**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ**

ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

| **ПОГОДЖЕНО** | **ДОПУСКАЄТЬСЯ ДО ЗАХИСТУ** |
|---|---|
| Декан факультету інформаційних технологій | Завідувач кафедри Комп'ютерних систем, мереж та кібербезпеки |
| _____ Болбот І.М., д.т.н, проф. | _____ Касаткін Д.Ю., к.п.н., доц. |
| підпис ПІБ, вчене звання і ступінь | підпис ПІБ, вчене звання і ступінь |
| «___»_____2024 р. | «___»_____2024 р. |

# КВАЛІФІКАЦІЙНА МАГІСТЕРСЬКА РОБОТА

**на тему: «Дослідження системи управління та моніторингу IoT елементами»**

**Спеціальність – 123 «Комп'ютерна інженерія»**

**15.04 - КМР.1999 «С» 2023.11.01.05 ПЗ**

Спеціальність: 123 «Комп'ютерна інженерія»
Освітня програма: "Комп'ютерні системи і мережі"

Гарант освітньої програми
д.т.н., доцент _____ / Шкарупило В.В./
науковий ступінь, вчене звання підпис ПІБ

Керівник магістерської кваліфікаційної роботи
д.т.н., професор _____ / Коваленко О.Є./
підпис ПІБ

Виконав: _____ /Грабко К.М./
підпис ПІБ

**КИЇВ-2024**

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ**
**ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

**«ЗАТВЕРДЖУЮ»**
**завідувач кафедри**
комп'ютерних систем, мереж та кібербезпеки
_____/ Касаткін Д.Ю., к.п.н., доц. /
підпис ПІБ, вчене звання і ступінь
«__» _____ 20__ р.

# З А В Д А Н Н Я

# ДО ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ БАКАЛАВРСЬКОЇ СТУДЕНТУ

Грабко Кіріл Миколайович
(прізвище, ім'я, по батькові)

Спеціальність (напрям підготовки): 123 «Комп'ютерна інженерія»_____

1. Тема кваліфікаційної магістерської роботи: «Дослідження системи управління та моніторингу IoT елементами»_____
затверджена наказом ректора НУБіП України від "27" грудня 2024р. №_1893 «С»_
Термін подання завершеної роботи на кафедру __2024 р.___

2. Термін подання студентом роботи до екзаменаційної комісії _08_ __06___ 2023 р.
3. Вихідні дані до роботи: дослідження існуючих засобів, визначення методів управління та моніторингу елементами Інтернет речей; засоби, протоколи, апаратне та програмне забезпечення, за допомогою якого відбувається управління та моніторинг елементами IoT._____
Об'єкт дослідження – система, яка забезпечує моніторинг та управління компонентами елементами IoT.
Перелік графічного матеріалу (за потреби) _____
_____

**Дата видачі завдання "27" грудня 2024 р**.

**Керівник магістерської роботи** _____ _Коваленко О.Є., д.т.н., проф.__
( підпис ) (прізвище та ініціали)
**Завдання прийняв до виконання** _____ ____Грабко К.М._____
(підпис ) (прізвище та ініціали студента)

# КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів дипломного проекту (роботи) | Строк виконання етапів проекту (роботи) | Примітка |
|---|---|---|---|
| 1 | Постановка задачі магістерської роботи | 15.11.2023 | Виконано |
| 2 | Аналіз предметної області | 13.04.2024 | Виконано |
| 3 | Проектування системи | 20.05.2024 | Виконано |
| 4 | Реалізація системи | 27.06.2024 | Виконано |
| 5 | Тестування розробленої системи | 11.09.2024 | Виконано |
| 6 | Оформлення пояснювальної записки | 18.10.2024 | Виконано |
| 7 | Оформлення графічного матеріалу | 26.10.2024 | Виконано |
| 8 | Постерний передзахист | 07.11.2024 | Виконано |

**Студент** _____ Грабко К. М.
( підпис ) (ініціали та прізвище)

**Керівник роботи** _____ Коваленко О.Є
( підпис ) (ініціали та прізвище)

# РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи «Дослідження системи управління та моніторингу IoT елементами»: 90 сторінок, 43 рисунка, 11 таблиць, 34 використаних джерел.

ДОСЛІДЖЕННЯ, КОМП'ЮТЕРНА СИСТЕМА, МОНІТОРИНГ, УПРАВЛІННЯ, ІОТ, МІКРОКОМП'ЮТЕР, RASPBERRY PI 4, СЕРВЕР, ДАТЧИКИ, HOME ASSISTANT, АДАПТЕР.

Мета – дослідження існуючих засобів, визначення методів управління та моніторингу елементами Інтернет речей.

Об'єкт дослідження – система, яка забезпечує моніторинг та управління компонентами елементами IoT.

Предмет дослідження – засоби, протоколи, апаратне та програмне забезпечення, за допомогою якого відбувається управління та моніторинг елементами IoT.

Робота складається з вступу, трьох розділів, висновків, додатків, списку використаних джерел; містить таблиці, рисунки, діаграми.

Перший розділ присвячено аналізу предметної області. Проводиться детальний огляд архітектурних рішень, технологій, протоколів, типів мереж, основних елементів інфраструктури IoT.

У другому розділі магістерської роботи було проведено порівняльний аналіз, апаратного обладнання та засобів моніторингу для управління елементами Інтернет речей.

Третій розділ присвячено розробці схему моніторингу та управління елементами IoT. Запропоновано методи удосконалення досліджуваної схеми.

Проведене дослідження дозволило зробити наступні висновки та запропонувати пропозиції, які мають як теоретичне, так і практичне значення.

**ЗМІСТ**

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ

ОЗП – Оперативний запам'ятовуючий пристрій
ПЗ – Програмне забезпечення
ПК – Персональний комп'ютер, Personal Computer
ЦП – Central Processing Unit, центральний процесор
AMQP – Advanced Message Queuing Protocol, розширений протокол черги повідомлень
ANR – Application Not Responding, додаток не відповідає
API – Application Programming Interface, інтерфейс програмування додатків
BLE – Bluetooth Low Energy, низькоенергетичний Bluetooth
CoAP – Constrained Application Protocol, обмежений прикладний протокол
CRC – Cyclic Redundancy Check, циклічний надлишковий код
DDoS – Distributed Denial of Service, відмова в обслуговуванні
DDS – Data Distribution Service, послуга розподілу даних
GPIO – General-Purpose Input/Output, узагальнені входи/виходи
GPS – Global Positioning System, глобальна система позиціонування
GPU – Graphics Processing Unit, графічний процесор
HD – High Definition, висока чіткість
HTTP – Hypertext Transfer Protocol, протокол передачі гіпертексту
HTTPS – Безпечний протокол передачі гіпертексту, Hypertext Transfer Protocol Secure
iOS – iPhone Operating System, мобільна операційна система, що розробляється і випускається компанією Apple
IoT – Інтернет речей, Internet of Things
IP – Internet Protocol, протокол Інтернету

IPv6 – Internet Protocol version 6, протокол Інтернету версії 6

LAN – Local Area Network, локальна обчислювальна мережа LoRa – Long Range, довгий діапазон

LoRaWAN – Long Range Wide Area Network, протокол для забезпечення бездротового зв'язку на великі відстані

LPWAN – Low-Power Wide Area Network, мережа широкого охоплення з низьким споживанням енергії

LTE – Long–Term Evolution, стандартом мобільного зв'язку

M2M – Machine–to–Machine, комунікацію та взаємодію між пристроями або системами без прямої участі людини

MCU – Microcontroller Unit, мікроконтролер

MPU – Microprocessor Unit, мікропроцесор

MQTT – Message Queuing Telemetry Transport, протоколом, який використовується для передачі повідомлень

NB–IoT – Narrowband Internet of Things, вузькосмуговий IoT

NFC – Near Field Communication, безконтактна комунікація в ближньому полі

OSI – Open Systems Interconnection, модель, що визначає протоколи і принципи комунікації в комп'ютерних мережах

RFID – Radio Frequency Identification, систему ідентифікації за допомогою радіочастотного сигналу

SDRAM – Synchronous Dynamic Random Access Memory, синхронна динамічна пам'ять із довільним доступом

TCP – Transmission Control Protocol, протокол управління передачею, для надійної передачі даних між пристроями

UDP – User Datagram Protocol, протокол датаграм користувача

UPS – Uninterruptible Power Supply, джерело безперебійного живлення

Wi–Fi – Wireless Fidelity, технологія, яка дозволяє бездротового підключатися пристроям до локальної мережі Інтернету

# ВСТУП

Інтернет став невід'ємною складовою сучасного суспільства, він необхідний для щоденного обміну, пошуку та зберігання інформації.

На сьогоднішній день, внаслідок постійного технологічного прогресу, нас оточують безліч розумних пристроїв, які змінюють наше повсякденне життя. Розумні будинки вміють автоматично регулювати освітлення та опалення, а різноманітну техніку можна віддалено керувати за допомогою смартфонів.

Термін IoT (Internet of Things), або Інтернет речей охоплює відразу кілька явищ. Це самі пристрої, які вийшли в мережу та взаємодіють між собою. Це і спосіб підключення – M2M – тобто машини–до–машини, без участі людини. Це і великі дані, які тепер генерують пристрої. Дані, які можна (і потрібно) збирати, аналізувати і надалі використовувати для керуванням системою. На сьогоднішній день до пристроїв IoT відносяться мільярди технічних засобів по всьому світу завдяки яким можливо збирати, обробляти та аналізувати великі обсяги даних, що дає змогу зробити більш обґрунтовані рішення, підвищити ефективність процесів та покращити якість життя людей.

Розгортання та управління системами IoT не є тривіальним завданням. Вимагаються комплексні підходи до архітектури, мережевого з'єднання, апаратного та програмного забезпечення. Крім того, збільшення кількості підключених пристроїв до IoT мережі вносить нові виклики та проблеми. Одна з таких проблем – це ефективний моніторинг та управління елементами IoT системи. Забезпечення безперебійної роботи, збір і аналіз даних, оптимізація ресурсів, керування пристроями та забезпечення безпеки є критичними завданнями для ефективної функціонування IoT систем.

На ринку існує безліч об’єктів, які можна автоматизувати та фірм, які пропонують свої послуги для цього. Також не варто забувати, що потреби в автоматизації та віддаленому керуванні виникають не тільки у великих підприємств,

а й у звичайних споживачів, яким потрібно пропонувати негірший продукт, але значно дешевший.

Саме тому і було обрано дану тему, щоб дослідити, що являє собою мережа інтернету речей, які технології та протоколи там застосовуються, яке мінімальне апаратне та програмне забезпечення необхідне для функціонування системи керування та моніторингу елементами Інтернет речей.

У даній роботі було проведено дослідження та розроблено систему моніторингу та управління пристроями Інтернет речей на базі міні–ПК Raspberry Pi. Проект передбачає створення компактної мережі з можливістю підключення до Інтернету. Ця система забезпечує високий рівень захисту та дозволяє контролювати та відстежувати стан датчиків за допомогою мережі Інтернет. Користувач матиме можливість безпосередньо увійти в систему та взаємодіяти з пристроями у реальному часі. Система також надає гнучкість у розміщенні різноманітних вимірювальних приладів з відповідними інтерфейсами. Вона має численні переваги, такі як енергоефективність, інтелектуальність, доступну ціну, портативність та високу продуктивність.

Мета роботи полягає в дослідженні існуючих засобів, визначення методів управління та моніторингу елементами IoT. Реалізація поставленої мети обумовила необхідністю вирішення наступних завдань:

– аналіз архітектури Інтернету Речей для розуміння його структури та компонентів;

– дослідження існуючих типів мереж та протоколів, які використовуються для з'єднання пристроїв в єдину мережу IoT;

– оцінка апаратного забезпечення, що використовується у системах IoT та його налаштування;

– аналіз та налаштування програмного забезпечення для моніторингу та управління системами IoT;

– синтез та узагальнення отриманих результатів для розробки рекомендацій та рішень, що вирішують поставлену мету.

Об’єктом дослідження є система, яка забезпечує моніторинг та управління компонентами IoT.

Предметом дослідження є засоби, протоколи, апаратне та програмне забезпечення, за допомогою якого відбувається управління та моніторинг елементами IoT.

Методи дослідження. Теоретичною і методичною основою дослідження став аналітичний метод, що дозволяє розкрити особливості та переваги різних рішень, монографічний, графічний, статистичний, розрахунковий методи. Використання цих методів дослідження дозволяє отримати обґрунтовані висновки та рекомендації щодо розгортання та управління системами IoT з погляду їх ефективності та конкурентоспроможності.

Інформаційною базою досліджень є наукова література, статистичні дані, актуальні дослідження та публікації у галузі IoT технологій, систем моніторингу та управління. Ресурси мережі Інтернет, дані з практичних досліджень, власні експерименти та спостереження.

Робота складається з вступу, 3 розділів, висновків; містить 90 сторінок тексту, 43 рисунків, 11 таблиць. Список використаних джерел складається з 34 найменувань.

# РОЗДІЛ 1
# ОГЛЯД ТА АНАЛІЗ АРХІТЕКТУРНИХ РІШЕНЬ МЕРЕЖІ ІНТЕРНЕТУ РЕЧЕЙ

Інтернет речей отримав свою назву завдяки Кевіну Ештону, підприємцю, який був одним із співзасновників центру Auto–ID на Технічному університеті в Бостоні. Він та його команда придумали концепцію пов'язування об'єктів з Інтернетом за допомогою міток радіочастотної ідентифікації (RFID). У 1999 році Ештон вперше використав термін «Інтернет речей», який з тих пір став широко використовуваним (рис. 1.1).

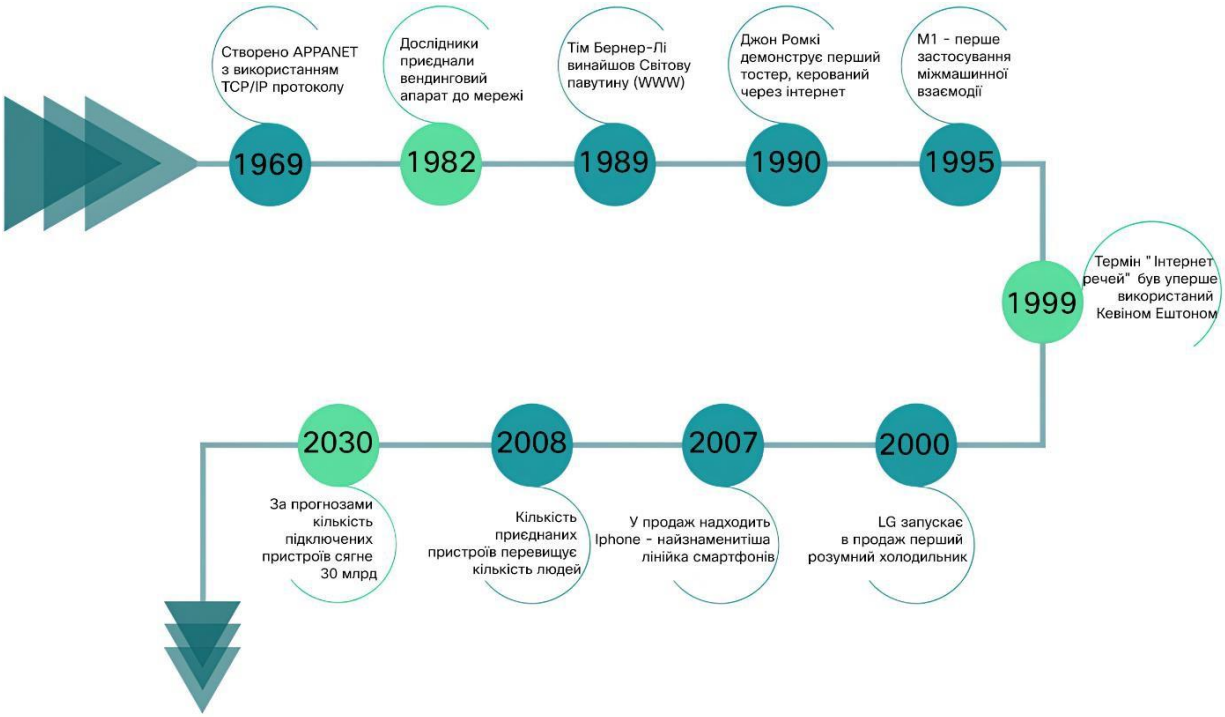


Рисунок 1.1 – Розвиток Інтернет речей

Інтернет речей (IoT – Internet of Things) – це концепція обчислювальної мережі фізичних об'єктів, які мають вбудовані технології для обміну інформацією між собою та навколишнім середовищем. Ці об'єкти можуть включати різні пристрої, датчики, автоматичні системи та інші фізичні об'єкти, які спроможні підключатися до

Інтернету та взаємодіяти з ним для виконання різноманітних завдань і операцій. Інтернет речей є актуальною та динамічно розвиваючою галуззю досліджень. Вона представляє модифіковану та інтегровану версію звичайного Інтернету. За прогнозами, кількість пристроїв Інтернет речей (IoT) у світі майже подвоїться з 15,1 мільярда у 2020 році до понад 29 мільярдів у 2030 році. Таким чином кількість пристроїв, що підключаються до мережі, зростає щодня, а їх з'єднання за допомогою проводових та бездротових технологій надає нам потужне джерело інформації, доступне нам на кінчиках пальців. Ідея Інтернету речей полягає в розширенні можливостей для взаємодії між розумними машинами та передовою технологією. Варто відзначити, що технології, що лежать в основі Інтернету речей, не є новими [1].

Моніторинг та контроль пристроїв стали невід'ємною частиною нашої безпеки, комфорту та зручності. За допомогою Інтернет–технологій та бездротових сенсорних мереж, ми можемо відстежувати та керувати пристроями в реальному часі. Користувачі мають великі очікування від Інтернет–підтримуваних систем, оскільки вони дозволяють віддалено відстежувати, діагностувати, налагоджувати та оновлювати програмне забезпечення, що дозволяє знизити витрати на обслуговування та експлуатацію [1].

Можливість контролювати розумний будинок та Інтернет–прилади забезпечує зручність та безпеку, особливо, коли власників немає вдома. За допомогою дистанційного моніторингу можна стежити за станом житлових та промислових об'єктів і отримувати сповіщення в разі пожежі, несанкціонованого проникнення або витоку рідини чи газу, що дозволяє вчасно повідомити про загрозу аварійним службам [8].

Зараз Інтернет речей набуває все більшої популярності у нашому щоденному житті, і це стало можливим завдяки наступним тенденціям та факторам. По–перше, розвиток сучасних цифрових технологій дозволило створити пристрої IoT більш доступнішими. По–друге, значно збільшилась кількість пристроїв, які мають можливість підключатись до Інтернету за допомогою Wi–Fi. Крім того, смартфони стали невід'ємною складовою нашого повсякденного життя та дозволяють нам керувати іншими пристроями за допомогою своїх функцій [11].

В наш час, Інтернет речей широко використовується в різних галузях (рис. 1.2).

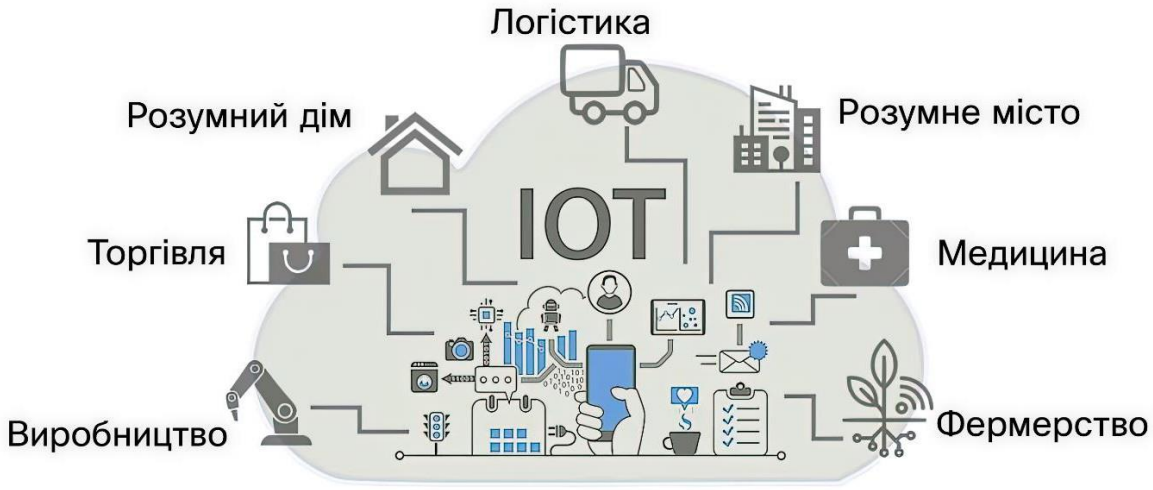


Рисунок 1.2 – Застосування інтернету речей

Проте, на сьогоднішній день можна виділити кілька основних напрямків, де він успішно застосовується – логістика, розумний дім і місто, медицина, виробництво, торгівля, фермерство.

Інтернет речей можна класифікувати за категоріями, такими як локальний та глобальний IoT, дротовий та бездротовий, а також побутовий та промисловий [8].

Глобальний або всесвітній Інтернет речей об'єднує пристрої, що розташовані на великій відстані один від одного, і вони зазвичай спілкуються між собою за допомогою бездротового з'єднання. Локальний Інтернет речей використовує як дротові, так і бездротові з'єднання. Наприклад, система «Розумний дім» є прикладом локального, закритого проекту, де всі пристрої взаємодіють між собою, а доступ до них має лише власник приміщення. Це забезпечує безпеку та виключає можливість зовнішнього втручання у цю екосистему [16].

Дротовий Інтернет речей широко використовується на об'єктах з обмеженим доступом, де безпека є головним аспектом для запобігання несанкціонованому доступу або втрати цінних даних. Використання провідного підходу обумовлено високою надійністю передачі і спеціальними вимогами до середовища. Наприклад, на промислових об'єктах, де пристрої спілкуються між собою, використовується

провідне підключення, а оператори можуть спостерігати за станом датчиків і виконавчих пристроїв на моніторах. З іншого боку, бездротовий Інтернет речей є найпопулярнішим способом організації взаємодії пристроїв. Його популярність пояснюється високою швидкістю передачі даних та мобільністю. Бездротовий Інтернет речей стає доступним для кожної людини через різноманітні пристрої.

У промисловому середовищі, такому як виробництва, склади, заводи і лабораторії, широко використовується промисловий Інтернет речей з метою автоматизації та оптимізації виробничих процесів. Його головна мета полягає в полегшенні та автоматизації роботи в промислових умовах. З іншого боку, в домашньому оточенні застосовується побутовий Інтернет речей, де різні побутові пристрої об'єднуються між собою. Він використовується для автоматизації повсякденних завдань та віддаленого керування різними пристроями.

## 1.1 Дослідження технологій, протоколів та типів мережі у контексті Інтернету речей

IoT є концепцією, що дозволяє з'єднати фізичні об'єкти з цифровим світом, утворюючи мережу для обміну даними та забезпечення зв'язку. Аналіз технологій, що використовуються в Інтернет речах, включає розгляд різних аспектів, таких як засоби збору даних, протоколи комунікації, хмарні та крайові обчислення, безпека та приватність, аналітика даних, інтеграція та стандартизація. Цей детальний аналіз допоможе краще зрозуміти можливості цих технологій та їх вплив на розвиток систем IoT [19].

У системах Інтернет речей застосовуються різноманітні типи мереж, вибір яких залежить від конкретних потреб та ситуацій. Окремі типи мереж використовуються для підключення пристроїв з обмеженим живленням та передачі даних на великі відстані, інші – для локального з'єднання на короткій відстані, а є й такі, що забезпечують підключення до Інтернету на широкій території.

Вибір конкретного типу мережі залежить від вимог та умов конкретного проекту. До основних типів мереж, які зустрічаються в побудові систем IoT, включають наступні види [3].

Мережі з низьким енергоспоживанням і обмеженим діапазоном є ідеальним варіантом для використання в домі, квартирі, офісах та в інших невеликих приміщеннях. Для їх застосування можна використовувати невеликі акумулятори, а іноді повністю обійтися без них, при цьому більшість з цих мереж є економічними в експлуатації. Наведемо найпоширеніші приклади:

– Wi–Fi (Wireless Fidelity) є найпоширенішою технологією бездротового зв'язку, оскільки вона має низьку вартість експлуатації. Wi–Fi є стандартним варіантом для домашнього та робочого використання і ґрунтується на стандартах IEEE 802.11. Однак, не всі сценарії підходять для використання Wi–Fi через обмежений діапазон дії та постійне споживання енергії.

– Bluetooth, використовує короткодіючу радіохвильову технологію, забезпечує високу швидкість передачі даних і надсилає сигнали голосу та інші типи інформації на відстань до 10 метрів [4].

– Zigbee є популярним вибором для систем автоматизації в домашньому середовищі та медичних пристроїв. Він найкраще підходить для створення особистих мереж з невеликими пристроями, які споживають мало енергії, мають обмежену пропускну здатність і використовуються в обмеженому діапазоні.

