Introduction:

Learning how to program an LCD Display.

Equipment:

- Simon Board
- Atmel Studio 7
- 2x16 Character LCD Module
- Connection Wires

Procedure:

- Open Atmel Studio 7 and create a new assembly project.
- Properly Wire the LCD Module to Simon board.
- Define the ports and control pins used by the LCD module.
- Start the LCD initialization process by setting PortD and PortE to output mode and clearing the data pins.
- Set the function set to initialize the LCD in 8-bit, 2 line, and 5x7 pixel mode.
- Set the display turn on and cursor on bit to 1 and clear the display bits.
- Write the individual characters of your first name and then move the cursor to the second line.
- Write the individual characters of your last name and clear the display.
- Return the cursor home to begin the loop again.

Results:

The LCD Display spelled out my name on both lines and clear it after it was displayed.

Questions:

1. The LCD has three important control pins which are PORTE2, PORTE3, and PORTE4. The pins control read/write, enable, and command/data register. The LCD is initialized to 8-bit mode, 2 line, and 5x7 settings.
2. The LCD is level-triggered because the LCD displays the characters by levels.
3. I had my name displayed only on the first line. I fixed the issue by setting my last name on the second loop.

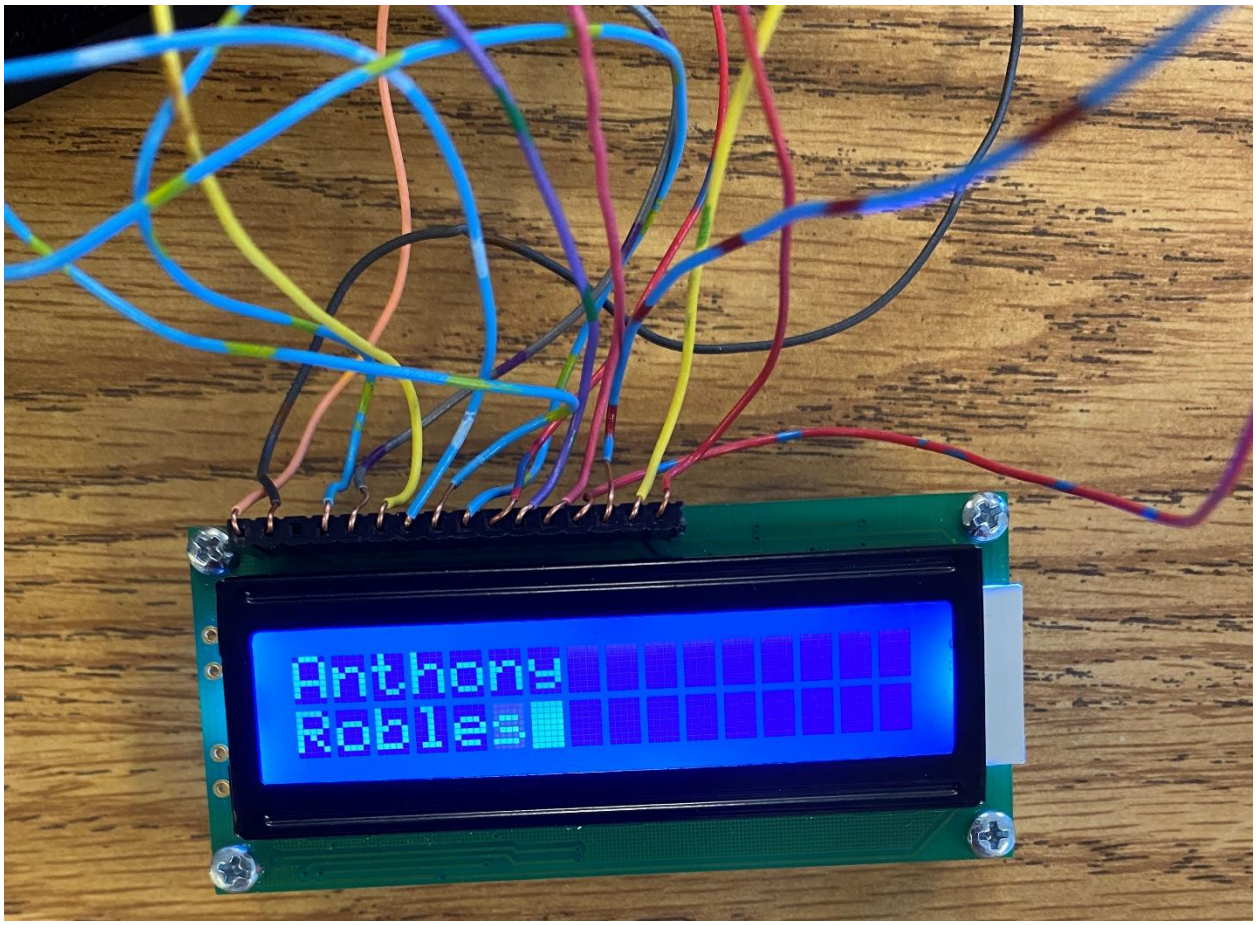
Conclusion:

