

```
#include <stdio.h>
```

```
#include <string.h>
```

```
#include <delays.h>
```

```
#include <stdlib.h>
```

```
#include <usart.h>
```

```
#include <adc.h>
```

```
#include <p18f8722.h>
```

```
#include <xlcd.h>
```

```
#pragma config OSC = HS
```

```
#pragma config IESO = OFF
```

```
#pragma config PWRT = ON
```

```
#pragma config WDT = OFF
```

```
#pragma config MCLRE = ON
```

```
#pragma config LVP = OFF
```

```
#define BAUD_RATE_GEN 16 // =FOSC/((64 * baudrate)-1), set for 9600bps
```

```
/*                      Definitions for referencing various components of the system  
*/
```

```
#define KeyState PORTDbits.RD4
```

```
#define KeyNotPressed PORTDbits.RD5
```

```
#define GreenButton PORTCbits.RC0
```

```
#define RedButton PORTCbits.RC1
```

```
#define SyringelsHome PORTCbits.RC2
```

```
#define Solenoid PORTCbits.LATC3
```

```
/*                                     Definitions to make life easier
                                     */
```

```
#define FALSE 0
```

```
#define TRUE 1
```

```
#define NO 0
```

```
#define YES 1
```

```
#define OPEN 1
```

```
#define CLOSED 0
```

```
/* Some variables that are used in this code */
```

```
long int count=0;
```

```
char Input[100]="\0";
```

```
char ScreenPrint[20];
```

```
unsigned int TimeCount;
```

```
int MaxPress;
```

```
int MaxVol;
```

```
int Step;
```

```
int StepCount;
```

```
int CurrentVol;
```

```
int PressureI;
```

```
float PressureF;
```

```
int Keypad;
```

```
int Lastkey;  
  
char Keyvar;  
  
int AbortRequest;
```

```
/* DDRAM Addresses for visible LCD lines */
```

```
int Line1 = 128;  
  
int Line2 = 168;  
  
int Line3 = 148;  
  
int Line4 = 212;
```

```
/* Character Code Definitions for Keys */
```

```
#define Key0      0x04  
  
#define Key1      0x0A  
  
#define Key2      0x0B  
  
#define Key3      0x0C  
  
#define Key4      0x03  
  
#define Key5      0x06  
  
#define Key6      0x09  
  
#define Key7      0x02  
  
#define Key8      0x05  
  
#define Key9      0x08  
  
#define FuncKey   //not needed in this version  
  
#define MenuKey   0x07
```

```
#define EnterKey      0x0F
#define ClearKey      0x0D
#define UpKey         //not needed in this version
#define DownKey       //not needed in this version
#define NullKey       0x00
```

```
/* Prototypes for C18 LCD library */
```

```
void OpenXLCD( unsigned char lcdtype );
```

```
void DelayFor18TCY( void );
```

```
void DelayPORXLCD (void);
```

```
void DelayXLCD (void);
```

```
void putsXLCD( char *buffer );
```

```
/* Prototypes for my code functions */
```

```
void DisplayMainMenu(void);
```

```
void RunPressureTest(void);
```

```
void GetKey(void);
```

```
void DisplayKey(void);
```

```
/* 18 cycle (approx) delay for LCD routines */
```

```
void DelayFor18TCY( void )
```

```
{
```

```
    Nop();
```

```
    Nop();
```

```

        Nop();

        Nop();

        Nop();

        Nop();

        Nop();

        Nop();

        Nop();

        Nop();

        Nop();

        Nop();

        Nop();

        Nop();

        Nop();

        Nop();

    }

/* 15 msec (approx) delay for LCD routines */
void DelayPORXLCD (void)
{
    Delay1KTCYx(43); // Delay of 15ms

    // Cycles = (TimeDelay * Fosc) / 4

    // Cycles = (15ms * 10MHz) / 4

    // Cycles = 37,500

    return;
}

