VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS
ELEKTRONIKOS FAKULTETAS
ELEKTRONINIŲ SISTEMŲ KATEDRA

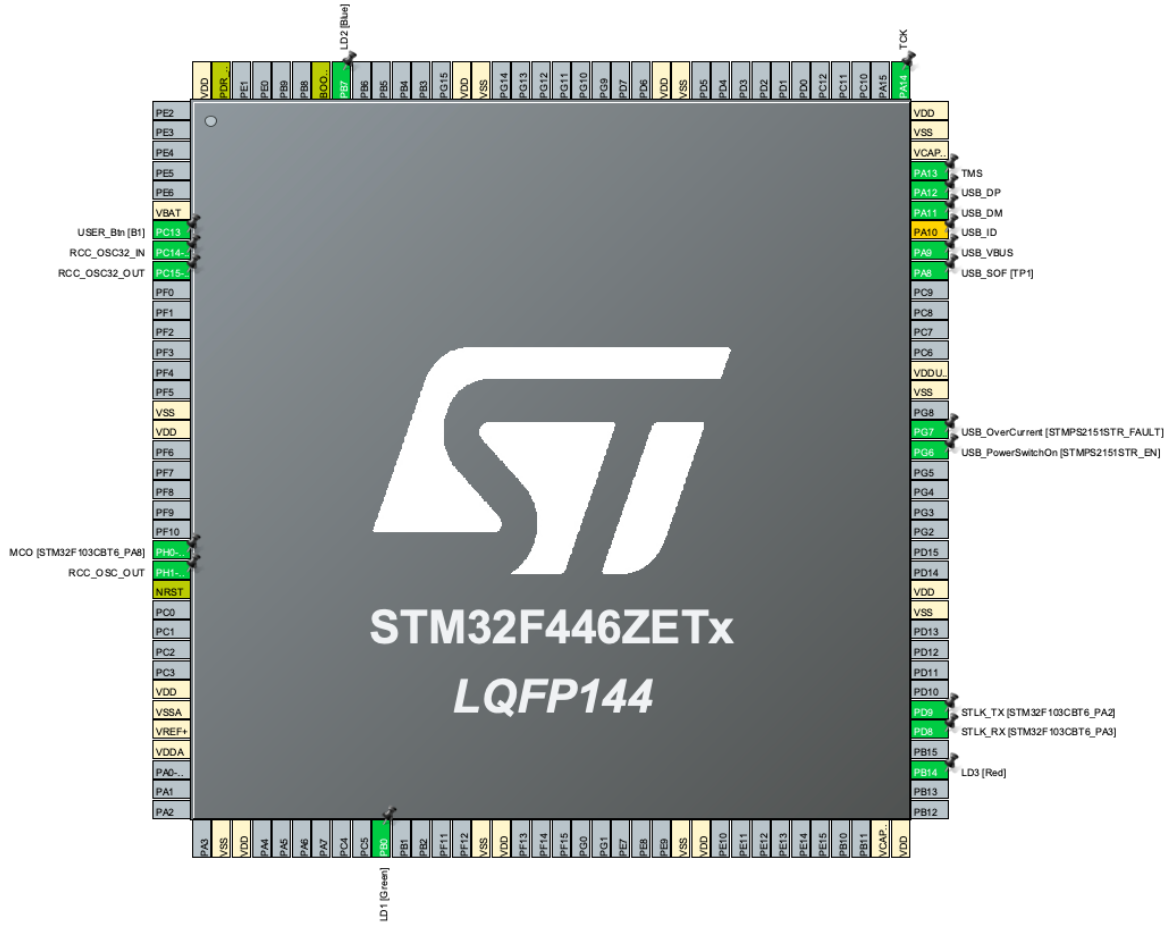
LABORATORINIS DARBAS 1

Įterptinių sistemų inžinerija

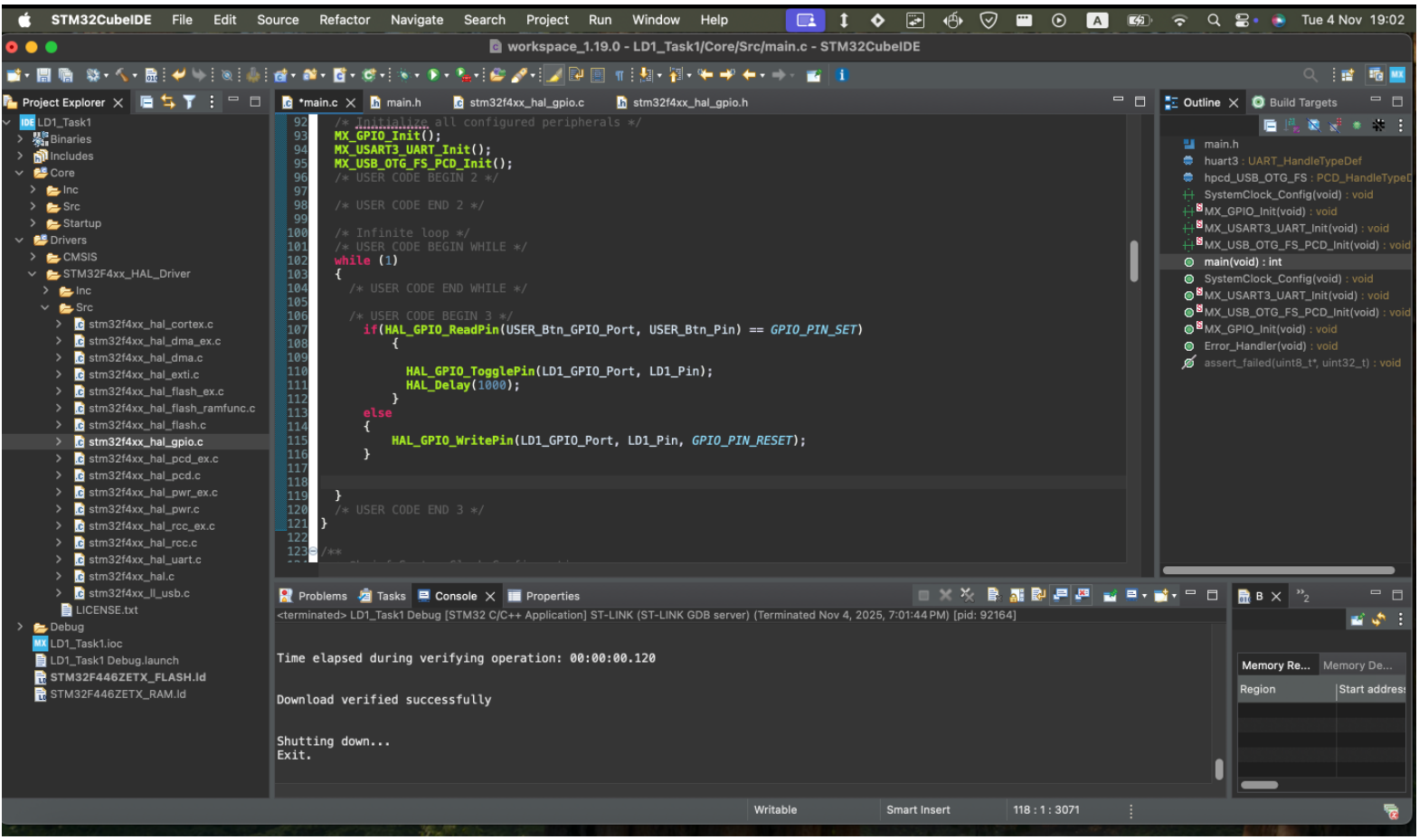
Atliko: EKSfm-24 gr. Ignas Malinauskas
Tikrino: dr. Eldar Šabanovič

