

Міністерство освіти і науки України
Національний технічний університет
«Дніпровська політехніка»

Інститут електроенергетики

(інститут)

Факультет інформаційних технологій

(факультет)

Кафедра Програмного забезпечення комп'ютерних систем

(повна назва)

ПОЯСНЮВАЛЬНА ЗАПИСКА
кваліфікаційної роботи ступеня

магістра

(назва освітньо-кваліфікаційного рівня)

студентки *Дмитровської Мар'яни Михайлівни*

(ПІБ)

академічної групи *122М-19-1*

(шифр)

спеціальності *122 Комп'ютерні науки*

(код і назва спеціальності)

на тему: *Методи, алгоритми та інформаційна технологія покращення*

процесів передачі даних радіосигналами

М.М.Дмитровська

Керівники	Прізвище, ініціали	Оцінка за шкалою		Підпис
		рейтинг овою	інституці йною	
розділів кваліфікаційної роботи				
спеціальний	Проф. Алексєєв М.О.			
економічний	Доц. Касьяненко Л.В.			

Рецензент				
-----------	--	--	--	--

Нормоконтролер	Доц. Сироткіна О.І.			
----------------	---------------------	--	--	--

Дніпро
2020

Міністерство освіти і науки України
Національний технічний університет
«Дніпровська політехніка»

ЗАТВЕРДЖЕНО:

Завідувач кафедри

Програмного забезпечення комп'ютерних систем

(повна назва)

І.М. Удовик

(підпис)

(прізвище, ініціали)

« »

2020 Року

ЗАВДАННЯ

на виконання кваліфікаційної роботи магістра

спеціальності _____ *122 Комп'ютерні науки*
 (код і назва спеціальності)

студентці _____ *Дмитровській М.М.*
 (група) (прізвище та ініціали)

Тема кваліфікаційної роботи _____
Методи, алгоритми та інформаційна технологія покращення процесів передачі даних радіосигналами

1 ПІДСТАВИ ДЛЯ ПРОВЕДЕННЯ РОБОТИ

Наказ ректора НТУ «Дніпровська політехніка» від 22.10.2020 р. № 888-с

2 МЕТА ТА ВИХІДНІ ДАНІ ДЛЯ ПРОВЕДЕННЯ РОБІТ

Об'єкт досліджень – процеси збору та передачі даних з магнітних та температурних сенсорів до користувача

Предмет досліджень – методи передачі радіосигналів за допомогою технології LoRa та мови програмування PHP на базі фреймворків Slim та Angular

Мета роботи – підвищення швидкості передачі та обробки даних з магнітних та температурних сенсорів за допомогою технології LoRa

3 ОЧІКУВАНІ НАУКОВІ РЕЗУЛЬТАТИ

Наукова новизна отриманих результатів кваліфікаційної роботи визначається тим, що було досліджено та удосконалено метод підвищення швидкості передачі радіосигналів за допомогою математичних методів головної

концепції технології LoRa та мови програмування PHP на базі фреймворків Angular та Slim.

Практична цінність результатів полягає в тому, що запропонована в роботі метод може бути використана в напрямку підвищення ефективності роботи різних систем, функціонування яких потребує використання технології передачі даних на великі відстані.

4 ЕТАПИ ВИКОНАННЯ РОБІТ

Найменування етапів робіт	Строки виконання робіт (початок – кінець)
Збір інформації для дослідження предметної області	04.09.2020 – 02.10.2020
Дослідження методів для вирішення поставленого завдання	03.10.2020 – 04.11.2020
Експериментальні дослідження	05.11.2020 – 23.11.2020
Економічна частина	24.11.2020 – 30.11.2018

Завдання видав

_____ (підпис)

Алексєєв М.О.

_____ (прізвище, ініціали)

Завдання прийняла до виконання

_____ (підпис)

Дмитровска М.М..

_____ (прізвище, ініціали)

Дата видачі завдання: 03.09.2020 р.

Термін подання кваліфікаційної роботи до ЕК 03.12.2020 р.

РЕФЕРАТ

Пояснювальна записка: 75 стор., 27 рис., 10 таблиць, 3 додатка, 44 джерел.

Об'єкт досліджень: процеси збору та передачі даних з магнітних та температурних сенсорів до користувача.

Предмет досліджень: методи передачі радіосигналів за допомогою технології LoRa та мови програмування PHP на базі фреймворків Slim та Angular.

Мета магістерської роботи: вивчення та удосконалення існуючих методів підвищення швидкості передачі та обробки даних з магнітних та температурних сенсорів за допомогою технології LoRa.

Методи дослідження: при вирішенні поставлених завдань виконано аналіз і наукове узагальнення літературних джерел по вихідним посилам досліджень.

Наукова новизна: досліджено та удосконалено метод підвищення швидкості передачі радіосигналів за допомогою технології LoRa та мови програмування PHP на базі фреймворків Angular та Slim.

Практична цінність результатів полягає в тому, що запропонований в роботі метод може бути використаний в напрямку підвищення ефективності роботи різних систем, функціонування яких потребує використання технології передачі даних на великі відстані.

У розділі «Економіка» проведені розрахунки трудомісткості розробки програмного забезпечення, витрат на створення ПЗ і тривалості його розробки, а також проведені маркетингові дослідження ринку збуту створеного програмного продукту.

Список ключових слів: інформаційний сигнал, бездротова мережа, радіохвилі, сенсори, пакетування даних, фільтрування даних, технологія LoRa, веб-додаток, Angular.

ABSTRACT

Explanatory note: 75 pages, 27 figures, 10 tables, 3 applications, 44 sources.

Objects of research: processes of data collection and transmission from magnetic and temperature sensors to the user using PHP programming language on base of Slim and Angular Frameworks.

Subject of research: methods of radio transmission using LoRa technology.

Purpose of the Master's thesis: research and improvement of raise of data transmission rate and processing from magnetic and temperature sensors using LoRa technology.

Research methods: in the solution of the tasks set, analysis and scientific generalization of the literature on the initial premises of the research were carried out.

Originality of research consists of research and improvement of the process of radio signal transmission using LoRa technology.

Practical value of the results consists of that proposed method can be used to improve the efficiency of various systems, the operation of which requires the use of long-distance data transmission technology.

In Economics section the calculations of the complexity of software development, the cost of creating software and the duration of its development, as well as marketing studies of the market for the created software product were calculated.

List of keywords: information signal, wireless network, radio waves, sensors, data packaging, data whitening, LoRa technology, web-application, Angular.

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

БД – база даних;

КС – комп'ютерна система;

AES – Advanced Encryption Standard;

HTML – HyperText Markup Language;

LFSR – Linear Feedback Shift Register;

LoRa – Long Range;

LPWAN – Low Power Wide Area Network;

MVC – Model View Controller;

VSAT – Very Small Aperture Terminal;

WNS – Speediwing World Network.

ЗМІСТ

ВСТУП.....	9
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА	
ЗАВДАННЯ.....	12
1.1. Поняття бездротових сенсорних мереж.....	12
1.2. Поняття енергоефективної мережі далекого радіусу дії.....	13
1.3. Технологія модуляції LoRa.....	16
1.4. Апаратний маршрутизатор LoRa	17
1.5. Структура мережі LoRaWan.....	18
1.6. Кінцевий пристрій з підтримкою LoRaWan	20
1.7. Безпека елементів мережі LoRaWan	21
1.8. Висновки до першого розділу	22
РОЗДІЛ 2. ІНФОРМАЦІЙНІ СИСТЕМИ В ПРОЦЕСАХ ЗБОРУ ТА ПЕРЕДАЧІ	
ДАНИХ РАДІОХВИЛЯМИ	23
2.1. Впровадження програмних засобів в технологію LoRa	23
2.2. Поняття принципу пакетування даних	25
2.3. Структура побайтового фільтрування даних	26
2.4. Впровадження програмного забезпечення в алгоритм побайтового фільтрування даних.....	28
2.5. Структура побітового фільтрування даних.....	29
2.6. Впровадження програмного забезпечення в алгоритм побітового фільтрування даних	32
2.7. Інтеграція інформаційної системи.....	34
2.8. Висновки другого розділу	36
РОЗДІЛ 3. ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ ТА ПРАКТИЧНЕ	
ЗАСТОСУВАННЯ	37
3.1. Вступ у дослідження	37

3.2. Принципи розширення спектру	38
3.3. Маніпуляція плином частоти сигналу	41
3.4. Механізм поширення багатопроменевих джерел та співіснування мережі ..	43
3.5. Дослідження рівня покращення ємності приймача інформаційного сигналу	47
3.6. Порівняння індикатору сили сигналу	48
3.7. Опис веб-застосунку користувача	50
3.7. Висновки третього розділу	53
РОЗДІЛ 4. ЕКОНОМІЧНИЙ РОЗДІЛ	55
4.1. Визначення трудомісткості та вартості розробки програмного забезпечення	55
4.2. Витрати на створення програмного забезпечення	59
4.3. Маркетингові дослідження ринку збуту програмного продукту	61
4.4. Оцінка економічної ефективності впровадження розробленого програмного забезпечення	62
ВИСНОВКИ	66
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	67
ДОДАТОК А. КОД ПРОГРАМИ	69
ДОДАТОК Б. ВІДГУК КЕРІВНИКА ЕКОНОМІЧНОГО РОЗДІЛУ	73
ДОДАТОК В. ПЕРЕЛІК ДОКУМЕНТІВ НА ОПТИЧНОМУ НОСІЇ	74

ВСТУП

Актуальність роботи. В рамках кваліфікаційної роботи магістра було розглянуто методи передачі радіосигналів за допомогою технології LoRa, так як вони мають ряд значних переваг:

- передача даних на великі відстані (вище 10 км);
- метод ґрунтується на технології модуляції з розширеним спектром і варіації лінійної частотної модуляції з інтегрованою прямою корекцією помилок;
- значно підвищує чутливість приймача і, аналогічно іншим методам модуляції з розширеним спектром, використовує всю ширину смуги пропускання каналу для передачі сигналу, що робить його стійким до каналним шумів і нечутливим до зсувів, викликаних неточностями в налаштуванні частот при використанні недорогих опорних кварцових резонаторів;
- забезпечує низьке споживання енергії, що, у свою чергу, подовжує період роботи батареї або акумуляторів сенсорів;
- збільшена швидкість передачі даних (здійснюється за допомогою алгоритму адаптивної швидкості передачі даних) у порівнянні з відомими аналогами;
- можливість зменшення мережних перешкод.

LoRa використовує запатентовану модуляцію розширеного спектра, подібну до модуляції розширеного спектру (CSS) і похідну від неї. Модуляція LoRa з розширеним спектром виконується шляхом представлення кожного біта інформації корисного навантаження за допомогою декількох інформаційних повідомлень. Швидкість, з якою надсилається інформація про розповсюдження, називається швидкістю передачі символів, співвідношення між номінальною швидкістю передачі символів та частотою зміни потоку є коефіцієнтом розповсюдження (SF) і представляє кількість символів, відправлених на біт інформації. LoRa може

обмінюватися швидкість передачі даних для чутливості з фіксованою пропускнуою здатністю каналу, вибираючи обсяг використовуваного розповсюдження (вибір радіопараметру від 7 до 12). Нижче значення SF означає більше звукових сигналів за секунду, отже, можливість закодувати більше даних в секунду. Більш високий рівень SF передбачає меншу кількість зміни потоку в секунду; отже, даних для кодування в секунду менше. Порівняно з меншим SF, надсилання однакової кількості даних із більшим SF потребує більше часу передачі, відомого як ефірний час. Більша кількість ефірного часу означає, що модем працює довше і споживає більше енергії. Перевагою високого SF є те, що більш тривалий ефірний час надає приймачу більше можливостей для вибірки потужності сигналу, що призводить до кращої чутливості.

Крім того, LoRa використовує кодування виправлення мережевих перешкод, щоб поліпшити стійкість передачі інформації.

Мета магістерської роботи: дослідження та удосконалення методики підвищення швидкості передачі та обробки даних з магнітних та температурних сенсорів за допомогою технології LoRa.

Об'єкт дослідження: процеси передачі радіосигналів за допомогою технології LoRa.

Предмет дослідження: методи збору та передачі даних з магнітних та температурних сенсорів до користувача за допомогою мови програмування PHP на базі фреймворків Angular та Slim.

Методи дослідження: при вирішенні поставлених завдань виконано аналіз і наукове узагальнення літературних джерел по вихідним посилам досліджень.

Наукова новизна: досліджено та удосконалено метод підвищення якості та швидкості передачі радіосигналів за допомогою технології LoRa та мови програмування PHP на базі фреймворків Angular та Slim.

Практичне значення: розроблено метод, який може бути використана в напрямку підвищення ефективності роботи різних систем, функціонування яких потребує використання технології передачі даних на великі відстані.

Особистий внесок автора полягає в розробці теоретичної частини магістерської роботи, в дослідженні і систематизації знання про існуючі методики, розробці методів досліджень і технологій реалізації, в оцінці отриманих результатів.

Структура та обсяг кваліфікаційної роботи. Робота складається з вступу, чотирьох розділів і висновків. Містить 75 сторінки друкованого тексту, в тому числі 66 сторінок тексту основної частини з 27 рисунками, списку використаних джерел з 44 найменуваннями на 3 сторінках, 3 додатків на 7 сторінках.

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ І ПОСТАНОВКА ЗАВДАННЯ

1.1. Поняття бездротових сенсорних мереж

Бездротовий зв'язок - це електромагнітна передача інформації між двома або більше точками, які пов'язані електричним провідником. Найбільш поширені бездротові технології використовують радіохвилі. Він охоплює різні типи фіксованих, мобільних і портативних додатків, включаючи двосторонню радіозв'язок, стільникові телефони, персональні комп'ютери (ПК) і бездротові мережі.

Мережа бездротових датчиків (WSN) відноситься до групи просторово розподілених та виділених датчиків для моніторингу та реєстрації фізичних умов навколишнього середовища та організації зібраних даних у центральному місці. WSN вимірюють умови навколишнього середовища, такі як температура, звук, рівень забруднення, вологість, вітер тощо [1].

Вони схожі на бездротові спеціальні мережі в тому сенсі, що вони покладаються на бездротове підключення та спонтанне формування мереж, щоб дані датчиків могли передаватися бездротовим способом. WSN - це просторово розподілені автономні датчики для моніторингу фізичних або екологічних умов, таких як температура, звук, тиск тощо, та спільної передачі своїх даних через мережу до основного місця. Більш сучасні мережі є двосторонньо спрямованими, збирають дані з розподілених датчиків [2], і дозволяють контролювати активність датчиків. [3] Сьогодні такі мережі використовуються в багатьох промислових та споживчих програмах, таких як моніторинг та контроль промислових процесів, моніторинг стану машин тощо.

Основні характеристики WSN включають:

- Обмеження енергоспоживання для вузлів, що використовують батареї або збирання енергії.
- Здатність справлятися з несправностями вузлів (стійкість)
- Мобільність вузлів, однорідність неоднорідність
- Масштабованість до великих масштабів розгортання
- Здатність протистояти суворим екологічним умовам
- Простота використання
- Оптимізація між шарами

1.2. Поняття енергоефективної мережі далекого радіусу дії

Енергоефективна мережа далекого радіусу дії (LPWAN) або широкополосна мережа низької потужності (LPWA) або мережа низького енергоспоживання (LPN) - це тип бездротової телекомунікаційної широкосмугової мережі, призначений для забезпечення зв'язку на великих діапазонах при низькому біті швидкість серед підключених предметів, таких як датчики, що працюють від акумулятора. [4] Низька потужність, низька швидкість передачі даних та передбачуване використання відрізняють цей тип мережі від бездротової глобальної мережі, призначеної для підключення користувачів та передачі більшої кількості даних, використовуючи більше енергії. Швидкість передачі даних LPWAN коливається від 0,3 кбіт / с до 50 кбіт / с на канал. [5]

На фізичному рівні принципу передачі даних за технологією LPWAN лежить властивість радіосистем - збільшення енергетики, а значить і дальності зв'язку при зменшенні швидкості передачі. Чим нижче бітова швидкість передачі, тим більше енергії вкладається в кожен біт і тим легше виділити його на тлі мережевих

перешкод в приймальні частини системи. Таким чином, низька швидкість передачі даних дозволяє домогтися більшої дальності їх прийому.

Пристрій або модем з LPWAN-модулем передає дані по радіоканалу на базову станцію. Станція приймає сигнали від всіх пристроїв в радіусі своєї дії, оцифровує і передає на віддалений сервер, використовуючи доступний канал зв'язку: Ethernet, стільниковий зв'язок, VSAT.

Отримані на сервері дані використовуються для відображення, аналізу, побудови звітів та прийняття рішень.

Управління пристроями, оновлення програмного забезпечення відбувається з використанням зворотного каналу зв'язку.

Для передачі даних по радіоканалу, як правило, застосовується неліцензованому спектр частот, дозволених до вільного використання в регіоні побудови мережі: 2,4 ГГц, 868/915 МГц, 433 МГц, 169 МГц [6].

