

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

Навчально-науковий інститут телекомунікаційних систем

Кафедра телекомунікацій

До захисту допущено:

В.о.завідувача кафедри

_____ Сергій КРАВЧУК

«__» _____ 2026 р.

Дипломна робота

на здобуття ступеня бакалавра

**за освітньо-професійною програмою «Інженерія та програмування
інфокомунікацій»**

спеціальності 172 «Телекомунікації та радіотехніка»

**на тему: «Моделювання IoT телекомунікацій для створення
завадостійкої системи в умовах міської забудови»**

Виконав:

Студент IV курсу, групи ТЗ-22

Ростовецький Ярослав Павлович _____

Керівник:

Асистент кафедри ТК НН ІТС,

Сайченко Іван Олегович _____

Рецензент:

Доцент кафедри ІТТ НН ІТС, к.т.н., доцент

Правило Валерій Володимирович _____

Засвідчую, що у цій дипломній роботі
немає запозичень з праць інших авторів
без відповідних посилань.

Студент _____

Київ – 2026 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Навчально-науковий інститут телекомунікаційних систем
Кафедра телекомунікацій

Рівень вищої освіти – перший (бакалаврський)

Спеціальність – 172 «Телекомунікації та радіотехніка»

Освітньо-професійна програма «Інженерія та програмування інфокомунікацій»

ЗАТВЕРДЖУЮ

В.о.завідувача кафедри

_____ Сергій КРАВЧУК

«__» _____ 2026 р.

ЗАВДАННЯ

на дипломну роботу студенту

Ростовецькому Ярославу Павловичу

1. Тема роботи «Моделювання IoT телекомунікацій для створення завадостійкої системи в умовах міської забудови», керівник роботи Сайченко Іван Олегович, затверджені наказом по університету від «26» травня 2026 р. № 1912-с.

2. Термін подання студентом роботи 10 червня 20265 р.

3. Вихідні дані до роботи: апаратний вузол на базі Arduino/ESP32; сенсори температури, вологості, освітленості та відстані; протоколи Wi-Fi, LoRa/LoRaWAN і MQTT; алгоритми CRC-перевірки та повторної передачі; умови міської забудови з наявністю завад і втрат пакетів.

4. Зміст роботи:

Проаналізувати особливості роботи IoT-телекомунікаційних систем та їх застосування для передавання сенсорних даних.

Дослідити вплив міської забудови на якість бездротового зв'язку.

Розглянути протоколи зв'язку Wi-Fi, LoRa та MQTT для IoT-систем.

Розробити архітектуру IoT-системи з апаратним вузлом Arduino/ESP32, сенсорами та модулями зв'язку.

Побудувати модель передавання даних в умовах міських завод.

Реалізувати CRC-перевірку цілісності даних і механізм повторної передачі повідомлень.

Порівняти ефективність передавання даних за показниками стабільності зв'язку, кількості втрачених пакетів і надійності доставки.

Перевірити роботу системи шляхом тестування апаратного вузла, симуляційної моделі та веб-інтерфейсу.

5. Перелік ілюстративного матеріалу (із зазначенням плакатів, презентацій тощо):

Слайд 1 Назва

Слайд 2 Назва

І т.д.

6. Дата видачі завдання 10.03.2026 р.

Календарний план

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання етапів дипломної роботи	Примітка
1	Узгодження організаційних питань з науковим керівником	10.03.2026-22.03.2026	Виконано
2	Узгодження теми роботи та початок роботи над першим розділом	22.03.2026-30.04.2026	Виконано
3	Робота над першим розділом	01.04.2026-07.04.2026	Виконано
4	Робота над другим розділом. Проектування архітектури IoT-системи	08.04.2026-11.05.2026	Виконано
5	Розробка моделі передавання даних та алгоритмів завадостійкості	12.05.2026-15.05.2026	Виконано
6	Підготовка третього розділу та тестування системи	16.05.2026-18.05.2026	Виконано
7	Формування висновків щодо ефективності рішення	18.05.2026.19.05.2026	Виконано
8	Оформлення дипломної роботи	19.05.2026-30.05.2026	Виконано
9	Підготовка презентації до захисту	01.06.2026-05.06.2026	Виконано

Студент

Ярослав РОСТОВЕЦЬКИЙ

Керівник

Іван САЙЧЕНКО

РЕФЕРАТ

Дипломна робота містить: 102 сторінки, ____ рисунків, __ таблиць, 33 джерела.

Присвячена розробленню та перевірці IoT-системи для стабільного передавання сенсорних даних у складному радіосередовищі. Роботу виконано в рамках НДР кафедри інформаційних технологій в телекомунікаціях НН ІТС № 0124U001559 «Технології Інтернету Речей для розумного будинку та розумного міста».

Актуальність теми пов'язана з тим, що IoT-пристрої дедалі частіше використовуються для моніторингу параметрів середовища, автоматизації та контролю технічних об'єктів. У міській забудові якість бездротового зв'язку знижується через стіни, перекриття, металеві конструкції, щільність будівель, відбиття сигналу, електромагнітні завади та роботу інших мереж. Це призводить до затримок, спотворення повідомлень, CRC-помилки і втрат пакетів, тому система має не тільки збирати дані, а й контролювати їхню цілісність та повторно передавати повідомлення у разі помилки.

Метою роботи є розроблення та перевірка моделі завадостійкої IoT-телекомунікаційної системи для передавання сенсорних даних в умовах міської забудови. Об'єктом дослідження є процес передавання даних в IoT-телекомунікаційній системі. Предметом дослідження є методи моделювання та підвищення завадостійкості із використанням Wi-Fi, LoRa, MQTT, CRC-перевірки та механізму повторної передачі повідомлень.

Методами дослідження є аналіз літературних джерел, узагальнення матеріалів щодо IoT-телекомунікацій, порівняння Wi-Fi, LoRa та MQTT, структурно-функціональне моделювання архітектури системи, імітаційне моделювання передавання сигналу, алгоритмізація CRC-контролю, експериментальна перевірка апаратного вузла і графічне подання результатів.

У роботі визначено структуру IoT-системи, що поєднує сенсори, мікроконтролер, канал зв'язку, серверну частину, базу даних і веб-інтерфейс. Для побудови системи обґрунтовано використання Wi-Fi для локального

підключення, LoRa/LoRaWAN для передавання коротких повідомлень на більшу відстань і MQTT для обміну між вузлом, брокером, сервером та інтерфейсом користувача.

Запроєктована система побудована на базі Arduino/ESP32 із сенсорами температури, вологості, освітленості та відстані. Апаратний вузол зчитує показники, формує інформаційний пакет, додає CRC-код, контролює номер пакета, передає повідомлення через Wi-Fi або LoRa та фіксує статус доставки. Програмна частина реалізується на основі Flask і SQLite, а веб-інтерфейс використовується для перегляду стану вузлів, сенсорних значень, параметрів каналу та помилок передавання.

Моделювання виконано для Wi-Fi і LoRa з урахуванням відстані, стін, металевих конструкцій, дерев, щільної забудови та інших бездротових мереж. Якість передавання оцінювалася за показниками RSSI, SNR, PDR, PLR, кількістю CRC-помилки і повторних передач. Результати показали, що зі збільшенням відстані та кількості перешкод рівень сигналу знижується, втрати пакетів зростають, а повторна передача стає необхідною умовою стабільної роботи системи.

Наукова новизна роботи полягає у поєднанні моделювання IoT-каналу в умовах міської забудови з практичною реалізацією CRC-контролю, нумерації пакетів і повторної передачі сенсорних повідомлень. Практична цінність полягає у можливості використання запропонованої архітектури для локальних і міських IoT-систем моніторингу.

За результатами перевірки встановлено, що алгоритми завадостійкості підвищують стабільність передавання даних. Для Wi-Fi у складних умовах PDR зріс із 74 % до 89 %, а втрати пакетів зменшилися з 26 % до 11 %. Для LoRa PDR збільшився з 82 % до 90 %, а втрати пакетів знизилися з 18 % до 10 %. Це підтверджує ефективність CRC-контролю та повторної передачі для створення завадостійкої IoT-системи в умовах міської забудови.

Ключові слова: IoT, ІНТЕРНЕТ РЕЧЕЙ, ЗАВАДОСТІЙКІСТЬ, МІСЬКА ЗАБУДОВА, WI-FI, LORA, MQTT, ESP32, CRC-КОНТРОЛЬ,

ПОВТОРНА ПЕРЕДАЧА, СЕНСОРНІ ДАНІ, RSSI, SNR, PDR, PLR.

ABSTRACT

The bachelor's thesis "Modelling of IoT Telecommunications for Creating an Interference-Resistant System in Urban Built-Up Areas" is devoted to the development and testing of an IoT telecommunication system designed to provide stable transmission of sensor data under the influence of noise, interference, packet loss and unstable wireless communication channels.

The relevance of the topic is determined by the active use of IoT systems for environmental monitoring, automation, technical control and real-time data transmission. In urban built-up areas, the quality of wireless communication is affected by walls, floors, metal structures, dense buildings, signal reflections, electromagnetic interference and the operation of other wireless networks. As a result, delays, data distortion, CRC errors and packet loss may occur during sensor data transmission. Therefore, it is important to create an IoT system that can not only collect data from sensors, but also check data integrity, retransmit lost messages and monitor the quality of the communication channel.

The aim of the thesis is to develop and test a model of an interference-resistant IoT telecommunication system for transmitting sensor data in urban built-up areas, taking into account the influence of noise, obstacles, packet loss and instability of the wireless channel.

The object of the research is the process of data transmission in an IoT telecommunication system in urban built-up areas.

The subject of the research is the methods of modelling and improving the interference resistance of an IoT system during sensor data transmission using Wi-Fi, LoRa, MQTT, CRC checking and message retransmission.

The research methods include the analysis of scientific and technical sources, systematization of materials on IoT telecommunications and communication protocols, comparative analysis of Wi-Fi, LoRa and MQTT, structural and functional modelling of the IoT system architecture, simulation of signal transmission in urban conditions, development of CRC checking and

retransmission algorithms, experimental testing of the hardware node and graphical presentation of the obtained results.

The thesis analyses the theoretical foundations of IoT telecommunication systems. It is determined that such a system consists of sensor, microcontroller, network, server and user interface levels. Sensors record environmental parameters, the microcontroller forms data packets, the communication channel transmits them, the server receives and processes data, and the web interface displays the current state of the system. Special attention is paid to the influence of urban buildings on communication quality, since obstacles, signal reflection, noise and channel overload reduce the stability of data transmission.

The system architecture is based on Arduino/ESP32 hardware nodes with sensors for temperature, humidity, illumination and distance measurement. The ESP32 reads sensor values, performs primary data processing, forms packets, adds a CRC code, transmits data via Wi-Fi or LoRa, controls packet numbers and records delivery status. Wi-Fi is used for local data transmission and connection to the server part, LoRa/LoRaWAN is used for transmitting short sensor messages over longer distances, and MQTT is used to organize message exchange between the node, broker, server and web interface.

The software part of the system is implemented using Flask and SQLite. The database stores information about sensor nodes, sensor readings, communication channel parameters and event logs. The system records temperature, humidity, illumination, distance, packet number, RSSI, SNR, CRC status, number of retransmissions and delivery result. The web interface is used to monitor the current state of nodes, sensor values, communication parameters, errors and service messages.

Signal transmission modelling was performed for Wi-Fi and LoRa in urban built-up conditions. The model takes into account the distance between nodes, walls, floors, metal structures, trees, dense buildings and other wireless networks. Transmission quality was evaluated using RSSI, SNR, PDR, PLR, the number of CRC errors and retransmissions. The results showed that with an increase in

distance and the number of obstacles, the signal level decreases, packet loss increases and the need for retransmission becomes higher.

As a result of testing, it was found that the use of interference-resistance algorithms improves data transmission stability. For Wi-Fi in difficult conditions, PDR increased from 74% to 89%, while packet loss decreased from 26% to 11%. For LoRa, PDR increased from 82% to 90%, while packet loss decreased from 18% to 10%. These results confirm the effectiveness of CRC checking, packet numbering and retransmission for creating an interference-resistant IoT system in urban built-up areas.

Keywords: IoT, INTERNET OF THINGS, INTERFERENCE RESISTANCE, URBAN BUILT-UP AREAS, WI-FI, LORA, LORAWAN, MQTT, ESP32, ARDUINO, CRC CHECKING, RETRANSMISSION, SENSOR DATA, RSSI, SNR, PDR, PLR, WEB INTERFACE.

ЗМІСТ

ВСТУП	12
РОЗДІЛ 1 ТЕОРЕТИЧНІ ЗАСАДИ ПОБУДОВИ IoT- ТЕЛЕКОМУНІКАЦІЙНИХ СИСТЕМ В УМОВАХ МІСЬКОЇ ЗАБУДОВИ. 15	
1.1. Поняття, структура та особливості функціонування IoT- телекомунікаційних систем.....	15
1.2. Аналіз сучасних протоколів зв'язку для IoT-систем: Wi-Fi, LoRa та MQTT.....	28
1.3. Вплив міської забудови та завад на якість передавання даних в IoT- мережах	46
Висновки до 1 розділу	63
РОЗДІЛ 2 ПРОЄКТУВАННЯ ЗАВАДОСТІЙКОЇ IoT-СИСТЕМИ ДЛЯ ПЕРЕДАВАННЯ СЕНСОРНИХ ДАНИХ	65
2.1. Розроблення архітектури IoT-системи та вибір апаратних компонентів. 65	
2.2. Обґрунтування вибору протоколів зв'язку та алгоритмів підвищення завадостійкості.....	70
Висновки до 2 розділу	82
РОЗДІЛ 3 МОДЕЛЮВАННЯ, РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ ЗАВАДОСТІЙКОЇ IoT-СИСТЕМИ.....	83
3.1 Моделювання передавання сигналу в умовах міської забудови.....	83
3.2. Реалізація апаратного вузла на базі Arduino/ESP32 та перевірка передавання даних.....	88
3.3. Оцінювання ефективності алгоритмів завадостійкості та тестування роботи системи	93
Висновки до 3 розділу	97
ЗАГАЛЬНІ ВИСНОВКИ ПО РОБОТІ	98
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	101

ПЕРЕЛІК СКОРОЧЕНЬ

IoT	Internet of Things – Інтернет речей
Wi-Fi	Wireless Fidelity – технологія бездротового передавання даних
LoRa	Long Range – технологія далекого бездротового зв'язку з низьким енергоспоживанням
LoRaWAN	Long Range Wide Area Network – мережевий протокол для побудови IoT-мереж на основі LoRa
MQTT	Message Queuing Telemetry Transport – протокол обміну повідомленнями для IoT-систем
CRC	Cyclic Redundancy Check – циклічний надлишковий код для перевірки цілісності даних
ESP32	Мікроконтролер із підтримкою Wi-Fi та Bluetooth, що використовується для побудови IoT-вузлів
RSSI	Received Signal Strength Indicator – показник рівня прийнятого сигналу
SNR	Signal-to-Noise Ratio – співвідношення сигнал/шум
PDR	Packet Delivery Ratio – коефіцієнт доставки пакетів
PLR	Packet Loss Ratio – коефіцієнт втрати пакетів
БД	База даних
ПЗ	Програмне забезпечення
АЦП	Аналого-цифровий перетворювач
IP	Internet Protocol – протокол міжмережевої взаємодії
HTTP	HyperText Transfer Protocol – протокол передавання гіпертекстових даних
API	Application Programming Interface – програмний інтерфейс взаємодії
SQLite	Вбудована система керування базами даних

ВСТУП

Робота IoT-систем у міському середовищі залежить від того, наскільки стабільно сенсорні вузли можуть передавати дані через бездротові канали зв'язку. У реальних умовах такий зв'язок не є постійно однаковим: на нього впливають відстань між пристроями, щільність забудови, стіни, перекриття, металеві конструкції, інші мережі та електромагнітні завади. Через це під час передавання даних можуть виникати затримки, втрати пакетів або спотворення повідомлень, що знижує надійність усієї системи моніторингу. У дипломній роботі розглядається моделювання IoT-телекомунікацій в умовах міської забудови, вибір протоколів зв'язку та створення завадостійкої системи на базі апаратного вузла Arduino/ESP32 із використанням алгоритмів контролю помилок і повторної передачі даних.

Актуальність теми

Актуальність теми пов'язана з тим, що IoT-пристрої часто використовуються для збирання та передавання даних у складному радіосередовищі, де якість зв'язку може швидко змінюватися. Для міської забудови характерні послаблення сигналу, багатопроменеве поширення, відбиття від будівель, накладання сигналів від інших бездротових пристроїв і періодичні втрати пакетів. Якщо система не має механізмів перевірки цілісності повідомлень і повторної передачі, навіть простий обмін сенсорними даними може бути нестабільним. Тому важливим є моделювання роботи IoT-каналу, порівняння протоколів Wi-Fi, LoRa та MQTT, а також перевірка ефективності CRC-контролю й повторної передачі. Такий підхід дає змогу оцінити, які технічні рішення підвищують завадостійкість IoT-системи та забезпечують більш надійну передачу даних у міських умовах.

Мета дослідження – розробити та перевірити модель завадостійкої IoT-телекомунікаційної системи для передавання сенсорних даних в умовах міської забудови з урахуванням впливу шуму, перешкод, втрат пакетів і нестабільності бездротового каналу зв'язку.

Завдання дослідження:

1. Проаналізувати особливості роботи IoT-телекомунікаційних систем та їх застосування для передавання сенсорних даних.
2. Дослідити вплив міської забудови на якість бездротового зв'язку, зокрема послаблення сигналу, відбиття, шум і втрату пакетів.
3. Розглянути протоколи зв'язку, які можуть використовуватися в IoT-системах, зокрема Wi-Fi, LoRa та MQTT.
4. Розробити архітектуру IoT-системи з апаратним вузлом на базі Arduino або ESP32, сенсорами та модулями зв'язку.
5. Побудувати модель передавання даних в умовах міських завод і виконати імітацію роботи телекомунікаційного каналу.
6. Реалізувати алгоритми підвищення завадостійкості, зокрема CRC-перевірку цілісності даних і механізм повторної передачі повідомлень.
7. Порівняти ефективність різних варіантів передавання даних за показниками стабільності зв'язку, кількості втрачених пакетів і надійності доставки повідомлень.
8. Перевірити роботу розробленої системи шляхом тестування апаратного вузла, симуляційної моделі та, веб-інтерфейсу для моніторингу параметрів системи.

Об'єкт дослідження – процес передавання даних в IoT-телекомунікаційній системі в умовах міської забудови, де на якість зв'язку впливають перешкоди, шум, відбиття сигналу та втрати пакетів.

Предмет дослідження – методи моделювання та підвищення завадостійкості IoT-системи під час передавання сенсорних даних із використанням протоколів Wi-Fi, LoRa, MQTT, CRC-перевірки та механізму повторної передачі повідомлень.

Методи дослідження: аналіз літературних джерел; систематизація та узагальнення матеріалів щодо IoT-телекомунікацій, протоколів зв'язку та завадостійкості; порівняльний аналіз протоколів Wi-Fi, LoRa та MQTT; структурно-функціональне моделювання архітектури IoT-системи; імітаційне

модельовання передавання сигналу в умовах міської забудови; експериментальний метод; метод алгоритмізації для побудови механізмів CRC-перевірки та повторної передачі; метод тестування апаратного вузла; графічний метод для подання схем, моделей і результатів перевірки системи.

Структура роботи – Робота складається зі вступу , трьох розділів, висновків , списку використаних джерел та додатків.

РОЗДІЛ 1

ТЕОРЕТИЧНІ ЗАСАДИ ПОБУДОВИ ІОТ-ТЕЛЕКОМУНІКАЦІЙНИХ СИСТЕМ В УМОВАХ МІСЬКОЇ ЗАБУДОВИ

1.1. Поняття, структура та особливості функціонування ІоТ-телекомунікаційних систем

Під ІоТ-системою розуміють сукупність фізичних пристроїв, які мають датчики, засоби обробки даних і канали зв'язку для передавання інформації між собою або до віддалених сервісів. У такій системі технічний об'єкт стає елементом мережі: він може фіксувати параметри середовища, передавати отримані дані, отримувати команди й брати участь в автоматизованому керуванні. Б. Ю. Жураковський та І. О. Зенів визначають Internet of Things як глобальну мережу підключених до Інтернету пристроїв, оснащених сенсорами, датчиками та засобами передавання сигналів. За поясненням авторів, такі пристрої сприймають сигнали з навколишнього середовища, взаємодіють з іншими пристроями, обмінюються даними для віддаленого моніторингу, аналізу інформації та прийняття рішень [4, с. 8].

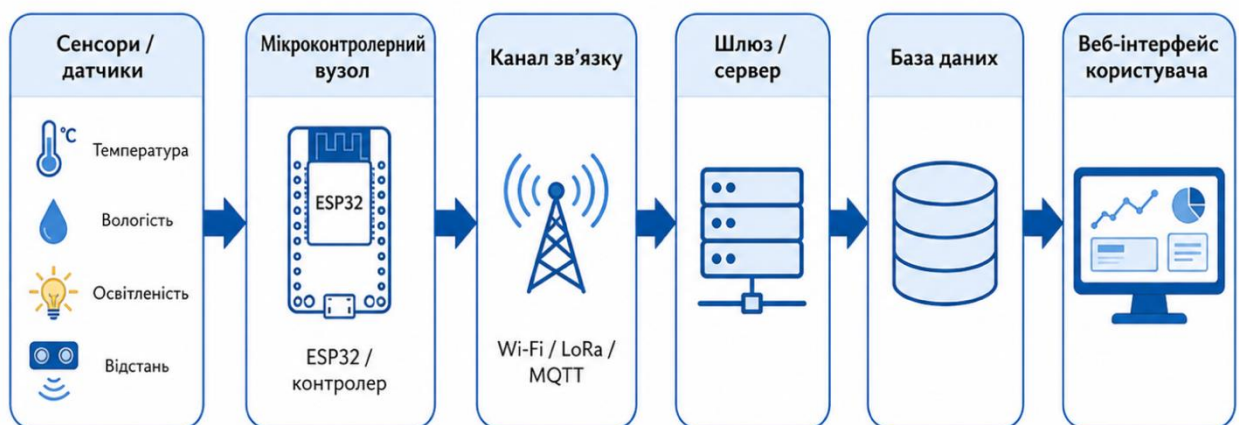


Рис. 1.1 – Узагальнена структура ІоТ-телекомунікаційної системи

IoT-система не зводиться лише до датчика або електронного модуля. Її робота будується як інформаційний ланцюг: сенсор фіксує фізичний параметр, мікроконтролер обробляє сигнал, канал зв'язку передає дані, а сервер, база даних або веб-інтерфейс забезпечують їх зберігання й відображення. Наприклад, датчик температури сам по собі тільки вимірює показник, але після підключення до Arduino або ESP32 і передавання даних через Wi-Fi, LoRa чи інший канал він стає частиною IoT-телекомунікаційної системи.

Поява поняття IoT пов'язується з ідеєю Кевіна Ештона, який у 1999 р. запропонував термін «Інтернет речей». Ідея полягала в тому, що фізичні речі можуть мати цифрове представлення в інформаційному середовищі, тобто фактично отримувати власний цифровий образ [4, с. 8]. У цьому проявляється технічна сутність IoT: пристрій існує не тільки як фізичний об'єкт, а й як джерело даних, яке включається в мережеву взаємодію. До прикладів таких об'єктів належать гаражні двері, кавоварки, телевізори, мобільні телефони, відеокамери, датчики світла й температури [4, с. 8].

Масштабність розвитку IoT підтверджується кількісними даними. За даними Ericsson, у світі налічувалося понад 16 млрд пристроїв, підключених до Інтернету, а у 2018 р. їх кількість уже перевищила кількість мобільних телефонів. Прогнозувалося, що до 2022 р. кількість підключених пристроїв могла досягти 29 млрд, з яких 18 млрд мали становити саме IoT-пристрої [4, с. 8]. Такі показники пояснюють, чому стабільне передавання даних стає важливим для телекомунікаційної інфраструктури: велика кількість підключених вузлів збільшує навантаження на мережу, ризик затримок, втрат пакетів і взаємних завад.

