

# Dokumentacja projektu - Systemy operacyjne i aplikacje dla Systemów Wbudowanych

Wojciech Mańczak 141083

9 stycznia 2023

## Stacja pogodowa w oparciu o system operacyjny czasu rzeczywistego Zephyr oraz płytę prototypową Arduino Nano 33 BLE

### 1 Wstęp

Celem projektu jest użycie systemu czasu rzeczywistego **Zephyr** do pomiaru i przesyłania danych z czujników temperatury i ciśnienia wchodzących w skład sensora **BMP280**. Do implementacji użyta została platforma **Arduino Nano 33 BLE**, natomiast do odbierania danych aplikacja na systemie Android o nazwie **nRF Connect**. Na projekt zostały nałożone następujące wymagania: Uruchomienie systemu Zephyr na płycie prototypowej Arduino Nano 33 BLE, pomiar danych z czujnika BMP280, połączenie aplikacji z telefonem z płytą przez Bluetooth, aktualizacja danych w aplikacji, dokumentacja w Latexu, prezentacja

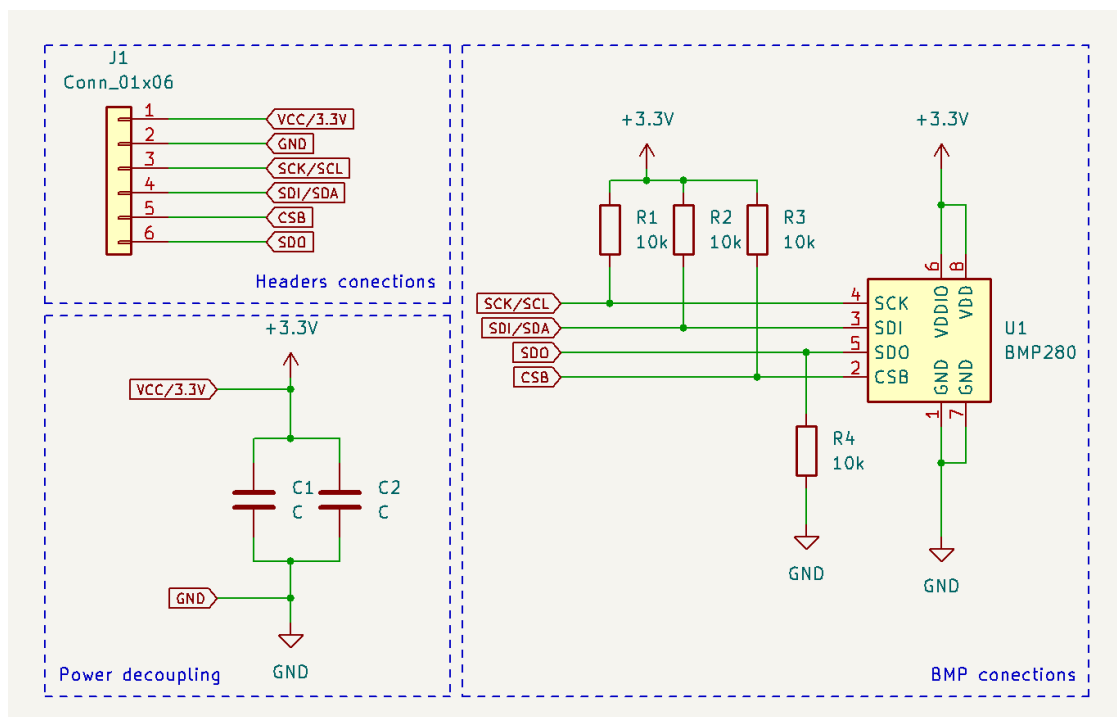
### 2 Wykorzystane oprogramowanie

Do stworzenia projektu użyto edytora Visual Studio Code w wersji 1.68, narzędzia west będącego meta-tool'em wchodzącym w skład systemu Zephyr (w wersji 0.15.2). Dodatkowo użyto oprogramowania Arduino IDE oraz jego narzędzia BOSSAC. Z poziomu telefonu użyto aplikacji nRF Connect.

