

# STM32CubeIDE - poradnik

Wprowadzenie, konfiguracja, programowanie, debugowanie

## Spis treści

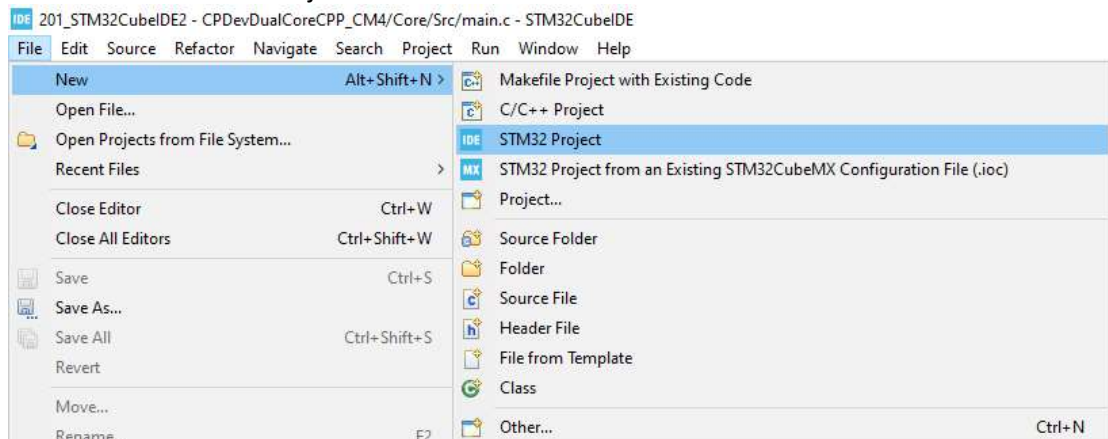
1.	Wstęp.....	2
2.	Utworzenie projektu w STM32CubeIDE .....	2
3.	Praca z konfiguratorem STM32CubeMX.....	3
4.	GPIO – uniwersalne wejścia/wyjścia .....	5
	Wyjście .....	5
	Wejście .....	5
	Przerwania.....	5
5.	ADC .....	6
6.	Uruchamianie .....	6
7.	Debugowanie.....	7
8.	Podpowiadanie składni.....	7

## 1. Wstęp

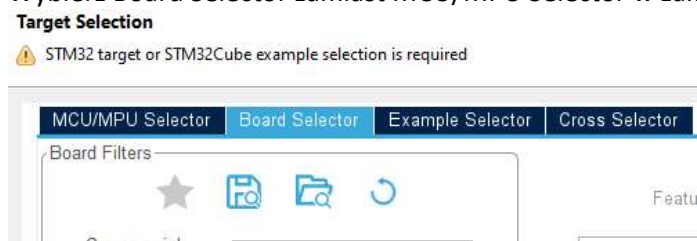
Do pracy posłużą nam kompleksowe środowisko o nazwie STM32CubeIDE, które integruje IDE z konfiguratorem STM32CubeMX.

## 2. Utworzenie projektu w STM32CubeIDE

1. File -> New -> STM32 Project.

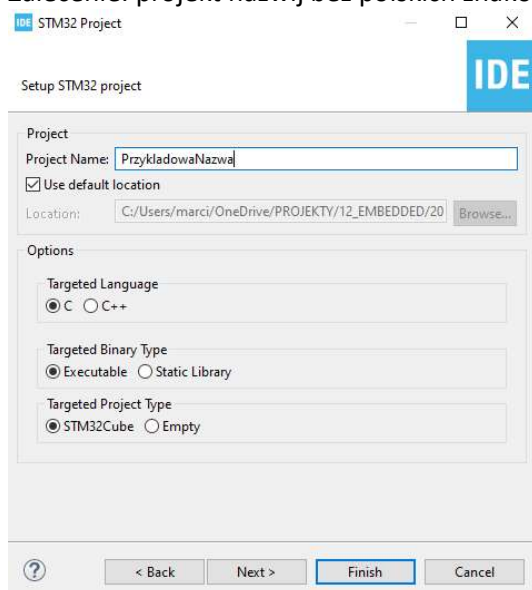


2. Wybierz Board Selector zamiast MCU/MPU Selector w zakładkach.

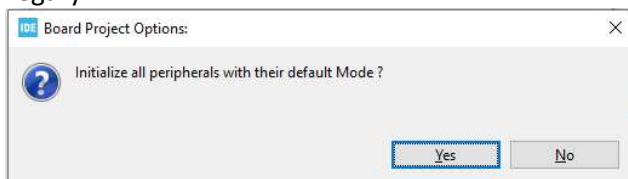


3. Wyszukaj płytę którą otrzymałeś na zajęciach i kliknij przycisk Next. Następnie pojawi się okno pozwalające wpisać nazwę, wybrać język programowania i inne opcje. Po wpisaniu nazwy projektu pozostaw ustawienia domyślne i kliknij przycisk Finish.