Практичне значення IoT добре видно на прикладах віддаленого моніторингу. Переносні датчики можуть використовуватися в медицині для дистанційного спостереження за станом пацієнтів і реагування на зміни в режимі реального часу. Також описано систему компанії AT&T у США, яка фіксує падіння літньої людини та автоматично зв'язується з call-центром для

надання допомоги [4, с. 9]. У таких випадках якість зв'язку має пряме значення: якщо повідомлення від датчика не буде доставлене або надійде із затримкою, система не зможе виконати свою функцію вчасно.

IoT виступає частиною ширшої інтернет-інфраструктури, що поєднує Internet of People, Internet of Things, Internet of Media та Internet of Services [4, с. 10]. Тобто IoT не працює ізольовано, а взаємодіє з користувачами, сервісами, платформами обробки даних і телекомунікаційними каналами. Розвиток таких систем пов'язаний із потребою у швидкісному Інтернеті, мережах п'ятого покоління, зменшенні затримки передавання даних, підтримці великої кількості підключень і збільшенні строку роботи «розумних» пристроїв до 10 років [4, с. 10].

З технологічної точки зору IoT описується як «мережа мереж», що складається з унікально ідентифікованих об'єктів, здатних взаємодіяти між собою через IP-підключення без постійної участі людини [4, с. 11]. У цьому полягає його місце в телекомунікаційній інфраструктурі: для роботи IoT потрібні адресація, маршрутизація, протоколи передавання даних, шлюзи, сервери та засоби обробки інформації. Промисловий Інтернет відрізняється від звичайного збору даних тим, що датчики поєднуються з аналітичними або керуючими системами, а отримана інформація використовується для автоматичного прийняття рішень [4, с. 11].

Архітектура IoT охоплює датчики, виконавчі механізми, Edge-рівень, мережу Internet, платформу обробки подій, хмарні або туманні обчислення та кінцеві застосунки. Дані, отримані від sensors та actuators, передаються на рівень обробки, де вони зберігаються, аналізуються, перенаправляються до потрібних додатків і можуть використовуватися для машинного навчання [4, с. 18–19]. У цій структурі сенсорний вузол є лише початковою ланкою, а телекомунікаційний канал забезпечує передавання даних до наступних рівнів системи.

Передавання даних є однією з базових умов існування IoT. Основою Інтернету речей є не тільки датчики або програмні додатки, а можливість

встановити з'єднання [4, с. 20]. У роботі IoT можуть використовуватися різні типи мереж і протоколів: PAN, Bluetooth, mesh-мережі, Zigbee, Z-Wave, Wi-Fi на основі IEEE 802.11, 6LoWPAN, Thread, 3G, 4G LTE, Cat-1, Cat-NB, LoRaWAN і Sigfox [4, с. 21]. Для обміну даними між вузлами, маршрутизаторами та хмарними сервісами застосовуються MQTT, AMQP і CoAP [4, с. 23].

Практичний рівень побудови IoT-вузлів розкрито у працях К. П. Сторчака, А. М. Тушича, І. М. Срібної, Н. Д. Яковенко та Д. В. Кравця. Автори досліджували роботу з Arduino, датчиками, електронними схемами, індикаторами та інтерфейсами передавання інформації [11, с. 4–5]. Arduino подано як платформу зі зручною архітектурою та програмними кодами для створення електронних пристроїв [11, с. 5]. У складі Arduino RFID Kit наведено плату Arduino Uno R3, макетну плату, датчики, резистори, діоди та інші елементи, що відповідає типовій структурі навчального IoT-вузла [11, с. 5–6].

Конкретні характеристики компонентів показують, що навіть простий IoT-вузол складається з кількох технічних рівнів. Макетна плата MB-102 містить 830 точок і 200 входів для підключення живлення, LCD-дисплей має дві строчки по 16 символів і працює від напруги 5 В, а RFID-модуль працює від 3,3 В і підключається через інтерфейс SPI [11, с. 6–7]. Така структура поєднує мікроконтролер, сенсорні елементи, живлення, інтерфейси підключення та програмну частину.

Сенсорний рівень представлений різними типами датчиків. Описано цифровий датчик температури, датчик рівня рідини з робочим живленням 5 В, датчик температури й вологості з ємнісним сенсором вологості та термістором, а також аналоговий датчик температури з діапазоном вимірювання від -55 до 150 °C [11, с. 8–9]. Для DHT11 наведено такі параметри: живлення 3,5–5 В, вимірювання температури від 0 до 50 °C з точністю 2 °C, вимірювання вологості від 20 % до 95 % з точністю 5 % [11, с.

46]. Підключення DHT11 виконується через бібліотеку DHT.h, а дані після зчитування можуть виводитися через послідовний порт [11, с. 47–48].

Ще одним прикладом сенсорного пристрою є ультразвуковий далекомір HC-SR04. Він вимірює відстань від 1–2 см до 4–6 м, а точність становить 0,5–1 см. У технічних характеристиках також наведено діапазон вимірювання від 2 до 500 см, точність 0,3 см, робочий кут менше 15°, кут вимірювання 30°, напругу живлення 5 В, струм спокою менше 2 мА та робочий струм 15 мА [11, с. 59–60]. Це показує, що IoT-вузол може працювати з різними типами даних: температурою, вологістю, відстанню, рівнем рідини або ідентифікаційною інформацією.

RFID як технологія радіочастотної ідентифікації використовує радіохвилі для автоматичного розпізнавання об'єктів. RFID-система складається зі зчитувача, мітки та програмного забезпечення [11, с. 49]. Дальня ідентифікація працює на відстані від 5 до 300 м, середня – від 20 см до 5 м, ближня – до 20 см. Пасивні RFID-мітки можуть працювати на відстані до 8 м, а активні – до 300 м і можуть передавати сигнал навіть через воду або метал [11, с. 49–50]. Такі рішення можуть включати не тільки ідентифікацію об'єкта, а й передавання інформації про його стан або умови середовища.

Сутність IoT-систем полягає у поєднанні фізичних пристроїв, сенсорів, мікроконтролерів, каналів зв'язку, шлюзів, серверних платформ, баз даних і користувацьких інтерфейсів в єдину телекомунікаційну інфраструктуру.

Архітектура IoT-системи будується як послідовний ланцюг передавання даних від фізичного об'єкта до програмного середовища, де ці дані зберігаються, обробляються і відображаються для користувача. На початковому рівні працюють датчики, які фіксують температуру, вологість, рух, освітленість, рівень рідини або інші параметри. Далі мікроконтролер обробляє отриманий сигнал і передає його через канал зв'язку до шлюзу, сервера, хмарної платформи або веб-інтерфейсу. У такій структурі сенсорний вузол виступає не окремим пристроєм, а частиною телекомунікаційної системи.

Г. В. Антонова, А. В. Кедич та О. В. Ковирьова розглядали IoT і бездротові смарт-мережі на прикладі точного землеробства. Автори показують, що сенсори можуть розміщуватися стаціонарно або на рухомих об'єктах, а отримані дані використовуються для оперативного інформування, відображення в хмарному середовищі та прийняття управлінських рішень [2, с. 120]. Така логіка підходить і для міського IoT-моніторингу: датчик фіксує параметр, мережа передає його далі, а програмна частина перетворює дані на зрозумілий результат.

Організація обміну сенсорними даними залежить від кількості вузлів, типу датчиків, обсягу повідомлень, частоти передавання, енергоспоживання та вибраного протоколу. У бездротових сенсорних мережах значення мають відстань між вузлами, наявність перешкод, логічна структура мережі, маршрути передавання та тип бездротового з'єднання [2, с. 123–124]. Для умов міської забудови ці чинники особливо важливі, оскільки сигнал може слабшати через стіни, металеві конструкції, інші мережі та перевантаження радіоканалу.

Таблиця 1.1

Основні елементи архітектури IoT-системи можна подати так:

Елемент архітектури	Роль у системі	Зв'язок із передаванням сенсорних даних
Датчики та сенсори	Фіксують фізичні параметри середовища або стан об'єкта	Формують первинні дані для передавання [2, с. 120]
Мікроконтролер	Приймає сигнал від датчика та виконує первинну обробку	Готує дані до передавання через Wi-Fi, LoRa або інший канал
Канал зв'язку	Забезпечує передавання повідомлень між вузлами системи	Визначає стабільність, затримку і втрати пакетів
Шлюз	Поеднує локальні IoT-пристрої з основною мережею	Передає дані від сенсорного рівня до серверної частини [7, с. 34]
Сервер або хмарна платформа	Зберігає, обробляє й аналізує отримані дані	Забезпечує доступ до результатів моніторингу [7, с. 34]
Веб-інтерфейс	Відображає параметри системи для користувача	Дає змогу контролювати стан датчиків і якість каналу

А. В. Погребняк, В. О. Яковенко, В. Ю. Клим та Т. Ю. Яковенко подають IoT-архітектуру через сенсорне обладнання, шлюзи, хмарні сервери

та програмні платформи. Сенсорне обладнання збирає дані з оточення, шлюзи з'єднують IoT-пристрої з основними мережами, хмарні сервери забезпечують зберігання й аналіз, а програмні платформи відповідають за керування пристроями та обробку інформації [7, с. 34]. Такий поділ добре показує, що IoT-система працює не на одному рівні, а як сукупність апаратних, мережових і програмних компонентів.

Рівнева структура IoT у цих авторів охоплює рівень пристроїв, шлюзовий рівень, хмарний рівень та аналітичний рівень. Перший рівень відповідає за фізичне збирання даних, другий – за передавання до мережі, третій – за зберігання та обробку, четвертий – за перетворення інформації на рішення для користувача [7, с. 34]. У системі міського моніторингу така структура може виглядати так: датчик збирає дані, ESP32 або Arduino передає їх через Wi-Fi чи LoRa, сервер зберігає показники, а веб-панель відображає температуру, вологість, рівень сигналу або кількість втрачених пакетів.

Вибір технології зв'язку залежить від завдань системи. За порівнянням мережових технологій, Wi-Fi має високу швидкість передавання, але обмежену кількість підключень; LoRaWAN підтримує велику кількість пристроїв і підходить для автономних сенсорних мереж, але має нижчу швидкість; 5G забезпечує дуже високу швидкість і велику кількість підключень, однак потребує складнішого управління [7, с. 34]. Це пояснює, чому в IoT-системах часто комбінують різні протоколи: один зручний для швидкого локального передавання, інший – для далекого зв'язку та низького енергоспоживання.

Таблиця 1.2

Особливості передавання даних у різних сценаріях функціонування IoT-систем

Сценарій IoT	Характер передавання даних	Технічна особливість
«Розумний дім»	Невеликі обсяги даних, помірні вимоги до затримки	Підходить для Wi-Fi, Bluetooth, MQTT
Міський моніторинг	Періодичне передавання даних від багатьох вузлів	Важлива стабільність каналу та масштабованість

Інтелектуальний транспорт	Високі вимоги до швидкості реакції	Критичне значення має мала затримка [7, с. 35]
Промисловий IoT	Велика кількість даних і високі вимоги до надійності	Потрібні стійкі канали й контроль помилок [7, с. 35]

Н. І. Праворська та Н. В. Грипинська звертають увагу на серверну частину обміну даними. Зі збільшенням кількості пристроїв для моніторингу та керування може виникати перевантаження мережі й зменшення пропускної спроможності [8, с. 193]. Локальний IoT-модуль передає дані на головний сервер періодично або після певної події. Якщо кількість підключень перевищує можливості каналу, з'являються затримки, повторні передачі й ризик відмови в обслуговуванні [8, с. 193].

Для організації такого обміну розглядаються IPv6, 6LoWPAN, MQTT, ZeroMQ і туманні обчислення. IPv6 важливий через обмеження IPv4, який має 4,3 млрд унікальних адрес; 6LoWPAN використовується для підключення малопотужних пристроїв до IP-мереж; MQTT і ZeroMQ забезпечують перенесення даних; туманні обчислення зменшують навантаження на Інтернет, бо частина обробки переноситься ближче до пристроїв – на маршрутизатори, комутатори або інші граничні елементи [8, с. 193–194].

Для оцінювання передавання через Wi-Fi автори використовують модель Б'янкі, яка враховує пропускну здатність і ймовірність зіткнення пакетів [8, с. 195–196]. Результати моделювання показали, що при кількості станцій до 1000 швидкість передавання даних може зменшуватися до трьох разів, а затримка передавання не перевищує 10 секунд [8, с. 196]. Для IoT-системи в умовах міської забудови це важливий результат: навіть за правильної роботи датчиків загальна надійність залежить від того, чи витримує канал зв'язку одночасну роботу багатьох вузлів.

О. М. Трофимчук та В. Є. Іцкович описують сервіс-орієнтовану мережу для отримання даних про стан довкілля. Інформація передається від давачів-джерел до серверної частини, де виконується накопичення, обробка та подальше використання показників [12, с. 117–119]. Такий підхід близький

до IoT-моніторингу міського середовища: датчики можуть збирати параметри повітря, температури, вологості або інших показників, а серверна система забезпечує їх подальший аналіз.

Отже, архітектура IoT-системи включає сенсорний, мережевий, серверний та аналітичний рівні. На сенсорному рівні формуються первинні дані, мережевий рівень відповідає за їх передавання, серверний – за зберігання й обробку, а аналітичний або веб-рівень забезпечує відображення результатів. Для завадостійкої системи в умовах міської забудови головним є не лише зчитування показників, а й стабільна доставка повідомлень через канал зв'язку без критичних затримок, втрат пакетів і перевантаження мережі.

Надійність IoT-телекомунікацій залежить не тільки від якості датчиків або мікроконтролерів, а й від стабільності передавання даних, захищеності каналу зв'язку, цілісності повідомлень і здатності системи працювати при зміні умов середовища. Для IoT-систем це особливо важливо, оскільки сенсорні вузли часто працюють автономно, мають обмежені енергетичні та обчислювальні ресурси, а передавання даних відбувається через бездротові канали, які можуть бути вразливими до перешкод, перехоплення або навмисного впливу.

М. Б. Александер, С. М. Балабан, М. П. Карпінський, С. А. Райба та В. М. Чиж розглядають безпроводові сенсорні мережі як основу багатьох автоматизованих систем моніторингу. До їхніх переваг автори відносять низьку вартість і малі габарити інформаційних вузлів, швидке розгортання, високу енергетичну ефективність, масштабованість, самоналаштування та самовідновлення. Окремо зазначено, що висока надійність і відмовостійкість досягаються за рахунок комунікаційної надлишковості, тобто наявності альтернативних маршрутів передавання даних [1, с. 4–5].

Для IoT-систем у міській забудові це має пряме значення. Якщо один канал передавання погіршується через перешкоди, перевантаження або ослаблення сигналу, система має зберігати можливість доставити дані іншим

маршрутом або повторити передавання. У безпроводових сенсорних мережах відстань між інформаційними вузлами може становити від 1 м до 10 км, а сама мережа може охоплювати як одну кімнату, так і населений пункт або адміністративну територію [1, с. 6–7]. Тому надійність залежить від масштабу мережі, щільності вузлів, способу маршрутизації та стійкості радіоканалу.

Безпека IoT не обмежується захистом одного пристрою. Вона охоплює фізичний рівень, канал зв'язку, маршрутизацію, серверну частину, програмні сервіси та доступ користувачів. О. Є. Коваленко визначає мету моделі безпеки IoT через забезпечення конфіденційності, цілісності та доступності даних, які передаються між пристроями, а також захист кінцевих користувачів [5, с. 43]. У цій логіці конфіденційність означає, що дані не повинні бути доступні стороннім особам; цілісність – що повідомлення не має змінюватися під час передавання; доступність – що система повинна залишатися працездатною для легальних користувачів.

Основні проблеми безпеки IoT пов'язані зі слабкою або відсутньою автентифікацією, недостатнім захистом програмного забезпечення, слабкою мережевою безпекою, обмеженою фізичною безпекою пристроїв, неналежним захистом даних, складністю оновлення пристроїв, недостатньою видимістю та контролем, а також застарілими методами захисту [5, с. 45]. Для сенсорних вузлів це особливо помітно: пристрій може бути технічно простим, але через слабкий пароль, незашифрований канал або відсутність оновлень він стає точкою входу для атаки на всю систему.

Таблиця 1.3

Основні ризики безпеки та засоби захисту IoT-систем

Напрямок загрози	Як проявляється в IoT-системі	Засоби захисту
Фізичний доступ до вузла	пошкодження датчика, підміна пристрою, зняття даних із пам'яті	фізичний захист, обмеження доступу, контроль справності [9, с. 88–89]
Порушення каналу зв'язку	перехоплення, MITM, DoS/DDoS, втрата пакетів	TLS/SSL, IPsec, фільтрація пакетів, ACL, брандмауер [9, с. 89–90]

Порушення цілісності даних	зміна повідомлення, підrobка даних, replay-атаки	автентифікація, цифровий підпис, шифрування, контроль цілісності [13, с. 130–131]
Слабке ПЗ пристроїв	застаріла прошивка, відсутність оновлень, відкриті вразливості	оновлення ПЗ, безпечне налаштування, контроль версій [5, с. 47]
Централізована обробка	ризик єдиної точки відмови, перевантаження хмари	туманні обчислення, розподілені вузли, децентралізація [13, с. 132]

Л. А. Савицька, Т. І. Коробейнікова, О. І. Костюк, І. С. Колесник та О. В. Дудник звертають увагу на те, що кількість підключених пристроїв швидко зростає: у 2018–2019 рр. вона вже перевищила населення Землі, а до 2025 р. прогнозувалося від 50 до 70 млрд підключених пристроїв [9, с. 83]. Таке зростання збільшує ризики для інформації, яка збирається, передається та зберігається в IoT-системах. Автори також пов'язують це з вимогами захисту персональних даних, зокрема GDPR [9, с. 83].

У запропонованому підході до безпечного дизайну IoT виділено чотири рівні: рівень датчиків, рівень локальних інтерфейсів, рівень корпоративної комп'ютерної мережі та службовий рівень [9, с. 86]. Такий поділ зручний для аналізу завадостійкої IoT-системи, оскільки кожен рівень має свої ризики. На рівні датчиків головними проблемами є фізична безпека, порушення цілісності даних і зловмисне ПЗ. На рівні локальних інтерфейсів виникають ризики MAC-, DNS-, ARP-spoofing, RFID- і WSN-атак. На рівні корпоративної мережі розглядаються DoS, DDoS, MITM, проблеми IPv4/IPv6, MQTT, DDS, TLS/SSL, TCP і UDP [9, с. 89–90].

Для рівня датчиків важливими є автентифікація, шифрування даних, обмеження локального зберігання інформації та можливість оновлення пристроїв. Якщо пристрій не має доступу до оновлення прошивки або налаштувань, нові вразливості залишаються не виправленими й послаблюють безпеку всієї системи [9, с. 89]. На рівні корпоративної мережі головними вимогами залишаються конфіденційність, цілісність і доступність, а також захист від надмірної кількості підключень, несанкціонованого доступу й атак типу Man-in-the-Middle [9, с. 90].

Л. В. Чепель та Ю. В. Бойко підкреслюють, що для IoT немає універсального механізму безпеки через різноманітність пристроїв і їхню апаратну обмеженість. Автори наводять показовий приклад: протокол MQTT має 129 згадок щодо вразливостей у базі CVE, причому значна частина цих вразливостей стосується програмної реалізації [13, с. 130]. Це означає, що безпека IoT не може будуватися лише на виборі популярного протоколу. Потрібно враховувати реалізацію, налаштування, серверну частину, канал зв'язку та поведінку пристрою в мережі.

Атаки в IoT можуть стосуватися маршрутизації, RFID, MITM, DoS, спаму, підміни QR-кодів та інших елементів системи. Для атак маршрутизації, зокрема wormhole, Sybil, selective forwarding, blackhole, sinkhole, replay, spoofing і hello flood, пропонуються розподілені мережі, оцінювання ризиків у критичних зонах і шифрування з використанням еліптичних кривих. Для MITM та DoS до засобів захисту віднесено фільтрування пакетів, ACL і шифрування, а для спам-атак – цифровий підпис [13, с. 130].

Окремою проблемою є обмеженість кінцевих пристроїв. Для менш потужних вузлів може застосовуватися DTLS у поєднанні з CoAP або MQTT, оскільки DTLS побудований на UDP і є швидшим та менш ресурсоємним порівняно з TCP [13, с. 131]. Водночас використання сертифікатів може створювати надмірне навантаження на пам'ять і ресурси пристрою. Якщо зломисник фізично отримає доступ до незашифрованої пам'яті, він може витягти сертифікат для з'єднання із сервером і підмінити пристрій [13, с. 131]. Тому захист IoT-вузла має поєднувати криптографію, контроль фізичного доступу та виявлення аномальної поведінки.

Для обмежених внутрішніх мереж автори виділяють ZigBee, 6LoWPAN, Z-Wave і BLE, а для глобальних мереж – Wi-Fi, Ethernet, 3G/4G/5G із використанням TLS/DTLS [13, с. 131]. При цьому кожна технологія має власні ризики: ZigBee не гарантує комплексної безпеки, 6LoWPAN використовує симетричне шифрування без механізму ротації

ключів і може бути вразливим до DoS-атак, BLE використовує AES-128, але такий захист також може розглядатися як потенційно вразливий у перспективі складніших атак [13, с. 131].

Для підвищення надійності обробки та зберігання даних розглядається не тільки хмарна модель, а й туманні обчислення. Дані в класичних IoT-системах часто передаються у хмару, але при цьому залишаються ризики єдиної точки підключення, MITM-атак і DoS-атак. Туманні обчислення передбачають зберігання й обробку даних на ближчих вузлах, що зменшує залежність від одного центрального вузла і може використовуватися в розумних містах, роботизованих фабриках та інших локально організованих системах [13, с. 132].

Блокчейн розглядається як додатковий рівень безпеки для IoT, оскільки він може зменшити ризики модифікації, видалення або підміни даних навіть за наявності скомпрометованого каналу передавання. За рахунок консенсус-алгоритмів ускладнюється додавання фальшивої інформації або підробленого вузла до мережі [13, с. 133]. Разом із цим блокчейн має обмеження: не всі IoT-пристрої мають достатню потужність для роботи повноцінного блокчейн-вузла або складних алгоритмів цифрового підпису, а зі збільшенням мережі ускладнюється узгодження нових блоків [13, с. 134].

Для малопотужних IoT-вузлів запропоновано гібридний підхід: класичні методи захисту на локальному рівні поєднуються з блокчейн-елементами. Один із наведених прикладів – механізм Blockchain Dynamic Table з AES-шифруванням, який може працювати на енергоефективному мікроконтролері з частотою 16 МГц і 8 КБ оперативної пам'яті [13, с. 134]. Такий приклад показує, що навіть для пристроїв з обмеженими ресурсами можна будувати додатковий рівень захисту, якщо рішення не перевантажує апаратну частину.

Таким чином, надійність і безпека IoT-телекомунікацій формуються на кількох рівнях: фізичному, мережевому, програмному, серверному та організаційному. Праці М. Б. Александера, С. М. Балабана, М. П.

Карпінського, С. А. Райби та В. М. Чижа показують значення відмовостійкості, альтернативних маршрутів і класифікації атак на безпроводові сенсорні мережі [1, с. 4–5; 55–58]. О. Є. Коваленко акцентує увагу на конфіденційності, цілісності, доступності та багаторівневих моделях безпеки IoT [5, с. 43–48]. Л. А. Савицька, Т. І. Коробейнікова, О. І. Костюк, І. С. Колесник та О. В. Дудник деталізують захист за рівнями технологічного дизайну IoT [9, с. 86–90], а Л. В. Чепель та Ю. В. Бойко розкривають ризики протоколів, обмеження малопотужних пристроїв і можливості блокчейн-підходу [13, с. 130–134]. Для завадостійкої IoT-системи в умовах міської забудови ці положення означають, що стабільне передавання даних має поєднуватися з автентифікацією, шифруванням, контролем цілісності, захистом від атак і здатністю системи працювати при зміні стану каналу зв'язку.

