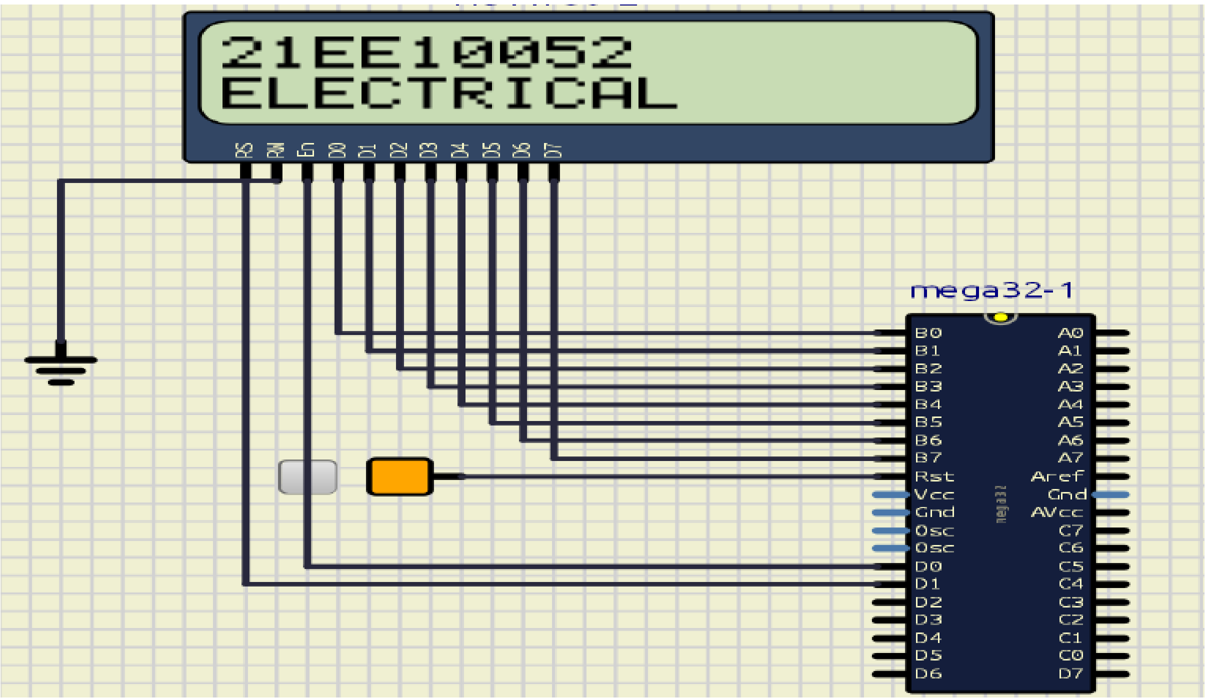
```
                    ST X+,R16
                    LDI R16,'2'
                    ST X+,R16
                    LDI R16,'1'
                    ST X+,R16
                    LDI R16,'E'
                    ST X+,R16
                    LDI R16,'E'
                    ST X+,R16
                    LDI R16,'1'
                    ST X+,R16
                    LDI R16,'0'
                    ST X+,R16
                    LDI R16,'0'
                    ST X+,R16
                    LDI R16,'5'
                    ST X+,R16
                    LDI R16,'2'
                    ST X+,R16
                    LDI R16,0
                    ST X,R16
                    // Data Direction Register Settings
LDI R16,0xFF
OUT DDRB,R16
SBI DDRD,PIND0 //Falling Edged Enable
SBI DDRD,PIND1 //Register Select
CALL SUB_R

SUB_R: // LCD Initialization
CBI PORTD,PIND1 // Command Register Enable
LDI R16,0x38 //2 lines and 5x7 matrix
OUT PORTB,R16
CALL ENABLE
LDI R16,0x02 // Return Home
OUT PORTB,R16
CALL ENABLE
LDI R16,0x01 // Clear display screen
OUT PORTB,R16
CALL ENABLE
LDI R16,0x0C //Display on, cursor off
OUT PORTB,R16
CALL ENABLE
LDI R16,0x06 // Shift Cursor to right automatically after print on LCD
OUT PORTB,R16
CALL ENABLE
//Set Cursor Coordinate
LDI R16,0x80 //Set Cursor at begining of 1st Line
```

```
OUT PORTB,R16
CALL ENABLE
//Data Read and print on LCD
LDI R26,0x80
LCD_PRINT1: SBI PORTD,PIND1 // Data Register Enable
  LD R16,X+
  OUT PORTB,R16
  CALL ENABLE
  LD R16,X
  CPI R16,0
  BRNE LCD_PRINT1
//2nd Line set cursor coordinate
CBI PORTD,PIND1
LDI R16,0xC0 //Set Cursor at begining of 2nd Line
OUT PORTB,R16
CALL ENABLE

//Data Read and print on LCD
INC R26 // To Skip previous null character
LCD_PRINT2: SBI PORTD,PIND1 // Data Register Enable
  LD R16,X+
  OUT PORTB,R16
  CALL ENABLE
  LD R16,X
  CPI R16,0
  BRNE LCD_PRINT2

Delay:
 LDI R16,0xFF //Loop 1
L1: LDI R17,0xFF // Loop 2
L2: LDI R18,0x07 // Loop 3
L3: NOP
 DEC R18
 BRNE L3 //Loop 3 End
 DEC R17
 BRNE L2 //Loop 2 End
 DEC R16
 BRNE L1 // Loop 1 End
RET
ENABLE : SBI PORTD,PIND0
  LDI R18,0x50
  LOOP2: LDI R17,0xFF
   LOOP1: NOP
    DEC R17
    BRNE LOOP1
   DEC R18
   BRNE LOOP2
```

```
CBI PORTD,PIND0
RET
```

## Simulide Simulation





## Hardware Implementation