– Z–Wave є ще однією бездротовою технологією, спеціально розробленою для мереж Інтернету речей з низьким рівнем енергоспоживання. Вона також працює на низькочастотному діапазоні і забезпечує надійну взаємодію між системами автоматизації в домашньому середовищі на рівні додатків.

Мережі з низьким енергоспоживанням та великим діапазоном покриття (Low–Power Wide Area Networks – LPWAN) є одними з ключових технологій для бездротових мереж Інтернет речей. Ці мережі спеціально розроблені для забезпечення зв'язку на великій території з низьким споживанням енергії і використовуються для більшості пристроїв IoT. Наприклад, LoRaWAN (Long Range
Wide Area Network) створена для забезпечення бездротового зв'язку на великі

відстані, в містах, селах, регіонах, тощо. Ця технологія відрізняється високою проникністю і дальністю покриття. Вона забезпечує зв'язок між мобільними пристроями, що працюють від акумулятора, і мають двонапрямну захищену передачу даних. Нижче наведені найпоширеніші приклади таких технологій:

– Cellular Networks (Стільникові мережі) 4G LTE (Long–Term Evolution) покоління мобільних мереж забезпечує високу швидкість передачі даних, надійність та низьку затримку. Воно підтримує велику кількість одночасних підключень та дає змогу отримати відомості чи оновлення в реальному часі.

– 5G (Fifth Generation) найновіша генерація мобільних мереж пропонує ще більшу швидкість передачі даних, знижену затримку та підвищену пропускну здатність. Вона також забезпечує покращену масштабованість для підключення великої кількості IoT–пристроїв;

– Cat–0 відносяться до стандартів передачі даних у мережах з низькою швидкістю на основі LTE. Він забезпечує можливість передачі текстових повідомлень, контролю датчиків, віддаленого керування та інших сценаріїв, де не вимагається передача великих обсягів даних.

– Cat–1 є стандартом для мобільного IoT в кінцевому підсумку замінить 3G. Мережі Cat–1 легко налаштовувати. Це відмінний варіант для додатків, яким потрібен голосовий або браузерний інтерфейс.

– LTE Cat–M1, також відомий як LTE–M, є стандартом мережі з низьким енергоспоживанням, який дозволяє підключати мільйони пристроїв до однієї мережі, забезпечуючи надійний і ефективний обмін даними в IoT. Він працює на базі технології LTE і забезпечує зв'язок на відстані до кількох кілометрів.

– NB–IoT (Narrowband IoT) технологія працює на основі мобільних мереж і забезпечує низьку швидкість передачі даних, проте витрачає дуже мало енергії. Вона ідеально підходить для простих пристроїв Інтернет речей, які вимагають довгого терміну служби на одному заряді батареї.

– Cat–M2 використовує технологію LTE (Long–Term Evolution) із спеціальною категорією M2, яка забезпечує підключення IoT пристроїв. Він пропонує більш

високу швидкість передачі даних порівняно з NB–IoT, але все одно залишається енергоефективним.

– Sigfox є провідним міжнародним постачальником мережевих послуг для Інтернет речей, який пропонує бездротові мережі для підключення пристроїв з невисоким енергоспоживанням, які постійно генерують дані.

Після детального розгляду різних технологій та типів мережі, наступним кроком у нашому дослідженні є аналіз протоколів, які використовуються у мережах IoT [2].

Протоколи в контексті Інтернету речей використовуються для забезпечення комунікації та взаємодії між різними пристроями, які мають можливість підключення до Інтернету. Інтернет речей передбачає підключення фізичних пристроїв, таких як сенсори, виконавчі пристрої та інші «речі», до глобальної мережі, щоб вони могли обмінюватися даними та взаємодіяти один з одним.

Протоколи визначають правила та формати, за якими пристрої мають обмінюватися даними. Вони включають в себе стандарти та специфікації, які дозволяють різним пристроям розуміти один одного та ефективно спілкуватися. Це забезпечує сумісність та взаємодію між різними пристроями, виробниками та додатками в мережі IoT.

Вибір протоколу Інтернету речей залежить від рівня архітектури системи, на якому потрібно передавати дані. Модель OSI надає структуру різних рівнів, на яких здійснюється передача даних. Кожен протокол в системі Інтернету речей забезпечує взаємодію між пристроями, пристроєм і шлюзом, шлюзом і центром обробки даних або шлюзом і хмарою, а також обмін даними між центрами обробки даних.

Рівень додатків в системі Інтернету речей виступає інтерфейсом для обміну даними між користувачем і пристроєм. На цьому рівні знаходяться протоколи, які визначають команди, формати та методи обміну даними між пристроями та додатками. Він забезпечує спосіб комунікації, що дозволяє користувачам взаємодіяти з пристроями IoT, надсилати команди, отримувати дані та керувати функціями пристроїв.

– MQTT (Message Queuing Telemetry Transport) легкий протокол передачі повідомлень, який використовується для ефективної комунікації між пристроями IoT. MQTT забезпечує масштабовану розсилку повідомлень в мережі, де пристрої можуть підписуватись на топіки (теми) та отримувати повідомлення, які відповідають цим топікам.

– AMQP (Advanced Message Queuing Protocol) протокол передачі повідомлень, який забезпечує потужну та гнучку комунікацію між пристроями IoT. AMQP дозволяє створювати складні повідомлення зі структурою, метаданими та властивостями, а також підтримує різні сценарії доставки повідомлень, включаючи надійну доставку та підтвердження отримання.

– CoAP (Constrained Application Protocol) протокол, призначений для передачі даних в обмежених умовах, таких як пристрої з обмеженими ресурсами (наприклад, низька потужність, обмежена мережева пропускна здатність). CoAP використовує просту структуру запит–відповідь для ефективної комунікації між пристроями IoT та серверами, забезпечуючи низьку затримку мережі та економію енергії.

– DDS (Data Distribution Service) універсальний протокол зв'язку, розроблений для роботи з різноманітними пристроями, починаючи від невеликих пристроїв до високопродуктивних мереж. DDS спрощує процес розгортання, підвищує надійність, зменшує складність комунікації та надає зручний механізм для передачі даних між пристроями в розподілених системах.

– HTTP (Hypertext Transfer Protocol) широко використовуваний протокол передачі даних в Інтернеті. В IoT, HTTP використовується для комунікації між пристроями IoT та хмарними платформами, дозволяючи передавати дані у форматі запит–відповідь. HTTP підтримує різні методи запитів, такі як GET, POST, PUT, DELETE, що дозволяє взаємодіяти з даними та виконувати різні дії на пристроях IoT.

– WebSocket протокол, який забезпечує двосторонній зв'язок між клієнтом і сервером через одне з'єднання TCP. В IoT, WebSocket використовується для забезпечення постійного з'єднання між пристроями та іншими вузлами мережі, дозволяючи передавати дані в режимі реального часу без необхідності повторних підключень [12].

Рівень транспорту забезпечує надійну передачу даних між рівнями комунікаційної моделі, а також відповідає за захист цих даних. Один з основних протоколів на рівні транспорту – це Transmission Control Protocol (TCP).

TCP забезпечує надійну передачу даних між двома пристроями у мережі. Це означає, що він гарантує, що дані будуть доставлені у правильному порядку та без пошкоджень. Це досягається шляхом встановлення з'єднання між відправником і отримувачем, розбиття даних на пакети, нумерації пакетів для відновлення порядку, підтвердження отримання та повторної передачі в разі втрати або пошкодження.

Крім TCP, існує також User Datagram Protocol (UDP), який також працює на рівні транспорту. UDP є протоколом без з'єднання та ненадійним транспортувальником даних. Він не забезпечує механізми контролю передачі та відновлення помилок, тому використовується для швидкої передачі даних, де втрата окремих пакетів не є критичною.

Мережевий рівень допомагає окремим пристроям взаємодіяти з маршрутизаторами або іншими мережевими вузлами в IoT–системі.

Він виконує розподіл адрес IP та маршрутизацію даних між пристроями, що знаходяться в одній мережі або між різними мережами. Кожен пристрій в IoT–системі може мати унікальну IP–адресу, яка дозволяє ідентифікувати його в мережі. Рівень мережі використовує протоколи IP і мережеві технології, такі як Ethernet, Wi–Fi, Bluetooth або Zigbee, для забезпечення зв'язку між пристроями та маршрутизаторами.

– Internet Protocol (IP) протокол використовується для маршрутизації пакетів даних в мережі IoT. Він визначає формат пакетів даних та присвоює унікальні IP-адреси пристроям, що дозволяє їм взаємодіяти в мережі.

– Internet Protocol version 6 (IPv6) нова версія протоколу IP, яка створена для забезпечення більшої кількості доступних IP–адрес. З урахуванням великої кількості IoT–пристроїв, IPv6 надає більше адрес для забезпечення їх підключення до мережі та взаємодії між собою.

–6LoWPAN (IPv6 over Low-Power Wireless Personal Area Networks) протокол, який дозволяє передавати пакети IPv6 через бездротові мережі з низьким споживанням енергії, такі як Zigbee або Bluetooth.

На канальному рівні дані передаються в межах архітектури системи, а також здійснюється виявлення та виправлення помилок, які можуть виникати на фізичному рівні.

Протоколи на канальному рівні забезпечують надійну передачу даних між IoT–пристроями, включаючи в себе методи виявлення та корекції помилок. Це може включати використання контрольних сум (checksums), перевірки циклічного резервування (CRC – cyclic redundancy check) або інших методів для виявлення помилок у переданих даних.

Якщо під час передачі даних виявляються помилки, на канальному рівні можуть застосовуватись різні методи виправлення помилок. Наприклад, може бути використана повторна передача пакетів з помилками або виправлення помилок за допомогою додаткової інформації, яка передається разом з даними.

– IEEE 802.15.4 стандартний бездротовий протокол, який використовується для мереж IoT з низьким енергоспоживанням. Він визначає фізичний та канальний рівні, що дозволяє передавати дані на короткі відстані з низьким споживанням електроенергії. Він використовується разом з протоколом Zigbee, стандартом 6LoWPAN та іншими стандартами для створення бездротових мереж.

– LPWAN (Low–Power Wide Area Network) є одним з ключових протоколів, який забезпечує дальній зв’язок та енергоефективну комунікацію для пристроїв IoT.

– LPWAN протоколи, такі як LoRaWAN (Long Range Wide Area Network) створений для забезпечення бездротового зв'язку на великі відстані, такі як міста, села або регіони. LoRaWAN використовує протокол LoRa (Long Range), який дозволяє передавати дані на великі відстані з низьким споживанням енергії. Ця технологія побудована на концепції сполучних вузлів (gateways), які приймають дані від кінцевих пристрої. Особливістю LoRaWAN є його висока проникність і дальність покриття. Завдяки низькій швидкості передачі даних, LoRaWAN може працювати на значно більших відстанях, ніж інші технології бездротового зв'язку, такі як NB–IoT або LTE–M.

На фізичному рівні здійснюється обмін даними між фізичними пристроями, такими як датчики, актуатори, розумні пристрої тощо. На цьому рівні IoT–мережі

передають електромагнітні сигнали, які кодують інформацію, що необхідна для збирання, обробки та передачі даних. Ці сигнали можуть бути передані за допомогою радіохвиль, інфрачервоного випромінювання або інших методів залежно від використовуваних протоколів та технологій.

– Ethernet один з варіантів підключення IoT–пристроїв до локальної мережі або хмарної інфраструктури.

– Ethernet є однією з основних технологій, що використовуються в LAN. Вона забезпечує передачу даних по фізичному кабелю, такому як вита пара (UTP/STP), оптичний кабель (fiber optic) та коаксіальний (coaxial).На фізичному рівні IoT може використовувати Ethernet для підключення пристроїв, збирання даних з датчиків, керування та передачі інформації до різних платформ для подальшої обробки.

– Bluetooth з низьким споживанням енергії (BLE) представляє собою технологію, що дозволяє суттєво зменшити енергетичні витрати. Він здатен працювати в тому ж самому діапазоні підключення, що й класичний Bluetooth. BLE безпосередньо вбудований у мобільні операційні системи, що сприяє його широкому поширенню. Спеціально розроблена для побутової електроніки, і швидко набуває популярності завдяки своїй економічності та здатності забезпечувати тривалу автономну роботу від акумулятора [4].

– Long–Term Evolution (LTE) LTE є стандартом бездротового зв'язку, розробленим для передачі даних на широкій області покриття з високою швидкістю і низькою затримкою. Він використовує радіохвильовий спектр і мережеву інфраструктуру для передачі та обробки даних та може бути використаний для передачі відео, аудіо, зображень та іншої інформації в IoT–системах.

– NFC (Near Field Communication) набір протоколів зв'язку, що використовує електромагнітні поля для передачі інформації на невеликій відстані (зазвичай до 10 см). За допомогою NFC можна здійснювати безконтактні платежі, аутентифікацію пристроїв та забезпечувати надійний обмін інформацією.

– RFID (Radio Frequency Identification) використовується в IoT для ідентифікації та взаємодії з об'єктами за допомогою безконтактного радіочастотного зв'язку. Вона включає в себе теги RFID, зчитувачі та системи керування даними.

Основна ідея технології RFID полягає у тому, що об'єкти (такі як товари, пристрої, транспортні засоби тощо) мають вбудовані RFID–теги, які містять унікальний ідентифікатор та можуть передавати дані за допомогою радіохвиль. Зчитувачі RFID здатні безпосередньо або на відстані зчитувати ці теги, що дозволяє автоматично ідентифікувати та отримувати дані про об'єкти.

– Протокол Wi–Fi використовується на фізичному рівні для забезпечення бездротового зв'язку між різними пристроями Інтернет речей. Цей протокол дозволяє підключати різні пристрої до однієї мережі та здійснювати комунікацію між ними. Однак, варто враховувати, що він має обмежений діапазон дії та споживає постійну енергію.

Після аналізу типів мереж у контексті Інтернету речей, було визначено, що основними технологіями, які широко використовуються, є ZigBee, Wi–Fi, Bluetooth Low Energy (BLE), через їхні універсальні можливості і широкий спектр застосувань. Кожна з цих технологій має свої унікальні особливості та застосування в різних сферах IoT.

В таблиці 1.1 наведена порівняльна характеристика основних мережевих технологій.

Таблиця 1.1 – Порівняння основних мережевих технології

| Технологія | ZigBee | Wi–Fi | BLE |
|---|---|---|---|
| Стандарт зв'язку | IEEE 802.15.4 | IEEE 802.11 | IEEE 802.15.1 |
| Швидкість передачі даних | до 0,25 Мбіт/с | до 300 Мбіт/с | до 3 Мбіт/с |
| Споживання енергії | Низьке | Високе | Дуже низьке |
| Частотний діапазон | 2,4 ГГц | 2,4 ГГц | 2.400 – 2.4835 ГГц |
| Радіус роботи | до 100 м | до 100 м | 30 м |

Продовження таблиці 1.1

| | | | |
|---|---|---|---|
| Максимальна кількість вузлів у мережі | 65535 | 32 | 7 |
| Базова топологія | Сітка (mesh) | Зірка | Piconet |

Проведено дослідження різних технологій, протоколів та типів мережі в контексті Інтернет речей. Вивчені їх переваги та недоліки, а також встановлено, які типи мережі найкраще підходять для різних сценаріїв. Правильне розуміння цих аспектів дозволяє забезпечити ефективну комунікацію між пристроями та надійну передачу даних.

## 1.2. Типи пристроїв у мережі IoT

У мережі Інтернет речей можна виділити різні типи пристроїв, які взаємодіють між собою, збирають, обробляють та передають дані.

Мікроконтролери (MCU) є ключовою складовою для забезпечення функціональності, керування та комунікації пристроїв в мережі IoT. Це невеликі комп'ютери, які вбудовані в мікросхеми і мають в собі центральний процесор (ЦП), оперативну пам'ять (ОЗП) і постійну пам'ять (ПЗУ). Мікроконтролери використовуються для керування простими функціями і виконання специфічних завдань [25].

Мікроконтролери мають низьке споживання енергії, може підтримувати різні протоколи комунікації (Wi–Fi, Bluetooth, Zigbee, LoRa) для обміну даними, мати вбудовані механізми шифрування та аутентифікації, а також роз'єми, які дозволяють підключати додаткові модулі або датчики для розширення функціональності пристрою.

Мікропроцесор (MPU) є ще одним типом пристрою, що використовується у мережі Інтернет речей. Вони мають більшу обчислювальну потужність порівняно з мікроконтролерами і здатні виконувати складніші завдання.

Ці пристрої зазвичай мають можливість підключати зовнішні пристрої та модулі для розширення функціональності, мати вбудовані механізми захисту від несанкціонованого доступу, шифрування та аутентифікації даних. Крім того, вони взаємодіють з багатьма мережевими протоколами.

Вбудовані системи – це пристрої, які об'єднують у собі як обладнання, так і програмне забезпечення. Вбудовані системи зазвичай базуються на мікропроцесорах або мікроконтролерах і можуть мати обмежені обчислювальні та ресурсні можливості. Їх головна мета – забезпечувати виконання специфічних функцій, керувати певними процесами, збирати та обмінюватися даними з іншими пристроями, що сприяє розширенню можливостей автоматизації та оптимізації різних сфер життя.

Такі вбудовані системи знаходять застосування у різних галузях. Наприклад, в промисловості вони використовуються для контролю та автоматизації виробничих процесів. В автомобільній промисловості забезпечують керування системами безпеки, комфорту та енергоефективності автомобілів. Також вбудовані системи IoT використовуються в побутовій техніці для управління побутовими пристроями, такими як холодильники, пральні машини, системи опалення та інші.

Інтелектуальні системи (також відомі як розумні системи або системи штучного інтелекту) – це пристрої, мають здатність виявляти, аналізувати та інтерпретувати дані з оточення. Вони мають більшу обчислювальну потужність і можуть виконувати складніші завдання, порівняно з вбудованими системами [26].

Інтелектуальні системи в мережі IoT застосовуються в різних сферах. Наприклад, в смарт–будинках вони контролюють освітлення, опалення, кондиціонування повітря, тощо. В розумних містах вони керують освітленням вулиць, оптимізують транспортний рух та відстежують якість повітря. В галузі здоров'я і медицини вони допомагають контролювати стан здоров'я, нагадувати про прийом ліків та забезпечувати дистанційну медичну діагностику. В сільському

господарстві вони оптимізують полив, контролюють якість ґрунту та рослин, а також прогнозують погодні умови.

Пристрої, які не призначені для обчислень – ці пристрої функціонують як точки з'єднання і передачі даних. Вони можуть включати сенсори, актуатори або інші пристрої, які збирають і передають інформацію про оточуюче середовище, але не мають можливості виконувати складні обчислення.

Перетворювачі – це пристрої, які забезпечують зв'язок та взаємодію між фізичними пристроями та мережею Інтернет речей. Вони перетворюють дані з датчиків або інших пристроїв в цифровий формат, що дозволяє забезпечити моніторинг, оптимізацію та автоматизацію процесів у різних галузях.

### 1.3 Елементи інфраструктури IoT

Для подальшого вибору відповідного обладнання та розуміння роботи системи IoT важливим етапом є аналіз інфраструктури Інтернет речей.

Інтернет речей привертає увагу завдяки своєму потенціалу полегшити та покращити наше повсякденне життя. Він дозволяє фізичним пристроям, обладнанню та датчикам спілкуватися та обмінюватися даними через мережу, що відкриває безліч можливостей для покращення ефективності, комфорту та безпеки повсякденного життя. Однак, для повного використання потенціалу IoT потрібна надійна та ефективна архітектура, яка забезпечить безперебійну роботу всіх компонентів системи. Аналіз інфраструктури є важливим етапом у розробці та впровадженні системи IoT. Він допомагає зрозуміти зв'язки, вимоги та проблеми, що виникають під час створення цієї складної системи [12].

Інтернет речей охоплює широку екосистему різноманітних інструментів. Розуміння основних компонентів та методів їх інтеграції може стати корисним при проектуванні різних систем.

На базовому рівні, IoT означає будь–яку систему зв'язаних пристроїв, які мають датчики і вбудовані обчислювальні можливості. Важливо зазначити, що для цих пристроїв не обов'язкове використання Інтернету. Навіть локально підключені пристрої, які взаємодіють та обмінюються даними, можуть утворювати систему IoT.

Маючи це на увазі, ми можемо розглянути складові інфраструктуру для створення системи IoT і поділити її на окремі елементи (табл. 1.2).

Таблиця 1.2 – Елементи інфраструктури IoT

| Елемент інфраструктури IoT | Опис |
|---|---|
| Датчики | Використовуються для вимірювання фізичних величин, якими IoT–пристрої діляться через мережу |
| Контролери | Мозок пристрою; виступають мостом між датчиком і мережею, виконують обчислення і зберігають дані |
| Мережа | Технологія, що використовується для обміну даними з іншими пристроями в системі або хмарою |
| Хмара | Обчислювальні ресурси, сховища та шлюзи, доступні через Інтернет |
| Користувацькі програми | Мобільні та веб–додатки, які дозволяють користувачеві взаємодіяти з системою IoT |
| Аналітика даних | Інструменти та ресурси, які дозволяють користувачам отримувати інформацію від системи IoT. |

Також важливим аспектом, що охоплює всі ці елементи інфраструктури, є забезпечення безпеки.

Датчики є важливим компонентом в IoT. Вони можуть контролювати такі умови, як температура в приміщенні за допомогою температурного датчика або відстежувати транспортний засіб за допомогою GPS. Датчики IoT зазвичай живляться від акумулятора або зовнішнього джерела постійного струму.

Багато систем Інтернету речей мають датчики, але деякі можуть обходитись без них. Наприклад, якщо ви хочете керувати розумним освітленням за допомогою програми, вам знадобиться лише IoT–контролер.

Більшість систем Інтернету речей мають датчики у своїй архітектурі, але деякі можуть обходитись без них. Наприклад, якщо потрібно керувати розумним світлом за допомогою програми, в такому випадку потрібен лише IoT–контролер.

IoT–контролер виконує важливу роль у системі Інтернет речей. Він є своєрідним мозком, забезпечуючи зв'язок між датчиками та мережею. Крім того, контролер часто виконує локальні обчислення. Сучасні контролери стають все потужнішими, маючи більшу пам'ять та обчислювальні можливості. Ця еволюція сприяє розповсюдженню периферійних обчислень, які дозволяють зближати зберігання та обробку даних ближче до джерела даних.

Як і датчики, контролери використовують батарею або зовнішнє джерело живлення.

Для того щоб бути пристроєм Інтернету речей, необхідно мати з'єднання з мережею. Якщо пристрій не підключений до мережі, то він просто функціонує як автономний обчислювальний пристрій. Для створення системи Інтернету речей пристрої повинні здійснювати зв'язок з іншими пристроями або хмарою.

Існує кілька типів підключення, які залежать від використовуваної програми. Ці типи можуть варіюватись від хмарного підключення до локального підключення на короткі відстані.

Для взаємодії пристроїв Інтернету речей з хмарою часто використовують хмарну платформу IoT. Існують два основних варіанти: створити власну хмарну інфраструктуру або скористатися послугами хмарного провайдера.

Хмарна інфраструктура для додатків Інтернету речей містить традиційні сервіси обробки даних, такі як сервіси додатків, віртуальні машини та інші рішення, що допомагають працювати з зібраною інформацією (рис. 1.3). Крім того, вона також включає сервіси зберігання, такі як бази даних кешування, тощо. Також важливою складовою хмарної інфраструктури є шлюзові сервіси, які забезпечують збір вхідних даних та взаємодію з пристроями. (HTTP/MQTT–сервер, WebSocket–сервер).

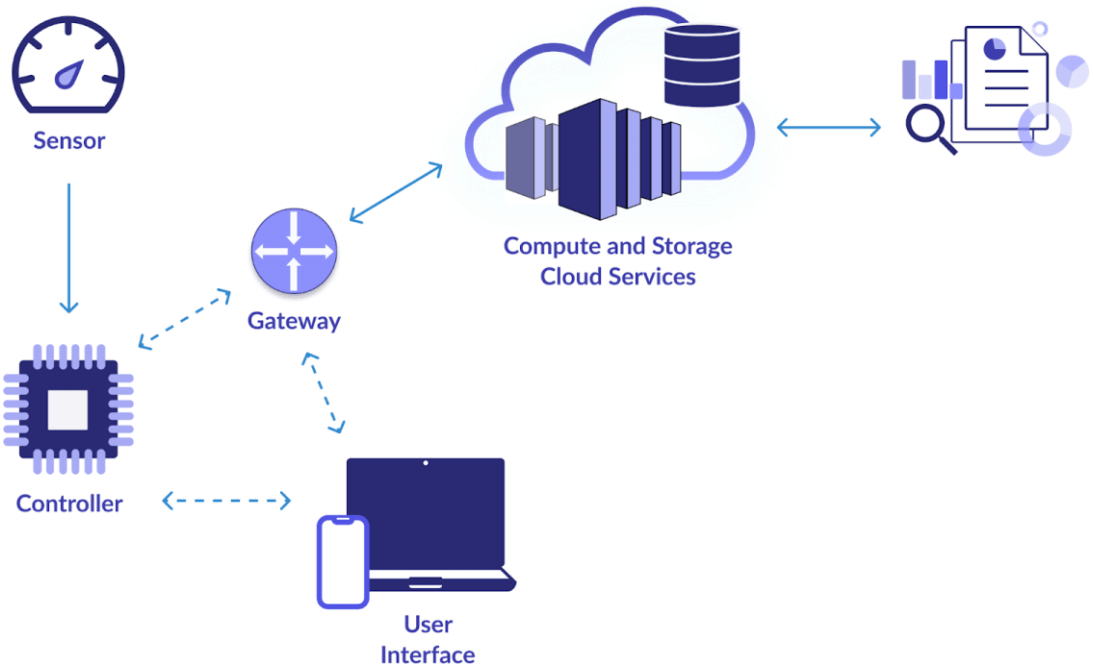


Рисунок 1.3 – Елементи інфраструктури для системи IoT

Масштабування є важливим фактором для хмарної інфраструктури, яка використовується для додатків Інтернету речей, особливо для підприємств з великою кількістю пристроїв. Це одна з причин, чому підприємства можуть відмовитися від створення власної серверної інфраструктури і перейти на використання хмарних платформ IoT. Хмарні платформи дозволяють легше масштабувати інфраструктуру, ніж будувати власні фізичні сервери.