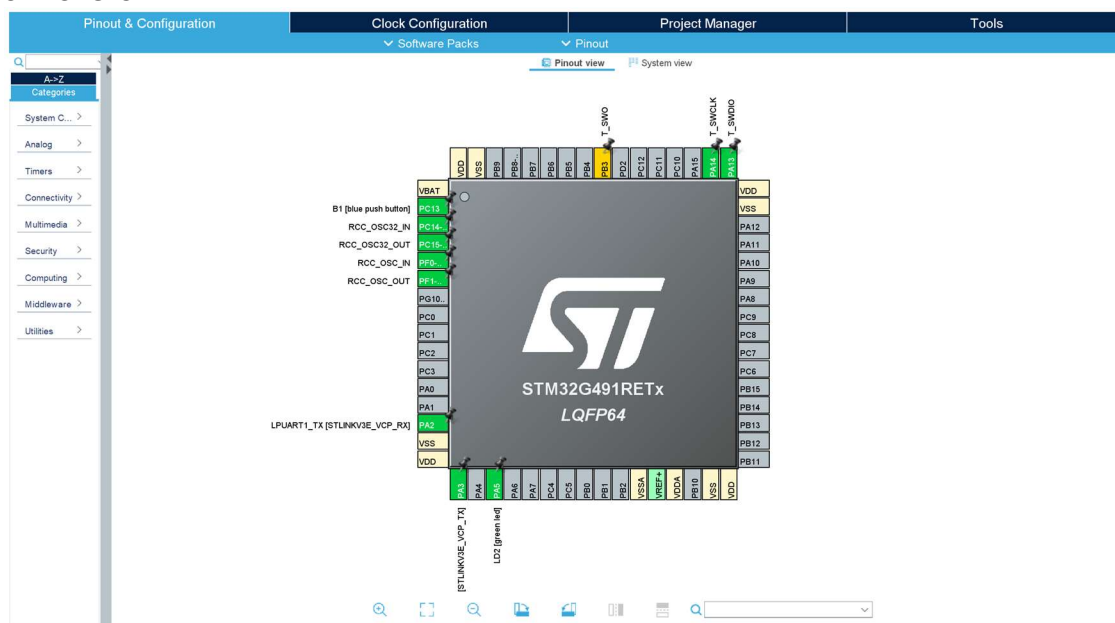
Zalecenie: projekt nazwij bez polskich znaków i bez spacji.



4. W zależności od polecenia w głównej instrukcji wybierz jedną z opcji. Domyślna konfiguracja zawiera inicjację przygotowanych przez producenta peryferii dla danej płyty deweloperskiej, takich jak przyciski, diody, wyświetlacz, komunikacja UART, ustawienia debugowania, czy zegary.



5. Plik programu STM32CubeMX z rozszerzeniem .ioc przechowuje przygotowaną konfigurację mikrokontrolera. W zakładce Pinout & Configuration po lewej stronie znajdują się wszystkie dostępne peryferia wybranego układu. Po prawej stronie znajduje się widok wykorzystywanego układu, który prezentuje przypisanie pinów do konkretnych peryferii z opcjonalnymi nazwami. W celu podejrzenia lub przypisania trybu pinu należy kliknąć LPM (lewym przyciskiem myszy) w odpowiedni prostokąt. W celu zmiany nazwy pinu używamy PPM (prawy przycisk myszy). Na poniższym zrzucie zaprezentowana została przykładowa konfiguracja po wybraniu inicjacji domyślnych peryferii dla płyty Nucleo-64 z układem STM32G491RE.



#### UWAGA!

Po każdej zmianie, przed przystąpieniem do pisania kodu, należy wygenerować nowe ustawienia do projektu (skrót Ctrl + S) lub kliknąć PPM na pliku .ioc i wybrać Generate Code.

