Control & Automation Engineering Department

KON309E Microcontroller Systems

Midterm Project

Question 1:

You are expected to do a research and prepare a **detailed** report about the register **assigned** to you. You can find your assigned register in the file named "register_list" on Ninova.

In your report, introduce the register, explain its function and how it is used, and give application examples.

Question 2:

a) You are expected to accomplish a speed control simulation using 3 LEDs, 2 buttons and a potentiometer.

Materials:

- 3 LEDs (green, yellow, red)
- 2 Buttons
- USB-TTL Converter
- Potentiometer
- Breadboard
- Jumper Cables

Task:

- Design a circuit with 3 LEDs, 2 buttons and a potentiometer.
- While the button 1 is **held down (held pressed)**, duty cycle of the PWM signal will increase 10% **each 0.25 seconds** $(0\% \rightarrow 10\% \rightarrow 20\% \rightarrow \ldots \rightarrow 90\% \rightarrow 100\%)$.

- In the beginning, when the button 1 is held down for 0.25 seconds, the green LED will be ON with a 10% duty cycle PWM signal.
- While button 1 is held down for total of 0.5 seconds, duty cycle of green LED's PWM signal will be 20% and after 2.5 seconds duty cycle will be 100%.
- If button 1 is still held down **after 0.25 second while the duty cycle for green LED is 100%,** yellow LED will be ON with 10% duty cycle.
- Same process is also valid for yellow and red LEDs.
- When button 1 is released at any time, duty cycles will **remain as they are**.
- Button 2 have the **reverse** effects. For each 0.25 second button 2 is held down, PWM signal will **decrease** by 10% $(100\% \rightarrow 90\% \rightarrow 80\% \rightarrow \ldots \rightarrow 10\% \rightarrow 0\%)$.
- You can think of button 1 as the gas pedal and button 2 as the brake pedal.
- Take 100km/h as maximum speed.
  - 90% duty cycle for green LED will correspond to 30 km/h and 10% duty cycle for red LED will correspond to 70 km/h of speed.
- With the potentiometer, you will **restrict the maximum** speed.
  - Map the value you read from potentiometer to a speed value.
  - E.g. if your potentiometer is half way through, you can only speed up to 50 km/h.
- Use external intterupts for the buttons and output compare timer interrupts to generate PWM signals.
- As you know, a potentiometer is a type of resistor. You can adjust the resistance by rotating the knob (Figure 1).

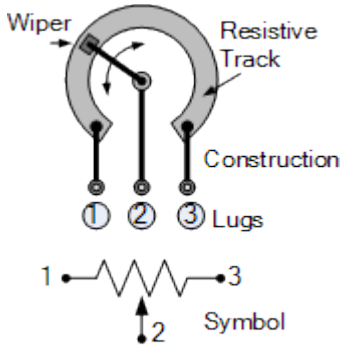
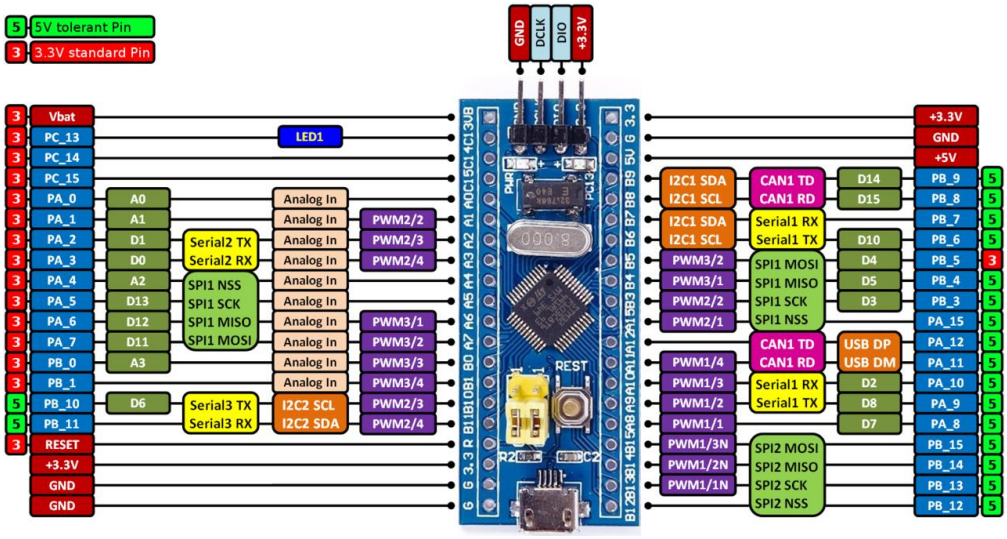


Figure 1: Potentiometer and its connections.

b) You are expected to send the speed data to PC via UART and visualize it in **real-time** on the MATLAB environment.

Prerequisites:

- Following image displays the pinout of our microcontroller board. Each pin is assigned to perform different functions. In this question, we are interested in the pins related with serial communication (TX and RX pins).



- Firstly, you should setup your pins (e.g PB6 and PB7 for usart1).
- Secondly you need to configure your USART communication parameters as:
  Baudrate: **9600,** Data Bits: **8**, Parity: **None**, Stop Bits: **1**
- After finishing setup, you will create a byte sender function which sends a given byte value through USART.
- You can send the data to PC at 0.5 second intervals.

Task:

In question 2, you are supposed to construct a LED speedometer controlled by the buttons. For better understanding your speed scale should be designed as follows:

0					50					100	0 100	0 100
GREEN%											YELLOW%	RED%

As it is seen, you have a scale between 0 and 300 for this speedometer. Maximum speed ranges between 0 and 100 thus you need to map speedometer scale to real speed (0 to max_speed) and send this speed information to computer via USART communication.

It is planned to introduce you a MATLAB script to visualize speed data in real-time obtained from STM32 board.

Note: If there is anything you do not understand, please contact one of the course assistants.

Please consider the following steps when preparing your reports.

1. Describe the experiment **in your own words**.
2. Add your main codes as screen shots.
   - Don't forget to comment your codes **in your own words** explaining how each line of code works.
3. Add a photo of your whole circuit.
4. Include the plots you generate using MATLAB

Take a video of your system while running, upload it on YouTube, Drive, etc. and include the link on your report.