

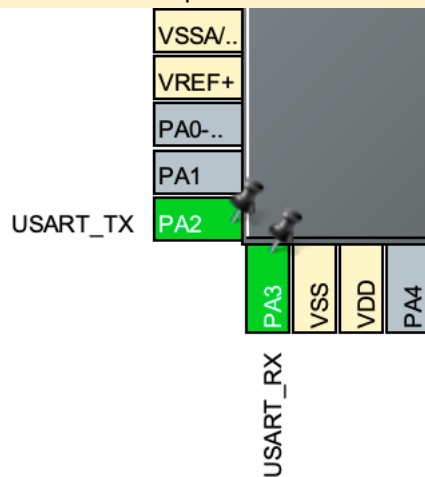
Mark	/11
------	-----

Team name:	A1		
Homework number:	HOMEWORK 04		
Due date:	15/10/24		
Contribution	NO	Partial	Full
Piombo			x
Fumagalli			x
Pierfederici			x
Zenoni			x
Ferraro			x
Notes:			

Project name	UART + LCD		
Not done	Partially done (major problems)	Partially done (minor problems)	Completed
			x

Part 1a:

First of all, we checked the pins used for UART communication (PA2 as transmitter and PA3 as receiver)



In “connectivity” -> “USART2” we configured the DMA transmitter, as shown in the Laboratory slides:

The screenshot shows the 'DMA Settings' tab in STM32CubeMX. A table lists the DMA configuration:

DMA Request	Stream	Direction	Priority
USART2_TX	DMA1 Stream 6	Memory To Peripheral	High

Below the table are 'Add' and 'Delete' buttons. The 'DMA Request Settings' section includes:

- Mode: Normal (dropdown)
- Increment Address: ☐ (Peripheral) and ☒ (Memory)
- Use Fifo: ☐
- Threshold: (dropdown)
- Data Width: Byte (dropdown)
- Burst Size: (dropdown)

Then we enabled the global interrupt of USART2 in order to perform the transmission:

The screenshot shows the 'NVIC Settings' tab. The 'NVIC Interrupt Table' is as follows:

	Enabled	Preemption Priority	Sub Priority
DMA1 stream6 global interrupt	<input checked="" type="checkbox"/>	0	0
USART2 global interrupt	<input checked="" type="checkbox"/>	0	0

In main.c file we used the proper HAL functions to send the string to the PC:

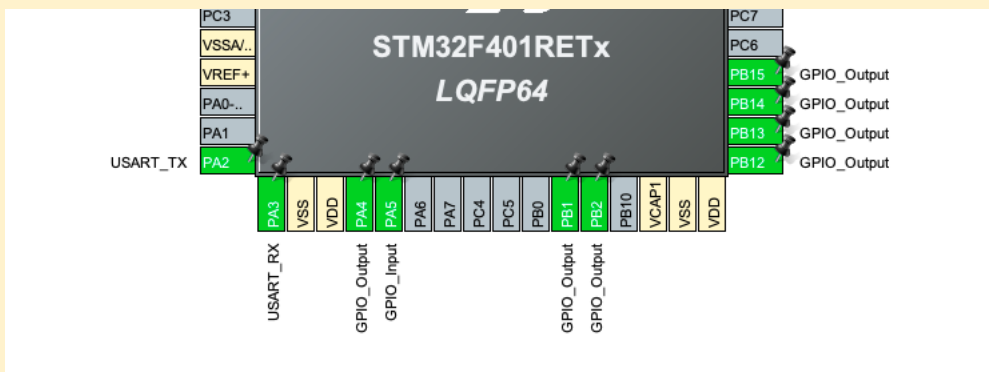
```
while (1)
{
    /* USER CODE END WHILE */

    /* USER CODE BEGIN 3 */

    HAL_UART_Transmit_DMA(&huart2, string, length);
    HAL_Delay(1000);
}
/* USER CODE END 3 */
}
```

Part 1b:

The configuration of the pin has been set to use the LCD, as shown in the slide:



Timer 2 has been used in interrupt mode to refresh the LCD every second:

TIM2 Mode and Configuration

Mode

Slave Mode **Disable**

Trigger Source **Disable**

Clock Source **Internal Clock**

Configuration

Reset Configuration

☒ Parameter Settings
 ☒ User Constants
 ☒ NVIC Settings
 ☒ DMA Settings

Configure the below parameters :

Search (Ctrl+F)

Counter Settings

Prescaler (PSC - 16 bits value)	8400-1
Counter Mode	Up
Counter Period (AutoReload Register - 32 bits value)	10000-1
Internal Clock Division (CKD)	No Division
auto-reload preload	Disable

☒ Parameter Settings
 ☒ User Constants
 ☒ NVIC Settings
 ☒ DMA Settings

NVIC Interrupt Table	Enabled	Preemption Priority	Sub Priority
TIM2 global interrupt	<input checked="" type="checkbox"/>	0	0

In main.c file we included the given library for the LCD (**#include "PMDB16_LCD.h"**).

We created an array of strings as a global variable to store the names.

In "int main" function we initialized the LCD, printed the first name and started the timer interrupt.

```

/* USER CODE BEGIN 2 */
lcd_initialize();
lcd_backlight_ON();

lcd_println(names[indexx], 1);

__HAL_TIM_SET_COUNTER(&htim2, 0);          // reset timer counter
__HAL_TIM_CLEAR_IT(&htim2, TIM_IT_UPDATE);  // clear interrupt request BEFORE enabling tim interrupt
HAL_TIM_Base_Start_IT(&htim2);              // start tim interrupt function
/* USER CODE END 2 */

```

The interrupt service routine changes the name printed on the display by increasing the index of the array.

```

void HAL_TIM_PeriodElapsedCallback (TIM_HandleTypeDef *htim) {
    if (htim == &htim2) {
        indexx++;
        lcd_println(names[indexx-1], 0);
        if (indexx == 5) {
            indexx = 0;
        };
        lcd_println(names[indexx], 1);
    };
}

```