Залежно від системи, хмарна інфраструктура може вимагати взаємодії з користувацькими додатками, які можуть бути мобільними або веб–додатками. Це дозволяє кінцевим користувачам відображати дані та надсилати команди до пристроїв. Таким чином реєстрація, вхід, відновлення пароля, а також доступ до API даних, стають додатковими аспектами хмарної інфраструктури.

У деяких випадках (особливо BLE–додатки) пристрій може безпосередньо спілкуватися з додатком кінцевого користувача без хмарного посередника [18].

Зі зростанням кількості пристроїв Інтернету речей отримується все більше даних, які ці пристрої надсилають до хмарного сервісу. Це надає можливість використовувати дані для аналізу та отримання цінної інформації. В рамках хмарної

інфраструктури можна мати систему для зберігання цих даних, а також ресурси для використання методів аналітики та машинного навчання.

Іншими словами, коли пристрої Інтернету речей надсилають дані до хмари, можна використовувати ці дані для отримання корисної інформації. Хмарна інфраструктура надає засоби для зберігання цих даних і використання різних аналітичних методів для отримання цінних висновків.

Одним з важливих аспектів інфраструктури є безпека. Вона має велике значення під час вибору та розробки кожного компонента інфраструктури. Залежно від рівня чутливості даних, може бути необхідно їх захищати під час передачі та зберігання. Наприклад, для забезпечення безпеки можна використовувати протокол HTTPS замість HTTP.

Ця вимога до безпеки може призвести до використання контролера з апаратними прискорювачами для виконання важких операцій шифрування. Також може знадобитися встановлення правил контролю доступу до хмарних баз даних та налаштування мережевих правил для серверів з метою обмеження передачі та отримання даних. При роботі з особистими даними користувачів необхідно дотримуватися відповідних правил, і регулярно проводити аудит безпеки.

Інфраструктура Інтернету речей може варіюватись залежно від конкретного рішення. Однак, основні елементи, які були описані вище, є ключовими для побудови системи, яка буде стійкою, функціональною, масштабованою, доступною, ремонтопридатною і економічно ефективною [19].

## 1.4 Моніторинг IoT

Моніторинг пристроїв IoT передбачає безперервне відстеження та управління підключеними пристроями в мережі IoT (рис. 1.4). Він дозволяє збирати в режимі реального часу дані про стан, продуктивність і безпеку пристроїв, забезпечуючи безперебійну роботу інфраструктури Інтернету речей. Завдяки моніторингу пристроїв

можна завчасно виявляти проблеми, аномалії і вжити необхідні заходи, щоб запобігти потенційним збоям.

Система IoT складається з цих основних компонентів (рис. 1.4):

– «річ» – контролери і датчики IoT;

– мережа – середовище, що використовується для передачі даних;

– хмара – віддалений сервер, що збирає та обробляє дані;

– інтерфейс користувача – мобільний або веб–додаток для кінцевих користувачів.

Моніторинг речей включає в себе моніторинг обладнання, прошивки та додатків. Для апаратного забезпечення можна відстежувати такі параметри, як напруга, рівень струму, температура та вологість. Моніторинг дозволяє виявляти коливання та налаштовувати порогові значення для сповіщень. Якщо ці рівні перевищують певний поріг, можна надіслати сповіщення та вжити необхідних заходів для виправлення ситуації.

Іншим варіантом використання апаратного моніторингу є моніторинг підключення окремих датчиків.

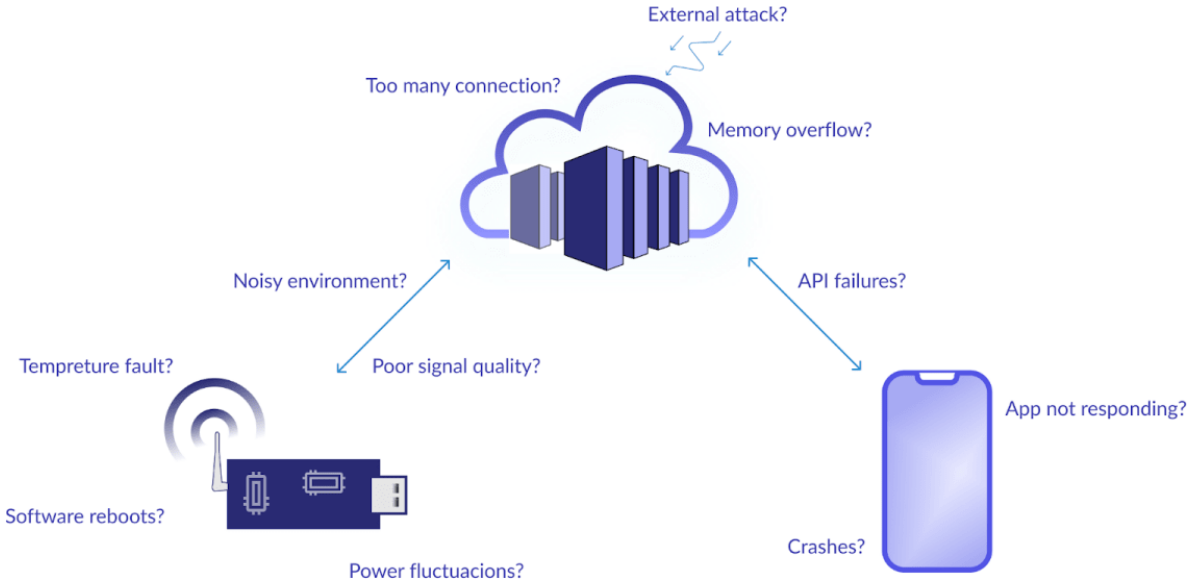


Рисунок 1.4 – Моніторинг IoT з першого погляду

Наприклад, можна періодично сканувати шину I2C, щоб переконатися, що датчики I2C активні, і здіймати тривогу, якщо датчик не реагує (що може свідчити

про проблему фізичного з'єднання). Також можна перевірити діапазон вимірювань, переданих датчиком. Якщо вони знаходяться далеко за межами діапазону (постійно 0 або максимальне значення), це може свідчити про несправність датчика.

Для моніторингу прошивки IoT можна відстежувати такі речі, як кількість перезавантажень (як апаратних, так і програмних), використання пам'яті та коди помилок на рівні драйверів. Наприклад, якщо в прошивці спостерігається надмірна кількість перезавантажень програмного забезпечення, то, можливо, в коді є помилка, яка може вимагати оновлення. Якщо використання пам'яті перевищує певний поріг, командам може знадобитися вимкнути деякі завдання з нижчим пріоритетом, доки використання пам'яті не повернеться до рівня, нижчого за поріг.

Залежно від вашого сценарію використання, можуть існувати й інші показники, які слід відстежувати. Наприклад, розглянемо програму, в якій пристрій перебуває в сплячому режимі, коли він не використовується, і переходить в активний режим при взаємодії з користувачем. Час, необхідний для переходу з режиму сну в активний режим, є специфічною метрикою для конкретного додатка.

Хорошим прикладом такого пристрою є розумний дверний замок. Він залишається у сплячому режимі, доки не виявить взаємодії з користувачем. У сплячому режимі пристрій споживає мінімум енергії, щоб підтримувати свої основні функції. Зазвичай це передбачає очікування певних сигналів для пробудження, наприклад, сигналу від датчиків, що вказують на взаємодію з користувачем.

Розумний дверний замок може перейти в активний режим, коли користувач наближається до дверей із зареєстрованим смартфоном. Замок може використовувати технологію Bluetooth Low Energy (BLE), щоб виявити наближення смартфона користувача. Як тільки смартфон користувача виявлено, розумний замок прокидається і стає готовим до використання. Після того, як користувач успішно відчинив двері, пристрій повертається в сплячий режим через заданий проміжок часу, таким чином зберігаючи заряд акумулятора.

Це лише один із прикладів того, як пристрої Інтернету речей можуть використовувати сплячий і активний режими для оптимізації енергоспоживання. Ця

технологія особливо важлива для пристроїв, що живляться від батареї, або пристроїв, для яких енергозбереження є ключовим питанням.

Моніторинг мережі. Потоки даних IoT потрібно контролювати на наявність помилок. Для моніторингу мережі слід відстежувати такі показники, як затримка, помилки в пакетах, кількість таймаутів з'єднання, фазове тремтіння цифрового сигналу даних і повторні передачі.

Наприклад, припустимо, що пристрій IoT використовує стільниковий зв'язок. Кожен пристрій має SIM–карту від оператора стільникового зв'язку. Якщо пристрої в певному регіоні мають поганий зв'язок і високі затримки, це може означати, що потрібно змінити постачальника послуг для пристроїв, які постачаються в цей регіон.

Крім того, велика кількість невідповідностей контрольних сум може свідчити про те, що пристрій працює в дуже шумному середовищі. Наприклад, у застосунку для відстеження транспортних засобів, пристрій може бути переміщений подалі від двигуна.

Моніторинг хмари. Щоразу, коли програма передбачає підключення до хмари, моніторинг хмари є критично важливим. Важливими показниками для хмарного моніторингу IoT є:

– завантаження процесора;

– використання пам'яті;

– кількість активних і неактивних з'єднань;

– кількість невдалих запитів;

– кількість невдалих спроб автентифікації.

Наприклад, якщо в мережі 1000 пристроїв, 50000 активних з'єднань і великий відсоток невдалих запитів, це може свідчити про те, що сервер зазнав розподіленої атаки на відмову в обслуговуванні (DDoS–атака). Якщо є 1000 активних з'єднань і 500 неактивних, це може вказувати на те, що пристрої не завершують попередні з'єднання, коли ініціюють нові. Щоб вирішити цю проблему, може знадобитися оновлення прошивки пристрою IoT.

Якщо зі збільшенням кількості пристроїв завантаження процесора і пам'яті починає перетинати комфортний рівень, це може означати, що настав час або перейти на ресурси більшої ємності, або збільшити кількість ресурсів з тією ж ємністю.

Моніторинг користувацького інтерфейсу включає в себе відстеження метрик та використання додатків кінцевими користувачами. Наприклад, можна відслідковувати такі показники, як кількість випадків, коли програма працює некоректно, кількість помилок «додаток не відповідає» (ANR) і проблем з API.

Тепер, коли ми з'ясували, що таке моніторинг Інтернет речей, розглянемо, як це робиться.

Система моніторингу IoT вимагає механізмів передачі даних, які забезпечують швидку доставку сповіщень до необхідного місця призначення. На високому рівні існує три загальні підходи до налаштування механізмів передачі даних для моніторингу IoT:

– безперервний – при безперервному моніторингу дані безперервно надсилаються до вашої хмари в кожному пакеті даних;

– на основі сповіщень – при моніторингу на основі сповіщень до хмари надсилаються лише пріоритетні пакети;

– опитування – під час моніторингу на основі опитування хмара або користувач опитує пристрої для отримання метрик.

Налаштування сповіщень є важливою складовою моніторингу IoT, оскільки забезпечується вчасне інформування користувачів про можливі проблеми. Для цього можна налаштувати сповіщення на електронну пошту та мобільні телефони.

У той час як сповіщення допомагають привернути увагу до негайної проблеми, інформаційні панелі надають більш повну картину і допомагають зорієнтуватися щодо подальших дій.

Інформаційна панель має велике значення для моніторингу IoT, оскільки вона забезпечує доступ до великої кількості даних з різних пристроїв в одному місці. За допомогою цієї панелі користувач може контролювати та взаємодіяти з підключеними пристроями, а також отримувати сповіщення про неполадки та потребу у технічному обслуговуванні.

Користувачі можуть налаштовувати інформаційну панель Інтернету речей так, щоб вона відповідала їхнім потребам, а також передавати дані з неї на свої пристрої. При налаштуванні інформаційної панелі, вони можуть вибирати тільки ту інформацію, яка їх цікавить.

При виборі інструментів для моніторингу компонентів Інтернет речей варто враховувати наступні критерії, які допоможуть знайти найкращі рішення на ринку:

– інструмент повинен надавати необхідні функції для моніторингу і керування компонентами IoT. Це може бути відстеження стану пристроїв, аналіз даних, сповіщення про події тощо;

– інструмент має ефективно працювати з великою кількістю підключених пристроїв і обробляти великі обсяги даних. Важливо, щоб він міг масштабуватися разом з ростом вашої IoT інфраструктури;

– приділіть увагу заходам безпеки, які надає інструмент. Він повинен мати механізми шифрування даних, аутентифікацію користувачів і захист від потенційних загроз кібербезпеки;

– переконайтеся, що інструмент здатний працювати з вашими підключеними пристроями та платформами IoT. Він повинен підтримувати потрібні стандарти комунікації і протоколи;

– виберіть інструмент, який має зрозумілий інтерфейс користувача і простий процес налаштування. Це допоможе зменшити час і зусилля, необхідні для розгортання і використання системи моніторингу;

– перевірте, яким чином надається технічна підтримка від розробників інструменту. Важливо, щоб вони були доступні для вирішення проблем і надання оновлень з усуненнями помилок та новими функціями;

– оцініть вартість інструменту, включаючи витрати на ліцензії, підтримку та оновлення. Порівняйте ціну з його функціональністю та можливостями, щоб знайти оптимальний баланс.

З часом до вже побудованої системи можна додати автоматизовану реакцію на проблеми та їх усунення. Наприклад, коли сервер вийде з ладу, система може

автоматично надіслати SMS–повідомлення або сповістити користувачів про проблему.

Аналогічно, якщо температура критично важливого компонента перевищує певний поріг, система може бути вимкнена на короткий час. Це може захистити компонент від пошкодження.

Важливо розуміти, що моніторинг допомагає виявити симптоми проблеми, але не завжди розкриває її кореневу причину. Наприклад, високе навантаження на хмарний сервер - це лише симптом. Першопричиною може бути DDoS–атака, що вимагає змін у системі безпеки, або несправність пристрою, яка змушує його встановлювати багато з'єднань з хмарою, що потребує оновлення прошивки. Для успішного вирішення проблеми важливо мати комплексне розуміння системи, витратити часу на виявлення та усунення кореневої причини. Також важливо враховувати минулі помилки та навчатися на них, щоб уникнути подібних проблем у майбутньому.

Переваги моніторингу Інтернет речей:

– моніторинг продуктивності пристроїв Інтернет речей допомагає оптимізувати розподіл ресурсів та підвищувати загальну продуктивність мережі. Активне вирішення проблем продуктивності дозволяє користувачам забезпечувати безперебійну роботу і максимізувати ефективність всієї системи;

– аналізуючи дані пристроїв IoT, моніторинг дозволяє передбачати потребу в технічному обслуговуванні пристроїв та запобігати можливим неполадкам. Це дозоляє планувати обслуговування заздалегідь, що допомагає скоротити час простою та ефективно використовувати ресурси;

– управління все більшою кількістю пристроїв IoT стає складним завданням. Моніторинг надає централізований контроль та видимість, дозволяючи розширювати масштаби Інтернету речей без впливу на продуктивність або безпеку мережі;

– отримання сповіщень у режимі реального часу дозволяє вам оперативно реагувати на надзвичайні ситуації та вирішувати проблеми, як тільки вони виникають. Ці сповіщення допомагають скоротити затримки та забезпечити найвищу продуктивність всіх пристроїв;

– моніторинг IoT допомагає скоротити необхідність у ручній праці. Наприклад, проведення перевірок вимагає часу та зусиль, оскільки майстру потрібно їхати до місця розташування обладнання. Але платформи моніторингу Інтернету речей зменшують і навіть усувають цей вид праці, оскільки вся інформація про обладнання буде відображатися на інформаційній панелі.

Моніторинг грає важливу роль у побудові масштабованих та надійних систем. Кожен компонент потребує уважного контролю. Використовуючи концепції, які ми розглянули вище, можна розробити власну стратегію моніторингу IoT.

## 1.5 Вибір технології для побудови мережі IoT

Вибір технології має важливе значення, оскільки він визначає можливості, функціональність та ефективність системи. Перед вибором технології для моніторингу та управління елементами IoT, проводився ретельний аналіз характеристик та вимог, якими повинна відповідати ця система [22].

У багатьох сценаріях потрібні бездротові мережі зв'язку, які не мають високої швидкості передавання, але є надійними, стійкими та простим. Також важливо, щоб обладнання для цих мереж могло працювати тривалий час від автономних джерел живлення, було доступним за ціною та компактним. Один із прикладів такого застосування – «розумний дім».

У результаті, після визначення основних характеристик, вимог і аналізу різних технологій, протоколів та типів мережі, було прийнято рішення вибрати технологію ZigBee. Ця технологія описує стійкі, масштабовані бездротові мережі, які є простими у розгортанні та підтримують найрізноманітніші застосунки.

Мережі ZigBee, на відміну від інших бездротових мереж передачі даних, повністю відповідають перерахованим вище вимогам, а саме:

– забезпечує самовідновлення та гарантовану доставку пакетів завдяки використанню мережевої топології «сітка» (mesh) та спеціальних алгоритмів

маршрутизації. У разі обриву зв'язку між окремими вузлами або відмовою якогось елемента, ця мережа здатна відновити зв'язок і забезпечити надійну передачу пакетів;

– специфікація ZigBee включає криптографічний захист даних, що передаються через бездротові канали, а також гнучку політику безпеки;

– пристрої ZigBee характеризуються низьким споживанням електроенергії, особливо кінцеві пристрої, які можуть працювати до трьох років від звичайної батарейки формату AA. Це досягається завдяки наявності режиму «сну», який дозволяє пристроям економити енергію;

– мережа ZigBee є самоорганізованою, її структура формується автоматично шляхом приєднання (або повторного приєднання) пристроїв до мережі, враховуючи параметри профілю стека конфігуратора. Це дозволяє забезпечити простоту розгортання та легкість масштабування шляхом простого підключення додаткових пристроїв;

– пристрої ZigBee є компактними і мають доступну ціну.

У мережі ZigBee зв'язок відбувається шляхом послідовної передачі пакетів від вузла джерела до вузла адресата. Для цього використовуються кілька альтернативних алгоритмів маршрутизації, які вибираються автоматично.

Стандарт ZigBee дозволяє використовувати канали у різних частотних діапазонах. Найкраща швидкість передачі та найвища стійкість до перешкод досягаються в діапазоні від 2,4 до 2,48 ГГц. У цьому діапазоні передбачено 16 каналів шириною 5 МГц.

У мережах ZigBee, де акцент робився на зменшенні споживання енергії, компактності та доступності, була досягнута відносно низька швидкість передачі даних.

Відстань між робочими станціями у мережі ZigBee може становити десятки метрів всередині приміщень і сотні метрів на відкритому повітрі. Завдяки ретрансляції сигналу мережа може покривати значні території, до кількох тисяч квадратних метрів у приміщенні і кількох гектарів на відкритому просторі.

# РОЗДІЛ 2
# ДОСЛІДЖЕННЯ АПАРАТНОГО ОБЛАДНАННЯ ТА ЗАСОБІВ МОНІТОРИНГУ ДЛЯ УПРАВЛІННЯ ЕЛЕМЕНТАМИ ІНТЕРНЕТУ РЕЧЕЙ

Для забезпечення функціонування системи управління та моніторингу потрібно вибрати таке апаратне забезпечення, яке буде виконувати поставлені задачі та цілі.

Рішення на базі Інтернет речей стають все більш затребуваними саме тому, що дають постачальникам «розумних» рішень можливість отримувати додатковий прибуток, де «розумна» поведінка може дати істотний приріст «корисності», споживчої вартості пристрою або системи. Так, вентилятор, який «сам» вимикається при досягненні потрібної температури, економить власнику електроенергію і тому може коштувати для нього дорожче. А вентилятор, який ще й «бачить», коли в приміщенні є люди, а коли ні – цінний ще більше.

Але як техніка може стати «розумною»? По–перше, за рахунок, власне, своєї конструкції – ця конструкція може бути такою, що поведінка і логіка роботи системи буде виглядати розумною.

По–друге, за рахунок «інтелектуалізації» – оснащення системи пристроями збору інформації, її обробки і прийняття рішень. Такий підхід дозволяє забезпечити досить складне і «розумне» рішення набагато простішими способами, ніж за рахунок створення відповідної конструкції [16].

Нарешті, третій шлях – поведінка системи стає «розумною» внаслідок того, що вона взаємодіє з іншими системами. Так, для економії енергії системі опалення потрібно короткостроковий прогноз погоди. Цей прогноз можна отримати, встановивши відповідні датчики і систему обробки інформації з них, здатну прогнозувати погоду (міні–метеостанцію), а можна просто запросити погоду в Інтернеті. І в тому, і в іншому випадку поведінка системи оточення буде виглядати «розумною».

Важливо, що в останньому прикладі з точки зору замовника система веде себе практично однаково – відповідно, замовник готовий заплатити за цю функціональність одну і ту ж ціну. Однак для постачальника такої системи організація підключення її до Інтернету буде коштувати значно дешевше, ніж розробка інтелектуальної метеостанції.

Завдяки інтелекту і зв'язності устаткування з'являється новий набір функцій. Їх можна розділити на чотири групи:

– моніторинг;

– управління;

– автоматизація;

– автономність.

Кожна функція, важлива і сама по собі, виявляється свого роду сходинкою для наступного рівня. Наприклад, функція моніторингу є основою для управління, оптимізації та автономності техніки. У результаті функція моніторингу дозволяє значно полегшити та оптимізувати користування системою, яка моніториться. Компанія може вибирати такий набір функцій, щоб її продукція була максимально корисною для споживача і таким чином, вона може зміцнювати свою конкурентну позицію на ринку IoT.

Візьмемо, наприклад, автоматичний сад, парник, теплицю, де самостійно здійснюється полив, підтримання потрібної температури, вологості, рівня освітленості та інше. Така система користуватиметься попитом тими, хто не хоче витрачати багато часу на догляд за рослинами, а також може не мати для цього можливості в періоди довгої відсутності персоналу, відрядження, відпустки і т.д.

Однією з проблем, яку вирішує функція моніторингу є усунення занепокоєння щодо того, чи все в порядку з рослинами під час відсутності: чи є вода в системі, що споживає електрику на даний момент та протягом дня, чи може система вентиляції забезпечити потрібну температуру, якщо в приміщенні стало занадто спекотно та ін.

Клієнт напевно заплатить більше, якщо надати йому можливість в будь–який момент знати, які умови в його «розумній теплиці».

Таким чином, продажна вартість цього рішення з функцією віддаленого

моніторингу параметрів може зрости істотно, в той час, як її реалізація для виробниками буде досить простою. В результаті застосування технології інтернету речей дозволить виробнику отримати додатковий прибуток.

Ще вище споживча вартість буде, якщо додати функцію управління, щоб замовник міг дистанційно не тільки отримувати інформацію про умови, а й змінювати їх на свій розсуд відповідно від сезону та набору рослин.

Напевно в теплиці підігрів включається автоматично, якщо температура падає нижче заданої межі, але можливо, не варто його включати, якщо знати, що по прогнозом погоди зовсім скоро очікується підвищення температури? Таким чином, функція автоматизація за рахунок використання додаткової інформації дозволить заощадити гроші на утримання теплиці і отримати урожай з меншими витратами.

Нарешті, засобами Інтернету речей нескладно почати стежити за кількістю витрачених матеріалів, наприклад добрив. Автоматизувати їх замовлення, або контролювати стан елементів, що вимагають заміни або обслуговування: насосів, вентиляторів, що нагрівають, організувавши таким чином самодіагностику і самообслуговування теплиці аж до повної її автономності.

## 2.1 Технічні характеристики мікрокомп'ютеру Raspberry Pi та його альтернативи

Для досягнення поставлених цілей було обрано та куплено одноплатний мікрокомп’ютер Raspberry Pi 4 Model B. Він здатний забезпечувати як і прості практичні завдання, наприклад вивчення комп'ютера і основ роботи з ним, інтернет–серфінгу, а також для програвання відео та прослуховування аудіофайлів, так і більш складні задачі, наприклад бути частиною IoT системи, а саме: сервером керування та моніторингу елементами IoT [5].

