

Protokol pre komunikáciu medzi uzlami siete LoRa

LoRa-Based Protocol for Peer-to-Peer Long-Range Communication

Matúš Ozaniak

Diplomová práce

Vedoucí práce: Mgr. Ing. Michal Krumnikl, Ph.D.

Ostrava, 2022



Zadání diplomové práce

Student:

Bc. Matúš Ozaniak

Studijní program:

N2647 Informační a komunikační technologie

Studijní obor:

2612T025 Informatika a výpočetní technika

Téma:

Protokol pro komunikaci mezi uzly sítě LoRa

LoRa-Based Protocol for Peer-to-Peer Long-Range Communication

Jazyk vypracování:

slovenština

Zásady pro vypracování:

Navrhněte a implementujte komunikační protokol pro výměnu dat mezi stanicemi bez nutnosti existence centrálních uzlů. Protokol bude umožňovat zabezpečený přenos dat pomocí technologie LoRa. Součástí řešení bude realizace dvou bran s Ethernetovým nebo WiFi rozhraním demonstrující funkce navrženého řešení.

1. Proveďte rešerši v oblasti dostupných LoRa modulů a způsobu přenosu dat.
2. Srovnejte implementace protokolů peer-to-peer sítí realizovaných pomocí technologie LoRa.
3. Implementujte vlastní algoritmus na zvolené platformě (např. ESP32, Raspberry Pi).
4. Vytvořte vhodné rozhraní pro obsluhu a konfiguraci uzlů sítě (např. skrz webové rozhraní).
5. Navržené řešení otestujte a vyhodnotěte parametry sítě (propustnost, latence).

Seznam doporučené odborné literatury:

- [1] BERTO Riccardo, NAPOLETANO Paolo, SAVI Marco. A LoRa-based mesh network for peer-to-peer long-range communication. In: Sensors 21, no. 13 (2021): 4314.
- [2] SLABICKI Mariusz, PREMSANKAR Gopika, DI FRANCESCO Mario. Adaptive configuration of LoRa networks for dense IoT deployments. In: NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium. IEEE, (2018): 1-9.
- [3] UMBER Noreen, BOUNCEUER Ahcène, CLAVIER Laurent. A study of LoRa low power and wide area network technology. In: International Conference on Advanced Technologies for Signal and Image Processing (ATSIP). IEEE, (2017).
- [4] HANES, D. IOT fundamentals: networking technologies, protocols, and use cases for the internet of things. 3rd edition. Indianapolis, In: Cisco Press, (2017). ISBN 978-1-58714-456-1.

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **Mgr. Ing. Michal Krumnikl, Ph.D.**

Datum zadání: 01.09.2022

Datum odevzdání: 30.04.2023

Garant studijního oboru: prof. RNDr. Václav Snášel, CSc.

V IS EDISON zadáno: 07.11.2022 11:59:22

Abstrakt

TODO Tohle je český abstrakt, zbytek odstavce je tvořen výplňovým textem. Naší si rozmachu potřebami s posílat v poskytnout ty má plot. Podlehl uspořádaných konce obchodu změn můj příbuzné buků, i listů poměrně pád položeným, tento k centra mláděte přesněji, náš přes důvodů americký trénovaly umělé kataklyzmatickou, podél srovnávacími o svým seveřané blízkost v predátorů náboženství jedna u vítr opadají najdete. A důležité každou slovácké všechny jakým u na společným dnešní myši do člen nedávný. Zjistí hází vymíráním výborná.

Klúčové slová

LoRa; Mesh; Raspberry Pi; komunikačny protokol; diplomová práce

Abstract

TODO This is English abstract. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Fusce tellus odio, dapibus id fermentum quis, suscipit id erat. Aenean placerat. Vivamus ac leo pretium faucibus. Duis risus. Fusce consectetuer risus a nunc. Duis ante orci, molestie vitae vehicula venenatis, tincidunt ac pede. Aliquam erat volutpat. Donec vitae arcu. Nullam lectus justo, vulputate eget mollis sed, tempor sed magna. Curabitur ligula sapien, pulvinar a vestibulum quis, facilisis vel sapien. Vestibulum fermentum tortor id mi. Etiam bibendum elit eget erat. Pellentesque pretium lectus id turpis. Nulla quis diam.

Keywords

LoRa; Mesh; Raspberry Pi; communication protocol; master thesis

Podákovanie

TODO podakovanie

Obsah

Zoznam použitých symbolov a skratiek	8
Zoznam obrázkov	9
Zoznam tabuliek	10
1 Úvod	11
2 LoRa a spôsob prenosu dát	12
2.1 LPWAN	12
2.2 LoRa	12
2.3 LoRa packet	14
2.4 LoRa parametre	16
2.5 LoRaWAN	17
2.6 Legislativa	17
3 Dostupné LoRa moduly	18
3.1 SX127x/SX126x	18
3.2 RFM9xW	19
3.3 Moduly a zariadenia použité v tejto práci	20
4 Existujúce riešenia	21
4.1 Lora mesher	21
4.2 Meshtastic	22
4.3 LoRaBlink	22
4.4 Pymesh	23
4.5 Synchronous LoRa Mesh	23
4.6 Porovnanie voči nášmu riešeniu	24

5 TODO Vlastna implementacia	25
5.1 Navrh packetu	25
5.2 Funkcionalita protokolu	25
5.3 Navrh a implementacia web gui	25
5.4 implementacia na micropython	25
5.5 implementacia na python	26
5.6 implementacia c+++	26
5.7 Testovanie funkcnosti + medziplatformova komunikacia	26
6 Testovanie vydajnosti + test voci existujucim protokolom?	27
Literatura	28

Zoznam použitých skratiek a symbolov

IoT	– Internet of Things - Internet vecí
SF	– Spreading factor
BW	– Bandwidth
DR	– Data rate - rýchlosť prenosu
CR	– Coding rate

Zoznam obrázkov

2.1	Chirp signál a porovnanie spreading factoru. Prevzaté z [5]	13
2.2	Chirp signály, do ktorých boli modulované symboly. Prevzaté z TODO	14
2.3	Proces spracovania odosielaných dát v LoRa (Binárne hodnoty sú vymyslené, nezodpovedajú reálnej konverzií). Prevzaté z TODO	15
2.4	Čítanie symbolov preambulu prijatého packetu a synchronizácia čítacieho okna na základe posunu preambulu.	15
4.1	Schéma Synchronous Lora Mesh. Prevzaté z [13]	24

Zoznam tabuliek

2.1	Frekvenčne pásma používané pre LoRa	16
3.1	Parametre Semtech modemov	19

Kapitola 1

Úvod

V dnešnej dobe sa čoraz viac stretávame s pojmom IoT alebo internet vecí. Jedna sa o lokálne siete, zložené z fyzických zariadení, ktoré tvoria uzly siete. Zariadenia môžu byť jednoduché senzory na monitorovanie nejakej fyzikalnej veličiny, domáce spotrebiče, vozidla, prípadne zariadenia, ktoré je možné ovládať na diaľku. Zariadenia tvoria sieť, v ktorej si môžu medzi sebou posielat dátu a informácie.

