Pushbutton and GPIO Input

ELE 271: Laboratory 4

Introduction

The basic goal of this experiment is to learn how to use the general purpose input output (GPIO) pins for digital input. Note that the GPIO pins on the NUCLEO board are 5 V tolerant.

Part 1

In the first exercise the goal is to use the blue pushbutton (PC.13) to turn the green LED (PA.5) on and off.

Configure GPIO PA.5 pin as output.

Configure GPIO PC.13 pin as input.

Pressing the pushbutton should turn-off the LED. Releasing the button should turn it back on.

```
#include "stm32l476xx.h"

// PA.5  <--> Green LED
// PC.13 <--> Blue user button

void configure_LED_pin(){
  // Enable the clock to GPIO Port A
    <your code goes here>
  // GPIOA Mode: Output(01)
    <your code goes here>
  // GPIOA Speed: Low speed (00)
    <your code goes here>
  // GPIOA Output Type: Output push-pull (0, reset)
    <your code goes here>
  // GPIOA Push-Pull: No pull-up, no pull-down (00)
    <your code goes here>
}

void configure_BUTTON_pin(){
    // Enable the clock to GPIO Port C
        <your code goes here>
    // GPIOC Mode: Input(00)
        <your code goes here>
    // GPIOC Push-Pull: No pull-up, no pull-down (00)
        <your code goes here>
}

void turn_on_LED(){
```

```
    <your code goes here>
}

void turn_off_LED(){
    <your code goes here>
}


int main(void){

    configure_LED_pin();
    configure_BUTTON_pin();

  // Dead loop
    while(1){
        if(<your code goes here>) { // check value of GPIOC->ODR bit 13
            turn_on_LED();
        } else {
            turn_off_LED();
        }
    }
}
```

Modify the code so that each push-release sequence toggles the LED (see Example 14.7 in the textbook).

Part 2

Use the pushbutton to blink the LED or have it turned-off. When pushbutton is released the LED blinks, when pressed the LED is off.

Part 3

Use a counter to capture the length of time the pushbutton is pressed. Once the pushbutton is released turn on the green LED for exactly that same period of time.

Part 4

Connect PB505 ground to STM32NUCLEO ground. Connect the function generator on the PB505 to pin PC.1. Configure the pin as input. Set the function generator to square wave (use TTL output only!!!) and use a counter to determine the period of the square wave. Record count values for frequencies 10 Hz, 100 Hz, 1 KHz, 10 KHz and 100 KHz. Use the logic analyzer or oscilloscope to adjust the frequencies as close to the given values as possible. Record the actual frequencies used.