

**Dokumentacja Projektu grupowego**

**Dokumentacja techniczna projektu**

**Wydział Elektroniki, Telekomunikacji i Informatyki**

**Politechnika Gdańska**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nazwa i akronim projektu:  Lądowy dorn komunikacyjny 5G | | Zleceniodawca:  dr inż. Krzysztof Gierłowski | | | |
| Numer zlecenia:  13@KTIN’2023/23 | | Kierownik projektu:  Mikołaj Storoniak | | Opiekun projektu:  dr inż. Krzysztof Gierłowski | |
|  | | | | | |
| Nazwa / kod dokumentu:  Dokumentacja techniczna produktu – DTP | | Nr wersji:  1.0 | | | |
| Odpowiedzialny za dokument:  Łukasz Czarzasty | | Data pierwszego sporządzenia:  20.01.2024 | | | |
| Data ostatniej aktualizacji:  20.01.2024 | | | |
| Semestr realizacji Projektu grupowego:  1 | | | |
| Historia dokumentu | | | | | |
| Wersja | Opis modyfikacji | Rozdział / strona | Autor modyfikacji | | Data |
| 1.00 | Wstępna wersja | całość | Karolina Rychert | | 20.02.2024 |

**Spis treści**

[1 Wprowadzenie - o dokumencie 3](#__RefHeading___Toc114649344)

[1.1 Cel dokumentu 3](#__RefHeading___Toc114649345)

[1.2 Zakres dokumentu 3](#__RefHeading___Toc114649346)

[1.3 Odbiorcy 3](#__RefHeading___Toc114649347)

[1.4 Terminologia 3](#__RefHeading___Toc114649348)

[2 Dokumentacja techniczna projektu 3](#__RefHeading___Toc114649349)

[3 Załączniki 3](#__RefHeading___Toc114649350)

# Wprowadzenie - o dokumencie

## Cel dokumentu

Celem dokumentu jest udokumentowanie informacji dotyczących produktu, jego cech funkcjonalnych, parametrów technicznych, schematów blokowych, oprogramowania, wyników działania, zdjęć produktu, pomiarów, testów oraz innych elementów wymaganych przez opiekuna i klienta.

## Zakres dokumentu

W chwili obecnej dokument ten zawiera instrukcję uruchomienia wszystkich modułów potrzebnych do realizacji projektu, wraz z całą potrzebną konfiguracją.

## Odbiorcy

*Członkowie zespołu projektowego: Mikołaj Storoniak, Mateusz Sagan, Karolina Rychert, Łukasz Czarzasty*

*Katedra Teleinformatyki*

*Dr inż. Krzysztof Gierłowski*

## Terminologia

*Moduł komunikacyjny – urządzenie odpowiedzialne za funkcje komunikacji bezprzewodowej i wykorzystujące WiFi, 5G, ZigBee i Bluetooth*

*Mobilna Platforma – Jeżdżąca platforma sterowana przy pomocy RC, na której zamontowany zostanie moduł komunikacyjny.*

*Dron / Dron komunikacyjny – Produkt końcowy projektu: moduł komunikacyjny osadzony na mobilnej platformie*

# Dokumentacja techniczna projektu

**2.1 Komunikacja między raspberry pi i czujnikiem iNode CS#3 z pomocą Bluetooth LE**

2.1.1 Wymagania

W celu odebrania sygnałów przez raspberry pi, należy pobrać oprogramowanie BlueZ. W tym celu

należy użyć polecenia

sudo apt install bluez

2.1.2 Uruchomienie czujnika

W celu uruchomienia czujnika należy otworzyć obudowę i umieścić baterię w przeznaczonym na nie

miejsce. Dioda czujnika powinna wtedy zacząć świecić, co oznacza że czujnik został uruchomiony.

2.1.3 Komunikacja między raspberry pi i czujnikiem

Po pobranie BlueZ oraz uruchomieniu czujnika należy przejść do folderu „tools” znajdujący się w

folderze pakietu Bluez. Ścieżka nie jest zawsze taka sama więc należy wcześniej zlokalizować folder

pakietu. Należy więc użyć polecenia

cd /ścieżka\_do\_tools

Następnie należy zresetować interfejs poleceniem

sudo hciconfig hci0 down

sudo hciconfig hci0 up

Po zresetowaniu interfejsu można rozpocząć komunikację z czujnikiem. W tym celu używa się

polecenia

sudo hcidump

Polecenie to czyta i wyświetla ruch Bluetooth jako zbiór bajtów w postaci szesnastkowej. W celu

znalezienia danych z czujnika iNode CS#3 należy zwrócić uwagę na pierwszy bajt w linijce który dla

tego czujnika powinien mieć wartość 93. Reszta danych można zinterpretować na podstawie

dokumentu ze strony producenta czujnika

https://docs.google.com/document/d/1hcBpZ1RSgHRL6wu4SlTq2bvtKSL5\_sFjXMu\_HRyWZiQ/edit#h

eading=h.zfwfp5sey5ug.

**2.2 Uruchomienie Waveshare SIM8262E-M2 5G HAT**

2.2.1 Wymagania

Do prawidłowego uruchomienia sprzętu i wykorzystania go w celu postawienia hotspota wi-fi

wymagane są programy NetworkManager oraz Minicom.

Wymagane jest również całkowite wyłączenie programu ModemManager poleceniem

sudo killall ModemManager

Jego działanie uniemożliwia modemowi prawidłową współpracę z NetworkManagerem.

2.2.2 Hotspot Wi-Fi

W celu utworzenia hotspota bez hasła należy użyć polecenia

nmcli connection add \

type wifi \

con-name <nazwa> \

autoconnect no \

wifimode ap \

wifissid <nazwa> \

ipv4.method shared \

ipv6,method shared

Z kolei polecenie

Nmcli connection up <nazwa>

Pozwala go uruchomić.

