VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV INFORMAČNÍCH SYSTÉMŮ
DEPARTMENT OF INFORMATION SYSTEMS

NÍZKOENERGETICKÝ GPS LOKÁTOR S POUŽITÍM LORA SÍTĚ

LOW ENERGY GPS TRACKER USING LORA NETWORK

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE — **Bc. MILAN MÚČKA**
AUTHOR

VEDOUCÍ PRÁCE — **Ing. JAN PLUSKAL**
SUPERVISOR

BRNO 2020

Vysoké učení technické v Brně
Fakulta informačních technologií

Ústav informačních systémů (UIFS) Akademický rok 2019/2020

Zadání diplomové práce

Student: **Múčka Milan, Bc.**
Program: Informační technologie Obor: Bezpečnost informačních technologií
Název: **Nízkoenergetický GPS lokátor s použitím LoRa sítě**
Low Energy GPS Tracker Using LoRa Network
Kategorie: Vestavěné systémy

Zadání:

1. Nastudujte možnosti komunikace LoRa sítí a jejich propojení s veřejně dostupnou komunitní sítí The Things Network (TTN).
2. Vyberte vhodnou HW platformu pro vytvoření funkčního vzorku GPS lokátoru, který bude využívat TTN pro zasílání polohových informací.
3. Proveďte návrh funkčního vzorku GPS lokátoru, který konzultujte s vedoucím.
4. Návrh implementujte po konzultaci s vedoucím a dbejte jeho připomínek.
5. Experimentujte s vytvořeným zařízením z pohledu spolehlivosti, dostupnosti a energetické náročnosti. Diskutujte dosažené výsledky.

Literatura:

- Fargas, B. C., & Petersen, M. N. (2017, June). GPS-free Geolocation using LoRa in Low-Power WANs. In *2017 global internet of things summit (Giots)* (pp. 1-6). IEEE.
- Blenn, N., & Kuipers, F. (2017). LoRaWAN in the Wild: Measurements from The Things Network. *arXiv preprint arXiv:1706.03086*.
- Lavric, A., & Popa, V. (2017, June). LoRa Wide-Area Networks from an Internet of Things Perspective. In *2017 9th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)* (pp. 1-4). IEEE.

Při obhajobě semestrální části projektu je požadováno:

- Body 1, 2 a 3.

Podrobné závazné pokyny pro vypracování práce viz https://www.fit.vut.cz/study/theses/

Vedoucí práce: **Pluskal Jan, Ing.**
Vedoucí ústavu: Kolář Dušan, doc. Dr. Ing.
Datum zadání: 1. listopadu 2019
Datum odevzdání: 31. července 2020
Datum schválení: 25. října 2019

Abstrakt

Táto práca sa zaoberá problematikou určovania polohy s veľmi nízkou spotrebou energie. Jej hlavným cieľom bolo navrhnúť a zostrojiť zariadenie používajúce vhodnú platformu a senzoriku pre získanie polohy, následné odoslanie pomocou IoT LoRa siete a spracovanie údajov o polohe na serverovej strane. Pri riešení problému bola využitá rozšírená komunitná sieť TheThingsNetwork. V práci som vytvoril energeticky úsporné zariadenie, ktoré dokáže získať polohu pomocou GPS systému a následne ju doručiť užívateľovi pomocou LoRaWAN siete.

Abstract

This thesis deals with determine location using very low power consumption device. The main target was to design and construct device using right platform and sensors for localization, sending data using IoT LoRa network and than processing data on server's side. To solve the problem, it was used community network TheThingsNetwork. In thesis I created power efficient device, which can find current location using GPS system and deliver it to user using LoRaWAN network.

Kľúčové slová

LoRa, LoRaWAN, OTAA, ABP, TheThingsNetwork, GPS, Úspora energie, STM32

Keywords

LoRa, LoRaWAN, OTAA, ABP, TheThingsNetwork, GPS, Power saving, STM32

Citácia

MÚČKA, Milan. *Nízkoenergetický GPS lokátor s použitím LoRa sítě*. Brno, 2020. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Jan Pluskal

Nízkoenergetický GPS lokátor s použitím LoRa sítě

Prehlásenie

Prehlasujem, že som túto prácu vypracoval samostatne pod vedením Ing. Jan Pluskal. Uviedol som všetky literárne pramene, publikácie a ďalšie zdroje, z ktorých som čerpal.

.....
Milan Múčka
29. júla 2020

Poďakovanie

Chcel by som poďakovať vedúcemu práce Ing. Jan Pluskal za odbornú pomoc a konzultácie v danej téme. Ďalej môjmu ocovi za odbornú diskusiu pri orientácii sa v rádiovej komunikácii a prenosu. Môjmu bratovi ďakujem za vzájomnú podporu a motiváciu počas štúdia.

Obsah

1 Úvod **4**

2 LoRa **5**
2.1 Vysielacie pásmo 5
2.2 Štandard v Európskej únii 6
2.3 LoRaWAN™ 6
2.4 Zabezpečenie komunikácie 8
2.4.1 Symetrická šifra AES-128 8
2.4.2 Prihlásenie do siete 9

3 The Things Network **12**
3.1 Limity siete 13
3.2 Pokrytie 13
3.3 Rozhranie 14
3.4 Integrácia 15

4 Lokalizácia **16**
4.1 Satelitné systémy 16
4.2 Určenie polohy pomocou siete 17
4.3 Akcelerometer 17

5 Návrh lokátora **18**
5.1 Zloženie lokátora 18
5.1.1 Mikrokontrolér MCU 18
5.1.2 LoRa modul 20
5.1.3 GPS modul 20
5.1.4 Akcelerometer 21
5.1.5 Displej 22
5.2 Proces činnosti lokátora 22
5.3 Úspora energie 24
5.4 Odhadovaná spotreba energie 26
5.5 Návrh zapojenia 28
5.6 Návrh plošného spoja 29
5.7 Návrh antény 30
5.8 Zabezpečenie 31

6 Zostrojenie lokátora **32**
6.1 Vytvorenie a osadenie plošného spoja 32

6.2 Firmware lokátora 35

7 Počítačová aplikácia **38**
7.1 Serverová aplikácia 38
7.2 Užívateľská aplikácia 41

8 Experimenty **43**

9 Záver **46**

Literatúra **48**

A Obsah pamäťového média **51**

B Obrázky **52**

Zoznam obrázkov

2.1 Model siete LoRaWAN 7
2.2 Triedy koncových zariadení LoRaWAN 8
2.3 LoRaWAN packet 9
2.4 OTAA diagram prihlásenia do siete 10
2.5 ABP diagram komunikácie 11

3.1 The Things Network infraštruktúra 12
3.2 TheThingsNetwork stanice gateway v ČR 13
3.3 Pokrytie mesta Brno podľa TTN Mapper 14
3.4 Komunikácia MQTT 15

4.1 Lokalizácia pomocou multilaterácie 17

5.1 AiThinker RHF76-052 20
5.2 Príklad schémy zapojenia NEO-M8N 21
5.3 MEMS princíp merania akcelerácie 22
5.4 Proces činnosť lokátora 23
5.5 Odosielaná správa 25
5.6 Režimy lokátora a ich prechod 27
5.7 Konfigurácia rozhraní MCU STM32L051C8T6 29
5.8 Schéma zapojenia lokátora 30
5.9 Návrh plošného spoja 31

6.1 Vytvorený plošný spoj 32
6.2 STM32L053x6/8 UFQFPN48 33
6.3 Zakúpený modul pre GPS NEO-M8N 34
6.4 Modul pre akcelerometer MPU-6050 34
6.5 Modul s displejom SSD1306 35
6.6 Osadený plošný spoj lokátora 35

7.1 Komunikácia prostredníctvom počítačovej aplikácie 38
7.2 Entity Relationship diagram 40
7.3 Webová stránka užívateľskej aplikácie 41

B.1 Schéma zapojenia lokátora 52
B.2 TTN Užívateľské rozhranie pre správu aplikácie 53
B.3 TTN Užívateľské rozhranie pre správu zariadenia 54

Kapitola 1

Úvod

Veľmi často sa objavujú pokročilé sledovače polohy v detektívnych filmoch, kde pomocou mikroskopického, nenápadného zariadenia ukrytého v oblečení, môžu hlavné postavy veľmi presne sledovať a lokalizovať páchateľa. Takéto mikroskopické zariadenia sú zatiaľ iba sci-fi, ale s rozširovaním nových technológií a nových komunikačných sietí sa postupne stávajú realitou. Najzložitejšie je vyvinúť zariadenie, ktoré je spoľahlivé, má malé rozmery a dlhú výdrž batérie. Pravidlom ostáva, že čím väčšia batéria, tým sa zväčšujú rozmery a hmotnosť výsledného zariadenia.

V tejto práci sa pokúsim navrhnúť a zostrojiť lokalizačné zariadenie s využitím energeticky nenáročnej IoT siete LoRa. Lokátory polohy majú v praxi veľké využitie, najmä pri zabezpečovaní majetku, monitorovaní pohybu zvierat, sledovaní nebezpečných ľudí a iné. V budúcnosti môžu zohrať úlohu v zdravotníctve, pri organizovaní akcií, riadení premávky, verejnej dopravy, alebo v iných múdrych systémoch pre zvýšenie bezpečnosti.

Existujúce riešenia využívajú často mobilné a telefónne siete pre odosielanie informácie o svojej polohe. Toto riešenie má značné nevýhody v tom, že nie je stavané na úspornú komunikáciu, teda výdrž batérie je nižšia. Taktiež komunikácia je väčšinou spoplatnená poskytovateľom siete. Siete BLE a WiFi sú použiteľné, ale majú príliš nízky dosah signálu. Postupne sa však rozširujú IoT siete, ktoré cielia na veľký dosah a nízku spotrebu energie. Práve preto by mohli byť vhodné pre použitie lokátora polohy.

Informácie o polohe sa pokladajú za veľmi citlivé, preto je ich potrebné dostatočne chrániť. Musia byť bezpečne uložené a dostupné iba užívateľovi, ktorý má k nim skutočne oprávnenie. Je ich možné ľahko zneužiť.

V prvej časti práce priblížim prenos dát v sieti LoRa a jej nadstavbu LoRaWAN spolu s topológiou a popisom jednotlivých častí. Taktiež zhrniem zabezpečenie a spôsob autentifikácie zariadení v sieti. V druhej časti popíšem otvorenú komunitnú platformu *The Things Network* a jej integráciu siete LoRaWAN. Následne priblížim spôsob získavania polohy a lokalizácie. V ďalšej časti uvediem návrh zariadenia a prestavím jednotlivé moduly potrebné pre činnosť lokátora spolu s ich návrhmi pre ušetrenie energie. Následne vytvorím schému zapojenia jednotlivých komponentov zariadenia a návrh plošného spoja lokátora. V šiestej kapitole (6) osadím jednotlivé komponenty podľa schémy na vytvorený plošný spoj a zostrojím výsledné zariadenie lokátora. Taktiež vytvorím program potrebný pre beh lokátora pozostávajúci z niekoľkých verejných knižníc v jazyku C. V siedmej kapitole vytvorím prehľadnú počítačovú aplikáciu pre zobrazovanie informácií o polohe zariadenia a ich správe, komunikujúcu so sieťou The Things Network. V poslednej časti vykonám niekoľko experimentov a zhrniem testovanie zariadenia v reálnom prostredí. V závere práce zhrniem zistené poznatky pri skúmaní siete a využiteľnosti vytvoreného lokátora.

Kapitola 2

LoRa

LoRa je pomerne nová bezdrôtová sieť určená pre úsporné zariadenia s veľmi nízkou spotrebou energie. Je využiteľná pre pravidelné odosielanie údajov medzi takýmito zariadeniami na veľké vzdialenosti. V zastavaných oblastiach je dosah signálu uvádzaný niekoľko jednotiek kilometrov a vo voľnom prostredí až niekoľko desiatok kilometrov [3]. Rekordný prenos[1] bol uskutočnený na vzdialenosť 832 km s výkonom iba 25 mW. Predstavuje fyzickú vrstvu pre prenos dát [9], čo umožňuje ju využiť aj pre už existujúce protokoly a siete. Na jej základe vznikla otvorená sieť **LoRaWAN** popísaná v sekcii 2.3.

Sieť je použiteľná v rôznych odvetviach priemyslu, pri automatizácii, chove zvierat, v prírode pre dlhodobé merania a získavanie dát a vo veľa iných odvetviach. Prenosová rýchlosť však je veľmi nízka, preto je vhodná skôr pre zariadenia nenáročné na prenos dát o veľkosti maximálne niekoľko desiatok bytov za dlhší časový okamžik. Je vyvinutá spoločnosťou Cycleo vo Francúzsku a odkúpená spoločnosťou Semtech. Jediným výrobcom čipov pre komunikáciu LoRa je aktuálne firma **Semtech** [27].

V tejto kapitole priblížim sieť LoRa, jej spôsob rádiovej komunikácie, limity Európskej únie a predstavím sieť LoRaWAN, ktorá tvorí nadstavbu nad fyzickou sieťou LoRa, jej topológiu a zabezpečenie.

2.1 Vysielacie pásmo

Sieť LoRa využíva pre komunikáciu technológiu **rozšíreného spektra** (Chirp Spread Spectrum Modulation), kde sa moduluje signál pomocou generovania postupností narastania („*upchirp*“) a klesania („*downchirp*“) vysielacej frekvencie. Prenášané dáta sú rozprestreté do sekvencií s nižšou frekvenciou, ako je zmena frekvencie signálu a namodulované na tento signál [17]. Pomer zmien frekvencie („*chirpov*“) ku jednému prenesenému symbolu sa označuje **Spreading Factor** (SF). Typicky sa pohybuje v rozpätí SF7–SF14 [13].

Takýto prenos má síce nižšiu priepustnosť, ale je ho možné presnejšie demodulovať aj na vyššiu vzdialenosť oproti fixnému frekvenčnému prenosu. Navyše je odolný proti úzko frekvenčnému rušeniu, Dopplerovmu efektu a interferencii. Šírka pásma, ktoré sa využíva je typicky 125 kHz, 250 kHz, alebo 500 kHz [13].

Je možné dynamicky meniť priepustnosť pomocou zmeny frekvencie modulovania, čo umožňuje regulovať dobu vysielania a dosah signálu. Táto metóda sa nazýva **ADR** (Adap-

[1] LoRa rekord vzdialenosti — https://www.thethingsnetwork.org/article/lorawan-world-record-broken-twice-in-single-experiment-1 [dostupné 26.6.2020]

tive Data Rade). Pri menej kvalitnom signáli je možné znížiť prenosovú rýchlosť, čím sa zjednoduší príjem dát, ale predĺži čas odosielania, čo taktiež zvýši energetickú záťaž [13].

Pri prenášaní dát sa môžu vyskytnúť chyby a preto sa ku dátam pridávajú kódy pre opravu chýb vzniknutých pri prenose, čo má za výsledok síce viac prenášaných dát a dlhší vysielací čas, ale podstatne spoľahlivejší prenos. Typicky sa pomer prenášaných bitov ku celkovým preneseným pohybuje $4/(5-8)$ [17].

Konfigurácia siete musí byť u všetkých účastníkoch komunikácie rovnaká pre úspešný prenos dát. V každej krajine je definované pásmo, ktoré je možné bez licencie používať pre sieť LoRa, no typicky sa používajú frekvencie pod 1 GHz.

2.2 Štandard v Európskej únii

Rádiová komunikácia je v jednotlivých štátoch prísne regulovaná a nie je možné vysielať na ľubovolných frekvenciách bez vhodnej licencie. Európska únia vydala štandard pravidiel, podľa ktorého je možné prevádzkovať komunikáciu na jednotlivých frekvenciách. Štáty následne túto medzinárodnú dohodu akceptovali pre zjednotenie pravidiel. Sieť LoRa využíva pásmo, ktoré nespadá pod potrebu vlastniť špeciálne povolenie, ale vysielanie musí spĺňať určité kritériá. Využíva pásmo 868 MHz v ktorom môže zariadenie odosielať údaje iba 1 % z celkového času. Toto pásmo je rozdelené na niekoľko kanálov, ktoré sú uvedené v tabuľke 2.1. Maximálny vysielací výkon, ktorý zariadenie môže použiť pri odosielaní je iba 25 mW. Tieto nariadenia majú za účel limitovať prílišné využitie siete, aby nedošlo k jej zahlteniu a zabezpečujú spoľahlivú dostupnosť siete.

Frekvencia	**Min. priepustnosť**	**Max. priepustnosť**
868.1	SF7 BW125	SF12 BW125
868.3	SF7 BW125	SF12 BW125 + SF7 BW250
868.5	SF7 BW125	SF12 BW125
867.1	SF7 BW125	SF12 BW125
867.3	SF7 BW125	SF12 BW125
867.5	SF7 BW125	SF12 BW125
867.7	SF7 BW125	SF12 BW125
867.9	SF7 BW125	SF12 BW125
868.8	FSK	FSK

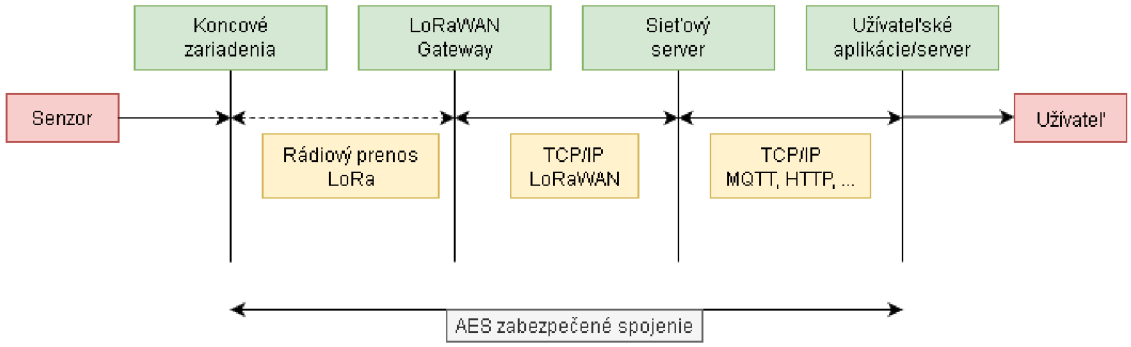
Tabuľka 2.1: Frekvencie používané v Európskej únii [8].

SF znamená povolený Spreading Factor a **BW** je maximálna šírka kanálu v kHz. Čím väčší SF, tým je nižšia prenosová rýchlosť, ale jednoduchší príjem dát [21].

2.3 LoRaWAN™

LoRaWAN™(Long-Range Wide-Area Network) definuje komunikačný protokol a model siete postavenej na bezdrôtovej technológii LoRa [9]. Je vytvorený a udržiavaný neziskovou organizáciou LoRa Alliance (zostavenou v 2015) a je plne otvorený[27]. Náklady na prevádzku sú pomerne nízke a každý si môže vytvoriť svoju vlastnú sieť. Vzniklo niekoľko platforiem poskytujúcich LoRaWAN sieť, z nich najznámejšia je *The Things Network* (kapitola 3).

Sieť bezpečne sprostredkúva komunikáciu medzi jednotlivými koncovými zariadeniami a internetovými aplikáciami pomocou špecifickej sieťovej topológie, ktorá sa skladá z niekoľkých skupín zariadení (obrázok 2.1). Koncové zariadenia komunikujú prostredníctvom siete LoRa so stanicou Gateway, ktorá dáta smeruje cez internetovú sieť na sieťový server. Z tohto serveru sú následne dostupné jednotlivým aplikačným serverom a koncovým užívateľom [23]. Topológia má centralizovanú podobu, pretože všetky dáta smerujú cez sieťový server.



Obr. 2.1: Model siete LoRaWAN

Koncové zariadenia

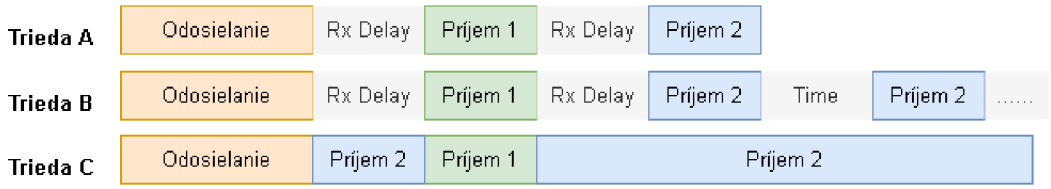
Sú nízko nákladné zariadenia, často napájané batériou, ktoré priamo komunikujú so sieťou a prenášajú potrebné informácie. Sú nakonfigurované na konkrétnu aplikáciu a konkrétny účel. Zvyčajne obsahujú senzor a odosielajú dáta v určitých časových intervaloch, po ktorých sa uspia pre maximálnu úsporu energie.