1.2. Аналіз сучасних протоколів зв'язку для IoT-систем: Wi-Fi, LoRa та MQTT

Wi-Fi та інші бездротові мережі ближнього радіусу дії в IoT-системах найчастіше використовуються тоді, коли сенсорний вузол працює в межах будівлі, лабораторії, виробничого приміщення або локальної міської зони. Їхня перевага полягає у простому підключенні до вже наявної мережевої інфраструктури, можливості передавати дані на локальний сервер або веб-інтерфейс і підтримці стандартних інтернет-протоколів. Разом із цим така мережа залежить від якості радіоканалу, кількості підключених вузлів, пропускну здатності, затримки та втрат пакетів.

Y. Chen та T. Kunz досліджували роботу IoT-протоколів у бездротовій мережі з обмеженою якістю зв'язку. Їхній експеримент був побудований на медичній IoT-системі, де сенсори пацієнта передавали дані на локальний шлюз, далі – на центральний сервер, а потім інформація ставала доступною для пристрою користувача або медичного працівника [19, с. 1]. Така схема добре відповідає загальній логіці IoT: сенсорний вузол збирає дані, шлюз

передає їх через бездротову мережу, а сервер забезпечує обробку та подальший доступ.

Апаратна частина експерименту включала Arduino Uno revision 3, набір eHealth sensor kit revision 2, Raspberry Pi model 2 як шлюз пацієнта, центральний сервер на ноутбучі ASUS Zenbook з Ubuntu у VirtualBox та маршрутизатор Linksys для створення бездротової мережі між шлюзом, сервером і клієнтським застосунком [19, с. 3]. Сенсорний набір містив датчик частоти серцевих скорочень і насичення крові киснем, температурний датчик, датчик опору шкіри, датчик провідності шкіри та акселерометр/датчик орієнтації [19, с. 3]. Для IoT-системи це показує типову структуру локального бездротового рішення: датчики – мікроконтролер – шлюз – Wi-Fi/локальна мережа – сервер.

Передавання даних у цій роботі перевірялося не в ідеальних умовах, а через імітацію слабкого бездротового каналу. Для цього використовувалися NetEM для встановлення втрат пакетів і затримки, а також TBF для обмеження пропускної здатності [19, с. 3]. Кожне випробування тривало 10 хвилин і повторювалося 3 рази; результат подавався як середнє значення. Один пакет мав формат JSON і містив 409 байтів даних про частоту серцевих скорочень, рівень кисню в крові, вологість шкіри та показники акселерометра. Щосекунди передавалося 4 пакети, тобто за 10 хвилин обсяг прикладних даних становив 981 600 байтів [19, с. 3]. Ці параметри можна використати як приклад для побудови власної моделі передавання сенсорних повідомлень.

У межах експерименту порівнювалися CoAP, MQTT, DDS і власний UDP-протокол. MQTT працює за моделлю publish-subscribe: пристрій публікує дані в певну тему, а підписник отримує повідомлення через брокер. Для реалізації MQTT використовувалися сервер HiveMQ і клієнти Mosquitto [19, с. 2–3]. Така модель зручна для IoT-систем, де кілька датчиків передають дані на сервер або веб-панель, а користувач отримує лише ті параметри, на які підписаний.

Таблиця 1.4

Поведінка IoT-протоколів у бездротовій мережі з обмеженою якістю зв'язку

Умова або показник	Результат дослідження	Значення для IoT-системи
Втрати пакетів у мережі	Тестувалися умови від 0 % до 25 % втрат пакетів [19, с. 4]	Дозволяє оцінити, як протокол працює при нестабільному каналі
Затримка в мережі	Імітувалася затримка від 0 до 400 мс [19, с. 4]	Важливо для систем моніторингу, де дані мають надходити вчасно

Продовження таблиці 1.4

MQTT DDS	Не мали фактичних втрат пакетів навіть за 25 % втрат у мережі та 400 мс затримки [19, с. 6]	ТСР-механізми повторної передачі підвищують надійність доставки
MQTT	При втраті пакетів від 5 % до 25 % затримка могла перевищувати 1000 мс [19, с. 5]	Надійність зберігається, але час доставки може зростати
DDS	У складних умовах мав нижчу затримку, ніж MQTT, але споживав більше пропускну здатності [19, с. 5–6]	Підходить для систем, де критична швидкість доставки
CoAP UDP	Мали менше службового навантаження, але фактичні втрати пакетів наближалися до рівня втрат у мережі [19, с. 5–6]	Підходять не для всіх систем, бо доставка менш гарантована

Окремо автори змоделювали слабку бездротову мережу з такими параметрами: 10 % втрат пакетів, затримка $100 \text{ мс} \pm 20 \text{ мс}$ і обмеження пропускну здатності 6,25 кБ/с [19, с. 6]. У таких умовах DDS створював приблизно 289 байтів службових даних на кожні 100 байтів користувацьких даних, а MQTT – близько 76 байтів. При цьому середня затримка DDS становила 234 мс, тоді як MQTT мав 578 мс [19, с. 6]. Це показує важливий компроміс: MQTT має менше службове навантаження, але в поганому каналі може давати більшу затримку; DDS швидше доставляє дані, але потребує більше пропускну здатності.

Для Wi-Fi та локальних бездротових мереж цей результат має практичне значення. Сам по собі канал зв'язку не гарантує стабільності IoT-системи, тому вибір прикладного протоколу впливає на втрати, затримку та службове навантаження. Якщо система передає некритичні дані, наприклад температуру або вологість раз на кілька секунд, допустима більша затримка. Якщо повідомлення пов'язане з аварійним станом або швидкою реакцією, затримка й повторні передачі стають критичними параметрами.

К. Mekki, Е. Bajic, F. Chaxel та F. Meyer порівнювали LPWAN-технології для великомасштабного IoT, але їхня робота також допомагає зрозуміти межі використання мереж ближнього радіусу дії. Автори пояснюють, що IoT-застосунки часто потребують великої дальності, низької

швидкості передавання, малого енергоспоживання та низької вартості. Короткодистанційні технології, зокрема ZigBee і Bluetooth, не підходять для сценаріїв, де потрібне передавання на значну відстань, а стільникові рішення 2G, 3G і 4G забезпечують ширше покриття, але споживають більше енергії пристрою [27, с. 1].

У цьому порівнянні LPWAN подано як рішення для тих випадків, де короткодистанційної мережі вже недостатньо. Дальність LPWAN може становити 10–40 км у сільській місцевості та 1–5 км у міських умовах, а строк роботи батареї може перевищувати 10 років [27, с. 1–2]. Для міської IoT-системи це означає, що Wi-Fi зручний для локального вузла, лабораторного стенда або приміщення, але для розміщених по місту автономних датчиків часто потрібні технології більшої дальності.

Таблиця 1.5

Порівняння локальних бездротових мереж і LPWAN-рішень для IoT

Тип технології	Сильні сторони	Обмеження
Wi-Fi	Висока швидкість, просте підключення до локальної мережі, зручність для веб-інтерфейсу й серверної частини	Залежність від покриття роутера, перевантаження каналу, помітний вплив втрат пакетів і затримки [19, с. 4–6]
ZigBee / Bluetooth	Підходять для коротких відстаней і локальних сенсорних мереж	Не пристосовані для сценаріїв із великою дальністю передавання [27, с. 1]
LoRaWAN	Дальність до 5 км у місті та до 20 км у сільській місцевості, корисне навантаження до 243 байтів, AES-128, висока стійкість до завад [27, с. 3]	Нижча швидкість передавання порівняно з Wi-Fi
NB-IoT	До 200 кбіт/с, корисне навантаження до 1600 байтів, підтримка великої кількості пристроїв	Працює в ліцензованих LTE-частотах і залежить від операторської інфраструктури [27, с. 3; 5]

За технічними параметрами LoRaWAN працює в неліцензованих ISM-діапазонах 868 МГц у Європі, 915 МГц у Північній Америці та 433 МГц в Азії. Для LoRaWAN подано ширину каналу 125 або 250 кГц, максимальну швидкість до 50 кбіт/с, максимальну довжину повідомлення 243 байти, дальність 5 км у місті та 20 км у сільській місцевості, а також підтримку

AES-128 [27, с. 3]. Ці дані показують різницю між локальною мережею та мережами далекого радіусу: Wi-Fi дає більшу швидкість у межах локального покриття, тоді як LPWAN краще підходить для рідкісного передавання невеликих сенсорних повідомлень на великій відстані.

Для теми завадостійкої IoT-системи важливо враховувати не лише сам тип мережі, а й характер повідомлень. Якщо сенсорний вузол працює у приміщенні або на невеликій відстані від точки доступу, Wi-Fi може бути достатнім для передавання даних на сервер або веб-панель. Якщо вузли розміщуються в різних частинах міської забудови, де немає стабільного локального покриття, потрібні технології з більшою дальністю, нижчим енергоспоживанням і кращою стійкістю до перешкод. Саме тому Wi-Fi у такій системі краще розглядати як локальний канал, а LoRaWAN або інші LPWAN-рішення – як варіант для розширення мережі за межі короткого радіусу.

Отже, Wi-Fi і мережі ближнього радіусу дії зручні для локальних IoT-систем, де сенсорні вузли працюють у межах одного приміщення, будівлі або невеликої зони. Вони дають змогу швидко передавати дані на сервер, використовувати MQTT та інші прикладні протоколи, підключати веб-інтерфейс і виконувати тестування системи. Разом із цим експерименти Y. Chen та T. Kunz показують, що при втраті пакетів, затримці й обмеженій пропускній здатності поведінка протоколів суттєво відрізняється [19, с. 4–6]. Порівняння K. Mekki, E. Bajic, F. Chahel та F. Meyer доповнює цей висновок: короткодистанційні технології мають свої межі, а для міського IoT з рознесеними сенсорними вузлами часто потрібні рішення більшої дальності й нижчого енергоспоживання [27, с. 1–3].

Технологія LoRa/LoRaWAN належить до групи LPWAN-рішень, тобто малопотужних глобальних мереж, які використовуються для передавання невеликих обсягів даних на значні відстані. Її головна особливість полягає не у високій швидкості, а в поєднанні дальності зв'язку, низького енергоспоживання та можливості роботи великої кількості сенсорних вузлів.

О. О. Борківська, Б. Ю. Жураковський та А. В. Платоненко пояснюють, що LPWAN-технології з'явилися через потребу підключати велику кількість вимірювальних і телеметричних пристроїв до хмарних серверів для централізованого збирання даних [3, с. 115].

LoRa побудована на модуляції Chirp Spread Spectrum (CSS), за якої дані кодуються широкосмуговими імпульсами з частотами, що з часом зростають або зменшуються. Такий підхід підвищує стійкість приймача до відхилень частоти та спрощує вимоги до тактового генератора, що дає змогу використовувати дешевші компоненти [3, с. 115]. Технологія працює в субгігерцових діапазонах: 169, 433, 868 і 915 МГц; для Європи найчастіше використовується 868 МГц, для США – 915 МГц [3, с. 116]. Передавання також підтримує пряму корекцію помилок FEC, що важливо для роботи в умовах завад [3, с. 116].

A. Augustin, J. Yi, T. Clausen та W. M. Townsley розглядають LoRa як довгодистанційну, малопотужну й низькошвидкісну телекомунікаційну систему для IoT. Кінцеві пристрої передають повідомлення до шлюзів одним радіострибком, а шлюзи вже пересилають дані до центрального мережевого сервера [15, с. 1]. Така схема добре підходить для сенсорних систем, де потрібно передавати короткі повідомлення з датчиків, але немає потреби в постійному високошвидкісному каналі.

Таблиця 1.6

Технічні характеристики LoRa/LoRaWAN для IoT-систем

Показник	Значення / характеристика	Практичне значення
Тип мережі	LPWAN, малопотужна мережа великої дальності	Підходить для автономних сенсорних вузлів
Фізичний рівень	LoRa, модуляція CSS	Підвищує стійкість до шуму, завмирання сигналу та інтерференції [26, с. 4]
Мережевий протокол	LoRaWAN	Організовує взаємодію кінцевих пристроїв, шлюзів і серверів
Швидкість передавання	0,3–50 кбіт/с за даними української статті; 0,3–37,5 кбіт/с за оглядовими працями [3, с. 116; 18, с. 66; 32, с. 373]	Достатньо для коротких сенсорних повідомлень, але не для великих потоків даних

Продовження таблиці 1.6

Дальність	до 2,5 км у місті та до 45 км поза містом за порівняльною таблицею; в інших роботах – 5–15 км залежно від умов [3, с. 116; 32, с. 373]	Дає змогу покривати значні території без щільної мережі точок доступу
Час автономної роботи	понад 10 років або 5–10 років залежно від режиму роботи й навантаження [3, с. 116; 32, с. 376]	Важливо для датчиків, які складно часто обслуговувати
Корисне навантаження	до 243–250 байтів у різних описах [27, с. 3; 32, с. 373]	Орієнтація на невеликі повідомлення
Безпека	AES-128, автентифікація, MAC-контроль цілісності [17, с. 2; 32, с. 373]	Захист даних під час передавання
Топологія	«зірка» / star-of-stars	Кінцеві вузли передають дані до шлюзів, а далі – на сервер [3, с. 117; 26, с. 3]

LoRaWAN має виражену орієнтацію на сенсорні системи. V. Bonilla, B. Campoverde та S. G. Yoo пояснюють, що ця технологія розрахована на пристрої, які обмінюються інформацією із сервером на низькій швидкості та з великими інтервалами між передачами – іноді раз на годину або навіть раз на кілька днів [17, с. 2]. Це добре відповідає системам міського моніторингу, де датчик температури, вологості, газу, руху або рівня освітленості не передає постійний потік, а надсилає короткі повідомлення через певний проміжок часу або після події.

Архітектура LoRaWAN складається з трьох основних елементів: кінцевих пристроїв, шлюзів і серверної частини. Серверна частина охоплює мережевий сервер, join-сервер і сервер застосунків [17, с. 2]. С. Li та Z. Cao подають подібну схему: розподілені кінцеві вузли генерують сенсорні дані, шлюзи пересилають пакети, мережевий сервер усуває дублікати, виконує перевірки безпеки, планує підтвердження, а потім передає інформацію до серверів застосунків [26, с. 3]. Для дипломної роботи це важливо, бо LoRaWAN не завершується на радіоканалі: після передавання сигналу дані проходять через мережеву й серверну обробку.

F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melià-Seguí та T. Watteyne звертають увагу на межі застосування LoRaWAN. Максимальна «сира» швидкість LoRaWAN становить 27 кбіт/с, а при використанні FSK –

до 50 кбіт/с [14, с. 34]. Через це технологія не підходить для великих потоків даних. Наприклад, для IP-відеоспостереження потрібна швидкість від 130 кбіт/с для низькоякісного MJPEG до 4 Мбіт/с для відео 1920×1080 при 30 fps, тому LoRaWAN не може використовуватися для таких задач [14, с. 38]. Натомість вона добре працює там, де потрібні невеликі періодичні або події повідомлення.

Сфери застосування LoRaWAN найкраще розкриваються через Smart City. О. О. Борківська, Б. Ю. Жураковський та А. В. Платоненко відносять до таких рішень інтелектуальне міське середовище, відеоспостереження, датчики освітленості, Wi-Fi-точки, інтелектуальний транспорт, розумну парковку, системи каршерінгу, розумні будинки, датчики температури й вологості, датчики відкриття дверей, розумні лічильники води, газу й електроенергії, а також датчики виявлення витоків [3, с. 117]. Для таких завдань LoRaWAN підходить краще, ніж класичний Wi-Fi, коли сенсорні вузли розміщуються на значній відстані або працюють від батареї.

Окрему роль відіграє енергоспоживання. У LoRaWAN передбачено три класи кінцевих пристроїв: Class A, Class B і Class C. Class A має найвищу енергоефективність, але найбільшу затримку для downlink-повідомлень; Class B відкриває додаткові приймальні вікна у визначений час; Class C тримає приймальні вікна відкритими постійно, тому споживає найбільше енергії [17, с. 2]. У системах моніторингу найчастіше важливий саме Class A, бо датчик більшу частину часу може перебувати в режимі сну, прокидатися для передавання пакета і знову знижувати енергоспоживання.

Для фізичного рівня LoRa ключовими параметрами є Spreading Factor (SF), ширина смуги Bandwidth (BW), потужність передавання та частота каналу. SF може змінюватися від 7 до 12; вищий SF збільшує дальність і стійкість до шуму, але збільшує час перебування пакета в ефірі й зменшує швидкість передавання. BW може становити 125, 250 або 500 кГц; менша смуга зменшує швидкість, але підвищує стійкість до шуму [26, с. 4]. Саме

тому LoRa/LoRaWAN добре підходить для завадостійкого передавання коротких повідомлень, але має обмеження для частих або великих передач.

Порівняння з іншими LPWAN-рішеннями показує, що LoRa займає проміжну позицію між низькою швидкістю і широким покриттям. М. Centenaro, L. Vangelista, A. Zanella та M. Zorzi порівнюють SIGFOX, Ingenu і LoRa: для LoRa наведено швидкість 0,3–37,5 кбіт/с, роботу в різних субгігагерцових ISM-діапазонах, двосторонній зв'язок і приблизно 10^4 вузлів на одну базову станцію [18, с. 66]. У міському тесті в Падуї LoRa забезпечила покриття близько 2 км у складних умовах поширення сигналу, але з урахуванням запасу до завод і змін бюджету лінії автори взяли номінальне значення 1,2 км [18, с. 65–66]. Для міської забудови це показує реальніший підхід: паспортна дальність може бути великою, але фактичне планування має враховувати будівлі, завади й запас лінії зв'язку.

У цьому ж дослідженні подано приклад планування покриття міста Падуя площею близько 100 км². За консервативною оцінкою, для покриття всієї території потрібно близько 30 шлюзів, що менше половини кількості майданчиків одного з великих мобільних операторів у цьому місті [18, с. 66]. Для міських IoT-систем це важливий приклад: LoRa/LoRaWAN може зменшити кількість інфраструктурних точок порівняно зі стільниковими мережами, якщо передаються невеликі сенсорні повідомлення.

Таблиця 1.7

Придатність LoRa/LoRaWAN для різних IoT-сценаріїв

Сценарій	Придатність LoRa/LoRaWAN	Пояснення
Температура, вологість, CO ₂ , освітленість	Висока	Невеликі пакети, передавання з інтервалами, низьке енергоспоживання [17, с. 2; 17, с. 6]
Розумні лічильники	Висока	Потрібна дальність і автономність, а обсяг даних невеликий [3, с. 117; 14, с. 38]
Smart parking, smart lighting, waste collection	Висока	Дані передаються подієво або періодично, затримка не завжди критична [14, с. 38]
Інтелектуальний транспорт з критичною затримкою	Обмежена	LoRaWAN не гарантує детерміновану затримку через ALOHA-доступ [14, с. 38–39]

Продовження таблиці 1.7

Відеоспостереження	Низька	Швидкість LoRaWAN 0,3–50 кбіт/с не відповідає вимогам відеопотоку [14, с. 38]
Передавання зображень або мультимедіа	Обмежена	LoRa ефективна для скалярних даних, але мультимедіа обмежується швидкістю, payload і duty-cycle [21, с. 1]

Переваги LoRaWAN не скасовують її обмежень. Технологія використовує ALOHA-доступ, тобто кінцеві пристрої можуть передавати пакет тоді, коли він готовий, без повної синхронізації з іншими вузлами. Через це виникають колізії, особливо у великих мережах. А. Farhad, D.-Н. Kim та J.-Y. Pyun пояснюють, що в LoRaWAN пакетні колізії можуть виникати через intra-SF та inter-SF інтерференцію, що підвищує рівень втрат пакетів [23, с. 1]. У підтвердженому режимі втрати змушують пристрої повторювати передачі й підвищувати SF після невдалих спроб, але більший SF збільшує час перебування пакета в ефірі та може ще сильніше підвищити ймовірність колізій [23, с. 2].

Для підвищення успішності доставки застосовуються різні схеми розподілу SF. У статті А. Farhad, D.-Н. Kim та J.-Y. Pyun розглянуто channel-adaptive SF recovery і розподіл SF на основі чутливості кінцевого пристрою. Автори показують, що адаптація SF за станом каналу може підвищити packet success ratio порівняно з типовими схемами розподілу SF [23, с. 1–2]. Також у наведеному ними огляді згадуються результати інших підходів: collision-aware SF assignment може зменшити packet error rate до 42 %, оптимізація кількості пристроїв для певного SF може збільшити мережеву місткість до 700 %, а EXplora-C дає покращення data extraction rate до 38 % порівняно з legacy ADR [23, с. 3–4].

С. Li та Z. Cao розглядають LoRa-мережі через стек рівнів: PHY, link, MAC і application. На фізичному рівні працює CSS-модуляція, на link-рівні налаштовуються SF, BW, потужність і частота каналу, на MAC-рівні вирішуються питання доступу до середовища й колізій, а прикладний рівень забезпечує доставку даних відповідно до задачі застосунку [26, с. 3–4]. Такий

поділ важливий для моделювання завадостійкої системи: втрати пакетів можуть бути спричинені не тільки слабким сигналом, а й неправильним вибором SF, перевантаженням каналу, ALOHA-колізіями або невідповідністю протоколу характеру трафіку.

J. P. S. Sundaram, W. Du та Z. Zhao також виділяють LoRa як технологію для автономних LPWAN-мереж без сторонньої інфраструктури. За їхнім порівнянням, LoRa працює в ISM sub-GHz діапазонах, використовує CSS/FSK, має швидкість 0,3–37,5 кбіт/с, дальність 5–15 км, payload до 250 байтів, симетричну автентифікацію та AES-128 [32, с. 372–373]. Вони також зазначають, що IoT-вузли часто мають працювати 5–10 років без обслуговування, тому енергоспоживання є одним із головних параметрів LoRa-мереж [32, с. 371; 376].

Практичні приклади, наведені J. P. S. Sundaram, W. Du та Z. Zhao, показують економічний ефект від використання LoRa. Для системи управління відходами в Сеулі згадується зменшення частоти збору сміття на 66 %, скорочення витрат на 83 % і збільшення рівня переробки на 46 %. Для моніторингу сонячних електростанцій у Карсон-Сіті наведено скорочення операційних витрат на 15 % і збільшення вироблення чистої енергії до 75 000 кВт·год. У Ліоні та Греноблі моніторинг енергоспоживання допоміг зменшити споживання електроенергії на 16 % [32, с. 375]. Такі приклади показують, що LoRaWAN має практичну цінність не лише як технологія зв'язку, а і як інфраструктура для міських сервісів.

Огляд V. Bonilla, B. Campoverde та S. G. Yoo підтверджує, що LoRaWAN активно використовується в екологічному моніторингу, сільському господарстві, медицині, мобільності, smart buildings і smart city. Найбільше робіт у їхньому аналізі стосувалися environmental monitoring, а серед апаратних платформ часто використовувалися Semtech, Arduino та Raspberry Pi: Semtech згадується у 15 проєктах, Arduino – у 12, Raspberry Pi – в 11 [17, с. 10–12].

Таким чином, LoRa/LoRaWAN є ефективним рішенням для далекого та енергоефективного передавання невеликих сенсорних повідомлень у IoT-системах. Її переваги пов'язані з роботою в субгігагерцових діапазонах, CSS-модуляцією, низьким енергоспоживанням, великою дальністю, можливістю обслуговування значної кількості вузлів і підтримкою захищеного обміну даними. Обмеження проявляються у низькій швидкості, невеликому payload, duty-cycle, ALOHA-колізіях, зростанні затримки та неможливості передавати великі мультимедійні потоки. Для завадостійкої системи в умовах міської забудови LoRaWAN підходить як канал для періодичного або подієвого передавання даних від автономних сенсорів, але потребує правильного вибору SF, BW, кількості шлюзів, режиму підтвердження та алгоритмів повторної передачі.

