

# Przerwania, liczniki

## Przerwania - sekwencja wywoływania funkcji:

Funkcje obsługi przerwania znajdują się w pliku `stm32f1xx_it.c` i mają nazwy w formacie `xxx_IRQHandler()`. Litery `xxx` oznaczają ciąg znaków zależący od przerwania. Przerwania zewnętrzne mogą być powodowane odpowiednią zmianą stanów na wyprowadzeniach portów mikrokontrolera. Przerwania wewnętrzne mogą być wyzwalane przez różne układy mikrokontrolera np. liczniki, komparatory, DMA, itd. Po wystąpieniu przerwania wywoływana jest odpowiednia funkcja `xxx_IRQHandler()`. W przypadku niekorzystania z biblioteki HAL należałoby wpisać odpowiedni kod bezpośrednio w jej wnętrzu. Biblioteka HAL umieszcza w niej automatyczne wywołanie funkcji `HAL_xxx_IRQHandler()` znajdującej się w pliku `stm32f1xx_hal_xxx.c`. Wewnątrz tej funkcji przeważnie znajduje się m.in. kasowanie flagi przerwania oraz wywołanie funkcji o nazwie `HAL_xxx_Callback`, którą programista powinien zaimplementować w pliku `main.c`. W typowych sytuacjach w `HAL_xxx_Callback` programista wprowadza kod, który ma się wykonać po wywołaniu przerwania.

## Liczniki:

Mikrokontroler `stm32f302` zawiera następujące liczniki:

SysTick – (Cortex System Timer) – 24bitowy licznik systemowy służący do odliczania czasu m.in. w bibliotece HAL (funkcja `HAL_DELAY`) oraz w systemie czasu rzeczywistego RTOS.

RTC – (Real Timer Clock) – możliwe podtrzymywanie baterijne (Backup Domain), jeżeli jest podłączona bateria do wyprowadzenia VBAT.

TIM1 – najbardziej rozbudowany licznik uniwersalny,

TIM2, TIM15, TIM16, TIM17 – liczniki uniwersalne

TIM6 – licznik do wyzwalania DAC.

Po skonfigurowaniu licznika w CubeMX, do rozpoczęcia jego pracy należy użyć odpowiedniej funkcji z tab. 1.

Tab. 1 Funkcje z biblioteki HAL do aktywacja licznika

Tryb pracy	Aktywacja	Aktywacja z przerwaniem	Aktywacja z DMA
Tryb podstawowy	<code>HAL_TIM_Base_Start</code>	<code>HAL_TIM_Start_IT</code>	<code>HAL_TIM_Base_Start_DMA</code>
Output_Compare	<code>HAL_TIM_OC_Start</code>	<code>HAL_TIM_OC_Start_IT</code>	<code>HAL_TIM_OC_Start_DMA</code>
Input_Capture	<code>HAL_TIM_IC_Start</code>	<code>HAL_TIM_IC_Start_IT</code>	<code>HAL_TIM_IC_Start_DMA</code>
PWM	<code>HAL_TIM_PWM_Start</code>	<code>HAL_TIM_PWM_Start_IT</code>	<code>HAL_TIM_PWM_Start_DMA</code>
Generacja	<code>HAL_TIM_OnePulse_Start</code>	<code>HAL_TIM_OnePulse_Start_IT</code>	<code>HAL_TIM_OnePulse_Start_DMA</code>

pojedynczego impulsu			
Praca z enkoderem	HAL_TIM_Encoder_Start	HAL_TIM_Encoder_Start_IT	HAL_TIM_Encoder_Start_DMA

Przed przystąpieniem do ćwiczeń proszę przeczytać więcej informacji o przerwaniach i timerach

### Zadania:

**Zad. 1.** W funkcji obsługi zewnętrznego przerwania (po naciśnięciu niebieskiego przycisku na nucleo) powinien być zmieniany stan diody LED na nucleo za pomocą funkcji `HAL_GPIO_TogglePin(LD2_GPIO_Port, LD2_Pin);`

Układ przerwań należy skonfigurować żeby rosnące zbocze powodowało przerwanie.