VILNIUS 2025

1. 1 laboratorinio darbo mikrovaldiklio GPIO kaiščių konfigūracija

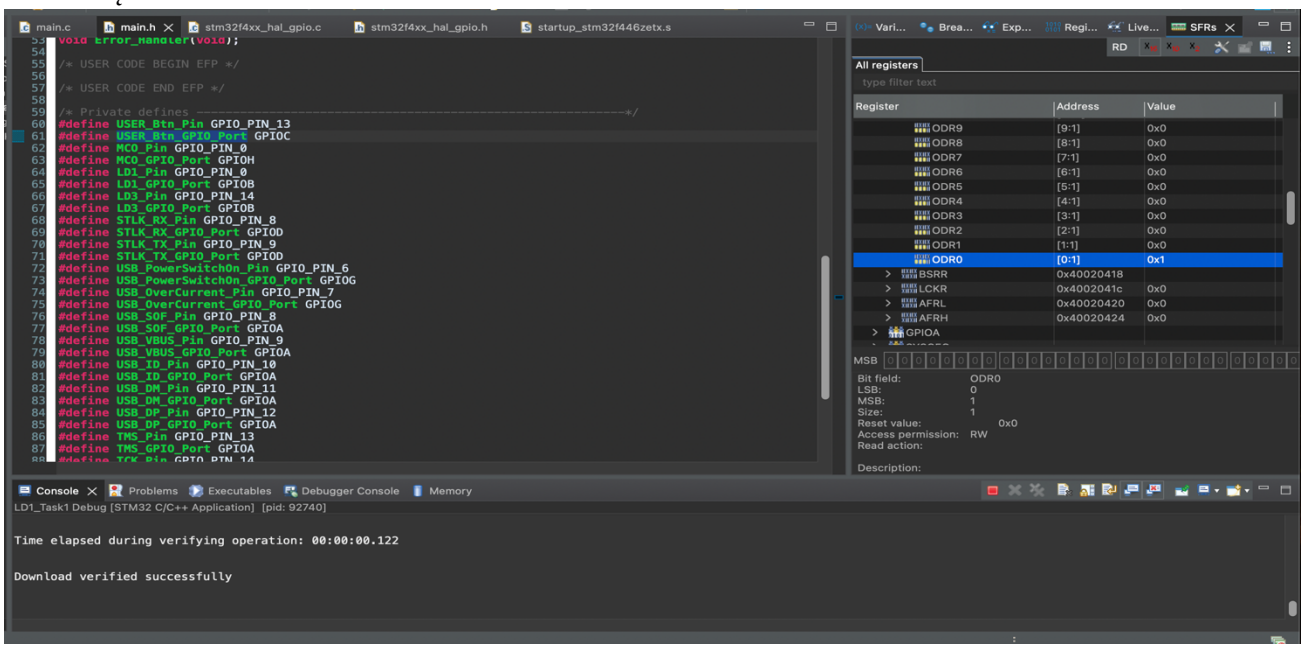


2. LED 1 mirksėjimo kodas



3. LED GPIO vertė keičiama per derinimą

Šioje vietoje išvestis pakeičiama per ODR (Output Data Register) susiradus LED1 atitinkantį bitą