```

```
/* 5 msec (approx) delay for LCD routines */
```

```
void DelayXLCD (void)
```

```
{
```

```
Delay1KTCYx(17); // Delay of 5ms
```

```
// Cycles = (TimeDelay * Fosc) / 4
```

```
// Cycles = (5ms * 10MHz) / 4
```

```
// Cycles = 12,500
```

```
return;
```

```
}
```

```
/*                                     MAIN MENU DISPLAY ROUTINE
```

```
//
```

```
void DisplayMainMenu()
```

```
{
```

```
    while(BusyXLCD());
```

```
    WriteCmdXLCD(DON&CURSOR_OFF&BLINK_OFF); //Don't want cursor blinking during menu  
display
```

```
    while(BusyXLCD());
```

```
    WriteCmdXLCD(0x01);      // Clear display
```

```
    while(BusyXLCD());
```

```
    SetDDRamAddr( Line1 );
```

```
    while(BusyXLCD());
```

```
    putrsXLCD("1 - Pressure Test");
```

```
    SetDDRamAddr( Line2 );
```

```
    while(BusyXLCD());
```

```

    putsXLCD("2 - Volume Test");

    SetDDRamAddr( Line3 );

    while(BusyXLCD());

    putsXLCD("3 - Test Settings");

    SetDDRamAddr( Line4 );

    while(BusyXLCD());

    putsXLCD("4 - Settings");

    while(BusyXLCD());

}

```

```

/*                                KEYPRESS ROUTINE
                                */

void GetKey(void){

    while(1){

        while(KeyNotPressed);

        Keypad = PORTD;           //Save new keypress

        if(!KeyState)            //Only break out if this is a key press, and not release

            break;

    }

    Keypad=Keypad & 0x0F;        //Keypad data is on RD3:RD0

    Delay1KTCYx(35);            //Debounce delay of at least 10ms

}

```

```

/*                                DIGIT DISPLAY ROUTINE
                                */

void DisplayKey(void)
{
    if (count < 3){
        Input[count] = Keyvar;

        count++;

        Input[count] = '\0';    //Terminate end of string
        while(BusyXLCD());
        SetDDRamAddr( Line2 );    //Write current string to Line 2
        while(BusyXLCD());
        putsXLCD(Input);
    }
}

```

```

/*                                PRESSURE TEST ROUTINE
                                */

void RunPressureTest(void)
{
    Input[0] = '\0'; //          Reset counter, Input string, and terminate null
    count = 0;

    /* PROMPT USER FOR DESIRED PRESSURE, TURN CURSOR BLINK ON */

    while(BusyXLCD());

    WriteCmdXLCD(0x01);

```

```

while(BusyXLCD());

SetDDRamAddr( Line1 );

while(BusyXLCD());

putsXLCD("Target Pressure?");

while(BusyXLCD());

SetDDRamAddr( Line2 );

while(BusyXLCD());

putsXLCD(" ____ (mmHg)");

while(BusyXLCD());

WriteCmdXLCD(DON&CURSOR_ON&BLINK_ON);

while(BusyXLCD());

SetDDRamAddr( Line2 );

while(BusyXLCD());

/*
*/

AbortRequest = NO;

while(!AbortRequest)
{

    GetKey();           //Get keypress

    switch (Keypad)      // Select case according to keypress.
    {
        Append appropriate character to Input string. Advance Counter.
    }
}

```

```
{  
    case Key0:  
        Keyvar = '0';  
        DisplayKey();  
        break;  
    case Key1:  
        Keyvar = '1';  
        DisplayKey();  
        break;  
    case Key2:  
        Keyvar = '2';  
        DisplayKey();  
        break;  
    case Key3:  
        Keyvar = '3';  
        DisplayKey();  
        break;  
    case Key4:  
        Keyvar = '4';  
        DisplayKey();  
        break;  
    case Key5:  
        Keyvar = '5';  
        DisplayKey();  
        break;
```

```

case Key6:

Keyvar = '6';

DisplayKey();

break;

case Key7:

Keyvar = '7';

DisplayKey();

break;

case Key8:

Keyvar = '8';

DisplayKey();

break;

case Key9:

Keyvar = '9';

DisplayKey();

break;

case ClearKey:                                //Clear Display, reset character counter

    while(BusyXLCD());

    SetDDRamAddr( Line2 );

    while(BusyXLCD());

    putsXLCD(" ____");

    while(BusyXLCD());

    SetDDRamAddr( Line2 );

    while(BusyXLCD());

    Input[0] = '\0';

