1

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра обчислювальної техніки

**МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА**

на тему «**МЕТОДИ ТА ЗАСОБИ ОПРАЦЮВАННЯ ІНФОРМАЦІЇ ДЛЯ ІОТ**»

Виконав: студент 2 курсу, групи 1КІ-23мспеціальності 123 – Комп'ютерна інженерія
_____ Вівчарик Д. В.
Керівник к.т.н., доц.
_____Колесник І.С.
Опонент д.т.н. проф. каф.МБІС, директор Центру інформаційних технологій і захисту інформації, голова секції «Управління інформаційною безпекою»
_____ Яремчук Ю. Є.

Допущено до захисту
Завідувач кафедри ОТ
д.т.н., проф. Азаров О.Д.
«____» _____ 2024 р.

Вінниця 2024

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра обчислювальної техніки
Освітньо-кваліфікаційний рівень магістр
Спеціальність 123 Комп'ютерна інженерія

ЗАТВЕРДЖУЮ
Завідувач кафедри
обчислювальної техніки
проф., д.т.н. О. Д. Азаров
_____
«_18_» _вересня_ 2024 р.

# З А В Д А Н Н Я

## НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

Вівчарику Дмитру Валерійовичу

1 Тема роботи «Методи та засоби опрацювання даних для IoT», керівник роботи к.т.н., доц.затверджені наказом вищого навчального закладу від 17.09.2024 року №310.

2 Строк подання студентом роботи 9.12.2024 р.

3 Вихідні дані до роботи: мова програмування технічна документація , засоби та архітектура IoT, мережеві протоколи MQTT, XMPP.

4 Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити): вступ, огляд і аналіз аналогів технології IoT, аналіз моделей IoT, методи та засоби збору та обробки даних для IoT, тестування системи, висновки.

5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень):.

6 Консультантів розділів роботи представлено в таблиці 1.

3

Таблиця 1— Консультанти розділів роботи

| Розділ | Прізвище, ініціали та посада консультанта | Підпис, дата | |
|---|---|---|---|
| | | завдання видав | завдання прийняв |
| 1,2,3,4 | Колесник І.С., к.т.н., доц. каф. ОТ | | |
| 5 | Небава М.І., к.е.н., проф. каф. ЕП і ВМ | | |

7 Дата видачі завдання _18_._вересня_._2024_ р.

8 Календарний план наведено в таблиці 2.

Таблиця 2 — Календарний план

| №з/п | Назва етапів виконання магістерської роботи | Строк виконання етапів роботи | |
|---|---|---|---|
| 1 | Постановка мети та задач роботи | 21.10.24 | |
| 2 | Аналіз потенціалу та можливостей IoT | 25.10-30.10.24 | |
| 3 | Перспективи розвитку та існуючі проблеми IoT | 31.10-08.11.24 | |
| 4 | Огляд і аналіз аналогів технології IoT | 09.11-15.11.24 | |
| 5 | Аналіз моделей IoT | 16.11-.20.1124 | |
| 6 | Методи та засоби збору та обробки даних для IoT | 21.11-25.11.24 | |
| 7 | Організація інфраструктури IoT | 26.11-31.11.24 | |
| 8 | Тестування системи | 01.12-03.12.24 | |
| 9 | Розрахунок економічної частини роботи | 04.12-07.12.24 | |
| 10 | Оформлення пояснювальної записки та ілюстративного матеріалу | 08.12.24 | |
| 11 | Аналіз виконання роботи, висновки, додатки | | |
| 12 | Перевірка якості виконання магістерської роботи та усунення недоліків | | |

Студент_____Вівчарик Д.В.

Керівник роботи _____ Колесник І.С.

# АНОТАЦІЯ

УДК 004.4:373.3

Вівчарик Д.В.

Методи та засоби опрацювання даних для IoT. Магістерська кваліфікаційна робота зі спеціальності 123 — комп'ютерна інженерія, освітня програма — комп'ютерна інженерія. Вінниця: ВНТУ, 2024, 117 с.

На укр.мові. Бібліогр.: 29 назв, рис. 28, табл. 7.

Технологія Інтернету Речей (IoT) є однією з найбільш перспективних та швидко розвиваючих технологій, що застосовується в різних сферах, від побутових пристроїв до промислових систем. Однак з ростом обсягів і складності даних, які генеруються IoT-пристроями, виникають проблеми ефективної обробки, зберігання та передачі цих даних. В роботі досліджуються методи та засоби обробки інформації для IoT, а також інфраструктура для забезпечення надійності та продуктивності таких систем. Практична цінність роботи полягає у розробці нових підходів до організації інфраструктури IoT, що дозволяють підвищити ефективність збору, обробки та аналізу даних. Наукова новизна полягає у пропозиції удосконалених методів обробки даних і оптимізації протоколів для інтеграції пристроїв в єдину IoT-систему. Результати дослідження були апробовані на науковій конференції "Молодь в науці: дослідження, проблеми, перспективи (МН–2024)".

Ключовіслова:ІнтернетРечей,InternetofThings,архітектураIoT,MQTT,XMPP.

# ABSTRACT

Vivcharuk D.V.

Methods and tools for data processing for IoT.Master's qualification work in the specialty 123 — computer engineering, educational program computer engineering. Vinnitsa, VTNU, 2024, 117 p.

In the Ukr.leng. Libr.name 29, figure 28, table 7

The Internet of Things (IoT) technology is one of the most promising and rapidly developing technologies, applied in various fields ranging from household devices to industrial systems. However, with the growing volume and complexity of data generated by IoT devices, challenges arise in effectively processing, storing, and transmitting this data. This work investigates methods and tools for information processing in IoT, as well as the infrastructure required to ensure the reliability and performance of such systems. The practical value of the work lies in the development of new approaches to organizing IoT infrastructure, which improve the efficiency of data collection, processing, and analysis. The scientific novelty consists in the proposal of enhanced methods for data processing and optimization of protocols for integrating devices into a unified IoT system. The research results were presented at the scientific conference "Youth in Science: Research, Issues, Prospects (MN–2024)."

Keywords: Internet of Things, IoT architecture, MQTT, XMPP.

# ЗМІСТ



| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | | | 08-28.МКР.028.00.000 ПЗ | | |
| Змн. | Лист | № докум. | Підпис | Дата | | | |
| Розроб. | | Вівчарик Д. В. | | | Методи та засоби опрацювання даних для IoT Пояснювальна записка | Літ. | Арк. 6 / Акрушів 107 |
| Перевір. | | Колесник І. С. | | | | | |
| Реценз. | | Яремчук Ю. Є. | | | | ВНТУ, гр. 2КІ-23м | |
| Н. Контр. | | Швець С.І. | | | | | |
| Затверд. | | Азаров О.Д. | | | | | |

# ВСТУП

Інтернет речей (IoT) є однією з провідних технологій сучасності, яка знайшла застосування в різних сферах діяльності. Інтернет речей (IoT) виник на основі кількох попередніх технологічних напрямків, таких як сенсорні мережі, розподілені інформаційні системи та вбудовані обчислювальні системи. Термін «Система IoT» є точнішим, оскільки він підкреслює структуроване використання технології, а не просто її інтеграцію у глобальний Інтернет. Більшість IoT-пристроїв взаємодіють у рамках конкретних завдань, формуючи спеціалізовані мережі, а не загальнодоступні інтернет-структури.

На відміну від звичайних інформаційних систем, IoT розширює можливості обчислень завдяки інноваційним рішенням у фізичних пристроях. Яскравим прикладом є "розумні холодильники", які автоматизують процес ідентифікації вмісту для оптимізації побутових завдань. Сучасні концепції таких пристроїв включають сканування даних у реальному часі, усуваючи потребу у ручному введенні інформації, як це було у ранніх версіях програм.

Сенсорні мережі, що стали фундаментом IoT, охоплюють різноманітні конфігурації, призначені для збирання й передавання невеликих обсягів даних на централізовані сервери для обробки. Проте класичні сенсорні мережі обмежувалися лише збором даних, тоді як IoT дозволяє інтегрувати обробку інформації на рівні самої мережі.

Вбудовані обчислювальні системи, які використовуються в окремих пристроях або компактних мережах (наприклад, у транспортних системах), також внесли вагомий вклад у розвиток IoT. Основними сферами їх застосування стали споживча електроніка та кіберфізичні системи, де пріоритетними є точність, надійність та конкретно визначені завдання.

Існує кілька підходів до визначення IoT-систем, серед яких:

—фізичні пристрої, що працюють у мережі, але не завжди через Інтернет-протокол;

—сенсорні системи реального часу, що забезпечують динамічний збір

інформації;

—гнучкі та розвиваючі мережі вбудованих пристроїв, які адаптуються до змін умов роботи.

Основна відмінність IoT полягає в орієнтації на конкретні завдання та врахуванні динаміки фізичних процесів. Більшість IoT-систем складаються з датчиків, що фіксують дані, а також виконавчих елементів для впливу на навколишнє середовище. Ключова мета полягає у якісній обробці сигналів та даних із часовою структурою.

Подальший розвиток IoT став можливим завдяки мікро електромеханічним системам (MEMS), що забезпечили створення доступних та енергоефективних датчиків, таких як акселерометри, гіроскопи та хімічні сенсори. Їх поява відкрила нові можливості для застосування IoT-технологій, оскільки вони значно перевершують традиційні лабораторні та промислові рішення за доступністю й ефективністю. Сучасні сенсорні системи підштовхнули розвиток IoT у напрямку складних технологій обробки сигналів.

**Актуальність** теми "Методи та засоби опрацювання інформації для IoT" обумовлена швидким розвитком технологій Інтернету речей (IoT), які охоплюють величезну кількість сфер діяльності, від побутових пристроїв до промислових систем. IoT технології дозволяють з'єднувати різноманітні пристрої, збирати й обробляти величезні обсяги даних, що виникають від сенсорів, камер, розумних пристроїв і машин, що взаємодіють в реальному часі. Проте, з ростом обсягів і складності даних виникає проблема ефективної їх обробки, зберігання та передачі в рамках обмежених ресурсів (пам'ять, пропускна здатність, енергоспоживання).

У цьому контексті важливим є дослідження методів і засобів обробки інформації, що дозволяють забезпечити надійність, масштабованість і високопродуктивність IoT-систем. Особливо актуальним є застосування новітніх методів обробки даних, таких як штучний інтелект, машинне навчання та розподілені обчислення, для покращення ефективності аналізу даних, прогнозування та автоматизації процесів в розумних будинках, на

підприємствах, в медицині та інших галузях.

Таким чином, дослідження методів і засобів обробки інформації для IoT є важливим кроком до розвитку більш розумних, адаптивних та безпечних систем, що відповідають вимогам сучасного світу.

**Метою** є вдосконалення методу і засобів опрацювання інформації та розробка ефективної інфраструктури для IoT, ґрунтуючись на запропонованій архітектурі.

Для досягнення мети пропонується вирішити такі **завдання:**

— провести огляд і аналіз технологій IoT;

— оцінити сучасний розвиток IoT;

— проаналізувати наявні моделі IoT;

— розглянути методи збору та обробки даних в IoT ;

— оцінити підходи до збору та обробки даних та організації інфраструктури IoT;

— вдосконалити метод опрацювання інформації IoT з урахуванням запропонованої архітектури;

— спроектувати апаратно-програмний засіб опрацювання інформації IoT з урахуванням запропонованої архітектури;

— провести економічний розрахунок.

**Об’єктом** дослідження є метод та засоби опрацювання інформації для Інтернету Речей.

**Предметом** дослідження є протоколи збору та передачі даних у контексті Інтернету речей (IoT).

**Практична цінність** полягає у розробці нових методів організації інфраструктури Інтернету Речей (IoT), які можуть бути застосовані в ряді реальних сфер діяльності. Запропоновані підходи дозволяють підвищити ефективність збору, обробки та аналізу даних пристроїв IoT, що сприяє покращенню продуктивності та безпеки IoT-систем.

**Наукова новизна** полягає у впровадженні оригінальних підходів до

опрацювання інформації для Інтернету Речей (IoT), зокрема в аспектах збору, аналізу та обробки даних пристроїв IoT. У роботі запропоновано удосконалені методи, що дозволяють більш ефективно використовувати існуючі технології для інтеграції пристроїв в єдину IoT-систему. Особливу увагу приділено аналізу протоколів та моделей IoT, що можуть оптимізувати взаємодію пристроїв з навколишнім середовищем.

**Апробацію** результатів наукової роботи було проведено на науковій конференції «Молодь в науці: дослідження, проблеми, перспективи (МН–2024)», доповідь на тему “ Метод та засоби опрацювання інформації для IoT ”.

# 1 ОГЛЯД І АНАЛІЗТЕХНОЛОГІЇІНТЕРНЕТУРЕЧЕЙ

## 1.1 Сучаснийстанрозвитку IoT

Інтернет речей (IoT) став можливим завдяки значному здешевленню цифрової та аналогової електроніки, зокрема завдяки технологіям VLSI (Very Large Scale Integration). Важливо підкреслити, що IoT-вузли не залежать від передових технологічних процесів виробництва мікросхем. Натомість вони базуються на старіших та більш економічних виробничих лініях, які, попри обмежену кількість інтегрованих компонентів, здатні забезпечити достатню продуктивність для багатьох завдань у системах IoT.

Оскільки системи IoT часто функціонують в умовах обмеженого енергоспоживання, зменшення витрат енергії є критично важливим аспектом їх розробки. Ефективне споживання енергії напряму впливає на загальну вартість володіння системою та її життєвий цикл. Досягнення мінімальних рівнів енергоспоживання вимагає ретельного підходу до проектування як апаратного забезпечення, так і програмного коду, а також оптимізації алгоритмів управління.

Безпека є одним із найважливіших викликів для проєктування та впровадження IoT-систем. Поєднання фізичних і обчислювальних систем вимагає інтеграції методів захисту фізичних пристроїв та цифрової інфраструктури, що раніше розглядалися як окремі завдання. Надійний захист даних та пристроїв є обов’язковою умовою успішного впровадження IoT.

Очікується, що IoT створить глобальну мережу з мільярдів, а можливо й трильйонів пристроїв, які взаємодіють один з одним. IoT не є революційним проривом, а скоріше комплексним використанням існуючих технологій, які відкривають нові форми комунікації. Ця технологія поєднує фізичний та віртуальний світи, інтегруючи різноманітні концепції та інноваційні компоненти, включаючи мініатюризацію пристроїв, мобільний зв'язок, всепроникні мережі та створення нової екосистеми.

Серед ключових особливостей IoT можна виділити структурну

організацію додатків, послуг, мереж та кінцевих вузлів. Це дозволяє кардинально змінити підхід до збору, обробки та використання даних. IoT забезпечує вивчення складних процесів і взаємозв’язків завдяки симбіозу реального та цифрового світів. Фізичні об'єкти отримують цифрові аналоги та віртуальні представництва, завдяки чому пристрої можуть сприймати контекст, обмінюватися інформацією, реагувати на зміни та адаптувати свою роботу.

IoT відкриває нові можливості для бізнесу, створюючи послуги на основі актуальних даних з фізичного світу. Майже будь-який об'єкт, незалежно від того, чи він належить до фізичного або віртуального середовища, може бути підключений до IoT. При цьому важливо забезпечити доступність технології для всіх за низькою вартістю, а також впровадження механізмів захисту від шахрайства, зловмисних атак і порушень конфіденційності.

IoT можна розглядати як логічне продовження взаємодії між людиною та програмними системами, де новим елементом є "Речі", які забезпечують спілкування та інтеграцію. Процес впровадження IoT є масштабним технологічним викликом, який охоплює розробку міжгалузевих рішень та інноваційних платформ. На початкових етапах головний акцент робиться на управлінні додатками, що відповідають специфічним потребам окремих галузей.

Зокрема, прикладом ефективної інтеграції IoT є поєднання транспортних засобів із сенсорними мережами, системами глобального позиціонування (GPS) та технологіями радіозв'язку. Це дозволяє забезпечити такі послуги, як навігація, моніторинг, розваги та обмін даними в реальному часі. Зібрана інформація стає основою для оптимізації конструкції транспортних засобів, поліпшення процесів їх обслуговування та створення нових послуг протягом усього життєвого циклу. Постачальники технічних рішень, оператори зв'язку та кінцеві споживачі отримують можливість співпрацювати та обмінюватися даними, підвищуючи ефективність систем управління транспортом.

## 1.2 Аналіз потенціалу таможливостейIoT

У цьому підрозділі розглядаються програми Інтернету речей (IoT), які мають наступні функціональні можливості, які описані нижче.

Зондування місцезнаходження та обмін даними про позиці передбачає, що IoT-системи здатні збирати інформацію про місцезнаходження кінцевих пристроїв і вузлів, а також надавати послуги, що базуються на цих даних. Зібрана інформація може включати географічні координати, отримані через GPS, CellID, RFID та інші технології, а також абсолютні або відносні дані про взаємне положення об'єктів. Основні застосування IoT у цій сфері включають:

— відстеження рухомих активів: дана програма дозволяє контролювати місцезнаходження та стан товарів за допомогою пристроїв, оснащених датчиками позиціювання та функціями передачі даних, це забезпечує точний моніторинг і управління логістичними процесами;

— управління автопарком: системи управління автопарком дозволяють оптимізувати розклад і маршрути транспортних засобів та водіїв, орієнтуючись на бізнес-вимоги та дані про місцезнаходження автомобілів у реальному часі, це допомагає підвищити ефективність використання ресурсів і скоротити витрати;

— системи дорожнього моніторингу: завдяки збору інформації про трафік і виявленню перевантажених ділянок дороги за допомогою даних про місцезнаходження численних транспортних засобів, система надає водіям актуальні рекомендації, це дозволяє обирати оптимальні маршрути та уникати заторів, покращуючи швидкість і ефективність пересування.

Таким чином, застосування IoT у відстеженні місцезнаходження та обміні даними дозволяє значно підвищити продуктивність, забезпечити кращий контроль за активами та покращити навігаційні послуги.

Моніторинг навколишнього середовища —системи IoT здатні збирати та обробляти різноманітні фізичні та хімічні параметри довкілля за допомогою локально або масштабно розгорнутих сенсорних терміналів. До основних

екологічних показників відносяться температура, вологість, рівень шуму, видимість, інтенсивність освітлення, спектральний аналіз, рівень радіації, концентрація забруднювальних речовин (CO, $CO_2$ тощо), а також зображення та біометричні дані.

Основні напрямки застосування включають аналіз стану навколишнього середовища: IoT-системи дозволяють проводити моніторинг та аналіз природних середовищ, таких як ліси чи льодовики, а також контролювати загрози та природні катастрофи, наприклад, вулканічну активність або сейсмічні явища,крім того, технології застосовуються для промислового моніторингу, де параметри довкілля, зібрані численними сенсорами, допомагають автоматично активувати системи оповіщення та запобігання надзвичайним ситуаціям.

Дистанційний медичний моніторинг системи IoT використовуються для збору та аналізу даних про стан здоров'я пацієнтів за допомогою пристроїв, що носяться на тілі. Завдяки обробці цих даних можна відстежувати зміни ключових показників організму, аналізувати тенденції та надавати рекомендації щодо підтримання здоров’я або медичні консультації на основі отриманих результатів.

Таким чином, IoT забезпечує ефективний збір, обробку та використання інформації для аналізу довкілля та віддаленого моніторингу здоров'я, що значно підвищує рівень контролю та безпеки в обраних сферах застосування.

Дистанційне управління — системи IoT дозволяють здійснювати управління терміналами та виконувати необхідні функції, ґрунтуючись на командах додатків, зібраній інформації від об’єктів та вимогах до обслуговування.

Управління приладами, завдяки IoT-технологіям користувачі можуть дистанційно контролювати стан і роботу різноманітних пристроїв, що забезпечує зручність та підвищує ефективність їх використання.

Ліквідація наслідків катастроф, системи IoT дають змогу користувачам віддалено запускати обладнання для мінімізації наслідків надзвичайних

ситуацій. На основі даних моніторингу, отриманих від сенсорів, можна своєчасно активувати системи очищення, відновлення або аварійного реагування, що дозволяє знизити рівень збитків та забезпечити безпеку.

Таким чином, дистанційне управління через IoT забезпечує точне керування обладнанням і ефективне реагування на надзвичайні події, що є важливим кроком у розвитку автоматизованих систем.

Спеціалізована мережа передбачає, щосистема IoT повинна мати здатність швидко самоорганізовувати свою мережу та інтегруватися з мережевим або сервісним рівнем для надання супутніх послуг [1]. В транспортних мережах, для ефективної передачі даних, може бути швидко налаштована мережа, яка забезпечує взаємодію між транспортними засобами та/або дорожньою інфраструктурою. Це дозволяє забезпечити злагоджену роботу транспортних систем, знижуючи час на налаштування та оптимізуючи передачу інформації в реальному часі.

Безпечне спілкування —система IoT повинна забезпечувати створення захищених каналів передачі даних між додатками, щоб гарантувати конфіденційність і цілісність інформації. Це включає використання шифрування та інших методів захисту для забезпечення безпеки переданих даних та захисту від несанкціонованого доступу або зловмисних атак.

Системи IoT мають широке застосування в різноманітних сферах:

— промислові системи використовують датчики для моніторингу якості продукції та стану обладнання,наприклад, датчики, що встановлюються на електродвигунах, збирають дані, які дозволяють прогнозувати можливі несправності, що дає змогу вчасно проводити обслуговування та знижувати витрати на ремонти;

— розумні будівлі застосовують датчики для визначення місцезнаходження людей та стану будівлі,ці дані використовуються для оптимізації роботи систем опалення, вентиляції, кондиціонування та освітлення, що дозволяє знижувати енергоспоживання та витрати на експлуатацію,також датчики моніторять структурний стан будівлі,

забезпечуючи своєчасне виявлення проблем;

— розумні міста використовують датчики для контролю пішохідного та транспортного руху, а також інтегрують ці дані з іншими системами, такими як розумні будівлі, для поліпшення міської інфраструктури та забезпечення ефективного використання ресурсів;

— транспортні засоби оснащуються мережею датчиків для моніторингу стану автомобіля, що дозволяє поліпшити динаміку руху, зменшити витрату пального та знизити викиди шкідливих речовин в атмосферу;

— медичні системи інтегрують різноманітні датчики для відстеження стану пацієнтів, які можуть перебувати вдома, в автомобілях швидкої допомоги, в кабінетах лікарів або в лікарнях,це забезпечує безперервний моніторинг здоров'я та покращує ефективність медичних послуг.

Розглянувши існуючі програми IoT, можна визначити основні вимоги до таких систем:

— сенсорна мережа —система може діяти виключно як система збору даних, що отримуються від датчиків, які збирають необхідну інформацію про навколишнє середовище або стан об'єктів;

— система оповіщення —дані, зібрані датчиками, аналізуються, і сповіщення генеруються, коли виконуються певні умови або критерії що дозволяє своєчасно реагувати на зміни або аномалії;

— система аналізу — датчики збирають дані, які потім піддаються глибокому аналізу, а результати можуть бути представлені у вигляді звітів, які можуть оновлюватися періодично (щогодини, щодня тощо), або відображатися в реальному часі для швидкого прийняття рішень;

— реактивна система —в результаті аналізу даних датчиків може спрацьовувати реакція в системі, наприклад, включення або відключення пристроїв чи механізмів,ці системи не використовують класичні методи управління, а швидко реагують на події;

— система управління —дані з датчиків використовуються в алгоритмах управління для генерації відповідних команд для виконавчих механізмів,це

дозволяє автоматизувати процеси на основі зібраної інформації.

Окрім основних функціональних вимог, IoT системи також повинні задовольняти низку нефункціональних вимог, перерхованих нижче.

Затримка події —час, який проходить від захоплення події до її обробки, є критичним для деяких програм, особливо в реальному часі, де швидка реакція є важливою.

Пропускна здатність події —швидкість, з якою події можуть бути зафіксовані, передані та оброблені, залежить від пропускної здатності вузлів, мережі та хмари. Висока пропускна здатність необхідна для ефективної роботи системи.

Частота втрат подій та ємність буфера —якщо система не здатна обробити всі події через обмеження в її пропускній здатності, може відбутися втрата подій. Важливо мати можливість запобігти втратам або мати достатньо потужності для буферизації даних.

Час затримки та пропускна здатність —час затримки в обробці подій та пропускна здатність сервісів також є важливими аспектами для ефективної роботи IoT систем.

Надійність та доступність — оскільки IoT системи мають широкий розподіл, важливо забезпечити високу надійність частин мережі, навіть якщо вся система може бути більш вразливою. Доступність є ключовою для забезпечення стабільної роботи розподілених компонентів.

Термін служби — IoT системи часто мають довший термін служби порівняно з традиційними комп'ютерними системами. Тривалість експлуатації може значно перевищувати термін служби окремих компонентів, особливо коли використовуються надлишкові датчики та інші компоненти для підвищення надійності системи.

Ключовим аспектом IoT є дискретизація даних, яка може бути як керована подіями, так і аперіодична. Традиційні методи обробки та контролю цифрових сигналів передбачають періодичні вибірки, що призводить до створення часових рядів. Однак такі підходи можуть споживати значну

кількість енергії у вузлах і вимагати великої пропускної здатності мережі, що є недоцільним в умовах IoT. Не всі додатки потребують періодичних вибірок і можуть обробляти дані на основі аперіодичних зчитувань.

Обмеження по потужності та пропускній здатності стимулюють використання розподілених обчислень на основі подій, де відносно прості процесори можуть обробляти великі потоки даних. Розпізнавання важливих подій безпосередньо на пристрої, за допомогою обробки на краю мережі (edge computing), дозволяє значно зменшити витрати на пропускну здатність мережі. Цей підхід також знижує енергоспоживання, оскільки бездротовий зв'язок, який потребує значної енергії, використовується лише в разі необхідності.

Додаткову обробку цих даних можна здійснювати через хмарні обчислення (де дані обробляються на централізованих серверах) або через обчислення туману (де сервери знаходяться ближче до кінцевих пристроїв, на "краю" мережі). Це дає змогу оптимізувати обробку даних і значно знизити навантаження на мережу та енергетичні ресурси.

Бездротові мережі є важливою складовою частиною систем IoT, оскільки вони спрощують процес підключення та експлуатації пристроїв. Проте вони також спричиняють певні проблеми та обмеження. Радіозв'язок, як правило, вимагає більше енергії порівняно з дротовим з'єднанням, що може бути важливим фактором для IoT-систем з обмеженими ресурсами. Крім того, багато бездротових мереж, що використовуються в сучасних пристроях IoT, були спочатку розроблені для інших цілей, таких як телефонія та мультимедіа. Це означає, що вони не завжди оптимізовані для ефективного управління подіями та можуть споживати багато енергії через неефективні протоколи зв'язку.

Неприємністю в IoT є те, що багато пристроїв на краю мережі та їх бездротові мережі не підтримують Інтернет-протокол (IP), який вводить додаткові накладні витрати на пакетування та обробку даних. Замість того, щоб використовувати IP, багато IoT-пристроїв обходять його і використовують вузли для забезпечення своєї присутності в Інтернеті. Це також вказує на те, що більшість мереж IoT не керуються фахівцями з комп'ютерних наук і повинні

бути простими в розгортанні та самокерованими.

Характеристики систем, керованих подіями, дозволяють вузлам IoT залишатися досить простими. Крім того, вимоги до низького енергоспоживання також спонукають до зменшення рівня інтеграції вузлів, що сприяє їх більшій простоті та ефективності.

