



POLITECHNIKA WARSZAWSKA

Wydział Mechatroniki

Praca magisterska

Konrad Traczyk

Projekt urządzenia do lokalizacji pojazdów
w trybie on i off-line

Opiekun pracy:
prof. dr hab. Michał Bartysi

Warszawa, 2017

Streszczenie

Projekt urządzenia do lokalizacji pojazdów w trybie on i off-line

Celem pracy było zaprojektowanie, wykonanie i oprogramowanie urządzenia stanowiącego dodatkowe zabezpieczenie antykradzieżowe pojazdu w postaci lokalizatora wykorzystującego system GNSS (*ang. Global Navigation Satellite System*) oraz GSM (*ang. Global System for Mobile Communications*), zdolnego do analizy stylu jazdy kierowcy, a także całego systemu informatycznego, który pozwoliłby na obsłużenie pozyskanych danych. Założeniem jest, aby projekt mógł być zasilany z instalacji elektrycznej pojazdu, lecz poza tym mógł być od niego całkowicie niezależny.

System opisywany w pracy przeznaczony jest głównie dla trzech grup osób. Pierwszą z nich są właściciele firm posiadających flotę pojazdów, prowadzonych przez pracowników. Umożliwia im zdalny podgląd tras przebywanych przez zatrudnione osoby, a także syntetyczną informację o sposobie jazdy kierowcy. Drugą kategorię stanowią osoby zainteresowane możliwością zdalnego zlokalizowania pojazdu w przypadku kradzieży, dzięki wykorzystaniu mechanizmu alarmowania poprzez wiadomości SMS. Ostatnia grupa to osoby, które chcieliby poprawić swoje umiejętności ekologicznego i bezpiecznego prowadzenia pojazdów, na podstawie dynamicznej oceny wydawanej przez lokalizator.

W skład systemu wchodzą dwa urządzenia elektroniczne. Jedno z nich służy do lokalizacji pojazdu, ciągłej oceny stylu jazdy, wysyłania danych na serwer wykorzystując sieć GSM oraz awaryjnego powiadomiania właściciela w przypadku wykrycia kradzieży. Drugie urządzenie ma za zadanie autoryzować ruch pojazdu i deaktywować funkcję alarmu. Komunikacja pomiędzy obydwooma urządzeniami odbywa się zabezpieczonym kanałem poprzez Bluetooth Low Energy z wykorzystaniem algorytmu szyfrowania AES128.

Kolejnym elementem systemu jest aplikacja serwerowa napisana w języku C++. Jej celem jest odbieranie danych transmitowanych przez urządzenia, przetworzenie ich i zapisanie w bazie danych. Ponadto, jest ona odpowiedzialna za obsługę zapytań HTTP pochodzących ze strony internetowej stanowiący ostatni moduł pracy. Jest ona wykorzystywana w celu rejestracji i logowania użytkowników do systemu, dodawania urządzeń, a także wyświetlania danych o przebytych trasach oraz ocenach stylu jazdy i ich wizualizacji na mapach od firmy Google.

W ramach pracy przeprowadzono również badania nad sposobem analizy i oceny stylu jazdy kierowcy, w oparciu o informacje dotyczące przyspieszenia i jego zmian pochodzące z akcelerometru. Badania zaowocowały przedstawieniem autorskiego algorytmu wykrywania agresywnego sposobu prowadzenia pojazdów. Wykonane samodzielnie testy drogowe wykazały jego skuteczność.

Słowa kluczowe: analiza stylu jazdy, Bluetooth Low Energy, GPS, GSM, lokalizacja, pojazd, samochód, szyfrowanie AES

Abstract

The project of the device localizing vehicles on and off-line

The main goal of this thesis is to design and build the entire IT system which would be used to gather information about vehicles localization and perform dynamic driving style assessment of the driver. It bases on the dedicated electronic device which utilizes systems like GPS for localization, GSM for long range communication or Bluetooth Low Energy and Near Field Communication for mid and short range data exchange. Gathered data is transmitted nearly in the real time to the database server from which it can be displayed via designed as a part of the system website. Moreover, the device has additional function to alarm the car owner about its location in case of theft.

Key words: Bluetooth Low Energy, car, driving style analysis, GPS, GSM, localization, vehicle, AES encryption

Spis treści

Spis treści

1 Wstęp	9
1.1 Zakres pracy	9
1.2 Schemat blokowy urządzeń	11
1.3 Istniejące rozwiązania	12
2 Wstęp teoretyczny	15
2.1 Zastosowane protokoły i systemy	15
2.2 System GSM	16
2.3 System GPS	19
2.4 Protokół NMEA 0183	24
2.5 Protokół Bluetooth Low Energy	27
2.6 Interfejs NFC	30
2.7 Podsumowanie	35
3 Schematy elektroniczne urządzeń	37
3.1 Urządzenie lokalizujące	37
3.1.1 Schemat zasilania	40
3.1.2 Moduł mikrokontrolera	43
3.1.3 Moduł GSM i GPS	44
3.1.4 Moduł pamięci flash	46
3.1.5 Moduł akcelerometru	47
3.1.6 Moduł NFC	48
3.2 Urządzenie deaktywujące	50
4 Schematy płyt drukowanych	53
4.1 Urządzenie deaktywujące	53
4.2 Urządzenie lokalizujące	55

5 Bezpieczeństwo komunikacji	61
5.1 AES	62
5.2 Dodatkowe warianty szyfrowania AES	66
5.3 Realizacja szyfrowania komunikacji w projekcie	67
6 Oprogramowanie	69
6.1 Urządzenie lokalizujące	69
6.2 Urządzenie deaktywujące	77
6.3 Aplikacja serwerowa	80
6.4 Strona internetowa	83
7 Analiza stylu jazdy	89
7.1 Wstęp	89
7.2 Istniejące metody	90
7.3 Badania	91
7.4 Metoda zastosowana w pracy	98
7.5 Rezultaty	101
8 Problemy i ich rozwiązania	107
8.1 Zawieszanie urządzenia lokalizującego	107
8.2 Brak danych z GPS	108
8.3 Kompensacja wpływu przyspieszenia ziemskiego	109
9 Podsumowanie	111

Bibliografia

Wykaz skrótów

Spis rysunków

Spis tabel

Rozdział 1

Wstęp

1.1 Zakres pracy

Celem pracy było zaprojektowanie, wykonanie i oprogramowanie urządzenia stanowiącego dodatkowe zabezpieczenie antykradzieżowe pojazdu w postaci lokalizatora wykorzystującego system GNSS (*ang. Global Navigation Satellite System*) oraz GSM (*ang. Global System for Mobile Communications*), zdolnego do analizy stylu jazdy kierowcy, a także całego systemu informatycznego, który pozwoliłby na obsłużenie pozyskanych danych. W jego skład wchodzą:

- Strona WWW, umożliwiająca zdalny podgląd danych pochodzących z przypisanych do użytkownika urządzeń.
- Aplikacja serwerowa, która obsługuje zapytania użytkownika oraz zapisuje napływające dane do bazy danych SQLite.

Do dodatkowych wymagań stawianych urządzeniu należą:

- Zapewnienie bezpiecznego szyfrowanego kanału komunikacji deaktywującej tryb alarmu.
- Posiadanie zastępczego źródła zasilania, umożliwiającego pracę przy wyłączonym silniku pojazdu, bądź w razie odłączenia akumulatora.
- Niewielkie wymiary urządzenia w celu umożliwienia łatwego ukrycia w pojeździe.

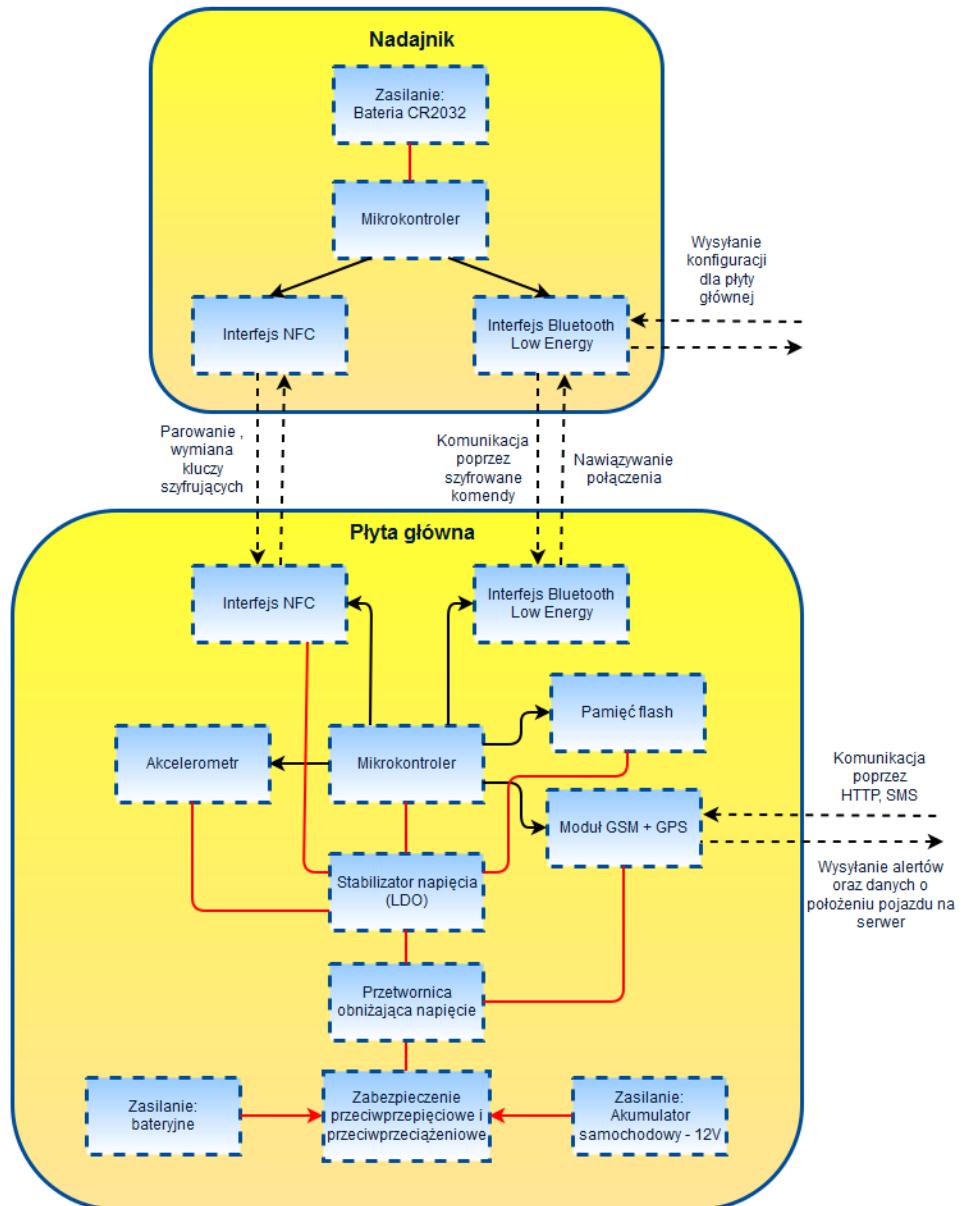
Moduł umożliwia działanie w dwóch trybach. Pierwszy z nich polega na cyklicznym wysyłaniu na serwer pozycji samochodu w trakcie ruchu wraz z m.in. jego prędkością i przyspieszeniem. Dzięki temu możliwy jest zdalny podgląd stylu jazdy kierowcy, co ułatwia sprawowanie kontroli nad flotą pojazdów.

Drugi tryb uaktywnia się w trakcie postoju i stanowi system alarmowego powiadamiania właściciela pojazdu o nieautoryzowanym jego przemieszczeniu w przypadku kradzieży.

W celu zapewnienia bezpieczeństwa, postanowiono rozbić projekt na dwa urządzenia. Jedno z nich – płytka lokalizatora, stanowi rdzeń systemu, umożliwiający lokalizację pojazdu oraz wysyłanie danych na serwer. Drugi moduł stanowi układ deaktywujący, którego zadaniem jest wyłączenie trybu alarmu po odpaleniu samochodu przez upoważnioną do tego osobę. Obie płytki komunikują się ze sobą poprzez protokół Bluetooth Low Energy, zapewniający energooszczędną wymianę danych. Pozwoli to na zasilenie układu deaktywującego z niewielkiej baterii i jego nieprzerwaną pracę nawet przez kilka lat bez konieczności wymiany źródła zasilania. Ponadto, aby umożliwić bezpieczną transmisję niezbędne jest zastosowanie szyfrowania komunikacji. W celu eliminacji ryzyka podsłuchania procesu wymiany klucza szyfrującego, oba urządzenia zostały wyposażone w moduł NFC (*ang. Near Field Communication*), zapewniającego bezkontaktową komunikację na odległość do 10 cm.

1.2 Schemat blokowy urządzeń

Na przedstawionym poniżej rysunku 1.1 zaprezentowano schemat blokowy urządzeń, które stanowią główną część projektu - moduł płyty głównej oraz moduł dezaktywatora.



Rysunek 1.1: Schemat blokowy urządzeń wchodzących w skład systemu.
 Źródło: Twórczość własna

1.3 Istniejące rozwiązania

W ramach pracy przeprowadzono analizę rynkową pod kątem istniejących, ciekawych rozwiązań. Poniżej zaprezentowano trzy najbardziej charakterystyczne z nich.

- Spark Nano 5.0 GPS Tracker

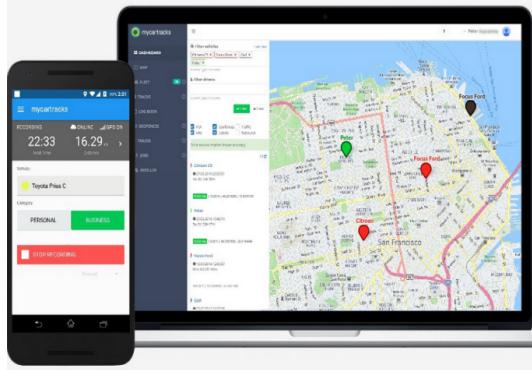
To przenośne urządzenie, posiadające zasilanie baterijne i służące do śledzenia pozycji geograficznej przy pomocy systemu GPS. Zapewnia zdalne powiadomianie użytkownika o lokalizacji urządzenia poprzez sieć CDMA z dokładnością do 2m. Wymiary urządzenia: 64,5 x 40 x 20,5 mm. Urządzenie pozwala na działanie przez ok. 2 tygodnie, przy założeniu pracy przez 1 godzinę dziennie. Producent udostępnia platformę online oraz aplikacje na smartfony z systemem Android oraz IOS, do przedstawiania danych użytkownikowi. Urządzenie domyślnie raportuje położenie co minutę, lecz producent umożliwia zdalne zwiększenie częstotliwości w razie chęci użytkownika. Cena urządzenia: 129,99\$. Wizualizację urządzenia przedstawiono na rysunku 1.2.



Rysunek 1.2: Spark Nano 5.0 GPS Tracker. Źródło: [1]

- MyCarTracks - aplikacja mobilna

Jest to aplikacja na smartphona, która dodaje do niego funkcjonalność trackera GPS. Stanowi rozwiązanie typowo programowe, które wykorzystuje zasoby zawarte w telefonie – moduł GPS, GSM oraz internet. Jest ono proste i tanie, lecz nie pozbawione wad. Ponieważ to aplikacja na telefon, a nie osobne urządzenie, konieczne jest umieszczenie smartphone'a w pojeździe na stałe, jeśli użytkownik chciałby użytkować ją jako zabezpieczenie antykradzieżowe. Ponadto, telefony pobierają stosunkowo dużo energii co wymusza częste ich ładowanie. W rezultacie efektywne ukrycie urządzenia jest utrudnione. Do kosztów rozwiązania należy wliczyć cenę telefonu (używane urządzenie kosztuje ok. 200-300zł) oraz 7\$ za każdy pojazd miesięcznie. Aplikację przedstawiono na rysunku 1.3.



Rysunek 1.3: Aplikacja MyCarTracks. Źródło: [2]

- STI GL300

Jest to kolejne niewielkie, przenośne urządzenie wykorzystujące moduł GPS do lokalizacji. Przekazuje ono informacje o położeniu w czasie rzeczywistym (co 60, 10 lub 5 sekund w zależności od wykupionej taryfy). Producent nie przedstawił informacji o sposobie komunikacji z serwerem, lecz najprawdopodobniej również wykorzystuje sieć GSM. Urządzenie to posiada baterię pozwalającą na ciągłą pracę do 2 tygodni. Przy tym nie ogranicza się ono jedynie do lokalizacji pojazdów dzięki niewielkim wymiarom. Producent wprowadza ciekawe funkcjonalności: powiadamianie poprzez wiadomość sms o osiągnięciu przez pojazd danej pozycji geograficznej, wejście w zdefiniowany obszar czy osiągnięcie pewnej prędkości. Aktualne oraz historyczne dane są przedstawiane użytkownikowi poprzez stronę internetową na mapach od firmy Google. Wymiary urządzenia to zaledwie ok. 5 cm x 2,5 cm x 2 cm. W opcji dodatkowej można dokupić wodooodporną obudowę, pozwalającą na zamontowanie urządzenia na zewnątrz pojazdu. Cena urządzenia to 70\$ oraz od 25\$ do 40\$ miesięcznej opłaty. Wygląd urządzenia pokazano na rysunku 1.4.



Rysunek 1.4: Urządzenie STI GL300. Źródło: [3]

Rozdział 2

Wstęp teoretyczny

2.1 Zastosowane protokoły i systemy

Zgodnie z założeniami pracy, aby zrealizować postawiony urządzeniu cel, zdecydowano się wykorzystać wymienione poniżej protokoły i interfejsy:

- **GSM** - Wykorzystywany do komunikacji zdalnej dalekiego zasięgu (wysyłanie danych na serwer oraz komunikacja z użytkownikiem)
- **GPS** - Wykorzystywany do wyznaczenia lokalizacji pojazdu, a także jego prędkości oraz azymutu.
- **NMEA 0183** - Jest to standard w jakim moduł GPS wysyła dane do mikrokontrolera
- **Bluetooth Low Energy** - Wykorzystywany do komunikacji bliskiego i średniego zasięgu (komunikacja z użytkownikiem oraz z urządzeniem deaktywującym)
- **NFC** - Wykorzystywany do komunikacji bardzo bliskiego zasięgu (do wysłania klucza szyfrującego do oraz komendy deaktywującej z urządzenia deaktywującego)

Wszystkie z powyższych protokołów zostały pokrótko opisane, a następnie krótko zestawione w tym rozdziale.

2.2 System GSM

Poniższy podrozdział powstał na podstawie źródeł [4], [5] oraz [6].

System GSM (*ang. Global System for Mobile Communication*) jest obecnie najpowszechniej stosowanym systemem służącym do komunikacji bezprzewodowej dalekiego zasięgu. Wykorzystywany jest powszechnie do przesyłania głosu oraz serwisów danych. Pomyśl na stworzenie sieci umożliwiającej komunikację głosową wyłonił się we wczesnych latach 70. ubiegłego wieku z opracowywanej w siedzibie Bell Laboratories mobilnej sieci radiowej. Jednakże dopiero dwanaście lat później, w 1982 roku powstał oficjalny komitet normalizacyjny nazwany *Groupe Spécial Mobile*, którego zadaniem było utworzenie jednolitego, otwartego standardu dla telefonii komórkowej.

Pierwotna wersja standardu działała w paśmie 900 MHz (880 - 960 MHz) i umożliwiała jedynie transmisję głosową. Jego kolejna wersja została opublikowana w 1990r. i definiowała ona dodatkowe pasmo 1800 MHz (1710 - 1880 MHz). Ponadto, umożliwiała przesyłanie krótkich wiadomości SMS (*ang. Short Message System*), a także faxu czy transmisję danych. Dalsze prace nad systemem wprowadziły do standardu techniki zwiększające przepustowość transmisji (maksymalna prędkość odbioru - 57.6 kb/s, maksymalna prędkość nadawania - 14.5 kb/s) oraz mechanizm przesyłania danych w pakietach GPRS (*ang. General Packet Radio Service*) z przepustowością 30 - 80 kb/s).

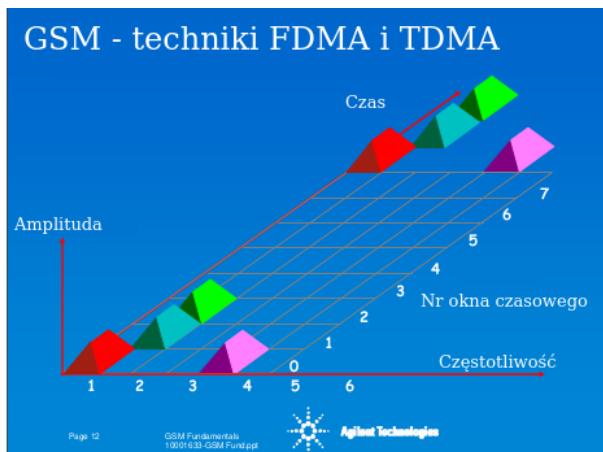
Pomimo pojawienia się na świecie nowszych rozwiązań, takich jak sieci UMTS i LTE, ze względu na ogólną popularność, architektura sieci GSM wciąż jest rozwijana.

System GSM umożliwia skorzystanie z następujących usług:

- Połączenia głosowe - Stanowią one sztandardową funkcjonalność sieci GSM. Jej standard definiuje kodek GSM, który służy do zamiany głosu (skonwertowanego przez mikrofon do napięciowego sygnału analogowego) na postać cyfrową, która jest następnie kompresowana stratnie i transmitowana do odbiorcy. Stosowana jest kompresja na podstawie algorytmu LPC (*ang. Linear Predictive Coding*). Po stronie odbiorcy sygnał jest dekodowany, lecz ze względu na stratność LPC, słyszalny jest zniekształcony, nienaturalny głos rozmówcy.
- Transmisja danych - Umożliwia dostęp do internetu z urządzenia GSM, a także korzystanie z transmisji strumieniowej.

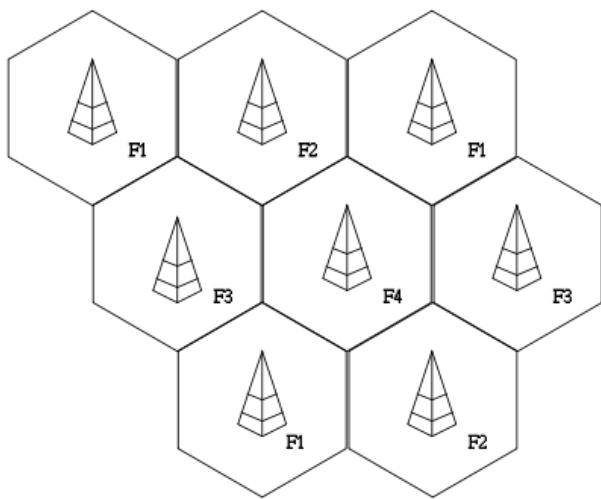
- Wiadomości tekstowe i multimedialne - Usługa przesyłania krótkich wiadomości tekstowych, o długości do 160 znaków, pod warunkiem korzystania jedynie z alfabetu łacińskiego. W przypadku stosowania znaków diakrytycznych maksymalny rozmiar wiadomości spada do 70 znaków. Wiadomości multimedialne (inaczej MMS), umożliwiają przesyłanie zdjęć, filmów czy dźwięków. Ich rozmiar maksymalny jest uzależniony od ograniczeń telefonu oraz operatora.

Jednym z głównych założeń systemu jest możliwość korzystania z niego przez wielu użytkowników jednocześnie. Aby rozwiązać ten problem, postanowiono zastosować technikę zmiany częstotliwości FDMA (*ang. Frequency Division Multiple Access*) oraz okien czasowych TDMA (*ang. Time Division Multiple Access*). Oznacza to, że pasmo częstotliwości GSM jest podzielone na wąskie kanały, o szerokości 200 kHz każdy. Czas użytkowania każdego kanału podzielony jest na 8 okien czasowych. Każde urządzenie ma zatem dostęp do sieci dostrajając się do odpowiedniego kanału w czasie trwania przydzielonego okna czasowego. Przedstawiono to na rysunku 2.1.



Rysunek 2.1: Podział pasma częstotliwości na kanały i okna czasowe. Źródło: [4]

Architektura sieci GSM powstała w oparciu o komórkowy system radiowy, skąd powszechnie stosowana nazwa - sieć komórkowa. Charakteryzuje się ona tym, że obszar terenu, na którym ma być prowadzona komunikacja radiowa dzieli się na tzw. komórki. Każdej komórce przypisana jest stacja bazowa (*ang. Base Transceiver Station*), która stanowi bramę dostępową do sieci. Urządzenie mobilne GSM, takie jak na przykład telefon, znajdując się na obszarze komórki najczęściej odbiera sygnał z więcej niż jednej stacji bazowej, jednakże zawiera połączenie z tą, której sygnał jest najsilniejszy. W razie spadku mocy sygnału stacji z którą urządzenie jest połączone, możliwa jest dynamiczna zmiana połączenia do innego BTS'a.



Rysunek 2.2: Podział obszaru na komórki. Źródło: [6]

Aby móc korzystać z sieci GSM, urządzenie muszą posiadać kartę SIM (*ang. Subscriber Identification Module*). Oprócz przydatnych dla użytkownika wbudowanej pamięci na wiadomości SMS i kontakty, posiada ona unikalny na całym świecie numer identyfikujący użytkownika w sieci. W celu zalogowania się do sieci, urządzenie GSM musi podać ten numer w trakcie nawiązywania połączenia ze stacją bazową.

Każda stacja bazowa wykorzystuje wiele kanałów GSM. Jednakże, aby nie dopuścić do wzajemnego zakłócania się, stacje bazowe z przylegającymi do siebie cel wykorzystują inne ich zestawy. Dodatkowo, w stacjach bazowych stosuje się dwa rodzaje anten - dookółne i kierunkowe o pokryciu 120° . Anteny dookółne pokrywają cały obszar komórki tym samym zbiorem kanałów, natomiast kierunkowe - dla każdego podobszaru wykorzystują ich inny zestaw. Ze względu na skończoną prędkość sygnału radiowego, istnieje również maksymalny promień pojedynczej komórki. W praktyce wynosi on około 35 km. Poniżej istnieje konieczność umożliwienia prowadzenia komunikacji na tak duży zasięg, standard ten nie należy do najbardziej energoszczędnych. W zależności od klasy urządzenia, minimalna moc nadawajnika może wynosić od 1 do 20 mW, natomiast maksymalna nawet do 8 W. Urządzenia GSM mają możliwość dostosowywania mocy transmisji na podstawie mocy sygnału odebranego od stacji bazowej, w celu ograniczenia wysokiego zużycia energii.

Biorąc jednak pod uwagę fakt, iż transmisja przebiega w oknach czasowych, moc średnia jest niższa. Każde okno czasowe trwa $577 \mu s$, a w jego czasie można wysłać jedną z kilku ramek komunikacyjnych. W trakcie każdej z nich można wysłać 148 bitów danych. W przypadku ramki nadawanej w trakcie rozmowy, głos kodowany jest jedynie na 57 bitach. Każde z urządzeń w sieci otrzymuje okno czasowe co 4.615 ms , liczone od początku okna, do rozpoczęcia następnego. Stąd wynika, że w trakcie pojedynczego cyklu nadawania, urządzenie transmituje dane jedynie przez 12.5% czasu. Dla przykładu, pobór prądu przez moduł GSM zastosowany w pracy, w zależności

od odległości do nadajnika, a więc od mocy nadawania, może wówczas wynosić nawet do 1.5 A. Przyjmując napięcie zasilania układu GSM wynoszące 4 V, maksymalna pobierana moc średnia może wynosić:

$$P_{\text{sr}} = U \cdot I \cdot \tau \quad (2.1)$$

$$P_{\text{sr}} = 4V \cdot 1.5A \cdot 0.125 = 0.75W$$

gdzie:

P_{sr} - moc średnia

U - napięcie zasilania układu,

I - natężenie prądu w momencie transmisji

τ - współczynnik wypełnienia impulsu (czas trwania okna czasowego podzielony przez czas pomiędzy oknami czasowymi)

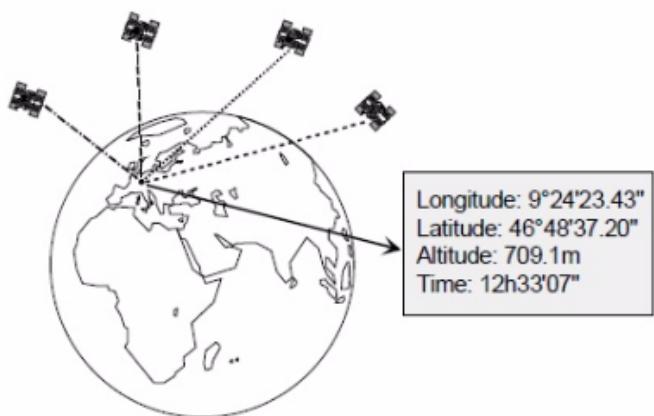
Wartość mocy średniej wynosząca 0.75 W odpowiada ciąglemu zużyciu prądu rzędu 187.5 mA przy napięciu zasilania układu rzędu 4 V.

2.3 System GPS

Poniższy podrozdział powstał na podstawie źródeł [7] oraz [8].

System GPS (*ang. Global Positioning System*) był historycznie pierwszym systemem nawigacji satelitarnej GNSS (*ang. Global Navigation Satellite System*). Powstał w wyniku prac w Departamencie Obrony Stanów Zjednoczonych i jest w pełni własnością rządu tego kraju. Oprócz niego istnieją jeszcze rosyjski GLONASS, a od niedawna europejski Galileo oraz chiński Beidou. Ostatnie dwa systemy satelitarne nie są jeszcze w pełni funkcjonalne, stanowią raczej systemy o zasięgu regionalnym niż globalnym.

Głównym elementem składowym systemów nawigacji satelitarnej, są jak sama nazwa wskazuje satelity. Poruszają się one po ściśle określonych, stałych orbitach, które są tak dobrane, aby z dowolnego punktu na globie, w dowolnym momencie była możliwość odebrania sygnału z co najmniej czterech z nich. Przedstawia to rysunek 2.3.

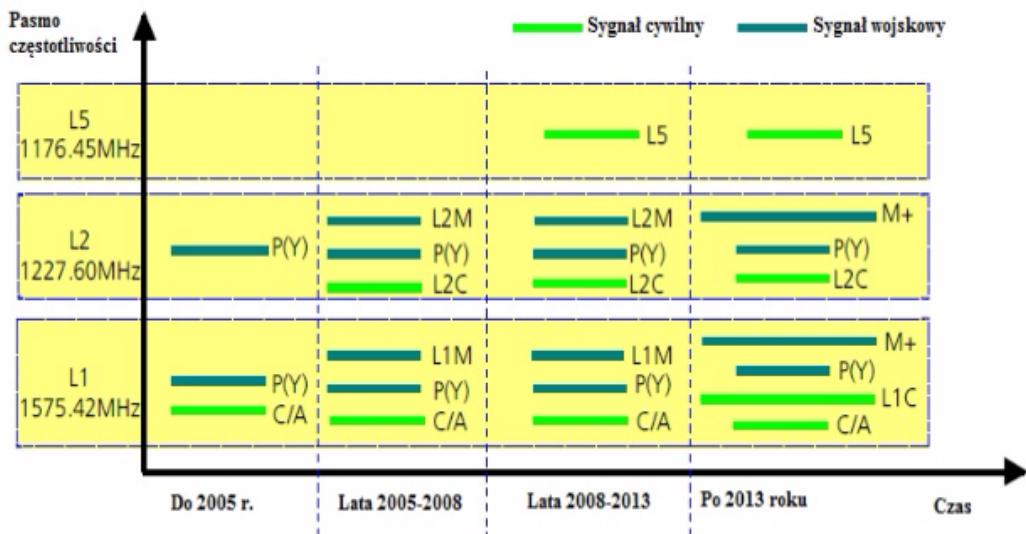


Rysunek 2.3: Model działania systemu GPS. Źródło: [7]

Podstawę w systemach GNSS stanowi czas. Każdy z satelitów posiada 4 zegary atomowe, które stanowią najdokładniejsze źródło czasu znane ludzkości. Zegary te posiadają błąd rzędu 1 sekundy po upływie najwcześniej 30000 lat. Dodatkowo, są one co pewien czas synchronizowane ze źródłami na Ziemi.

Zadaniem każdego z satelitów jest nadawanie w formie rozgłoszeniowej sygnału, w którym zawarta jest wiadomość o jego lokalizacji na orbicie oraz czasie w momencie wysyłania wiadomości. Sygnał ten nadawany jest drogą radiową więc jego prędkość jest równa prędkości światła. Po dotarciu na Ziemię jest on bardzo słaby, przez co praktycznie niemożliwe jest jego odebranie wewnętrz budynków, a w pobliżu wysokich obiektów dokładność lokalizacji spada. Najdokładniejsze wyniki wyznaczania pozycji można osiągnąć na otwartej przestrzeni. Podstawowy sygnał przesyłany jest na fali nośnej o częstotliwości 1575.42 MHz, która nosi nazwę L1.

Ponadto, sygnał GPS (a także pochodzący z innych systemów lokalizacji satelitarnej) łatwo poddaje się zakłóceniom w momencie przejścia przez jonasferę, bowiem fala elektromagnetyczna ulega na niej załamaniu, przez co zmienia swój tor i droga przebycia jest wydłużona. W efekcie, pomiary odległości od odbiornika do satelity, niezbędne do wyznaczenia lokalizacji przestają być dokładne i pojawia się błąd lokalizacji. Problem ten rozwiązano na 2 sposoby. Pierwszym z nich jest nadawanie sygnału przez satelity na kilku częstotliwościach. Każda z nich, przehodząc przez jonasferę ulega załamaniu, lecz pod innym kątem, przez co sygnały pokonają różne długości drogi przebytej zanim trafią do odbiornika, a tym samym zostaną odebrane w różnych momentach. Dzięki temu, odbiornik jest w stanie wyznaczyć korektę i wyeliminować błąd. Pierwotnie, rozwiązanie to było dostępne jedynie w celach militarnych, lecz od 2005 roku, Departament Obrony Stanów Zjednoczonych udostępnił częstotliwość L2 (1227.60 MHz) do celów cywilnych, a od 2008r. - również L5 (1176.45 MHz). Częstotliwości dostępne w systemie GPS pokazano na rysunku 2.4.

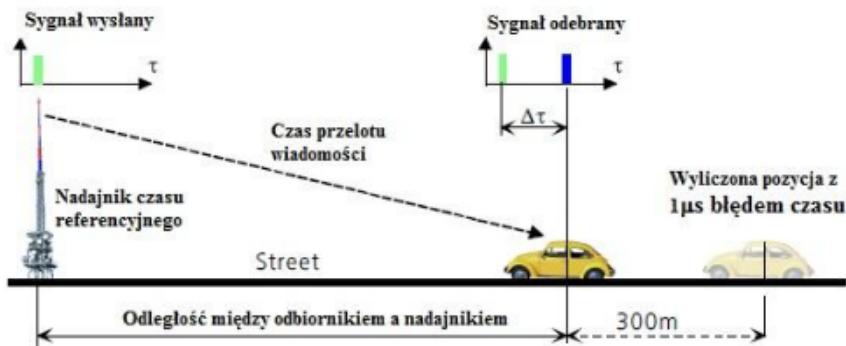


Rysunek 2.4: Zbiór częstotliwości wykorzystywanych w systemie GPS. Źródło: [8]

Druga metoda to wyznaczenie korekty dla przejścia przez jonoferę w stacjach naziemnych, a następnie rozgłaszenie jej w postaci depeszy poprzez sieć stacji bazowych. Rozwiązanie to nosi miano DGPS (*ang. Differential GPS*). Dzięki zastosowaniu tej techniki, dokładność lokalizacji wzrasta z nominalnych 15 m nawet do 10 cm.

Układy GPS, które umożliwiają skorzystanie z któregoś z tych dwóch rozwiązań są jednak kosztowne, więc na rynku cywilnym najpowszechniej stosowane są moduły wykorzystujące jedynie częstotliwość L1. Dzięki temu, za kilkadziesiąt złotych można uzyskać system, który w sprzyjających warunkach charakteryzuje się dokładnością do kilku metrów [8] (Rozdział 7 - Testy urządzenia). W pracy tej uzyskano dokładność systemu rzędu 1 - 2 metrów w terenie otwartym, oraz 5 - 7 metrów idąc chodnikiem wzdłuż wysokich budynków.