```

```

        count = 0;

        break;

case MenuKey:

    AbortRequest = YES;

    break;

```

```

case EnterKey: //enter pressed

```

```

        MaxPress = atoi(Input);

        itoa(MaxPress,Input); //gets rid of leading zeros

        while(BusyXLCD());

        WriteCmdXLCD(DON&CURSOR_OFF&BLINK_OFF);

        while(BusyXLCD());

        WriteCmdXLCD(0x01);      // Clear display

        while(BusyXLCD());

        SetDDRamAddr( Line1);

        while(BusyXLCD());

        putsXLCD("Test Pressure:"); //Input has desired pressure in

mmHg

        while(BusyXLCD());

        SetDDRamAddr( Line2);

        while(BusyXLCD());

        putsXLCD(Input);

        while(BusyXLCD());

        putsXLCD(" mmHg");

        while(BusyXLCD());

```

```
Delay10KTCYx(255);
```

```
if (MaxPress > 100 || MaxPress <=0)           // Check that  
pressure is in allowable range. If not, warn user.
```

```
{
```

```
    while(BusyXLCD());
```

```
    WriteCmdXLCD(0x01);           //Clear Display
```

```
    while(BusyXLCD());
```

```
    SetDDRamAddr( Line3);
```

```
    while(BusyXLCD());
```

```
    putsXLCD("***Pressure must be***");
```

```
    while(BusyXLCD());
```

```
    SetDDRamAddr( Line4);
```

```
    while(BusyXLCD());
```

```
    putsXLCD("*** 1 - 100 mmHg ***");
```

```
    while(BusyXLCD());
```

```
    SetDDRamAddr( Line1 );
```

```
    while(BusyXLCD());
```

```
    putsXLCD("Target Pressure?");
```

```
    while(BusyXLCD());
```

```
    SetDDRamAddr( Line2 );
```

```
    while(BusyXLCD());
```

```
    putsXLCD(" ____ (mmHg)");
```

```
    while(BusyXLCD());
```

```
    SetDDRamAddr( Line2 );
```

```
    WriteCmdXLCD(DON&CURSOR_ON&BLINK_ON);
```

```

        while(BusyXLCD());

        Input[0] = '\0';

        count = 0;

    }

    else

    {

        while(BusyXLCD());

        WriteCmdXLCD(0x01);

        // Clear Display

        while(BusyXLCD());

        SetDDRamAddr( Line1 );

        while(BusyXLCD());

        putsXLCD("Starting Test...");           //      Inform
user that test is beginning.

        while(BusyXLCD());

//while(1)

//{

//if (PORTD == 0x0F)

//{

//ConvertADC(); //perform ADC conversion

//while(BusyADC()); //wait for result

//NEEDS WORK, also

should go below pre-test delay

//Pressure1 = ReadADC(); //get ADC result

//Delay10KTCYx(100);

//SetDDRamAddr( Line1 );      while(BusyXLCD());      putsXLCD("%i",Pressure1);