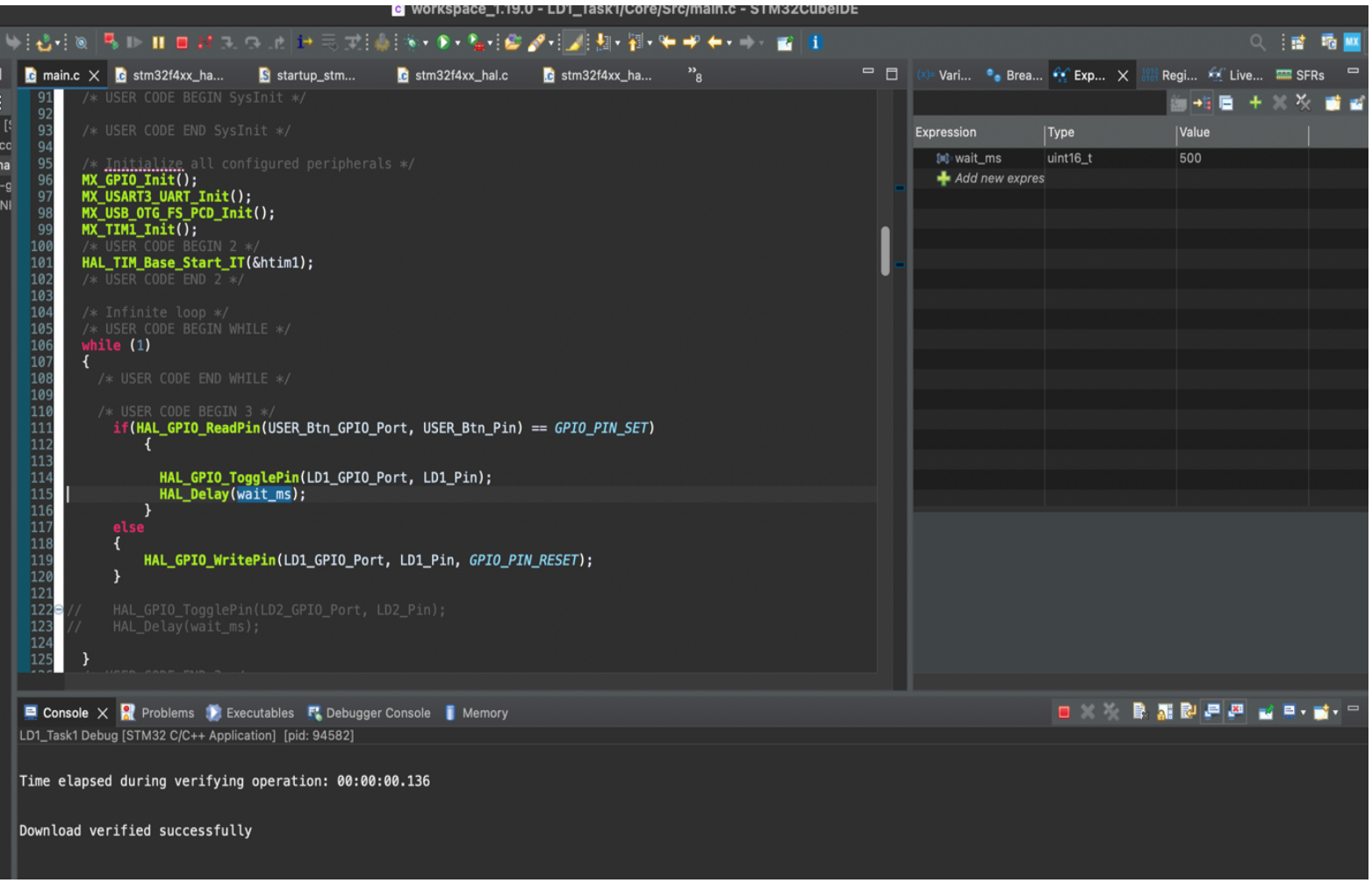


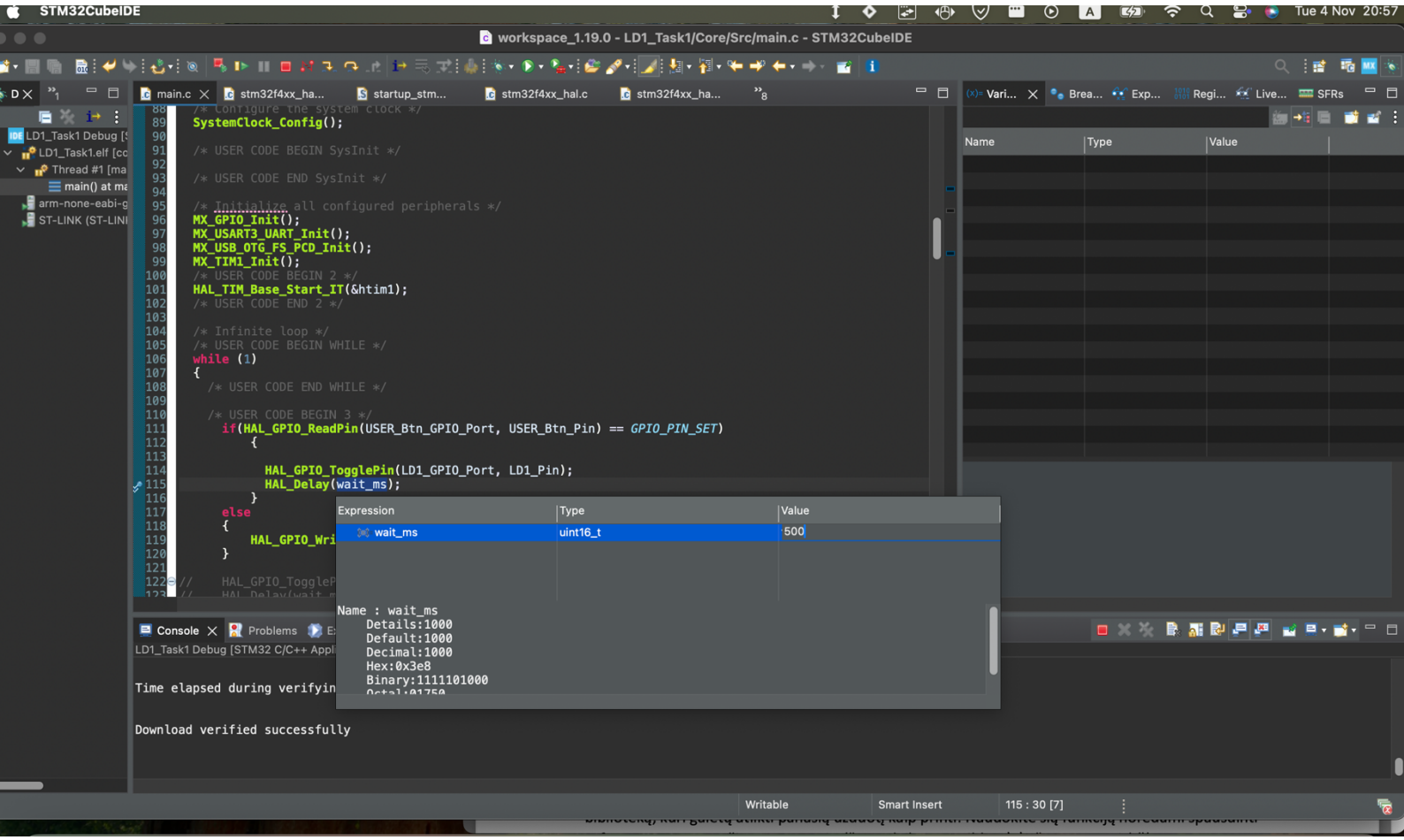
4. 2 LED mirksėjimas nustatytu dažniu pagal wait_ms kintamąjį

```
    uint16_t wait_ms = 500;
```

```