Aby zrozumieć zasadę działania systemu GPS proszę wyobrazić sobie sytuację jak na rysunku 2.5:



Rysunek 2.5: Zasada działania systemu GPS. Źródło: [8]

Załóżmy, że w przedstawionym pojeździe znajduje się odbiornik GPS. W pewnym momencie odbiera on sygnał z nadajnika referencyjnego. W sygnale znajduje się informacja o czasie w momencie wysłania wiadomości. Ze względu na skończoną prędkość światła ($c = 299792458m/s$), zostanie ona odebrana przez odbiornik z pewnym opóźnieniem. Wykorzystując ten fakt i znając prędkość transmisji (wartość prędkości światła), można wyznaczyć odległość do nadajnika:

$$D = c \cdot \Delta t \quad (2.2)$$

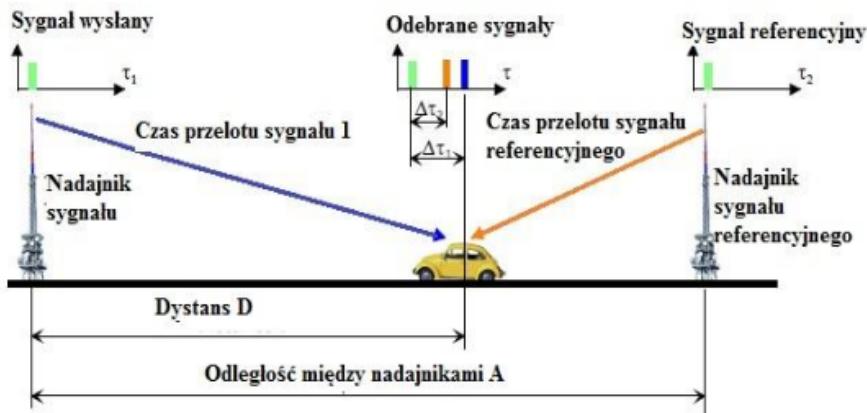
gdzie,

D - odległość między nadajnikiem i odbiornikiem

c - prędkość światła

Δt - różnica czasu między wysłaniem i odebraniem wiadomości

Wynika stąd, że aby wyznaczyć różnicę czasu, odbiornik powinien mieć również własny zegar. Powinien być on przy tym niezwykle dokładny i zsynchronizowany z zegarem w nadajniku, bowiem błąd rzędu $1 \mu s$ powoduje błąd lokalizacji rzędu 300 m. Ponieważ uzyskanie takiej dokładności oraz synchronizacji w każdym odbiorniku jest niemożliwe, należało znaleźć sposób umożliwiający rezygnację z konieczności posiadania przez nie zegara. Przedstawiono go na rysunku 2.6:



Rysunek 2.6: Zasada działania systemu GPS - sygnał referencyjny. Źródło: [8]

Polega ona na zastosowaniu dodatkowego sygnału referencyjnego czasu. Wówczas po odebraniu obu sygnałów (które zostały wysłane w tym samym momencie) otrzymujemy:

$$\begin{cases} \Delta\tau_1 \cdot c = D \\ \Delta\tau_2 \cdot c = A - D \end{cases} \quad (2.3)$$

Gdzie:

$\Delta\tau_1$ - różnica czasu między wysłaniem sygnału z nadajnika, a momentem jego odebrania

$\Delta\tau_2$ - różnica czasu między wysłaniem sygnału z nadajnika referencyjnego, a momentem jego odebrania

A - odległość między nadajnikami

D - odległość od nadajnika do odbiornika

Po odjęciu stronami drugiego równania od pierwszego otrzymamy:

$$(\Delta\tau_1 - \Delta\tau_1) \cdot c = 2D - A \quad (2.4)$$

$$D = \frac{(\Delta\tau_1 - \Delta\tau_1) \cdot c + A}{2}$$

Ponieważ jednak:

$$\begin{cases} \Delta\tau_1 = t - \tau_1 \\ \Delta\tau_2 = t - \tau_2 \end{cases} \quad (2.5)$$

to

$$(\Delta\tau_1 - \Delta\tau_2) = ((t - \tau_1) - (t - \tau_2)) = \tau_2 - \tau_1 \quad (2.6)$$

Gdzie:

t - czas w momencie nadania sygnału przez nadajniki

τ_1 - czas odebrania przez odbiornik sygnału nadanego przez nadajnik

τ_2 - czas odebrania przez odbiornik sygnału nadanego przez nadajnik referencyjny

Powyższe równania prowadzą do wniosku, że zastosowanie dodatkowego nadajnika referencyjnego, zsynchronizowanego z satelitami powoduje eliminację konieczności posiadania zegara w odbiorniku.

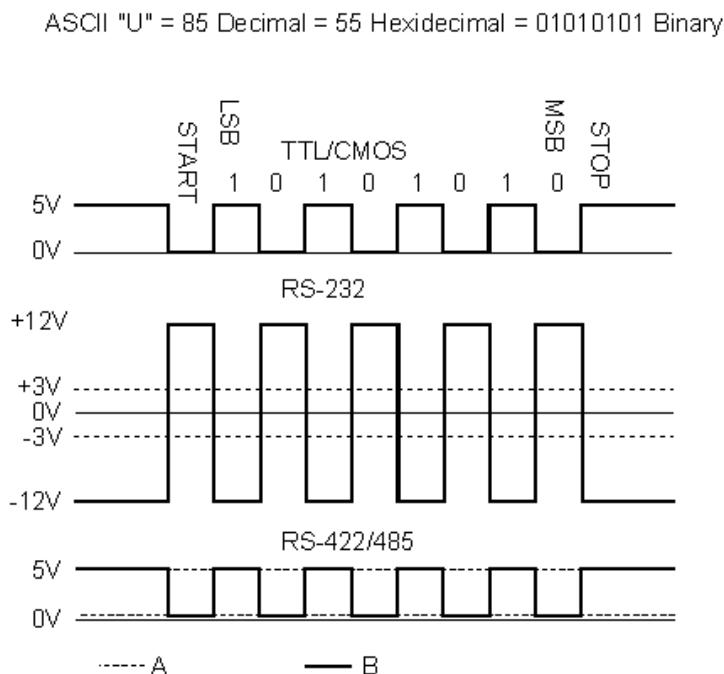
W rzeczywistości, takim nadajnikiem referencyjnym jest inny satelita GPS, bowiem są one ze sobą zsynchronizowane i nadają w dokładnie tym samym momencie. Przy tym, każdy z nich nadaje swoją lokalizację na orbicie więc możliwe jest wyznaczenie odległości między nimi.

Wyznaczona powyżej odległość D dotyczy odcinka (jednej osi współrzędnych), a w rzeczywistości do wyznaczenia lokalizacji niezbędne są trzy osie. Uwzględniając zatem satelitę referencyjnego, do lokalizacji odbiornika potrzeba sygnału z co najmniej 4 satelitów.

2.4 Protokół NMEA 0183

Niniejszy podrozdział powstał na podstawie źródeł [8] oraz [10].

Protokół ten stanowi standard komunikacji między urządzeniami elektronicznymi wykorzystywanymi w urządzeniach morskich, zwłaszcza urządzeniami do lokalizacji i nawigacji. Zawiera on specyfikację elektryczną oraz opis ramek (wiadomości) wymienianych między modułami. Powstał w Stanach Zjednoczonych w *National Marine Electronics Association*. Jego najnowsza, czwarta wersja pochodzi z listopada 2008 roku. Protokół ten pierwotnie wykorzystywał interfejs RS232, lecz w wersji drugiej dokonano jego zmiany na RS422. Interfejsy te różnią się jedynie poziomami napięć przypisanym logicznym wartościom bitów 0 i 1. Oba z nich umożliwiają wysyłanie danych bajt po bajcie. Każdy z nich rozpoczyna się bitem START (stan niski w RS422), który umożliwia odbiornikowi wykrycie początku bajtu i synchronizację. Następnie wysłanych jest kolejno 8 bitów danych, po których przesyłany jest bit parzystości (0 gdy liczba bitów w bajcie danych o wartości 1 jest parzysta lub 1 gdy jest nieparzysta) i na koniec - bit stopu. Przedstawiono to na rysunku 2.7.



Rysunek 2.7: Poziomy napięć i kolejność bitów w interfejsach RS232 i RS422. Źródło: [9]

Na podstawie tego interfejsu, NMEA nabudowała wyższą warstwę protokołu w postaci wiadomości. Każda wiadomość rozpoczyna się symbolem '\$', po którym występuje 2 literowy kod mówiący o typie urządzenia (GP - urządzenie GPS, GN - urządzenie GLONASS) i 3 literowy

kod definiujący typ przesyłanej wiadomości. Po kodzie występuje przecinek, a następnie lista pól danych oddzielonych przecinkami. W obrębie wiadomości, każde pole ma ścisłe określoną funkcję i w razie nie występowania, musi zostać przesłane jako puste. Za ostatnim polem występuje znak '*', po którym znajduje się suma kontrolna, liczona jako funkcja Exclusive Or (XOR) ze wszystkich znaków między '\$', a '*' bez ich uwzględnienia.

Lista zdefiniowanych wiadomości jest bardzo dłuża, jednak w tabeli 2.4 zestawiono najpowszechniej wykorzystywane w odbiornikach GPS.

Tabela 2.1: Najczęściej wykorzystywane wiadomości NMEA0183 w odbiornikach GPS.

Źródło: [8]

GGA	Najczęściej wykorzystywane dane związane z ustalaniem pozycji GPS
GGL	Pozycja geograficzna – długość, szerokość
GSA	Informacje o aktywnych satelitach oraz o jakości połączenia (DOP – ang. <i>Dilution of Position</i>)
GSV	Informacje o satelitach w zasięgu
RMC	Rekomendowane minimum danych GNSS
VTG	Dane o kursie oraz prędkości
ZDA	Wiadomość z aktualnym czasem i datą

W niniejszej pracy, aby zrealizować założenia projektu niezbędne było sparsowanie (przeanalizowanie) danych z dwóch wiadomości nadawanych przez moduł GPS: GGA i VTG. Z wiadomości GGA wyłoniono informacje o długości i szerokości geograficznej, wskaźnikach półkul, statusie wyznaczenia lokalizacji, jakości odebranego sygnału, liczbie satelitów w zasięgu oraz wysokości nad poziomem morza. Wiadomość VTG została wykorzystana w celu uzyskania informacji o kursie (azymucie) oraz prędkości odbiornika. Ich struktury przedstawiono kolejno w tabelach 2.2 i 2.3.

Tabela 2.2: Struktura wiadomości GGA. Źródło: Twórczość własna

Pole	Opis
\$	Symbol początku wiadomości
GP	Typ urządzenia
GGA	Typ wiadomości
130305.743	Czas UTC w formacie hhmmss.sss
4717.115	Szerokość geograficzna w formacie ddmm.mmm
N	Wskaźnik półkuli (N - północna, S - południowa)
00833.912	Długość geograficzna w formacie dddmm.mmm
E	Wskaźnik półkuli (W - zachodnia, E - wschodnia)
1	Wskaźnik informujący o statusie wyznaczania pozycji 0 - pozycja nieustalona 1 - pozycja ustalona na podstawie sygnału z satelitów 2 - pozycja ustalona przy pomocy DGPS 6 - pozycja wyestymowana za pomocą mechanizmu <i>Dead reckoning</i>
08	Liczba satelitów z których odebrano sygnał
0.94	Wskaźnik jakości sygnału (0.5 - najlepsza, 20 - bardzo niska)
00499	Wysokość nad poziomem morza
M	Jednostki wysokości (M - metry)
047	Różnica w wysokości między geoidą (Ziemią), a elipsoidą (przybliżeniem Ziemi)
M	Jednostki wysokości (M - metry)
„	Dane DGPS (pole puste)
0000	Numer identyfikacyjny stacji bazowej DGPS
*	Znak końca danych
58	Suma kontrolna
<CR><LF>	Znak końca wiadomości

Tabela 2.3: Struktura wiadomości VTG. Źródło: Twórczość własna

Pole	Opis
\$	Symbol początku wiadomości
GP	Typ urządzenia
VTG	Typ wiadomości
227.15	Kurs (azymut) w stopniach
T	Pole stałe, zawierające symbol T
„	Kurs magnetyczny (nie zaimplementowany przez producenta)
M	Pole stałe, zawierające symbol M
0.00	Prędkość
N	Pole stałe opisujące jednostki prędkości (N - węzły, K - kilometry na godzinę)
0.00	Prędkość
K	Pole stałe opisujące jednostki prędkości (N - węzły, K - kilometry na godzinę)
A	Tryb pozycjonowania N - brak pozycji A - pozycja na podstawie sygnału z satelitów D - pozycja na podstawie DGPS
*	Znak końca danych
3E	Suma kontrolna
<CR><LF>	Znak końca wiadomości

2.5 Protokół Bluetooth Low Energy

Poniższy podrozdział powstał na podstawie źródeł [11] oraz [8].

Projekt protokołu BLE (*ang. Bluetooth Low Energy*) został zapoczątkowany przez firmę Wibree należącą do grupy Nokia. Celem nadzawanym, który przyświecał jego autorom nie było utworzenie kolejnego protokołu, którego zastosowanie byłoby przesadnie szerokie. Zamiast tego, zdecydowali się oni na zaprojektowanie standardu radiowego, umożliwiającego najniższe możliwe zużycie energii, a przy tym nieskomplikowanego, przez co możliwe byłoby zastosowanie go w systemach o niskich kosztach budowy. Innymi słowy są to założenia idealne dla rynku smartfonów oraz IoT (*ang. Internet of Things*), gdzie urządzenia zasilane są z niewielkich baterii (często CR2032). Dla przykładu, zakup w liczbie 1000 sztuk mikrokontrolera użytego w tej pracy - nRF52832 wyprodukowanego przez Nordic Semiconductor, który posiada wbudowany stos BLE, kosztuje jedynie 9.78 zł za sztukę.

W trakcie prac, projekt został przejęty przez grupę Bluetooth SIG (*ang. Bluetooth Special Interests Group*), zrzeszającą dziesiątki firm i organizacji z wielu dziedzin przemysłu, zainteresowanych wykorzystywaniem i rozwojem protokołu Bluetooth. W roku 2010, Bluetooth Low Energy, znany również jako Bluetooth Smart został włączony do standardu Bluetooth 4.0 obok

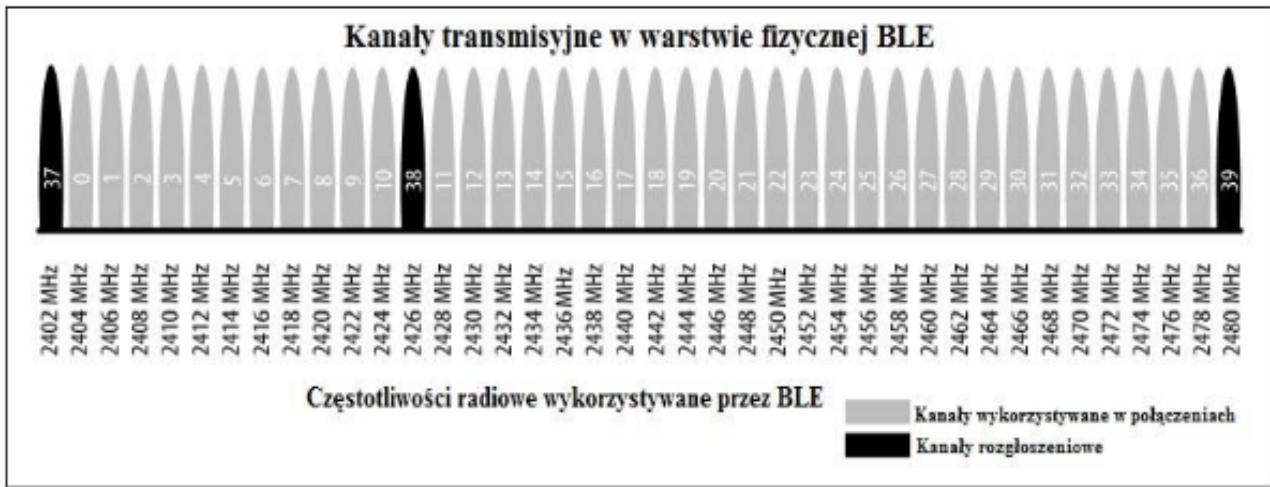
klasycznego protokołu Bluetooth Classic. Nie należy jednakże mylić tych dwóch protokołów komunikacyjnych, ponieważ poza warstwą fizyczną (interfejs radiowy o częstotliwości 2.4 GHz w pasmie ISM) znacznie różnią się w swych założeniach. Bluetooth Classic jest bowiem typowym protokołem umożliwiającym szybką, lecz energochłonną komunikację. W grudniu 2013 roku wprowadzono pierwszą dużą poprawkę do protokołu (Bluetooth 4.1), a rok później dalsze modyfikacje w postaci standardu Bluetooth 4.2.

Bluetooth Low Energy, ze względu na swoje założenie o energoszczędności posiada pewne ograniczenia. Pierwszym z nich jest przepustowość danych. Góra granica prędkości transmisji wynosi 1 Mb/s, jednakże jest to wartość jedynie teoretyczna. W praktyce jest ona obwarowana wieloma ograniczeniami sprzętowymi producentów układów. W standardzie zawarte jest, iż pojedynczy pakiet danych może zawierać maksymalnie 20 bajtów. Ograniczenie sprzętowe wynika tu z częstotliwości wysyłania pakietów. Dla mikrokontrolera Nordic Semiconductor z rodziny nRF51, wynosi ona do 6 pakietów na każdy interwał połączenia. Jest to konfigurowalny parametr, określający odcinek czasu w obrębie którego jeśli nie dojdzie do transmisji pakietu, połączenie zostanie uznane za zerwane. Interwał połączenia może wynosić od 7.5 ms do 4 s. Przy założeniu najmniejszej wartości tego parametru otrzymujemy przepustowość:

$$\text{Przepustowość} = 6 \text{pakietów/interwał} \cdot \frac{1000 \text{ms}}{7.5 \text{ms}} \cdot 20 \text{bajtów} = 15960 \text{bajtów/s} \approx 128 \text{kb/s} \quad (2.7)$$

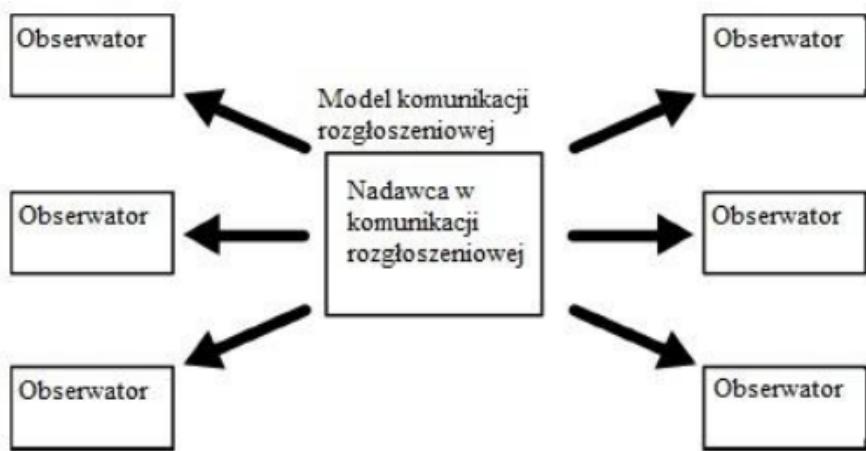
Jak widać, jest to wartość znacznie odbiegająca od 1 Mb/s, jednakże w porównaniu do zysku na zużyciu energii jest to i tak bardzo dobry wynik.

Kolejne ograniczenie to zasięg komunikacji. Oficjalnie, Bluetooth Low Energy posiada zasięg rzędu 50 m. Jest to jednak wartość trudna do uzyskania, silnie zależna od otoczenia (między urządzeniami nie może być przeszkód), mocy transmisji (rekonfigurowalna, im mniejsza tym mniejszy zasięg) oraz liczby innych urządzeń znajdujących się w pobliżu. Duża liczba nadajników BLE wpływa na zajętość kanałów komunikacyjnych i dodatkowo zmniejsza przepustowość łącza. Pasmo częstotliwości (od 2.402 GHz do 2.480 GHz), wykorzystywanej przez protokół podzielone jest na 40 kanałów. Przedstawiono to na rysunku 2.8.



Rysunek 2.8: Struktura pasma 2.4 ISM wykorzystywanego przez Bluetooth Low Energy.
 Źródło: [8]

Protokół Bluetooth Low Energy oferuje dwie możliwości wysyłania danych. Pierwszym z nich jest bezpołączeniowe rozgłaszczenie (*ang. Advertising*). Wówczas, każde z urządzeń wysyła cyklicznie w eter pakiet danych o pojemności do 31 bajtów. Interwał między pakietami może wynosić od 20 ms do 10.24 s. Jest to jednak komunikacja jednokierunkowa. Wysłane w ten sposób dane może odebrać każde urządzenie będące w zasięgu.



Rysunek 2.9: Model komunikacji rozgłoszeniowej. Źródło: [8]

Drugą metodą jest wysyłanie danych będąc w połączeniu. Wówczas komunikacja może być dwustronna. W tym przypadku, BLE definiuje dwa możliwe typy urządzeń. Jedno z nich, które inicjuje połączenie określone jest jako *Central*, natomiast urządzenie akceptujące połączenie - *Peripheral*. Nie występuje przy tym ograniczenie, że urządzenie może mieć tylko jedną rolę. Może ono będąc w połączeniu urządzeniem typu *Peripheral* zainicjować samodzielnie połączenie

z innym odbiornikiem, a więc stać się dla tego połączenia *Central’em*. Ważne jest, że dla danego połączenia, realizowanego typu punkt - punkt, urządzenie posiada tylko jedną rolę.

Standard Bluetooth Low Energy definiuje logiczny podział struktur danych. Główną jednostką są tak zwane serwisy. Są to zgrupowania pewnych funkcjonalności, zwanych charakterystykami. Charakterystyki stanowią podstawowe jednostki komunikacji. Mogą one zawierać tzw. deskryptory, które są krótkimi, zrozumiałymi dla ludzi informacjami, jak na przykład nazwa charakterystyki. Można to porównać do definicji klasy (serwis), zawierającej definicje metod (charakterystyki). Struktura ta zarządzana jest przez serwer GATT (*ang. Generic Attribute Server*). Przedstawiono ją na rysunku 2.10.



Rysunek 2.10: Model struktury danych serwera GATT. Źródło: [8]

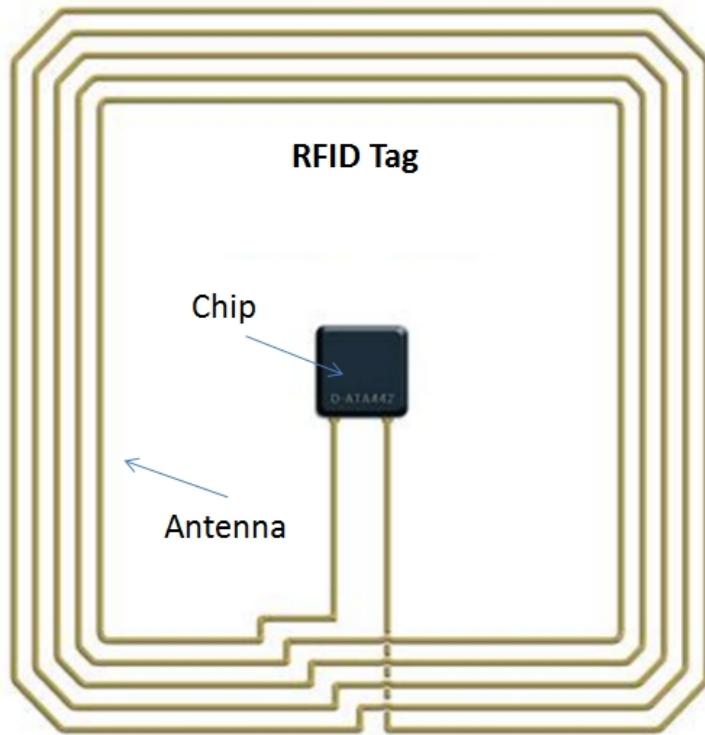
Dla charakterystyk zostały zdefiniowane 4 metody komunikacji. Pierwsza z nich - *Write*, polega na wysłaniu danych z urządzenia typu *Central* do *Peripheral*. Druga - *Read*, umożliwia inicjatorowi połączenia odczytanie danych z urządzenia podległego. Pozostałe 2 metody - *Notify* oraz *Indicate* polegają na wysłaniu danych z urządzenia typu *Peripheral* do urządzenia typu *Central* lub odwrotnie, bez żadnego żądania transmisji ze strony odbiorcy. Różnica polega na tym, że *Indicate* wymaga od odbiornika wysłania potwierdzenia odbioru, a *Notify* nie.

2.6 Interfejs NFC

Poniższy rozdział powstał na podstawie źródeł [12] i [14].

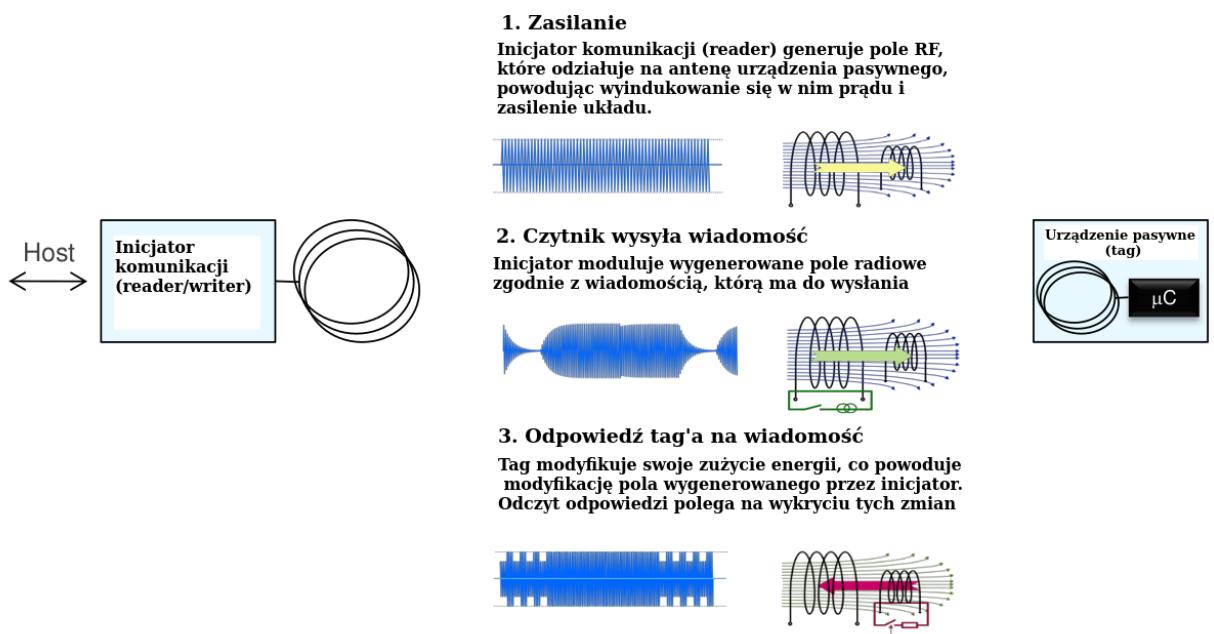
Near Field Communication to protokół radiowy, stanowiący rozszerzenie swego starszego brata - interfejsu RFID. Mimo, iż jest do niego bardzo podobny w wielu aspektach, różni się znacznie pod względem założeń. RFID (*ang. Radio Frequency Identification*), nie stanowi

prawdziwego protokołu komunikacyjnego, bowiem pozwala jedynie na wymianę bardzo krótkich informacji, zwanych identyfikatorami. Urządzenia RFID stanowią bardzo proste układy. Składają się one zazwyczaj z niewielkiego chipa, zawierającego pamięć nieulotną (zazwyczaj do 1 kB) oraz anteny. Przedstawiono to na rysunku 2.11.



Rysunek 2.11: Budowa tag'a RFID. Źródło: [13]

W odróżnieniu od tego, NFC (*ang. Near Field Communication*) stanowi pełnoprawny protokół komunikacyjny. Umożliwia on wymianę długich wiadomości. Został on zbudowany na podstawie RFID i wykorzystuje jego warstwę fizyczną. Tak samo jak w RFID, w NFC można wyróżnić 2 typy urządzeń - pasywne oraz aktywne. Urządzenie pasywne nie generuje swojego własnego pola elektromagnetycznego, w przeciwieństwie do urządzenia aktywnego, które inicjuje komunikację. Ponadto, urządzenie bierne, tak samo jak w przypadku RFID nie posiada nawet własnego źródła zasilania. Gdy znajdzie się ono w polu wygenerowanym przez urządzenie aktywne, w jego antenie wyindukuje się prąd, który jest w stanie zasilić niewielki moduł. Komunikacja zwrotna odbywa się poprzez modyfikację zużycia energii tag'a (urządzenia pasywnego) zgodnie z bitowym wzorcem, który należy wysłać. Dynamiczne zmiany parametrów zużycia powodują pewne zaburzenia wygenerowanego przez inicjatora pola RF. Odczyt danych przez nie polega na odczycie zmian tego pola. Przedstawiono to na rysunku 2.12.



Rysunek 2.12: Zasada działania komunikacji pomiędzy urządzeniem aktywnym i pasywnym.
 Źródło: [14]

Istnieje również możliwość komunikacji pomiędzy dwoma urządzeniami aktywnymi. Wówczas zamiast modyfikować pole wyindukowane, urządzenie odpowiada swoim własnym, wygenerowanym z energii źródła zasilania. Dzięki temu, możliwy do osiągnięcia zasięg komunikacji jest większy.

Na tym w zasadzie podobieństwa między NFC i RFID się kończą. RFID pozwala bowiem jedynie na odpowiedź w postaci swojego unikalnego numeru identyfikacyjnego UID (*ang. Unique Identifier Number*), natomiast moduły NFC stanowią najczęściej urządzenia programowalne, pozwalające na przesłanie dowolnej wiadomości. Ponadto, kolejną różnicą jest fakt, że RFID nie posiada jednego wspólnego standardu komunikacji. Co więcej, nie posiada nawet stałej częstotliwości komunikacji, a jej wybór zależy od producenta sprzętu. Zasięg komunikacji w przypadku RFID również jest zmienny i zależy od częstotliwości sygnału. Dla wartości rzędu 125 - 134.3 kHz wynosi ona do 30 cm (zazwyczaj około 10 cm), dla częstotliwości 13.56 MHz - do 1.5 metra, a w przypadku 433 MHz - nawet do 500 metrów. Ta różnorodność i brak pojedynczego standardu komunikacji stała się główną przyczyną powstania protokołu NFC.

NFC pracuje na ścisłe określonej częstotliwości o wartości 13.56 MHz. Zasięg komunikacji jest niewielki (rzędu 10 cm), a urządzenia mają możliwość emulowania tagów RFID, czyli zachowania się jak one gdy wykryte zostanie pole RF. Dodatkowym atutem NFC jest zdefiniowanie formatu komunikacji pomiędzy urządzeniami - NDEF (*ang. NFC Data Exchange Format*).

Istnieją pewne dobrze znane struktury danych, możliwe do wysłania poprzez NFC.

Są to:

- Wiadomości tekstowe
- Adresy internetowe URI
- Proste komendy
- Podpisy cyfrowe

Organizacją zajmującą się standaryzacją i rozwijaniem NFC jest NFC Forum. Definiuje ona 4 rodzaje urządzeń pasywnych:

1. Typ 1

- Bazuje na specyfikacji ISO-14443A
- Może być tylko do odczytu lub mieć zdolność do zapisu i odczytu
- Rozmiar pamięci od 96 B do 2 kB
- Prędkość komunikacji - 106 kb/s
- Brak ochrony przed kolizją pól

2. Typ 2

- Bazuje na specyfikacji ISO-14443A
- Może być tylko do odczytu lub mieć zdolność do zapisu i odczytu
- Rozmiar pamięci od 96 B do 2 kB
- Prędkość komunikacji - 106 kb/s
- Zapewnia mechanizm ochrony przed kolizją

3. Typ 3

- Bazuje na specyfikacji ISO-18092 i JS-X-6319-4
- Może być tylko do odczytu lub mieć zdolność do zapisu i odczytu
- Rozmiar pamięci do 1 MB
- Prędkość komunikacji - 212 lub 424 kb/s
- Zapewnia mechanizm ochrony przed kolizją

4. Typ 4

- Bazuje na specyfikacji ISO-18092 i JS-X-6319-4
- Może być tylko do odczytu lub mieć zdolność do zapisu i odczytu
- Rozmiar pamięci: 2, 4 lub 8 kB
- Prędkość komunikacji - 106, 212 lub 424 kb/s
- Zapewnia mechanizm ochrony przed kolizją

2.7 Podsumowanie

We wcześniejszych podrozdziałach dokonano krótkiej analizy każdego z protokołów i interfejsów wykorzystanych w pracy. Niniejszy podrozdział stanowi ich podsumowanie ze wskazaniem najważniejszych cech, ograniczeń i możliwości wykorzystania. Z rozważań wykluczono jednakże protokół NMEA 0183 ze względu na fakt, iż jest to jedynie narzucony przez producentów modułów GPS standard komunikacji.

*Tabela 2.4: Podsumowanie cech systemów i protokołów GSM, GPS, BLE oraz NFC.
 Źródło: Twórczość własna*

Parametr	GSM	GPS	BLE	NFC
Właściwości	<ul style="list-style-type: none"> - Ogromny zasięg, dzięki rozbudowanej sieci stacji naziemnych - Możliwość odbioru i transmisji - Prędkość rzędu 57.6 kb/s (odbiór) i 14.5 kb/s (transmisja) - Wysoki pobór prądu (w szczycie do 1.5 A) 	<ul style="list-style-type: none"> - Zasięg globalny - Tylko do odczytu - 50 bit/s - Średni pobór prądu (ok. 30 mA w trakcie śledzenia pozycji) 	<ul style="list-style-type: none"> - Średni zasięg (do 50 m) - Możliwość odbioru i transmisji - Około 125 kb/s - Niski pobór prądu (ok. 7 - 14 mA nateżenia chwilowego w trakcie transmisji) 	<ul style="list-style-type: none"> - Bardzo bliski zasięg (do 10 cm) - Możliwość odbioru i transmisji - Około 106 kb/s - Pobór energii rzędu 100 mA w trybie inicjatora, 0 mA w trybie pasywnym
Możliwości wykorzystania	<ul style="list-style-type: none"> - Rozmowy głosowe - Wiadomości SMS - Dostęp do internetu 	<ul style="list-style-type: none"> - Odczyt lokalizacji - Bardzo dokładny odczyt prędkości - Bardzo dokładne źródło czasu 	<ul style="list-style-type: none"> - Energooszczędna transmisja danych 	<ul style="list-style-type: none"> - Bezpieczna, bezkontaktowa transmisja danych - Parowanie urządzeń - Wymiana kluczy szyfrujących

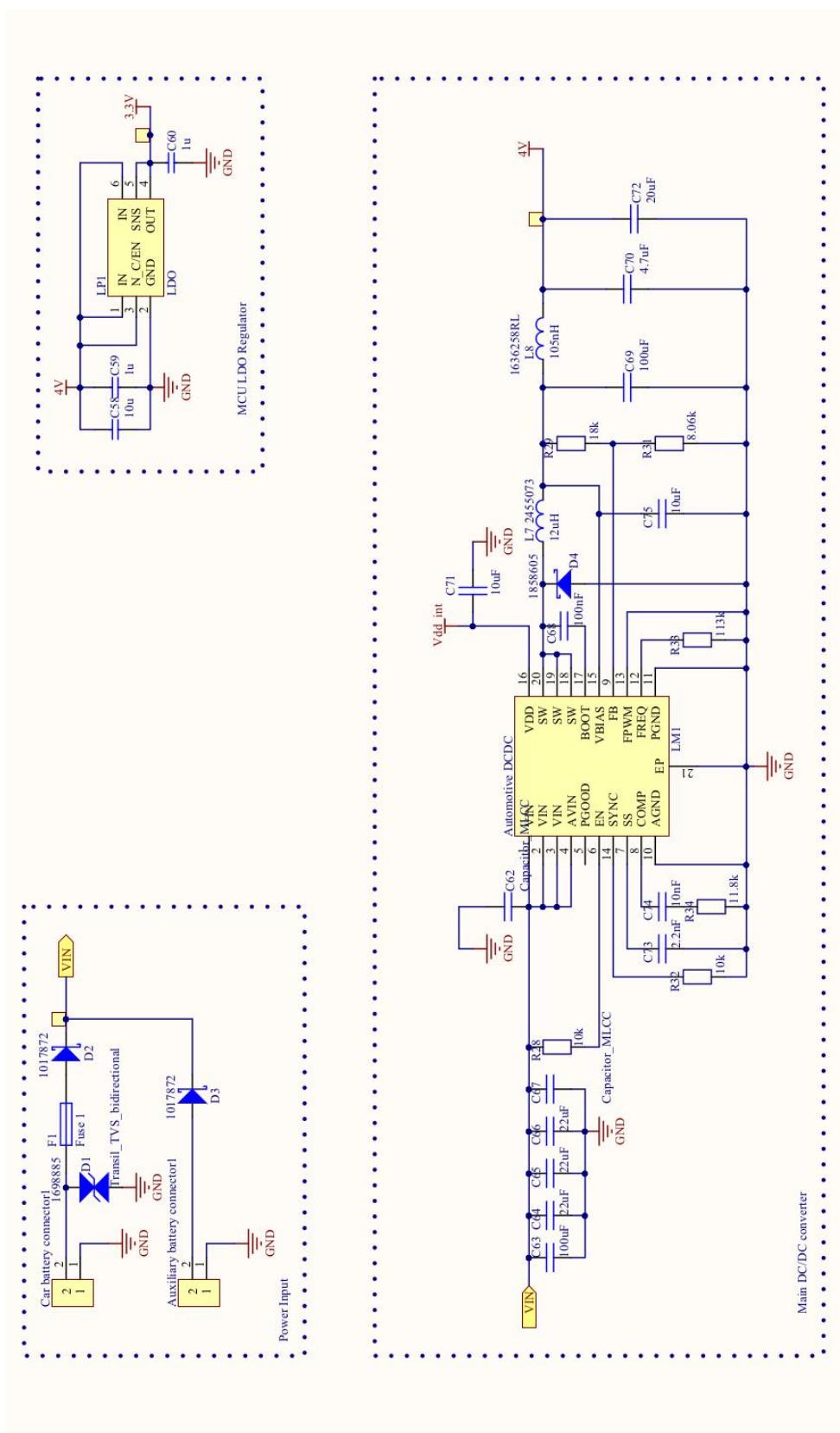
Rozdział 3

Schematy elektroniczne urządzeń

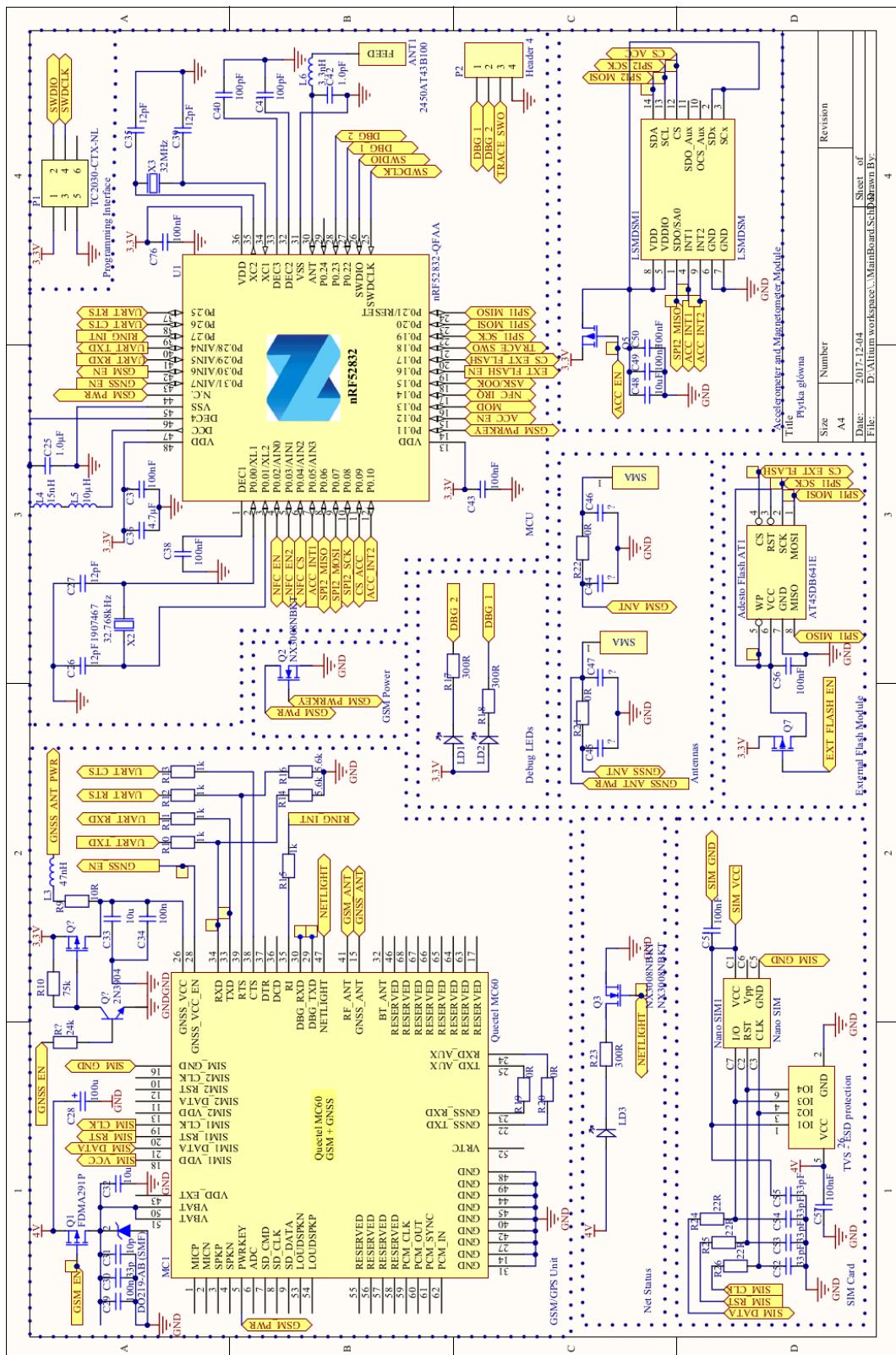
3.1 Urządzenie lokalizujące

Ze względu na poziom skomplikowania układu, schemat elektroniczny musiał zostać rozbity na podschematy. W urządzeniu lokalizującym można wyróżnić trzy znaczące moduły elektroniczne, realizujące odpowiednie funkcje. Są to:

