California State University, Northridge

ECE 422L - Computer Architecture

Fall 2022

Lab 4

Blinky LED on Tiva C LaunchPad

March 15, 2023

Instructor: Dr. Shahnam Mirzaei

Mark Attia

**Table of Contents**

**Introduction 2**

**Procedure 2-3**

**Testing Strategy 4**

**Results 4-5**

**Conclusion 6**

**Appendix 6**

**Introduction**

The purpose of this experiment is to learn the process of using the Tiva C Series datasheet to locate the necessary registers and information about memory mapping for our application. The datasheet will be our go to for any information regarding the microcontroller, so it is crucial to begin getting familiar with it. By the end of this experiment, we will build on our understanding of how to locate the necessary registers and peripherals for RGB and button manipulations using the Tiva-C board. This lab will allow us to integrate different memory maps of the board together and test switch/button debouncing.

**Procedure**

1. Create Project Lab5

2. Ensure all details are correct for compiling code

3. Simulation debugger

4. Drivers for board installed -Using Tiva Core

5. Use sample program provided in manual and test for proper execution

6. Carefully monitor memory activity in step-by-step debug mode

7. Optimize the code given in task 1.

7. Complete task 2

Diagram

Description automatically generated

**Testing Strategy**

The testing strategy for this experiment comes from being able to understand what should be happening inside of our memory based on what we have learned memory mapping and what addresses are needed to control LEDs and buttons. So long as we understand how to set things like GPIO\_DIR and offsets, we should be able to control LEDs simply by writing values to single bits in the proper registers by using the OR operation. The difference here is that we will be doing this conditionally based on a button press by reading the state of that button and setting a condition.

**Results**

A picture containing text

Description automatically generated

Text

Description automatically generated

Figure 1: No debounce LED Toggle

Logic for toggling LED on button press.

Graphical user interface, text, application, email

Description automatically generated

Text

Description automatically generated

Figure 2: Adding Delay Debounce

Logic stays identical except for adding a delay to debounce button presses.

<https://photos.app.goo.gl/fs1UeRV7MT87dotH9>

**Conclusion**

This experiment allowed us to get introduced to using the datasheet and accessing information about registers, memory mapping, and RGB manipulation. We learned how to write to a specific registers in a specified location and toggle specific bits to control certain RGB LEDs. The twist here was integrating that logic from our previous experiment and tying it in with a button press, which has a memory location of its own. Then we got to add switch debouncing in the form of delay to regulate the signals being generated by a button press. The main issue we had here was that there didn’t seem to be any crazy issues or varying signals being sent even before implementing the debounce. This caused both implementations to be essentially the same. I understand how debouncing works and have seen the impacts of not properly debouncing inputs, yet this did not seem to be a great example of that and I’m not sure why. This is the fundamentals to learning how to manipulate the different aspects that Tiva-C series or any microcontroller has to offer.

**Appendix**

https://github.com/EgyptiansFTW/Computer-Architecure-Lab.git