Технологія VLSI та закон Мура є важливими чинниками розвитку систем IoT, оскільки вони дозволяють виробляти надзвичайно дешеві вузли. Мікросхеми дуже малого розміру можуть забезпечити достатньо обчислювальних потужностей, пам'яті та мережевих ресурсів для виконання функцій вузлів IoT. На відміну від традиційних мікропроцесорних та споживчих електронних пристроїв, де розміри мікросхем часто перевищують кілька квадратних міліметрів, ці мікросхеми є достатньо великими для багатьох IoT-вузлів.

Безпека є важливою вимогою для всіх типів комп'ютерних систем, включаючи IoT, але вона часто залишається недостатньою в багатьох IoT-системах порівняно з традиційними операційними системами Windows, Mac або Linux. Проблеми безпеки в IoT виникають через кілька факторів: відсутність належних функцій безпеки в апаратних засобах, погано розроблене програмне забезпечення з численними вразливостями, використання паролів за замовчуванням та інші проектні помилки. Небезпечні вузли IoT можуть створювати загрози для безпеки всієї системи. Оскільки ці вузли часто мають тривалий термін служби (кілька років), величезна кількість незахищених пристроїв ще довго буде створювати проблеми з безпекою. Небезпечні IoT-пристрої можуть також стати вразливими для атак, таких як відмова в обслуговуванні, що впливає на інфраструктуру Інтернету.

Конфіденційність тісно пов'язана з безпекою, але вимагає специфічних заходів на рівнях програм, мереж та пристроїв. Дані користувачів повинні бути захищені не лише від викрадення, але й сама мережа повинна бути спроектована так, щоб менше приватні дані не давали змоги отримати доступ до більш чутливих даних.

## 1.3 Перспективи розвиткутаіснуючі проблеми IoT

Тенденції розвитку IoT повинні бути інтегрованими, стандартизованими та широко впровадженими. Масштабне впровадження сервісів має відбуватися на основі визначених стандартів. Проте IoT залучає велику кількість виробників, охоплює різні галузі та значно відрізняється в залежності від сценаріїв використання і вимог користувачів, що в свою чергу впливає на масове комерційне впровадження супутніх сервісів. Розвиток IoT є поетапним процесом, і існує ще багато невирішених проблем, таких як створення енергоефективних вузлів і обчислювальних систем, забезпечення низької вартості та мінімальних затримок зв'язку, вдосконалення технологій ідентифікації і позиціонування, самоорганізація систем і впровадження розподіленого інтелекту.

Хоча IoT відкриває нові можливості для різних галузей і кінцевих користувачів, на даний момент сама концепція IoT потребує розвитку в теоретичному, архітектурному та стандартному аспектах, щоб інтегрувати віртуальний і фізичний світи в єдину систему [2]. Зважаючи на це, виділяються кілька ключових проблем, які потребують вирішення.

Архітектурні виклики — IoT охоплює величезний спектр технологій, що включає зростаючу кількість розумних пристроїв і датчиків (наприклад, камер, біометричних, фізичних та хімічних сенсорів), які часто працюють непомітно та прозоро. Очікується, що зв'язок між цими пристроями буде постійним і відбуватиметься в будь-якому місці для різноманітних супутніх послуг, часто через бездротові, автономні та спеціалізовані канали. Послуги стають мобільнішими, децентралізованими та складнішими. Тому інтеграція даних у різних середовищах в IoT є складним завданням, що вимагає модульних взаємодіючих компонентів. Для ефективної інфраструктури необхідні системи, які збиратимуть дані з різних джерел, аналізуватимуть та інтерпретуватимуть ці дані, порівнюватимуть їх з історичними відомостями і підтримуватимуть прийняття рішень. Тому єдиний архітектурний підхід не може охопити всі

можливі сценарії застосування.

Різноманітні еталонні архітектури повинні співіснувати в IoT. Вони повинні бути відкритими та сумісними зі стандартами, не обмежуючи користувачів у виборі рішень і підтримці інновацій. Архітектури IoT повинні бути достатньо гнучкими для підтримки таких технологій, як ідентифікація (RFID, мітки), розумні пристрої та об'єкти (апаратні та програмні рішення).

Технічні виклики: технологія IoT може бути складною з кількох причин. По-перше, в існуючих мережах та додатках присутні застарілі і різнорідні архітектури, оскільки різні програми та середовища потребують різних мережевих технологій, що має суттєві відмінності, зокрема в діапазонах і характеристиках стільникових, бездротових локальних мереж і технологій RFID [3]. По-друге, комунікаційні технології, включаючи стаціонарні і мобільні зв'язки, лінії електропередач, бездротові мережі та технології малого діапазону, повинні забезпечувати надійне підключення за низькими витратами для різних пристроїв, як простих, так і складних. Нарешті, існує безліч різних додатків, що потребують специфічних вимог щодо того, як пристрої мають взаємодіяти, а також які заходи безпеки необхідно застосовувати. У результаті, складність і різноманітність технологій можуть створити проблеми, а непотрібна конкуренція та бар'єри для розгортання на ринках заважають розвитку. Залежності в системах і комунікаційних механізмах можуть блокувати перехід до найбільш економічних і ефективних платформ. Все це може ускладнити процес підключення великої кількості пристроїв у рамках IoT.

Апаратні виклики — розумні пристрої з підвищеною взаємодією між собою сприятимуть розвитку інтелектуальних систем, які зможуть автономно функціонувати. Це дозволить швидше впроваджувати програми IoT та створювати нові сервіси. Тому основна увага в апаратних дослідженнях зосереджена на розробці бездротових ідентифікованих систем, що мають малий розмір, низьку вартість, але при цьому достатню функціональність. Оскільки пропускна здатність IoT-терміналів може коливатися від kbps до mbps, залежно від застосування (від простих даних до відеопотоків), вимоги до обладнання

значно варіюються. Однак дві основні вимоги залишаються незмінними: дуже низьке енергоспоживання в режимі сну та низька вартість. Якщо припустити, що час сну в мільйон разів перевищує активний час, споживана потужність у сплячому режимі повинна бути на мільйон разів меншою, ніж при активному стані. Однак це поки що неможливо при використанні сучасних технологій, коли термінал IoT спить і все одно приймає радіочастотні сигнали. Це створює труднощі навіть з використанням вдосконалених КМОП-кремнієвих технологій, що мають відносно високий рівень витоку енергії. Тому апаратне забезпечення та архітектура протоколу сну залишаються головним завданням для IoT. Враховуючи можливість використання мільярдів IoT-терміналів, вартість кожного пристрою повинна бути надзвичайно низькою.

Наразі ще не існує дешевих рішень для точного позиціонування IoT-пристроїв, особливо для терміналів короткого діапазону, де точність повинна бути дуже високою. Низька активна потужність також є проблемою для недорогих терміналів [4]. Традиційно, низька вартість означає знижену продуктивність або більшу затримку обробки, що, в свою чергу, призводить до підвищеного енергоспоживання. Оскільки ресурс спектра обмежений, IoT може використовувати вищі частоти, такі як діапазони понад 5 ГГц. Однак, чим вища частота, тим більше енергії буде споживати РЧ ПА. Як варіант, для майбутнього IoT може бути розглянута дуже вузька смуга спектру між двома використовуваними смугами, хоча вартість пасивних компонентів у такому випадку буде вищою, що є потенційною проблемою.

Проблема конфіденційності та безпеки: у порівнянні з традиційними мережами, питання безпеки та конфіденційності в системах IoT стають все більш актуальними [5]. Багато зібраної інформації стосується особистої конфіденційності користувачів, що робить захист цих даних важливим аспектом безпеки в Інтернеті речей. Оскільки IoT поєднує в собі різні пристрої, послуги та мережі, безпека в цій сфері повинна охоплювати більше об'єктів та рівнів управління, ніж традиційні мережі. Існуючі системи безпеки, розроблені з урахуванням потреб людського спілкування, можуть не підходити або бути

непридатними для безпосереднього застосування в контексті IoT. Використання старих механізмів безпеки може ускладнити взаємодію між різними елементами IoT. Для цієї сфери необхідні економічно ефективні рішення, орієнтовані на взаємодію між пристроями (M2M), щоб забезпечити належний рівень конфіденційності та безпеки. Більшість досліджень будуть зосереджені на контролі за конфіденційністю. Важливими аспектами стануть низька вартість, низька затримка та енергоефективні криптографічні алгоритми, а також відповідне гнучке обладнання для сенсорів та пристроїв.

Стандартизаційні виклики — стандарти є ключовими для розвитку IoT. Вони мають важливе значення для забезпечення рівного доступу та використання технологій усіма учасниками. Розробка та узгодження стандартів, а також пропозицій щодо них, сприяють ефективному розвитку інфраструктури IoT, додатків, послуг і пристроїв. Загалом стандарти, що створюються за участю багатосторонніх сторін, а також інформаційні моделі та протоколи в цих стандартах, повинні бути відкритими. Процес розробки стандартів повинен бути прозорим для всіх зацікавлених учасників, а прийняті стандарти мають бути загальнодоступними та безкоштовними для використання. В умовах сучасного мережевого світу глобальні стандарти часто є більш актуальними та важливими, ніж локальні угоди.

Виклики бізнесу: для зрілих програм їх бізнес-моделі та сценарії застосування зазвичай чіткі і легко інтегруються в технічні вимоги, що дозволяє розробникам зосередитися на технічних аспектах, не витрачаючи час на бізнес-питання. Однак для IoT ситуація складніша: бізнес-моделі та сценарії застосування включають велику кількість можливостей та невизначеностей, що ускладнює узгодження бізнесу і технологій. Оскільки одне рішення не підходить для всіх, IoT стає більш складною бізнес-моделлю. Хоча деякі дрібномасштабні програми виявилися вигідними в окремих галузях, це не завжди працює при їх поширенні на інші сфери. На ранніх етапах розвитку IoT важливо враховувати бізнес-аспекти, щоб знизити ризик можливих невдач.

Перспективи IoT. З розвитком розподілених інтелектуальних технологій

обробки інформації, системи IoT забезпечують доступ до інтелектуального зондування завдяки обміну даними та співпраці. Постійне впровадження та удосконалення стандартів сприятиме інтеграції IoT в наше повсякденне життя. IoT відкриває нові можливості для веб-сервісів, що, у свою чергу, підвищує комерційний і соціальний потенціал майбутнього IoT [6]. З прогресом у масштабах, спільному використанні та інтелектуальних рішеннях, завдяки технологічному розвитку, стандартизації та практичному досвіду, IoT продовжить розширювати своє застосування в різних галузях, залучаючи все більше підприємств до цієї трансформації.

Взаємодія: інформаційна сумісність стане основною між різними пристроями, підприємствами, галузями та країнами. Моделі застосування змінюватимуться від закритих до відкритих, що сприятиме створенню глобальної системи додатків IoT, яка обслуговуватиме різноманітні сфери. Сумісність є критично важливою проблемою для інтеграції різних рівнів, таких як фізичний, пристрій, зв'язок (протоколи утиліт та спектр), функція та застосування. Ці рівні традиційно базуються на різних мовах і протоколах, тому необхідно розробити прозорі мови та протоколи для кожного рівня і домену. Потрібен комплексний підхід для ефективного вирішення питань взаємодії пристроїв і послуг IoT на різних рівнях.

Інтелектуальна система — IoT дозволить створювати інтегровані ділові та соціальні мережі завдяки швидким, надійним і безпечним мережам. Розвиток системного інтелекту буде ключовим для IoT, з акцентом на контекстну обізнаність та обмін інформацією між речами. Збільшення та адаптація інтелекту на рівні пристроїв стануть основними напрямками досліджень, зокрема інтеграція датчиків і виконавчих механізмів, висока ефективність, багатостандартні та адаптивні підсистеми зв'язку, а також адаптовані антени. Інтелектуальні можливості можуть бути реалізовані за допомогою мікроконтролерів (MCU) на вищих шарах системи. Однак фізичний рівень поки що відстає від необхідного рівня інтелекту, наприклад, для адаптації пристроїв IoT до різних радіоінфраструктур. Для цього потрібно вдосконалити чотири

основні частини фізичного рівня, щоб пристрої IoT могли адаптуватися або стати інтелектуальними. Зокрема, програмовані процесори базової смуги будуть використовуватися для адаптації до різних алгоритмів модуляції, алгоритмів виправлення помилок, ширини смуги каналів і різних сценаріїв каналу. Програмне забезпечення для контролю ВЧ-систем буде важливим для адаптації трансиверів до місцевих радіочастотних вимог. Повністю цифровий РЧ-випромінювач стане необхідним для зниження енергоспоживання і запропонує програмовані можливості для підсилювачів потужності, щоб адаптуватися до вимог радіопередачі. В кінці кінців, керовані інтегровані пасивні компоненти стануть важливою складовою для з'єднання інтелектуальних напівпровідникових елементів в датчикові вузли з низькою вартістю, компактними розмірами і мінімальним енергоспоживанням.

Енергетична стійкість —у майбутньому енергоефективні та автономні системи стануть критичними для розвитку IoT. Для цього необхідно розробити нові способи отримання енергії з навколишнього середовища. Також важливо покращити ефективність обробки та зв'язку через нові схеми, інноваційні підходи в програмуванні та подальший розвиток енергоефективних протоколів і інтелектуальних антен. Ключовими факторами для впровадження автономних бездротових інтелектуальних систем стануть нові, компактні та ефективні акумулятори, паливні елементи, а також пристрої для виробництва енергії, які поєднують методи передачі та збирання енергії. Зарядка глобальних IoT-терміналів, енергоспоживання точок доступу та шлюзів, а також енергоспоживання процесів обробки даних в інфраструктурах IoT будуть основними джерелами енергоспоживання у майбутньому. Механічний збір енергії також буде достатнім для мережі тіла як частини IoT.

# 2АНАЛІЗМОДЕЛЕЙІНТЕРНЕТУРЕЧЕЙ

## 2.1 Протоколи IoT, орієнтовані на ефективність

У цьому підрозділі розглянемо деякі протоколи, що використовуються для обробки даних в IoT-системах. Протоколи зв'язку можуть не надавати достатньої абстракції для багатьох додатків. Системи IoT вимагають мульти-стрибкових, наскрізних з'єднань і можуть мати складні взаємозв'язки між джерелами даних та їх отримувачами. Протоколи більш високого рівня можуть забезпечувати послуги, що точніше відображають потреби систем IoT. Оскільки більшість IoT-систем є неоднорідними і довговічними, зазвичай використовуються стандарти, а не спеціалізовані протоколи. Існує кілька різних протоколів, що частково застосовуються в IoT-системах [7], але користувацький простір ще не обрав єдиний стандарт для служб зв'язку IoT. Враховуючи поширення подійно орієнтованих моделей в IoT, протокол має підтримувати комунікацію у вигляді подій. Протокол HTTP використовує модель запиту/відповіді: клієнт відправляє запит на гіпертекстовий об'єкт, а сервер відповідає відповідним об'єктом. Протокол запиту/відповіді [8] вимагає меншої взаємодії між клієнтом і сервером, як показано на рис. 2.1.

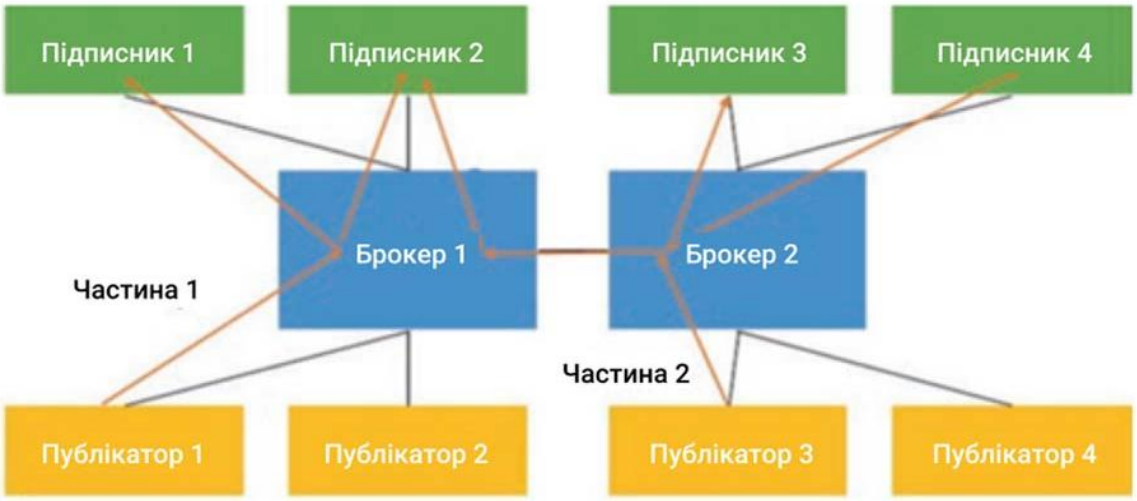



Рисунок 2.1 —Модельзапиту/відповіді

Сервер, який виконує роль видавця, класифікує повідомлення за певними

категоріями. Клієнти підписуються на ті категорії, які їх цікавлять. Видавці публікують повідомлення, які потім передаються їхнім передплатникам. Повідомлення можуть бути організовані за темами, і всі повідомлення, що належать до конкретної теми, поширюються брокерами серед передплатників цієї теми. Брокер має інформацію про підписників, а видавець — ні. Брокери можуть взаємодіяти між собою через мостовий протокол, що дозволяє побічно публікувати повідомлення: спочатку воно надходить від видавця до першого брокера, потім передається до другого брокера і, зрештою, до підписників, які не підключені до першого брокера.

Система розподілу даних (DDS) [9] є програмною архітектурою для публікації та підписки, при цьому існують різні її реалізації. Домен DDS створює єдиний логічний простір для даних, який управляється через кілька локальних сховищ. Видавці та підписники автоматично виявляються в мережі. Видавці можуть задавати параметри якості обслуговування (QoS), які обов'язково застосовуються брокерами.

Протокол запиту/відповіді в реальному часі (RTPS) [10] є так званим дротовим протоколом, що визначає правила комунікації для DDS і інших систем публікації/підписки. RTPS забезпечує необхідні параметри QoS, стійкість до відмов та безпеку.

Розроблена архітектура чутливих до часу систем публікації/підписки, яка має бути масштабованою для використання в Інтернеті. Визначено три основні цілі проектування: мінімізація затримок, гарантована доставка навіть у разі кількох відмов та забезпечення працездатності при масштабуванні. Для таких систем були розроблені моделі несправностей, що включають: проблеми мережі (втрата, упорядкування, пошкодження, затримки, перевантаження, розділення), збої в посиланнях, збої вузлів і випадкове відключення або приєднання вузлів до системи.

Архітектура системи має три рівні абстракції: на мережевому рівні є домени з вузлами; на рівні вузлів існують кластери, де всі члени одного кластера належать до того ж домену; на рівні координаторів забезпечується

маршрутизація повідомлень через деревоподібну топологію, побудовану на основі розподіленої хеш-таблиці. Координатор є надлишковим для забезпечення відмовостійкої координації.

Для забезпечення відмовостійкості розроблено модель різноманітності шляхів, яку можна обчислити з обмеженими знаннями про мережу. Для зменшення накладних витрат і підвищення надійності застосовано механізм зв'язку, знайомий за семантикою. Система використовує оцінювачі простору станів на видавці та абоненті, що дозволяє підтримувати безперервність значень датчиків навіть за наявності змін у мережі. Оцінка стану виглядає як xk+1 = Fk+1xk. Архітектор системи визначає модель точності δ для кожного датчика. Це обмеження використовується для регулювання вимог до пропускної здатності.

Система також інтегрує DDS із програмно визначеним мережевим протоколом OpenFlow для забезпечення параметрів QoS. Додатково вводяться два нових параметри QoS — MINIMUM_SEPARATION та E2E_LATENCY, які визначаються абонентами і не можуть бути легко визначені за допомогою стандартних параметрів DDS.

Протоколи можна поділити на дві основні категорії: ті, що залежать від конкретного фізичного рівня, і ті, що не мають такої залежності. Протоколи, які прив’язані до певного фізичного рівня, не використовують Інтернет-протокол, тоді як протоколи, незалежні від фізичного рівня, працюють із IP.

Zigbee [11] є сітчастою мережею, розробленою для застосування в умовах низького енергоспоживання. Існують різні стандарти похідних додатків, що спеціалізуються на таких галузях, як розумні будинки та комунальні послуги. Zigbee базується на стандартах IEEE 802.15.4 для фізичного та канального рівнів (PHY та MAC). Стандарт 802.15.4 підтримує три діапазони частот: 868 МГц, 915 МГц і 2,4 ГГц, забезпечуючи швидкість передачі даних від 20 до 250 кбіт/с, залежно від використовуваної частоти.

Шар Zigbee NWK розташований поверх рівня MAC 802.15.4 і забезпечує послуги управління даними та адресацію. Рівень APL включає три основні

підшари: підтримку додатків, об'єкти пристроїв Zigbee та програмний фреймворк.

Zigbee пропонує два основні типи моделей мережевої безпеки: централізовану модель безпеки, що керується координатором або центром довіри, і розподілену модель безпеки, де немає центрального контролера довіри. Вузли можуть приєднуватися до будь-якої з цих мереж і адаптуватися до відповідної моделі безпеки. Мережі формуються координаторами або маршрутизаторами, які вибирають доступний канал для з'єднання. Координатори формують централізовані мережі безпеки, а маршрутизатори — розподілені. Мережеве управління є процесом приєднання вузлів до мережі, що включає виявлення відкритої мережі та отримання мережевого ключа. Кластери визначають інтерфейси функцій і домени мережі.

Bluetooth Low Energy (BLE) [12] — це частина стандарту Bluetooth, розроблена для низькоспоживчих пристроїв, таких як ті, що живляться від монетних батарей. Пристрій BLE може функціонувати як передавач, приймач або в обох ролях одночасно. Стек протоколів для Bluetooth Classic зображений на рисунку 2.2.

Посилальний рівень надає послуги рекламування, що дозволяють пристроям сканувати та ідентифікувати вузли та мережі. Крім того, пристрої можуть виступати як шлюзи до Інтернету за допомогою перекладу мережевих адрес. Протокол BLE є державним, і його основною характеристикою є оптимізація для зниження споживання енергії.

LoRa [13] розроблений для великих IoT додатків, що вимагають покриття великих територій, із базовою станцією, що може охоплювати сотні квадратних кілометрів. Цей протокол підтримує мережеву топологію із шлюзами для кінцевих пристроїв, організованих у зіркову мережу. Швидкість передачі даних в LoRa варіюється від 0,3 до 50 кбіт/с.


Рисунок2.2— Стек протоколів для Bluetooth Classic

MQTT [14] — це протокол, орієнтований на IoT, що базується на семантиці публікації/підписки. Він розроблений з метою мінімізації накладних витрат і є нейтральним щодо корисного навантаження даних. Протокол забезпечує три рівні якості обслуговування (QoS): "щонайменше один раз" (найкращі зусилля), "принаймні один раз" (можливе дублювання повідомлень) та "рівно один раз" (гарантована доставка без дублювання).

MQTT використовує модель публікації/підписки, де кожне повідомлення при публікації може мати атрибут утримання. Повідомлення з рівнями QoS "принаймні один раз" або "рівно один раз" повинні бути позначені прапором збереження. Новий передплатник теми отримає останнє публікацію на цю тему. Під час встановлення з'єднання клієнт може передати серверу заповіт, щоб вказати повідомлення для публікації у разі його несподіваного відключення.

Повідомлення класифікуються за допомогою ієрархічних рядків тем, подібних до шляхів файлів. Теми організовані у вигляді дерева, де назви тем відображаються як шляхи до вузлів дерева, з іменами, розділеними знаком "/". Абоненти можуть використовувати підстановочні символи у рядках тем: "+" замінює один рівень дерева тем, а "#" — кілька рівнів дерева.

XMPP — це протокол для потокового передавання XML, який забезпечує безпеку, автентифікацію, а також надає інформацію про доступність мережі та реєстрацію клієнтів. XMPP-IoT є діалектом XMPP, спеціально розробленим для використання в додатках Інтернету речей (IoT).

REST [15] є популярним підходом для веб-служб і застосовується також для обслуговування IoT. Це архітектурний шаблон для передачі даних без обов'язкового використання протоколу HTTP. REST забезпечує структуру, що дозволяє працювати з ресурсами, організованими за каталогами. Для передачі даних у форматах XML або JSON клієнти можуть використовувати методи GET, PUT, POST і DELETE для доступу до ресурсів.

CoAP [16] — це протокол веб-передачі, заснований на архітектурному шаблоні REST, який спеціально розроблений для пристроїв IoT. CoAP підтримує різні типи корисних даних, зокрема XML та JSON, і дозволяє ефективно передавати інформацію в середовищі з обмеженими ресурсами, забезпечуючи низьке споживання енергії та пропускної здатності.

Google Cloud Pub/Sub [17] є платформою для надання послуг публікації та підписки, яка може бути використана як для IoT, так і для інших систем. Теми та підписки організовані як колекції в архітектурі REST. Система має дві основні площини: площину даних для передачі повідомлень і площину управління для розподілу даних через маршрутизатори, які відомі як сервери управління. Експедитори відповідають за зберігання та доставку даних.

Маршрутизатори забезпечують узгодженість і однорідність даних, застосовуючи алгоритм послідовного хешування. Життєвий цикл повідомлення в системі включає кілька етапів: видавець надсилає повідомлення, яке записується на зберігання. Абоненти отримують повідомлення і підтверджують

його доставку. Після підтвердження повідомлення видаляється з пам'яті, коли хоча б один абонент для кожної підписки підтвердить отримання. Система має механізми для моніторингу та пом'якшення проблем, що виникають при обслуговуванні.

Amazon Web Services (AWS) IoT [18] — це керований хмарний сервіс для пристроїв IoT, які називаються "речами". У цій системі кожна річ має свою хмарну модель, відому як "тінь речі". Механізм правил дозволяє перетворювати повідомлення на основі визначених правил і направляти результати до інших служб AWS. Брокер повідомлень в AWS IoT використовує протокол MQTT. Для кожної речі в системі реєструється унікальна ідентичність.

Microsoft Azure також надає послуги для IoT. Однією з її ключових технологій є Service Fabric, що є системою комунікації проміжного програмного забезпечення, яка підтримує мікросервіси на кластері. Мікросервіси можуть бути без державного статусу або з ним. Azure також надає контейнерну модель для додатків, забезпечуючи ізольоване середовище, яке залежить від операційної системи, на відміну від віртуальних машин. Цей сервіс пропонує як структуровані, так і неструктуровані бази даних, а також API для доступу до служб штучного інтелекту.

Azure IoT Hub — це платформа для підключення, моніторингу та управління пристроями IoT. Вона дозволяє двосторонню комунікацію між пристроями та хмарною платформою, забезпечуючи безпечну передачу даних. IoT Hub підтримує різноманітні протоколи, включаючи MQTT, HTTPS і AMQP. Платформа також пропонує механізми автентифікації, дозволяючи легко управляти пристроями, що підключаються, і забезпечує їх безпеку через обмеження доступу та шифрування.