  while (1)
  {
      HAL_GPIO_TogglePin(LD2_GPIO_Port, LD2_Pin);
      HAL_Delay(wait_ms);
}
```

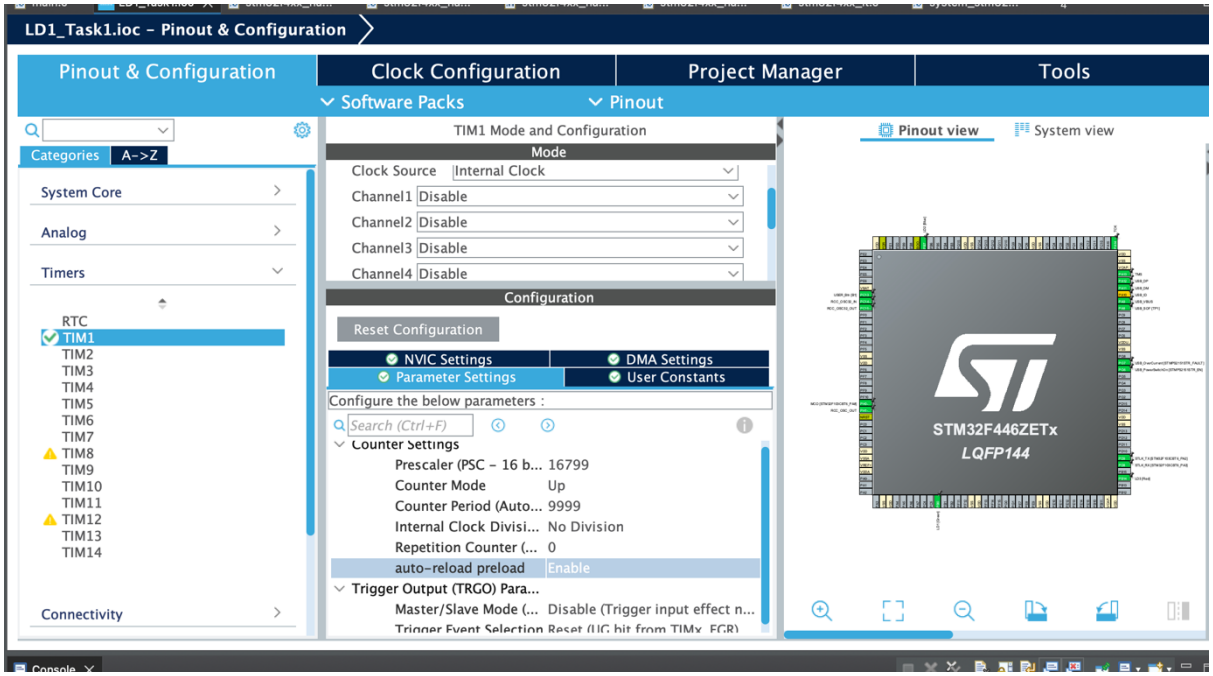
5. Wait_ms kintamojo keitimas derinimo metu





