Lab 3 – User Input and LCD

1 INTRODUCTION

In this lab you will develop a PSoC Creator project to write messages to the LCD, and update the LCD based on user pushbutton usage.

The LCD can be useful for debugging, and for application development.

2 PART 1, PROCEDURE

Add a "Character LCD" device to your project. Your main.c program will consist of an initialization part, and an infinite loop. In the initialization part put any component initialization procedure calls that are necessary.

Add a digital input pin, and map its pin to one of the pushbuttons on the board.

Add to the initialization part code to display the following message:

```
My First Microcontroller I/O Console Program
Microcontroller Programmer
Press a switch to continue...
```

Write code in the infinite loop that waits for a button press, and then updates the display to:

```
Program completion succeeded
```

At this point the program will just stay in an infinite loop.

Build your project and download it to the development board. Keep notes as you debug your project, and include your debugging notes in your report.

Demonstrate your working project to your instructor.

3 PART 2, PROCEDURE

Create a project to meet the following requirements:

1) Upon program START, the system will turn on two LED's.
2) The system will display a greeting message.
3) The system will display a prompt message asking the user to push a button switch.
4) The system will wait for that action by the user.
5) The system will then instruct the user to press one of the two buttons to toggle the state of the corresponding LED, and enters the following infinite loop:

a. The system will wait for a button press on either button, and will toggle the corresponding LED.
b. The system will update the LCD, indicating which LED was toggled last.

4 NOTES

1) You will need to initialize your LCD component.
2) You will need to map the pins of your LCD component to the physical chip pins that are wired to the LCD on the board. You can find the port/pins that are connected to the LCD by reading the Development Kit Guide.
3) You will need to manage the display of lots of characters on the limited 16x2 line character display. You can use the horizontal scrolling procedure included in the API, or work your own vertical scrolling, etc., using time delays to make the display pleasant for humans.
4) You will need a pullup resistor on your pushbutton line – this can be accomplished as a configuration of the digital input pin component.
5) A pushbutton has a very long "bounce" – meaning that each time the button goes up or down, the PSoC sees an erratic stream of 0's and 1's, until ultimately settling on the new state. The bounce period is generally less than 100 ms. Your software must take this into account, so that the toggling only happens once for each button press. There are many ways that you can "debounce" the button – one way is to delay some amount (e.g. 100 ms) after each detected button_up, or button_down, before resampling the pin state.

5 REFERENCES

- CY8CKIT-050 PSoC® 5LP Development Kit Guide

6 WRITE A REPORT

The report is to include, but not limited to the following:

a) **Cover Sheet** with Title, Class, Names, etc.
b) **Introduction**.
c) Brief recap of **Procedure**. You can paste in code snippets, but use mono-spaced font and ensure that code is well formatted. Include any equations or other relevant information that helps you to explain what, why and how you did what you did.
d) **Results**, presented in tables, figures, or other organized means, and a discussion of the results that you obtained. You should include your debug experiences under **Results**.
e) The report must be understandable to another engineer not working on this project.
f) A conclusion of your results and discussion of anything you found especially interesting or not expected from your work on this project.

REPORT NOTES:

- One report per team

- You may use the IEEE paper format, if you like: (template is 2014_04_msw_usltr_format.doc). In this case the cover sheet info is embedded at the top.
- Microsoft Word, .docx file.
- Upload via GitHub (**one upload per team**).
- Also upload all code files that you wrote, or hand edited (and **only** those files!), via GitHub (**one upload per team**).