```

```
//}
```

```
//}
```

```
/*
```

```
PRE-TEST TIME DELAY
```

```
*/
```

```
while(BusyXLCD());
```

```
WriteCmdXLCD(DON&CURSOR_OFF&BLINK_OFF);
```

```
for(TimeCount = 10; TimeCount > 0; TimeCount--){
```

```
//Update screen with pre-test time count in seconds
```

```
while(BusyXLCD());
```

```
SetDDRamAddr( Line2 );
```

```
while(BusyXLCD());
```

```
putsXLCD("Time: ");
```

```
while(BusyXLCD());
```

```
putsXLCD( itoa(TimeCount,ScreenPrint) );
```

```
while(BusyXLCD());
```

```
putsXLCD(" sec ");
```

```
Delay10KTCYx(250);
```

```
}
```

```
/*
```

```
PRESSURE TEST STARTS HERE
```

```
*/
```

```
Solenoid = CLOSED;
```

```
//put a data display here too
```

```

//Insert feedback loop test

/*
                                POST-TEST TEST TIME DELAY                                */

                                while(BusyXLCD());

                                WriteCmdXLCD(0x01);

// Clear Display

                                while(BusyXLCD());

                                SetDDRamAddr( Line1 );

                                while(BusyXLCD());

                                putsXLCD("Test Completing...");                                //

Inform user that test is ending.

                                while(BusyXLCD());

                                for(TimeCount = 10; TimeCount > 0; TimeCount--){ //
                                    while(BusyXLCD());

                                    SetDDRamAddr( Line2 );

                                    while(BusyXLCD());

                                    putsXLCD("Time: ");

                                    while(BusyXLCD());

                                    putsXLCD( itoa(TimeCount,ScreenPrint) );

                                    while(BusyXLCD());

                                    putsXLCD(" sec ");

                                    Delay10KTCYx(250);

                                }

```

```
DisplayMainMenu(); // *****Main Menu
```

```
}
```

```
pressure switch statement      default:           //This is the end of the target
```

```
break;
```

```
}
```

```
}
```

```
DisplayMainMenu();           //Lassie's going home
```

```
}
```

```
/*                           END PRESSURE TEST ROUTINE  
*/
```

```
/*                           VOLUME TEST ROUTINE  
*/
```

```
void RunPressureTest(void)
```

```
{
```

```
Input[0] = '\0'; //          Reset counter, Input string, and terminate null
count = 0;
```

```
/* PROMPT USER FOR DESIRED MAX VOLUME, TURN CURSOR BLINK ON */
```

```
while(BusyXLCD());

WriteCmdXLCD(0x01);

while(BusyXLCD());

SetDDRamAddr( Line1 );

while(BusyXLCD());

putsXLCD("Input Test Volume:");

while(BusyXLCD());

SetDDRamAddr( Line2 );

while(BusyXLCD());

putsXLCD("___ (mL)");

while(BusyXLCD());

WriteCmdXLCD(DON&CURSOR_ON&BLINK_ON);

while(BusyXLCD());

SetDDRamAddr( Line2 );

while(BusyXLCD());
```

```
/*
```

```
*/
```

```
AbortRequest = NO;
```

```
while(!AbortRequest)
```

```

{

    GetKey();                //Get keypress

    switch (Keypad)          // Select case according to keypress.
Append appropriate character to Input string. Advance Counter.
    {
        case Key0:
            Keyvar = '0';
            DisplayKey();
            break;
        case Key1:
            Keyvar = '1';
            DisplayKey();
            break;
        case Key2:
            Keyvar = '2';
            DisplayKey();
            break;
        case Key3:
            Keyvar = '3';
            DisplayKey();
            break;
        case Key4:
            Keyvar = '4';
            DisplayKey();

```

```
break;

case Key5:

    Keyvar = '5';

    DisplayKey();

    break;

case Key6:

    Keyvar = '6';

    DisplayKey();

    break;

case Key7:

    Keyvar = '7';

    DisplayKey();

    break;

case Key8:

    Keyvar = '8';

    DisplayKey();

    break;

case Key9:

    Keyvar = '9';

    DisplayKey();

    break;

case ClearKey:                                //Clear Display, reset character counter

    while(BusyXLCD());

    SetDDRamAddr( Line2 );

    while(BusyXLCD());
```

```

        putsXLCD("___");

        while(BusyXLCD());

        SetDDRamAddr( Line2 );

        while(BusyXLCD());

        Input[0] = '\0';

        count = 0;

        break;

case MenuKey:

        AbortRequest = YES;

        break;


case EnterKey: //enter pressed


        Maxvol = atoi(Input);

        itoa(MaxVol,Input); //gets rid of leading zeros

        while(BusyXLCD());

        WriteCmdXLCD(DON&CURSOR_OFF&BLINK_OFF);

        while(BusyXLCD());

        WriteCmdXLCD(0x01);          // Clear display

        while(BusyXLCD());

        SetDDRamAddr( Line1);

        while(BusyXLCD());

        putsXLCD("Input Test Volume:"); //Input has desired volume in

        while(BusyXLCD());

        SetDDRamAddr( Line2);