6. Laikmačio ir pertraukties panaudojimas LED mirksėjimui

6.1. CubeMX nustatymuose sukonfigūruojamas vienos sekundės laikmatis



6.2. Pertraukties kodas, įvykdomas kai sutiksi laikmatis – aktyvuojamas handleris ir nukreipiama į mirksėjimo funkciją

```
7. void TIM1_UP_TIM10_IRQHandler(void)
8. {
9.   /* USER CODE BEGIN TIM1_UP_TIM10_IRQn 0 */
10.
11.    /* USER CODE END TIM1_UP_TIM10_IRQn 0 */
12.    HAL_TIM_IRQHandler(&htim1);
13.    /* USER CODE BEGIN TIM1_UP_TIM10_IRQn 1 */
14.
15.    /* USER CODE END TIM1_UP_TIM10_IRQn 1 */
16.  }
```

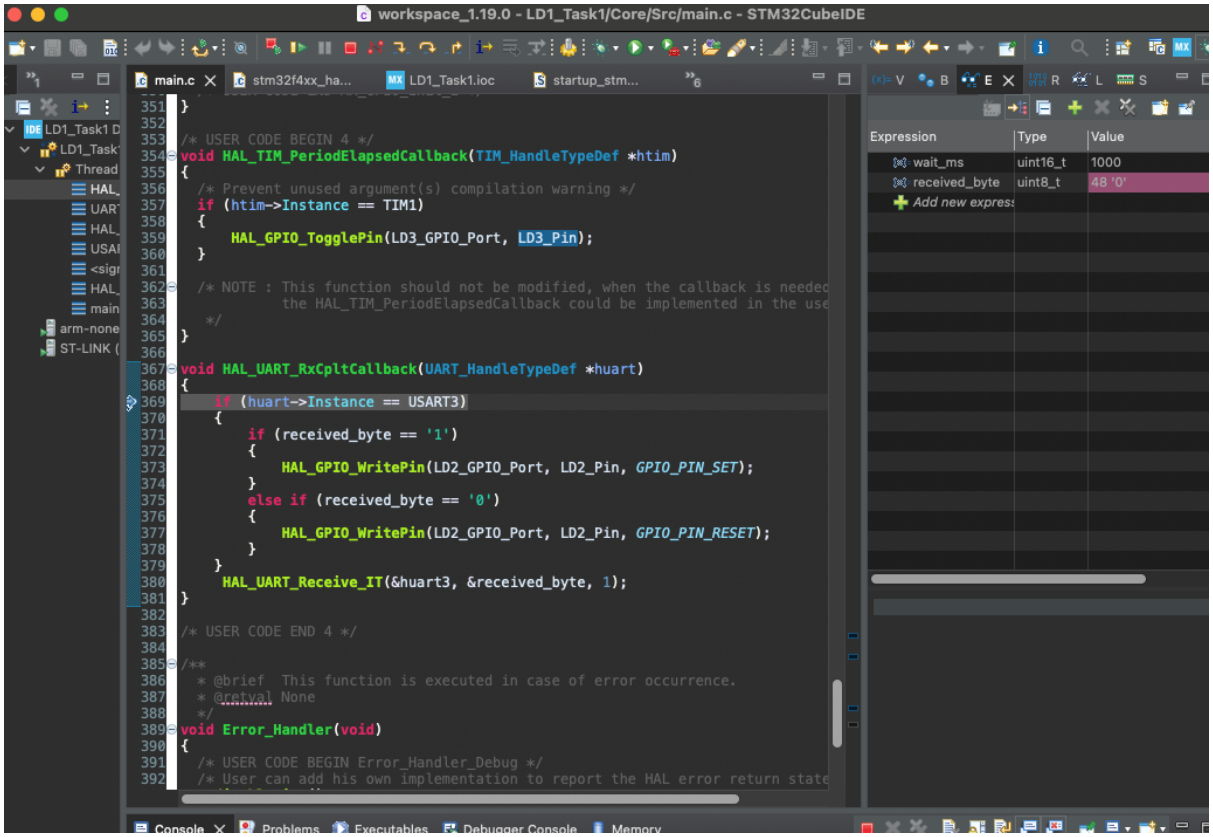
```
void HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef *htim)
{
  /* Prevent unused argument(s) compilation warning */
  if (htim->Instance == TIM1)
  {
      HAL_GPIO_TogglePin(LD3_GPIO_Port, LD3_Pin);
  }

  /* NOTE : This function should not be modified, when the
callback is needed,
           the HAL_TIM_PeriodElapsedCallback could be
implemented in the user file
   */
}
```

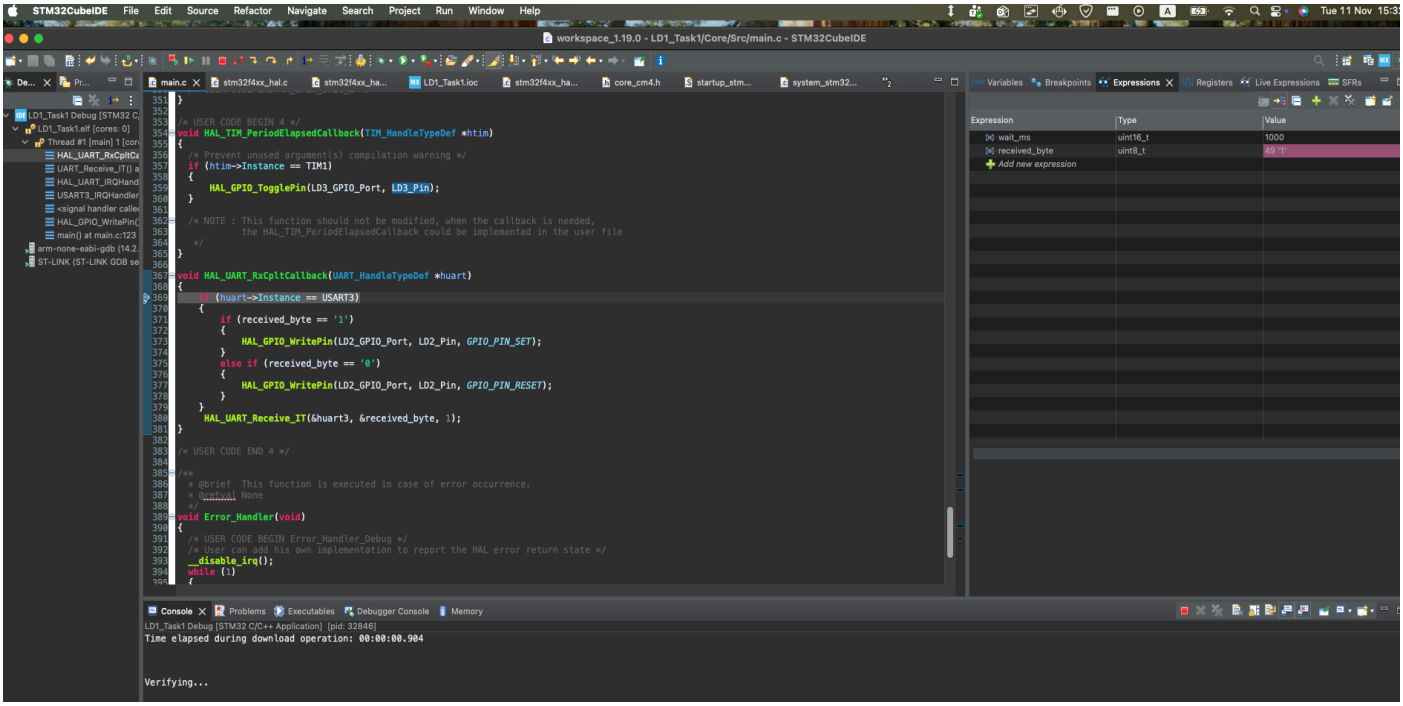
8. UART su pertrauktim panaudojimas

Kodo gabaliukas, kuriame naudojamas UART teksto spausdinimui į terminalą.

```
uint32_t last_print = 0;
while (1)
{
  /* USER CODE END WHILE */

  /* USER CODE BEGIN 3 */


    if(HAL_GPIO_ReadPin(USER_Btn_GPIO_Port, USER_Btn_Pin) == GPIO_PIN_SET)
        {

          HAL_GPIO_TogglePin(LD1_GPIO_Port, LD1_Pin);
          HAL_Delay(wait_ms);
          HAL_UART_Transmit(&huart3, "Hello\r\n", 7, 1000);
        }
```

Kodo fragmentas, kuriame aprašyta logika – kai iš serijinio prievado gaunamos vertės 1 arba 0, vykdoma pertrauktis ir atitinkamai įjungiama arba išjungiama LED2.



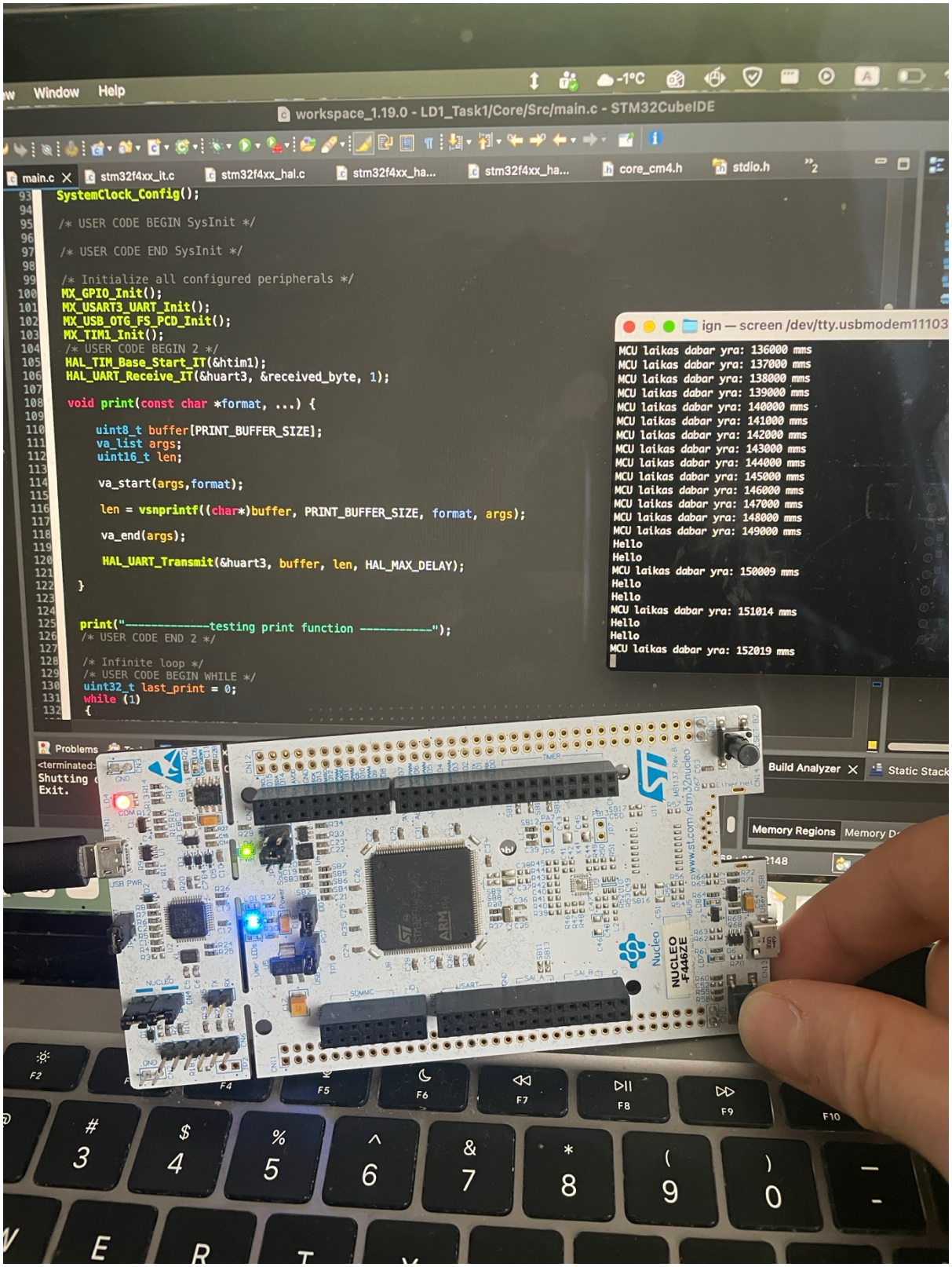


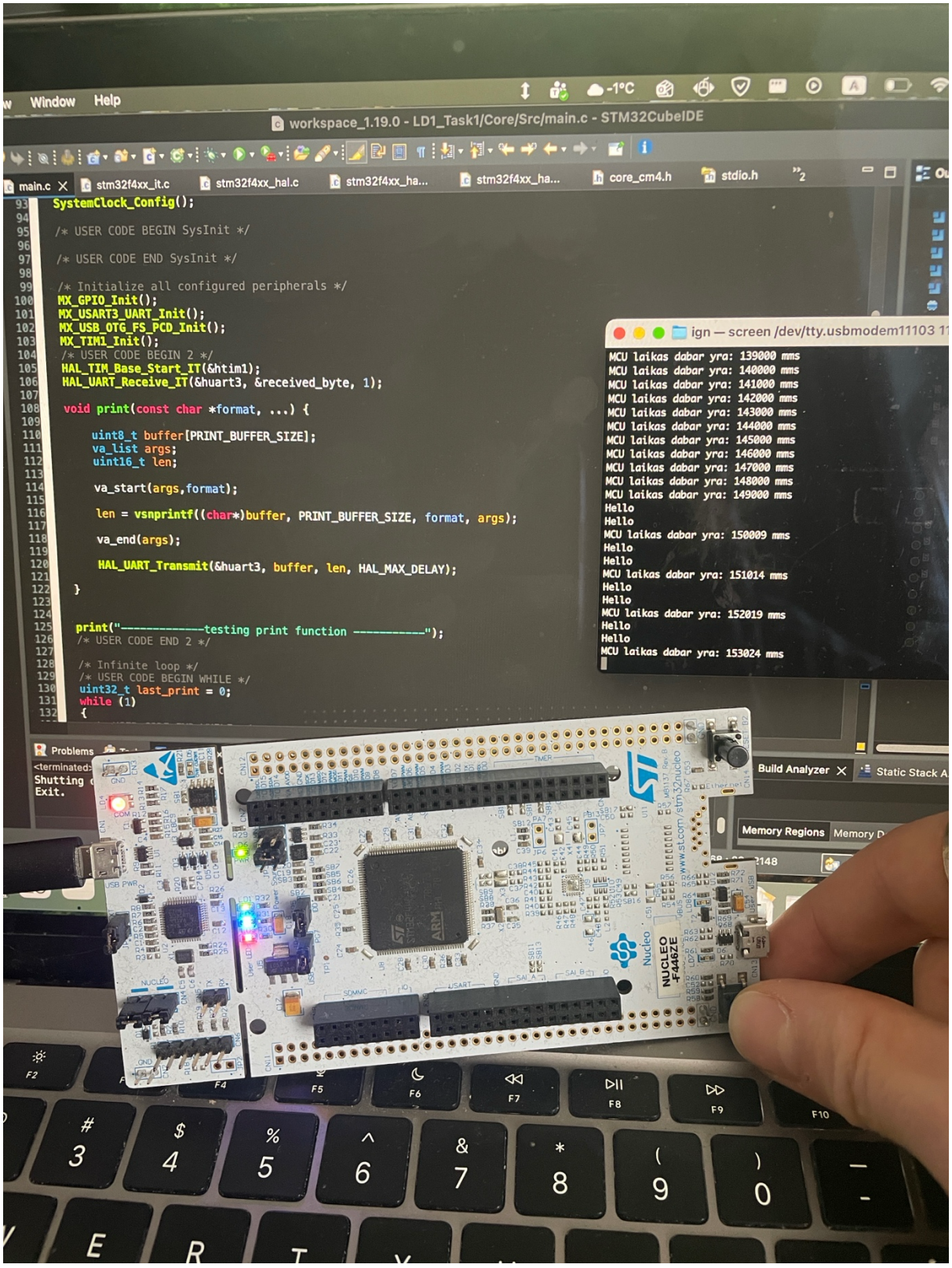
9. Print funkcijos kodas ir mikrovaldiklio bei serijinio terminalo nuotraukos

```
#define PRINT_BUFFER_SIZE 128
```

```
  void print(const char *format, ...) {

        uint8_t buffer[PRINT_BUFFER_SIZE];
        va_list args;
        uint16_t len;

        va_start(args,format);

        len = vsnprintf((char*)buffer, PRINT_BUFFER_SIZE,
format, args);

        va_end(args);

        HAL_UART_Transmit(&huart3, buffer, len, HAL_MAX_DELAY);

  }
  print("-------------testing print function -----------");
  /* USER CODE END 2 */

  /* Infinite loop */
  /* USER CODE BEGIN WHILE */
  uint32_t last_print = 0;
```

```
  while (1)
  {
        if (HAL_GetTick() - last_print >= 1000){
             last_print += 1000;
             print("MCU laikas dabar yra: %ld mms\r\n",
HAL_GetTick());
        }
```


