

Міністерство освіти і науки України

Луцький національний технічний університет

(повне найменування вищого навчального закладу)

Факультет комп'ютерних та інформаційних технологій

(повне найменування факультету)

Кафедра електроніки та телекомунікацій

(повне найменування кафедри)

КВАЛІФІКАЦІЙНА РОБОТА ЗА СТУПЕНЕМ ВИЩОЇ ОСВІТИ «МАГІСТР»

РОЗРОБКА БЕЗДРОТОВОГО КОМУНІКАЦІЙНОГО ПРИСТРОЮ НА ОСНОВІ STM32 ТА LORA

DEVELOPMENT OF A WIRELESS COMMUNICATION DEVICE BASED ON STM32 AND LORA

спеціальність 172 Електронні комунікації та радіотехніка
(шифр і назва спеціальності)

освітня програма «Телекомунікації та радіотехніка»
(назва освітньої програми)

Виконав: здобувач вищої освіти
групи ЕКРМ-21
Кречик Андрій Петрович

(підпис)

Керівник: к.ф.-м.н., доцент
Хвищун Микола Вячеславович

(підпис)

Кваліфікаційну роботу
допущено до захисту
«__» грудня 2025 р.
Гарант освітньої програми:
к.т.н., доцент
Селепина Йосип Романович

(підпис)

Луцьк – 2025 року

ЛУЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних та інформаційних технологій
Кафедра електроніки та телекомунікацій
Ступінь вищої освіти: магістр
Галузь знань: 17 Електроніка та телекомунікації
Спеціальність: 172 Електронні комунікації та радіотехніка
Освітня програма: «Телекомунікації та радіотехніка»

ЗАТВЕРДЖУЮ
Завідувач кафедри

«__» _____ 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧУ ВИЩОЇ ОСВІТИ

Кречуку Андрію Петровичу

1. Тема кваліфікаційної роботи: *Розробка бездротового комунікаційного пристрою на основі STM32 та LoRa*

Керівник роботи: *к.ф.-м.н., доцент Хвищун Микола Вячеславович*

затверджені наказом закладу вищої освіти від «14» січня 2025 р. № 20/01-02

2. Строк подання здобувачем вищої освіти кваліфікаційної роботи: 05.12.2025 р.

3. Вихідні дані до роботи: *Інформаційні матеріали для мікроконтролера STM32, модуль LoRa, технічні характеристики периферійних інтерфейсів (UART, SPI, I²C, USB, CAN) та можливостей апаратної комутації на базі STM32. Структурований план розробки апаратної та програмної частини комутаційної системи.*

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити)

Вступ

Розділ 1 Аналіз вихідних даних та шляхи вирішення поставленого завдання

Розділ 2 Теоретична частина

Розділ 3 Технічне забезпечення вирішення поставленого завдання

Розділ 4 Спеціальна частина:прикладне програмне забезпечення

Висновки

5. Перелік графічного(ілюстративного) матеріалу

Презентація PowerPoint 12 слайдів.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис	
		завдання видав	завдання прийняв
<i>Аналіз вихідних даних та обґрунтування теми кваліфікаційної роботи магістра</i>	<i>Хвищун М. В., доцент</i>		
<i>Теоретична частина</i>	<i>Хвищун М. В., доцент</i>		
<i>Технічне забезпечення вирішення поставленого завдання</i>	<i>Хвищун М. В., доцент</i>		
<i>Спеціальна частина: прикладне програмне забезпечення</i>	<i>Хвищун М. В., доцент</i>		
<i>Висновки</i>	<i>Хвищун М. В., доцент</i>		
<i>Нормоконтроль</i>	<i>Селепина Й. Р., доцент</i>		
<i>Гарант ОП</i>	<i>Селепина Й. Р., доцент</i>		
<i>Показник запозичень тексту</i>		%	
<i>Академічна доброчесність</i>	<i>Хвищун М. В., доцент</i>		

7. Дата видачі завдання 03.02.2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1.	<i>Розділ 1 Аналіз вихідних даних та обґрунтування теми кваліфікаційної роботи магістра</i>	до 18.09.2025 р.	
2.	<i>Розділ 2 Теоретична частина</i>	до 26.10.2025 р.	
3.	<i>Розділ 3 Технічне забезпечення вирішення поставленого завдання</i>	до 02.11.2025 р.	
4.	<i>Розділ 4 Спеціальна частина:прикладне програмне забезпечення</i>	до 09.11.2025 р.	
5.	<i>Висновки</i>	до 16.11.2025 р.	
6.	<i>Формування списку використаних джерел</i>	до 23.11.2025 р.	
7.	<i>Оформлення ілюстративного матеріалу</i>	до 28.11.2025 р.	
8.	<i>Нормоконтроль</i>	до 01.12.2025 р.	
9.	<i>Інструментальна перевірка на академічний плагіат</i>	до 05.12.2025 р.	
10.	<i>Представлення кваліфікаційної роботи магістра до захисту</i>	до 30.12.2025 р.	

Здобувач вищої освіти

_____ (підпис)

Кречик А. П.

_____ (прізвище, ініціали)

Керівник кваліфікаційної роботи

_____ (підпис)

Хвищун М. В.

_____ (прізвище, ініціали)

АНОТАЦІЯ

Кречик А. П. Розробка бездротового комунікаційного пристрою на основі STM32 та LoRa. Рукопис.

Кваліфікаційна робота магістра ОП «Телекомунікації та радіотехніка» спеціальності 172 Електронні комунікації та радіотехніка. Луцький національний технічний університет. Луцьк, 2025. 84 с.

Кваліфікаційна робота магістра складається з вступу, чотирьох розділів, висновків, переліку використаних джерел та додатків.

У роботі розглянуто питання створення бездротового комунікаційного пристрою для передавання даних на великі відстані з використанням мікроконтролера STM32 та технології LoRa. Проведено аналіз сучасних методів організації низькопотужних мереж IoT, розглянуто принципи функціонування LoRa та LoRaWAN, досліджено параметри радіомодулів SX1276/78 та їх сумісність із сімейством STM32.

У дослідженні виконано обґрунтований вибір апаратної платформи STM32 та радіомодуля LoRa, розроблено структурну та функціональну схеми пристрою, визначено особливості реалізації інтерфейсів SPI, UART та енергозберігаючих режимів мікроконтролера. За допомогою STM32CubeMX та STM32CubeIDE здійснено проєктування та програмну реалізацію протоколу бездротового обміну даними. Проведено моделювання та експериментальне тестування пристрою, оцінено радіус зв'язку, енергоспоживання, стійкість до завад та якість передавання даних у різних умовах середовища.

Подано рекомендації щодо інтеграції розробленого пристрою в телекомунікаційні системи IoT, оптимізації енергоспоживання та підвищення надійності зв'язку.

Ключові слова: STM32, LoRa, SX1276, бездротовий зв'язок, IoT, мікроконтролер, SPI, радіоканал, енергозбереження, STM32CubeIDE.

ANNOTATION

Krechyk A. Development of a Wireless Communication Device Based on STM32 and LoRa. Manuscript. Master's qualification thesis of the educational program «Telecommunications and radio engineering» in specialty 172 Electronic communications and radio engineering. Lutsk National Technical University. Lutsk, 2025. 84 p.

The master's qualification thesis consists of an introduction, four chapters, conclusions, a list of references, and appendices. The work addresses the development of a wireless communication device designed for long-distance data transmission using an STM32 microcontroller and LoRa technology. An analysis of modern approaches to building low-power IoT networks is conducted; the operating principles of LoRa and LoRaWAN are examined; and the technical parameters of SX1276/78 radio modules and their compatibility with STM32 microcontroller families are investigated.

The research includes a well-grounded selection of the STM32 hardware platform and the LoRa radio module. Structural and functional diagrams of the device are developed, and specific features of implementing SPI, UART interfaces and low-power modes of the microcontroller are defined. Using STM32CubeMX and STM32CubeIDE, the design and software implementation of the wireless communication protocol were carried out. System modeling and experimental testing were conducted, encompassing evaluations of communication range, energy consumption, interference immunity, and data transmission quality under various environmental conditions.

Recommendations are provided for integrating the developed device into IoT telecommunications systems, optimizing energy consumption, and enhancing communication reliability.

Keywords: STM32, LoRa, SX1276, Wireless Communication, IoT, Microcontroller, SPI, Radio Channel, Low-Power Operation, STM32CubeIDE.

ЗМІСТ

ВСТУП.....	7
РОЗДІЛ 1 АНАЛІЗ ВИХІДНИХ ДАНИХ ТА ШЛЯХИ ВИРІШЕННЯ ПОСТАВЛЕНОГО ЗАВДАННЯ	9
1.1 Огляд технологій бездротової передачі даних (Wi-Fi, BLE, ZigBee, LoRa, NB-IoT).....	9
1.2 Огляд мікроконтролерів STM32 для систем зв'язку та аналіз їх переваг	16
1.3 Формування технічного завдання на розробку пристрою та загальні висновки до розділу	19
РОЗДІЛ 2 ТЕОРЕТИЧНА ЧАСТИНА.....	23
2.1 Принцип дії LoRa-модуля (SX1276/78): модуляція Chirp Spread Spectrum, параметри зв'язку	23
2.2 Архітектура та структура мережі LoRaWAN.....	30
2.3 Основи проектування приймально – передавальних систем на базі STM32.....	34
РОЗДІЛ 3 ТЕХНІЧНЕ ЗАБЕЗПЕЧЕННЯ ВИРІШЕННЯ ПОСТАВЛЕНОГО ЗАВДАННЯ.....	41
3.1 Розробка структурної та функціональної схеми пристрою.....	41
3.2 Обґрунтування вибору елементної бази	44
3.3 Розробка принципової електричної схеми та її опис.....	47
РОЗДІЛ 4 СПЕЦІАЛЬНА ЧАСТИНА: ПРИКЛАДНЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ.....	52
4.1 Алгоритм взаємодії мікроконтролера STM32 з LoRa – модулем	52
4.2 Розробка та опис прошивки на мові C у середовищі CubeIDE....	58
4.3 Тестування та налагодження системи зв'язку, аналіз стабільності канал	62
ВИСНОВКИ	65
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	66
ДОДАТКИ.....	68

ВСТУП

У сучасному світі, де технології Інтернету речей (IoT) активно інтегруються у всі сфери діяльності людини, особливого значення набуває ефективно та надійне передавання даних між віддаленими вузлами мереж. З кожним роком кількість сенсорних пристроїв, систем моніторингу та автоматизованих комплексів невідмінно зростає, що породжує високі вимоги до їх автономності, енергоефективності та дальності зв'язку. Традиційні бездротові технології, такі як Wi-Fi або Bluetooth, у багатьох випадках не забезпечують необхідного радіусу дії чи тривалої роботи від автономних джерел живлення. Тому зростає потреба у застосуванні інноваційних низькопотужних радіотехнологій, здатних забезпечити стабільний зв'язок на значні відстані без суттєвих енерговитрат.

Однією з таких технологій є LoRa (Long Range), яка дозволяє здійснювати передачу даних на кілька кілометрів навіть у складних міських умовах. Висока завадостійкість та мінімальне енергоспоживання роблять LoRa одним із ключових рішень для сучасних IoT-систем. У поєднанні з продуктивними та енергоефективними мікроконтролерами сімейства STM32 технологія LoRa забезпечує можливість створення компактних, автономних та надійних вузлів збору інформації, придатних для використання в екологічному моніторингу, системах «розумного міста», аграрних IoT-рішеннях, мережах промислового контролю тощо.

Актуальність теми полягає у потребі розробки доступного, гнучкого та енергоощадного бездротового комунікаційного пристрою, здатного забезпечувати стабільний зв'язок на значні відстані та інтегруватися в сучасні системи IoT. Використання мікроконтролерів STM32 у поєднанні з технологією LoRa дає змогу створити апаратно-програмний комплекс, який відповідає сучасним вимогам до ефективності, надійності та масштабованості.

Метою кваліфікаційної роботи є розробка, моделювання та дослідження бездротового комунікаційного пристрою на основі мікроконтролера STM32 і модуля LoRa SX1278, здатного забезпечувати надійну передачу даних при

мінімальному енергоспоживанні.

Об'єктом дослідження є бездротові сенсорні системи передавання даних у мережах Інтернету речей.

Предметом дослідження: технології та методи побудови апаратно – програмних засобів комунікаційного пристрою на основі STM32 та LoRa.

Для досягнення мети були поставлені наступні завдання:

– провести аналіз сучасних технологій бездротового зв'язку та обґрунтувати вибір LoRa для енергоефективної передачі даних;

– дослідити архітектуру мікроконтролерів STM32 та принципи роботи модуля LoRa SX1278;

– розробити структурну та принципову електричну схему комунікаційного пристрою на базі програми Proteus 8;

– створити програмне забезпечення для організації роботи бездротового обміну даними між STM32, LoRa та давачами з використанням програми STM32CubeIDE;

– оцінити стабільність зв'язку досліджуваної системи.

Наукова новизна кваліфікаційної роботи полягає в тому, що було запропоновано нову архітектуру бездротового комунікаційного вузла на основі мікроконтролера STM32 та модуля LoRa SX1276/78, що поєднує енергоефективні режими мікроконтролера з адаптивним управлінням параметрами радіоканалу, створено прототип універсального комунікаційного модуля, який може бути інтегрований у системи моніторингу навколишнього середовища, «розумного міста» та промислового IoT, забезпечуючи стабільний зв'язок на великих відстанях із мінімальними вимогами до живлення.

Окремі результати роботи були представлені у тезах доповіді (Додаток А): Хвищун М. В., Кречик А. П., Хвищун Д. М., Рубльов В. В. Розроблення бездротового комунікаційного пристрою на основі STM32 та LoRa. International Scientific and Practical Conference of Young Scientists and Students «Actual Problems of Automation and Control», №13. 27 листопада 2025. м. Луцьк. С.137-142.

РОЗДІЛ 1

АНАЛІЗ ВИХІДНИХ ДАНИХ ТА ШЛЯХИ ВИРІШЕННЯ ПОСТАВЛЕНОГО ЗАВДАННЯ

1.1 Огляд технологій бездротової передачі даних (Wi-Fi, BLE, ZigBee, LoRa, NB-IoT)

Розвиток бездротових технологій став основою сучасних систем зв'язку, моніторингу та автоматизації. Технології Wi-Fi, Bluetooth Low Energy (BLE), ZigBee, LoRa і NB-IoT забезпечують обмін даними між пристроями з різними вимогами до швидкості, енергоспоживання, радіусу дії та пропускної здатності. Вибір конкретної технології визначається умовами застосування – від локальних IoT-вузлів до глобальних телеметричних систем.

Сучасні системи зв'язку, зокрема в галузі Інтернету речей (IoT), побудовані на бездротових технологіях, що дозволяють передавати дані між пристроями без фізичного з'єднання. Розвиток таких технологій, як Wi-Fi, Bluetooth Low Energy (BLE), ZigBee, LoRa та NB-IoT, відкрив нові можливості для створення автономних сенсорних систем, розумних міст, промислового моніторингу й автоматизації процесів.

Кожна з цих технологій має власну архітектуру, енергетичну ефективність, швидкість і дальність передачі, що визначає сферу її застосування. Нижче подано детальний аналіз і порівняння основних характеристик.

Wi-Fi (Wireless Fidelity) є найпоширенішою технологією бездротової передачі даних, стандартизованою як IEEE 802.11. Вона забезпечує високошвидкісний обмін даними в локальних мережах (LAN). Переваги: велика швидкість (до 1 Гбіт/с у стандарті Wi-Fi 6), сумісність з існуючою інфраструктурою, простота інтеграції. Недоліки: високе енергоспоживання, обмежена дальність (до 100 м у приміщеннях), необхідність у точці доступу. Застосування: підключення комп'ютерів, смартфонів, IoT-пристроїв із живленням від мережі.

Bluetooth Low Energy (BLE) – це енергоефективна версія Bluetooth,

призначена для обміну невеликими обсягами даних між пристроями. Переваги: дуже низьке споживання енергії, підтримка одночасного підключення декількох пристроїв, висока надійність. Недоліки: невелика дальність дії (до 50 м), низька пропускну здатність. Застосування: носимі пристрої, медичні датчики, трекери, побутові прилади.

ZigBee базується на стандарті IEEE 802.15.4 і призначена для створення мереж із топологією mesh. Переваги: висока стабільність зв'язку завдяки маршрутизації пакетів, низьке енергоспоживання, простота масштабування мережі. Недоліки: мала швидкість (до 250 кбіт/с), обмежена дальність (до 100 м). Застосування: системи «розумного дому», автоматизація освітлення, бездротові сенсорні мережі.

LoRa (Long Range) – це технологія передачі даних великої дальності з низькою швидкістю, яка використовує модуляцію Chirp Spread Spectrum (CSS). Переваги: радіус дії до 10...15 км, надзвичайно низьке споживання енергії, висока стійкість до завад. Недоліки: мала швидкість передачі (0,3...50 кбіт/с), затримки при великих обсягах даних. Застосування: сенсорні системи моніторингу, агротехніка, «розумне місто», енергетика.

NB-IoT (Narrow Band Internet of Things) – стільникова технологія вузької смуги пропускання, що працює у ліцензованому спектрі операторів мобільного зв'язку. Переваги: покриття до 15 км, підтримка тисяч пристроїв на одній базовій станції, інтеграція в існуючу LTE-мережу. Недоліки: потреба в SIM-карті, залежність від оператора, середнє енергоспоживання. Застосування: розумні лічильники, системи безпеки, транспортний моніторинг [1].

У таблиці 1.1 подано порівняльний аналіз вище наведених технологій за основними технічними характеристиками.

Порівняння бездротових технологій показує, що кожна з них оптимізована під свої завдання. Wi-Fi забезпечує дуже високу швидкість, але має велике енергоспоживання й обмежену дальність, тому використовується для локальних мереж. BLE та ZigBee орієнтовані на малопотужні пристрої: BLE – для коротких дистанцій і носимих гаджетів, ZigBee – для автоматизації завдяки підтримці mesh-мереж. LoRa виділяється надзвичайно великою

дальністю дії та мінімальним енергоспоживанням, що робить її ідеальною для IoT-моніторингу. NB-IoT працює у стільникових мережах, забезпечує найкраще покриття та підходить для масового підключення сенсорів і смарт – лічильників.

Таблиця 1.1 – Порівняльний аналіз технологій за основними технічними характеристиками [2]

Параметр / технологія	Wi-Fi	BLE	ZigBee	LoRa	NB-IoT
Дальність дії, м	50...100	10...50	10...100	1 000...10 000	1 000...15 000
Швидкість передачі, кбіт/с	10 000...100 000	125...2 000	до 250	0,3...50	до 250
Енергоспоживання	Високе	Дуже низьке	Низьке	Дуже низьке	Середнє
Частотний діапазон, МГц	2 400 / 5 000	2 400	2 400 / 868	433 / 868 / 915	700...2 100
Топологія мережі	Точка – багато точок	Зірка	Mesh	Зірка	Стільникова
Типові застосування	Локальні мережі	Носимі пристрої	Автоматизація	IoT/Моніторинг	Smart Metering

На рисунку 1.1 подано діаграми порівняння технологій енергоспоживання (відносно Wi-Fi).

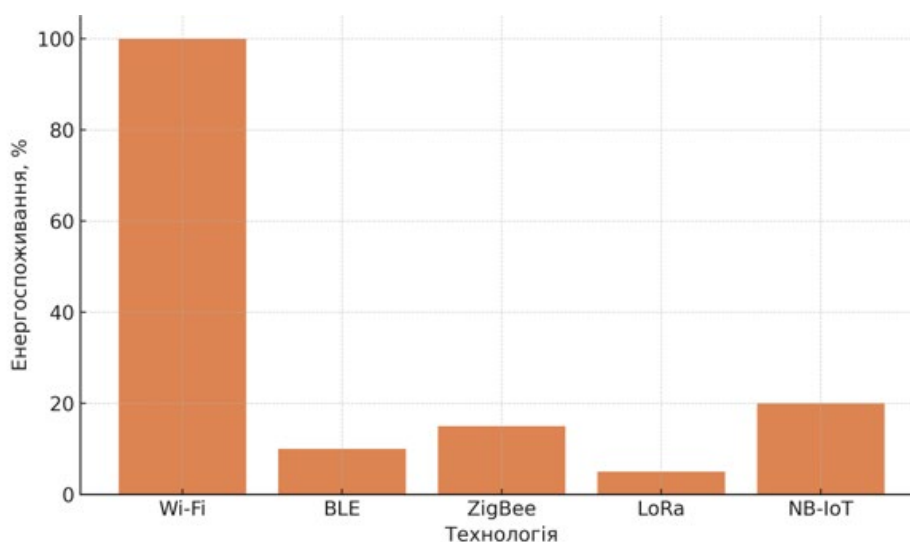


Рисунок 1.1 – Порівняння енергоспоживання різних технологій [2]

За результатами, поданими на рисунку 1.1 видно, що:

1. Wi-Fi має найвище енергоспоживання – приблизно у 5...10 разів більше, ніж інші бездротові технології, тому менш придатний для автономних

IoT-пристроїв на батареї.

2. BLE (Bluetooth Low Energy) демонструє одне з найнижчих значень енергоспоживання, що робить його оптимальним для короткодистанційних сенсорних мереж і носимих пристроїв.

3. ZigBee має дещо більше споживання, ніж BLE, але все ще залишається енергоефективною технологією для мереж з великою кількістю вузлів.

4. LoRa характеризується мінімальним енергоспоживанням, що у поєднанні з великим радіусом дії робить її ідеальною для автономних IoT-систем тривалого використання.

5. NB-IoT споживає більше енергії, ніж ZigBee і LoRa, але значно менше Wi-Fi, забезпечуючи при цьому широке покриття та високу надійність зв'язку.