Переваги LPWAN:

- Велика дальність передачі сигналу у порівнянні з іншими бездротовими технологіями, використовуваними для телеметрії, що досягає 10-15 км [7].
- Низький рівень споживання енергії у кінцевих пристроїв, завдяки мінімальним витратам енергії на передачу невеликого пакета даних [8].
- Висока проникаюча здатність радіосигналу в міській забудові при використанні частот субгігагерцового діапазону.
- Висока масштабованість мережі на великих територіях.
- Відсутність необхідності отримання частотного дозволу та плати за радіочастотний спектр, внаслідок використання неліцензованому частот (ISM band) [9]

Недоліки LPWAN:

- Відносно низька пропускна здатність, внаслідок використання низької частоти радіоканалу. Варіюється в залежності від використовуваної технології передачі даних на фізичному рівні, становить від кількох сотень біт / с до декількох десятків Кбіт / с [8].
 - Затримка передачі даних від датчика до кінцевого додатку, пов'язана з часом передачі радіосигналу, може досягати від декількох секунд до декількох десятків секунд.
 - Відсутність єдиного стандарту, який визначає фізичний шар і управління доступом до середовища для бездротових LPWAN-мереж.
- Наступний рисунок демонструє топологію бездротової LPWAN-мережі:

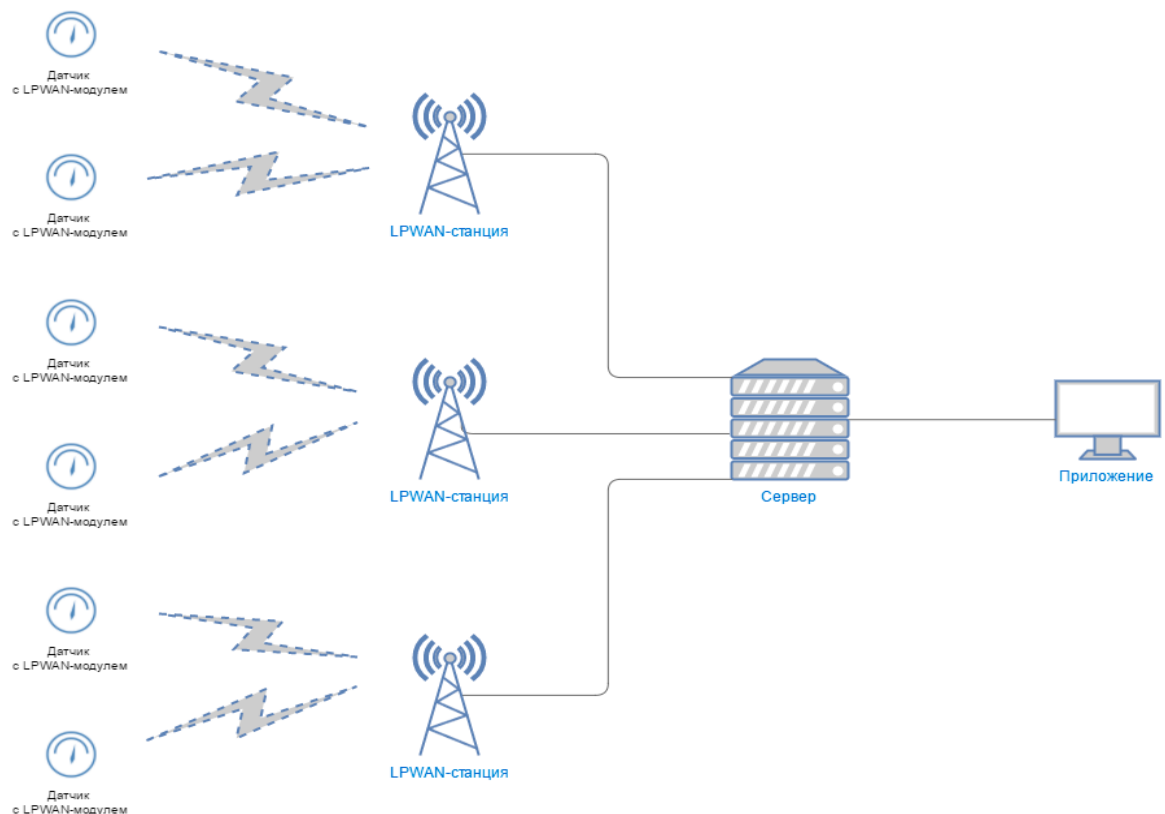


Рис.1.2. Топологія бездротової LPWAN-мережі

1.3. Технологія модуляції LoRa

Технологія модуляції LoRa (Long Range) являє собою метод модуляції, який забезпечує значно більшу дальність зв'язку (зону покриття), ніж інші конкуруючі з ним способи. Метод ґрунтується на технології модуляції з розширеним спектром і варіації лінійної частотної модуляції (Chirp Spread Spectrum, CSS) з інтегрованою прямою корекцією помилок (Forward Error Correction, FEC). Технологія LoRa значно підвищує чутливість приймача і, аналогічно іншим методам модуляції з розширеним спектром, використовує всю ширину смуги пропускання каналу для передачі сигналу, що робить його стійким до канальним шумів і нечутливим до зсувів, викликаних неточностями в налаштуванні частот при використанні недорогих опорних кварцових резонаторів. Технологія LoRa дозволяє здійснювати демодуляцію сигналів з рівнями на 19,5 дБ нижче рівня шумів, притому що для правильної демодуляції більшості систем з частотної маніпуляцією (Frequency Shift Keying, FSK) потрібна потужність сигналу як мінімум на 8-10 дБ вище рівня шуму. Модуляція LoRa визначає той фізичний рівень¹ (Physical Layer, PHY, іноді його називають шар), який може бути використаний з різними протоколами і в різних варіантах мережевий архітектури, таких як сітка (Mesh), зірка (Star), точка-до-якій точці (point-to-point) і т. п.

Модуляція LoRa є, фізичним рівнем, а LoRaWAN (Long Range Wide-Area Networks, LoRaWAN) це MAC-протокол для високоємкий мереж з великим радіусом дії і низьким власним споживанням потужності, який організація LoRa Alliance стандартизувала для малопотужних глобальних радіальних мереж (Low Power Wide Area Networks, LPWAN) типу зірка. Протокол LoRaWAN оптимізований для малобюджетних сенсорів з роботою від батарей і включає в себе різні класи вузлів, забезпечуючи компроміс між швидкістю доставки інформації і часом роботи пристроїв при використанні живлення від батарей / акумуляторів. Протокол забезпечує повну двосторонню зв'язок, а архітектура (за допомогою

спеціальних методів шифрування) забезпечує загальну надійність і безпеку всієї системи. Архітектура LoRaWAN також була розроблена з метою полегшити виявлення мобільних об'єктів для відстеження активів підприємств, що є одним з найбільш швидко зростаючих додатків на рівні Інтернету речей (Internet of Things, IoT). Протокол LoRaWAN розробляється для використання в загальнонаціональних мережах великих операторів зв'язку. З цією метою організація LoRa Alliance стандартизує свій протокол LoRaWAN з урахуванням сумісності і взаємодії з усіма основними світовими операторами зв'язку.

1.4. Апаратний маршрутизатор LoRa

Апаратні маршрутизатори LoRa призначені для використання в радіальних зіркоподібних мережних архітектурах великого радіусу дії, вони використовуються в системі LoRaWAN. Через властивостей технології LoRa ці шлюзи можуть являти собою багатоканальні мультімодемні трансивери, які здатні виконувати демодуляцію на декількох каналах одночасно, і навіть одночасну демодуляцію безлічі сигналів на одному і тому ж каналі. Ці шлюзи використовують інші радіочастотні компоненти, ніж ті, які застосовуються в кінцевій точці задля забезпечення високої потужності випромінювання безпосередньо радіосигналу. Шлюзи служать в якості інтерфейсу у вигляді прозорого моста для передачі повідомлень між кінцевими пристроями і центральним сервером мережі.

Дані маршрутизатори підключаються до мережевого сервера через стандартні IP-з'єднання, а кінцеві пристрої використовують однокачкову бездротовий зв'язок до одного або кількох шлюзів. Всі кінцеві точки зв'язку, як правило, є двонаправленими, але вони також підтримують функціонування в режимі, що забезпечує можливість здійснення групового оновлення програмного забезпечення через стільникову мережу або передачу інших масових повідомлень, що дозволяє скоротити активний час на їх передачу. Залежно від бажаної їх каналної ємності та

місць встановлення доступні різні версії шлюзів, які можуть встановлюватися всередині приміщень (Рис. 1.4.).



Рис 1.4. Зовнішній вигляд пристроїв

1.5. Структура мережі LoRaWan

Пакети з інформацією, що використовують різні фактори розповсюдження, є ортогональними, що означає, що вони невидимі один для одного: як уже згадувалося раніше, вони просто здаються шумом один для одного. Отже, два пакети, які одночасно надходять на один і той же канал прийому з різними факторами розповсюдження, не зіткнуться, і обидва вони будуть демодульовані чіпом модемного шлюзу. Однак два пакети з однаковим коефіцієнтом розповсюдження, що надходять одночасно по одному каналу, можуть призвести до зіткнення.

Ємність мережі LoRaWAN є функцією щільності її шлюзу. Щоб максимізувати пропускну здатність мережі, важливо використовувати механізм адаптивної швидкості передачі даних (ADR). Головною метою ADR є економія заряду акумулятора кінцевих вузлів LoRaWAN. Завдяки тому, що кінцеві вузли, найближчі до шлюзу, передають з використанням найнижчого коефіцієнта розповсюдження, їх час у повітрі зводиться до мінімуму, тим самим продовжуючи термін служби батареї. Більш віддалені датчики передають з вищим коефіцієнтом розповсюдження. Компроміс проводиться між зарядом акумулятора та відстанню, враховуючи, що більш високий коефіцієнт розповсюдження дозволяє шлюзу підключатися до віддалених пристроїв.

LoRa є фізичним (PHY) рівнем, тобто бездротовою модуляцією, що використовується для створення зв'язку далекого зв'язку. LoRaWAN - це відкритий мережевий протокол, який забезпечує безпечний двонаправлений зв'язок, мобільність та послуги локалізації, стандартизовані та підтримувані Альянсом LoRa, яка має наступну архітектуру:

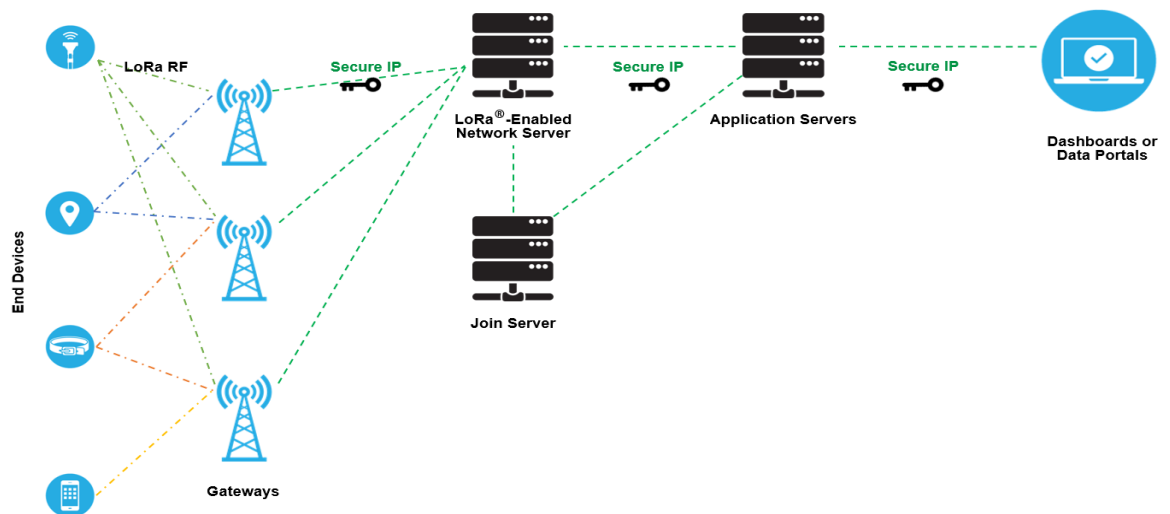


Рис 1.5. Схема мережевої архітектури.

1.6. Кінцевий пристрій із підтримкою LoRaWan

Кінцевий пристрій із підтримкою LoRaWAN - це датчик або виконавчий механізм, який бездротово підключений до мережі LoRaWAN через радіошлюзи за допомогою RF-модуляції LoRa.

У більшості програм кінцевий пристрій - це автономний датчик, який часто працює від батареї, який оцифровує фізичні умови та екологічні явища. Типові випадки використання приводу включають: вуличне освітлення, бездротові замки, перекриття водяного клапана, запобігання витоку, серед іншого.

Коли вони виготовляються, пристроям на основі LoRa присвоюється кілька унікальних ідентифікаторів. Ці ідентифікатори використовуються для безпечної активації та адміністрування пристрою, забезпечення безпечного транспортування пакетів через приватну або загальнодоступну мережу та доставки зашифрованих даних у Хмару (Cloud).

Усі отримані дані налаштовані на передачу користувачеві на кінцеві прилади мережею, яка має наступну архітектуру:

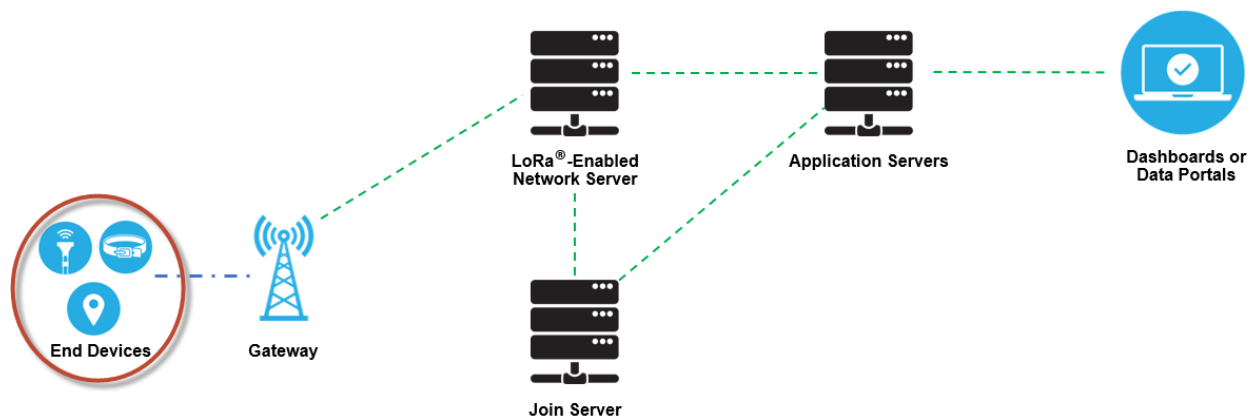


Рис.1.6. Архітектура мережі кінцевого пристрою

1.7. Безпека елементів мережі LoRaWAN

Існують два ключові елементи безпеки мережі LoRaWAN: процедура приєднання та автентифікація повідомлень. Процедура приєднання встановлює взаємну автентифікацію між кінцевим пристроєм та мережею LoRaWAN, до якої він підключений. До мережі можуть приєднуватися лише авторизовані пристрої. LoRaWAN MAC і повідомлення додатків мають автентифікацію джерела, захищені цілісністю та шифруються наскрізно (тобто від кінцевого пристрою до сервера додатків і навпаки).

Ці функції безпеки гарантують, що:

- Мережний трафік не змінено
- До мережі LoRaWAN підключені лише законні пристрої
- Неможливо прослуховувати мережевий трафік (без підслуховування)
- Мережевий трафік неможливо захопити та відтворити
- З цією основою ми розглянемо заходи безпеки LoRaWAN більш докладно.

Спочатку, окремі кореневі ключі надійно зберігаються на кінцевих пристроях, а відповідні ключі надійно зберігаються на сервері приєднання. Після чого кінцевий пристрій надсилає повідомлення із запитом на приєднання до сервера. Після того, як сервер об'єднання аутентифікує пристрій, що вимагає приєднання до мережі, він повертає на пристрій повідомлення про прийняття з'єднання. Керування трафіком між кінцевим пристроєм та мережевим сервером забезпечується 128-бітовим ключем сеансу AES мережі (NwkSKey). Трафік даних, що переміщується між кінцевим пристроєм та сервером додатків, захищений 128-розрядним ключем сеансу додатків (AppSKey). Цей метод гарантує, що ні шлюз, ні мережевий сервер не можуть прочитати дані користувача.

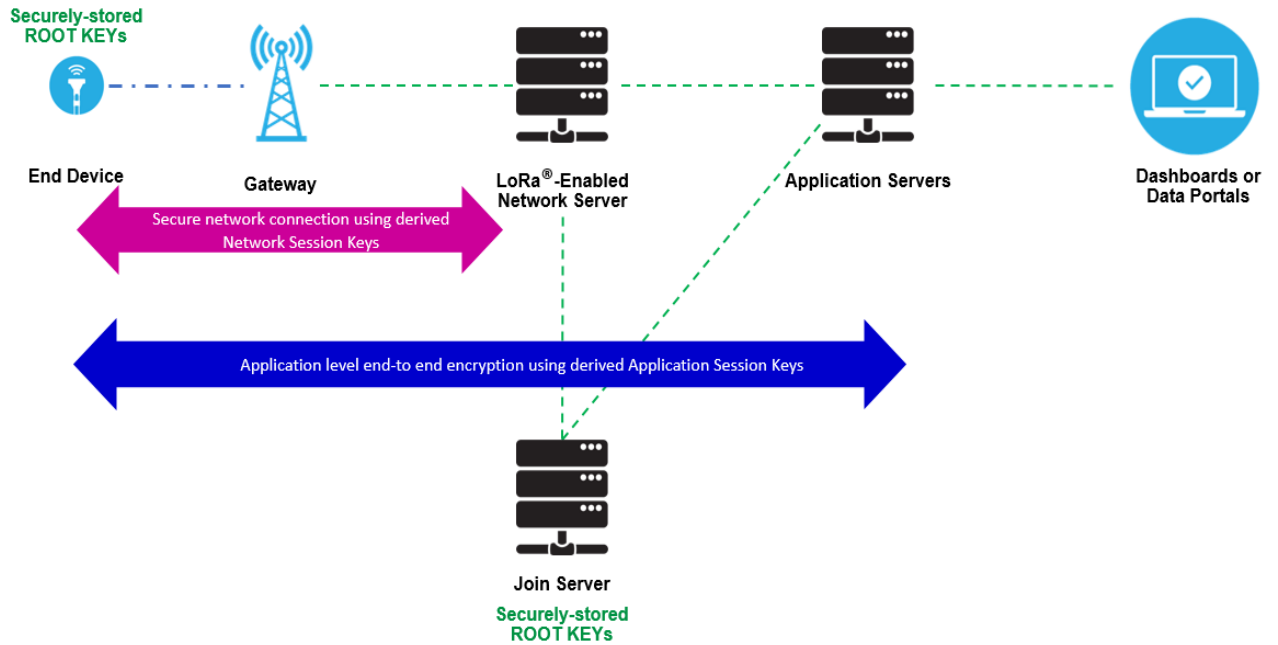


Рис 1.7. Безпечне передання пакетів інформації.