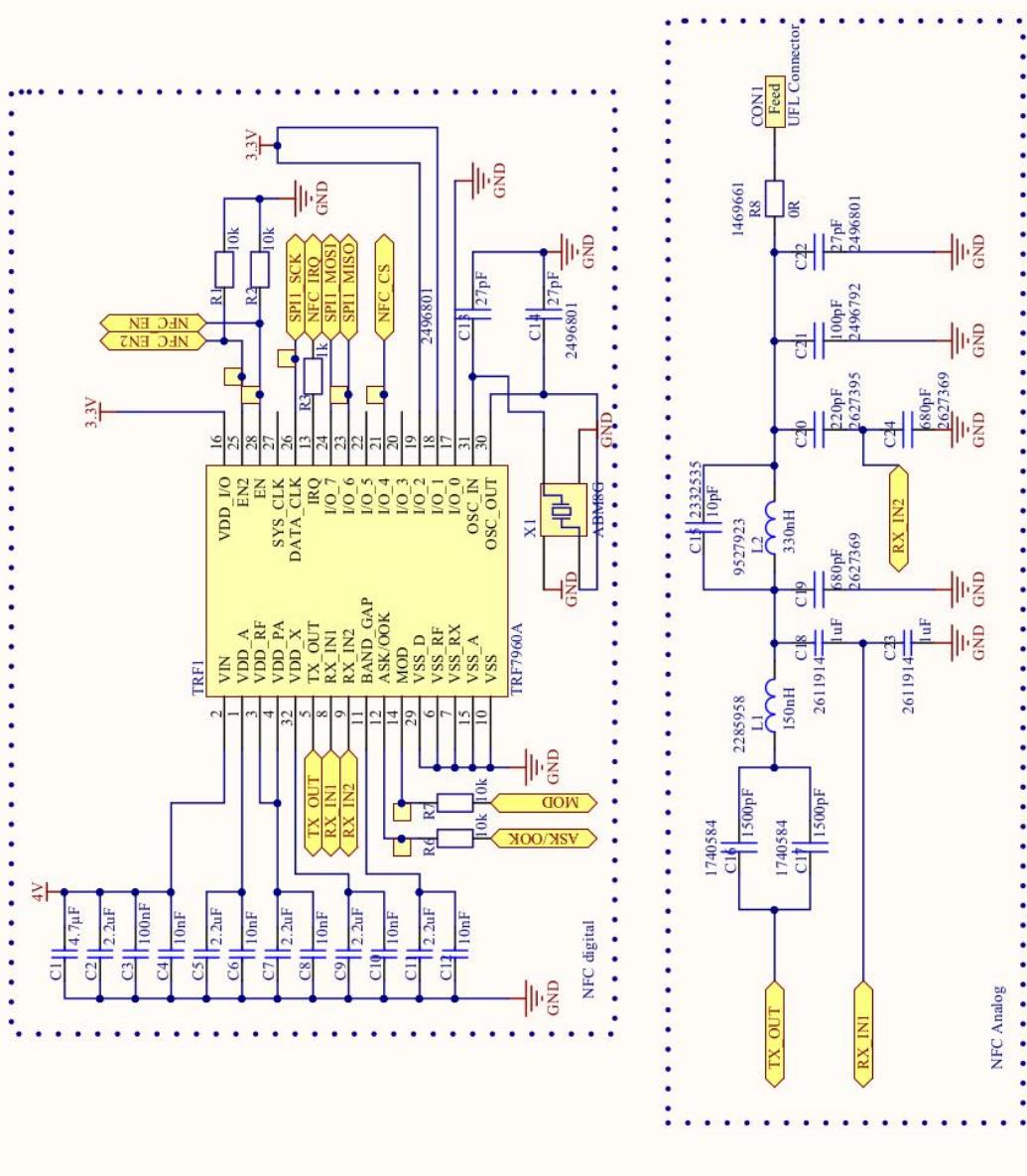
- Moduł zasilania, przedstawiony na rysunku 3.1
- Moduł funkcjonalny, przedstawiony na rysunku 3.2
- Moduł NFC, przedstawiony na rysunku 3.3



Rysunek 3.1: Schemat modułu zasilania urządzenia lokalizującego.
 Źródło: Twórczość własna



Rysunek 3.2: Schemat modułu funkcjonalnego urządzenia lokalizującego.
 Źródło: Twórczość własna.



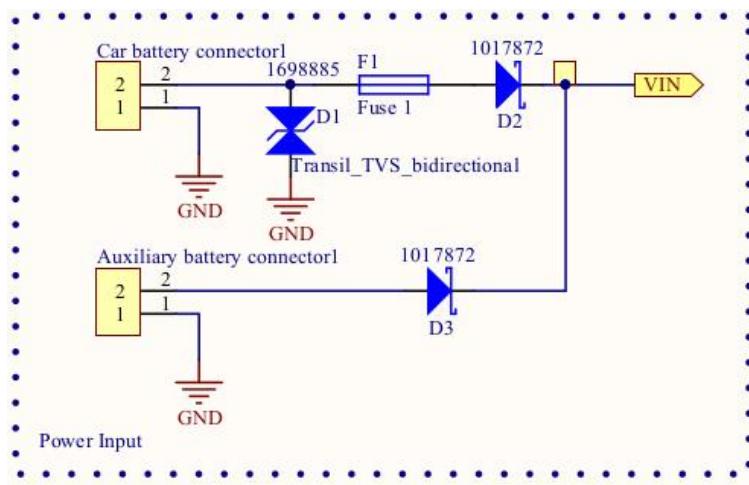
Rysunek 3.3: Schemat modułu NFC urządzenia lokalizującego.

Źródło: Twórczość własna

3.1.1 Schemat zasilania

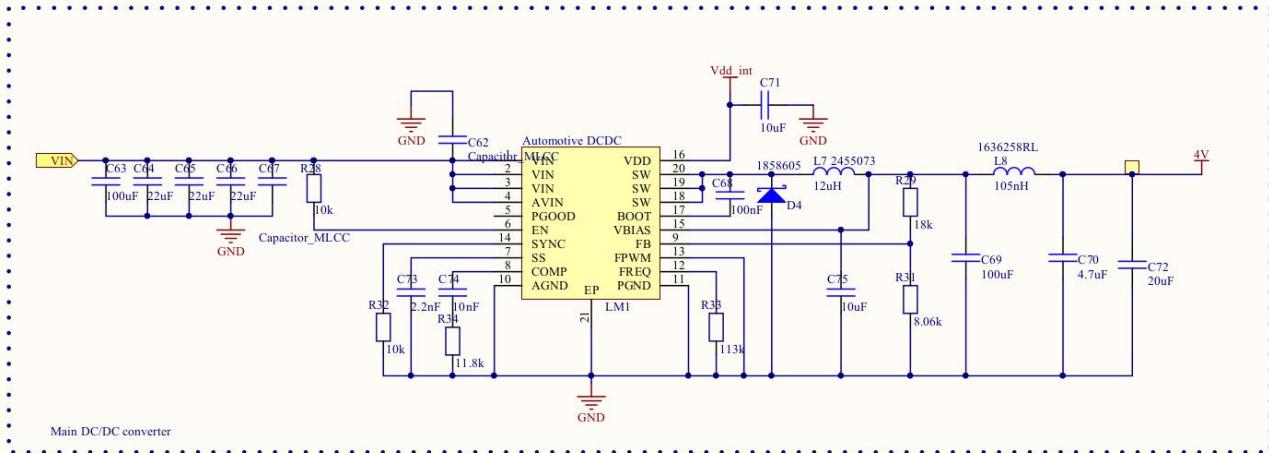
Akumulator samochodowy jest bardzo wygodnym źródłem zasilania układów elektronicznych, lecz gdy są one niewłaściwie zaprojektowane, bywa on dla nich zabójczy. Bliska odległość do alternatora i innych urządzeń indukcyjnych powoduje generowanie silnych zakłóceń na linii zasilającej. Niekiedy "szpilki" napięciowe osiągają wartość rzędu 100 V. Z tego powodu należy stosować transile – diody zabezpieczające. Powodują one ograniczenie zbyt dużego napięcia do pewnej maksymalnej wartości. W przypadku zastosowanego przeze mnie komponentu wynosi

ono 24.4 V. Zabezpieczenie przeciążeniowe stanowi bezpiecznik samochodowy o wartości 4 A. Ponieważ jednym z wymagań układu jest możliwość zasilania baterii pojazdu, konieczne jest zastosowanie dodatkowego przyłącza zasilania. Urządzenie można zasilić dowolną baterią o napięciu od 4 V do 38 V i wydajności prądowej co najmniej 3 A w szczytowym. Ze względu na prawdopodobieństwo wystąpienia różnic napięć pomiędzy dodatkową baterią, a akumulatorem samochodu i wiążącym się z tym przepływem prądu z jednego źródła do drugiego, konieczne jest zastosowanie diód zabezpieczających przed rozładowaniem baterii przez akumulator (gdy napięcie akumulatora jest niższe niż napięcie baterii) lub mogącym doprowadzić baterię do zniszczenia doładowywaniem jej bezpośrednio z akumulatora (gdy napięcie baterii jest niższe od napięcia akumulatora). W trakcie projektowania, zdecydowano się na zastosowanie diód Schottky'ego ze względu na ich niski spadek napięcia (0.2 V - 0.55 V w zależności od natężenia prądu) oraz szybki czas przełączania ze stanu zaporowego do przewodzenia (ograniczenie krótkotrwałych zaników zasilania przy wyłączaniu pojazdu). Na rysunku 3.4 przedstawiono część wejściową dla zasilania całej płytki.



Rysunek 3.4: Schemat modułu zasilania wejściowego urządzenia lokalizującego.
Źródło: Twórczość własna

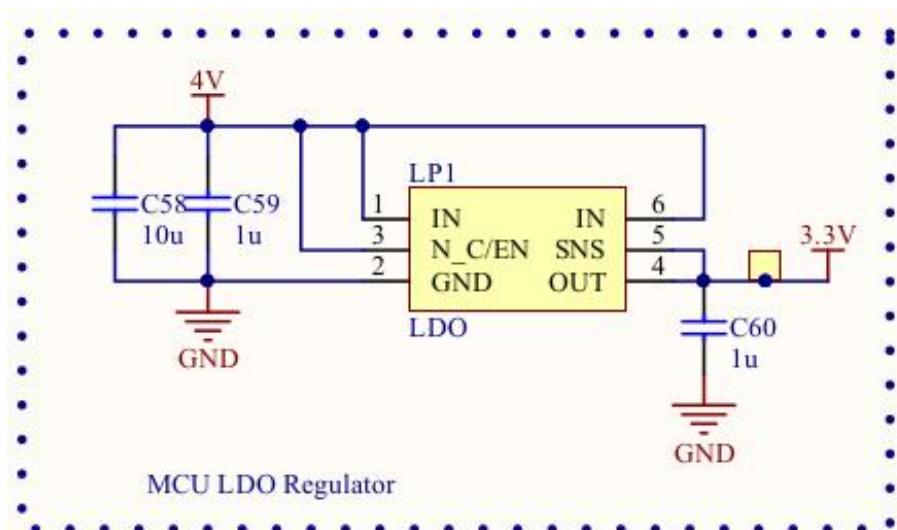
Niestety, często napięcie wejściowe, nawet po zadziałaniu zabezpieczenia w postaci transila, jest nadal zbyt duże dla zwykłych układów zasilających. Stąd konieczne jest stosowanie przetwornic impulsowych klasy automotive, które mogą być zasilane napięciem wejściowym do kilkudziesięciu woltów. Schemat wykorzystanej przetwornicy przedstawiono na rysunku 3.5.



Rysunek 3.5: Schemat przetwornicy impulsowej modułu zasilania urządzenia lokalizującego.
 Źródło: Twórczość własna

Zastosowana w urządzeniu przetwornica umożliwia zasilanie napięciami od 4 V do 38 V. Wybrano ją ze względu na niewielką liczbę, w porównaniu do innych modułów, zewnętrznych komponentów, niezbędnych do jej działania, a także wysoką sprawność rzędu od 85% do 90% w zależności od chwilowego natężenia prądu. Wytwarza ona na wyjściu napięcie o wartości 4V, którym zasilany jest moduł GSM oraz dalszy stopień obniżania napięcia.

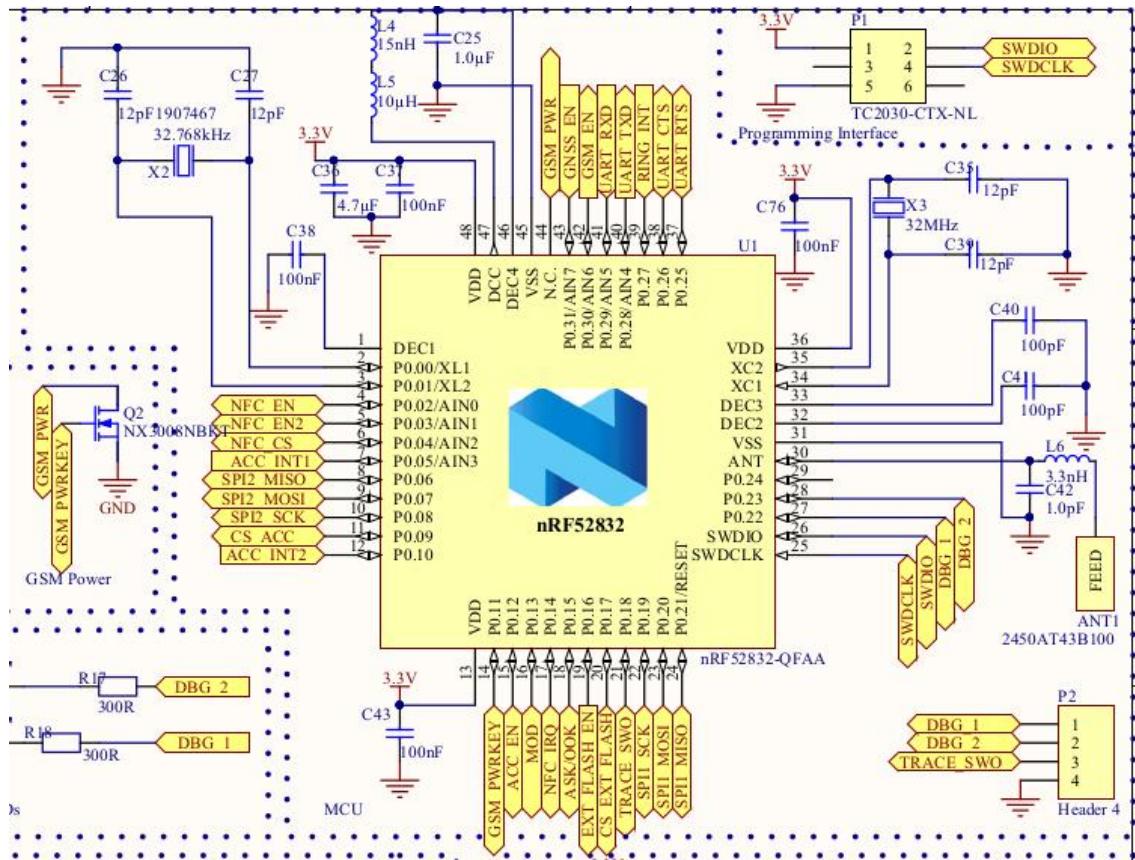
Ostatni stopień zasilania generuje z napięcia wyjściowego z przetwornicy napięcie o wartości 3.3V. Jest ono niezbędne do zasilania układów mikrokontrolera, pamięci flash, akcelerometru oraz układu GPS. Szacowany pobór prądu przez te układy wynosi ok. 200 mA w szczytce, stąd dla bezpieczeństwa wykorzystano stabilizator napięcia LDO (*ang. Low Dropout Stabilizer*) o maksymalnym natężeniu wyjściowym 0.5 A. Jego schemat przedstawiono na rysunku 3.6.



Rysunek 3.6: Schemat stabilizatora napięcia modułu zasilania urządzenia lokalizującego.
 Źródło: Twórczość własna

3.1.2 Moduł mikrokontrolera

Serce urządzenia stanowi mikrokontroler nRF52832 firmy Nordic Semiconductor. Układ ten posiada 32 bitowy rdzeń Cortex-M4 zaprojektowany przez firmę ARM, sprzętową jednostkę FPU, 512 kB wewnętrznej pamięci Flash oraz 64 kB pamięci RAM. Zdecydowano się na wykorzystanie tego mikrokontrolera ze względu na kilka czynników. Pierwszym z nich jest jego wyposażenie - posiada wbudowany układ radiowy działający na częstotliwości 2.4 GHz i umożliwiający komunikację w standardzie Bluetooth Low Energy, ANT lub wykorzystanie własnego protokołu. Dodatkowym atutem tego mikrokontrolera jest wyposażenie w sprzętowy interfejs NFCT, umożliwiający wykorzystanie modułu jako tag (urządzenie podrzędne) w komunikacji poprzez interfejs NFC. Ponadto ma bardzo duże możliwości obliczeniowe – wewnętrzny zegar 64 MHz umożliwia bardzo szybkie wykonywanie zaprogramowanych zadań i szybki powrót do trybu oszczędzania energii. Zużycie energii przez ten procesor jest bardzo niewielkie. W trakcie wykonywania programu pobór prądu wynosi $58 \mu A /MHz$ gdy kod wykonywany jest z pamięci flash, natomiast w trybie oszczędzania energii pobór spada do ok $1.9 \mu A$. Ostatnim i być może najważniejszym czynnikiem decydującym na wybranie tego układu jest posiadane przez autora doświadczenie zawodowe w programowaniu układów od tego producenta, a zatem bardzo dobra znajomość jego możliwości i SDK (*ang. Software Development Kit*). Schemat mikrokontrolera przedstawiono na rysunku 3.7.



Rysunek 3.7: Schemat modułu mikrokontrolera w urządzeniu lokalizującym.

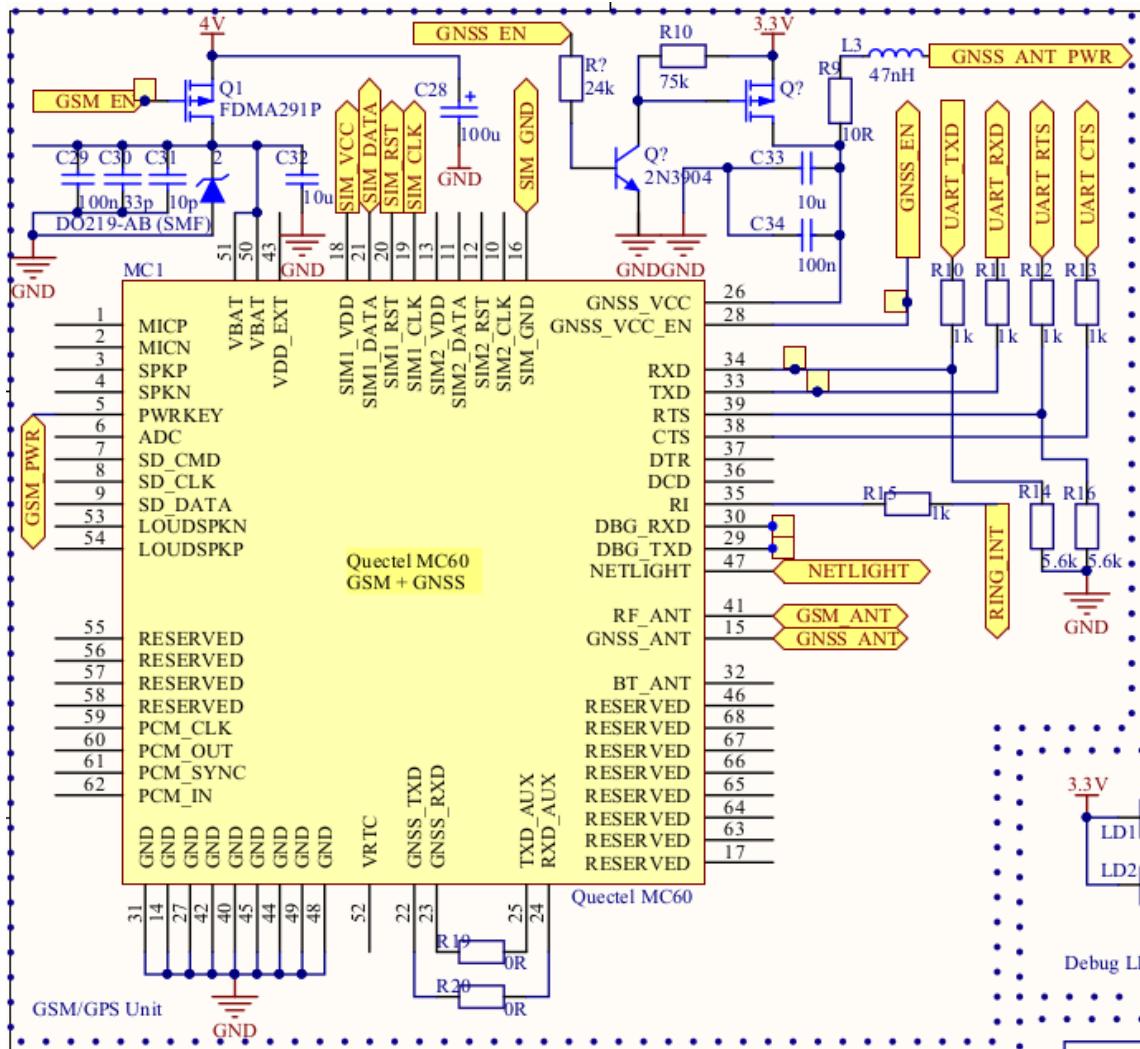
Źródło: Twórczość własna

3.1.3 Moduł GSM i GPS

Jako moduł realizujący funkcję lokalizacji i głównej transmisji w urządzeniu wybrano układ Quectel MC60. Stanowi on połączenie modułu GSM oraz GPS w jednym chipie. Umożliwia wykorzystanie wielu protokołów, takich jak: TCP/IP, UDP, FTP, PPP, HTTP czy NTP. Ponadto możliwe jest odbieranie i wysyłanie danych w postaci krótkich wiadomości SMS. Układ posiada niewielkie wymiary: 18.7 mm x 16 mm x 2.1 mm, dzięki czemu możliwe jest zmniejszenie rozmiarów całego urządzenia. Zużycie energii wynosi:

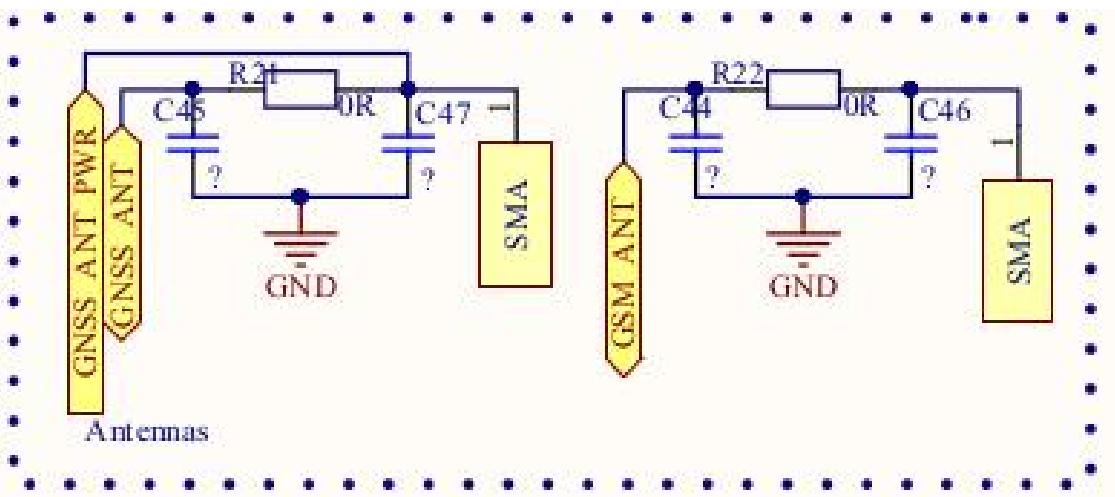
- Około 25 mA gdy działa jedynie moduł GPS
- Do 1.5 A w trakcie transmisji danych poprzez sieć GSM

Ponadto, kombinacja tych dwóch systemów umożliwia wykorzystanie funkcjonalności AGPS. Polega ona na podaniu do modułu GPS zgrubnych danych o położeniu satelitów, pobranych z sieci GSM. Dzięki temu, ustalenie własnej lokalizacji, nawet po długotrwałym braku zasilania, trwa ok. sekundy (tzw. *warm start*). Schemat modułu GSM i GPS przedstawiono na rysunku 3.8.



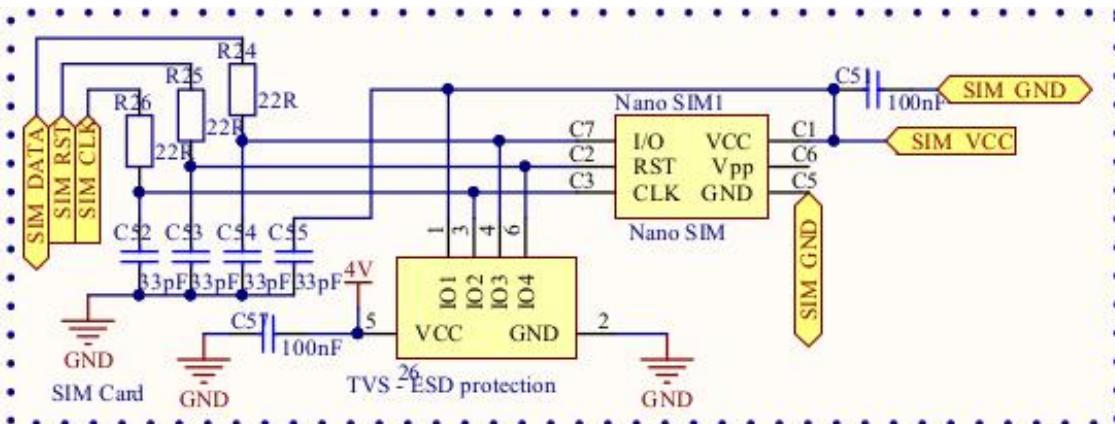
Rysunek 3.8: Schemat modułu układu GSM i GPS w urządzeniu lokalizującym.
 Źródło: Twórczość własna

W celu zwiększenia niezawodności działania urządzenia, zdecydowano zastosować zewnętrzne anteny GSM i GPS, poprawiające jakość sygnału. Dodatkowo, w celu dalszej jej zwiększenia, antena GPS jest anteną aktywną. Oznacza to, że dostarczane jest do niej dodatkowe zasilanie, powodujące wzmacnienie odebranego sygnału. Schemat anten przedstawiono na rysunku 3.9. Zawarte na nim znaki zapytania zamiast wartości pojemności kondensatorów oznaczają, że należy je dobrać po złożeniu płytki i przebadaniu jej pod kątem jak najlepszego dopasowania impedancji. Na etapie uruchomienia układu okazało się, iż kondensatory te nie są niezbędne w celu poprawnego działania urządzenia.



Rysunek 3.9: Schemat modułu anten dla GSM i GPS w urządzeniu lokalizującym.
 Źródło: Twórczość własna

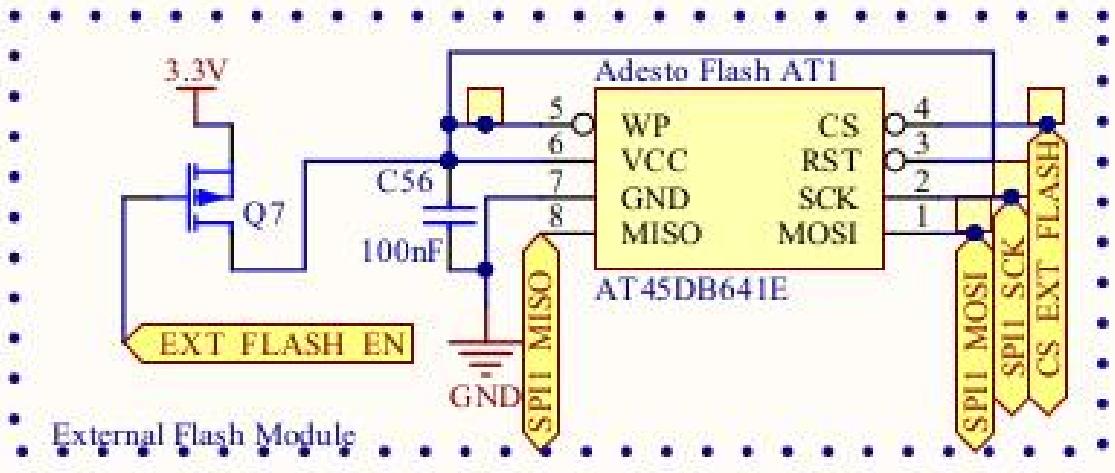
Ostatnią częścią układu GSM jest połączenie modułu z kartą SIM, umożliwiającą zalogowanie do sieci. Przedstawiono je na rysunku 3.10. Widać na nim układ TVS, który jest odpowiedzialny za zabezpieczenie wrażliwej elektroniki w karcie SIM przed wyładowaniami statycznymi ESD (ang. **Electrostatic discharge**).



Rysunek 3.10: Schemat modułu karty SIM w urządzeniu lokalizującym.
 Źródło: Twórczość własna

3.1.4 Moduł pamięci flash

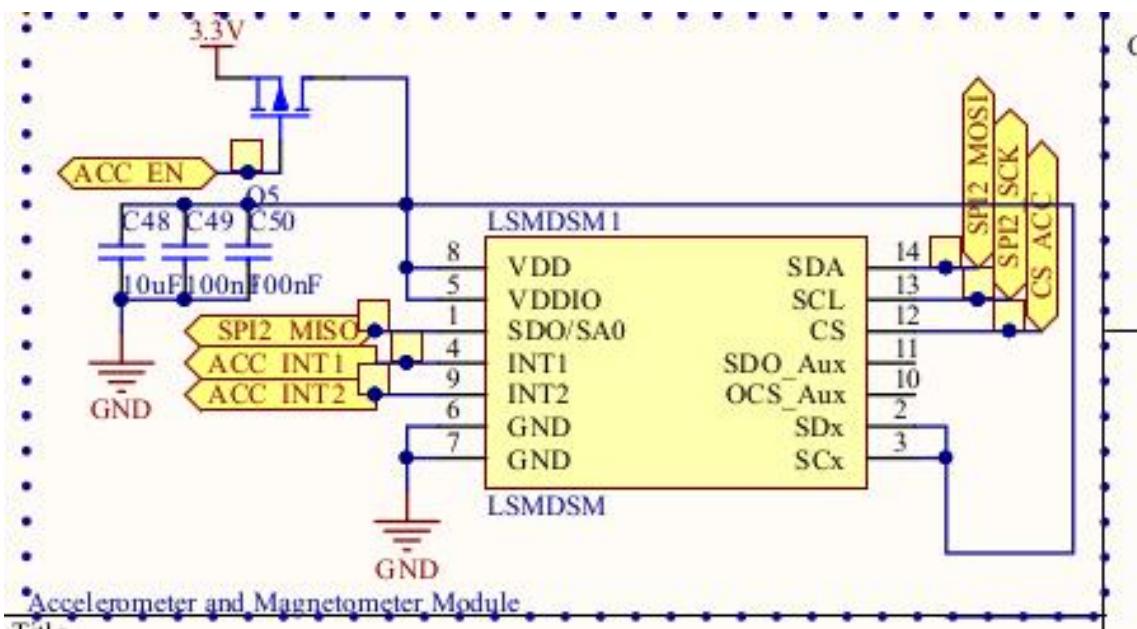
Wewnętrzna pamięć flash mikrokontrolera jest niewystarczająca, aby przechowywać w niej trasy wraz z parametrami jazdy. Stąd też pojawia się konieczność zastosowania zewnętrznego układu pamięci nieulotnej. Zastosowana w urządzeniu pamięć flash posiada pojemność 8 MB, co umożliwia przechowywanie wielu długich tras wraz z dodatkowymi parametrami je opisującymi. Schemat podłączenia pamięci w urządzeniu lokalizującym pokazano na rysunku 3.11.



Rysunek 3.11: Schemat modułu pamięci flash w urządzeniu lokalizującym.
 Źródło: Twórczość własna

3.1.5 Moduł akcelerometru

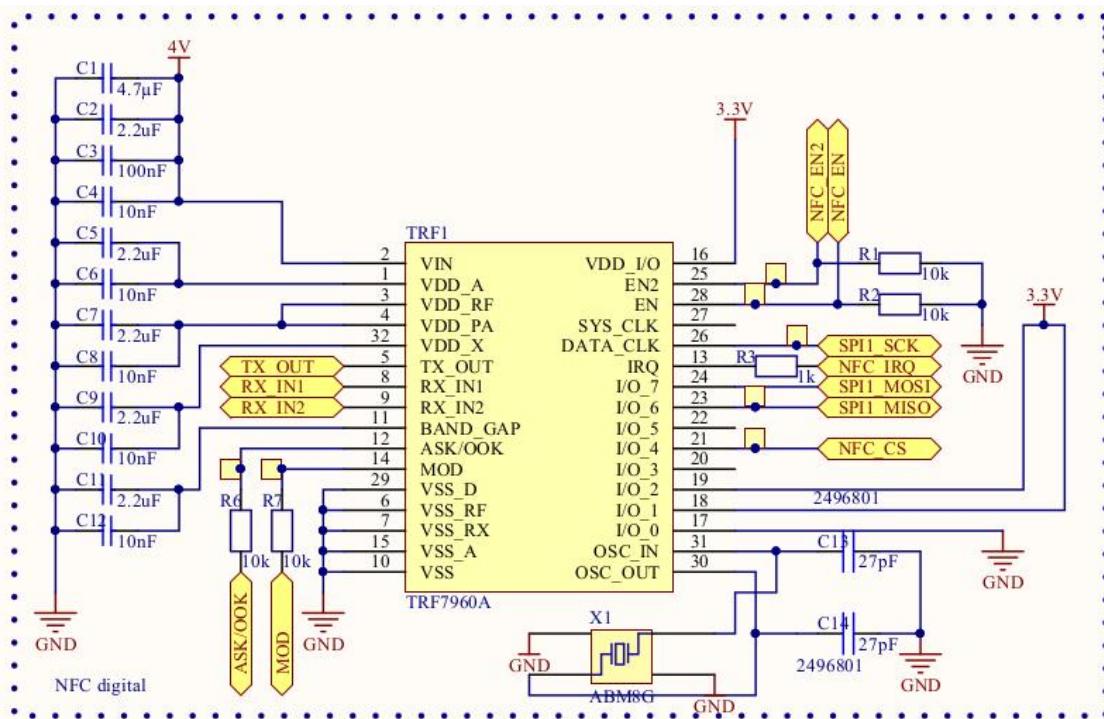
Kolejną ważną częścią urządzenia jest moduł akcelerometru. Pozwala on na wybudzenie urządzenia w momencie wykrycia ruchu pojazdu, a w razie braku deaktywacji funkcji - uruchomienie procedury alarmowej. Ponadto, dzięki jego wskazaniom możliwe jest wyznaczenie przyspieszenia pojazdu pozwalające na profilowanie stylu prowadzenia pojazdu przez kierowcę. Wbudowany żyroskop pozwoli na dokładniejsze profilowanie stylu jazdy kierowcy w trakcie pokonywania zakrętów oraz zmiany pasa. Schemat modułu akcelerometru przedstawiono na rysunku 3.12.



