

# Podstawy techniki mikroprocesorowej 2

## Lab 2 - Przerwania i liczniki - biblioteka HAL

<b>Imię i Nazwisko:</b>	Remigiusz Mielcarz, Grzegorz Salzburg
<b>Nr indeksu:</b>	252887, 252912

<b>Termin zajęć: dzień tygodnia, godzina:</b>	Środa 14:10-17:10 TP
<b>Numer grupy ćwiczeniowej:</b>	Y03-45f
<b>Data wykonania ćwiczenia:</b>	27.10.2021
<b>Termin do oddania sprawozdania:</b>	10.11.2021
<b>Prowadzący kurs:</b>	Dr inż. Krzysztof Halawa

### Spis treści

1 Cel projektu	2
2 Zadania do wykonania	2
3 Schemat podłączenia	2
4 Kod programu	3
5 Konfiguracja wewnętrznego timera	5
6 Konfiguracja timera TIM1	5
7 Konfiguracja timera TIM2	6
8 Podsumowanie	6

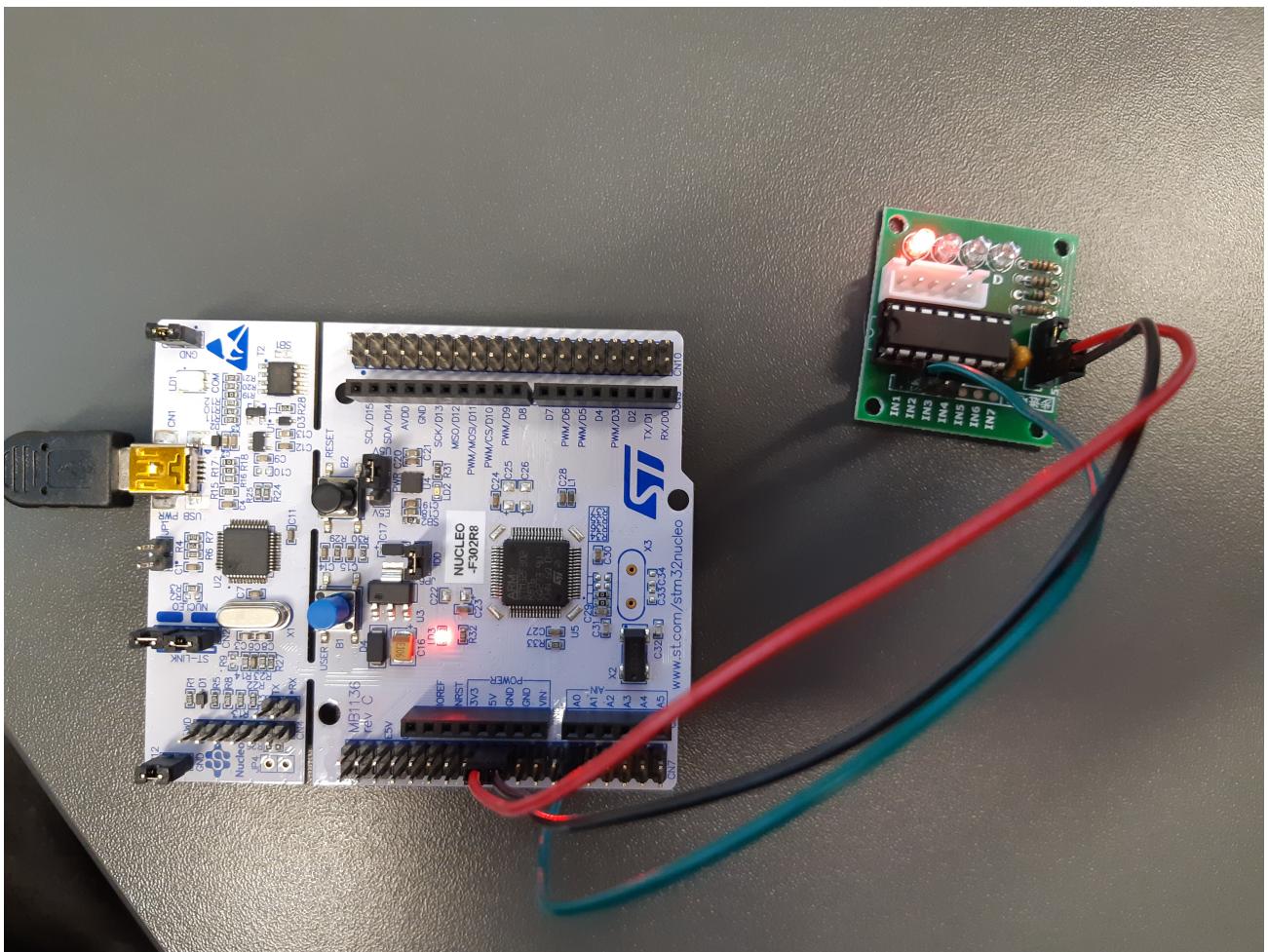
# 1 Cel projektu

Celem projektu jest zapoznanie się z działaniem przerwań i timerów, które dostępne są w układach z rodziny STM32 w bibliotece HAL.

## 2 Zadania do wykonania

- W funkcji obsługi zewnętrznego przerwania (po naciśnięciu niebieskiego przycisku na nucleo) powinien być zmieniany stan diody LED na nucleo.
- Modyfikacja zadania 1 aby opadające zbocze wywoływało przerwanie.
- Za pomocą timera TIM1 należy co jedna sekundę wywoływać przerwanie. W funkcji obsługi przerwania należy zmieniać stan LED na przeciwny.
- Konfiguracja w CubeNX TIM2 do pracy w trybie PWM. Powinien on sterować diodą LED na module silnika krokowego. Konfiguracja TIM2 aby dioda LED była cyklicznie zapalana na 1s i gaszona na 2 sekundy.