Azure Digital Twins — це ще один важливий сервіс для IoT, який дозволяє створювати цифрові моделі реальних об'єктів і середовищ. Ця технологія дозволяє відстежувати стан та взаємодії реальних систем в режимі реального часу, забезпечуючи точніші дані для аналізу та оптимізації. Платформа підтримує інтеграцію з іншими компонентами Azure для створення складних

IoT рішень.

Інтерфейс Azure IoT Central— це рішення для побудови, керування та моніторингу IoT-додатків, що не вимагає глибоких знань програмування. Ця платформа надає користувачам інтуїтивно зрозумілий інтерфейс для створення і налаштування IoT-систем, що дозволяє швидко адаптувати її під конкретні бізнес-потреби.

## 2.2 АналізархітектуритамоделіIoT

Інтернет речей (IoT) значно змінив наш погляд на застосування технологій у різних сферах людської діяльності. Очікується, що ця трансформація буде лише посилюватися в найближчі роки. IoT розширюється на численні галузі, такі як розумна енергетика, виробництво, сільське господарство, охорона здоров'я, безпека, інтелектуальні міста, розумні будівлі та екологічні ініціативи. У всіх цих випадках застосовується одна й та сама основна модель: безліч інтелектуальних пристроїв, взаємопов'язаних через дротові або бездротові канали зв'язку, взаємодіють і координують свої дії для досягнення поставленої мети.

У промисловому секторі значну роль у розвитку IoT відіграє концепція розумних фабрик [19], що є частиною стратегії Industrie 4.0. Ця ініціатива, поряд із розвитком Індустріального Інтернету та Європейською ініціативою для фабрик майбутнього, сприяє впровадженню IoT у виробничі процеси. Метою є підвищення гнучкості та ефективності виробництва при одночасному зниженні витрат на виготовлення продукції.

Концепція індустріального Інтернету речей (IIoT) є важливою складовою загальної еволюції Інтернету речей (IoT), але має свої унікальні особливості, які створюють специфічні виклики. Основною проблемою є інтеграція традиційних програмованих логічних контролерів (PLC) та систем контролю і збору даних (SCADA), що становлять основу промислових мереж та інфраструктури операційних технологій (OT).

Системи PLC і SCADA традиційно працюють в окремому середовищі, не

інтегрованому з інформаційними технологіями (IT), оскільки вони орієнтовані на забезпечення специфічних вимог промисловості, таких як безперервна робота, реальний час, і безпека. Вони є критично важливими для таких систем, як енергетичне виробництво та мережі розподілу енергії, де час і надійність є вирішальними.

Однак, розвиток IIoT створює нові можливості для промисловості, але також викликає складні проблеми для інтеграції OT з традиційними IT-системами підприємств. Наприклад, сучасні системи ERP (планування ресурсів підприємства) потребують розширення, щоб включати виробничі операції, які зараз керуються через системи MES (виконання виробничих процесів). Проблеми сумісності між MES і ERP системами створюють значні труднощі в їх інтеграції.

Ідеальна інтегрована система, яка охоплює всі рівні підприємства (від бізнес-процесів до датчиків), може забезпечити значну гнучкість та відкриває нові можливості для підприємств. Вона дозволяє оптимізувати всі етапи — від управління та планування до збору та аналізу даних — що підвищує ефективність і знижує витрати. Проте для реалізації такої системи необхідно вирішити ще ряд проблем, таких як сумісність, стандартизація та забезпечення безпеки на всіх рівнях взаємодії між IT і OT системами.

Інфраструктура виробництва та розподілу енергії є ключовою частиною операційних технологій (OT), на яких базуються сучасні розумні мережі. Енергетичний сектор є основним пріоритетом розвитку IIoT, оскільки зростаюча потреба в енергії, зокрема через приріст населення, підвищує необхідність у ефективному управлінні енергетичними ресурсами. Енергоменеджмент відіграє важливу роль не лише в забезпеченні надійності постачання енергії, а й у досягненні економічної ефективності та зниженні витрат у промисловості.

Окрім енергетики, системи IIoT активно інтегруються в інші критичні сектори інфраструктури, такі як управління водними ресурсами та транспорт, де застосування інтелектуальних технологій дозволяє оптимізувати ресурси,

підвищувати ефективність і забезпечувати безпеку.

Проблеми сумісності між IT та OT у контексті IIoT є одними з основних викликів для інтеграції та розвитку індустріальних технологій. Ці проблеми виникають через відмінності в архітектурах, протоколах і підходах до управління системами, а також через необхідність адаптації традиційних технологій до нових вимог. Для ефективної реалізації IIoT необхідно розробити уніфіковані архітектури, які зможуть підтримувати різноманітні інтерфейси між системами IT та OT.

Одним із важливих аспектів є безпека та конфіденційність. Інтеграція інформаційних і операційних технологій вимагає уніфікованого підходу до забезпечення безпеки, оскільки на рівні OT акцент робиться на безперервність роботи і фізичну безпеку, тоді як в IT безпека стосується захисту даних і інформаційних систем. Важливо, щоб системи IIoT мали надійні механізми захисту, які одночасно відповідають суворим вимогам до відповідності в реальному часі і низького енергоспоживання.

Попри прогрес у розробці стандартів, таких як ISO/IEC 27000 та ISA/IEC 62443, що покривають аспекти безпеки, кібербезпека та конфіденційність залишаються важливими викликами. Існуючі технології повинні бути адаптовані до нових вимог, щоб гарантувати безпеку та стабільність IIoT-систем, забезпечуючи захист даних на всіх етапах від збору та обробки до зберігання та передачі інформації.

Цей розділ розглядає концептуальні основи та етапи розвитку Індустріального Інтернету речей (IIoT), починаючи з концепції Industrie 4.0 і переходячи до сучасного Індустріального Інтернету. Представлено еволюцію еталонних архітектур IIoT, що виникають у результаті зусиль Міжнародної організації зі стандартизації (МСЕ) і стають частиною ініціатив Індустріального Інтернет-консорціуму. Окрім цього, розглядаються ключові проблеми, що виникають при розвитку та впровадженні IIoT, із особливою увагою до енергетичного сектора. Зазначено, що питання безпеки та конфіденційності є важливими як у контексті IIoT, так і в рамках загальної екосистеми Інтернету

речей (IoT).

Industrie 4.0 є стратегічною ініціативою, спрямованою на інтеграцію технологій Інтернету речей (IoT) у виробничу та промислову сфери. Її основна мета полягає у забезпеченні Німеччини лідерства у виробничому секторі шляхом забезпечення ефективного та економічного виробництва через гнучкі робочі процеси. Досягнення цієї мети передбачає активне впровадження кіберфізичних систем у виробничі процеси, що дозволяє вводити інтелектуальні елементи в системи і процеси, забезпечувати їх високу зв'язність та взаємодію, а також досягати координації в складних, але гнучких процесах, які призводять до створення високоякісної та недорогої продукції.

Industrie 4.0 отримала свою назву від визначення нової стадії, що розвивається, як четверту промислову революцію. Зазначено, що промислове виробництво пережило три попередні революції. Перша революція, що відбулася в 18-19 століттях, полягала у введенні механізованих виробничих процесів, де енергія забезпечувалася водою та парою. Друга революція, яка відбулася з появою електричної енергії, призвела до масового виробництва та підвищення продуктивності завдяки використанню електрики. Третя революція, що почалася після Другої світової війни, включала інтеграцію електроніки та програмного забезпечення, що сприяло автоматизації на високих рівнях через використання промислових інформаційних технологій. Зараз багато представників промисловості вважають, що ми стоїмо на порозі четвертої революції, що буде визначена широким впровадженням кіберфізичних систем. Це дозволить не лише значно підвищити рівень автоматизації, але й забезпечить гнучкість у виробничих процесах завдяки програмованим та налаштовуваним виробничим лініям, що зробить масове виробництво більш ефективним і доступним.

Зусилля Industrie 4.0 базуються на широкому використанні обчислювальних та комунікаційних ресурсів. Останні два десятиліття відзначаються значним прогресом у розвитку високопродуктивних процесорів з низьким енергоспоживанням, пам'яті та компонентів для комунікацій, що

забезпечують ефективну обробку даних та з'єднання мереж. Ці технологічні досягнення відкрили нові можливості для обробки даних великою кількістю пристроїв, які застосовуються як у побуті, так і на виробництвах. Розумні споживчі пристрої стали стандартом, а смартфони надають доступ до сотень додатків, що включають послуги від планування маршрутів до мобільного банкінгу та моніторингу здоров'я. Розумні телевізори інтегрують різноманітні форми розваг та мережевих послуг, від індивідуального налаштування каналів до онлайн-ігор і управління домашніми пристроями. Розумні побутові прилади здійснюють моніторинг параметрів, таких як температура та споживання води та енергії, що дозволяє користувачам ефективно управляти своїми домогосподарствами, забезпечуючи комфорт та знижуючи експлуатаційні витрати.

Широке використання обчислювальних ресурсів та з'єднань стає очевидним завдяки значній кількості вбудованих процесорів і компонентів, які випускаються сьогодні. Згідно з [22], більшість вироблених процесорів, близько 98%, застосовуються у вбудованих системах. Крім того, обсяг напівпровідникової пам'яті також зростає, і її виробництво, за прогнозами, збільшиться на 40% порівняно з минулим роком [23]. Останні досягнення в розвитку дротових та бездротових мереж за останні два десятиліття сприяли створенню універсальних мережевих з'єднань, які вже досягають навіть малих міст і селищ. Така розвинена інфраструктура обробки та комунікації веде до еволюції ієрархії вбудованих систем і послуг, наближаючись до рівня Інтернету речей, даних та сервісів.

Прикладом цієї еволюції є застосування вбудованих систем у транспортній сфері. Вони використовуються для управління різноманітними функціями, від розважальних систем автомобіля до налаштування автокрісел. Вбудовані процесори в таких системах налаштовуються для контролю конкретних параметрів, наприклад, висоти та переміщення крісел, відповідно до запитів користувача. Проте вбудовані системи в автомобілях також взаємодіють між собою, або в межах автомобіля, або з навколишнім

середовищем, що дозволяє реалізувати такі послуги, як автоматичні системи оплати проїзду. У такому випадку вбудовані системи в автомобілі та на пунктах оплати спільно працюють для завершення електронних транзакцій.

Наприклад, платіжні системи, що використовуються на кількох платних дорогах, можна інтегрувати в більш складну розподілену систему. Така система забезпечує ефективніше управління дорожнім рухом та збором плати за проїзд, що сприяє зменшенню часу очікування, витрат на пальне для водіїв та експлуатаційних витрат. В результаті цього зростає дохід транспортних органів. В перспективі можна очікувати ще більшу інтеграцію таких транспортних систем з розумними містами, де управління транспортом поєднується з іншими послугами, такими як енергетичне забезпечення, цивільна та аварійна служби, що дозволяє адаптуватися до різних ситуацій у залежності від часу, місця або особливих обставин.

Досягнення в сенсорних технологіях, разом із розвитком вбудованих систем та мереж зв'язку, сприяють реалізації багатьох сучасних сценаріїв. Особливо важливо, що датчики відіграють ключову роль у з'єднанні фізичного світу з цифровим, надаючи цифровим системам цінну інформацію та забезпечуючи інтелектуальне управління системами та процесами. Виробнича та промислова автоматизація, традиційно базуючись на IT-технологіях, активно використовує датчики та електромеханічні системи для розвитку та впровадження технологій інтелектуального управління, що в свою чергу сприяє створенню нових концепцій і послуг. Отже, розробка стратегії Industrie 4.0 та відповідних ініціатив є природним етапом розвитку промислових технологій, що активно інтегруються з досягненнями в споживчих технологіях Інтернету речей.Концепція "розумного заводу" є ключовим елементом стратегії Industrie4.0, оскільки вона ґрунтується на використанні кіберфізичних систем для досягнення високого рівня автоматизації та інтеграції. Вона орієнтована на забезпечення гнучкості, ефективності, автономності, стійкості, безпеки та зниження витрат. Розумні машини будуть взаємодіяти між собою для створення ефективних виробничих систем, а їх інтеграція дозволить досягти більш

високого рівня автоматизації. В рамках цієї концепції всі етапи виробничого процесу будуть автоматизовані, включаючи ефективне управління матеріалами та ресурсами. Виробничі процеси будуть адаптуватися в режимі реального часу для мінімізації витрат та оптимізації операцій, з одночасною безпекою для інфраструктури та операційного персоналу. Це дозволить не тільки скоротити витрати, але й забезпечити можливість взаємодії з клієнтами, які зможуть відстежувати процес виготовлення продукції, а виробники — оптимізувати логістичні ланцюги.

Індустріальний Інтернет речей (IIoT) розвинувся як загальна концепція використання Інтернету речей у промисловості. Це, по суті, розширення ідеї Industrie 4.0, що робить акцент на підвищенні ефективності промислових процесів. IIoT охоплює різні аспекти промислових операцій, зокрема не тільки оптимізацію процесів, але й управління активами, обслуговування та інші важливі елементи.

Оскільки IIoT є частиною широкої концепції IoT, але з фокусом на промисловому секторі, можна сказати, що Industrie 4.0 є підмножиною IIoT, як показано на рисунку 2.3. Таким чином, варто виділити відмінності між IoT та IIoT, враховуючи їхній контекст та цілі.

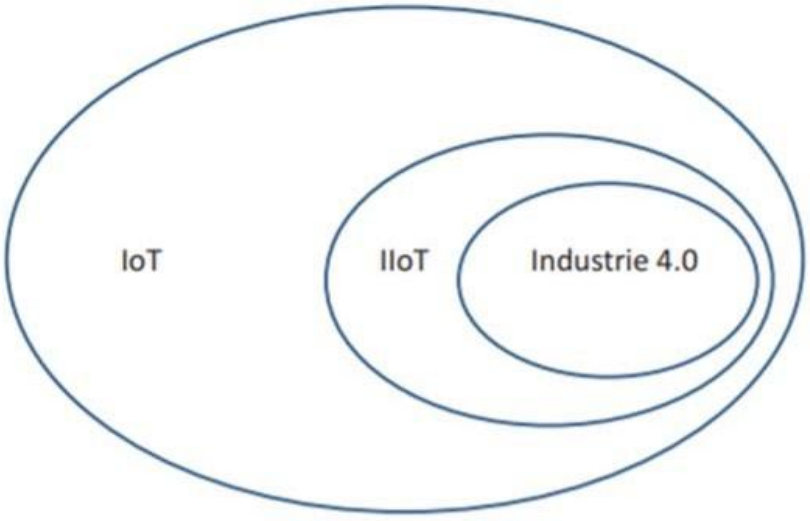



Рисунок2.3— Взаємозв'язок між IoT, IIoT та концепцією Industrie 4.0

Хоча основні концепції IoT та IIoT подібні, зокрема інтеграція

інтелектуальних пристроїв для дистанційного моніторингу, збору даних, обробки та управління, існують відмінності в вимогах до безперервної роботи та безпеки, а також у використанні операційних технологій у промисловому секторі. Наприклад, різниця між побутовою послугою, такою як програма для моніторингу здоров'я на смарт-годиннику, і промисловою послугою, такою як моніторинг парового насоса, є значною. Хоча обидва додатки збирають дані в реальному часі — кроки або температуру тіла в разі охорони здоров'я і тиск або об'єм пари в паровому насосі — та здійснюють передачу даних, визначення подій і надання зворотного зв'язку або команд операторам, у випадку парового насоса вимоги до безперервної роботи та безпеки є набагато жорсткішими. Поломка парового насоса може мати катастрофічні наслідки, призвести до дорогого простою та навіть спричинити травми або загибель людей.

Характеристики промислового сектору, такі як специфічні технології та вимоги, вимагають розробки спеціалізованих рішень, що виправдовує орієнтацію цього сектора на спеціалізовану концепцію IoT. Це призвело до значного інтересу до розробки спеціалізованих концепцій, що охоплюють стратегію, застосування та технології. У зв'язку з численними взаємозалежностями між учасниками процесу, від ланцюгів постачання до просування послуг, виникає велика потреба в сумісних рішеннях на різних рівнях — від пристроїв до послуг. Тому необхідна скоординована діяльність для еволюції до IIoT, що підтримується консорціумами, такими як Індустріальний Інтернет-консорціум, який надає важливе керівництво в цій новій сфері.

Компанія General Electric вперше ввела термін "Індустріальний Інтернет" у 2012 році, ставши лідером у розвитку "Індустріального Інтернету Речей" (IIoT). Вони визначили основні технології, які становлять основу бачення IIoT, зокрема міжмеханічний зв'язок, SCADA, HMI, аналіз промислових даних та кібербезпеку. Цікаво, що компанія прогнозує, що індустріальний Інтернет матиме вплив на 46% світової економіки, а в енергетичному секторі вони очікують, що IIoT вплине на 100% виробництва енергії та 44% споживання

енергії у глобальному масштабі [25].

Розробка та впровадження систем і послуг IIoT потребує створення архітектур, що забезпечують ефективні операції та сумісність, враховуючи передбачувані послуги і велику кількість зацікавлених сторін, включаючи пристрої, кіберфізичні системи, мережі зв'язку, постачальників послуг та бізнес-розробників. З цією метою значна частина зусиль спрямована на створення стандартів та еталонних архітектур, які можуть бути прийняті різними учасниками процесу. У 2012 році Міжнародний союз електрозв'язку (МСЕ) опублікував Рекомендацію ITU-T Y.2060, яка пропонує довідкову архітектуру для Інтернету речей, включаючи програми, що відносяться до IIoT, такі як розумні мережі, інтелектуальні транспортні системи та електронне охорону здоров'я. Індустріальний Інтернет-консорціум (IIC) також розробляє еталонну архітектуру для IIoT і нещодавно випустив Версію 1.7 своєї довідкової архітектури. Ця архітектура значно детальніша, ніж МСЕ, і охоплює всі важливі аспекти для різних категорій зацікавлених сторін. Можна вважати модель IIC еволюційним кроком у порівнянні з моделлю МСЕ, оскільки вона розглядає специфічні аспекти IIoT більш детально, ніж загальна модель IoT, запропонована МСЕ.

Зусилля МСЕ сприяли розширенню розуміння комунікацій, включивши концепції передачі "чого завгодно" та зв'язку "в будь-який час" і "в будь-якому місці". Це важливо, оскільки МСЕ охоплює всі передбачувані програми, зокрема промислові, згадуючи інтелектуальні мережі та транспортні системи серед інших. МСЕ визначає "речі" як фізичні та віртуальні об'єкти, які можна ідентифікувати та підключити до мереж зв'язку, при цьому ці об'єкти мають інформацію, що може бути як статичною, так і динамічною. Оскільки комунікація є критично важливою частиною концепції IoT, фізичні об'єкти повинні бути підключені до "пристроїв", які взаємодіють із мережами, щоб будь-яка аналогова інформація могла бути перетворена в цифрову та передана через мережі. Пристрої можуть бути різними: від тих, що лише передають, зберігають чи фіксують дані, до більш складних пристроїв контролю та

управління, а також універсальних пристроїв із вбудованими можливостями обробки та зв'язку, таких як машини, прилади чи споживча електроніка.

Однією з важливих проблем у довідковій моделі МСЕ є модель зв'язку між пристроями. Як показано на рисунку 2.4, модель включає три методи зв'язку, які використовують шлюзи (G) та мережу зв'язку (CN).

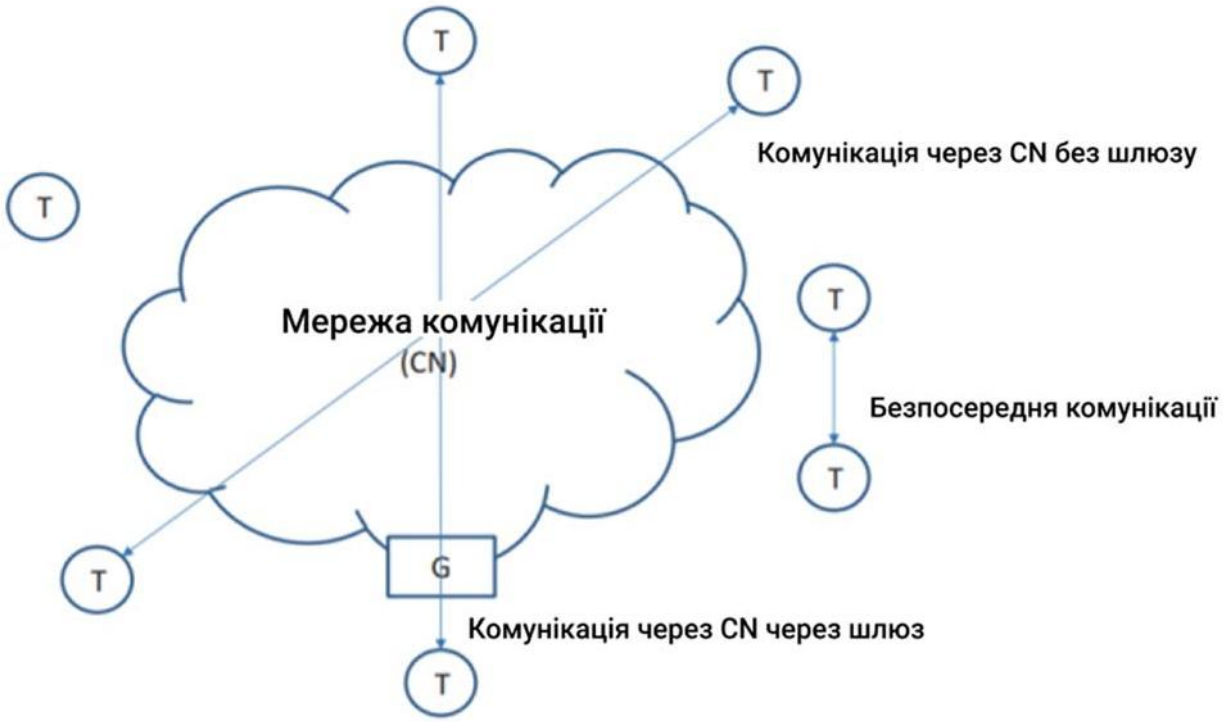



Рисунок 2.4 — Способи з'єднання пристроїв IoT

Пристрої можуть взаємодіяти безпосередньо, без шлюзів, через локальні мережі або мережу зв'язку, або через мережу, що використовує шлюзи для зв'язку. Модель МСЕ визначає основні характеристики IoT, серед яких є взаємозв'язок, масштабованість, неоднорідність, послуги для речей і динамічний характер інформації про пристрої та зв'язок. Взаємозв'язок є критичним, оскільки "все" може підключатися до глобальної мережі для різних цілей. З ростом кількості підключених пристроїв масштабованість стає важливим фактором, який потрібно враховувати на всіх етапах IoT та IIoT. Проблема масштабованості стосується не лише кількості пристроїв та точок зв'язку, але й обсягу генерованих і передаваних даних, а також їх управління,

зберігання і обробки. Динамічний характер пристроїв, що постійно підключаються та відключаються від мереж, робить цей процес складнішим. Відкритий характер IoT і велика кількість зацікавлених сторін, разом з гнучкими та складними ланцюгами поставок, ставлять перед нами завдання інтеграції різнорідних "речей", пристроїв, платформ та послуг. Послуги для "речей" також мають враховувати обмежені ресурси пристроїв, а також вимоги до безпеки, конфіденційності та безпечної роботи, щоб уникнути потенційних проблем і аварій.

Основні характеристики (I)IoT вимагають відповідності еталонної архітектури певним вимогам. Серед основних вимог, зазначених МСЕ, є сумісність, підключення на основі ідентифікації, автономність роботи мереж та послуг, розміщення послуг за місцем знаходження, безпека та конфіденційність, а також можливості управління пристроями та послугами, зокрема підтримка принципу plug and play. Рисунок 2.5 ілюструє еталонну модель ITU IoT, яка була розроблена для задоволення цих вимог.

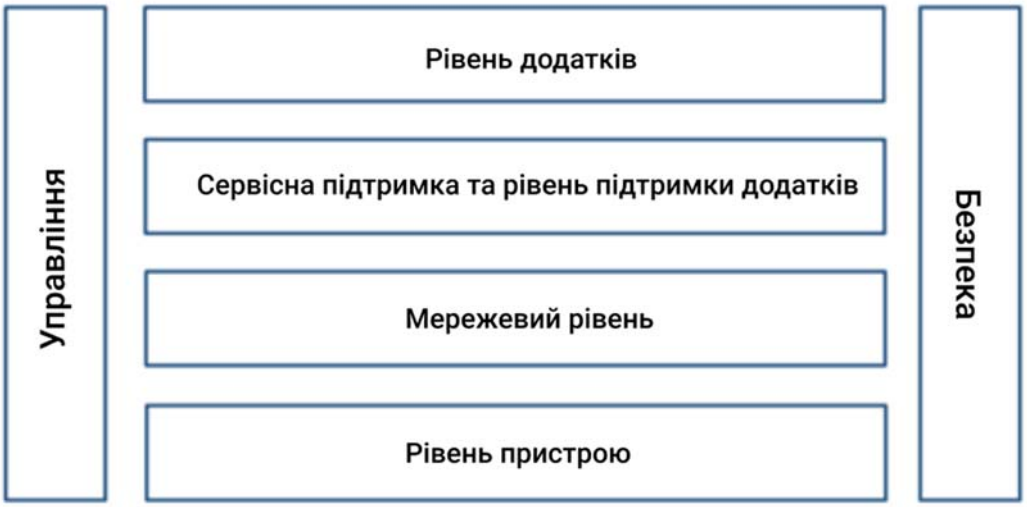



Рисунок2.5 — Еталонна модель ITU IoT

Ця модель є типовою багатошаровою архітектурою з чотирма ієрархічними рівнями: пристрій, мережа, підтримка додатків та послуг, а також прикладний рівень. Крім того, існують два вертикальні рівні, які охоплюють всі

ієрархічні рівні, забезпечуючи функції управління та захисту для кожного з них.

Найнижчий рівень, рівень пристроїв, містить функціональність пристроїв та комунікаційних шлюзів. Оскільки MCE акцентує увагу на зв'язку, цей рівень описує функції пристроїв, орієнтованих на комунікацію:

— пристрої, що передають і приймають інформацію через мережу безпосередньо, без шлюзів;

— пристрої, що використовують шлюзи для передачі та прийому даних;

— пристрої, які здійснюють прямий зв'язок без мережі, але можуть використовувати локальні мережі або створювати спеціалізовані мережі;

— пристрої, здатні вмикати і вимикати функціональність для економії енергії.

Що стосується шлюзів, рівень пристроїв включає всі відповідні комунікаційні технології, як дротові (наприклад, шина CAN), так і бездротові (Wi-Fi, Bluetooth, Zigbee тощо). Крім того, рівень пристроїв охоплює перетворення протоколів, оскільки пристрої можуть використовувати різні протоколи, і для їх взаємодії необхідне перетворення між ними.

Мережевий рівень відповідає за інкапсуляцію даних пристроїв та перетворення протоколів в протоколи, що відповідають мережевому рівню. Цей рівень включає функціональність мережевого та транспортного рівнів з еталонної моделі OSI. Для забезпечення роботи в мережі, мережевий рівень включає управління мережевими підключеннями, мобільністю, а також автентифікацією, авторизацією і обліком. Щодо транспортного рівня, він відповідає за транспортування користувацького трафіку та передачу контрольної і управлінської інформації для послуг IoT та відповідних програм.