Отже, технології LoRa та BLE є найбільш енергоефективними для IoT-пристроїв, тоді як Wi-Fi – найбільш енерговитратною опцією. NB-IoT і ZigBee займають проміжні позиції з прийнятним рівнем споживання та добрим покриттям.

На рисунку 1.2 подано діаграми порівняння радіусу дії технологій.

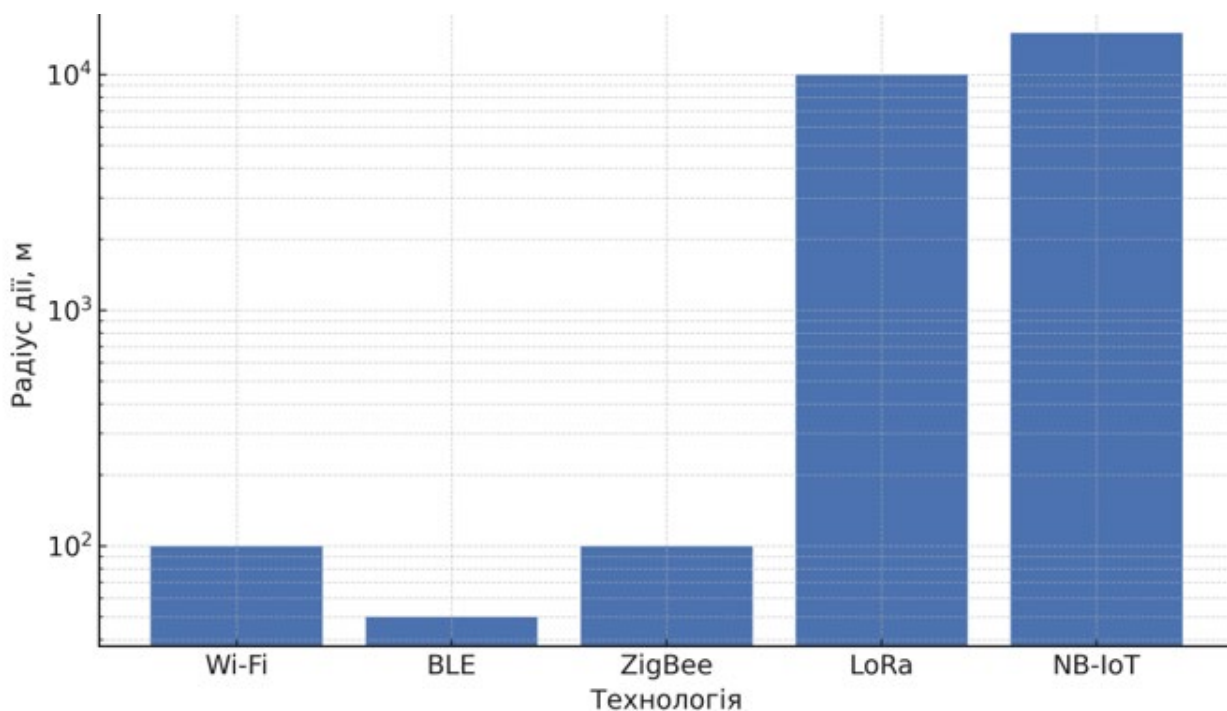


Рисунок 1.2 – Порівняння радіусу дії технологій [3]

Згідно результатів, поданих на рисунку 1.2 бачимо, що:

1. Wi-Fi, BLE та ZigBee мають відносно малий радіус дії – приблизно 10...100 м, що обмежує їх використання локальними мережами в межах приміщень або невеликих територій.

2. LoRa забезпечує радіус дії до ~10 км, що робить її однією з найкращих технологій для розгортання великих IoT-мереж на відкритій місцевості (сільськогосподарський моніторинг, міська інфраструктура).

3. NB-IoT має найбільший радіус дії – понад 10 км, завдяки використанню стільникової інфраструктури, що забезпечує надійне покриття навіть у важкодоступних місцях.

Отже, для локальних сценаріїв (квартира, офіс, будівля) достатньо Wi-Fi, BLE та ZigBee. Для великих територій, польових умов і розподілених IoT-систем оптимальними є LoRa та NB-IoT, які забезпечують на порядок більший радіус дії.

На рисунку 1.3 подано діаграми порівняння швидкості передачі даних для технологій.

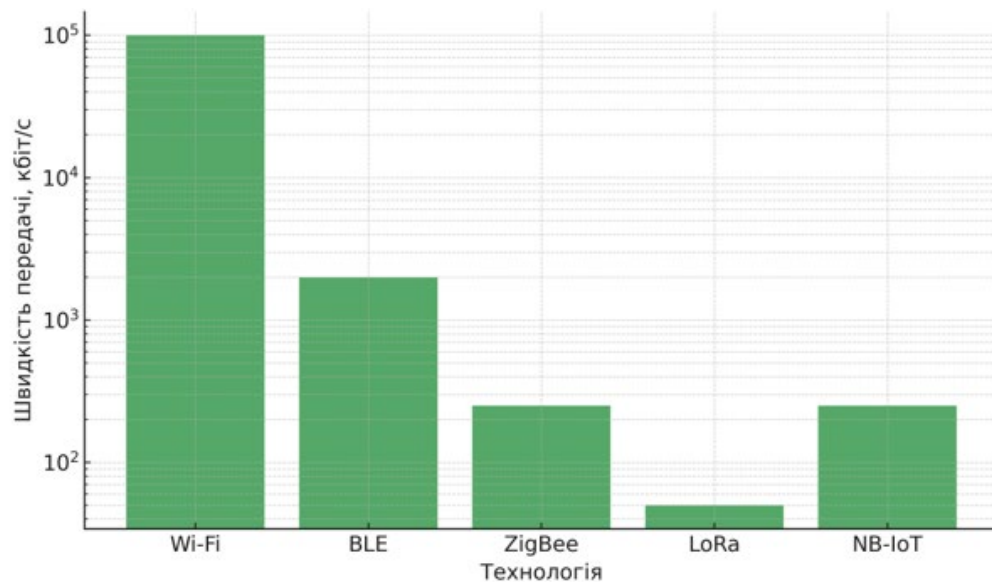


Рисунок 1.3 – Порівняння швидкості передачі даних для технологій [2, 3]

Порівнюючи результати, подані на рисунку 1.3 сформулюємо висновки щодо швидкості передачі даних для технологій:

1. Wi-Fi має найбільшу швидкість – понад 100 000 кбіт/с, що робить його найкращим варіантом для передавання великих обсягів даних, мультимедіа,

відео та високошвидкісного доступу.

2. BLE забезпечує середню швидкість (~2000 кбіт/с), якої достатньо для сенсорних IoT-додатків, передачі телеметрії та взаємодії з мобільними пристроями.

3. ZigBee (~250 кбіт/с) підходить для мереж з великою кількістю пристроїв, де важлива не швидкість, а стабільність та мале енергоспоживання.

4. NB-IoT (~200 кбіт/с) має обмежену, але стабільну швидкість, оптимізовану для масових IoT-мереж зі складною інфраструктурою та високою проникністю сигналу (підвали, шахти).

5. LoRa має найнижчу швидкість (~50 кбіт/с і нижче), оскільки її головна мета – далекодістанційна передача невеликих пакетів даних із мінімальним енергоспоживанням.

Отже, Wi-Fi – лідер за швидкістю, BLE та ZigBee – середньошвидкісні рішення для локальних IoT-мереж, NB-IoT – для розподілених систем з помірною швидкістю, а LoRa – для передачі малих пакетів на дуже великі відстані з акцентом на енергоефективність.

На основі приведених даних можемо робити наступні висновки:

1. Кожна технологія має свою нішу застосування.
2. Wi-Fi переважає за швидкістю, проте не придатний для автономних пристроїв.

3. BLE і ZigBee є ефективними у побутових короткодіапазонних системах.

4. LoRa і NB-IoT забезпечують баланс між енергоспоживанням та дальністю, тому оптимальні для IoT-рішень великого радіусу дії.

5. Для розробки енергоефективних бездротових сенсорних систем на базі STM32 доцільно використовувати LoRa, яка поєднує високу дальність зв'язку, низьку вартість і простоту реалізації.

У таблиці 1.2 подано технічні характеристики технологій та типові приклади застосування.

На основі приведених даних проведемо порівняльний аналіз основних технічних параметрів. Як бачимо, LoRa та NB-IoT демонструють найбільший

радіус дії – понад 10 км у відкритій місцевості, що у 100 разів перевищує можливості Wi-Fi чи BLE.

Таблиця 1.2 – Типові приклади застосування та технічні характеристики технологій [3]

Технологія	Діапазон частот, МГц	Радіус дії, м	Швидкість передачі, кбіт/с	Енергоспоживання	Топологія мережі	Типові застосування
Wi-Fi (802.11n/ac)	2400 / 5000	50...100	до 100 000	Високе	Точка-багато точок	Передача мультимедіа, доступ до Інтернету
BLE (Bluetooth Low Energy)	2400	10...50	125...2000	Низьке	Зірка	Носимі пристрої, медичні сенсори
ZigBee (802.15.4)	2400 / 868	10...100	до 250	Низьке	Mesh / Зірка	Автоматизація, «розумний дім»
LoRa (Long Range)	433 / 868 / 915	1000...10 000	0,3...50	Дуже низьке	Зірка	Сенсорні мережі, моніторинг середовища
NB-IoT (Narrow Band IoT)	700...2100	1000...15 000	до 250	Низьке	Стільникова	Розумні лічильники, трекінг, промисловість

Це забезпечується завдяки використанню вузькосмугової або розширеної модуляції (Chirp Spread Spectrum). Щодо енергоспоживання, то найнижче енергоспоживання спостерігається у LoRa та BLE. LoRa-пристрій може працювати від батареї понад 6 місяців, тоді як Wi-Fi споживає у 50...100 разів більше енергії через постійне утримання з'єднання. Для пропускної здатності, то Wi-Fi забезпечує найвищу швидкість – до 100 Мбіт/с, що підходить для мультимедіа, але не для енергоефективних IoT-мереж. LoRa та ZigBee мають обмежену пропускну здатність (до 50 кбіт/с), проте цього достатньо для передачі телеметричних даних. Розглядаючи топологію мереж, то ZigBee підтримує топологію mesh, що забезпечує гнучкість мережі. LoRaWAN та NB-IoT працюють за принципом «зірка-шлюз», що спрощує управління вузлами і мінімізує витрати енергії.

Отже, на основі порівняльного аналізу літературних джерел, можна

зробити наступні висновки:

1. Технології LoRa та NB-IoT мають найкраще співвідношення між дальністю зв'язку та енергоспоживанням, що робить їх пріоритетними для IoT-рішень і віддаленого моніторингу.
2. BLE доцільно використовувати у короткодіапазонних пристроях з низьким споживанням (носимі гаджети, медичні сенсори).
3. ZigBee залишається зручним рішенням для мереж середнього масштабу у системах автоматизації.
4. Wi-Fi залишається ефективним у високошвидкісних додатках, але не придатним для автономних вузлів.
5. Для створення енергоефективних мереж сенсорів оптимальною є комбінація STM32 + LoRa, що дозволяє забезпечити стабільний обмін даними на відстані до 10 км при мінімальних витратах енергії.

1.2 Огляд мікроконтролерів STM32 для систем зв'язку та аналіз їх переваг

Мікроконтролери серії STM32 компанії STMicroelectronics є одними з найпоширеніших рішень у галузі вбудованих систем завдяки поєднанню високої продуктивності, енергоефективності та гнучких можливостей для реалізації комунікаційних протоколів. У системах зв'язку вони використовуються як базові обчислювальні ядра для обробки сигналів, керування радіомодулями (LoRa, Wi-Fi, BLE, ZigBee, NB-IoT), формування пакетів даних та інтеграції з периферією. Розглянемо типові серії даних мікроконтролерів, які виробляються, та їх основні технічні характеристики, що подано в таблиці 1.3.

На основі даних таблиці, перерахуємо переваги мікроконтролерів STM32 у системах зв'язку:

1. Широкий вибір інтегрованих інтерфейсів UART, SPI, I²C, CAN, USB, Ethernet, USART, SDIO – дають змогу реалізувати взаємодію з будь-якими бездротовими модулями.

Таблиця 1.3 – Порівняльна характеристика серій мікроконтролерів STM32 за їх основними характеристиками[4]

Серія STM32	Ядро	Тактова частота, МГц	Пам'ять Flash / RAM	Особливості	Типові застосування
STM32F0	Cortex-M0	48	16...256 кБ / 4...32 кБ	Базова функціональність, низька вартість	Просте керування, базові пристрої
STM32F1	Cortex-M3	72	64...512 кБ / 20...64 кБ	Класична універсальна серія	Промислові контролери, автоматика
STM32F3	Cortex-M4	72–100	64...512 кБ / 32...80 кБ	DSP-розширення, АЦП 12–16 біт	Обробка сигналів, системи вимірювання
STM32F4	Cortex-M4	до 180	512 кБ...2 МБ / 64...256 кБ	FPU, DMA, підтримка Ethernet/USB	Комунікаційні шлюзи, обробка даних
STM32L4 / L0	Cortex-M4 / M0+	32–80	64...1024 кБ / 20...128 кБ	Низьке енергоспоживання (L = Low Power)	ІоТ, портативна електроніка
STM32WB	Cortex-M4 + M0	64–100	256...1024 кБ / 64...256 кБ	Вбудований BLE/ZigBee модуль	Бездротові сенсорні системи
STM32WL	Cortex-M4 + LoRa модем	48–64	64...256 кБ / 16...64 кБ	Вбудований LoRa SX126x	ІоТ, LPWAN, моніторинг

2. Висока енергоефективність. Режими Sleep, Stop, Standby та Low Power Run знижують споживання енергії до одиниць мікроампер.

3. Підтримка комунікаційних стеків STMicroelectronics надає бібліотеки для LoRaWAN, BLE, ZigBee, USB і TCP/IP (через STM32Cube Middleware).

4. Розвинене середовище розробки. Використання безкоштовного STM32CubeIDE дозволяє швидко налаштовувати периферію, генерувати код і налагоджувати програми.

5. Висока точність АЦП / ЦАП забезпечує якісне вимірювання аналогових сигналів у сенсорних і радіосистемах.

6. Наявність серій із вбудованими радіомодулями (STM32WL, STM32WB) дає змогу скоротити кількість зовнішніх компонентів і зменшити розміри пристрою.

Розглянемо приклади застосування STM32 у бездротових системах, поданих у таблиці 1.4.

Таблиця 1.4 – Застосування STM32 у бездротових системах [4, 5]

Пристрій / система	Тип мікроконтролера	Комунікаційний модуль	Призначення
Сенсорна станція моніторингу довкілля	STM32L0	LoRa SX1278	Передача даних про температуру, вологість, тиск
Розумний лічильник енергії	STM32F3	NB-IoT модуль Quectel BC95	Віддалена передача показників
BLE-маячок (beacon)	STM32WB	Вбудований BLE	Локальне позиціонування, навігація
ІоТ-шлюз	STM32F4	Ethernet + Wi-Fi ESP32	Збір і передача даних з вузлів у хмару
Промисловий контролер	STM32F1	ZigBee	Автоматизація технологічних процесів

Розглянемо діаграму енергоефективності серій STM32, поданих на рисунку 1.4.

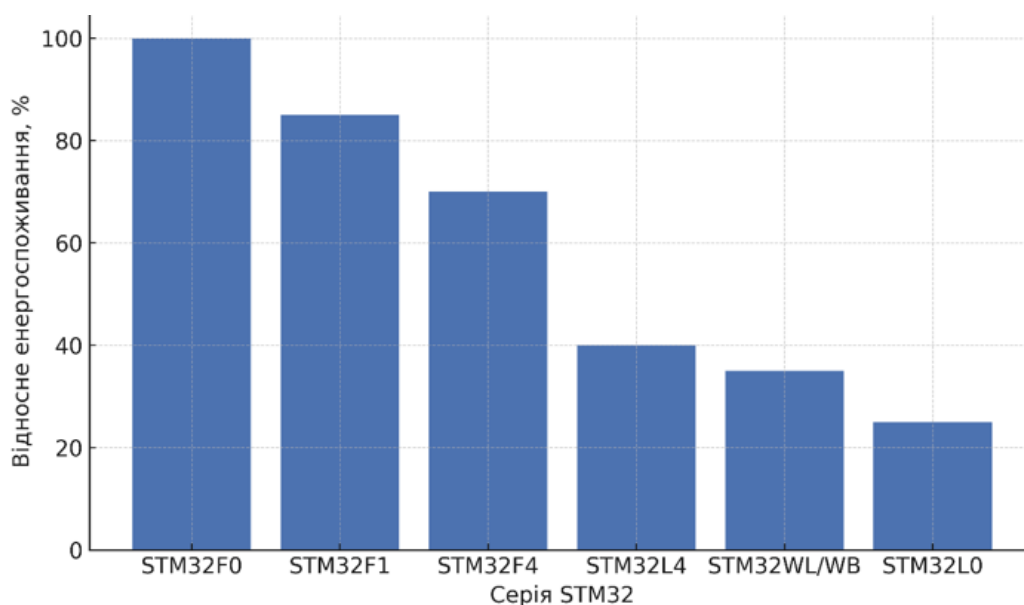


Рисунок 1.4 – Діаграма енергоефективності серій STM32 [5]

Дана діаграма відображає відносне енергоспоживання різних серій STM32 (F0, F1, F4, L4, WL / WB, L0), що демонструє переваги low-power лінійок STM32L і STM32WL / WB для систем зв'язку та ІоТ.

Отже, на основі вище поданої інформації сформулюємо наступні

ВИСНОВКИ:

1. Сімейство STM32 охоплює всі потреби сучасних систем зв'язку – від малопотужних сенсорів до комунікаційних шлюзів.

2. Найоптимальнішими для бездротових IoT-застосувань є STM32L0, STM32L4 (через енергоефективність) та STM32WL / WB (з інтегрованими модулями LoRa / BLE).

3. Використання STM32 у поєднанні з LoRa або NB-IoT дозволяє будувати енергоефективні, стабільні та масштабовані системи передачі даних.

1.3 Формування технічного завдання на розробку пристрою та загальні висновки до розділу

Розроблюваний пристрій призначений для бездротової передачі даних між віддаленими вузлами системи моніторингу з використанням мікроконтролера STM32F103C8T6 та модуля LoRa SX1278 повинен забезпечувати збір даних від сенсорів навколишнього середовища (температура, вологість, тиск тощо) і передавання цих даних на центральний шлюз або сервер за допомогою протоколу LoRa.

Сформулюємо вимоги до функціональних можливостей пристрою.

1. Функції пристрою:

- зчитування аналогових або цифрових даних з давачів;
- обробка даних мікроконтролером STM32;
- передача даних по бездротовому каналу LoRa SX1278;
- індикація стану пристрою на LCD-дисплеї або через UART-термінал;
- передача службової інформації (рівень сигналу, стан батареї, ID вузла).

2. Режими роботи:

- активний режим: збір і передача даних з періодом 1...10 с;
- режим сну: мінімальне споживання енергії між сеансами передачі;
- діагностичний режим: тестування радіоканалу і відображення параметрів.

3. Вимоги до стабільності зв'язку:

- рівень сигналу (RSSI) не гірше -110 дБм;
- коефіцієнт помилок BER $\leq 10^{-3}$;
- стійкість до завад у смузі 433 / 868 МГц.

4. Вимоги до апаратної частини подамо у вигляді таблиці 1.5.

Подана таблиця демонструє, що система побудована на основі енергоефективного мікроконтролера STM32F103C8T6 та LoRa-модуля SX1278, що забезпечує далекодіапазонний зв'язок. Комплекс датчиків (HTS221, BME280, DS18B20) дозволяє здійснювати точний моніторинг параметрів середовища, а LCD1602A надає локальне відображення інформації. Автономність забезпечується Li-Ion акумулятором, а стабільну роботу – надійні стабілізатори напруги. Використання універсальних інтерфейсів SPI, UART, I²C спрощує інтеграцію компонентів. У сукупності це утворює компактний, надійний і функціональний вузол для IoT-моніторингу.

Таблиця 1.5 – Вимоги до апаратної частини

Компонент	Тип / модель	Призначення
Мікроконтролер	STM32F103C8T6	Центральний контролер обробки даних
Радіомодуль	LoRa SX1278 (433 / 868 МГц)	Передача даних
Датчики	HTS221 / BME280 / DS18B20	Вимірювання параметрів середовища
Дисплей	LCD1602A	Відображення даних
Джерело живлення	Li-Ion 3,7 В / 2200 мА·год	Автономна робота пристрою
Стабілізатор напруги	AMS1117-3,3 або MP1584	Перетворення живлення
Інтерфейси зв'язку	SPI, UART, I ² C	Обмін між компонентами
Корпус	Захищений IP54	Польові умови експлуатації

5. Вимоги до програмного забезпечення:

- мова програмування: C;
- середовище розробки: STM32CubeIDE (вигляд панелі середовища розробки подано на рисунку 1.5);

- інтерфейс зв'язку з LoRa: SPI;
- використання бібліотек HAL для роботи з периферією.

Алгоритм обробки даних включає:

- ініціалізацію модуля LoRa;

- формування пакета даних;
- контроль передачі / прийому;
- індикацію результату;
- перехід у режим зниженого енергоспоживання.