## 3 Schemat podłączenia



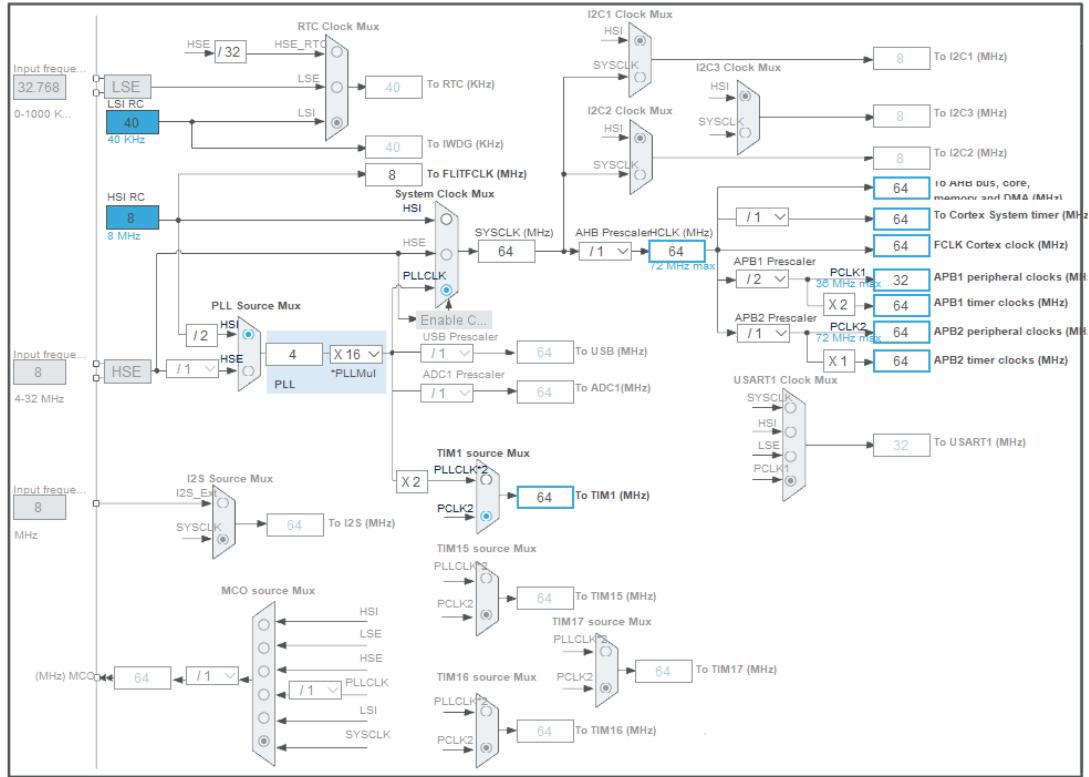
## 4 Kod programu

```
1 #include "main.h"
2
3 /* Private variables -----*/
4 TIM_HandleTypeDef htim1;
5 TIM_HandleTypeDef htim2;
6
7 /* Private function prototypes -----*/
8 void SystemClock_Config(void);
9 static void MX_GPIO_Init(void);
10 static void MX_TIM1_Init(void);
11 static void MX_TIM2_Init(void);
12
13
14 /* Private user code -----*/
15 /* USER CODE BEGIN 0 */
16
17 // Poczatek zad 1 i 2
18 void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin)
19 {
20     if (GPIO_Pin == B1_Pin)
21     {
22         HAL_GPIO_TogglePin(LD2_GPIO_Port, LD2_Pin);
23     }
24 }
25 // Koniec zad 1 i 2
26
27 //Poczatek zad 3
28 void HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef* htim)
29 {
30     if (htim == &htim1)
31     {
32         HAL_GPIO_TogglePin(LD2_GPIO_Port, LD2_Pin);
33     }
34 }
35 // Koniec zad 3
36
37 /* USER CODE END 0 */
38
39 /**
40  * @brief  The application entry point.
41  * @retval int
42  */
43 int main(void)
44 {
45     /* MCU Configuration-----*/
```

```

46
47 /* Reset of all peripherals, Initializes the Flash interface and the Systick. */
48 HAL_Init();
49
50 /* Configure the system clock */
51 SystemClock_Config();
52
53 /* USER CODE BEGIN SysInit */
54
55 /* USER CODE END SysInit */
56
57 /* Initialize all configured peripherals */
58 MX_GPIO_Init();
59 MX_TIM1_Init();
60 MX_TIM2_Init();
61 /* USER CODE BEGIN 2 */
62
63 //Poczatek zad 3
64     HAL_TIM_Base_Start_IT(&htim1);
65 // Koniec zad 3
66
67 //Poczatek zad 4
68     HAL_TIM_Base_Start_IT(&htim2);
69     HAL_TIM_PWM_Start(&htim2, TIM_CHANNEL_1);
70 // Koniec zad 4
71
72 /* USER CODE END 2 */
73
74 /* Infinite loop */
75 /* USER CODE BEGIN WHILE */
76 while (1)
77 {
78     /* USER CODE END WHILE */
79
80     /* USER CODE BEGIN 3 */
81 }
82 /* USER CODE END 3 */
83 }
```

## 5 Konfiguracja wewnętrznego timera



## 6 Konfiguracja timera TIM1

**TIM1 Mode and Configuration**

**Mode**

Slave Mode	Disable
Trigger Source	Disable
Clock Source	Internal Clock
Channel1	Disable
Channel2	Disable
Channel3	Disable
Channel4	Disable
Channel5	Disable
Channel6	Disable
Combined Channels	Disable

**Configuration**

**Reset Configuration**

**Parameter Settings** | **User Constants** | **NVIC Settings** | **DMA Settings**

Configure the below parameters :