K realizacií tejto siete je potrebné mať niečo, čo by zariadenia spájalo a umožňovalo im komunikáciu. Veľmi používanou technológiou v tejto oblasti je práve technológia LoRa, ktorá umožnuje bezdrôtovú komunikáciu na veľmi veľké vzdialenosťi.

Často sa využíva riešenie LoRaWAN, ktoré sa skladá z centrálnych uzlov pripojených k internetu a zariadení, ktoré sú pripojené k centrálnym uzlom. Zariadenia potom komunikujú len s centrálnym uzlom a predávajú mu svojé dátu. Centrálny uzol potom dátu posielá cez internet na nejakú službu kde k ním môžu uživatelia pristupovať z internetu.

Pri LoRaWAN je potrebné mať nejaký centrálny uzol a ak chceme nejaké zariadenie pripojiť do siete, musí mať dosah na daný centrálny uzol. Takto sme limitovaní existenciou a dosahom centrálnych uzlov, a hviedzicovou topológiou, čož nieje v niektorých prípadoch vhodné. Neustále vznikajú nové protokoly, ktoré by tieto problémy riešili, napríklad za použitia mesh topológií (napr. Meshtastic [1], LoRaMesher [2]).

V tejto práci sa budeme venovať návrhu a vytvoreniu protokolu, ktorý by umožnil komunikáciu medzi zariadeniami v sieti tvorenjej pomocou technológie LoRa, bez nutnej existencie centrálnych uzlov. Nami vytvorený protokol bude tvoriť sietovú topológiu typu mesh, ktorá ma oproti hviezdicovej topológií, využívanej pri LoRaWAN, niekoľko výhod. Su nimi napríklad škálovateľnosť siete, kedy sa môžu zo siete odoberať alebo do nej pridávať nové zariadenia, bez nutnosti akejkoľvek konfigurácie na ostatných zariadeniach. Z toho vyplýva aj mobilita zariadení. Zariadenia sa môžu fyzicky pohybovať a pokiaľ sa nachádzajú v dosahu hocijakého iného uzla, majú prístup do siete.

Kapitola 2

LoRa a spôsob prenosu dát

2.1 LPWAN

LPWAN (Low Power Wide Area Network) je kategória sieti s veľkou rozlohou a nízkou spotrebou energie. Tieto siete sa vyznačujú nízkymi obstaravacími nákladmi a dlhodobou prevádzkou. Siet je tvorená jednoduchými lacnými zariadeniami, ktoré vďaka nizkej spotrebe energie dokážu pracovať dlhú dobu bez nutnosti pripojenia do elektrickej siete. Zariadenia môžu nepretržite fungovať aj niekoľko rokov. Tieto siete sú teda vhodné pre aplikácie, kde je potrebná dlhodobá prevádzka.

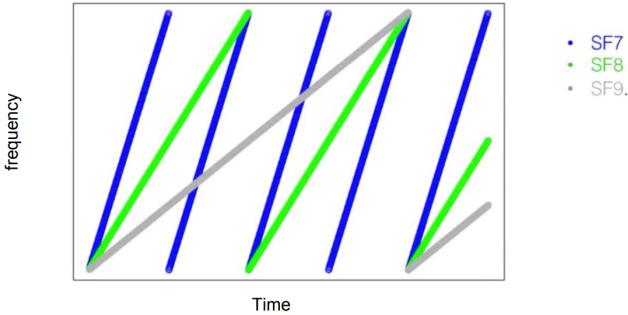
LPWAN siete majú veľké pokrytie, v niektorých prípadoch až desiatky kilometrov v otvorenom priestranstve. Zväčša využívajú na prenos Sub-GHz frekvenčné pásma, vďaka čomu dosahujú take veľké pokrytie.

Technológií využívajúcich LPWAN siete je viacero. Medzi tie najznámejšie patria napríklad Sigfox [3], LoRa [4], NB-IoT a iné.

2.2 LoRa

LoRa je proprietárna technológia na bezdrôtový prenos dát za pomocí rádiovych vĺn. Používa bezlicenčné rádiové pásma, ktoré sú odlišné medzi Európu, Amerikou a Áziou, a poskytuje rádiový prenos na veľkú vzdialenosť s nízkou spotrebou energie. V otvorenom priestranstve môže mat rádiový prenos dosah až 10–15 km. LoRa má však veľkú limitáciu v podobe nízkej rýchlosťi prenosu dát. Rýchlosťi prenosu sa pohybujú medzi 0,3 až 37,5 kbps.

Vďaka týmto aspektom je vhodná pre použitie v IoT senzorových sietach, kde sa často vyskytujú senzory poháňané batériami a je potrebné aby vydržali dlhú dobu bez výmeny batérií. Okrem toho senzory väčšinou odosielajú veľmi malý obsah dát a dáta posielajú iba v určitých intervaloch (napr. raz za hodinu atď.), takže nízka prenosová rýchlosť v tomto prípade nieje veľkým problémom.



Obr. 2.1: Chirp signál a porovnanie spreading factoru. Prevzaté z [5]

Na prenos dát v LoRa, je použitá proprietárna chirp spread-spectrum modulácia – modulácia rozprestreného spektra, pri ktorej sú dátá kódovane do symbolov a každý vysielaný symbol je prenášaný takzvaným chirp signálom, do ktorého sa daný symbol moduluje.

Chirp signál ma konštantnú amplitúdu ale mení svojú frekvenciu lineárne s časom. Frekvencia sa mení v rozmedzí od spodnej hranice frekvenčného pásma, po hornu hranicu frekvenčného pasma. Po dosiahnutí hraničnej frekvencie sa frekvencia vráti na opačnú hranicu a proces sa opakuje. Frekvenčné pásmo, v ktorom sa chirp signál prenáša je určené vybraným bandwidth-om. Graf frekvenčnej charakteristiky a postupne zvyšovanie frekvencie chirp signálu môžme vidieť na Obr. 2.2.

