**ES LAB 1: LED & PUSHBUTTON.**



A circuit board with different colored lights and wires

Description automatically generated with medium confidence

*Circuit used: the red wire is to simulate being connected to the 5V pin. The black wire is to simulate being connected to the GND pin. The green wires are to simulate being connected to my chosen pins that you will read about later.*

Student: Johnson Domacasse

Date: 12 sept 2023

Student#: 4471709

Teacher: Suzana Andova

## Introduction:

The first lab assignment for ES is to gives us an understanding of how we can use simple bit manipulation, low-level programming and basic circuitry to understand how an Stm32 works. Through out this lab there will be a number of things we go through in order create what needs to be made.

I will write this report for the teacher to understand my way of thinking and how I got to my solution. I will also write this report in a way that I can come back to it in the future if I forget my basics regarding low-level programming.

## Procedure:

In this section of the report I will be going over what research I did, I step by step guide on where I got the information from and how I translated this into a solution. For reference I will be using the following documents in search for information: UM1724 user manual, STM32F303xE data sheet and the RM0316 reference manual. I also did some self study by going through some videos and PowerPoint presentation that were provided to me.

### LED setup:

In order to get the LED to work, we must configure a few things on the board itself. Here is my personal step by step guide on how I did it.

#### Step by step guide for input:

* Determine which pins you will be using on your board.
* Determine which port these pins are connected to. **(user manual page 31**)
* Enable the clock that is associated with the previously determined port. **(reference manual section 9.4.6)**
* Set up the MODER I/O to the designated pin number. They are all set to **ANALOG** by default. **(reference manual section 11.4.1**)
* Configure the OTYPER to the designated pin number. Coming future you will use push-pull but if you are working with a component whose signals are constantly fluctuating, then you would use open-drain. **(reference manual section 11.4.2)**
* Use the ODR to manipulate your pin whether it be a high signal or low signal. **(reference manual section 11.4.6)**

By following the step by step guide I was able to set up my LED. For this Lab I am using pin PA5 and PA6 for my LEDs. One of them is internal and the other is a separate pin I chose *specifically*. I tested by simply turning it on the while loop to see if there was an issue in my configuration. Luckily there weren’t any. Now that I had the both LEDs working I went on to the next section.

### Button setup:

In order to get the button to work, we use a similar approach as the LED with some minor changes.

#### Step by step guide for output:

* Determine which pins you will be using on your board.
* Determine which port these pins are connected to. **(user manual page 31**)
* Enable the clock that is associated with the previously determined port. **(reference manual section 9.4.6)**
* Set up the MODER I/O to the designated pin number. They are all set to **ANALOG** by default. **(reference manual section 11.4.1**)
* Configure the PUPDR to the designated pin number. This you will change depending on which I/O configuration you want for your button. **(reference manual section 11.4.4)**
* Use the IDR to read and process from your pin. **(reference manual section 11.4.5)**

Like before, I used the above mentioned step by step guide to set up my button configurations. For this Lab I will be using the internal button assigned at PC13. I will use another *specifically* chosen pin PC7 for my external button. I tested to see if it works by using a simple program to turn the LED on if the button is pressed. I did this for both buttons to see if there were any issues. They work fine.

### Final solution:

Now That I have both my buttons and LEDs working it was a matter of combining the two in order for them to work. When the first button is pressed, it will turn an LED on and when it is released it will turn the LED off. When the second button is pressed, it will turn toggle the other LED on and off until the button is released.

### Something cool:

For my something cool section I wanted to see how power is being transferred to the LED pins when the buttons are pressed. Ideally I would’ve used an oscilloscope for this. I used a multi meter to measure the voltage everytime the button is pressed. When the button is pressed the results will show that there is some voltage being sent to the LED.