MQTT належить до прикладних протоколів обміну повідомленнями, які використовуються для зв'язку між IoT-пристроями, сервером і програмним середовищем моніторингу. Його основна роль полягає не у фізичному передаванні сигналу, як у Wi-Fi або LoRa, а в організації логіки обміну даними: який пристрій передає повідомлення, у яку тему, хто його отримує і з яким рівнем надійності. У специфікації OASIS MQTT визначено як клієнт-серверний протокол передавання повідомлень за моделлю publish/subscribe. Він є легким, відкритим, простим для реалізації та придатним для M2M- і IoT-середовищ, де важливі малий обсяг програмного коду та економне використання пропускну здатності [29, с. 1].

MQTT працює поверх TCP/IP або інших мережевих протоколів, які забезпечують упорядковане, безвтратне та двостороннє з'єднання [29, с. 1]. Це відрізняє його від протоколів, які працюють поверх UDP і самостійно мають реалізовувати частину механізмів надійності. Для IoT-системи така особливість важлива: сенсорний вузол може передавати дані брокеру, а TCP-рівень уже забезпечує контроль доставки, повторну передачу втрачених сегментів і збереження порядку повідомлень.

Структура MQTT базується на трьох основних ролях: publisher, subscriber і broker. Publisher публікує повідомлення в певну тему, subscriber підписується на потрібні теми, а broker приймає повідомлення та пересилає їх тим клієнтам, чий підписки відповідають темі повідомлення [31, с. 4]. У практичній IoT-системі publisher може бути вузол ESP32 або Arduino з датчиком температури, broker – сервер Mosquitto або інший MQTT-брокер, а subscriber – веб-панель, база даних або застосунок моніторингу.

Таблиця 1.8

Основні елементи MQTT-обміну в IoT-системі

Елемент	Функція в MQTT	Приклад у системі моніторингу
Publisher	Публікує повідомлення в тему	ESP32 надсилає температуру або рівень сигналу
Broker	Приймає повідомлення й пересилає його підписникам	Mosquitto або інший сервер MQTT
Subscriber	Отримує повідомлення з потрібної теми	Веб-інтерфейс або база даних
Topic	Логічний канал обміну даними	sensors/temperature, channel/rssi, alerts/error
Payload	Корисне навантаження повідомлення	Значення температури, вологості, RSSI або стану вузла

Поняття topic має центральне значення для MQTT. У специфікації вказано, що Topic Name визначає інформаційний канал, до якого публікуються дані payload; він має бути присутнім у PUBLISH-пакеті як UTF-8-рядок [29, с. 55–56]. Для IoT-системи це зручно, тому що різні типи сенсорних даних можна рознести за окремими темами. Наприклад, температура передається в одну тему, рівень сигналу – в іншу, а повідомлення про помилку або втрату зв'язку – в окремий канал.

MQTT-повідомлення передаються через MQTT Control Packets. Їхня структура включає фіксований заголовок, змінний заголовок і payload; фіксований заголовок є обов'язковим для всіх MQTT Control Packets, а змінний заголовок і payload використовуються залежно від типу пакета [31, с. 4]. До основних пакетів належать CONNECT, CONNACK, PUBLISH, SUBSCRIBE, SUBACK, UNSUBSCRIBE, PINGREQ, PINGRESP і DISCONNECT [31, с. 4]. Для сенсорної системи найбільш показовим є

PUBLISH, оскільки саме через нього передаються прикладні дані від пристрою до брокера і далі до підписників.

Надійність доставки в MQTT задається через три рівні QoS. У специфікації MQTT подано такі рівні: QoS 0 – “at most once”, QoS 1 – “at least once”, QoS 2 – “exactly once” [29, с. 54]. У практичному сенсі QoS 0 означає, що повідомлення надсилається один раз без підтвердження; QoS 1 гарантує доставку щонайменше один раз, але допускає дублікати; QoS 2 забезпечує доставку рівно один раз, однак має найбільше службове навантаження [29, с. 94–95].

Для IoT-системи вибір QoS залежить від значення повідомлення. Дані температури або вологості, які передаються кожні кілька секунд, можуть використовувати QoS 0, бо втрата одного вимірювання не завжди критична. Аварійні повідомлення або дані про збій каналу краще передавати з QoS 1, щоб підвищити ймовірність доставки. QoS 2 має сенс там, де неприпустимі і втрата, і дублювання повідомлення, але для малопотужних вузлів його використання може створювати зайве навантаження.

Таблиця 1.9

Рівні QoS у MQTT та їх значення для IoT

Рівень QoS	Принцип доставки	Особливість для IoT-системи
QoS 0	Повідомлення доставляється максимум один раз	Мінімальне навантаження, але можлива втрата повідомлення [29, с. 94]
QoS 1	Повідомлення доставляється щонайменше один раз	Краща надійність, але можливі дублікати [29, с. 94]
QoS 2	Повідомлення доставляється рівно один раз	Найвища надійність, але найбільший службовий обмін [29, с. 95]

V. Seoane, C. García-Rubio, F. Almenares та C. Campo також визначають MQTT як легкий publish/subscribe-протокол, придатний для IoT, і підкреслюють його орієнтацію на надійність завдяки роботі поверх TCP [31, с. 4]. До переваг MQTT вони відносять низьке службове навантаження, компактний заголовок, надійність, гнучкість QoS, а також flow control і congestion control, які забезпечуються TCP [31, с. 5]. Ці властивості

пояснюють, чому MQTT часто використовується для передавання сенсорних даних на сервер або у веб-інтерфейс.

Разом із перевагами MQTT має обмеження. Безпека не вбудована в сам протокол як окремий прикладний механізм, тому для захищеного зв'язку MQTT зазвичай переноситься через TLS між MQTT і TCP [31, с. 5]. Інша проблема пов'язана з брокером: якщо broker стає недоступним, обмін між publisher і subscriber фактично припиняється, бо саме він виконує роль центрального посередника [31, с. 5]. Для завадостійкої системи це означає, що потрібно контролювати не тільки канал між датчиком і мережею, а й доступність брокера.

Експериментальна частина роботи V. Seoane та інших авторів дає корисні числові дані для оцінювання MQTT. Для тестування використовувалися Raspberry Pi 3 + Rainbow HAT як серверний сенсорний вузол, віртуальні машини Linux для клієнта, брокера й емулятора мережі, також NetEm для моделювання втрат пакетів, затримок, дублювання, перестановки та пошкодження пакетів [31, с. 8]. Для MQTT було проведено 45 експериментів: три рівні QoS, три режими безпеки – NOSEC, PSK і PKI – та п'ять рівнів втрат пакетів: 0 %, 5 %, 10 %, 15 % і 20 %. У кожному експерименті publisher публікував 500 MQTT-повідомлень, а трафік контролювався через Wireshark [31, с. 10].

Таблиця 1.10

Результати оцінювання MQTT за V. Seoane та іншими авторами

Показник	Результат	Значення для IoT-системи
Кількість MQTT-експериментів	45 експериментів: QoS 0/1/2 × NOSEC/PSK/PKI × 0–20 % втрат [31, с. 10]	Дає змогу оцінити MQTT у різних умовах мережі
Кількість повідомлень	500 MQTT-повідомлень у кожному експерименті [31, с. 10]	Можна використовувати як приклад тестування власної системи
Службове навантаження	MQTT потребує більше пропускної здатності через TCP-overhead [31, с. 18–20]	Надійність досягається ціною більшого трафіку

Продовження таблиці 1.10

Порівняння з CoAP	У незахищеній мережі без втрат CoAP scenario B використовував 226 байтів, MQTT з найвищим QoS – 1560 байтів [31, с. 18]	MQTT менш економний за трафіком
Кількість пакетів	CoAP scenario B – 4 пакети, MQTT з найвищим QoS – 21 пакет [31, с. 18]	Вищий QoS у MQTT збільшує кількість службових обмінів
Захист TLS	TLS для MQTT збільшує bandwidth usage на 74–понад 200 % [31, с. 20]	Захист підвищує навантаження на канал
CPU usage	Для MQTT приріст CPU usage становив близько 27 % для PSK і 36 % для PKI [31, с. 20]	Криптографія помітно навантажує вузол
Затримка при втратах	TCP-ретрансляції MQTT можуть давати затримки в кілька секунд [31, с. 19]	У нестабільному каналі доставка зберігається, але час може зрости

Ці результати показують головний компроміс MQTT. Протокол добре підходить для IoT-систем, де потрібна надійна доставка повідомлень, структурований обмін через теми й просте підключення веб-інтерфейсу або бази даних. Водночас TCP, QoS 1/2 і TLS збільшують службове навантаження, кількість пакетів, використання CPU і затримку при втратах. Тому в системі міського моніторингу MQTT зручно використовувати на ділянці між шлюзом, сервером і веб-інтерфейсом, а на рівні далекого радіоканалу поєднувати його з Wi-Fi або LoRa залежно від умов передавання.

Для завадостійкої IoT-системи MQTT має практичне значення як протокол прикладного рівня, який забезпечує впорядкований обмін повідомленнями між вузлами. Сенсорний пристрій може публікувати значення температури, вологості, RSSI, SNR, кількість повторних передач або CRC-помилки в окремі теми, а веб-панель може підписуватися на ці теми й відображати стан системи в реальному часі. У такій архітектурі MQTT не усуває фізичні завади, але допомагає організувати контроль доставки, логіку повідомлень, поділ даних за темами та обмін між апаратною і програмною частинами системи.

1.3. Вплив міської забудови та завад на якість передавання даних в IoT-мережах

Поширення радіосигналу в умовах міської забудови має складніший характер, ніж у відкритому просторі, оскільки між передавачем і приймачем часто немає прямої видимості. На якість зв'язку впливають щільність будівель, висота забудови, матеріали стін, рельєф місцевості, відбиття від поверхонь, дифракція на краях будівель, поглинання сигналу та наявність інших джерел радіозавад. Для IoT-систем це означає, що фактична дальність передавання може суттєво відрізнятися від теоретичних або паспортних значень, а стабільність зв'язку потрібно оцінювати не тільки за відстанню, а й за умовами середовища.

J. A. Azevedo та F. Mendonça пояснюють, що для розгортання LoRa-мереж важливими є моделі поширення сигналу, оскільки саме вони дозволяють оцінити втрати на трасі, рівень прийнятого сигналу та очікувану якість покриття. Автори поділяють моделі на емпіричні, детерміновані та моделі активних вимірювань. Більшість досліджень LoRa використовують емпіричні моделі, бо детерміновані моделі потребують великого обсягу даних про середовище: рельєф, будівлі, перешкоди та інші просторові параметри [16, с. 1–2]. Для міської забудови це важливо, оскільки універсальна модель не завжди точно описує реальні втрати сигналу в конкретному районі.

У радіоканалі виділяють великомасштабні та маломасштабні втрати. Великомасштабні втрати пов'язані із середнім ослабленням сигналу на значних відстанях і наявністю перешкод між передавачем та приймачем. Маломасштабні зміни виникають через різке коливання рівня сигналу навіть при невеликому переміщенні приймача. Для такого опису використовують розподіли Rayleigh, Rician, log-normal або Nakagami [16, с. 1]. У міських умовах ці явища проявляються особливо помітно: сигнал може одночасно

відбиватися від будівель, частково блокуватися стінами та доходити до приймача кількома шляхами.

Для опису втрат сигналу в різних умовах застосовують моделі Okumura–Hata, COST 231-Hata, Lee, Walfisch–Ikegami, ITU-R та інші. Наприклад, модель Okumura–Hata використовується для частот від 200 до 1500 МГц, відстаней від 1 до 20 км, висоти антени базової станції від 30 до 200 м і висоти антени термінала від 1 до 20 м [16, с. 4]. Це пояснює її часте використання для аналізу LoRaWAN, оскільки LoRa в Європі працює в діапазоні 868 МГц, а сама мережа орієнтована на передавання невеликих повідомлень на значні відстані.

Таблиця 1.11

Чинники, які впливають на поширення радіосигналу в міській забудові

Чинник	Як впливає на сигнал	Значення для IoT-системи
Щільність будівель	створює зони затінення та NLOS-умови	зменшує дальність і стабільність зв'язку
Висота будівель	може перекривати пряму видимість між вузлом і шлюзом	впливає на вибір місця розміщення антени
Відбиття сигналу	формує багатопроменеве поширення	може спричиняти коливання RSSI та SNR
Дифракція на краях будівель	частково дозволяє сигналу огинати перешкоди	підтримує зв'язок без прямої видимості, але з додатковими втратами
Відстань між вузлами	збільшує path loss	потребує правильного вибору SF, потужності та шлюзів
Перешкоди й завади	погіршують якість приймання	підвищують ризик втрат пакетів

R. El Chall, S. Lahoud та M. El Helou досліджували LoRaWAN-канал у діапазоні 868 МГц у різних умовах Лівану: у приміщеннях, міському середовищі Бейрута та сільській місцевості долини Бекаа. За результатами вимірювань автори отримали дальність покриття до 8 км у міському середовищі та до 45 км у сільському середовищі [22, с. 1]. Така різниця показує прямий вплив забудови: одна й та сама технологія може мати значно меншу дальність у місті через щільність будівель, NLOS-умови та складний профіль місцевості.

У цьому дослідженні також зазначено, що реальні LoRa/LoRaWAN-випробування в різних роботах показували дальність 2–8 км у міських умовах і 10–30 км у сільській місцевості [22, с. 1]. Крім того, рослинність і підвищена температура можуть зменшувати дальність зв'язку, а в приміщеннях можливі проблеми із покриттям у підвальних рівнях [22, с. 1]. Для міської IoT-системи це означає, що оцінювання радіоканалу не можна обмежувати лише розрахунком за відстанню: потрібно враховувати тип забудови, висоту антен, наявність перешкод і місце встановлення шлюзу.

Особливу увагу автори приділили NLOS-умовам. У міському середовищі Бейрута були виділені помірно та сильно екрановані ділянки: у першому випадку прямий шлях частково перекривають дерева або краї будівель, у другому – сигнал значно блокується щільною забудовою. Для міського сценарію стандартне відхилення затінення становило близько 7,2 дБ, а моделі Okumura–Hata та COST 231-Hata у певних випадках прогнозували більші втрати, ніж було отримано за вимірюваннями [22, с. 9]. Це підтверджує, що для міського середовища потрібна перевірка моделей реальними вимірюваннями.

W. Ingabire, H. Larijani та R. M. Gibson виконали оцінювання моделей поширення LoRaWAN у міському середовищі Глазго. Для моделювання використано програму ATDI ICS Telecom, а результати моделей Okumura–Hata, COST-231 Hata, Extended Hata та ITU-R 1225 порівнювалися з реальними вимірюваннями в місті [25, с. 1]. У роботі підкреслено, що для точного моделювання важливі цифрова модель рельєфу, дані про забудову, карти, векторні шари та clutter layer, тобто інформація про типи поверхонь і перешкод [25, с. 1; с. 4].

Параметри моделювання були такими: частота 868 МГц, ширина смуги 125 кГц, spreading factor SF12, висота антени шлюзу 50 м, максимальна відстань між передавачем і приймачем 2275 м, потужність передавання кінцевого пристрою 14 dBm, висота антени кінцевого пристрою 1,5 м [25, с.

4]. Аналіз проводився для NLOS-умов, тобто без прямої видимості, що добре відповідає ситуації щільної міської забудови.

Таблиця 1.12

Параметри міського моделювання LoRaWAN у Глазго

Параметр	Значення
Частота роботи	868 МГц
Ширина смуги	125 кГц
Spreading Factor	SF12
Висота антени шлюзу	50 м
Максимальна відстань Tx–Rx	2275 м
Потужність передавання вузла	14 dBm
Висота антени кінцевого пристрою	1,5 м

Результати показали, що втрати пакетів у деяких точках можуть пояснюватися високою щільністю та наявністю кількох високих будівель у Глазго [25, с. 4]. Серед чотирьох моделей найкраще до реальних вимірювань наблизилася ITU-R 1225: середня абсолютна похибка становила 1,13 dBm, а стандартне відхилення – 5,49 dBm. Найменш точною була Extended Hata, для якої MAE становила 1,82 dBm, а стандартне відхилення – 9,19 dBm [25, с. 4–5].

Таблиця 1.13

Точність моделей поширення LoRaWAN у міському середовищі Глазго

Модель	AE	MAE	STD	Оцінка результату
Okumura–Hata	–0,91	1,58	8,81	занижувала рівень прийнятого сигналу
COST-231 Hata	–0,84	1,56	8,56	близька до Okumura–Hata
Extended Hata	0,58	1,82	9,19	найменш точна серед розглянутих
ITU-R 1225	0,29	1,13	5,49	найточніше наближення до вимірювань

Результати Глазго показують, що для міської IoT-системи модель поширення потрібно підбирати під конкретне середовище. Okumura–Hata та COST-231 Hata занижували рівень прийнятого сигналу, тоді як ITU-R 1225 і Extended Hata його завищували [25, с. 5]. Найменша похибка ITU-R 1225 означає, що ця модель краще описувала конкретні умови вимірювання, але це не робить її універсальною для будь-якого міста. У практичній системі результат залежить від щільності забудови, висоти шлюзу, маршруту вимірювань, частоти, SF і розміщення кінцевого пристрою.

R. Mena, M. Ramos, L. Urquiza та J. D. Vega-Sánchez перевіряли роботу LoRaWAN у міських і сільських умовах Кіто. У міському сценарії шлюз був

установлений на терасі будівлі в районі La Florida, де наявна щільна забудова та більша кількість перешкод. Для порівняння в сільському сценарії шлюз розміщували на платформі висотою 3 м у зоні Cocotog із низькою щільністю забудови та більш рівним рельєфом [28, с. 3]. У сільському сценарії дані збиралися в 9 точках, а в міському – у 4 точках [28, с. 3].

У роботі використано прототип на основі Arduino MKR WAN 1310, Raspberry Pi 3 model B+ як шлюзу та The Things Stack як мережевого сервера [28, с. 5]. До апаратної частини входили датчики температури й вологості ST HTS221, барометричний датчик ST LPS22 HB, датчик освітленості VISHAY TEMT 6000, ультрафіолетовий датчик VISHAY VEMML 6075, напівхвильова дипольна антена та Li-ion батарея [28, с. 6]. Це показує типову структуру LoRaWAN-вузла для моніторингу параметрів середовища.

У результатах Кіто простежується зв'язок між якістю сигналу, RSSI, SNR, spreading factor і airtime. Нижчі значення RSSI, приблизно від -105 до -100 dBm, пов'язувалися з більшим airtime при вищих значеннях SF, наприклад SF9 або SF10 [28, с. 4]. Це означає, що слабший сигнал може залишатися придатним для зв'язку, але система компенсує його збільшенням spreading factor, через що пакет довше перебуває в ефірі та споживає більше енергії.

У міських умовах із високими будівлями дальність була меншою через перешкоди, але зв'язок залишався стабільним на відстанях до 2 км. У сільській місцевості без великої кількості перешкод сигнал досягав до 5 км [28, с. 4–5]. Також зазначено, що збільшення відстані підвищує ймовірність втрат пакетів, а в місті навіть невеликі зміни відстані або положення вузла можуть спричинити повну втрату зв'язку між вузлом і шлюзом [28, с. 5].

Таблиця 1.14

Порівняння міського та сільського поширення LoRaWAN за результатами Кіто

Умова	Міське середовище	Сільське середовище
Розміщення шлюзу	тераса будівлі в районі La Florida	платформа 3 м у зоні Cocotog
Характер середовища	щільна забудова, перешкоди, інтерференція	низька щільність забудови, рівніший рельєф
Кількість точок вимірювання	4 точки	9 точок
Дальність стабільного зв'язку	до 2 км	до 5 км
Основна проблема	скорочення дальності через будівлі та завади	менший вплив перешкод
Поведінка сигналу	слабший RSSI потребує вищого SF і більшого airtime	кращі умови для поширення сигналу

Отже, поширення радіосигналу в умовах міської забудови визначається не лише відстанню між вузлом і шлюзом, а й просторовою структурою середовища. Щільність будівель, NLOS-умови, дифракція, відбиття, висота антен і розміщення шлюзу можуть суттєво змінювати RSSI, SNR, airtime і кількість втрачених пакетів.

Втрати пакетів в IoT-мережах виникають тоді, коли повідомлення від сенсорного вузла не доходить до шлюзу, сервера або іншого приймача. Причинами можуть бути слабкий рівень сигналу, завади, перевантаження каналу, колізії між повідомленнями різних вузлів, помилки в бітах, некоректна маршрутизація або навмисний вплив на канал зв'язку. Для IoT-систем це особливо критично, бо більшість сенсорних вузлів передають невеликі пакети даних, але саме ці пакети містять основні параметри моніторингу: температуру, вологість, стан пристрою, рівень сигналу, повідомлення про подію або аварійний стан.

В. І. Маліновський, Л. М. Куперштейн, В. В. Лукічов та А. В. Дудатьєв розглядають проблеми передавання даних в IoT через ризики порушення цілісності, стабільності й захищеності інформаційних потоків. До основних загроз автори відносять перехоплення і спотворення даних у каналах IoT, ін'єкцію шкідливого коду, перехоплення трафіку, втручання в інформаційні

потоки, перехоплення керування пристроями, DoS/DDoS-атаки, підміну програмного забезпечення, MITM-атаки, атаки на Wi-Fi, Bluetooth і кабельні комунікації [6, с. 106–107]. Такі впливи можуть не тільки порушувати безпеку, а й призводити до втрат пакетів, затримок, підміни повідомлень або передавання некоректних даних.

Окремо автори звертають увагу на те, що багато IoT-протоколів і каналів передавання, зокрема LoRa/LoRaWAN, ZigBee, AMQP, MQTT, HTTP/HTTPS, CoAP, DDS, Wi-Fi/WiMax, Z-Wave, RFID, Bluetooth, LTE/BLE та NFC, мають схожі проблеми захисту й надійності. На системному рівні шифрування та захисні механізми часто не працюють автоматично, а потребують налаштування і контролю [6, с. 108]. Через це передавання даних може бути вразливим не лише до фізичних завад, а й до помилок конфігурації, слабкого захисту та некоректної роботи мережевих компонентів.

G. Ferré та A. Giremus досліджують колізії та втрати пакетів саме в LoRaWAN-мережі. У LoRaWAN Class A доступ до каналу побудований за принципом ALOHA: вузол передає пакет тоді, коли він готовий до передавання, без повної синхронізації з іншими пристроями [24, с. 2654]. Через це два або більше вузлів можуть одночасно використовувати один канал і той самий spreading factor, що створює колізію. Автори показують, що класичний підхід через пуассонівський процес не завжди точно описує таку ситуацію, бо час перебування пакета в ефірі та інтервал між передачами залежать від конкретного IoT-застосування [24, с. 2654].

У LoRaWAN колізія залежить від кількості вузлів, часу перебування пакета в ефірі, spreading factor, розміру payload і режиму підтвердження. Якщо пакет перебуває в ефірі довше, вікно можливого накладання з іншими передачами збільшується. Для моделювання автори розглядали систему з одним шлюзом і N вузлами, які використовують однаковий spreading factor; ширина смуги становила 125 кГц, duty cycle – 1 %, а вузли отримували доступ до каналу випадково в часі [24, с. 2657].

Таблиця 1.15

Час перебування пакета LoRaWAN в ефірі залежно від spreading factor

Spreading factor	Мінімальний time on air, мс	Максимальний time on air, мс	Розмір payload, байти
SF12	761,86	3219,49	1–59
SF11	380,93	1740,80	1–59
SF10	190,46	935,94	1–59
SF9	95,23	992,26	1–123
SF8	47,62	987,65	1–230
SF7	23,81	567,55	1–230

З таблиці видно, що збільшення spreading factor суттєво підвищує час перебування пакета в ефірі. Наприклад, для SF12 максимальний time on air становить 3219,49 мс, тоді як для SF7 – 567,55 мс [24, с. 2657]. Для міської IoT-мережі це має практичне значення: високий SF допомагає передавати сигнал на більшу відстань або в складніших умовах, але водночас збільшує ризик колізій, бо пакет довше займає канал.