Pri aktivácii sa zariadenie autentifikuje v sieti LoRaWAN a následne komunikuje zabezpečene (sekcia 2.4). Dáta sa odosielajú v zašifrovanej podobe vo forme broadcastu všetkým dostupným základňovým staniciam. Vďaka tomu nie je nutné nové prihlasovanie pri pohybe zariadenia (handover) [14]. Podľa spôsobu príjmu dát sa rozdeľujú do troch tried (obrázok 2.2):

- **Trieda A** — základná trieda, ktorú musia spĺňať všetky zariadenia pripojené do siete. Umožňuje príjem dát iba krátko po ukončení odosielania, kedy sa otvoria práve dve za sebou idúce prijímacie okná s časovým rozostupom RxDelay. Prvé okno slúži pre príjem potvrdenia o prijatí sieťou (ACK), alebo doručenie informácii o sieti a druhé pre užívateľské dáta zariadeniu [7].
- **Trieda B** — podobná triede A, ale druhé prijímacie okno pre príjem dát sa pravidelne otvára v synchronizačných slotoch po dohodnutom intervale [7].
- **Trieda C** — príjem užívateľských dát prebieha neustále, preruší sa iba v prípade odosielania, alebo prijímania informácií zo siete [7].

LoRaWAN Gateway

Jedná sa o stacionárnu základňovú stanicu, ktorá sprostredkúva bezdrôtovú komunikáciu LoRa medzi koncovými zariadeniami a sieťovým serverom. Taktiež meria čas letu, silu



Obr. 2.2: Triedy koncových zariadení LoRaWAN

signálu a iné informácie, ktoré ku dátam pribalí [16]. Stanica musí byť neustále aktívna pre príjem komunikácie na všetkých dostupných kanáloch a mať aktívne pripojenie ku internetu. Taktiež má v sebe prijímač GPS, alebo správcom nastavenú polohu.

Zariadenie odosiela dáta všetkým dostupným staniciam, teda jeden packet môže byť zachytený a preposlaný viacerými stanicami gateway. Na serveri sa uloží zoznam staníc s dostupnými informáciami(poloha stanice, kvalita signálu, čas prijatia packetu, doba doručenia, a podobne), vďaka čomu je možné neskôr tieto dáta využiť pri získavaní polohy (sekcia 4.1).

LoRaWAN Server

LoRaWAN server je internetový server, ktorý zabezpečuje overovanie konzistentnosti prijatých dát, odstraňuje duplicitné dáta, spravuje dostupné aplikácie a smeruje dáta na aplikačné servery. Spravuje taktiež konfiguráciu topológie siete. Pri prihlasovaní do siete môže poskytnúť informácie o konfigurácii a dostupných komunikačných kanáloch siete pre zariadenie [14].

Aplikačný server

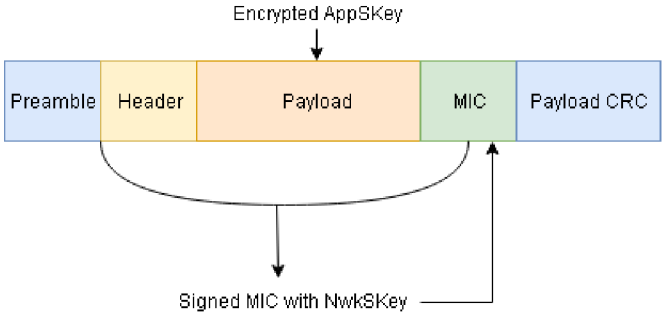
Je server, ktorý spravuje zaregistrované zariadenia v sieti a ich aplikácie. Zabezpečuje overenie zariadenia, šifrovaný prenos, nadviazanie spojenia so zariadením a distribúciu dát. Obsahuje určitú formu databáze a umožňuje základné operácie nad dátami. Dáta ďalej poskytuje pomocou rôznych rozhraní užívateľovi.

2.4 Zabezpečenie komunikácie

Komunikácia v sieti LoRaWAN je šifrovaná po celý čas od koncového zariadenia, až po aplikačný server. Sieť využíva symetrickú šifru **AES-128** (sekcia 2.4.1) a každý packet obsahuje čítač poradia. Aplikačné dáta (payload) sa šifrujú pomocou kľúča AppSKey a celý packet sa kvôli integrite podpíše MIC (Message Integrity Code) pomocou kľúča NwkSKey (obrázok 2.3) [12]. Tieto symetrické kľúče sa generujú pri každom prihlásení zariadenia do siete a sú platné iba pre danú reláciu. Ich generovanie je popísané pri prihlasovaní do siete (sekcia 2.4.2). Kľúč AppSKey pozná jedine koncové zariadenie a aplikačný server, ale kľúč NwkSKey pozná taktiež sieťový server pre overenie integrity dát.

2.4.1 Symetrická šifra AES-128

Advanced Encryption Standard (AES) je algoritmus symetrickej blokovej šifry používaný pre šifrovanie a dešifrovanie blokov dát rovnakej veľkosti. Používa substitučno-permutačnú sieť pre generovanie dátového toku, ktorý sa následne zlúči s pôvodnými dátami (CTR



Obr. 2.3: LoRaWAN packet

režim) [6]. Šifra je veľmi rýchla pre použitie v softweru, aj v hardweru. Dnes patrí medzi najpoužívanejšie šifrovacie algoritmy komunikácie. Veľkosť jedného bloku je pevne stanovená na 128 bitov. Pri kľúči o dĺžke 128 bitov sa vykoná 10 iterácií šifry, čo sa pokladá v dobe písania práce za dostatočne bezpečné [6]. Pre šifrovanie a dešifrovanie sa použije rovnaký kľúč. Šifrovanie bloku prebieha v štyroch základných krokoch:

1. **SubBytes** — jednoduchá substitúcia bytov podľa vloženého kľúča a preddefinovaných tabuliek (Rijndael-S-Box).
2. **ShiftRows** — jednotlivé byty sa poprehadzujú podľa daného poradia
3. **MixColumns** — dochádza ku prehádzaniu stĺpcov a každý stĺpec je vynásobený polynómom c(x)
4. **AddRoundKey** — kombinácia každého bytu so subkľúčom odvodeného z pôvodného kľúču pomocou danej tabuľky

2.4.2 Prihlásenie do siete

Pri pripojení zariadenia do siete sa musí overiť jeho totožnosť a dostupnosť aplikácie Aplikačným serverom. Zariadenie musí byť dopredu nakonfigurované pre konkrétnu LoRaWAN aplikáciu a musí sa pri prihlasovaní do siete identifikovať. Každá aplikácia má svoj jednoznačný 64-bit identifikátor **AppEUI** a každé zariadenie má 64-bit identifikátor **DevEUI**. Pripojenie do siete sa zabezpečuje jedným z troch spôsobov: **OTAA**, **ABP** a **nechránené**.

OTAA — Over the Air Activation

Pri registrácii zariadenia do aplikácie sa vygeneruje registrátorom 128-bit kľúč **AppKey**, ktorý sa uloží do zariadenia spolu s AppEUI a DevEUI. Pri prihlasovaní do siete sa na server odošle žiadosť o pripojenie, ktorá je podpísaná MIC (Message Integrity Code) pomocou AppKey kvôli znemožneniu modifikácie správy a správa obsahuje:

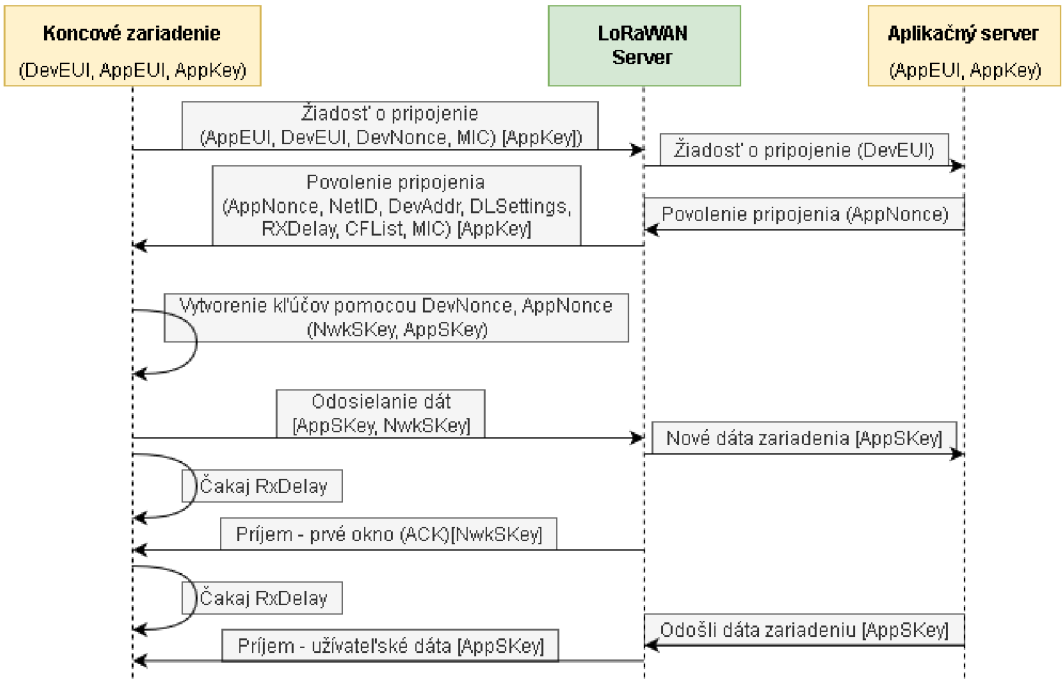
- AppEUI, DevEUI — identifikátor aplikácie a identifikátor zariadenia
- **DevNonce** — náhodne vygenerované číslo s dĺžkou 16 bitov

Server môže odpovedať povolením prístupu správou obsahujúcou:

- **AppNonce** — náhodne vygenerované číslo s dĺžkou 24 bitov

- **NetID** — ID LoRaWAN siete s dĺžkou 24bit (TTN má ID 0x000013)
- **DevAddr** — identifikátor zariadenia na fyzickej vrstve LoRa s dĺžkou 32 bitov
- **RXDelay** — čas medzi otvorením okien pre príjem dát (8bit)
- **CFList** — zoznam dostupných kanálov siete , ktoré sa budú používať pre komunikáciu

Táto správa je šifrovaná pomocou AppKey a taktiež podpísaná MIC [14]. Následne je možné v zariadení a rovnako aj v aplikačnom serveri pomocou náhodne vygenerovaných DevNonce a AppNonce vytvoriť symetrické šifrovacie kľúče **AppSKey** (pre šifrovanie obsahu správy) a **NwkSKey** (pre podpísanie správy) využívané pre zabezpečenie následnej komunikácie (obrázok 2.4).



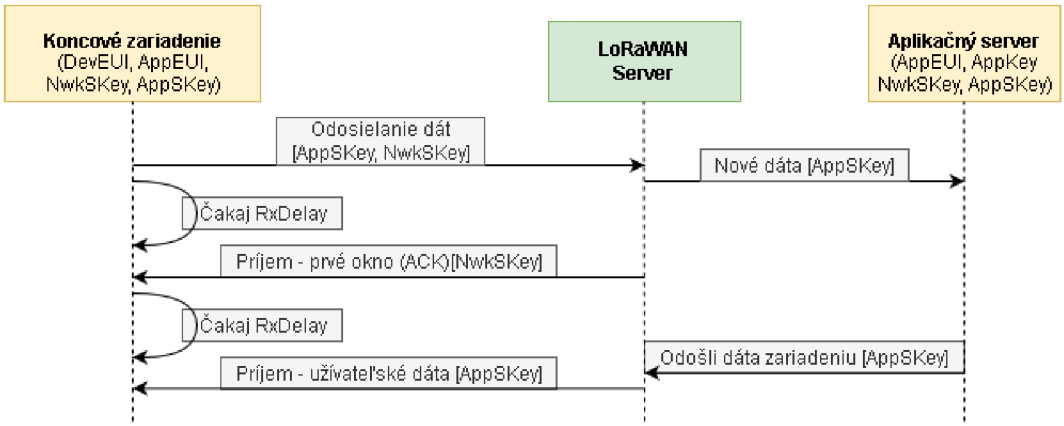
Obr. 2.4: OTAA diagram prihlásenia do siete

ABP — Authentication By Personalisation

ABP overenie je podobné spôsobu autentifikácii OTAA, no kľúče **AppSKey** a **NwkSKey** sa vygenerujú iba raz a to pri registrácii zariadenia. Sú celý čas uložené v zariadení a nemenia sa. Táto metóda sa nepokladá za bezpečnú, pretože sa zakaždým využívajú rovnaké šifrovacie kľúče. Používa sa najmä kvôli jednoduchej implementácii a pokiaľ nie je dostupný bezpečný pseudonáhodný generátor v zariadení [12] (obrázok 2.5).

Nechránené

Pri nechránenej komunikácii sa používajú verejne definované kódy. Zariadenie nie je potrebné registrovať do siete, alebo aplikácie. Prihlasovanie je podobné ABP, ale kódy sú verejne známe. Často sa komunikácia ani nešifruje. Štandardný kľúč AppSKey a NwkSKey zvyčajne býva: 2B7E151628AED2A6ABF7158809CF4F3C [16]. Môže sa použiť pri prostriedkoch, ktoré sú verejne známe.



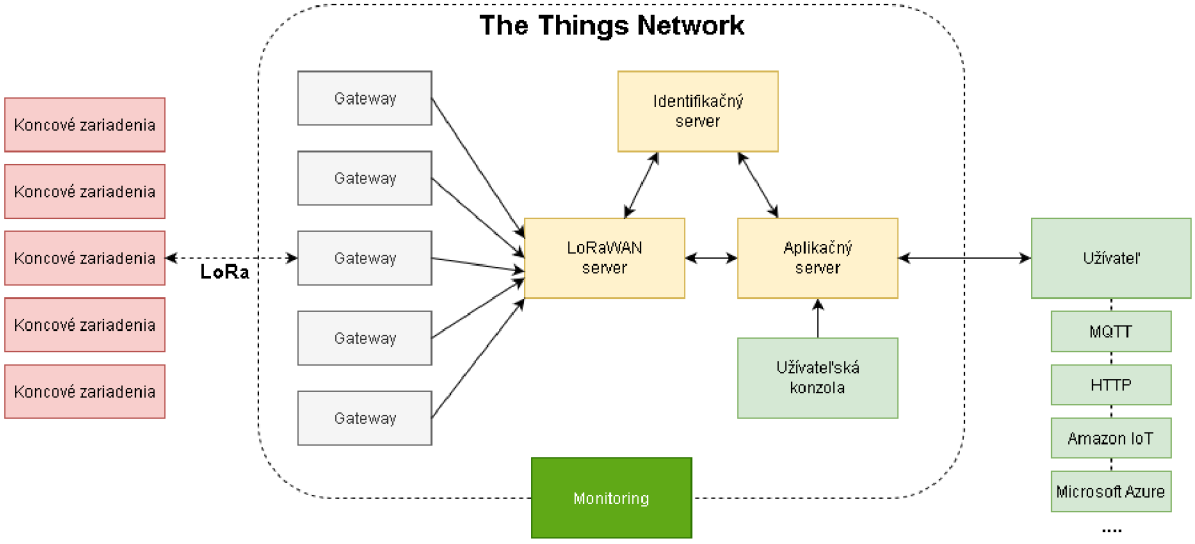
Obr. 2.5: ABP diagram komunikácie

Kapitola 3

The Things Network

The Things Network(TTN) je otvorená, decentralizovaná, komunitná platfoma pre sieť LoRaWAN, ktorá vznikla kvôli vybudovaniu cenovo dostupných IoT riešení pre bežných užívateľov. Jej komunitná verzia je zdarma a na jej budovaní sa môže účastniť každý[28]. Užívateľ si môže jednoducho vytvoriť aplikáciu, zaregistrovať zariadenie a využívať sieť LoRaWAN bez poplatkov. Taktiež je možné jednoducho pripojiť vlastnú stanicu gateway do infraštruktúry, ktorá môže byť nastavená ako súkromná, alebo verejná.

Poskytuje voľne dostupné LoRaWAN gateway, sieťový server, aplikačný server, správu a monitoring koncových zariadení a integráciu s ďalšími IoT platformami, sprístupnené užívateľom cez internet pomocou rôznych protokolov a doplnkov od tretích strán (obrázok 3.1). V dobe písania práce sieť disponuje 11971 stanicami gateway v 150 krajinách [24].



Obr. 3.1: The Things Network infraštruktúra

Platforma bola spustená 21.9.2015 v Amsterdame, kde stačilo 10 vysielačov pre pokrytie celého mesta. Účelom bolo pôvodne upozorniť na vnikajúcu vodu vlastníkov lodí v prístave [28]. Vďaka nízkym nákladom na jej budovanie a otvorenosti sa sieť veľmi rýchlo celosvetovo rozrástla a má široké využitie.

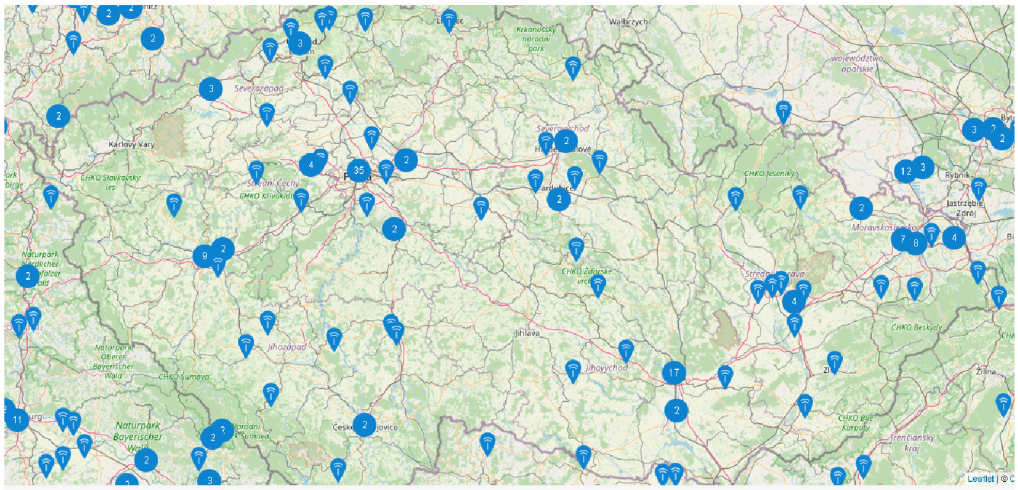
3.1 Limity siete

Sieť LoRaWAN je primárne vytvorená pre nízku spotrebu energie, veľký dosah a pomerne nízku nákupnú cenu. Umožňuje odosielať len malé množstvo dát a nie je vhodná pre prenos informácií v reálnom čase, zvuku, obrázkov, ani videa [22]. Neodporúča sa prenášať JSON, alebo plain text. S narastajúcim časom vysielania rastie spotreba energie a obsadenie siete. Platforma v dobe písania práce podporuje iba zariadenia triedy **A** a **B**. Nepodporuje nezabezpečenú komunikáciu a odporúča overovanie **OTAA** [22]. Veľkosť odosielaných dát je obmedzená na 51 bytov. Jej rozšírenie nie je príliš veľké a pokrýva iba niektoré väčšie mestá.

Sieť musí spĺňať kritériá pre vysielanie uzákonené daným štátom. V Európskej únii sú zjednotené kritériá opísané v kapitole 2.2. V rámci týchto pravidiel, sieť nemôže odosielať dáta príliš často. Maximálny čas vysielania v jednom cykle je stanovený na 1%. To znamená, že pokiaľ vysielanie trvá dobu 1s, je ho možné znovu opakovať až po 100s. Platforma TheThingsNetwork navyše stanovila maximálny čas pre odosielanie dát len 30 sekúnd za 24 hodín a pre prijímanie dát len 10 správ za 24 hodín na jedno zariadenie. Tieto obmedzenia sa vzťahujú iba na verejné, komunitné siete [20].

3.2 Pokrytie

Platforma sa teší stále narastajúcej popularite medzi užívateľmi a pokladá sa za najväčšiu LoRaWAN sieť. Pokrytie ale nie je príliš veľké, no postupne narastá. Dostupnosť siete je iba vo väčších mestách a na strategických miestach. Je preto veľmi problematické využívať túto sieť mimo určitých oblastí, no náklady na vlastnú stanicu gateway sa stále znižujú a preto sa predpokladá, že o nejaký čas môže mať každý užívateľ vlastnú stanicu.

