

LAB: USART - LED, Bluetooth

Date: 2022-11-15

Author/Partner: Dongmin Kim, Jinho Kook

Github:

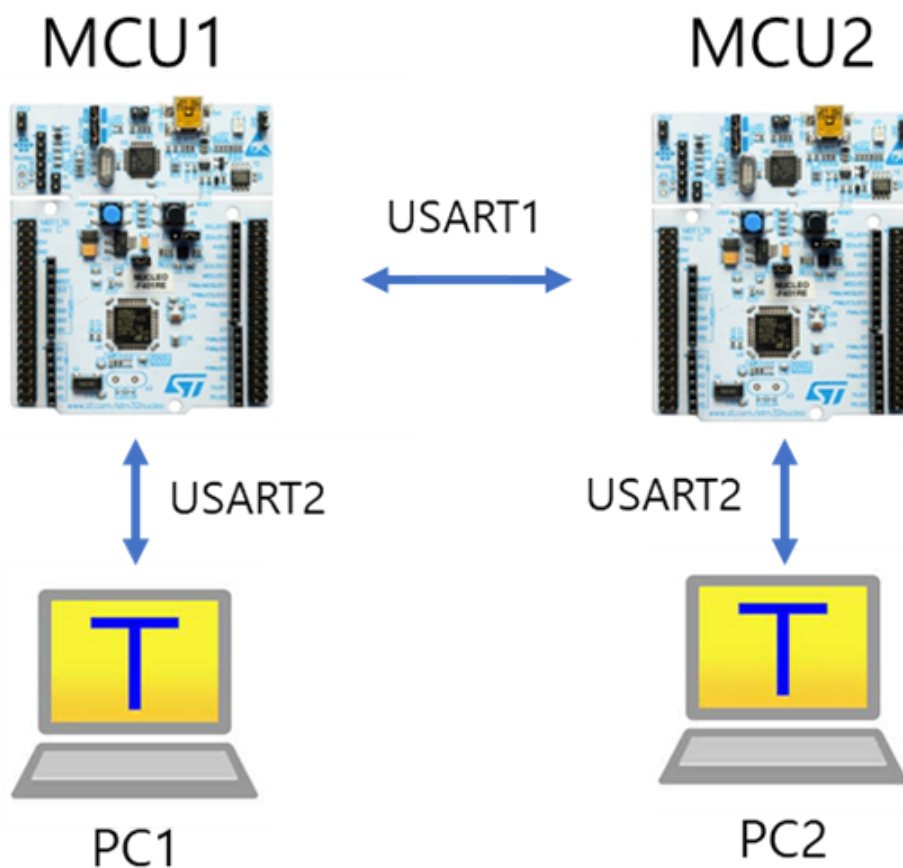
Demo Video:

PDF version:

Introduction

In this lab, we will learn how to configure and use 'USART(Universal synchronous asynchronous receiver transmitter)' of MCU. Then, we will learn how to communicate between your PC and MCU and MCU to another MCU with wired serial communication.

- **Mission 1:** Control LED(LED2) of each other MCU.
- **Mission 2:** Run DC motors with Bluetooth



You must submit

- LAB Report (*.md & *.pdf)
- Zip source files(main*.c, ecRCC.h, ecGPIO.h, ecSysTick.c etc...).
- Only the source files. Do not submit project files

Requirement

Hardware

- MCU
 - NUCLEO-F411RE
- Actuator/Sensor/Others:
 - DC motor, DC motor driver(L9110s),
 - Bluetooth Module(HC-06),

Software

- Keil uVision, CMSIS, EC_HAL library

Problem 1: Create HAL library

Create HAL library

Declare and Define the following functions in your library. You must update your header files located in the directory `EC \lib\`.

ecUSART.h

```
/* Given to Students */

void UART2_init();

void USART_write(USART_TypeDef* USARTx, uint8_t* buffer, uint32_t nBytes);

void USART_delay(uint32_t us);


/* Exercise*/

void USART_begin(USART_TypeDef* USARTx, GPIO_TypeDef* GPIO_TX, int pinTX,
GPIO_TypeDef* GPIO_RX, int pinRX, int baud);

void USART_init(USART_TypeDef* USARTx, int baud);
// default pins.

uint8_t USART_getc(USART_TypeDef * USARTx);

uint32_t is_USART_RXNE(USART_TypeDef * USARTx);
```

Problem 2: MCU1 - MCU2 via UART RS-232

Procedure

1. Create a new project under the directory `\repos\EC\LAB\LAB_USART_LED`

- The project name is "**LAB_USART_LED**".
- Create a new source files named as "**LAB_USART_LED.c**"

You MUST write your name on the source file inside the comment section.

