# README - STM32F4 Button Interrupt with USART Communication

## Project Overview

This project is designed for the STM32F4 microcontroller and demonstrates how to:

- Configure GPIOC Pin 13 (PC13) as an external interrupt (EXTI) input.

- Implement debouncing for button press detection.

- Send a message "CMSY's Team\r\n" over USART2 when the button is pressed.

- Use the SysTick timer for millisecond-accurate delays.

## Hardware Requirements

- STM32F4 series board (e.g., STM32F407 Discovery).

- USB-to-serial adapter for USART2 communication (or ST-Link for monitoring).

- Push button (if not already available on the board).

- LED (optional) for debugging.

## Project Features

- GPIO Configuration:

- PC13 is configured as an input with an internal pull-up resistor.

- External Interrupt (EXTI13):

- Configured to trigger on rising edge.

- Uses NVIC to enable EXTI interrupts.

- SysTick Timer:

- Configured for 1 ms timing to facilitate software debouncing.

- USART2 Communication:

- Configured for 9600 baud rate.

- Sends a message "CMSY's Team\r\n" on a button press.

## Software Implementation

### 1. GPIO Configuration

- Enables GPIOC clock.

- Configures PC13 as an input with an internal pull-up resistor.

### 2. EXTI13 Interrupt Configuration

- Maps PC13 to EXTI line 13.

- Enables rising edge trigger detection.

- Enables NVIC for EXTI15\_10 interrupt line.

### 3. SysTick Timer Configuration

- Configured to generate an interrupt every 1 ms.

- Used for debouncing the button.

### 4. USART2 Configuration

- Configured to send data at 9600 baud rate.

- Uses PA2 (TX) and PA3 (RX) in Alternate Function mode.

### 5. Interrupt Handler (EXTI15\_10\_IRQHandler)

- Detects button press on PC13.

- Implements a 50 ms debounce delay using SysTick.

- Sends "CMSY's Team\r\n" over USART2.

- Clears the EXTI pending flag.

## Code Structure

|  |  |
| --- | --- |
| File | Description |
| main.c | Contains the main program logic |

## How to Use

1. Flash the firmware onto an STM32F4 board using Keil, STM32CubeIDE, or OpenOCD.

2. Connect a USB-to-serial adapter to PA2 (TX) and PA3 (RX).

3. Monitor the USART output using a serial terminal (e.g., PuTTY or RealTerm).

4. Press the button connected to PC13.

5. The message "CMSY's Team" should appear in the serial monitor.

## Potential Enhancements

- Implement Falling edge trigger for active-low buttons.

- Add LED feedback on button press.

- Improve debounce filtering using a state machine.

## License

This project is open-source and can be freely modified and used.