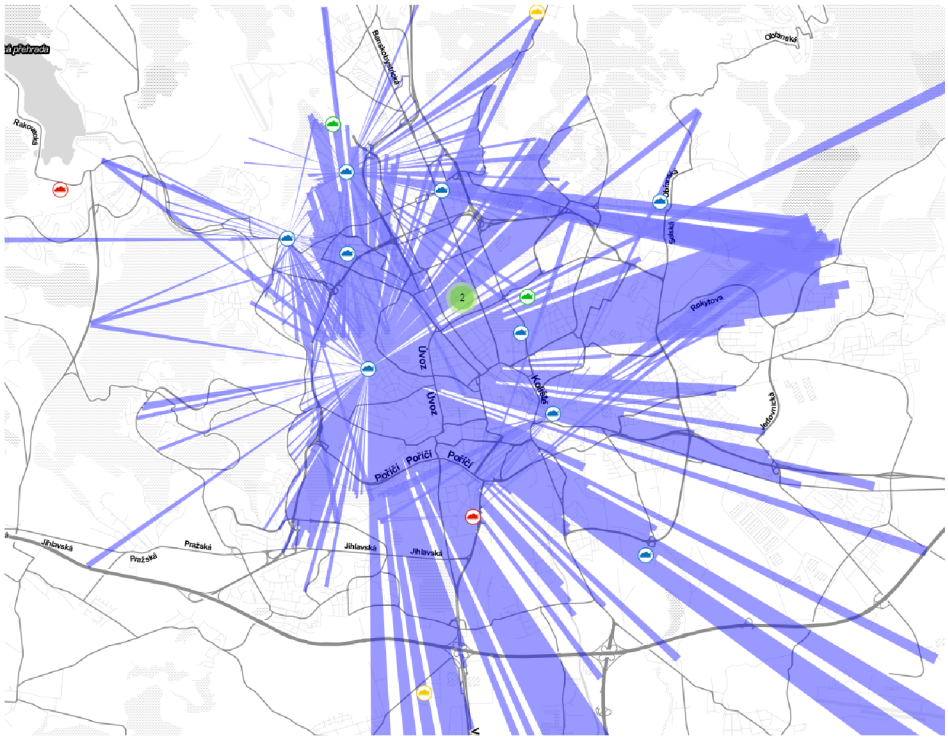


Obr. 3.2: TheThingsNetwork stanice gateway v ČR

Pre zmapovanie približného pokrytia sieťou vznikla služba TTN Mapper[1], ktorá umožňuje užívateľom merať reálny dosah signálu pomocou jednoduchého prijímača a mobilnej aplikácie. Mobilná aplikácia získava informácie o signáli z LoRaWAN koncového zariadenia, pridá k nim aktuálne GPS súradnice a odošle údaje na server TTNMapper. Pomocou webu

[1]TTN Mapper — https://ttnmapper.org/ [dostupné 10.7.2020]

si každý môže zistiť pokrytie sieťou. Nevýhodou je, že tieto merania vytvárajú užívatelia a nie sú príliš presné.



Obr. 3.3: Pokrytie mesta Brno podľa TTN Mapper

3.3 Rozhranie

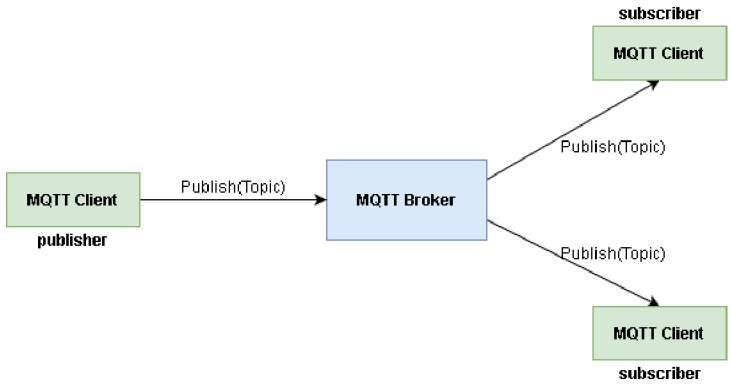
Platforma poskytuje prehľadné webové rozhranie pre správu aplikácií a ich zariadení (príloha obrázok B.2). Pomocou tohto rozhrania je možné nastaviť prístup jednotlivých koncových zariadení a sledovať ich aktivitu. Je taktiež možné naplánovať odosielanie údajov zariadeniu, prípadne simulovať dáta zo zariadenia (príloha obrázok B.3).

Každá aplikácia má svoj vlastný prístupový kľúč **AccessKey**, ktorý sa vygeneruje pri vytvorení aplikácie. Pomocou neho je možné z nej získavať údaje otvorenými rozhraniami. Medzi najpoužívanejšie rozhrania platformy patrí protokol MQTT a HTTP s rozhraním RestAPI, ktoré umožňuje získať dáta za posledných 7 dní.

MQTT rozhranie

MQTT (Message Queuing Telemetry Transport) je nenáročný protokol pre odosielanie správ medzi zariadeniami. Je vyvinutý pre komunikácie s nízkou záťažou na priepustnosť a výkon. Zvyčajne funguje nad TCP/IP vrstvou. Je ideálny pre IoT siete a zariadenia.

Využíva model **publish-subscribe**, kde zariadenia odosielajúce dáta sa nazývajú *publisher* a zariadenia prijímajúce dáta sa nazývajú *subscriber*. Správy sú odosielané do centrálneho uzla nazývaného *Broker*, kde sú triedené podľa témy. Odberatelia sa prihlasujú na jednotlivé témy u Brokra a správy sú im preposielané. Zariadenia môžu zastávať zároveň viacero rolí pre odoberanie a odosielanie správ do viacerých tématˇ [10].



Obr. 3.4: Komunikácia MQTT

Platforma The Things Network aktívne podporuje MQTT komunikáciu, pretože protokol je často implementovaný vo veľa existujúcich IoT aplikáciách. Po vytvorení TTN aplikácie sú dostupné údaje pre prihlásenie na MQTT Broker. Komunikácia je šifrovaná symetricky šifrou pomocou kľúča AccessKey. Hlavné a v práci využívané sú tieto témy:

`<AppID>/devices/<DevID>/up` — odoslanie správy zariadeniu

`<AppID>/devices/<DevID>/down` — príjem správy od zariadenia

`<AppID>/devices/<DevID>/events/activations` — prihlásenie zariadenia do siete

3.4 Integrácia

Platforma spolupracuje s veľkými IoT službami, ktoré pomáhajú vývojárom pri integrácii ich aplikácií do vlastnej siete. Medzi známe služby s priamou integráciou do TTN patria: Amazon AWS IoT, AllThingsTalk, myDevices Cayenne, Collos, OpenSensors, Tago, ThingSpeak, EVRYTHNG a iné[2]. Platforma poskytuje taktiež základné úložisko pre dočasné ukladanie prijatých správ a následné HTTP RestAPI rozhranie.

The Things Network podporuje veľkú komunitu vývojárov a obsahuje prehľadné fórum s veľa užitočnými informáciami[3]. Poskytujú bohatú dokumentáciu siete s návodmi pre integráciu zariadení a užívateľských rozhraní[4].

[2]The Things Network Integrations — https://www.thethingsnetwork.org/docs/applications/integrations.html [dostupné 2.7.2020]

[3]The Things Network fórum — https://www.thethingsnetwork.org/forum/ [dostupné 2.7.2020]

[4]The Things Network dokumentácia — https://www.thethingsnetwork.org/docs/ [dostupné 2.7.2020]

Kapitola 4

Lokalizácia

Lokalizácia zariadenia pozostáva zo získania jeho polohy a odoslania polohy na server. Získanie polohy môže byť zabezpečené pomocou satelitných systémov, alebo pomocou informácií získaných zo siete a okolia zariadenia. Moderné telefóny kombinujú tieto metódy pre čo najefektívnejšie zistenie polohy. To však nie je vždy úplne možné, pretože údaje zo satelitných systémov nie sú vždy dostupné a pre určenie polohy pomocou siete je potrebné získať dostatok informácií. Presnosť určenia polohy závisí na zvolenom spôsobe jej merania a dostupnosti informácií. Vnútri budov je lokalizácia veľmi problematická, kvôli obmedzenej dostupnosti signálu. Najefektívnejšia je kombinácia všetkých dostupných spôsobov.

4.1 Satelitné systémy

Satelitné systémy fungujú na princípe získavania informácie prevažne z umelých družíc obiehajúcich okolo planéty Zem. Jedná sa o jednosmernú komunikáciu. Tieto družice vysielajú signál s informáciami o svojej polohe a aktuálny čas. Zariadenie tieto signály prijíma zo všetkých dostupných družíc a snaží sa určiť aktuálnu polohu. Existuje niekoľko projektov, pomocou ktorých je možné získať polohu. Najznámejší a najstarší je americký systém GPS.

GPS

Global Positioning System (GPS) je komplexný navigačný systém Spojených štátov amerických, pozostávajúci z 31 umelých družíc obiehajúcich okolo Zeme vo výške okolo 20000 km. Pomocou neho je možné získať polohu zariadenia s presnosťou 30–500cm a taktiež presný čas [26]. Pre získanie polohy GPS je potrebné prijať informácie aspoň zo 4 družíc a dopočítať polohu pomocou doby letu signálu. Každá informácia zo satelitu GPS obsahuje čas odoslania, ktorý je na všetkých satelitoch presne synchronizovaný. Pomocou porovnania týchto časov je možné odhadnúť vzdialenosť satelitu od zariadenia. Ďalšie možnosti získania polohy pomocou GPS sú fázová, Dopplerová, alebo triangulačná (uhlová). Tieto metódy sú ale príliš náročné a vyžadujú veľmi presné merania [5].

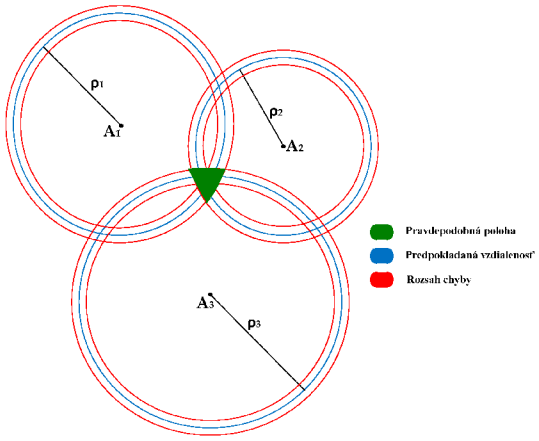
Pre príjem a spracovanie GPS existuje pomerne veľké množstvo nezávislých GPS modulov, ktoré sú pripravené na tento účel. Pre zistenie polohy používajú často kódovú metódu. Moduly využívajú rôzne spôsoby komunikácie so zariadením, najčastejšie však pomocou sériového rozhrania. GPS systémy sú pomerne náročné na spotrebu a ich využitie je nutné regulovať len na dobu, kedy je to potrebné.

Galileo

Je nový navigačný systém vyvíjaný Európskou úniou, ktorý má zdokonaliť kvalitu a rýchlosť určovania polohy. Satelity využívajú novšiu technológiu, ktorá podstatne skráti dobu čakania na lokalizáciu [2]. Tento projekt je ale stále iba v Beta testovaní, aj keď je možné ho už využívať.

4.2 Určenie polohy pomocou siete

Každá LoRaWAN stanica gateway obsahuje informáciu o svojej polohe, ktorú pribalí vždy ku packetu smerujúcemu na server. Pomocou tejto informácie a sily prijatého signálu je možné približne určiť polohu a radius pravdepodobného výskytu zariadenia. Pre presnejšie určenie pozície je potrebné signál zachytiť viacerými stanicami gateway. Pri troch a viacerých staniciach môžeme predpokladať pravdepodobný výskyt zariadenia pomerne presne. Táto metóda získavania polohy zariadenia pomocou známych bodov a vzdialeností sa nazýva Multilaterácia [4]. Využíva sa často na určovanie polohy v mobilných telefónoch, najmä pre spresnenie a zrýchlenie lokalizačných služieb. V sieti LoRaWAN sa vykonáva dodatočne na aplikačnej úrovni. Zariadenie nevie o všetkých okolitých staniciach gateway.



Obr. 4.1: Lokalizácia pomocou multilaterácie

Toto riešenie ale nie je možné vždy použiť. Sieť LoRaWAN nie je stále dostatočne rozšírená a hustota staníc gateway je pomerne nízka. Práve preto je potrebné implementovať taktiež zameriavanie pomocou satelitných systémov.

4.3 Akcelerometer

Akcelerometer slúži pre zaznamenanie pohybu zariadenia. Do určitej mieri je možné odhadnúť zmenu polohy pomocou akcelerometra a dopočítať novú polohu s využitím gyroskopu. Toto meranie však je veľmi nepresné, ale je ho možné použiť pre upozornenie na pohyb a úsporu energie. V práci využívam akcelerometer najmä pre prebudenie zariadenia a GPS prijímača pre získanie presnej polohy.

Kapitola 5

Návrh lokátora

Zariadenie lokátora vytvorené v tomto projekte slúži pre overenie funkcionality a myšlienky využitia siete LoRaWAN pre tento účel. Prvým krokom pre zostrojenie takéhoto zariadenia je jeho podrobný návrh. Navrhnuté zariadenie musí spĺňať podmienky pre vhodný GPS lokátor a teda musí mať veľmi nízku energetickú náročnosť, malé rozmery, byť odolné voči nárazom a hlavne musí byť schopné spoľahlivo získať a odoslať polohu, keď je to potrebné.

V tejto kapitole priblížim návrh zariadenia lokátora, jeho zloženie z jednotlivých modulov, ich zapojenie a možnosti úspory energie, princíp jeho činnosti a odhadovanú spotrebu energie. Na konci kapitoly zostrojím schému zapojenia obvodu lokátora a návrh osadenia plošného spoja.

5.1 Zloženie lokátora

Zariadenie lokátora navrhnuté v práci pozostáva z niekoľkých dostupných modulov potrebných pre jeho činnosť. Základom je vhodný mikrokontrolér (MCU), ktorý riadi jeho činnosť a ku ktorému sú pripojené ostatné moduly. Pre rádiovú komunikáciu je potrebný LoRa modul, pre získanie polohy pomocou satelitov GPS modul, a pre detekciu pohybu sa využíva modul Akcelerometra. Pre zobrazovanie informácií o činnosti lokátora je možné využiť displej. Ku komponentom je potrebné pridať taktiež niektoré pomocné periférie a obvody.

5.1.1 Mikrokontrolér MCU

Mikrokontrolér je integrovaný jedno-čipový mikropočítač, ktorý obsahuje procesor a všetky základné prvky potrebné pre beh počítačového programu. Používajú sa na jednoduché riadiace aplikácie a vstavané systémy, kde nezáleží príliš na vysokom výkone. Tvorí hlavnú časť lokátora, spravuje všetky moduly a rozhoduje o nadchádzajúcich akciách. Je naprogramovaný pre určitú činnosť vhodným strojovým jazykom.

Podmienky pre výber mikrokontroléru lokátora sú hlavne nízka spotreba energie a dostupnosť vhodných knižníc pre vývoj. Najvhodnejšou platformou je MCU z rodiny **STM-32L0**, pretože vynikajú veľmi nízkou spotrebou, úsporným režimom uspania a pokročilými možnosťami konfigurácie. Obsahujú dostatočný výkon pre potreby lokátora za rozumnú cenu.

Firma IBM vyvinula vhodnú knižnicu pre komunikáciu s LoRaWAN sieťou na platformu STM32, nazývanú *IBM LMIC* (LoraMAC-in-C). Je pokladaná ako základná vzorová knižnica a vychádzajú z nej aj mnohé iné knižnice pre komunikáciu LoRa na iných platformách.

Mikroprocesory STM32L0

STM32L0 je rodina 32 bitových integrovaných mikrokontrolérov od firmy *STMicroelectronics* založených na architektúre ARM Cortex-M0+ s maximálnou pracovnou frekvenciou 32 MHz [19]. Sú veľmi rozšírené, jednoduché na vývoj a dostupné pre užívateľov. Nedosahujú príliš vysoký výpočetný výkon, ale ich spotreba a kúpna cena sú veľmi nízke. Sú preto určené najmä pre zariadenia s nízkymi energetickými požiadavkami, napájané batériou. Majú taktiež označenie *Ultra-low-power platform*. Pre vývoj sú rozšírené frameworky — HAL (Hardware Abstract Layer):

- **STM32duino** — kompatibilné s veľmi rozšíreným Arduino frameworkom, ale nie je príliš vhodný pre pokročilé projekty
- **STM32Cube** — framework od tvorcov MCU STMicroelectronics, veľmi efektívny s možnosťou pokročilej konfigurácie. Je k nemu vývojárske prostredie STM32CubeIDE s nástrojmi pre uľahčenie vývoja
- **CMSIS** — framework poskytujúci jednotné rozhranie na viacerých platformách, umožňujúci jednoduchšiu prenositelnosť softwaru medzi platformami a zároveň pokročilé nastavenia zariadenia
- **MBED** — postavený na CMSIS s jednoduchším rozhraním pre rýchlejšie tvorenie programu, podporuje RTOS, USB a sieťové knižnice

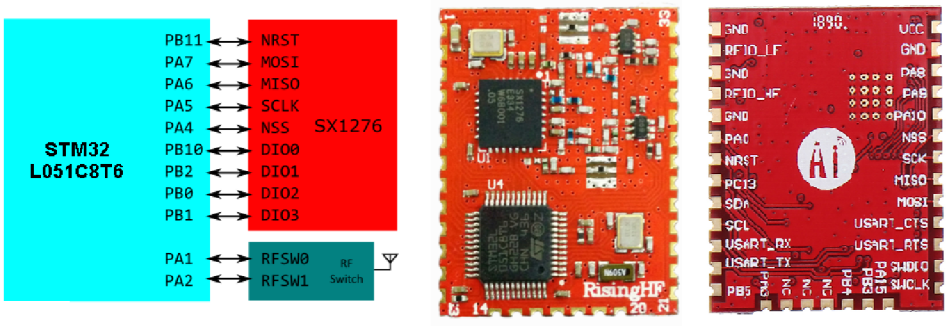
Pre zariadenie lokátora som zvolil konkrétny MCU čip **STM32L051C8T6**. Jeho hlavné výhody sú nízka spotreba energie a rýchle uvedenie do prevádzky (tabuľka 5.1). Najvhodnejší framework je STM32Cube, vďaka pokročilej správe napájania a existujúcej knižnici IBM LMIC.

Napájanie	1.65 V až 3.6 V
Teplotný rozsah	-40 až 125°C
Standby mode (2 wakeup pins)	0.27 µA
Stop mode (16 wakeup lines)	0.4 µA
Stop mode + RTC + 8-Kbyte RAM retention	0.8 µA
Run mode	88 µA/MHz
Wakeup time (from RAM)	3.5 µs
Wakeup time (from Flash memory)	5 µs

Tabuľka 5.1: Hlavné výhody STM32L051C8T6 [19]

AiThinker RHF76-052

Pri hľadaní vhodného mikrokontroléru a komunikačného modulu pre sieť LoRa som narazil na modul od firmy AiThinker s označením **RHF76-052**. Tento modul kombinuje práve zvolený MCU **STM32L051C8T6** spolu s rádiovým modulom **Semtech SX1276** a podpornými obvodmi. V lokátore tvorí hlavnú časť zariadenia, ktorá prepája všetky ostatné časti. Modul podporuje dva rozsahy frekvencií, nízko frekvenčnú 433 MHz komunikáciu a taktiež 868 MHz komunikáciu pre EU normu (sekcia 2.2). Vyvedených je množstvo kontaktov priamo z MCU a nie je problém pripojiť ďalšie moduly a zariadenia. Návrh modulu je na obrázku 5.1 a hlavné detaily v tabuľke 5.2.



Obr. 5.1: AiThinker RHF76-052

Spotreba pri uspaní	1.45 µA
Citlivosť	-140 dBm
Vysielací výkon	19 dBm
Vysielacia frekvencia	Dual band, 434/470 MHz a 868/915 MHz
Rozhranie	SPI/USART/I2C/USB
Rozmery	23 mm X 28 mm
Pamäť	ROM 64KB, RAM 8KB

Tabuľka 5.2: Ultra low power LoRa/LoRaWAN modul RHF76-052 [1]

Modul je od výrobcu naprogramovaný pre ovládanie cez sériové rozhranie pomocou **AT príkazov**. AT príkazy sú špecifickým ovládacím jazykom pozostávajúcim z krátkych textových reťazcov, ktoré môžu byť kombinované pre vykonanie operácie. Využívajú sa predovšetkým v modemoch a zariadeniach komunikujúcich so sieťou. Odosielajú sa pomocou sériového rozhrania a MCU ich dokáže dekódovať a vykonať požadované operácie [1]. Táto komunikácia by ale vyžadovala pridanie ďalšieho MCU a pre použitie v lokátore nie je vhodná. Pre optimalizáciu spotreby energie a zefektívnenie je vhodné modul preprogramovať.

