```
/*  ************************************************
    * FILENAME: lab2.c                              *
    * AUTHOR: Dan Kass kassd@msoe.edu               *
    * DATE: December 6, 2012                        *
    * CLASS: CE-2811 Lab 2 Professor Barnicki       *
    * Provdies:                                     *
    *       Test program for the lcd.c file         *
    *                                               *
    ************************************************ */

#include <avr/io.h>
#include <stdio.h>

#include "lcd.h"
#include <MSOE/delay.c>

int main(void)
{
    uint16_t delay = 2000;
    lcd_init();

    char test[] = "sample";
    char test2[] = "hey there";
    char custom[] = {0b00000000,
                     0b00000100,
                     0b00001010,
                     0b00011111,
                     0b00011111,
                     0b00001010,
                     0b00000100,
                     0b00000000};

    while(1){

        char *testPtr = test;
        char *test2Ptr = test2;
        char *customPtr = custom;

        //sends sample to the screen
        delay_ms(delay/4);
        lcd_string(testPtr);
        delay_ms(delay/4);
        //clears the screen
        lcd_ff();
        delay_ms(delay/2);
        //goes to position 0,5
        lcd_gotoxy(0, 5);
        //snets hey there to the sreen
        lcd_string(test2Ptr);
        delay_ms(delay/4);
        //goes back one char
        lcd_left();
        //sents 7 to the screen
```

```
        lcd_char('7');
        delay_ms(delay/4);
        //turns off the backlight
        lcd_backlight(OFF);
        delay_ms(delay/2);
        //turns on the the backlight
        lcd_backlight(ON);
        delay_ms(delay/4);
        //clears the screen
        lcd_clear();
        delay_ms(delay);
        //sends hey there to the screen
        lcd_string(test2Ptr);
        delay_ms(delay/2);
        //turns display off
        lcd_switch(OFF);
        delay_ms(delay/2);
        //turns display on
        lcd_switch(ON);
        delay_ms(delay/2);
        //clears the display
        lcd_clear();
        delay_ms(delay/2);
        //custom char
        lcd_define_char(3, customPtr);
        lcd_custom(3);
        delay_ms(delay);
        //clears the display
        lcd_clear();

    } //while
}//main
```

```
/*************************************************************
 * FILENAME: lcd.h                                           *
 * AUTHOR: Dan Kass kassd@msoe.edu                           *
 * DATE: December 6, 2012                                    *
 * CLASS: CE-2811 Lab 2 Professor Barnicki                   *
 * Provdies:                                                 *
 *       Funtion prototypes for the lcd                      *
 *       provides output to the lcd                          *
 *       and funtion control to the lcd screen               *
 *                                                           *
 *************************************************************/

#ifndef lcd_h
#define lcd_h

#define F_CPU   16.00E6
#define UART_BAUD_RATE  9.60E3
#define UBRR_VAL (F_CPU/(UART_BAUD_RATE*16)-1)

#define BS 0x08  //backspace (left)
#define FS 0x09  //forwardspace (right)
#define LF 0x0a  //line feed
#define FF 0x0c  //form feed
#define CR 0x0d  //carriage return

#define LIGHTON 0x11  //backlight on
#define LIGHTOFF 0x12  //backlight off

#define DISPLAYOFF 0x15 //display off
#define DISPLAYON 0x18  //default display on

#define ON '1'
#define OFF '0'

/*****************************************************************
*
*   Author: Dan Kass kassd@msoe.edu
*   Purpose: Initializes the serial communications
*   Parameters: void
*   Returns: void
*
*****************************************************************/
void lcd_init(void);

/*****************************************************************
*
*   Author: Dan Kass kassd@msoe.edu
*   Purpose: Prints a character out to the lcd
*   Parameters: char. The character that is to be printed
*   Returns: void
*
*****************************************************************/
void lcd_char(char);
```

```
/*****************************************************************
*
*   Author: Dan Kass kassd@msoe.edu
*   Purpose: Prints a string out to the lcd
*   Parameters: char*. The point to the character array for the
*       string
*   Returns: void
*
*****************************************************************/
void lcd_string(char*);