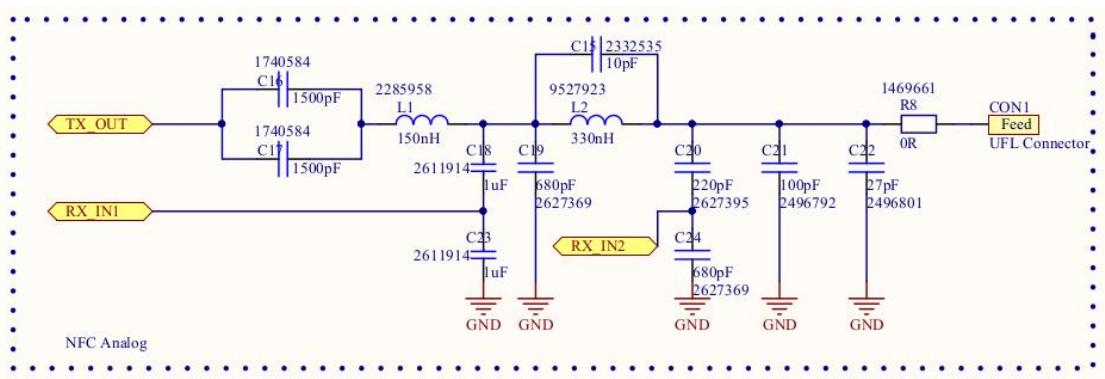
Rysunek 3.12: Schemat modułu akcelerometru w urządzeniu lokalizującym.
 Źródło: Twórczość własna

3.1.6 Moduł NFC

Moduł ten stanowi istotną część z punktu widzenia bezpieczeństwa komunikacji bezprzewodowej. Jest ono zapewnione poprzez zastosowanie szyfrowania wiadomości. Jeśli jednak ktoś podsłucha transmisję inicjalizacji urządzenia, w której przekazywane są klucze szyfrujące, cały koncept traci sens. Dzięki zastosowaniu modułu NFC, możliwość podsłuchania transmisji wymiany kluczy szyfrujących zostaje zniwelowana poprzez fizyczne ograniczenia zasięgu komunikacji. NFC posiada bowiem maksymalny zasięg do 10 cm. Komunikacja odbywa się pomiędzy dwoma urządzeniami. Ze względu na sposób transmisji, jedno z urządzeń inicjuje komunikację. Inicjator generuje zmienne pole magnetyczne, w którym może (lecz nie musi) zawrzeć dane wysypane do urządzenia docelowego. Urządzenie docelowe wykrywa to pole i może odpowiedzieć poprzez odpowiednie zwiększenie go, które jest wykrywane przez inicjator. Urządzenie docelowe nie generuje żadnego pola magnetycznego. Może jedynie zwiększać pole generowane przez inicjator. Stąd wynika, że inicjator musi mieć znacznie większe zużycie energii niż urządzenie docelowe – tag. W urządzeniu lokalizacyjnym zastosowano moduł inicjatora NFC, którego schemat przedstawiono na rysunkach 3.13 - część cyfrowa oraz 3.14 - część analogowa.



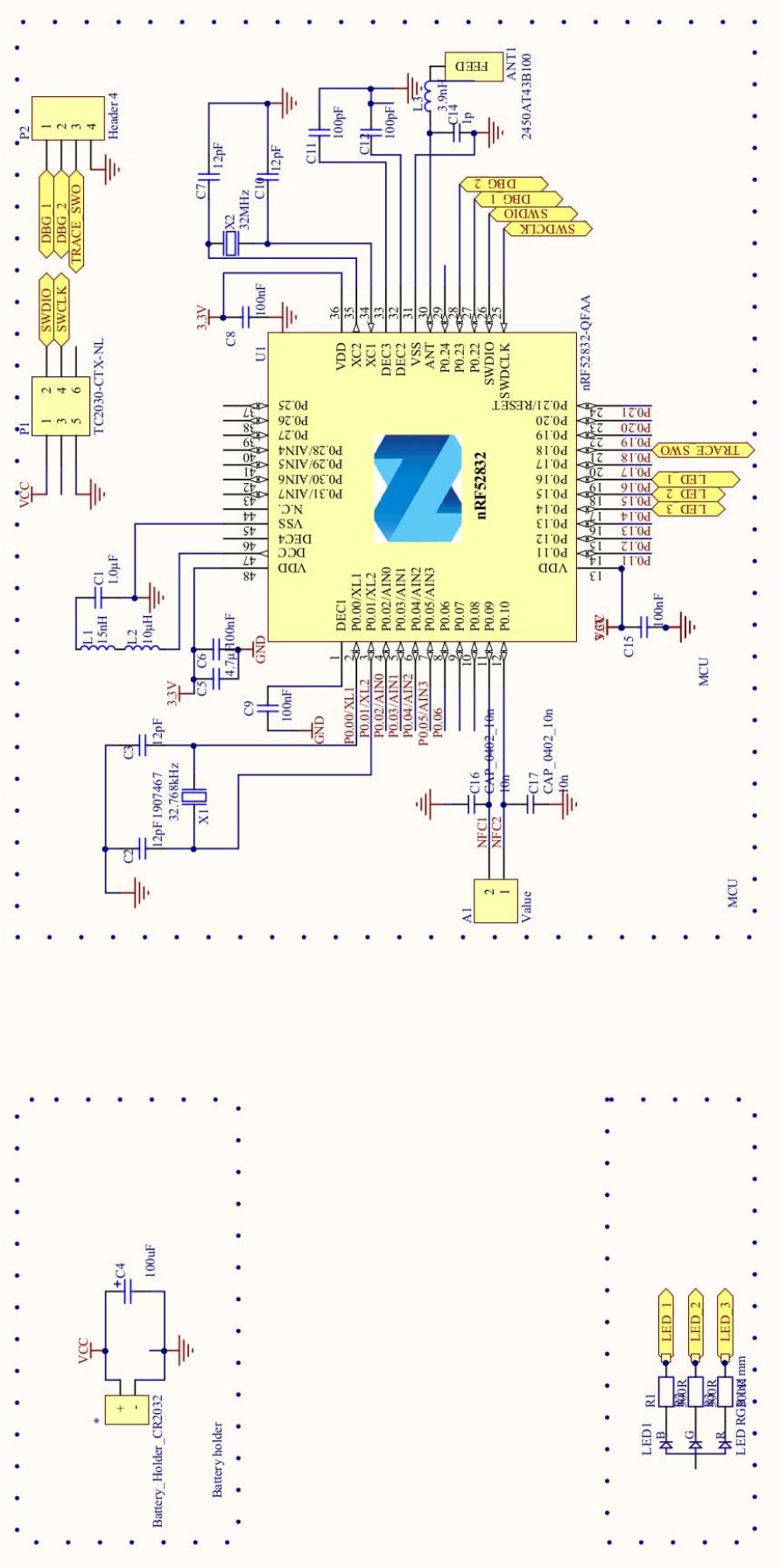
Rysunek 3.13: Schemat części cyfrowej modułu NFC w urządzeniu lokalizującym.
 Źródło: Twórczość własna



Rysunek 3.14: Schemat części analogowej modułu NFC w urządzeniu lokalizującym.
 Źródło: Twórczość własna

3.2 Urządzenie deaktywujące

Główym zadaniem tego urządzenia jest cykliczne rozgłaszenie swej obecności. Po wykryciu przez urządzenie lokalizujące, łączy się ono z deaktywatorem oraz bezpiecznym kanałem dokonywane jest wyłączenie funkcji alarmu. Dzięki temu, że urządzenie to ma tak proste zadanie, nie pobiera ona dużo energii, więc możliwe jest zasilenie go ze standardowej baterii CR2032 o promieniu 20 mm i grubości 3.2 mm. Urządzenie to, przy odpowiedniej konfiguracji parametrów transmisji może działać kilka lat bez konieczności jej wymiany. Zastosowanie wspomnianego źródła zasilania stanowi kompromis pomiędzy czasem działania i rozmiarem urządzenia, które docelowo powinno być umieszczone przy kluczach samochodowych. Schemat deaktywatora przedstawiono na rysunku 3.3.



Rysunek 3.15: Schemat modułu zasilania urządzenia deaktywującego.
 Źródło: Twórczość własna

Rozdział 4

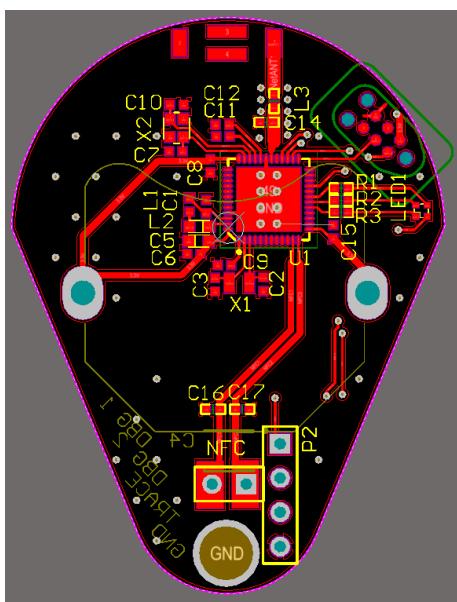
Schematy płytEK drukowanych

4.1 Urządzenie deaktywujące

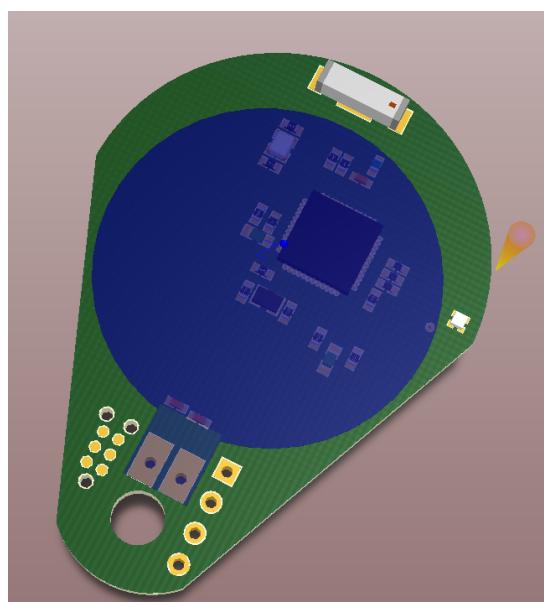
Ze względu na pełniony przez urządzenie cel, powinno ono zawsze towarzyszyć osobie upoważnionej do uruchomienia pojazdu. Biorąc pod uwagę przykład zastosowania urządzenia we flotach pojazdów, szybko można zauważyc, że zazwyczaj do pojazdu nie jest przypisana jedna osoba, lecz może być on używany przez wielu kierowców. Stąd też logiczny staje się wniosek, że urządzenie nie może być przyporządkowane do kierowcy, lecz do pojazdu. Idealnym rozwiązaniem wydaje się umieszczenie go jako dodatkowy brelok przy kluczykach lub karcie umożliwiającej uruchomienie pojazdu bezkluczykowo. Z tego powodu ważne stają się wymiary samego urządzenia. Nie powinno być ono zbyt grube, aby nie przeszkadzało w kieszeni, ani zbyt duże, aby nie obijało się o nogi, a tym samym nie rozpraszało kierującego w trakcie jazdy. Wymiary płytki urządzenia deaktywującego wynoszą odpowiednio 32 mm x 43 mm szerokości i wysokości.

Ze względu na prostotę urządzenia, składa się ono z bardzo niewielu modułów. Na górnej warstwie płytka znajduje się serce układu - mikrokontroler nRF52832 wraz z anteną 2.4 GHz ISM do komunikacji poprzez Bluetooth Low Energy. Dodatkowo, znajdują się tam złącze do programowania, złącze debugowe oraz antena NFC, zwizualizowana jako koło w kolorze niebieskim. Górną warstwę płytki przedstawiono na rysunku 4.1.

Centralne miejsce na dolnej warstwie płytki zajmuje bateria litowa CR2032, która zapewni kilkuletnią pracę dezaktywatora. Posiada ona średnicę 20 mm oraz grubość 3.2 mm, co stanowi idealny kompromis pomiędzy wymiarami urządzenia, a czasem jego pracy. Mniejsze baterie oferują mniejszą pojemność liczoną w miliampero godzinach, a także większy opór wewnętrzny co zwiększa straty energetyczne. Wygląd oraz wizualizację dolnej warstwy płytki przedstawiono na rysunku 4.2.

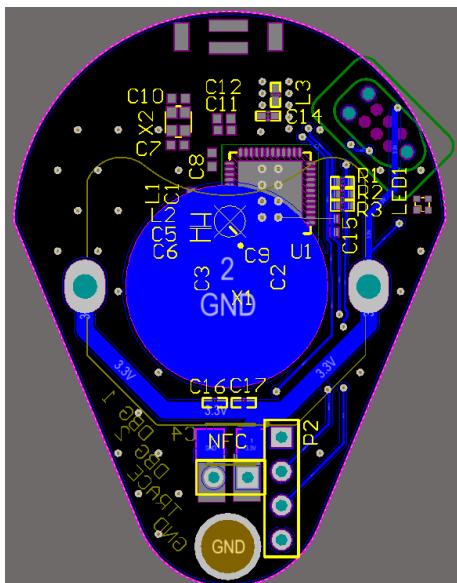


(a) Wygląd górnej warstwy płytka

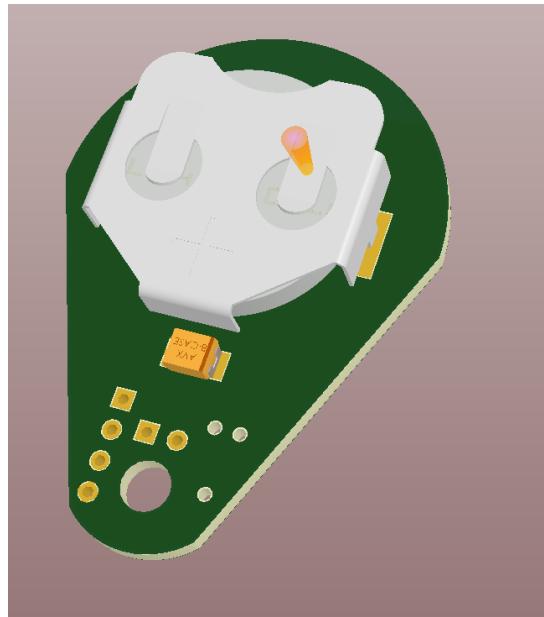


(b) Wizualizacja górnej warstwy płytka

Rysunek 4.1: Wygląd górnej warstwy płytka urządzenia deaktywującego oraz jej wizualizacja.
Źródło: Twórczość własna



(a) Wygląd dolnej warstwy płytka



(b) Wizualizacja dolnej warstwy płytka

Rysunek 4.2: Wygląd dolnej warstwy płytka urządzenia deaktywującego oraz jej wizualizacja.
Źródło: Twórczość własna

4.2 Urządzenie lokalizujące

Płytkę lokalizującą jest znacznie bardziej skomplikowana od urządzenia deaktywującego. Wynika to głównie z faktu, iż stanowi podstawę funkcjonalności całego systemu, a zatem posiada wiele modułów realizujących określone zadania. Płytkę ma wymiary 50 mm x 50 mm, co powinno umożliwić ukrycie jej w większości miejsc w pojeździe (na przykład pod plastikowymi zabudowami kokpitu).

Wygląd i wizualizacje urządzenia przedstawiono na końcu tego rozdziału, na rysunkach 4.4 oraz 4.5.

Ze względu na użycie kilku układów radiowych, wykorzystujących częstotliwości od 13.56 MHz (NFC), poprzez 900 MHz/ 1800 MHz (GSM) i 1575.42 MHz (GPS) aż po 2.4 GHz (Bluetooth), a także przetwornicy impulsowej o znacznym szczytowym natężeniu prądu (aż do 1.5 A), niezbędne jest odpowiednie rozłożenie elementów na płytce, które zminimalizowałoby ich wzajemny wpływ. W związku z tym, postanowiono umieścić kluczowe elementy zasilające oraz radiowe w rogach płytki, aby zmaksymalizować wzajemne odległości. W ten sposób, w lewym górnym rogu płytka umieszczono złącza zasilania wejściowego, w prawym górnym - cewkę indukcyjną, stanowiącą główny element impulsowej stabilizacji napięcia. Cewka ta przy okazji stanowi główne źródło zakłóceń sygnałów. W lewym dolnym rogu znajduje się antena Bluetooth Low Energy, natomiast w prawym dolnym rogu - złącze anteny GPS.

Przewody anten GPS oraz GSM są dodatkowo ekranowane, dzięki czemu znacznie zmniejszona jest podatność tych sygnałów na zakłócenia w trakcie przepływu od anteny do płytki. Jednakże na samej płytce sygnały te nie posiadają ekranu elektromagnetycznego, przez co są podatne na szумy. Z tego względu niezbędna jest minimalizacja długości ścieżek między złączem anteny oraz wejściami układów. W dodatku, sygnał GPS stanowi najsłabszy ze wszystkich sygnałów radiowych, wykorzystywanych w urządzeniu, przez co niezbędnie staje się jak największe oddalenie toru GPS od pozostałych układów. Ze względu na fakt, iż sygnał GSM jest znacznie mocniejszy przez co jest mniej podatny na zakłócenia, jego tor radiowy znajduje się na środku prawego boku płytki, bliżej cewki indukcyjnej przetwornicy impulsowej.

Wszystkie sygnały radiowe użyte w urządzeniu są sygnałami analogowymi. Są one podatne na zjawisko odbicia fali elektromagnetycznej, które polega na odbiciu sygnału na końcu przewodu, bądź ścieżki elektrycznej i nałożeniu się na sygnał pierwotny. Wprowadza to dodatkowe zakłócenia w transmisji sygnału, a spowodowane jest niedopasowaniem impedancji toru transmisyjnego. Aby zminimalizować ten efekt, należy zaprojektować ścieżki po których przesyłany jest sygnał wysokiej częstotliwości tak, aby miały odpowiednią impedancję, zgodną z impedancją anteny. Dokonuje się to poprzez dobór grubości (wynika ona z grubości warstwy miedzi, zazwyczaj $35 \mu\text{m}$) oraz szerokości (wybór pod kątem optymalności zużycia miejsca na PCB)

ścieżek radiowych. Na podstawie tak wyznaczonych parametrów wylicza się ich niezbędną długość, aby osiągnąć założoną impedancję. W przypadku sygnałów GPS, GSM oraz Bluetooth wynosi ona 50Ω . Ostateczną impedancję, uwzględniającą pojemności i indukcyjności pasożytnicze między ścieżkami, zmierzona po złożeniu płytka można jeszcze skorygować poprzez dobór elementów w filtrach przyantenowych (filtry: C45, R21, C47 oraz C44, R22, C46). Dodatkowo, ścieżki wysokiej częstotliwości prowadzi się łagodnymi łukami, bez ostrych załamań, które mogłyby zwiększać pojemność, a tym samym mogących zmienić impedancję.

W dodatku, w trakcie projektowania urządzenia należy pamiętać o wysokim chwilowym poborze prądu (aż do około 2 A). Z tego względu, trzeba zaprojektować odpowiednio grube ścieżki zasilające. Jest to wymagane z dwóch powodów. Pierwszym z nich jest fakt oporności ścieżki.

$$R = \rho \cdot l / S \quad (4.1)$$

gdzie:

R - oporność ścieżki,

ρ - oporność właściwa materiału, z którego wykonano ścieżkę,

l - długość ścieżki,

S - powierzchnia (liczona jako iloczyn grubości i szerokości) ścieżki,

Jak widać, im większa szerokość ścieżki, tym większa jej powierzchnia, a więc mniejsza rezystancja. Im mniejsza rezystancja, tym straty napięcia na samej ścieżce będą mniejsze, a tym samym większe napięcie dostarczane do układów funkcjonalnych i mniejsze straty na ciepło, generujące niepotrzebne zużycie energii.

$$U = R \cdot I \quad (4.2)$$

gdzie:

U - strata napięcie na ścieżce,

R - opór ścieżki,

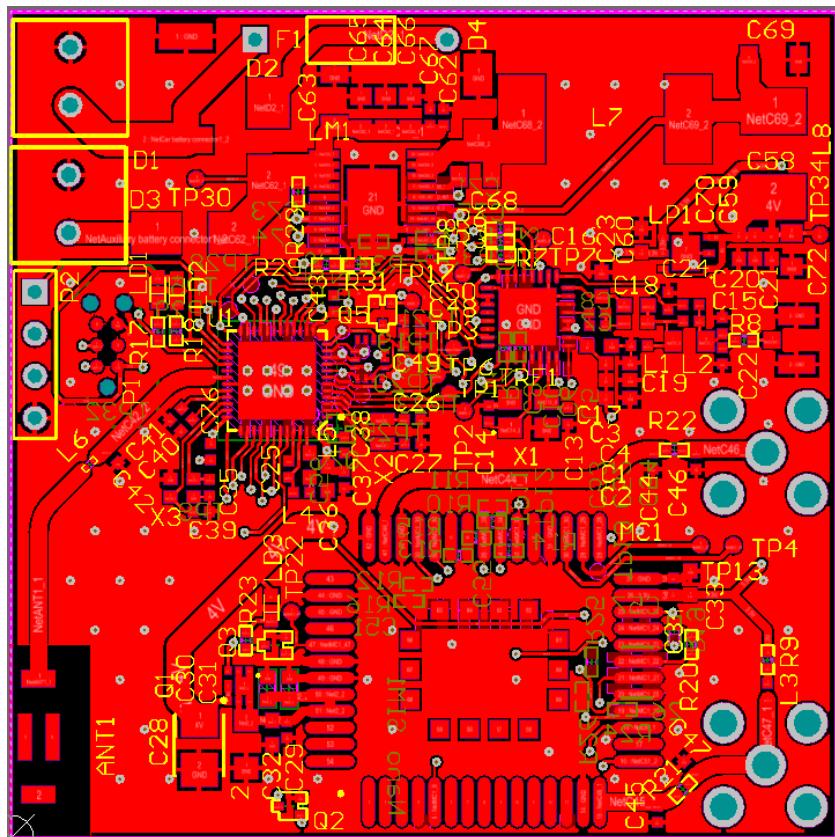
I - prąd płynący przez ścieżkę

Drugi przypadek wynika niejako z pierwszego. Gdy opór ścieżki jest zbyt duży, w momencie zwiększonego poboru prądu, energia tracona na ścieżce może być tak duża, że ulega ona przepaleniu i całe urządzenie przestaje działać. Aby się przed tym ustrzec, ścieżki zasilające mają grubość 2 mm, co pozwala na przepływ około 3,5 A natężenia ciągłego prądu w temperaturze 20 stopni Celsjusza. Ze względu jednak na fakt, iż główne obciążenie urządzenia stanowi prąd chwilowy, trwający bardzo krótko, średnie natężenie prądu będzie dużo niższe od tej wartości. Szerokość ścieżek została dobrana z zapasem, aby nie uległy one przepaleniu przed zadziałaniem bezpiecznika zwłocznego i zmniejszyć straty energii. Tabela zestawiająca zależność między grubością ścieżek na płytce PCB od wartości maksymalnego dopuszczalnego prądu ciągłego, przepływającego przez nią, przedstawiona na rysunku 4.3.

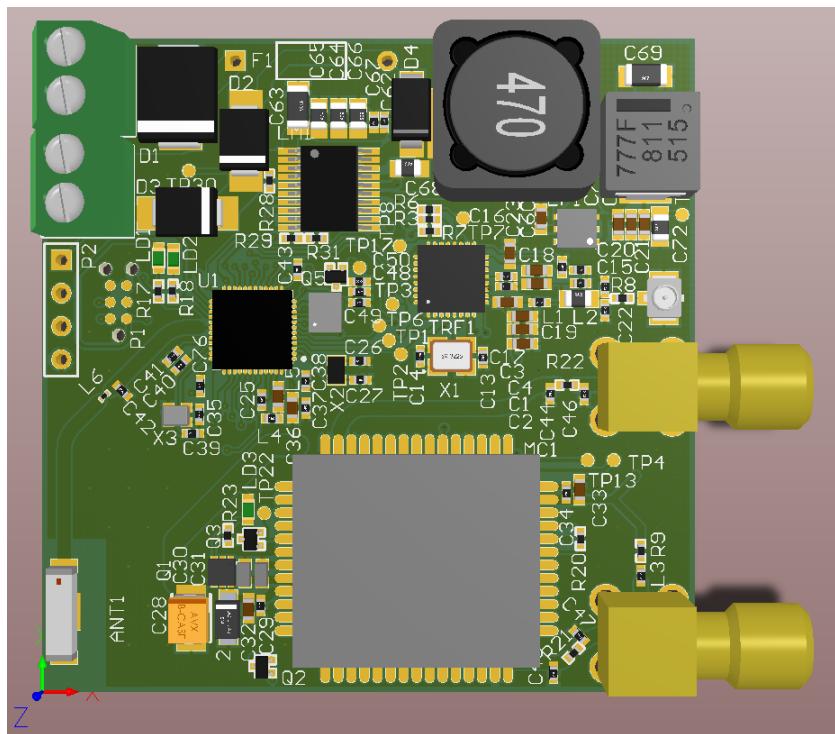
Szerokość ścieżki	Dopuszczalny prąd		
	$\Delta T=20^\circ\text{C}$	$\Delta T=80^\circ\text{C}$	prąd niszczący
0,5mm (20mil)	1,5A	3,5A	6A
1mm (40mil)	2,5A	5A	8A
2mm (80mil)	3,5A	7A	12A
3mm (120mil)	5A	10A	18A

Uwaga! dotyczy typowej płytki drukowanej o grubości miedzi 0,035...0,038mm

Rysunek 4.3: Tabela opisująca korelację między grubością ścieżek, a maksymalnym dopuszczalnym natężeniem prądu. Źródło: [18]

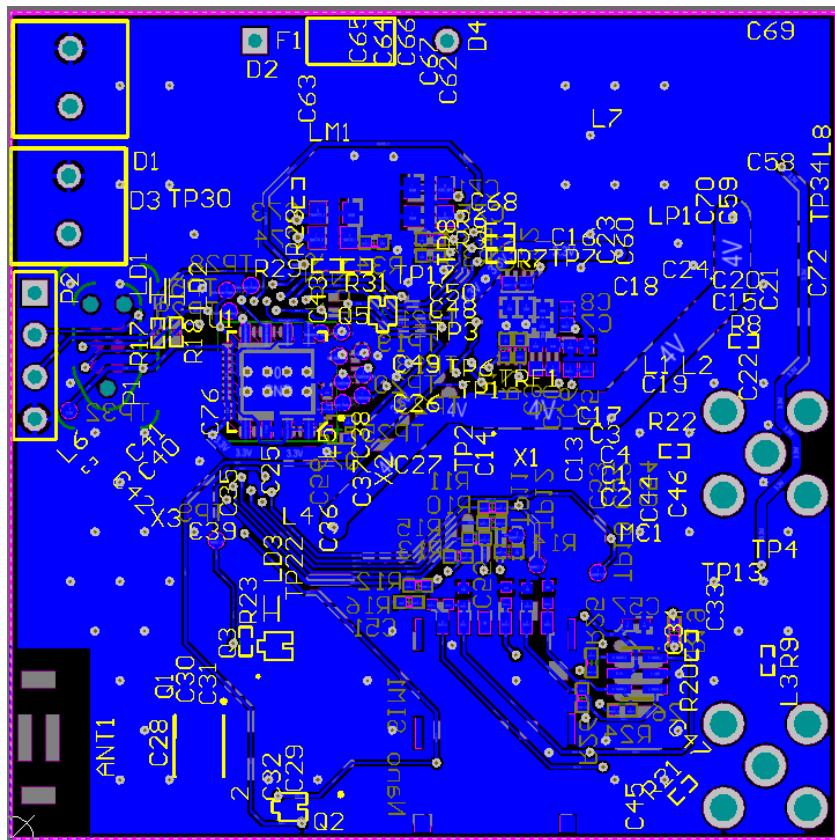


(a) Wygląd górnej warstwy płytki

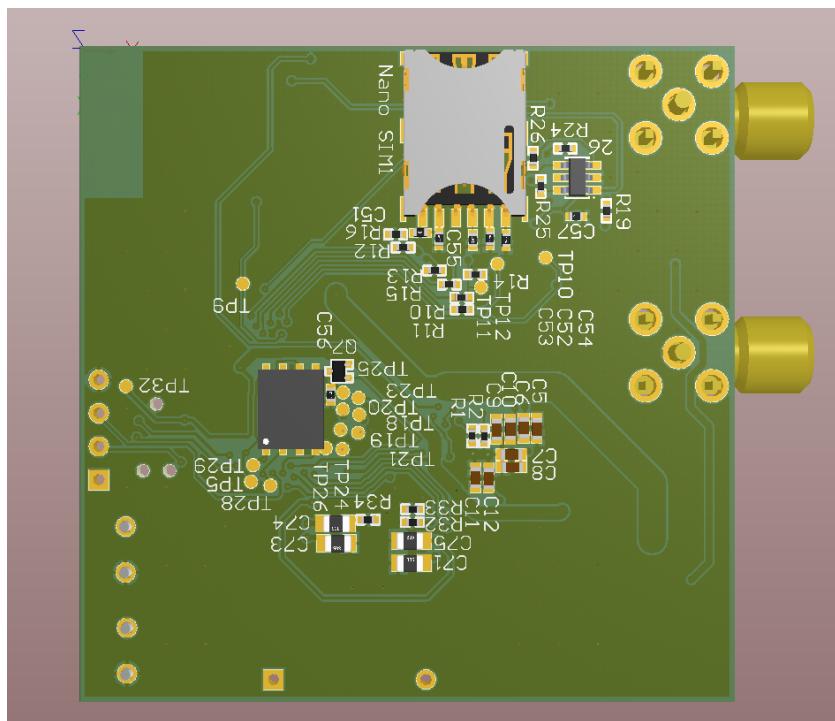


(b) Wizualizacja górnej warstwy płytki

Rysunek 4.4: Wygląd górnej warstwy płytki urządzenia lokalizującego oraz jej wizualizacja.
 Źródło: Twórczość własna



(a) Wygląd dolnej warstwy płytki



(b) Wizualizacja dolnej warstwy płytki

Rysunek 4.5: Wygląd dolnej warstwy płytki urządzenia lokalizującego oraz jej wizualizacja.
 Źródło: Twórczość własna

Rozdział 5

Bezpieczeństwo komunikacji

Jednym z podstawowych wymagań dotyczących tej pracy jest bezpieczna wymiana komunikatów poprzez Bluetooth Low Energy. Jest to tak kluczowe, ponieważ za pomocą tego protokołu, poprzez bezprzewodowe medium, przesyłane są kluczowe dane, zwłaszcza komendy deaktywujące tryb alarmowy urządzenia. Transmisja fizycznie jest zawsze realizowana rozgłoszeniowo, co powoduje, że jej podsłuchanie nie jest trudnym zadaniem. Jest to niebezpieczne z dwóch powodów. Pierwszym z nich jest fakt wysyłania wrażliwych danych, jak na przykład dane lokalizujące pojazd. Dzięki nim, potencjalny złodziej mógłby po krótkiej analizie bezproblemowo określić miejsca, w których regularnie przebywa pojazd, a następnie wybrać dla niego najbardziej korzystne i przygotować się do kradzieży. Po drugie, będąc w pobliżu pojazdu w trakcie wyłączania trybu alarmowego, byłby w stanie podsłuchać komendę deaktywującą, a następnie zapisać ją w celu późniejszego odtworzenia, co umożliwiłoby bezproblemową kradzież pojazdu.

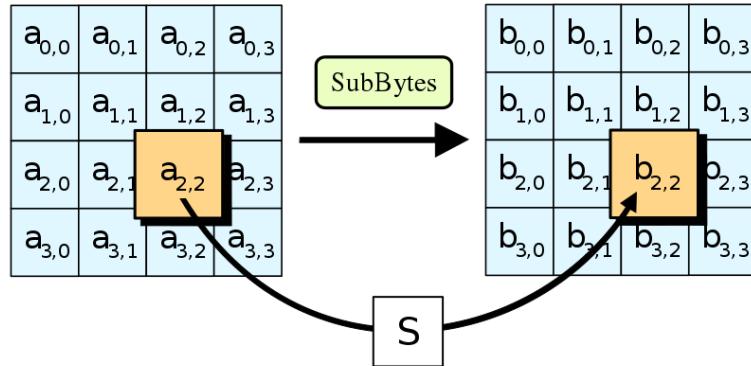
Z przytoczonych powyżej powodów komunikacja bezprzewodowa musi być szyfrowana. Jednakże operacja ta sama w sobie nie zabezpiecza tak naprawdę komendy deaktywującej, a jedynie wrażliwe dane. Wynika to z faktu, iż w przypadku przechwycenia danych przesyłanych bezprzewodowo, dzięki szyfrowaniu są one nadal bezpieczne, ponieważ są one kompletnie niezrozumiałe. Inaczej jest w przypadku stałej komendy autoryzującej. Wynika to z faktu, że nie musi być ona tak naprawdę zrozumiała dla potencjalnego złodzieja. Wystarczy, że jedynie ją odtworzy, nawet w formie zaszyfrowanej. Urządzenie wówczas ją zdeszyfruje i wykona deaktywację alarmu. W wyniku szyfrowania stałej komendy stałym kluczem szyfrującym, uzyskamy oczywiście również stały i powtarzalny pakiet zaszyfrowanych danych, które mogą być bezcenne w ręku potencjalnego złodzieja. W celu zabezpieczenia się przed tym, do komunikacji należy wprowadzić element zmienności w czasie.

5.1 AES

Jako główny algorytm szyfrowania w niniejszej pracy wykorzystano algorytm AES (ang. Advanced Encryption Standard) w wersji ze 128-bitowym kluczem szyfrującym. Wyboru tego dokonano, ponieważ zastosowany w pracy mikrokontroler nRF52832 firmy Nordic Semiconductor posiada sprzętowe wsparcie szyfrowania danych wykorzystując właśnie AES128. Algorytm ten powstał w 2001 roku w Stanach Zjednoczonych w ośrodku NIST (ang. National Institute of Standards and Technology) w wyniku prac badawczych dwóch belgijskich kryptografów - Vincenta Rijmena i Joan'a Daemen, od których nazwisk powstała oryginalna nazwa algorytmu – Rijndael. Stanowi on jeden z najpopularniejszych na świecie szyfrów symetrycznych, a o jego skuteczności stanowi fakt, że w 2002r. został przyjęty jako federalny standard szyfrowania w Stanach Zjednoczonych. Pojęcie szyfr symetryczny oznacza, że do zaszyfrowania oraz zdeszyfrowania stosuje się ten sam klucz szyfrujący (w przeciwieństwie do algorytmów asymetrycznych, gdzie stosuje się dwa klucze, jeden do szyfrowania, a drugi do deszyfrowania). Z tego powodu, klucz szyfrujący stanowi ekstremalnie wrażliwą daną, której pod żadnym pozorem nie powinno się przesyłać poprzez ogólnie dostępne medium komunikacyjne. Wyciek klucza szyfrującego powoduje narażenie całej komunikacji w wyniku czego przestaje ona być uznawana za bezpieczną. Proces szyfrowania składa się z kilku kroków. Pierwszym z nich jest podzielenie danych wejściowych (zwyczajowo nazywanych tekstem jawnym) na bloki o rozmiarze 128 bitów, czyli szesnastu bajtów. Każdy blok przedstawiany jest jako macierz o wymiarach 4 bajty x 4 bajty, szeregowana kolumnami. Macierze te nazywają się macierzami stanu. Następnie, na każdej z tych macierzy (bloku danych) wykonywane są kolejne operacje:

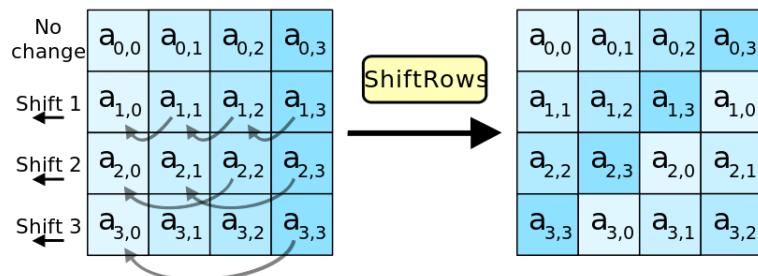
1. Utworzenie podkluczy – Etap ten polega na wygenerowaniu w sposób losowy klucza pierwotnego, a następnie na jego podstawie - po jednym podkluczu dla każdej z rund szyfrujących. Ich liczba jest uzależniona od rozmiaru klucza. Dla klucza 128-bitowego występuje 10 rund, dla klucza 192-bitowego – 12, a dla klucza 256-bitowego – 14 powtórzeń, wliczając klucz pierwotny.
2. Wykonanie rundy wstępnej (inicjującej) – Polega na wykonaniu operacji alternatywy wyłączającej – XOR (ang. Exclusive Or) dla każdego bajtu z bloku danych oraz odpowiadającego mu bajtu w kluczu pierwotnym.
3. Wykonanie rund szyfrujących – Etap ten jest wykonywany kilkukrotnie, w zależności od liczby cykli. Każda runda składa się z kilku kroków.
 - W pierwszym z nich, każdy bajt danych jest zastępowany innym bajtem pobranym ze zdefiniowanej tablicy (ang. *lookup table*) nazywanej S-Boxe'm Rijndael'a. Operacja

ta nazywa się w skrócie SB (*ang. Substitute Bytes*) i przedstawiono ją na rysunku 5.1. Zgodnie z zamysłem twórców, tablica ta gwarantuje nieliniowość przekształcenia, a w efekcie i całego szyfrowania.



