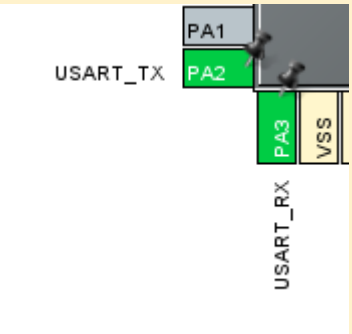
Mark	/11

Team name:	*A5*		
Homework number:	*HOMEWORK 04*		
Due date:	13/10/2024		
Contribution	NO	Partial	Full
Alessio Spineto			*x*
Riccardo Lamarca			*x*
Sofia Cecchetto			*x*
Annamaria De Togni			*x*
Emma Crespi			*x*
Notes: none			

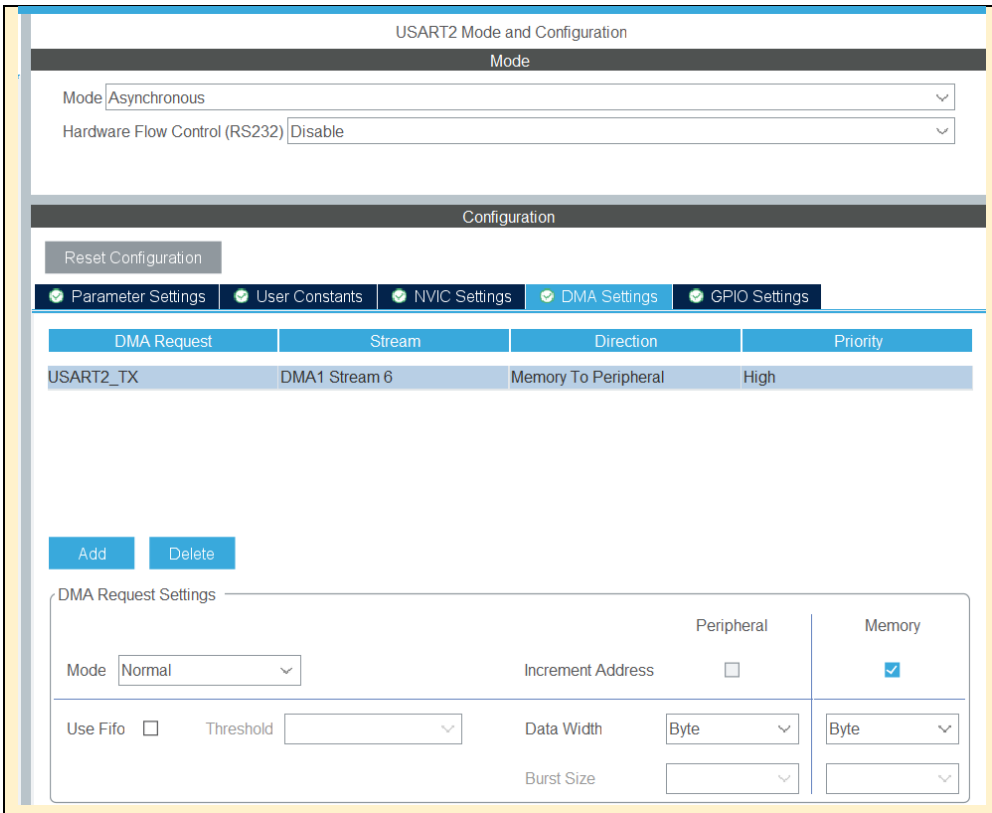
Project name	USART DMA + LCD		
Not done	Partially done (major problems)	Partially done (minor problems)	Completed

Part 1a: Complete the UART project with Direct Memory Access functions.

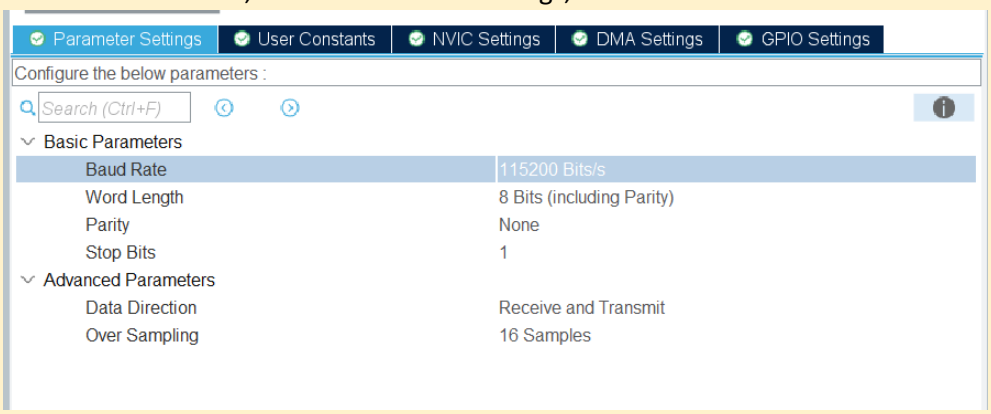
The two USART pins are enabled by default.



In the connectivity section of the GUI, we selected USART2, opening its 'Mode and Configuration' page. There, in the DMA settings, we added a new DMA request, selecting the transfer mode USART2_TX.