**Search (Ctrl+F)** **?** **i**

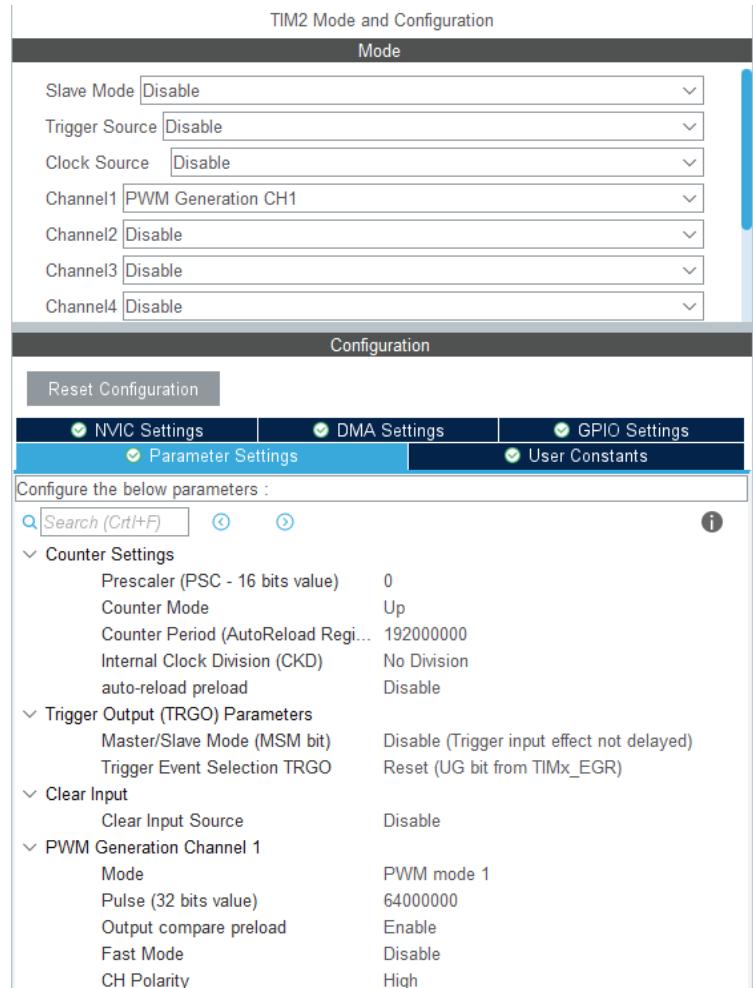
**Counter Settings**

- Prescaler (PSC - 16 bits value) : 1000
- Counter Mode : Up
- Counter Period (AutoReload Regi...) : 64000
- Internal Clock Division (CKD) : No Division
- Repetition Counter (RCR - 16 bits ...) : 0
- auto-reload preload : Disable

**Trigger Output (TRGO) Parameters**

- Master/Slave Mode (MSM bit) : Disable (Trigger input effect not delayed)
- Trigger Event Selection TRGO : Reset (UG bit from TIMx\_EGR)
- Trigger Event Selection TRGO2 : Reset (UG bit from TIMx\_EGR)

## 7 Konfiguracja timera TIM2



## 8 Podsumowanie

- Zad 1 i Zad 2 są zadaniami analogcznymi, z różnicą w rodzaju zbocza. Zadania korzystają z funkcji `void HAL_GPIO_Callback`, w której sprawdzana jest zgodność przycisku, i następnie zmiana stanu diody.
- Zad 3 polega na co sekundowym wywołaniu przerwania, z użyciem timera TIM1. Zadanie w funkcji `main` zawiera start zegara z użyciem polecenia `HAL_TIM_Base_Start_IT(&htim1)`. Poza funkcją `main` korzystamy z funkcji `void HAL_TIM_PeriodElapsedCallback`, która zmienia nam stany diody led na przeciwny.
- Zad 4 polega na cyklicznym zapalaniu i zgaszaniu diody led (1s i 2s). Do tego zadania skonfigurowany został timer TIM2 do pracy w trybie PWM. Zadanie zawiera tylko 2 polecenia w funkcji `main`, start timera i jego uruchomienie na kanale 1 (channel 1). Po poprawnym ustawieniu parametru **Counter period** na 192 miliony i ustawieniu parametru **Pulse** na wartość 3 razy mniejszą czyli 64 miliony, otrzymujemy poprawne działanie programu, gdzie dioda led pali się 1 sekundę i nie pali przez 2 sekundy.