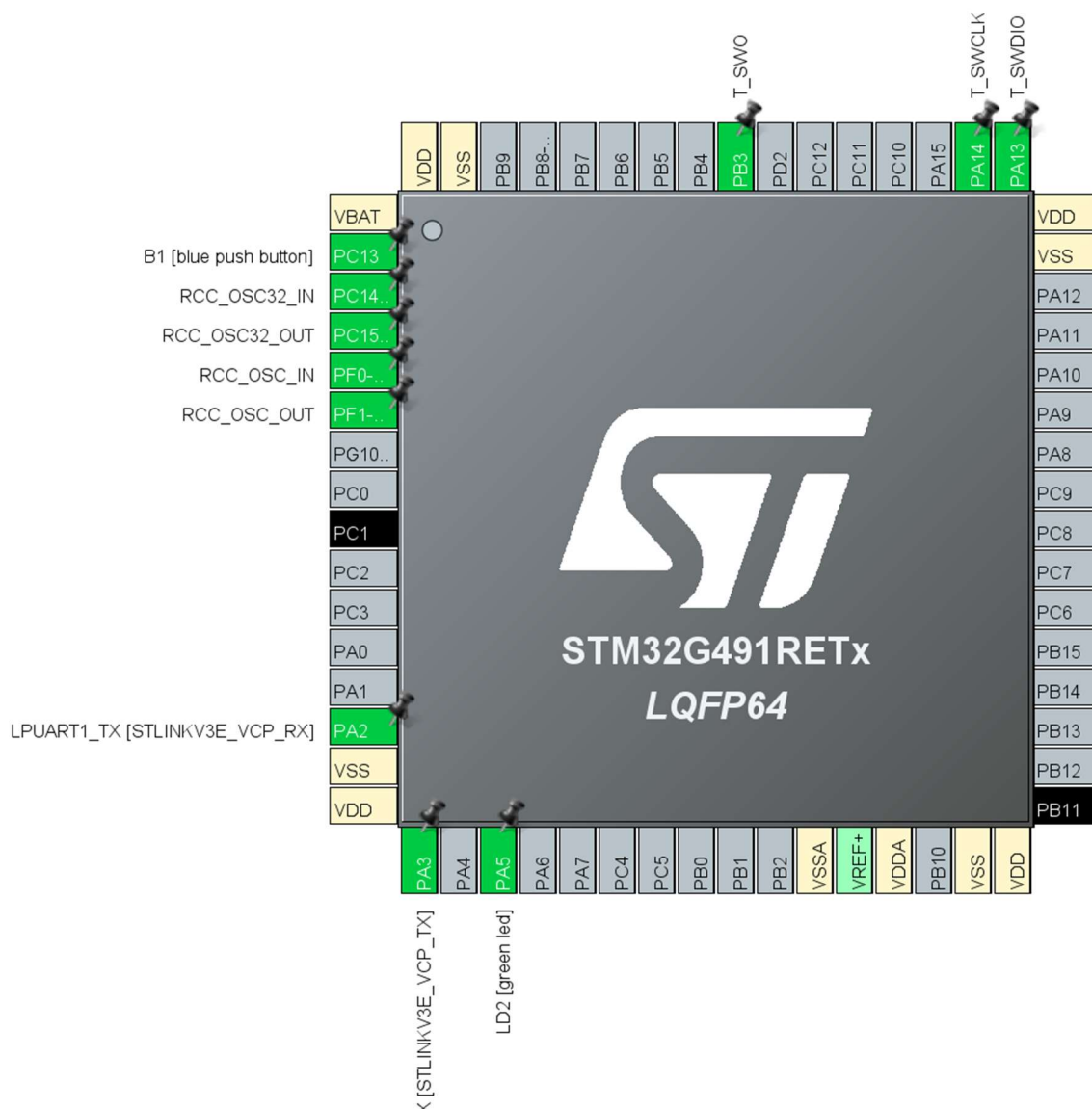
### 3. Praca z konfiguratorem STM32CubeMX

Uzyskanie konkretnych zachowań peryferii na wybranych pinach wymaga odpowiedniej inicjacji. Podstawową ścieżką jest wybieranie interesującego nas pinu i podglądnięcie po kliknięciu LPM dostępnych peryferii sprzętowych. W celu wybrania konkretnego ustawienia wystarczy wybrać interesującą opcję.

Oznaczenia kolorów:

- szary – piny niezainicjowane,
- zielony – piny poprawnie zainicjowane z przypisanymi funkcjami,
- żółty – wybrana funkcjonalność bez kompletnej ilości ustawień do poprawnego działania.

Część dostępnych opcji jest mapowana pozwalając wykorzystać np. interfejs UART na trzech różnych parach pinów. Dostępne opcje mapowania można sprawdzić klikając LPM trzymając LCTRL na zainicjowany pin. Dostępne opcje mapowania podświetlą odpowiednie piny na czarno. Poniżej przedstawiono przykład możliwego zmapowania LPUART\_1 na inną parę pinów.