1.8. Висновки до першого розділу

У ході виконання першого розділу було виконано вступ у теоретичну частину дослідження з її подальшим розгорненням, виконано постанову завдання, описано актуальність роботи, наведено загальну інформацію з предметної області, деталізовано опис апаратної частини майбутнього програмного забезпечення. Описано принцип роботи бездротових мереж, їх переваги та недоліки, важливість у сучасному світі інформаційних технологій. У загальних поняттях наведено принципи роботи технології LoRa, на використанні її алгоритмів розгортається увесь подальший хід виконання кваліфікаційної роботи.

РОЗДІЛ 2

ВПРОВАДЖЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ В ПРОЦЕСИ ЗБОРУ ТА ПЕРЕДАЧІ ДАНИХ РАДІОХВИЛЯМИ

2.1. Впровадження програмних засобів в технологію LoRa

Важливе місце у наданні змоги передання даних від технічних засобів (сенсорів) до користувача займає правильний підбір програмних засобів та їх функціоналів для вирішення поставлених задач. У зв'язку з чим, в процесі виконання завдання кваліфікаційної роботи магістра було використано наступні програмні засоби: мова програмування PHP на базі фреймворків Angular та Slim, платформа для розробки API Postman, інструмент для візуального проектування баз даних MySQL Workbench 8.0., Apache веб-сервер 2.4.46., мова програмування C як база для написання технічних алгоритмів, HTML та CSS як допоміжні інструменти для створення інтерфейсу користувача.

Angular – це динамічний front-end веб-фреймворк з відкритим вихідним кодом, призначений для створення веб-застосунків. Його ціллю є розширення браузерних додатків на основі MVC-шаблону, спрощення процесів тестування та розробки. Фреймворк працює з HTML, що містить додаткові атрибути, які описуються директивами, і пов'язує введення або виведення області сторінки з моделлю, яка представляє собою звичайні змінні JavaScript. Значення цих змінних задаються вручну або викликаються зі статичних або динамічних JSON-даних.

Slim – зручний у застосуванні мікро-фреймворк PHP з широким спектром особливостей:

- Швидка та потужна маршрутизація HTTP для розробки SEO-зручних URL-адрес
- Кешування HTTP для поліпшення продуктивності веб-сайту та використання меншої кількості кешу

- Захищені файли cookie для підвищення безпеки системи
- Обробка помилок та налагодження для зручності тестування
- Введення залежності для управління зовнішніми інструментами.

Для проведення розробки та дослідження інформаційної системи для покращення процесів передачі даних, необхідно використовувати інструменти, які дозволяють маніпулювати об'єктами "Запит" та "Відповідь". Ідея проміжного програмного забезпечення фреймворку Slim чудово допомагає з цією метою, так як і з захистом програмних засобів для міжсайтових запитів та їх аутентифікації перед запуском запитів. Створюйте програми за допомогою проміжного програмного забезпечення, налаштовуйте HTTP-запити та об'єкти відповіді навколо програми Slim. Наступний принцип розкриває сутність роботи Slim: він додає проміжне програмне забезпечення як шари навколо основного додатка. Після цього структура розширюється, додаючи нові проміжні шари. Останній доданий шар проміжного програмного забезпечення є першим, що виконується:

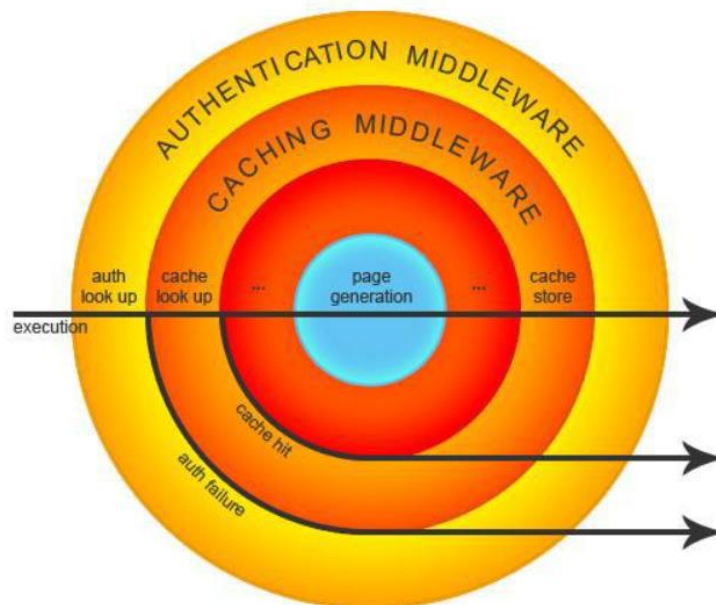


Рис. 2.1. Створення додаткових шарів програмного забезпечення

2.2. Поняття принципу пакетування даних

У радіочастотній системі та в усіх системах зв'язку взагалі дані, що передаються, групуються пакети. Ці пакети часто можуть містити довгі послідовності 1 і 0, що вводить зміщення постійного струму в переданий сигнал. Це зміщення постійного струму виробляє нерівномірний розподіл потужності по зайнятому каналу. Для виправлення цієї проблеми необхідно рандомізувати дані та переконатися, що передані вони не залежні від постійного струму.

Дані без постійного струму можна отримати за допомогою кодування Манчестера, що гарантує відсутність у кодуванні більше ніж двох сигналів 0 або 1. Однак, це зменшує ефективний бітрейт системи, оскільки він подвоює обсяг даних, що підлягає передачі, і, отже, зменшує вдвічі ефективну швидкість передачі даних.

Інший прийом, що називається «відбілюванням» або «скремблюванням», широко використовується для рандомізації даних до користувачів перед початком передачі радіосигналу. Дані вибілюють за допомогою випадкової послідовності на стороні Tx (передатчика), при цьому на стороні Rx (приймача) – відбувається протилежний «відбілюванню» процес, з використанням тієї ж послідовності обробки даних. У процесі «відбілювання» даних часто є обов'язковим наявність контрольної суми в кінці корисного навантаження, що слугує для перевірки достовірності отриманих даних. З цією метою було розроблено два досить подібних алгоритми для виконання поставленої мети. Вони розроблені з метою реверсивного «відбілювання» на стороні приймача за допомогою вищезгаданої випадкової послідовності. Це налаштування обмежує кількість послідовних 1 або 0 до 9 повторень. Процес «відбілювання» даних використовується лише тоді, коли дані користувача мають високу кореляцію з довгими рядками 0 і 1.

2.3. Структура побайтового фільтрування даних

Після отримання необхідного пакету даних, стає необхідним використання правильного підходу у процесі фільтрації отриманої інформації. Структура побайтового фільтрування («відбілювання») даних базується на 9-бітному поліномі $x^9 + x^5 + 1$. З цією структурою LSBit (з англ. Least significant bit – найменш важливий (останній) біт [13]) на виході регістру зсуву лінійного зворотного зв'язку XOR (функція XOR є істинною лише тоді, коли лише одне (і лише одне) з введених значень є істинним, а інакше – false [14]) з найбільш значущого біту даних. На початковому етапі кожному триггеру LFSR встановлюється значення "1". Регістр зсуву з лінійним зворотним зв'язком (LFSR) – це регістр зсуву, вхідний біт якого є лінійною функцією попереднього стану [15]. На рисунку нижче представлена структура побайтового фільтрування даних.

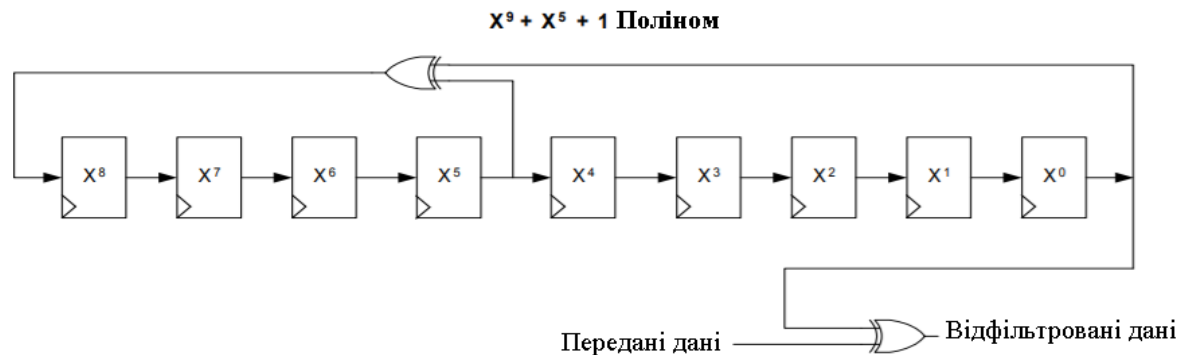


Рис.2.3. Структура побайтового фільтрування даних

Припустимо, що є чотирибайтова корисна навантаження, яку потрібно «відбілити», як показано в таблиці нижче:

Таблиця 2.3

Чотирибайтна корисна навантаження

Номер байту	Дані
0	0x01
1	0x9A
2	0x55
3	0x29

У той же час, можливо передбачення вихідних даних LFSR у міру його зсуву вліво, що наведено у таблиці нижче:

Таблиця 2.3.1

Вихідні значення кожного біту

Номер біту	Значення
1-9	1
10-13	0
14-17	1
18	0
19-21	1
22-25	0
26	1
27	0
28-29	1
30-31	0
32	1

Розподілені на байти, ці значення матимуть наступний вигляд:

Таблиця 2.3.2

Вихідні дані чотирибайтної корисної навантаження

Номер байту	Дані
0	0xFF
1	0x87
2	0xB8
3	0x59

На цьому етапі кожному байту даних з FIFO (в обчислювальній техніці та в теорії систем - це метод організації маніпуляцій зі структурою даних - часто, зокрема, буфером даних, - де спочатку обробляється найперший біт з черги [10]) передається спочатку найбільш значущий біт MSB (у обчисленнях найбільш значущим бітом є бітова позиція у двійковому числі, що має найбільше значення, останній біт у записі [11]) і ці дані, передаючись побітово, переходять до вищезазначеного процесу «відбілювання», що, у свою чергу, створює наступну послідовність вихідних байтів:

Таблиця 2.3.4

Результат поетапного фільтрування даних

Номер байту	Вхідні дані	Вихідні дані	«Відбілені» дані
0	0x01	0xFF	0xFE
1	0x9A	0x87	0x1D
2	0x55	0xB8	0xED
3	0x29	0x59	0x70

2.4. Впровадження програмного забезпечення в алгоритм побайтового фільтрування даних

Минулий етап дослідження досить легко досягає реалізації на кодї мови програмування C. Ідея полягає в тому, щоб просто перенести LFSR для кожного

нового біта даних і XOR. Як можна бачити, вихідні значення XNOR є просто оберненими до відповідних вихідних значень XOR [12]) останнього триггера. Наведений нижче приклад - це проста функція, яка використовує заданий буфер даних і виконує процес «відбілювання» даних у цей же буфер.

```
void SX1232RadioComputeWhitening( uint8_t *buffer, uint16_t bufferSize )
{
    uint8_t i = 0;
    uint16_t j = 0;
    uint8_t WhiteningKeyMSBPrevious = 0;
    uint8_t revertedWhiteningKeyLSB = 0;

    revertedWhiteningKeyLSB = WhiteningKeyLSB;    // WhiteningKeyLSB is 0xFF at init

    for( j = 0; j < bufferSize - 1; j++ )
    {
        buffer[j] ^= revertedWhiteningKeyLSB;

        for( i = 0; i < 8; i++ )
        {
            WhiteningKeyMSBPrevious = WhiteningKeyMSB;
            WhiteningKeyMSB = ( WhiteningKeyLSB & 0x01 ) ^ ( ( WhiteningKeyLSB >> 5 ) & 0x01 );
            WhiteningKeyLSB = ( ( ( WhiteningKeyMSBPrevious << 7 ) & 0x80 | ( WhiteningKeyLSB >> 1 ) & 0xFF ) );
        }
        revertedWhiteningKeyLSB = (WhiteningKeyLSB & 0xF0) >> 4 | (WhiteningKeyLSB & 0x0F) << 4;
        revertedWhiteningKeyLSB = (revertedWhiteningKeyLSB & 0xCC) >> 2 | (revertedWhiteningKeyLSB & 0x33) << 2;
        revertedWhiteningKeyLSB = (revertedWhiteningKeyLSB & 0xAA) >> 1 | (revertedWhiteningKeyLSB & 0x55) << 1;
    }
}
```

Рис.2.4. Код функції побайтового фільтрування даних

2.5. Структура побітового фільтрування даних

На відміну від попередньо описаного алгоритму, алгоритм побітового фільтрування даних базується на роботі з пакетами інформації, розподілених побітово. Ці процеси дуже подібні, але послідовність зберігання результату зовсім інша, тому виникає необхідність правильної реалізації належного алгоритму, що використовується для коректного фільтрування даних для подальшої передачі користувачу. Наступний рисунок розкриває принцип структури алгоритму побітового фільтрування даних:

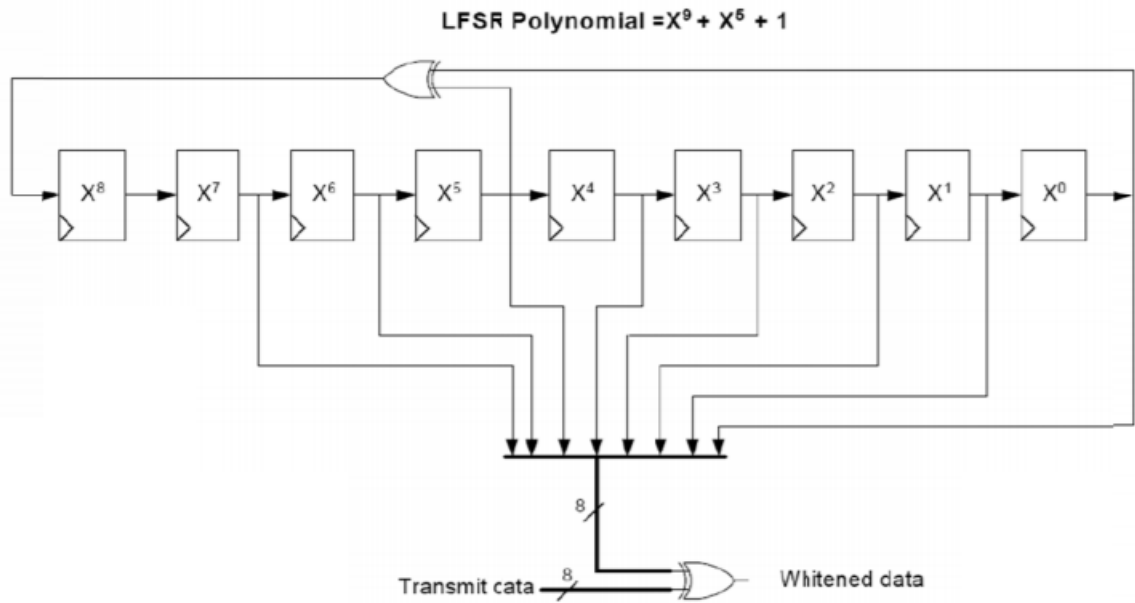


Рис. 2.4. Структура побітового фільтрування даних

Для подальшого розглянення принципу роботи алгоритму, буде розглянутий наступний набір вхідних даних чотирибайтної корисної навантаження:

Таблиця 2.4

Вхідні дані чотирибайтної корисної навантаження

Номер байту	Дані
0	0x01
1	0x9A
2	0x55
3	0x29

У той же час, можливо передбачення вихідних даних LFSR у міру його зсуву вліво, що наведено у таблиці нижче:

Таблиця 2.4.1

Вихідні значення кожного біту

Номер біту	Значення
1-11	1
12-15	0
16	1
17-19	0
20-22	1
23	0
24-25	1
26-27	0
28-29	1
30-31	0
32	1

Розподілені на байти, ці значення матимуть наступний вигляд:

Таблиця 2.4.2

. Вихідні дані чотирибайтної корисної навантаження

Номер байту	Дані
0	0xFF
1	0xE1
2	0x1D
3	0x9A

Після завершення минулих етапів, кожному байту даних передається спочатку найбільш значущий біт MSB і ці дані, передаючись побітово, переходять до вищезазначеного процесу «відбілювання», що, у свою чергу, створює наступну послідовність вихідних байтів:

Таблиця 2.4.3

Результат поетапного фільтрування даних

Номер байту	Вхідні дані	Вихідні дані	«Відбілені» дані
0	0x01	0xFF	0xFE
1	0x9A	0xE1	0x7B
2	0x55	0x1D	0x48
3	0x29	0x85	0xB3

2.6. Впровадження програмного забезпечення в алгоритм побітового фільтрування даних

Минулий етап дослідження досить легко досягає реалізації на кодї мови програмування C, що згодом використовується у ходї виконання роботи та розробці проекту і викликається функціями, написаними мовою РНР. Великою перевагою даної програмної реалізації є той факт, що вона симетрична і може використовуватися як для «відбілювання» даних, так і для «реверсивного відбілювання». Представлений нижче рисунок описує функціональну частину коду, у якому використовується функція побайтового фільтрування даних.


```

static uint8_t WhiteningKeyMSB;           // Global variable so the value is kept after starting the
static uint8_t WhiteningKeyLSB;         // de-whitening process

WhiteningKeyMSB = 0x01;                 // Init value for the LFSR, these values should be initialise only
WhiteningKeyLSB = 0xFF;                 // at the start of a whitening or a de-whitening process

// *buffer is a char pointer indicating the data to be whiten / de-whiten
// bufferSize is the number of char to be whiten / de-whiten

// >> The whitened / de-whitened data are directly placed into the pointer

void SX1232RadioComputeWhitening( uint8_t *buffer, uint16_t bufferSize )
{
    uint8_t i = 0;
    uint16_t j = 0;
    uint8_t WhiteningKeyMSBPrevious = 0; // 9th bit of the LFSR

    for( j = 0; j < bufferSize; j++ )    // byte counter
    {
        buffer[j] ^= WhiteningKeyLSB;    // XOR between the data and the whitening key

        for( i = 0; i < 8; i++ )        // 8-bit shift between each byte
        {
            WhiteningKeyMSBPrevious = WhiteningKeyMSB;
            WhiteningKeyMSB = ( WhiteningKeyLSB & 0x01 ) ^ ( ( WhiteningKeyLSB >> 5 ) & 0x01 );
            WhiteningKeyLSB = ( ( WhiteningKeyLSB >> 1 ) & 0xFF ) | ( ( WhiteningKeyMSBPrevious << 7 ) & 0x80 );
        }
    }
}

```

Рис.2.6. Код функції побайтового фільтрування даних

Для коректного функціонування алгоритмів, яке забезпечить надійне передання правильно відфільтрованих даних до користувача, необхідне виконання циклічної перевірки надмірності (CRC), що є кодом виявлення помилок, і зазвичай використовується в цифрових мережах та пристроях зберігання даних для виявлення випадкових змін необроблених даних. Блоки даних, що надходять у ці системи, отримують коротке контрольне значення, додане на основі залишку поліноміального поділу їх вмісту. Під час отримання обчислення повторюється, і якщо контрольні значення не збігаються, можна вжити коригувальних заходів проти пошкодження даних. CRC можна використовувати для виправлення помилок [16].