/*****************************************************************
*
*   Author: Dan Kass kassd@msoe.edu
*   Purpose: Puts the lcd curser in the correct x, y corrdinate spot
*   Parameters: uint8_t, uint8_t. row, column
*       (0,0) is the upper left, 0 <= x or y <= 15
*   Returns: void
*
*****************************************************************/
void lcd_gotoxy(uint8_t, uint8_t);

/*****************************************************************
*
*   Author: Dan Kass kassd@msoe.edu
*   Purpose: Clears the lcd and returns the curser to (0,0)
*   Parameters: void
*   Returns: void
*
*****************************************************************/
void lcd_clear(void);

/*****************************************************************
*
*   Author: Dan Kass kassd@msoe.edu
*   Purpose: Sends a form feed to the lcd
*   Parameters: void
*   Returns: void
*
*****************************************************************/
void lcd_ff(void);

/*****************************************************************
*
*   Author: Dan Kass kassd@msoe.edu
*   Purpose: Sends a backspace to the lcd
*   Parameters: void
*   Returns: void
*
*****************************************************************/
void lcd_left(void);
```

```
/*****************************************************************
*
*   Author: Dan Kass kassd@msoe.edu
*   Purpose: Turns the lcdbacklight on or off
*   Parameters: char. Either "ON" or "OFF" #define 1 and 0
*   Returns: void
*
*****************************************************************/
void lcd_backlight(char);

/*****************************************************************
*
*   Author: Dan Kass kassd@msoe.edu
*   Purpose: Puts the lcd curser in the correct x, y corrdinate spot
*   Parameters: char. Either "ON" or "OFF" #define 1 and 0
*   Returns: void
*
*****************************************************************/
void lcd_switch(char);

/*****************************************************************
*
*   Author: Dan Kass kassd@msoe.edu
*   Purpose: assigns character array to a custom character spot in
*       the lcd
*   Parameters: uint8_t, char*. 0 to 7 custom character
*       spot, char pointer to cust character
*   Returns: void
*
*****************************************************************/
void lcd_define_char(uint8_t, char*);

/*****************************************************************
*
*   Author: Dan Kass kassd@msoe.edu
*   Purpose: Prints the custom character out to the lcd
*   Parameters: uint8_t. 0 to 7 custom character spot
*   Returns: void
*
*****************************************************************/
void lcd_custom(uint8_t);

#endif
```

```
   /*****************************************************************
    * FILENAME: lcd.c                                               *
    * AUTHOR: Dan Kass kassd@msoe.edu                               *
    * DATE: December 6, 2012                                        *
    * CLASS: CE-2811 Lab 2 Professor Barnicki                       *
    * Provdies:                                                     *
    *       Funtion for the lcd                                     *
    *       provides output to the lcd                              *
    *       and funtion control to the lcd screen                   *
    *                                                               *
    *****************************************************************/

#include <avr/io.h>
#include <inttypes.h>

#include "lcd.h"

/*****************************************************************
*
*   Author: Dan Kass kassd@msoe.edu
*   Purpose: Initializes the serial communications
*   Parameters: void
*   Returns: void
*
*****************************************************************/
void lcd_init(void)
{
    UBRRH = 0;
    //UBRRL = 51;
    UBRRL = UBRR_VAL;                   // 9600 baud
    UCSRB = (1<<TXEN)|(1<<RXEN);    // Enable Tx and Rx
}

/*****************************************************************
*
*   Author: Dan Kass kassd@msoe.edu
*   Purpose: Prints a character out to the lcd
*   Parameters: char. The character that is to be printed
*   Returns: void
*
*****************************************************************/
void lcd_char(char x)
{
        while(!(UCSRA&(1<<UDRE)));  // Wait for UDR empty
        UDR = x;                    // Send char
}

/*****************************************************************
*
*   Author: Dan Kass kassd@msoe.edu
*   Purpose: Prints a string out to the lcd