Wszelkie opcje konfiguracyjne dla wybranych sprzętowych peryferii na pinach dostępne są po lewej stronie w odpowiednich zakładkach. Można tam znaleźć również opcje, które nie są bezpośrednio związane z pinami np. FreeRTOS.

## 4. GPIO – uniwersalne wejścia/wyjścia

### Wyjście

Na interesujący nas pin klikamy LPM, a następnie wybieramy opcję GPIO\_Output. Nowo wybrany pin pojawi się w zakładce GPIO. W niej możliwa jest zmiana parametrów pracy, takich jak stan początkowy, tryb pracy itd.

### Wejście

W celu uruchomienia trybu wejściowego (Input) dla danego pinu postępujemy analogicznie wybierając jednak opcję GPIO\_Input. Możliwy jest odczyt stanu na pinach oznaczonych jako wyjściowe.

The screenshot shows the STM32CubeMX Configuration window. On the left, the 'Categories' list has 'GPIO' selected. The main 'Configuration' panel shows a table of pins and their configurations. Below the table, the 'PA5 Configuration' section is expanded, showing various settings for pin PA5.

Pin Na...	Signal on Pin	GPIO out...	GPIO mode	GPIO Pull...	Maximum ...	Fast Mode	User Label	Modified
PA5	n/a	Low	Output Pu...	No pull-up...	Low	n/a	LD2 [gree...	✓
PC13	n/a	n/a	External In...	No pull-up...	n/a	n/a	B1 [blue p...	✓

PA5 Configuration :

- GPIO output level: Low
- GPIO mode: Output Push Pull
- GPIO Pull-up/Pull-down: No pull-up and no pull-down
- Maximum output speed: Low
- User Label: LD2 [green led]

### Przerwania

Aby uruchomić przerwania od pinu, przy inicjacji należy wybrać opcję GPIO\_EXTIx. W zakładce GPIO warto zwrócić uwagę na tryb pracy przerwań tj. sytuacje wywołujące przerwanie ze względu na zmiany na wejściu (zrzut poniżej).

The screenshot shows a dropdown menu for 'GPIO mode'. The menu is open, displaying several options. The first option, 'External Interrupt Mode with Rising edge trigger detection', is highlighted.