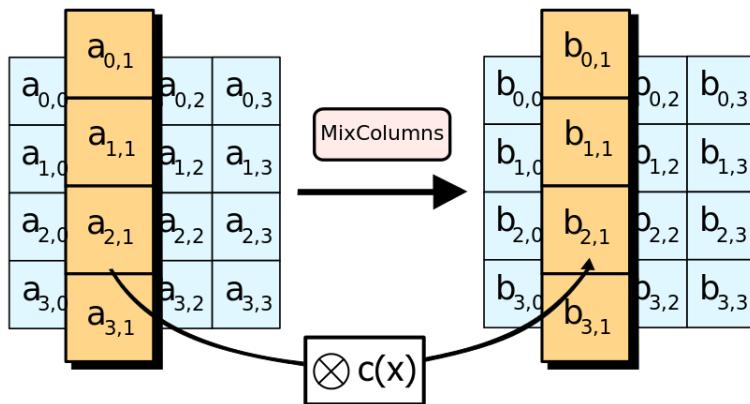
Rysunek 5.1: Wykonanie operacji Substitute Bytes. Źródło: [20]

- Kolejny krok to zamiana wierszy. Polega na przesunięciu bajtów w trzech ostatnich wierszach bloku. Pierwszy wiersz pozostaje bez zmian, w drugim wierszu bajty są przesuwane o jeden w lewo, w trzecim o dwie pozycje w lewo, a w ostatnim o 3 miejsca w tym samym kierunku. Każdy bajt, który w wyniku przesunięcia znajdzie się poza wierszem, zostaje umieszczony na jego ostatniej pozycji (wiersze w wyniku rotacji się zawijają). Operacja ta nosi miano SR (*ang. Shift Rows*). Przedstawiono ją na rysunku 5.2.



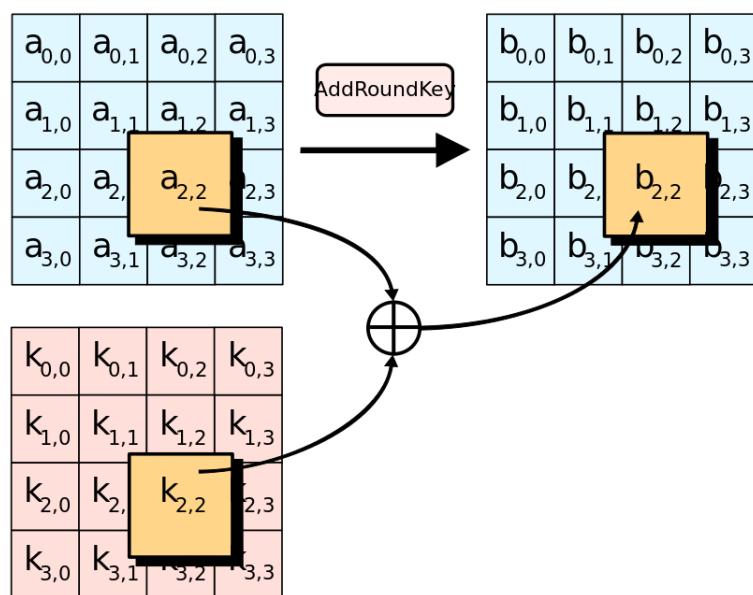
Rysunek 5.2: Wykonanie operacji Shift Rows. Źródło: [20]

- Trzecim z kolei krokiem jest operacja mieszania kolumn – MC (*ang. Mix Columns*). W tym etapie, każda z kolumn jest przemnażana lewostronnie przez stałą macierz o wymiarach 4 x 4, w wyniku czego powstaje kolumna z nowymi wartościami. Operacja ta przedstawiona jest na rysunku 5.3.



Rysunek 5.3: Wykonanie operacji Mix Columns. Źródło: [20]

- Ostatni krok nazywany jest AR (ang. **Add Round Key**) i polega na wykonaniu operacji XOR na każdym bajcie bloku danych i odpowiadającym mu bajcie w kluczu przypisany do danej rundy. Wizualizację kroku przedstawiono na rysunku 5.4.



Rysunek 5.4: Wykonanie operacji Add Round Key. Źródło: [20]

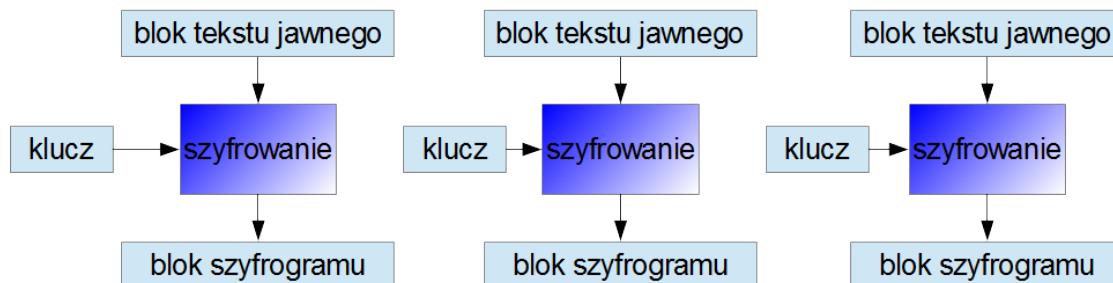
- Ostatni etap to runda kończąca – W jej trakcie wykonywane są operacje identyczne jak w rundach szyfrujących, za wyjątkiem mnożenia kolumn, które nie występuje.

Deszyfrowanie jest operacją odwrotną do szyfrowania i polega na przekształceniu danych zaszyfrowanych na tekst jawny. Tak samo jak w przypadku szyfrowania, tekst dzieli się na 16-bajtowe bloki. W jego trakcie wykonuje się analogiczne operacje co w przypadku szyfrowania.

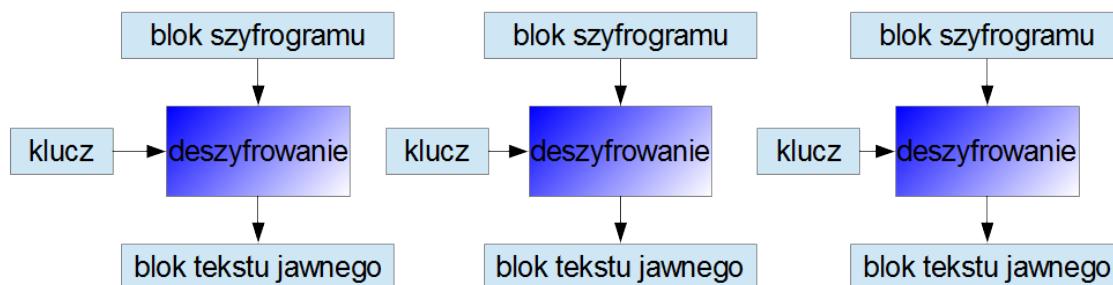
- Odwrotne podstawianie bajtów – polega na ponownym zastosowaniu tablicy S-Box w celu podmiany bajtów.

2. Przesuwanie bajtów w wierszach w prawo. Zasada jest taka sama jak w operacji SR, zmienia się jedynie kierunek.
3. Wykonanie operacji XOR dla każdego bajtu bloku danych z odpowiadającym mu bajtem w podkluczu przypisany do danej rundy deszyfrującej. Podklucze są takie same jak w trakcie szyfrowania, lecz powinny być brane w kolejności odwrotnej (zaczynając od ostatniego, a kończąc na kluczu pierwotnym).
4. Ostatnia operacja to odwrócone mnożenie kolumn.

W efekcie uzyskujemy blok danych zdeszyfrowanych. Przedstawiony tutaj wariant algorytmu szyfrowania nosi miano ECB (ang. ***Electronic Codebook***) i stanowi najprostszą metodę szyfrowania. Można go przedstawić na rysunkach 5.5 oraz 5.6.



Rysunek 5.5: Operacja szyfrowania metodą ECB. Źródło: [19]

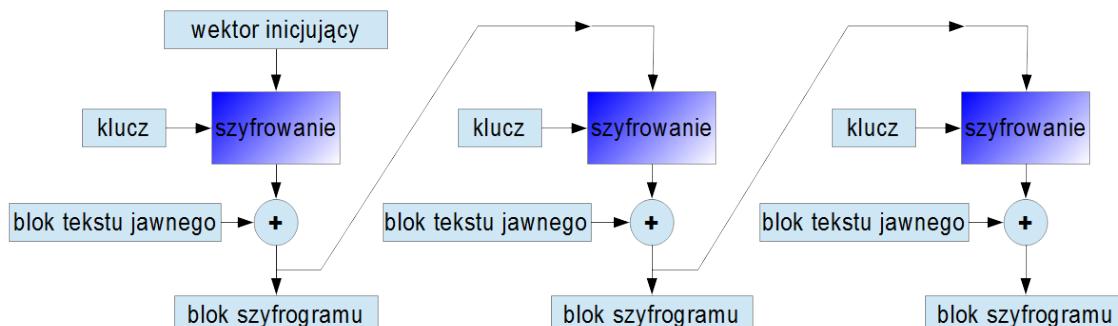


Rysunek 5.6: Operacja deszyfrowania metodą ECB. Źródło: [19]

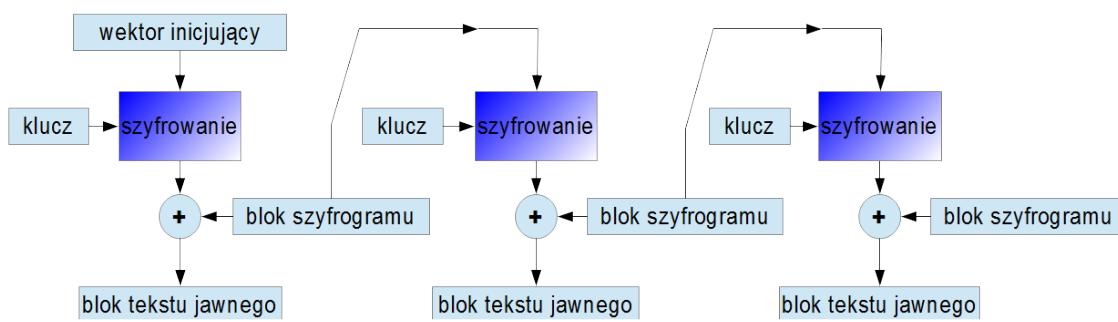
Czas trwania szyfrowania pojedynczego bloku danych o długości szesnastu bajtów na mikrokontrolerze nRF52832 wynosi w przybliżeniu $30 \mu\text{s}$. Deszyfrowanie trwa zaś około $60 \mu\text{s}$.

5.2 Dodatkowe warianty szyfrowania AES

Jak zostało przedstawione wcześniej, mechanizmy szyfrowania doskonale działają w przypadku wrażliwych danych. Nie sprawdzają się natomiast w przypadku przesyłania komend, ze względu na brak zmienności pakietów w czasie i możliwości odtworzenia zaszyfrowanego pakietu przez niepowołane osoby. Z tego powodu, do komunikacji należy wprowadzić element zmienności. Jednym z wariantów algorytmu AES jest tzw. CFB (ang. *Cipher Feedback*), przedstawiony na rysunkach 5.7 oraz 5.8. Stanowi on wysokopoziomowy algorytm, który bazuje na wariantie ECB, zmieniając jedynie logiczną strukturę informacji niezbędnych do szyfrowania. Przede wszystkim, wprowadza pojęcie wektora inicjującego (ang. *initializing vector*), który stanowi niezbędny dodatkowy element zmienności. Klucz główny jest zazwyczaj niezmienny dla pary komunikujących się ze sobą urządzeń, co w rezultacie wprowadza konieczność niepubliczniania go. Wektor inicjujący jest natomiast generowany przy każdej nowej komunikacji.



Rysunek 5.7: Operacja szyfrowania metodą CFB. Źródło: [19]



Rysunek 5.8: Operacja deszyfrowania metodą CFB. Źródło: [19]

W odróżnieniu od wariantu ECB, zamiast tekstu jawnego szyfrowaniu ulega wektor inicjujący. Jego postać zaszyfrowana jest następnie poddawana operacji XOR z blokiem danych tekstu jawnego, a powstały w ten sposób szyfrogram stanowi nowy wektor inicjujący dla następnego bloku danych. W przypadku deszyfrowania korzysta się oczywiście z tego samego wektora inicjującego oraz klucza szyfrującego. Co ciekawe, w odróżnieniu od wariantu ECB, w metodzie

CFB deszyfrowanie jest to tak naprawdę szyfrowanie. Oznacza to, że wystarczy zaimplementować jedynie mechanizm szyfrowania w algorytmie AES, aby móc zarówno szyfrować jak i deszyfrować wiadomości. Zaszyfrowany wektor inicjujący jest poddawany operacji XOR z blokiem tekstu zaszyfrowanego w efekcie czego uzyskujemy blok tekstu jawnego. Natomiast blok tekstu zaszyfrowanego stanowi wektor inicjujący dla następnych bloków szyfru.

5.3 Realizacja szyfrowania komunikacji w projekcie

W pracy zdecydowano się na wykorzystanie zarówno metod ECB oraz CFB. Pierwszym, a zarazem najbardziej podstawowym etapem jest generowanie klucza szyfrującego. Operacja ta jest realizowana przez płytę główną systemu lokalizującego. Następnie, klucz jest przekazywany w trakcie inicjalizacji przez interfejs NFC (*ang. Near Field Communication*) do urządzenia deaktywującego, pełniącego rolę beacona (urządzenia rozgłaszającego). Zastosowanie NFC jest powszechnie uważane za bezpieczną metodę komunikacji, ze względu na jej bardzo niską moc transmisji, a tym samym bardzo niewielki zasięg (do 10 cm). Ogranicza to zatem możliwość podsłuchania klucza szyfrującego do zera. Przy pomocy tego klucza, za każdym razem gdy płyta główna systemu połączy się z urządzeniem deaktywującym w celu uzyskania od niego komendy deaktywującej, wpierw wysłany zostanie zaszyfrowany, nowo wygenerowany na potrzeby danego połączenia wektor inicjalizacyjny. Umożliwi to dalszą komunikację wykorzystując wariant CFB oraz niezbędną zmienność zaszyfrowanych pakietów, praktycznie niwelującą skuteczność podsłuchiwanego transmisji.

Rozdział 6

Oprogramowanie

6.1 Urządzenie lokalizujące

Urządzenie lokalizujące stanowi główny moduł, na którym opiera się cały system opisany w niniejszej pracy. Wynika to z faktu, iż jest on odpowiedzialny za praktycznie wszystkie funkcje narzucone w tej pracy. Są to:

- Wygenerowanie głównego klucza szyfrującego i sparowanie z urządzeniem deaktywującym
- Zapewnienie bezpiecznego kanału komunikacji w trakcie połączenia z *Key Tag'iem*
- Wykrywanie ruchu pojazdu
- Komunikacja z urządzeniem deaktywującym, w celu podjęcia próby deaktywacji alarmu
- Alarmowe powiadamianie właściciela w przypadku nieautoryzowanego przemieszczenia pojazdu
- Pobieranie próbek lokalizacji, prędkości, przyspieszenia, a także aktualnego kursu (azygumetu) i innych parametrów po wykryciu ruchu oraz ich cykliczne wysyłanie na zdalny serwer danych
- Analiza stylu jazdy kierowcy
- Wysyłanie poprzez SMS lokalizacji pojazdu na żądanie użytkownika.

Duża liczba zadań do wykonania przez mikrokontroler sterujący działaniem płytki oraz konieczność wywoływania ich po ścisłe określonym upływie czasu spowodowała, że niezbędnym stało się wprowadzenie modułu planisty. Stanowi on bardzo prostą funkcjonalność, bez możliwości wywłaszczenia zadań i zmiany kontekstu, więc nie wprowadza wielowątkowości znanej z pełnoprawnych systemów operacyjnych. Jego celem jest zakolejkowanie zadań, oznaczenie ich jako gotowych do wykonania po upływie wymaganego czasu, a następnie wykonaniu ich przy pierwszej sposobności po wyjściu programu z przerwania. Kod służący do kolejkowania zadań przedstawiono na listingu 6.1. Na listingu 6.2 umieszczono funkcję, która jest cyklicznie wywływana (co 10 ms) w przerwaniu od energooszczędnego timera. Jej zadaniem jest oznaczenie zadań, które można już wykonać. Na ostatnim listingu, oznaczonym numerem 6.3, przedstawiono procedurę wykonującą oczekujące zadania. Ograniczeniem funkcjonalnym jest tutaj fakt, iż żadne z zadań nie może przyjmować argumentów, ani zwracać wartości.

Listing 6.1: Funkcja do kolejkowania zadań

```
scheduler_error_code_e SchedulerAddOperation(void (*callback)(void),
                                              volatile uint32_t timeMsFromNow,
                                              volatile uint8_t* taskIndex,
                                              bool isCyclic){
    scheduler_entry_t entry;

    // Just safe guard not to miss the time
    if (timeMsFromNow < 2)
    {
        timeMsFromNow = 2;
    }

    for (uint8_t i=0; i< SCHEDULER_BUFFER_SIZE; ++i)
    {
        if (_scheduleBuffer[i].isInProgress == false)
        {
            entry.isInProgress = true;
            entry.isTimedOut = false;
            entry.callback = callback;
            entry.timePeriodMs = timeMsFromNow;
            entry.triggerTime = scheduler_current_time_ms + timeMsFromNow;
            entry.isCyclic = isCyclic;
            memcpy(&_scheduleBuffer[i], &entry, sizeof(scheduler_entry_t));
            if (taskIndex != NULL)
                *taskIndex = i;
            return E_SCHEDULER_OK;
        }
    }

    return E_SCHEDULER_NO_RESOURCES;
}
```

Listing 6.2: Funkcja do sprawdzania czy nie należy wykonać zadania

```
scheduler_error_code_e SchedulerCheckOperations(){
    scheduler_current_time_ms += 10;
    for (uint8_t i=0; i< SCHEDULER_BUFFER_SIZE; ++i)
    {
        if (_scheduleBuffer[i].isInProgress == true &&
            _scheduleBuffer[i].triggerTime <= scheduler_current_time_ms)
        {
            // If it is cyclic task - reschedule the next cycle
            if (_scheduleBuffer[i].isCyclic)
            {
                _scheduleBuffer[i].triggerTime = scheduler_current_time_ms
                    +_scheduleBuffer[i].timePeriodMs;
            }

            _scheduleBuffer[i].isTimedOut = true;
        }
    }

    return E_SCHEDULER_OK;
}
```

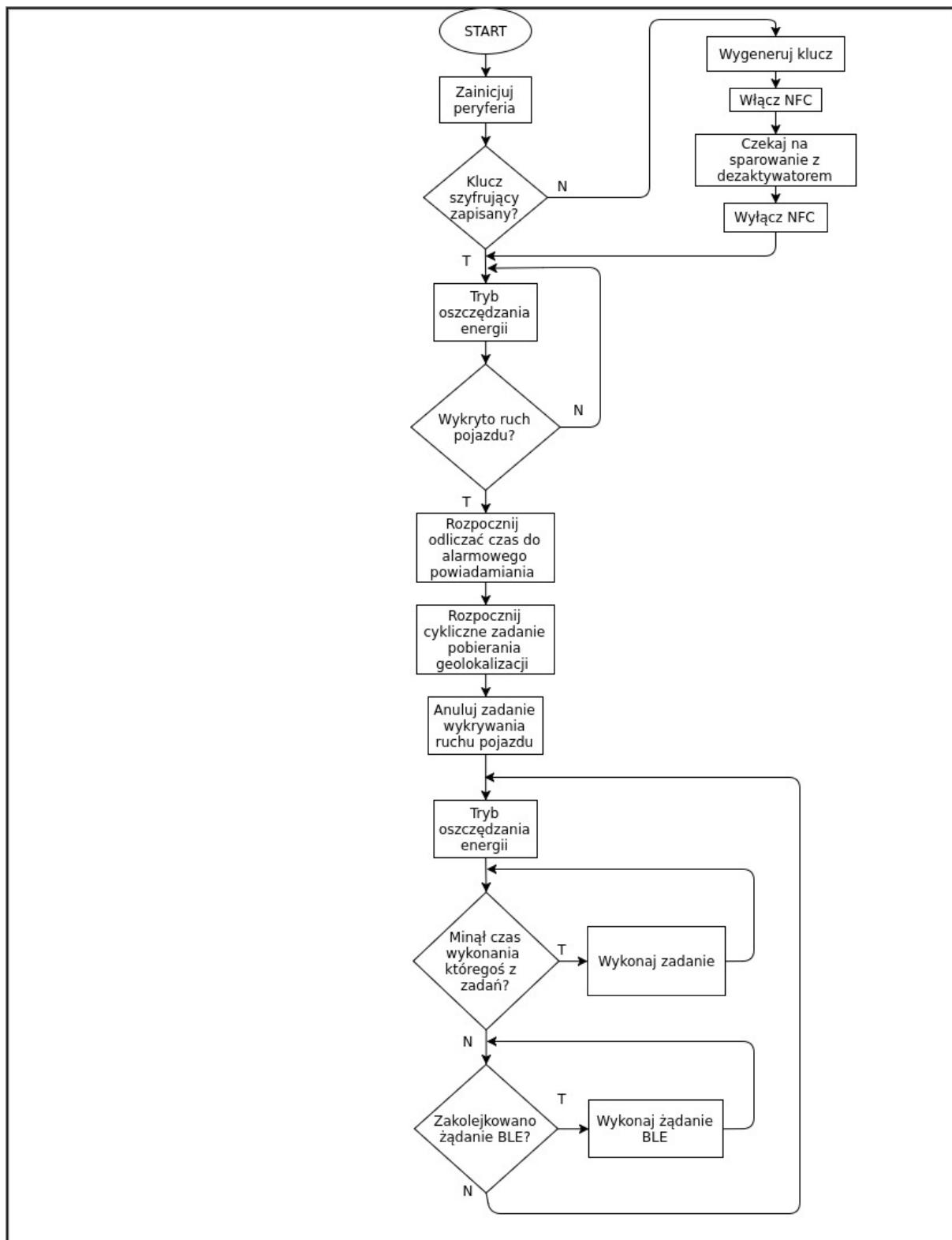
Listing 6.3: Funkcja do wykonywania zadań

```
scheduler_error_code_e ScheduleExecutePendingOperations(){
    for (uint8_t i=0; i< SCHEDULER_BUFFER_SIZE; ++i)
    {
        if (_scheduleBuffer[i].isInProgress == true &&
            _scheduleBuffer[i].isTimedOut == true)
        {
            if (_scheduleBuffer[i].isCyclic == false)
            {
                _scheduleBuffer[i].isInProgress = false;
            }

            _scheduleBuffer[i].callback();
            _scheduleBuffer[i].isTimedOut = false;
        }
    }

    return E_SCHEDULER_OK;
}
```

Główny cykl działania urządzenia został przedstawiony na rysunku 6.1.



Rysunek 6.1: Główny algorytm działania urządzenia.
 Źródło: Twórczość własna

Jak widać na rysunku 6.1, po uruchomieniu i zainicjalizowaniu peryferiów i modułów na płytce, mikrokontroler dokonuje sprawdzenia czy wygenerowany został klucz szyfrujący, służący do zabezpieczenia komunikacji z dedykowanym urządzeniem deaktywującym. Jeśli klucza nie ma, oznacza to, że urządzenie nie zostało jeszcze sparowane. Wówczas, włączony zostaje układ NFC, procesor wchodzi w tryb oszczędzania energii w trakcie oczekiwania na parowanie, a całe urządzenie staje się nieoperacyjne, dopóki nie zostanie powiązane z modułem deaktywującym. Algorytm komunikacji został przedstawiony w podrozdziale 6.2 na rysunku 6.3. Gdy zostanie to zrobione, moduł NFC zostaje wyłączony i nie jest używany aż do momentu powrotu do ustawień fabrycznych, a mikrokontroler wchodzi w tryb oszczędzania energii w oczekiwaniu na nadchodzące zadania.

Pierwszym z nich jest wykrywanie ruchu pojazdu. Zostało to zrealizowane poprzez wykorzystanie funkcji akcelerometru - wybudzenia w razie wykrycia przyspieszenia powyżej programowanego progu, które utrzymywałoby się przez pewien konfigurowalny czas. W wyniku badań eksperymentalnych, został on ustalony na wartość $0.9 \frac{m}{s^2}$, trwającą przez więcej niż 1 sekundę, co pozwala na wykrycie drgań spowodowanych zamknięciem drzwi pojazdu. Akcelerometr jest cykliczne (co 5 sekund) odpytywany przez mikrokontroler, w celu sprawdzenia czy nie nastąpił ruch pojazdu. Jeśli nie zostało to wykryte, mikrokontroler wchodzi w tryb oszczędzania energii i cały cykl się powtarza. Jeśli wykryto ruch, zadanie okresowego sprawdzania przemieszczenia jest wyłączone, natomiast do kolejki zadań ładowane są 3 najważniejsze z punktu widzenia całego systemu procedury - zadanie alarmu, skanowania w poszukiwaniu *Key Tag'a* oraz cyklicznego pobierania próbek lokalizacji. Pierwsze z nich stanowi zegar, który odlicza 30 sekund. Jeśli w tym czasie, alarm zostanie deaktywowany poprzez nawiązanie połączenia z urządzeniem dezaktywującym i nadanie odpowiedniego komunikatu, wówczas zadanie zostaje anulowane i jedynym cyklicznym zadaniem jest próbkowanie lokalizacji. W przeciwnym razie, uruchamiane jest zadanie cykliczne, co 10 minutowego powiadamiania właściciela o lokalizacji pojazdu poprzez wiadomości SMS. Okres ten został dobrany w ten sposób, aby nie wyczerpać za szybko środków na koncie karty SIM, użytej w module, a przy tym uzyskać rozsądną częstotliwość wysyłania SMS'ów. Zadanie to można anulować, w przypadku fałszywego alarmu, wysyłając odpowiednią komendę poprzez SMS z numeru właściciela.

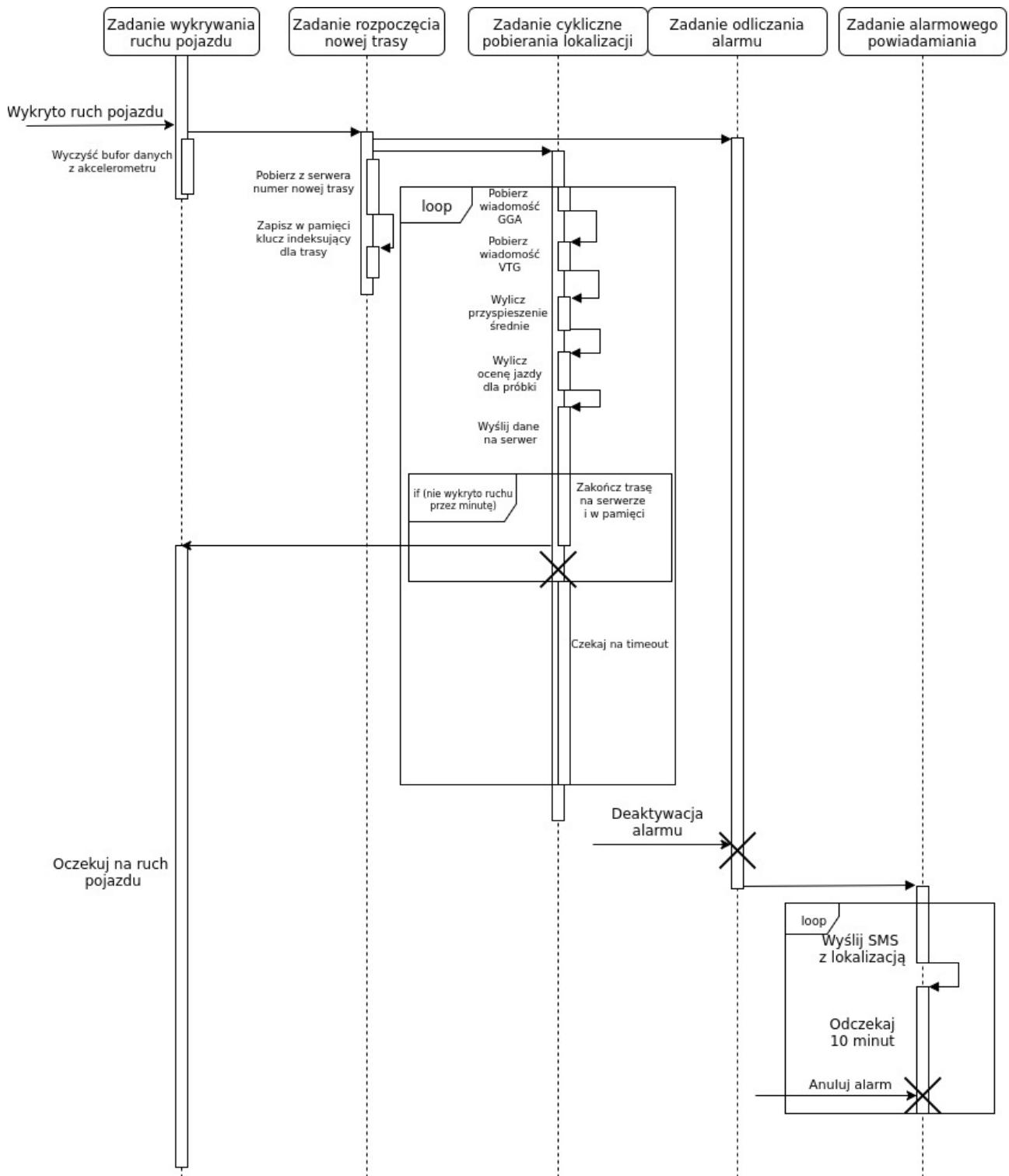
Kolejne zadanie to alarm w Pierwszy krok w procesie deaktywacji alarmu jest wykonywany przez płytke główną. Ze względu na fakt, iż urządzenie powinno być ukryte, nie powinno ono rozmawiać żadnych pakietów w sposób ciągły, co zminimalizuje jego wykrywalność w aplikacjach skanujących poprzez Bluetooth Low Energy. Z tego powodu, w momencie wykrycia ruchu pojazdu, to ono nawiązuje połączenie z *Key Tag'iem*. W tym celu dokonuje skanowania urządzeń posiadających odpowiedni zestaw serwisów i charakterystyk. Dla każdego z nich, sprawdza jego nazwę. Jeśli urządzenie nazywa się odpowiednio, dopiero wówczas nawiązywane jest

z nim połączenie. W trakcie połączenia, urządzenie peryferyjne musi przesłać kod deaktywujący (wygenerowany w trakcie parowania), zaszyfrowany kluczem szyfrowania wygenerowanym na potrzeby danego połączenia. Jeśli zostanie on poprawnie odszyfrowany, alarm jest deaktywowany. W przeciwnym razie - połączenie zostaje zerwane. Algorytm deaktywowania alarmu przedstawiono w rozdziale 6.2 na rysunku 6.4.

Ponadto, niezależnie od tego czy alarm został deaktywowany czy nie, uruchamiane jest zadanie cyklicznego pobierania próbek lokalizacji. Okres próbkowania wynosi 10 sekund i w momencie pobierania informacji gromadzone są dane takie jak:

- Status lokalizacji
- Lokalizacja pojazdu
- Prędkość pojazdu
- Średnie przyspieszenie pojazdu z okresu pomiędzy próbami
- Azymut ruchu
- Ocena jazdy z okresu pomiędzy próbami
- Parametr HDOP informujący o jakości sygnału GPS
- Liczba satelitów z których odebrano sygnał
- Czas pobrania próbki

Na rysunku 6.2 przedstawiono diagram interakcji, który pokazuje przepływ sterowania pomiędzy zadaniami.

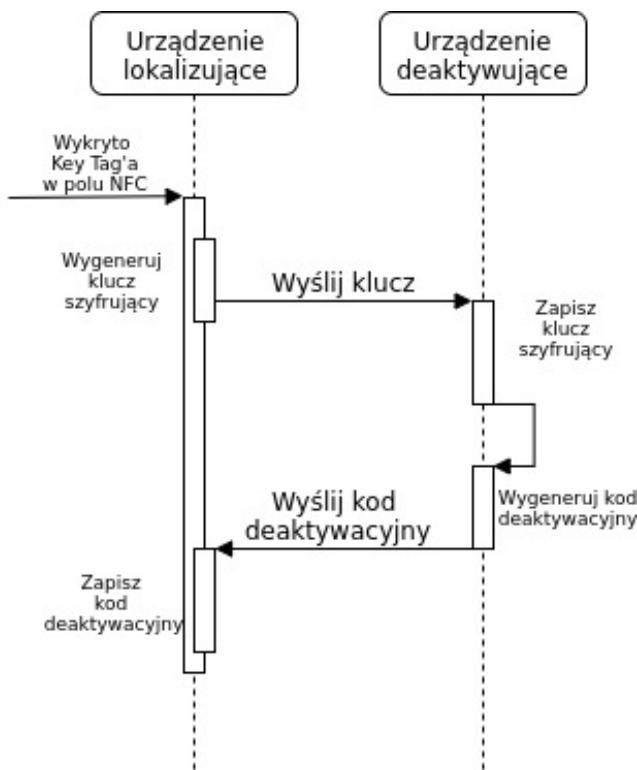


Rysunek 6.2: Przepływ sterowania w przypadku wykrycia ruchu pojazdu.
 Źródło: Twórczość własna

6.2 Urządzenie deaktywujące

Urządzenie deaktywujące stanowi najprostszy moduł w całym systemie. W jego obrębie nosi miano *Key Tag'a*, ze względu na założenie, iż będzie się ono znajdować przy kluczach pojazdu. Z racji tego, że posiada on tylko jedną, ale jakże istotną funkcję - deaktywację alarmu, jego główną cechą powinna być energooszczędność. Z tego powodu, urządzenie to pozbawione jest zewnętrznych układów poza anteną BLE i przez większość czasu znajduje się w trybie oszczędzania energii.

Aby system w samochodzie stał się operacyjny, pierwszym krokiem jest sparowanie ze sobą modułu lokalizującego z przeznaczonym dla niego urządzeniem deaktywacyjnym. Proces ten polega na wygenerowaniu przez urządzenie lokalizujące 16-bajtowego klucza szyfrującego dla algorytmu AES128, oraz 16-bajtowej komendy deaktywującej po stronie *Key Tag'a*. Losowanie komendy deaktywującej zamiast wprowadzenie jej jako stałej do pamięci mikrokontrolera wszystkich urządzeń stanowi dodatkowe zabezpieczenie systemu. Jak zostało przedstawione we wcześniejszych rozdziałach, operacja parowania realizowana jest poprzez interfejs NFC. Przedstawiono ją na rysunku 6.3.

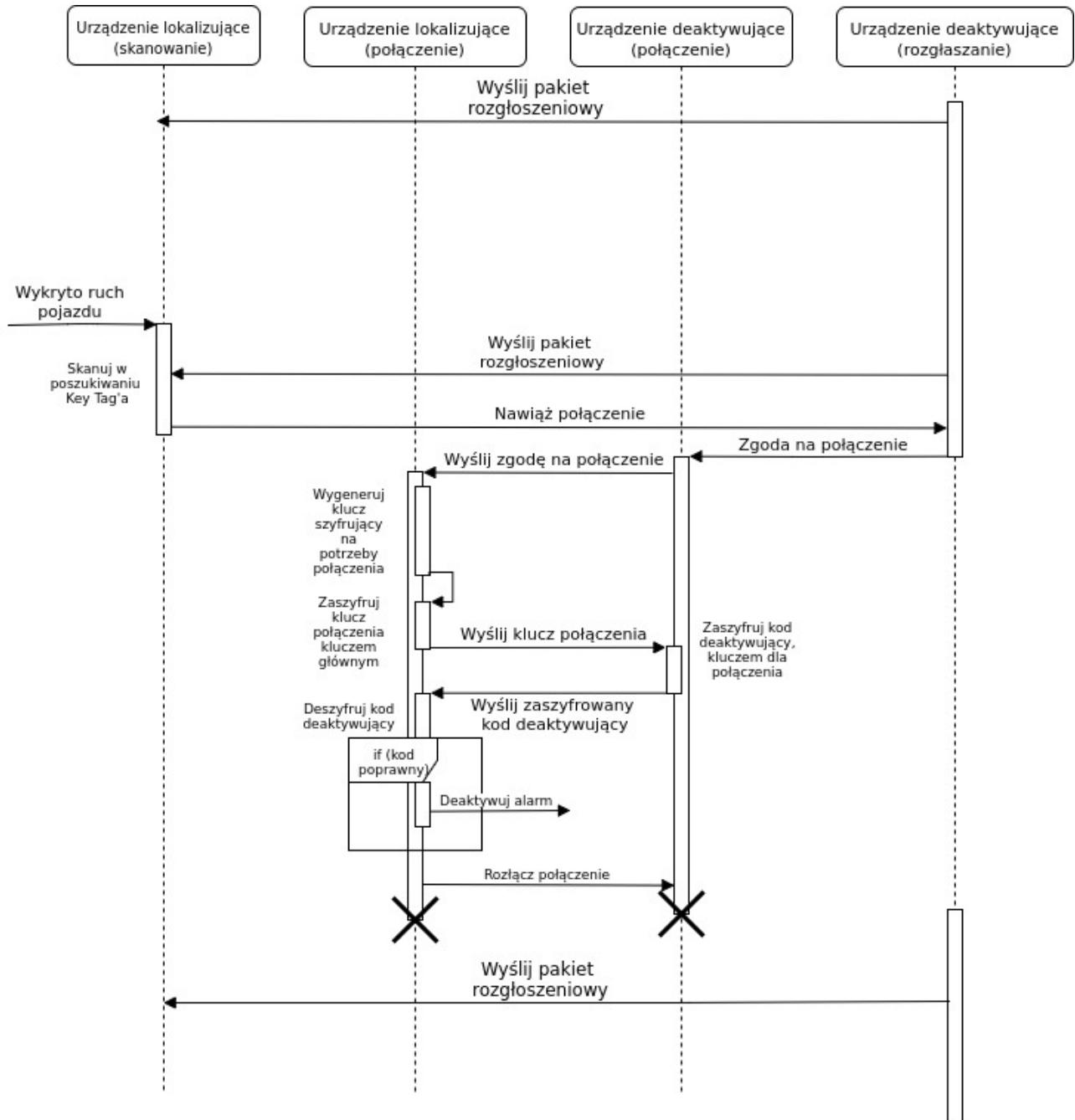


Rysunek 6.3: Przepływ sterowania w trakcie parowania urządzenia lokalizującego z urządzeniem deaktywującym.