Моделювання показало, що для 1000 вузлів ймовірність хоча б однієї колізії залежить від spreading factor і розміру payload. З погляду ймовірності колізій автори вказують, що вигідніше використовувати SF8, SF9 або SF12, тоді як SF10 і SF11 дають таку саму ймовірність колізії, як SF12, але мають нижчу чутливість [24, с. 2658]. Це означає, що вибір SF не можна робити тільки за дальністю: потрібно враховувати ще завантаження мережі, кількість вузлів, розмір пакета і ризик накладання передач.

Втрати пакетів у LoRaWAN можуть виникати не лише через накладання uplink-повідомлень від вузлів. G. Ferré та A. Giremus показують ще одну важливу ситуацію: якщо шлюз передає АСК-підтвердження одному вузлу, він у цей момент не може приймати інший пакет. Тому повідомлення іншого вузла може бути втрачене, якщо воно надходить під час передавання АСК від шлюзу [24, с. 2656–2657]. Для завадостійкої системи це важливо, бо підтвердження доставки підвищує надійність одного повідомлення, але в навантаженій мережі може створювати додаткові втрати для інших вузлів.

Таблиця 1.16

Основні причини втрат і помилок у IoT-мережах

Причина	Як проявляється	Наслідок для системи
Колізія пакетів	два або більше вузлів передають одночасно	пакет не приймається шлюзом [24, с. 2654–2656]
Довгий time on air	пакет довго займає канал	зростає шанс накладання з іншими передачами [24, с. 2657]
АСК від шлюзу	шлюз передає підтвердження і тимчасово не приймає інші пакети	можливі втрати пакетів інших вузлів [24, с. 2656–2657]
Бітові помилки	пакет прийнятий, але частина бітів пошкоджена	CRC не проходить, пакет відкидається [33, с. 1–2]
Кібервтручання або спотворення даних	перехоплення, MITM, DoS/DDoS, підміна трафіку	порушення цілісності, доступності й стабільності передавання [6, с. 106–108]

Е. Tsimbalo, Х. Fafoutis та Р. J. Piechocki розглядають помилки передавання через механізм CRC у BLE та IEEE 802.15.4. У цих стандартах CRC традиційно використовується для виявлення пошкоджених пакетів: якщо після приймання перевірка не проходить, пакет вважається помилковим і відкидається [33, с. 1–2]. На прикладному рівні це сприймається як втрата пакета. У режимі без підтверджень така втрата обмежує дальність роботи системи, а в режимі з підтвердженнями спричиняє повторні передачі й додаткове енергоспоживання передавача [33, с. 1].

У BLE використовується CRC-24, тобто до пакета додається 24 надлишкові біти, а в IEEE 802.15.4 – CRC-16 із 16 надлишковими бітами [33, с. 2]. Автори пропонують не тільки виявляти помилки через CRC, а й використовувати цю надлишковість для виправлення частини пошкоджених пакетів на приймальному боці. Такий підхід не змінює передавач і не додає йому енергетичного навантаження, оскільки додаткова обробка виконується вже на приймачі [33, с. 1–2].

Під час моделювання для BLE розглядалися пакети розміром 8, 21 і 39 байтів. Для цих розмірів корекція CRC дала вигоду за SNR приблизно 2,5 дБ, 2 дБ і 1,8 дБ відповідно при packet error rate 10^{-2} [33, с. 4]. Менші пакети

виграли більше, бо в них частка CRC-надлишковості відносно корисних даних вища. Це має значення для IoT-систем, де повідомлення часто короткі: температура, вологість, стан вузла або рівень сигналу можуть передаватися невеликими пакетами, для яких контроль і виправлення помилок можуть бути ефективнішими.

Таблиця 1.17

Результати використання CRC-корекції для зменшення помилок передавання

Показник	Результат	Практичне значення
BLE, пакети 8 байтів	виграш SNR до 2,5 дБ при PER 10^{-2}	можливе збільшення дальності або зменшення потужності передавання [33, с. 4]
BLE, пакети 21 байт	виграш SNR близько 2 дБ	ефективно для коротких сенсорних повідомлень [33, с. 4]
BLE, пакети 39 байтів	виграш SNR близько 1,8 дБ	ефект менший, але все ще помітний [33, с. 4]
BLE-21 у реальних умовах	виправлено до 35 % пошкоджених пакетів	зменшення кількості втрат і повторних передач [33, с. 6]
BLE-39 у реальних умовах	виправлено до 22 % пошкоджених пакетів	ефективність нижча для більших пакетів [33, с. 6]
IEEE 802.15.4	виправлено до 15 % пакетів 21 байт і до 12 % пакетів 39 байтів	CRC-16 має менший потенціал виправлення, ніж CRC-24 [33, с. 6]

Реальні випробування проводилися на платформі TI CC2650, яка підтримує BLE та IEEE 802.15.4. Пакети збиралися в лабораторному середовищі з робочими столами, обладнанням, офісними меблями, комп'ютерами та активними Wi-Fi-мережами в тому самому діапазоні 2,4 ГГц [33, с. 5]. Було сформовано чотири набори даних: BLE-21, BLE-39, 802.15.4-21 і 802.15.4-39; кожен містив щонайменше 1000 пошкоджених пакетів [33, с. 5]. Така ситуація близька до реальних IoT-умов у приміщенні або міській забудові, де сенсорний канал працює поруч з іншими бездротовими мережами.

За результатами реальних даних, метод ADMM зміг виправити до 35 % пошкоджених пакетів BLE-21 і до 22 % BLE-39. Для IEEE 802.15.4 результати були нижчими: до 15 % для пакетів 21 байт і до 12 % для пакетів

39 байтів [33, с. 6]. У висновках автори також підкреслюють, що більшість реальних пошкоджених пакетів мали не більше 3 бітових помилок, тому CRC-корекція має практичний потенціал для зменшення втрат у коротких IoT-повідомленнях [33, с. 8].

Для завадостійкої IoT-системи ці результати мають два наслідки. По-перше, частину втрат можна зменшити не тільки повторною передачею, а й виправленням помилок на приймальному боці. По-друге, повторна передача не завжди є найкращим рішенням: вона підвищує енергоспоживання вузла, збільшує навантаження на канал і може створювати нові колізії. Тому CRC-перевірка, аналіз помилок, контроль кількості повторних передач і правильний вибір параметрів каналу мають працювати разом.

Втрати пакетів, колізії та помилки передавання в IoT-мережах формуються на різних рівнях. На фізичному та MAC-рівні вони пов'язані з завадами, випадковим доступом до каналу, довгим time on air, ACK-повідомленнями та одночасною роботою багатьох вузлів [24, с. 2654–2658]. На рівні цілісності даних помилки проявляються через пошкоджені біти, невдалу CRC-перевірку, відкидання пакета та потребу в повторній передачі [33, с. 1–2]. На рівні безпеки й стабільності додаткові ризики створюють перехоплення, спотворення трафіку, DoS/DDoS, MITM-атаки та некоректні налаштування захисту [6, с. 106–108]. Для міської IoT-системи це означає, що завадостійкість має включати не один механізм, а поєднання контролю каналу, вибору параметрів передавання, CRC-перевірки, повторної передачі та захисту від втручання в інформаційний обмін.

Моделювання IoT-каналів зв'язку потрібне для того, щоб ще до повного впровадження системи оцінити поведінку мережі при різних кількості вузлів, зміні відстані до шлюзу, появі завад, втрат пакетів, колізій і перевантаження каналу. У реальному міському середовищі складно одразу перевірити всі можливі сценарії, тому частину процесів переносять у симуляційне середовище. Так можна порівняти варіанти розміщення вузлів,

типи протоколів, параметри передавання, кількість шлюзів і показники надійності без зайвих витрат на фізичне розгортання мережі.

Т. А. Самолук розглядає напівнатурне моделювання як підхід, за якого частина системи моделюється, а інша частина залишається реальною [10, с. 101]. Для IoT-систем це зручний варіант, оскільки реальний сенсорний вузол може передавати дані, а поведінка каналу, хмарного середовища або мережевих затримок може відтворюватися програмно. У такій схемі легше перевірити, як система реагує на втрати, затримки, зниження пропускну здатності або зміну кількості підключених пристроїв.

Вбудовані IoT-системи, за описом Т. А. Самолук, мають зв'язок із хмарними онлайн-сервісами, компоненти підключення, зокрема IoT-шлюзи, а також потребують налаштування хмарного середовища, проєктування інтерфейсів користувача й фінального тестування [10, с. 98]. Для моделювання це означає, що перевіряти потрібно не тільки сам радіоканал, а весь шлях даних: датчик – мікроконтролер – бездротовий канал – шлюз – сервер – інтерфейс користувача.

Бездротові сенсорні мережі описуються як самоорганізовані мережі, що складаються з великої кількості датчиків і виконавчих пристроїв, об'єднаних між собою радіосигналами. До таких технологій віднесено RFID, NFC, Bluetooth, BLE, Wi-Fi/802.11, Z-Wave, Zigbee, LoRaWAN, 4G LTE, 5G, Cat-M1, NB-IoT, Cat-M2 і Sigfox [10, с. 99]. Для оцінювання завадостійкості важливо враховувати смугу частот, швидкість передавання, структуру пакета, методи авторизації та захисту даних, дальність дії, енергоефективність, методи модуляції, керування частотною смугою і поділ каналів [10, с. 99].

Таблиця 1.18

Параметри моделювання IoT-каналів зв'язку

Параметр	Що показує	Значення для оцінювання завадостійкості
RSSI	рівень прийнятого сигналу	дозволяє оцінити ослаблення сигналу між вузлом і шлюзом
SNR	співвідношення сигнал/шум	показує, наскільки сигнал відрізняється від шумового фону
PDR	частка доставлених пакетів	основний показник надійності доставки даних
PLR / PER	частка втрачених або помилкових пакетів	відображає втрати через завади, колізії або слабкий сигнал
Spreading Factor	параметр LoRa, що впливає на дальність і швидкість	вищий SF підвищує дальність, але збільшує час передавання
Кількість вузлів	навантаження на мережу	зі збільшенням кількості вузлів зростає ризик колізій
Кількість шлюзів	просторове покриття мережі	кілька шлюзів можуть підвищити надійність приймання

B. Citoni, S. Ansari, Q. H. Abbasi, M. A. Imran та S. Hussain досліджували міське розгортання LoRaWAN навколо кампусу Університету Глазго та порівнювали його з NS-3-симуляцією. Було розгорнуто мережу з 20 LoRaWAN-вузлів, після чого створено симуляційну модель, максимально наближену до реальної топології [20, с. 17216]. Автори підкреслюють, що симуляції LoRaWAN корисні для аналізу сценаріїв, які складно або дорого реалізувати фізично через масштаб і кількість потрібних пристроїв [20, с. 17216].

Вузли в експерименті були поділені на 5 груп передавання. Перша група передавала дані кожні 300 с, друга – настільки швидко, наскільки дозволяв поточний spreading factor, третя – з випадковим інтервалом від 300 до 900 с, четверта – з випадковим інтервалом від найшвидшого можливого для поточного SF до потрійного значення цього інтервалу, а п'ята – кожні 600 с протягом 12 годин і приблизно кожні 7200 с протягом наступних 12 годин [20, с. 17218]. Такий підхід добре показує, що в реальній IoT-мережі вузли можуть мати різні режими передавання, а не працювати за одним однаковим графіком.

У мережі використовувалися три власні шлюзи, але під час експерименту пакети приймали також сторонні шлюзи мережі IoT Scotland. Загалом було зафіксовано приймання даних 15 різними шлюзами [20, с. 17218]. Алгоритм ADR був увімкнений на вузлах і на мережевому сервері, тому spreading factor і потужність передавання могли змінюватися протягом роботи мережі [20, с. 17219]. Експеримент тривав близько місяця, але всі вузли розрядилися приблизно через 3,5 тижні [20, с. 17219].

Для оцінювання якості роботи автори використовували PDR – Packet Delivery Ratio, а також згадували PLR, RSSI, SNR і throughput як поширені метрики аналізу LoRaWAN [20, с. 17219]. PDR розраховується як відношення кількості прийнятих пакетів до кількості надісланих пакетів, помножене на 100 % [20, с. 17220]. У результатах було помітно, що групи з повільнішим інтервалом передавання мали кращий PDR: Tx Groups 1 і 5 показали приблизно на 7 % вищий PDR, ніж Tx Groups 2 і 4 [20, с. 17220]. Це підтверджує, що частіше передавання підвищує ризик колізій і втрат.

Таблиця 1.19

Приклад параметрів реального розгортання LoRaWAN і симуляції

Показник	Значення
Кількість LoRaWAN-вузлів	20
Кількість груп передавання	5
Інтервал передавання групи 1	300 с
Інтервал передавання групи 3	300–900 с
Інтервал передавання групи 5	600 с вдень і 7200 с вночі
Кількість шлюзів, які приймали пакети	15
Час роботи вузлів від батареї	близько 3,5 тижнів
Основна метрика	PDR

Моделювання в NS-3 будувалося на реальних координатах вузлів і шлюзів, а також на наближеному врахуванні будівель, у яких були розміщені пристрої [20, с. 17221]. Для моделі каналу використовувалися Log-Distance, Okumura-Hata та Friis, а також додаткові втрати, пов'язані з будівлями, fast fading і slow fading [20, с. 17221–17222]. Порівняння показало, що симуляційний і емпіричний PDR загалом відрізнялися в межах приблизно 20 % [20, с. 17222]. Це важливий результат: симуляція не відтворює реальну

мережу ідеально, але може дати достатньо близьке наближення для аналізу впливу параметрів.

У таблиці з порівнянням моделей видно, що «ідеальна» мережа без path loss дала середню різницю PDR 25,14 %, Okumura-Hata – 40,58 %, а варіант Log-Distance з емпірично отриманими коефіцієнтами, без building loss, але з fast і slow fading – 21,3 % [20, с. 17222]. Це показує, що для міської забудови важливо не просто взяти стандартну модель поширення сигналу, а налаштовувати її за реальними RSSI-даними.

В. Reynders, Q. Wang та S. Pollin представили LoRaWAN-модуль для NS-3, який відповідає специфікації LoRaWAN 1.0 Class A і дозволяє досліджувати вплив різних параметрів на продуктивність мережі [30, с. 61]. Автори підкреслюють, що через складну топологію LoRaWAN і ALOHA-характер MAC-рівня одних аналітичних формул недостатньо для оцінювання роботи мережі в широкому наборі сценаріїв. Симуляція дає змогу перевірити вплив часу відповіді ACK, spreading factor, bandwidth та інших параметрів [30, с. 61].

Структура LoRaWAN у цьому модулі відповідає типовій архітектурі: сенсорний вузол передає дані на шлюз за допомогою LoRa, після чого IP-мережа з вищою пропускнуою здатністю пересилає інформацію до мережевого сервера [30, с. 62]. Для LoRaWAN Class A вузли передають невелику кількість пакетів і більшу частину часу перебувають у режимі сну [30, с. 62]. Spreading factor може змінюватися від SF7 до SF12: нижчий SF дає вищу швидкість і коротший час передавання, але потребує вищого SNR і має меншу дальність [30, с. 62].

Модель також враховує технічні обмеження шлюзу. Шлюз може відстежувати до 8 LoRa-сигналів одночасно, що пов'язано з обмеженнями апаратної частини [30, с. 64]. Uplink і downlink у LoRaWAN використовують ті самі канали й ту саму модуляцію, тому шлюз не має повноцінної full-duplex-здатності: якщо він передає, то може знищити приймання повідомлень

на цьому каналі [30, с. 64]. Для завадостійкості це важливо, бо втрати можуть виникати не лише через слабкий сигнал, а й через режим роботи шлюзу.

Для оцінювання LoRaWAN-модуля автори провели три сценарії. Перший – проста кругова топологія з одним центральним шлюзом і непідтвердженими повідомленнями. Другий – подібна топологія з кількома шлюзами для перевірки просторового різноманіття. Третій – передавання підтверджених повідомлень, щоб показати, що низькопотужна мережа не може підтверджувати кожне повідомлення при великому навантаженні [30, с. 66]. У результатах зазначено, що біля шлюзу packet error ratio низький, а на більшій відстані якість погіршується; зі збільшенням кількості конкуруючих вузлів PER також зростає [30, с. 67].

Таблиця 1.20

Сценарії моделювання LoRaWAN у NS-3

Сценарій	Умови моделювання	Що перевіряється
Scenario 1	вузли розміщені по колу, один центральний шлюз, unconfirmed data	вплив відстані та кількості вузлів на PER [30, с. 66–67]
Scenario 2	кілька шлюзів у тій самій топології	вплив просторового різноманіття і кількості шлюзів [30, с. 66–67]
Scenario 3	confirmed messages	обмеження мережі при підтвердженні кожного повідомлення [30, с. 66–67]

У другому сценарії кілька шлюзів значно підвищували надійність приймання, оскільки вузол міг бути прийнятий не тільки центральним шлюзом, а й іншим шлюзом, ближчим до його позиції [30, с. 67]. У третьому сценарії при збільшенні кількості вузлів packet error ratio різко зростав, особливо для пристроїв, розміщених далеко від шлюзу [30, с. 67]. Це підтверджує, що механізм підтвердження доставки не завжди покращує систему: у перевантаженій мережі він може створювати додатковий трафік і підвищувати ризик втрат.

Для оцінювання енергоспоживання в модулі використовується окрема модель, яка відстежує стани фізичного рівня: RX, TX та IDLE. Для кожного стану задається відповідний струм, а використана енергія обчислюється через

множення струму на напругу живлення та час перебування в цьому стані [30, с. 66]. Це дає змогу оцінювати не лише якість передавання, а й автономність сенсорного вузла, що важливо для IoT-систем у міському середовищі.

Таблиця 1.21

Напрями оцінювання завадостійкості IoT-системи

Напрямок оцінювання	Що аналізується	Приклад показника
Якість радіоканалу	рівень сигналу, шум, відстань, втрати на трасі	RSSI, SNR
Доставка пакетів	скільки повідомлень дійшло до шлюзу або сервера	PDR, PLR, PER
Навантаження на мережу	частота передавання, кількість вузлів, колізії	інтервал TX, кількість одночасних вузлів
Параметри LoRa	SF, BW, потужність, час пакета в ефірі	SF7–SF12, airtime
Інфраструктура	кількість і розміщення шлюзів	один шлюз / кілька шлюзів
Енергоспоживання	витрати енергії в TX, RX та IDLE	час автономної роботи вузла

Отже, моделювання IoT-каналів зв'язку поєднує апаратну, мережеву й програмну частини системи. Напівнатурний підхід дає змогу частину системи залишити реальною, а частину – відтворити програмно [10, с. 101]. Реальні міські розгортання LoRaWAN показують, що PDR залежить від інтервалу передавання, розміщення вузлів і шлюзів, роботи ADR, кількості приймачів та моделі втрат сигналу [20, с. 17218–17222]. NS-3-моделювання дає змогу перевірити вплив spreading factor, bandwidth, ACK, кількості шлюзів, packet error ratio та енергоспоживання без повного фізичного розгортання мережі [30, с. 61–67]. Для завадостійкої системи в умовах міської забудови такий підхід дозволяє оцінити, які параметри найбільше впливають на стабільність передавання даних і які рішення зменшують втрати пакетів.

Висновки

IoT-телекомунікаційні системи є складними багаторівневими структурами, у яких поєднуються сенсори, мікроконтролери, канали зв'язку, шлюзи, серверні платформи, бази даних і веб-інтерфейси. Їхня робота базується на послідовному передаванні сенсорних даних від фізичного об'єкта до програмного середовища, де інформація зберігається, обробляється та використовується для моніторингу або керування. Надійність такої системи залежить не тільки від апаратної частини, а й від стабільності каналу зв'язку, правильного вибору протоколу, захисту даних і здатності мережі працювати за наявності перешкод.

Для IoT-систем можуть застосовуватися різні протоколи та технології зв'язку, зокрема Wi-Fi, LoRa та MQTT. Wi-Fi зручний для локального передавання даних і швидкого підключення до наявної мережевої інфраструктури, однак він залежить від покриття, завантаженості каналу та впливу інших бездротових мереж. LoRa краще підходить для передавання невеликих обсягів даних на більші відстані та може використовуватися в автономних сенсорних мережах. MQTT забезпечує зручний обмін повідомленнями між пристроями, сервером і користувацьким інтерфейсом, але ефективність його роботи також залежить від затримок, втрат пакетів і стабільності мережевого з'єднання.

Міська забудова створює складні умови для функціонування IoT-мереж, оскільки стіни, перекриття, металеві конструкції, відбиття сигналу, електромагнітні завади та велика кількість підключених пристроїв можуть погіршувати якість зв'язку. У таких умовах зростає ризик послаблення сигналу, затримок, втрати пакетів і порушення цілісності повідомлень. Тому побудова завадостійкої IoT-системи має ґрунтуватися на поєднанні правильного вибору каналу зв'язку, контролю цілісності даних, повторної передачі повідомлень, захисту мережі та моделювання впливу міського середовища на передавання сенсорної інформації.

РОЗДІЛ 2

ПРОЄКТУВАННЯ ЗАВАДОСТІЙКОЇ ІОТ-СИСТЕМИ ДЛЯ ПЕРЕДАВАННЯ СЕНСОРНИХ ДАНИХ

2.1. Розроблення архітектури ІоТ-системи та вибір апаратних компонентів

Запроектована ІоТ-система призначена для збирання сенсорних показників, передавання їх через бездротовий канал зв'язку, контролю цілісності повідомлень і подальшого відображення результатів у веб-інтерфейсі. Система має завершену структуру: сенсорний вузол фіксує параметри середовища, мікроконтролер обробляє ці значення, модуль зв'язку передає повідомлення, серверна частина приймає дані, база даних зберігає результати, а веб-інтерфейс відображає поточний стан системи.

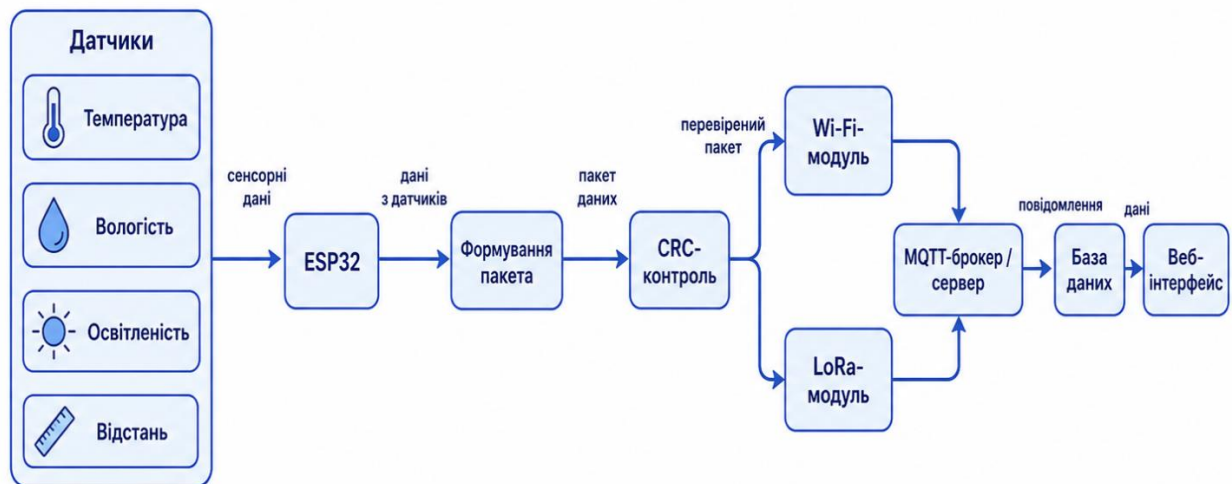


Рис. 2.1 – Структурна схема запроєктованої завадостійкої ІоТ-системи

Структурна схема ІоТ-системи складається з таких основних блоків: сенсорний блок, мікроконтролер ESP32, блок формування пакета даних, блок CRC-контролю, модуль бездротового зв'язку, сервер приймання повідомлень, база даних і веб-інтерфейс моніторингу. Такий поділ показує

повний шлях проходження інформації від моменту вимірювання до її збереження й аналізу.