Рівень підтримки сервісів та додатків охоплює як загальні, так і специфічні функціональні можливості для надання послуг та додатків IoT. Оскільки послуги та додатки IoT мають розподілений характер, цей рівень включає загальні функції, такі як обробка та зберігання даних, а також спеціалізовані функції для кожного конкретного додатку або послуги. Нові

послуги можуть мати різні вимоги, наприклад, система інтелектуальної мережі має інші вимоги до конфіденційності, ніж система управління дорожнім рухом для транспортних послуг.

Прикладний рівень, який є найвищим в ієрархії, включає додатки та послуги IoT. Вертикальний рівень управління забезпечує як загальну, так і специфічну функціональність для кожного домену додатка. Загальні функції включають управління конфігурацією, топологією, ресурсами, продуктивністю, помилками, безпекою та обліковими записами. Спеціалізовані функції стосуються конкретних вимог програм, таких як інтелектуальний моніторинг лічильників у розумних мережах.

Бізнес-моделі IoT визначаються через п’ять основних ролей, які можуть виконувати зацікавлені сторони: (a) постачальник пристроїв, (b) постачальник мереж, (c) постачальник платформ, (d) постачальник додатків та (e) кінцевий користувач додатків. Постачальники пристроїв надають необхідне обладнання для IoT, тоді як постачальники мереж забезпечують мережеві системи, шлюзи та з'єднання для IoT. Постачальники платформ створюють єдину розподілену IT-платформу з чітко визначеними інтерфейсами, через які додатки можуть обслуговуватися, а постачальники додатків надають послуги через ці платформи, мережі та пристрої. Клієнти додатків є кінцевими користувачами послуг або програм IoT. MCE визначає п’ять типів бізнес-моделей, що залежать від кількості операторів, які беруть участь у програмі, та їхніх конкретних ролей.

На рисунку 2.6 зображені п’ять бізнес-моделей (Моделі 1–5), які ілюструють ділові ролі через складені блоки, подібно до вертикального шару моделі. Кожен блок містить поля (ролі), де однаковий шаблон заповнення у межах стека вказує на те, що одна і та ж організація виконує ролі, відповідні цим полям.

Наприклад, у Моделі 1 одна організація виконує функції постачальника пристроїв, мережі та платформи для додатків, тоді як у Моделі 2 різні учасники виконують окремі ролі: одна сторона відповідає за постачання пристроїв,

мережі та платформи, а інша – за надання додатків.

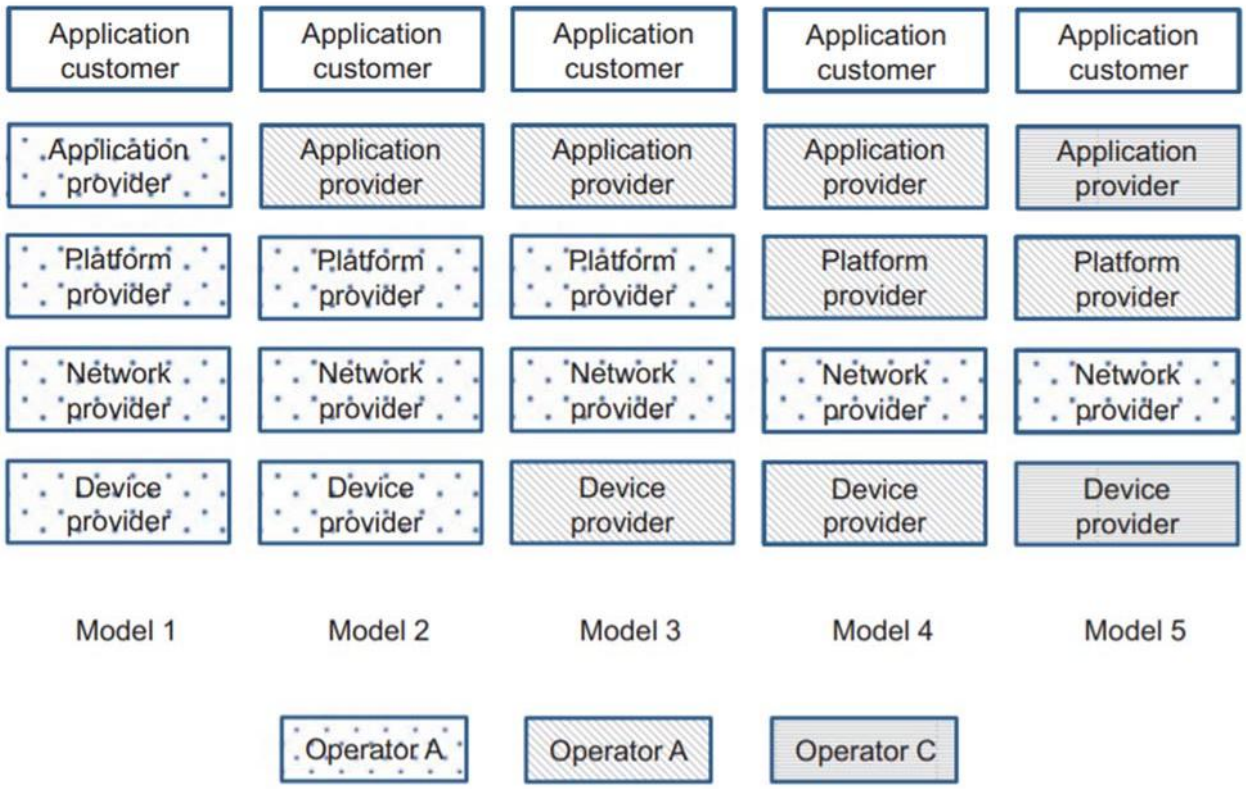



Рисунок 2.6 —Бізнес-моделі IoT, згідно з MCE

Індустріальний Інтернет-консорціум (IIC) орієнтується на подібні принципи та розробляє еталонну архітектуру IIoT, що має ряд схожих аспектів з підходом та моделлю MCE. Підхід IIC до створення архітектури включає комплексний підхід, який враховує інтереси та проблеми різних зацікавлених сторін, опираючись на випадки використання та фокусуючись на повних бізнес-моделях та додатках на всіх етапах, від пристроїв до послуг IIoT. IIC приймає концепцію, згідно з якою зацікавлені сторони, котрі потребують різних рішень, мають архітектурні точки зору, що відрізняються рівнем абстракції. Це дає можливість кожному учаснику зосереджуватися на найбільш актуальних для нього аспектах та створювати відповідні архітектури для досягнення своїх цілей і вирішення визначених проблем. Для цього IIC визначив чотири основні точки зору: (a) бізнесова, (b) користувацька, (c) функціональна та (d) впровадження.

Бізнесова точка зору орієнтована на проблеми зацікавлених сторін у бізнесі, які визначають та формулюють системи та послуги IIoT для своїх організацій або споживачів. Ці проблеми, зокрема, такі як рентабельність інвестицій, витрати на технічне обслуговування тощо, вирішуються через модель, яка дозволяє визначити бачення та цінності, що перетворюються на ключові цілі, а потім на бізнес-завдання високого рівня, що отримують назву фундаментальних можливостей. Серед зацікавлених сторін — бізнес-розробники, системні інженери та менеджери продуктів.

Точка зору використання фокусується на тому, як система виконує ключові цілі та можливості, що були визначені через бізнесову точку зору. Вона описується моделлю, що ідентифікує систему та її діяльність, учасників — людей чи машини — та їх ролі, а також завдання, які виконуються учасниками в певних ролях. Завдання, будучи діями в системі, описуються точно для кожної ролі за допомогою функціональних карт і карт реалізації, що визначають конкретні функції та підсистеми, необхідні для повного виконання завдання. До зацікавлених сторін, що беруть участь у точці зору використання, відносяться не лише системні інженери та менеджери продуктів, але й усі сторони, залучені в специфікацію системи та послуг IIoT, включаючи кінцевих споживачів.

Функціональна точка зору зосереджена на функціональній архітектурі системи IIoT, визначаючи її компоненти, взаємозв'язки та координацію, що відповідає вимогам і специфікаціям, розробленим через точку зору використання. До зацікавлених сторін, які беруть участь у цій точці зору, відносяться розробники систем та підсистем, розробники продуктів, менеджери, а також системні інтегратори. Враховуючи зростаюче впровадження індустріальних систем управління (ІСУ) в різних секторах і їх роль у керуванні критичною інфраструктурою, еталонна модель IIC зосереджена на інтеграції ІСУ з традиційними інформаційними технологіями (IT) в єдину, ефективну модель, яка задовольняє потреби всіх зацікавлених сторін, зазначених у бізнесових та користувацьких точках зору. Це дозволяє

приймати ефективні рішення. Однак включення ІСУ та ІТ в єдину модель породжує низку складних проблем.

Промислові системи управління та операційні технології (ОТ) розвивалися окремо від традиційних інформаційних технологій (ІТ) через різні цілі та вимоги, які характерні для них. Ці системи, зазвичай, повинні забезпечувати безперервну роботу, високу безпеку та здатність працювати в режимі реального часу. Системи ОТ розроблялись і експлуатувались інженерами, які використовують спеціалізовані технології для обробки даних, зв'язку та взаємодії з навколишнім середовищем через датчики і виконавчі механізми. Зазвичай ці системи управляються їх власниками і є частиною критично важливих інфраструктур з вимогами до надійності та безперебійної роботи.

На відміну від ІТ-систем, технології, практики та стандарти в ОТ еволюціонували окремо. Однак розвиток таких технологій, як датчики, виконавчі механізми, процесори і пам'ять, дозволив інтегрувати в ОТ більш складні функції, типові для ІТ-систем, такі як обробка великих обсягів даних, багатоваріантне моделювання і оптимізація. Водночас зростання складності ОТ-систем призвело до їх більшої вразливості до збоїв і кібератак, що ставить перед ними нові функціональні вимоги для забезпечення ефективної і безпечної роботи.

Для вирішення задачі інтеграції ІТ і ОТ в єдиній моделі, ІІС розробив еталонну архітектуру, яка розподіляє системи ІІоТ на п'ять основних доменів. Кожен з цих доменів об'єднує функціональні можливості, необхідні для забезпечення логічно чіткої роботи системи на високому рівні. До цих доменів належать: (а) контроль, (б) операції, (в) інформація, (г) додатки та (е) бізнес. На рисунку 2.7 [27] представлено розподіл функціональних компонентів системи ІІоТ на ці п'ять доменів, а також показані потоки даних і управління між ними, згідно з моделлю ІІС.

Домен управління ефективно відображає контур керування, який реалізують промислові системи управління. Цей домен включає датчики,

логічні блоки та механізми спрацьовування, які складають систему, що управляється однією або кількома промисловими системами управління. Домен операцій охоплює функції, необхідні для роботи цих систем в управлінській сфері. Операційні функції включають моніторинг та керування системами, а також оптимізацію їх ефективності, враховуючи вимоги до безперервної роботи, обробки в реальному часі та досягнення низької споживаної потужності для різних областей застосування.

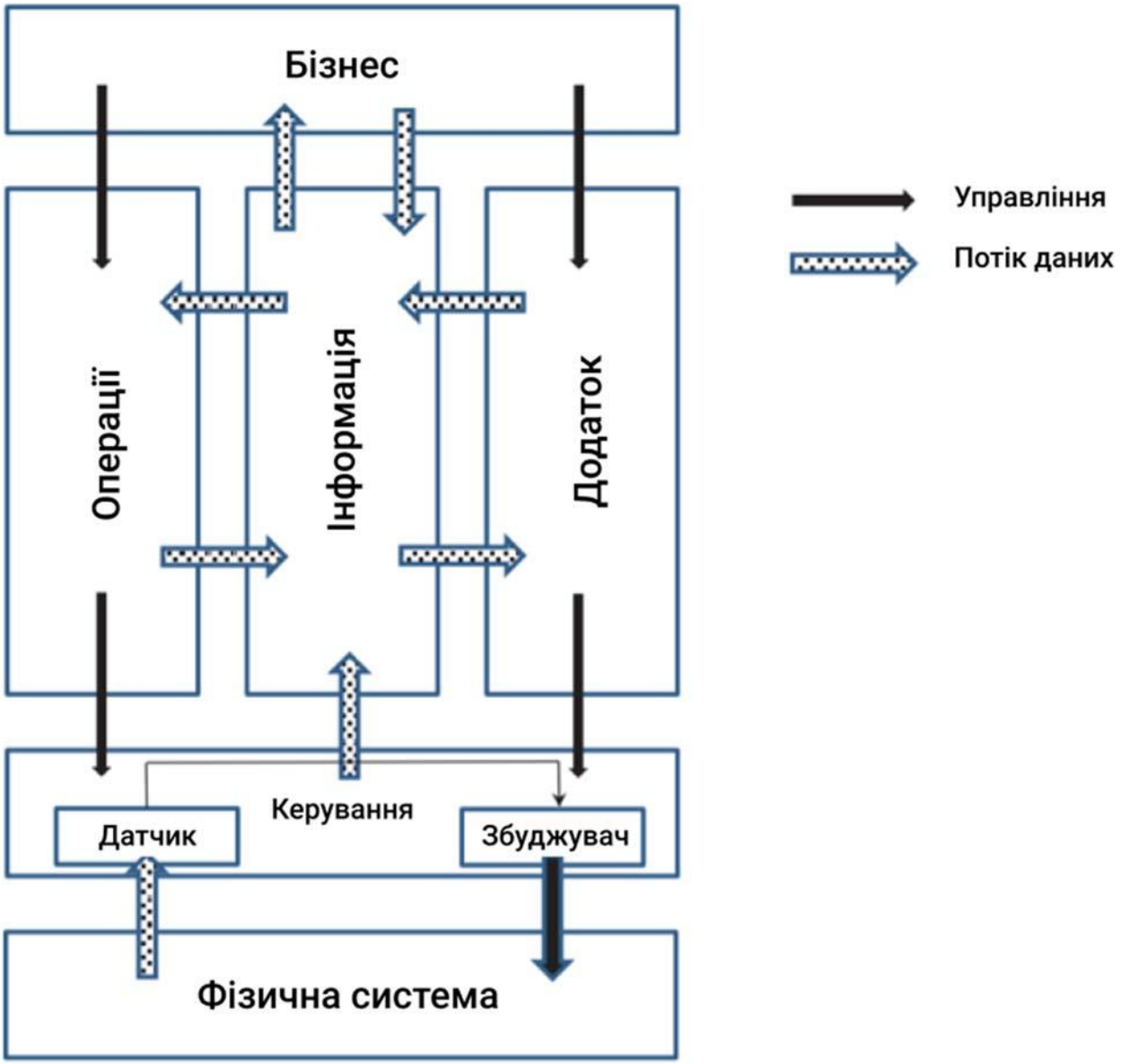



Рисунок2.7— Функціональні області еталонної архітектури IIC

Інформаційний домен відповідає за збір та аналіз даних з різних частин

системи для забезпечення прийняття високорівневих рішень. Це включає координацію та оптимізацію роботи різних промислових систем управління в межах управлінського домену. Домен додатків містить функції, специфічні для кожного застосування, зокрема моделі та правила, що визначають роботу програми. Важливою складовою цього домену є набір API та інтерфейсів, що дозволяє програмам і користувачам взаємодіяти з додатками. Домен бізнесу охоплює системи, які дозволяють приймати управлінські рішення на рівні бізнесу, такі як інтеграція з ERP-системами або MES для управління виробничими процесами.

Варто підкреслити, що підхід IIC орієнтований на концепцію контрольної установки, що означає розгляд усіх точок зору у межах циклу управління, який забезпечує роботу системи. Цикли управління можуть бути як простими (з однією системою), так і складними (з кількома системами, організованими в ієрархічну структуру). Функціональна декомпозиція доменів, яку пропонує IIC, застосовується на будь-якому рівні ієрархії. На відміну від шаруватого підходу ITU, модель IIC забезпечує логічний поділ функцій як усередині одного рівня, так і між різними рівнями ієрархії.

Однією з ключових особливостей архітектури IIC є визначення «перехресних функцій». Ці функції є ієрархічними або багаторівневими елементами IT-інфраструктури, необхідними для створення комплексних IIoT-додатків. До них належать підключення, розподілене управління даними, аналітика, гнучке управління ресурсами та інші функції, які залежать від конкретного випадку використання чи домену програми. Наприклад, функція підключення повинна бути ієрархічно організованою, дотримуючись стандартів і практик для з’єднання компонентів як у межах однієї промислової системи управління, так і між кількома системами, які можуть містити всі п’ять доменів.

Аналізуючи перехресні функції, можна помітити їх схожість із шаруватою архітектурою, яку пропонує ITU. У цьому контексті підходи IIC та ITU можна вважати взаємодоповнювальними. Архітектура IIC є певним узагальненням моделі ITU, оскільки включає перехресні функції, подібні до

рівнів ITU, але водночас дозволяє більш детально опрацьовувати функціональні можливості на різних рівнях. Це забезпечує підтримку повного циклу управління та враховує інтереси всіх зацікавлених сторін — від інженерів, що розробляють пристрої, до бізнес-менеджерів, які приймають стратегічні рішення.

Ця аналогія та взаємодоповнюваність стають ще більш наочними на етапі реалізації, який деталізує функціональну точку зору, розроблену для систем IIoT. Точка зору реалізації охоплює всі технічні та технологічні аспекти, необхідні для побудови повноцінної IIoT-системи та її застосування. Це включає функціональні можливості системи, технологічні вимоги, комунікаційні та мережеві протоколи, різноманітні типи інтерфейсів, а також відображення функціональних блоків, визначених у функціональній точці зору, на конкретні архітектурні рішення. Серед таких рішень можна виділити трирівневу архітектуру, що складається з рівнів краю (Edge), платформи та підприємства (Enterprise), а також архітектуру, побудовану на основі багатошарової шини даних.

## 2.3Моделювання IoT-мереж, що взаємодіють із зовнішнім середовищем

Розглянемо IoT-модель мережі, що фокусується на аналізі подій у системі. Оскільки події характеризуються тривалим життєвим циклом, ключову роль у цій моделі відіграє пам’ять, організована у вигляді черг.

Обчислювальна модель побудована на основі обробки подій з довгим життям. Подія створюється у певному вузлі, зберігається протягом певного часу до моменту її завершення, після чого обробляється або споживається іншим вузлом.

Подія визначається п’ятьма кортежами:

$$\langle \text{key}, \text{value}, \text{dest}, \text{gen\_time}, \text{release\_time} \rangle$$

Семантика події визначається у вигляді пар ключ-значення. Призначення події полягає у передачі до пристрою, що обробляє відповідну пару ключ-

значення. Важливо зазначити, що методи моделювання, представлені в цій роботі, не розглядають семантику самих пар ключ-значення.

Крім цього, для події необхідно враховувати часові характеристики, що задаються двома параметрами. Час генерації події $\vartheta$ відображає момент її створення. Це значення корисне для аналітики, однак його відстеження залежить від конкретної реалізації системи. Час випуску події $\rho$ визначає можливість виконання дії із затримкою — подія в IoT-середовищі може не викликати миттєву реакцію, а бути реалізованою пізніше. Різниця між часом створення та випуску визначає тривалість події:

$$\lambda = \rho - \vartheta.$$

Події можуть генеруватися як періодично, так і аперіодично. Аналогічно, час їх активації або випуску також може мати періодичний або аперіодичний характер.

Мережа складається із вузлів і з’єднань, де комунікаційні зв’язки моделюються як односпрямовані. Хоча більшість фізичних хабів підтримують повнодуплексний режим, у цьому дослідженні зв’язки розглядаються як односпрямовані для спрощення аналізу.

Існує три основні підходи до моделювання взаємодії системи IoT із навколишнім середовищем, кожен з яких характеризується певним рівнем точності та деталізації.

Найбільш проста модель розглядає пристрій і користувача як часові кінцеві автомати. Проходячи через автомат користувача, що описує конкретний сценарій використання, можна побудувати добуток автомата пристрою та маршруту автомата користувача. У результаті утворюється кінцевий автомат із часовими характеристиками, які визначаються сценарієм. Цей автомат показує, коли пристрій генерує події, що дозволяє створити трасування послідовності подій.

Покращена модель розглядає користувача як процес прийняття рішень

Маркова (MDP) з обмеженим часовим горизонтом. MDP є стохастичною моделлю, що використовується для оптимізації. Ця модель визначається набором станів, можливими діями для кожного стану та винагородами \( R \), які присвоюються діям. Ймовірності переходу між станами залежать від обраної дії. Для пошуку шляху, що максимізує винагороду, можна застосувати різні алгоритми, такі як динамічне програмування або методи лінійної алгебри. У цьому підході обчислюється оптимальний шлях з точки зору винагороди, а потім його об'єднують із моделлю пристрою, враховуючи фіксовану часову структуру. Приклад простої моделі пристрою та користувача наведено на рисунку 2.8.

Модель пристрою об'єднує всі дії компонентних пристроїв, що беруть участь у виконанні певної рутини, в один автомат для спрощення аналізу. Дії у користувацькій моделі MDP відповідають станам моделі пристрою.

Більш складний підхід передбачає використання безперервного процесу прийняття рішень Маркова (CTMDP). У цій моделі переходи між станами відбуваються у безперервному часі, а їх тривалість найчастіше моделюється як процес Пуассона [28]. Для визначення оптимального шляху у CTMDP можна застосовувати стандартні методи MDP, адаптуючи їх до часу переходів, заданих через розподіл Пуассона.

Подія не обов’язково має зберігатися на концентраторі, який є джерелом або кінцевим призначенням події. У системах з декількома концентраторами події можна розміщувати на віддалених концентраторах для запобігання переповненню черги або для продовження автономної роботи вузла. Якщо подія зберігається на віддаленому концентраторі, необхідно враховувати час її передачі, щоб забезпечити своєчасне надходження до пристрою призначення.

Для спрощення, припустимо, що витрати енергії на передачу подій через мережу відсутні. Позначимо Ре — як енергію, яку витрачається на збереження однієї події в пам'яті протягом одиниці часу. Якщо ми маємо набір подій П, то загальна енергія, необхідна для зберігання всіх подій у системі до моменту їх випуску, обчислюється як:

$$E_{\text{pop}} = \sum_{e \epsilon \Pi} \left[ \rho(e) - \vartheta(e) \right] P_e$$

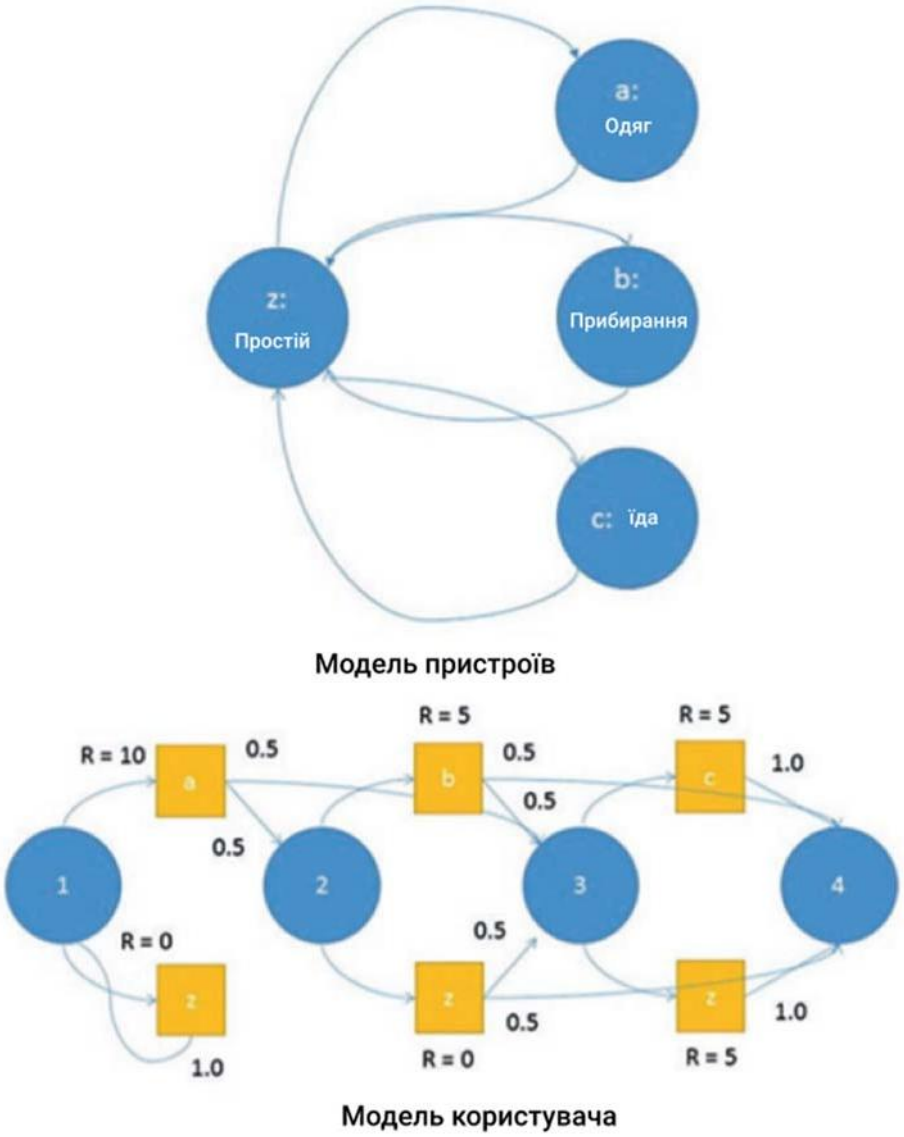



Рисунок 2.8 — Проста модель пристрою та користувача

Маємо набір концентрацій H, кожен з яких має певну кількість енергії батареї Eh(i) . Потрібно визначити оптимальний розподіл подій між цими концентраторами.

$$\forall i \epsilon H : E_{\text{pop}}(i) \leq E_{\text{h}}(i)$$

Це типова задача упаковки, яка розв'язується як розподілена проблема без

централізованого переліку подій. В реальних умовах енергія передачі обмежує можливі варіанти розподілу подій.

Пропонується наступний евристичний алгоритм для переміщення подій:

— створюється часткове впорядкування концентраторів так, щоб два сусідні концентратори не потрапляли до одного набору, при цьому всі концентратори мають бути охоплені;

— концентратори переміщуються в порядку, що виключає одночасне розвантаження суміжних концентраторів;

— кожен концентратор вивантажує необхідну кількість подій для забезпечення акумулятора, переміщаючи події до сусідніх концентраторів з найбільшим доступним запасом енергії.

# 3 МЕТОД ТА ЗАСОБИ ОПРАЦЮВАННЯ ІНФРМАЦІЇ ІНТЕРНЕТУ РЕЧЕЙ

## **3.1** Метод збору та аналізу даних пристроїв IoT

Потокова передача даних в IoT. Кожна подія в IoT генерує дані. Процес передачі даних передбачає використання стандартних протоколів, таких як MQTT, WAMP, HTTP, CoAP або Sigfox, кожен з яких має свої переваги та відповідні сценарії застосування. Ці протоколи забезпечують передачу оновлень або іншої інформації від пристроїв IoT до централізованої платформи для подальшої обробки.

На цьому етапі важливо визначити, як будуть агрегуватися та зберігатися дані для майбутнього використання. Подальші дії залежать від того, як ці дані будуть використовуватися. Тут обирається підхід, який дозволить вирішити, чи слід передавати дані в режимі реального часу або в пакетах. Також визначається, в якому порядку повинні формуватися дані для забезпечення максимально точного аналізу.