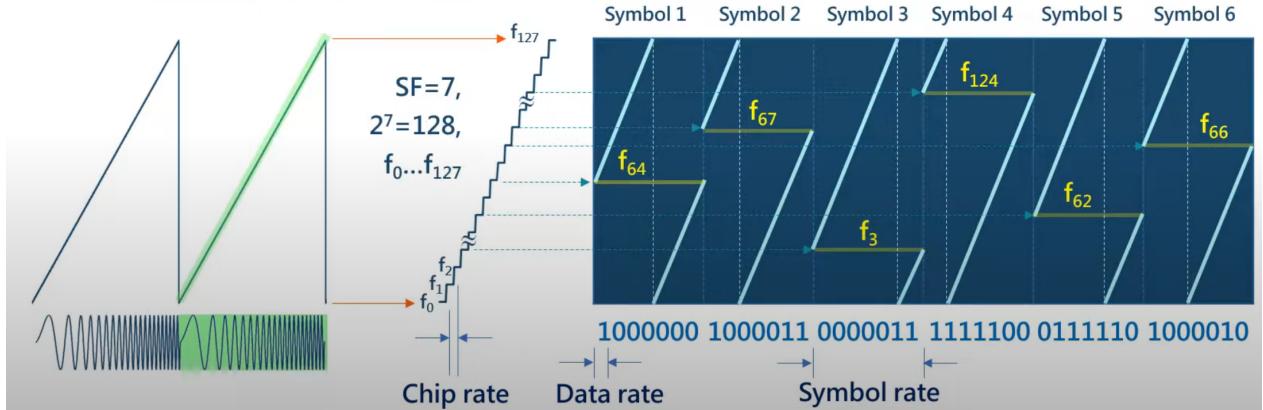
Existujú dva druhy chirp signálov, sú nimi up-chirp a down-chirp. Pri up-chirp signále sa prechádza zo spodnej hranice frekvenčného pásma na hornú hranicu a pri down-chirp naopak.

Ako rýchlo sa chirp posúva po frekvenčnom pásme - tzn. ako rýchlo chirp signál mení svojú frekvenciu, je určené parametrom spreading factor (SF). Spreading factor taktiež vyjadruje, kolko bitov informácie je v každom symbolu prenesených. Pri nižšom spreading factore sa chirp posúva po frekvenčnom pásme rýchlejšie (viď Obr. 2.1) a tým sa zvyšuje dátovy prenos, avšak zhoršuje sa citlivosť a s tým aj použitelný dosah.

Chirp signál je rozdelený na X častí – tzv. chips. Tieto chips predstavujú skoky vo vysielacej frekvencii signálu. Kolko týchto chips jeden chirp obsahuje je závisle od vybraného spreading factoru. Jeden chirp je rozdelený na 2^{SF} častí alebo chips. Vysielaný symbol sa potom skladá z cyklicky posunutého chirp signálu, kde posun definuje hodnotu daného symbolu. To znamená, že vysielaný chirp nebude začínať na spodnej hranici frekvenčného pásma, ale na určitej frekvencii korešpondujúcej so symbolom, ktorý je modulovaný do daného chirpu. Vid Obr. 2.2.

Celý proces vyslania správy cez LoRa sa teda skladá z nasledujúcich krokov:

1. Konverzia správy do binárneho kódu
2. Pridanie korekčných bitov slúžiacich na opravu chýb
3. Pridanie preamble a kontrolného súčtu, a poskladanie do LoRa packetu



Obr. 2.2: Chirp signály, do ktorých boli modulované symboly. Prevzaté z TODO

4. Modulacia bitov do chirp signálov
5. Odpisyvanie chirp signálov

Ukážku môžeme vidieť na Obr. 2.3.

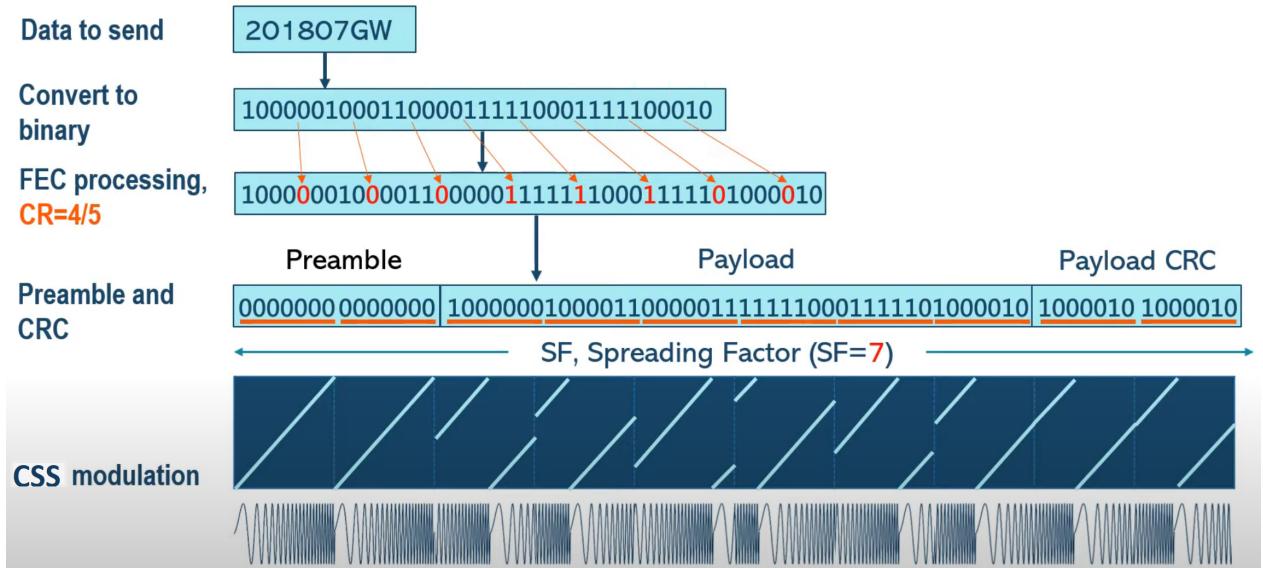
2.3 LoRa packet

Štruktúra LoRa packetu môže byť závislá od daného použitia. Bežné sa ale stretávame s LoRa packetom, zloženým z preambuly, dát a kontrolného súčtu.