```

mL

```

while(BusyXLCD());

putsXLCD(Input);

while(BusyXLCD());

putsXLCD(" mmL");

while(BusyXLCD());

Delay10KTCYx(255);

```

```

if (Maxvol > 10 || MaxPress <=0)           // Check that
pressure is in allowable range. If not, warn user.

```

```

{
    while(BusyXLCD());

    WriteCmdXLCD(0x01);           //Clear Display

    while(BusyXLCD());

    SetDDRamAddr( Line3);

    while(BusyXLCD());

    putsXLCD("***Volume must be***");

    while(BusyXLCD());

    SetDDRamAddr( Line4);

    while(BusyXLCD());

    putsXLCD("*** 1 - 10 mL ***");

    while(BusyXLCD());

    SetDDRamAddr( Line1 );

    while(BusyXLCD());

    putsXLCD("Test Volume?");

    while(BusyXLCD());

    SetDDRamAddr( Line2 );
}

```

```

        while(BusyXLCD());

        putsXLCD("___ (mmL)");

        while(BusyXLCD());

        SetDDRamAddr( Line2 );

        WriteCmdXLCD(DON&CURSOR_ON&BLINK_ON);

        while(BusyXLCD());

        Input[0] = '\0';

        count = 0;

    }

    else

    {

        while(BusyXLCD());

        WriteCmdXLCD(0x01);

        // Clear Display

        while(BusyXLCD());

        SetDDRamAddr( Line1 );

        while(BusyXLCD());

        putsXLCD("Starting Test...");           //      Inform

        user that test is beginning.

        while(BusyXLCD());

    //while(1)

    //{

    //if (PORTD == 0x0F)

    //{

    //ConvertADC(); //perform ADC conversion

```

```
//while(BusyADC()); //wait for result
```

```
//NEEDS WORK, also
```

```
should go below pre-test delay
```

```
//Pressure1 = ReadADC(); //get ADC result
```

```
//Delay10KTCYx(100);
```

```
//SetDDRamAddr( Line1 );    while(BusyXLCD());    putsXLCD("%i",Pressure1);
```

```
//}
```

```
//}
```

```
/*
```

```
PRE-TEST TIME DELAY
```

```
*/
```

```
while(BusyXLCD());
```

```
WriteCmdXLCD(DON&CURSOR_OFF&BLINK_OFF);
```

```
for(TimeCount = 10; TimeCount > 0; TimeCount--){
```

```
//Update screen with pre-test time count in seconds
```

```
while(BusyXLCD());
```

```
SetDDRamAddr( Line2 );
```

```
while(BusyXLCD());
```

```
putsXLCD("Time: ");
```

```
while(BusyXLCD());
```

```
putsXLCD( itoa(TimeCount,ScreenPrint) );
```

```
while(BusyXLCD());
```

```
putsXLCD(" sec ");
```

```
Delay10KTCYx(250);
```

```
}
```

```

/*                                PRESSURE TEST STARTS HERE                                */

                                Solenoid = CLOSED;

                                //put a data display here too


                                //Insert feedback loop test


/*                                POST-TEST TEST TIME DELAY                                */

                                while(BusyXLCD());

                                WriteCmdXLCD(0x01);

// Clear Display

                                while(BusyXLCD());

                                SetDDRamAddr( Line1 );

                                while(BusyXLCD());

                                putsXLCD("Test Completing...");                                //

Inform user that test is ending.

                                while(BusyXLCD());

                                for(TimeCount = 10; TimeCount > 0; TimeCount--){ //

                                    while(BusyXLCD());

                                    SetDDRamAddr( Line2 );

                                    while(BusyXLCD());

                                    putsXLCD("Time: ");

                                    while(BusyXLCD());

                                    putsXLCD( itoa(TimeCount,ScreenPrint) );