Aby sprawdzić, czy hotspot został prawidłowo uruchomiony, należy posłużyć się komendą

nmcli device

Nazwa hotspota powinna znajdować się obok interfejsu wlan0 (domyślnego interfejsu wifi w

Raspberry).

2.2.3 Konfiguracja modemu

Po podłączeniu i uruchomieniu, modem udostępni systemowi urządzenia od /dev/ttyUSB0

do /dev/ttyUSB3.

ttyUSB2 to urządzenie, którym możemy się posłużyć w celu wysłania komend do modemu

(czasem zdarza się, że zamiast ttyUSB0,1,2 i 3 pojawią się ttyUSB0,2,3 i 4; w takim

przypadku należy podłączyć się do ttyUSB3).

Uczynić to można komendą:

sudo minicom –D /dev/ttyUSB2

Jeśli urządzenie jest zablokowane, należy upewnić się, że nie korzysta z niego żaden inny

program.

Po uzyskaniu połączenia wprowadzamy następujące komendy (w tym przypadku dla sieci Play):

AT+CRESET

ATZ

ATQ0 V1 E1 S0=0 &C1 &D2

AT+CFUN=1 //ustawiamy pełną funkcjonalność modemu

AT+CREG=1 +CGREG=1

AT+CGDCONT=1,”IP”,”internet” // internet to nazwa APN sieci PLAY

AT+CGACT=1,1

Po wprowadzeniu tych ustawień, modem powinien je zapamiętać, dzięki czemu po resecie nie

będzie konieczne ich ponowne wprowadzanie.

Odpowiedni interfejs sieciowy powinien podłączyć się samoistnie dzięki NetworkManagerowi.

Aby upewnić się, że tak się stało, możemy ponownie użyć polecenia

nmcli device

*Nowe połączenie oznaczone jest jako* „wired connection”.

2.2.4 Dostęp do internetu

Po wykonaniu powyższych czynności, powinniśmy być w stanie podłączyć się do internetu

przez hotspot udostępniany przez raspberry pi.

2.3 ZigBee

2.3.1 Wymagania

Do prawidłowego działania zaprezentowanej poniżej konfiguracji na raspberry pi musi być zainstalowany Docker w wersji oraz Docker-Compose. Ten drugi w w wersji 1.25 lub wyższej, aby możliwe było obsłużenie pliku docker-compose w wersji 3.3. Domyślne repozytorium raspberry nie zawiera tych wersji, dlatego konieczne jest dodanie użycie repozytorium dockera.

Wymagane jest także podłączenie do raspberry urządzenia SONOFF Zigbee 3.0 USB dongle plus, a następnie zbadanie jak nazywa się odpowiadający mu plik w katalogu /dev. W naszym przypadku jest to /dev/ttyACM0. Nazwę przyłączonego urządzenia można zbadać przy pomocy komendy dmesg.

2.3.2 Zigbee2Mqtt

Do wysyłania dalej komunikatów odebranych przez drona będziemy używać Zigbee2Mqtt. W celu uruchomienia usługi skorzystamy z Dockera oraz Docker-Compose. Poniżej przedstawiono treść pliku docker-compose.yml:

version: ‘3.3’

services:

mqtt:

image: eclipse-mosquitto:2

restart: unless\_stopped

volumes:

- „./mosquitto-data:/mosquitto”

ports:

- „1883:1883”

command: „mosquitto –c /mosquitto-no-auth.conf”

zigbee2mqtt:

container\_name: zigbee2mqtt

restart: unless\_stopped

image: koenkk/zigbee2mqtt:1.30.0

volumes:

- ./zigbee2mqtt-data:/app/data

- /run/udev:/run/udev:ro

ports:

- 8080:8080

environment:

– TZ=Europe/Warsaw

devices:

- /dev/ttyACM0:/dev/ttyACM0

Kluczowe są sekcje environment/TZ oraz devices w usłudze zigbee2mqtt. Jako TZ należy podać strefę czasową w której będzie pracować dron. W devices należy podać nazwę urządzenia zigbee dongle, zbadaną przy pomocy komendy dmsg.

2.3.3 Konfiguracja

W tej samej lokacji, w której znajduje się plik docker-compose należy utworzyć folder o nazwie „zigbee2mqtt-data”. W folderze tym należy utworzyć plik o nazwie „configuration.yaml”, zawierający informacje o konfiguracji usługi zigbee2mqtt. Jego treść powinna wyglądać następująco:

permit\_join: true

mqtt:

base\_topic: zigbee2mqtt

server: mqtt://mqtt

serial:

port: /dev/ttyACM0

adapter: ezsp

frontend:

port: 8080

#frontend: false, jeśli nie będzie potrzeby odczytywania wyników

#pomiarów z przeglądarki

advanced:

network\_key: GENERATE

Kluczowe są następujące sekcje:

* Permit\_join, gdzie możemy pozwolić urządzenium na swobodne dołączenie do sieci.
* serial/port, gdzie musimy podać nazwę naszego urządzenia
* serial/adapter, gdzie musimy podać nazwę adaptera którego używa zigbee dongle i który
* zależy od użytego w nim SoC.
* frontend, gdzie można wyłączyć stronę prezentującą wyniki pomiarów.
* Base\_topic pozwala określić do jakiego topicu mqtt usługa będzie publikować dane.

2.3.4 Uruchomienie

Aby uruchomić usługę, należy wykonać polecenie docker-compose up. Dzięki zastosowanym

opcjom konfiguracji urządzenia będą mogły swobodnie dołączać do sieci i przesyłać wyniki

pomiarów.