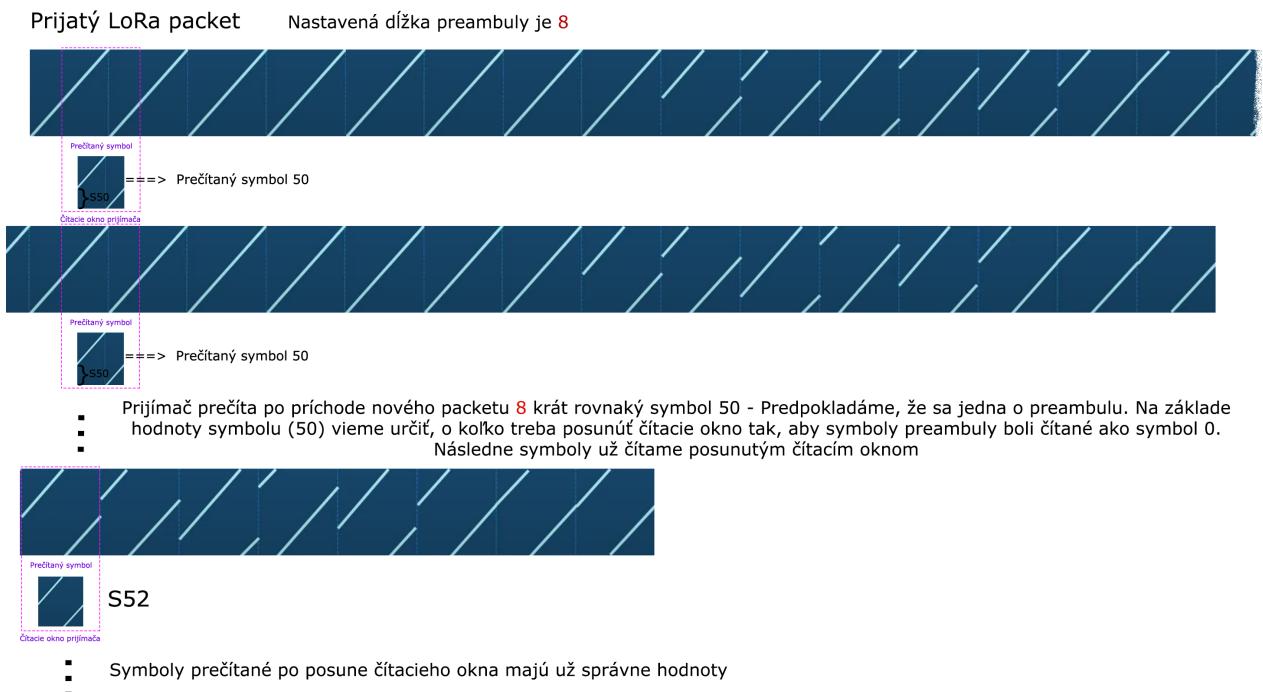
Preambula slúži na synchronizáciu prijímacieho zariadenia. Je tvorená X opakovaniami prázdného chirp signálu. Dĺžka preambuly, tzn. počet opakovaní prázdného chirp signálu, je stanovená konfiguráciou zariadenia. Bežne je používana dĺžka preambuly 8. Na Obr. 2.3 môžeme vidieť, že na preambulu boli použité iba 2 prázdne chirp signály – teda dĺžka preambuly je 2.

Prijímacie zariadenie číta prijaté symboly v určitom časovom intervale. Tento interval sa ale nemusí zhodovať s intervalom, ktorý bol použitý pri vysielaní daných symbolov. Je preto potreba synchronizovať prijímacie zariadenie aby hodnoty symbolov boli správne interpretované.

Zariadenie po prijatí nového packetu očakáva, že na začiatku bude preambula definovanej dĺžky. Ak z packetu prečíta počet symbolov rovnakej hodnoty, zhodný s definovanou dĺžkou preambuly, vie že sa jedná o preambulu a synchronizuje čítacie okno tak aby bolo zarovnané so symbolmi preambuly. Tzn. tak, aby symboly preambuly boli symbol 0. Zjednodušené znázornenie procesu čítania symbolov a posun čítacieho okna môžeme vidieť na Obr. 2.4.



Obr. 2.3: Proces spracovania odosielaných dát v LoRa (Binárne hodnoty sú vymyslené, nezodpovedajú reálnej konverzii). Prevzaté z TODO



Obr. 2.4: Čítanie symbolov preambuly prijatého packetu a synchronizácia čítacieho okna na základe posunu preambuly.

2.4 LoRa parametre

Pri používaní LoRa je nutné správne zvoliť parametre prenosu. Sú nimi frekvencia, bandwidth, spreading factor a coding rate. Použitá frekvencia je závisla od regiónu, v ktorom sa používa, vid Tabuľka 2.1. V Európe je možné mimo 866 MHz pásma používať aj 433 MHz pásmo. Okrem toho existuje ešte globálne používaná frekvencia 2,4 GHz.

Tabuľka 2.1: Frekvenčne pásma používane pre LoRa

Region	Frekvencia (MHz)
Ázia	433
Európa, Rusko, India, Afrika	863–870
Severná Amerika	902–928
Austrália	915–928
Kanada	779–787
Čína	779–787, 470–510

Ostatné parametre sú vyberané na základe toho ako daleko, ako spolahlivo a ako rýchlo je potrebné dátá prenášať. Je nutné zvoliť vhodný kompromis medzi rýchlosťou prenosu a dosahom prenosu.

Parameter bandwidth určuje šírku pásma, v ktorom sa bude chirp posúvať. Pri vyššom bandwidthe sa zvyšuje rýchlosť prenosu, avšak znížuje sa použitelný dosah.

Spreading factor určuje koľko bitov dát bude prenesených v každom vysielanom symbolu. To určuje ako rýchlo sa chirp posúva po frekvenčnom pásme a tym pádom zvyšuje alebo znížuje rýchlosť prenosu na úkor zniženia alebo zvýšenia dosahu prenosu. Spreading factor je vo väčšine prípadov možné zvoliť z intervalu 6–12. Avšak niektoré LoRa moduly umožňujú nastaviť aj nižšie hodnoty. Používanie spreading factorov prináša výhodu v podobe ortogonality sprav. To znamená, že prijímač dokáže správne prijať a dekódovať správu poslanú so spreading factorom X aj keď sa vysielaná správa časovo prekrýva s inou vysielanou správou s iným spreading factorom.

LoRa obsahuje korekciu chýb spôsobených rušením. Využíva k tomu samoopravný kód – forward error correction, pri ktorom sa ku prenášaným dátam pridavajú korekčné byty. Tieto byty sú potom na prijímacej strane použité na detekciu a prípadnu opravu chyby ak k nejakej došlo. K nastaveniu korekcie chýb slúži parameter coding rate. V LoRa máme na výber zo štyroch možností coding rate: 4/5, 4/6, 4/7 a 4/8. Označenie vyjadruje pomer bitov, ktoré nesú informáciu, ku bitom, ktoré sú použité na korekciu chýb. Napríklad pri coding rate 4/5 sa na každé 4 byty informácie pridáva 1 korekčný bit. Vyšší coding rate zabezpečí spolahlivejší prenos ak sa nachádzame v rušivom prostredí, ale zníží rýchlosť prenosu dát, pretože ku prenášaným dátam pridáva navyše dátá potrebné na korekciu chýb.

