#include <htc.h>

#define Input RC7

#define Input1 RC6

#define \_XTAL\_FREQ 20000000

#define RS RD2

#define EN RD3

#define D4 RD4

#define D5 RD5

#define D6 RD6

#define D7 RD7

#include <xc.h>

#pragma config FOSC = HS // Oscillator Selection bits (HS oscillator)

#pragma config WDTE = OFF // Watchdog Timer Enable bit (WDT disabled)

#pragma config PWRTE = ON // Power-up Timer Enable bit (PWRT enabled)

#pragma config BOREN = ON // Brown-out Reset Enable bit (BOR enabled)

#pragma config LVP = OFF // Low-Voltage (Single-Supply) In-Circuit Serial Programming Enable bit (RB3 is digital I/O, HV on MCLR must be used for programming)

#pragma config CPD = OFF // Data EEPROM Memory Code Protection bit (Data EEPROM code protection off)

#pragma config WRT = OFF // Flash Program Memory Write Enable bits (Write protection off; all program memory may be written to by EECON control)

#pragma config CP = OFF // Flash Program Memory Code Protection bit (Code protection off)

//LCD Functions Developed by Circuit Digest.

void Lcd\_SetBit(char data\_bit) //Based on the Hex value Set the Bits of the Data Lines

{

if(data\_bit& 1)

D4 = 1;

else

D4 = 0;

if(data\_bit& 2)

D5 = 1;

else

D5 = 0;

if(data\_bit& 4)

D6 = 1;

else

D6 = 0;

if(data\_bit& 8)

D7 = 1;

else

D7 = 0;

}

void Lcd\_Cmd(char a)

{

RS = 0;

Lcd\_SetBit(a); //Incoming Hex value

EN = 1;

\_\_delay\_ms(4);

EN = 0;

}

Lcd\_Clear()

{

Lcd\_Cmd(0); //Clear the LCD

Lcd\_Cmd(1); //Move the curser to first position

}

void Lcd\_Set\_Cursor(char a, char b)

{

char temp,z,y;

if(a== 1)

{

temp = 0x80 + b - 1; //80H is used to move the curser

z = temp>>4; //Lower 8-bits

y = temp & 0x0F; //Upper 8-bits

Lcd\_Cmd(z); //Set Row

Lcd\_Cmd(y); //Set Column

}

else if(a== 2)

{

temp = 0xC0 + b - 1;

z = temp>>4; //Lower 8-bits

y = temp & 0x0F; //Upper 8-bits

Lcd\_Cmd(z); //Set Row

Lcd\_Cmd(y); //Set Column

}

}

void Lcd\_Start()

{

Lcd\_SetBit(0x00);

for(int i=1065244; i<=0; i--) NOP();

Lcd\_Cmd(0x03);

\_\_delay\_ms(5);

Lcd\_Cmd(0x03);

\_\_delay\_ms(11);

Lcd\_Cmd(0x03);

Lcd\_Cmd(0x02); //02H is used for Return home -> Clears the RAM and initializes the LCD

Lcd\_Cmd(0x02); //02H is used for Return home -> Clears the RAM and initializes the LCD

Lcd\_Cmd(0x08); //Select Row 1

Lcd\_Cmd(0x00); //Clear Row 1 Display

Lcd\_Cmd(0x0C); //Select Row 2

Lcd\_Cmd(0x00); //Clear Row 2 Display

Lcd\_Cmd(0x06);

}

void Lcd\_Print\_Char(char data) //Send 8-bits through 4-bit mode

{

char Lower\_Nibble,Upper\_Nibble;