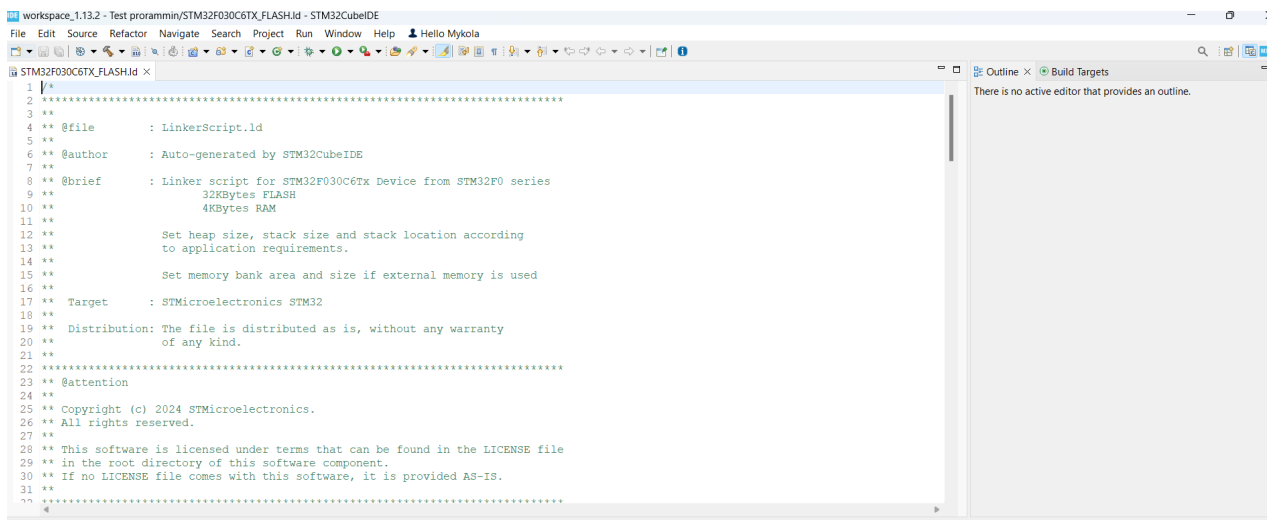


Рисунок 1.5 – Вигляд панелі середовища розробки STM32CubeIDE

За допомогою даної програми зручно налаштовувати роботу периферійних портів вводу / виводу мікроконтролерів серії STM32.

У результаті проведеного аналізу сучасних технологій бездротової передачі даних, мікроконтролерних платформ і технічних рішень для побудови енергоефективних систем зв'язку було отримано такі узагальнення:

1. Проведений огляд бездротових технологій (Wi-Fi, BLE, ZigBee, LoRa, NB-IoT) показав, що найоптимальнішим рішенням для реалізації автономних сенсорних систем із великим радіусом дії є технологія LoRa, яка забезпечує стабільний обмін даними на відстанях до 10...15 км при мінімальному енергоспоживанні.

2. Порівняльний аналіз основних бездротових технологій за критеріями енергоспоживання, швидкості передачі та дальності показав, що LoRa має найкраще співвідношення між споживанням енергії та дальністю зв'язку, поступаючись за швидкістю лише високошвидкісним технологіям типу Wi-Fi, які, однак, не є придатними для автономних систем.

3. У ході огляду мікроконтролерів STM32 встановлено, що сімейство

STM32 має широкі можливості для реалізації систем бездротового зв'язку завдяки:

- великій кількості вбудованих інтерфейсів (SPI, UART, I²C, CAN, USB, Ethernet);
- низькому енергоспоживанню та підтримці режимів енергозбереження (Sleep, Stop, Standby);
- наявності серій STM32WL / WB із інтегрованими радіомодулями LoRa та BLE;
- можливості використання середовища STM32CubeIDE для швидкої розробки та налагодження.

4. Розроблене технічне завдання визначає вимоги до апаратної та програмної частини пристрою, включаючи перелік компонентів (STM32F103C8T6, LoRa SX1278, датчики HTS221 / BME280, стабілізатор AMS1117), режими роботи, характеристики енергоспоживання, параметри стабільності зв'язку та умови експлуатації.

5. Визначено, що використання STM32 у поєднанні з LoRa дозволить створити енергоефективний комунікаційний пристрій, здатний забезпечити стабільний канал зв'язку на великій відстані, що є актуальним для систем моніторингу, автоматизації та промислових IoT-рішень.

6. Результати аналізу закладають основу для подальших етапів роботи – розробки структурної схеми, програмного забезпечення та прототипу пристрою, а також проведення експериментальних досліджень його характеристик.

РОЗДІЛ 2

ТЕОРЕТИЧНА ЧАСТИНА

2.1. Принцип дії LoRa-модуля (SX1276 / 78): модуляція Chirp Spread Spectrum, параметри зв'язку

Технологія LoRa (Long Range) є однією з ключових платформ для реалізації мереж LPWAN (Low Power Wide Area Network), призначених для передачі невеликих обсягів даних на великі відстані з мінімальним енергоспоживанням.

Дана технологія належить до класу енергоефективних радіотехнологій, призначених для побудови бездротових сенсорних мереж та розподілених систем моніторингу. Її відмінною рисою є застосування модуляції CSS (Chirp Spread Spectrum), яка забезпечує передачу вузькосмугових сигналів з високою завадостійкістю та великою дальністю зв'язку при мінімальній споживаній потужності. CSS є різновидом DSSS (Direct Sequence Spread Spectrum). У DSSS кожен символ розширюється псевдовипадковою послідовністю для збільшення спектра та покращення завадостійкості.

Модулі SX1276 / 78, розроблені компанією Semtech, реалізують фізичний рівень LoRa на основі модуляції Chirp Spread Spectrum (CSS), що забезпечує високий рівень завадостійкості та стабільний зв'язок навіть при низькому відношенні сигнал/шум (SNR до -20 дБ).

Розглянемо принцип дії модуля SX1276 / 78, який перетворює цифрові дані в радіосигнал шляхом модуляції, передає його по радіоканалу та приймає сигнали з інших пристроїв для подальшої демодуляції. Основу принципу дії становить Chirp Spread Spectrum (CSS) – метод розширення спектра, у якому кожен біт даних кодується у вигляді частотно – модульованого імпульсу (chirp), частота якого змінюється лінійно з часом.

На рисунку 2.1 подано зовнішній вигляд модуля SX1278 та його розпінування.

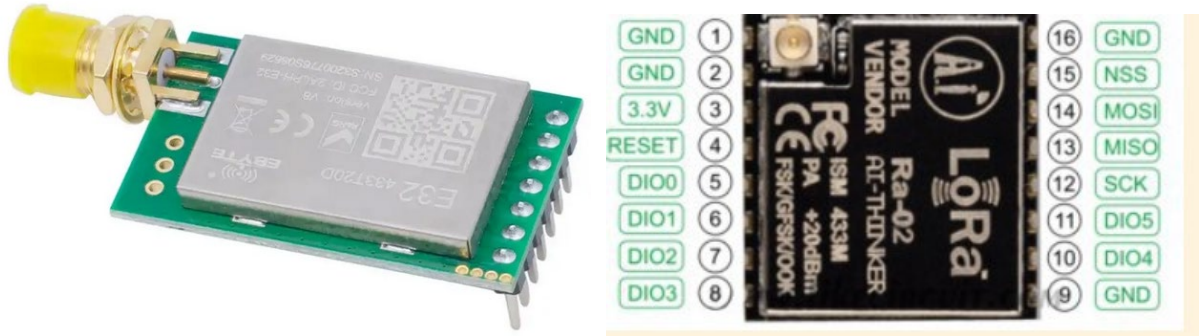


Рисунок 2.1 – Зовнішній вигляд модуля SX1278 та її розпінування

Основні етапи роботи модуля:

1. Передача даних: цифрова інформація із мікроконтролера (STM32) надходить через інтерфейс SPI до модуля SX1278.
2. Модуляція CSS: дані кодується у вигляді «чірпів» – сигналів із лінійною зміною частоти вгору або вниз у межах заданої смуги.
3. Розширення спектра: сигнал займає ширшу смугу (125...500 кГц), що підвищує стійкість до шумів і знижує вплив завад.
4. Передача по радіоканалу: модуль випромінює сигнал у діапазоні 433 / 868 / 915 МГц з потужністю до 20 дБм, де дБм – це логарифмічна одиниця потужності, відносно 1 мВт (20 дБм = 100 мВт).
5. Прийом: інший вузол синхронізується з «чірпами» передавача, демодулює їх і відновлює цифрові дані.

Модуляція Chirp Spread Spectrum (CSS) використовує зміну частоти з часом – «chirp», що поступово збільшується або зменшується. Кожен символ передається у вигляді одного або кількох чірпів, і залежно від параметрів модуляції тривалість сигналу та швидкість передачі можуть змінюватися.

Розглянемо вплив параметрів, що впливають на дальність і якість зв'язку:

1. Spreading Factor (SF): зі збільшенням SF покращується співвідношення сигнал / шум (SNR), що збільшує дальність зв'язку. Наприклад, SF=7 забезпечує передачу до 3 км, а SF=12 – до 10 км.
2. Bandwidth (BW): зменшення BW підвищує чутливість приймача, але зменшує швидкість передачі.
3. Coding Rate (CR): підвищення CR (наприклад, з 4 / 5 до 4 / 8) зменшує

кількість помилок, але збільшує тривалість передачі пакета.

4. Вихідна потужність: пряма залежність дальності від рівня потужності (до 20 дБм), проте підвищення потужності збільшує споживання енергії.

На рисунку 2.2. подано спрощену структурну схему LoRa-модуля SX1278.

Основними блоками є:

1. RF Front-End (Передавач / Приймач) – підсилювач, фільтр, мікшер, детектор.

2. Modem Core – реалізація CSS-модуляції / демодуляції, FEC, синхронізація, генерація preamble.

3. SPI Interface – обмін даними з мікроконтролером STM32.

4. Registers – конфігураційні регістри (SF, BW, CR, TxPower).

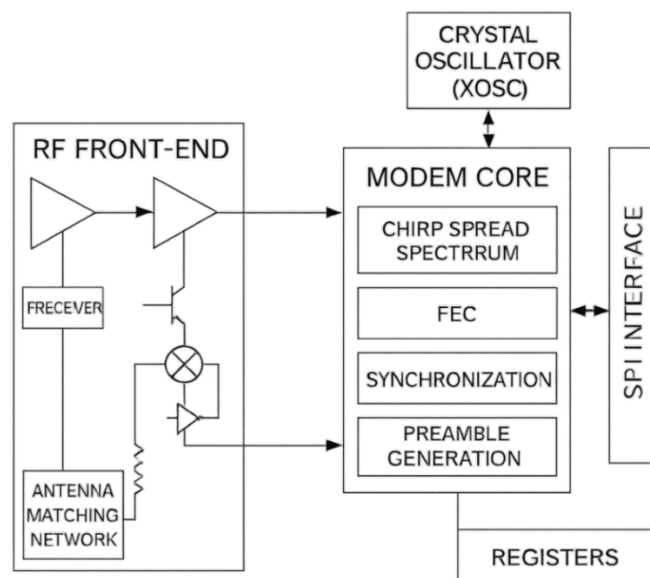


Рисунок 2.2 – Спрощена структурна схема LoRa-модуля SX1278

5. Crystal Oscillator (XOSC) – опорна частота 32 МГц.

6. Antenna Matching Network – узгодження антени для мінімальних втрат.

Основні параметри радіомодуля LoRa визначають баланс між дальністю зв'язку, швидкістю передавання даних та енергоспоживанням. Spreading Factor (SF) безпосередньо впливає на дальність: збільшення SF підвищує чутливість приймача, що дозволяє передавати дані на більші відстані, однак знижує

швидкість та збільшує час передачі кадру. Смуга пропускання (BW) задає пропускну здатність каналу: більша BW забезпечує вищий бітрейт, але зменшує чутливість приймача. Coding Rate (CR) підвищує стійкість до завад за рахунок додаткової корекції помилок, проте зменшує корисну пропускну здатність.

Робоча частота (F_c) визначає допустимі діапазони застосування та характеристики поширення хвиль: у нижчих діапазонах (433 МГц) дальність вища, але ширина каналу менша; у діапазоні 868 / 915 МГц можлива більша швидкість. Бітрейт (R_b) є похідним параметром, який формується комбінацією SF, BW та CR і визначає швидкість передачі у каналі.

Технологія LoRa базується на принципах розширення спектра (Spread Spectrum), зокрема на різновиді DSSS – CSS (Chirp Spread Spectrum). У традиційній DSSS кожен символ множиться на псевдовипадкову послідовність, що розширює спектр сигналу. Це демонструється на рисунку 2.3.

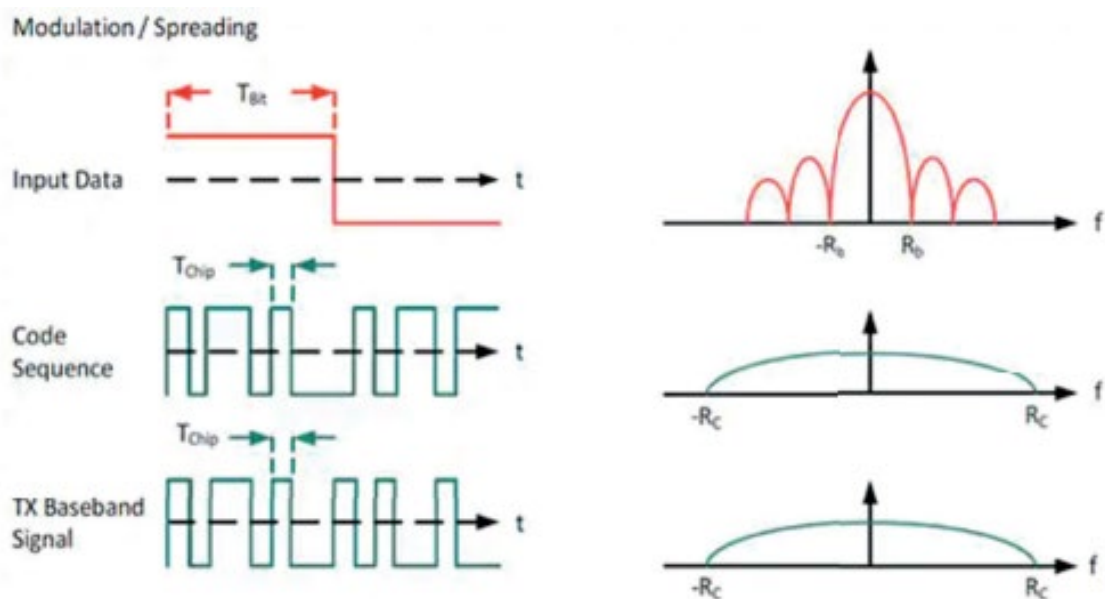


Рисунок 2.3 – Принцип DSSS-модуляції [4]

У модуляції CSS передавання інформації здійснюється за допомогою чірп-сигналів – сигналів з лінійною зміною частоти. На відміну від звичайних фазових або частотних модуляцій, CSS забезпечує можливість коректного прийому сигналу навіть при низькому відношенні сигнал/шум (до -20 dB).

Графічне представлення чірп-сигналу подано на рисунку 2.4.

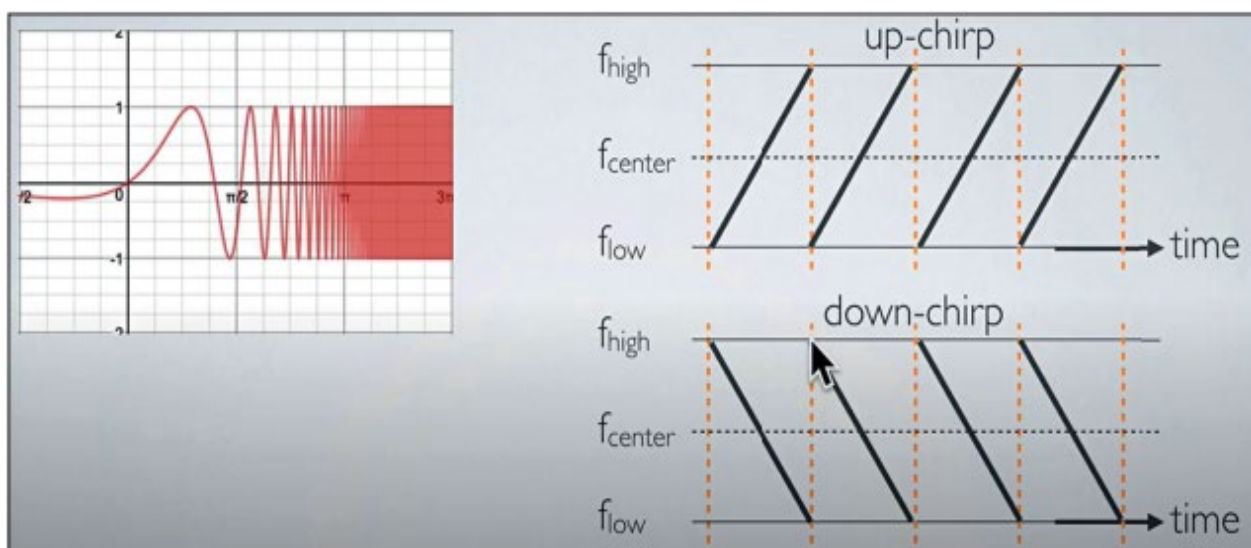


Рисунок 2.4 – Приклад чірп-сигналу (up-chirp / down-chirp) [4]

Ця особливість робить LoRa одним з найкращих виборів для енергоефективних IoT-вузлів.

У таблиці 2.1 подано основні параметри CSS.

Таблиця 2.1 – Основні параметри CSS

Параметр	Позначення	Типові значення	Опис
Spreading Factor	SF	7...12	Кількість бітів на символ; чим більший SF, тим більша дальність і менша швидкість
Bandwidth	BW	125 / 250 / 500 кГц	Смуга пропускання сигналу; більша BW → вища швидкість, але менша чутливість
Coding Rate	CR	4/5 ... 4/8	Коефіцієнт корекції помилок (FEC); зменшує BER
Carrier Frequency	Fc	433 / 868 / 915 МГц	Робоча частота модуля
Transmit Power	Pout	до +20 дБм	Вихідна потужність передавача
Sensitivity	–	до –148 дБм	Гранична чутливість приймача
Bitrate	Rb	0,3...50 кбіт/с	Залежить від SF, BW і CR

Основні параметри модуля SX1278 подано в таблиці 2.2.

Таблиця 2.2 – Основні параметри модуля SX1278 [5]

Параметр	Значення
Діапазон частот	137...1020 МГц
Потужність передавача	до +20 дБм
Чутливість приймача	до -148 дБм
Максимальна швидкість	50 кбіт/с
Робоча напруга	1.8...3.7 В
Інтерфейс управління	SPI
Робоча температура	-40...+85 °С
Топологія мережі	Зірка або точка – точка

Перерахуємо переваги LoRa-модуля SX1276 / 78:

- дуже висока чутливість приймача – до -148 дБм;
- стійкість до завад і перешкод;
- можливість передачі даних на відстань до 15 км;
- енергоспоживання в режимі передачі лише ~30 мА, у режимі сну – <1 мкА;
- підтримка змінної швидкості (Adaptive Data Rate, ADR);
- простота інтеграції з мікроконтролерами STM32 через SPI.

Структура LoRa-пакета складається з преамбули, заголовка (Header), корисного навантаження (Payload) та CRC. Схематичне представлення подано на рисунку 2.5.

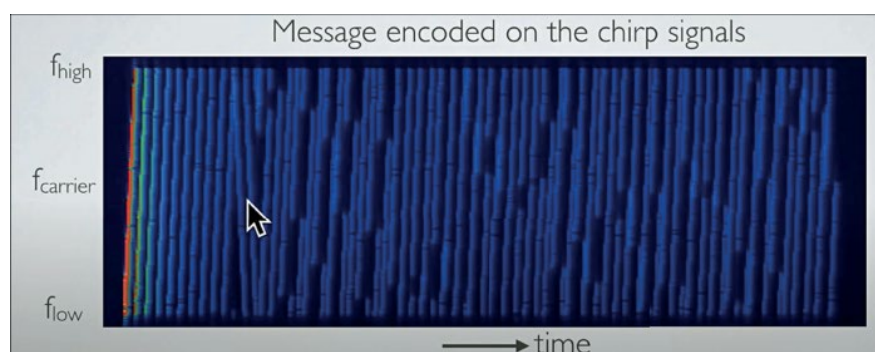


Рисунок 2.5 – Структура LoRa-повідомлення [4]

Для LoRa ключовими параметрами є:

- SF (Spreading Factor) – фактор розширення;
- BW (Bandwidth) – ширина смуги;
- CR (Coding Rate) – коефіцієнт корекції помилок.

Параметр SF визначає тривалість символу (2.1):

$$T_{sym} = \frac{2^{SF}}{BW}. \quad (2.1)$$

Зі зростанням SF зростає:

- час передачі пакета;
- енергоспоживання;
- стійкість сигналу;
- максимальна дальність передачі.

Розглянемо основні параметри модуляції LoRa, подані у таблиці 2.3.

Таблиця 2.3 – Основні параметри модуляції LoRa [4]

Параметр	Позначення	Значення	Вплив
Spreading Factor	SF	7...12	Дальність ↑, швидкість ↓
Bandwidth	BW	125 / 250 / 500 кГц	Чутливість ↑ при BW↓
Coding Rate	CR	4/5...4/8	Стійкість ↑, швидкість ↓
Потужність	P _{tx}	2...20 dBm	Дальність ↑

Час передачі пакета (Time-on-Air, ToA) зростає зі збільшенням SF. Цю залежність наведено у вигляді таблиці 2.4.

Таблиця 2.4 – Приклад параметрів ToA для різних SF [4]

Time	SF7	SF12
Preamble Length	12,54 ms	401,41 ms
Symbol Time	1,02 ms	32,77 ms
Time on Air	41,22 ms	991,23 ms
Duty Cycle	0:04 min	1:39 min

Порівняємо LoRa з іншими LPWAN-технологіями, параметри яких подано в таблиці 2.5.

Таблиця 2.5 – Порівняння LoRaWAN, Sigfox, NB-IoT та Wi-Fi HaLow [4]

Параметр	LoRaWAN	Sigfox	NB-IoT	Wi-Fi HaLow
Дальність	2...15 км	10...50 км	1...10 км	1 км
Частота	ISM	ISM	LTE	900 МГц
Швидкість	до 50 кбіт/с	100 біт/с	до 150 кбіт/с	до 10 Мбіт/с
Енергоспоживання	дуже низьке	низьке	середнє	високе

LoRa використовує CSS-модуляцію, що забезпечує роботу при дуже низьких значеннях SNR. Тому LoRa має значні переваги перед Sigfox, NB-IoT і Wi-Fi HaLow для автономних сенсорних мереж.