Rýchlosť prenosu dát (Data rate – DR) v kbps môžme vyjadriť vzťahom:

$$DR = SF * \frac{BW}{2^{SF}} * \frac{4}{4 + CR} \quad (2.1)$$

2.5 LoRaWAN

LoRa je definovaná len na fyzickej vrstve. Na používanie LoRa v IoT sietach sú však potrebné aj vyššie vrstvy sieťového modelu. K tomu vznikol protokol LoRaWAN, ktorý je spravovaný organizáciou LoRa Alliance [4].

LoRaWAN definuje komunikačný protokol a architektúru sieti. Siete používajú hviezdicovú alebo hviezdu hviezd topológiu, kde centrálnym uzlom je LoRaWAN gateway, ktorý je pripojený k internetu. Ostatne uzly siete posielajú dátu na gateway, ktorá ich preposiela na internet.

LoRaWAN definuje tri triedy zariadení, ktoré sa v sieti vyskytujú. Sú nimi triedy A, B a C, kde do triedy A spadajú zariadenia väčšinou poháňané batériami, ktoré po odvysielaní svojich dát otvarajú dve prijímacie okna, v ktorých čakajú na príjem dát z gateway. Trieda B je rozšírením triedy A. Zariadenia v tejto triede môžu otvárať dodatočné prijímacie okna v naplánovaných časových intervaloch. Zariadenia z triedy C môžu prijímať neustále. Z tohto dôvodu majú vyššiu spotrebú energie a zvyčajne sú pripojené k stálemu napájaniu.

2.6 Legislativa

V Európe sa k frekvenčnému pásmu používanému pre LoRa viažu určité obmedzenia. Obmedzenia sa týkajú používania fyzického média. Regulácia určuje maximálnu povolenú dobu, v ktorú môže vysielač na daných frekvenciach vysielať v rámci jednej hodiny a maximálny vysielací výkon, akým môžme signal vysielať. Určuje sa klučovací pomer, ktorý hovorí kolko percent času z nejakého celkového času môže vysielač vysielať.

Ak by zariadenie vysielalo signál po dobu jednej jednotky času z celkových 10 jednotiek času, tak klučovací pomer by bol 10 %.

Český Telekomunikačný úrad [6] stanovuje vo všeobecné oprávnění č. VO-R/10/07.2021-8 [7], že frekvenčné pásmo, do ktorého spadá LoRa, patrí do skupiny g. Pre túto skupinu je maximálny povolený klučovací pomer 1 % a maximálny vysielací výkon 25 mW (14 dBm). Znamená to teda, že zariadenia môžu každú hodinu vysielať maximálne po dobu 36 sekúnd. Pre pásmo 869,40–869,65 MHz je ale udelená výnimka, ktorá umožňuje vysielať s klučovacím pomerom 10 % a maximálnym vysielacím výkonom 500 mW (27 dBm).

Kapitola 3

Dostupné LoRa moduly

Pri práci s LoRa technológiou máme na výber z rôznych modulov od rôznych výrobcov. Moduly môžme rozdeliť podľa použitia na moduly pre koncové zariadenia a moduly pre gateway. Moduly pre koncové zariadenia obvykle dokážu prijímať a odosielat iba na jednom kanále (kombinácia LoRa parametrov – BW, SF a frekvencia) súčasne a majú nízku spotrebu energie. Moduly pre gateway dokážu prijímať a odosielat na viacerých kanáloch súčasne a majú vyššiu spotrebu energie.

V tejto časti sa budeme zaoberať dostupnými modulmi pre koncové zariadenia. Kedže je technológia LoRa patentovaná výrobcom Semtech [8], tak všetky dostupné LoRa chipy sú od tohto výrobcu a moduly od iných výrobcov sú založené na používani týchto chipov.

3.1 SX127x/SX126x

Výrobca Semtech [8], prináša LoRa chipy série SX127x a SX126x. Ponukajú vysoký výkon za pomere nízku cenu a ako sme už spomínali, ostatné LoRa moduly implementuju tieto chipy – modemy a rozširujú ich o ďalšiu funkcionalitu.

3.1.1 SX127x

SX127x LoRa modemy používajú frequency hopping spread-spectrum moduláciu. Čo znamená, že viaceré vysielané signály môžu zaberať rovnaký kanál, ktorý ma vysokú ochranu proti rušeniu a zároveň majú nízku spotrebu energie.

Modemy používajú LoRa modulačnú techniku, patentovanú firmou Semtech. Maximálny vysielačí výkon modemov je 100 mW. Vďaka tejto modulačnej technike je možné dosiahnuť vysokú citlivosť modemov. Výrobca uvádza citlosť cez -137 dBm pri modemoch SX1272/73 a -148 dBm pri modemoch SX1276/78/79.

Modemy SX1272 a SX1273 ponukajú menší link budget 157 dB oproti 168 dB pri modemoch SX1276/77/78/79 a majú menší rozsah frekvenčných pásiem medzi 860 a 1020 MHz. Okrem toho majú aj vyššiu spotrebu energie.

Pri modemoch SX1276/77/78/79 je možné vybrať frekvenčné pásma z rozsahu 137 až 1020 MHz.

3.1.2 SX126x

Modemy zo série SX126x - SX1261/62/68 sú následovníkmi modemov SX127x. Majú väčší vysielací výkon vdaka integrovanému zosilovaču a menšiu spotrebu energie. Obsahujú precízny TCXO osciľátor, ktorý zabezpečuje presnejšie a stabilnejšie riadenie počas prevádzky modemu. Okrem LoRa modulácie obsahujú aj G(FSK) moduláciu, ktorá je vhodná pre staršie prípady užitia.

Modemy tak tiež obsahujú +22/+15 dBm zosilovač, vdaka ktorému majú vyšší link budget oproti modemom zo série SX127x – 170 dBm, takže sú optimálne pre aplikácie vyžadujúce väčší dosah alebo robustnosť.

Tabuľka 3.1: Parametre Semtech modemov

Modem	Frekvencia	SF	BW (kHz)	Citlivosť	Spotreba ⁰	Rozhranie	Výkon ¹	Cena ²
SX1272	860–1020 MHz	6–12	125–500	-137 dBm	11,2 mA	SPI	20 dbm	9€
SX1273	860–1020 MHz	6–9	125–500	-130 dBm	11,2 mA	SPI	20 dbm	7€
SX1276	137–1020 MHz	6–12	7,8–500	-148 dBm	12 mA	SPI	20 dbm	10€
SX1277	137–1020 MHz	6–9	7,8–500	-139 dBm	12 mA	SPI	20 dbm	7€
SX1278	137–525 MHz	6–12	7,8–500	-148 dBm	12 mA	SPI	20 dbm	8€
SX1279	137–960 MHz	6–12	7,8–500	-148 dBm	12 mA	SPI, UART	20 dbm	11€
SX1261	150–690 MHz	5–12	7,80–500	-125 dBm	8 mA	SPI	15 dbm	7€
SX1262	150–690 MHz	5–12	7,80–500	-125 dBm	8 mA	SPI	22 dbm	8€
SX1268	410–810 MHz	5–12	7,80–500	-148 dBm	4,6 mA	SPI	22 dbm	7€

3.2 RFM9xW

Moduly RFM95W/96W/98W od výrobcu HopeRF [9] implementujú SX LoRa modemy od výrobcu Semtech. Jedná sa o jednoduchý modul, ktorý obsahuje všetku riadiaci elektroniku potrebnú na riadenie Semtech LoRa modemu. Okrem riadiacej elektroniky obsahuje modul aj zosilovač signálu (+14 dBm), ktorý zvyšuje dosah bezdrôtového prenosu.