## 3 Schematy połączeń

### 3.1 BMP280

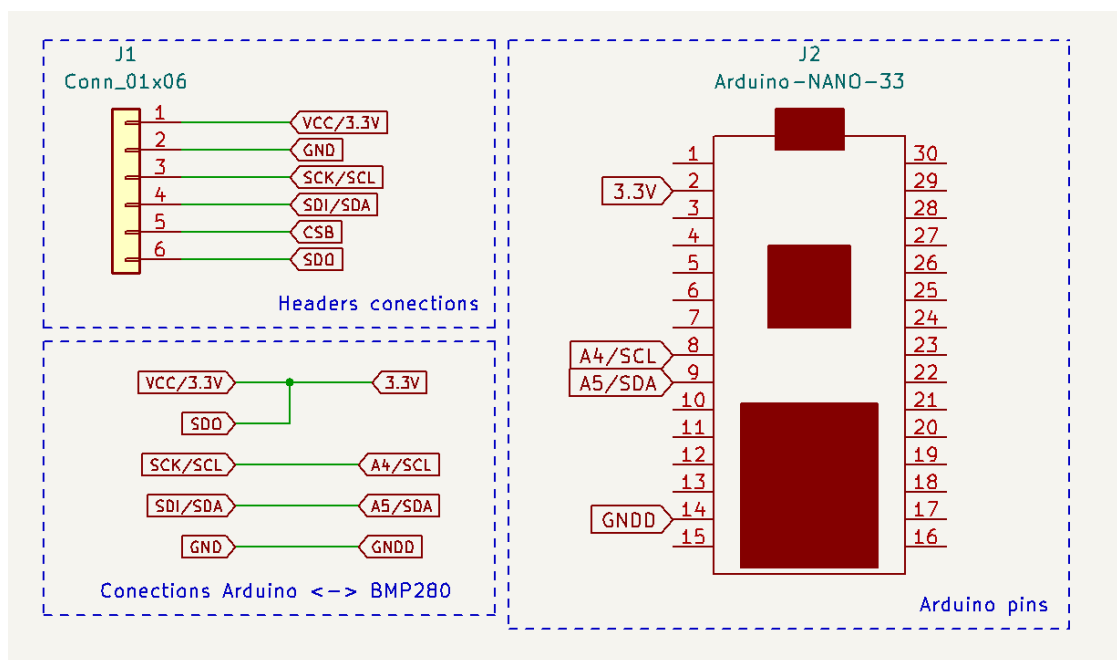
Ponieważ sam sensor BMP przez swoją obudowę jest niewygodny do połączenia, użyto gotową płytkę wraz z osprzętem (Rys: 1).



Rysunek 1: Schemat płytki z BMP280

## 3.2 Arduino nano 33 BLE

Sensor BMP został połączony z Arduino przy pomocy magistrali I2C (Rys: 2). Dprowadza ona również zasilanie do sensora.



Rysunek 2: Schemat połączeń z arduino

## 4 Prezentacja działania

### 4.1 UART Arduino nano 33 BLE

Po uruchomieniu płytki, procesor skonfiguruje czujnik BMP oraz BLE, a następnie przejdzie do pobierania wartości temperatury i ciśnienia oraz przesyłania ich do aplikacji mobilnej. Dodatkowo wszystkie zmierzone parametry oraz komunikaty wysłane zostaną w celu debugowania przez UART (Rys: 3).