CRC називаються так, оскільки значення перевірки даних є надмірністю, що розширює повідомлення без додавання інформації, а алгоритм базується на циклічних кодах. CRC популярні, оскільки їх легко впровадити в двійкове обладнання та проаналізувати математично, що особливо добре допомагає виявити

типові помилки, спричинені шумом у каналах передачі. Оскільки контрольне значення має фіксовану довжину, функція, яка його генерує, іноді використовується як хеш-функція. Наступний програмний код розкриває сутність реалізації циклічної перевірки надмірності, яка базується на використанні двох попередніх алгоритмів:

```

// CRC types
#define CRC_TYPE_CCITT          0
#define CRC_TYPE_IBM           1

// Polynomial = X^16 + X^12 + X^5 + 1
#define POLYNOMIAL_CCITT       0x1021
// Polynomial = X^16 + X^15 + X^2 + 1
#define POLYNOMIAL_IBM        0x8005

// Seeds
#define CRC_IBM_SEED           0xFFFF
#define CRC_CCITT_SEED         0x1D0F

uint16_t RadioComputeCRC( uint8_t *buffer, uint8_t length, uint8_t crcType )
{
    uint8_t i = 0;
    uint16_t crc = 0;
    uint16_t polynomial = 0;

    polynomial = ( crcType == CRC_TYPE_IBM ) ? POLYNOMIAL_IBM : POLYNOMIAL_CCITT;
    crc = ( crcType == CRC_TYPE_IBM ) ? CRC_IBM_SEED : CRC_CCITT_SEED;

    for( i = 0; i < length; i++ )
    {
        crc = ComputeCrc( crc, buffer[i], polynomial );
    }

    if( crcType == CRC_TYPE_IBM )
    {
        return crc;
    }
    else
    {
        return( ( uint16_t ) ( ~crc ) );
    }
}

uint16_t ComputeCrc( uint16_t crc, uint8_t dataByte, uint16_t polynomial )
{
    uint8_t i;

    for( i = 0; i < 8; i++ )
    {
        if( ( ( ( crc & 0x8000 ) >> 8 ) ^ ( dataByte & 0x80 ) ) != 0 )
        {
            crc <<= 1;          // shift left once
            crc ^= polynomial;  // XOR with polynomial
        }
        else
        {
            crc <<= 1;          // shift left once
        }
        dataByte <<= 1;        // Next data bit
    }
    return crc;
}

```

Рис.2.7. Код алгоритму циклічної перевірки надмірності

2.7.Інтеграція інформаційної системи

Для впровадження алгоритмів фільтрації та розрахунку CRC у програмне забезпечення, необхідно встановити пристрій у форматі пакетів необмеженої довжини. Це робиться шляхом встановлення біту формату пакета на 0 (пакет з фіксованою довжиною) і встановлення значення PayloadLength (змінна з коду алгоритму) на 0. Також для розрахунку CRC необхідно встановити початкове значення, встановивши біт CrcOn на 0.

Для налаштування передавача необхідно створення пакетів, які відповідають їхній звичайній структурі, як показано на наступному рисунку:



Рис.2.5. Створення пакетів даних для передавача

Пакети даних мають змінну довжину та контрольну суму CRC, при необхідності можливо додати байт адреси. Першим кроком є побудова пакета, який таким чином складається з 1 байта довжини корисної навантаження та n байтів передаваної інформації. Другим кроком є розрахунок CRC для цього складеного корисного навантаження (1 байт пакету даних та фактичне корисне навантаження). Останній крок – «відбілення» отриманого пакету, який тепер складається з 1 байта довжини корисного навантаження, n байтів фактичного корисного навантаження та 2 байтів CRC.

На стороні приймача відбувається зворотній процес, і виникає необхідність «відбілити» пакет під час передачі інформації, щоб його довжина пакета могла бути отримана з вхідних даних. Тут важливо зауважити, що пристрій не знає розміру

пакета, який він збирається отримати, і, отже, CRC або готове корисне навантаження ніколи не згенеруються.

Оскільки приймач не володіє інформацією про довжину пакету, необхідно встановити поріг рівня потоку інформації досить низьким, щоб генерувалося принаймні одне переривання заданого рівня пропуску даних, і було можливо «реверсивно відбілити» весь пакет даних, починаючи з підрахування їх кількості, поки не буде отримано усі.

2.8. Висновки другого розділу

У процесі виконання другого розділу було опрацьовано впровадження програмних засобів у використання технології LoRa в інформаційній системі, їх опис та методи застосування. Було вивчено методи підвищення швидкості передачі даних радіосигналами, наприклад, за допомогою принципу пакетування даних, що використовується для оптимізації процесу передачі інформації, оскільки дозволяє передавати більшу кількість інформації в одну секунду. Наступним етапом підвищення швидкості передачі інформації є впровадження алгоритмів побайтового та побітового фільтрування даних. Існують також їх реверсивні версії. Реалізація алгоритмів виконується мовою програмування C++ з подальшою інтерпретацією мовою РНР з метою можливості написання функцій для виклику цих алгоритмів у програмній частині розробляемого програмного забезпечення. Після закінчення використання принципу пакетування даних алгоритмів з метою перевірки правильності проведення операції використовується алгоритм циклічної перевірки надмінності.

РОЗДІЛ 3

ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ ТА ПРАКТИЧНЕ ЗАСТОСУВАННЯ

3.1. Вступ у дослідження

Технологія LoRa є власною схемою модуляції розширеного спектру, що є похідною модуляції розширення спектру (у цифрових комунікаціях, розширений спектр чірпінгу (CSS) - це метод розширеного спектра, який використовує широкопasmові імпульси чірпірування, модульовані лінійною частотою, для кодування інформації [17]), і яка обмінює швидкість передачі даних на чутливість у межах смуги пропускання фіксованого каналу. Чірп - це сигнал, у якому частота збільшується або зменшується з часом передачі, залежно від його типу [18]. Він реалізує змінну швидкість передачі даних, що дозволяє розробнику інформаційної системи обмінювати швидкість передачі даних на діапазон або потужність, щоб оптимізувати продуктивність мережі в постійній смузі пропускання. Наступний розділ магістерської роботи присвячений експериментальному дослідженню та практичному застосуванню даного процесу оптимізації, на базі вищезгаданого обміну та алгоритму фільтрації даних описаного у другому розділі.

3.2. Принципи розширення спектру

В основу концепції експерименту та теорії інформації покладено теорему Шеннона – Хартлі, що встановлює максимальну швидкість передачі інформації через канал зв'язку заданої смуги пропускання в присутності шуму.

Теорема встановлює пропускну здатність каналу Шеннона [19] для лінії зв'язку та визначає максимальну швидкість передачі даних (інформації), яка може

передаватися в межах заданої смуги пропускання за наявності перешкод і визначається за формулою:

$$C = B * \log_2\left(1 + \frac{S}{N}\right), \quad (1)$$

де C – ємність каналу (біт/с), B – пропускна здатність каналу (Гц), S – середня потужність отриманого сигналу (Вт), N – середня потужність шуму або перешкод (Вт).

Після підставлення значення натурального логарифму з основою 2 та наступних перетворень рівняння (1), стає зрозуміло, що для підвищення швидкості передачі пакетів інформації в каналі з фіксованим відношенням рівня перешкод до сигналу, потрібно збільшувати лише пропускну здатність переданого сигналу. Як було зазначено вище, збільшуючи смугу пропускання сигналу, можна компенсувати погіршення відношення «сигнал/шум» (або «шум/сигнал») радіоканалу.

У традиційних системах розширеного спектру прямого розподілу початкова фаза (фаза передавання) передавача змінюється відповідно до послідовності кодів. Зазвичай цей процес досягається множенням сигналу даних, що передаються, на розширювальний код. Таким чином, передача певної послідовності коротших кодованих бітів відбувається набагато швидше, ніж цілий сигнал даних, поширюється смуга пропускання сигналу за вихідну смугу, зайняту лише вихідним сигналом [20]. Усі ці дії спрямовані на підвищення швидкості передачі цілих пакетів даних, що є метою виконання роботи. Наступний рисунок ілюструє принцип розширення спектру:

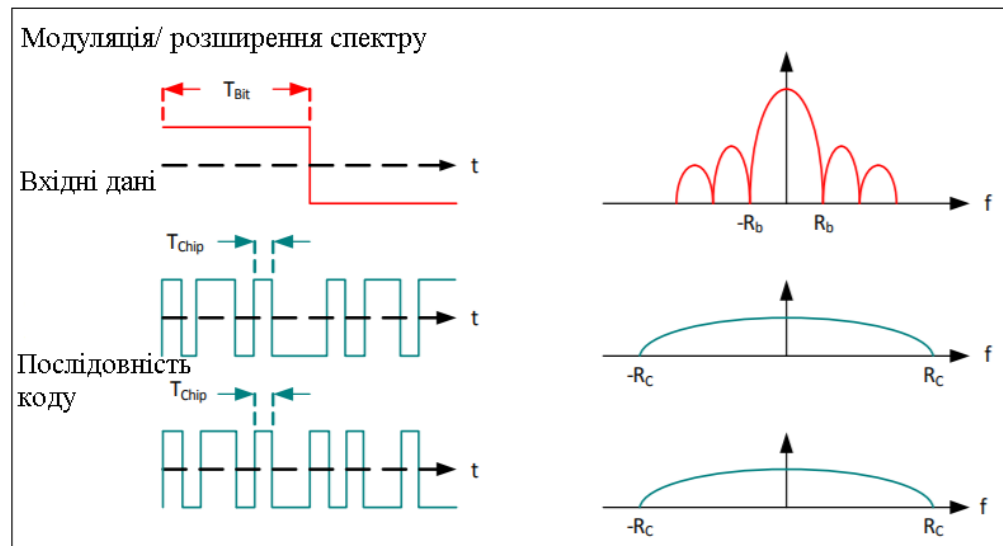


Рис 3.2. Процес модуляції/розширення спектру

У приймачі відшукується сигнал необхідних даних шляхом повторного множення з локально сформованою реплікою послідовності розповсюдження. Цей процес множення в приймачі ефективно стискає розповсюджений сигнал назад до вихідної нерозширеної смуги пропускання, як показано на Рис 3.3. Слід зазначити, що в приймачі повинна використовуватися та ж послідовність коротших кодованих бітів, що і в передавачі, щоб правильно відновити інформацію.

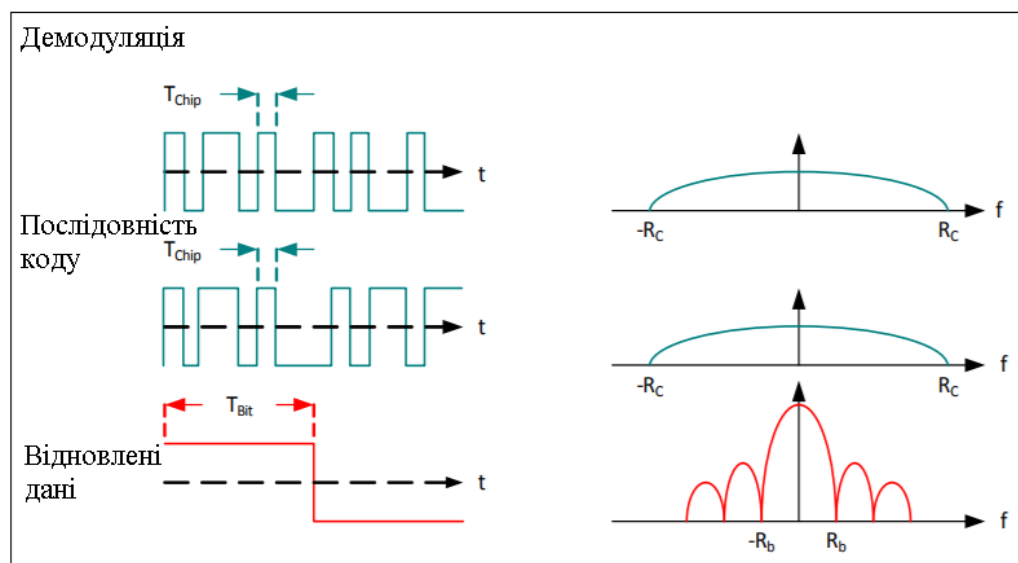


Рис.3.3. Демодуляція процесу розширення спектру

Даний алгоритм передбачає зменшення перешкодних сигналів, а також забезпечення властивого посилення обробки для поточної передачі (що дозволяє приймачеві правильно відновити сигнал даних) за рахунок посилення процесу приймача. Вони поширюються за межі бажаної пропускної здатності інформації і їх легко видалити фільтруванням. Крім того, чим довший код розподілу або послідовність пакетів даних, тим довший час, необхідний приймачеві для здійснення кореляції по всій його довжині.

При модуляції LoRa розповсюдження спектра досягається генерацією сигналу чирінгу, який постійно змінюється за частотою [21]. Перевагою цього методу є те, що часові та частотні зсуви між передавачем і приймачем рівнозначні, що значно зменшує складність конструкції приймача. Пропускна здатність частоти цього сигналу еквівалентна спектральній смузі пропускання сигналу. Даний фактор впливає на подальший розвиток розробки алгоритму покращення швидкості передачі даних за допомогою технології LoRa.

Потрібний сигнал змінюється зі збільшенням швидкості передачі даних і модулюється на сигнал чирінгу [22]. Оскільки для виконання практичних завдань потрібно розглядати різні типи сигналів, зв'язок між необхідною швидкістю передачі даних, швидкістю передачі символів та швидкістю змінних сигналів для модуляції LoRa може бути виражена таким чином:

Ми можемо визначити бітову частоту модуляції як:

$$R_b = SF * \frac{1}{\frac{2^{SF}}{BW}} \text{ біт/с}, \quad (2)$$

де SF – фактор розподілення (7..12)

BW – смуга пропускання подуляції (Гц)

Після визначення частоти та періоду передавання одного біту інформації, можливо визначити також частоту чірп-сигналу:

$$Rc = Rs * 2^{SF} \text{ сигнал/с} \quad (3)$$

Це може бути визначено як «надсилання одного чірп-сигнал в секунду на 1Гц пропускної здатності». Наступним необхідним етапом є включення в модуляції LoRa схему змінної корекції помилок, яка покращує стійкість переданого сигналу за рахунок надмірності [23][24]. Таким чином, ми можемо визначити номінальну частоту сигналу даних як:

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

де SF – фактор розподілення (7..12)

CR – частота коду (1..4)

BW – смуга пропускання модуляції (Гц)

3.3. Маніпуляція плинном частоти сигналу

Перевагою масштабованої пропускної та частотної здатності розробленого алгоритму є те, що він може бути використаний як для вузькосмугового стрибка частоти, так і для широкосмугових прямих послідовних додатків. На відміну від існуючих вузькосмугових або широкосмугових схем модуляції, LoRa можна легко адаптувати для будь-якого режиму роботи лише за допомогою декількох простих змін конфігураційного реєстру.

Завдяки функціональному направленню алгоритму на високу пропускну здатність та асинхронну природу обробки сигналів, він являється дуже стійким як до внутрішніх, так і до позасмугових перешкод [26], наприклад, можна отримати типові показники селективності каналу приймача 90 дБ з відхиленням на 20 дБ.

Принцип збільшення необхідної пропускної здатності сигналу для передачі даних без помилок на більшу відстань (тобто за наявності зростаючого відношення

перешкод до сигналу) є основним принципом розширеного спектру, що доведено низкою проведених експериментів у ході порівняння чутливості датчиків LoRa та конкурентного середовища трансиверів з маніпуляцією з частотним зміщенням, як показано на Рис. 3.4.:

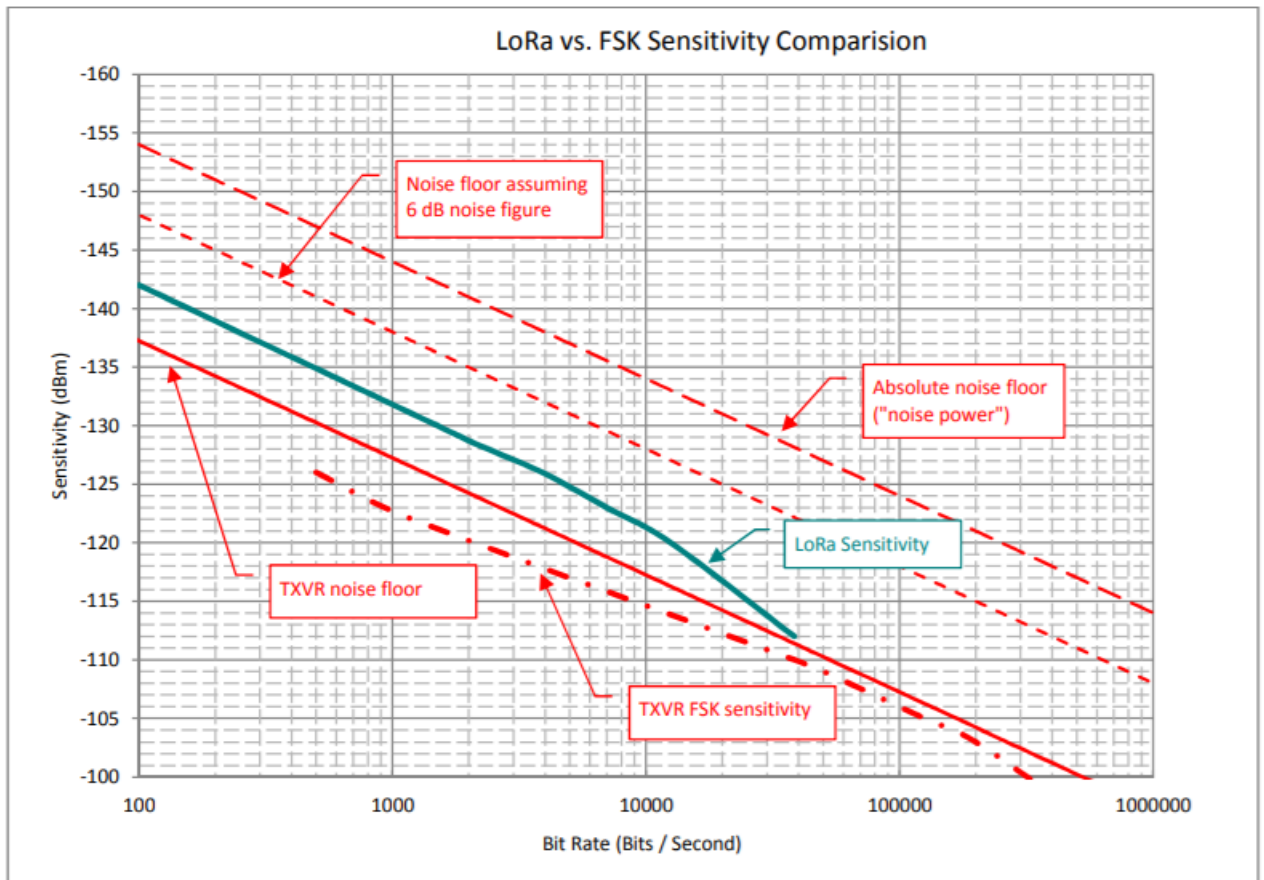


Рис. 3.4. Порівняння чутливості сенсорів

При порівнянні теоретичного мінімального рівня шуму з типовими показниками чутливості, стає помітним, що при низьких швидкостях передачі даних зазначена чутливість відхиляється від теоретичного інциденту потужності на рівні перешкод приймача через збільшення смуги пропускання каналу (фільтра), необхідної для компенсації очікуваних частотних помилок між передавачем і цільовим приймачем [27]. Важливо також зауважити, що поточний алгоритм

підвищення швидкості передачі інформації пропонує значне покращення чутливості приймання сигналів. Як видно із графіку, ця чутливість знаходиться в межах 6 дБ від рівня відносного шуму [28]

3.4. Механізми поширення багатопроменивих шляхів та співіснування мережі

Завдяки оснащенню технології LoRa багатопроменевою смугою, що є явищем поширення, яке призводить до того, що переданий радіосигнал досягає приймача двома або більше шляхами. Багатопроменеві механізми включають відображення від перешкод, що значно покращує та підтримує роботу розробляемого алгоритму. Механізми загасання багатопроменивих шляхів можна розглядати як плавне або частотно-вибіркове затухання [29].

У разі плоского затухання смуга пропускання каналу розповсюдження більша, ніж пропускання сигналу. У цьому випадку, навіть при тому, що спектральні властивості сигналу при прийомі не змінюються, амплітуда сигналу коливається з часом внаслідок змін коефіцієнта підсилення каналу, спричиненого багатопроменивим шляхом. Вузькосмугові системи частотного зсуву намагаються пом'якшити наслідки плоского затухання, застосовуючи методи спектрального розмаїття, такі як стрибок частоти.

Відправлені вузькосмугові сигнали мають кращу спритність частоти та здатність "пробивати" широкосмугові сигнали розширеного спектра (які традиційно обмежуються межами спектральної щільності потужності і, отже, передаються з меншою потужністю, ніж вузькосмугові сигнали), як показано нижче на Рис.3.5.:

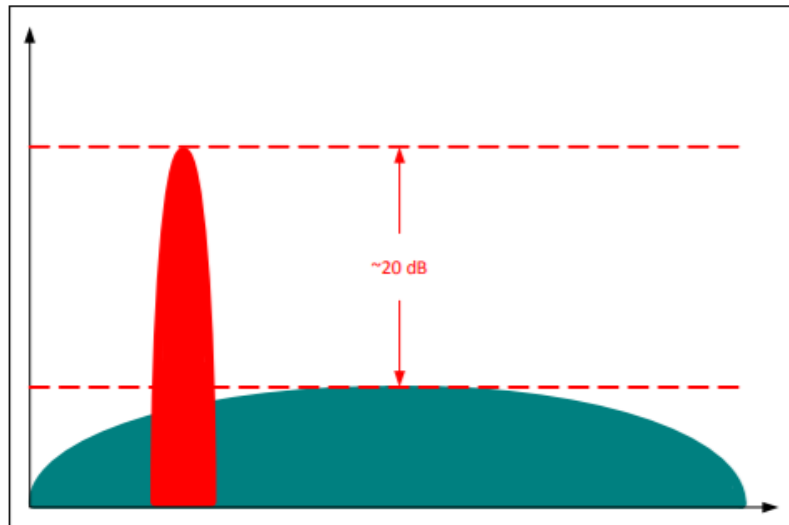


Рис.3.5. Пробивання широкосмугової перешкоди вузькосмуговим сигналом

Таким чином, широкосмуговий модульований LoRa сигнал в смузі пропускання смугою 6 дБ 500 кГц може передавати включно до на 27 дБм більше, що є близький до 30 дБм, дозволених для вузькосмугової модуляції, як показано на Рис. 3.6., без будь-яких обмежень часу перебування в каналі передачі.

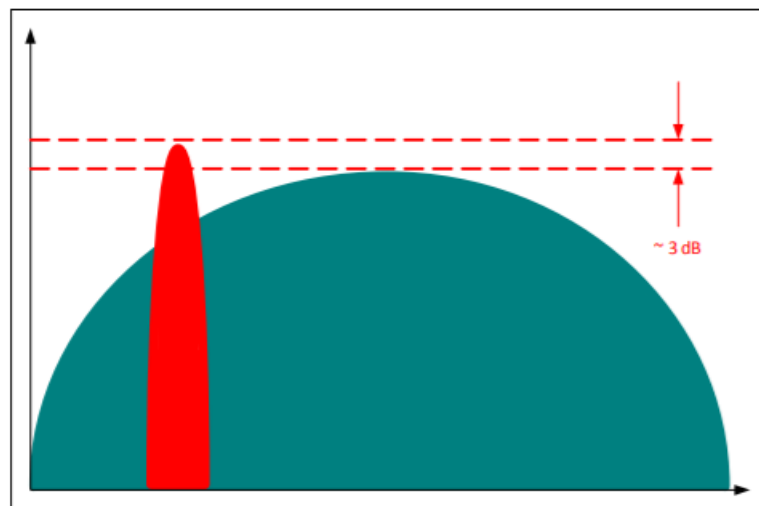


Рис.3.6. Вузькосмуговий сигнал проти широкосмугового перешкоди

Як зазначалося, при модуляції з розширеним спектром вплив перешкодних сигналів зменшується за рахунок посилення процесу, властивого модуляції. Ці

перешкоджаючі сигнали поширюються за межі необхідної пропускну здатності інформації і можуть бути легко видалені за допомогою алгоритму фільтрування, принцип якого описаний у другому розділі. При вузькосмуговій модуляції перешкодні сигнали не поширюються процесом демодуляції. Як видно з останнього рисунку, широкосмугова перешкода блокує вузькосмугову передачу сигналу, спричиняючи втрату пакету [30].

Оскільки тривалість періоду приймання широкосмугового сигналу може бути набагато більшою, ніж тривалість вузькосмугових сигналів, що періодично змінюють частоту, можна очікувати, що численні вузькосмугові сигнали цілком можуть попадати на широкосмугову модуляцію, як показано нижче на Рис. 3.7.:

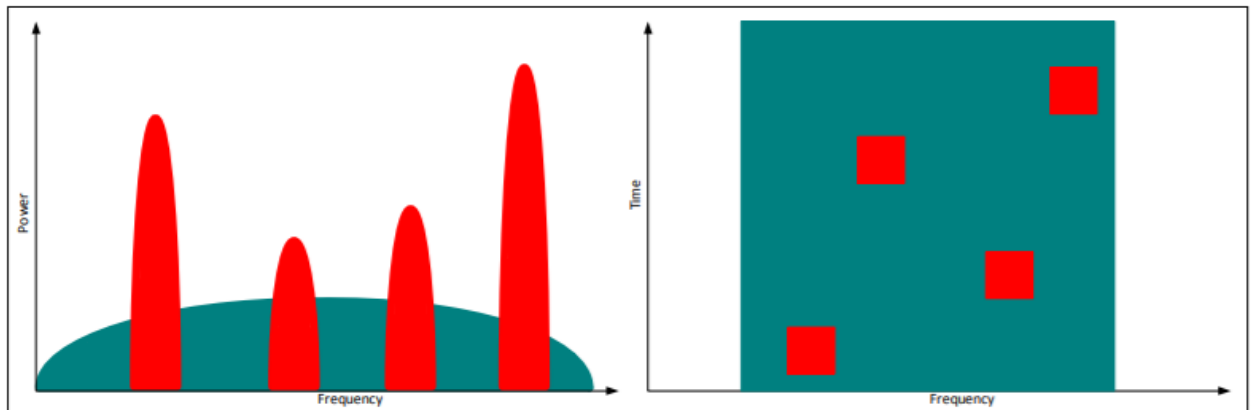


Рис.3.7. Широкасмуговий сигнал проти вузькосмугового перешкоди
(відношення частота/потужність та частота/час)

У ході виконання експерименту чотири вузькосмугові сигнали було пропущено через широкосмуговий сигнал. Однак тривалість вузькосмугових сигналів така, що у часовій області період перешкод є коротким у відношенні до широкосмугового сигналу. Завдяки надмірності, асоційованій із широкосмуговою модуляцією широкого спектра, модуляція є досить стійкою до механізму перешкод, який виглядає як короточасні імпульси. Цю тенденцію відображає наступний графік:

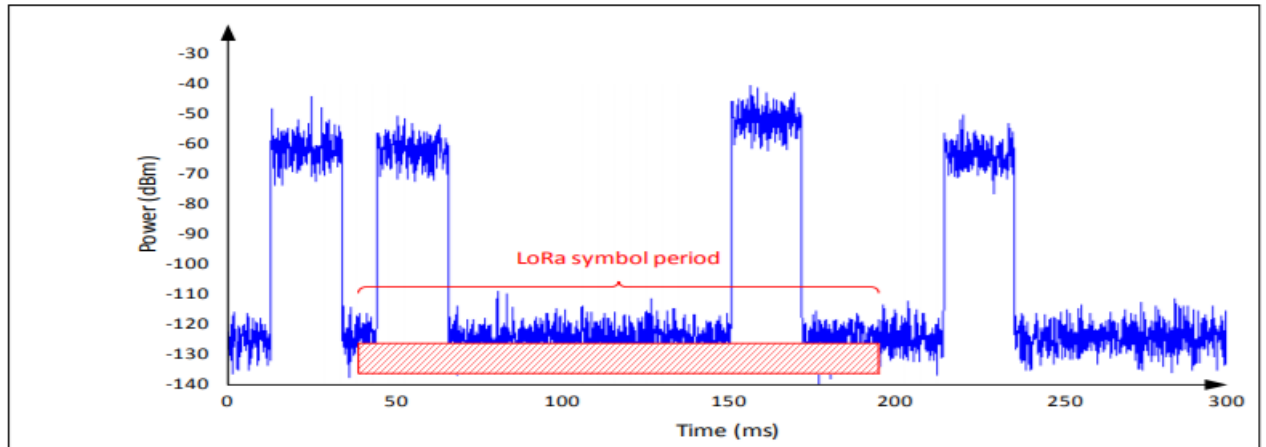


Рис.3.8. Візуалізація стійкості сигналів до серійних перешкод

Як наслідок, відбувається покращення передачі сигналів за рахунок налагодження механізмів інтерференційних сплесків довільних рівнів потужності до 30% довжини символу з погіршенням чутливості менше 6 дБ. Дана тенденція відображується на наступному графіку:

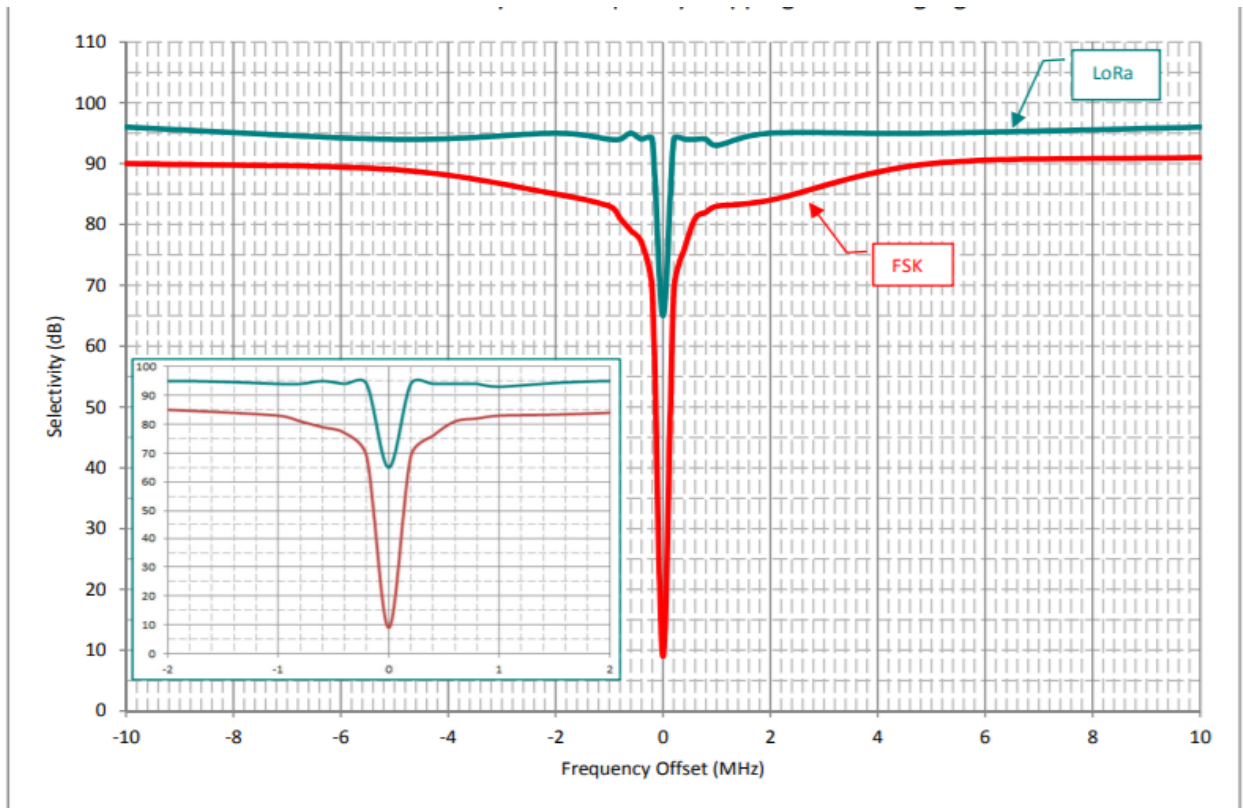


Рис.3.9. Візуалізація процесу покращення якості передання радіосигналу

Таким чином, спростерігається діапазон від 15 до 20 дБ у підвищенні стійкості до небажаних перешкод і діапазон у приблизно 10 дБ для допустимого зсуву частоти, що перевищують 5 МГц (як правило, зміщення понад 1 МГц, домінує над лінійними властивостями приймача) [32].