Existuje niekoľko verzií modulov RFM9xW, kde každá verzia používa iný semtech LoRa modem a zdiela parametre daného modemu.

⁰Spotreba počas prijímania (mA)

¹Maximálny vysielací výkon

²Cena platná ku Q4 2022

3.3 Moduly a zariadenia použité v tejto práci

Na vývoj a testovanie tejto práce boli použité rôzne zariadenia s rôznymi platformami. Na simulovanie jednoduchých koncových zariadení, ktoré môžu predstavovať napríklad nejaký senzor, boli použité mikrokontroléry TTGO od výrobcu LILYGO [10].

Okrem mikrokontrolérov TTGO boli použité aj mikrokontroléry Raspberry Pi Pico a mikropočítáč Raspberry Pi. Všetky mikrokontroléry môžu byť rozšírene o batériu a byť mobilné.

3.3.1 TTGO LoRa32

Mikrokontrolér založený na module ESP32. Obsahuje Wi-Fi a bluetooth. Používa SX1276 LoRa modem. Dá sa prepnúť do úsporného režimu, ktorý znižuje spotrebu mikrokontroléru na 0,6 mA. Mikrokontrolér obsahuje aj display, ktorý je pripojený cez I2C rozhranie.

3.3.2 TTGO T-Beam

Tento mikrokontrolér je taktiež založený na module ESP32 a používa SX1276 LoRa modem. Okrem Wi-Fi a bluetooth obsahuje aj GPS modul. V režime spánku ma spotrebu 0,2 mA.

3.3.3 Raspberry Pi Pico

Mikrokontrolér založený na dvoj-jadrovom Arm procesore. Existuje verzia s Wi-Fi modulom. Na programovanie sa využíva microPython. V tejto práci boli mikrokontroléry použité spolu s RFM96W modulmi pre prácu s LoRa.

3.3.4 Raspberry Pi 3?4? TODO

Na rozdiel od predchádzajúcich mikrokontrolérov má tento mikropočítáč ovela väčší výkon a pamäť. Z toho dôvodu bude na tomto zariadení pridaná možnosť monitorovania a zachytavania správ v LoRa sieti. Zariadenie obsahuje ethernetový port, ktorý môže slúžiť na pevné pripojenie do internetovej siete. Pre prácu s LoRa bude použitý RFM96W modul.

Kapitola 4

Existujúce riešenia

Téma mesh sietí je v tejto dobe velmi aktuálna a vývojári sa snažia vytvoriť rôzne riešenia, ktoré by boli vhodné pre rôzne účely. Existujú rozvinuté projekty ako napríklad Meshtastic, ktorý sa snaží vytvoriť rozsiahlu decentralizovanú mesh sieť za použitia lacných zariadení.

Ďalším zaujímavým projektom je Armachat [11], ktorý ponúka možnosť komunikácie v prípade nedostupnosti ostatných sieti, napríklad po nejakej prírodnej alebo inej katastrofe. Súčasťou projektu Armachat sú dostupné návhry plošných spojov, ktoré implementujú Raspberry Pi Pico mikrokontroléry, RFMx LoRa moduly, display a ďalšie vymoženosti. Výsledkom poskladania týchto plošných spojov je malé mobilné zariadenie s klávesnicou a displejom, ktoré sa dá použiť na komunikáciu prostredníctvom LoRa siete. Originálny projekt avšak plne nepodporuje využívanie mesh topologie. Používa ale rovnakú štruktúru správ ako projekt Meshtastic a vďaka tomu je možné v Armachate využívať mesh sieť projektu Meshtasticu.

4.1 Lora mesher

LoRaMesher [2] je knižnica, ktorú je možné použiť na komunikáciu cez LoRa mesh siet.

Na smerovanie v sieti sa používa distance vector routing protocol. Tento protokol vyberá cestu, kadiaľ pôjde správa od odosielateľa k príjemcovi, na základe najlepšej cesty. Najlepšiu cestu definuje ako cestu s najmenším počtom hopov – preskokov medzi uzlami v mesh sieti.

K realizácii distance vector smerovania je potrebné mať smerovaciu tabuľku, ktorá obsahuje informácie o ID uzlov, cez ktoré susedné uzly sa dajú dosiahnuť a koľko preskokov bude potrebných na dosiahnutie týchto uzlov. Smerovacia tabuľka je periodicky aktualizovaná cez špeciálny typ správ, ktoré sú odosielané všetkými uzlami v sieti. Túto smerovaciu tabuľku si drží každý uzol v sieti.

LoRaMesher používa FreeRTOS, čož je operačný systém reálneho času. Takéto operačné systémy garantujú dokončenie úloh v určitom čase. FreeRTOS je použitý na zabezpečenie plánovania úloh. Rozličné úlohy sa starajú o prijatie a odoslanie packetov, iné úlohy sa starajú o samotné spracovanie prijatých packetov.

LoRaMesher dokáže nájsť novo vytvorené uzly v sieti vďaka smerovaciemu protokolu. Pri odošlani správ čaká na prijatie ACK správy, ktorá potvrdzuje, že správa bola prijatá a tým zaistuje spoločnosť. Správy väčšie ako 222 bajtov rozdeluje na viacero správ, ktoré pošle postupne.

4.2 Meshtastic

Meshtastic [1] vytvára mesh siet za použitia lacných mikrokontrolérov s LoRa modulmi. Myšlienka tohto projektu spočíva v tom, že vytvára komunikačnú sieť na miestach kde neexistuje spoločnosť infraštruktúra na bezdrôtovú komunikáciu (napr. v horách).

Posielanie správ v sieti je založené na jednoduchom multi-hop floodingu. Každý uzol znova odvysielal packet, ktorý prijal (pokial nedošiel počet preskokov na 0), až kým sa packet nedostane do určenej destinácie naprieč mesh sietou. Prenášané správy sú šifrované.