Передавання даних у системі відбувається послідовно. Спочатку датчики фіксують параметри середовища. ESP32 зчитує отримані значення, формує інформаційний пакет, додає до нього номер вузла, час вимірювання, тип датчика, значення показника, номер пакета та контрольну суму CRC. Після цього повідомлення передається через Wi-Fi або LoRa. Сервер приймає пакет, перевіряє його цілісність, записує коректні дані в базу, а повідомлення з помилками фіксує в журналі подій.

У запроєктованій системі Wi-Fi використовується для локального передавання даних на сервер або MQTT-брокер, а LoRa – для передавання коротких сенсорних повідомлень на більшу відстань. Таке поєднання дає змогу перевірити роботу системи у двох режимах: локальному, коли вузол перебуває в зоні Wi-Fi-покриття, і розширеному, коли сенсорний вузол передає дані через LoRa-канал в умовах перешкод.

Структура інформаційного пакета має фіксований набір полів. До пакета входить ідентифікатор вузла, час вимірювання, тип датчика, виміряне значення, номер пакета, CRC-код і статус обробки. Номер пакета використовується для виявлення пропущених повідомлень. CRC-код забезпечує перевірку цілісності даних. Статус обробки показує, чи пакет прийнято коректно, чи зафіксовано помилку передавання.

Таблиця 2.1

Структура інформаційного пакета сенсорних даних

Поле пакета	Призначення
Ідентифікатор вузла	Визначає, з якого сенсорного вузла надійшли дані
Час вимірювання	Фіксує момент отримання показника
Тип датчика	Вказує, який параметр передається
Значення показника	Містить числовий результат вимірювання
Номер пакета	Дозволяє виявляти пропущені повідомлення
CRC-код	Використовується для перевірки цілісності даних
Статус пакета	Показує результат приймання та перевірки повідомлення

Для оцінювання завадостійкості система зберігає не тільки сенсорні показники, а й службові параметри передавання. До них належать номер пакета, статус доставки, кількість повторних передач, результат CRC-перевірки, рівень сигналу та співвідношення сигнал/шум для LoRa-з'єднання. Такі дані потрібні для подальшого аналізу якості каналу й визначення умов, за яких виникають втрати повідомлень.

Сформована структурна схема забезпечує зв'язок між апаратною та програмною частинами системи. Сенсорний вузол відповідає за вимірювання, ESP32 – за обробку й формування пакета, модуль зв'язку – за передавання, сервер – за приймання, база даних – за збереження, а веб-інтерфейс – за візуалізацію результатів. Така структура створює основу для вибору конкретних компонентів апаратного вузла.

Апаратний вузол запроєктованої IoT-системи побудовано на базі мікроконтролера ESP32. Його обрано як центральний елемент системи, оскільки він забезпечує зчитування даних із сенсорів, первинну обробку показників, формування інформаційного пакета, роботу з Wi-Fi-з'єднанням і передавання даних на серверну частину. ESP32 підходить для навчального та практичного проєктування IoT-системи, оскільки поєднує обчислювальний модуль і бездротовий інтерфейс в одному пристрої.

До сенсорного блоку входять датчик температури й вологості DHT22, датчик відстані HC-SR04 та датчик освітленості BH1750. Такий набір дає змогу перевірити передавання різних типів даних: кліматичних параметрів, просторового показника та рівня освітленості. Для проєктування завадостійкої системи цього достатньо, оскільки головний акцент зроблено не на складності датчиків, а на стабільності передавання повідомлень від вузла до серверної частини.

Таблиця 2.2

Основні апаратні компоненти IoT-вузла

Компонент	Призначення в системі	Причина вибору
ESP32	Центральний мікроконтролерний вузол	Має вбудований Wi-Fi, підтримує підключення датчиків і обробку даних
DHT22	Вимірювання температури та вологості	Передає кліматичні показники для моніторингу середовища
HC-SR04	Вимірювання відстані	Дає змогу перевірити передавання просторових даних
BH1750	Вимірювання рівня освітленості	Формує додатковий сенсорний параметр для моніторингу
LoRa SX1276 / SX1278	Далеке бездротове передавання	Забезпечує передавання коротких пакетів на більшу відстань
Wi-Fi ESP32	Локальне передавання даних	Використовується для підключення до сервера або MQTT-брокера
Джерело живлення	Живлення апаратного вузла	Забезпечує автономну або стаціонарну роботу пристрою

ESP32 виконує роль головного керуючого елемента. Він приймає дані від датчиків, перетворює їх у єдиний формат повідомлення, додає службову інформацію та передає пакет через вибраний канал зв'язку. У системі також передбачено збереження номера кожного пакета. Це дає змогу відстежувати втрати повідомлень: якщо сервер отримує пакети з пропущеними номерами, система фіксує втрату даних.

Датчик DHT22 використовується для вимірювання температури й вологості. Ці показники часто застосовуються в IoT-моніторингу, тому вони підходять для перевірки регулярного передавання сенсорних даних. Датчик формує значення, які ESP32 зчитує з певним інтервалом і передає на сервер. У базі даних ці показники зберігаються разом із часом вимірювання, номером пакета та статусом доставки.

Датчик HC-SR04 використовується для вимірювання відстані. Його включено до системи для перевірки передавання даних іншого типу, не пов'язаного з температурою або вологістю. Таке рішення дозволяє показати, що запроектована IoT-система працює не з одним конкретним показником, а з різними сенсорними параметрами.

Датчик BH1750 використовується для вимірювання освітленості. Він доповнює сенсорний блок і дає змогу аналізувати зміну рівня світла в середовищі. Дані освітленості передаються в тому самому форматі, що й інші сенсорні значення, тому система зберігає єдину структуру повідомлень незалежно від типу датчика.

Для організації бездротового передавання використовується два канали: Wi-Fi та LoRa. Wi-Fi-з'єднання забезпечує передавання даних у межах локальної мережі. Цей канал використовується для тестування апаратного вузла, підключення до MQTT-брокера та передавання даних на серверну частину. LoRa-модуль SX1276 або SX1278 використовується для передавання коротких повідомлень на більшу відстань, що відповідає умовам міської забудови, де зв'язок ускладнюється перешкодами й ослабленням сигналу.

Таблиця 2.3

Функціональний розподіл каналів зв'язку в системі

Канал зв'язку	Роль у системі	Тип переданих даних
Wi-Fi	Локальне підключення ESP32 до сервера або MQTT-брокера	Сенсорні значення, статус вузла, службові параметри
LoRa	Передавання коротких повідомлень на більшу відстань	Сенсорні пакети, номер пакета, CRC-статус, параметри каналу
MQTT	Прикладний рівень обміну повідомленнями	Структуровані повідомлення між вузлом, брокером і веб-інтерфейсом

У запроєктованому апаратному вузлі кожен компонент має окрему функцію. Сенсори формують первинні дані, ESP32 відповідає за логіку роботи вузла, Wi-Fi забезпечує локальне передавання, LoRa розширює дальність зв'язку, а CRC-контроль перевіряє цілісність повідомлення. Така комплектація дає змогу не тільки передавати сенсорні показники, а й оцінювати якість передавання за службовими параметрами.

2.2. Обґрунтування вибору протоколів зв'язку та алгоритмів підвищення завадостійкості

Для запроєктованої IoT-системи обрано комбінацію Wi-Fi, LoRa/LoRaWAN та MQTT, оскільки ці технології виконують різні функції в загальній структурі передавання сенсорних даних. Wi-Fi забезпечує локальне підключення апаратного вузла до серверної частини, LoRa/LoRaWAN використовується для далекого передавання коротких повідомлень, а MQTT організовує прикладний обмін даними між вузлом, брокером, сервером і веб-інтерфейсом.

Wi-Fi використовується в системі як локальний канал передавання даних. Він забезпечує підключення ESP32 до мережі та передавання сенсорних показників на сервер або MQTT-брокер. Такий варіант підходить для тестування апаратного вузла в приміщенні, лабораторії або в межах невеликої ділянки. Основна перевага Wi-Fi полягає у високій швидкості обміну, простому підключенні до наявної мережі та зручній інтеграції з веб-сервером.

LoRa/LoRaWAN використовується як канал для передавання коротких повідомлень на більшу відстань. Для системи, яка працює в умовах міської забудови, це важливо через наявність стін, будівель, металевих конструкцій та інших перешкод. LoRa забезпечує нижчу швидкість, ніж Wi-Fi, але має більшу дальність і краще підходить для автономних сенсорних вузлів, які передають невеликі пакети даних через певний інтервал часу.

MQTT використовується як прикладний протокол обміну повідомленнями. Він не замінює Wi-Fi або LoRa, а працює поверх мережевого з'єднання й організовує передачу даних за принципом «публікація – підписка». Сенсорний вузол публікує повідомлення в певну тему, MQTT-брокер приймає його, а сервер або веб-інтерфейс отримує дані через підписку на цю тему. Такий підхід спрощує структуру обміну й

дозволяє окремо передавати сенсорні значення, службові параметри каналу, повідомлення про помилки та статуси доставки.

Таблиця 2.4

Порівняння протоколів зв'язку, використаних у запроєктованій IoT-системі

Технологія	Роль у системі	Переваги	Обмеження
Wi-Fi	Локальне передавання даних від ESP32 до сервера або MQTT-брокера	Висока швидкість, просте підключення, підтримка роботи з веб-сервером	Обмежена дальність, залежність від покриття роутера, чутливість до перевантаження мережі
LoRa/LoRaWAN	Далеке передавання коротких сенсорних повідомлень	Велика дальність, низьке енергоспоживання, придатність для автономних вузлів	Низька швидкість, невеликий обсяг пакета, ризик колізій при великій кількості вузлів
MQTT	Прикладний обмін повідомленнями між вузлом, брокером і веб-інтерфейсом	Зручна модель publish/subscribe, підтримка QoS, поділ даних за темами	Залежність від брокера, збільшення затримки при повторних передачах і високих рівнях QoS

У запроєктованій системі Wi-Fi використовується для основного локального тестування та швидкого передавання даних на серверну частину. LoRa додається як канал для перевірки роботи системи в умовах більшої відстані та перешкод. MQTT забезпечує логіку обміну повідомленнями й структурує дані за окремими темами. Наприклад, сенсорні значення передаються в одну тему, параметри каналу – в іншу, а повідомлення про помилки – в окрему службову тему.

Таблиця 2.5

Розподіл функцій між Wi-Fi, LoRa/LoRaWAN та MQTT

Частина обміну	Використане рішення	Призначення
Передавання в локальній мережі	Wi-Fi	Надсилення даних із ESP32 на сервер або MQTT-брокер
Передавання на більшій відстані	LoRa/LoRaWAN	Передавання коротких повідомлень від сенсорного вузла до приймача або шлюзу
Логіка повідомлень	MQTT	Публікація сенсорних даних, статусів і параметрів каналу
Контроль доставки	MQTT QoS + номер пакета	Перевірка факту отримання повідомлення
Контроль цілісності	CRC	Виявлення пошкоджених повідомлень

Для системи обрано комбінований підхід, оскільки один протокол не закриває всі задачі. Wi-Fi забезпечує швидку локальну передачу, LoRa розширює дальність роботи вузла, а MQTT формує зручний прикладний рівень обміну. Така схема відповідає задачі проєктування завадостійкої IoT-системи, де важливе не тільки передавання показників, а й контроль стану каналу, виявлення помилок і фіксація втрачених пакетів.

Для підвищення завадостійкості в запроєктованій IoT-системі використовується поєднання CRC-контролю, нумерації пакетів, статусу доставки та повторної передачі повідомлень. Ці механізми забезпечують виявлення пошкоджених або втрачених пакетів і дозволяють оцінити стабільність роботи каналу зв'язку.

CRC-контроль використовується для перевірки цілісності повідомлення. Перед передаванням ESP32 формує інформаційний пакет із сенсорними даними та обчислює для нього контрольну суму. Після приймання серверна частина повторно перевіряє CRC. Якщо контрольне значення збігається, пакет отримує статус коректного. Якщо значення не збігається, пакет позначається як пошкоджений, а інформація про помилку записується в журнал подій.

Нумерація пакетів використовується для виявлення втрат. Кожне повідомлення отримує унікальний порядковий номер. Сервер перевіряє послідовність отриманих номерів. Якщо після пакета №105 надходить пакет

№107, система фіксує втрату пакета №106. Такий механізм дозволяє відрізнити пошкоджене повідомлення від повністю втраченого.

Повторна передача використовується для тих повідомлень, які не були підтверджені сервером або прийняті з помилкою CRC. Після відправлення пакета вузол очікує підтвердження. Якщо підтвердження не надходить у визначений час, виконується повторне передавання. Кількість повторних спроб обмежується, щоб система не створювала надмірне навантаження на канал зв'язку.

Таблиця 2.6

Механізми підвищення завадостійкості IoT-системи

Механізм	Призначення	Результат використання
CRC-контроль	Перевірка цілісності повідомлення	Виявлення пошкоджених пакетів
Номер пакета	Контроль послідовності повідомлень	Виявлення пропущених пакетів
Підтвердження доставки	Перевірка факту приймання пакета	Визначення успішної або невдалої передачі
Повторна передача	Повторне надсилання пакета після помилки або тайм-ауту	Зменшення кількості втрачених повідомлень
Журнал подій	Фіксація помилок і статусів	Аналіз якості роботи каналу

Алгоритм передавання повідомлення в системі має таку послідовність. Спочатку ESP32 зчитує значення з датчика та формує пакет. До нього додаються ідентифікатор вузла, час вимірювання, тип датчика, значення показника, номер пакета та CRC-код. Далі пакет передається через Wi-Fi або LoRa. Сервер приймає повідомлення, перевіряє його структуру та CRC. Якщо пакет коректний, дані записуються в базу, а вузол отримує підтвердження. Якщо пакет пошкоджений або підтвердження не отримано, виконується повторна передача.

Таблиця 2.7

Послідовність роботи алгоритму контролю цілісності та повторної передачі

Етап	Дія системи	Результат
1	Зчитування сенсорних даних	Отримано значення температури, вологості, освітленості або відстані
2	Формування пакета	Створено повідомлення з номером вузла, часом, типом датчика і значенням
3	Обчислення CRC	До пакета додано контрольну суму
4	Передавання пакета	Повідомлення надіслано через Wi-Fi або LoRa
5	Приймання на сервері	Пакет отримано серверною частиною
6	Перевірка CRC	Визначено коректність або пошкодження повідомлення
7	Запис у базу даних	Коректні дані збережено
8	Контроль підтвердження	Вузол отримує статус успішної доставки
9	Повторна передача	Пакет надсилається повторно після помилки або тайм-ауту
10	Фіксація в журналі	Записано статус пакета, кількість повторів і тип помилки

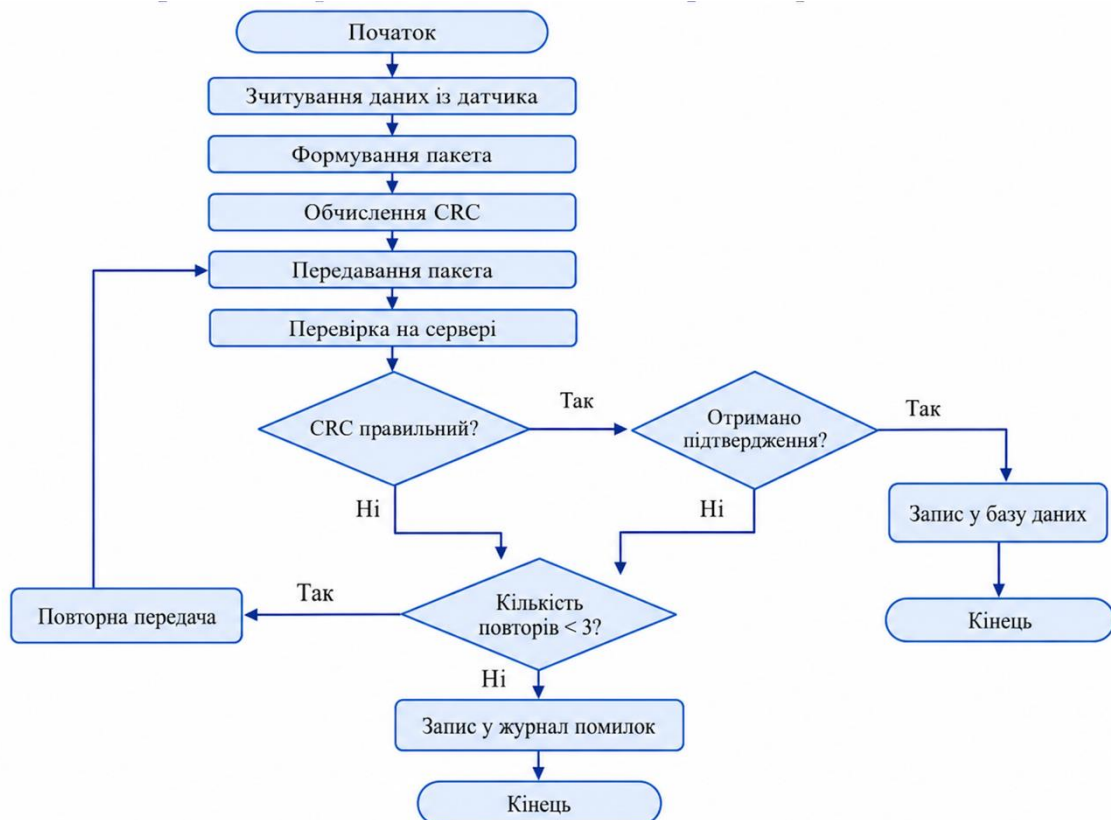


Рис. 2.2 – Алгоритм контролю цілісності та повторної передачі повідомлень

Для обмеження навантаження на канал у системі встановлюється максимальна кількість повторних передач. Після вичерпання ліміту пакет отримує статус недоставленого, а подія записується в базу даних. Такий підхід не блокує роботу вузла на одному повідомленні й дозволяє продовжити передавання наступних даних.

У базі даних для кожного пакета фіксуються сенсорні значення та службові параметри: номер пакета, статус CRC-перевірки, кількість повторних передач, статус доставки, час приймання, а для LoRa-з'єднання – RSSI та SNR. Завдяки цьому система не тільки передає дані, а й накопичує інформацію для оцінювання завадостійкості.

Таблиця 2.8

Статуси пакетів у запроєктованій системі

Статус пакета	Значення	Статус пакета
Прийнято	Пакет отримано, CRC-перевірка пройдена, дані записано в базу	Прийнято
Пошкоджено	Пакет отримано, але CRC-перевірка не пройдена	Пошкоджено
Втрачено	Пакет не надійшов на сервер або пропущений за номером	Втрачено
Повторено	Пакет передано повторно після помилки або відсутності підтвердження	Повторено
Недоставлено	Пакет не доставлено після встановленої кількості повторних спроб	Недоставлено

Розроблений алгоритм підвищує надійність передавання даних у складних умовах зв'язку. CRC-контроль виявляє пошкоджені повідомлення, нумерація пакетів показує втрати, повторна передача зменшує кількість недоставлених даних, а журнал подій дає змогу оцінити якість каналу. Для IoT-системи в умовах міської забудови це важливо, оскільки перешкоди, завади й перевантаження мережі впливають не лише на рівень сигналу, а й на фактичну доставку сенсорних повідомлень.

2.3. Проєктування структури бази даних і веб-інтерфейсу для моніторингу параметрів системи

Серверна частина запроєктованої IoT-системи виконує приймання сенсорних повідомлень, перевірку їхньої цілісності, збереження результатів і відображення параметрів у веб-інтерфейсі. У програмній частині фіксуються сенсорні показники та службові параметри передавання: номер пакета, результат CRC-перевірки, кількість повторних передач, статус доставки, рівень сигналу та співвідношення сигнал/шум. Завдяки цьому система відображає не тільки значення датчиків, а й якість роботи каналу зв'язку.

У програмній частині використовується зв'язка Flask + SQLite. Flask виконує приймання повідомлень від IoT-вузла, обробку запитів і формування веб-інтерфейсу моніторингу. SQLite зберігає сенсорні показники, параметри каналу зв'язку, результати CRC-перевірки, кількість повторних передач і записи журналу подій. Така побудова поєднує апаратний вузол, серверну частину, базу даних і панель візуалізації результатів в єдину систему моніторингу.

База даних запроєктованої IoT-системи зберігає інформацію про сенсорні вузли, результати вимірювань, параметри каналу зв'язку та події, пов'язані з передаванням даних. Такий поділ відокремлює основні сенсорні значення від службової інформації. У результаті в системі фіксуються температура, вологість, освітленість, відстань, номер пакета, статус доставки, результат CRC-перевірки, кількість повторних передач, RSSI та SNR.



Рис. 2.3 – Структура бази даних IoT-системи

Структура бази даних складається з чотирьох основних таблиць: «Сенсорні вузли», «Сенсорні показники», «Параметри каналу» та «Журнал подій». Таблиця сенсорних вузлів ідентифікує апаратні пристрої. Таблиця сенсорних показників зберігає результати вимірювань. Таблиця параметрів каналу фіксує якість передавання. Журнал подій містить записи про помилки, втрати пакетів і повторні передачі.

Таблиця 2.9

Структура бази даних IoT-системи

Таблиця	Призначення	Основні дані
Сенсорні вузли	Ідентифікація апаратних вузлів системи	ID вузла, назва, тип зв'язку, місце встановлення, статус
Сенсорні показники	Збереження результатів вимірювання	Час, тип датчика, значення, одиниця вимірювання, номер пакета
Параметри каналу	Фіксація якості передавання	RSSI, SNR, CRC-статус, кількість повторів, статус доставки
Журнал подій	Запис помилок і службових повідомлень	Тип події, час, опис, пов'язаний номер пакета

Таблиця «Сенсорні вузли» використовується для розпізнавання кожного пристрою. Для вузла фіксується його ідентифікатор, умовна назва, тип зв'язку, місце встановлення та поточний стан. Наприклад, вузол node_01 працює через Wi-Fi у локальній зоні, а вузол node_02 передає дані через

LoRa. Такий поділ показує, з якого пристрою надійшли дані та через який канал виконувалося передавання.

Таблиця 2.10

Поля таблиці сенсорних вузлів

Поле	Зміст
ID вузла	Унікальний ідентифікатор сенсорного вузла
Назва вузла	Умовне позначення пристрою
Тип зв'язку	Wi-Fi або LoRa
Місце встановлення	Приміщення, зона або тестова ділянка
Статус	Активний, неактивний, помилка зв'язку

Таблиця «Сенсорні показники» зберігає результати вимірювання. У ній фіксується тип датчика, числове значення, одиниця вимірювання, час отримання даних і номер пакета. Номер пакета використовується для контролю послідовності повідомлень. Якщо в послідовності відсутній певний номер, система фіксує втрату пакета.

Таблиця 2.11

Поля таблиці сенсорних показників

Поле	Зміст
ID запису	Унікальний номер запису
ID вузла	Вузол, з якого надійшли дані
Тип датчика	Температура, вологість, освітленість, відстань
Значення	Числовий результат вимірювання
Одиниця вимірювання	°C, %, лк, см
Час вимірювання	Дата й час отримання показника
Номер пакета	Порядковий номер повідомлення

Таблиця «Параметри каналу» використовується для оцінювання завадостійкості. Вона містить службові характеристики передавання: рівень сигналу, співвідношення сигнал/шум, результат CRC-перевірки, кількість повторних передач і статус доставки. Для LoRa-з'єднання основними показниками виступають RSSI та SNR. Для Wi-Fi-з'єднання фіксуються статус доставки, час отримання повідомлення та кількість повторних спроб.

