EE4930 Advanced Microprocessor

Section 011/021, Winter 2022/23

Professor: Kerry R. Widder, Ph.D.

Electrical Engineering and Computer Science Department

Milwaukee School of Engineering

Laboratory 1: “Tool Time”

Name: Jorge Jurado-Garcia

Courses: EE 4930 011

Date: 12/9/2022

# Objectives

This lab objectives are to install Code Composer Studio (CCS) software tool and become familiar with using it. Afterwards we plan on gaining experience using the MSP432 and the LCD display. We wrote a simple C program that on the first line of the LCD prints the text “EE4930-011”, using a pushbutton to change the rest of the LCD screen. If the pushbutton is pressed, write **ON** and your **last name** on the **second** and **fourth** lines of the screen. Else we will display **OFF** and your **First Name** on the first and third lines of the Screen. Regardless of the pushbutton state the previous state of the changed lines should be erased.

# Description

For this lab, I installed code composer studio and downloaded the MSOE LCD library per the directions of Professor Johnson PowerPoint guide, on this site. Afterwards I included the library folder into my workbench and appended it to my working Code Composer Project. With setup completed I started to create my Lab\_1 module that included the needed functions to write my program. With the module complete and tested. I then moved into my main and initialized my lab1 code and using write a simple if-else logic statement in a while loop for reading the pushbutton and changing the state of my LCD screen.

# Conclusion

Some difficulties that I had when starting this program and getting a refresher involved the setup of my project and implementing the MSOE LCD library. I later found out the reason of my issues where my Project path having spaces and ‘-‘ which caused some unknown errors. To fix this issue I simply moved my workspace and project to my C drive and restarted the process. This seemed to have fixed the issue and allowed me to continue. From this lab I was able to learn a lot about the functionality of MSOE LCD library as well as how to read legacy or other engineer’s code. From my experience I learned the power of documentation and leaving comments when needed how to apply it appropriately to my code.

# Attachments

## Main

/\* Copyright (C) 2022 MSOE

\*

\* All Rights Reserved

\* You may not use, distribute or modify this code without the

\* express written permission of MSOE

\*

\* Contact Info

\* jorgejuradogarcia2@gmail.com

\* 608-312-5950

\*

\*/

**#include** <stdio.h>

**#include** "msp432.h"

**#include** "msoe\_lib\_all.h"

**#include** "defines.h"

// Lab modules

**#include** "EE4930\_LAB1.h"

**#include** "EE4930\_LAB2.h"

**#define** LAB1\_ON TRUE

**#define** LAB2\_ON FALSE

**#define** CLEAR 21

/\*

\* main.c

\*/

**unsigned** adc\_val;

**float** ADC\_Percentage;

**float** PWM\_dutycycle;

**char** LCD\_SET\_FLAG=0;

**int** Calculated\_dutycycle;

**int** **main**(**void**){

**#if** LAB1\_ON

Init\_Lab1();

**while**(1)

{

Lab1\_Poll();

}

**#endif**

## EE4930\_LAB1.h

/\*

\* EE4930\_LAB1.h

\*

\* Created on: Dec 1, 2022

\* Author: jurado-garciaj

\* Initialize GPIO. Setup inputs and outputs as follows:

\*

\* Port pin in/out Pullup/down Connect

\* P1.1 In UP s1

\* P2.2 Out N/A LED2-B

\*

\*

\*\*\*\*\*\*\*\*\*\*\*\* Nokia LCD interface reference \*\*\*\*\*\*\*\*\*\*\*\*\*\*

//

// Red SparkFun Nokia 5110 (LCD-10168)

// -----------------------------------

// Signal (Nokia 5110) LaunchPad pin

// 3.3V (VCC, pin 1) power

// Ground (GND, pin 2) ground

// UCA3STE (SCE, pin 3) connected to P9.4

// Reset (RST, pin 4) connected to P9.3

// Data/Command (D/C, pin 5) connected to P9.2

// UCA3SIMO (DN, pin 6) connected to P9.7

// UCA3CLK (SCLK, pin 7) connected to P9.5

// back light (LED, pin 8) not connected, consists of 4 3.3 V white LEDs which draw ~80mA total

