VIETNAM NATIONAL UNIVERSITY, HO CHI MINH CITY

HO CHI MINH CITY UNIVERSITY OF TECHNOLY

REPORT

LAB 5

Class: Microprocessors-Microcontrollers – CC01

Lecture: NGUYỄN THIÊN ÂN

No.	Full Name	ID Student
1.	Nguyễn Huy Tài	2110513

Ho Chi Minh City, December 10th 2024

Contents

1 Code functions of UART

```c
/* USER CODE BEGIN 0 */
#define MAX_LENGTH_BUFFER 30
uint8_t temp = 0;
uint8_t buffer[MAX_LENGTH_BUFFER];
uint8_t buffer_index = 0;
uint8_t flag_buff = 0;
uint32_t adcValue = 0;
char str[50];


void HAL_UART_RxCpltCallback(UART_HandleTypeDef *huart)
{
  if (huart->Instance == USART2)
  {
    if (temp == '\r' || temp == '\n'  || buffer_index >= MAX_LENGTH_BUFFER)
    {
      // buffer[buffer_index] = '\r';
      // buffer[buffer_index + 1] = '\n';
      buffer_index = 0;
    }
    else
    {
      buffer[buffer_index] = temp;
      buffer_index++;
    }
    HAL_UART_Receive_IT(&huart2, &temp, 1);
    HAL_UART_Transmit(&huart2, &temp, 1, 100);
    flag_buff = 1;
  }
}

uint8_t command_flag = 0;
int last_response = 0;
uint8_t resend_flag = 1;

void command_parser_fsm()
{
  if(strcmp((char *)buffer, "!RST#", 5) == 0)
  {
    command_flag = 1;

    HAL_ADC_Start(&hadc1);
    HAL_ADC_PollForConversion(&hadc1, HAL_MAX_DELAY);
    adcValue = HAL_ADC_GetValue(&hadc1);
    HAL_ADC_Stop(&hadc1);

    memset(buffer, 0, MAX_LENGTH_BUFFER);
    buffer_index = 0;
    flag_buff = 0;
  }

  else if (strcmp((char *)buffer, "!OK#", 4) == 0)
  {
    const char *response = "end\r\n";
    HAL_UART_Transmit(&huart2, (uint8_t *)response, strlen(response), 1000);
    command_flag = 0;

    memset(buffer, 0, MAX_LENGTH_BUFFER);
    buffer_index = 0;
    flag_buff = 0;
  }
}


void uart_communicate_fsm()
{
  if(command_flag == 1)
  {
    if(resend_flag == 1)
    {
      char str[50];
      HAL_UART_Transmit(&huart2, (char *)str, sprintf(str, "!ADC:%d#\r\n", adcValue), 1000);
      resend_flag = 0;
    }
  }
}

/* USER CODE END 0 */
```

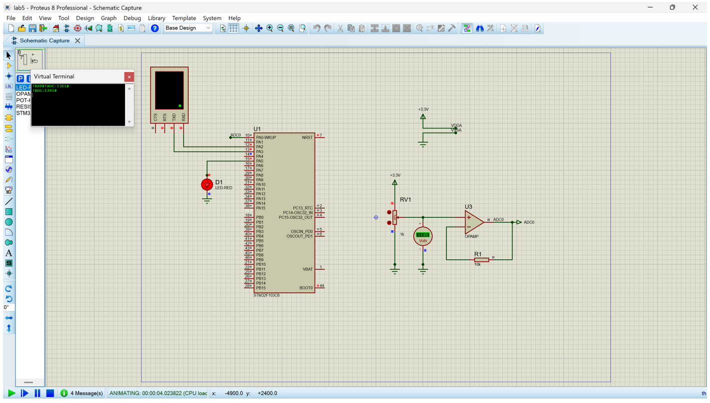
Program 1: Functions of UART communicate

2 Code main function

```
int main(void)
{
    /*
    *
    */

    /* Initialize all configured peripherals */
    MX_GPIO_Init();
    MX_ADC1_Init();
    MX_USART2_UART_Init();
    /* USER CODE BEGIN 2 */

    HAL_UART_Receive_IT(&huart2, &temp, 1);
    /* USER CODE END 2 */

    /* Infinite loop */
    /* USER CODE BEGIN WHILE */
    while (1)
    {

        if (flag_buff == 1)
        {
          command_parser_fsm();
          flag_buff = 0;
        }

        uart_communicate_fsm();

        if(HAL_GetTick() - last_response >= 3000)
        {
          resend_flag = 1;
          last_response = HAL_GetTick();

          memset(buffer, 0, MAX_LENGTH_BUFFER);
          buffer_index = 0;
          flag_buff = 0;

          HAL_GPIO_TogglePin(LED_RED_GPIO_Port, LED_RED_Pin);
        }
        /* USER CODE END WHILE */

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

Program 2: function of main

3 Schematic

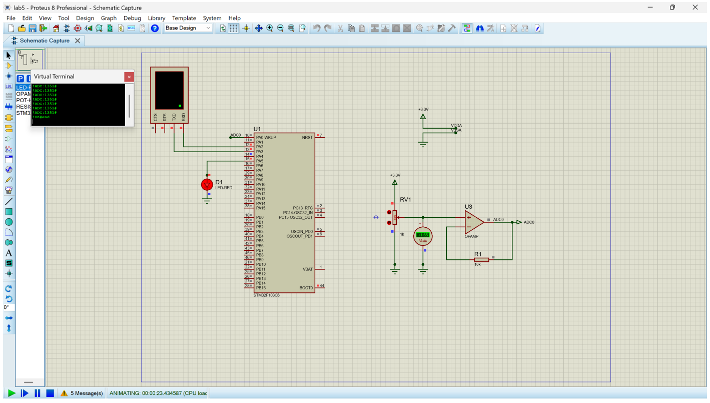


lab5 - Proteus 8 Professional - Schematic Capture
File Edit View Tool Design Graph Debug Library Template System Help
Base Design
Schematic Capture
Virtual Terminal
LED-R
OPAM
POT-H
RESIS
STM3
U1
D1
LED-RED
RV1
U3
R1
ADC0
+3.3V
VDDA
VSSA
STM32F103C6

```
!RST#!ADC:1351#
!ADC:1351#
```

4 Message(s) ANIMATING: 00:00:04.023822 (CPU loac x: -4900.0 y: +2400.0

Figure 1: *The system after typing !RST#*



lab5 - Proteus 8 Professional - Schematic Capture
File Edit View Tool Design Graph Debug Library Template System Help
Base Design
Schematic Capture
Virtual Terminal
LED-R
OPAM
POT-H
RESIS
STM3
U1
D1
LED-RED
RV1
U3
R1
ADC0
+3.3V
VDDA
VSSA
STM32F103C6

```
!ADC:1351#
!ADC:1351#
!ADC:1351#
!ADC:1351#
!ADC:1351#
!ADC:1351#
!ADC:1351#
!OK#end
```

5 Message(s) ANIMATING: 00:00:23.434587 (CPU loac

Figure 2: *The system after typing !OK#*

4 Link github

You can find the source code on my GitHub repository: **My GitHub Repository**.