5.1.2 LoRa modul

Pre komunikáciu so sieťou LoRa je potrebný vhodný rádiový modul, ktorý komunikuje na vhodných frekvenciách a umožňuje prijímať a odosielať dáta podľa požiadavkov siete. Tieto moduly vytvára jedine firma Semtech a vhodných modulov existuje viac. Ja som pre prácu zvolil modul **Semtech SX1276**. Je to rádiový vysielač a prijímač, poskytujúci veľmi veľký dosah pomocou modulácie rozšíreného spektra a vysokej citlivosti. Disponuje SPI rozhraním pre komunikáciu s mikrokontrolérom a umožňuje LoRa komunikáciu v triede A a B. Obsahuje kontakty pre externé prerušenie MCU pri prijatej správe. Nachádza sa na module RHF76-052 (sekcia 5.1.1), jeho zapojenie je znázornené na obrázku 5.1 a niektoré detaily sú zaznačené v tabuľke 5.3.

5.1.3 GPS modul

Pre získanie polohy pomocou satelitnej služby GPS je potrebný vhodný prijímač a dekodér satelitného signálu. Ako GPS modul som zvolil zariadenie od firmy U-blox, **NEO-M8N**. Tento modul dekóduje prijatú informáciu zo satelitov a pomocou výpočtov a porovnaní

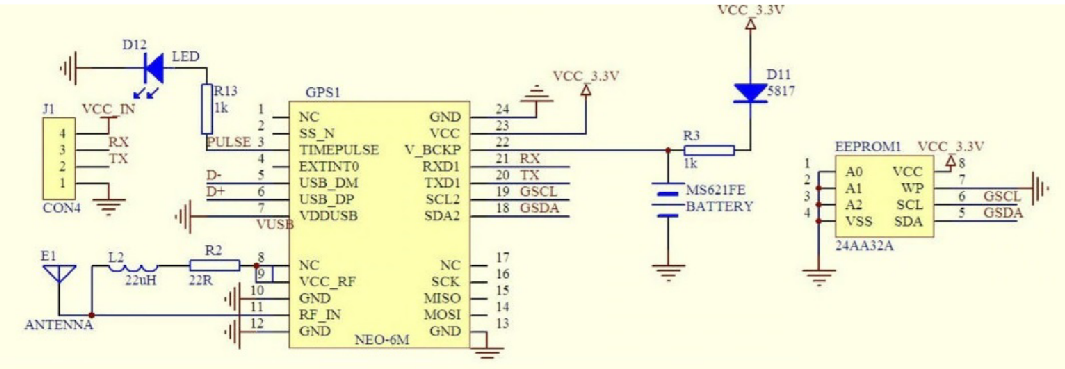
Frekvencia vysielania	137 MHz až 1020 MHz
Citlivosť	-146 dBm
Vysielací výkon	14 dBm pri 100 mW
Spotreba pri nečinnosti	1.6 µA
Spotreba pri prijímaní	9.9 mA
Spotreba pri vysielaní	120 mA

Tabuľka 5.3: Semtech SX1276 Long Range Low Power Transceiver [18]

zistí aktuálnu polohu. Je spätne kompatibilný so staršími modulmi NEO-6M a NEO-7M. Má nízku spotrebu energie, obsahuje externú anténu a umožňuje jednoduchú komunikáciu po sériovom rozhraní. Taktiež môže byť pripojený k iným zariadeniam, notebooku, alebo tabletu [25]. Modul podporuje lokalizáciu GPS a taktiež Galileo, vďaka čomu je čas pre získanie polohy nižší.

Pri štarte modul zmapuje dostupné satelity a uloží si ich informácie do pamäte EEPROM pre ďalší rýchlejší štart. Pri strate napájania modul podporuje záložnú batériu, kedy ostane aktívna časť pamäte označená ako BBR (Battery Backed RAM) a generátor hodinového času RTC. Vďaka tomuto je ďalšie skoré zapnutie veľmi rýchle a nie je nutné skenovať satelity. Modul je možné teda dočasne deaktivovať odpojením napájania a následne znovu rýchlo aktivovať. Príklad zapojenia modulu je na obrázku 5.2. Viac informácií je uvedených v tabuľke 5.5.

Komunikácia s modulom prebieha pomocou sériového rozhrania výmenou textových NMEA príkazov a informácií. NMEA (National Marine Electronics Association) je štandard textových správ pre komunikáciu medzi námornou technikou a všeobecne tiež medzi GPS prijímačmi. Je potrebné teda implementovať parser pre získanie polohy z GPS modulu. Tento parser často pozostáva zo zásobníkového automatu, ktorý overuje validnosť prijatého textu, filtruje a zaznamenáva informácie.



Obr. 5.2: Príklad schémy zapojenia NEO-M8N

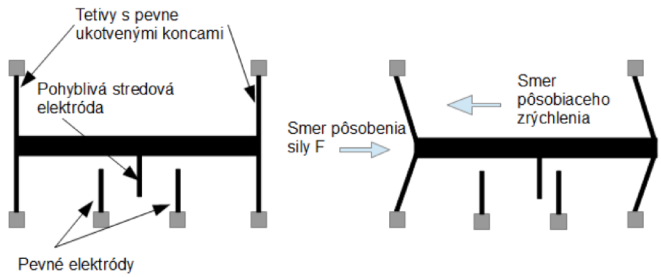
5.1.4 Akcelerometer

Akcelerometer slúži pre detekciu pohybu a zrýchlenia zariadenia. Meria akceleráciu vo všetkých smeroch. Ako vhodný akcelerometer som zvolil modul **MPU-6050**, ktorý umožňuje vyvolať prerušenie na mikrokontroléri, čím ho môže zobudiť z úsporného režimu. Pri zrýchlení, ktoré je vyššie ako nastavená hranica, sa aktivuje pin pre prerušenie. Tento pin je

pripojený na vstup mikrokontroléru, ktorý naň môže zareagovať a vykonať obsluhu prerušenia. Pomocou tohto prístupu je možné ušetriť značné množstvo energie, pretože pri nezmenenej polohe nie je nutné odosielať žiadne údaje.

Akcelerometer využíva prvky **MEMS** systémov (MicroElectroMechanical Systems). Sú to mikroskopické, elektro-mechanické systémy, ktoré snímajú vonkajšie fyzikálne javy. Princíp činnosti MEMS akcelerometru je, že sníma zmenu kapacity spôsobenú zrýchlením. Akcelerometer sa skladá zo zotrvačnej doštičky, ktorá je upevnená dvomi tetivami s ukotvenými koncami, na ktorej sú kolmo elektródy. Druhé elektródy sú pevné v rovnakom smere (obrázok 5.3). Zmenou polohy zotrvačnej doštičky sa mení geometria plôch, čím dochádza ku zmene kapacity. Ak na snímač nepôsobí žiadna vonkajšia sila, obe kapacity sú rovnaké[15].

MPU-6050 disponuje taktiež gyroskopom, pomocou ktorého je možné zistiť natočenie zariadenia. Komunikácia prebieha pomocou rozhrania I2C. Takto je možné nastaviť citlivosť a vyčítať aktuálne údaje [11].



Obr. 5.3: MEMS princíp merania akcelerácie

5.1.5 Displej

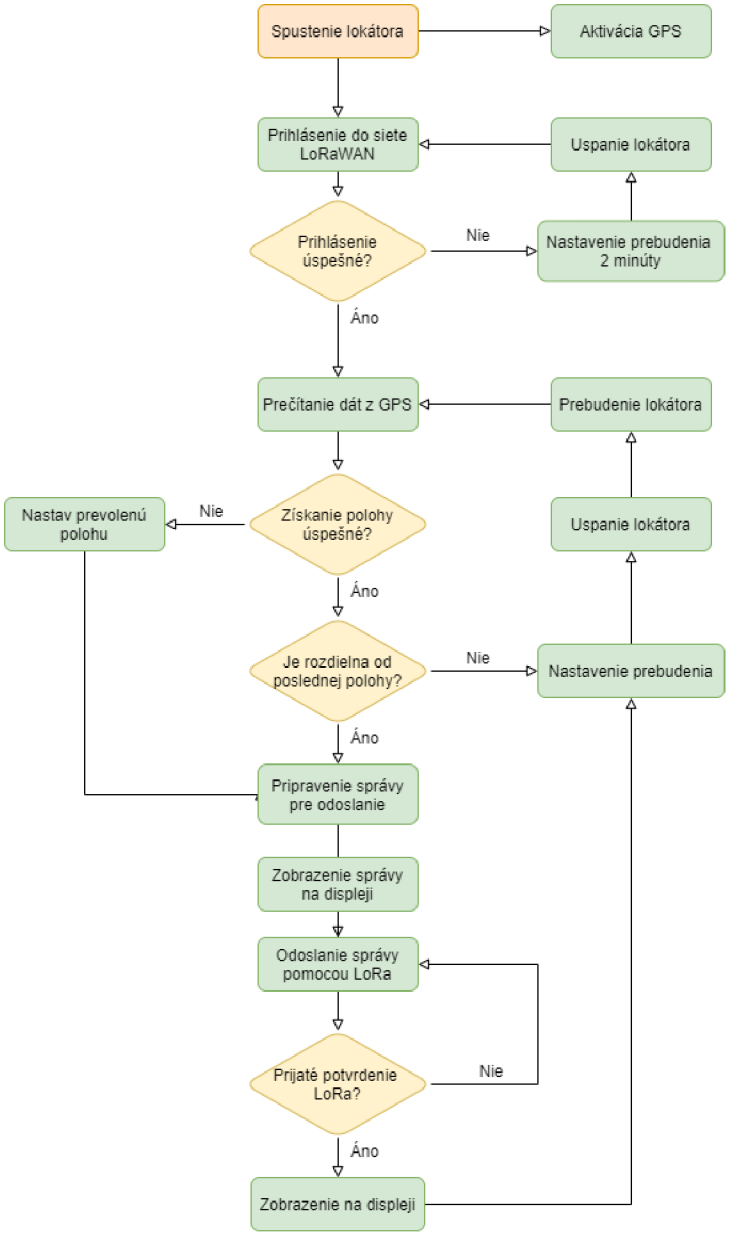
Slúži hlavne pre účely ladenia zariadenia, pre zobrazovanie užitočných informácií o sieti a aktuálnej pozícii. Pri konečnom zariadení nie je potrebný, ale je vhodný pre informovanie užívateľa o činnosti lokátoru. Ako displej som zvolil **OLED** modul **SSD1306** s rozlíšením 128x64 pixelov a uhlopriečkou 0.96". Displej disponuje I2C rozhraním a vyžaduje napájanie 2.7V až 3.6 V. Pomocou I2C rozhrania sa do pamäte displeja zapíšu dáta pre zobrazenie a následne sa vykreslia vo forme aktívnych pixelov. Displej umožňuje zobraziť iba jednu farbu.

Kvôli úspore energie sa displej vždy aktivuje iba na čas 5 sekúnd pre zobrazenie aktuálnej informácie a následne sa znovu vypne. Vhodnou alternatívou je taktiež E-ink displej, ktorý má podstatne nižšiu spotrebu a po zobrazení informácií ho nie je potrebné vypínať, pretože zobrazené dáta na ňom ostanú. Tieto displeje ale nie sú tak dostupné a pre účely monitorovania zariadenia sú zbytočné.

5.2 Proces činnosti lokátora

Po spustení lokátora sa aktivuje GPS prijímač a lokátor sa pokúsi o prihlásenie do siete LoRaWAN pomocou OTAA overenia. Pokiaľ je prihlásenie do siete neúspešné, nastaví sa prebudenie o 2 minúty pre ďalší opakovaný pokus prihlásenia a lokátor sa uspí. Po úspešnom prihlásení do siete sa prečíta aktuálna poloha z GPS modulu a pripraví sa správa. Pokiaľ sa nepodarí úspešne získať poloha, tak sa nastaví predvolená hodnota a server sa bude snažiť získať informácie zo siete. Pripravená správa sa zobrazí na displeji a odošle pomocou

LoRa do siete. Po prijatí potvrdenia o obdržaní správy sieťou LoRaWAN sa zariadenie uspí pre úsporu energie. Po prejdení určitej doby, alebo po prebudení akcelerometrom, sa cyklus opakuje. Lokátor získa znovu aktuálnu polohu pomocou GPS modulu, ktorú porovná s poslednou známou uloženou polohou. Pokiaľ sa poloha zmenila, pripraví sa nová správa, ktorá sa zobrazí na displeji a znovu odošle pomocou siete LoRa a proces sa opakuje. Pokiaľ sa poloha nezmení, nič sa neodosiela a zariadenie sa naďalej uspí. Tento proces je znázornený diagramom na obrázku 5.4.



Obr. 5.4: Proces činnosť lokátora

5.3 Úspora energie

Spotreba energie lokátora záleží hlavne na spotrebe jednotlivých modulov zariadenia. Pre šetrenie energie je potrebné pri každom module ovládať jeho činnosť a spotrebu. Navrhol som preto niekoľko postupov pri každom module, pomocou ktorých je možné ušetriť značné množstvo energie.

Úspora MCU STM32L051C8T6

Je dôležité, aby MCU bol v aktívnom režime iba po čas na to nevyhnutný, následne sa uspal a prebudil až keď je znovu potrebný. Prebudenie môže nastať buď z internej činnosti, ako je prerušenie od časovača nastaveného na určitý čas, alebo externej činnosti, ako je Akcelerometer pri pohybe zariadenia, alebo činnosťou LoRa modulu po prijatí novej správy. Pri prerušení sa vykoná obsluha daného prerušenia a je možné znovu prejsť do úsporného režimu.

Mikrokontrolér má niekoľko režimov pracovnej činnosti (tabuľka 5.1). Najmenšiu spotrebu má v **Standby mode**, kedy je ho možné znovu aktivovať jedine externým vyvolaním prerušenia. Tento režim nie je vhodný, pretože nie je aktívny generátor hodín a nie je možné načasovať ďalšie vysielanie. Vhodnejší režim je **Stop mode**, kedy ostávajú všetky periférie aktívne a spotreba je len okolo 0.8 µA.

Taktiež je potrebné zvoliť takú frekvenciu MCU, ktorá dokáže dostatočne rýchlo odbaviť potrebné prerušenie a zároveň nemá príliš vysokú spotrebu. Mikrokontrolér má maximálnu frekvenciu 32 MHz, kedy je jeho výkon a spotreba najvyššia. Ja som zvolil v návrhu frekvenciu 16 MHz, ktorá je dostatočná pre všetky časti lokátora a má podstatne nižšiu spotrebu. Predpokladaná spotreba MCU podľa údajov od výrobcu je počas činnosti 2816 µA (pri 16 MHz) a počas spánku (Stop mode) 0.8 µA, čo je pomerne veľký rozdiel.

Úspora LoRa

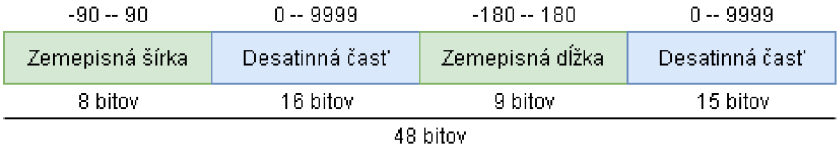
LoRa modul má taktiež niekoľko režimov činnosti. Najvyššia spotreba je pri odosielaní údajov do siete, preto je dobré odosielať správy len vtedy, keď je to skutočne potrebné a snažiť sa o nízky vysielací čas. Čas vysielania závisí od dĺžky správy a použitej modulácie[1].

Dĺžka správy je pri každom odosielaní rovnaká a je ju možné skrátiť odstránením zbytočných dát a kompresiou. Pre odosielanie polohy som preto navrhol jednoduchý protokol. Na lokalizáciu zariadenia je potrebné odoslať aktuálnu polohu, ktorá sa zapisuje vo formáte **WGS84** a pozostáva z dvoch súradníc — zemepisná šírka a zemepisná dĺžka.

Zemepisná šírka dosahuje hodnoty $-90° - 90°$. Zemepisná dĺžka dosahuje hodnoty $0° - 180°$ na západ od nultého poludníka a taktiež na východ od nultého poludníka. Pre dostatočnú presnosť sa zvyknú uvádzať 4 desatinné miesta. Pred odoslaním sa poloha zakóduje uložením hodnôt za seba, do formátu správy zobrazenej na obrázku 5.5. Takto sa docieli veľkosť odoslanej správy iba 6 bytov, čo odpovedá časom odosielania v tabuľke 5.4.

Pri odosielaní správy je možné zvoliť, či je potrebné čakať na potvrdenie o prijatí správy sieťou LoRa. Pri nedoručení potvrdenia sa opakovane odosiela správa, pokiaľ nie je potvrdená zo strany siete. Toto má za následok podstatne vyššiu spotrebu. Keďže citlivosť prijímača zariadenia nie je taká veľká, ako citlivosť brány gateway a často aj pri doruče-

[1] Kalkulačka času odosielania — https://www.thethingsnetwork.org/airtime-calculator [dostupné 10.7.2020]



Obr. 5.5: Odosielaná správa

Faktor rozloženia	**Čas odosielania**
SF7	51.5 ms
SF8	92.7 ms
SF9	185.3 ms
SF10	329.7 ms
SF11	659.5 ms
SF12	1318.9 ms

Tabuľka 5.4: Čas odosielania správy dlhej 6 bytov

nej správe sa nepodarí spätne doručiť potvrdenie do zariadenia, pre úsporu energie lokátor nemusí čakať na potvrdenie sieťou.

Úspora GPS

GPS prijímač má pomerne vysokú spotrebu, aj pokiaľ sa nachádza v úspornom móde. Preto je dobré počas stálej polohy zariadenia, modul úplne deaktivovať. Vďaka záložnej batérii je možné v prípade potreby modul dostatočne rýchlo znovu uviesť do prevádzky. V tomto prípade má zariadenie spotrebu iba okolo 8 µA pri 3.3 V. V návrhu som preto implementoval **tranzistor**, ktorý úplne deaktivuje napájanie GPS modulu, zatiaľ čo napájací pin slúžiaci pre batériu ostane aktívny počas celého behu lokátora. Spotreba prijímača je zobrazená v tabuľke 5.5.

Napájanie	2.7 V až 3.6 V
Spotreba normálna	32 mA
Spotreba úsporný mód	5.3 mA
Záložné napájanie	1.4 V až 3.6 V
Spotreba na batériu	15 µA pri 1.8 V
Studený štart	26 s
Prebudenie	1 s

Tabuľka 5.5: NEO-M8N spotreba [25]

Úspora displej

Displej slúži primárne pre účely ladenia. Nie je potrebný pri používaní a testovaní spotreby, preto je ho možné aktivovať len v prípade potreby a to podobne ako v prípade GPS modulu, pomocou **tranzistoru**. V prípade aktívneho displeja sa zobrazí informácia iba na krátky čas, po tomto čase displej zhasne a má nízku spotrebu, pokiaľ nie je potrebné zobraziť ďalšiu informáciu. Pre účely indikácie stavu je taktiež možné použiť LED diódu, ktorá má menšiu spotrebu, ale nezobrazí toľko informácií.

Spotreba displeja závisí od počtu zobrazených znakov a aktívnych pixelov. Výhoda **OLED** panelov je práve v tom, že nie je potrebné osvetliť celú plochu displeja ako v prípade LCD, ale každý pixel sa rozsvieti zvlášť. Pokiaľ nie je potrebné zobrazovať žiadne informácie, displej má minimálnu spotrebu. Výrobca udáva spotrebu 10 mA pri polovičnom rozsvietení displeja.

Úspora akcelerometer

Akcelerometer slúži pre sledovanie zmeny polohy a práve preto musí byť aktívny najmä pokiaľ sú ostatné moduly neaktívne. Jeho spotreba je pomerne nízka a pre zníženie spotreby je možné aktivovať úsporný režim, deaktivovať gyroskop a znížiť frekvenciu merania, čo zníži spotrebu, ale taktiež presnosť meraní (tabuľka 5.6). V lokátori je použitá frekvencia 5 Hz pre presnejšie a stále úsporné meranie zmeny polohy — akcelerácie.