In the same window, under Parameter Settings, we selected a 115200 baud rate and 8 bit word length.



Inside the *main()* function, we first defined the string that we wanted to transmit, then we computed its length with *strlen()* function. In the *while(1)* loop we transmitted our string with HAL_UART_Transmit_DMA function. We then implemented the requested delay of one second between each transmission using the HAL_Delay function.

```
  /* Initialize all configured peripherals */
  MX_GPIO_Init();
  MX_DMA_Init();
  MX_USART2_UART_Init();
  /* USER CODE BEGIN 2 */
  char* frase= "Name Surname 2001\r\n";
  int length = strlen(frase);
  /* USER CODE END 2 */

  /* Infinite loop */
  /* USER CODE BEGIN WHILE */
  while (1)
  {
          HAL_UART_Transmit_DMA(&huart2, frase, length);
          HAL_Delay(1000);
    /* USER CODE END WHILE */

    /* USER CODE BEGIN 3 */
```

After connecting our pc to the Nucleo board, we can use a terminal emulator of our choice to see the output, by setting the same baud rate as the one set to transmit data. Then, we also selected the right COM port connected to the Nucleo board.

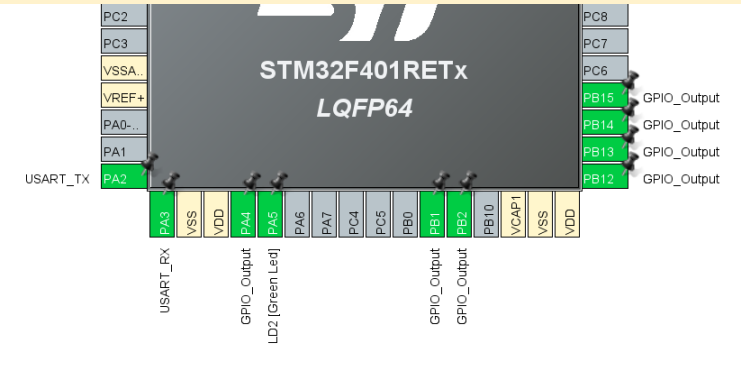
```
11:25:06.968 -> Name Surname 2001
11:25:07.947 -> Name Surname 2001
11:25:08.926 -> Name Surname 2001
11:25:09.950 -> Name Surname 2001
11:25:10.931 -> Name Surname 2001
11:25:11.913 -> Name Surname 2001
11:25:12.894 -> Name Surname 2001
```

Scorrimento automatico | Visualizza orario | Entrambi (NL & CR) | 115200 baud

```
120    RCC_ClkInitTypeDef RCC_ClkInitStruct = {0};
```

Part 1b: Write on the LCD the name of each member of your group, one per line, in alphabetical order.

We configured the PA4, PB1, PB2, PB12-15 pins as *GPIO_Output* and imported and included the "PMDB16_LCD" library.




```
17    */
18 /* USER CODE END Header */
19 /* Includes ------------------------------
20 #include "main.h"
21
22 /* Private includes ----------------------
23 /* USER CODE BEGIN Includes */
24 #include "PMDB16_LCD.h"
25 #include <string.h>
26 /* USER CODE END Includes */
27
```

We defined an array containing the names of the group members and an *insertion_sort* function to sort the names in alphabetical order.

```c
/* Private user code ---------------------------------------------------------*/
/* USER CODE BEGIN 0 */

int members_index;

 char *members[] = {"Riccardo","Alessio", "Sofia", "Anna", "Emma"};

 int length = sizeof(members)/sizeof(char*);

void insertion_sort(char* arr[], int n)
{
    for (int i = 1; i < n; ++i) {
        char* key = arr[i];
        int j = i - 1;

        /* Move elements of arr[0..i-1], that are
           greater than key, to one position ahead
           of their current position */
        while (j >= 0 && strcmp(arr[j], key)>0) {
            arr[j + 1] = arr[j];
            j = j - 1;
        }
        arr[j + 1] = key;
    }
}

```

```c
int main(void)
{

  /* USER CODE BEGIN 1 */
    insertion_sort(members, length);
  /* USER CODE END 1 */

```

We initialized the LCD controller and its backlight and defined an array containing the names in the main.c before the while(1) loop.

```c
  /* USER CODE END SysInit */

  /* Initialize all configured peripherals */
  MX_GPIO_Init();
  MX_USART2_UART_Init();
  /* USER CODE BEGIN 2 */

  lcd_initialize();
  lcd_backlight_ON();

  /* USER CODE END 2 */

```

First, we manually wrote the first name in the second row. After that, at each iteration the names scroll in alphabetical order appearing first in row ‘1’ and then in row ‘0’. The elements of the *members* array are selected using the remainder of the integer division by 5, which does not require the counter to be reset.

```c
  /* Infinite loop */
  /* USER CODE BEGIN WHILE */

  int counter = 1;
  lcd_println(members[0], 1);
  HAL_Delay(1000);
  while (1)
  {

      lcd_clear();
      lcd_println(members[(counter + 1) % length], 1);
      lcd_println(members[(counter ) % length], 0);
      counter = (counter + 1)%length;
      HAL_Delay(1000);

    /* USER CODE END WHILE */

    /* USER CODE BEGIN 3 */
  }
```

Professor comments: