VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

ELEKTRONIKOS FAKULTETAS

ELEKTRONINIŲ SISTEMŲ KATEDRA

LABORATORINIS DARBAS 2

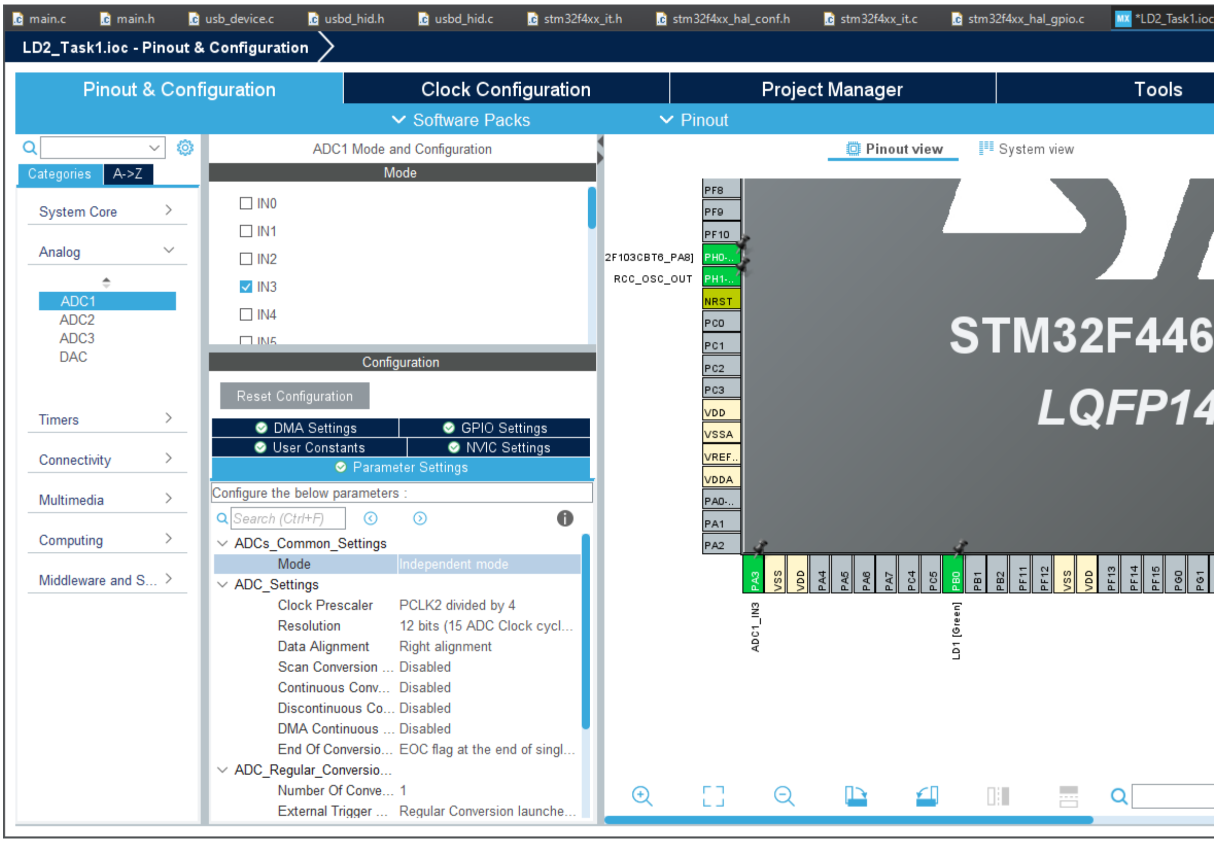
Įterptinių sistemų inžinerija

Atliko: EKSfm-24 gr. Ignas Malinauskas

Tikrino: dr. Eldar Šabanovič

VILNIUS 2025

1. LD_Tas1 projekto konfigūracija



LD2_Task1.ioc - Pinout & Configuration

ADC1 Mode and Configuration

Mode

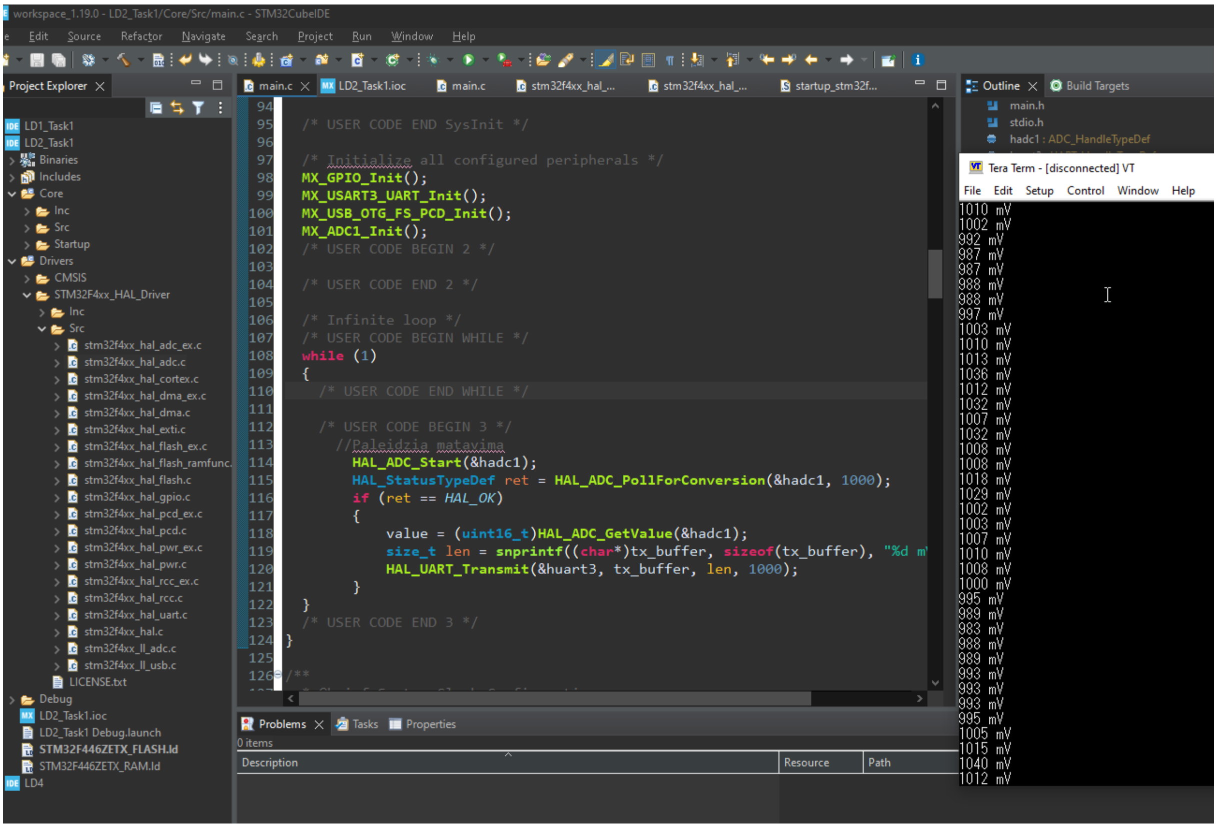
☐ IN0
☐ IN1
☐ IN2
☑ IN3
☐ IN4
☐ IN5

Configuration

Configure the below parameters :

```
ADCs_Common_Settings
    Mode                      Independent mode
ADC_Settings
    Clock Prescaler           PCLK2 divided by 4
    Resolution                12 bits (15 ADC Clock cycl...
    Data Alignment            Right alignment
    Scan Conversion ...       Disabled
    Continuous Conv...        Disabled
    Discontinuous Co...       Disabled
    DMA Continuous ...        Disabled
    End Of Conversio...       EOC flag at the end of singl...
ADC_Regular_Conversio...
    Number Of Conve...        1
    External Trigger ...      Regular Conversion launche...
```

2. ADC (A0) patikrinimas be potenciometro per Tera Term



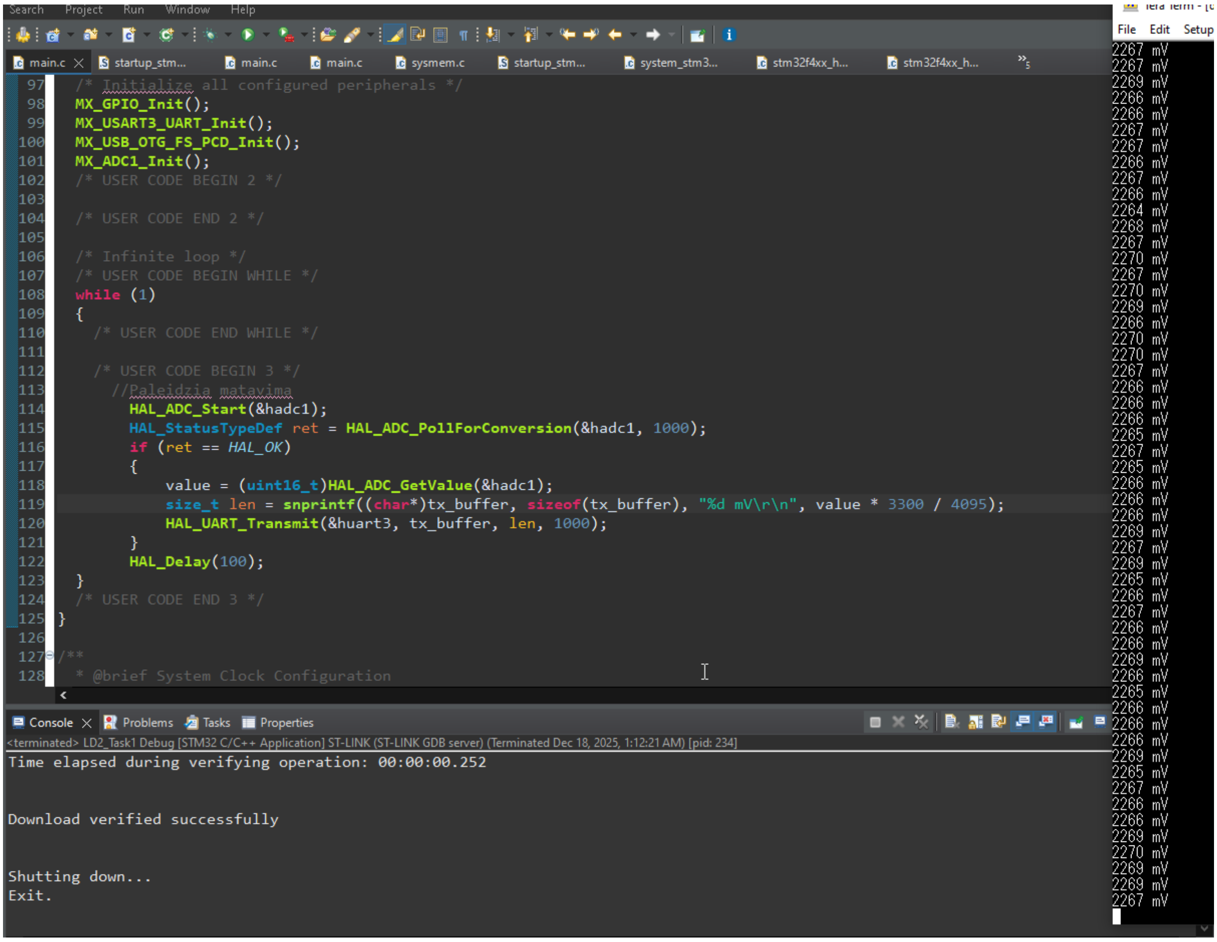
workspace_1.19.0 - LD2_Task1/Core/Src/main.c - STM32CubeIDE

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

  /* Initialize all configured peripherals */
  MX_GPIO_Init();
  MX_USART3_UART_Init();
  MX_USB_OTG_FS_PCD_Init();
  MX_ADC1_Init();
  /* USER CODE BEGIN 2 */

  /* USER CODE END 2 */

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

    /* USER CODE BEGIN 3 */
      //Paleidzia matavima
        HAL_ADC_Start(&hadc1);
        HAL_StatusTypeDef ret = HAL_ADC_PollForConversion(&hadc1, 1000);
        if (ret == HAL_OK)
        {
            value = (uint16_t)HAL_ADC_GetValue(&hadc1);
            size_t len = snprintf((char*)tx_buffer, sizeof(tx_buffer), "%d m'
            HAL_UART_Transmit(&huart3, tx_buffer, len, 1000);
        }
  }
  /* USER CODE END 3 */
}

/**
```

Tera Term - [disconnected] VT

```
1010 mV
1002 mV
992 mV
987 mV
987 mV
988 mV
988 mV
997 mV
1003 mV
1010 mV
1013 mV
1036 mV
1012 mV
1032 mV
1007 mV
1032 mV
1008 mV
1008 mV
1018 mV
1029 mV
1002 mV
1003 mV
1007 mV
1010 mV
1008 mV
1000 mV
995 mV
989 mV
983 mV
988 mV
989 mV
993 mV
993 mV
993 mV
995 mV
1005 mV
1015 mV
1040 mV
1012 mV
```

3. ADC nuskaitytos vertės iš potenciometro parodomos per UART ir multimetrą



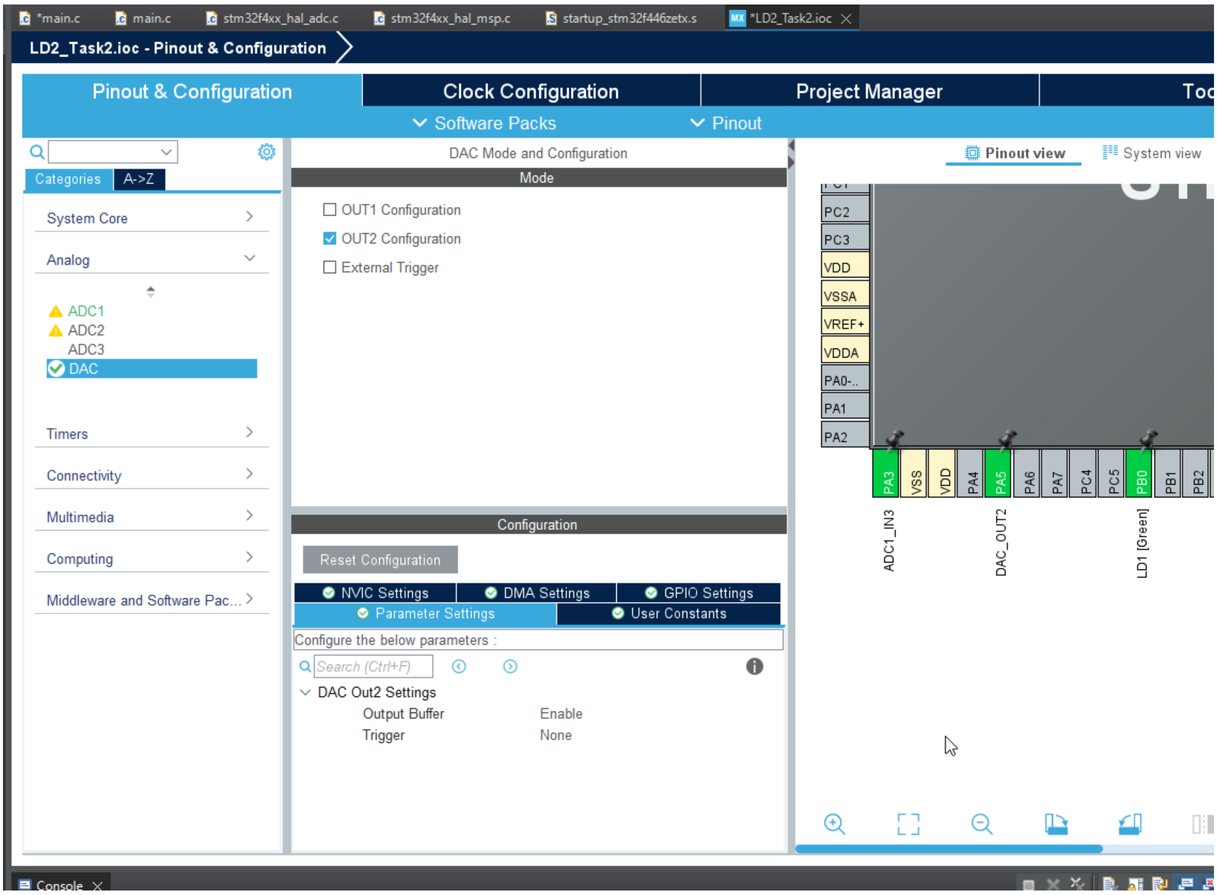
```c
  /* Initialize all configured peripherals */
  MX_GPIO_Init();
  MX_USART3_UART_Init();
  MX_USB_OTG_FS_PCD_Init();
  MX_ADC1_Init();
  /* USER CODE BEGIN 2 */

  /* USER CODE END 2 */

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

    /* USER CODE BEGIN 3 */
      //Paleidzia matavima
      HAL_ADC_Start(&hadc1);
      HAL_StatusTypeDef ret = HAL_ADC_PollForConversion(&hadc1, 1000);
      if (ret == HAL_OK)
      {
          value = (uint16_t)HAL_ADC_GetValue(&hadc1);
          size_t len = snprintf((char*)tx_buffer, sizeof(tx_buffer), "%d mV\r\n", value * 3300 / 4095);
          HAL_UART_Transmit(&huart3, tx_buffer, len, 1000);
      }
      HAL_Delay(100);
  }
  /* USER CODE END 3 */
}

/**
  * @brief System Clock Configuration
```

```
<terminated> LD2_Task1 Debug [STM32 C/C++ Application] ST-LINK (ST-LINK GDB server) (Terminated Dec 18, 2025, 1:12:21 AM) [pid: 234]
Time elapsed during verifying operation: 00:00:00.252


Download verified successfully


Shutting down...
Exit.
```

```
2267 mV
2267 mV
2269 mV
2266 mV
2266 mV
2267 mV
2267 mV
2266 mV
2267 mV
2266 mV
2264 mV
2268 mV
2267 mV
2270 mV
2267 mV
2270 mV
2269 mV
2266 mV
2270 mV
2270 mV
2267 mV
2266 mV
2266 mV
2266 mV
2265 mV
2267 mV
2265 mV
2266 mV
2266 mV
2266 mV
2269 mV
2267 mV
2269 mV
2265 mV
2266 mV
2267 mV
2266 mV
2266 mV
2269 mV
2266 mV
2265 mV
2266 mV
2266 mV
2266 mV
2269 mV
2265 mV
2267 mV
2266 mV
2266 mV
2269 mV
2270 mV
2269 mV
2269 mV
2267 mV
```

išmatuot su multimetru

4. LD2_Task2 projekto konfigūracija



DAC Mode and Configuration

Mode

☐ OUT1 Configuration

☑ OUT2 Configuration

☐ External Trigger

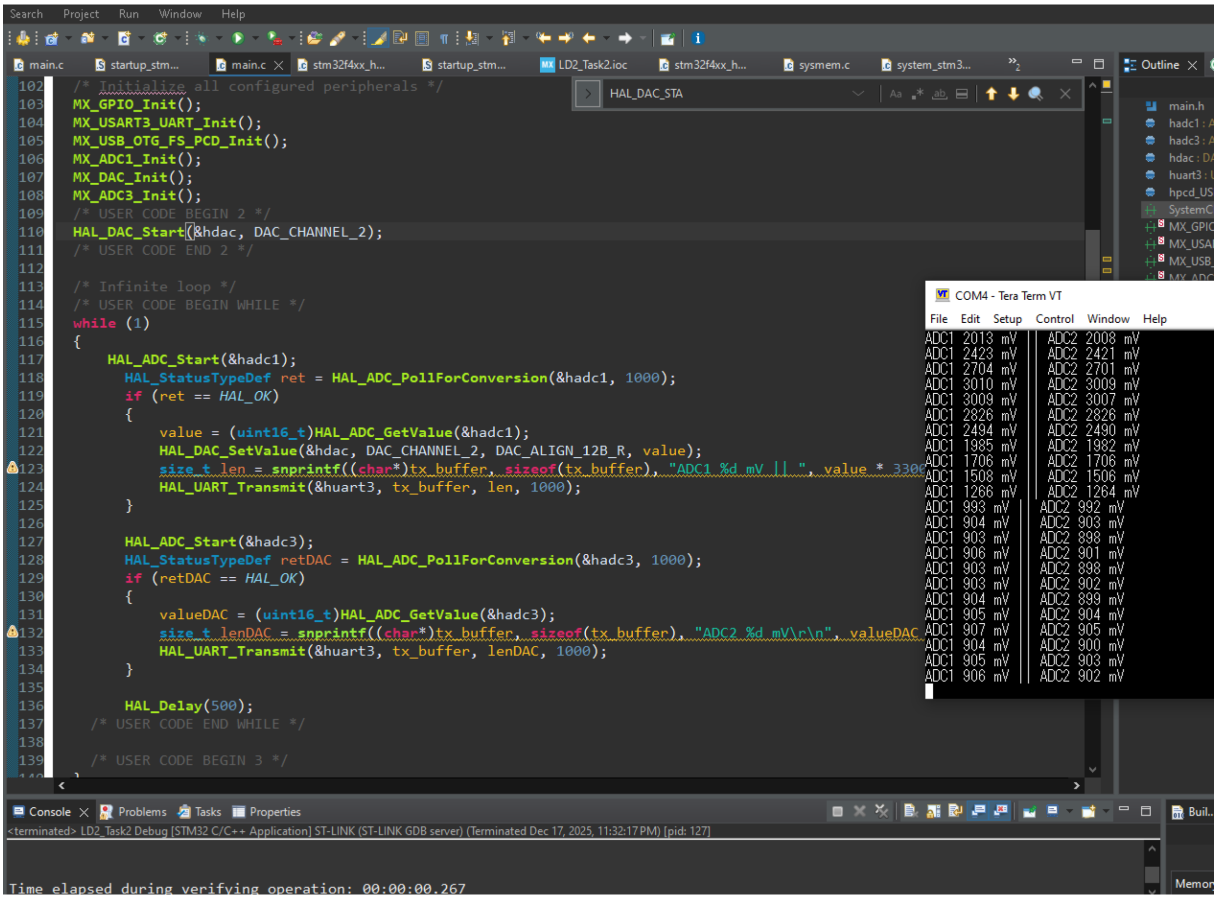
Configuration

Configure the below parameters :

DAC Out2 Settings

```
Output Buffer        Enable
Trigger              None
```

5. DAC (D13 (PA_5)) išvesties kodas ir patikrinimas su ADC (A3(PF3)

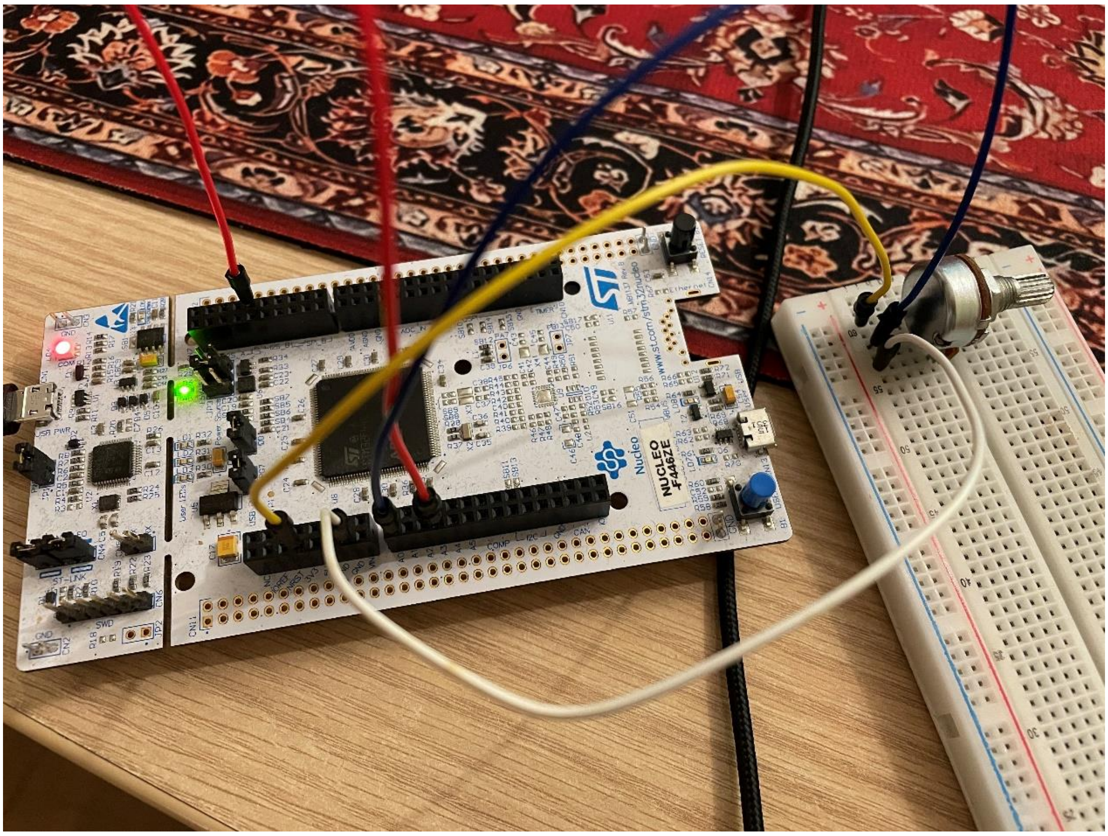


```c
/* Initialize all configured peripherals */
MX_GPIO_Init();
MX_USART3_UART_Init();
MX_USB_OTG_FS_PCD_Init();
MX_ADC1_Init();
MX_DAC_Init();
MX_ADC3_Init();
/* USER CODE BEGIN 2 */
HAL_DAC_Start(&hdac, DAC_CHANNEL_2);
/* USER CODE END 2 */

/* Infinite loop */
/* USER CODE BEGIN WHILE */
while (1)
{
    HAL_ADC_Start(&hadc1);
      HAL_StatusTypeDef ret = HAL_ADC_PollForConversion(&hadc1, 1000);
      if (ret == HAL_OK)
      {
          value = (uint16_t)HAL_ADC_GetValue(&hadc1);
          HAL_DAC_SetValue(&hdac, DAC_CHANNEL_2, DAC_ALIGN_12B_R, value);
          size_t len = snprintf((char*)tx_buffer, sizeof(tx_buffer), "ADC1 %d mV || ", value * 3300
          HAL_UART_Transmit(&huart3, tx_buffer, len, 1000);
      }

      HAL_ADC_Start(&hadc3);
      HAL_StatusTypeDef retDAC = HAL_ADC_PollForConversion(&hadc3, 1000);
      if (retDAC == HAL_OK)
      {
          valueDAC = (uint16_t)HAL_ADC_GetValue(&hadc3);
          size_t lenDAC = snprintf((char*)tx_buffer, sizeof(tx_buffer), "ADC2 %d mV\r\n", valueDAC
          HAL_UART_Transmit(&huart3, tx_buffer, lenDAC, 1000);
      }

      HAL_Delay(500);
  /* USER CODE END WHILE */

  /* USER CODE BEGIN 3 */
```

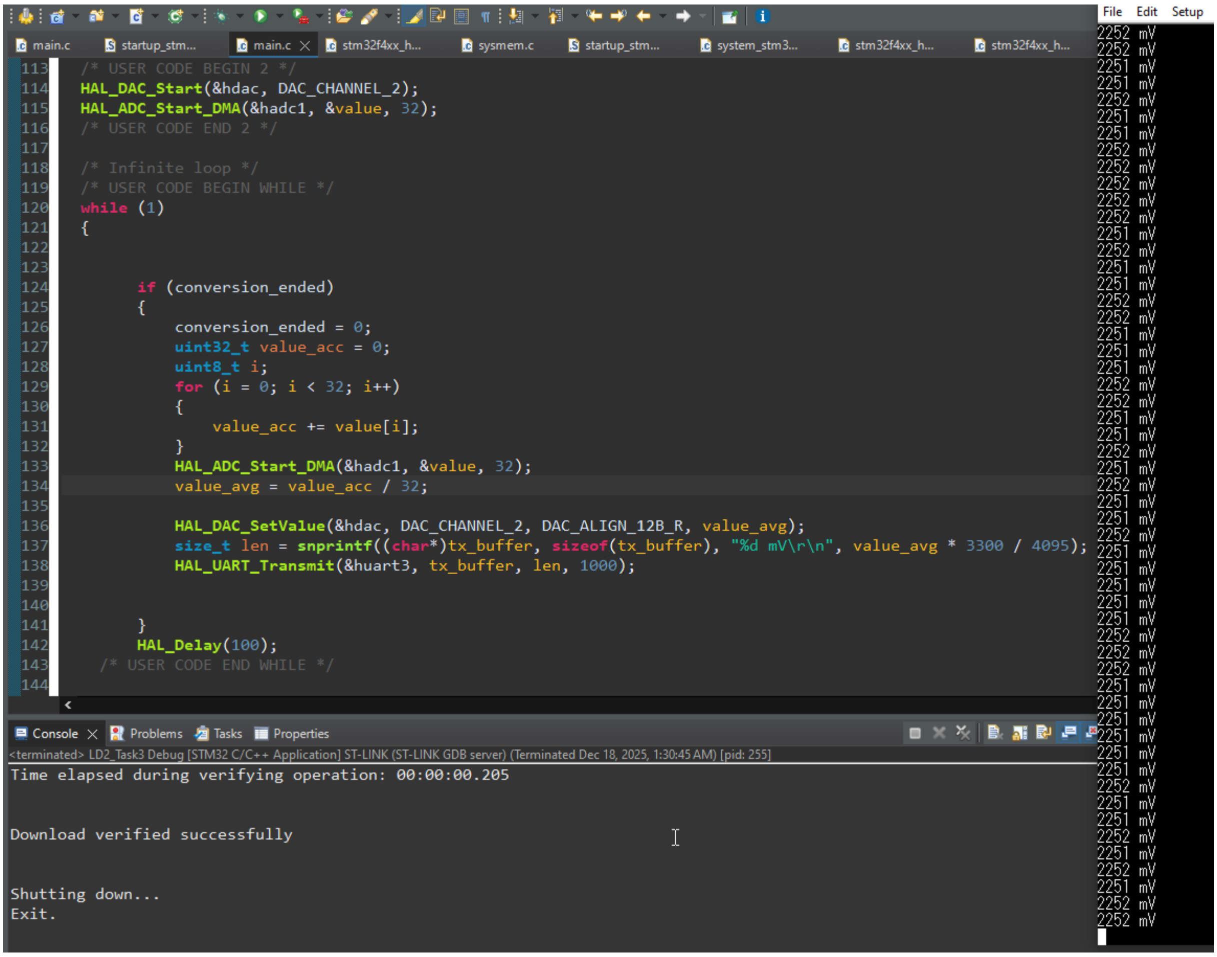
COM4 - Tera Term VT

```
ADC1 2013 mV || ADC2 2008 mV
ADC1 2423 mV || ADC2 2421 mV
ADC1 2704 mV || ADC2 2701 mV
ADC1 3010 mV || ADC2 3009 mV
ADC1 3009 mV || ADC2 3007 mV
ADC1 2826 mV || ADC2 2826 mV
ADC1 2494 mV || ADC2 2490 mV
ADC1 1985 mV || ADC2 1982 mV
ADC1 1706 mV || ADC2 1706 mV
ADC1 1508 mV || ADC2 1506 mV
ADC1 1266 mV || ADC2 1264 mV
ADC1 993 mV || ADC2 992 mV
ADC1 904 mV || ADC2 903 mV
ADC1 903 mV || ADC2 898 mV
ADC1 906 mV || ADC2 901 mV
ADC1 903 mV || ADC2 898 mV
ADC1 903 mV || ADC2 902 mV
ADC1 904 mV || ADC2 899 mV
ADC1 905 mV || ADC2 904 mV
ADC1 907 mV || ADC2 905 mV
ADC1 904 mV || ADC2 900 mV
ADC1 905 mV || ADC2 903 mV
ADC1 906 mV || ADC2 902 mV
```

```
<terminated> LD2_Task2 Debug [STM32 C/C++ Application] ST-LINK (ST-LINK GDB server) (Terminated Dec 17, 2025, 11:32:17 PM) [pid: 127]

Time elapsed during verifying operation: 00:00:00.267
```



6.) ADC DMA su pertrauktim kodas ir matavimai

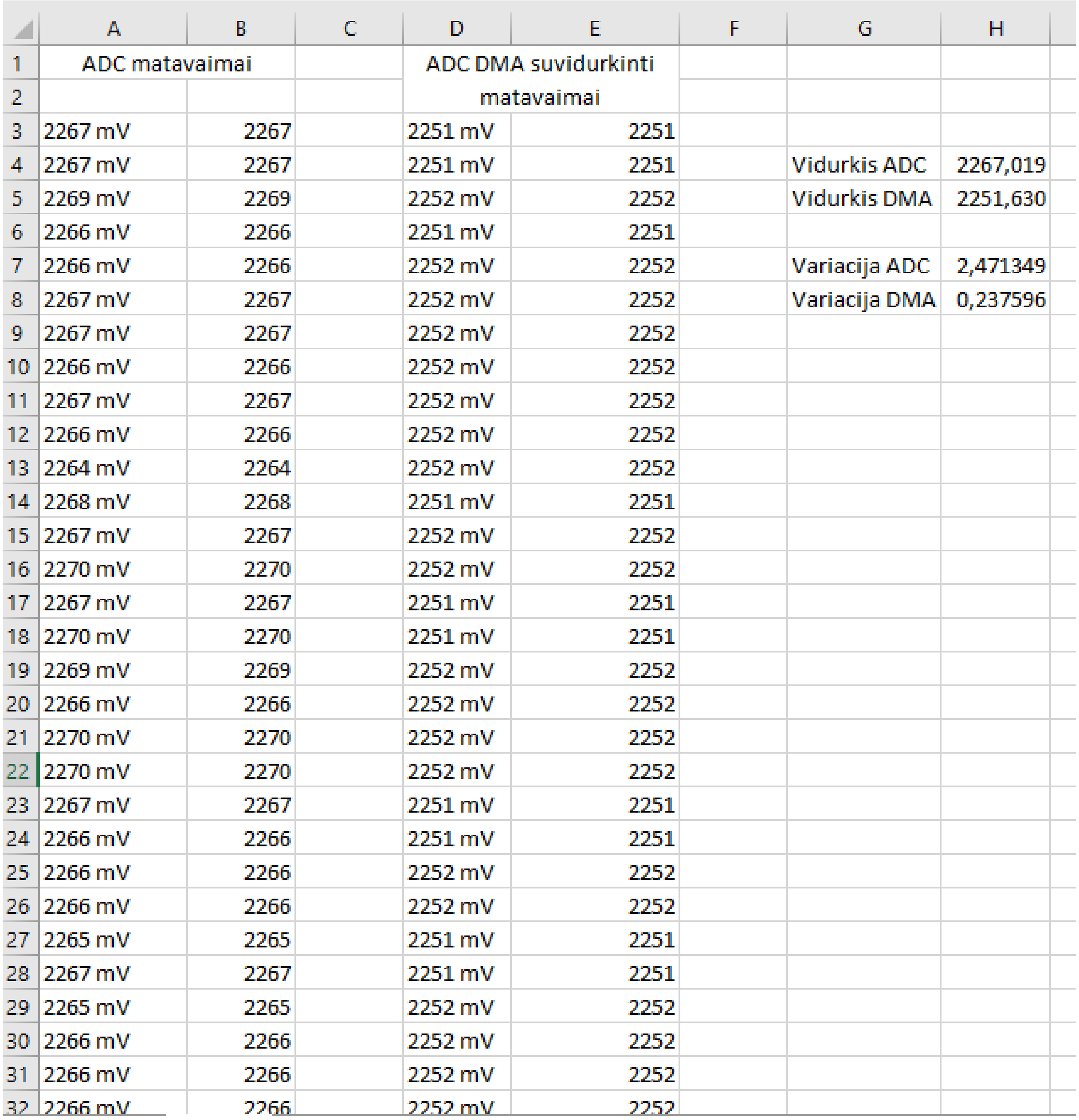


```c
  /* USER CODE BEGIN 2 */
  HAL_DAC_Start(&hdac, DAC_CHANNEL_2);
  HAL_ADC_Start_DMA(&hadc1, &value, 32);
  /* USER CODE END 2 */

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


	  if (conversion_ended)
	  {
		  conversion_ended = 0;
		  uint32_t value_acc = 0;
		  uint8_t i;
		  for (i = 0; i < 32; i++)
		  {
			  value_acc += value[i];
		  }
		  HAL_ADC_Start_DMA(&hadc1, &value, 32);
		  value_avg = value_acc / 32;

		  HAL_DAC_SetValue(&hdac, DAC_CHANNEL_2, DAC_ALIGN_12B_R, value_avg);
		  size_t len = snprintf((char*)tx_buffer, sizeof(tx_buffer), "%d mV\r\n", value_avg * 3300 / 4095);
		  HAL_UART_Transmit(&huart3, tx_buffer, len, 1000);


	  }
	  HAL_Delay(100);
    /* USER CODE END WHILE */
```

```
<terminated> LD2_Task3 Debug [STM32 C/C++ Application] ST-LINK (ST-LINK GDB server) (Terminated Dec 18, 2025, 1:30:45 AM) [pid: 255]
Time elapsed during verifying operation: 00:00:00.205


Download verified successfully


Shutting down...
Exit.
```

```
2252 mV
2252 mV
2251 mV
2251 mV
2252 mV
2251 mV
2251 mV
2252 mV
2252 mV
2252 mV
2252 mV
2252 mV
2251 mV
2252 mV
2251 mV
2251 mV
2252 mV
2252 mV
2251 mV
2251 mV
2251 mV
2252 mV
2252 mV
2251 mV
2251 mV
2252 mV
2251 mV
2252 mV
2251 mV
2251 mV
2252 mV
2251 mV
2251 mV
2251 mV
2251 mV
2251 mV
2252 mV
2252 mV
2252 mV
2251 mV
2251 mV
2251 mV
2251 mV
2251 mV
2251 mV
2252 mV
2251 mV
2251 mV
2252 mV
2251 mV
2252 mV
2251 mV
2252 mV
2252 mV
```

7. Skaičiavimai nekeičiant potenciometro pozicijos



	A	B	C	D	E	F	G	H
1	ADC matavaimai			ADC DMA suvidurkinti matavaimai				
2								
3	2267 mV	2267		2251 mV	2251			
4	2267 mV	2267		2251 mV	2251		Vidurkis ADC	2267,019
5	2269 mV	2269		2252 mV	2252		Vidurkis DMA	2251,630
6	2266 mV	2266		2251 mV	2251			
7	2266 mV	2266		2252 mV	2252		Variacija ADC	2,471349
8	2267 mV	2267		2252 mV	2252		Variacija DMA	0,237596
9	2267 mV	2267		2252 mV	2252			
10	2266 mV	2266		2252 mV	2252			
11	2267 mV	2267		2252 mV	2252			
12	2266 mV	2266		2252 mV	2252			
13	2264 mV	2264		2252 mV	2252			
14	2268 mV	2268		2251 mV	2251			
15	2267 mV	2267		2252 mV	2252			
16	2270 mV	2270		2252 mV	2252			
17	2267 mV	2267		2251 mV	2251			
18	2270 mV	2270		2251 mV	2251			
19	2269 mV	2269		2252 mV	2252			
20	2266 mV	2266		2252 mV	2252			
21	2270 mV	2270		2252 mV	2252			
22	2270 mV	2270		2252 mV	2252			
23	2267 mV	2267		2251 mV	2251			
24	2266 mV	2266		2251 mV	2251			
25	2266 mV	2266		2252 mV	2252			
26	2266 mV	2266		2252 mV	2252			
27	2265 mV	2265		2251 mV	2251			
28	2267 mV	2267		2251 mV	2251			
29	2265 mV	2265		2252 mV	2252			
30	2266 mV	2266		2252 mV	2252			
31	2266 mV	2266		2252 mV	2252			
32	2266 mV	2266		2252 mV	2252			

Su DMA apskaičiuotos reikšmės tikslesnės, nes DMA neapkrauna procesoriaus ir atlieka nuskaitymus vienodo laiko intervale. Reikšmės nekeliauja per magistrale, tad dar papildomai sumažėja triukšmo bei paprasčiausiai apskaičiuojamas vidurkis sumažina atsitiktinį triukšmą.