Frekvencia merania	**Spotreba**
1.25 Hz	10 µA
5 Hz	20 µA
20 Hz	60 µA
40 Hz	110 µA

Tabuľka 5.6: Spotreba akcelerometra v závislosti na frekvencii meraní

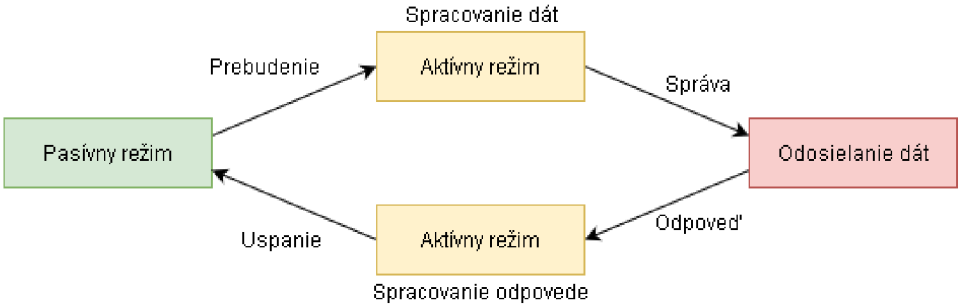
5.4 Odhadovaná spotreba energie

Je pomerne náročné podľa návrhu odhadnúť konečnú spotrebu energie, pretože často sa reálna spotreba od spotreby udávanej výrobcom líši. V rámci týchto udávaných hodnôt nie je vždy započítaná spotreba podporných elektrických súčastí a skutočná hodnota teda môže byť vyššia.

Zariadenie lokátora pracuje najmä v troch režimoch — aktívny, pasívny a odosielanie dát. V pasívnom režime zariadenie nevykonáva žiadne akcie, iba čaká na vyvolanie prerušenia, po ktorom sa prepne do aktívneho režimu. V aktívnom režime zariadenie sleduje svoju polohu, prijíma správy a plánuje odosielanie dát. Pri odosielaní dát má lokátor najvyššiu spotrebu. Tieto režimy sa striedajú podľa pohybu a potreby zariadenia (obrázok 5.6). Spotreba závisí na spotrebe jednotlivých komponent v daných režimoch. Predpokladaná spotreba je zaznačená v tabuľke 5.7. Doba prevádzky lokátora v jednotlivých režimoch nie je pevne stanovená. Závisí najmä od dostupnosti siete a času potrebného pre získanie polohy.

Spotreba	**MCU**	**LoRa**	**Akcelerometer**	**GPS**	**Displej**	**Súčet**
Aktívny režim	2.8 mA	9.9 mA	0.2 mA	32 mA	10 mA	**54.9 mA**
Pasívny režim	0.8µA	1.6µA	0.2 mA	8µA	0	**0.21 mA**
Odosielanie dát	2.8 mA	120 mA	0.2 mA	8µA	10 mA	**133 mA**

Tabuľka 5.7: Odhadovaná spotreba energie



Obr. 5.6: Režimy lokátora a ich prechod

Výdrž na batériu

Výdrž batérie záleží na jej kapacite a na energii, ktorá sa z nej odoberá. Pri reálnych batériách má veľký vplyv aj ich starnutie, samovoľné vybíjanie a okolité podmienky (teplota). Pre účely práce predpokladajme ideálnu batériu s kapacitou 1000 mAh a stálym napätím 3.8V. Takáto batéria môže poskytnúť 3.8 Wh energie ($E = U * I * t$).

Odhadovanú spotrebu môžeme určiť zo spotreby jednotlivých režimov (tabuľka 5.7) a ich časového pomeru. Môžeme predpokladať určitú periódu odosielania polohy a pomocou toho približne vypočítať priemernú spotrebu energie v danom prípade. Zariadenie sa väčšinu času nachádza v pasívnom režime. Spotreba v jednom cykle taktiež závisí od faktorov ako je doba získania polohy a kvalita signálu LoRa siete.

Uveďme prípad, kedy zariadenie vyšle svoju polohu priemerne každých 5 minút (5.4) a prípad, kedy zariadenie odosiela polohu častejšie, priemerne každú 1 minútu (5.4). V prvom prípade je možné vidieť, že výdrž batérie dosahuje skoro až 50 dní, no pri druhom prípade táto výdrž dosahuje iba 12 dní. Toto je veľký rozdiel a teda na spotrebu energie má najväčší vplyv perióda odosielania polohy. Preto prioritou úspory energie bude odosielať polohu iba pokiaľ to je potrebné.

Prípad číslo 1

Stanovme periódu pre jeden cyklus na čas 5 minút (jedno odosielanie za 300 sekúnd). Prebudenie GPS trvá 1 s a získanie dát v ideálnom prípade taktiež 1 s. Doba odosielania závisí od dostupnosti siete a zvolenej frekvencie podľa tabuľky 5.4. V ideálnom prípade SF7 iba 51,5 ms. Prijatie odpovede, jej zobrazenie na displeji a pripravenie uspania trvá približne 2 s. V jednom cykle dlhom 300 s sa teda zariadenie nachádza 295,9485 s v pasívnom režime, 4 s v aktívnom režime a iba 0.0515 s aktívne vysiela. Približnú spotrebu v jednom cykle teda môžeme určiť pomocou výpočtu 5.1.

$$
\begin{aligned}
E &= P * t \\
E &= U * I * t \\
E &= U * \frac{t_p * i_p + t_a * i_a + t_o * i_o}{t_c} * t \\
E &= 3,3\,V * \frac{295,9485\,s * 0,21\,mA + 4\,s * 54,9\,mA + 0,0515\,s * 133\,mA}{300\,s} * 1h \\
E &= 3,3\,V * \frac{62,15\,mAs + 219,6\,mAs + 6,85\,mAs}{300\,s} * 1h \\
E &= 3,3\,V * 0,962\,mA * 1h \\
E &\approx 3,1746\,mWh \approx 0,003175\,Wh
\end{aligned}
\tag{5.1}
$$

Výdrž batérie môžeme v tomto prípade určiť ako $\frac{3,8\,Wh}{0,003175\,Wh} \approx 1197\,h$, čo je vo výsledku 49 dní aj 21 hodín.

Prípad číslo 2

Stanovme periódu pre jeden cyklus na čas 1 minútu (jedno odosielanie polohy za 60 sekúnd) a podmienky dostupnosti siete a GPS rovnaké ako v prípade 1 (sekcia 5.4). Prebudenie GPS trvá 1 s a získanie dát taktiež 1 s, doba odosielania v prípade SF7 iba 51.5 ms. Prijatie odpovede, jej zobrazenie na displeji a pripravenie uspania trvá približne 2 s. V jednom cykle dlhom 60 s sa teda zariadenie nachádza 55,9485 s v pasívnom režime, 4 s v aktívnom režime a iba 0.0515 s aktívne vysiela. Približnú spotrebu v jednom cykle teda môžeme určiť pomocou výpočtu 5.2.

$$
\begin{aligned}
E &= 3,3\,V * \frac{55,9485\,s * 0,21\,mA + 4\,s * 54,9\,mA + 0,0515\,s * 133\,mA}{60\,s} * 1h \\
E &= 3,3\,V * \frac{11,75\,mAs + 219,6\,mAs + 6,85\,mAs}{60\,s} * 1h \\
E &= 3,3\,V * 3,969978\,mA * 1h \\
E &\approx 13,1\,mWh \approx 0,013\,Wh
\end{aligned}
\tag{5.2}
$$

Výdrž batérie môžeme v tomto prípade určiť ako $3,8\,Wh/0,013\,Wh \approx 290\,h$. Čo je iba 12 dní a 2 hodiny.

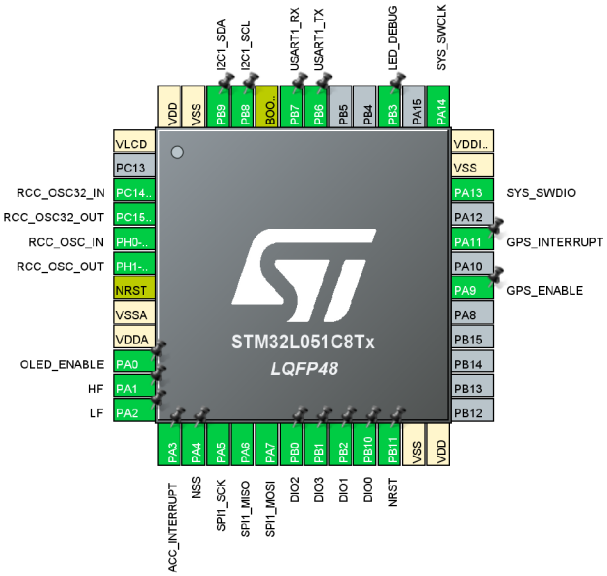
5.5 Návrh zapojenia

Hlavným jadrom lokátora je mikrokontrolér, ku ktorému sú pripojené ostatné moduly. Mikrokontrolér disponuje niekoľkými rozhraniami pre komunikáciu s modulmi, v práci sú využité nasledovné rozhrania:

- **GPIO** (General-Purpose Input/Output) — slúži pre pripojenie jednoduchých zariadení, jeho využitie je prenechané na programátora. V zariadení implementované pre aktivovanie/deaktivovanie GPS a akcelerometra. Taktiež slúži pre potreby externého prerušenia.
- **I2C** (Inter-Integrated Circuit) — synchrónna sériová zbernica, slúži pre pripojenie jednoduchých externých modulov na krátku vzdialenosť. V zariadení sa využíva pre displej a akcelerometer.

- **SPI** (Serial Peripheral Interface) — synchrónna sériová zbernica, slúži pre pripojenie zložitejších externých modulov na krátku vzdialenosť, väčšia rýchlosť prenosu. V zariadení využité pre komunikáciu s LoRa modulom znázornené na obrázku 5.1.
- **UART** (Universal asynchronous receiver-transmitter) — asynchrónna sériová zbernica, slúži pre komunikáciu textovými reťazcami. V lokátore slúži pre GPS prijímač a pre komunikáciu s počítačom počas ladenia.

Konfigurácia týchto rozhraní na MCU je znázornená obrázkom 5.7.



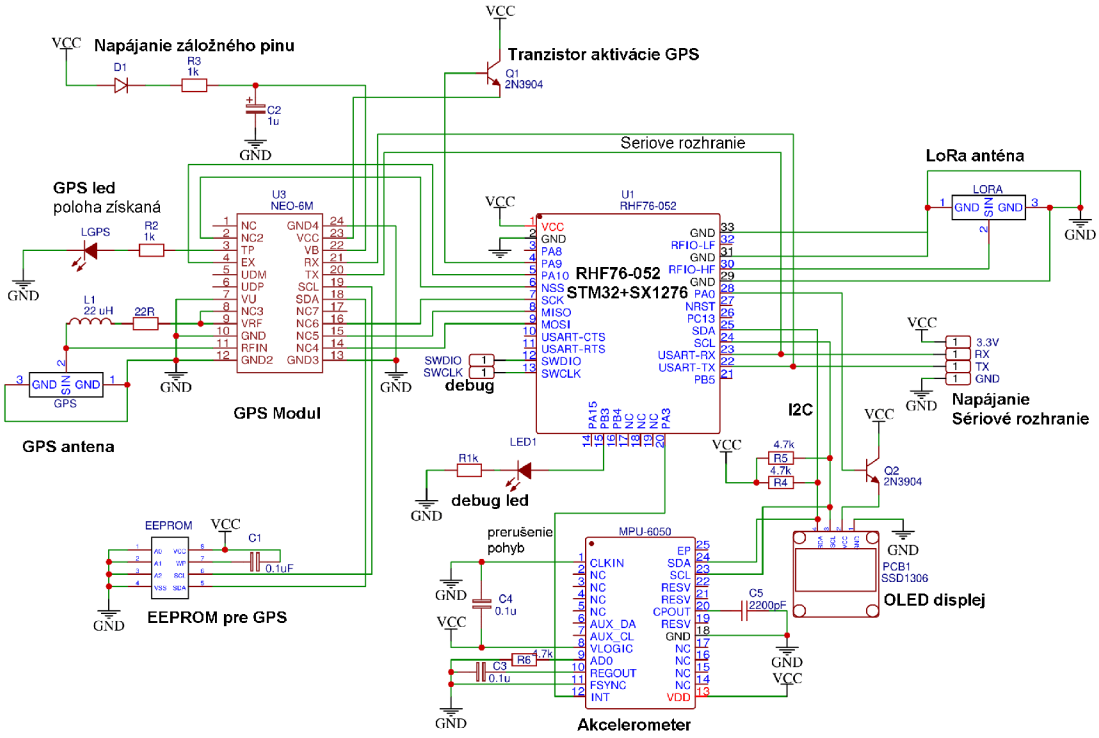
Obr. 5.7: Konfigurácia rozhraní MCU STM32L051C8T6

Všetky moduly musia byť kompatibilné. Pre návrh zapojenia lokátora som vytvoril prehľadnú schému. Kvôli ilustračným účelom je schéma v práci okomentovaná (obrázok 5.8). V návrhu som sa držal odporúčaní výrobcov a užívateľov, pre vyššiu spoľahlivosť zariadenia. Napájanie všetkých kompornent zariadenia je 3.3V. Centrom je modul RHF76-052 na ktorom sa nachádza LoRa modul SX1276 a MCU STM32L051C8T6. Je programovateľný pomocou rozhrania ST-link prostredníctvom vyvedených portov SWDIO a SWCLK. Na tento modul sú pripojené všetky ostatné moduly. Pre rozhranie I2C sú osadené PullUp rezistory pre vyššiu spoľahlivosť komunikácie.

5.6 Návrh plošného spoja

Pomocou navrhnutej schémy zapojenia (obrázok 5.8) som zostavil návrh plošného spoja (obrázok 5.9) rozmiestnením jednotlivých komponent pomocou internetového nástroja *EasyEDA*[2]. Návrh plošného spoja sa skladá z viacerých vrstiev, hlavné vrstvy sa označujú ako horná vrstva a spodná vrstva (TopLayer a BottomLayer). Obsahujú kontaktné spoje pre súčiastky a komponenty zariadenia spolu s ich prepojením (cestičkami). Vrstvy sú vo

[2]EasyEDA editor — https://easyeda.com/editor [dostupné 10.7.2020]



Obr. 5.8: Schéma zapojenia lokátora

výslednom obvode oddelené a môžu byť prepojené pomocou prevŕtaných pokovených dierok (Via). Ďalšie vrstvy môžu obsahovať popis jednotlivých častí spoja pre zjednodušenie osadzovania a zvýšenie prehľadnosti.

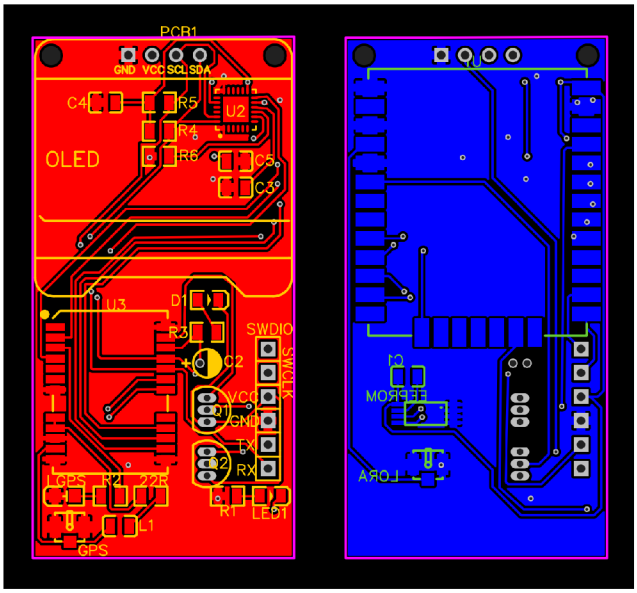
Po vhodnom rozložení všetkých komponent a ich prepojení nekrižujúcimi sa spojeniami vznikol návrh, ktorý som exportoval do archívu so štruktúrou zvanou Gerber file. Ten obsahuje jednotlivé vrstvy plošného spoja uložené ako 2D vektorové obrázky a je možné s nimi neskôr pracovať (obrázok 5.9). Po kontrole, revízii a overení návrhu som zadal objednávku spoločnosti *JLCPCB*[3] pre vytvorenie plošného spoja. Výsledný plošný spoj má rozmery 55 m x 27 mm x 2 mm.

5.7 Návrh antény

Správna anténa je dôležitá pre čo najvyšší dosah a efektivitu lokátora. Dĺžka antény priamo závisí od vlnovej dĺžky vysielacieho signály. Výpočet vlnovej dĺžky pre frekvenciu 868 MHz je uvedený v rovnici 5.3, kde v je rýchlosť vlnenia (v našom prípade rýchlosť svetla), f je frekvencia vysielania a λ je výsledná vlnová dĺžka.

$$\begin{aligned} \lambda &= \frac{v}{f} \\ \lambda &= \frac{2,998 * 10^8\, ms^{-1}}{868 * 10^6 s^{-1}} \\ \lambda &\approx 34.54cm \end{aligned} \tag{5.3}$$

[3] JLCPCB — https://jlcpcb.com/ [dostupné 10.7.2020]



Obr. 5.9: Návrh plošného spoja

Pre účely LoRa siete je často využívaná vše-smerová anténa typu polvlnný dipól. Skladaním tejto antény sa dá docieliť smerovosť a vyššia efektivita, ale zároveň väčšie rozmery. Ja som sa rozhodol využiť anténu dĺžky štvrť vlny kvôli menším rozmerom. Táto anténa má teda dĺžku iba 8,64 cm a mala by byť dostatočná pre účely lokátora.

5.8 Zabezpečenie

GPS údaje o polohe sa pokladajú za veľmi citlivé informácie, ktoré je možné zneužiť. Sieť LoRaWAN poskytuje šifrovanie len po úroveň aplikačného serveru, kde sa dáta dešifrujú a pomocou vhodných rozhraní sa poskytujú ďalej.

Pre zvýšenie bezpečnosti lokátora som sa rozhodol odosielané dáta dodatočne šifrovať pomocou symetrickej šifry **AES-128** (sekcia 2.4.1). Toto šifrovanie zvýši bezpečnosť odosielaných dát a znemožní čítanie dát v sieti TheThingsNetwork. Je ale potrebné do zariadenia bezpečne dodať náhodný šifrovací kľúč, ktorý je rovnaký ako na strane príjemcu dát. Pri použití počítačovej aplikácie (kapitola 7) sa vygeneruje kľúč, ktorý užívateľ zadá do zdrojového kódu lokátora.

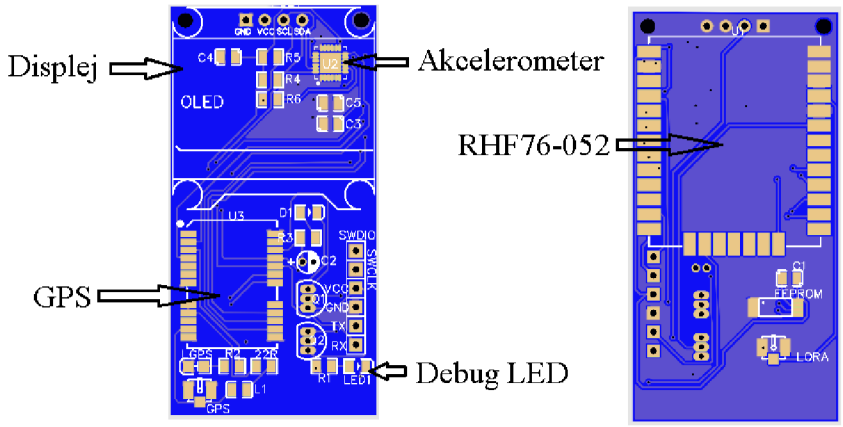
Kapitola 6

Zostrojenie lokátora

V tejto kapitole uvediem postup pre zostrojenie zariadenia lokátora podľa vytvoreného návrhu. Najskôr je potrebné získať všetky potrebné komponenty, otestovať ich funkčnosť a kompatibilitu, osadiť ich na vytvorený plošný spoj a vytvoriť vhodný program (firmware) pre lokátor bezpečne odosielajúci polohu pomocou siete LoRaWAN. Nakoniec som navrhol a vytlačil vhodný 3D model púzdra, ktorý chráni jednotlivé komponenty pri testovaní.

6.1 Vytvorenie a osadenie plošného spoja

Pre pomoc s vytvorením a osadením plošného spoja som kontaktoval Fakultu Elektrotechniky a Komunikačných Technológií, Ústav rádioelektroniky. Bohužiaľ kvôli kríze vírusu Covid-19 táto spolupráca nebola uskutočnená. Po vytvorení objednávky plošného spoja podľa návrhu (sekcia 5.6) spoločnosťou *JLCPCB*, trvalo okolo 25 dní kým bol spoj vytvorený a úspešne doručený. Výsledný plošný spoj je zobrazený na obrázku 6.1. Objednávka pozostávala z vytvorenia prototypu 5 kusov plošného spoja. Kvalita plošného spoja bola veľmi dobrá a po úspešnom otestovaní všetkých spojov je možné na plošný spoj postupne osadiť všetky komponenty.