```

}

```
DisplayMainMenu(); // *****Main Menu
```

}

```
default: //This is the end of the target
```

pressure switch statement

```
break;
```

}

}

```
DisplayMainMenu();           //Lassie's going home
```

}

$$/^{*}$$
 $\ast/$

END VOLUME TEST ROUTINE

```

/*                                SMALL ANIMAL BAROSTAT
*/
//

void main()
{
    TRISB=0b00000000; //This is not necessary anymore after implementation of the XLCD routines
    ADCON1=0b00001101; //We want channels AN0 and AN1 active
    TRISC=0b11111011;      //Only port C output is RC2 which is used for the solenoid


/*    UART can be written to with printf statements if this
    section of code is enabled */

    stdout = _H_USART; // printf -> UART


    Open1USART(USART_TX_INT_OFF & USART_RX_INT_OFF
    & USART_ASYNC_MODE & USART_EIGHT_BIT & USART_CONT_RX
    & USART_BRGH_LOW, BAUD_RATE_GEN); //Open UART1

```

```
/*      End UART stdout code */
```

```
/*      LCD Setup */
```

```
Delay10KTCYx(255); // Wait for LCD to start up, needs at least 200ms
```

```
OpenXLCD( FOUR_BIT & LINES_5X7 ); //Open LCD
```

```
while(BusyXLCD());
```

```
WriteCmdXLCD(0x01);      // Clear display
```

```
while(BusyXLCD());
```

```
WriteCmdXLCD(DON&CURSOR_OFF&BLINK_OFF);
```

```
while(BusyXLCD());
```

```
SetDDRamAddr( Line1 );
```

```
while(BusyXLCD());
```

```
putsXLCD("Starting Up...");
```

```
/* End LCD Setup */
```

```
/* Set up Stepper Motor drive system */
```

```
    //TO-DO
```

```
Solenoid = OPEN;
```

```
while(BusyXLCD());
```

```
SetDDRamAddr( Line1 );
```

```
while(BusyXLCD());
```

```

putsXLCD("Homing Syringe...");

printf("BD1\r\n");

Delay1KTCYx(15);          //Wait for characters to be transmitted


while(!SyringIsHome){

    printf("M-\r\n");

    Delay1KTCYx(15);          //Wait for characters to be transmitted

}

printf("H0\r\n");

/* End drive system setup */


/* Control loop begins here */


//      OpenADC(ADC_FOSC_2 & ADC_RIGHT_JUST & ADC_12_TAD, ADC_CH0 &
// ADC_INT_OFF & ADC_VREFPLUS_VDD & , 4); //open adc port for reading


DisplayMainMenu();    // *****Main Menu


while(1) //MAIN LOOP BEGINS HERE
{

    GetKey();          //Get a keypress

```

```

switch (Keypad)
{
    case Key1:
        RunPressureTest();    //Start Pressure Test
        break;

    case Key2:
        RunVolumeTest//Start Volume Test
        break;

    case Key3:
        //3 pressed
        break;

    case Key4:
        WriteCmdXLCD(0x01);    // Clear display
        while(BusyXLCD());
        SetDDRamAddr( Line1 );
        while(BusyXLCD());
        putrsXLCD("-----");
        while(BusyXLCD());
        SetDDRamAddr( Line2 );
        while(BusyXLCD());
        putrsXLCD("- ECE 480 -");
        while(BusyXLCD());
        SetDDRamAddr( Line3 );
        while(BusyXLCD());

```

```
    putsXLCD("- DESIGN TEAM 7 -");  
    while(BusyXLCD());  
    SetDDRamAddr( Line4 );  
    while(BusyXLCD());  
    putsXLCD("-----");  
    GetKey();           //Wait for any key
```

```
    default:  
        break;  
    //    End of keypress loop */
```

```
}
```

```
    DisplayMainMenu();           //Back to the main menu
```

```
}
```

```
/* End of control loop */
```

```
    Close1USART();
```

```
}
```