Raspberry Pi 4 Model B – це мініатюрний, розміром з кредитну картку, комп'ютер вартістю близько 35 доларів за базову модель і 75 за максимальну. Модель

Raspberry Pi 4 виконано на однокристальній системі Broadcom BCM2711. Кристал включає 4–ядерний 64–бітний процесор CPU Cortex-A72 (ARM v8) з частотою 1,5 ГГц і графічний процесор GPU VideoCore VI з частотою 500 МГц. За даними виробника, система на новій архітектурі стала на 50% швидшою, ніж минулі покоління RPi (рис. 2.1).

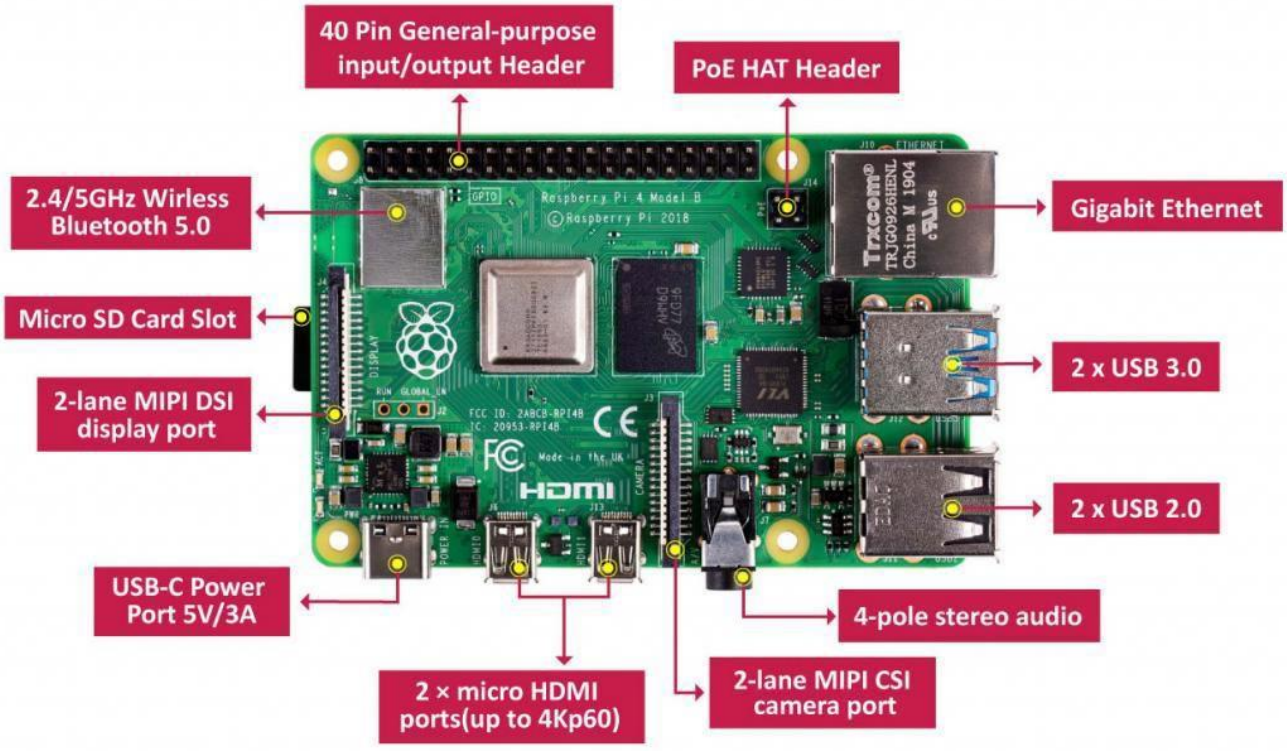


Рисунок 2.1 – Мікрокомп’ютер Raspberry Pi 4 Model B

Дана модель оснащена 4 ГБ оперативної пам'яті SDRAM, яка ділиться між CPU та GPU. Ресурсів мікрокомп’ютера з 4 ГБ пам'яті вистачить на веб–серфінг з безліччю вкладок та роботу з важкими документами в офісних програмах.

Контролер підтримує апаратне декодування H.265/HEVC (до 4Kp60) та H.264 (до 1080p60). Це розвантажує CPU під час роботи з потоковим відео в інтернеті та перегляді «важких» відеофайлів у режимі медіацентру.

Бездротовий модуль підтримує стандарти W–Fi 802.11 b/g/n/ac та протокол Bluetooth 5.0 BLE [5].

Для підключення HDMI–дисплеїв, моніторів та телевізорів передбачено два роз'єми micro–HDMI 2.0. Через них можна одночасно вивести зображення на різні

екрани.

На Raspberry Pi 4 розташовано 40 контактів GPIO (рис. 2.2) для підключення цифрових датчиків, модулів розширення та іншої периферії. Контактна група GPIO повністю сумісна з попередніми версіями Raspberry Pi, тому у користувачів не виникає проблем із перенесенням існуючих проектів.

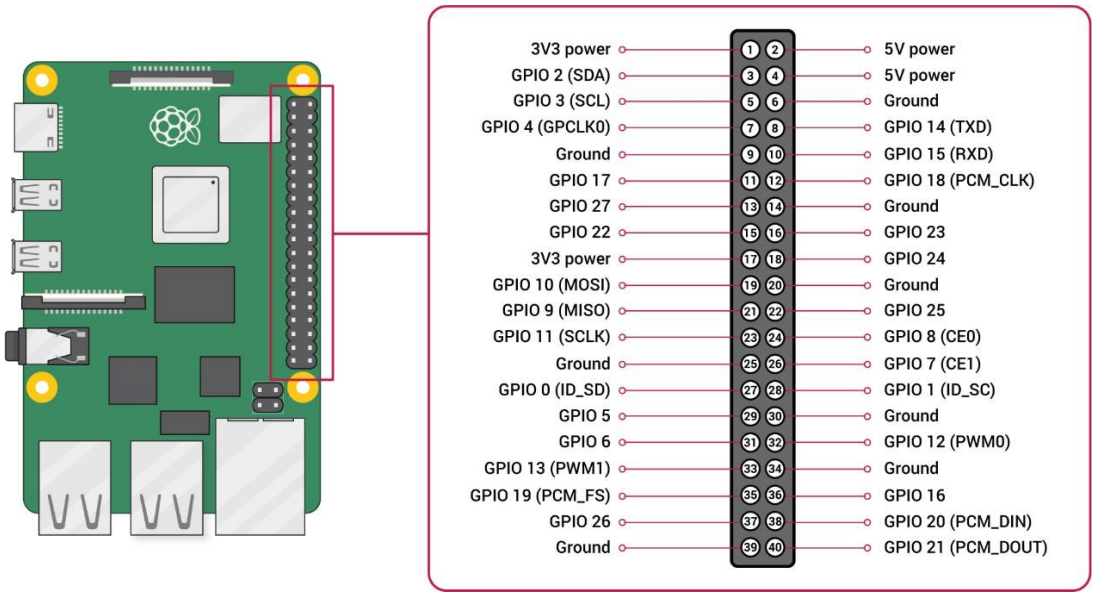


Рисунок 2.2 – Контактна група GPIO в Raspberry Pi 4 Model B

Збоку контролера знаходяться два порти USB 2.0 та два USB 3.0. До них підключається стандартна периферія для комфортної роботи: клавіатура, миша, джойстик та інші пристрої USB.

Порт MIPI CSI призначений для підключення камер Raspberry Pi. Порт MIPI DSI необхідний прямого підключення екранних модулів. Аудіо та відеовихід об'єднані в 4–контактному роз'ємі 3,5 мм, що дозволяє вивести аналоговий звук та композитне відео.

У нижній частині плати знаходиться роз'єм USB Type–C для підключення живлення. Рекомендоване джерело живлення з вихідною напругою 5 вольт та струмом 3 ампери [5].

На ринку існують альтернативні пристрої по відношенню до Raspberry Pi. Такими альтернативами є Orange Pi Prime, Banana Pi M3, Rock64, ASUS Tinker board S, Libre Computer Renegade, Odroid H2. Розглянемо їх більш детально.

Orange Pi Prime відрізняється від Raspberry Pi наявністю лише 2 Гбайт ОЗП і вбудованим в SoC AllWinner H5 відеоприскорювачем Mali–450 GPU, що дозволяє відтворювати лише 2K відео. Серед цікавих особливостей варто відзначити наявність ІК–приймача; платою можна керувати з пульта дистанційного керування або з деяких моделей стільникових телефонів з вбудованим ІЧ–світлодіодом. З нестандартного обладнання є також вбудований мікрофон і відеоінтерфейс CSI, що підтримує відеопотік до 1080p на швидкості 30 fps (рис. 2.3).

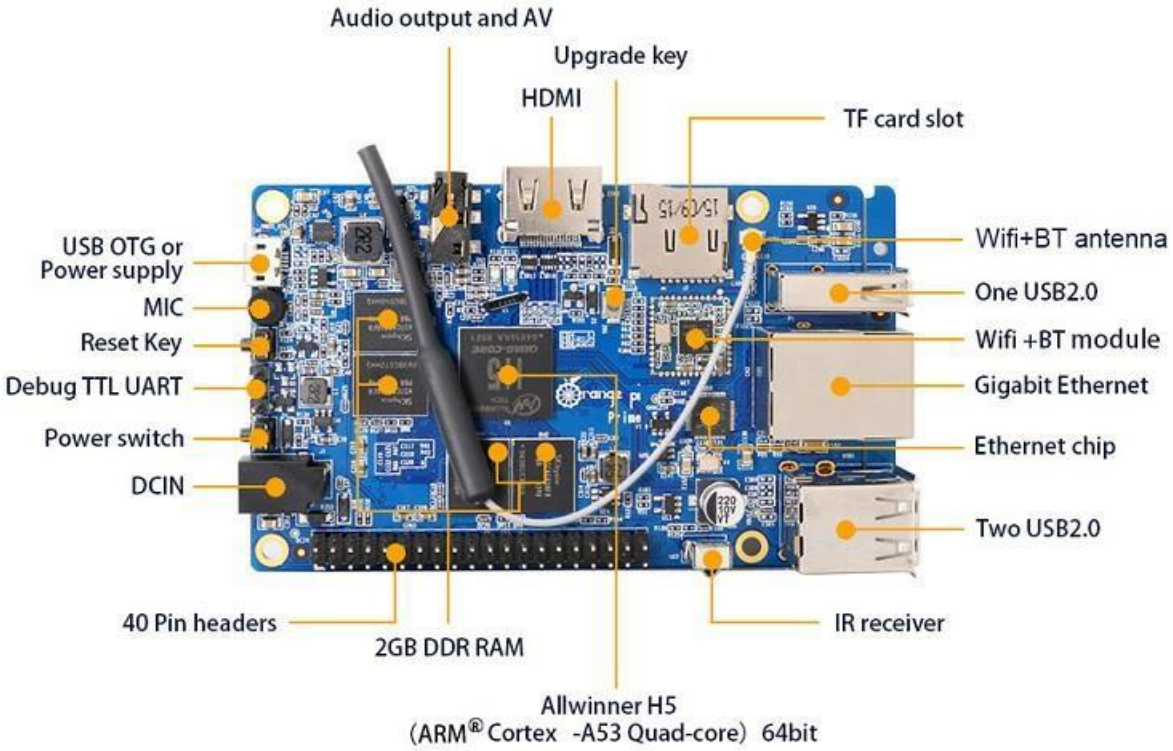


Рисунок 2.3 – Мікрокомп’ютер Orange Pi Prime

На платі розміром 98 × 60 мм знайшлося місце для роз'єму для карт пам'яті (до 32 Гбайт), Wi–Fi 802.11 b/g/n, Bluetooth 4.0, гігабітного Ethernet, чотирьох USB (три USB 2.0 Host і один USB 2.0 OTG) і GPIO–гребінки. Є навіть окремо виведений UART з TTL рівнями, що дозволяє в терміналі спостерігати за деталями завантаження ОС.

З аудіо обладнання, крім згаданого вище мікрофона, є ще лінійний вихід і аудіовихід в HDMI. Відеоприскорювач підтримує OpenGL ES 2.0 і OpenVG 1.1. Серед підтримуваних ОС присутні це Ubuntu, Debian і Android 5.1.

Флагман Banana Pi M3 побудований на базі восьмиядерного SoC Allwinner A83T (процесори ARM Cortex–A7, графічний процесор PowerVR SGX544MP1),

розганяється до 1.8 ГГц і працює в оточенні 2 Гбайт ОЗП і 8 Гбайт флеш-пам'яті. Крім гігабітного Ethernet, двох USB, Wi–Fi 802.11 b / g / n, Bluetooth 4.0 і HDMI, на платі присутній SATA роз’єм (рис. 2.4).

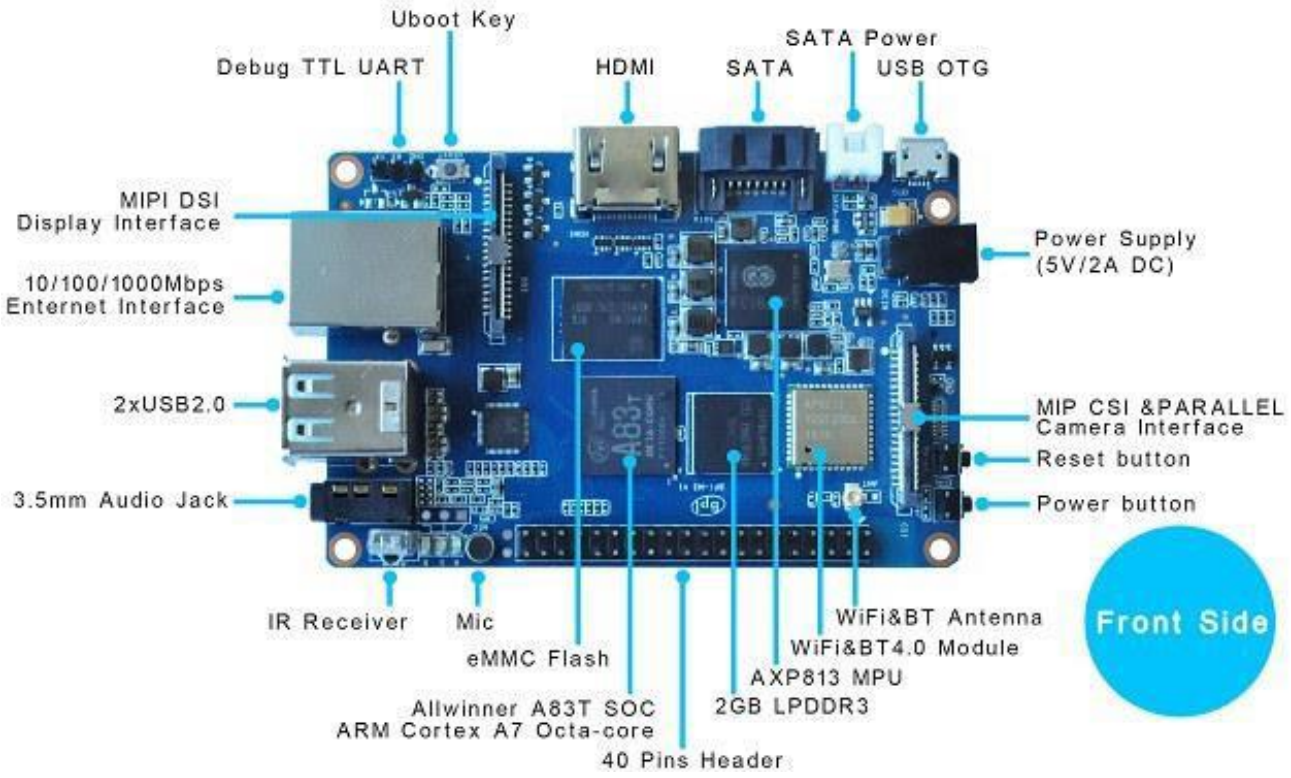


Рисунок 2.4 – Мікрокомп’ютер Banana Pi M3

Так само, як і у Orange Pi Prime, у M3 є ІК–приймач, відеоінтерфейс CSI, оцінний UART, мікрофон, лінійний вихід і аудіовихід в HDMI. На відміну від Orange, у Banana є інтерфейс дисплея MIPI DSI, об'єднаний з I2C для сенсорного екрану. Також є і GPIO–гребінка.

Одноплатний комп'ютер Rock64 комплектується вже 4 Гбайт ОЗП, що обслуговують 64–х бітний ARM Cortex A53, відеопідсистема здатна впоратись з потоком 4K на частоті 60 fps. Графічна підсистема ARM Mali 450MP2 відповідає OpenGL ES 2.0, OpenVG1.1 (рис. 2.5). У Rock64 детальна документація і живе, активне ком'юніті, так що, з урахуванням непоганих апаратних специфікацій і ціни, цей міні ПК – непоганий претендент на заміну Raspberry Pi 4 в деяких проектах [6].


Рисунок 2.5 – Мікрокомп’ютер Rock64

Розробники Rock64 додали 64 контакти GPIO, вивівши на них навіть сигнали Ethernet.

ASUS Tinker board S може похвалитися більш високою продуктивністю, ніж інші одноплатні комп'ютери, за рахунок використання потужного чотириядерного процесора Rockchip RK3288 з архітектурою ARM, доповненого двоканальною оперативною пам'яттю LPDDR3 обсягом 2 ГБ і вбудованим eMMC–накопичувачем на 16 ГБ. Крім того, у ньому застосовується інтерфейс SD 3.0, що забезпечує прискорений обмін даними з картою пам'яті microSD, яка може використовуватися для зберігання операційної системи, додатків і файлів користувача (рис. 2.6).

Графічне ядро Mali T760 MP4 з архітектурою ARM, яким оснащений комп'ютер Tinker Board S, є сумісним із різними програмними інтерфейсами й прекрасно підходить для цілого спектра мультимедійних додатків, починаючи від ігор і домашнього кінотеатру й закінчуючи системами машинного зору та розпізнавання жестів [6].

Комп'ютер Tinker Board S має підтримку функції HDMI–CEC, що дозволяє використовувати для керування ним пульт дистанційного керування сучасного телевізора. Також він підтримує апаратне прискорення під час відтворення відео у форматах H.264 і H.265 (в тому числі з роздільною здатністю HD та Ultra HD).

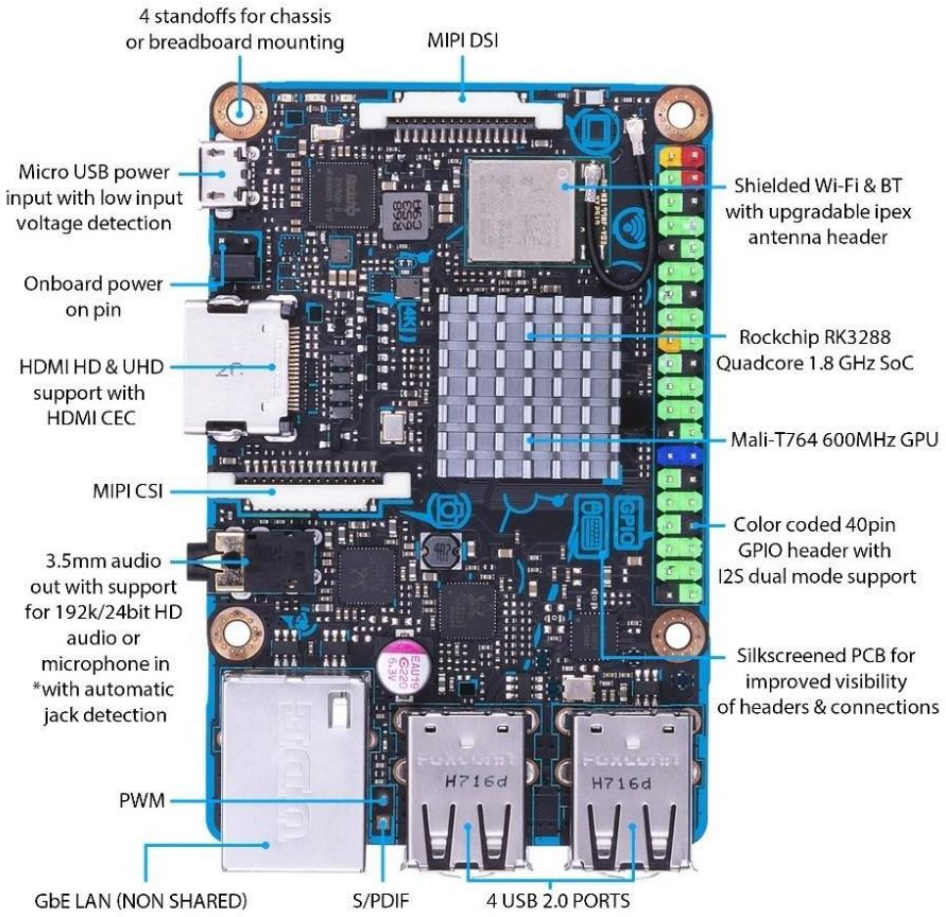


Рисунок 2.6 – Мікрокомп’ютер ASUS Tinker board S

Одноплатний комп'ютер Tinker Board S має стандартні для свого класу інтерфейси, зокрема 40–контактний GPIO і I2S з режимами Master і Slave для більшої сумісності. Використовуючи програмний інтерфейс GPIO, можна сполучити Tinker Board S з різними пристроями введення, як-от кнопки, перемикачі, сенсори, світлодіоди. Також пропонується два інтерфейси MIPI: DSI MIPI для підключення дисплеїв і сенсорних панелей і CSI MIPI для камер.

Для підключення до дротової мережі Tinker Board S пропонує інтерфейс Gigabit Ethernet із гарантовано високою пропускною спроможністю за рахунок шини, ресурси якої не розподіляються з іншими пристроями. Крім того, доступні бездротові інтерфейси Wi–Fi і Bluetooth, причому їх модуль екранований для захисту від електромагнітних завад, а роз'єм і–PEX дозволяє легко міняти антену на найбільш відповідну для поточного проекту.

Libre Computer Renegade (рис. 2.7) конструктивно розроблений настільки

схожим на Raspberry, наскільки це тільки можливо; наприклад, є можливість розмістити ROC–Rk3328–CC прямо в корпусі Raspberry Pi.

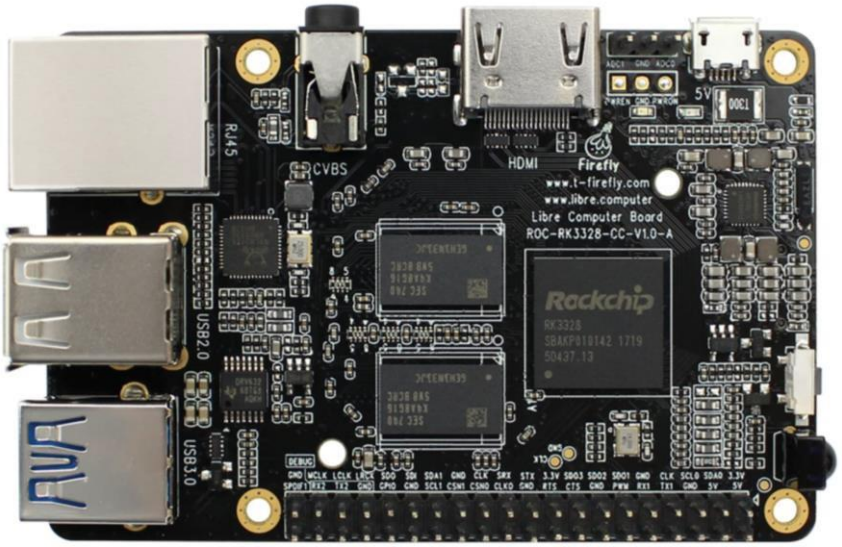


Рисунок 2.7 – Мікрокомп’ютер Libre Computer Renegade

ROC RK–3328 побудована на основі чотирьох ядерного 64–х бітного процесора ARM Cortex–A53 з робочою частотою до 1.5 ГГц. SoC така ж, як і в Rock64, так що тут ви теж маєте той же GPU Mali 450MP2 з робочою частотою 500 МГц. З операційних систем на даний момент доступні Ubuntu 18.04, Debian 9, OpenMediaVault 4, Station OS і Android 7.1.

Плату ODROID-H2 було представлено як перший x86 SBC від Hardkernel (рис. 2.8).