### Wskazówki:

W CubeMX dla odpowiedniej linii trzeba włączyć i skonfigurować obsługę przerwania zewnętrznego (ang. External Interrupt). Między innymi w System Core po lewej stronie należy wybrać GPIO i w zakładce GPIO odpowiednio skonfigurować przerwanie zewnętrzne. Na podstawie analizy schematu płytki nucleo można zauważyć, że dla PC13 można ustawić No Pull up Pull down. Dodatkowo w zakładce NVIC (akronim od ang. Nested Vector Interrupt Control) należy włączyć odpowiednie przerwanie.

Proszę obejrzeć funkcję `void EXTI15_10_IRQHandler(void)` w pliku `stm32f3xx_it`

W pliku `main.c` zmianę stanu diody LED należy zaimplementować wewnątrz funkcji

```
void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin)
{
    /
}
```

Funkcję tę należy utworzyć w odpowiednim miejscu w plku `main.c`

**Zad 2.** Należy zmodyfikować zadanie 1 aby opadające zbocze wywoływało przerwanie.

**Zad. 3.** Za pomocą timera TIM1 należy co jedna sekundę wywoływać przerwanie. W funkcji obsługi przerwania należy zmieniać stan LED na przeciwny. W funkcji `main` należy wystartować licznik za pomocą instrukcji

```
HAL_TIM_Base_Start_IT(&htim1);
```

IT z tyłu nazwy funkcji oznacza start licznika z użyciem przerwań (przerwania należy uprzednio odpowiednio skonfigurować w CubeMX)

Zmianę stanu diody należy zaimplementować w pliku `main.c` wewnątrz funkcji

```
void HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef* htim)
{
    HAL_GPIO_TogglePin(LD2_GPIO_Port, LD2_Pin);
}
```

### Wskazówki do konfiguracji TIM2 w CubeMX.

Częstotliwość taktowania mikrokontrolera można sprawdzić w CubeMX w zakładce Clock Configuration. Częstotliwość zegara systemowego podana jest w prostokącie przy (SYSCLK).

W CubeMX należy wybrać w Timers licznik TIM1. Następnie w TIM1 Mode and Configuration trzeba wybrać w Clock Source opcje Internal Clock. Spowoduje to zliczanie przez licznik taktów sygnału SYSCLK.

Znając częstotliwość SYSCLK można ustalić okres sygnału przez ustawienie wartości, do której zlicza licznik, którą konfiguruje się w zakładce Parameters Settings w Period Settings (AutoReload Register - 16 bits value ). W celu uzyskania dłuższych okresów czasu można w polu Prescaler wpisać wartość 1000, która spowoduje zliczanie co tysięcznego taktu SYSCLK. W zakładce NVIC Settings należy włączyć generowanie przerw przy przepełnieniu licznika (TIM1 update and TIM16 interrupts).

Warto obejrzeć funkcję `static void MX_TIM2_Init(void)` w main.c.

**Zad. 4.** Należy skonfigurować w CubeMX TIM2 do pracy w trybie PWM. Powinien on sterować diodą LED na module silnika krokowego (silnik należy odłączyć od modułu). Należy TIM2 skonfigurować aby dioda LED była cyklicznie zapalana na 1s i gaszona na 2 sekundy. Nie należy używać przerw. Licznik w funkcji main należy uruchomić za pomocą instrukcji `HAL_TIM_PWM_Start(&htim2, TIM_CHANNEL_1);`

#### **Wskazówki do konfiguracji TIM2 w CubeMX.**

Częstotliwość taktowania mikrokontrolera można sprawdzić w CubeMX w zakładce Clock Configuration. Częstotliwość zegara systemowego podana jest w prostokącie przy (SYSCLK).

W CubeMX należy wybrać w Timers licznik TIM2. Następnie w TIM2 Mode and Configuration trzeba wybrać dla Channel1 opcję PWMGeneration CH1. Proszę zwrócić uwagę, przy którym wyprowadzeniu mikrokontrolera pojawi się napis TIM2\_CH1. Na tym wyprowadzeniu będzie automatycznie generowany sygnał PWM.

Znając częstotliwość SYSCLK można ustalić okres sygnału przez ustawienie wartości, do której zlicza licznik, którą konfiguruje się w zakładce Parameters Settings w polu Counter Period (AutoReload Register - 32 bits value ). W PWM Generation Channel1 proszę wybrać tryb PWM Mode1 w polu Mode. Wypełnienie sygnału generowanego na wyprowadzeniu TIM2\_CH1 jest ustalane w polu Pulse (32bit value).

Warto obejrzeć funkcję `static void MX_TIM2_Init(void)` w main.c.