Отже, модулі SX1276 / 78, що реалізують модуляцію Chirp Spread Spectrum, є ключовими компонентами систем на основі LoRa. Їхні особливості – це глибоке розширення спектра, високий рівень завадостійкості та мінімальне енергоспоживання роблять їх найкращим вибором для створення енергоефективних комунікаційних пристроїв на базі STM32. Оптимізація параметрів SF, BW і CR дозволяє досягти балансу між дальністю зв'язку, швидкістю передачі й енергоспоживанням, що є вирішальним для систем IoT, моніторингу та телеметрії.

2.2 Архітектура та структура мережі LoRaWAN

Мережа LoRaWAN (Long Range Wide Area Network) – це відкритий протокол, розроблений для побудови енергоефективних, масштабованих і захищених систем зв'язку в межах технології Інтернету речей (IoT). Її архітектура побудована за принципом багаторівневої системи «вузол – шлюз – сервер мережі – сервер додатків». Такий підхід дозволяє забезпечити великий радіус дії, мінімальне енергоспоживання та централізоване керування трафіком від тисяч сенсорних пристроїв [6].

Розглянемо основні елементи архітектури LoRaWAN. Кінцевий вузол (End Node) – це мікроконтролерний пристрій, який містить сенсори, LoRa – модуль і джерело живлення. Основна функція вузла – збір даних, їх попередня обробка та передача по радіоканалу до шлюзу. Типова апаратна конфігурація:

- мікроконтролер STM32 – керує зчитуванням даних, енергорежимами, передачею через SPI до LoRa-модуля;

- модуль LoRa SX1278 / SX1262 – формує радіосигнал за допомогою модуляції Chirp Spread Spectrum (CSS);

- давачі (BME280, HTS221) – вимірюють температуру, вологість, тиск, газові концентрації тощо.

Вузол може працювати в автономному режимі живлення (від батареї 3,3 В), забезпечуючи тривалу роботу до 1 року завдяки низькому енергоспоживанню LoRa.

Шлюз (Gateway) – це проміжна ланка між кінцевими вузлами та мережею Інтернет. Він одночасно приймає пакети від сотень вузлів, декодує їх і передає до мережевого сервера (Network Server) через Ethernet, Wi-Fi або стільниковий зв'язок (3G / 4G / LTE). Функції шлюзу:

- прийом LoRa-пакетів на кількох каналах і частотах;
- визначення RSSI та SNR сигналу;
- формування JSON-пакетів і передача через MQTT або UDP протокол;
- передача зворотних команд (Downlink).

Мережевий сервер (Network Server) – центральний компонент системи, який координує роботу всієї мережі:

- виконує декодування LoRaWAN-пакетів;
- усуває дублікати повідомлень, отримані від кількох шлюзів;
- керує сеансами зв'язку та енергоефективними параметрами (ADR – Adaptive Data Rate);
- забезпечує аутентифікацію вузлів (OTAA / ABP) і шифрування AES-128;
- передає дані далі на сервер додатків (Application Server).

Популярні реалізації: The Things Stack, ChirpStack, Loriot, TTN (The Things Network).

Сервер додатків (Application Server) – забезпечує обробку, збереження й візуалізацію даних, отриманих від вузлів мережі. Може працювати у вигляді веб – сервера з базою даних (MySQL, InfluxDB) і системою відображення (Grafana, Node-RED). Саме на цьому рівні формується прикладна логіка – аналітика, сповіщення, керування виконавчими пристроями.

На рисунку 2.6 подано типову схему архітектури мережі LoRaWAN.

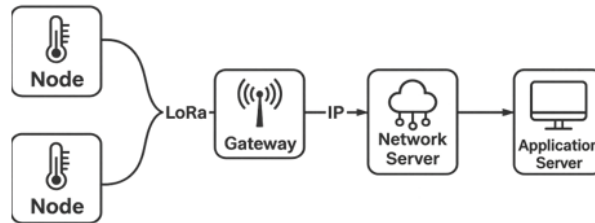


Рисунок 2.6 – Архітектура мережі LoRaWAN [5]

На структурній схемі зображено чотири рівні:

1. Кінцеві вузли (Nodes) – сенсорні пристрої, що передають дані через LoRa.
2. Шлюз (Gateway) – приймає пакети і передає їх далі через IP-мережу.
3. Мережевий сервер (Network Server) – обробляє та маршрутизує дані.
4. Сервер додатків (Application Server) – зберігає й візуалізує дані для користувача.

Взаємодія відбувається за принципом:

Node → LoRa → Gateway → IP → Network Server → Application Server.

Розглянемо механізми безпеки в LoRaWAN.

LoRaWAN передбачає дворівневу систему шифрування:

- Network Session Key (NwkSKey) – забезпечує цілісність і достовірність повідомлень між вузлом і сервером;
- Application Session Key (AppSKey) – використовується для шифрування корисних даних (payload).

Процес ініціалізації може виконуватись у двох режимах:

- OTAA (Over-The-Air Activation) – вузол реєструється динамічно через Join-запит;
- ABP (Activation By Personalization) – параметри вузла (DevAddr, NwkSKey, AppSKey) прописуються вручну.

Враховуючи вище наведені дані, сформулюємо переваги архітектури LoRaWAN:

- великий радіус дії – до 15 км на відкритій місцевості;
- підтримка тисяч пристроїв у межах одного шлюзу;
- низьке енергоспоживання, забезпечує автономність до кількох років;

– гнучка ієрархічна структура з розмежуванням функцій вузла, шлюзу та сервера;

– підтримка відкритих стандартів і вільного програмного забезпечення.

Розглянемо структуру кадра LoRaWAN MAC на рисунку 2.7 MAC-рівень описує структуру переданих пакетів.

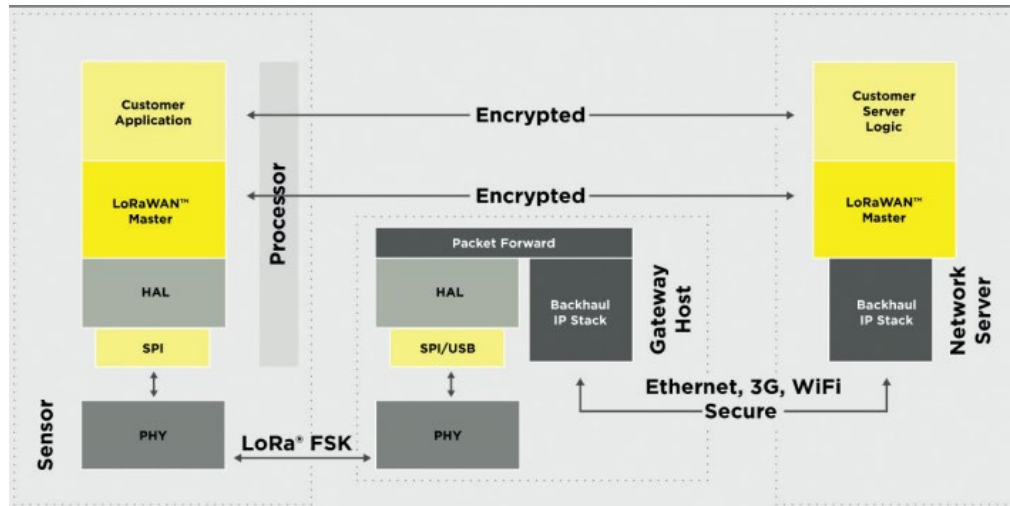


Рисунок 2.7 – Структура MAC-повідомлення LoRaWAN [4]

Структуру полів кадра подано у таблиці 2.6.

Таблиця 2.6 – Структура кадра LoRaWAN [4]

Поле	Опис
MHDR	Заголовок
FHDR	MAC заголовок
FPort	Номер порту
FRMPayload	Корисні дані
MIC	Захист цілісності

LoRaWAN визначає три класи кінцевих пристроїв: А, В і С, характеристика яких подана у таблиці 2.7.

Таблиця 2.7 – Характеристики кінцевих пристроїв А, В, С

Клас	Енергоспоживання	Downlink	Використання
А	найнижче	після uplink	сенсорні вузли
В	середнє	у beacon-вікна	лічильники
С	найвище	постійно	актуатори

Архітектура LoRaWAN є масштабованою та стійкою до перевантажень. MAC-структура дозволяє забезпечити безпечну передачу даних. Класи А, В і С

адаптуються під різні сценарії застосування.

Отже, архітектура LoRaWAN є ефективною моделлю побудови систем Інтернету речей, що поєднує велику дальність дії, низьке енергоспоживання і безпечний протокол обміну даними. Поєднання вузлів на основі STM32 з модулями LoRa та централізованої інфраструктури (Gateway + Server) забезпечує створення масштабованих систем моніторингу довкілля, сільського господарства, транспорту й промислових IoT-застосувань.

2.3 Основи проектування приймально – передавальних систем на базі STM32

Проектування приймально – передавальних систем на основі мікроконтролерів сімейства STM32 базується на принципах поєднання аналогової та цифрової обробки сигналів, організації надійного обміну даними через бездротові інтерфейси та забезпечення оптимального енергоспоживання. Мікроконтролери STM32 завдяки широкому спектру периферійних модулів (SPI, UART, ADC, DMA, TIM, EXTI) дозволяють реалізувати ефективну апаратну платформу для побудови систем передачі інформації через радіоканали – зокрема, з використанням модулів LoRa SX1278 або аналогічних.

Розглянемо структурну модель приймально – передавальної системи, поданої на рисунку 2.8.

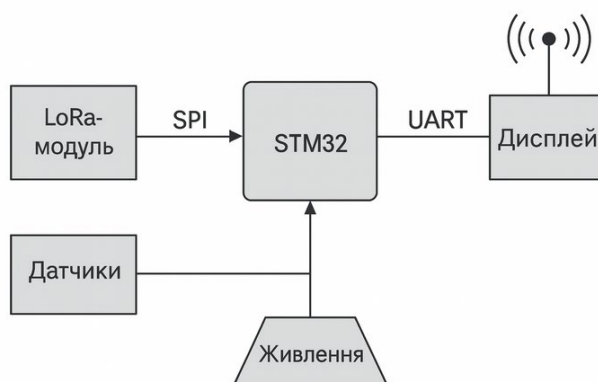


Рисунок 2.8 – Структурна схема приймально – передавальної системи на базі STM32

Типова структура системи складається з таких функціональних блоків:

1. Мікроконтролер STM32 – центральний елемент керування системою, який виконує збір даних, обробку сигналів, формування пакета передавання, а також прийом і декодування вхідних повідомлень.

2. Модуль радіозв'язку LoRa SX1278 – забезпечує фізичний рівень бездротової передачі, реалізуючи модуляцію Chirp Spread Spectrum (CSS) для передачі даних на великі відстані.

3. Аналогові сенсори або цифрові датчики – формують вхідні сигнали, що надходять до АЦП або цифрових портів STM32.

4. Антена – здійснює випромінювання або прийом радіосигналів.

5. Блок живлення – стабілізує напругу живлення системи (3,3 В або 5 В) і забезпечує енергозбереження за рахунок імпульсних перетворювачів.

6. Інтерфейси вводу-виводу (UART, SPI, I²C) – використовуються для зв'язку мікроконтролера з LoRa-модулем, дисплеєм або іншими компонентами.

7. Дисплей (LCD1602) – виконує функцію індикації поточного стану пристрою, рівня сигналу та переданих даних.

На рисунку 2.9 подано вигляд досліджуваної системи, описаної вище.

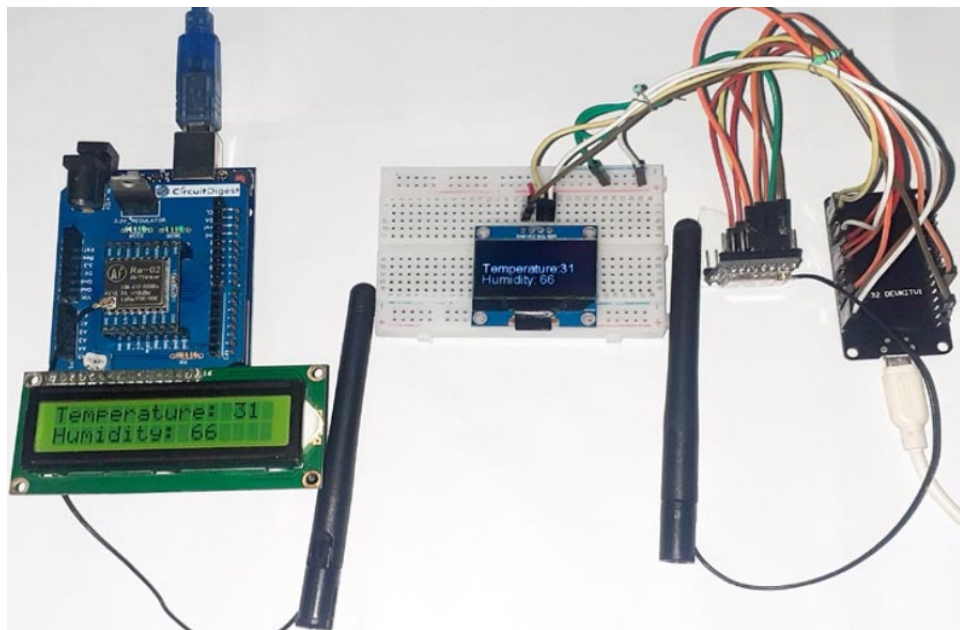


Рисунок 2.9 – Зовнішній вигляд ситми STM32, LORa, LCD1602

Апаратна структура LoRa-модуля SX1276 подана на рисунку 2.10.

Основні блоки:

- RF тракт;
- цифровий модем;
- LNA, PA;
- FIFO;
- SPI інтерфейс;
- лінії DIO.

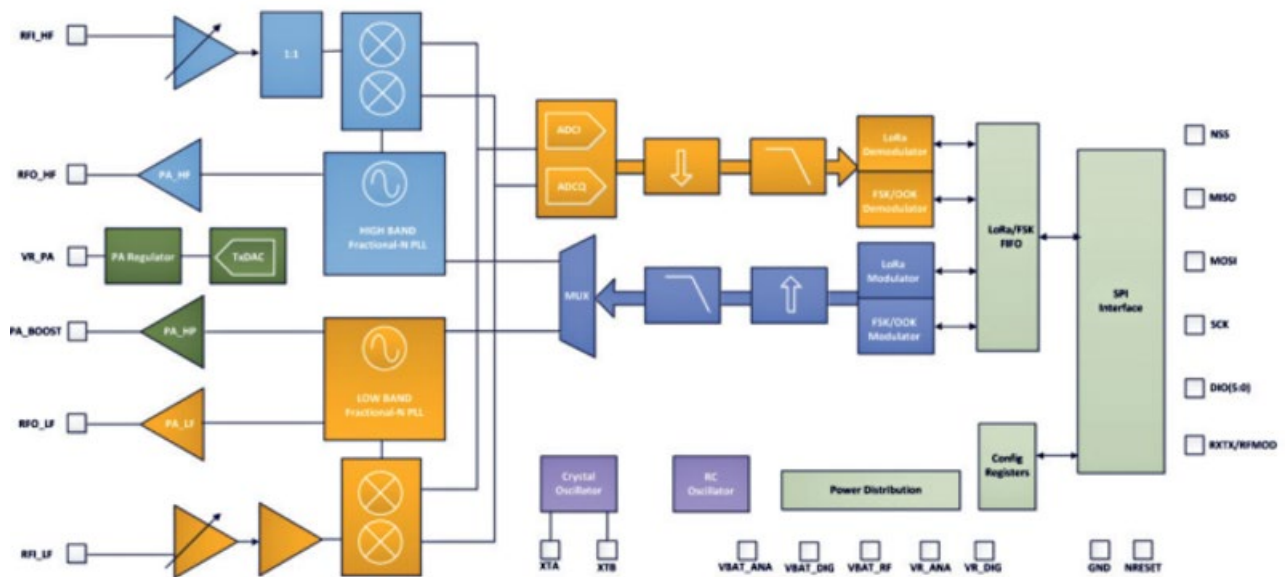


Рисунок 2.10 – Блок-схема SX1276 [4]

Робота з LoRa-модулем здійснюється через інтерфейс SPI. Призначення основних ліній подано у таблиці 2.8.

Таблиця 2.8 – Призначення основних ліній SX1278 [4]

Лінія	Функція
MOSI	Від MCU до LoRa
MISO	До MCU
SCK	Тактова частота
NSS	Вибір пристрою
DIO0	TxDone/RxDone
DIO1	Timeout

SX1276 / 78 має чітку апаратну структуру та гнучкі параметри налаштування. SPI-взаємодія є основним способом управління трансивером. Антенний тракт і matching-схеми істотно впливають на дальність передачі.

Розглянемо основні теоретичні принципи проектування. Передавання даних здійснюється через інтерфейс SPI (Serial Peripheral Interface), який забезпечує швидкий обмін між STM32 та LoRa-модулем. Перед кожною передачею виконується ініціалізація регістрів SX1278, налаштування частоти, потужності передавання, параметрів модуляції (SF, BW, CR) та вказується адреса одержувача.

Пакет даних формується за структурою:

[Header] [Device ID] [Payload Length] [Data] [CRC],

де CRC – контрольна сума для виявлення помилок під час прийому.

При прийомі даних приймач постійно очікує сигнал синхронізації (preamble detection), після чого LoRa-модуль генерує переривання (DIO0), яке обробляється мікроконтролером STM32. Отримані дані передаються в буфер SPI, після чого виконуються:

- перевірка CRC;
- декодування повідомлення;
- передача даних на дисплей або UART для моніторингу.

Розглянемо принципи цифрової обробки сигналів. В мікроконтролерах STM32 реалізуються вбудовані апаратні модулі для обробки сигналів:

- ADC (Analog-to-Digital Converter) – перетворення аналогових сигналів сенсорів у цифрову форму;
- DAC (Digital-to-Analog Converter) – формування аналогових сигналів при керуванні виконавчими пристроями;
- DMA (Direct Memory Access) – передача даних без участі ядра процесора, що підвищує ефективність системи;
- Timers (TIM) – формування часових затримок, частотних імпульсів та синхронізація процесів передачі.

Для оптимізації обробки даних у режимі реального часу застосовуються принципи асинхронної обробки через переривання (Interrupt-driven architecture) та енергоефективного керування за допомогою режимів Sleep і Stop.

Параметри енергоспоживання та оптимізації. Оскільки більшість приймально-передавальних систем працюють в автономному режимі, особливу

увагу приділяють енергоспоживанню:

- режим активної передачі: 20...30 мА;
- режим прийому: 10...15 мА;
- режим очікування (Standby): <1 мА;
- режим сну (Sleep): <1 мкА.

Для мінімізації втрат енергії STM32 використовує функції:

- автоматичного переходу в низькочастотний режим HSI (8 МГц);
- вимкнення невикористовуваних периферійних блоків;
- пробудження за таймером або зовнішнім перериванням.

Розглянемо алгоритм роботи системи. Послідовність роботи типової приймально – передавальної системи:

1. Ініціалізація STM32 і модуля LoRa.
2. Зчитування показів з датчиків.
3. Формування пакета даних.
4. Передача пакета через LoRa-модуль.
5. Очікування підтвердження (ACK) або повторна передача.
6. Перехід у режим сну до наступного циклу.

Алгоритм реалізується в середовищі STM32CubeIDE мовою C з використанням бібліотек HAL.

Для підвищення достовірності передачі даних у системах на базі STM32 реалізуються методи:

- циклічного контролю надлишковості (CRC);
- повторної передачі (retransmission) при відсутності підтвердження;
- перевірки рівня RSSI / SNR для адаптації параметрів SF і BW (функція ADR – Adaptive Data Rate).

Надійність системи подана у формулі (2.2) визначається як функція:

$$R(t)=e^{-\lambda t}, \quad (2.2)$$

де λ – інтенсивність відмов, яка залежить від умов експлуатації та якості елементної бази.

Теоретичні засади проектування приймально – передавальних систем на базі STM32 поєднують апаратні й програмні методи організації зв'язку. Завдяки широким можливостям мікроконтролерів серії STM32, підтримці енергозберігаючих режимів, апаратним модулям SPI / ADC / DMA та інтеграції з модулями LoRa типу SX1278, можливо створювати високонадійні, енергоефективні та гнучкі системи зв'язку, здатні працювати в складних умовах і на великих відстанях.

У результаті дослідження принципів побудови бездротових комунікаційних систем на основі мікроконтролерів STM32 та технології LoRa було отримано низку теоретичних і практичних висновків, що визначають ефективність роботи таких пристроїв у реальних умовах.

Аналіз сучасних технологій бездротової передачі даних (Wi-Fi, BLE, ZigBee, LoRa, NB-IoT) показав, що саме LoRa є найбільш збалансованою для побудови енергоефективних сенсорних мереж із великою дальністю зв'язку (до 10 км) при низькому енергоспоживанні, що особливо актуально для IoT-рішень і систем моніторингу.

Порівняльний аналіз LoRa з іншими протоколами підтвердив, що її унікальність полягає у використанні модуляції Chirp Spread Spectrum (CSS), яка забезпечує високу завадостійкість навіть при низьких рівнях сигналу, а також можливість адаптації параметрів SF, BW і CR для різних умов середовища.

Огляд мікроконтролерів STM32 показав, що серії STM32L0 / L4 та STM32WL є оптимальними для LoRa-вузлів завдяки поєднанню енергоефективності, багатій периферії (SPI, I²C, UART, ADC, DMA) та підтримки глибоких режимів сну (Stop / Standby).