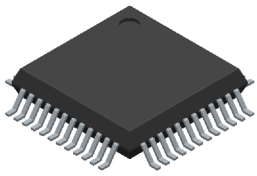
Obr. 6.1: Vytvorený plošný spoj

Všetky komponenty potrebné pre lokátor som zaobstaral dopredu z čínskeho internetového obchodného portálu *AliExpress*[1]. Na tento web má možnosť inzertovať produkty viacero čínskych podnikateľov a firiem, vďaka čomu zabezpečuje pomerne nízke ceny. Problém je s dobou doručenia, na niektoré predmety som musel čakať aj viac ako 60 dní.

Osadenie a úprava RHF76-052

Tento modul je dodávaný s výrobcom pred-programovaným firmwarom pre komunikáciu pomocou AT príkazov. Pre účely lokátora podľa návrhu (sekcia 5.1.1) je potrebné zmeniť firmware a teda naprogramovať vlastným programom. Bohužiaľ objednaný modul disponoval mikrokontrolérom s uzamknutým bootloaderom, ktorý pri nahrávaní nového programu kontroloval jeho podpis pomocou certifikačnej autority. Do takéhoto zamknutého MCU nebolo možné nahrať vlastný, modifikovaný firmware.

Po neúspešnom odskúšaní niekoľkých typov odomknutia a zmeny bootloaderu som kontaktoval výrobcu modulu, ktorý potvrdil uzamknutie a nemožnosť modifikovať MCU vlastným programom. Rozhodol som sa preto vymeniť uzamknutý čip STM32L051C8T6 novým. Nový MCU som zaobstaral pomocou českého obchodu *GM electronic, spol. s.r.o.*, ktorý disponoval iba modelom **STM32l053C8T6**[2](obrázok 6.2). Tento MCU je veľmi podobný a líši sa iba v rozhraní USB, ktoré ale nie je potrebné pre lokátor. Je ho možné pokladať pre účely lokátora za rovnaký čip ako STM32L051C8T6.



Obr. 6.2: STM32L053x6/8 UFQFPN48

Výmenu MCU som vykonal pomocou teplovzdušnej pištole a vysokej dávky trpezlivosti. Po výmene čipu bolo možné zariadenie naprogramovať pomocou rozhrania ST-link. Po otestovaní všetkých funkcií som celý modul osadil na správne miesto plošného spoja.

Osadenie GPS modul

Objednaný GPS modul NEO-M8N bol dodaný s existujúcim plošným spojom (obrázok 6.3) a vhodnou anténou. Tento plošný spoj pridával možnosť napájacieho napätia až 5V a umožňoval komunikáciu so zbernicou USB. Pre účely lokátora ale toto zapojenie nie je potrebné, preto som pomocou teplovzdušnej pištole odobral všetky potrebné komponenty a postupne ich preniesol na vytvorený plošný spoj lokátora.

Na plošnom spoji sa nachádzala aj EEPROM pamäť, ktorá slúži pre ukladanie dočasných informácií o stave GPS a aktuálnej polohe satelitov. Pomocou nej je možné zrýchliť

[1] AliExpress — https://www.aliexpress.com/ [dostupné 10.7.2020]
[2] GM Electronic — https://www.gme.cz/mikrokontroler-arm-stm32l053c8t6 [Dostupné 8.7.2020]

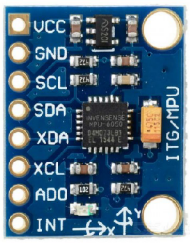
Obr. 6.3: Zakúpený modul pre GPS NEO-M8N

štart GPS prijímača, preto som aj túto pamäť zahrnul do návrhu a preniesol na plošný spoj. Po získaní aktuálnej polohy sa rozsvieti LED dioda indikujúca dostupnosť polohy.

Napájanie je regulované pomocou tranzistora NPN 2N3904, ktorý je možné aktivovať pomocou ovládateľného GPIO pinu mikrokontroléra. Ako je spomínané v návrhu (sekcia 5.3), je potrebné pripraviť aj napájanie záložnej batérie. Toto napájanie je privedené priamo z napájania lokátora a je aktívne po celý čas.

Osadenie akcelerometera

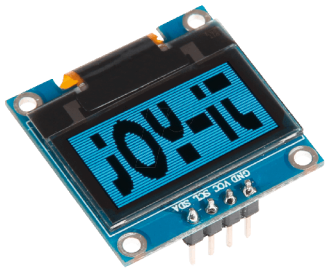
Objednaný akcelerometer MPU-6050 prišiel taktiež ako modul s vlastným plošným spojom (obrázok 6.4). Pre účely lokátora som potrebné súčasti preniesol na plošný spoj podobne, ako u GPS prijímača. Čip má osadenie typu BGA, ktoré je náročnejšie na presun a musí byť správne očistené a nasmerované. Akcelerometer sa na plošnom spoji nachádza pod displejom a preto ho nie je možné vo výslednom zariadení jednoducho vidieť. Modul má vyvedený pin prerušenia a I2C zbernicu pre komunikáciu s MCU.



Obr. 6.4: Modul pre akcelerometer MPU-6050

Displej

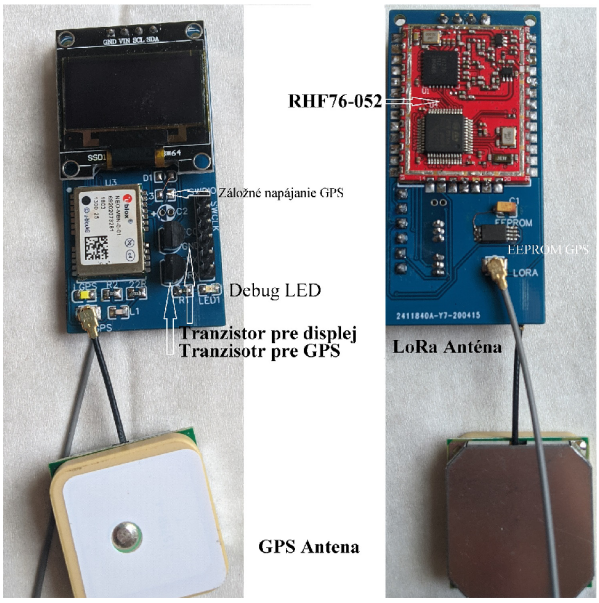
Displej je osadený na osobitnom plošnom spoji a pripojený pomocou kovových pinov, ktoré spájajú plošné spoje. Prenášajú informácie a taktiež napájanie (obrázok 6.5). Napájanie je ovládané rovnakým spôsobom ako u GPS, pomocou NPN tranzistoru. Vďaka tomu ho je možné úplne deaktivovať pomocou GPIO rozhrania MCU, pokiaľ nie je potrebný.



Obr. 6.5: Modul s displejom SSD1306

Výsledné zapojenie

Po osadení všetkých komponent má výsledný spoj podobu ako na obrázku 6.6. Nachádza sa tu však aj odlišnosť oproti návrhu, ktorá zjednodušuje zapojenie napájania pre GPS prijímač. Nie je osadená záložná batéria, namiesto nej je však po celý čas privedené napájanie.



Obr. 6.6: Osadený plošný spoj lokátora

6.2 Firmware lokátora

Po vytvorenom a úspešne otestovanom hardwari zariadenia lokátora prichádza na rad firmware. Firmware je program, ktorý slúži pre ovládanie vstavaného systému, v našom prípade lokátora. Nachádza sa v pamäti MCU a riadi celú prevádzku zariadenia. Program pre ovládanie lokátora vznikal postupne. Najskôr sa overila funkcionalita každého modulu osobitne a až po overení funkčnosti sa vytvoril výsledný program kombináciou jednotlivých funkčných častí. Výsledný program lokátora je vytvorený v programovacom jazyku **Embedded C** s využitím abstraktnej vrstvy STM32Cube. Pri vývoji bolo použité programovacie prostredie STM32CubeIDE.

Pre efektívne vyvíjanie firmwaru boli využité už existujúce, voľne dostupné knižnice pre jednotlivé moduly a ich ovládanie. Všetky knižnice použité v práci sú voľne dostupné pre

komunitné a aj komerčné využitie. Veľa z nich však bolo potrebné upraviť pre kompatibilitu s použitou platformou a zariadením lokátora.

Knižnica pre LoRa IBM-LMIC

Pre komunikáciu s LoRaWAN sieťou vznikla knižnica od firmy IBM s názvom *LMIC* (LoRa Mac-in-C)[3]. Táto knižnica komunikuje s LoRa modulom SX1276 pomocou zbernice SPI. Zabezpečuje prístup do siete LoRaWAN, šifrovanie, odosielanie a prijímanie správ siete. Implementuje celú logiku LoRaWAN a pripravuje jednotlivé rámce pre fyzickú vrstvu LoRa, ktoré sú následne odosielané modulom do siete.

Funguje na princípe systému úloh, kde sú presne naplánované jednotlivé operácie pre komunikáciu so sieťou v časových slotoch. Tieto úlohy sú zoradené vo fronte a postupne vykonávané. Výhodou tohto systému je, že je možné zariadenie kompletne uspať až do času ďalšej úlohy. Takýto prístup znižuje spotrebu zariadenia a zvyšuje efektívnosť vykonávaných úloh. Knižnica taktiež implementuje úsporu energie, spravuje jednotlivé režimy behu MCU a plánuje jeho prebudenia.

Knižnica vznikala dlhú dobu, no bohužiaľ dnes už nie je udržiavaná a nie je kompatibilná s aktuálnou špecifikáciou LoRaWAN. Narazil som na problémy s rozdielom časových slotov a udržiavaní komunikácie. Na jej základe však vzniklo veľa knižníc s modifikáciami, ktoré sú vhodné aj pre dnešné použitie. Najčastejšie je používaná modifikovaná knižnica určená pre framework Arduino[4]. Po otestovaní funkčnosti knižnice s využitím frameworku STM32Duino sa mi podarilo upraviť knižnicu pre funkčnosť taktiež na navrhnutom frameworku STM32Cube. Vyskytli sa však problémy s nastavením frekvencií a generátora hodinového času, ktorý funguje odlišne na daných frameworkoch. Nakoniec sa mi však podarilo úspešne a spoľahlivo rozchodiť komunikáciu so sieťou LoRaWAN.

Odosielané dáta sa zabalia podľa obrázku 5.5, následne zašifrujú symetrickou šifrou AES-128 (sekcia 2.4.1) a pomocou tejto knižnice úspešne odošlú do siete LoRaWAN. Pri príchode novej správy sa MCU prebudí a vykoná sa obsluha prerušenia registrovaná v knižnici.

Knižnica pre GPS

GPS modul poskytuje informácie pomocou spomínaných NMEA textových reťazcov cez Sériové rozhranie. Takáto komunikácia vyžaduje parser, ktorý validuje vstupné informácie a filtruje iba podstatné dáta. Pre prácu som implementoval vlastnú knižnicu, ktorá po zavolaní načíta zo vstupu 255 znakov a skontroluje, či sa v nich nachádza informácia o polohe. Pokiaľ je nájdená poloha, uloží ju a nastaví príznak dostupnosti. Pokiaľ nie, poloha nie je dostupná a príznak nastavený nie je. Príklad NMEA dát na vstupe:

```
$GPRMC,213342.00,A,4912.76326,N,01635.61162,E,0.203,,280720,,,A*70
$GPVTG,,T,,M,0.203,N,0.376,K,A*20
$GPGGA,213342.00,4912.76326,N,01635.61162,E,1,04,3.13,256.7,M,42.4,M,,*53
$GPGSA,A,3,05,28,30,07,,,,,,,,,6.42,3.13,5.60*0B
$GPGSV,2,1,05,05,60,260,43,07,56,062,39,09,,,35,28,27,163,31*43
$GPGSV,2,2,05,30,87,159,48*41
$GPGLL,4912.76326,N,01635.61162,E,213342.00,A,A*67
```

[3]IBM LMIC — https://github.com/mcci-catena/ibm-lmic [dostupné 10.7.2020]

[4]MCCI Arduino LoRaWAN Library — https://github.com/mcci-catena/arduino-lorawan [dostupné 10.7.2020]

Knižnica pre akcelerometer

Akcelerometer komunikuje prostredníctvom **I2C** rozhrania. Toto rozhranie využívajú najmä jednoduchšie moduly bez vlastnej logiky. Komunikácia je vždy riadená zo strany MCU a funguje na báze dostupnosti registrov a pamäti modulu. Pri získavaní dát MCU zapíše na zbernicu adresu modulu, adresu pre čítanie a následne prečíta požadovanú hodnotu.

Pre komunikáciu s akcelerometrom som využil knižnicu **SD_HAL_MPU6050**[5]. Táto knižnica obsahuje dostupné adresy hodnôt v akcelerometri a umožňuje vyčítať zrýchlenie, naklonenie a aktuálnu teplotu akcelerometra prostredníctvom I2C. Pre účely lokátora a jeho úspory energie som doplnil funkcie pre zníženie spotreby a deaktiváciu gyroskopu slúžiaceho na meranie náklonu.

Pri inicializácii prebieha konfigurácia akcelerometru, kde sa nastaví jeho citlivosť a obnovovacia frekvencia. Taktiež sa zapíšu vhodné hodnoty pre zníženie spotreby a pre vyvolanie prerušenia. Obsluhu prerušenia som sprostredkoval funkcii, ktorá aktivuje GPS modul a čaká na vyhodnotenie GPS parseru.

Knižnica pre displej

Displej komunikuje taktiež prostredníctvom I2C rozhrania, ako akcelerometer, ale jeho adresa sa líši. Vďaka tomu môže byť umiestnených viacero jednoduchých modulov na jednom rozhraní I2C. Pre displej som využil knižnicu **stm32-ssd1306**[6]. Knižnica implementuje základné grafické operácie na displeji. Obsahuje grafické fonty pre zobrazenie znakov, základné geometrické útvary a operácie nad nimi.

Všetky operácie po vykonaní uložia výsledný výstup do vyrovnávacej matice v pamäti MCU o rovnakej veľkosti, ako je rozlíšenia displeja. Po vyžiadaní aktualizácie údajov na displeji sa zapíše obsah tejto matice pomocou zbernice I2C a displej zobrazí údaje ako jednotlivé pixely. Displej je aktívny iba v prípade, že je povolený a priradený tranzistor pre jeho správu je otvorený. Toto sa môže ovládať aj za behu zariadenia a tým taktiež resetovať aktuálny stav displeja.

Výsledný firmware

Výsledný program pre lokátor spočíva v implementácii uvedených knižníc s úpravami a doplnenej logike činnosti lokátora. Jednotlivé knižnice sú pri aktivácii inicializované a počas behu volané podľa potreby. Program obsahuje taktiež rozhranie pre ladenie a zobrazovanie informácií o aktuálnom stave prostredníctvom sériového rozhrania.

Púzdro lokátora

Pre ochranu lokátora počas testovania som navrhol jednoduchý 3D model ochranného púzdra pomocou internetového nástroja **Tinkercad**[7] od firmy Autodesk. Púzdro som následne vytlačil z materiálu PLA na 3D tlačiarni Tronxy P802. Rozmery výsledného púzdra sú 58 mm x 31 mm x 21 mm. Lokátor som osadil do púzdra a zabezpečil pomocou tavnej pištole.

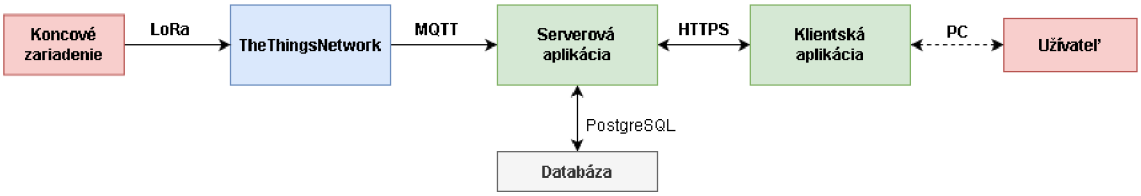
[5] SD_HAL_MPU6050 — https://github.com/sinadarvi/SD_HAL_MPU6050 [dostupné 10.7.2020]
[6] stm32-ssd1306 — https://github.com/afiskon/stm32-ssd1306 [dostupné 10.7.2020]
[7] Tinkercad — https://www.tinkercad.com/ [dostupné 10.7.2020]

Kapitola 7

Počítačová aplikácia

Pre uchovávanie a interpretáciu výsledkov pozícií z lokátora som vytvoril počítačovú aplikáciu, ktorá prijíma a zaznamenáva dáta prijaté sieťou LoRa z platformy The Things Network (kapitola 3) pomocou protokolu MQTT (sekcia 3.3). Dešifruje ich a následne uloží do databázy. Tieto dáta sú dostupné užívateľovi prostredníctvom prehľadného webového rozhrania. Aplikácia umožňuje obsluhovať viacej lokátorov zároveň.

Systém je rozdelený na dve časti. Prvá časť (server) zbiera dáta zo siete, ktoré dešifruje, uloží do databázy a poskytuje ďalej prostredníctvom protokolu HTTP. Je nevyhnutné, aby táto časť bola stále aktívna. Druhá časť tieto dáta vhodne prezentuje užívateľovi prostredníctvom spomínaného webového rozhrania, ktoré umožňuje tieto informácie zobraziť, filtrovať alebo mazať. Táto časť je aktívna iba po vyžiadaní používateľom a beží vo webovom prehliadači (obrázok 7.1).



Obr. 7.1: Komunikácia prostredníctvom počítačovej aplikácie

Ako je písané v návrhu (kapitola 5.8), údaje GPS sú dodatočne šifrované pomocou symetrickej šifry. Dáta prijaté sieťou LoRaWAN je potrebné teda aplikáciou dešifrovať. Pre šifrovanie je aplikáciou vygenerovaný náhodný symetrický kľúč, ktorý sa uloží do databáze a zapíše pri programovaní do zariadenia.

7.1 Serverová aplikácia

Serverová časť je vytvorená pre neprerušovaný beh programu na serveri, bez užívateľského rozhrania. Účelom je získavať dáta z platformy The Things Network pomocou rozhrania MQTT. Následne tieto dáta analyzovať a vyfiltrovať potrebné údaje, ktoré sa rozšifrujú a uložia do databázy PostgreSQL pomocou frameworku EntityFramework Core. Aplikácia je vytvorená v jazyku C# pre platformu .NET Core 3.1, vďaka čomu je prenositeľná na veľké množstvo systémov.

Pridanie zariadenia

Serverová aplikácia si uchováva v databáze zoznam všetkých zariadení, ktoré spravuje. Pri novom zariadení sa preto musí vytvoriť záznam s potrebnými informáciami. Najskôr sa ale vytvorí záznam na platforme The Things Network, kde sa lokátor registruje pod unikátnym DeviceID a aplikáciou s ApplicationID. Každé zariadenie patrí práve jednej aplikácii v platforme TTN, ku ktorej je možné vytvoriť kľúč AccessKey pre získavanie správ pomocou externých rozhraní (obrázok príloha B.2).

Pomocou užívateľskej aplikácie sa vyvolá akcia pridania nového zariadenia a poskytne sa DeviceID, ApplicationID a AccessKey. Serverová aplikácia si tieto údaje uloží v databáze a vygeneruje náhodný šifrovací kľúč a inicializačný vektor, ktoré sa uložia do databázy a odošlú sa do užívateľskej aplikácie. Každý lokátor má navyše príznak aktívnosti.

Získanie dát z TheThingsNetwork

Po spustení aplikácie sa načítajú z databáze údaje lokátorov, ktoré sú označené príznakom ako aktívne, spolu s ich kľúčmi pre odoberanie MQTT správ. Pre každé zariadenie sa potom vytvorí osobitný MQTT Client, ktorý naviaže spojenie na server TTN broker – `eu.thethings.network:1883` pomocou DeviceID, ApplicationID a AccessKey. Odoberá správy s témou `"{ApplicationID}/devices/{DeviceID}/up"`. Tieto správy obsahujú informácie a obsah packetov, ktoré sú odosielané z lokátora do siete. Sú prijímané vo formáte **JSON**, ktorý obsahuje veľa metadát zo siete TheThingsNetwork a taktiež **payload**, v ktorom sa nachádza odoslaná správa. Po overení validity a analýze prijatej JSON správy, sa rozšifruje obsah payload pomocou šifrovacieho kľúča a inicializačného vektoru algoritmom AES-128 (sekcia 2.4.1).

Obsah správy sa spolu s niektorými informáciami uloží do databáze. Pokiaľ zariadenie nie je označené ako aktívne, tak sa preň nevytvára žiadne spojenie MQTT. Pre MQTT Client v práci využívam vhodnú knižnicu **MQTTnet**[1], ktorá podporuje platformu .NET Core 3.1 a je dostupná pomocou NuGet balíčkov.