Źródło: Twórczość własna

W momencie wykrycia ruchu pojazdu, urządzenie lokalizujące uaktywnia mechanizm skanowania urządzeń wykorzystujących BLE, w poszukiwaniu *Key Tag'a*. Gdy znajdzie urządzenie o pasującej specyfikacji (nazwa oraz struktura serwisów i charakterystyk), łączy się z nim i generuje tymczasowy 16-bajtowy klucz szyfrujący na potrzeby aktualnego połączenia, który jest następnie szyfrowany kluczem głównym i wysyłany do urządzenia deaktywującego. W kolejnym kroku *Key Tag* dokonuje deszyfrowania klucza tymczasowego, i zaszyfrowania nim kodu deaktywującego, wylosowanego na etapie parowania. Jest on przesyłany z powrotem do urządzenia lokalizującego, które sprawdza czy klucz jest poprawny i jeśli tak - deaktywuje alarm. Jeśli nie, alarm pozostaje aktualny. W każdym wypadku, po przesłaniu klucza, połączenie zostaje przerwane, a urządzenie lokalizujące wyłącza skanowanie. Zostanie ono włączone dopiero gdy zostanie wykryty nowy ruch pojazdu. Schemat operacji deaktywacji przedstawiono na rysunku 6.4.

Dzięki takiemu rozwiązaniu, następuje minimalizacja liczby nawiązywanych połączeń. Jest to korzystne z punktu widzenia obu urządzeń, ponieważ połączenie poprzez Bluetooth Low Energy stanowi najbardziej energochłonny element komunikacyjny pomiędzy urządzeniami.



Rysunek 6.4: Przepływ sterowania w momencie deaktywacji alarmu.

Źródło: Twórczość własna

6.3 Aplikacja serwerowa

Kolejnym etapem pracy było zaprogramowanie aplikacji serwerowej, która obsługiwałaby zapytania HTTP (*ang. Hypertext Transfer Protocol*) użytkownika oraz zapytania do bazy danych. Aplikacja została napisana przy pomocy framework'a Qt. Jest to ostatnimi czasy jedno z najpopularniejszych zbiorów bibliotek do języka C++. Wybór środowiska nastąpił z kilku powodów. Pierwszym i jednocześnie najważniejszym z nich jest wbudowany moduł obsługi relacyjnych baz danych. Dzięki temu, wykorzystując kilka wysokopoziomowych funkcji można szybko operować na zgromadzonych danych. Ponadto, aplikacja napisana w C++ statystycznie zapewnia większą wydajność niż podobna napisana w języku Java. Ostatnim z powodów jest wieloletnia znajomość tej biblioteki oraz jej metodologii komunikacji wewnętrznej przez autora pracy.

Aplikacja składa się z dwóch głównych modułów: manager'a bazy danych oraz serwera HTTP.

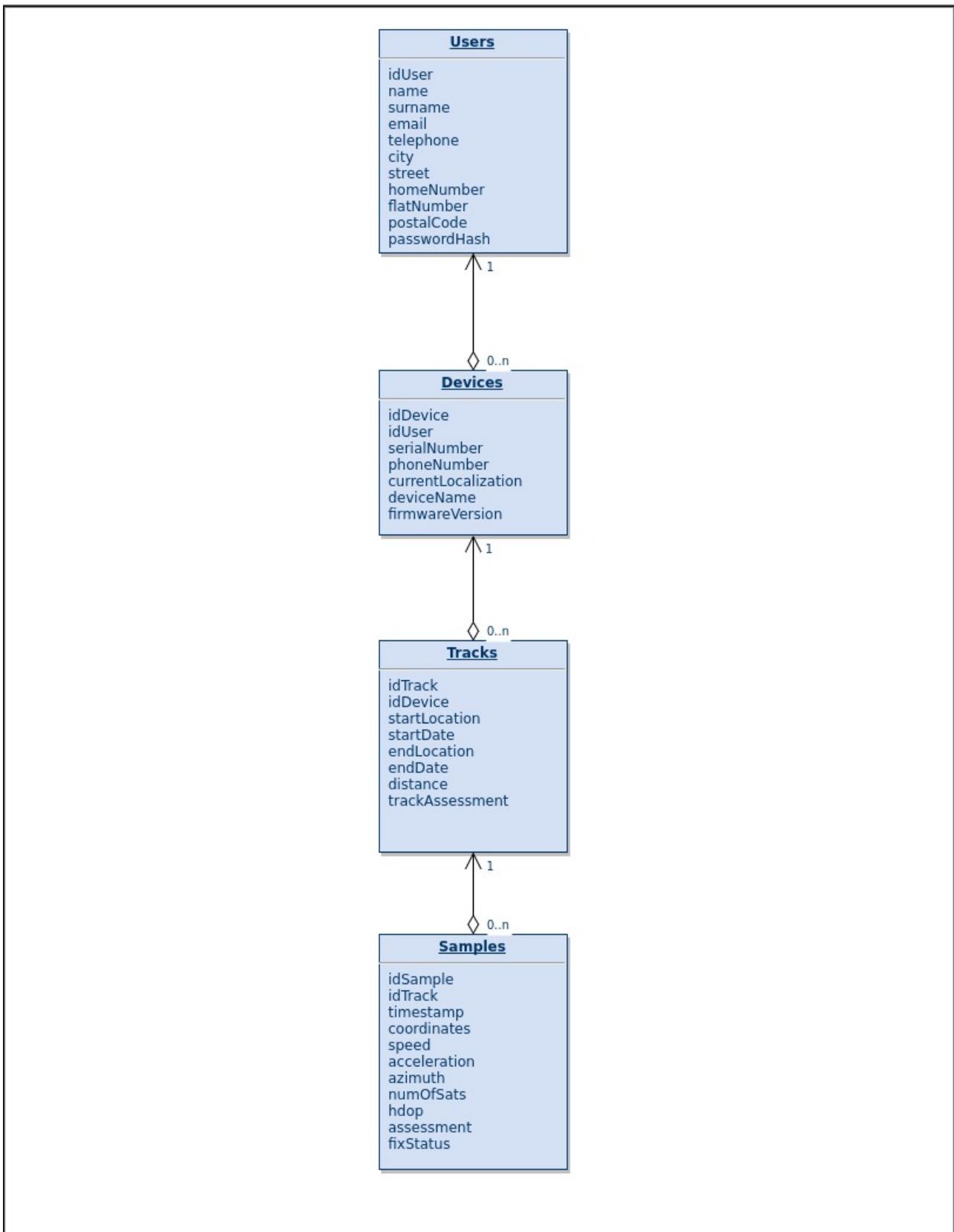
Baza danych składa się z 4 tabel. Są to :

- Tabela użytkowników
- Tabela urządzeń
- Tabela tras
- Tabela próbek

Jako silnik bazodanowy zdecydowano się wykorzystać system SQLite. Stanowi on uproszczony, lecz bardzo wydajny sterownik opierający się na zapytaniach SQL, służący do obsługi relacyjnych baz danych. Kolejną zaletą jest fakt, iż jest on obsługiwany wewnętrznie przez bibliotekę Qt. Relacje pomiędzy poszczególnymi tabelami bazy danych zostały przedstawione na rysunku 6.5.

Założenia struktury bazy danych są następujące:

- Każdy zarejestrowany użytkownik może posiadać więcej niż jedno urządzenie
- Każde urządzenie może otrzymać opisującą je krótką nazwę
- Do każdego z urządzeń może być przypisana więcej niż jedna trasa
- Wpis trasy posiada wpisy o miejscach i czasach ich rozpoczęcia i zakończenia, oraz zbiór przypisanych do niej próbek lokalizacji zbieranych cyklicznie w czasie jej trwania.



Rysunek 6.5: Schemat relacji między tabelami w bazie danych.
 Źródło: Twórczość własna

Jako serwer HTTP zastosowano bibliotekę QtcpServer autorstwa użytkownika supamii[21]. Została ona napisana pod licencją MIT, co zapewnia swobodę użytkowania i modyfikacji kodu źródłowego, a nawet komercyjne zastosowanie pod warunkiem umieszczenia oryginalnych warunków licencyjnych i informacji o autorze. Biblioteka umożliwia komunikację zarówno poprzez zapytania HTTP typu GET jak i POST. Zapytania te różnią się pomiędzy sobą tym, że w zapytaniu typu GET zmienne przekazywane są jawnie wewnątrz adresu URL, natomiast w zapytaniu typu POST są one ukryte.

Przy pomocy aplikacji, można wykonać następujące operacje:

- Logowanie użytkownika
- Wylogowanie użytkownika
- Rejestracja nowego użytkownika
- Pobieranie danych o użytkowniku z bazy danych
- Zmianę danych użytkownika
- Zmianę hasła użytkownika
- Kasowanie konta użytkownika
- Dodawanie urządzenia do konta użytkownika
- Pobieranie listy urządzeń przypisanych do użytkownika
- Pobieranie informacji o urządzeniu
- Usuwanie urządzenia z bazy danych
- Dodawanie nowej trasy do urządzenia
- Pobieranie listy tras przypisanych do urządzenia
- Pobieranie informacji o trasie
- Dodawanie próbek do trasy
- Zakończanie trasy
- Usuwanie trasy

6.4 Strona internetowa

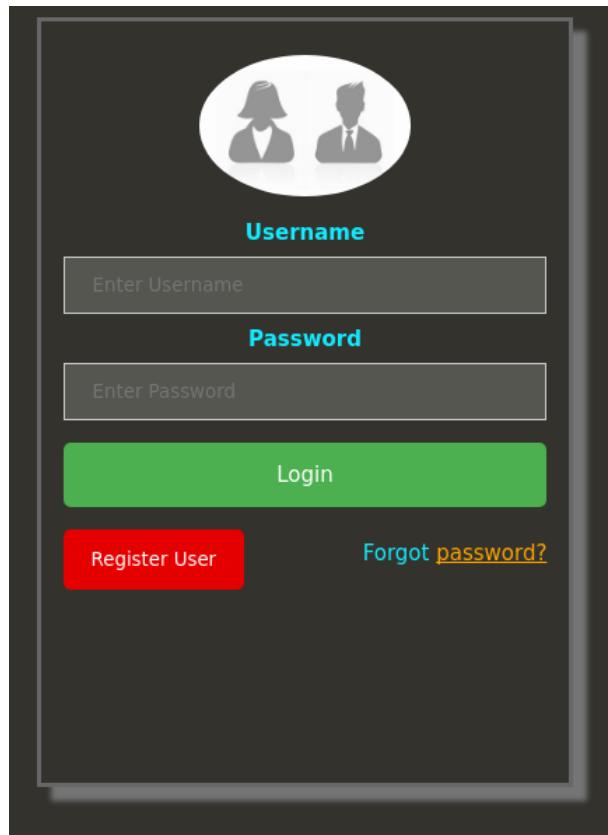
Ostatnim elementem systemu opisywanego w tej pracy jest strona internetowa. Pozwala ona na zdalny i czytelny podgląd danych, które napływają praktycznie w czasie rzeczywistym. Z tego względu stanowi ona równie istotną część pracy co pozostałe. Dzięki wykorzystaniu strony internetowej, możliwe staje się stworzenie konta użytkownika, przypisanie do niego urządzeń, a następnie podgląd tras - zarówno aktualnie przebywanej jak i historycznych wraz z ich parametrami, zapisanymi na serwerze.

Szkielet strony internetowej został zaprojektowany w języku HTML (*ang. Hypertext Markup Language*), natomiast jej część funkcjonalna powstała przy użyciu języka *JavaScript*. Jest to wysokopoziomowy, obiektowo-funkcyjny język skryptowy, wykorzystywany głównie w przeglądarkach, charakteryzujący się jednowątkowością.

Dzięki wykorzystaniu *JavaScript*, możliwe staje się dodanie wielu funkcjonalności do struktury strony internetowej, które stanowią jej połaczenie ze światem zewnętrznym oraz umożliwiają wykorzystanie wielu efektownych rozwiązań wizualnych. Są to na przykład animowane przewijanie strony, wyskakujące okienka czy interaktywne wykresy.

W niniejszej pracy, język *JavaScript* został wykorzystany do połączenia strony internetowej z bazą danych w celu użycia wymienionych w poprzednim podrozdziale funkcji, umożliwienia interakcji użytkownika ze stroną internetową, a także wyświetlenia tras na mapie w postaci znaczników oraz przypisanych do nich okienek informacyjnych.

Na rysunkach 6.6, 6.7, 6.8 oraz 6.9 przedstawiono podstawowe ekranы: logowania, rejestracji użytkownika, ekran główny oraz ekran trasy.



Rysunek 6.6: Strona logowania. Źródło: Twórczość własna

Register User	
Enter Username	Username
Enter Name	Name
Enter Surname	Surname
Enter Email	Email
Enter Telephone number	Telephone Number
Enter City	City
Enter Street	Street
Enter Home Number	Home number
Enter Flat Number	Flat number
Enter Postal Code	Postal code
Enter Password	Password
Repeat Password	Repeat Password

Rysunek 6.7: Ekran rejestracji użytkownika. Źródło: Twórczość własna

Logged in as: konrad@122

Select the device:

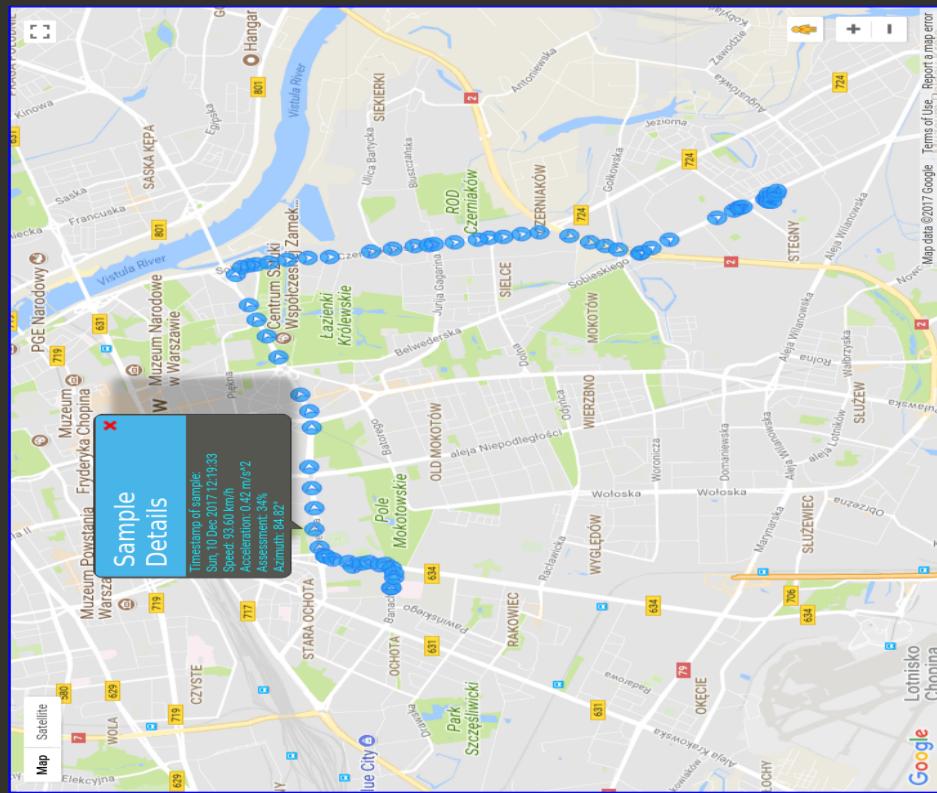
Device number	Device Name	Last known location	Phone Number	Serial Number	Firmware Version
1	Honda	52°10'8055N 21°3'3554	731370554	123456789	1.0.0

Select the track:

Track number	Track start date	Track start coordinates	Track end date	Track end coordinates	Track length [km]	Driving assessment
1	Mon, 04 Dec 2017 21:30:56	52°10'5584N 21°3'2218E	Sat, 04 Nov 2017 21:47:57	52°12.0974N 20°58'6357	12.0km	75%
2	Mon, 04 Dec 2017 22:10:04	52°12.8833N 20°58.6692E	Mon, 04 Dec 2017 22:28:55	52°10.7684N 21°3'3972	11.4km	79%
3	Sat, 09 Dec 2017 10:06:40	52°12.8186N 20°59.2733E	Sat, 09 Dec 2017 10:20:21	52°10.5320N 20°56.5626	6.9km	81%
4	Sat, 09 Dec 2017 18:12:08	52°10'7371N 21°3.5127E	Sat, 09 Dec 2017 18:31:29	52°12.8202N 20°59.2697	10.7km	76%
5	Sat, 09 Dec 2017 18:34:24	52°12.7160N 20°59.2935E	Sat, 09 Dec 2017 18:47:35	52°11.2744N 21°3'17261	9.2km	68%
6	Sat, 09 Dec 2017 19:25:18	52°11.2595N 21°3.7394E	Sat, 09 Dec 2017 19:33:38	52°10.8055N 21°3.3524	1.6km	86%

Rysunek 6.8: Strona główna. Źródło: Twórczość własna

Track Number 9



Rysunek 6.9: Okno trasy. Źródło: Twórczość własna

Ekran trasy przedstawia poszczególne próbki lokalizacji, należące do trasy, na mapie od firmy Google. W tym celu, niezbędne staje się wykorzystanie API producenta dla modułu Google Maps. Pozwala ono na wyświetlenie mapy, przybliżanie i oddalenie, odnalezienie lokalizacji, nanieśenie znaczników (w tym autorskich - zdefiniowanych w formacie grafiki wektorowej - .svg). Ponadto, dzięki zastosowaniu modułu *Info Bubble*, możliwe staje się wyświetlenie na mapie interaktywnego okienka informacyjnego, powiązanego z konkretnym znacznikiem geolokalizacyjnym.

Aby wykorzystać API Google Maps, należy zarejestrować aplikację na stronie <https://developers.google.com/maps/documentation/javascript/get-api-key>. Po rejestracji, do projektu przypisany zostanie klucz, który należy zawrzeć wewnątrz zapytania HTTP wykonywanego przy ładowaniu strony, umożliwiającego ściągnięcie z internetu plików źródłowych zawierających kod obsługujący mapy. Przedstawiono to na listingu 6.4.

Listing 6.4: Fragment kodu pozwalający na użycie API Google Maps

```
<script async defer src="https://maps.googleapis.com/maps/api/js?key=TWoj_KLUCZ_API&callback=initMap" type="text/javascript"></script>
```


Rozdział 7

Analiza stylu jazdy

7.1 Wstęp

W rozdziale przedstawiono opis podstawowej funkcjonalności zaprogramowanego urządzenia, stanowiącej analizę stylu jazdy kierowcy. Zawiera on autorskie badania, krótki opis istniejących rozwiązań oraz propozycję własnego algorytmu oceny sposobu jazdy.

Funkcjonalność opisująca styl jazdy jest niezwykle istotna z punktu widzenia jednej z grup docelowych, do których kierowane jest urządzenie - firm posiadających flotę pojazdów. Wynika to z faktu rosnących kosztów prowadzenia działalności oraz użytkowania pojazdów (w tym wzrostu cen paliwa, części zamiennych i usług). Można do nich zaliczyć nadmiernie szybkie zużycie części eksploatacyjnych, jak na przykład klocki hamulcowe czy opony, a także koszty związane z wypadkami losowymi jak stłuczki. W przypadku firm, są one często generowane przez nieodpowiedzialnych pracowników, którzy nie szanują własności pracodawcy i prowadzą pojazdy w sposób lekkomyślny i agresywny. Ograniczenie tego procederu jest o tyle problematyczne, iż trudno o jednoznaczne dowody winy pracownika - kierowcy. Odpowiadając na tę potrzebę rynkową, opisywany w pracy system pozwala nie tylko na ocenę stylu jazdy i jego zdalne monitorowanie na bieżąco, lecz także na zapisywanie historii ocen przypisanych do punktów przebytej przez pracownika trasy wraz z dodatkowymi parametrami, opisywanymi we wcześniejszych rozdziałach. Pozwala to nie tylko na uzyskanie informacji czy pracownik jechał zbyt agresywnie, lecz także kiedy i gdzie to nastąpiło.

7.2 Istniejące metody

W ramach przygotowania do implementacji algorytmu analizy stylu jazdy, dokonano przeglądu artykułów naukowych, opisujących istniejące już metody. Najciekawszy z nich ([23]) opisuje wykorzystanie telefonu typu smartphone jako platformy czujników pomiarowych. Metoda opisana w artykule jest bardzo podobna do sposobu wykrywania gestów w kontrolerach ruchu dedykowanych do gier. Wykorzystywane są w tym celu dane z akcelerometru oraz żyroskopu, a także system GPS. Pierwsze dwa z nich umożliwiają wykrycie łagodnych i ostrzych skrętów, manewru zawracania, a także przyspieszania i hamowania zarówno gwałtownych, jak i spokojnych. Moduł GPS służy do uzyskania informacji o prędkości pojazdu. Głównym algorytmem wykrywania manewrów jest DTW (*ang. Dynamic Time Warping*), który służy do wyznaczenia miary podobieństwa pomiędzy dwoma sygnałami. Pierwszym krokiem w zastosowaniu algorytmu jest kalibracja telefonu. Autorzy umieszczają telefon na desce rozdzielczej i odpowiednio go orientują względem pojazdu. Działający na nim program dokonuje filtracji danych filtrem dolnoprzepustowym o częstotliwości granicznej 25 Hz ze względu na drgania pochodzące od pracującego silnika. Dane pozyskiwane są w postaci zbioru kilku tysięcy próbek. Pierwszym etapem jest wykrycie rozpoczęcia manewru. W tym celu wykorzystano średnią kroczącą:

$$SMA = \frac{g(i)^2 + g(i-1)^2 + \dots + g(i-k-1)^2}{k} \quad (7.1)$$

gdzie $g(i)$ - wartość próbki przyspieszenia k - liczba próbek w oknie sygnału

Skok cyklicznie wyliczanej w ten sposób średniej powyżej założonego przez autorów progu traktowany jest jako początek manewru. Trwa on dopóki wartość SMA nie spadnie poniżej progu końca manewru. Jeśli czas trwania wykrytego w ten sposób ruchu jest dłuższy niż 15 sekund, jest on traktowany jako błąd pomiaru i odrzucany.

Wykryte w ten sposób manewry poddawane są następnie przetworzeniu przez algorytm DTW. Pozwala on na znalezienie najmniejszej odległości między dwoma sygnałami, czyli stopnia ich podobieństwa (korelacji). Oznacza to, że w pamięci programu zapisane są pewne uśrednione modele wszystkich wykrywanych manewrów, z którymi porównywane są aktualnie przetwarzane dane. Wyliczona korelacja z modelami jazdy agresywnej może służyć za ocenę stylu jazdy.

Metoda ta pozwala na wykrycie wielu różnych manewrów, lecz jest kosztowna obliczeniowo i pamięciowo. Nie jest to problem dla telefonów znajdujących się obecnie na rynku, ale stanowi kluczową kwestię w systemach wbudowanych, posiadających niewielkie zasoby. Ponadto, problemem samego algorytmu jest brak precyzyjnej definicji czym jest manewr łagodny, a czym gwałtowny, przez co bazuje on na subiektywnie wybranych modelach manewrów. Co więcej, fakt zastosowania smartfona powoduje wzrost kosztów systemu, a także konieczność jego cyklicznego

ładowania co znacznie utrudnia możliwość jego ukrycia wewnątrz pojazdu. W związku z faktem iż praca powstała na Wydziale Mechatroniki, która stanowi interdyscyplinarną, synergiczną dziedziną łączącą mechanikę, elektronikę i sterowanie, zdecydowano się na przeprowadzenie własnych badań i zaproponowanie autorskiego rozwiązania, które można byłoby zastosować w opisywanym w pracy systemie.

7.3 Badania

Na ocenę stylu jazdy kierowcy wpływ mają głównie dwa czynniki - prędkość oraz przyspieszenie. Pierwszy z nich niesie informację jak często i o ile kierowca przekraczał limit dopuszczalny prawem. Wykorzystanie tego parametru jest bardzo proste w implementacji, lecz okazuje się kosztowne. W wykorzystywanej w pracy bibliotece do obsługi map od firmy Google istnieje moduł drogowy (Google Maps Road API[22]), jednak w wersji darmowej (prowadzącej dzienne limity zapytań) nie jest udostępniona informacja o ograniczeniach prędkości na drogach. Aby z niej skorzystać należy wykupić licencję Premium. Z tego powodu postanowiono zrezygnować z czynnika przekraczania prędkości w zautomatyzowanej analizie, a wartość bezwzględna szybkości pozostawić do oceny indywidualnej.

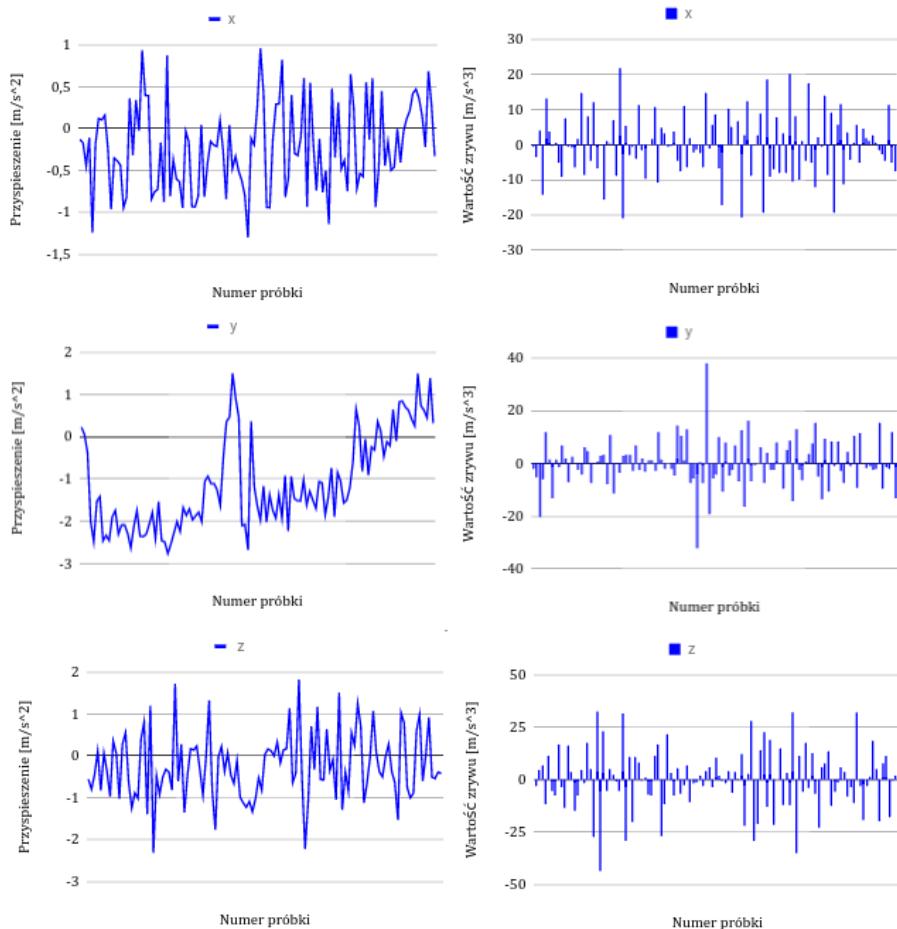
Drugim, znacznie ciekawszym parametrem jest przyspieszenie. Jest ono tak interesujące, ponieważ ma wpływ nie tylko na bezpieczeństwo, lecz także na ponoszone przez pracodawcę koszty. Znaczne przyspieszenie powoduje:

- Zużycie opon w przypadku zerwania przyczepności przy ruszaniu
- Oderwanie odważników wyważających koła, co ma wpływ na komfort jazdy ale również na elementy zawieszenia pojazdu (drgania)
- Zużycie sprzęgła w przypadku agresywnego ruszania
- Duże obciążenie elementów przeniesienia napędu
- Szybsze zużycie elementów wewnętrznych silnika
- Wysokie zużycie paliwa i wzrost zanieczyszczeń wydzielanych do atmosfery
- Zużycie klocków, przegrzanie i wygięcie tarcz hamulcowych w przypadku gwałtownego hamowania
- Możliwość wejścia w poślizg i utraty kontroli nad pojazdem co może skutkować uderzeniem w barierki lub inne pojazdy

Dodatkowo, w ramach rozważań uwzględniono, że wpływ na bezpieczeństwo i ekonomię ma nie tylko wartość akceleracji, lecz także jego zmienność reprezentowana przez zryw, czyli pochodną przyspieszenia po czasie. Z tego powodu postanowiono wykorzystać zamontowany na płytce lokalizatora akcelerometr i zbadać przebiegi przyspieszenia oraz zrywu w osiach X, Y i Z w trakcie wykonywania różnych manewrów na drodze. W każdym z testów poczyniono założenie o odpowiedniej orientacji urządzenia względem pojazdu. Zostało ono w każdym przypadku ustalone tak, aby oś Y pokrywała się z kierunkiem jazdy na wprost, oś Z była umieszczona prostopadle do podłoża, a wynikowo oś X wskazywała kierunek od drzwi do drzwi pojazdu.

W trakcie eksperymentów bardzo istotne było wyeliminowanie wpływu przyspieszenia ziemskiego oraz jego rzutów na osie X i Y, wynikających z niedokładnej orientacji urządzenia. W związku z tym, po uruchomieniu, przez sekundę zbiera ono próbki przyspieszeń, po czym dokonuje ich uśrednienia i zapisuje wyniki w pamięci. Zmierzone w ten sposób wartości są odejmowane od każdej pobranej z akcelerometru próbki. Dzięki zastosowaniu tej metody uzyskano bardzo dokładną kompensację wpływu grawitacji w bezruchu pojazdu. Końcowy wynik był rzędu $0.005 \frac{m}{s^2}$.

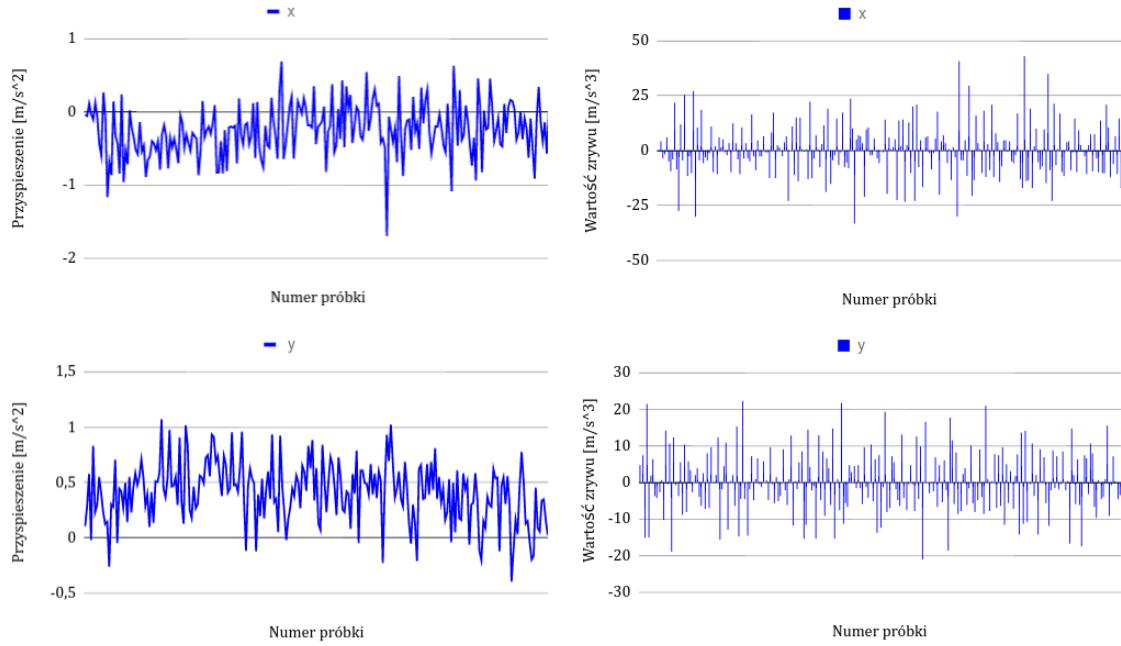
Testy rozpoczęto od najniższej dostępnej częstotliwości próbkowania - 12.5 Hz, w celu osiągnięcia jak najmniejszego zużycia energii przez akcelerometr i ograniczenia liczby niezbędnych do wykonania działań. Wyniki przedstawiono na rysunku 7.1. W przypadku tego testu, ujemna część osi Y skierowana była zgodnie z ruchem pojazdu, oś X wskazywała przyspieszenia boczne, a oś Z - przyspieszenia pionowe.



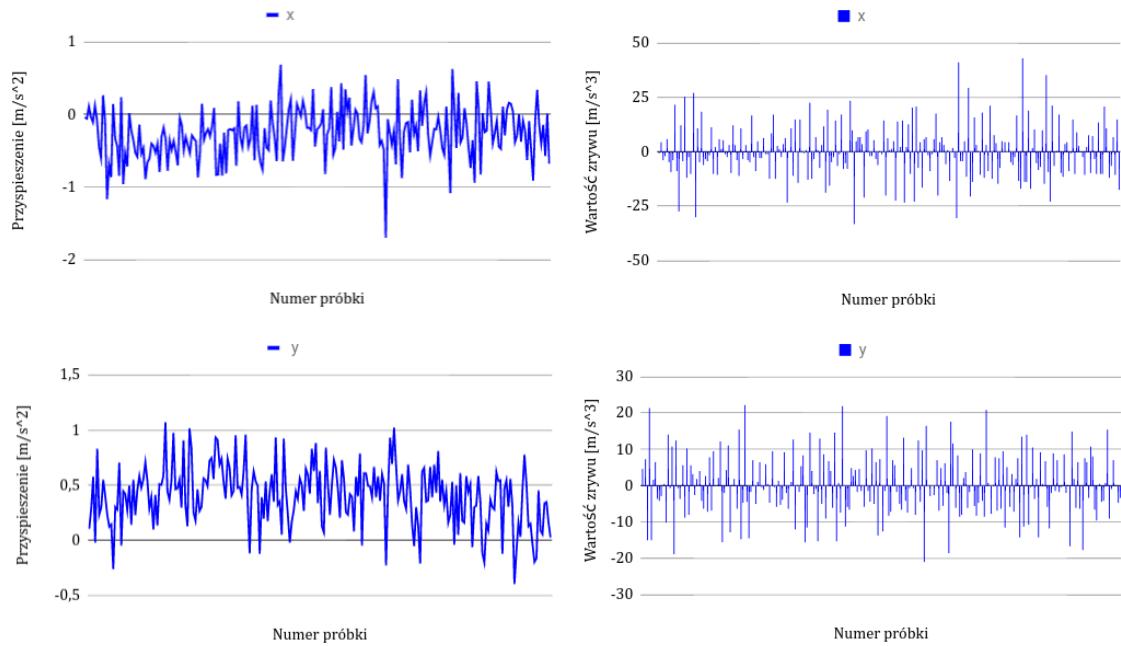
*Rysunek 7.1: Wykresy przyspieszenia i zrywu w trakcie przyspieszania przy częstotliwości próbkowania 12.5 Hz.
 Źródło: Twórczość własna*

Jak widać, dane (zwłaszcza te dotyczące zrywu) wydają się niekompletne, "poszatkowane". W wyniku obserwacji wyników pierwszych testów postanowiono zrezygnować z uwzględniania przyspieszeń w osi Z ze względu na ich znikomy wkład w opis stylu jazdy kierowcy.

Następnym krokiem badań było sprawdzenie wpływu zwiększenia częstotliwości próbkowania o jeden krok - do 26 Hz. W przypadku tego testu skorygowano pomyłkę orientacji urządzenia tak, aby zwrot osi Y pokrywał się z kierunkiem jazdy na wprost. Pozostałe warunki orientacji pozostały bez zmian. Wyniki badań przedstawiono na rysunkach 7.2 i 7.3.