3.5. Дослідження рівня покращення ємності приймача інформаційного сигналу

Однією з помилкових думок щодо використання широкосмугової модуляції із розширеним спектром є те, що вона є спектрально неефективною порівняно з вузькосмуговою модуляцією. Тому в даному підрозділі буде розглянуто випадок вузькосмугової системи, що працює у віртуальному каналі з пропускною здатністю 125 кГц.

У якості прикладу використано випадок, у якому пакети даних з інформацією передаються через 12 вузькосмугових каналів 2-FSK (відомого аналогу приладу LoRa Semtech), які передають з еквівалентною швидкістю передачі даних 1,2 кБ/с, при цьому загальну теоретичну пропускну здатність каналу можна розрахувати як:

3.6. Практичне застосування покращеної мережі

У якості прикладу практичного застосування мережі сенсорів з покращеним алгоритмом передачі даних за допомогою радіосигналів на базі технології LoRa (у порівнянні з відомими аналогами, сенсорами з вузькосмуговою маніпуляцією зі зміщенням частоти) в непростих міських умовах, будуть розглянуті дані, що передаються з магнітних та температурних сенсорів з паркувальних місць паркувального комплексу, розташованого у м. Рим (Італія). Даний процес відображений на наступному рисунку:

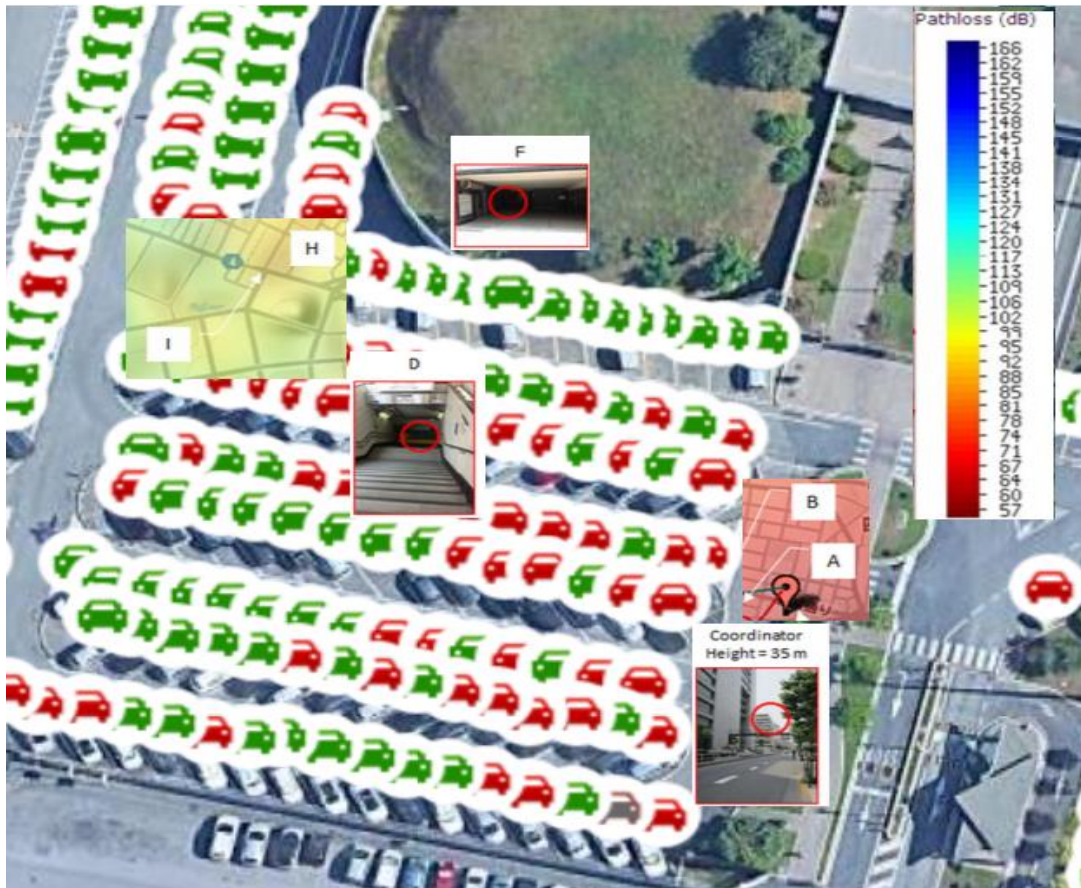


Рис 3.6. Практичне застосування мережі

Ref. #	Distance (m)	2-FSK: 4.8 kb/s		LoRa: 125 kHz BW, SF = 8 (3.125 kb/s)	
		Rssi (dBm)	PER (%)	Rssi (dBm)	PER (%)
A	80	-97	0	-91	0
B	150	-100	0	-102	0
C	280	-112	1	-114	0
D	330	-	100	-124	10
E	480	-118	8	-120	0
F	560	-	100	-121	0
G	1180	-	100	-112	0
H	1350	-	100	-126	10
I	1750	-	100	-127	100

Рис.3.6. Порівняння індикатору сили сигналу

У наведеному вище прикладі, LoRa Semtech та відомий аналог приладу (2-FSK) були встановлені у вищезгаданому паркувальному комплексі для передачі короткого корисного навантаження при номінальній вихідній потужності вищій за

13 дБм. Як можна спостерігати, у межах цього неоднорідного міського середовища досяжний діапазон рішення Semtech LoRa, як правило, втричі перевищує показники, отримані з приладу 2-FSK.

$$C_{FSK} = 12 * 1,2 \text{ кБ/с} = 14,4 \text{ кБ/с} \quad (5)$$

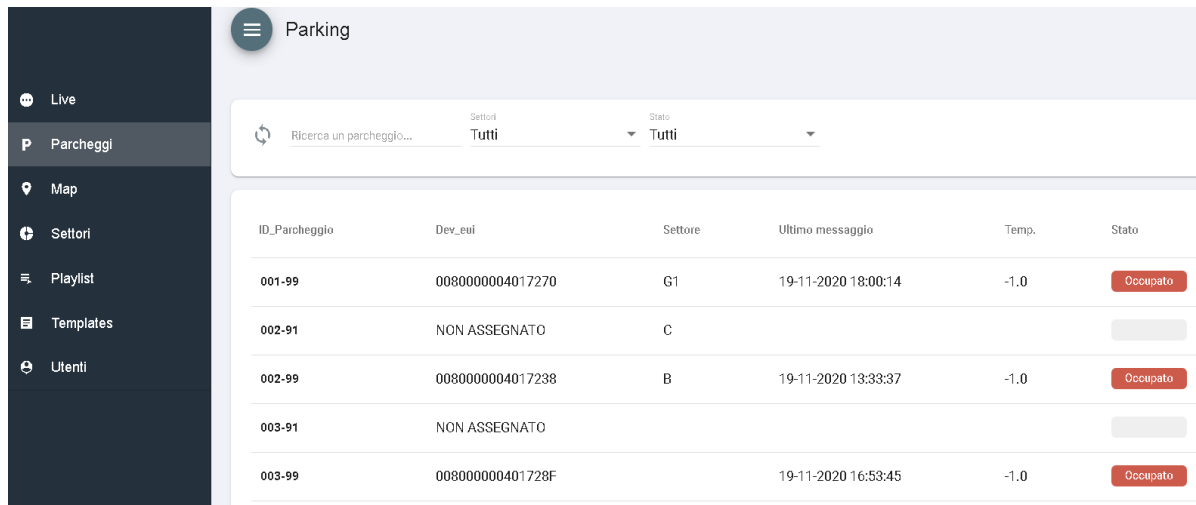
Розглядаючи той самий спектр передаваних сигналів, що розгортається як один канал LoRa 125 кГц, і користуючись перевагами оновлених за допомогою розробленого алгоритму факторів розповсюдження, еквівалентна пропускна здатність каналу стала майже вдвічі вищою (25 кБ/с).

Таким чином, спостерігається, що навіть під час процесу передачі даних, що еквівалентна швидкість якого є в 4 рази більше звичайної, модуляція LoRa забезпечує чутливість, подібну до звичайної системи FSK. Коли швидкість передачі даних приблизно еквівалентна, поліпшення LoRa становить від 7 до 10 дБ.

Розроблений та застосований алгоритм підвищення швидкості та якості передачі радіосигналів пристроями на базі технології LoRa Semtech – це програмна реалізація, яка забезпечує значне покращення процесу передачі інформації за допомогою радіосигналів порівняно із відомими аналогами приладу, що використовують звичайну вузькосмугову модуляцію. Крім того, підвищена надійність і вибірковість, що забезпечуються модуляцією розширеного спектру, дозволяють отримати більшу відстань передачі, навіть при наявності складних умов (перешкод). Це дозволяє одночасно передавати в межах одного каналу кілька пакетів даних, що значно покращує ефективність та пропускну здатність мережі, співіснування та взаємодію з існуючими розгортаннями застарілих мереж [33].

3.7. Опис веб-застосунку користувача

Після проведення роботи з дослідження, розробки та впровадження алгоритму для покращення процесу передачі інформації за допомогою радіосигналів через приймачі на базі технології LoRa, з метою систематизації, обробки та передачі даних до користувача з подальшою практичною експлуатацією. На наступних рисунках будуть представлені знімки екрану деяких функціоналів розробленого веб-застосунку користувача.



The screenshot shows a web application interface for parking management. On the left is a dark sidebar menu with options: Live, Parcheggi, Map, Settori, Playlist, Templates, and Utenti. The main content area is titled 'Parking' and features a search bar with a refresh icon and a search placeholder 'Ricerca un parcheggio...'. Below the search bar are two dropdown menus for 'Settori' (set to 'Tutti') and 'Stato' (set to 'Tutti'). A table displays the following data:

ID_Parcheggio	Dev_eui	Settore	Ultimo messaggio	Temp.	Stato
001-99	0080000004017270	G1	19-11-2020 18:00:14	-1.0	Occupato
002-91	NON ASSEGNATO	C			
002-99	0080000004017238	B	19-11-2020 13:33:37	-1.0	Occupato
003-91	NON ASSEGNATO				
003-99	008000000401728F		19-11-2020 16:53:45	-1.0	Occupato

Рис.3.7. Розділ користувацького меню з інформацією про паркувальні місця, їх статус та приналежність до певного сектору

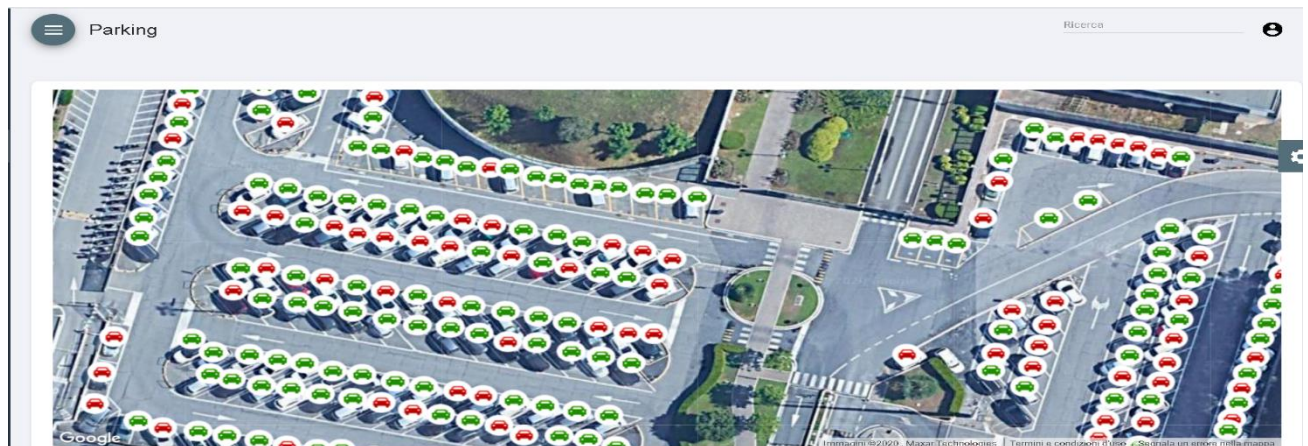
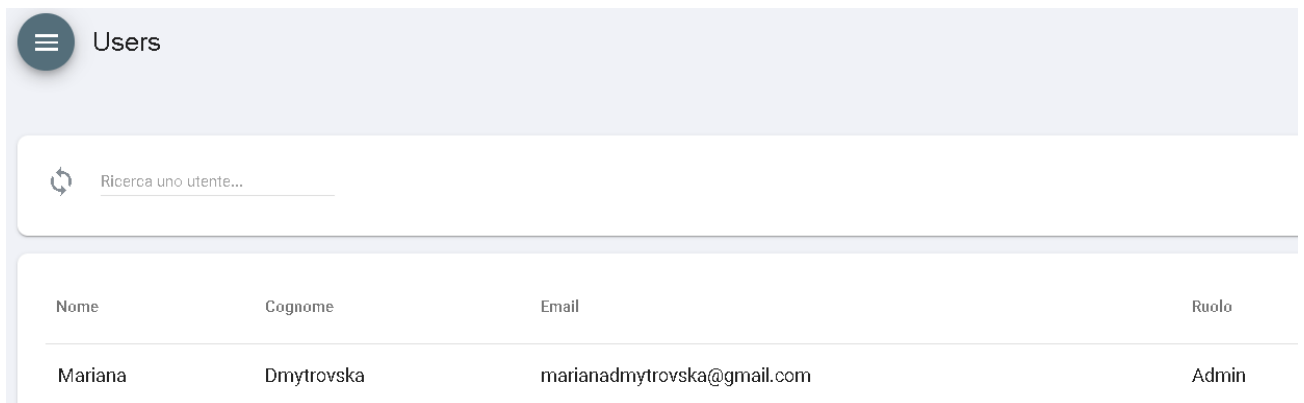
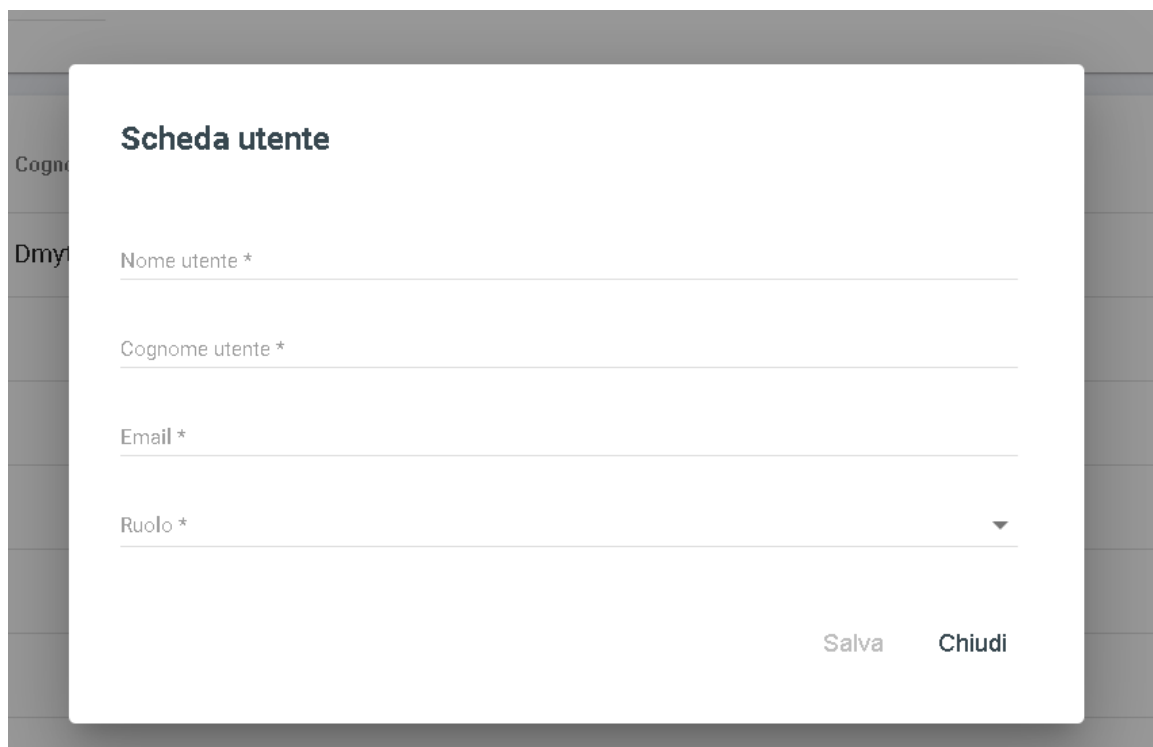


Рис. 3.8. Мапа паркувального комплексу



Nome	Cognome	Email	Ruolo
Mariana	Dmytrovska	marianadmytrovska@gmail.com	Admin

Рис. 3.9. Розділ зі списком зареєстрованих користувачів, їх дані, контакти та ролі



Scheda utente

Nome utente *

Cognome utente *

Email *

Ruolo *

Salva Chiudi

3.10. Функція реєстрації користувача в системі

Окрім веб-застосунку, були розроблені мобільні версії застосунків для Android та IOS операційних систем та окремі функції які викликаються для відображенні інформації на великому екрані у самому паркувальному комплексі.

Завдяки протоколу LoraWan існує можливість виклику іншого типу зворотного дзвінка, наприклад: приєднання до мережі, повідомлення по висхідній лінії та звіт по низхідній лінії зв'язку.

Нижче буде наведено приклад того, як реалізувати обробник зворотного виклику повідомлень висхідної лінії зв'язку за допомогою PHP для внутрішнього сервера додатків. Тип зворотного виклику задається параметром “ty” [як Type] із запитуваного URI. Потім тіло запиту містить дані json із усією інформацією про повідомлення висхідної лінії зв'язку. Багато інформації міститься в документі json. Тут для спрощення я просто витягнув дані з простого тексту та часову позначку GMT повідомлення. Відмітка часу в MS з часу ЕРОС.