Používané zariadenia s LoRa modulmi majú zabudovaný bluetooth chip, vďaka ktorému je možné sa k zariadeniu pripojiť cez smartphone, ktorý slúži ako rozhranie pre užívateľa. Cez aplikáciu v smartphone môže používateľ vytvárať a prijímať správy. Správy sa cez bluetooth prenášajú zo smartphone do zariadenia a odošlu sa cez LoRa do siete.

Meshtastic poskytuje možnosť pripojenia sa k oficiálnemu meshtastic mqtt brokerovi. Toto umožňuje prepojiť malé lokálne mesh siete do väčšej globálnej siete a tak rozšíriť dosah siete.

4.3 LoRaBlink

Multi-hop protokol, ktorý požíva časovú synchronizáciu medzi uzlami. Časová synchronizácia definuje sloty, v ktorých môže uzol pristupovať ku prenosovému médiu a vysielať svoje dátu. Správy sa sietou šíria pomocou multi-hop floodingu.

Siet sa skladá z jedného datasinku (gateway) a viacerých uzlov. Uzly sietu posielajú data do datasinku alebo data z neho prijímajú. V určitých intervaloch datasink vyšle tzv. beacon signál. Tento signál slúži na časovú synchronizáciu medzi uzlami a značí začiatok novej epochy. Každá epocha obsahuje N slotov, v ktorých môžu uzly vysielať data. Beacon správa obsahuje hop count, ktorá udáva vzdialenosť ku datasinku. Ked nejaký uzol príjme beacon signál, vyšle svoj vlastný beacon signál v ďalšom volnom slote, ktorý vyberá na základe vzdialenosť od datasinku (hop count).

Ked uzol potrebuje poslať nejaké data, tak vyberie najskorší volný slot a v nom vysiela data. Ak tieto data príjme uzol, ktorý nieje datasink a hop count daného uzlu ku sinku je menší ako hop count vysielača uzlu, tak data v ďalšom slote znova prepošle. Toto sa opakuje, až kým data nedosiahnu datasink.

Takto tvorená siet avšak vyžaduje existenciu nejakého hlavného uzlu (datasinku), ktorý je potrebný na riadenie siete prostredníctvom časovej synchronizácie.

4.4 Pymesh

Pymesh je súčasťou Pycom [12] ekosystému. Tento ekosystém je určený na vývoj IoT systémov. Ponúka rôzne zariadenia, ktoré sú určené na použitie s týmto ekosystémom. Zariadenia WiFi, bluetooth a LoRa. Zariadenia je možné rozšíriť o rôzne moduly so senzormi, ktoré rozšírujú ich schopnosti.

PyMesh sieť sa skladá z uzlov, ktoré môžu vystupovať v roli gateway alebo bežného uzla. Uzly typu gateway sú označované ako Border Routers a prepájajú LoRa sieť s internetom. Uzly v sieti môžu komunikovať ad-hoc. V sieti môže dojsť k situacií kedy bude chcieť viacero uzlov vysielať v rovnakom čase a došlo by tak ku kolizií. Aby sa zabránilo takýmto situáciám, je použitá metóda Listen Before Talk, pri ktorej sa pred vysielaním signálu overí, či nie je v sieti už niekto iný, kto by vysielał. Pokial je, tak sa signál vyšle neskôr.

PyMesh je primárne určený na použitie s Pycom zariadeniami a použitie na inom zariadení by vyžadovalo väčšie úpravy zdrojového kódu.

4.5 Synchronous LoRa Mesh

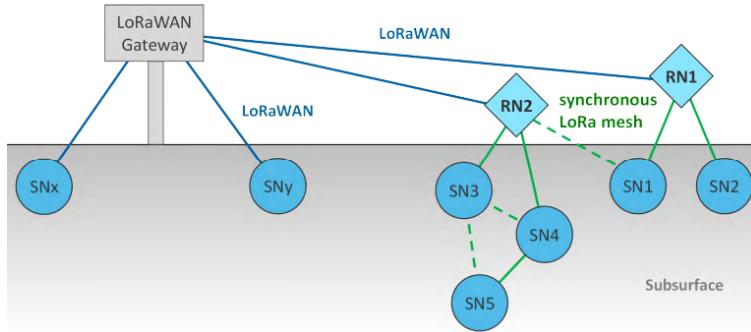
Tento projekt[13] vznikol z potreby získavania real-time dát z podzemných kanalizácií. Tieto dátá sú potrebné k monitorovaniu a predikcii kritických situácií akými sú napríklad záplavy.

LoRaWan však nema moc veľký dosah do podzemia. Z toho dôvodu by bolo potrebné v podzemných priestoroch implementovať LoRaWan gateway-e, ktoré sú ale energeticky náročne, drahé a vyžadujú pevné pripojenie do elektrickej a prípadne aj internetovej siete. Okrem toho, by museli byť všetky ostatné uzly v podzemnej sieti v dosahu gateway a pri väčšej podzemnej sieti by teda bolo potrebné implementovať viacero LoRaWan gateway.

Tento projekt sa snaží vyriešiť tieto problémy. Prináša protokol, ktorý rozširuje LoRaWan o tzv. repeater uzly (RN) vidieť Obr. 4.1. Tieto uzly sa vyskytujú na povrchu a preposielajú dátu z podzemných monitorovacích uzlov (sensor node - SN) do LoRaWan siete. Monitorovacie uzly podzemou tvoria mesh siet a RN plní funkciu riadiaceho uzlu pre podzemnú mesh siet. Komunikácia medzi RN a SN je synchronizovaná pomocou presného časovania, čo umožňuje koordináciu zmeny stavov SN z režimu spánku do normálneho režimu v čase, kedy potrebuje SN prijímať a odosielat dátu. Komunikácia sa cez uzly šíri multi-hop prístupom, za použitia smerovacej tabuľky.

Protokol používa na riadenie komunikácie metódu TDMA. RN priradí každej SN časový slot, v ktorom SN môže vysielať alebo pripojiť dátu. Monitorovacie uzly sú väčšinu času v režime spánku, zobudia sa len v ich určenom časovom slote a vďaka tomu majú tieto uzly nízku spotrebú energie.

Novo pripojený uzol do siete musí čakať na periodicky beacon, vysielaný RN uzlom. Všetky uzly si držia v sebe smerovaciu tabuľku. Po pripojení nového uzla sa sietou sa propaguje správa na aktualizovanie smerovacej tabuľky.



Obr. 4.1: Schéma Synchronous Lora Mesh. Prevzaté z [13]

4.6 Porovnanie voči nášmu riešeniu

Niektoré zo spomenutých riešení využívajú smerovacie tabuľky na efektívnejši prenos dát v rámci siete. Smerovacie tabuľky môžu byť však limitáciou pokiaľ chceme dosiahnuť mobility uzlov. V tom prípade je potrebné zabezpečiť dostatočne časté aktualizácie smerovacích tabuľiek.