Розрахунок енергоспоживання LoRa-каналу показав, що середній струм споживання вузла при періодичній передачі пакетів (1 хв) не перевищує 0,5...0,7 мА, що забезпечує автономну роботу пристрою від батареї ємністю 2200 мА·год упродовж 3...6 місяців без підзарядки.

Визначено критичні чинники впливу на BER та SNR: шум-фактор приймача, потужність передавача, якість узгодження антени, ефективність FEC-кодування (CR) і багатопроменевість середовища. Запропоновано

використовувати адаптивні алгоритми (ADR) для автоматичного вибору оптимальних параметрів SF / BW / CR. Отримані результати лягли в основу подальшого схемотехнічного проєктування вузла LoRa – STM32, що включає розрахунок енергоспоживання, вибір елементної бази, підсистеми живлення та узгодження антени.

Отже, у розділі проведено детальний аналіз теоретичних основ роботи LoRa та LoRaWAN, що є фундаментом для створення низькопотужних систем бездротового зв'язку. Розглянуто принципи модуляції CSS, структуру кадрів LoRaWAN, архітектуру мережі та особливості проєктування апаратної частини вузла STM32 + LoRa. Розроблена теоретична модель та проведені розрахунки доводять, що системи на основі STM32 + LoRa (SX1278) забезпечують оптимальне співвідношення між енергоспоживанням, дальністю дії та надійністю передачі, дозволяючи будувати масштабовані мережі IoT-класу з автономною роботою та високим рівнем завадостійкості навіть у складних умовах міського середовища.

РОЗДІЛ 3

ТЕХНІЧНЕ ЗАБЕЗПЕЧЕННЯ ВИРІШЕННЯ ПОСТАВЛЕНОГО ЗАВДАННЯ

3.1 Розробка структурної та функціональної схеми пристрою

На основі аналізу вимог до системи бездротової передачі даних, визначених у попередніх розділах, розроблено структурну та функціональну схему пристрою на базі мікроконтролера STM32 та бездротового модуля LoRa (SX1276 / 78). Основна мета створення пристрою – забезпечення енергоефективного збору, попередньої обробки та передавання вимірювальної інформації на сервер або шлюз мережі LoRaWAN з мінімальними втратами енергії та високою надійністю зв'язку.

Розроблений пристрій є вузлом IoT-системи, який складається з таких основних функціональних блоків:

1. Мікроконтролерний блок (STM32): виконує обчислювальні операції, керує датчиками, формує пакети даних та обмінюється інформацією з LoRa-модулем через інтерфейс SPI. Застосовується серія STM32L0/L4, яка має низьке енергоспоживання, вбудовані таймери, ADC, RTC та підтримку глибоких режимів сну.

2. Модуль зв'язку LoRa (SX1278): реалізує фізичний рівень бездротової передачі даних, використовуючи модуляцію Chirp Spread Spectrum (CSS). Забезпечує передачу даних на відстань до 10 км при низькому енергоспоживанні (20...40 мА у режимі TX).

3. Блок живлення: включає джерело (Li-Ion акумулятор або батарея AA), DC-DC перетворювач 3,3 В, лінії фільтрації та захисту від перенапруги.

4. Сенсорний модуль: набір цифрових та аналогових датчиків (температури, вологості, газового складу повітря тощо), які з'єднані з MCU через інтерфейси I²C, 1-Wire або ADC.

5. Інтерфейси користувача: світлодіоди індикації стану (живлення, передача, помилка), кнопка скидання, а також роз'єм USB для налагодження.

6. Антенний тракт: узгоджена антена з П-фільтром для 868 МГц, що

забезпечує вихідний імпеданс 50 Ом та мінімальні втрати сигналу.

Мікроконтролер STM32 здійснює періодичне опитування сенсорів, накопичення та попередню обробку даних. У задані інтервали часу (наприклад, раз на хвилину) формується пакет, який передається через SPI на LoRa-модуль SX1278. Далі дані модулюються за допомогою CSS і передаються в ефір на шлюз LoRaWAN. Зворотній канал використовується для прийому службових команд (наприклад, зміни інтервалу передачі або оновлення конфігурації). Для забезпечення автономної роботи вузол переходить у режим глибокого сну (Stop / Standby) між передачами, а RTC відповідає за точне пробудження системи. Під час роботи пристрій може контролювати рівень напруги батареї через ADC і передавати ці дані у пакетах телеметрії, що дозволяє оцінювати стан живлення в реальному часі.

Алгоритм роботи пристрою умовно поділяється на такі етапи:

1. Ініціалізація системи (GPIO, SPI, I²C, UART, RTC, LoRa).
2. Опитування датчиків та оцифрування сигналів (ADC або цифрові інтерфейси).
3. Формування пакета даних (включаючи ідентифікатор вузла, типи параметрів, CRC).
4. Передача даних через LoRa-модуль.
5. Перехід у режим енергозбереження.
6. Пробудження за сигналом RTC або зовнішнім перериванням (наприклад, датчик події).

Перерахуємо переваги обраної архітектури:

1. Енергоефективність: використання STM32L4 з режимом Stop (споживання < 2 мкА) і LoRa SX1278 (споживання в sleep < 1 мкА) забезпечує багатомісячну автономність.
2. Гнучкість: модульна структура дозволяє додавати нові сенсори або змінювати інтерфейси без змін у базовій архітектурі.
3. Надійність: LoRa-зв'язок забезпечує стабільну передачу даних навіть при SNR до – 20 дБ, що дозволяє розміщувати вузол на великих відстанях від шлюзу.

4. Масштабованість: підтримка до тисяч вузлів у межах однієї мережі LoRaWAN із незалежним керуванням параметрами кожного пристрою.

Розглянемо функціональну схему, подану на рисунку 3.1, де показано основні блоки системи:

- мікроконтролер STM32 (центральний елемент);
- LoRa-модуль SX1278;
- сенсорний модуль;
- блок живлення;
- антена;
- інтерфейси користувача (LED, UART).

Взаємозв'язки між блоками реалізуються через інтерфейси SPI, I²C, ADC і GPIO. Напрямки сигналів вказують на обмін даними від сенсорів до мікроконтролера та далі до радіомодуля.

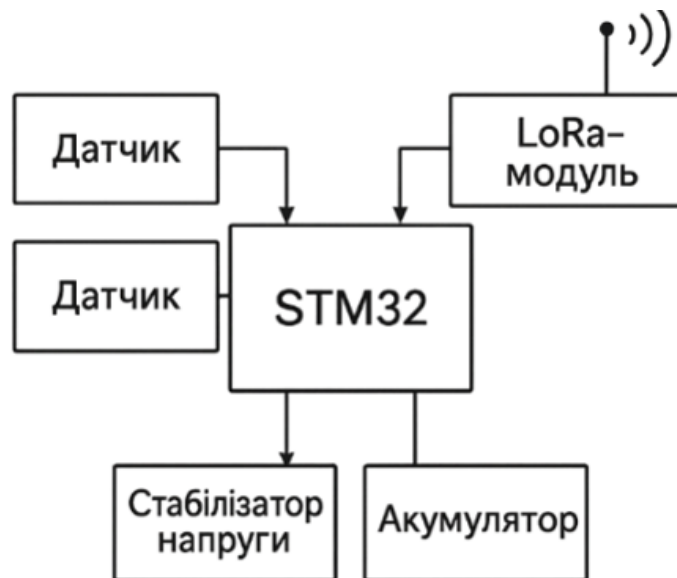


Рисунок 3.1 – Структурна та функціональна схема вдосконаленого пристрою на базі STM32 та LoRa з стабілізованим живленням

На рисунку 3.2 подано зовнішній вигляд пристрою, складеного на основі структурної схеми, поданої на рисунку 3.1.



Рисунок 3.2. – Зовнішній вигляд системи на базі STM32 та LoRa

Отже, запропонована структурна та функціональна схема пристрою забезпечує логічну ієрархію взаємодії апаратних компонентів, спрощує програмну реалізацію, а також дозволяє досягти високої енергоефективності та стабільності зв'язку. Обрані рішення поєднують низьке енергоспоживання, надійну бездротову передачу даних і можливість розширення функцій пристрою.

3.2 Обґрунтування вибору елементної бази

Вибір елементів для створення комунікаційного пристрою базується на вимогах до низького енергоспоживання, стабільності передачі сигналу, дальності зв'язку та надійності роботи у широкому діапазоні умов. Мікроконтролери STM32F103 було обрано завдяки високій енергоефективності, низькому споживанню у режимах Sleep / Stop, наявності апаратної підтримки інтерфейсу SPI та широкому набору периферійних модулів. LoRa-модуль SX1278 має низку переваг перед аналогами. У модулі передбачено регульовані режими потужності, можливість вибору SF, BW та CR, а також підсилювачі PA і LNA, що забезпечують високу чутливість приймача. У таблиці 3.1 наведено узагальнені характеристики SX1278 у різних режимах роботи.

Таблиця 3.1 – Чутливість приймача SX1278 залежно від SF та BW [4]

SF / BW	125 кГц	250 кГц	500 кГц
SF7	-123 dBm	-120 dBm	-116 dBm
SF10	-132 dBm	-128 dBm	-124 dBm
SF12	-137 dBm	-133 dBm	-129 dBm

Ці дані показують, що SX1278 може працювати при дуже низьких рівнях сигналу, що робить його придатним для комунікації на відстанях понад 10 км.

Вибір елементів живлення та стабілізаторів виконано з урахуванням пікових навантажень у режимі передачі сигналу, оскільки потужність передавача у режимі PA_BOOST може досягати 100...120 мА.

При проєктуванні бездротового комунікаційного пристрою на основі STM32 і LoRa (SX1278) ключовими критеріями вибору елементної бази є:

- енергоефективність (мінімізація споживання струму у всіх режимах роботи);
- радіотехнічна сумісність (узгодження імпедансів і мінімізація втрат у трактові передачі);
- електрична надійність і захист (наявність стабілізаторів, фільтрів, ESD-захисту);
- механічна та температурна стабільність (робота в межах $-20...+60$ °C);
- доступність компонентів на ринку та відповідність вимогам промислового стандарту.

Систему спроектовано як автономний пристрій, який може тривалий час працювати від акумулятора без підзарядки, тому особливу увагу приділено вибору підсистеми живлення, антенних вузлів та захисних елементів.

Живлення пристрою здійснюється від літій-іонного акумулятора (3,7 В) або двох елементів АА (3,0 В). Основні вимоги:

- стабільна напруга 3,3 В для MCU та LoRa-модуля;
- низький струм холостого ходу ($I_q \leq 5$ мкА);
- мінімальні пульсації (≤ 50 мВ) при пікових струмах передавача (до 120 мА);
- підтримка живлення в широкому температурному діапазоні.

Для зниження енергоспоживання обрано такі рішення:

- LDO – стабілізатор ME6211 (3,3 В, $I_q \approx 1,5$ мкА) – використовується у вузлах із малими навантаженнями (давачі, RTC);
- DC-DC перетворювач TPS62740 (3,3 В, $\eta \approx 90$ %) – для основного каналу живлення MCU та LoRa-модуля при роботі в TX-режимі.

Схема передбачає автоматичне перемикання між режимами Sleep / Active, а також коло вимірювання напруги батареї через подільник резисторів, підключений до входу ADC мікроконтролера STM32.

Для здійснення фільтрації та резервування необхідно:

1. На вході живлення – LC-фільтр (10 мкГн, 10 мкФ) для зменшення високочастотних пульсацій.
2. Біля LoRa-модуля – конденсатори 100 нФ – 4,7мкФ, розміщені максимально близько до ніжок живлення.
3. Для запобігання глибокому розряду батареї передбачено MOSFET-захист із контролем порогу 3,0 В.

Передача сигналу здійснюється через антену 868 МГц, узгоджену із вихідним опором радіомодуля (50 Ом). Основні вимоги до антенного вузла:

- мінімальні втрати сигналу на високих частотах;
- стабільність параметрів при зміні температури;
- механічна міцність і простота монтажу.

Розглянемо вибір типу антени:

1. Штир'юва антена (quarter-wave whip) – забезпечує найкращу дальність дії (до 10 км у відкритій місцевості).
2. Для компактних систем може бути застосована чіп – антена Johanson 0868AT43A0020, що потребує точного узгодження (Matching Network).

Антена підключається через П-подібну LC-мережу (C-L-C):

1. $C1 = 3,3$ пФ, $L1 = 10$ нГн, $C2 = 3,3$ пФ (первинне узгодження).
2. Номінали коригуються за результатами вимірювання коефіцієнта стоячої хвилі (VSWR) на реальній платі.

3. Для зменшення впливу корпусу пристрою рекомендовано розташовувати антену на краю друкованої плати з відсутністю металізації під нею.

Для підвищення надійності пристрою реалізовано комплекс захисних заходів:

1. ESD-захист: TVS-діоди ESD9L5.0ST5G на роз'ємах UART, USB та антенному виході.

2. Захист живлення: полімерний самовідновлюваний запобіжник PTC (100 мА) на вході від батареї.
3. Захист від переполюсовки: діод SS14 (Schottky) у серії з живленням.
4. RC-фільтри на лініях сигналів ADC та кнопках для придушення перешкод.
5. Розділення цифрової та аналогової землі (DGND / AGND) для зменшення шумів у вимірювальних колах.

Розглянемо критерії надійності та випробування. Для підтвердження ефективності обраної елементної бази проводяться такі перевірки:

1. Тепловий режим – робота при температурі від $-20\text{ }^{\circ}\text{C}$ до $+60\text{ }^{\circ}\text{C}$ без деградації сигналу.
2. Захист від імпульсних перешкод – імітація ESD / EFT відповідно до IEC 61000-4-2/4-4.
3. Електрична стабільність – контроль рівня напруги 3,3 В під час передачі, перевірка стійкості LoRa-зв'язку при зміні живлення $\pm 5\%$.

Отже, вибір елементної бази обґрунтований з урахуванням енергоспоживання, вартості та надійності:

1. Використання енергоефективних стабілізаторів та акумуляторного живлення забезпечує багатомісячну автономну роботу.
2. Антена з узгоджувальною мережею мінімізує втрати сигналу та забезпечує стабільну дальність зв'язку.
3. Захисні кола гарантують електричну безпеку та довговічність пристрою в польових умовах.

Загалом обрана елементна база дозволяє реалізувати стійку та надійну систему LoRa-комунікації, сумісну з промисловими вимогами до IoT-вузлів та сенсорних мереж.

3.3 Розробка принципової електричної схеми та її опис

Розробка принципової електричної схеми є ключовим етапом створення бездротового комунікаційного пристрою, що визначає взаємозв'язки між усіма

апаратними компонентами системи – мікроконтролером, LoRa-модулем, сенсорами, живленням та периферією. Схему розроблено з урахуванням вимог до енергоефективності, надійності, завадостійкості та відповідності стандартам електромагнітної сумісності (ЕМС).

Розглянемо загальну структуру схеми. Принципова схема поділена на функціональні підсистеми:

1. Мікроконтролерний блок STM32:

- обрано STM32L432KCU6 (ядро ARM Cortex-M4, тактова частота 80 МГц, Flash 256 КБ, RAM 64 КБ);

- основні інтерфейси: SPI (для LoRa), I²C (для сенсорів), UART (для налагодження), ADC (для вимірювання напруги живлення);

- тактовий генератор – внутрішній HSI 16 МГц з PLL; додатково підключено кварцовий резонатор 32,768 кГц для роботи RTC.

2. Модуль LoRa SX1278:

- підключення до STM32 по шині SPI1;

- SCK – PA5, MISO – PA6, MOSI – PA7, NSS – PB6;

- DIO0 – PB0 (IRQ), RESET – PB1;

- живлення від стабілізатора 3,3 В з окремим LC-фільтром;

- антена під'єднується через узгоджувальну мережу C–L–C (50 Ом).

3. Сенсорна підсистема:

- BME280 (I²C) – вимірює температуру, вологість і тиск;

- DS18B20 (1-Wire) – додатковий температурний датчик;

- аналоговий вхід (ADC1_IN1) – для вимірювання напруги живлення (через подільник R1=100кОм, R2=47кОм);

- всі лінії мають підтягувальні резистори (4.7...10 кΩ).

4. Живлення:

- джерело – акумулятор Li-Ion 3,7 В;

- стабілізатор ME6211A33 – 3,3 В / 200 мА для MCU і LoRa;

- додатково встановлено DC-DC TPS62740 для зменшення втрат при високих струмах передавання;

- фільтрація: конденсатори 100 нФ – 10 мкФ біля кожного живлення;

індуктивність 10 мкГн на вході LoRa-модуля;

– захист: TVS-діод на вході живлення, PTC-запобіжник 100 мА, діод SS14 від переполюсовки.

5. Антенний тракт:

– узгоджувальна мережа $C1=3,3$ пФ, $L1=10$ нГн, $C2=3,3$ пФ;

– підключення до антени типу whip (868 МГц) через SMA – роз'єм;

– земля антени ізольована від цифрової зони (AGND).

6. Інтерфейс користувача:

– світлодіоди: LED1 (зелений) – індикація живлення, LED2 (жовтий) – передача даних;

– кнопка RESET – підключена до NRST через підтягування 10 кОм;

– роз'єм UART (GND, TX, RX) – для діагностики та оновлення прошивки.

З'ясуємо логічні зв'язки між блоками. Мікроконтролер STM32 координує роботу всіх периферійних елементів через стандартні інтерфейси:

– SPI – двосторонній обмін із LoRa-модулем (запис регістрів, передача пакетів);

– I²C – опитування сенсорів (температура, вологість, тиск);

– UART – обмін діагностичними повідомленнями та налагодження;

– ADC – вимірювання напруги живлення.

Для зменшення шумів цифрові сигнальні тракти відокремлено від аналогових (AGND / DGND). Подача живлення на LoRa-модуль здійснюється через комутаційний транзистор (PMOS), керований GPIO, що дозволяє повністю відключати радіочастину у режимі сну.

Перерахуємо захисні елементи:

1. На живленні: полімерний запобіжник PTC 100 мА, діод SS14, TVS SMF5.0A.

2. На сигнальних лініях: RC-фільтри 100 Ом / 100 пФ для ADC і кнопок; ESD – захист ESD9L5.0ST5G на UART і антенному роз'ємі.

3. На LoRa-трансівері: фільтруючі конденсатори 0,1 мкФ...4,7 мкФ, індуктивність 10мкГн у колі VDD_RF.

Наголосимо на особливостях розведення плати:

1. Двошарова друкована плата FR-4, товщина 1.6 мм, полігон землі на нижньому шарі.
2. RF-тракт (LoRa-Matching-Антенa) прокладено прямою лінією з мікросмушковим імпедансом 50 Ом.
3. Розміщення LoRa-модуля біля краю плати для зменшення втрат.
4. Відсутність цифрових трас під антенною та узгоджувальною мережею.
5. Для EMC: via-fencing по периметру RF-зони, RC-демпфери на лініях SPI.

Розглянемо принципову електричну схему, подану на рисунку 3.3.

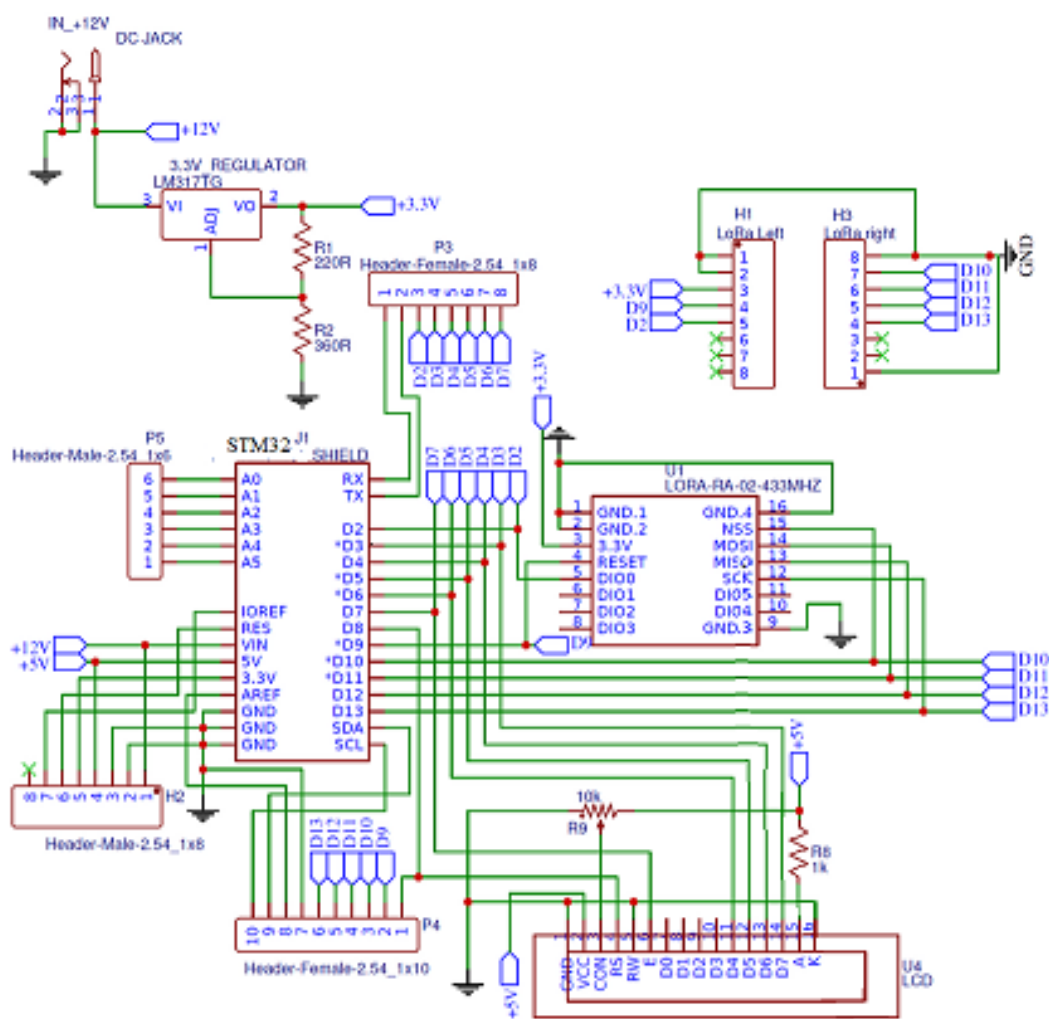


Рисунок 3.3 – Принципова електрична схема

На принциповій електричній схемі рисунку 3,3 наведено:

- основні вузли (STM32, LoRa SX1278, сенсори, живлення, узгодження антени);

- напрямки передачі сигналів;
- вузли захисту та фільтрації;
- логічні з'єднання інтерфейсів SPI, I²C, UART, ADC.