Зберігання даних IoT. Використання даних у реальному часі забезпечує максимальну точність, оскільки гарантує доступ до всіх даних, що генеруються кожним пристроєм IoT. Проте цей підхід часто веде до обробки величезних обсягів вхідної інформації. Однією з проблем є правильне віднесення позначок часу до даних, що надходять від кількох пристроїв IoT. У цьому контексті необхідно розглядати системи, які можуть ефективно обробляти швидкість та обсяг цих вхідних даних для забезпечення безперервної та точнішої роботи.

Вибір методу збору та аналізу даних IoT у режимі реального часу повинен бути підкріплений обґрунтованими доводами. Інакше, великі обсяги даних можуть негативно вплинути на хмарні системи, навантажуючи мережеві та обчислювальні ресурси, необхідні для їх обробки.

Також важливо враховувати специфіку конкретних застосунків IoT, оскільки вони можуть мати різні вимоги до затримки, енергоспоживання та точності. Деякі програми допускають певну затримку, тоді як інші, зокрема

безпекові застосунки, є критично важливими для часу і не можуть дозволяти жодних затримок.

У багатьох випадках високої точності не вимагається, і можна надсилати дані групами. Якщо дані передаються пакетами або мікропакетами, це дозволяє зберігати всю необхідну інформацію, хоча і з певною затримкою, що визначена заздалегідь. Однак вибір конкретного підходу залежить від вимог конкретного випадку використання. У деяких ситуаціях для аналізу потрібні точні дані в реальному часі, в інших же історичні дані можуть бути достатніми.

Платформовий підхід. Якщо ви плануєте безпосередньо працювати з вхідними даними, на цьому етапі необхідно вибрати платформу, яка здатна обробляти великі обсяги даних, що надходять з численних пристроїв IoT у реальному часі. Така платформа повинна мати можливість реагувати на тимчасові проблеми з підключенням, зокрема втрачені з'єднання, перебої в роботі чи збої серверів. Вона дозволяє мінімізувати ризик втрати даних, забезпечуючи збереження їх у доступній та зручній для подальшого моделювання та аналізу формі. Це дозволяє гарантувати точність результатів, отриманих на основі зібраних даних.

Цільові підходи до збору даних. Вибір стратегії збору та зберігання даних IoT значною мірою залежить від специфічних вимог кожного конкретного випадку використання. Ці підходи охоплюють, але не обмежуються, процедури збору даних, що враховують такі аспекти, як точність даних, енергоспоживання, час відгуку та захист конфіденційності. Одним із поширених методів для зменшення обсягів зібраних даних є агрегування, фільтрація, інтерпретація та стиснення даних безпосередньо на рівні датчика або пристрою IoT, максимально наближаючи ці операції до джерела даних. Далі ми зібрали кілька основних підходів до збору даних, які залежать від прогнозованого застосування, описаного в цьому дослідженні.

Стратегії, орієнтовані на точність. Ця стратегія збору даних IoT передбачає балансування між частотою запитів на вимірювання та необхідною точністю даних. Вона ґрунтується на адаптації частоти вимірювань в

залежності від рівня точності, яка потрібна для конкретного випадку використання. У деяких сценаріях досягання надмірної точності не дає значних переваг і не виправдовує додаткових зусиль. Тому в межах цієї стратегії зменшується частота вимірювань, що дозволяє зменшити витрати ресурсів на збір даних, при цьому зберігаючи оптимальний рівень точності, який був визначений для конкретного застосування.

Стратегії, орієнтовані на критичність часу. Цей підхід до збору даних IoT передбачає заздалегідь визначену максимальну допустиму затримку. Час, що минув з моменту останнього вимірювання, повинен залишатися в межах цієї затримки. У випадках, де важливими є вимоги до часу, кожне нове вимірювання повинно здійснюватися в межах визначеного максимального часу від останньої позначки часу. Таким чином, час, що пройшов від останнього вимірювання, визначає «свіжість» даних.

Стратегії збору даних IoT, орієнтовані на енергоспоживання. У цьому підході головним чинником є ефективне використання енергії. Завдання полягає в досягненні потрібної точності при мінімізації енергетичних витрат. Стратегія збору даних, спрямована на економію енергії, прагне досягти найвищої ефективності. Оцінка такої стратегії проводиться шляхом вимірювання різниці між корисністю, що досягається для заданої точності даних, та енергоспоживанням, необхідним для отримання цієї точності. Як і в стратегії, орієнтованій на точність, передбачається, що для кожного конкретного застосування буде визначена оптимальна точність, причому підвищення точності не дасть значної додаткової вигоди.

Стратегії збору даних IoT, орієнтовані на конфіденційність. Якщо захист конфіденційності кінцевих користувачів є пріоритетним, застосовується такий підхід. В цьому випадку зусилля спрямовані на зменшення кількості запитів до датчиків для збору даних і на додавання шуму до результатів за допомогою методів захисту конфіденційності. Основна мета — забезпечити захист конфіденційності користувачів шляхом модифікації точності окремих вимірювань, водночас зберігаючи необхідний рівень "адекватної точності" для

загальних результатів.

Додавання "шуму" до результатів здійснюється через метод "різної конфіденційності". Це дозволяє збирати дані, що стосуються групи користувачів, не розкриваючи при цьому особисту інформацію. Такий підхід включає додавання "шуму" безпосередньо під час процесу збору даних, що не призводить до значних змін у статистичних результатах. Це рішення сприяє забезпеченню конфіденційності на етапі збору даних IoT, а не обробки. Експериментальні дослідження підтверджують, що таке втручання має мінімальний вплив на кінцеві статистичні показники.

Налаштування через платформу Record Evolution пропонує різноманітні підходи до збору та аналізу даних IoT. Платформа надає студії обробки даних можливість підключатися до джерел даних IoT, передаючи їх безпосередньо в зручне та компактне сховище. Вона дозволяє точно налаштувати процес збору даних, забезпечуючи постійний моніторинг та управління потоками даних, що дозволяє ефективно контролювати весь процес збору та аналізу інформації.

## **3.2** Метод організації інфраструктури IoT

Інфраструктура IoT. Для досягнення максимального ефекту від інвестицій в IoT організаціям важливо зосередитися на семи основних компонентах інфраструктури. Якщо кожен із цих компонентів буде належним чином розроблений і стандартизований, це дозволить повною мірою використовувати потенціал IoT. Мережі IoT. Мережі забезпечують з’єднання між датчиками та виконавчими механізмами IoT. Датчики використовуються для вимірювання різних параметрів, таких як температура, вологість, рух і тиск, а виконавчі механізми — для виконання дій, таких як управління контролерами, роботизованими системами або безпілотними літальними апаратами. Зазвичай для з’єднання використовуються бездротові мережі. Датчики та виконавчі механізми можуть працювати в окремих мережах або спільно в одній, залежно від конкретного застосування IoT. Для забезпечення зв'язку використовуються різні технології, такі як Wi-Fi, Bluetooth, а також бездротові мережі 4G або 5G.

Правильне проектування мережі та вибір відповідної технології є критично важливими для успішного впровадження IoT.

На рисунку 3.1 представлена класична архітектура IoT.


Рисунок 3.1 — Класична архітектура IoT

Шина зв'язку IoT. У багатьох випадках інфраструктура IoT передбачає використання шини для з'єднання мереж з платформами аналітики та агрегування даних. Шина забезпечує централізовану точку з'єднання, що спрощує комунікацію. Замість того, щоб підтримувати численні окремі з'єднання з платформами агрегації, організації використовують одну загальну шину для ефективного управління та захисту всього трафіку IoT.

Платформа аналітики та агрегації. Дані з датчиків і виконавчих механізмів збираються та часто обробляються на крайніх обчислювальних пристроях або хмарних платформах. В деяких випадках для управління датчиками та виконавчими механізмами використовуються окремі платформи, що дозволяє більш ефективно організувати обробку та аналіз даних в залежності від конкретних потреб IoT-системи.

Мережа платформ IoT. Це мережа, що з'єднує платформи аналітики та агрегації з іншими платформами аналітики та візуалізації, які можуть бути, але не обов'язково, хмарними. Такі мережі платформ зазвичай використовують послуги дротового підключення з низькою затримкою, а також технології, як-от

програмно визначена глобальна мережа, оператори обміну WAN і прямі з'єднання з хмарними сервісами.

Платформа агрегації та візуалізації. Це хмарна платформа, що містить озера даних та аналітичні інструменти. Вона зазвичай включає в себе різноманітні інструменти для візуалізації, управління даними та проведення аналітичних операцій.

Система управління та контролю. Ця панель управління надає технологам підприємств повний огляд компонентів IoT та їх взаємодії з мережами та платформами.

Кібербезпека. Всі елементи інфраструктури Інтернету речей повинні бути інтегровані в платформу та структуру кібербезпеки, що відповідає вимогам усіх застосувань IoT, і є частиною загальної системи кібербезпеки організації.

Інші важливі компоненти, що часто реалізуються на платформах агрегації, аналітики та візуалізації, включають програмне забезпечення для управління базами даних IoT та аналізу даних, інтеграційне програмне забезпечення, яке зв'язує інфраструктуру IoT з існуючими ERP-системами та системами для виконання виробничих процесів, а також будь-які програми, що застосовуються до загальних або більшості ініціатив IoT в організації.

Система IoT є гібридною архітектурою, що означає наявність різних архітектурних підсистем. В основному, системи IoT побудовані на двох типах управління: управлінні подіями та управлінні часом.

Сенсори, що працюють за принципом подій, передають дані лише коли виявляють зміни в навколишньому середовищі, наприклад, спрацьовує сигнал тривоги, коли двері відкриваються вночі. У свою чергу, архітектура, заснована на часі, передбачає безперервну передачу даних протягом визначеного періоду (наприклад, датчики системи клімат-контролю можуть вимірювати температуру кожну секунду). Такі сенсори зазвичай працюють циклічно після налаштованих пауз, тривалість яких можна регулювати як на рівні окремих пристроїв, так і через центральну систему управління, яка через певний час ініціюватиме запити до кінцевих точок і датчиків [29]. Архітектура системи

опрацювання інформації IoT, що реалізує запропонований метод опрацювання даних зображений на рисунку 3.2.

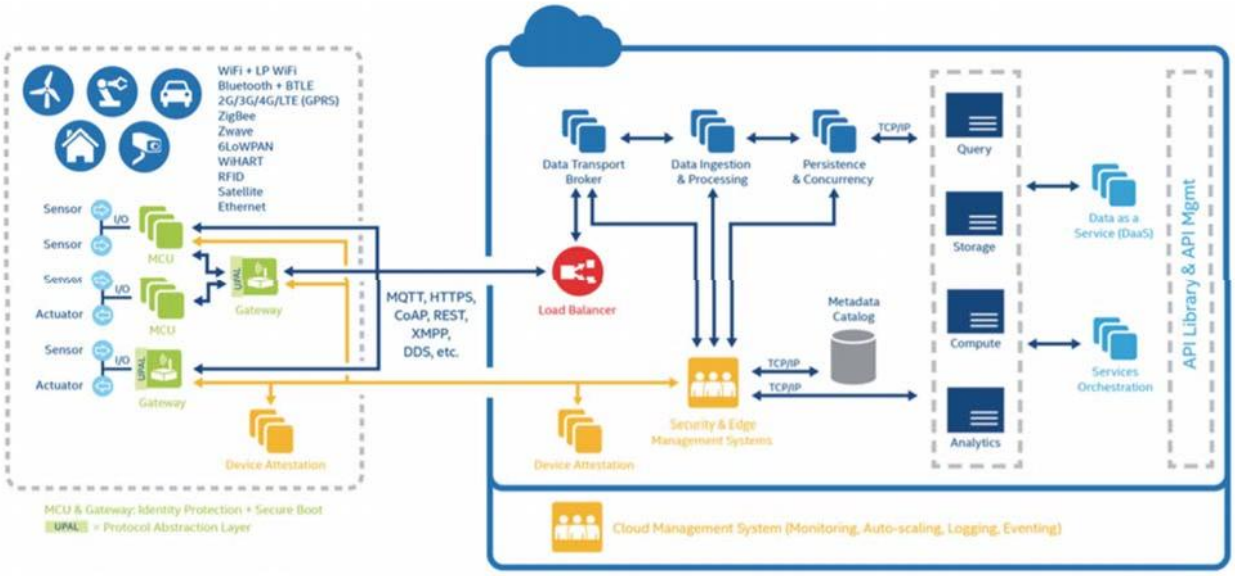



Рисунок 3.2 — Архітектура системи опрацювання інформації IoT

Мережеву архітектуру можна поділити на три основні топології: точка-точка, зірка та сітка.

Топологія "точка-точка" передбачає встановлення прямого з'єднання між двома станціями для передачі даних [30]. Основною перевагою цієї топології є її простота, що забезпечує низьку вартість з'єднання. Однак, вона має й обмеження, оскільки не дозволяє підключати пристрої поза цією мережею, що знижує її гнучкість.

Топологія "зірка" складається з численних кінцевих вузлів, що з'єднані з єдиним центральним концентратором. Всі вузли можуть обмінюватися даними, але лише через цей центральний вузол. Така архітектура забезпечує низьку затримку, високу пропускну здатність і підвищену надійність, оскільки відмова одного з вузлів не впливає на роботу інших. Однак, порівняно з топологією "точка-точка", мережа зіркового типу обмежена залежністю від центрального концентратора: якщо він виходить з ладу, вся мережа може бути відключена від зовнішнього середовища.

Меш-мережа є децентралізованою топологією, що використовує одну з

двох схем підключення: повну або часткову сітку. У повній сітці кожен вузол підключений до всіх інших, забезпечуючи максимальну кількість з'єднань. У частковій топології лише деякі вузли мають з'єднання з усіма іншими, тоді як решта з'єднані лише з тими вузлами, з якими вони обмінюються найбільше даних. Меш-мережа підходить для застосувань, що потребують великого покриття та високої пропускної здатності, дозволяючи створювати мережі дуже великого розміру. Однак її складність у порівнянні з топологіями «точка-точка» та «зірка» може призвести до збільшення затримки, витрат та потенційних технічних проблем у мережі.

Передача та зберігання даних у динамічних мережах IoT є складними та важливими завданнями. IoT-системи можуть включати величезну кількість пристроїв, що інтегровані в локальні та глобальні мережі, як фізичні, так і бездротові. Ці автоматизовані пристрої та датчики генерують великі обсяги даних у реальному часі, які не матимуть значення без належної фільтрації та обробки.

Передача даних забезпечується різними мережевими протоколами — набором правил, що визначають семантику та синтаксис обміну даними між функціональними блоками комп'ютерної мережі. У комп'ютеризованих мережах, побудованих згідно з вимогами архітектури відкритої системи, протокол регламентує поведінку сутностей на кожному рівні під час передачі даних [32].

Створення мережі IoT є складним завданням, особливо коли мова йде про датчики, які не можуть бути включені до глобальної схеми адресації, що ускладнює їх інтеграцію в повноцінний вузол. Традиційні протоколи IP не завжди є оптимальними для обміну даними в таких мережах. Крім того, вузли IoT зазвичай залежать від постійних джерел енергії, пропускної здатності мережі та вимог до зберігання даних, що ускладнює управління ресурсами.

У разі бездротових датчиків потрібно інтегрувати радіосигнал у мережу, що додає додаткові технічні вимоги. Спочатку зібрана інформація зберігається в резервуарах даних, а потім передається до інших вузлів мережі. Важливо

правильно вибрати стратегію передачі даних між датчиками та кінцевими пристроями, враховуючи їхнє розташування та конфігурацію. Це може значно покращити пропускну здатність мережі IoT, зменшити енергетичні витрати та уникнути дублювання передачі одних і тих самих даних [34], [35].

Передача даних у мережах IoT є складним процесом, який може потребувати значних мережевих ресурсів для забезпечення ефективного обміну інформацією. Характер даних, що передаються, може варіюватися залежно від типу пристрою та вибраних протоколів передачі. Наприклад, стандарт ISO 8583, який використовується для формування повідомлень платіжних терміналів, створює рядок даних, що представляє платіжну транзакцію користувача (рис. 3.3).

```
02004200040000000002161234567890123456060917303012345678
9ABC10001234567890123456789012345678901234567890123456789
0123456789012345678901234567890123456789012345678901234567
89
```

Рисунок 3.3 — Повідомлення формату ISO 8583

Коротке повідомлення містить всі необхідні дані про платіжну картку, термінал та фінансову частину транзакції. Проте без додаткових інструкцій ані людина, ані комп'ютер не здатні використати цю інформацію. Власники систем стикаються з вибором: використовувати компактні повідомлення в середовищі IoT, що потребують більше ресурсів для дешифрування, або передавати повніші повідомлення, що можуть призвести до перевантаження мережі. Зашифровані повідомлення надійні, але їхня інформація не може бути збережена або використана для аналізу в початковому вигляді. Для більш ефективної передачі даних необхідно розробити стратегію, яка працює з інформацією, готовою до використання, як, наприклад, новий стандарт платіжних терміналів ISO 20022, заснований на розширюваній мові розмітки XML. Цей стандарт, орієнтований на мережеві документи, значно полегшує обробку та аналіз даних. Рисунок 3.4 демонструє приклад повідомлення ISO20022.

```
<CdtTrfTxInf>
  <IntrBkSttlmAmt Ccy='USD'>12500</IntrBkSttlmAmt>
  <IntrBkSttlmDt>2009-10-29</IntrBkSttlmDt>
  <Dbtr>
    <Nm>ACME NV.</Nm>
    <PstlAdr>
      <StrtNm>Amstel</StrtNm>
      <BldgNb>344</BldgNb>
      <TwnNm>Amsterdam</TwnNm>
      <Ctry>NL</Ctry>
    </PstlAdr>
  </Dbtr>
  <DbtrAcct>
    <Id>
      <Othr>
        <Id>8754219990</Id>
      </Othr>
    </Id>
  </DbtrAcct>
  <DbtrAgt>
    <FinInstnId>
      <BIC>EXABNL2U</BIC>
    </FinInstnId>
  </DbtrAgt>
</CdtTrfTxInf>
```

Рисунок 3.4 — Повідомлення формату ISO 20022

Неоднорідність пристроїв IoT є однією з основних характеристик, що водночас є і перевагою, і слабким місцем цієї технології. Залежно від складності архітектура IoT може включати кілька рівнів пристроїв, кожен з яких має свою спеціалізацію для виконання певних функцій. Різниця між рівнями полягає не лише у використовуваних протоколах, але й в організації пристроїв, що безпосередньо впливає на використання обчислювальних ресурсів і обсяг переданих даних. Архітектуру IoT можна поділити на кілька рівнів, де найнижчий рівень, що характеризується високою неоднорідністю, включає пристрої та датчики кінцевих точок, які взаємодіють із зовнішнім середовищем. З поступовим підвищенням рівня складності переходять до більш просунутих маршрутизуючих і обчислювальних пристроїв (див. рис. 3.5). В оптимізованих архітектурах кожен наступний рівень повинен містити менше пристроїв.


Рисунок 3.5 — Графік неоднорідності IoT

Неоднорідність IoT полягає у використанні різноманітних пристроїв, таких як персональні пристрої користувачів, датчики, маршрутизатори, комутатори, концентратори, бази даних, обчислювальні сервери тощо. Кожен пристрій в IoT має свою конкретну роль та виконує лише необхідні функції, щоб не перевантажувати систему. Однак багато пристроїв виконують кілька функцій одночасно, які можуть бути схожими або навіть взаємодоповнюючими, що дозволяє їм заміщувати одне одного в разі потреби. Згрупувавши за функціональністю, всі пристрої IoT можна поділити на три основні категорії: пристрої кінцевих точок, пристрої введення-виведення та обчислювальні пристрої. Пристрої кінцевих точок взаємодіють з зовнішнім середовищем або основним блоком і генерують дані на основі отриманих команд або сигналів. Зазвичай ці пристрої важко налаштувати, і їх

програмування вимагає специфічних знань. Пристрої вводу-виводу, через обмеженість обчислювальних ресурсів, зазвичай виконують роль посередників між кінцевими пристроями та потужнішими обчислювальними системами. Остання категорія пристроїв займається ресурсомісткими завданнями, такими як фільтрація та обробка даних. Всі три типи пристроїв можуть бути окремими фізичними елементами в мережі IoT або інтегровані як логічні вузли в одному пристрої, в залежності від складності системи IoT та особливостей самої технології.

## 3.3 Засоби опрацювання інформації IoT

Засоби опрацювання інформації в системах IoT включають апаратні та програмні компоненти, які працюють разом для збору, обробки, аналізу та передачі даних. Завдяки цьому забезпечується ефективне функціонування IoT-систем у різних сферах діяльності.

Основу апаратної частини IoT становлять сенсори, мікроконтролери та пристрої зберігання даних. Сенсори є ключовими елементами, що забезпечують збір даних про фізичні параметри середовища, такі як температура, вологість, освітлення чи тиск. Вони передають дані на мікроконтролери або інші обчислювальні пристрої. Мікроконтролери, такі як Arduino, Raspberry Pi або ESP32, виконують попередню обробку зібраної інформації та передають її далі для аналізу.

Також важливу роль відіграють пристрої зберігання даних, що дозволяють тимчасово накопичувати інформацію для подальшої обробки. В IoT-системах використовуються компактні сховища, такі як SD-карти чи вбудована пам’ять, а також хмарні сервіси для довгострокового збереження великих обсягів даних.

Програмні компоненти є критично важливими для обробки інформації IoT. Вони включають платформи для збору та аналізу даних, протоколи передачі інформації та алгоритми для обробки сигналів.

Для обміну даними в IoT-системах застосовуються різноманітні

протоколи, такі як MQTT, CoAP та HTTP. MQTT (Message Queuing Telemetry Transport) є популярним протоколом завдяки своїй легковаговості та надійності при передачі даних у мережах з обмеженими ресурсами. CoAP (Constrained Application Protocol) оптимізований для пристроїв з низьким енергоспоживанням і часто використовується у сенсорних мережах.

Обробка даних відбувається за допомогою аналітичних інструментів і алгоритмів. Для великих обсягів даних використовуються платформи на основі штучного інтелекту та машинного навчання. Зокрема, системи прогнозної аналітики дозволяють виявляти закономірності та аномалії у зібраних даних, що важливо для автоматизації процесів у IoT-середовищі.

Хмарні технології та розподілені обчислення

Значна частина обробки IoT-даних виконується у хмарних обчислювальних системах. Хмарні платформи, такі як Amazon Web Services (AWS), Microsoft Azure чи Google Cloud, надають інструменти для зберігання та обробки великих обсягів інформації. Вони забезпечують високу продуктивність, масштабованість і доступність оброблених даних.

Крім того, важливу роль відіграють технології "edge computing" — розподілені обчислення, де обробка даних виконується на периферії мережі, близько до сенсорів. Це дозволяє зменшити затримки у передачі даних та знизити навантаження на центральний сервер.

Для зручного аналізу та інтерпретації результатів обробки інформації застосовуються інструменти візуалізації. Вони перетворюють отримані дані у графіки, діаграми та інші форми для зручного перегляду. Популярні інструменти, такі як Grafana та Power BI, дозволяють створювати інтуїтивно зрозумілі звіти та дашборди для моніторингу IoT-систем у реальному часі.

Таким чином, засоби опрацювання інформації для IoT охоплюють широкий спектр апаратних і програмних рішень. Вони забезпечують ефективний збір, обробку та аналіз даних, що є необхідним для функціонування IoT-систем у сучасному технологічному середовищі.

Провівши дослідження існуючих рішень, та, для реалізації

запропонованого методу опрацювання даних, що є основою побудови архітектури системи, доцільно обрати апаратно-програмні рішення. Вибір проводився серед найбільш популярних мікроконтролерів та мікропроцесорів таких, як ESP8266 (рис. 3.6, справа), ESP32(рис. 3.6, зліва), STM32 (див. рис. 3.7), а також процесорів ARM (див. рис. 3.8), які поєднують компактні розміри, низьке енергоспоживання та достатню обчислювальну потужність.

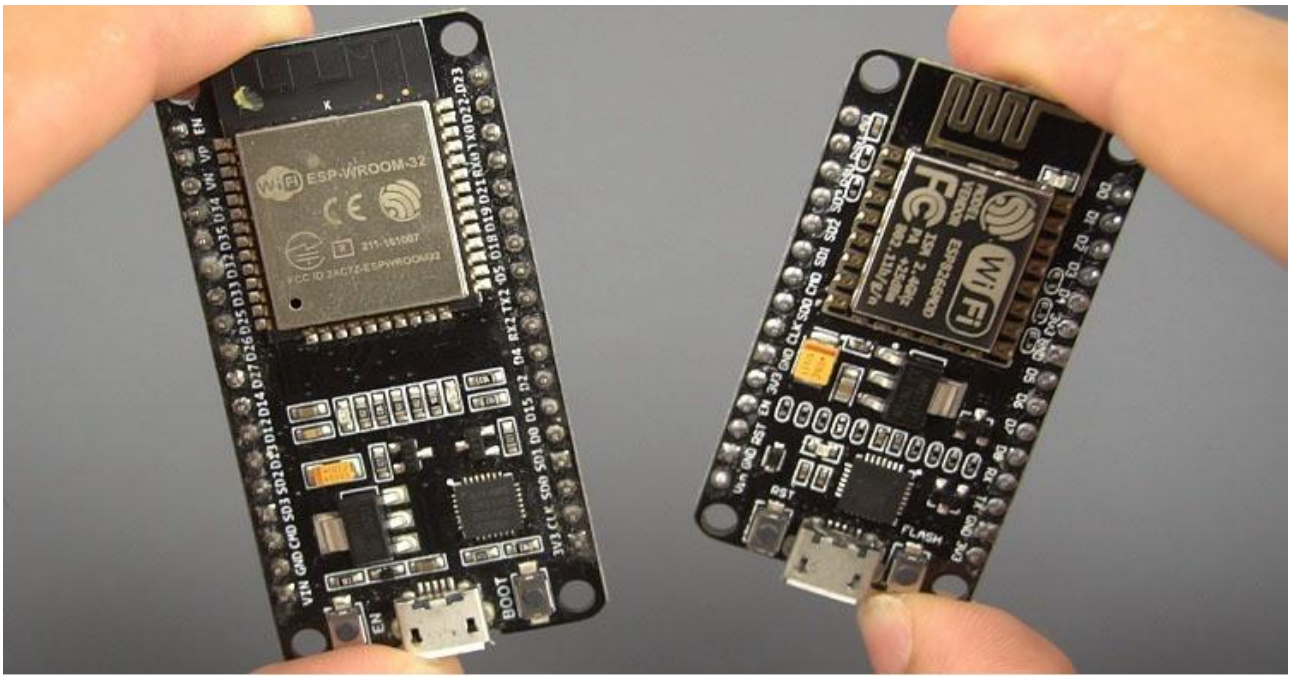



Рисунок 3.6 — Мікроконтролери серії ESP

Рисунок 3.7 — Мікроконтролери серії STM32

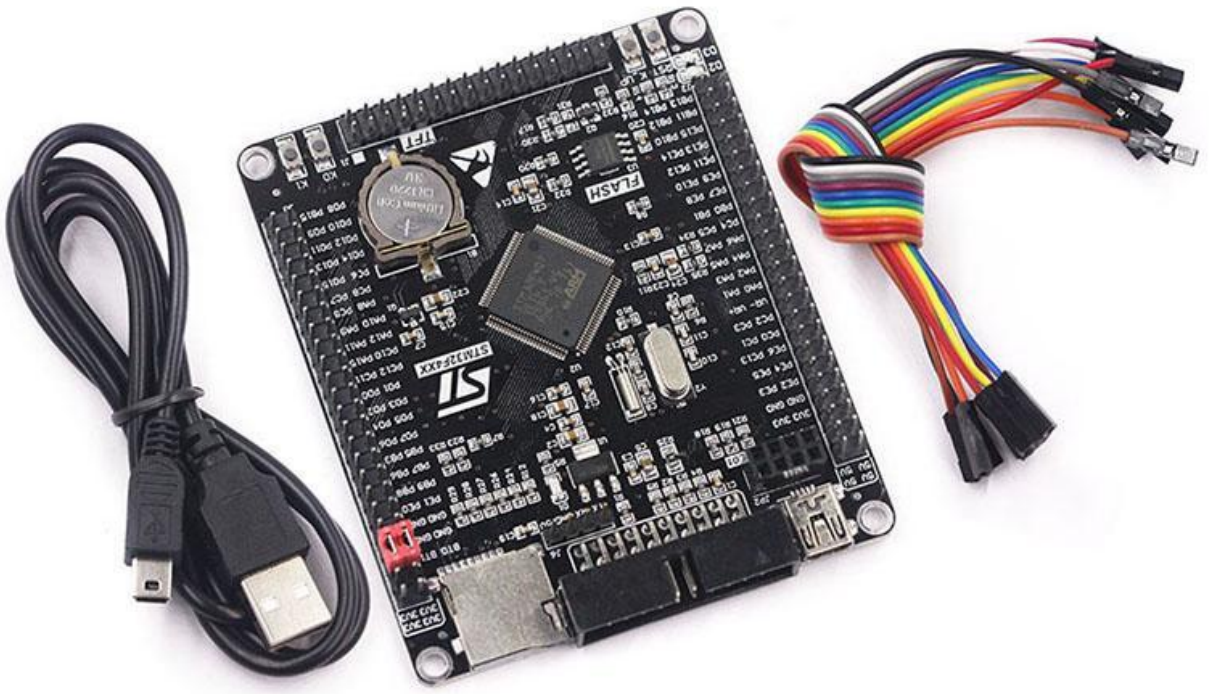


Рисунок 3.8 — Відлагоджувальна плата на процесорі ARM

Також враховувались задачі по можливості роботи з периферійними пристроями (датчики та сенсори). Ці компоненти відповідають за вимірювання фізичних величин (температури, вологості, тиску, рівня освітленості тощо) і передають дані на мікроконтролер. Також не залишилось осторонь питання вбудованих систем — компактних апаратно-програмних рішень, що можуть обробляти дані на рівні пристрою без передачі у хмару.