У цьому прикладі створюється новий файл json obj.json з 3 записами на повідомлення: час, час, що читається людиною, і дані у текстовому форматі.

Тіло запиту читається за допомогою рядка `file_get_contents (“php://input”)`.

Json декодується методом `json_decode`. цей створює об'єкт із усіма елементами json, упорядкованими як у файлі json у ньому. Тоді кожен елемент може бути доступний під власною назвою, як це задокументовано в об'єктивному API.

```
<?php
    $_callbackType = $_GET["ty"];
    if ( $_callbackType == "20020" ) {
        $_fileName = "obj.json";
        $_file = fopen($_fileName,'a');
        $postdata = file_get_contents("php://input");
        $json = json_decode($postdata);
        $_data = $json->{'UplinkIndication'}->{'LORA-FRMPayloadClearText'};
        $_time = round ( $json->{'UplinkIndication'}->{'GTW-Timestamp'} / 1000
    );

        $_htime = gmdate("Y-m-d H:i:s",$_time);
        fwrite($_file,'{ "time" : "' . $_time . '", "htime" : "' . $_htime . '",
"data" : "' . $_data . '" },');
        fwrite($_file,"\n");
        fclose($_file);
    }
?>
```

3.7. Висновки третього розділу

У ході написання третього розділу кваліфікаційної роботи було вивчено існуючі методи підвищення швидкості та якості передачі інформації радіосигналами за допомогою технології LoRa. У даному випадку, під якістю розуміється стійкість сигналів до мережевих перешкод. Цієї мети досягнуто за допомогою сполучення використання алгоритмів, описаних у попередньому розділі разом із методом розширення спектру, який було вивчено та описано у третьому розділі. Даний метод базується на теоремі Шеннона – Хартлі, що встановлює максимальну швидкість передачі інформації через канал зв'язку заданої смуги пропускання в присутності шуму. Вона встановлює пропускну здатність каналу Шеннона для лінії зв'язку та визначає максимальну швидкість передачі даних (інформації), яка може передаватися в межах заданої смуги пропускання за наявності перешкод.

Після залучення усіх попередньо згаданих алгоритмів покращення швидкості та якості передачі даних, було проведено експеримент у якому пристрій на базі технології LoRa було порівняно з відомим аналогом (передатчиком інформації). У ході експерименту були виявлені значні відмінності в аспектах якості та швидкості передачі даних пристроями на базі технології LoRa, що і потрібно було довести. Після успішного проведення експерименту, на базі отриманих знань та удосконалених методів передачі даних радіосигналами, було розроблено програмне забезпечення у вигляді веб та мобільних застосунків користувача, які збирають, зберігають, організовують та передають дані до користувача щодо стану та температури паркувальних місць на території паркувального комплексу. Крім вищезгаданих основних функцій, було розроблено додатковий функціонал для користувача у вигляді можливості створення особистого кабінету, пошуку місцезнаходження вільного паркувального місця, можливості їх моніторингу в режимі реального часу, тощо.

РОЗДІЛ 4

ЕКОНОМІЧНА ЧАСТИНА

4.1. Визначення трудомісткості та вартості розробки програмного продукту

Початкові дані:

1. передбачуване число операторів програми – 12000;
2. коефіцієнт складності програми – 1,7;
3. коефіцієнт корекції програми в ході її розробки – 0,07;
4. годинна заробітна плата програміста – 90 грн/год;
5. коефіцієнт збільшення витрат праці внаслідок недостатнього опису задачі – 1,2;
6. коефіцієнт кваліфікації програміста, обумовлений від стажу роботи з даної спеціальності – 1,2;
7. вартість машино-години ЕОМ – 15 грн/год.

Нормування праці в процесі створення ПЗ істотно ускладнено в силу творчого характеру праці програміста. Тому трудомісткість розробки ПЗ може бути розрахована на основі системи моделей з різною точністю оцінки.

Трудомісткість розробки ПЗ можна розрахувати за формулою:

$$t = t_o + t_u + t_a + t_n + t_{отл} + t_{\partial}, \text{ людино-годин, (4.1)}$$

де t_o - витрати праці на підготовку й опис поставленої задачі (приймається 50 людино-годин);

t_u - витрати праці на дослідження алгоритму рішення задачі;

t_a - витрати праці на розробку блок-схеми алгоритму;

t_n -витрати праці на програмування по готовій блок-схемі;

t_{oml} -витрати праці на налагодження програми на ЕОМ;

t_{∂} - витрати праці на підготовку документації.

Складові витрати праці визначаються через умовне число операторів у програмному забезпеченні, яке розробляється.

Умовне число операторів:

$$Q = q \cdot C \cdot (1 + p), \quad (4.2)$$

де q - передбачуване число операторів (12000);

C - коефіцієнт складності програми (1,7);

p - коефіцієнт корекції програми в ході її розробки (0,07).

Звідси умовне число операторів в програмі:

$$Q = 1,7 \cdot 12000 \cdot (1 + 0,07) = 22000$$

Витрати праці на вивчення опису задачі t_u визначається з урахуванням уточнення опису і кваліфікації програміста:

$$t_u = \frac{Q \cdot B}{(75..85) \cdot k}, \text{ людино-годин, } (4.3)$$

де B - коефіцієнт збільшення витрат праці внаслідок недостатнього опису задачі;

k - коефіцієнт кваліфікації програміста, обумовлений від стажу роботи з даної спеціальності. При стажі роботи від 3 до 5 років він складає 1,2.

Прийmemo збільшення витрат праці внаслідок недостатнього опису завдання не більше 50% ($B = 1,2$). З урахуванням коефіцієнта кваліфікації $k = 1,2$, отримуємо витрати праці на вивчення опису завдання:

$$t_u = \frac{(22000 \cdot 1,2)}{(75 \cdot 1,2)} = 294 \text{ людино-годин}$$

Витрати праці на розробку алгоритму рішення задачі визначаються за формулою:

$$t_a = \frac{Q}{(20 \dots 25) \cdot k}, \text{ людино-годин, (4.4)}$$

де Q – умовне число операторів програми;

k – коефіцієнт кваліфікації програміста.

Підставивши відповідні значення в формулу (4.4), людино-годин:

$$t_a = \frac{22000}{(20 \cdot 1,2)} = 917 \text{ людино-годин.}$$

Витрати на складання програми по готовій схемі:

$$t_n = \frac{Q}{(20 \dots 25) \cdot k}, \text{ людино-годин, (4.5)}$$

$$t_n = \frac{(22000 \cdot 1,2)}{(20 \cdot 1,2)} = 1100 \text{ людино-годин.}$$

Витрати праці на налагодження програми на ЕОМ:

- за умови автономного налагодження одного завдання:

$$t_{отл} = \frac{Q}{(4..5) \cdot k}, \text{ людино-годин, (4.6)}$$

$$t_{omл} = \frac{22000}{(5 \cdot 1,2)} = 3665 \text{ людино-годин.}$$

- за умови комплексного налагодження завдання:

$$t_{omл}^k = 1,5 \cdot t_{omл}, \text{ людино-годин, (4.7)}$$

$$t_{omл}^k = 1,5 \cdot 3665 = 5497 \text{ людино-годин.}$$

Витрати праці на підготовку документації визначаються за формулою:

$$t_{\partial} = t_{\partial p} + t_{\partial o}, \text{ людино-годин, (4.8)}$$

де $t_{\partial p}$ -трудомісткість підготовки матеріалів і рукопису:

$$t_{\partial p} = \frac{Q}{(15..20) \cdot k}, \text{ людино-годин, (4.9)}$$

$t_{\partial o}$ - трудомісткість редагування, печатки й оформлення документації:

$$t_{\partial o} = 0,75 \cdot t_{\partial p}, \text{ людино-годин, (4.10)}$$

Підставляючи відповідні значення, отримаємо:

$$t_{\partial p} = \frac{22000}{(18 \cdot 1,2)} = 1018 \text{ людино-годин.}$$

$$t_{\partial o} = 0,75 \cdot 1018 = 763 \text{ людино-годин.}$$

$$t_{\partial} = 1018 + 763 = 1781 \text{ людино-годин.}$$

Повертаючись до формули (3.1), отримаємо повну оцінку трудомісткості розробки програмного забезпечення:

$$t = 50 + 294 + 917 + 1100 + 3665 + 1781 = 7807 \text{ людино-годин.}$$

4.2. Витрати на створення програмного забезпечення

Витрати на створення ПЗ $K_{ПО}$ включають витрати на заробітну плату виконавця програми $Z_{ЗП}$ і витрат машинного часу, необхідного на налагодження програми на ЕОМ:

$$K_{ПО} = Z_{ЗП} + Z_{МВ}, \text{ грн. (4.11)}$$

Заробітна плата виконавців визначається за формулою:

$$Z_{ЗП} = t \cdot C_{ПР}, \text{ грн, (4.12)}$$

де: t - загальна трудомісткість, людино-годин;

$C_{ПР}$ - середня годинна заробітна плата програміста, грн/година

З урахуванням того, що середня годинна зарплата програміста становить 90 грн / год, отримуємо:

$$Z_{ЗП} = 7807 \cdot 90 = 702,630 \text{ грн.}$$

Вартість машинного часу, необхідного для налагодження програми на ЕОМ, визначається за формулою:

$$Z_{mv} = t_{отл} \cdot C_{мч}, \text{ грн, (4.13)}$$

де $t_{отл}$ - трудомісткість налагодження програми на ЕОМ, год;

$C_{мч}$ - вартість машино-години ЕОМ, грн/год (15 грн/год).

Підставивши в формулу (4.13) відповідні значення, визначимо вартість необхідного для налагодження машинного часу:

$$Z_{mv} = 3665 \cdot 15 = 54,975 \text{ грн.}$$

Звідси витрати на створення програмного продукту:

$$K_{ПО} = 702,630 + 54,975 = 757,605 \text{ грн.}$$

Очікуваний період створення ПЗ:

$$T = \frac{t}{B_k \cdot F_p}, \text{ міс.}$$

де B_k - число виконавців (дорівнює 1);

F_p - місячний фонд робочого часу (при 40 годинному робочому тижні $F_p=176$ годин).

Звідси витрати на створення програмного продукту:

$$T = \frac{7807}{(1 \cdot 176)} \approx 9 \text{ місяців.}$$

4.3. Маркетингові дослідження ринку збуту розробленого програмного продукту

У сучасному світі розвиток інновацій у секторі інформаційних технологій щодня набирає обертів та розширює свої потреби у вигляді програмних продуктів. Технологія модуляції LoRa (Long Range) являє собою метод модуляції, який забезпечує значно більшу дальність зв'язку (зону покриття), ніж інші конкуруючі з ним способи. Метод ґрунтується на технології модуляції з розширеним спектром і варіації лінійної частотної модуляції (Chirp Spread Spectrum, CSS) з інтегрованою прямою корекцією помилок (Forward Error Correction, FEC). Технологія LoRa значно підвищує чутливість приймача і, аналогічно іншим методам модуляції з розширеним спектром, використовує всю ширину смуги пропускання каналу для передачі сигналу, що робить його стійким до канальним шумів і нечутливим до зсувів, викликаних неточностями в налаштуванні частот при використанні недорогих резонаторів. Через універсальність використання пристроїв для передачі інформації на базі технології LoRa, окрім можливості їх використання в якості впровадженні програмного забезпечення для паркувального комплексу, існує дуже багато цікавих ринків збуту для даного продукту, таких як:

- електроенергетична промисловість;
- сфера комунальних послуг;
- сфера розробки систем безпеки (наприклад, «Розумний Дім»)
- сфера охорони здоров'я;
- сфера охорони навколишнього середовища;
- сфера громадського харчування;
- сільськогосподарська сфера та інші;

В рамках кваліфікаційної роботи було розглянуто прилади передачі інформації за допомогою радіохвиль на базі технології LoRa, так як вони мають ряд значних переваг:

- низький рівень споживання електричної енергії
- передача пакетів даних (сигналів) на більш великі відстані у порівнянні з відомими аналогами, а також у середовищі, з більшою кількістю мережових перешкод;
 - легкість в автоматизації алгоритмів обробки даних, що поліпшує процес контролю системи;
 - оптимізована чутливість сенсорів, за допомогою чого отримується вища якість процесу передачі та обробки радіосигналів.

На базі проведених досліджень та експериментів, було покращено алгоритми передачі та обробки пакетів інформації, що дозволило покращити швидкість даних процесів та стійкість переданих даних до мережових перешкод.

Після чого було розроблено власне програмний продукт у вигляді веб-застосунку, який збирає та організовує дані про стан паркувальних місць на паркувальному комплексі у м. Рим (Італія), організовуючи їх та передаючи в особистий кабінет користувача, що згодом допомагає йому орієнтуватися на території, знаходити потрібне паркувальне місце та переглядати дані щодо його стану. Окрім веб-застосунку, для користувачів доступні також його Android та IOS версії, та розроблені окремі функції для виводу інформації щодо стану паркувальних місць на великий екран комплексу.

Використання даного програмного продукту адаптоване під будь-який тип користувача, навіть найменш досвідченого, що значно розширює не тільки ринок збуту продукту, а також і споживацький ринок. Це досягнуто завдяки впровадженню простого та зрозумілого у користуванні інтуїтивного інтерфейсу користувача. Розроблені також Android та IOS додатки, що стирає абсолютно будь-які ліміти для користувачів, тому що робить програмний продукт незалежним від фактору уподобань конкретної операційної системи або пристрою.

Розроблене програмне забезпечення відноситься до складної програмної продукції, яка потребує спеціального налагодження та використання прямого

маркетингу між виробником і споживачем. Продукт розроблений за замовленням споживача, що передбачає можливість у разі потреби його доробки та обслуговування, випуску нових версій.

4.4. Оцінка економічної ефективності впровадження розробленого програмного забезпечення

У ході виконання кваліфікаційної роботи було розроблено програмне забезпечення з цільовою направленістю на приватних користувачів. У зв'язку з чим економічна оцінка ефективності запропонованого продукту поки що не може бути оцінена, оскільки впровадження розробленого програмного продукту ще не було реалізовано, проте можливо визначити його соціальний ефект.

Основною перевагою у впровадженні розробленого програмного продукту з точки зору соціальної ефективності є скорочення кількості працівників у паркувальному комплексі за рахунок автоматизації деяких дій за допомогою даного програмного забезпечення, що забезпечує зменшення потоку персоналу на користь постійного кругообігу відвідувачів паркувального комплексу. Крім цього, забезпечується легкість та скорочення часу в пересуванні по території об'єкту, до якого залучається використання даного програмного продукту, що є важливим фактором у задоволенні його клієнтів та залучення нових.

Впровадження програмного забезпечення також підвищує рівень безпеки підприємства та продуктивність праці робітників за рахунок налагодження руху на його території та зменшення витрат в часі на пошук вільного паркувального місця в бажаній зоні.

ВИСНОВКИ

У ході виконання магістерської роботи було вивчено та досліджено методи передачі радіосигналів за допомогою технології LoRa та мови програмування PHP на базі фреймворків Slim та Angular. Було досягнуто поставленої мети, яка полягала у дослідженні та вдосконаленні методів підвищення швидкості передачі та обробки даних з магнітних та температурних сенсорів за допомогою технології LoRa. Поставленої цілі було досягнуто за рахунок впровадження покращення в структуру вже існуючих алгоритмів фільтрації та передачі даних за рахунок розширення приймаючого каналу математичним методом, що дозволяє приймання більшої кількості пакетів інформації, що прискорює швидкість процесу передачі даних.

Цієї мети досягнуто за допомогою сполучення використання алгоритмів, описаних у попередньому розділі разом із методом розширення спектру, який було вивчено та описано у третьому розділі. Даний метод базується на теоремі Шеннона – Хартлі, що встановлює максимальну швидкість передачі інформації через канал зв'язку заданої смуги пропускання в присутності шуму. Вона встановлює пропускну здатність каналу Шеннона для лінії зв'язку та визначає максимальну швидкість передачі даних (інформації), яка може передаватися в межах заданої смуги пропускання за наявності перешкод.

Разом з цим було збільшено також місткість та чутливість каналів приймання інформації, що дозволило досягти покращення стійкості приймачів до мережеских перешкод. У ході експерименту було також проведено порівняння показників з відомим аналогом пристрою. Використовуючи створені покращення, було розроблено інформаційну систему для систематизації процесів передачі даних до користувача у вигляді веб та мобільних додатків, що мають також практичне використання.