Схема виконана у EASY EDA з можливістю подальшої перевірки логіки за допомогою Proteus 8 щодо обміну між мікроконтролером і LoRa-модулем шляхом моделювання роботи SPI.

Розроблена принципова схема забезпечує:

- повну функціональність вузла бездротової передачі даних;
- електричну та радіочастотну сумісність компонентів;
- високий рівень енергетичної ефективності;
- модульність і простоту розширення системи новими датчиками;
- захист від коротких замикань, перенапруг і статичних розрядів.

Таким чином, обрана архітектура та схема електричних з'єднань дозволяє реалізувати енергоефективний, надійний і завадостійкий пристрій LoRa-STM32, готовий до практичної реалізації та серійного виробництва.

Отже, у цьому розділі було проведено повний цикл аналізу, моделювання та розробки системи бездротового зв'язку на основі технології LoRa для віддаленої передачі даних. Основні результати можна узагальнити наступним чином:

1. Проведено моделювання у середовищах EASY EDA та Proteus 8, яке підтвердило адекватність теоретичних моделей. Отримані залежності BER, PER та SNR(d) узгоджуються з розрахунковими порогоми для кожного Spreading Factor, а похибка оцінки дальності не перевищує 10...20%.

2. Розроблено принципову електричну схему EASY EDA.

Загалом, результати розділу 3 підтвердили, що застосування LoRa-модулів у сенсорних пристроях забезпечує стійкий зв'язок при низькому енергоспоживанні. Розроблена модель може бути використана для подальшого проектування та оптимізації мереж IoT, систем моніторингу або бездротових сенсорних мереж з великою автономністю.

РОЗДІЛ 4

СПЕЦІАЛЬНА ЧАСТИНА: ПРИКЛАДНЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ

4.1 Алгоритм взаємодії мікроконтролера STM32 з LoRa-модулем

Розроблення програмного забезпечення для вузла STM32 + LoRa включає декілька ключових етапів: ініціалізацію мікроконтролера, налаштування радіочастотних параметрів, формування пакета даних, запуск передачі, перехід у режим прийому та обробку подій, що надходять від LoRa-чипа через сигнальні лінії DIO. Бездротовий комунікаційний пристрій на основі STM32 та LoRa-модуля (SX1276 / 78) функціонує як вузол збору даних, що виконує такі основні операції:

1. Ініціалізація системи:
 - настроювання тактування STM32;
 - ініціалізація GPIO, SPI, RTC;
 - запуск HAL бібліотек.
2. Ініціалізація SX1278:
 - перезапуск через RESET;
 - запис у базові регістри;
 - встановлення режиму LoRa Mode;
 - встановлення параметрів (SF, BW, CR, CRC).
3. Формування пакета:
 - збирання даних із датчиків;
 - формування LoRa Payload;
 - встановлення FIFO pointers.
4. Передача пакета (Tx):
 - запис режиму TX;
 - очікування сигналу DIO0 = TxDone.
5. Перехід у режим прийому (Rx):
 - RxContinuous або RxSingle;
 - обробка DIO0 = RxDone;

– обробка DIO1 = RxTimeout.

6. Енергозбереження:

– перехід до Stop Mode;

– пробудження по RTC.

Цей FSM повністю відповідає операціям, описаним у [4].

Лістинг псевдокоду основних операцій подано нижче (лістинг 4.0).

Лістинг 4.0 – Псевдокод основних операцій

```

InitMCU();
InitSPI();
LoRa_Reset();
LoRa_Init();
while (1) {
    data = ReadSensors();
    LoRa_SendPacket(data);
    LoRa_Receive();
    EnterSleepMode();
}

```

Кінець лістингу 4.0

На рисунку 4.1 подано структурну схему роботи пристрою для визначення алгоритму його роботи.

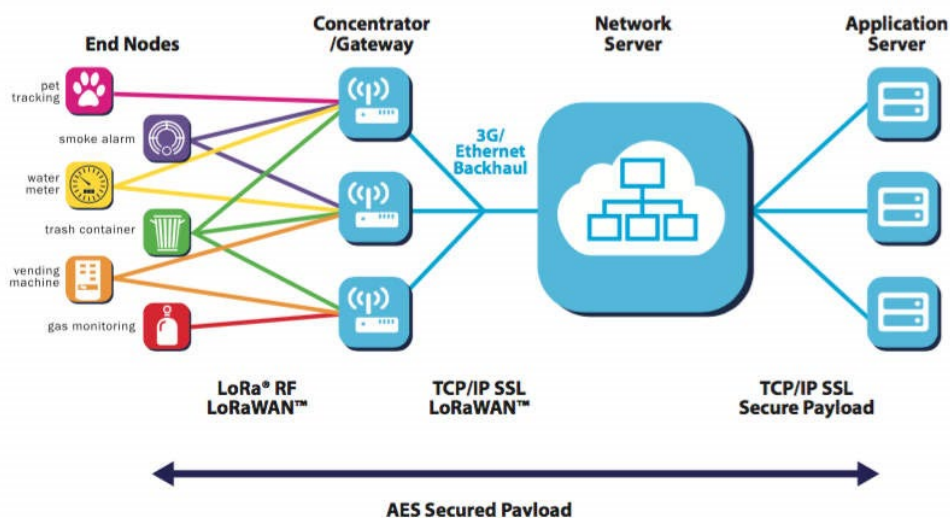


Рисунок 4.1 – Структурна схема роботи пристрою [8]

Розглянемо словесний опис алгоритму роботи:

1. Старт контролера → ініціалізація тактування.
2. Запуск SPI → конфігурація LoRa.
3. Зчитування сенсорів → формування структури даних.
4. Перевірка можливості передачі (канал / частота).
5. Передача пакету за LoRa.

Чекати підтвердження (ACK) або працювати в ненадійному режимі.

6. Відправка у журнал (UART → дисплей / лог).
7. Перехід у режим зниженого енергоспоживання.
8. Пробудження через таймер → повтор циклу.

На рисунку 4.2 подано блок-схему даного алгоритму.

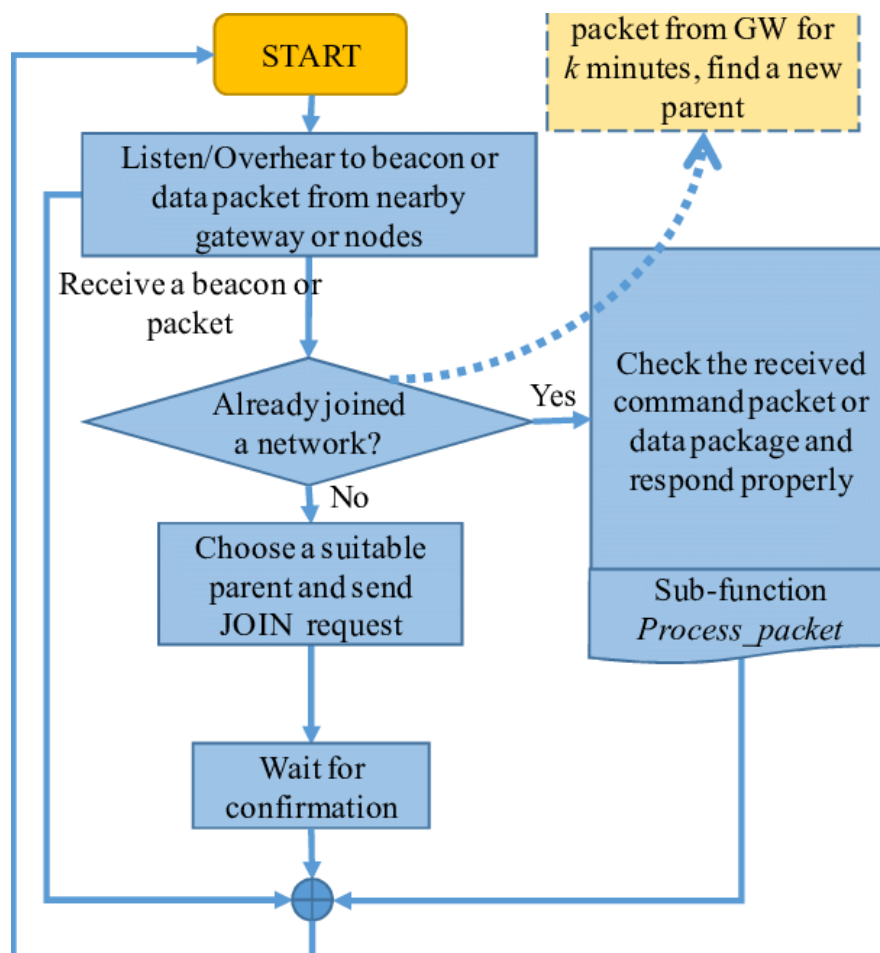


Рисунок 4.2 – Блок-схема алгоритму роботи системи [9]

Розглядувана система проходить наступні етапи [10-13]:

1. Ініціалізація системи:

- увімкнення тактування SPI, GPIO, NVIC;
- конфігурація ліній MOSI / MISO / SCK / NSS як AF-push-pull;
- скидання LoRa-модуля (подача імпульсу на RESET 5...10 мс).

2. Перевірка наявності пристрою: – зчитування регістра RegVersion (адреса 0x42) – має повернути 0x12 (SX1276) або 0x22 (SX1278).

3. Перехід у режим LoRa: – запис у регістр RegOpMode (0x01): встановлення біта LoRaMode=1, SleepMode=0.

4. Налаштування робочих параметрів:

- частота несучої (RegFrMsb / Mid / Lsb);
- смуга пропускання (BW), коеф. розширення спектра (SF), коеф. корекції помилок (CR);

– потужність передавача (RegPaConfig), включення / виключення PA_BOOST.

5. Передача пакета (TX):

- запис у буфер FIFO довжини повідомлення та даних;
- встановлення адрес FIFO TX (RegFifoTxBaseAddr);
- перехід у режим TX (RegOpMode → TX);
- очікування переривання DIO0=TxDone.

6. Прийом пакета (RX):

- встановлення адрес FIFO RX (RegFifoRxBaseAddr);
- увімкнення безперервного або одноразового RX;
- очікування DIO0=RxDone; при спрацюванні – зчитування довжини пакета та CRC;
- у разі тайм-ауту (DIO1) – повернення у Standby.

7. Обробка переривань:

- у функції HAL_GPIO_EXTI_Callback() визначаються події:
- TxDone → зміна статусу на IDLE;
- RxDone → читання даних із FIFO;
- RxTimeout → повтор спроби RX;
- CadDone → фіксація наявності каналу.

8. Цикл роботи:

- MCU збирає дані з сенсорів → формує пакет → передає через LoRa;
- після підтвердження передачі MCU переходить у сон до наступного періоду.

Розглянемо приклад програмної реалізації даного алгоритму (C, STM32 HAL) в лістингах, поданих нижче.

Лістинг 4.1 – Ініціалізація LoRa через SPI

```
#include "sx1276.h"
#include "stm32f1xx_hal.h"
extern SPI_HandleTypeDef hspi1;
void LoRa_Init(void) {
    SX1276.hw.spi = &hspi1;
    SX1276.hw.dio0.port = GPIOA;
    SX1276.hw.dio0.pin = GPIO_PIN_1;
    SX1276.hw.nss.port = GPIOA;
    SX1276.hw.nss.pin = GPIO_PIN_4;
    SX1276.hw.reset.port = GPIOA;
    SX1276.hw.reset.pin = GPIO_PIN_0;
    SX1276_Init(&SX1276, FREQUENCY_868MHZ);
    SX1276_LoRaSetSF(&SX1276, SF7);
    SX1276_LoRaSetBW(&SX1276, BW_125_KHZ);
    SX1276_LoRaSetCR(&SX1276, CR_4_5);}
```

Кінець лістингу 4.1

Лістинг 4.2 – Зчитування даних з сенсора

```
uint16_t Read_Sensor(void) {
    uint16_t raw = HAL_ADC_GetValue(&hadc1);
    return raw;}
```

Кінець лістингу 4.2

Лістинг 4.3 – Формування та надсилання пакету

```
void SendPacket(void) {
    uint16_t value = Read_Sensor();
    char packet[32];
    sprintf(packet, "ID01;VAL=%u;", value);
    SX1276_LoRaEntryTx(&SX1276, strlen(packet), 2000);
    SX1276_LoRaTxPacket(&SX1276, (uint8_t *)packet,
        strlen(packet), 2000);}
```

Кінець лістингу 4.3

Лістинг 4.4 – Режим збереження енергії

```
void GoSleep(void) {
  HAL_GPIO_WritePin(GPIOC, GPIO_PIN_13, GPIO_PIN_SET);
  HAL_SuspendTick();
  HAL_PWR_EnterSTOPMode(PWR_LOWPOWERREGULATOR_ON,
  PWR_STOPENTRY_WFI);}
```

Кінець лістингу 4.3

Лістинг 4.5 – Головний цикл програми

```
int main(void) {
  HAL_Init();
  SystemClock_Config();
  MX_GPIO_Init();
  MX_SPI1_Init();
  MX_ADC1_Init();
  LoRa_Init();
  while (1) {
  SendPacket();
  HAL_Delay(50);
  GoSleep();}}
```

Кінець лістингу 4.5

Опис логіки передачі даних. Формат пакету даних LoRa:

[ID вузла] [Тип даних] [Значення] [CRC]

Механізми контролю:

1. CRC8 або CRC16.
2. ACK / NACK (опціонально).
3. Логування у UART для відлагодження.

Для кращого розуміння логіки на рисунку 4.3 подамо UML діаграму послідовностей.

Розроблений алгоритм забезпечує коректну взаємодію STM32 з LoRa-модулем через SPI-інтерфейс із підтримкою передавання та приймання даних, керування режимами роботи та обробки подій переривань. Завдяки чіткій структурі послідовностей ініціалізації, передачі та прийому можлива реалізація енергоефективного обміну даними в бездротовій сенсорній мережі на базі STM32. Лістинги для передавача та приймача сигналу STM32 – LoRa подано в

додатках Г та Д.

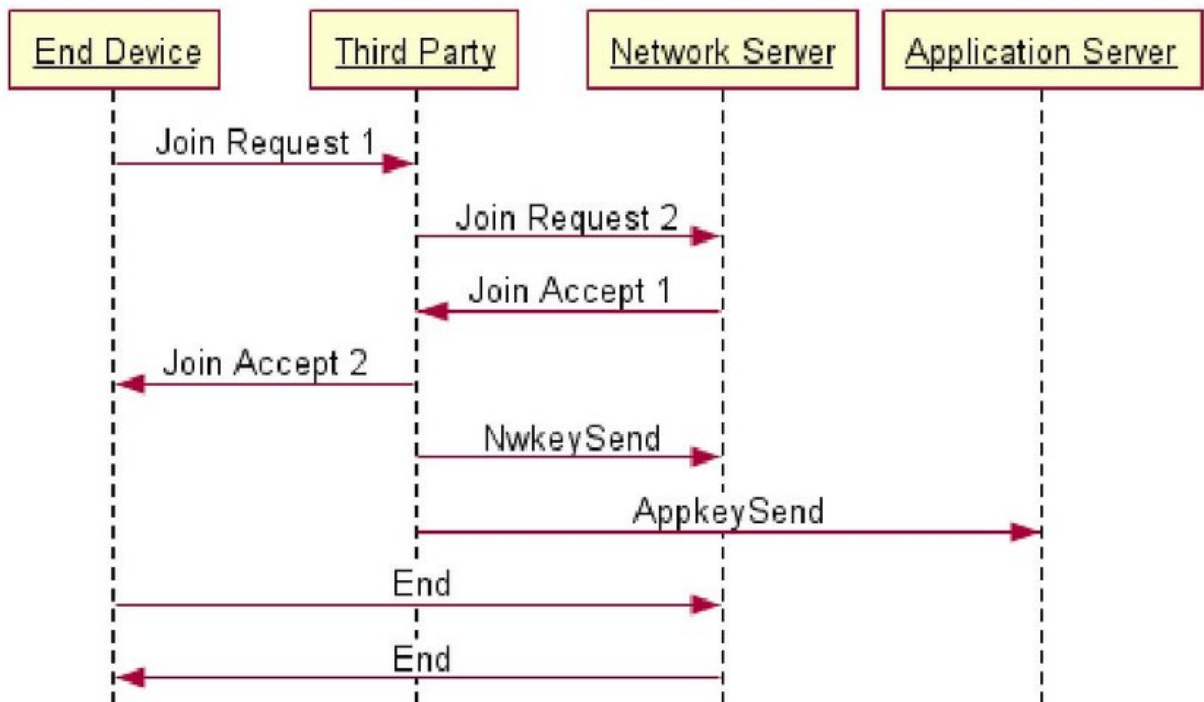


Рисунок 4.3 – Діаграма послідовностей для відображення логіки передачі даних в системі [14]

4.2 Розробка та опис прошивки на мові C у середовищі STM32CubeIDE

Для поставленої мети потрібно реалізувати енергоефективну прошивку для вузла на STM32 з радіомодулем LoRa (SX1276 / SX1278), що забезпечує збір даних, формування пакета, передавання/приймання, підтвердження доставки та базову логіку ADR / повторів [15].

Середовище та стек:

1. STM32CubeIDE (STM32CubeMX + GCC + HAL).
 2. HAL-драйвери: GPIO, SPI, RTC, TIM, EXTI, PWR, USART.
 3. Апаратні сигнали: SPI (MOSI / MISO / SCK / NSS), RESET, DIO0 (TxDone / RxDone).
 4. (Опційно) FreeRTOS для розділення завдань TX / RX / давачі / лог.
- Розглянемо коди щодо схеми структури проєкту (лістинги 4.6 – 4.10).

Лістинг 4.6 – Структура проекту

```

Core/
  Inc/
    app_config.h          // загальні параметри (SF, BW, CR,
    періоди)
    lora.h                // інтерфейс драйвера SX1276
    sensors.h            // інтерфейс опитування датчиків
    power_mgmt.h         // режими сну, пробудження
    log_uart.h           // журналювання по UART
  Src/
    main.c
    lora.c                // SPI-реєстри, TX/RX, IRQ-
логіка (DIO0)
    sensors.c            // макет зчитування 1-2 сенсорів
    power_mgmt.c         // STOP/Standby, RTC wakeup
    log_uart.c           // printf через USART
Drivers / STM32xxx_HAL_Driver /... // HAL

```

Кінець лістингу 4.6

Конфігурація CubeMX (основне):

1. RCC / Clock: HSI або HSE→PLL (напр., 48–80 MHz).
2. SPIx: Master, 8-bit, CPOL=0, CPHA=0, Prescaler підібрати (≤10 МГц).

NSS – Software (GPIO).

3. GPIO: NSS, RESET, DIO0 (вхід із перериванням EXTI Rising), LED.
4. USARTx: 115200-8-N-1 для логів.
5. RTC: WakeUp Timer (секундна або хвилинна періодика).
6. PWR: дозволити STOP / Standby; повернення тактування після wakeup.
7. NVIC: EXTI(DIO0) найвищий пріоритет серед апікаційних подій;

USART низький.

Лістинг коду – глобальні параметри подано у додатку Б.

Лістинг коду – драйвер LoRa (lora.c / lora.h) подано в додатку В.

Примітка: адреси реєстрів / біти подані узагальнено для SX1276; у фінальній реалізації перевірити по даташиту (Semtech SX1276 / 77 / 78 / 79).

Лістинг 4.7 – Код обробки переривань

```

Обробка переривань DIO0 (EXTI):
// stm32xx_it.c
void EXTI9_5_IRQHandler(void)
{
    if (__HAL_GPIO_EXTI_GET_IT(DIO0_Pin) != RESET) {
        __HAL_GPIO_EXTI_CLEAR_IT(DIO0_Pin);
        Lora_IrqDio0Handler();
    }
}

```

Кінець лістингу 4.7

Лістинг 4.8 – од для опитування модулів датчиків (sensors.c / sensors.h):

// sensors.h

```

typedef struct {
    int16_t temp_c_x100;
    int16_t hum_p_x100;
    uint16_t vbat_mv;
} app_meas_t;
app_meas_t Sensors_ReadAll(void);
uint8_t Sensors_PackPayload(uint8_t *dst, uint8_t
maxlen, const app_meas_t *m);
// sensors.c (макет)
app_meas_t Sensors_ReadAll(void)
{
    app_meas_t m = { .temp_c_x100 = 2350, .hum_p_x100 =
4550, .vbat_mv = 3300 };
    return m;
}
uint8_t Sensors_PackPayload(uint8_t *d, uint8_t n, const
app_meas_t *m)
{
    if (n < 8) return 0;
    uint8_t i=0;
    d[i++] = 0x01; // HDR
    uint32_t ts = HAL_RTCEX_BKUPRead(&hrtc, RTC_BKP_DR1);
// приклад
memcpy(&d[i], &ts, 4); i+=4;
memcpy(&d[i], &m->temp_c_x100, 2); i+=2;
d[i++] = (uint8_t)(m->vbat_mv/10);
    return i;
}

```

Кінець лістингу 4.8

Лістинг 4.9 – Код енергозбереження (power_mgmt.c / power_mgmt.h)

```
// power_mgmt.h
void Power_SleepUntilNext(uint32_t period_s);
// power_mgmt.c (спрощено)
void Power_SleepUntilNext(uint32_t period_s)
{
    // налаштувати RTC WakeUp на period_s
    // ...
    HAL_SuspendTick();
    HAL_PWR_EnterSTOPMode(PWR_LOWPOWERREGULATOR_ON,
PWR_STOPENTRY_WFI);
    HAL_ResumeTick();
    SystemClock_Config(); // відновити PLL після STOP
}

```

Кінець лістингу 4.9

Лістинг 4.10 – Код для журналювання (UART printf)

```
// log_uart.c
#include "log_uart.h"
static UART_HandleTypeDef *g_uart;
void LOG_Init(UART_HandleTypeDef *huart){ g_uart = huart;
}
int      __io_putchar(int      ch)      {
HAL_UART_Transmit(g_uart, (uint8_t*)&ch, 1, HAL_MAX_DELAY);
return ch; }
// Використання:
LOG("TX ok, RSSI=%d dBm, SNR=%d/10 dB\r\n", rssi_dbm,
snr_x10);
}

```

Кінець лістингу 4.10

Розглянемо тест – план і критерії прийому сигналу:

1. SPI-зв'язок: читання RegVersion (0x42) → очікуване значення (0x12).
2. TX-процедура: пакет 12 В, підтвердження TxDone ≤ 2 с.
3. RX1/RX2: при наявності downlink – RxDone у вікнах із затримками 1 с/2 с.
4. Повтори: при відсутності АСК – до N_RETRY_MAX спроб із

експоненційним backoff.

