mbed Lab 1: Hello World and Pushing Buttons

CS 220

Oct. 23

Introduction

Hello, World! is traditionally a programmer's first program in a new language or in a new development environment. In this lab, you'll start with the mbed analogue to Hello, World! to orient yourself both to the mbed development environment and the Freescale FRDM-KL25Z (KL25Z) ARM processor-based microcontroller board that you'll be working with in labs for the rest of the semester.

Lab Objectives

- 1. Create an mbed development account and configure it for the KL25Z.
- 2. Download and run KL25Z programs.
- 3. Make simple modifications to a KL25Z program, controlling the board's built-in tri-color LED using the mbed's DigitalOut interface.
- 4. Make further modifications to the program to accept control from a button, using the mbed's DigitalIn interface.

Lab Procedure

Read each of the following steps **completely** before acting on them.

- 1. Visit the developer.mbed.org website. Create an account and sign-in.
- 2. Go to the Platforms page. From there, go to the FRDM-KL25Z page. Find the "Add to your mbed Compiler" button and click it. (This step is necessary so that the compiler knows what architecture to generate code for. Code generated for one ARM chip won't necessarily run on another ARM chip. I wonder why...)

3. Connect the small end of your USB cable to the KL25Z's USB port labeled "SDA." Refer to this figure:



- 4. Connect the other end of your USB cable to your workstation. Driver software will begin to install. After the installation completes, you'll notice that an MBED drive has appeared in My Computer.
- 5. Back in your web browser, on the KL25Z platform page, jump/scroll down to the "Downloading A Program" section and follow the steps there to download and run Hello, World!.
- 6. Scroll down the page a tiny bit, and import the mbed_blinky program. The import process should open the development IDE in a new browser tab. You'll need your original mbed tab in order to refer to mbed documentation, so keep it open.

7. Open the main.cpp source file (Yes, you're about to become a C++ programmer.). Here's a copy, with comments drawing analogies to Java constructs:

```
#include "mbed.h"
                          // Similar to Java's import.
DigitalOut myled(LED1);
                          // DigitalOut is a class name. LED1, a KL25Z
                          // pin name, is the argument to DigitalOut's
                          // constructor. myled is the name of the
                          // created object. Most KL25Z pins can be
                          // programmed as inputs or outputs.
int main() {
                          // Similar to "public static void main()" in Java.
    while(1) {
        myled = 1;
                          // Set the pin to 1.
        wait(0.2);
                          // Wait for 0.2 sec.
        myled = 0;
        wait(0.2);
    }
}
```

The pins names LED1, LED2, and LED3 have aliases — LED_RED, LED_GREEN, and LED_BLUE, which are slightly more descriptive.

8. Modify the program so that it blinks a white light at a rate of 1 sec. on and 1 sec. off. Compile the program, which will have the side-effect of downloading it. Confirm that the running program behaves as it should.

Does setting an LED's pin to 1 turn the LED on or off? Physics majors/minors and electronics geeks: Explain why.

9. If you've never used a breadboard, or it's been a while since you've used one, navigate to the Wikipedia page on the topic and read the section on Solderless Breadboards. You can stop after you've read the subsection covering Jumper Wires.

10. You're going to connect your button to the breadboard. Internally, the button's electrical connections are like this:



Orient your button so that pins C and D line-up with column D on your breadboard with pin C in row 10 and pin D in row 12. Pins A and D should line-up with column G, like so:



Insert the button into the breadboard.

- 11. You're about to wire the button to the KL25Z. It's generally a very bad idea to modify the wiring of an electronic device while it's under power, so disconnect the USB cable at the KL25Z end. Otherwise, you may damage the KL25Z in which case your wallet will also be damaged! Be careful while disconnecting the cable, because mini USB connectors aren't exactly known for their durability.
- 12. For the following wiring steps, use the separate KL25Z pin diagram to locate KL25Z pins.
- 13. Grab two jumper wires. In electronics, the convention is to use red wires for the voltage supply and black wires for ground. So, try to find a black wire for your ground connection and a wire any color other than red or black for your button to KL25Z input pin connection.

14. On the KL25Z, locate a GND pin (I used the GND pin across from pin PTE4.) Using your black wire, wire one electrical side of your button to GND. Now, locate pin PTA1 and wire it to the button's other electrical side.

Your connections should look pretty much like this:



and this:



Have your partner double-check that you have the KL25Z oriented the same way as in the

diagram above and that you've connected the correct pins. If you want to triple-check, have me take a look at your wiring job.

15. You've just built the following switch circuit:

By default, DigitalIn interfaces on the KL25Z have a pullup resistor attached to them. As the name implies, this pulls the input pin up to logical 1 ($V_{in} = 3.3V$). Closing the switch (pushing the button), directly connects the KL25Z's input pin to logical 0 (Ground, usually abbreviated GND). Physics majors/minors and electronics geeks: explain the purpose of the pullup resistor.

- 16. Reconnect the KL25Z to power. The white light should resume blinking.
- 17. In the mbed Handbook, find the documentation for the DigitalIn interface. You'll notice that there's sample code.
- 18. Using the sample code as an example (What pin will you pass to your DigitalIn object's constructor? It's neither the p5 nor the PTB0 pins shown in the sample code.), modify your program so that the button controls the LED. With the button released, the LED should be off, and with the button pushed, the LED should be on.

KL25Z Pin Diagram



USB connector-end of the $\rm KL25Z$