Розроблене програмне забезпечення створене з метою впровадження більш зручного та легкого способу для навігації клієнтів та робітників паркувального

комплексу по його території, знаходження, зберігання та отримання інформації щодо місцезнаходження, температури та статусу («вільний» або «зайнятий») паркувального місця. Вартість даного програмного забезпечення становить 757,605 грн. з можливими додатковими витратами при розробці проекту при бажанні замовника випустити оновлену версію додатку через деякий час. Очікуваний час розробки - 9 місяців. Цей термін пов'язаний зі значною кількістю операторів і включає в себе час для дослідження та розробки алгоритму розв'язання задачі, розробки дизайну, створення мобільного та веб додатків та підготовку документації.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ullo, Silvia Liberata; Sinha, G. R. (2020-05-31). "Advances in Smart Environment Monitoring Systems Using IoT and Sensors". *Sensors (Basel, Switzerland)*. 20 (11):3113. doi:10.3390/s20113113. ISSN 8220. PMC 7309034. PMID 32486411.
2. FrancescoMario, Di; K, DasSajal; AnastasiGiuseppe (2011-08-01). "Data Collection in Wireless Sensor Networks with Mobile Elements". *ACM Transactions on Sensor Networks (TOSN)*. 8: 1–31. doi:10.1145/1993042.1993049. S2CID 15576441.
3. Xia, Feng; Tian, Yu-Chu; Li, Yanjun; Sun, Youxian (2007-10-09). "Wireless Sensor/Actuator Network Design for Mobile Control Applications". *Sensors(Basel,Switzerland)*. 7 (10):2157–2173. doi:10.3390/s7102157. ISSN 1424-8220. PMC 3864515. PMID 28903220.
4. Beser, Nurettin Burcak. "Operating cable modems in a low power mode." U.S. Patent No. 7,389,528. 17 June 2008.
5. Schwartzman, Alejandro, and Chrisanto Leano. "Methods and apparatus for enabling and disabling cable modem receiver circuitry." U.S. Patent No. 7,587,746. 8 September 2009.
6. Article “Long-Range Communications in Unlicensed Bands: the Rising Stars in the IoT and Smart City Scenarios”.
7. Low Power, Wide Area. A Survey of Longer-Range IoT Wireless Protocols
8. Приложение № 11 к решению ГКРЧ от 7 мая 2007 года № 07-20-03-001
9. Article “Long-Range Communications in Unlicensed Bands: Could street lighting be the platform for smart city LPWANs?”
10. Kruse, Robert L. (1987) [1984]. *Data Structures & Program Design* (second edition). Joan L. Stone, Kenny Beck, Ed O'Dougherty (production process staff workers) (second (hc) textbook ed.). Englewood Cliffs, New Jersey 07632: Prentice-Hall, Inc. div. of Simon & Schuster. pp. 150. ISBN 0-13-195884-4.

11. IBM SNA Formats Bit Ordering is Opposite of Intel Convention". Microsoft. 2014-02-23.
12. R. Gonzalez and R. Woods Digital Image Processing, Addison-Wesley Publishing Company, 1992, pp 47 - 51.
13. Buczynski, Don (2002-09-05). "MSB/LSB Tutorial"
14. Craig, Edward, ed. (1998), Routledge Encyclopedia of Philosophy, 10, Taylor & Francis, p. 496
15. Geremia, Patrick. "Cyclic Redundancy Check Computation: An Implementation Using the TMS320C54x" (PDF). Texas Instruments. p. 6.
16. "An Algorithm for Error Correcting Cyclic Redundancy Checks". drdobbs.com. Archived from the original on 20 July 2017. Retrieved 28 June 2017.
17. Computer Society, (August 31, 2007). IEEE Standard 802.15.4a-2007. New York, NY: IEEE.
18. Weisstein, Eric W. "Sweep Signal". From MathWorld--A Wolfram Web Resource.
19. Варгаузин, В.А., Цикин, И.А. Методы повышения энергетической и спектральной эффективности цифровой радиосвязи.
20. Ellingson, Steven W. (2016). Radio Systems Engineering. Cambridge University Press. pp. 16–17.
21. Balanis, Constantine A. (2017). Antenna theory: Analysis and Design, 3rd Ed. John Wiley and Sons. pp. 10
22. Ellingson, Steven W. (2016). Radio Systems Engineering. Cambridge University Press. pp. 16–17.
23. Kimmel WD, Gerke D (2018). Electromagnetic Compatibility in Medical Equipment: A Guide for Designers and Installers. Routledge. p. 6.
24. Semtech Application Note AN1200.13, "SX1272/3/6/7/8: LoRa Modem Designer's Guide".

25. Semtech Application Note AN1200.17, “SX1272/3/6/7/8: LoRa Energy Consumption Design”.
26. Rappaport, “Wireless Communications Principles and Practices,” (Prentice-Hall).
27. Xiong, “Digital Modulation Techniques, 2nd ed.,” (Artech House).
28. Siwak, Bahreini, “Radiowave Propagation and Antennas,” (Artech House).
29. FCC Office of Engineering and Technology Laboratory Division “Guidance for Performing Compliance Measurements on Digital Transmission Systems (DTS) Operating”.
30. OFCOM, “Use of Short Range Devices alongside mobile broadband services operating in the 800MHz band”.
31. CEPT ECC Report 207, Adjacent band co-existence of SRDs in the band 863-870 MHz in light of the LTE usage below 862 MHz.
32. ERA Technology, “Investigation on the receiver characteristics of SRD equipment in the 863- 870 MHz band”.
33. NPSTC, “Public Safety Related Spectrum Issues in the 902-928 MHz Band”.
34. "Semtech Selected for Smart Irrigation System | San Fernando Valley Business Journal". sfvbj.com. Retrieved 2019-02-11.
35. "LoRa Solution Offers Wireless View Into Soil Health - 2018-11-09 - Page 1 - RFID Journal". www.rfidjournal.com. Retrieved 2019-02-11.
36. "Spain pilots smart water technology". Smart Cities World. Retrieved 2019-02-11.
37. "Dutch firm launches LoRa smart agriculture solution on Senet network". enterpriseiotinsights.com. Retrieved 2019-02-11.
38. "Calgary, Alberta unveils 4 new smart city initiatives". Smart Cities Dive. Retrieved 2019-02-15.

39. "Orange has performed, within the Intelligent City Pilot project, the installation of a LPWA (Low Power Wide Area) network that covers the entire city". Alba Iulia Smartcity. Retrieved 2019-07-17.
40. "Semtech's LoRa Alliance grows to 500 members". Pacific Coast Business Times. 2017-06-29. Retrieved 2019-02-09.
41. "Member Directory | LoRa Alliance". lora-alliance.org. Retrieved 2019-02-09.
42. "LoRa Alliance passes 100 LoRaWAN network operator milestone". Electronic Products & Technology. 2019-01-25. Retrieved 2019-02-11.
43. Gallivan, Joseph (5 January 2018). "Las Vegas Arrivals". Business Tribune. Retrieved 5 February 2020. The alliance is managed by the VTM Group in Beaverton.... Geoff Mulligan is the Chairman of the LoRa Alliance. He was a presidential innovation Fellow at the National Institute of Standards and Technology (NIST) under the Obama administration.
44. "Board, Chairs & Staff | LoRa Alliance®". lora-alliance.org. Retrieved 2020-05-27.

ЛІСТИНГ ПРОГРАМИ

```

<?php
require 'vendor/autoload.php';
// Slim framework
use Psr\Http\Message\ResponseInterface as Response;
use Psr\Http\Message\ServerRequestInterface as Request;
use Slim\Routing\RouteCollectorProxy as RouteCollectorProxy;
use Slim\Exception\HttpNotFoundException as HttpNotFoundException;
use Slim\Factory\AppFactory;
use Slim\Middleware\OutputBufferingMiddleware;
set time limit( 0 );
error_reporting(1);
ini_set('display_errors', 1);
ini_set('display_startup_errors', 1);
putenv('DB_HOST=localhost');
putenv('DB_USER=unipark');
putenv('DB_PASSWORD=unipark.2016.01');
putenv('DB_NAME=mylora');
putenv('DB_PORT=3306');
define( "USER_ADMIN", 4 );
define( "USER_TERZO", 3 );
define( "USER_AMMINISTRATORE", 2 );
define( "USER_CONDOMINO", 1 );
$app = AppFactory::create();
$app->setBasePath('/sogei dev');
Logger::api($app);
//$auth = new Auth("TOKENOTTIMIS", "func", "idRole", "scopes", "extra");
$auth = new Auth("TOKENOTTIMIS", "func", "idRole", "scopes", "extra");
$authMW = function (Request $request, RequestHandler $handler) use ($auth) {
    $authHeaders = $request->getHeader('Authorization');
    if (sizeof($authHeaders) == 0) {
        $response = new Slim\Psr7\Response();
        $response->getBody()->write('Utente non autenticato');
        return $response
            ->withStatus(401);
    }
    $token = substr($authHeaders[0], 7);
    $ret = verifyToken( $token, $auth );

    if ($ret['success']) {
        $request = $request->withAttribute('user', $ret['user'] );
        $response = $handler->handle($request);
        return $response;
    } else {
        $response = new Slim\Psr7\Response();
        $response->getBody()->write("La sessione è scaduta, effettua nuovamente il login.");
        return $response
            ->withStatus(401);
    }
};
$tokenRenew = function (Request $request, RequestHandler $handler) use ($auth) {
    $authHeaders = $request->getHeader('Authorization');
    $token = substr($authHeaders[0], 7);
    if ($token != "") {
        $ret = verifyToken($token, $auth);
    }
    if ($ret['user']) {
        $request = $request->withAttribute('token', $token);
        $request = $request->withAttribute('user', $ret['user']);
    }
    $response = $handler->handle($request);
    return $response;
};

```

```

};
$userMW = function (Request $request, RequestHandler $handler) use ($auth) {
    $log = new Logger();
    $authHeaders = $request->getHeader('Authorization');
    $token = substr($authHeaders[0], 7);
    if ($token != "") {
        $ret = verifyToken($token, $auth);
    } else {
        $ret['user'] = "";
    }

    if ($ret['user'] && $ret['user'] != '') {
        $request = $request->withAttribute('user', $ret['user']);
    }
    $response = $handler->handle($request);
    return $response;
};

$app->options('/{routes:.+}', function ($request, $response, $args) {
    return $response;
});

$app->add(function ($request, $handler) {
    $log = new Logger();
    $response = $handler->handle($request);
    return $response
        ->withHeader('Access-Control-Allow-Origin', '*')
        ->withHeader('Access-Control-Allow-Headers', 'X-Requested-With, Content-Type, Accept,
Origin, Authorization')
        ->withHeader('Access-Control-Allow-Methods', 'GET, POST, PUT, DELETE, OPTIONS');
});

$app->get('/', function (Request $request, Response $response, $args) {
    $log = new Logger();
    $log->log('prove');
    $response->getBody()->write('Ciao 4');
    return $response;
});

require 'import.php';
$app->post('/app data', function (Request $request, Response $response, $args) {
    $db = new dataBase();
    $log = new Logger();
    $p = json_decode($request->getBody(), true);
    $log->log( "appdata: " . json_encode( $p ) );
    $dev_eui = "";
    foreach( $p as $k => $rec ) {
        $dev_eui .= ($dev_eui!="")?","":"";
        $dev_eui .= sprintf( "%s", $rec['data']['devEui'] );
    }
    // AND rx_time_received > (now() - INTERVAL 30 SECOND)
    // $sql = sprintf( "SELECT dev_eui as devEui, park state as parkState,state changed
    // FROM sogei.devices changes
    // WHERE dev_eui IN (%s)", $dev_eui );
    $sql = "SELECT dev_eui as devEui, park state as parkState,state changed,
DATE_FORMAT(rx_time_received,'%d-%m-%y %H:%i:%s') lastUpdate FROM sogei.devices_changes";
    $log->log( $sql );
    $db->query( $sql );
    while( $rec = $db->fetchassoc() ) {
        $ar[] = $rec;
    }
    $response->getBody()->write( json_encode( $ar ) );
    return $response
        ->withStatus(200)
        ->withHeader('Content-Type', 'application/json');
});

$app->post('/lora status', function (Request $request, Response $response, $args) {
    $db = new dataBase();
    $log = new Logger();
    $log->log("chiamata lora status");
    $sql = "SELECT dev_eui asdevEui, park_state as parkState,state_changed,
DATE_FORMAT(rx time received,'%d-%m-%y %H:%i:%s') lastUpdate FROM sogei.devices changes";
    $log->log( $sql );

    $db->query( $sql );
    while( $rec = $db->fetchassoc() ) {
        $ar[] = $rec;
    }
}

```

```

$response->getBody()->write( json_encode( $ar ) );

return $response
    ->withStatus(200)
    ->withHeader('Content-Type', 'application/json');
});

$app->post('/lora_status', function (Request $request, Response $response, $args) {
    $db = new DataBase();
    $log = new Logger();

    $log->log("chiamata lora_status");

    $sql = "SELECT dev_eui as devEui,
              _park_state as parkState,
              state changed,
              DATE_FORMAT(rx_time_received,'%d-%m-%y %H:%i:%s') lastUpdate
            FROM sogei.devices changes
            WHERE dev_eui='0080000004008872'";

    $db->query($sql);
    while ($rec = $db->fetchassoc()) {
        $ar[] = $rec;
    }

    $log->log(json_encode($ar));
    $response->getBody()->write(json_encode($ar));

    return $response
        ->withStatus(200)
        ->withHeader('Content-Type', 'application/json');
});

$app->post('/packet/send', function (Request $request, Response $response, $args) {

    putenv( 'DB_NAME=sogei' );

    $log = new Logger();
    $Utils = new Utils();

    $parsedBody = json_decode( $request->getBody(), true );

    $log->log( "Pacchetto: " . json_encode( $parsedBody, JSON_PRETTY_PRINT ) );

    // se il messaggio è downlink, risponde e lo scarta
    if( $parsedBody['method'] == "downlink" ) {
        $result = array(
            "jsonrpc" => "2.0",
            "id" => $parsedBody['id'],
            "result" => "ok"
        );

        $response->getBody()->write(json_encode($result));
        return $response
            ->withStatus(200)
            ->withHeader('Content-Type', 'application/json');
    }

    $ar = array(
        "id" => $parsedBody['id'],
        "dev_eui" => $parsedBody['params']['dev_eui'],
        "rx_time" => $parsedBody['params']['rx_time'],
        "rx_time_received" => date( "Y-m-d H:i:s", $parsedBody['params']['rx_time'] ),
        "payload" => json_encode( $parsedBody )
    );

    $log->log( "pacchetto data: " . json_encode( $ar ) );

```

```

// scorriamo l'array dei dati del pacchetto
$ar['temperature'] = -1;

foreach( $parsedBody['params']['parsed_payload']['payload'] as $data ) {
    switch( $data['type'] ) {
        case "park_status":
        case "keepalive":
            $ar['park_status'] = ($data['data'] == "free") ? 0 : 1;
            break;
        case "temperature":
            $ar['temperature_time'] = "now()";
            $ar['temperature'] = ($data['data'] > 0) ? $data['data'] : 0;
            break;
    }
}

$log->log( "pacchetto data 2: " . json_encode( $ar ) );

$ret = $Utils->dbSql( true, "sogei.uplink packet", $ar, "", "" );
$log->log( "ret: " . json_encode( $ret, JSON_PRETTY_PRINT ) );

if( $ret['success'] == 1 ) {
    $result = array(
        "jsonrpc" => "2.0",
        "id" => $parsedBody['id'],
        "result" => "ok"
    );

    // elabora lo stato attuale
    $recLast = last status get( $ar['id'] );
    //print_r( $recLast );
    $bInsert = true;
    if( $recLast['msgid'] == $ar['id'] )
        $bInsert = false;

    if( $recLast['park_status'] == $ar['park_status'] ) {
        $state_changed = 0;
    } else {
        $state_changed = 1;
    }

    //TODO: verificare lo stato precedente a questo
    $arUpdate = array(
        "msgid" => $ar['id'],
        "dev_eui" => $ar['dev_eui'],
        "rx time received" => $ar['rx time received'],
        "park_status" => $ar['park_status'],
        "temperature" => $ar['temperature'] == -1 ? -1 : $ar['temperature'],
        "state_changed" => $state_changed
    );

    if( $ar['temperature'] != -1 )
        $arUpdate['temperature time'] = "now()";

    $retChanges = $Utils->dbSql( $bInsert, "sogei.sogei changes", $arUpdate, "dev_eui",
    $ar['dev_eui'] );
    if( $retChanges['success'] == 0 )
        $log->log( "retChanges: " . json_encode( $retChanges, JSON_PRETTY_PRINT ) );

    //print_r( $retChanges );

    $response->getBody()->write(json_encode($result));
    return $response
        ->withStatus(200)
        ->withHeader('Content-Type', 'application/json');
} else {
    return $response
        ->withStatus(400);
}

```



```

}

});

$app->get('/data_check', function (Request $request, Response $response, $args) {
    $log = new Logger();
    $db = new DataBase();

    $data = [
        ["id":"3A3789B6-E4F6-9081-692E-6D84CE8FE872","rssi":-
        73,"advertising":{"kCBAdvDataLocalName":"PNI-
        DFU","kCBAdvDataManufacturerData":[],"kCBAdvDataTxPowerLevel":-
        11,"kCBAdvDataIsConnectable":1},"name":"PNI-
        DFU","mfgData":[4,2,6,5,2,11,0,128,0,0,4,0,151,56],"status":11,"statusa":"b","dataParsed":"00800
        00004009738","dataParsedRaw":"04020605020b0080000004009738"}];
    $dataj = json_decode( $data, true );
    $sql = sprintf( "select * from sogei.devices changes WHERE dev eui='%s'",
    $dataj[0]['dataParsed'] );
    //echo $sql;

    $db->query( $sql );
    while( $rec = $db->fetchassoc() ) {
        $ar[] = $rec;
    }

    //where Login time > date sub(now(), interval 3 minute) ;

    return $res;

?>

```

ДОДАТОК Б

ВІДГУК
керівника економічного розділу
на кваліфікаційну роботу магістра
на тему:
«Методи, алгоритми та інформаційна технологія покращення процесів
передачі даних радіосигналами»
студент групи 122м-19-1 Дмитровської Мар'яни Михайлівни

Керівник економічного розділу
доцент каф. ПЕП та ПУ, к.е.н.

Л. В. Касьяненко

ДОДАТОК В

ПЕРЕЛІК ДОКУМЕНТІВ НА ОПТИЧНОМУ НОСІЇ

Ім'я файла	Опис
Пояснювальні документи	
Диплом_Дмитровська.doc	Пояснювальна записка до магістерської роботи. Документ Word.
Диплом_Дмитровська.pdf	Пояснювальна записка до до магістерської роботи в форматі PDF
Програма	
Program.rar	Архів. Містить коди програми і откомпільовану програму
Презентація	
Презентація_Дмитровська.ppt	Презентація до магістерської роботи