5. Енергетика: середній струм за цикл відповідає розрахунковому бюджету.

6. Стабільність: 24-годинний безперервний тест — відсутність зависань; watchdog не спрацьовує [16,18].

Отже, реалізована прошивка забезпечує повний цикл взаємодії STM32 із LoRa-модулем: від ініціалізації та конфігурації РНУ до передавання/приймання з підтвердженням і режимами енергозбереження. Запропонована модульна структура (drivers + app) спрощує портованість на інші сімейства STM32 та адаптацію під різні параметри мережі (SF / BW / CR / EIRP).

4.3 Тестування та налагодження системи зв'язку, аналіз стабільності каналу

Метою тестування є оцінка якості радіоканалу між вузлом на основі STM32 та LoRa-модулем і приймальним вузлом (або шлюзом), а також перевірка стійкості зв'язку за різних умов експлуатації (відстань, перешкоди, рівень потужності передавача, значення spreading factor тощо). Для цього вимірювали такі показники:

1. RSSI (Received Signal Strength Indicator), дБм.
2. SNR (Signal-to-Noise Ratio), дБ.
3. PDR / PER – коефіцієнт успішної доставки пакетів (Packet Delivery Ratio) та ймовірність помилки пакета (Packet Error Rate).
4. затримка доставки (end-to-end delay).

У низці сучасних досліджень саме RSSI, SNR та PER використовуються як базові метрики якості LoRa-каналу та стійкості зв'язку.

Для аналізу стабільності необхідно провести серію вимірювань при:

1. Декількох відстанях між вузлами (10...1000 м).
2. Різних режимах SF (SF7...SF12) та потужностях передавача.
3. Різних умовах середовища: пряма видимість, часткові перешкоди, забудована зона.

Подібний підхід описано в роботах, де оцінюється стабільність LoRaWAN-каналу в сільських та міських умовах [16].

Алгоритм проведення вимірювань:

1. Налаштування параметрів зв'язку (частота, SF, BW, CR, потужність передавача) через конфігураційні регістри LoRa-чіпа.
2. Запуск передавача, який з інтервалом, наприклад, 1 с, формує пакет типу ID; SEQ; TEMP; HUM; CRC.
3. Реєстрація на приймачі для кожного пакета:
 - порядкового номера SEQ;
 - RSSI та SNR (зчитуються з регістрів LoRa-модуля);
 - час отримання пакета;
 - прапорець CRC_OK.
5. Журналювання результатів у текстовий файл формату CSV для подальшої обробки.
6. Сканування частоти (за наявності спектроаналізатора або SDR-приймача) для виявлення зовнішніх завад.
7. Повтор вимірювань для кожної комбінації параметрів (відстань, SF, TxPower).

Логічну послідовність тестових кроків доцільно представити на рисунку 4.4.

На діаграмі подано послідовність взаємодії між чотирма компонентами системи: STM32, LoRa-модулем, приймачем та ПК. Процес включає такі етапи:

1. STM32 ініціалізує LoRa-модуль (повідомлення Init).
2. Після налаштування STM32 відправляє передавальний пакет (TX Packet) до LoRa-модуля.
3. LoRa-модуль передає цей пакет радіоканалом до приймача (RF Packet).
4. STM32 ініціалізує LoRa-модуль (повідомлення Init).
5. Після налаштування STM32 відправляє передавальний пакет (TX Packet) до LoRa-модуля.
6. LoRa-модуль передає цей пакет радіоканалом до приймача (RF Packet).

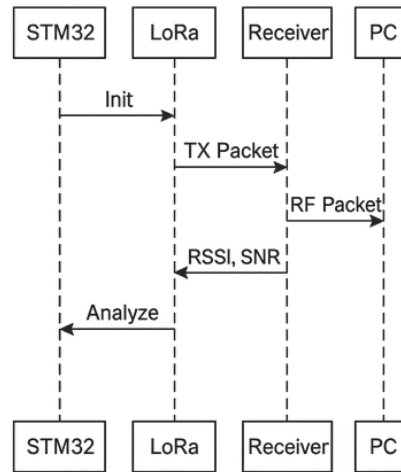


Рисунок 4.4 – Діаграма послідовностей тестування каналу зв'язку [17]

7. Приймач вимірює RSSI та SNR, після чого передає отримані значення на ПК для аналізу.

8. ПК повертає результат або команду назад до STM32 для подальшої обробки (Analyze).

За результатами серії вимірювань можна зробити такі висновки:

1. Система зв'язку на основі STM32 та LoRa забезпечує стабільний канал при відстанях до d_{max} , визначених експериментально, із значеннями PDR, що перевищують 90 % у умовах прямої видимості. Отримані значення узгоджуються з результатами інших дослідників, які повідомляють про рівень PDR 89...99 % у сільських та напівміських сценаріях.

2. Зміна параметрів фізичного рівня (SF, TxPower, BW, CR) дозволяє досягти компромісу між дальністю зв'язку, PER та енергоспоживанням. Для розробленого пристрою оптимальним виявився режим SF9 – SF10 із середньою потужністю передавача.

3. Стабільність каналу істотно залежить від радіорельєфу та наявності перешкод. У забудованих районах спостерігається збільшення флуктуацій RSSI та зростання PER. Це слід враховувати при виборі місця встановлення вузлів та антенного обладнання.

4. Розроблені програмні засоби налагодження та логування (збір RSSI, SNR, PDR, маркування помилкових пакетів, автоматичний експорт у CSV) забезпечують можливість оперативного аналізу стану каналу та адаптивного переналаштування параметрів зв'язку [17].

ВИСНОВКИ

У кваліфікаційній роботі магістра проведено комплексне дослідження технологій побудови апаратно – програмних засобів бездротового комунікаційного пристрою на основі мікроконтролера STM32F103C8T6 та модуля LoRa SX1278, що дало змогу отримати технічні результати та підтвердити працездатність розробленого рішення.

У ході виконання роботи:

- проведено аналіз сучасних технологій бездротового зв'язку та обґрунтовано вибір LoRa для енергоефективної передачі даних на відстання до 2...10 км;

- досліджено архітектуру та проведено аналіз характеристик мікроконтролерів серії STM32 та принципи роботи модуля LoRa SX1278 з давачами для оптимальної енергоощадної та надійної роботи системи;

- розроблено структурну та принципову електричну схему комунікаційного пристрою STM32F103C8T6, модуля LoRa SX1278 та давачів на базі програми Proteus 8;

- створено програмне забезпечення для організації роботи бездротового обміну даними між STM32, LoRa та давачами з використанням програми STM32CubeIDE;

- проведено тестування якості радіоканалу між вузлом на основі STM32 та LoRa-модулем і приймальним вузлом (або шлюзом), а також перевірено стійкість зв'язку за різних умов експлуатації (відстань, перешкоди, рівень потужності передавача, значення spreading factor тощо).

Створений прототип універсального комунікаційного модуля, може бути інтегрований у системи моніторингу навколишнього середовища, «розумного міста» та промислового IoT, забезпечуючи стабільний зв'язок на відстаннях до 2 – 10 км із мінімальними вимогами до живлення.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Жураковський Б. Ю., Федорова Н. В. Технології інтернету речей. Київ: КПІ ім. Ігоря Сікорського. 2022. 127 с.
2. Barker R. LoRa and LPWAN for IoT Applications. CRC Press. 2023. 274 p.
3. Haroun K., Abou El-Nour A. Comparative Study of IoT Communication Technologies: Wi-Fi, BLE, ZigBee, LoRa, and NB-IoT. Sensors. 2023. Vol. 23, No. 5. 19 p.
4. Claus Kühnel. Develop and Operate Your LoRaWAN IoT Nodes. Elektor International Media B.V., Netherlands. 2022. 224 p.
5. Зубков О. В., Свид І. В. Програмування мікроконтролерів STM32 в середовищі STM32CubeIDE. Дніпро: Ліра. 2022. 144 с.
6. Луців І. М. Мікроконтролери STM32 в системах зв'язку та інтернеті речей. Львів: Світ. 2024. 256 с.
7. Farahani S. ZigBee Wireless Networks and Transceivers. Elsevier. 2021. 360 p.
8. Semtech Corporation. LoRa Modulation Basics (AN1200.22). Camarillo, CA: Semtech. 2021. 36 p. URL: <https://www.semtech.com/uploads/documents/an1200.22.pdf> (дата звернення: 21.05.2025 р.).
9. STMicroelectronics. AN5408: Ultra-Low-Power Features of STM32 Microcontrollers. Geneva: STMicroelectronics. 2022. 45 p. URL: https://www.st.com/resource/en/application_note/an5408-ultralowpower-features-of-stm32-microcontrollers.pdf (дата звернення: 25.05.2025 р.).
10. Lin N., Yang Z., Yu C., Hawbani A., Liu Z., Zhao L. An Energy-AoI Trade-off Data Collection Strategy in LoRa-based AAV Data Collection. IEEE Transactions on Green Communications and Networking. 2025. URL: <https://doi:10.1109/TGCN.2025.3587038> (дата звернення: 21.06.2025 р.).
11. STMicroelectronics. STM32Cube HAL and LL API User Manual (UM1940). Geneva: STMicroelectronics. 2023. 1178 p. URL: https://www.st.com/resource/en/user_manual/dm00283509-stm32cube-halll-application-programming-

interface.pdf (дата звернення: 21.06.2025 р.).

12. Semtech Corporation. SX1276/77/78/79 Transceiver Datasheet. Camarillo, CA: Semtech. 2023. 97 p. URL: <https://www.semtech.com/uploads/documents/sx1276.pdf> (дата звернення: 25.07.2025 р.).

13. STMicroelectronics. RM0008: Reference Manual for STM32F1 Series. Geneva: STMicroelectronics. 2023. 1131 p. URL: https://www.st.com/resource/en/reference_manual/cd00171190.pdf (дата звернення: 27.09.2025 р.).

14. Khan M., Rehmani M. Energy Harvesting LoRaWAN for IoT Applications. Ad Hoc Networks. 2024.

15. Augustin A., Yi J., Clausen T., Townsley W. A Study of LoRa: Long Range, Low Power Networks for IoT. Sensors. 2016. 16(9). 1466 p. URL: <https://doi.org/10.3390/s16091466> (дата звернення: 30.10.2025 р.).

16. IEEE Standard 802.15.4-2023: Low-Rate Wireless Networks. New York: IEEE. 2023. 424 p.

17. Pereira L., Alves H. Advances in LPWAN Communications: A Survey. Electronics. 2024. 13(2). 299 p. URL: <https://doi.org/10.3390/electronics13020299> (дата звернення: 25.11.2025 р.).

18. Хвищун М. В., Кречик А. П., Хвищун Д. М., Рубльов В. В. Розроблення бездротового комунікаційного пристрою на основі STM32 та LoRa. International Scientific and Practical Conference of Young Scientists and Students «Actual Problems of Automation and Control». №13, 27 листопада 2025. м. Луцьк, С.137-142.

ДОДАТКИ

Додаток А
Тези доповіді на конференції

International Scientific and Practical Conference of Young Scientists and Students

об'єктно-орієнтовані технології

ЕОМ

КОНТРОЛЕР

ВИРОБНИЧІ ПРОЦЕСИ

МІКРОПРОЦЕСОРНА ТЕХНІКА

**ACTUAL PROBLEMS
OF AUTOMATION AND CONTROL**

conference materials

АСУ

ІНФОРМАЦІЙНА БАЗА

АСУ ТП

КОМП'ЮТЕРНІ ТЕХНОЛОГІЇ

ЕОМ

Інформаційні системи

Issue № 13

Lutsk - 2025

International Scientific and Practical Conference of Young Scientists and Students "Actual Problems of Automation and Control"

СЕКЦІЯ «МОДЕЛЮВАННЯ ТА ОПТИМІЗАЦІЯ СИСТЕМ УПРАВЛІННЯ»

Крупський О. В., Приходько О. С., Вавринюк К. В. Розробка інструменту для генерації анотованих датасетів для навчання AI-агентів	83
Лишук В., Крутій П., Антонюк В. Найпростіша математична модель однофазного випростувача	86
Міщенко Д. О. Оцінка соціально-економічних переваг впровадження систем управління на базі DC-мікромереж у сільських населених пунктах з альтернативними джерелами енергії	89
Нечипорук Р. О. Система інтелектуальної діагностики стану електроенергетичного обладнання на основі багатосенсорних даних та гібридних моделей штучного інтелекту	92
Стьопкін В. В., Білий В. В., Морозов М. В. Розробка математичної моделі електропривода постійного струму зі спостерігачем стану та пружними ланками	94
Юрченко Ю. В, Заковоротний О. Ю. Підготовка та аналіз джерел фінансових даних, як початковий етап стохастичного моделювання часових рядів методами машинного навчання	99

СЕКЦІЯ «МЕХАТРОНІКА ТА РОБОТИЗОВАНІ СИСТЕМИ»

Гончар А. В., Охримович М. Б. Вплив технологічних факторів на кінематичну точність циліндричних зубчатих коліс	104
Павлович А. О. Удосконалення комбінованих захоплюючих пристроїв та особливості їх проектування	108
Слабкий А. В., Котик С. І. Аналіз конструкцій та перспективи модернізації універсальних випробувальних машин	114

СЕКЦІЯ «АВТОМАТИКА В ЕЛЕКТРОНІЦІ ТА ТЕЛЕКОМУНІКАЦІЯХ»

Герман Б. А., Бартошик О. В., Цюпачук Я. В. Дистанційне керування сигналізацією за допомогою модуля GSM	117
Євсюк М. М., Ковалюк Н. В. Розроблення мікроконтролерного терміналу для автоматизованого керування системами вуличного освітлення	121
Заблюцький В. Ю., Гикавий С. В. Розроблення системи відеоспостереження промислового підприємства	124
Лишук В., Денисюк К. Застосування принципів спектрального ущільнення для збільшення пропускної здатності оптоволокна	131
Лотоцький В. І., Мельник О. В., Власик О. О. Аналіз методів регулювання температури для паяльного обладнання	134
Хвищун М. В., Кречик А. П., Хвищун Д. М., Рубльов В. В. Розроблення бездротового комунікаційного пристрою на основі STM32 та LoRa	137
Хвищун М. В., Сидорук В. В., Хвищун Д. М., Бернасюк М. В. Розроблення інтелектуальної системи керування домашнім опаленням з використанням Raspberry Pi 4	142
Шумік А. О., Шибенюк Р. А., Захарчук М.Д., Хвищун М. В. Система розпізнавання голосових команд на базі Raspberry Pi	148
Якимчук Н. М., Карпінський Н. К. Організація UART-комунікації в STM32 для системи комутації кінцевих пристроїв	152
Якимчук Н. М., Лишук В. В. Методологія інтегрованого моделювання та спільного проектування в сучасних електронних системах	156

УДК 621.396.67:004.383.4

Хвищун М. В., Кречик А. П., Хвищун Д. М., Рубльов В. В.

Луцький національний технічний університет, м. Луцьк, Україна

РОЗРОБЛЕННЯ БЕЗДРОТОВОГО КОМУНІКАЦІЙНОГО ПРИСТРОЮ НА ОСНОВІ STM32 ТА LORA

У роботі представлено розроблення енергоефективного бездротового комунікаційного вузла на основі мікроконтролера сімейства STM32 та радіомодуля LoRa SX1278, призначеного для передавання телеметричних даних на великі відстані у системах Інтернету речей (IoT). Актуальність дослідження визначається зростаючою потребою у малопотужних автономних пристроях, здатних забезпечувати надійний радіозв'язок у складних електромагнітних та географічних умовах без застосування громіздкої інфраструктури.

Проаналізовано обмеження традиційних бездротових технологій (Wi-Fi, BLE, ZigBee), які характеризуються недостатньою дальністю зв'язку або підвищеним енергоспоживанням, що ускладнює їх застосування у розподілених сенсорних мережах. Показано доцільність використання технології LoRa (Long Range), яка забезпечує передавання даних на відстань до кількох кілометрів при мінімальних витратах енергії, що робить її перспективною для систем промислового моніторингу, аграрних об'єктів, розумного міста та екологічного контролю.

International Scientific and Practical Conference of Young Scientists and Students "Actual Problems of Automation and Control"

Запропоноване рішення базується на застосуванні мікроконтролера STM32F103C8T6, який завдяки достатньому обсягу пам'яті, високій продуктивності та широким можливостям периферії забезпечує ефективну інтеграцію з модулем LoRa SX1278. У роботі розроблено структурну та принципову електричну схеми пристрою, обрано елементну базу, реалізовано програмну частину прошивки мовою C у середовищі STM32CubeIDE, а також виконано моделювання радіообміну у середовищі Proteus.

Проведено експериментальну перевірку роботи прототипу в умовах міської зашумленості. Встановлено, що пристрій забезпечує стабільну передачу даних на швидкості 2,4 кбіт/с, рівень сигналу RSSI до -110 дБм та споживання у режимі сну на рівні 15 мкА, що підтверджує можливість його довготривалої автономної роботи. Отримані результати свідчать про ефективність розробленого рішення та його придатність до використання у розподілених мережах збору даних, системах моніторингу довкілля, контролю енергоспоживання та інших IoT-застосуваннях.

Ключові слова: STM32, LoRa, бездротовий зв'язок, енергоефективність, IoT, передача даних.

Khvyshchun M. V., Krechyk A. P., Khvyshchun D. M., Rubliov V. Development of a wireless communication device based on STM32. The article presents the development of an energy-efficient wireless communication node based on the STM32 microcontroller family and the LoRa SX1278 radio module, designed for long-distance transmission of telemetry data in Internet of Things (IoT) systems. The relevance of the research is determined by the growing need for low-power autonomous devices capable of providing reliable radio communication in complex electromagnetic and geographical environments without the deployment of bulky infrastructure.

The limitations of traditional wireless technologies (Wi-Fi, BLE, ZigBee) are analyzed, as they demonstrate insufficient communication range or increased power consumption, which complicates their application in distributed sensor networks. The feasibility of using LoRa (Long Range) technology is demonstrated, as it enables data transmission over distances of several kilometers with minimal energy consumption, making it a promising solution for industrial monitoring, agricultural facilities, smart city systems, and environmental control applications.

The proposed solution is based on the STM32F103C8T6 microcontroller, which, due to its sufficient memory capacity, high performance, and wide peripheral capabilities, ensures effective integration with the LoRa SX1278 module. The study encompasses the development of structural and electrical schematic diagrams for the device, the selection of a component base, the implementation of firmware in C programming language using STM32CubeIDE, as well as the simulation of radio communication processes in the Proteus environment.

Experimental testing of the prototype was carried out under conditions of urban electromagnetic noise. It was found that the device provides stable data transmission at a rate of 2.4 kbit/s, an RSSI level of up to -110 dBm, and a sleep-mode power consumption of 15 μ A, confirming the possibility of long-term autonomous operation. The obtained results demonstrate the effectiveness of the developed solution and its suitability for use in distributed data collection networks, environmental monitoring systems, energy consumption control, and other IoT applications.

Keywords: STM32, LoRa, wireless communication, energy efficiency, IoT, data transmission.

Постановка проблеми. У сучасних системах автоматизації, моніторингу та телеметрії одним із ключових викликів є забезпечення стабільного та енергоефективного радіозв'язку на великих відстанях. Традиційні радіотехнології (Wi-Fi, Bluetooth, ZigBee) часто не забезпечують необхідної дальності, надійності або автономності роботи, що обмежує їх використання в умовах промислових об'єктів, аграрного сектору та систем «розумного міста». Для розподілених сенсорних мереж необхідні малопотужні комунікаційні пристрої, здатні працювати місяцями або роками без необхідності обслуговування, а також забезпечувати стійкий зв'язок у складних радіочастотних умовах.

Особливої актуальності набувають технології LPWAN (Low-Power Wide-Area Network), серед яких LoRa (Long Range) є однією з найбільш поширених і ефективних. Завдяки широкосмуговій модуляції CSS та високій чутливості приймача LoRa дозволяє реалізовувати зв'язок на кілька кілометрів, забезпечуючи при цьому мінімальне енергоспоживання. Це робить технологію перспективною для створення бездротових комунікаційних вузлів, призначених для збору даних із сенсорів та передачі інформації до центрального сервера або шлюзу.

International Scientific and Practical Conference of Young Scientists and Students "Actual Problems of Automation and Control"

Таким чином, проблема створення універсального, надійного та енергоефективного бездротового вузла на основі сучасних мікроконтролерів є актуальною та потребує детального технічного опрацювання[1].