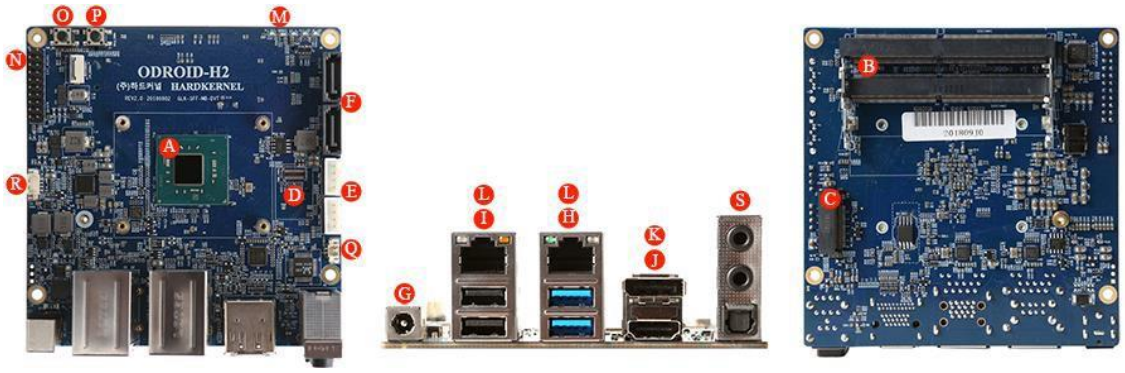


Рисунок 2.8 – Мікрокомп’ютер Odroid H2

На платі встановлений процесор Intel Celeron J4105 Gemini Lake, два слоти SO–DIMM для пам'яті, два порти SATA та слот M.2 NVMe для зберігання, два порти

Gigabit Ethernet, плата поєднує порти USB 3.0 та 2.0, а також HDMI та відеовиходи DisplayPort.

У таблиці 2.1 наведемо порівняльну характеристику розглянутих пристроїв та міні–ПК Raspberry Pi.

Таблиця 2.1 – Порівняльна характеристика Raspberry Pi та аналогів

| Модель | SoC | Процесор | Графіка | Ядра | Частота, ГГц | Розмір, мм | Ціна, $ |
|---|---|---|---|---|---|---|---|
| Raspberry Pi 4B | Broadcom BCM2711 | ARM Cortex–A72 | Broadcom VideoCore VI | 4 | 1.5 | 85.6 × 56.5 | 35 |
| Orange Pi Prime | AllWinner H5 | ARM Cortex–A53 | Mali–450 | 4 | 1.4 | 98 × 60 | 68 |
| Banana Pi M3 | Allwinner A83T | ARM Cortex–A7 | PowerVR 544MP1 | 8 | 1.8 | 92 × 60 | 68 |
| Rock64 | Rockchip RK3328 | ARM Cortex A53 | Mali 450MP2 | 4 | 1.5 | 56 × 85 | 45 |
| Asus Tinker board S | Rockchip RK3288 | ARM Cortex–A17 | Mali T760 MP4 | 4 | 1.8 | 54 × 86 | 92 |
| Libre Computer Renegade | Rockchip RK–3328 | ARM Cortex–A53 | Mali 450MP2 | 4 | 1.5 | 85 × 56 | 80 |
| Odroid H2 | – | Intel Celeron J4105 | Intel UHD Graphics 600 | 4 | 2.3 | 110 × 110 | 111 |

За даними таблиці можна зробити наступні висновки: Raspberry Pi 4B вирізняється найнижчою ціною серед усіх моделей, всього 35 доларів. Banana Pi M3 має найбільшу кількість ядер – 8, що може бути важливим для паралельної обробки даних. Odroid H2 використовує процесор Intel Celeron J4105 та графічний процесор Intel UHD Graphics 600, що дає йому перевагу в обробці графіки. Розміри моделей варіюються від 54 × 86 мм до 110 × 110 мм, що також варто враховувати при побудові

системи. Багато моделей використовують ARM–процесори з різними конфігураціями ядер і графічними процесорами, що дає широкий вибір залежно від потреб користувача. Загалом, вибір конкретної моделі буде залежати від потреб, бюджету, продуктивності та розміру проекту (рис. 2.9).

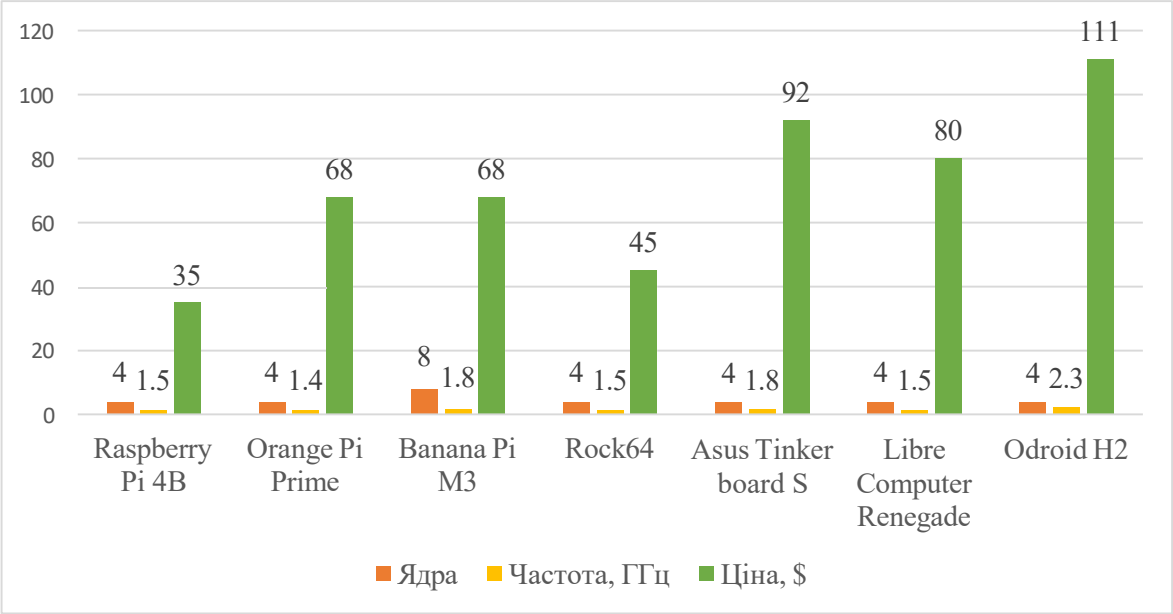


Рисунок 2.9 – Порівняльна діаграма за ціною, кількістю ядер та частотою

У таблиці 2.2 наведено порівняння підтримуваних інтерфейсів та оперативної пам’яті.

Таблиця 2.2 – Порівняння підтримуваних інтерфейсів Raspberry Pi та аналогів

| Модель | ОЗП | Флеш | GPIO | USB | Ethernet | Wi–Fi | Bluetooth |
|---|---|---|---|---|---|---|---|
| Raspberry Pi 4B | 4 Гб LPDDR4 3200 | Слот MicroSDHC | 40 | 4 | 1000 Мбит/с | 802.11b/g/n/ac 2.4/5 ГГц | 5 BLE |
| Orange Pi Prime | 2 Гб LPDDR3 | Слот MicroSDHC | 40 | 4 (3 × 2.0, 1 × OTG) | 1000 Мбит/с | 802.11 b/g/n | 4 |
| Banana Pi M3 | 2 Гб LPDDR3 | 8 Гб eMMC | 64 | 3 (2 × 2.0, 1 × OTG) | 1000 Мбіт/с | 802.11 b/g/n | 4 |
| Rock64 | 4 Гб LPDDR3 | 128 Мбайт | 64 | 3 (3.0, 2.0, OTG) | 1000 Мбіт/с | 802.11 b/g/n | 4 |
| Asus Tinker board S | 2 Гб LPDDR3 | 16 Гб eMMC | 40 | 4 × USB 2.0 | 1000 Мбіт/с | 802.11 b/g/n | 4 |

Продовження таблиці 2.2

| Модель | ОЗП | Флеш | GPIO | USB | Ethernet | Wi–Fi | Bluetooth |
|---|---|---|---|---|---|---|---|
| Libre Computer Renegade | 4 Гб DDR4 | Слот MicroSDHC | 40 | 3 (1 × 3.0, 1 × 2.0) | 1000 Мбіт/с | – | – |
| OdroidH2 | 2 слота DDR4 SO–DIMM | 128 Мбайт (BIOS), слотеMMC | – | 4 (2 × 3.0, 2 × 2.0) | 2 × 1000 Мбіт/с | – | – |

Отже, згідно даного порівняння та у ході аналізу було визначено, що для даного дослідження оптимальним по ціні та своїм технічним показникам буде міні–ПК Raspberry Pi 4 Model B.

## 2.2 Підбір необхідного обладнання, елементів IoT

Підбір необхідного обладнання є важливим кроком, оскільки знадобиться не тільки сам центральний пристрій, такий як раніше обраний міні–ПК Raspberry Pi, а й периферійні.

Так як міні–ПК Raspberry Pi не має вбудованої внутрішньої пам’яті, то було обрано карту пам’яті Samsung формату microSDHC 32GB EVO Plus (рис. 2.10).

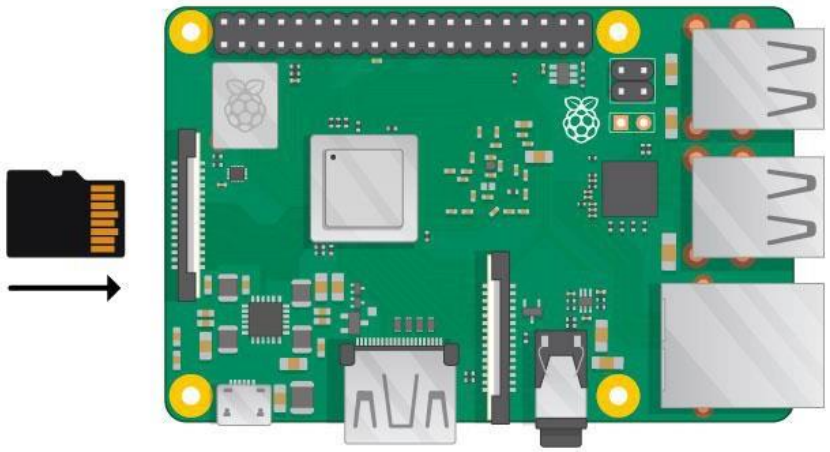

Рисунок 2.10 – Підключення карти пам’яті до міні–ПК Raspberry Pi

Вона має у своєму складі 32 Гб постійної пам’яті, чого вистачить для встановлення системи та подальших маніпуляцій. Щоб встановити карту потрібно вставити її у слот MicroSD.

Так як міні–ПК Raspberry Pi не оснащений модулем ZigBee, то було обрано адаптер SONOFF Zigbee 3.0 USB Dongle Plus.

SONOFF Zigbee 3.0 USB Dongle Plus – це універсальний USB координатор–шлюз Zigbee, призначений для підключення безпосередньо до комп'ютера або мікрокомп'ютера Raspberry Pi і може використовуватися як координатор або маршрутизатор для мережі Zigbee через ZHA або Zigbee2MQTT на платформах автоматизації з відкритим вихідним кодом, таких як Home Assistant.

Він забезпечує роботу з широким спектром пристроїв, що підтримуються: цей модуль може використовуватися як шлюз Zigbee 3.0 на платформах автоматизації з відкритим вихідним кодом для управління різними суб–пристроями від різних виробників або прошивкою флеш-маршрутизатора для розширення діапазону мережі. Він має широкий спектр пристроїв, що підтримуються, таких як BASICZBR3, S31 Lite zb, SNZB01, SNZB02, SNZB03, SNZB04, ZBMINI, S26R2ZB і подібні.

Щоб встановити цей адаптер потрібно всунути у слот USB Type–A (рис. 2.11).

Рисунок 2.11 – Підключення ZigBee адаптеру до міні–ПК Raspberry Pi

Під час тестів зібраної системи я помітив високу температуру процесора, коли температура досягла 80°C, відбувається перегрів і починається тротлінг. Процесор

починає пропускати такти, щоб запобігти подальшому підвищенню температури. Для стабільної роботи системи потрібно було прибрати перегрів та тротлінг.

Для цього було обрано вентилятор – ICE Tower. Це кулер із радіатором, дизайн якого розробила компанія 52Pi (рис. 2.12).

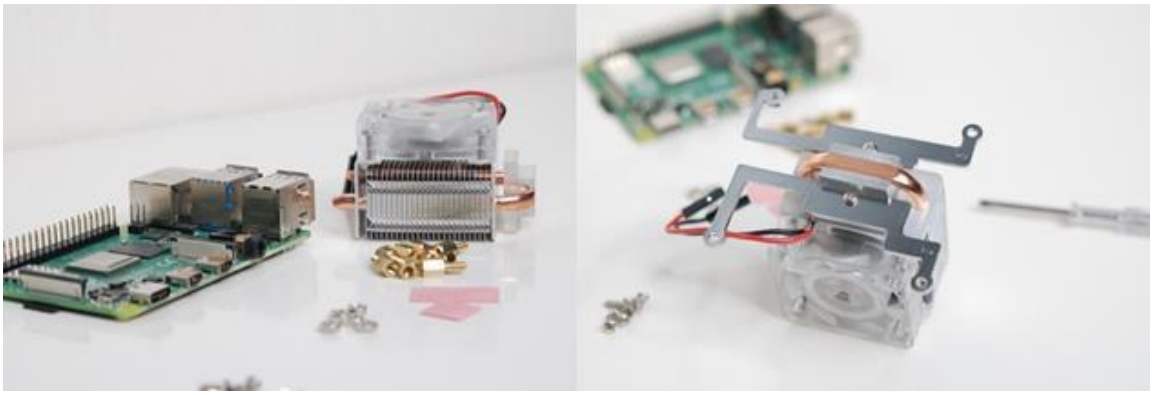

Рисунок 2.12 – Встановлення кулеру із радіатором на міні–ПК Raspberry Pi

Встановлення та складання ICE Tower досить просте. У комплекті є все необхідне та покрокова інструкція з фотографіями.

Після всіх проведених тестів, застосувавши утиліту vcgencmd та sysbench, було отримано такі результати (табл. 2.3):

Таблиця 2.3 – Температура ЦП з та без додаткового охолодження

| Навантаження на ЦП, % | Температура без охолодження, ℃ | Перегрів (без охолодження) | Температура з охолодженням, ℃ | Перегрів (без охолодження) |
|---|---|---|---|---|
| 3 | 48 | Ні | 28 | Ні |
| 15 | 53 | Ні | 29 | Ні |
| 32 | 61 | Ні | 30 | Ні |
| 53 | 65 | Ні | 32 | Ні |
| 55 | 73 | Ні | 36 | Ні |
| 67 | 81 | Так | 37 | Ні |
| 89 | 80 | Так | 38 | Ні |
| 100 | 80 | Так | 38 | Ні |

Для наочного порівняння показників температури з та без встановленого кулера побудуємо діаграму, на якій відображатимуться значення температури для кожного навантаження. Це дозволить візуально порівняти ефективність охолодження

та вплив кулера на зниження температури ЦП (рис. 2.13).

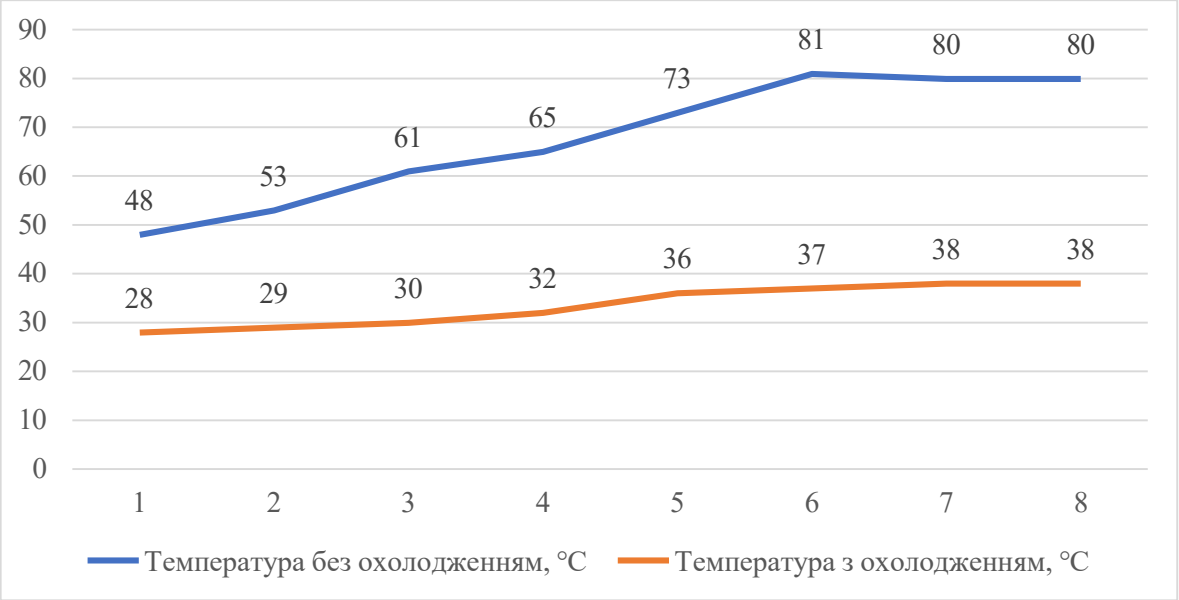


Рисунок 2.13 – Порівняльна діаграма температур

Різниця дуже помітна. Додаткове охолодження не тільки знижує температуру процесору, а й збільшує термін служби пристрою, тобто показник MTBF (Mean time between failures) зростає в рази.

Але для стабільної роботи 24x7 потрібне також і якісне живлення. Для цього було обрано ДБЖ MARSRIVA KP3 (рис. 2.14).

Рисунок 2.14 – ДБЖ Marsriva Smart Mini UPS KP3

Джерело безперебійного живлення Marsriva Smart Mini UPS KP3 призначене для забезпечення безперервного живлення модемів, роутерів, маршрутизаторів, міні–ПК, акваріумного обладнання, камер відеоспостереження та інших електроприладів.

Перемикання ДБЖ Marsriva Smart Mini UPS KP3 на автономну роботу здійснюється миттєво. Надійний акумулятор ємністю 10000 мА*г може заживити відразу декілька пристроїв з сумарною потужністю до 18 Вт через роз'єми DC та USB.

Джерело безперебійного живлення оснащене інтелектуальною схемою із захистом від надмірного заряджання та розряджання, короткого замикання та можливістю вибору вихідної напруги постійного струму 5В/9В/12В.

Для під'єднання мікрокомп'ютеру Raspberry Pi до ДБЖ потрібно використати кабель живлення USB Type–A – Type–C.

Після всіх маніпуляцій маємо готову систему для встановлення ПЗ, його налаштування та побудови ZigBee мережі (рис. 2.15).


Рисунок 2.15 – Готова система керування та моніторингу елементів IoT

Щодо елементів IoT, то в якості датчику температури та вологості було обрано Tuya IH–K009 – це мініатюрний бездротовий пристрій для моніторингу параметрів температури, вологості [31].

Завдяки показанням датчика в системі «Розумного Будинку» можна будувати різноманітні автоматизації. Наприклад, при досягненні температури повітря нижче певного значення, увімкнути пристрій обігріву. Або не вимикати витяжний вентилятор (за допомогою реле), поки вологість у ванній не опуститься до заданого значення.

Датчик температури працює за бездротовим протоколом зв'язку ZigBee, що дозволяє мінімізувати розміри корпусу пристрою, а також підвищити енергоефективність його роботи. Елементом живлення служить мініатюрна батарея CR2450, якої вистачає на період більше року. В експлуатації датчик показує себе як стабільний і надійний пристрій (рис. 2.16).

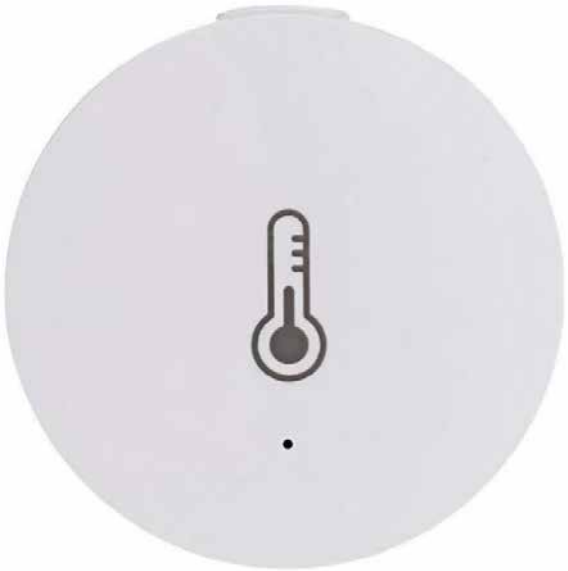

Рисунок 2.16 – Датчик температури та вологості ZigBee Tuya IH–K009

Датчик відкриття Tuya TS0203 призначений для забезпечення безпеки приміщень. Робота пристрою заснована на подачі сигналу через Інтернет на підключений пристрій при розмиканні магнітного контакту. Це дає користувачеві впевненість у стовідсотковому спрацьовуванні та своєчасному отриманні інформації [32].

Цей розумний датчик дозволяє не тільки значно підвищити безпеку будинку або офісу, за рахунок миттєвого відправлення повідомлення про відкриття дверей або вікон. Його також можна використовувати для реалізації різних сценаріїв, які можуть

значно підвищити рівень комфорту життя. Наприклад, при відкритті дверей вмикатиметься світло у передпокої, при зачиненні вікна спрацює система кондиціювання в будинку або навпаки, коли йдете, освітлення вимкнеться, відкриєте вікно, кондиціонер відключиться і так далі (рис. 2.17).


Рисунок 2.17 – Датчик відкриття ZigBee Tuya TS0203

Датчик руху моделі Tuya 809WZT призначений для виявлення руху людей або тварин. Він має широкий кут виявлення 120 градусів і може спостерігати на відстані до 5 метрів. Цей датчик здатен забезпечувати моніторинг у реальному часі як вдень, так і вночі. Його основна функція полягає у підвищенні рівня безпеки, наприклад, в тих випадках, коли вдома немає жодної людини. Цей датчик є частиною системи безпеки, яка дозволяє користувачам відстежувати події, пов'язані з рухом, і отримувати сповіщення навіть на віддаленій від дому відстані. Завдяки надійному функціонуванню і зручному інтерфейсу, він є незамінним рішенням для підвищення рівня безпеки та контролю над приміщенням (рис. 2.18).

Рисунок 2.18 – Датчик руху ZigBee Tuya 809WZT

Розумний датчик води/затоплення ZigBee Tuya TS0207 призначений для виявлення затоплення в приміщеннях.

Пристрій можна розмістити у ванній кімнаті, під раковиною або ванною, при відкритих вікнах, поруч з трубами, під дахом – скрізь, де є ймовірність затоплення або витоку води [34].

При появі води, контакти датчика замикаються, таким чином надсилаючи сигнал (тривогу) про затоплення на центральний шлюз (рис. 2.19).

Рисунок 2.19 – Датчик протікання води ZigBee Tuya TS0207

За допомогою електроприводів ZigBee Tuya можна налаштувати сценарій, щоб у разі спрацьовування датчику протікання води електроприводи перекривали воду.

У таблиці 2.4 наведений повний перелік пристроїв, їх моделі та необхідна кількість.

Таблиця 2.4 – Перелік використаного обладнання

| Тип пристрою | Модель | Кількість |
|---|---|---|
| Міні–ПК | Raspberry Pi 4 Model B | 1 |
| ДБЖ | Marsriva Smart Mini UPS КР3 | 1 |
| Кабель живлення | Ugreen USB Type-A – Type C | 1 |
| Карта пам’яті | Samsung microSDHC 32GB EVO Plus UHS-I Class 10 | 1 |
| ZigBee адаптер | SONOFF ZBDongle–E | 1 |
| Кулер охолодження | ICE Tower 52Pi | 1 |
| Датчик температури та вологості | Tuya IH–K009 | 1 |
| Датчик відкриття | Tuya TS0203 | 1 |
| Датчик руху | Tuya 809WZT | 1 |
| Датчик протікання води | Tuya TS0207 | 1 |

Для забезпечення функціонування системи управління та моніторингу потрібно вибрати таке апаратне забезпечення, яке забезпечуватиме виконання поставлених задач та цілей. Тому було запропоновано міні–ПК Pasbperry Pi, як серце системи. Він здатний забезпечувати як і прості практичні завдання, наприклад вивчення комп'ютера і основ роботи з ним, програвання відео та прослуховування аудіо, так і більш складні задачі, наприклад бути частиною IoT системи. Також варто зазначити, що міні–ПК Pasbperry Pi є економним варіантом для побудови системи. А саме такі цілі і ставились у даному дослідженні.

Були наведені технічні характеристики і можливості міні–ПК Pasbperry Pi та додаткового обладнання, необхідного для налаштування системи управління та моніторингу елементами IoT.

## 2.3 Аналіз та вибір програмного забезпечення для системи моніторингу та управління елементами IoT

Багато пристроїв Інтернет речей мають обмежені обчислювальні можливості. Програмне забезпечення (ПЗ) IoT полегшує переміщення та обробку даних, зібраних цими пристроями, щоб вони могли віддалено виконувати свої дії або функції. Воно також має важливу роль у створенні інтерфейсу для користувачів, що дозволяє автоматизувати та керувати розумними пристроями, а також забезпечує доступ до даних для отримання актуальної інформації. Також, програми, призначені для роботи в мережі IoT, допомагають ефективно збирати дані з датчиків.

Програмне забезпечення для моніторингу та управління пристроями Інтернету речей має безліч застосувань і переваг.

Головна перевага полягає у зручному процесі реєстрації та автентифікації пристроїв. Крім цього, платформа дозволяє автоматизувати процес реєстрації та налаштування, що дозволяє скоротити час і зусилля.

