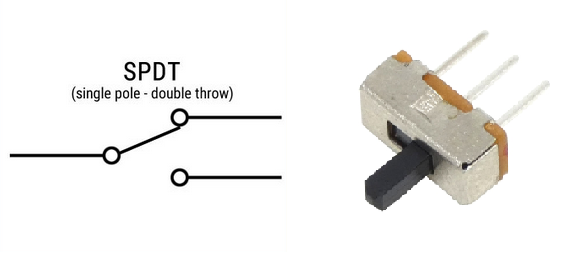
**CMPE 443 PRINCIPLES OF EMBEDDED SYSTEMS DESIGN**

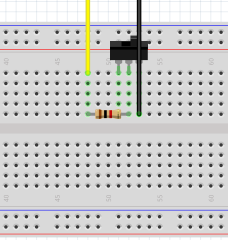
**PRELAB #005 “Data Structure”**

1. **Problem Definition**

There are some user LEDs on the board. You will use these LEDs and a spdt switch in this prelab. When the switch head is at the left side, Red, Blue, Green LEDs will be turned on in sequence with 1 second intervals (Red ON others OFF, 1 second later Blue On others Off …). When the switch head is at the right, it will stop until switching the head location.

1. **SPDT Switch**

****

****

Right leg of the switch should be connected to the GND. For the input, you will choose a suitable pin.

* Which pin do you choose for input? \_\_\_PB11\_\_\_\_\_\_\_
* What should be its pull type? \_\_\_up\_\_\_\_\_\_\_

1. **Struct Definitions**

In order to write a readable code, you need to define the registers in a data structure.

* Define the GPIO data structure by only using **uint16\_t**
* typedef struct
* {
* volatile uint16\_t MODER[2];
* volatile uint16\_t OTYPER[2];
* volatile uint16\_t OSPEEDR[2];
* volatile uint16\_t PUPDR[2];
* volatile uint16\_t IDR;
* uint16\_t RESERVED1;
* volatile uint16\_t ODR[2];
* volatile uint16\_t BSRR[2];
* volatile uint16\_t LCKR[2];
* volatile uint16\_t AFRL[2];
* volatile uint16\_t AFRH[2];
* volatile uint16\_t BRR[2];
* uint16\_t RESERVED2[2];
* volatile uint16\_t SECCFGR[2];
* } GPIO\_Type;

1. **Code**

In this prelab, you need to write code for solving the problem which is described earlier, with data structure and you will write inline assembly code for counting sequence numbers (increase at every sequence end).

1. **Submission**

You will submit one zip file which contains this document and your project (all the files with the last configuration)

The naming of the zip file should be:

PRELAB<exp num>\_<StudentID>.zip

1. **Related Videos and Links**

STM32 GPIO Input:

<https://www.youtube.com/watch?v=JZsC34jfbEg>

STM32 Inline Assembly

<https://www.codeinsideout.com/blog/stm32/assembly/>