\*

\*

\*/

**#ifndef** APP\_EE4930\_LAB1\_H\_

**#define** APP\_EE4930\_LAB1\_H\_

**extern** **void** **Init\_Lab1**( **void** );

**extern** **void** **Lab1\_Poll**( **void** );

**#endif** /\* APP\_EE4930\_LAB1\_H\_ \*/

## EE4930\_LAB1.c

/\*

\* EE4930\_LAB1.c

\*

\* Created on: Dec 1, 2022

\* Author: jurado-garciaj

\* Initialize GPIO. Setup inputs and outputs as follows:

\*

\* Port pin in/out Pullup/down Connect

\* P1.1 In UP s1

\* P2.2 Out N/A LED2-B

\*

\*

\*\*\*\*\*\*\*\*\*\*\*\* Nokia LCD interface reference \*\*\*\*\*\*\*\*\*\*\*\*\*\*

//

// Red SparkFun Nokia 5110 (LCD-10168)

// -----------------------------------

// Signal (Nokia 5110) LaunchPad pin

// 3.3V (VCC, pin 1) power

// Ground (GND, pin 2) ground

// UCA3STE (SCE, pin 3) connected to P9.4

// Reset (RST, pin 4) connected to P9.3

// Data/Command (D/C, pin 5) connected to P9.2

// UCA3SIMO (DN, pin 6) connected to P9.7

// UCA3CLK (SCLK, pin 7) connected to P9.5

// back light (LED, pin 8) not connected, consists of 4 3.3 V white LEDs which draw ~80mA total

\*

\*

\*/

// Included files

**#include** <stdio.h>

**#include** "msp432.h"

**#include** "msoe\_lib\_all.h"

**#include** "defines.h"

// Drivers

**#include** <MSP432P401R\_GPIO.h>

**void** **PushButton\_Pressed**( **void** );

**void** **PushButton\_NotPressed**( **void** );

**void** **Init\_Lab1**( **void** )

{

Stop\_watchdog(); // stop Watch dog timer

Set\_ports\_to\_out(); //sets all the ports to outputs to prevent floating inputs

LCD\_Config(); // Configures the LCD display screen clears display afterwards

GPIO\_setAsInputPinWithPullUpResistor( GPIO\_PORT\_P1, GPIO\_PIN1 ); //Set P1.1 as an Input with Pull up resistor high

GPIO\_setAsOutputPin( GPIO\_PORT\_P2, GPIO\_PIN2); // P2.2 as an output pin

GPIO\_setOutputLowOnPin( GPIO\_PORT\_P2, GPIO\_PIN2 ); //Set P2.2 Low

//Printing "EE4930-011" on the first line of the LCD screen

LCD\_goto\_xy(0,0);

LCD\_print\_str("EE4930-011");

}

**void** **Lab1\_Poll**( **void** )

{

**static** **char** Previous\_Press = 0;

**char** Current\_Press;

// read the press from the input

Current\_Press = GPIO\_getInputPinValue( GPIO\_PORT\_P1, GPIO\_PIN1 );

**if**( Previous\_Press != Current\_Press)

{

**if**( Current\_Press == PRESSED ) // value will be 0 if pressed

{

PushButton\_Pressed();

}

**else**

{

PushButton\_NotPressed();

}

}

Previous\_Press = Current\_Press;

// do a while loop, while pushbutton

}

**void** **PushButton\_Pressed**( **void** )

{

//Turn off an LED

GPIO\_setOutputHighOnPin( GPIO\_PORT\_P2, GPIO\_PIN2 ); //Set P2.2 Low

// Display text on the second and fourth row

LCD\_goto\_xy(0,1);

LCD\_print\_str("ON ");

LCD\_goto\_xy(0,3);

LCD\_print\_str("Jurado");

LCD\_goto\_xy(0,2);

//blank out the third row

LCD\_print\_str(" "); //clearing by writing nothing

}

**void** **PushButton\_NotPressed**( **void** )

{

//turn on an LED

GPIO\_setOutputLowOnPin( GPIO\_PORT\_P2, GPIO\_PIN2 ); //Set P2.2 Low

// Display text on the second and third row

LCD\_goto\_xy(0,1);

LCD\_print\_str("OFF");

LCD\_goto\_xy(0,2);

LCD\_print\_str("Jorge");

LCD\_goto\_xy(0,3);

//blank out the Fourth row

LCD\_print\_str(" "); //clearing by writing nothing

}

## Wiring Diagram

Diagram, schematic

Description automatically generated

Figure 1. Wiring Diagram for LCD to MSP432 Board