workspace_1.19.0 - LD1_Task1/Core/Src/main.c - STM32CubeIDE
main.c
stm32f4xx_it.c
stm32f4xx_hal.c
SystemClock_Config();
/* USER CODE BEGIN SysInit */
/* USER CODE END SysInit */
/* Initialize all configured peripherals */
MX_GPIO_Init();
MX_USART3_UART_Init();
MX_USB_OTG_FS_PCD_Init();
MX_TIM1_Init();
/* USER CODE BEGIN 2 */
HAL_TIM_Base_Start_IT(&htim1);
HAL_UART_Receive_IT(&huart3, &received_byte, 1);
void print(const char *format, ...) {
uint8_t buffer[PRINT_BUFFER_SIZE];
va_list args;
uint16_t len;
va_start(args,format);
len = vsnprintf((char*)buffer, PRINT_BUFFER_SIZE, format, args);
va_end(args);
HAL_UART_Transmit(&huart3, buffer, len, HAL_MAX_DELAY);
}
print("------------testing print function ----------");
/* USER CODE END 2 */
/* Infinite loop */
/* USER CODE BEGIN WHILE */
uint32_t last_print = 0;
while (1)
{
ign — screen /dev/tty.usbmodem11103
MCU laikas dabar yra: 136000 mms
MCU laikas dabar yra: 137000 mms
MCU laikas dabar yra: 138000 mms
MCU laikas dabar yra: 139000 mms
MCU laikas dabar yra: 140000 mms
MCU laikas dabar yra: 141000 mms
MCU laikas dabar yra: 142000 mms
MCU laikas dabar yra: 143000 mms
MCU laikas dabar yra: 144000 mms
MCU laikas dabar yra: 145000 mms
MCU laikas dabar yra: 146000 mms
MCU laikas dabar yra: 147000 mms
MCU laikas dabar yra: 148000 mms
MCU laikas dabar yra: 149000 mms
Hello
Hello
MCU laikas dabar yra: 150009 mms
Hello
Hello
MCU laikas dabar yra: 151014 mms
Hello
Hello
MCU laikas dabar yra: 152019 mms
Problems
Build Analyzer
Static Stack
Memory Regions
NUCLEO F446ZE


Window Help
workspace_1.19.0 - LD1_Task1/Core/Src/main.c - STM32CubeIDE
main.c
stm32f4xx_it.c
stm32f4xx_hal.c
stm32f4xx_ha...
stm32f4xx_ha...
core_cm4.h
stdio.h
SystemClock_Config();
/* USER CODE BEGIN SysInit */
/* USER CODE END SysInit */
/* Initialize all configured peripherals */
MX_GPIO_Init();
MX_USART3_UART_Init();
MX_USB_OTG_FS_PCD_Init();
MX_TIM1_Init();
/* USER CODE BEGIN 2 */
HAL_TIM_Base_Start_IT(&htim1);
HAL_UART_Receive_IT(&huart3, &received_byte, 1);
void print(const char *format, ...) {
uint8_t buffer[PRINT_BUFFER_SIZE];
va_list args;
uint16_t len;
va_start(args,format);
len = vsnprintf((char*)buffer, PRINT_BUFFER_SIZE, format, args);
va_end(args);
HAL_UART_Transmit(&huart3, buffer, len, HAL_MAX_DELAY);
}
print("------------testing print function ----------");
/* USER CODE END 2 */
/* Infinite loop */
/* USER CODE BEGIN WHILE */
uint32_t last_print = 0;
while (1)
{
ign — screen /dev/tty.usbmodem11103
MCU laikas dabar yra: 139000 mms
MCU laikas dabar yra: 140000 mms
MCU laikas dabar yra: 141000 mms
MCU laikas dabar yra: 142000 mms
MCU laikas dabar yra: 143000 mms
MCU laikas dabar yra: 144000 mms
MCU laikas dabar yra: 145000 mms
MCU laikas dabar yra: 146000 mms
MCU laikas dabar yra: 147000 mms
MCU laikas dabar yra: 148000 mms
MCU laikas dabar yra: 149000 mms
Hello
Hello
MCU laikas dabar yra: 150009 mms
Hello
Hello
MCU laikas dabar yra: 151014 mms
Hello
Hello
MCU laikas dabar yra: 152019 mms
Hello
Hello
MCU laikas dabar yra: 153024 mms
Problems
Shutting
Exit.
Build Analyzer
Static Stack A
Memory Regions
Memory D
NUCLEO-F446ZE