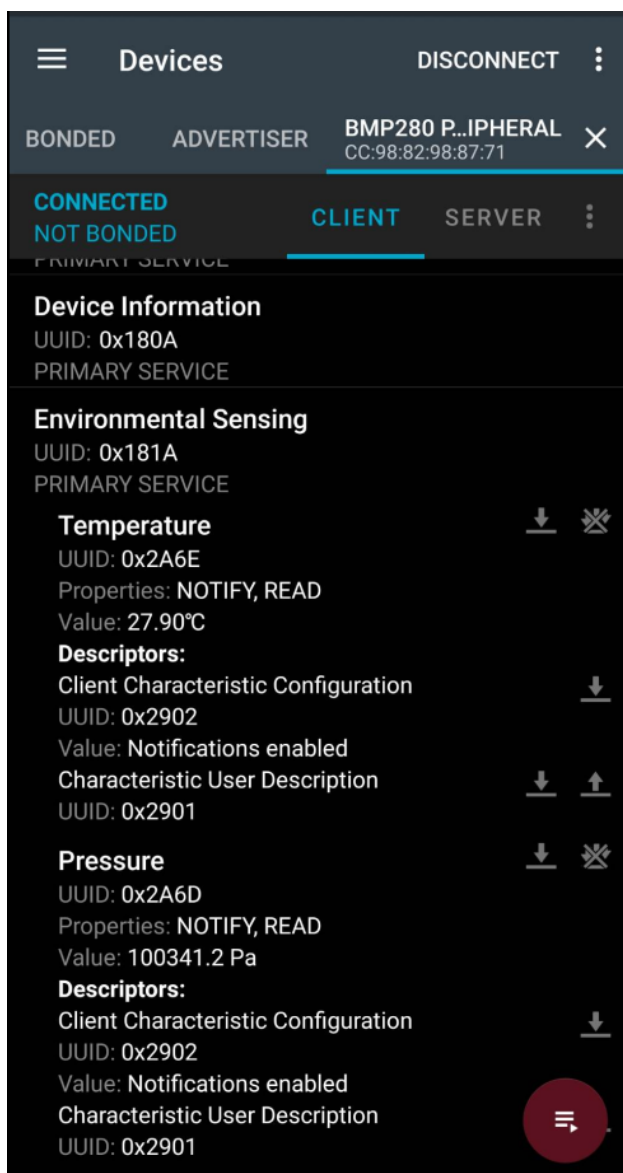
```
[00:00:00.005,828] [][0m<dbg> BME280: bme280_chip_init: ID OK (BMP280)[][0m
[00:00:00.014,282] [][0m<dbg> BME280: bme280_chip_init: "bme280@77" OK[][0m
*** Booting Zephyr OS build zephyr-v3.2.0-2607-gff830fb0f8b5 ***
Found device "bme280@77", getting sensor data
Bluetooth initialized
Advertising successfully started
[00:00:00.018,585] [][0m<inf> usb_cdc_acm: Device suspended[][0m
temp: 27.840000; press: 99.688394;
[00:00:00.179,504] [][0m<inf> usb_cdc_acm: Device resumed[][0m
[00:00:00.179,534] [][0m<inf> usb_cdc_acm: from suspend[][0m
[00:00:00.495,269] [][0m<inf> usb_cdc_acm: Device configured[][0m
temp: 27.840000; press: 99.688394;
temp: 27.840000; press: 99.688394;
temp: 27.840000; press: 99.688394;
temp: 27.840000; press: 99.651562;
temp: 27.840000; press: 99.651562;
temp: 27.840000; press: 99.651562;
temp: 27.840000; press: 99.651562;
temp: 27.840000; press: 99.651562;
temp: 27.840000; press: 99.817578;
Connected
temp: 27.840000; press: 99.817578;
temp: 27.840000; press: 99.817578;
temp: 27.840000; press: 99.817578;
temp: 27.860000; press: 99.999824;
temp_ccc_cfg_changed
temp: 27.860000; press: 99.999824;
press_ccc_cfg_changed
temp: 27.860000; press: 99.999824;
temp: 27.880000; press: 100.176421;
temp: 27.880000; press: 100.176421;
temp: 27.880000; press: 100.176421;
temp: 27.880000; press: 100.176421;
temp: 27.900000; press: 100.341226;
temp: 27.900000; press: 100.341226;
```

Start Zephyra  
Konfiguracja BMP280  
Start BLE  
Ciągły odczyt BMP280 i aktualizacja BLE

Rysunek 3: Dane przekazane poprzez UART od płytki

## 4.2 Aplikacja nRF Connect

W aplikacji (Rys: 4), oprócz danych dotyczących samego urządzenia (takich jak nazwa, rodzaj sensora itp), uzyskujemy podgląd dwóch wartości mierzących środowisko - temperatury i ciśnienia. Wartości te możemy odświeżać ręcznie klikając pojedynczą strzałkę, lub odświeżać tak często jak często odświeża je Arduino.



Rysunek 4: Podgląd danych w aplikacji nRF Connect

## 5 Pomysły na rozwój

Kilka pomysłów na rozszerzenie funkcjonalności projektu:

1. Dodanie większej ilości czujników do projektu, takich jak czujnik wilgotności lub czujnik jakości powietrza, aby mierzyć dodatkowe parametry środowiskowe.
2. Dodanie akumulatora, aby uczynić projekt bardziej przenośnym, i zaimplementować tryb niskiego zużycia energii, aby wydłużyć żywotność baterii.
3. Wdrożenie przetwarzania danych z czujników takich jak wykorzystanie algorytmów uczenia maszynowego do wykrywania wzorców lub trendów w danych.

## 6 Wnioski

Udało się pomyślnie połączyć z czujnikiem BMP280 i odczytać z niego odczyty temperatury i ciśnienia. Zephyr był w stanie przesłać dane z czujnika przez BLE do aplikacji mobilnej, umożliwiając zdalne wyświetlanie danych i dostęp do nich. Uzyskano praktyczną wiedzę na temat komponentów sprzętowych i programowych zaangażowanych w projekt, w tym systemu operacyjnego Zephyr, komunikacji BLE i interfejsów czujników.