GPIO mode

GPIO Pull-up/Pull-down

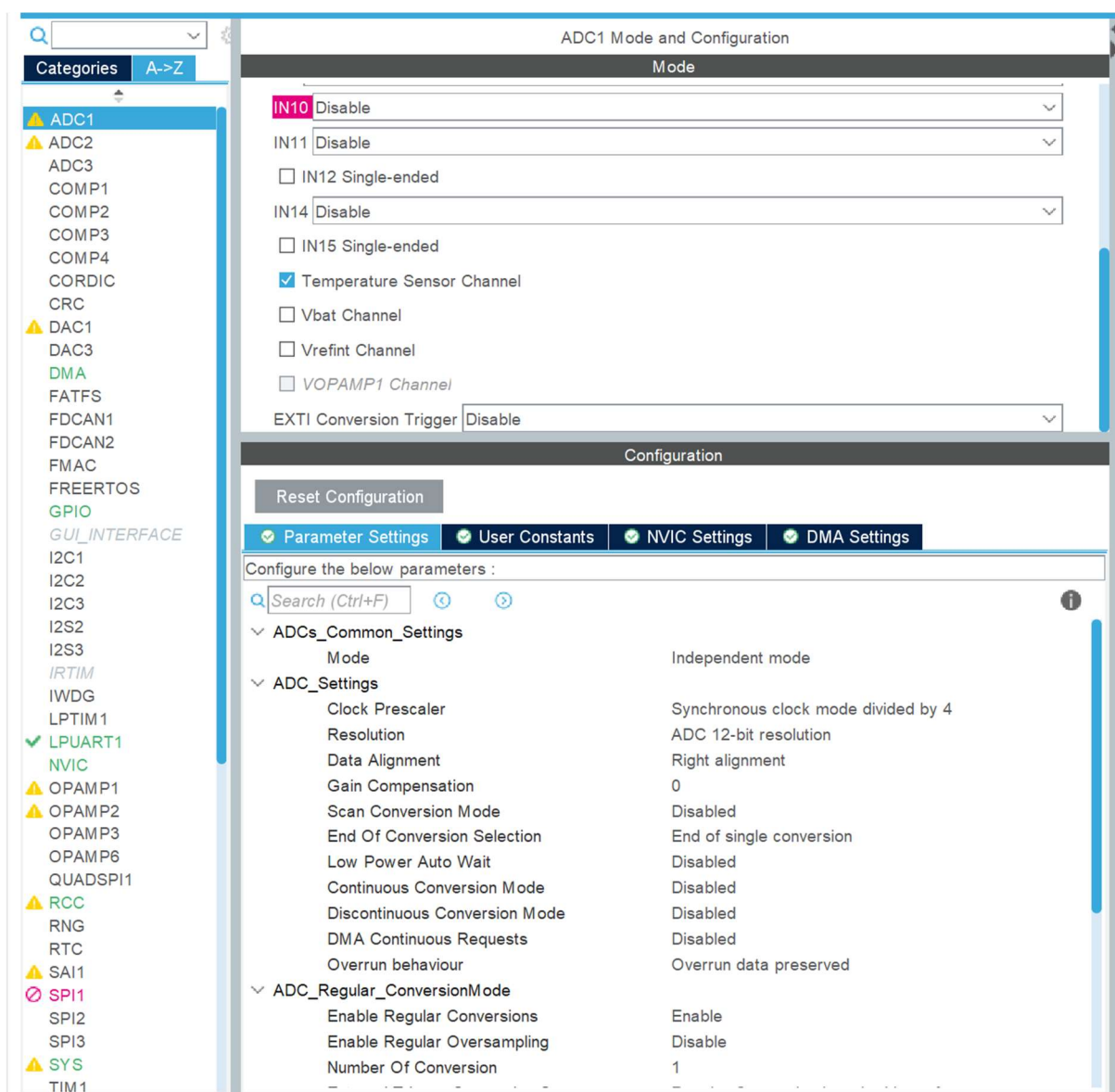
User Label

- External Interrupt Mode with Rising edge trigger detection
- External Interrupt Mode with Rising edge trigger detection
- External Interrupt Mode with Falling edge trigger detection
- External Interrupt Mode with Rising/Falling edge trigger detection
- External Event Mode with Rising edge trigger detection
- External Event Mode with Falling edge trigger detection
- External Event Mode with Rising/Falling edge trigger detection

## 5. ADC

W celu wybrania pomiaru napięcia na pinie należy kliknąć w niego LPM, a następnie wybrać opcję ADCx\_Iny, gdzie x oznacza numer przetwornika ADC w układzie, a y numer kanału multipleksowanego na wejście tego przetwornika.

Wybór pomiaru temperatury wykonuje się bezpośrednio w zakładce ADC, opcja dostępna pod nazwą Temperature Sensor Channel. Dodatkowe, precyzyjne opcje ustawień peryferii można znaleźć poniżej.



Jeżeli projekt nie wymaga precyzyjnego pomiaru napięcia, zaleca się zwiększenie cykli pomiaru.

## 6. Uruchamianie

W celu wgrania i uruchomienia programu na płycie należy wybrać opcję Run z menu Run. Spowoduje to zapisanie projektu, zbudowanie i wgranie za pomocą domyślnie ustawionego połączenia z układem (w naszym przypadku ST-Link).

Przy pierwszym wybraniu opcji Run dla nowego projektu pojawi się ekran opcji konfiguracji, które pozostawiamy i akceptujemy z domyślnymi ustawieniami.

## 7. Debugowanie

Włączenie trybu debugowania przypomina klasyczne uruchomienie programu. Po operacji wgrania układ jest blokowany domyślnie na linii `HAL_Init()`. Należy wybrać opcję Resume (F8) aby rozpocząć normalną pracę mikrokontrolera.

Klikając na numery linii można dodawać breakpointy w czasie rzeczywistym. Breakpoint służy do oznaczania instrukcji, po osiągnięciu której program zatrzyma się w trybie debugowania.



Zakładka po prawej zawiera kilka podstawowych opcji, takich jak włączone breakpointy w plikach, możliwość podglądu zmiennych globalnych w czasie rzeczywistym lub odczytu zmiennych lokalnych dostępnych w danym miejscu bez ich uprzedniego wprowadzenia do programu, jak również dostęp do rejestrów i kodu asemblera.

## 8. Podpowiadanie składni

W edytorze kodu STM32CubeIDE dostępne jest podpowiadanie składni (podobne w działaniu do mechanizmu podpowiadania w środowiskach Visual Studio, Pycharm itp.). Nie aktywuje się jednak ono automatycznie. Aby uzyskać podpowiedź należy po rozpoczęciu pisania wywołania funkcji wcisnąć kombinację Ctrl + Spacja.