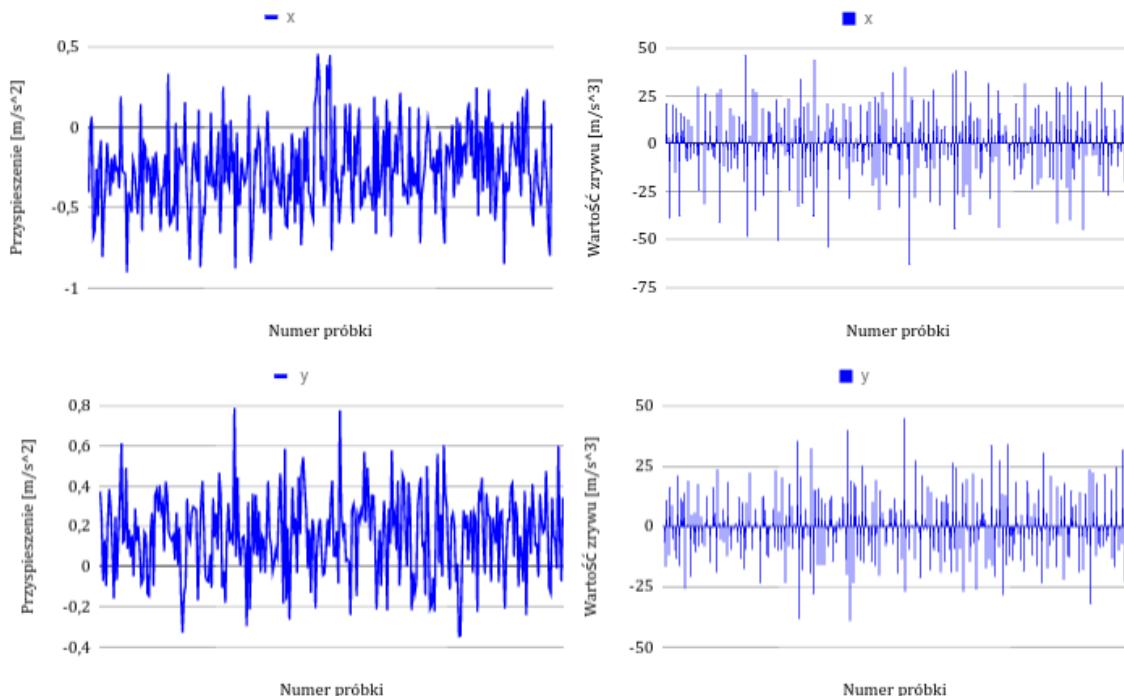
*Rysunek 7.2: Wykresy przyspieszenia i zrywu w trakcie stabilnej jazdy przy częstotliwości próbkowania 26 Hz.
 Źródło: Twórczość własna*



*Rysunek 7.3: Wykresy przyspieszenia i zrywu w trakcie agresywnego przyspieszania i hamowania przy częstotliwości próbkowania 26 Hz.
 Źródło: Twórczość własna*

Jak widać, wzrost częstotliwości bardzo pozytywnie wpłynął na jakość zebranych danych w przypadku przyspieszenia, lecz wyznaczone wartości zrywu wciąż wydają się niekompletne. Kolejnym wnioskiem pochodząącym z obserwacji jest fakt, iż dla agresywnego przyspieszania i hamowania występują różne zakresy osiąganych wartości przyspieszeń. Jest to wynik spodziewany, gdyż zazwyczaj zdolność pojazdów do hamowania jest znacznie większa niż do przyspieszania.

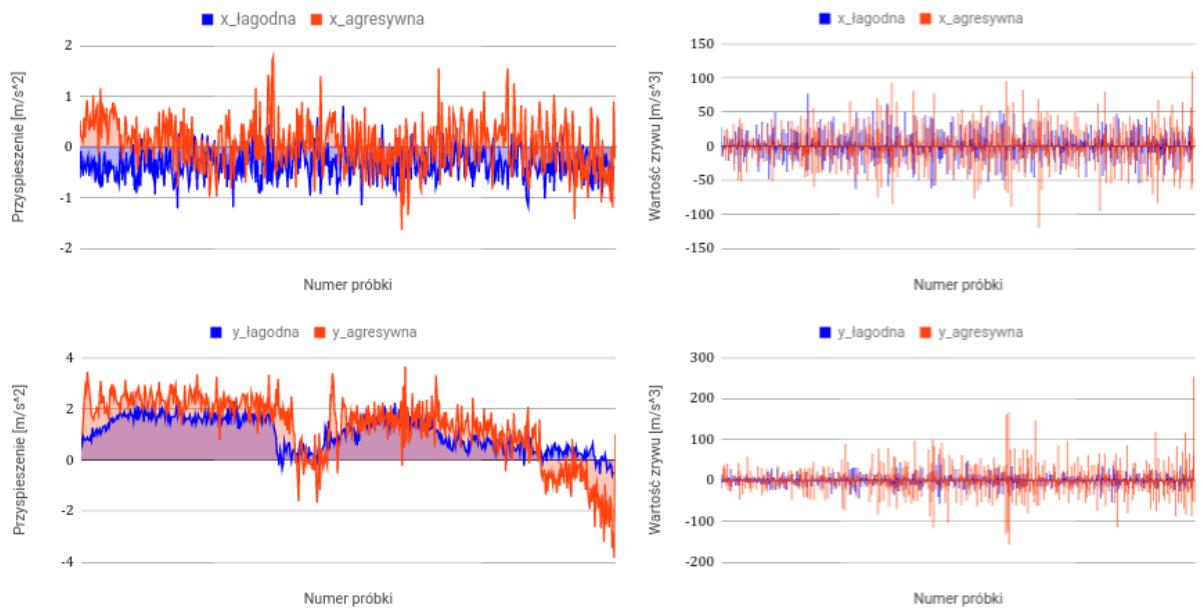
Po zebraniu kilku innych zestawów próbek, postanowiono dokonać dalszego zwiększenia częstotliwości próbkowania o kolejny krok - do 52 Hz. Wyniki dla jazdy na wprost ze stałą prędkością przedstawiono na rysunku 7.4.



Rysunek 7.4: Wykresy przyspieszenia i zrywu w trakcie stabilnej jazdy przy częstotliwości próbkowania 52 Hz.

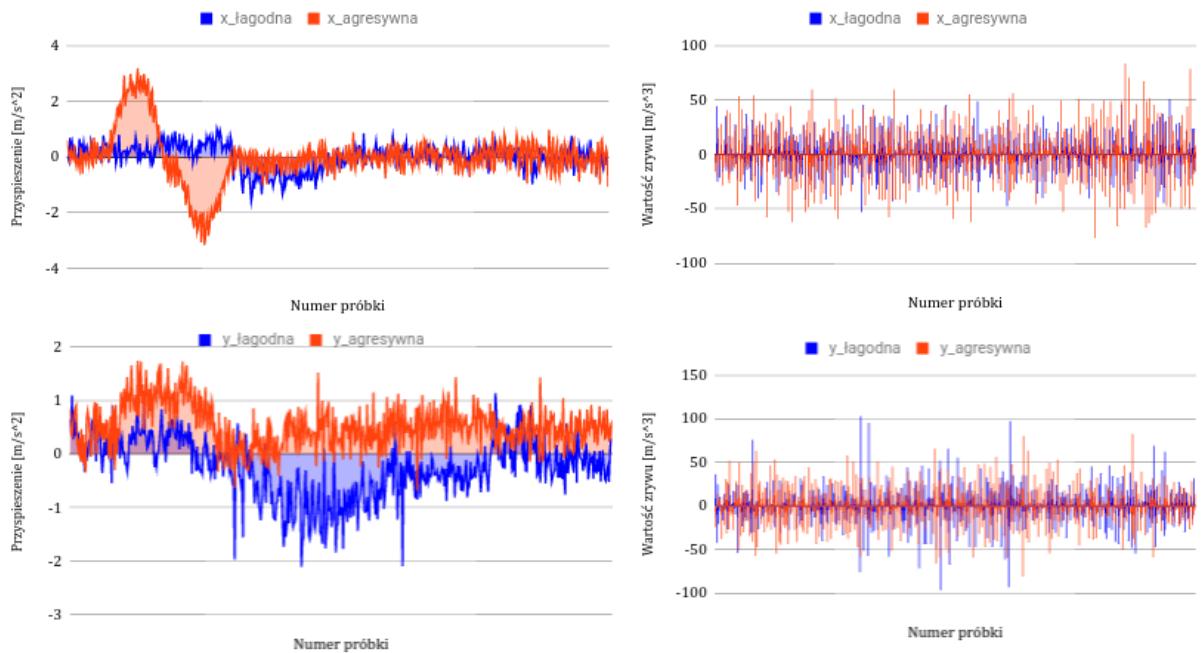
Źródło: Twórczość własna

Na powyższym obrazku widać, że dane są już kompletne, nie wykazują nagłej zmienności. Z tego powodu zdecydowano się na zastosowanie w urządzeniu częstotliwości próbkowania wynoszącej 52 Hz. Częstotliwość ta stanowi dodatkowo kompromis pomiędzy dokładnością danych, a czasem ich przetwarzania i niezbędną ilością pamięci w mikrokontrolerze do ich zgromadzenia. Pozostałe przypadki rozpatrywanych zachowań na drodze przedstawiono na rysunkach: 7.5 i 7.6.



Rysunek 7.5: Zestawienie wykresów przyspieszenia i zrywu w trakcie łagodnego i agresywnego ruszania przy częstotliwości próbkowania 52 Hz.

Źródło: Twórczość własna



Rysunek 7.6: Zestawienie wykresów przyspieszenia i zrywu w trakcie łagodnej i agresywnej zmiany pasa przy częstotliwości próbkowania 52 Hz.

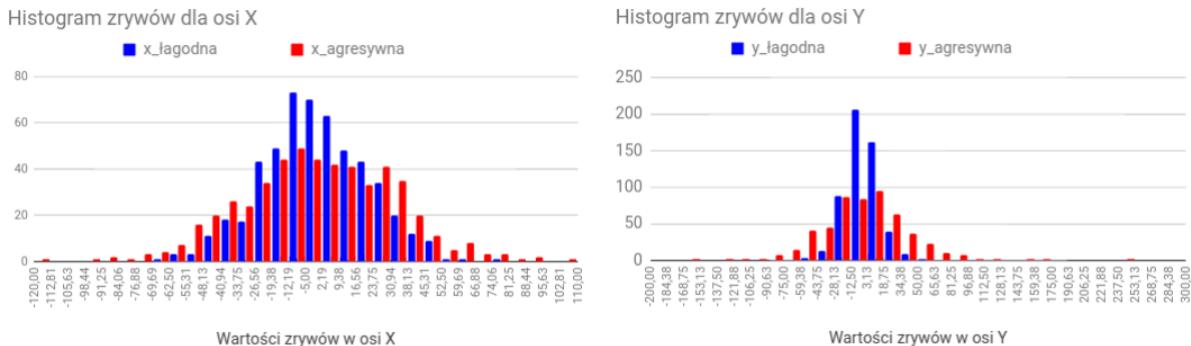
Źródło: Twórczość własna

Obserwując powyższe wykresy można zauważać, że są one na tyle dokładne, iż widać na nich nawet moment zmiany biegu (nagły, krótkotrwały spadek przyspieszenia dla osi Y). Można

również dostrzec, że wartości przyspieszeń w przypadku łagodnego i agresywnego przyspieszania nie różnią się od siebie znacząco, w przeciwieństwie do ich chwilowych zmian, opisywanych przez zryw. Jest to wynikiem rozbieżnością stopni przełożen w skrzyni biegów co powoduje, że na niskich biegach pojazd uzyskuje większe przyspieszenie niż na wysokich. W rezultacie oznacza to, że wartość przyspieszenia nie może stanowić głównego czynnika oceny stylu jazdy.

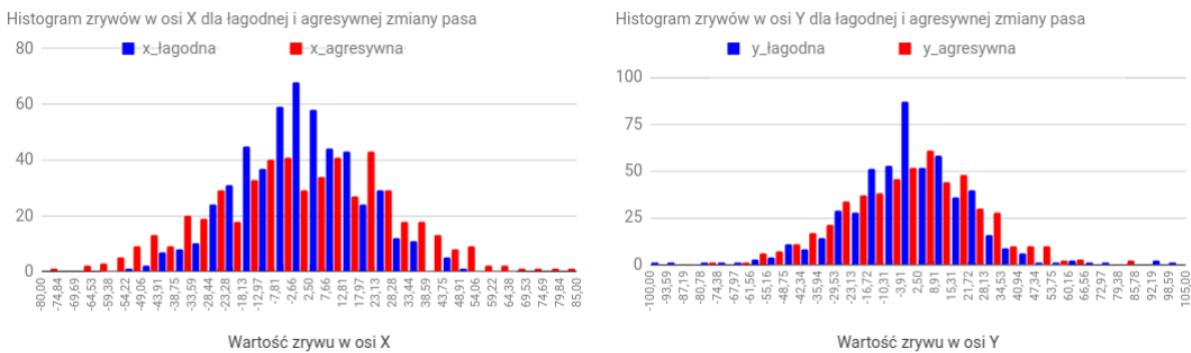
Kolejnym wnioskiem na podstawie obserwacji uzyskanych danych jest fakt, iż w przypadku łagodnej zmiany pasa pojazd zwalniał (ujemne przyspieszenie w osi Y), a w przypadku agresywnej - przyspieszał (dodatnie przyspieszenie w osi Y). Informacja ta jest bardzo ciekawa, gdyż wskazuje na dodatkową korelację pomiędzy stylem jazdy w osiach X i Y przy zmianie pasa ruchu, co powoduje wzmacnienie jego negatywnej lub pozytywnej oceny.

W celu poprawienia widoczności zależności opisującej styl jazdy w danych dotyczących zrywów, parametry te zestawiono na histogramach przedstawionych na rysunkach 7.7 i 7.8.



Rysunek 7.7: Zestawienie histogramów zrywu w osiach X i Y w trakcie łagodnego i agresywnego ruszania przy częstotliwości próbkowania 52 Hz.

Źródło: Twórczość własna



Rysunek 7.8: Zestawienie histogramów zrywu w osiach X i Y w trakcie łagodnej i agresywnej zmiany pasa przy częstotliwości próbkowania 52 Hz.

Źródło: Twórczość własna

Jak widać, w przypadku jazdy agresywnej, histogramy opisujące ruch w osi głównej (dla

zmiany pasa ruchu - oś X, dla jazdy na wprost - oś Y) ulegają spłaszczeniu i rozszerzeniu. Oznacza to, że mniej próbek uzyskuje niską wartość bezwzględną zrywu, a więcej wysoką, w porównaniu do histogramu dla jazdy łagodnej. Stanowi to potwierdzenie tezy o wzroście wariancji pochodnej przyspieszenia wraz ze wzrostem poziomu agresywności stylu jazdy.

Podsumowując, na podstawie danych przedstawionych na powyższych rysunkach otrzymano następujące wnioski:

- Dane nie posiadają znaczącego czynnika losowego w postaci szumu, zatem nie jest konieczne ich dodatkowe filtrowanie.
- Wartość i rozkład przyspieszenia w obrębie okna czasowego, w czasie którego gromadzono próbki bardzo wyraźnie informują o rodzaju wykonywanego manewru.
- Wartość zrywu jest w przybliżeniu symetryczna względem zera. Oznacza to, że jego średnia wartość w całym zakresie próbek jest bardzo niewielka.
- Wraz ze wzrostem poziomu agresji stylu jazdy, wartość wariancji zrywu oraz moduł przyspieszenia średniego rośnie, przy czym pierwszy parametr wykazuje większą zmienność w zależności od sposobu jazdy.

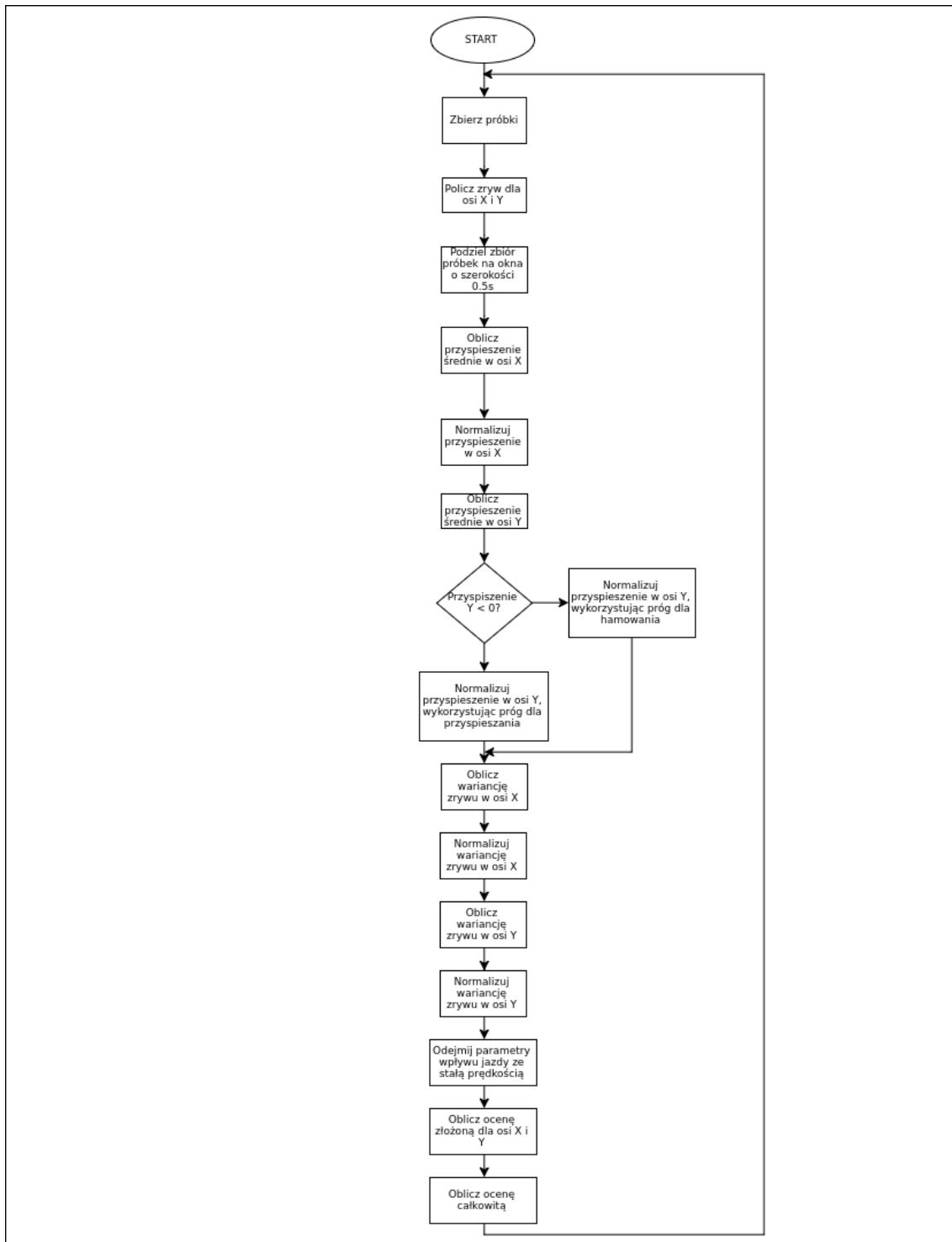
7.4 Metoda zastosowana w pracy

Metoda powstała na podstawie analizy wniosków z podrozdziału 7.3. W swych podstawach wykorzystuje ona przyspieszenie oraz wariancję zrywu. Jedynym jej założeniem jest konieczność odpowiedniej orientacji urządzenia względem kierunku jazdy. Mianowicie, jak już zostało wcześniej wspomniane, oś Y akcelerometru musi pokrywać się z kierunkiem jazdy na wprost, a oś Z powinna być skierowana prostopadle do podłoża. Algorytm jest następujący:

1. Pobierz wartości przyspieszeń zgromadzone w pamięci akcelerometru z okresu pomiędzy próbками lokalizacji (domyślnie 10 sekund).
2. Dla otrzymanego zbioru danych wylicz wartości zrywu dla osi X i Y.
3. Podziel zbiór próbek na okna czasowe o szerokości 0.5 sekundy. Zastosowanie tego kroku pozwala na kwantyzację czasową w celu uśrednienia danych. Szerokość okna została dobrana na podstawie wyników badań z podrozdziału 7.3 tak, aby w wyniku uśredniania nie utracić informacji o krótkotrwałych skokach wartości.
4. Dla każdego okna wylicz przyspieszenia średnie w osiach X i Y, a także wariancję zrywów w tych osiach.

5. Normalizuj każdy z powyższych parametrów, dzieląc go przez pewien dobrany eksperymentalnie próg ustalony dla danej osi, wyróżniając przy tym przyspieszanie od hamowania w kierunku jazdy na wprost, ze względu na konieczność zastosowania odrębnych progów normalizujących.
6. Od każdego ze znormalizowanych parametrów odejmij wartość stałą stanowiącą jego odpowiednik dla jazdy z jednostajną prędkością. Krok ten jest efektem rozumowania, iż agresywny styl jazdy jest niejako stylem łagodnym, z nałożonym na niego pewnym wzorcem ostrej jazdy. Dzięki niemu, oceniany jest jedynie zmienny wpływ czynnika "agresywności", bez uwzględniania wpływu od jazdy ze stałą prędkością.
7. Oblicz ocenę złożoną dla każdej z osi X i Y. Stanowi ona średnią ważoną składowych średniego przyspieszenia znormalizowanego i znormalizowanej wariancji zrywu z wagami wynoszącymi odpowiednio 1 i 2. Wynika to z faktu, iż na podstawie obserwacji danych przedstawionych w podrozdziale 7.3, wariancja zrywu wykazuje lepszą zdolność do klasyfikacji sposobu jazdy.
8. Wyznaczenie oceny dla próbki jako średniej arytmetycznej ocen z osi X i Y.
9. Wyznaczenie oceny całej trasy jako średniej ważonej ocen każdej próbki należącej do tej trasy. Oceny powyżej progu 50% posiadają wagę 1, natomiast te poniżej tej wartości - 2. Ma to na celu wzmocnienie reakcji algorytmu w przypadku agresywnej jazdy tak, aby ocena trasy ulegała bardziej zdecydowanie obniżeniu w przypadku zbyt dynamicznego sposobu jazdy.

Algorytm ten przedstawiono na rysunku 7.9.

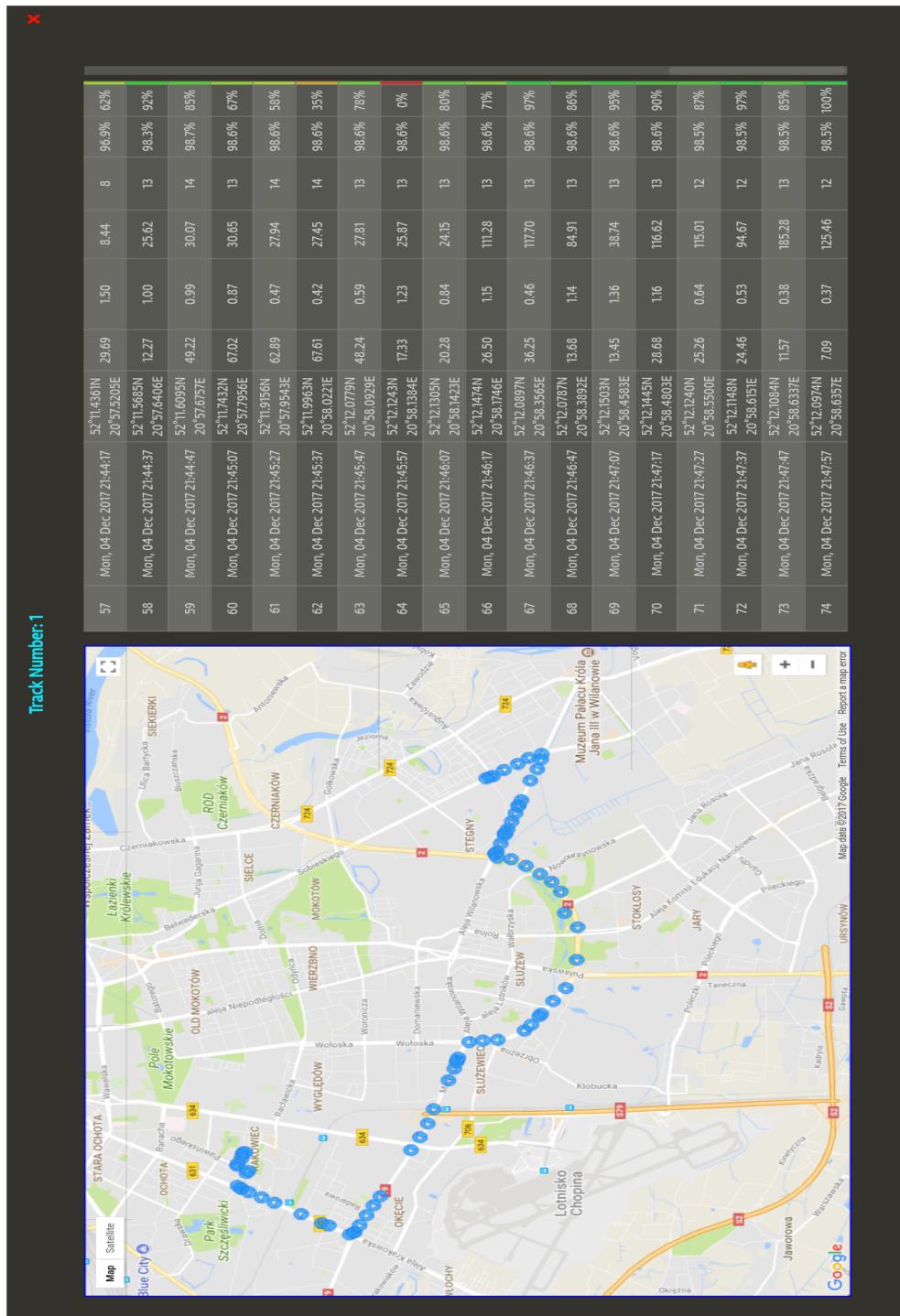


Rysunek 7.9: Algorytm oceny stylu jazdy. Źródło: Twórczość własna

Wyliczona ocena zawiera się w przedziale [0.00; 1,00], gdzie wynik 0.00 oznacza jazdę łagodną, a 1.00 - bardzo agresywną. Na stronie internetowej są one przedstawiane w skali procentowej, gdzie 0% oznacza bardzo agresywną jazdę, a 100% bardzo spokojną. Ponadto, są one wizualizowane na kolorowym pasku, którego barwa płynnie określa ocenę stylu jazdy.

7.5 Rezultaty

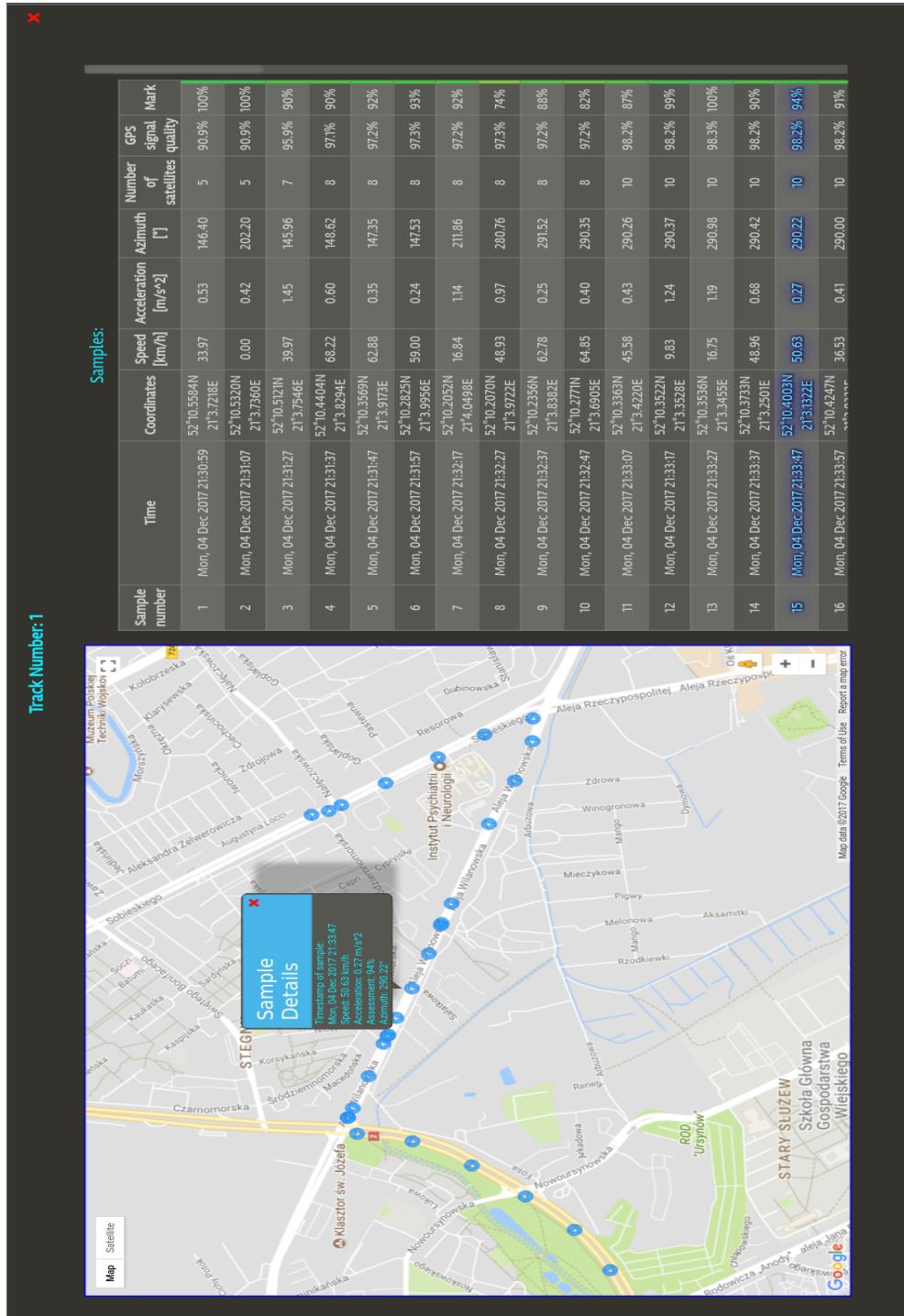
W ramach dokonywania testów skuteczności algorytmu, wykonano kilka przejazdów samochodem po różnych trasach. Jedną z wielu tras przedstawiono na rysunku 7.10.



Rysunek 7.10: Trasa próbna nr 1. Źródło: Twórczość własna

Start nastąpił o godzinie 21:30:59 z ulicy Sobieskiego w Warszawie. Poruszano się w kie-

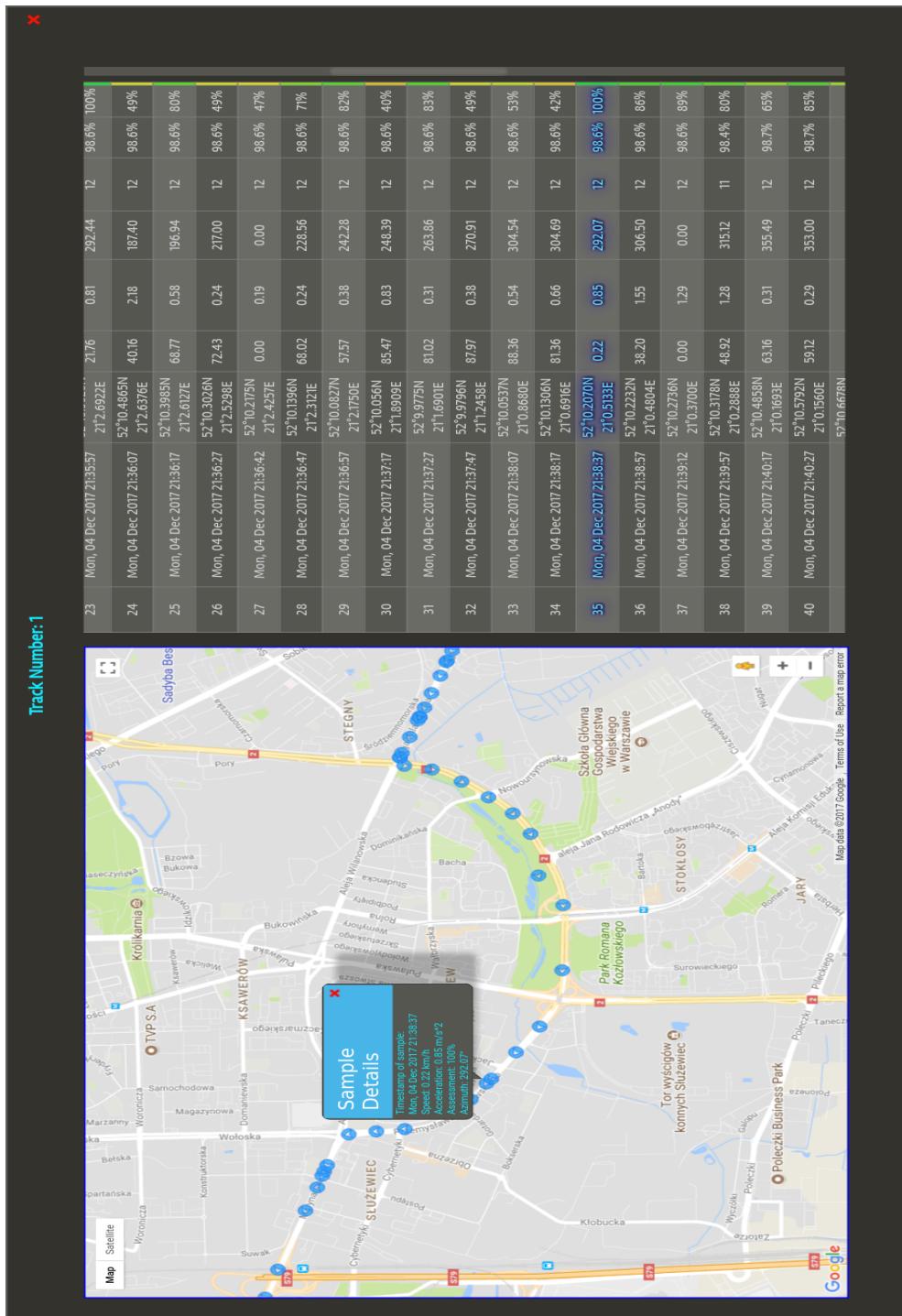
runku Wilanowa, a następnie dokonano skrętu w ul. Wilanowską w kierunku zachodnim aż do skrzyżowania z Doliną Służewiecką. Przejazd do tego momentu był łagodny, bez ostrzych zmian pasów czy przyspieszeń. Wyniki przedstawiono na rysunku 7.11.



Rysunek 7.11: Trasa próbna nr 1 - pierwsza część trasy. Źródło: Twórczość własna

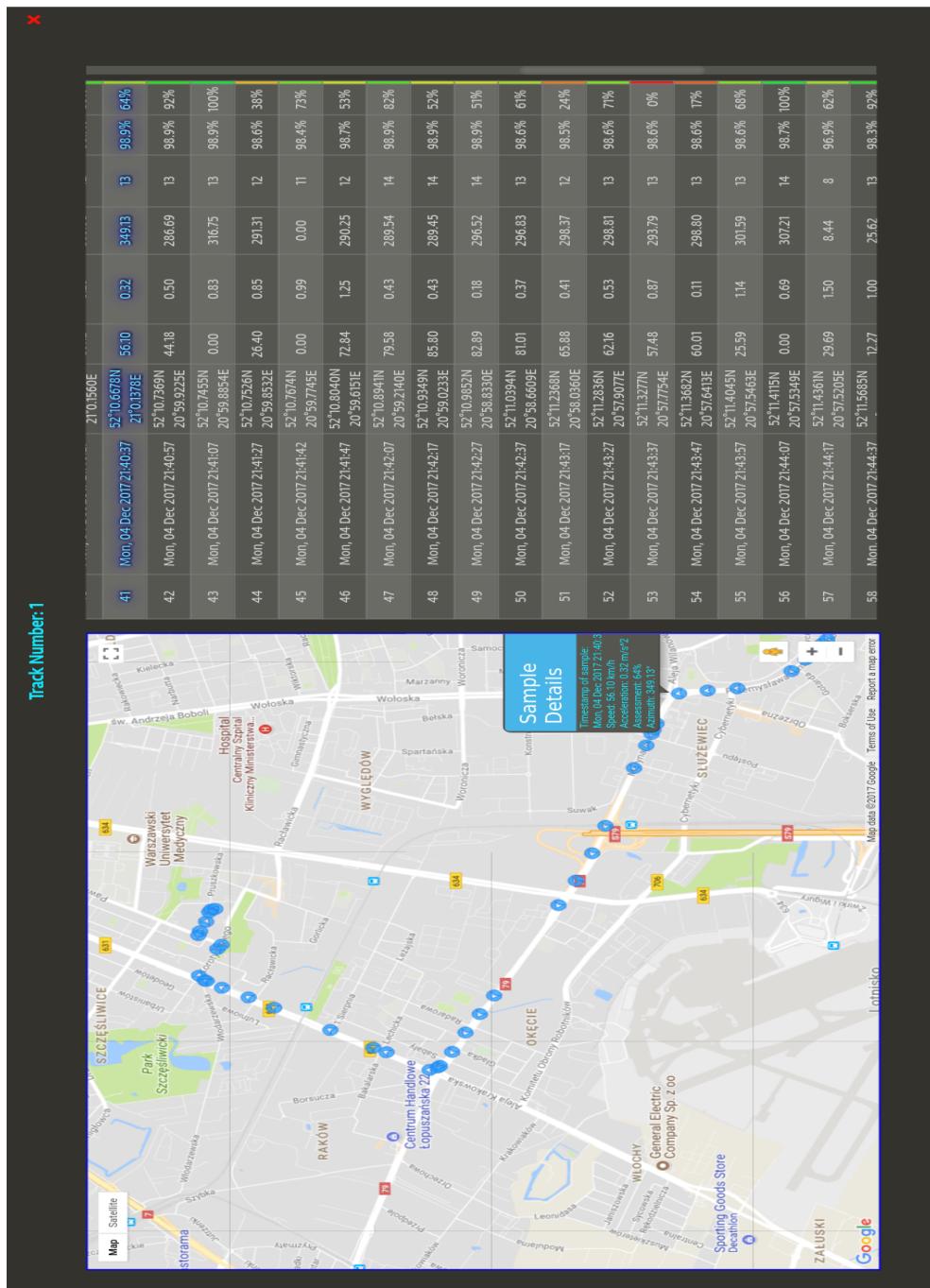
Następnie skręcono w Dolinę Służewiecką, która przeszła w ulicę Rzymowskiego i poruszano

się nią aż do skrzyżowania z ulicą Marynarską. Krótko po rozpoczęciu poruszania się Doliną Ślużewiecką dokonano zwiększenia dynamiki jazdy - bardziej zdecydowanie przyspieszono oraz kilkukrotnie zmieniano pasy ruchu. Manerwy te jednak nie były w stopniu, który można by określić agresywnym, bądź niebezpiecznym. Przedstawiono to na rysunku 7.12.