Аналіз останніх досліджень. За результатами багатьох досліджень [1–7], системи радіозв'язку типу Wi-Fi, BLE та ZigBee характеризуються обмеженою дальністю, високим енергоспоживанням або необхідністю створення густої інфраструктури ретрансляторів. LPWAN-технології, серед яких LoRaWAN, Sigfox та NB-IoT, дозволяють вирішити ці проблеми, однак потребують відповідної апаратної платформи та оптимізації алгоритмів роботи.

Останні наукові праці демонструють широке застосування LoRa-технології у сферах:

- розподілених сенсорних мереж (WSN);
- моніторингу довкілля;
- аграрних технологій;
- промислових систем контролю;
- телеметрії енергетичних об'єктів.

Мікроконтролери сімейства STM32, згідно з дослідженнями [4], показують високу ефективність завдяки 32-бітній архітектурі ARM Cortex-M, широкому набору периферійних інтерфейсів та підтримці енергозберігаючих режимів, що робить їх оптимальними для IoT-рішень.

Таким чином, об'єднання мікроконтролера STM32 та LoRa SX1278 є перспективним підходом до побудови універсального телеметричного вузла [3–6].

Мета роботи. Розроблення енергоефективного бездротового комунікаційного пристрою для систем Інтернету речей на основі мікроконтролера STM32F103C8T6 та модуля LoRa SX1278, що забезпечує:

- стабільну передачу даних на відстань до 2 км;
- низьке енергоспоживання;
- гнучкість інтеграції із сенсорами;
- можливість автономної роботи протягом тривалого часу.

Викладення основного матеріалу. У межах проведеного дослідження було розроблено енергоефективний бездротовий комунікаційний вузол на основі мікроконтролера STM32F103C8T6 та радіомодуля LoRa SX1278. Основним завданням створеної системи є забезпечення стабільного і надійного каналу зв'язку для передавання телеметричних даних у мережах Інтернету речей (IoT). Пристрій орієнтований на роботу в умовах тривалого автономного живлення та в середовищах зі значним рівнем радіозашумленості [4].

На рисунку 1 представлено структурну схему розробленого комунікаційного пристрою. Центральним елементом системи є мікроконтролер STM32, який виконує функції збору даних від сенсорів, попередньої обробки інформації, формування пакетів та передачі їх через модуль LoRa SX1278 [5–7].

У даній структурній діаграмі пристрою продемонстровано взаємозв'язки між основними модулями:

- STM32 – центральний контролер, що координує роботу системи;
- HTS221 – давач температури та вологості, під'єднаний через PC.
- LoRa SX1278 – радіомодуль, що з'єднаний зі STM32 по SPI та містить UART-інтерфейс для службового обміну;
- стабілізатор живлення – забезпечує стабільну роботу мікроконтролера та модулів;
- акумулятор – повністю автономне джерело живлення пристрою;
- антена – формує канал радіозв'язку для передачі інформації.

Всі функціональні блоки зв'язані між собою стрілками, що позначають напрямок обміну даними або подачу живлення. Схема демонструє логічну структуру пристрою та відображає принципи взаємодії елементів у бездротовому комунікаційному комплексі.

Основний цикл роботи пристрою виглядає так:

1. Ініціалізація STM32 після подачі живлення.
2. Ініціалізація LoRa SX1278 та сенсора HTS221.
3. Перехід у енергозберігаючий режим.
4. Пробудження за таймером.
5. Зчитування показів температури та вологості.
6. Формування пакета даних (payload).
7. Передача даних через LoRa-модуль.
8. Повернення в режим низького енергоспоживання.

International Scientific and Practical Conference of Young Scientists and Students "Actual Problems of Automation and Control"

Такий алгоритм дозволяє мінімізувати витрати енергії й забезпечити тривалу автономну роботу пристрою.

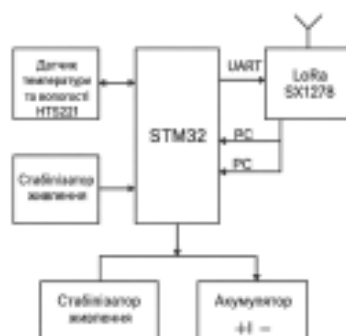


Рисунок 1 – Структурна схема розробленого комунікаційного пристрою

Мікроконтролер сімейства STM32 побудований на 32-бітній архітектурі ARM Cortex-M, що забезпечує:

- високу обчислювальну продуктивність;
- низьке енергоспоживання;
- розвинену периферію, включно з інтерфейсами SPI, UART, PC, АЦП;
- можливість роботи в режимах STOP та STANDBY для економії енергії.

Саме ці особливості роблять STM32 оптимальним рішенням для компактних і автономних телеметричних систем [5].

До STM32 під'єднано цифровий датчик HTS221, призначений для вимірювання температури та відносної вологості повітря.

Взаємодія з давачем відбувається через інтерфейс PC, що забезпечує:

- двопровідне підключення (SDA, SCL);
- можливість паралельного під'єднання кількох сенсорів;
- низьке енергоспоживання та високу стійкість до перешкод.

Отримані дані передаються до STM32 для подальшої обробки.

Модуль LoRa SX1278 використовується для формування вузькосмугового радіосигналу, здатного передавати інформацію на великі відстані (до кількох кілометрів) [7].

Комунікація між STM32 та SX1278 здійснюється через інтерфейс SPI, який забезпечує високошвидкісний обмін. Окремо використовується лінія UART у разі потреби діагностики або передачі службових команд. Антена, під'єднана до LoRa-модуля, формує канал радіозв'язку, що дозволяє передавати пакети даних у середовищах із високими перешкодами.

Пристрій може працювати від автономного джерела – акумулятора. Це забезпечує здатність до тривалої роботи незалежно від зовнішніх джерел живлення.

У схемі передбачено два стабілізатори:

– основний стабілізатор живлення – формує необхідні рівні напруги для мікроконтролера та периферії;

– додатковий стабілізатор – взаємодіє з акумулятором, забезпечує захист і стабільність живлення у випадку переходу на автономний режим.

Це дозволяє системі працювати без збоїв навіть у разі нестабільності мережі або перебоїв у подачі електроенергії.

Апаратна архітектура вузла зв'язку. Основою пристрою є мікроконтролер STM32F103C8T6, побудований на ядрі ARM Cortex-M3, який має:

- тактову частоту до 72 МГц;
- розвинений набір периферії (SPI, PC, UART, ADC);
- підтримку режимів STOP та STANDBY для зменшення енергоспоживання.

У якості радіомодуля обрано LoRa SX1278, що працює у діапазоні 433 МГц та підтримує:

- Spreading Factor (SF) 6–12,
- Bandwidth (BW) 7.8–500 кГц,
- Coding Rate (CR) 4/5–4/8.

International Scientific and Practical Conference of Young Scientists and Students "Actual Problems of Automation and Control"

Між STM32 та SX1278 використано інтерфейс SPI (лінії SCK, MISO, MOSI, NSS), а забезпечення роботи інтерфейсів DIO0 та RESET дозволяє організувати повноцінне керування передавачем і приймачем. Живлення пристрою реалізовано від літій-іонного акумулятора 3,7 В із стабілізацією до 3,3 В. Для довготривалої автономної роботи застосовано DC-DC перетворювач із ККД понад 85%.

Вимірювально-сенсорна інфраструктура. До STM32 можуть підключатися:

- цифрові датчики температури та вологості (I²C);
- датчі освітленості;
- аналогові сенсори (через ADC);
- сенсори контролю стану батарей.

Таким чином, вузол може функціонувати як універсальна платформа збору телеметрії для різних галузей.

Програмна частина. Програмне забезпечення розроблено у STM32CubeIDE і містить:

- налаштування SPI-інтерфейсу для LoRa;
- реалізацію енергозберігаючих режимів;
- алгоритм циклічного пробудження;
- протокол обміну з фіксацією ідентифікатора пристрою;
- модуль перевірки доставлення пакету.

Алгоритм роботи вузла:

1. Ініціалізація мікроконтролера та радіомодуля.
2. Перехід у режим STANDBY.
3. Пробудження за таймером.
4. Опитування сенсорів.
5. Формування LoRa-пакета.
6. Передача даних.
7. Перехід у режим сну.

Експериментальні дослідження. Випробування проводилися в умовах міської забудови.

Отримано такі результати, подані в таблиці 1.

Таблиця 1 – Результати експериментальних досліджень

Параметр	Значення
Дальність зв'язку	до 2 км
Швидкість передачі	2,4 кбіт/с
RSSI	до -110 дБм
Енергоспоживання у сні	15 мкА
Енергоспоживання під час TX	35–45 мА

Отримані значення підтверджують можливість тривалої автономної роботи пристрою та його ефективність у розподілених сенсорних мережах.

Висновки. Запропонований варіант побудови бездротового комунікаційного пристрою на основі STM32F103C8T6 та LoRa SX1278 забезпечує реалізацію ключових функцій сучасних IoT-систем, включаючи збір телеметричних параметрів, передачу даних на великі відстані, енергоефективність та автономність. Практична реалізація підтвердила стабільність радіоканалу в умовах міської радіозашумленості, низьке споживання енергії та можливість використання пристрою як основи для побудови мереж моніторингу довкілля, агропромислових комплексів, систем контролю енергоспоживання та рішень «розумного міста».

ПЕРЕЛІК ПОСИЛАНЬ

1. Pasquale A. J. *Microcontrollers*. Glen Ellyn: College of DuPage, 2021. 195 p.
2. Iqbal M. A., Hussain S., Xing H., Imran M. A. *Enabling the Internet of Things: Fundamentals, Design, and Applications*. Hoboken: John Wiley & Sons, 2021..
3. Dargie W., Poellabauer C. *Fundamentals of Wireless Sensor Networks: Theory and Practice*. Chichester: John Wiley & Sons, 2010. 331 p.
4. Semtech Corporation. SX1276/77/78/79 – LoRa Modem Designer's Guide. Camarillo: Semtech, 2021. 132 p.
5. STMicroelectronics. STM32F103 Reference Manual. Geneva: STMicroelectronics, 2023. 1137 p.

International Scientific and Practical Conference of Young Scientists and Students "Actual Problems of Automation and Control"

6. Adelantado F. et al. Understanding the limits of LoRaWAN. *IEEE Communications Magazine*, 2021, 55(9): p.34–40. doi:10.1109/MCOM.2017.1600613.

7. LoRa Alliance. LoRaWAN® 1.1 Specification. Fremont: LoRa Alliance, 2021. 210 p.

Додаток Б

Лістинг – Глобальні параметри (app_config.h)

```

// app_config.h
#pragma once
// LoRa PHY
#define LORA_FREQ_HZ          868100000UL
#define LORA_BW_HZ           125000UL
#define LORA_SF               7           // 7..12
#define LORA_CR_DEN           5           // CR = 4/5 → 5
#define LORA_TX_POWER_DBM    14
// Таймінги
#define APP_MEAS_PERIOD_S     600        // період
опитування, 10 хв
#define RX1_DELAY_MS         1000
#define RX2_DELAY_MS         2000
// Повтори/ADR
#define N_RETRY_MAX           3
#define ADR_WIN               12        // ковзне вікно
SNR
// Живлення
#define VDD_MV_MIN_TX         3100
Ініціалізація та «скелет» застосунку (main.c):
// main.c (скорочено)
#include "main.h"
#include "app_config.h"
#include "lora.h"
#include "sensors.h"
#include "power_mgmt.h"
#include "log_uart.h"
static void App_Init(void);
int main(void)
{
    HAL_Init();
    SystemClock_Config();
    MX_GPIO_Init();
    MX_SPI1_Init();
    MX_USART2_UART_Init();
    MX_RTC_Init();
    LOG_Init(&huart2);
    LOG("Boot\r\n");
    App_Init();
    for (;;) {
        // 1) Сон до наступного вимірювання або поки не треба
        передати «борги»
        Power_SleepUntilNext(APP_MEAS_PERIOD_S);
        // 2) Підготовка даних

```

```

    app_meas_t m = Sensors_ReadAll();           // запустити
живлення сенсорів усередині
    uint8_t payload[32];
    uint8_t len = Sensors_PackPayload(payload,
sizeof(payload), &m);
    // 3) TX + вікна RX1/RX2 + повтори
    lora_tx_params_t prm = Lora_DefaultTxParams();
    bool delivered = Lora_TxWithAck(payload, len, &prm);
    // 4) Проста ADR-політика (на основі середнього SNR)
    Lora_ADR_Update(&prm);
    // 5) Обслуговування "боргів" у межах duty-cycle
    Lora_TrySendDebts(&prm);
}}
static void App_Init(void){
    // SPI + LoRa
    Lora_Init(&hspi1, LORA_FREQ_HZ);
    Lora_SetModulation(LORA_SF, LORA_BW_HZ, LORA_CR_DEN);
    Lora_SetTxPower(LORA_TX_POWER_DBM);
    // Перехід у Standby для швидкого старту TX/RX
    Lora_SetModeStandby();
}

```

Кінець лістингу

Додаток В

Лістинг – Глобальні параметри (app_config.h)

```

    Інтерфейс (lora.h):
// lora.h (скорочено)
#pragma once
#include <stdbool.h>
#include <stdint.h>
#include "stm32xx_hal.h"
typedef struct {
    uint8_t sf;          // 7..12
    uint32_t bw;        // 125k, 250k, 500k
    uint8_t cr_den;    // 5..8 (CR = 4/den)
    int8_t  eirp_dbm;
} lora_tx_params_t;
void Lora_Init(SPI_HandleTypeDef *hspi, uint32_t
freq_hz);
void Lora_SetModulation(uint8_t sf, uint32_t bw_hz,
uint8_t cr_den);
void Lora_SetTxPower(int8_t dbm);
void Lora_SetModeStandby(void);
bool Lora_TxWithAck(const uint8_t *buf, uint8_t len,
lora_tx_params_t *prm);
void Lora_ADR_Update(lora_tx_params_t *prm);
void Lora_TrySendDebts(lora_tx_params_t *prm);
// Внутрішні/утиліти
void Lora_IrqDio0Handler(void); // викликати з
HAL_GPIO_EXTI_Callback
int16_t Lora_GetLastSNRdB10(void); // SNR×10
    Основні операції (фрагменти lora.c)
// lora.c (фрагменти)
#include "lora.h"
#include "app_config.h"
#include "log_uart.h"
static SPI_HandleTypeDef *g_hspi;
static volatile bool g_txDone = false;
static volatile bool g_rxDone = false;
static inline void spi_select(void) {
HAL_GPIO_WritePin(NSS_GPIO_Port, NSS_Pin,
GPIO_PIN_RESET); }
static inline void spi_deselect(void) {
HAL_GPIO_WritePin(NSS_GPIO_Port, NSS_Pin, GPIO_PIN_SET);
}
static uint8_t reg_read(uint8_t addr) {
    uint8_t tx[2] = { addr & 0x7F, 0 };
    uint8_t rx[2] = {0};
    spi_select();

```

```

    HAL_SPI_TransmitReceive(g_hspi, tx, rx, 2,
    HAL_MAX_DELAY);
    spi_deselect();
    return rx[1];
}
static void reg_write(uint8_t addr, uint8_t val){
    uint8_t tx[2] = { addr | 0x80, val };
    spi_select();
    HAL_SPI_Transmit(g_hspi, tx, 2, HAL_MAX_DELAY);
    spi_deselect();
}
void Lora_Init(SPI_HandleTypeDef *hspi, uint32_t freq_hz)
{
    g_hspi = hspi;
    // RESET pulse апаратно → тут пропускаємо
    // LoRa mode + Sleep → Standby
    reg_write(0x01, 0x80); // LongRangeMode=1, Sleep
    HAL_Delay(5);
    reg_write(0x01, 0x81); // Standby
    // Частота (FRF = Freq * (2^19 / F_XOSC)), спростимо
    під 32 МГц
    // ... тут підставити коректний розрахунок FRF ...
    // Базові адреси FIFO
    reg_write(0x0E, 0x00); // FifoTxBaseAddr
    reg_write(0x0F, 0x80); // FifoRxBaseAddr
    // Налаштування DIO0: TxDone/RxDone
    reg_write(0x40, 0x40); // DIO0=TxDone в TX
    reg_write(0x41, 0x00); // ін.
}
void Lora_SetModulation(uint8_t sf, uint32_t bw_hz,
uint8_t cr_den)
{
    // BW map (125k→0x70, 250k→0x80, 500k→0x90 для SX1276)
    uint8_t bw = (bw_hz == 125000 ? 0x70 : (bw_hz==250000 ?
0x80 : 0x90));
    // CR map (4/5→1, 4/6→2, ...)
    uint8_t cr = (cr_den - 4) & 0x07;
    // RegModemConfig1: BW | CR | ExplicitHeader
    reg_write(0x1D, (bw | (cr<<1) | 0x00));
    // RegModemConfig2: SF | CRC_on | RxPayloadCrcOn
    uint8_t sfbits = (sf<<4) & 0xF0;
    reg_write(0x1E, (sfbits | 0x04)); // CRC on
    // LowDataRateOptimize для SF>=11 при 125 кГц
    uint8_t ldr = ( (sf>=11 && bw_hz==125000) ? 0x08 : 0x00
);
    reg_write(0x26, ldr); // RegModemConfig3
}

```

```

void Lora_SetTxPower(int8_t dbm)
{
    // PA_BOOST → RegPaConfig
    uint8_t p = (uint8_t)(dbm + 2); // спрощено
    reg_write(0x09, 0x80 | (p & 0x0F));
}
static bool tx_packet(const uint8_t *buf, uint8_t len)
{
    // Запис довжини + даних у FIFO
    reg_write(0x0D, 0x00); // FifoAddrPtr
    reg_write(0x22, len); // Payload length
    // burst write
    spi_select();    uint8_t    a    =    0x80    |    0x00;
    HAL_SPI_Transmit(g_hspi, &a, 1, HAL_MAX_DELAY);
    HAL_SPI_Transmit(g_hspi, (uint8_t*)buf, len,
    HAL_MAX_DELAY); spi_deselect();
    // TX mode
    g_txDone = false;
    reg_write(0x01, 0x83); // TX
    // Очікуємо DIO0
    uint32_t t0 = HAL_GetTick();
    while (!g_txDone && (HAL_GetTick()-t0) < 5000) {
        __WFE(); }
    return g_txDone;
}
static bool rx_single(uint16_t timeout_ms)
{
    g_rxDone = false;
    reg_write(0x01, 0x86); // RX single
    uint32_t t0 = HAL_GetTick();
    while (!g_rxDone && (HAL_GetTick()-t0) < timeout_ms) {
        __WFE(); }
    return g_rxDone;
}
bool Lora_TxWithAck(const uint8_t *buf, uint8_t len,
lora_tx_params_t *prm)
{
    // Налаштувати PHY
    Lora_SetModulation(prm->sf, prm->bw, prm->cr_den);
    Lora_SetTxPower(prm->eirp_dbm);
    for (int k=0; k<=N_RETRY_MAX; ++k){
        if (tx_packet(buf, len)) {
            HAL_Delay(RX1_DELAY_MS);
            if (rx_single(400)) return true;
            HAL_Delay(RX2_DELAY_MS - RX1_DELAY_MS);
            if (rx_single(600)) return true;
        }
    }
}

```

```
    // backoff (експоненційний)
    uint32_t tb = 2000U << k;
    HAL_Delay(tb);
}
// додати в «борги»
LOG("TX fail → store debt\r\n");
return false;
}
void Lora_IrqDio0Handler(void)
{
    uint8_t irq = reg_read(0x12); // RegIrqFlags
    if (irq & 0x08) { // TxDone
        g_txDone = true;
    }
    if (irq & 0x40) { // RxDone
        g_rxDone = true;
        // зчитати довжину, RSSI/SNR за потреби ...
    }
    reg_write(0x12, 0xFF); // clear all
}
```

Кінець лістингу

Додаток Г

Лістинг – Вихідний код для передавача LoRa – STM32

```
#include <LoRa_STM32.h>
#define SS PA4
#define RST PB0
#define DI0 PA1
#define TX_P 17
#define BAND 433E6
#define ENCRYPT 0x78
int counter = 0;
void setup() {
  Serial.begin(115200);
  while (!Serial);
  Serial.println("LoRa Sender");
  LoRa.setTxPower(TX_P);
  LoRa.setSyncWord(ENCRYPT);

  LoRa.setPins(SS, RST, DI0);
  if (!LoRa.begin(BAND))
  {
    Serial.println("Starting LoRa failed!");
    while (1);
  }
}
void loop() {
  Serial.print("Sending packet: ");
  Serial.println(counter);
  // send packet
  LoRa.beginPacket();
  LoRa.print("hello ");
  LoRa.print(counter);
  LoRa.endPacket();
  counter++;
  delay(5000);
}
```

Кінець лістингу

Додаток Д

Лістинг – Вихідний код для приймача LoRa – STM32

```
#include <LoRa_STM32.h>
#define SS PA4
#define RST PB0
#define DI0 PA1
#define TX_P 17
#define BAND 433E6
#define ENCRYPT 0x78
void setup()
{
  Serial.begin(115200);
  while (!Serial);
  Serial.println("LoRa Receiver");
  LoRa.setTxPower(TX_P);
  LoRa.setSyncWord(ENCRYPT);

  LoRa.setPins(SS, RST, DI0);
  if (!LoRa.begin(BAND))
  {
    Serial.println("Starting LoRa failed!");
    while (1);
  }
}
void loop() {
  // try to parse packet
  int packetSize = LoRa.parsePacket();
  if (packetSize) {
    // received a packet
    Serial.print("Received packet ");
    // read packet
    while (LoRa.available()) {
      Serial.print((char)LoRa.read());
    }
    // print RSSI of packet
    Serial.print("' with RSSI ");
    Serial.println(LoRa.packetRssi());
  }
}
```

Кінець лістингу