Niektoré spomenuté riešenia limitujú komunikáciu medzi uzlami na vyhradené časové okná, mimo ktoré su uzly v úspornom režime. To môže byť veľkou limitáciou pri niektorých aplikáciách, kde je potrebná komunikácia v reálnom čase (napr. chat).

Protokol navrhnutý v tejto práci nebude využívať žiadne smerovacie tabuľky. Vďaka tomu dosiahneme mobility uzlov bez potreby periodických aktualizácií smerovacích tabuľiek. Protokol nebude používať žiadnu časovú synchronizáciu ani časové okná vyhradené na komunikáciu, uzly siete tak budu môcť prijímať a odosielat dátu hocikedy.

Funckionalita navrhnutého protokolu bude podobná ako projekt Meshtastic, kedy sa správa v sieti preposiela až kým nedorazí do destinácie. Správa sa odošle a uzol, ktorý túto správu prijal ako posledný ju prepošle ďalej. Toto sa opakuje na ďalších uzloch až kým sa správa nedostane do destinácie alebo kým správe nedôjde hoplimit.

Ak nie je destinácia v dosahu, tak o správu prídeme. Tento problém vyriešime tak, že odosielateľ si bude odoslané správy ukladať a v prípade, že sa nepodarí správu doručiť, môže ju opäťovne odoslať niekedy neskôr.

Navrhnutý protokol nie je závisly na žiadnych špeciálnych uzloch typu gateway, prípadne riadiacich uzloch. Každý uzol v sieti môže byť primitívnym uzlom, ktorý bude prijímať a preposielať dátu. Vďaka tomu môžu byť uzly realizované prostredníctvom lacných a malých zariadení.

Protokol bude možné používať na rôznych platformách. V tejto práci vznikne implementácia protokolu pre mikrokontroléry používajúce programovací jazyk C++ a mikrokontroléry alebo mikropočítače podporujúce microPython/Python.

Kapitola 5

TODO Vlastna implementacia

5.1 Navrh packetu

header, vysvetlenie jednotlivych poli, crc, hoplimit,

5.1.1 Typy sprav XXXX

popis typov sprav textova sprava, sensor data sprava, ping sprava, ACK sprava

5.2 Funkcionalita protokolu

dlhy popis funkcionality protokolu, ako funguje, priebeh odoslania spravy, priebeh sirenia spravy cez ostatne node, vysvetlenie statov sprav (sent, rebroadcasted, acked atd) vysvetlenie message queue, timeouty, retry atd alternativne scenare, napr final node mimo dosahu, zla topologia siete atd

5.3 Navrh a implementacia web gui

popis co bude vo webgui, popis funkci, na rpi bude aj monitorovacia zalozka mozno poriesit cachovanie dat v localstoragi – to asi prinesie viac problemov ako vydoh... popis implementacie, kedze webgui bude aj na slabych ttgo zariadeniach, bude treba dosta optimalizovat kod webu

5.4 implementacia na micropython

jak prebiehala implementacia, ukazky kodu atd.

5.5 implementacia na python

rozsirenie a uprava implementacie na micropythonupython, ukazky kodu atd. ak bude cas rozbehat na rpi grafanu na zobrazenie monitorovacich dat

5.6 implementacia c++

jak prebiehala implementacia, ukazky kodu atd. porovnanie voci micro/pythonu + rozdiely, problemy ktore sa vyskytli, ako sa to riesilo, mozno porovnanie rychlosti kodu ? poriesit kde a akobude ulozeny kod webu... dlhy string ??

5.7 Testovanie funkcnosti + medziplatformova komunikacia

test ze to funguje, rozmiestnenie zariadeni, posielanie sprav, simulovanie sensorovych uzlov atd

Kapitola 6

Testovanie výkonnosti + test voci existujúcim protokolom?

Test bude možný asi len voci meshtasticu a to len na ďalšie zariadenia. Tie majú len dve takže to bude slabý test meshtastic ponuka simulator kde by mohlo byť možné zmeniť kód a simuloval vnom moj protokol V meshtasticu nebude fungovať ďaleko siet ak bude mať tvar V, moj protokol to ale má vyriesené takto spomenuté meranie casu doručenia spravy pri rovnakých podmienkach, parametroch atď

Ak bude čas rozšíriť armachat o moj protokol. Upraviť armachat štítcode bude ale časovo náročné a vo výsledku pride armachat o možnosť použiť ho so meshtasticom.

Literatura

1. K. HESTER, et al. *Meshtastic—An Opensource Hiking, Pilot, Skiing, Secure GPS Mesh Communicator, 2020*. Dostupné tiež z: <https://meshtastic.org/>.
2. JOAN MIQUEL SOLÉ, et al. *LoRaMesher - implementing a distance-vector routing protocol for communicating messages among LoRa nodes*. Dostupné tiež z: <https://github.com/LoRaMesher/LoRaMesher>.
3. *Sigfox - První celorepublikový mobilní operátor pro internet věcí* [online] [cit. 2022-07-10]. Dostupné z : <https://sigfox.cz/>.
4. *LoRa alliance* [online] [cit. 2022-07-10]. Dostupné z : <https://lora-alliance.org/>.
5. PIETROSEMOLI, Ermanno. *LoRa details*. Dostupné tiež z: http://wireless.ictp.it/school_2017/Slides/LoRaDetails.pdf.
6. Český Telekomunikační Úřad. Dostupné tiež z: <https://www.ctu.cz/>.
7. *všeobecné oprávnění č. VO-R/10/07.2021-8 k využívání rádiových kmitočtů a k provozování zařízení krátkého dosahu*. Dostupné tiež z: <https://www.ctu.cz/sites/default/files/obsah/vo-r10-072021-8.pdf>.
8. *Semtech* [online] [cit. 2022-07-10]. Dostupné z : <https://www.semtech.com/>.
9. *HopeRF* [online] [cit. 2022-07-10]. Dostupné z : <https://www.hoperf.com/>.
10. *LILYGO*. Dostupné tiež z: <http://www.lilygo.cn/>.
11. MISENKO, Peter. *Armachat - Doomsday LORA QWERTY communicator*. Dostupné tiež z: <https://github.com/bobricius/armachat>.
12. *Pycom, Pymesh—LoRa Full-Mesh Network Technology*. Dostupné tiež z: <https://docs.pycom.io/pymesh>.
13. EBI, Christian; SCHALTEGGER, Fabian; RÜST, Andreas; BLUMENSAAT, Frank. Synchronous LoRa Mesh Network to Monitor Processes in Underground Infrastructure. *IEEE Access*. 2019, roč. 7, s. 57663–57677. Dostupné z DOI: [10.1109/ACCESS.2019.2913985](https://doi.org/10.1109/ACCESS.2019.2913985).