Databáza

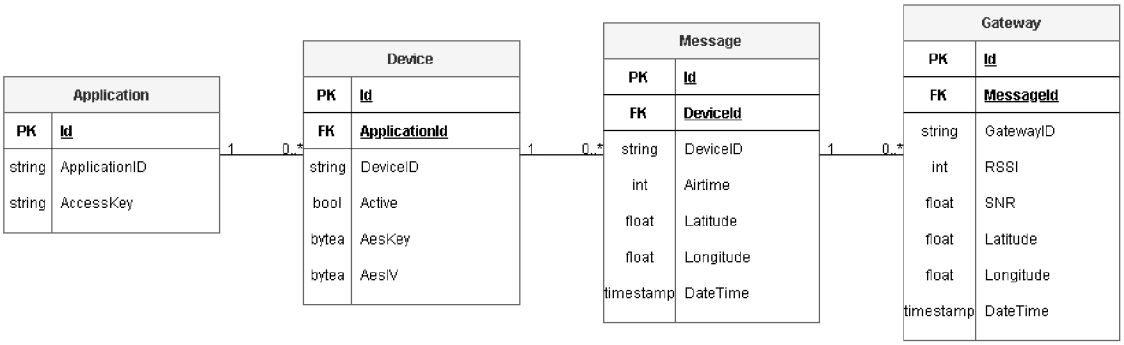
Databáza obsahuje hlavne zoznam uložených lokátorov a ich polôh. Každé zariadenie v sieti TTN je priamo viazané na TTN aplikáciu. Pomocou aplikácie je možné zariadenia spravovať a získavať ich správy. Zariadenie v databáze obsahuje šifrovacie informácie pre dešifrovanie správ a príznak aktívnosti. Každá prijatá správa o polohe obsahuje ID zariadenia, aktuálnu polohu, čas prijatia a zoznam brán, ktorými bola prijatá, spolu s ich informáciami o polohe a kvalite signálu. Pomocou tohto je možné dodatočne odhadnúť polohu. ER diagram databáze je zobrazený na obrázku 7.2.

Ako databázový server som zvolil PostgreSQL, ktorý nemá problém s operáciami ani pri veľmi veľkých databázach. V aplikácii som využil rozhranie EntityFramework Core nastavené pre pripojenie ku databázovému serveru. Následne som vytvoril štruktúru databáze a navrhol funkcie pre zápis a čítanie údajov.

Rozhranie RestAPI

Komunikácia medzi serverovou a užívateľskou aplikáciou prebieha prostredníctvom rozhrania RestAPI protokolu HTTP. Toto rozhranie je presne definované na oboch stranách a

[1] MQTTnet — https://github.com/chkr1011/MQTTnet [dostupné 20.7.2020]



Obr. 7.2: Entity Relationship diagram

slúži pre prenos údajov formou dotaz/odpoveď. Serverová aplikácia obsahuje tri kontrolére poskytujúce dáta: ApplicationsController, DevicesController a MessagesController. Každý z kontrolérov poskytuje správu nad danou časťou databázy. V práci sú využité nasledovné prístupové body aplikácie:

- **GET /api/applications** — získa všetky aplikácie so zariadeniami a ich vlastnosťami
- **POST /api/applications** — pridá novú aplikáciu a vráti jej ID s vlastnosťami
- **DELETE /api/applications/{ID}** — vymaže aplikáciu so všetkými zariadeniami
- **GET /api/devices/{ID}** — získa vlastnosti zariadenia s ID
- **POST /api/devices** — pridá nové zariadenie a vráti informácie pre šifrovanie
- **GET /api/devices/activate/{ID}** — aktivuje zariadenie pre príjem správ
- **GET /api/devices/deactivate/{ID}** — deaktivuje zariadenie pre príjem správ
- **DELETE /api/devices/{ID}** — odstráni zariadenie z databáze aj s údajmi
- **GET /api/messages** — získa všetky polohy všetkých zariadení
- **GET /api/messages/{ID}** — získa polohy zariadenia s konkrétnym ID
- **GET /api/messages/{ID}/{startDateTime}/{endDateTime}** — získa zoznam polôh zariadenia v určitom časovom intervale
- **DELETE /api/messages/{ID}** — vymaže správu s polohou a bránami

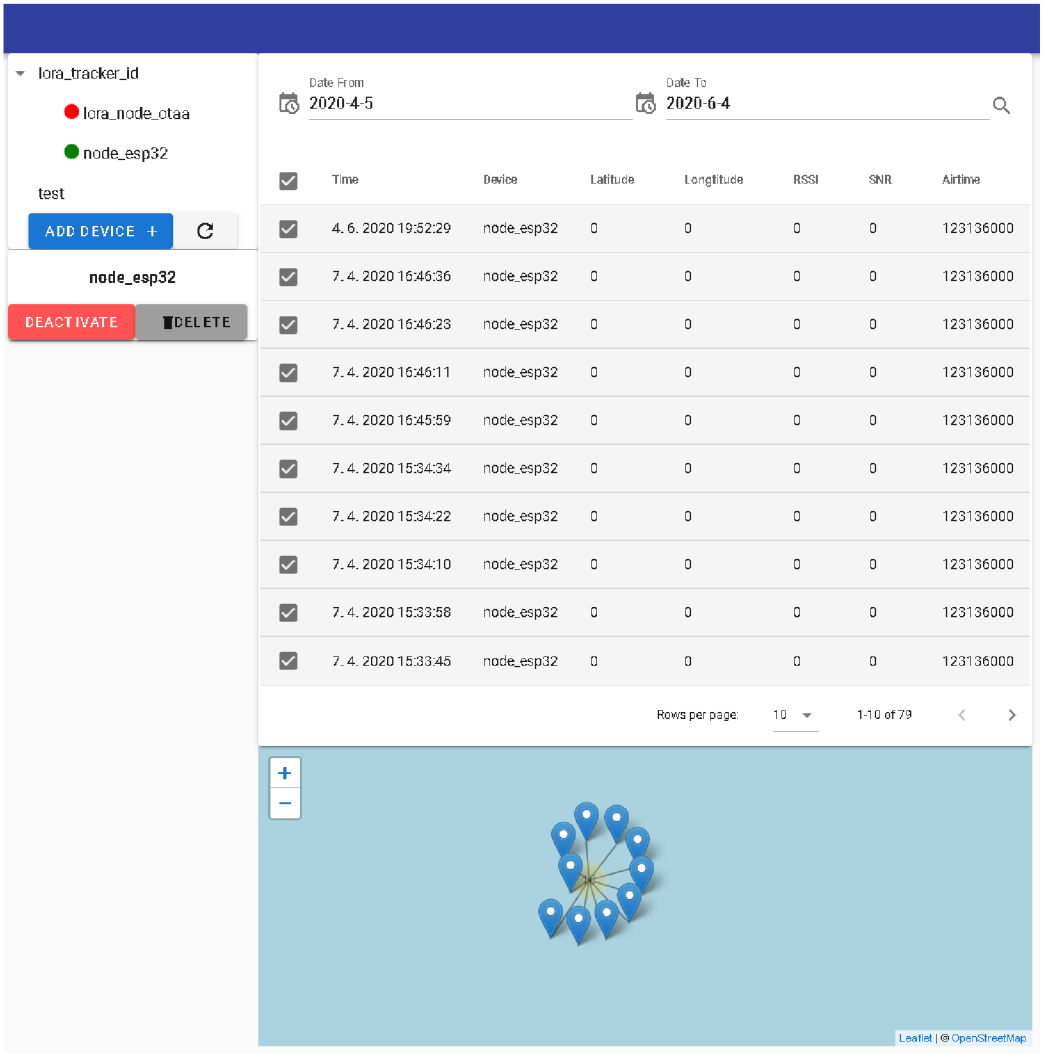
Docker

Pre lepšie nasadenie serverovej aplikácie som využil kontajnerový systém Docker. S pomocou tohto systému je možné jednoducho aplikáciu prenášať aj na iné systémy a pre iné inštancie lokátora. Pre vytvorenie kontajneru je potrebný Dockerfile, ktorý obsahuje informácie o použitom obraze a výslednom programe. Vďaka tomu nie je potrebné na server nič inštalovať, ani konfigurovať.

7.2 Užívateľská aplikácia

Klientská aplikácia je vo formáte webovej stránky a slúži pre užívateľa na zobrazovanie a správu jednotlivých lokátorov a ich zaznamenaných polôh. Je vytvorená s využitím Javascriptového frameworku **VueJS**, ktorý slúži pre vytváranie užívateľských rozhraní a jednostránkových aplikácií. Hlavná užívateľská časť aplikácie je zobrazená na obrázku 7.3.

Ľavú časť stránky tvorí zoznam všetkých aplikácií a ich zariadení. Po zvolení zariadenia sa načítajú všetky jeho zaznamenané polohy a je možné zariadenie *Deaktivovať*, *Aktivovať*, alebo *Zmazať*. Pri pokuse zmazať sa objaví potvrdzujúci dialóg. Po zvolení možnosti pridania nového zariadenia sa zobrazí formulár pre vyplnenie informácií z platformy TTN a po úspešnom pridaní sa zobrazí správa so šifrovacími údajmi pre vloženie do zariadenia. Je možné taktiež vynútiť znovu načítanie zoznamu. Hlavná časť obsahuje tabuľku a mapu.



Obr. 7.3: Webová stránka užívateľskej aplikácie

Tabuľka

Tabuľka slúži pre zobrazenie záznamov polohy zvoleného lokátora. V hlavičke tabuľky je možné zvoliť časové obdobie pre zobrazenie záznamov. Po kliknutí na dátum sa zobrazí kalendár, v ktorom je možné listovať a zvoliť vhodný dátum. Po potvrdení sa rozsah dátumov spolu s ID lokátora odošle na serverovú aplikáciu, ktorá odpovie požadovanými dátami.

Záznamy obsahujú dátum a **čas** získania danej polohy, **názov** zariadenia, **zemepisnú šírku** a **dĺžku**, **kvalitu signálu** a **dobu letu** signálu zo zariadenia lokátora. V tabuľke je možné zoradiť záznamy podľa jednotlivých údajov. Na jednej stránke tabuľky je zobrazených iba 10 záznamov kvôli prehľadnosti, no je možné zvoliť aj vyšší počet.

Mapa

Mapa slúži pre názorné zobrazenie polohy zariadenia. Po zvolení riadku v tabuľke sa tento údaj zobrazí na mape. Je možné zvoliť niekoľko riadkov súčasne a tým sledovať polohu. Mapa sa priblíži na posledný zvolený záznam a pri oddialení sa body zhlukujú. Po kliknutí na bod je zobrazený jeho dátum čas prijatia.

Pre zobrazenie mapy som zvolil Javascript knižnicu **Leaflet**. Funguje efektívne na všetkých hlavných platformách a môže byť rozšírená množstvom dodatočných doplnkov. Ako zdroj podkladov pre mapu som zvolil otvorenú platformu **OpenStreetMaps**, ktorá poskytuje celosvetové mapy zadarmo. Tieto mapy dostatočne detailne zobrazujú aktuálne informácie.

Rozhranie

Pre komunikáciu so serverovou aplikáciou slúži RestAPI rozhranie popísané v sekcii 7.1. Na jeho implementáciu som využil Javascriptovú knižnicu **Axios**, ktorá slúži práve pre prenos dát pomocou protokolu HTTP.

Pre zabezpečenie je možné použiť protokol **HTTPS**, ktorý obsahuje zabezpečenie TLS/SSL a všetka komunikácia so serverovou aplikáciou je šifrovaná. Vyžaduje však vytvorenie bezpečnostných kľúčov certifikačnou autoritou a pre účely testovania lokátora v práci to nie je potrebné.

Kapitola 8

Experimenty

V tejto kapitole otestujem zariadenie lokátora vytvorené v práci, zmeriam jeho skutočnú spotrebu energie, dosah a dostupnosť siete LoRaWAN v reálnom prostredí. Na koniec zhrniem vlastnosti a spoľahlivosť vytvoreného zariadenia.

Všetky merania v práci boli uskutočnené navrhnutým a vytvoreným zariadením lokátora v reálnych podmienkach mesta Brno (pokrytie na obrázku 3.3). Merania neboli uskutočnené na plne odladenom lokátore kvôli časovým dôvodom. Preto sa môže nameraná spotreba v experimentoch od udávanej líšiť a je ju možné ešte znížiť dôkladnejším ladením.

Pri experimentoch bola taktiež použitá vlastná stanica gateway **RAK7258** od firmy RAK Wireless. Jedná sa o vnútornú, otvorenú stanicu určenú pre LoRaWAN sieť postavenú na platforme OpenWRT so vstavaným LoRa smerovačom. Obsahuje taktiež sieťový a aplikačný server a je ju možné využívať ako samostatnú LoRaWAN sieť pre 128 zariadení. Počas testovania bola stanica integrovaná do platformy The Things Network a využívali ju aj iní používatelia a zariadenia v blízkosti.

Spotreba energie

Zariadenie bolo počas experimentoch napájané stálym napätím 3.3V pre konzistentnosť meraní spotreby. Spotreba energie je kalkulovaná ako množstvo prúdu odoberaného zariadením vynásobené vstupným napätím zariadenia. Testy a merania spotreby boli vykonané pomocou ampérmetru s presnosťou merania $\pm$ 0.01 mA. Pre presnejšie merania by bolo potrebné využiť pokročilejší ampérmeter, alebo osciloskop. Pre účely odhadu výdrže batérie v práci táto presnosť postačuje. Spotreba bola meraná najskôr pri každom module a móde osobitne, neskôr sčítaná do výslednej spotreby.

Spotreba	**MCU**	**LoRa**	**Akcelerometer**	**GPS**	**Displej**	**Súčet**
Aktívny režim	5.59 mA	10.5 mA	1.2 mA	38 mA	5.14 mA	**60.43 mA**
Pasívny režim	0.01 mA	0.00 mA	0.2 mA	0.02 mA	0.01 mA	**0.24 mA**
Odosielanie dát	0.25 mA	128 mA	0.2 mA	0.02 mA	5.14 mA	**133.6 mA**

Tabuľka 8.1: Odoberaný prúd zariadenia pri pracovných režimoch

Nameraná spotreba energie je až nečakane podobná spotrebe odhadnutej v sekcii návrhu 5.4. Spotreba v pasívnom režime je vyššia (0,24 mA oproti 0,0016 mA) pravdepodobne kvôli elektronickým súčastiam, ktoré neboli započítané v návrhu (PullUp rezistory, Diody, EEPROM). V aktívnom režime je spotreba takmer rovnaká (60,43 mA oproti 60,9 mA).

Režim	Prúd	Spotreba
Aktívny režim	**60.43 mA**	**199.419 mW**
Pasívny režim	**0.24 mA**	**0.957 mW**
Odosielanie dát	**133.6 mA**	**440.88 mW**

Tabuľka 8.2: Spotreba energie pri pracovných režimoch

Spotreba pri odosielaní dát je zrejme vyššia kvôli použitej anténe a anténnemu obvodu LoRa. Spotreba MCU sa dynamicky líši podľa aktuálnej záťaže jeho činnosti. Taktiež som implementoval viacero úsporných režimov MCU a je možné medzi nimi jednoducho prepnúť. Jeho spotreba je ale stále vyššia, ako spotreba udávaná výrobcom. Spotreba LoRa modulu je podobná ako udávaná, no v Standby režime som nebol schopný zachytiť žiaden prúd s využitým ampérmetrom. Záleží taktiež na využitej anténe. Akcelerometer má rovnakú spotrebu až na prípad jeho aktívneho čítania. Pri komunikácii I2C vykazuje zvýšenú záťaž. GPS prijímač pri odpojenej anténe mal rovnakú spotrebu ako udáva výrobca (32 mA). Pri pripojení externej antény lokátora táto hodnota vzrástla na 38 mA. Displej má nižšiu spotrebu ako v návrhu, pretože pri zobrazovaní informácií je aktívna plocha menšia ako predpokladaná v návrhu.

Výdrž batérie

Výdrž záleží na kapacite batérie a na energii z nej odoberanej. Kvôli časovým dôvodom nebola testovaná dlhodobá výdrž na reálnej batérii. Pri použití batérie je potrebné upraviť napätie na vhodnú, stálu hodnotu 3,3 V potrebnú pre beh lokátora. Ako bolo ukázané v návrhu, spotreba záleží najmä na pravidelnosti odosielania dát a to závisí na pohybe zariadenia. Pokiaľ sa zariadenie celý deň nepohne, jeho spotreba je minimálna.

Dosah siete LoRa

Dosah siete bol testovaný s využitím dipólových antén. Výsledný dosah siete LoRa ale nie je podľa predpokladaných očakávaní. Dosah najviac záleží od prekážok a vzdialenosti medzi stanicou a zariadením. Veľký rozdiel teda tvorí dosah signálu v zastavanom a vo voľnom priestranstve. V meste, v blízkosti obytných domov sa signál veľmi rýchlo tratil. Reálny dosah je iba okolo 500m. V testovaní na voľnú viditeľnosť bol signál lepší, dosahoval takmer 3km. V reálnom prostredí je ale problém dosiahnuť podmienky pre volnú viditeľnosť. Na kvalitu signálu má taktiež veľký vplyv aj výška uloženia základňovej stanice. Ako je možné vidieť na obrázku 3.3, najväčší signál má stanica uložená na vysielači niekoľko metrov nad okolitým terénom. Ostatné stanice majú značný problém s dosahom. Je možné využiť smerové antény pre komunikáciu, ktoré majú podstatne vyšší zisk signálu, ale iba jednosmernú charakteristiku a ich využitie pri lokátore je veľmi problematické.

Lokalizácia

Lokalizácia pomocou GPS pri testovaní nebola vždy dostupná. Prevažne v budovách je kvalita satelitného signálu veľmi nízka a nebolo možné zistiť polohu. Taktiež doba pre získanie polohy bola podstatne väčšia ako udávaná výrobcom. V nezastavanom priestranstve bola účinnosť podstatne lepšia, ale po neaktívnosti GPS a znovu prebudení modulu bol

problém získať polohu v dostatočnom čase. Po získaní polohy sa síce spotreba znížila, ale stále bola príliš veľká pre neustálu aktivitu (24 mA).

Lokalizácia sieťou bola len čiastočne úspešná, pretože sa počas testovania nikdy nepodarilo zachytiť správu viacerými stanicami zároveň a nebolo možné využiť metódu pre presnejšiu lokalizáciu. Stále ale bolo možné pri každej správe určiť radius približnej polohy zariadenia okolo konkrétnej stanice.

Spoľahlivosť

Spoľahlivosť lokátora závisí najmä od dostupnosti siete LoRaWAN. Bohužiaľ táto sieť stále nie je dostatočne rozšírená a jej dostupnosť a spoľahlivosť pri využití LoRa lokátora je pomerne nízka. V praxi experimenty ukázali, že nie je problém s detekciou polohy, ale hlavný problém je s jej odoslaním pomocou siete LoRa. Lokalizácia zariadenia je teda možná iba vo veľmi obmedzenom prostredí.

Počítačová aplikácia

Aplikácia sa osvedčila ako vhodný nástroj pre uchovávanie, sledovanie a zobrazenie uložených dát. Pri experimentovaní aplikácia zobrazila presnú polohu prijatú zo zariadenia pomocou navrhnutej tabuľky a mapy. Počas vykonávania experimentov bola webová aplikácia verejne dostupná na adrese `http://lora.mucka.sk`, pomocou ktorej som mohol vzdialene sledovať úspešnosť odosielania a aktuálnu polohu prostredníctvom mobilného telefónu.

Vylepšenia

Je možné aplikovať niekoľko vylepšení, ktoré neboli zahrnuté v návrhu a mohli by sa implementovať v ďalšej iterácii zariadenia. Najväčšou slabinou lokátora je slabá dostupnosť siete a nedostatočný signál. Toto by mohlo byť vyriešené lepšou anténou, ktorá by ale zaberala podstatne viac miesta a lokátor by sa stal problematickým pre prenos.

Kapitola 9

Záver

Cieľom práce bolo navrhnúť a vytvoriť vhodné zariadenie pre odosielanie aktuálnej polohy GPS s veľmi nízkou spotrebou energie pomocou využitia energeticky nízko náročnej siete LoRa a priblížiť spôsob komunikácie so sieťou LoRaWAN spolu s platformou The Things Network. Vytvorené zariadenie lokátora v práci sa ukázalo ako funkčné s určitými nedostatkami. Na základe meraní a experimentov zariadenie preukázalo, že je možné vytvoriť lokátor s veľmi dlhou výdržou batérie, ale jeho využitie je veľmi obmedzené. Bohužiaľ sieť LoRaWAN nie je príliš rozšírená a to veľmi limituje prostredie, v ktorom je možné lokátor využívať. Dosah siete LoRa nie je dostatočný pre účely komunikácie na veľké vzdialenosti, napriek jej vhodným podmienkam a propagácii výrobcami. Jej dosah a výkon obmedzujú nariadenia pre voľnú a nelicencovanú komunikáciu, ktoré sú ale nevyhnutné pre zabránenie zahltenia volného pásma.