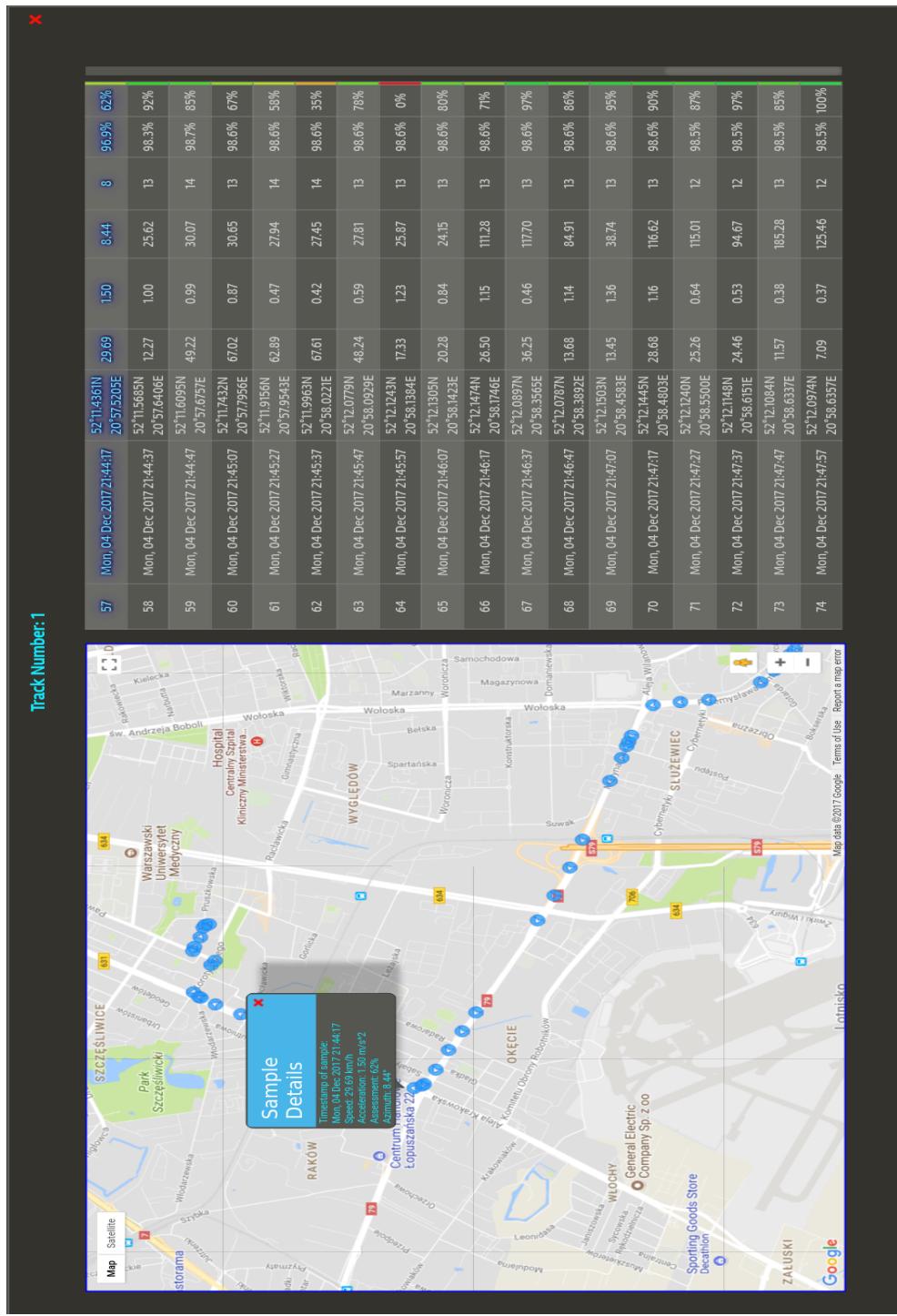
Rysunek 7.12: Trasa próbna nr 1 - druga część trasy. Źródło: Twórczość własna

Kolejnym etapem był zjazd w ulicę Marynarską i jazda nią aż do Alei Krakowskiej. Od pewnego momentu pierwszej z nich, mniej więcej na wysokości trasy S79 występuje podwyższenie dopuszczalnej prędkości do 80 km/h. Tam dokonano testów agresywnej jazdy. Polegały one na cyklicznym, chwilowym, lecz częstym, mocnym przyspieszaniu oraz gdy warunki na to pozwalały - hamowaniu. Ostre hamowanie nastąpiło tuż przed zjazdem w ulicę Aleja Krakowska. Wyniki testów przedstawiono na rysunku 7.13.



Rysunek 7.13: Trasa próbna nr 1 - trzecia część trasy. Źródło: Twórczość własna

Ostatnia część trasy - przejazd ulicami: Aleja Krakowska, Korotyńskiego i Pruszkowska przebiegła już łagodnie, co przedstawiono na rysunku 7.14.



Rysunek 7.14: Trasa próbna nr 1 - czwarta część trasy. Źródło: Twórczość własna

Poniżej w celu zwiększenia przejrzystości danych, zestawiono je w tabeli dla przypadków łagodnej oraz dynamicznej jazdy.

*Tabela 7.1: Zestawienie próbek w przypadku jazdy łagodnej oraz agresywnej.
 Źródło: Twórczość własna*

Czas pomiaru	Prędkość w momencie pomiaru [km/h]	Przyspieszenie średnie z okna pomiarowego [m/s ²]	Ocena stylu jazdy
Jazda łagodna			
04.12.2017 21:30:59	33.97	0.53	100%
04.12.2017 21:31:07	0.00	0.42	100%
04.12.2017 21:31:27	39.97	1.45	90%
04.12.2017 21:31:37	68.22	0.60	90%
04.12.2017 21:31:47	62.88	0.35	92%
04.12.2017 21:31:57	59.00	0.24	93%
04.12.2017 21:32:17	16.84	1.14	92%
04.12.2017 21:32:27	48.93	0.97	74%
04.12.2017 21:32:37	62.78	0.25	88%
04.12.2017 21:32:47	64.85	0.40	82%
04.12.2017 21:33:07	45.58	0.43	87%
04.12.2017 21:33:17	9.83	1.24	99%
04.12.2017 21:33:27	16.75	1.19	100%
Jazda dynamiczna			
04.12.2017 21:36:07	40.16	2.18	49%
04.12.2017 21:36:17	68.77	0.58	80%
04.12.2017 21:36:27	72.43	0.24	49%
04.12.2017 21:36:42	0.00	0.19	47%
04.12.2017 21:36:47	68.02	0.24	71%
04.12.2017 21:36:57	57.57	0.38	82%
04.12.2017 21:37:17	85.47	0.83	40%
04.12.2017 21:37:27	81.02	0.31	83%
04.12.2017 21:37:47	87.97	0.38	49%
04.12.2017 21:38:07	88.36	0.54	53%
04.12.2017 21:38:17	81.36	0.66	42%
04.12.2017 21:38:37	0.22	0.85	100%
04.12.2017 21:38:57	38.20	1.55	86%

Jak widać, wraz ze wzrostem dynamiki jazdy oceny zmalały, co dowodzi skuteczności algorytmu. Ponadto, można dostrzec iż oceny pomiędzy próbками nie są ze sobą powiązane, dając możliwość do szybkiej korekcji stylu jazdy oraz nauki kierowania w sposób ekologiczny.

Rozdział 8

Problemy i ich rozwiązania

Nieodłączną częścią każdego projektu informatycznego i elektronicznego są problemy związane z uruchomieniem. Nie inaczej było w przypadku tej pracy i opisywanego w niej systemu. W trakcie jego rozwoju napotkano szereg różnych komplikacji, z którymi należało sobie poradzić. W rozdziale dokonano wyboru i krótkiego opisu istoty problemu wraz z jego solucją trzech najciekawszych z nich.

8.1 Zawieszanie urządzenia lokalizującego

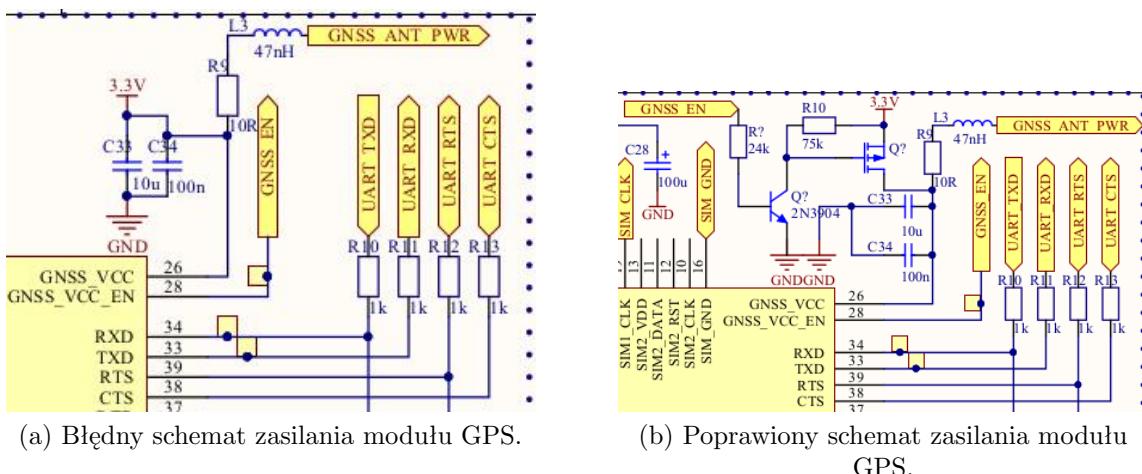
Pierwszy z wybranych problemów dotyczył wczesnych momentów uruchomienia lokalizatora. Polegał on na tym, że mimo poprawnego zaprogramowania mikrokontrolera i działającej poprawnie aplikacji z podłączonym do niej debuggerem, po jego odłączeniu i dokonaniu resetu urządzenia zawieszało się ono gdzieś we wstępnej fazie inicjalizacji. Można było to wywnioskować po fakcie, iż zapalała się dioda LED sterowana programowo, lecz moduł GSM nigdy nie rozpoczynał swej konfiguracji. Po wnikliwym zapoznaniu się z notą katalogową producenta mikrokontrolera okazało się, iż problem stanowił wykorzystywany do odmierzania czasu, wbudowany w rdzeń przez firmę ARM zegar systemowy SysTick. Powodem jego błędnej pracy było każdorazowe wejście mikrokontrolera w tryb oszczędzania energii w trakcie oczekiwania na upłynięcie określonej ilości czasu (*ang. delay*). Powoduje on bowiem zatrzymanie sygnału takującego rdzeń mikrokontrolera, a tym samym - SysTick'a, wykorzystującego zegar systemowy. Efektem jest wówczas praca w pętli i oczekiwanie na upłynięcie losowej ilości czasu. Losowość ta jest wywołana faktem, iż rdzeń wybudzany jest dowolnym przerwaniem na pewien krótki czas, a zatem SysTick "ożywał" na krótko, by znów ulec po chwili zatrzymania. Co ciekawe, problem ten nie występował w trakcie pracy z debuggerem, ponieważ wymusza on ciągłą pracę zegara systemowego niezależnie od trybu oszczędzania energii.

Rozwiązaniem było zastąpienie SysTick'a innym zegarem - wbudowanym w mikrokontroler

bardzo energooszczędnym zegarem RTC (*ang. Real Time Clock*).

8.2 Brak danych z GPS

Kolejny problem związany był typowo z elektroniką. Polegał on na tym, że mimo włączania modułu GPS zgodnie z notą katalogową producenta układu GSM i GPS - poprzez komendę AT wysyłaną przez interfejs UART do pośredniczącego modułu GSM, moduł GPS nie odpowiadał. Ponadto, nie był on responsywny na żadną inną komendę. Problem ten wynikał z błędного zrozumienia noty katalogowej, dotyczącej zasilania modułu GPS. Posiada on bowiem osobny pin, do którego powinno zostać dostarczone zasilanie (*GNSS_VCC*) oraz pin (*GNSS_VCC_EN*), który przyjmuje stan wysoki (napięcie o wartości ok. 3.3 V) gdy GPS jest włączony oraz stan niski (napięcie bliskie 0 V) gdy jest wyłączony. Pin *GNSS_VCC_EN* reagował poprawnie - zmieniał stan po włączeniu modułu GPS komendą AT. Błędnym rozumowaniem okazało się założenie, że jest to jedynie fizyczny wskaźnik statusu zasilania modułu GPS. W praktyce, zamysłem producenta modułu było, aby pin ten wysterowywał tranzystor bądź zewnętrzne źródło zasilania, w celu podania napięcia na pin *GNSS_VCC*. Brak odpowiedzi wynikał z faktu zaburzenia sztywnych zależności czasowych w wewnętrznej komunikacji między modułami GSM i GPS. Rozwiązańem tego problemu było przeprojektowanie zasilania modułu GPS zgodnie z założeniem producenta chip'a. Przedstawiono to na rysunku 8.1.



Rysunek 8.1: Schematy zasilania modułu GPS. Źródło: Twórczość własna

8.3 Kompensacja wpływu przyspieszenia ziemskiego

Występowanie przyspieszenia ziemskiego w systemach pomiaru przyspieszeń bywa często uciążliwe. Problem ten występowałby jedynie dla jednej osi, gdyby urządzenie były idealnie zorientowane względem Ziemi tak, aby oś Z akcelerometru pokrywała się z kierunkiem działania siły grawitacji. Ze względu na fakt, iż jest to praktycznie niemożliwe do osiągnięcia, zwłaszcza w pojazdach, wartość przyspieszenia ziemskiego jest rzutowana nie tylko na oś Z, ale także X i Y. Jest to działanie niepożądane, ponieważ stanowi stałego składnika dodawanego do mierzonych przyspieszeń (*ang. offset*), co zaburza pomiar. W związku z tym, na etapie inicjalizacji urządzenia lokalizującego należy dokonać pomiaru przyspieszeń we wszystkich osiach (przy założeniu bezruchu urządzenia), a następnie od każdej próbki przyspieszenia odejmować te wartości. Pierwotnie, wykorzystywano w tym celu funkcjonalność wbudowaną w moduł akcelerometru. Producent udostępnił w nim rejestrów kompensujących, których zawartości odejmowane są sprzętowo od wyników pomiaru. Rozwiążanie to byłoby idealne, ponieważ ze względu na wykonanie tej operacji przez akcelerometr, nie obciążałoby to programu działającego na mikrokontrolerze. Niestety, problemem okazała się rozdzielcość tych rejestrów. Posiadają one pojemność 8 bitów ze znakiem (wartości od -128 do 127) i programowalną w dwustopniowym zakresie rozdzielcość. Do wyboru są dwie wartości: $2^{-6}g/LSB$ (*ang. Least Significant Bit*) lub $2^{-10}g/LSB$. Po krótkich obliczeniach okazało się, że drugi tryb nie pozwoli na pełną likwidację wartości przyspieszenia ziemskiego, ponieważ:

$$A_{max} = 2^{-10} \cdot 9.81 \frac{m}{s^2} \cdot 127 \approx 1.216 \frac{m}{s^2} \quad (8.1)$$

Gdzie A_{max} wskazuje maksymalną wartość, która może zostać skompensowana. Jak widać, w przypadku tej rozdzielcości nie jest ona dostatecznie duża. Pozostaje więc jedynie wybór $2^{-6}g$:

$$A_{max} = 2^{-6} \cdot 9.81 \frac{m}{s^2} \cdot 127 \approx 19.467 \frac{m}{s^2} \quad (8.2)$$

Jak widać, przy tej rozdzielcości można teoretycznie bezproblemowo usunąć wpływ grawitacji. Jednakże w tym przypadku pojawia się inny kłopot - rozdzielcość pojedynczego bitu:

$$A_{min} = 2^{-6} \cdot 9.81 \frac{m}{s^2} \approx 0.153 \frac{m}{s^2} \quad (8.3)$$

gdzie A_{min} jest wartością przyspieszenia kompensowaną przez pojedynczą jednostkę w wartości rejestru. Maksymalny błędny wynik pomiaru, pomimo kompensacji wpływu grawitacji wynosi około $0.152(9) \frac{m}{s^2}$, co w przypadku analizy stylu jazdy wprowadza znaczny błąd.

Z tego powodu postanowiono zrezygnować z wykorzystania sprzętowej kompensacji i za-

stosować programową. W tym celu, w trakcie inicjalizacji urządzenia, przez sekundę zbierane są próbki przyspieszenia przy założeniu bezruchu pojazdu. Ich wartości w osiach X, Y i Z są uśredniane, a następnie odejmowane od każdej zebranej próbki w przyszłości. Dzięki temu, zmierzony doświadczalnie błąd kompensacji wynosił maksymalnie:

$$A_{min} = 5 \cdot 19.62 \frac{m}{s^2} / 2^{15} \approx 0.003 \frac{m}{s^2} \quad (8.4)$$

Wynik ten jest dwa rzędy wielkości lepszy niż przy sprzętowej kompensacji.

Rozdział 9

Podsumowanie

Celem pracy było zaprojektowanie, wykonanie i oprogramowanie urządzenia stanowiącego dodatkowe zabezpieczenie antykradzieżowe pojazdu, w postaci lokalizatora wykorzystującego system GNSS oraz GSM, zdolnego do analizy stylu jazdy kierowcy, a także całego systemu informatycznego, który pozwoliłby na obsłużenie pozyskanych danych.

W jej ramach zaprojektowano oraz wykonano dwa urządzenia - lokalizujące, stanowiące główny ciężar funkcjonalny pracy oraz pomocnicze, służące do deaktywacji alarmu, czyli autoryzacji ruchu pojazdu. Urządzenia udało się poprawnie uruchomić oraz zaprogramować w celu wykonywania przeznaczonych im zadań. Komunikację pomiędzy nimi zabezpieczono poprzez zastosowanie szyfru AES128. Dodatkowy wzrost bezpieczeństwa został osiągnięty dzięki implementacji algorytmu zmennego klucza szyfrującego.

Urządzenie lokalizujące poprawnie zbiera dane o lokalizacji, prędkości i kursie (azymucie) poprzez wykorzystanie systemu GPS oraz o przyspieszeniu dzięki zastosowanemu w nim akcelerometrowi. W celu przetestowania tej funkcjonalności wykonano szereg testów drogowych, które miały na celu potwierdzenie przydatności urządzenia oraz dokładności lokalizacji. Wszystkie wypadły pozytywnie, pozycjonując pojazd na mapie z dokładnością do 2 - 3 metrów. Co więcej, urządzenie z sukcesem wysyła dane wykorzystując protokół HTTP, a także odbiera i transmituje krótkie wiadomości SMS oraz aktualizuje swój wewnętrzny zegar na podstawie czasu pobranego z sieci GSM.

W celu gromadzenia danych niezbędnych do spełnienia założonych celów, napisano aplikację serwerową w języku C++, która obsługuje bazę danych SQL oraz zapytania HTTP. Służą one zarówno do odbierania danych z urządzenia lokalizującego jak i komunikacji ze stroną internetową napisaną w języku Java Script. Jej zadaniem jest wizualna reprezentacja tras przebytych przez pojazdy, a także powiązanych z nimi danych opisujących ich przebieg upoważnionym do tego użytkownikom. Oba te elementy zostały umieszczone na domowym serwerze w postaci minikomputera Raspberry PI i są dostępne z internetu.

Zwieńczeniem prac były eksperymentalne badania oraz zaproponowanie algorytmu służącego do oceny stylu jazdy kierowców. W trakcie badań wykonano szereg pomiarów, mających na celu ustalenie zależności między przyspieszeniem i jego pochodną - zrywem oraz częstotliwością próbkowania, a sposobem jazdy kierowców. Na ich podstawie z sukcesem zaimplementowano w urządzeniu tę funkcjonalność. Wykonane samodzielnie testy drogowe potwierdziły skuteczność proponowanej metody.

Pomimo spełnienia wszystkich założeń i celów, należy przedstawić pewne możliwości dalszego rozwoju projektu. Aby opisany w pracy system stał się pełnowartościowym produktem, który mógłby trafić do konsumentów, należy wprowadzić kilka dodatkowych elementów. Są to:

1. Aplikacja mobilna na systemy Android i IOS - w dzisiejszych czasach, smartfony zajmują równoważne miejsce w stosunku do komputerów. Są one dodatkowym łącznikiem ze światem technologii, który wydaje się być idealny wręcz do konfiguracji urządzenia, komunikacji z nim i reprezentacji danych.
2. Bootloader - jest to niezbędny fragment programu urządzeń wbudowanych, który pozwala na zdalną aktualizację oprogramowania. O sile produktu, oprócz jego możliwości, stanowi bowiem jego zdolność do szybkiego rozwoju i odpowiedzi na nieprzewidziane wcześniej wymagania i sytuacje.
3. Wykorzystanie biblioteki map drogowych - dzięki temu można jeszcze dokładniej przypisać próbki do dróg i adresów, a także powiązać je z dodatkowymi danymi jak na przykład ograniczenia prędkości obowiązujących na przebytych drogach.
4. Umożliwienia komunikacji alarmowej poprzez wiadomości e-mail - pozwoliłoby to na zmniejszenie kosztów ponoszonych w razie alarmu, ponieważ koszt wysłania e-mail'a jest nieporównanie niższy od kosztu wiadomości SMS.
5. Wykorzystanie systemów GLONASS i Beidou - dzięki temu uzyskano by jeszcze większą dokładność lokalizacji poprzez GNSS.
6. Zaprojektowanie wariantu urządzenia wykorzystującego sieć LTE-M bądź LORA - stosunkowo nowe standardy komunikacji bezprzewodowej. Pierwszy z nich umożliwia zwiększenie przepustowości transmisji, a drugi - jej energooszczędności.
7. Wprowadzenie do algorytmu analizy stylu jazdy zdolności "uczenia", czyli funkcjonalności samoczynnej aktualizacji zaprogramowanych progów określających czułość oceny.

Podsumowując, uważam opisywany w niniejszej pracy projekt za zakończony w pełni sukcesem. Zrealizowano wszystkie założone cele, pokonano wszystkie napotkane problemy, zgromadzone dane posiadają dużą dokładność, a tym samym potencjalną wartość rynkową. Mocną

stroną systemu są jego ograniczone wymiary, możliwość podglądu lokalizacji pojazdów na bieżąco oraz całkowita niezależność od instalacji elektrycznej pojazdu. Dzięki zastosowaniu funkcji oceny stylu jazdy, a także alarmu w przypadku kradzieży stanowi produkt przewyższający dostępna na rynku konkurencję.

Bibliografia

- [1] Spark Nano. *Spark Nano 5.0 GPS Tracker*. https://ii.brickhousesecurity.com/fcgi-bin/iipsrv.fcgi?FIF=/images/brickhousesecurity//source/GPS-SN5_6.tif&wid=335&cvt=jpeg.
- [2] My Car Tracks. *Aplikacja na smartphone MyCarTracks*. <https://www.mycartracks.com/features;jsessionid=3D9E99856BB52CB325C6D699AA3EF5F0.mct-node-one>.
- [3] STI. *STI GL300 GPS Tracker*. https://images-na.ssl-images-amazon.com/images/I/81Xv5REeNxL._SL1500_.jpg.
- [4] Agilent Technologies. *GSM presentation*.
http://mars.merhot.dk/w/images/8/88/GSM_praesentation_noter.pdf, 2000.
- [5] Tutorials Point Pvt. Ltd. *GLOBAL SYSTEM FOR MOBILE COMMUNICATIONS (GSM) TUTORIAL*.
<https://www.tutorialspoint.com/gsm/gsmTutorial.pdf>, 2014.
- [6] *Cellular network architecture image*.
https://en.wikipedia.org/wiki/Cellular_network.
- [7] Zogg J. M. U-Blox AG. *GPS Essentials of Sattelite Navigation Compendium*, 2009.
- [8] Konrad Traczyk. *Projekt personalnego urządzenia śledzącego pozycję geograficzną, sprzązonego z aplikacją mobilną*, 2016.
- [9] Obrazek prezentujący strukturę bitów i poziomy napięć w interfejsach rs232 i rs422.
<http://www.bb-elec.com/Images/whitepaper-images/DataByte.aspx>.
- [10] Tiger Chen. *MC60 Series Hardware Design*. Quectel, Shanghai, China, May 2017.
- [11] Townsend K., Cuff C., Davidson A., and R. *Getting started with Bluetooth Low Energy*. O'Reilly Media, 2014.

- [12] Igoe T., Coleman D., and Jepson B. *Beginning NFC. Near Field Communication with Arduino, Android and PhoneGap*. O'Reilly Media, 2014.
- [13] *Obrazek przedstawiający budowę tag'a RFID.*
<http://1oomzzme3s617r8yzr8qutjk.wpengine.netdna-cdn.com/wp-content/uploads/2017/04/RFID-basics-Fig-3.jpg>.
- [14] Leroux E. and NXP. Presentation: Nfc reader design: How to build your own reader, Luty 2015.
- [15] Nordic Semiconductor. *nRF52832 Product Specification v1.4*, October 2017.
- [16] Nordic Semiconductor. *Dokumentacja SDK mikrokontrolerów rodziny nRF5x*.
<https://infocenter.nordicsemi.com/index.jsp>.
- [17] Monika Jaworowska. *Projektowanie płytEK PCB dla linii szybkiej komunikacji*.
<https://elektronikab2b.pl/technika/19012-projektowanie-plytek-drukowanych-z-ukladami-wgYv75-YUW1>, 2013.
- [18] Piotr Górecki. *O paskudztwach i czarodziejach, czyli zakłócenia w układach elektrycznych*.
https://elportal.pl/pdf/2003/edw_2003_09_s20.pdf, 2003.
- [19] Krzysztof Kowalczyk. *Advanced Encryption Standard*.
<http://www.crypto-it.net/pl/symetryczne/aes.html?tab=1>.
- [20] en.wikipedia.org. *Advanced Encryption Standard*.
https://en.wikipedia.org/wiki/Advanced_Encryption_Standard.
- [21] Son-Huy Pham. *QtppServer*.
<https://github.com/supamii/QtppServer/blob/QTTPv1.0.0/LICENSE>.
- [22] Google Inc. *QtppServer*.
<https://developers.google.com/maps/documentation/roads/usage-limits>.
- [23] Derick A. Johnson and Mohan M. Trivedi. *Driving Style Recognition Using a Smartphone as a Sensor Platform*. IEEE, IEEE.

Wykaz skrótów

AES	Advanced Encryption Standard
API	Application Programming Interface
BLE	Bluetooth Low Energy
BTS	Base Tranceiver Station
ESD	Electrostatic Discharge
GATT	Generic Attibute
GCC	GNU Compiler Collection
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GSM	Global System for Mobile Communication
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
ISM	Industrial, Scientific, Medical (pasmo częstotliwości)
LDO	Low Dropout Stabilizer
SDK	Software Development Kit
SMS	Short Message System
SIM	Subscriber Identification Module
UHF	Ultra High Frequency

Spis rysunków

1.1	Schemat blokowy urządzeń wchodzących w skład systemu. Źródło: Twórczość własna	11
1.2	Spark Nano 5.0 GPS Tracker. Źródło: [1]	12
1.3	Aplikacja MyCarTracks. Źródło: [2]	13
1.4	Urządzenie STI GL300. Źródło: [3]	13
2.1	Podział pasma częstotliwości na kanały i okna czasowe. Źródło: [4]	17
2.2	Podział obszaru na komórki. Źródło: [6]	18
2.3	Model działania systemu GPS. Źródło: [7]	20
2.4	Zbiór częstotliwości wykorzystywanych w systemie GPS. Źródło: [8]	21
2.5	Zasada działania systemu GPS. Źródło: [8]	21
2.6	Zasada działania systemu GPS - sygnał referencyjny. Źródło: [8]	22
2.7	Poziomy napięć i kolejność bitów w interfejsach RS232 i RS422. Źródło: [9]	24
2.8	Struktura pasma 2.4 ISM wykorzystywanego przez Bluetooth Low Energy. Źródło: [8]	29
2.9	Model komunikacji rozgłoszeniowej. Źródło: [8]	29
2.10	Model struktury danych serwera GATT. Źródło: [8]	30
2.11	Budowa tag'a RFID. Źródło: [13]	31
2.12	Zasada działania komunikacji pomiędzy urządzeniem aktywnym i pasywnym. Źródło: [14]	32
3.1	Schemat modułu zasilania urządzenia lokalizującego. Źródło: Twórczość własna	38
3.2	Schemat modułu funkcjonalnego urządzenia lokalizującego. Źródło: Twórczość własna	39
3.3	Schemat modułu NFC urządzenia lokalizującego. Źródło: Twórczość własna	40
3.4	Schemat modułu zasilania wejściowego urządzenia lokalizującego. Źródło: Twórczość własna	41

3.5	Schemat przetwornicy impulsowej modułu zasilania urządzenia lokalizującego. Źródło: Twórczość własna	42
3.6	Schemat stabilizatora napięcia modułu zasilania urządzenia lokalizującego. Źródło: Twórczość własna	42
3.7	Schemat modułu mikrokontrolera w urządzeniu lokalizującym. Źródło: Twórczość własna	44
3.8	Schemat modułu układu GSM i GPS w urządzeniu lokalizującym. Źródło: Twórczość własna	45
3.9	Schemat modułu anten dla GSM i GPS w urządzeniu lokalizującym. Źródło: Twórczość własna	46
3.10	Schemat modułu karty SIM w urządzeniu lokalizującym. Źródło: Twórczość własna	46
3.11	Schemat modułu pamięci flash w urządzeniu lokalizującym. Źródło: Twórczość własna	47
3.12	Schemat modułu akcelerometru w urządzeniu lokalizującym. Źródło: Twórczość własna	47
3.13	Schemat części cyfrowej modułu NFC w urządzeniu lokalizującym. Źródło: Twórczość własna	48
3.14	Schemat części analogowej modułu NFC w urządzeniu lokalizującym. Źródło: Twórczość własna	49
3.15	Schemat modułu zasilania urządzenia deaktywującego. Źródło: Twórczość własna	51
4.1	Wygląd górnej warstwy płytki urządzenia deaktywującego oraz jej wizualizacja. Źródło: Twórczość własna	54
4.2	Wygląd dolnej warstwy płytki urządzenia deaktywującego oraz jej wizualizacja. Źródło: Twórczość własna	54
4.3	Tabela opisująca korelację między grubością ścieżek, a maksymalnym dopuszczalnym natężeniem prądu. Źródło: [18]	57
4.4	Wygląd górnej warstwy płytki urządzenia lokalizującego oraz jej wizualizacja. Źródło: Twórczość własna	58
4.5	Wygląd dolnej warstwy płytki urządzenia lokalizującego oraz jej wizualizacja. Źródło: Twórczość własna	59
5.1	Wykonanie operacji Substitute Bytes. Źródło: [20]	63
5.2	Wykonanie operacji Shift Rows. Źródło: [20]	63
5.3	Wykonanie operacji Mix Columns. Źródło: [20]	64
5.4	Wykonanie operacji Add Round Key. Źródło: [20]	64
5.5	Operacja szyfrowania metodą ECB. Źródło: [19]	65

5.6	Operacja deszyfrowania metodą ECB. Źródło: [19]	65
5.7	Operacja szyfrowania metodą CFB. Źródło: [19]	66
5.8	Operacja deszyfrowania metodą CFB. Źródło: [19]	66
6.1	Główny algorytm działania urządzenia. Źródło: Twórczość własna	73
6.2	Przepływ sterowania w przypadku wykrycia ruchu pojazdu. Źródło: Twórczość własna	76
6.3	Przepływ sterowania w trakcie parowania urządzenia lokalizującego z urządzeniem deaktywującym. Źródło: Twórczość własna	77
6.4	Przepływ sterowania w momencie deaktywacji alarmu. Źródło: Twórczość własna	79
6.5	Schemat relacji między tabelami w bazie danych. Źródło: Twórczość własna	81
6.6	Strona logowania. Źródło: Twórczość własna	84
6.7	Ekran rejestracji użytkownika. Źródło: Twórczość własna	84
6.8	Strona główna. Źródło: Twórczość własna	85
6.9	Okno trasy. Źródło: Twórczość własna	86
7.1	Wykresy przyspieszenia i zrywu w trakcie przyspieszania przy częstotliwości próbkowania 12.5 Hz. Źródło: Twórczość własna	93
7.2	Wykresy przyspieszenia i zrywu w trakcie stabilnej jazdy przy częstotliwości próbkowania 26 Hz. Źródło: Twórczość własna	94
7.3	Wykresy przyspieszenia i zrywu w trakcie agresywnego przyspieszania i hamowania przy częstotliwości próbkowania 26 Hz. Źródło: Twórczość własna	94
7.4	Wykresy przyspieszenia i zrywu w trakcie stabilnej jazdy przy częstotliwości próbkowania 52 Hz. Źródło: Twórczość własna	95
7.5	Zestawienie wykresów przyspieszenia i zrywu w trakcie łagodnego i agresywnego ruszania przy częstotliwości próbkowania 52 Hz. Źródło: Twórczość własna	96
7.6	Zestawienie wykresów przyspieszenia i zrywu w trakcie łagodnej i agresywnej zmiany pasa przy częstotliwości próbkowania 52 Hz. Źródło: Twórczość własna	96
7.7	Zestawienie histogramów zrywu w osiach X i Y w trakcie łagodnego i agresywnego ruszania przy częstotliwości próbkowania 52 Hz. Źródło: Twórczość własna	97
7.8	Zestawienie histogramów zrywu w osiach X i Y w trakcie łagodnej i agresywnej zmiany pasa przy częstotliwości próbkowania 52 Hz. Źródło: Twórczość własna	97
7.9	Algorytm oceny stylu jazdy. Źródło: Twórczość własna	100
7.10	Trasa próbna nr 1. Źródło: Twórczość własna	101
7.11	Trasa próbna nr 1 - pierwsza część trasy. Źródło: Twórczość własna	102
7.12	Trasa próbna nr 1 - druga część trasy. Źródło: Twórczość własna	103
7.13	Trasa próbna nr 1 - trzecia część trasy. Źródło: Twórczość własna	104

7.14 Trasa próbna nr 1 - czwarta część trasy. Źródło: Twórczość własna	105
8.1 Schematy zasilania modułu GPS. Źródło: Twórczość własna	108

Spis tabel

2.1	Najczęściej wykorzystywane wiadomości NMEA0183 w odbiornikach GPS. Źródło: [8]	25
2.2	Struktura wiadomości GGA. Źródło: Twórczość własna	26
2.3	Struktura wiadomości VTG. Źródło: Twórczość własna	27
2.4	Podsumowanie cech systemów i protokołów GSM, GPS, BLE oraz NFC. Źródło: Twórczość własna	35
7.1	Zestawienie próbek w przypadku jazdy łagodnej oraz agresywnej. Źródło: Twórczość własna	106

Załączniki

Na załączonej do pracy płycie CD znajdują się następujące treści:

- Niniejsza praca w formacie PDF – plik
Praca/Praca_Magisterska_Konrad_Traczyk.pdf.
- Nota katalogowa mikrokontrolera nRF52832 firmy Nordic Semiconductor – plik
Noty_katalogowe/nRF52832_PS_v1.4.pdf.
- Nota katalogowa przetwornicy impulsowej zastosowanej w urządzeniu lokalizacyjnym firmy Texas Instruments – plik *Noty_katalogowe/lm26003-q1.pdf.*
- Nota katalogowa opisująca projekt elektroniczny modułu GSM i GPS firmy Quectel – plik *Noty_katalogowe/Quectel_MC60_Series_Hardware_Design_V2.0.pdf.*
- Spis komend AT modułu GSM firmy Quectel – plik
Noty_katalogowe/Quectel_MC60_AT_Commands_Manual_V1.1.pdf.
- Spis komend AT modułu GPS firmy Quectel – plik
Noty_katalogowe/Quectel_MC60_Series_GNSS_AT_Commands_Manual_V1.3.pdf.
- Nota katalogowa zawierająca opis protokołu komunikacji modułu GPS firmy Quectel – plik *Noty_katalogowe/Quectel_MC60_Series_GNSS_Protocol_Specification_V1.1.pdf.*
- Nota katalogowa modułu akcelerometru i żyroskopu LSM6DSMTR firmy STMicroelectronics – plik *Noty_katalogowe/LSM6DSMTR_datasheet.pdf.*
- Nota aplikacyjna modułu akcelerometru i żyroskopu LSM6DSMTR firmy STMicroelectronics – plik *Noty_katalogowe/LSM6DSMTR_ApplicationNote.pdf.*
- Nota katalogowa modułu NFC TRF7960A firmy Texas Instruments – plik
Noty_katalogowe/trf7960a.pdf.
- Archiwum zawierające kod programu urządzenia lokalizującego – plik
Kody_źródłowe/Lokalizator.zip

- Archiwum zawierające kod programu urządzenia deaktywującego – plik
Kody_źródłowe/Deaktywator.zip
- Archiwum zawierające kod programu aplikacji serwerowej – plik
Kody_źródłowe/Serwer.zip
- Archiwum zawierające kod programu strony internetowej – plik
Kody_źródłowe/StronaWWW.zip