*   Parameters: char. The point to the character array for the
*       string
```

```
*   Returns: void
*
*********************************************************************/
void lcd_string(char *c)
{
    for(int i=0; c[i] != 0; i++)
    {
        lcd_char(c[i]);
    }
    lcd_char(CR);
    lcd_char(LF);
}

 /*********************************************************************
*
*   Author: Dan Kass kassd@msoe.edu
*   Purpose: Puts the lcd curser in the correct x, y corrdinate spot
*   Parameters: uint8_t, uint8_t. row, column
*       (0,0) is the upper left, 0 <= x or y <= 15
*   Returns: void
*
*********************************************************************/
void lcd_gotoxy(uint8_t x, uint8_t y)
{
    uint8_t output = 128; //cursor at line 0 position 0
    //makes sure the x and y corridnates fit within the lcd screen
    if((x <= 1) && (y <= 15))
    {
        //0 line is 128 + y for each position
        //1 line is 148 + y for each position
        output += (x*20) + y;
    }
    lcd_char(output);
}

/*********************************************************************
*
*   Author: Dan Kass kassd@msoe.edu
*   Purpose: Clears the lcd and returns the curser to (0,0)
*   Parameters: void
*   Returns: void
*
*********************************************************************/
void lcd_clear(void)
{
    lcd_ff();
}

/*********************************************************************
*
*   Author: Dan Kass kassd@msoe.edu
*   Purpose: Sends a form feed to the lcd
*   Parameters: void
```

```
*   Returns: void
*
*******************************************************************/
void lcd_ff(void)
{
    lcd_char(FF);
}

/*******************************************************************
*
*   Author: Dan Kass kassd@msoe.edu
*   Purpose: Sends a backspace to the lcd
*   Parameters: void
*   Returns: void
*
*******************************************************************/
void lcd_left(void)
{
    lcd_char(BS);
}

/*******************************************************************
*
*   Author: Dan Kass kassd@msoe.edu
*   Purpose: Turns the lcdbacklight on or off
*   Parameters: char. Either "ON" or "OFF" #define 1 and 0
*   Returns: void
*
*******************************************************************/
void lcd_backlight(char state)
{
    // sets the light to ON as default
    uint8_t backlight = LIGHTON;
    if ( state == '0' )
    {
        // sets the light to off
        backlight = LIGHTOFF;
    }
    lcd_char(backlight);
}

/*******************************************************************
*
*   Author: Dan Kass kassd@msoe.edu
*   Purpose: Puts the lcd curser in the correct x, y corrdinate spot
*   Parameters: char. Either "ON" or "OFF" #define 1 and 0
*   Returns: void
*
*******************************************************************/
void lcd_switch(char state)
{
    // sets the power to OFF as default
    uint8_t power = DISPLAYOFF;
```

```
    if ( state == '1' )
    {
        //turns the back light on then turns on the display
        lcd_backlight(ON);
        // sets the display on cursor on and character blink
        power = DISPLAYON;
    }
    else
    {
        //When the Display goes off the back light will also go off.
        lcd_backlight(OFF);
    }
    lcd_char(power);
}

/*********************************************************************
*
*   Author: Dan Kass kassd@msoe.edu
*   Purpose: assigns character array to a custom character spot in
*       the lcd
*   Parameters: uint8_t, char*.0 to 7 custom character
*       spot,char pointer to custom character
*   Returns: void
*
*********************************************************************/
void lcd_define_char(uint8_t num, char *data)
{
    lcd_char(num+247);
    for(int i=0; i < 8; i++)
    {
        lcd_char(data[i]);
    }
}

/*********************************************************************
*
*   Author: Dan Kass kassd@msoe.edu
*   Purpose: Prints the custom character out to the lcd
*   Parameters: uint8_t. 0 to 7 custom character spot
*   Returns: void
*
*********************************************************************/
void lcd_custom(uint8_t num)
{
    if(num <= 8)
    //num works becasue the custom chars codes are 0-7
    //and will be stored in the same style
    lcd_char(num-1);
}
```

I had some problems with my LCD Screen I ended up having to borrown a friends screen to test the lab becasue mine is broken. I didn't have to many problems with writing up the libary for the lcd screen to work.