#include <p18f4520.h> //Include Controller specific .h

//Configuration bit settings

#pragma config OSC = HS //Oscillator Selection

#pragma config WDT = OFF //Disable Watchdog timer

#pragma config LVP = OFF //Disable Low Voltage Programming

#pragma config PBADEN = OFF //Disable PORTB Analog inputs

//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 LCD

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

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

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

//Start of Main Program

void main(void)

{

char var1[] = " Wel-Come";//Declare message to be displayed

char var2[] = "Smart Logic Tech";

ADCON1 = 0x0F; //Configuring the PORTE pins as digital I/O

TRISD = 0x00; //Configuring PORTD as output

TRISE = 0x00; //Configuring PORTE as output

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

msdelay(50); // delay of 50 milliseconds

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 < 710; j++);//Calibrated for a 1 ms delay in MPLAB

}

void init\_LCD(void) // Function to initialize 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 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(static char \*str)

{

int i = 0;

while (str[i] != '\0') //Check for end of the string

{

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

msdelay(15);

i++;

}

}