За допомогою програмного забезпечення можна зручно контролювати пристрої в режимі реального часу. Воно надає багато параметрів, таких як продуктивність, підключення, передача даних, час безвідмовної роботи та використання ресурсів.

У мережі Інтернет речей можуть бути пристрої з різними конфігураціями. Проте, це не є проблемою. Програмне забезпечення IoT надає можливість керувати конфігураціями пристроїв, оновленнями прошивки та політикою безпеки з одного інтерфейсу.

Програмне забезпечення для моніторингу та управління пристроями IoT може надавати різноманітні функції, що включає шифрування, контроль доступу та автентифікацію, що забезпечує безпеку даних.

У пристроїв Інтернет речей є важлива перевага порівняно з традиційними пристроями – вони можуть бути керовані та доступні для управління віддалено. Завдяки спеціальному програмному забезпеченню, можна виконувати оновлення,

встановлювати нові версії прошивки та підтримувати систему безпеки пристрою. Це допомагає усунути можливі вразливості та розширити функціональні можливості пристрою.

При виборі програмного забезпечення для управління та моніторингу пристроями Інтернет речей, важливо враховувати низку факторів. Наведемо основні з них, які допоможуть обрати найкраще ПЗ для надійної і ефективної системи.

При виборі відповідного програмного забезпечення важливо переконатися, що воно дозволяє змінювати інформаційну панель. Це дозволить налаштувати її згідно з вашими потребами та переглядати важливі аспекти. Деякі програми також дають можливість мати кілька персональних інформаційних панелей.

Мережа Інтернет речей може бути складною. Тому важливо, щоб програмне забезпечення мало різноманітні функції моніторингу та звітності, такі як аналітика, оповіщення в реальному часі, створення індивідуальних звітів, тощо.

Програма має забезпечувати зручний доступ до периферійних пристроїв, таких як датчики температури, руху, відкривання дверей та інші. Це спрощує моніторинг та керування.

Дані з датчиків, якщо потраплять у руки кіберзлочинців, можуть призвести до серйозних проблем. Тому при виборі програмного забезпечення варто звернути увагу на функції безпеки, такі як багатофакторна автентифікація, контроль доступу, шифрування даних та інші.

Крім функціональності, важливо також переконатися, що постачальник програмного забезпечення працює на ринку протягом тривалого часу.

Залежно від потреб користувача, у майбутньому може знадобитися масштабування системи. Тому важливо обрати платформу, яка здатна масштабуватися і працювати з додатковими пристроями та великими обсягами даних.

Обране програмне забезпечення повинно мати можливість підтримувати широкий спектр пристроїв IoT від різних брендів. Крім того, варто переконатися, що обрана платформа сумісна з іншими системами.

На сьогоднішній день існує велика кількість програмних застосунків для моніторингу та управління елементами Інтернету речей. Кількість таких застосунків

постійно зростає, оскільки IoT стає все більш поширеним і важливим напрямком. Нові розробки з'являються на ринку, а існуюче програмне забезпечення оновлюються та розширює свої можливості.

Вибір підходящого програмного застосунку для системи моніторингу та управління елементами Інтернет речей є складною та відповідальною задачею. Від цього вибору залежить ефективність всієї мережі IoT, тому необхідно враховувати всі вимоги до розроблювальної системи і аспекти розглянуті вище, щоб підібрати найкраще рішення.

Після проведення ринкового аналізу було виявлено кілька програмних застосунків, які відповідають нашим вимогам для розробки системи моніторингу та управління елементами Інтернет речей. Кожен з цих застосунків був детально розглянутий з метою визначення найкращого варіанту для нашої системи.

Home Assistant – це платформа домашньої автоматизації з відкритим вихідним кодом, яка легко встановлюється і налаштовується на різних пристроях, включаючи Raspberry Pi, і написана на мові Python.

Home Assistant має ряд значних переваг, які роблять його привабливим вибором для реалізації проектів домашньої автоматизації. По–перше, він базується на відкритому вихідному коді, що дозволяє повністю налаштувати його під свої потреби без обмежень. Це дозволяє вносити власні зміни, розроблювати нові функції або використовувати розширення, щоб створити ідеальну систему. Крім того, Home Assistant є безкоштовним у використанні. Також варто відзначити наявність великої та активної спільноти користувачів і розробників, що дозволяє отримати необхідну допомогу та підтримку у разі потреби. Проте, варто враховувати кілька недоліків. По–перше, Home Assistant вимагає самостійного налаштування та підтримки, оскільки є самодостатньою платформою. По–друге, використання багатьох інтеграцій та автоматизацій може спричинити значне навантаження на ресурси системи, що може призвести до сповільнення її роботи.

Однією зі функцій, які пропонує Home Assistant, є гнучка в налаштуванні інформаційна панель, до якої можна отримати доступ з будь–якого пристрою за допомогою веб–браузера або додатку. Є можливість додавати і видаляти віджети, а

також змінювати їх розміщення. Крім того, Home Assistant надає можливість автоматизації, що дозволяє налаштовувати правила, які виконують певні дії на основі заданих умов. Наприклад, можна налаштувати освітлення так, щоб воно вмикалося при поверненні власника додому, або налаштувати термостат так, щоб він регулював температуру, враховуючи прогноз погоди.

Отже, якщо вам потрібна платформа для домашньої автоматизації, яка доволі легко налаштовується і має відкритий вихідний код, Home Assistant може бути надійним вибором.

OpenHAB – це також платформа для домашньої автоматизації з відкритим вихідним кодом, яка дозволяє користувачам інтегрувати та керувати пристроями від різних виробників. Вона надає уніфікований інтерфейс для управління всіма компонентами системи і розширені можливості для написання сценаріїв та програмування, що робить її привабливим вибором для користувачів з поглибленими технічними знаннями. Однак OpenHAB може бути менш зручним для користувачів без технічних навичок і вимагати встановлення додаткового програмного забезпечення та конфігурацій. Вона також може бути встановлена і налаштована на доступний за ціною пристрій, наприклад, Raspberry Pi. OpenHAB підтримує багато популярних платформ, включаючи Linux, Windows і macOS, і розроблена на мові програмування Java. Також наявна активна спільнота користувачів, які діляться налаштуваннями та надають підтримку [28].

OpenHAB пропонує різноманітні варіанти інтерфейсу, проте водночас він має дублювання та надмірну кількість схожих варіантів, що може заплутати користувачів. Зовнішній вигляд та елементи інтерфейсу також не є найсучаснішими у більшості варіантів.

І OpenHAB, і Home Assistant пропонують офіційні мобільні додатки для користувачів Android та iOS. Додатки забезпечують зручний доступ до керування всієї системи з кишені. На обох платформах доступні додаткові датчики, такі як виявлення присутності, рівень заряду батареї та геолокація. Найкращою функцією обох додатків є можливість отримувати пуш–сповіщення від систем автоматизації.

На підставі проведеного аналізу платформи OpenHAB можна стверджувати, що вона надає широкі можливості автоматизації та інтеграції з різноманітними пристроями і сервісам для досвідчених користувачів завдяки своїй потужній структурі Eclipse SmartHome. Варто відзначити, що дизайн OpenHAB може вважатися трохи застарілим порівняно з іншими сучасними платформами.

Domoticz – це ще одна платформа для домашньої автоматизації, яка відзначається своєю легкістю порівняно з OpenHAB і Home Assistant, але при цьому забезпечує достатню кількість функцій. Конфігурація в основному здійснюється через веб–інтерфейс, і для розширення функціональності можна використовувати плагіни [29].

Domoticz має значно меншу спільноту користувачів порівняно з Home Assistant і OpenHab. Крім того, варто зазначити, що документація може вважатися застарілою, і не завжди легко знайти відповіді на проблеми, з якими можна зіштовхнися. Також вона є дуже стабільною платформою, яка добре справляється з основними завданнями. Однак, на сьогоднішній день має обмежену підтримку пристроїв і конфігурацій.

Domoticz, безумовно, відстає, коли йдеться про підтримку останніх пристроїв, особливо тих, що належать до пропрієтарних брендів і не використовують широко відомі протоколи.

Після аналізу основних характеристик Domoticz було встановлено, що вона не має суттєвої переваги перед Home Assistant або OpenHab. Хоча колись вона була чудовою платформою, зараз спільнота втратила свою силу. Користувачі все частіше переходять на OpenHab або Home Assistant. Порівняно з іншими двома платформами, Domoticz є менш гнучкою і обмежено підтримує пристрої. На сьогоднішній день, Domoticz не є найкращим вибором.

Проведене дослідження популярних платформ домашньої автоматизації з відкритим вихідним кодом надає можливість створити порівняльну таблицю, яка допоможе систематизувати та чітко відобразити отримані результати дослідження (табл 2.5).

Таблиця 2.5 – Порівняння Home Assistant, OpenHAB, Domoticz

| | Home Assistant | openHAB | Domoticz |
|---|---|---|---|
| Підтримувані пристрої та платформи | Raspberry Pi<br>ODROID<br>ASUS Tinkerboard<br>Generic x86-64<br>Windows<br>macOS<br>Linux<br>Docker<br>VirtualBox<br>KVM/Proxmox<br>VMware<br>ESXi/vSphere | Linux<br>macOS<br>Windows<br>Raspberry Pi<br>Docker | Raspberry Pi<br>Linux<br>Windows |
| Відкритий вихідний код | Так | Так | Так |
| Мова | Python | Java | C++ |
| Мобільні додатки | Android та iOS | Android та iOS | Android та iOS |
| Підтримка | Так | Так | Так |
| Інтеграція зі сторонніми розробниками | 2500+ | 2000+ | Близько 100 |

Популярність платформи має велике значення при виборі системи для управління та моніторингу елементів Інтернету речей (IoT). Це обумовлено тим, що популярні платформи зазвичай мають широку спільноту користувачів, активну підтримку, а також більший набір доступних розширень та можливостей інтеграції з різноманітними пристроями та системами. Вибір популярної платформи може сприяти зручності в її використанні, швидкому розвитку проекту та досвіду, наявних у спільноті користувачів.

Home Assistant надзвичайно популярний, і станом на листопад 2023 року він міг похвалитися понад 270000 активних інсталяцій через вбудований інструмент аналітики. На платформі GitHub репозиторій Home Assistant Core нараховує понад 63000 збережень користувачами, а проектом поділилися понад 25000 разів.

OpenHAB та Domoticz є менш популярними платформами порівняно з Home Assistant. Для збору обширної статистики будемо використовувати різні інструменти. Один з таких інструментів – Google Trends, який надає можливість зібрати статистику щодо популярності та тенденцій у використанні різних платформ (рис. 2.20).


Рисунок 2.20 – Порівняння Home Assistant, OpenHAB та Domoticz у Google Trends за останні 5 років

Використання Google Trends допоможе надати об'єктивну інформацію про популярність згаданих платформ і дозволить оцінити зміни цієї популярності протягом певного періоду. Це дозволить зробити висновки та порівняти різні платформи з точки зору їх впливу та зацікавленості користувачів.

Рейтинг на GitHub відображає загальну популярність проекту, а також його якість та загальну довіру. У наведеній нижче таблиці вказано кількість рейтингових зірок для кожного репозиторію (табл. 2.6).

Таблиця 2.6 – Рейтингу на GitHub

| Платформа | Рейтинг (жовтень 2023) |
|---|---|
| Home Assistant | 63800 |
| OpenHAB | 1800 |
| Domoticz | 3344 |

Розмір спільноти Reddit для кожного продукту не є особливо важливим, але дані в таблиці нижче можуть бути корисними для порівняння між собою (табл. 2.7).

Таблиця 2.7 – Кількість спільноти на Reddit

| Платформа | Розмір спільноти (жовтень 2022) |
|---|---|
| Home Assistant | 255000 |
| OpenHAB | 6000 |
| Domoticz | 972 |

За статистикою за грудень 2021 року, майже 40% активних інсталяцій Home Assistant працюють на Raspberry Pi 4. Загалом, близько 57% інсталяцій Home Assistant працюють на різних моделях Raspberry Pi. Це свідчить про популярність цього міні–ПК серед користувачів.

Друге місце за кількістю інсталяцій (33%) належить віртуальним машинам. Віртуальна машина може бути встановлена на різних пристроях, включаючи NAS (системи зберігання мережевих даних), ноутбуки, а також міні–ПК та інші пристрої. Ця гнучкість дає можливість користувачам вибирати оптимальну платформу для використання Home Assistant залежно від їх потреб та наявних ресурсів.

Аналіз розподілу активних інсталяцій Home Assistant демонструє популярність використання Raspberry Pi, зокрема моделі RPi4, а також можливість встановлення платформи на різних типах пристроїв. Це надає широкий вибір і гнучкість для користувачів при розгляді варіантів реалізації системи моніторингу та управління елементами Інтернету речей з використанням Home Assistant (рис. 2.21).

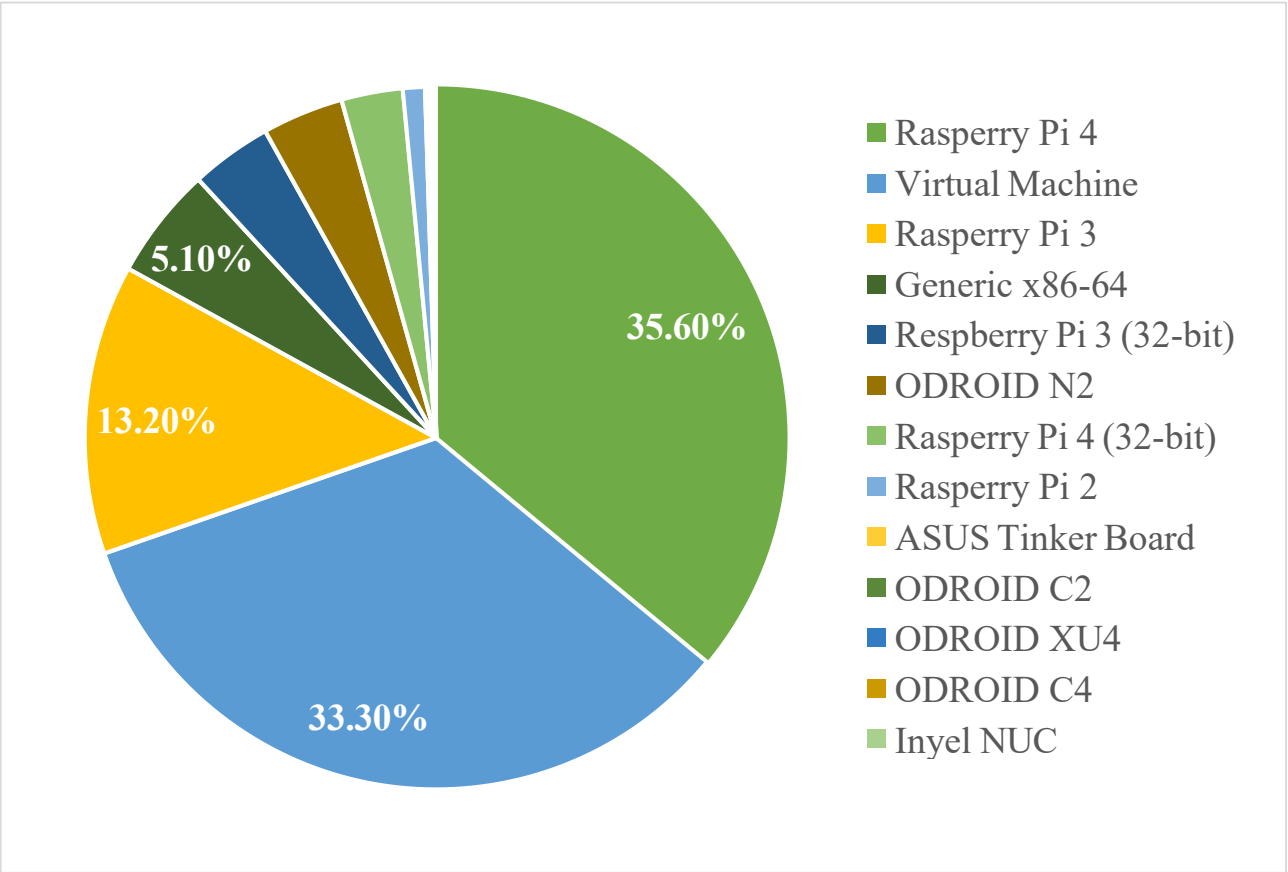


Рисунок 2.21 – Популярність апаратного обладнання для встановлення Home Assistant

Розглянуті платформи пропонують широкий спектр функцій і дозволяють користувачам керувати різноманітними розумними пристроями, але вони відрізняються за простотою використання, сумісністю та підтримкою спільноти. На підставі всієї розглянутої інформації в даному розділі, для майбутньої системи моніторингу та управління IoT було обрано Home Assistant.

# РОЗДІЛ 3
# НАЛАШТУВАННЯ ЗАСОБІВ МОНІТОРИНГУ ТА УПРАВЛІННЯ ЕЛЕМЕНТАМИ ІОТ

## 3.1 Налаштування програмного забезпечення.

Home Assistant є універсальною вільною операційною системою, призначеною для управління пристроями Інтернету речей різних виробників та їх автоматизації. Це локальне рішення, яке не залежить від хмари або доступу до Інтернету, і підтримує широкий спектр протоколів, таких як Wi-Fi, Zigbee, Z–Wave, Bluetooth, Bluetooth Mesh, IR та інші.

У Home Assistant існує розмаїття сценаріїв його використання. Наприклад, він може бути використаний для керування безпекою, які включають камери, системи сигналізації та різноманітні датчики, такі як датчики задимлення, відкриття, руху та вібрації [7].

Також, це програмне забезпечення може використовуватись для керування освітленням, забезпечуючи можливість регулювати яскравість, колір, теплоту світла та підсвітки.

Крім того, система може бути застосована для керування побутовою технікою, такою як кондиціонери, пилососи, телевізори та чайники. Home Assistant також дозволяє керувати електроживленням, контролюючи розетки, вимикачі, реле та лічильники. Система може служити для моніторингу різних параметрів, таких як температура, вологість, тиск, рівень CO2 та виявлення протікання.

Програмне забезпечення Home Assistant можна встановити за допомогою чотирьох рекомендованих методів установки.

Home Assistant OS є рекомендованим та найлегшим способом встановлення. Це мінімальна операційна система, спеціально оптимізована для роботи з Home Assistant. Хоча вона має обмеження на встановлення стороннього програмного

забезпечення та контейнерів, не було визначено жодних недоліків. За даними, цей варіант встановлення використовується 67% користувачів.

Home Assistant Container – це автономний спосіб встановлення, який працює на основі контейнерів і не використовує магазин додатків. Однією з переваг такого способу є легкість управління контейнерами, наприклад, за допомогою Portainer. За кількістю встановлень, цей варіант займає друге місце.

Home Assistant Core – це програма, яку можна використовувати на різних операційних системах. Доступний у вигляді Docker–образу. Проте, використання його не дозволяє встановлювати Docker–контейнери, додатки або створювати резервні копії [7].

Home Assistant Supervised – це варіант встановлення, який надає всі функції Home Assistant OS і може бути встановлений на Linux. Єдина відмінність полягає у тому, що він не надає автоматичного оновлення пакетів [7].

Таблиця 3.1 – Порівняння типів встановлення ПЗ Home Assistant

| Тип ПЗ Home Assistant | OS | Container | Core | Supervised |
|---|---|---|---|---|
| Автоматизації | Підтримується | Підтримується | Підтримується | Підтримується |
| Панель Lovelace | Підтримується | Підтримується | Підтримується | Підтримується |
| Інтеграції | Підтримується | Підтримується | Підтримується | Підтримується |
| Проекти | Підтримується | Підтримується | Підтримується | Підтримується |
| Використання контейнерів | Підтримується | Підтримується | Не підтримується | Підтримується |
| Супервізор | Підтримується | Не підтримується | Не підтримується | Підтримується |
| Доповнення | Підтримується | Не підтримується | Не підтримується | Підтримується |
| Резервні копії | Підтримується | Не підтримується | Не підтримується | Підтримується |
| Керування ОС | Підтримується | Не підтримується | Не підтримується | Не підтримується |

Так як іншого функціоналу та можливостей від міні–ПК не досліджується, окрім керування та моніторингу елементів IoT, було обрано тип встановлення програмного забезпечення Home Assistant OS.

Спершу потрібно підготувати карту пам’яті. Для цього заходимо на офіційний сайт виробника Raspberry Pi та завантажуємо інсталятор ОС та ПЗ Raspberry Pi Imager для відповідної операційної системи.

Raspberry Pi Imager – це програма, яка монтує образ операційної системи на SD картку. Завантажуємо і встановлюємо програму (рис. 3.1).

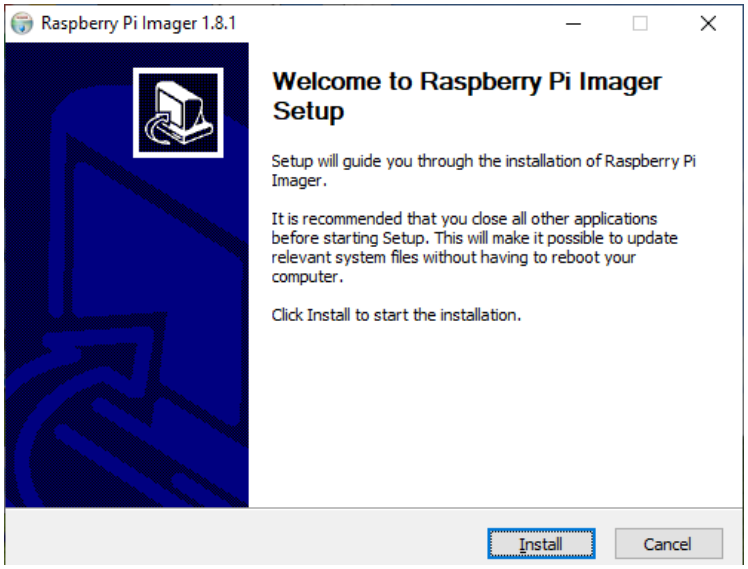


Рисунок 3.1 – Вікно встановлення програми Raspberry Pi Imager версії 1.8.1

Відкриваємо програму Raspberry Pi Imager, вибираємо тип обладнання – Raspberry Pi 4, операційну систему Home Assistant, носія, в нашому випадку це карта пам’яті micsoSD та чекаємо деякий час встановлення ОС (рис. 3.2).

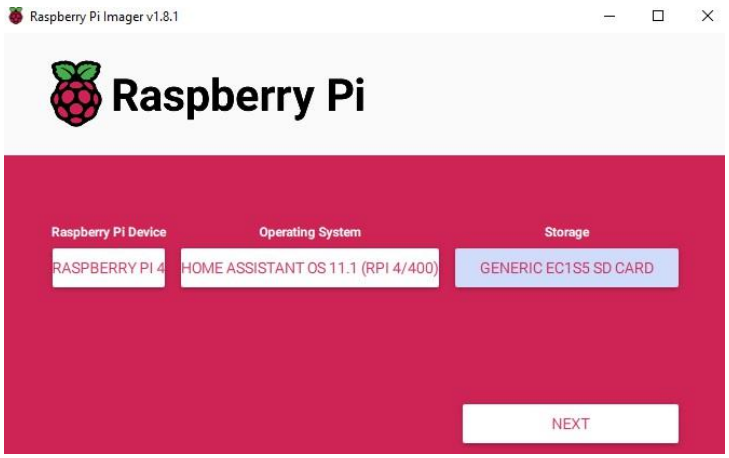


Рисунок 3.2 – Вікно програми Raspberry Pi Imager версії 1.8.1

Після чого встановлюємо готову SD–карту в Raspberry Pi, підключаємо адаптер ZigBee, кабелі Ethernet та живлення для запуску. У браузері протягом кількох хвилин отримуємо доступ до Home Assistant за адресою homeassistant.local:8123 або http://XXXX:8123, де XXXX IP–адреса Raspberry Pi (рис. 3.3).

