/\*------------- 16x2 LCD Interfacing ------------------------

\* Display the messages on 16x2 LCD

\* Interface Details

\* LCD Data (D0 to D7) - PORTD (RD0 to RD7)

\* LCD RS - RE0

\* LCD RW - RE1

\* LCD EN - RE2

-------------------------------------------------------------\*/

#include <xc.h>

#pragma config FOSC = HS // Oscillator Selection bits (HS oscillator: High-speed crystal/resonator on RA6/OSC2/CLKOUT and RA7/OSC1/CLKIN)

#pragma config WDTE = OFF // Watchdog Timer Enable bit (WDT disabled and can be enabled by SWDTEN bit of the WDTCON register)

//Declarations

#define LCD\_DATA PORTD //LCD data port to PORTD

#define ctrl PORTE //LCD control port to PORTE

#define rs PORTEbits.RE0 //register select signal to RE0

#define rw PORTEbits.RE1 //read/write signal to RE1

#define en PORTEbits.RE2 //enable signal to RE2

//Function Prototypes

void init\_LCD(void); //Function to initialize the LCD

void LCD\_command(unsigned char cmd); //Function to pass command to the LCD

void LCD\_data(unsigned char data); //Function to write character to the LCD

void LCD\_write\_string(unsigned char \*str);//Function to write string to the LCD

void msdelay (unsigned int time); //Function to generate delay

unsigned char var1[] = "Welcome";//Declare message to be displayed

unsigned char var2[] = "to sspu";

void main (void)

{

ANSEL = 0x00;

ANSELH = 0x00;

TRISD = 0x00; //Configuring PORTD as output

TRISE = 0x00; //Configuring PORTE as output

init\_LCD(); // call function to initialise of LCD

msdelay(50); // delay of 50 mili seconds

LCD\_write\_string(var1);//Display message on first line

msdelay(15);

LCD\_command(0xC0); // initiate cursor to second line

LCD\_write\_string(var2);//Display message on second line

while (1); //Loop here

} //End of Main

//Function Definitions

void msdelay (unsigned int time) //Function to generate delay

{

unsigned int i, j;

for (i = 0; i < time; i++)

for (j = 0; j < 315; j++);//Calibrated for a 1 ms delay in MPLAB

}

void init\_LCD(void) // Function to initialise the LCD

{

LCD\_command(0x38); // initialization of 16X2 LCD in 8bit mode

msdelay(15);

LCD\_command(0x01); // clear LCD

msdelay(15);

LCD\_command(0x0C); // cursor off

msdelay(15);

LCD\_command(0x80); // go to first line and 0th position

msdelay(15);

}

void LCD\_command(unsigned char cmd) //Function to pass command to the LCD

{

LCD\_DATA = cmd; //Send data on LCD data bus

rs = 0; //RS = 0 since command to LCD

rw = 0; //RW = 0 since writing to LCD

en = 1; //Generate High to low pulse on EN

msdelay(15);

en = 0;

}

void LCD\_data(unsigned char data)//Function to write data to the LCD

{

LCD\_DATA = data; //Send data on LCD data bus

rs = 1; //RS = 1 since data to LCD

rw = 0; //RW = 0 since writing to LCD

en = 1; //Generate High to low pulse on EN

msdelay(15);

en = 0;

}

//Function to write string to LCD

void LCD\_write\_string( unsigned char \*str)

{

int i = 0;

while (str[i] != 0)

{

LCD\_data(str[i]); // sending data on LCD byte by byte

msdelay(15);

i++;

}

}