2. Include your updated library in `\repos\EC\lib\` to your project.

- **ecGPIO.h, ecGPIO.c**
- **ecRCC.h, ecRCC.c**
- **ecUART.h, ecUART.c**

3. Connect each MCU to each PC with USART 2 via USB cable (ST-Link)

4. Connect MCU1 and MCU2 with USART 1

- connect RX/TX pins as
 - MCU1 TXD - MCU2 RXD
 - MCU1 RXD - MCU2 TXD
5. Check that sent message from a PC(Teraterm) is displayed on the other PC(Teraterm). Note that you have to press "Enter" to end the message.
6. Turn other MCU's LED(LD2) On/OFF by sending text of "**L0**" or "**L1**".

Configuration

Type	Port - Pin	Configuration
System Clock		PLL 84MHz
USART2 : USB cable (ST-Link)		No Parity, 8-bit Data, 1-bit Stop bit, 38400 baud-rate
USART1 : MCU1 - MCU2	TXD: PA9 RXD: PA10	No Parity, 8-bit Data, 1-bit Stop bit, 38400 baud-rate
Digital Out: LD2	PA5	

Code

Your code goes here: [ADD Code LINK such as github](#)

Explain your source code with necessary comments.

// YOUR MAIN CODE ONLY

// YOUR CODE

Result

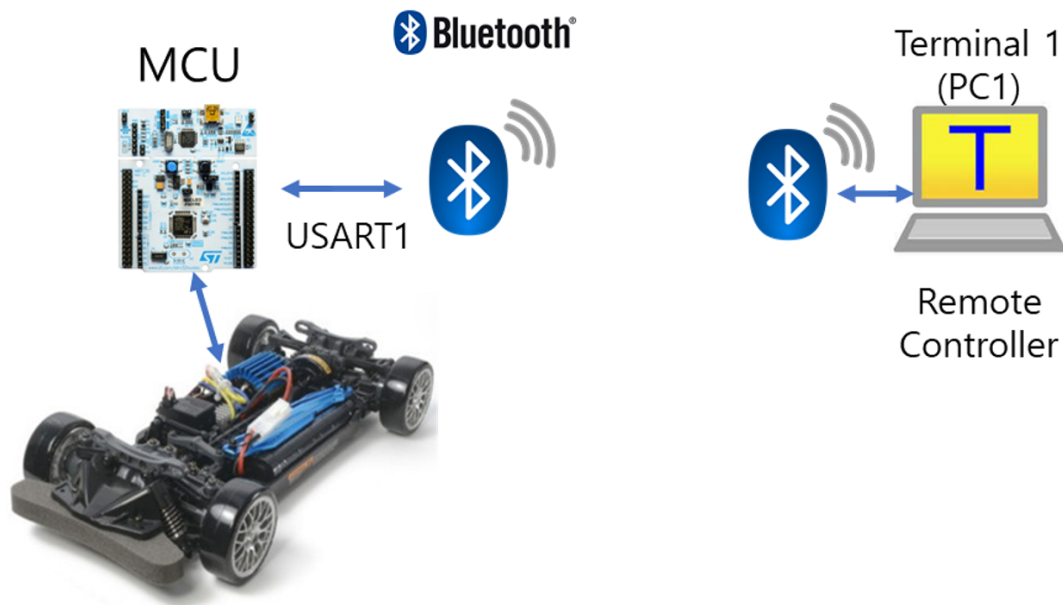
Experiment images and results

Show experiment images /results

Add [demo video link](#)

Problem 3: MCU - DC Motor via Bluetooth

Bluetooth



image

Search for the bluetooth module specification sheet (HC-06) and study the pin configurations. The default PIN number is 1234.