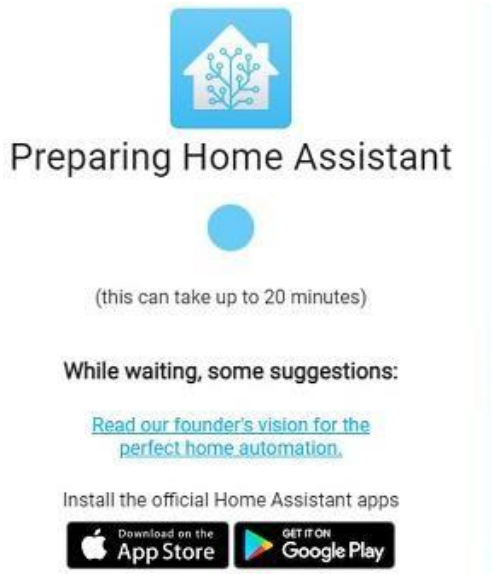


Рисунок 3.3 – Завантаження ОС Home Assistant

Після успішної установки система запропонує створити обліковий запис (рис. 3.4). Цей обліковий запис буде адміністратором і завжди зможе все змінити [7]. Вводимо ім’я, ім’я користувача, пароль і натискаємо CREATE ACCOUNT (Створити обліковий запис):

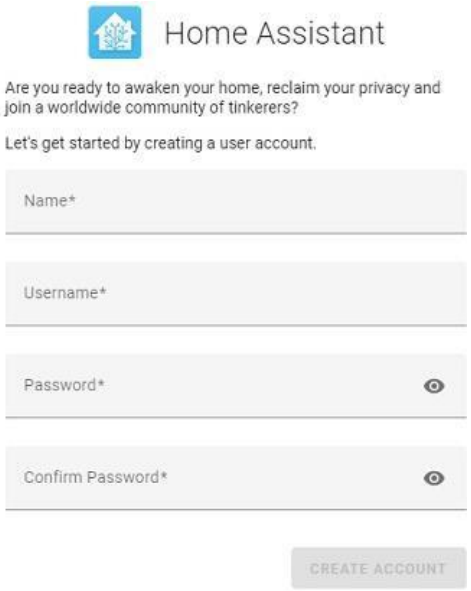


Рисунок 3.4 – Налаштування Home Assistant

Далі вводимо назву свого дому та встановлюємо своє розташування та систему одиниць. Натискаємо «DETECT», щоб знайти своє місцезнаходження та встановити часовий пояс і систему одиниць вимірювання на основі цього місцезнаходження. Також можна встановити ці значення вручну (рис. 3.5).

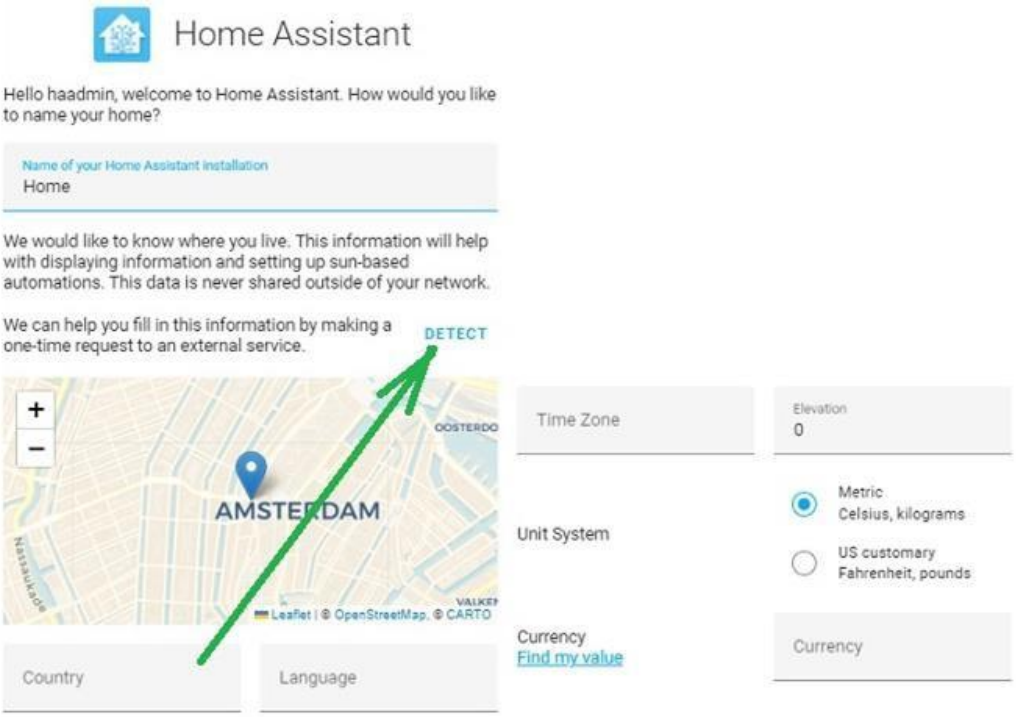


Рисунок 3.5 – Налаштування Home Assistant

Після проведення всіх базових налаштувань робимо авторизацію за логіном та паролем, встановлений нами (рис. 3.6).

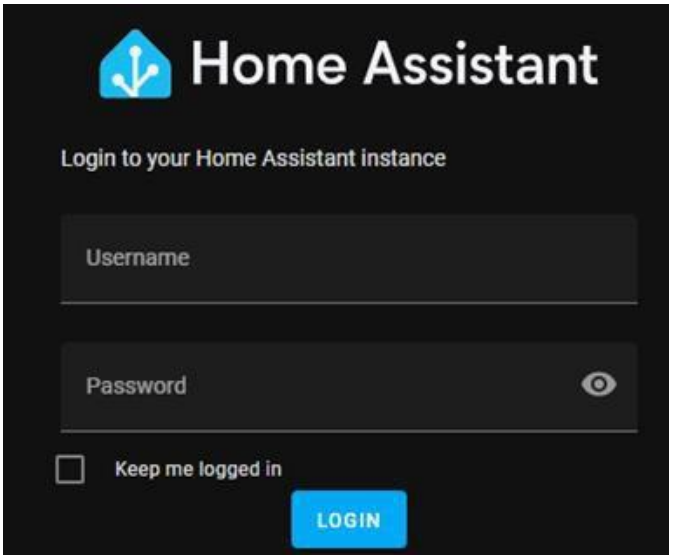


Рисунок 3.6 – Сторінка авторизації Home Assistant

Як і було описано раніше, в якості координатору ZigBee мережі було обрано адаптер Sonoff ZBDongle–E, але для його роботи потрібно правильне налаштування. Є два способи для налаштування ZBDongle–E в Home Assistant: інтеграція через Zigbee Home Automation та доповнення Zigbee2MQTT.

ZigBee Home Assistant (ZHA) – це вбудований компонент Home Assistant, який має просте налаштування. Він підтримує меншу кількість пристроїв (приблизно 800), але має практично повне покриття популярних моделей. Для налаштування ZHA необхідно прошити один з підтримуваних ZigBee адаптерів.

ZigBee2MQTT – це програмне забезпечення координатора ZigBee мережі, яке дозволяє локально керувати пристроями різних виробників. Воно поєднує мережу Zigbee з MQTT–протоколом. Завдяки якісній підтримці, воно підтримує велику кількість пристроїв (2245) та відрізняється надійністю. Для його роботи необхідно мати один з підтримуваних шлюзів [7].

Було обрано доповнення Zigbee2MQTT через його стабільну, надійну роботу та велику кількість підтримуваних пристроїв. Для правильної роботи Zigbee2MQTT потрібний брокер MQTT. Mosquitto є рекомендованим брокером MQTT [7].

Для налаштування потрібно перейти в «Додатки», «Магазин доповнень», обрати та встановити Mosquito broker (рис. 3.7).

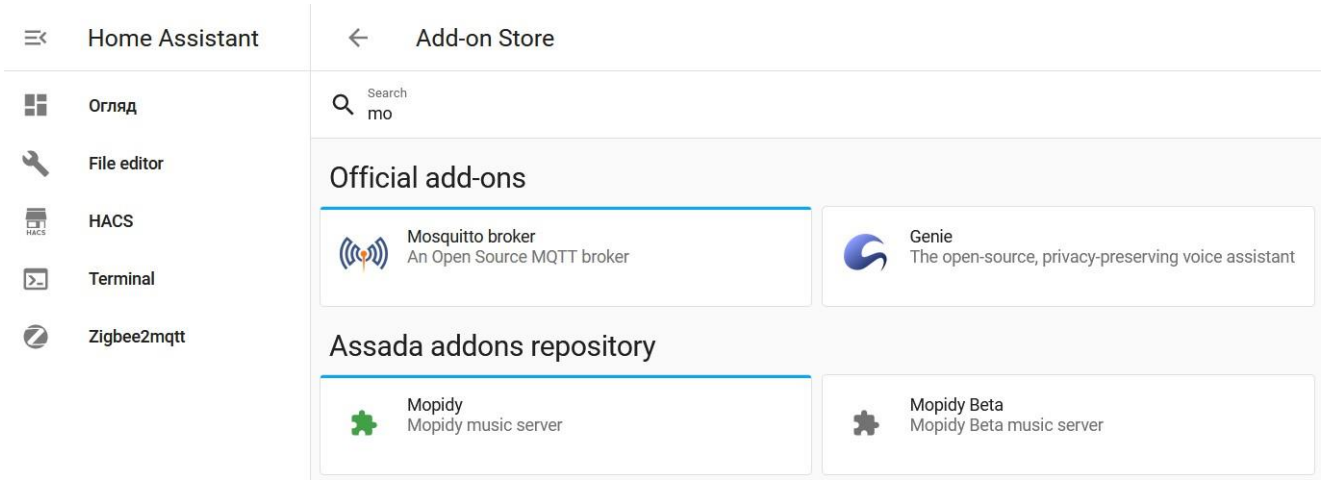


Рисунок 3.7 –Встановлення доповнення Mosquito broker

Після чого потрібно відкрити сторінку додатку Mosquito broker, вкладку «Configuration». У параметрах в пункті «logins» вписуємо: username: mqtt та password: mqtt (рис. 3.8).

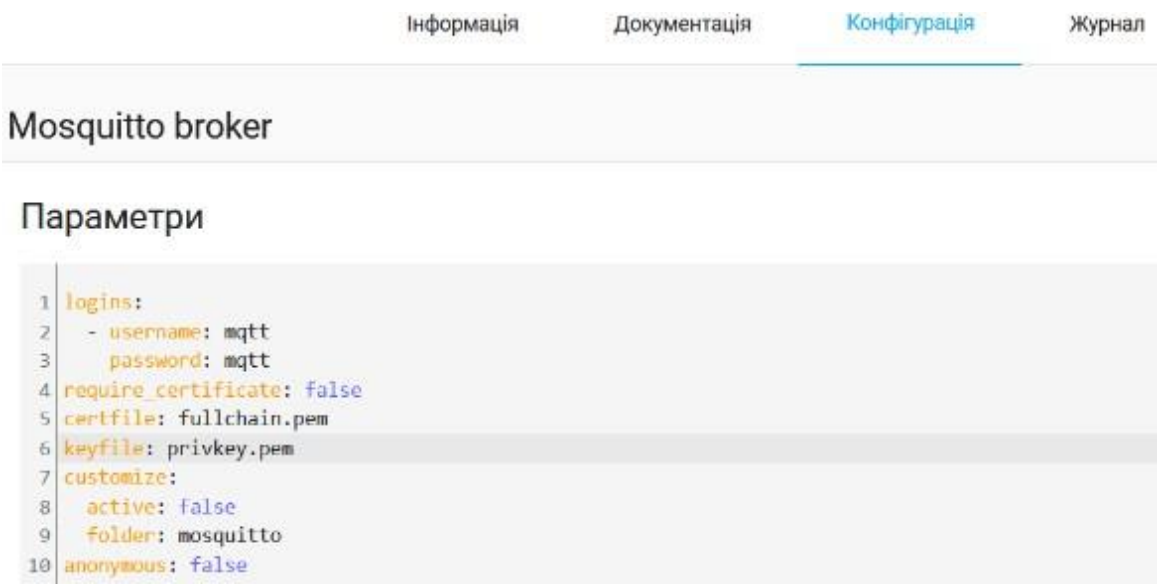


Рисунок 3.8 – Конфігурація доповнення Mosquito broker

Для встановлення ZigBee2MQTT – додаємо репозиторій ZigBee2MQTT: налаштування, «Додатки», «Магазин доповнень». Натискаємо меню зверху справа, «Репозиторії». У пустій стрічці знизу вставляємо адресу репозиторію https://github.com/Zigbee2mqtt/hassio-Zigbee2mqtt та натискаємо «додати» (рис. 3.9).

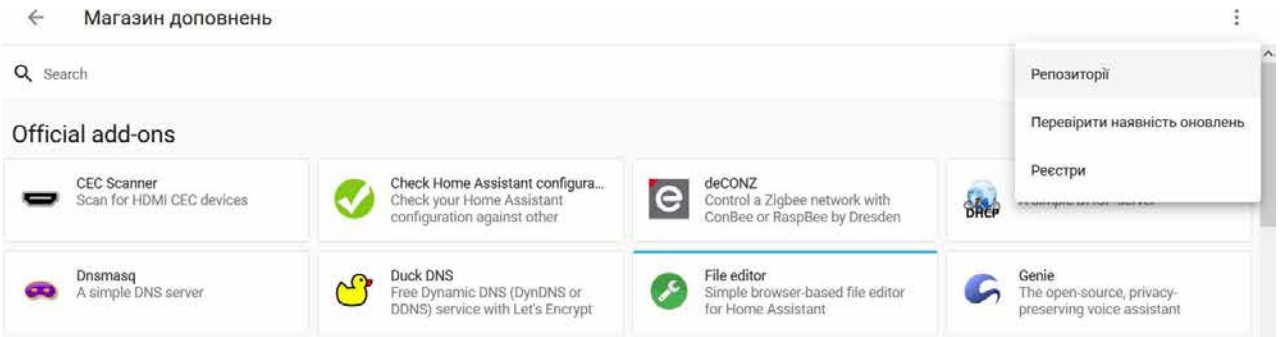


Рисунок 3.9 – Встановлення ZigBee2MQTT

В магазині доповнень з’являться два нових додатки ZigBee2MQTT Edge – це розробницька гілка додатку, у ній раніше з’являється підтримка нових пристроїв. Натисніть та встановіть звичайну стабільну версію ZigBee2MQTT (рис. 3.10).

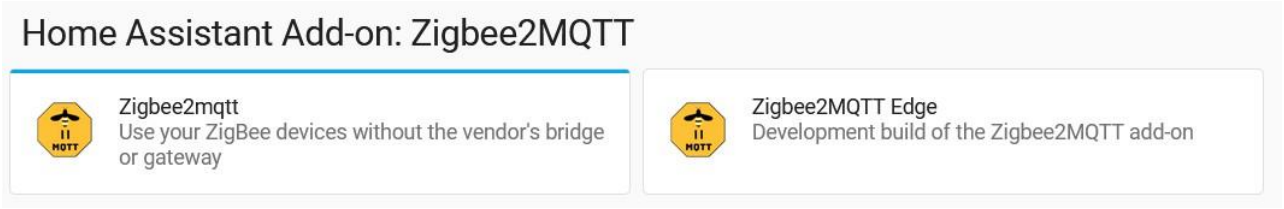


Рисунок 3.10 – Встановлення ZigBee2MQTT

Перейдіть на сторінку додатку ZigBee2MQTT, вкладку «Configuration». У параметрах потрібно написати наступні дані:

Лістинг 3.1 – Встановлення ZigBee2MQTT

```
base_topic: zigbee2mqtt
server: mqtt://core-mosquitto
user: mqtt
password: mqtt
```

Лістинг 3.2 – Налаштування в режимі YAML

```
data_path: /config/zigbee2mqtt
socat:
  enabled: false
  master: pty,raw,echo=0,link=/tmp/ttyZ2M,mode=777
  slave: tcp-
listen:8485,keepalive,nodelay,reuseaddr,keepidle=1,keepintvl=1,keepcnt
=5
  options: '-d -d'
  log: false
mqtt:
  base_topic: zigbee2mqtt
  server: <a href="mqtt://core-mosquitto">mqtt://core-
mosquitto</a>
  user: mqtt
  password: mqtt
serial:
  port: /dev/ttyACM0
```

Заходимо в доповнення ZigBee2MQTT та додаємо датчики ZigBee, включивши дозвіл на приєднання в мережу. Після додавання в мережу додані датчики розташовуємо у відповідних місяцях приміщення (рис. 3.11).

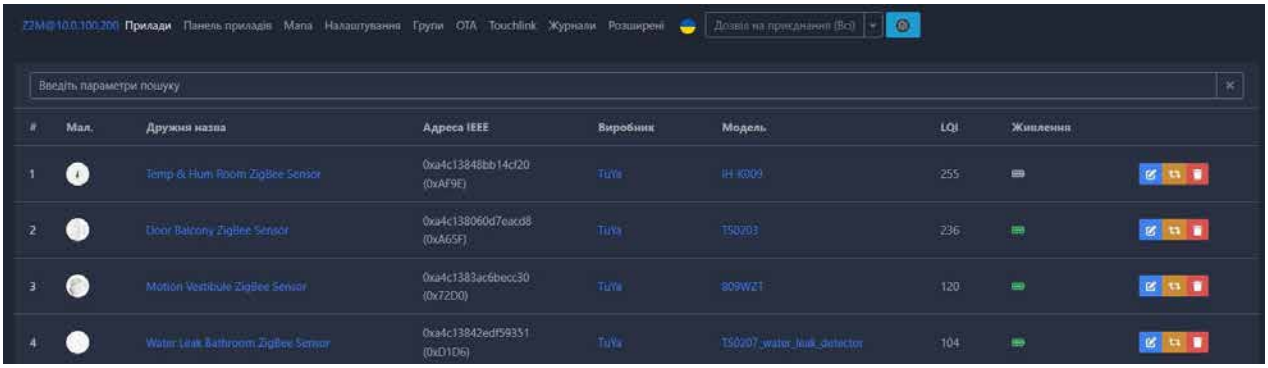


Рисунок 3.11 – Головна сторінка доповнення ZigBee2MQTT

Завантажуємо мапу для відображення підключених датчиків ZigBee у мережі (рис. 3.12).

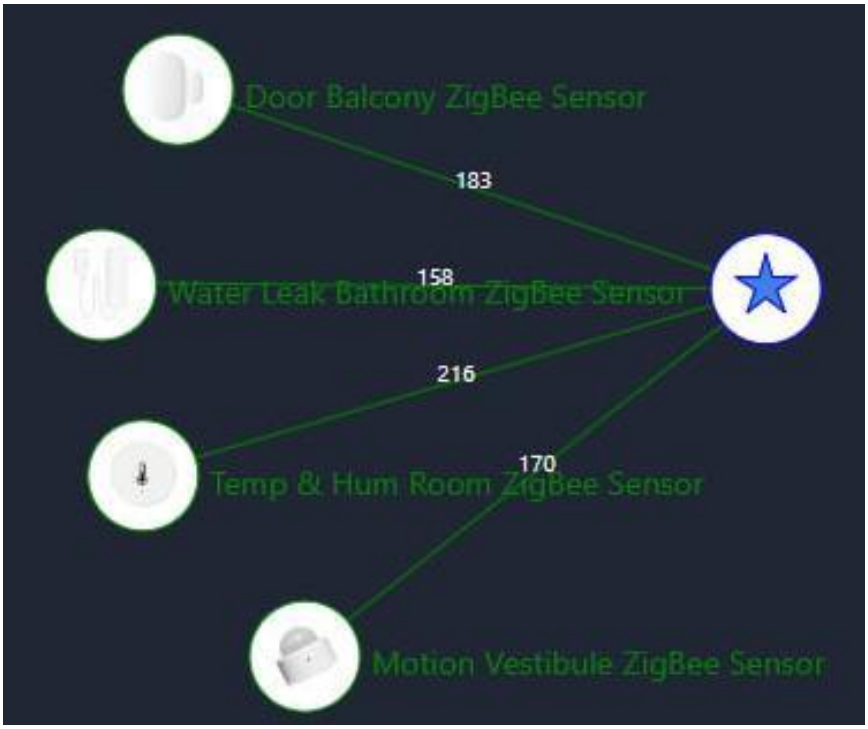


Рисунок 3.12 – Мапа налаштованої мережі ZigBee

В даному розділі було розглянуто налаштування програмного забезпечення Home Assistant, яке є потужним і гнучким інструментом для автоматизації та контролю різних пристроїв у мережі IoT.

## 3.2. Створення сценаріїв автоматизації та тестування системи моніторингу та управління елементами IoT

Для створення сценаріїв переходимо в «Налаштування», «Автоматизація та сцени», «Створити автоматизацію».

Спершу налаштуємо сценарій витоку води, для цього в якості тригеру вибираємо «сутність» і вказуємо раніше доданий датчик протікання Tuya TS0207. Умови роботи не вказуємо, для того, щоб сценарій працював завжди. В дію обираємо мобільний телефон та відправку сповіщення на нього у разі спрацьовування (рис. 3.13).

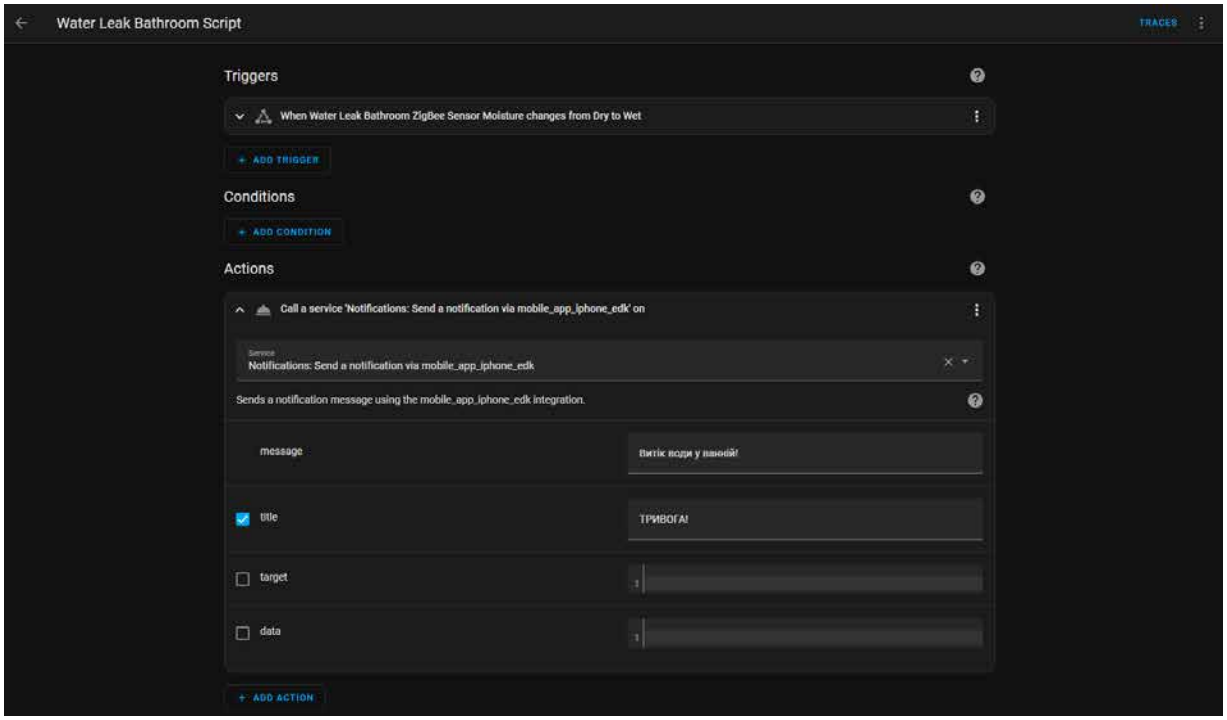


Рисунок 3.13 – Створення сценарію витоку води у ПЗ Home Assistant

Таким же чином створюємо і інші сценарії, в нашому випадку це сповіщення на телефон (встановлений мобільний додаток Home Assistant) у разі спрацьовування датчику руху у тамбурі та відкриття дверей балкону (рис. 3.14).

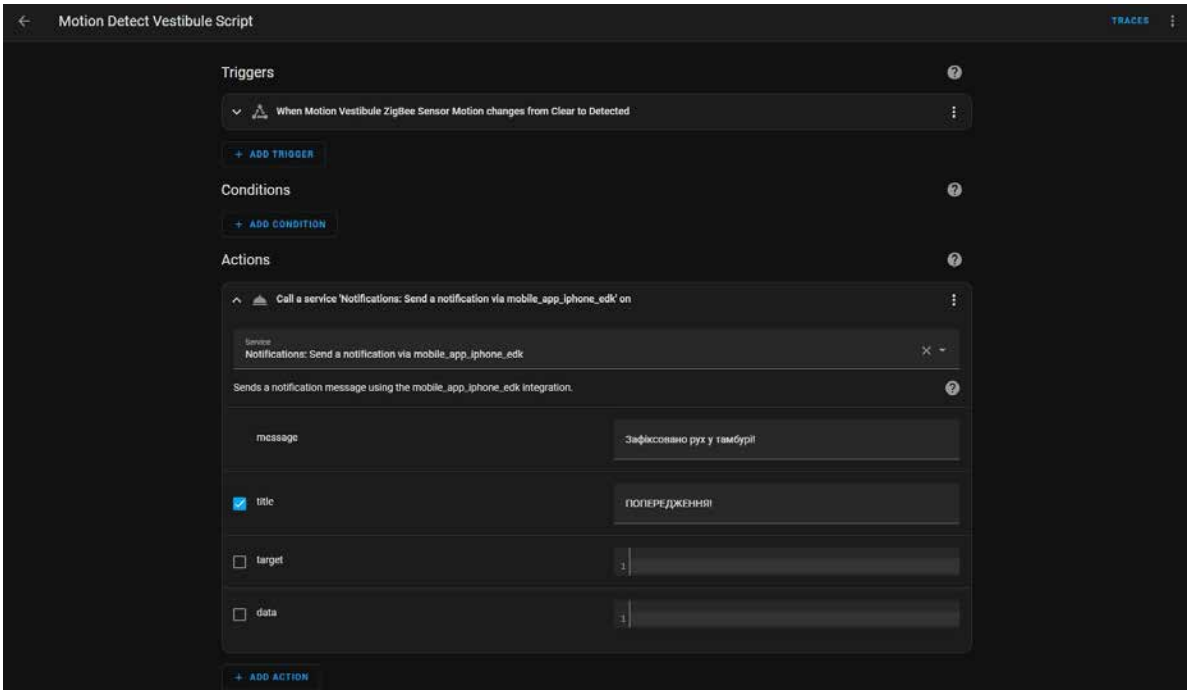


Рисунок 3.14 – Створення сценарію виявлення руху у ПЗ Home Assistant

Для створення сценарію відкриття дверей у програмному забезпеченні Home Assistant, також скористаємося вікном «Створити автоматизацію» (рис. 3.15).