V prvej časti práce boli zhrnuté teoretické poznatky o sieti LoRa, jej vysielacia charakteristika a obmedzenia stanovené Európskou úniou. Nasledoval model siete LoRaWAN spolu s topológiou a priblížením funkcií jednotlivých zariadení využívaných v sieti. Nechýba zabezpečenie komunikácie a overovanie zariadenia pri aktivácii. V druhej časti boli priblížené princípy fungovania otvorenej komunitnej platformy The Things Network, jej komunikácia, limity, rozšírenia a pokrytie. Myšlienka verejne prístupnej IoT siete je aktuálna a má budúcnosť, no stále nie je dostatočne rozšírená. Nasleduje priblíženie spôsobu získavania polohy pomocou rôznych systémov a informácií poskytnutých zo siete. V ďalšej časti bol zhrnutý návrh zariadenia spolu s jednotlivými komponentami potrebnými pre lokátor. Ku každej komponente boli navrhnuté opatrenia pre úsporu energie a nasledoval odhad celkovej spotreby a výdrže batérie. Prepojené komponenty vytvorili schému zapojenia zariadenia lokátora, ku ktorej bol vytvorený návrh plošného spoja. V ďalšej časti je opísané vytvorenie lokátora plošného spoja jednotlivými komponentmi a ich úpravy pre vhodné použitie. Ku zostrojeniu patrí aj vytvorenie programu pre beh lokátora, ktorý pozostáva z dostupných knižníc s miernymi úpravami. Pre účely monitorovania polohy a zaznamenávanie dát bola vytvorená počítačová aplikácia, v ktorej užívateľ môže spravovať jednotlivé zariadenia lokátora a sledovať históriu jednotlivých údajov prijatých sieťou The Things Network. Na koniec boli vykonané experimenty s vytvoreným zariadením, ktoré testovali implementáciu a vhodnosť zariadenia pre daný účel. Experimenty preukázali nízku energetickú spotrebu zariadenia a taktiež problém s nedostatočným pokrytím sieťou.

Výsledkom práce je plne funkčné zariadenie lokátora, ktoré umožňuje odosielať aktuálnu polohu pomocou siete LoRaWAN, pokiaľ je táto sieť dostupná. Zariadenie spĺňa kritériá nízko-energetického GPS lokátora s použitím LoRa siete. Bohužiaľ jeho použitie je obmedzené malým dosahom signálu a teda len na oblasti v okolí staníc LoRaWAN. V práci

taktiež vznikla aplikácia pre prehľadné zobrazenie prijatých informácií o polohe a získavanie dát z platformy TTN. Aplikácia obsahuje prívetivé užívateľské rozhranie pre správu lokátorov a ich histórii polohy.

Záverom práce je, že sieť LoRa bohužiaľ nemá dostatočný dosah a pokrytie, jej využitie je veľmi obmedzené na malú oblasť. Po vykonaní experimentov s vytvoreným zariadením bolo zistené, že sieť nie je príliš vhodná pre využitie lokátora. Podmienkou správneho lokátora musí byť najmä spoľahlivosť a dostupnosť vo väčšom prostredí. Postupným budovaním nových sietí a zvyšovaním pokrytia je možné v budúcnosti toto zariadenie využívať na väčšom území.

Literatúra

[1] Ai-Thinker. *How To Use Ai-Thinker's LoRaWAN Modem*. Máj 2017. 15 s. RHF76-052 v 0.5.

[2] Alkan, R., Karaman, H. a Sahin, M. GPS, GALILEO and GLONASS satellite navigation systems & GPS modernization. In:. Júl 2005, s. 390 – 394. DOI: 10.1109/RAST.2005.1512598. ISBN 0-7803-8977-8.

[3] Augustin, A., Yi, J., Clausen, T. H. a Townsley, W. A Study of LoRa: Long Range & Low Power Networks for the Internet of Things. *Sensors*. Október 2016, zv. 16, s. 1466. DOI: 10.3390/s16091466.

[4] Azam, M. F. i. *Location and Position Estimation in Wireless Sensor Networks*. University of Engineering and Technology, Lahore, Pakistan, 2016. 52 s.

[5] Blatný, P. *Mobilní aplikace využívající GPS modul*. 2012. 36 s. Diplomová práca. Vysoké učení technické v Brně.

[6] Dobbertin, H., Rijmen, V. a Sowa, A. *Advanced Encryption Standard - AES: 4th International Conference, AES 2004, Bonn, Germany, May 10-12, 2004, Revised Selected and Invited Papers*. Springer, 2005. Lecture Notes in Computer Science. ISBN 9783540265573.

[7] Eric B. *LoRa Documentation*. 2019. [dostupné 22.7.2020]. Dostupné z: https://lora.readthedocs.io/en/latest/.

[8] ETSI. *Electromagnetic compatibilityand Radio spectrum Matters (ERM); Short Range Devices (SRD); Radio equipment to be used in the 25 MHz to 1 000 MHz frequency range with power levels ranging up to 500 mW; Part 1: Technical characteristics and test methods*. Standard. European Telecommunications Standards Institute, 2012.

[9] Fargas, B. C. a Petersen, M. N. GPS-free geolocation using LoRa in low-power WANs. In: *2017 Global Internet of Things Summit (GIoTS)*. June 2017. DOI: 10.1109/GIOTS.2017.8016251. ISBN 978-1-5090-5873-0.

[10] Hillar, G. *MQTT Essentials - A Lightweight IoT Protocol*. Packt Publishing, 2017. ISBN 9781787285149.

[11] InvenSense Inc.. *MPU-6050 Product Specification*. PS-MPU-6000A-00, Revision 3.4. InvenSense Inc., August 2013. 52 s.

[12] Ismail Butun, M. G. Security Risk Analysis of LoRaWAN and Future Directions. *Future Internet 2019, 11, 3.* December 2018, zv. 22, s. 1466. DOI: 10.3390/fi11010003.

[13] Jeřábek, O. *Signálová analýza LoRa s využitím SDR.* 2018. 85 s. Diplomová práca. Vysoké učení technické v Brně.

[14] LoRa Alliance™. *LoRaWAN™1.1 Specification.* LoRa Alliance™, október 2017. 101 s.

[15] Mihalko, J. *MEMS INERCIÁLNÍ SNÍMAČE.* 2012. 77 s. Diplomová práca. Vysoké učení technické v Brně.

[16] Norbert Blenn, F. K. *LoRaWAN in the Wild: Measurements from The Things Network.* 1706.03086v1. Delft University of Technology, Mekelweg 4, 2628 CD Delft, The Netherlands, Jun 2017. 9 s.

[17] Semtech. *LoRa™ Modulation Basics.* AN1200.22, Revision 2. ©2015 Semtech Corporation, May 2015. 26 s.

[18] Semtech. *SX1276/77/78/79 - 137 MHz to 1020 MHz Low Power Long Range Transceiver.* August 2067. 133 s. Datasheet Rev.5.

[19] STMicroelectronics. *STM32L051x6 STM32L051x8.* STMicroelectronics, október 2019. 133 s. DS10184 Rev 10.

[20] The Things Industries. *Duty Cycle, The Things Network.* 2019. [dostupné 22.7.2020]. Dostupné z: https://www.thethingsnetwork.org/docs/lorawan/duty-cycle.html.

[21] The Things Industries. *Frequency Plans, The Things Network.* 2019. [dostupné 22.7.2020]. Dostupné z: https://www.thethingsnetwork.org/docs/lorawan/frequency-plans.html.

[22] The Things Industries. *Limitations, The Things Network.* 2019. [dostupné 22.7.2020]. Dostupné z: https://www.thethingsnetwork.org/docs/lorawan/limitations.html.

[23] The Things Industries. *LoRaWAN Architecture, The Things Network.* 2019. [dostupné 22.7.2020]. Dostupné z: https://www.thethingsnetwork.org/docs/lorawan/architecture.html.

[24] The Things Industries. *The Things Network.* 2019. [dostupné 22.7.2020]. Dostupné z: https://www.thethingsnetwork.org.

[25] u-blox. *NEO-M8: u-blox M8 concurrent GNSS modules.* Apríl 2020. 29 s. UBX-15031086 - R07.

[26] Wikipedia contributors. *Global Positioning System Wikipedia, The Free Encyclopedia.* 2020. [dostupné 22.7.2020]. Dostupné z: https://en.wikipedia.org/w/index.php?title=Global_Positioning_System&oldid=937902158.

[27] Wikipedia contributors. *LoRa Wikipedia, The Free Encyclopedia*. 2020. [dostupné 22.7.2020]. Dostupné z: https://en.wikipedia.org/w/index.php?title=LoRa&oldid=935084403.

[28] Wikipedie. *The Things Network — Wikipedie: Otevřená encyklopedie*. 2019. [dostupné 2.7.2020]. Dostupné z: https://cs.wikipedia.org/w/index.php?title=The_Things_Network&oldid=17984295.

Príloha A

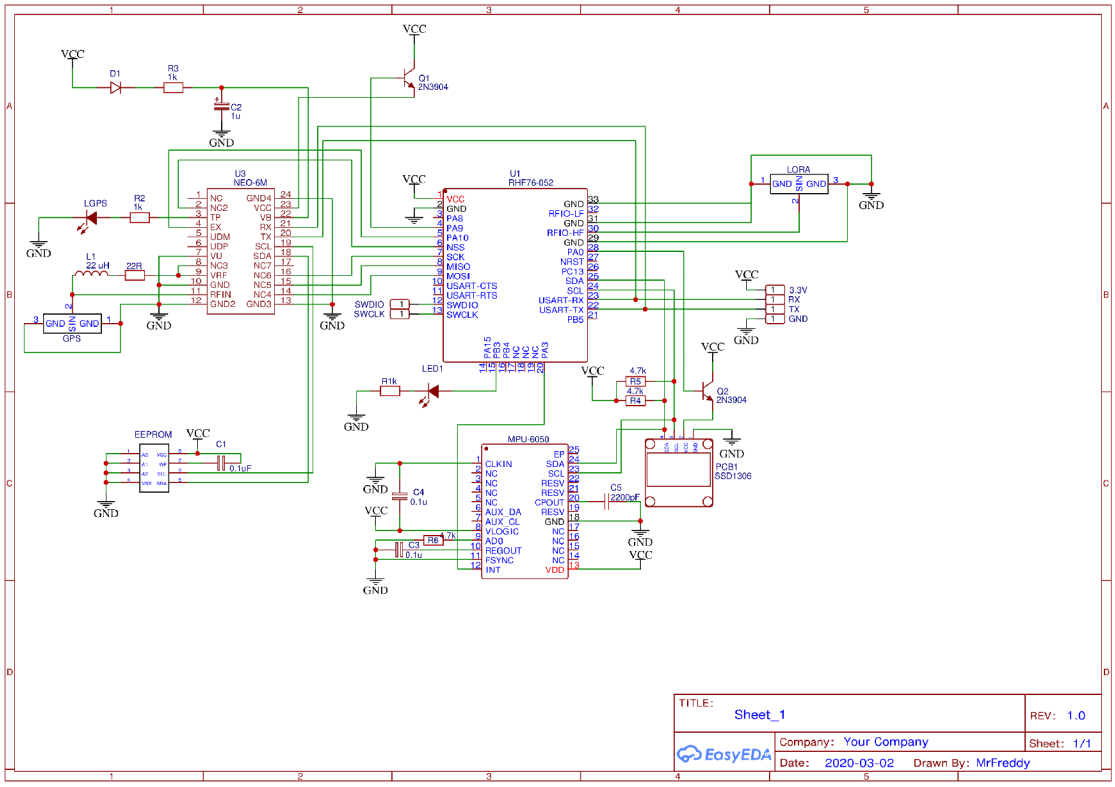
Obsah pamäťového média

Priložené médium CD obsahuje v práci vytvorené zdrojové kódy a materiály. V tabuľke je zobrazený jeho obsah a popis jednotlivých priečinkov.

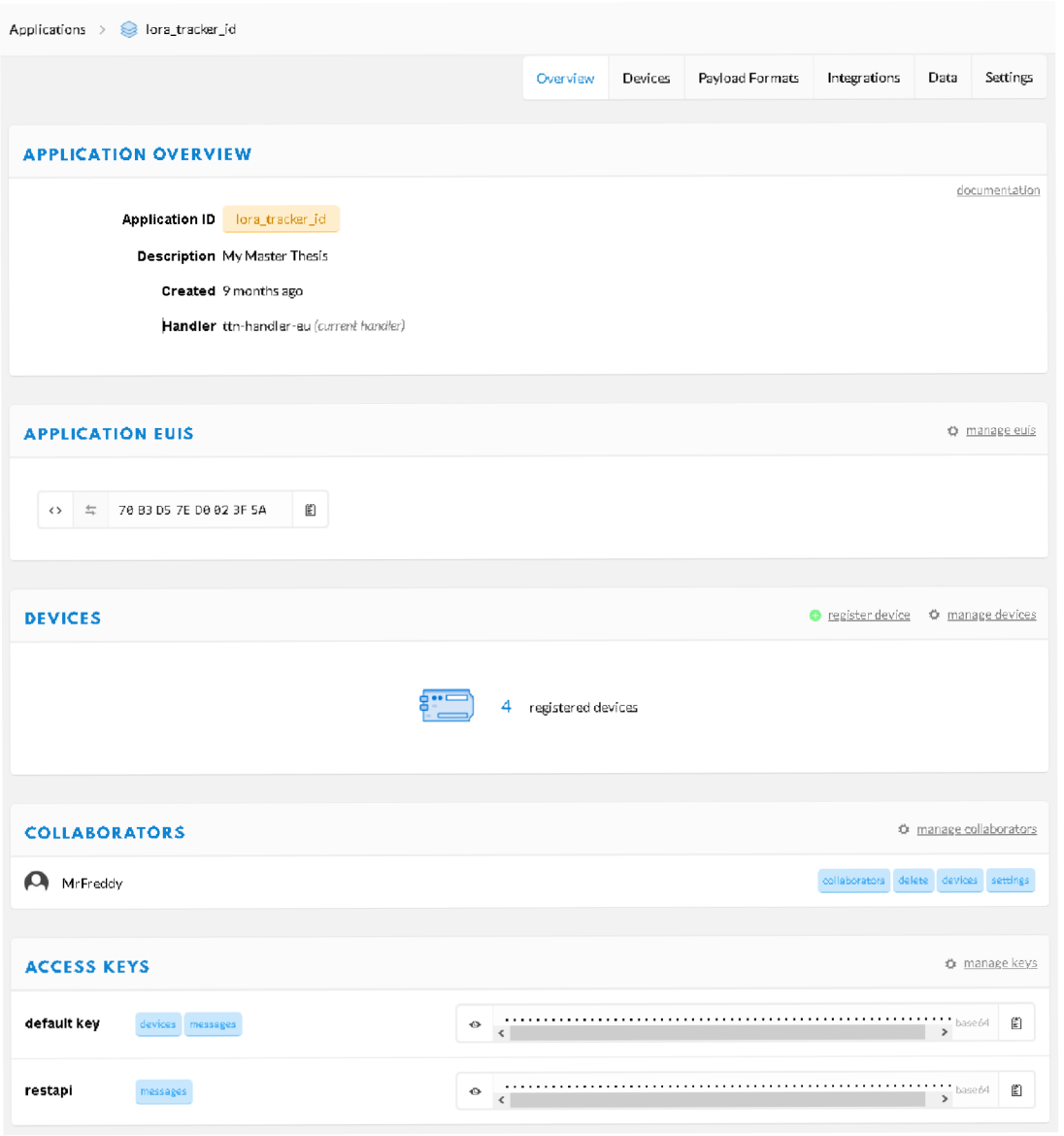
- hardware/
  - Schematic_MasterThesis.pdf — schéma zapojenia zariadenia lokátora
  - Gerber_PCB.zip — návrh plošného spoja zariadenia vo formáte Gerberfile.
- lora-gateway/ — obsahuje konfiguráciu a nastavenie stanice gateway počas testovania
  - Configuration.md — frekvencií a vysielacieho plánu
  - RAK7258.tar.gz — záloha konfigurácie gateway so všetkými nastaveniami
- lora-stm32-cube/ — kód pre framework STM32Cube
- lora-stm32-duino/ — kód pre framework STM32Duino
- masterthesis/ — zdrojový kód tejto práce s obrázkami
- mqtt-cloud/ — serverová aplikácia v C#
  - MyMasterThesis/ — zdrojové súbory pre serverovú aplikáciu a jej beh
  - MQTTCloud.sln — projekt pre Microsoft Visual Studio
- web-usermap/
  - dist/ — preložená webová stránka aplikácie
  - public/ — statické súbory
  - src/ — zdrojové kódy webovej aplikácie
  - package.json — informácie potrebné pre build
- praca.pdf — dokument tejto práce

Príloha B

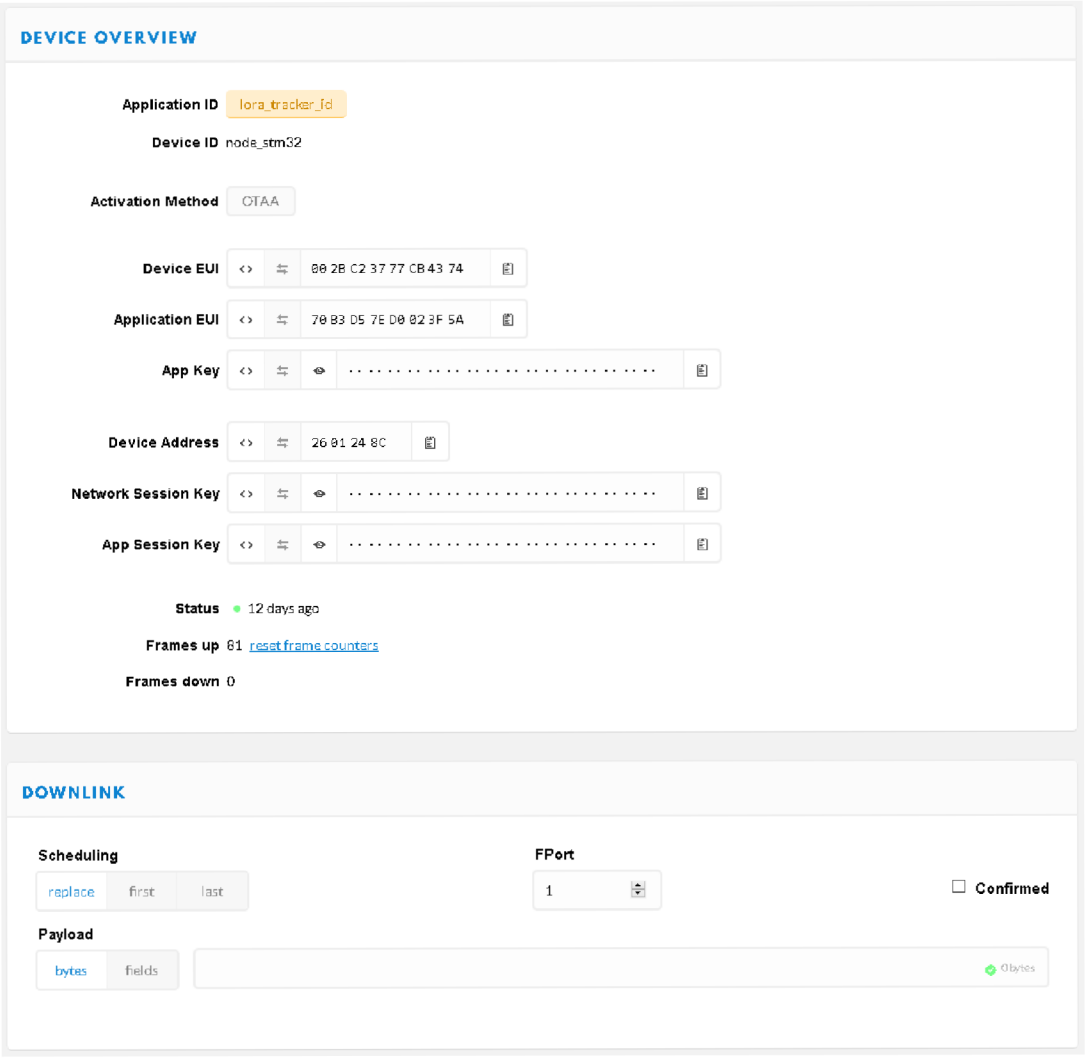
Obrázky



Obr. B.1: Schéma zapojenia lokátora



Obr. B.2: TTN Užívateľské rozhranie pre správu aplikácie



Obr. B.3: TTN Užívateľské rozhranie pre správu zariadenia