Example of connecting to USART1

Bluetooth Module (HC-06)	STM32F411RE
RxD	PA_9(UART1_TX)
TxD	PA_10(UART1_RX)
GND	GND
VCC	5V

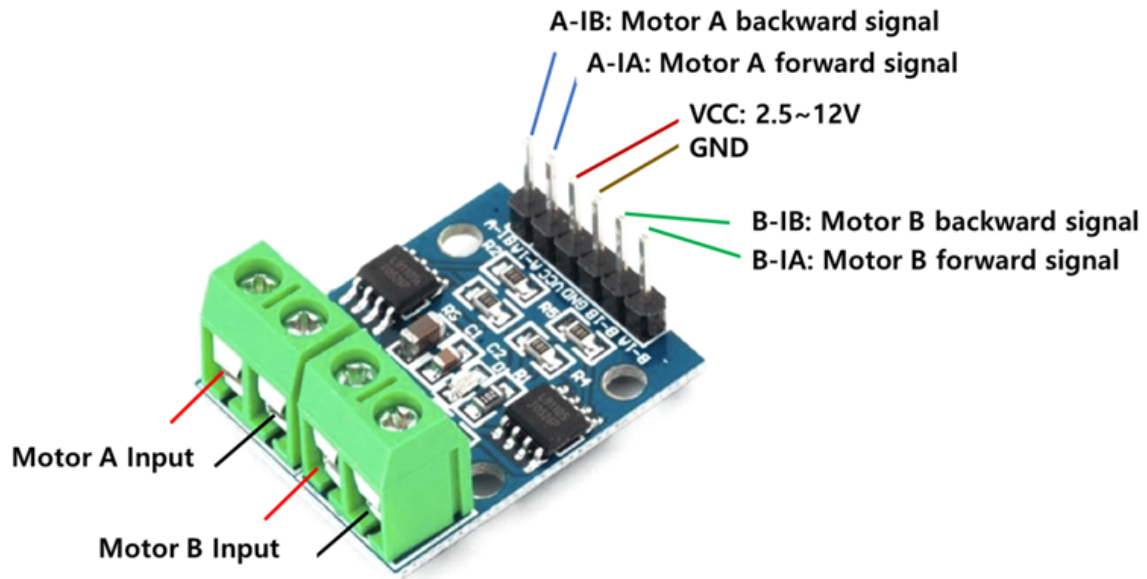
image

DC Motor Driver

Connect DC motor driver(L9110s) module pins to MCU as shown below.

DO NOT use MCU's VCC to motor driver. You should use external voltage source.

- A- IA: PWM pin (0~100% duty) for Motor A
- A- IB: Direction Pin (Digital Out H or L) for Motor B



image

Procedure

1. Create a new project under the directory ``repos\EC\LAB\LAB_USART_Bluetooth

- The project name is "**LAB_USART_Bluetooth**".
- Create a new source files named as "**LAB_USART_Bluetooth.c**"

You MUST write your name on the source file inside the comment section.

2. Include your updated library in `\repos\EC\lib\` to your project.

- **ecGPIO.h, ecGPIO.c**
- **ecRCC.h, ecRCC.c**
- **ecUART.h, ecUART.c**
- **ecTIM.h, ecTIM.c**

3. Connect the MCU to PC via Bluetooth. Use USART 1

- connect RX/TX pins as
 - MCU TXD - BLUE RXD
 - MCU RXD - BLUE TXD

1. 1.

Check the Bluetooth connection by turning MCU's LED(LD2) On/OFF by sending text of "**L0**" or "**L1**" from PC.

2. 2.

Run 2 DC motors(Left-wheel, Right-wheel) to steer.

- Turn Left: MotorA / MotorB = (50 / 80%) duty
- Turn Right: MotorA / MotorB = (80 / 50%) duty
-

Go straight: MotorA / MotorB = (80 / 80 %) duty

o

STOP: MotorA / MotorB = (0 / 0 %) duty

You may use the key inputs as your preference for Left, Right, Straight.

- Ex) 'L', 'R', 'U' 'S'

Configuration

Type	Port - Pin	Configuration
System Clock		PLL 84MHz
USART1 : MCU - Bluetooth	TXD: PA9 RXD: PA10	No Parity, 8-bit Data, 1-bit Stop bit, 9600 baud-rate
Digital Out: LD2	PA5	
PWM (Motor A)	TIM2-Ch1	PWM period (2kHz~10kHz)
PWM (Motor B)	TIM2-Ch2	

Code

Your code goes here: [ADD Code LINK such as github](#)

Explain your source code with necessary comments.

```
// YOUR MAIN CODE ONLY
```

```
// YOUR CODE
```

Result

Experiment images and results

 Show experiment images /results

Add [demo video link](#)

Reference

Complete list of all references used (github, blog, paper, etc)

Troubleshooting

(Option) You can write Troubleshooting section