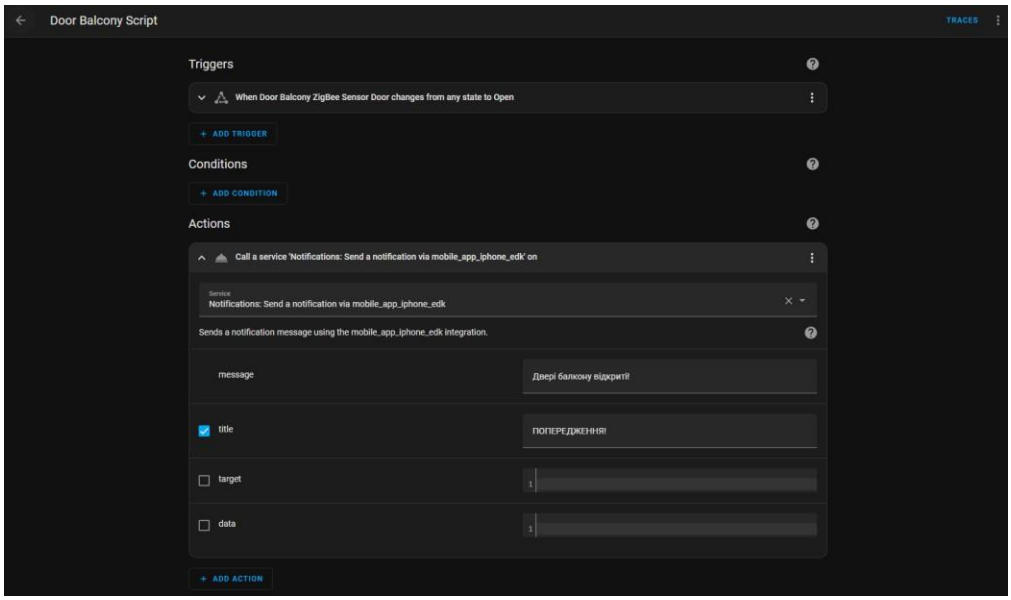


Рисунок 3.15 – Створення сценарію відкриття дверей у ПЗ Home Assistant

У наступному етапі нашого дослідження, було зосереджено увагу на перевірці функціональності розроблених сценаріїв автоматизації. Для цього активуємо кожен з встановлених датчиків. Після активації кожного датчика, спостерігаємо за реакцією

системи та перевірямо, чи надсилаються відповідні сповіщення на смартфон (рис. 3.16).

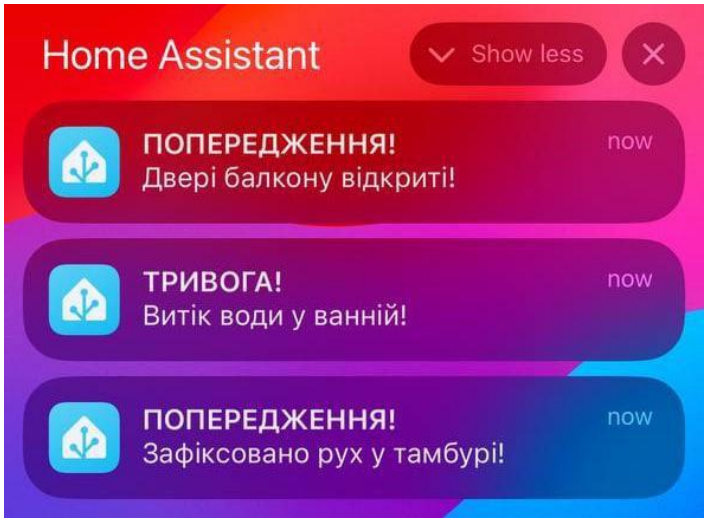


Рисунок 3.16 – Перевірка роботи сценаріїв автоматизації

У даному розділі було проведено налаштування засобів моніторингу. А саме завантажено та встановлено операційну систему на міні–ПК Pasbperry Pi для його подальшої роботи. Ще одним важливим кроком у даному розділі є встановлення та конфігурація Home Assistant.

Варто зазначити, що дане дослідження спрямоване на виявлення мінімальних необхідних засобів та кроків для побудови своєї власної системи управління та моніторингу елементами IoT. Тому описані вище кроки є важливою частиною цього дослідження та мають суттєве значення для його успішного виконання.

## 3.3 Методи удосконалення досліджуваної схеми засобів моніторингу та управління елементами IoT

В системах моніторингу і управління елементами IoT важливу роль відіграє потреба в поліпшенні їх функціональності та ефективності. Поліпшення функціональності означає розширення можливостей системи, включаючи нові

функції, підтримку та додавання різноманітних пристроїв та протоколів, а також забезпечення більш гнучкого налаштування та керування. Це дозволяє користувачам використовувати IoT систему для різних потреб і розширювати її функціональні можливості з часом [7].

Одним із методів поліпшення системи моніторингу та управління елементами IoT є додавання сценаріїв автоматизації. Цей підхід дозволяє забезпечити більш гнучке та розумне управління системою.

Використання сценаріїв автоматизації дозволяє налаштувати різноманітні дії та реакції системи на певні події або умови. Наприклад, можна створити сценарій, що автоматично вмикає опалення, коли температура опускається нижче заданого порогу, або сценарій, що автоматично вимикає освітлення, коли немає присутності людини в приміщенні протягом певного періоду часу.

У основі сценаріїв автоматизації Home Assistant лежить конфігураційний файл YAML. Це формат даних, що використовується для представлення структурованої інформації у зручному форматі. Це текстовий формат, який містить інформацію про компоненти системи, їх конфігурацію та взаємодію між ними. Однак, для створення сценаріїв автоматизації також можна використовувати графічний інтерфейс та брати за основу наявні проекти, які можуть бути досить легко імпортовані.

Основні терміни які використовуються при налаштуванні автоматизації.

Сутність (entities) в системі Home Assistant використовується для представлення пристроїв, датчиків, сервісів та інших елементів, які можуть бути контрольовані. Кожна сутність має свій унікальний ідентифікатор і може мати різні атрибути або властивості, які відображають її стан, змінні дані або іншу інформацію.

Наприклад, сутністю може бути датчик температури, який має атрибути, такі як поточна температура, мінімальна та максимальна температура за добу. Іншим прикладом може бути сутність вмикача, яка може мати стан «увімкнено» або «вимкнено» та додаткові атрибути, які відображають інформацію про стан пристрою (рис. 3.17).

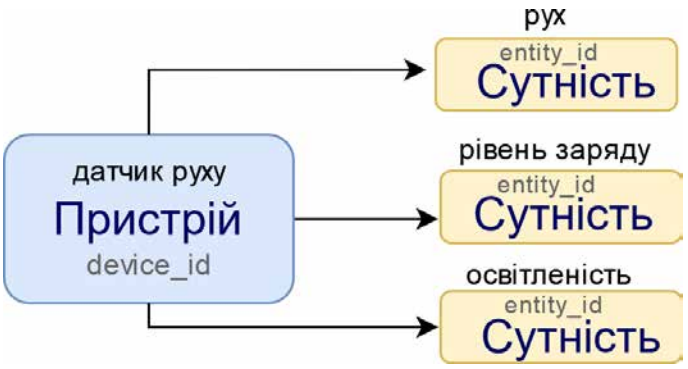


Рисунок 3.17 – Схема сутностей пристрою

Сутності є основою для налаштування автоматизації, створення інтерфейсу користувача та взаємодії з різними пристроями та сервісами в системі Home Assistant.

Зазвичай, автоматизація включає три основні компоненти.

Тригер (trigger) – це подія, яка спричиняє запуск автоматизації. Коли будь–який з тригерів відбувається, Home Assistant перевіряє наявні умови (якщо такі встановлені) і виконує певну дію. Тригером може бути, наприклад, вмикання певної сутності, повернення людини до дому, певний час, тощо [7].

Умова (condition) – це необов'язковий компонент автоматизації, який використовується для здійснення дії лише в певних умовах. Наприклад, це може бути обмеження роботи автоматизації лише у вихідні дні або увімкнення світла тільки після заходу сонця [7].

Дія (action) – це вчинок, який виконується під час активації автоматизації. Дія може включати в себе конкретні вказівки для пристрою, виклик певної послуги, очікування на настання певного часу та інші дії (рис. 3.18).

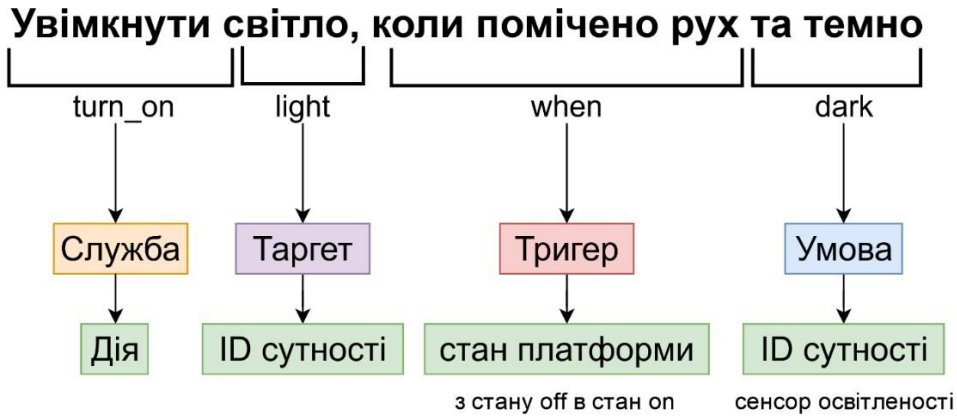


Рисунок 3.18 – Процес автоматизації

Home Assistant пропонує різні типи тригерів, умов і дій для налаштування автоматизацій (табл. 3.2).

Таблиця 3.2 – Тригери, умови, дія в Home Assistant

| Тригер | Умова | Дія |
|---|---|---|
| Home Assistant | Або | if–then |
| MQTT | Не | Активувати сцену |
| Webhook | Та | Вибір |
| Геолокація | Тригер | Виклик служби |
| Зона | Зона | Виклик події |
| Календар | Сонце | Виконати паралельно |
| Подія | Стан | Відтворити |
| Пристрій | Час | Зупинити |
| Сонце | Числовий стан | Очікування спливання часу |
| Стан | Шаблон | Очікування тригера |
| Тег | | Очікування шаблону |
| Час | | Повторення |
| Числовий стан | | Пристрій |
| Шаблон | | Умова |
| Шаблон часу | | |

Додавання сценаріїв автоматизації дозволяє покращити зручність та ефективність використання системи IoT. Користувач може налаштувати бажані дії та реакції системи, що спрощує її управління та забезпечує більш інтелектуальне функціонування.

Дієвим способом поліпшення створеної системи є додавання різноманітних датчиків. Це дозволить системі отримувати більше інформації про оточуюче середовище і реагувати на зміни більш точно. Наприклад, встановлення датчиків

освітленості може допомогти автоматично регулювати освітлення в залежності від рівня освітленості в приміщенні. Система зможе вмикати світло, якщо рівень освітленості впаде нижче заданого порогу.

Датчик газу допоможе виявити наявність певних газів у повітрі, таких як дим, вуглекислий газ або пропан. Це допомагає вчасно виявити потенційно небезпечні ситуації і застосувати відповідні заходи безпеки.

Датчик вогню виявляє наявність вогню або високої температури. В разі спрацювання датчика, він може активувати аварійні сигнали або викликати службу пожежної безпеки.

Важливо зазначити, що система, яка працює на Raspberry Pi з використанням Home Assistant, є досить гнучкою і модифікованою. Home Assistant надає широкі можливості налаштування та інтеграції з різними пристроями і платформами. Завдяки цьому, систему можна легко розширити, додавши нові датчики, налаштувати скрипти та автоматизації, а також інтегрувати її з іншими системами чи сервісами [7].

Розширення системи додатковими датчиками дозволить отримати більше інформації, покращити контроль та автоматизацію різних аспектів домашнього середовища та забезпечити більш високий рівень безпеки і комфорту.

Доцільно також додати додаткові координатори, або ZigBee шлюзи, що дозволить розширити зону покриття мережі IoT. Обидва варіанти мають свої переваги і вибір залежить від вимог і особливостей самої системи.

Додавання додаткових координаторів дозволяє розширити покриття мережі, оскільки кожен координатор може підключати до себе датчики та пристрої в своїй зоні дії. Це особливо корисно, коли система IoT охоплює велику територію або складається з різних зон, наприклад, великого будинку чи офісного приміщення зі зв'язаними, але віддаленими частинами. Додаткові координатори дозволяють забезпечити стабільний зв'язок та збільшити кількість підключених пристроїв.

З іншого боку, встановлення ZigBee шлюзу також може покращити покриття мережі IoT. Шлюз може виступати як центральна вузлова точка, яка взаємодіє з координатором і пристроями у системі. Він може бути розташований в ключовому місті, що дозволяє розширити зону покриття та підключати пристрої, які знаходяться

далеко від основного контролера. Завдяки шлюзу можна забезпечити стійкий та надійний зв'язок з пристроями, що знаходяться на віддаленій відстані [7].

Загалом, додавання додаткових координаторів допомагає розширити покриття мережі IoT, а встановлення ZigBee шлюзу сприяє централізованій комунікації та забезпечує більш гнучкий зв'язок з віддаленими пристроями. Вибір між ними залежить від конкретних потреб та вимог системи IoT.

# ВИСНОВКИ

Проведення дослідження системи моніторингу та управління елементами Інтернет речей у магістерській роботі дозволило зробити наступні висновки та запропонувати пропозиції, які мають як теоретичне, так і практичне значення. Ці висновки та пропозиції спрямовані на полегшення управління та моніторингу елементів Інтернету речей, а також на виявлення необхідних засобів для роботи такої системи.

У результаті проведених досліджень стало зрозуміло, які програмні та апаратні засоби необхідні для побудови надійної та ефективної системи управління та моніторингу елементами IoT. Важливим аспектом є визначення оптимального балансу між вартістю та якістю матеріальних витрат для успішного впровадження даної системи.

Дане дослідження є ефективними і корисними для використання, оскільки в ньому наведено конкретні пристрої, програмні засоби та кроки, необхідні для синтезу програмного та апаратного забезпечення. Це дає змогу використовувати отримані результати для особистих цілей та побудови власної мережі Інтернет речей.

У першому розділі магістерської роботи було проведено теоретичне дослідження архітектурних рішень мережі Інтернет речей. Ретельно розглянуто різні технології, протоколи та типи мереж, які використовуються в контексті IoT. Виявлено, що існує різноманітність пристроїв, які взаємодіють між собою, збирають, обробляють та передають дані.

Розглянуті основні елементи інфраструктури IoT, які включають датчики, контролери, мережу, хмарні платформи, користувацькі програми та аналітику даних. Особлива увага була приділена моніторингу, яка є важливою складовою частиною всієї системи. Розглянуто різні методи та засоби моніторингу, спрямовані на забезпечення ефективності та надійності функціонування мережі Інтернет речей.

Одним із ключових висновків розділу є необхідність вибору відповідної технології для побудови мережі. Виявлено, що вибір технології повинен

здійснюватися з урахуванням вимог системи, характеристик пристроїв та інфраструктури, а також масштабу проекту. У результаті було отримано значний обсяг теоретичних знань про архітектурні рішення мережі Інтернет речей. Ці знання є важливою основою для подальшого розвитку та вдосконалення систем моніторингу та управління у сфері IoT.

У другому розділі магістерської роботи було проведено аналіз, апаратного обладнання та засобів моніторингу для управління елементами Інтернет речей. Для забезпечення ефективної роботи системи необхідно вибрати відповідне апаратне забезпечення, яке зможе виконувати поставлені завдання та досягати поставлених цілей. Інтернет речей – це не просто набір різноманітних пристроїв і датчиків, які взаємодіють між собою через дротові та бездротові канали зв'язку. Це складне поєднання реального та віртуального світу, де відбувається взаємодія між людьми і пристроями. У зв'язку з цим, було запропоновано використовувати міні–ПК Raspberry Pi як основу для системи. Він надає всі необхідні можливості для роботи системи, є економічним та портативним варіантом для побудови системи. Використання Raspberry Pi дозволяє забезпечити необхідну функціональність системи і впоратися з поставленими завданнями. Крім того, він є економічним і портативним варіантом, що є важливими факторами при створенні системи управління та моніторингу елементів IoT.

Зроблено порівняльний аналіз різних системи моніторингу і управління елементами IoT та обрано Home Assistant як платформу для розробки системи. Це потужна система домашньої автоматизації, що дозволяє контролювати і налаштовувати різноманітні пристрої в домашньому середовищі. Вона надає широкий спектр можливостей, зокрема віддалене керування освітленням, вимикачами, датчиками та лічильниками, такими як температура, вологість, рух, якість повітря, енергоспоживання та інше. Home Assistant також підтримує інтеграцію з різними виробниками пристроїв, що дозволяє об'єднати різні пристрої від різних вендорів в єдину систему управління. Вона має зручний веб-інтерфейс та мобільні додатки, які дозволяють керувати та налаштовувати систему з будь–якого

місця. Дозволяє створювати автоматичні правила, графіки споживання енергії, отримувати сповіщення про події та багато іншого.

Дане дослідження має велике практичне значення, оскільки надає змогу зрозуміти, яке обладнання необхідне для побудови системи управління та моніторингу, які програмні комплекси використовуються в цьому процесі, а також які кроки потрібно зробити, щоб система належним чином працювала.

У третьому розділі магістерської роботи було розроблено схему моніторингу та управління елементами IoT з використанням технології ZigBee, мікрокомп'ютера Raspberry Pi та програмного забезпечення Home Assistant. Використання цієї схеми дозволяє створити ефективну та гнучку систему моніторингу та управління, яка може працювати з різними типами пристроїв та забезпечувати широкі можливості налаштування та контролю.

У роботі також запропоновано методи удосконалення досліджуваної схеми і включають такі заходи:

– сценаріїв автоматизації дозволяє системі виконувати певні дії автоматично відповідно до заданих умов. Це сприяє зручності та ефективності в управлінні різними пристроями та процесами в середовищі IoT;

– додавання різноманітних датчиків дозволить системі отримувати більше інформації про оточуюче середовище і реагувати на зміни більш точно;

– додавання координаторів і шлюзів ZigBee дозволить розширити зону покриття, створити резервне з'єднання, забезпечити більшу стійкість, розподілити навантаження та дозволить легко розширювати мережу IoT.

В цілому, розроблена схема моніторингу та управління елементами IoT є ефективним інструментом для створення розумного середовища, забезпечуючи зручність, ефективність та безпеку. Пропоновані заходи для покращення ще більше розширять можливості системи та сприятимуть подальшому розвитку сфери IoT технологій.

## СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ


1. Das S. First Things on Internet and Their History. Independently Published, 2019 (pp. 18–22).

2. Internet of Things: Architectures, Protocols and Standards / M. Picone et al. Wiley & Sons, Incorporated, John, 2018. 408 p.

3. LTE Cat–M1. URL: https://www.4gltemall.com/ue-category/lte-cat-m1.html?limit=24 (дата звернення: 11.10.2022).

4. Bluetooth Low Energy (BLE): A Complete Guide. URL: https://novelbits.io/bluetooth-low-energy-ble-complete-guide/ (дата звернення: 12.09.2023)

5. Raspberry Pi Documentation. Official resource. URL: https://www.raspberrypi.com/documentation/ (дата звернення: 04.10.2023).

6. Headman V. Raspberry Pi 4 Advanced Users Guide: The Complete Guide to Mastering the Raspberry Pi 4: Raspberry Pi 4 Guide. Independently Published, 2021. 255 pages.

7. Home Assistant Documentation. Official resource. URL: https://www.home-assistant.io/ (дата звернення: 05.10.2023).

8. Kodali, R. K., Jain, V., Bose, S., & Boppana, L. (2016, April). IoT based smart security and home automation system. In 2016 international conference on computing, communication and automation (ICCCA) (pp. 1286-1289). IEEE.

9. Main Page domoticz. Official resource. URL: https://www.domoticz.com/wiki/Main_Page (дата звернення: 13.09.2023)

10. openHAB Documentation. Official resource. URL: https://www.openhab.org/docs/ (дата звернення: 13.09.2023).

11. Pavithra, D., & Balakrishnan, R. (2015, April). IoT based monitoring and control system for home automation. In 2015 global conference on communication technologies (GCCT) (pp. 169-173). IEEE.

12. Saha, S., Ishraque, H., Islam, M.T., & Rahman, M.A. (2019). IoT based smart home automation and energy management. In 2019 Thesis & Report, BSc (Electrical and Electronic Engineering) (Department of Electrical and Electronic Engineering, Brac University) P. 85

13. Uncovering IoT Threats in the Cybercrime Underground // Trend Micro Research. – 2019. URL: https://documents.trendmicro.com/assets/white_papers/wp-the-internet-of-things-in-the-cybercrime-underground.pdf (дата звернення: 20.09.2023).

15. Основні проблеми розумних будинків і як їх можна вирішити? // Кластер. Інженетрі системи та мережі. – 2019. URL: https://klaster.ua/ua/stati-i-obzory/osnovnye-problemy-umnyhdomov-i-kak-ih-mozhno-reshit/ (дата звернення: 05.02.2023).

16. Полякова О.В. Класифікація функціональних складових елементів системи інтелектуального керування середовищем при проектуванні житла // Вісник Київського національного університету технологій та дизайну. Серія: Технічні науки. – 2016. – № 4. – С. 133–141.

17. MQTT Essentials. URL: https://www.hivemq.com/mqtt-essentials/ (дата звернення: 15.03.2023).

18. Kranz M. Building the Internet of Things: Implement New Business Models, Disrupt Competitors, Transform Your Industry. Wiley & Sons, Incorporated, John, 2016. 272 c.

19. Madakam S. Internet of Things: Smart Things. International Journal of Future Computer and Communication. 2015. Vol. 4, no. 4. P. 250–253. URL: http://www.ijfcc.org/vol4/395-ICNT2014-2-203.pdf (дата звернення: 03.11.2022).

20. Eltayeb M. (2017). Privacy and Security. In DawsonM.M. EltayebOmarM. (Eds.), Security Solutions for Hyperconnectivity and the Internet of Things (pp. 89–112).

21. Valg af hardware til Home Assistant. URL: https://smart-home-guide.dk/index.php/2022/08/30/valg-af-hardware-til-home-assistant/ (дата звернення: 27.09.2023).

22. Norris D. Internet of Things: Do-It-Yourself at Home Projects for Arduino, Raspberry Pi and BeagleBone Black. McGraw-Hill Education, 2015. 352 p.

24. Elahi A., Gschwender A. ZigBee Wireless Sensor and Control Network. Pearson Education, Limited, 2021.

25. IoT Monitoring. URL: https://www.macrometa.com/iot-infrastructure/iot-monitoring (дата звернення: 17.09.2023).

26. How to Choose the Right IoT Device Management Platform? URL: https://patrickhq.medium.com/how-to-choose-the-right-iot-device-management-platform-9404088bdc85 (дата звернення: 19.09.2023).

27. Github home-assistant/core. URL: https://github.com/home-assistant/core (дата звернення: 27.10.2023).

28. Github OpenHAB/openhab-core. URL: https://github.com/openhab/openhab-core (дата звернення: 27.10.2023).

29. Github domoticz URL: https://github.com/domoticz/domoticz (дата звернення: 27.10.2023).

30. Zigbee 3.0 USB Dongle Plus–ZBDongle–E EFR32MG21. URL: https://sonoff.tech/product/gateway-and-sensors/sonoff-zigbee-3-0-usb-dongle-plus-e/ (дата звернення: 13.03.2023).

31. Датчик температури та вологості TuYa IH–K009. URL: https://www.zigbee2mqtt.io/devices/IH-K009.html (дата звернення: 13.03.2023).

32. Датчик відкриття ZigBee Tuya TS0203. URL: https://www.zigbee2mqtt.io/devices/TS0203.html (дата звернення: 13.03.2023).

33. Датчик руху ZigBee TuYa 809WZT. URL: https://www.zigbee2mqtt.io/devices/809WZT.html (дата звернення: 13.03.2023).

34. Датчик протікання води ZigBee Tuya TS0207. URL: https://www.zigbee2mqtt.io/devices/TS0207_water_leak_detector.html (дата звернення: 13.03.2023).