Найбільш вдалим варіантом є мінікомп’ютер Raspberry Pi, що часто використовується для локальної обробки інформації. Найсучаснішим варіантом є Raspberry Pi версії 5. Зовнішній вигляд показано на рисунку 3.9.

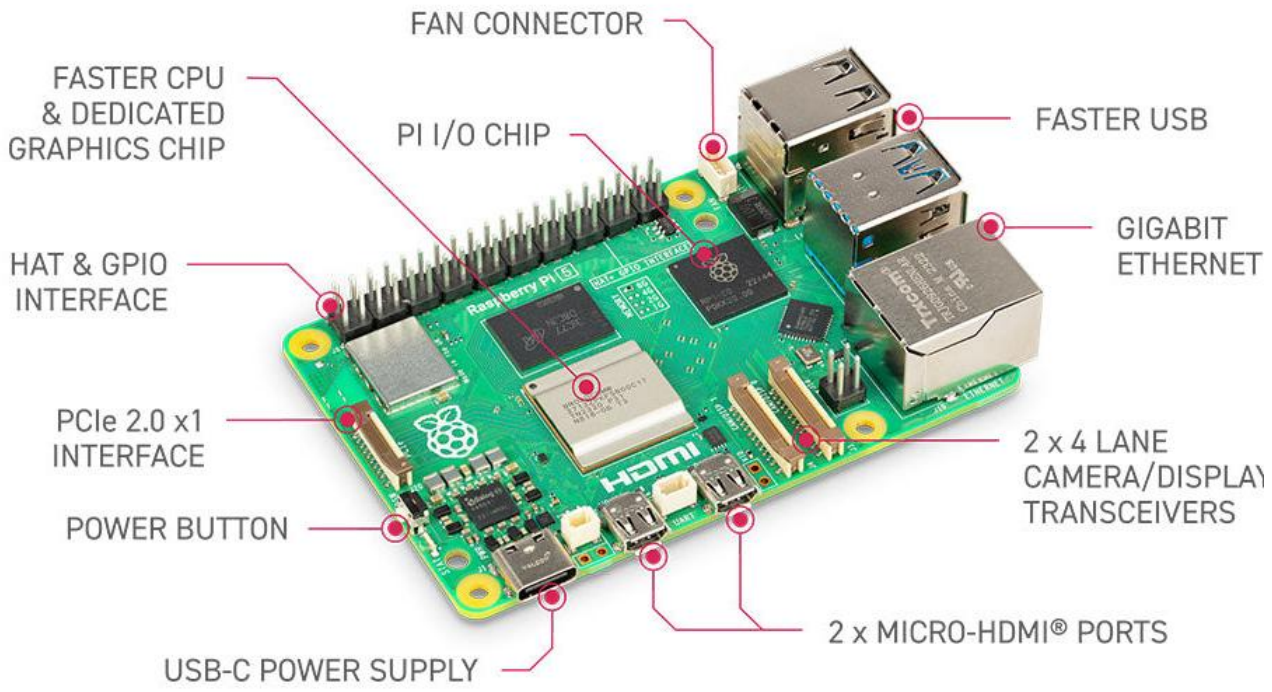



Рисунок 3.9 — Мінікомп’ютер Raspberry Pi 5

Таблиця 3.1 — Технічні характеристики Raspberry Pi 5

| | |
|---|---|
| Країна-виробник | Велика Британія |
| RAM | 8 GB |
| Процесор | Broadcom BCM2712 |
| Кількість ядер процесора | 4 |
| Частота процесора | 2.4 ГГц |
| Живлення | 5V 5A |
| Бездротові можливості | Bluetooth |
| GPU | VideoCore VII |
| GPIO контактів | 40 pins |
| Інтерфейси | RJ45, USB, microHDMI |
| Мережеві можливості | Ethernet 10/100/1000,<br>WiFi 2.4G/5.8G 300 mb/s |
| Сховище eMMC | microSD |
| USB порти | 4 |
| Країна реєстрації бренду | Велика Британія |
| Гарантія | 12 місяців |

Raspberry Pi 5 побудовано з використанням контролера вводу-виводу RP1. USB 3 має більшу загальну пропускну здатність для набагато більшої швидкості передачі. Роз’єми для камери та дисплея DSI взаємозамінні.

Raspberry Pi 5 оснащений чотирьохядерним процесором Broadcom BCM2712 Arm Cortex A76 з частотою 2,4 ГГц, що робить його втричі швидшим за попереднє покоління. З варіантами оперативної пам’яті до 8 ГБ це найшвидший Raspberry Pi.

Raspberry Pi 5 швидший і потужніший, ніж Raspberry Pi попереднього покоління, і, як і більшість комп’ютерів загального призначення, він працюватиме найкраще з активним охолодженням.

Для реалізації завдання розглянемо ситуацію, коли потрібно визначити спосіб організації зв'язку між двома модулями. Основним середовищем для передачі даних буде використовуватись мережа Wi-Fi. Як протокол для обміну даними обрано MQTT.

Для роботи з протоколом MQTT необхідно розгорнути сервер. В якості сервера планується використати плату Raspberry Pi 5. Для її функціонування необхідно встановити операційну систему.

Зазвичай на Raspberry Pi застосовується спеціалізована версія Linux під назвою Raspberry Pi OS [1]. Щоб встановити цю операційну систему на плату, потрібно скористатися утилітою Raspberry Pi Imager. Інтерфейс даної утиліти представлено на рисунку (див. рис. 3.10).

Оскільки операційна система встановлюється на зовнішній носій пам’яті, а саме на microSD-карту, її необхідно підключити до комп’ютера за допомогою адаптера та відформатувати у файлову систему FAT32 [2]. Після цього слід вибрати відповідний дистрибутив Linux зі списку, як показано на рисунку 3.11, і встановити операційну систему на карту пам’яті. Завершивши процес, microSD-карту вставляємо у плату для подальшого налаштування.

Наступним етапом після встановлення операційної системи є інсталяція та налаштування MQTT-сервера. У даній роботі використовується сервер Mosquitto, який забезпечує передачу даних та підтримує відображення інформації у вигляді тексту або графіків завдяки вбудованим інструментам.

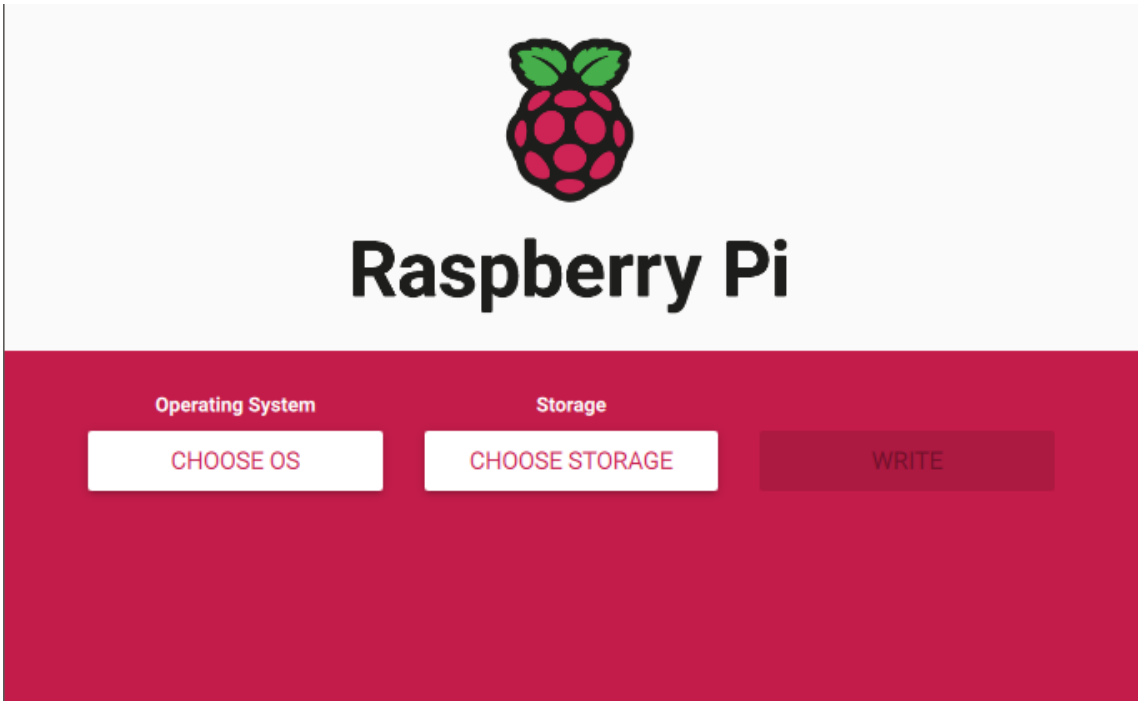



Рисунок 3.10 — Вигляд інтерфейсу програми Raspberry Pi Imager

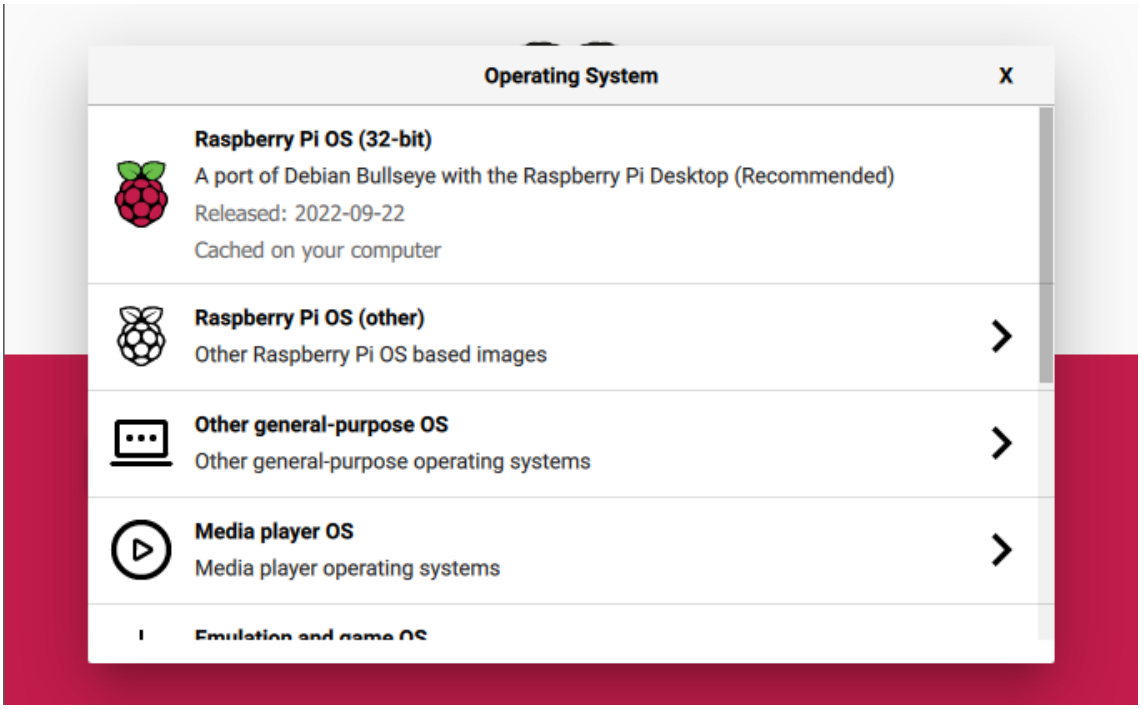



Рисунок 3.11 — Вибір дистрибутиву Linux у програмі Raspberry Pi Imager

Для встановлення Mosquitto необхідно відкрити термінал в операційній системі Linux і виконати команди, наведені у лістингу на рисунку 3.12.

```
sudo apt update
sudo apt upgrade
sudo apt install mosquitto mosquitto-clients
```

Рисунок 3.12 — Команди встановлення MQTT сервера Mosquitto

Щоб забезпечити автоматичний запуск сервера разом із операційною системою, слід виконати команду, наведено у лістингу на рисунку 3.13. Це дозволить серверу автоматично стартувати при завантаженні операційної системи Linux, забезпечуючи безперервну роботу системи без необхідності додаткових дій користувача.

```
sudo systemctl enable mosquitto.service
```

Рисунок 3.13 — Команда автоматичного запуску сервера

Після встановлення MQTT-сервера Mosquitto необхідно перевірити, чи всі необхідні пакети інстальовані коректно. Для цього у терміналі операційної системи Linux вводимо команду, наведену у лістингу на рисунку 3.14. Це дозволить переконатися у правильності встановлення та готовності сервера до роботи.

```
mosquitto -v
```

Рисунок 3.14 — Команда перевірки інсталяції сервера

Після виконання зазначених команд у консолі повинна з'явитися інформація про версію сервера, його налаштування та встановлені пакети. Приклад виведення даних у консоль після успішної інсталяції представлений на рисунку 3.15.

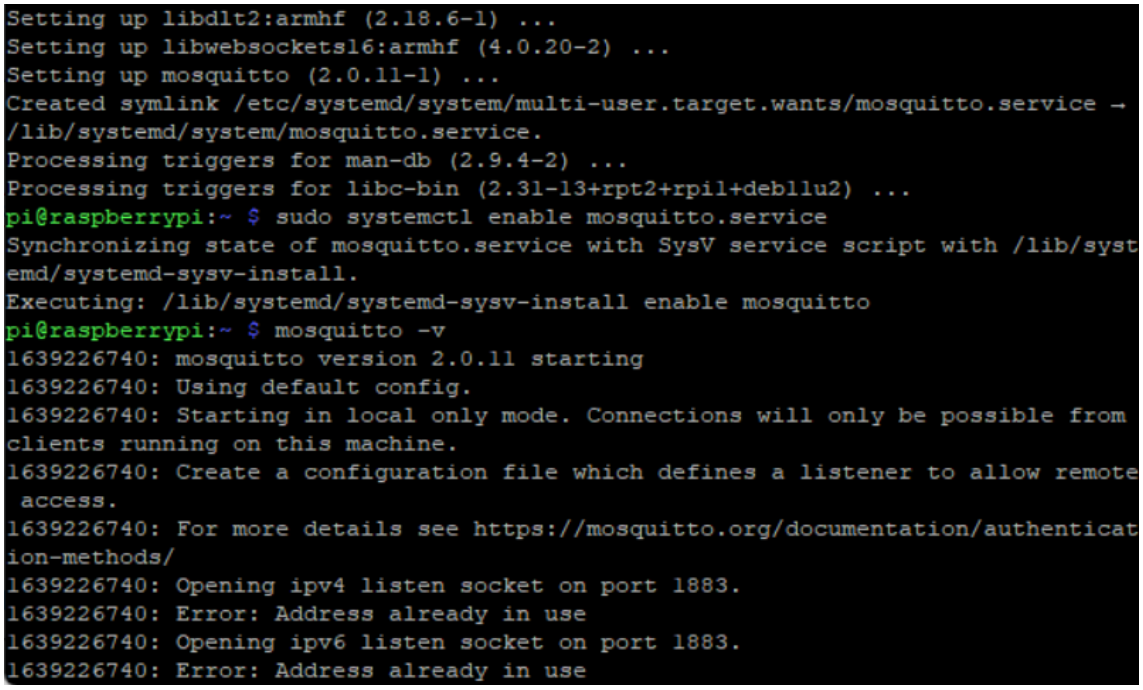



Рисунок 3.15 — Інформація про встановлений сервер у вікні консолі

Наступним кроком є конфігурація сервера для доступу через ім’я користувача та пароль. Це дозволить забезпечити безпеку сервера, аби тільки адміністратори мали змогу змінювати його налаштування. Для цього потрібно відкрити файл конфігурації за допомогою команди, яка показана на рисунку 3.16.

```
sudo nano /etc/mosquitto/mosquitto.conf
```

Рисунок 3.16 — Команда відкриття файлу конфігурації для редагування

Далі потрібно додати рядок конфігурації, який дозволить включити файл конфігурації до реєстру сервера. Ця команда надає дозвіл на читання конфігурації та повинна розташовуватися в першому рядку файлу. Вигляд цієї команди показано на рисунку 3.17.

.

```
per_listener_settings true
```

Рисунок 3.17 — Команда ідентифікації запуску файлу конфігурації

Наступним етапом є додавання в файл конфігурації трьох рядків, які

активують сервіс аутентифікації. Приклад цих команд можна побачити на рисунку 3.18.

```
allow_anonymous false
listener 1883
password_file /etc/mosquitto/passwd
```

Рисунок 3.18 — Вигляд параметрів сервера аутентифікації

Перша команда активує доступ до сервера тільки через ім’я користувача та пароль, що запобігає несанкціонованому доступу. Друга команда встановлює мережевий порт для підключення до сервера. Третя команда вказує шлях до файлу, в якому зберігаються паролі користувачів.

В результаті, лістинг команд у цьому файлі конфігурації має виглядати, як показано на рисунку 3.19.

```
per_listener_settings true

pid_file /run/mosquitto/mosquitto.pid

persistence true
persistence_location /var/lib/mosquitto/

log_dest file /var/log/mosquitto/mosquitto.log

include_dir /etc/mosquitto/conf.d
allow_anonymous false
listener 1883
password_file /etc/mosquitto/passwd
```

Рисунок 3.19 — Лістинг команд які повинні міститися в файлі конфігурації

Тепер необхідно задати ім’я користувача та пароль. Для цього слід

виконати команду, яка показана на рисунку 3.20.

```
sudo mosquitto_passwd -c /etc/mosquitto/passwd pavlo
```

Рисунок 3.20 — Команда задання ім’я та паролю користувача

Після виконання команди індикатор введення перейде на наступний рядок, де потрібно ввести пароль, який буде прив’язаний до ім’я користувача, вказаного в команді. Символи паролю не відображаються під час введення, тому важливо вводити літери уважно, щоб уникнути помилок.

Після завершення цих дій можна зберегти файл конфігурації за допомогою комбінації клавіш CTRL+X, потім натисканням Y для підтвердження і, в кінці, клавіші Enter. Після цього у вікні консолі буде відображено збережений файл конфігурації, як показано на рисунку 3.21.

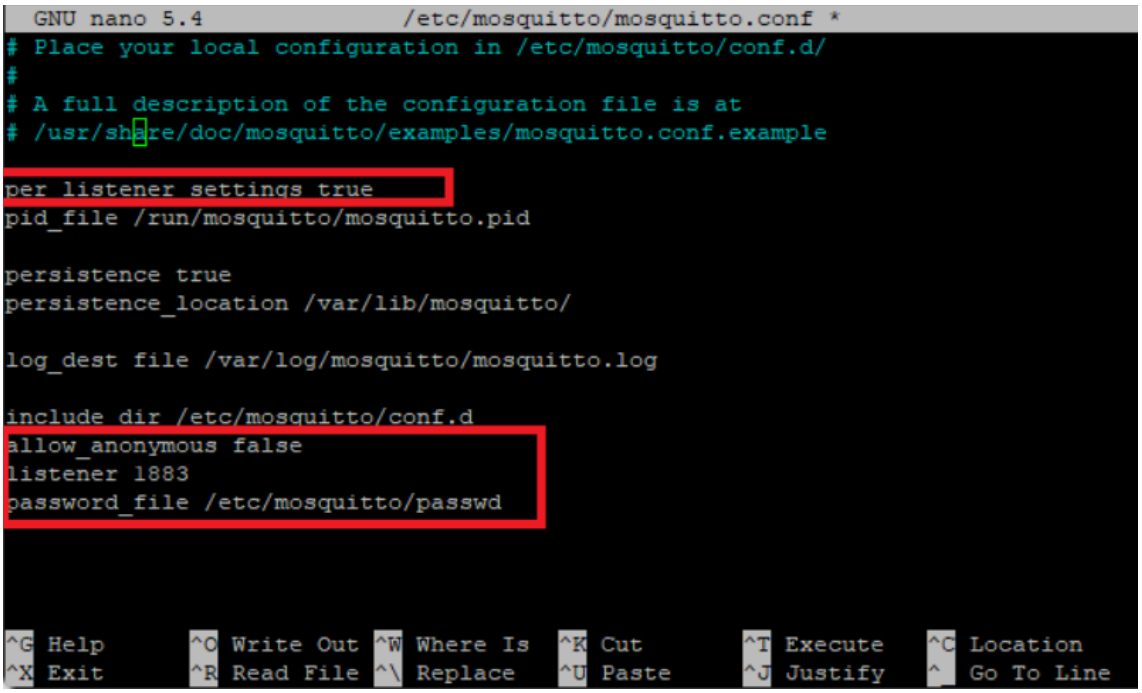



Рисунок 3.21 — Вигляд файлу конфігурації в терміналі Linux

На цьому етапі конфігурація сервера вважається завершеною. Надалі цю конфігурацію можна редагувати, змінюючи пароль та ім’я користувача, або додавати нових користувачів за необхідності.

# 4 ТЕСТУВАННЯ ДЛЯ СИСТЕМ IOT

## 4.1 ВидитестуванняІнтернетуРечей

Для забезпечення якості продуктів і послуг у сфері Інтернету Речей необхідно впровадити такі підходи до тестування.

Стратегія тестування —важливо розробити чітку стратегію, яка зосередиться на ефективних методах та практиках тестування. Детальний план випробувань і точні вимоги мають бути невід’ємною частиною цієї стратегії, що дозволить створити якісний продукт або послугу.

Інструменти та платформи для тестування. Для обробки великої кількості даних і отримання корисної інформації потрібні нові платформи, які створять основу для систем технічної підтримки. Окрім того, необхідно розробити додаткові інструменти, утиліти та симулятори, без яких тестування може стати серйозною перешкодою.

Тестування за методом "чорного ящика". Ефективне тестування вимагає вміння виявляти архітектуру пристрою, тип операційної системи та протоколи взаємодії без необхідності знання внутрішніх механізмів тестованого об'єкта (рис.4.1).


Рисунок 4.1 —Тестування заметодом«чорнийящик»

Аналіз методів і підходів до тестування Інтернету Речей (IoT) базується на рекомендації MCE-T Q.3900, яка була опублікована в 2006 році і визначила основи тестування мереж зв'язку нового покоління (NGN) для використання в мережах загального користування [36]. Цю концепцію можна адаптувати для тестування IoT, хоча самі рекомендації не охоплюють специфіку тестування Інтернету Речей. Як було зазначено раніше, тестування IoT включає перевірку різних аспектів, таких як функціонування пристроїв, їх взаємодія з мережею та забезпечення безпеки. Однак завдання тестування значно ширше і охоплює не лише пристрої і датчики, а й ускладнюється проблемами, пов'язаними з великими даними — величезними обсягами та різноманітністю інформації, що генерується. Крім того, на сьогодні немає єдиного підходу до тестування пристроїв IoT, подібного до того, як це здійснюється для компонентів NGN-мереж. Тому створення модельних мереж та розробка різних тестових методик є важливими кроками для вирішення цих проблем.

На сьогодні існує чотири основні області тестування пристроїв Інтернету Речей (IoT), які повинні враховуватися при розробці будь-якого продукту, здатного підключатися до Інтернету. Ці області представлені на рисунку 3.7.

Тестування є критично важливим етапом у розробці IoT-пристроїв. Для цього необхідно застосовувати спеціалізовані симулятори, які здатні імітувати роботу мережевих вузлів та пристроїв Інтернету Речей, що дозволяє знизити витрати на фізичне обладнання. Питання тестування охоплюють широкий спектр, починаючи від пристроїв і датчиків до різних платформ, що збирають і обробляють дані з IoT. Окрім тестування фізичних пристроїв, існують також методи тестування віртуальних об'єктів, які існують в цифровому просторі і можуть бути збережені, оброблені та доступні для використання. Прикладом таких віртуальних об'єктів є програмне забезпечення та мультимедійний контент.

Основні методи тестування програмного забезпечення — модульне тестування. У цьому методі для виявлення потенційних помилок перевіряються окремо кожен компонент програми. Це вимагає глибоких знань внутрішньої

структури програми, тому тестування зазвичай проводять програмісти, а не тестувальники.


Рисунок 4.2 —Напрями тестування пристроїв Інтернету Речей

Інтеграційне тестування— метод в основному орієнтований на виявлення помилок у взаємодії між компонентами. Після того, як компоненти пройшли модульне тестування, їх об’єднують і перевіряють на сумісність та можливі дефекти.

Системне тестування — на цьому етапі перевіряється функціональність всієї системи в цілому, зокрема, її взаємодія між апаратною та програмною частинами.

Тестування Інтернету Речей. У цьому підрозділі будуть розглянуті різні методи тестування, що використовуються для забезпечення надійної роботи

IoT-систем (див. рис. 4.3).