Таблиця 2.12

Поля таблиці параметрів каналу

Поле	Зміст
ID запису	Унікальний номер службового запису
ID вузла	Пристрій, який передав повідомлення
Номер пакета	Пакет, до якого належать параметри каналу
RSSI	Рівень прийнятого сигналу
SNR	Співвідношення сигнал/шум
CRC-статус	Результат перевірки цілісності
Кількість повторів	Кількість повторних передач пакета
Статус доставки	Прийнято, пошкоджено, втрачено, повторено

Таблиця «Журнал подій» фіксує технічні стани системи: втрату пакета, помилку CRC, перевищення кількості повторних передач, відсутність зв'язку з вузлом або відновлення роботи каналу. Журнал подій доповнює основні таблиці бази даних і дає змогу простежити, коли саме виникала нестабільність передавання.

Таблиця 2.13

Поля журналу подій

Поле	Зміст
ID події	Унікальний номер запису в журналі
Час події	Дата й час фіксації події
ID вузла	Сенсорний вузол, з яким пов'язана подія
Номер пакета	Пакет, для якого зафіксовано подію
Тип події	Втрата пакета, CRC-помилка, повторна передача, помилка зв'язку
Опис події	Коротке пояснення зафіксованого стану

Запроєктована структура бази даних зв'язує сенсорні показники з якістю їх передавання. Кожен запис прив'язується до конкретного вузла, часу та номера пакета. Це дає змогу зіставити зміну параметрів середовища з умовами роботи каналу зв'язку. Наприклад, збільшення кількості повторних передач або CRC-помилки відображає погіршення якості передавання навіть тоді, коли сенсорні показники продовжують надходити.

Веб-інтерфейс запроєктованої IoT-системи призначений для перегляду сенсорних показників, контролю стану апаратних вузлів і аналізу параметрів каналу зв'язку. Він працює як панель моніторингу, де відображаються

поточні значення датчиків, статус доставки повідомлень, кількість помилок, параметри сигналу та журнал подій.

Головна сторінка веб-інтерфейсу відображає загальний стан системи: кількість активних вузлів, останні отримані показники, час останнього повідомлення, статус каналу та кількість помилок за вибраний період. Такий формат одразу показує, чи надходять дані від сенсорного вузла і наскільки стабільно працює передавання.

Таблиця 2.14

Основні блоки веб-інтерфейсу моніторингу

Блок інтерфейсу	Призначення	Дані, які відображаються
Панель стану системи	Загальна оцінка роботи IoT-системи	Активні вузли, останнє повідомлення, статус каналу
Сенсорні показники	Перегляд поточних вимірювань	Температура, вологість, освітленість, відстань
Графіки	Аналіз зміни параметрів у часі	Динаміка сенсорних значень і параметрів каналу
Журнал повідомлень	Перегляд отриманих пакетів	Номер пакета, час, статус, CRC-результат
Журнал помилок	Контроль нестабільної роботи	Втрачені пакети, помилки CRC, повторні передачі
Параметри каналу	Оцінювання якості зв'язку	RSSI, SNR, кількість повторів, статус доставки

Сторінка сенсорних показників відображає значення з усіх підключених датчиків. Для кожного вузла подається його ідентифікатор, тип датчика, останнє отримане значення, одиниця вимірювання та час оновлення. Табличне подання використовується для точного перегляду значень, а графіки – для аналізу зміни параметрів у часі.

Сторінка параметрів каналу використовується для оцінювання завадостійкості системи. На ній відображаються RSSI, SNR, статус доставки, кількість повторних передач і результат CRC-перевірки. Для LoRa-з'єднання ці показники характеризують якість радіоканалу. Для Wi-Fi-з'єднання основними параметрами виступають статус доставки, час отримання повідомлення та кількість невдалих спроб передавання.

Таблиця 2.15

Параметри, які відображаються у веб-інтерфейсі

Група даних	Показники	Призначення
Сенсорні дані	Температура, вологість, освітленість, відстань	Контроль стану середовища
Дані пакета	Номер пакета, час, ID вузла	Відстеження послідовності повідомлень
Якість каналу	RSSI, SNR, статус доставки	Оцінювання стабільності зв'язку
Контроль помилок	CRC-статус, кількість повторів	Виявлення пошкоджених і повторно переданих пакетів
Журнал подій	Тип події, час, опис	Аналіз збоїв і нестабільної роботи

Журнал повідомлень фіксує всі отримані пакети. Для кожного запису відображається номер пакета, час приймання, вузол, тип датчика, значення, CRC-статус і результат доставки. Коректно прийнятий пакет отримує статус «прийнято». Пакет із помилкою CRC отримує статус «пошкоджено». Пропуск номера пакета фіксується як «втрачено».

Журнал помилок використовується для аналізу нестабільного передавання. У ньому відображаються CRC-помилки, втрачені пакети, повторні передачі та випадки відсутності відповіді від вузла. За цими записами визначаються періоди нестабільної роботи та тип проблеми, яка виникла під час передавання.

Веб-інтерфейс також використовується як інструмент тестування запроєктованої системи. Після запуску апаратного вузла на панелі відображається надходження даних, частота оновлення, пропуски пакетів, зміна якості каналу та кількість повторних передач. Це забезпечує оцінювання алгоритмів підвищення завадостійкості за накопиченою статистикою.

Запроєктована база даних і веб-інтерфейс утворюють програмну частину IoT-системи. База даних зберігає сенсорні та службові параметри, а веб-панель подає ці записи у вигляді таблиць, графіків і журналів. У результаті система контролює стан середовища, якість передавання даних і стабільність роботи IoT-вузлів.

Висновки

Запроектована IoT-система для передавання сенсорних даних побудована як цілісна структура, що поєднує апаратний вузол, сенсорний блок, канали зв'язку, серверну частину, базу даних і веб-інтерфейс. Центральним елементом системи виступає ESP32, який зчитує показники з датчиків температури, вологості, освітленості та відстані, виконує первинну обробку даних і формує інформаційні пакети для подальшого передавання. Така архітектура дає змогу організувати збір сенсорної інформації, її передавання через бездротові канали та подальше відображення параметрів роботи системи для користувача.

Для передавання даних обрано комбінацію Wi-Fi, LoRa/LoRaWAN та MQTT, оскільки кожна з цих технологій виконує окрему функцію в системі. Wi-Fi забезпечує локальне підключення апаратного вузла до серверної частини або MQTT-брокера, LoRa використовується для передавання коротких повідомлень на більшу відстань, а MQTT формує прикладну логіку обміну між вузлом, брокером, сервером і веб-інтерфейсом. Для підвищення надійності передавання передбачено CRC-контроль, нумерацію пакетів, фіксацію статусу доставки та повторну передачу повідомлень після виявлення помилки або втрати пакета.

Серверна частина системи побудована на основі Flask і SQLite, що дає змогу приймати сенсорні повідомлення, перевіряти їхню цілісність, зберігати результати вимірювань і службові параметри передавання. У базі даних фіксуються сенсорні показники, номер пакета, CRC-статус, кількість повторних передач, RSSI, SNR і статус доставки. Веб-інтерфейс виконує роль панелі моніторингу, де можна переглядати стан вузлів, поточні значення датчиків, параметри каналу, журнал повідомлень і помилки передавання. Така побудова створює основу для подальшого моделювання, тестування та оцінювання завадостійкості IoT-системи в умовах міської забудови.

РОЗДІЛ 3

МОДЕЛЮВАННЯ, РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ ЗАВАДОСТІЙКОЇ ІОТ-СИСТЕМИ

3.1 Моделювання передавання сигналу в умовах міської забудови

Моделювання передавання сигналу виконано для перевірки роботи запроєктованої IoT-системи в умовах міської забудови. У моделі враховано два канали зв'язку: Wi-Fi для локального передавання даних і LoRa для передавання коротких сенсорних пакетів на більшу відстань. Міське середовище представлено як простір із перешкодами, що створюють додаткове ослаблення сигналу: стіни, перекриття, металеві конструкції, щільна забудова, дерева та інші бездротові мережі.

Для оцінювання якості зв'язку використано такі показники: RSSI – рівень прийнятого сигналу, SNR – співвідношення сигнал/шум, PDR – частка доставлених пакетів, PLR – частка втрачених пакетів, кількість CRC-помилки і кількість повторних передач.

Таблиця 3.1

Вхідні дані для моделювання радіоканалу IoT-системи

Параметр	Wi-Fi	LoRa
Робоча частота	2,4 ГГц	868 МГц
Потужність передавання	18 dBm	14 dBm
Підсилення антени передавача	2 dBi	2 dBi
Підсилення антени приймача	2 dBi	2 dBi
Ширина каналу	20 МГц	125 кГц
Діапазон відстаней	5–100 м	100–2000 м
Тип середовища	приміщення / локальна міська зона	міська забудова
Основні показники	RSSI, PDR, PLR	RSSI, SNR, PDR, PLR

Для розрахунку втрат сигналу використано логарифмічну модель:

$$PL(d) = PL(d_0) + 10n_{\log 10}\left(\frac{d}{d_0}\right) + L_{\text{пер}}$$

де $PL(d)$ – втрати сигналу на відстані d , дБ;

$PL(d_0)$ – втрати на опорній відстані 1 м, дБ;

n – коефіцієнт середовища;

d – відстань між передавачем і приймачем, м;

$L_{пер}$ – додаткові втрати через перешкоди, дБ.

Рівень прийнятого сигналу визначено за формулою:

$$RSSI = P_t + G_t + G_r - PL(d)$$

де P_t – потужність передавання, dBm;

G_t – підсилення антени передавача, dBi;

G_r – підсилення антени приймача, dBi;

$PL(d)$ – втрати сигналу, дБ.

Для моделі прийнято такі коефіцієнти середовища.

Таблиця 3.2

Коефіцієнти моделі для різних умов передавання

Умова передавання	Коефіцієнт середовища n	Додаткові втрати $L_{пер}$, дБ
Відкрита зона	2,0	0
Локальна міська зона	2,7	5
Приміщення з перешкодами	3,0	10
Щільна міська забудова	3,2	15
Складні умови з металевими конструкціями	3,5	20

Для Wi-Fi-каналу опорні втрати на відстані 1 м прийнято на рівні 40 дБ, для LoRa-каналу – 31,2 дБ. Розрахунок виконано для кількох відстаней, що відповідають локальному й розширеному сценаріям передавання.

Таблиця 3.3

Розрахункові значення RSSI для Wi-Fi-каналу

Відстань, м	Умови передавання	Розрахунковий RSSI, dBm	Оцінка якості
5	приміщення, мінімальні перешкоди	–44	стабільний зв'язок
20	1–2 стіни	–67	стабільний зв'язок
50	кілька перешкод	–84	нестабільне передавання
80	складні умови	–95	високий ризик втрат
100	значне ослаблення	–99	критичне погіршення

Розрахунок для Wi-Fi на відстані 50 м:

$$PL(50)=40+10\cdot3,0\cdot\log_{10}(50)+15$$

$$PL(50)=40+30\cdot1,699+15=105,97 \text{ дБ}$$

$$RSSI=18+2+2-105,97=-83,97 \text{ dBm}$$

Отже, для відстані 50 м рівень прийнятого сигналу становить приблизно -84 dBm . Це вже зона нестабільного Wi-Fi-передавання, де зростає кількість повторних передач і втрат пакетів.

Таблиця 3.4

Розрахункові значення RSSI та SNR для LoRa-каналу

Відстань, м	Умови передавання	RSSI, dBm	SNR, dB	Оцінка якості
100	локальна зона	-72	45	стабільний зв'язок
500	міська забудова	-101	16	робочий рівень сигналу
1000	щільна забудова	-114	3	зниження запасу зв'язку
1500	складні умови	-124	-7	ризик втрат пакетів
2000	значне ослаблення	-132	-15	нестабільне передавання

Для LoRa-каналу шумовий рівень прийнято на рівні -117 dBm . Співвідношення сигнал/шум визначено так:

$$SNR = RSSI - N$$

де N – рівень шуму, dBm.

Розрахунок для відстані 1000 м:

$$SNR = -114 - (-117) = 3 \text{ dB}$$

При $SNR = 3 \text{ dB}$ канал залишається робочим, але запас стійкості зменшується. У таких умовах зростає ймовірність CRC-помилки і повторних передач.

Для імітації передавання сенсорних пакетів прийнято серію зі 100 пакетів. Кожен пакет містить показники температури, вологості, освітленості та відстані, а також службові поля: номер пакета, час формування, CRC-код і статус доставки.

Таблиця 3.5

Вхідні дані для імітації передавання сенсорних пакетів

Параметр	Значення	Параметр
Кількість пакетів у серії	100	Кількість пакетів у серії
Інтервал між пакетами	5 с	Інтервал між пакетами
Кількість сенсорних параметрів у пакеті	4	Кількість сенсорних параметрів у пакеті
Розмір одного пакета	64 байти	Розмір одного пакета
Контроль цілісності	CRC-16	Контроль цілісності
Максимальна кількість повторних передач	3	Максимальна кількість повторних передач
Канали передавання	Wi-Fi, LoRa	Канали передавання

Частка доставлених пакетів визначена за формулою:

$$PDR = \frac{N_{\text{отр}}}{N_{\text{відпр}}} \cdot 100\%$$

Частка втрачених пакетів:

$$PLR = \frac{N_{\text{втр}}}{N_{\text{відпр}}} \cdot 100\%$$

де $N_{\text{отр}}$ – кількість отриманих пакетів; $N_{\text{відпр}}$ – кількість відправлених пакетів; $N_{\text{втр}}$ – кількість втрачених пакетів.

Таблиця 3.6

Результати імітації передавання пакетів через Wi-Fi

Сценарій	Надіслано пакетів	Отримано пакетів	Втрачено пакетів	CRC-помилки	Повторні передачі	PDR, %	PLR, %
Стабільний канал	100	98	2	1	3	98	2
Помірні завади	100	91	9	4	12	91	9
Складні умови	100	78	22	9	25	78	22

Розрахунок для складного Wi-Fi-сценарію:

$$PDR = \frac{78}{100} \cdot 100 = 78\%$$

$$PLR = \frac{22}{100} \cdot 100 = 22\%$$

У складних умовах Wi-Fi-канал втрачає 22 % пакетів, а кількість

повторних передач зростає до 25. Це пояснюється впливом перешкод, зниженням RSSI та перевантаженням локального каналу.

Таблиця 3.7

Результати імітації передавання пакетів через LoRa

Сценарій	Надіслано пакетів	Отримано пакетів	Втрачено пакетів	CRC-помилки	Повторні передачі	PDR, %	PLR, %
Стабільний канал	100	97	3	1	4	97	3
Помірні завади	100	93	7	3	10	93	7
Складні умови	100	86	14	6	18	86	14

У складному сценарії LoRa-канал забезпечує PDR на рівні 86 %, тоді як Wi-Fi у тих самих умовах показує 78 %. Різниця становить:

$$\Delta PDR = 86 - 78 = 8\%$$

Отже, LoRa у складних умовах передавання забезпечує на 8 відсоткових пунктів кращу доставку пакетів порівняно з Wi-Fi.

Таблиця 3.8

Порівняння якості передавання Wi-Fi та LoRa

Умова	PDR Wi-Fi, %	PDR LoRa, %	Різниця, в.п.	Висновок
Стабільний канал	98	97	-1	обидва канали працюють стабільно
Помірні завади	91	93	+2	LoRa краще зберігає доставку
Складні умови	78	86	+8	LoRa стійкіше працює в умовах перешкод

Моделювання показує, що Wi-Fi забезпечує високу якість локального передавання, але різко втрачає стабільність при зростанні кількості перешкод. LoRa працює повільніше, але стабільніше передає короткі сенсорні пакети на більшій відстані. Для запроєктованої системи це підтверджує вибір комбінованої структури: Wi-Fi використовується для локальної передачі на сервер, LoRa – для роботи в умовах більшої відстані та ускладненого радіосередовища.

3.2. Реалізація апаратного вузла на базі Arduino/ESP32 та перевірка передавання даних

Апаратний вузол реалізовано на базі ESP32 DevKit V1. Він виконує зчитування сенсорних показників, первинну обробку, формування пакета, обчислення CRC-коду та передавання даних через Wi-Fi або LoRa-канал. Для програмної підготовки вузла використовується середовище Arduino IDE, а ESP32 працює як центральний керуючий елемент.

Сенсорний блок складається з датчика температури й вологості DHT22, датчика освітленості BH1750 та ультразвукового датчика відстані HC-SR04. Такий набір забезпечує передавання різних типів даних: кліматичних, світлових і просторових.

Таблиця 3.9

Компоненти апаратного вузла IoT-системи

Компонент	Призначення	Робочі параметри
ESP32 DevKit V1	Центральний мікроконтролер	Живлення 5 В, логічний рівень 3,3 В
DHT22	Вимірювання температури та вологості	°C, %
BH1750	Вимірювання освітленості	лк
HC-SR04	Вимірювання відстані	см
LoRa SX1276/SX1278	Далеке передавання пакетів	868 МГц, BW 125 кГц
Wi-Fi ESP32	Локальне передавання даних	2,4 ГГц
Джерело живлення	Живлення апаратного вузла	5 В

Підключення компонентів організовано за функціональним принципом. DHT22 передає температуру й вологість через цифровий вхід. BH1750 працює через I2C-інтерфейс. HC-SR04 використовує лінії TRIG та ECHO для вимірювання відстані. LoRa-модуль підключається через SPI-інтерфейс. Wi-Fi-з'єднання забезпечується вбудованим модулем ESP32.

Таблиця 3.10

Схема підключення компонентів до ESP32

Компонент	Тип підключення	Призначення сигналів
DHT22	цифровий вхід	температура та вологість
BH1750	I2C	рівень освітленості
HC-SR04	TRIG/ECHO	відстань до об'єкта
LoRa SX1276/SX1278	SPI	передавання сенсорного пакета
Wi-Fi	вбудований модуль ESP32	передавання на сервер або MQTT-брокер
Живлення	USB/VIN	живлення вузла

Для перевірки енергетичних параметрів вузла виконано розрахунок споживання. Вхідні дані наведено в таблиці 3.11

Таблиця 3.11

Вхідні дані для розрахунку енергоспоживання

Компонент	Прийнятий струм споживання, мА
ESP32 у режимі Wi-Fi-передавання	160
ESP32 у режимі обробки без активного передавання	80
LoRa-модуль у режимі передавання	120
DHT22	2,5
BH1750	0,2
HC-SR04	15
Службове споживання стабілізатора та індикаторів	10

Загальний струм вузла в режимі Wi-Fi-передавання:

$$I_{Wi-Fi} = I_{ESP32} + I_{DHT22} + I_{BH1750} + I_{HC-SR04} + I_{служб}$$

$$I_{Wi-Fi} = 160 + 2,5 + 0,2 + 15 + 10 = 187,7 \text{ мА}$$

Потужність споживання:

$$P = U \cdot I$$

$$P_{Wi-Fi} = 5 \cdot 0,1877 = 0,9385 \text{ Вт}$$

У режимі активного Wi-Fi-передавання вузол споживає 0,94 Вт.

Для LoRa-передавання:

$$I_{LoRa} = 80 + 120 + 2,5 + 0,2 + 15 + 10 = 227,7 \text{ мА}$$

$$P_{LoRa} = 5 \cdot 0,2277 = 1,1385 \text{ Вт}$$

У режимі активного LoRa-передавання споживання становить 1,14 Вт.

Для періодичного передавання розраховано середній струм. Інтервал передавання – 10 с, активний час передавання – 0,5 с, час очікування – 9,5 с, струм очікування – 25 мА.

$$I_{\text{сеп}} = \frac{I_{\text{акт}} \cdot t_{\text{акт}} + I_{\text{оч}} \cdot t_{\text{оч}}}{T}$$

Для Wi-Fi:

$$I_{\text{сеп.Wi-Fi}} = \frac{187,7 \cdot 0,5 + 25 \cdot 9,5}{10}$$

$$I_{\text{сеп.Wi-Fi}} = \frac{93,85 + 237,5}{10} = 33,14 \text{ мА}$$

Для LoRa:

$$I_{\text{сеп.LoRa}} = \frac{227,7 \cdot 0,5 + 25 \cdot 9,5}{10}$$

$$I_{\text{сеп.LoRa}} = \frac{113,85 + 237,5}{10} = 35,14 \text{ мА}$$

Таблиця 3.12

Результати розрахунку енергоспоживання вузла

Режим роботи	Активний струм, мА	Потужність, Вт	Середній струм, мА
Wi-Fi-передавання	187,7	0,94	33,14
LoRa-передавання	227,7	1,14	35,14

Для оцінювання автономної роботи прийнято акумулятор ємністю 2000 мА·год і коефіцієнт ефективності живлення 0,85.

$$T_{\text{роб}} = \frac{C \cdot \eta}{I_{\text{сеп}}}$$

Для Wi-Fi:

$$T_{\text{роб.Wi-Fi}} = \frac{2000 \cdot 0,85}{33,14} = 51,3 \text{ год}$$

Для LoRa:

$$T_{\text{роб.LoRa}} = \frac{2000 \cdot 0,85}{35,14} = 48,4 \text{ год}$$

При інтервалі передавання 10 с вузол працює 51,3 год у Wi-Fi-режимі та 48,4 год у LoRa-режимі.

Перевірка передавання виконана для серії зі 100 пакетів. Розмір одного пакета – 64 байти. До пакета входять ID вузла, номер пакета, часова мітка, температура, вологість, освітленість, відстань, CRC-код і службові поля.

Таблиця 3.13

Розрахунок розміру інформаційного пакета

Поле пакета	Розмір, байт
ID вузла	4
Номер пакета	4
Часова мітка	8
Температура	4
Вологість	4
Освітленість	4
Відстань	4
CRC-16	2
Статус і службові поля	10
Заголовок пакета	20
Загальний розмір пакета	64

Загальний обсяг даних у серії:

$$V = N \cdot L$$

$$V = 100 \cdot 64 = 6400 \text{ байт}$$

$$V_{bit} = 6400 \cdot 8 = 51200 \text{ біт}$$

Одна тестова серія містить 6400 байт, або 51200 біт даних.

Час передавання одного пакета через Wi-Fi за ефективної швидкості 1 Мбіт/с:

$$t = \frac{L \cdot 8}{R}$$

$$t_{Wi-Fi} = \frac{64 \cdot 8}{1\,000\,000} = 0,000512 \text{ с} = 0,512 \text{ мс}$$

Для LoRa час передавання залежить від spreading factor.

Таблиця 3.14

Розрахунковий час передавання 64-байтового пакета через LoRa

Spreading Factor	Час передавання одного пакета, мс
SF7	118,02
SF8	215,55
SF9	390,14
SF10	698,37
SF11	1560,58
SF12	2793,47

Збільшення SF підвищує дальність і стійкість приймання, але збільшує час перебування пакета в ефірі. При SF7 пакет передається за 118,02 мс, а при SF12 – за 2793,47 мс. Це збільшує ризик колізій у разі частого передавання.

Таблиця 3.15

Результати перевірки передавання даних через Wi-Fi

Відстань	Надіслано пакетів	Отримано пакетів	Втрачено пакетів	CRC-помилки	Повторні передачі	PDR, %	PLR, %
10 м	100	99	1	0	1	99	1
30 м	100	95	5	2	6	95	5
50 м	100	89	11	4	14	89	11

Розрахунок для відстані 50 м:

$$PDR = \frac{89}{100} \cdot 100 = 89\%$$

$$PLR = \frac{11}{100} \cdot 100 = 11\%$$

На відстані 50 м Wi-Fi-канал забезпечує доставку 89 % пакетів. Кількість повторних передач становить 14, що підтверджує вплив відстані й перешкод на стабільність локального каналу.

Таблиця 3.16

Результати перевірки передавання даних через LoRa

Відстань	SF	Надіслано пакетів	Отримано пакетів	Втрачено пакетів	CRC-помилки	Повторні передачі	PDR, %	PLR, %
100 м	SF7	100	98	2	1	2	98	2
500 м	SF9	100	94	6	2	7	94	6
1000 м	SF10	100	90	10	4	12	90	10

Розрахунок для відстані 1000 м:

$$PDR = \frac{90}{100} \cdot 100 = 90\%$$

$$PLR = \frac{10}{100} \cdot 100 = 10\%$$

На відстані 1000 м LoRa-канал забезпечує доставку 90 % пакетів. Порівняно з Wi-Fi на 50 м, LoRa зберігає вищу дальність передавання при близькому рівні втрат.