I learned how to program an LCD Display to print out my name and to clear the LCD after the line is displayed.

Appendices:

Appendix 1:

LCD Display



Appendix 2:

Code:

```
/*
 * lab4.c
 *
 * Created: 3/4/2020 10:45:14 AM
 * Author : argk4
 */

#include <avr/io.h>

#define F_CPU 16000000UL
#include "util/delay.h"

#define LCD_DATA PORTD
#define LCD_DATA_DDR DDRD

#define LCD_CNTRL PORTE
#define LCD_CNTRL_DDR DDRE

#define RS PORTE2
#define RW PORTE3
#define EN PORTE4
```

```
void lcd_clear();
void lcd_write_data(unsigned char data);
void lcd_write_cmd(unsigned char cmd);
void lcd_ininit();
void lcd_init();
void lcd_return_home();
void move_cursor(unsigned char addr);

int main(void)
{
    lcd_ininit();
       lcd_init();

    while (1)
    {
              lcd_write_data('A');
              lcd_write_data('n');
              lcd_write_data('t');
              lcd_write_data('h');
              lcd_write_data('o');
              lcd_write_data('n');
              lcd_write_data('y');
              move_cursor(0x40);
              lcd_write_data('R');
              lcd_write_data('o');
              lcd_write_data('b');
              lcd_write_data('l');
              lcd_write_data('e');
              lcd_write_data('s');
              lcd_clear();
              lcd_return_home();
    }
}

void lcd_clear()
{
       lcd_write_cmd(0x01);
}

void lcd_return_home()
{
       lcd_write_cmd(0x02);
}

void move_cursor(unsigned char addr)
{
       addr |= (1<<7);
       lcd_write_cmd(addr);
}

void lcd_write_data(unsigned char data)
{
       LCD_CNTRL |= (1<<RS); // RS = 1, data register
       LCD_CNTRL &= ~(1<<RW); // RW = 0, write mode
       LCD_DATA = data;
       _delay_ms(100);
       LCD_CNTRL |= (1<<EN); // EN = 1, enable read or write
       _delay_ms(100);
```

```
	LCD_CNTRL &= ~(1<<EN); // EN = 0, disable read or write
	_delay_ms(100);
}

void lcd_write_cmd(unsigned char cmd)
{
	LCD_CNTRL &= ~(1<<RS); // RS = 0, command register
	LCD_CNTRL &= ~(1<<RW); // RW = 0, write mode
	LCD_DATA = cmd;
	_delay_ms(100);
	LCD_CNTRL |= (1<<EN); // EN = 1, enable read or write
	_delay_ms(100);
	LCD_CNTRL &= ~(1<<EN); // EN = 0, disable read or write
	_delay_ms(100);
}

void lcd_ininit()
{
	LCD_DATA_DDR = 0xFF; // port in output mode
	LCD_DATA = 0x00; // clear port pins
	LCD_CNTRL_DDR |= ((1<<RS)|(1<<RW)|(1<<EN)); //  port pin set to output mode
	LCD_CNTRL &= ~((1<<RS)|(1<<RW)|(1<<EN)); // clear port pins
}

void lcd_init()
{
	_delay_ms(25);
	lcd_write_cmd(0x30);
	_delay_ms(10);
	lcd_write_cmd(0x30);
	_delay_ms(5);
	lcd_write_cmd(0x30);
	// function set
	// 0,0,1,DL=1(8bit), N=1(2line), F= 0(5x8 set), 0,0
	lcd_write_cmd(0b00111000);
	// display on
	// 0,0,0,0,1,D=1(disp on),C=1(Cursor on), B=1(blink)
	lcd_write_cmd(0x0F);
	// clear display
	lcd_write_cmd(0x01);
	// Entry mode set
	// 0,0,0,0,0,1,I/D=1, SH=0
	lcd_write_cmd(0b00000110);
}
```