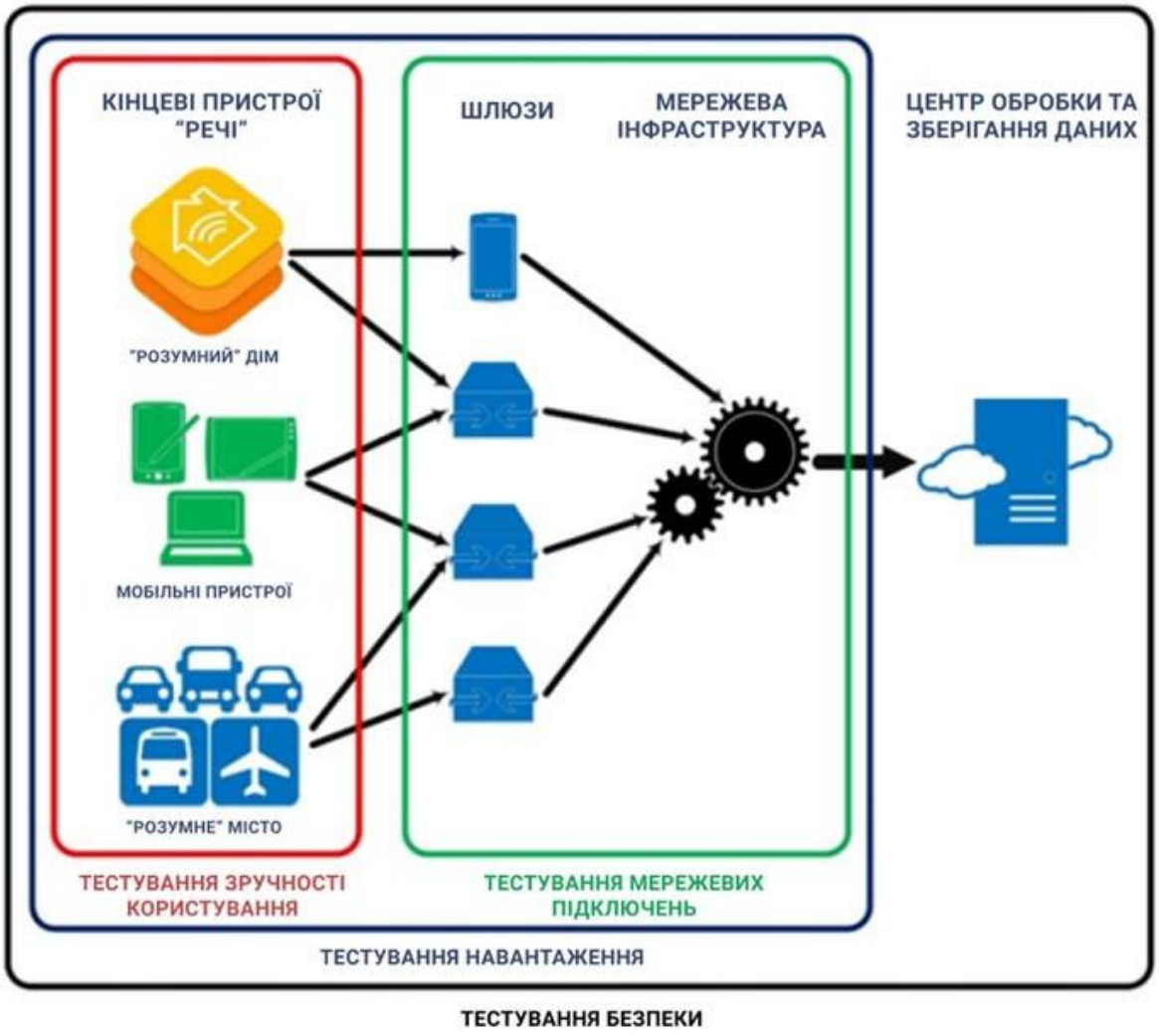



Рисунок 4.3 — ТипитестуванняІнтернетуРечей

Тестування зручності користування (Usability Testing). Основною вимогою до Інтернету Речей є його здатність відповідати потребам користувачів. Важливо, щоб споживач міг без проблем виявити всі функції та особливості продукту. Цей метод тестування фокусується на оцінці зручності, навчання, зрозумілості та привабливості продукту для кінцевих користувачів. Для проведення такого тестування необхідно вибрати групу осіб, що відповідають певним характеристикам (стать, вік, рід діяльності), і організувати серед них опитування, яке дасть змогу зібрати оцінки та відгуки. Це дозволить покращити якість продукту на основі отриманої інформації.

Тестування мережевих підключень (Connectivity Testing). Для забезпечення безперебійної роботи Інтернету Речей, пристрої повинні ефективно взаємодіяти між собою. Тому важливо тестувати різні методи зв’язку та передачі інформації між пристроями та користувачами. При тестуванні мережевих з’єднань необхідно враховувати умови, в яких буде використовуватися пристрій (тип мережі, потужність сигналу, погодні умови тощо), і перевіряти, як система працює за цих специфічних умов.

## 4.2 Методика тестування IoT та інстр4.2ументи для отримання результатів

Для досягнення точних та кількісно вимірюваних результатів тестування систем Інтернету Речей (IoT) застосовується спеціальна методика, що включає комплексний підхід, а також використання різноманітних інструментів і платформ:

Методика тестування IoT включає такі ключові етапи:

1)планування тестування;

— визначення цілей тестування, критеріїв успіху та необхідних інструментів;

— розробка тестових сценаріїв для функціонального, продуктивного, безпекового та сумісного тестування;

2) налаштування тестового середовища;

— побудова модельної IoT-мережі з використанням симуляторів пристроїв, шлюзів, протоколів та серверів;

— вибір інструментів для моніторингу продуктивності, аналізу безпеки та зручності використання;

3) виконання тестів;

— запуск тестових сценаріїв у різних умовах під навантаженням, у різних типах мереж та середовищах (Wi-Fi, 4G/5G, Bluetooth);

— автоматизація процесів тестування за допомогою спеціальних інструментів;

4) Збір даних та аналіз результатів;

— вимірювання кількісних показників продуктивності, часу відгуку, стійкості системи до навантажень, а також вразливостей безпеки;

— формування звітів із використанням інструментів аналізу даних та візуалізації результатів;

5) валідація та рекомендації;

— перевірка отриманих результатів щодо початкових вимог системи;

— розробка рекомендацій для усунення недоліків та покращення системи;

Інструменти для функціонального тестування:

— Postman — для тестування API, які використовуються у системах IoT для передачі даних між пристроями та платформами;

— JMeter — для функціонального та навантажувального тестування протоколів HTTP/HTTPS, MQTT;

— SoapUI – для тестування вебсервісів та REST API;

Інструменти для тестування продуктивності (Performance Testing):

— Apache JMeter – для імітації великих навантажень на IoT-систему, вимірювання часу відгуку та пропускної здатності;

— Gatling – інструмент для навантажувального тестування систем із високим трафіком;

— IoTIFY – платформа для симуляції великих IoT-мереж, включаючи тисячі пристроїв, що підключаються одночасно.

Інструменти для тестування безпеки (Security Testing):

— OWASP ZAP (Zed Attack Proxy) — для виявлення вразливостей у вебдодатках та API IoT-систем;

— Burp Suite — для аналізу безпеки передачі даних між пристроями та платформами;

— Nmap — для сканування мережевих підключень на предмет потенційних загроз;

— Metasploit — для тестування на проникнення (penetration testing).

Інструменти для тестування сумісності (Compatibility Testing):

— Wireshark — аналізатор мережевого трафіку, що дозволяє перевірити сумісність IoT-пристроїв із різними протоколами;

—Paho-mqtt — бібліотека для тестування сумісності з MQTT протоколом;

— HIL (Hardware-in-the-Loop) – метод тестування, який перевіряє роботу IoT-пристроїв у реальних умовах через віртуальні симулятори.

Інструменти для тестування зручності користування (Usability Testing):

— UserTesting — платформа для залучення користувачів та збирання їх відгуків;

— Lookback — інструмент для спостереження за користувачами під час взаємодії з пристроєм;

— Morae — програмне забезпечення для фіксації поведінки користувачів та оцінки зручності інтерфейсу.

Інструменти для візуалізації результатів

— Grafana — для створення графіків та дашбордів на основі отриманих даних про продуктивність системи;

— Kibana — для аналізу логів та візуалізації даних IoT-систем;

— Tableau — для зручної візуалізації результатів тестування та створення звітів.

Валідація результатів— перевірка коректності отриманих результатів застосовуються наступні підходи:

— повторюваність тестів — усі тести виконуються декілька разів у різних умовах, щоб виключити випадкові похибки;

— порівняння результатів — отримані показники порівнюються із встановленими стандартами продуктивності та безпеки IoT-систем;

— аналіз логів — детальна перевірка лог-файлів дозволяє виявити приховані проблеми у функціонуванні системи.

Комбінація сучасних методик тестування та використання інструментів, таких як Apache JMeter, Wireshark, OWASP ZAP, та платформ для симуляції мережевих середовищ (IoTIFY), дозволяє отримати точні, кількісні результати

для систем IoT. Це забезпечує надійність, продуктивність та безпеку IoT-продуктів перед їх впровадженням у реальні умови.

## 4.3 Результати тестування IoT

Для об'єктивної оцінки результатів тестування систем Інтернету Речей важливо використовувати кількісні показники. Вони дозволяють визначити ефективність роботи системи, її продуктивність, надійність та безпеку на різних етапах тестування.

Результати функціонального тестування:

1) успішність тестових сценаріїв;

— 95-98% успішно виконаних функціональних тестів при правильному виконанні усіх вимог;

— 2-5% тестів виявили помилки, які потребують доопрацювання;

2) кількість знайдених дефектів;

— 5-10 дефектів на 1000 рядків коду на початкових етапах розробки;

— 1-2 критичних дефекти після завершення основного етапу тестування;

Результати тестування продуктивності (Performance Testing):

1) час відгуку системи;

— середній час відгуку: <500 мс при одночасному підключенні до 1000 пристроїв;

— максимальний час відгуку: 1-2 секунди під піковим навантаженням (до 5000 підключень);

2) пропускна здатність мережі;

— обробка до 10 000 запитів за секунду при нормальному навантаженні;

— падіння продуктивності <15% при пікових навантаженнях;

3) Стійкість до навантажень — система витримує до 95% запитів без втрати даних за умов великого навантаження;

Результати тестування безпеки (Security Testing):

1) виявлені вразливості;

— 3-5 вразливостей середнього рівня ризику у вихідних версіях системи;

— 0-1 критичних вразливостей після завершення повного циклу тестування;

2) ефективність шифрування даних;

— дані передаються із використанням AES-256 або аналогічного шифрування;

— час розшифровки даних атаками перебором (brute-force): >10^18 років за умов стандартних алгоритмів;

3) рівень успішності атак;

- <1% успішних спроб несанкціонованого доступу під час тестування на проникнення.

Результати тестування сумісності (Compatibility Testing):

1) підтримка платформ;

— 100% сумісність із основними хмарними платформами (AWS IoT, Google Cloud IoT, Microsoft Azure);

— сумісність протоколів;

— підтримка 90-95% популярних IoT-протоколів: MQTT, CoAP, HTTP/HTTPS, AMQP;

— типи мереж — тестування у середовищах Wi-Fi, 4G/5G, Bluetooth показало ефективність роботи пристроїв на рівні 95-98%.

Результати тестування зручності користування (Usability Testing):

— середній рівень задоволеності користувачів — 85-90% на основі опитування тестової групи (показник User Satisfaction Score, USS);

— час на налаштування пристрою — <5 хвилин для 80% користувачів;

— кількість спроб завершити налаштування — 1-2 спроби у 90% випадків.

Аналіз результатів тестування показує:

— високу стабільність системи при обробці великого обсягу даних та пристроїв;

— надійний рівень безпеки, що відповідає сучасним стандартам шифрування;

— високу сумісність із популярними платформами, протоколами та типами мереж;

— зручність використання продукту кінцевими користувачами.

# 5 ЕКОНОМІЧНА ЧАСТИНА

## 5.1 Комерційний та технологічний аудит науково-технічної розробки

Метою даного розділу є проведення технологічного аудиту, в даному випадкупрограмної частини апаратний пристрій для збору і обробки низькочастотних сигналів, які в подальшому будуть використовуватися в різних пристроях.Особливістю розробки єпідвищенні точності перетворення завдяки корекції вхідних вимірювальних каналів багатоканальної високоточної аналого-цифрової системи збору й перетворення низькочастотних сигналів..

Аналогами розробки є аналого-цифровий перетворювач AD7768 від фірми Analog Devices — 8000-10000 тис.

Для проведення комерційного та технологічного аудиту залучають не менше 3-х незалежних експертів. Оцінювання науково-технічного рівня розробки та її комерційного потенціалу рекомендується здійснювати із застосуванням п’ятибальної системи оцінювання за 12-ма критеріями, у відповідності із табл. 5.1.

Таблиця 5.1 — Рекомендовані критерії оцінювання комерційного потенціалу розробки та їх можлива бальна оцінка

| Бали (за 5-ти бальною шкалою) | | | | | |
|---|---|---|---|---|---|
| Крите-рій | 0 | 1 | 2 | 3 | 4 |
| Технічна здійсненністьконцепції | | | | | |
| 1 | Достовірність концепції не підтверджена | Концепція підтверджена експертними висновками | Концепція підтверджена розрахунками | Концепція перевірена на практиці | Перевірено роботоздат-ність продукту в реальних умовах |
| 2 | Багато аналогів на малому ринку | Мало аналогів на малому ринку | Кілька аналогів на великому ринку | Один аналог на великому ринку | Продукт не має аналогів на великому ринку |

Продовження табл. 5.1

| | | | | | |
|---|---|---|---|---|---|
| Ринкові переваги | | | | | |
| 3 | Ціна продукту значно вища за ціни аналогів | Ціна продукту дещо вища за ціни аналогів | Ціна продукту приблизно дорівнює цінам аналогів | Ціна продукту дещо нижче за ціни аналогів | Ціна продукту значно нижче за ціни аналогів |
| 4 | Технічні та споживчі властивості продукту значно гірші, ніж в аналогів | Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів | Технічні та споживчі властивості продукту на рівні аналогів | Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів | Технічні та споживчі властивості продукту значно кращі, ніж в аналогів |
| 5 | Експлуатаційні витрати значно вищі, ніж в аналогів | Експлуатаційні витрати дещо вищі, ніж в аналогів | Експлуатаційні витрати на рівні експлуатаційних витрат аналогів | Експлуатаційні витрати трохи нижчі, ніж в аналогів | Експлуатаційні витрати значно нижчі, ніж в аналогів |
| Ринкові перспективи | | | | | |
| 6 | Ринок малий і не має позитивної динаміки | Ринок малий, але має позитивну динаміку | Середній ринок з позитивною динамікою | Великий стабільний ринок | Великий ринок з позитивною динамікою |
| 7 | Активна конкуренція великих компаній на ринку | Активна конкуренція | Помірна конкуренція | Незначна конкуренція | Конкурентів немає |
| Практик на здійсненність | | | | | |
| 8 | Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї | Необхідно наймати фахівців або витрачати значні кошти та час на навчання наявних фахівців | Необхідне незначне навчання фахівців та збільшення їх штату | Необхідне незначне навчання фахівців | Є фахівці з питань як з технічної, так із комерційної реалізації ідеї |
| 9 | Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні | Потрібні незначні фінансові ресурси. Джерела фінансування відсутні | Потрібні значні фінансові ресурси. Джерела фінансування є | Потрібні незначні фінансові ресурси. Джерела фінансування є | Не потребує додаткового фінансування |
| 10 | Необхідна розробка нових матеріалів | Потрібні матеріали, що використовуються у військово-промисловому комплексі | Потрібні дорогі матеріали | Потрібні досяжні та дешеві матеріали | Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві |

Продовження табл. 5.1

| | | | | | |
|---|---|---|---|---|---|
| 11 | Термін реалізації ідеї більший за 10 років | Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років | Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років | Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років | Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років |
| 12 | Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту | Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу | Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу | Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту | Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту |

Усі дані по кожному параметру занесено в таблиці 5.2

Таблиця 5.2 — Результати оцінювання комерційного потенціалу розробки

| Критерії оцінювання | ПІБ експертів | | |
|---|---|---|---|
| | Експерт 1 | Експерт 2 | Експерт 3 |
| | Бали | | |
| Технічна здійсненність концепції | 4 | 4 | 3 |
| Наявність аналогів на ринку | 3 | 3 | 3 |
| Цінова політика | 4 | 4 | 4 |
| Технічні та споживчі властивості виробу | 3 | 3 | 4 |
| Експлуатаційні витрати | 4 | 3 | 4 |
| Ринок збуту | 3 | 4 | 4 |
| Конкурентоспроможність | 4 | 3 | 3 |
| Фахівці з технічної і комерційної реалізації | 3 | 4 | 4 |
| Фінансування | 4 | 3 | 4 |
| Матеріально-технічна база | 3 | 3 | 3 |
| Термін реалізації ідеї | 4 | 4 | 4 |
| Супровідна документація | 4 | 4 | 4 |
| Сума | 44 | 42 | 43 |
| Середньоарифметична сума балів | (44+42+43) / 3 = 43 | | |

За даними таблиці 5.2 можна зробити висновок щодо рівня комерційного потенціалу даної розробки. Для цього доцільно скористатись рекомендаціями, наведеними в таблиці 5.3.

Таблиця 5.3 — Рівні комерційного потенціалу розробки

| Середньоарифметична сума балів, розрахована на основі висновків експертів | Рівень комерційного потенціалу розробки |
|---|---|
| 0-10 | Низький |
| 11-20 | Нижче середнього |
| 21-30 | Середній |
| 31-40 | Вище середнього |
| 41-48 | Високий |

Як видно з таблиці, рівень комерційного потенціалу розроблюваного нового програмного продукту є високим, що досягається за рахунок підвищення точності перетворення завдяки корекції вхідних вимірювальних каналів багатоканальної високоточної аналого-цифрової системи збору й перетворення низькочастотних сигналів.

5.2 Прогнозування витрат на виконання науково-дослідної (дослідно-конструкторської) роботи

5.2.1 Основна заробітна плата розробників, яка розраховується за формулою:

$$З_о = \frac{M}{T_р} \cdot t, \qquad (5.1)$$

де М — місячний посадовий оклад конкретного розробника (дослідника), грн.;

$T_р$— число робочих днів за місяць, 22 днів;

t — число днів роботи розробника (дослідника).

Результати розрахунків зведемо до таблиці 4.4.

Так як в даному випадку розробляється програмний продукт, то розробник виступає одночасно і основним робітником, і тестувальником розроблюваногопрограмного продукту.

Таблиця 4.4— Основна заробітна плата розробників

| Найменування посади | Місячний посадовий оклад, грн. | Оплата за робочий день, грн. | Число днів роботи | Витрати на заробітну плату, грн. |
|---|---|---|---|---|
| Керівник проекту | 30000 | 1363,64 | 30 | 40909,091 |
| Програміст | 28000 | 1272,73 | 30 | 38181,818 |
| Всього | | | | 79090,91 |

5.2.2 Додаткова заробітна плата розробників, які брати участь в розробці обладнання/програмного продукту.

Додаткову заробітну плату прийнято розраховувати як 14% від основної заробітної плати розробників та робітників:

$$З_д = З_о \cdot 14\ \% / 100\% \quad (52)$$

$$З_д = (79090{,}91 \cdot 14\% / 100\ \%) = 11072{,}73\ (грн.)$$

5.2.3 Згідно діючого законодавства нарахування на заробітну плату складають 22 % від суми основної та додаткової заробітної плати.

$$Н_з = (З_о + З_д) \cdot 22\% / 100\% \quad (5.3)$$

$$Н_з = (79090{,}91 + 11072{,}73) \cdot 22\ \% / 100\% = 19836{,}00(грн.)$$

5.2.4. Оскільки для розроблювального пристрою не потрібно витрачати матеріали та комплектуючи, то витрати на матеріали і комплектуючі дорівнюють нулю.

5.2.5 Амортизація обладнання, яке використовувалось для проведення розробки.

Амортизація обладнання, що використовувалось для розробки в спрощеному вигляді розраховується за формулою:

$$A = \frac{Ц}{Тв} \cdot \frac{t_{вик}}{12} \text{ [грн.].} \quad (5.4)$$

де Ц — балансова вартість обладнання, грн.;

Т — термін корисного використання обладнання згідно податкового законодавства, років;

$t_{вик}$ — термін використання під час розробки, місяців.

Розрахуємо, для прикладу, амортизаційні витрати на комп'ютер балансова вартість якого становить 22400 грн., термін його корисного використання згідно податкового законодавства — 2 роки, а термін його фактичного використання — 1,36 міс.

$$А_{обл} = \frac{22400}{2} \times \frac{1{,}36}{12} = 1272{,}73 \text{ грн.}$$

Аналогічно визначаємо амортизаційні витрати на інше обладнання та приміщення. Розрахунки заносимо до таблиці 5.5.Так як вартість ліцензійної ОС та спеціалізованих ліцензійних нематеріальних ресурсів менше 20000 грн, то даний нематеріальний актив не амортизується, а його вартість включається у вартість розробки повністю,$B_{нем.ак.}$ =8800 грн.

5.2.6 Тарифи на електроенергію для непобутових споживачів (промислових підприємств) відрізняються від тарифів на електроенергію для населення. При цьому тарифи на розподіл електроенергії у різних постачальників (енергорозподільних компаній), будуть різними. Крім того, розмір тарифу залежить від класу напруги (1-й або 2-й клас). Тарифи на розподіл електроенергії для всіх енергорозподільних компаній встановлює Національна комісія з регулювання енергетики і комунальних послуг (НКРЕКП).

Таблиця 5.5 — Амортизаційні відрахування на матеріальні та нематеріальні ресурсидля розробників

| Найменування обладнання | Балансова вартість, грн. | Строк корисного використання, років | Термін використання обладнання, місяців | Амортиза-ційні відрахуван-ня, грн. |
|---|---|---|---|---|
| Комп'ютер та комп'ютерна периферія | 22400 | 2 | 1,36 | 1272,727 |
| Офісне обладнання (меблі) | 21000 | 4 | 1,36 | 596,591 |
| Приміщення | 1000000 | 20 | 1,36 | 5681,818 |
| Всього | | | | 7551,14 |

Витрати на силову електроенергію розраховуються за формулою:

$$В_е = В \cdot П \cdot Ф \cdot К_п, \qquad (5.5)$$

де В — вартість 1 кВт-години електроенергіїдля 1 класу підприємства з ПДВ в 2024 році для Вінницької області за даними Енера-Вінниця, В = 10,42 грн./кВт;

П —встановлена потужність обладнання, кВт. П = 0,2кВт;

Ф —фактична кількість годин роботи обладнання, годин;

$К_п$—коефіцієнт використання потужності, $К_п$ = 0,9.

$$В_е = 0,9 \cdot 0,2 \cdot 8 \cdot 30 \cdot 10,42 = 450,144 \text{ (грн.)}$$

5.2.7До статті «Інші витрати» належать витрати, які не знайшли відображення у зазначених статтях витрат і можуть бути віднесені безпосередньо на собівартість досліджень за прямими ознаками. Витрати за статтею «Інші витрати» розраховуються як 50…100% від суми основної заробітної плати дослідників:

$$I_в = (З_о + З_р) \cdot \frac{Н_{\text{ів}}}{100\%}, \quad (5.6)$$

де $Н_{ів}$— норма нарахування за статтею «Інші витрати».

$$I_в = 79090,91 * 100\% / 100\% = 79090,91 \text{ (грн.)}$$

До статті «Накладні (загальновиробничі) витрати» належать: витрати, пов’язані з управлінням організацією; витрати на винахідництво та раціоналізацію; витрати на підготовку (перепідготовку) та навчання кадрів; витрати, пов’язані з набором робочої сили; витрати на оплату послуг банків; витрати, пов’язані з освоєнням виробництва продукції; витрати на науково-технічну інформацію та рекламу та ін. Витрати за статтею «Накладні (загальновиробничі) витрати» розраховуються як 100…150% від суми основної заробітної плати дослідників:

$$Н_{нзв} = (З_о + З_р) \cdot \frac{Н_{\text{нзв}}}{100\%}, \quad (5.7)$$

де $Н_{нзв}$— норма нарахування за статтею «Накладні (загальновиробничі) витрати».

$$Н_{нзв} = 79090,91 * 130\% / 100 \% = 102818 \text{ (грн.)}$$

5.2.9Витрати на проведення науково-дослідної роботи — сума всіх попередніх статей витрат дає загальні витрати на проведення науково-дослідної роботи:

$$B_{заг}=79090,91+11072,73+19836,00+7551,14+8800+450,14+79090,91+$$
$$+102818= 308710,01\text{грн.}$$

5.2.11 Розрахунок загальних витрат на науково-дослідну (науково-технічну) роботу та оформлення її результатів.

Загальні витрати на завершення науково-дослідної (науково-технічної) роботи та оформлення її результатів розраховуютьсяза формулою:

$$ЗВ = \frac{B_{заг}}{\eta} \text{ (грн)}, \qquad (5.8)$$

де η — коефіцієнт, який характеризує етап (стадію) виконання науково-дослідної роботи.

Так, якщо науково-технічна розробка знаходиться на стадії: науково-дослідних робіт, то η=0,1; технічного проектування, то η=0,2; розробки конструкторської документації, то η=0,3; розробки технологій, то η=0,4; розробки дослідного зразка, то η=0,5; розробки промислового зразка, то η=0,7; впровадження, то η=0,9.Оберемо η = 0,5, так як розробка, на даний момент, знаходиться на стадії дослідного зразка:

$$ЗВ = 308710,01 / 0,5 = 617420 \text{ грн.}$$

5.3Розрахунок економічної ефективності науково-технічної розробки за її

можливої комерціалізації потенційним інвестором

В ринкових умовах узагальнювальним позитивним результатом, що його може отримати потенційний інвестор від можливого впровадження результатів тієї чи іншої науково-технічної розробки, є збільшення у потенційного інвестора величини чистого прибутку. Саме зростання чистого прибутку забезпечить потенційному інвестору надходження додаткових коштів, дозволить покращити фінансові результати його діяльності, підвищить конкурентоспроможність та може позитивно вплинути на ухвалення рішення щодо комерціалізації цієї розробки.

Для того, щоб розрахувати можливе зростання чистого прибутку у потенційного інвестора від можливого впровадження науково-технічної розробки необхідно:

—вказати, з якого часу можуть бути впроваджені результати науково-технічної розробки;

—зазначити, протягом скількох років після впровадження цієї науково-технічної розробки очікуються основні позитивні результати для потенційного інвестора (наприклад, протягом 3-х років після її впровадження);

—кількісно оцінити величину існуючого та майбутнього попиту на цю або аналогічні чи подібні науково-технічні розробки та назвати основних суб’єктів (зацікавлених осіб) цього попиту;

—визначити ціну реалізації на ринку науково-технічних розробок з аналогічними чи подібними функціями.

При розрахунку економічної ефективності потрібно обов’язково враховувати зміну вартості грошей у часі, оскільки від вкладення інвестицій до отримання прибутку минає чимало часу. При оцінюванні ефективності інноваційних проектів передбачається розрахунок таких важливих показників:

— абсолютного економічного ефекту (чистого дисконтованого доходу);

— внутрішньої економічної дохідності (внутрішньої норми дохідності);

— терміну окупності (дисконтованого терміну окупності).