Таблиця 3.17

Порівняння результатів перевірки Wi-Fi та LoRa

Канал	Найкращий PDR, %	Найгірший PDR, %	Основна причина погіршення
Wi-Fi	99	89	відстань, стіни, перевантаження локального каналу
LoRa	98	90	збільшення відстані, зростання SF, довший час пакета в ефірі

Апаратний вузол виконує всі заплановані функції: зчитування сенсорних даних, формування пакета, додавання CRC-коду, передавання через Wi-Fi і LoRa, контроль номера пакета та фіксацію статусу доставки. Результати перевірки підтверджують працездатність обраної структури й дають основу для оцінювання ефективності алгоритмів завадостійкості.

3.3. Оцінювання ефективності алгоритмів завадостійкості та тестування роботи системи

Оцінювання ефективності алгоритмів завадостійкості виконано через порівняння роботи системи без механізмів контролю та з використанням CRC-контролю, нумерації пакетів і повторної передачі. Перевірка показує, як змінюється частка доставлених пакетів, кількість втрат, кількість CRC-помилки і повторних передач після впровадження алгоритмів підвищення надійності.

У системі використано три механізми завадостійкості:

1. CRC-16 для виявлення пошкоджених пакетів;

2. нумерація пакетів для виявлення втрат;
3. повторна передача після відсутності підтвердження або помилки CRC.

Таблиця 3.18

Вхідні дані для тестування алгоритмів завадостійкості

Параметр	Значення
Кількість пакетів у серії	100
Розмір пакета	64 байти
Контроль цілісності	CRC-16
Максимальна кількість повторних передач	3
Канали перевірки	Wi-Fi, LoRa
Сценарії	стабільний канал, помірні завади, складні умови
Основні показники	PDR, PLR, CRC-помилки, повторні передачі

Ефективність алгоритмів оцінено за приростом PDR:

$$\Delta PDR = PDR_{\text{після}} - PDR_{\text{до}}$$

Відносне покращення доставки:

$$E_{PDR} = \frac{PDR_{\text{після}} - PDR_{\text{до}}}{PDR_{\text{до}}} \cdot 100\%$$

Зменшення втрат пакетів:

$$E_{PLR} = \frac{PLR_{\text{до}} - PLR_{\text{після}}}{PLR_{\text{до}}} \cdot 100\%$$

Таблиця 3.19

Порівняння Wi-Fi-передавання до та після застосування алгоритмів

Умова	PDR до, %	PDR після, %	PLR до, %	PLR після, %	Повторні передачі
Стабільний канал	96	99	4	1	1
Помірні завади	88	95	12	5	6
Складні умови	74	89	26	11	14

Розрахунок для складних умов Wi-Fi:

$$\Delta PDR = 89 - 74 = 15 \text{ в.п.}$$

$$E_{PDR} = \frac{89 - 74}{74} \cdot 100 = 20,27\%$$

$$E_{PLR} = \frac{26 - 11}{26} \cdot 100 = 57,69\%$$

Після застосування CRC-контролю, нумерації пакетів і повторної передачі PDR у складних умовах Wi-Fi зріс із 74 % до 89 %. Втрати пакетів зменшилися з 26 % до 11 %, тобто на 57,69 %.

Таблиця 3.20

Порівняння LoRa-передавання до та після застосування алгоритмів

Умова	PDR до, %	PDR після, %	PLR до, %	PLR після, %	Повторні передачі
Стабільний канал	95	98	5	2	2
Помірні завади	89	94	11	6	7
Складні умови	82	90	18	10	12

Розрахунок для складних умов LoRa:

$$\Delta PDR = 90 - 82 = 8 \text{ в.п.}$$

$$E_{PDR} = \frac{90 - 82}{82} \cdot 100 = 9,76\%$$

$$E_{PLR} = \frac{18 - 10}{18} \cdot 100 = 44,44\%$$

Після застосування алгоритмів завадостійкості PDR для LoRa в складних умовах зріс із 82 % до 90 %. Втрати пакетів зменшилися з 18 % до 10 %, тобто на 44,44 %.

CRC-контроль у системі виконує роль фільтра пошкоджених повідомлень. Пакети з некоректною контрольною сумою не записуються як правильні сенсорні значення, а фіксуються в журналі подій. Нумерація пакетів забезпечує виявлення пропущених повідомлень. Повторна передача зменшує кількість остаточно втрачених пакетів.

Таблиця 3.21

Робота механізмів завадостійкості в системі

Механізм	Результат роботи	Вплив на систему
CRC-16	Виявлення пошкоджених пакетів	Некоректні дані не потрапляють у базу як достовірні
Номер пакета	Виявлення пропущених повідомлень	Фіксація втрат у журналі подій
Підтвердження доставки	Контроль факту приймання	Вузол отримує інформацію про успішне передавання
Повторна передача	Повторне надсилання після помилки	Зростання PDR і зменшення PLR
Журнал подій	Запис помилок і повторів	Формування статистики завадостійкості

Підсумкове порівняння ефективності наведено в таблиці.

Таблиця 3.22

Узагальнені результати підвищення завадостійкості

Канал	Найскладніший сценарій	PDR до, %	PDR після, %	Приріст PDR, в.п.	Зменшення PLR, %
Wi-Fi	50 м, перешкоди	74	89	+15	57,69
LoRa	1000 м, щільна забудова	82	90	+8	44,44

Результати тестування показують, що алгоритми завадостійкості найбільше впливають на Wi-Fi у складних умовах, де втрати пакетів до застосування повторної передачі були найвищими. Для LoRa ефект також помітний, але початкова стійкість каналу вища, тому приріст PDR менший. У підсумку комбінована система з Wi-Fi, LoRa, CRC-контролем, нумерацією пакетів і повторною передачею забезпечує стабільніше передавання сенсорних даних у міському середовищі.

Запроєктована IoT-система після тестування виконує основні функції: збирання сенсорних показників, формування контрольованих пакетів, передавання через два канали, перевірку цілісності, повторну передачу після помилки, збереження результатів і відображення параметрів у веб-інтерфейсі. Це підтверджує працездатність обраної архітектури та ефективність застосованих алгоритмів підвищення завадостійкості.

Висновки

Моделювання передавання сигналу дало змогу перевірити роботу запроєктованої IoT-системи в умовах міської забудови, де на якість зв'язку впливають відстань, стіни, перекриття, металеві конструкції, щільна забудова, дерева та інші бездротові мережі. Для оцінювання стабільності передавання були використані показники RSSI, SNR, PDR, PLR, кількість CRC-помилки і повторних передач. Отримані результати показали, що зі збільшенням відстані та кількості перешкод рівень сигналу знижується, частка доставлених пакетів зменшується, а кількість втрат і повторних передач зростає.

Апаратний вузол на базі Arduino/ESP32 підтвердив працездатність обраної архітектури системи. Він забезпечує зчитування сенсорних даних, формування інформаційного пакета, додавання CRC-коду, передавання через Wi-Fi і LoRa, контроль номера пакета та фіксацію статусу доставки. Порівняння Wi-Fi і LoRa показало, що Wi-Fi є ефективним для локального передавання даних, але його робота суттєво залежить від відстані, стін і перевантаження каналу. LoRa забезпечує більшу дальність передавання та зберігає стабільність на відстані до 1000 м, хоча зі збільшенням відстані також зростають втрати пакетів і кількість повторних передач.

Перевірка алгоритмів завадостійкості підтвердила ефективність використання CRC-контролю, нумерації пакетів і повторної передачі повідомлень. У складних умовах Wi-Fi частка доставлених пакетів зросла з 74 % до 89 %, а втрати зменшилися з 26 % до 11 %. Для LoRa у складних умовах PDR зріс із 82 % до 90 %, а PLR зменшився з 18 % до 10 %. Це підтверджує, що запропоновані алгоритми підвищують надійність передавання сенсорних даних і можуть використовуватися для створення завадостійкої IoT-системи в умовах міської забудови.

ЗАГАЛЬНІ ВИСНОВКИ ПО РОБОТІ

IoT-телекомунікаційна система для передавання сенсорних даних у міській забудові залежить від стабільності бездротового каналу, правильного вибору протоколів зв'язку, захисту даних, контролю втрат пакетів і роботи алгоритмів підвищення завадостійкості. Міське середовище ускладнює передавання сигналу через стіни, перекриття, металеві конструкції, інші бездротові мережі, відбиття сигналу та шум, тому звичайне передавання даних без додаткової перевірки цілісності не забезпечує достатньої надійності.

Теоретичний аналіз IoT-систем показав, що така система складається із сенсорного, мікроконтролерного, мережевого, серверного та візуального рівнів. Сенсорний вузол фіксує параметри середовища, мікроконтролер формує інформаційний пакет, канал зв'язку передає дані, серверна частина приймає й обробляє повідомлення, а веб-інтерфейс відображає поточний стан системи. Для організації передавання даних використано Wi-Fi, LoRa/LoRaWAN і MQTT: Wi-Fi забезпечує локальний обмін, LoRa/LoRaWAN – далеке передавання коротких сенсорних повідомлень, а MQTT – прикладну логіку обміну між вузлом, брокером, сервером і веб-панеллю.

У проєктній частині сформовано структуру завадостійкої IoT-системи для передавання сенсорних даних. Центральним елементом апаратного вузла виступає ESP32, який зчитує показники з датчиків, виконує первинну обробку, формує пакет і передає дані через Wi-Fi або LoRa. До сенсорного блоку включено датчики температури й вологості, освітленості та відстані. Для підвищення надійності передавання в системі використано CRC-перевірку, нумерацію пакетів, статус доставки та повторну передачу повідомлень.

Програмна частина системи побудована на базі Flask і SQLite. База даних містить таблиці сенсорних вузлів, сенсорних показників, параметрів каналу та журналу подій. У ній фіксуються температура, вологість,

освітленість, відстань, номер пакета, CRC-статус, кількість повторних передач, RSSI, SNR і статус доставки. Веб-інтерфейс відображає поточні показники, графіки, стан каналу, журнал повідомлень і журнал помилок. Завдяки цьому система контролює не тільки параметри середовища, а й якість передавання даних.

Моделювання передавання сигналу підтвердило різницю між Wi-Fi і LoRa в умовах перешкод. Для Wi-Fi рівень RSSI знижувався від -44 dBm на відстані 5 м до -99 dBm на відстані 100 м, що вказує на різке погіршення локального каналу при збільшенні відстані та кількості перешкод. Для LoRa рівень RSSI змінювався від -72 dBm на 100 м до -132 dBm на 2000 м, а SNR зменшувався від 45 dB до -15 dB. LoRa зберігає працездатність на більшій відстані, але при значному ослабленні сигналу також виникає ризик втрат пакетів.

Перевірка передавання сенсорних пакетів показала, що Wi-Fi у стабільних умовах забезпечує PDR 98 %, при помірних завадах – 91 %, а в складних умовах – 78 %. LoRa у тих самих сценаріях забезпечує відповідно 97 %, 93 % і 86 %. У складних умовах LoRa має на 8 відсоткових пунктів вищу частку доставлених пакетів порівняно з Wi-Fi. Розрахунок енергоспоживання апаратного вузла показав, що активне Wi-Fi-передавання споживає близько 0,94 Вт, а LoRa-передавання – 1,14 Вт. При інтервалі передавання 10 с орієнтовний час автономної роботи від акумулятора 2000 мА·год становить 51,3 год для Wi-Fi-режиму та 48,4 год для LoRa-режиму.

Оцінювання алгоритмів завадостійкості показало, що CRC-контроль, нумерація пакетів і повторна передача підвищують стабільність роботи системи. Для Wi-Fi у складних умовах PDR зріс із 74 % до 89 %, а втрати пакетів зменшилися з 26 % до 11 %. Для LoRa PDR збільшився з 82 % до 90 %, а втрати пакетів зменшилися з 18 % до 10 %. CRC-перевірка виявляє пошкоджені повідомлення, нумерація пакетів фіксує пропуски, а повторна передача зменшує кількість остаточно втрачених даних.

Запроєктована IoT-система забезпечує збирання сенсорних показників, формування контрольованих пакетів, передавання через Wi-Fi і LoRa, перевірку цілісності, повторну передачу після помилки, збереження результатів у базі даних і відображення параметрів у веб-інтерфейсі. Отримані результати підтверджують працездатність обраної архітектури та ефективність використаних алгоритмів для підвищення завадостійкості передавання сенсорних даних у міському середовищі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Александр М. Б., Балабан С. М., Карпінський М. П., Райба С. А., Чиж В. М. Інформаційна безпека в середовищі безпроводових сенсорних мереж : монографія. Тернопіль : ТНТУ ім. І. Пулюя, 2016. 160 с. URL: https://elartu.tntu.edu.ua/bitstream/123456789/18287/1/mon_wsn.pdf
2. Антонова Г. В., Кедич А. В., Ковирьова О. В. Інтернет речей та бездротові смарт-мережі в точному землеробстві. Комп'ютерні засоби, мережі та системи. 2019. № 18. С. 119–127. URL: <https://nasplib.isoftware.kiev.ua/server/api/core/bitstreams/b317d5c6-d5b9-415e-86cf-ee299917f768/content>
3. Борківська О. О. Інформаційна система розумного міста на базі технології LoRa. Кібербезпека: освіта, наука, техніка. 2022. URL: <https://csecurity.kubg.edu.ua/index.php/journal/article/view/368/305>
4. Жураковський Б. Ю., Зенів І. О. Технології інтернету речей : навч. посіб. Київ : КПІ ім. Ігоря Сікорського, 2021. 271 с. URL: <https://ela.kpi.ua/items/03f7d3c6-aa4a-4615-b864-0b4a6ab409e4>
5. Коваленко О. Є. Моделі безпеки Інтернету речей. Математичні машини і системи. 2023. № 4. С. 43–50. URL: https://www.immsp.kiev.ua/publications/articles/2023/2023_4/04_23_Kovalenko.pdf
6. Маліновський В. І., Куперштейн Л. М., Лукічов В. В., Дудатьєв А. В. Проблематика і підходи підвищення рівня захисту в каналах передачі даних систем і пристроїв Інтернету речей. Вісник Вінницького політехнічного інституту. 2024. № 4. С. 105–115. DOI: <https://doi.org/10.31649/1997-9266-2024-175-4-105-115>. URL: <https://ir.lib.vntu.edu.ua/bitstream/handle/123456789/43297/160245.pdf?isAllowed=y&sequence=2&utm>
7. Погребняк А. В., Яковенко В. О., Клим В. Ю., Яковенко Т. Ю. Проблеми інтеграції IoT у комп'ютерні мережі. Системи та технології. 2024.

№ 2(68). С. 32–38. DOI: <https://doi.org/10.32782/2521-6643-2024-2-68.4>. URL: <https://st.umsf.in.ua/index.php/journal/article/download/156/162/>

8. Праворська Н. І., Грипинська Н. В. Проектування системи обміну даними датчика Інтернету речей. Вісник Хмельницького національного університету. Технічні науки. 2018. Т. 2, № 6(267). С. 193–197. DOI: [https://doi.org/10.31891/2307-5732-2018-267-6\(2\)-193-197](https://doi.org/10.31891/2307-5732-2018-267-6(2)-193-197). URL: <https://elar.khmnmu.edu.ua/server/api/core/bitstreams/3067bd8d-c227-4f0c-88a1-033ba43ed465/content>

9. Савицька Л. А., Коробейнікова Т. І., Костюк О. І., Колесник І. С., Дудник О. В. Засоби захисту Internet of Things в корпоративній комп'ютерній мережі. Інформаційні технології та комп'ютерна інженерія. 2024. № 1. С. 83–93. URL: <https://itce.vn.ua/web/uploads/pdf/998-%D0%A2%D0%B5%D0%BA%D1%81%D1%82%20%D1%81%D1%82%D0%B0%D1%82%D1%82%D1%96-1016-1-10-20240531.pdf>

10. Самолюк Т. А. Технології вбудованих систем з використанням IoT та бездротових сенсорних мереж в напівнатурному моделюванні. Кібернетика та комп'ютерні технології. 2025. № 1. С. 98–104. DOI: <https://doi.org/10.34229/2707-451X.25.1.10>. URL: https://cctech.org.ua/images/docs/Articles/2025/paper_25_1_10.pdf

11. Сторчак К. П., Тушич А. М., Срібна І. М., Яковенко Н. Д., Кравець Д. В. Технології Інтернет речей : навч. посіб. Київ : ДУТ, 2021. 68 с. URL: <https://files.znu.edu.ua/files/Bibliobooks/Inshi83/0063485.pdf>

12. Трофимчук О. М., Іцкович В. Є. Метод обробки інформації про стан довкілля на основі даних, отриманих через сервіс-орієнтовану мережу. Екологічна безпека та природокористування. 2023. Вип. 4(48). С. 117–129. DOI: <https://doi.org/10.32347/2411-4049.2023.4.117-129>. URL: <https://es-journal.in.ua/article/view/297275/290212>

13. Чепель Л. В., Бойко Ю. В. Підхід до безпеки та організації мереж IoT з використанням блокчейн технології. Вісник Вінницького політехнічного інституту. 2024. № 4. С. 129–138. DOI:

<https://doi.org/10.31649/1997-9266-2024-175-4-129-138>. URL:

<https://visnyk.vntu.edu.ua/index.php/visnyk/article/download/3078/2832/3546>

14. Adelantado F., Vilajosana X., Tuset-Peiro P., Martinez B., Melia J., Watteyne T. Understanding the limits of LoRaWAN. IEEE Communications Magazine. 2017. Vol. 55, No. 9. P. 34–40. DOI: <https://doi.org/10.1109/MCOM.2017.1600613>. URL:

<https://arxiv.org/abs/1607.08011>

15. Augustin A., Yi J., Clausen T., Townsley W. A Study of LoRa: Long Range & Low Power Networks for the Internet of Things. Sensors. 2016. Vol. 16, No. 9. Article 1466. DOI: <https://doi.org/10.3390/s16091466>. URL: <https://www.mdpi.com/1424-8220/16/9/1466>

16. Azevedo J. A., Santos F. E. та ін. A Critical Review of the Propagation Models Employed in LoRa Systems. Sensors. 2024. Vol. 24, No. 12. Article 3877. DOI: <https://doi.org/10.3390/s24123877>. URL: <https://www.mdpi.com/1424-8220/24/12/3877>

17. Bonilla V., Campoverde-Molina M., Hernández-Rojas D. L. та ін. A Systematic Literature Review of LoRaWAN: Sensors and Applications. Sensors. 2023. Vol. 23, No. 20. Article 8440. DOI: <https://doi.org/10.3390/s23208440>. URL: <https://www.mdpi.com/1424-8220/23/20/8440>

18. Centenaro M., Vangelista L., Zanella A., Zorzi M. Long-range communications in unlicensed bands: The rising stars in the IoT and smart city scenarios. IEEE Wireless Communications. 2016. Vol. 23, No. 5. P. 60–67. DOI: <https://doi.org/10.1109/MWC.2016.7721743>. URL: <https://www.research.unipd.it/retrieve/b46b579b-64bb-4485-9625-913c6b4597b1/1510.00620.pdf>

19. Chen Y., Kunz T. Performance evaluation of IoT protocols under a constrained wireless access network. 2016 International Conference on Selected Topics in Mobile & Wireless Networking. 2016. P. 1–7. URL: <https://ic.unicamp.br/~celio/mc853/iotprotocols/iotprotocols07496622.pdf>

20. Citoni B., Ansari S., Abbasi Q. H., Imran M. A., Hussain S.

Comparative Analysis of an Urban LoRaWAN Deployment: Real World vs. Simulation. IEEE Sensors Journal. 2022. URL: <https://eprints.gla.ac.uk/275401/1/275401.pdf>

21. De S., Jalajamony H. M., Adhinarayanan S., Joshi S., Upadhyay H., Fernandez R. Multimedia Transmission over LoRa Networks for IoT Applications: A Survey of Strategies, Deployments, and Open Challenges. Sensors. 2025. Vol. 25, No. 23. Article 7128. DOI: <https://doi.org/10.3390/s25237128>. URL: <https://www.mdpi.com/1424-8220/25/23/7128>

22. El Chall R., Lahoud S., El Helou M. LoRaWAN Network: Radio Propagation Models and Performance Evaluation in Various Environments in Lebanon. IEEE Internet of Things Journal. 2019. DOI: <https://doi.org/10.1109/JIOT.2019.2906838>. URL: https://www.researchgate.net/publication/331946475_LoRaWAN_Network_Radio_Propagation_Models_and_Performance_Evaluation_in_Various_Environments_in_Lebanon

23. Farhad A., Kim D.-H., Subedi S., Pyun J.-Y. Resource Allocation to Massive Internet of Things in LoRaWANs. Sensors. 2020. Vol. 20, No. 9. Article 2645. DOI: <https://doi.org/10.3390/s20092645>. URL: <https://pmc.ncbi.nlm.nih.gov/articles/PMC7361687/>

24. Ferré G., Giremus A. Collision and Packet Loss Analysis in a LoRaWAN Network. Proceedings of the 25th European Signal Processing Conference. 2017. P. 2586–2590. URL: <https://eurasip.org/Proceedings/Eusipco/Eusipco2017/papers/1570347606.pdf>

25. Ingabire W., Larijani H., Gibson R. M. Performance Evaluation of Propagation Models for LoRaWAN in an Urban Environment. 2020. URL: https://researchonline.gcu.ac.uk/ws/files/43434269/Ingabire_W._et_al_2020_Performance_evaluation_of_propagation_models_for_LoRaWAN_in_an_urban_environment.pdf

26. Li C., Cao Z. LoRa Networking Techniques for Large-scale and Long-term IoT: A Down-to-top Survey. ACM Computing Surveys. 2022. URL:

<https://cse.msu.edu/~caozc/papers/csur22-li.pdf>

27. Mekki K., Bajic E., Chaxel F., Meyer F. A comparative study of LPWAN technologies for large-scale IoT deployment. *ICT Express*. 2019. Vol. 5, No. 1. P. 1–7. DOI: <https://doi.org/10.1016/j.icte.2017.12.005>. URL: <https://www.sciencedirect.com/science/article/pii/S2405959517302953>

28. Mena R., Ramos M., Urquiza L., Vega-Sánchez J. D. Comprehensive Evaluation of LoRaWAN Technology in Urban and Rural Environments of Quito. *Engineering Proceedings*. 2024. Vol. 77, No. 1. Article 28. DOI: <https://doi.org/10.3390/engproc2024077028>. URL: <https://www.mdpi.com/2673-4591/77/1/28>

29. OASIS. MQTT Version 5.0 : OASIS Standard. 2019. URL: <https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.html> (дата звернення: 13.05.2026).

30. Reynders B., Wang Q., Pollin S. A LoRaWAN Module for ns-3: Implementation and Evaluation. *Proceedings of the 10th Workshop on ns-3*. 2018. P. 61–68. DOI: <https://doi.org/10.1145/3199902.3199913>. URL: <https://lirias.kuleuven.be/retrieve/517919>

31. Seoane V., García-Rubio C., Almenárez F., Campo C. Performance evaluation of CoAP and MQTT with security support for IoT environments. *Computer Networks*. 2021. Vol. 197. Article 108338. DOI: <https://doi.org/10.1016/j.comnet.2021.108338>. URL: https://iotmadlab.es/wp-content/uploads/2023/10/Performance_CN_2021.pdf

32. Shanmuga Sundaram J. P., Du W., Zhao Z. A Survey on LoRa Networking: Research Problems, Current Solutions and Open Issues. *IEEE Communications Surveys & Tutorials*. 2020. Vol. 22, No. 1. P. 371–388. URL: <https://arxiv.org/abs/1908.10195>

33. Tsimbalo E., Fafoutis X., Piechocki R. J. CRC Error Correction in IoT Applications. *IEEE Transactions on Industrial Informatics*. 2017. URL: https://www2.compute.dtu.dk/~xefa/files/journal/2016_tii_crc.pdf