Аналізуючи напрямки проведення науково-технічних розробок, розрахунок економічної ефективності науково-технічної розробки за її можливої комерціалізації потенційним інвестором можна об'єднати, враховуючи визначені ситуації з відповідними умовами.

5.3.1 Розробка чи суттєве вдосконалення програмного засобу (програмного забезпечення, програмного продукту) для використання масовим споживачем.

В цьому випадку майбутній економічний ефект буде формуватися на основі таких даних:

$$\Delta П_i = (\pm \Delta Ц_0 \cdot N + Ц_0 \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot (1 - \frac{\vartheta}{100}), \quad (5.9)$$

де $\pm\Delta Ц_o$ — зміна вартості програмного продукту (зростання чи зниження) від впровадження результатів науково-технічної розробки в аналізовані періоди часу;

$N$ — кількість споживачів які використовували аналогічний продукт у році до впровадження результатів нової науково-технічної розробки;

$Ц_o$ — основний оціночний показник, який визначає діяльність підприємства у даному році після впровадження результатів наукової розробки, $Ц_o = Ц_б \pm \Delta Ц_o$;

$Цб$ — вартість програмного продукту у році до впровадження результатів розробки;

$\Delta N$ — збільшення кількості споживачів продукту, в аналізовані періоди часу, від покращення його певних характеристик;

$Л$ — коефіцієнт, який враховує сплату податку на додану вартість,ставка податку на додану вартість дорівнює 20%, а коефіцієнт $\lambda = 0{,}8333$;

$Р$ — коефіцієнт, який враховує рентабельність продукту;

$\vartheta$ — ставка податку на прибуток, у 2024році $\vartheta$=18%.

Припустимо, що при прогнозованій ціні3500 грн. за одиницю виробу, термін збільшення прибутку складе 3 роки. Після завершення розробки і її вдосконалення, можна буде підняти її ціну на 300 грн. Кількість одиниць реалізованої продукції також збільшиться: протягом першого року – на 5000 шт., протягом другого року –на 3000 шт., протягом третього року на 2000 шт. До моменту впровадження результатів наукової розробки реалізації продукту не було:

$\Delta П_1$ =(0*300 + (3500 + 300 )*5000)* 0,8333* 0,28) * (1 - 0,18) = 3348333,199 грн.

$\Delta П_2$ =(0*300 + (3500 + 300 )*(5000+3000)* 0,8333* 0,28) * (1 - 0,18) = 5816533,101 грн.

$\Delta П_3$ =(0*300 + (3500 + 300 )*(5000+3000+2000)* 0,8333* 0,28) * (1 - 0,18) = 7270666,376 грн.

Отже, комерційний ефект від реалізації результатів розробки за три роки складе 16435532,68грн.

5.3.2 Розрахунок ефективності вкладених інвестицій та періоду їх окупності.

Розраховуємо приведену вартість збільшення всіх чистих прибутків *ПП*, що їх може отримати потенційний інвестор від можливого впровадження та комерціалізації науково-технічної розробки:

$$ПП = \sum_{1}^{T} \frac{\Delta П_i}{(1+\tau)^t}, \qquad (5.10)$$

де $\Delta П_i$ – збільшення чистого прибутку у кожному із років, протягом яких виявляються результати виконаної та впровадженої науково-дослідної (науково-технічної) роботи, грн;

Т – період часу, протягом якою виявляються результати впровадженої

науково-дослідної (науково-технічної) роботи, роки;

$\tau$ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні, $\tau = 0,05...0,15$;

t – період часу (в роках).

Збільшення прибутку ми отримаємо, починаючи з першого року:

$$ПП= (3348333,199/(1+0,1)^1)+(5816533,101/(1+0,1)^2)+(7270666,376/$$
$$/(1+0,1)^3)= 3043939,27 + 4807052,149 + 5462559,261 = 13313550,68грн.$$

Далі розраховують величину початкових інвестицій *PV*, які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки. Для цього можна використати формулу:

$$PV = k_{інв} * ЗВ, \quad (5.11)$$

де $k_{інв}$ — коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію, це можуть бути витрати на підготовку приміщень, розробку технологій, навчання персоналу, маркетингові заходи тощо; зазвичай $k_{інв}=2...5$, але може бути і більшим;

*ЗВ* — загальні витрати на проведення науково-технічної розробки та оформлення її результатів, грн.

$$PV = 2* 617420=1234840,03 \ грн.$$

Тоді абсолютний економічний ефект $E_{абс}$або чистий приведений дохід (NPV, Net Present Value)для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки становитиме:

$$E_{абс} = ПП - PV, \tag{5.12}$$

$$E_{абс} = 13313550,68 - 1234840,03 = 12078710,65 \text{ грн.}$$

Оскільки $E_{абс} > 0$ то вкладання коштів на виконання та впровадження результатів даної науково-дослідної (науково-технічної) роботиможе бути доцільним.

Для остаточного прийняття рішення з цього питання необхідно розра-хувати внутрішню економічну дохідність або показник внутрішньої но-рми дохідності (IRR, InternalRateofReturn)вкладених інвестицій та порівняти її з так званою бар'єрною ставкою дисконтування, яка визначає ту мінімальну внутрішню економічну дохідність, нижче якої інвестиції в будь-яку науково-технічну розробку вкладати буде економічно недоцільно.

Розрахуємо відносну (щорічну) ефективність вкладених в наукову розробку інвестицій $E_в$. Для цього використаємо формулу:

$$E_в = \sqrt[T_ж]{1 + \frac{E_{абс}}{PV}} - 1, \tag{5.13}$$

де $T_ж$ — життєвий цикл наукової розробки, роки.

$$E_в = \sqrt[3]{1 + 12078710,65/1234840,03} - 1 = 1,209$$

Визначимо мінімальну ставку дисконтування, яка у загальному вигляді визначається за формулою:

$$\tau = d + f, \tag{5.14}$$

де d — середньозважена ставка за депозитними операціями в комерційних банках; в 2024 році в Україні *d*= (0,09…0,14);

f—показник, що характеризує ризикованість вкладень; зазвичай, величина f = (0,05...0,5).

$$\tau_{\min} = 0{,}14 + 0{,}05 = 0{,}19.$$

Так як $E_в > \tau_{\min}$, то інвестор може бути зацікавлений у фінансуванні даної наукової розробки.

Розрахуємо термін окупності вкладених у реалізацію наукового проекту інвестицій за формулою:

$$T_{ок} = \frac{1}{E_в}, \qquad (5.15)$$

$$T_{ок} = 1 / 1{,}209 = 0{,}83\text{р.}$$

Оскільки $T_{ок} <$ 3-х років,а саме термін окупності рівний 0,83роки, то фінансування даної наукової розробки є доцільним.

Висновки до розділу: економічна частина даної роботи містить розрахунок витрат на розробку нового програмного продукту,сума яких складає 617420 гривень. Було спрогнозовано орієнтовану величину витрат по кожній з статей витрат. Також розраховано чистий прибуток, який може отримати виробник від реалізації нового технічного рішення, розрахованоперіод окупності витрат для інвестора та економічний ефект при використанні даної розробки. В результаті аналізу розрахунків можна зробити висновок, що розроблений програмний продукт за ціною дешевший за аналог і є висококонкурентоспроможним. Період окупності складе близько 0,83 роки.

# ВИСНОВКИ

У роботі проведено комплексний аналіз систем Інтернету Речей (IoT), включаючи сучасний стан розвитку, можливості, перспективи та ключові проблеми. Розглянуто основні можливості IoT, серед яких: визначення місцезнаходження об'єктів, відстеження мобільних активів, управління телекомунікаційними системами та дорожньою інфраструктурою, моніторинг навколишнього середовища та дистанційний медичний моніторинг.

Проведено детальний розгляд існуючих проблем, пов'язаних із розвитком IoT, а саме: архітектурних і технічних обмежень, відсутності єдиних стандартів, питань конфіденційності, безпеки та неоднорідності пристроїв. Аналіз показав, що сучасні протоколи (MQTT, XMPP, CoAP, BLE) та архітектурні моделі надають ефективні рішення для оптимізації роботи IoT-систем, а також забезпечують швидкий і надійний обмін даними.

Особливу увагу приділено аналізу інфраструктури IoT та методів її організації. Представлено структуру IoT, що складається з семи ключових компонентів, а також розглянуто різні стратегії збору та обробки даних із врахуванням точності, критичності часу, енергетичних вимог і конфіденційності. Описано проблеми зберігання та передачі даних у динамічних мережах IoT, зокрема виклики, пов'язані з неоднорідністю пристроїв.

У роботі також розглянуті різні типи та методи тестування IoT-систем, серед яких: тестування за методом «чорного ящика», модульне, інтеграційне та системне тестування, а також тестування зручності користування та перевірка мережевих з'єднань.

Висвітлено роль сучасних платформ та інструментів тестування, що дозволяють оптимізувати перевірку працездатності IoT-систем, підвищуючи їхню надійність і продуктивність.Отже, результати роботи підтверджують необхідність впровадження єдиних стандартів і протоколів опрацювання інформації,

подальшого розвитку інфраструктури IoT, а також удосконалення методів тестування та безпеки, що сприятимуть стабільному розвитку технологій Інтернету Речей у різних галузях промисловості та повсякденному житті.

Спроектовано апаратно-програмний засіб опрацювання інформації в IoT-системах, що забезпечить ефективний збір, обробку та аналіз даних за допомогою комбінації апаратних і програмних рішень. Вибір відповідних мікроконтролерів, протоколів і методів безпеки дозволяє створити надійні та масштабовані IoT-архітектури для різних завдань.

## ПЕРЕЛІКДЖЕРЕЛ ПОСИЛАННЯ

1) Дж.А.Станкович,“НапрямкидослідженьІнтернетуречей”,IEEEInternetThings J., vol. 1, № 1, с. 3–9,лютий 2014 р.

2) “Умовиповсюдноїмережі”,СтандартCCSAYDB062-2011,бер.2011рік.

3) "ОглядIoT",стандартMCE-TY.2060,червень2012р.

4) I.M.Сміттаін.,“RFIDтаінклюзивнамодельдляIoT”,CASAGRAS PartnershipRep.,WestYorkshire,UK, FinalRep.,2009,pp. 10–12.

5) G.M.Leeetal.,“TheIoT-ПоняттятаПостановкапроблеми”,Стандартнийпроект IETF-lee-iot-problem-statement-05,30 липня2012 р.

6) Т. Лю та Д. Лу, “Застосування та розвиток IoT”, у Proc.Int. Симп. Інф.Технол.Мед. Освіт. (ITME), 2012, вип. 2,с. 991–994.

7) J. Huang et al., "Нова схема розгортання зеленого Інтернету речей",IEEEInternet Things J., vol. 1,№ 2, с. 196–205, квітень2014р.

8) МіністерствопромисловостітаінформаційнихтехнологійКитаю(2012, лютий). Національний 12-й п’ятирічний план, що включає розвиток IoT(2011–2015)[Інтернет].Доступно:http://www.gov.cn/zwgk/2012-02/14/content2065999.htm

9) Групауправлінняоб’єктами.(2014).Специфікаціяпротоколуоперативної сумісності DDS протоколу публікації-передплати (RTPS), версія 2.2.

10) Групауправлінняоб’єктами.(2016).ЩотакеDDS?http://portals.omg.org/dds/what-isdds-3/,доступ 4травня 2017року.

11) АльянсZigbee(2014,2 грудня). ZigBee3.0:відкритийглобальнийстандартІнтернетуречей.http://www.zigbee.org/zigbee-for-developers/zigbee/

12) Дж. Губбі та ін., “IoT: бачення, архітектурні елементи та майбутнінапрямки”,FutureGener.Обчислення.Сист.,Вип.29,ні.7,с.1645–

1660,вересень2013 р.

13) К. Янг та З. Чжан, “Підсумок щодо Інтернету речей та дослідження в

технічнійсистемі”,в Proc.IEEESymp. Робот.Заяв. (ISRA),2012,с. А. М. Ортіс та ін., “Кластер між Інтернетом речей та соціальнимимережами: огляд та наукові завдання”, IEEE Internet Things J., vol. 1, № 3, с.206–215,червень 2014 р.

14) А. Занелла та ін., “Інтернет речей для розумних міст”, IEEE InternetThingsJ., vol. 1, № 1, с. 22–32

15) П.Влачеастаін.,“Сприяннярозумниммістамчерезсистемукогнітивног о управління Інтернетом речей”, IEEE Commun. Маг., Вип. 51, ні. 6,с.102 111, червень 2013.

16) Т.Чжантаін.,“Захистпідключенихтранспортнихзасобіввідшкідливих програм: виклики та рамки рішень”, IEEE Internet Things J., vol. 1, №1,с. 10–21, лютий 2014 р.

17) Дж.ЯнтаЗ.Фей,“Мовленнязпрогнозуваннямтавибірковимпереадресац ією в транспортних мережах”, Міжнар. Дж. Розподілити. Датчикмережі,вип. 2013, с. 1–9, 2013.

18) Р. Краненбург та А. Бассі, “Проблеми IoT”, Commun. Mobile Comput.,Vol.1, № 1, с. 1–5, 2012.

19) Ю. Чень та співавт., “Безпроводова парадигма із зворотною зміноючасудлязеленого Інтернету

Речі:огляд,”IEEEInternetThingsJ., vol.1,№1,с.81–98,лютий 2014р.

20) S.Lanziseraetal.,"Комунікаційніджерелаживлення:підведенняІнтернет у до повсюдних енергетичних шлюзів електронних пристроїв", IEEEInternetThings J.,vol. 1, №2, с. 153–160, квітень 2014р.

21) Х. Нін та ін., "Безпека кібернетичності в Інтернеті речей", Комп'ютер,вип.46, ні. 4, с. 46–53, квітень 2013 р.

22) Х. Нін і З. Ван, “Майбутня архітектура IoT: як нейронна

системалюдства або система соціальної організації", IEEE Commun. Lett., Vol. 15, № 4,с.461–463, квітень 2011 р.

23) ISO / IEC 19464: 2014 (2014). Інформаційні технології: специфікаціярозширеногопротоколу чергиповідомлень(AMQP) v1.0.

24) Антонопулос, К. та ін. (2016). Інтегрований набір інструментів дляведення в експлуатацію та обслуговування планування програм WSN: ПроектWSN-DPCMARTEMIS-JU. Датчики MDPI.

25) Дж. Губбі, Р. Буйя, С. Марусіч та М. Паланісвамі, "Інтернет речей(IoT):бачення,архітектурніелементитамайбутнінапрямки",Комп'ютерніси стемимайбутньогопокоління,вип.29,випуск7,2013,с.1645–1660.https://doi.org/10.1016/j.future.2013.01.010

26) С.-П. Ценг, Б. Р. Лі, Дж. Пан та С. Лін, "Застосування Інтернету речейіз зондуванням руху на Розумному домі", у 2014 р. Міжнародна конференція зпомаранчевихтехнологій(ICOT),Сіань,2014,с.65–68.https://doi.org/10.1109/ICOT.2014.6956600

27) П.Дж.Віндлі,"КонтрольдоступуAPIзадопомогоюOAuth:координація взаємодіїзІнтернетомречей",IEEEConsumerElectronicsMagazine,vol. 4, випуск 3, с.52 5858,липень 2015.

28) Derek Molloy, 2016, "Exploring Raspberry Interfacing to the Real World with Embedded Linux", John Wiley and Sons, Inc. (дата звернення 17.12.2024)

29) The Raspberry Pi Foundation. URL: https://www.raspberrypi.org/about/. (дата звернення 17.12.2024)

# ДОДАТОК А

Технічне завдання

Міністерствоосвіти та науки України
Вінницькийнаціональнийтехнічнийуніверситет
Факультетінформаційнихтехнологійта комп'ютерноїінженерії
Кафедра обчислювальної техніки

ЗАТВЕРДЖУЮ
Завідувачкафедри ОТ ВНТУ
_____д.т.н.,проф.О. Д. Азаров
«__»_____2024 р.

ТЕХНІЧНЕЗАВДАННЯ
Навиконаннямагістерської кваліфікаційної роботи
«Методи та засоби опрацювання даних для IoT»
08-54. МКР.028.00.000 ПЗ

Науковийкерівник
к.т.н.,доц.,
_____Колесник І.С.
Виконав:
Магістрант 2 курсу
_____Вівчарик Д.В.

Вінниця2024

1 Підстава виконання магістерської кваліфікаційної роботи

Наказ про затвердження теми магістерськоїкваліфікаційної роботи від 17.09.2024 р. №310.

2 Мета і призначення КМКР

2.1 Метою є вдосконалення методу і засобів опрацювання інформації та розробка ефективної інфраструктури для IoT, ґрунтуючись на запропонованій архітектурі.

2.2 Призначення розробки — виконання магістерської кваліфікаційної роботи.

3 Вихідні дані для виконання МКР

Вихідні дані: технічна документація , засоби та архітектура IoT, мережеві протоколи MQTT, XMPP.

4 Вимоги до виконання КМКР

МКР повинна задовільняти такі вимоги:

— проведення огляду і аналізу технологій IoT;

— оцінка сучасного розвитку IoT;

— наліз наявних моделей IoT;

— розгляд методів збору та обробки даних в IoT ;

— оцінка підходів до збору та обробки даних та організації інфраструктури IoT;

— вдосконалення методів та засобів опрацювання інформації IoT з урахуванням запропонованої архітектури.

5 Етапи МКР та очікувані результати наведені в табл.. А.1

6 Технічна документація до IoT

ТаблицяА.1 —ЕтапиМКР

| №з/п | Назва етапів виконання магістерської роботи | Строк виконання етапів роботи |
|---|---|---|
| | Постановка мети та задач роботи | 21.10.24 |
| 2 | Аналіз потенціалу та можливостей IoT | 25.10-30.10.24 |
| 3 | Перспективи розвитку та існуючі проблеми IoT | 31.10-08.11.24 |
| 4 | Огляд і аналіз аналогів технології IoT | 09.11-15.11.24 |
| 5 | Аналіз моделей IoT | 16.11-.20.1124 |
| 6 | Методи та засоби збору та обробки даних для IoT | 21.11-25.11.24 |
| 7 | Організація інфраструктури IoT | 26.11-31.11.24 |
| 8 | Тестування системи | 01.12-03.12.24 |
| 9 | Розрахунок економічної частини роботи | 04.12-07.12.24 |
| 10 | Оформлення пояснювальної записки та ілюстративного матеріалу | 08.12.24 |
| 11 | Аналіз виконання роботи, висновки, додатки | |
| 12 | Перевірка якості виконання магістерської роботи та усунення недоліків | |

7 Матеріали,щоподаютьсядозахистуКМКР

До захисту МКР подаються: пояснювальна записка МКР, ілюстративні та графічні матеріали, протокол попереднього захисту МКР на кафедрі, відгук наукового керівника, відгук опонента, протоколи складання державних екзаменів, анотації до МКР українською та іноземною мовами, довідка про відповідність оформлення МКР діючим вимогам.

8 ПорядокконтролювиконаннятазахистуМКР

ВиконанняетапівграфічноїтарозрахунковоїдокументаціїМКРконтролюєть сянауковимкерівникомзгіднозівстановленимитермінами.ЗахистМКРвідбуваєтьс яназасіданніДержавноїекзаменаційноїкомісії, затвердженоюнаказомректора.

9 Вимогидо оформленняМКР

9.1 При оформленні КМКР використовується:

— ДСТУ3008:2015«Звітивсферінаукиітехніки.Структуратаправилаофор млювання»;

— ДСТУ8302:2015«Бібліографічніпосилання.Загальніположеннятаправи ласкладання»;

— Методичні вказівки до виконання магістерських кваліфікаційних робіт зі спеціальності 123 — «Комп’ютерна інженерія». Кафедра обчислювальної техніки ВНТУ 2022.

9.2 Порядок виконання МКР викладено в «Положення про кваліфікаційні роботи на другому (магістерському) рівні вищої освіти СУЯ ВНТУ–03.02.02П.001.01:21

# ДОДАТОК Б

Модельзапиту/відповіді

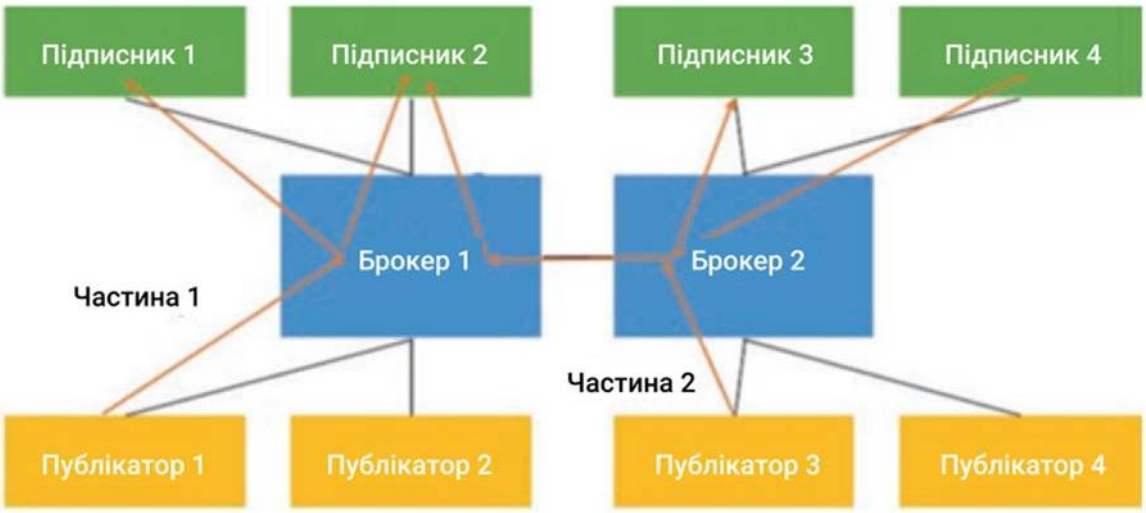



Рисунок Б.1 — Модельзапиту/відповіді

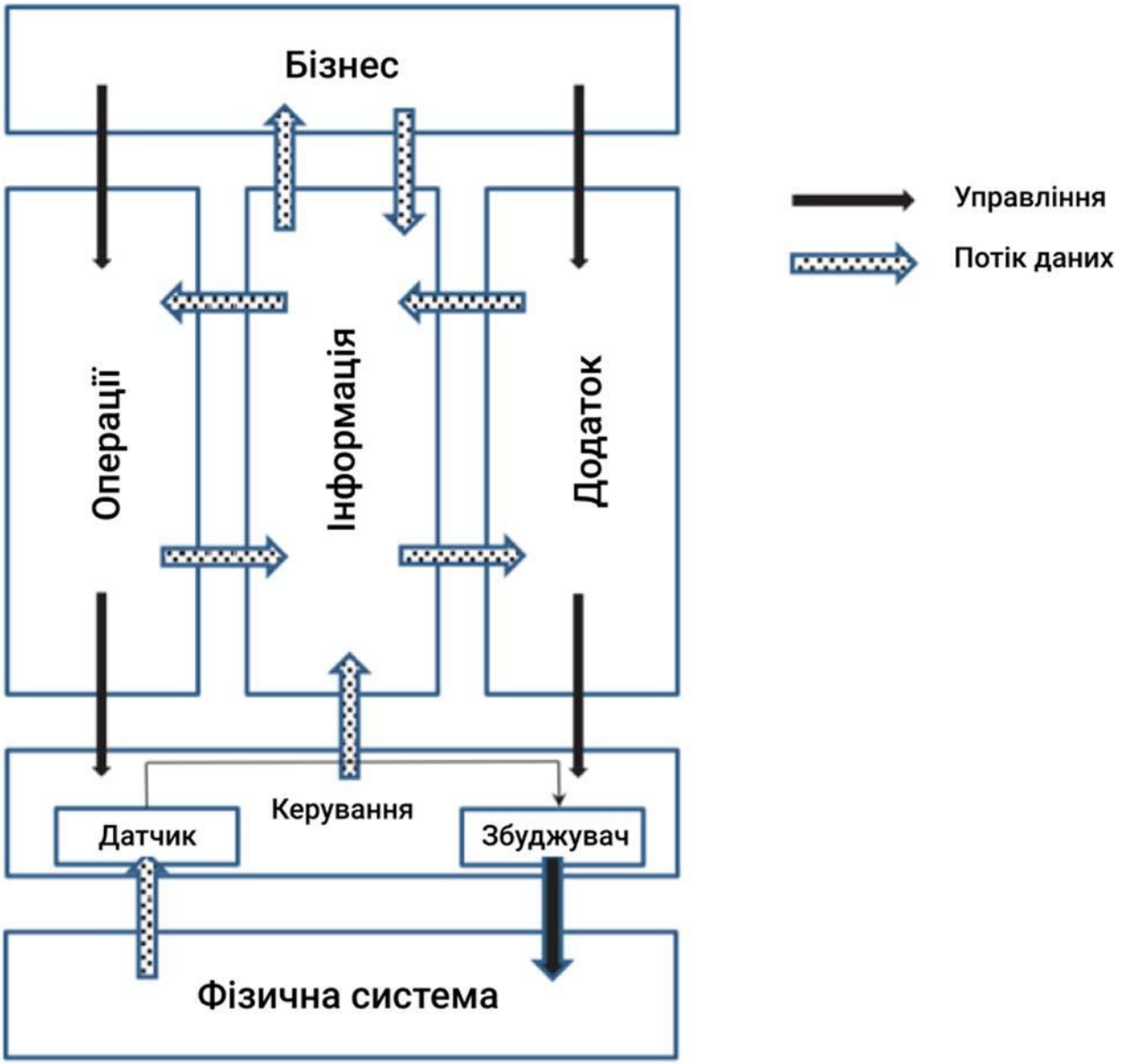
# ДОДАТОК В

Протоколи для Bluetooth Classic


Рисунок В.1— Стек протоколів для Bluetooth Classic

# ДОДАТОК Г

Функціональні області еталонної архітектури ІІС




РисунокГ.1 — Функціональні області еталонної архітектури ІІС

# ДОДАТОК Д

Прокол перевірки навчальної (кваліфікаційної) роботи

Назва роботи:_____Методи та засобиопрацюванняінформації для IoT_____

Тип роботи:_____Магістерська кваліфікаційна робота_____

(кваліфікаційна робота, курсовий проект (робота), реферат, аналітичний огляд, інше (зазначити))

Підрозділ _____кафедра обчислювальної техніки_____

(кафедра, факультет (інститут), навчальна група)

Керівник_____Колесник І.С., доц. каф. ОТ_____

(прізвище, ініціали, посада)

Показники звіту подібності

| Turnitin | |
|---|---|
| Оригінальність | 72% |
| Загальнасхожість | 28% |

Аналіз звіту подібності (відмітитипотрібне)

- ✓ Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.
- ☐ Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її автора. Роботу направити на доопрацювання.
- ☐ Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Заявляю, що ознайомлений (-на) з повним звітом подібності, який був згенерований Системою щодо роботи (додається)

Автор_____Вівчарик Д.В,._____

(підпис) (прізвище, ініціали)

Опис прийнятого рішення

_____

_____

_____

_____

Особа, відповідальна за перевірку_____ _Захарченко С.М__

(підпис) (прізвище, ініціали)

Експерт _____ _____

(за потреби) (підпис) (прізвище, ініціали, посада)